TSEND_C: Establishing a connection and sending data

Validity

The following description of the "TSEND_C" instruction is valid for:

- Ethernet
  - CPU S7-1200 with firmware version ≥ V4.0 and CPU S7-1500
  - CPU S7-1500 firmware version V2.1 and higher: UDP multicast communication
  - CPU S7-1500 as of firmware version V2.0 and TSEND_C as of instruction version V3.2: Secure communication
- PROFIBUS

  FDL connections of the S7-1500 with CM 1542-5 as of V2.0 with the system data type TCON_FDL

Description

The "TSEND_C" instruction sets up and establishes a communication connection. Once the connection has been set up and established, it is automatically maintained and monitored by the CPU.

The instruction is executed asynchronously and has the following functions:

- Setting up and establishing a communication connection
- Sending data via an existing communication connection
- Terminating or resetting the communication connection

Internally, the instruction "TSEND_C" uses the communication instructions "TCON", "TSEND", "T_DIAG", "T_RESET" and "TDISCON".

> **Note**
>
> Support when programming connections
>
> If you select an instruction for communication TCON, TSEND_C or TRCV_C in a program block and wish to create and assign parameters to connections of the type TCP, UDP, ISO-on-TCP or FDL, you can use the support of the connection parameter assignment. UDP multicast connections via the integrated PROFINET interfaces are possible for S7-1500-CPUs as of firmware version V2.1 and higher.
>
> You can find the connection parameter assignment in the Inspector window of the program editor.

Setting up and establishing a communication connection

The communication connection is set up and established with CONT=1. For information on the number of possible communication connections, refer to the technical specifications for your CPU. The connection description specified at the CONNECT parameter is used to set up the communication connection. The following connection types can be used:

- Programmed connections (structure of connection via "TCON"):
  - TCP / UDP: Connection description via the TCON_IP_v4 system data type
  - TCP / UDP with secure communication: Connection description via the system data types TCON_IP_V4_SEC or TCON_QDN_SEC
  - ISO-on-TCP: Connection description via the TCON_IP_RFC system data type

  - ISO: Connection description via the TCON_ISOnative system data type (for CP1543-1 only)
  - Telecontrol connections to SMS clients: Connection description via the TCON_PHONE system data type

    For this connection type, the station requires access to the mobile network via a mobile network CP.
  - FDL connections of the S7-1500 with CM 1542-5 as of V2.0 with the system data type TCON_FDL
- Configured connections
  - Specify an existing connection in the TCON_Configured system data type.

An existing connection is terminated and the connection which has been set up is removed when the CPU goes into STOP mode. To set up and establish the connection again, you must execute "TSEND_C" again.

Sending data via an existing communication connection

The send job is executed when a rising edge is detected at the REQ parameter. As described above, the communication connection is established first.

You specify the send area with the DATA parameter. This includes the address and the length of the data to be sent. Do not use a data area with the data type BOOL or Array of BOOL at the DATA parameter. With the LEN parameter, you specify the maximum number of bytes sent with a send job. If you use a send area with optimized access at the DATA parameter, the LEN parameter must have the value "0".

The data to be sent must not be edited until the send job is completed.

Terminating and resetting the communication connection

The communication connection is terminated when the CONT parameter is set to "0" even if an ongoing data transfer is not complete yet. This does not apply if you are using a configured connection for "TSEND_C".

The connection can be reset at any time by setting the parameter COM_RST to "1". This terminates the existing communication connection and a new connection is established. If data is being transferred at this time, this can lead to data loss.

Parameters

The following table shows the parameters of the "TSEND_C" instruction:

Parameter	Declaration	Data type	Memory area	Description
REQ	Input	BOOL	I, Q, M, D, L, T, C or constant	Starts the send job on a rising edge.
CONT	Input	BOOL	I, Q, M, D, L	Controls the communication connection: • 0: Disconnect the communication connection. • 1: Establish and maintain the communication connection.
LEN	Input	UDINT	I, Q, M, D, L or constant	Optional parameter (hidden) Maximum number of bytes to be sent with the job. If you use a send

				area with optimized access at the DATA parameter, the value "0" must be used at the LEN parameter. For FDL connections of the CM 1542-5, the maximum length is 240 bytes. In this regard, note the maximum lengths that can be processed by the connection partner.
CONNECT	InOut	VARIANT	D	Pointer to the structure of the connection description: • Programmed connection: ○ For TCP or UDP, use the TCON_IP_v4 system data type. For a description, refer to**Connection parameters with structure according to TCON_IP_v4**. ○ For TCP or UDP with secure communication, use the structure TCON_IP_V4_SEC or TCON_QDN_SEC. For a description, refer to: **Connection parameters in accordance with TCON_IP_V4_SEC** or **connection parameters in accordance with TCON_QDN_SEC** ○ For ISO-on-TCP, use the TCON_IP_RFC system data type. For a description, refer to **Connection parameters with structure according to TCON_IP_RFC**. ○ For ISO, use the TCON_ISO-native system data type (for CP 1543-1 only). For description, refer to instruction "**TCON**". ○ For connections to SMS clients, use the TCON_PHONE system data type. For a description, refer to **Connection parameters to TCON_Phone**. ○ For FDL connections of the CM 1542-5, use the system data type TCON_FDL; see **Connection parameters to TCON_FDL**. • Configured connection: ○ For existing connections, use the TCON_Configured system data type. For a description, see "System data type

				for configured connections" below.
DATA	InOut	VARIANT	I, Q, M, D, L	Pointer to the send area containing the address and the length of the data to be sent. When transferring structures, the structures must be identical at the sending and receiving end.
ADDR	InOut	VARIANT	D	Hidden parameter that needs to be used, however, with UDP. In this case it contains a pointer to the system data type TADDR_Param. Store the address information of the recipient (IP address and port number) in a data block with the system data type TADDR_Param. See also: **Structure of the address information of the remote partner with UDP**
COM_RST	InOut	BOOL	I, Q, M, D, L	Optional parameter (hidden) Resets the connection: • 0: Irrelevant • 1: The existing connection is reset. The COM_RST parameter is reset after evaluation by the "TSEND_C" instruction and should not, therefore, be interconnected statically.
DONE	Output	BOOL	I, Q, M, D, L	Status parameter with the following values: • 0: Send job not yet started or still in progress. • 1: Send job executed without error. This state is only displayed for one cycle. The output parameter DONE is set if an intermediate step was completed successfully during processing (connection establishment, sending, connection termination) and if the execution of "TSEND_C" was completed successfully.
BUSY	Output	BOOL	I, Q, M, D, L	Status parameter with the following values: • 0: Send job not yet started or already completed. • 1: Send job not yet completed. A new send job cannot be started.
ERROR	Output	BOOL	I, Q, M, D, L	Status parameter with the following values:

				• 0: No error • 1: Error occurred during connection establishment, data transfer or connection termination. The output parameter ERROR can be set due to an error in the "TSEND_C" instruction or the communication instructions used internally.
STATUS	Output	WORD	I, Q, M, D, L	Status of instruction (see the "ERROR and STATUS parameters" description).

You can find additional information on valid data types under "**Overview of the valid data types**".

REQ, CONT and COM_RST parameters

The parameter CONT controls the connection establishment of the "TSEND_C" instruction regardless of the REQ parameter. The behavior of the CONT parameter partially depends on whether a programmed or a configured connection is used:

- With CONT = "0": No data is sent (regardless of whether a programmed or a configured connection is used).
- When changing CONT = "0" to "1":
  - With a programmed connection, it is established with "TCON".
  - With a configured connection, it is checked with "T_DIAG".
- With CONT = "1":
  - As long as no data is sent (REQ="0"), the connection is checked with "T_DIAG".
  - If the internally used communication instructions signal that no connection end point exists, the connection is automatically reestablished with "TCON".
- When changing CONT = "1" to "0":
  - With a programmed connection, it is terminated with "TDISCON".
  - With a configured connection, it is reset with "T_RESET".

The parameter COM_RST resets the connection when changing from "0" to "1":

- If a connection is established, it is reset with "T_RESET" (regardless of whether a programmed or configured connection is used).
- If no connection is established, the setting of the parameter has no effect.

The REQ and COM_RST parameters only have an effect if CONT has been set to "1". The following table shows the relationship between the REQ, CONT and COM_RST parameters:

REQ	CONT	COM_RST	Status of the instruction	Description
Irrelevant	0	Irrelevant	Not yet executed	No job active (STATUS = 7000).
Irrelevant	0	Irrelevant	Initialization	Connection is being terminated. The instruction is being reset.
Irrelevant	0 > 1	Irrelevant	Connection establishment	Connection is being established. Data is not being transferred yet.

0	1	0	Connection established	The connection is established and is monitored with the instruction "T_DIAG".
Irrelevant	1	0 > 1	Connection established	The connection is interrupted by "T_RESET" briefly and reset.
0 > 1	1	0	Connection established	Instruction starts sending.
Irrelevant	1	0 > 1	Data is being sent	Data transfer is interrupted. The connection is being reset.

System data type for configured connections

For configured connections at the CONNECT parameter, use the following structure for connection description to TCON_Configured:

Byte	Parameter	Data type	Start value	Description
0 … 1	InterfaceID	HW_ANY	-	Hardware identifier of the local interface (value range: 0 to 65535).
2 … 3	ID	CONN_OUC	-	Reference to the connection (value range: 1 to 4095). Enter the connection ID of the existing connection.
4	Connection-Type	BYTE	-	Connection type Select 254 (decimal) for a configured connection.

Note

For reasons of compatibility, the parameters InterfaceID and ConnectionType are part of the structure for the connection description to TCON_Configured. These parameters do not have an effect on the connection parameter assignment; only the parameter ID for the connection ID is evaluated in the connection parameter assignment.

BUSY, DONE and ERROR parameters

You can check the status of the job with the BUSY, DONE, ERROR and STATUS parameters. The BUSY parameter indicates the processing status. With the DONE parameter, you can check whether or not a send job executed successfully. The ERROR parameter is set if errors occur during execution of "TSEND_C". The error information is output at the STATUS parameter.

The following table shows the relationship between the BUSY, DONE and ERROR parameters:

DONE	BUSY	ERROR	Description
0	0	0	The instruction has not been executed yet (no rising edge at REQ parameter).
0	1	0	The instruction is being executed and calls the internally used communication instructions.

1	0	0	The send job was completed successfully. "0000" is output at the STATUS parameter. DONE = "1" is only displayed for one cycle.
0	0	1	The execution of the instruction or an intermediate step during processing was terminated with an error. If there is a subsequent error due to an internally used communication instruction, the error that occurred first during processing is displayed. This state is only displayed for one cycle.

ERROR and STATUS parameters

ERROR	**STATUS*** **(W#16#...)**	**Description**
0	0000	Send job was executed without error.
0	0001	Communication connection established.
0	0003	Communication connection closed.
0	7000	No active send job execution; no communication connection established.
0	7001	Initial call for establishing a connection.
0	7002	Second call for establishing a connection
0	7003	Communication connection is being terminated.
0	7004	Communication connection has been established and is being monitored. No send job execution active.
0	7005	Data transfer in progress.
1	80A1	• Connection or port already being used by user. • Communication error: ○ The specified connection has not yet been established. ○ The specified connection is being terminated. Transfer via this connection is not possible. ○ The interface is being re-initialized.
1	80A3	The nested "T_DIAG" instruction has reported that the connection has closed.
1	80A4	IP address of the remote endpoint of the connection is invalid or it matches the IP address of the local partner.
1	80A7	Communication error: You called the instruction with COM_RST = 1 before the send job was complete.
1	80AA	A connection is currently being established with the same connection ID by another block. Repeat the job with a new rising edge at the REQ parameter.
1	80B3	• When using the protocol variant UDP the ADDR parameter does not contain any data. • Error in the connection description • The local port is already being used in a different connection description.
1	80B4	You have violated one or both of the following conditions for passive connection establishment (active_est = FALSE) when using the ISO-on-TCP protocol variant (connection_type = B#16#12):

		• local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
1	80B5	Only passive connection establishment is permitted for connection type 13 = UDP.
1	80B6	Parameter assignment error in the connection_type parameter of the data block for connection description.
1	80B7	• For system data type TCON_Param: Error in one of the following parameters of the data block for connection description: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len. • For system data types TCON_IP_V4 and TCON_IP_RFC: IP address of the partner end point was set to 0.0.0.0.
1	8085	The LEN parameter is larger than the highest permitted value.
1	8086	The ID parameter within the CONNECT parameter is outside the permitted range.
1	8087	Maximum number of connections reached; no additional connection possible.
1	8088	The value at the LEN parameter does not correspond to the receive area set at the DATA parameter.
1	8089	• The CONNECT parameter does not point to a data block. • The CONNECT parameter does not point to a connection description. • The manually created connection description has an incorrect structure for the selected connection type.
1	8091	Maximum nesting depth exceeded.
1	809A	The CONNECT parameter points to a field that does not correspond to the length of the connection description.
1	809B	InterfaceID is invalid: • It does not point to a local CPU interface or a CP. • If you are using the connection parameter assignment, it cannot have the value 0. • It must not have the value 0 in the used TCON_xxx structure. See **TCON: Establish communication connection**
1	80C3	• All connection resources are in use. • A block with this ID is already being processed in a different priority group.
1	80C4	Temporary communication error: • The connection cannot be established at this time. • The connection cannot be established because the firewalls on the connection path are not open for the required ports. • The interface is receiving new parameters or the connection is being established. • The configured connection is currently being removed by a "**TDISCON**" instruction. • The connection used is being terminated by a call with COM_RST = 1.

		• Temporarily no receive resources available at the connection partner. The connection partner is not ready to receive.
1	80C5	• Connection terminated by the communication partner. • LSAP of the remote connection partner is not released
1	80C6	Network error: • Remote partner cannot be reached. • Physical interruption on PROFIBUS
1	8722	Parameter CONNECT: The source area is invalid. The area does not exist in the DB.
1	873A	Parameter CONNECT: Access to the connection description is not possible (for example, because the DB is not available).
1	877F	Parameter CONNECT: Internal error.
1	8822	Parameter DATA: Invalid source area, the area does not exist in the DB.
1	8824	Parameter DATA: Area error in the VARIANT pointer.
1	8832	Parameter DATA: The DB number is too high.
1	883A	Parameter DATA: No access to the data area, for example because the data block does not exist
1	887F	Parameter DATA: Internal error, e.g. invalid VARIANT reference.
1	893A	Parameter ADDR: Access to send area not possible (e.g. because the DB does not exist).
* The error codes can be displayed as integer or hexadecimal values in the program editor. For information on switching the display formats, refer to "See also".		

Note

Error messages of the instructions "TCON", "TSEND", "T_DIAG", "T_RESET" and "TDISCON"

Internally, the "TSEND_C" instruction uses the instructions "**TCON**", "**TSEND**", "**T_DIAG**", "**T_RESET**" and "**TDISCON**". The error messages of these instructions can also be output at the STATUS parameter. The meaning of the error codes is described in the corresponding instructions. In the event of identical error codes for internally used instructions with different meanings, the instance data block of "TSEND_C" can be used to determine which instruction output the error.

Example

You can find the example here: **Program example for send functions**.

You can find additional information and the program code for the example here: **Sample Library for Instructions**.

See also

Basics of Open User Communication (S7-1200, S7-1500)
Difference between synchronous and asynchronous instructions (S7-1200, S7-1500)
Overview of connection configuration (S7-1200, S7-1500)
Starting connection parameter assignment (S7-1200, S7-1500)
Creating and assigning parameters to connections (S7-1200, S7-1500)