

SIEMENS

Ingenuity for life

Industry Online Support

Home

Offene Benutzer- kommunikation mit TSEND_C und TRCV_C

SIMATIC S7-1200 CPU

<https://support.industry.siemens.com/cs/ww/de/view/67196808>

Siemens
Industry
Online
Support



Dieser Beitrag stammt aus dem Siemens Industry Online Support. Es gelten die dort genannten Nutzungsbedingungen (www.siemens.com/nutzungsbedingungen).

Security-hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Lösungen, Maschinen, Geräten und/oder Netzwerken unterstützen. Sie sind wichtige Komponenten in einem ganzheitlichen Industrial Security-Konzept. Die Produkte und Lösungen von Siemens werden unter diesem Gesichtspunkt ständig weiterentwickelt. Siemens empfiehlt, sich unbedingt regelmäßig über Produkt-Updates zu informieren.

Für den sicheren Betrieb von Produkten und Lösungen von Siemens ist es erforderlich, geeignete Schutzmaßnahmen (z. B. Zellenschutzkonzept) zu ergreifen und jede Komponente in ein ganzheitliches Industrial Security-Konzept zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen.

Weitergehende Informationen über Industrial Security finden Sie unter <http://www.siemens.com/industrialsecurity>.

Um stets über Produkt-Updates informiert zu sein, melden Sie sich für unseren produktspezifischen Newsletter an. Weitere Informationen hierzu finden Sie unter <http://support.automation.siemens.com>.

Inhaltsverzeichnis

| | | |
|----------|--|----------|
| 1 | Einleitung | 3 |
| 2 | Programmbeispiel | 5 |
| 2.1 | OB100 | 5 |
| 2.2 | OB1 | 5 |
| 2.3 | FC100 "FC_SEND" | 6 |
| 2.3.1 | Ein- und Ausgänge der Anweisung TSEND_C | 6 |
| 2.3.2 | Sendeauftrag starten | 8 |
| 2.3.3 | Kommunikationsverbindung aufbauen und halten | 8 |
| 2.3.4 | LEN | 9 |
| 2.3.5 | Sendebereich | 9 |
| 2.3.6 | "GeneralData".sendReq zurücksetzen | 9 |
| 2.3.7 | STATUS der Anweisung TSEND_C speichern | 9 |
| 2.4 | FC200 "FC_RECV" | 9 |
| 2.4.1 | Ein- und Ausgänge der Anweisung TRCV_C | 11 |
| 2.4.2 | Empfang von Daten freigeben | 13 |
| 2.4.3 | Kommunikationsverbindung aufbauen und halten | 13 |
| 2.4.4 | LEN | 13 |
| 2.4.5 | Empfangsbereich | 13 |
| 2.4.6 | STATUS der Anweisung TRCV_C speichern | 13 |
| 2.4.7 | Länge der empfangenen Daten speichern | 14 |
| 2.5 | Verbindungsparameter nach TCON_IP_RFC | 14 |
| 2.6 | Verbindungsparameter nach TCON_IP_V4 | 15 |

1 Einleitung

Für den Datenaustausch über die integrierte PROFINET-Schnittstelle der S7-1200 CPU können Sie u. a. die offene Benutzerkommunikation mittels der Anweisungen TSEND_C und TRCV_C nutzen.

In STEP 7 (TIA Portal) finden Sie die Anweisungen TSEND_C und TRCV_C in der Task Card "Anweisungen" in der Palette "Kommunikation > Open User Communication".

Tabelle 1-1

| Anweisung | Beschreibung |
|-----------|--|
| TSEND_C | Die Anweisung TSEND_C wird asynchron ausgeführt und hat folgende Funktionen: <ul style="list-style-type: none"> • Kommunikationsverbindung einrichten und aufbauen • Daten über die bestehende Kommunikationsverbindung senden • Kommunikationsverbindung abbauen |
| TRCV_C | Die Anweisung TRCV_C wird asynchron ausgeführt und hat folgende Funktionen: <ul style="list-style-type: none"> • Kommunikationsverbindung einrichten und aufbauen • Daten über die bestehende Kommunikationsverbindung empfangen • Kommunikationsverbindung abbauen |

Hierbei werden folgende Protokolle unterstützt:

- ISO-on-TCP
- TCP
- UDP

Beschreibung des Programmbeispiels

Das Programmbeispiel ist in STEP 7 (TIA Portal) V14 erstellt. Das Projekt beinhaltet eine S7-1200 CPU und eine S7-1500 CPU inklusive Hardware-Konfiguration und Anwenderprogramm.

In beiden CPUs ist das Merkerbyte 10 (MB10) als Taktmerkerbyte konfiguriert.

Das Anwenderprogramm beinhaltet den Aufruf der Anweisungen TSEND_C und TRCV_C sowie die Parametrierung der ISO-on-TCP Verbindung für den Datenaustausch zwischen S7-1200 CPU und S7-1500 CPU. Die Verbindungsparameter zum Aufbau der ISO-on-TCP Verbindung werden im Datenbaustein "GeneralData" gespeichert.

Das Anwenderprogramm der S7-1200 CPU besteht aus folgenden Bausteinen.

Tabelle 1-2

| Baustein | Symbolischer Name | Beschreibung |
|----------|-------------------|---|
| OB100 | Startup | Anlauf-OB |
| OB1 | Main | Im OB1 werden die Funktionen FC100 "FC_SEND" und FC200 "FC_RECV" aufgerufen. |
| FC100 | FC_SEND | Die Funktion FC100 "FC_SEND" ruft intern die Anweisung TSEND_C auf, um Daten über eine ISO-on-TCP Verbindung zu senden. |

| Baustein | Symbolischer Name | Beschreibung |
|----------|-------------------|--|
| FC200 | FC_RECV | Die Funktion FC200 "FC_SEND" ruft intern die Anweisung TSEND_C auf, um Daten über eine ISO-on-TCP Verbindung zu empfangen. |
| DB3 | SendData | Im Datenbaustein DB3 werden die Sendedaten gespeichert. |
| DB5 | RecvData | Im Datenbaustein DB5 werden die Empfangsdaten gespeichert. |

2 Programmbeispiel

2.1 OB100

Der OB100 ist ein Anlauf-OB und wird beim Neustart (Warmstart) der S7-1200 CPU durchlaufen. In diesem OB werden die folgenden Variablen auf den Wert "1" gesetzt, um zwei ISO-on-TCP Verbindungen aufzubauen und den Auftrag zum Empfang der Daten freizugeben.

- "GeneralData".contSend
- "GeneralData".contRcv
- "GeneralData".enable

2.2 OB1

Der OB1 wird zyklisch aufgerufen. Im OB1 werden die Funktionen FC100 "FC_SEND" und FC200 "FC_RECV" aufgerufen.

Abbildung 2-1

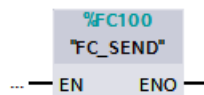
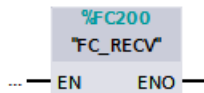


Abbildung 2-2

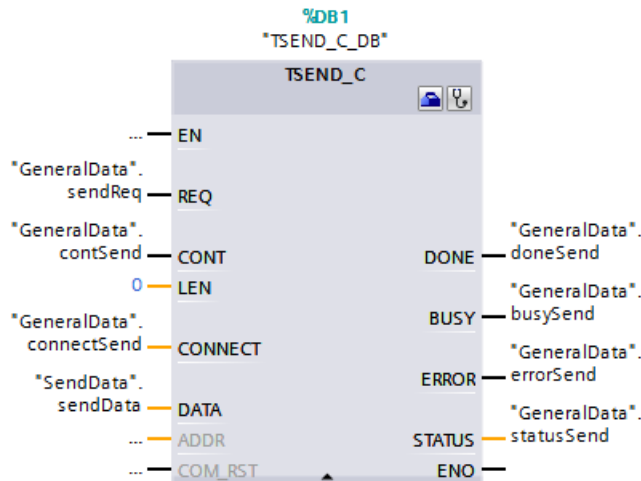


2.3 FC100 "FC_SEND"

Die Funktion FC100 "FC_SEND" ruft die Anweisung TSEND_C auf, um eine Kommunikationsverbindung über ISO-on-TCP, TCP oder UDP aufzubauen und Daten über die Kommunikationsverbindung zu senden.

Die folgende Abbildung zeigt den Aufruf der Anweisung TSEND_C.

Abbildung 2-3



2.3.1 Ein- und Ausgänge der Anweisung TSEND_C

Eingänge

Die folgende Tabelle zeigt eine Übersicht:

- der Eingänge der Anweisung TSEND_C und
- der Variablen, die den Eingängen zugeordnet sind.

Tabelle 2-1

| Eingang | Datentyp | Variable / Wert | Beschreibung |
|---------|----------|------------------------|---|
| REQ | BOOL | "GeneralData".sendReq | Startet den Sendeauftrag bei einer steigenden Flanke |
| CONT | BOOL | "GeneralData".contSend | Steuert die Kommunikationsverbindung: 0: Kommunikationsverbindung trennen 1: Kommunikationsverbindung aufbauen und halten Wenn CONT=1, dann richtet die Anweisung TSEND_C eine Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der S7-1200 CPU gehalten und überwacht. |

| Eingang | Datentyp | Variable / Wert | Beschreibung |
|---------|----------|---------------------------|--|
| LEN | UINT | 0 | Anzahl von Bytes, die mit dem Auftrag gesendet werden. |
| CONNECT | VARIANT | "GeneralData".connectSend | Zeiger auf die Struktur zur Verbindungs-beschreibung <ul style="list-style-type: none"> Bei ISO-on-TCP verwenden Sie den Systemdatentyp TCON_IP_RFC. Zur Beschreibung siehe Kapitel 2.5) Bei TCP oder UDP verwenden Sie den Systemdatentyp TCON_IP_V4. Zur Beschreibung siehe Kapitel 2.6) |
| DATA | VARIANT | "SendData".sendData | Zeiger auf den Sendebereich, der die Adresse und die Länge der zu sendenden Daten enthält |

Ausgänge

Die folgende Tabelle zeigt eine Übersicht:

- der Ausgänge der Anweisung TSEND_C und
- der Variablen, die den Ausgängen zugeordnet sind.

Tabelle 2-2

| Ausgang | Datentyp | Variable | Beschreibung |
|---------|----------|------------------------|---|
| DONE | BOOL | "GeneralData".doneSend | Statusparameter mit folgenden Werten: 0: Auftrag noch nicht gestartet oder noch in Bearbeitung 1: Auftrag fehlerfrei ausgeführt |
| BUSY | BOOL | "GeneralData".busySend | Statusparameter mit folgenden Werten: 0: Auftrag noch nicht gestartet oder bereits beendet 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden |

| Ausgang | Datentyp | Variable | Beschreibung |
|---------|----------|-------------------------|--|
| ERROR | BOOL | "GeneralData".errorSend | Statusparameter mit folgenden Werten: 0: kein Fehler 1: Fehler aufgetreten |
| STATUS | WORD | "GeneralData".status | Status der Anweisung |

2.3.2 Sendeauftrag starten

Der Start des Sendeauftrags wird mit dem Taktmerker M10.7 und mit den Variablen "GeneralData".sendReq und "GeneralData".contSend gesteuert.

Beim Absenden von Daten muss der Eingang CONT der Anweisung TSEND_C auf den Wert "1" gesetzt sein, um die Kommunikationsverbindung aufzubauen. Der Eingang CONT wird über die Variable "GeneralData".contSend auf den Wert "1" gesetzt. D .h. wenn "GeneralData".contSend auf den Wert "1" gesetzt ist, kann der Sendeauftrag gestartet werden.

Wenn der Sendeauftrag läuft, dann ist "GeneralData".sendReq auf den Wert "1" gesetzt, d. h. solange der Sendeauftrag läuft kann kein neuer Sendeauftrag gestartet werden.

Abbildung 2-4



2.3.3 Kommunikationsverbindung aufbauen und halten

Beim Neustart (Warmstart) der S7-1200 CPU wird die Variable "GeneralData".contSend dauerhaft auf den Wert "1" gesetzt.

Der Eingang CONT der Anweisung TSEND_C wird über die Variable "GeneralData".contSend dauerhaft auf den Wert "1" gesetzt, um die Kommunikationsverbindung aufzubauen und zu halten.

Wenn die S7-1200 CPU in den Betriebszustand STOP geht, wird eine bestehende Kommunikationsverbindung abgebrochen und die eingerichtete Kommunikationsverbindung entfernt. Zum Erneuten Einrichten und Aufbauen der Kommunikationsverbindung muss die Anweisung TSEND_C erneut ausgeführt werden.

2.3.4 LEN

In diesem Programmbeispiel wird am Parameter DATA ein Sendebereich mit optimiertem Zugriff verwendet. Aus diesem Grund muss am Parameter LEN der Wert "0" angegeben werden.

2.3.5 Sendebereich

In diesem Programmbeispiel wird der Sendebereich "SendData".sendData mit optimiertem Zugriff definiert.

Der Sendebereich ist 100 Byte lang und die zu sendenden Daten sind im Datenbaustein DB3 gespeichert.

2.3.6 "GeneralData".sendReq zurücksetzen

Wenn kein Sendeauftrag läuft, hat der Ausgang BUSY der Anweisung TSEND_C den Wert "0" und "GeneralData".sendReq wird auf den Wert "0" zurückgesetzt. Damit kann ein neuer Sendeauftrag angestoßen werden kann, wenn der vorhergehende abgeschlossen ist.

Abbildung 2-5



2.3.7 STATUS der Anweisung TSEND_C speichern

Wenn die Anweisung TSEND_C erfolgreich oder mit Fehler ausgeführt wurde, wird der Status der Anweisung TSEND_C in der Variable "GeneralData".statusSendSave gespeichert. Der Status gibt Ihnen Auskunft über die Ursache, wenn der Sendeauftrag nicht läuft.

Abbildung 2-6

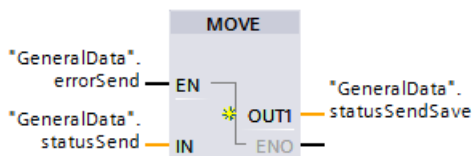
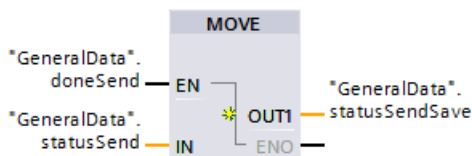


Abbildung 2-7

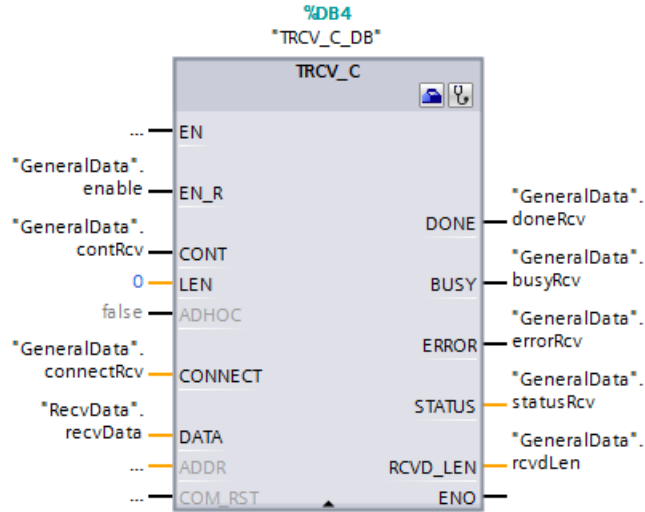


2.4 FC200 "FC_RECV"

Die Funktion FC200 "FC_RECV" ruft die Anweisung TRCV_C auf, um eine Kommunikationsverbindung über ISO-on-TCP, TCP oder UDP aufzubauen und Daten über die Kommunikationsverbindung zu empfangen.

Die folgende Abbildung zeigt den Aufruf der Anweisung TRCV_C.

Abbildung 2-8



2.4.1 Ein- und Ausgänge der Anweisung TRCV_C

Eingänge

Die folgende Tabelle zeigt eine Übersicht:

- der Eingänge der Anweisung TRCV_C und
- der Variablen, die den Eingängen zugeordnet sind.

Tabelle 2-3

| Eingang | Datentyp | Variable | Beschreibung |
|---------|----------|-----------------------|--|
| EN_R | BOOL | "GeneralData".enable | Empfangsfreigabe, bei EN_R = 1 |
| CONT | BOOL | "GeneralData".contRcv | Steuert die Kommunikationsverbindung: 0: Kommunikationsverbindung trennen 1: Kommunikationsverbindung aufbauen und halten Wenn CONT=1, dann richtet die Anweisung TRCV_C eine Kommunikationsverbindung ein und baut diese auf. Nach dem Einrichten und Aufbauen der Verbindung wird sie automatisch von der S7-1200 CPU gehalten und überwacht. |
| LEN | UINT | 0 | Maximale Anzahl von Bytes, die mit dem Auftrag gesendet werden. Hinweis Wenn Sie einen Empfangsbereich mit optimiertem Zugriff am Parameter DATA verwenden, dann muss am Parameter DATA der Wert "0" verwendet werden. |

| Eingang | Datentyp | Variable | Beschreibung |
|---------|----------|--------------------------|--|
| CONNECT | VARIANT | "GeneralData".connectRcv | Zeiger auf die Struktur zur Verbindungs-beschreibung <ul style="list-style-type: none"> Bei ISO-on-TCP verwenden Sie den Systemdatentyp TCON_IP_RFC. Zur Beschreibung siehe Kapitel 2.5) Bei TCP oder UDP verwenden Sie den Systemdatentyp TCON_IP_V4. Zur Beschreibung siehe Kapitel 2.6) |
| DATA | VARIANT | "RecvData".recvData | Zeiger auf den Empfangsbereich, der die Adresse und die Länge der zu sendenden Daten enthält |

Ausgänge

Die folgende Tabelle zeigt eine Übersicht:

- der Ausgänge der Anweisung TRCV_C und
- der Variablen, die den Ausgängen zugeordnet sind.

Tabelle 2-4

| Ausgang | Datentyp | Variable | Beschreibung |
|----------|----------|-------------------------|---|
| DONE | BOOL | "GeneralData".doneRcv | Statusparameter mit folgenden Werten: 0: Auftrag noch nicht gestartet oder noch in Bearbeitung 1: Auftrag fehlerfrei ausgeführt |
| BUSY | BOOL | "GeneralData".busyRcv | Statusparameter mit folgenden Werten: 0: Auftrag noch nicht gestartet oder bereits beendet 1: Auftrag noch nicht beendet. Ein neuer Auftrag kann nicht gestartet werden |
| ERROR | BOOL | "GeneralData".errorRcv | Statusparameter mit folgenden Werten: 0: kein Fehler 1: Fehler aufgetreten |
| STATUS | WORD | "GeneralData".statusRcv | Status der Anweisung |
| RCVD_LEN | UINT | "GeneralData".rcvdLen | Tatsächlich empfangene Datenmenge in Byte |

2.4.2 Empfang von Daten freigeben

Beim Neustart (Warmstart) der S7-1200 CPU wird die Variable "GeneralData".enable dauerhaft auf den Wert "1" gesetzt. Der Eingang EN_R der Anweisung TRCV_C wird über die Variable "GeneralData".enable dauerhaft auf den Wert "1" gesetzt, um den Empfang von Daten freizugeben.

2.4.3 Kommunikationsverbindung aufbauen und halten

Beim Neustart (Warmstart) der S7-1200 CPU wird die Variable "GeneralData".contRcv dauerhaft auf den Wert "1" gesetzt.

Der Eingang CONT der Anweisung TRCV_C wird über die Variable "GeneralData".contRcv dauerhaft auf den Wert "1" gesetzt, um die Kommunikationsverbindung aufzubauen und zu halten.

Wenn die S7-1200 CPU in den Betriebszustand STOP geht, wird eine bestehende Kommunikationsverbindung abgebrochen und die eingerichtete Kommunikationsverbindung entfernt. Zum Erneuten Einrichten und Aufbauen der Kommunikationsverbindung muss die Anweisung TRCV_C erneut ausgeführt werden.

2.4.4 LEN

In diesem Programmbeispiel wird am Parameter DATA ein Empfangsbereich mit optimiertem Zugriff verwendet. Aus diesem Grund muss am Parameter LEN der Wert "0" angegeben werden.

2.4.5 Empfangsbereich

In diesem Programmbeispiel wird der Empfangsbereich "RecvData".recvData mit optimiertem Zugriff definiert.

Der Empfangsbereich ist 100 Byte lang und die empfangenen Daten werden im Datenbaustein DB5 gespeichert.

2.4.6 STATUS der Anweisung TRCV_C speichern

Wenn die Anweisung TRCV_C erfolgreich oder mit Fehler ausgeführt wurde, wird der Status der Anweisung TRCV_C in der Variable "GeneralData".statusRcvSave gespeichert. Der Status gibt Ihnen Auskunft über die Ursache, wenn die Daten nicht erfolgreich empfangen wurden.

Abbildung 2-9

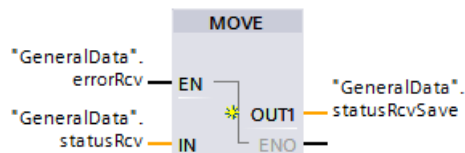
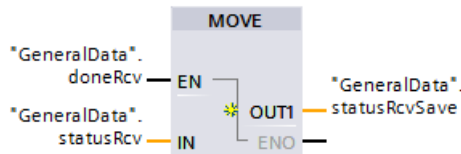


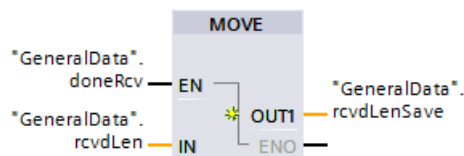
Abbildung 2-10



2.4.7 Länge der empfangenen Daten speichern

Wenn die Anweisung TRCV_C erfolgreich ausgeführt wurde, wird die Länge der empfangenen Daten in der Variable "GeneralData".rcvdLenSave gespeichert.

Abbildung 2-11



2.5 Verbindungsparameter nach TCON_IP_RFC

Um die Kommunikationsverbindung bei ISO-on-TCP zu parametrieren wird für die CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_IP_RFC verwendet. Die feste Datenstruktur des TCON_IP_RFC enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden.

Die folgende Tabelle zeigt den Aufbau der Verbindungsbeschreibung nach TCON_IP_RFC.

Tabelle 2-5

| Byte | Parameter | Datentyp | Wert | Beschreibung |
|---------|-------------------|----------|-------|---|
| 0 und 1 | Interfaceld | HW_ANY | 64 | Hardware-Kennung der integrierten PROFINET-Schnittstelle der CPU |
| 2 und 3 | ID | CONN_OUC | 1 | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C und TRCV unter ID angeben. |
| 4 | ConnectionType | BYTE | 16#0C | Verbindungstyp: <ul style="list-style-type: none"> 12: ISO-on-TCP (16#0C hex = 12 dez) |
| 5 | ActiveEstablished | BOOL | False | Kennung für die Art des Verbindungsaufbaus <ul style="list-style-type: none"> False: passiver Verbindungsaufbau True: aktiver Verbindungsaufbau |

| Byte | Parameter | Datentyp | Wert | Beschreibung |
|--------------|-----------------|-------------------------|------|--|
| 6 bis 9 | RemoteAddress | ARRAY [1..4] of BYTE | - | IP-Adresse des Partner- Endpunkts, z. B. für 192.168.0.3 <ul style="list-style-type: none"> • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 3 |
| 10 bis 43 | RemoteTSelector | TSelector | - | TSelector des entfernten Verbindungspartners: <ul style="list-style-type: none"> • TSelLength = Wertebereich 0 bis 32 als UINT • TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes |
| 44 bis 77 | LocalTSelector | TSelector | - | TSelector des lokalen Verbindungspartners: <ul style="list-style-type: none"> • TSelLength = Wertebereich 0 bis 32 als UINT • TSel[1-32] = Wertebereich jeweils 0 bis 255 in Bytes |

Hinweis Für den TSEND_C und TRCV_C müssen Sie unterschiedliche Verbindungsnummern verwenden, da sonst zwei Verbindungen mit derselben Verbindungsnummer aufgebaut werden würden.

In diesem Programmbeispiel wird für den remoten und lokalen Verbindungspartner derselbe TSelector verwendet.

2.6 Verbindungsparameter nach TCON_IP_V4

Um die Kommunikationsverbindung bei TCP oder UDP zu parametrieren wird für die CPUs der S7-1200 ab V4.0 und S7-1500 ein Verbindungsbeschreibungs-DB mit einer Struktur nach TCON_IP_V4 verwendet. Die feste Datenstruktur des TCON_IP_V4 enthält die notwendigen Parameter, die zum Aufbau der Verbindung benötigt werden.

Die folgende Tabelle zeigt den Aufbau der Verbindungsbeschreibung nach TCON_IP_V4.

Tabelle 2-6

| Byte | Parameter | Datentyp | Wert | Beschreibung |
|-----------|-------------------|----------------------|-------|--|
| 0 und 1 | Interfaceld | HW_ANY | 64 | Hardware-Kennung der integrierten PROFINET-Schnittstelle der CPU |
| 2 und 3 | ID | CONN_OUC | 1 | Referenz auf diese Verbindung (Wertebereich: 1 bis 4095). Den Wert dieses Parameters müssen Sie für die Anweisung TSEND_C und TRCV unter ID angeben. |
| 4 | ConnectionType | BYTE | 16#0B | Verbindungstyp: <ul style="list-style-type: none"> • 11: TCP (16#0B hex = 11 dez) • 19: UDP (16#13 hex = 19 dez) |
| 5 | ActiveEstablished | BOOL | False | Kennung für die Art des Verbindungsaufbaus <ul style="list-style-type: none"> • False: passiver Verbindungsaufbau • True: aktiver Verbindungsaufbau |
| 6 bis 9 | RemoteAddress | ARRAY [1..4] of BYTE | - | IP-Adresse des Partner-Endpunkts, z. B. für 192.168.0.3 <ul style="list-style-type: none"> • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 3 |
| 10 und 11 | RemotePort | UINT | 2000 | Port-Adresse des entfernten Verbindungspartners (Wertebereich: 1 bis 49151) |
| 12 und 13 | LocalPort | UINT | 2000 | Port-Adresse des lokalen Verbindungspartners (Wertebereich: 1 bis 49151) |

Hinweis Für den TSEND_C und TRCV_C müssen unterschiedliche Verbindungs-IDs verwendet, da sonst zwei Verbindungen mit derselben Verbindungs-ID aufgebaut werden würden.

In diesem Programmbeispiel wird für den remoten und lokalen Verbindungspartner dieselbe Port-Adresse verwendet.