



VIPA System 300S



SPEED7 - CPU SC | 312-5BE03 | Handbuch

HB140D_CPU-SC | RD_312-5BE03 | Rev. 09/45

November 2009

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 (91 32) 744 -0

Fax.: +49 9132 744 1864

E-Mail: info@vipa.de

<http://www.vipa.de>

Hinweis

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.

CE-Konformität

Hiermit erklärt VIPA GmbH, dass die Produkte und Systeme mit den grundlegenden Anforderungen und den anderen relevanten Vorschriften der folgenden Richtlinien übereinstimmen:

- 2004/108/EG Elektromagnetische Verträglichkeit
- 2006/95/EG Niederspannungsrichtlinie

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, 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

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 1150/1180 (Hotline)

E-Mail: support@vipa.de

Inhaltsverzeichnis

Über dieses Handbuch	1
Sicherheitshinweise	2
Teil 1 Grundlagen	1-1
Sicherheitshinweis für den Benutzer	1-2
Allgemeine Beschreibung System 300	1-3
Arbeitsweise einer CPU.....	1-4
Programme einer CPU	1-5
Operanden einer CPU	1-5
CPU 312SC.....	1-7
Teil 2 Montage und Aufbaurichtlinien	2-1
Übersicht.....	2-2
Einbaumaße.....	2-3
Montage	2-4
Verdrahtung	2-5
Aufbaurichtlinien.....	2-8
Teil 3 Hardwarebeschreibung	3-1
Leistungsmerkmale	3-2
Aufbau.....	3-3
Ein-/Ausgabe-Bereiche CPU 312SC	3-6
Technische Daten	3-8
Teil 4 Einsatz CPU 312SC	4-1
Montage	4-2
Anlaufverhalten	4-3
Adressierung	4-4
Adressbelegung	4-6
Initialisierung des Ethernet-PG/OP-Kanals.....	4-7
Zugriff auf integrierte Web-Seite	4-10
Projektierung als CPU 312C.....	4-11
Einstellung der CPU-Parameter	4-14
Parametrierung von Modulen	4-18
Projekt transferieren.....	4-19
Betriebszustände.....	4-23
Urlöschen	4-26
Firmwareupdate	4-28
Rücksetzen auf Werkseinstellung	4-32
Speichererweiterung mit MCC.....	4-33
Erweiterter Know-how-Schutz	4-34
MMC-Cmd - Autobefehle.....	4-36
VIPA-spezifische Diagnose-Einträge.....	4-38
Mit Testfunktionen Variablen steuern und beobachten.....	4-42

Teil 5	Einsatz E/A-Peripherie	5-1
	Übersicht	5-2
	Ein-/Ausgabe-Bereiche CPU 312SC	5-3
	Adressbelegung	5-5
	Digital-Teil	5-6
	Digital-Teil - Parametrierung	5-8
	Zähler - Schnelleinstieg	5-9
	Zähler - Steuern	5-14
	Zähler - Funktionen	5-18
	Zähler - Zusatzfunktionen.....	5-24
	Zähler - Diagnose und Alarm.....	5-31
Teil 6	Einsatz PtP-Kommunikation.....	6-1
	Schnelleinstieg	6-2
	Prinzip der Datenübertragung	6-3
	Einsatz der RS485-Schnittstelle	6-4
	Parametrierung	6-5
	Kommunikation	6-8
	Protokolle und Prozeduren	6-14
	Modbus - Funktionscodes	6-18
	Modbus - Beispiel zur Kommunikation.....	6-22
Teil 7	WinPLC7	7-1
	Systemvorstellung	7-2
	Installation	7-3
	Beispiel zur Projektierung.....	7-4
Anhang		A-1
	Index	A-1

Über dieses Handbuch

Das Handbuch beschreibt die SPEED7 CPU 312SC aus dem System 300S von VIPA. Hier finden Sie alle Informationen, die für Inbetriebnahme und Betrieb erforderlich sind.

Überblick

Teil 1: Grundlagen

Im Rahmen dieser Grundlagen folgen Hinweise im Umgang und Informationen über Projektierung eines System 300 mit der CPU 312SC von VIPA. Auch finden Sie hier allgemeine Angaben wie Maße, und Umgebungsbedingungen.

Teil 2: Montage und Aufbaurichtlinien

In diesem Kapitel finden Sie alle Informationen, die für den Aufbau und die Verdrahtung einer Steuerung aus den Komponenten des System 300 in Verbindung mit der CPU 312SC erforderlich sind.

Teil 3: Hardwarebeschreibung

Hier wird näher auf die Hardware-Komponenten der CPU 312SC eingegangen. Die Technischen Daten finden Sie am Ende des Kapitels.

Teil 4: Einsatz CPU 312SC

In diesem Kapitel ist der Einsatz der CPU 312SC mit SPEED7-Technologie im System 300 beschrieben. Die Beschreibung bezieht sich hierbei auf die CPU direkt und auf den Einsatz in Verbindung mit Peripherie-Modulen, die sich zusammen mit der CPU am Bus auf einer Profilschiene befinden und über den Rückwandbus verbunden sind.

Teil 5: Einsatz E/A-Peripherie

Das Kapitel stellt alle Informationen zur Verfügung, die zum Einsatz der Ein-/ Ausgabe-Peripherie der CPU 312SC erforderlich sind.

Teil 6: Einsatz PtP-Kommunikation

Inhalt dieses Kapitels ist der Einsatz der RS485-Schnittstelle für die serielle PtP-Kommunikation. Sie erhalten hier alle Informationen zu den Protokollen und zur Projektierung der Schnittstelle, die für die serielle Kommunikation über RS485 erforderlich sind.

Teil 7: WinPLC7

In diesem Teil wird die Programmier- und Simulationssoftware WinPLC7 von VIPA vorgestellt. WinPLC7 eignet sich für alle mit Siemens STEP[®]7 programmierbaren Steuerungen.

Neben der Systemvorstellung und der Installation finden Sie hier die Grundzüge der Programmbedienungen an einem Beispielprojekt erklärt. Nähere Informationen zum Einsatz von WinPLC7 können Sie der Online-Hilfe bzw. der Online-Dokumentation von WinPLC7 entnehmen.

Zielsetzung und Inhalt

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

Dieses Handbuch ist Bestandteil des Dokumentationspakets mit der Best.-Nr.: HB140D_CPU_SC und gültig für :

Produkt	Best.-Nr.	ab Stand: CPU-HW	CPU-FW
CPU 312SC	VIPA 312-5BE03	01	V328

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
- Übersicht der beschriebenen Themen am Anfang jedes Kapitels
- Stichwortverzeichnis (Index) am Ende des Handbuchs

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.

**Achtung!**

Bei Nichtbefolgen sind Sachschäden möglich.

**Hinweis!**

Zusätzliche Informationen und nützliche Tipps

Sicherheitshinweise

Bestimmungsgemäße Verwendung

Die SPEED7-CPU ist konstruiert und gefertigt für:

- alle VIPA System-300-Komponenten
- 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



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!

Teil 1 Grundlagen

Überblick

Im Rahmen dieser Grundlagen folgen Hinweise im Umgang und Informationen über Projektierung eines System 300 mit der CPU 312SC von VIPA.

Auch finden Sie hier allgemeine Angaben wie Maße, und Umgebungsbedingungen.

Inhalt

Thema	Seite
Teil 1 Grundlagen	1-1
Sicherheitshinweis für den Benutzer	1-2
Allgemeine Beschreibung System 300	1-3
Arbeitsweise einer CPU.....	1-4
Programme einer CPU	1-5
Operanden einer CPU	1-5
CPU 312SC.....	1-7

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 Bau- gruppen

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

- Potentialfreie 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.



Achtung!

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

Allgemeine Beschreibung System 300

Das System 300

Das System 300 ist ein modulares zentral wie dezentral einsetzbares Automatisierungssystem für Anwendungen im mittleren und oberen Leistungsbereich. Die einzelnen Module werden direkt auf der Profilschiene montiert und über Busverbinder, die von hinten an die Module gesteckt werden, gekoppelt.

Die CPUs des System 300 von VIPA sind befehlskompatibel zur S7-300 von Siemens.

System 300V System 300S

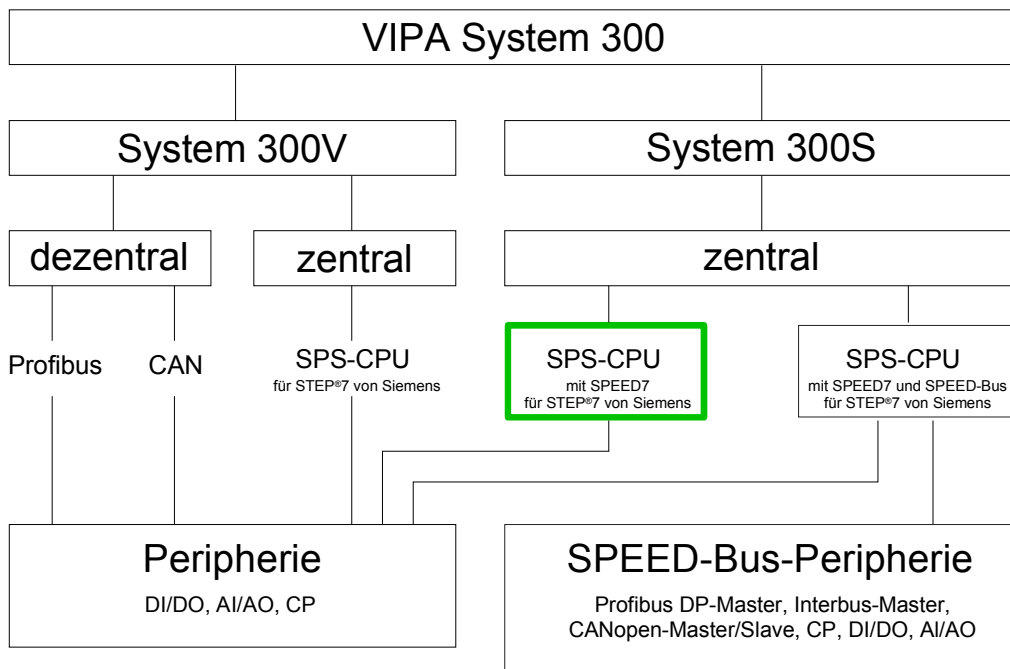
Bei VIPA wird unterschieden zwischen System 300V und System 300S.

- System 300V

Mit dem System 300V können Sie Automatisierungsaufgaben zentral und dezentral lösen. Die einzelnen Module des System 300V von VIPA sind funktionsgleich zu Siemens. Durch den kompatiblen Rückwandbus sind somit Module von VIPA und Siemens mischbar.

- System 300S

Das System 300S erweitert den zentralen Bereich um Hochgeschwindigkeits-CPU's, die den SPEED7-Chip integriert haben. Zusätzlich besitzen manche CPU's des System 300S einen parallelen SPEED-Bus, über den Sie modular schnelle Peripherie-Module, wie IO's oder Bus-Master ankopeln können.



Übersicht Handbücher

In diesem Handbuch ist eine SC-CPU aus dem System 300S beschrieben. Hierbei handelt es sich um eine CPU mit Ein-/Ausgabe-Peripherie und integrierter SPEED7-Technologie ohne SPEED-Bus.

Die Beschreibung der System 300V CPU 31x und der zugehörigen Peripherie-Module, wie digitale und analoge Ein-/Ausgabe-Module, Spannungsversorgungen und Bus-Koppler finden Sie in HB 130.

Arbeitsweise einer CPU

Allgemein

Die CPU enthält einen Standardprozessor mit internem Programmspeicher. In Verbindung mit der angekoppelten SPEED7-Technologie erhalten Sie ein leistungsfähiges Gerät zur Prozessautomatisierung innerhalb der System 300 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 Zyklusdurchlauf 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 Alarmereignis 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.

Programme einer CPU

Übersicht	Das in jeder CPU vorhandene Programm unterteilt sich in: <ul style="list-style-type: none">• 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.

Operanden einer CPU

Übersicht	Die CPU stellt Ihnen für das Programmieren folgende Operandenbereiche zur Verfügung: <ul style="list-style-type: none">• 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: <ul style="list-style-type: none">• Einzelbits• Bytes• Wörter• Doppelwörter <p>Sie können mit Ihrem Anwenderprogramm über den Bus direkt auf Peripheriebaugruppen zugreifen. Folgende Datentypen sind möglich:</p> <ul style="list-style-type: none">• 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 Anwenderprogramm 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

CPU 312SC

Übersicht

Die SC-CPU 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 WinPLC7 von VIPA oder den Siemens SIMATIC Manager verwenden.

Aufgrund des SPEED7-Chipsatzes verhält sich die CPU wie eine CPU 318. Hier kommt der Befehlssatz der S7-400 von Siemens zum Einsatz.

Die CPU besitzt einen Ethernet-PG/OP-Kanal, eine MPI- und RS485-Buchse für PtP-Kommunikation und garantiert so eine komfortable Integration in ein Netzwerk oder den Anschluss von zusätzlichen Endgeräten.

Das Anwenderprogramm wird im batteriegepufferten RAM oder auf einem zusätzlich steckbaren MMC-Speichermodul gespeichert.

Speicher- management

Die CPU hat einen Arbeitsspeicher integriert. Hiervon werden während des Programmablaufs 50% für Programmcode und 50% für Daten verwendet.

Mit einer MCC Speichererweiterungskarte haben Sie die Möglichkeit den Gesamtspeicher bis zum Maximalspeicher zu erweitern.

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 "Adresse vergeben" bzw. über ein "Minimalprojekt" können Sie über die "Zielsystem"-Funktionen den Ethernet-PG/OP-Kanal direkt ansprechen und Ihre CPU programmieren bzw. fernwarten. Hier sind maximal 2 PG/OP-Verbindungen möglich.

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...2,5mm²
- Vollisolierung der Verdrahtung bei Modulwechsel
- Potenzialtrennung aller Peripherie-Module zum Rückwandbus
- ESD/Burst gemäß IEC 61000-4-2/IEC 61000-4-4 (bis Stufe 3)
- Schockfestigkeit gemäß IEC 60068-2-6 / IEC 60068-2-27 (1G/12G)

Umgebungsbedingungen

- Betriebstemperatur: 0 ... +60°C
- Lagertemperatur: -25 ... +70°C
- Relative Feuchte: 5 ... 95% ohne Betauung
- Lüfterloser Betrieb

Aufbau/Maße

- Maße Grundgehäuse: 2fach breit: (BxHxT) in mm: 80x125x120
- Verfügbare Länge der Profilschiene in mm: 160, 482, 530, 830 und 2000

Kompatibilität

Module und CPUs aus dem System 300 von VIPA und Siemens können als Mischkonfiguration am "Standard"-Bus eingesetzt werden.

Die Projektierung erfolgt unter WinPLC7 von VIPA oder im Hardware-Konfigurator von Siemens.

Die SPEED7-CPU's von VIPA sind befehlskompatibel zur Programmiersprache STEP[®]7 von Siemens und können unter WinPLC7 von VIPA oder im Siemens SIMATIC Manager programmiert werden. Hierbei kommt der Befehlssatz der S7-400 von Siemens zum Einsatz.

**Hinweis!**

Bitte verwenden Sie zur Projektierung der VIPA CPU 312SC immer die **CPU 312C (6ES7 312-5BE03-0AB0 V2.0)** von Siemens aus dem Hardware-Katalog.

Zur Projektierung werden fundierte Kenntnisse im Umgang mit dem Siemens SIMATIC Manager vorausgesetzt!

Integriertes Netzteil

Die CPU hat ein Netzteil integriert. Das Netzteil ist mit DC 24V zu versorgen. Über die Versorgungsspannung wird neben der internen Elektronik auch der Rückwandbus für die Peripheriemodule versorgt. Das Netzteil ist gegen Verpolung und Überstrom geschützt.

Teil 2 Montage und Aufbaurichtlinien

Überblick

In diesem Kapitel finden Sie alle Informationen, die für den Aufbau und die Verdrahtung einer Steuerung aus den Komponenten des System 300 in Verbindung mit der CPU 312SC erforderlich sind.

Inhalt

Thema	Seite
Teil 2 Montage und Aufbaurichtlinien	2-1
Übersicht	2-2
Einbaumaße	2-3
Montage	2-4
Verdrahtung	2-5
Aufbaurichtlinien	2-8

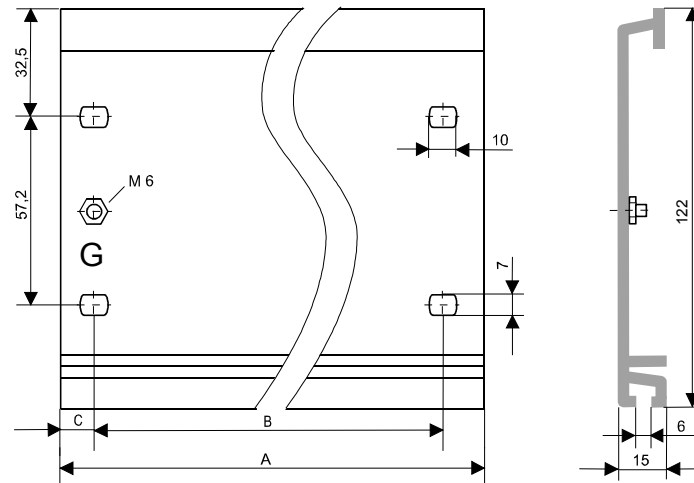
Übersicht

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

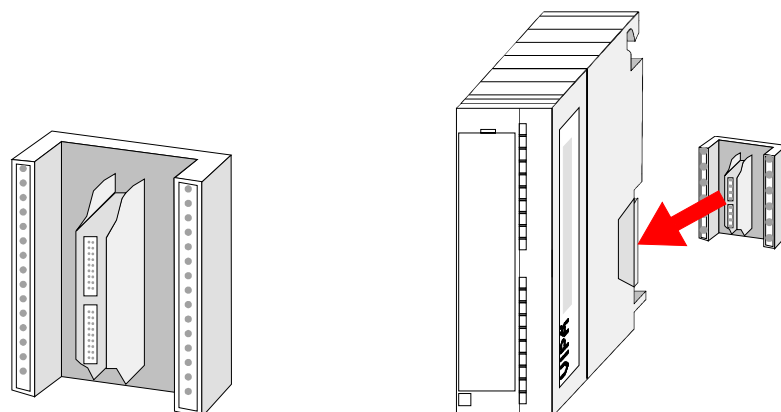


Bestellnummer	A	B	C
VIPA 390-1AB60	160mm	140mm	10mm
VIPA 390-1AE80	482mm	466mm	8,3mm
VIPA 390-1AF30	530mm	500mm	15mm
VIPA 390-1AJ30	830mm	800mm	15mm
VIPA 390-9BC00*	2000mm	Bohrungen nur links	15mm

* Verpackungseinheit 10 Stück

Busverbinder

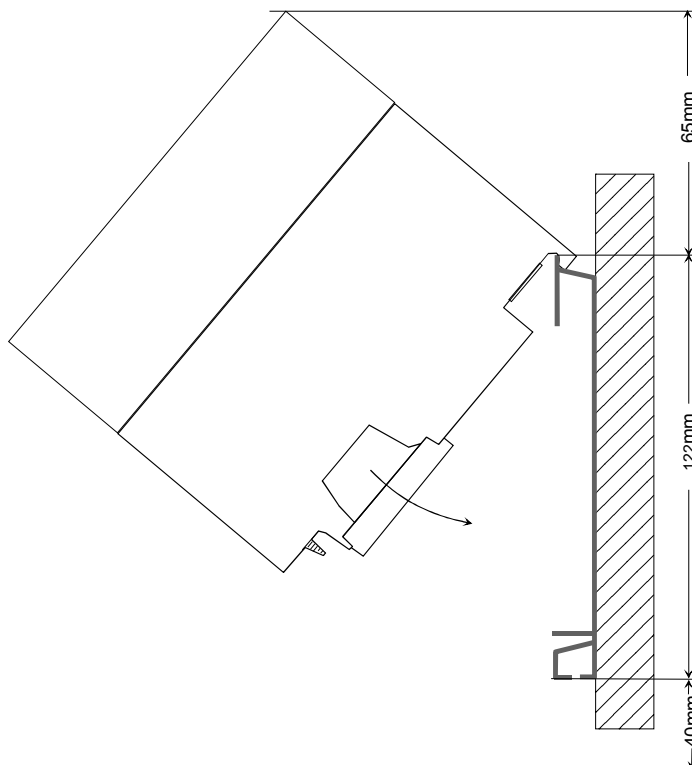
Für die Kommunikation der Module untereinander wird beim System 300 ein Rückwandbus-Verbinder eingesetzt. Die Rückwandbus-Verbinder sind im Lieferumfang der Peripherie-Module enthalten und werden vor der Montage von hinten an das Modul gesteckt.



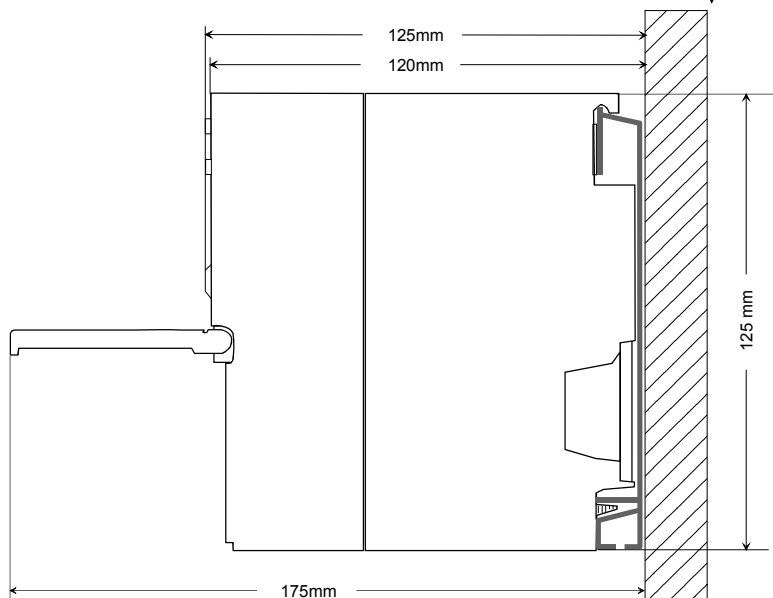
Einbaumaße

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

Montagemaße

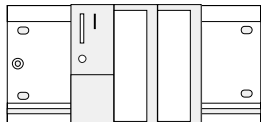


Maße montiert



Montage

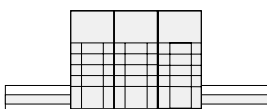
waagrechtter Aufbau



senkrechter Aufbau



liegender Aufbau



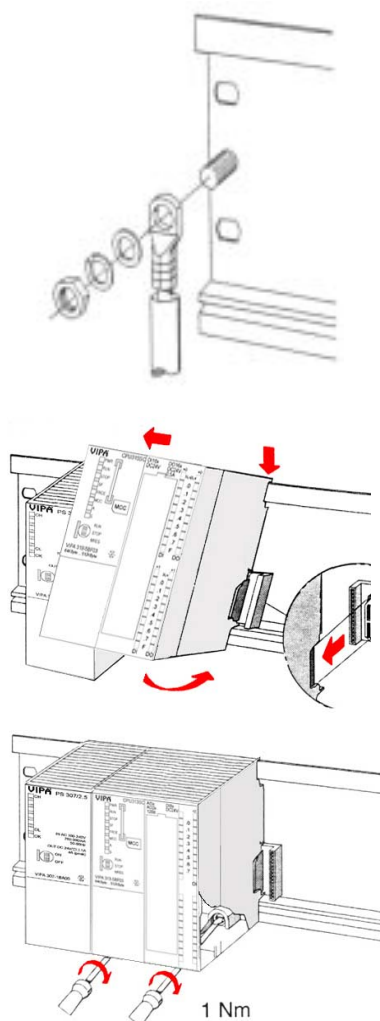
Montagemöglichkeiten

Beachten Sie bitte die hierbei zulässigen Umgebungstemperaturen:

- waagrechtter Aufbau: von 0 bis 60°C
- senkrechter Aufbau: von 0 bis 40°C
- liegender Aufbau: von 0 bis 40°C

Vorgehensweise

- Verschrauben Sie die Profilschiene mit dem Untergrund (Schraubengröße: M6) so, dass mindestens 65mm Raum oberhalb und 40mm unterhalb der Profilschiene bleibt.
- Achten Sie bei geerdetem Untergrund auf eine niederohmige Verbindung zwischen Profilschiene und Untergrund.
- Verbinden Sie die Profilschiene mit dem Schutzleiter. Für diesen Zweck befindet sich auf der Profilschiene ein Stehbolzen mit M6-Gewinde.
- Der Mindestquerschnitt der Leitung zum Schutzleiter muss 10mm² betragen.
- Hängen Sie die Spannungsversorgung ein und schieben Sie diese nach links bis an den Erdungsbolzen der Profilschiene.
- Schrauben Sie die Spannungsversorgung fest.
- Nehmen Sie einen Rückwandbus-Verbinder und stecken Sie ihn wie gezeigt von hinten an die CPU.
- Hängen Sie die CPU rechts von der Spannungsversorgung ein und schieben Sie diese bis an die Spannungsversorgung.
- Klappen Sie die CPU nach unten und schrauben Sie die CPU wie gezeigt fest.
- Verfahren Sie auf die gleiche Weise mit Ihren Peripherie-Modulen, 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.



Gefahr!

- 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.

Verdrahtung

Übersicht

Die CPU wird ausschließlich mit Federklemm-Kontakten ausgeliefert. Die Anbindung der Ein-/Ausgabe-Peripherie erfolgt über 40-fach Frontstecker.



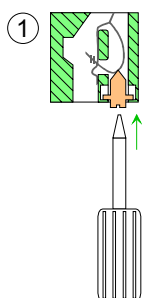
Gefahr!

- 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!

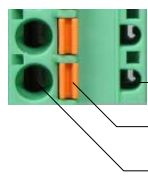
Federklemmtechnik (grün)

Zur Verdrahtung der Spannungsversorgung der CPU kommt eine grüne Anschlussklemmen mit Federzugklemmtechnik zum Einsatz.

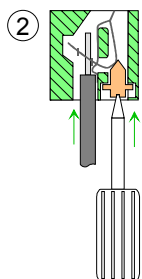
Die Anschlussklemme ist als Stecker ausgeführt, der im verdrahteten Zustand vorsichtig abgezogen werden kann.



Hier können Sie Drähte mit einem Querschnitt von $0,08\text{mm}^2$ bis $2,5\text{mm}^2$ anschließen. Hierbei dürfen sowohl flexible Litzen ohne Aderendhülse, als auch starre Leiter verwendet werden.

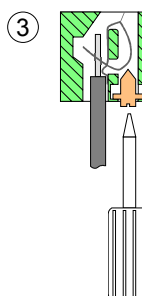


- [1] Prüfabgriff für 2mm Messspitze
- [2] Verriegelung (orange) für Schraubendreher
- [3] Runde Öffnung für Drähte



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



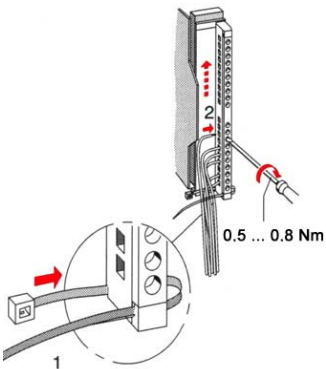
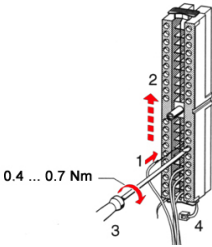
- Zum Verdrahten drücken Sie mit einem geeigneten Schraubendreher, wie in der Abbildung gezeigt, die Verriegelung senkrecht nach innen und halten Sie den Schraubendreher in dieser Position.
- Führen Sie durch die runde Öffnung Ihren abisolierten Draht ein. Sie können Drähte mit einem Querschnitt von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



Frontstecker der Ein-/Ausgabe-Module

Im System 300 gibt es 20- und 40-fach Frontstecker-Varianten. Die Anbindung der Ein-/Ausgabe-Peripherie der CPU erfolgt über 40-fach Frontstecker.

Nachfolgend wird die Verdrahtung der 2 Varianten gezeigt:

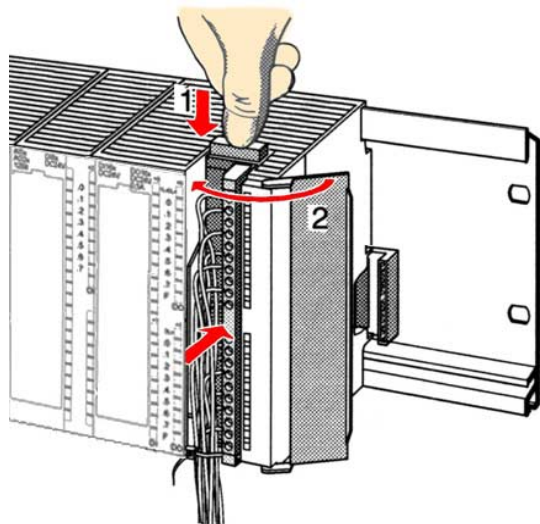
<p>20-fach Schraubtechnik VIPA 392-1AJ00</p>	<p>40-fach Schraubtechnik VIPA 392-1AM00</p>
	
<p>Öffnen Sie die Frontklappe Ihres Ein-/Ausgabe-Moduls.</p>	
<p>Bringen Sie den Frontstecker in Verdrahtungsstellung. Hierzu stecken Sie den Frontstecker auf das Modul, bis er einrastet. In dieser Stellung ragt der Frontstecker aus dem Modul heraus und hat noch keinen Kontakt.</p>	
<p>Isolieren Sie Ihre Leitungen ab. Verwenden Sie ggf. Aderendhülsen.</p>	
<p>Fädeln Sie den beiliegenden Kabelbinder in den Frontstecker ein.</p>	
<p>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.</p>	
<p>Schrauben Sie die Anschlusschrauben der nicht verdrahteten Schraubklemmen ebenfalls fest.</p>	
	<p>Legen Sie den beigefügten Kabelbinder um den Leitungsstrang und den Frontstecker herum.</p> 
<p>Ziehen Sie den Kabelbinder für den Leitungsstrang fest.</p>	

Fortsetzung ...

... Fortsetzung

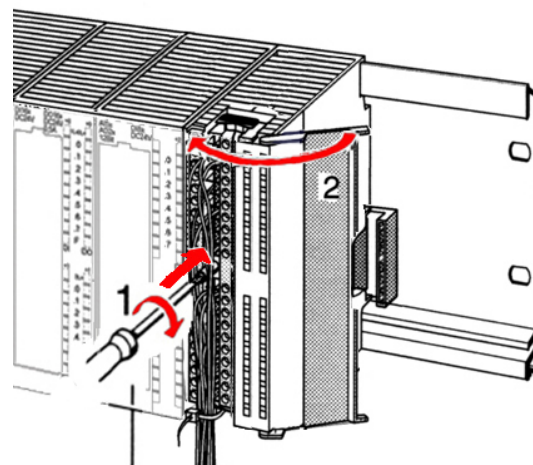
20-fach Schraubtechnik

Drücken Sie die Entriegelungstaste am Frontstecker an der Moduloberseite und drücken Sie gleichzeitig den Frontstecker in das Modul, bis er einrastet.



40-fach Schraubtechnik

Schrauben Sie die Befestigungsschraube für den Frontstecker fest.



0.4 ... 0.7 Nm

Der Frontstecker ist nun elektrisch mit Ihrem Modul verbunden.

Schließen Sie die Frontklappe.

Füllen Sie den Beschriftungsstreifen zur Kennzeichnung der einzelnen Kanäle aus und schieben Sie den Streifen in die Frontklappe.

Aufbaurichtlinien

Allgemeines	<p>Die Aufbaurichtlinien enthalten Informationen über den störsicheren Aufbau des System 300. Es wird beschrieben, wie Störungen in Ihre Steuerung gelangen können, wie die elektromagnetische Verträglichkeit (EMV) sichergestellt werden kann und wie bei der Schirmung vorzugehen ist.</p>
Was bedeutet EMV?	<p>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.</p> <p>Alle System 300 Komponenten sind für den Einsatz in rauen 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.</p>
Mögliche Störeinträge	<p>Elektromagnetische Störungen können sich auf unterschiedlichen Pfaden in Ihre Steuerung einkoppeln:</p> <ul style="list-style-type: none">• Felder• E/A-Signalleitungen• Bussystem• Stromversorgung• Schutzleitung <p>Je nach Ausbreitungsmedium (leitungsgebunden oder -ungebunden) und Entfernung zur Störquelle gelangen Störungen über unterschiedliche Kopplungsmechanismen in Ihre Steuerung.</p> <p>Man unterscheidet:</p> <ul style="list-style-type: none">• 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.
 - Beschalten Sie alle Induktivitäten mit Löschgliedern, die nicht von System 300 Modulen angesteuert werden.
 - Benutzen Sie zur Beleuchtung von Schränken Glühlampen und vermeiden Sie Leuchtstofflampen.
- 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 dem System 300 sternförmig mit dem Erde/Schutzleitersystem. Sie vermeiden so die Bildung von Erdschleifen.
 - Verlegen Sie bei Potentialdifferenzen zwischen Anlagenteilen und Schränken ausreichend dimensionierte Potentialausgleichsleitungen.

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. μA) 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 zum System 300 Modul weiter, legen Sie ihn dort jedoch **nicht** erneut auf!



Bitte bei der Montage beachten!

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

Abhilfe: Potenzialausgleichsleitung.

Teil 3 Hardwarebeschreibung

Überblick

In diesem Kapitel wird näher auf die Hardware-Komponenten der CPU 312SC eingegangen.

Die Technischen Daten finden Sie am Ende des Kapitels.

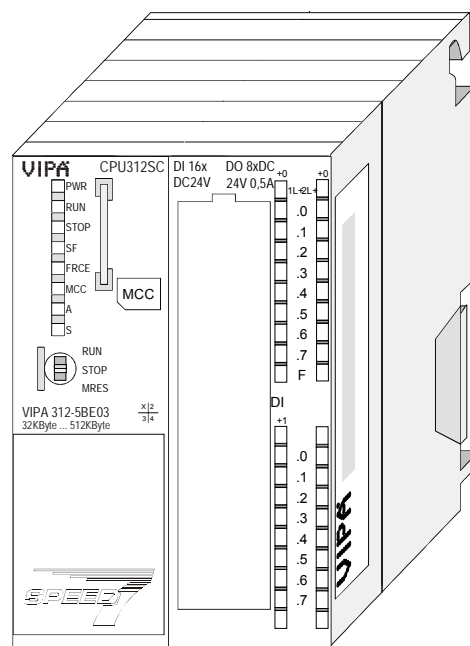
Inhalt

Thema	Seite
Teil 3 Hardwarebeschreibung	3-1
Leistungsmerkmale	3-2
Aufbau.....	3-3
Ein-/Ausgabe-Bereiche CPU 312SC	3-6
Technische Daten	3-8

Leistungsmerkmale

CPU 312SC 312-5BE03

- SPEED7-Technologie integriert
- Befehlskompatibel zu STEP®7 von Siemens mit Zugriff auf die Peripheriemodule des System 300V am Standard-Bus
- Integriertes DC24V-Netzteil
- 32kByte Gesamtspeicher (16kByte für Code, 16kByte für Daten)
- Speicher erweiterbar bis max. 512kB (256kB Code, 256kB Daten)
- MCC-Slot für externe Speichermedien und Speichererweiterung
- Status-LEDs für Betriebszustand und Diagnose
- Echtzeituhr akkugepuffert
- Ethernet-PG/OP-Schnittstelle integriert
- MPI-Schnittstelle
- RS485-Schnittstelle für PtP
- Digitale E/As: DI 16xDC24V / DO 8xDC24V, 0,5A
- 2 Zähler (10kHz)
- 512 Zeiten
- 512 Zähler
- 8192 Merker-Byte

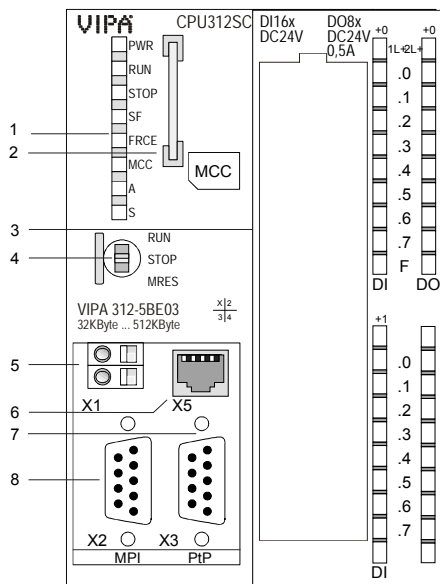


Bestelldaten

Typ	Bestellnummer	Beschreibung
CPU 312SC	VIPA 312-5BE03	MPI-Interface, Karten-Slot, Echtzeituhr, Ethernet-Interface für PG/OP, PtP über RS485, DI 16xDC24V / DO 8xDC 24V, 0,5A, 2 Zähler

Aufbau

CPU 312SC 312-5BE03



- [1] LEDs des CPU-Teils
- [2] Steckplatz für MCC
- [3] LEDs des E/A-Teils
- [4] Betriebsarten-Schalter CPU

Folgende Komponenten befinden sich unter der Frontklappe

- [5] Anschluss für DC 24V Spannungsversorgung
- [6] Ethernet-Schnittstelle für PG/OP
- [7] PtP-Schnittstelle
- [8] MPI-Schnittstelle

Komponenten

LEDs CPU-Teil

Die CPU besitzt auf der Front eine LED-Reihe. Die Verwendung und die jeweiligen Farben der LEDs finden Sie in der nachfolgenden Tabelle:

Bezeichnung	Farbe	Bedeutung
PWR	grün	CPU-Teil wird intern mit 5V versorgt
RUN	grün	CPU befindet sich im Zustand RUN
STOP	gelb	CPU befindet sich im Zustand STOP
SF	rot	Leuchtet bei System Fehler (Hardware-Defekt)
FRCE	gelb	Leuchtet, sobald Variablen geforced (fixiert) werden.
MCC	gelb	Blinkt bei Zugriff auf Speicherkarte.
A	grün	Activity: an: physikalisch verbunden aus: keine physikalische Verbindung blinkt: zeigt Ethernet-Aktivität an
S	grün	Speed: an: 100MBit aus: 10MBit



Hinweis!

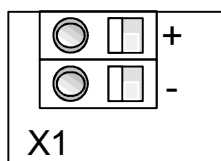
Alle LEDs des CPU-Teils blinken dreimal, bei Zugriff auf eine ungültige Speicherkarte oder wenn diese während des Lesens gezogen wird.

Steckplatz für Speichermedien

Über diesen Steckplatz können Sie eine MMC (**M**ultimedia **C**ard) als externes Speichermedium für Programme und Firmware oder eine MCC Speichererweiterungskarte stecken. Zusätzlich kann die MCC auch als externes Speichermedium eingesetzt werden.

Beide VIPA-Speicherkarten sind mit dem PC-Format FAT vorformatiert und können mit einem Kartenlesegerät beschrieben werden. Ein Zugriff auf die Speicherkarten erfolgt immer nach Urlöschen und PowerON.

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.

Bitte beachten Sie, dass das integrierte Netzteil den Rückwandbus mit max. 5A versorgen kann.

Betriebsarten-Schalter



Mit dem Betriebsartenschalter können Sie bei der CPU zwischen den Betriebsarten STOP und RUN wählen. Die Betriebsart ANLAUF wird von der CPU automatisch zwischen STOP und RUN ausgeführt.

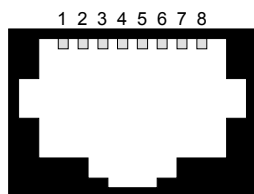
Mit der Tasterstellung MRES (Memory Reset) fordern Sie das Urlöschen an mit anschließendem Laden von Speicherkarte (Projekt oder Firmware).

Ethernet-PG/OP-Kanal

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-Site zugreifen. Projektierbare Verbindungen sind nicht möglich. Die CPU verfügt über max. 2 PG/OP-Verbindungen bei einer Übertragungsrates von 100MBit (voll duplex).

Damit Sie online auf den Ethernet-PG/OP-Kanal zugreifen können, müssen Sie diesem IP-Adress-Parameter zuweisen. Näheres hierzu finden Sie im Teil "Einsatz CPU 31..." unter "Initialisierung des Ethernet-PG/OP-Kanals". Die Buchse hat folgende Belegung:

8-polige RJ45-Buchse:



Pin	Signal	Pin	Signal
1	Transmit +	5	-
2	Transmit -	6	Receive -
3	Receive +	7	-
4	-	8	-

Speicher-management

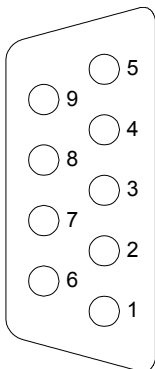
Die CPU hat einen Arbeitsspeicher integriert. Hiervon werden während des Programmablaufs 50% für Programmcode und 50% für Daten verwendet.

Sie haben die Möglichkeit den Gesamtspeicher mittels einer MCC Speichererweiterungskarte bis zum Maximalspeicher zu erweitern.

MPI-Schnittstelle X2

Die MPI-Schnittstelle dient zur Datenübertragung zwischen CPU und PC. In einer Buskommunikation können Sie Programme und Daten zwischen den CPUs transferieren, die über MPI verbunden sind.

Die MPI-Buchse hat folgende Pinbelegung:

9polige SubD-Buchse

Pin	Belegung
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.

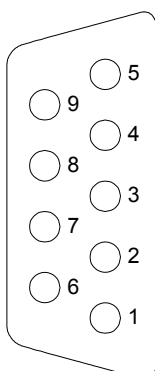
PtP-Schnittstelle X3

Die CPU 312SC besitzt eine RS485-Schnittstelle. Die Schnittstelle ist fix auf PtP-Kommunikation eingestellt.

Mit der PtP-Kommunikation 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).

Die RS485-Schnittstelle hat folgende Pinbelegung:

9-polige SubD-Buchse:

Pin	Belegung
1	Schirm
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.

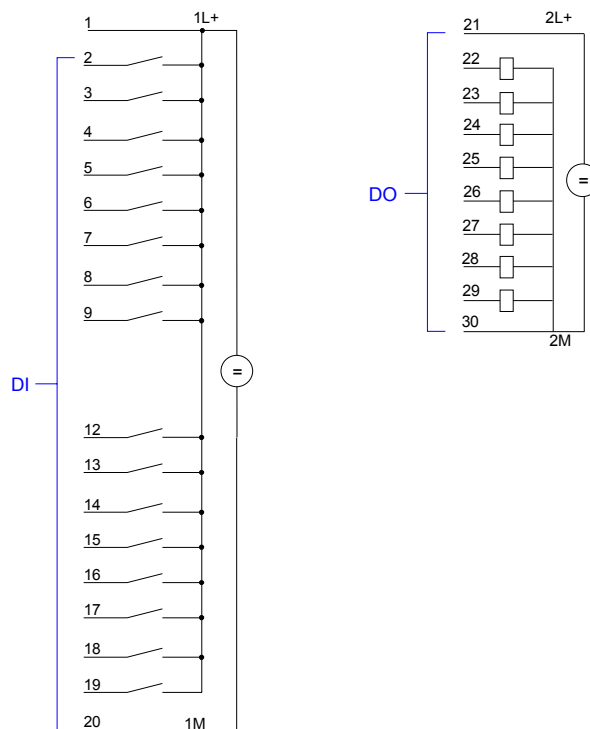
Ein-/Ausgabe-Bereiche CPU 312SC

Übersicht CPU 312SC

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

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

Jeder der digitalen Ein- bzw. Ausgänge zeigt seinen Zustand über eine LED an. Über die Parametrierung können Sie jedem digitalen Eingang Alarm-Eigenschaften zuweisen. Zusätzlich lassen sich die digitalen Eingänge als Zähler parametrieren.



Achtung!

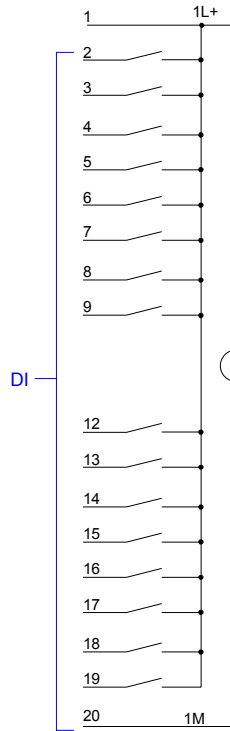
Die an einem digitalen Ausgabe-Kanal anliegende Spannung muss immer \leq der über L+ anliegenden Versorgungsspannung sein.

CPU 312SC: Steckerbelegung und Statusanzeige

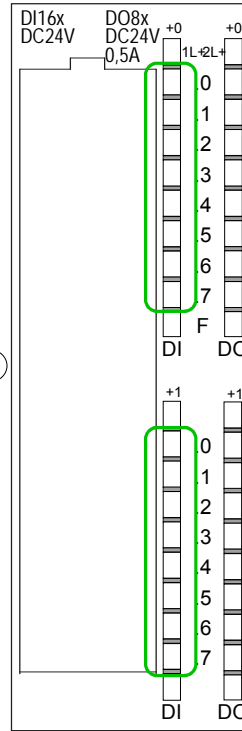
Pin Belegung

1	Versorgungssp. +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

Anschluss



LEDs



DI:
L1+ LED (grün)
Versorgungsspannung für DI liegt an

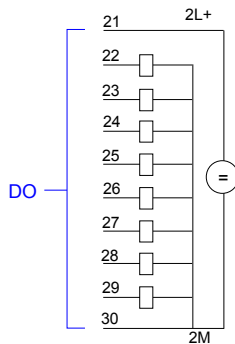
.07 LEDs (grün)
E+0.0 bis E+0.7 bzw. E+1.0 bis E+1.7 ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert

CPU 312SC: Steckerbelegung und Statusanzeige

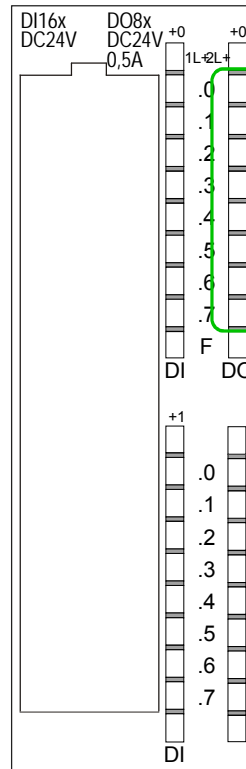
Pin Belegung

21	Versorgungssp. +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	nicht belegt
32	nicht belegt
33	nicht belegt
34	nicht belegt
35	nicht belegt
36	nicht belegt
37	nicht belegt
38	nicht belegt
39	nicht belegt
40	nicht belegt

Anschluss



LEDs



DO:
L2+ LED (grün)
Versorgungsspannung für DO liegt an

.07 LEDs (grün)
A+0.0 bis A+0.7 leuchtet bei aktiviertem Ausgang

F LED (rot)
Fehler bei Überlast oder Kurzschluss

Technische Daten

CPU 312SC

Baugruppenbezeichnung	312-5BE03
Maße und Gewicht	
Abmessungen B x H x T	80 x 125 x 120mm
Gewicht	410g
Spannung, Ströme, Potentiale	
Versorgungsnennspannung	DC 24V
- zulässiger Bereich	20,4 ... 28,8V
- Verpolschutz	ja
Stromaufnahme im Leerlauf	135mA
Einschaltstrom typ.	11A
Stromaufnahme (Nennwert)	1,0A
I^2t	500mA ² s
Externe Absicherung für Versorgungsleitungen (Empfehlung)	2A
Verlustleistung	14W
Akku	Lithium
- Pufferzeit der Uhr	30 Tage
Speicher	
Arbeitsspeicher	
- integriert	32kByte
- erweiterbar	bis 512kByte
Bearbeitungszeiten	
Bearbeitungszeiten für	
- Bitoperation	0,021µs
- Wortoperation	0,021µs
- Festpunktarithmetik	0,021µs
- Gleitpunktarithmetik	0,125µs
Zeiten/Zähler und deren Remanenz	
S7-Zähler	512
- Remanenz	einstellbar 0 ... 128
- voreingestellt	von Z0 bis Z7
- Zählbereich	0 bis 999
IEC-Zähler	ja
- Art	SFB
- Anzahl	Unbegrenzt (Begrenzung durch Arbeitsspeicher)
S7-Zeiten	512
- Remanenz	einstellbar 0 ... 128
- voreingestellt	keine Remanenz
- Zeitbereich	10ms bis 9990s
ICE-Timer	ja
- Art	SFB
- Anzahl	Unbegrenzt (Begrenzung durch Arbeitsspeicher)
Datenbereiche und deren Remanenz	
Merker	8192Byte
- Remanenz	einstellbar 0 ... 128
- voreingestellt	von MB0 bis MB15
Taktmerker	8 (1 Merkerbyte)
Datenbausteine	max. 4095
- Größe	64kByte
Lokaldaten	
- je Prioritätsklasse	510Byte

Fortsetzung ...

... Fortsetzung

Baugruppenbezeichnung	312-5BE03
Bausteine	
OBs - Größe	siehe Operationsliste 64kByte
Schachtelungstiefe - je Prioritätsklasse - zusätzlich innerhalb eines Fehler-OBs	8 4
FBs - Größe	2048 64kByte
FCs - Größe	2048 64kByte
Adressbereiche (Ein-/Ausgänge)	
Peripherieadressbereich gesamt	1024Byte/1024Byte
Prozessabbild E/A	128Byte/128Byte
Digitale Kanäle - davon zentral - integrierte Kanäle	272/264 272/264 16DI/8DO
Analoge Kanäle - davon zentral - integrierte Kanäle	64/64 64/64 keine
Ausbau	
Baugruppen über Rückwandbus	8
Uhrzeit	
Uhr - gepuffert - Pufferungsdauer - Genauigkeit	ja (HW-Uhr) ja 6 Wochen Abweichung pro Tag < 10s
Betriebsstundenzähler - Nummer - Wertebereich - Granularität - Remanent	8 0 2 ¹⁵ 1Stunde ja, muss bei jedem Neustart neu gestartet werden
Test- und Inbetriebnahmefunktion	
Status/Steuern Variable - Variable - Anzahl Variable davon Status Variable davon Steuer Variable	ja E, A, M, DB, T, Z 30 30 14
Forcen - Variable - Anzahl Variable	ja E, A 10
Status Baustein	ja
Einzelschritt	ja
Anzahl Haltepunkte	3
Diagnosepuffer - Anzahl der Einträge (nicht einstellbar)	ja 100

Fortsetzung ...

... Fortsetzung

Baugruppenbezeichnung	312-5BE03
Kommunikationsfunktionen	
PG/OP-Kommunikation	ja
Globale Datenkommunikation	ja
- Anzahl der GD-Kreise	4
- Anzahl der GD-Pakete	4
Sender	4
Empfänger	4
- Größe der GD-Pakete	22Byte
S7-Basiskommunikation	ja
- Nutzdaten pro Auftrag	76Byte
davon konsistent	76Byte (bei X_SEND bzw. X_RCV) 64Byte (bei X-PUT bzw. X_GET als Server)
Schnittstellen	
Hardwarebeschreibung 1. Schnittstelle	
Physik	RS 485
Potentialgetrennt	nein
Stromversorgung an Schnittstelle (15 bis 30V DC)	200mA
Funktionalität 1. Schnittstelle	
MPI	ja
Profibus DP	nein
Punkt-zu-Punkt-Kopplung	nein
MPI 1. Schnittstelle	
Dienste	
- PG-/OP-Kommunikation	ja
- Globaldatenkommunikation	ja
- S7-Basiskommunikation	ja
- S7-Kommunikation	
als Server	ja
als Client	nein
- Übertragungsgeschwindigkeiten	187,5kbit/s
Hardwarebeschreibung 2. Schnittstelle	
Physik	RS 485
Potenzialgetrennt	ja
Stromversorgung an Schnittstelle (15 bis 30V DC)	200mA
Funktionalität 2. Schnittstelle	
MPI	nein
Profibus DP	nein
Punkt-zu-Punkt-Kopplung	ja
- Übertragungsgeschwindigkeit halbduplex	max. 115,2kbit/s
- Protokolle	
- ASCII	ja
- STX/ETX	ja
- 3964(R)	ja
- USS	ja
- Modbus Master	ja
Hardwarebeschreibung 3. Schnittstelle	
Physik	RJ45
Funktionalität 3. Schnittstelle	
Profinet	Ethernet
Dienste	
- PG-/OP-Kommunikation	ja

Fortsetzung ...

... Fortsetzung

Integrierte Funktionen	
Anzahl der Zähler - Zählerfrequenz max.	2 10kHz
Programmierung	
Programmiersprache	KOP/FUP/AWL
Operationsvorrat	siehe Operationsliste
Klammerebenen	8
Systemfunktionen (SFC)	siehe Operationsliste
Systemfunktionsbausteine (SFB)	siehe Operationsliste
Anwenderprogrammschutz	ja

Digitale Eingabe	312-5BE03
Baugruppenspezifische Daten	
Anzahl der Eingänge	16
Leitungslänge - ungeschirmt - geschirmt	600m 1000m
Spannung, Ströme, Potentiale	
Lastnennspannung L+	DC 24V
Verpolschutz	ja
Potentialtrennung - zwischen Kanälen und Rückwandbus - zwischen den Kanälen	ja nein
zulässige Potentialdifferenz - zwischen versch. Stromkreisen	DC 75V / AC 60V
Isolation geprüft mit	DC 500V
Stromaufnahme - aus Lastspannung L+ (ohne Last)	70mA
Status, Alarmer, Diagnosen	
Statusanzeige	grüne LED pro Kanal
Alarmer	ja
Diagnosefunktionen	keine
Daten zur Auswahl eines Gebers	
Eingangsnennspannung - Nennwert - für Signal "1" - für Signal "0"	DC 24V 15V bis 28,8V 0V bis 5V
Eingangsstrom - bei Signal "1"	6mA
Eingangsverzögerung (bei Nennwert der Eingangsspannung) - Eingänge parametrierbar	0,1ms-0,35ms
Eingangskennlinie	nach IEC 1131, Typ 1
Anschluss von 2-Draht-BEROs - zulässiger Ruhestrom	möglich max. 1,5mA

Fortsetzung ...

... Fortsetzung

Digitale Ausgabe	312-5BE03
Baugruppenspezifische Daten	
Anzahl der Ausgänge	8
Leitungslänge - ungeschirmt - geschirmt	600m 1000m
Spannung, Ströme, Potentiale	
Lastnennspannung L+	DC 24V
Verpolschutz	nein
Summenstrom der Ausgänge - waagerechter Aufbau bis 40°C - waagerechter Aufbau bis 60°C - senkrechter Aufbau bis 40°C	3A 2A 2A
Potentialtrennung - zwischen Kanälen und Rückwandbus - zwischen den Kanälen in Gruppen zu 8	ja nein ja
zulässige Potentialdifferenz - zwischen versch. Stromkreisen	DC 75V / AC 60V
Isolation geprüft mit	DC 500V
Stromaufnahme - aus Lastspannung L+	100mA
Status, Alarmer, Diagnosen	
Statusanzeige	grüne LED pro Kanal
Alarmer	keine
Diagnosefunktionen - Sammelfehler-LED - Versorgungsspannungsanzeige - Kanalfehleranzeige	rote F-LED pro Gruppe grüne LED pro Gruppe keine
Daten zur Auswahl eines Aktors	
Ausgangsspannung - für Signal "1"	min. L+ (-0,8V)
Ausgangsstrom - bei Signal "1" Nennwert - bei Signal "0"	0,5A 5mA bis 0,6A 0,5mA
Lastwiderstandsbereich	48Ω bis 4kΩ
Lampenlast	5W
Parallelschaltung von zwei Ausgängen - zur redundanten Ansteuerung einer Last - zur Leistungserhöhung	möglich nicht möglich
Ansteuern eines Digitaleingangs	möglich
Schaltfrequenz - bei ohmscher Last - bei induktiver Last (IEC 947-5 DC 13) - bei Lampenlast	2,5kHz 0,5Hz 2,5kHz
Begrenzung (intern) der induktiven Abschaltspannung auf	typ. L+ (-52V)
Kurzschlusschutz des Ausgangs - Ansprechschwelle	ja, elektronisch typ. 1A

Teil 4 Einsatz CPU 312SC

Überblick

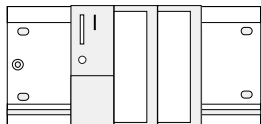
In diesem Kapitel ist der Einsatz der CPU 312SC mit SPEED7-Technologie im System 300 beschrieben. Die Beschreibung bezieht sich hierbei auf die CPU direkt und auf den Einsatz in Verbindung mit Peripherie-Modulen, die sich zusammen mit der CPU am Bus auf einer Profilschiene befinden und über den Rückwandbus verbunden sind.

Inhalt

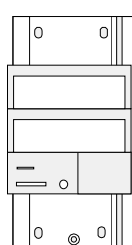
Thema	Seite
Teil 4 Einsatz CPU 312SC	4-1
Montage	4-2
Anlaufverhalten	4-3
Adressierung	4-4
Adressbelegung	4-6
Initialisierung des Ethernet-PG/OP-Kanals.....	4-7
Zugriff auf integrierte Web-Seite	4-10
Projektierung als CPU 312C.....	4-11
Einstellung der CPU-Parameter	4-14
Parametrierung von Modulen	4-18
Projekt transferieren.....	4-19
Betriebszustände.....	4-23
Urlöschen.....	4-26
Firmwareupdate	4-28
Rücksetzen auf Werkseinstellung	4-32
Speichererweiterung mit MCC.....	4-33
Erweiterter Know-how-Schutz	4-34
MMC-Cmd - Autobefehle.....	4-36
VIPA-spezifische Diagnose-Einträge.....	4-38
Mit Testfunktionen Variablen steuern und beobachten.....	4-42

Montage

waagrechtlicher Aufbau



senkrechter Aufbau



liegender Aufbau



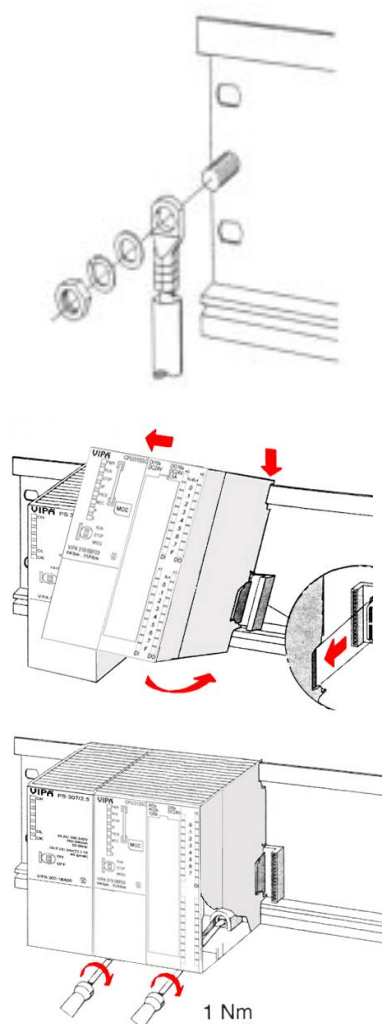
Montagemöglichkeiten

Beachten Sie bitte die hierbei zulässigen Umgebungstemperaturen:

- waagrechtlicher Aufbau: von 0 bis 60°C
- senkrechter Aufbau: von 0 bis 40°C
- liegender Aufbau: von 0 bis 40°C

Vorgehensweise

- Verschrauben Sie die Profilschiene mit dem Untergrund (Schraubengröße: M6) so, dass mindestens 65mm Raum oberhalb und 40mm unterhalb der Profilschiene bleibt.
- Achten Sie bei geerdetem Untergrund auf eine niederohmige Verbindung zwischen Profilschiene und Untergrund.
- Verbinden Sie die Profilschiene mit dem Schutzleiter. Für diesen Zweck befindet sich auf der Profilschiene ein Stehbolzen mit M6-Gewinde.
- Der Mindestquerschnitt der Leitung zum Schutzleiter muss 10mm² betragen.
- Hängen Sie die Spannungsversorgung ein und schieben Sie diese nach links bis an den Erdungsbolzen der Profilschiene.
- Schrauben Sie die Spannungsversorgung fest.
- Nehmen Sie einen Rückwandbus-Verbinder und stecken Sie ihn wie gezeigt von hinten an die CPU.
- Hängen Sie die CPU rechts von der Spannungsversorgung ein und schieben Sie diese bis an die Spannungsversorgung.
- Klappen Sie die CPU nach unten und schrauben Sie die CPU wie gezeigt fest.
- Verfahren Sie auf die gleiche Weise mit Ihren Peripherie-Modulen, 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.



Gefahr!

- 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.

Anlaufverhalten

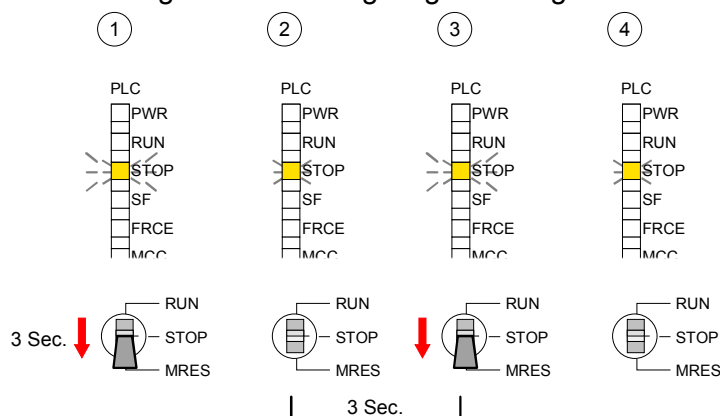
Stromversorgung einschalten

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

Sie können jetzt aus Ihrem Projektier-Tool heraus über MPI Ihr Projekt in die CPU übertragen bzw. eine MMC mit Ihrem Projekt stecken und Urlöschen ausführen.

Urlöschen

Die nachfolgende Abbildung zeigt die Vorgehensweise:



Hinweis!

Das Übertragen des Anwenderprogramms von der MMC in die CPU erfolgt immer nach Urlöschen!

Auslieferungszustand

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

Anlauf mit gültigen Daten 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.

Informationen, wie sie Bausteine geschützt in der CPU ablegen, finden Sie in diesem Kapitel unter "Erweiterter Know-how-Schutz".

Abhängig von der Stellung des RUN/STOP-Schalters geht die CPU in RUN bzw. bleibt im STOP.

Dieser Vorgang wird im Diagnosepuffer unter folgendem Eintrag festgehalten: "Start Urlöschen automatisch (ungepuffert NETZ-EIN)".

Adressierung

Übersicht

Damit die integrierte Ein-/Ausgabe-Peripherie und 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 Hardwareprojektierung vorliegt, legt die CPU gesteckte Analogmodule bei der automatischen Adressierung auf gerade Adressen ab 256 ab.

Die integrierte Ein-/Ausgabe-Peripherie wird ebenfalls im Adressbereich der CPU eingebunden. Näheres hierzu finden Sie unter "Adressbelegung".

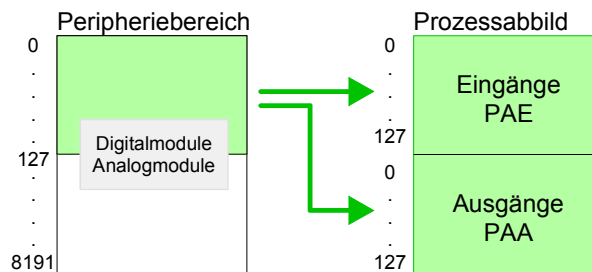
Adressierung Rückwandbus Peripherie

Bei der CPU 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 312SC kann maximal 8 Module adressieren. Eine Erweiterung mittels Erweiterungs-Rack ist nicht möglich.

Über Hardware-Konfiguration Adressen definieren

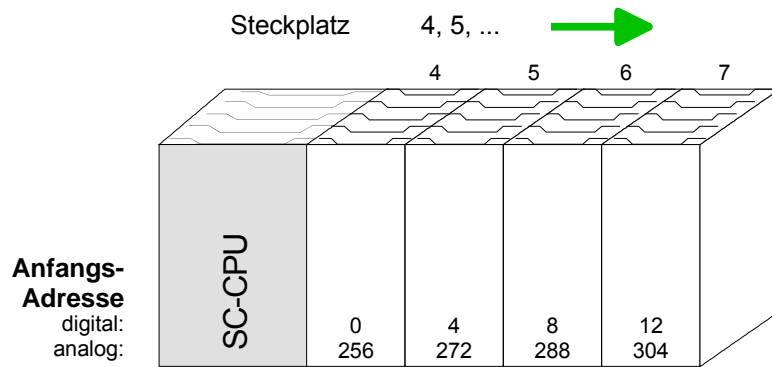
Über Lese- bzw. Schreibzugriffe auf die Peripheriebytes oder auf das Prozessabbild können Sie die Module ansprechen.

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

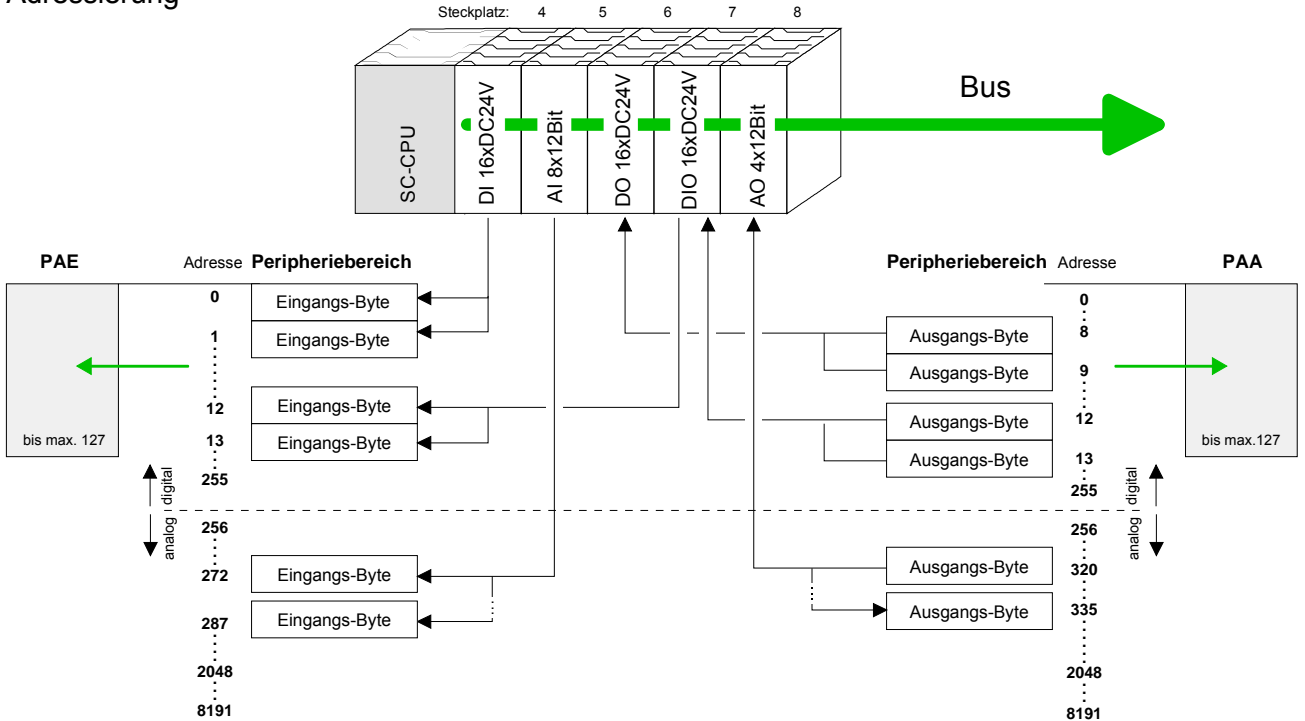
Falls Sie keine Hardware-Konfiguration verwenden möchten, tritt eine automatische Adressierung in Kraft. Bei der automatischen Adressierung belegen steckplatzabhängig DIOs immer 4Byte und AIOs, FMs, CPs immer 16Byte am Bus. Nach folgenden Formeln wird steckplatzabhängig die Anfangsadresse ermittelt,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

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



Beispiel Automatische Adressierung

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



Adressbelegung

Eingabebereich

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

Ausgabebereich

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

Initialisierung des Ethernet-PG/OP-Kanals

Übersicht

Die CPU 312SC hat einen Ethernet-PG/OP-Kanal integriert. Über diesen Kanal können Sie mit max. 2 Verbindungen Ihre CPU programmieren und fernwarten.

Mit dem 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.

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".

Möglichkeiten der "Urtaufe"

Für die Zuweisung von IP-Adress-Parametern (Urtaufe) haben Sie folgende Möglichkeiten:

- Zielsystemfunktionen über *Ethernet-Adresse vergeben*
- Hardwareprojektierung mit CP (Minimalprojekt)

Voraussetzung

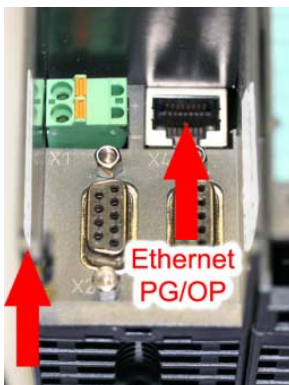
Hierzu ist folgende Software erforderlich:

- SIMATIC Manager von Siemens ab V. 5.1
- SIMATIC NET

"Urtaufe" über Zielsystemfunktionen

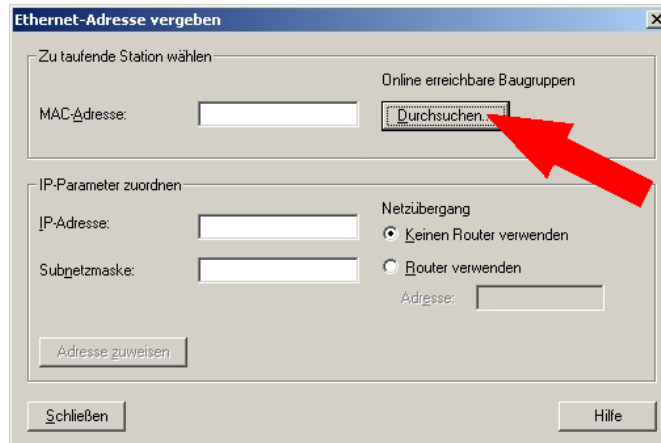
Die Urtaufe über die Zielsystemfunktion erfolgt nach folgender Vorgehensweise:

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

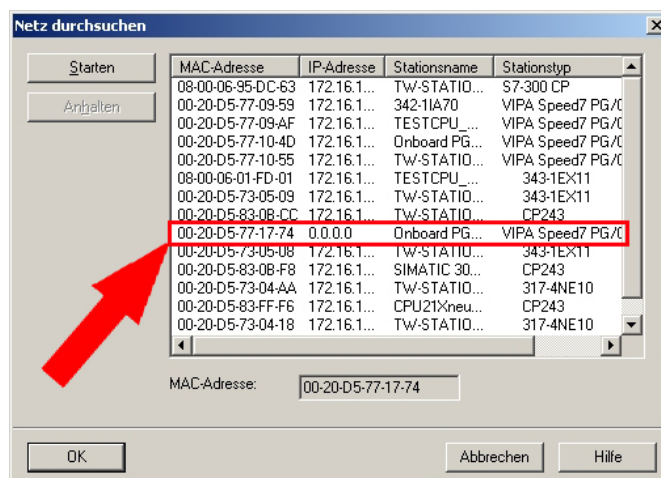


Ethernet address
Ethernet PG/OP

- Stellen Sie eine Netzwerkverbindung zwischen dem Ethernet-PG/OP-Kanal der CPU und Ihrem PC her.
- Starten Sie auf Ihrem PC den Siemens SIMATIC Manager.
- Stellen Sie über **Extras** > *PG/PC-Schnittstelle einstellen* den *Zugriffsweg* auf "TCP/IP -> Netzwerkkarte Protokoll RFC 1006" ein.
- Öffnen Sie mit **Zielsystem** > *Ethernet-Adresse vergeben* das Dialogfenster zur "Taufe" einer Station.



- Benutzen Sie die Schaltfläche [Durchsuchen], um die über MAC-Adresse erreichbare CPU-Komponente zu ermitteln. Solange der Ethernet-PG/OP-Kanal noch nicht initialisiert wurde, besitzt dieser die IP-Adresse 0.0.0.0 und den Stationsnamen "Onboard PG/OP".

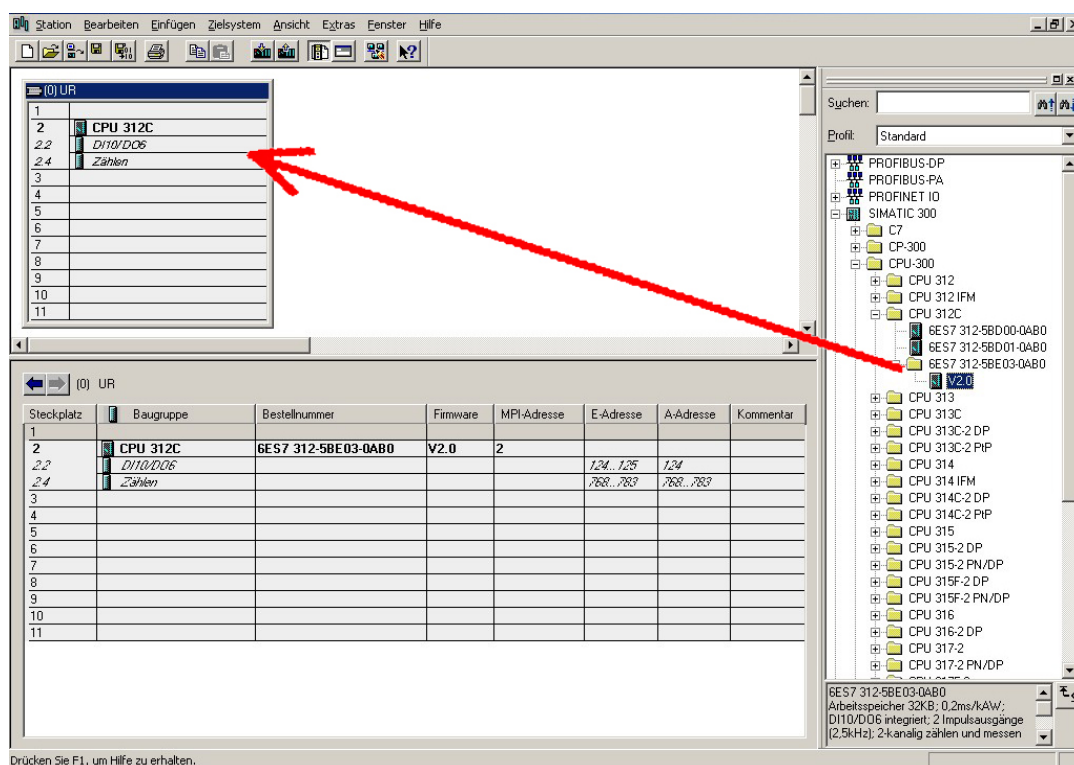


- Wählen Sie die ermittelte Baugruppe an und klicken Sie auf [OK].
- Stellen Sie nun die IP-Konfiguration ein, indem Sie IP-Adresse, Subnetzmaske und den Netzübergang eintragen. Sie können aber auch über einen DHCP-Server eine IP-Adresse beziehen. Hierzu ist dem DHCP-Server je nach gewählter Option die MAC-Adresse, der Gerätenamen oder die hier eingebare Client-ID zu übermitteln. Die Client-ID ist eine Zeichenfolge aus maximal 63 Zeichen. Hierbei dürfen folgende Zeichen verwendet werden: Bindestrich "-", 0-9, a-z, A-Z
- Bestätigen Sie Ihre Eingabe mit der Schaltfläche [Adresse zuweisen].

Direkt nach der Zuweisung ist die CPU-Komponente mit dem Siemens SIMATIC Manager über die angegebenen IP-Adress-Parameter und dem *Zugriffsweg* "TCP/IP -> Netzwerkkarte Protokoll RFC 1006" erreichbar.

Urtaufe über Minimalprojekt

- Stellen Sie eine Netzwerkverbindung zwischen dem Ethernet-PG/OP-Kanal der CPU und Ihrem PC her.
- Starten Sie den Siemens SIMATIC Manager und legen Sie ein neues Projekt an.
- Fügen Sie mit **Einfügen** > *Station* > *SIMATIC 300-Station* eine neue System 300-Station ein.
- Aktivieren Sie die Station "SIMATIC 300" und öffnen Sie den Hardware-Konfigurator indem Sie auf "Hardware" klicken.
- Projektieren Sie ein Rack (SIMATIC 300 \ Rack-300 \ Profilschiene)
- Platzieren Sie aus dem Hardware-Katalog die entsprechende CPU 312C von Siemens mit der Best.-Nr. 6ES7 312-5BE03-0AB0 V2.0. Sie finden diese unter SIMATIC 300 \ CPU 300 \ CPU 312C.
- Binden Sie auf Steckplatz 4 den CP 343-1EX11 ein (SIMATIC 300 \ CP 300 \ Industrial Ethernet \ CP 343-1).



- Geben Sie unter "Eigenschaften" des CP 343-1 die gewünschte IP-Adresse und Subnetzmaske an und vernetzen Sie den CP mit "Ethernet".
- Speichern und übersetzen Sie Ihr Projekt.
- Übertragen Sie Ihr Projekt via MPI oder MMC in Ihre CPU. Näheres zu den Transfermethoden finden Sie unter "Projekt transferieren".

Direkt nach der Projektübertragung ist der Ethernet-PG/OP-Kanal mit dem Siemens SIMATIC Manager über die angegebenen IP-Adress-Parameter und dem *Zugriffsweg* auf "TCP/IP -> Netzwerkkarte Protokoll RFC 1006" erreichbar.

Zugriff auf integrierte Web-Seite

Zugriff auf Web-Seite

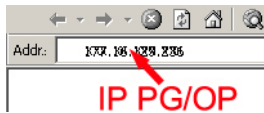
Über die IP-Adresse des Ethernet-PG/OP-Kanals steht Ihnen eine Web-Seite zur Verfügung, die Sie mit einem Internet-Browser aufrufen können. Auf der Web-Seite finden Sie Informationen zu Firmwarestand, aktuelle Zyklus-Zeiten usw.

Mit dem MMC-Cmd WEBPAGE wird der aktuelle Inhalt der Web-Seite auf MMC gespeichert. Nähere Informationen hierzu finden Sie unter "MMC-Cmd - Autobefehle".

Voraussetzung

Es wird vorausgesetzt, dass zwischen dem PC mit Internet-Browser und der CPU 312SC 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 WITH ETHERNET-PG/OP

Slot 100

```
VIPA 312-5BE03 V3.2.9 Px000074.pkg,
SERIALNUMBER 02118
SUPPORTDATA: PRODUCT V3290, HARDWARE ...
OnBoardEthernet : MacAddress : 0020d5771524,
IP-Address : , SubnetMask : , Gateway :
Cpu state : RUN
FunctionRS485 X2 : MPI
FunctionRS485 X3 : PtP
```

```
Cycletime [microseconds] : min=17000
cur=17000 ave=17000 max=17000
```

MCC-Trial-Time: 70:23

Slot 202

```
VIPA DI16/DO8 V3.2.9,SUPPORTDATA:PRODUCT...
SUPPORTDATA: PRODUCT V3290, Module Type ...
Address Input 124...125
Address Output 124...124
```

Slot 204

```
VIPA 2 COUNTERS V3.2.9,
SUPPORTDATA: PRODUCT V3290, Module Type ...
Address Input 768...783
Address Output 768...783
```

Standard Bus 8 Bit Mode

Best.-Nr., Firmware-Vers., Package, Serien-Nr.

Angaben für den Support
Ethernet-PG/OP: Adressangaben

CPU-Statusangabe
RS485-Funktion von X2
RS485-Funktion von X3

CPU-Zykluszeit:
min= minimale, cur= aktuelle,
ave= durchschnittliche, max= maximale

Verbleibende Zeit bis bei gezogener MCC der Erweiterungsspeicher wieder deaktiviert wird.

Zusätzliche CPU-Komponenten:

Slot 202 (Digitale E/As)
Name, Firmware-Version, Modul-Typ
Angaben für den Support
Projektierte Eingabe-Basis-Adressen
Projektierte Ausgabe-Basis-Adressen
Slot 204 (Zähler)
Name, Firmware-Version, Modul-Typ
Angaben für den Support
Projektierte Eingabe-Basis-Adressen
Projektierte Ausgabe-Basis-Adressen

Module am Standard-Bus

Projektierung als CPU 312C

Übersicht

Die Projektierung der CPU 312SC erfolgt im Hardware-Konfigurator von Siemens und besteht aus folgenden Teilen:

- Projektierung CPU 312SC als CPU 312C von Siemens (6ES7 312-5BE03-0AB0 V2.0)
- Projektierung der am Bus gesteckten Module
- Projektierung immer als letztes Modul Ethernet-PG/OP-Kanal als CP 343-1 (343-1EX11-0XE0)

Voraussetzung

Der Hardware-Konfigurator ist Bestandteil des Siemens SIMATIC Managers. Er dient der Projektierung. 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!

Schnelleinstieg

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

Steckpl.	Modul
1	
2	CPU 312C
2.2	<i>DI10/DO6</i>
2.4	<i>Zähler</i>
3	
Module am Bus	
	343-1EX11 (Ethernet-PG/OP)

- Hardware-Konfigurator von Siemens starten.
- CPU 312C (6ES7 312-5BE03-0AB0 V2.0) von Siemens projektieren.
- Beginnend mit Steckplatz 4, die System 300 Module am Bus in gesteckter Reihenfolge platzieren.
- Für den internen Ethernet-PG/OP-Kanal, den jede SC-CPU besitzt, ist immer als letztes Modul ein Siemens CP 343-1 (343-1EX11) zu platzieren. Lassen Sie unter *Optionen* die Einstellung "Projektierdaten in der CPU speichern" aktiviert!

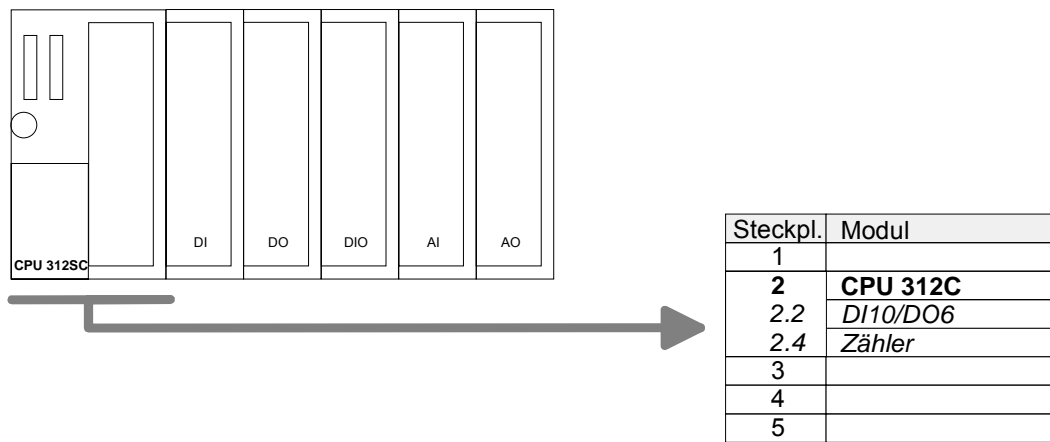
Schritte der Projektierung

Die Projektierung im Hardware-Konfigurator von Siemens gliedert sich in folgende 3 Teile:

- Projektierung der CPU
- Projektierung der gesteckten Module
- Projektierung PG/OP-Kanal

Projektierung der CPU als CPU 312C

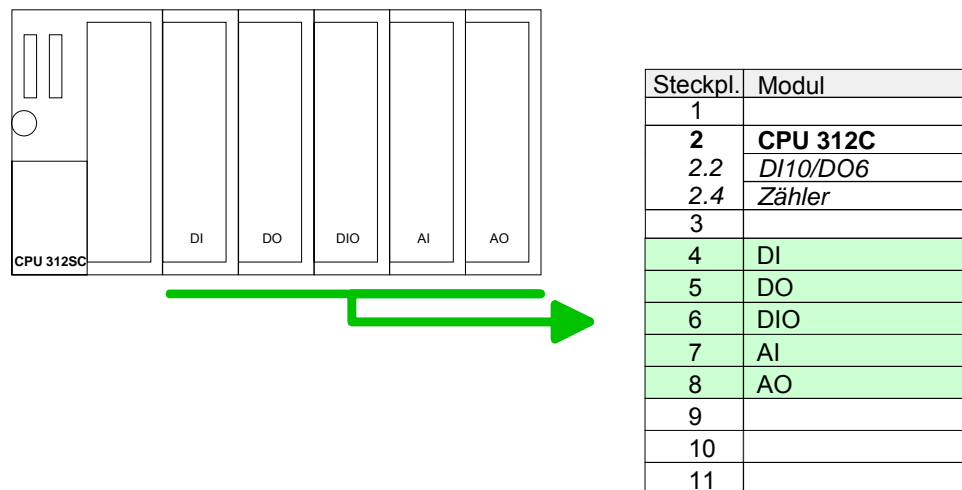
- Starten Sie den Hardware-Konfigurator von Siemens mit einem neuen Projekt und fügen Sie aus dem Hardware-Katalog eine Profilschiene ein.
- Platzieren Sie auf Steckplatz 2 folgende Siemens CPU:
CPU 312C (6ES7 312-5BE03-0AB0 V2.0)



Projektierung der Module am Bus

Die am Bus der CPU befindlichen Module sind nach folgenden Vorgehensweisen zu projektieren:

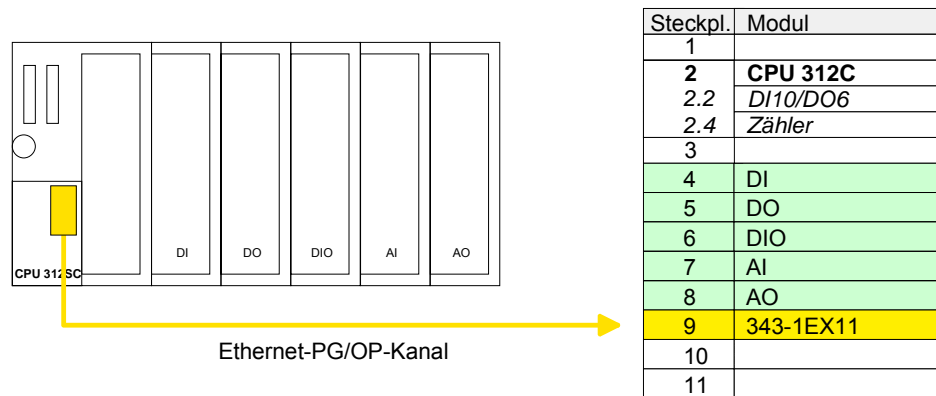
- Binden Sie beginnend mit Steckplatz 4 Ihre System 300 Module auf dem Bus in der gesteckten Reihenfolge ein.
- Parametrieren Sie ggf. die CPU bzw. die Module. Das Parameterfenster wird geöffnet, sobald Sie auf das entsprechende Modul doppelklicken.



**Projektierung
Ethernet-PG/OP-
Kanal als 343-1EX11**

Für den internen Ethernet-PG/OP-Kanal, ist immer als letztes Modul ein Siemens CP 343-1 (343-1EX11) zu platzieren. Sie finden diesen im Hardware-Katalog unter SIMATIC 300 \ CP 300 \ Industrial Ethernet \ CP 343-1 \ 6GK7 343-1EX11 0XE0.

Lassen Sie bei dem CP343-1 unter *Optionen* die Einstellung "Projektierdaten in der CPU speichern" aktiviert!



**IP-Parameter
einstellen**

Öffnen Sie durch Doppelklick auf den CP 343-1EX11 die "Objekteigenschaften". Klicken Sie unter "Allgemein" auf [Eigenschaften]. Geben Sie für den CP *IP-Adresse*, *Subnet-Maske* und *Gateway* an und wählen Sie das gewünschte *Subnetz* aus.

**Maximal 8
Peripherie-Module**

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

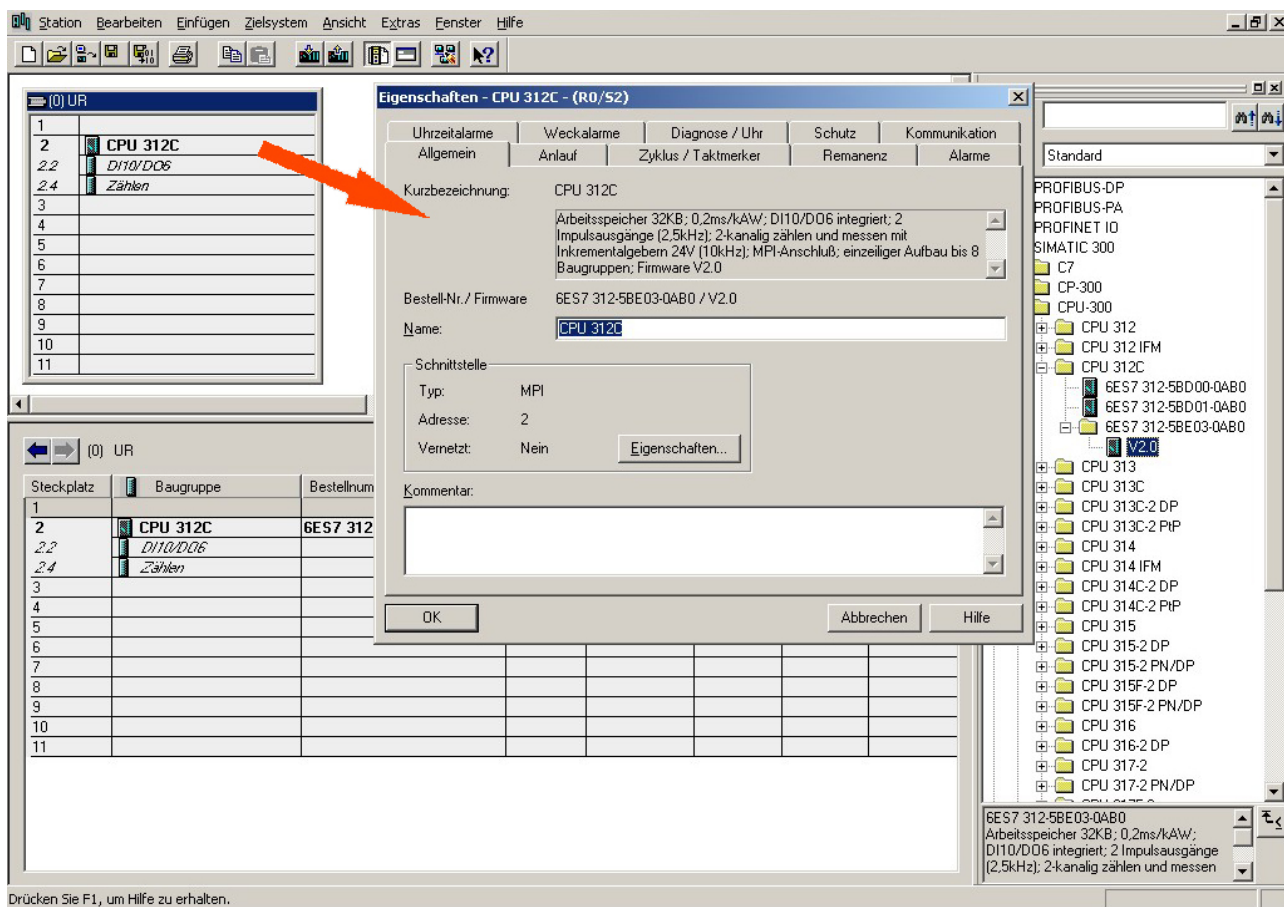
Einstellung der CPU-Parameter

Übersicht

Da die CPU 312SC von VIPA im Hardware-Konfigurator von Siemens als Siemens CPU 312C zu projektieren ist, können Sie bei der Hardware-Konfiguration unter den "Eigenschaften" der CPU 312C die Parameter für die CPU 312SC einstellen.

Durch Doppelklick auf die CPU 312C gelangen Sie in das Parametrierfenster für die CPU.

Über die Register haben Sie Zugriff auf alle Parameter Ihrer CPU.



Hinweis!

Eine Beschreibung der Parameter der Submodule *DI10/DO6* und *Zähler* finden Sie im Teil "Einsatz E/A-Peripherie".

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

Allgemein

Kurzbezeichnung Da die CPU 312SC von VIPA als CPU 312C von Siemens projektiert wird, steht hier die Kurzbezeichnung CPU 312C.

Bestell-Nr./ Firmware Bestellnummer und Firmware sind identisch zu den Angaben im Fenster "Hardware Katalog".

Name Hier steht die Kurzbezeichnung der CPU. Wenn Sie die Bezeichnung ändern, erscheint die neue Bezeichnung im 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 Sollaufbau 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 in STOP.

Wenn "Anlauf bei Sollaufbau 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 NETZ-EIN. 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

Zyklus- überwachungszeit	Hier geben Sie die Zyklusüberwachungszeit in ms ein. Wenn die Zykluszeit die Zyklusüberwachungszeit überschreitet, geht die CPU in STOP. Ursachen für eine Überschreitung: <ul style="list-style-type: none"> • Kommunikationsprozesse • Häufung von Alarmereignissen • Fehler im CPU-Programm
Zyklusbelastung durch Kommunikation	Mit diesem Parameter können Sie die Dauer von Kommunikationsprozessen, die immer auch die Zykluszeit verlängern, in bestimmten Grenzen steuern. Ohne zusätzliche asynchrone Ereignisse verlängert sich die OB 1-Zykluszeit um folgenden Faktor:

$$\frac{100}{100 - \text{Zyklusbelastung durch Kommunikation \%}}$$

Bei Einstellung der Zyklusbelastung durch Kommunikation auf 50% kann sich eine Verdopplung der OB 1-Zykluszeit ergeben. Außerdem wird der OB 1-Zyklus zusätzlich durch asynchrone Ereignisse (z.B. Prozessalarme) verlängert.

OB85-Aufruf bei Peripheriezugriffs- fehler	Sie können die voreingestellte Reaktion der CPU bei Peripheriezugriffsfehlern während der systemseitigen Aktualisierung des Prozessabbildes ändern. Die CPU 312SC ist so voreingestellt, dass sie bei Peripheriezugriffsfehlern keinen OB 85 aufruft und auch keinen Eintrag in den Diagnosepuffer erzeugt.
Taktmerker	Aktivieren Sie dieses Kästchen, wenn Sie einen Taktmerker einsetzen und geben Sie die Nummer des Merkerbytes ein.



Hinweis!

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

Remanenz

Anzahl Merkerbytes ab MB0	Hier tragen Sie die Anzahl der remanenten Merkerbytes ab Merkerbyte 0 ein.
Anzahl S7-Timer ab T0	Hier tragen Sie die Anzahl der remanenten S7-Timer ab T0 ein. Jeder S7-Timer belegt 2Byte.
Anzahl S7-Zähler ab Z0	Hier tragen Sie die Anzahl der remanenten S7-Zähler ab Z0 ein.

Alarmer

Prozessalarmer Die voreingestellte Priorität ist nicht änderbar.

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 Wird nicht unterstützt.

Weckalarmer

Priorität Die voreingestellte Priorität ist nicht änderbar.

Ausführung Geben Sie die Zeitabstände in ms an, in denen die Weckalarm-OBs bearbeitet werden. Startzeitpunkt ist der Betriebszustandwechsel von STOP nach RUN.

Phasenverschiebung Nicht einstellbar.

Teilprozessabbild 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:

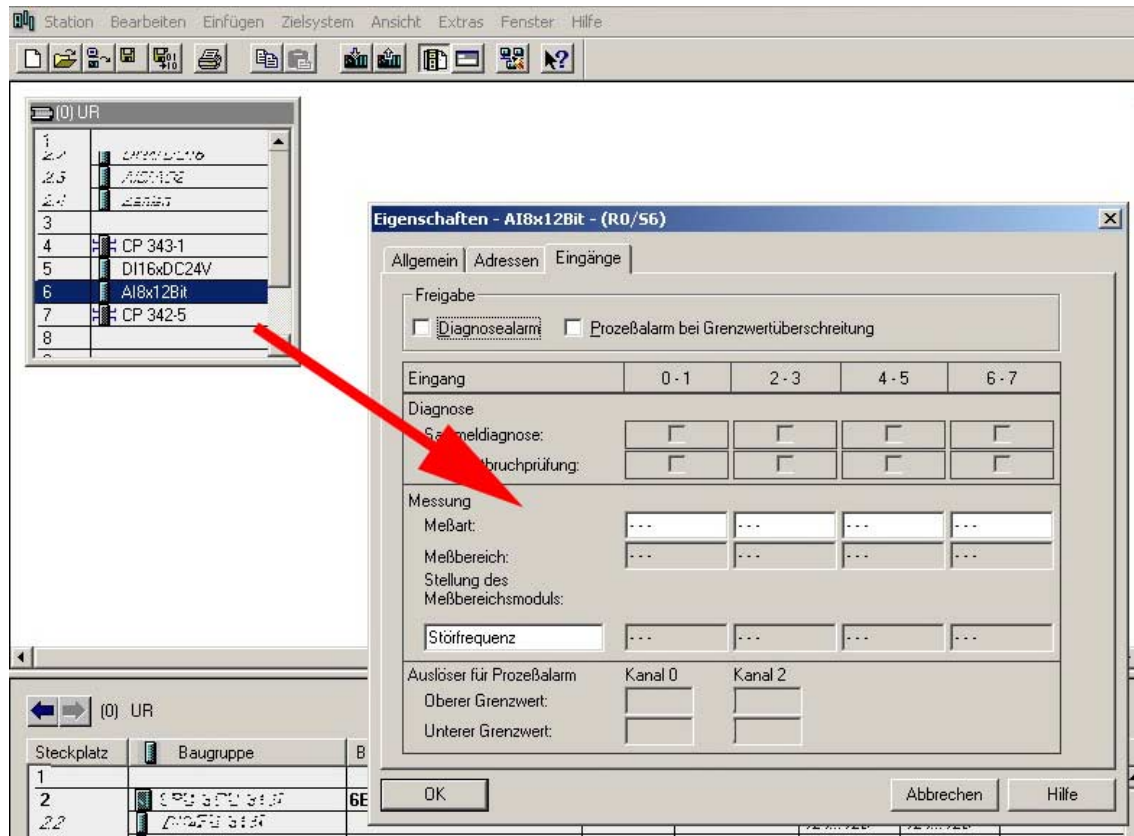
- Kenntnis des Passworts: lesender und schreibender Zugriff
- Unkenntnis des Passworts: nur lesender Zugriff

Schutzstufe 3:

- Kenntnis des Passworts: lesender und schreibender Zugriff
- Unkenntnis des Passworts: weder lesender noch schreibender Zugriff

Parametrierung von Modulen

Vorgehensweise Durch Einsatz des Siemens SIMATIC Managers können Sie jederzeit für parametrierbare System 300 Module Parameter vorgeben. Doppelklicken Sie hierzu bei der Projektierung in Ihrer Steckplatzübersicht auf das zu parametrierende Modul. In dem sich öffnenden Dialogfenster können Sie dann Ihre Parametereinstellungen vornehmen.



Parametrierung zur Laufzeit 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.

Projekt transferieren

Übersicht

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

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

Transfer über MPI

Für den Transfer über MPI besitzt die CPU eine MPI-Schnittstelle. Diese MPI-Schnittstelle unterstützt maximal 32 PG/OP-Kanäle.

MPI-Programmierkabel

Die MPI-Programmierkabel erhalten Sie in verschiedenen Varianten von VIPA. Der Einsatz dieser Kabel ist identisch. Die Kabel bieten einen busfähigen RS485-Anschluss für die MPI-Buchse der CPU und einen RS232- bzw. USB-Anschluss für den PC.

Aufgrund des RS485-Anchlusses dürfen Sie die MPI-Programmierkabel direkt auf einen an der MPI-Buchse schon gesteckten MPI-Stecker aufstecken. Jeder Busteilnehmer identifiziert sich mit einer eindeutigen MPI-Adresse am Bus, wobei die Adresse 0 für Programmiergeräte reserviert ist.

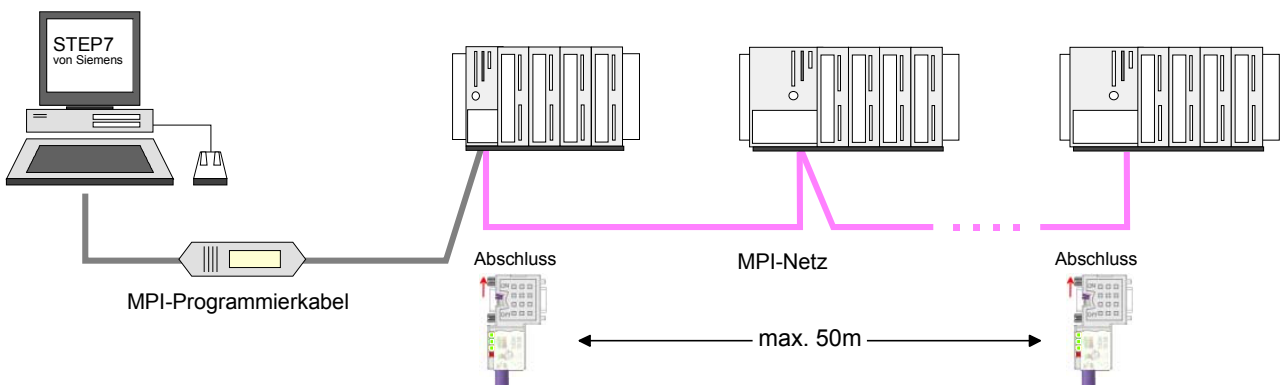
Netzstruktur

Der Aufbau eines MPI-Netzes ist prinzipiell gleich 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 diese CPU, dass die Gesamtausdehnung des MPI-Netzes 50m nicht überschreitet. Defaultmäßig wird das MPI-Netz mit 187,5kBaude betrieben. VIPA-CPU's werden mit der MPI-Adresse 2 ausgeliefert.

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, während des Hochlaufs und des Betriebs immer mit Spannung versorgt sind.



Vorgehensweise
Transfer über MPI

Maximal 32 PG/OP-Verbindungen sind unter MPI möglich. Der Transfer über MPI erfolgt nach folgender Vorgehensweise:

- Verbinden Sie Ihren PC über ein MPI-Programmierkabel mit der MPI-Buchse Ihrer CPU.
- Laden Sie im Siemens SIMATIC Manager Ihr Projekt.
- Wählen Sie im Menü **Extras** > *PG/PC-Schnittstelle einstellen*.
- Wählen Sie in der Auswahlliste "PC Adapter (MPI)" aus; ggf. müssen Sie diesen erst hinzufügen und klicken Sie auf [Eigenschaften].
- Stellen Sie im Register *MPI* die Übertragungsparameter Ihres MPI-Netzes ein und geben Sie eine gültige *Adresse* an.
- Wechseln Sie in das Register *Lokaler Anschluss*.
- Geben Sie den COM-Port des PCs an und stellen Sie für Ihr MPI-Programmierkabel die Übertragungsrate 38400Baud ein.
- 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.

**Transfer über
MMC**

Die MMC (**Memory Card**) dient als externes Speichermedium und Transfermedium für Programme und Firmware. Sie besitzt das PC-kompatible FAT16-Filesystem.

Es dürfen sich mehrere Projekte und Unterverzeichnisse auf einer MMC befinden. Bitte beachten Sie, dass sich Ihre aktuelle Projektierung bzw. die Datei mit dem reservierten Dateinamen im Root-Verzeichnis befindet.

Mit Urlöschen, NetzEIN oder CPU-STOP wird automatisch von der MMC gelesen. Durch Vorgabe eines reservierten Dateinamens können Sie die Funktionalität der CPU entsprechend beeinflussen.

Reservierte
Dateinamen

Dateiname	Beschreibung
S7PROG.WLD	Projektdatei - wird nach Urlöschen gelesen bzw. kann über Schreibbefehl von CPU geschrieben werden.
AUTOLOAD.WLD	Projektdatei - wird nach NetzEIN gelesen.
PROTECT.WLD	Geschützte Projektdatei (siehe "Erweiterter Know-how-Schutz")
VIPA_CMD.MMC	Kommando-Datei - wird bei CPU-STOP einmalig bis zum nächsten NetzEIN ausführen. (siehe "MMC-Command - Autobefehle")
*.pkg	Firmware-Datei - wird nach NetzEIN erkannt und kann mit einer Update-Anforderung installiert werden. (siehe "Firmwareupdate")

**Transfer
MMC → CPU**

Das Übertragen des Anwenderprogramms von der MMC in die CPU erfolgt je nach Dateiname nach Urlöschen oder nach PowerON. Das Blinken der LED "MCC" der CPU kennzeichnet den Übertragungsvorgang. Eine Übertragung von der MMC in die CPU erfolgt nur, wenn der Anwenderspeicher größer als das Anwenderprogramm ist. Ansonsten müssen Sie eine Komprimierung durchführen.

**Projekt-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 Hardware-Konfigurator von Siemens über **Zielsystem** > *RAM nach ROM kopieren*. Während des Schreibvorgangs blinkt die "MCC"-LED. Erlischt die LED, ist der Schreibvorgang beendet.

**Kontrolle des
Transfervorgangs**

Nach einem Schreibvorgang auf die MMC wird ein entsprechendes ID-Ereignis im Diagnosepuffer der CPU eingetragen. Zur Anzeige der Diagnoseeinträge gehen Sie im Siemens SIMATIC Manager auf **Zielsystem** > *Baugruppenzustand*. Über das Register "Diagnosepuffer" gelangen Sie in das Diagnosefenster.

Folgende Ereignisse können beim Schreiben auf eine MMC auftreten:

Ereignis-ID	Bedeutung
0xE100	MMC-Zugriffsfehler
0xE101	MMC-Fehler Filesystem
0xE102	MMC-Fehler FAT
0xE200	MMC schreiben erfolgreich beendet

Transfer über Ethernet	Die CPU besitzt für den Transfer über Ethernet einen Ethernet PG/OP-Kanal. Der Ethernet-PG/OP-Kanal unterstützt maximal 4 PG/OP-Verbindungen.
Initialisierung	Damit Sie auf den Ethernet-PG/OP-Kanal online zugreifen können, müssen Sie diesem durch die "Initialisierung" bzw. "Urtaufe" IP-Adress-Parameter zuweisen.
Ethernet Adresse ermitteln	Bei der Initialisierung ist die Ethernet (MAC) Adresse des PG/OP-Kanals anzugeben. Sie finden diese unter der Frontklappe der CPU auf einem Aufkleber auf der linken Seite. Die Adresse beginnt mit "EA: ...".
Vorgehensweise	<ul style="list-style-type: none">• Netzwerkverbindung zwischen dem Ethernet-PG/OP-Kanal der CPU und PC herstellen.• Im Siemens SIMATIC Manager über Extras > <i>PG/PC-Schnittstelle einstellen</i> den <i>Zugriffsweg</i> auf "TCP/IP -> Netzwerkkarte Protokoll RFC 1006" einstellen.• Mit Zielsystem > <i>Ethernet-Adresse vergeben</i> das Dialogfenster zur "Taufe" einer Station öffnen.• Die Station mit der entsprechenden Ethernet (MAC) Adresse suchen und die gewünschten IP-Adress-Parameter zuweisen. Solange der Ethernet-PG/OP-Kanal noch nicht initialisiert wurde, besitzt dieser die IP-Adresse 0.0.0.0 und den Stationsnamen "Onboard PG/OP". <p>Direkt nach der Initialisierung ist der Ethernet-PG/OP-Kanal der CPU mit dem Siemens SIMATIC Manager über die angegebenen IP-Adress-Parameter erreichbar. Nähere Informationen hierzu finden Sie auch unter "Initialisierung des PG/OP-Kanals."</p>
Transfer	<ul style="list-style-type: none">• Für den Transfer öffnen Sie Ihr Projekt im Siemens SIMATIC Manager.• Öffnen Sie Ihr Projekt im Siemens SIMATIC Manager.• Stellen Sie über Extras > <i>PG/PC-Schnittstelle einstellen</i> den <i>Zugriffsweg</i> "TCP/IP -> Netzwerkkarte Protokoll RFC 1006" ein.• Gehen Sie auf Zielsystem > <i>Laden in Baugruppe</i> → es öffnet sich das Dialogfenster "Zielbaugruppe auswählen". Wählen Sie die Zielbaugruppe aus und geben Sie als Teilnehmeradresse die IP-Adress-Parameter des Ethernet-PG/OP-Kanals an. Sofern keine neue Hardware-Konfiguration in die CPU übertragen wird, wird der hier angegebene Ethernet-PG/OP-Kanal dauerhaft als Transferkanal im Projekt gespeichert.• Starten Sie mit [OK] den Transfer. Systembedingt erhalten Sie eine Meldung, 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.

Betriebszustände

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 Ausgänge 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. Die Länge des OBs ist nicht beschränkt. Auch wird der Ablauf 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
- 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 digitalen Ausgänge sind freigegeben.
- RUN-LED an
- STOP-LED aus

Betriebszustand HALT	Die CPU bietet Ihnen die Möglichkeit bis zu 4 Haltepunkte zur Programm-diagnose einzusetzen. Das Setzen und Löschen von Haltepunkten erfolgt in Ihrer Programmierumgebung. Sobald ein Haltepunkt erreicht ist, können Sie schrittweise Ihre Befehlszeilen abarbeiten, wobei Ein- und Ausgänge aktiviert werden können.
Voraussetzung	Für die Verwendung von Haltepunkten müssen folgende Voraussetzungen erfüllt sein: <ul style="list-style-type: none"> • Das Testen im Einzelschrittmodus ist nur in AWL möglich, ggf. über Ansicht > <i>AWL Ansicht</i> in AWL ändern. • Der Baustein muss online geöffnet und darf nicht geschützt sein. • Der geöffnete Baustein darf im Editor nicht verändert worden sein.
Vorgehensweise zur Arbeit mit Haltepunkten	<ul style="list-style-type: none"> • Blenden Sie über Ansicht > <i>Haltepunktleiste</i> diese ein. • Setzen Sie Ihren Cursor auf die Anweisungszeile, in der ein Haltepunkt gesetzt werden soll. • Setzen Sie den Haltepunkt mit Test > <i>Haltepunkt setzen</i>. Die Anweisungszeile wird mit einem Kreisring markiert. • Zur Aktivierung des Haltepunkts gehen Sie auf Test > <i>Haltepunkt aktiv</i>. Der Kreisring wird zu einer Kreisfläche. • 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. • Nun können Sie mit Test > <i>Nächste Anweisung ausführen</i> schrittweise Ihren Programmcode durchfahren oder über Test > <i>Fortsetzen</i> Ihre Programmausführung bis zum nächsten Haltepunkt fortsetzen. • Mit Test > <i>(Alle) Haltepunkte löschen</i> können Sie (alle) Haltepunkte wieder löschen.
Verhalten im Betriebszustand HALT	<ul style="list-style-type: none"> • LED RUN blinkt, LED STOP leuchtet. • Die Bearbeitung des Codes ist angehalten. Alle Ablaufebenen werden nicht weiterbearbeitet. • Alle Zeiten werden eingefroren. • Echtzeituhr läuft weiter. • Ausgänge werden abgeschaltet, können zu Testzwecken aber freigegeben werden. • Passive CP-Kommunikation ist möglich.
	<p>Hinweis!</p> <p>Der Einsatz von Haltepunkten ist immer möglich. Eine Umschaltung in die Betriebsart Testbetrieb ist nicht erforderlich.</p> <p>Sobald Sie mehr als 3 Haltepunkte gesetzt haben, ist eine Einzelschritt-Bearbeitung nicht mehr möglich.</p>

**Funktions-
sicherheit**

Die CPU besitzt 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 (B efehls- A usgabe- S perre) wird gesetzt.
	zentrale digitale Ausgänge	Die Ausgänge werden auf 0V gesetzt.
	zentrale analoge Ausgänge	Die Spannungsversorgung für die Ausgabe-Kanäle wird abgeschaltet.
	dezentrale Ausgänge	Die Ausgänge werden auf 0V gesetzt.
STOP → RUN bzw. Netz-Ein	dezentrale Eingänge	Die Eingänge werden vom Slave konstant gelesen und die aktuellen Werte zur Verfügung gestellt.
	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.
	zentrale analoge Ausgänge	Das Verhalten der Ausgänge bei Neustart kann voreingestellt werden.
RUN	dezentrale Eingänge	Die Eingänge werden vom Slave konstant gelesen und die aktuellen Werte zur Verfügung gestellt.
	allgemein	Der Programmablauf ist zyklisch und damit vorhersehbar: PAE lesen → OB 1 → PAA schreiben.

PAE = Prozessabbild der Eingänge
PAA = Prozessabbild der Ausgänge

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 Siemens SIMATIC Manager



Hinweis!

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

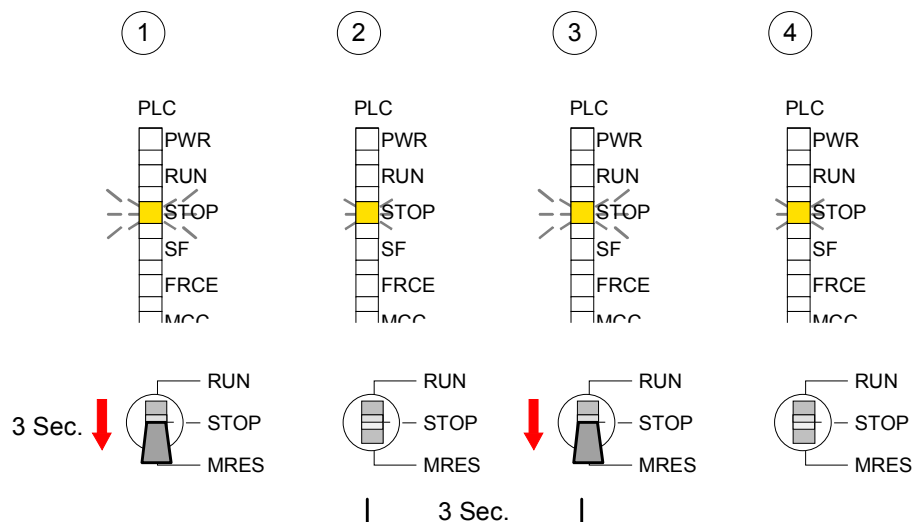
Voraussetzung

Ihre CPU muss sich im STOP-Zustand befinden. Stellen Sie hierzu den CPU-Betriebsartenschalter auf "STOP" → die STOP-LED leuchtet.

Urlöschen

- Bringen Sie den Betriebsartenschalter in Stellung MRES und halten Sie ihn ca. 3 Sekunden. → Die STOP-LED geht von Blinken über in Dauerlicht.
- 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).
- Das Urlöschen ist abgeschlossen, wenn die STOP-LED in Dauerlicht übergeht → Die STOP-LED leuchtet.

Die nachfolgende Abbildung zeigt nochmals die Vorgehensweise:



Automatisch nachladen	<p>Nun versucht die CPU ihr Anwenderprogramm von der Memory Card neu zu laden. → Die MCC-LED blinkt.</p> <p>Nach dem Nachladen erlischt die LED. Abhängig von der Einstellung des Betriebsartenschalters bleibt die CPU in STOP bzw. geht in RUN.</p>
Urlöschen über Siemens SIMATIC Manager	<p><i>Voraussetzung</i></p> <p>Ihre CPU muss sich im STOP-Zustand befinden.</p> <p>Mit dem Menübefehl Zielsystem > <i>Betriebszustand</i> bringen Sie Ihre CPU in STOP.</p> <p><i>Urlöschen</i></p> <p>Über den Menübefehl Zielsystem > <i>Urlöschen</i> fordern Sie das Urlöschen an.</p> <p>In dem Dialogfenster können Sie, wenn noch nicht geschehen, Ihre CPU in STOP bringen und das Urlöschen starten.</p> <p>Während des Urlöschvorgangs blinkt die STOP-LED.</p> <p>Geht die STOP-LED in Dauerlicht über, ist der Urlöschvorgang abgeschlossen.</p>
Automatisch nachladen	<p>Nun versucht die CPU ihr Anwenderprogramm von der Memory Card neu zu laden. → Die MCC-LED blinkt.</p> <p>Nach dem Nachladen erlischt die LED. Abhängig von der Einstellung des Betriebsartenschalters bleibt die CPU in STOP bzw. geht in RUN.</p>
Rücksetzen auf Werkseinstellung	<p>Die folgende Vorgehensweise löscht das interne RAM der CPU vollständig und bringt diese zurück in den Auslieferungszustand.</p> <p>Bitte beachten Sie, dass hierbei auch die MPI-Adresse defaultmäßig auf 2 zurückgestellt wird!</p> <ul style="list-style-type: none">• Drücken Sie den Reset-Schalter für ca. 30 Sekunden nach unten. Hierbei blinkt die ST-LED. Nach ein paar Sekunden wechselt die Anzeige zu statischem Licht. Zählen Sie die Anzahl des statischen Lichts, da die Anzeige jetzt wechselt zwischen statischem Licht und Blinken.• Nach dem 6. Mal statischem Licht 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.• Schalten Sie die Stromzufuhr aus und wieder an. <p>Nähere Informationen hierzu finden Sie unter "Rücksetzen auf Werkseinstellung" weiter unten.</p>

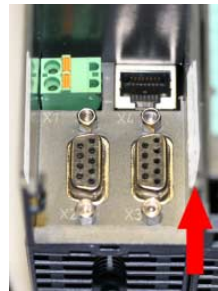
Firmwareupdate

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.

Sobald sich beim Hochlauf eine entsprechende pkg-Datei auf der MMC befindet und es sich bei Firmware um eine aktuellere Firmware handelt als in den Komponenten, werden alle der pkg-Datei zugeordneten Komponenten innerhalb der CPU mit der neuen Firmware beschrieben.



**Firmware package
and version**

Aktuelle Firmware auf ftp.vipa.de

Die 2 aktuellsten Firmwarestände finden Sie auf www.vipa.de im Service-Bereich und auf dem VIPA-ftp-Server unter [ftp.vipa.de/support/firmware](ftp://ftp.vipa.de/support/firmware).

Beispielsweise ist für das Firmwareupdate der CPU 312-5BE03 für den Ausgabestand 1 folgende Datei erforderlich: Px000074_Vxxx.zip



Achtung!

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 SC-CPU hat eine Web-Seite integriert, die auch Informationen zum Firmwarestand der Ein-/Ausgabe-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. Näheres hierzu finden Sie unter "Zugriff auf integrierte Web-Seite".

CPU-Firmwarestand über Baugruppenzustand ermitteln

Stellen Sie eine Online-Verbindung zu Ihrer CPU her. Zur Anzeige des Baugruppenzustands gehen Sie in Ihrem Siemens SIMATIC Manager auf **Zielsystem > Baugruppenzustand**.

Über das Register "Allgemein" gelangen Sie in das Fenster zur Anzeige des Hard- und Firmware-Ausgabestands.

Aus softwaretechnischen Gründen ergibt sich für die CPU 312SC von VIPA eine Änderung gegenüber der CPU 312C von Siemens:

Die Ausgabestände für Hard- und Firmware finden Sie unter "Best-Nr./Bezeichn." Die Versionsangabe unter "Ausgabestand" ist irrelevant.

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 AG01 10V3.2.8</td> <td>Firmware</td> <td>V2.0.0</td> </tr> </tbody> </table>			Best.-Nr./Bezeichn.	Komponente	Ausgabestand	6ES7 312-5BE03	Hardware	1	VIPA AG01 10V3.2.8	Firmware	V2.0.0
Best.-Nr./Bezeichn.	Komponente	Ausgabestand										
6ES7 312-5BE03	Hardware	1										
VIPA AG01 10V3.2.8	Firmware	V2.0.0										

Hardware:
Ausgabestand
Subversion

Firmware:
Version

irrelevant



Hinweis!

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.de.
- Klicken Sie auf Service > Download > Firmware Updates.
- Klicken Sie auf "Firmware für System 300S".
- Wählen Sie die entsprechenden Baugruppen (CPU, DPM, CP...) aus und laden Sie die Firmware Px.....zip auf Ihren PC.
- Entpacken Sie die zip-Datei und kopieren Sie die extrahierte Datei auf Ihre MMC.
- Übertragen Sie auf diese Weise alle erforderlichen Firmware-Dateien auf Ihre MMC.

**Voraussetzungen
für ftp-Zugriff**

Zur Ansicht von ftp-Seiten in Ihrem Web-Browser sind ggf. folgende Einstellungen vorzunehmen:

Internet Explorer

ftp-Zugriff nur möglich ab Version 5.5

Extras > *Internetoptionen*, Register "Erweitert" im Bereich Browsing:

- aktivieren: "Ordneransicht für ftp-Sites aktivieren"
- aktivieren: "Passives ftp verwenden ..."

Netscape

ftp-Zugriff nur möglich ab Version 6.0

Es sind keine zusätzlichen Einstellungen erforderlich.

Sollte es immer noch Probleme mit dem ftp-Zugriff geben, fragen Sie Ihren Systemverwalter.

**Achtung!**

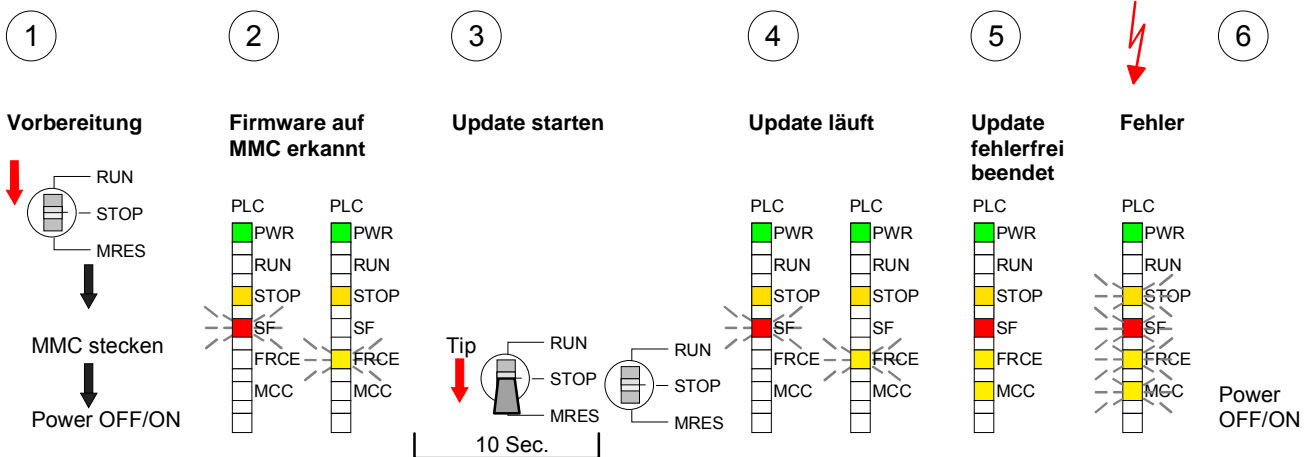
Beim Firmwareupdate wird automatisch ein Urlöschen durchgeführt. Sollte sich Ihr Programm nur im Ladespeicher der CPU befinden, so wird es hierbei gelöscht! Sichern Sie Ihr Programm, bevor Sie ein Firmwareupdate durchführen! Auch sollten Sie nach dem Firmwareupdate ein "Rücksetzen auf Werkseinstellung"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

Firmware von MMC in CPU übertragen

1. Bringen Sie den RUN-STOP-Schalter 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 FRCE 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 RUN/STOP-Schalter kurz nach MRES tippen und dann den Schalter in der STOP-Position belassen.
4. Während des Update-Vorgangs blinken die LEDs SF und FRCE abwechselnd und die MCC-LED leuchtet. Dieser Vorgang kann mehrere Minuten dauern.
5. Das Update ist fehlerfrei beendet, wenn die LEDs PWR, STOP, SF, FRCE und MCC 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 FRCE. 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.



Rücksetzen auf Werkseinstellung

Vorgehensweise

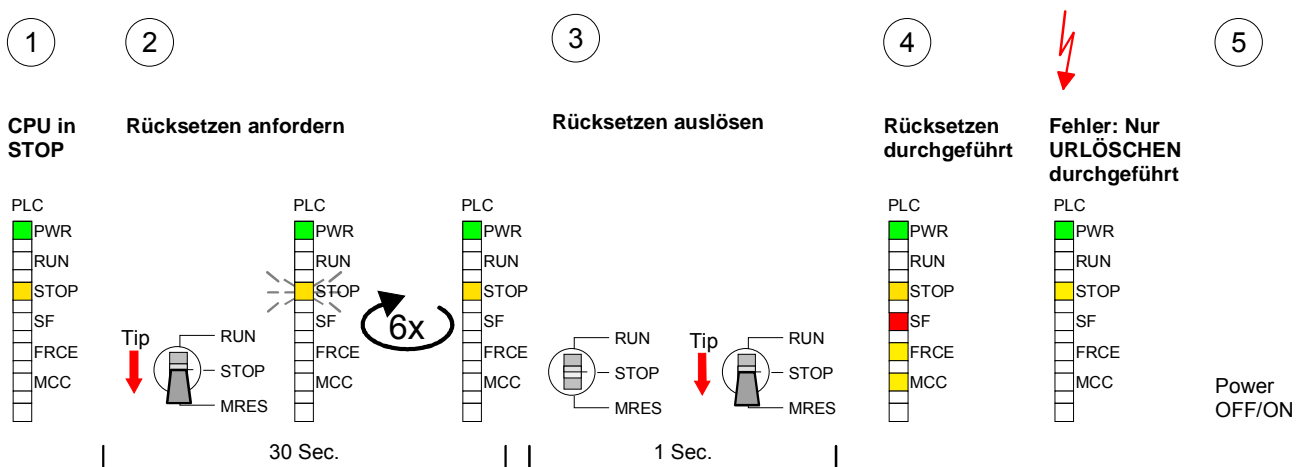
Die folgende Vorgehensweise löscht das interne RAM der CPU vollständig und bringt diese zurück in den Auslieferungszustand.

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-Cmd FACTORY_RESET ausführen. Nähere Informationen hierzu finden Sie unter "MMC-Cmd - Autobefehle".

1. Bringen Sie die CPU in STOP.
2. Drücken Sie den Betriebsarten-Schalter 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.
3. Nach dem 6. Mal Leuchten der STOP-LED lassen Sie den Reset-Schalter wieder los, um ihn nochmals kurzzeitig nach unten zu drücken.
4. Zur Bestätigung des Rücksetzvorgangs leuchten die LEDs PWR, STOP, SF, FRCE und MCC. Leuchtet diese nicht, wurde nur Urlöschen 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.
5. Nach dem Rücksetzen auf Werkseinstellung ist die Spannungsversorgung aus- und wieder einzuschalten.

Die nachfolgende Abbildung soll die Vorgehensweise nochmals verdeutlichen:



Hinweis!

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

Speichererweiterung mit MCC

Übersicht



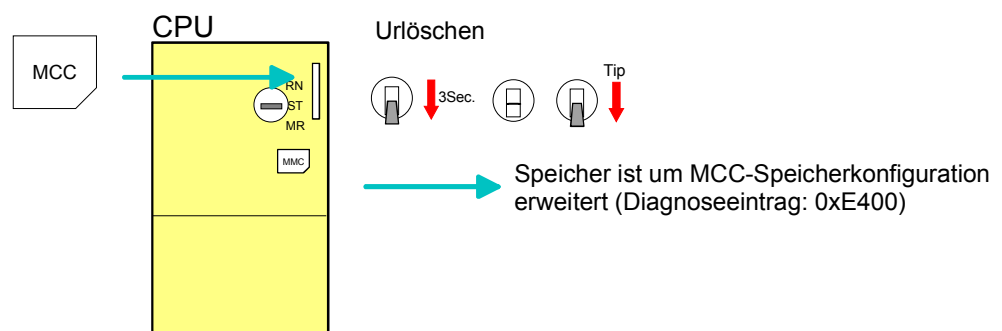
Bei der SC-CPU haben Sie die Möglichkeit den Arbeitsspeicher Ihrer CPU zu erweitern.

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 im Siemens SIMATIC Manager über den *Baugruppenzustand* unter "Speicher" ermitteln.



Achtung!

Bitte beachten Sie, dass, sobald Sie eine Speichererweiterung auf Ihrer CPU durchgeführt haben, die MCC gesteckt bleiben muss. Ansonsten geht die CPU nach 72h 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 72h geht die CPU in STOP. Hier ist ein Anlauf erst wieder möglich nach Stecken der MCC oder nach Urlöschen.

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.

Erweiterter Know-how-Schutz

Übersicht

Neben dem "Standard" Know-how-Schutz besitzt die CPU von VIPA einen "erweiterten" Know-how-Schutz, der einen sicheren Baustein-Schutz vor Zugriffen 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.

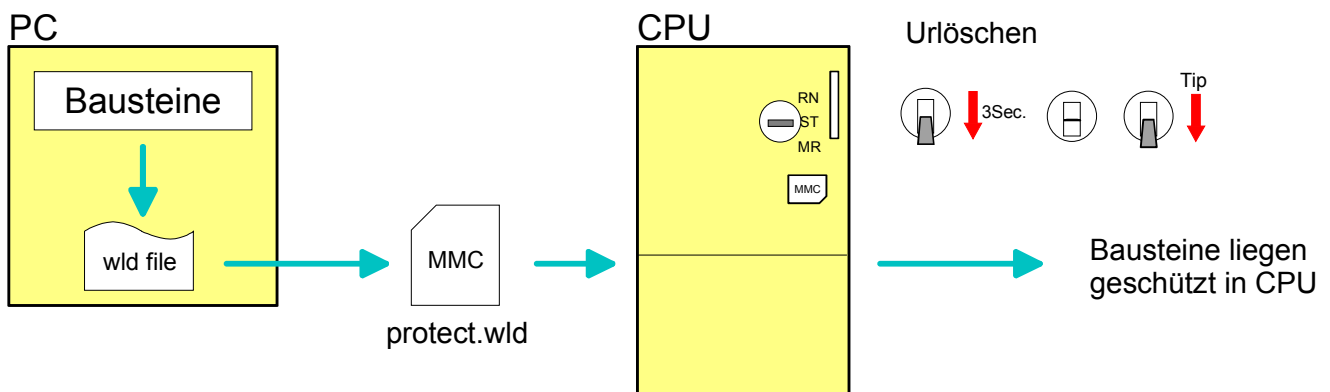
Erweiterter Schutz

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 Source bleibt in der CPU und somit vor dem Zugriff Dritter geschützt.



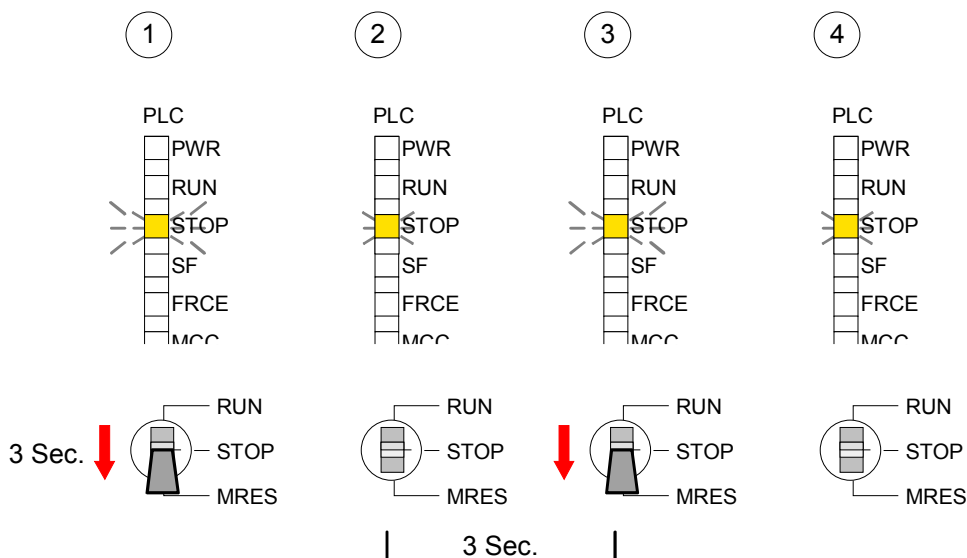
Bausteine mit protect.wld schützen

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 Urlöschen 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 "Baustein-hüllen" 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.

MMC-Cmd - Autobefehle

Übersicht

Eine *Kommando-Datei* kann auf einer MMC automatisch ausgeführt werden, sobald die MMC gesteckt ist und die CPU sich im STOP befindet. Solange die MMC gesteckt ist wird die Kommando-Datei bei CPU-STOP einmalig bis zum nächsten PowerON ausgeführt.

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

Nachfolgend finden Sie eine Übersicht der 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 <i>CMD_START</i> stehen.	0xE801
	Fehlt <i>CMD_START</i> 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 "Urlöschen 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
SAVE_PROJECT	Speichert das Anwenderprojekt (Bausteine und Hardwarekonfiguration) auf der MMC als "s7prog.wld". Falls bereits eine Datei mit dem Namen "s7prog.wld" existiert, wird diese in "s7prog.old" umbenannt.	0xE806
FACTORY_RESET	Führt "Rücksetzen auf Werkseinstellung" durch.	0xE807
DIAGBUF	Speichert den Diagnosebuffer 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 xxx.xxx.xxx.xxx einzugeben. Wird kein Gateway verwendet, tragen Sie die IP-Adresse als Gateway ein.	0xE80E
CMD_END	In der letzten Zeile muss <i>CMD_END</i> 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)
DIAGBUF	Diagnosebuffer 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)
SET_NETWORK 172.16.129.210,255.255.224.0,172.16.129.210	IP-Parameter (0xE80E)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
WAIT1SECOND	Wartet ca. 1 Sekunde (0xE803)
WEBPAGE	Web-Seite als "webpage.htm" speichern (0xE804)
DIAGBUF	Diagnosebuffer 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.



Hinweis!

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.

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, die ausschließlich in Form einer Ereignis-ID angezeigt werden.

Mit dem MMC-Cmd DIAGBUF wird der aktuelle Inhalt des Diagnosepuffers auf MMC gespeichert. Nähere Informationen hierzu finden Sie unter "MMC-Cmd - Autobefehle".



Hinweis!

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 Diagnoseeinträge

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

Nr.	Uhrzeit	Datum	Ereignis
8	13:18:15:203	28.03.06	Automatische Neustart (Warmstart)-Anforderung
9	13:18:15:203	28.03.06	Betriebszustandsübergang von STOP nach ANLAUF
10	13:18:11:370	28.03.06	STOP durch Stopschalter-Bedienung
11	02:39:35:498	28.03.06	Ereignis-ID: 16# E0CC
12	02:39:10:768	28.03.06	Betriebszustandsübergang von ANLAUF nach RUN
13	02:39:10:768	28.03.06	Automatische Neustart (Warmstart)
14	02:39:10:768	28.03.06	Betriebszustandsübergang von STOP nach ANLAUF
15	02:38:55:603	28.03.06	Ereignis-ID: 16# E0CC

Details zum Ereignis: 1 von 30 Ereignis-ID: 16# 4302

Betriebszustandsübergang von ANLAUF nach RUN
 Anlaufinformation:
 - Soll-/Istdifferenz vorhanden
 - Uhr für Zeitstempel bei letztem NETZ-EIN gepuffert
 - Einprozessorbetrieb
 Aktuelle/letzte durchgeführte Anlaufart:

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

Auf der Folgeseite finden Sie eine Übersicht der VIPA-spezifischen Ereignis-IDs.

Übersicht der Ereignis-ID

Ereignis-ID	Bedeutung
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 Slavekonfiguration
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 Masterperipherie
0xE019	Fehler bei Erkennung des Standard Rückwandbus Systems
0xE01A	Fehler bei Erkennung der Betriebsart (8 / 9 Bit)
0xE01B	Fehler - maximale Anzahl steckbarer Baugruppen überschritten
0xE030	Fehler vom Standard-Bus
0xE0B0	Speed7 kann nicht mehr gestoppt werden (evtl. undefinierter BCD-Wert bei Timer)
0xE0C0	Nicht genug Speicherplatz im Arbeitsspeicher für Codebaustein (Baustein zu groß)
0xE0CC	Kommunikationsfehler MPI / Seriell
0xE0CD	Fehler bei DPV1 Auftragsverwaltung
0xE0CE	Fehler: Timeout beim Senden der i-Slave Diagnose
0xE100	MMC-Zugriffsfehler
0xE101	MMC-Fehler Filesystem
0xE102	MMC-Fehler FAT
0xE104	MMC Fehler beim Speichern
0xE200	MMC schreiben beendet (Copy Ram2Rom)

Fortsetzung ...

... Fortsetzung

Ereignis-ID	Bedeutung
0xE210	MMC Lesen beendet (Nachladen nach Urlöschen)
0xE21F	MMC Lesen: Fehler beim Nachladen (nach Urlöschen), Lesefehler, Speicher voll
0xE400	Speichererweiterungs-MCC wurde gesteckt
0xE401	Speichererweiterungs-MCC wurde gezogen
0xE801	MMC-Cmd: CMD_START erkannt und erfolgreich ausgeführt
0xE802	MMC-Cmd: CMD_END erkannt und erfolgreich ausgeführt
0xE803	MMC-Cmd: WAIT1SECOND erkannt und erfolgreich ausgeführt
0xE804	MMC-Cmd: WEBPAGE erkannt und erfolgreich ausgeführt
0xE805	MMC-Cmd: LOAD_PROJECT erkannt und erfolgreich ausgeführt
0xE806	MMC-Cmd: SAVE_PROJECT erkannt und erfolgreich ausgeführt
0xE807	MMC-Cmd: FACTORY_RESET erkannt und erfolgreich ausgeführt
0xE80B	MMC-Cmd: DIAGBUF erkannt und erfolgreich ausgeführt
0xE80E	MMC-Cmd: SET_NETWORK erkannt und erfolgreich ausgeführt
0xE8FB	MMC-Cmd: Fehler: Initialisierung des Ethernet-PG/OP-Kanals mittels SET_NETWORK fehlerhaft.
0xE8FC	MMC-Cmd: Fehler: In SET_NETWORK wurden nicht alle IP-Parameter angegeben.
0xE8FE	MMC-Cmd: Fehler: CMD_START nicht gefunden
0xE8FF	MMC-Cmd: Fehler: Fehler beim Lesen des CMD-Files (MMC-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
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!
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

Fortsetzung ...

... Fortsetzung

Ereignis-ID	Bedeutung
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
0xEA24	Fehler - Projektierung RS485-Schnittstelle X2/X3: Schnittstelle/Protokoll ist nicht vorhanden, die Defaulteinstellungen werden verwendet. Zinfo2: Projektierter Wert für X2 Zinfo2: 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!
0xEA98	Timeout beim Warten, dass ein SBUS-Modul (Server) rebootet hat
0xEA99	Fehler beim File-Lesen über SBUS
0xEE00	Interner Fehler - Kontaktieren Sie bitte die VIPA-Hotline!

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.



Hinweis!

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

Die Statusbearbeitung kann durch Sprungbefehle oder Zeit- und Prozessalarmlen unterbrochen werden. 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 Prozessalarmen 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 "?".

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.

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.

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 SC-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.

Teil 5 Einsatz E/A-Peripherie

Überblick

In diesem Kapitel finden Sie alle Informationen, die zum Einsatz der Ein-/Ausgabe-Peripherie der CPU 312SC erforderlich sind. Beschrieben sind Funktionalität, Projektierung und Diagnose des Digital-Teils.

Inhalt

Thema	Seite
Teil 5 Einsatz E/A-Peripherie	5-1
Übersicht	5-2
Ein-/Ausgabe-Bereiche CPU 312SC	5-3
Adressbelegung	5-5
Digital-Teil	5-6
Digital-Teil - Parametrierung	5-8
Zähler - Schnelleinstieg	5-9
Zähler - Steuern	5-14
Zähler - Funktionen	5-18
Zähler - Zusatzfunktionen.....	5-24
Zähler - Diagnose und Alarm.....	5-31

Übersicht

Hardware	Bei der CPU 312SC sind die Anschlüsse für digitale Ein-/ Ausgabe und Technologische Funktionen in einem 2-fach breiten Gehäuse untergebracht.
Projektierung Parametrierung	<p>Die Projektierung erfolgt im Siemens SIMATIC Manager als CPU 312C von Siemens (6ES7 312-5BE03-0AB0 V2.0).</p> <p>Hierbei parametrieren Sie Ihre CPU über den "Eigenschaften"-Dialog der Siemens CPU 312C.</p> <p>Für die Parametrierung der digitalen Ein-/Ausgabeperipherie und der Technologischen Funktionen sind die entsprechenden Submodule der CPU 312C zu verwenden.</p>
E/A- Peripherie	<p>Die integrierten Ein-/Ausgänge der CPU 312SC können für Technologische Funktionen bzw. als Standardperipherie genutzt werden.</p> <p>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.</p>
Technologische Funktionen	<p>Für 2 Kanäle sind Technologische Funktionen parametrierbar. Die Parametrierung des entsprechenden Kanals erfolgt im Hardware-Konfigurator über das <i>Zähler</i>-Submodul der CPU 312C.</p> <p>Folgende Technologische Funktionen stehen Ihnen zur Verfügung:</p> <ul style="list-style-type: none">• Endlos Zählen• Einmalig Zählen• Periodisch Zählen <p>Die Steuerung der entsprechenden Zähler-Betriebsart erfolgt aus dem Anwenderprogramm über den SFB COUNT (SFB 47).</p>

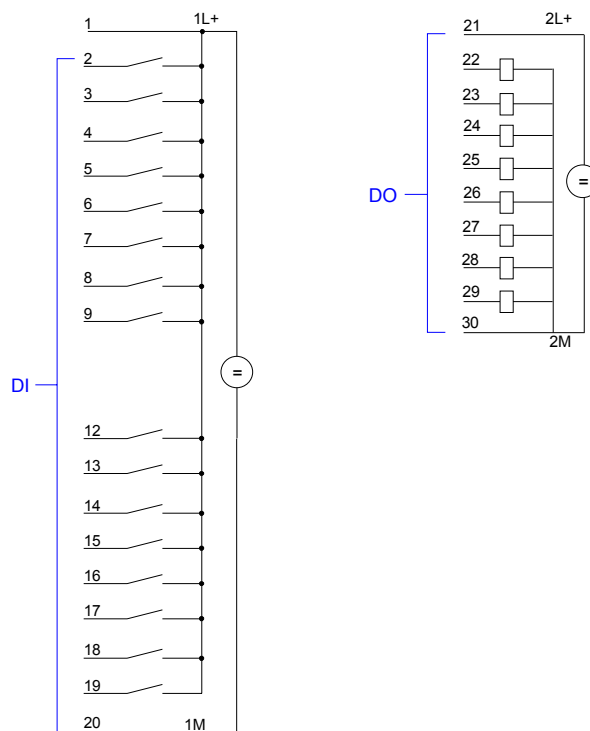
Ein-/Ausgabe-Bereiche CPU 312SC

Übersicht CPU 312SC

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

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

Jeder der digitalen Ein- bzw. Ausgänge zeigt seinen Zustand über eine LED an. Über die Parametrierung können Sie jedem digitalen Eingang Alarm-Eigenschaften zuweisen. Zusätzlich lassen sich die digitalen Eingänge als Zähler parametrieren.



Achtung!

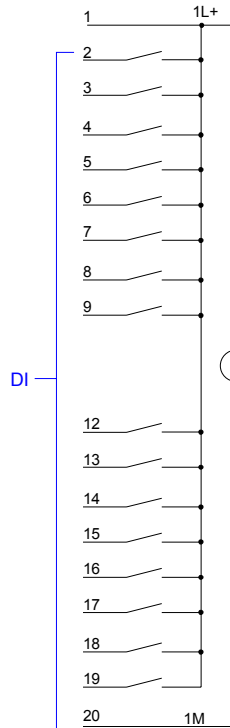
Die an einem digitalen Ausgabe-Kanal anliegende Spannung muss immer \leq der über L+ anliegenden Versorgungsspannung sein.

CPU 312SC: Steckerbelegung und Statusanzeige

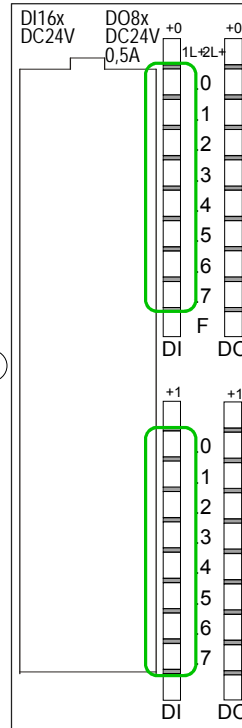
Pin Belegung

1	Versorgungsspg. +DC 24V
2	E+0.0 / Kanal 0 (A)/Impuls
3	E+0.1 / Kanal 0 (B)/Richtung
4	E+0.2 / Kanal 0 Hardwareretor
5	E+0.3 / Kanal 1 (A)/Impuls
6	E+0.4 / Kanal 1 (B)/Richtung
7	E+0.5 / Kanal 1 Hardwareretor
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

Anschluss



LEDs



DI:
L1+ LED (grün)
Versorgungsspannung für DI liegt an

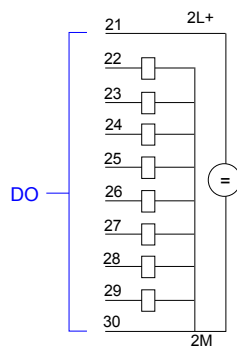
.07 LEDs (grün)
E+0.0 bis E+0.7 bzw. E+1.0 bis E+1.7 ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert

CPU 312SC: Steckerbelegung und Statusanzeige

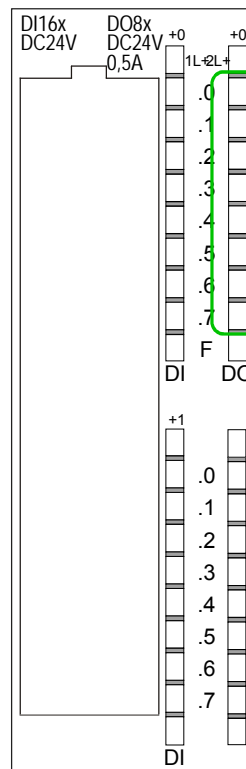
Pin Belegung

21	Versorgungsspg. +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	nicht belegt
32	nicht belegt
33	nicht belegt
34	nicht belegt
35	nicht belegt
36	nicht belegt
37	nicht belegt
38	nicht belegt
39	nicht belegt
40	nicht belegt

Anschluss



LEDs



DO:
L2+ LED (grün)
Versorgungsspannung für DO liegt an

07 LEDs (grün)
A+0.0 bis A+0.7 leuchtet bei aktiviertem Ausgang

F LED (rot)
Fehler bei Überlast oder Kurzschluss

Adressbelegung

Eingabebereich

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

Ausgabebereich

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

Digital-Teil

Digitaler Teil CPU 312SC

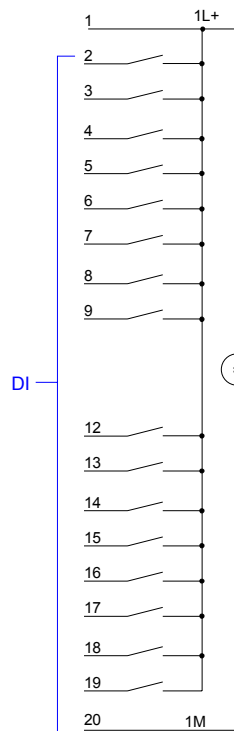
Der digitale Bereich besteht aus 16 Eingabe-, 8 Ausgabe-Kanälen und 2 Kanälen für Technologische Funktionen. Jeder der digitalen Ein- bzw. Ausgänge zeigt seinen Zustand über eine LED an. Über die Parametrierung können Sie den Eingängen E+0.0 bis E+1.7 Alarm-Eigenschaften zuweisen.

CPU 312SC: Steckerbelegung und Statusanzeige

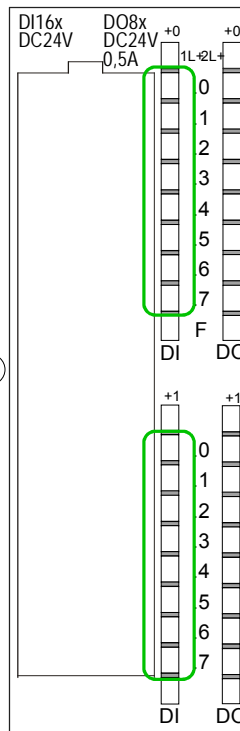
Pin Belegung

1	Versorgungsspg. +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

Anschluss



LEDs



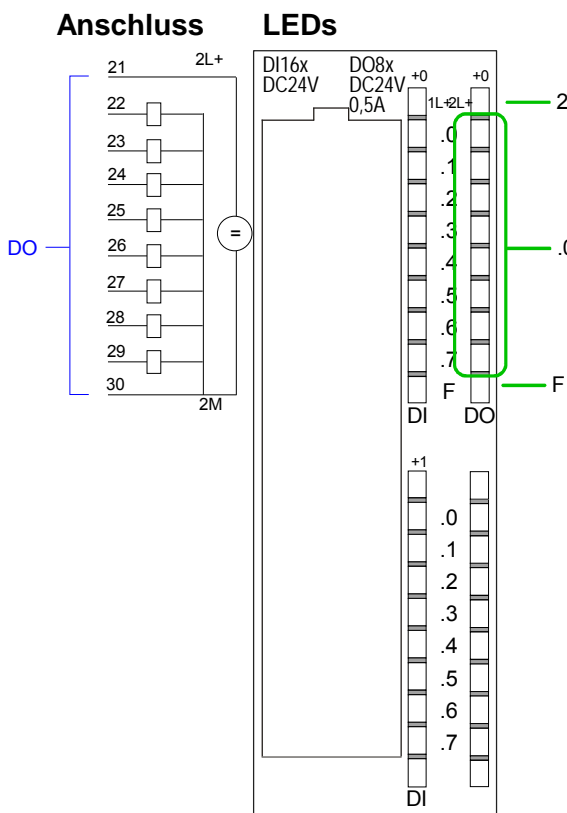
DI:
L1+ LED (grün)
Versorgungsspannung für DI liegt an

.07 LEDs (grün)
E+0.0 bis E+0.7 bzw. E+1.0 bis E+1.7 ab ca. 15V wird das Signal "1" am Eingang erkannt und die entsprechende LED angesteuert

CPU 312SC: Steckerbelegung und Statusanzeige

Pin Belegung

- 21 Versorgungsspg. +DC 24V
- 22 A+0.0 / Kanal 0 Ausgang
- 23 A+0.1 / Kanal 1 Ausgang
- 24 A+0.2 / Kanal 2 Ausgang
- 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 nicht belegt
- 32 nicht belegt
- 33 nicht belegt
- 34 nicht belegt
- 35 nicht belegt
- 36 nicht belegt
- 37 nicht belegt
- 38 nicht belegt
- 39 nicht belegt
- 40 nicht belegt



- DO: L2+ LED (grün)
Versorgungsspannung für DO liegt an
- 07 LEDs (grün)
A+0.0 bis A+0.7 leuchtet bei aktiviertem Ausgang
- F LED (rot)
Fehler bei Überlast oder Kurzschluss

Zugriff auf den Digital-Teil

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

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

Digital-Teil - Parametrierung

Parameterbereich	<p>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.</p>
Allgemein	<p>Hier finden Sie die Kurzbezeichnung der digitalen Peripherie. Unter <i>Kommentar</i> können Sie den Einsatzzweck kommentieren.</p>
Adressen	<p>In diesem Register legen Sie die Anfangsadresse der Ein- und Ausgabe-Peripherie fest.</p>
Eingänge	<p>Hier haben Sie folgende Einstellmöglichkeiten:</p> <ul style="list-style-type: none">• Prozessalarm• Eingangsverzögerung <p>Für die digitalen Ausgabe-Kanäle gibt es keine Parametriermöglichkeiten.</p>
Prozessalarm	<p>Ein Prozessalarm kann wahlweise bei steigender bzw. fallender Flanke an einem Eingang ausgelöst werden.</p> <p>Diagnosealarm wird nur in Verbindung mit Prozessalarm verloren unterstützt.</p> <p>Wählen Sie mit den Pfeiltasten den Eingang an und aktivieren Sie die gewünschten Prozessalarme.</p>
Eingangsverzögerung	<p>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.</p>

Zähler - Schnelleinstieg

Übersicht

Die CPU 312SC 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

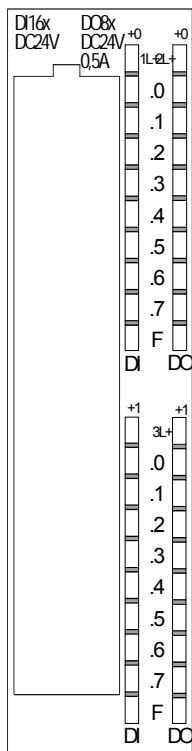
Für die CPU 312SC beträgt die maximale Frequenz unabhängig von der Anzahl der aktivierten Zähler 10kHz.

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

Steckerbelegung

Pin Belegung

1	Versorgungssp. +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 DI



Pin Belegung

21	Versorgungssp. +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 DO
31	nicht belegt
32	nicht belegt
33	nicht belegt
34	nicht belegt
35	nicht belegt
36	nicht belegt
37	nicht belegt
38	nicht belegt
39	nicht belegt
40	nicht belegt

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

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.

**Hinweis!**

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_x-Eingang am Modul

Schließen (deaktivieren): Flankenwechsel 1-0 am Hardwaretor_x-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_x (A)*
Impulseingang für Zählsignal bzw. die Spur A eines Gebers. Hierbei können Sie Geber mit 1-, 2- oder 4-facher Auswertung anschließen.
- Kanal_x (B)*
Richtungssignal bzw. die Spur B des Gebers. Über die Parametrierung können Sie das Richtungssignal invertieren.
- Hardwaretor_x*
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_x*
Mit einer positiven Flanke an Latch_x 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.

- Parameter-übersicht** 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 CPU 312SC wird der Diagnosealarm der digitalen Ein-/Ausgabe-Peripherie nur in Verbindung mit "Prozessalarm verloren" unterstützt.

Zählen

Parameter	Beschreibung	Wertebereich	Vorbelegung
Haupt-zählrichtung	<ul style="list-style-type: none"> • <i>Keine</i>: Keine Einschränkung des Zählbereiches • <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. • <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. 	<ul style="list-style-type: none"> • Keine • Vorwärts • Rückwärts (nicht bei Endlos Zählen) 	Keine
Endwert/ Startwert	<i>Endwert</i> bei Hauptzählrichtung vorwärts. <i>Startwert</i> bei Hauptzählrichtung rückwärts.	2...2147483647 ($2^{31}-1$)	2147483647 ($2^{31}-1$)
Torfunktion	<ul style="list-style-type: none"> • <i>Zählvorgang abbrechen</i>: Der Zählvorgang beginnt nach dem Schließen des Tors und erneutem Torstart wieder ab dem Ladewert. • <i>Zählvorgang unterbrechen</i>: Der Zählvorgang wird nach dem Schließen des Tors und erneutem Torstart beim letzten aktuellen Zählerstand fortgesetzt. 	<ul style="list-style-type: none"> • Zählvorgang abbrechen • Zählvorgang unterbrechen 	Zählvorgang abbrechen
Vergleichswert	Der Zählwert wird mit dem Vergleichswert verglichen. Siehe hierzu auch Parameter "Verhalten des Ausgangs": <ul style="list-style-type: none"> • Keine Hauptzählrichtung • Hauptzählrichtung vorwärts • Hauptzählrichtung rückwärts 	-2^{31} bis $+2^{31}-1$ -2^{31} bis Endwert-1 1 bis $+2^{31}-1$	0
Hysterese	Die Hysterese dient zur Vermeidung von häufigen Schaltvorgängen des Ausgangs, wenn der Zählwert im Bereich des Vergleichswertes liegt. 0 und 1 bedeuten: Hysterese abgeschaltet	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

Fortsetzung ...

... Fortsetzung

Parameter	Beschreibung	Wertebereich	Vorbelegung
max. Frequenz: Latch	Die maximale Frequenz des Latch-Signals können sie in festen Stufen einstellen.	10, 5, 2, 1kHz	10kHz
Signalauswertung	Am Eingang wird das Zählsignal und das Richtungssignal angeschlossen. Am Eingang wird ein Drehgeber angeschlossen (Einfach-, Zweifach- oder Vierfachauswertung).	<ul style="list-style-type: none"> • Impuls/Richtung • Drehgeber einfach • Drehgeber zweifach • Drehgeber vierfach 	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"> • aktiviert • deaktiviert 	deaktiviert
Zählrichtung invertiert	Im aktivierten Zustand ist das Eingangssignal "Richtung" invertiert.	<ul style="list-style-type: none"> • aktiviert • deaktiviert 	deaktiviert
Verhalten des Ausgangs	Abhängig von diesem Parameter wird der Ausgang und das Statusbit "Vergleicher" (STS_CMP) gesetzt.	<ul style="list-style-type: none"> • Kein Vergleich • Zählerwert \geq Vergleichswert • Zählerwert \leq Vergleichswert • Impuls bei Vergleichswert 	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"> • aktiviert • deaktiviert 	
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"> • aktiviert • deaktiviert 	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"> • aktiviert • deaktiviert 	deaktiviert
Prozessalarm: Überlauf	Im aktivierten Zustand wird beim Überlauf (Überschreiten der oberen Zählgrenze) ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> • aktiviert • deaktiviert 	deaktiviert
Prozessalarm: Unterlauf	Im Aktivierten Zustand wird beim Unterlauf (Unterschreiten der unteren Zählgrenze) ein Prozessalarm erzeugt.	<ul style="list-style-type: none"> • aktiviert • deaktiviert 	deaktiviert

Zähler - Steuern

Übersicht

Die Steuerung des entsprechenden Zählers erfolgt aus dem Anwenderprogramm über den SFB COUNT (SFB 47). 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 SFB 47

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_GATE	INPUT	BOOL	4.0	FALSE	Softwaretor freigegeben
CTRL_DO	INPUT	BOOL	4.1	FALSE	Ausgang DO freigegeben False: Standard Digitaler Ausgang
SET_DO	INPUT	BOOL	4.2	FALSE	Parameter wird nicht ausgewertet
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 Aufträge
STS_GATE	OUTPUT	BOOL	12.0	FALSE	Status internes Tor
STS_STRT	OUTPUT	BOOL	12.1	FALSE	Status Hardwaretor (wird nur dann aktualisiert, wenn zuvor "HW-Tor" in Hardware-Konfiguration aktiviert wurde)
STS_LTCH	OUTPUT	BOOL	12.2	FALSE	Status Latch-Eingang
STS_DO	OUTPUT	BOOL	12.3	FALSE	Status Ausgang
STS_C_DN	OUTPUT	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	OUTPUT	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	OUTPUT	DINT	14.0	0	Aktueller Zählwert
LATCHVAL	OUTPUT	DINT	18.0	0	Aktueller Latchwert
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)	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 Vergleich ^{*)} 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). Dieser Parameter wird nur dann aktualisiert, wenn in der Hardware-Konfiguration unter "Verhalten des Ausganges" ein Vergleichswert eingestellt ist.
RES04	BOOL	26.4	FALSE	reserviert
STS_OFLW	BOOL	26.5	FALSE	Status Überlauf - wird ausschließlich bei Bereichsüberlauf gesetzt ^{*)}
STS_UFLW	BOOL	26.6	FALSE	Status Unterlauf - wird ausschließlich bei Bereichsunterlauf gesetzt ^{*)}
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



Hinweis!

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 Auftrags-schnittstelle

Zum Beschreiben bzw. Auslesen der Zählregister steht ihnen mit dem SFB 47 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	Daten-typ	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 (siehe Tabelle Folgeseite)

- 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	Daten-typ	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

- Mit JOB_DONE = TRUE kann ein neuer Auftrag gestartet werden.
- 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 ($-2^{31}+1$) bis +2147483646 ($2^{31}-2$)
Ladewert schreiben	-2147483647 ($-2^{31}+1$) bis +2147483646 ($2^{31}-2$)
Vergleichswert schreiben	-2147483648 (-2^{31}) bis +2147483647 ($2^{31}-1$)
Hysterese schreiben	0 bis 255
Impulsdauer schreiben*	0 bis 510ms

Einmalig/Periodisch Zählen, keine Hauptzählrichtung:

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

Einmalig/Periodisch Zählen, Hauptzählrichtung vorwärts:

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

Einmalig/Periodisch Zählen, Hauptzählrichtung rückwärts:

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

Zähler - Funktionen

- Parametriervorgang**
- Starten Sie den Siemens SIMATIC Manager mit Ihrem Projekt und öffnen Sie den Hardware-Konfigurator.
 - Platzieren Sie eine Profilschiene.
 - Projektieren Sie auf Steckplatz 2 die entsprechende Siemens CPU CPU 31xC.
 - Doppelklicken Sie auf das *Zähler*-Submodul Ihrer CPU. Sie gelangen in das Dialogfeld "Eigenschaften".
 - Sobald Sie für den entsprechenden Kanal die gewünschte Betriebsart einstellen, wird ein Dialogfenster für diese Zähler-Betriebsart mit Defaultwerten eingerichtet und angezeigt.
 - Führen Sie die gewünschten Parametrierungen durch.
 - Speichern Sie Ihr Projekt mit **Station** > *Speichern und übersetzen*.
 - Transferieren Sie Ihr Projekt in Ihre CPU.

Ladewert, Endwert Über die Parametrierung haben Sie die Möglichkeit für jeden Zähler eine Hauptzählrichtung anzugeben. Ist "keine" oder "endlos" angewählt, steht Ihnen der gesamte Zählbereich zur Verfügung:

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

Ansonsten können Sie diesen Bereich durch Angabe eines Startwerts als *Ladewert* und eines *Endwerts* nach unten und oben begrenzen.

Hauptzählrichtung *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.

Bitte beachten Sie, dass Sie einen Ladewert ausschließlich über die Auftragschnittstelle des Zählers vorgeben können.

Hauptzählrichtung rückwärts

Einschränkung des Zählbereiches nach unten. Der Zähler zählt vom parametrierten Start- bzw. Ladewert in negativer Richtung bis 0 oder zum parametrierten Endwert+1 und springt dann mit dem darauffolgenden Geberimpuls wieder auf den Startwert.

Bitte beachten Sie, dass Sie einen Endwert ausschließlich über die Auftragschnittstelle des Zählers vorgeben können.

Endlos Zählen

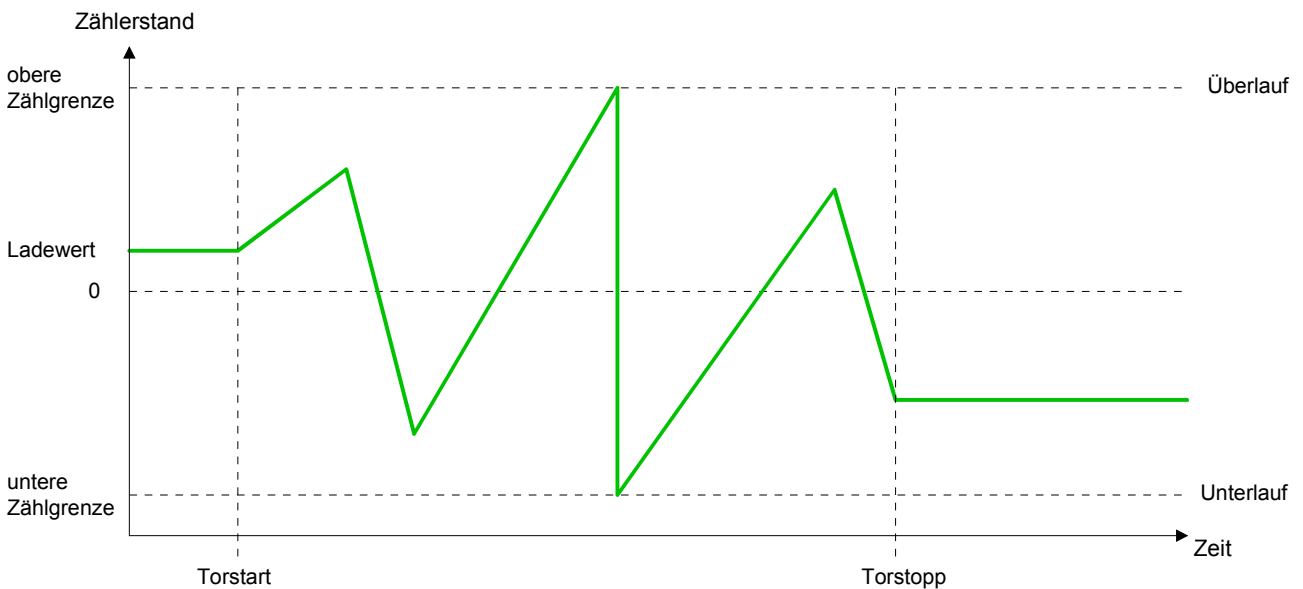
In dieser Betriebsart zählt der Zähler ab 0 bzw. ab dem Ladewert.

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 er zählt von dort weiter.

Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.

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



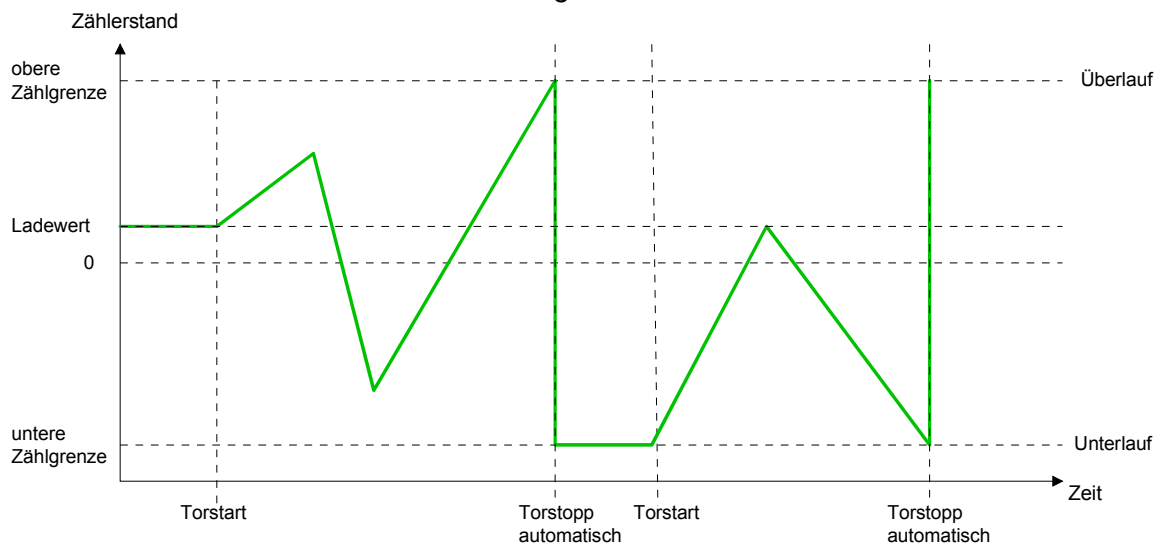
Einmalig Zählen

Keine Hauptzählrichtung

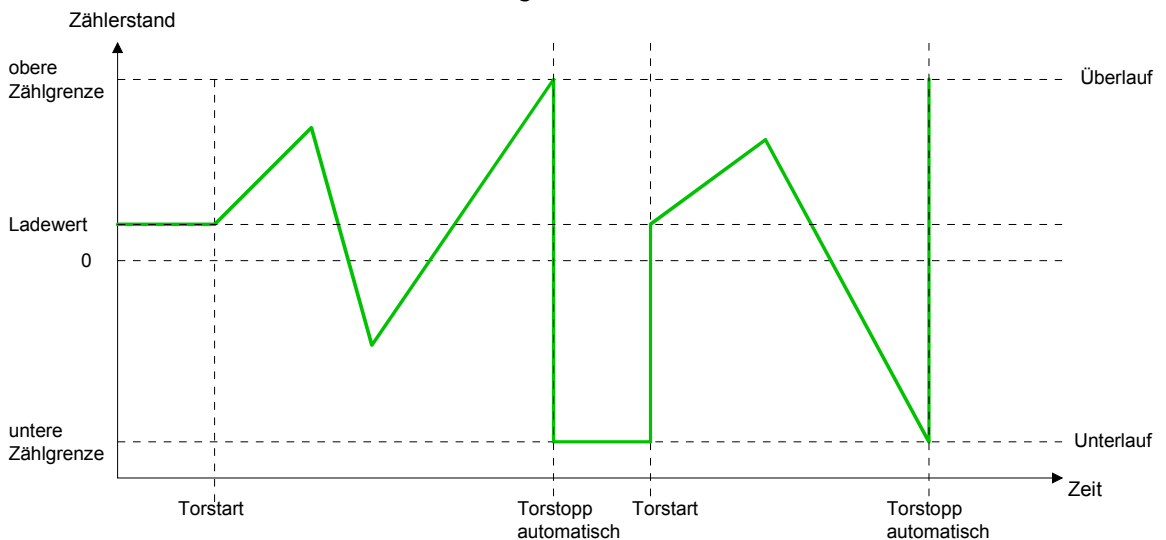
- Der Zähler zählt ab dem Ladewert einmalig.
- Es wird vorwärts oder rückwärts gezählt.
- Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.
- Bei Über- oder Unterlauf an den Zählgrenzen springt der Zähler auf die jeweils andere Zählgrenze und das Tor wird automatisch geschlossen.
- Zum erneuten Start des Zählvorgangs müssen Sie eine Flanke 0-1 des Tors erzeugen.
- Bei unterbrechender Torsteuerung wird der Zählvorgang beim aktuellen Zählstand fortgesetzt.
- Bei abbrechender Torsteuerung beginnt der Zähler ab dem Ladewert.

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

Unterbrechende Torsteuerung:



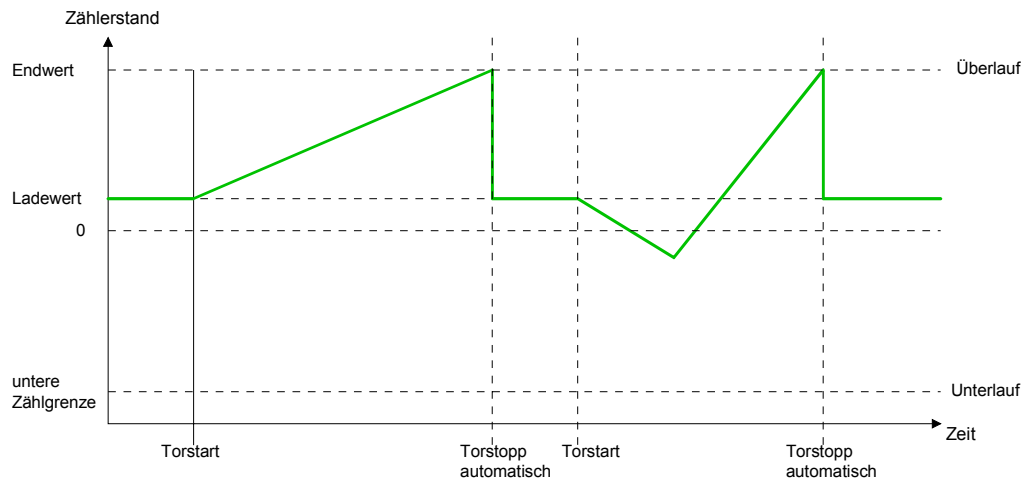
Abbrechende Torsteuerung:



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 Tor wird automatisch geschlossen.
- Zum erneuten Start des Zählvorgangs müssen Sie eine Flanke 0-1 der Torsteuerung erzeugen. Der Zähler beginnt ab dem Ladewert.

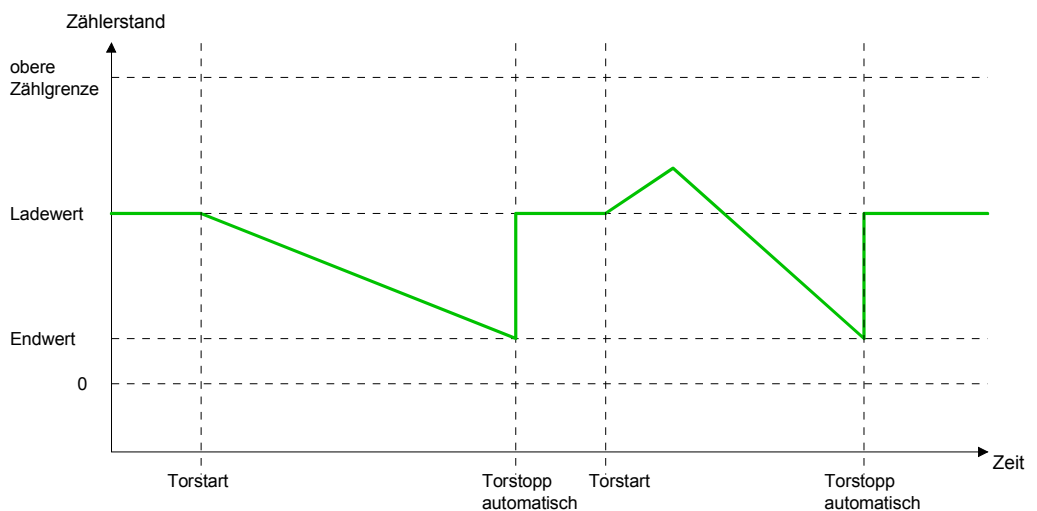
	Gültiger Wertebereich
Endwert	-2 147 483 646 ($-2^{31}+1$) bis +2 147 483 646 ($2^{31}-1$)
Untere Zählgrenze	-2 147 483 648 (-2^{31})



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 Tor wird automatisch geschlossen.
- Zum erneuten Start des Zählvorgangs müssen Sie eine Flanke 0-1 der Torsteuerung erzeugen. Der Zähler beginnt ab dem Ladewert.

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

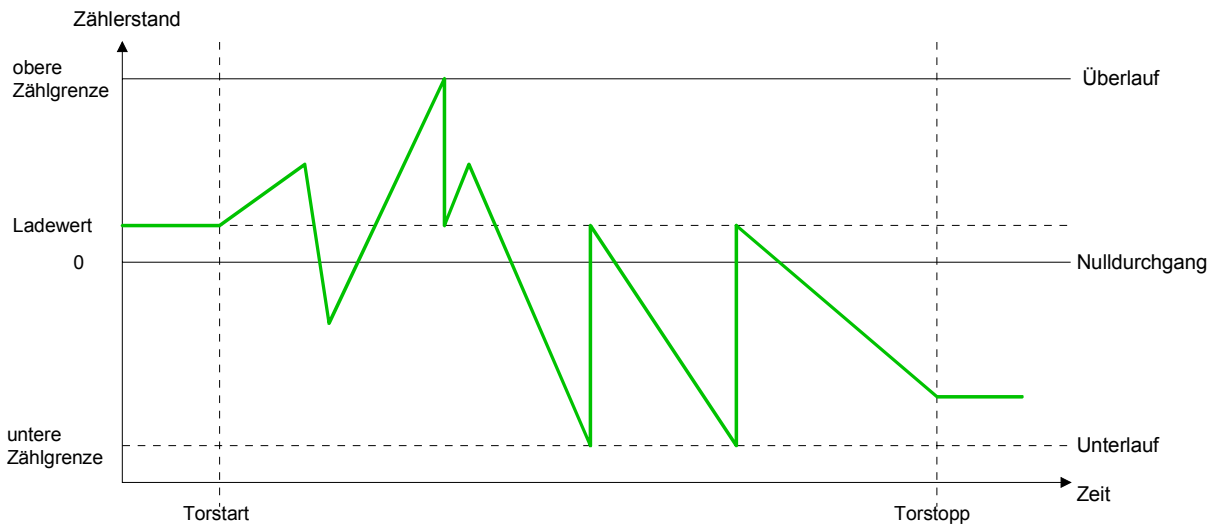


Periodisch Zählen

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.
- Die Zählgrenzen sind auf den maximalen Zählbereich fest eingestellt.

	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.

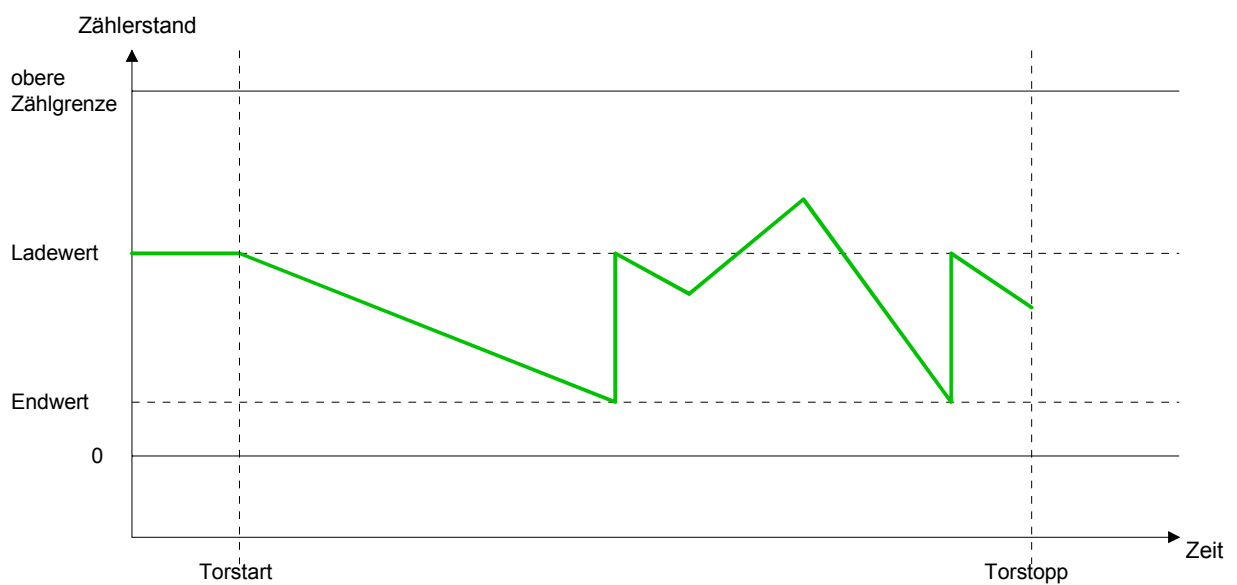
	Gültiger Wertebereich
Endwert	-2 147 483 647 ($-2^{31}+1$) bis +2 147 483 647 ($2^{31}-1$)
Untere Zählgrenze	-2 147 483 648 (-2^{31})



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.
- Sie können über die obere Zählgrenze hinaus zählen.

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



Zähler - Zusatzfunktionen

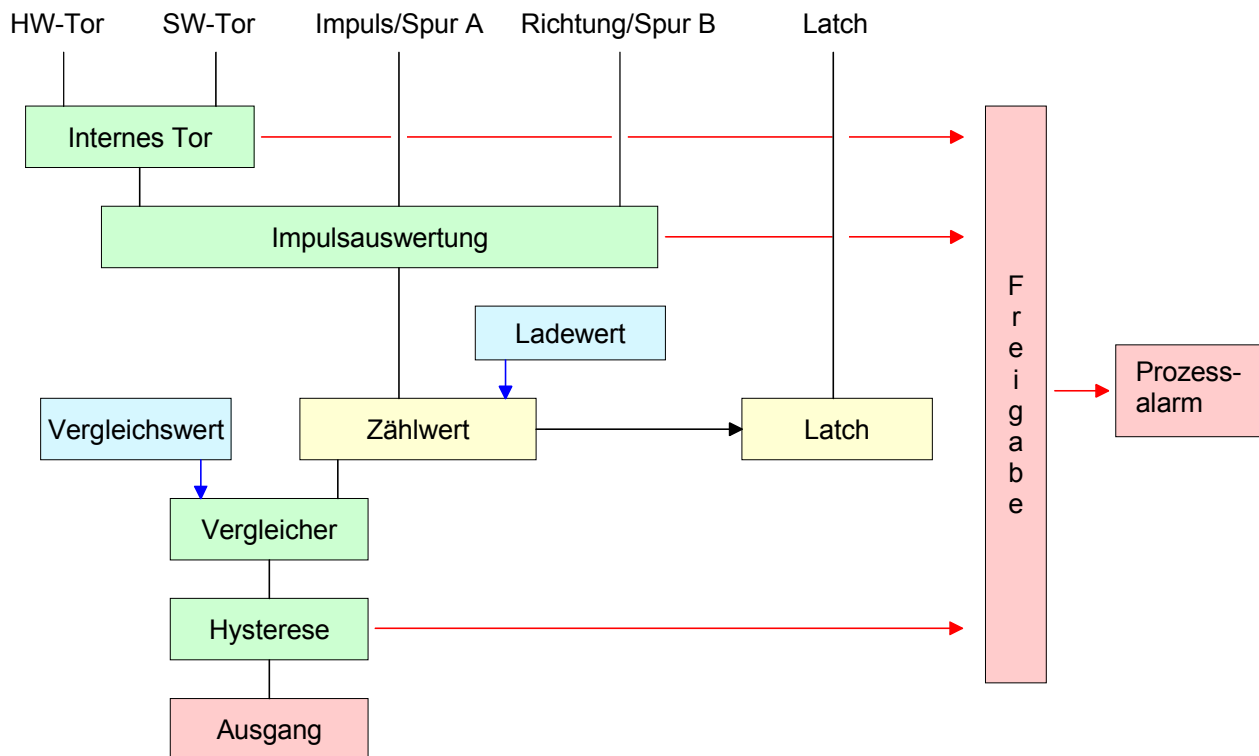
Übersicht

Die nachfolgend aufgeführten Zusatzfunktionen können Sie für jeden 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_x-Eingang am Modul

Schließen (deaktivieren): Flankenwechsel 1-0 am Hardwaretor_x-Eingang am Modul

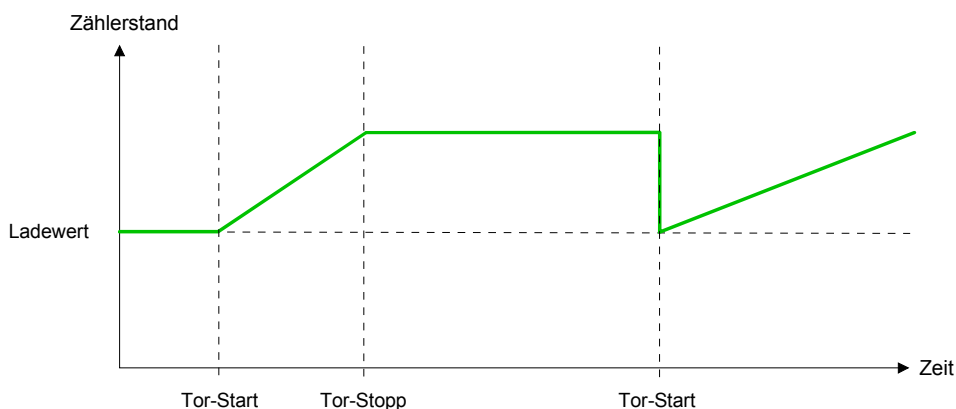
SW-Tor: Öffnen (aktivieren): Im Anwenderprogramm durch Setzen von SW_GATE des SFB 47

Schließen (deaktivieren): Im Anwenderprogramm durch Rücksetzen von SW_GATE des SFB 47

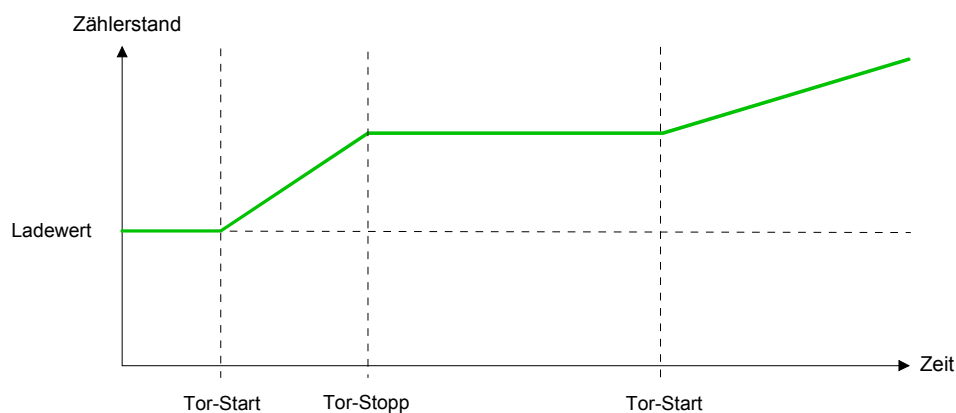
Abbrechende und unterbrechende Tor-Funktion

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 Ladewert

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 Vergleicher" STS_CMP und dem Prozessalarm zugeordnet ist. Abhängig von Zähl- und Vergleichswert, kann der Digitalausgang aktiviert werden. Einen Vergleichswert können Sie über die Parametriermaske bzw. über die Auftragschnittstelle des SFB 47 eingeben.

Verhalten des Ausgangs Über die Parametrierung können Sie das Verhalten des Zählerausgangs festlegen:

- kein Vergleich
- Zählerstand \geq Vergleichswert
- Zählerstand \leq Vergleichswert
- Impuls bei Vergleichswert

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 Nachtriggerung 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.

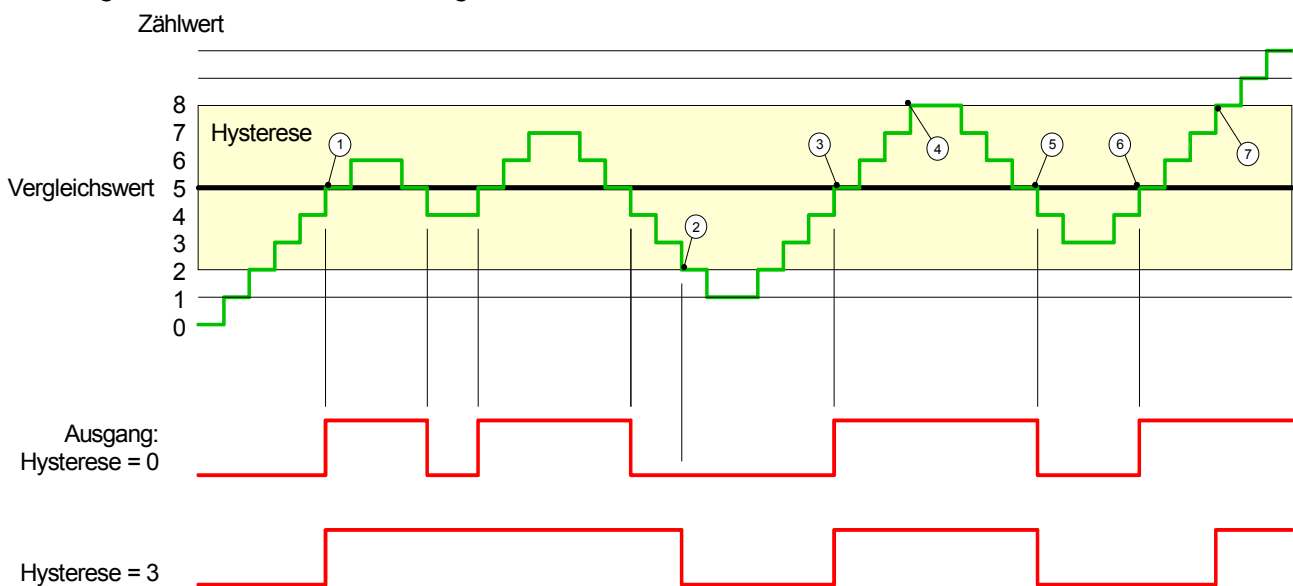
Hysterese

Die Hysterese dient beispielsweise zur Vermeidung von häufigen Schaltvorgängen des Ausgangs und 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- und Unterlauf.

Eine aktive Hysterese bleibt nach der Änderung aktiv. Der neue Hysteresebereich wird beim nächsten Erreichen des Vergleichswertes übernommen.

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

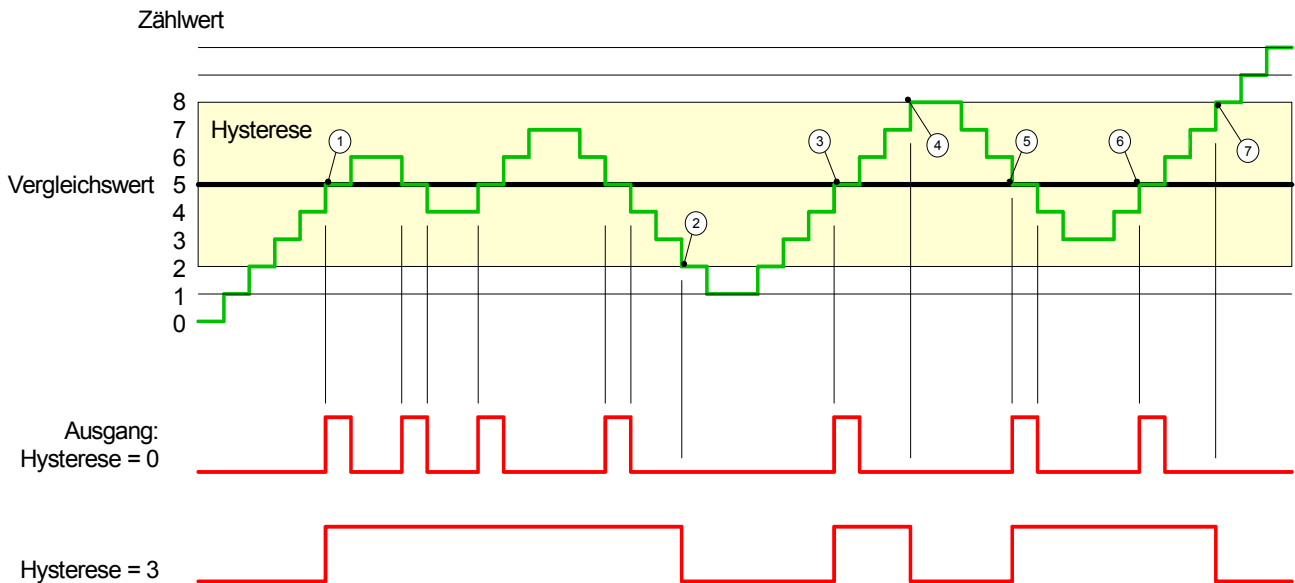
Wirkungsweise bei Zählerwert \geq Vergleichswert



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

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.

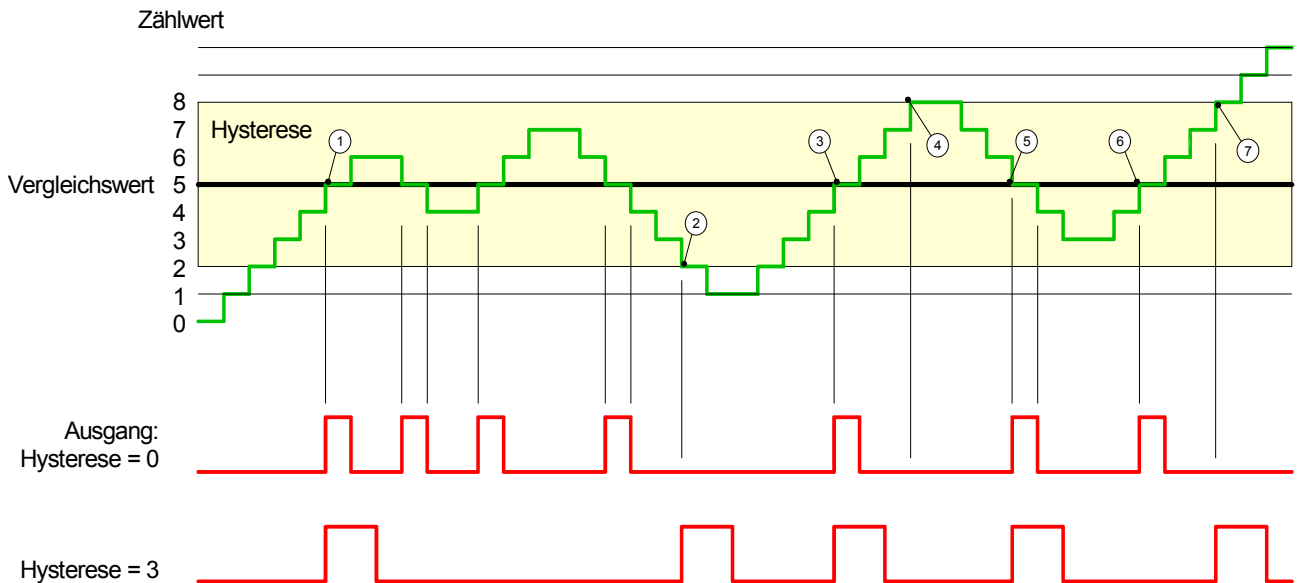
Wirkungsweise bei Impuls bei Vergleichswert mit Impulsdauer Null



- 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 bei Impuls bei Vergleichswert mit Impulsdauer ungleich Null



- ① Zählerwert = Vergleichswert → Impuls der parametrierten Dauer wird ausgegeben, die Hysterese aktiviert und die Zählrichtung gespeichert
- ② Verlassen des Hysterese-Bereichs entgegen der gespeicherten Zählrichtung → Impuls der parametrierten Dauer wird ausgegeben und die Hysterese deaktiviert
- ③ Zählerwert = Vergleichswert → Impuls der parametrierten Dauer 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 parametrierten Dauer 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 parametrierten Dauer wird ausgegeben und die Hysterese deaktiviert

Mit dem Erreichen der Vergleichsbedingung wird die Hysterese aktiv und ein Impuls der parametrierten Dauer ausgegeben. Solange sich der Zählwert innerhalb des Hysterese-Bereichs befindet, wird kein weiterer Impuls abgegeben. Mit Aktivierung der Hysterese wird in der CPU die Zählrichtung festgehalten. Verlässt der Zählwert den Hysterese-Bereich entgegen der gespeicherten Zählrichtung, wird ein Impuls der parametrierten Dauer ausgegeben. Beim Verlassen des Hysterese-Bereichs ohne Richtungsänderung erfolgt keine Impulsausgabe.

Zähler - Diagnose und Alarm

Übersicht

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

- 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

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

Das *Lokaldoppelwort 8* des OB 40 hat folgenden Aufbau:

Lokalbyte	Bit 7 ... Bit 0
8	Bit 0: Flanke an E+0.0 Bit 1: Flanke an E+0.1 Bit 2: Flanke an E+0.2 Bit 3: Flanke an E+0.3 Bit 4: Flanke an E+0.4 Bit 5: Flanke an E+0.5 Bit 6: Flanke an E+0.6 Bit 7: Flanke an E+0.7
9	Bit 0: Flanke an E+1.0 Bit 1: Flanke an E+1.1 Bit 2: Flanke an E+1.2 Bit 3: Flanke an E+1.3 Bit 4: Flanke an E+1.4 Bit 5: Flanke an E+1.5 Bit 6: Flanke an E+1.6 Bit 7: Flanke an E+1.7
10	Bit 0: Tor Zähler 0 geöffnet (aktiviert) Bit 1: Tor Zähler 0 geschlossen Bit 2: Über-/Unterlauf Zähler 0 Bit 3: Zähler 0 hat Vergleichswert erreicht Bit 4: Tor Zähler 1 geöffnet (aktiviert) Bit 5: Tor Zähler 1 geschlossen Bit 6: Über-/Unterlauf Zähler 1 Bit 7: Zähler 1 hat Vergleichswert erreicht
11	Bit 7...0: reserviert

Diagnosealarm

Sie haben die Möglichkeit über die Parametrierung (Datensatz 7Fh) global einen Diagnosealarm für den Analog- und Digitalteil zu aktivieren.

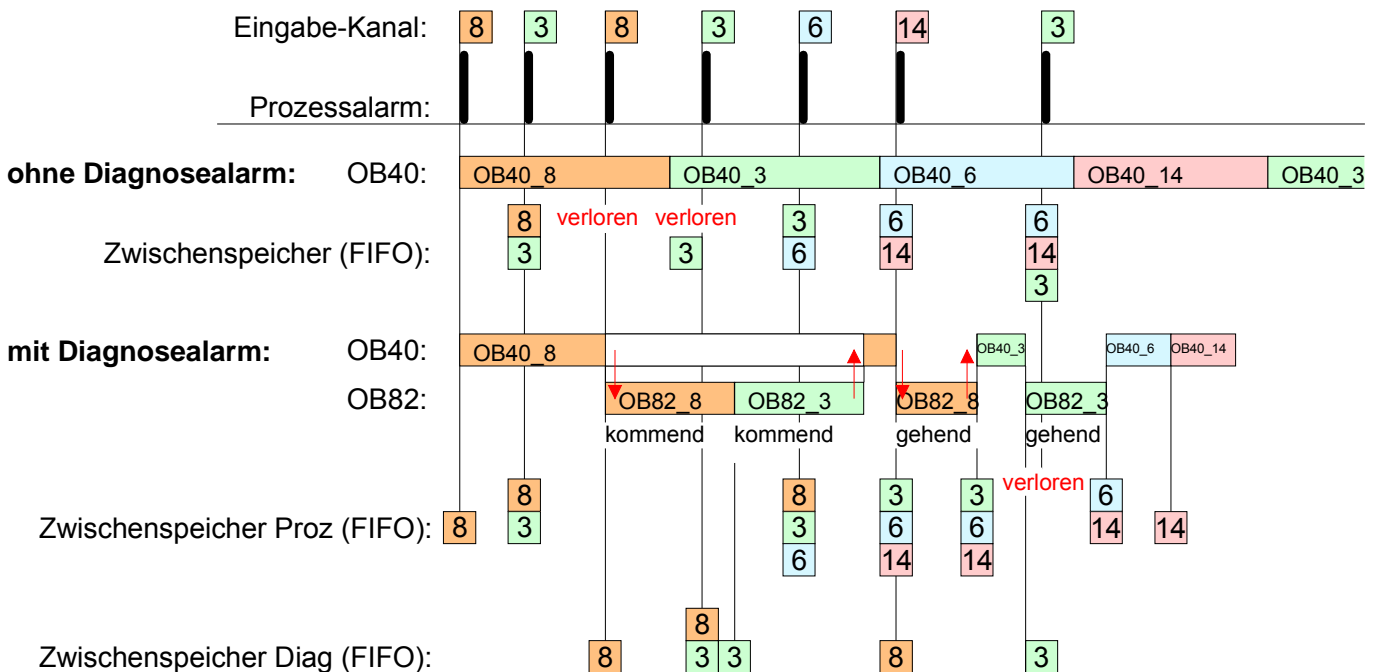
Ein Diagnosealarm tritt auf, sobald während einer Prozessalarmbearbeitung im OB 40, für das gleiche Ereignis ein weiterer Prozessalarm ausgelöst wird. Durch Auslösen eines Diagnosealarms wird die aktuelle Prozessalarmbearbeitung im OB 40 unterbrochen und in OB 82 zur Diagnosealarmbearbeitung_{kommend} 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_{kommend} bearbeitet wird bzw. zwischengespeichert ist, weitere Prozessalarme auf, gehen diese verloren. Ist ein Prozessalarm, für welchen ein Diagnosealarm_{kommend} ausgelöst wurde, abgearbeitet, erfolgt erneut ein Aufruf der Diagnosealarmbearbeitung als Diagnosealarm_{gehend}.

Alle Ereignisse eines Kanals zwischen Diagnosealarm_{kommend} und Diagnosealarm_{gehend} werden nicht zwischengespeichert und gehen verloren. Innerhalb dieses Zeitraums (1. Diagnosealarm_{kommend} bis letzter Diagnosealarm_{gehend}) leuchtet die SF-LED der CPU. Zusätzlich erfolgt für jeden Diagnosealarm_{kommend/gehend} ein Eintrag im Diagnosepuffer der CPU.

Beispiel



Diagnosealarm-
bearbeitung

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_{kommend}

Byte	Bit 7 ... Bit 0
0	Bit 0: gesetzt, wenn Baugruppenstörung Bit 1: 0 (fix) Bit 2: gesetzt, bei Fehler extern Bit 3: gesetzt, bei Kanalfehler vorhanden Bit 4: gesetzt, wenn externe Hilfsspannung fehlt Bit 7 ... 5: 0 (fix)
1	Bit 3 ... 0: Modulklasse 0101b: Analog 1111b: Digital Bit 4: Kanalinformation vorhanden Bit 7 ... 5: 0 (fix)
2	Bit 3 ... 0: 0 (fix) Bit 4: Ausfall Baugruppeninterne Versorgungsspannung (Ausgang überlastet) Bit 7 ... 5: 0 (fix)
3	Bit 5 ... 0: 0 (fix) Bit 6: Prozessalarm verloren Bit 7: 0 (fix)

Datensatz 0
Diagnose_{gehend}

Nach der Fehlerbehebung erfolgt, sofern die Diagnosealarmfreigabe noch aktiv ist, eine Diagnosemeldung_{gehend}.

Byte	Bit 7 ... Bit 0
0	Bit 0: gesetzt, wenn Baugruppenstörung Bit 1: 0 (fix) Bit 2: gesetzt, bei Fehler extern Bit 3: gesetzt, bei Kanalfehler vorhanden Bit 4: gesetzt, wenn externe Hilfsspannung fehlt Bit 7 ... 5: 0 (fix)
1	Bit 3 ... 0: Modulklasse 0101b: Analogmodul 1111b: Digital Bit 4: Kanalinformation vorhanden Bit 7 ... 5: 0 (fix)
2	00h (fix)
3	00h (fix)

Diagnose
Datensatz 1
(Byte 0 ... 15)

Der Datensatz 1 enthält die 4Byte des Datensatzes 0 und zusätzlich 12Byte modulspezifische Diagnosedaten.

Die Diagnosebytes haben folgende Belegung:

Byte	Bit 7 ... Bit 0
0 ... 3	Inhalte Datensatz 0 (siehe vorherige Seite)
4	Bit 6 ... 0: Kanaltyp (hier 70h) 70h: Digitaleingabe 71h: Analogeingabe 72h: Digitalausgabe 73h: Analogausgabe 74h: Analogein-/ausgabe Bit 7: Weitere Kanaltypen vorhanden 0: nein 1: ja
5	Anzahl der Diagnosebits, die das Modul pro Kanal ausgibt (hier 08h)
6	Anzahl der Kanäle eines Moduls (hier 08h)
7	Bit 0: Fehler in Kanalgruppe 0 (E+0.0 ... E+0.3) Bit 1: Fehler in Kanalgruppe 1 (E+0.4 ... E+0.7) Bit 2: Fehler in Kanalgruppe 2 (E+1.0 ... E+1.3) Bit 3: Fehler in Kanalgruppe 3 (E+1.4 ... E+1.7) Bit 4: Fehler in Kanalgruppe 4 (Zähler 0) Bit 5: Fehler in Kanalgruppe 5 (Zähler 1) Bit 6: reserviert Bit 7: reserviert
8	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ...Eingang E+0.0 Bit 1: 0 (fix) Bit 2: ...Eingang E+0.1 Bit 3: 0 (fix) Bit 4: ... Eingang E+0.2 Bit 5: 0 (fix) Bit 6: ... Eingang E+0.3 Bit 7: 0 (fix)
9	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ...Eingang E+0.4 Bit 1: 0 (fix) Bit 2: ...Eingang E+0.5 Bit 3: 0 (fix) Bit 4: ... Eingang E+0.6 Bit 5: 0 (fix) Bit 6: ... Eingang E+0.7 Bit 7: 0 (fix)
10	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ...Eingang E+1.0 Bit 1: 0 (fix) Bit 2: ...Eingang E+1.1 Bit 3: 0 (fix) Bit 4: ... Eingang E+1.2 Bit 5: 0 (fix) Bit 6: ... Eingang E+1.3 Bit 7: 0 (fix)

Fortsetzung ...

... Fortsetzung Datensatz 1

Byte	Bit 7 ... Bit 0
11	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ... Eingang E+1.4 Bit 1: 0 (fix) Bit 2: ... Eingang E+1.5 Bit 3: 0 (fix) Bit 4: ... Eingang E+1.6 Bit 5: 0 (fix) Bit 6: ... Eingang E+1.7 Bit 7: 0 (fix)
12	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ... Tor Zähler 0 geschlossen Bit 1: 0 (fix) Bit 2: ... Tor Zähler 0 geöffnet Bit 3: 0 (fix) Bit 4: ... Über-/Unterlauf Zähler 0 Bit 5: 0 (fix) Bit 6: ... Zähler 0 hat Vergleichswert erreicht Bit 7: 0 (fix)
13	Diagnosealarm wegen Prozessalarm verloren auf... Bit 0: ... Tor Zähler 1 geschlossen Bit 1: 0 (fix) Bit 2: ... Tor Zähler 1 geöffnet Bit 3: 0 (fix) Bit 4: ... Über-/Unterlauf Zähler 1 Bit 5: 0 (fix) Bit 6: ... Zähler 1 hat Vergleichswert erreicht Bit 7: 0 (fix)
14	reserviert
15	reserviert

Teil 6 Einsatz PtP-Kommunikation

Überblick

Inhalt dieses Kapitels ist der Einsatz der RS485-Schnittstelle für die serielle PtP-Kommunikation.

Sie erhalten hier alle Informationen zu den Protokollen und zur Projektierung der Schnittstelle, die für die serielle Kommunikation über RS485 erforderlich sind.

Inhalt

Thema	Seite
Teil 6 Einsatz PtP-Kommunikation	6-1
Schnelleinstieg	6-2
Prinzip der Datenübertragung	6-3
Einsatz der RS485-Schnittstelle	6-4
Parametrierung	6-5
Kommunikation	6-8
Protokolle und Prozeduren	6-14
Modbus - Funktionscodes	6-18
Modbus - Beispiel zur Kommunikation.....	6-22

Schnelleinstieg

Allgemein Die CPU 312SC besitzt eine RS485-Schnittstelle, die fix auf PtP-Kommunikation (point to point) eingestellt ist. Dies ermöglicht die serielle Prozessankopplung zu verschiedenen Ziel- oder Quellsystemen.

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

Parametrierung Die Parametrierung erfolgt zur Laufzeit unter Einsatz des SFC 216 (SER_CFG). Hierbei sind für alle Protokolle mit Ausnahme von ASCII die Parameter in einem DB abzulegen.

Kommunikation Mit SFCs steuern Sie die Kommunikation. Das Senden erfolgt unter Einsatz des SFC 217 (SER_SND) und das Empfangen über SFC 218 (SER_RCV).

Durch erneuten Aufruf des 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 SFC 218 SER_RCV nach einem SER_SND das Quittungstelegramm auslesen.

Die SFCs befinden sich im Lieferumfang der CPU.

Übersicht der SFCs für die serielle Kommunikation

Folgende SFCs kommen für die serielle Kommunikation zum Einsatz:

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

Prinzip der Datenübertragung

Übersicht

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

Prinzip

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.

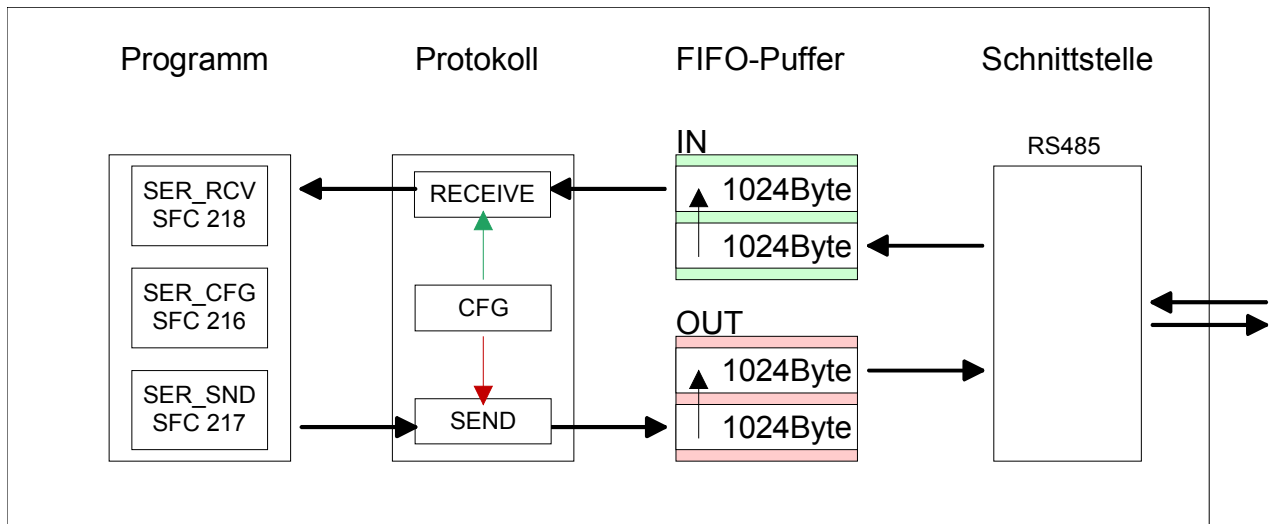
Sofern Daten mittels eines Protokolls übertragen werden, erfolgt die Einbettung der Daten in das entsprechende Protokoll automatisch.

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 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 Quittungs-telegramm durch Aufruf des SFC 218 SER_RCV auszulesen.

RS485-PtP-Kommunikation



Einsatz der RS485-Schnittstelle

Eigenschaften RS485

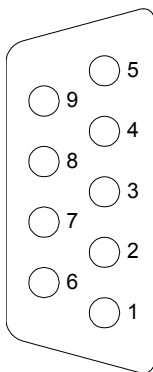
Die CPU 312SC besitzt eine RS485-Schnittstelle, die fix auf PtP-Kommunikation eingestellt ist. Parametrierung und Kommunikation erfolgen mittels SFCs.

Folgende Eigenschaften zeichnen die RS485-Schnittstelle aus:

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

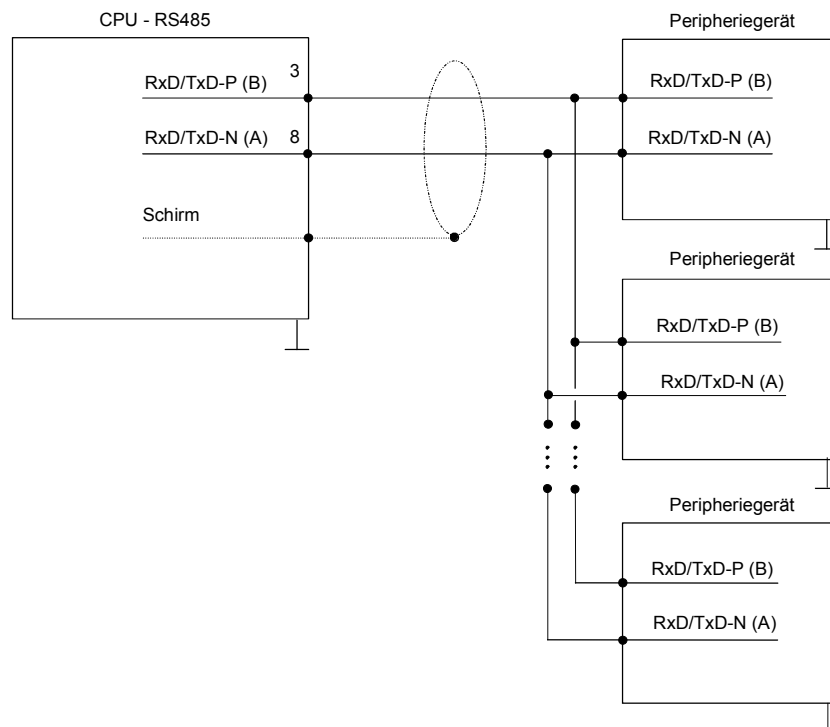
Anschluss RS485

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



Parametrierung

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

Name	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=None, 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)

Parameter- beschreibung

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



Hinweis!

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

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)
DBW4:	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 RST=0, beim Empfangen ist RTS=1.

RetVal SFC 216 (Fehlermeldung SER_CFG) Rückgabewerte, die der Baustein liefert:

Fehlercode	Beschreibung
0000h	kein Fehler
809Ah	Schnittstelle ist nicht vorhanden bzw. Schnittstelle wird für Profibus verwendet
8x24h	Fehler in SFC-Parameter x, mit x: 1: Fehler in "Protokoll" 2: Fehler in "Parameter" 3: Fehler in "Baudrate" 4: Fehler in "CharLength" 5: Fehler in "Parity" 6: Fehler in "StopBits" 7: Fehler in "FlowControl" (Parameter fehlt)
809xh	Fehler in Wert des SFC-Parameter x, mit x: 1: Fehler in "Protokoll" 3: Fehler in "Baudrate" 4: Fehler in "CharLength" 5: Fehler in "Parity" 6: Fehler in "StopBits"
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 ...

Kommunikation

Übersicht

Die Kommunikation erfolgt über die Sende- und Empfangsbausteine SFC 217 (SER_SND) und SFC 218 (SER_RCV). Die SFCs befinden sich im Lieferumfang der CPU.

SFC 217 (SER_SND)

Mit diesem Baustein werden Daten über die serielle Schnittstelle gesendet. Durch erneuten Aufruf des 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 SFC 218 SER_RCV auszulesen.

Parameter

Name	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 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 SFC 217
(Fehlermeldung
SER_SND)**

Rückgabewerte, die der Baustein liefert:

Fehlercode	Beschreibung
0000h	Daten gesendet - fertig
1000h	Nichts gesendet (Datenlänge 0)
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 SFC-Parameter x, mit x: 1: Fehler in "DataPtr" 2: Fehler in "DataLen"
8122h	Fehler in Parameter "DataPtr" (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)
9001h	Daten sind zu lang (>1024Byte)
9002h	Daten sind zu kurz (0Byte)

... Fortsetzung
RetVal SFC 217
SER_SND

USS

Fehlercode	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)

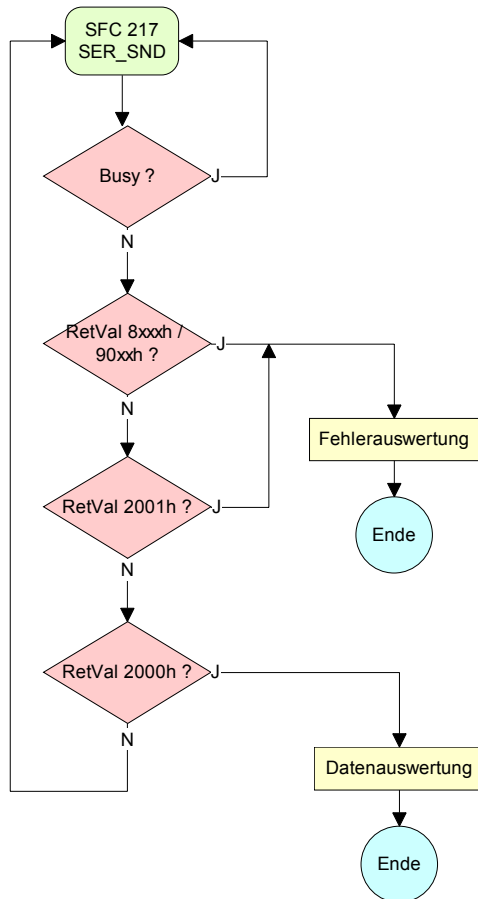
Modbus RTU/ASCII Master

Fehlercode	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)

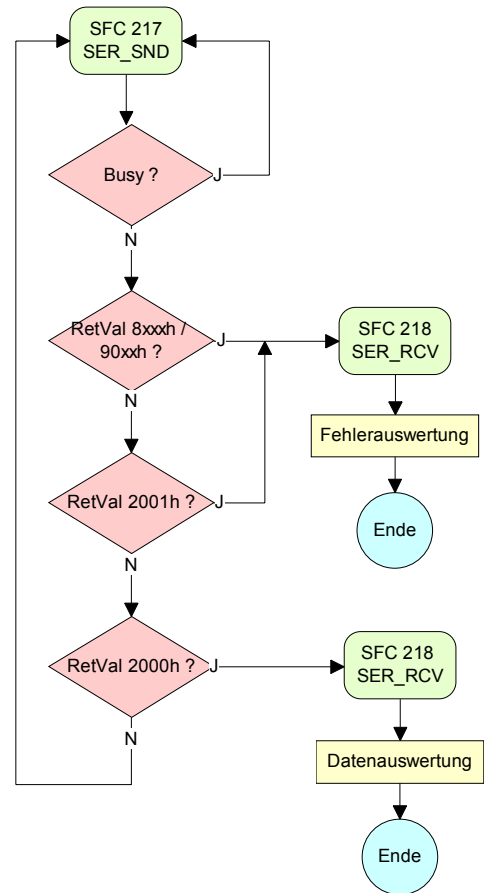
Prinzip der Programmierung

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

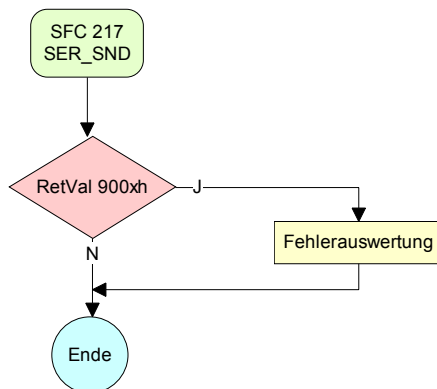
3964R



USS / Modbus



ASCII / STX/ETX



**SFC 218
(SER_RCV)**

Mit diesem Baustein werden Daten über die serielle Schnittstelle empfangen. Bei den Protokollen USS und Modbus können Sie durch Aufruf des SFC 218 SER_RCV nach einem SER_SND das Quittungstelegramm auslesen.

Parameter

Name	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 (Stopbitfehler)
2	parity	Paritätsfehler
3	overflow	Der Puffer ist voll.

STX/ETX

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

3964R / Modbus RTU/ASCII Master

Bit	Fehler	Beschreibung
0	overflow	Das empfangene Telegramm übersteigt die Größe des Empfangspuffers.

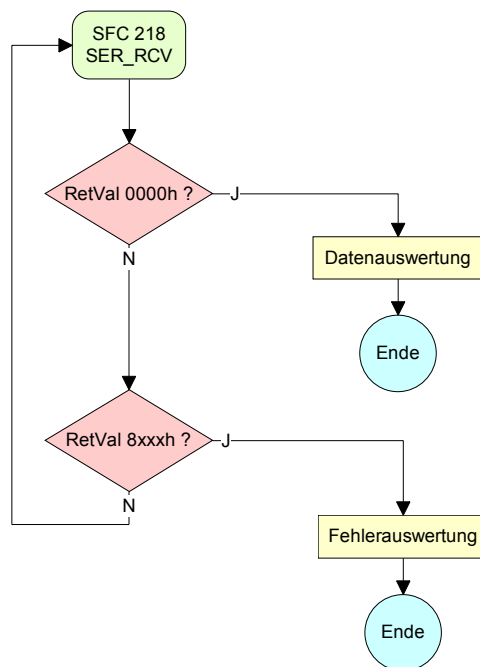
**RetVal SFC 218
(Fehlermeldung
SER_RCV)**

Rückgabewerte, die der Baustein liefert:

Fehlercode	Beschreibung
0000h	kein Fehler
1000h	Empfangspuffer ist zu klein (Datenverlust)
8x24h	Fehler in SFC-Parameter x, mit x: 1: Fehler in "DataPtr" 2: Fehler in "DataLen" 3: Fehler in "Error"
8122h	Fehler in Parameter "DataPtr" (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.



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-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 unter ftp.vipa.de.

STX/ETX

STX/ETX ist ein einfaches Protokoll mit Start- und Ende-Kennung. Hierbei stehen STX für **Start of Text** und ETX für **End of Text**.

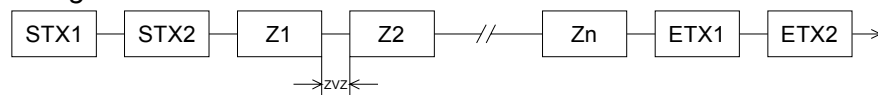
Die Prozedur STX/ETX wird zur Übertragung von ASCII-Zeichen eingesetzt. Sie arbeitet ohne Blockprüfung (BCC). Sollen Daten von der Peripherie eingelesen werden, muss das Start-Zeichen vorhanden sein, anschließend folgen die zu übertragenden Zeichen. Danach muss das Ende-Zeichen vorliegen.

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 SFC 217 (SER_SND) übergeben und von dort mit angefügten Start- und Endezeichen über die serielle Schnittstelle an den Kommunikationspartner übertragen.

Telegrammaufbau:



Sie können bis zu 2 Anfangs- und Endezeichen frei definieren.

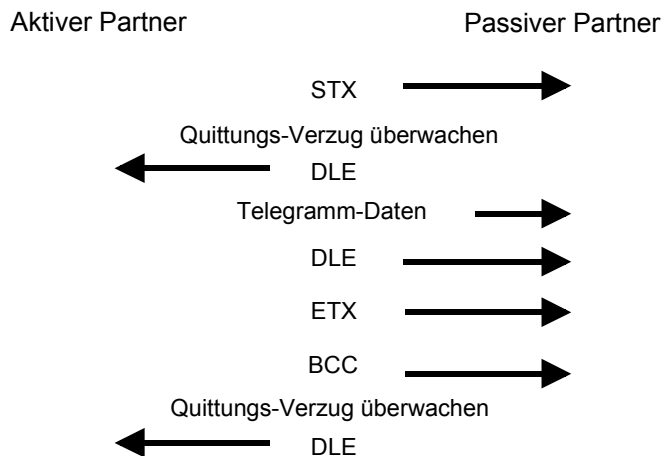
Es kann mit 1, 2 oder keiner Start- und mit 1, 2 oder keiner Ende-Kennung gearbeitet werden. Als Start- bzw. Ende-Kennung sind alle Hex-Werte von 00h bis 1Fh zulässig. Zeichen größer 1Fh werden ignoriert und nicht berücksichtigt. In den Nutzdaten sind Zeichen kleiner 20h nicht erlaubt und können zu Fehlern führen. Die Anzahl der Start- und Endezeichen kann unterschiedlich sein (1 Start, 2 Ende bzw. 2 Start, 1 Ende oder andere Kombinationen). Wird kein Ende-Zeichen definiert, so werden alle gelesenen Zeichen nach Ablauf einer parametrierbaren Zeichenverzugszeit (Timeout) an die CPU übergeben.

3964R

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 **Start of Text**
- DLE **Data Link Escape**
- ETX **End of Text**
- BCC **Block Check Character**
- NAK **Negative Acknowledge**

Prozedurablauf

Sie können pro Telegramm maximal 255Byte übertragen.

**Hinweis!**

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

Am Bus können 1 Master und max. 31 Slaves angebunden sein, wobei die einzelnen Slaves vom Master über ein Adresszeichen im Telegramm angewählt werden. Die Kommunikation erfolgt ausschließlich über den Master im Halbduplex-Betrieb.

Nach einem Sendeauftrag ist das Quittungstelegramm durch Aufruf des SFC 218 SER_RCV auszulesen.

Die Telegramme für Senden und Empfangen haben folgenden Aufbau:

Master-Slave-Telegramm

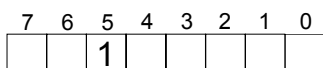
STX	LGE	ADR	PKE		IND		PWE		STW		HSW		BCC
02h			H	L	H	L	H	L	H	L	H	L	

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 | |

Broadcast mit gesetztem Bit 5 in ADR-Byte



↑
Broadcast

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" Sendeauftrag ist beim Broadcast keine Telegrammauswertung über 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 SFC 218 SER_RCV auszulesen.

Die Anforderungs-Telegramme, die ein Master sendet und die Antwort-Telegramme eines Slaves haben den gleichen Aufbau:

Start- zeichen	Slave- Adresse	Funktions- Code	Daten	Fluss- kontrolle	Ende- zeichen
-------------------	-------------------	--------------------	-------	---------------------	------------------

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 SFC 218 SER_RCV erforderlich.

Nur Schreibaufträge dürfen als Broadcast gesendet werden.

ASCII-, RTU-Modus

Bei Modbus gibt es zwei unterschiedliche Übertragungsmodi:

- 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.

Die Modus-Wahl erfolgt zur Laufzeit unter Einsatz des SFC 216 SER_CFG.

Unterstützte
Modbus-Protokolle

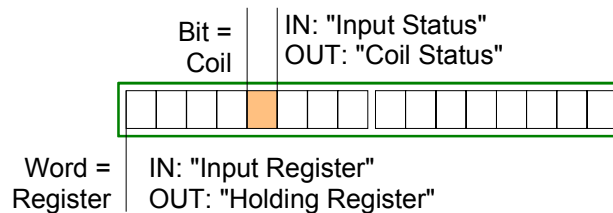
Die RS485-Schnittstelle unterstützt folgende Modbus-Protokolle:

- Modbus RTU Master
- Modbus ASCII Master

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.

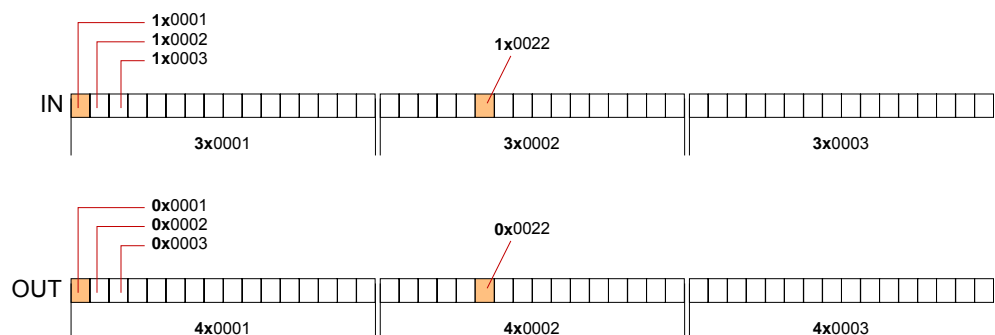
Bereichsdefinitionen

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 Folgeseiten.

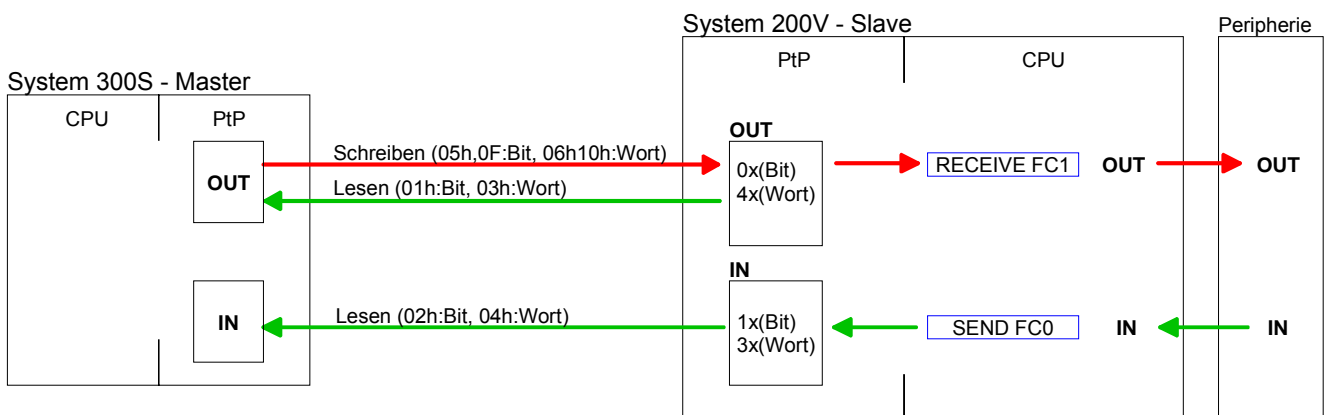
Übersicht

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

Für die Byte-Reihenfolge im Wort gilt immer: *1 Wort*
 High- Low-
 Byte 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 Code 01h: n Bit lesen von Master-Ausgabe-Bereich 0x
01h, 02h 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 max. 250Byte		1Wort

Read n Words 03h: n Worte lesen von Master-Ausgabe-Bereich 4x
03h, 04h 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 max. 125Worte		1Wort

Write 1 Bit Code 05h: 1 Bit schreiben in Master-Ausgabe-Bereich 0x
05h 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
1 Byte	1 Byte	1 Wort	1 Wort	1 Byte	1 Byte	1 Byte	1 Byte	1 Wort
						max. 250 Byte		

Antworttelegramm

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

**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
1 Byte	1 Byte	1 Wort	1 Wort	1 Byte	1 Wort	1 Wort	1 Wort	1 Wort
						max. 125 Worte		

Antworttelegramm

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

Modbus - Beispiel zur Kommunikation

Übersicht

In dem Beispiel wird eine Kommunikation zwischen einem Master und einem Slave über Modbus aufgebaut.

Das System besteht aus folgenden Komponenten:

Modbus-Master (M)	Modbus-Slave (S)
CPU 31xS	CPU 21xSER-1

Komponenten

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

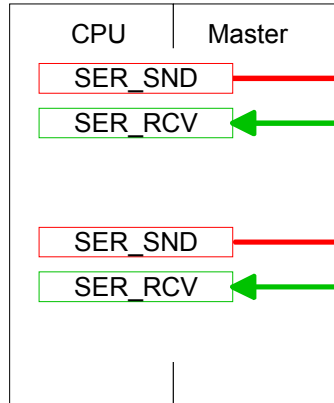
- Bauen Sie ein Modbus-System bestehend aus CPU 31xS als Modbus-Master und CPU 21xSER-1 als Modbus-Slave und Modbus-Kabel auf.
- Projektieren Sie die Master-Seite!
Erstellen Sie hierzu ein SPS-Anwenderprogramm nach folgender Struktur:
OB 100: Aufruf SFC 216 (Konfiguration als Modbus RTU-Master) mit Timeout-Angabe 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.
- 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.

Auf der Folgeseite ist die Struktur für die jeweiligen SPS-Programme für Master- und Slave-Seite dargestellt.

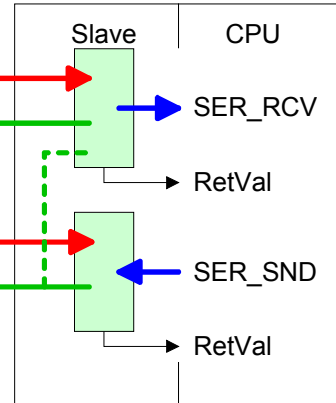
Master

Slave

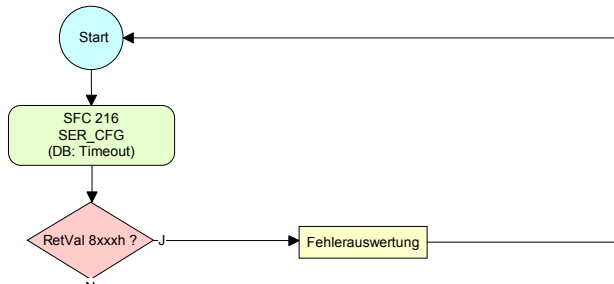
CPU 31xS



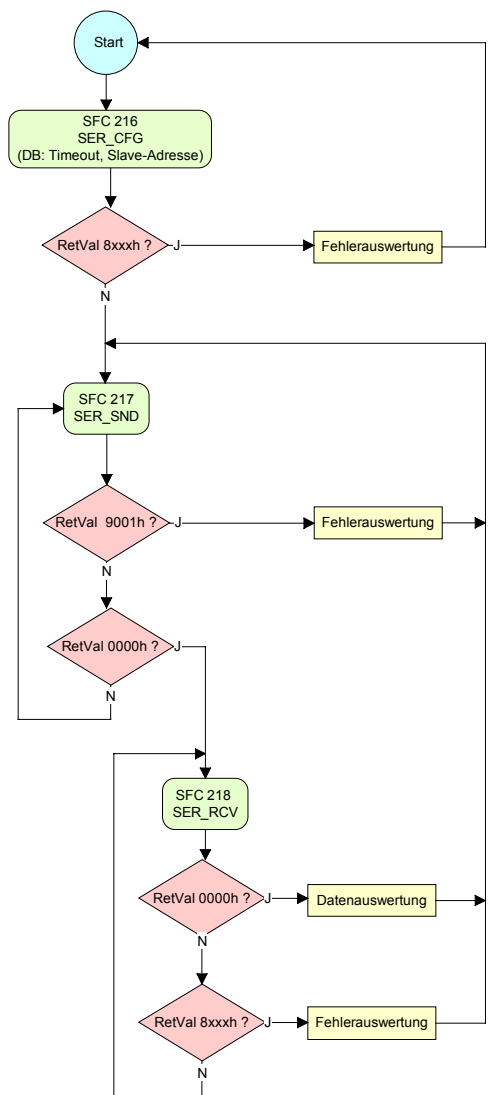
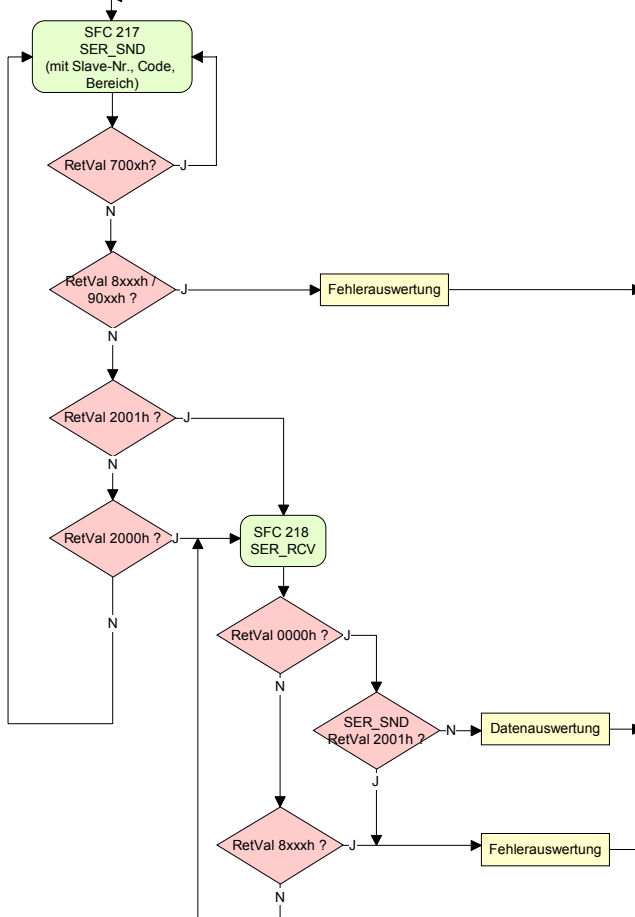
CPU 21xSER-1



OB100:



OB1:



Teil 7 WinPLC7

Überblick

In diesem Teil wird die Programmier- und Simulationssoftware WinPLC7 von VIPA vorgestellt. WinPLC7 eignet sich für alle mit Siemens STEP[®]7 programmierbaren Steuerungen.

Neben der Systemvorstellung und der Installation finden Sie hier die Grundzüge der Programmbedienungen an einem Beispielprojekt erklärt.

Nähere Informationen zum Einsatz von WinPLC7 können Sie der Online-Hilfe bzw. der Online-Dokumentation von WinPLC7 entnehmen.

Inhalt

Thema	Seite
Teil 7 WinPLC7	7-1
Systemvorstellung	7-2
Installation	7-3
Beispiel zur Projektierung	7-4

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 Fehlverhalten und Diagnosemöglichkeit über Diagnosebuffer, USTACK und BSTACK.



Hinweis!

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 den SIMATIC Manager von Siemens zu verwenden. Die Vorgehensweisen hierzu finden Sie in diesem Handbuch.

Systemvoraussetzungen

- Pentium mit 233MHz und 64MByte Arbeitsspeicher
- Grafikkarte mit mind. 16Bit Farben - wir empfehlen eine Bildschirmauflösung von mind. 1024x768 Pixel.
- Windows 98SE/ME, Windows 2000, Windows XP (Home und Professional), Windows Vista

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.de im Service-Bereich unter *Downloads* finden Sie einen Link auf die aktuellste Demo-Version und auf Updates von WinPLC7.

CD

Best.-Nr.	Beschreibung
SW211C1DD	WinPLC7 Einzellizenz, CD, mit deutscher Beschreibung
SW211C1ED	WinPLC7 Einzellizenz, CD, mit englischer Beschreibung
SW900T0LA	ToolDemo VIPA-Software-Sammlung mit kostenfreien bzw. freischaltbaren Demoversionen

Installation

Voraussetzung

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

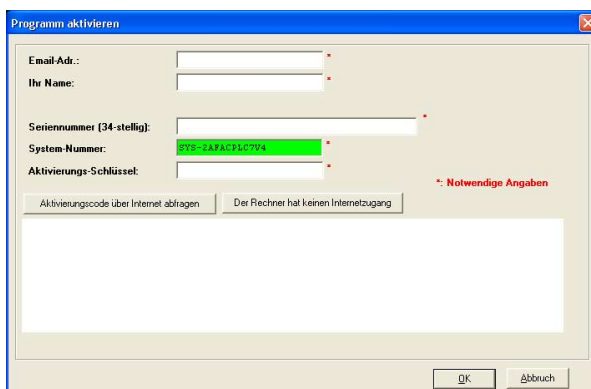
Installation WinPLC7 Demo

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

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

Aktivierung der "Profi"-Version

- Starten Sie WinPLC7. Es erscheint der "Demo"-Dialog.
- Drücken Sie die Taste <q>. Es erscheint folgender Aktivierungsdialog:



- Füllen Sie folgende Felder aus:
Email-Adr., *Ihr Name* und *Seriennummer*. Ihre Seriennummer finden Sie auf einem Aufkleber auf der CD-Hülle von WinPLC7.
- 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.
- 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 WinPLC7-V4/WinPcap_4_0.exe.

Führen Sie diese Datei aus und folgen Sie den Anweisungen.

Beispiel zur Projektierung

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.

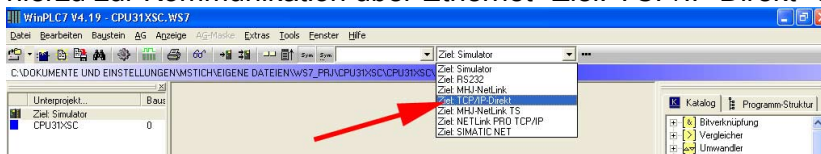
Hierbei soll gelten:


- 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.

- Projektierung**
- Starten Sie WinPLC7 ("Profi"-Version)
 - Legen Sie mit **Datei** > *Projekt öffnen/erzeugen* ein neues Projekt an und öffnen Sie dies.

- Hardware-Konfiguration**
- 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.



- Starten Sie mit  den Hardware-Konfigurator. Bitte beachten Sie, dass die Auswahl eines Objekts im Hardware-Katalog durch Doppelklick erfolgt.
- Wählen Sie im Register *SPS-System selektieren* den Parameter "VIPA SPEED7" und klicken Sie auf [Erzeugen]. Eine neue Station wird angelegt.
- Sichern Sie die leere Station. Vor der Sicherung können Sie Ihrer Station einen Namen vergeben und Kommentar hinzufügen.
- Gehen Sie im Hardware-Katalog auf CPU SPEED7 und fügen Sie die entsprechende VIPA-CPU durch Doppelklick in der Station ein.
- Platzieren Sie für die Ausgabe ein digitales Ausgabe-Modul und geben Sie diesem die Anfangsadresse 124.
- Sichern Sie die Hardware-Konfiguration.

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

- Öffnen Sie die *CPU-Eigenschaften*, indem Sie im Hardware-Konfigurator auf die CPU auf Steckplatz 2 doppelklicken.
- Klicken Sie auf die Schaltfläche [Ethernet CP-Einstellungen (PG/OP-Kanal)]. Es öffnet sich *Eigenschaften CP343*.
- Wählen Sie das Register *Allgemeine Parameter* an.
- Klicken Sie auf [Eigenschaften Ethernet].
- Wählen Sie das Subnetz "PG_OP_Ethernet".
- Geben Sie eine gültige IP-Adresse und Subnetzmaske an. Sie erhalten diese von Ihrem Systemadministrator.
- Schließen Sie alle Dialogfenster mit [OK].
- Stellen Sie, wenn nicht schon geschehen, "Ziel: Extern TCP/IP direkt" ein.
- Öffnen Sie mit **Online** > *Konfiguration übertragen* den gleichnamigen Dialog.
- Klicken Sie auf [Erreichbare Teilnehmer]. Bitte beachten Sie, dass hierzu WinPCap installiert sein muss!
- 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.
- 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 parametrisiert haben und klicken Sie auf [Parameter schreiben].
- 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.
- 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.



Hinweis!

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.

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

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.

Baustein FC 1 erzeugen

- Wählen Sie **Datei > Neuen Baustein erzeugen**.
- 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.

- Gehen Sie auf der *Parametertabelle* in die Zeile "in -->" und tragen Sie im Feld *Name* "value1" ein. Drücken Sie die [Return]-Taste. Der Cursor springt zu der Spalte für den Datentyp.
- Sie können jetzt entweder den Datentyp direkt eingeben oder durch Drücken der [Return]-Taste aus einer Liste verfügbarer Datentypen auswählen. Geben Sie als Datentyp INT an und betätigen Sie die [Return]-Taste. Der Cursor springt zu der Spalte für den *Kommentar*.
- Geben Sie hier "1. Vergleichswert" an und drücken Sie die [Return]-Taste. Eine neue "in -->"-Zeile wird erzeugt und der Cursor in *Name* gesetzt.
- Verfahren Sie für *value2* auf die gleiche Weise wie unter *value1* beschrieben.
- Speichern Sie den Baustein.

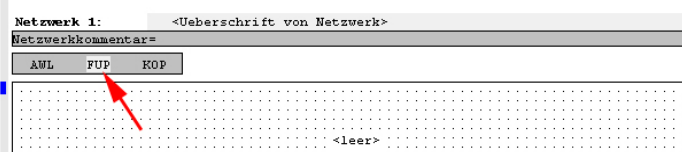
Die Parametertabelle enthält nun folgende Einträge:

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

Programm eingeben

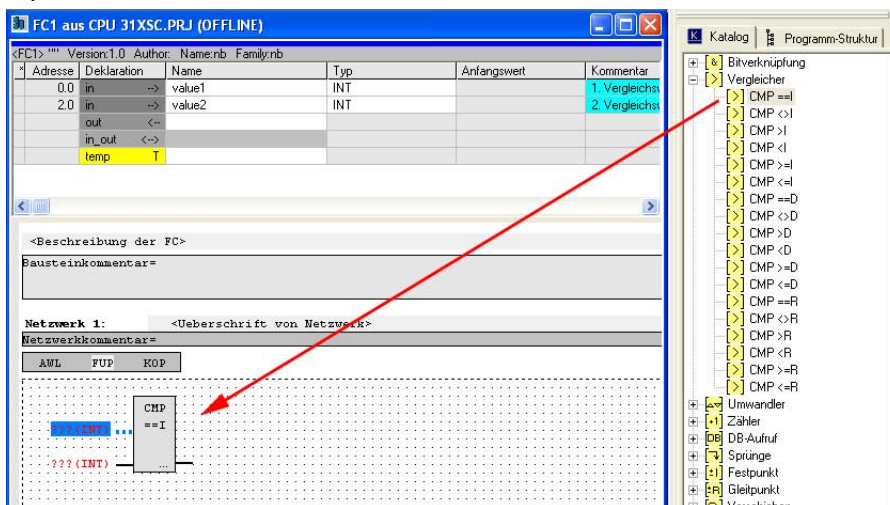
Wie in der Aufgabenstellung gefordert soll je nach Vergleich von *value1* und *value2* der entsprechende Ausgang aktiviert werden. Für jede Vergleichsoperation ist ein Netzwerk anzulegen.

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



- 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.

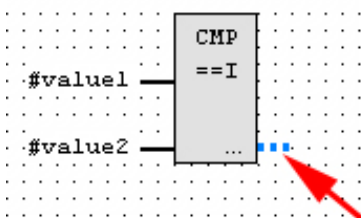
- Öffnen Sie im *Katalog* die Kategorie "Vergleicher" und fügen Sie die Operation "CMP==I" in Ihr Netzwerk ein.



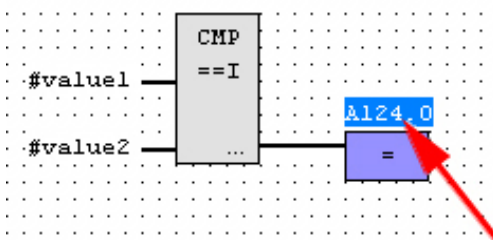
- Klicken Sie auf den linken oberen Eingang und fügen Sie *value1* ein. Da es sich hierbei um Bausteinparameter handelt, können Sie durch Eingabe von "#" eine Auswahlliste der Bausteinparameter öffnen.
- Geben Sie "#" ein und betätigen Sie die [Return]-Taste.
- Wählen Sie aus der Auswahlliste den entsprechenden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

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

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



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



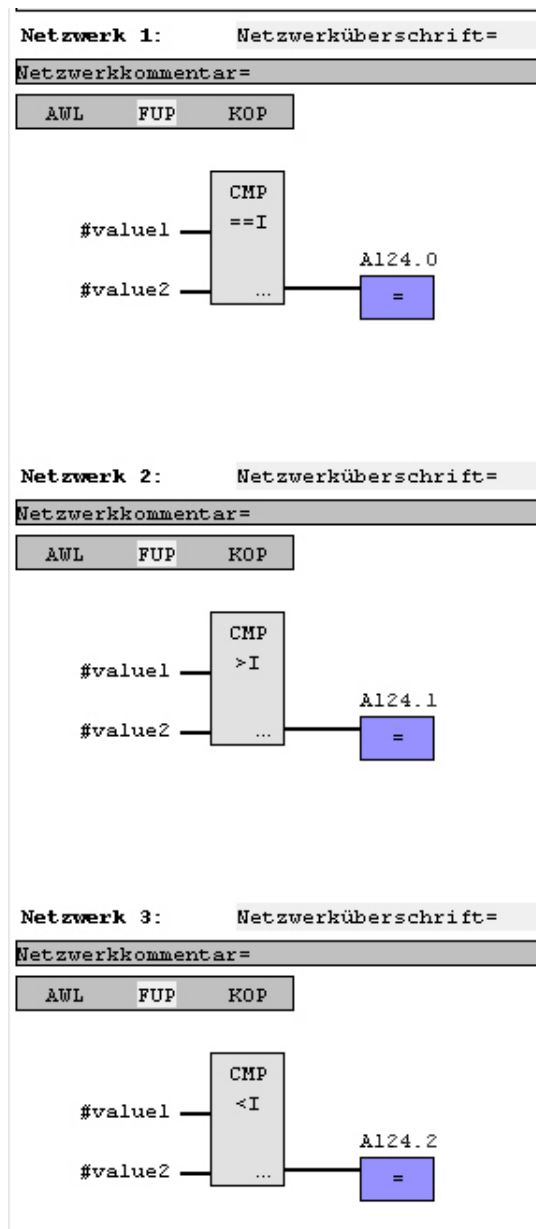
Hiermit ist Netzwerk1 abgeschlossen.

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:

- Bewegen Sie Ihre Maus auf das Editor-Fenster an beliebiger Stelle und betätigen Sie die rechte Maustaste.
- Wählen Sie aus dem Kontextmenü "Einfügen neues Netzwerk". Es öffnet sich ein Dialogfeld zur Vorgabe von Position und Anzahl der Netzwerke.
- Verfahren Sie auf die gleiche Weise wie für "Netzwerk 1" beschrieben.
- Speichern Sie den FC 1 mit **Datei** > *Aktuelles Fenster speichern* bzw. mit [Strg]+[S].

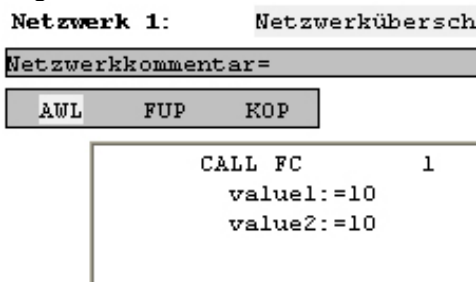
Nachdem Sie die noch fehlenden Netzwerke ausprogrammiert haben, hat der FC 1 folgenden Aufbau:



Baustein OB 1 erzeugen

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

- Zur Erzeugung des OB 1 wählen Sie entweder **Datei** > *Neuen Baustein erzeugen* oder klicken Sie im *Projektnavigator* auf die Schaltfläche [OB 1 anzeigen] und legen Sie den OB 1 an.
- Wechseln Sie in die AWL-Ansicht.
- Geben Sie "Call FC 1" ein und betätigen Sie die [Return]-Taste. Die FC-Parameter werden automatisch angezeigt und die folgenden Parameter zugeordnet:

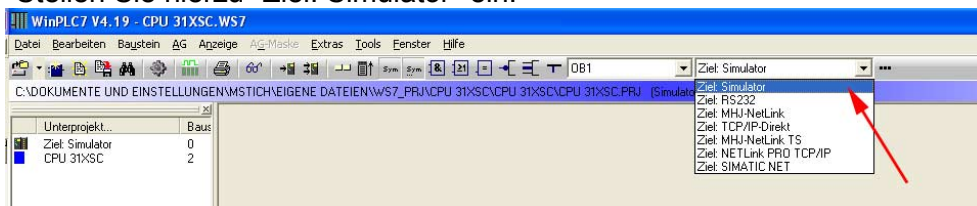


- Speichern Sie den OB 1 mit **Datei** > *Aktuelles Fenster speichern* bzw. mit [Strg]+[S].

SPS-Programm in Simulator testen

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

- Stellen Sie hierzu "Ziel: Simulator" ein.



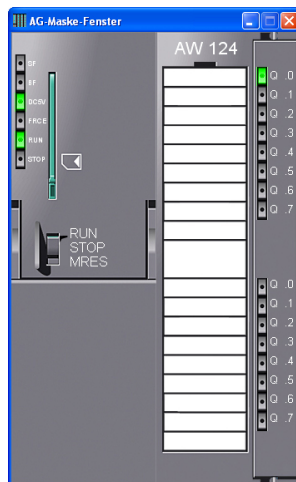
- Übertragen Sie die Bausteine in den Simulator mit **AG** > *Alle Bausteine senden*.
- Schalten Sie Ihre CPU in RUN, indem Sie auf das Foto "Schalter/Betriebszustand" klicken und wählen Sie im Dialogfeld die Schaltfläche [Neustart] an. Die Anzeige wechselt von STOP nach RUN.
- Zur Anzeige des Prozessabbild gehen Sie auf **Anzeige** > *PAA/PAE-Fenster anzeigen*.
- Doppelklicken Sie auf das Prozessabbild und geben Sie im Register "Zeile2" die Adresse PAB124 an. Bestätigen Sie Ihre Eingabe mit [OK]. Ein mit roter Farbe hinterlegter Wert entspricht einer logischen "1".
- Öffnen Sie den OB 1 über die Schaltfläche [OB 1 anzeigen].
- 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.

- Öffnen Sie die *AG-Maske* über **Anzeige** > *AG-Maske*. Eine CPU wird grafisch dargestellt.
- Durch Betätigung der rechten Maustaste innerhalb der *AG-Maske* öffnet sich das Kontextmenü. Wählen Sie aus dem Kontextmenü für unser Beispiel "Digitale Ausgabebaugruppe (16Bit) einfügen". Die Baugruppe wird rechts von der CPU dargestellt.
- Öffnen Sie durch Doppelklick auf die Ausgabebaugruppe den Eigenschaften-Dialog und stellen Sie die *Baugruppenadresse* 124 ein.
- Schalten Sie mit der Maus den Betriebsarten-Schalter in RUN. Ihr Programm wird im Simulator ausgeführt und dargestellt.



SPS-Programm in CPU übertragen und ausführen

- Zur Übertragung in Ihre CPU stellen Sie "Ziel: TCP/IP-Direkt" ein.
- Zur Vorgabe der Ethernet-Daten klicken Sie auf [...] und klicken Sie auf [Erreichbare Teilnehmer].
- Wählen Sie Ihre Netzwerkkarte aus und klicken Sie auf [Teilnehmer ermitteln]. Nach einer gewissen Wartezeit werden alle verfügbaren Teilnehmer aufgelistet.
- Wählen Sie Ihre CPU aus, die Sie über die Hardware-Konfiguration mit TCP/IP-Adress-Parametern schon versorgt haben und klicken Sie auf [übernehmen].
- Schließen Sie den "Ethernet-Daten"-Dialog mit [OK].
- Übertragen Sie die Bausteine in Ihre CPU mit **AG** > *Alle Bausteine senden*.
- Schalten Sie Ihre CPU in RUN.
- Öffnen Sie den OB 1 über die Schaltfläche [OB 1 anzeigen]
- 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.

Anhang

A Index

3	
3964R	6-15
A	
Aderquerschnitt	1-7
Adressbelegung	4-6, 5-5, 5-7
Adressierung	4-4
automatisch	4-5
Beispiel	4-5
Anlaufverhalten.....	4-3
ASCII	6-14
Aufbau	3-3
Aufbau Richtlinien	2-1, 2-8
B	
Batteriepufferung	4-3
Betriebsart	
Schalter.....	3-4
Betriebszustände.....	4-23
D	
Diagnose	
Puffer	4-38
Zähler.....	5-31
Digital-Teil.....	3-6, 5-3, 5-6
Adressbelegung.....	5-7
E	
Ein-/Ausgabe-Bereich	3-6, 5-3
Einsatz	
CPU 312SC	4-1
E/A-Peripherie	5-1
PtP-Kommunikation.....	6-1
EMV	2-8
Grundregeln.....	2-9
Ereignis ID	4-38
ESD	1-7
Ethernet-PG/OP-Kanal.....	4-7, 4-22
F	
Firmware	
Info über Baugruppenzustand	4-29
Info über Web-Seite	4-29
übertragen	4-31
Update	4-28
G	
Grundlagen.....	1-1
H	
Haltepunkte	4-24
Hardwarebeschreibung.....	3-1
K	
Know-how-Schutz.....	4-34
Kommunikation	
PG/OP	4-7, 4-22
RS485	3-5
PtP	3-5
Kompatibilität.....	1-8
L	
LEDs	3-3
M	
MCC	3-4, 4-33
MMC.....	4-20
Projekttransfer.....	4-20
Diagnose.....	4-21
MMC-Cmd	
Autobefehle	4-36
Modbus	6-17
Beispiel	6-22
Funktionscodes	6-18
Slave-Antwort	6-19
Telegramm	6-18
Montage	2-1
Ausrichtung.....	2-4, 4-2
Vorgehensweise	2-4, 4-2
MPI.....	4-19
P	
Parametrierung	
CPU 312SC	4-14
Digital-Teil	5-8
Module.....	4-18
Zähler	5-12
PG/OP-Kanal	4-7, 4-22
pkg-Dateien.....	4-28
Profilschiene	2-2
Projektierung.....	4-11
Buserweiterung	4-4
Schnelleinstieg	4-11
Transfer	4-19
Prozeduren.....	6-15
Prozessabbild	4-4
PtP-Kommunikation.....	6-1
3964R	6-15
ASCII	6-14
Broadcast	6-16
Fehlermeldungen	6-7, 6-9, 6-12
Kommunikation	6-8

Modbus	6-17	USS	6-16
Parametrierung	6-5	V	
Prinzip Datenübertragung	6-3	Verdrahtung	2-5
Protokolle	6-14	Frontstecker	2-6
RS485-Schnittstelle	6-4	W	
Schnelleinstieg	6-2	Watchdog	4-25
SFCs	6-2, 6-8	Web-Seite	4-7, 4-10
STX/ETX	6-14	Werkseinstellung	4-27, 4-32
USS	6-16	WinPLC7	7-1
R		wld-Dateien	4-20, 4-34
Rückwandbus-Verbinder	2-2	Z	
S		Zähler	5-9
Schirmung von Leitungen	2-10	Adressbelegung	5-7
Schnittstellen	3-5	Auftragsschnittstelle	5-16
Ethernet-PG/OP	3-4	Diagnose	5-31
MPI	3-5	Alarm	5-32
RS485	3-5	Digital-Ausgang	5-27
PtP	3-5, 6-1	einmalig zählen	5-20
Schockfestigkeit	1-7	endlos zählen	5-19
Sicherheitshinweise	1-2	Endwert	5-18
Spannungsversorgung	1-8, 3-4	Funktionen	5-18
Speichererweiterung	4-33	Hauptzählrichtung	5-18
Speichermanagement	3-4	Hysterese	5-28
Steckerbelegung	3-7, 5-4	Ladewert	5-18
Störeinträge	2-8	Latch-Funktion	5-17, 5-26
STX/ETX	6-14	periodisch zählen	5-22
T		Prozessalarm	5-31
Technische Daten	3-8	Schnelleinstieg	5-9
Testfunktionen	4-42	SFB 47	5-14
Ü		Steckerbelegung	5-9
Übersicht		Steuern	5-14
CPU 312SC	1-7	Tor-Funktion	5-25
System 300	1-3	Vergleicher	5-27
U		Wertebereiche	5-17
Umgebungsbedingungen	1-8	Zusatzfunktionen	5-24
Urlöschen	4-26	Zielsystemfunktionen	4-43
Werkseinstellung	4-27	Zykluszeitüberwachung	4-25