

# VIPA SPEED7 Library

OPL\_SP7-LIB | SW90FS0MA V10.001 | Handbuch

HB00 | OPL\_SP7-LIB | SW90FS0MA V10.001 | de | 18-28

Baustein Bibliothek - Network Communication



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

## Inhaltsverzeichnis

<b>1</b>	<b>Allgemeines</b>	<b>4</b>
1.1	Copyright © VIPA GmbH	4
1.2	Über dieses Handbuch	5
<b>2</b>	<b>Wichtige Hinweise</b>	<b>6</b>
2.1	Allgemein	6
2.2	Intern verwendete Bausteine	6
<b>3</b>	<b>Bibliothek einbinden</b>	<b>7</b>
3.1	Einbinden in Siemens SIMATIC Manager	7
3.2	Einbinden in Siemens TIA Portal	8
<b>4</b>	<b>Bausteinparameter</b>	<b>9</b>
4.1	Allgemeine und spezifische Fehlercodes RET_VAL	9
<b>5</b>	<b>Netzwerkkommunikation - "Network Communication"</b>	<b>12</b>
5.1	Offene Kommunikation - "Open Communication"	12
5.1.1	Verbindungsorientierte Protokolle	12
5.1.2	Verbindungslose Protokolle	12
5.1.3	FB 63 - TSEND - Daten senden - TCP native und ISO on TCP	13
5.1.4	FB 64 - TRCV - Daten empfangen - TCP native und ISO on TCP	16
5.1.5	FB 65 - TCON - Verbindung aufbauen	20
5.1.6	UDT 65 - TCON_PAR - Datenstruktur für FB 65	22
5.1.7	FB 66 - TDISCON - Verbindung abbauen	27
5.1.8	FB 67 - TUSEND - Daten senden - UDP	29
5.1.9	FB 68 - TURCV - Daten empfangen - UDP	32
5.1.10	UDT 66 - TADDR_PAR - Datenstruktur	34
5.2	Ethernet-Kommunikation - "Ethernet Communication"	35
5.2.1	Kommunikation - FC 5...6 für CP 343	35
5.2.2	FC 5 - AG_SEND - Senden an CP 343	38
5.2.3	FC 6 - AG_RECV - Empfangen von CP 343	41
5.2.4	FC 10 - AG_CNTRL - Control CP 343	43
5.2.5	FC 62 - C_CNTR - Zustand einer Verbindung abfragen	51
5.2.6	FB/SFB 8 - FB 55 - Übersicht	52
5.2.7	FB/SFB 8 - USEND - Unkoordiniertes Senden	53
5.2.8	FB/SFB 9 - URCV - Unkoordiniertes Empfangen	55
5.2.9	FB/SFB 12 - BSEND - Blockorientiertes Senden	57
5.2.10	FB/SFB 13 - BRCV - Blockorientiertes Empfangen	60
5.2.11	FB/SFB 14 - GET - Remote CPU lesen	62
5.2.12	FB/SFB 15 - PUT - Remote CPU schreiben	64
5.2.13	FB 55 - IP_CONF - Progr. Kommunikationsverbindungen	66

# 1 Allgemeines

## 1.1 Copyright © VIPA GmbH

### All Rights Reserved

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

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

Zur Genehmigung von Vervielfältigung oder Verteilung wenden Sie sich bitte an: VIPA, Gesellschaft für Visualisierung und Prozessautomatisierung mbH Ohmstraße 4, D-91074 Herzogenaurach, Germany

Tel.: +49 9132 744 -0

Fax.: +49 9132 744-1864

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

<http://www.vipa.com>



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

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

### EG-Konformitätserklärung

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

### Informationen zur Konformitätserklärung

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

### Warenzeichen

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

SPEED7 ist ein eingetragenes Warenzeichen der profichip GmbH.

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

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

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

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

**Dokument-Support**

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

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefax: +49 9132 744-1204

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

**Technischer Support**

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

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Telefon: +49 9132 744-1150 (Hotline)

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

## 1.2 Über dieses Handbuch

**Zielsetzung und Inhalt**

Das Handbuch beschreibt die Baustein-Bibliothek *"Network Communication"* von VIPA:

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

**Piktogramme Signalwörter**

Besonders wichtige Textteile sind mit folgenden Piktogrammen und Signalworten ausgezeichnet:

**GEFAHR!**

Unmittelbar drohende oder mögliche Gefahr. Personenschäden sind möglich.

**VORSICHT!**

Bei Nichtbefolgen sind Sachschäden möglich.



*Zusätzliche Informationen und nützliche Tipps.*

## 2 Wichtige Hinweise

### 2.1 Allgemein



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

### 2.2 Intern verwendete Bausteine



#### **VORSICHT!**

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

FC/SFC	Bezeichnung	Beschreibung
FC/SFC 192	CP_S_R	wird intern für FB 7 und FB 8 verwendet
FC/SFC 196	AG_CNTRL	wird intern für FC 10 verwendet
FC/SFC 200	AG_GET	wird intern für FB/SFB 14 verwendet
FC/SFC 201	AG_PUT	wird intern für FB/SFB 15 verwendet
FC/SFC 202	AG_BSEND	wird intern für FB/SFB 12 verwendet
FC/SFC 203	AG_BRCV	wird intern für FB/SFB 13 verwendet
FC/SFC 204	IP_CONF	wird intern für FB 55 IP_CONF verwendet
FC/SFC 205	AG_SEND	wird intern für FC 5 AG_SEND verwendet
FC/SFC 206	AG_RECV	wird intern für FC 6 AG_RECV verwendet
FC/SFC 253	IBS_ACCESS	wird intern für SPEED-Bus-INTERBUS-Master verwendet
SFB 238	EC_RWOD	wird intern für EtherCAT-Kommunikation verwendet
SFB 239	FUNC	wird intern für FB 240, FB 241 verwendet

### 3 Bibliothek einbinden

#### Baustein-Bibliothek "Network Communication"

Die Baustein-Bibliothek finden Sie im "Service/Support"-Bereich auf [www.vipa.com](http://www.vipa.com) unter "Downloads → VIPA Lib" als "Baustein-Bibliothek Network Communication - SW90FS0MA" zum Download. Die Bibliothek liegt als gepackte zip-Dateien vor. Sobald Sie die Bausteine verwenden möchten, müssen Sie diese in Ihr Projekt importieren.




#### Folgende Bausteinbibliotheken stehen zur Verfügung

Datei	Beschreibung
NetworkCom_S7_V0001.zip	<ul style="list-style-type: none"> <li>■ Bausteinbibliothek für Siemens SIMATIC Manager.</li> <li>■ Für den Einsatz in CPUs von VIPA bzw. S7-300 CPUs von Siemens.</li> </ul>
NetworkCom_TIA_V0001.zip	<ul style="list-style-type: none"> <li>■ Bausteinbibliothek für Siemens TIA Portal V14.</li> <li>■ Für den Einsatz in CPUs von VIPA bzw. S7-300 CPUs von Siemens.</li> </ul>


### 3.1 Einbinden in Siemens SIMATIC Manager

#### Übersicht






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

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



#### ZIP-Datei laden

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

#### Bibliothek dearchivieren

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

#### Bibliothek öffnen und Bausteine in Projekt über- tragen

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



Werden anstelle der SFCs FCs verwendet, so werden diese von den VIPA CPUs ab Firmware 3.6.0 unterstützt.

## 3.2 Einbinden in Siemens TIA Portal

### Übersicht

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

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

### ZIP-Datei laden

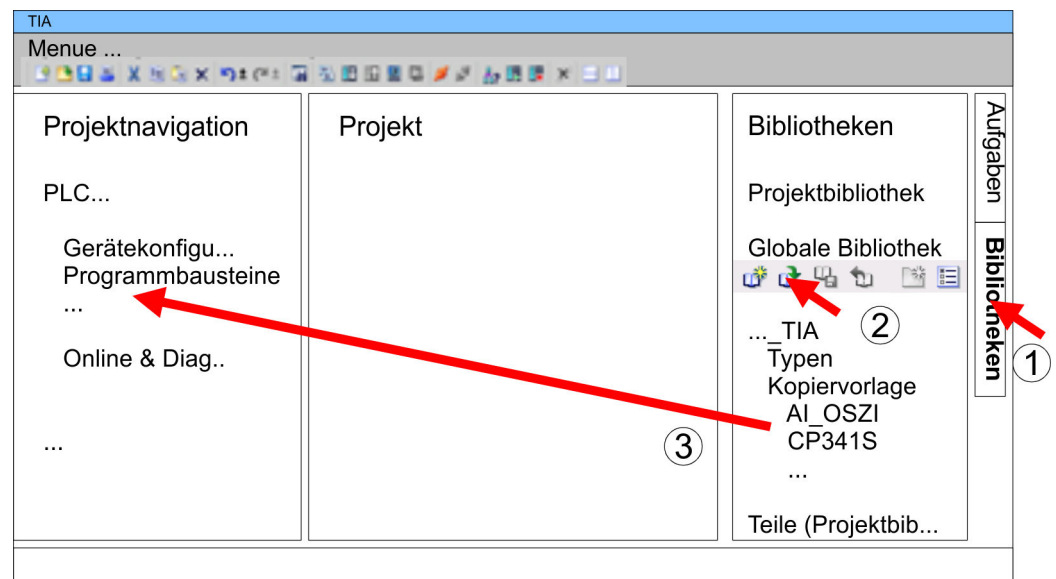
1. Navigieren Sie auf der Webseite zu der ZIP-Datei, welche zu Ihrer Programmversion passt.
2. Laden und speichern Sie diese in Ihrem Arbeitsverzeichnis.

### ZIP-Datei entpacken

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

### Bibliothek öffnen und Bausteine in Projekt übertragen

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



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



## 4 Bausteinparameter

### 4.1 Allgemeine und spezifische Fehlercodes RET\_VAL

#### Übersicht

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

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

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

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

#### RET\_VAL (Rückgabewert)

Die folgende Tabelle zeigt den Aufbau eines Fehlercodes:

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

#### Spezifischer Fehlercode

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

Ein spezifischer Fehlercode besteht aus:

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

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

**Allgemeine Fehlercodes  
RET\_VAL**

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

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

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

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

**Allgemeine Fehlercodes**

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

Fehlercode	Beschreibung
8x7Fh	Interner Fehler. Dieser Fehlercode zeigt einen internen Fehler am Parameter x an. Dieser Fehler wurde nicht vom Anwender verursacht und kann von ihm auch nicht behoben werden.
8x01h	Unzulässige Syntaxkennung bei einem ANY-Parameter.
8x22h	Bereichslängenfehler beim Lesen eines Parameters.
8x23h	Bereichslängenfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x vollständig oder teilweise außerhalb des Operandenbereichs befindet oder die Länge eines Bitfeldes bei einem ANY-Parameter nicht durch 8 teilbar ist.
8x24h	Bereichsfehler beim Lesen eines Parameters.
8x25h	Bereichsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass sich der Parameter x in einem Bereich befindet, der für die Systemfunktion unzulässig ist. Die Beschreibung der jeweiligen Funktion gibt die Bereiche an, die für die Funktion unzulässig sind.
8x26h	Der Parameter enthält eine zu große Nummer einer Zeitzelle. Dieser Fehlercode zeigt an, dass die Zeitzelle, die in Parameter x angegeben wird, nicht vorhanden ist.
8x27h	Der Parameter enthält eine zu große Nummer einer Zählerzelle (Nummernfehler des Zählers). Dieser Fehlercode zeigt an, dass die Zählerzelle, die in Parameter x angegeben wird, nicht vorhanden ist.
8x28h	Ausrichtungsfehler beim Lesen eines Parameters.
8x29h	Ausrichtungsfehler beim Schreiben eines Parameters. Dieser Fehlercode zeigt an, dass der Verweis auf den Parameter x ein Operand ist, dessen Bitadresse ungleich 0 ist.
8x30h	Der Parameter befindet sich in dem schreibgeschützten Global-DB.
8x31h	Der Parameter befindet sich in dem schreibgeschützten Instanz-DB. Dieser Fehlercode zeigt an, dass der Parameter x sich in einem schreibgeschützten Datenbaustein befindet. Wenn der Datenbaustein von der Systemfunktion selbst geöffnet wurde, gibt die Systemfunktion immer den Wert 8x30h aus.
8x32h	Der Parameter enthält eine zu große DB-Nummer (Nummernfehler des DBs).

Fehlercode	Beschreibung
8x34h	Der Parameter enthält eine zu große FC-Nummer (Nummernfehler des FCs).
8x35h	Der Parameter enthält eine zu große FB-Nummer (Nummernfehler des FBs). Dieser Fehlercode zeigt an, dass der Parameter x eine Bausteinnummer enthält, die größer ist als die maximal zulässige Bausteinnummer.
8x3Ah	Der Parameter enthält die Nummer eines DBs, der nicht geladen ist.
8x3Ch	Der Parameter enthält die Nummer eines FCs, der nicht geladen ist.
8x3Eh	Der Parameter enthält die Nummer eines FBs, der nicht geladen ist.
8x42h	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter aus dem Peripheriebereich der Eingänge auslesen wollte.
8x43h	Es ist ein Zugriffsfehler aufgetreten, während das System einen Parameter in den Peripheriebereich der Ausgänge schreiben wollte.
8x44h	Fehler beim n-ten ( $n > 1$ ) Lesezugriff nach Auftreten eines Fehlers.
8x45h	Fehler beim n-ten ( $n > 1$ ) Schreibzugriff nach Auftreten eines Fehlers. Dieser Fehlercode zeigt an, dass der Zugriff auf den gewünschten Parameter verweigert wird.

## 5 Netzwerkcommunication - "Network Communication"

### 5.1 Offene Kommunikation - "Open Communication"

#### 5.1.1 Verbindungsorientierte Protokolle

- Verbindungsorientierte Protokolle bauen vor der Datenübertragung eine (logische) Verbindung zum Kommunikationspartner auf und bauen diese nach Abschluss der Datenübertragung ggf. wieder ab.
- Verbindungsorientierte Protokolle werden eingesetzt, wenn es bei der Datenübertragung insbesondere auf Sicherheit ankommt. Auch wird hier die richtige Reihenfolge der empfangenen Pakete gewährleistet.
- Über eine physikalische Leitung können in der Regel mehrere logische Verbindungen bestehen.
- Bei den FBs zur Offenen Kommunikation über Industrial Ethernet werden die folgenden verbindungsorientierten Protokolle unterstützt:
  - TCP native gemäß RFC 793 (Verbindungstypen 01h und 11h)
  - ISO on TCP gemäß RFC 1006 (Verbindungstyp 12h)

#### TCP native

- Bei der Datenübertragung über TCP nativ werden weder Informationen zur Länge noch über Anfang und Ende einer Nachricht übertragen. Auch besteht keine Möglichkeit zu erkennen, wo ein Datenstrom endet und der nächste beginnt.
- Die Übertragung ist stream-orientiert. Aus diesem Grund sollten Sie in den FBs bei Sender und Empfänger identische Datenlängen angeben.
- Falls die empfangene Anzahl der Daten von der parametrisierten Länge abweicht, erhalten Sie entweder Daten, welche nicht die vollständigen Telegrammdaten enthalten oder mit dem Inhalt eines nachfolgenden Telegramms aufgefüllt sind.
- Der Empfangsbaustein kopiert so viele Bytes in den Empfangsbereich, wie Sie als Länge parametrisiert haben. Anschließend setzt er NDR auf TRUE und beschreibt RCVD\_LEN mit dem Wert von LEN. Mit jedem weiteren Aufruf erhalten Sie damit einen weiteren Block der gesendeten Daten.

#### ISO on TCP

- Bei der Datenübertragung werden Informationen zur Länge und zum Ende einer Nachricht übertragen. Die Übertragung ist blockorientiert.
- Falls Sie die Länge der zu empfangenden Daten größer gewählt haben als die Länge der gesendeten Daten, kopiert der Empfangsbaustein die gesendeten Daten vollständig in den Empfangsdatenbereich. Anschließend setzt er NDR auf TRUE und beschreibt RCVD\_LEN mit der Länge der gesendeten Daten.
- Falls Sie die Länge der zu empfangenden Daten kleiner gewählt haben als die Länge der gesendeten Daten, kopiert der Empfangsbaustein keine Daten in den Empfangsdatenbereich, sondern liefert folgende Fehlerinformation: ERROR = 1, STATUS = 8088h.

#### 5.1.2 Verbindungslose Protokolle

Bei den verbindungslosen Protokollen entfallen Verbindungsauf- und Verbindungsabbau zum remoten Partner. Verbindungslose Protokolle übertragen die Daten unquittiert und damit ungesichert zum remoten Partner. Bei den FBs zur Offenen Kommunikation über Industrial Ethernet wird das folgende verbindungslose Protokoll unterstützt:

- UDP gemäß RFC 768 (Verbindungstyp 13h)

#### UDP

- Bei Aufruf des Sendebausteins ist ein Verweis auf die Adressparameter des Empfängers (IP-Adresse und Port-Nr.) anzugeben. Auch werden Informationen zur Länge und zum Ende einer Nachricht übertragen.
- Analog erhalten Sie nach Abschluss des Empfangsbausteins einen Verweis auf die Adressparameter des Senders (IP-Adresse und Port-Nr.).

- Damit sie Sende- und Empfangsbaustein nutzen können, müssen Sie zuvor sowohl auf der Sender- als auch auf der Empfängerseite einen lokalen Kommunikationszugangspunkt einrichten.
- Bei jedem Sendauftrag können Sie den remoten Partner durch Angabe seiner IP-Adresse und seiner Port-Nr. neu referenzieren.
- Falls Sie die Länge der zu empfangenden Daten größer gewählt haben als die Länge der gesendeten Daten, kopiert der Empfangsbaustein die gesendeten Daten vollständig in den Empfangsdatenbereich. Anschließend setzt er NDR auf TRUE und beschreibt RCVD\_LEN mit der Länge der gesendeten Daten.
- Falls Sie die Länge der zu empfangenden Daten kleiner gewählt haben als die Länge der gesendeten Daten, kopiert der Empfangsbaustein keine Daten in den Empfangsdatenbereich, sondern liefert folgende Fehlerinformation: ERROR = 1, STATUS = 8088h.

### 5.1.3 FB 63 - TSEND - Daten senden - TCP native und ISO on TCP

#### Beschreibung

- Der FB 63 TSEND sendet Daten über eine bestehende Kommunikationsverbindung. Er ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe.
- Sie starten den Sendevorgang, indem Sie den FB 63 mit *REQ* = 1 aufrufen.
- Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs).
- In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *DONE* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 63 aktuell befindet bzw. wann der Verbindungsaufbau beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.



*Durch die asynchrone Arbeitsweise des FB 63 TSEND müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter DONE oder der Parameter ERROR den Wert TRUE annimmt.*

# Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter <i>REQ</i> , aktiviert den Sendevorgang der durch <i>ID</i> gegebenen Verbindung bei steigender Flanke. Beim erstmaligen Aufruf mit <i>REQ</i> = 1 werden Daten aus dem mit Parameter <i>DATA</i> angegebenen Bereich übergeben.
ID	INPUT	WORD	M, D, Konstante	Referenz auf die zugehörige Verbindung. <i>ID</i> muss identisch sein mit dem zugehörigen Parameter <i>ID</i> in der lokalen Verbindungsbeschreibung.  Wertebereich: 0001h ... 0FFFh
LEN	INPUT	INT	E, A, M, D, L	Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen Wertebereich: <ul style="list-style-type: none"> <li>■ 1 ... 1460, falls Verbindungstyp = 01h</li> <li>■ 1 ... 8192, falls Verbindungstyp = 11h</li> <li>■ 1 ... 1452, falls Verbindungstyp = 12h und ein CP benutzt wird</li> <li>■ 1 ... 8192, falls Verbindungstyp = 12h und kein CP benutzt wird.</li> </ul>
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>■ 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>■ 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <i>BUSY</i> = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li>■ <i>BUSY</i> = 0: Der Auftrag ist beendet.</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Statusinformationen
DATA	IN_OUT	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge. Die Adresse verweist auf: <ul style="list-style-type: none"> <li>■ das Prozessabbild der Eingänge</li> <li>■ das Prozessabbild der Ausgänge</li> <li>■ einen Merker</li> <li>■ einen Datenbaustein</li> </ul> Zulässige referenzierte Datentypen: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

## Statusinformationen

ERROR	STATUS	Beschreibung
0	0000h	Sendeauftrag wurde ohne Fehler abgeschlossen.
0	7000h	Erstaufruf mit <i>REQ</i> = 0, kein Anstoß zum Senden.
0	7001h	Erstaufruf mit <i>REQ</i> = 1, Anstoß des Sendevorgangs.
0	7002h	Zwischenaufwurf ( <i>REQ</i> irrelevant), Auftrag ist in Bearbeitung <b>Hinweis:</b> In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendepuffer <i>DATA</i> zu.
1	8085h	Parameter <i>LEN</i> hat den Wert 0 oder ist größer als der größte zulässige Wert.
1	8086h	Parameter <i>ID</i> liegt in einem unzulässigen Wertebereich.
0	8088h	Parameter <i>LEN</i> ist größer als der in <i>DATA</i> angegebene Speicherbereich.
1	80A1h	Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Zu der angegebenen <i>ID</i> wurde noch kein FB 65 TCON aufgerufen.</li> <li>■ Die angegebene Verbindung wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich.</li> <li>■ Schnittstelle wird neu initialisiert.</li> </ul>
1	80B3h	Der parametrisierte Verbindungstyp (Parameter <i>connection_type</i> in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie den FB 67 TUSEND.
1	80C3h	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
1	80C4h	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Die Verbindung zum Kommunikationspartner kann momentan nicht aufgebaut werden.</li> <li>■ Die Schnittstelle wird neu parametrisiert.</li> </ul>
1	8822h	Parameter <i>DATA</i> : Quellbereich ungültig, Bereich im DB nicht vorhanden.
1	8824h	Parameter <i>DATA</i> : Bereichsfehler im ANY-Pointer.
1	8832h	Parameter <i>DATA</i> : DB-Nummer ist zu groß.
1	883Ah	Parameter <i>DATA</i> : Zugriff auf Sendepuffer nicht möglich (z.B. gelöschter DB).
1	887Fh	Parameter <i>DATA</i> : Interner Fehler, z.B. unzulässige ANY-Referenz.
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation ↪ Kapitel 4.1 "Allgemeine und spezifische Fehlercodes RET_VAL" auf Seite 9

### 5.1.4 FB 64 - TRCV - Daten empfangen - TCP native und ISO on TCP

#### Beschreibung

Der FB 64 TRCV empfängt Daten über eine bestehende Kommunikationsverbindung. Für den Empfang und die Weiterverarbeitung der Daten gibt es folgende Varianten:

- Variante 1: Empfangenen Datenblock sofort weiterverarbeiten.
- Variante 2: Empfangene Datenblöcke in einem Empfangspuffer speichern und erst dann weiterverarbeiten, wenn dieser gefüllt ist.

Der Zusammenhang zwischen Verbindungstyp und den beiden Varianten ist in folgender Tabelle dargestellt.

Verbindungstyp	Variante
01h und 11h	Sie können die Variante selbst bestimmen.
12h	Variante 2 (fix)

Die beiden Varianten sind in der folgenden Tabelle näher beschrieben.

Empfangene Daten ...	Wertebereich von <i>LEN</i>	Wertebereich von <i>RCVD_LEN</i>	Beschreibung
stehen sofort zur Verfügung.	0	1 ... x	<p>Sie übergeben einen Puffer, dessen Länge x im ANY-Pointer des Empfangspuffers hinterlegt ist (Parameter <i>DATA</i>).</p> <p>Nachdem ein Datenblock empfangen wurde, wird dieser sofort im Empfangspuffer zur Verfügung gestellt.</p> <p>Die Anzahl der empfangenen Daten (Parameter <i>RCVD_LEN</i>) kann maximal so groß sein wie die im Parameter <i>DATA</i> hinterlegte Größe. Der Empfang wird angezeigt durch <i>NDR</i> = 1.</p>
im Empfangspuffer speichern. Sie stehen zur Verfügung, sobald die projektierte Länge erreicht wird.	<p>1 ... 1460, falls Verbindungstyp = 01h</p> <p>1 ... 8192, falls Verbindungstyp = 11h</p> <p>1 ... 1452, falls Verbindungstyp = 12h und ein CP benutzt wird</p> <p>1 ... 8192, falls Verbindungstyp = 12h und kein CP benutzt wird</p>	gleicher Wert wie im Parameter <i>LEN</i>	<p>Sie übergeben die Empfangslänge am Parameter <i>LEN</i>. Wenn diese parametrisierte Länge erreicht ist, werden die Empfangsdaten im Parameter <i>DATA</i> zur Verfügung gestellt (<i>NDR</i> = 1).</p>



**Arbeitsweise**

- Der FB 64 TRCV ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe. Sie starten den Empfangsvorgang, indem Sie den FB 64 mit *REQ* = 1 aufrufen.
- Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs).
- In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *NDR* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 64 TRCV aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.



Durch die asynchrone Arbeitsweise des FB 64 TRCV sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter *NDR* den Wert *TRUE* annimmt.

**Parameter**

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	INPUT	BOOL	E, A, M, D, L	Mit EN_R = 1 wird der FB 64 TRCV empfangsbereit (Steuerparameter). Der Empfangsauftrag wird bearbeitet.
ID	INPUT	WORD	M, D, Konstante	Referenz auf die zugehörige Verbindung. <i>ID</i> muss identisch sein mit dem zugehörigen Parameter <i>id</i> in der lokalen Verbindungsbeschreibung.  Wertebereich: 0001h ... 0FFFh

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
LEN	INPUT	INT	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <math>LEN = 0</math> (Ad-Hoc-Mode) empfohlen bei ISO on TCP: implizite Längenangabe im ANY-Pointer DATA verwenden. Die empfangenen Daten werden beim Bausteinaufruf unmittelbar bereitgestellt. Die Anzahl der empfangenen Daten steht in RCVD_LEN zur Verfügung.</li> <li>■ <math>1 \leq LEN \leq \text{max}</math>: Anzahl der Bytes, die empfangen werden sollen. Die Anzahl der tatsächlich empfangenen Daten steht in RCVD_LEN zur Verfügung. Die Daten stehen Ihnen dann zur Verfügung, wenn sie vollständig empfangen wurden. max hängt vom Verbindungstyp ab: <ul style="list-style-type: none"> <li>– max = 1460 beim Verbindungstyp 01h</li> <li>– max = 8192 beim Verbindungstyp 11h</li> <li>– max = 1452 beim Verbindungstyp 12h mit CP-Einsatz</li> <li>– max = 8192 beim Verbindungstyp 12h ohne CP-Einsatz</li> </ul> </li> </ul>
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>NDR</i> : <ul style="list-style-type: none"> <li>■ <math>NDR = 0</math>: Auftrag wurde noch nicht gestartet oder läuft noch</li> <li>■ <math>NDR = 1</math>: Auftrag wurde erfolgreich abgeschlossen</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <math>ERROR = 1</math>: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <math>BUSY = 1</math>: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li>■ <math>BUSY = 0</math>: Der Auftrag ist beendet.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Statusinformationen
RCVD_LEN	OUTPUT	INT	E, A, M, D, L	Anzahl der tatsächlich empfangenen Daten in Bytes
DATA	IN_OUT	ANY	E, A, M, D	Empfangsbereich (Adresse und Länge). Die Adresse verweist auf: <ul style="list-style-type: none"> <li>■ das Prozessabbild der Eingänge</li> <li>■ das Prozessabbild der Ausgänge</li> <li>■ einen Merker</li> <li>■ einen Datenbaustein</li> </ul> Zulässige referenzierte Datentypen: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

## Statusinformationen

ERROR	STATUS	Beschreibung
0	0000h	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in <i>RCVD_LEN</i> angezeigt.
0	7000h	Erstaufruf mit <i>REQ</i> = 0, kein Anstoß zum Empfangen
0	7001h	Baustein ist empfangsbereit. Empfangsauftrag wurde aktiviert.
0	7002h	Zwischenaufruf, Auftrag ist in Bearbeitung <b>Hinweis:</b> In dieser Bearbeitungsphase schreibt das Betriebssystem Daten in den Empfangspuffer <i>DATA</i> . Deshalb können im Fehlerfall inkonsistente Daten im Empfangspuffer stehen.
1	8085h	Parameter <i>LEN</i> ist größer als der größte zulässige Wert, oder Sie haben den Wert von <i>LEN</i> gegenüber dem Erstaufruf geändert
1	8086h	Parameter <i>ID</i> liegt in einem unzulässigen Wertebereich
1	8088h	<ul style="list-style-type: none"> <li>■ Zielpuffer (<i>DATA</i>) ist zu klein</li> </ul> Wert in <i>LEN</i> ist größer als der durch <i>DATA</i> vorgegebene Empfangsbereich. Fehlerbehebung falls Verbindungstyp = 12h: Vergrößern Sie den Zielpuffer <i>DATA</i> .
1	80A1h	Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Zu der angegebenen <i>ID</i> wurde noch kein FB 65 TCON aufgerufen</li> <li>■ Die angegebene Verbindung wird momentan abgebaut. Ein Empfangsvorgang über diese Verbindung ist nicht möglich.</li> <li>■ Die Schnittstelle wird neu parametriert.</li> </ul>
1	80B3h	Der parametrierte Verbindungstyp (Parameter <i>connection_type</i> in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie den FB 68 TURCV.
1	80C3h	Die Betriebsmittel (Speicher) der CPU sind temporär belegt
1	80C4h	Temporärer Kommunikationsfehler: Die Verbindung wird gerade aufgebaut.
1	8922h	Parameter <i>DATA</i> : Zielbereich ungültig, Bereich im DB nicht vorhanden
1	8924h	Parameter <i>DATA</i> : Bereichsfehler im ANY-Pointer
1	8932h	Parameter <i>DATA</i> : DB-Nummer ist zu groß
1	893Ah	Parameter <i>DATA</i> : Zugriff auf Empfangspuffer nicht möglich (z.B. gelöschter DB)
1	897Fh	Parameter <i>DATA</i> : Interner Fehler, z.B. unzulässige ANY-Referenz
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation → Kapitel 4.1 "Allgemeine und spezifische Fehlercodes <i>RET_VAL</i> " auf Seite 9

### 5.1.5 FB 65 - TCON - Verbindung aufbauen

#### Verwendung bei TCP native und ISO on TCP

Beide Kommunikationspartner rufen den FB 65 TCON zum Aufbau der Kommunikationsverbindung auf. In der Parametrierung hinterlegen Sie, welcher der aktive und welcher der passive Kommunikationsendpunkt ist. Die Anzahl der möglichen Verbindungen entnehmen Sie den Technischen Daten Ihrer CPU. Nach dem Aufbau der Verbindung wird diese automatisch von der CPU überwacht und gehalten. Bei Verbindungsabbruch durch z.B. Leitungsunterbrechung oder durch den remoten Kommunikationspartner versucht der aktive Partner die Verbindung wieder aufzubauen. Sie müssen den FB 65 TCON nicht erneut aufrufen. Mit dem Aufruf des FB 66 TDISCON oder im Betriebszustand STOP der CPU wird eine bestehende Verbindung abgebrochen. Zum erneuten Aufbau der Verbindung müssen Sie den FB 65 TCON nochmals aufrufen.

#### Verwendung bei UDP

Beide Kommunikationspartner rufen den FB 65 TCON auf, um ihren lokalen Kommunikationszugangspunkt einzurichten. Dabei wird eine Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems eingerichtet. Es erfolgt kein Verbindungsaufbau zum remoten Partner. Der lokale Zugangspunkt wird zum Senden und Empfangen von UDP Telegrammen verwendet.

#### Beschreibung

Der FB 65 TCON ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe. Sie starten den Verbindungsaufbau, indem Sie den FB 65 mit *REQ* = 1 aufrufen. Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs). In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *DONE* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 65 aktuell befindet bzw. wann der Verbindungsaufbau beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

#### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter <i>REQ</i> , aktiviert den Verbindungsaufbau bei steigender Flanke
ID	INPUT	WORD	M, D, Konstante	Referenz auf die aufzubauende Verbindung zum remoten Partner bzw. zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: 0001h ... 0FFFh
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <i>BUSY</i> = 1: Der Auftrag ist noch nicht beendet.</li> <li>■ <i>BUSY</i> = 0: Der Auftrag ist beendet.</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Statusinformationen
CONNECT	IN_OUT	ANY	D	Zeiger auf die zugehörige Verbindungsbeschreibung. ↪ Kapitel 5.1.6 "UDT 65 - TCON_PAR - Datenstruktur für FB 65" auf Seite 22

## Statusinformationen

ERROR	STATUS	Beschreibung
0	0000h	Verbindung konnte aufgebaut werden
0	7000h	Aufruf mit <i>REQ</i> = 0, kein Anstoß zum Verbindungsaufbau
0	7001h	Erstaufbau mit <i>REQ</i> = 1, Verbindung wird aufgebaut
0	7002h	Zwischenaufbau ( <i>REQ</i> irrelevant), Verbindung wird aufgebaut
1	8086h	Der Parameter ID darf nicht den Wert Null haben.
0	8087h	Maximale Anzahl der Verbindungen ist erreicht, keine weitere Verbindung möglich.
1	8089h	Der Parameter <i>CONNECT</i> zeigt nicht auf einen Datenbaustein.
1	809Ah	Der Parameter <i>CONNECT</i> zeigt auf ein Feld, das nicht die Länge der Verbindungsbeschreibung (UDT 65) hat.
1	809Bh	Die über <i>local_device_id</i> und <i>next_staddr</i> angegebene Kommunikationsschnittstelle wird von der CPU nicht unterstützt.
1	80A1h	Verbindung bzw. Port ist bereits vom Anwender belegt.
1	80A2h	Lokaler oder remoter Port ist vom System belegt.
1	80A3h	Es wird versucht, eine bestehende Verbindung erneut aufzubauen.
1	80A4h	IP-Adresse des remoten Verbindungsendpunkts ist ungültig.
1	80A7h	Kommunikationsfehler: Sie haben nach einem nicht abgeschlossenen TCON einen TDISCON aufgerufen. Der Verbindungsaufbau wurde durch Aufruf eines TDISCON vorzeitig abgebrochen.
1	80B4h	Beim Protokoll ISO on TCP sind beim passiven Verbindungsaufbau eine oder mehrere der folgenden Bedingungen verletzt worden: <ul style="list-style-type: none"> <li>■ <i>local_tsap_id_len</i> ≥ 02h</li> <li>■ <i>local_tsap_id</i>[1] = E0h bei <i>local_tsap_id_len</i> = 02h</li> <li>■ <i>local_tsap_id</i>[1] ein ASCII-Zeichen bei <i>local_tsap_id_len</i> ≥ 03h</li> <li>■ <i>local_tsap_id</i>[1] ist ein ASCII-Zeichen und <i>local_tsap_id_len</i> ≥ 03h</li> </ul>
1	80B5h	Parameter <i>active_est</i> (UDT 65) ist TRUE bei der Protokollvariante UDP.
1	80B6h	Parameter <i>connection_type</i> ist ungültig (UDT 65).

ERROR	STATUS	Beschreibung
1	80B7h	Fehler in einem der folgenden Parameter des UDT 65: <ul style="list-style-type: none"> <li>■ block_length</li> <li>■ local_tsap_id_len</li> <li>■ rem_subnet_id_len</li> <li>■ rem_staddr_len</li> <li>■ rem_tsap_id_len</li> <li>■ next_staddr_len</li> </ul>
1	80B8h	Parameter <i>id</i> in der lokalen Verbindungsbeschreibung (UDT 65) und Parameter ID sind unterschiedlich.
1	80C3h	Temporärer Ressourcenmangel der CPU
1	80C4h	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Verbindung kann momentan nicht aufgebaut werden.</li> <li>■ Die Schnittstelle wird neu parametrisiert .</li> </ul>
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation <a href="#">↗ Kapitel 4.1 "Allgemeine und spezifische Fehlercodes RET_VAL" auf Seite 9</a>

## 5.1.6 UDT 65 - TCON\_PAR - Datenstruktur für FB 65

### 5.1.6.1 Datenstruktur für Verbindung

In der Verbindungsparametrierung von TCP native bzw. ISO on TCP legen Sie fest, welcher Kommunikationspartner den Verbindungsaufbau aktiviert und welcher auf eine Anforderung des Kommunikationspartners hin einen passiven Verbindungsaufbau durchführt. Wenn beide Kommunikationspartner ihren Verbindungsaufbau angestoßen haben, kann das Betriebssystem die Kommunikationsverbindung herstellen. Für die Kommunikation ist ein DB erforderlich. Hierbei erhält der DB seine Datenstruktur aus dem UDT 65 TCON\_PAR. Für jede Verbindung ist solch eine Datenstruktur erforderlich, die Sie auch in einem globalen DB zusammenfassen können. Der Verbindungsparameter CONNECT des FB 65 TCON enthält einen Verweis auf die Adresse der zugehörigen Verbindungsbeschreibung (z.B. P#DB10.DBX0.0 Byte 64).

#### Datenstruktur

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	block_length	WORD	40h	Länge des UDT 65: 64 Bytes (fest)
2 ... 3	id	WORD	0000h	<ul style="list-style-type: none"> <li>■ Referenz auf diese Verbindung (Wertebereich: 0001h ... 0FFFh)</li> <li>■ Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei <i>ID</i> angeben.</li> </ul>
4	connection_type	BYTE	01h	Verbindungstyp: <ul style="list-style-type: none"> <li>■ 11h: TCP/IP native</li> <li>■ 12h: ISO on TCP</li> <li>■ 13h: UDP</li> <li>■ 01h: TCP/IP native - Kompatibilitätsmode</li> </ul>

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
5	active_est	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus: TCP, TCP, IoT: <ul style="list-style-type: none"> <li>■ FALSE: passiver Verbindungsaufbau</li> <li>■ TRUE: aktiver Verbindungsaufbau</li> </ul> UDP: <ul style="list-style-type: none"> <li>■ FALSE</li> </ul>
6	local_device_id	BYTE	02h	Kommunikationsschnittstelle <ul style="list-style-type: none"> <li>■ 00h: Ethernet-PG/OP-Kanal der CPU</li> <li>■ 02h: Ethernet-CP der CPU</li> </ul>
7	local_tsap_id_len	BYTE	02h	Verwendete Länge des Parameters <i>local_tsap_id</i> ; mögliche Werte: TCP <ul style="list-style-type: none"> <li>■ Aktive Seite: 0 (dynamische Portvergabe) oder 2</li> <li>■ Passive Seite: 2</li> </ul> ISO on TCP <ul style="list-style-type: none"> <li>■ 2 ... 16</li> </ul> UDP <ul style="list-style-type: none"> <li>■ 2</li> </ul> TCP <ul style="list-style-type: none"> <li>■ Aktive Seite: 0</li> <li>■ Passive Seite: 2</li> </ul>
8	rem_subnet_id_len	BYTE	00h	Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit 00h belegen.
9	rem_staddr_len	BYTE	00h	Länge der Adresse des remoten Verbindungsendpunkts: TCP/ISO on TCP/TCP (Komp.) <ul style="list-style-type: none"> <li>■ 0: unspezifiziert, d.h. Parameter <i>rem_staddr</i> ist irrelevant.</li> <li>■ 4: gültige IP-Adr. im Parameter <i>rem_staddr</i></li> </ul> UDP <ul style="list-style-type: none"> <li>■ 0*</li> </ul>

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
10	rem_tsap_id_len	BYTE	00h	<p>Verwendete Länge des Parameters <i>rem_tsap_id</i>; mögliche Werte:</p> <p>TCP</p> <ul style="list-style-type: none"> <li>■ Aktive Seite: 2 (Der Port muss spezifiziert werden.)</li> <li>■ Passive Seite: 0 oder 2</li> </ul> <p>ISO on TCP</p> <ul style="list-style-type: none"> <li>■ 0 oder 2 ... 16</li> </ul> <p>UDP</p> <ul style="list-style-type: none"> <li>■ Dieser Parameter wird nicht verwendet. Parameter mit 00h belegen.</li> </ul> <p>TCP (Komp.)</p> <ul style="list-style-type: none"> <li>■ Aktive Seite: 2 (Der Port muss spezifiziert werden.)</li> </ul> <p>Für die passive Seite ist nur der Wert 0 zulässig.</p>
11	next_staddr_len	BYTE	00h	<p>Verwendete Länge des Parameters <i>next_staddr</i></p> <ul style="list-style-type: none"> <li>■ 00h: Ethernet-CP der CPU</li> <li>■ 01h: Ethernet-PG/OP-Kanal der CPU</li> </ul>
12 ... 27	local_tsap_id	ARRAY [1..16] of BYTE	00h ...	<p>Bei <i>connection_type</i>:</p> <p>TCP, UDP</p> <ul style="list-style-type: none"> <li>■ local_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung</li> <li>■ local_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung</li> <li>■ local_tsap_id[3-16] = 00h</li> </ul> <p>ISO on TCP</p> <ul style="list-style-type: none"> <li>■ lokale TSAP-ID (mögliche Werte: 2000 ... 5000) <ul style="list-style-type: none"> <li>– local_tsap_id[1] = E0h (Verbindungstyp T-Verbindung)</li> <li>– local_tsap_id[2] = Rack und Steckplatz der eigenen CPU (Bits 0 ... 4 Steckplatz, Bits 5 ... 7: Racknummer)</li> <li>– local_tsap_id[3-16] = TSAP-Erweiterung</li> </ul> </li> </ul> <p>TCP (Komp.)</p> <ul style="list-style-type: none"> <li>■ local_tsap_id[1] = low byte der Port-Nr. in hexadezimaler Darstellung</li> <li>■ local_tsap_id[2] = high byte der Port-Nr. in hexadezimaler Darstellung</li> <li>■ local_tsap_id[3-16] = 00h</li> </ul> <p><b>Hinweis:</b> Stellen Sie sicher, dass jeder Wert von <i>local_tsap_id</i>, den Sie auf Ihrer CPU verwenden, eindeutig ist.</p>
28 ... 33	rem_subnet_id	ARRAY [1..6] of BYTE	00h ...	<p>Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit 00h belegen.</p>



Byte	Parameter	Datentyp	Anfangswert	Beschreibung
34 ... 39	rem_staddr	ARRAY [1..6] of BYTE	00h ...	<p>IP-Adresse des remoten Verbindungsendpunkts, z.B. 192.168.002.003: bei <i>connection_type</i></p> <ul style="list-style-type: none"> <li>■ TCP / ISO on TCP <ul style="list-style-type: none"> <li>– rem_staddr[1] = C0h (192)</li> <li>– rem_staddr[2] = A8h (168)</li> <li>– rem_staddr[3] = 02h (002)</li> <li>– rem_staddr[4] = 03h (003)</li> <li>– rem_staddr[5-6] = irrelevant</li> </ul> </li> <li>■ UDP <ul style="list-style-type: none"> <li>– Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 00h belegen.</li> </ul> </li> <li>■ TCP (Komp.) <ul style="list-style-type: none"> <li>– rem_staddr[1] = 03h (003)</li> <li>– rem_staddr[2] = 02h (002)</li> <li>– rem_staddr[3] = A8h (168)</li> <li>– rem_staddr[4] = C0h (192)</li> <li>– rem_staddr[5-6] = irrelevant</li> </ul> </li> </ul>
40 ... 55	rem_tsap_id	ARRAY [1..16] of BYTE	00h ...	<p>Bei <i>connection_type</i></p> <ul style="list-style-type: none"> <li>■ TCP: remote Port-Nr. (mögliche Werte: 2000 ... 5000), <ul style="list-style-type: none"> <li>– rem_tsap_id[1] = High-Byte der Port-Nr. in hexadezimaler Darstellung,</li> <li>– rem_tsap_id[2] = Low-Byte der Port-Nr. in hexadezimaler Darstellung,</li> <li>– rem_tsap_id[3-16] = 00h</li> </ul> </li> <li>■ ISO on TCP: remote TSAP-ID: <ul style="list-style-type: none"> <li>– rem_tsap_id[1] = E0h (Verbindungstyp T-Verbindung),</li> <li>– rem_tsap_id[2] = Rack und Steckplatz des remoten Verbindungsendpunkts (CPU) (Bits 0 ... 4: Steckplatz, Bits 5 ... 7: Racknummer),</li> <li>– rem_tsap_id[3-16] = TSAP-Erweiterung</li> </ul> </li> <li>■ UDP <p>Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 00h belegen.</p> </li> <li>■ TCP (Komp.): remote Port-Nr. (mögliche Werte: 2000 ... 5000), <ul style="list-style-type: none"> <li>– local_tsap_id[1] = Low-Byte der Port-Nr. in hexadezimaler Darstellung,</li> <li>– local_tsap_id[2] = High-Byte der Port-Nr. in hexadezimaler Darstellung,</li> <li>– local_tsap_id[3-16] = 00h</li> </ul> </li> </ul>
56 ... 61	next_staddr	ARRAY [1..6] of BYTE	00h ...	<p>Rack und Steckplatz des projektierten CP für die PG/OP-Schnittstelle</p> <ul style="list-style-type: none"> <li>■ 00h (Ethernet-PG/OP-Kanal) <ul style="list-style-type: none"> <li>– next_staddr[1]: 04h</li> <li>– next_staddr[2-6]: 00h</li> </ul> </li> <li>■ 02h (Ethernet-CP) <ul style="list-style-type: none"> <li>– next_staddr[1-6]: 00h</li> </ul> </li> </ul>

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
62 ... 63	spare	WORD	0000h	irrelevant

\*) Die Partner IP-Adresse wird beim Aufruf von TUSEND/TURECV über den Parameter *ADDR* angegeben.

### 5.1.6.2 Datenstruktur für Kommunikationszugangspunkt

Ein Kommunikationszugangspunkt stellt die Verbindung zwischen Anwenderprogramm der Kommunikationsschicht des Betriebssystems dar. Für die Kommunikation über UDP muss jeder Kommunikationspartner einen Kommunikationszugangspunkt mittels eines DB definieren. Hierbei erhält der DB seine Datenstruktur aus dem UDT 65 "TCON\_PAR".

#### Datenstruktur

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 1	block_length	WORD	40h	Länge des UDT 65: 64 Bytes (fest)
2 ... 3	id	WORD	0000h	<ul style="list-style-type: none"> <li>Referenz auf diese Verbindung zwischen-Anwenderprogramm und Kommunikationsschicht des Betriebssystems (Wertebereich: 0001h ... 0FFFh)</li> <li>Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei ID angeben.</li> </ul>
4	connection_type	BYTE	01h	Verbindungstyp: <ul style="list-style-type: none"> <li>13h: UDP</li> </ul>
5	active_est	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus: Diesen Parameter müssen Sie mit FALSE belegen, da über den Kommunikationszugangspunkt Daten sowohl gesendet als auch empfangen werden können.
6	local_device_id	BYTE	02h	Kommunikationsschnittstelle <ul style="list-style-type: none"> <li>00h: Ethernet-PG/OP-Kanal der CPU</li> <li>02h: Ethernet-CP der CPU</li> </ul>
7	local_tsap_id_len	BYTE	02h	Verwendete Länge des Parameters <i>local_tsap_id</i> ; möglicher Wert: 2
8	rem_subnet_id_len	BYTE	00h	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
9	rem_staddr_len	BYTE	00h	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
10	rem_tsap_id_len	BYTE	00h	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
11	next_staddr_len	BYTE	00h	Dieser Parameter wird nicht verwendet. Wert 00h (fix)

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
12 ... 27	local_tsap_id	ARRAY [1..16] of BYTE	00h ...	<ul style="list-style-type: none"> <li>lokale Port-Nr. (mögliche Werte: 2000 ... 5000) <ul style="list-style-type: none"> <li>local_tsap_id[1] = High-Byte der Port-Nr. in hexadezimaler Darstellung</li> <li>local_tsap_id[2] = Low-Byte der Port-Nr. in hexadezimaler Darstellung</li> <li>local_tsap_id[3-16] = irrelevant</li> </ul> </li> </ul> <p><b>Hinweis:</b> Stellen Sie sicher, dass jeder Wert von <i>local_tsap_id</i>, den Sie auf Ihrer CPU verwenden, eindeutig ist.</p>
28 ... 33	rem_subnet_id	ARRAY [1..6] of BYTE	00h ...	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
34 ... 39	rem_staddr	ARRAY [1..6] of BYTE	00h ...	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
40 ... 55	rem_tsap_id	ARRAY [1..16] of BYTE	00h ...	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
56 ... 61	next_staddr	ARRAY [1..6] of BYTE	00h ...	Dieser Parameter wird nicht verwendet. Wert 00h (fix)
62 ... 63	spare	WORD	0000h	irrelevant

### 5.1.7 FB 66 - TDISCON - Verbindung abbauen

#### Verwendung bei TCP native und ISO on TCP

Der FB 66 TDISCON baut eine Kommunikationsverbindung der CPU zu einem Kommunikationspartner ab.

#### Verwendung bei UDP

Der FB 66 TDISCON löst den lokalen Kommunikationszugangspunkt auf, d.h. die Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird abgebaut.

#### Beschreibung

Der FB 66 TDISCON ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe. Sie starten den Verbindungsabbau, indem Sie den FB 66 mit *REQ* = 1 aufrufen.

Nach dem erfolgreichen Durchlauf des FB 66 TDISCON ist die beim FB 65 TCON angegebene ID nicht mehr gültig und kann damit weder zum Senden noch zum Empfangen verwendet werden.

Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs).

In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *DONE* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 66 aktuell befindet bzw. wann der Verbindungsaufbau beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter <i>REQ</i> , aktiviert den Abbau der durch <i>ID</i> gegebenen Verbindung. Der Anstoß erfolgt bei steigender Flanke.
ID	INPUT	WORD	M, D, Konstante	Referenz auf die abzubauende Verbindung zum remoten Partner bzw. zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. <i>ID</i> muss identisch sein mit dem zugehörigen Parameter <i>ID</i> in der lokalen Verbindungsbeschreibung. Wertebereich: 0001h ... 0FFFh
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li><i>BUSY</i> = 1: Der Auftrag ist noch nicht beendet.</li> <li><i>BUSY</i> = 0: Der Auftrag ist beendet.</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li><i>ERROR</i> = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Statusinformationen

### Statusinformationen

ERROR	STATUS	Erläuterung
0	0000h	Verbindung konnte abgebaut werden
0	7000h	Erstaufruf mit <i>REQ</i> = 0, kein Anstoß zum Verbindungsabbau
0	7001h	Erstaufruf mit <i>REQ</i> = 1, Start der Auftragsbearbeitung, Verbindung wird abgebaut
0	7002h	Zwischenaufruf ( <i>REQ</i> irrelevant), Verbindung wird abgebaut
1	8086h	Parameter <i>ID</i> liegt in einem unzulässigen Wertebereich
1	80A3h	Es wird versucht, eine nicht bestehende Verbindung abzubauen
1	80C4h	Temporärer Kommunikationsfehler: Schnittstelle wird neu parametrisiert

ERROR	STATUS	Erläuterung
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation ↪ Kapitel 4.1 "Allgemeine und spezifische Fehlercodes RET_VAL" auf Seite 9

### 5.1.8 FB 67 - TUSEND - Daten senden - UDP

#### Beschreibung

Der FB 67 TUSEND sendet Daten über UDP an den über den Parameter *ADDR* adressierten remoten Partner.



Bei aufeinander folgenden Sendevorgängen an verschiedene Partner müssen Sie bei den Aufrufen des FB 67 TUSEND lediglich den Parameter *ADDR* anpassen. Der erneute Aufruf der FB 65 TCON und FB 66 TDISCON hingegen entfällt.

#### Arbeitsweise

- Der FB 67 TUSEND ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe. Sie starten den Sendevorgang, indem Sie den FB 67 mit *REQ* = 1 aufrufen.
- Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs).
- In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *DONE* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 67 aktuell befindet bzw. wann der Sendevorgang beendet ist.

BUSY	DONE	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.



Durch die asynchrone Arbeitsweise des FB 67 TUSEND müssen Sie die Daten im Sendebereich so lange konsistent halten, bis der Parameter *DONE* oder der Parameter *ERROR* den Wert *TRUE* annimmt.

# Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter <i>REQ</i> , aktiviert den Sendevorgang bei steigender Flanke.  Beim erstmaligen Aufruf mit <i>REQ</i> = 1 werden Bytes aus dem mit Parameter <i>DATA</i> angegebenen Bereich übergeben.
ID	INPUT	WORD	M, D, Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems.  <i>ID</i> muss identisch sein mit dem zugehörigen Parameter <i>ID</i> in der lokalen Verbindungsbeschreibung.  Wertebereich: 0001h ... 0FFFh
LEN	INPUT	INT	E, A, M, D, L	Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen:  Wertebereich: 1 ... 1460
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li><i>BUSY</i> = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li><i>BUSY</i> = 0: Der Auftrag ist beendet.</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li><i>ERROR</i> = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Fehlerinformation
DATA	IN_OUT	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge  Die Adresse verweist auf: <ul style="list-style-type: none"> <li>das Prozessabbild der Eingänge</li> <li>das Prozessabbild der Ausgänge</li> <li>einen Merker</li> <li>einen Datenbaustein</li> </ul> Zulässige referenzierte Datentypen: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
ADDR	IN_OUT	ANY	D	Zeiger auf die Adresse des Empfängers (z.B. P#DB100.DBX0.0 Byte 8), siehe Struktur der Adressinformation des remoten Partners bei UDP

## Fehlerinformation

ERROR	STATUS	Erläuterung
0	0000h	Sendeauftrag wurde ohne Fehler abgeschlossen.
0	7000h	Erstaufruf mit <i>REQ</i> = 0, kein Anstoß zum Senden.
0	7001h	Erstaufruf mit <i>REQ</i> = 1, Anstoß des Sendevorgangs.
0	7002h	Zwischenaufruf ( <i>REQ</i> irrelevant), Auftrag ist in Bearbeitung <b>Hinweis:</b> In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendepuffer <i>DATA</i> zu.
1	8085h	Parameter <i>LEN</i> hat den Wert 0 oder ist größer als der größte zulässige Wert.
1	8086h	Parameter <i>ID</i> liegt in einem unzulässigen Wertebereich.
0	8088h	Parameter <i>LEN</i> ist größer als der in <i>DATA</i> angegebene Speicherbereich.
1	8089h	Parameter <i>ADDR</i> zeigt nicht auf einen Datenbaustein.
1	80A1h	Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Zu der angegebenen <i>ID</i> wurde noch kein FB 65 TCON aufgerufen</li> <li>■ Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich.</li> <li>■ Schnittstelle wird neu initialisiert.</li> </ul>
1	80A4h	Die IP-Adresse des Kommunikationspartners ist ungültig.
1	80B3h	<ul style="list-style-type: none"> <li>■ Der parametrierte Verbindungstyp (Parameter <i>connection_type</i> in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie den FB 63 TSEND.</li> <li>■ Parameter <i>ADDR</i>: ungültige Angaben für Port-Nr. oder IP-Adresse.</li> </ul>
1	80B7h	Längenfehler: Im Parameter <i>ADDR</i> ist die Längenangabe < 8Byte.
1	80C4h	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Der Kommunikationspartner ist momentan nicht erreichbar.</li> <li>■ Die Verbindung wird gerade konfiguriert (bzw. TCON läuft noch).</li> </ul>
1	8822h	Parameter <i>DATA</i> : Quellbereich ungültig, Bereich im DB nicht vorhanden
1	8824h	Parameter <i>DATA</i> : Bereichsfehler im ANY-Pointer.
1	8832h	Parameter <i>DATA</i> : DB-Nummer ist zu groß.
1	883Ah	Parameter <i>DATA</i> : Zugriff auf Sendepuffer nicht möglich (z.B. gelöschter DB).
1	887Fh	Parameter <i>DATA</i> : Interner Fehler, z.B. unzulässige ANY-Referenz.
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation ↪ Kapitel 4.1 "Allgemeine und spezifische Fehlercodes <i>RET_VAL</i> " auf Seite 9

### 5.1.9 FB 68 - TURCV - Daten empfangen - UDP

#### Beschreibung

- Der FB 68 TURCV empfängt Daten über UDP. Nach erfolgreichem Abschluss des FB 68 TURCV wird Ihnen am Parameter *ADDR* die Adresse des remoten Partners, also des Senders, zur Verfügung gestellt.
- Der FB 68 TURCV ist ein asynchron arbeitender FB, d.h. die Bearbeitung erstreckt sich über mehrere FB-Aufrufe. Sie starten den Sendevorgang, indem Sie den FB 68 mit *REQ* = 1 aufrufen.
- Über den Ausgangsparameter *BUSY* und den Ausgangsparameter *STATUS* wird der Zustand des Auftrags angezeigt. Dabei entspricht *STATUS* dem Ausgangsparameter *RET\_VAL* der asynchron arbeitenden SFCs (siehe auch Bedeutung von *REQ*, *RET\_VAL* und *BUSY* bei asynchron arbeitenden SFCs).
- In der folgenden Tabelle ist der Zusammenhang zwischen *BUSY*, *NDR* und *ERROR* angegeben. Mit ihrer Hilfe können Sie feststellen, in welchem Zustand sich der FB 68 aktuell befindet bzw. wann der Empfangsvorgang beendet ist.

BUSY	NDR	ERROR	Beschreibung
TRUE	irrelevant	irrelevant	Der Auftrag ist in Bearbeitung.
FALSE	TRUE	FALSE	Der Auftrag wurde erfolgreich abgeschlossen.
FALSE	FALSE	TRUE	Der Auftrag wurde mit einem Fehler beendet. Die Fehlerursache können Sie dem Parameter <i>STATUS</i> entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.



Durch die asynchrone Arbeitsweise des FB 68 TURCV sind die Daten im Empfangsbereich erst dann konsistent, wenn der Parameter *NDR* den Wert *TRUE* annimmt.

#### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	INPUT	BOOL	E, A, M, D, L	Steuerparameter enabled to receive: Mit <i>EN_R</i> = 1 wird der FB 68 TURCV empfangsbereit.
ID	INPUT	WORD	M, D, Konstante	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems.  <i>ID</i> muss identisch sein mit dem zugehörigen Parameter <i>ID</i> in der lokalen Verbindungsbeschreibung.  Wertebereich: 0001h ... 0FFFh
LEN	INPUT	INT	E, A, M, D, L	$1 \leq LEN \leq 1472$ : Anzahl der Bytes, die empfangen werden sollen.  Die empfangenen Daten werden beim Bausteinufruf unmittelbar bereitgestellt. Die Anzahl der empfangenen Daten steht in <i>RCVD_LEN</i> zur Verfügung.



Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>NDR</i> : <ul style="list-style-type: none"> <li>■ <i>NDR</i> = 0: Auftrag wurde noch nicht gestartet oder läuft noch</li> <li>■ <i>NDR</i> = 1: Auftrag wurde erfolgreich abgeschlossen</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 1: Bei der Bearbeitung ist ein Fehler aufgetreten. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul>
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <i>BUSY</i> = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden.</li> <li>■ <i>BUSY</i> = 0: Der Auftrag ist beendet.</li> </ul>
STATUS	OUTPUT	WORD	M, D	Zustandsparameter <i>STATUS</i> : Fehlerinformation
RCVD_LEN	OUTPUT	INT	E, A, M, D, L	Anzahl der tatsächlich empfangenen Daten in Bytes
DATA	IN_OUT	ANY	E, A, M, D	Empfangsbereich, enthält Adresse und Länge. Die Adresse verweist auf: <ul style="list-style-type: none"> <li>■ das Prozessabbild der Eingänge</li> <li>■ das Prozessabbild der Ausgänge</li> <li>■ einen Merker</li> <li>■ einen Datenbaustein</li> </ul> Zulässige referenzierte Datentypen: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
ADDR	IN_OUT	ANY	D	Zeiger auf die Adresse des Senders (z.B. P#DB100.DBX0.0 Byte 8), siehe Struktur der Adressinformation des remoten Partners bei UDP

## Fehlerinformation

ERROR	STATUS	Erläuterung
0	0000h	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in <i>RCVD_LEN</i> angezeigt.
0	7000h	Erstaufruf mit <i>REQ</i> = 0, kein Anstoß zum Empfangen
0	7001h	Baustein ist empfangsbereit
0	7002h	Zwischenaufruf, Auftrag ist in Bearbeitung <b>Hinweis:</b> In dieser Bearbeitungsphase schreibt das Betriebssystem Daten in den Empfangspuffer <i>DATA</i> . Deshalb können im Fehlerfall inkonsistente Daten im Empfangspuffer stehen.
1	8085h	Parameter <i>LEN</i> ist größer als der größte zulässige Wert, oder Sie haben den Wert von <i>LEN</i> gegenüber dem Erstaufruf geändert
1	8086h	Parameter <i>ID</i> liegt in einem unzulässigen Wertebereich

ERROR	STATUS	Erläuterung
1	8088h	<ul style="list-style-type: none"> <li>■ Zielpuffer (<i>DATA</i>) ist zu klein</li> <li>■ Wert in <i>LEN</i> ist größer als der durch <i>DATA</i> vorgegebene Empfangsbereich.</li> </ul>
1	8089h	Parameter <i>ADDR</i> zeigt nicht auf einen Datenbaustein.
1	80A1h	Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Zu der angegebenen <i>ID</i> wurde noch kein FB 65 "TCON" aufgerufen</li> <li>■ Die angegebene Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Empfangsvorgang über diese Verbindung ist nicht möglich.</li> <li>■ Die Schnittstelle wird neu parametrieret.</li> </ul>
1	80B3h	Der parametrierete Verbindungstyp (Parameter <i>connection_type</i> in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie den FB 64 TRCV.
1	80B7h	Längenfehler: Im Parameter <i>ADDR</i> ist die Längenangabe < 8Byte.
1	80C4h	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> <li>■ Die Verbindung wird gerade konfiguriert (bzw. TCON läuft noch).</li> </ul>
1	8922h	Parameter <i>DATA</i> : Zielbereich ungültig, Bereich im DB nicht vorhanden
1	8924h	Parameter <i>DATA</i> : Bereichsfehler im ANY-Pointer
1	8932h	Parameter <i>DATA</i> : DB-Nummer ist zu groß
1	893Ah	Parameter <i>DATA</i> : Zugriff auf Empfangspuffer nicht möglich (z.B. gelöschter DB)
1	897Fh	Parameter <i>DATA</i> : Interner Fehler, z.B. unzulässige ANY-Referenz
1	8F7Fh	Interner Fehler (VIPA spezifisch)
1	8xyyh	Allgemeine Fehlerinformation → Kapitel 4.1 "Allgemeine und spezifische Fehlercodes RET_VAL" auf Seite 9

## 5.1.10 UDT 66 - TADDR\_PAR - Datenstruktur

### 5.1.10.1 Datenstruktur für Verbindung

#### Beschreibung

- Beim FB 67 TUSEND übergeben Sie am Parameter *ADDR* die Adresse des Empfängers. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.
- Beim FB 68 TURCV erhalten Sie am Parameter *ADDR* die Adresse des Absenders der empfangenen Daten. Diese Adressinformation muss die im Folgenden angegebene Struktur haben.

#### Datenbaustein

Sie müssen einen DB anlegen, der eine oder mehrere Datenstrukturen gemäß UDT 66 TADDR\_PAR enthält.

Im Parameter *ADDR* des FB 67 TUSEND übergeben Sie und am Parameter *ADDR* des FB 68 TURCV erhalten Sie einen Zeiger auf die Adresse des zugehörigen remoten Partners (z.B. P#DB10.DBX0.0 Byte 8).

**Aufbau der Adressinformation des remoten Partners**

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 ... 3	rem_ip_addr	ARRAY [1..4] of BYTE	00h ...	IP-Adresse des remoten Partners, z.B. 192.168.002.003: <ul style="list-style-type: none"> <li>■ rem_ip_addr[1] = C0h (192)</li> <li>■ rem_ip_addr[2] = A8h (168)</li> <li>■ rem_ip_addr[3] = 02h (002)</li> <li>■ rem_ip_addr[4] = 03h (003)</li> </ul>
4 ... 5	rem_port_nr	ARRAY [1..2] of BYTE	00h ...	remote Port-Nr. (mögliche Werte: 2000 ... 5000) <ul style="list-style-type: none"> <li>■ rem_port_nr[1] = High-Byte der Port-Nr. in hexadezimaler Darstellung</li> <li>■ rem_port_nr[2] = Low-Byte der Port-Nr. in hexadezimaler Darstellung</li> </ul>
6 ... 7	spare	ARRAY [1..2] of BYTE	00h ...	reserviert (00h)

**5.2 Ethernet-Kommunikation - "Ethernet Communication"****5.2.1 Kommunikation - FC 5...6 für CP 343**

Die beiden Bausteine dienen der Verarbeitung von Verbindungsaufträgen auf SPS-Seite eines Ethernet-CP 343. Durch Einbindung dieser Bausteine in den Zyklus-Baustein OB1 können Sie zyklisch Daten senden und empfangen. Innerhalb dieser Bausteine werden die FCs 205 und 206 aufgerufen, die als Sonderfunktionsbausteine in der CPU abliegen.



*Bitte beachten Sie, dass Sie in Ihrem Anwenderprogramm für die Kommunikation mit VIPA-CPs ausschließlich die SEND/RECV-FCs von VIPA einsetzen dürfen. Bei Wechsel zu VIPA-CPs in einem schon bestehenden Projekt können die bestehenden AG\_SEND / AG\_LSEND bzw. AG\_RECV / AG\_LRECV durch AG\_SEND bzw. AG\_RECV von VIPA ohne Anpassung ersetzt werden. Da sich der CP automatisch an die Länge der zu übertragenden Daten anpasst ist die L-Variante von SEND bzw. RECV bei VIPA nicht erforderlich.*

**Kommunikationsbausteine**

Für die Kommunikation zwischen CPU und Ethernet-CP 343 stehen Ihnen folgende FCs zur Verfügung:

- AG\_SEND (FC 5)
  - Dieser Baustein übergibt die Nutzdaten aus dem über SEND angegebenen Datenbereich an den über ID und LADDR spezifizierten CP. Als Datenbereich können Sie einen PA-, Merker- oder Datenbaustein-Bereich angeben. Wurde der Datenbereich fehlerfrei übertragen, so wird "Auftrag fertig ohne Fehler" zurückgemeldet.
- AG\_RECV (FC 6)
  - Der Baustein übernimmt vom CP die Nutzdaten und legt sie in dem über RECV definierten Datenbereich ab. Als Datenbereich können Sie einen PE-, Merker- oder Datenbaustein-Bereich angeben. Wurde der Datenbereich fehlerfrei übernommen, so wird "Auftrag fertig ohne Fehler" zurückgemeldet.

### Statusanzeigen

Der CP bearbeitet Sende- und Empfangsaufträge unabhängig vom CPU Zyklus und benötigt hierzu eine Übertragungszeit. Die Schnittstelle mit den FC-Bausteinen zum Anwenderprogramm wird hierbei über Quittungen synchronisiert. Für die Statusauswertung liefern die Kommunikationsbausteine Parameter zurück, die Sie in Ihrem Anwenderprogramm direkt auswerten können. Diese Statusanzeigen werden bei jedem Baustein-Aufruf aktualisiert.

### Einsatz unter hoher Kommunikationslast

Verwenden Sie keine zyklischen Aufrufe der Kommunikationsbausteine im OB 1. Dies führt zu einer ständigen Kommunikation zwischen CPU und CP. Programmieren Sie statt dessen Ihre Kommunikationsbausteine in einem Zeit-OB, deren Zykluszeit größer ist als die des OB1 bzw. ereignisgesteuert.

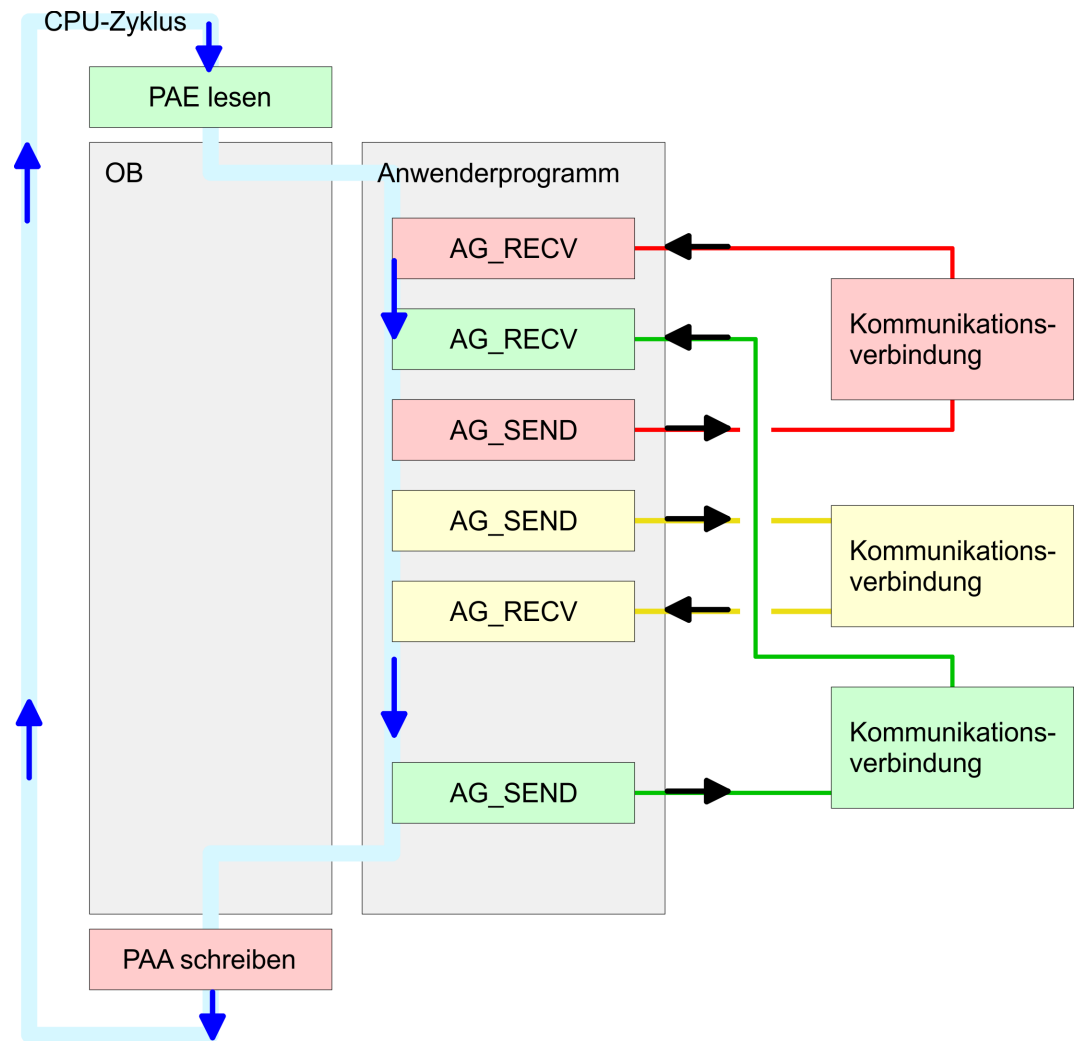
### Aufruf FC schneller als CP-Übertragungszeit

Wird ein Baustein im Anwenderprogramm erneut aufgerufen, bevor die Daten vollständig gesendet oder empfangen wurden, wird an der Schnittstelle der FC-Bausteine wie folgt verfahren:

- AG\_SEND
  - Es wird kein Auftrag entgegen genommen, bis die Datenübertragung über die Verbindung vom Partner quittiert wurde. Solange erhalten Sie die Meldung "Auftrag läuft", bis der CP den nächsten Auftrag für die gleiche Verbindung übernehmen kann.
- AG\_RECV
  - Der Auftrag wird mit der Meldung "Es liegen noch keine Daten vor" quittiert, solange der CP die Empfangsdaten noch nicht vollständig empfangen hat.

### AG\_SEND, AG\_RECV im Anwenderprogramm

Eine mögliche Ablaufsequenz für die FC-Bausteine zusammen mit den Organisations- und Programmbausteinen im CPU-Zyklus ist nachfolgend dargestellt:



Die FC-Bausteine mit zugehöriger Kommunikationsverbindung sind farblich zusammengefasst. Hier können Sie auch erkennen, dass Ihr Anwenderprogramm aus beliebig vielen Bausteinen bestehen kann. Somit können Sie ereignis- bzw. programmgesteuert an beliebiger Stelle im CPU-Zyklus mit AG\_SEND Daten senden bzw. mit AG\_RECV Daten empfangen. Sie können die Bausteine für **eine** Kommunikationsverbindung auch mehrmals in einem Zyklus aufrufen.

## 5.2.2 FC 5 - AG\_SEND - Senden an CP 343

Mit AG\_SEND werden die zu sendenden Daten von der CPU an einen Ethernet-CP übertragen.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 205 AG\_SEND aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

### Parameter

Parameter	Deklaration	Datentyp	Beschreibung
ACT	INPUT	BOOL	Aktivierung des Senders 0: Aktualisiert die <i>DONE</i> , <i>ERROR</i> und <i>STATUS</i> 1: Der unter <i>SEND</i> mit der Länge <i>LEN</i> abgelegte Datenbereich wird gesendet
ID	INPUT	INT	Verbindungsnummer 1 ... 16 (identisch mit <i>ID</i> aus NetPro)
LADDR	INPUT	WORD	Logische Basisadresse des CPs (identisch mit <i>LADDR</i> aus NetPro)
SEND	INPUT	ANY	Datenbereich
LEN	INPUT	INT	Anzahl der Bytes, die aus dem Datenbereich zu übertragen sind
DONE	OUTPUT	BOOL	Zustandsparameter für den Auftrag 0: Auftrag läuft 1: Auftrag fertig ohne Fehler
ERROR	OUTPUT	BOOL	Fehleranzeige 0: Auftrag läuft (bei <i>DONE</i> = 0) 0: Auftrag fertig ohne Fehler (bei <i>DONE</i> = 1) 1: Auftrag fertig mit Fehler
STATUS	OUTPUT	WORD	Statusanzeige, die in Verbindung mit <i>DONE</i> und <i>ERROR</i> zurückgeliefert wird. Näheres hierzu finden Sie in der nachfolgenden Tabelle.

### DONE, ERROR, STATUS

In der nachfolgenden Tabelle sind alle Meldungen aufgeführt, die der Ethernet-CP nach einem SEND-Auftrag bzw. RECV-Auftrag zurückliefern kann. Ein "-" bedeutet, dass diese Meldung für den entsprechenden SEND- bzw. RECV-Auftrag nicht existiert.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
1	-	0	0000h	Auftrag fertig ohne Fehler.
-	1	0	0000h	Neue Daten wurden ohne Fehler übernommen.
0	-	0	0000h	Kein Auftrag in Bearbeitung.
-	0	0	8180h	Es liegen noch keine Daten vor.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
0	0	0	8181h	Auftrag läuft
0	0	1	8183h	Für diesen Auftrag gibt es keine CP-Projektierung.
0	-	1	8184h	Es ist ein Systemfehler aufgetreten.
-	0	1	8184h	Es ist ein Systemfehler aufgetreten (Quelldatenbereich fehlerhaft)
0	-	1	8185h	Parameter <i>LEN</i> größer als Quell-Bereich <i>SEND</i> .
	0	1	8185h	Ziel-Puffer (RECV) ist zu klein.
0	0	1	8186h	Parameter <i>ID</i> ungültig (nicht im Bereich 1 ... 16).
0	-	1	8302h	Keine Empfangsressourcen bei Ziel-Station, Empfänger-Station kann empfangene Daten nicht schnell genug verarbeiten bzw. hat keine Empfangsressourcen bereitgestellt.
0	-	1	8304h	Die Verbindung ist nicht aufgebaut. Der Sendeauftrag sollte erst nach einer Wartezeit > 100ms erneut abgesetzt werden.
-	0	1	8304h	Die Verbindung ist nicht aufgebaut. Der Empfangsauftrag sollte erst nach einer Wartezeit > 100ms erneut abgesetzt werden.
0	-	1	8311h	Zielstation ist unter der angegebenen Ethernet-Adresse nicht erreichbar.
0	-	1	8312h	Ethernet-Fehler im CP
0		1	8F22h	Quell-Bereich ungültig, wenn beispielsweise Bereich im DB nicht vorhanden Parameter <i>LEN</i> < 0.
-	0	1	8F23h	Quell-Bereich ungültig, wenn beispielsweise Bereich im DB nicht vorhanden Parameter <i>LEN</i> < 0.
0	-	1	8F24h	Bereichsfehler beim Lesen eines Parameters.
-	0	1	8F25h	Bereichsfehler beim Schreiben eines Parameters.
0	-	1	8F28h	Ausrichtungsfehler beim Lesen eines Parameters.
-	0	1	8F29h	Ausrichtungsfehler beim Schreiben eines Parameters.
-	0	1	8F30h	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein
-	0	1	8F31h	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein
0	0	1	8F32h	Parameter enthält zu große DB-Nummer.
0	0	1	8F33h	DB-Nummer Fehler
0	0	1	8F3Ah	Bereich nicht geladen (DB)
0	-	1	8F42h	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
-	0	1	8F43h	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	-	1	8F44h	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt.
-	0	1	8F45h	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
0	0	1	8F7Fh	Interner Fehler z.B. unzulässige ANY-Referenz z.B. Parameter <i>LEN</i> = 0.
0	0	1	8090h	Baugruppe mit dieser Baugruppen-Anfangsadresse nicht vorhanden oder CPU in STOP.
0	0	1	8091h	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	0	1	8092h	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.
-	0	1	80A0h	Negative Quittung beim Lesen von Baugruppe.
0	0	1	80A4h	reserviert
0	0	1	80B0h	Baugruppe kennt den Datensatz nicht.
0	0	1	80B1h	Die Längenangabe (im Parameter <i>LEN</i> ) ist falsch.
0	0	1	80B2h	reserviert
0	0	1	80C0h	Datensatz kann nicht gelesen werden.
0	0	1	80C1h	Der angegebene Datensatz ist gerade in Bearbeitung.
0	0	1	80C2h	Es liegt ein Auftragsstau vor.
0	0	1	80C3h	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	0	1	80C4h	Kommunikationsfehler (tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll).
0	0	1	80D2h	Baugruppen-Anfangsadresse ist falsch.

#### Status-Parameter bei Neuanlauf

Bei einem Neuanlauf des CP werden die Ausgabe-Parameter wie folgt zurückgesetzt:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS = 8180h (bei AG\_RECV)
- STATUS = 8181h (bei AG\_SEND)



### 5.2.3 FC 6 - AG\_RECV - Empfangen von CP 343

Mit dem 1. Aufruf von AG\_RECV richten Sie einen Empfangspuffer zwischen der CPU und einem Ethernet CP 343 ein. Von jetzt ab werden empfangene Daten automatisch in diesem Puffer abgelegt. Sobald nach einem Aufruf von AG\_RECV der Rückgabewert *NDR* = 1 zurückgeliefert wird, liegen gültige Daten ab. Da mit einem weiteren Aufruf von AG\_RECV der Empfangspuffer für den Empfang neuer Daten wieder freigegeben wird, müssen Sie die zuvor empfangenen Daten sichern.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 206 AG\_RECV aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

#### Parameter

Parameter	Deklaration	Datentyp	Beschreibung
ID	INPUT	INT	Verbindungsnummer 1 ... 16 (identisch mit <i>ID</i> aus NetPro)
LADDR	INPUT	WORD	Logische Basisadresse des CPs (identisch mit <i>LADDR</i> aus NetPro)
RECV	INPUT	ANY	Datenbereich für die empfangenen Daten.
NDR	OUTPUT	BOOL	Zustandsparameter für den Auftrag 0: Auftrag läuft 1: Auftrag fertig Daten wurden ohne Fehler übernommen
ERROR	OUTPUT	BOOL	Fehleranzeige 0: Auftrag läuft (bei <i>NDR</i> = 0) 0: Auftrag fertig ohne Fehler ( <i>NDR</i> = 1) 1: Auftrag fertig mit Fehler
STATUS	OUTPUT	WORD	Statusanzeige, die in Verbindung mit <i>NDR</i> und <i>ERROR</i> zurückgeliefert wird. Näheres hierzu finden Sie in der nachfolgenden Tabelle.
LEN	OUTPUT	INT	Anzahl der Bytes, die empfangen wurden.

#### DONE, ERROR, STATUS

In der nachfolgenden Tabelle sind alle Meldungen aufgeführt, die der Ethernet-CP 343 nach einem SEND-Auftrag bzw. RECV-Auftrag zurückliefern kann.

Ein "-" bedeutet, dass diese Meldung für den entsprechenden SEND- bzw. RECV-Auftrag nicht existiert.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
1	-	0	0000h	Auftrag fertig ohne Fehler.
-	1	0	0000h	Neue Daten wurden ohne Fehler übernommen.
0	-	0	0000h	Kein Auftrag in Bearbeitung.
-	0	0	8180h	Es liegen noch keine Daten vor.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
0	0	0	8181h	Auftrag läuft
0	0	1	8183h	Für diesen Auftrag gibt es keine CP-Projektierung.
0	-	1	8184h	Es ist ein Systemfehler aufgetreten.
-	0	1	8184h	Es ist ein Systemfehler aufgetreten (Quelldatenbereich fehlerhaft)
0	-	1	8185h	Parameter <i>LEN</i> größer als Quell-Bereich <i>SEND</i> .
	0	1	8185h	Ziel-Puffer (RECV) ist zu klein.
0	0	1	8186h	Parameter <i>ID</i> ungültig (nicht im Bereich 1 ... 16).
0	-	1	8302h	Keine Empfangsressourcen bei Ziel-Station, Empfänger-Station kann empfangene Daten nicht schnell genug verarbeiten bzw. hat keine Empfangsressourcen bereitgestellt.
0	-	1	8304h	Die Verbindung ist nicht aufgebaut. Der Sendeauftrag sollte erst nach einer Wartezeit > 100ms erneut abgesetzt werden.
-	0	1	8304h	Die Verbindung ist nicht aufgebaut. Der Empfangsauftrag sollte erst nach einer Wartezeit > 100ms erneut abgesetzt werden.
0	-	1	8311h	Zielstation ist unter der angegebenen Ethernet-Adresse nicht erreichbar.
0	-	1	8312h	Ethernet-Fehler im CP
0		1	8F22h	Quell-Bereich ungültig, wenn beispielsweise Bereich im DB nicht vorhanden Parameter <i>LEN</i> < 0.
-	0	1	8F23h	Quell-Bereich ungültig, wenn beispielsweise Bereich im DB nicht vorhanden Parameter <i>LEN</i> < 0.
0	-	1	8F24h	Bereichsfehler beim Lesen eines Parameters.
-	0	1	8F25h	Bereichsfehler beim Schreiben eines Parameters.
0	-	1	8F28h	Ausrichtungsfehler beim Lesen eines Parameters.
-	0	1	8F29h	Ausrichtungsfehler beim Schreiben eines Parameters.
-	0	1	8F30h	Parameter liegt im schreibgeschützten 1. akt. Datenbaustein
-	0	1	8F31h	Parameter liegt im schreibgeschützten 2. akt. Datenbaustein
0	0	1	8F32h	Parameter enthält zu große DB-Nummer.
0	0	1	8F33h	DB-Nummer Fehler
0	0	1	8F3Ah	Bereich nicht geladen (DB)
0	-	1	8F42h	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
-	0	1	8F43h	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
0	-	1	8F44h	Adresse des zu lesenden Parameters in der Zugriffsspur gesperrt.
-	0	1	8F45h	Adresse des zu schreibenden Parameters in der Zugriffsspur gesperrt.

DONE (SEND)	NDR (RECV)	ERROR	STATUS	Beschreibung
0	0	1	8F7Fh	Interner Fehler z.B. unzulässige ANY-Referenz z.B. Parameter <i>LEN</i> = 0.
0	0	1	8090h	Baugruppe mit dieser Baugruppen-Anfangsadresse nicht vorhanden oder CPU in STOP.
0	0	1	8091h	Baugruppen-Anfangsadresse nicht auf Doppel-Wort-Raster.
0	0	1	8092h	In ANY-Referenz ist eine Typangabe ungleich BYTE angegeben.
-	0	1	80A0h	Negative Quittung beim Lesen von Baugruppe.
0	0	1	80A4h	reserviert
0	0	1	80B0h	Baugruppe kennt den Datensatz nicht.
0	0	1	80B1h	Die Längenangabe (im Parameter <i>LEN</i> ) ist falsch.
0	0	1	80B2h	reserviert
0	0	1	80C0h	Datensatz kann nicht gelesen werden.
0	0	1	80C1h	Der angegebene Datensatz ist gerade in Bearbeitung.
0	0	1	80C2h	Es liegt ein Auftragsstau vor.
0	0	1	80C3h	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	0	1	80C4h	Kommunikationsfehler (tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll).
0	0	1	80D2h	Baugruppen-Anfangsadresse ist falsch.

**Status-Parameter bei Neuanlauf**

Bei einem Neuanlauf des CP werden die Ausgabe-Parameter wie folgt zurückgesetzt:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS = 8180h (bei AG\_RECV)
- STATUS = 8181h (bei AG\_SEND)

**5.2.4 FC 10 - AG\_CNTRL - Control CP 343****Beschreibung**

Mit dem FC 10 von VIPA besteht die Möglichkeit, Verbindungen des Ethernet-CP 343 zu diagnostizieren bzw. einen neuen Verbindungsaufbau zu initialisieren.

Folgende Aktionen können über parametrierbare Kommandos ausgeführt werden:

- Auslesen von Verbindungsinformationen
- Rücksetzen von projektierten Verbindungen

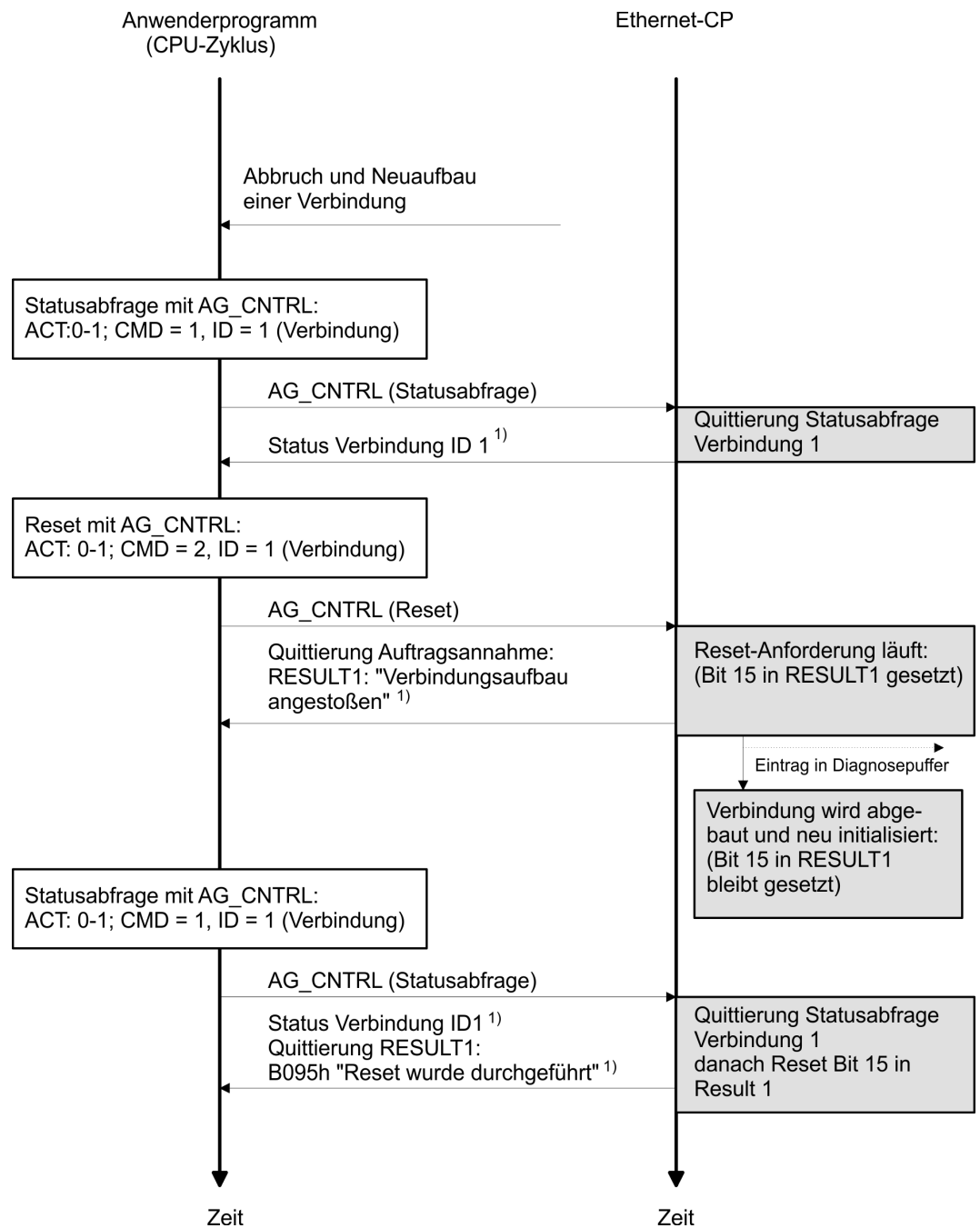
Die Kommandos dieses Bausteins sind nur für SEND-/RECV-Verbindungen zulässig, die auf die Protokolle ISO/RFC/TCP und UDP aufsetzen.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 196 AG\_CNTRL aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

## FC 10 im Anwenderprogramm

Die folgende Darstellung zeigt einen typischen Ablauf von AG\_CNTRL. Hier sehen Sie, wie zunächst der Verbindungsstatus abgefragt und in einem zweiten Auftrag der Verbindungsaufbau mittels Reset-Kommando veranlasst wird.



1) Parameterübergabe DONE, ERROR, STATUS und RESULT1/2

## Parameter

Parameter	Deklaration	Datentyp	Beschreibung
ACT	INPUT	BOOL	Auftrag wird durch Flankenwechsel 0-1 von ACT ausgeführt.
ID	INPUT	INT	Verbindungsnummer aus der Projektierung
LADDR	INPUT	WORD	Anfangsadresse des CP in der Hardware-Konfiguration

Parameter	Deklaration	Datentyp	Beschreibung
CMD	INPUT	INT	Kommando-Nr.
DONE	OUTPUT	BOOL	Zustandsparameter des Auftrags
ERROR	OUTPUT	BOOL	Fehler des Auftrags
STATUS	OUTPUT	WORD	Status des Auftrags
RESULT1	OUTPUT	DWORD	Rückmeldung 1 gemäß Kommando
RESULT2	OUTPUT	DWORD	Rückmeldung 2 gemäß Kommando

<b>ACT</b>	<p>Wertebereich: 0, 1</p> <p>Der Aufruf des FC muss mit Flankenwechsel 0-1 von <i>ACT</i> erfolgen.</p> <p>Beim Aufruf mit <i>ACT</i> = 0 erfolgt kein Funktionsaufruf und der Baustein wird sofort wieder verlassen.</p>
<b>ID</b>	<p>Wertebereich: 1, 2 ... n, oder 0</p> <p>Im Parameter <i>ID</i> wird die Verbindungsnummer der Verbindung angegeben. Die Verbindungsnummer ist der Projektierung zu entnehmen. Die maximale Anzahl der Verbindungen ist n.</p> <p>Bei einem Aufruf, der alle Verbindungen anspricht, muss als <i>ID</i> 0 angegeben werden (_ALL-Funktion mit <i>CMD</i> 3 bzw. <i>CMD</i> 4).</p>
<b>LADDR</b>	<p>Baugruppen-Anfangsadresse</p> <p>Bei der Konfiguration des CP im Hardware-Konfigurator wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben.</p> <p>Geben Sie diese Adresse hier an.</p>
<b>CMD</b>	Kommandos an den FC AG_CNTRL
<b>DONE</b>	<p>0: Auftrag ist noch in Bearbeitung bzw. noch nicht angestoßen</p> <p>1: Auftrag ist ausgeführt</p> <p>Der Zustandsparameter zeigt an, ob der Auftrag fehlerfrei abgewickelt wurde.</p> <p>Bei <i>DONE</i> = 1 kann <i>RESULT</i> ausgewertet werden.</p>
<b>ERROR</b>	<p>0: kein Fehler</p> <p>1: Fehleranzeige</p>
<b>STATUS</b>	Statusanzeige
<b>RESULT1/2</b>	Rückmeldung gemäß Kommando an den FC AG_CNTRL
<b>DONE, ERROR, STATUS</b>	In der nachfolgenden Tabelle sind alle Meldungen aufgeführt, die der Ethernet-CP 343 nach einem AG_CNTRL zurückliefern kann.

Zusätzlich sind die Kommando-Ergebnisse in den Parametern *RESULT1* und *RESULT2* auszuwerten.

DONE	ERROR	STATUS	Beschreibung
1	0	0000h	Auftrag fertig ohne Fehler
0	0	0000h	Kein Auftrag in Bearbeitung
0	0	8181h	Auftrag läuft, der Bausteinanruf ist mit den gleichen Parametern zu wiederholen, bis <i>DONE</i> oder <i>ERROR</i> gemeldet werden.
0	1	8183h	Für diesen Auftrag gibt es keine CP-Projektierung oder der Dienst im Ethernet-CP 343 ist noch nicht gestartet.
0	1	8186h	Parameter <i>ID</i> ungültig. Die zulässige <i>ID</i> ist abhängig vom gewählten Kommando.
0	1	8187h	Der Parameter <i>CMD</i> ist ungültig
0	1	8188h	Sequenzfehler bei der <i>ACT</i> -Steuerung
0	1	8090h	Baugruppe mit dieser Baugruppen-Anfangsadresse nicht vorhanden oder CPU in STOP.
0	1	8091h	Baugruppen-Anfangsadresse nicht auf Doppel-Wort Raster.
0	1	80B0h	Baugruppe kennt den Datensatz nicht.
0	1	80C0h	Datensatz kann nicht gelesen werden.
0	1	80C1h	Der angegebene Datensatz ist gerade in Bearbeitung.
0	1	80C2h	Es liegt ein Auftragsstau vor.
0	1	80C3h	Die Betriebsmittel (Speicher) der CPU sind temporär belegt.
0	1	80C4h	Kommunikationsfehler (tritt temporär auf; daher ist eine Wiederholung im Anwenderprogramm sinnvoll).
0	1	80D2h	Baugruppen-Anfangsadresse ist falsch.

#### Status-Parameter bei Neuanlauf

Bei einem Neuanlauf des CP werden die Ausgabe-Parameter wie folgt zurückgesetzt:

- *DONE* = 0
- *NDR* = 0
- *ERROR* = 8180h (bei AG\_RECV)
- *ERROR* = 8181h (bei AG\_SEND)



#### Hinweis!

Bitte beachten Sie, dass Sie den Baustein nur dann mit neuen Parametern aufrufen können, wenn ein zuvor gestarteter Auftrag mit *DONE* = 1 abgeschlossen wurde.

#### Kommandos und Auswertung der Auftragsergebnisse

In der nachfolgenden Tabelle finden Sie die möglichen Kommandos und die in den Parametern *RESULT1* und *RESULT2* auswertbaren Ergebnisse.

#### CMD 0

NOP - no operation

Der Baustein wird ohne Auftrag an den CP durchlaufen.

RESULT	Hex-Wert/Bereich	Beschreibung
RESULT 1	0000 0001h	Ablauf ohne Fehler
RESULT 2	0000 0000h	Default

**CMD 1**

CN\_STATUS - connection status

Dieses Kommando liefert den Status der mit *ID* gewählten Verbindung des über *LADDR* ausgewählten CP. Sollte das Bit 15 (RESET-Kennung) gesetzt sein, wird dieses automatisch zurückgesetzt (dieses Verhalten entspricht CMD 5 - CN\_CLEAR\_RESET).

RESULT	Hex-Wert/Bereich	Beschreibung
RESULT 1	0000 000xh	Bit 3 ... 0: Anzeige Senderichtung (ausgeschlossen: 0010 <sub>b</sub> ) Bit 0: Sende/Empfangsverbindung reserviert Bit 1: Sendeauftrag in Bearbeitung Bit 3, 2: Vorangegangener Auftrag 00: Keine Information 01: Sendeauftrag positiv abgeschlossen 10: Sendeauftrag negativ abgeschlossen
	0000 00x0h	Bit 7 ... 4: Anzeige Empfangsrichtung (ausgeschlossen: 0010 <sub>b</sub> ) Bit 4: Sende/Empfangsverbindung reserviert Bit 5: Empfangsauftrag in Bearbeitung Bit 7, 6: Vorangegangener Auftrag 00: Keine Information 01: Empfangsauftrag positiv abgeschlossen 10: Empfangsauftrag negativ abgeschlossen

RESULT	Hex-Wert/Bereich	Beschreibung
	0000 0x00h	<p>Bit 11 ... 8: Anzeigen für FETCH/WRITE (ausgeschlossen: 0011<sub>b</sub>, 0111<sub>b</sub>, 1000<sub>b</sub>, 1011<sub>b</sub>, 0010<sub>b</sub>)</p> <p>Bit 8: Verbindungstyp 0: keine FETCH-Verbindung 1: Verbindung für FETCH-Aufträge reserviert</p> <p>Bit 9: Verbindungstyp 0: keine WRITE-Verbindung 1: Verbindung für WRITE-Aufträge reserviert</p> <p>Bit 10: Auftragsstatus (FETCH/ WRITE) 0: Auftragsstatus OK 1: Auftragsstatus nicht OK</p> <p>Diese Kennung wird in folgenden Fällen gesetzt:</p> <ul style="list-style-type: none"> <li>- Der Auftrag wurde von der CPU negativ quittiert</li> <li>- Der Auftrag wurde von der CPU negativ quittiert</li> <li>- Der Auftrag konnte nicht zur CPU weitergeleitet werden, weil sich die Verbindung im Zustand "LOCKED" befand.</li> <li>- Der Auftrag wurde abgelehnt, weil der FETCH/WRITE-Header nicht korrekt aufgebaut war.</li> </ul> <p>Bit 11: Status FETCH/WRITE-Auftrag 0: es läuft kein Auftrag 1: es läuft ein Auftrag von LAN</p>
	0000 x000h	<p>Bit 15 ... 12: Allgemeine CP-Informationen (ausgeschlossen: 0011<sub>b</sub>, 1011<sub>b</sub>)</p> <p>Bit 13, 12: Verbindungszustand (nur für SEND/RECV-Verbindungen verfügbar, die auf die Protokolle ISO/RFC/TCP aufsetzen; bei UDP werden die entsprechenden internen Informationen ausgegeben) 00: Verbindung ist abgebaut 01: Verbindungsaufbau läuft 10: Verbindungsabbau läuft 11: Verbindung ist aufgebaut</p> <p>Bit 14: CP-Information 0: CP in STOP 1: CP in RUN</p> <p>Bit 15: Reset-Kennung 0: Es wurde durch den FC 10 noch kein Verbindungs-Reset durchgeführt bzw. die Reset-Kennung wurde zurück genommen. 1: Es wurde durch den FC 10 ein Verbindungs-Reset durchgeführt.</p>
	xxxx 0000h	Bit 31 ... 16: Reserviert für spätere Erweiterungen
RESULT 2	0000 0000h	Reserviert für spätere Erweiterungen



**CMD 2****CN\_RESET** - connection reset

Dieses Kommando setzt die mit *ID* gewählte Verbindung des über *LADDR* ausgewählten CP zurück.

Das Rücksetzen der Verbindung bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung).

Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.

RESULT	Hex-Wert/Bereich	Beschreibung
RESULT 1	0000 0001h	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau wurden angestoßen.
	0000 0002h	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z.B. CP in STOP).
RESULT 2	0000 0000h	Default

**CMD 3****CN\_STATUS\_ALL** - all connections status

Dieses Kommando liefert in den Parametern *RESULT1/2* (insgesamt 8Byte Sammelinformation) den Verbindungsstatus aller Verbindungen (aufgebaut/abgebaut) des über *LADDR* ausgewählten CP.

Hierbei muss der Parameter *ID* auf "0" gesetzt sein (wird auf "0" überprüft).

Detailinformationen einer abgebauten oder nicht projektierten Verbindung erhält man bei Bedarf über einen erneuten, auf die Verbindung gezielten Status-Aufruf mit *CMD* = 1.

RESULT	Hex-Wert/Bereich	Beschreibung
RESULT 1	xxxx xxxxh	32 Bit: Verbindung 1 ... 32 0: Verbindung abgebaut / nicht projektiert 1: Verbindung aufgebaut
RESULT 2	xxxx xxxxh	32 Bit: Verbindung 33 ... 64 0: Verbindung abgebaut / nicht projektiert 1: Verbindung aufgebaut

**CMD 4****CN\_RESET\_ALL** - all connections reset

Dieses Kommando setzt alle Verbindungen des über *LADDR* angewählten CP zurück.

Der Parameter *ID* muss auf "0" gesetzt sein (wird auf "0" überprüft).

Das Rücksetzen der Verbindung bewirkt einen Verbindungsabbruch und einen erneuten Verbindungsaufbau (aktiv oder passiv, abhängig von der Projektierung).

Es wird zusätzlich ein Eintrag im Diagnosepuffer erzeugt, dem das Auftragsergebnis zu entnehmen ist.

RESULT	Hex-Wert/ Bereich	Beschreibung
RESULT 1	0000 0001h	Der Reset-Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch und der anschließende Verbindungsaufbau aller Verbindungen wurden angestoßen.
	0000 0002h	Der Reset-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z.B. CP in STOP).
RESULT 2	0000 0000h	Default

#### CMD 5

CN\_CLEAR\_RESET - clear the reset *ID*

Dieses Kommando setzt die Reset-Kennung (Bit 15 in RESULT1) für die mit *ID* gewählte Verbindung des über *LADDR* ausgewählten CP zurück.

Dieser Auftrag wird auch automatisch beim Lesen des Verbindungsstatus ausgeführt (*CMD* = 1); der hier beschriebene separat absetzbare Auftrag ist nur in Sonderfällen erforderlich.

RESULT	Hex-Wert/ Bereich	Beschreibung
RESULT 1	0000 0001h	Der Clear-Auftrag wurde erfolgreich an den CP übertragen.
	0000 0002h	Der Clear-Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z.B. CP in STOP).
RESULT 2	0000 0000h	Default

#### CMD 6

CN\_DISCON - connection disconnect

Dieses Kommando setzt die Verbindung zurück, die mit *ID* und *LADDR* ausgewählt wurde. Das Rücksetzen der Verbindung wird durch einen Verbindungsabbruch durchgeführt.

Eventuell im Stack gespeicherte Daten gehen ohne Hinweis verloren. Im Anschluss erfolgt kein automatischer Aufbau der Verbindung. Die Verbindung kann durch den Control-Auftrag CN\_STARTCON wieder aufgebaut werden. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftragsergebnis entnehmen können.

RESULT	Hex-Wert/ Bereich	Beschreibung
RESULT 1	0000 0001h	Der Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsabbruch wurde eingeleitet.
	0000 0002h	Der Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z.B. CP in STOP).
RESULT 2	0000 0000h	Default

**CMD 7**

CN\_STARTCON - start connection

Dieses Kommando baut eine Verbindung auf, die mit *ID* und *LADDR* ausgewählt und zuvor mit dem Control-Auftrag CN\_DISCON abgebrochen wurde. Es wird ein Diagnosepuffereintrag erzeugt, dem Sie das Auftragsergebnis entnehmen können.

RESULT	Hex-Wert/ Bereich	Beschreibung
RESULT 1	0000 0001h	Der Auftrag wurde erfolgreich an den CP übertragen. Der Verbindungsaufbau wurde eingeleitet.
	0000 0002h	Der Auftrag konnte nicht an den CP übertragen werden, da der Dienst im CP nicht gestartet ist (z.B. CP in STOP).
RESULT 2	0000 0000h	Default

**5.2.5 FC 62 - C\_CNTR - Zustand einer Verbindung abfragen****Beschreibung**

Mit dem FC 62 ermitteln Sie den Zustand einer Verbindung. Nach Aufruf der Systemfunktion mit dem Wert 1 am Steuereingang *EN\_R* wird der momentane Zustand der über *ID* adressierten Verbindung ermittelt. Intern wird der SFC196 aufgerufen.

**Parameter**

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	INPUT	BOOL	E, A, M, D, L, Konst.	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft, wenn der Eingang gesetzt ist.
ID	INPUT	WORD	M, D, Konst.	Adressierungsparameter <i>ID</i>
RET_VAL	OUTPUT	INT	E, A, M, D, L	Fehlerinformation
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> und <i>STATUS</i>
STATUS	OUTPUT	WORD	E, A, M, D, L	<ul style="list-style-type: none"> <li>■ <i>ERROR</i>=0 und <i>STATUS</i> hat den Wert: <ul style="list-style-type: none"> <li>– 0000h: weder Warnung noch Fehler</li> <li>– &lt;&gt; 0000h: Warnung, <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i>=1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor. <i>STATUS</i> liefert detaillierte Auskunft über die Art des Fehlers.</li> </ul> </li> </ul>
C_CONN	OUTPUT	BOOL	E, A, M, D, L	Zustand der zugehörigen Verbindung. Mögliche Werte: <ul style="list-style-type: none"> <li>■ 0: Verbindung abgebrochen oder nicht aufgebaut.</li> <li>■ 1: Verbindung vorhanden.</li> </ul>
C_STATUS	OUTPUT	WORD	E, A, M, D, L	Verbindungszustand: <ul style="list-style-type: none"> <li>■ W#16#0000: Verbindung ist nicht eingerichtet</li> <li>■ W#16#0001: Verbindung wird gerade eingerichtet</li> <li>■ W#16#0002: Verbindung ist eingerichtet</li> <li>■ W#16#000F: Keine Daten für Verbindungsstatus verfügbar (z. B. beim CP-Anlauf)</li> <li>■ W#16#00FF: Verbindung ist nicht projektiert</li> </ul>

## Fehlerinformationen

Der Ausgangsparameter *RET\_VAL* kann bei der FC 62 C\_CNTRL die folgenden beiden Werte annehmen:

- 0000h: Bei der Ausführung des FC ist kein Fehler aufgetreten.
- 8000h: Bei der Ausführung des FC ist ein Fehler aufgetreten.



*Auch dann, wenn im Ausgangsparameter *RET\_VAL* der Wert 0000h angezeigt wird, sind die Ausgangsparameter *ERROR* und *STATUS* auszuwerten.*

ERROR	STATUS (dezimal)	Beschreibung
1	10	Fehler beim Zugriff auf den CP, weil gerade ein anderer Auftrag läuft. Auftragswiederholung zu einem späteren Zeitpunkt.
1	27	Für diesen Baustein existiert in der CPU kein Funktionscode.

## 5.2.6 FB/SFB 8 - FB 55 - Übersicht

Mit der Siemens S7-Kommunikation können Sie größere Datenmengen zwischen über Ethernet verbundenen SPS-Systemen auf Basis von Siemens STEP7® übertragen. Die Kommunikationsverbindungen sind statisch, d.h. sie sind über eine Verbindungstabelle zu projektieren.

## Möglichkeiten der Kommunikationsfunktionen

- Siemens S7-300-Kommunikationsfunktionen
  - Durch Einbindung der VIPA-spezifischen Funktionsbausteine FB 8 ... FB 55 können Sie auf die Siemens S7-300-Kommunikationsfunktionen zugreifen.  
 ↳ Kapitel 3 "Bibliothek einbinden" auf Seite 7
- Siemens S7-400-Kommunikationsfunktionen
  - Für die Siemens S7-400-Kommunikationsfunktionen verwenden Sie die SFB 8 ... SFB 23, die im Betriebssystem der CPU integriert sind. Hierzu kopieren Sie die Schnittstellenbeschreibung der SFBs aus der Siemens Standard-Bibliothek in das Verzeichnis "Bausteine", generieren für jeden Aufruf einen Instanzen-Datenbaustein und rufen den SFB mit dem zugehörigen Instanzen-Datenbaustein auf.

## Projektierung

Voraussetzung für die Siemens S7-Kommunikation ist eine projektierte Verbindungstabelle, in der die Kommunikationsverbindungen definiert werden. Hierzu können Sie beispielsweise WinPLC7 von VIPA oder NetPro von Siemens verwenden. Eine Kommunikationsverbindung ist durch eine Verbindungs-ID für jeden Kommunikationspartner spezifiziert. Die lokale ID verwenden Sie für die Parametrierung des FB/SFB der SPS von der aus die Verbindung betrachtet wird und die Partner-ID für die Parametrierung des FB/SFB in der Partner-SPS.

## Funktionsbausteine

FB/SFB	Bezeichnung	Beschreibung
FB/SFB 8	USEND	Unkoordiniertes Senden
FB/SFB 9	URCV	Unkoordiniertes Empfangen
FB/SFB 12	BSEND	Blockorientiertes Senden
FB/SFB 13	BRCV	Blockorientiertes Empfangen
FB/SFB 14	GET	Remote CPU lesen

FB/SFB	Bezeichnung	Beschreibung
FB/SFB 15	PUT	Remote CPU schreiben
FB 55	IP_CONF	Programmierbare Verbindungen



Bitte verwenden Sie für die Siemens S7-Kommunikation ausschließlich die hier aufgeführten FB/SFBs. Der direkte Aufruf der zugehörigen internen FC/SFCs führt zu Fehlern im entsprechenden Instanz-DB!

## 5.2.7 FB/SFB 8 - USEND - Unkoordiniertes Senden

### Beschreibung

Mit dem FB/SFB 8 USEND können Daten an einen remoten Partner- FB/SFB vom Typ URCV (FB/SFB 9) gesendet werden, wobei darauf zu achten ist, dass der Parameter *R\_ID* bei beiden FB/SFBs identisch ist. Der Sendevorgang wird mit einer positiven Flanke am Steuereingang *REQ* gestartet und verläuft ohne Koordination mit dem Partner-FB/SFB.

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- Siemens S7-300-Kommunikation (FB 8)
  - Der Sendevorgang erfolgt nach einer positiven Flanke an *REQ*. Mit jeder positiven Flanke an *REQ* werden die Parameter *R\_ID*, *ID* und *SD\_1* übernommen. Nach Abschluss eines Auftrags können Sie den Parametern *R\_ID*, *ID* und *SD\_1* neue Werte zuweisen.
- Siemens S7-400-Kommunikation (SFB 8)
  - Der Sendevorgang erfolgt nach einer positiven Flanke an *REQ*. Die zu sendenden Daten werden durch die Parameter *SD\_1* ... *SD\_4* referenziert, wobei diese vier Sendeparameter nicht alle belegt sein müssen.

### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke (gegenüber letztem FB/SFB-Aufruf)
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. <i>ID</i> muss in der Form wxyzh angegeben werden.
R_ID	INPUT	DWORD	E, A, M, D, L, Konstante	Adressierungsparameter <i>R_ID</i> . Ist in der Form DW#16#wxyzWXYZ anzugeben.
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>■ 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>■ 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> = 0000h <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> ungleich 0000h <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i> = 1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
SD_i, 1 ≤ i ≤ 4	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf i-ten Sendebereich.  Zulässig sind nur die Datentypen: BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.



Bitte beachten Sie, dass die über die Parameter *SD\_1/SD\_1...SD\_4* und *RD\_1/RD\_1...RD\_4* (beim zugehörigen Partner-FB/SFB URCV) definierten Bereiche übereinstimmen in Anzahl, Länge und Datentyp.

Der Parameter *R\_ID* muss bei beiden FB/SFBs identisch sein. Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter *DONE* mit einer logischen 1 angezeigt.

#### Fehlerinformationen

ERROR	STATUS (dezimal)	Bedeutung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z.B. <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B. Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	4	Fehler in den Sendebereichszeigern <i>SD_i</i> bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>■ ein Instanz-DB, der nicht zum FB/SFB 8 gehört, angegeben</li> <li>■ kein Instanz-DB, sondern ein Global-DB angegeben</li> <li>■ kein Instanz-DB gefunden (Laden eines neuen Instanz-DB vom PG)</li> </ul>

ERROR	STATUS (dezimal)	Bedeutung
1	18	$R\_ID$ existiert bereits in der Verbindung $ID$ .
1	20	Zu wenig Arbeitsspeicher vorhanden.

**Datenkonsistenz**

Um sicherzustellen, dass die Datenkonsistenz nicht beeinträchtigt wird, dürfen die aktuell benutzten Sendebereiche  $SD\_i$  erst dann wieder beschrieben werden, wenn der aktuelle Sendevorgang abgeschlossen ist. Dazu muss der Parameter  $DONE$  ausgewertet werden. Der gesamte Sendevorgang ist erst dann abgeschlossen, wenn der Zustandsparameter  $DONE$  den Wert 1 annimmt.

**5.2.8 FB/SFB 9 - URCV - Unkoordiniertes Empfangen****Beschreibung**

Mit dem FB/SFB 9 URCV können Daten asynchron von einem remoten Partner-FB/SFB vom Typ USEND (FB/SFB 8) empfangen werden, wobei darauf zu achten ist, dass der Parameter  $R\_ID$  bei beiden FB/SFBs identisch ist. Der Baustein ist empfangsbereit, wenn am Eingang  $EN\_R$  eine logische 1 anliegt. Mit  $EN\_R=0$  kann ein laufender Auftrag abgebrochen werden.

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- Siemens S7-300-Kommunikation (FB 9)
  - Mit jeder positiven Flanke an  $EN\_R$  werden die Parameter  $R\_ID$ ,  $ID$  und  $RD\_1$  übernommen. Nach Abschluss eines Auftrags können Sie den Parametern  $R\_ID$ ,  $ID$  und  $RD\_1$  neue Werte zuweisen.
- Siemens S7-400-Kommunikation (SFB 9)
  - Die Empfangsdatenbereiche werden durch die Parameter  $RD\_1...RD\_4$  referenziert.

**Parameter**

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	INPUT	BOOL	E, A, M, D, L	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. $ID$ muss in der Form wxyz angegeben werden.
R_ID	INPUT	DWORD	E, A, M, D, L, Konstante	Adressierungsparameter $R\_ID$ . Er muss in der Form DW#16#wxyzWXYZ angegeben werden.
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter $NDR$ : neue Daten übernommen.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter $ERROR$ : <ul style="list-style-type: none"> <li>■ <math>ERROR = 0 + STATUS = 0000h</math> <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <math>ERROR = 0 + STATUS</math> ungleich 0000h               <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <math>STATUS</math> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <math>ERROR = 1</math> <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>



Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
RD_i, $1 \leq i \leq 4$	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf i-ten Empfangsbereich.  Zulässig sind nur die Datentypen: BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.



Es ist darauf zu achten, dass die über die Parameter *SD\_i* und *RD\_i*,  $1 \leq i \leq 4$ , definierten Bereiche in Anzahl, Länge und im Datentyp übereinstimmen (*SD\_i* ist der Sendebereich des zugehörigen Partner-FB/SFB, siehe FB/SFB 8). Beim Erstaufwurf des FB/SFB 9 wird das "Empfangsfach" angelegt, bei den Nachfolgerufen müssen die zu empfangenden Daten in dieses Empfangsfach hineinpassen. Wird der Kopiervorgang erfolgreich abgeschlossen, hat der Parameter *NDR* den Wert 1.

## Fehlerinformationen

ERROR	STATUS (dezimal)	Beschreibung
0	9	Overrun-Warnung: Ältere Empfangsdaten wurden von neueren Empfangsdaten überschrieben.
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z.B. <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B. Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	4	Fehler in den Empfangsbereichszeigern <i>RD_i</i> bezüglich der Datenlänge oder des Datentyps.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>■ ein Instanz-DB, der nicht zum FB/SFB 9 gehört, angegeben</li> <li>■ kein Instanz-DB, sondern ein Global-DB angegeben</li> <li>■ kein Instanz-DB gefunden (Laden eines neuen Instanz-DB vom PG)</li> </ul>
1	18	<i>R_ID</i> existiert bereits in der Verbindung <i>ID</i> .
1	19	Der zugehörige FB/SFB USEND sendet schneller Daten als diese vom FB/SFB URCV in die Empfangsbereiche kopiert werden können.
1	20	Zu wenig Arbeitsspeicher vorhanden.



**Datenkonsistenz**

Die Daten werden konsistent empfangen, wenn Sie folgendes beachten:

- Siemens S7-300-Kommunikation:
  - Nachdem der Zustandsparameter *NDR* den Wert 1 angenommen hat, müssen Sie den FB 9 URCV mit dem Wert 0 an *EN\_R* sofort erneut aufrufen. Damit wird gewährleistet, dass der Empfangsbereich nicht bereits überschrieben wird, bevor Sie ihn ausgewertet haben. Werten Sie den Empfangsbereich *RD\_1* vollständig aus, bevor Sie den Baustein wieder mit dem Wert 1 am Steuereingang *EN\_R* aufrufen.
- Siemens S7-400-Kommunikation:
  - Nachdem der Zustandsparameter *NDR* den Wert 1 angenommen hat, befinden sich neue Empfangsdaten in Ihren Empfangsbereichen (*RD\_i*). Ein erneuter Bausteinanruf kann diese Daten mit neuen Empfangsdaten überschreiben. Wenn Sie dies verhindern wollen, müssen Sie den SFB 9 URCV mit dem Wert 0 an *EN\_R* so lange aufrufen (z.B. bei zyklischer Bausteinbearbeitung), bis Sie die Bearbeitung der Empfangsdaten beendet haben.

**5.2.9 FB/SFB 12 - BSEND - Blockorientiertes Senden****Beschreibung**

Mit dem FB/SFB 12 BSEND können Daten an einen remoten Partner- FB/SFB vom Typ BRCV (FB/SFB 13) gesendet werden. Der zu sendende Datenbereich wird segmentiert. Jedes Segment wird einzeln an den Partner gesendet. Das letzte Segment wird vom Partner bereits bei seiner Ankunft quittiert, unabhängig vom zugehörigen Aufruf des FB/SFB BRCV. Aufgrund der Segmentierung können Sie mit einem Sendeauftrag bis zu 65534Byte große Daten übertragen.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 202 AG\_BSEND aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- Siemens S7-300-Kommunikation (FB 12)
  - Der Sendevorgang erfolgt nach einer positiven Flanke an *REQ*. Mit jeder positiven Flanke an *REQ* werden die Parameter *R\_ID*, *ID*, *SD\_1* und *LEN* übernommen. Nach Abschluss eines Auftrags können Sie den Parametern *R\_ID*, *ID*, *SD\_1* und *LEN* neue Werte zuweisen. Zur Übertragung von segmentierten Daten ist der Baustein zyklisch im Anwenderprogramm aufzurufen. Die Anfangsadresse und die maximale Länge der zu sendenden Daten werden durch *SD\_1* vorgegeben. Die Länge des Datenblocks legen Sie auftragsbezogen durch *LEN* fest.
- Siemens S7-400-Kommunikation (SFB 12)
  - Die Aktivierung des Sendevorgangs erfolgt nach Aufruf des Bausteins und positiver Flanke an *REQ*. Das Senden der Daten aus dem Anwenderspeicher erfolgt asynchron zur Bearbeitung des Anwenderprogramms. Die Anfangsadresse der zu sendenden Daten wird durch *SD\_1* vorgegeben. Die Länge der Sendedaten legen Sie auftragsbezogen durch *LEN* fest. *LEN* ersetzt damit den Längenanteil von *SD\_1*.

**Funktion**

- Bei einer positiven Flanke am Steuereingang *R* wird ein laufender Sendevorgang abgebrochen.
- Der erfolgreiche Abschluss des Sendevorgangs wird am Zustandsparameter *DONE* mit 1 angezeigt.

- Nach Abschluss eines Sendevorganges, kann erst wieder ein neuer Sendeauftrag bearbeitet werden, wenn die Zustandsparameter *DONE* oder *ERROR* den Wert 1 angenommen haben.
- Aufgrund der asynchronen Datenübertragung kann ein erneutes Senden von Daten erst gestartet werden, wenn die vorhergehenden Daten durch Aufruf des Partner-FB/SFB abgeholt wurden. Bis die Daten abgeholt werden, wird beim Aufruf des FB/SFB BSEND der Statuswert 7 ausgegeben.



Der Parameter *R\_ID* muss bei den zusammengehörenden FB/SFBs identisch sein.

## Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke (gegenüber letztem FB/SFB-Aufruf)
R	INPUT	BOOL	E, A, M, D, L, Konstante	Steuerparameter reset: Abbruch des aktuellen Auftrags
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. <i>ID</i> muss in der Form W#16#xxxx angegeben werden.
R_ID	INPUT	DWORD	E, A, M, D, L, Konstante	Adressierungsparameter <i>R_ID</i> . Er muss in der Form DW#16#wxyzWXYZ angegeben werden.
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : <ul style="list-style-type: none"> <li>■ 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt.</li> <li>■ 1: Auftrag wurde fehlerfrei ausgeführt.</li> </ul>
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> = 0000h <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> ungleich 0000h <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i> = 1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
SD_1	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf Sendebereich. Die Längenangabe wird nur beim ersten Bausteinaufruf nach dem Start ausgewertet. Sie gibt die maximale Länge des Sendebereichs vor. Zulässig sind nur die Datentypen: BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.
LEN	IN_OUT	WORD	E, A, M, D, L	Länge des zu sendenden Datenblocks in Byte.

## Fehlerinformationen

ERROR	STATUS (dezimal)	Bedeutung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z.B.: <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B. Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	2	Negative Quittung vom Partner-FB/SFB. Die Funktion ist nicht ausführbar.
1	3	<i>R_ID</i> ist auf der durch <i>ID</i> vorgegebenen Kommunikationsverbindung nicht bekannt, oder der Empfangsbaustein wurde noch nie aufgerufen.
1	4	Fehler im Sendebereichszeiger <i>SD_1</i> bezüglich der Datenlänge oder des Datentyps, oder am Parameter <i>LEN</i> wurde der Wert 0 übergeben oder Fehler im Empfangsbereichszeiger <i>RD_1</i> des zugehörigen FB/SFB 13 BRCV
1	5	Resetanforderung wurde ausgeführt.
1	6	Partner-FB/SFB befindet sich im Zustand DISABLED ( <i>EN_R</i> hat den Wert 0).
1	7	Partner-FB/SFB befindet sich in falschem Zustand. (Der Empfangsbaustein wurde nach der letzten Datenübertragung nicht mehr aufgerufen).
1	8	Zugriff auf remoten Objekt im Anwenderspeicher wurde abgelehnt.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>■ ein Instanz-DB, der nicht zum FB/SFB 12 gehört, angegeben</li> <li>■ kein Instanz-DB, sondern ein Global-DB angegeben</li> <li>■ kein Instanz-DB gefunden (Laden eines neuen Instanz-DB vom PG)</li> </ul>
1	18	<i>R_ID</i> existiert bereits in der Verbindung <i>ID</i> .
1	20	Zu wenig Arbeitsspeicher vorhanden.

## Datenkonsistenz

Um sicherzustellen, dass die Datenkonsistenz nicht beeinträchtigt wird, darf der aktuell benutzte Teil des Sendebereichs *SD\_1* erst dann wieder beschrieben werden, wenn der aktuelle Sendevorgang abgeschlossen ist. Dazu muss der Parameter *DONE* ausgewertet werden.

## 5.2.10 FB/SFB 13 - BRCV - Blockorientiertes Empfangen

### Beschreibung

Mit dem FB/SFB 13 BRCV können Daten von einem remoten Partner- FB/SFB vom Typ BSEND (FB/SFB 12) empfangen werden, wobei darauf zu achten ist, dass der Parameter *R\_ID* bei beiden FB/SFBs identisch ist. Nach jedem empfangenen Datensegment wird eine Quittung an den Partner-FB/SFB geschickt, und der Parameter *LEN* aktualisiert.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 203 AG\_BRCV aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- **Siemens S7-300-Kommunikation (FB 13)**
  - Mit jeder positiven Flanke an *EN\_R* werden die Parameter *R\_ID*, *ID* und *RD\_1* übernommen. Nach Abschluss eines Auftrags können Sie den Parametern *R\_ID*, *ID* und *RD\_1* neue Werte zuweisen. Zur Übertragung von segmentierten Daten muss der Baustein zyklisch im Anwenderprogramm aufgerufen werden.
- **Siemens S7-400-Kommunikation (SFB 13)**
  - Der Empfang der Daten aus dem Anwenderspeicher erfolgt asynchron zur Bearbeitung des Anwenderprogramms.

### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
EN_R	INPUT	BOOL	E, A, M, D, L, Konstante	Steuerparameter enabled to receive, signalisiert Empfangsbereitschaft
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. <i>ID</i> muss in der Form W#16#xxxx angegeben werden.
R_ID	INPUT	DWORD	E, A, M, D, L, Konstante	Adressierungsparameter <i>R_ID</i> . Er muss in der Form DW#16#wxyzWXYZ angegeben werden.
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>NDR</i> : neue Daten übernommen.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> = 0000h <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> ungleich 0000h <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i> = 1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
RD_1	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf Empfangsbereich. Die Längenangabe gibt die maximale Länge des zu empfangenden Blocks vor. Zulässig sind nur die Datentypen: BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.
LEN	IN_OUT	WORD	E, A, M, D, L	Länge der bisher empfangenen Daten in Byte.

**Funktion**

- Der FB/SFB 13 ist empfangsbereit, wenn am Steuereingang *EN\_R* der Wert 1 anliegt. Durch den Parameter *RD\_1* wird die Anfangsadresse des Empfangsbereichs angegeben. Der Partner-FB/SFB erhält nach jedem empfangenen Datensegment eine Quittung und der Parameter *LEN* des FB/SFB 13 wird aktualisiert. Wenn der Baustein während des asynchronen Empfangs aufgerufen wird, erfolgt die Ausgabe einer Warnung im Zustandsparameter *STATUS*.
- Wenn dieser Aufruf mit dem Wert 0 am Steuereingang *EN\_R* erfolgt, wird der Empfangsvorgang abgebrochen, und der FB/SFB in seinen Grundzustand zurückversetzt. Wird der fehlerfreie Empfang sämtlicher Datensegmente erfolgreich abgeschlossen, hat der Parameter *NDR* den Wert 1. Die empfangenen Daten bleiben solange unverändert, bis der FB/SFB 13 erneut mit dem Parameter *EN\_R* = 1 aufgerufen wird.

**Fehlerinformationen**

ERROR	STATUS (dezimal)	Bedeutung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	17	Warnung: Baustein empfängt asynchron Daten.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.
1	1	Kommunikationsprobleme, z.B. <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B. Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	2	Funktion nicht ausführbar.
1	4	Fehler im Empfangsbereichszeiger <i>RD_1</i> bezüglich der Datenlänge oder des Datentyps (gesendeter Datenblock ist länger als der Empfangsbereich).
1	5	Resetanforderung eingetroffen, unvollständige Übertragung.
1	8	Zugriff auf remotes Objekt im Anwenderspeicher wurde abgelehnt.
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>■ ein Instanz-DB, der nicht zum FB/SFB 13 gehört, angegeben</li> <li>■ kein Instanz-DB, sondern ein Global-DB angegeben</li> <li>■ kein Instanz-DB gefunden (Laden eines neuen Instanz-DB vom PG)</li> </ul>
1	18	<i>R_ID</i> existiert bereits in der Verbindung <i>ID</i> .
1	20	Zu wenig Arbeitsspeicher vorhanden.

## Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie folgendes beachten:

- Nach Abschluss des Kopiervorganges (der Parameter *NDR* hat den Wert 1), muss der FB/SFB 13 erneut mit dem Wert 0 am Parameter *EN\_R* aufgerufen werden, um zu gewährleisten, dass der Empfangsbereich nicht bereits wieder überschrieben wird, bevor er ausgewertet wurde.
- Der zuletzt benutzte Empfangsbereich *RD\_1* muss vollständig ausgewertet werden, bevor der Baustein wieder empfangsbereit gemacht werden kann (Aufruf mit dem Wert 1 am Parameter *EN\_R*).

### Datenempfang Siemens S7-400

- Falls die Empfänger-CPU mit einem empfangsbereiten BRCV-Baustein (d.h. ein Aufruf mit *EN\_R* = 1 ist bereits erfolgt) in STOP geht, bevor der zugehörige Sendebaustein das erste Datensegment eines Auftrags abgeschickt hat, geschieht folgendes:
- Die Daten des ersten Auftrags nach Übergang der Empfänger-CPU in STOP werden vollständig in den Empfangsbereich eingetragen,
- Der Partner-SFB BSEND erhält darüber eine positive Quittung.
- Weitere BSEND-Aufträge können von der Empfänger-CPU im STOP-Zustand nicht mehr angenommen werden.
- Solange sie sich im STOP-Zustand befindet, haben *NDR* und *LEN* den Wert 0.
- Damit Ihnen die Information über die empfangenen Daten nicht verloren geht, müssen Sie bei der Empfänger-CPU einen Wiederanlauf durchführen und die SFB 13 BRCV mit *EN\_R* = 1 aufrufen.

## 5.2.11 FB/SFB 14 - GET - Remote CPU lesen

### Beschreibung

Mit dem FB/SFB 14 GET können Daten aus einer remoten CPU ausgelesen werden, wobei sich die CPU im Betriebszustand RUN oder STOP befinden kann.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 200 AG\_GET aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- **Siemens S7-300-Kommunikation (FB 14)**
  - Der Lesevorgang erfolgt nach einer positiven Flanke an *REQ*. Mit jeder positiven Flanke an *REQ* werden die Parameter *ID*, *ADDR\_1* und *RD\_1* übernommen. Nach Abschluss eines Auftrags können Sie den Parametern *ID*, *ADDR\_1* und *RD\_1* neue Werte zuweisen.
- **Siemens S7-400-Kommunikation (SFB 14)**
  - Bei einer positiven Flanke an *REQ* wird der SFB gestartet. Dabei werden die relevanten Zeiger auf die auszulesenden Bereiche (*ADDR\_i*) an die Partner-CPU gesendet.

### Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke (gegenüber letztem FB/SFB-Aufruf)
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. <i>ID</i> muss in der Form W#16#xxxx angegeben werden.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>NDR</i> : Daten aus Partner-CPU übernommen.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> = 0000h <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> ungleich 0000h <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i> = 1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
ADDR_1	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen
ADDR_2	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen
ADDR_3	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen
ADDR_4	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, die gelesen werden sollen
RD_i, 1 ≤ i ≤ 4	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der eigenen CPU, in der die gelesenen Daten abgelegt werden. Zulässig sind nur die Datentypen: BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.

### Funktion

- Die Remote-CPU sendet die Dateninhalte zurück und die Antwort wird daraufhin auf Zugriffsprobleme beim Lesen der Daten ausgewertet, des weiteren wird eine Datentypprüfung vorgenommen.
- Bei einer fehlerfreien Datenübertragung werden die empfangenen Daten beim nächsten Aufruf des FB/SFB 14 in die projizierten Empfangsbereiche (*RD\_i*) kopiert und der Parameter *NDR* erhält den Wert 1.
- Ein erneuter Lesevorgang kann erst dann wieder aktiviert werden, wenn der letzte Lesevorgang abgeschlossen wurde. Sie müssen darauf achten, dass die über die Parameter *ADDR\_i* und *RD\_i* definierten Bereiche in der Anzahl, in der Länge und im Datentyp zueinander passen.

### Fehlerinformationen

ERROR	STATUS (dezimal)	Bedeutung
0	11	Warnung: Neuer Auftrag ist unwirksam, da voran-gegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.



ERROR	STATUS (dezimal)	Bedeutung
1	1	Kommunikationsprobleme, z.B. <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B.: Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Empfangsbereichszeigern $RD_i$ bezüglich der Datenlänge oder des Datentyps.
1	8	Zugriffsfehler bei der Partner-CPU
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>■ ein Instanz-DB, der nicht zum FB/SFB 14 gehört, angegeben</li> <li>■ kein Instanz-DB, sondern ein Global-DB angegeben</li> <li>■ kein Instanz-DB gefunden (Laden eines neuen Instanz-DB vom PG)</li> </ul>
1	20	Zu wenig Arbeitsspeicher vorhanden.

## Datenkonsistenz

Die Daten werden konsistent empfangen, wenn Sie den aktuell verwendeten des Empfangsbereichs  $RD_i$  vollständig auswerten, bevor Sie einen erneuten Auftrag aktivieren.

## 5.2.12 FB/SFB 15 - PUT - Remote CPU schreiben

### Beschreibung

Mit dem FB/SFB 15 PUT können Daten in eine remote CPU geschrieben werden, wobei sich die CPU im Betriebszustand RUN oder STOP befinden kann.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 201 AG\_PUT aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

Je nach Kommunikationsfunktion haben Sie folgendes Verhalten:

- **Siemens S7-300-Kommunikation (FB 15)**
  - Der Sendevorgang erfolgt nach einer positiven Flanke an  $REQ$ . Mit jeder positiven Flanke an  $REQ$  werden die Parameter  $ID$ ,  $ADDR_1$  und  $SD_1$  übernommen. Nach Abschluss eines Auftrags können Sie den Parametern  $ID$ ,  $ADDR_1$  und  $SD_1$  neue Werte zuweisen.
- **Siemens S7-400-Kommunikation (SFB 15)**
  - Bei einer positiven Flanke an  $REQ$  wird der SFB gestartet. Dabei werden die Zeiger auf die zu schreibenden Bereiche ( $ADDR_i$ ) und die Daten ( $SD_i$ ) an die Partner-CPU gesendet.



## Parameter

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
REQ	INPUT	BOOL	E, A, M, D, L	Steuerparameter request, aktiviert den Datenaustausch bei steigender Flanke (gegenüber letztem FB/SFB-Aufruf)
ID	INPUT	WORD	E, A, M, D, Konstante	Referenz auf die Verbindung. <i>ID</i> muss in der Form W#16#xxxx angegeben werden.
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>DONE</i> : Funktion ausgeführt.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter <i>ERROR</i> : <ul style="list-style-type: none"> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> = 0000h <ul style="list-style-type: none"> <li>– Weder Warnung noch Fehler.</li> </ul> </li> <li>■ <i>ERROR</i> = 0 + <i>STATUS</i> ungleich 0000h <ul style="list-style-type: none"> <li>– Es liegt eine Warnung vor. <i>STATUS</i> liefert detaillierte Auskunft.</li> </ul> </li> <li>■ <i>ERROR</i> = 1 <ul style="list-style-type: none"> <li>– Es liegt ein Fehler vor.</li> </ul> </li> </ul>
STATUS	OUTPUT	WORD	E, A, M, D, L	Zustandsparameter <i>STATUS</i> , liefert detaillierte Auskunft über die Art des Fehlers.
ADDR_1	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll
ADDR_2	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll
ADDR_3	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll
ADDR_4	IN_OUT	ANY	z.B. E, A, M, D	Zeiger auf diejenigen Bereiche in der Partner-CPU, in die geschrieben werden soll
SD_i, 1 ≤ i ≤ 4	IN_OUT	ANY	E, A, M, D, T, Z	Zeiger auf diejenigen Bereiche in der eigenen CPU, die die zu versendenden Daten enthalten. Zulässig sind nur die Datentypen BOOL (nicht erlaubt: Bitfeld), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.

## Funktion

- Die Partner-CPU legt die gesendeten Daten unter den mitgeführten Adressen ab und sendet eine Ausführungsquittung zurück.
- Diese Ausführungsquittung wird ausgewertet und bei einer fehlerfreien Datenübertragung erhält der Parameter *DONE* beim nächsten Aufruf des FB/SFB 15 den Wert 1.
- Ein erneuter Schreibvorgang kann erst dann wieder aktiviert werden, wenn der letzte Schreibvorgang abgeschlossen wurde. Die über die Parameter *ADDR\_i* und *SD\_i*, 1 ≤ i ≤ 4, definierten Bereiche müssen in Anzahl, Länge und im Datentyp übereinstimmen.

## Fehlerinformationen

ERROR	STATUS (dezimal)	Bedeutung
0	11	Warnung: Neuer Auftrag ist unwirksam, da vorangegangener Auftrag noch nicht abgeschlossen ist.
0	25	Die Kommunikation wurde angestoßen. Der Auftrag ist in Bearbeitung.

ERROR	STATUS (dezimal)	Bedeutung
1	1	Kommunikationsprobleme, z.B. <ul style="list-style-type: none"> <li>■ Verbindungsbeschreibung nicht geladen (lokal oder remote)</li> <li>■ Verbindung unterbrochen (z.B.: Kabel, CPU ausgeschaltet, CP in STOP)</li> </ul>
1	2	Negative Quittung vom Partnergerät. Die Funktion ist nicht ausführbar.
1	4	Fehler in den Sendebereichszeigern <i>SD_i</i> bezüglich der Datenlänge oder des Datentyps.
1	8	Zugriffsfehler bei der Partner-CPU
1	10	Zugriff auf lokalen Anwenderspeicher nicht möglich (z.B. Zugriff auf gelöschten DB).
1	12	Beim Aufruf des FB/SFB wurde <ul style="list-style-type: none"> <li>ein Instanz-DB, der nicht zum FB/SFB 15 gehört, angegeben.</li> <li>kein Instanz-DB, sondern ein Global-DB angegeben.</li> <li>kein Instanz-DB gefunden (laden eines neuen Instanz-DB vom PG).</li> </ul>
1	20	Zu wenig Arbeitsspeicher vorhanden.

## Datenkonsistenz

- **Siemens S7-300-Kommunikation**
  - Um Datenkonsistenz zu gewährleisten, dürfen Sie den Sendebereich *SD\_1* erst dann wieder beschreiben, wenn der aktuelle Sendevorgang abgeschlossen ist. Dies ist der Fall, wenn der Zustandsparameter *DONE* den Wert "1" annimmt.
- **Siemens S7-400-Kommunikation**
  - Mit dem Aktivieren eines Sendevorgangs (steigende Flanke an *REQ*) sind die zu sendenden Daten der Sendebereiche *SD\_i* aus dem Anwenderprogramm kopiert. Sie können diese Bereiche nach dem Bausteinanruf neu beschreiben, ohne die aktuellen Sendedaten zu verfälschen.

## 5.2.13 FB 55 - IP\_CONF - Progr. Kommunikationsverbindungen

### 5.2.13.1 Übersicht

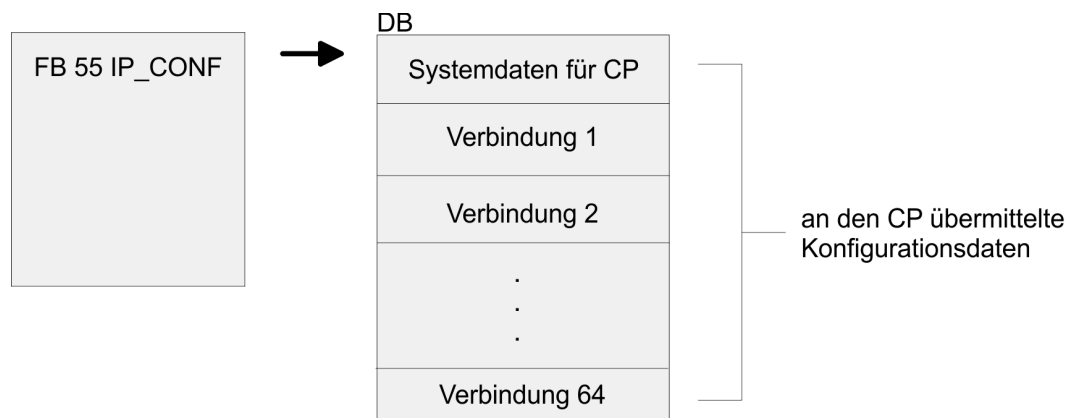
Der FB 55 - IP\_CONF ermöglicht zur Einrichtung flexibler Kommunikationsverbindungen die programmgesteuerte Übergabe von Datenbausteinen mit Projektierdaten an einen CP.



*Bitte beachten Sie, dass dieser Baustein intern den FC bzw. SFC 204 IP\_CONF aufruft. Dieser darf nicht überschrieben werden! Der direkte Aufruf eines internen Bausteins führt zu Fehler im entsprechenden Instanz-DB!*

## Prinzip

Über den FB 55, der im Anwenderprogramm aufgerufen wird, können Konfigurationsdaten für Kommunikationsverbindungen an den CP übertragen werden. Der Konfigurations-DB kann jederzeit in den CP geladen werden.

**VORSICHT!**

Sobald das Anwenderprogramm über den FB 55 IP\_CONF die Verbindungsdaten übergibt, schaltet die CPU den CP kurzzeitig in STOP. Der CP übernimmt die Systemdaten (inklusive IP-Adresse) und die neuen Verbindungsdaten und arbeitet diese im Anlauf ab (RUN).

**5.2.13.2 FB 55 - IP\_CONF**

Je nach Größe des Konfigurations-DB erfolgt die Übertragung zum CP in mehreren Segmenten. Sie müssen daher den FB solange erneut aufrufen, bis der FB mit dem *DONE*-Bit = 1 die vollständige Übertragung signalisiert. Der Auftrag wird ausgeführt, sobald *ACT* = 1 übergeben wird.

**Parameter**

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ACT	INPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> <li>Beim FB-Aufruf mit <i>ACT</i> = 1 wird der DBxx an den CP gesendet.</li> <li>Beim FB-Aufruf mit <i>ACT</i> = 0 werden nur die Statusanzeigen <i>DONE</i>, <i>ERROR</i> und <i>STATUS</i> aktualisiert.</li> </ul>
LADDR	INPUT	WORD	E, A, M, D, Konstante	Baugruppen-Anfangsadresse Bei der Hardware-Konfiguration wird die Baugruppen-Anfangsadresse in der Konfigurationstabelle ausgegeben. Geben Sie diese Adresse hier an.
CONF_DB	INPUT	ANY	E, A, M, D	Der Parameter zeigt die Anfangsadresse des Konfigurations-Datenbereichs in einem DB.
LEN	INPUT	INT	E, A, M, D, Konstante	Längenangabe in Byte für den Konfigurations-Datenbereich.
DONE	OUTPUT	BOOL	E, A, M, D, L	Der Parameter zeigt an, ob der Konfigurations-Datenbereich vollständig übertragen wurde. Beachten Sie, dass der FB je nach Größe des Konfigurations-Datenbereichs (in mehreren Zyklen) mehrfach angestoßen werden muss, bis die Anzeige <i>DONE</i> = 1 den Abschluss signalisiert.

Parameter	Deklaration	Datentyp	Speicherbereich	Beschreibung
ERROR	OUTPUT	BOOL	E, A, M, D, L	Fehleranzeige
STATUS	OUTPUT	WORD	E, A, M, D	Statusanzeige
EXT_STATUS	OUTPUT	WORD	E, A, M, D	<p>Der Parameter zeigt bei einer fehlerhaften Auftragsausführung an, welcher Parameter im Konfigurations-DB als Fehlerursache erkannt wurde.</p> <ul style="list-style-type: none"> <li>■ High-Byte: Index des Parameter-Blocks</li> <li>■ Low-Byte: Index des Subblocks innerhalb des Parameterblocks</li> </ul>

### Fehlerinformationen

ERROR	STATUS	Beschreibung
0	0000h	Auftrag fertig ohne Fehler
0	8181h	Auftrag läuft
1	80B1h	Anzahl der zu sendenden Daten überschreitet die für diesen Dienst zulässige Obergrenze.
1	80C4h	Kommunikationsfehler Fehler kann temporär auftreten, deshalb ist eine Wiederholung im Anwenderprogramm sinnvoll.
1	80D2h	Projektierungsfehler, die eingesetzte Baugruppe unterstützt diesen Dienst nicht.
1	8183h	CP lehnt die angeforderte Datensatznummer ab.
1	8184h	Systemfehler bzw. unzulässiger Parametertyp.
1	8185h	Wert des Parameters <i>LEN</i> ist größer als der <i>CONF_DB</i> abzüglich des reservierten Headers (4Byte) oder die Längenangabe ist falsch.
1	8186h	Unzulässigen Parameter erkannt. ANY-Pointer <i>CONF_DB</i> verweist nicht auf einen DB.
1	8187h	Ungültiger Zustand des FBs. Daten im Header des <i>CONF_DB</i> überschrieben oder Instanz-DB nicht vorhanden.
1	8A01h	Statusanzeige im gelesenen Datensatz ist ungültig (Wert >=3)
1	8A02h	Es läuft kein Auftrag auf dem CP; der FB hat jedoch eine Quittung für gelaufenen Auftrag erwartet.
1	8A03h	Es läuft kein Auftrag auf dem CP und der CP ist nicht bereit; der FB hat einen ersten Auftrag für Datensatz-Lesen angestoßen.
1	8A04h	Es läuft kein Auftrag auf dem CP und der CP ist nicht bereit; der FB hat jedoch eine Quittung für den gelaufenen Auftrag erwartet.
1	8A05h	Auftrag läuft, eine Quittung ist jedoch noch nicht erfolgt; der FB hat einen ersten Auftrag für Datensatz-Lesen angestoßen.
1	8A06h	Auftrag ist fertig; der FB hat jedoch einen ersten Auftrag für Datensatz-Lesen angestoßen.
1	8B01h	Kommunikationsfehler, DB konnte nicht übertragen werden.
1	8B02h	Parameterfehler, Doppelter Parameterblock
1	8B03h	Parameterfehler, Subblock im Parameterblock ist nicht erlaubt.

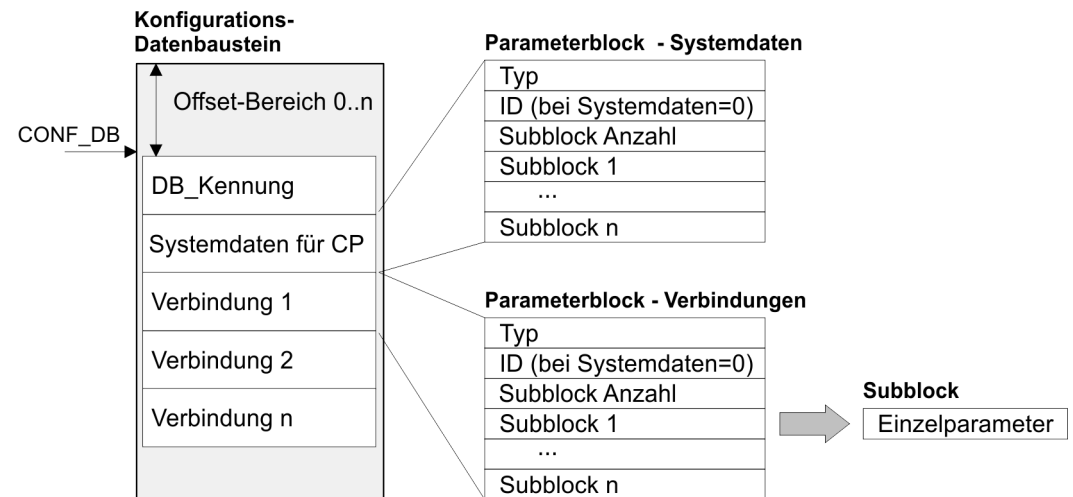
ERROR	STATUS	Beschreibung
1	8B04h	Parameterfehler, Länge die im FB angegeben wurde, stimmt nicht mit der Länge der Parameterblöcke/Subblöcke überein.
1	8B05h	Parameterfehler, Länge des Parameterblocks ist ungültig.
1	8B06h	Parameterfehler, Länge des Subblocks ist ungültig.
1	8B07h	Parameterfehler, ID des Parameterblocks ist ungültig.
1	8B08h	Parameterfehler, ID des Subblocks ist ungültig.
1	8B09h	Systemfehler, Verbindung existiert nicht.
1	8B0Ah	Datenfehler, Inhalt des Subblocks ist nicht korrekt.
1	8B0Bh	Strukturfehler, Subblock ist doppelt aufgeführt.
1	8B0Ch	Datenfehler, im Parameterblock sind nicht alle erforderlichen Parameter enthalten.
1	8B0Dh	Datenfehler, der <i>CONF_DB</i> enthält keinen Parameterblock für Systemdaten.
1	8B0Eh	Datenfehler/Strukturfehler, Typ des <i>CONF_DB</i> ist ungültig.
1	8B0Fh	Systemfehler, CP hat zu wenig Ressourcen, um den <i>CONF_DB</i> vollständig bearbeiten zu können.
1	8B10	Datenfehler, es ist nicht eingestellt, das die Konfiguration über das Anwenderprogramm erfolgt.
1	8B11	Datenfehler, der angegebene Typ des Parameterblocks ist ungültig.
1	8B12	Datenfehler, es wurden zu viele Verbindungen angegeben.
1	8B13	CP interner Fehler
1	8F22h	Bereichslängenfehler beim Lesen eines Parameters.
1	8F23h	Bereichslängenfehler beim Schreiben eines Parameters.
1	8F24h	Bereichsfehler beim Lesen eines Parameters.
1	8F25h	Bereichsfehler beim Schreiben eines Parameters.
1	8F28h	Ausrichtungsfehler beim Lesen eines Parameters.
1	8F29h	Ausrichtungsfehler beim Schreiben eines Parameters.
1	8F30h	Parameter liegt im schreibgeschützten 1. aktuellen DB
1	8F31h	Parameter liegt im schreibgeschützten 2. aktuellen DB
1	8F32h	Parameter enthält eine zu große DB-Nummer.
1	8F33h	DB-Nummernfehler
1	8F3Ah	Zielbereich wurde nicht geladen (DB).
1	8F42h	Quittungsverzug beim Lesen eines Parameters aus dem Peripheriebereich.
1	8F43h	Quittungsverzug beim Schreiben eines Parameters in den Peripheriebereich.
1	8F44h	Zugriff auf einen in der Bausteinbearbeitung zu lesenden Parameter ist gesperrt.
1	8F45h	Zugriff auf einen in der Bausteinbearbeitung zu schreibenden Parameter ist gesperrt.
1	8F7Fh	Interner Fehler

### 5.2.13.3 Konfigurations- Datenbaustein

Der Konfigurations-Datenbaustein (*CONF\_DB*) enthält sämtliche Verbindungsdaten und Konfigurationsdaten (IP-Adresse, Subnetz-Maske, Default-Router, NTP-Uhrzeit-Server und weitere) für einen Ethernet-CP. Der Konfigurations-DB wird mit dem FB 55 an den CP übergeben.

#### Aufbau

Der *CONF\_DB* kann über einen Offset-Bereich an beliebiger Stelle innerhalb eines Datenbausteines beginnen. Jede Verbindung sowie die Systemdaten werden durch einen identisch aufgebauten Parameterblock beschrieben. Einzelne Parameter werden durch Subblöcke typisiert.



#### Parameterblock für Systemdaten für CP

Nachfolgend finden Sie die für die Vernetzung des CPs relevanten Subblöcke. Diese sind im Parameterblock für Systemdaten für CP anzugeben. Je nach Anwendungsfall sind nicht alle Subblock-Typen erforderlich.

#### Aufbau

<b>Typ = 0</b>
<b>ID = 0</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
1	SUB_IP_V4	4 + 4	IP-Adresse lokale Station gemäß IPv4		zwingend
2	SUB_NETMASK	4 + 4	Subnetzmaske der lokalen Station		zwingend
4	SUB_DNS_SERV_ADDR	4 + 4	DNS Server Adresse	Kann bis zu 4 mal vorkommen. Erster Eintrag ist der primäre DNS Server.	optional
8	SUB_DEF_ROUTER	4 + 4	IP Adresse des Default Routers		optional

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
14	SUB_DHCP_ENABLE	4 + 1	IP-Adresse von einem DHCP-Server beziehen	0: kein DHCP 1: DHCP	optional
15	SUB_CLIENT_ID	Länge Client-ID + 4	-	-	optional
51	MAC-ADR	4 + 6	MAC-Adresse lokaler Knoten		optional

### Parameterblöcke für Verbindung

Sie erfahren nachfolgend, welche Werte in die Parameterblöcke einzutragen sind und welche Subblöcke zu den jeweiligen Verbindungstypen zu verwenden sind. Je nach Anwendungsfall sind nicht alle Subblock-Typen erforderlich. Von besonderer Bedeutung ist der ID-Parameter, der jedem Verbindungsparameterblock neben der Typkennung vorangestellt wird. Bei programmierten Verbindungen können Sie diese ID innerhalb des zulässigen Wertebereichs frei vergeben. Sie müssen diese ID dann an der Aufrufsstelle der FCs für die SEND/RECV-Schnittstelle zur Identifizierung der Verbindung verwenden.

Wertebereich für die Verbindungs-ID: 1, 2 ... 64

### TCP-Verbindung

<b>Typ = 1</b>
<b>ID = Verbindungs-ID</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
1	SUB_IP_V4	4 + 4	IP-Adresse lokale Station gemäß IPv4		zwingend <sup>1</sup>
9	SUB_LOC_PORT	4 + 2	Port der lokalen Station		zwingend
10	SUB_REM_PORT	4 + 2	Port der fernen Station		zwingend <sup>1</sup>
18	SUB_CONNECT_NAME	Länge Name + 4	Name der Verbindung		optional

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
19	SUB_LOC_MODE	4 + 1	Lokale Betriebsart der Verbindung, Wertebereich: 0x00 = SEND/RECV 0x10 = S5-Adressierungsmodus bei FETCH/WRITE <sup>2</sup> 0x80 = FETCH <sup>2</sup> 0x40 = WRITE <sup>2</sup> Defaulteinstellung bei Verzicht auf den Parameter ist SEND/RECV. FETCH/WRITE erfordern die Einstellung passiver Verbindungsaufbau.		optional
21	SUB_KBUS_ADR	-	-	Wert: fix 2	optional
22	SUB_CON_ESTABL	4 + 1	Typ des Verbindungsaufbaus. Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von dieser Station aus erfolgen soll. Wertebereich: 0 = passiv 1 = aktiv		zwingend

1) optional bei passiver Verbindung

2) die Codierungen können mittels ODER-Verknüpfungen kombiniert werden

## UDP-Verbindung

<b>Typ = 2</b>
<b>ID = Verbindungs-ID</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
1	SUB_IP_V4	4 + 4	IP-Adresse lokale Station gemäß IPv4		zwingend
9	SUB_LOC_PORT	4 + 2	Port der lokalen Station		zwingend
10	SUB_REM_PORT	4 + 2	Port der fernen Station		zwingend
18	SUB_CONNECT_NAME	Länge Name + 4	Name der Verbindung		optional



Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
19	SUB_LOC_MODE	4 + 1	Lokale Betriebsart der Verbindung, Wertebereich: 0x00 = SEND/RECV 0x10 = S5-Adressierungsmodus bei FETCH/ WRITE <sup>1</sup> 0x80 = FETCH <sup>1</sup> 0x40 = WRITE <sup>1</sup> Defaulteinstellung bei Verzicht auf den Parameter ist SEND/RECV. FETCH/WRITE erfordern die Einstellung passiver Verbindungsaufbau.		optional
21	SUB_KBUS_ADR	-	-	Wert: fest auf 2	optional
23	SUB_ADDR_IN_DATA_BLOCK	4 + 1	Freie UDP-Verbindung wählen. Der ferne Teilnehmer wird vom Anwenderprogramm beim AG_SEND Aufruf im Auftragsheader des Auftragspuffers eingetragen. Damit kann jeder beliebige Teilnehmer am Ethernet/LAN/WAN erreicht werden. Wertebereich: 1 = Freie UDP-Verbindung 0 = sonstige		optional

1) die Codierungen können mittels ODER-Verknüpfungen kombiniert werden.

### ISO-on-TCP-Verbindung

<b>Typ = 3</b>
<b>ID = Verbindungs-ID</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
1	SUB_IP_V4	4 + 4	IP-Adresse der fernen Station gemäß IPv4		zwingend <sup>1</sup>
11	SUB_LOC_PORT	TSAP-Länge + 4	TSAP der lokalen Station		zwingend

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
12	SUB_REM_PORT	TSAP-Länge + 4	TSAP der fernen Station		zwingend <sup>1</sup>
18	SUB_CONNECT_NAME	Länge Name + 4	Name der Verbindung		optional
19	SUB_LOC_MODE	4 + 1	Lokale Betriebsart der Verbindung, Wertebereich: 0x00 = SEND/RECV 0x10 = S5-Adressierungsmodus bei FETCH/WRITE <sup>2</sup> 0x80 = FETCH <sup>2</sup> 0x40 = WRITE <sup>2</sup> Defaulteinstellung bei Verzicht auf den Parameter ist SEND/RECV. FETCH/WRITE erfordern die Einstellung passiver Verbindungsaufbau.		optional
21	SUB_KBUS_ADR	-	-	Wert fest auf 2	optional
22	SUB_CON_ESTABL	4 + 1	Typ des Verbindungsaufbaus. Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von dieser Station aus erfolgen soll. Wertebereich: 0 = passiv 1 = aktiv		zwingend

1) optional bei passiver Verbindung

2) die Codierungen können mittels ODER-Verknüpfungen kombiniert werden

## H1-Verbindung (ISO)

<b>Typ = 10</b>
<b>ID = Verbindungs-ID</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
51	SUB_MAC	4 + 6	MAC-Adresse der fernen Station		zwingend
11	SUB_LOC_TSAP	TSAP-Länge + 4	TSAP der lokalen Station		zwingend

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
12	SUB_REM_TSAP	TSAP-Länge + 4	TSAP der fernen Station		zwingend <sup>1</sup>
18	SUB_CONNECT_NAME	Länge Name + 4	Name der Verbindung		optional
19	SUB_LOC_MODE	4 + 1	Lokale Betriebsart der Verbindung, Wertebereich: 0x00 = SEND/RECV 0x10 = S5-Adressierungsmodus bei FETCH/WRITE <sup>2</sup> 0x80 = FETCH <sup>2</sup> 0x40 = WRITE <sup>2</sup> Defaulteinstellung bei Verzicht auf den Parameter ist SEND/RECV. FETCH/WRITE erfordern die Einstellung passiver Verbindungsaufbau.		optional
22	SUB_CON_ESTABL	4 + 1	Typ des Verbindungsaufbaus. Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von dieser Station aus erfolgen soll. Wertebereich: 0 = passiv; 1 = aktiv		zwingend
52	SUB_TIME_CON_RETRAN	4 + 2	Zeitabstand nach welchem ein fehlgeschlagener Verbindungsaufbau erneut angestoßen wird. (1...60s, Default: 5s)	irrelevant bei passivem Verbindungsaufbau	optional
53	SUB_TIME_DAT_RETRAN	4 + 2	Zeitabstand nach welchem ein fehlgeschlagener Sendeversuch erneut angestoßen wird. (100...30000ms, Default: 1000ms)		optional
54		4 + 2	Anzahl der Sendeversuche inkl. 1. Versuch (1...100, Default: 5)		optional
55		4 + 2	Zeitabstand nach welchem eine Verbindung abgebaut wird, sofern von der Partner-Station keine Rückmeldung kommt. (6...160s, Default: 30s)		optional

1) optional bei passiver Verbindung

2) die Codierungen können mittels ODER-Verknüpfungen kombiniert werden

## Siemens-S7-Verbindung

<b>Typ = 11</b>
<b>ID = Verbindungs-ID</b>
Subblock Anzahl = n
Subblock 1
Subblock 2
Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
56	SUB_S/_C_DETAIL	4 + 14	Verbindungsspezifische Parameter		zwingend
18	SUB_CONNECT_NAME	Länge Name + 4	Name der Verbindung		optional
1	SUB_IP_V4	4 + 4	IP-Adresse gemäß IPv4	IP-Adresse des Partners	zwingend <sup>1</sup>
51	SUB_MAC	4 + 6	MAC-Adresse der Partnerstation		zwingend
22	SUB_CON_ESTABL	4 + 1	Typ des Verbindungsaufbaus. Legen Sie mit dieser Option fest, ob der Verbindungsaufbau von dieser Station aus erfolgen soll.  Wertebereich: 0 = passiv 1 = aktiv		zwingend

1) optional bei passiver Verbindung

## SUB\_S/\_C\_DETAIL

Parameter	Deklaration	Datentyp	Beschreibung
SubBlockID	IN	WORD	Kennung
SubBlockLen	IN	WORD	Länge
TcpIpActive	IN	INT	Verbindung über MAC- oder IP-Adresse (MAC=0, IP=1)
LocalResource	IN	WORD	Lokale Ressource 0001h ... 00DFh (1=PG, 2=OP, 0010h ... 00DFh=unspezifiziert)
LocalRack	IN	WORD	Nummer lokales Rack 0000h ... 0002h
LocalSlot	IN	WORD	Nummer lokaler Steckplatz 0002h ... 000Fh (2=CPU, 4=VIPA-PG/OP, 5=CP int., 6=CP ext.)
RemoteResource	IN	WORD	Remote Ressource 0001h ... 00DFh (1=PG, 2=OP, 0010h ... 00DFh=unspezifiziert)

Parameter	Deklaration	Datentyp	Beschreibung
RemoteRack	IN	WORD	Nummer remote Rack 0000h ... 0002h
RemoteSlot	IN	WORD	Nummer remote Steckplatz 0002h ... 000Fh (2=CPU, 4=VIPA-PG/OP, 5=CP int., 6=CP ext.)

Der "Lokale TSAP" wird gebildet aus *LocalResource*, *LocalRack* und *LocalSlot*.

Der "Remote TSAP" wird gebildet aus *RemoteResource*, *RemoteRack* und *RemoteSlot*.

### Beispiel zur Parametrierung einer Siemens S7-Verbindung

Die Parametrierung einer dynamischen Siemens S7-Verbindung mit IP\_CONF erfolgt analog zur Parametrierung einer fixen Siemens S7-Verbindung. Ausgehend von Siemens NetPro haben Sie folgende Parameter, welche folgenden Sub-Blöcken entsprechen:

Eigenschaften - Siemens S7- Verbindung	
Siemens NetPro	FB55 - IP_CONFIG
aktiver Verbindungsaufbau	SUB_CON_ESATBL.CON_ESTABL
TCP/IP	SUB_S7_C_DETAILS.TcpIpActive
IP- bzw. MAC-Adresse Partner	SUB_IP_V4.rem_IP.IP_0...IP_3 bzw. SUB_MAC.rem_MAC.MAC_0...MAC5
Lokale ID	Verbindungs-ID

Adressdetails	
Siemens NetPro	FB55 - IP_CONFIG
Lokales Rack	SUB_S7_C_DETAILS.LocalRack
Lokaler Steckplatz	SUB_S7_C_DETAILS.LocalSlot
Lokale Verbindungsressource	SUB_S7_C_DETAILS.LocalResource
Remote Rack	SUB_S7_C_DETAILS.RemoteRack
Remote Steckplatz	SUB_S7_C_DETAILS.RemoteSlot
Remote Verbindungsressource	SUB_S7_C_DETAILS.RemoteResource

### Zusätzliche Parameterblöcke

#### Block\_VIPA\_HWK

Sobald der Block\_VIPA\_HWK (Sonderkennung 99) im DB enthalten ist, bleiben alle Verbindungen, die im NETPRO projiziert wurden, erhalten. Damit ist es möglich, nur die Systemdaten (IP, Netmask usw.) mit IP\_CONFIG zu ändern. Wenn der Block\_VIPA\_HWK gefunden wurde, dürfen keine sonstigen Verbindungsdaten im DB parametrier sein, sonst wird Fehler im RetVal gemeldet. Ist die Sonderkennung Block\_VIPA\_HWK nicht im DB, so werden (wie bei Siemens) alle Verbindungen aus NETPRO entfernt und nur noch die Verbindungen aus diesem DB parametrier.

Typ = 99

ID = 0

Subblock Anzahl = 0

### Block\_VIPA\_BACNET

Sobald der Block\_VIPA\_BACNET (Sonderkennung 100) im DB enthalten ist, wird eine BACNET-Konfiguration aus dem DB abgeleitet und es werden keine weiteren Blöcke danach ausgewertet.

**Typ = 100**

Subblock Anzahl = 0

### Block\_VIPA\_IPK

**Typ = 101**

**ID = Verbindungs-ID**

Subblock Anzahl = n

Subblock 1

Subblock 2

Subblock n

Subblock				Parameter	
ID	Typ	Länge (Byte)	Bedeutung	Besonderheiten	Anwendung
1	VIPA_IPK_CYCLE	4 + 4	IPK Zykluszeit für Verbindungs-ID	VIPA-spezifisch	optional

### Beispiel-DB

Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
0.0	DB_Ident	WORD	W#16#1	W#16#1	
2.0	Systemdaten.Type	INT	0	0	Systemdaten
4.0	Systemdaten.Verblid	INT	0	0	fix 0
6.0	Systemdaten.SubBlock_Anzahl	INT	3	3	
8.0	Systemdaten.ip.SUB_IP_V4	WORD	W#16#1	W#16#1	
10.0	Systemdaten.ip.SUB_IP_V4_LEN	WORD	W#16#8	W#16#8	
12.0	Systemdaten.ip.IP_0	BYTE	B#16#0	B#16#AC	
13.0	Systemdaten.ip.IP_1	BYTE	B#16#0	B#16#14	
14.0	Systemdaten.ip.IP_2	BYTE	B#16#0	B#16#8B	
15.0	Systemdaten.ip.IP_3	BYTE	B#16#0	B#16#61	
16.0	Systemdaten.netmask.SUB_NETMASK	WORD	W#16#2	W#16#2	
18.0	Systemdaten.netmask.SUB_NETMASK_LEN	WORD	W#16#8	W#16#8	
20.0	Systemdaten.netmask.NETMASK_0	BYTE	B#16#0	B#16#FF	
21.0	Systemdaten.netmask.NETMASK_1	BYTE	B#16#0	B#16#FF	
22.0	Systemdaten.netmask.NETMASK_2	BYTE	B#16#0	B#16#FF	
23.0	Systemdaten.netmask.NETMASK_3	BYTE	B#16#0	B#16#0	
24.0	Systemdaten.router.SUB_DEF_ROUTER	WORD	W#16#8	W#16#8	
26.0	Systemdaten.router.SUB_DEF_ROUTER_LEN	WORD	W#16#8	W#16#8	
28.0	Systemdaten.router.ROUTER_0	BYTE	B#16#0	B#16#AC	
29.0	Systemdaten.router.ROUTER_1	BYTE	B#16#0	B#16#14	
30.0	Systemdaten.router.ROUTER_2	BYTE	B#16#0	B#16#8B	

Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
31.0	Systemdaten.router.ROUTER_3	BYTE	B#16#0	B#16#61	
32.0	Con_TCP_ID1.Typ	INT	1	1	TCP-Verbindung
34.0	Con_TCP_ID1.Verblid	INT	0	1	Verbindungs-ID
36.0	Con_TCP_ID1.SubBlock_Anzahl	INT	4	4	
38.0	Con_TCP_ID1.ip1.SUB_IP_V4	WORD	W#16#1	W#16#1	
40.0	Con_TCP_ID1.ip1.SUB_IP_V4_LEN	WORD	W#16#8	W#16#8	
42.0	Con_TCP_ID1.ip1.IP_0	BYTE	B#16#0	B#16#AC	
43.0	Con_TCP_ID1.ip1.IP_1	BYTE	B#16#0	B#16#14	
44.0	Con_TCP_ID1.ip1.IP_2	BYTE	B#16#0	B#16#8B	
45.0	Con_TCP_ID1.ip1.IP_3	BYTE	B#16#0	B#16#62	
46.0	Con_TCP_ID1.locport.SUB_LOC_PORT	WORD	W#16#9	W#16#9	
48.0	Con_TCP_ID1.locport.SUB_LOC_PORT_LEN	WORD	W#16#6	W#16#6	
50.0	Con_TCP_ID1.locport.LOC_PORT	WORD	W#16#0	W#16#3E9	
52.0	Con_TCP_ID1.remport.SUB_REM_PORT	WORD	W#16#A	W#16#A	
54.0	Con_TCP_ID1.remport.SUB_REM_PORT_LEN	WORD	W#16#6	W#16#6	
56.0	Con_TCP_ID1.remport.REM_PORT	WORD	W#16#0	W#16#3E9	
58.0	Con_TCP_ID1.con_est.SUB_CON_ESTABL	WORD	W#16#16	W#16#16	
60.0	Con_TCP_ID1.con_est.SUB_CON_ESTABL_LEN	WORD	W#16#6	W#16#6	
62.0	Con_TCP_ID1.con_est.CON_ESTABL	BYTE	B#16#0	B#16#1	
64.0	Con_ISO_ID3.Typ	INT	3	3	ISO-on-TCP-Verbindung
66.0	Con_ISO_ID3.Verblid	INT	0	3	Verbindungs-ID
68.0	Con_ISO_ID3.SubBlock_Anzahl	INT	4	4	
70.0	Con_ISO_ID3.ip1.SUB_IP_V4	WORD	W#16#1	W#16#1	
72.0	Con_ISO_ID3.ip1.SUB_IP_V4_LEN	WORD	W#16#8	W#16#8	
74.0	Con_ISO_ID3.ip1.IP_0	BYTE	B#16#0	B#16#AC	
75.0	Con_ISO_ID3.ip1.IP_1	BYTE	B#16#0	B#16#10	
76.0	Con_ISO_ID3.ip1.IP_2	BYTE	B#16#0	B#16#8B	
77.0	Con_ISO_ID3.ip1.IP_3	BYTE	B#16#0	B#16#62	
78.0	Con_ISO_ID3.loc_TSAP.SUB_LOC_PORT	WORD	W#16#B	W#16#B	
80.0	Con_ISO_ID3.loc_TSAP.SUB_LOC_PORT_LEN	WORD	W#16#A	W#16#A	
82.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[0]	BYTE	B#16#0	B#16#54	
83.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[1]	BYTE	B#16#0	B#16#53	
84.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[2]	BYTE	B#16#0	B#16#41	
85.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[3]	BYTE	B#16#0	B#16#50	
86.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[4]	BYTE	B#16#0	B#16#30	
87.0	Con_ISO_ID3.loc_TSAP.LOC_TSAP[5]	BYTE	B#16#0	B#16#31	
88.0	Con_ISO_ID3.rem_TSAP.SUB_REM_PORT	WORD	W#16#C	W#16#C	
90.0	Con_ISO_ID3.rem_TSAP.SUB_REM_PORT_LEN	WORD	W#16#A	W#16#A	
92.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[0]	BYTE	B#16#0	B#16#54	
93.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[1]	BYTE	B#16#0	B#16#53	

Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
94.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[2]	BYTE	B#16#0	B#16#41	
95.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[3]	BYTE	B#16#0	B#16#50	
96.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[4]	BYTE	B#16#0	B#16#30	
97.0	Con_ISO_ID3.rem_TSAP.REM_TSAP[5]	BYTE	B#16#0	B#16#31	
98.0	Con_ISO_ID3.con_est.SUB_CON_ESTABL	WORD	W#16#16	W#16#16	
100.0	Con_ISO_ID3.con_est.SUB_CON_ESTABL_LEN SUB_CON_ESTABL SUB_CON_ESTABL_LEN	WORD	W#16#6	W#16#6	
102.0	Con_ISO_ID3.con_est.CON_ESTABL	BYTE	B#16#0	B#16#1	
104.0	S7_Verb.Typ	INT	11	11	S7-Verbindung
106.0	S7_Verb.Verb_ID	INT	0	0	Verb.-ID
108.0	S7_Verb.SubBlock_Anzahl	INT	5	5	
110.0	S7_Verb.Verb_Parameter.SUB_S7_C_DETAIL	INT	56	56	
112.0	S7_Verb.Verb_Parameter.SUB_S7_C_DETAIL_LEN	INT	18	18	
114.0	S7_Verb.Verb_Parameter.TcpIpActive	INT	0	1	
116.0	S7_Verb.Verb_Parameter.LocalResource	INT	0	2	
118.0	S7_Verb.Verb_Parameter.LocalRack	INT	0	0	
120.0	S7_Verb.Verb_Parameter.LocalsSlot	INT	0	2	
122.0	S7_Verb.Verb_Parameter.RemoteResource	INT	0	2	
124.0	S7_Verb.Verb_Parameter.RemoteRack	INT	0	0	
126.0	S7_Verb.Verb_Parameter.RemoteSlot	INT	0	2	
128.0	S7_Verb.ipl.SUB_IP_V4	WORD	W#16#1	W#16#1	
130.0	S7_Verb.ipl.SUB_IP_V4_LEN	WORD	W#16#8	W#16#8	
132.0	S7_Verb.ipl.IP_0	BYTE	B#16#0	B#16#AC	
133.0	S7_Verb.ipl.IP_1	BYTE	B#16#0	B#16#10	
134.0	S7_Verb.ipl.IP_2	BYTE	B#16#0	B#16#8B	
135.0	S7_Verb.ipl.IP_3	BYTE	B#16#0	B#16#62	
136.0	S7_Verb.Mac.SUB_MAC	INT	51	51	
138.0	S7_Verb.Mac.SUB_MAC_LEN	INT	10	10	
140.0	S7_Verb.Mac.MAC_0	BYTE	B#16#0	B#16#0	
141.0	S7_Verb.Mac.MAC_1	BYTE	B#16#0	B#16#20	
142.0	S7_Verb.Mac.MAC_2	BYTE	B#16#0	B#16#D5	
143.0	S7_Verb.Mac.MAC_3	BYTE	B#16#0	B#16#77	
144.0	S7_Verb.Mac.MAC_4	BYTE	B#16#0	B#16#53	
145.0	S7_Verb.Mac.MAC_5	BYTE	B#16#0	B#16#9B	
146.0	S7_Verb.con_est.SUB_CON_ESTABL	WORD	W#16#16	W#16#16	
148.0	S7_Verb.con_est.SUB_CON_ESTABL_LEN	WORD	W#16#6	W#16#6	
150.0	S7_Verb.con_est.CON_ESTABL	BYTE	B#16#0	B#16#1	
152.0	S7_Verb.name_verb.SUB_CONNECT_NAME	WORD	W#16#12	W#16#12	
154.0	S7_Verb.name_verb.SUB_CONNECT_NAME_LEN	WORD	W#16#23	W#16#23	
156.0	S7_Verb.name_verb.CONNECT_NAME[0]	CHAR	''	'V'	S7-Verbindung mit IP-Konfig. 1
157.0	S7_Verb.name_verb.CONNECT_NAME[1]	CHAR	''	'e'	



Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
158.0	S7_Verb.name_verb.CONNECT_NAME[2]	CHAR	' '	'r'	
159.0	S7_Verb.name_verb.CONNECT_NAME[3]	CHAR	' '	'b'	
160.0	S7_Verb.name_verb.CONNECT_NAME[4]	CHAR	' '	'i'	
161.0	S7_Verb.name_verb.CONNECT_NAME[5]	CHAR	' '	'n'	
162.0	S7_Verb.name_verb.CONNECT_NAME[6]	CHAR	' '	'd'	
163.0	S7_Verb.name_verb.CONNECT_NAME[7]	CHAR	' '	'u'	
164.0	S7_Verb.name_verb.CONNECT_NAME[8]	CHAR	' '	'n'	
165.0	S7_Verb.name_verb.CONNECT_NAME[9]	CHAR	' '	'g'	
166.0	S7_Verb.name_verb.CONNECT_NAME[10]	CHAR	' '	' '	
167.0	S7_Verb.name_verb.CONNECT_NAME[11]	CHAR	' '	'S'	
168.0	S7_Verb.name_verb.CONNECT_NAME[12]	CHAR	' '	'7'	
169.0	S7_Verb.name_verb.CONNECT_NAME[13]	CHAR	' '	' '	
170.0	S7_Verb.name_verb.CONNECT_NAME[14]	CHAR	' '	'm'	
171.0	S7_Verb.name_verb.CONNECT_NAME[15]	CHAR	' '	'i'	
172.0	S7_Verb.name_verb.CONNECT_NAME[16]	CHAR	' '	't'	
173.0	S7_Verb.name_verb.CONNECT_NAME[17]	CHAR	' '	' '	
174.0	S7_Verb.name_verb.CONNECT_NAME[18]	CHAR	' '	'l'	
175.0	S7_Verb.name_verb.CONNECT_NAME[19]	CHAR	' '	'P'	
176.0	S7_Verb.name_verb.CONNECT_NAME[20]	CHAR	' '	'.'	
177.0	S7_Verb.name_verb.CONNECT_NAME[21]	CHAR	' '	'C'	
178.0	S7_Verb.name_verb.CONNECT_NAME[22]	CHAR	' '	'o'	
179.0	S7_Verb.name_verb.CONNECT_NAME[23]	CHAR	' '	'n'	
180.0	S7_Verb.name_verb.CONNECT_NAME[24]	CHAR	' '	'f'	
181.0	S7_Verb.name_verb.CONNECT_NAME[25]	CHAR	' '	'i'	
182.0	S7_Verb.name_verb.CONNECT_NAME[26]	CHAR	' '	'g'	
183.0	S7_Verb.name_verb.CONNECT_NAME[27]	CHAR	' '	' '	
184.0	S7_Verb.name_verb.CONNECT_NAME[28]	CHAR	' '	'1'	
185.0	S7_Verb.name_verb.CONNECT_NAME[29]	CHAR	' '	' '	
186.0	S7_Verb.name_verb.CONNECT_NAME[30]	CHAR	' '	' '	