

SIEMENS

Inhaltsverzeichnis

Offene Kommunikation über
Industrial Ethernet

1

SIMATIC

Index

Offene Kommunikation über Industrial Ethernet

Handbuch

Ausgabe 12/2005
A5E00711636-01

Sicherheitshinweise

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.



Gefahr

bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Warnung

bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Vorsicht

mit Warndreieck bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Vorsicht

ohne Warndreieck bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Achtung

bedeutet, dass ein unerwünschtes Ergebnis oder Zustand eintreten kann, wenn der entsprechende Hinweis nicht beachtet wird.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zugehörige Gerät/System darf nur in Verbindung mit dieser Dokumentation eingerichtet und betrieben werden. Inbetriebsetzung und Betrieb eines Gerätes/Systems dürfen nur von **qualifiziertem Personal** vorgenommen werden. Qualifiziertes Personal im Sinne der sicherheitstechnischen Hinweise dieser Dokumentation sind Personen, die die Berechtigung haben, Geräte, Systeme und Stromkreise gemäß den Standards der Sicherheitstechnik in Betrieb zu nehmen, zu erden und zu kennzeichnen.

Bestimmungsgemäßer Gebrauch

Beachten Sie Folgendes:



Warnung

Das Gerät darf nur für die im Katalog und in der technischen Beschreibung vorgesehenen Einsatzfälle und nur in Verbindung mit von Siemens empfohlenen bzw. zugelassenen Fremdgeräten und -komponenten verwendet werden. Der einwandfreie und sichere Betrieb des Produktes setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung und Montage sowie sorgfältige Bedienung und Instandhaltung voraus.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Inhaltsverzeichnis

1	Offene Kommunikation über Industrial Ethernet	1-1
1.1	Übersicht.....	1-1
1.2	Arbeitsweise der FBs zur Offenen Kommunikation über Industrial Ethernet ...	1-2
1.3	Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP	1-5
1.4	Parametrierung des lokalen Kommunikationszugangspunkts bei UDP	1-9
1.5	Struktur der Adressinformation des remoten Partners bei UDP	1-11
1.6	Beispiele zur Parametrierung der Kommunikationsverbindungen	1-12
1.7	Verbindung aufbauen mit dem FB 65 "TCON"	1-21
1.8	Verbindung abbauen mit dem FB 66 "TDISCON"	1-24
1.9	Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"	1-26
1.10	Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"	1-29
1.11	Senden von Daten über UDP mit dem FB 67 "TUSEND"	1-34
1.12	Empfangen von Daten über UDP mit dem FB 68 "TURCV"	1-37
Index		Index-1

1 Offene Kommunikation über Industrial Ethernet

1.1 Übersicht

Um mit anderen Ethernet-fähigen Kommunikationspartnern per Anwenderprogramm Daten austauschen zu können, stellt Ihnen STEP 7 in der Bibliothek "Standard Library" unter "Communication Blocks" die folgenden FBs und UDTs zur Verfügung:

- Verbindungsorientierte Protokolle: TCP native gemäß RFC 793, ISO on TCP gemäß RFC 1006:
 - UDT 65 "TCON_PAR" mit der Datenstruktur zur Verbindungsparametrierung
 - FB 65 "TCON" zum Verbindungsaufbau
 - FB 66 "TDISCON" zum Verbindungsabbau
 - FB 63 "TSEND" zum Senden von Daten
 - FB 64 "TRCV" zum Empfangen von Daten
- Verbindungsloses Protokoll: UDP gemäß RFC 768
 - UDT 65 "TCON_PAR" mit der Datenstruktur zur Parametrierung des lokalen Kommunikationszugangspunkts
 - UDT 66 "TCON_ADR" mit der Datenstruktur der Adressierungsparameter des remoten Partners
 - FB 65 "TCON" zur Einrichtung des lokalen Kommunikationszugangspunkts
 - FB 66 "TDISCON" zur Auflösung des lokalen Kommunikationszugangspunkts
 - FB 67 "TUSEND" zum Senden von Daten
 - FB 68 "TURCV" zum Empfangen von Daten

1.2 Arbeitsweise der FBs zur Offenen Kommunikation über Industrial Ethernet

Verbindungsorientierte und verbindungslose Protokolle

Man unterscheidet in der Datenkommunikation zwischen folgenden beiden Protokollarten:

- Verbindungsorientierte Protokolle:
Diese 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. 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 B#16#01 und B#16#11)
- ISO on TCP gemäß RFC 1006 (Verbindungstyp B#16#12)

- Verbindungslose Protokolle:
Diese arbeiten ohne Verbindung. Der Verbindungsauf- und der Verbindungsabbau zum remoten Partner entfallen also. 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 B#16#13)

Die Arbeitsweise der Funktionsbausteine ist abhängig von der benutzten Protokollvariante. Darauf wird im Folgenden näher eingegangen.

TCP native

Bei der Datenübertragung werden weder Informationen zur Länge noch über Anfang und Ende einer Nachricht übertragen. Beim Senden ist dies unproblematisch, da der Sender weiß, wie viele Datenbytes er verschicken will. Der Empfänger jedoch hat keine Möglichkeit zu erkennen, wo eine Nachricht im Datenstrom endet und wo die nächste beginnt. Es wird daher empfohlen, dem Parameter LEN des FB 64 "TRCV" (Anzahl der Bytes, die empfangen werden sollen) denselben Wert zuzuweisen wie dem Parameter LEN des FB 63 "TSEND" beim Kommunikationspartner (Anzahl der Bytes, die gesendet werden sollen).

Falls Sie die Länge der zu empfangenden Daten (Parameter LEN des FB 64 "TRCV") größer gewählt haben als die Länge der gesendeten Daten, kopiert der FB 64 "TRCV" die empfangenen Daten erst dann in den von Ihnen vorgegebenen Empfangsbereich (Parameter DATA), wenn die parametrierte Länge erreicht ist. Das geschieht erst dann, wenn Daten eines nachfolgenden Auftrags empfangen wurden. Bitte beachten Sie, dass sich in diesem Fall Daten aus zwei unterschiedlichen Sendeaufträgen in ein und demselben Empfangsbereich befinden. Wenn Sie die exakte Länge der ersten Nachricht nicht kennen, haben Sie keine Möglichkeit, das Ende der ersten bzw. den Anfang der zweiten Nachricht zu erkennen.

Falls Sie die Länge der zu empfangenden Daten (Parameter DATA des FB 64 "TRCV") kleiner gewählt haben als die Länge der gesendeten Daten, kopiert der FB 64 so viele Bytes in den Empfangsdatenbereich, wie Sie am Parameter LEN vorgegeben 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.

Falls Sie die Länge der zu empfangenden Daten (Parameter LEN des FB 64 "TRCV") größer gewählt haben als die Länge der gesendeten Daten, kopiert der FB 64 "TRCV" 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 (Parameter DATA des FB 64 "TRCV") kleiner gewählt haben als die Länge der gesendeten Daten, kopiert der FB 64 keine Daten in den Empfangsdatenbereich, sondern liefert folgende Fehlerinformation: ERROR=1, STATUS=W#16#8088.

UDP

Sie führen keinen Verbindungsaufbau wie bei den Protokollen TCP native und ISO on TCP durch. Daher müssen Sie beim Aufruf des Sendebausteins FB 67 "TUSEND" einen Verweis auf die Adressparameter des Empfängers (IP-Adresse und Port-Nr.) angeben. Analog erhalten Sie nach Abschluss des Empfangsbausteins FB 68 "TURCV" einen Verweis auf die Adressparameter des Senders (IP-Adresse und Port-Nr.).

Damit Sie die FBs 67 "TUSEND" und 68 "TURCV" nutzen können, müssen Sie zuvor sowohl auf der Sender- als auch auf der Empfängerseite den FB 65 "TCON" aufrufen, um den lokalen Kommunikationszugangspunkt einzurichten.

Bei jedem Aufruf des FB 67 "TUSEND" können Sie den remoten Partner durch Angabe seiner IP-Adresse und seiner Port-Nr. neu referenzieren.

Bei der Datenübertragung werden Informationen zur Länge und zum Ende einer Nachricht übertragen.

Falls Sie die Länge der zu empfangenden Daten (Parameter LEN des FB 68 "TURCV") größer gewählt haben als die Länge der gesendeten Daten, kopiert der FB 68 "TURCV" 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 (Parameter LEN des FB 68 "TURCV") kleiner gewählt haben als die Länge der gesendeten Daten, kopiert der FB 68 keine Daten in den Empfangsdatenbereich, sondern liefert folgende Fehlerinformation: ERROR = 1, STATUS = W#16#8088.

1.3 Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP

Datenbaustein für Verbindungsparametrierung

Um die Kommunikationsverbindungen bei TCP native und ISO on TCP zu parametrieren, legen Sie einen DB an, der die Datenstruktur aus dem UDT 65 "TCON_PAR" enthält. Diese Datenstruktur enthält die notwendigen Parameter, die Sie zum Aufbau der Verbindung benötigen. Für jede Verbindung benötigen Sie solch eine Datenstruktur, 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#DB100.DBX0.0 Byte 64).

Aufbau der Verbindungsbeschreibung (UDT 65)

Byte	Parameter	Daten- typ	Anfangs- wert	Beschreibung
0 bis 1	block_length	WORD	W#16#40	Länge des UDT 65: 64 Bytes (fest)
2 bis 3	id	WORD	W#16#0000	Referenz auf diese Verbindung (Wertebereich: W#16#0001 bis W#16#0FFF) Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei ID angeben.
4	connection_type	BYTE	B#16#01	Verbindungstyp: <ul style="list-style-type: none"> B#16#11: TCP/IP native B#16#12: ISO on TCP B#16#01: TCP/IP native (Kompatibilitätsmode)
5	active_est	BOOL	FALSE	Kennung für die Art des Verbindungsaufbaus: <ul style="list-style-type: none"> FALSE: passiver Verbindungsaufbau TRUE: aktiver Verbindungsaufbau
6	local_device_id	BYTE	B#16#02	<ul style="list-style-type: none"> B#16#00: Kommunikation über CP B#16#02: Kommunikation über die integrierte IE-Schnittstelle bei den CPUs 315-2 PN/DP und 317-2 PN/DP B#16#03: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 319-3 PN/DP Nr. der parametrisierten IE-Schnittstelle bei WinAC RTX 2005 (mögliche Werte: B#16#01 bis B#16#04)
7	local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id; mögliche Werte: <ul style="list-style-type: none"> 0 oder 2, falls Verbindungstyp = B#16#01 oder B#16#11 Für die aktive Seite ist nur der Wert B#16#00 zulässig. 2 bis 16, falls Verbindungstyp = B#16#12

Byte	Parameter	Daten-typ	Anfangs-wert	Beschreibung
8	rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
9	rem_staddr_len	BYTE	B#16#00	Länge der Adresse des remoten Verbindungsendpunkts: <ul style="list-style-type: none"> • 0: un spezifiziert, d.h. Parameter rem_staddr ist irrelevant. • 4: gültige IP-Adresse im Parameter rem_staddr
10	rem_tsap_id_len	BYTE	B#16#00	verwendete Länge des Parameters rem_tsap_id; mögliche Werte: <ul style="list-style-type: none"> • 0 oder 2, falls Verbindungstyp = B#16#01 oder B#16#11 Für die passive Seite ist nur der Wert B#16#00 zulässig. • 2 bis 16, falls Verbindungstyp = B#16#12
11	next_staddr_len	BYTE	B#16#00	verwendete Länge des Parameters next_staddr
12 bis 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	bei connection_type = <ul style="list-style-type: none"> • B#16#11: lokale Port-Nr. (mögliche Werte: 2000 bis 5000), local_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = irrelevant • B#16#12: lokale TSAP-ID: local_tsap_id[1] = B#16#E0 (Verbindungstyp T-Verbindung), local_tsap_id[2] = Rack und Steckplatz der eigenen CPU (Bits 0 bis 4 Steckplatz, Bits 5 bis 7: Racknummer), local_tsap_id[3-16] = TSAP-Erweiterung • B#16#01: lokale Port-Nr. (mögliche Werte: 2000 bis 5000), local_tsap_id[1] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = irrelevant <p>Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id, den Sie auf Ihrer CPU verwenden, eindeutig ist.</p>
28 bis 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird derzeit nicht verwendet. Sie müssen ihn mit 0 belegen.

Byte	Parameter	Daten- typ	Anfangs- wert	Beschreibung
34 bis 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	IP-Adresse des remoten Verbindungsendpunkts, z. B. 192.168.002.003: bei connection_type = <ul style="list-style-type: none"> B#16#1x: rem_staddr[1] = B#16#C0 (192), rem_staddr[2] = B#16#A8 (168), rem_staddr[3] = B#16#02 (002), rem_staddr[4] = B#16#03 (003), rem_staddr[5-6]= irrelevant B#16#01: rem_staddr[1] = B#16#03 (003), rem_staddr[2] = B#16#02 (002), rem_staddr[3] = B#16#A8 (168), rem_staddr[4] = B#16#C0 (192), rem_staddr[5-6]= irrelevant
40 bis 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	bei connection_type = <ul style="list-style-type: none"> B#16#11: remote Port-Nr. (mögliche Werte: 2000 bis 5000), rem_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, rem_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, rem_tsap_id[3-16] = irrelevant B#16#12: remote TSAP-ID: rem_tsap_id[1] = B#16#E0 (Verbindungstyp T- Verbindung), rem_tsap_id[2] = Rack und Steckplatz des remoten Verbindungsendpunkts (CPU) (Bits 0 bis 4: Steckplatz, Bits 5 bis 7: Racknummer), rem_tsap_id[3-16] = TSAP-Erweiterung B#16#01: remote Port-Nr. (mögliche Werte: 2000 bis 5000), local_tsap_id[1] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = irrelevant
56 bis 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	bei local_device_id = <ul style="list-style-type: none"> B#16#00: next_staddr[1]: Rack und Steckplatz des zugehörigen (lokalen) CP (Bits 0 bis 4: Steckplatz, Bits 5 bis 7: Racknummer) next_staddr[2-6]: irrelevant B#16#02: next_staddr[1-6]: irrelevant
62 bis 63	spare	WORD	W#16#0000	irrelevant

CPU-Abhängigkeit der Verbindungstypen

Im Folgenden wird angegeben, welchen Verbindungstyp Sie auf welcher CPU einsetzen können:

- connection_type=B#16#11 (TCP native): CPUs 31x-2 PN/DP ab Firmwarestand V2.4, WinAC RTX ab V4.2 (WinAC RTX 2005)
- connection_type=B#16#12 (ISO on TCP): CPUs 31x-2 PN/DP ab Firmwarestand V2.4, S7-400-CPU (ohne CPU 414-4H und CPU 417-4H) ab Firmwarestand V4.1
- connection_type=B#16#01 (TCP native, Kompatibilitätsmode): alle CPUs 31x-2 PN/DP, WinAC RTX ab V4.2 (WinAC RTX 2005)

Die Anzahl der möglichen Verbindungen entnehmen Sie den Technischen Daten Ihrer CPU.

Verbindungsaufbau

Ein Kommunikationspartner A muss den aktiven Verbindungsaufbau anstoßen. Ein Kommunikationspartner B muss den passiven Verbindungsaufbau anstoßen. Wenn beide Kommunikationspartner ihren Verbindungsaufbau angestoßen haben, kann das Betriebssystem die Kommunikationsverbindung vollständig aufbauen.

In der Verbindungsparametrierung legen Sie fest, welcher Kommunikationspartner den Verbindungsaufbau aktiviert und welcher auf eine Anforderung des Kommunikationspartners hin einen passiven Verbindungsaufbau durchführt.

Siehe auch:

Beispiele zur Parametrierung der Kommunikationsverbindungen

1.4 Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Datenbaustein für Parametrierung des lokalen Kommunikationszugangspunkts

Um den lokalen Kommunikationszugangspunkt zu parametrieren, legen Sie einen DB an, der die Datenstruktur aus dem UDT 65 "TCON_PAR" enthält. Diese Datenstruktur enthält die notwendigen Parameter, die Sie zum Einrichten der Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems benötigen.

Der Parameter CONNECT des FB 65 "TCON" enthält einen Verweis auf die Adresse der zugehörigen Verbindungsbeschreibung (z. B. P#DB100.DBX0.0 Byte 64).

Aufbau der Verbindungsbeschreibung (UDT 65)

Byte	Parameter	Daten- typ	Anfangs- wert	Beschreibung
0 bis 1	block_length	WORD	W#16#40	Länge des UDT 65: 64 Bytes (fest)
2 bis 3	id	WORD	W#16#0000	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems (Wertebereich: W#16#0001 bis W#16#0FFF) Den Wert dieses Parameters müssen Sie im jeweiligen Baustein bei ID angeben.
4	connection_type	BYTE	B#16#01	Verbindungstyp: <ul style="list-style-type: none"> B#16#13: UDP
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	B#16#02	<ul style="list-style-type: none"> B#16#02: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 317-2 PN/DP B#16#03: Kommunikation über die integrierte IE-Schnittstelle bei der CPU 319-3 PN/DP
7	local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id; möglicher Wert: 2
8	rem_subnet_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
9	rem_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
10	rem_tsap_id_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.
11	next_staddr_len	BYTE	B#16#00	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit B#16#00 belegen.

Byte	Parameter	Daten- typ	Anfangs- wert	Beschreibung
12 bis 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	lokale Port-Nr. (mögliche Werte: 2000 bis 5000), local_tsap_id[1] = high byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[2] = low byte der Port-Nr. in hexadezimaler Darstellung, local_tsap_id[3-16] = irrelevant Hinweis: Stellen Sie sicher, dass jeder Wert von local_tsap_id, den Sie auf Ihrer CPU verwenden, eindeutig ist.
28 bis 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 0 belegen.
34 bis 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 0 belegen.
40 bis 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 0 belegen.
56 bis 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Dieser Parameter wird nicht verwendet. Sie müssen ihn mit 0 belegen.
62 bis 63	spare	WORD	W#16#0000	irrelevant

CPU-Abhängigkeit des Verbindungstyps UDP

Den Verbindungstyp UDP (connection_type=B#16#13) gibt es bei den CPUs 31x-2 PN/DP ab Firmwarestand V2.4.

Die Anzahl der möglichen Verbindungen zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems entnehmen Sie den Technischen Daten Ihrer CPU.

Einrichten des lokalen Kommunikationszugangspunkts

Jeder Kommunikationspartner muss unabhängig vom anderen seinen lokalen Kommunikationszugangspunkt einrichten, d. h. die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems aufbauen.

Siehe auch:

Beispiele zur Parametrierung der Kommunikationsverbindungen

1.5 Struktur der Adressinformation des remoten Partners bei UDP

Übersicht

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 für die Adressinformation des remoten Partners

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#DB100.DBX0.0 Byte 8).

Aufbau der Adressinformation des remoten Partners (UDT 66)

Byte	Parameter	Datentyp	Anfangswert	Beschreibung
0 bis 3	rem_ip_addr	ARRAY [1..4] of BYTE	B#16#00 ...	IP-Adresse des remoten Partners, z. B. 192.168.002.003: <ul style="list-style-type: none"> • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#02 (002) • rem_ip_addr[4] = B#16#03 (003)
4 bis 5	rem_port_nr	ARRAY [1..2] of BYTE	B#16#00 ...	remote Port-Nr. (mögliche Werte: 2000 bis 5000) <ul style="list-style-type: none"> • rem_port_nr[1] = high byte der Port-Nr. in hexadezimaler Darstellung • rem_port_nr[2] = low byte der Port-Nr. in hexadezimaler Darstellung
6 bis 7	spare	ARRAY [1..2] of BYTE	B#16#00 ...	irrelevant

Siehe auch:

Beispiele zur Parametrierung der Kommunikationsverbindungen

1.6 Beispiele zur Parametrierung der Kommunikationsverbindungen

Beispiel 1: Zwei S7-400-CPUs über CP 443-1 Adv.

Die beiden Kommunikationspartner sind zwei CPUs 414-2 mit Firmwarestand V4.1.0. Die Kommunikation wird über zwei CPs 443-1 Adv. mit Firmwarestand V2.2 abgewickelt.

Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 414-2 (FW V4.1.0) mit CP 443-1 Adv. (FW V2.2)	Kommunikationspartner B: CPU 414-2 (FW V4.1.0) mit CP 443-1 Adv. (FW V2.2)
Verbindungsaufbau	aktiv	passiv
IP-Adresse	192.168.4.14	192.168.4.16
Physikalische Adresse der CPU	Rack 0, Steckplatz 3	Rack 0, Steckplatz 4
Physikalische Adresse des zugehörigen CP	Rack 0, Steckplatz 6	Rack 1, Steckplatz 8
Lokale TSAP-ID (Hinweis: Die Codierung des eigentlichen TSAP zur Unterscheidung der Verbindung erfolgt ab dem 3. Byte.)	0xE0 03 54 43 50 2D 31	0xE0 04 54 43 50 2D 31

Die für den aktiven Verbindungsaufbau von Kommunikationspartner A relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Datentyp	Wert im Beispiel	Beschreibung
id	WORD	W#16#0414	Referenz auf diese Verbindung
connection_type	BYTE	B#16#12	Verbindungstyp: ISO on TCP
active_est	BOOL	TRUE	Aktiver Verbindungsaufbau
local_device_id	BYTE	B#16#00	Kommunikation AS-intern über CP
local_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters local_tsap_id
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: <ul style="list-style-type: none"> • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#01	verwendete Länge des Parameters next_staddr

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
local_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • local_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • local_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • local_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • local_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • local_tsap_id[8-16] = irrelevant 	lokale TSAP-ID: 0xE0035443502D31
rem_staddr	ARRAY [1..6] of BYTE	<p>"192.168.4.16"</p> <ul style="list-style-type: none"> • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#10 (16) • rem_staddr[5-6] = irrelevant 	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#04 • rem_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • rem_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • rem_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • rem_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • rem_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • rem_tsap_id[8-16] = irrelevant 	remote TSAP-ID: 0xE0045443502D31
next_staddr	ARRAY [1..6] of BYTE	<ul style="list-style-type: none"> • next_staddr[1] = B#16#06 • next_staddr[2-6] = irrelevant 	Rack = 0, Steckplatz = 6 (Bits 7 bis 5: Rack-Nr., Bits 4 bis 0: Steckplatz-Nr.)

Die für den passiven Verbindungsaufbau von Kommunikationspartner B relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Daten-typ	Wert im Beispiel	Beschreibung
id	WORD	W#16#0416	Referenz auf diese Verbindung
connection_type	BYTE	B#16#12	Verbindungstyp: ISO on TCP
active_est	BOOL	FALSE	Passiver Verbindungsaufbau
local_device_id	BYTE	B#16#00	Kommunikation AS-intern über CP
local_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters local_tsap_id
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: <ul style="list-style-type: none"> • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#07	verwendete Länge des Parameters rem_tsap_id
next_staddr_len	BYTE	B#16#01	verwendete Länge des Parameters next_staddr
local_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#04 • local_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • local_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • local_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • local_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • local_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • local_tsap_id[8-16] = irrelevant 	lokale TSAP-ID: 0xE0045443502D31
rem_staddr	ARRAY [1..6] of BYTE	"192.168.4.14" <ul style="list-style-type: none"> • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#0E (14) • rem_staddr[5-6] = irrelevant 	IP-Adresse des remoten Verbindungsendpunkts

Parameter	Daten-typ	Wert im Beispiel	Beschreibung
rem_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#54 (ASCII-Äquivalent von "T") • rem_tsap_id[4] = B#16#43 (ASCII-Äquivalent von "C") • rem_tsap_id[5] = B#16#50 (ASCII-Äquivalent von "P") • rem_tsap_id[6] = B#16#2D (ASCII-Äquivalent von "-") • rem_tsap_id[7] = B#16#31 (ASCII-Äquivalent von "1") • rem_tsap_id[8-16] = irrelevant 	remote TSAP-ID: 0xE0035443502D31
next_staddr	ARRAY [1..6] of BYTE	<ul style="list-style-type: none"> • next_staddr[1] = B#16#28 • next_staddr[2-6] = irrelevant 	Rack = 1, Steckplatz = 8 (Bits 7 bis 5: Rack-Nr., Bits 4 bis 0: Steckplatz-Nr.)

Beispiel 2: Zwei S7-300-CPUs mit integrierter PROFINET-Schnittstelle

Die beiden Kommunikationspartner sind zwei CPUs 319-3 PN/DP mit Firmwarestand V2.4.0. Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 319-3 PN/DP (FW V2.4.0)	Kommunikationspartner B: CPU 319-3 PN/DP (FW V2.4.0)
Verbindungsaufbau	aktiv	passiv
IP-Adresse	192.168.3.142	192.168.3.125
Lokale Port-Nr.	irrelevant	2005

Die für den aktiven Verbindungsaufbau von Kommunikationspartner A relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
id	WORD	W#16#0014	Referenz auf diese Verbindung
connection_type	BYTE	B#16#11	Verbindungstyp: TCP/IP native
active_est	BOOL	TRUE	Aktiver Verbindungsaufbau
local_device_id	BYTE	B#16#02	Kommunikation über die integrierte Ethernet-Schnittstelle
local_tsap_id_len	BYTE	B#16#00 (nur dieser Wert ist möglich)	Parameter local_tsap_id wird nicht verwendet
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: <ul style="list-style-type: none"> • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#02 (nur dieser Wert ist möglich)	verwendete Länge des Parameters rem_tsap_id
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.125" <ul style="list-style-type: none"> • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#7D (125) • rem_staddr[5-6] = irrelevant 	IP-Adresse des remoten Verbindungsendpunkts
rem_tsap_id	ARRAY [1..16] of BYTE	"2005" <ul style="list-style-type: none"> • rem_tsap_id[1] = B#16#07 • rem_tsap_id[2] = B#16#D5 • rem_tsap_id[3-16] = irrelevant 	Remote Port-Nr.: 2005 = W#16#07D5

Die für den passiven Verbindungsaufbau von Kommunikationspartner B relevanten Parametereinträge im DB zeigt die folgende Tabelle:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
id	WORD	W#16#000F	Referenz auf diese Verbindung
connection_type	BYTE	B#16#11	Verbindungstyp: TCP/IP native
active_est	BOOL	FALSE	Passiver Verbindungsaufbau
local_device_id	BYTE	B#16#02	Kommunikation über die integrierte Ethernet-Schnittstelle
local_tsap_id_len	BYTE	B#16#02 (nur dieser Wert ist möglich)	verwendete Länge des Parameters local_tsap_id
rem_staddr_len	BYTE	B#16#04	Länge der Adresse des remoten Verbindungsendpunkts: <ul style="list-style-type: none"> • 4: gültige IP-Adresse im Parameter rem_staddr
rem_tsap_id_len	BYTE	B#16#00 (nur dieser Wert ist möglich)	verwendete Länge des Parameters rem_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	"2005" <ul style="list-style-type: none"> • local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = irrelevant 	Lokale Port-Nr.: 2005 = W#16#07D5
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.142" <ul style="list-style-type: none"> • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#8E (142) • rem_staddr[5-6] = irrelevant 	IP-Adresse des remoten Verbindungsendpunkts

Beispiel 3: Zwei S7-300-CPUs mit integrierter PROFINET-Schnittstelle (Bsp. für Kommunikation über UDP)

Die beiden Kommunikationspartner sind zwei CPUs 319-3 PN/DP mit Firmwarestand V2.4.0. Die folgende Tabelle zeigt die wichtigsten Daten der beiden Kommunikationspartner:

Eigenschaft	Kommunikationspartner A: CPU 319-3 PN/DP (FW V2.4.0)	Kommunikationspartner B: CPU 319-3 PN/DP (FW V2.4.0)
Sender/Empfänger	Sender	Empfänger
IP-Adresse	192.168.3.142	192.168.3.125
Lokale Port-Nr.	2004	2005

Die für den Sender (Kommunikationspartner A) relevanten Einträge im DB für die Parametrierung des lokalen Kommunikationszugangspunkts zeigt die folgende Tabelle:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
id	WORD	W#16#0014	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems
connection_type	BYTE	B#16#13	Verbindungstyp: UDP
active_est	BOOL	FALSE	Nur dieser Wert ist möglich beim Verbindungstyp UDP.
local_device_id	BYTE	B#16#03	Kommunikation über die integrierte Ethernet-Schnittstelle
local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D4 • local_tsap_id[3-16] = irrelevant 	Lokale Port-Nr.: 2004 = W#16#07D4

Die für den Empfänger (Kommunikationspartner B) relevanten Einträge im DB für die Parametrierung des lokalen Kommunikationszugangspunkts zeigt die folgende Tabelle:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
id	WORD	W#16#000F	Referenz auf diese Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems
connection_type	BYTE	B#16#13	Verbindungstyp: UDP
active_est	BOOL	FALSE	Nur dieser Wert ist möglich beim Verbindungstyp UDP.
local_device_id	BYTE	B#16#03	Kommunikation über die integrierte Ethernet-Schnittstelle
local_tsap_id_len	BYTE	B#16#02	verwendete Länge des Parameters local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	<ul style="list-style-type: none"> • local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = irrelevant 	Lokale Port-Nr.: 2005 = W#16#07D5

Beim Aufruf des FB 67 "TUSEND" auf dem Sender übergeben Sie in einem DB die folgenden Adressparameter des Empfängers:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
rem_ip_addr	ARRAY [1..4] of BYTE	<ul style="list-style-type: none"> • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#7D (125) 	IP-Adresse des Empfängers: 192.168.3.125
rem_port_nr	ARRAY [1..2] of BYTE	<ul style="list-style-type: none"> • rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D5 	Port-Nr. des Empfängers: 2005 = W#16#07D5

Beim Aufruf des FB 68 "TURCV" auf dem Empfänger erhalten Sie in einem DB die folgenden Adressparameter des Senders:

Parameter	Daten- typ	Wert im Beispiel	Beschreibung
rem_ip_addr	ARRAY [1..4] of BYTE	<ul style="list-style-type: none"> • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#8E (142) 	IP-Adresse des Senders: 192.168.3.142
rem_port_nr	ARRAY [1..2] of BYTE	<ul style="list-style-type: none"> • rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D4 	Port-Nr. des Senders: 2004 = W#16#07D4

1.7 Verbindung aufbauen mit dem FB 65 "TCON"

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.

Arbeitsweise

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.

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 STATUS 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 REQUEST, aktiviert den Verbindungsaufbau bei steigender Flanke
ID	INPUT	WORD	M, D, Konst.	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: W#16#0001 bis W#16#0FFF
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter DONE: <ul style="list-style-type: none"> 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> BUSY = 1: Der Auftrag ist noch nicht beendet. BUSY = 0: Der Auftrag ist beendet.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
CONNECT	IN_OUT	ANY	D	Zeiger auf die zugehörige Verbindungsbeschreibung (UDT 65), siehe Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP und Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Verbindung konnte aufgebaut werden
0	7000	Aufruf mit REQ=0, kein Anstoß zum Verbindungsaufbau
0	7001	Erstaufruf mit REQ=1, Verbindung wird aufgebaut
0	7002	Zwischenaufruf (REQ irrelevant), Verbindung wird aufgebaut
1	8086	Der Parameter ID darf nicht den Wert Null haben.
0	8087	Maximale Anzahl der Verbindungen ist erreicht, keine weitere Verbindung möglich
1	809B	Die in der Verbindungsbeschreibung angegebene local_device_id passt nicht zur Ziel-CPU.
1	80A3	Es wird versucht, eine bestehende Verbindung erneut aufzubauen.
1	80A7	Kommunikationsfehler: Sie haben nach einem nicht abgeschlossenen TCON einen TDISCON aufgerufen. Der Verbindungsaufbau wurde durch Aufruf eines TDISCON vorzeitig abgebrochen.
1	80B3	Inkonsistente Parametrierung: <ul style="list-style-type: none"> • Fehler in der Verbindungsbeschreibung • lokaler Port (Parameter local_tsap_id) in einer anderen Verbindungsbeschreibung schon vorhanden • id in der Verbindungsbeschreibung und Parameter ID sind unterschiedlich
1	80B4	Sie haben bei der Protokollvariante ISO on TCP (connection_type = B#16#12) beim passiven Verbindungsaufbau (active_est = FALSE) die Bedingung "local_tsap_id_len >= B#16#02" oder die Bedingung "local_tsap_id[1] = B#16#E0" oder beide verletzt.
1	80C3	Temporärer Ressourcenmangel der CPU
1	80C4	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> • Verbindung kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametrier
1	8722	Parameter CONNECT: Quellbereich ungültig, Bereich im DB nicht vorhanden
1	8732	Parameter CONNECT: Die DB-Nummer liegt außerhalb des CPU-spezifischen Nummernbands
1	873A	Parameter CONNECT: Zugriff auf Verbindungsbeschreibung nicht möglich (z.B. DB nicht vorhanden)
1	877F	Parameter CONNECT: Interner Fehler, z. B. unzulässige ANY-Referenz

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP
 Parametrierung des lokalen Kommunikationszugangspunkts bei UDP
 Verbindung abbauen mit dem FB 66 "TDISCON"
 Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"
 Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"
 Senden von Daten über UDP mit dem FB 67 "TUSEND"
 Empfangen von Daten über UDP mit dem FB 68 "TURCV"

1.8 Verbindung abbauen mit dem FB 66 "TDISCON"

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.

Arbeitsweise

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.

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 STATUS 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 REQUEST, aktiviert den Abbau der durch ID gegebenen Verbindung. Der Anstoß erfolgt bei steigender Flanke.
ID	INPUT	WORD	M, D, Konst.	Referenz auf die abzubauen 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: W#16#0001 bis W#16#0FFF
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter DONE: <ul style="list-style-type: none"> 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> BUSY = 1: Der Auftrag ist noch nicht beendet. BUSY = 0: Der Auftrag ist beendet.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Verbindung konnte abgebaut werden
0	7000	Erstaufruf mit REQ=0, kein Anstoß zum Verbindungsabbau
0	7001	Erstaufruf mit REQ=1, Verbindung wird abgebaut
0	7002	Zwischenaufruf (REQ irrelevant), Verbindung wird abgebaut
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	80A3	Es wird versucht, eine nicht bestehende Verbindung abzubauen
1	80C4	Temporärer Kommunikationsfehler: Schnittstelle wird neu parametrier

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP
 Parametrierung des lokalen Kommunikationszugangspunkts bei UDP
 Verbindung aufbauen mit dem FB 65 "TCON"
 Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"
 Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"
 Senden von Daten über UDP mit dem FB 67 "TUSEND"
 Empfangen von Daten über UDP mit dem FB 68 "TURCV"

1.9 Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"

Beschreibung

Der FB 63 "TSEND" sendet Daten über eine bestehende Kommunikationsverbindung.

Arbeitsweise

Der FB 63 "TSEND" 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.

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 STATUS entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

Hinweis

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 REQUEST, aktiviert den Sendevorgang bei steigender Flanke. Beim erstmaligen Aufruf mit REQ=1 werden Daten aus dem mit Parameter DATA angegebenen Bereich übergeben.
ID	INPUT	WORD	M, D, Konst.	Referenz auf die zugehörige Verbindung. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
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"> • 1 bis 1460, falls Verbindungstyp = B#16#01 • 1 bis 8192, falls Verbindungstyp = B#16#11 • 1 bis 1452, falls Verbindungstyp = B#16#12 und ein CP benutzt wird • 1 bis 8192, falls Verbindungstyp = B#16#12 und kein CP benutzt wird
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter DONE: <ul style="list-style-type: none"> • 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. • 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> • BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. • BUSY = 0: Der Auftrag ist beendet.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> • ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
DATA	IN_OUT	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge Die Adresse verweist auf: <ul style="list-style-type: none"> • das Prozessabbild der Eingänge • das Prozessabbild der Ausgänge • einen Merker • einen Datenbaustein

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Sendeauftrag wurde ohne Fehler abgeschlossen
0	7000	Erstaufruf mit REQ=0, kein Anstoß zum Senden
0	7001	Erstaufruf mit REQ=1, Anstoß des Sendevorgangs
0	7002	Zwischenaufruf (REQ irrelevant), Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendepuffer DATA zu.
1	8085	Parameter LEN hat den Wert 0 oder ist größer als der größte zulässige Wert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
0	8088	Parameter LEN ist größer als der in DATA angegebene Speicherbereich
1	80A1	Kommunikationsfehler: <ul style="list-style-type: none"> • Zu der angegebenen ID wurde noch kein FB 65 "TCON" aufgerufen • Die angegebene Verbindung wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich. • Schnittstelle wird neu initialisiert
1	80B3	Der parametrisierte Verbindungstyp (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie den FB 67 "TUSEND".
1	80C3	Die Betriebsmittel (Speicher) der CPU sind temporär belegt
1	80C4	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> • Die Verbindung zum Kommunikationspartner kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametrisiert.
1	8822	Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden
1	8824	Parameter DATA: Bereichsfehler im ANY-Pointer
1	8832	Parameter DATA: DB-Nummer ist zu groß
1	883A	Parameter DATA: Zugriff auf Sendepuffer nicht möglich (z.B. gelöschter DB)
1	887F	Parameter DATA: Interner Fehler, z. B. unzulässige ANY-Referenz

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Verbindung aufbauen mit dem FB 65 "TCON"

Verbindung abbauen mit dem FB 66 "TDISCON"

Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"

Senden von Daten über UDP mit dem FB 67 "TUSEND"

Empfangen von Daten über UDP mit dem FB 68 "TURCV"

1.10 Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"

Beschreibung

Der FB 64 "TRCV" empfängt Daten über eine bestehende Kommunikationsverbindung.

Für den Empfang und die Weiterverarbeitung der Daten gibt es 2 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
B#16#01 und B#16#11	Der Anwender kann die Variante selbst bestimmen.
B#16#12	Variante 2 (fest)

Die beiden Varianten sind in der folgenden Tabelle näher beschrieben.

Empfangene Daten ...	Wert(ebereich) von LEN	Wert(ebereich) von RCVD_LEN	Beschreibung
stehen sofort zur Verfügung	0	1 bis x	Sie übergeben einen Puffer, dessen Länge x im ANY-Pointer des Empfangspuffers hinterlegt ist (Parameter DATA). Nachdem ein Datenblock empfangen wurde, wird dieser sofort im Empfangspuffer zur Verfügung gestellt. Die Anzahl der empfangenen Daten (Parameter RCVD_LEN) kann maximal so groß sein wie die im Parameter DATA hinterlegte Größe. Der Empfang wird angezeigt durch NDR = 1.
im Empfangspuffer speichern. Sie stehen zur Verfügung, sobald die projektierte Länge erreicht wird	<ul style="list-style-type: none"> • 1 bis 1460, falls Verbindungstyp = B#16#01 • 1 bis 8192, falls Verbindungstyp = B#16#11 • 1 bis 1452, falls Verbindungstyp = B#16#12 und ein CP benutzt wird • 1 bis 8192, falls Verbindungstyp = B#16#12 und kein CP benutzt wird 	gleicher Wert wie im Parameter LEN	Sie übergeben die Empfangslänge am Parameter LEN. Wenn diese parametrisierte Länge erreicht ist, werden die Empfangsdaten im Parameter DATA zur Verfügung gestellt (NDR = 1).

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.

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	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 STATUS entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

Hinweis

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	Steuerparameter enabled to receive: Mit EN_R= 1 wird der FB 64 "TRCV" empfangsbereit.
ID	INPUT	WORD	M, D, Konst.	Referenz auf die zugehörige Verbindung. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	INPUT	INT	E, A, M, D, L	<ul style="list-style-type: none"> LEN = 0 (Ad-Hoc-Mode): implizite Längenangabe im ANY-Pointer DATA verwenden. Die empfangenen Daten werden beim Bausteinanruf unmittelbar bereitgestellt. Die Anzahl der empfangenen Daten steht in RCVD_LEN zur Verfügung. 1 <= LEN <= max: 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: max = 1460 beim Verbindungstyp B#16#01, max = 8192 beim Verbindungstyp B#16#11, max = 1452 beim Verbindungstyp B#16#12 mit CP-Einsatz, max = 8192 beim Verbindungstyp B#16#12 ohne CP-Einsatz
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter NDR: <ul style="list-style-type: none"> NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch NDR = 1: Auftrag wurde erfolgreich abgeschlossen
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. BUSY = 0: Der Auftrag ist beendet.
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: 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"> das Prozessabbild der Eingänge das Prozessabbild der Ausgänge einen Merker einen Datenbaustein

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.
0	7000	Erstaufruf mit REQ=0, kein Anstoß zum Empfangen
0	7001	Baustein ist empfangsbereit
0	7002	Zwischenaufruf, Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase schreibt das Betriebssystem Daten in den Empfangspuffer DATA. Deshalb können im Fehlerfall inkonsistente Daten im Empfangspuffer stehen.
1	8085	Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben den Wert von LEN gegenüber dem Erstaufruf geändert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	<ul style="list-style-type: none"> • Zielpuffer (DATA) ist zu klein • Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich Fehlerbehebung, falls Verbindungstyp = B#16#12: Vergrößern Sie den Zielpuffer DATA.
1	80A1	Kommunikationsfehler: <ul style="list-style-type: none"> • Zu der angegebenen ID wurde noch kein FB 65 "TCON" aufgerufen • Die angegebene Verbindung wird momentan abgebaut. Ein Empfangsvorgang über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu parametrierd.
1	80B3	Der parametrierdte Verbindungstyp (Parameter connection_type in der Verbindungsbeschreibung) ist UDP. Bitte verwenden Sie den FB 68 "TURCV".
1	80C3	Die Betriebsmittel (Speicher) der CPU sind temporär belegt
1	80C4	Temporärer Kommunikationsfehler: Die Verbindung wird gerade aufgebaut.
1	8922	Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden
1	8924	Parameter DATA: Bereichsfehler im ANY-Pointer
1	8932	Parameter DATA: DB-Nummer ist zu groß
1	893A	Parameter DATA: Zugriff auf Empfangspuffer nicht möglich (z.B. gelöschter DB)
1	897F	Parameter DATA: Interner Fehler, z. B. unzulässige ANY-Referenz

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Verbindung aufbauen mit dem FB 65 "TCON"

Verbindung abbauen mit dem FB 66 "TDISCON"

Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"

Senden von Daten über UDP mit dem FB 67 "TUSEND"

Empfangen von Daten über UDP mit dem FB 68 "TURCV"

1.11 Senden von Daten über UDP mit dem FB 67 "TUSEND"

Beschreibung

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

Hinweis

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 FBs 65 "TCON" und 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.

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 STATUS entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

Hinweis

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 REQUEST, aktiviert den Sendevorgang bei steigender Flanke. Beim erstmaligen Aufruf mit REQ=1 werden Bytes aus dem mit Parameter DATA angegebenen Bereich übergeben.
ID	INPUT	WORD	M, D, Konst.	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	INPUT	INT	E, A, M, D, L	Anzahl der Bytes, die mit dem Auftrag gesendet werden sollen Wertebereich: 1 bis 1460
DONE	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter DONE: <ul style="list-style-type: none"> 0: Auftrag wurde noch nicht gestartet oder wird noch ausgeführt. 1: Auftrag wurde fehlerfrei ausgeführt
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. BUSY = 0: Der Auftrag ist beendet.
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: Fehlerinformation
DATA	IN_OUT	ANY	E, A, M, D	Sendebereich, enthält Adresse und Länge Die Adresse verweist auf: <ul style="list-style-type: none"> das Prozessabbild der Eingänge das Prozessabbild der Ausgänge einen Merker einen Datenbaustein
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

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Sendeauftrag wurde ohne Fehler abgeschlossen
0	7000	Erstaufruf mit REQ=0, kein Anstoß zum Senden
0	7001	Erstaufruf mit REQ=1, Anstoß des Sendevorgangs
0	7002	Zwischenaufruf (REQ irrelevant), Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase greift das Betriebssystem auf die Daten im Sendepuffer DATA zu.
1	8085	Parameter LEN hat den Wert 0 oder ist größer als der größte zulässige Wert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
0	8088	Parameter LEN ist größer als der in DATA angegebene Speicherbereich
1	80A1	Kommunikationsfehler: <ul style="list-style-type: none"> • Zu der angegebenen ID wurde noch kein FB 65 "TCON" aufgerufen • Die angegebene Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Sendevorgang über diese Verbindung ist nicht möglich. • Schnittstelle wird neu initialisiert
1	80B3	<ul style="list-style-type: none"> • Der parametrierte Verbindungstyp (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie den FB 63 "TSEND". • Parameter ADDR: ungültige Angaben für Port-Nr. oder IP-Adresse
1	80C3	Die Betriebsmittel (Speicher) der CPU sind temporär belegt
1	80C4	Temporärer Kommunikationsfehler: <ul style="list-style-type: none"> • Die Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems kann momentan nicht aufgebaut werden. • Die Schnittstelle wird neu parametrier.
1	8822	Parameter DATA: Quellbereich ungültig, Bereich im DB nicht vorhanden
1	8824	Parameter DATA: Bereichsfehler im ANY-Pointer
1	8832	Parameter DATA: DB-Nummer ist zu groß
1	883A	Parameter DATA: Zugriff auf Sendepuffer nicht möglich (z. B. gelöschter DB)
1	887F	Parameter DATA: Interner Fehler, z. B. unzulässige ANY-Referenz

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Verbindung aufbauen mit dem FB 65 "TCON"

Verbindung abbauen mit dem FB 66 "TDISCON"

Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"

Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"

Empfangen von Daten über UDP mit dem FB 68 "TURCV"

1.12 Empfangen von Daten über UDP mit dem FB 68 "TURCV"

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.

Arbeitsweise

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.

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 STATUS entnehmen.
FALSE	FALSE	FALSE	Dem FB wurde kein (neuer) Auftrag erteilt.

Hinweis

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 EN_R = 1 wird der FB 68 "TURCV" empfangsbereit.
ID	INPUT	WORD	M, D, Konst.	Referenz auf die zugehörige Verbindung zwischen Anwenderprogramm und Kommunikationsschicht des Betriebssystems. ID muss identisch sein mit dem zugehörigen Parameter id in der lokalen Verbindungsbeschreibung. Wertebereich: W#16#0001 bis W#16#0FFF
LEN	INPUT	INT	E, A, M, D, L	1 <= LEN <= 1460: Anzahl der Bytes, die empfangen werden sollen. Die empfangenen Daten werden beim Baustein aufruf unmittelbar bereitgestellt. Die Anzahl der empfangenen Daten steht in RCVD_LEN zur Verfügung.
NDR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter NDR: <ul style="list-style-type: none"> NDR = 0: Auftrag wurde noch nicht gestartet oder läuft noch NDR = 1: Auftrag wurde erfolgreich abgeschlossen
ERROR	OUTPUT	BOOL	E, A, M, D, L	Zustandsparameter ERROR: <ul style="list-style-type: none"> ERROR=1: Bei der Bearbeitung ist ein Fehler aufgetreten. STATUS liefert detaillierte Auskunft über die Art des Fehlers
BUSY	OUTPUT	BOOL	E, A, M, D, L	<ul style="list-style-type: none"> BUSY = 1: Der Auftrag ist noch nicht beendet. Es kann kein neuer Auftrag angestoßen werden. BUSY = 0: Der Auftrag ist beendet.
STATUS	OUTPUT	WORD	M, D	Zustandsparameter STATUS: 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"> das Prozessabbild der Eingänge das Prozessabbild der Ausgänge einen Merker einen Datenbaustein
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

Fehlerinformationen

ERROR	STATUS (W#16#...)	Erläuterung
0	0000	Neue Daten wurden übernommen. Die aktuelle Länge der empfangenen Daten wird in RCVD_LEN angezeigt.
0	7000	Erstaufruf mit REQ=0, kein Anstoß zum Empfangen
0	7001	Baustein ist empfangsbereit
0	7002	Zwischenaufruf, Auftrag ist in Bearbeitung Hinweis: In dieser Bearbeitungsphase schreibt das Betriebssystem Daten in den Empfangspuffer DATA. Deshalb können im Fehlerfall inkonsistente Daten im Empfangspuffer stehen.
1	8085	Parameter LEN ist größer als der größte zulässige Wert, oder Sie haben den Wert von LEN gegenüber dem Erstaufruf geändert
1	8086	Parameter ID liegt in einem unzulässigen Wertebereich
1	8088	<ul style="list-style-type: none"> • Zielpuffer (DATA) ist zu klein • Wert in LEN ist größer als der durch DATA vorgegebene Empfangsbereich
1	80A1	Kommunikationsfehler: <ul style="list-style-type: none"> • Zu der angegebenen ID wurde noch kein FB 65 "TCON" aufgerufen • Die angegebene Verbindung zwischen Anwenderprogramm und der Kommunikationsschicht des Betriebssystems wird momentan abgebaut. Ein Empfangsvorgang über diese Verbindung ist nicht möglich. • Die Schnittstelle wird neu parametrierd.
1	80B3	Der parametrierd Verbindungstyp (Parameter connection_type in der Verbindungsbeschreibung) ist nicht UDP. Bitte verwenden Sie den FB 68 "TRCV".
1	80C3	Die Betriebsmittel (Speicher) der CPU sind temporär belegt
1	80C4	Temporärer Kommunikationsfehler: Die Verbindung wird gerade aufgebaut.
1	8922	Parameter DATA: Zielbereich ungültig, Bereich im DB nicht vorhanden
1	8924	Parameter DATA: Bereichsfehler im ANY-Pointer
1	8932	Parameter DATA: DB-Nummer ist zu groß
1	893A	Parameter DATA: Zugriff auf Empfangspuffer nicht möglich (z.B. gelöschter DB)
1	897F	Parameter DATA: Interner Fehler, z. B. unzulässige ANY-Referenz

Siehe auch:

Parametrierung der Kommunikationsverbindungen bei TCP native und ISO on TCP

Parametrierung des lokalen Kommunikationszugangspunkts bei UDP

Verbindung aufbauen mit dem FB 65 "TCON"

Verbindung abbauen mit dem FB 66 "TDISCON"

Senden von Daten über TCP native und ISO on TCP mit dem FB 63 "TSEND"

Empfangen von Daten über TCP native und ISO on TCP mit dem FB 64 "TRCV"

Senden von Daten über UDP mit dem FB 67 "TUSEND"

Index

E	
Empfangen von Daten mit dem FB 64 "TRCV"	1-28
Empfangen von Daten über UDP mit dem FB 68 "TURCV"	1-36
F	
FB 63 "TSEND"	1-25
FB 64 "TRCV"	1-28, 1-30, 1-31
FB 65 "TCON"	1-20
FB 66 "TDISCON"	1-23
FB 67 "TUSEND"	1-33
FB 68 "TURCV"	1-36, 1-37
O	
offene Kommunikation.....	1-1, 1-2, 1-4, 1-8, 1-10, 1-11
Arbeitsweise der FBs	1-2
Beispiele zur Parametrierung der Kommunikationsverbindungen.....	1-11
Parametrierung der Verbindungen bei TCP native und ISO on TCP	1-4
Parametrierung des lokalen Zugangspunkts bei UDP	1-8
S	
Struktur der Adressinformation des remoten Partners bei UDP	1-10
Übersicht	1-1
offene Kommunikation über Industrial Ethernet	1-1
S	
Senden von Daten mit dem FB 63 "TSEND"	1-25
Senden von Daten über UDP mit dem FB 67 "TUSEND"	1-33
T	
TCON	1-22
TDISCON	1-23
TRCV	1-28
TSEND	1-25
TURCV	1-36
TUSEND	1-33
V	
Verbindung abbauen mit dem FB 66 "TDISCON"	1-23
Verbindung aufbauen mit dem FB 65 "TCON"	1-20

