

# VIPA System 300S

## CPU-SC | 312-5BE13 | Handbuch

HB140 | CPU-SC | 312-5BE13 | DE | 15-50

SPEED7 CPU 312SC

VIPA GmbH  
Ohmstr. 4  
91074 Herzogenaurach  
Telefon: +49 9132 744-0  
Telefax: +49 9132 744-1864  
E-Mail: [info@vipa.com](mailto:info@vipa.com)  
Internet: [www.vipa.com](http://www.vipa.com)

## Inhaltsverzeichnis

<b>1</b>	<b>Allgemeines</b> .....	<b>6</b>
	1.1 Copyright © VIPA GmbH .....	6
	1.2 Über dieses Handbuch.....	7
	1.3 Sicherheitshinweise.....	8
<b>2</b>	<b>Grundlagen</b> .....	<b>10</b>
	2.1 Sicherheitshinweis für den Benutzer.....	10
	2.2 Arbeitsweise einer CPU.....	11
	2.2.1 Allgemein.....	11
	2.2.2 Programme .....	11
	2.2.3 Operanden.....	12
	2.3 CPU 312-5BE13.....	13
	2.4 Allgemeine Daten.....	14
<b>3</b>	<b>Montage und Aufbaurichtlinien</b> .....	<b>16</b>
	3.1 Einbaumaße.....	16
	3.2 Montage Standard-Bus.....	17
	3.3 Verdrahtung.....	18
	3.4 Aufbaurichtlinien.....	21
<b>4</b>	<b>Hardwarebeschreibung</b> .....	<b>25</b>
	4.1 Leistungsmerkmale.....	25
	4.2 Aufbau.....	26
	4.2.1 Allgemein.....	26
	4.2.2 Schnittstellen.....	27
	4.2.3 Ein-/Ausgabe-Bereich CPU 312-5BE13.....	29
	4.2.4 Speichermanagement.....	32
	4.2.5 Steckplatz für Speichermedien.....	32
	4.2.6 Batteriepufferung für Uhr und RAM.....	33
	4.2.7 Betriebsartenschalter.....	33
	4.2.8 LEDs.....	34
	4.3 Technische Daten.....	35
<b>5</b>	<b>Einsatz CPU 312-5BE13</b> .....	<b>47</b>
	5.1 Montage.....	47
	5.2 Anlaufverhalten.....	47
	5.3 Adressierung.....	48
	5.3.1 Übersicht.....	48
	5.3.2 Adressierung Rückwandbus Peripherie.....	48
	5.4 Adressbelegung.....	49
	5.5 Hardware-Konfiguration - CPU.....	50
	5.6 Hardware-Konfiguration - I/O-Module.....	51
	5.7 Hardware-Konfiguration - Ethernet-PG/OP-Kanal.....	51
	5.8 Einstellung der CPU-Parameter.....	53
	5.8.1 Parametrierung über Siemens CPU.....	53
	5.8.2 Parameter CPU.....	54
	5.9 Projekt transferieren.....	56
	5.9.1 Transfer über MPI.....	56
	5.9.2 Transfer über Ethernet.....	57
	5.9.3 Transfer über MMC.....	58
	5.10 Zugriff auf integrierte Web-Seite.....	59

5.11	Betriebszustände.....	61
5.11.1	Übersicht.....	61
5.11.2	Funktionssicherheit.....	63
5.12	Urlöschen.....	63
5.13	Firmwareupdate.....	65
5.14	Rücksetzen auf Werkseinstellung.....	68
5.15	Steckplatz für Speichermedien.....	69
5.16	Speichererweiterung mit MCC.....	70
5.17	Erweiterter Know-how-Schutz.....	71
5.18	MMC-Cmd - Autobefehle.....	73
5.19	VIPA-spezifische Diagnose-Einträge.....	75
5.20	Mit Testfunktionen Variablen steuern und beobachten..	91
<b>6</b>	<b>Einsatz E/A-Peripherie.....</b>	<b>93</b>
6.1	Übersicht.....	93
6.2	Ein-/Ausgabe-Bereich CPU 312-5BE13.....	94
6.3	Adressbelegung.....	97
6.4	Digital-Teil.....	98
6.4.1	Zugriff auf den E/A Bereich.....	100
6.4.2	Parametrierung - Digital-Teil.....	101
6.5	Zähler.....	101
6.5.1	Zähler - Schnelleinstieg.....	101
6.5.2	SFB 47 - COUNT - Zähler steuern.....	107
6.5.3	Zähler - Funktionen.....	112
6.5.4	Zähler - Zusatzfunktionen.....	117
6.6	Frequenzmessung.....	125
6.6.1	Übersicht.....	125
6.6.2	Eingänge für die Frequenzmessung.....	126
6.6.3	Parametriervorgang.....	127
6.6.4	SFB 48 - FREQUENC - Frequenzmessung steuern...	128
6.7	Pulsweitenmodulation - PWM.....	131
6.7.1	Übersicht.....	131
6.7.2	Parametriervorgang.....	132
6.7.3	SFB 49 - PULSE - Pulsweitenmodulation.....	134
6.8	Diagnose und Alarm.....	138
6.8.1	Prozessalarm.....	138
6.8.2	Diagnosealarm.....	140
<b>7</b>	<b>Einsatz PtP-Kommunikation.....</b>	<b>147</b>
7.1	Schnelleinstieg.....	147
7.2	Prinzip der Datenübertragung.....	147
7.3	Einsatz der RS485-Schnittstelle für PtP.....	148
7.4	Parametrierung.....	149
7.4.1	FC/SFC 216 - SER_CFG.....	149
7.5	Kommunikation.....	153
7.5.1	Übersicht.....	153
7.5.2	FC/SFC 217 - SER_SND.....	153
7.5.3	FC/SFC 218 - SER_RCV.....	158
7.6	Protokolle und Prozeduren .....	160
7.7	Modbus - Funktionscodes .....	164
7.8	Modbus - Beispiel zur Kommunikation.....	169

<b>8</b>	<b>WinPLC7</b> .....	<b>172</b>
	8.1 Systemvorstellung.....	172
	8.2 Installation.....	172
	8.3 Beispiel zur Projektierung.....	174
	8.3.1 Aufgabenstellung.....	174
	8.3.2 Projektierung.....	174
	8.3.3 SPS-Programm in <i>Simulator</i> testen.....	181
	8.3.4 SPS-Programm in CPU übertragen und ausführen....	182
<b>9</b>	<b>Projektierung im TIA Portal</b> .....	<b>184</b>
	9.1 TIA Portal - Arbeitsumgebung .....	184
	9.1.1 Allgemein.....	184
	9.1.2 Arbeitsumgebung des TIA Portals.....	184
	9.2 TIA Portal - Hardware-Konfiguration - CPU .....	185
	9.3 TIA Portal - Hardware-Konfiguration - I/O-Module.....	186
	9.4 TIA Portal - Hardware-Konfiguration - Ethernet-PG/OP- Kanal.....	187
	9.5 TIA Portal - VIPA-Bibliothek einbinden.....	190
	9.6 TIA Portal - Projekt transferieren.....	191

# 1 Allgemeines

## 1.1 Copyright © VIPA GmbH

### All Rights Reserved

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

Dieses Material ist durch Urheberrechtsgesetze geschützt. Ohne schriftliches Einverständnis von VIPA und dem Besitzer dieses Materials darf dieses Material weder reproduziert, verteilt, noch in keiner Form von keiner Einheit (sowohl VIPA-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: VIPA, Gesellschaft für Visualisierung und Prozessautomatisierung mbH Ohmstraße 4, D-91074 Herzogenaurach, Germany

Tel.: +49 9132 744 -0

Fax.: +49 9132 744-1864

E-Mail: [info@vipa.de](mailto:info@vipa.de)

<http://www.vipa.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 VIPA 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 VIPA GmbH.

**Warenzeichen**

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S und Commander Compact sind eingetragene Warenzeichen der VIPA Gesellschaft für Visualisierung und Prozessautomatisierung mbH.

SPEED7 ist ein eingetragenes Warenzeichen der profichip GmbH.

SIMATIC, STEP, SINEC, TIA Portal, S7-300 und S7-400 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 VIPA GmbH, wenn Sie Fehler anzeigen oder inhaltliche Fragen zu diesem Dokument stellen möchten. Ist eine solche Stelle nicht erreichbar, können Sie VIPA über folgenden Kontakt erreichen:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefax: +49 9132 744-1204

E-Mail: [documentation@vipa.de](mailto:documentation@vipa.de)

**Technischer Support**

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

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefon: +49 9132 744-1150 (Hotline)

E-Mail: [support@vipa.de](mailto:support@vipa.de)

**1.2 Über dieses Handbuch**

**Zielsetzung und Inhalt**

Das Handbuch beschreibt die SPEED7 CPU-SC 312-5BE13 aus dem System 300S von VIPA. Beschrieben wird Aufbau, Projektierung und Anwendung.

Produkt	Best.-Nr.	ab Stand:	
		CPU-HW	CPU-FW
CPU 312SC	312-5BE13	02	V3.6.0

**Zielgruppe**

Das Handbuch ist geschrieben für Anwender mit Grundkenntnissen in der Automatisierungstechnik.

**Aufbau des Handbuchs**

Das Handbuch ist in Kapitel gegliedert. Jedes Kapitel beschreibt eine abgeschlossene Thematik.

**Orientierung im Dokument**

Als Orientierungshilfe stehen im Handbuch zur Verfügung:

- Gesamt-Inhaltsverzeichnis am Anfang des Handbuchs
- Verweise mit Seitenangabe

**Verfügbarkeit**

Das Handbuch ist verfügbar in:

- gedruckter Form auf Papier
- in elektronischer Form als PDF-Datei (Adobe Acrobat Reader)

**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*

### 1.3 Sicherheitshinweise

**Bestimmungsgemäße Verwendung**

Das System ist konstruiert und gefertigt für:

- Kommunikation und Prozesskontrolle
- Allgemeine Steuerungs- und Automatisierungsaufgaben
- den industriellen Einsatz
- den Betrieb innerhalb der in den technischen Daten spezifizierten Umgebungsbedingungen
- den Einbau in einen Schaltschrank

**GEFAHR!**

Das Gerät ist nicht zugelassen für den Einsatz  
– in explosionsgefährdeten Umgebungen (EX-Zone)

**Dokumentation**

Handbuch zugänglich machen für alle Mitarbeiter in

- Projektierung
- Installation
- Inbetriebnahme
- Betrieb

**VORSICHT!**

**Vor Inbetriebnahme und Betrieb der in diesem Handbuch beschriebenen Komponenten unbedingt beachten:**

- Änderungen am Automatisierungssystem nur im spannungslosen Zustand vornehmen!
- Anschluss und Änderung nur durch ausgebildetes Elektro-Fachpersonal
- Nationale Vorschriften und Richtlinien im jeweiligen Verwenderland beachten und einhalten (Installation, Schutzmaßnahmen, EMV ...)

**Entsorgung**

**Zur Entsorgung des Geräts nationale Vorschriften beachten!**

## 2 Grundlagen

### 2.1 Sicherheitshinweis für den Benutzer

#### Handhabung elektrostatisch gefährdeter Baugruppen

VIPA-Baugruppen sind mit hochintegrierten Bauelementen in MOS-Technik bestückt. Diese Bauelemente sind hoch empfindlich gegenüber Überspannungen, die z.B. bei elektrostatischer Entladung entstehen. Zur Kennzeichnung dieser gefährdeten Baugruppen wird nachfolgendes Symbol verwendet:



Das Symbol befindet sich auf Baugruppen, Baugruppenträgern oder auf Verpackungen und weist so auf elektrostatisch gefährdete Baugruppen hin. Elektrostatisch gefährdete Baugruppen können durch Energien und Spannungen zerstört werden, die weit unterhalb der Wahrnehmungsgrenze des Menschen liegen. Hantiert eine Person, die nicht elektrisch entladen ist, mit elektrostatisch gefährdeten Baugruppen, können Spannungen auftreten und zur Beschädigung von Bauelementen führen und so die Funktionsweise der Baugruppen beeinträchtigen oder die Baugruppe unbrauchbar machen. Auf diese Weise beschädigte Baugruppen werden in den wenigsten Fällen sofort als fehlerhaft erkannt. Der Fehler kann sich erst nach längerem Betrieb einstellen. Durch statische Entladung beschädigte Bauelemente können bei Temperaturänderungen, Erschütterungen oder Lastwechseln zeitweilige Fehler zeigen. Nur durch konsequente Anwendung von Schutzeinrichtungen und verantwortungsbewusste Beachtung der Handhabungsregeln lassen sich Funktionsstörungen und Ausfälle an elektrostatisch gefährdeten Baugruppen wirksam vermeiden.

#### Versenden von Baugruppen

Verwenden Sie für den Versand immer die Originalverpackung.

#### Messen und Ändern von elektrostatisch gefährdeten Baugruppen

Bei Messungen an elektrostatisch gefährdeten Baugruppen sind folgende Dinge zu beachten:

- Potenzialfreie Messgeräte sind kurzzeitig zu entladen.
- Verwendete Messgeräte sind zu erden.

Bei Änderungen an elektrostatisch gefährdeten Baugruppen ist darauf zu achten, dass ein geerdeter LötKolben verwendet wird.



#### **VORSICHT!**

Bei Arbeiten mit und an elektrostatisch gefährdeten Baugruppen ist auf ausreichende Erdung des Menschen und der Arbeitsmittel zu achten.

## 2.2 Arbeitsweise einer CPU

### 2.2.1 Allgemein

Die CPU enthält einen Standardprozessor mit internem Programmspeicher. In Verbindung mit der integrierten SPEED7-Technologie erhalten Sie ein leistungsfähiges Gerät zur Prozessautomatisierung innerhalb der System 300S Familie. In einer CPU gibt es folgende Arbeitsweisen:

- zyklische Bearbeitung
- zeitgesteuerte Bearbeitung
- alarmgesteuerte Bearbeitung
- Bearbeitung nach Priorität

#### Zyklische Bearbeitung

Die **zyklische** Bearbeitung stellt den Hauptanteil aller Vorgänge in der CPU. In einem endlosen Zyklus werden die gleichen Bearbeitungsfolgen wiederholt.

#### Zeitgesteuerte Bearbeitung

Erfordern Prozesse in konstanten Zeitabschnitten Steuersignale, so können Sie neben dem zyklischen Ablauf **zeitgesteuert** bestimmte Aufgaben durchführen z.B. zeitunkritische Überwachungsfunktionen im Sekundenraster.

#### Alarmgesteuerte Bearbeitung

Soll auf ein Prozesssignal besonders schnell reagiert werden, so ordnen Sie diesem einen **alarmgesteuerten** Bearbeitungsabschnitt zu. Ein Alarm kann in Ihrem Programm eine Bearbeitungsfolge aktivieren.

#### Bearbeitung nach Priorität

Die oben genannten Bearbeitungsarten werden von der CPU nach Wichtigkeitsgrad behandelt (**Priorität**). Da auf ein Zeit- oder Alarmergebnis schnell reagiert werden muss, unterbricht die CPU zur Bearbeitung dieser hochpriorären Ereignisse die zyklische Bearbeitung, reagiert auf diese Ereignisse und setzt danach die zyklische Bearbeitung wieder fort. Die zyklische Bearbeitung hat daher die niedrigste Priorität.

### 2.2.2 Programme

Das in jeder CPU vorhandene Programm unterteilt sich in:

- Systemprogramm
- Anwenderprogramm

#### Systemprogramm

Das Systemprogramm organisiert alle Funktionen und Abläufe der CPU, die nicht mit einer spezifischen Steuerungsaufgabe verbunden sind.

#### Anwenderprogramm

Hier finden Sie alle Funktionen, die zur Bearbeitung einer spezifischen Steuerungsaufgabe erforderlich sind. Schnittstellen zum Systemprogramm stellen die Operationsbausteine zur Verfügung.

### 2.2.3 Operanden

Die CPU stellt Ihnen für das Programmieren folgende Operandenbereiche zur Verfügung:

- Prozessabbild und Peripherie
- Merker
- Zeiten und Zähler
- Datenbausteine

#### Prozessabbild und Peripherie

Auf das Prozessabbild der Aus- und Eingänge PAA/PAE kann Ihr Anwenderprogramm sehr schnell zugreifen. Sie haben Zugriff auf folgende Datentypen:

- Einzelbits
- Bytes
- Wörter
- Doppelwörter

Sie können mit Ihrem Anwenderprogramm über den Bus direkt auf Peripheriebaugruppen zugreifen. Folgende Datentypen sind möglich:

- Bytes
- Wörter
- Blöcke

#### Merker

Der Merkerbereich ist ein Speicherbereich, auf den Sie über Ihr Anwenderprogramm mit entsprechenden Operationen zugreifen können. Verwenden Sie den Merkerbereich für oft benötigte Arbeitsdaten.

Sie können auf folgende Datentypen zugreifen:

- Einzelbits
- Bytes
- Wörter
- Doppelwörter

#### Zeiten und Zähler

Sie können mit Ihrem Anwendungsprogramm eine Zeitzelle mit einem Wert zwischen 10ms und 9990s laden. Sobald Ihr Anwenderprogramm eine Startoperation ausführt, wird dieser Zeitwert um ein durch Sie vorgegebenes Zeitraster dekrementiert, bis Null erreicht wird.

Für den Einsatz von Zählern können Sie Zählerzellen mit einem Anfangswert laden (max. 999) und diesen hinauf- bzw. herunterzählen.

#### Datenbausteine

Ein Datenbaustein enthält Konstanten bzw. Variablen im Byte-, Wort- oder Doppelwortformat. Mit Operanden können Sie immer auf den aktuellen Datenbaustein zugreifen.

Sie haben Zugriff auf folgende Datentypen:

- Einzelbits
- Bytes
- Wörter
- Doppelwörter

## 2.3 CPU 312-5BE13

### Übersicht

Die CPU 312-5BE13 basiert auf der SPEED7-Technologie. Hierbei wird die CPU durch Coprozessoren im Bereich Programmierung und Kommunikation unterstützt und erhält somit eine Leistungssteigerung, so dass diese höchsten Anforderungen genügt.

- Programmiert wird die CPU in STEP®7 von Siemens. Hierzu können Sie den SIMATIC Manager von Siemens verwenden. Hierbei kommt der Befehlssatz der S7-400 von Siemens zum Einsatz.
- Module und CPUs aus dem System 300S von VIPA und Siemens können als Mischkonfiguration am Bus eingesetzt werden.
- Das Anwenderprogramm wird im batteriegepufferten RAM oder auf einem zusätzlich steckbaren MMC-Speichermodul gespeichert.
- Projektiert wird die CPU 312-5BE13 als CPU 312C (6ES7 312-5BE03-0AB0/V2.6) von Siemens.

### Speicher

Die CPU hat einen Speicher integriert. Angaben über die Speicherkapazität finden Sie auf der Frontseite Ihrer CPU. Der Speicher gliedert sich in folgende Teile:

- Ladespeicher 512kByte
- Codespeicher (50% des Arbeitsspeichers)
- Datenspeicher (50% des Arbeitsspeichers)
- Arbeitsspeicher 64kByte
  - Sie haben die Möglichkeit den Arbeitsspeicher mittels einer MCC Speichererweiterungskarte bis zur maximal aufgedruckten Kapazität 512kByte zu erweitern.

### Integrierte PtP-Funktionalität

Die CPU besitzt eine RS485-Schnittstelle, welche fix auf PtP-Kommunikation (**point to point**) eingestellt ist. Dies ermöglicht die serielle Prozessankopplung zu verschiedenen Ziel- oder Quellsystemen.

### Integrierter Ethernet-PG/OP-Kanal

Auf der CPU befindet sich eine Ethernet-Schnittstelle für PG/OP-Kommunikation. Nach der Zuweisung von IP-Adress-Parametern über Ihr Projektier-Tool können Sie über die "Zielsystem"-Funktionen den Ethernet-PG/OP-Kanal direkt ansprechen und Ihre CPU programmieren bzw. fernwarten. Sie haben auch die Möglichkeit über diese Verbindungen mit einer Visualisierungs-Software auf die CPU zuzugreifen.

### Betriebssicherheit

- Anschluss über Federzugklemmen an Frontstecker
- Aderquerschnitt 0,08...2,5mm<sup>2</sup>
- Vollisolierung der Verdrahtung bei Modulwechsel
- Potenzialtrennung aller Peripherie-Module zum Rückwandbus

### Aufbau/Maße

Maße Grundgehäuse:

- 2fach breit: (BxHxT) in mm: 80x125x120

**Integriertes Netzteil**

Die CPU hat ein Netzteil integriert. Das Netzteil ist mit DC 24V zu versorgen. Über die Versorgungsspannung werden neben der internen Elektronik auch die angeschlossenen Module über den Rückwandbus versorgt. Das Netzteil ist gegen Verpolung und Überstrom geschützt.

**2.4 Allgemeine Daten**

Konformität und Approbation		
Konformität		
CE	2006/95/EG	Niederspannungsrichtlinie
	2004/108/EG	EMV-Richtlinie
Approbation		
UL	UL 508	Zulassung für USA und Kanada
Sonstiges		
RoHS	2011/65/EU	Produkte bleifrei; Richtlinie zur Beschränkung der Verwendung bestimmter gefährlicher Stoffe in Elektro- und Elektronikgeräten

Personenschutz und Geräteschutz		
Schutzart	-	IP20
Potenzialtrennung		
Zum Feldbus	-	Galvanisch entkoppelt
Zur Prozessebene	-	Galvanisch entkoppelt
Isolationsfestigkeit		-
Isolationsspannung gegen Bezugserde		
Eingänge / Ausgänge	-	AC / DC 50V, bei Prüfspannung AC 500V
Schutzmaßnahmen	-	gegen Kurzschluss

Umgebungsbedingungen gemäß EN 61131-2		
Klimatisch		
Lagerung /Transport	EN 60068-2-14	-25...+70°C
Betrieb		
Horizontaler Einbau hängend	EN 61131-2	0...+60°C
Horizontaler Einbau liegend	EN 61131-2	0...+55°C
Vertikaler Einbau	EN 61131-2	0...+50°C
Luftfeuchtigkeit	EN 60068-2-30	RH1 (ohne Betauung, relative Feuchte 10 ... 95%)
Verschmutzung	EN 61131-2	Verschmutzungsgrad 2
Aufstellhöhe max.	-	2000m

**Umgebungsbedingungen gemäß EN 61131-2**

Mechanisch		
Schwingung	EN 60068-2-6	1g, 9Hz ... 150Hz
Schock	EN 60068-2-27	15g, 11ms

**Montagebedingungen**

Einbauort	-	Im Schaltschrank
Einbaulage	-	Horizontal und vertikal

EMV	Norm	Bemerkungen	
Störaussendung	EN 61000-6-4	Class A (Industriebereich)	
Störfestigkeit Zone B	EN 61000-6-2	Industriebereich	
		EN 61000-4-2	ESD 8kV bei Luftentladung (Schärfegrad 3), 4kV bei Kontaktentladung (Schärfegrad 2)
		EN 61000-4-3	HF-Einstrahlung (Gehäuse) 80MHz ... 1000MHz, 10V/m, 80% AM (1kHz) 1,4GHz ... 2,0GHz, 3V/m, 80% AM (1kHz) 2GHz ... 2,7GHz, 1V/m, 80% AM (1kHz)
		EN 61000-4-6	HF-Leitungsgeführt 150kHz ... 80MHz, 10V, 80% AM (1kHz)
		EN 61000-4-4	Burst, Schärfegrad 3
	EN 61000-4-5	Surge, Installationsklasse 3 *	

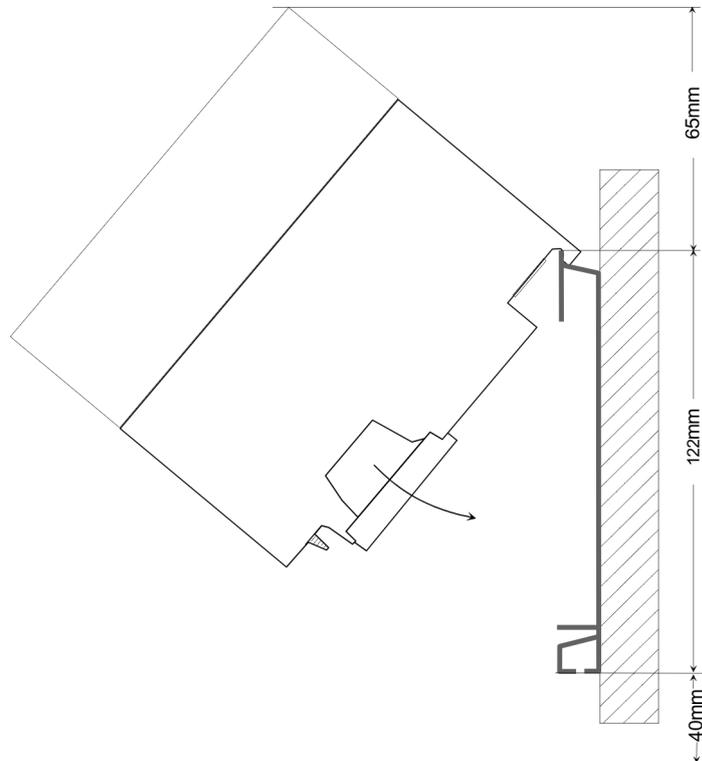
\*) Aufgrund der energiereichen Einzelimpulse ist bei Surge eine angemessene externe Beschaltung mit Blitzschutzelementen wie z.B. Blitzstromableitern und Überspannungsableitern erforderlich.

### 3 Montage und Aufbaurichtlinien

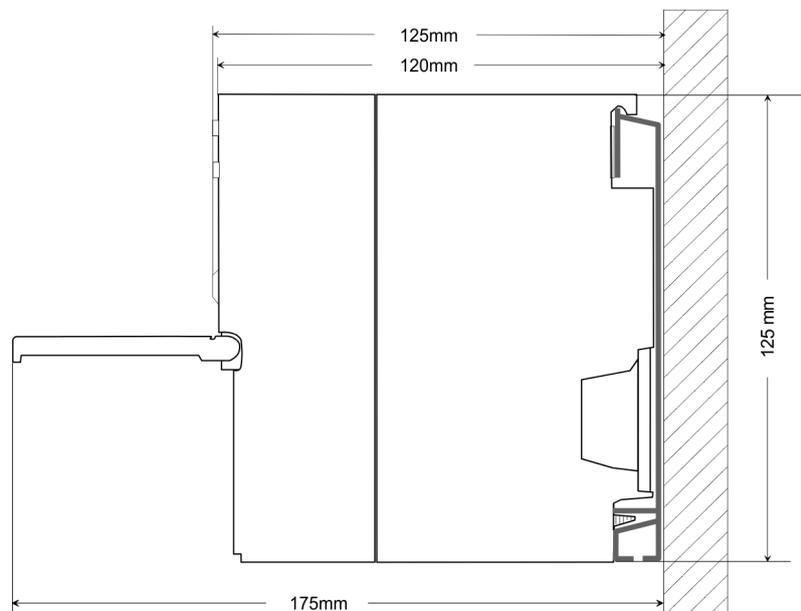
#### 3.1 Einbaumaße

**Maße Grundgehäuse** 2fach breit (BxHxT) in mm: 80 x 125 x 120

#### Montagemaße



#### Maße montiert



### 3.2 Montage Standard-Bus

#### Allgemein

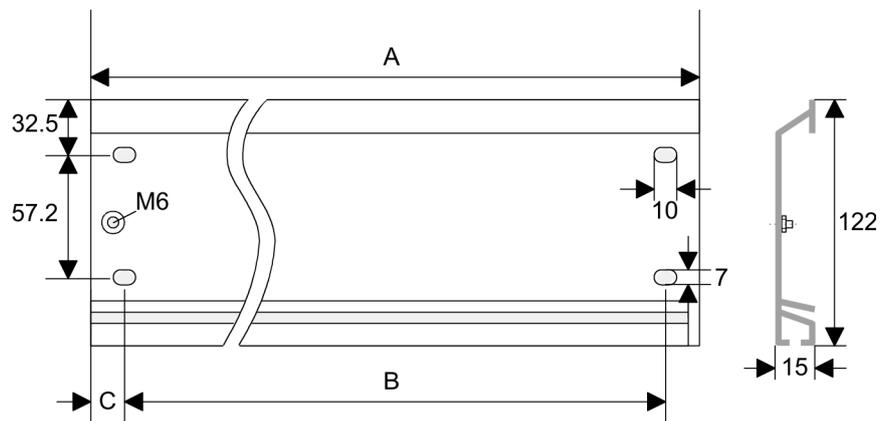
Die einzelnen Module werden direkt auf eine Profilschiene montiert und über den Rückwandbus-Verbinder verbunden. Vor der Montage ist der Rückwandbus-Verbinder von hinten an das Modul zu stecken. Die Rückwandbus-Verbinder sind im Lieferumfang der Peripherie-Module enthalten.

#### Profilschiene

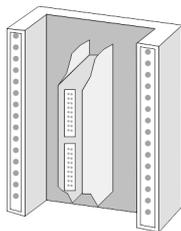
Bestellnummer	A	B	C
390-1AB60	160	140	10
390-1AE80	482	466	8,3
390-1AF30	530	500	15
390-1AJ30	830	800	15
390-9BC00*	2000	Bohrungen nur links	15

\*) Verpackungseinheit 10 Stück

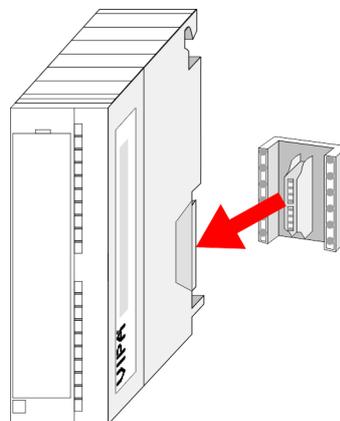
Maße in mm



#### Busverbinder



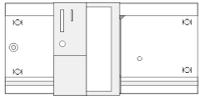
Für die Kommunikation der Module untereinander wird beim System 300S ein Rückwandbus-Verbinder eingesetzt. Die Rückwandbus-Verbinder sind im Lieferumfang der Peripherie-Module enthalten und werden vor der Montage von hinten an das Modul gesteckt.



## Verdrahtung

## Montagemöglichkeiten

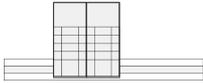
waagrechter Aufbau



senkrechter Aufbau



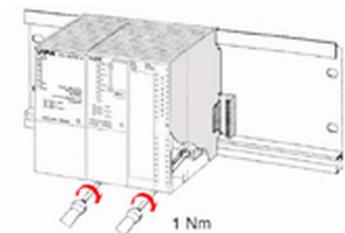
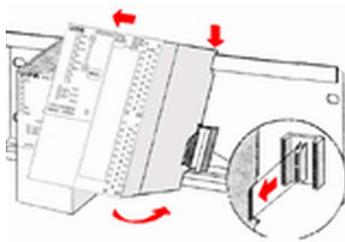
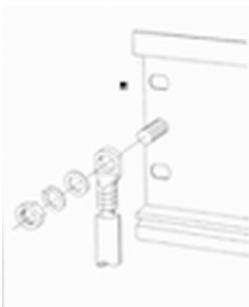
liegender Aufbau



Beachten Sie bitte die hierbei zulässigen Umgebungstemperaturen:

- waagrechter Aufbau: von 0 bis 60°C
- senkrechter Aufbau: von 0 bis 50°C
- liegender Aufbau: von 0 bis 55°C

## Vorgehensweise



1. ▶ Verschrauben Sie die Profilschiene mit dem Untergrund (Schraubengröße: M6) so, dass mindestens 65mm Raum oberhalb und 40mm unterhalb der Profilschiene bleibt.
2. ▶ Achten Sie bei geerdetem Untergrund auf eine niederohmige Verbindung zwischen Profilschiene und Untergrund.
3. ▶ Verbinden Sie die Profilschiene mit dem Schutzleiter. Für diesen Zweck befindet sich auf der Profilschiene ein Stehbolzen mit M6-Gewinde.
4. ▶ Der Mindestquerschnitt der Leitung zum Schutzleiter muss 10mm<sup>2</sup> betragen.
5. ▶ Hängen Sie die Spannungsversorgung ein und schieben Sie diese nach links bis an den Erdungsbolzen der Profilschiene.
6. ▶ Schrauben Sie die Spannungsversorgung fest.
7. ▶ Nehmen Sie einen Rückwandbus-Verbinder und stecken Sie ihn wie gezeigt von hinten an die CPU.
8. ▶ Hängen Sie die CPU rechts von der Spannungsversorgung ein und schieben Sie diese bis an die Spannungsversorgung.
9. ▶ Klappen Sie die CPU nach unten und schrauben Sie die CPU wie gezeigt fest.
10. ▶ Verfahren Sie auf die gleiche Weise mit Ihren Peripheriemodulen, indem Sie jeweils einen Rückwandbus-Verbinder stecken, Ihr Modul rechts neben dem Vorgänger-Modul einhängen, dieses nach unten klappen, in den Rückwandbus-Verbinder des Vorgängermoduls einrasten lassen und das Modul festschrauben.

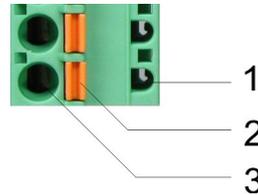
## 3.3 Verdrahtung

**VORSICHT!**

- Die Spannungsversorgungen sind vor dem Beginn von Installations- und Instandhaltungsarbeiten unbedingt freizuschalten, d.h. vor Arbeiten an einer Spannungsversorgung oder an der Zuleitung, ist die Spannungszuführung stromlos zu schalten (Stecker ziehen, bei Festanschluss ist die zugehörige Sicherung abzuschalten)!
- Anschluss und Änderungen dürfen nur durch ausgebildetes Elektro-Fachpersonal ausgeführt werden.

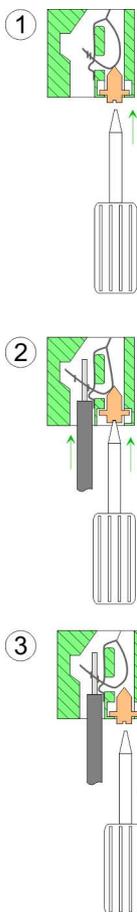
**Federklemmtechnik (grün)**

Zur Verdrahtung der Spannungsversorgung der CPU kommt eine grüne Anschlussklemmen mit Federzugklemmtechnik zum Einsatz. Die Anschlussklemme ist als Stecker ausgeführt, der im verdrahteten Zustand vorsichtig abgezogen werden kann. Hier können Sie Drähte mit einem Querschnitt von 0,08mm<sup>2</sup> bis 2,5mm<sup>2</sup> anschließen. Hierbei dürfen sowohl flexible Litzen ohne Aderendhülse, als auch starre Leiter verwendet werden.



- 1 Prüfabgriff für 2mm Messspitze
- 2 Verriegelung (orange) für Schraubendreher
- 3 Runde Öffnung für Drähte

Die nebenstehende Abfolge stellt die Schritte der Verdrahtung in der Draufsicht dar.



- 1. ➤ Zum Verdrahten drücken Sie mit einem geeigneten Schraubendreher, wie in der Abbildung gezeigt, die Verriegelung senkrecht nach innen und halten Sie den Schraubendreher in dieser Position.
- 2. ➤ Führen Sie durch die runde Öffnung Ihren abisolierten Draht ein. Sie können Drähte mit einem Querschnitt von 0,08mm<sup>2</sup> bis 2,5mm<sup>2</sup> anschließen.
- 3. ➤ Durch Entfernen des Schraubendrehers wird der Draht über einen Federkontakt sicher mit dem Steckverbinder verbunden.

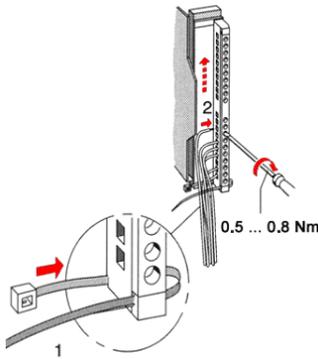
**Frontstecker der Ein-/Ausgabe-Module**

Nachfolgend ist die Verdrahtung der 2 Frontstecker-Varianten aufgezeigt.

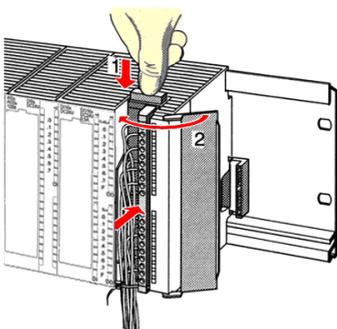
## 20-fach Schraubtechnik 392-1AJ00



1. ▶ Öffnen Sie die Frontklappe Ihres Ein-/Ausgabe-Moduls.
2. ▶ Bringen Sie den Frontstecker in Verdrahtungsstellung.  
Hierzu stecken Sie den Frontstecker auf das Modul, bis er einrastet. In dieser Stellung ragt der Frontstecker aus dem Modul heraus und hat noch keinen Kontakt.
3. ▶ Isolieren Sie Ihre Leitungen ab. Verwenden Sie ggf. Aderendhülsen.
4. ▶ Fädeln Sie den beiliegenden Kabelbinder in den Frontstecker ein.
5. ▶ Beginnen Sie mit der Verdrahtung von unten nach oben, wenn Sie die Leitungen nach unten aus dem Modul herausführen möchten, bzw. von oben nach unten, wenn die Leitungen nach oben herausgeführt werden sollen.
6. ▶ Schrauben Sie die Anschlusschrauben der nicht verdrahteten Schraubklemmen ebenfalls fest.

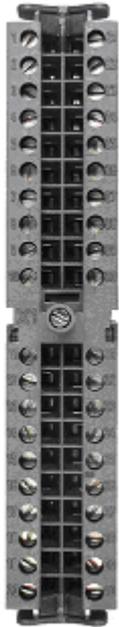


7. ▶ Ziehen Sie den Kabelbinder für den Leitungsstrang fest.

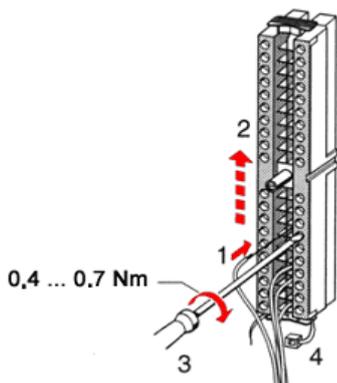


8. ▶ Drücken Sie die Entriegelungstaste am Frontstecker an der Moduloberseite und drücken Sie gleichzeitig den Frontstecker in das Modul, bis er einrastet.
9. ▶ Der Frontstecker ist nun elektrisch mit Ihrem Modul verbunden.
10. ▶ Schließen Sie die Frontklappe.
11. ▶ Füllen Sie den Beschriftungsstreifen zur Kennzeichnung der einzelnen Kanäle aus und schieben Sie den Streifen in die Frontklappe.

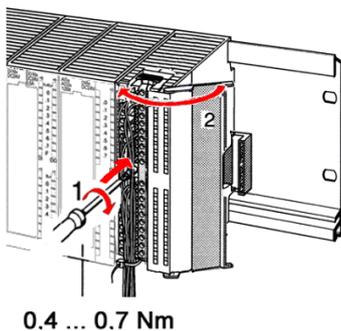
### 40-fach Schraubtechnik 392-1AM00



1. ▶ Öffnen Sie die Frontklappe Ihres Ein-/Ausgabe-Moduls.
2. ▶ Bringen Sie den Frontstecker in Verdrahtungsstellung.  
Hierzu stecken Sie den Frontstecker auf das Modul, bis er einrastet. In dieser Stellung ragt der Frontstecker aus dem Modul heraus und hat noch keinen Kontakt.
3. ▶ Isolieren Sie Ihre Leitungen ab. Verwenden Sie ggf. Aderendhülsen.
4. ▶ Beginnen Sie mit der Verdrahtung von unten nach oben, wenn Sie die Leitungen nach unten aus dem Modul herausführen möchten, bzw. von oben nach unten, wenn die Leitungen nach oben herausgeführt werden sollen.
5. ▶ Schrauben Sie die Anschlusschrauben der nicht verdrahteten Schraubklemmen ebenfalls fest.



6. ▶ Legen Sie den beigegefügteten Kabelbinder um den Leitungsstrang und den Frontstecker herum.
7. ▶ Ziehen Sie den Kabelbinder für den Leitungsstrang fest.



8. ▶ Schrauben Sie die Befestigungsschraube für den Frontstecker fest.
9. ▶ Der Frontstecker ist nun elektrisch mit Ihrem Modul verbunden.
10. ▶ Schließen Sie die Frontklappe.
11. ▶ Füllen Sie den Beschriftungsstreifen zur Kennzeichnung der einzelnen Kanäle aus und schieben Sie den Streifen in die Frontklappe.

## 3.4 Aufbaurichtlinien

### Allgemeines

Die Aufbaurichtlinien enthalten Informationen über den störsicheren Aufbau eines SPS-Systems. Es werden die Wege beschrieben, wie Störungen in Ihre Steuerung gelangen können, wie die elektromagnetische Verträglichkeit (EMV) sicher gestellt werden kann und wie bei der Schirmung vorzugehen ist.

**Was bedeutet EMV?**

Unter Elektromagnetischer Verträglichkeit (EMV) versteht man die Fähigkeit eines elektrischen Gerätes, in einer vorgegebenen elektromagnetischen Umgebung fehlerfrei zu funktionieren, ohne vom Umfeld beeinflusst zu werden bzw. das Umfeld in unzulässiger Weise zu beeinflussen.

Die Komponenten von VIPA sind für den Einsatz in Industrieumgebungen entwickelt und erfüllen hohe Anforderungen an die EMV. Trotzdem sollten Sie vor der Installation der Komponenten eine EMV-Planung durchführen und mögliche Störquellen in die Betrachtung einbeziehen.

**Mögliche Störeinträge**

Elektromagnetische Störungen können sich auf unterschiedlichen Pfaden in Ihre Steuerung einkoppeln:

- Elektromagnetische Felder (HF-Einkopplung)
- Magnetische Felder mit energietechnischer Frequenz
- Bus-System
- Stromversorgung
- Schutzleiter

Je nach Ausbreitungsmedium (leitungsgebunden oder -ungebunden) und Entfernung zur Störquelle gelangen Störungen über unterschiedliche Kopplungsmechanismen in Ihre Steuerung.

Man unterscheidet:

- galvanische Kopplung
- kapazitive Kopplung
- induktive Kopplung
- Strahlungskopplung

**Grundregeln zur Sicherstellung der EMV**

Häufig genügt zur Sicherstellung der EMV das Einhalten einiger elementarer Regeln. Beachten Sie beim Aufbau der Steuerung deshalb die folgenden Grundregeln.

- Achten Sie bei der Montage Ihrer Komponenten auf eine gut ausgeführte flächenhafte Massung der inaktiven Metallteile.
  - Stellen Sie eine zentrale Verbindung zwischen der Masse und dem Erde/Schutzleitersystem her.
  - Verbinden Sie alle inaktiven Metallteile großflächig und impedanzarm.
  - Verwenden Sie nach Möglichkeit keine Aluminiumteile. Aluminium oxidiert leicht und ist für die Massung deshalb weniger gut geeignet.
- Achten Sie bei der Verdrahtung auf eine ordnungsgemäße Leitungsführung.
  - Teilen Sie die Verkabelung in Leitungsgruppen ein. (Starkstrom, Stromversorgungs-, Signal- und Datenleitungen).
  - Verlegen Sie Starkstromleitungen und Signal- bzw. Datenleitungen immer in getrennten Kanälen oder Bündeln.
  - Führen Sie Signal- und Datenleitungen möglichst eng an Masseflächen (z.B. Tragholme, Metallschienen, Schrankbleche).

- Achten sie auf die einwandfreie Befestigung der Leitungsschirme.
  - Datenleitungen sind geschirmt zu verlegen.
  - Analogleitungen sind geschirmt zu verlegen. Bei der Übertragung von Signalen mit kleinen Amplituden kann das einseitige Auflegen des Schirms vorteilhaft sein.
  - Legen Sie die Leitungsschirme direkt nach dem Schrankeintritt großflächig auf eine Schirm-/Schutzleiterschiene auf, und befestigen Sie die Schirme mit Kabelschellen.
  - Achten Sie darauf, dass die Schirm-/Schutzleiterschiene impedanzarm mit dem Schrank verbunden ist.
  - Verwenden Sie für geschirmte Datenleitungen metallische oder metallisierte Steckergehäuse.
- Setzen Sie in besonderen Anwendungsfällen spezielle EMV-Maßnahmen ein.
  - Erwägen Sie bei Induktivitäten den Einsatz von Löschgliedern.
  - Beachten Sie, dass bei Einsatz von Leuchtstofflampen sich diese negativ auf Signalleitungen auswirken können.
- Schaffen Sie ein einheitliches Bezugspotential und erden Sie nach Möglichkeit alle elektrischen Betriebsmittel.
  - Achten Sie auf den gezielten Einsatz der Erdungsmaßnahmen. Das Erden der Steuerung dient als Schutz- und Funktionsmaßnahme.
  - Verbinden Sie Anlagenteile und Schränke mit Ihrer SPS sternförmig mit dem Erde/Schutzleitersystem. Sie vermeiden so die Bildung von Erdschleifen.
  - Verlegen Sie bei Potenzialdifferenzen zwischen Anlagenteilen und Schränken ausreichend dimensionierte Potenzialausgleichsleitungen.

## Schirmung von Leitungen

Elektrische, magnetische oder elektromagnetische Störfelder werden durch eine Schirmung geschwächt; man spricht hier von einer Dämpfung. Über die mit dem Gehäuse leitend verbundene Schirmschiene werden Störströme auf Kabelschirme zur Erde hin abgeleitet. Hierbei ist darauf zu achten, dass die Verbindung zum Schutzleiter impedanzarm ist, da sonst die Störströme selbst zur Störquelle werden.

Bei der Schirmung von Leitungen ist folgendes zu beachten:

- Verwenden Sie möglichst nur Leitungen mit Schirmgeflecht.
- Die Deckungsdichte des Schirmes sollte mehr als 80% betragen.
- In der Regel sollten Sie die Schirme von Leitungen immer beidseitig auflegen. Nur durch den beidseitigen Anschluss der Schirme erreichen Sie eine gute Störunterdrückung im höheren Frequenzbereich. Nur im Ausnahmefall kann der Schirm auch einseitig aufgelegt werden. Dann erreichen Sie jedoch nur eine Dämpfung der niedrigen Frequenzen. Eine einseitige Schirmanbindung kann günstiger sein, wenn:
  - die Verlegung einer Potenzialausgleichsleitung nicht durchgeführt werden kann.
  - Analogsignale (einige mV bzw.  $\mu\text{A}$ ) übertragen werden.
  - Folienschirme (statische Schirme) verwendet werden.
- Benutzen Sie bei Datenleitungen für serielle Kopplungen immer metallische oder metallisierte Stecker. Befestigen Sie den Schirm der Datenleitung am Steckergehäuse. Schirm nicht auf den PIN 1 der Steckerleiste auflegen!
- Bei stationärem Betrieb ist es empfehlenswert, das geschirmte Kabel unterbrechungsfrei abzuisolieren und auf die Schirm-/Schutzleiterschiene aufzulegen.

- Benutzen Sie zur Befestigung der Schirmgeflechte Kabelschellen aus Metall. Die Schellen müssen den Schirm großflächig umschließen und guten Kontakt ausüben.
- Legen Sie den Schirm direkt nach Eintritt der Leitung in den Schrank auf eine Schirmschiene auf. Führen Sie den Schirm bis zu Ihrer SPS weiter, legen Sie ihn dort jedoch nicht erneut auf!

**VORSICHT!****Bitte bei der Montage beachten!**

Bei Potentialdifferenzen zwischen den Erdungspunkten kann über den beidseitig angeschlossenen Schirm ein Ausgleichsstrom fließen.

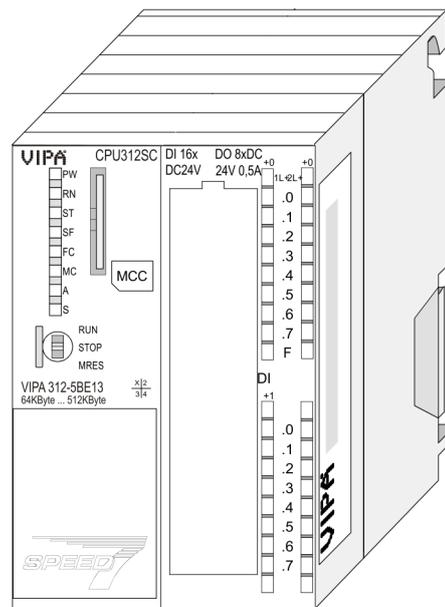
Abhilfe: Potentialausgleichsleitung.

## 4 Hardwarebeschreibung

### 4.1 Leistungsmerkmale

#### CPU 312-5BE13

- SPEED7-Technologie integriert
- 64kByte Arbeitsspeicher integriert (32kByte Code, 32kByte Daten)
- Arbeitsspeicher erweiterbar bis max. 512kByte (256kByte Code, 256kByte Daten)
- 512kByte Ladespeicher
- RS485-Schnittstelle für PtP-Kommunikation
- Ethernet-PG/OP-Schnittstelle integriert
- MPI-Schnittstelle
- MCC-Slot für externe Speichermedien und Speichererweiterung (verriegelbar)
- Status-LEDs für Betriebszustand und Diagnose
- Echtzeituhr akkugepuffert
- Digitale E/As: DI 16xDC24V / DO 8xDC 24V, 0,5A
- 2 Kanäle für Zähler, Frequenzmessung und Pulsweitenmodulation
- E/A-Adressbereich digital/analog 1024Byte
- 512 Zeiten
- 512 Zähler
- 8192 Merker-Byte



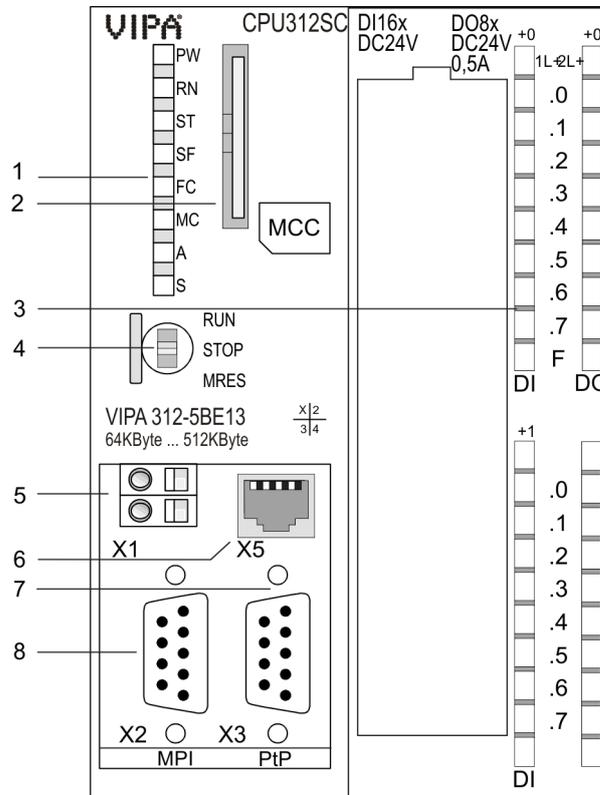
#### Bestelldaten

Typ	Bestellnummer	Beschreibung
312SC	312-5BE13	MPI-Interface, Karten-Slot, Echtzeituhr, Ethernet-Interface für PG/OP, DI 16xDC24V / DO 8xDC 24V, 0,5A, 2 Kanäle Technologische Funktionen

## 4.2 Aufbau

### 4.2.1 Allgemein

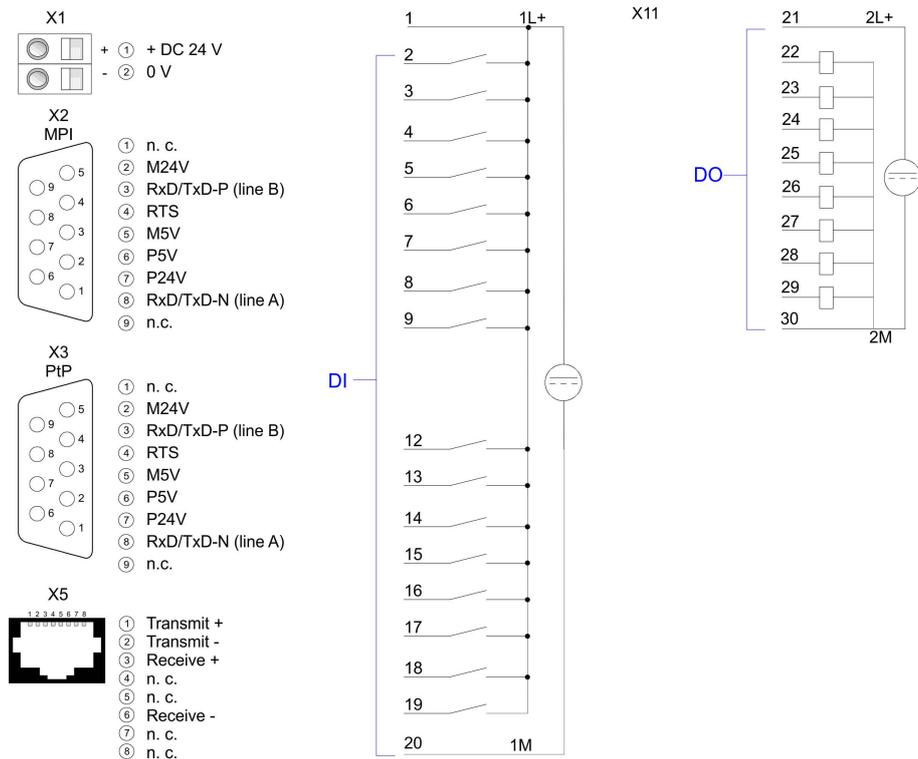
#### CPU 312-5BE13



- 1 LEDs des CPU-Teils
- 2 Steckplatz für Speichermedien (verriegelbar)
- 3 LEDs des E/A-Teils
- 4 Betriebsarten-Schalter CPU
- 5 Anschluss für DC 24V Spannungsversorgung
- 6 Twisted Pair Schnittstelle für Ethernet-PG/OP-Kanal
- 7 PtP-Schnittstelle
- 8 MPI-Schnittstelle

**Komponenten 5 - 8 befinden sich unter der Frontklappe!**

### 4.2.2 Schnittstellen



#### X1: Spannungsversorgung

Die CPU besitzt ein eingebautes Netzteil:

- Das Netzteil ist mit DC 24V zu versorgen. Hierzu dient der DC 24V Anschluss, der sich unter der Frontklappe befindet.
- Mit der Versorgungsspannung werden neben der CPU-Elektronik auch die angeschlossenen Module über den Rückwandbus versorgt.
- Das Netzteil ist gegen Verpolung und Überstrom geschützt.
- Die interne Elektronik ist galvanisch an die Versorgungsspannung gebunden.

#### X2: MPI-Schnittstelle

9polige SubD-Buchse:

- Die MPI-Schnittstelle dient zur Verbindung zwischen Programmiergerät und CPU.
- Hierüber erfolgt beispielsweise die Projektierung und Programmierung.
- MPI dient zur Kommunikation zwischen mehreren CPUs oder zwischen HMIs und CPU.
- Standardmäßig ist die MPI-Adresse 2 eingestellt.

#### X3: PtP-Schnittstelle

9polige SubD-Buchse:

Die CPU besitzt eine PtP-Schnittstelle. Die Schnittstelle ist fix auf PtP-Kommunikation eingestellt.

■ PtP-Funktionalität

- Mit der Funktionalität PtP ermöglicht die RS485-Schnittstelle eine serielle Punkt-zu-Punkt-Prozessan­kopplung zu verschiedenen Ziel- oder Quell-Systemen.
- Unterstützt werden die Protokolle ASCII, STX/ETX, 3964R, USS und Modbus-Master (ASCII, RTU).
- Die PtP-Kommunikation parametrieren Sie zur Laufzeit mit dem SFC 216 (SER\_CFG). Die Kommunikation erfolgt unter Einsatz der SFC 217 (SER\_SND) und SFC 218 (SER\_RCV).

**X5: Ethernet-PG/OP-Kanal**

*8polige RJ45-Buchse:*

- Die RJ45-Buchse dient als Schnittstelle zum Ethernet-PG/OP-Kanal.
- Mittels dieser Schnittstelle können Sie Ihre CPU programmieren bzw. fernwarten und auf die integrierte Web-Seite zugreifen.
- Projektierbare Verbindungen sind nicht möglich.
- Damit Sie online auf den Ethernet-PG/OP-Kanal zugreifen können, müssen Sie diesem IP-Adress-Parameter zuweisen.

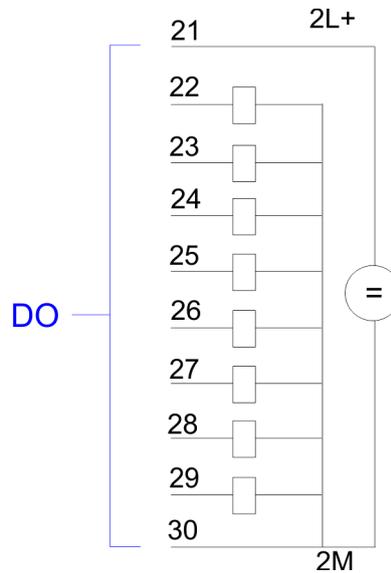
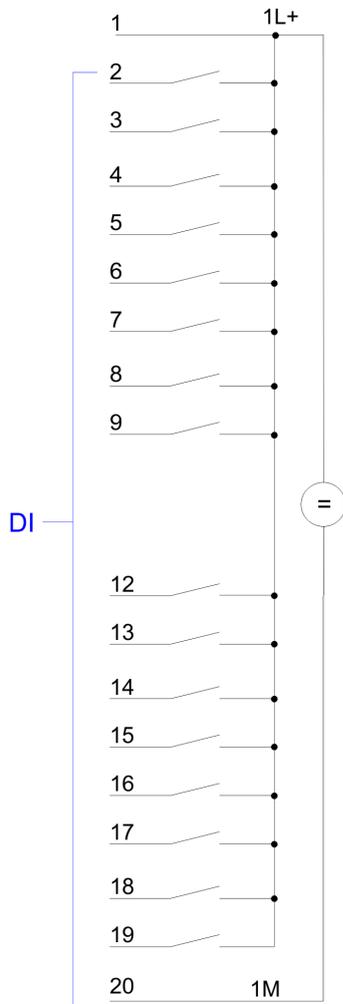
### 4.2.3 Ein-/Ausgabe-Bereich CPU 312-5BE13

#### Übersicht

Bei der CPU 312-5BE13 sind folgende digitale Ein-/Ausgabe-Kanäle in einem Gehäuse untergebracht:

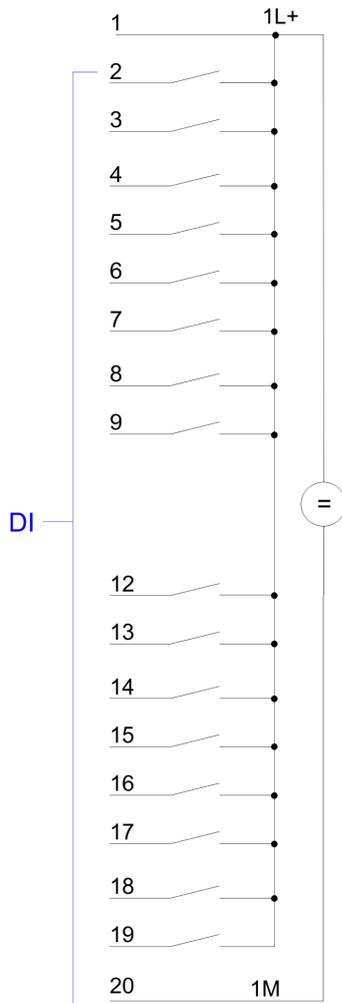
- Digitale Eingabe: 16xDC 24V, alarmfähig
- Digitale Ausgabe: 8xDC 24V, 0,5A
- Technologische Funktionen: 2 Kanäle

X11:

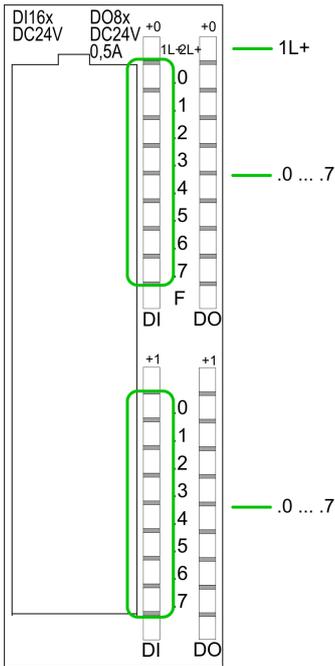


**VORSICHT!**

Bitte beachten Sie, dass die an einem Ausgabe-Kanal anliegende Spannung immer  $\leq$  der über L+ anliegenden Versorgungsspannung ist.

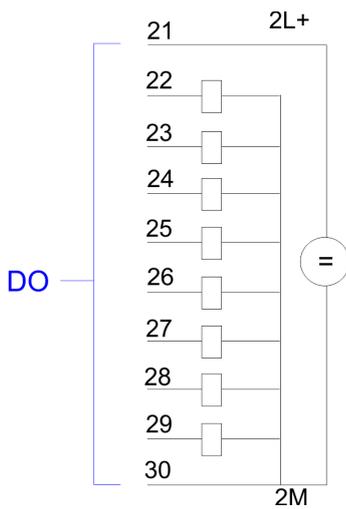
**Steckerbelegung X11: DI**

Pin	Belegung
1	1L+ Spannungsversorgung +DC 24V
2	E+0.0 / Kanal 0 (A) / Impuls
3	E+0.1 / Kanal 0 (B) / Richtung
4	E+0.2 / Kanal 0 Hardwaretor
5	E+0.3 / Kanal 1 (A) / Impuls
6	E+0.4 / Kanal 1 (B) / Richtung
7	E+0.5 / Kanal 1 Hardwaretor
8	E+0.6
9	E+0.7
10	nicht belegt
11	nicht belegt
12	E+1.0
13	E+1.1
14	E+1.2
15	E+1.3
16	E+1.4 / Kanal 0 Latch
17	E+1.5 / Kanal 1 Latch
18	E+1.6
19	E+1.7
20	Masse 1M DI



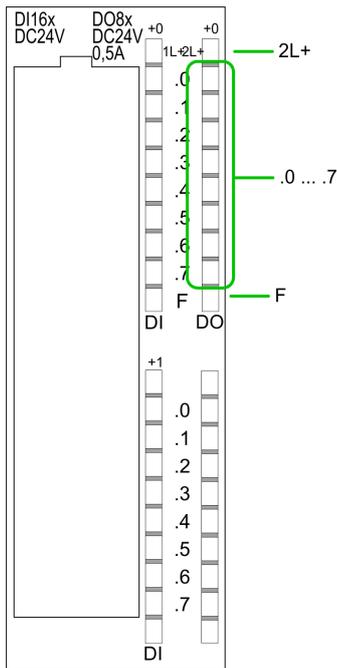
**Statusanzeige X11: DI**

- 1L+
    - LED (grün)
    - Versorgungsspannung für DI liegt an
  - .0... .7
    - LEDs (grün)
    - E+0.0 ... E+0.7
    - E+1.0 ... E+1.7
- Ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert



**Steckerbelegung X11: DO**

Pin	Belegung
21	2L+ Spannungsversorgung +DC 24V
22	A+0.0 / Kanal 0 Ausgang
23	A+0.1 / Kanal 1 Ausgang
24	A+0.2
25	A+0.3
26	A+0.4
27	A+0.5
28	A+0.6
29	A+0.7
30	Masse 2M DO
31 ... 40	nicht belegt



### Statusanzeige X11: DO

- 2L+
  - LED (grün)
  - Versorgungsspannung für DO liegt an
- .0... .7
  - LEDs (grün)
  - A+0.0 ... A+0.7
  - Die entsprechende LED leuchtet bei aktiviertem Ausgang
- F
  - LED (rot)
  - Fehler bei Überlast oder Kurzschluss

## 4.2.4 Speichermanagement

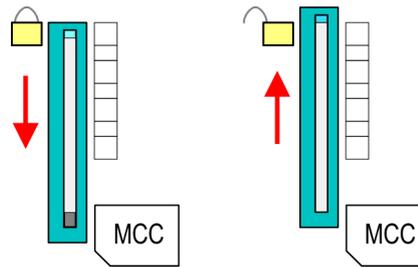
### Speicher

Die CPU hat einen Speicher integriert. Angaben über die Speicherkapazität finden Sie auf der Frontseite Ihrer CPU. Der Speicher gliedert sich in folgende Teile:

- Ladespeicher 512kByte
- Codespeicher (50% des Arbeitsspeichers)
- Datenspeicher (50% des Arbeitsspeichers)
- Arbeitsspeicher 64kByte
  - Sie haben die Möglichkeit den Arbeitsspeicher mittels einer MCC Speichererweiterungskarte bis zur maximal aufgedruckten Kapazität 512kByte zu erweitern.

## 4.2.5 Steckplatz für Speichermedien

- Über diesen Steckplatz können Sie eine MMC (**M**ultimedia **C**ard) als externes Speichermedium für Programme und Firmware stecken.
- Die VIPA-Speicherkarten sind mit dem PC-Format FAT vorformatiert und können mit einem Kartenlesegerät beschrieben werden.
- Nach PowerON bzw. nach Umrücken überprüft die CPU, ob eine Speicherkarte gesteckt ist und sich hier für die CPU gültige Daten befinden.
- Schieben Sie ihr Speichermedium in den Steckplatz, bis dieses geführt durch eine Federmechanik einrastet. Dies gewährleistet eine sichere Kontaktierung.
- Mit der Schiebemechanik können Sie durch Schieben nach unten ein gestecktes Speichermedium gegen Herausfallen sichern.
- Zum Entnehmen schieben Sie die Schiebemechanik wieder nach oben und drücken Sie das Speichermedium gegen den Federdruck nach innen, bis dieses mit einem Klick entriegelt wird.



**VORSICHT!**

Sofern das Speichermedium schon durch die Federmechanik entriegelt wurde, kann dieses bei Betätigung der Schiebemechanik herauspringen!

**4.2.6 Batteriepufferung für Uhr und RAM**

Die CPU besitzt einen internen Akku, der zur Sicherung des RAMs bei Stromausfall dient. Zusätzlich wird die interne Uhr über den Akku gepuffert. Der Akku wird direkt über die eingebaute Spannungsversorgung über eine Ladeelektronik geladen und gewährleistet eine Pufferung für max. 30 Tage.



- Bitte schließen Sie die CPU mindestens für 24 Stunden an die Spannungsversorgung an, damit der interne Akku entsprechend geladen wird.
- Bitte beachten Sie, dass bei wiederholten Teilladezyklen (Laden/Puffern) sich die Pufferzeit fortlaufend reduzieren kann. Nur nach einer Ladezeit von 24 Stunden ist eine Pufferung für max. 30 Tage möglich.



**VORSICHT!**

- Bei leerem Akku läuft die CPU nach einem Spannungsreset mit einem BAT-Fehler an und führt ein automatisches Urlöschen der CPU durch. Der BAT-Fehler hat keinen Einfluss auf den Ladevorgang.
- Den BAT-Fehler können Sie wieder löschen, wenn einmalig beim Power-Cycle zwischen dem Aus- und Einschalten der Versorgungsspannung mindestens 30sec. liegen und der Akku der CPU voll geladen ist. Ansonsten bleibt bei einem kurzen Power-Cycle der BAT-Fehler bestehen und die CPU wird urlöscht.

**4.2.7 Betriebsartenschalter**



- Mit dem Betriebsartenschalter können Sie bei der CPU zwischen den Betriebsarten STOP und RUN wählen.
- Beim Übergang vom Betriebszustand STOP nach RUN durchläuft die CPU den Betriebszustand ANLAUF.
- Mit der Tasterstellung MRES (Memory Reset) fordern Sie das Urlöschen an mit anschließendem Laden von Speicherkarte, sofern dort ein Projekt hinterlegt ist.

## 4.2.8 LEDs

## LEDs CPU

Sobald die CPU intern mit 5V versorgt wird, leuchtet die grüne PW-LED (Power).

RN (RUN)	ST (STOP)	SF (SFAIL)	FC (FRCE)	MC (MMC)	Bedeutung
grün 	gelb 	rot 	gelb 	gelb 	
Bootvorgang nach NetzEIN					
●	BB*	●	●	●	* Blinken mit 10Hz: Firmware wird geladen.
●	●	●	●	●	Initialisierung: Phase 1
●	●	●	●	○	Initialisierung: Phase 2
●	●	●	○	○	Initialisierung: Phase 3
○	●	●	○	○	Initialisierung: Phase 4
Betrieb					
○	●	X	X	X	CPU befindet sich im Zustand STOP.
BB	○	X	X	X	CPU befindet sich im Zustand Anlauf. Solange der OB100 durchlaufen wird, blinkt die RUN-LED, mindestens für 3s.
●	○	○	X	X	CPU befindet sich ohne Fehler im Zustand RUN.
X	X	●	X	X	Es liegt ein Systemfehler vor. Nähere Informationen hierzu finden Sie im Diagnosepuffer der CPU.
X	X	X	●	X	Variablen sind geforced (fixiert).
X	X	X	X	●	Zugriff auf Speicherkarte.
X	BB*	○	○	○	* Blinken mit 10Hz: Konfiguration wird geladen.
Urlöschen					
○	BB	X	X	X	Urlöschen wird angefordert.
○	BB*	X	X	X	* Blinken mit 5Hz: Urlöschen wird durchgeführt.
Rücksetzen auf Werkseinstellung					
●	●	○	○	○	Rücksetzen auf Werkseinstellung wird durchgeführt.
○	●	●	●	●	Rücksetzen auf Werkseinstellung war erfolgreich.
Firmwareupdate					
○	●	BB	BB	●	Das abwechselnde Blinken zeigt an, dass neue Firmware auf der Speicherkarte vorhanden ist.
○	○	BB	BB	●	Das abwechselnde Blinken zeigt an, dass ein Firmwareupdate durchgeführt wird.
○	●	●	●	●	Firmwareupdate wurde fehlerfrei durchgeführt.

RN (RUN)	ST (STOP)	SF (SFALL)	FC (FRCE)	MC (MMC)	Bedeutung
○	BB*	BB*	BB*	BB*	* Blinken mit 10Hz: Fehler bei Firmwareupdate.

an: ● | aus: ○ | blinkend (2Hz): BB | nicht relevant: X

#### LEDs Ethernet-PG/OP-Kanal L/A, S

Die grüne L/A-LED (Link/Activity) zeigt an, dass der Ethernet-PG/OP-Kanal physikalisch mit Ethernet verbunden ist. Unregelmäßiges Blinken der L/A-LED zeigt Kommunikation des Ethernet-PG/OP-Kanals über Ethernet an.

Leuchtet die grüne S-LED (Speed), so hat der Ethernet-PG/OP-Kanal eine Übertragungsrate von 100MBit/s ansonsten 10MBit/s.

## 4.3 Technische Daten



Bitte beachten Sie, dass bei der Projektierung mit dem Siemens TIA Portal die Anzahl von Timer und Zähler auf die maximal mögliche Anzahl der entsprechenden Siemens CPU limitiert wird!

Artikelnr.	312-5BE13
Bezeichnung	CPU 312SC
SPEED-Bus	-
<b>Technische Daten Stromversorgung</b>	
Versorgungsspannung (Nennwert)	DC 24 V
Versorgungsspannung (zulässiger Bereich)	DC 20,4...28,8 V
Verpolschutz	✓
Stromaufnahme (im Leerlauf)	135 mA
Stromaufnahme (Nennwert)	500 mA
Einschaltstrom	11 A
I <sup>2</sup> t	0,7 A <sup>2</sup> s
max. Stromabgabe am Rückwandbus	3 A
Verlustleistung	8 W
<b>Technische Daten digitale Eingänge</b>	
Anzahl Eingänge	16
Leitungslänge geschirmt	1000 m
Leitungslänge ungeschirmt	600 m
Lastnennspannung	DC 24 V
Verpolschutz der Lastnennspannung	✓
Stromaufnahme aus Lastspannung L+ (ohne Last)	70 mA

## Technische Daten

Artikelnr.	312-5BE13
Nennwert	DC 24 V
Eingangsspannung für Signal "0"	DC 0...5 V
Eingangsspannung für Signal "1"	DC 15...28,8 V
Eingangsspannung Hysterese	-
Frequenzbereich	-
Eingangswiderstand	-
Eingangsstrom für Signal "1"	6 mA
Anschluss von 2-Draht-BERO möglich	✓
max. zulässiger BERO-Ruhestrom	1,5 mA
Eingangsverzögerung von "0" nach "1"	0,1 / 0,35 ms
Eingangsverzögerung von "1" nach "0"	0,1 / 0,35 ms
Anzahl gleichzeitig nutzbarer Eingänge waagrechtlicher Aufbau	16
Anzahl gleichzeitig nutzbarer Eingänge senkrechter Aufbau	16
Eingangskennlinie	IEC 61131-2, Typ 1
Eingangsdatengröße	2 Byte
<b>Technische Daten digitale Ausgänge</b>	
Anzahl Ausgänge	8
Leitungslänge geschirmt	1000 m
Leitungslänge ungeschirmt	600 m
Lastnennspannung	DC 24 V
Verpolschutz der Lastnennspannung	-
Stromaufnahme aus Lastspannung L+ (ohne Last)	100 mA
Summenstrom je Gruppe, waagrechtlicher Aufbau, 40°C	3 A
Summenstrom je Gruppe, waagrechtlicher Aufbau, 60°C	2 A
Summenstrom je Gruppe, senkrechter Aufbau	2 A
Ausgangsspannung "1"-Signal bei minimalem Strom	L+ (-0,8 V)
Ausgangsspannung "1"-Signal bei maximalem Strom	L+ (-0,8 V)
Ausgangsstrom bei "1"-Signal, Nennwert	0,5 A
Ausgangsstrom, zulässiger Bereich bis 40°C	5 mA bis 0,6 A
Ausgangsstrom, zulässiger Bereich bis 60°C	5 mA bis 0,6 A
Ausgangsstrom bei "0"-Signal (Reststrom) max.	0,5 mA
Ausgangsverzögerung von "0" nach "1"	100 µs

Artikelnr.	312-5BE13
Ausgangsverzögerung von "1" nach "0"	100 µs
Mindestlaststrom	-
Lampenlast	5 W
Parallelschalten von Ausgängen zur redundanten Ansteuerung	möglich
Parallelschalten von Ausgängen zur Leistungserhöhung	nicht möglich
Ansteuern eines Digitaleingangs	✓
Schaltfrequenz bei ohmscher Last	max. 2,5 kHz
Schaltfrequenz bei induktiver Last	max. 0,5 Hz
Schaltfrequenz bei Lampenlast	max. 2,5 kHz
Begrenzung (intern) der induktiven Abschaltspannung	L+ (-52 V)
Kurzschlusschutz des Ausgangs	ja, elektronisch
Ansprechschwelle des Schutzes	1 A
Anzahl Schaltspiele der Relaisausgänge	-
Schaltvermögen der Relaiskontakte	-
Ausgangsdatengröße	1 Byte
<b>Technische Daten Analoge Eingänge</b>	
Anzahl Eingänge	-
Leitungslänge geschirmt	-
Lastnennspannung	-
Verpolschutz der Lastnennspannung	-
Stromaufnahme aus Lastspannung L+ (ohne Last)	-
Spannungseingänge	-
min. Eingangswiderstand im Spannungsbereich	-
Eingangsspannungsbereiche	-
Gebrauchsfehlergrenze Spannungsbereiche	-
Gebrauchsfehlergrenze Spannungsbereiche mit SFU	-
Grundfehlergrenze Spannungsbereiche	-
Grundfehlergrenze Spannungsbereiche mit SFU	-
Zerstörgrenze Spannung	-
Stromeingänge	-
max. Eingangswiderstand im Strombereich	-
Eingangsstrombereiche	-

## Technische Daten

Artikelnr.	312-5BE13
Gebrauchsfehlergrenze Strombereiche	-
Gebrauchsfehlergrenze Strombereiche mit SFU	-
Grundfehlergrenze Strombereiche	-
Grundfehlergrenze Strombereiche mit SFU	-
Zerstörgrenze Stromeingänge (Strom)	-
Zerstörgrenze Stromeingänge (Spannung)	-
Widerstandseingänge	-
Widerstandsbereiche	-
Gebrauchsfehlergrenze Widerstandsbereiche	-
Gebrauchsfehlergrenze Widerstandsbereiche mit SFU	-
Grundfehlergrenze Widerstandsbereiche	-
Grundfehlergrenze Widerstandsbereiche mit SFU	-
Zerstörgrenze Widerstandseingänge	-
Widerstandsthermometereingänge	-
Widerstandsthermometerbereiche	-
Gebrauchsfehlergrenze Widerstandsthermometerbereiche	-
Gebrauchsfehlergrenze Widerstandsthermometerbereiche mit SFU	-
Grundfehlergrenze Widerstandsthermometerbereiche	-
Grundfehlergrenze Widerstandsthermometerbereiche mit SFU	-
Zerstörgrenze Widerstandsthermometereingänge	-
Thermoelementeingänge	-
Thermoelementbereiche	-
Gebrauchsfehlergrenze Thermoelementbereiche	-
Gebrauchsfehlergrenze Thermoelementbereiche mit SFU	-
Grundfehlergrenze Thermoelementbereiche	-
Grundfehlergrenze Thermoelementbereiche mit SFU	-
Zerstörgrenze Thermoelementeingänge	-
Temperaturkompensation parametrierbar	-
Temperaturkompensation extern	-
Temperaturkompensation intern	-

Artikelnr.	312-5BE13
Technische Einheit der Temperaturmessung	-
Auflösung in Bit	-
Messprinzip	-
Grundwandlungszeit	-
Störspannungsunterdrückung für Frequenz	-
Eingangsdatengröße	-
<b>Technische Daten Analoge Ausgänge</b>	
Anzahl Ausgänge	-
Leitungslänge geschirmt	-
Lastnennspannung	-
Verpolschutz der Lastnennspannung	-
Stromaufnahme aus Lastspannung L+ (ohne Last)	-
Spannungsausgang Kurzschlusschutz	-
Spannungsausgänge	-
min. Bürdenwiderstand im Spannungsbereich	-
max. kapazitive Last im Spannungsbereich	-
max. Kurzschlussstrom des Spannungsausgangs	-
Ausgangsspannungsbereiche	-
Gebrauchsfehlergrenze Spannungsbereiche	-
Grundfehlergrenze Spannungsbereiche mit SFU	-
Zerstörgrenze gegen von außen angelegte Spannungen	-
Stromausgänge	-
max. Bürdenwiderstand im Strombereich	-
max. induktive Last im Strombereich	-
typ. Leerlaufspannung des Stromausgangs	-
Ausgangsstrombereiche	-
Gebrauchsfehlergrenze Strombereiche	-
Grundfehlergrenze Strombereiche mit SFU	-
Zerstörgrenze gegen von außen angelegten Strom	-
Einschwingzeit für ohmsche Last	-
Einschwingzeit für kapazitive Last	-
Einschwingzeit für induktive Last	-
Auflösung in Bit	-

## Technische Daten

Artikelnr.	312-5BE13
Wandlungszeit	-
Ersatzwerte aufschaltbar	-
Ausgangsdatengröße	-
<b>Technische Daten Zähler</b>	
Anzahl Zähler	2
Zählerbreite	32 Bit
maximale Eingangsfrequenz	10 kHz
maximale Zählfrequenz	10 kHz
Betriebsart Inkrementalgeber	✓
Betriebsart Impuls/Richtung	✓
Betriebsart Impuls	✓
Betriebsart Frequenzmessung	✓
Betriebsart Periodendauermessung	✓
Gate-Anschluss möglich	✓
Latch-Anschluss möglich	✓
Reset-Anschluss möglich	-
Zähler-Ausgang möglich	✓
<b>Lade- und Arbeitsspeicher</b>	
Ladespeicher integriert	512 KB
Ladespeicher maximal	512 KB
Arbeitsspeicher integriert	64 KB
Arbeitsspeicher maximal	512 KB
Speicher geteilt 50% Code / 50% Daten	✓
Memory Card Slot	MMC-Card mit max. 1 GB
<b>Ausbau</b>	
Baugruppenträger max.	1
Baugruppen je Baugruppenträger	8
Anzahl DP-Master integriert	0
Anzahl DP-Master über CP	4
Betreibbare Funktionsbaugruppen	8
Betreibbare Kommunikationsbaugruppen PtP	8
Betreibbare Kommunikationsbaugruppen LAN	8
<b>Status, Alarm, Diagnosen</b>	
Statusanzeige	ja
Alarmer	ja
Prozessalarm	ja

Artikelnr.	312-5BE13
Diagnosealarm	ja
Diagnosefunktion	nein
Diagnoseinformation auslesbar	möglich
Versorgungsspannungsanzeige	grüne LED
Sammelfehleranzeige	rote SF-LED
Kanalfehleranzeige	rote LED pro Gruppe
<b>Potenzialtrennung</b>	
zwischen den Kanälen	✓
zwischen den Kanälen in Gruppen zu	16
zwischen Kanälen und Rückwandbus	✓
zwischen Kanälen und Spannungsversorgung	-
max. Potenzialdifferenz zwischen Stromkreisen	DC 75 V/ AC 50 V
max. Potenzialdifferenz zwischen Eingängen (Ucm)	-
max. Potenzialdifferenz zwischen Mana und Mintern (Uiso)	-
max. Potenzialdifferenz zwischen Eingängen und Mana (Ucm)	-
max. Potenzialdifferenz zwischen Eingängen und Mintern (Uiso)	-
max. Potenzialdifferenz zwischen Mintern und Ausgängen	-
Isolierung geprüft mit	DC 500 V
<b>Befehlsbearbeitungszeiten</b>	
Bitoperation, min.	0,02 µs
Wortoperation, min.	0,02 µs
Festpunktarithmetik, min.	0,02 µs
Gleitpunktarithmetik, min.	0,12 µs
<b>Zeiten/Zähler und deren Remanenz</b>	
Anzahl S7-Zähler	512
Anzahl S7-Zeiten	512
<b>Datenbereiche und Remanenz</b>	
Anzahl Merker	8192 Byte
Anzahl Datenbausteine	4095
max. Datenbausteingröße	64 KB
max. Lokaldatengröße je Ablaufebene	510 Byte
<b>Bausteine</b>	
Anzahl OBs	15

Artikelnr.	312-5BE13
Anzahl FBs	2048
Anzahl FCs	2048
maximale Schachtelungstiefe je Prioklasse	8
maximale Schachtelungstiefe zusätzlich innerhalb Fehler OB	4
<b>Uhrzeit</b>	
Uhr gepuffert	✓
Uhr Pufferungsdauer (min.)	6 w
Genauigkeit (max. Abweichung je Tag)	10 s
Anzahl Betriebsstundenzähler	8
Uhrzeit Synchronisation	✓
Synchronisation über MPI	Master/Slave
Synchronisation über Ethernet (NTP)	nein
<b>Adressbereiche (Ein-/Ausgänge)</b>	
Peripherieadressbereich Eingänge	1024 Byte
Peripherieadressbereich Ausgänge	1024 Byte
Prozessabbild Eingänge maximal	128 Byte
Prozessabbild Ausgänge maximal	128 Byte
Digitale Eingänge	272
Digitale Ausgänge	264
Digitale Eingänge zentral	272
Digitale Ausgänge zentral	264
Integrierte digitale Eingänge	16
Integrierte digitale Ausgänge	8
Analoge Eingänge	64
Analoge Ausgänge	64
Analoge Eingänge zentral	64
Analoge Ausgänge zentral	64
Integrierte analoge Eingänge	0
Integrierte analoge Ausgänge	0
<b>Kommunikationsfunktionen</b>	
PG/OP Kommunikation	✓
Globale Datenkommunikation	✓
Anzahl GD-Kreise max.	4
Größe GD-Pakete, max.	22 Byte
S7-Basis-Kommunikation	✓

Artikelnr.	312-5BE13
S7-Basis-Kommunikation Nutzdaten je Auftrag	76 Byte
S7-Kommunikation	✓
S7-Kommunikation als Server	✓
S7-Kommunikation als Client	-
S7-Kommunikation Nutzdaten je Auftrag	160 Byte
Anzahl Verbindungen gesamt	32
<b>PWM Daten</b>	
PWM Kanäle	2
PWM-Zeitbasis	0,1ms/1ms
Periodendauer	4...65535 / 1...65535 * Zeitbasis
minimale Pulsbreite	0...0,5*Periodendauer
Ausgangstyp	Highside mit 1,1kOhm Pulldown
<b>Funktionalität Sub-D Schnittstellen</b>	
Bezeichnung	X2
Physik	RS485
Anschluss	9polige SubD Buchse
Potenzialgetrennt	-
MPI	✓
MP <sup>2</sup> I (MPI/RS232)	-
DP-Master	-
DP-Slave	-
Punkt-zu-Punkt-Kopplung	-
Bezeichnung	X3
Physik	RS485
Anschluss	9polige SubD Buchse
Potenzialgetrennt	✓
MPI	-
MP <sup>2</sup> I (MPI/RS232)	-
DP-Master	-
DP-Slave	-
Punkt-zu-Punkt-Kopplung	✓
<b>Funktionalität MPI</b>	
Anzahl Verbindungen, max.	32
PG/OP Kommunikation	✓
Routing	-

## Technische Daten

Artikelnr.	312-5BE13
Globale Datenkommunikation	✓
S7-Basis-Kommunikation	✓
S7-Kommunikation	✓
S7-Kommunikation als Server	✓
S7-Kommunikation als Client	-
Übertragungsgeschwindigkeit, min.	19,2 kbit/s
Übertragungsgeschwindigkeit, max.	187,5 kbit/s
<b>Funktionalität PROFIBUS Master</b>	
PG/OP Kommunikation	-
Routing	-
S7-Basis-Kommunikation	-
S7-Kommunikation	-
S7-Kommunikation als Server	-
S7-Kommunikation als Client	-
Aktivieren/Deaktivieren von DP-Slaves	-
Direkter Datenaustausch (Querverkehr)	-
DPV1	-
Übertragungsgeschwindigkeit, min.	-
Übertragungsgeschwindigkeit, max.	-
Anzahl DP-Slaves, max.	-
Adressbereich Eingänge, max.	-
Adressbereich Ausgänge, max.	-
Nutzdaten Eingänge je Slave, max.	-
Nutzdaten Ausgänge je Slave, max.	-
<b>Funktionalität PROFIBUS Slave</b>	
PG/OP Kommunikation	-
Routing	-
S7-Kommunikation	-
S7-Kommunikation als Server	-
S7-Kommunikation als Client	-
Direkter Datenaustausch (Querverkehr)	-
DPV1	-
Übertragungsgeschwindigkeit, min.	-
Übertragungsgeschwindigkeit, max.	-
Automatische Baudratesuche	-
Übergabespeicher Eingänge, max.	-

Artikelnr.	312-5BE13
Übergabespeicher Ausgänge, max.	-
Adressbereiche, max.	-
Nutzdaten je Adressbereich, max.	-
<b>Point-to-Point Kommunikation</b>	
PtP-Kommunikation	✓
Schnittstelle potentialgetrennt	✓
Schnittstelle RS232	-
Schnittstelle RS422	-
Schnittstelle RS485	✓
Anschluss	9polige SubD Buchse
Übertragungsgeschwindigkeit, min.	150 bit/s
Übertragungsgeschwindigkeit, max.	115,5 kbit/s
Leitungslänge, max.	500 m
<b>Point-to-Point Protokolle</b>	
Protokoll ASCII	✓
Protokoll STX/ETX	✓
Protokoll 3964(R)	✓
Protokoll RK512	-
Protokoll USS Master	✓
Protokoll Modbus Master	✓
Protokoll Modbus Slave	-
Spezielle Protokolle	-
<b>Funktionalität RJ45 Schnittstellen</b>	
Bezeichnung	X5
Physik	Ethernet 10/100 MBit
Anschluss	RJ45
Potenzialgetrennt	✓
PG/OP Kommunikation	✓
max. Anzahl Verbindungen	4
Produktiv Verbindungen	-
<b>Gehäuse</b>	
Material	PPE
Befestigung	Profilschiene System 300
<b>Mechanische Daten</b>	
Abmessungen (BxHxT)	80 mm x 125 mm x 120 mm
Gewicht	410 g

## Technische Daten

<b>Artikelnr.</b>	<b>312-5BE13</b>
<b>Umgebungsbedingungen</b>	
Betriebstemperatur	0 °C bis 60 °C
Lagertemperatur	-25 °C bis 70 °C
<b>Zertifizierungen</b>	
Zertifizierung nach UL	ja

## 5 Einsatz CPU 312-5BE13

### 5.1 Montage



*Informationen zur Montage und zur Verdrahtung:  
↳ Kapitel 3 "Montage und Aufbaurichtlinien" auf Seite 16*

### 5.2 Anlaufverhalten

#### **Stromversorgung einschalten**

Nach dem Einschalten der Stromversorgung geht die CPU in den Betriebszustand über, der am Betriebsartenschalter eingestellt ist.

#### **Auslieferungszustand**

Im Auslieferungszustand ist die CPU urgelöscht. Nach einem STOP→RUN Übergang geht die CPU ohne Programm in RUN.

#### **Anlauf mit gültiger Projektierung in der CPU**

Die CPU geht mit dem Programm, das sich im batteriegepufferten RAM befindet, in RUN.

#### **Anlauf bei leerem Akku**

- Der Akku wird direkt über die eingebaute Spannungsversorgung über eine Ladeelektronik geladen und gewährleistet eine Pufferung für min. 30 Tage. Wird dieser Zeitraum überschritten, kann es zur vollkommenen Entladung des Akkus kommen. Hierbei wird das batteriegepufferte RAM gelöscht.
- In diesem Zustand führt die CPU ein Urlöschen durch. Ist eine MMC gesteckt, werden Programmcode und Datenbausteine von der MMC in den Arbeitsspeicher der CPU übertragen. Ist keine MMC gesteckt, transferiert die CPU permanent abgelegte "protected" Bausteine, falls diese vorhanden sind, in den Arbeitsspeicher.
- Abhängig von der Stellung des Betriebsartenschalters geht die CPU in RUN, sofern der OB81 vorhanden ist, bzw. bleibt im STOP. Dieser Vorgang wird im Diagnosepuffer unter folgendem Eintrag festgehalten: "Start Urlöschen automatisch (ungepuffert NetzEIN)".



#### **VORSICHT!**

Bei leerem Akku läuft die CPU nach einem Spannungsreset mit einem BAT-Fehler an und führt ein automatisches Urlöschen der CPU durch. Den BAT-Fehler können Sie wieder löschen, wenn einmalig beim Power-Cycle zwischen dem Aus- und Einschalten der Versorgungsspannung mindestens 30sec. liegen und der Akku der CPU voll geladen ist. Ansonsten bleibt bei einem kurzen Power-Cycle der BAT-Fehler bestehen und die CPU wird urgelöscht.

## 5.3 Adressierung

### 5.3.1 Übersicht

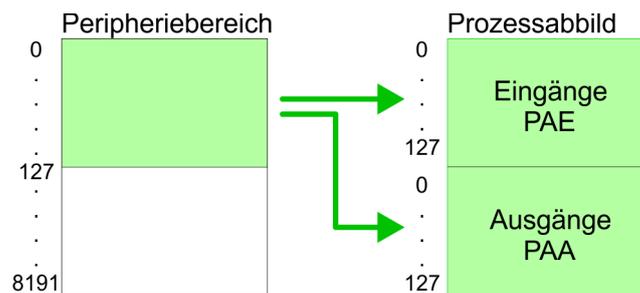
Damit die gesteckten Peripheriemodule gezielt angesprochen werden können, müssen ihnen bestimmte Adressen in der CPU zugeordnet werden. Beim Hochlauf der CPU vergibt diese steckplatzabhängig automatisch von 0 an aufsteigend Peripherieadressen für die gesteckten digitalen Ein- /Ausgabe-Module. Sofern keine Hardwareprojektion vorliegt, legt die CPU gesteckte Analog- Module bei der automatischen Adressierung auf gerade Adressen ab 256 ab.

### 5.3.2 Adressierung Rückwandbus Peripherie

Bei der CPU 312-5BE13 gibt es einen Peripheriebereich (Adresse 0 ... 8191) und ein Prozessabbild der Ein- und Ausgänge (je Adresse 0 ... 127). Beim Prozessabbild werden die Signalzustände der unteren Adresse (0 ... 127) zusätzlich in einem besonderen Speicherbereich gespeichert.

Das Prozessabbild ist in zwei Teile gegliedert:

- Prozessabbild der Eingänge (PAE)
- Prozessabbild der Ausgänge (PAA)



Nach jedem Zyklusdurchlauf wird das Prozessabbild aktualisiert.

#### Maximale Anzahl steckbarer Module

Die CPU 312-5BE13 kann maximal 8 Module adressieren. Eine Erweiterung mittels Erweiterungs-Rack ist nicht möglich.

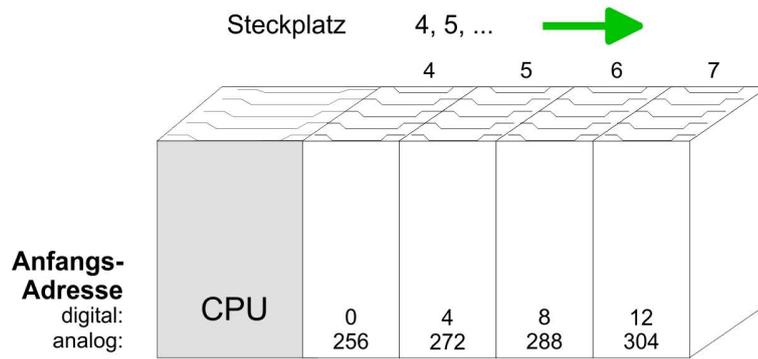
#### Über Hardware-Konfiguration Adressen definieren

Über Lese- bzw. Schreibzugriffe auf die Peripheriebytes oder auf das Prozessabbild können Sie die Module ansprechen. Mit einer Hardware-Konfiguration können Sie Adressen definieren. Klicken Sie hierzu auf die Eigenschaften des entsprechenden Moduls und stellen Sie die gewünschte Adresse ein.

#### Automatische Adressierung

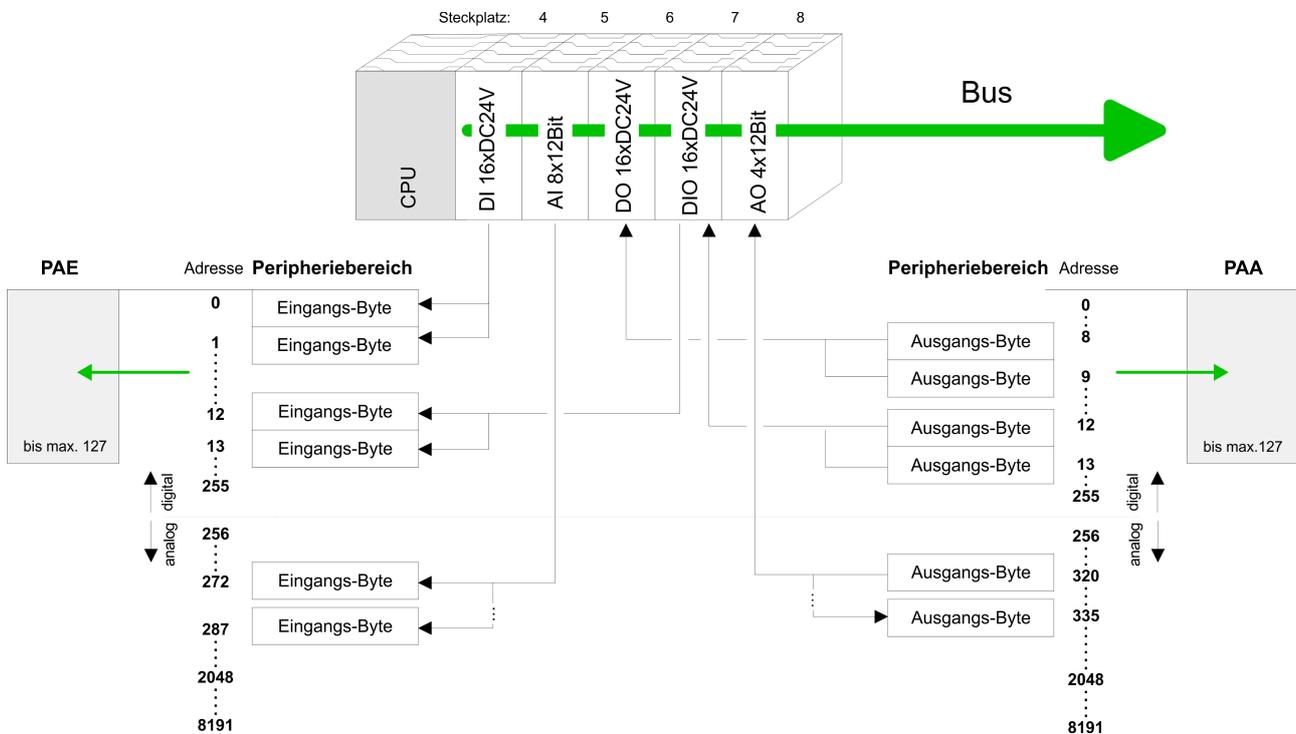
Falls Sie keine Hardware-Konfiguration verwenden möchten, tritt eine automatische Adressierung in Kraft. Bei der automatischen Adressierung belegen steckplatzabhängig DIOs immer 4Byte und AIOs, FMs, CPs immer 16Byte am Bus. Nach folgenden Formeln wird steckplatzabhängig die Anfangsadresse ermittelt, ab der das entsprechende Modul im Adressbereich abgelegt wird:

- DIOs: Anfangsadresse =  $4 \times (\text{Steckplatz} - 4)$
- AIOs, FMs, CPs: Anfangsadresse =  $16 \times (\text{Steckplatz} - 4) + 256$



**Beispiel Automatische Adressierung**

In dem nachfolgenden Beispiel ist die Funktionsweise der automatischen Adressierung nochmals aufgeführt:



**5.4 Adressbelegung**

**Eingabebereich**

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Eingabe E+0.0 ... E+0.7
	125	Byte	Digitale Eingabe E+1.0 ... E+1.7
Zähler	768	DInt	Kanal 0: Zählerwert / Frequenzwert
	772	DInt	Kanal 1: Zählerwert / Frequenzwert

Submodul	Default-Adresse	Zugriff	Belegung
	776	DInt	reserviert
	780	DInt	reserviert

### Ausgabebereich

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Ausgabe A+0.0 ... A+0.7
Zähler	768	DWort	reserviert
	772	DWort	reserviert
	776	DWort	reserviert
	780	DWort	reserviert

## 5.5 Hardware-Konfiguration - CPU

### Voraussetzung

Die Konfiguration der CPU erfolgt im *"Hardware-Konfigurator"* von Siemens. Der Hardware-Konfigurator ist Bestandteil des Siemens SIMATIC Managers. Die Module, die hier projiziert werden können, entnehmen Sie dem Hardware-Katalog, ggf. müssen Sie mit *"Extras → Katalog aktualisieren"* den Hardware-Katalog aktualisieren.

Für die Projektierung werden fundierte Kenntnisse im Umgang mit dem Siemens SIMATIC Manager und dem Hardware-Konfigurator vorausgesetzt!



*Bitte beachten Sie, dass diese SPEED7-CPU 4 AKKUs besitzt. Nach einer arithmetischen Operation (+I, -I, \*I, /I, +D, -D, \*D, /D, MOD, +R, -R, \*R, /R) wird der Inhalt des AKKUs 3 und 4 in die AKKUs 2 und 3 geladen. Dies kann bei Programmen, die einen unveränderten AKKU 2 voraussetzen, zu Konflikten führen.*

*Nähere Informationen hierzu finden Sie im Handbuch "VIPA Operationsliste SPEED7" unter "Unterschiede zwischen SPEED7 und 300V Programmierung".*

### Vorgehensweise

Steckpl.	Modul
1	
2	<b>CPU 312C</b>
2.2	DI10/DO6
2.4	Zähler
3	

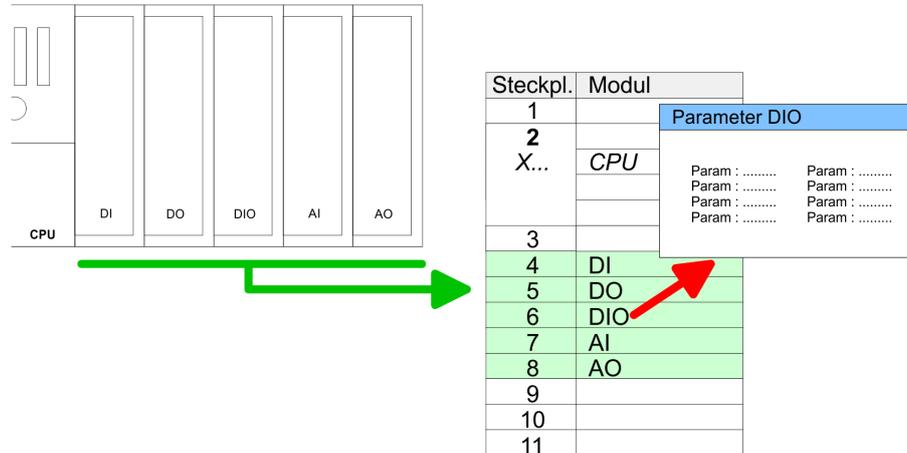
Um kompatibel mit dem Siemens SIMATIC Manager zu sein, sind folgende Schritte durchzuführen:

1. ▶ Starten Sie den Hardware-Konfigurator von Siemens mit einem neuen Projekt.
2. ▶ Fügen Sie aus dem Hardware-Katalog eine Profilschiene ein.
3. ▶ Platzieren Sie auf "Slot"-Nummer 2 die CPU 312C (6ES7 312-5BE03-0AB0 V2.6).

### 5.6 Hardware-Konfiguration - I/O-Module

#### Hardware-Konfiguration der Module

Binden Sie nach der Hardware-Konfiguration der CPU beginnend mit Steckplatz 4 Ihre System 300 Module auf dem Bus in der gesteckten Reihenfolge ein.



#### Parametrierung

Zur Parametrierung doppelklicken Sie in Ihrer Steckplatzübersicht auf das zu parametrierende Modul. Daraufhin öffnet sich ein Dialogfenster. Hier können Sie Ihre Parametereinstellungen vornehmen. Unter Einsatz der SFCs 55, 56 und 57 können Sie zur Laufzeit Parameter ändern und an die entsprechenden Module übertragen. Hierbei sind die modulspezifischen Parameter in sogenannten "Datensätzen" abzulegen. Näheres zum Aufbau der Datensätze finden Sie in der Beschreibung zu den Modulen.

#### Maximal 8 Peripherie-Module

Die CPU 312-5BE13 kann maximal 8 Module adressieren. Eine Erweiterung mittels Erweiterungs-Rack ist nicht möglich!

### 5.7 Hardware-Konfiguration - Ethernet-PG/OP-Kanal

#### Übersicht

Die CPU 312-5BE13 hat einen Ethernet-PG/OP-Kanal integriert. Über diesen Kanal können Sie Ihre CPU programmieren und fernwarten. Mit dem PG/OP-Kanal haben Sie auch Zugriff auf die interne Webseite, auf der Sie Informationen zu Firmwarestand, angebundene Peripherie, aktuelle Zyklus-Zeiten usw. finden. Bei Erstinbetriebnahme bzw. nach dem Rücksetzen auf Werkseinstellungen besitzt der Ethernet-PG/OP-Kanal keine IP-Adresse. Damit Sie online über den Ethernet-PG/OP-Kanal auf die CPU zugreifen können, müssen Sie diesem gültige IP-Adress-Parameter über den Siemens SIMATIC Manager zuordnen. Diesen Vorgang nennt man "Initialisierung" oder "Urtaufe".

#### Montage und Inbetriebnahme

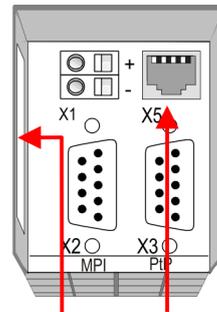
1. ► Bauen Sie Ihr System 300S mit Ihrer CPU auf.
2. ► Verdrahten Sie das System, indem Sie die Leitungen für Spannungsversorgung und Signale anschließen.
3. ► Verbinden Sie die Ethernet-Buchse des Ethernet-PG/OP-Kanals mit Ethernet.

**4.** Schalten Sie die Spannungsversorgung ein

⇒ Nach kurzer Hochlaufzeit ist der CP bereit für die Kommunikation. Er besitzt ggf. noch keine IP-Adressdaten und erfordert eine Urtaufe.

**"Urtaufe" über Zielsystemfunktionen**

Die Urtaufe über die Zielsystemfunktion erfolgt nach folgender Vorgehensweise:



**Ethernet address    PG/OP channel**

→ Ermitteln Sie die aktuelle Ethernet (MAC) Adresse Ihres Ethernet PG/OP-Kanals. Sie finden diese immer als 1. Adresse unter der Frontklappe der CPU auf einem Aufkleber auf der linken Seite.

**IP-Adress-Parameter zuweisen**

Gültige IP-Adress-Parameter erhalten Sie von Ihrem Systemadministrator. Die Zuweisung der IP-Adress-Daten erfolgt online im Siemens SIMATIC Manager ab Version V 5.3 & SP3 nach folgender Vorgehensweise:

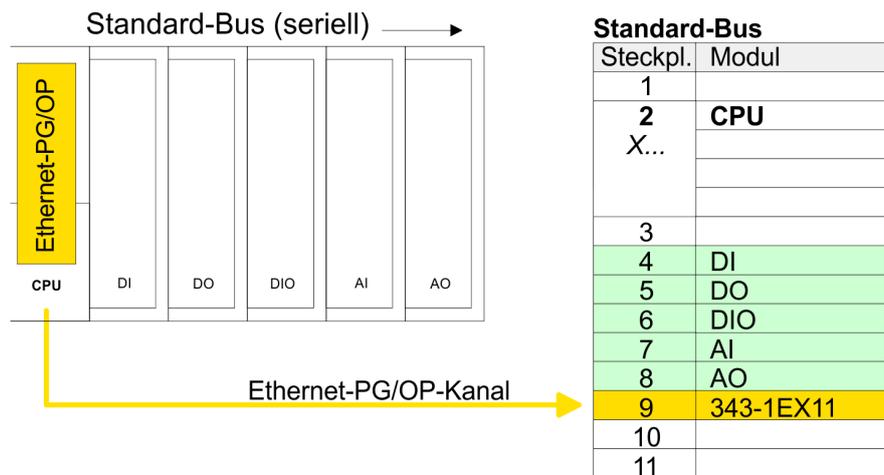
- 1.** Starten Sie den Siemens SIMATIC Manager und stellen Sie über "Extras → PG/PC-Schnittstelle einstellen" auf "TCP/IP -> Netzwerkkarte ...." ein.
- 2.** Öffnen Sie mit "Zielsystem → Ethernet-Teilnehmer bearbeiten" das gleichnamige Dialogfenster.
- 3.** Benutzen Sie die Schaltfläche [Durchsuchen], um die über MAC-Adresse erreichbaren Geräte zu ermitteln oder tragen Sie die MAC-Adresse ein. Die MAC-Adresse finden Sie auf dem 1. Aufkleber unter der Frontklappe der CPU.
- 4.** Wählen Sie ggf. bei der Netzwerksuche aus der Liste die Baugruppe mit der Ihnen bekannten MAC-Adresse aus.
- 5.** Stellen Sie nun die IP-Konfiguration ein, indem Sie IP-Adresse, Subnetz-Maske und den Netzübergang eintragen.
- 6.** Bestätigen Sie mit [IP-Konfiguration zuweisen] Ihre Eingabe.
 

⇒ Direkt nach der Zuweisung ist der Ethernet-PG/OP-Kanal über die angegebenen IP-Adress-Daten online erreichbar. Der Wert bleibt bestehen, solange dieser nicht neu zugewiesen, mit einer Hardware-Projektierung überschrieben oder Rücksetzen auf Werkseinstellung ausgeführt wird.

**IP-Adress-Parameter in Projekt übernehmen**

- 1.** Öffnen Sie den Siemens Hardware-Konfigurator und projektieren Sie die Siemens CPU 312C (6ES7 312-5BE03-0AB0 V2.6).
- 2.** Projektieren Sie die Module am Standard-Bus.

3. Für den Ethernet-PG/OP-Kanal ist immer unterhalb der reell gesteckten Module ein Siemens CP 343-1 (SIMATIC 300 \ CP 300 \ Industrial Ethernet \ CP 343-1 \ 6GK7 343-1EX11 0XE0) zu platzieren.
4. Öffnen Sie durch Doppelklick auf den CP 343-1EX11 den Eigenschaften-Dialog und geben Sie für den CP unter "Eigenschaften" die zuvor zugewiesenen IP-Adress-Daten an.
5. Ordnen Sie den CP einem "Subnetz" zu. Ohne Zuordnung werden die IP-Adress-Daten nicht übernommen!
6. Übertragen Sie Ihr Projekt.

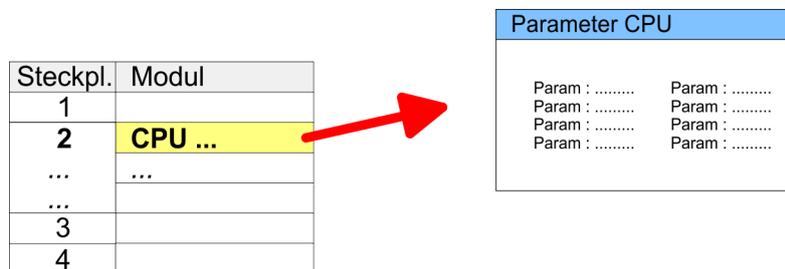


## 5.8 Einstellung der CPU-Parameter

### 5.8.1 Parametrierung über Siemens CPU

#### Parametrierung über Siemens CPU 312C

Da die CPU im Hardware-Konfigurator als Siemens CPU 312C (CPU 312-5BE03-0AB0 V2.6) zu projektieren ist, können Sie bei der Hardware-Konfiguration unter den "Eigenschaften" der CPU 312C die Parameter für die CPU 312-5BE13 einstellen. Durch Doppelklick auf die CPU 312-5BE13 gelangen Sie in das Parametrierfenster für die CPU. Über die Register haben Sie Zugriff auf alle Parameter Ihrer CPU.



Beschreibung der Parameter der Submodule "DI10/DO6" und "Zähler" ↪ Kapitel 6 "Einsatz E/A-Peripherie" auf Seite 93

## 5.8.2 Parameter CPU

### Parameter, die unterstützt werden

Die CPU wertet nicht alle Parameter aus, welche Sie bei der Hardware-Konfiguration einstellen können. Folgende Parameter werden zur Zeit in der CPU ausgewertet:

#### Allgemein

- Kurzbezeichnung: Die Kurzbezeichnung der Siemens CPU 312-5BE03 ist CPU 312C.
- Bestell-Nr./ Firmware: Bestellnummer und Firmware sind identisch zu den Angaben im Fenster "Hardware Katalog".
- Name: Als Name steht hier die Kurzbezeichnung der CPU. Wenn Sie den Namen ändern, erscheint dieser im Siemens SIMATIC Manager.
- Schnittstelle: Hier ist die Adresse der MPI-Schnittstelle angegeben.
- Eigenschaften: Über diese Schaltfläche können Sie die Eigenschaften der MPI-Schnittstelle einstellen.
- Kommentar: Hier können Sie den Einsatzzweck der Baugruppe eingeben.

#### Anlauf

- Anlauf bei Sollausbau ungleich Istausbau: Wenn "*Anlauf bei Sollausbau ungleich Istausbau*" deaktiviert ist und mindestens eine Baugruppe nicht auf dem projektierten Steckplatz steckt, oder dort eine Baugruppe von einem anderen Typ steckt, geht die CPU nicht in RUN und verbleibt in STOP. Wenn "*Anlauf bei Sollausbau ungleich Istausbau*" aktiviert ist, läuft die CPU an, auch wenn Baugruppen nicht auf den projektierten Steckplätzen stecken oder dort Baugruppen eines anderen Typs stecken (z.B. bei Inbetriebnahme).
- Überwachungszeit für Fertigmeldung durch Baugruppen [100ms]: Maximale Dauer für die Fertigmeldung aller konfigurierten Baugruppen nach NetzeIN. Wenn nach Ablauf dieser Zeit die Baugruppen keine Fertigmeldung an die CPU senden, ist der Istausbau ungleich dem Sollausbau.
- Überwachungszeit für Übertragung der Parameter an Baugruppen [100ms]: Maximale Dauer für die Übertragung der Parameter an die parametrierbaren Baugruppen. Wenn nach Ablauf dieser Zeit nicht alle Baugruppen parametrierbar sind, ist der Istausbau ungleich dem Sollausbau.

#### Zyklus / Taktmerker

- OB1-Prozessabbild zyklisch aktualisieren: Dieser Parameter ist nicht relevant.
- Zyklusüberwachungszeit: Hier geben Sie die Zyklusüberwachungszeit in ms ein. Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, geht die CPU in STOP.  
Ursachen für eine Überschreitung:
  - Kommunikationsprozesse
  - Häufung von Alarmereignissen
  - Fehler im CPU-Programm
- Mindestzykluszeit: Dieser Parameter ist nicht relevant.
- Zyklusbelastung durch Kommunikation: Dieser Parameter ist nicht relevant.
- Größe Prozessabbild der Ein-/Ausgänge: Hier können Sie die Größe des Prozessabbilds max. 2048 für die Ein-/ Ausgabe-Peripherie festlegen.

- OB85-Aufruf bei Peripheriezugriffsfehler: Sie können die voreingestellte Reaktion der CPU bei Peripheriezugriffsfehlern während der systemseitigen Aktualisierung des Prozessabbildes ändern. Die VIPA-CPU ist so voreingestellt, dass sie bei Peripheriezugriffsfehlern keinen OB 85 aufruft und auch keinen Eintrag im Diagnosepuffer erzeugt.
- Taktmerker: Aktivieren Sie dieses Kästchen, wenn Sie einen Taktmerker einsetzen und geben Sie die Nummer des Merkerbytes ein.



*Das gewählte Merkerbyte kann nicht für die Zwischenspeicherung von Daten genutzt werden.*

### Remanenz

- Anzahl Merkerbytes ab MB0: Die Anzahl der remanenten Merkerbytes ab Merkerbyte 0 können Sie hier angeben.
- Anzahl S7-Timer ab T0: Hier tragen Sie die Anzahl der remanenten S7-Timer ab T0 ein.
- Anzahl S7-Zähler ab Z0: Tragen Sie die Anzahl der remanenten S7-Zähler ab Z0 hier ein.
- Bereiche: Diese Parameter sind nicht relevant.

### Alarmer

- Priorität: Hier werden die Prioritäten angezeigt, nach denen der entsprechende Alarm-OB (Prozessalarm, Verzögerungsalarm, Asynchronfehleralarm) bearbeitet wird.

### Uhrzeitalarmer

- Priorität: Die Priorität ist nicht änderbar.
- Aktiv: Bei aktiviertem Kästchen, wird der Uhrzeitalarm-OB bei einem Neustart automatisch gestartet.
- Ausführung: Hier wählen Sie aus, wie oft die Alarmer ausgeführt werden sollen. Die Intervalle von minütlich bis jährlich beziehen sich auf die Einstellungen unter *Startdatum* und *Uhrzeit*.
- Startdatum/Uhrzeit: Hier geben Sie an, wann der Uhrzeitalarm zum ersten Mal ausgeführt werden soll.
- Teilprozessabbild: Dieser Parameter wird nicht unterstützt.

### Weckalarmer

- Priorität: Hier können Sie die Prioritäten bestimmen, nach denen der entsprechende Weckalarm-OB bearbeitet werden soll. Mit Priorität "0" wählen Sie den entsprechenden OB ab.
- Ausführung: Geben Sie die Zeitabstände in ms an, in denen die Weckalarm-OBs bearbeitet werden. Startzeitpunkt ist der Betriebszustandwechsel von STOP nach RUN.
- Phasenverschiebung: Geben Sie hier eine Zeit in ms an, um welche der tatsächliche Ausführungszeitpunkt des Weckalarms verzögert werden soll. Dies ist sinnvoll, wenn mehrere Weckalarmer aktiv sind. Mit der *Phasenverschiebung* können diese über den Zyklus hinweg verteilt werden.
- Teilprozessabbild: Dieser Parameter wird nicht unterstützt.

**Schutz**

- Schutzstufe: Hier können Sie eine von 3 Schutzstufen einstellen, um die CPU vor unbefugtem Zugriff zu schützen.
  - *Schutzstufe 1 (voreingestellt):*  
kein Passwort parametrierbar; keine Einschränkungen
  - *Schutzstufe 2 mit Passwort:*  
Kenntnis des Passworts: lesender und schreibender Zugriff  
Unkenntnis des Passworts: nur lesender Zugriff.
  - *Schutzstufe 3:*  
Kenntnis des Passworts: lesender und schreibender Zugriff  
Unkenntnis des Passworts: weder lesender noch schreibender Zugriff

## 5.9 Projekt transferieren

**Übersicht**

Sie haben folgende Möglichkeiten für den Projekt-Transfer in die CPU:

- Transfer über MPI
- Transfer über Ethernet
- Transfer über MMC

### 5.9.1 Transfer über MPI

**Allgemein**

Für den Transfer über MPI besitzt die CPU folgende Schnittstelle:

- X2: MPI-Schnittstelle

**Netz-Struktur**

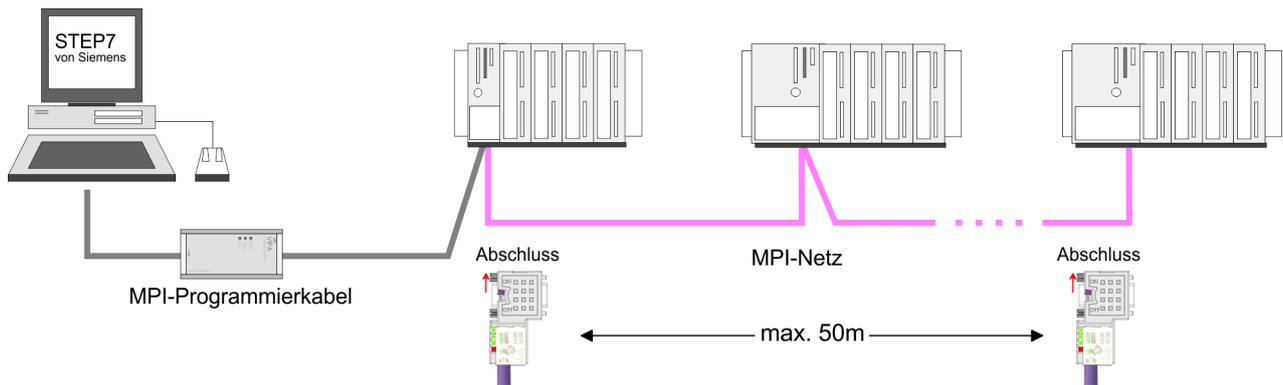
Der Aufbau eines MPI-Netzes gleicht elektrisch dem Aufbau eines PROFIBUS-Netzes. Das heißt, es gelten dieselben Regeln und Sie verwenden für beide Netze die gleichen Komponenten zum Aufbau. Die einzelnen Teilnehmer werden über Busanschlussstecker und PROFIBUS-Kabel verbunden. Bitte beachten Sie hierbei für die CPU 312-5BE13, dass die Gesamtausdehnung des MPI-Netzes 50m nicht überschreitet. Defaultmäßig wird das MPI-Netz mit 187,5kbaud betrieben. VIPA-CPU's werden mit der MPI-Adresse 2 ausgeliefert.

**MPI-Programmierkabel**

Die MPI-Programmierkabel erhalten Sie in verschiedenen Varianten von VIPA. Die Kabel bieten einen RS232- bzw. USB-Anschluss für den PC und einen busfähigen RS485-Anschluss für die CPU. Aufgrund des RS485-Anschlusses dürfen Sie die MPI-Programmierkabel direkt auf einen an der RS485-Buchse schon gesteckten Stecker aufstecken. Jeder Busteilnehmer identifiziert sich mit einer eindeutigen Adresse am Bus, wobei die Adresse 0 für Programmiergeräte reserviert ist.

**Abschlusswiderstand**

Eine Leitung muss mit ihrem Wellenwiderstand abgeschlossen werden. Hierzu schalten Sie den Abschlusswiderstand am ersten und am letzten Teilnehmer eines Netzes oder eines Segments zu. Achten Sie darauf, dass die Teilnehmer, an denen der Abschlusswiderstand zugeschaltet ist, immer mit Spannung versorgt sind. Ansonsten kann es zu Störungen auf dem Bus kommen.



### Vorgehensweise Transfer über MPI- Schnittstelle

1. ▶ Verbinden Sie Ihren PC über ein MPI-Programmierkabel mit der MPI-Buchse Ihrer CPU.
2. ▶ Laden Sie im Siemens SIMATIC Manager Ihr Projekt.
3. ▶ Wählen Sie im Menü "Extras → PG/PC-Schnittstelle einstellen".
4. ▶ Wählen Sie in der Auswahlliste "PC Adapter (MPI)" aus; ggf. müssen Sie diesen erst hinzufügen und klicken Sie auf [Eigenschaften].
5. ▶ Stellen Sie im Register MPI die Übertragungsparameter Ihres MPI-Netzes ein und geben Sie eine gültige Adresse an.
6. ▶ Wechseln Sie in das Register *Lokaler Anschluss*.
7. ▶ Geben Sie den COM-Port des PCs an und stellen Sie für Ihr MPI-Programmierkabel die Übertragungsrate 38400Baud ein.
8. ▶ Mit "Zielsystem → Laden in Baugruppe" können Sie Ihr Projekt über MPI in die CPU übertragen und mit "Zielsystem → RAM nach ROM kopieren" auf einer MMC sichern, falls diese gesteckt ist.

### 5.9.2 Transfer über Ethernet

Die CPU besitzt für den Transfer über Ethernet folgende Schnittstelle:

- X5: Ethernet-PG/OP-Kanal

#### Initialisierung

Damit Sie auf die Ethernet-Schnittstelle online zugreifen können, müssen Sie dieser durch die "Initialisierung" bzw. "Urtaufe" IP-Adress-Parameter zuweisen. ↪ "Hardware-Konfiguration - Ethernet-PG/OP-Kanal" auf Seite 51

#### Transfer

1. ▶ Für den Transfer verbinden Sie, wenn nicht schon geschehen, die entsprechende Ethernet-Buchse mit Ihrem Ethernet.
2. ▶ Öffnen Sie Ihr Projekt im Siemens SIMATIC Manager.
3. ▶ Stellen Sie über "Extras → PG/PC-Schnittstelle" den Zugriffsweg "TCP/IP → Netzwerkkarte ...." ein.

4. ▶ Gehen Sie auf *"Zielsystem → Laden in Baugruppe"* es öffnet sich das Dialogfenster *"Zielbaugruppe auswählen"*. Wählen Sie die Zielbaugruppe aus und geben Sie als Teilnehmeradresse die IP-Adress-Parameter des entsprechenden Ethernet-Schnittstelle an. Sofern keine neue Hardware-Konfiguration in die CPU übertragen wird, wird die hier angegebene Ethernet-Verbindung dauerhaft als Transferkanal im Projekt gespeichert.
5. ▶ Starten Sie mit [OK] den Transfer.



*Systembedingt kann es zu einer Meldung kommen, dass sich die projektierte von der Zielbaugruppe unterscheidet. Quittieren Sie diese Meldung mit [OK].*

*→ Ihr Projekt wird übertragen und kann nach der Übertragung in der CPU ausgeführt werden.*

### 5.9.3 Transfer über MMC

Die MMC (**Memory Card**) dient als externes Speichermedium. Es dürfen sich mehrere Projekte und Unterverzeichnisse auf einer MMC befinden. Bitte beachten Sie, dass sich Ihre aktuelle Projektierung im Root-Verzeichnis befindet und einen der folgenden Dateinamen hat:

- S7PROG.WLD
- AUTOLOAD.WLD

Mit *"Datei → Memory-Card-Datei → Neu"* können Sie im Siemens SIMATIC Manager eine WLD-Datei erzeugen. Danach kopieren Sie aus dem Baustein-Ordner Ihres Projekts alle Bausteine und die *Systemdaten* in die WLD-Datei.

#### Transfer MMC → CPU

Das Übertragen des Anwenderprogramms von der MMC in die CPU erfolgt je nach Dateiname nach *Urlöschen* oder nach *PowerON*.

- *S7PROG.WLD* wird nach *Urlöschen* von der MMC gelesen.
- *AUTOLOAD.WLD* wird nach *NetzEIN* von der MMC gelesen.

Das Blinken der MC-LED der CPU kennzeichnet den Übertragungsvorgang. Bitte beachten Sie, dass Ihr Anwenderspeicher ausreichend Speicherplatz für Ihr Anwenderprogramm bietet, ansonsten wird Ihr Anwenderprogramm unvollständig geladen und die SF-LED leuchtet.

#### Transfer CPU → MMC

Bei einer in der CPU gesteckten MMC wird durch einen Schreibbefehl der Inhalt des batteriegepufferten RAMs als *S7PROG.WLD* auf die MMC übertragen.

Den Schreibbefehl starten Sie aus dem Siemens SIMATIC Manager auf Bausteinebene über *"Zielsystem → RAM nach ROM kopieren"*. Während des Schreibvorgangs blinkt die MC-LED. Erlischt die LED, ist der Schreibvorgang beendet.

Soll dieses Projekt automatisch nach einem *NetzEIN* von der MMC geladen werden, so müssen Sie dieses auf der MMC in *AUTOLOAD.WLD* umbenennen.

**Kontrolle des Transfer-  
vorgangs**

Nach einem MMC-Zugriff erfolgt ein Diagnose-Eintrag der CPU. Zur Anzeige der Diagnoseeinträge gehen Sie im Siemens SIMATIC Manager auf "Zielsystem → Baugruppenzustand". Über das Register "Diagnosepuffer" gelangen Sie in das Diagnosefenster.

Nähere Informationen zu den Ereignis-IDs ↪ "Übersicht der Ereignis-IDs" auf Seite 76.

**5.10 Zugriff auf integrierte Web-Seite**

**Zugriff auf Web-Seite**

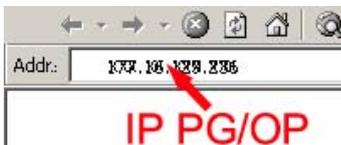
Über die IP-Adresse des Ethernet-PG/OP-Kanals steht Ihnen eine Web-Seite zur Verfügung, die Sie mit einem Internet-Browser aufrufen können. Auf der Web-Seite finden Sie Informationen zu Firmwarestand, aktuelle Zyklus-Zeiten usw. Mit dem MMC-Cmd WEBPAGE wird der aktuelle Inhalt der Web-Seite auf MMC gespeichert.  
↪ Kapitel 5.18 "MMC-Cmd - Autobefehle" auf Seite 73

**Voraussetzung**

Es wird vorausgesetzt, dass zwischen dem PC mit Internet-Browser und der CPU 312-5BE13 eine Verbindung über den PG/OP-Kanal besteht. Dies können Sie testen über Ping auf die IP-Adresse des Ethernet-PG/OP-Kanals.

**Web-Seite**

Der Zugriff auf die Web-Seite erfolgt über die IP-Adresse des Ethernet-PG/OP-Kanals. Die Web-Seite dient ausschließlich der Informationsausgabe. Die angezeigten Werte können nicht geändert werden.



**CPU mit Ethernet-PG/OP**

<b>Slot 100</b>		
VIPA 312-5BE13 V.... Px000135.pkg, SERIALNUMBER 05439		Best.-Nr., Firmware-Vers., Package, Serien-Nr.
SUPPORTDATA : PRODUCT V3420, HARDWARE V0114, 5679H-V20 , HX000027.110 , Bx000227 V6420, Ax000086 V1200, Ax000056 V0220, fx000007.wld V1140, FlashFileSystem : V102		Angaben für den Support
Memorysize(Bytes):LoadMem:LoadMem : 524288, Work- MemCode : 26214, WorkMemData : 26214		Angaben zum Speicherausbau Ladespeicher, Arbeitsspeicher (Code/Daten)
OnBoardEthernet : MacAddress : 0020D50144C1, IP-Add- ress : 172.20.120.62, SubnetMask : 255.255.255.0, Gateway : 172.20.120.62		Ethernet-PG/OP: Adressan- gaben
Cpu state : Run		CPU-Statusangabe
FunctionRS485 X2/COM1: MPI FunctionRS485 X3/COM2: PtP		Betriebsart RS485 PtP: Punkt zu Punkt-Betrieb

Zugriff auf integrierte Web-Seite

Slot 100	
Cycletime [microseconds] : min=0 cur=770 ave=750 max=878	CPU-Zykluszeit: min= minimale cur= aktuelle max= maximale
MCC-Trial-Time: 70:23	Verbleibende Zeit in hh:mm bis bei gezogener MCC der Erweiterungsspeicher wieder deaktiviert wird.
ArmLoad [percent] : cur=67, max=70	Angaben für den Support
PowerCycleHxRetries : 29, 0, 0, 0, 0	
AutoCompress activated	

Slot 202		CPU-Komponente: Digitale E/As
VIPA DI16/DO8 V3.2.9,SUPPORTDATA:PRODUCT...	Name, Firmware-Version, Modul-Typ	
SUPPORTDATA: PRODUCT V3290, Module Type ...	Angaben für den Support	
Address Input 124...125	Projektierte Eingabe-Basis-Adressen	
Address Output 124...124	Projektierte Ausgabe-Basis-Adressen	

Slot 204		CPU-Komponente: Zähler
VIPA 2 COUNTERS V3.2.9,	Name, Firmware-Version, Modul-Typ	
SUPPORTDATA: PRODUCT V3290, Module Type ...	Angaben für den Support	
Address Input 768...783	Projektierte Eingabe-Basis-Adressen	
Address Output 768...783	Projektierte Ausgabe-Basis-Adressen	

### Standard Bus

Standard Bus		Module am Standard-Bus
8 Bit Mode	Angaben für den Support	

## 5.11 Betriebszustände

### 5.11.1 Übersicht

Die CPU kennt 4 Betriebszustände:

- Betriebszustand STOP
- Betriebszustand ANLAUF
- Betriebszustand RUN
- Betriebszustand HALT

In den Betriebszuständen ANLAUF und RUN können bestimmte Ereignisse auftreten, auf die das Systemprogramm reagieren muss. In vielen Fällen wird dabei ein für das Ereignis vorgesehener Organisationsbaustein als Anwenderschnittstelle aufgerufen.

#### Betriebszustand STOP

- Das Anwenderprogramm wird nicht bearbeitet.
- Hat zuvor eine Programmbearbeitung stattgefunden, bleiben die Werte von Zählern, Zeiten, Merkern und des Prozessabbilds beim Übergang in den STOP-Zustand erhalten.
- Die Befehlsausgabe ist gesperrt, d.h. alle digitalen Ausgaben sind gesperrt.
- RUN-LED aus
- STOP-LED an

#### Betriebszustand ANLAUF

- Während des Übergangs von STOP nach RUN erfolgt ein Sprung in den Anlauf-Organisationsbaustein OB 100. Der Ablauf des OBs wird zeitlich nicht überwacht. Im Anlauf-OB können weitere Bausteine aufgerufen werden.
- Beim Anlauf sind alle digitalen Ausgaben gesperrt, d.h. die Befehlsausgabesperre ist aktiv.
- RUN-LED blinkt, solange der OB 100 bearbeitet wird und für mindestens 3s, auch wenn der Anlauf kürzer ist oder die CPU aufgrund eines Fehler in STOP geht. Dies zeigt den Anlauf an.
- STOP-LED aus

Wenn die CPU einen Anlauf fertig bearbeitet hat, geht Sie in den Betriebszustand RUN über.

#### Betriebszustand RUN

- Das Anwenderprogramm im OB 1 wird zyklisch bearbeitet, wobei zusätzlich alarmgesteuert weitere Programmteile eingeschachtelt werden können.
- Alle im Programm gestarteten Zeiten und Zähler laufen und das Prozessabbild wird zyklisch aktualisiert.
- Das BASP-Signal (Befehlsausgabesperre) wird deaktiviert, d.h. alle Ausgänge sind freigegeben.
- RUN-LED an
- STOP-LED aus

#### Betriebszustand HALT

Die CPU bietet Ihnen die Möglichkeit bis zu 3 Haltepunkte zur Programmdiagnose einzusetzen. Das Setzen und Löschen von Haltepunkten erfolgt in Ihrer Programmierumgebung. Sobald ein Haltepunkt erreicht ist, können Sie schrittweise Ihre Befehlszeilen abarbeiten.

**Voraussetzung**

Für die Verwendung von Haltepunkten müssen folgende Voraussetzungen erfüllt sein:

- Das Testen im Einzelschrittmodus ist in AWL möglich, ggf. über "Ansicht → AWL" Ansicht in AWL ändern
- Der Baustein muss online geöffnet und darf nicht geschützt sein.

**Vorgehensweise zur Arbeit mit Haltepunkten**

1. ► Blenden Sie über "Ansicht → Haltepunktleiste" diese ein.
2. ► Setzen Sie Ihren Cursor auf die Anweisungszeile, in der ein Haltepunkt gesetzt werden soll.
3. ► Setzen Sie den Haltepunkt mit "Test → Haltepunkt setzen".  
⇒ Die Anweisungszeile wird mit einem Kreisring markiert.
4. ► Zur Aktivierung des Haltepunkts gehen Sie auf "Test → Haltepunkt" aktiv.  
⇒ Der Kreisring wird zu einer Kreisfläche.
5. ► Bringen Sie Ihre CPU in RUN.  
⇒ Wenn Ihr Programm auf den Haltepunkt trifft, geht Ihre CPU in den Zustand HALT über, der Haltepunkt wird mit einem Pfeil markiert und die Registerinhalte werden eingeblendet.
6. ► Nun können Sie mit "Test → Nächste Anweisung ausführen" schrittweise Ihren Programmcode durchfahren oder über "Test → Fortsetzen" Ihre Programmausführung bis zum nächsten Haltepunkt fortsetzen.
7. ► Mit "Test → (Alle) Haltepunkte löschen" können Sie (alle) Haltepunkte wieder löschen.

**Verhalten im Betriebszustand HALT**

- RUN-LED blinkt und die STOP-LED leuchtet.
- Die Bearbeitung des Codes ist angehalten. Alle Ablafebeneen werden nicht weiterbearbeitet.
- Alle Zeiten werden eingefroren.
- Echtzeituhr läuft weiter.
- Ausgänge werden abgeschaltet (BASP ist aktiv).
- Projektierte CP-Verbindungen bleiben bestehen.



*Der Einsatz von Haltepunkten ist immer möglich. Eine Umschaltung in die Betriebsart Testbetrieb ist nicht erforderlich.*

*Sobald Sie mehr als 2 Haltepunkte gesetzt haben, ist eine Einzelschrittbearbeitung nicht mehr möglich.*

### 5.11.2 Funktionssicherheit

Die CPUs besitzen Sicherheitsmechanismen, wie einen Watchdog (100ms) und eine parametrierbare Zykluszeitüberwachung (parametrierbar min. 1ms), die im Fehlerfall die CPU stoppen bzw. einen RESET auf der CPU durchführen und diese in einen definierten STOP-Zustand versetzen. Die CPUs von VIPA sind funktionssicher ausgelegt und besitzen folgende Systemeigenschaften:

Ereignis	betrifft	Effekt
RUN → STOP	allgemein	BASP (Befehls-Ausgabe-Sperre) wird gesetzt.
	zentrale digitale Ausgänge	Die Ausgänge werden abgeschaltet.
	zentrale analoge Ausgänge	Die Ausgänge werden abgeschaltet. <ul style="list-style-type: none"> <li>■ Spannungsausgänge geben 0V aus</li> <li>■ Stromausgänge 0...20mA geben 0mA aus</li> <li>■ Stromausgänge 4...20mA geben 4mA aus</li> </ul> Falls parametriert können auch Ersatzwerte ausgegeben werden.
	dezentrale Ausgänge	Verhalten wie bei zentralen digitalen/analogen Ausgängen
	dezentrale Eingänge	Die Eingänge werden von der dezentralen Station zyklisch gelesen und die aktuellen Werte zur Verfügung gestellt.
STOP → RUN bzw. NetzEin	allgemein	Zuerst wird das PAE gelöscht, danach erfolgt der Aufruf des OB 100. Nachdem dieser abgearbeitet ist, wird das BASP zurückgesetzt und der Zyklus gestartet mit: PAA löschen → PAE lesen → OB 1.
	dezentrale Eingänge	Die Eingänge werden von der dezentralen Station einmalig gelesen und die aktuellen Werte zur Verfügung gestellt.
RUN	allgemein	Der Programmablauf ist zyklisch und damit vorhersehbar: PAE lesen → OB 1 → PAA schreiben.

PAE: Prozessabbild der Eingänge, PAA: Prozessabbild der Ausgänge

## 5.12 Urlöschen

### Übersicht

Beim Urlöschen wird der komplette Anwenderspeicher gelöscht. Ihre Daten auf der Memory Card bleiben erhalten. Sie haben 2 Möglichkeiten zum Urlöschen:

- Urlöschen über Betriebsartenschalter
- Urlöschen über Konfigurations-Software wie z.B. Siemens SIMATIC Manager



*Vor dem Laden Ihres Anwenderprogramms in Ihre CPU sollten Sie die CPU immer urlöschen, um sicherzustellen, dass sich kein alter Baustein mehr in Ihrer CPU befindet.*

**Urlöschen über Betriebsartenschalter****Voraussetzung**

➔ Ihre CPU muss sich im STOP-Zustand befinden. Stellen Sie hierzu den CPU-Betriebsartenschalter auf "STOP".

⇒ Die STOP-LED leuchtet.

**Urlöschen**

**1.** ➔ Bringen Sie den Betriebsartenschalter in Stellung MRES und halten Sie ihn ca. 3 Sekunden.

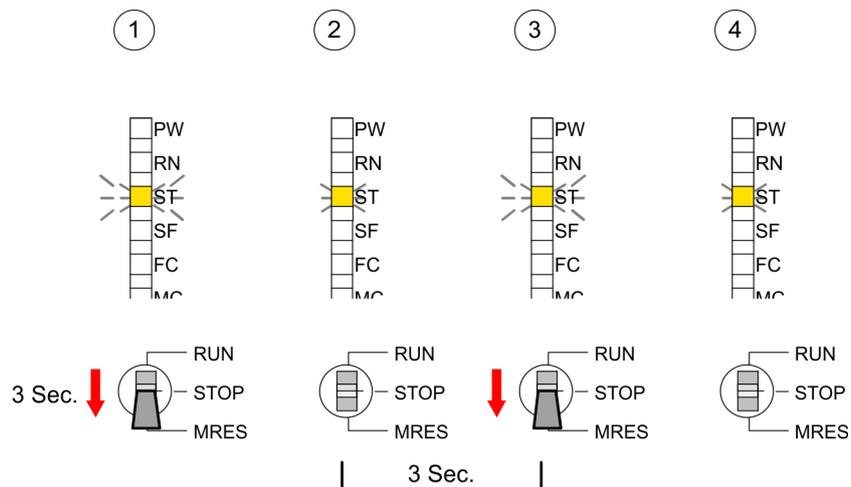
⇒ Die STOP-LED geht von Blinken über in Dauerlicht.

**2.** ➔ Bringen Sie den Betriebsartenschalter in Stellung STOP und innerhalb von 3 Sekunden kurz in MRES dann wieder auf STOP.

⇒ Die STOP-LED blinkt (Urlösch-Vorgang).

**3.** ➔ Das Urlöschen ist abgeschlossen, wenn die STOP-LED in Dauerlicht übergeht.

⇒ Die STOP-LED leuchtet. Die nachfolgende Abbildung zeigt nochmals die Vorgehensweise:

**Urlöschen über Siemens SIMATIC Manager**

- Voraussetzung: Ihre CPU muss sich im STOP-Zustand befinden. Mit dem Menübefehl "*Zielsystem* → *Betriebszustand*" bringen Sie Ihre CPU in STOP.
- Urlöschen: Über den Menübefehl "*Zielsystem* → *Urlöschen*" fordern Sie das Urlöschen an. In dem Dialogfenster können Sie, wenn noch nicht geschehen, Ihre CPU in STOP bringen und das Urlöschen starten. Während des Urlöschvorgangs blinkt die STOP-LED. Geht die STOP-LED in Dauerlicht über, ist der Urlöschvorgang abgeschlossen.

**Automatisch nachladen**

Falls nach dem Urlöschen auf der MMC ein Projekt S7PROG.WLD vorhanden ist, versucht die CPU dieses von der MMC neu zu laden. → Die MC-LED leuchtet. Nach dem Nachladen erlischt die LED. Abhängig von der Einstellung des Betriebsartenschalters bleibt die CPU in STOP bzw. geht in RUN.

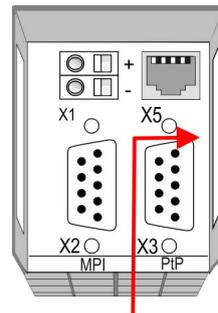
## Rücksetzen auf Werks-einstellung

Das *Rücksetzen auf Werkseinstellung* löscht das interne RAM der CPU vollständig und bringt diese zurück in den Auslieferungszustand. Bitte beachten Sie, dass hierbei auch die MPI-Adresse defaultmäßig auf 2 zurückgestellt wird! ↪ *Kapitel 5.14 "Rücksetzen auf Werkseinstellung" auf Seite 68*

## 5.13 Firmwareupdate

### Übersicht

- Sie haben die Möglichkeit unter Einsatz einer MMC für die CPU und ihre Komponenten ein Firmwareupdate durchzuführen. Hierzu muss sich in der CPU beim Hochlauf eine entsprechend vorbereitete MMC befinden.
- Damit eine Firmwaredatei beim Hochlauf erkannt und zugeordnet werden kann, ist für jede update-fähige Komponente und jeden Hardware-Ausgabestand ein pkg-Dateiname reserviert, der mit "px" beginnt und sich in einer 6-stelligen Ziffer unterscheidet. Bei jedem updatefähigen Modul finden Sie den pkg-Dateinamen unter der Frontklappe auf einem Aufkleber auf der rechten Seite des Moduls.
- Nach NetzEIN und CPU-STOP prüft die CPU, ob eine \*.pkg-Datei auf der MMC vorhanden ist. Wenn sich diese Firmware-Version von der zu überschreibenden Firmware-Version unterscheidet, zeigt die CPU dies über LED-Blinken an und sie können die Firmware über eine Updateanforderung installieren.



**Firmware package  
and version**

### Aktuelle Firmware auf [www.vipa.com](http://www.vipa.com)

Die aktuellsten Firmwarestände finden Sie auf [www.vipa.com](http://www.vipa.com) im Service-Bereich. Beispielsweise sind für den Firmwareupdate der CPU 312-5BE13 und Ihrer Komponenten für den Ausgabestand 1 folgende Dateien erforderlich:

- 312-5BE13, Ausgabestand 1: Px000135.pkg

**VORSICHT!**

Beim Aufspielen einer neuen Firmware ist äußerste Vorsicht geboten. Unter Umständen kann Ihre CPU unbrauchbar werden, wenn beispielsweise während der Übertragung die Spannungsversorgung unterbrochen wird oder die Firmware-Datei fehlerhaft ist. Setzen Sie sich in diesem Fall mit der VIPA-Hotline in Verbindung!

Bitte beachten Sie auch, dass sich die zu überschreibende Firmware-Version von der Update-Version unterscheidet, ansonsten erfolgt kein Update.

### Firmwarestand des SPEED7-Systems über Web-Seite ausgeben

Die CPU hat eine Web-Seite integriert, die auch Informationen zum Firmwarestand der SPEED7-Komponenten bereitstellt. Über den Ethernet-PG/OP-Kanal haben Sie Zugriff auf diese Web-Seite. Zur Aktivierung des PG/OP-Kanals müssen Sie diesem IP-Parameter zuweisen. Dies kann im Siemens SIMATIC Manager entweder über eine Hardware-Konfiguration erfolgen, die Sie über MMC bzw. MPI einspielen oder über Ethernet durch Angabe der MAC-Adresse unter "Zielsystem → Ethernet-Adresse vergeben". Danach können Sie mit einem Web-Browser über die angegebene IP-Adresse auf den PG/OP-Kanal zugreifen.

↪ Kapitel 5.10 "Zugriff auf integrierte Web-Seite" auf Seite 59

### CPU-Firmwarestand über Baugruppenzustand ermitteln

1. ▶ Stellen Sie eine Online-Verbindung zu Ihrer CPU her.
2. ▶ Zur Anzeige des Baugruppenzustands gehen Sie in Ihrem Siemens SIMATIC Manager auf "Zielsystem → Baugruppenzustand".
3. ▶ Über das Register "Allgemein" gelangen Sie in das Fenster zur Anzeige des Hard- und Firmware-Ausgabestands.

⇒ Aus softwaretechnischen Gründen ergibt sich für die CPU 312-5BE13 von VIPA eine Änderung gegenüber der CPU 312C von Siemens:

Bezeichnung:	CPU 312C	Systemkennung:	SIMATIC 300									
Name:	CPU 312C											
Version:	<table border="1"> <thead> <tr> <th>Best.-Nr./Bezeichn.</th> <th>Komponente</th> <th>Ausgabestand</th> </tr> </thead> <tbody> <tr> <td>6ES7 312-5BE03</td> <td>Hardware</td> <td>1</td> </tr> <tr> <td>VIPA 312-5BE13-0100</td> <td>Firmware</td> <td>V3.6.0</td> </tr> </tbody> </table>			Best.-Nr./Bezeichn.	Komponente	Ausgabestand	6ES7 312-5BE03	Hardware	1	VIPA 312-5BE13-0100	Firmware	V3.6.0
Best.-Nr./Bezeichn.	Komponente	Ausgabestand										
6ES7 312-5BE03	Hardware	1										
VIPA 312-5BE13-0100	Firmware	V3.6.0										
	1    2 3		4									

- 1 VIPA-Best.-Nr. (VIPA 312-5BE13)
- 2 Hardware-Ausgabestand (01)
- 3 Interne Hardware-Version (00)
- 4 Firmwareversion (V3.6.0)



Die CPUs von VIPA unterstützen alle Register des Baugruppenzustands. Eine nähere Beschreibung der einzelnen Register finden Sie in der Online-Hilfe Ihres Siemens SIMATIC Manager.

### Firmware laden und auf MMC übertragen

- Gehen Sie auf [www.vipa.com](http://www.vipa.com)
- Klicken Sie auf "Service → Download → Firmware".
- Navigieren Sie über "System 300S → CPU" zu Ihrer CPU und laden Sie die zip-Datei auf Ihren PC.
- Entpacken Sie die zip-Datei und kopieren Sie die extrahierten pkg-Dateien auf Ihre MMC.



### VORSICHT!

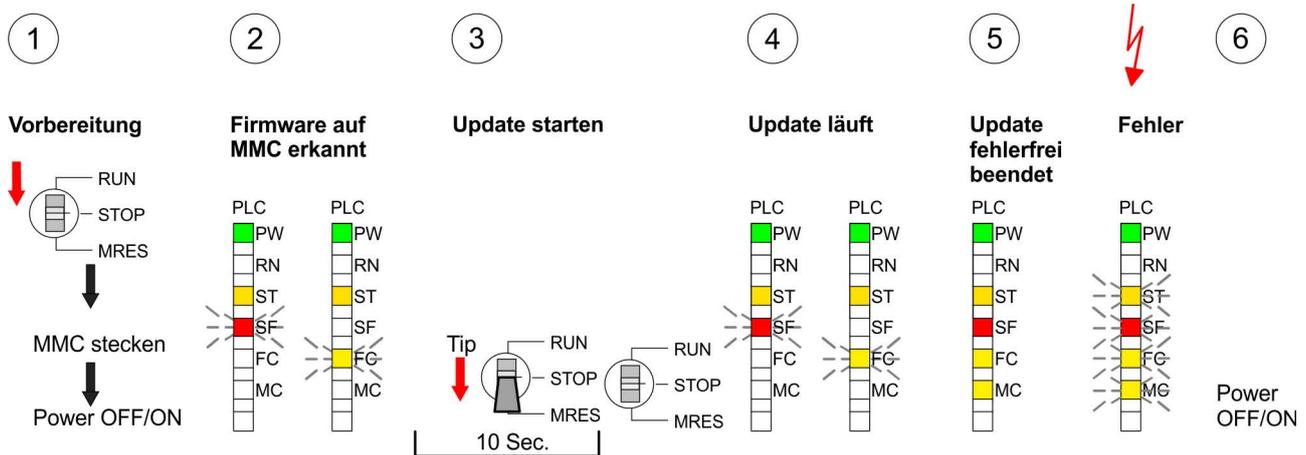
Beim Firmwareupdate wird automatisch ein Urlöschen durchgeführt. Sollte sich Ihr Programm nur im Ladespeicher der CPU befinden, so wird es hierbei gelöscht! Sichern Sie Ihr Programm, bevor Sie ein Firmwareupdate durchführen! Auch sollten Sie nach dem Firmwareupdate ein "Rücksetzen auf Werkseinstellung" durchführen.  
 ↪ Kapitel 5.14 "Rücksetzen auf Werkseinstellung" auf Seite 68

### Firmware von MMC in CPU übertragen

1. ▶ Bringen Sie den Betriebsartenschalter Ihrer CPU in Stellung STOP. Schalten Sie die Spannungsversorgung aus. Stecken Sie die MMC mit den Firmware-Dateien in die CPU. Achten Sie hierbei auf die Steckrichtung der MMC. Schalten Sie die Spannungsversorgung ein.
2. ▶ Nach einer kurzen Hochlaufzeit zeigt das abwechselnde Blinken der LEDs SF und FC an, dass auf der MMC mindestens eine aktuellere Firmware-Datei gefunden wurde.
3. ▶ Sie starten die Übertragung der Firmware, sobald Sie innerhalb von 10s den Betriebsartenschalter kurz nach MRES tippen und dann den Schalter in der STOP-Position belassen.
4. ▶ Während des Update-Vorgangs blinken die LEDs SF und FC abwechselnd und die MC-LED leuchtet. Dieser Vorgang kann mehrere Minuten dauern.
5. ▶ Das Update ist fehlerfrei beendet, wenn die LEDs PW, ST, SF, FC und MC leuchten. Blinken diese schnell, ist ein Fehler aufgetreten.

6. ➤ Schalten Sie die Spannungsversorgung aus und wieder ein. Jetzt prüft die CPU, ob noch weitere Firmware-Updates durchzuführen sind. Ist dies der Fall, blinken, wiederum nach einer kurzen Hochlaufzeit, die LEDs SF und FC. Fahren Sie mit Punkt 3 fort.

⇒ Blinken die LEDs nicht, ist das Firmware-Update abgeschlossen. Führen Sie jetzt wie nachfolgend beschrieben ein *Rücksetzen auf Werkseinstellungen* durch. Danach ist die CPU wieder einsatzbereit.



## 5.14 Rücksetzen auf Werkseinstellung

### Vorgehensweise

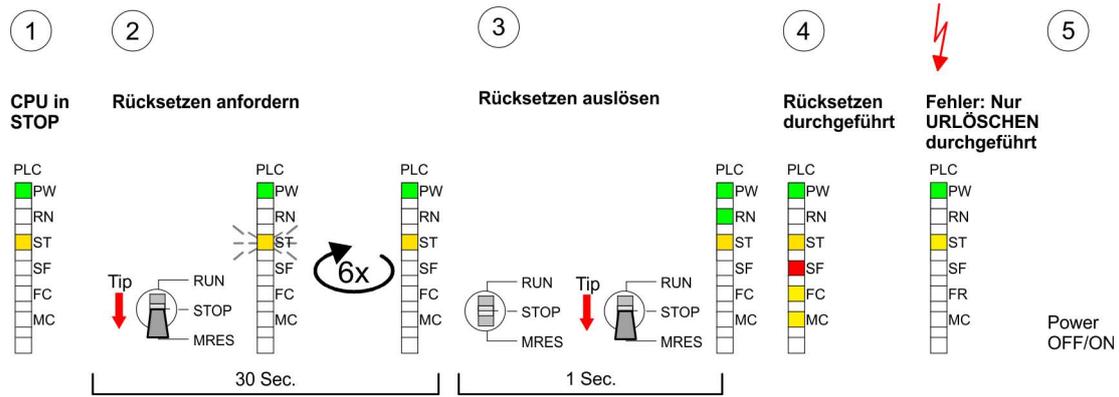
Die folgende Vorgehensweise löscht das interne RAM der CPU vollständig und bringt diese zurück in den Auslieferungszustand.

Bitte beachten Sie, dass hierbei auch die MPI-Adresse auf 2 und die IP-Adresse des Ethernet-PG/OP-Kanals auf 0.0.0.0 zurückgestellt wird!

Sie können auch das Rücksetzen auf Werkseinstellung mit dem MMC-Command `FACTORY_RESET` ausführen. ➤ Kapitel 5.18 "MMC-Command - Autobefehle" auf Seite 73

- Bringen Sie die CPU in STOP.
- Drücken Sie den Betriebsartenschalter für ca. 30 Sekunden nach unten in Stellung MRES. Hierbei blinkt die STOP-LED. Nach ein paar Sekunden leuchtet die STOP-LED. Die STOP-LED wechselt jetzt von Leuchten in Blinken. Zählen Sie, wie oft die STOP-LED leuchtet.
- Nach dem 6. Mal Leuchten der STOP-LED lassen Sie den Reset-Schalter wieder los, um ihn nochmals kurzzeitig nach unten zu drücken. Jetzt leuchtet die grüne RUN-LED einmal auf. Das bedeutet, dass das RAM vollständig gelöscht ist.
- Zur Bestätigung des Rücksetzvorgangs leuchten die LEDs PW, ST, SF, FC und MC. Leuchtet diese nicht, wurde nur Umräumen ausgeführt und das Rücksetzen auf Werkseinstellung ist fehlgeschlagen. In diesem Fall können Sie den Vorgang wiederholen. Das Rücksetzen auf Werkseinstellung wird nur dann ausgeführt, wenn die STOP-LED genau 6 Mal geleuchtet hat.
- Am Ende des Rücksetzvorgangs leuchten die LEDs PW, ST, SF, FC und MC. Danach ist die Spannungsversorgung aus- und wieder einzuschalten.

Die nachfolgende Abbildung soll die Vorgehensweise verdeutlichen:



Bitte führen Sie nach einem Firmwareupdate der CPU immer ein Rücksetzen auf Werkseinstellung durch.

## 5.15 Steckplatz für Speichermedien

### Übersicht

Auf der Frontseite der CPU befindet sich ein Steckplatz für Speichermedien. Über diesen Steckplatz können Sie eine Multimedia Card (MMC) als externes Speichermedium für Programme und Firmware stecken. Mittels vorgegebener Dateinamen können Sie die CPU veranlassen automatisch ein Projekt zu laden bzw. eine Kommandodatei auszuführen.

## Zugriff auf das Speichermedium

Zu folgenden Zeitpunkten erfolgt ein Zugriff auf ein Speichermedium:

- Nach Urlöschen
  - Die CPU prüft, ob ein Projekt mit dem Namen S7PROG.WLD vorhanden ist. Wenn ja, wird dieses automatisch geladen.
  - Die CPU prüft, ob ein Projekt mit dem Namen PROTECT.WLD mit geschützten Bausteinen vorhanden ist. Wenn ja, wird dieses automatisch geladen. Diese Bausteine verbleiben in der CPU bis zum Rücksetzen der CPU auf Werkseinstellungen oder Laden einer "leeren" PROTECT.WLD.
  - Die CPU prüft, ob eine MCC-Speichererweiterung gesteckt ist. Falls ja, erfolgt die Freigabe der Speichererweiterung, ansonsten wird eine zuvor aktivierte Speichererweiterung deaktiviert.
- Nach NetzEIN
  - Die CPU prüft, ob ein Projekt mit dem Namen AUTO-LOAD.WLD vorhanden ist. Wenn ja, wird Urlöschen durchgeführt und das Projekt automatisch geladen.
  - Die CPU prüft, ob eine Kommandodatei mit dem Namen VIPA\_CMD.MMC vorhanden ist. Wenn ja, wird die Kommandodatei geladen und die enthaltenen Befehle werden ausgeführt.
  - Nach NetzEIN und CPU-STOP prüft die CPU, ob eine \*.pkg-Datei (Firmware-Datei) vorhanden ist. Wenn ja, zeigt die CPU dies über LED-Blinken an und sie können die Firmware über eine Updateanforderung installieren.
- Einmalig im Zustand STOP
  - Wird eine Speicherkarte mit einer Kommandodatei mit dem Namen VIPA\_CMD.MMC gesteckt, so wird die Kommandodatei geladen und die enthaltenen Befehle werden ausgeführt.

## 5.16 Speichererweiterung mit MCC

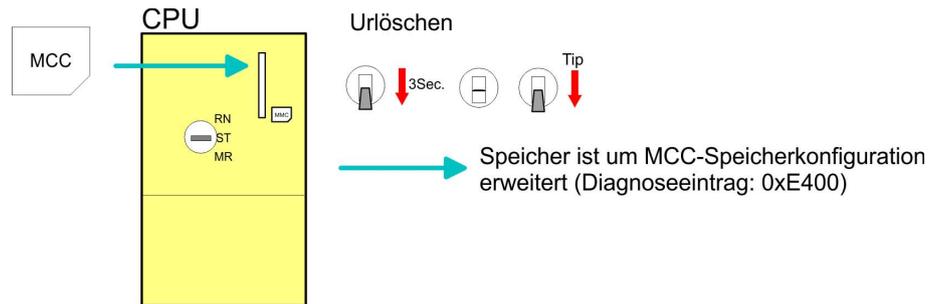
### Übersicht



Bei der CPU haben Sie die Möglichkeit den Arbeitsspeicher zu erweitern. Hierzu ist bei VIPA eine MCC Speichererweiterungskarte verfügbar. Bei der MCC handelt es sich um eine speziell vorbereitete MMC (**M**ultimedia **C**ard). Durch Stecken der MCC im MCC-Slot und anschließendem Urlöschen wird die entsprechende Speichererweiterung freigeschaltet. Es kann immer nur eine Speichererweiterung aktiviert sein. Auf der MCC befindet sich die Datei `memory.key`. Diese Datei darf weder bearbeitet noch gelöscht werden. Sie können die MCC auch als "normale" MMC zur Speicherung Ihrer Projekte verwenden.

### Vorgehensweise

Zur Erweiterung des Speichers stecken Sie die MCC in den mit "MCC" bezeichneten Kartenslot der CPU und führen Sie Urlöschen durch.



Sollte die Speichererweiterung auf der MCC den maximal erweiterbaren Speicherbereich der CPU überschreiten, wird automatisch der maximal mögliche Speicher der CPU verwendet. Den aktuellen Speicherausbau können Sie über die integrierte Web-Seite oder mit dem Siemens SIMATIC Manager über den Baugruppenzustand unter "Speicher" ermitteln.



**VORSICHT!**

Bitte beachten Sie, dass, sobald Sie eine Speichererweiterung auf Ihrer CPU durchgeführt haben, die MCC gesteckt bleiben muss. Ansonsten geht die CPU nach 72 Stunden in STOP. Auch kann die MCC nicht gegen eine MCC mit gleicher Speicherkonfiguration getauscht werden.

**Verhalten**

Wurde die MCC-Speicherkonfiguration übernommen, finden Sie den Diagnoseeintrag 0xE400 im Diagnosepuffer der CPU.

Nach Ziehen der MCC erfolgt der Eintrag 0xE401 im Diagnosepuffer, die SF-LED leuchtet und nach 72 Stunden geht die CPU in STOP. Hier ist ein Anlauf erst wieder möglich nach Stecken der MCC oder nach Urlöschen.

Die verbleibende Zeit nach dem Ziehen der MCC können Sie jederzeit über den Parameter *MCC-Trial-Time* auf der Web-Seite ermitteln.

Nach erneutem Stecken der MCC erlischt die SF-LED und 0xE400 wird im Diagnosepuffer eingetragen. Sie können jederzeit die Speicherkonfiguration Ihrer CPU auf den ursprünglichen Zustand wieder zurücksetzen, indem Sie Urlöschen ohne MCC ausführen.

**5.17 Erweiterter Know-how-Schutz**

**Übersicht**

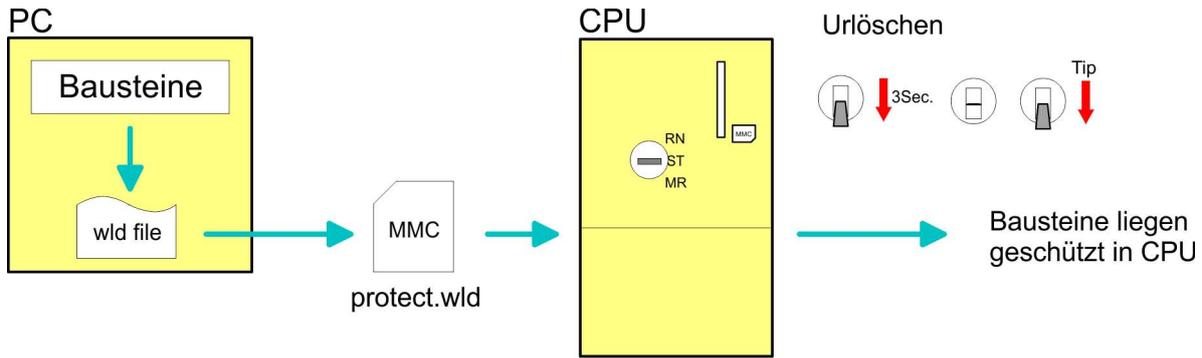
Neben dem "Standard" Know-how-Schutz besitzen die SPEED7-CPU's von VIPA einen "erweiterten" Know-how-Schutz, der einen sicheren Baustein-Schutz vor Zugriff Dritter bietet.

**Standard-Schutz**

Beim Standard-Schutz von Siemens werden auch geschützte Bausteine in das PG übertragen, aber deren Inhalt nicht dargestellt. Durch entsprechende Manipulation ist der Know-how-Schutz aber nicht sichergestellt.

**Erweiterter Schutz**

Mit dem von VIPA entwickelten "erweiterten" Know-how-Schutz besteht aber die Möglichkeit Bausteine permanent in der CPU zu speichern. Beim "erweiterten" Schutz übertragen Sie die zu schützenden Bausteine in eine WLD-Datei mit Namen protect.wld. Durch Stecken der MMC und anschließendem Urlöschen werden die in protect.wld gespeicherten Bausteine permanent in der CPU abgelegt. Geschützt werden können OBs, FBs und FCs. Beim Zurücklesen von geschützten Bausteinen in Ihr PG werden ausschließlich die Baustein-Header geladen. Der schützenswerte Baustein-Code bleibt in der CPU und kann nicht ausgelesen werden.

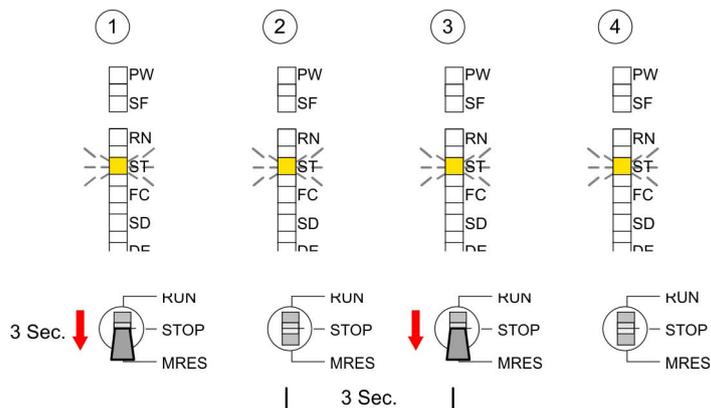


**Bausteine mit protect.wld schützen**

Erzeugen Sie in Ihrem Projekttool mit "Datei → Memory Card Datei → Neu" eine WLD-Datei und benennen Sie diese um in "protect.wld". Übertragen Sie die zu schützenden Bausteine in die Datei, indem Sie diese mit der Maus aus Ihrem Projekt in das Dateifenster von protect.wld ziehen.

**protect.wld mit Urlöschen in CPU übertragen**

Übertragen Sie die Datei protect.wld auf eine MMC-Speicherkarte, stecken Sie die MMC in Ihre CPU und führen Sie nach folgender Vorgehensweise Urlöschen durch:



Mit Urlöschen werden die in protect.wld enthaltenen Bausteine, permanent vor Zugriffen Dritter geschützt, in der CPU abgelegt.

**Schutzverhalten**

Geschützte Bausteine werden durch eine neue protect.wld überschrieben. Mit einem PG können Dritte auf geschützte Bausteine zugreifen, hierbei wird aber ausschließlich der Baustein-Header in das PG übertragen. Der schützenswerte Baustein-Code bleibt in der CPU und kann nicht ausgelesen werden.

**Geschützte Bausteine überschreiben bzw. löschen**

Sie haben jederzeit die Möglichkeit geschützte Bausteine durch gleichnamige Bausteine im RAM der CPU zu überschreiben. Diese Änderung bleibt bis zum nächsten Umräumen erhalten. Geschützte Bausteine können nur dann vom PG dauerhaft überschrieben werden, wenn diese zuvor aus der protect.wld gelöscht wurden. Durch Übertragen einer leeren protect.wld von der MMC können Sie in der CPU alle geschützten Bausteine löschen.

**Einsatz von geschützten Bausteinen**

Da beim Auslesen eines "protected" Bausteins aus der CPU die Symbol-Bezeichnungen fehlen, ist es ratsam dem Endanwender die "Bausteinshüllen" zur Verfügung zu stellen. Erstellen Sie hierzu aus allen geschützten Bausteinen ein Projekt. Löschen Sie aus diesen Bausteinen alle Netzwerke, so dass diese ausschließlich die Variablen-Definitionen in der entsprechenden Symbolik beinhalten.

**5.18 MMC-Cmd - Autobefehle**

**Übersicht**

Eine *Kommando*-Datei auf einer MMC wird unter folgenden Bedingungen automatisch ausgeführt:

- CPU befindet sich in STOP und MMC wird gesteckt
- Bei jedem Einschaltvorgang (NetzeIN)

**Kommando-Datei**

Bei der *Kommando*-Datei handelt es sich um eine Text-Datei mit einer Befehlsabfolge, die unter dem Namen **vipa\_cmd.mmc** im Root-Verzeichnis der MMC abzulegen ist. Die Datei muss mit dem 1. Befehl `CMD_START` beginnen, gefolgt von den gewünschten Befehlen (kein anderer Text) und ist immer mit dem letzten Befehl `CMD_END` abzuschließen.

Texte wie beispielsweise Kommentare nach dem letzten Befehl `CMD_END` sind zulässig, da diese ignoriert werden. Sobald eine Kommandodatei erkannt und ausgeführt wird, werden die Aktionen in der Datei `Logfile.txt` auf der MMC gespeichert. Zusätzlich finden Sie für jeden ausgeführten Befehl einen Diagnoseeintrag im Diagnosepuffer.

**Befehle**

Bitte beachten Sie, dass Sie immer Ihre Befehlsabfolge mit `CMD_START` beginnen und mit `CMD_END` beenden.

Kommando	Beschreibung	Diagnoseeintrag
CMD_START	In der ersten Zeile muss CMD_START stehen.	0xE801
	Fehlt CMD_START erfolgt ein Diagnoseeintrag	0xE8FE
WAIT1SECOND	Wartet ca. 1 Sekunde.	0xE803
WEBPAGE	Speichert die Web-Seite der CPU als Datei "webpage.htm" auf der MMC.	0xE804
LOAD_PROJECT	Ruft die Funktion "Umräumen mit Nachladen von der MMC" auf. Durch Angabe einer wld-Datei nach dem Kommando, wird diese wld-Datei nachgeladen, ansonsten wird die Datei "s7prog.wld" geladen.	0xE805

Kommando	Beschreibung	Diagnoseeintrag
SAVE_PROJECT	Speichert das Anwenderprojekt (Bausteine und Hardware-Konfiguration) auf der MMC als "s7prog.wld". Falls bereits eine Datei mit dem Namen "s7prog.wld" existiert, wird diese in "s7prog.old" umbenannt. Sollte Ihre CPU durch ein Passwort geschützt sein, so müssen Sie dies als Parameter mitliefern. Ansonsten wird kein Projekt geschrieben. Beispiel: SAVE_PROJECT password	0xE806
FACTORY_RESET	Führt "Rücksetzen auf Werkseinstellung" durch.	0xE807
DIAGBUFF	Speichert den Diagnosepuffer der CPU als Datei "diagbuff.txt" auf der MMC.	0xE80B
SET_NETWORK	Mit diesem Kommando können Sie die IP-Parameter für den Ethernet-PG/OP-Kanal einstellen. Die IP-Parameter sind in der Reihenfolge IP-Adresse, Subnetz-Maske und Gateway jeweils getrennt durch ein Komma im Format von x.x.x.x einzugeben. Wird kein Gateway verwendet, tragen Sie die IP-Adresse als Gateway ein.	0xE80E
CMD_END	In der letzten Zeile muss CMD_END stehen.	0xE802

## Beispiele

Nachfolgend ist der Aufbau einer Kommando-Datei an Beispielen gezeigt. Den jeweiligen Diagnoseeintrag finden Sie in Klammern gesetzt.

### Beispiel 1

CMD_START	Kennzeichnet den Start der Befehlsliste (0xE801)
LOAD_PROJECT proj.wld	Urlöschen und Nachladen von "proj.wld" (0xE805)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
WEBPAGE	Web-Seite als "webpage.htm" speichern (0xE804)
DIAGBUFF	Diagnosepuffer der CPU als "diagbuff.txt" speichern (0xE80B)
CMD_END	Kennzeichnet das Ende der Befehlsliste (0xE802)
... beliebiger Text ...	Texte nach dem CMD_END werden nicht mehr ausgewertet.

### Beispiel 2

CMD_START	Kennzeichnet den Start der Befehlsliste (0xE801)
LOAD_PROJECT proj2.wld	Urlöschen und Nachladen von "proj2.wld" (0xE805)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
	IP-Parameter (0xE80E)
SET_NETWORK 172.16.129.210,255.255.224.0,172.16.129.210	
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)

WEBPAGE	Web-Seite als "webpage.htm" speichern (0xE804)
DIAGBUFF	Diagnosepuffer der CPU als "diagbuff.txt" speichern (0xE80B)
CMD_END	Kennzeichnet das Ende der Befehlsliste (0xE802)
... beliebiger Text ...	Texte nach dem CMD_END werden nicht mehr ausgewertet.



Die Parameter IP-Adresse, Subnetz-Maske und Gateway erhalten Sie von Ihrem Systemadministrator.

Wird kein Gateway verwendet, tragen Sie die IP-Adresse als Gateway ein.

### 5.19 VIPA-spezifische Diagnose-Einträge

#### Einträge im Diagnosepuffer

Sie haben die Möglichkeit im Siemens SIMATIC Manager den Diagnosepuffer der CPU auszulesen. Neben den Standardeinträgen im Diagnosepuffer gibt es in den CPUs der VIPA noch zusätzliche Einträge, welche ausschließlich in Form einer Ereignis-ID angezeigt werden.

Mit dem CMD DIAGBUFF wird der aktuelle Inhalt des Diagnosepuffers auf die Speicherkarte gespeichert.



Die CPUs von VIPA unterstützen alle Register des Baugruppenzustands. Eine nähere Beschreibung der einzelnen Register finden Sie in der Online-Hilfe Ihres Siemens SIMATIC Managers.

#### Anzeige der Diagnose-einträge

Zur Anzeige der Diagnoseeinträge gehen Sie in Ihrem Siemens SIMATIC Manager auf "Zielsystem → Baugruppenzustand". Über das Register "Diagnosepuffer" gelangen Sie in das Diagnosefenster:

The screenshot shows the 'Baugruppenzustand' window in SIMATIC Manager. The 'Diagnosepuffer' register is selected. The window displays the following table:

Nr.	Uhrzeit	Datum	Ereignis
8	...	...	...
9	...	...	...
10	13:18:11:370	19.12.2011	Ereignis-ID: 16# E0CC
11	...	...	...
12	...	...	...
13	...	...	...

A red arrow points to the entry 'Ereignis-ID: 16# E0CC', which is labeled as 'VIPA-ID'.

Für die Diagnose ist der Betriebszustand der CPU irrelevant. Es können maximal 100 Diagnoseeinträge in der CPU gespeichert werden.

## Übersicht der Ereignis-IDs

Ereignis-ID	Bedeutung
0x115C	Herstellerspezifischer Alarm (OB 57) bei EtherCAT OB: OB-Nummer (57) ZInfo1: Logische Adresse des Slaves, der den Alarm ausgelöst hat ZInfo2: Alarmtyp ZInfo3: Reserviert
0xE003	Fehler beim Zugriff auf Peripherie Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz
0xE004	Mehrfach-Parametrierung einer Peripherieadresse Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz
0xE005	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE006	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE007	Konfigurierte Ein-/Ausgangsbytes passen nicht in Peripheriebereich
0xE008	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE009	Fehler beim Zugriff auf Standard-Rückwandbus
0xE010	Nicht definierte Baugruppe am Rückwandbus erkannt Zinfo2: Steckplatz Zinfo3: Typkennung
0xE011	Masterprojektierung auf Slave-CPU nicht möglich oder fehlerhafte Slave-Konfiguration
0xE012	Fehler bei Parametrierung
0xE013	Fehler bei Schieberegisterzugriff auf Standardbus-Digitalmodule
0xE014	Fehler bei Check_Sys
0xE015	Fehler beim Zugriff auf Master Zinfo2: Steckplatz des Masters (32=Kachelmaster)
0xE016	Maximale Blockgröße bei Mastertransfer überschritten Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz
0xE017	Fehler beim Zugriff auf integrierten Slave
0xE018	Fehler beim Mappen der Master-Peripherie
0xE019	Fehler bei Erkennung des Standard Rückwandbus Systems
0xE01A	Fehler bei Erkennung der Betriebsart (8 / 9 Bit)

Ereignis-ID	Bedeutung
0xE01B	Fehler - Maximale Anzahl steckbarer Baugruppen überschritten
0xE020	Fehler - Alarminformationen undefiniert
0xE030	Fehler vom Standard-Bus
0xE033	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE0B0	SPEED7 kann nicht mehr gestoppt werden (Evtl. undefinierter BCD-Wert bei Timer)
0xE0C0	Nicht genug Speicherplatz im Arbeitsspeicher für Codebaustein (Baustein zu groß)
0xE0CB	Fehler bei SZL-Zugriff Zinfo1: 4=SZL falsch, 5=SubSZL falsch, 6=Index falsch Zinfo2: SZL-ID Zinfo3: Index
0xE0CC	Kommunikationsfehler MPI / Seriell Zinfo1: Code 1: Falsche Priorität 2: Pufferüberlauf 3: Telegrammformatfehler 4: Falsche SZL-Anforderung (SZL-ID ungültig) 5: Falsche SZL-Anforderung (SZL-SubID ungültig) 6: Falsche SZL-Anforderung (SZL-Index ungültig) 7: Falsche Wert 8: Falscher RetVal 9: Falscher SAP 10: Falscher Verbindungstyp 11: Falsche Sequenznummer 12: Fehlerhafte Bausteinnummer im Telegramm 13: Fehlerhafter Bausteintyp im Telegramm 14: Inaktive Funktion 15: Fehlerhafte Größe im Telegramm 20: Fehler beim Schreiben auf die Speicherkarte 90: Fehlerhafte Puffergröße 98: Unbekannter Fehler 99: Interner Fehler
0xE0CD	Fehler bei DP-V1 Auftragsverwaltung
0xE0CE	Fehler: Timeout beim Senden der i-Slave Diagnose
0xE0CF	Timeout beim Laden einer neuen HW-Konfiguration (Timeout-Zeit: 39 Sekunden)

## VIPA-spezifische Diagnose-Einträge

Ereignis-ID	Bedeutung
0xE100	Speicherkarten-Zugriffsfehler
0xE101	Speicherkarten-Fehler Filesystem
0xE102	Speicherkarten-Fehler FAT
0xE104	Speicherkarten-Fehler beim Speichern
0xE200	Speicherkarte schreiben beendet (Copy Ram2Rom)
0xE210	Speicherkarte Lesen beendet (Nachladen nach Urlöschen)
0xE21E	Speicherkarte Lesen: Fehler beim Nachladen (nach Urlöschen), Datei "Protect.wld" zu groß
0xE21F	Speicherkarte Lesen: Fehler beim Nachladen (nach Urlöschen), Lesefehler, Speicher voll
0xE300	Internes Flash Schreiben beendet (Copy Ram2Rom)
0xE310	Internes Flash Lesen beendet (Nachladen nach Batterieausfall)
0xE311	Internes Flash fx0000yy.wld Datei zu groß, Laden fehlerhaft
0xE400	Speicherkarte mit der Option Speichererweiterung wurde gesteckt.
0xE401	Speicherkarte mit der Option Speichererweiterung wurde gezogen.
0xE402	Die PROFIBUS-DP-Master-Funktionalität ist nicht aktiviert. Die Schnittstelle ist weiter als MPI-Schnittstelle aktiv.
0xE403	Die PROFIBUS-DP-Slave-Funktionalität ist nicht aktiviert. Die Schnittstelle ist weiter als MPI-Schnittstelle aktiv.
0xE500	Speicherverwaltung: Baustein ohne zugehörigen Eintrag in der BstListe gelöscht Zinfo2: BlockTyp Zinfo3: BlockNr
0xE604	Mehrfach-Parametrierung einer Peripherieadresse für Ethernet-PG/OP-Kanal Zinfo1: Peripherie-Adresse Zinfo3: 0: Peripherie-Adresse ist Eingang, 1: Peripherie-Adresse ist Ausgang
0xE701	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE703	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE720	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE721	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xE801	CMD - Autobefehl: CMD_START erkannt und erfolgreich ausgeführt
0xE802	CMD - Autobefehl: CMD_END erkannt und erfolgreich ausgeführt

Ereignis-ID	Bedeutung
0xE803	CMD - Autobefehl: WAIT1SECOND erkannt und erfolgreich ausgeführt
0xE804	CMD - Autobefehl: WEBPAGE erkannt und erfolgreich ausgeführt
0xE805	CMD - Autobefehl: LOAD_PROJECT erkannt und erfolgreich ausgeführt
0xE806	CMD - Autobefehl: SAVE_PROJECT Zinfo3: 0x0000: SAVE_PROJECT erkannt und erfolgreich ausgeführt Zinfo3: 0x8000: Fehler beim Ausführen von SAVE_PROJECT z.B. falsches Passwort
0xE807	CMD - Autobefehl: FACTORY_RESET erkannt und erfolgreich ausgeführt
0xE80B	CMD - Autobefehl: DIAGBUFF erkannt und erfolgreich ausgeführt
0xE80E	CMD - Autobefehl: SET_NETWORK erkannt und erfolgreich ausgeführt
0xE816	CMD - Autobefehl: SAVE_PROJECT: Fehler - CPU urgelöscht - es wurde keine wld-Datei erzeugt.
0xE8FB	CMD - Autobefehl: Fehler: Initialisierung des Ethernet-PG/OP-Kanals mittels SET_NETWORK fehlerhaft.
0xE8FC	CMD - Autobefehl: Fehler: In SET_NETWORK wurden nicht alle IP-Parameter angegeben.
0xE8FE	CMD - Autobefehl: Fehler: CMD_START nicht gefunden
0xE8FF	CMD - Autobefehl: Fehler: Fehler beim Lesen des CMD-Files (Speicherkarten-Fehler)
0xE901	Checksummen-Fehler
0xEA00	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA01	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA02	SBUS: Interner Fehler (intern gestecktes Submodul nicht erkannt) Zinfo1: Interner Steckplatz
0xEA03	SBUS: Kommunikationsfehler CPU - PROFINET-IO-Controller Zinfo1: Steckplatz Zinfo2: Status (0: OK, 1: ERROR, 2: BUSY, 3: TIMEOUT, 4: LOCKED, 5: UNKNOWN)
0xEA04	SBUS: Mehrfach-Parametrierung einer Peripherieadresse Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz Zinfo3: Datenbreite
0xEA05	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA07	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!

Ereignis-ID	Bedeutung
0xEA08	SBUS: Parametrierte Eingangsdatenbreite ungleich der gesteckten Eingangsdatenbreite Zinfo1: Parametrierte Eingangsdatenbreite Zinfo2: Steckplatz Zinfo3: Eingangsdatenbreite der gesteckten Baugruppe
0xEA09	SBUS: Parametrierte Ausgangsdatenbreite ungleich der gesteckten Ausgangsdatenbreite Zinfo1: Parametrierte Ausgangsdatenbreite Zinfo2: Steckplatz Zinfo3: Ausgangsdatenbreite der gesteckten Baugruppe
0xEA10	SBUS: Eingangs-Peripherieadresse außerhalb des Peripheriebereiches Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz Zinfo3: Datenbreite
0xEA11	SBUS: Ausgangs-Peripherieadresse außerhalb des Peripheriebereiches Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz Zinfo3: Datenbreite
0xEA12	SBUS: Fehler beim Datensatz schreiben Zinfo1: Steckplatz Zinfo2: Datensatznummer Zinfo3: Datensatzlänge
0xEA14	SBUS: Mehrfach-Parametrierung einer Peripherieadresse (Diagnoseadresse) Zinfo1: Peripherie-Adresse Zinfo2: Steckplatz Zinfo3: Datenbreite
0xEA15	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA18	SBUS: Fehler beim Mappen der Masterperipherie Zinfo2: Steckplatz des Masters
0xEA19	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA20	Fehler - RS485-Schnittstelle ist nicht auf PROFIBUS-DP-Master eingestellt aber es ist ein PROFIBUS-DP-Master projektiert.
0xEA21	Fehler - Projektierung RS485-Schnittstelle X2/X3: PROFIBUS-DP-Master ist projektiert aber nicht vorhanden Zinfo2: Schnittstelle x
0xEA22	Fehler - RS485-Schnittstelle X2 - Wert ist außerhalb der Grenzen Zinfo: Projektierter Wert von X2
0xEA23	Fehler - RS485-Schnittstelle X3 - Wert ist außerhalb der Grenzen Zinfo: Projektierter Wert von X3

Ereignis-ID	Bedeutung
0xEA24	Fehler - Projektierung RS485-Schnittstelle X2/X3: Schnittstelle/Protokoll ist nicht vorhanden, die Defaulteinstellungen werden verwendet. Zinfo2: Projektierter Wert für X2 Zinfo3: Projektierter Wert für X3
0xEA30	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA40	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA41	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA50	Fehler - PROFINET-Konfiguration Zinfo1: User-Slot des PROFINET-IO-Controllers Zinfo2: IO-Device-Nr. Zinfo3: IO-Device Slot
0xEA51	Fehler - Kein PROFINET-IO-Controller auf dem projektierten Slot erkannt Zinfo1: User-Slot des PROFINET-IO-Controllers Zinfo2: Erkannte Typkennung auf dem projektierten Slot
0xEA53	Fehler - PROFINET-Konfiguration - Es sind zu viele PROFINET-IO-Devices projektiert Zinfo1 : Anzahl der projektierten Devices Zinfo2 : Steckplatz Zinfo3 : Maximal mögliche Anzahl Devices
0xEA54	Fehler - PROFINET-IO-Controller meldet Mehrfachparametrierung einer Peripherieadresse Zinfo1: Peripherieadresse Zinfo2: User-Slot des PROFINET-IO-Controllers Zinfo3: Datenbreite
0xEA61 ... 0xEA63	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!

Ereignis-ID	Bedeutung
0xEA64	PROFINET/EtherCAT-CP Konfigurationsfehler Zinfo1: Bit 0: Zu viele Devices Bit 1: Zu viele Devices pro Millisekunde Bit 2: Zu viele Eingangsbytes pro Millisekunde Bit 3: Zu viele Ausgangsbytes pro Millisekunde Bit 4: Zu viele Eingangsbytes pro Device Bit 5: Zu viele Ausgangsbytes pro Device Bit 6: Zu viele Produktiv-Verbindungen Bit 7: Zu viele Eingangsbytes im Prozessabbild Bit 8: Zu viele Ausgangsbytes im Prozessabbild Bit 9: Konfiguration nicht verfügbar Bit 10: Konfiguration ungültig Bit 11: Zykluszeit zu klein Bit 12: Aktualisierungszeit zu groß Bit 13: Ungültige Devicenummer Bit 14: CPU ist als I-Device konfiguriert Bit 15: IP Adresse auf anderem Weg beziehen, wird für die IP-Adresse des Controllers nicht unterstützt
0xEA65	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA66	PROFINET-IO-Controller Fehler im Kommunikationsstack PK: Rackslot OBNr: StackError.Service DatId: StackError.DeviceRef ZInfo1: StackError.Error.Code ZInfo2: StackError.Error.Detail ZInfo3: StackError.Error.AdditionalDetail << 8 + StackError.Error.AreaCode
0xEA67	Fehler - PROFINET-IO-Controller - Datensatz lesen PK: Fehlertyp 0: DATA_RECORD_ERROR_LOCAL 1: DATA_RECORD_ERROR_STACK 2: DATA_RECORD_ERROR_REMOTE OBNr: PROFINET-IO-Controller slot DatId: Device-Nr ZInfo1: Datensatznummer ZInfo2: Datensatzhandle ZInfo3: Interner Fehlercode für Service-Zwecke

Ereignis-ID	Bedeutung
0xEA68	Fehler - PROFINET-IO-Controller - Datensatz schreiben PK: Fehlertyp 0: DATA_RECORD_ERROR_LOCAL 1: DATA_RECORD_ERROR_STACK 2: DATA_RECORD_ERROR_REMOTE OBNr: PROFINET-IO-Controller slot DatId: Device-Nr ZInfo1: Datensatznummer ZInfo2: Datensatzhandle ZInfo3: Interner Fehlercode für Service-Zwecke
0xEA69	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEA6A	PROFINET-IO-Controller Service-Fehler im Kommunikationsstack PK: Rackslot OBNr: ServiceIdentifier DatId: 0 ZInfo1: ServiceError.Code ZInfo2: ServiceError.Detail ZInfo3: ServiceError.AdditionalDetail
0xEA6B	PROFINET-IO-Controller Vendor ID mismatch PK: Rackslot OBNr: PLC-Mode DatId: 0 ZInfo1: Device ID ZInfo2: - ZInfo3: -
0xEA6C	PROFINET-IO-Controller Device ID mismatch PK: Rackslot OBNr: PLC-Mode DatId: 0 ZInfo1: Device ID ZInfo2: - ZInfo3: -

## VIPA-spezifische Diagnose-Einträge

Ereignis-ID	Bedeutung
0xEA6D	PROFINET-IO-Controller No empty name PK: Rackslot OBNr: PLC-Mode DatId: 0 ZInfo1: Device ID ZInfo2: - ZInfo3: -
0xEA6E	PROFINET-IO-Controller RPC response missing PK: Rackslot OBNr: PLC-Mode DatId: 0 ZInfo1: Device ID ZInfo2: - ZInfo3: -
0xEA6F	PROFINET-IO-Controller PN module mismatch PK: Rackslot OBNr: PLC-Mode DatId: 0 ZInfo1: Device ID ZInfo2: - ZInfo3: -
0xEA97	Speicherfehler SBUS-Service Kanal ZInfo3 = Slot
0xEA98	Timeout beim Warten, dass ein SBUS-Modul (Server) rebootet hat
0xEA99	Fehler beim File-Lesen über SBUS

Ereignis-ID	Bedeutung
0xEAA0	Emac Error ist aufgetreten OBNr: Aktueller PLC-Mode ZInfo1: Diagnoseadresse des Masters / Controllers ZInfo2: 0: Kein Rx Queue ist voll 1: Kein Sendepuffer verfügbar 2: Sendestrom ist abgerissen; senden fehlgeschlagen 3: Wiederholungsversuche ausgeschöpft 4: Kein Empfangspuffer in Emac DMA verfügbar 5: Emac DMA Transfer abgebrochen 6: Queue Overflow 7: Nicht erwartetes Packet empfangen ZInfo3: Anzahl der aufgetretenen Fehler
0xEAB0	Ungültiger Link-Mode OBNr: Aktueller PLC-Mode ZInfo1: Diagnoseadresse des Masters/Controllers ZInfo2: Aktueller LinkMode 0x01: 10MBit Full-Duplex 0x02: 100MBit Half-Duplex 0x03: 100Mbit Full-Duplex 0x05: 10Mbit Half-Duplex 0xFF: Link Mode nicht definiert
0xEB03	SLIO Fehler beim IO-Mapping
0xEB10	SLIO Fehler: Busfehler ZInfo1: Fehlerart 0x82: ErrorAlarm
0xEB20	SLIO Fehler: Alarminformationen undefiniert
0xEB21	SLIO Fehler bei Zugriff auf Konfigurationsdaten

Ereignis-ID	Bedeutung
0xEC03	<p>EtherCAT: Konfigurationsfehler</p> <p>ZInfo1: Errorcode</p> <p>1: NUMBER_OF_SLAVES_NOT_SUPPORTED</p> <p>2: SYSTEM_IO_NR_INVALID</p> <p>3: INDEX_FROM_SLOT_ERROR</p> <p>4: MASTER_CONFIG_INVALID</p> <p>5: MASTER_TYPE_ERROR</p> <p>6: SLAVE_DIAG_ADDR_INVALID</p> <p>7: SLAVE_ADDR_INVALID</p> <p>8: SLAVE_MODULE_IO_CONFIG_INVALID</p> <p>9: LOG_ADDR_ALREADY_IN_USE</p> <p>10: NULL_PTR_CHECK_ERROR</p> <p>11: IO_MAPPING_ERROR</p> <p>12: ERROR</p>
0xEC04	<p>EtherCAT: Mehrfach-Parametrierung einer Peripherieadresse</p> <p>Zinfo1 : Peripherie-Adresse</p> <p>Zinfo2 : Steckplatz</p>
0xEC10	<p>EtherCAT: Wiederkehr Bus mit allen Slaves</p> <p>OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xEC10</p> <p>DatID:</p> <p>0xXXYY:</p> <p>XX=0x54 bei Eingangsadresse in ZInfo1,</p> <p>XX=0x55 bei Ausgangsadresse.</p> <p>YY=0x00 Station nicht verfügbar,</p> <p>YY=0x01 Station verfügbar (Prozessdaten)</p> <p>ZInfo1: 0xXXYY (XX=OldState, YY=NewState)</p> <p>ZInfo2: Diagnoseadresse des Masters</p> <p>ZInfo3: Anzahl der Station, die nicht im selben State sind, wie der Master (&gt; 0)</p>

Ereignis-ID	Bedeutung
0xEC11	<p>EtherCAT: Wiederkehr Bus mit fehlenden Slaves                      OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xEC11                      DatID:                      0xXXYY:                      XX=0x54 bei Eingangsadresse in ZInfo1,                      XX=0x55 bei Ausgangsadresse.                      YY=0x00 Station nicht verfügbar,                      YY=0x01 Station verfügbar (Prozessdaten)                      ZInfo1: 0xXXYY (XX=OldState, YY=NewState)                      ZInfo2: Diagnoseadresse des Masters                      ZInfo3: Anzahl der Station, die nicht im selben State sind, wie der Master (&gt; 0)</p>
0xEC12	<p>EtherCAT: Wiederkehr Slave                      OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xEC12                      DatID:                      0xXXYY:                      XX=0x54 bei Eingangsadresse in ZInfo1,                      XX=0x55 bei Ausgangsadresse.                      YY=0x00 Station nicht verfügbar,                      YY=0x01 Station verfügbar (Prozessdaten)                      ZInfo1: 0xXXYY (XX=OldState, YY=NewState)                      ZInfo2: Diagnoseadresse der Station                      ZInfo3: AIStatusCode</p>
0xEC30	<p>EtherCAT: Topologie OK                      OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xEC30                      ZInfo2: Diagnoseadresse des Masters</p>
0xEC50	<p>EtherCAT: DC nicht in Sync                      ZInfo1: Diagnoseadresse des Masters</p>
0xED10	<p>EtherCAT: Ausfall Bus                      OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED10                      DatID:                      0xXXYY:                      XX=0x54 bei Eingangsadresse in ZInfo1,                      XX=0x55 bei Ausgangsadresse.                      YY=0x00 Station nicht verfügbar,                      YY=0x01 Station verfügbar (Prozessdaten)                      ZInfo1: 0xXXYY (XX=OldState, YY=NewState)                      ZInfo2: Diagnoseadresse des Masters                      ZInfo3: Anzahl der Station, die nicht im selben State sind, wie der Master</p>

Ereignis-ID	Bedeutung
0xED12	<p>EtherCAT: Ausfall Slave</p> <p>OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED12</p> <p>DatID:</p> <p>0xXXYY:</p> <p>XX=0x54 bei Eingangsadresse in ZInfo1, XX=0x55 bei Ausgangsadresse.</p> <p>YY=0x00 Station nicht verfügbar, YY=0x01 Station verfügbar (Prozessdaten)</p> <p>ZInfo1: 0xXXYY (XX=OldState, YY=NewState)</p> <p>ZInfo2: Diagnoseadresse der Station</p> <p>ZInfo3: AIStatusCode</p>
0xED20	<p>EtherCAT: Bus-Statuswechsel, der keinen OB86 hervorruft</p> <p>OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED20</p> <p>DatID:</p> <p>0xXXYY:</p> <p>XX=0x54 bei Eingangsadresse in ZInfo1, XX=0x55 bei Ausgangsadresse.</p> <p>YY=0x00 Station nicht verfügbar, YY=0x01 Station verfügbar (Prozessdaten)</p> <p>ZInfo1: 0xXXYY (XX=OldState, YY=NewState)</p> <p>ZInfo2: Diagnoseadresse des Masters</p> <p>ZInfo3: Anzahl der Station, die nicht im selben State sind, wie der Master</p>
0xED21	<p>EtherCAT: fehlerhafter Bus-Statuswechsel</p> <p>OB: 0x00</p> <p>PK: 0x00</p> <p>DatID:</p> <p>0xXXYY:</p> <p>XX=0x54 bei Eingangsadresse in ZInfo1, XX=0x55 bei Ausgangsadresse.</p> <p>YY=0x00 Station nicht verfügbar, YY=0x01 Station verfügbar (Prozessdaten)</p> <p>ZInfo1: 0xXXYY (XX = current state, YY = expected state)</p> <p>ZInfo2: Diagnoseadresse des Masters</p> <p>ZInfo3: ErrorCode:</p> <p>0x0008: Busy</p> <p>0x000B: Unzulässige Parameter</p> <p>0x000E: Unzulässiger Status</p> <p>0x0010: Zeitüberschreitung</p>

Ereignis-ID	Bedeutung
0xED22	<p>EtherCAT: Slave-Statuswechsel, der keinen OB86 hervorruft            OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED22            DatID:            0xXXYY:            XX=0x54 bei Eingangsadresse in ZInfo1,            XX=0x55 bei Ausgangsadresse.            YY=0x00 Station nicht verfügbar,            YY=0x01 Station verfügbar (Prozessdaten)            ZInfo1: 0xXXYY (XX=OldState, YY=NewState)            ZInfo2: Diagnoseadresse der Station            ZInfo3: AIStatusCode</p>
0xED30	<p>EtherCAT: Topolgy Mismatch            OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED30            ZInfo2: Diagnoseadresse des Masters</p>
0xED31	<p>EtherCAT: Alarm Queue Overflow            OB-StartInfo (Lokaldaten) StartEvent und Eventclass: 0xED31            ZInfo2: Diagnoseadresse des Masters</p>
0xED40 ... 0xED4F	<p>Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!</p>
0xED50	<p>EtherCAT: DC in Sync            ZInfo1: Diagnoseadresse des Masters</p>
0xED60	<p>EtherCAT: Diagnosepuffer CP:            Slave-Statuswechsel            PK: 0            OB: PLC-Mode            DatID 1/2: 0            ZInfo1: 0x00YY:            YY: Neuer EtherCAT-Status des Slaves            ZInfo2: EtherCAT-Stationsadresse            ZInfo3: AIStatusCode (EtherCAT-spezifischer Fehlercode)</p>

## VIPA-spezifische Diagnose-Einträge

Ereignis-ID	Bedeutung
0xED61	EtherCAT: Diagnosepuffer CP: CoE-Emergency PK: EtherCAT-Stationsadresse (Low-Byte) OB: EtherCAT-Stationsadresse (High-Byte) DatID 1/2: Error-Code ZInfo1: 0xYYZZ: YY: Error-Register ZZ: MEF Byte 1 ZInfo 2: 0xYYZZ: YY: MEF Byte 2 ZZ: MEF Byte 3 Zinfo3: 0xYYZZ: YY: MEF Byte 4 ZZ: MEF Byte 5
0xED62	EtherCAT: Diagnosepuffer CP: Fehler bei SDO-Zugriff während State-Wechsel PK: EtherCAT-Stationsadresse (Low-Byte) OB: EtherCAT-Stationsadresse (High-Byte) DatID 1/2: Subindex ZInfo1: Index ZInfo2: SDO-Errorcode (High-Word) Zinfo3: SDO-Errorcode (Low-Word)
0xED70	EtherCAT: Diagnosepuffer CP: Doppelte Hot Connect Gruppe erkannt PK: 0 OB: PLC-Mode DatID 1/2: 0 ZInfo1: Diagnoseadresse des Masters ZInfo2: EtherCAT-Stationsadresse Zinfo3: 0
0xEE00	Zusatzinformation bei UNDEF_OPCODE
0xEE01	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
0xEEEE	CPU wurde komplett urgelöscht, da der Hochlauf nach NetzEIN nicht beendet werden konnte.
0xEF11 ... 0xEF13	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!

Ereignis-ID	Bedeutung
0xEFFF	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!
PK: C-Sourcemodulnummer   DatID: Zeilennummer	

## 5.20 Mit Testfunktionen Variablen steuern und beobachten

### Übersicht

Zur Fehlersuche und zur Ausgabe von Variablenzuständen können Sie in Ihrem Siemens SIMATIC Manager unter dem Menüpunkt **Test** verschiedene Testfunktionen aufrufen.

- Mit der Testfunktion **"Test → Beobachten"** können die Signalzustände von Operanden und das VKE angezeigt werden.
- Mit der Testfunktion **"Zielsystem → Variablen beobachten/steuern"** können die Signalzustände von Variablen geändert und angezeigt werden.

### **"Test → Beobachten"**

Diese Testfunktion zeigt die aktuellen Signalzustände und das VKE der einzelnen Operanden während der Programmbearbeitung an. Es können außerdem Korrekturen am Programm durchgeführt werden.



*Die CPU muss bei der Testfunktion "Beobachten" in der Betriebsart RUN sein!*

Die Statusbearbeitung kann durch Sprungbefehle oder Zeit- und Prozessalarmlen unterbrochen werden. Die Unterbrechung der Statusbearbeitung hat keinen Einfluss auf die Programmbearbeitung, sondern macht nur deutlich, dass die angezeigten Daten ab der Unterbrechungsstelle nicht mehr gültig sind. Die CPU hört an der Unterbrechungsstelle auf, Daten für die Statusanzeige zu sammeln und übergibt dem PG anstelle der noch benötigten Daten nur Daten mit dem Wert 0. Deshalb kann es bei Verwendung von Sprungbefehlen oder von Zeit- und Prozessalarmlen vorkommen, dass in der Statusanzeige eines Bausteins während dieser Programmbearbeitung nur der Wert 0 angezeigt wird für:

- das Verknüpfungsergebnis VKE
- Status / AKKU 1
- AKKU 2
- Zustandsbyte
- absolute Speicheradresse SAZ. Hinter SAZ erscheint dann ein "?".

### **"Zielsystem → Variablen beobachten/steuern"**

Diese Testfunktion gibt den Zustand eines beliebigen Operanden (Eingänge, Ausgänge, Merker, Datenwort, Zähler oder Zeiten) am Ende einer Programmbearbeitung an. Diese Informationen werden aus dem Prozessabbild der ausgesuchten Operanden entnommen. Während der "Bearbeitungskontrolle" oder in der Betriebsart STOP wird bei den Eingängen direkt die Peripherie eingelesen. Andernfalls wird nur das Prozessabbild der aufgerufenen Operanden angezeigt.

---

Mit Testfunktionen Variablen steuern und beobachten

- Steuern von Ausgängen
  - Dadurch kann die Verdrahtung und die Funktionstüchtigkeit von Ausgabebaugruppen kontrolliert werden.
  - Auch ohne Steuerungsprogramm können Ausgänge auf den gewünschten Signalzustand eingestellt werden. Das Prozessabbild wird dabei nicht verändert, die Sperre der Ausgänge jedoch aufgehoben.
- Steuern von Variablen
  - Folgende Variablen können geändert werden: E, A, M, T, Z und D.
  - Unabhängig von der Betriebsart der CPU wird das Prozessabbild binärer und digitaler Operanden verändert.
  - In der Betriebsart RUN wird die Programmbearbeitung mit den geänderten Prozessvariablen ausgeführt. Im weiteren Programmablauf können sie jedoch ohne Rückmeldung wieder verändert werden.
  - Die Prozessvariablen werden asynchron zum Programmablauf gesteuert.

## 6 Einsatz E/A-Peripherie

### 6.1 Übersicht

#### Hardware

Bei der 312-5BE13 sind die Anschlüsse für digitale Ein-/Ausgabe und Technologische Funktionen in einem 2-fach breiten Gehäuse untergebracht.

#### Projektierung Parametrierung

Die Projektierung erfolgt im Siemens SIMATIC Manager als CPU 312C von Siemens (6ES7 312-5BE03-0AB0 V2.6). Hierbei parametrieren Sie Ihre CPU 312-5BE13 über den "Eigenschaften"-Dialog der Siemens CPU 312C. Für die Parametrierung der digitalen Ein-/Ausgabeperipherie und der Technologischen Funktionen sind die entsprechenden Submodule der CPU 312C zu verwenden.

#### E/A- Peripherie

Die integrierten Ein-/Ausgänge der 312-5BE13 können für Technologische Funktionen bzw. als Standardperipherie genutzt werden. Soweit dies hardwareseitig möglich ist, dürfen Technologische Funktionen und Standardperipherie gleichzeitig genutzt werden. Belegte Eingänge von Technologischen Funktionen können gelesen werden. Belegte Ausgänge lassen sich nicht beschreiben.

#### Technologische Funktionen

Für 2 Kanäle sind Technologische Funktionen parametrierbar. Die Parametrierung des entsprechenden Kanals erfolgt im Hardware-Konfigurator über das *Zähler*-Submodul der CPU 312C. Folgende Technologische Funktionen stehen Ihnen zur Verfügung:

- Endlos Zählen
- Einmalig Zählen
- Periodisch Zählen
- Frequenzmessung
- Pulsweitenmodulation (PWM)

Die Steuerung der entsprechenden Zähler-Betriebsart erfolgt aus dem Anwenderprogramm über den SFB COUNT (SFB 47).

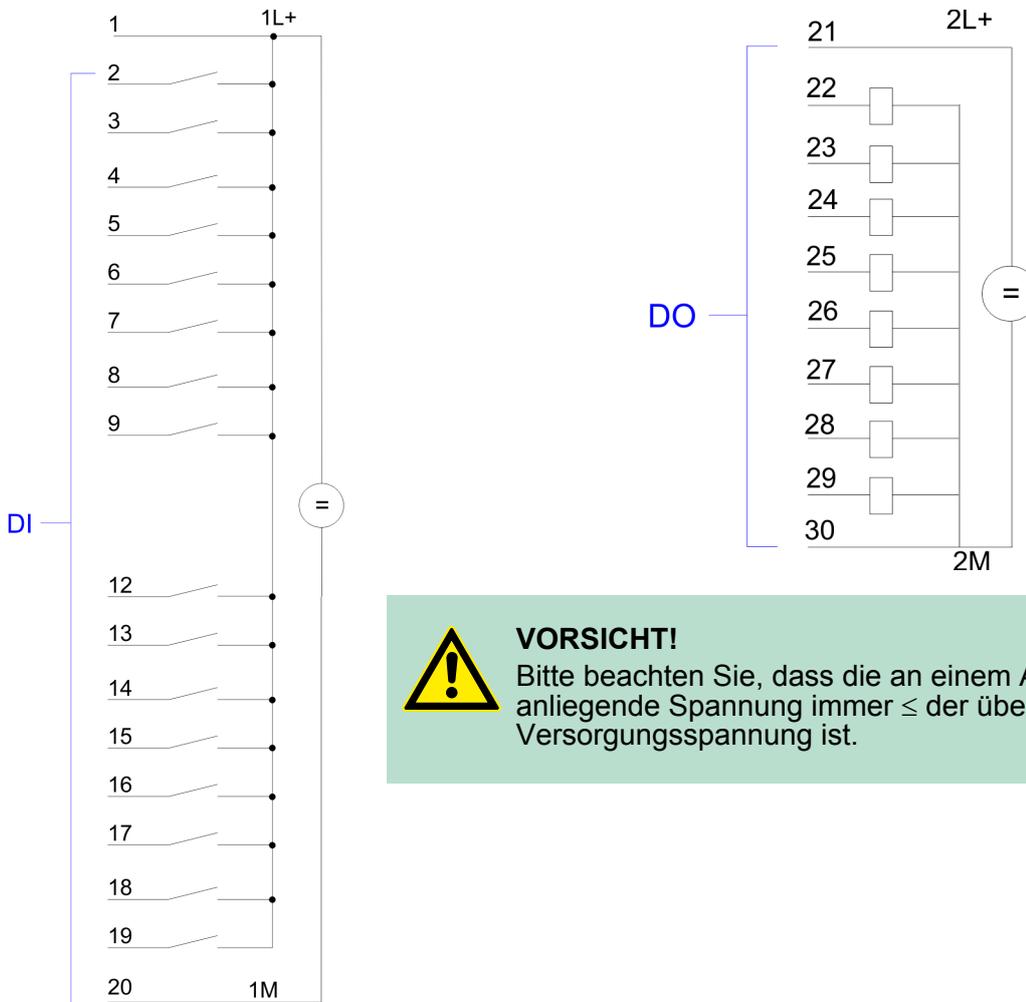
## 6.2 Ein-/Ausgabe-Bereich CPU 312-5BE13

### Übersicht

Bei der CPU 312-5BE13 sind folgende digitale Ein-/Ausgabe-Kanäle in einem Gehäuse untergebracht:

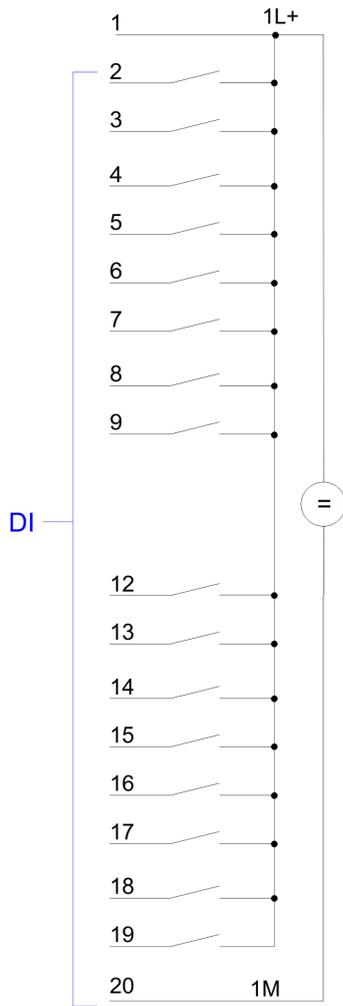
- Digitale Eingabe: 16xDC 24V, alarmfähig
- Digitale Ausgabe: 8xDC 24V, 0,5A
- Technologische Funktionen: 2 Kanäle

X11:



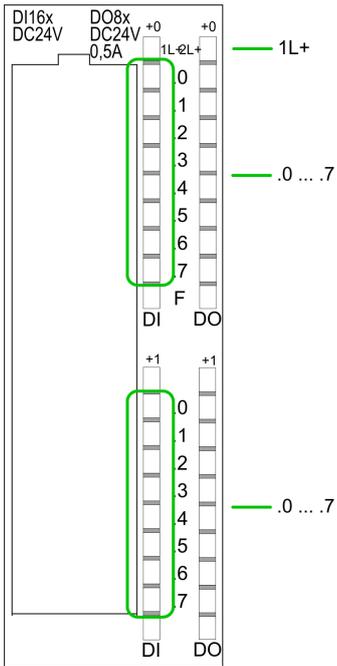
#### VORSICHT!

Bitte beachten Sie, dass die an einem Ausgabe-Kanal anliegende Spannung immer  $\leq$  der über L+ anliegenden Versorgungsspannung ist.



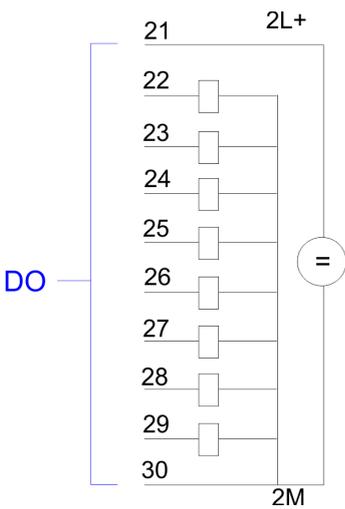
**Steckerbelegung X11: DI**

Pin	Belegung
1	1L+ Spannungsversorgung +DC 24V
2	E+0.0 / Kanal 0 (A) / Impuls
3	E+0.1 / Kanal 0 (B) / Richtung
4	E+0.2 / Kanal 0 Hardwaretor
5	E+0.3 / Kanal 1 (A) / Impuls
6	E+0.4 / Kanal 1 (B) / Richtung
7	E+0.5 / Kanal 1 Hardwaretor
8	E+0.6
9	E+0.7
10	nicht belegt
11	nicht belegt
12	E+1.0
13	E+1.1
14	E+1.2
15	E+1.3
16	E+1.4 / Kanal 0 Latch
17	E+1.5 / Kanal 1 Latch
18	E+1.6
19	E+1.7
20	Masse 1M DI



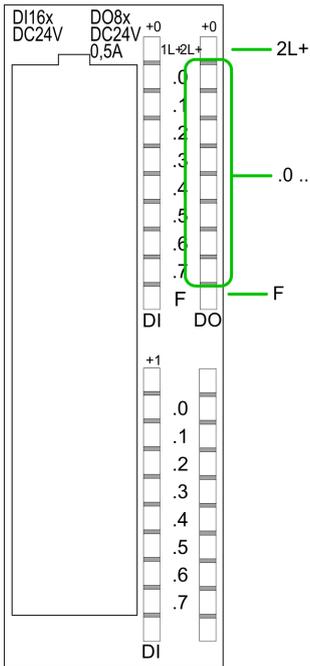
**Statusanzeige X11: DI**

- 1L+
  - LED (grün)
  - Versorgungsspannung für DI liegt an
- .0... .7
  - LEDs (grün)
  - E+0.0 ... E+0.7
  - E+1.0 ... E+1.7
  - Ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert



**Steckerbelegung X11: DO**

Pin	Belegung
21	2L+ Spannungsversorgung +DC 24V
22	A+0.0 / Kanal 0 Ausgang
23	A+0.1 / Kanal 1 Ausgang
24	A+0.2
25	A+0.3
26	A+0.4
27	A+0.5
28	A+0.6
29	A+0.7
30	Masse 2M DO
31 ... 40	nicht belegt



**Statusanzeige X11: DO**

- 2L+
  - LED (grün)
  - Versorgungsspannung für DO liegt an
- .0... .7
  - LEDs (grün)
  - A+0.0 ... A+0.7
  - Die entsprechende LED leuchtet bei aktiviertem Ausgang
- F
  - LED (rot)
  - Fehler bei Überlast oder Kurzschluss

**6.3 Adressbelegung**

**Eingabebereich**

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Eingabe E+0.0 ... E+0.7
	125	Byte	Digitale Eingabe E+1.0 ... E+1.7
Zähler	768	DInt	Kanal 0: Zählerwert / Frequenzwert
	772	DInt	Kanal 1: Zählerwert / Frequenzwert
	776	DInt	reserviert
	780	DInt	reserviert

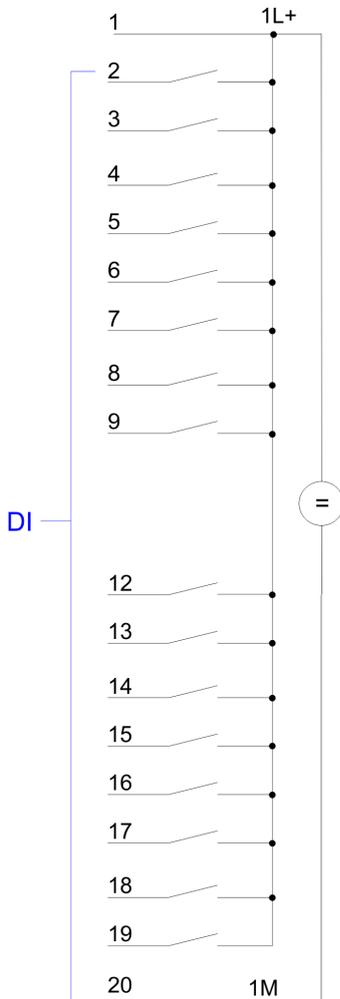
**Ausgabebereich**

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Ausgabe A+0.0 ... A+0.7
Zähler	768	DWort	reserviert
	772	DWort	reserviert
	776	DWort	reserviert
	780	DWort	reserviert

## 6.4 Digital-Teil

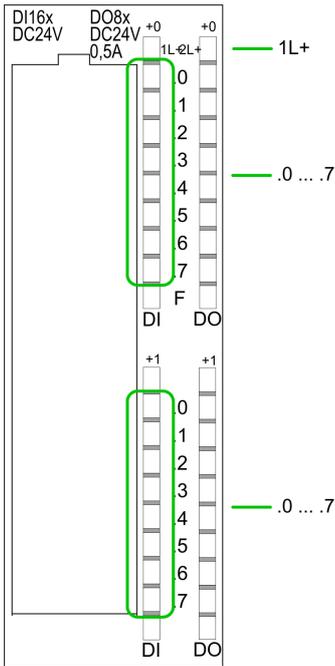
### 312-5BE13

Der digitale Bereich besteht aus 16 Eingabe-, 8 Ausgabe-Kanälen und 2 Kanälen für Technologische Funktionen. Jeder der digitalen Ein- bzw. Ausgänge zeigt seinen Zustand über eine LED an. Über die Parametrierung können Sie den Eingängen E+0.0 bis E+1.1 Alarm-Eigenschaften zuweisen.



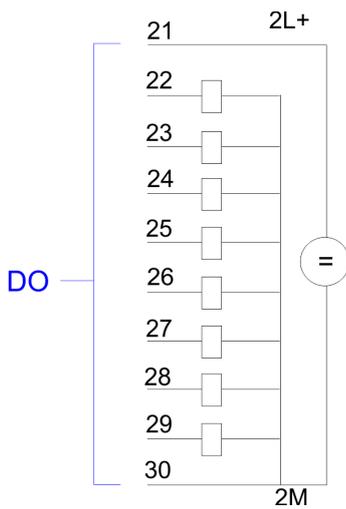
#### Steckerbelegung X11: DI

Pin	Belegung
1	1L+ Spannungsversorgung +DC 24V
2	E+0.0 / Kanal 0 (A) / Impuls
3	E+0.1 / Kanal 0 (B) / Richtung
4	E+0.2 / Kanal 0 Hardwaretor
5	E+0.3 / Kanal 1 (A) / Impuls
6	E+0.4 / Kanal 1 (B) / Richtung
7	E+0.5 / Kanal 1 Hardwaretor
8	E+0.6
9	E+0.7
10	nicht belegt
11	nicht belegt
12	E+1.0
13	E+1.1
14	E+1.2
15	E+1.3
16	E+1.4 / Kanal 0 Latch
17	E+1.5 / Kanal 1 Latch
18	E+1.6
19	E+1.7
20	Masse 1M DI



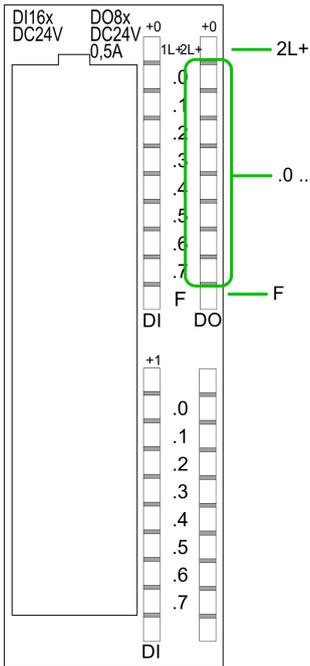
**Statusanzeige X11: DI**

- 1L+
    - LED (grün)
    - Versorgungsspannung für DI liegt an
  - .0... .7
    - LEDs (grün)
    - E+0.0 ... E+0.7
    - E+1.0 ... E+1.7
- Ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert



**Steckerbelegung X11: DO**

Pin	Belegung
21	2L+ Spannungsversorgung +DC 24V
22	A+0.0 / Kanal 0 Ausgang
23	A+0.1 / Kanal 1 Ausgang
24	A+0.2
25	A+0.3
26	A+0.4
27	A+0.5
28	A+0.6
29	A+0.7
30	Masse 2M DO
31 ... 40	nicht belegt



**Statusanzeige X11: DO**

- 2L+
  - LED (grün)
  - Versorgungsspannung für DO liegt an
- .0... .7
  - LEDs (grün)
  - A+0.0 ... A+0.7
  - Die entsprechende LED leuchtet bei aktiviertem Ausgang
- F
  - LED (rot)
  - Fehler bei Überlast oder Kurzschluss

**6.4.1 Zugriff auf den E/A Bereich**

Die 312-5BE13 legt in ihrem Peripheriebereich einen Bereich für Datenein- bzw. -ausgabe an. Ohne Hardware-Konfiguration werden die nachfolgend aufgeführten Default-Adressen verwendet.

**6.4.1.1 Adressbelegung**

**Eingabebereich**

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Eingabe E+0.0 ... E+0.7
	125	Byte	Digitale Eingabe E+1.0 ... E+1.7
Zähler	768	DInt	Kanal 0: Zählerwert / Frequenzwert
	772	DInt	Kanal 1: Zählerwert / Frequenzwert
	776	DInt	reserviert
	780	DInt	reserviert

**Ausgabebereich**

Submodul	Default-Adresse	Zugriff	Belegung
DI10/DO6	124	Byte	Digitale Ausgabe A+0.0 ... A+0.7
Zähler	768	DWort	reserviert
	772	DWort	reserviert
	776	DWort	reserviert
	780	DWort	reserviert

## 6.4.2 Parametrierung - Digital-Teil

<b>Parameterbereich</b>	Über das <i>DI10/DO6</i> -Submodul der CPU 312C von Siemens können Sie in der Hardware-Konfiguration Parameter für den Digital-Teil vorgeben. Nachfolgend sind alle Parameter aufgeführt, die Sie bei der Hardware-Konfiguration der digitalen Peripherie verwenden können.
<b>Allgemein</b>	Hier finden Sie die Kurzbezeichnung der digitalen Peripherie. Unter <i>Kommentar</i> können Sie den Einsatzzweck kommentieren.
<b>Adressen</b>	In diesem Register legen Sie die Anfangsadresse der Ein- und Ausgabe-Peripherie fest.
<b>Eingänge</b>	Hier haben Sie folgende Einstellmöglichkeiten: <ul style="list-style-type: none"> <li>■ Prozessalarm <ul style="list-style-type: none"> <li>– Ein Prozessalarm kann wahlweise bei steigender bzw. fallender Flanke an einem Eingang ausgelöst werden. Diagnosealarm wird nur in Verbindung mit Prozessalarm verloren unterstützt. Wählen Sie mit den Pfeiltasten den Eingang an und aktivieren Sie die gewünschten Prozessalarme.</li> </ul> </li> <li>■ Eingangsverzögerung <ul style="list-style-type: none"> <li>– Die Eingangsverzögerung können Sie kanalweise in Vierergruppen einstellen. Bitte beachten Sie, dass Sie im Parameterfenster ausschließlich den Wert 0,1ms einstellen können. Bei allen anderen Werten wird intern eine Eingangsverzögerung von 0,35ms verwendet.</li> </ul> </li> </ul>
<b>Ausgänge</b>	Für die digitalen Ausgabe-Kanäle gibt es keine Parametriermöglichkeiten.

## 6.5 Zähler

### 6.5.1 Zähler - Schnelleinstieg

#### Übersicht

Die CPU 312-5BE13 besitzt Ein-/Ausgänge, die Sie für Technologische Funktionen bzw. als Standardperipherie verwenden können. Soweit dies hardwareseitig möglich ist, können Sie Technologische Funktionen und Standardperipherie gleichzeitig nutzen. Belegte Eingänge von Technologischen Funktionen können gelesen werden. Belegte Ausgänge lassen sich nicht beschreiben. Die Parametrierung des entsprechenden Kanals erfolgt im Hardware-Konfigurator über das *Zähler*-Submodul der CPU 312C von Siemens. Zur Zeit stehen Ihnen folgende Technologische Funktionen auf 2 Kanälen zur Verfügung:

- Endlos Zählen, z.B. zur Wegerfassung mit Inkrementalgebern
- Einmalig Zählen, z.B. Stückguterfassung bis zu einer maximalen Grenze
- Periodisch Zählen, z.B. Anwendungen mit wiederholten Zählvorgängen

Für die CPU 312-5BE13 beträgt die maximale Frequenz unabhängig von der Anzahl der aktivierten Zähler 10kHz.

Die Steuerung der entsprechenden Betriebsarten erfolgt aus dem Anwenderprogramm über den SFB COUNT (SFB 47).

**Steckerbelegung** Kapitel 6.4 "Digital-Teil" auf Seite 98**Zähler vorbelegen bzw. parametrieren**

Während des Zählvorgangs wird das Zählersignal erfasst und ausgewertet. Jeder Zähler belegt im Eingabebereich ein Doppelwort für das *Zählerregister*. In den Betriebsarten "Einmalig Zählen" und "Periodisch Zählen" können Sie über die Parametrierung entsprechend der Zählrichtung vorwärts bzw. rückwärts einen Zählerbereich als End- bzw. Startwert definieren. Für jeden Zähler stehen Ihnen parametrierbare Zusatzfunktionen zur Verfügung wie Tor-Funktion, Latch-Funktion, Vergleicher, Hysterese und Prozessalarm. Über das *Zähler-Submodul* der CPU 312C von Siemens können Sie in der Hardware-Konfiguration alle Zählerparameter vorgeben. Hier definieren Sie unter anderem:

- Alarmverhalten
- max. Frequenz
- Zählerbetriebsart bzw. -verhalten
- Start-, End-, Vergleichswert und Hysterese

**Parametriervorgang**

1. ▶ Starten Sie den Siemens SIMATIC Manager mit Ihrem Projekt und öffnen Sie den Hardware-Konfigurator.
2. ▶ Platzieren Sie eine Profilschiene.
3. ▶ Projektieren Sie auf Steckplatz 2 die entsprechende Siemens CPU 312C.
4. ▶ Doppelklicken Sie auf das *Zähler-Submodul* Ihrer CPU. Sie gelangen in das Dialogfeld "Eigenschaften".
5. ▶ Sobald Sie für den entsprechenden Kanal die gewünschte Betriebsart einstellen, wird ein Dialogfenster für diese Zähler-Betriebsart mit Defaultwerten eingerichtet und angezeigt.
6. ▶ Führen Sie die gewünschten Parametrierungen durch.
7. ▶ Speichern Sie Ihr Projekt mit "*Station*"  
→ *Speichern und übersetzen*".
8. ▶ Transferieren Sie Ihr Projekt in Ihre CPU.

**Zählerfunktionen steuern**

Zur Ansteuerung der einzelnen Zählerfunktionen ist der SFB COUNT (SFB 47) zyklisch (z.B. OB 1) zu verwenden. Der SFB ist mit zugehörigem Instanz-DB aufzurufen. Hier liegen die Parameter für den SFB ab. Unter anderem bietet der SFB 47 eine Auftragsschnittstelle. Hiermit haben Sie lesenden und schreibenden Zugriff auf die Register des entsprechenden Zählers. Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit JOB\_DONE = TRUE abgeschlossen sein. Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.



*Wenn Sie in Ihrem Programm einen SFB programmiert haben, dürfen Sie in einem Programmteil mit einer anderen Prioritätsklasse nicht den selben SFB nochmals aufrufen, da der SFB sich nicht selbst unterbrechen darf. Beispiel: Es ist nicht zulässig einen SFB im OB 1 und den selben SFB im Alarm-OB aufzurufen.*

**Zähler steuern**

Gesteuert wird der Zähler über das interne Tor (I-Tor). Das I-Tor ist das Verknüpfungsergebnis von Hardware- (HW) und Software-Tor (SW), wobei die HW-Tor-Auswertung über die Parametrierung deaktiviert werden kann.

HW-Tor:	Öffnen (aktivieren):	Flankenwechsel 0-1 am Hardwaretor <sub>x</sub> -Eingang am Modul
	Schließen (deaktivieren):	Flankenwechsel 1-0 am Hardwaretor <sub>x</sub> -Eingang am Modul
SW-Tor:	Öffnen (aktivieren):	Im Anwenderprogramm durch Setzen von SW_GATE des SFB 47
	Schließen (deaktivieren):	Im Anwenderprogramm durch Rücksetzen von SW_GATE des SFB 47

**Zähler auslesen**

Mit dem SFB 47 können Sie über den Ausgabeparameter COUNTVAL den aktuellen Zählerwert auslesen. Sie haben aber auch die Möglichkeit über die Eingangsadresse des *Zähler*-Submoduls direkt auf die Zählerstände zuzugreifen.

**Zähler-Eingänge (Anschlüsse)**

Die Baugruppe bietet für die Technologischen Funktionen folgende Anschlussmöglichkeiten:

- 24V-Inkrementalgeber mit zwei um 90° phasenversetzten Spuren
- 24V-Impulsgeber mit Richtungspegel
- 24V-Initiator wie BERO oder Lichtschranke

Da nicht alle Eingänge gleichzeitig zur Verfügung stehen, können Sie über die Parametrierung die Eingangsbelegung für jeden Zähler bestimmen. Je Zähler stehen Ihnen folgende Eingänge zur Verfügung:

- *Kanal<sub>x</sub> (A)*  
Impulseingang für Zählsignal bzw. die Spur A eines Gebers. Hierbei können Sie Geber mit 1-, 2- oder 4-facher Auswertung anschließen.
- *Kanal<sub>x</sub> (B)*  
Richtungssignal bzw. die Spur B des Gebers. Über die Parametrierung können Sie das Richtungssignal invertieren.
- *Hardwaretor<sub>x</sub>*  
Über diesen Eingang können Sie mit einem High-Pegel das HW-Tor öffnen und somit einen Zählvorgang starten. Die Verwendung des HW-Tors ist parametrierbar.
- *Latch<sub>x</sub>*  
Mit einer positiven Flanke an Latch<sub>x</sub> wird der aktuelle Zählerstand in einem Speicher abgelegt, den Sie bei Bedarf auslesen können.

**Zähler-Ausgänge**

Jedem Zähler ist ein Ausgabe-Kanal zugeordnet. Folgendes Verhalten können Sie für den entsprechenden Ausgabe-Kanal über die Parametrierung einstellen:

- Kein Vergleich: Ausgang wird nicht angesteuert und wie ein Standard-Ausgang geschaltet.
- Zählwertstand  $\geq$  Vergleichswert: Ausgang wird gesetzt, solange Zählerstand  $\geq$  Vergleichswert gilt.
- Zählwertstand  $\leq$  Vergleichswert: Ausgang wird gesetzt, solange Zählerstand  $\leq$  Vergleichswert gilt.
- Impuls bei Vergleichswert: Zur Anpassung an die verwendete Aktoren können Sie eine Impulsdauer vorgeben. Der Ausgang wird für die vorgegebene Impulsdauer gesetzt, sobald der Zählerstand den Vergleichswert erreicht hat. Wenn Sie eine Hauptzählrichtung eingestellt haben, wird der Ausgang nur bei Erreichen des Vergleichswertes aus der Hauptzählrichtung gesetzt. Die maximale Impulsdauer darf 510ms betragen. Durch Vorgabe von 0 als *Impulsdauer* wird der Ausgang solange gesetzt wie die Vergleichsbedingung erfüllt ist.

**6.5.1.1 Parameterübersicht**

Nachfolgend sind alle Parameter aufgeführt, die Sie bei der Hardware-Konfiguration der Zählerfunktionen verwenden können.

**Allgemein**

Hier finden Sie die Kurzbezeichnung der Zählerfunktion. Unter *Kommentar* können Sie den Einsatzzweck kommentieren.

**Adressen**

In diesem Register legen Sie die Anfangsadresse der Ein- und Ausgabe-Peripherie fest.

**Grundparameter**

Hier können Sie einstellen, welche Alarmer die Zählerfunktion auslösen sollen. Sie haben folgende Auswahlmöglichkeiten:

- keine: Es wird kein Alarm ausgelöst.
- Prozess: Die Zählerfunktion löst einen Prozessalarm aus.
- Diagnose und Prozess: Bei der 312-5BE13 wird der Diagnosealarm der digitalen Ein-/Ausgabe-Peripherie nur in Verbindung mit "Prozessalarm verloren" unterstützt.

**Zählen**

Parameter	Beschreibung	Wertebereich	Vorbelegung
Hauptzählrichtung	<ul style="list-style-type: none"> <li>■ <i>Keine</i>: Keine Einschränkung des Zählbereiches</li> <li>■ <i>Vorwärts</i>: Einschränkung des Zählbereiches nach oben. Zähler zählt von 0 bzw. Ladewert in positiver Richtung bis zum parametrisierten Endwert-1 und springt dann mit dem darauf folgenden positiven Geberimpuls wieder auf den Ladewert.</li> <li>■ <i>Rückwärts</i>: Einschränkung des Zählbereiches nach unten. Zähler zählt vom parametrisierten Startwert bzw. Ladewert in negativer Richtung bis 1 und springt dann mit dem darauf folgenden negativen Geberimpuls wieder auf den Startwert.</li> </ul>	<ul style="list-style-type: none"> <li>■ Keine</li> <li>■ Vorwärts</li> <li>■ Rückwärts (nicht bei Endlos Zählen)</li> </ul>	<ul style="list-style-type: none"> <li>■ Keine</li> </ul>
Endwert/Startwert	<p><i>Endwert</i> bei Hauptzählrichtung vorwärts.</p> <p><i>Startwert</i> bei Hauptzählrichtung rückwärts.</p>	2...2147483647 (2 <sup>31</sup> -1)	2147483647 (2 <sup>31</sup> -1)
Torfunktion	<ul style="list-style-type: none"> <li>■ <i>Zählvorgang abbrechen</i>: Der Zählvorgang beginnt nach dem Schließen des Tors und erneutem Torstart wieder ab dem Ladewert.</li> <li>■ <i>Zählvorgang unterbrechen</i>: Der Zählvorgang wird nach dem Schließen des Tors und erneutem Torstart beim letzten aktuellen Zählerstand fortgesetzt.</li> </ul>	<ul style="list-style-type: none"> <li>■ Zählvorgang abbrechen</li> <li>■ Zählvorgang unterbrechen</li> </ul>	Zählvorgang abbrechen
Vergleichswert	<p>Der Zählwert wird mit dem Vergleichswert verglichen. Siehe hierzu auch Parameter "Verhalten des Ausgangs":</p> <ul style="list-style-type: none"> <li>■ Keine Hauptzählrichtung</li> <li>■ Hauptzählrichtung vorwärts</li> <li>■ Hauptzählrichtung rückwärts</li> </ul>	<p>-2<sup>31</sup> bis +2<sup>31</sup>-1</p> <p>-2<sup>31</sup> bis Endwert-1</p> <p>1 bis +2<sup>31</sup>-1</p>	0
Hysterese	<p>Die Hysterese dient zur Vermeidung von häufigen Schaltvorgängen des Ausgangs, wenn der Zählwert im Bereich des Vergleichswertes liegt.</p> <p>0 und 1 bedeuten: Hysterese abgeschaltet</p>	0 bis 255	0
max. Frequenz: Zählsignale/HW-Tor	Die maximale Frequenz der Signale Spur A/Impuls, Spur B/Richtung und Hardwaretor können Sie in festen Stufen einstellen.	10, 5, 2, 1kHz	10kHz
max. Frequenz: Latch	Die maximale Frequenz des Latch-Signals können sie in festen Stufen einstellen.	10, 5, 2, 1kHz	10kHz

Parameter	Beschreibung	Wertebereich	Vorbelegung
Signalauswertung	Am Eingang wird das Zählsignal und das Richtungssignal angeschlossen. Am Eingang wird ein Drehgeber angeschlossen (Einfach-, Zweifach- oder Vierfachausswertung).	<ul style="list-style-type: none"> <li>■ Impuls/Richtung</li> <li>■ Drehgeber einfach</li> <li>■ Drehgeber zweifach</li> <li>■ Drehgeber vierfach</li> </ul>	Impuls/Richtung
Hardware-Tor	Im aktivierten Zustand erfolgt die Torsteuerung über SW-Tor und HW-Tor, ansonsten nur über SW-Tor.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert
Zählrichtung invertiert	Im aktivierten Zustand ist das Eingangssignal "Richtung" invertiert.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert
Verhalten des Ausgangs	Abhängig von diesem Parameter wird der Ausgang und das Statusbit "Vergleicher" (STS_CMP) gesetzt.	<ul style="list-style-type: none"> <li>■ Kein Vergleich</li> <li>■ Zählerwert <math>\geq</math> Vergleichswert</li> <li>■ Zählerwert <math>\leq</math> Vergleichswert</li> <li>■ Impuls bei Vergleichswert</li> </ul>	Kein Vergleich
Impulsdauer	Bei Einstellung "Verhalten des Ausgangs: Impuls bei Vergleichswert" können Sie hier die Impulsdauer für das Ausgangssignal angeben. Es sind nur gerade Werte möglich. Der Wert wird intern mit 1,024ms multipliziert.	0 bis 510	0
Prozessalarm: Öffnen des HW-Tors	Im aktivierten Zustand wird beim Öffnen des Hardwaretors bei geöffnetem Softwaretor ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	
Prozessalarm: Schließen des HW-Tors	Im aktivierten Zustand wird beim Schließen des Hardwaretors bei geöffnetem Softwaretor ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert
Prozessalarm: Erreichen des Vergleichers	Im aktivierten Zustand wird beim Erreichen (Ansprechen) des Vergleichers ein Prozessalarm erzeugt.  Der Prozessalarm kann aber nur dann ausgelöst werden, wenn Sie unter "Verhalten des Ausgangs" nicht den Wert "kein Vergleich" eingestellt haben.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert

Parameter	Beschreibung	Wertebereich	Vorbelegung
Prozessalarm: Überlauf	Im aktivierten Zustand wird beim Überlauf (Überschreiten der oberen Zählgrenze) ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert
Prozessalarm: Unterlauf	Im Aktivierten Zustand wird beim Unterlauf (Unterschreiten der unteren Zählgrenze) ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> <li>■ aktiviert</li> <li>■ deaktiviert</li> </ul>	deaktiviert

### 6.5.2 SFB 47 - COUNT - Zähler steuern

#### Beschreibung

Bei dem SFB 47 handelt es sich um einen speziell für die VIPA CPU entwickelten Baustein, der zur Ansteuerung der Zähler dient. Der SFB ist immer zyklisch mit zugehörigem Instanz-DB aufzurufen. Hier liegen die Parameter für den SFB ab. Mit dem SFB COUNT (SFB 47) stehen Ihnen folgende Funktionalitäten zur Verfügung:

- Zähler Starten/Stoppen mit dem Softwaretor *SW\_GATE*
- Freigabe/Steuern des digitalen Ausgang
- Auslesen von Statusbits
- Auslesen von Zähler- und Latchwert
- Aufträge zum Lesen und Schreiben der internen Zählregister

## Parameter

Name	Datentyp	Adresse (Instanz-DB)	Defaultwert	Kommentar
LADDR	WORD	0.0	300h	Dieser Parameter wird nicht ausgewertet. Es wird immer die interne Ein-/ Ausgabe-Peripherie angesprochen.
CHANNEL	INT	2.0	0	Kanalnummer
SW_GATE	BOOL	4.0	FALSE	Softwaretor freigegeben
CTRL_DO	BOOL	4.1	FALSE	Ausgang DO freigegeben False: Standard Digitaler Ausgang
SET_DO	BOOL	4.2	FALSE	Parameter wird nicht ausgewertet
JOB_REQ	BOOL	4.3	FALSE	Auftragsanstoß (Flanke 0-1)
JOB_ID	WORD	6.0	0	Auftragsnummer
JOB_VAL	DINT	8.0	0	Wert für schreibende Aufträge
STS_GATE	BOOL	12.0	FALSE	Status internes Tor
STS_STRT	BOOL	12.1	FALSE	Status Hardwaretor
STS_LTCH	BOOL	12.2	FALSE	Status Latch-Eingang
STS_DO	BOOL	12.3	FALSE	Status Ausgang
STS_C_DN	BOOL	12.4	FALSE	Status Richtung rückwärts. Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf des SFB hat STS_C_DN den Wert FALSE.
STS_C_UP	BOOL	12.5	FALSE	Status Richtung vorwärts. Es wird immer die letzte Zählrichtung angezeigt. Nach dem ersten Aufruf des SFB hat STS_C_UP den Wert TRUE.
COUNTVAL	DINT	14.0	0	Aktueller Zählwert
LATCHVAL	DINT	18.0	0	Aktueller Latchwert
JOB_DONE	BOOL	22.0	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	BOOL	22.1	FALSE	Auftrag fehlerhaft
JOB_STAT	WORD	24.0	0	Auftragsfehler-Nummer

**Lokaldaten nur im Instanz-DB**

Name	Datentyp	Adresse (Instanz-DB)	Defaultwert	Kommentar
RES00	BOOL	26.0	FALSE	reserviert
RES01	BOOL	26.1	FALSE	reserviert
RES02	BOOL	26.2	FALSE	reserviert
STS_CMP	BOOL	26.3	FALSE	Status Vergleicher * Das Statusbit STS_CMP zeigt an, dass die Vergleichsbedingung des Komparators erfüllt ist oder erfüllt war. Mit STS_CMP wird auch angezeigt, dass der Ausgang gesetzt war (STS_DO = TRUE).
RES04	BOOL	26.4	FALSE	reserviert
STS_OFLW	BOOL	26.5	FALSE	Status Überlauf *
STS_UFLW	BOOL	26.6	FALSE	Status Unterlauf *
STS_ZP	BOOL	26.7	FALSE	Status Nulldurchgang * Wird nur gesetzt beim Zählen ohne Hauptzählrichtung. Zeigt Nulldurchgang an. Wird auch gesetzt, wenn der Zähler auf 0 gesetzt wird oder der Zähler ab Ladewert = 0 zählt.
JOB_OVAL	DINT	28.0		Ausgabewert für Leseaufträge
RES10	BOOL	32.0	FALSE	reserviert
RES11	BOOL	32.1	FALSE	reserviert
RES_STS	BOOL	32.2	FALSE	Status-Bits rücksetzen: Setzt die Status-Bits: STS_CMP, STS_OFLW, STS_ZP zurück. Zum Zurücksetzen der Statusbits werden zwei Aufrufe des SFB benötigt.

\*) wird mit RES\_STS zurückgesetzt



*Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.*

**Zähler Auftragsschnittstelle**

Zum Beschreiben bzw. Auslesen der Zählregister steht Ihnen mit dem SFB 47 eine Auftragsschnittstelle zur Verfügung. Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit **JOB\_DONE = TRUE** abgeschlossen sein.

**Vorgehensweise**

Der Einsatz der Auftragschnittstelle erfolgt nach folgendem Ablauf:

**1.** ► Versorgen Sie folgende Eingangsparameter:

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_REQ	BOOL	4.3	FALSE	Auftragsanstoß (Flanke 0-1) *
JOB_ID	WORD	6.0	0	Auftragsnummer: 00h Auftrag ohne Funktion 01h Zählwert schreiben 02h Ladewert schreiben 04h Vergleichswert schreiben 08h Hysterese schreiben 10h Impulsdauer schreiben 20h Endwert schreiben 82h Ladewert lesen 84h Vergleichswert lesen 88h Hysterese lesen 90h Impulsdauer lesen A0h Endwert lesen
JOB_VAL	DINT	8.0	0	Wert für schreibende Aufträge

\*) Zustand bleibt auch nach einem CPU STOP-RUN-Übergang gesetzt.

**2.** ► Rufen Sie den SFB auf. Der Auftrag wird sofort bearbeitet. *JOB\_DONE* geht für den Durchlauf des SFB auf FALSE. Im Fehlerfall wird *JOB\_ERR* = TRUE gesetzt und die Fehlerursache in *JOB\_STAT* zurückgeliefert:

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_DONE	BOOL	22.0	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	BOOL	22.1	FALSE	Auftrag fehlerhaft
JOB_STAT	WORD	24.0	0000h	Auftragsfehler-Nummer 0000h kein Fehler 0121h Vergleichswert zu klein 0122h Vergleichswert zu groß 0131h Hysterese zu klein 0132h Hysterese zu groß 0141h Impulsdauer zu klein 0142h Impulsdauer zu groß 0151h Ladewert zu klein 0152h Ladewert zu groß 0161h Zählerstand zu klein 0162h Zählerstand zu groß 01FFh Auftrags-Nr. ungültig

- 3. ▶ Mit *JOB\_DONE* = TRUE kann ein neuer Auftrag gestartet werden.
- 4. ▶ Bei Leseaufträgen finden Sie den zu lesenden Wert im Parameter *JOB\_OVAL* im Instanz-DB auf Adresse 28.

**Zulässiger Wertebereich für JOB\_VAL**

**Endlos Zählen:**

Auftrag	Gültiger Wertebereich
Zähler direkt schreiben	-2147483647 ( $-2^{31}+1$ ) ... +2147483646 ( $2^{31}-2$ )
Ladewert schreiben	-2147483647 ( $-2^{31}+1$ ) ... +2147483646 ( $2^{31}-2$ )
Vergleichswert schreiben	-2147483648 ( $-2^{31}$ ) ... +2147483647 ( $2^{31}-1$ )
Hysterese schreiben	0 ... 255
Impulsdauer schreiben*	0 ... 510ms

**Einmalig/Periodisch Zählen, keine Hauptzählrichtung:**

Auftrag	Gültiger Wertebereich
Zähler direkt schreiben	-2147483647 ( $-2^{31}+1$ ) ... +2147483646 ( $2^{31}-2$ )
Ladewert schreiben	-2147483647 ( $-2^{31}+1$ ) ... +2147483646 ( $2^{31}-2$ )
Vergleichswert schreiben	-2147483648 ( $-2^{31}$ ) ... +2147483647 ( $2^{31}-1$ )
Hysterese schreiben	0 ... 255
Impulsdauer schreiben*	0 ... 510ms

**Einmalig/Periodisch Zählen, Hauptzählrichtung vorwärts:**

Auftrag	Gültiger Wertebereich
Endwert	2 ... +2147483646 ( $2^{31}-1$ )
Zähler direkt schreiben	-2147483648 ( $-2^{31}$ ) ... Endwert -2
Ladewert schreiben	-2147483648 ( $-2^{31}$ ) ... Endwert -2
Vergleichswert schreiben	-2147483648 ( $-2^{31}$ ) ... Endwert -1
Hysterese schreiben	0 ... 255
Impulsdauer schreiben*	0 ... 510ms

**Einmalig/Periodisch Zählen, Hauptzählrichtung rückwärts:**

Auftrag	Gültiger Wertebereich
Zähler direkt schreiben	2 ... +2147483647 ( $2^{31}-1$ )
Ladewert schreiben	2 ... +2147483647 ( $2^{31}-1$ )
Vergleichswert schreiben	1 ... +2147483647 ( $2^{31}-1$ )

Auftrag	Gültiger Wertebereich
Hysterese schreiben	0 ... 255
Impulsdauer schreiben*	0 ... 510ms

\*) Es sind nur gerade Werte erlaubt. Ungerade Werte werden automatisch abgerundet.

**Latch-Funktion**

Sobald während eines Zählvorgangs am "Latch"-Eingang eines Zählers eine Flanke 0-1 auftritt, wird der aktuelle Zählerwert im entsprechenden Latch-Register gespeichert.

Mit dem Parameter *LATCHVAL* des SFB 47 haben Sie Zugriff auf den Latch-Wert.

Nach einem STOP-RUN-Übergang der CPU bleibt ein zuvor in *LATCHVAL* geladener Wert erhalten.

**6.5.3 Zähler - Funktionen****Übersicht**

Sie können vorwärts und rückwärts zählen und hierbei zwischen folgenden Zählerfunktionen wählen:

- Endlos Zählen, z.B. zur Wegerfassung mit Inkrementalgebern
- Einmalig Zählen, z.B. Stückguterfassung bis zu einer maximalen Grenze
- Periodisch Zählen, z.B. Anwendungen mit wiederholten Zählvorgängen

In den Betriebsarten "Einmalig Zählen" und "Periodisch Zählen" können Sie über die Parametrierung einen Zählerbereich als Start- bzw. Endwert definieren. Für den Zähler stehen Ihnen parametrierbare Zusatzfunktionen zur Verfügung wie z.B. Tor-Funktion, Vergleich, Hysterese und Prozessalarm.

**Hauptzählrichtung**

Über die Parametrierung haben Sie die Möglichkeit für den Zähler eine Hauptzählrichtung anzugeben. Ist "keine" ausgewählt, steht Ihnen der gesamte Zählbereich zur Verfügung:

Grenzen	Gültiger Wertebereich
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )

**Hauptzählrichtung vorwärts**

Einschränkung des Zählbereiches nach oben. Der Zähler zählt 0 bzw. Ladewert in positiver Richtung bis zum parametrierten Endwert -1 und springt dann mit dem darauffolgenden Geberimpuls wieder auf den Ladewert.

**Hauptzählrichtung rückwärts**

Einschränkung des Zählbereiches nach unten. Der Zähler zählt vom parametrierten Start- bzw. Ladewert in negativer Richtung bis zum parametrierten Endwert +1 und springt dann mit dem darauffolgenden Geberimpuls wieder auf den Startwert.

**Torfunktion abbrechen / unterbrechen**

Ist das HW-Tor freigegeben, so wirken die Torfunktionen ausschließlich auf das HW-Tor. Ein Öffnen und Schließen des SW-Tors wirkt nur unterbrechend.

*Zählvorgang abbrechen*

Der Zählvorgang beginnt nach Schließen des Tors und erneutem Torstart wieder ab dem Ladewert.

*Zählvorgang unterbrechen*

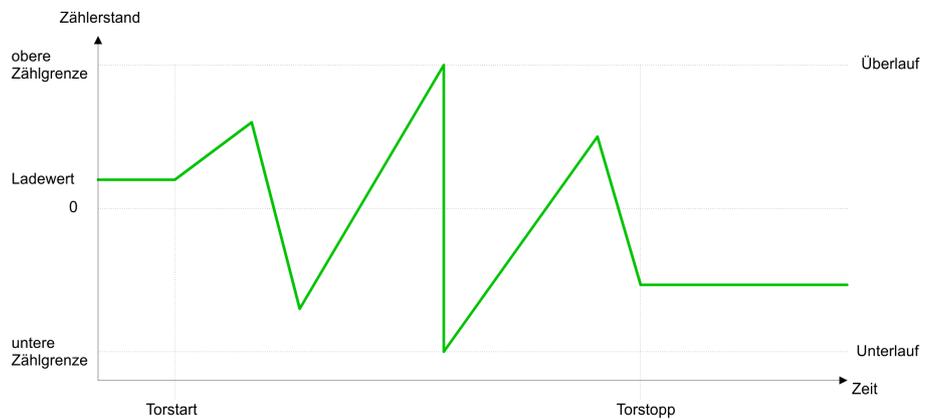
Der Zählvorgang wird nach Schließen des Tors und erneutem Torstart beim letzten aktuellen Zählerstand fortgesetzt.

**Endlos Zählen**

In dieser Betriebsart zählt der Zähler ab dem Ladewert. Erreicht der Zähler beim Vorwärtszählen die obere Zählgrenze und kommt ein weiterer Zählimpuls in positiver Richtung, springt er auf die untere Zählgrenze und zählt von dort weiter. Erreicht der Zähler beim Rückwärtszählen die untere Zählgrenze und kommt ein weiterer negativer Zählimpuls, springt er auf die obere Zählgrenze und zählt von dort weiter. Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.

Grenzen	Gültiger Wertebereich
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )

Bei Über- bzw. Unterschreitung werden die Status-Bits STS\_OFLOW bzw. STS\_UFLOW gesetzt. Diese Bits bleiben gesetzt, bis diese mit RES\_STS wieder zurückgesetzt werden. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.



**Einmalig Zählen**

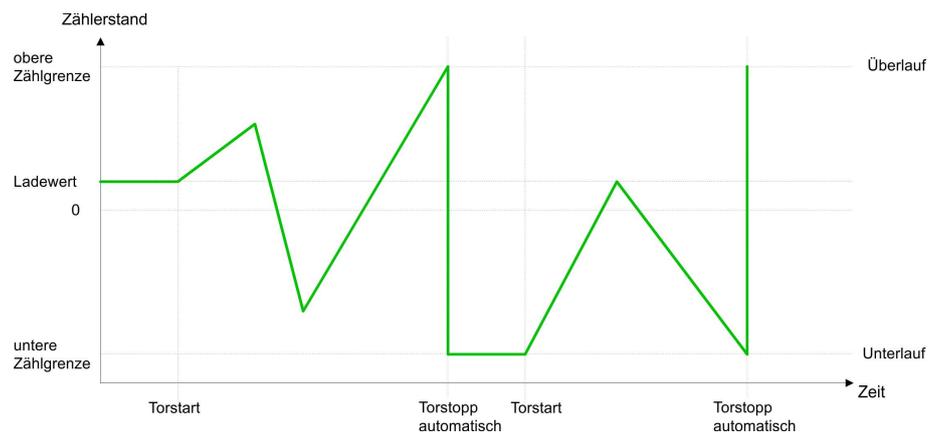
*Keine Hauptzählrichtung*

- Der Zähler zählt ab dem *Ladewert* einmalig.
- Es wird vorwärts oder rückwärts gezählt.
- Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.
- Bei Über- oder Unterlauf an den Zählgrenzen springt der Zähler auf die jeweils andere Zählgrenze und das Tor wird automatisch geschlossen.
- Zum erneuten Start des Zählvorgangs müssen Sie eine positive Flanke des Tors erzeugen.

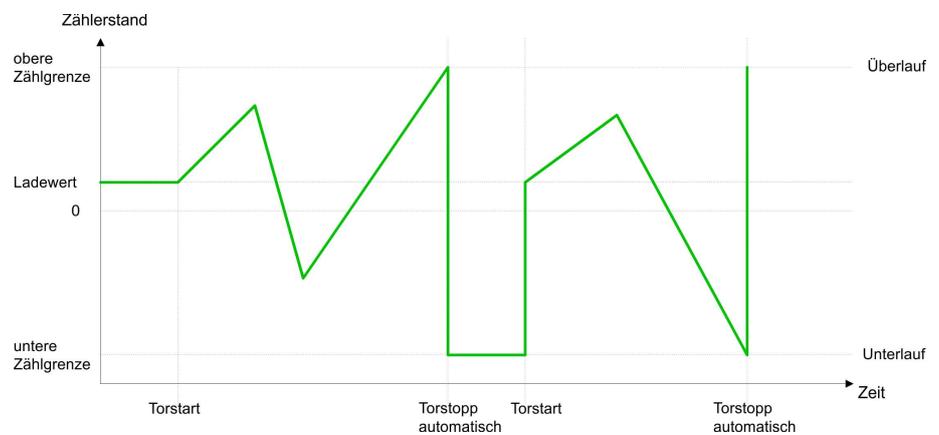
- Bei unterbrechender Torsteuerung wird der Zählvorgang beim aktuellen *Zählstand* fortgesetzt.
- Bei abbrechender Torsteuerung beginnt der Zähler ab dem *Ladewert*.

Grenzen	Gültiger Wertebereich
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )

#### Unterbrechende Torsteuerung:



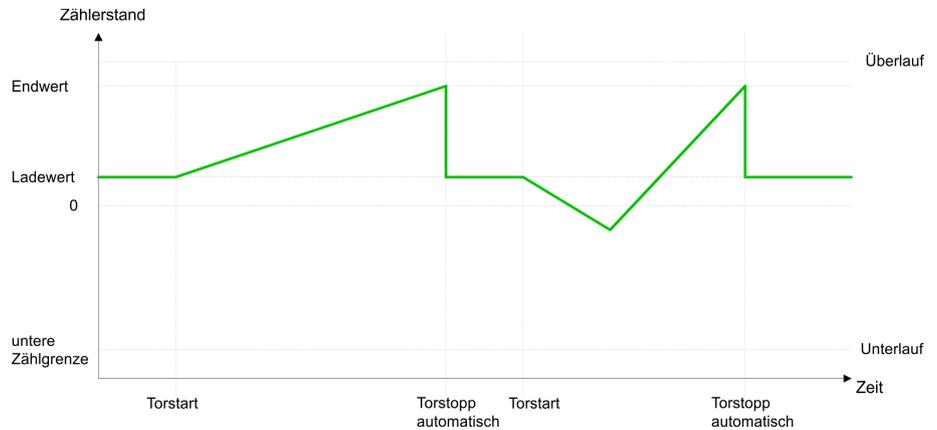
#### Abbrechende Torsteuerung:



#### Hauptzählrichtung vorwärts

- Der Zähler zählt ab dem *Ladewert* vorwärts.
- Erreicht der Zähler in positiver Richtung den *Endwert* -1, springt er beim nächsten Zählimpuls auf den *Ladewert* und das interne Tor wird automatisch geschlossen. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.
- Zum erneuten Start des Zählvorgangs müssen Sie das interne Tor wieder öffnen. Der Zähler beginnt ab dem *Ladewert*.
- Sie können über die untere Zählgrenze hinaus zählen.

Grenzen	Gültiger Wertebereich
Endwert	-2 147 483 647 ( $-2^{31} + 1$ ) bis +2 147 483 647 ( $2^{31} - 1$ )
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )



**Hauptzählrichtung rückwärts**

- Der Zähler zählt ab dem *Ladewert* rückwärts.
- Erreicht der Zähler in negativer Richtung den Endwert +1, springt er beim nächsten Zählimpuls auf den *Ladewert* und das interne Tor wird automatisch geschlossen. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.
- Zum erneuten Start des Zählvorgangs müssen Sie das interne Tor wieder öffnen. Der Zähler beginnt ab dem *Ladewert*.
- Sie können über die obere Zählgrenze hinaus zählen.

Grenzen	Gültiger Wertebereich
Endwert	-2 147 483 648 ( $-2^{31}$ ) bis +2 147 483 646 ( $2^{31} - 2$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )



**Periodisch Zählen**

*Keine Hauptzählrichtung*

- Der Zähler zählt ab *Ladewert* vorwärts oder rückwärts.
- Beim Über- oder Unterlauf an der jeweiligen Zählgrenze springt der Zähler zum *Ladewert* und zählt von dort weiter. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.
- Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.

Grenzen	Gültiger Wertebereich
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )



*Hauptzählrichtung vorwärts*

- Der Zähler zählt ab dem *Ladewert* vorwärts.
- Erreicht der Zähler in positiver Richtung den *Endwert* -1, springt er beim nächsten positiven Zählimpuls auf den *Ladewert* und zählt von dort weiter. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.
- Sie können über die untere Zählgrenze hinaus zählen.

Grenzen	Gültiger Wertebereich
Endwert	-2 147 483 647 ( $-2^{31} + 1$ ) bis +2 147 483 647 ( $2^{31} - 1$ )
Untere Zählgrenze	-2 147 483 648 ( $-2^{31}$ )



**Hauptzählrichtung rückwärts**

- Der Zähler zählt ab dem *Ladewert* rückwärts.
- Erreicht der Zähler in negativer Richtung den *Endwert* +1, springt er beim nächsten negativen Zählimpuls auf den *Ladewert* und zählt von dort weiter. Falls freigegeben, wird zusätzlich ein Prozessalarm ausgelöst.
- Sie können über die obere Zählgrenze hinaus zählen.

Grenzen	Gültiger Wertebereich
Endwert	-2 147 483 648 ( $-2^{31}$ ) bis +2 147 483 646 ( $2^{31} - 2$ )
Obere Zählgrenze	+2 147 483 647 ( $2^{31} - 1$ )



**6.5.4 Zähler - Zusatzfunktionen**

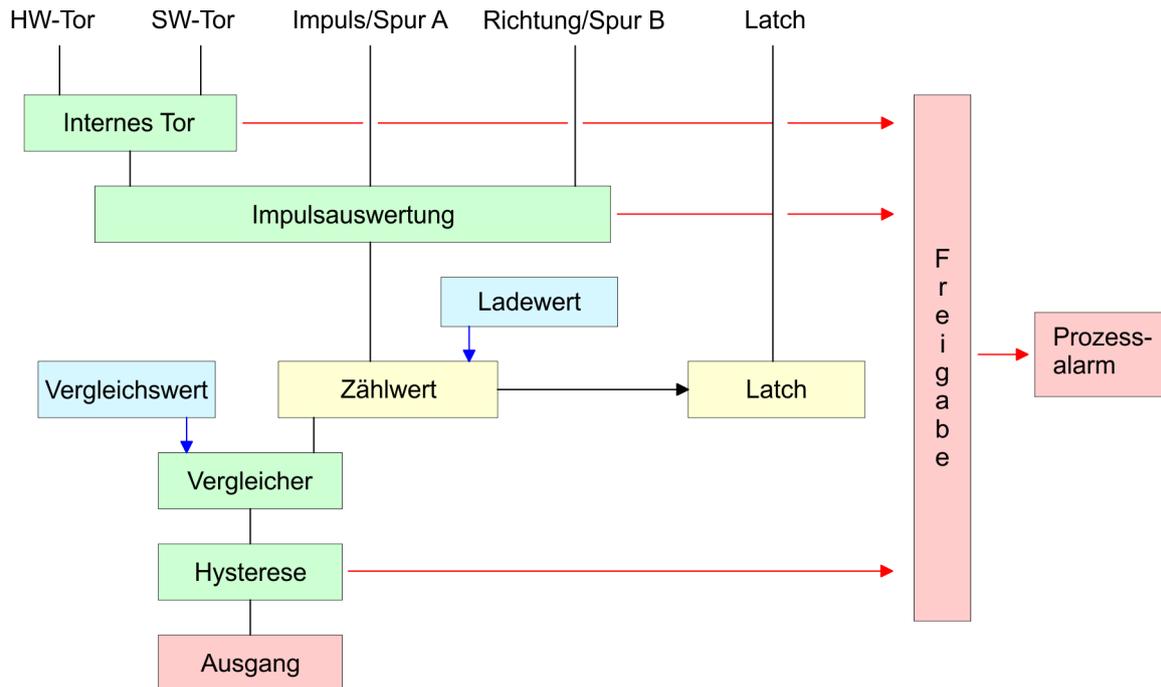
**Übersicht**

Die nachfolgend aufgeführten Zusatzfunktionen können Sie für den Zähler über die Parametrierung einstellen:

- **Tor-Funktion**  
Die Tor-Funktion dient zum Starten, Stoppen und Unterbrechen einer Zählfunktion.
- **Latchfunktion**  
Sobald am Latch-Eingang eine Flanke 0-1 auftritt, wird der aktuelle Zählerwert im Latch-Register gespeichert.
- **Vergleicher**  
Sie können einen Vergleichswert angeben, der abhängig vom Zählerwert einen Digitalausgang aktiviert bzw. einen Prozessalarm auslöst.
- **Hysterese**  
Durch Angabe einer Hysterese können Sie beispielsweise das ständige Schalten des Ausgangs verhindern, wenn der Wert eines Gebersignals um einen Vergleichswert schwankt.

**Schematischer Aufbau**

Die Abbildung zeigt, wie die Zusatzfunktionen das Zählverhalten beeinflussen. Auf den Folgeseiten sind diese Zusatzfunktionen näher erläutert:



**Tor-Funktion**

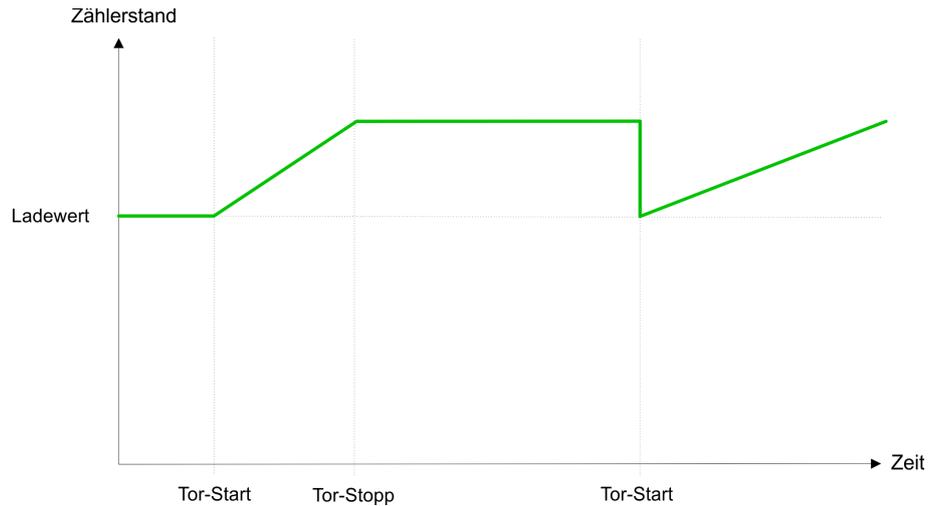
Gesteuert wird der Zähler über das interne Tor (I-Tor). Das I-Tor ist das Verknüpfungsergebnis von Hardware- (HW) und Software-Tor (SW), wobei die HW-Tor-Auswertung über die Parametrierung deaktiviert werden kann.

HW-Tor:	Öffnen (aktivieren):	Flankenwechsel 0-1 am Hardwaretor <sub>x</sub> -Eingang am Modul
	Schließen (deaktivieren):	Flankenwechsel 1-0 am Hardwaretor <sub>x</sub> -Eingang am Modul
SW-Tor:	Öffnen (aktivieren):	Im Anwenderprogramm durch Setzen von SW_GATE des SFB 47
	Schließen (deaktivieren):	Im Anwenderprogramm durch Rücksetzen von SW_GATE des SFB 47

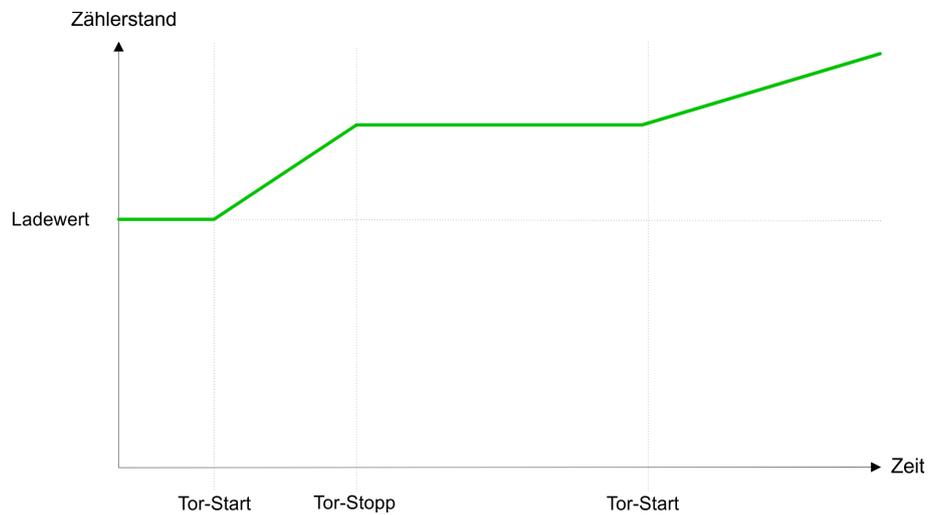
**Abbrechende und unterbrechende Tor-Funktion**

Über die Parametrierung bestimmen Sie, ob das Tor den Zählvorgang abbrechen oder unterbrechen soll.

- Bei *abbrechender Tor-Funktion* beginnt der Zählvorgang nach erneutem Tor-Start ab dem Ladewert.



- Bei *unterbrechender Tor-Funktion* wird der Zählvorgang nach Tor-Start beim letzten aktuellen Zählerwert fortgesetzt.



**Torsteuerung abbrechend, unterbrechend**

Über die Parametrieremaske stellen Sie mit dem Parameter *Torfunktion* ein, wie die CPU auf das Öffnen des SW-Tors reagieren soll. Die Verwendung des Hardware-Tors bestimmen Sie über den Parameter *HW-Tor*.

**Torsteuerung ausschließlich über SW-Tor, abbrechend (HW-Tor deaktiviert, Torfunktion: Zählvorgang abbrechen)**

SW-Tor	HW-Tor	Reaktion Zähler
Flanke 0-1	deaktiviert	Neustart mit Ladewert

**Torsteuerung ausschließlich über SW-Tor, unterbrechend (HW-Tor deaktiviert, Torfunktion: Zählvorgang unterbrechen)**

SW-Tor	HW-Tor	Reaktion Zähler
Flanke 0-1	deaktiviert	Fortsetzung

**Torsteuerung über SW/HW-Tor, abbrechend (HW-Tor: aktiviert, Torfunktion: Zählvorgang abbrechen)**

SW-Tor	HW-Tor	Reaktion Zähler
Flanke 0-1	1	Fortsetzung
1	Flanke 0-1	Neustart mit Lade- wert

**Torsteuerung über SW/HW-Tor, unterbrechend (HW-Tor aktiviert, Torfunktion: Zählvorgang unterbrechen)**

SW-Tor	HW-Tor	Reaktion Zähler
Flanke 0-1	1	Fortsetzung
1	Flanke 0-1	Fortsetzung

**Torsteuerung "Einmalig Zählen"**

Torsteuerung über SW/HW-Tor, Betriebsart "Einmalig Zählen":  
Wurde das interne Tor automatisch geschlossen, kann es nur über folgende Bedingungen geöffnet werden:

SW-Tor	HW-Tor	Reaktion I-Tor
1	Flanke 0-1	1
Flanke 0-1 (nach Flanke 0-1 am HW-Tor)	Flanke 0-1	1

**Latch-Funktion**

Sobald während eines Zählvorgangs am "Latch"-Eingang eines Zählers eine Flanke 0-1 auftritt, wird der aktuelle Zählerwert im entsprechenden Latch-Register gespeichert. Mit dem Parameter LATCHVAL des SFB 47 haben Sie Zugriff auf den Latch-Wert. Nach einem STOP-RUN-Übergang der CPU bleibt ein zuvor in LATCHVAL geladener Wert erhalten.

**Vergleicher**

In der CPU können Sie einen Vergleichswert ablegen, der dem Digitalausgang, dem Statusbit "Status Vergleich" STS\_CMP und dem Prozessalarm zugeordnet ist. Abhängig von Zähl- und Vergleichswert, kann der Digitalausgang aktiviert werden. Einen Vergleichswert können Sie über die Parametrieremaske bzw. über die Auftragsschnittstelle des SFB 47 eingeben.

**Verhalten des Ausgangs**

Über die Parametrierung können Sie das Verhalten des Zählerausgangs festlegen:

- kein Vergleich  
Der Ausgang wird wie ein normaler Ausgang geschaltet. Der SFB-Eingangsparameter CTRL\_DO ist unwirksam. Die Statusbits STS\_DO und STS\_CMP (Status Vergleicher im Instanz-DB) bleiben rückgesetzt.
- Zählerstand  $\geq$  Vergleichswert bzw. Zählerstand  $\leq$  Vergleichswert  
Solange der Zählwert größer oder gleich bzw. kleiner oder gleich dem Vergleichswert ist, bleibt der Ausgang gesetzt. Hierzu muss das Steuerbit CTRL\_DO sitzen. Das Vergleichsergebnis wird mit dem Statusbit STS\_CMP angezeigt. Rücksetzen können Sie dieses Statusbit erst, wenn die Vergleichsbedingung nicht mehr erfüllt ist.
- Impuls bei Vergleichswert  
Erreicht der Zähler den Vergleichswert, wird der Ausgang für die parametrierte Impulsdauer gesetzt. Wenn sie eine Hauptzählrichtung eingestellt haben, wird der Ausgang nur bei Erreichen des Vergleichswertes aus der Hauptzählrichtung geschaltet. Hierzu ist das Steuerbit CTRL\_DO zuvor zu setzen. Das Statusbit STS\_DO beinhaltet immer den Zustand des Digitalausgangs. Das Vergleichsergebnis wird mit dem Statusbit STS\_CMP angezeigt. Rücksetzen können Sie dieses Statusbit erst, wenn die Impulsdauer abgelaufen ist. Mit Impulsdauer = 0 ist, wird der Ausgang so lange gesetzt, wie die Vergleichsbedingung erfüllt ist.

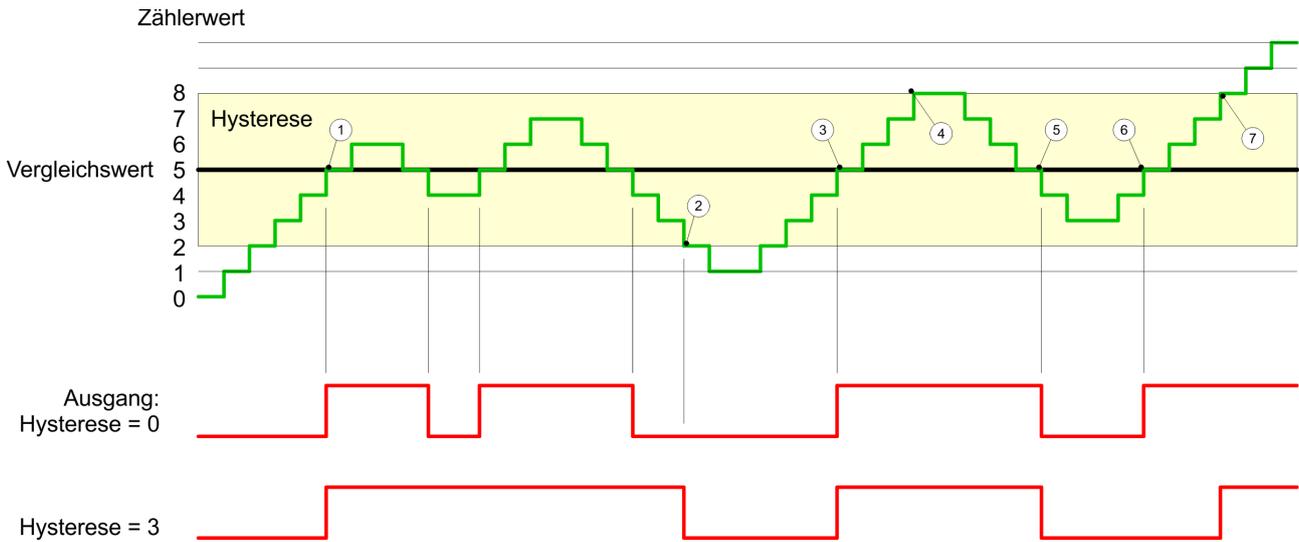
*Impulsdauer*

Zur Anpassung an die verwendeten Aktoren können Sie eine Impulsdauer vorgeben. Die Impulsdauer gibt an, wie lange der Ausgang gesetzt werden soll. Sie kann in Schritten zu 2ms zwischen 0 und 510ms vorgewählt werden. Die Impulsdauer beginnt mit dem Setzen des jeweiligen Digitalausgangs. Die Ungenauigkeit der Impulsdauer ist kleiner als 1ms. Es erfolgt keine Nachtrigierung der Impulsdauer, wenn der Vergleichswert während einer Impulsausgabe verlassen und wieder erreicht wurde. Wird die Impulsdauer im laufenden Betrieb geändert, wird sie mit dem nächsten Impuls wirksam.

**Hysterese**

- Die *Hysterese* dient beispielsweise zur Vermeidung von häufigen Schaltvorgängen des Ausgangs und Auslösen des Alarms, wenn der *Zählerwert* im Bereich des *Vergleichswertes* liegt.
- Für die Hysterese können Sie einen Bereich zwischen 0 und 255 vorgeben.
- Mit den Einstellungen 0 und 1 ist die *Hysterese* abgeschaltet.
- Die *Hysterese* wirkt auf Nulldurchgang, Über-/ Unterlauf und Vergleichswert.
- Eine aktive *Hysterese* bleibt nach der Änderung aktiv. Der neue *Hysterese*-Bereich wird beim nächsten *Hysterese*-Ereignis aktiv.

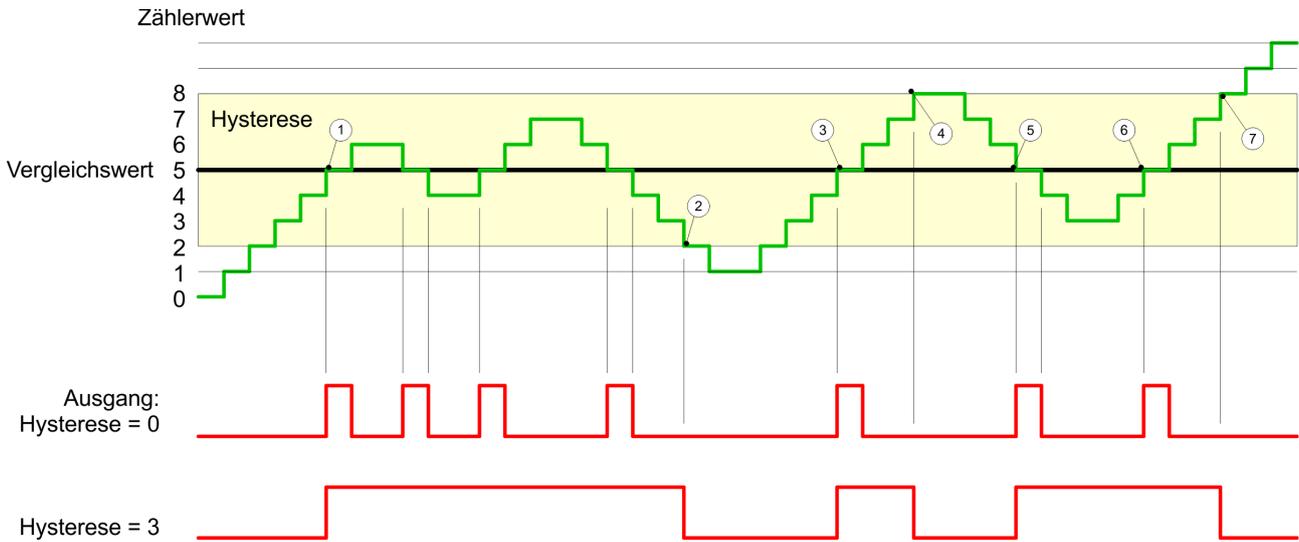
In den nachfolgenden Abbildungen ist das Verhalten des Ausgangs bei *Hysterese* 0 und *Hysterese* 3 für die entsprechenden Bedingungen dargestellt:

**Wirkungsweise bei Zählerwert  $\geq$  Vergleichswert**

- 1 Zählerwert  $\geq$  Vergleichswert  $\rightarrow$  Ausgang wird gesetzt und *Hysteresis* aktiviert
- 2 Verlassen des *Hysteresis*-Bereichs  $\rightarrow$  Ausgang wird zurückgesetzt
- 3 Zählerwert  $\geq$  Vergleichswert  $\rightarrow$  Ausgang wird gesetzt und *Hysteresis* aktiviert
- 4 Verlassen des *Hysteresis*-Bereichs, Ausgang bleibt gesetzt, da Zählerwert  $\geq$  Vergleichswert
- 5 Zählerwert  $<$  Vergleichswert und *Hysteresis* aktiv  $\rightarrow$  Ausgang wird zurückgesetzt
- 6 Zählerwert  $\geq$  Vergleichswert  $\rightarrow$  Ausgang wird nicht gesetzt, da *Hysteresis* aktiviert ist
- 7 Verlassen des *Hysteresis*-Bereichs, Ausgang wird gesetzt, da Zählerwert  $\geq$  Vergleichswert

Mit dem Erreichen der Vergleichsbedingung wird die *Hysteresis* aktiv. Bei aktiver *Hysteresis* bleibt das Vergleichsergebnis solange unverändert, bis der Zählerwert den eingestellten *Hysteresis*-Bereich verlässt. Nach Verlassen des *Hysteresis*-Bereichs wird erst wieder mit Erreichen der Vergleichsbedingungen die *Hysteresis* aktiviert.

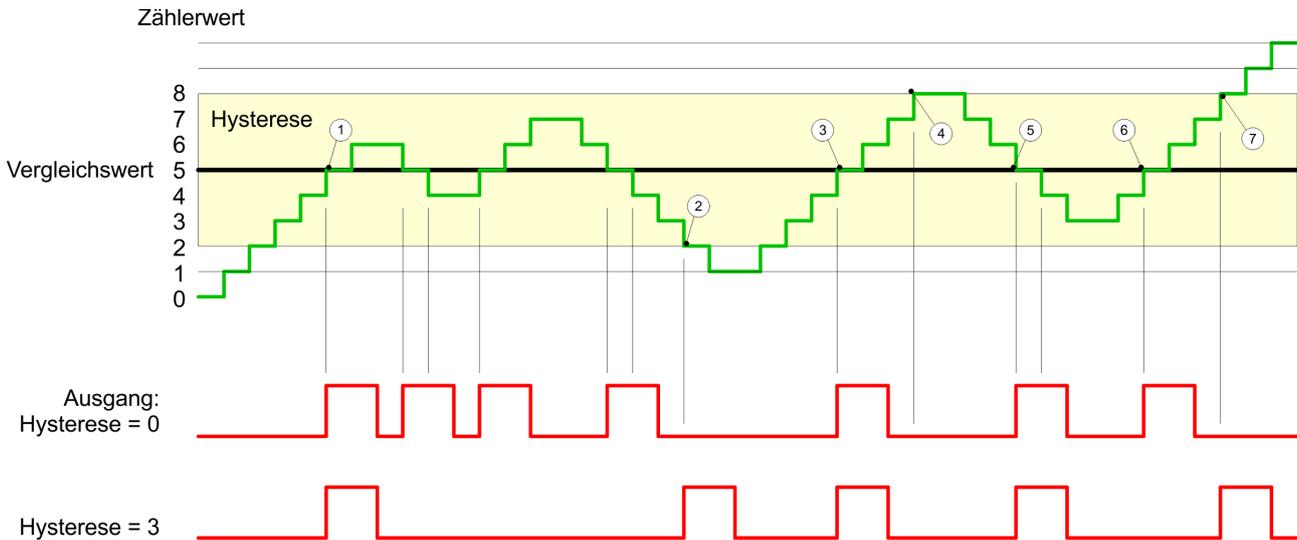
### Wirkungsweise bei Vergleichswert mit Impulsdauer Null



- 1 Zählerwert = Vergleichswert → Ausgang wird gesetzt und Hysterese aktiviert
- 2 Verlassen des Hysterese-Bereichs → Ausgang wird zurückgesetzt und Zählerwert < Vergleichswert
- 3 Zählerwert = Vergleichswert → Ausgang wird gesetzt und Hysterese aktiviert
- 4 Ausgang wird zurückgesetzt, da Verlassen des Hysterese-Bereichs, und Zählerwert > Vergleichswert
- 5 Zählerwert = Vergleichswert → Ausgang wird gesetzt und Hysterese aktiviert
- 6 Zählerwert = Vergleichswert und Hysterese aktiv → Ausgang bleibt gesetzt
- 7 Verlassen des Hysterese-Bereichs und Zählerwert > Vergleichswert → Ausgang wird zurückgesetzt

Mit dem Erreichen der Vergleichsbedingung wird die *Hysterese* aktiv. Bei aktiver *Hysterese* bleibt das Vergleichsergebnis solange unverändert, bis der Zählerwert den eingestellten *Hysterese*-Bereich verlässt. Nach Verlassen des *Hysterese*-Bereichs wird erst wieder mit Erreichen der Vergleichsbedingungen die *Hysterese* aktiviert.

### Wirkungsweise Vergleichswert mit Impulsdauer ungleich Null



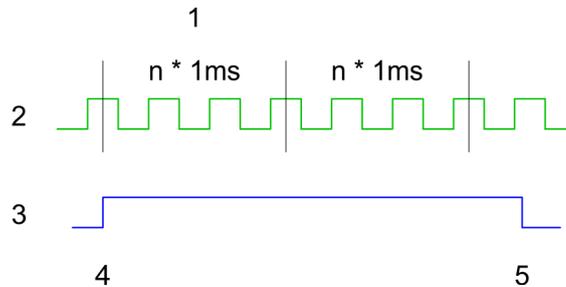
- 1 Zählerwert = Vergleichswert → Impuls der parametrisierten Dauer wird ausgegeben, die *Hysterese* aktiviert und die Zählrichtung gespeichert
- 2 Verlassen des *Hysterese*-Bereichs entgegen der gespeicherten Zählrichtung → Impuls der parametrisierten *Impulsdauer* wird ausgegeben und die *Hysterese* deaktiviert
- 3 Zählerwert = Vergleichswert → Impuls der parametrisierten *Impulsdauer* wird ausgegeben, die *Hysterese* aktiviert und die Zählrichtung gespeichert
- 4 *Hysterese*-Bereich wird ohne Änderung der Zählrichtung verlassen → *Hysterese* wird deaktiviert
- 5 Zählerwert = Vergleichswert → Impuls der parametrisierten *Impulsdauer* wird ausgegeben, die *Hysterese* aktiviert und die Zählrichtung gespeichert
- 6 Zählerwert = Vergleichswert und *Hysterese* aktiv → kein Impuls
- 7 Verlassen des *Hysterese*-Bereichs entgegen der gespeicherten Zählrichtung → Impuls der parametrisierten *Impulsdauer* wird ausgegeben und die *Hysterese* deaktiviert

Mit dem Erreichen der Vergleichsbedingung wird die *Hysterese* aktiv und ein Impuls der parametrisierten Dauer ausgegeben. Solange sich der *Zählerwert* innerhalb des *Hysterese*-Bereichs befindet, wird kein weiterer Impuls abgegeben. Mit Aktivierung der *Hysterese* wird im Modul die Zählrichtung festgehalten. Verlässt der *Zählerwert* den *Hysterese*-Bereich entgegen der gespeicherten Zählrichtung, wird ein Impuls der parametrisierten Dauer ausgegeben. Beim Verlassen des *Hysterese*-Bereichs ohne Richtungsänderung erfolgt keine Impulsabgabe.

## 6.6 Frequenzmessung

### 6.6.1 Übersicht

In dieser Betriebsart zählt die CPU die Impulse, die in einer vorgegebenen Integrationszeit eintreffen und gibt sie als Frequenzwert aus. Hierbei können Sie die Integrationszeit zwischen 10ms und 10000ms in Schritten von 1ms einstellen. Die Vorgabe der Integrationszeit kann entweder über die Parametrieremaske oder über die Auftragschnittstelle des SFB FREQUENC (SFB 48) erfolgen.



- 1 Integrationszeit
- 2 Zählimpuls
- 3 Internes Tor (SW-Tor)
- 4 Frequenzmessung Start
- 5 Frequenzmessung Stop

### Ablauf der Messung

Die Messung wird während der Integrationszeit durchgeführt und nach Ablauf der Integrationszeit aktualisiert. Ist die Periodendauer der gemessenen Frequenz größer als die parametrisierte Integrationszeit, d.h. wurde während der Messung nicht eine steigende Flanke ermittelt, so wird als Messwert 0 zurückgemeldet. Der Wert der ermittelten Frequenz wird mit der Einheit "mHz" zur Verfügung gestellt. Diesen können Sie über den Parameter *MEAS\_VAL* auslesen. Die in den technischen Daten angegebene maximale Frequenz ist unabhängig von der Anzahl aktivierter Kanäle.



- 1 Integrationszeit
- 2 Zählimpuls
- 3 Internes Tor (SW-Tor)
- 4 Berechnete Frequenz

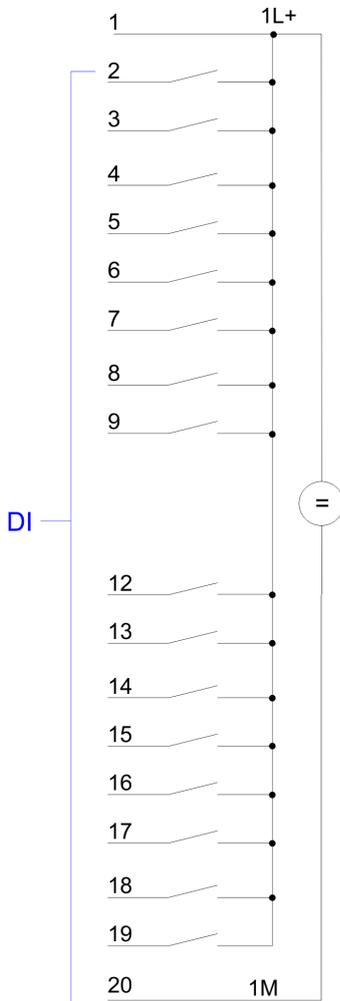


Die Zählfunktion ist während der Frequenzmessung auf dem gleichen Kanal deaktiviert.

## 6.6.2 Eingänge für die Frequenzmessung

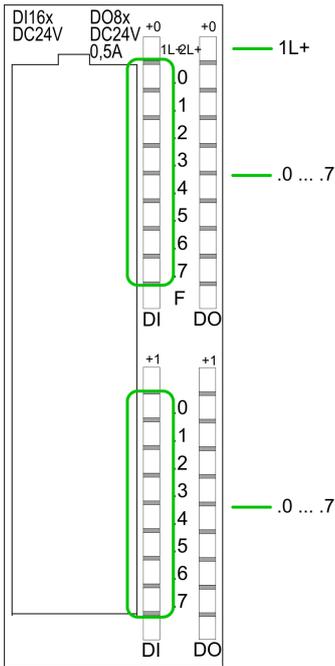
Für die Frequenzmessung schließen Sie Ihr zu messendes Signal am Eingang B an:

- Kanal 0: Pin 3
- Kanal 1: Pin 6



### Steckerbelegung X11: DI

Pin	Belegung
1	1L+ Spannungsversorgung +DC 24V
2	E+0.0 / Kanal 0 (A) / Impuls
3	E+0.1 / Kanal 0 (B) / Richtung
4	E+0.2 / Kanal 0 Hardwaretor
5	E+0.3 / Kanal 1 (A) / Impuls
6	E+0.4 / Kanal 1 (B) / Richtung
7	E+0.5 / Kanal 1 Hardwaretor
8	E+0.6
9	E+0.7
10	nicht belegt
11	nicht belegt
12	E+1.0
13	E+1.1
14	E+1.2
15	E+1.3
16	E+1.4 / Kanal 0 Latch
17	E+1.5 / Kanal 1 Latch
18	E+1.6
19	E+1.7
20	Masse 1M DI



**Statusanzeige X11: DI**

- 1L+
  - LED (grün)
  - Versorgungsspannung für DI liegt an
- .0...7
  - LEDs (grün)
  - E+0.0 ... E+0.7
  - E+1.0 ... E+1.7
  - Ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert

**6.6.3 Parametriervorgang**

1. ▶ Starten Sie den Siemens SIMATIC Manager mit Ihrem Projekt und öffnen Sie den Hardware-Konfigurator.
2. ▶ Platzieren Sie eine Profilschiene.
3. ▶ Projektieren Sie auf Steckplatz 2 die Siemens CPU 312C (6ES7 312-5BE03-0AB0 V2.6).
4. ▶ Doppelklicken Sie auf das *Zähler*-Submodul Ihrer CPU. Sie gelangen in das Dialogfeld "Eigenschaften".
5. ▶ Sobald Sie für den entsprechenden Kanal die Betriebsart "Frequenzmessen" einstellen, wird ein Dialogfenster für die Frequenzmessung mit Defaultwerten eingerichtet und angezeigt.
6. ▶ Führen Sie die gewünschten Parametrierungen durch.
7. ▶ Speichern Sie Ihr Projekt mit "*Station*" → *Speichern und übersetzen*".
8. ▶ Transferieren Sie Ihr Projekt in Ihre CPU.

**Parameterübersicht**

Nachfolgend sind alle Parameter aufgeführt, die Sie bei der Hardware-Konfiguration der Frequenzmessung verwenden können. Nicht aufgeführte Parameter werden von der CPU ignoriert.

- *Allgemein*  
Hier finden Sie die Kurzbezeichnung der Zähler-Komponente. Unter Kommentar können Sie den Einsatzzweck kommentieren.
- *Adressen*  
In diesem Register legen Sie die Anfangsadresse der Zähler-Komponente fest.

#### ■ Grundparameter

Hier können Sie einstellen, welche Alarmer die Zähler-Komponente auslösen soll. Sie haben folgende Auswahlmöglichkeiten:

- keine: Es wird kein Alarm ausgelöst.
- Prozess: Die Zählerkomponente löst einen Prozessalarm aus.
- Diagnose und Prozess: Bei der CPU wird der Diagnosealarm der digitalen Ein-/Ausgabe-Peripherie nur in Verbindung mit "Prozessalarm verloren" unterstützt.

#### ■ Frequenzmessen

Folgende Parameter sind für die Frequenzmessung relevant. Nicht aufgeführte Parameter werden von der CPU ignoriert.

- *Integrationszeit:*  
Integrationszeit für die Frequenzmessung  
Wertebereich: 10 ... 10000ms
- *Prozessalarm:*  
Messende (Ende der Integrationszeit)  
Im aktivierten Zustand wird mit jedem Ende der Integrationszeit ein Prozessalarm ausgelöst.

### 6.6.4 SFB 48 - FREQUENC - Frequenzmessung steuern

#### Beschreibung

Bei dem SFB 48 handelt es sich um einen speziell für die CPU 31xSC entwickelten Baustein, welcher der Frequenzmessung dient.

- Zur Steuerung des Frequenzmessers ist der SFB FREQUENC zyklisch (z.B. OB 1) zu verwenden.
- Der SFB ist mit zugehörigem Instanz-DB aufzurufen. Hier liegen die Parameter für den SFB ab.
- Unter anderem bietet der SFB 48 eine Auftragsschnittstelle. Hiermit haben Sie lesenden und schreibenden Zugriff auf die Register des Frequenzmessers.
- Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit *JOB\_DONE* = TRUE abgeschlossen sein.
- Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.
- Mit dem SFB FREQUENC (SFB 48) stehen Ihnen folgende Funktionalitäten zur Verfügung:
  - Frequenzmesser Starten/Stoppen mit dem Softwaretor *SW\_GATE*
  - Auslesen von Statusbits
  - Auslesen der berechneten Frequenz
  - Aufträge zum Lesen und Schreiben der internen Register des Frequenzmessers.

#### Parameter

Name	DeklARATION	Datentyp	Adresse (Inst.-DB)	Default Wert	Kommentar
LADDR	INPUT	WORD	0.0	300h	Dieser Parameter wird nicht ausgewertet. Es wird immer die interne Ein-/ Ausgabe-Peripherie angesprochen.
CHANNEL	INPUT	INT	2.0	0	Kanalnummer

Name	Deklara- tion	Datentyp	Adresse (Inst.-DB)	Default Wert	Kommentar
SW_GATE	INPUT	BOOL	4.0	FALSE	Softwaretor freigegeben
JOB_REQ	INPUT	BOOL	4.3	FALSE	Auftragsanstoß (Flanke 0-1)
JOB_ID	INPUT	WORD	6.0	0	Auftragsnummer
JOB_VAL	INPUT	DINT	8.0	0	Wert für schreibende Auf- träge
STS_GATE	OUTPUT	BOOL	12.0	FALSE	Status internes Tor
MEAS_VAL	OUTPUT	DINT	14.0	0	Berechnete Frequenz
JOB_DONE	OUTPUT	BOOL	22.0	TRUE	Neuer Auftrag kann gestartet werden.
JOB_ERR	OUTPUT	BOOL	22.1	FALSE	Auftrag fehlerhaft
JOB_STAT	OUTPUT	WORD	24.0	0	Auftragsfehler-Nummer

#### Lokaldaten nur im Instanz-DB

Name	Datentyp	Adresse (Instanz-DB)	Default	Kommentar
JOB_OVAL	DINT	28.0	-	Ausgabewert für Leseauf- träge



*Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.*

#### Frequenzmesser Auf- tragsschnittstelle

Zum Beschreiben bzw. Auslesen der Register des Frequenzmessers steht Ihnen mit dem SFB 48 eine Auftragschnittstelle zur Verfügung. Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit **JOB\_DONE = TRUE** abgeschlossen sein.

#### Vorgehensweise

Der Einsatz der Auftragschnittstelle erfolgt nach folgendem Ablauf:

➔ Versorgen Sie folgende Eingangsparameter:

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_REQ	BOOL	4.3	FALSE	Auftragsanstoß (Flanke 0-1)
JOB_ID	WORD	6.0	0	Auftragsnummer: 00h Auftrag ohne Funktion 04h Integrationszeit schreiben 84h Integrationszeit lesen
JOB_VAL	DINT	8.0	0	Wert für schreibende Aufträge. Zulässiger Wertebereich für Integrationszeit: 10 ... 10000ms

➔ Rufen Sie den SFB auf. Der Auftrag wird sofort bearbeitet. *JOB\_DONE* geht für den Durchlauf des SFB auf FALSE. Im Fehlerfall wird *JOB\_ERR* = TRUE gesetzt und die Fehlerursache in *JOB\_STAT* zurückgeliefert.

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_DONE	BOOL	22.0	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	BOOL	22.1	FALSE	Auftrag fehlerhaft
JOB_STAT	WORD	24.0	0000h	Auftragsfehler-Nummer 0000h kein Fehler 0221h Integrationszeit zu klein 0222h Integrationszeit zu groß 02FFh Auftrags-Nr. ungültig 8001h Parametrierfehler 8009h Kanal-Nr. ungültig

1. ➔ Mit *JOB\_DONE* = TRUE kann ein neuer Auftrag gestartet werden.

2. ➔ Bei Leseaufträgen finden Sie den zu lesenden Wert im Parameter *JOB\_OVAL* im Instanz-DB auf Adresse 28.

### Kanal-Nr. ungültig

(8009h und Parametrierfehler 8001h)

Haben Sie mit CHANNEL eine Kanal-Nr. größer 3 übergeben, bekommen Sie die Rückmeldung Kanal-Nr. (8009h) ungültig. Sofern Sie mit CHANNEL eine Kanal-Nr. übergeben haben, die größer ist als die maximal verfügbare Kanalanzahl der CPU, bekommen Sie einen Parametrierfehler (8001h).

**Frequenzmesser steuern**

Gesteuert wird der Frequenzmesser über das interne Tor (I-Tor). Das I-Tor ist identisch mit dem Software-Tor (SW-Tor).

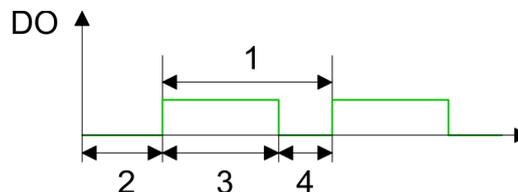
SW-Tor:

öffnen (aktivieren): Im Anwenderprogramm durch Setzen von `SW_GATE` des SFB 48

schließen (deaktivieren): Im Anwenderprogramm durch Rücksetzen von `SW_GATE` des SFB 48

**6.7 Pulsweitenmodulation - PWM****6.7.1 Übersicht****PWM**

Bei der Pulsweitenmodulation (PWM) wird durch Vorgabe von Zeitparametern eine Impulsfolge mit dem gewünschten Impuls-/Pause-Verhältnis ermittelt und über den entsprechenden Ausgabekanal ausgegeben.



- 1 Periodendauer
- 2 Einschaltverzögerung
- 3 Impulsdauer
- 4 Impulspause



*Die Zählfunktion ist während der Pulsweitenmodulation auf dem gleichen Kanal deaktiviert.*

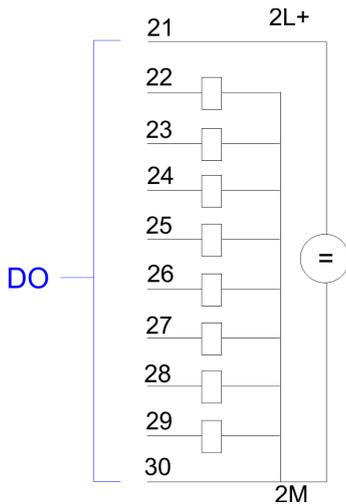
**PWM-Ausgänge**

Schließen Sie für die Pulsweitenmodulation Ihre Aktoren an folgende Pins an:

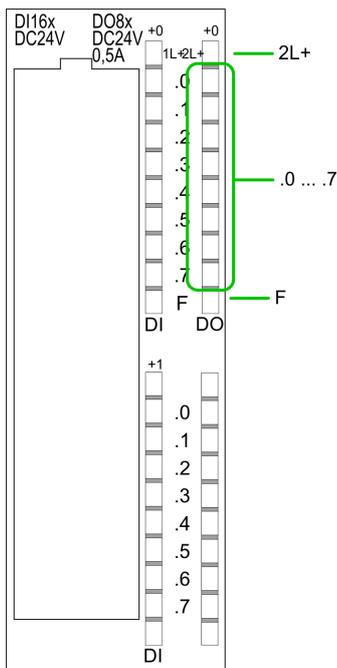
Kanal 0: Pin 22

Kanal 1: Pin 23

Die gemeinsame Masse legen Sie auf Pin 30.

**Steckerbelegung X11: DO**

Pin	Belegung
21	2L+ Spannungsversorgung +DC 24V
22	A+0.0 / Kanal 0 Ausgang
23	A+0.1 / Kanal 1 Ausgang
24	A+0.2
25	A+0.3
26	A+0.4
27	A+0.5
28	A+0.6
29	A+0.7
30	Masse 2M DO
31 ... 40	nicht belegt

**Statusanzeige X11: DO**

- 2L+
  - LED (grün)
  - Versorgungsspannung für DO liegt an
- .0... .7
  - LEDs (grün)
  - A+0.0 ... A+0.7
  - Die entsprechende LED leuchtet bei aktiviertem Ausgang
- F
  - LED (rot)
  - Fehler bei Überlast oder Kurzschluss

**6.7.2 Parametriervorgang**

1. ▶ Starten Sie den Siemens SIMATIC Manager mit Ihrem Projekt und öffnen Sie den Hardware-Konfigurator.
2. ▶ Platzieren Sie eine Profilschiene.
3. ▶ Projektieren Sie auf Steckplatz 2 die Siemens CPU 312C (6ES7 312-5BE03-0AB0 V2.6).
4. ▶ Doppelklicken Sie auf das *Zähler*-Submodul Ihrer CPU. Sie gelangen in das Dialogfeld "Eigenschaften".
5. ▶ Sobald Sie für den entsprechenden Kanal die Betriebsart "Pulsweitenmodulation" einstellen, wird ein Dialogfenster für die Pulsweitenmodulation mit Defaultwerten eingerichtet und angezeigt.

6. ► Führen Sie die gewünschten Parametrierungen durch.
7. ► Speichern Sie Ihr Projekt mit "Station" → *Speichern und übersetzen*".
8. ► Transferieren Sie Ihr Projekt in Ihre CPU.

### 6.7.2.1 Parameterübersicht

Nachfolgend sind alle Parameter aufgeführt, die Sie bei der Hardware-Konfiguration der Pulsweitenmodulation verwenden können. Nicht aufgeführte Parameter werden von der CPU ignoriert.

#### Allgemein

Hier finden Sie die Kurzbezeichnung der Zähler-Komponente. Unter *Kommentar* können Sie den Einsatzzweck kommentieren.

#### Adressen

In diesem Register legen Sie die Anfangsadresse der Zähler-Komponente fest.

#### Grundparameter

Hier können Sie einstellen, welche Alarme die Zähler-Komponente auslösen soll. Sie haben folgende Auswahlmöglichkeiten:

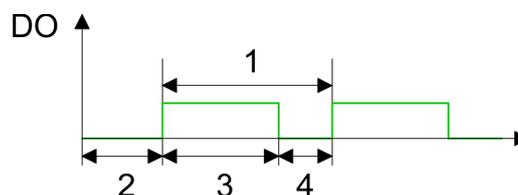
- keine: Es wird kein Alarm ausgelöst.
- Prozess: Die Zähler-Komponente löst einen Prozessalarm aus.
- Diagnose und Prozess: Bei der CPU wird der Diagnosealarm der digitalen Ein-/Ausgabe-Peripherie nur in Verbindung mit "Prozessalarm verloren" unterstützt.



*Bei der Pulsweitenmodulation gibt es keine Alarme.*

#### Pulsweitenmodulation

Folgende Parameter sind für die Pulsweitenmodulation relevant. Nicht aufgeführte Parameter werden von der CPU ignoriert.



- 1 Periodendauer
- 2 Einschaltverzögerung
- 3 Impulsdauer
- 4 Impulspause

*Ausgabeformat*

- Wählen Sie hier den Wertebereich des Ausgabewertes. Hiermit berechnet die CPU die Impulsdauer:

Ausgabeformat	Wertebereich	Impulsdauer
Promille (Default)	0 ... 1000	(Ausgabewert / 1000) x Periodendauer
S7-Analogwert	0 ... 27648	(Ausgabewert / 27648) x Periodendauer

*Zeitbasis*

- Stellen Sie hier die Zeitbasis ein, die für Auflösung und Wertebereich von Periodendauer, Mindestimpulsdauer und Einschaltverzögerung gelten soll.
- Wenn Sie das Optionsfeld "1ms" anklicken, dann können Sie Zeiten in einer Auflösung von 1ms einstellen
- Wenn Sie das Optionsfeld "0,1ms" anklicken, dann können Sie Zeiten in einer Auflösung von 0,1ms einstellen.
- Default: "0,1 ms"

*Einschaltverzögerung*

- Tragen Sie hier einen Wert für die Zeit ein, die ab dem Start der Ausgabesequenz bis zur Ausgabe des Impulses ablaufen soll. Die Impulsfolge wird nach Ablauf der Einschaltverzögerung am Kanal-Ausgang ausgegeben.
- Wertebereich: 0 ... 65535ms bzw. 0 ... 6553,5ms

*Periodendauer*

- Mit der Periodendauer definieren Sie die Länge der Ausgabesequenz, bestehend aus Impulsdauer und Impulspause.
- Wertebereich: 1 ... 65535ms bzw. 0,4 ... 6553,5ms
- Defaultwert: 20000

*Mindestimpulsdauer*

- Mit der Mindestimpulsdauer können Sie kurze Ausgangsimpulse und kurze Impulspausen unterdrücken. Alle Impulse bzw. Pausen, die kleiner als die Mindestimpulsdauer sind, werden unterdrückt. Hiermit können Sie sehr kurze Schaltimpulse (Spikes), die von der Peripherie nicht mehr registriert werden können, ausfiltern.
- Wertebereich:  
0 ... Periodendauer/2 \* 1ms bzw.  
2 ... Periodendauer/2 \* 0,1ms
- Default: 2

**6.7.3 SFB 49 - PULSE - Pulsweitenmodulation****Beschreibung**

Bei dem SFB 49 handelt es sich um einen speziell für die CPU 31xSC entwickelten Baustein, der zur Pulsweitenmodulation dient.

- Zur Steuerung der Pulsweitenmodulation ist der SFB PULSE zyklisch (z.B. OB 1) zu verwenden.
- Der SFB ist mit zugehörigem Instanz-DB aufzurufen. Hier liegen die Parameter für den SFB ab.
- Unter anderem bietet der SFB 49 eine Auftragsschnittstelle. Hiermit haben Sie lesenden und schreibenden Zugriff auf die Register der Pulsweitenmodulation.

- Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit *JOB\_DONE* = TRUE abgeschlossen sein.
- Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.
- Mit dem SFB PULSE (SFB 49) stehen Ihnen folgende Funktionalitäten zur Verfügung:
  - Pulsweitenmodulation Starten/Stoppen mit dem Softwaretor SW\_GATE
  - Freigabe/Steuern des PWM-Ausgangs
  - Auslesen von Status-Bits
  - Aufträge zum Lesen und Schreiben der internen Register der Pulsweitenmodulation

**Parameter**

Name	Deklaration	Datentyp	Adresse (Inst.-DB)	Default Wert	Kommentar
LADDR	INPUT	WORD	0.0	300h	Dieser Parameter wird nicht ausgewertet. Es wird immer die interne Ein- / Ausgabe-Peripherie angesprochen.
CHANNEL	INPUT	INT	2.0	0	Kanalnummer
SW_EN	INPUT	BOOL	4.0	FALSE	Softwaretor freigegeben
OUTP_VAL	INPUT	INT	6.0	0	Ausgabewert
JOB_REQ	INPUT	BOOL	8.0	FALSE	Auftragsanstoß (Flanke 0-1)
JOB_ID	INPUT	WORD	10.0	0	Auftragsnummer
JOB_VAL	INPUT	DINT	12.0	0	Wert für schreibende Aufträge
STS_EN	OUTPUT	BOOL	16.0	FALSE	Status internes Tor
JOB_DONE	OUTPUT	BOOL	16.3	TRUE	Neuer Auftrag kann gestartet werden.
JOB_ERR	OUTPUT	BOOL	16.4	FALSE	Auftrag fehlerhaft
JOB_STAT	OUTPUT	WORD	18.0	0	Auftragsfehler-Nummer

**Lokaldaten nur im Instanz-DB**

Name	Datentyp	Adresse (Instanz-DB)	Default	Kommentar
JOB_OVAL	DINT	20.0	-	Ausgabewert für Leseaufträge



*Pro Kanal dürfen Sie den SFB immer nur mit dem gleichen Instanz-DB aufrufen, da hier die für den internen Ablauf erforderlichen Daten abgelegt werden. Ein schreibender Zugriff auf Ausgänge des Instanz-DB ist nicht zulässig.*

### PWM Auftragsschnittstelle

Zum Beschreiben bzw. Auslesen der Register der Pulsweitenmodulation steht Ihnen mit dem SFB 49 eine Auftragsschnittstelle zur Verfügung.

Damit ein neuer Auftrag ausgeführt werden kann, muss immer der letzte Auftrag mit `JOB_DONE = TRUE` abgeschlossen sein.

### Vorgehensweise

Der Einsatz der Auftragsschnittstelle erfolgt nach folgendem Ablauf:

➔ Versorgen Sie folgende Eingangsparameter:

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_REQ	BOOL	8.0	FALSE	Auftragsanstoß (Flanke 0-1)
JOB_ID	WORD	10.0	0	Auftragsnummer: 00h Auftrag ohne Funktion 01h Periodendauer schreiben 02h Einschaltverzögerung schreiben 04h Mindestimpulsdauer schreiben 81h Periodendauer lesen 82h Einschaltverzögerung lesen 84h Mindestimpulsdauer lesen
JOB_VAL	DINT	8.0	0	Wert für schreibende Aufträge -2147483648 ( $-2^{31}$ ) bis +2147483647 ( $2^{31}-1$ )

➔ Rufen Sie den SFB auf. Der Auftrag wird sofort bearbeitet. *JOB\_DONE* geht für den Durchlauf des SFB auf FALSE. Im Fehlerfall wird *JOB\_ERR* = TRUE gesetzt und die Fehlerursache in *JOB\_STAT* zurückgeliefert.

Name	Datentyp	Adresse (DB)	Default	Kommentar
JOB_DONE	BOOL	22.0	TRUE	Neuer Auftrag kann gestartet werden
JOB_ERR	BOOL	22.1	FALSE	Auftrag fehlerhaft
JOB_STAT	WORD	24.0	0000h	Auftragsfehler-Nummer 0000h kein Fehler 0411h Periodendauer zu klein 0412h Periodendauer zu groß 0421h Einschaltverzögerung zu klein 0422h Einschaltverzögerung zu groß 0431h Mindestimpulsdauer zu klein 0432h Mindestimpulsdauer zu groß 04FFh Auftrags-Nr. ungültig 8001h Parametrierfehler 8009h Kanal-Nr. ungültig

1. ➔ Mit *JOB\_DONE* = TRUE kann ein neuer Auftrag gestartet werden.
2. ➔ Bei Leseaufträgen finden Sie den zu lesenden Wert im Parameter *JOB\_OVAL* im Instanz-DB auf Adresse 28.

**Kanal-Nr. ungültig (8009h) und Parametrierfehler (8001h)**

Haben Sie mit CHANNEL eine Kanal-Nr. größer 3 übergeben, bekommen Sie die Rückmeldung Kanal-Nr. (8009h) ungültig. Sofern Sie mit CHANNEL eine Kanal-Nr. übergeben haben, die größer ist als die maximal verfügbare Kanalanzahl der CPU, bekommen Sie einen Parametrierfehler (8001h).

**PWM steuern**

Gesteuert wird der Pulsweitenmodulation über das interne Tor (I-Tor). Das I-Tor ist identisch mit dem Software-Tor (SW-Tor).

SW-Tor:

Öffnen (aktivieren): Im Anwenderprogramm durch Setzen von *SW\_EN* des SFB 49

Schließen (deaktivieren): Im Anwenderprogramm durch Rücksetzen von *SW\_EN* des SFB 49



*Werden Werte während der PWM-Ausgabe geändert, so werden die neuen Werte erst mit dem Anfang einer neuen Periode ausgegeben. Eine gestartete Periode wird immer zu Ende geführt!*

## 6.8 Diagnose und Alarm

### Übersicht

Über die Parametrierung können Sie folgende Auslöser für einen Prozessalarm definieren, die einen Diagnosealarm auslösen können:

#### Zählerfunktion

- Flanke an einem digitalen Eingang
- Öffnen des HW-Tors (bei geöffnetem SW-Tor)
- Schließen des HW-Tors (bei geöffnetem SW-Tor)
- Erreichen des Vergleichswerts
- Überlauf bzw. bei Überschreiten der oberen Zählgrenze
- Unterlauf bzw. bei Unterschreiten der unteren Zählgrenze

#### Frequenzmessung

- Flanke an einem digitalen Eingang
- Messende (Ende der Integrationszeit)

#### Pulsweitenmodulation

- Hier gibt es keinen Prozessalarm

### 6.8.1 Prozessalarm

#### Aktivierung



*Die Aktivierung des Prozessalarms erfolgt nur, wenn gleichzeitig der Zählerausgang aktiviert wird und in den Grundparametern "Diagnose+Prozess" ausgewählt ist.*

#### Prozessalarm

Ein Prozessalarm bewirkt einen Aufruf des OB 40. Innerhalb des OB 40 haben Sie die Möglichkeit über das *Lokalwort 6* die logische Basisadresse des Moduls zu ermitteln, das den Prozessalarm ausgelöst hat. Nähere Informationen zum auslösenden Ereignis finden Sie in *Lokaldoppelwort 8*.

**Lokaldoppelwort 8 des OB 40 bei Zählerfunktion**

Lokalbyte	Bit 7...0
8	<ul style="list-style-type: none"> <li>■ Bit 0: Flanke an E+0.0</li> <li>■ Bit 1: Flanke an E+0.1</li> <li>■ Bit 2: Flanke an E+0.2</li> <li>■ Bit 3: Flanke an E+0.3</li> <li>■ Bit 4: Flanke an E+0.4</li> <li>■ Bit 5: Flanke an E+0.5</li> <li>■ Bit 6: Flanke an E+0.6</li> <li>■ Bit 7: Flanke an E+0.7</li> </ul>
9	<ul style="list-style-type: none"> <li>■ Bit 0: Flanke an E+1.0</li> <li>■ Bit 1: Flanke an E+1.1</li> <li>■ Bit 7 ... 2: reserviert</li> </ul>
10	<ul style="list-style-type: none"> <li>■ Bit 0: Tor Zähler 0 geöffnet (aktiviert)</li> <li>■ Bit 1: Tor Zähler 0 geschlossen</li> <li>■ Bit 2: Über-/Unterlauf Zähler 0</li> <li>■ Bit 3: Zähler 0 hat Vergleichswert erreicht</li> <li>■ Bit 4: Tor Zähler 1 geöffnet (aktiviert)</li> <li>■ Bit 5: Tor Zähler 1 geschlossen</li> <li>■ Bit 6: Über-/Unterlauf Zähler 1</li> <li>■ Bit 7: Zähler 1 hat Vergleichswert erreicht</li> </ul>
11	<ul style="list-style-type: none"> <li>■ Bit 7 ... 0: reserviert</li> </ul>

**Lokaldoppelwort 8 des OB 40 bei Frequenzmessung**

Lokalbyte	Bit 7...0
8	<ul style="list-style-type: none"> <li>■ Bit 0: Flanke an E+0.0</li> <li>■ Bit 1: Flanke an E+0.1</li> <li>■ Bit 2: Flanke an E+0.2</li> <li>■ Bit 3: Flanke an E+0.3</li> <li>■ Bit 4: Flanke an E+0.4</li> <li>■ Bit 5: Flanke an E+0.5</li> <li>■ Bit 6: Flanke an E+0.6</li> <li>■ Bit 7: Flanke an E+0.7</li> </ul>
9	<ul style="list-style-type: none"> <li>■ Bit 0: Flanke an E+1.0</li> <li>■ Bit 1: Flanke an E+1.1</li> <li>■ Bit 7 ... 2: reserviert</li> </ul>
10	<ul style="list-style-type: none"> <li>■ Bit 0: Messende Kanal 0 (Ende der Integrationszeit)</li> <li>■ Bit 3 ... 1: reserviert</li> <li>■ Bit 4: Messende Kanal 1 (Ende der Integrationszeit)</li> <li>■ Bit 7 ... 5: reserviert</li> </ul>
11	<ul style="list-style-type: none"> <li>■ Bit 7 ... 0: reserviert</li> </ul>

### 6.8.2 Diagnosealarm

#### Aktivierung

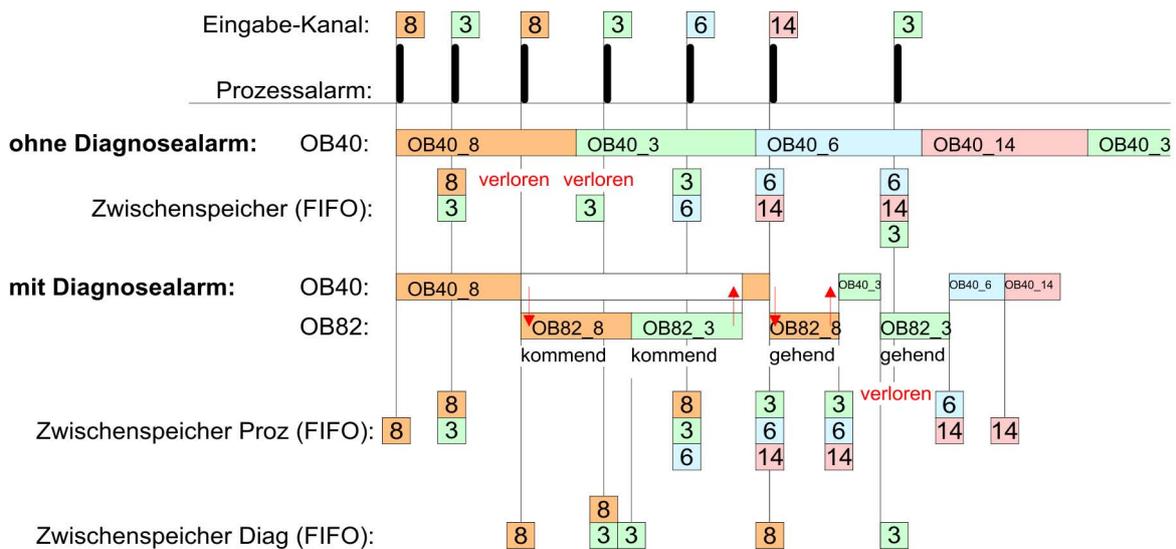


Bitte beachten Sie, dass Diagnosealarme nur dann freigegeben sind, wenn Sie eine der Technologiefunktionen (Zählen, Frequenzmessung, PWM) gewählt, und in den Grundparametern "Diagnose+Prozessalarme" ausgewählt haben.

#### Funktionsweise

Sie haben die Möglichkeit über die Parametrierung (Datensatz 7Fh) global einen Diagnosealarm für das Modul zu aktivieren. Ein Diagnosealarm tritt auf, sobald während einer Prozessalarmbearbeitung im OB 40, für das gleiche Ereignis ein weiterer Prozessalarm ausgelöst wird. Durch Auslösen eines Diagnosealarms wird die aktuelle Prozessalarm-Bearbeitung im OB 40 unterbrochen und in OB 82 zur Diagnosealarmbearbeitung<sub>kommend</sub> verzweigt. Treten während der Diagnosealarmbearbeitung auf anderen Kanälen weitere Ereignisse auf, die einen Prozess- bzw. Diagnosealarm auslösen können, werden diese zwischengespeichert. Nach Ende der Diagnosealarmbearbeitung werden zunächst alle zwischengespeicherten Diagnosealarme in der Reihenfolge ihres Auftretens abgearbeitet und anschließend alle Prozessalarme. Treten auf einem Kanal, für welchen aktuell ein Diagnosealarm<sub>kommend</sub> bearbeitet wird bzw. zwischengespeichert ist, weitere Prozessalarme auf, gehen diese verloren. Ist ein Prozessalarm, für welchen ein Diagnosealarm<sub>kommend</sub> ausgelöst wurde, abgearbeitet, erfolgt erneut ein Aufruf der Diagnosealarmbearbeitung als Diagnosealarm<sub>gehend</sub>. Alle Ereignisse eines Kanals zwischen Diagnosealarm<sub>kommend</sub> und Diagnosealarm<sub>gehend</sub> werden nicht zwischengespeichert und gehen verloren. Innerhalb dieses Zeitraums (1. Diagnosealarm<sub>kommend</sub> bis letzter Diagnosealarm<sub>gehend</sub>) leuchtet die SF-LED der CPU. Zusätzlich erfolgt für jeden Diagnosealarm<sub>kommend/gehend</sub> ein Eintrag im Diagnosepuffer der CPU.

#### Beispiel:



**Diagnosealarmbearbeitung**

Mit jedem OB 82-Aufruf erfolgt ein Eintrag mit Fehlerursache und Moduladresse im Diagnosepuffer der CPU. Unter Verwendung des SFC 59 können Sie die Diagnosebytes auslesen. Bei deaktiviertem Diagnosealarm haben Sie Zugriff auf das jeweils letzte Diagnose-Ereignis. Haben Sie in Ihrer Hardware-Konfiguration die Diagnosefunktion aktiviert, so befinden sich bei Aufruf des OB 82 die Inhalte von Datensatz 0 bereits im Lokaldoppelwort 8. Mit dem SFC 59 können Sie zusätzlich den Datensatz 1 auslesen, der weiterführende Informationen beinhaltet. Nach Verlassen des OB 82 ist keine eindeutige Zuordnung der Daten zum letzten Diagnosealarm mehr möglich. Die Datensätze des Diagnosebereichs haben folgenden Aufbau:

**Datensatz 0 Diagnose<sub>kommend</sub>**

Byte	Bit 7...0
0	<ul style="list-style-type: none"> <li>■ Bit 0: gesetzt wenn Baugruppenstörung</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: gesetzt bei Fehler extern</li> <li>■ Bit 3: gesetzt bei Kanalfehler vorhanden</li> <li>■ Bit 7 ... 4: 0 (fix)</li> </ul>
1	<ul style="list-style-type: none"> <li>■ Bit 3 ... 0: Modulkategorie                             <ul style="list-style-type: none"> <li>– 0101b: Analog</li> <li>– 1111b: Digital</li> </ul> </li> <li>■ Bit 4: Kanalinformation vorhanden</li> <li>■ Bit 7 ... 5: 0 (fix)</li> </ul>
2	<ul style="list-style-type: none"> <li>■ Bit 7 ... 0: 0 (fix)</li> </ul>
3	<ul style="list-style-type: none"> <li>■ Bit 5 ... 0: 0 (fix)</li> <li>■ Bit 6: Prozessalarm verloren</li> <li>■ Bit 7: 0 (fix)</li> </ul>

**Datensatz 0 Diagnose<sub>gehend</sub>**

Nach der Fehlerbehebung erfolgt, sofern die Diagnosealarmfreigabe noch aktiv ist, eine Diagnosemeldung<sub>gehend</sub>

Byte	Bit 7...0
0	<ul style="list-style-type: none"> <li>■ Bit 0: gesetzt wenn Baugruppenstörung</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: gesetzt bei Fehler extern</li> <li>■ Bit 3: gesetzt bei Kanalfehler vorhanden</li> <li>■ Bit 7 ... 4: 0 (fix)</li> </ul>
1	<ul style="list-style-type: none"> <li>■ Bit 3 ... 0: Modulkategorie                             <ul style="list-style-type: none"> <li>– 0101b: Analog</li> <li>– 1111b: Digital</li> </ul> </li> <li>■ Bit 4: Kanalinformation vorhanden</li> <li>■ Bit 7 ... 5: 0 (fix)</li> </ul>
2	00h (fix)
3	00h (fix)



Der Datensatz 0 ist bei Zählerfunktion, Frequenzmessung und Pulsweitenmodulation gleich aufgebaut. Unterschiede gibt es im Aufbau von Datensatz 1.

### Diagnose Datensatz 1 bei Zählerfunktionen

Der Datensatz 1 enthält die 4Byte des Datensatzes 0 und zusätzlich 12Byte modulspezifische Diagnosedaten. Die Diagnosebytes haben folgende Belegung:

Byte	Bit 7...0
0 ... 3	Inhalte Datensatz 0 ↪ "Datensatz 0 Diagnose <sub>kommend</sub> " auf Seite 141
4	<ul style="list-style-type: none"> <li>■ Bit 6 ... 0: Kanaltyp (hier 70h) <ul style="list-style-type: none"> <li>– 70h: Digitaleingabe</li> <li>– 71h: Analogeingabe</li> <li>– 72h: Digitalausgabe</li> <li>– 73h: Analogausgabe</li> <li>– 74h: Analogein-/ausgabe</li> </ul> </li> <li>■ Bit 7: Weitere Kanaltypen vorhanden <ul style="list-style-type: none"> <li>– 0: nein</li> <li>– 1: ja</li> </ul> </li> </ul>
5	Anzahl der Diagnosebits, die das Modul pro Kanal ausgibt (hier 08h)
6	Anzahl der Kanäle eines Moduls (hier 08h)
7	<ul style="list-style-type: none"> <li>■ Bit 0: Fehler in Kanalgruppe 0 (E+0.0 ... E+0.3)</li> <li>■ Bit 1: Fehler in Kanalgruppe 1 (E+0.4 ... E+0.7)</li> <li>■ Bit 2: Fehler in Kanalgruppe 2 (E+1.0 ... E+1.1)</li> <li>■ Bit 3: reserviert</li> <li>■ Bit 4: Fehler in Kanalgruppe 4 (Zähler 0)</li> <li>■ Bit 5: Fehler in Kanalgruppe 5 (Zähler 1)</li> <li>■ Bit 6: reserviert</li> <li>■ Bit 7: reserviert</li> </ul>
8	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+0.0</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+0.1</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Eingang E+0.2</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Eingang E+0.3</li> <li>■ Bit 7: 0 (fix)</li> </ul>

Byte	Bit 7...0
9	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+0.4</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+0.5</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Eingang E+0.6</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Eingang E+0.7</li> <li>■ Bit 7: 0 (fix)</li> </ul>
10	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+1.0</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+1.1</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 7 ... 4: reserviert</li> </ul>
11	<ul style="list-style-type: none"> <li>■ Bit 7 ... 0: reserviert</li> </ul>
12	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Tor Zähler 0 geschlossen</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Tor Zähler 0 geöffnet</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Über-/Unterlauf Zähler 0</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Zähler 0 hat Vergleichswert erreicht</li> <li>■ Bit 7: 0 (fix)</li> </ul>
13	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Tor Zähler 1 geschlossen</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Tor Zähler 1 geöffnet</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Über-/Unterlauf Zähler 1</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Zähler 1 hat Vergleichswert erreicht</li> <li>■ Bit 7: 0 (fix)</li> </ul>
14	reserviert
15	reserviert

**Diagnose Datensatz 1 bei *Frequenzmessung***

Der Datensatz 1 enthält die 4Byte des Datensatzes 0 und zusätzlich 12Byte modulspezifische Diagnosedaten. Die Diagnosebytes haben folgende Belegung:

Byte	Bit 7...0
0 ... 3	Inhalte Datensatz 0 ↗ " <i>Datensatz 0 Diagnose<sub>kommend</sub></i> " auf Seite 141
4	<ul style="list-style-type: none"> <li>■ Bit 6 ... 0: Kanaltyp (hier 70h) <ul style="list-style-type: none"> <li>– 70h: Digitaleingabe</li> <li>– 71h: Analogeingabe</li> <li>– 72h: Digitalausgabe</li> <li>– 73h: Analogausgabe</li> <li>– 74h: Analogein-/ausgabe</li> </ul> </li> <li>■ Bit 7: Weitere Kanaltypen vorhanden <ul style="list-style-type: none"> <li>– 0: nein</li> <li>– 1: ja</li> </ul> </li> </ul>
5	Anzahl der Diagnosebits, die das Modul pro Kanal ausgibt (hier 08h)
6	Anzahl der Kanäle eines Moduls (hier 08h)
7	<ul style="list-style-type: none"> <li>■ Bit 0: Fehler in Kanalgruppe 0 (E+0.0 ... E+0.3)</li> <li>■ Bit 1: Fehler in Kanalgruppe 1 (E+0.4 ... E+0.7)</li> <li>■ Bit 2: Fehler in Kanalgruppe 2 (E+1.0 ... E+1.1)</li> <li>■ Bit 3: reserviert</li> <li>■ Bit 4: Fehler in Kanalgruppe 4 (Frequenzmesser 0)</li> <li>■ Bit 5: Fehler in Kanalgruppe 5 (Frequenzmesser 1)</li> <li>■ Bit 6: reserviert</li> <li>■ Bit 7: reserviert</li> </ul>
8	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+0.0</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+0.1</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Eingang E+0.2</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Eingang E+0.3</li> <li>■ Bit 7: 0 (fix)</li> </ul>
9	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+0.4</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+0.5</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Eingang E+0.6</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Eingang E+0.7</li> <li>■ Bit 7: 0 (fix)</li> </ul>
10	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+1.0</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+1.1</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 7 ... 4: reserviert</li> </ul>
11	■ Bit 7 ... 0: reserviert

Byte	Bit 7...0
12	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: Messende Kanal 0 (Ende der Integrationszeit)</li> <li>■ Bit 7 ... 1: 0 (fix)</li> </ul>
13	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: Messende Kanal 1 (Ende der Integrationszeit)</li> <li>■ Bit 7 ... 1: 0 (fix)</li> </ul>
14	reserviert
15	reserviert

**Diagnose Datensatz 1 bei Pulsweitenmodulation**

Der Datensatz 1 enthält die 4Byte des Datensatzes 0 und zusätzlich 12Byte modulspezifische Diagnosedaten. Die Diagnosebytes haben folgende Belegung:

Byte	Bit 7...0
0 ... 3	Inhalte Datensatz 0 ↪ "Datensatz 0 Diagnose <sub>kommend</sub> " auf Seite 141
4	<ul style="list-style-type: none"> <li>■ Bit 6 ... 0: Kanaltyp (hier 70h)                             <ul style="list-style-type: none"> <li>– 70h: Digitaleingabe</li> <li>– 71h: Analogeingabe</li> <li>– 72h: Digitalausgabe</li> <li>– 73h: Analogausgabe</li> <li>– 74h: Analogein-/ausgabe</li> </ul> </li> <li>■ Bit 7: Weitere Kanaltypen vorhanden                             <ul style="list-style-type: none"> <li>– 0: nein</li> <li>– 1: ja</li> </ul> </li> </ul>
5	Anzahl der Diagnosebits, die das Modul pro Kanal ausgibt (hier 08h)
6	Anzahl der Kanäle eines Moduls (hier 08h)
7	<ul style="list-style-type: none"> <li>■ Bit 0: Fehler in Kanalgruppe 0 (E+0.0 ... E+0.3)</li> <li>■ Bit 1: Fehler in Kanalgruppe 1 (E+0.4 ... E+0.7)</li> <li>■ Bit 2: Fehler in Kanalgruppe 2 (E+1.0 ... E+1.1)</li> <li>■ Bit 7 ... 3: reserviert</li> </ul>
8	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"> <li>■ Bit 0: ... Eingang E+0.0</li> <li>■ Bit 1: 0 (fix)</li> <li>■ Bit 2: ... Eingang E+0.1</li> <li>■ Bit 3: 0 (fix)</li> <li>■ Bit 4: ... Eingang E+0.2</li> <li>■ Bit 5: 0 (fix)</li> <li>■ Bit 6: ... Eingang E+0.3</li> <li>■ Bit 7: 0 (fix)</li> </ul>

Byte	Bit 7...0
9	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"><li>■ Bit 0: ... Eingang E+0.4</li><li>■ Bit 1: 0 (fix)</li><li>■ Bit 2: ... Eingang E+0.5</li><li>■ Bit 3: 0 (fix)</li><li>■ Bit 4: ... Eingang E+0.6</li><li>■ Bit 5: 0 (fix)</li><li>■ Bit 6: ... Eingang E+0.7</li><li>■ Bit 7: 0 (fix)</li></ul>
10	Diagnosealarm wegen "Prozessalarm verloren" auf... <ul style="list-style-type: none"><li>■ Bit 0: ... Eingang E+1.0</li><li>■ Bit 1: 0 (fix)</li><li>■ Bit 2: ... Eingang E+1.1</li><li>■ Bit 3: 0 (fix)</li><li>■ Bit 7 ... 4: reserviert</li></ul>
11 ... 15	■ Bit 7 ... 0: reserviert

## 7 Einsatz PtP-Kommunikation

### 7.1 Schnelleinstieg

#### Allgemein

Die CPU besitzt eine RS485-Schnittstelle X3, die fix auf PtP-Kommunikation (**point to point**) eingestellt ist.

- PtP-Funktionalität
  - Mit der Funktionalität PtP ermöglicht die RS485-Schnittstelle eine serielle Punkt-zu-Punkt-Prozessankopplung zu verschiedenen Ziel- oder Quell-Systemen.

#### Protokolle

Unterstützt werden die Protokolle bzw. Prozeduren ASCII, STX/ETX, 3964R, USS und Modbus.

#### Parametrierung

Die Parametrierung der seriellen Schnittstelle erfolgt zur Laufzeit unter Einsatz des FC/SFC 216 (SER\_CFG). Hierbei sind für alle Protokolle mit Ausnahme von ASCII die Parameter in einem DB abzuliegen.

#### Kommunikation

Mit FCs/SFCs steuern Sie die Kommunikation. Das Senden erfolgt unter Einsatz des FC/SFC 217 (SER\_SND) und das Empfangen über FC/SFC 218 (SER\_RCV). Durch erneuten Aufruf des FC/SFC 217 SER\_SND bekommen Sie bei 3964R, USS und Modbus über RetVal einen Rückgabewert geliefert, der unter anderem auch aktuelle Informationen über die Quittierung der Gegenseite beinhaltet. Bei den Protokollen USS und Modbus können Sie durch Aufruf des FC/SFC 218 SER\_RCV nach einem SER\_SND das Quittungstelegramm auslesen. Die FCs/SFCs befinden sich im Lieferumfang der CPU.

#### Übersicht der FCs/SFCs für die serielle Kommunikation

Folgende FC/SFCs kommen für die serielle Kommunikation zum Einsatz:

FC/SFC		Beschreibung
FC/SFC 216	SER_CFG	RS485 Parametrieren
FC/SFC 217	SER_SND	RS485 Senden
FC/SFC 218	SER_RCV	RS485 Empfangen

### 7.2 Prinzip der Datenübertragung

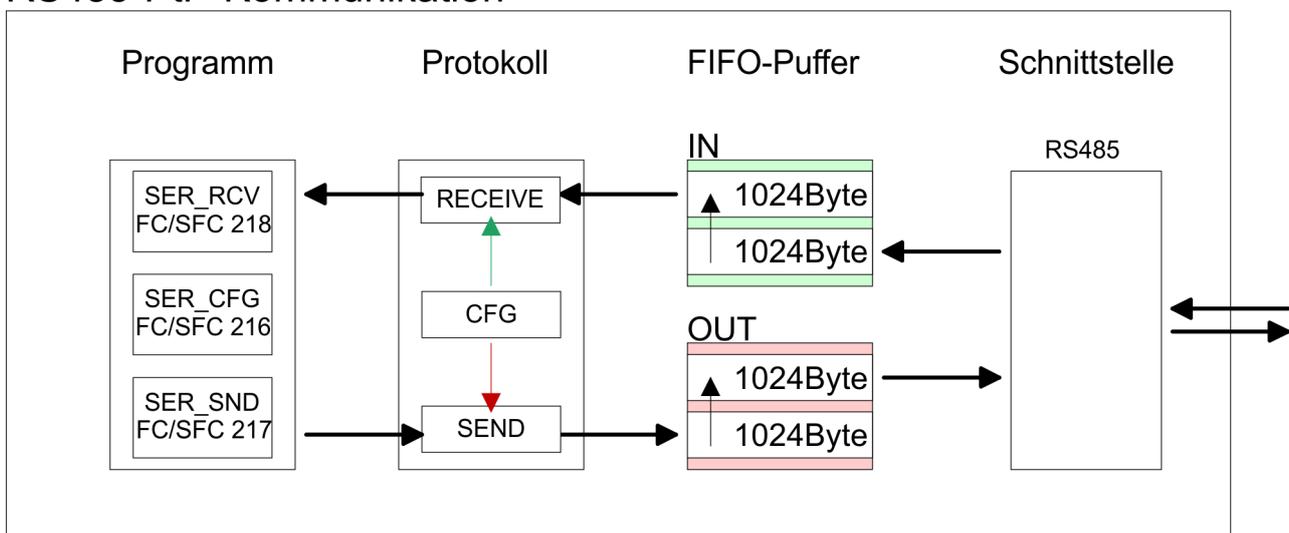
#### Übersicht

Die Datenübertragung wird zur Laufzeit über FC/SFCs gehandhabt. Das Prinzip der Datenübertragung ist für alle Protokolle identisch und soll hier kurz gezeigt werden.

- Daten, die von der CPU in den entsprechenden Datenkanal geschrieben werden, werden in einen FIFO-Sendepuffer (first in first out) mit einer Größe von 2x1024Byte abgelegt und von dort über die Schnittstelle ausgegeben.
- Empfängt die Schnittstelle Daten, werden diese in einem FIFO-Empfangspuffer mit einer Größe von 2x1024Byte abgelegt und können dort von der CPU gelesen werden.

- Sofern Daten mittels eines Protokolls übertragen werden, erfolgt die Einbettung der Daten in das entsprechende Protokoll automatisch.
- Im Gegensatz zu ASCII- und STX/ETX erfolgt bei den Protokollen 3964R, USS und Modbus die Datenübertragung mit Quittierung der Gegenseite.
- Durch erneuten Aufruf des FC/SFC 217 SER\_SND bekommen Sie über RetVal einen Rückgabewert geliefert, der unter anderem auch aktuelle Informationen über die Quittierung der Gegenseite beinhaltet.
- Zusätzlich ist bei USS und Modbus nach einem SER\_SND das Quittungstelegramm durch Aufruf des FC/SFC 218 SER\_RCV auszulesen.

## RS485-PtP-Kommunikation



### 7.3 Einsatz der RS485-Schnittstelle für PtP

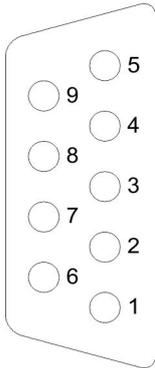
#### Übersicht

Die CPU besitzt eine RS485-Schnittstelle, die fix auf PtP-Kommunikation eingestellt ist. Parametrierung und Kommunikation erfolgen mittels SFCs.

#### Eigenschaften RS485

- Logische Zustände als Spannungsdifferenz zwischen 2 verdrehten Adern
- Serielle Busverbindung in Zweidrahttechnik im Halbduplex-Verfahren
- Datenübertragung bis 500m Entfernung
- Datenübertragungsrate bis 115,2kBit/s

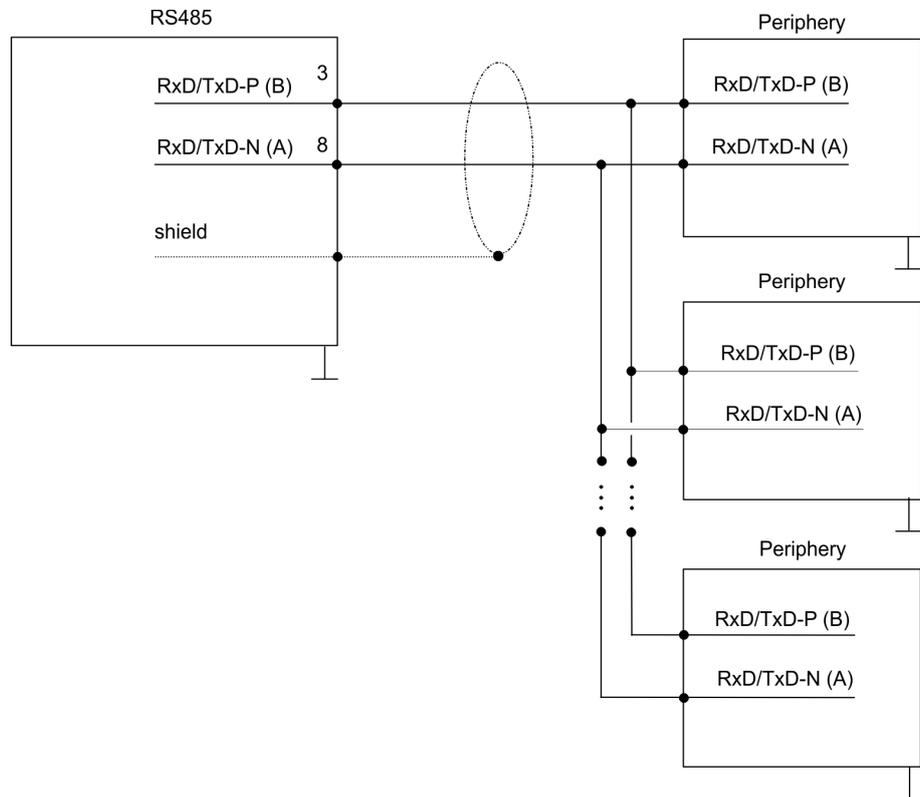
**RS485**



*9polige SubD-Buchse*

Pin	RS485
1	n.c.
2	M24V
3	RxD/TxD-P (Leitung B)
4	RTS
5	M5V
6	P5V
7	P24V
8	RxD/TxD-N (Leitung A)
9	n.c.

**Anschluss**



**7.4 Parametrierung**

**7.4.1 FC/SFC 216 - SER\_CFG**

**Beschreibung**

Die Parametrierung erfolgt zur Laufzeit unter Einsatz des FC/SFC 216 (SER\_CFG). Hierbei sind die Parameter für STX/ETX, 3964R, USS und Modbus in einem DB abzulegen.

**Parameter**

Parameter	Deklaration	Datentyp	Beschreibung
PROTOCOL	IN	BYTE	1=ASCII, 2=STX/ETX, 3=3964R
PARAMETER	IN	ANY	Zeiger zu den Protokoll-Parametern
BAUDRATE	IN	BYTE	Nr. der Baudrate
CHARLEN	IN	BYTE	0=5Bit, 1=6Bit, 2=7Bit, 3=8Bit
PARITY	IN	BYTE	0=Non, 1=Odd, 2=Even
STOPBITS	IN	BYTE	1=1Bit, 2=1,5Bit, 3=2Bit
FLOWCONTROL	IN	BYTE	1 (fix)
RETVAL	OUT	WORD	Rückgabewert (0 = OK)

Alle Zeitangaben für Timeouts sind als Hexadezimaler Wert anzugeben. Den Hex-Wert erhalten Sie, indem Sie die gewünschte Zeit in Sekunden mit der Baudrate multiplizieren.

Beispiel:

Gewünschte Zeit 8ms bei einer Baudrate von 19200Baud

Berechnung:  $19200\text{Bit/s} \times 0,008\text{s} \approx 154\text{Bit} \rightarrow (9Ah)$

Als Hex-Wert ist 9Ah vorzugeben.

**PROTOCOL**

Geben Sie hier das Protokoll an, das verwendet werden soll.

Zur Auswahl stehen:

- 1: ASCII
- 2: STX/ETX
- 3: 3964R
- 4: USS Master
- 5: Modbus RTU Master
- 6: Modbus ASCII Master

**PARAMETER (als DB)**

Bei eingestelltem ASCII-Protokoll wird dieser Parameter ignoriert.

Für die Protokolle geben Sie hier einen DB an, der die Kommunikationsparameter beinhaltet und für die jeweiligen Protokolle STX/ETX, 3964R, USS und Modbus folgenden Aufbau hat:

Datenbaustein bei STX/ETX			
DBB0:	STX1	BYTE	(1. Start-Zeichen in hexadezimaler Form)
DBB1:	STX2	BYTE	(2. Start-Zeichen in hexadezimaler Form)
DBB2:	ETX1	BYTE	(1. Ende-Zeichen in hexadezimaler Form)

DBB3:	ETX2	BYTE	(2. Ende-Zeichen in hexadezimaler Form)
DBW4:	TIMEOUT	WORD	(max. zeitlicher Abstand zwischen 2 Telegrammen)



*Das Zeichen für Start bzw. Ende sollte immer ein Wert kleiner 20 sein, ansonsten wird das Zeichen ignoriert!*

*Tragen Sie immer für nicht benutzte Zeichen FFh ein!*

Datenbaustein bei 3964R

DBB0:	Prio	BYTE	(Die Priorität beider Partner muss unterschiedlich sein)
DBB1:	ConnAttmptNr	BYTE	(Anzahl der Verbindungsaufbauversuche)
DBB2:	SendAttmptNr	BYTE	(Anzahl der Telegrammwiederholungen)
DBB4:	CharTimeout	WORD	(Zeichenverzugszeit)
DBW6:	ConfTimeout	WORD	(Quittungsverzugszeit)

Datenbaustein bei USS

DBW0:	Timeout	WORD	(Verzugszeit)
-------	---------	------	---------------

Datenbaustein bei Modbus-Master

DBW0:	Timeout	WORD	(Antwort-Verzugszeit)
-------	---------	------	-----------------------

**BAUDRATE**

Geschwindigkeit der Datenübertragung in Bit/s (Baud).

04h:	1200Baud	05h:	1800Baud	06h:	2400Baud	07h:	4800Baud
08h:	7200Baud	09h:	9600Baud	0Ah:	14400Baud	0Bh:	19200Baud
0Ch:	38400Baud	0Dh:	57600Baud	0Eh:	115200Baud		

**CHARLEN**

Anzahl der Datenbits, auf die ein Zeichen abgebildet wird.

0: 5Bit	1: 6Bit	2: 7Bit	3: 8Bit
---------	---------	---------	---------

**PARITY**

Die Parität ist je nach Wert gerade oder ungerade. Zur Paritätskontrolle werden die Informationsbits um das Paritätsbit erweitert, das durch seinen Wert ("0" oder "1") den Wert aller Bits auf einen vereinbarten Zustand ergänzt. Ist keine Parität vereinbart, wird das Paritätsbit auf "1" gesetzt, aber nicht ausgewertet.

0: NONE	1: ODD	2: EVEN
---------	--------	---------

**STOPBITS**

Die Stopbits werden jedem zu übertragenden Zeichen nachgesetzt und kennzeichnen das Ende eines Zeichens.

1: 1Bit	2: 1,5Bit	3: 2Bit
---------	-----------	---------

**FLOWCONTROL**

Der Parameter *FLOWCONTROL* wird ignoriert. Beim Senden ist RTS=1, beim Empfangen ist RTS=0.

**RETVAL FC/SFC 216 (Rückgabewert)**

Rückgabewerte, die der Baustein liefert:

Fehler-code	Beschreibung
0000h	kein Fehler
809Ah	Schnittstelle ist nicht vorhanden bzw. Schnittstelle wird für PROFIBUS verwendet.  Bei der VIPA System SLIO-CPU und FeatureSet PTP_NO ist nur das ASCII Protokoll konfigurierbar. Wird ein anderes Protokoll ausgewählt wird der FC/SFC 216 ebenfalls mit diesem Fehlercode verlassen.
8x24h	Fehler in FC/SFC-Parameter x, mit x: 1: Fehler in <i>PROTOKOLL</i> 2: Fehler in <i>PARAMETER</i> 3: Fehler in <i>BAUDRATE</i> 4: Fehler in <i>CHARLENGTH</i> 5: Fehler in <i>PARITY</i> 6: Fehler in <i>STOPBITS</i> 7: Fehler in <i>FLOWCONTROL</i> (Parameter fehlt)
809xh	Fehler in Wert des FC/SFC-Parameter x, mit x: 1: Fehler in <i>PROTOKOLL</i> 3: Fehler in <i>BAUDRATE</i> 4: Fehler in <i>CHARLENGTH</i> 5: Fehler in <i>PARITY</i> 6: Fehler in <i>STOPBITS</i>
8092h	Zugriffsfehler auf Parameter-DB (DB zu kurz)
828xh	Fehler in Parameter x von DB-Parameter mit x: 1: Fehler im 1. Parameter 2: Fehler im 2. Parameter ...

## 7.5 Kommunikation

### 7.5.1 Übersicht

Die Kommunikation erfolgt über die Sende- und Empfangsbausteine FC/SFC 217 (SER\_SND) und FC/SFC 218 (SER\_RCV). Die FCs/ SFCs befinden sich im Lieferumfang der CPU.

### 7.5.2 FC/SFC 217 - SER\_SND

#### Beschreibung

Mit diesem Baustein werden Daten über die serielle Schnittstelle gesendet. Durch erneuten Aufruf des FC/SFC 217 SER\_SND bekommen Sie bei 3964R, USS und Modbus über RETVAL einen Rückgabewert geliefert, der unter anderem auch aktuelle Informationen über die Quittierung der Gegenseite beinhaltet.

Zusätzlich ist bei USS und Modbus nach einem SER\_SND das Quittungstelegramm durch Aufruf des FC/SFC 218 SER\_RCV auszu- lesen.

#### Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DATAPTR	IN	ANY	Zeiger auf Sendedaten
DATALEN	OUT	WORD	Länge der Sendedaten
RETVAL	OUT	WORD	Rückgabewert (0 = OK)

#### DATAPTR

Geben Sie hier einen Bereich vom Typ Pointer für den Sendepuffer an, in den die Daten, die gesendet werden sollen, abzulegen sind. Anzugeben sind Typ, Anfang und Länge.

Beispiel:

Daten liegen in DB5 ab 0.0 mit einer Länge von 124Byte

DataPtr:=P#DB5.DBX0.0 BYTE 124

#### DATALEN

Wort, in dem die Anzahl der gesendeten Bytes abgelegt wird.

Werden unter **ASCII** die Daten intern mittels FC/SFC 217 schneller an die serielle Schnittstelle übertragen als sie gesendet werden können, kann aufgrund eines Pufferüberlaufs die zu sendende Datenlänge von *DATALEN* abweichen. Dies sollte im Anwenderprogramm berücksichtigt werden!

Bei **STX/ETX**, **3964R**, **Modbus** und **USS** wird immer die unter *DATAPTR* angegebene Länge oder 0 eingetragen.

#### RETVAL FC/SFC 217 (Rückgabewerte)

Rückgabewerte, die der Baustein liefert:

Fehler- code	Beschreibung
0000h	Daten gesendet - fertig
1000h	Nichts gesendet (Datenlänge 0)

Fehler-code	Beschreibung
20xxh	Protokoll wurde fehlerfrei ausgeführt mit xx-Bitmuster für Diagnose
7001h	Daten liegen im internen Puffer - aktiv (busy)
7002h	Transfer - aktiv
80xxh	Protokoll wurde fehlerhaft ausgeführt mit xx-Bitmuster für Diagnose (keine Quittung der Gegenseite)
90xxh	Protokoll wurde nicht ausgeführt mit xx-Bitmuster für Diagnose (keine Quittung der Gegenseite)
8x24h	Fehler in FC/SFC-Parameter x, mit x: 1: Fehler in <i>DATAPTR</i> 2: Fehler in <i>DATALEN</i>
8122h	Fehler in Parameter <i>DATAPTR</i> (z.B. DB zu kurz)
807Fh	Interner Fehler
809Ah	Schnittstelle nicht vorhanden bzw. Schnittstelle wird für PROFIBUS verwendet
809Bh	Schnittstelle nicht konfiguriert

### Protokollspezifische RETVAL-Werte

#### ASCII

Wert	Beschreibung
9000h	Pufferüberlauf (keine Daten gesendet)
9002h	Daten sind zu kurz (0Byte)

#### STX/ETX

Wert	Beschreibung
9000h	Pufferüberlauf (keine Daten gesendet)
9001h	Daten sind zu lang (>1024Byte)
9002h	Daten sind zu kurz (0Byte)
9004h	Unzulässiges Zeichen

#### 3964R

Wert	Beschreibung
2000h	Senden fertig ohne Fehler
80FFh	NAK empfangen - Fehler in der Kommunikation
80FEh	Datenübertragung ohne Quittierung der Gegenseite oder mit fehlerhafter Quittierung
9000h	Pufferüberlauf (keine Daten gesendet)

Wert	Beschreibung
9001h	Daten sind zu lang (>1024Byte)
9002h	Daten sind zu kurz (0Byte)

### USS

Fehler-code	Beschreibung
2000h	Senden fertig ohne Fehler
8080h	Empfangspuffer voll (kein Platz für Quittung)
8090h	Quittungsverzugszeit überschritten
80F0h	Falsche Checksumme in Rückantwort
80FEh	Falsches Startzeichen in der Rückantwort
80FFh	Falsche Slave-Adresse in der Rückantwort
9000h	Pufferüberlauf (keine Daten gesendet)
9001h	Daten sind zu lang (>1024Byte)
9002h	Daten sind zu kurz (<2Byte)

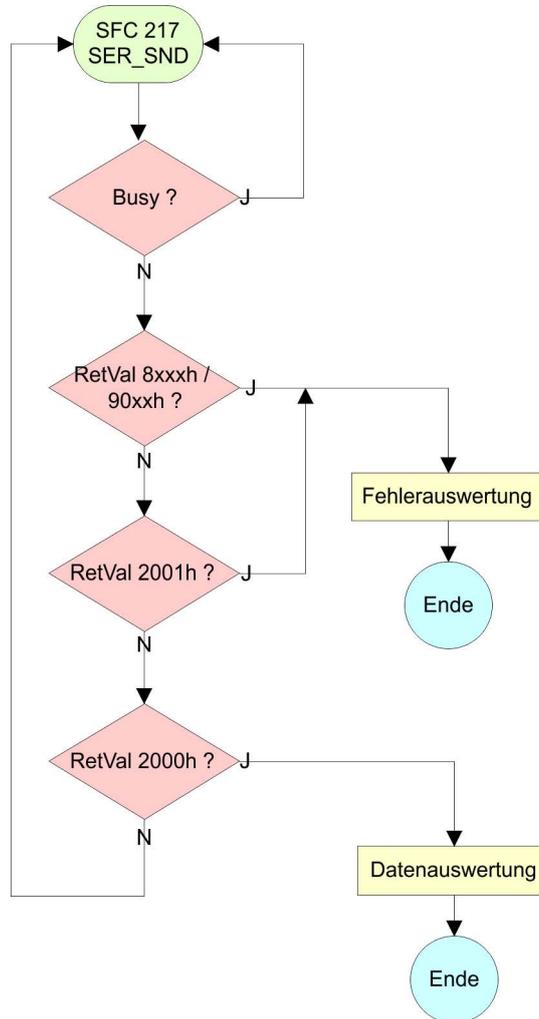
### Modbus RTU/ASCII Master

Fehler-code	Beschreibung
2000h	Senden fertig (positive Slave-Rückmeldung vorhanden)
2001h	Senden fertig (negative Slave-Rückmeldung vorhanden)
8080h	Empfangspuffer voll (kein Platz für Quittung)
8090h	Quittungsverzugszeit überschritten
80F0h	Falsche Checksumme in Rückantwort
80FDh	Länge der Rückantwort ist zu lang
80FEh	Falscher Funktionscode in der Rückantwort
80FFh	Falsche Slave-Adresse in der Rückantwort
9000h	Pufferüberlauf (keine Daten gesendet)
9001h	Daten sind zu lang (>1024Byte)
9002h	Daten sind zu kurz (<2Byte)

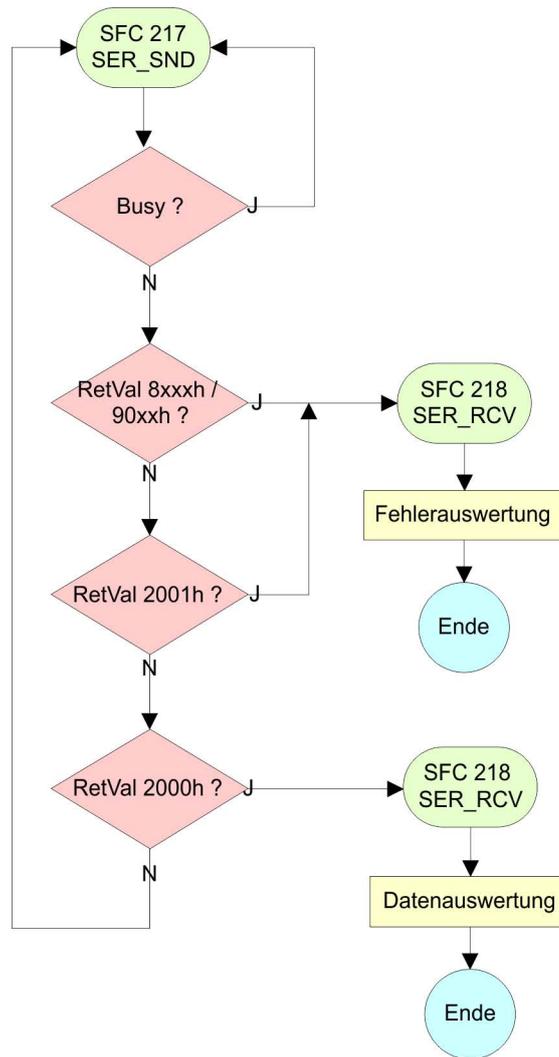
**Prinzip der Programmierung**

Nachfolgend soll kurz die Struktur zur Programmierung eines Sendeauftrags für die verschiedenen Protokolle gezeigt werden.

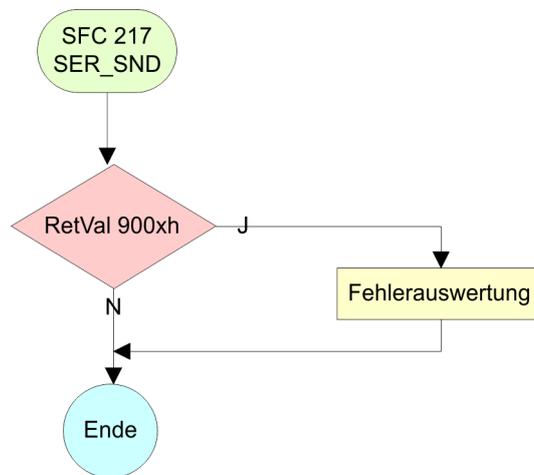
3964R



USS / Modbus



ASCII / STX/ETX



### 7.5.3 FC/SFC 218 - SER\_RCV

#### Beschreibung

Mit diesem Baustein werden Daten über die serielle Schnittstelle empfangen.

Bei den Protokollen USS und Modbus können Sie durch Aufruf des FC/SFC 218 SER\_RCV nach einem SER\_SND das Quittungstelegramm auslesen.

#### Parameter

Parameter	Deklaration	Datentyp	Beschreibung
DATAPTR	IN	ANY	Zeiger auf Empfangspuffer
DATALEN	OUT	WORD	Länge der empfangenen Daten
ERROR	OUT	WORD	Fehler-Nr.
RETVAL	OUT	WORD	Rückgabewert (0 = OK)

#### DATAPTR

Geben Sie hier einen Bereich vom Typ Pointer für den Empfangspuffer an, in den die Daten, die empfangen werden, abzulegen sind. Anzugeben sind Typ, Anfang und Länge.

Beispiel:

Daten sind in DB5 ab 0.0 mit einer Länge von 124Byte abzulegen  
DataPtr:=P#DB5.DBX0.0 BYTE 124

#### DATALEN

Wort, in dem die Anzahl der empfangenen Bytes abgelegt wird.

Bei **STX/ETX** und **3964R** wird immer die Länge der empfangenen Nutzdaten oder 0 eingetragen.

Unter **ASCII** wird hier die Anzahl der gelesenen Zeichen eingetragen. Dieser Wert kann von der Telegrammlänge abweichen.

#### ERROR

In diesem Wort erfolgt ein Eintrag im Fehlerfall.

Folgende Fehlermeldungen können protokollabhängig generiert werden:

#### ASCII

Bit	Fehler	Beschreibung
0	overrun	Überlauf, ein Zeichen konnte nicht schnell genug aus der Schnittstelle gelesen werden kann
1	framing error	Fehler, der anzeigt, dass ein definierter Bitrahmen nicht übereinstimmt, die zulässige Länge überschreitet oder eine zusätzliche Bitfolge enthält (Stoppbitfehler)
2	parity	Paritätsfehler
3	overflow	Der Puffer ist voll.

**STX/ETX**

Bit	Fehler	Beschreibung
0	over-flow	Das empfangene Telegramm übersteigt die Größe des Empfangspuffers.
1	char	Es wurde ein Zeichen außerhalb des Bereichs 20h ... 7Fh empfangen.
3	over-flow	Der Puffer ist voll.

**3964R / Modbus RTU/ASCII Master**

Bit	Fehler	Beschreibung
0	over-flow	Das empfangene Telegramm übersteigt die Größe des Empfangspuffers.

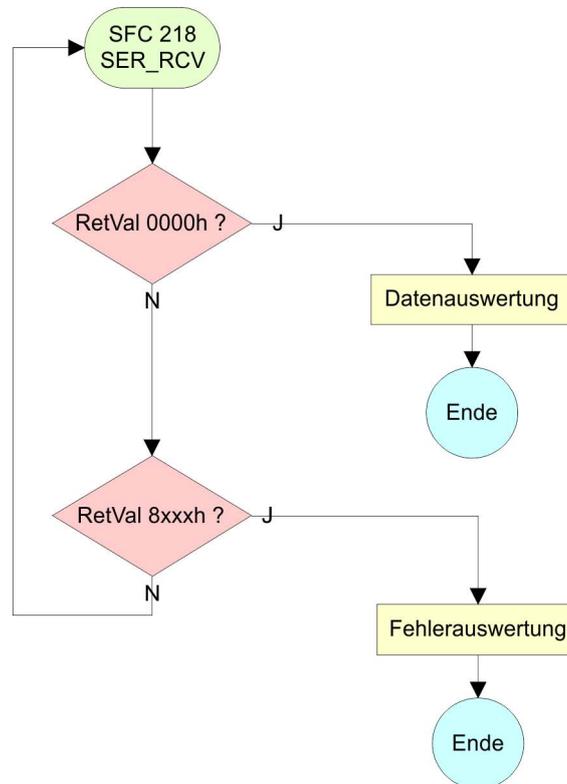
**RETVAL FC/SFC 218  
(Rückgabewert)**

Rückgabewerte, die der Baustein liefert:

Fehler-code	Beschreibung
0000h	kein Fehler
1000h	Empfangspuffer ist zu klein (Datenverlust)
8x24h	Fehler in FC/SFC-Parameter x, mit x: 1: Fehler in <i>DATAPTR</i> 2: Fehler in <i>DATALEN</i> 3: Fehler in <i>ERROR</i>
8122h	Fehler in Parameter <i>DATAPTR</i> (z.B. DB zu kurz)
809Ah	Schnittstelle nicht vorhanden bzw. Schnittstelle wird für PROFIBUS verwendet
809Bh	Schnittstelle ist nicht konfiguriert

**Prinzip der Programmierung**

Nachfolgend sehen Sie die Grundstruktur zur Programmierung eines Receive-Auftrags. Diese Struktur können Sie für alle Protokolle verwenden.

**7.6 Protokolle und Prozeduren****Übersicht**

Die CPU unterstützt folgende Protokolle und Prozeduren:

- ASCII-Übertragung
- STX/ETX
- 3964R
- USS
- Modbus

**ASCII**

Die Datenkommunikation via ASCII ist die einfachste Form der Kommunikation. Die Zeichen werden 1 zu 1 übergeben. Bei ASCII werden je Zyklus mit dem Lese-FC/SFC die zum Zeitpunkt des Aufrufs im Puffer enthaltenen Daten im parametrisierten Empfangsdatenbaustein abgelegt. Ist ein Telegramm über mehrere Zyklen verteilt, so werden die Daten überschrieben. Eine Empfangsbestätigung gibt es nicht. Der Kommunikationsablauf ist vom jeweiligen Anwenderprogramm zu steuern. Einen entsprechenden Receive\_ASCII-FB finden Sie im Service-Bereich unter [www.vipa.com](http://www.vipa.com).

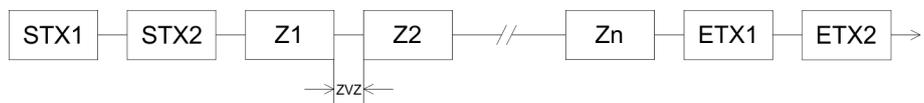
**STX/ETX**

STX/ETX ist ein einfaches Protokoll mit Start- und Ende-Kennung. Hierbei stehen STX für **S**tart of **T**ext und ETX für **E**nd of **T**ext. Die Prozedur STX/ETX wird zur Übertragung von ASCII-Zeichen eingesetzt. Sie arbeitet ohne Blockprüfung (BCC).

- Sollen Daten von der Peripherie eingelesen werden, muss das Start-Zeichen vorhanden sein, anschließend folgen die zu übertragenden Zeichen. Danach muss das Ende-Zeichen vorliegen. Abhängig von der Byte-Breite können folgende ASCII-Zeichen übertragen werden: 5Bit: nicht zulässig; 6Bit: 20...3Fh, 7Bit: 20...7Fh, 8Bit: 20...FFh.
- Die Nutzdaten, d.h. alle Zeichen zwischen Start- und Ende-Kennung, werden nach Empfang des Schlusszeichens an die CPU übergeben.
- Beim Senden der Daten von der CPU an ein Peripheriegerät werden die Nutzdaten an den FC/SFC 217 (SER\_SND) übergeben und von dort mit angefügten Start- und Endezeichen über die serielle Schnittstelle an den Kommunikationspartner übertragen.
- Es kann mit 1, 2 oder keiner Start- und mit 1, 2 oder keiner Ende-Kennung gearbeitet werden.
- Wird kein Ende-Zeichen definiert, so werden alle gelesenen Zeichen nach Ablauf einer parametrierbaren Zeichenverzugszeit (Timeout) an die CPU übergeben.

Als Start- bzw. Ende-Kennung sind alle Hex-Werte von 00h bis 1Fh zulässig. Zeichen größer 1Fh werden ignoriert und nicht berücksichtigt. In den Nutzdaten sind Zeichen kleiner 20h nicht erlaubt und können zu Fehlern führen. Die Anzahl der Start- und Endezeichen kann unterschiedlich sein (1 Start, 2 Ende bzw. 2 Start, 1 Ende oder andere Kombinationen). Für nicht verwendete Start- und Endezeichen muss in der Hardware-Konfiguration FFh eingetragen werden.

*Telegrammaufbau:*



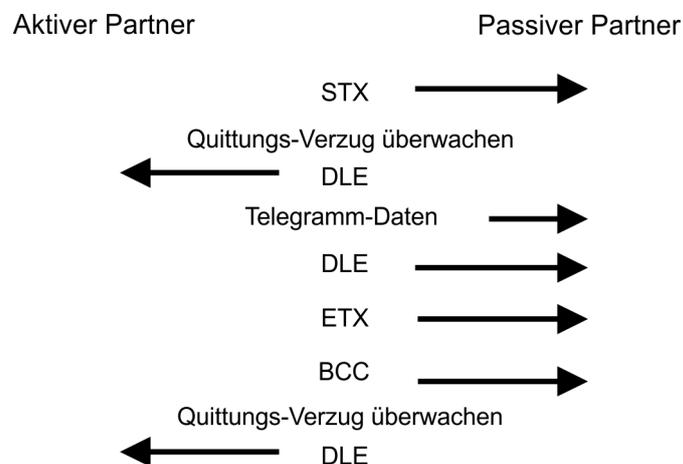
**3964**

Die Prozedur 3964R steuert die Datenübertragung bei einer Punkt-zu-Punkt-Kopplung zwischen der CPU und einem Kommunikationspartner. Die Prozedur fügt bei der Datenübertragung den Nutzdaten Steuerzeichen hinzu. Durch diese Steuerzeichen kann der Kommunikationspartner kontrollieren, ob die Daten vollständig und fehlerfrei bei ihm angekommen sind.

Die Prozedur wertet die folgenden Steuerzeichen aus:

- STX: **S**tart of **T**ext
- DLE: **D**ata **L**ink **E**scape
- ETX: **E**nd of **T**ext
- BCC: **B**lock **C**heck **C**haracter
- NAK: **N**egative **A**cknowledge

Sie können pro Telegramm maximal 255Byte übertragen.

*Prozedurablauf*

Wird ein "DLE" als Informationszeichen übertragen, so wird dieses zur Unterscheidung vom Steuerzeichen "DLE" beim Verbindungsauf- und -abbau auf der Sendeleitung doppelt gesendet (DLE-Verdoppelung). Der Empfänger macht die DLE-Verdoppelung wieder rückgängig.

Unter 3964R muss einem Kommunikationspartner eine niedrigere Priorität zugeordnet sein. Wenn beide Kommunikationspartner gleichzeitig einen Sendeauftrag erteilen, dann stellt der Partner mit niedriger Priorität seinen Sendeauftrag zurück.

**USS**

Das USS-Protokoll (**U**niverselle **s**erielle **S**chnittstelle) ist ein von Siemens definiertes serielles Übertragungsprotokoll für den Bereich der Antriebstechnik. Hiermit lässt sich eine serielle Buskopplung zwischen einem übergeordneten Master - und mehreren Slave-Systemen aufbauen. Das USS-Protokoll ermöglicht durch Vorgabe einer fixen Telegrammlänge einen zeitzyklischen Telegrammverkehr.

Folgende Merkmale zeichnen das USS-Protokoll aus:

- Mehrpunktfähige Kopplung
- Master-Slave Zugriffsverfahren

- Single-Master-System
- Maximal 32 Teilnehmer
- Einfacher, sicherer Telegrammrahmen

Es gilt:

- Am Bus können 1 Master und max. 31 Slaves angebunden sein.
- Die einzelnen Slaves werden vom Master über ein Adresszeichen im Telegramm angewählt.
- Die Kommunikation erfolgt ausschließlich über den Master im Halbduplex-Betrieb.
- Nach einem Senden-Auftrag ist das Quittungstelegramm durch Aufruf des FC/SFC 218 SER\_RCV auszulesen.

Die Telegramme für Senden und Empfangen haben folgenden Aufbau:

**Master-Slave-Telegramm**

STX	LGE	ADR	PKE		IND		PWE		STW		HSW		BCC
02h			H	L	H	L	H	L	H	L	H	L	

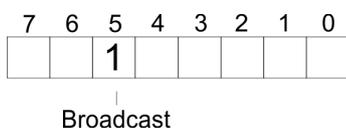
**Slave-Master-Telegramm**

STX	LGE	ADR	PKE		IND		PWE		ZSW		HIW		BCC
02h			H	L	H	L	H	L	H	L	H	L	

mit

- STX - Startzeichen
- STW - Steuerwort
- LGE - Telegrammlänge
- ZSW - Zustandswort
- ADR - Adresse
- HSW - Hauptsollwert
- PKE - Parameterkennung
- HIW - Hauptistwert
- IND - Index
- BCC - Block Check Character
- PWE - Parameterwert

**USS-Broadcast mit gesetztem Bit 5 in ADR-Byte**



Eine Anforderung kann an einen bestimmten Slave gerichtet sein oder als Broadcast-Nachricht an alle Slaves gehen. Zur Kennzeichnung einer Broadcast-Nachricht ist Bit 5 im ADR-Byte auf 1 zu setzen. Hierbei wird die Slave-Adr. (Bit 0 ... 4) ignoriert. Im Gegensatz zu einem "normalen" Senden-Auftrag ist beim Broadcast keine Telegrammauswertung über FC/SFC 218 SER\_RCV erforderlich. Nur Schreibaufträge dürfen als Broadcast gesendet werden.

**Modbus**

- Das Protokoll Modbus ist ein Kommunikationsprotokoll, das eine hierarchische Struktur mit einem Master und mehreren Slaves festlegt.
- Physikalisch arbeitet Modbus über eine serielle Halbduplex-Verbindung. Es treten keine Buskonflikte auf, da der Master immer nur mit einem Slave kommunizieren kann.
- Nach einer Anforderung vom Master wartet dieser solange auf die Antwort des Slaves, bis eine einstellbare Wartezeit abgelaufen ist. Während des Wartens ist eine Kommunikation mit einem anderen Slave nicht möglich.
- Nach einem Sende-Auftrag ist das Quittungstelegramm durch Aufruf des FC/SFC 218 SER\_RCV auszulesen.
- Die Anforderungs-Telegramme, die ein Master sendet und die Antwort-Telegramme eines Slaves haben den gleichen Aufbau:

**Telegrammaufbau**

Startzeichen	Slave-Adresse	Funktions-Code	Daten	Flusskontrolle	Endezeichen
--------------	---------------	----------------	-------	----------------	-------------

**Broadcast mit Slave-Adresse = 0**

- Eine Anforderung kann an einen bestimmten Slave gerichtet sein oder als Broadcast-Nachricht an alle Slaves gehen.
- Zur Kennzeichnung einer Broadcast-Nachricht wird die Slave-Adresse 0 eingetragen.
- Im Gegensatz zu einem "normalen" Send-Auftrag ist beim Broadcast keine Telegrammauswertung über FC/SFC 218 SER\_RCV erforderlich.
- Nur Schreibaufträge dürfen als Broadcast gesendet werden.

**ASCII-, RTU-Modus**

Bei Modbus gibt es zwei unterschiedliche Übertragungsmodi. Die Modus-Wahl erfolgt zur Laufzeit unter Einsatz des FC/SFC 216 SER\_CFG.

- ASCII-Modus: Jedes Byte wird im 2 Zeichen ASCII-Code übertragen. Die Daten werden durch Anfang- und Ende-Zeichen gekennzeichnet. Dies macht die Übertragung transparent aber auch langsam.
- RTU-Modus: Jedes Byte wird als ein Zeichen übertragen. Hierdurch haben Sie einen höheren Datendurchsatz als im ASCII-Modus. Anstelle von Anfang- und Ende-Zeichen wird eine Zeitüberwachung eingesetzt.

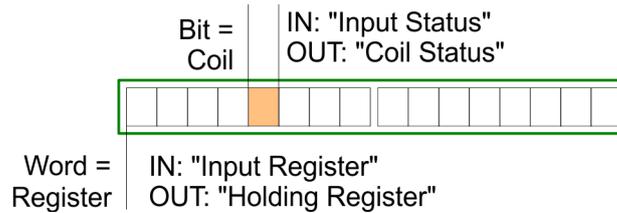
**Unterstützte Modbus-Protokolle**

Die RS485-Schnittstelle unterstützt folgende Modbus-Protokolle:

- Modbus RTU Master
- Modbus ASCII Master

**7.7 Modbus - Funktionscodes****Namenskonventionen**

Für Modbus gibt es Namenskonventionen, die hier kurz aufgeführt sind:



- Modbus unterscheidet zwischen Bit- und Wortzugriff; Bits = "Coils" und Worte = "Register".
- Bit-Eingänge werden als "Input-Status" bezeichnet und Bit-Ausgänge als "Coil-Status".
- Wort-Eingänge werden als "Input-Register" und Wort-Ausgänge als "Holding-Register" bezeichnet.

**Bereichsdefinitionen**

Üblicherweise erfolgt unter Modbus der Zugriff mittels der Bereiche 0x, 1x, 3x und 4x.

Mit 0x und 1x haben Sie Zugriff auf digitale Bit-Bereiche und mit 3x und 4x auf analoge Wort-Bereiche.

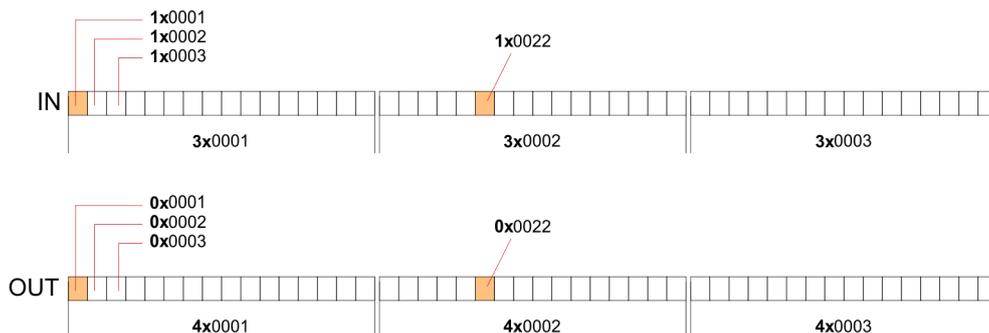
Da aber bei den CPs von VIPA keine Unterscheidung zwischen Digital- und Analogdaten stattfindet, gilt folgende Zuordnung:

0x - Bit-Bereich für Ausgabe-Daten des Masters  
Zugriff über Funktions-Code 01h, 05h, 0Fh

1x - Bit-Bereich für Eingabe-Daten des Masters  
Zugriff über Funktions-Code 02h

3x - Wort-Bereich für Eingabe-Daten des Masters  
Zugriff über Funktions-Code 04h

4x - Wort-Bereich für Ausgabe-Daten des Masters  
Zugriff über Funktions-Code 03h, 06h, 10h



Eine Beschreibung der Funktions-Codes finden Sie auf den Folge-seiten.

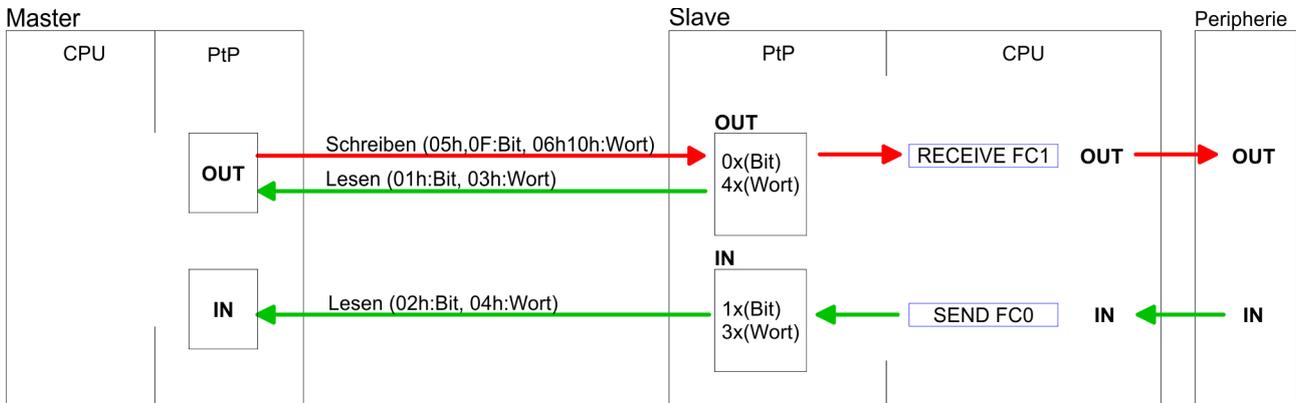
**Übersicht**

Mit folgenden Funktionscodes können Sie von einem Modbus-Master auf einen Slave zugreifen. Die Beschreibung erfolgt immer aus Sicht des Masters:

Code	Befehl	Beschreibung
01h	Read n Bits	n Bit lesen von Master-Ausgabe-Bereich 0x
02h	Read n Bits	n Bit lesen von Master-Eingabe-Bereich 1x
03h	Read n Words	n Worte lesen von Master-Ausgabe-Bereich 4x
04h	Read n Words	n Worte lesen von Master-Eingabe-Bereich 3x
05h	Write 1 Bit	1 Bit schreiben in Master-Ausgabe-Bereich 0x
06h	Write 1 Word	1 Wort schreiben in Master-Ausgabe-Bereich 4x
0Fh	Write n Bits	n Bit schreiben in Master-Ausgabe-Bereich 0x
10h	Write n Words	n Worte schreiben in Master-Ausgabe-Bereich 4x

*Sichtweise für "Eingabe"- und "Ausgabe"-Daten*

Die Beschreibung der Funktionscodes erfolgt immer aus Sicht des Masters. Hierbei werden Daten, die der Master an den Slave schickt, bis zu ihrem Ziel als "Ausgabe"-Daten (OUT) und umgekehrt Daten, die der Master vom Slave empfängt als "Eingabe"-Daten (IN) bezeichnet.



**Antwort des Slaves**

Liefert der Slave einen Fehler zurück, wird der Funktionscode mit 80h "verodert" zurückgesendet.

Ist kein Fehler aufgetreten, wird der Funktionscode zurückgeliefert.

Slave-Antwort:	Funktionscode OR 80h	→ Fehler
	Funktionscode	→ OK

**Byte-Reihenfolge im Wort**

1 Wort	
High-Byte	Low-Byte

**Prüfsumme CRC, RTU, LRC** Die aufgezeigten Prüfsummen CRC bei RTU- und LRC bei ASCII-Modus werden automatisch an jedes Telegramm angehängt. Sie werden nicht im Datenbaustein angezeigt.

**Read n Bits 01h, 02h** Code 01h: n Bit lesen von Master-Ausgabe-Bereich 0x  
Code 02h: n Bit lesen von Master-Eingabe-Bereich 1x

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse 1. Bit	Anzahl der Bits	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Anzahl der gelesenen Bytes	Daten 1. Byte	Daten 2. Byte	...	Prüfsumme CRC/LRC
1Byte	1Byte	1Byte	1Byte	1Byte		1Wort
			max. 250Byte			

**Read n Words 03h, 04h** 03h: n Worte lesen von Master-Ausgabe-Bereich 4x  
04h: n Worte lesen von Master-Eingabe-Bereich 3x

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse 1.Bit	Anzahl der Worte	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Anzahl der gelesenen Bytes	Daten 1. Wort	Daten 2. Wort	...	Prüfsumme CRC/LRC
1Byte	1Byte	1Byte	1Wort	1Wort		1Wort
			max. 125Worte			

**Write 1 Bit 05h**

Code 05h: 1 Bit schreiben in Master-Ausgabe-Bereich 0x

Eine Zustandsänderung erfolgt unter "Zustand Bit" mit folgenden Werten:

"Zustand Bit" = 0000h → Bit = 0

"Zustand Bit" = FF00h → Bit = 1

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse Bit	Zustand Bit	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Adresse Bit	Zustand Bit	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Write 1 Word 06h**

Code 06h: 1 Wort schreiben in Master-Ausgabe-Bereich 4x

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse Wort	Wert Wort	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Adresse Wort	Wert Wort	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Write n Bits 0Fh**

Code 0Fh: n Bit schreiben in Master-Ausgabe-Bereich 0x

Bitte beachten Sie, dass die Anzahl der Bits zusätzlich in Byte anzugeben sind.

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse 1. Bit	Anzahl der Bits	Anzahl der Bytes	Daten 1. Byte	Daten 2. Byte	...	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Byte	1Byte	1Byte	1Byte	1Wort
					max. 250Byte			

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Adresse 1. Bit	Anzahl der Bits	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**Write n Words 10h**

Code 10h: n Worte schreiben in Master-Ausgabe-Bereich

**Kommandotelegramm**

Slave-Adresse	Funktions-Code	Adresse 1. Wort	Anzahl der Worte	Anzahl der Bytes	Daten 1. Wort	Daten 2. Wort	...	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Byte	1Wort	1Wort	1Wort	1Wort
					max. 125Worte			

**Antworttelegramm**

Slave-Adresse	Funktions-Code	Adresse 1. Wort	Anzahl der Worte	Prüfsumme CRC/LRC
1Byte	1Byte	1Wort	1Wort	1Wort

**7.8 Modbus - Beispiel zur Kommunikation**

**Übersicht**

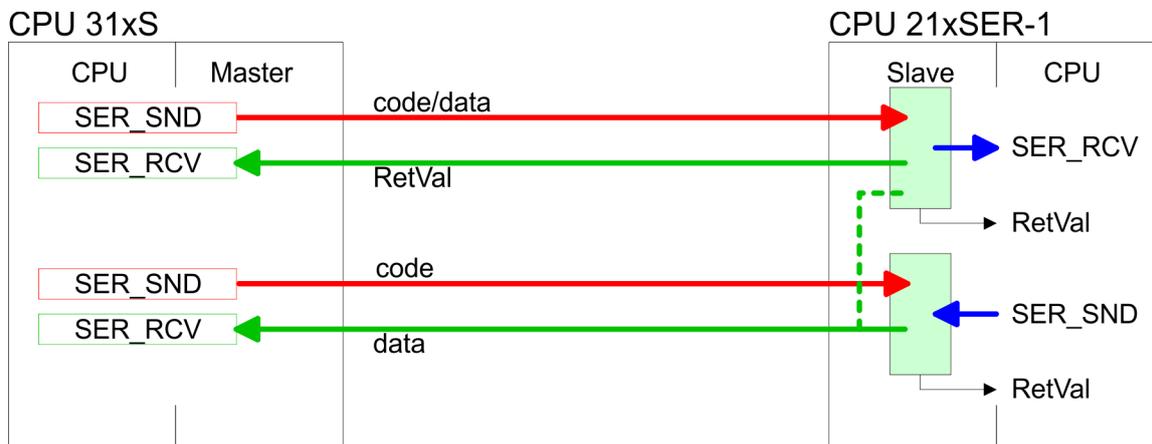
In dem Beispiel wird eine Kommunikation zwischen einem Master und einem Slave über Modbus aufgebaut. Folgende Komponenten sind für das Beispiel erforderlich:

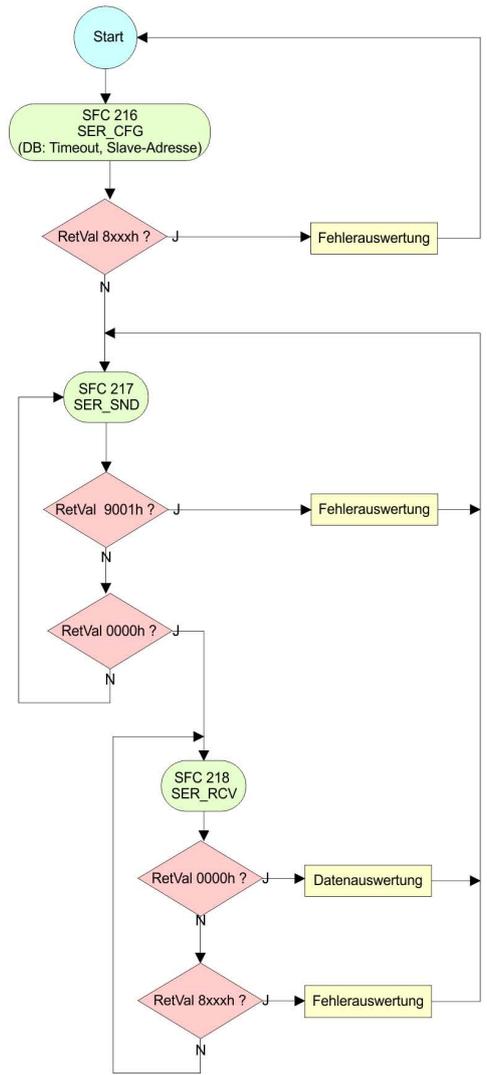
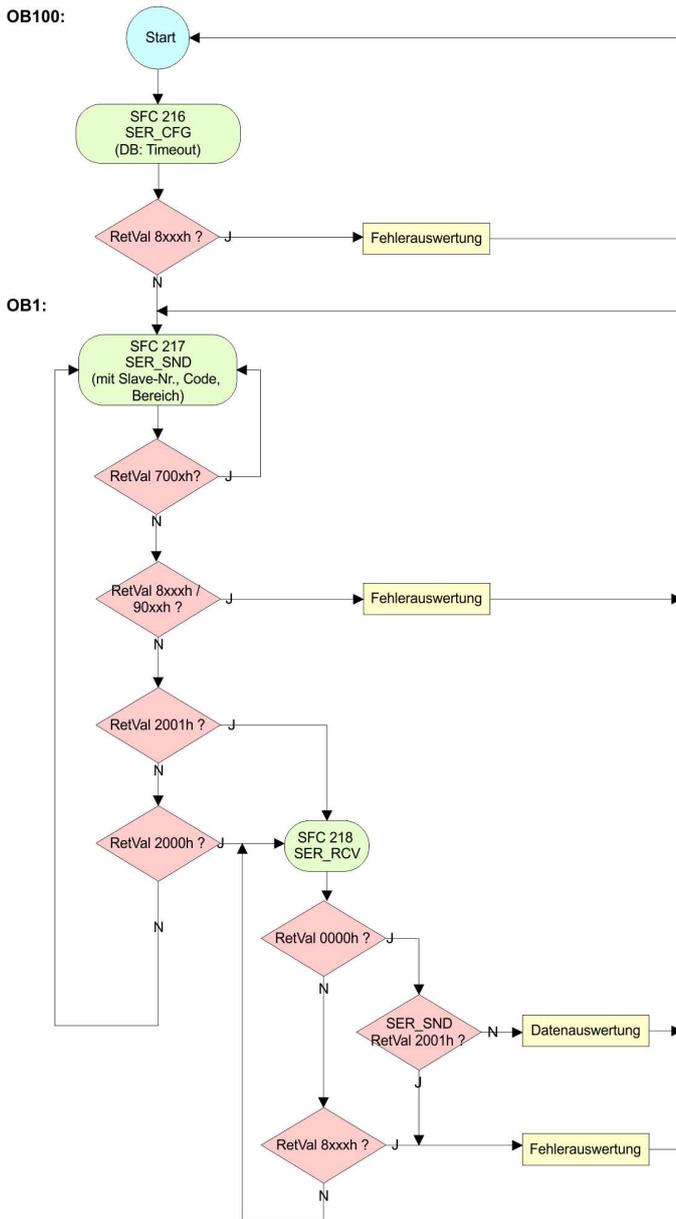
- CPU 31xS als Modbus RTU-Master
- CPU 21xSER-1 als Modbus RTU-Slave
- Siemens SIMATIC Manager und Möglichkeit für Projekttransfer
- Modbus-Kabel-Verbindung

**Vorgehensweise**

1. ► Bauen Sie ein Modbus-System bestehend aus CPU 31xS als Modbus-Master und CPU 21xSER-1 als Modbus-Slave und Modbus-Kabel auf.
2. ► Projektieren Sie die Master-Seite! Erstellen Sie hierzu ein SPS-Anwenderprogramm nach folgender Struktur:
  - OB 100:  
Aufruf SFC 216 (Konfiguration als Modbus RTU-Master) mit Timeout-Angabe und Fehlerauswertung.
  - OB 1:  
Aufruf des SFC 217 (SER\_SND) wobei mit Fehlerauswertung die Daten gesendet werden. Hierbei ist das Telegramm gemäß den Modbus-Vorgaben aufzubauen. Aufruf des SFC 218 (SER\_RECV) wobei mit Fehlerauswertung die Daten empfangen werden.
3. ► Projektieren Sie die Slave-Seite! Das SPS-Anwenderprogramm auf der Slave-Seite sollte folgenden Aufbau haben:
  - OB 100:  
Aufruf SFC 216 (Konfiguration als Modbus RTU-Slave) mit Timeout-Angabe und Modbus-Adresse im DB und Fehlerauswertung
  - OB 1:  
Aufruf des SFC 217 (SER\_SND) für den Datentransport von der Slave-CPU in den Ausgangs-Puffer. Aufruf des SFC 218 (SER\_RECV) für den Datentransport vom Eingangspuffer in die CPU. Für beide Richtungen ist eine entsprechende Fehlerauswertung vorzusehen.

Struktur für die jeweiligen SPS-Programme für Master- und Slave-Seite:





## 8 WinPLC7

### 8.1 Systemvorstellung

#### Allgemein

WinPLC7 ist eine Programmier- und Simulationssoftware von VIPA für alle mit Siemens STEP®7 programmierbaren Steuerungen. Hiermit können Sie Anwenderprogramme in FUP, KOP und AWL erstellen. Neben einer komfortablen Programmierumgebung hat WinPLC7 einen Simulator integriert, der ohne Einsatz zusätzlicher Hardware die Simulation Ihres Anwenderprogramms auf dem PC ermöglicht. Diese "Soft-SPS" wird wie eine reale SPS bedient und bietet gleiches Fehlerverhalten und Diagnosemöglichkeit über Diagnosebuffer, USTACK und BSTACK.



*Ausführliche Informationen und Programmier-Beispiele finden Sie in der Online-Hilfe bzw. in der Online-Dokumentation von WinPLC7.*

#### Alternativen

Sie haben auch die Möglichkeit, anstelle von WinPLC7 von VIPA, entsprechende Konfigurationstools von Siemens zu verwenden. Die Vorgehensweisen hierzu finden Sie in diesem Handbuch.

#### Systemvoraussetzungen

- Windows XP (SP3)
- Windows Vista
- Windows 7 (32 und 64 Bit)
- Windows 8 (32 und 64 Bit)

#### Bezugsquellen

Eine *Demoversion* können Sie von VIPA beziehen. Mit der *Demoversion* können Sie ohne Freischaltung die CPUs 11x aus dem System 100V von VIPA projektieren. Zur Projektierung der SPEED7 CPUs ist eine Lizenz für die "Profi"-Version erforderlich. Diese können Sie von VIPA beziehen und online aktivieren.

Für WinPLC7 gibt es folgende Bezugsquellen:

- Online
  - Unter [www.vipa.com](http://www.vipa.com) im Service-Bereich unter Downloads finden Sie einen Link auf die aktuellste Demo-Version und auf Updates von WinPLC7.
- CD
  - SW211C1DD: WinPLC7 Einzellizenz, CD, mit deutscher Beschreibung
  - SW211C1ED: WinPLC7 Einzellizenz, CD, mit englischer Beschreibung

### 8.2 Installation

#### Voraussetzung

Die Projektierung einer SPEED7-CPU von VIPA unter WinPLC7 ist ausschließlich mit einer aktivierten "Profi"-Version von WinPLC7 möglich.

## Installation WinPLC7 Demo

Die Installation und die Registrierung von WinPLC7 erfolgt nach folgender Vorgehensweise:

1. ▶ Zur Installation von WinPLC7 starten Sie das Setup-Programm von der entsprechenden CD bzw. führen Sie die online bezogene exe-Datei aus.
2. ▶ Wählen Sie die gewünschte Sprachvariante aus.
3. ▶ Stimmen Sie dem Softwarelizenzvertrag zu.
4. ▶ Geben Sie ein Installationsverzeichnis und eine Gruppenzuordnung an und starten Sie den Installationsvorgang.

## Aktivierung der "Profi"-Version

1. ▶ Starten Sie WinPLC7.  
⇒ Es erscheint der Dialog "Demo"
2. ▶ Klicken Sie auf [Vollversion aktivieren].  
⇒ Es erscheint folgender Aktivierungsdialog:

3. ▶ Füllen Sie folgende Felder aus:
  - Email-Adr.
  - Ihr Name
  - Seriennummer  
Ihre Seriennummer finden Sie auf einem Aufkleber auf der CD-Hülle von WinPLC7.
4. ▶ Sofern Ihr PC mit dem Internet verbunden ist, können Sie online über [Aktivierungscode über Internet abfragen] den Aktivierungs-Schlüssel anfordern. Ansonsten klicken Sie auf die Schaltfläche [Der Rechner hat keinen Internetzugang] und folgen Sie den Anweisungen.
  - ⇒ Bei erfolgreicher Registrierung wird der Aktivierungs-Schlüssel im Dialogfenster eingeblendet bzw. Sie erhalten diesen per E-Mail.
5. ▶ Geben Sie diesen unter "Aktivierungs-Schlüssel" ein und klicken Sie auf [OK].
  - ⇒ WinPLC7 ist jetzt als "Profi"-Version aktiviert.

## WinPCAP für Teilnehmersuche über Ethernet installieren

Für die Teilnehmersuche über Ethernet (Erreichbare Teilnehmer) ist der WinPCAP-Treiber zu installieren. Sie finden diesen auf Ihrem PC in Ihrem Installationsverzeichnis unter WinSPS-S7-V5/WinPcap\_... .exe. Führen Sie diese Datei aus und folgen Sie den Anweisungen.

## 8.3 Beispiel zur Projektierung

### 8.3.1 Aufgabenstellung

Im Beispiel wird ein FC 1 programmiert, welcher vom OB 1 zyklisch aufgerufen wird. Durch Vorgabe von 2 Vergleichswerten (value1 und value2) an den FC können Sie abhängig vom Vergleichsergebnis eine Ausgabe zur SPS aktivieren.

Hierbei soll gelten:

- wenn value1 = value2 aktiviere Ausgang A 124.0
- wenn value1 > value2 aktiviere Ausgang A 124.1
- wenn value1 < value2 aktiviere Ausgang A 124.2

### Voraussetzung

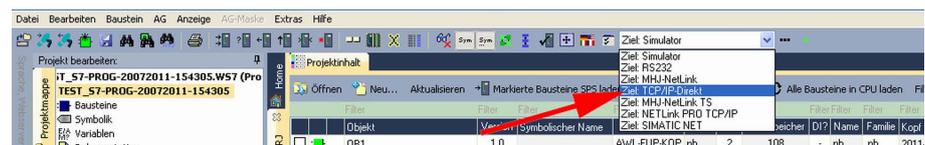
- Sie besitzen Administratorenrechte für Ihren PC.
- WinPLC7 ist installiert und als "Profi"-Version aktiviert.
- Eine SPEED7-CPU und ein digitales Ausgabe-Modul sind aufgebaut und verdrahtet.
- Der Ethernet-PG/OP-Kanal der CPU ist mit Ihrem Ethernet-Netzwerk verbunden. Mit einem Ethernet-Kabel können Sie Ihre CPU entweder direkt oder über einen Switch/Hub an Ihren PC anschließen.
- WinPcap für die Teilnehmersuche über Ethernet ist installiert.
- Die Spannungsversorgung von CPU und E/A-Peripherie ist eingeschaltet und die CPU befindet sich im STOP-Zustand.

### 8.3.2 Projektierung

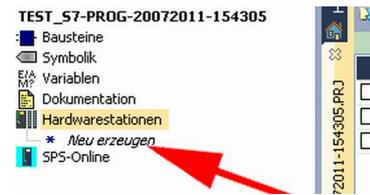
1. ▶ Starten Sie WinPLC7 ("Profi"-Version)
2. ▶ Legen Sie mit [Neue Projektmappe anlegen] ein neues Projekt an und öffnen Sie dies.

### Hardware-Konfiguration

1. ▶ Für den Aufruf des Hardware-Konfigurators ist es erforderlich WinPLC7 vom Simulations-Modus in den Offline-Modus zu schalten. Stellen Sie hierzu zur Kommunikation über Ethernet "Ziel: TCP/IP Direkt" ein.



2. ▶ Doppelklicken Sie auf "Hardwarestation" und hier auf "Neu erzeugen".



3. ➤ Geben Sie einen Stationsnamen an. Bitte beachten Sie, dass der Name keine Leerzeichen enthalten darf.
4. ➤ Nach der Ladeanimation wählen Sie im Register SPS-System selektieren das System "VIPA SPEED7" und klicken Sie auf [Erzeugen]. Eine neue Station wird angelegt.
5. ➤ Sichern Sie die leere Station mit [Strg]+[S].
6. ➤ Gehen Sie im Hardware-Katalog auf "CPU SPEED7" und fügen Sie die entsprechende VIPA-CPU durch Doppelklick in der Station ein.
7. ➤ Platzieren Sie für die Ausgabe ein digitales Ausgabe-Modul, geben Sie diesem die Anfangsadresse 124 und sichern Sie die Hardware-Konfiguration.

Online-Zugriff über Ethernet-PG/OP-Kanal einrichten:

1. ➤ Öffnen Sie die CPU-Eigenschaften, indem Sie im Hardware-Konfigurator auf die CPU auf Steckplatz 2 doppelklicken.
2. ➤ Klicken Sie auf die Schaltfläche [Ethernet CP-Einstellungen (PG/OP-Kanal)].  
⇒ Es öffnet sich der Dialog "Eigenschaften CP343"
3. ➤ Wählen Sie das Register "Allgemeine Parameter" an.
4. ➤ Klicken Sie auf [Eigenschaften Ethernet].
5. ➤ Wählen Sie das Subnetz "PG\_OP\_Ethernet".
6. ➤ Geben Sie eine gültige IP-Adresse und Subnetz-Maske an. Sie erhalten diese von Ihrem Systemadministrator.
7. ➤ Schließen Sie alle Dialogfenster mit [OK].
8. ➤ Stellen Sie, wenn nicht schon geschehen, "Ziel: Extern TCP/IP direkt" ein.
9. ➤ Öffnen Sie mit "Online → Konfiguration übertragen" den gleichnamigen Dialog.
10. ➤ Klicken Sie auf [Erreichbare Teilnehmer]. Bitte beachten Sie, dass hierzu WinPCap installiert sein muss!
11. ➤ Wählen Sie Ihre Netzwerkkarte aus und klicken Sie auf die Schaltfläche [Teilnehmer ermitteln].  
⇒ Nach einer Wartezeit werden alle erreichbaren Teilnehmer aufgelistet. Hier finden Sie auch Ihre CPU, die mit IP 0.0.0.0 gelistet ist. Zur Kontrolle wird hier auch die MAC-Adresse angezeigt, die sich als Aufkleber unterhalb der Frontabdeckung Ihrer CPU befindet.
12. ➤ Zur Vergabe einer temporären IP-Adresse wählen Sie Ihre CPU an und klicken Sie auf [IP Parameter temporär setzen]. Geben Sie hier die gleichen IP-Parameter an, die Sie in den CPU-Eigenschaften parametriert haben und klicken Sie auf [Parameter schreiben].

- 13.** ➤ Bestätigen Sie die Meldung, dass die CPU urgelöscht wird.
  - ⇒ Die IP-Parameter werden an die CPU übertragen und die Liste der erreichbaren Teilnehmer wird aktualisiert.
- 14.** ➤ Wählen Sie Ihre CPU aus und klicken Sie auf [Übernehmen].
  - ⇒ Sie befinden sich nun wieder im Dialog "Konfiguration übertragen".

#### Hardware-Konfiguration übertragen

- Wählen Sie Ihre Netzwerkkarte aus und klicken Sie auf [Konfiguration übertragen].
  - ⇒ Nach einer kurzen Zeit erhalten Sie die Meldung, dass die Konfiguration übertragen wurde.

Hiermit ist die Hardware-Konfiguration abgeschlossen und die CPU immer über die von Ihnen vergebene IP-Adresse auch über WinPLC7 zu erreichen.

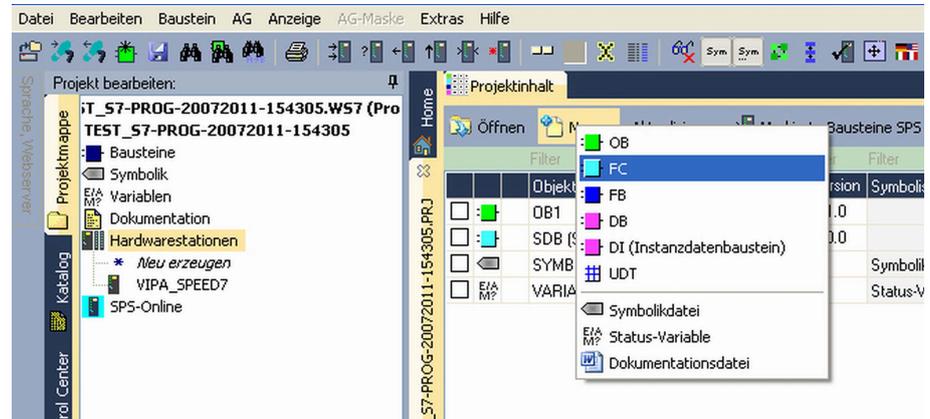


*In der Regel erfolgt die Online-Übertragung Ihrer Hardware-Konfiguration aus dem Hardware-Konfigurator. Sie können aber auch mit "Datei ➔ Aktive Station im WinPLC-Unterprojekt speichern" Ihre Hardware-Konfiguration als System-Datei in WinPLC7 übertragen und über WinPLC7 an Ihre CPU transferieren.*

**Programmierung von FC 1**

Die SPS-Programmierung findet in WinPLC7 statt. Schließen Sie den Hardware-Konfigurator und kehren Sie zu Ihrem Projekt in WinPLC7 zurück. Das SPS-Programm ist im Baustein FC 1 zu erstellen.

1. Wählen Sie in "Projekthinhalt" "Neu → FC".



2. Geben Sie als Baustein "FC1" an und bestätigen Sie Ihre Eingabe mit [OK].  
⇒ Der Editor für den FC 1 wird aufgerufen.

**Parameter anlegen**

Der obere Teil des Editors enthält die Parametertabelle. In diesem Beispiel sollen die 2 Integer-Werte *value1* und *value2* miteinander verglichen werden. Da beide Werte innerhalb der Funktion nur gelesen werden, sind diese als "in" zu deklarieren.

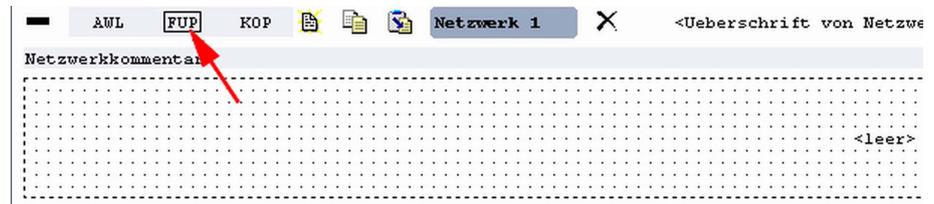
1. Gehen Sie auf der "Parametertabelle" in die Zeile "in →" und tragen Sie im Feld "Name" "value1" ein. Drücken Sie die [Eingabe]-Taste.  
⇒ Der Cursor springt zu der Spalte für den Datentyp.
2. Sie können jetzt entweder den Datentyp direkt eingeben oder durch Drücken der [Eingabe]-Taste aus einer Liste verfügbarer Datentypen auswählen. Geben Sie als Datentyp INT an und betätigen Sie die [Eingabe]-Taste.  
⇒ Der Cursor springt zu der Spalte für den "Kommentar".
3. Geben Sie hier "1. Vergleichswert" an und drücken Sie die [Eingabe]-Taste.  
⇒ Eine neue "in →"-Zeile wird erzeugt und der Cursor in "Name" gesetzt.
4. Verfahren Sie für *value2* auf die gleiche Weise wie unter *value1* beschrieben.
5. Speichern Sie den Baustein. Einen eventuellen Hinweis, dass die Schnittstelle des Bausteins geändert wurde, können Sie mit [Ja] quittieren.  
⇒ Die Parametertabelle enthält nun folgende Einträge:

Adresse	Deklaration	Name	Typ	Anfangswert	Kommentar
0.0	in →	value1	INT		1. Vergleichswert
2.0	in →	value2	INT		2. Vergleichswert
	out <				
	in_out <->				

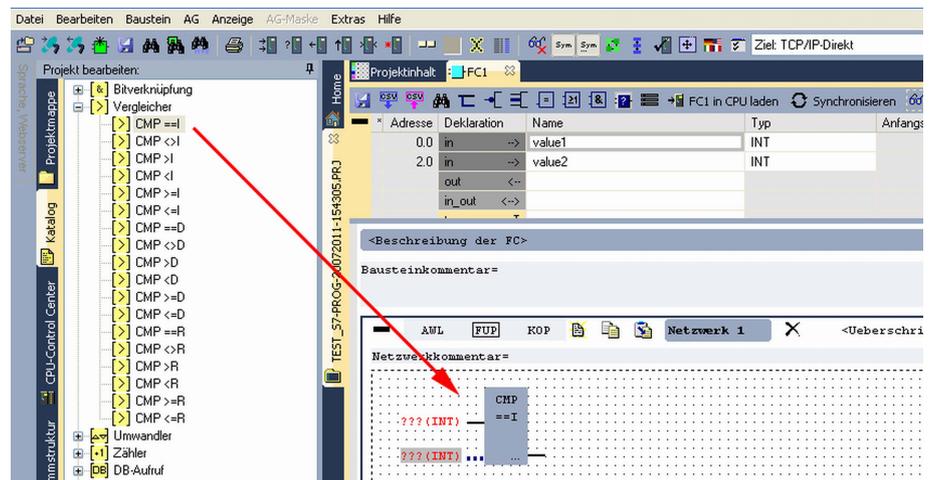
## Programm eingeben

Wie in der Aufgabenstellung gefordert soll je nach Vergleich von *value1* und *value2* der entsprechende Ausgang aktiviert werden. Für jede Vergleichsoperation ist ein Netzwerk anzulegen.

1. Das Programm soll als FUP (Funktionsplan) erzeugt werden. Wählen Sie hierzu durch Klicken auf "FUP" die FUP-Ansicht.



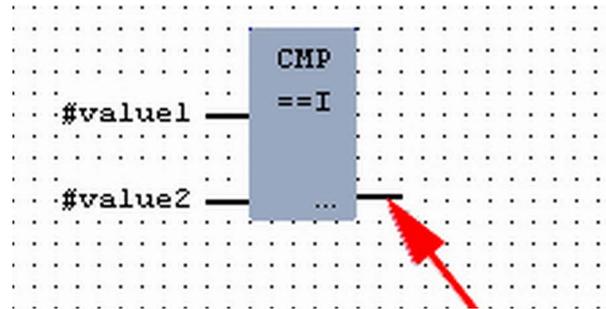
2. Klicken Sie in das mit "<leer>" bezeichnete Eingabefeld. Die zur Verfügung stehenden Operationen können Sie mit Drag&Drop aus dem *Katalog* in Ihr Projekt ziehen oder durch Doppelklick im *Katalog* in Ihr Projekt übernehmen.
3. Öffnen Sie im *Katalog* die Kategorie "Vergleicher" und fügen Sie die Operation "CMP==I" in Ihr Netzwerk ein.



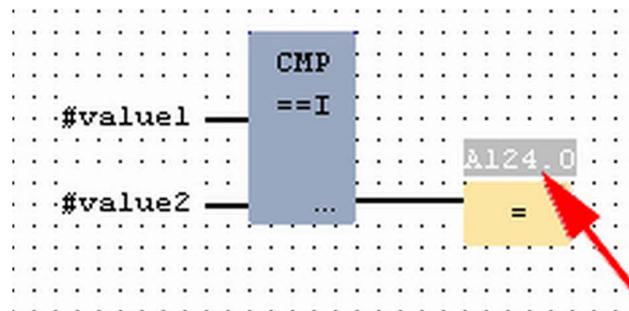
4. Klicken Sie auf den linken oberen Eingang und fügen Sie *value1* ein. Da es sich hierbei um Bausteinparameter handelt, können Sie durch Eingabe von "#" eine Auswahlliste der Bausteinparameter öffnen.
5. Geben Sie "#" ein und betätigen Sie mit der [Eingabe]-Taste
6. Wählen Sie aus der Auswahlliste den entsprechenden Parameter aus und übernehmen Sie mit der [Eingabe]-Taste.
7. Verfahren Sie auf die gleiche Weise mit dem Parameter *value2*.

Die Zuordnung zu dem korrespondierenden Ausgang, hier A 124.0, erfolgt nach folgender Vorgehensweise:

1. Klicken Sie auf den Ausgang auf der rechten Seite des Operators.



2. ▶ Öffnen Sie im *Katalog* die Kategorie "*Bitverknüpfung*" und wählen Sie die Verknüpfung "--[=]". Das Einfügen von "--[=]" ist bei WinPLC7 auf der Funktions-Taste [F7] abgelegt.
3. ▶ Geben Sie durch Klick auf den Operanden den Ausgang A 124.0 an.

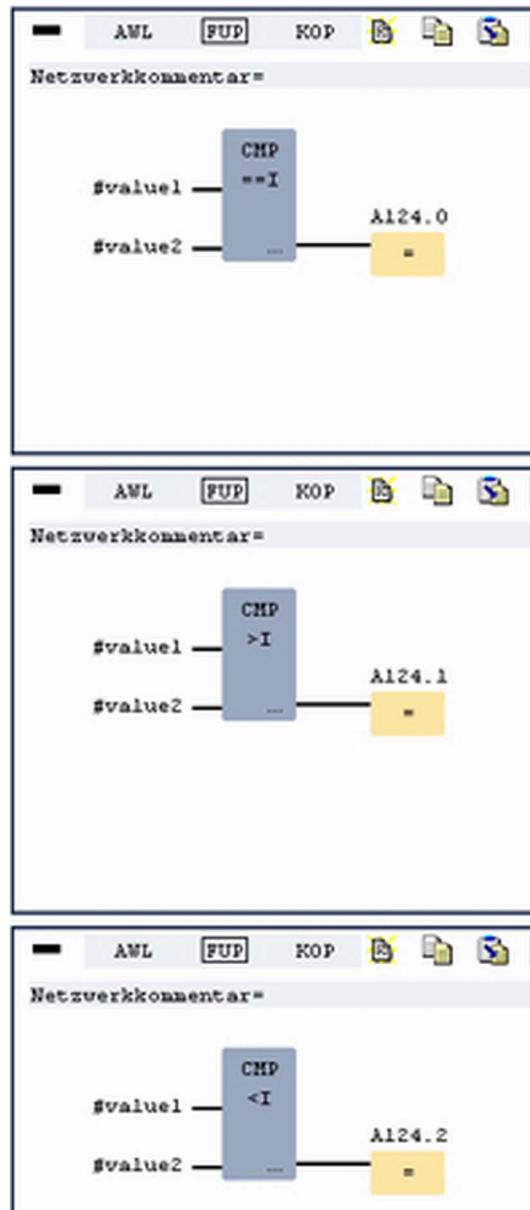


⇒ Hiermit ist Netzwerk1 abgeschlossen.

#### Neues Netzwerk einfügen

Für die weiteren Vergleiche sind die Operationen "CMP>I" auf A 124.1 und "CMP<I" auf A 124.2 erforderlich. Legen Sie für beide Operationen nach folgender Vorgehensweise ein Netzwerk an:

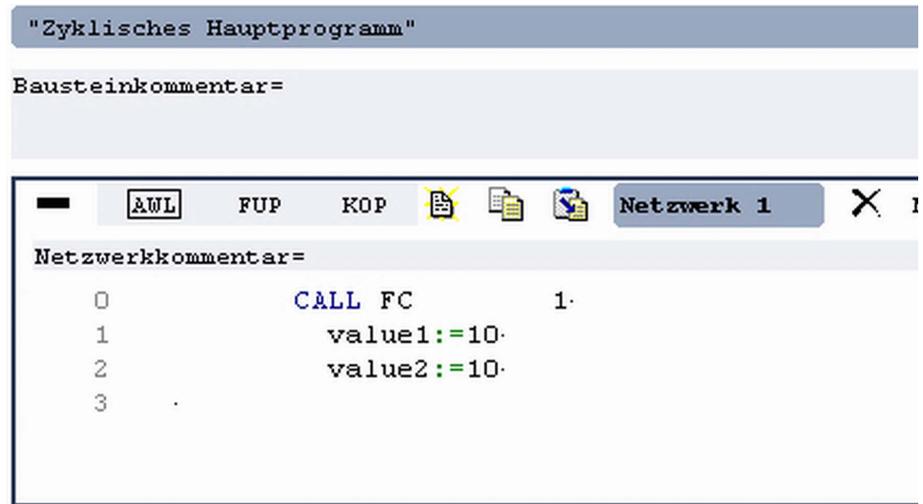
1. ▶ Bewegen Sie Ihre Maus auf das Editor-Fenster an beliebiger Stelle und betätigen Sie die rechte Maustaste.
2. ▶ Wählen Sie "*Kontextmenü* → *Einfügen neues Netzwerk*".
  - ⇒ Es öffnet sich ein Dialogfeld zur Vorgabe von Position und Anzahl der Netzwerke.
3. ▶ Verfahren Sie auf die gleiche Weise wie für "Netzwerk 1" beschrieben.
4. ▶ Speichern Sie den FC 1 mit "*Datei* → *Aktuelles Fenster speichern*" bzw. mit [Strg]+[S].
  - ⇒ Nachdem Sie die noch fehlenden Netzwerke ausprogrammiert haben, hat der FC 1 folgenden Aufbau:



**Baustein OB 1 erzeugen** Der Aufruf des FC 1 hat aus dem Zyklus-OB OB 1 zu erfolgen.

- 1.** Wechseln Sie in den OB 1, der bei der Projektanlage schon automatisch erzeugt wurde.
- 2.** Gehen Sie in "Projektinhalt" oder in Ihre "Projektmappe" und öffnen Sie den OB 1 durch Doppelklick.
- 3.** Wechseln Sie in die AWL-Ansicht.

4. ▶ Geben Sie "Call FC 1" ein und betätigen Sie die *[Eingabe]*-Taste.
  - ⇒ Die FC-Parameter werden automatisch angezeigt und die folgenden Parameter zugeordnet:



5. ▶ Speichern Sie den OB 1 mit bzw. mit *[Strg]+[S]*

### 8.3.3 SPS-Programm in *Simulator* testen

#### Vorgehensweise

WinPLC7 bietet Ihnen die Möglichkeit Ihr Projekt in einem *Simulator* zu testen.

1. ▶ Stellen Sie hierzu *"Ziel: Simulator"* ein.



2. ▶ Übertragen Sie die Bausteine in den Simulator mit *[Alle Bausteine in CPU laden]*.



3. ▶ Schalten Sie Ihre CPU in RUN, indem Sie unter *"Projekt bearbeiten"* in *"CPU-Control Center"* wechseln und hier auf *"RUN"* klicken.
  - ⇒ Die Anzeige wechselt von STOP nach RUN.
4. ▶ Zur Anzeige des Prozessabbilds gehen Sie auf *"Anzeige → PAA/PAE-Fenster anzeigen"* oder klicken Sie auf .
  - ⇒ Die verschiedenen Bereiche werden eingeblendet.

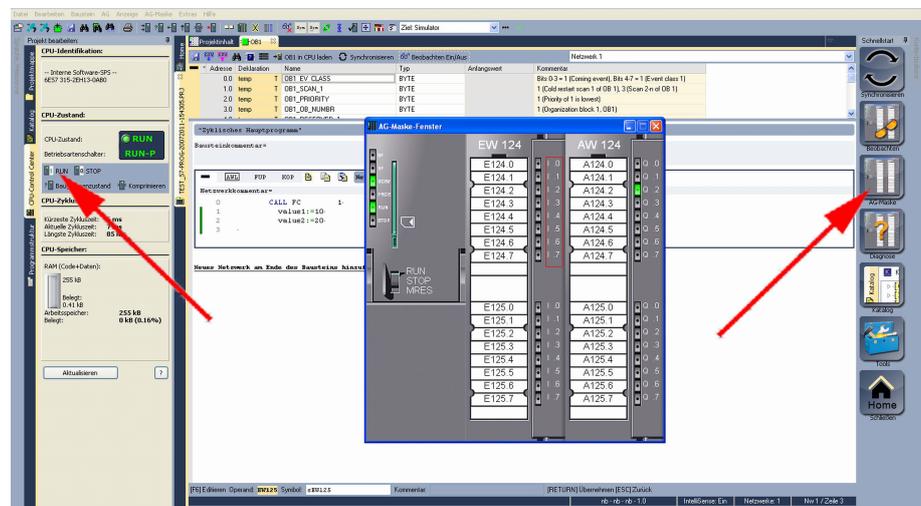
Beispiel zur Projektierung > SPS-Programm in CPU übertragen und ausführen

5. ▶ Doppelklicken Sie auf das Prozessabbild und geben Sie im Register "Zeile2" die Adresse PAB 124 an. Bestätigen Sie Ihre Eingabe mit [OK]. Ein mit roter Farbe hinterlegter Wert entspricht einer logischen "1".
6. ▶ Öffnen Sie den OB 1.
7. ▶ Ändern Sie den Wert einer Variablen, speichern Sie den OB 1 und übertragen Sie den Baustein in den Simulator.
  - ⇒ Gleich darauf ändert sich das Prozessabbild gemäß Ihren Vorgaben. Mit "*Baustein* → *Beobachten EIN/AUS*" können Sie den Status Ihrer Bausteine anzeigen.

## Visualisierung über AG-Maske

Ein weiterer Bestandteil des Simulators ist die *AG-Maske*. Hier wird grafisch eine CPU dargestellt, die mit digitalen und analogen Peripheriemodulen erweitert werden kann. Sobald sich die CPU im Simulator in RUN befindet, können Sie hier mit der Maus Eingänge aktivieren und das Verhalten der Ausgänge anzeigen.

1. ▶ Öffnen Sie die *AG-Maske* über "*Anzeige* → *AG-Maske*".
  - ⇒ Eine CPU wird grafisch dargestellt.
2. ▶ Öffnen Sie durch Doppelklick auf die Ausgabebaugruppe den Eigenschaften-Dialog und stellen Sie die Baugruppenadresse 124 ein.
3. ▶ Schalten Sie mit der Maus den Betriebsartenschalter in RUN.
  - ⇒ Ihr Programm wird im Simulator ausgeführt und dargestellt.



## 8.3.4 SPS-Programm in CPU übertragen und ausführen

### Vorgehensweise

1. ▶ Zur Übertragung in Ihre CPU stellen Sie "Ziel: TCP/IP-Direkt" ein.
2. ▶ Bei Einsatz mehrerer Netzwerkkarten können sie über "*Extras* → *Netzwerkkarte auswählen*" Ihre Netzwerkkarte bestimmen.
3. ▶ Zur Vorgabe der Ethernet-Daten klicken Sie auf [...] und klicken Sie auf [Erreichbare Teilnehmer].



4. ▶ Klicken Sie auf [Teilnehmer ermitteln].
  - ⇒ Nach einer gewissen Wartezeit werden alle verfügbaren Teilnehmer aufgelistet.
5. ▶ Wählen Sie Ihre CPU aus, die Sie über die Hardware-Konfiguration mit TCP/IP-Adress-Parametern schon versorgt haben und klicken Sie auf [übernehmen].
6. ▶ Schließen Sie den Dialog "Ethernet-Daten" mit [OK].
7. ▶ Übertragen Sie Ihr Projekt in Ihre CPU mit "AG"
  - ➔ *Alle Bausteine senden*.
8. ▶ Schalten Sie Ihre CPU in RUN.
9. ▶ Öffnen Sie den OB 1 durch Doppelklick
10. ▶ Ändern Sie den Wert einer Variablen, speichern Sie den OB 1 und übertragen Sie den Baustein in die CPU.
  - ⇒ Gleich darauf ändert sich das Ausgabe-Verhalten gemäß Ihren Vorgaben. Mit "*Baustein ➔ Beobachten EIN/AUS*" können Sie den Status Ihrer Bausteine anzeigen.

## 9 Projektierung im TIA Portal

### 9.1 TIA Portal - Arbeitsumgebung

#### 9.1.1 Allgemein

##### Allgemein

In diesem Teil wird die Projektierung der VIPA CPU im Siemens TIA Portal gezeigt. Hier soll lediglich der grundsätzliche Einsatz des Siemens TIA Portals in Verbindung mit der VIPA CPU gezeigt werden. TIA steht für **T**otally **I**ntegrated **A**utomation von Siemens. Hier können Sie Ihre VIPA-Steuerungen programmieren und vernetzen. Für die Diagnose stehen Ihnen Online-Werkzeuge zur Verfügung.

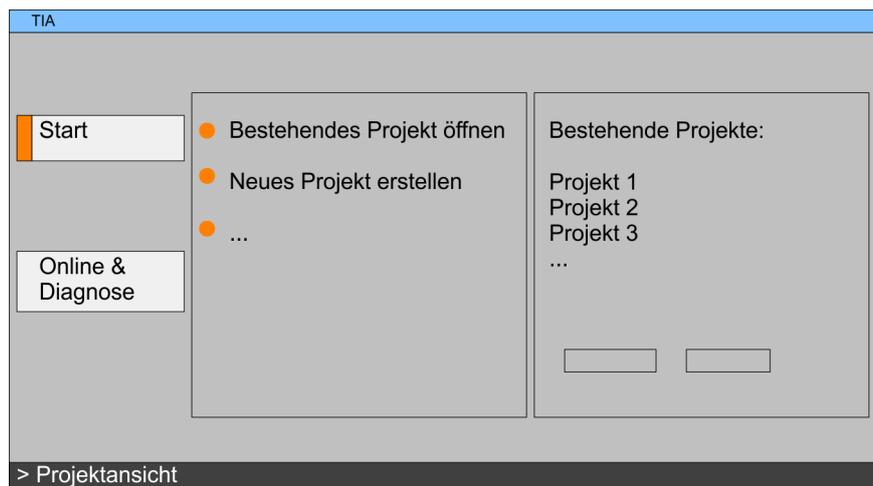


Nähere Informationen zum TIA Portal finden Sie in der zugehörigen Online-Hilfe bzw. Dokumentation.

##### TIA Portal starten

Zum Starten des Siemens TIA Portals wählen Sie unter Windows den Befehl "Start → Programme → Siemens Automation → TIA ..."

Daraufhin wird das TIA Portal mit den zuletzt verwendeten Einstellungen geöffnet.



##### TIA Portal beenden

Mit dem Menüpunkt "Projekt → Beenden" können Sie aus der "Projektansicht" das TIA Portal beenden. Hierbei haben Sie die Möglichkeit durchgeführte Änderungen an Ihrem Projekt zu speichern.

#### 9.1.2 Arbeitsumgebung des TIA Portals

Grundsätzlich besitzt das TIA Portal folgende 2 Ansichten. Über die Schaltfläche links unten können Sie zwischen diesen Ansichten wechseln:

##### Portalansicht

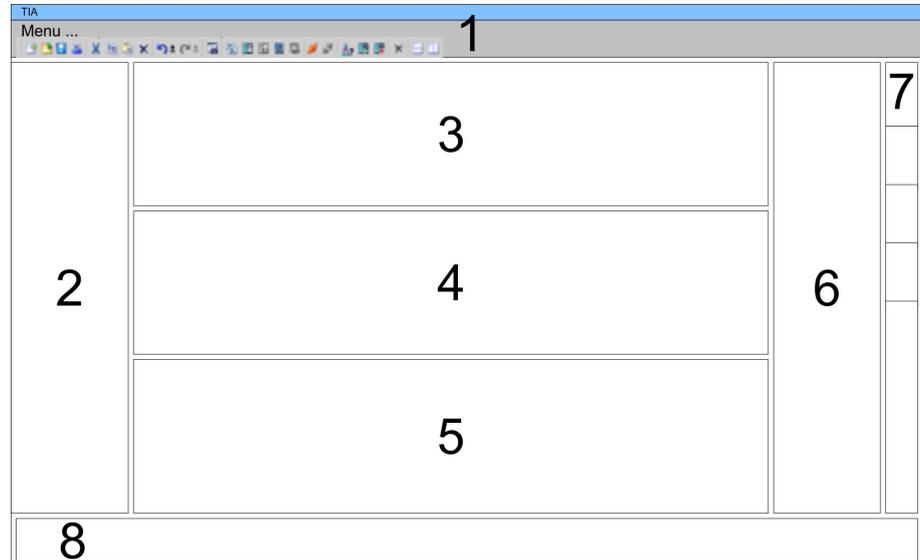
Die "Portalansicht" bietet eine "aufgabenorientierte" Sicht der Werkzeuge zur Bearbeitung Ihres Projekts. Hier haben Sie direkten Zugriff auf die Werkzeuge für eine Aufgabe. Falls erforderlich, wird für die ausgewählte Aufgabe automatisch zur Projektansicht gewechselt.

**Projektansicht**

Die "*Projektansicht*" ist eine "strukturierte" Sicht auf alle Bestandteile Ihres Projekts.

**Bereiche der Projektansicht**

Die Projektansicht gliedert sich in folgende Bereiche:



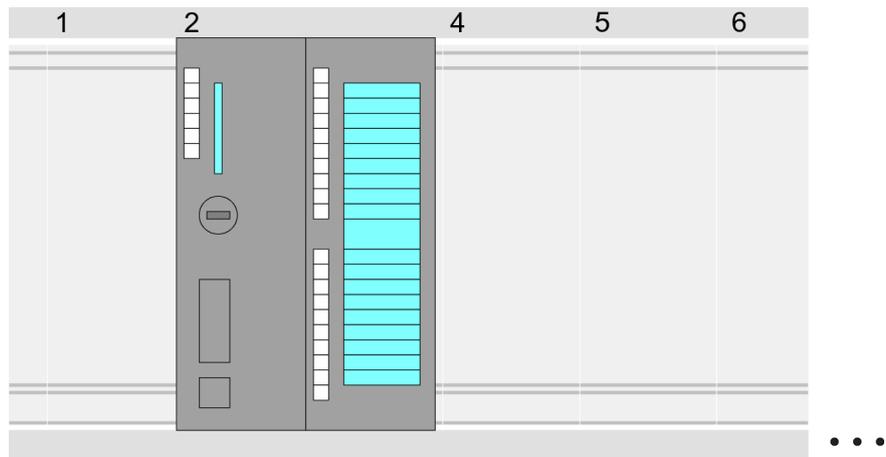
- 1 Menüleiste mit Funktionsleisten
- 2 Projektnavigation mit Detailansicht
- 3 Projektbereich
- 4 Geräteübersicht des Projekts bzw. Bereich für die Baustein-Programmierung
- 5 Eigenschaften-Dialog eines Geräts (Parameter) bzw. Informationsbereich
- 6 Hardware-Katalog und Tools
- 7 "Task-Cards" zur Auswahl von Hardware-Katalog, Anweisungen und Bibliotheken
- 8 Wechsel zwischen Portal- und Projektansicht

**9.2 TIA Portal - Hardware-Konfiguration - CPU****Projektierung Siemens CPU**

Mit dem Siemens TIA Portal ist die VIPA CPU 312-5BE13 als CPU 312C (6ES7 312-5BE03-0AB0 V2.6) von Siemens zu projektieren.

1. ▶ Starten Sie das Siemens TIA Portal.
2. ▶ Erstellen sie in der *Portalansicht* mit "*Neues Projekt erstellen*" ein neues Projekt.
3. ▶ Wechseln Sie in die *Projektansicht*.
4. ▶ Klicken Sie in der *Projektnavigation* auf "*Neues Gerät hinzufügen*".
5. ▶ Wählen Sie im Eingabedialog folgende CPU aus:  
SIMATIC S7-300 > CPU 312C > 6ES7 312-5BE03-0AB0 V2.6  
⇒ Die CPU wird mit einer Profilschiene eingefügt.

**Projektbereich:**



**Geräteübersicht:**

Baugruppe	...	Steckplatz	...	Typ	...
PLC...		2		CPU 312C	
MPI-Schnittstelle...		2 0		MPI-Schnittstelle	
DI10/DO6...		2 X2		DI10/DO6	
Zählen...		2 4		Zählen	
...					

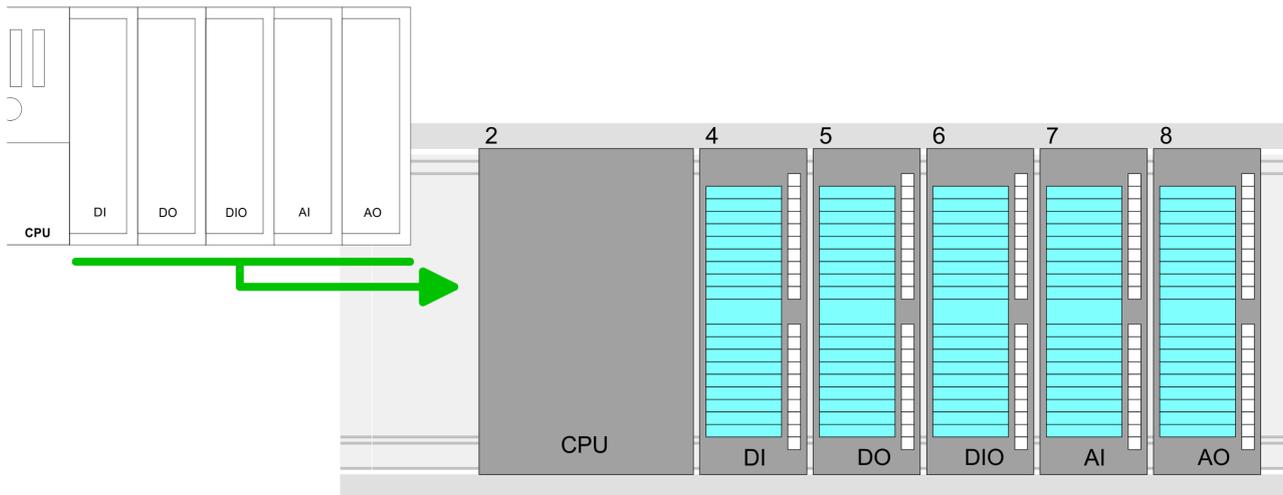
**Einstellung Standard CPU-Parameter**

Da die CPU 312-5BE13 von VIPA als Siemens-CPU 312C projiziert wird, erfolgt auch die Parametrierung über die Siemens-CPU. Zur Parametrierung klicken Sie im *Projektbereich* bzw. in der *Geräteübersicht* auf den CPU-Teil. Daraufhin werden die Parameter des CPU-Teils im *Eigenschaften*-Dialog aufgeführt. Hier können Sie Ihre Parametereinstellungen vornehmen. ↪ *Kapitel 5.8 "Einstellung der CPU-Parameter" auf Seite 53*

**9.3 TIA Portal - Hardware-Konfiguration - I/O-Module**

**Hardware-Konfiguration der Module**

Binden Sie nach der Hardware-Konfiguration der CPU Ihre System 300 Module auf dem Bus in der gesteckten Reihenfolge ein. Gehen Sie hierzu in den Hardware-Katalog und ziehen Sie das entsprechende Modul auf die entsprechende Position der Profilschiene im *Projektbereich* oder auf die entsprechende Position in der *Geräteübersicht*.



**Geräteübersicht**

Baugruppe	...	Steckplatz	...	Typ	...
PLC...		2		CPU ...	
...		...		...	
		3			
DI...		4		DI...	
DO...		5		DO...	
DIO...		6		DIO...	
AI...		7		AI...	
AO...		8		AO...	

**Parametrierung**

Zur Parametrierung klicken Sie im *Projektbereich* bzw. in der *Geräteübersicht* auf das zu parametrierende Modul. Daraufhin werden die Parameter des Moduls im Eigenschaftens-Dialog aufgeführt. Hier können Sie Ihre Parametereinstellungen vornehmen.

**9.4 TIA Portal - Hardware-Konfiguration - Ethernet-PG/OP-Kanal**

**Übersicht**

Die CPU hat einen Ethernet-PG/OP-Kanal integriert. Über diesen Kanal können Sie Ihre CPU programmieren und fernwarten.

- Mit dem Ethernet-PG/OP-Kanal haben Sie auch Zugriff auf die interne Web-Seite, auf der Sie Informationen zu Firmwarestand, angebundene Peripherie, aktuelle Zyklus-Zeiten usw. finden.
- Bei Erstinbetriebnahme bzw. nach dem Rücksetzen auf Werkseinstellungen besitzt der Ethernet-PG/OP-Kanal keine IP-Adresse.
- Damit Sie online über den Ethernet-PG/OP-Kanal auf die CPU zugreifen können, müssen Sie diesem gültige IP-Adress-Parameter zuordnen. Diesen Vorgang nennt man "Initialisierung" oder "Urtaufe".
- Dies kann mit dem Siemens TIA Portal erfolgen.

**Montage und Inbetriebnahme**

1. ► Bauen Sie Ihr System 300S mit Ihrer CPU auf.
2. ► Verdrahten Sie das System, indem Sie die Leitungen für Spannungsversorgung und Signale anschließen.
3. ► Verbinden Sie die Ethernet-Buchse des Ethernet-PG/OP-Kanals mit Ethernet.
4. ► Schalten Sie die Spannungsversorgung ein.  
⇒ Nach kurzer Hochlaufzeit ist der CP bereit für die Kommunikation. Er besitzt ggf. noch keine IP-Adressdaten und erfordert eine Urtaufe.

**"Urtaufe" über Onlinefunktionen**

Die Urtaufe über die Onlinefunktion erfolgt nach folgender Vorgehensweise:

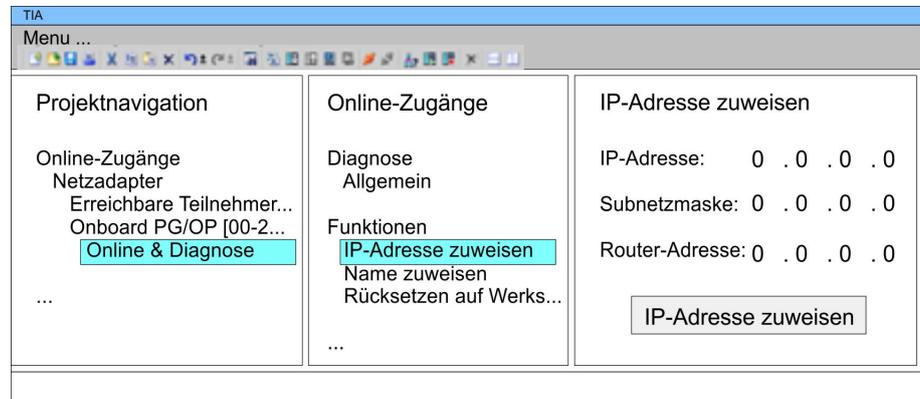
- Ermitteln Sie die aktuelle Ethernet (MAC) Adresse Ihres Ethernet PG/OP-Kanals. Sie finden diese immer als 1. Adresse unter der Frontklappe der CPU auf einem Aufkleber auf der linken Seite.

**IP-Adress-Parameter zuweisen**

Gültige IP-Adress-Parameter erhalten Sie von Ihrem Systemadministrator. Die Zuweisung der IP-Adress-Daten erfolgt online im Siemens TIA Portal nach folgender Vorgehensweise:

1. ► Starten Sie das Siemens TIA Portal.
2. ► Wechseln Sie in die "*Projektansicht*".
3. ► Klicken Sie in der "*Projektnavigation*" auf "*Online-Zugänge*" und wählen Sie hier durch Doppelklick Ihre Netzwerkkarte aus, welche mit dem Ethernet-PG/OP-Kanal verbunden ist.
4. ► Benutzen Sie "*Erreichbare Teilnehmer...*", um die über MAC-Adresse erreichbaren Geräte zu ermitteln. Die MAC-Adresse finden Sie auf dem 1. Aufkleber unter der Frontklappe der CPU.
5. ► Wählen Sie aus der Liste die Baugruppe mit der Ihnen bekannten MAC-Adresse (Onboard PG/OP [MAC-Adresse]) und öffnen Sie mit "Online & Diagnose" den Diagnose-Dialog im *Projektbereich*.
6. ► Navigieren Sie zu *Funktionen > IP-Adresse zuweisen*. Stellen Sie nun die IP-Konfiguration ein, indem Sie IP-Adresse, Subnetz-Maske und den Netzübergang eintragen.

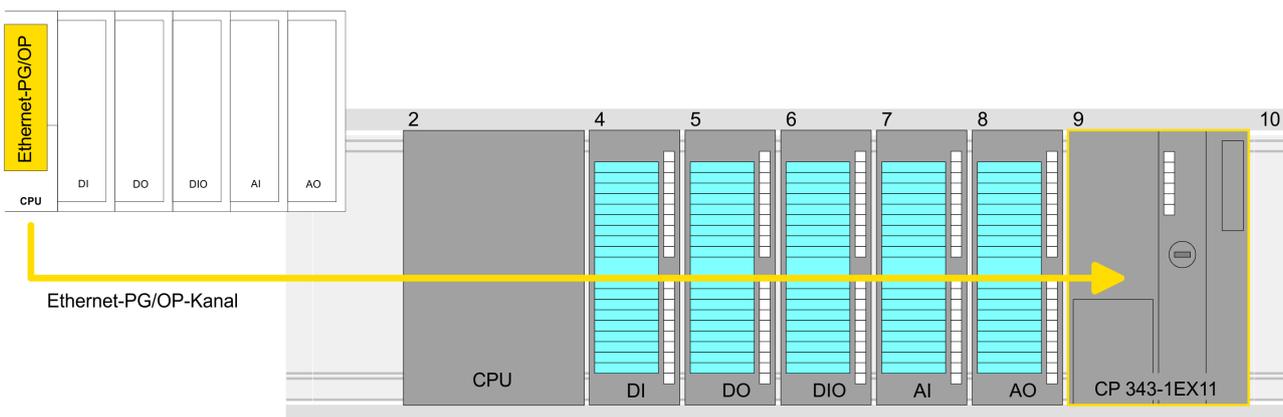
7. ➤ Bestätigen Sie mit [IP-Adresse zuweisen] Ihre Eingabe.
  - ⇒ Direkt nach der Zuweisung ist der Ethernet-PG/OP-Kanal über die angegebenen IP-Adress-Daten online erreichbar. Der Wert bleibt bestehen, solange dieser nicht neu zugewiesen, mit einer Hardware-Projektierung überschrieben oder Rücksetzen auf Werkseinstellung ausgeführt wird.



Systembedingt kann es zu einer Meldung kommen, dass die IP-Adresse nicht vergeben werden konnte. Diese Meldung können Sie ignorieren.

**IP-Adress-Parameter in Projekt übernehmen**

1. ➤ Öffnen Sie Ihr Projekt.
2. ➤ Projektieren Sie, wenn nicht schon geschehen, in der "Gerätekongfiguration" eine Siemens CPU 312C (6ES7 312-5BE03-0AB0 V2.6).
3. ➤ Projektieren Sie Ihre System 300 Module
4. ➤ Projektieren Sie für den Ethernet-PG/OP-Kanal immer als letztes Modul nach den reell gesteckten Modulen einen Siemens CP 343-1 (6GK7 343-1EX11 0XE0).
5. ➤ Öffnen Sie durch Klick auf den CP 343-1EX11 den "Eigenschaften"-Dialog und geben Sie für den CP in den "Eigenschaften" unter "Ethernet-Adresse" die zuvor zugewiesenen IP-Adress-Daten an.
6. ➤ Übertragen Sie Ihr Projekt.



**Geräteübersicht**

Baugruppe	...	Steckplatz	...	Typ	...
PLC...		2		CPU ...	
...		...		...	
		3			
DI...		4		DI...	
DO...		5		DO...	
DIO...		6		DIO...	
AI...		7		AI...	
AO...		8		AO...	
 CP 343-1		9		CP 343-1	

**9.5 TIA Portal - VIPA-Bibliothek einbinden****Übersicht**

- Die VIPA-spezifischen Bausteine finden Sie im "Service"-Bereich auf [www.vipa.com](http://www.vipa.com) unter *Downloads > VIPA LIB* als Bibliothek zum Download.
- Die Bibliothek liegt als gepackte zip-Datei *Fx000020\_V...* vor.
- Sobald Sie VIPA-spezifische Bausteine verwenden möchten, sind diese in Ihr Projekt zu importieren.  
Folgende Schritte sind hierzu erforderlich:
  - Die Datei *FX000020\_V...* .zip entpacken
  - Bibliothek öffnen und Bausteine in Projekt übertragen

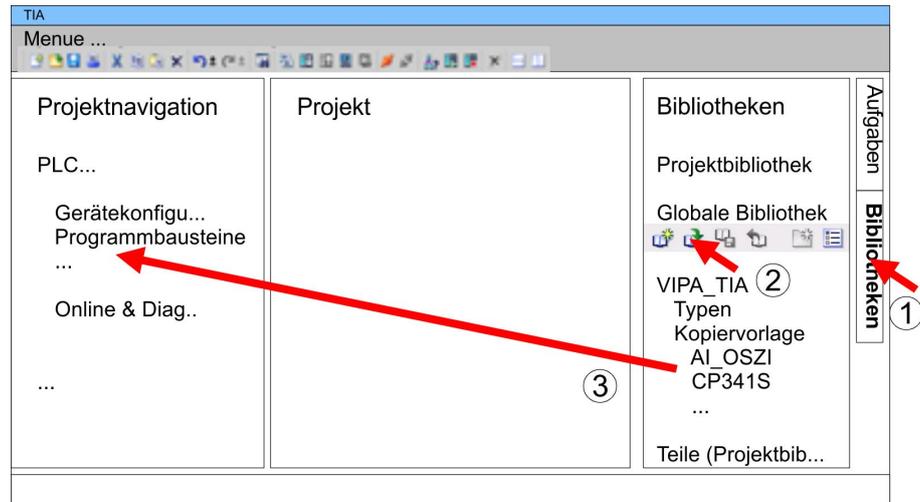
**FX000020\_V... .zip entpacken**

Starten Sie mit einem Doppelklick auf die Datei *FX000020\_V...* .zip ihr Unzip-Programm entpacken Sie Dateien und Ordner in ein Arbeits-Verzeichnis 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 VIPA\_TIA.al11.



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 VIPA-spezifischen Bausteine.

## 9.6 TIA Portal - Projekt transferieren

### Übersicht

Sie haben folgende Möglichkeiten für den Projekt-Transfer in die CPU:

- Transfer über MPI
- Transfer über Ethernet
- Transfer über Speicherkarte

### Transfer über MPI

Aktuell werden die VIPA Programmierkabel für den Transfer über MPI nicht unterstützt. Dies ist ausschließlich über Programmierkabel von Siemens möglich.

1. Stellen Sie mit dem entsprechenden Programmierkabel eine Verbindung über MPI mit ihrer CPU her. Informationen hierzu finden Sie in der zugehörigen Dokumentation zu Ihrem Programmierkabel.
2. Schalten Sie die Spannungsversorgung ihrer CPU ein und starten Sie das Siemens TIA Portal mit Ihrem Projekt.
3. Markieren Sie in der *Projektnavigation* Ihre CPU und wählen Sie für den Transfer der Hardware-Konfiguration "*Kontextmenü* → *Laden in Gerät* → *Hardwarekonfiguration*".
4. Ihr SPS-Programm übertragen Sie mit "*Kontextmenü* → *Laden in Gerät* → *Software*". Systembedingt müssen Sie Hardware-Konfiguration und SPS-Programm getrennt übertragen.

### Transfer über Ethernet

Die CPU besitzt für den Transfer über Ethernet folgende Schnittstelle:

- X5: Ethernet-PG/OP-Kanal

**Initialisierung**

Damit Sie auf die entsprechende Ethernet-Schnittstelle online zugreifen können, müssen Sie dieser durch die "Initialisierung" bzw. "Urtaufe" IP-Adress-Parameter zuweisen.

Bitte beachten Sie, dass Sie die IP-Adress-Daten in Ihr Projekt für den CP 343-1 übernehmen.

**Transfer**

1. ► Für den Transfer verbinden Sie, wenn nicht schon geschehen, die entsprechende Ethernet-Buchse mit Ihrem Ethernet.
2. ► Öffnen Sie Ihr Projekt im Siemens TIA Portal.
3. ► Klicken Sie in der *Projektnavigation* auf *Online-Zugänge* und wählen Sie hier durch Doppelklick Ihre Netzwerkkarte aus, welche mit der Ethernet- PG/OP-Schnittstelle verbunden ist.
4. ► Wählen Sie in der *Projektnavigation* Ihre CPU aus und klicken Sie auf [Online verbinden].
5. ► Geben Sie den Zugriffsweg vor, indem Sie als Schnittstellentyp "PN/IE" einstellen und als PG/PC-Schnittstelle Ihre Netzwerkkarte und das entsprechende Subnetz auswählen. Daraufhin wird ein Netz-Scan ausgeführt und der entsprechende Verbindungspartner aufgelistet.
6. ► Stellen Sie mit [Verbinden] eine Online-Verbindung her.
7. ► Gehen Sie auf "*Online → Laden in Gerät*".
  - ⇒ Der entsprechende Baustein wird übersetzt und nach einer Abfrage an das Zielgerät übertragen. Sofern keine neue Hardware-Konfiguration in die CPU übertragen wird, wird die hier angegebene Ethernet-Verbindung dauerhaft als Transferkanal im Projekt gespeichert.

**Transfer über Speicherkarte**

Die Speicherkarte dient als externes Speichermedium. Es dürfen sich mehrere Projekte und Unterverzeichnisse auf einer Speicherkarte befinden. Bitte beachten Sie, dass sich Ihre aktuelle Projektierung im Root-Verzeichnis befindet und einen der folgenden Dateinamen hat:

- S7PROG.WLD
- AUTOLOAD.WLD

1. ► Erzeugen Sie im Siemens TIA Portal mit "*Projekt → Memory-Card-Datei → Neu*" eine wld-Datei.
  - ⇒ Die wld-Datei wird in der *Projektnavigation* unter "SIMATIC Card Reader" als "Memory Card File" aufgeführt.
2. ► Kopieren Sie Ihre Bausteine aus *Programmbausteine* in die wld-Datei. Hierbei werden automatisch die Hardware-Konfigurationsdaten als "Systemdaten" in die wld-Datei kopiert.

**Transfer Speicherkarte  
→ CPU**

Das Übertragen des Anwenderprogramms von der Speicherkarte in die CPU erfolgt je nach Dateiname nach Urlöschen oder nach PowerON.

- *S7PROG.WLD* wird nach Urlöschen von der Speicherkarte gelesen.
- *AUTOLOAD.WLD* wird nach NetzeIN von der Speicherkarte gelesen.

Das Blinken der MC-LED der CPU kennzeichnet den Übertragungsvorgang. Bitte beachten Sie, dass Ihr Anwenderspeicher ausreichend Speicherplatz für Ihr Anwenderprogramm bietet, ansonsten wird Ihr Anwenderprogramm unvollständig geladen und die SF-LED leuchtet.

**Transfer CPU → Speicherkarte**

Bei einer in der CPU gesteckten Speicherkarte wird durch einen Schreibbefehl der Inhalt des RAMs als S7PROG.WLD auf die Speicherkarte übertragen. Den Schreibbefehl finden Sie im Siemens TIA Portal in der Task Card "Online-Tools" im Kommandobereich unter "Speicher" als Schaltfläche [Kopiere RAM nach ROM]. Während des Schreibvorgangs blinkt die MC-LED. Erlischt die LED, ist der Schreibvorgang beendet. Soll dieses Projekt automatisch nach einem NetzeIN von der Speicherkarte geladen werden, so müssen Sie dieses auf der Speicherkarte in *AUTOLOAD.WLD* umbenennen.

**Kontrolle des Transfervorgangs**

Nach einem Zugriff auf die Speicherkarte erfolgt ein Diagnose-Eintrag der CPU. Zur Anzeige der Diagnoseeinträge gehen Sie im Siemens TIA Portal auf *Online & Diagnose*. Hier haben Sie Zugriff auf den "Diagnosepuffer". ↪ Kapitel 5.19 "VIPA-spezifische Diagnose-Einträge" auf Seite 75