



Funktionsbausteine für PSx-3__ mit POWERLINK-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 PSx_type | 5 |
| 3 | Fehlerbeschreibung (Error ID) | 6 |
| 4 | Beschreibung und Anwendung der Funktionsbausteine | 8 |
| | 4.1 Bibliothek „h_w“ hinzufügen | 8 |
| | 4.2 Variablen für die einzelnen Positionierantriebe anlegen | 10 |
| | 4.3 Zuordnung der Kanalnamen zu Prozessvariablen („I/O-Mapping“) | 11 |
| | 4.4 Belegen der Variablen „Device“ und „NodeID“ | 11 |
| | 4.5 Instanzen der gewünschten Funktionsbausteine anlegen | 12 |
| | 4.6 Belegen der Eingangsvariablen „pDrive“ | 12 |
| | 4.7 Belegen der weiteren Ein- und Ausgangsvariablen | 12 |
| | 4.8 Gemeinsamkeiten aller Funktionsbausteine | 13 |
| | 4.9 Verriegelungen zwischen den Funktionsbausteinen | 13 |
| | 4.10 hwMC_Move | 13 |
| | 4.11 hwMC_Error | 16 |
| | 4.12 hwMC_ReadParameter | 17 |
| | 4.13 hwMC_Parametrization | 21 |
| | 4.14 hwMC_PositionParametrization | 25 |
| 5 | Beispielprojekt | 28 |

Bedeutung der Betriebsanleitung

Diese Betriebsanleitung erläutert die Funktionsbausteine für die Positioniersysteme PSx-3__-PL (mit POWERLINK-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.

© 2016

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__-PL eignen sich besonders zur automatischen Einstellung von Werkzeugen, Anschlägen oder Spindeln bei Holzverarbeitungsmaschinen, Verpackungsmaschinen, Druckmaschinen, Abfüllanlagen und bei Sondermaschinen.

Die PSx-3__-PL 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 PSx_type

Für jeden Antrieb gibt es eine Datenstruktur, in der einige Daten eines Antriebs abgelegt sind. Für jeden Antrieb wird eine Instanz dieser Struktur benötigt. Diese Instanz muss jedem FB übergeben werden, der auf den betr. Antrieb wirkt. Dazu wird der Eingangsvariable „pDrive“ des betr. FBs die Adresse dieser Instanz von PSx_type zugewiesen.

Beispiel: `hwMC_Move_1.pDrive := ADR(Motor_1);`

Hiermit soll z.B. sichergestellt werden, dass nicht gleichzeitig zwei Zugriffe aus unterschiedlichen FBs auf den Servicedatenkanal eines Antriebs durchgeführt werden können.

Des Weiteren werden mit Hilfe dieser Datenstruktur die Zuordnung der Ein-/Ausgangsdaten des jeweiligen Antriebs zu Programmvariablen hergestellt.

Datenstruktur PSx_type:

| Parametername | Datentyp | geschrieben von | Beschreibung |
|---------------|-------------|--------------------|---|
| NodeID | USINT | Benutzer | Knotennummer des Antriebs |
| Device | STRING[30] | Benutzer | Devicename der POWERLINK-Schnittstelle (z.B. „IF3“) |
| In | PSxIn_type | s.u. | Eingangsdaten (Daten vom Antrieb zur SPS) |
| Out | PSxOut_type | s.u. | Ausgangsdaten (Daten von der SPS zum Antrieb) |
| State | USINT | Funktionsbausteine | gibt an, ob aktuell einer der Funktionsbausteine den Servicedatenkanal zu diesem Antrieb belegt = 0 → SDO-Kanal nicht belegt > 0 → SDO-Kanal belegt |

Datenstruktur PSxIn_type:

| Parametername | Datentyp | geschrieben von | Beschreibung |
|----------------|----------|-----------------|---|
| ModuleOK | BOOL | SPS | Verbindung zum Antrieb FALSE → nicht ok TRUE → ok |
| StatusWord | UINT | Antrieb | Statuswort |
| ActualPosition | DINT | Antrieb | aktuelle Position |
| ActualSpeed | INT | Antrieb | aktuelle Drehzahl in U/min |

Datenstruktur PSxOut_type:

| Parametername | Datentyp | geschrieben von | Beschreibung |
|----------------|----------|-----------------|--------------|
| ControlWord | UINT | Benutzer | Steuerwort |
| TargetPosition | DINT | Benutzer | Sollposition |

3 Fehlerbeschreibung (Error ID)

Nachfolgend die Fehlercodes, die von den Funktionsbausteinen ausgegeben werden:

| ErrorID (hex) | Beschreibung |
|-----------------------|---|
| 16xF000 (mask) | FB |
| 16#1xxx | Error in hwMC_Move |
| 16#2xxx | Error in hwMC_Error |
| 16#3xxx | Error in hwMC_ReadParameter |
| 16#4xxx | Error in hwMC_Parametrization |
| 16#5xxx | Error in hwMC_PositionParametrization |
| 16#0F00 (mask) | Internal FB and PD errors |
| 16#x1xx | Error in state machine or other 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#x6xx | Unallowed input data change |
| 16#00F0 (mask) | Parameter errors |
| 16#xx1x | Parameter: SDO protocol timed out |
| 16#xx2x | Parameter: attempt to read a write only object |
| 16#xx3x | Parameter: attempt to write a read only object |
| 16#xx4x | Parameter: object does not exist in the object dictionary |
| 16#xx5x | Parameter: access failed due to an hardware error |
| 16#xx6x | Parameter: length of service parameter does not match |
| 16#xx7x | Parameter: Sub-index does not exist |
| 16#xx8x | Parameter: value range of parameter exceeded |
| 16#xx9x | Parameter: value of parameter written too high |
| 16#xxAx | Parameter: value of parameter written too low |
| 16#xxBx | Parameter: access failed due to the present device state |
| 16#xxFx | Parameter: 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:

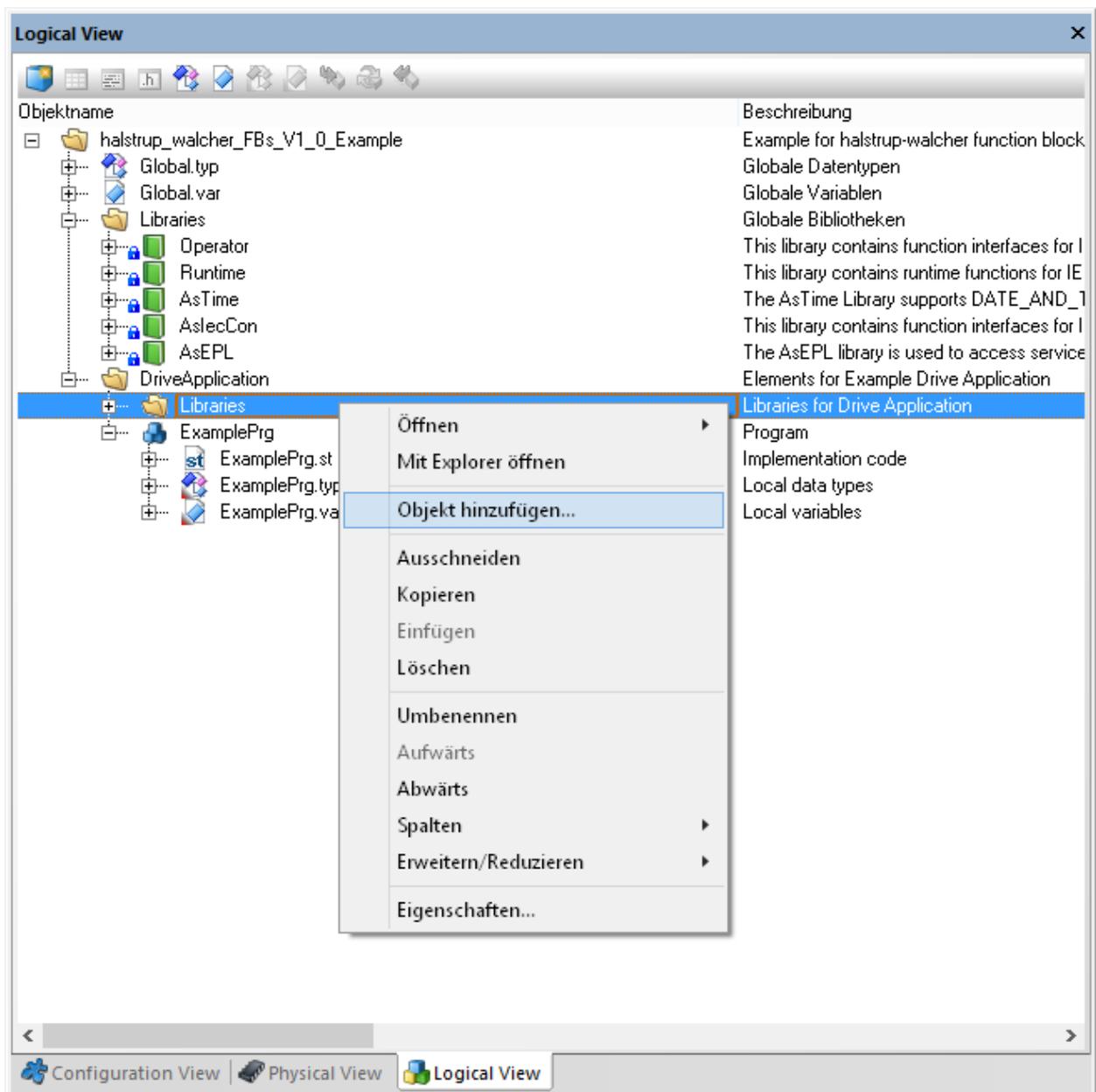
- Fahrauftrag (hwMC_Move) mit falschem Sollwert → ErrorID = 16#1008
- Parameter schreiben (hwMC_Parametrization) mit zu hohem Wert → ErrorID = 16#4090

4 Beschreibung und Anwendung der Funktionsbausteine

Dieses Kapitel umfasst zunächst eine Beschreibung, wie die Funktionsbausteine verwendet werden, anschließend folgt eine detaillierte Beschreibung der einzelnen Bausteine.

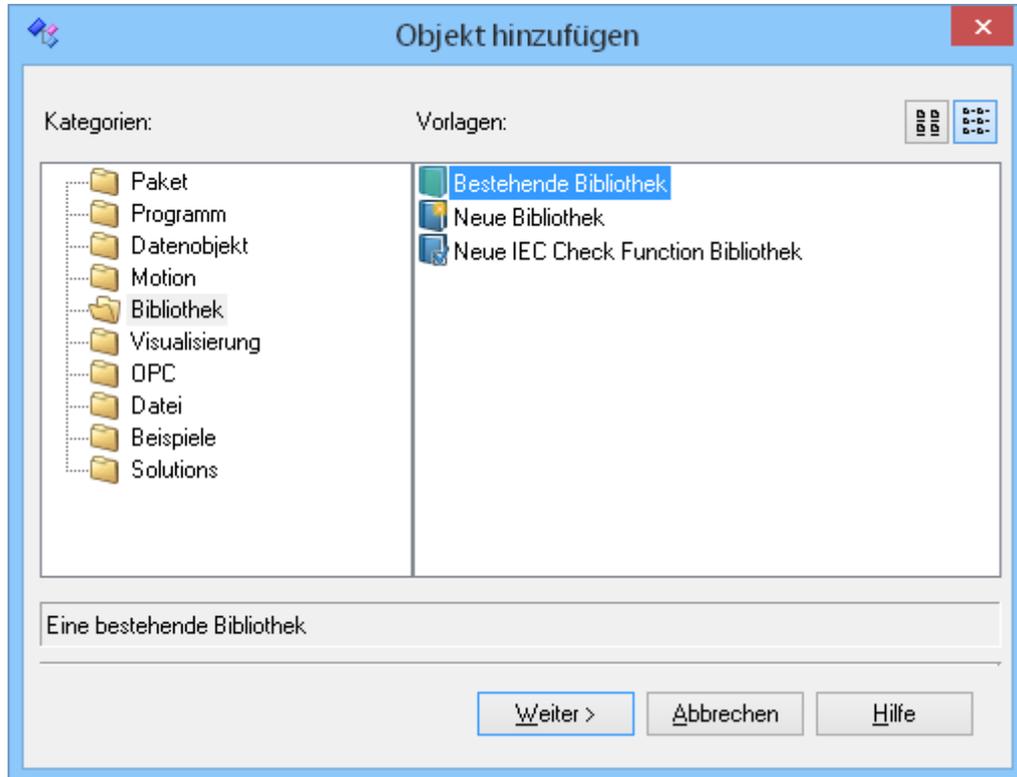
4.1 Bibliothek „h_w“ hinzufügen

Im ersten Schritt wird die Bibliothek „h_w“ dem Anwenderprojekt hinzugefügt, dies geschieht folgendermaßen:



Zunächst die Bibliothek von der Website von halstrup-walcher downloaden. Dann ggf. im Automation-Studio-Projekt ein Unterverzeichnis „Libraries“ für eingebundene Bibliotheken

erzeugen (falls im Projekt noch kein geeigneter Ort für eingebundene Bibliotheken vorhanden ist). In diesem Verzeichnis dann mit Rechtsklick das Pop-Up-Menü öffnen und dort auf „Objekt hinzufügen“ klicken.



Dann auf „Bibliothek → bestehende Bibliothek“ klicken. Im folgenden Dialog dann den Button „Projektordner“ anwählen, zur Bibliothek „h_w“ browsen, diese selektieren und auf „Weiter“ klicken.

Im folgenden Dialog dann die Bibliothek der aktiven CPU zuordnen und fertigstellen.



Falls die B&R-Bibliothek „AsEPL“ zu diesem Zeitpunkt noch nicht Bestandteil des Projekts ist, wird diese automatisch hinzugefügt. AsEPL stellt Mechanismen bereit, die von der Bibliothek „h_w“ benötigt werden. Ggf. kann AsEPL vom antriebsspezifischen Library-Verzeichnis in das projektübergreifende Library-Verzeichnis verschoben werden, um anzuzeigen, dass AsEPL ggf. auch von anderen Applikationen innerhalb des Projekts verwendet werden kann.

Im Ergebnis präsentiert sich die Bibliothek h_w dann folgendermaßen:

| Objektname | Beschreibung |
|-----------------------------------|--------------------------------------|
| halstrup_walcher_FBs_V1_0_Example | Example for halstrup-walcher func |
| Global.typ | Globale Datentypen |
| Global.var | Globale Variablen |
| Libraries | Globale Bibliotheken |
| Operator | This library contains function inter |
| Runtime | This library contains runtime funct |
| AsTime | The AsTime Library supports DAT |
| AslecCon | This library contains function inter |
| AsEPL | The AsEPL library is used to acce |
| DriveApplication | Elements for Example Drive Appli |
| Libraries | Libraries for Drive Application |
| h_w | halstrup-walcher PSx3xxPL functi |
| h_w.typ | Exported data types |
| PSx_type | |
| PSxIn_type | |
| PSxOut_type | |
| h_w.fun | Exported functions and function b |
| hwMC_Error | Reports the state of the drive |
| hwMC_Move | Starts a positioning run or manual |
| hwMC_Parametrization | Writes one or more parameters to |
| hwMC_PositionParametrization | Writes parametrization of position |
| hwMC_ReadParameter | Reads one or more parameters fr |
| ExamplePrg | Program |
| ExamplePrg.st | Implementation code |
| ExamplePrg.typ | Local data types |
| ExamplePrg.var | Local variables |

4.2 Variablen für die einzelnen Positionierantriebe anlegen

Für jeden Antrieb, der von einem oder mehreren Funktionsbausteinen angesprochen werden soll, ist eine Variable vom Typ „PSx_type“ anzulegen. Diese Variablen können global angelegt werden, oder lokal in demjenigen Programm, welches die Funktionsbausteine ausführt.

Für zwei Antriebe wären z.B. die folgenden Variablen anzulegen (hier in der Sprache „ST“):

```
Motor_1: PSx_type
Motor_2: PSx_type
```

4.3 Zuordnung der Kanalnamen zu Prozessvariablen („I/O-Mapping“)

Die Funktionsbausteine greifen auf die Ein- und Ausgangsdaten der verwendeten Antriebe zu. Im Datentyp „PSx_type“ sind bereits die entspr. Elemente vorgesehen, diese müssen aber noch mit den Kanälen der Antriebe verknüpft werden. Bei den beiden Antrieben gemäß obigem Beispiel ergeben sich folgende Zuordnungen:

Zuordnung der Kanäle von Motor 1:

```
ModuleOk → Motor_1.In.ModuleOK
TargetPosition_I2001Out → Motor_1.Out.TargetPosition
ActualPosition_I2003In → Motor_1.In.ActualPosition
ControlWord_I2024Out → Motor_1.Out.ControlWord
StatusWord_I2025 → Motor_1.In.StatusWord
ActualSpeed_I2030 → Motor_1.In.ActualSpeed
```

Zuordnung der Kanäle von Motor 2:

```
ModuleOk → Motor_2.In.ModuleOK
TargetPosition_I2001Out → Motor_2.Out.TargetPosition
ActualPosition_I2003In → Motor_2.In.ActualPosition
ControlWord_I2024Out → Motor_2.Out.ControlWord
StatusWord_I2025 → Motor_2.In.StatusWord
ActualSpeed_I2030 → Motor_2.In.ActualSpeed
```

Diese Zuordnungen werden durch Auswählen des entspr. Powerlink-Knotens in der Physical View, Öffnen des Pop-Up-Menüs mit Rechtsklick und Auswählen von „I/O Zuordnung“ angelegt.



Diese Knoten müssen zuvor wie gewohnt z.B. mit dem System Designer angelegt worden sein. Dazu ist die Gerätebeschreibungsdatei der PSx3xxPL (die sog. XDD-Datei) in den Hardware-Katalog von Automation Studio zu laden. Diese Datei ist auf der Website von halstrup-walcher abrufbar.

4.4 Belegen der Variablen „Device“ und „NodeID“

Im Datentyp „PSx_type“ sind zwei Elemente vorhanden, auf die die Funktionsbausteine zugreifen:

- Knotennummer des Antriebs
- Devicename der POWERLINK-Schnittstelle (z.B. „IF3“)

Wenn man annimmt, dass die beiden Motoren die Knotennummern 1 und 2 haben und an der Standard-Powerlink-Schnittstelle angeschlossen sind, wäre in dem obigen Beispiel die Belegung folgendermaßen vorzunehmen (hier in der Sprache „ST“):

```
Motor_1.NodeID := 1;
Motor_1.Device := 'IF3';
Motor_2.NodeID := 2;
Motor_2.Device := 'IF3';
```

Diese Belegung ist während dem Ablauf des Programms unveränderlich, in einem ST-Programm kann sie daher z.B. im INIT-Teil untergebracht werden.

4.5 Instanzen der gewünschten Funktionsbausteine anlegen

Für jeden Funktionsbaustein, der im Programm verwendet werden soll, ist nun eine Instanz pro Antrieb anzulegen.

Falls z.B. die Bausteine `hwMC_Move`, `hwMC_Error`, `hwMC_ReadParameter` und `hwMCPositionParametrization` für jeweils zwei Antriebe vorgesehen sind, sind z.B. die folgenden Instanzen von FBs anzulegen (hier in der Sprache „ST“):

```
hwMC_Move_1 : hwMC_Move
hwMC_Error_1 : hwMC_Error;
hwMC_ReadParameter_1 : hwMC_ReadParameter;
hwMC_PositionParametrization_1 : hwMC_PositionParametrization;
hwMC_Move_2 : hwMC_Move;
hwMC_Error_2 : hwMC_Error;
hwMC_ReadParameter_2 : hwMC_ReadParameter;
hwMC_PositionParametrization_2 : hwMC_PositionParametrization;
```

4.6 Belegen der Eingangsvariablen „pDrive“

Die Eingangsvariable „pDrive“ stellt den Bezug zwischen der betr. Instanz des Funktionsbausteins und der gewünschten Achse dar. „pDrive“ muss bei jedem Funktionsbaustein belegt werden, in dem obigen Beispiel wäre die Belegung folgendermaßen vorzunehmen (hier in der Sprache „ST“):

```
hwMC_Move_1.pDrive := ADR(Motor_1);
hwMC_Error_1.pDrive := ADR(Motor_1);
hwMC_ReadParameter_1.pDrive := ADR(Motor_1);
hwMC_PositionParametrization_1.pDrive := ADR(Motor_1);
hwMC_Move_2.pDrive := ADR(Motor_2);
hwMC_Error_2.pDrive := ADR(Motor_2);
hwMC_ReadParameter_2.pDrive := ADR(Motor_2);
hwMC_PositionParametrization_2.pDrive := ADR(Motor_2);
```

Diese Belegung ist während dem Ablauf des Programms unveränderlich, in einem ST-Programm kann sie daher z.B. im INIT-Teil untergebracht werden.

4.7 Belegen der weiteren Ein- und Ausgangsvariablen

Die restlichen Eingangsvariablen werden entweder direkt mit Werten beschrieben oder es wird eine Variable gleichen Typs deklariert und mit der Eingangsvariable verknüpft.

In derselben Weise können Ausgangsvariablen entweder direkt ausgewertet werden oder es wird eine Variable gleichen Typs deklariert und mit der Ausgangsvariable verknüpft.

Es müssen nicht alle Ein- und Ausgänge beschaltet werden. Wenn Teile eines Bausteins nicht benötigt werden, können die zugehörigen Eingänge unbeschaltet bleiben, es gilt dann der jeweilige Anfangswert für diesen Eingang. Nicht benötigte Ausgänge können ebenfalls offen bleiben.

4.8 Gemeinsamkeiten aller Funktionsbausteine

Die in Kap. 4.5 angelegten Instanzen der Funktionsbausteine müssen nun in einer zyklischen Task aufgerufen werden.

Der Eingang „pDrive“ ist in den folgenden Beschreibungen der einzelnen FBs nicht mehr gesondert aufgeführt.

4.9 Verriegelungen zwischen den Funktionsbausteinen

Die Bausteine sind z.T. gegeneinander verriegelt. Dadurch ist z.B. sichergestellt, dass nicht gleichzeitig zwei Zugriffe aus unterschiedlichen FBs auf den Servicedatenkanal eines Powerlink-Knotens durchgeführt werden können.

Dabei gilt die Regel, dass wenn ein FB aufgerufen wird, dieser so lange im Zustand „warten“ verharrt, bis der Servicedatenkanal frei ist. Dieser belegt dann den Servicedatenkanal für die Dauer eines SDO-Zugriffs.

Die Belegung des Servicedatenkanals führt dazu, dass bei den FBs „hwMC_Move“, „hwMC_ReadParameter“, „hwMC_Parametrization“ und „hwMC_PositionParametrization“ der Ausgang „Active“ aktiv wird.

Der FB „hwMC_Error“ benötigt den Servicedatenkanal nicht.

„hwMC_Move“ nutzt den Servicedatenkanal, um sicherzustellen, dass ein Fahrauftrag auch vom Antrieb übernommen wurde. Die anderen FBs nutzen SDOs zum Schreiben und Lesen von Parametern.

4.10 hwMC_Move

Dieser FB dient der Positionierung des Antriebs.

| Symbol | Name | Datentyp | Belegbar | Output-Typ |
|--------|--------------------------|----------|-------------------------------------|------------|
| → | pDrive | UDINT | <input type="checkbox"/> | VAR_INPUT |
| → | Release | BOOL | <input type="checkbox"/> | VAR_INPUT |
| → | Position | DINT | <input type="checkbox"/> | VAR_INPUT |
| → | ManualRunToLargerValues | BOOL | <input type="checkbox"/> | VAR_INPUT |
| → | ManualRunToSmallerValues | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ↔ | Active | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ↔ | InPosition | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ↔ | ActualPosition | DINT | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ↔ | Error | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ↔ | ErrorID | UINT | <input checked="" type="checkbox"/> | VAR_OUTPUT |

Release

Freigabe des Antriebs

- Typ: BOOL
- Anfangswert: FALSE
- Art: 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: 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.



Um den gleichen Sollwert z.B. nach einem Blockieren anzufahren, muss die Freigabe „Release“ zurückgesetzt und erneut gesetzt werden. Zwischen Zurücksetzen und Setzen muss ausreichend Zeit eingeplant werden, so dass der Low-Zustand mindestens ein Mal auf dem Bus übertragen wird.

ManualRunToLargerValues

Handfahrt zu größeren Werten

- Typ: BOOL
- Anfangswert: FALSE
- Art: INPUT

Beschreibung:

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



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: INPUT

Beschreibung:

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



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: 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: 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: 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: OUTPUT

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

ErrorID

Fehler-ID

- Typ: UINT
- Art: OUTPUT

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



Error und ErrorID von hwMC_Move werden stets aktualisiert – auch dann, wenn der Eingang „Release“ nicht gesetzt ist.

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 |

4.11 hwMC_Error

Dieser FB gibt den Status des Antriebs und des FBs als Fehlerbit, Fehler-ID („ErrorID“) und als Text aus. Falls sowohl hwMC_Error als auch hwMC_Move aktiviert sind, ist die Fehler-ID von hwMC_Error stets dieselbe wie diejenige des Bausteins hwMC_Move.

| FB hwMC_Error | | | | |
|---------------|------------------|------------|--------------------------|------------|
| | pDrive | UDINT | <input type="checkbox"/> | VAR_INPUT |
| | Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | Error | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorID | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorDescription | STRING[80] | <input type="checkbox"/> | VAR_OUTPUT |

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: INPUT

Error

Fehler bei der Ausführung des FB oder Fehler im Antrieb

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

ErrorID

Fehler-ID (siehe folgende Tabelle „ErrorID“)

- Typ: UINT
- Defaultwert: 0
- Art: OUTPUT

| | | | |
|--------------------------------|------|--------------------------|-----------|
| DeviceVariant_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| SoftwareVersion_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ControlWord_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| TargetPosition_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DirRotation_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| PosScaleNumerator_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| PosScaleDenominator_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ReferencingValue_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| UpperMappingEnd_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| UpperLimit_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| LowerLimit_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| PositionWindow_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| LoopLength_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DragError_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Readjustment_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DragErrorCorrection_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeed_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeedManual_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| SpeedLimitAbort_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Acceleration_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Deceleration_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxStartTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorqueEndRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| AbortRunTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| StartTorqueTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorqueEndRunTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DirectionChangeTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| BreakReleaseTimeEndRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MotVoltageFilterTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_1_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_2_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_3_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_4_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_5_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_6_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_7_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_8_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_9_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| GeneralPurpose_10_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MinVoltage_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxTemperature_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |

| | | | | |
|--|---------------------------------|------------|--------------------------|------------|
| | Address_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | ConfigConnectionTimeout_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | SafePositionConnTimeout_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | RepetitionTimeSafePosRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | DeliveryState_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | Active | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | Done | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | Error | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorID | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorParameter | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | StatusWord_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ActualSpeed_Value | INT | <input type="checkbox"/> | VAR_OUTPUT |
| | ActualPosition_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ActualTorque_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | MaxTorqueLastRun_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | U_Control_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | U_Motor_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | DeviceTemperature_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ProductionDate_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | SerialNumber_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | DeviceVariant_Value | STRING[31] | <input type="checkbox"/> | VAR_OUTPUT |
| | SoftwareVersion_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ControlWord_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | TargetPosition_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | DirRotation_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | PosScaleNumerator_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | PosScaleDenominator_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ReferencingValue_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | UpperMappingEnd_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | UpperLimit_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | LowerLimit_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | PositionWindow_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | LoopLength_Value | DINT | <input type="checkbox"/> | VAR_OUTPUT |
| | DragError_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | Readjustment_Value | USINT | <input type="checkbox"/> | VAR_OUTPUT |
| | DragErrorCorrection_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | TargetSpeed_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | TargetSpeedManual_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | SpeedLimitAbort_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | Acceleration_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | Deceleration_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | MaxStartTorque_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | MaxTorque_Value | UINT | <input type="checkbox"/> | VAR_OUTPUT |

| | | | | |
|---|--------------------------------|-------|---|------------|
| ◆ | MaxHoldTorqueEndRun_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | MaxHoldTorque_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | AbortRunTime_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | StartTorqueTime_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | MaxHoldTorqueEndRunTime_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | DirectionChangeTime_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | BreakReleaseTimeEndRun_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | MotVoltageFilterTime_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_1_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_2_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_3_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_4_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_5_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_6_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_7_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_8_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_9_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | GeneralPurpose_10_Value | UDINT | ■ | VAR_OUTPUT |
| ◆ | MinVoltage_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | MaxTemperature_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | Address_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | ConfigConnectionTimeout_Value | UINT | ■ | VAR_OUTPUT |
| ◆ | SafePositionConnTimeout_Value | DINT | ■ | VAR_OUTPUT |
| ◆ | RepetitionTimeSafePosRun_Value | DINT | ■ | VAR_OUTPUT |
| ◆ | DeliveryState_Value | INT | ■ | VAR_OUTPUT |

Execute

Start eines Lesevorgangs

- Typ: BOOL
- Anfangswert: FALSE
- Art: INPUT

Beschreibung:

Bei einer steigenden Flanke wird ein Lesevorgang gestartet. Es werden nacheinander diejenigen Parameter gelesen, bei denen der zugehörige FB-Eingang gesetzt ist (Endung „..._Enable“). Der gelesene Wert ist am jeweiligen Ausgang des FBs abrufbar. Beim Auftreten eines Fehlers werden keine weiteren Parameter gelesen und der Wert des aktuell gelesenen Parameters bleibt unverändert (Defaultwert: 0).

Für einen erneuten Lesevorgang muss erneut eine steigende Flanke generiert werden. Wird das Bit zurückgesetzt, so bleiben die gelesenen Werte weiterhin erhalten.

Active

Bit ist gesetzt, solange der Lesevorgang läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird zurückgesetzt, sobald alle angeforderten Werte gelesen wurden oder ein Fehler aufgetreten ist oder „Execute“ zurückgesetzt wird.

Done

Bit ist gesetzt, sobald alle angeforderten Werte erfolgreich gelesen wurden und an „..._Value“ anliegen

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

Error

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

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

ErrorID

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

- Typ: UINT
- Defaultwert: 0
- Art: OUTPUT

Antriebsfehler („Drive errors“) werden beim Lesen eines Parameters nicht beachtet. Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

ErrorParameter

SDO-Nr. desjenigen Parameters, der im Fall eines Fehlers beim Lesen den Fehler verursacht hat

- Typ: UINT
- Defaultwert: 0
- Art: OUTPUT

Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

4.13 hwMC_Parametrization

Mit diesem FB können Parameterwerte in den Antrieb geschrieben werden.

| Parameter | Datentyp | Defaultwert | Art |
|----------------------------|----------|-------------|-----------|
| pDrive | UDINT | | VAR_INPUT |
| Execute | BOOL | | VAR_INPUT |
| DeliveryState | BOOL | | VAR_INPUT |
| DirRotation_Enable | BOOL | | VAR_INPUT |
| DirRotation_Value | UINT | | VAR_INPUT |
| PosScaleNumerator_Enable | BOOL | | VAR_INPUT |
| PosScaleNumerator_Value | UINT | | VAR_INPUT |
| PosScaleDenominator_Enable | BOOL | | VAR_INPUT |
| PosScaleDenominator_Value | UINT | | VAR_INPUT |

| | | | |
|--------------------------------|-------|--------------------------|-----------|
| ActualPosition_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ActualPosition_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| ReferencingValue_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ReferencingValue_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| UpperMappingEnd_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| UpperMappingEnd_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| UpperLimit_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| UpperLimit_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| LowerLimit_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| LowerLimit_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| PositionWindow_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| PositionWindow_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| LoopLength_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| LoopLength_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| DragError_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DragError_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| Readjustment_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Readjustment_Value | USINT | <input type="checkbox"/> | VAR_INPUT |
| DragErrorCorrection_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| DragErrorCorrection_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeed_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeed_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeedManual_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| TargetSpeedManual_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| SpeedLimitAbort_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| SpeedLimitAbort_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| Acceleration_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Acceleration_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| Deceleration_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| Deceleration_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| MaxStartTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxStartTorque_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| MaxTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxTorque_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorqueEndRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorqueEndRun_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorque_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorque_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| AbortRunTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| AbortRunTime_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| StartTorqueTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| StartTorqueTime_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| MaxHoldTorqueEndRunTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |

| | | | | |
|---|---------------------------------|-------|-------------------------------------|------------|
| ➔ | MaxHoldTorqueEndRunTime_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | DirectionChangeTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | DirectionChangeTime_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | BreakReleaseTimeEndRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | BreakReleaseTimeEndRun_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MotVoltageFilterTime_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MotVoltageFilterTime_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_1_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_1_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_2_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_2_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_3_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_3_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_4_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_4_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_5_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_5_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_6_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_6_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_7_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_7_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_8_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_8_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_9_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_9_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_10_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | GeneralPurpose_10_Value | UDINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MinVoltage_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MinVoltage_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MaxTemperature_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | MaxTemperature_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | Address_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | Address_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | ConfigConnectionTimeout_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | ConfigConnectionTimeout_Value | UINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | SafePositionConnTimeout_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | SafePositionConnTimeout_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | RepetitionTimeSafePosRun_Enable | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | RepetitionTimeSafePosRun_Value | DINT | <input type="checkbox"/> | VAR_INPUT |
| ➔ | SaveSettings | BOOL | <input type="checkbox"/> | VAR_INPUT |
| ➔ | Active | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ➔ | Done | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ➔ | Error | BOOL | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ➔ | ErrorID | UINT | <input checked="" type="checkbox"/> | VAR_OUTPUT |
| ➔ | ErrorParameter | UINT | <input checked="" type="checkbox"/> | VAR_OUTPUT |

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 geschrieben
- 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“ 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“ 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: 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 wird ein evtl. laufender Parametriervorgang abgebrochen.

DeliveryState

Laden der Werkseinstellungen (zunächst ohne Speichern)

- Typ: BOOL
- Anfangswert: FALSE
- Art: INPUT

x_Enable

Falls gesetzt, wird der betr. Parameter geschrieben

- Typ: BOOL
- Anfangswert: FALSE
- Art: INPUT

x_Value

Sollwert des Parameters

- Anfangswert: 0
- Art: INPUT

Der Datentyp, eine Beschreibung sowie der Wertebereich kann entspr. dem Parameternamen der Betriebsanleitung des PSx-3__-PL entnommen werden.

SaveSettings

Speichern der Einstellungen

- Typ: BOOL
- Anfangswert: FALSE
- Art: INPUT

Active

Bit ist gesetzt, solange die Parametrierung läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird zurückgesetzt, sobald die Parameterierung erfolgreich beendet wurde oder ein Fehler aufgetreten ist oder „Execute“ zurückgesetzt wird.

Done

Bit ist gesetzt, sobald die Parameterierung erfolgreich beendet wurde

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

Error

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

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

ErrorID

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

- Typ: UINT
- Defaultwert: 0
- Art: OUTPUT

Antriebsfehler („Drive errors“) werden bei einer Parametrierung nicht beachtet. Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

ErrorParameter

SDO-Nr. desjenigen Parameters, der im Fall eines Fehlers beim Parameterieren den Fehler verursacht hat

- Typ: INT
- Defaultwert: 0
- Art: OUTPUT

Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

4.14 hwMC_PositionParametrization

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

| | | | | |
|----|------------------------------|-------|--------------------------|------------|
| FB | hwMC_PositionParametrization | | | |
| | pDrive | UDINT | <input type="checkbox"/> | VAR_INPUT |
| | Execute | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | Direction | UINT | <input type="checkbox"/> | VAR_INPUT |
| | StepsPerTurn | UINT | <input type="checkbox"/> | VAR_INPUT |
| | LowerLimit | DINT | <input type="checkbox"/> | VAR_INPUT |
| | UpperLimit | DINT | <input type="checkbox"/> | VAR_INPUT |
| | SetPoint | DINT | <input type="checkbox"/> | VAR_INPUT |
| | SaveSettings | BOOL | <input type="checkbox"/> | VAR_INPUT |
| | Active | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | Done | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | Error | BOOL | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorID | UINT | <input type="checkbox"/> | VAR_OUTPUT |
| | ErrorParameter | UINT | <input type="checkbox"/> | VAR_OUTPUT |

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 (SDO #202C) = Direction
 - Istwertbewertung Zähler (SDO #2010) = 400
 - Istwertbewertung Nenner (SDO #2011) = StepsPerTurn
 - Istwert (SDO #2003) = SetPoint
 - Falls (SetPoint > UpperLimit):
 - oberes Mapping-Ende (SDO #2028) = SetPoint + (3 x StepsPerTurn)
 - sonst:
 - oberes Mapping-Ende (SDO #2028) = UpperLimit + (3 x StepsPerTurn)
 - untere Endbegrenzung (SDO #2017) = LowerLimit
 - obere Endbegrenzung (SDO #2016) = UpperLimit
- Die Anzahl der Schritte pro Umdrehung „StepsPerTurn“ ergibt unmittelbar den Wert des Parameters „Istwertbewertung Nenner“ (SDO #2011). Dabei wird angenommen, dass der Wert von „Istwertbewertung Zähler“ (SDO #2010) 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 |
|---|---------|----------------|
| StepsPerTurn ≥ 1 | 16#50A0 | SDO #2011 |
| StepsPerTurn ≤ 10000 | 16#5090 | SDO #2011 |
| LowerLimit ≤ UpperLimit | 16#50A0 | SDO #2016 |
| $(UpperLimit - LowerLimit) / StepsPerTurn \leq 250$ | 16#50A0 | SDO #2017 |
| Falls SetPoint < LowerLimit: $(UpperLimit - SetPoint) / StepsPerTurn \leq 250$ | 16#50A0 | SDO #2003 |
| Falls SetPoint > UpperLimit: $(SetPoint - LowerLimit) / StepsPerTurn \leq 250$ | 16#5090 | SDO #2003 |

- 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: 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 wird ein evtl. laufender Parametriervorgang abgebrochen.

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: INPUT

StepsPerTurn

Schritte pro Umdrehung an der Abtriebswelle (Auflösung)

- Typ: INT
- Anfangswert: 0
- Art: INPUT

LowerLimit

Untere Endbegrenzung

- Typ: DINT
- Anfangswert: 0
- Art: INPUT

UpperLimit

Obere Endbegrenzung

- Typ: DINT
- Anfangswert: 0
- Art: INPUT

SetPoint

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

- Typ: DINT
- Anfangswert: 0
- Art: INPUT

SaveSettings

Speichern der Einstellungen

- Typ: BOOL

- Anfangswert: FALSE
- Art: INPUT

Active

Bit ist gesetzt, solange die Parametrierung läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird zurückgesetzt, sobald die Parameterierung erfolgreich beendet wurde oder ein Fehler aufgetreten ist oder „Execute“ zurückgesetzt wird.

Done

Bit ist gesetzt, sobald die Parameterierung erfolgreich beendet wurde

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

Error

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

- Typ: BOOL
- Defaultwert: FALSE
- Art: OUTPUT

Das Bit wird beim Zurücksetzen von „Execute“ zurückgesetzt.

ErrorID

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

- Typ: UINT
- Defaultwert: 0
- Art: OUTPUT

Antriebsfehler („Drive errors“) werden bei einer Parametrierung nicht beachtet. Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

ErrorParameter

SDO-Nr. desjenigen Parameters, der im Fall eines Fehlers beim Parameterieren den Fehler verursacht hat

- Typ: INT
- Defaultwert: 0
- Art: OUTPUT

Der Wert wird beim Zurücksetzen von „Execute“ auf 0 gesetzt.

5 Beispielprojekt

Das Projekt „halstrup_walcher_FBs_V1_0_Example.apj“ beinhaltet ein Beispiel, das die Anwendung der Funktionsbausteine „h_w“ zeigt. Es werden zwei Antriebe eingeführt („Motor_1“ und „Motor_2“), mit dem Programm „ExamplePrg“ können diese Antriebe verfahren werden. Es kann auch ein Dauertest gestartet werden, wobei jeweils ein Motor startet, sobald der andere eine Positionierung abgeschlossen hat. In jedem zehnten SPS-Zyklus wird zudem

das Lesen von Parametern angestoßen. Damit kann z.B. zyklisch das Auslesen des maximal aufgetretenen Drehmoments bei der letzten Fahrt gestartet werden.

Die FBs „hwMC_Parametrization“ und „hwMC_PositionParametrization“ werden zyklisch aufgerufen, die Eingaben dazu können (ebenso wie auch weitere Vorgaben für das Lesen von Parametern) im Watch-Fenster von „ExamplePrg“ getätigt werden:

