

# VIPA SPEED7 Library

OPL\_SP7-LIB | SW90LS0MA V10.008 | Handbuch

HB00 | OPL\_SP7-LIB | SW90LS0MA V10.008 | de | 20-08

Baustein Bibliothek - Device Specific



YASKAWA Europe GmbH  
Ohmstraße 4  
91074 Herzogenaurach  
Tel.: +49 9132 744 0  
Fax: +49 9132 744 186  
E-Mail: [info@yaskawa.eu.com](mailto:info@yaskawa.eu.com)  
Internet: [www.yaskawa.eu.com](http://www.yaskawa.eu.com)

## Inhaltsverzeichnis

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Allgemeines</b> .....   | <b>4</b>  |
| 1.1      | Copyright © YASKAWA Europe GmbH.....                                       | 4         |
| 1.2      | Über dieses Handbuch.....  | 5         |
| <b>2</b> | <b>Wichtige Hinweise</b> .....   | <b>6</b>  |
| 2.1      | Allgemein.....   | 6         |
| 2.2      | Intern verwendete Bausteine.....   | 6         |
| 2.3      | Kein optimierter Bausteinzugriff.....                                      | 7         |
| 2.4      | Deklarationstypen.....   | 7         |
| <b>3</b> | <b>Bibliothek einbinden</b> .....  | <b>8</b>  |
| 3.1      | Einbinden in Siemens SIMATIC Manager.....                                  | 9         |
| 3.2      | Einbinden in Siemens TIA Portal.....                                       | 10        |
| <b>4</b> | <b>Bausteinparameter</b> .....   | <b>11</b> |
| 4.1      | HW-Kennung - HW_ID.....  | 11        |
| 4.2      | Allgemeine und spezifische Fehlercodes RET_VAL.....                        | 11        |
| <b>5</b> | <b>Modulspezifisch - "Device Specific"</b> .....                           | <b>14</b> |
| 5.1      | Frequenzmessung - " <i>Frequency Measurement</i> ".....                    | 14        |
| 5.1.1    | FC 300 ... 303 - Frequenzmessung SLIO konsistent.....                      | 14        |
| 5.1.2    | FC 300 - FM_SET_CONTROL - Control Frequenzmessung konsistent.....          | 14        |
| 5.1.3    | FC 301 - FM_GET_PERIOD - Periodendauer berechnen konsistent.....           | 16        |
| 5.1.4    | FC 302 - FM_GET_FREQUENCY - Frequenz berechnen konsistent.....             | 19        |
| 5.1.5    | FC 303 - FM_GET_SPEED - Drehzahl berechnen konsistent.....                 | 21        |
| 5.1.6    | FC 310 ... 313 - Frequenzmessung SLIO.....                                 | 24        |
| 5.1.7    | FC 310 - FM_CONTROL - Control Frequenzmessung.....                         | 24        |
| 5.1.8    | FC 311 - FM_CALC_PERIOD - Periodendauer berechnen.....                     | 26        |
| 5.1.9    | FC 312 - FM_CALC_FREQUENCY - Frequenz berechnen.....                       | 28        |
| 5.1.10   | FC 313 - FM_CALC_SPEED - Drehzahl berechnen.....                           | 30        |
| 5.2      | Energiemessung - " <i>Energy Measurement</i> ".....                        | 32        |
| 5.2.1    | Übersicht.....   | 32        |
| 5.2.2    | FB 325 - EM_COM_R1 - Kommunikation mit 031-1PAXx.....                      | 35        |
| 5.2.3    | UDT 325 - EM_DATA_R1 - Datenstruktur für FB 325.....                       | 36        |
| 5.3      | Motion-Module - " <i>Motion Modules</i> ".....                             | 39        |
| 5.3.1    | Übersicht.....   | 39        |
| 5.3.2    | FB 320 - ACYC_RW - Azyklischer Zugriff auf System SLIO Motion-Modul.....   | 41        |
| 5.3.3    | FB 321 - ACYC_DS - Azyklische Parametrierung System SLIO Motion-Modul..... | 44        |
| 5.3.4    | UDT 321 - ACYC_OBJECT-DATA - Datenstruktur für FB 321.....                 | 48        |
| 5.4      | RAM nach WLD - " <i>WLD</i> ".....   | 49        |
| 5.4.1    | FB 240 - RAM_to_s7prog.wld - RAM nach s7prog.wld.....                      | 49        |
| 5.4.2    | FB 241 - RAM_to_autoload.wld - RAM nach autoload.wld.....                  | 49        |
| 5.5      | System 100V interne E/As - " <i>Onboard I/O System 100V</i> ".....         | 50        |
| 5.5.1    | SFC 223 - PWM - Pulsweitenmodulation.....                                  | 50        |
| 5.5.2    | SFC 224 - HSC - High-speed-Counter.....                                    | 52        |
| 5.5.3    | SFC 225 - HF_PWM - HF Pulsweitenmodulation.....                            | 53        |

# 1 Allgemeines

## 1.1 Copyright © YASKAWA Europe GmbH

### All Rights Reserved

Dieses Dokument enthält geschützte Informationen von YASKAWA und darf außer in Übereinstimmung mit anwendbaren Vereinbarungen weder offengelegt noch benutzt werden.

Dieses Material ist durch Urheberrechtsgesetze geschützt. Ohne schriftliches Einverständnis von YASKAWA und dem Besitzer dieses Materials darf dieses Material weder reproduziert, verteilt, noch in keiner Form von keiner Einheit (sowohl YASKAWA-intern als auch -extern) geändert werden, es sei denn in Übereinstimmung mit anwendbaren Vereinbarungen, Verträgen oder Lizenzen.

Zur Genehmigung von Vervielfältigung oder Verteilung wenden Sie sich bitte an:  
YASKAWA Europe GmbH, European Headquarters, Hauptstraße 185, 65760 Eschborn, Germany

Tel.: +49 6196 569 300

Fax.: +49 6196 569 398

E-Mail: [info@yaskawa.eu.com](mailto:info@yaskawa.eu.com)

Internet: [www.yaskawa.eu.com](http://www.yaskawa.eu.com)



*Es wurden alle Anstrengungen unternommen, um sicherzustellen, dass die in diesem Dokument enthaltenen Informationen zum Zeitpunkt der Veröffentlichung vollständig und richtig sind. Das Recht auf Änderungen der Informationen bleibt jedoch vorbehalten.*

*Die vorliegende Kundendokumentation beschreibt alle heute bekannten Hardware-Einheiten und Funktionen. Es ist möglich, dass Einheiten beschrieben sind, die beim Kunden nicht vorhanden sind. Der genaue Lieferumfang ist im jeweiligen Kaufvertrag beschrieben.*

### EG-Konformitätserklärung

Hiermit erklärt YASKAWA Europe GmbH, dass die Produkte und Systeme mit den grundlegenden Anforderungen und den anderen relevanten Vorschriften übereinstimmen. Die Übereinstimmung ist durch CE-Zeichen gekennzeichnet.

### Informationen zur Konformitätserklärung

Für weitere Informationen zur CE-Kennzeichnung und Konformitätserklärung wenden Sie sich bitte an Ihre Landesvertretung der YASKAWA Europe GmbH.

### Warenzeichen

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S und Commander Compact sind eingetragene Warenzeichen der YASKAWA Europe GmbH.

SPEED7 ist ein eingetragenes Warenzeichen der YASKAWA Europe GmbH.

SIMATIC, STEP, SINEC, TIA Portal, S7-300, S7-400 und S7-1500 sind eingetragene Warenzeichen der Siemens AG.

Microsoft und Windows sind eingetragene Warenzeichen von Microsoft Inc., USA.

Portable Document Format (PDF) und Postscript sind eingetragene Warenzeichen von Adobe Systems, Inc.

Alle anderen erwähnten Firmennamen und Logos sowie Marken- oder Produktnamen sind Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Eigentümer.

**Dokument-Support**

Wenden Sie sich an Ihre Landesvertretung der YASKAWA Europe GmbH, wenn Sie Fehler anzeigen oder inhaltliche Fragen zu diesem Dokument stellen möchten. Ist eine solche Stelle nicht erreichbar, können Sie YASKAWA Europe GmbH über folgenden Kontakt erreichen:

YASKAWA Europe GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Fax: +49 9132 744 29 1204

E-Mail: Documentation.HER@yaskawa.eu.com

**Technischer Support**

Wenden Sie sich an Ihre Landesvertretung der YASKAWA Europe GmbH, wenn Sie Probleme mit dem Produkt haben oder Fragen zum Produkt stellen möchten. Ist eine solche Stelle nicht erreichbar, können Sie den YASKAWA Kundenservice über folgenden Kontakt erreichen:

YASKAWA Europe GmbH,  
European Headquarters, Hauptstraße 185, 65760 Eschborn, Germany

Tel.: +49 6196 569 500 (Hotline)

E-Mail: support@yaskawa.eu.com

## 1.2 Über dieses Handbuch

**Zielsetzung und Inhalt**

Das Handbuch beschreibt die VIPA Baustein-Bibliothek *"Device Specific"*:

- Beschrieben wird Aufbau, Projektierung und Anwendung in verschiedenen Programmersystemen.
- Das Handbuch ist geschrieben für Anwender mit Grundkenntnissen in der Automatisierungstechnik.
- Das Handbuch ist in elektronischer Form als PDF-Datei verfügbar. Hierzu ist der Adobe Acrobat Reader erforderlich.
- Das Handbuch ist in Kapitel gegliedert. Jedes Kapitel beschreibt eine abgeschlossene Thematik.
- Als Orientierungshilfe stehen im Handbuch zur Verfügung:
  - Gesamt-Inhaltsverzeichnis am Anfang des Handbuchs
  - Verweise mit Seitenangabe

**Piktogramme Signalwörter**

Besonders wichtige Textteile sind mit folgenden Piktogrammen und Signalworten ausgezeichnet:

**GEFAHR!**

Unmittelbar drohende oder mögliche Gefahr. Personenschäden sind möglich.

**VORSICHT!**

Bei Nichtbefolgen sind Sachschäden möglich.



*Zusätzliche Informationen und nützliche Tipps.*

## 2 Wichtige Hinweise

### 2.1 Allgemein



Nachfolgend finden Sie wichtige Hinweise, die grundsätzlich beim Einsatz der Bausteine zu beachten sind.

### 2.2 Intern verwendete Bausteine



#### VORSICHT!

Folgende Bausteine werden intern verwendet und dürfen nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB! Bitte verwenden Sie für den Aufruf immer die zugehörige Funktion.

| FC/SFC     | Bezeichnung | Beschreibung  |
|------------|-------------|---|
| FC/SFC 131 | TSEND_      | wird intern für FB 63 verwendet                     |
| FC/SFC 132 | TRECV_      | wird intern für FB 64 verwendet                     |
| FC/SFC 133 | TCON_       | wird intern für FB 65 verwendet                     |
| FC/SFC 134 | TDISCON_    | wird intern für FB 66 verwendet                     |
| FC/SFC 135 | TUSEND_     | wird intern für FB 67 verwendet                     |
| FC/SFC 136 | TURECV_     | wird intern für FB 68 verwendet                     |
| FC/SFC 192 | CP_S_R      | wird intern für FB 7 und FB 8 verwendet             |
| FC/SFC 196 | AG_CNTRL    | wird intern für FC 10 verwendet                     |
| FC/SFC 198 | USEND_      | wird intern für FB 8 verwendet                      |
| FC/SFC 199 | URCV_       | wird intern für FB 9 verwendet                      |
| FC/SFC 200 | AG_GET      | wird intern für FB/SFB 14 verwendet                 |
| FC/SFC 201 | AG_PUT      | wird intern für FB/SFB 15 verwendet                 |
| FC/SFC 202 | AG_BSEND    | wird intern für FB/SFB 12 verwendet                 |
| FC/SFC 203 | AG_BRCV     | wird intern für FB/SFB 13 verwendet                 |
| FC/SFC 204 | IP_CONF     | wird intern für FB 55 IP_CONF verwendet             |
| FC/SFC 205 | AG_SEND     | wird intern für FC 5 AG_SEND verwendet              |
| FC/SFC 206 | AG_RECV     | wird intern für FC 6 AG_RECV verwendet              |
| FC/SFC 253 | IBS_ACCESS  | wird intern für SPEED-Bus-INTERBUS-Master verwendet |
| SFB 238    | EC_RWOD     | wird intern für EtherCAT-Kommunikation verwendet    |
| SFB 239    | FUNC        | wird intern für FB 240, FB 241 verwendet            |

## 2.3 Kein optimierter Bausteinzugriff



Bitte beachten Sie, dass die Bausteine für den Einsatz in S7-1500 CPUs von Siemens keinen optimierten Bausteinzugriff unterstützen! Bei verwendeten Instanz- und Datenbausteinen ist der optimierte Bausteinzugriff zu deaktivieren!

### Bausteinzugriff einstellen

1. Öffnen Sie im Siemens TIA Portal in der *Projektnavigation* die "Programmbausteine".
2. Wählen Sie den Baustein an, für welchen Sie den Bausteinzugriff ändern möchten und wählen Sie "Kontextmenü → Eigenschaften".  
⇒ Der "Eigenschaften"-Dialog des Bausteins wird geöffnet.
3. Wählen Sie "Attribute" an.
4. Deaktivieren Sie den Parameter "Optimierter Bausteinzugriff".
5. Bestätigen Sie Ihre Eingabe mit [OK].

Näheres hierzu finden Sie im Handbuch zum Siemens TIA Portal.

## 2.4 Deklarationstypen

Bitte beachten Sie, dass die Schreibweisen der Deklarationstypen in Siemens STEP7 und TIA Portal sich unterscheiden. In dieser Dokumentation wird die Schreibweise für Siemens STEP7 verwendet. Eine Gegenüberstellung der Schreibweisen können Sie der nachfolgenden Tabelle entnehmen.

| Siemens TIA Portal | Siemens STEP7 |
|--------------------|---------------|
| Input              | IN            |
| Output             | OUT           |
| InOut              | IN_OUT        |
| Static             | STAT          |
| Temp               | TEMP          |

### 3 Bibliothek einbinden

#### Baustein-Bibliothek "Device Specific"

Die Baustein-Bibliothek finden Sie im "Service/Support"-Bereich auf [www.vipa.com](http://www.vipa.com) unter "Downloads → VIPA Lib" als "Baustein-Bibliothek Device Specific - SW90LS0MA" zum Download. Die Bibliothek liegt als gepackte zip-Dateien vor. Sobald Sie die Bausteine verwenden möchten, müssen Sie diese in Ihr Projekt importieren.



*Bitte verwenden Sie immer das zu Ihrer Bibliothek zugehörige Handbuch. Solange es keine beschreibungsrelevante Änderungen gibt, können im Handbuch die Versionsangaben der Bibliothek und der zugehörigen Dateien von denen der Bibliothek abweichen.*

#### Folgende Bausteinbibliotheken stehen zur Verfügung




| Datei                             | Beschreibung  |
|-----------------------------------|---|
| DeviceSpecific_S7_V0006.zip       | <ul style="list-style-type: none"> <li>■ Bausteinbibliothek für Siemens SIMATIC Manager.</li> <li>■ Für den Einsatz in VIPA-CPU's bzw. S7-300 CPU's von Siemens.</li> </ul> |
| DeviceSpecific_TIA_V0008.zip      | <ul style="list-style-type: none"> <li>■ Bausteinbibliothek für Siemens TIA Portal V14.</li> <li>■ Für den Einsatz in VIPA-CPU's bzw. S7-300 CPU's von Siemens.</li> </ul>  |
| DeviceSpecific_TIA_1500_V0004.zip | <ul style="list-style-type: none"> <li>■ Bausteinbibliothek für Siemens TIA Portal V14.</li> <li>■ Für den Einsatz in S7-1500 CPU's von Siemens.</li> </ul>                 |
| DS_Demo_S7_V001.zip               | <ul style="list-style-type: none"> <li>■ Demoprojekt für Siemens SIMATIC Manager</li> <li>■ Für den Einsatz in VIPA-CPU's bzw. S7-300 CPU's von Siemens.</li> </ul>         |
| DS_Demo_TIA_V001.zip              | <ul style="list-style-type: none"> <li>■ Demoprojekt für Siemens TIA Portal V14.</li> <li>■ Für den Einsatz in VIPA-CPU's bzw. S7-300 CPU's von Siemens.</li> </ul>         |
| DS_Demo_TIA_1500_V001.zip         | <ul style="list-style-type: none"> <li>■ Demoprojekt für Siemens TIA Portal V14.</li> <li>■ Für den Einsatz in S7-1500 CPU's von Siemens.</li> </ul>                        |




### 3.1 Einbinden in Siemens SIMATIC Manager

#### Übersicht






Die Einbindung in den Siemens SIMATIC Manager erfolgt nach folgenden Schritten:

1.  ZIP-Datei laden
2.  Bibliothek "dearchivieren"
3.  Bibliothek öffnen und Bausteine in Projekt übertragen



#### ZIP-Datei laden

-  Navigieren Sie auf der Webseite zu der gewünschten ZIP-Datei, laden und speichern Sie diese in Ihrem Arbeitsverzeichnis.

#### Bibliothek dearchivieren

1.  Starten Sie den Siemens SIMATIC Manager mit Ihrem Projekt.
2.  Öffnen Sie mit "Datei → Dearchivieren" das Dialogfenster zur Auswahl der ZIP-Datei.
3.  Wählen Sie die entsprechende ZIP-Datei an und klicken Sie auf [Öffnen].
4.  Geben Sie ein Zielverzeichnis an, in dem die Bausteine abzulegen sind.
5.  Starten Sie den Entpackvorgang mit [OK].

#### Bibliothek öffnen und Bausteine in Projekt übertragen

1.  Öffnen Sie die Bibliothek nach dem Entpackvorgang.
2.  Öffnen Sie Ihr Projekt und kopieren Sie die erforderlichen Bausteine aus der Bibliothek in das Verzeichnis "Bausteine" Ihres Projekts.  
⇒ Nun haben Sie in Ihrem Anwenderprogramm Zugriff auf die Bausteine.



*Werden anstelle der SFCs FCs verwendet, so werden diese von den System 300S VIPA-CPU's ab Firmware 3.6.0 unterstützt.*

## 3.2 Einbinden in Siemens TIA Portal

### Übersicht

Die Einbindung in das Siemens TIA Portal erfolgt nach folgenden Schritten:

1. ZIP-Datei laden
2. ZIP-Datei entpacken
3. Bibliothek "dearchivieren"
4. Bibliothek öffnen und Bausteine in Projekt übertragen

### ZIP-Datei laden

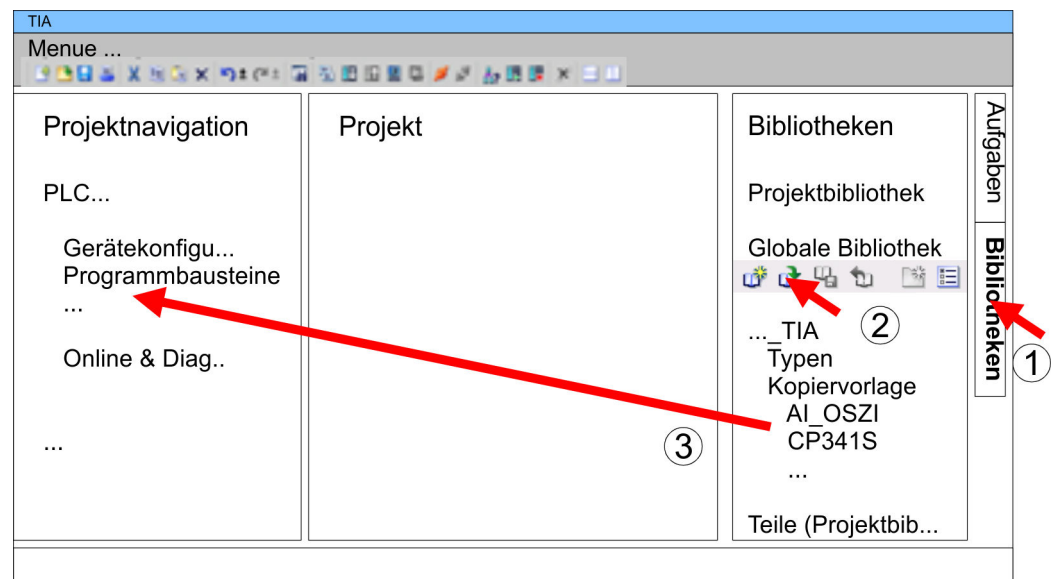
1. Navigieren Sie auf der Webseite zu der ZIP-Datei, welche zu Ihrer Programmversion passt.
2. Laden und speichern Sie diese in Ihrem Arbeitsverzeichnis.

### ZIP-Datei entpacken

- Entpacken Sie die ZIP-Datei mit Ihrem Entpackprogramm in ein Arbeitsverzeichnis für das Siemens TIA Portal.

### Bibliothek öffnen und Bausteine in Projekt übertragen

1. Starten Sie das Siemens TIA Portal mit Ihrem Projekt.
2. Wechseln sie in die *Projektansicht*.
3. Wählen Sie auf der rechten Seite die Task-Card "Bibliotheken".
4. Klicken Sie auf "Globale Bibliothek".
5. Klicken Sie auf "Globale Bibliothek öffnen".
6. Navigieren Sie zu ihrem Arbeitsverzeichnis und laden Sie die Datei ...\_TIA.al1x.



7. Kopieren Sie die erforderlichen Bausteine aus der Bibliothek in das Verzeichnis "Programmbausteine" in der *Projektnavigation* Ihres Projekts. Nun haben Sie in Ihrem Anwenderprogramm Zugriff auf die Bausteine.

## 4 Bausteinparameter

### 4.1 HW-Kennung - HW\_ID

#### HW-Kennung

- Den Parameter *HW\_ID* zur Angabe der *HW-Kennung* gibt es nur bei Einsatz in S7-1500 CPUs von Siemens.
- Bei der Konfiguration einer Hardware-Komponente wird für jedes Objekt der Hardware-Konfiguration automatisch eine Hardware-Kennung als *HW-Kennung* vergeben.
- Die *HW-Kennung* umfasst Module, Ports, Schnittstellen und E/A-Bereiche von Bus-Systemen.
- Die *HW-Kennung* ist eine dezimale Ganzzahl-Konstante vom Datentyp HW\_IO.
- Bei der *HW-Kennung* wird nicht zwischen Ein- und Ausgabebereich unterschieden.
- Mit Hilfe der *HW-Kennung* können Sie die entsprechenden Hardware-Komponenten adressieren.

#### HW-Kennung ermitteln

Die *HW-Kennung* für die jeweilige Komponente können Sie mit folgender Vorgehensweise ermitteln:

1. ➤ Öffnen Sie in der *Projektnavigation* die "*Gerätekonfiguration*".
2. ➤ Selektieren Sie die gewünschte Hardware-Komponente, deren *HW-Kennung* Sie ermitteln möchten.
3. ➤ Klicken Sie im *Inspektor*-Fenster auf "*Allgemein*".
  - ⇒ Die "*HW-Kennung*" wird angezeigt. Diese können Sie bei der Beschaltung der Bausteine in den Parameter *HW\_ID* übernehmen.

#### HW-Kennung und Systemkonstanten

Sie können auch die *HW-Kennung* über die "*Systemkonstanten*" ermitteln. Über "*Systemkonstanten*" im *Inspektor*-Fenster werden alle HW-Kennungen eines in der Gerätesicht markierten Objekts mit *Name* und *Typ* aufgelistet. *Name* und *Typ* werden bei der Zuordnung der HW-Kennung automatisch generiert. Hierbei besitzt *Name* einen hierarchischen Aufbau mit maximal 4 Hierarchieebenen, wobei jede Ebene durch ein "~" getrennt wird. Den Namen der Komponente der entsprechenden Hierarchieebene können Sie jederzeit über die Eigenschaften anpassen.

#### HW-Kennung im Anwenderprogramm

- Bei der Erstellung Ihres Anwenderprogramms können Sie über Doppelklick auf den entsprechenden Ein- bzw. Ausgabe-Parameter aus einer Liste aller möglichen Hardware-Komponenten die entsprechende Hardware-Komponente zuordnen.
- Bei einem Prozessalarm können sie über die Startinformationen die *HW-Kennung* als "*ID*" der alarmlösenden Hardware-Komponente ermitteln.

### 4.2 Allgemeine und spezifische Fehlercodes RET\_VAL

#### Übersicht

Der Rückgabewert *RET\_VAL* einer Systemfunktion stellt einen der beiden folgenden Fehlercodes zur Verfügung:

- *Allgemeiner Fehlercode*, der sich auf jeden beliebigen SFC beziehen kann.
- *Spezifischer Fehlercode*, der sich auf den jeweiligen SFC bezieht.

Es handelt sich beim Datentyp des Ausgangsparameters *RET\_VAL* zwar um eine Ganzzahl (INT), doch die Fehlercodes der Systemfunktionen werden nach hexadezimalen Werten gegliedert.

Wenn Sie einen Rückgabewert auswerten und den Wert mit den Fehlercodes vergleichen, so lassen Sie sich den Fehlercode im Hexadezimalformat ausgeben.

Allgemeine und spezifische Fehlercodes RET\_VAL

**RET\_VAL (Rückgabewert)** Die folgende Tabelle zeigt den Aufbau eines Fehlercodes:

| Bit      | Bedeutung  |
|----------|--|
| 7 ... 0  | Ereignisnummer bzw. Fehlerklasse und Einzelfehler  |
| 14 ... 8 | Bit 14 ... 8 = "0": <b>Spezifischer Fehlercode</b><br>Den spezifischen Fehlercode finden Sie in der Beschreibung der einzelnen SFCs.<br>Bit 14 ... 8 > "0": <b>Allgemeiner Fehlercode</b><br>Die möglichen allgemeinen Fehlercodes finden Sie auf der folgenden Seite. |
| 15       | Bit 15 = "1": zeigt an, dass ein Fehler aufgetreten ist.   |

### Spezifischer Fehlercode

Dieser Fehlercode zeigt an, dass ein Fehler, der zu einer bestimmten Systemfunktion gehört, während der Bearbeitung aufgetreten ist.

Ein spezifischer Fehlercode besteht aus:

- Fehlerklasse zwischen 0 und 7
- Einzelfehler zwischen 0 und 15

| Bit      | Bedeutung  |
|----------|--|
| 3 ... 0  | Einzelfehler   |
| 6 ... 4  | Fehlerklasse   |
| 7        | Bit 7 = "1"  |
| 14 ... 8 | Bit 14 ... 8 = "0"                                       |
| 15       | Bit 15 = "1": zeigt an, dass ein Fehler aufgetreten ist. |

### Allgemeine Fehlercodes RET\_VAL

Der Parameter *RET\_VAL* verschiedener SFCs liefert keine spezifischen, sondern nur allgemeine Fehlerinformationen zurück.

Der allgemeine Fehlercode enthält Fehlerinformationen, die bei allen Systemfunktionen auftreten können. Ein allgemeiner Fehlercode besteht aus den beiden folgenden Nummern:

- Eine Parameternummer zwischen 1 und 111, wobei 1 den ersten Parameter, 2 den zweiten Parameter usw. des aufgerufenen SFC anzeigt.
- Eine Ereignisnummer zwischen 0 und 127. Die Ereignisnummer zeigt einen synchronen Fehler an.

| Bit      | Bedeutung  |
|----------|--|
| 7 ... 0  | Ereignisnummer   |
| 14 ... 8 | Parameternummer  |
| 15       | Bit 15 = "1": zeigt an, dass ein Fehler aufgetreten ist. |

**Allgemeine Fehlercodes**

In der folgenden Tabelle werden die allgemeinen Fehlercodes eines Rückgabewerts erläutert. Die Darstellung erfolgt im Hexadezimalformat, wobei der Buchstabe x in jeder Codenummer nur als Platzhalter dient und die Nummer des Parameters der Systemfunktion darstellt, die den Fehler verursacht hat.

| Fehlercode | Beschreibung   |
|------------|--|
| 8x7Fh      | Interner Fehler. Dieser Fehlercode zeigt einen internen Fehler am Parameter x an. Dieser Fehler wurde nicht vom Anwender verursacht und kann von ihm auch nicht behoben werden.  |
| 8x01h      | Unzulässige Syntaxkennung bei einem ANY-Parameter.   |
| 8x22h      | Bereichslängenfehler beim Lesen eines Parameters.  |
| 8x23h      | Bereichslängenfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem ANY-Parameter nicht durch 8 teilbar ist.                                  |
| 8x24h      | Bereichsfehler beim Lesen eines Parameters.  |
| 8x25h      | Bereichsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x in einem Bereich befindet, der für die Systemfunktion unzulässig ist. Die Beschreibung der jeweiligen Funktion gibt die Bereiche an, die für die Funktion unzulässig sind.                   |
| 8x26h      | Der Parameter enthält eine zu große Nummer einer Zeitzelle. Dieser Fehlercode zeigt an, dass die Zeitzelle, die in Parameter x angegeben wird, nicht vorhanden ist.  |
| 8x27h      | Der Parameter enthält eine zu große Nummer einer Zählerzelle (Nummernfehler des Zählers). Dieser Fehlercode zeigt an, dass die Zählerzelle, die in Parameter x angegeben wird, nicht vorhanden ist.  |
| 8x28h      | Ausrichtungsfehler beim Lesen eines Parameters.  |
| 8x29h      | Ausrichtungsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass der Verweis auf den Parameter x ein Operand ist, dessen Bitadresse ungleich 0 ist.  |
| 8x30h      | Der Parameter befindet sich in dem schreibgeschützten Global-DB.   |
| 8x31h      | Der Parameter befindet sich in dem schreibgeschützten Instanz-DB. Dieser Fehlercode zeigt an, dass der Parameter x sich in einem schreibgeschützten Datenbaustein befindet. Wenn der Datenbaustein von der Systemfunktion selbst geöffnet wurde, gibt die Systemfunktion immer den Wert 8x30h aus. |
| 8x32h      | Der Parameter enthält eine zu große DB-Nummer (Nummernfehler des DBs).   |
| 8x34h      | Der Parameter enthält eine zu große FC-Nummer (Nummernfehler des FCs).   |
| 8x35h      | Der Parameter enthält eine zu große FB-Nummer (Nummernfehler des FBs). Dieser Fehlercode zeigt an, dass der Parameter x eine Bausteinnummer enthält, die größer ist als die maximal zulässige Bausteinnummer.  |
| 8x3Ah      | Der Parameter enthält die Nummer eines DBs, der nicht geladen ist.   |
| 8x3Ch      | Der Parameter enthält die Nummer eines FCs, der nicht geladen ist.   |
| 8x3Eh      | Der Parameter enthält die Nummer eines FBs, der nicht geladen ist.   |
| 8x42h      | Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.  |
| 8x43h      | Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.  |
| 8x44h      | Fehler beim n-ten ( $n > 1$ ) Lesezugriff nach Auftreten eines Fehlers.  |
| 8x45h      | Fehler beim n-ten ( $n > 1$ ) Schreibzugriff nach Auftreten eines Fehlers. Dieser Fehlercode zeigt an, dass der Zugriff auf den gewünschten Parameter verweigert wird.   |

## 5 Modulspezifisch - "Device Specific"

### 5.1 Frequenzmessung - "Frequency Measurement"

#### 5.1.1 FC 300 ... 303 - Frequenzmessung SLIO konsistent

##### Übersicht

Mit folgenden VIPA-spezifischen Funktionen können Sie System SLIO Frequenzmess-Module ansteuern, welche über PROFIBUS, PROFINET oder EtherCAT angebunden sind. Der Einsatz unter EtherCAT ist ausschließlich mit einer VIPA EtherCAT-CPU möglich. Von diesen Funktionen wird intern der SFC 14 - DPRD\_DAT bzw. SFC 15 - DPWR\_DAT für konsistentes Lesen bzw. Schreiben von Nutzdaten aufgerufen. Fehlermeldungen dieser Bausteine werden über den Parameter *ERROR* zurückgeliefert.

| Baustein | Symbol           | Kommentar   |
|----------|------------------|---|
| FC 300   | FM_SET_CONTROL   | Funktion zur Steuerung der Frequenzmessung mit integriertem konsistentem Zugriff. |
| FC 301   | FM_GET_PERIOD    | Funktion zur Berechnung der Periodendauer mit integriertem konsistentem Zugriff.  |
| FC 302   | FM_GET_FREQUENCY | Funktion zur Berechnung der Frequenz mit integriertem konsistentem Zugriff.       |
| FC 303   | FM_GET_SPEED     | Funktion zur Berechnung der Drehzahl mit integriertem konsistentem Zugriff.       |

#### 5.1.2 FC 300 - FM\_SET\_CONTROL - Control Frequenzmessung konsistent

##### Beschreibung

Mit dem FC 300 FM\_SET\_CONTROL können Sie das System SLIO Frequenzmess-Modul steuern. Von dieser Funktion wird intern der SFC 15 - DPWR\_DAT für konsistentes Schreiben von Nutzdaten aufgerufen. Hierbei werden Fehlermeldungen des Bausteins über *ERROR* ausgegeben.

##### Parameter

| Parameter         | Deklaration | Datentyp     | Speicherbereich | Beschreibung  |
|-------------------|-------------|--------------|-----------------|---|
| ENABLE_FM         | INPUT       | BOOL         | E, A, M, D, L   | Freigabe Frequenzmessung  |
| LADDR_OUT / HW_ID | INPUT       | WORD / HW_IO | E, A, M, D, L   | <ul style="list-style-type: none"> <li>■ LADDR_OUT           <ul style="list-style-type: none"> <li>– Logische Basis-Ausgabeadresse des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens.</li> </ul> </li> <li>■ HW_ID           <ul style="list-style-type: none"> <li>– <i>HW-Kennung</i> zur Adressierung des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in S7-1500 CPU's von Siemens.</li> </ul> </li> </ul> |
| PRESET_CH0        | INPUT       | DINT         | E, A, M, D, L   | Kanal 0: Messperiode  |
| PRESET_CH1        | INPUT       | DINT         | E, A, M, D, L   | Kanal 1: Messperiode  |
| DONE              | OUTPUT      | BOOL         | E, A, M, D, L   | Fertigmeldung (TRUE = OK)   |
| ERROR             | OUTPUT      | WORD         | E, A, M, D, L   | Rückgabewert (0 = OK)   |

**ENABLE\_FM**

Durch Setzen von *ENABLE\_FM* werden die über *PRESET\_CH0/1* vorgegebenen *Messperioden* an die Kanäle übergeben und bei beiden Kanälen die Messung gestartet. Durch Rücksetzen von *ENABLE\_FM* werden beide Kanäle gestoppt.



*Nur solange ENABLE\_FM gesetzt ist, können ermittelte Werte vom Modul abgerufen werden. Ansonsten erhalten Sie die Fehlermeldung, dass die Kanäle deaktiviert sind.*

**LADDR\_OUT**

Peripherieadresse:

- Dieser Parameter ist bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens verfügbar.
- Projektierte Anfangsadresse aus dem Ausgabebereich des System SLIO Frequenz-Messmoduls, in welcher geschrieben werden soll. Die Adresse wird hexadezimal angegeben.
- (Beispiel: Adresse 100: *LADDR\_OUT*: = W#16#64)

**HW\_ID**

HW-Kennung:

- Dieser Parameter ist nur bei Einsatz in S7-1500 CPU's von Siemens verfügbar.
- Geben Sie unter *HW\_ID* die *HW-Kennung* an, mit deren Hilfe Ihr Modul entsprechend adressiert werden kann. ↪ *Kap. 4.1 "HW-Kennung - HW\_ID" Seite 11*

**PRESET\_CHx**

Geben Sie hier die Messperiode in  $\mu\text{s}$  für den entsprechenden Kanal an.

Wertebereich:  $1\mu\text{s}$  ... 8 388 607 $\mu\text{s}$

**DONE**

Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**ERROR (Rückgabewert)**

Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D2 | Kanal 0:<br>Eingangswert Messperiode $\leq 0$                   |
| 0x80D3 | Kanal 1:<br>Eingangswert Messperiode $\leq 0$                   |
| 0x80D4 | Kanal 0:<br>Eingangswert Messperiode $> 8\,388\,607\mu\text{s}$ |
| 0x80D5 | Kanal 1:<br>Eingangswert Messperiode $> 8\,388\,607\mu\text{s}$ |

**Fehler des intern aufgerufenen SFC 15**

| Code   | Beschreibung  |
|--------|---|
| 0x808x | Systemfehler am Bus-Koppler   |
| 0x8090 | LADDR_OUT ist falsch, mögliche Gründe: <ul style="list-style-type: none"> <li>■ auf dieser Adresse ist kein Modul projektiert</li> <li>■ Einschränkung über die Länge der konsistenten Daten wurde nicht beachtet</li> <li>■ Anfangsadresse im Parameter LADDR_OUT wurde nicht hexadezimal angegeben</li> </ul> |
| 0x8093 | Für LADDR_OUT existiert kein Bus-Koppler, von dem Sie konsistente Daten lesen können.   |
| 0x80A0 | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.   |
| 0x80B0 | Systemfehler am Bus-Koppler   |
| 0x80B1 | Angegebene Länge des Quellbereichs entspricht nicht der projektierten Nutzdatenlänge.   |
| 0x80B2 | Systemfehler am Bus-Koppler   |
| 0x80B3 | Systemfehler am Bus-Koppler   |
| 0x80C1 | Die Daten des auf der Baugruppe vorangegangenen Leseauftrags sind von der Baugruppe noch nicht bearbeitet.  |
| 0x80C2 | Systemfehler am Bus-Koppler   |
| 0x80Fx | Systemfehler am Bus-Koppler   |
| 0x85xy | Systemfehler am Bus-Koppler   |
| 0x8xyy | Allgemeine Fehlerinformation<br>↪ Kap. 4.2 "Allgemeine und spezifische Fehlercodes RET_VAL" Seite 11  |

**5.1.3 FC 301 - FM\_GET\_PERIOD - Periodendauer berechnen konsistent**

**Beschreibung**

Mit dem FC 301 FM\_GET\_PERIOD können Sie die Periodendauer der Eingangssignale beider Kanäle des System SLIO Frequenzmess-Moduls berechnen. Von dieser Funktion wird intern der SFC 14 - DPRD\_DAT für konsistentes Lesen von Nutzdaten aufgerufen. Hierbei werden Fehlermeldungen des Bausteins über ERROR ausgegeben.

**Parameter**

| Parameter        | Deklaration | Datentyp     | Speicherbereich | Beschreibung  |
|------------------|-------------|--------------|-----------------|---|
| LADDR_IN / HW_ID | INPUT       | WORD / HW_IO | E, A, M, D, L   | <ul style="list-style-type: none"> <li>■ LADDR_IN                             <ul style="list-style-type: none"> <li>– Logische Basis-Eingabeadresse des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens.</li> </ul> </li> <li>■ HW_ID                             <ul style="list-style-type: none"> <li>– HW-Kennung zur Adressierung des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in S7-1500 CPU's von Siemens.</li> </ul> </li> </ul> |
| DONE             | OUTPUT      | BOOL         | E, A, M, D, L   | Fertigmeldung (TRUE = OK)   |



| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung             |
|------------|-------------|----------|-----------------|--------------------------|
| ERROR      | OUTPUT      | WORD     | E, A, M, D, L   | Rückgabewert<br>(0 = OK) |
| PERIOD_CH0 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 0: Periodendauer   |
| PERIOD_CH1 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 1: Periodendauer   |

**LADDR\_IN**

Peripherieadresse:

- Dieser Parameter ist bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens verfügbar.
- Projektierte Anfangsadresse aus dem Eingabebereich des System SLIO Frequenzmess-Moduls, aus welchem gelesen werden soll. Die Adresse wird hexadezimal angegeben.
- (Beispiel: Adresse 100: *LADDR\_IN*: = W#16#64)

**HW\_ID**

HW-Kennung:

- Dieser Parameter ist nur bei Einsatz in S7-1500 CPU's von Siemens verfügbar.
- Geben Sie unter *HW\_ID* die *HW-Kennung* an, mit deren Hilfe Ihr Modul entsprechend adressiert werden kann. ↪ *Kap. 4.1 "HW-Kennung - HW\_ID" Seite 11*

**DONE**

Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**PERIOD\_CHx**

Aktuell ermittelte Periodendauer des entsprechenden Kanals in 100ns.

**ERROR (Rückgabewert)**

Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D0 | Kanal 0 nicht im Status aktiv   |
| 0x80D1 | Kanal 1 nicht im Status aktiv   |
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                                      |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                                      |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80E0 | Kanal 0: Anzahl ermittelter Flanken = 0                               |
| 0x80E1 | Kanal 1: Anzahl ermittelter Flanken = 0                               |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                               |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                               |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der vorgegebenen Messperiode |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der vorgegebenen Messperiode |

**Fehler des intern aufgerufenen SFC 14**

| Code   | Beschreibung  |
|--------|---|
| 0x808x | Systemfehler am Bus-Koppler   |
| 0x8090 | <i>LADDR_IN</i> ist falsch, mögliche Gründe: <ul style="list-style-type: none"> <li>■ auf dieser Adresse ist kein Modul projektiert</li> <li>■ Einschränkung über die Länge der konsistenten Daten wurde nicht beachtet</li> <li>■ Anfangsadresse im Parameter <i>LADDR_IN</i> wurde nicht hexadezimal angegeben</li> </ul> |
| 0x8093 | Für <i>LADDR_IN</i> existiert kein Bus-Koppler, von dem Sie konsistente Daten lesen können.   |
| 0x80A0 | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.   |
| 0x80B0 | Systemfehler am Bus-Koppler   |
| 0x80B1 | Angegebene Länge des Quellbereichs entspricht nicht der projektierten Nutzdatenlänge.   |
| 0x80B2 | Systemfehler am Bus-Koppler   |
| 0x80B3 | Systemfehler am Bus-Koppler   |
| 0x80C1 | Die Daten des auf der Baugruppe vorangegangenen Leseauftrags sind von der Baugruppe noch nicht bearbeitet.  |
| 0x80C2 | Systemfehler am Bus-Koppler   |
| 0x80Fx | Systemfehler am Bus-Koppler   |

| Code   | Beschreibung   |
|--------|--|
| 0x85xy | Systemfehler am Bus-Koppler  |
| 0x8xyy | Allgemeine Fehlerinformation<br>↳ Kap. 4.2 "Allgemeine und spezifische Fehlercodes RET_VAL" Seite 11 |

### 5.1.4 FC 302 - FM\_GET\_FREQUENCY - Frequenz berechnen konsistent

#### Beschreibung

Mit dem FC 302 FM\_GET\_FREQUENCY können Sie die Frequenz der Eingangssignale beider Kanäle des System SLIO Frequenzmess-Moduls berechnen. Von dieser Funktion wird intern der SFC 14 - DPRD\_DAT für konsistentes Lesen von Nutzdaten aufgerufen. Hierbei werden Fehlermeldungen des Bausteins über *ERROR* ausgegeben.

#### Parameter

| Parameter           | Deklaration | Datentyp        | Speicherbereich | Beschreibung   |
|---------------------|-------------|-----------------|-----------------|--|
| LADDR_IN /<br>HW_ID | INPUT       | WORD /<br>HW_IO | E, A, M, D, L   | <ul style="list-style-type: none"> <li>■ LADDR_IN <ul style="list-style-type: none"> <li>– Logische Basis-Eingabeadresse des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens.</li> </ul> </li> <li>■ HW_ID <ul style="list-style-type: none"> <li>– <i>HW-Kennung</i> zur Adressierung des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in S7-1500 CPU's von Siemens.</li> </ul> </li> </ul> |
| DONE                | OUTPUT      | BOOL            | E, A, M, D, L   | Fertigmeldung<br>(TRUE = OK)   |
| ERROR               | OUTPUT      | WORD            | E, A, M, D, L   | Rückgabewert<br>(0 = OK)   |
| FREQUENCY_CH0       | OUTPUT      | DINT            | E, A, M, D, L   | Kanal 0: Frequenz  |
| FREQUENCY_CH1       | OUTPUT      | DINT            | E, A, M, D, L   | Kanal 1: Frequenz  |

#### LADDR\_IN

Peripherieadresse:

- Dieser Parameter ist bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens verfügbar.
- Projektierte Anfangsadresse aus dem Eingabebereich des System SLIO Frequenzmess-Moduls, aus welchem gelesen werden soll. Die Adresse wird hexadezimal angegeben.
- (Beispiel: Adresse 100: *LADDR\_IN*: = W#16#64)

#### HW\_ID

HW-Kennung:

- Dieser Parameter ist nur bei Einsatz in S7-1500 CPU's von Siemens verfügbar.
- Geben Sie unter *HW\_ID* die *HW-Kennung* an, mit deren Hilfe Ihr Modul entsprechend adressiert werden kann. ↳ Kap. 4.1 "HW-Kennung - HW\_ID" Seite 11

**DONE** Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**FREQUENCY\_CHx** Aktuell ermittelte Frequenz des entsprechenden Kanals in mHz.

**ERROR (Rückgabewert)** Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D0 | Kanal 0 nicht im Status aktiv                             |
| 0x80D1 | Kanal 1 nicht im Status aktiv                             |
| 0x80DA | Kanal 0: Gemessener Zeitwert = 0                          |
| 0x80DB | Kanal 1: Gemessener Zeitwert = 0                          |
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                          |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                          |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                 |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                 |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                   |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                   |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFFFF          |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFFFF          |
| 0x80E6 | Kanal 0: Frequenz > 600kHz                                |
| 0x80E7 | Kanal 1: Frequenz > 600kHz                                |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der Messperiode. |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der Messperiode. |

**Fehler des intern aufgerufenen SFC 14**

| Code   | Beschreibung  |
|--------|---|
| 0x808x | Systemfehler am Bus-Koppler   |
| 0x8090 | <i>LADDR_IN</i> ist falsch, mögliche Gründe: <ul style="list-style-type: none"> <li>■ auf dieser Adresse ist kein Modul projektiert</li> <li>■ Einschränkung über die Länge der konsistenten Daten wurde nicht beachtet</li> <li>■ Anfangsadresse im Parameter <i>LADDR_IN</i> wurde nicht hexadezimal angegeben</li> </ul> |
| 0x8093 | Für <i>LADDR_IN</i> existiert kein Bus-Koppler, von dem Sie konsistente Daten lesen können.   |
| 0x80A0 | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.   |
| 0x80B0 | Systemfehler am Bus-Koppler   |

| Code   | Beschreibung   |
|--------|--|
| 0x80B1 | Angegebene Länge des Quellbereichs entspricht nicht der projektierten Nutzdatenlänge.                      |
| 0x80B2 | Systemfehler am Bus-Koppler  |
| 0x80B3 | Systemfehler am Bus-Koppler  |
| 0x80C1 | Die Daten des auf der Baugruppe vorangegangenen Leseauftrags sind von der Baugruppe noch nicht bearbeitet. |
| 0x80C2 | Systemfehler am Bus-Koppler  |
| 0x80Fx | Systemfehler am Bus-Koppler  |
| 0x85xy | Systemfehler am Bus-Koppler  |
| 0x8xyy | Allgemeine Fehlerinformation<br>↳ Kap. 4.2 "Allgemeine und spezifische Fehlercodes RET_VAL" Seite 11       |

### 5.1.5 FC 303 - FM\_GET\_SPEED - Drehzahl berechnen konsistent

#### Beschreibung

Mit dem FC 303 FM\_GET\_SPEED können Sie die Drehzahl der Eingangssignale beider Kanäle des System SLIO Frequenzmess-Moduls berechnen. Von dieser Funktion wird intern der SFC 14 - DPRD\_DAT für konsistentes Lesen von Nutzdaten aufgerufen. Hierbei werden Fehlermeldungen des Bausteins über ERROR ausgegeben.

#### Parameter

| Parameter        | Deklaration | Datentyp     | Speicherbereich | Beschreibung  |
|------------------|-------------|--------------|-----------------|---|
| LADDR_IN / HW_ID | INPUT       | WORD / HW_IO | E, A, M, D, L   | <ul style="list-style-type: none"> <li>■ LADDR_IN <ul style="list-style-type: none"> <li>– Logische Basis-Eingabeadresse des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens.</li> </ul> </li> <li>■ HW_ID <ul style="list-style-type: none"> <li>– HW-Kennung zur Adressierung des Frequenzmess-Moduls.</li> <li>– Bei Einsatz in S7-1500 CPU's von Siemens.</li> </ul> </li> </ul> |
| RESOLUTION_CH0   | INPUT       | DINT         | E, A, M, D, L   | Kanal 0:<br>Auflösung des Gebers  |
| RESOLUTION_CH1   | INPUT       | DINT         | E, A, M, D, L   | Kanal 1:<br>Auflösung des Gebers  |
| DONE             | OUTPUT      | BOOL         | E, A, M, D, L   | Fertigmeldung<br>(TRUE = OK)  |
| ERROR            | OUTPUT      | WORD         | E, A, M, D, L   | Rückgabewert<br>(0 = OK)  |

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung         |
|-----------|-------------|----------|-----------------|----------------------|
| SPEED_CH0 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 0:<br>Drehzahl |
| SPEED_CH1 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 1:<br>Drehzahl |

**LADDR\_IN**

Peripherieadresse:

- Dieser Parameter ist bei Einsatz in VIPA-CPU's bzw. in S7-300 CPU's von Siemens verfügbar.
- Projektierte Anfangsadresse aus dem Eingabebereich des System SLIO Frequenzmess-Moduls, aus welchem gelesen werden soll. Die Adresse wird hexadezimal angegeben.
- (Beispiel: Adresse 100: *LADDR\_IN*: = W#16#64)

**HW\_ID**

HW-Kennung:

- Dieser Parameter ist nur bei Einsatz in S7-1500 CPU's von Siemens verfügbar.
- Geben Sie unter *HW\_ID* die *HW-Kennung* an, mit deren Hilfe Ihr Modul entsprechend adressiert werden kann. ↪ *Kap. 4.1 "HW-Kennung - HW\_ID" Seite 11*

**RESOLUTION\_CHx**

Geben Sie hier die Auflösung in Inkremente pro Umdrehung für den entsprechenden Kanal an.

**DONE**

Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**SPEED\_CHx**

Aktuell ermittelte Drehzahl des entsprechenden Kanals in Umdrehungen pro Minute (rpm).

**ERROR (Rückgabewert)**

Folgende Codes können zurückgeliefert werden:

| ERROR  | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D0 | Kanal 0 nicht im Status aktiv   |
| 0x80D1 | Kanal 1 nicht im Status aktiv   |
| 0x80D6 | Kanal 0: Eingangswert RESOLUTION_CH0 = 0                              |
| 0x80D7 | Kanal 1: Eingangswert RESOLUTION_CH1 = 0                              |
| 0x80D8 | Kanal 0: Eingangswert RESOLUTION_CH0 < 0                              |
| 0x80D9 | Kanal 1: Eingangswert RESOLUTION_CH1 < 0                              |
| 0x80DA | Kanal 0: Gemessener Zeitwert = 0                                      |
| 0x80DB | Kanal 1: Gemessener Zeitwert = 0                                      |
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                                      |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                                      |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                               |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                               |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E6 | Kanal 0: Ermittelte Drehzahl > max. (DINT)                            |
| 0x80E7 | Kanal 1: Ermittelte Drehzahl > max. (DINT)                            |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der vorgegebenen Messperiode |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der vorgegebenen Messperiode |

**Fehler des intern aufgerufenen SFC 14**

| Code   | Beschreibung  |
|--------|---|
| 0x808x | Systemfehler am Bus-Koppler   |
| 0x8090 | <i>LADDR_IN</i> ist falsch, mögliche Gründe: <ul style="list-style-type: none"> <li>■ auf dieser Adresse ist kein Modul projektiert</li> <li>■ Einschränkung über die Länge der konsistenten Daten wurde nicht beachtet</li> <li>■ Anfangsadresse im Parameter <i>LADDR_IN</i> wurde nicht hexadezimal angegeben</li> </ul> |
| 0x8093 | Für <i>LADDR_IN</i> existiert kein Bus-Koppler, von dem Sie konsistente Daten lesen können.   |
| 0x80A0 | Beim Zugriff auf die Peripherie wurde ein Zugriffsfehler erkannt.   |
| 0x80B0 | Systemfehler am Bus-Koppler   |
| 0x80B1 | Angegebene Länge des Quellbereichs entspricht nicht der projektierten Nutzdatenlänge.   |

| Code   | Beschreibung   |
|--------|--|
| 0x80B2 | Systemfehler am Bus-Koppler  |
| 0x80B3 | Systemfehler am Bus-Koppler  |
| 0x80C1 | Die Daten des auf der Baugruppe vorangegangenen Leseauftrags sind von der Baugruppe noch nicht bearbeitet. |
| 0x80C2 | Systemfehler am Bus-Koppler  |
| 0x80Fx | Systemfehler am Bus-Koppler  |
| 0x85xy | Systemfehler am Bus-Koppler  |
| 0x8xyy | Allgemeine Fehlerinformation<br>↪ Kap. 4.2 "Allgemeine und spezifische Fehlercodes RET_VAL" Seite 11       |

### 5.1.6 FC 310 ... 313 - Frequenzmessung SLIO

#### Übersicht

Mit folgenden VIPA-spezifischen Funktionen können Sie System SLIO Frequenzmess-Module ansteuern, wenn die Konsistenz der Nutzdaten über das Bus-Protokoll sichergestellt ist und konsistentes Lesen bzw. Schreiben mittels SFC 14 bzw. SFC 15 nicht möglich ist. Innerhalb der Funktionen befinden sich "FM\_..."-Parameter, deren Inhalte konsistent über das Bus-System mit dem entsprechenden Ein- bzw. Ausgabebereich des Frequenzmess-Moduls zu verschalten sind. Mit dem Aufruf der Funktionen werden die entsprechenden "FM\_..."-Parameter von der Funktion automatisch befüllt.

| Baustein | Symbol            | Kommentar                                  |
|----------|-------------------|--|
| FC 310   | FM_CONTROL        | Funktion zur Steuerung der Frequenzmessung |
| FC 311   | FM_CALC_PERIOD    | Funktion zur Berechnung der Periodendauer  |
| FC 312   | FM_CALC_FREQUENCY | Funktion zur Berechnung der Frequenz       |
| FC 313   | FM_CALC_SPEED     | Funktion zur Berechnung der Drehzahl       |

### 5.1.7 FC 310 - FM\_CONTROL - Control Frequenzmessung

#### Beschreibung

Mit dem FC 310 FM\_CONTROL können Sie das System SLIO Frequenzmess-Modul steuern. Da dieser FC keinen Baustein für konsistentes Schreiben intern aufruft, müssen Sie in Ihrem System die konsistente Übertragung der Daten sicherstellen.

#### Parameter

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung                |
|------------|-------------|----------|-----------------|-----------------------------|
| ENABLE_FM  | INPUT       | BOOL     | E, A, M, D, L   | Freigabe<br>Frequenzmessung |
| PRESET_CH0 | INPUT       | DINT     | E, A, M, D, L   | Kanal 0: Messperiode        |
| PRESET_CH1 | INPUT       | DINT     | E, A, M, D, L   | Kanal 1: Messperiode        |



| Parameter            | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|----------------------|-------------|----------|-----------------|--|
| DONE                 | OUTPUT      | BOOL     | E, A, M, D, L   | Fertigmeldung<br>(TRUE = OK)                                   |
| ERROR                | OUTPUT      | WORD     | E, A, M, D, L   | Rückgabewert<br>(0 = OK)                                       |
| FM_PRESET_PERIOD_CH0 | OUTPUT      | DWORD    | E, A, M, D, L   | Sollwertvorgabe für Frequenzmess-Modul Ausgabe-Adresse:<br>+0  |
| FM_PRESET_PERIOD_CH1 | OUTPUT      | DWORD    | E, A, M, D, L   | Sollwertvorgabe für Frequenzmess-Modul Ausgabe-Adresse:<br>+4  |
| FM_CONTROL_CH0       | OUTPUT      | WORD     | E, A, M, D, L   | Sollwertvorgabe für Frequenzmess-Modul Ausgabe-Adresse:<br>+8  |
| FM_CONTROL_CH1       | OUTPUT      | WORD     | E, A, M, D, L   | Sollwertvorgabe für Frequenzmess-Modul Ausgabe-Adresse:<br>+10 |

**ENABLE\_FM**

Durch Setzen von *ENABLE\_FM* wird das entsprechende CONTROL generiert und über *FM\_CONTROL\_CHx* ausgegeben. Sobald der Inhalt von *FM\_CONTROL\_CHx* über das Bussystem konsistent an das Frequenzmess-Modul übertragen wurde, werden beide Messungen der Kanäle gestartet. Durch Rücksetzen von *ENABLE\_FM* wird die Messung beider Kanäle gestoppt, nachdem *FM\_CONTROL\_CHx* über das Bussystem konsistent an das Frequenzmess-Modul übertragen wurde.



*Nur solange die Messung gestartet ist, können ermittelte Werte vom Modul abgerufen werden. Ansonsten erhalten Sie die Fehlermeldung, dass die Kanäle deaktiviert sind.*

**PRESET\_CHx**

Geben Sie hier die Messperiode in  $\mu\text{s}$  für den entsprechenden Kanal an.

Wertebereich:  $1\mu\text{s}$  ... 8 388 607 $\mu\text{s}$

**DONE**

Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**FM\_PRESET\_PERIOD\_CHx**

Dieser Parameter enthält die Messperiode für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +0 bzw. +4 im Ausgabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_CONTROL\_CHx**

Dieser Parameter enthält das CONTROL, welches über *ENABLE\_FM* generiert wird. Der jeweilige Inhalt ist für Kanal 0 bzw. Kanal 1 über das entsprechende Bussystem konsistent mit Adresse +8 bzw. +10 im Ausgabebereich des Frequenzmess-Moduls zu verschalten.

**ERROR (Rückgabewert)**

Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D2 | Kanal 0:<br>Eingangswert Messperiode $\leq 0$                   |
| 0x80D3 | Kanal 1:<br>Eingangswert Messperiode $\leq 0$                   |
| 0x80D4 | Kanal 0:<br>Eingangswert Messperiode $> 8\,388\,607\mu\text{s}$ |
| 0x80D5 | Kanal 1:<br>Eingangswert Messperiode $> 8\,388\,607\mu\text{s}$ |

**5.1.8 FC 311 - FM\_CALC\_PERIOD - Periodendauer berechnen****Beschreibung**

Mit dem FC 311 FM\_CALC\_PERIOD können die Periodendauer der Eingangssignale beider Kanäle berechnen. Da dieser FC keinen Baustein für konsistentes Lesen intern aufruft, müssen Sie in Ihrem System die konsistente Übertragung der Daten sicherstellen.

**Parameter**

| Parameter           | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|---------------------|-------------|----------|-----------------|---|
| FM_PERIOD_CH0       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+0  |
| FM_PERIOD_CH1       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+8  |
| FM_RISING_EDGES_CH0 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+4  |
| FM_RISING_EDGES_CH1 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+12 |
| FM_STATUS_CH0       | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+16 |
| FM_STATUS_CH1       | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+18 |
| DONE                | OUTPUT      | BOOL     | E, A, M, D, L   | Fertigmeldung<br>(TRUE = OK)                              |

| Parameter  | Deklaration | Datentyp | Speicherbereich | Beschreibung                         |
|------------|-------------|----------|-----------------|--------------------------------------|
| ERROR      | OUTPUT      | WORD     | E, A, M, D, L   | Rückgabewert<br>(0 = OK)             |
| PERIOD_CH0 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 0:<br>Berechnete Periodendauer |
| PERIOD_CH1 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 1:<br>Berechnete Periodendauer |

**FM\_PERIOD\_CHx** Dieser Parameter enthält den gemessenen Zeitwert von Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +0 bzw. +4 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_RISING\_EDGES\_CHx** Dieser Parameter enthält die ermittelte Anzahl steigender Flanken für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +8 bzw. +12 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_STATUS\_CHx** Dieser Parameter enthält den Status für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +16 bzw. +18 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**DONE** Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**PERIOD\_CHx** Aktuell ermittelte Periodendauer des entsprechenden Kanals in 100ns.

**ERROR (Rückgabewert)**

Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D0 | Kanal 0 nicht im Status aktiv   |
| 0x80D1 | Kanal 1 nicht im Status aktiv   |
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                                      |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                                      |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80E0 | Kanal 0: Anzahl ermittelter Flanken = 0                               |
| 0x80E1 | Kanal 1: Anzahl ermittelter Flanken = 0                               |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                               |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                               |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der vorgegebenen Messperiode |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der vorgegebenen Messperiode |

**5.1.9 FC 312 - FM\_CALC\_FREQUENCY - Frequenz berechnen****Beschreibung**

Mit dem FC 312 FM\_CALC\_FREQUENCY können Sie die Frequenz der Eingangssignale beider Kanäle berechnen. Da dieser FC keinen Baustein für konsistentes Lesen intern aufruft, müssen Sie in Ihrem System die konsistente Übertragung der Daten sicherstellen.

**Parameter**

| Parameter           | Deklaration | Datentyp | Speicherbereich | Beschreibung   |
|---------------------|-------------|----------|-----------------|--|
| FM_PERIOD_CH0       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul Eingabe-Adresse:<br>+0  |
| FM_PERIOD_CH1       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul Eingabe-Adresse:<br>+8  |
| FM_RISING_EDGES_CH0 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul Eingabe-Adresse:<br>+4  |
| FM_RISING_EDGES_CH1 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul Eingabe-Adresse:<br>+12 |

| Parameter     | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|---------------|-------------|----------|-----------------|---|
| FM_STATUS_CH0 | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+16 |
| FM_STATUS_CH1 | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+18 |
| DONE          | OUTPUT      | BOOL     | E, A, M, D, L   | Rückmeldung<br>(TRUE = OK)                                |
| ERROR         | OUTPUT      | WORD     | E, A, M, D, L   | Rückgabewert<br>(0 = OK)                                  |
| FREQUENCY_CH0 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 0:<br>Berechnete Frequenz                           |
| FREQUENCY_CH1 | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 1:<br>Berechnete Frequenz                           |

**FM\_PERIOD\_CHx** Dieser Parameter enthält den gemessenen Zeitwert für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +0 bzw. +4 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_RISING\_EDGES\_CHx** Dieser Parameter enthält die ermittelte Anzahl steigender Flanken für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +8 bzw. +12 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_STATUS\_CHx** Dieser Parameter enthält den Status für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +16 bzw. +18 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**DONE** Fertigmeldung der Funktion

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**FREQUENCY\_CHx** Aktuell ermittelte Frequenz des entsprechenden Kanals in mHz.

**ERROR (Rückgabewert)** Folgende Codes können zurückgeliefert werden:

| Code   | Beschreibung                     |
|--------|----------------------------------|
| 0x0000 | Kein Fehler                      |
| 0x80D0 | Kanal 0 nicht im Status aktiv    |
| 0x80D1 | Kanal 1 nicht im Status aktiv    |
| 0x80DA | Kanal 0: Gemessener Zeitwert = 0 |
| 0x80DB | Kanal 1: Gemessener Zeitwert = 0 |

| Code   | Beschreibung  |
|--------|---|
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                          |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                          |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                 |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                 |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                   |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                   |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFF            |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFF            |
| 0x80E6 | Kanal 0: Frequenz > 600kHz                                |
| 0x80E7 | Kanal 1: Frequenz > 600kHz                                |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der Messperiode. |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der Messperiode. |

### 5.1.10 FC 313 - FM\_CALC\_SPEED - Drehzahl berechnen

#### Beschreibung

Mit dem FC 313 FM\_CALC\_SPEED können Sie die Drehzahl der Eingangssignale beider Kanäle berechnen. Da dieser FC keinen Baustein für konsistentes Lesen intern aufruft, müssen Sie in Ihrem System die konsistente Übertragung der Daten sicherstellen.

#### Parameter

| Parameter           | Deklaration | Datentyp | Speicherbereich | Beschreibung  |
|---------------------|-------------|----------|-----------------|---|
| FM_PERIOD_CH0       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+0  |
| FM_PERIOD_CH1       | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+8  |
| FM_RISING_EDGES_CH0 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+4  |
| FM_RISING_EDGES_CH1 | INPUT       | DWORD    | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+12 |
| FM_STATUS_CH0       | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+16 |
| FM_STATUS_CH1       | INPUT       | WORD     | E, A, M, D, L   | Istwert von Frequenzmess-Modul<br>Eingabe-Adresse:<br>+18 |

| Parameter      | Deklaration | Datentyp | Speicherbereich | Beschreibung                     |
|----------------|-------------|----------|-----------------|----------------------------------|
| RESOLUTION_CH0 | INPUT       | DINT     | E, A, M, D, L   | Kanal 0:<br>Auflösung des Gebers |
| RESOLUTION_CH1 | INPUT       | DINT     | E, A, M, D, L   | Kanal 1:<br>Auflösung des Gebers |
| DONE           | OUTPUT      | BOOL     | E, A, M, D, L   | Fertigmeldung (TRUE = OK)        |
| ERROR          | OUTPUT      | WORD     | E, A, M, D, L   | Rückgabewert (0 = OK)            |
| SPEED_CH0      | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 0:<br>Berechnete Drehzahl  |
| SPEED_CH1      | OUTPUT      | DINT     | E, A, M, D, L   | Kanal 1:<br>Berechnete Drehzahl  |

**FM\_PERIOD\_CHx** Dieser Parameter enthält den gemessenen Zeitwert für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +0 bzw. +4 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_RISING\_EDGES\_CHx** Dieser Parameter enthält die ermittelte Anzahl steigender Flanken für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +8 bzw. +12 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**FM\_STATUS\_CHx** Dieser Parameter enthält den Status für Kanal 0 bzw. Kanal 1. Der Inhalt ist über das entsprechende Bussystem konsistent mit Adresse +16 bzw. +18 im Eingabebereich des Frequenzmess-Moduls zu verschalten.

**RESOLUTION\_CHx** Geben Sie hier die Auflösung in Inkremente pro Umdrehung für den entsprechenden Kanal an.

**DONE** Zustandsparameter des Funktionsbausteins

- TRUE: Funktion wurde ohne Fehler beendet.
- FALSE: Funktion nicht aktiv bzw. es ist ein Fehler aufgetreten.

**SPEED\_CHx** Aktuell ermittelte Drehzahl des entsprechenden Kanals in Umdrehungen pro Minute (rpm).

**ERROR (Rückgabewert)** Folgende Codes können zurückgeliefert werden:

| ERROR  | Beschreibung  |
|--------|---|
| 0x0000 | Kein Fehler   |
| 0x80D0 | Kanal 0 nicht im Status aktiv   |
| 0x80D1 | Kanal 1 nicht im Status aktiv   |
| 0x80D6 | Kanal 0: Eingangswert RESOLUTION_CH0 = 0                              |
| 0x80D7 | Kanal 1: Eingangswert RESOLUTION_CH1 = 0                              |
| 0x80D8 | Kanal 0: Eingangswert RESOLUTION_CH0 < 0                              |
| 0x80D9 | Kanal 1: Eingangswert RESOLUTION_CH1 < 0                              |
| 0x80DA | Kanal 0: Gemessener Zeitwert = 0                                      |
| 0x80DB | Kanal 1: Gemessener Zeitwert = 0                                      |
| 0x80DC | Kanal 0: Gemessener Zeitwert < 0                                      |
| 0x80DD | Kanal 1: Gemessener Zeitwert < 0                                      |
| 0x80DE | Kanal 0: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80DF | Kanal 1: Gemessener Zeitwert > 0x7FFFFFFF                             |
| 0x80E2 | Kanal 0: Anzahl ermittelter Flanken < 0                               |
| 0x80E3 | Kanal 1: Anzahl ermittelter Flanken < 0                               |
| 0x80E4 | Kanal 0: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E5 | Kanal 1: Anzahl ermittelter Flanken > 0xFFFFFFFF                      |
| 0x80E6 | Kanal 0: Ermittelte Drehzahl > max. (DINT)                            |
| 0x80E7 | Kanal 1: Ermittelte Drehzahl > max. (DINT)                            |
| 0x80E8 | Kanal 0: Keine gültige Messung innerhalb der vorgegebenen Messperiode |
| 0x80E9 | Kanal 1: Keine gültige Messung innerhalb der vorgegebenen Messperiode |

## 5.2 Energiemessung - "Energy Measurement"

### 5.2.1 Übersicht

#### 5.2.1.1 Begriffe

##### Messgröße

Eine *Messgröße* ist eine physikalische Größe, die zu messen ist, z.B. Strom, Spannung oder Temperatur.

##### Messwert

Ein *Messwert* ist ein Wert einer Messgröße, der durch Messung oder durch Berechnung ermittelt wird.

##### ID

Im Modul ist jeder *Messgröße* eine *ID* zugeordnet. Der Zugriff auf den Messwert einer Messgröße erfolgt durch Angabe der entsprechenden *ID*.



- DS-ID**
- Sobald das Modul über die DC 24V Leistungsversorgung versorgt wird, beginnt die Messwerterfassung und der Zählvorgang der Energiezähler wird mit den remanent gespeicherten Zählerständen fortgesetzt. Die Messwerte aller Messgrößen werden unter einer Datensatz-ID *DS-ID* im Modul gespeichert. Hierbei ist folgendes zu beachten:
- Alle Messwerte mit der gleichen *DS-ID* stammen aus der gleichen Messung und sind konsistent.
  - Durch Angabe der *DS-ID* können Sie die einzelnen Messwerte aus der gleichen Messung adressieren.
  - Die *DS-ID* umfasst die Werte 1 ... 15.
  - Zur Aktualisierung der Messwerte ist die *DS-ID* um 1 zu inkrementieren. Nach dem Wert 15 muss wieder die 1 folgen.
  - Wird die *DS-ID* inkrementiert und es liegt noch kein neuer Wert vor, wird der aktuelle Wert geliefert. Hierbei meldet das Energiemess-Modul einen Fehler.
  - *DS-ID* = 0 - Autoinkrement-Modus
    - Mit *DS-ID* = 0 erfolgt eine Anfrage im *Autoinkrement-Modus*. Hierbei liefert das Modul immer den aktuellsten Messwert zurück. Sobald hier ein neuer Messwert vorhanden ist, wird die *DS-ID* innerhalb der Werte 1 ... 15 um 1 inkrementiert. Sollte noch kein neuer Messwert vorliegen, bleibt die *DS-ID* unverändert. Hierbei meldet das Energiemess-Modul einen Fehler.
  - Die Eindeutigkeit eines Messwerts besteht immer aus der *ID* der Messgröße und der *DS-ID*.

- Frame**
- Im Modul können Sie mehrere Messgrößen zu einem Datenpaket (Frame) zusammenfassen, welches in einem Durchgang übertragen wird. Ein Datenpaket umfasst 12Byte Nutzdaten. Unter Berücksichtigung der Nutzdatenlänge von 12Byte können Sie durch Angabe der Messgrößen-*ID* die Inhalte eines Frames definieren. Sie können bis zu 256 Frames (*Frame 0* ... *Frame 255*) konfigurieren. Hierbei ist folgendes zu beachten:
- Die Definition von *Frame 1* bis *Frame 255* erfolgt mit dem Befehl *Set\_Frame*.
  - *Frame 0* mit den entsprechenden Messgrößen kann ausschließlich über die Parametrierung konfiguriert werden.
  - Mit dem Telegrammtyp *Zero Frame* haben Sie Zugriff auf das Datenpaket von *Frame 0*. Nach dem Hochlauf des Moduls erfolgen automatische *Zero Frame*-Anforderungen solange bis vom Kopfmodul die Prozessdatenkommunikation übernommen wird.

- FR-ID**
- Bei der Definition von Frames mittels "*Set Frame*" werden diese über die *FR-ID* mit einer beliebigen Ziffer aus 0 ... 255 versehen. Durch Angabe der *FR-ID* können Sie das entsprechende Frame anfordern.

- Datentyp**
- Nachfolgend sind die Datentypen aufgeführt, welche im Modul verwendet werden. Die Länge ist insbesondere bei der Definition von *Frames* zu berücksichtigen.

| Datentyp | Länge in Byte | Beschreibung                       |
|----------|---------------|------------------------------------|
| UINT_8   | 1             | Ganzzahl 8Bit                      |
| UINT_16  | 2             | Ganzzahl 16Bit                     |
| UINT_32  | 4             | Ganzzahl 32Bit                     |
| INT_8    | 1             | Vorzeichenbehaftete Ganzzahl 8Bit  |
| INT_16   | 2             | Vorzeichenbehaftete Ganzzahl 16Bit |
| INT_32   | 4             | Vorzeichenbehaftete Ganzzahl 32Bit |
| FLOAT    | 4             | 32Bit Gleitkommazahl nach IEEE 754 |

### 5.2.1.2 Funktionsweise

#### Übersicht

- Das Energiemess-Modul dient der Energiemessung eines 3-Phasen Anschlusses. Hierbei ermittelt das Modul neben Spannung, Strom und Phase viele weitere Messgrößen.
- Für manche Messgrößen lassen sich Grenzwerte parametrieren. Bei Über- bzw. Unterschreiten werden entsprechende Alarm-Status-Bits gesetzt. Das Modul unterstützt mehrere Kommandos (CMD). Beispielsweise lassen sich hiermit Alarm-Status-Bits wieder zurücksetzen.
- Mit dem Funktionsbaustein FB 325 und der zugehörigen Datenstruktur vom Typ UDT 325 können Sie Energie-Messwerte und Alarm-Status-Bits des Energiemess-Moduls lesen und Kommandos auf dem Modul ausführen. Hierbei kommuniziert der FB 325 über die zyklischen E/A-Daten (je 16 Byte) des Moduls, welche beim Aufruf des FB 325 entsprechend anzugeben sind.
- Die eigentliche Auftragsschnittstelle ist über die Datenstruktur vom Typ UDT 325 realisiert. Hierdurch ist eine einfache Ansteuerung und Auswertung beispielsweise über ein Touch Panel möglich.

#### Beschaltung des FB 325

- Bei der Projektierung ist auf eine korrekte Verschaltung der Parameter CHANNEL\_IN und CHANNEL\_OUT des FB 325 zu achten. Ansonsten erhalten Sie eine Timeout-Fehlermeldung.
  - CHANNEL\_IN ist auf die 16Byte Eingabedaten des Energiemess-Moduls zu verschalten.
  - CHANNEL\_OUT auf die 16Byte Ausgabedaten des Energiemess-Moduls zu verschalten.

#### Zyklische Messwert-Erfassung

- Indem Sie nach einem PowerON einen manuellen Reset am Energiemess-Modul durchführen, können Sie temporäre Fehlermeldungen vermeiden. Setzen Sie hierzu in der Datenstruktur MEAS\_DATA des FB 325 Bit 7 der Variable *Header.Control\_Global*.
- Mit den Grundeinstellungen der UDT 325 werden alle Messwerte des Energiemess-Moduls mit einer Periode von 1s gelesen und in der Datenstruktur MEAS\_DATA gespeichert. Die Periode können Sie über Variable *Header.Polltime* in der Datenstruktur MEAS\_DATA des FB 325 anpassen.



Zur Validierung der Aktualität Ihrer Messwerte können Sie den Parameter DS-ID auf den Zeitpunkt seiner letzten Änderung überprüfen. Sobald ein neuer Messwert vorhanden ist, wird die DS-ID innerhalb der Werte 1 ... 15 um 1 inkrementiert. Der Zeitpunkt der letzten Änderung ist gleichzeitig das Alter der letzten Messwerte.

#### Manuelle Messwernerfassung

Zur manuellen Erfassung der Messwerte setzen Sie in der Datenstruktur MEAS\_DATA des FB 325 Bit 1 der Variable *Header.Control\_Global*. Ist das Bit gesetzt, werden die Messwerte einmalig vom Energiemess-Modul gelesen und danach das Bit wieder zurück gesetzt.

#### Selektion von Messgrößen

Per Default werden die Messwerte aller *Messgrößen* periodisch gelesen. Sie haben aber die Möglichkeit die *Messgrößen* in der Datenstruktur MEAS\_DATA zu selektieren. Über Bit 0 der Variable *Data.[Name der Messgröße].Read\_Mode* können Sie den Zugriff auf den Messwert der entsprechenden *Messgröße* einstellen. Bitte beachten Sie, dass hier die Messgrößen-IDs in Gruppen zusammengefasst sind. Sobald mindestens ein Messwert einer *Messgröße* einer Gruppe gelesen werden soll, werden die Messwerte aller *Messgrößen* dieser Gruppe gelesen. Soll beispielsweise der Wert der Messgröße mit der ID-Nr. 4 gelesen werden, so werden auch die mit ID-Nr. 5 und 6 gelesen. Es gibt folgende Gruppierungen:

| Gruppe | IDs der Messgrößen | Gruppe | IDs der Messgrößen | Gruppe | IDs der Messgrößen |
|--------|--------------------|--------|--------------------|--------|--------------------|
| 1      | 1, 2, 3            | 6      | 16, 17, 18         | 11     | 31, 32, 33         |
| 2      | 4, 5, 6            | 7      | 19, 20, 21         | 12     | 34, 35, 36         |
| 3      | 7, 8, 9            | 8      | 22, 23, 24         | 13     | 37, 38, 39         |
| 4      | 10, 11, 12         | 9      | 25, 26, 27         | 14     | 40, 41             |
| 5      | 13, 14, 15         | 10     | 28, 29, 30         |        |                    |

Die gelesenen Messwerte werden in die entsprechenden Variablen von *Data.[Name der Messgröße].Value* eingetragen. Bei nicht gelesenen Messwerten ist *Value = 0*.

### Kommando-Schnittstelle

Kommandos können Sie über die Datenstruktur *MEAS\_DATA* durch Setzen der entsprechenden Bits in der Variablen *Header.Cmd* auslösen. Sind mehrere Bits gesetzt, werden sie nacheinander abgearbeitet. Hierbei stehen Ihnen folgende Kommandos zur Verfügung:

- Bit 0: Rücksetzen aller Wirkenergiezähler
- Bit 1: Reset auf dem Stromwandler auslösen
- Bit 2: Rücksetzen von *Status Messung*
- Bit 3: Schreiben der Energiesollwerte aus "SetValues" auf die ID3...ID8.

### Fehlerverhalten

- Fehlermeldungen, welche bei der Initialisierung des Bausteins bzw. beim Lesen von Messwerten auftreten, finden Sie in der Datenstruktur *MEAS\_DATA* unter *Header.Status\_Global*.
- Fehlermeldungen, welche bei der Kommando-Abarbeitung auftreten, finden Sie unter *Header.Status\_Cmd* und die Detailinformationen unter *Header.Error\_ID*
- Im Fehlerfall setzt der Funktionsbaustein die Auftragsbearbeitung fort. Hierbei werden die fehlerhaft ausgeführten Aufträge wiederholt. Die Messwerte in der Datenstruktur *MEAS\_DATA* werden durch Fehlermeldungen nicht beeinflusst.

## 5.2.2 FB 325 - EM\_COM\_R1 - Kommunikation mit 031-1PAxx

### Übersicht

Dieser Baustein ermöglicht die Kommunikation mit den Modulen 031-1PAxx zur Energiezählung und Leistungsmessung. Für die Kommunikation ist ein Datenbaustein erforderlich. Hierbei erhält der DB seine Struktur aus der UDT 325 EM\_DATA\_R1. Der Baustein besitzt folgende Funktionalitäten:

- Grundparameter laden nach dem Anlauf
- Ablage von Parameter, Grenzwerte, Messwerte und Meldungen
- Übertragung konsistenter Messwerte
- Schreiben von Sollwerten
- Definition der Messwerte mittels UDT-Struktur
- Kommunikation mittels Telegrammtyp und ID
- Funktionsdiagnose, Verbindungsüberwachung und Störmeldeauswertung

**Parameter**

| Parameter   | Deklaration | Datentyp | Beschreibung   |
|-------------|-------------|----------|--|
| MODE        | INPUT       | BYTE     | <ul style="list-style-type: none"> <li>0x01 = Datenaustausch via Prozessdaten</li> <li>Aktuell wird nur MODE = 0x01 unterstützt</li> </ul>   |
| CHANNEL_IN  | INPUT       | ANY      | Zeiger auf die Eingangsdaten <ul style="list-style-type: none"> <li>Mit MODE = 0x01 ist ausschließlich Datentyp BYTE und Länge 16 zulässig.</li> <li>Beispiel: P#E100.0 BYTE 16 oder P#DB10.DBX0.0 BYTE 16</li> </ul>                            |
| CHANNEL_OUT | INPUT       | ANY      | Zeiger auf die Ausgangsdaten <ul style="list-style-type: none"> <li>Mit MODE = 0x01 ist ausschließlich Datentyp BYTE und Länge 16 zulässig.</li> <li>Beispiel: P#A100.0 BYTE 16 oder P#DB10.DBX16.0 BYTE 16</li> </ul>                           |
| MEAS_DATA   | IN_OUT      | UDT      | <ul style="list-style-type: none"> <li>UDT für die Messwerte ↪ Kap. 5.2.3 "UDT 325 - EM_DATA_R1 - Datenstruktur für FB 325" Seite 36</li> <li>Bitte beachten Sie, dass diese Struktur nicht in den temporären Lokaldaten liegen darf!</li> </ul> |

**5.2.3 UDT 325 - EM\_DATA\_R1 - Datenstruktur für FB 325**

**5.2.3.1 Struktur**

**UDT 325** Die UDT 325 ist dynamisch aufgebaut und besitzt folgende Grundstruktur.

| UDT-Bereiche   | Beschreibung  |
|----------------|---|
| UDT - Header   | Struktur für die Header-Daten   |
| UDT - Daten    | Gleiche Datenstruktur für die einzelnen <i>Messgrößen</i> . Eine <i>Messgröße</i> ist eine physikalische Größe, die zu messen ist, z.B. Strom, Spannung oder Temperatur. Eine Übersicht der Messgrößen finden Sie im Handbuch zu ihrem Energiemess-Modul. |
| ...            |   |
| UDT - Daten    |   |
| UDT - SetValue | Struktur für die Sollwertvorgabe  |

| UDT - Header   | Deklaration | Datentyp | Beschreibung  |
|----------------|-------------|----------|---|
| Timeout        | INPUT       | TIME     | <ul style="list-style-type: none"> <li>Timeout-Zeit für die Auftragsbearbeitung. Bei Überschreiten von <i>Timeout</i> wird der Auftrag abgebrochen und eine entsprechende Fehlermeldung ausgegeben.</li> </ul>  |
| Polltime       | INPUT       | TIME     | <ul style="list-style-type: none"> <li>Intervall für das periodische Lesen</li> <li><i>Polltime</i> ist nur relevant, wenn die Messwerte im Intervall von <i>Polltime</i> periodisch gelesen werden d.h. wenn Bit 0 von <i>Header.Control_Global</i> gesetzt ist. Ist <i>Polltime</i> kleiner als das schnellstmögliche Intervall, werden die Messwerte im schnellstmöglichen Intervall gelesen.</li> </ul> |
| Control_Global | INPUT       | BYTE     | 0: nicht aktiv, 1: aktiv <ul style="list-style-type: none"> <li>Bit 0: Periodische Ausführung entsprechend der <i>Polltime</i> (Default)</li> <li>Bit 1: Einmalige Ausführung - Bit wird nach der Ausführung zurückgesetzt.</li> <li>Bit 6 ... 2: reserviert</li> <li>Bit 7: Neuinitialisierung des Bausteins, indem die Konfiguration neu gesendet wird</li> </ul>   |

| UDT - Header           | Deklaration | Datentyp                     | Beschreibung   |
|------------------------|-------------|------------------------------|--|
| Status_Global          | OUTPUT      | BYTE                         | Bausteinstatus <ul style="list-style-type: none"> <li>0x00: Nicht bearbeitet</li> <li>0x01: In Bearbeitung (BUSY)</li> <li>0x02: Fertig ohne Fehler (DONE)</li> <li>0x80: Fehler bei der Bearbeitung (ERROR)</li> </ul>  |
| Status<br>Alarm_Global | OUTPUT      | BYTE                         | Entspricht B3: Header-Byte 3 - <i>Sammelstatus</i> <ul style="list-style-type: none"> <li>Bit 0: Frequenz <math>F_{MAX}</math> überschritten</li> <li>Bit 1: Frequenz <math>F_{MIN}</math> unterschritten</li> <li>Bit 2: Temperatur <math>T_{MAX}</math> überschritten</li> <li>Bit 3: Spannung <math>VRMS_{MAX}</math> überschritten</li> <li>Bit 4: Spannung <math>VRMS_{MIN}</math> unterschritten</li> <li>Bit 5: Wirkungsgrad <math>PF_{MIN}</math> unterschritten</li> <li>Bit 6: Strom <math>IRMS_{MAX}</math> überschritten</li> <li>Bit 7: reserviert</li> </ul> |
| Cmd                    | INPUT       | BYTE                         | 0: nicht aktiv, 1: aktiv <ul style="list-style-type: none"> <li>Bit 0: Rücksetzen aller Wirkenergiezähler</li> <li>Bit 1: Reset auf dem Stromwandler auslösen</li> <li>Bit 2: Rücksetzen von <i>Status Messung</i></li> <li>Bit 3: Schreiben der Energiesollwerte aus "SetValues" auf die ID3 ... 8 und ID 39 ... 40.</li> </ul> <p>Sind mehrere Bits gesetzt, werden sie nacheinander abgearbeitet.</p> <p>Hinweis: Schreiben der Energiesollwerte setzt eine Protokollversion Major <math>\geq 1</math> Minor <math>\geq 1</math> voraus!</p>                            |
| Status_Cmd             | OUTPUT      | BYTE                         | Status Kommando <ul style="list-style-type: none"> <li>0x00: Nicht bearbeitet</li> <li>0x01: In Bearbeitung (BUSY)</li> <li>0x02: Fertig ohne Fehler (DONE)</li> <li>0x80: Fehler bei der Bearbeitung (ERROR) - siehe Error_ID</li> </ul>  |
| Jobtime                | OUTPUT      | TIME                         | <ul style="list-style-type: none"> <li>Zeitdauer, welche für das Auslesen der Messwerte bzw. für das Ausführen eines Kommandos erforderlich war.</li> </ul>  |
| DsID                   | OUTPUT      | BYTE                         | Nummer der aktuellen DS-ID<br>↪ "DS-ID" Seite 33   |
| Frame_ID               | OUTPUT      | BYTE                         | Nummer der aktuellen FR-ID<br>↪ "FR-ID" Seite 33   |
| Error_ID               | OUTPUT      | WORD                         | Detaillierte Fehlerinformationen   |
| Status_ReadVersion     | OUTPUT      | BYTE                         | Status <i>Read FW Version</i> <ul style="list-style-type: none"> <li>0x00: nie ausgeführt</li> <li>0x01: Busy</li> <li>0x02: Done</li> <li>0x80: Error</li> </ul>  |
| Reserve                | STATIC      | ARRAY of<br>BYTE<br>(1...15) | reserviert   |
| VersionInfo            |             | Struct                       | Die Firmware-Version wird automatisch ermittelt  |
| FirmwareMajor          | OUTPUT      | Byte                         | Firmware Version: Major  |
| FirmwareMinor          | OUTPUT      | Byte                         | Firmware Version: Minor  |

| UDT - Header     | Deklaration | Datentyp | Beschreibung                |
|------------------|-------------|----------|-----------------------------|
| FirmwareRevision | OUTPUT      | Byte     | Firmware Revision           |
| ProtocolMajor    | OUTPUT      | Byte     | Protokoll Version: Major    |
| ProtocolMinor    | OUTPUT      | Byte     | Protokoll Version: Minor    |
| ProtocolRevision | OUTPUT      | Byte     | Protokoll Version: Revision |
| ChipDateYear     | OUTPUT      | Word     | Datum Mess-Chip: Jahr       |
| ChipDateMonth    | OUTPUT      | Byte     | Datum Mess-Chip: Monat      |
| ChipDateDay      | OUTPUT      | Byte     | Datum Mess-Chip: Tag        |

**Gleiche Datenstruktur für die einzelnen Messgrößen. Eine Übersicht der Messgrößen finden Sie im Handbuch zu ihrem Energiemess-Modul.**

| UDT - Daten | Deklaration | Datentyp | Beschreibung  |
|-------------|-------------|----------|---|
| Name        | IN_OUT      | STRUCT   | ■ Name der Messgröße  |
| Read_Mode   | INPUT       | BYTE     | ■ Bit 0: Zugriff auf den Messwert der Messgröße<br>– 0: Messwert soll nicht gelesen werden.<br>– 1: Messwert soll gelesen werden. |
| Value       | OUTPUT      | DWORD    | ■ Aktueller Messwert  |

| UDT - SetValue             | Deklaration | Datentyp | Beschreibung   |
|----------------------------|-------------|----------|--|
| SetValues                  |             | STRUCT   |  |
| EN_L1_CONSUMED             | INPUT       | DWORD    | Sollwert für Wirkenergie L1 Verbraucher: UINT32, 1Wh   |
| EN_L1_DELIVERED            | INPUT       | DWORD    | Sollwert für Wirkenergie L1 Erzeuger: UINT32, 1Wh  |
| EN_L2_CONSUMED             | INPUT       | DWORD    | Sollwert für Wirkenergie L2 Verbraucher: UINT32, 1Wh   |
| EN_L2_DELIVERED            | INPUT       | DWORD    | Sollwert für Wirkenergie L2 Erzeuger: UINT32, 1Wh  |
| EN_L3_CONSUMED             | INPUT       | DWORD    | Sollwert für Wirkenergie L3 Verbraucher: UINT32, 1Wh   |
| EN_L3_DELIVERED            | INPUT       | DWORD    | Sollwert für Wirkenergie L3 Erzeuger: UINT32, 1Wh  |
| EXCESS_ACTIVE_EN_CONSUME   | INPUT       | DWORD    | Sollwert für Überlauf Energiezähler Phase 1 ... 3 Verbraucher<br>■ 0xXX112233<br>– XX: nicht genutzt<br>– 11: Sollwert (Byte) für Überlauf Energiezähler Phase 1 Verbraucher<br>– 22: Sollwert (Byte) für Überlauf Energiezähler Phase 2 Verbraucher<br>– 33: Sollwert (Byte) für Überlauf Energiezähler Phase 3 Verbraucher<br>Wird bei Überlauf des Energiezählers (ID = 1) um 1 inkrementiert |
| EXCESS_ACTIVE_EN_DELIVERED | INPUT       | DWORD    | Sollwert für Überlauf Energiezähler Phase 1 ... 3 Erzeuger<br>■ 0xXX112233<br>– XX: nicht genutzt<br>– 11: Sollwert (Byte) für Überlauf Energiezähler Phase 1 Erzeuger<br>– 22: Sollwert (Byte) für Überlauf Energiezähler Phase 2 Erzeuger<br>– 33: Sollwert (Byte) für Überlauf Energiezähler Phase 3 Erzeuger<br>Wird bei Überlauf des Energiezählers (ID = 2) um 1 inkrementiert             |

### 5.2.3.2 Fehlerrückmeldung

| ERROR ID | Beschreibung  |
|----------|---|
| 0x0000   | kein Fehler   |
| 0x8060   | Fehler: Eine aktuellere Protokollversion ist erforderlich   |
| 0x8070   | Fehler: Parameter MODE  |
| 0x8073   | Fehler: Parameter CHANNEL_IN passt nicht zu MODE  |
| 0x8074   | Fehler: Parameter CHANNEL_OUT passt nicht zu MODE   |
| 0x8080   | Fehler: "SET Frame": Timeout beim Zugriff erkannt   |
| 0x8081   | Fehler: "READ Frame": Timeout beim Zugriff erkannt  |
| 0x8082   | Fehler: "CMD Frame": Timeout beim Zugriff erkannt   |
| 0x8083   | Fehler: Timeout beim automatischen Lesen der Firmwareinformation  |
| 0x80A1   | Status Kommunikation: Fehler: Datensatz konnte nicht aktualisiert werden  |
| 0x80A2   | Status Kommunikation: Fehler: "DS-ID"   |
| 0x80A3   | Status Kommunikation: Fehler: Telegrammlänge  |
| 0x80A4   | Status Kommunikation: Fehler: Frame zu groß   |
| 0x80A5   | Status Kommunikation: Fehler: Frame nicht definiert   |
| 0x80A6   | Status Kommunikation: Fehler: Messgröße nicht vorhanden   |
| 0x80A7   | Status Kommunikation: "CMD Frame" - Kommando konnte nicht ausgeführt werden   |
| 0x80A8   | Status Kommunikation: Fehler: "SetFrame" - Ungültige Framedefinition (Set Frame)  |
| 0x80A9   | Status Kommunikation: Fehler: Telegrammtyp nicht vorhanden - ungültige Anfrage  |
| 0x80AA   | Status Kommunikation: Fehler: Parameter - der letzte Parametersatz war ungültig   |
| 0x80AB   | Fehler: Messmodul BUSY, es werden keine neuen Daten geliefert   |
| 0x80AE   | Externer Fehler - Bitte kontaktieren Sie unseren Support  |
| 0x80AF   | Interner Fehler: Aufgrund einer temporären Störung bei der Verarbeitung der Messdaten konnten diese nicht aktualisiert werden. Sollte dieser Fehler öfter auftreten, kontaktieren Sie bitte unsere Hotline. |

## 5.3 Motion-Module - "Motion Modules"

### 5.3.1 Übersicht

#### Bausteine

Mit den nachfolgend aufgeführten Bausteinen haben Sie Zugriff auf die System SLIO Motion-Module:

- FB 320 - ACYC\_RW - Azyklischer Zugriff auf System SLIO Motion-Modul
- FB 321 - ACYC\_DS - Azyklische Parametrierung System SLIO Motion-Modul
- UDT 321 - ACYC\_OBJECT-DATA - Datenstruktur für FB 321

#### Unterstützte Motion-Module

Folgende System SLIO Motion-Module werden unterstützt:


- 054-1BA00: FM 054 Motion Modul - Stepper
- 054-1CB00: FM 054 Motion Modul - 2xDC
- 054-1DA00: FM 054 Motion Modul - Pulse Train RS422



**Index - Subindex**

Die System SLIO Motion-Modul stellen ihre Daten wie z.B. "Profilgeschwindigkeit" über ein Objektverzeichnis zur Verfügung. In diesem Objektverzeichnis sind die Objekte organisiert und durch eine eindeutige Nummer, bestehend aus *Index* und *Subindex* adressierbar. Die Nummer wird wie folgt angegeben:


|                     |                     |   |                    |
|---------------------|---------------------|---|--------------------|
| 0x                  | Index (hexadezimal) | - | Subindex (dezimal) |
| Beispiel: 0x8400-03 |                     |   |                    |


 *Zur besseren Strukturierung und Erweiterung wurde beim System SLIO Motion-Modul eine andere Objektnummerierung (Index-Vergabe) gegenüber dem Standard CiA 402 gewählt.*

**Index - Bereiche**

Durch die Aufteilung in *Index* und *Subindex* ist eine Gruppierung möglich. Die einzelnen Bereiche sind in Gruppen zusammengehöriger Objekte gegliedert. Dieses Objektverzeichnis ist bei den System SLIO Motion Modul wie folgt strukturiert:

| Index-Bereich     | Inhalt                                      |
|-------------------|---|
| 0x1000 bis 0x6FFF | Allgemeine Daten und Systemdaten            |
| 0x7000 bis 0x7FFF | Daten der digitalen Ein- und Ausgabeeinheit |
| ab 0x8000         | Daten der Achse bzw. Antriebe               |

 *Näheres zum Aufbau des Objektverzeichnis finden Sie im Handbuch zu ihrem Motion-Modul.*

 *Jedes Objekt verfügt über einen Subindex 0. Durch Aufruf eines Objekts mit Subindex 0 bekommen Sie die Anzahl der verfügbaren Subindizes des entsprechenden Objekts zurückgeliefert.*



**E/A-Adressbereich** Die Motion-Module belegen eine gewisse Anzahl an Bytes im E/A-Adressbereich.

| Kopfmodul           | Rückwandbus | Motion-Modul |                   |
|---------------------|-------------|--------------|-------------------|
| CPU bzw. Buskoppler | →<br>←      | Prozessdaten | Azyklischer Kanal |

Über den *Azyklischen Kanal* können Sie azyklisch Schreib- und Lesebefehle ausführen. Hierzu wurden in den Ein-/Ausgabe-Bereich der Motion-Module Datenbereiche für die azyklische Kommunikation implementiert. Dieser Bereich umfasst 8Byte Ausgabe- und 8Byte Eingabe-Daten. Bei Einsatz der Bausteine erfolgt die Kommunikation über den *Azyklischen Kanal*.



*Der Datenaustausch mit dem Motion-Modul muss über die Länge der Ein- bzw. Ausgabedaten konsistent sein! Es wird daher die Ansteuerung über das Prozessabbild empfohlen. Sie können aber auch SFC 14 und 15 für konsistentes Lesen und Schreiben der Eingabe- bzw. Ausgabe-Daten verwenden.*

### Beschaltung der FBs

- Bei der Projektierung ist auf eine korrekte Verschaltung der Parameter CHANNEL\_IN und CHANNEL\_OUT der FBs zu achten.
  - CHANNEL\_IN ist auf die Eingabedaten des *Azyklischen Kanals* des Motion-Moduls zu verschalten.
  - CHANNEL\_OUT auf die Ausgabedaten des *Azyklischen Kanals* des Motion-Moduls zu verschalten.

Ausgehend von der Basisadresse ist die Anfangsadresse des *Azyklischen Kanals* für die Ein- und Ausgabedaten über folgenden Offset zu erreichen:

- 054-1BA00: FM 054 - Stepper: Basis-Adresse + 26
- 054-1CB00: FM 054 - 2xDC: Basis-Adresse + 50
- 054-1DA00: FM 054 - Pulse Train RS422: Basis-Adresse + 26

#### Beispiel mit Basisadresse 256:

```
CHANNEL_IN    :=P#E 282.0 BYTE 10 // Basisadresse 256 + 26
CHANNEL_OUT   :=P#A 282.0 BYTE 10 // Basisadresse 256 + 26
```



*Bitte beachten Sie, dass Sie eine Länge von 10Byte angeben, obwohl der Azyklische Kanal intern 8Byte verwendet!*

## 5.3.2 FB 320 - ACYC\_RW - Azyklischer Zugriff auf System SLIO Motion-Modul

### Beschreibung

Mit diesem Baustein können Sie aus Ihrem Anwenderprogramm auf das Objektverzeichnis der System SLIO Motion-Module zugreifen. Hierbei verwendet der Baustein einen azyklischen Kommunikationskanal, auf Basis einer Anfrage-/Antwort-Sequenz. Dieser ist Bestandteil des Ein-/Ausgabebereichs des Motion-Moduls.

Folgende System SLIO Motion-Module werden unterstützt:

- 054-1BA00: FM 054 - Stepper
- 054-1CB00: FM 054 - 2xDC
- 054-1DA00: FM 054 - Pulse Train RS422



Da der FB 321 intern den FB 320 aufruft und beide Bausteine auf die gleiche Datenbasis zugreifen, dürfen Sie je Antriebskanal (sofern mehrkanalig) nur einen dieser Bausteine in Ihrem Anwenderprogramm verwenden! Auch darf dieser Baustein nur einmal pro Zyklus aufgerufen werden!



Der Datenaustausch mit dem Motion-Modul muss über die Länge der Ein- bzw. Ausgabedaten konsistent sein! Es wird daher die Ansteuerung über das Prozessabbild empfohlen. Sie können aber auch SFC 14 und 15 für konsistentes Lesen und Schreiben der Eingabe- bzw. Ausgabe-Daten verwenden.

## Parameter

| Parameter    | Deklaration | Datentyp | Beschreibung   |
|--------------|-------------|----------|--|
| REQUEST      | IN          | BOOL     | Mit Flanke 0-1 wird die Auftragsbearbeitung gestartet.   |
| MODE         | IN          | BYTE     | Geben Sie hier 0x01 für das azyklische Protokoll an  |
| COMMAND      | IN          | BYTE     | 0x11 = Lesen eines Datenobjekts (max. 4Byte)<br>0x21 = Schreiben eines Datenobjekts (max. 4Byte)   |
| INDEX        | IN          | WORD     | Index des Objekts im Objektverzeichnis - siehe Handbuch zum System SLIO Motion-Modul.  |
| SUBINDEX     | IN          | BYTE     | Subindex des Objekts im Objektverzeichnis - siehe Handbuch zum System SLIO Motion-Modul.   |
| WRITE_LENGTH | IN          | DINT     | Länge der zu schreibenden Daten in Byte (max. 4Byte)   |
| WRITE_DATA   | IN          | ANY      | Zeiger auf die zu schreibenden Daten.  |
| READ_DATA    | IN          | ANY      | Zeiger auf die gelesenen Daten.  |
| CHANNEL_IN   | IN          | ANY      | Zeiger auf den Beginn des azyklischen Kanals im Eingabe-Bereich des Motion-Moduls.<br>Tragen Sie als Länge 10Byte ein.<br>Beispiele: P#E100.0 BYTE 10 oder P#DB10.DBX0.0 BYTE 10 |
| CHANNEL_OUT  | IN          | ANY      | Zeiger auf den Beginn des azyklischen Kanal im Ausgabe-Bereich des Motion-Moduls.<br>Tragen Sie als Länge 10Byte ein.<br>Beispiele: P#A100.0 BYTE 10 oder P#DB10.DBX10.0 BYTE 10 |
| READ_LENGTH  | OUT         | DINT     | Länge der empfangenen Daten in Byte.<br>Dieser Wert ist auf ein Vielfaches von 4 aufzurunden, da die Längenangabe nicht übertragen wird.   |
| DONE         | OUT         | BOOL     | 1: Auftrag wurde fehlerfrei ausgeführt   |
| BUSY         | OUT         | BOOL     | 0: Kein Auftrag in Bearbeitung<br>1: Auftrag wird bearbeitet   |
| ERROR        | OUT         | BOOL     | 0: Kein Fehler<br>1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter <i>ERROR_ID</i> angezeigt   |
| ERROR_ID     | OUT         | WORD     | Detaillierte Fehlerinformationen   |



Bitte beachten Sie, dass bei den Parametern `WRITE_DATA` und `READ_DATA` keine Überprüfung auf Datentyp und Länge stattfindet!

### Verhalten der Bausteinparameter

- Ausschließlichkeit der Ausgänge:
  - Die Ausgänge `BUSY`, `DONE` und `ERROR` schließen sich gegenseitig aus. Es kann immer nur einer dieser Ausgänge zur gleichen Zeit `TRUE` sein.
  - Sobald der Eingang `REQUEST` `TRUE` wird, muss einer der Ausgänge `TRUE` werden.
- Ausgangs-Zustand
  - Die Ausgänge `DONE`, `ERROR`, `ERROR_ID` und `READ_LENGTH` werden mit einer Flanke 1-0 am Eingang `REQUEST` zurückgesetzt, wenn der Funktionsbaustein nicht aktiv ist (`BUSY = FALSE`).
  - Eine Flanke 1-0 an `REQUEST` beeinflusst die Auftragsbearbeitung nicht.
  - Falls `REQUEST` bereits während der Auftragsbearbeitung zurückgesetzt wird, so ist sichergestellt, dass einer der Ausgänge am Ende des Auftrags für einen SPS-Zyklus gesetzt wird. Erst danach werden die Ausgänge zurückgesetzt.
- Eingangs-Parameter
  - Die Eingangs-Parameter werden mit Flanke 0-1 an `REQUEST` übernommen. Zur Änderung von Parametern, müssen Sie den Auftrag neu triggern.
  - Tritt während der Auftragsbearbeitung erneut eine Flanke 0-1 an `REQUEST` auf, wird ein Fehler ausgegeben, kein neues Kommando aktiviert und die Antwort vom laufenden Kommando verworfen!
- Fehlerbehandlung
  - Der Baustein besitzt 2 Fehlerausgänge zur Anzeige von Fehlern während der Auftragsbearbeitung. `ERROR` zeigt den Fehler an und `ERROR_ID` gibt eine ergänzende Fehlernummer aus.
  - Die Ausgänge `DONE` und `READ_LENGTH` bezeichnen eine erfolgreiche Auftragsbearbeitung und werden nicht gesetzt, wenn `ERROR` `TRUE` wird.
- Verhalten des `DONE` Ausgangs
  - Der `DONE` Ausgang wird gesetzt, wenn ein Auftrag erfolgreich ausgeführt wurde.
- Verhalten des `BUSY` Ausgangs
  - Der `BUSY` Ausgang zeigt an, dass der Funktionsbaustein aktiv ist.
  - `BUSY` wird sofort mit der Flanke 0-1 an `REQUEST` gesetzt und wird erst zurückgesetzt, wenn der Auftrag erfolgreich oder auch nicht erfolgreich beendet wurde.
  - Solange `BUSY` `TRUE` ist, muss der Baustein zyklisch aufgerufen werden um das Kommando ausführen zu können.



Tritt während der Auftragsbearbeitung erneut eine Flanke 0-1 an `REQUEST` auf, wird ein Fehler ausgegeben, kein neues Kommando aktiviert und die Antwort vom laufenden Kommando verworfen!







### ERROR\_ID

| ERROR_ID | Beschreibung  |
|----------|---|
| 0x0000   | Kein Fehler vorhanden   |
| 0x8070   | Fehlerhafter Parameter <code>MODE</code>                                      |
| 0x8071   | Fehlerhafter Parameter <code>COMMAND</code>                                   |
| 0x8072   | Parameter <code>WRITE_LENGTH</code> überschreitet die maximal Größe           |
| 0x8073   | Parameter <code>CHANNEL_IN</code> passt nicht zum Parameter <code>MODE</code> |

| ERROR_ID | Beschreibung  |
|----------|---|
| 0x8074   | Parameter <i>CHANNEL_OUT</i> passt nicht zum Parameter <i>MODE</i>                          |
| 0x8075   | Nicht zulässiger Befehl (Flanke 0-1 bei <i>REQUEST</i> während Auftrag ausgeführt wird)     |
| 0x8081   | Fehler - Lesezugriff - Daten nicht vorhanden<br>Befehl wurde nicht ausgeführt!              |
| 0x8091   | Fehler - Schreibzugriff - Daten nicht vorhanden<br>Befehl wurde nicht ausgeführt!           |
| 0x8092   | Fehler - Schreibzugriff - Datenbereich überschritten<br>Befehl wurde nicht ausgeführt!      |
| 0x8093   | Fehler - Schreibzugriff - Daten können nur gelesen werden<br>Befehl wurde nicht ausgeführt! |
| 0x8094   | Fehler - Schreibzugriff - Daten sind schreibgeschützt<br>Befehl wurde nicht ausgeführt!     |
| 0x8099   | Fehler während der azyklischen Kommunikation<br>Befehl wurde nicht ausgeführt!              |

### Programm-Struktur

Ist kein Auftrag aktiv, so sind alle Ausgabe-Parameter auf 0 zu setzen. Mit einer Flanke 0-1 an *REQUEST* aktivieren Sie nach folgender Vorgehensweise einen Auftrag:

1.  Überprüfen Sie, ob bereits ein Auftrag aktiv ist, ggf. Auftrag abbrechen und Fehler ausgeben.
  - ⇒ Abfragen auf *DONE* = 1 bzw. *BUSY* = 0
2.  Beschalten Sie die Eingabeparameter:
  - *MODE*
  - *COMMAND*
  - *WRITE\_LENGTH*
  - *CHANNEL\_IN*
  - *CHANNEL\_OUT*
  - ⇒ Im Fehlerfall Auftrag abbrechen, ansonsten weiter mit Schritt 3.
3.  Eingangsparameter intern speichern.
4.  Das gewünschte Kommando ausführen und warten bis dieses ausgeführt wurde.
5.  Ergebnis der Kommandoausführung intern speichern und ausgeben.
6.  Alle Ausgabeparameter wieder auf 0 setzen.

### 5.3.3 FB 321 - ACYC\_DS - Azyklische Parametrierung System SLIO Motion-Modul

#### Beschreibung

Mit diesem Baustein können Sie aus Ihrem Anwenderprogramm Ihr Motion-Modul parametrieren. Hierbei können Sie in einem Datenbaustein Ihre Parameter in Form einer *Objektliste* ablegen und diese über den azyklischen Kommunikationskanal in Ihr Motion-Modul übertragen.

Folgende System SLIO Module werden unterstützt:

- 054-1BA00: FM 054 Motion Modul - Stepper
- 054-1CB00: FM 054 Motion Modul - 2xDC
- 054-1DA00: FM 054 Motion Modul - Pulse Train RS422



*Da der FB 321 intern den FB 320 aufruft und beide Bausteine auf die gleiche Datenbasis zugreifen, dürfen Sie je Antriebskanal (sofern mehrkanalig) nur einen dieser Bausteine in Ihrem Anwenderprogramm verwenden! Auch darf dieser Baustein nur einmal pro Zyklus aufgerufen werden!*

## Parameter

| Parameter     | Deklaration | Datentyp | Beschreibung   |
|---------------|-------------|----------|--|
| REQUEST       | IN          | BOOL     | Mit Flanke 0-1 wird die Auftragsbearbeitung gestartet.   |
| MODE          | IN          | BYTE     | Geben Sie hier 0x01 für das azyklische Protokoll an.   |
| READ_BACK     | IN          | BOOL     | 0: Geschriebene Objekte werden nicht zurückgelesen.<br>1: Geschriebene Objekte werden direkt nach dem Schreibvorgang zurückgelesen und einem Vergleich unterzogen.               |
| GROUP         | IN          | WORD     | 0x01...0x7F: Selektion einer Gruppe in der Objektliste.<br>0xFF: Selektion aller Objekte in der Objektliste.   |
| OBJECT_DATA   | IN          | ANY      | Zeiger auf die UDT. ↪ <i>Kap. 5.3.4 "UDT 321 - ACYC_OBJECT_DATA - Datenstruktur für FB 321" Seite 48</i>   |
| CHANNEL_IN    | IN          | ANY      | Zeiger auf den Beginn des azyklischen Kanals im Eingabe-Bereich des Motion-Moduls.<br>Tragen Sie als Länge 10Byte ein.<br>Beispiele: P#E100.0 BYTE 10 oder P#DB10.DBX0.0 BYTE 10 |
| CHANNEL_OUT   | IN          | ANY      | Zeiger auf den Beginn des azyklischen Kanal im Ausgabe-Bereich des Motion-Moduls.<br>Tragen Sie als Länge 10Byte ein.<br>Beispiele: P#A100.0 BYTE 10 oder P#DB10.DBX10.0 BYTE 10 |
| DONE          | OUT         | BOOL     | 1: Auftrag wurde fehlerfrei ausgeführt.  |
| BUSY          | OUT         | BOOL     | 0: Kein Auftrag in Bearbeitung.<br>1: Auftrag wird bearbeitet.   |
| DATASET_INDEX | OUT         | INT      | Objekt, welches aktuell bearbeitet wird.   |
| ERROR         | OUT         | BOOL     | 0: Kein Fehler<br>1: Fehler aufgetreten. Die Fehlerursache wird über den Parameter <i>ERROR_ID</i> angezeigt.  |
| ERROR_ID      | OUT         | WORD     | Detaillierte Fehlerinformationen   |

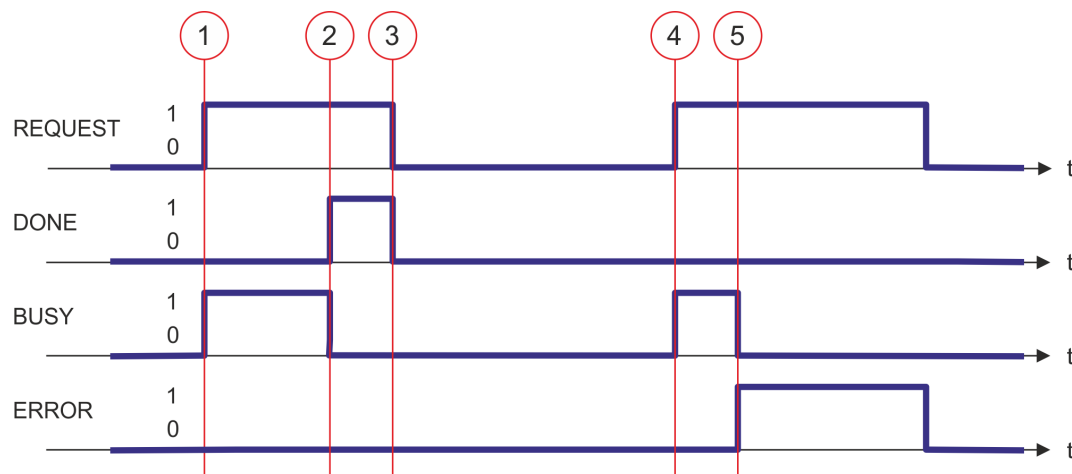
**Verhalten der Bausteinparameter**

- **Ausschließlichkeit der Ausgänge:**
  - Die Ausgänge *BUSY*, *DONE* und *ERROR* schließen sich gegenseitig aus. Es kann immer nur einer dieser Ausgänge zur gleichen Zeit TRUE sein.
  - Sobald der Eingang *REQUEST* TRUE wird, muss einer der Ausgänge TRUE werden.
- **Ausgangs-Zustand**
  - Nach Ende der Abarbeitung der Objektliste werden mit einer Flanke 1-0 an *REQUEST* die Ausgänge *DONE*, *ERROR*, *ERROR\_ID* und *DATASET\_INDEX* zurückgesetzt.
  - Falls *REQUEST* bereits während der Abarbeitung der Objektliste zurückgesetzt wird, so ist sichergestellt, dass die ganze Objektliste abgearbeitet wird.
  - Am Ende der Abarbeitung wird bei fehlerfreier Ausführung *DONE* für einen SPS-Zyklus gesetzt wird. Erst danach werden die Ausgänge zurückgesetzt.
- **Eingangs-Parameter**
  - Die Eingangs-Parameter werden mit Flanke 0-1 an *REQUEST* übernommen. Zur Änderung von Parametern, müssen Sie den Auftrag neu triggern.
  - Tritt während der Auftragsbearbeitung erneut eine Flanke 0-1 an *REQUEST* auf, wird ein Fehler ausgegeben (unzulässige Kommandofolge) und die Abarbeitung der Objektliste beendet.
- **Eingangs-Parameter *READ\_BACK***
  - Bei aktiviertem Parameter *READ\_BACK* werden zu schreibende Objekte unmittelbar nach dem Schreibauftrag des Objekts mit einem Leseauftrag zurückgelesen.
  - Der geschriebene und zurückgelesene Wert werden einem Vergleich unterzogen. Bei Gleichheit wird das nächste Objekt bearbeitet. Bei Ungleichheit erfolgt eine Fehlermeldung (*ERROR\_ID* = 0x8079) und die Abarbeitung der Objektliste wird beendet.
- **Eingangs-Parameter *GROUP***
  - In der Objektliste können Sie zur besseren Strukturierung jedem Objekt eine Gruppe zuordnen.
  - Über *GROUP* definieren Sie die Gruppe, deren Parameter zu übertragen sind. 0x01...0x7F: Objekte der gewählten Gruppe übertragen. 0xFF: Objekte aller Gruppen übertragen.
- **Fehlerbehandlung**
  - Der Baustein besitzt Fehlerausgänge zur Anzeige von Fehlern während der Auftragsbearbeitung. *ERROR* zeigt den Fehler an, *ERROR\_ID* gibt eine ergänzende Fehlernummer aus und *DATASET\_INDEX* zeigt an, bei welchem Objekt der Fehler aufgetreten ist.
  - Der Ausgang *DONE* bezeichnet eine erfolgreiche Auftragsbearbeitung und wird nicht gesetzt, wenn *ERROR* TRUE wird.
- **Verhalten des *DONE* Ausganges**
  - Der *DONE* Ausgang wird gesetzt, wenn ein Auftrag erfolgreich ausgeführt wurde.
- **Verhalten des *BUSY* Ausganges**
  - Der *BUSY* Ausgang zeigt an, dass der Funktionsbaustein aktiv ist.
  - *BUSY* wird sofort mit der Flanke 0-1 an *REQUEST* gesetzt und wird erst zurückgesetzt, wenn der Auftrag erfolgreich oder auch nicht erfolgreich beendet wurde.
  - Solange *BUSY* TRUE ist, muss der Baustein zyklisch aufgerufen werden, um das Kommando ausführen zu können.
- **Verhalten des *DATASET\_INDEX* Ausganges**
  - Der *DATASET\_INDEX* Ausgang zeigt an, welches Objekt der Objektliste aktuell bearbeitet wird.
  - Ist kein Auftrag aktiv, wird *DATASET\_INDEX* = 0 geliefert.
  - Tritt während der Objektbearbeitung ein Fehler auf, zeigt *DATASET\_INDEX* das fehlerverursachende Objekt an.



Tritt während der Auftragsbearbeitung erneut eine Flanke 0-1 an *REQUEST* auf, wird ein Fehler (*ERROR\_ID* = 0x8075) ausgegeben, kein neues Kommando aktiviert und die Antwort vom laufenden Kommando verworfen!

## Zustandsdiagramm



- (1) Mit Flanke 0-1 an *REQUEST* zum Zeitpunkt (1) wird die Abarbeitung gestartet und *BUSY* liefert den Wert TRUE.
- (2) Zum Zeitpunkt (2) ist die Abarbeitung abgeschlossen. *BUSY* liefert den Wert FALSE und *DONE* den Wert TRUE.
- (3) Zum Zeitpunkt (3) ist die Abarbeitung abgeschlossen und *REQUEST* wird gleich FALSE gesetzt und dadurch sämtliche Ausgangsparameter auf FALSE bzw. 0 gesetzt.
- (4) Zum Zeitpunkt (4) wird erneut die Abarbeitung mit einer Flanke 0-1 an *REQUEST* gestartet und *BUSY* liefert den Wert TRUE.
- (5) Zum Zeitpunkt (5) tritt ein Fehler bei der Abarbeitung auf. *BUSY* liefert den Wert FALSE und *ERROR* den Wert TRUE.

## ERROR\_ID

| ERROR_ID | Beschreibung  |
|----------|---|
| 0x0000   | Kein Fehler vorhanden   |
| 0x8070   | Fehlerhafter Parameter <i>MODE</i>  |
| 0x8071   | Fehlerhafter Parameter <i>OBJECT_DATA</i>   |
| 0x8075   | Nicht zulässiger Befehl (Flanke 0-1 bei <i>REQUEST</i> während Auftrag ausgeführt wird) |
| 0x8078   | Fehlerhafter Parameter <i>GROUP</i>   |
| 0x8079   | <i>READ_BACK</i> erkennt einen Fehler (geschriebener und gelesener Wert ungleich)       |
| 0x807A   | Zeiger bei <i>OBJECT_DATA</i> ungültig  |



Innerhalb des Funktionsbausteins erfolgt ein Aufruf des FB 320. Hierbei werden eventuelle Fehler des FB 320 an den FB 321 durchgereicht.  
 ↪ "*ERROR\_ID*" Seite 43



### 5.3.4 UDT 321 - ACYC\_OBJECT-DATA - Datenstruktur für FB 321

#### Datenstruktur für die Objektliste

Die Parameter sind in einem Datenbaustein als *Objektliste* abzulegen, welche aus einzelnen *Objekten* besteht. Die Struktur eines *Objekts* ist über eine UDT definiert.

#### Struktur eines Objekts

| Variable     | Deklaration | Datentyp | Beschreibung  |
|--------------|-------------|----------|---|
| Group        | IN          | WORD     | 0 < Group < 0x80 zulässig   |
| Command      | IN          | BYTE     | 0x11 = Lesen aus dem Objektverzeichnis<br>0x21 = Schreiben in das Objektverzeichnis   |
| Index        | IN          | WORD     | Index des Objekts   |
| Subindex     | IN          | BYTE     | Subindex des Objekts  |
| Write_Length | IN          | BYTE     | Länge der zu schreibenden Daten in Byte   |
| Data_Write   | IN          | DWORD    | Zu schreibende Daten.   |
| Data_Read    | OUT         | DWORD    | Gelesene Daten  |
| State        | OUT         | BYTE     | 0x00 = nie bearbeitet<br>0x01 = <i>BUSY</i> - in Bearbeitung<br>0x02 = <i>DONE</i> - erfolgreich bearbeitet<br>0x80 = <i>ERROR</i> - bei der Bearbeitung ist ein Fehler aufgetreten |



Bitte beachten Sie, dass Sie bei einem Schreibauftrag immer die zum Objekt passende Länge angeben!

#### Beispiel-DB

| Adr. | Name                   | Typ   | Anfangswert | Aktualwert | Kommentar |
|------|------------------------|-------|-------------|------------|-----------|
| 0.0  | Object(1).Group        | WORD  |             |            | 1. Objekt |
| 2.0  | Object(1).Command      | BYTE  |             |            |           |
| 4.0  | Object(1).Index        | WORD  |             |            |           |
| 6.0  | Object(1).Subindex     | BYTE  |             |            |           |
| 7.0  | Object(1).Write_Length | BYTE  |             |            |           |
| 8.0  | Object(1).Data_Write   | DWORD |             |            |           |
| 12.0 | Object(1).Data_Read    | DWORD |             |            |           |
| 16.0 | Object(1).State        | BYTE  |             |            |           |
| 18.0 | Object(2).Group        | WORD  |             |            | 2. Objekt |
| ...  | ...                    | ...   |             |            |           |
| 34.0 | Object(2).State        | BYTE  |             |            | 3. Objekt |
| 36.0 | Object(3).Group        | WORD  |             |            |           |
| ...  | ...                    | ...   |             |            |           |



| Adr. | Name            | Typ  | Anfangswert | Aktualwert | Kommentar |
|------|-----------------|------|-------------|------------|-----------|
| 52.0 | Object(3).State | BYTE |             |            |           |
| ...  | ...             | ...  |             |            | ...       |

## 5.4 RAM nach WLD - "WLD"

### 5.4.1 FB 240 - RAM\_to\_s7prog.wld - RAM nach s7prog.wld

#### Beschreibung

Mit *REQ* = TRUE kopiert dieser Baustein das aktuell geladene Projekt im RAM einer CPU auf eine gesteckte Speicherkarte als s7prog.wld. Bei einer VIPA SPEED7-CPU wird die s7prog.wld immer automatisch nach Umlöschen von einer gesteckten Speicherkarte gelesen. Der FB 240 ruft intern den Baustein SFB 239 mit den entsprechenden Parametern auf. Hierbei werden die Werte für *BUSY* und *RET\_VAL* vom SFB 239 an den FB 240 zurückgegeben.



Bitte beachten Sie, dass dieser Baustein in der Bibliothek für das Siemens TIA Portal nicht enthalten ist.

#### Parameter

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                       |
|-----------|-------------|----------|-----------------|------------------------------------|
| REQ       | IN          | BOOL     | E, A, M, D, L   | Funktionsanstoß mit <i>REQ</i> = 1 |
| BUSY      | OUT         | BOOL     | E, A, M, D, L   | Rückgabewert des SFB 239           |
| RET_VAL   | OUT         | WORD     | E, A, M, D, L   | Rückgabewert des SFB 239           |

### 5.4.2 FB 241 - RAM\_to\_autoload.wld - RAM nach autoload.wld

#### Beschreibung

Mit *REQ* = TRUE kopiert dieser Baustein das aktuell geladene Projekt im RAM einer CPU auf eine gesteckte Speicherkarte als autoload.wld. Bei einer VIPA SPEED7-CPU wird die autoload.wld immer automatisch nach NetZEIN von einer gesteckten Speicherkarte gelesen. Der FB 241 ruft intern den Baustein SFB 239 mit den entsprechenden Parametern auf. Hierbei werden die Werte für *BUSY* und *RET\_VAL* vom SFB 239 an den FB 241 zurückgegeben.



Bitte beachten Sie, dass dieser Baustein in der Bibliothek für das Siemens TIA Portal nicht enthalten ist.

#### Parameter

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung                       |
|-----------|-------------|----------|-----------------|------------------------------------|
| REQ       | IN          | BOOL     | E, A, M, D, L   | Funktionsanstoß mit <i>REQ</i> = 1 |
| BUSY      | OUT         | BOOL     | E, A, M, D, L   | Rückgabewert des SFB 239           |
| RET_VAL   | OUT         | WORD     | E, A, M, D, L   | Rückgabewert des SFB 239           |

## 5.5 System 100V interne E/As - "Onboard I/O System 100V"

### 5.5.1 SFC 223 - PWM - Pulsweitenmodulation

**Beschreibung**

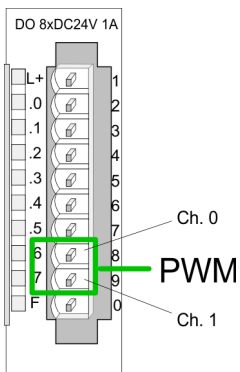
Dieser Baustein dient zur Parametrierung der Pulsweitenmodulation für die letzten beiden Ausgabe-Kanäle von X5.



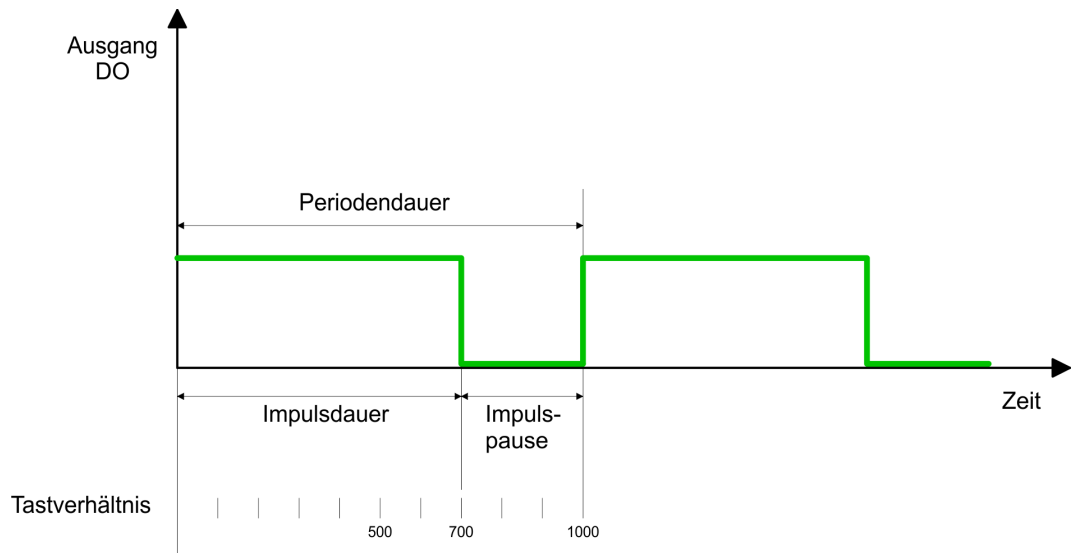
Bitte beachten Sie, dass dieser Baustein in der Bibliothek für das Siemens TIA Portal nicht enthalten ist.

**Parameter**

| Parameter | Deklaration | Typ  | Beschreibung                    |
|-----------|-------------|------|---------------------------------|
| CHANNEL   | IN          | INT  | Nummer des Ausgabekanal für PWM |
| ENABLE    | IN          | BOOL | Auftrag anstoßen                |
| TIMEBASE  | IN          | INT  | Zeitbasis                       |
| PERIOD    | IN          | DINT | Periode der PWM                 |
| DUTY      | IN          | DINT | Tastverhältnis in Promille      |
| MINLEN    | IN          | DINT | Minimale Impulsdauer            |
| RET_VAL   | OUT         | WORD | Rückgabewert (0 = OK)           |



- ➔ Sie geben Zeitbasis, Periode, Tastverhältnis und minimale Impulsdauer vor. Hieraus ermittelt die CPU eine Impulsfolge mit entsprechendem Impuls/Pausenverhältnis und gibt dies über den entsprechenden Ausgabe-Kanal aus.
- ⇒ Der SFC liefert einen Fehlercode zurück. Die entsprechenden Fehlermeldung finden Sie in der Tabelle auf der nächsten Seite. Die PWM-Parameter stehen in folgendem Verhältnis:



$$\text{Periodendauer} = \text{Zeitbasis} \times \text{Periode}$$

$$\text{Impulsdauer} = (\text{Periodendauer} / 1000) \times \text{Tastverhältnis}$$

Impulspause = Periodendauer - Impulsdauer

Die Parameter haben folgende Bedeutung:

- CHANNEL**
- Geben Sie hier den Ausgabe-Kanal an, den Sie ansteuern möchten.
    - Wertebereich: 0 ... 1
- ENABLE**
- Über diesen Parameter können Sie die PWM-Funktion aktivieren (true) bzw. deaktivieren (false).
    - Wertebereich: true, false
- TIMEBASE**
- *TIMEBASE* bedeutet "Zeitbasis" über die Sie die Auflösung und den Wertebereich der Impuls-, Perioden- und Mindestimpulsdauer je Kanal bestimmen.
  - Eingestellt werden können die Werte 0 für 0,1ms und 1 für 1ms.
    - Wertebereich: 0 ... 1
- PERIOD**
- Durch Multiplikation des unter *PERIOD* vorgegebenen Werts mit der *TIMEBASE* erhalten Sie die Periodendauer.
    - Wertebereich: 0 ... 60000
- DUTY**
- Mit diesem Parameter geben Sie das Tastverhältnis in Promille an. Hiermit bestimmen Sie, bezogen auf eine Periode, das Verhältnis zwischen Impulsdauer und Impulspause.
    - 1 Promille = 1 *TIMEBASE*
  - Ist die errechnete Impulsdauer kein Vielfaches von *TIMEBASE*, wird auf die nächst kleinere *TIMEBASE*-Grenze abgerundet.
    - Wertebereich: 0 ... 1000
- MINLEN**
- Über *MINLEN* bestimmen Sie die minimale Impulsdauer. Schalthandlungen werden nur dann durchgeführt, wenn der Impuls die hier eingestellte minimale Zeitdauer überschreitet.
    - Wertebereich: 0 ... 60000
- RET\_VAL (Rückgabewert)**
- Über den Parameter *RET\_VAL* bekommen Sie eine Fehler-Nr. zurückgeliefert. 0 bedeutet, dass alles OK ist. Die entsprechende Fehlermeldung entnehmen Sie bitte der nachfolgenden Tabelle:

| Wert  | Bedeutung   |
|-------|---|
| 0000h | Kein Fehler   |
| 8005h | Parameter <i>MINLEN</i> außerhalb der zulässigen Grenzen.                             |
| 8006h | Parameter <i>DUTY</i> außerhalb der zulässigen Grenzen.                               |
| 8007h | Parameter <i>PERIOD</i> außerhalb der zulässigen Grenzen.                             |
| 8008h | Parameter <i>TIMEBASE</i> außerhalb der zulässigen Grenzen.                           |
| 8009h | Parameter <i>CHANNEL</i> außerhalb der zulässigen Grenzen.                            |
| 9001h | Interner Fehler - Für einen Parameter konnte keine gültige Adresse zugeordnet werden. |

| Wert  | Bedeutung  |
|-------|--|
| 9002h | Interner Hardwarefehler - Kontaktieren Sie bitte die Hotline.                                |
| 9003h | Ausgang ist nicht als PWM-Ausgang parametrierbar bzw. Hardware-Konfiguration ist fehlerhaft. |
| 9004h | HF-PWM wurde parametrierbar aber SFC 223 wurde aufgerufen (bitte SFC 225 HF_PWM verwenden!). |

### 5.5.2 SFC 224 - HSC - High-speed-Counter

#### Beschreibung

Dieser SFC dient zur Parametrierung der Zählfunktionen (high speed counter) für die ersten 4 Eingänge.



*Bitte beachten Sie, dass dieser Baustein in der Bibliothek für das Siemens TIA Portal nicht enthalten ist.*

#### Parameter

| Parameter   | Deklaration | Typ  | Beschreibung                     |
|-------------|-------------|------|----------------------------------|
| CHANNEL     | IN          | INT  | Nummer des Eingabekanals für HSC |
| ENABLE      | IN          | BOOL | Auftrag anstoßen                 |
| DIRECTION   | IN          | INT  | Zählrichtung                     |
| PRESETVALUE | IN          | DINT | Ladewert                         |
| LIMIT       | IN          | DINT | Zählgrenze                       |
| RET_VAL     | OUT         | WORD | Rückgabewert (0 = OK)            |
| SETCOUNTER  | IN_OUT      | BOOL | Ladewert laden                   |

#### CHANNEL

- Geben Sie hier den Eingabe-Kanal an, den Sie als Zähler aktivieren möchten.
  - Wertebereich: 0 ... 3

#### ENABLE

- Über diesen Parameter können Sie die Zählerfunktion aktivieren (true) bzw. deaktivieren (false).
  - Wertebereich: true, false

#### DIRECTION

- Bestimmen sie mit *DIRECTION* die Zählrichtung.
  - Hierbei bedeuten:
    - 0: Zähler ist deaktiviert, entspricht *ENABLE* = false
    - 1: hochzählen
    - 2: runterzählen

#### PRESETVALUE

- Hiermit können sie einen Zählerinhalt vorgeben, der über *SETCOUNTER* = true in den entsprechenden Zähler transferiert wird.
  - Wertebereich: 0 ... FFFFFFFFh

- LIMIT**
- Über Limit geben Sie eine obere bzw. untere Grenze an, für die Zählrichtung Auf- bzw. Abwärts. Bei Erreichen der Grenze wird der entsprechende Zähler auf 0 gestellt und neu gestartet; ggf. erfolgt eine Alarmausgabe.
    - Wertebereich: 0 ... FFFFFFFFh

**RET\_VAL (Rückgabewert)** Über den Parameter RET\_VAL bekommen Sie eine Fehler-Nr. zurückgeliefert. Die entsprechende Fehlermeldung entnehmen Sie bitte der nachfolgenden Tabelle:

| Wert  | Bedeutung   |
|-------|---|
| 0000h | Kein Fehler   |
| 8002h | Der ausgewählte Kanal ist nicht als Zähler konfiguriert (Fehler in Hardware-Konfiguration). |
| 8008h | Parameter DIRECTION außerhalb der zulässigen Grenzen.                                       |
| 8009h | Parameter CHANNEL außerhalb der zulässigen Grenzen.   |
| 9001h | Interner Fehler - Für einen Parameter konnte keine gültige Adresse zugeordnet werden.       |
| 9002h | Interner Hardwarefehler - Kontaktieren Sie bitte die Hotline.                               |

- SETCOUNTER**
- Durch *SETCOUNTER* = true wird der mit *PRESETVALUE* übergebene Wert in den entsprechenden Zähler übertragen.
  - Das Bit wird vom SFC wieder zurückgesetzt.
    - Wertebereich: true, false

### 5.5.3 SFC 225 - HF\_PWM - HF Pulsweitenmodulation

#### Beschreibung

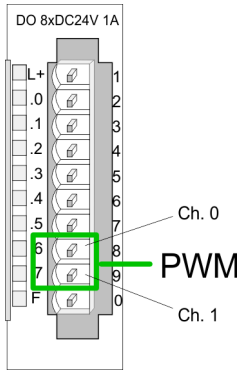
Dieser Baustein dient zur Parametrierung der Pulsweitenmodulation für die letzten beiden Ausgabe-Kanäle. Dieser Baustein hat die gleiche Funktion wie der SFC 223. Anstelle von *TIMEBASE* und *PERIODE* geben Sie hier eine Frequenz (bis zu 50kHz) vor.



*Bitte beachten Sie, dass dieser Baustein in der Bibliothek für das Siemens TIA Portal nicht enthalten ist.*

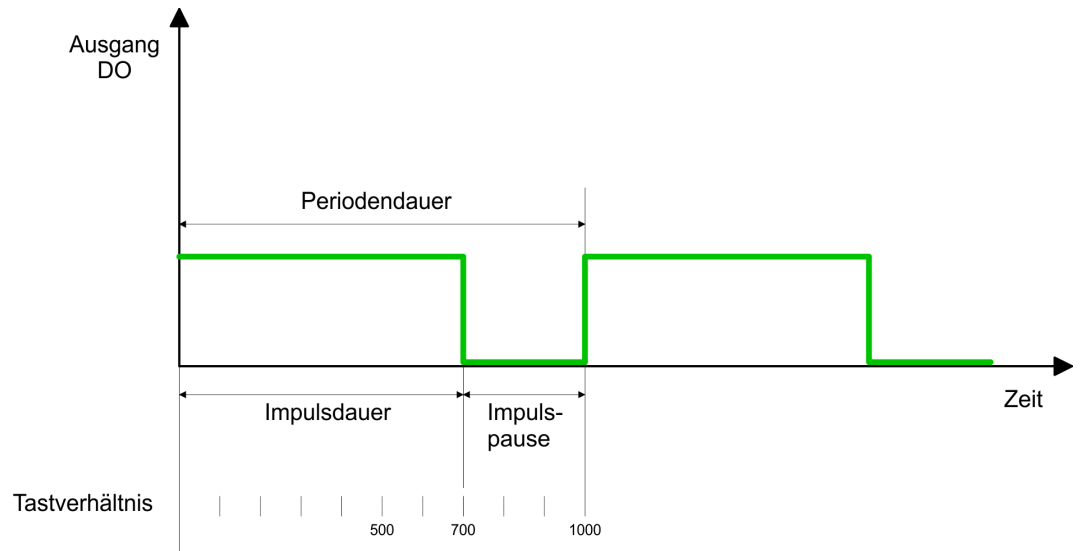
#### Parameter

| Parameter | Deklaration | Typ  | Beschreibung                        |
|-----------|-------------|------|-------------------------------------|
| CHANNEL   | IN          | INT  | Nummer des Ausgabekanals für HF-PWM |
| ENABLE    | IN          | BOOL | Auftrag anstoßen                    |
| FREQUENCE | IN          | WORD | Frequenz der HF-PWM                 |
| DUTY      | IN          | DINT | Tastverhältnis in Promille          |
| MINLEN    | IN          | DINT | Minimale Impulsdauer                |
| RET_VAL   | OUT         | WORD | Rückgabewert (0 = OK)               |



➔ Sie geben Frequenz, Tastverhältnis und minimale Impulsdauer vor. Hieraus ermittelt die CPU eine Impulsfolge mit entsprechendem Impuls/Pausenverhältnis und gibt dies über den entsprechenden Ausgabe-Kanal aus.

⇒ Der SFC liefert einen Fehlercode zurück. Die entsprechenden Fehlermeldung finden Sie in der Tabelle auf der nächsten Seite. Die PWM-Parameter stehen in folgendem Verhältnis:



Periodendauer = 1 / Frequenz

Impulsdauer = (Periodendauer / 1000) x Tastverhältnis

Impulspause = Periodendauer - Impulsdauer

**CHANNEL**

- Geben Sie hier den Ausgabe-Kanal an, den Sie ansteuern möchten.
  - Wertebereich: 0 ... 1

**ENABLE**

- Über diesen Parameter können Sie die PWM-Funktion aktivieren (true) bzw. deaktivieren (false).
  - Wertebereich: true, false

**FREQUENCY**

- Geben Sie hier die Frequenz in Hz als hexadezimalen Wert an.
  - Wertebereich: 09C4h ... C350h (2,5kHz ... 50kHz)

**DUTY**

- Mit diesem Parameter geben Sie das Tastverhältnis in Promille an. Hiermit bestimmen Sie, bezogen auf eine Periode, das Verhältnis zwischen Impulsdauer und Impulspause.
  - 1 Promille = 1 *TIMEBASE*
- Ist die errechnete Impulsdauer kein Vielfaches von *TIMEBASE*, wird auf die nächst kleinere *TIMEBASE*-Grenze abgerundet.
  - Wertebereich: 0 ... 1000

**MINLEN**

- Über *MINLEN* bestimmen Sie die minimale Impulsdauer in  $\mu\text{s}$ . Schalthandlungen werden nur dann durchgeführt, wenn der Impuls die hier eingestellte minimale Zeitdauer überschreitet.
  - Wertebereich: 0 ... 60000

**RET\_VAL (Rückgabewert)**

Über den Parameter *RET\_VAL* bekommen Sie eine Fehler-Nr. zurückgeliefert. 0 bedeutet, dass alles OK ist. Die entsprechende Fehlermeldung entnehmen Sie bitte der nachfolgenden Tabelle:

| Wert  | Bedeutung  |
|-------|--|
| 0000h | Kein Fehler  |
| 8005h | Parameter <i>MINLEN</i> außerhalb der zulässigen Grenzen.                                    |
| 8006h | Parameter <i>DUTY</i> außerhalb der zulässigen Grenzen.                                      |
| 8007h | Parameter <i>FREQUENCE</i> außerhalb der zulässigen Grenzen.                                 |
| 8008h | Parameter <i>TIMEBASE</i> außerhalb der zulässigen Grenzen.                                  |
| 8009h | Parameter <i>CHANNEL</i> außerhalb der zulässigen Grenzen.                                   |
| 9001h | Interner Fehler - Für einen Parameter konnte keine gültige Adresse zugeordnet werden.        |
| 9002h | Interner Hardwarefehler - Kontaktieren Sie bitte die Hotline.                                |
| 9003h | Ausgang ist nicht als PWM-Ausgang parametrierbar bzw. Hardware-Konfiguration ist fehlerhaft. |
| 9004h | HF-PWM wurde parametrierbar aber SFC 223 wurde aufgerufen (bitte SFC 225 HF_PWM verwenden!). |