

SIEMENS

Ingenuity for life



Examples of Open User Communication: UDP

Programmed UDP connection

<https://support.industry.siemens.com/cs/ww/de/view/109747710>

Siemens
Industry
Online
Support



Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (<https://support.industry.siemens.com>) shall also apply.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place. For additional information on industrial security measures that may be implemented, please visit <https://www.siemens.com/industrialsecurity>.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: <https://www.siemens.com/industrialsecurity>.

Table of contents

Legal information	2
1 Introduction	4
1.1 Overview.....	4
1.2 Principle of operation.....	6
1.3 Components used	10
2 Engineering	11
2.1 Interface description for S7-1200 and S7-1500	11
2.2 Interface description for S7-300 and S7-400	14
2.3 Integration into the user project.....	17
2.3.1 Open library in STEP 7 (TIA Portal)	17
2.3.2 Integrating the "LOpenUserComm_Udp" Function Module into the User Program	20
2.4 Determine hardware identification of the CPU or CP/CM interface.....	22
2.5 Error handling S7-1200 and S7-1500	23
2.5.1 Error when establishing connection	23
2.5.2 Error when receiving data	29
2.5.3 Error sending data	33
2.5.4 Error when disconnecting.....	36
2.6 Error handling S7-300 and S7-400	38
2.6.1 Error when establishing connection	38
2.6.2 Error when receiving data	44
2.6.3 Error sending data.....	47
2.6.4 Error when disconnecting.....	50
2.7 Explanations on the state machine	52
2.7.1 STATE_IDLE.....	52
2.7.2 STATE_CONNECT	52
2.7.3 STATE_CONNECTED.....	53
2.7.4 STATE_RECEIVE	53
2.7.5 STATE_SEND.....	53
2.7.6 STATE_DISCONNECT	54
2.7.7 STATE_ERROR.....	55
3 Useful information	59
3.1 Basics	59
3.1.1 Basics of the UDP protocol	59
3.1.2 Structure of the parameter data set "TCON_IP_V4".....	59
3.1.3 Structure of the "TCON_PAR" parameter data set	60
3.1.4 Structure of the address information of the communication partner according to "TADDR_Param".....	62
3.1.5 Structure of the address information of the communication partner according to "TADDR_PAR".....	63
3.2 Details on functionality	64
3.2.1 Structure of the program	64
3.2.2 Explanation of the region "edgeDetection".....	64
3.2.3 Explanation of the region "initStates"	64
3.2.4 Explanation of the region "stateMachine"	65
3.2.5 Explanation of the region "callOUCblocks"	65
4 Appendix	66
4.1 Service and support	66
4.2 Links and Literature.....	67
4.3 Change documentation	67

1 Introduction

1.1 Overview

Content of the application example

TCP/IP-based Open User Communication (OUC) has become the standard for communication with SIMATIC S7 CPUs.

In the S7 CPU, the OUC is implemented on the basis of statements (for example, TCON, TUSEND, TURCV, and TDISCON). The user must configure the instructions in his user program and call them in a fault-tolerant way. Every user has to rethink this task again and again. To facilitate this, we offer you a function block (FB) in SCL. The FB calls the OUC instructions in the order and manner recommended in the manuals. In addition, the FB already contains the following mechanisms

- Connection management with the instructions "TCON" and "TDISCON".
- Send data to a partner CPU
- Receive data from a partner CPU

You can use the FB as a template for your own communication projects.

The application example provides the following information:

- Library for STEP 7 (TIA Portal) containing the FB
- Description of the FB to communicate with **UDP** via a **programmed connection**.

The application example shows where you can integrate your individual extensions in the code.

Overview of all OUC variants of this series

This application example is part of a larger series of basic examples for PLC communication.

[Table 1-1](#) shows the variants of the OUC that are made available to the user in the series of basic examples. The variant described in this application example is color-coded.

Table 1-1

Protocol	Telegram length	Programmed connections	Note
ISO-on-TCP	Dynamic	✓	<ul style="list-style-type: none"> • Fast data transmission • Transmission of medium to large amounts of data You can find information on the maximum data volumes of the CPUs in the FAQ: 18909487. • Mainly used in SIMATIC homogeneous structures. • Acknowledged • Packet-oriented data transmission, i.e. length and information about beginning and end of a telegram are also transmitted.
TCP	Fixed	✓	<ul style="list-style-type: none"> • Fast data transmission • Transmission of medium to large amounts of data You can find information on the maximum data volumes of the CPUs in the FAQ. 18909487. • Flexible use with external systems. • Acknowledged • Data is transmitted as a data stream, i.e. length and information about beginning and end of a telegram are not transmitted. For this reason, when transmitting dynamic telegram lengths, the transmitter must define a message structure that can be interpreted by the receiver.
	Dynamic	✓	

Protocol	Telegram length	Programmed connections	Note
UDP	Dynamic	✓	<ul style="list-style-type: none"> Extremely fast data transmission Transmission of small to medium amounts of data. You can find information on the maximum data volumes of the CPUs in the FAQ. 18909487. Flexible use with external systems. There is no guarantee that a sent data packet will arrive at the recipient, or that a sent data packet will arrive at the recipient only once. The order in which the data packages arrive at the recipient cannot be predicted. Data packages with an incorrect checksum are discarded and not requested again. Packet-oriented data transmission, i.e. length and information about beginning and end of a telegram are also transmitted.

There is a separate FB for each variant of the OUC, which serves as a communication template. All communication templates are summarized in the library "LOpenUserComm". This application example only refers to the variant "Programmed connection with UDP".

1.2 Principle of operation

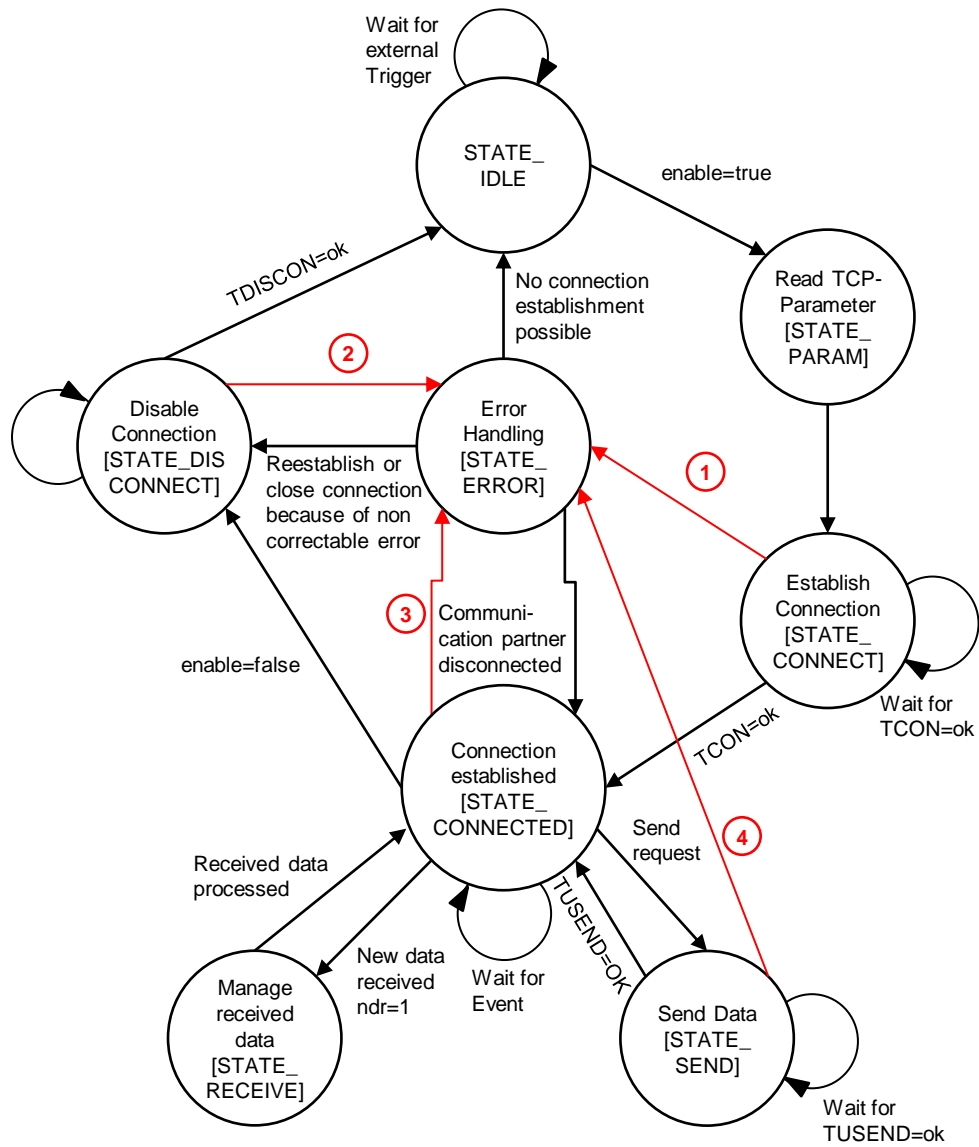
Realization as a state machine

The FB for controlling the OUC instructions (TCON, TUSEND, TURCV and TDISCON) is implemented as a state machine. The design model of a state machine is particularly suitable for modeling more complex asynchronous processes, such as communication between partners, which can extend over several cycles.

A certain state is run through cyclically until a transition condition is fulfilled and the machine switches to the next subsequent state. This not only improves the clarity compared to conventional link control, but also makes it easier to find any errors in the program logic more quickly.

A detailed description of the state machine can be found in section [2.7](#).

Figure 1-1



- ① Error on establishing connection
- ② Error on disconnect
- ③ Error on receiving data
- ④ Error on sending data

Description of the states

The following table provides an overview of the realized states and possible transitions. For more information, see the documented code.

Table 1-2

State	Description	Transition condition
STATE_IDLE (1)	In the idle state "STATE_IDLE" the FB has the following properties: <ul style="list-style-type: none"> No connection is active. Status variables are reset. 	The idle state "STATE_IDLE" is only left if a connection is initiated via a parameter (enable).
STATE_PARAM (2)	The UDP connection parameters are read in and assigned to the statement TCON.	The FB immediately changes to the state "STATE_CONNECT" without any transition condition.
STATE_CONNECT (3)	The UDP connection to the partner is established.	The state "STATE_CONNECT" is left if one of the following conditions is fulfilled: <ul style="list-style-type: none"> If the connection has not yet been established after the watchdog timer (30 s) has expired, the FB changes to the "STATE_ERROR" state. When the connection is established, the FB changes to the state "STATE_CONNECTED".
STATE_CONNECTED (7)	In the state "STATE_CONNECTED" the FB executes the following actions: <ul style="list-style-type: none"> It waits for the send request to be triggered in order to send data via the connection. It monitors whether data has been received from the partner. He monitors the connection to the partner. 	The state "STATE_CONNECTED" is left if one of the following conditions is fulfilled: <ul style="list-style-type: none"> If an error occurs while receiving data, the FB changes to the state "STATE_ERROR". If the connection is to be actively terminated, the FB changes to the state "STATE_DISCONNECT". If data are to be sent, the FB changes to the state "STATE_SEND". When new data is received, the FB changes to the state "STATE_RECEIVE".

State	Description	Transition condition
STATE_SEND (5)	In the state "STATE_SEND" the FB executes the following actions: <ul style="list-style-type: none"> • It activates the parameters of the OUC instruction "TUSEND". • It waits until the OUC statement "TUSEND" has been completed successfully (DONE = 1) or with an error (ERROR = 1). 	The state "STATE_SEND" is exited if one of the following conditions is fulfilled: <ul style="list-style-type: none"> • When the transmission process has been successfully completed, the FB returns to the state "STATE_CONNECTED". • If an error occurs during transmission, the FB changes to the state "STATE_ERROR".
STATE_RECEIVE (6)	The state "STATE_RECEIVE" is used for individual processing of the received data. In this application example, the state is empty.	The FB immediately returns to the state "STATE_CONNECTED" without any transition condition.
STATE_DISCONNECT (4)	In the following cases, the FB closes the connection to the partner in a controlled manner in the "STATE_DISCONNECT" state: <ul style="list-style-type: none"> • The user triggers the disconnection (enable). • The disconnection is initiated by the FB if the connection has to be re-established or an unrecoverable error has occurred. 	The state "STATE_DISCONNECT" is left if one of the following conditions is fulfilled: <ul style="list-style-type: none"> • If the connection has not been terminated after the watchdog timer (30 s) has expired, the FB changes to the "STATE_IDLE" state. • If the connection has been terminated without errors, the FB switches to the idle state "STATE_IDLE". • If an error occurs during disconnection, the FB changes to the state "STATE_ERROR".
STATE_ERROR (8)	In the state "STATE_ERROR" the FB executes the following actions: <ul style="list-style-type: none"> • It decides whether an attempt is made to remedy an error within the FB independently by calling other states. • It provides the output parameters with the error information. 	The state "STATE_ERROR" is left if one of the following conditions is fulfilled: <ul style="list-style-type: none"> • If the connection has to be re-established or an unrecoverable error has occurred, the FB changes to the state "STATE_DISCONNECT". • If the connection is not established, the FB switches to the idle state "STATE_IDLE". • When the partner has terminated the connection, the FB changes to the state "STATE_CONNECTED".

1.3 Components used

The following hardware and software components were used to create this application example:

Table 1-3

Components	Quantity	Article number	Note
CPU 1513-1 PN	1	6ES7513-1AL01-0AB0	Alternatively, you can use any S7-1500 CPU from firmware V2.0, ET 200SP CPU, ET 200pro CPU or the following CPs and CMs: <ul style="list-style-type: none"> • CP 1543-1 (Item number: 6GK7543-1AX00-0XE0) • CM 1542-1 (Item number: 6GK7542-6VX00-0XE0) • CP 1542SP-1 (Item number: 6GK7542-6UX00-0XE0) • CP 1542SP-1 IRC Article number: 6GK7542-6VX00-0XE0) • CP 1543SP-1 (Item number: 6GK7543-6WX00-0XE0)
CPU 1214C DC/DC/DC	1	6ES7214-1AG40-0AB0	Alternatively, you can use any S7-1200 CPU with firmware V4.0 or higher.
STEP 7 V15 Update 4	1	Package: 6ES7822-1AA05-0YA5 Download: 6ES7822-1AE05-0YA5	

2 Engineering

2.1 Interface description for S7-1200 and S7-1500

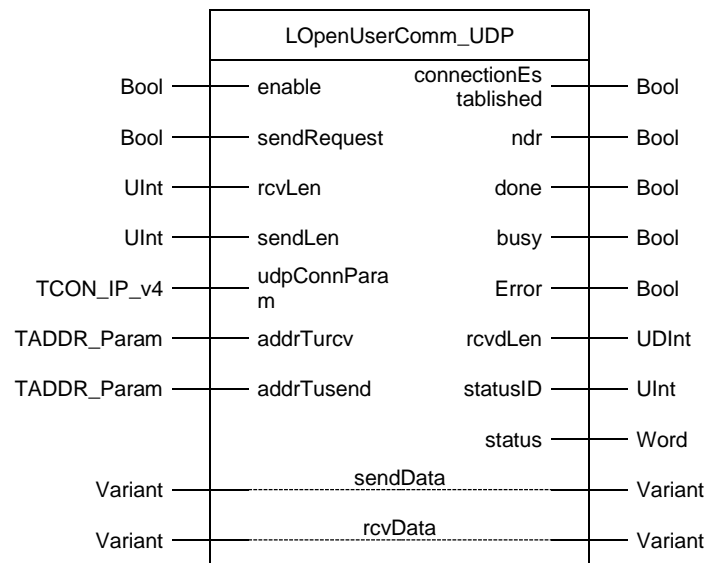
Functional description

The FB "LOpenUserComm_Udp" implements a complete UDP communication relationship to a partner. It encapsulates all OUC instructions in a user-friendly shell module to perform the following functions:

- Management of connection establishment and disconnection via the "enable" input.
- Send user data at input "sendData" with the length "sendLen" to the partner as soon as the input "sendRequest" detects a positive edge.
- Receiving data from a partner and saving it in the memory area created at parameter "rcvData".
- Output of the status of the transmission and connection at the output parameter "status".

Block interface

Figure 2-1



The following table shows the inputs and outputs of the function block "LOpenUserComm_Udp" for S7-1200 and S7-1500.

Table 2-1

Name	P type	Data type	Comment
Enable	IN	Bool	Release signal for connection establishment and data exchange
sendRequest	IN	Bool	Triggering the send request
rcvLen	IN	UInt	Length of the receive area in bytes: Value range: 0 (recommended) or 1 to 1472 (from firmware version V2.5 of the S7-1500 CPUs with Unicast or Multicast: 1 to 2048)
sendLen	IN	UInt	Number of bytes that are to be sent with the job. Value range: 1 to 1472 (from firmware version V2.5 of the S7-1500 CPUs with Unicast or Multicast: 1 to 2048)
udpConnParam	IN	TCON_IP_v4	Connection parameters Detailed information can be found in the section 3.1.2_
addrTurcv	IN	TADDR_Param	Address information of the communication partner for the "TURCV" instruction Detailed information can be found in the section 3.1.4_
addrTusend	IN	TADDR_Param	Address information of the communication partner for the "TSEND" instruction Detailed information can be found in the section 3.1.4_
connectionEstablished	OUT	Bool	Status display: Connection setup executed Note With UDP, there is no feedback (acknowledgement) from the communication partner that the connection has actually been established.
Ndr	OUT	Bool	Status display: Receive new data Note With UDP, no acknowledgement is sent to the communication partner that the received data has arrived.
Done	OUT	Bool	Status display: Send job completed Note With UDP there is no feedback (acknowledgement) from the communication partner that the transmitted data was actually received.
Busy	OUT	Bool	FB in process
Error	OUT	Bool	Error display
rcvdLen	OUT	UDInt	Length of received data (in byte)
statusID	OUT	UInt	Parameter shows which OUC instruction supplies the error (see Table 2-61).
Status	OUT	Word	Status display of the OUC instructions

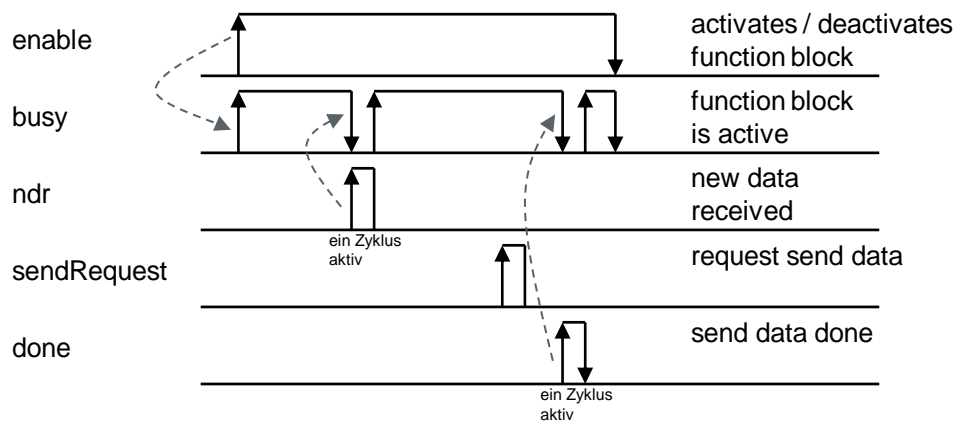
Name	P type	Data type	Comment
sendData	IN_OUT	Variant	Send data area (contains address and length) The address refers to: <ul style="list-style-type: none"> • A data block • A marker • The process image of inputs • The process image of outputs When structures are transmitted, the structures on the send and receive sides must be identical.
rcvData	IN_OUT	Variant	Receive data area The address refers to: <ul style="list-style-type: none"> • A data block • A marker • The process image of inputs • The process image of outputs When structures are transmitted, the structures on the send and receive sides must be identical.

Note When sending more than 1472 bytes, you must ensure that your receiver supports more than 1472 bytes. If this is not the case, you will not notice the failed reception on the transmitter side.

Function diagrams

The following function diagram shows how the most important output parameters are set depending on the input parameters.

Figure 2-2



2.2 Interface description for S7-300 and S7-400

Functional description

The FB "LOpenUserComm_Udp" implements a complete UDP communication relationship to a partner. It encapsulates all OUC instructions in a user-friendly shell module to perform the following functions:

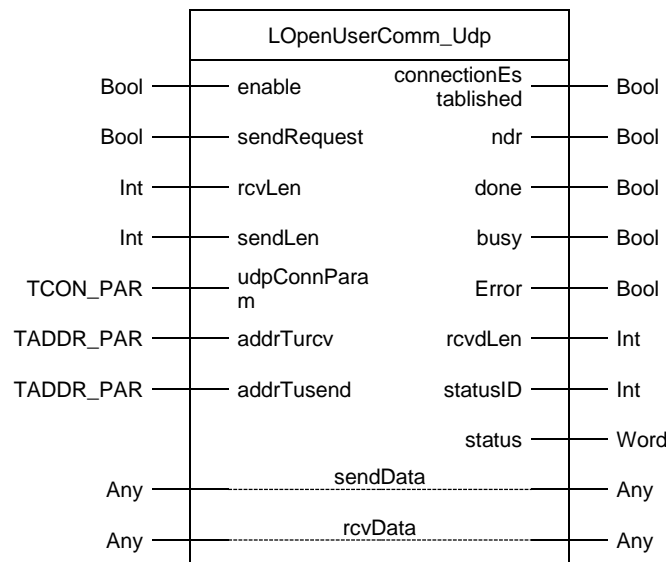
- Management of connection establishment and disconnection via the "enable" input.
- Send user data at input "sendData" with the length "sendLen" to the partner as soon as the input "sendRequest" detects a positive edge.
- Receiving data from a partner and storing it in the defined receive area. The length of the received data is defined by the parameter "rcvLen". The receive area is determined by the following two values:
 - Pointer to the start of the area
 - Length of the area
- Output of the status of the transmission and connection at the output parameter "status".

Note

The length of the receive area must be at least as large as the length of the receive data.

Block interface

Figure 2-3



The following table shows the inputs and outputs of the function block "LOpenUserComm_Udp" for S7-300 and S7-400.

Table 2-2

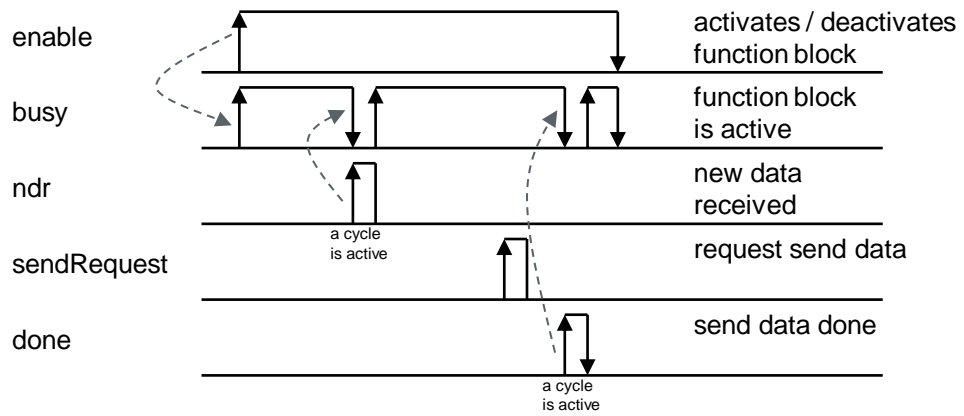
Name	P type	Data type	Comment
Enable	IN	Bool	Release signal for connection establishment and data exchange
sendRequest	IN	Bool	Triggering the send request
rcvLen	IN	Int	Length of receiving data (in bytes): 1 to 1472
sendLen	IN	Int	Maximum number of bytes sent with the job
udpConnParam	IN	TCON_PAR	Connection parameters Detailed information can be found in the section 3.1.3.
addrTurcv	IN	TADDR_PAR	Address information of the communication partner for the "TURCV" instruction Detailed information can be found in the section 3.1.5.
addrTusend	IN	TADDR_PAR	Address information of the communication partner for the "TSEND" instruction Detailed information can be found in the section 3.1.5.
connectionEstablished	OUT	Bool	Status display: Connection setup executed Note With UDP, there is no feedback (acknowledgement) from the communication partner that the connection has actually been established.
Ndr	OUT	Bool	Status display: Receive new data Note With UDP, no acknowledgement is sent to the communication partner that the received data has arrived.
Done	OUT	Bool	Status display Send job completed Note With UDP there is no feedback (acknowledgement) from the communication partner that the transmitted data was actually received.
Busy	OUT	Bool	FB in process
Error	OUT	Bool	Error display
rcvdLen	OUT	Int	Length in bytes of the previously received data
statusID	OUT	Int	Parameter indicates which OUC instruction returns the error.
Status	OUT	Word	Status display of the OUC instruction
sendData	IN_OUT	Any	Send data area (contains address and length) The address refers to: <ul style="list-style-type: none"> • A data block • A marker • The process image of inputs • The process image of outputs

Name	P type	Data type	Comment
rcvData	IN_OUT	Any	Receive data area (contains address and length) The address refers to: <ul style="list-style-type: none"> • A data block • A marker • The process image of inputs • The process image of outputs

Function diagrams

The following function diagram shows how the most important output parameters are set depending on the input parameters.

Figure 2-4



2.3 Integration into the user project

2.3.1 Open library in STEP 7 (TIA Portal)

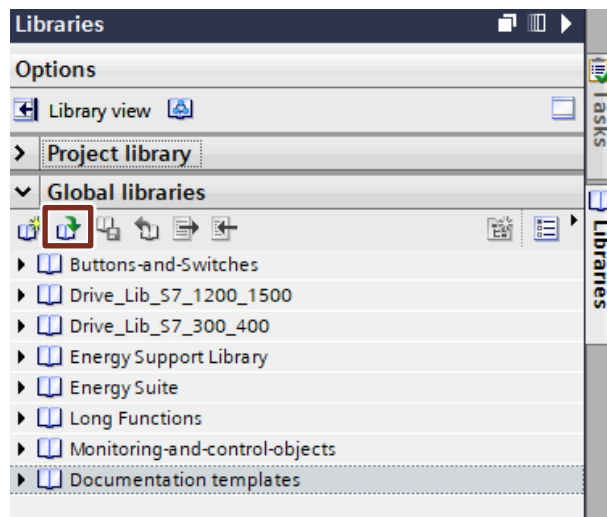
Follow the instructions below to open the "LOpenUserComm" library in STEP 7 (TIA Portal).

Requirement

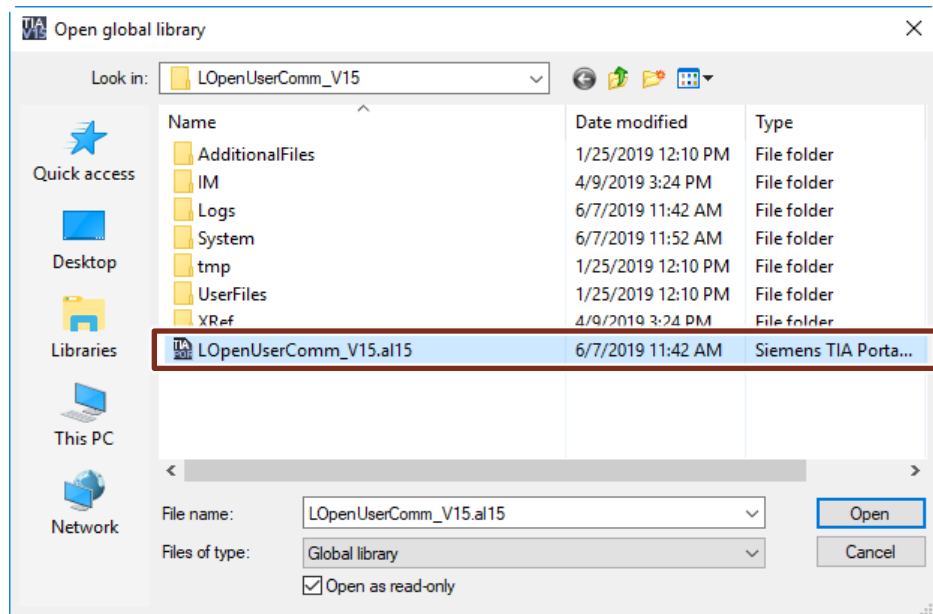
- STEP 7 (TIA Portal) is open.
- You have created a new project in STEP 7 (TIA Portal) or have opened an existing project.

Instructions

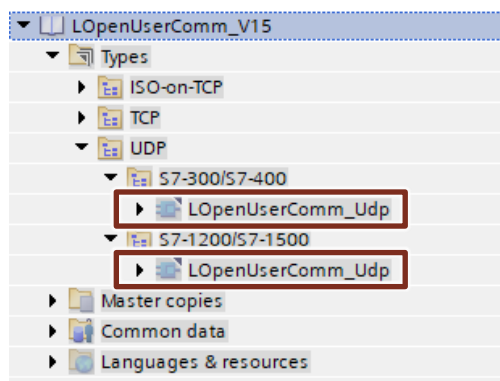
1. Open the "Libraries" task card.
2. In the "Global libraries" palette, click the "Open global library" button. The "Open global library" dialog is opened.



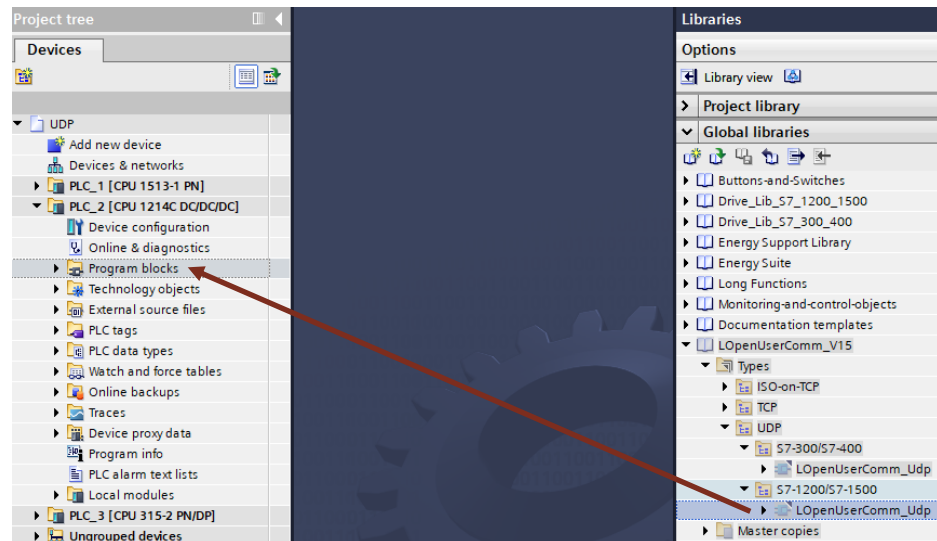
3. Select the library "LOpenUserComm.al15" and click on the button "Open". The library "LOpenUserComm" is opened in the palette "Global libraries".



4. The function blocks "LOpenUserComm_Udp" can be found in the library "LOpenUserComm" under "Types > UDP > S7-300/S7-400" ("Types > UDP > S7-300/S7-400") and under "Types > UDP > S7-1200/S7-1500" ("Types > UDP > S7-1200/S7-1500").



5. Insert the function block "LOpenUserComm_Udp" via drag & drop from the library into the folder "Program blocks" of your CPU.



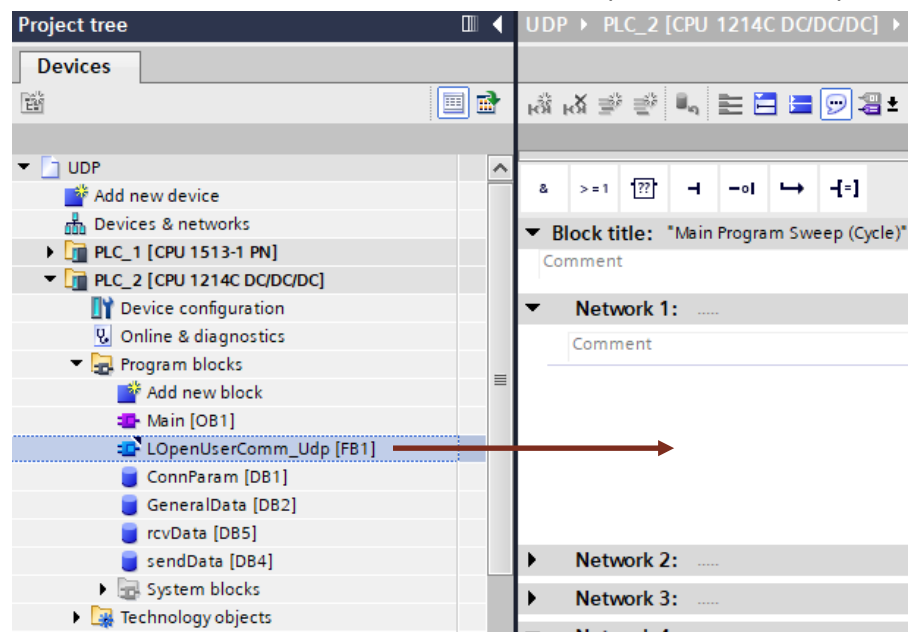
6. The function block "LOpenUserComm_Udp" is displayed in the folder "Program blocks" of your CPU.
7. Integrate the function block "LOpenUserComm_Udp" into the user program of your CPU to establish a communication connection to the partner and to send and receive data. (see section [2.3.2](#)).

2.3.2 Integrating the "LOpenUserComm_Udp" Function Module into the User Program

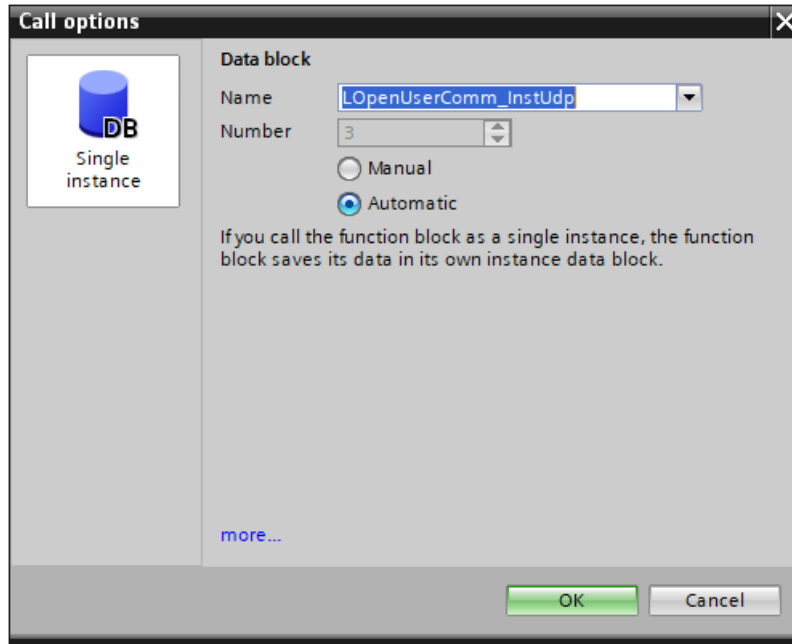
Follow the instructions below to integrate the function block "LOpenUserComm_Udp" into the user program of your CPU.

Call the function block "LOpenUserComm_Udp" e. g. cyclically in OB1.

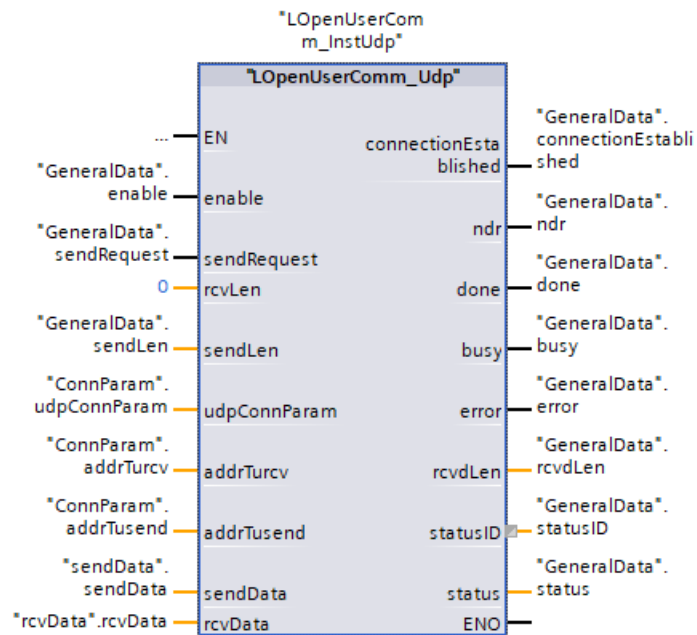
1. Double-click in the project navigation in the folder "Program blocks" of your CPU on the block "Main [OB1]". OB 1 is opened in the work area.
2. Mark the function block "LOpenUserComm_Udp" in the project navigation in the folder "Program blocks" of your CPU and insert it via drag & drop into a network of OB 1. The "Call options" dialog is opened automatically to create the instance data block of the function module "LOpenUserComm_Udp".



3. Enter the name of the instance DB, for example "LOpenUserComm_Udp". Select the "Automatic" option so that the instance DB number is automatically assigned by STEP 7 (TIA Portal).



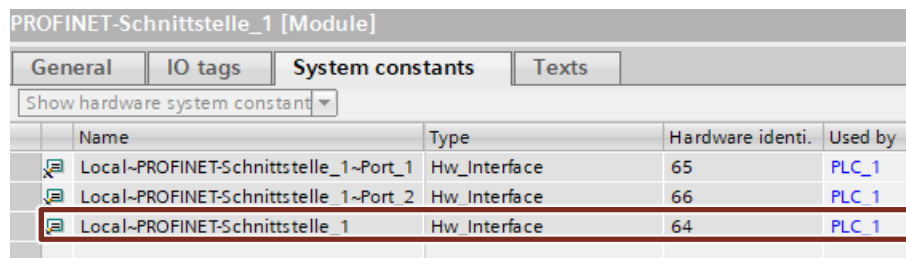
4. Assign the corresponding variables to the inputs and outputs of the function block "LOpenUserComm_Udp".






2.4 Determine hardware identification of the CPU or CP/CM interface

In the parameter data set "TCON_IP_V4" it is necessary to enter the correct hardware identification of the interface used. Follow the instructions below to determine the hardware identification of the interface:

1. In the network view or device view, select the CPU or CP/CM interface whose hardware ID you want to determine.
2. The properties of the CPU or CP/CM interface are displayed in the inspector window.
3. Open the "System constants" tab to display the hardware identification of the interface.



PROFINET-Schnittstelle_1 [Module]				
General		IO tags	System constants	Texts
Show hardware system constant ▾				
	Name	Type	Hardware identi.	Used by
	Local-PROFINET-Schnittstelle_1-Port_1	Hw_Interface	65	PLC_1
	Local-PROFINET-Schnittstelle_1-Port_2	Hw_Interface	66	PLC_1
	Local-PROFINET-Schnittstelle_1	Hw_Interface	64	PLC_1

2.5 Error handling S7-1200 and S7-1500

In the FB "LOpenUserComm_Udp" some error states are intercepted exemplarily and reactions to them are programmed. However, you can also catch all error states that the OUC statements supply here and implement your own reactions according to this schema, depending on your requirements.

Note

Since in the event of an error, the values at the output parameters are only output for one cycle, the values of "status" and "statusId" must be stored if error = 1.

2.5.1 Error when establishing connection

Job to establish connection with "TCON" could not be executed successfully (watchdog timer expired)

The watchdog timer is started as soon as a rising edge is detected at the "enable" input of the FB "LOpenUserComm_Udp".

If the job to establish the connection was successfully executed with "TCON" (DONE = 1), the watchdog timer is reset, but there is no acknowledgement from the communication partner that the communication connection has really been established successfully.

If the job to establish the connection with "TCON" has not been successfully completed after 30 s, the watchdog timer runs out and the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-3

Output parameters	Value	Description	Remedy
status	16#8102	Connection could not be established (watchdog timer (30 s) expired).	Check whether an error occurred when establishing a connection with "TCON" (ERROR = 1): Evaluate the output parameters "statusID" and "status" of the FB "LOpenUserComm_Udp" and, if necessary, correct the error when establishing the connection with "TCON". Initiate the connection setup again via the parameter enable = 1.
statusId	1	Internal error in function module	
Error	1	Error display: 1: Error identified	

Identifier for the type of connection not permitted

With UDP, only a passive connection setup is permitted. If you use the value "TRUE" for "ActiveEstablished" in the parameter data set "TCON_IP_V4", the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-4

Output parameters	Value	Description	Remedy
status	16#80B5	Status display of the "TCON" instruction	Enter the value "FALSE" for "ActiveEstablished" in the parameter data set "TCON_IP_V4". Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Parameterization error in parameter "ConnectionType"

If there is a parameterization error in the parameter "ConnectionType" of the parameter data set "TCON_IP_V4", the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-5

Output parameters	Value	Description	Remedy
status	16#80B6	Status display of the "TCON" instruction	Enter the value 0x13 (hex) for "ConnectionType" in the parameter data set "TCON_IP_V4". Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Value of the local port is "0".

If the local port "0" is specified in the parameter data set "TCON_IP_V4", the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-6

Output parameters	Value	Description	Remedy
status	16#80B7	Status display of the "TCON" instruction	Check the local port in the parameter data set "TCON_IP_V4". The valid value range is 1 to 49151. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection or port are already occupied

Each connection is uniquely defined by the connection ID and port number. If you use the same port number for several connections, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-7

Output parameters	Value	Description	Remedy
status	16#80A1	Status display of the "TCON" instruction	Use a different port number for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Local or remote port used by the system

If you define a port number for the local or remote port in the parameter data set "TCON_IP_V4" that is used by the system, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-8

Output parameters	Value	Description	Remedy
status	16#80A2	Status display of the "TCON" instruction	The following ports are reserved by the system: 20, 21, 80, 102, 135, 161, 162, 443, 34962, 34963, 34964 and the range 49152 to 65535 For the local and remote ports, use a port number that is not reserved by the system. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection ID is already in use

Each connection is uniquely defined by the connection ID and port number.

1. If you use the same connection ID for several connections, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-9

Output parameters	Value	Description	Remedy
status	16#80C8	Status display of the "TCON" instruction	Use a different connection ID for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

2. If the connection ID is already used by an established connection that uses the same parameter record at the "udpConnParam" parameter, an attempt is made to establish an existing connection. The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-10

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TCON" instruction	Use a separate parameter data set with different connection ID and port number for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection ID is already used by a configured connection

Each connection is uniquely defined by the connection ID and port number. If the connection ID is already used by a configured connection, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-11

Output parameters	Value	Description	Remedy
status	16#8085	Status display of the "TCON" instruction	Use a different connection ID for each configured and programmed connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection ID is outside the permitted range

If the parameter "ID" of the parameter data set "TCON_IP_V4" is outside the permissible range, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-12

Output parameters	Value	Description	Remedy
status	16#8086	Status display of the "TCON" instruction	In the parameter data set "TCON_IP_V4", enter a value for "ID" that lies in the value range from 1 to 4095. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Hardware identification in the connection parameters is not correct

If the parameter "Interfaceld" of the parameter data set "TCON_IP_V4" references no hardware identification of a CPU or CM/CP interface or has the value "0", the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-13

Output parameters	Value	Description	Remedy
status	16#809B	Status display of the "TCON" instruction	In the parameter data set "TCON_IP_V4" enter the hardware identification of the local interface (value range 0 to 65535) under "Interfaceld". Detailed information on how to determine the hardware ID can be found in section 2.4. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Communication error: "TDISCON" was executed before "TCON" was finished

If the instruction "TCON" has finished the connection establishment with DONE = 1 or ERROR = 1, a job to terminate the connection may be initiated.

If the connection establishment is aborted prematurely by calling a "TDISCON", the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-14

Output parameters	Value	Description	Remedy
status	16#80A7	Status display of the "TCON" instruction	Since the connection setup was aborted prematurely by calling a "TDISCON", the connection is terminated by the FB. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Order to establish connection is initiated during disconnection

If the job to establish a connection is triggered while the connection is being terminated, an attempt is made to establish an existing connection. The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-15

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TCON" instruction	Since you are trying to re-establish an existing connection, the connection is terminated by the FB. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

2.5.2 Error when receiving data

Parameterized length of received data invalid

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-16](#) if the following conditions are given:

- The parameterized length of the receive data is greater than the largest permissible value (1472 bytes (from firmware version V2.5 of the S7-1500 CPUs with unicast or multicast: 2048 Bytes).
- The value of the parameter "rcvLen" or "rcvData" was changed after the first call.

Table 2-16

Output parameters	Value	Description	Remedy
status	16#8085	Status display of the "TURCV" instruction	Enter the length of the data to be received at the parameter "rcvLen". Specify the reception range at the "rcvData" parameter. Detailed information about the maximum amount of user data that can be transferred with UDP can be found in the following FAQ: 18909487 .
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Receive range wrong parameterized

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-17](#) if the following conditions are given:

- Receive area is too small
- Value at parameter "rcvLen" greater than the receive range specified at parameter "rcvData"

Table 2-17

Output parameters	Value	Description	Remedy
status	16#8088	Status display of the "TURCV" instruction	The value at parameter "rcvLen" must not be larger than the receive range specified at parameter "rcvData".
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Size of receive buffer exceeded

If the length of the receive area is less than the length of the data sent by the communication partner, the received data is longer than expected. The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-18

Output parameters	Value	Description	Remedy
status	16#80C9	Status display of the "TURCV" instruction	The receive range that you specify at parameter "rcvData" must be at least as large as the length of the data that the communication partner sends. The length that you specify in the parameter "rcvLen" must be at least as large as the length of the data that the communication partner sends.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-19](#) if one of the following communication errors occurs:

- The specified connection between user program and communication layer of the operating system has not yet been established.
- The specified connection between the user program and the communication layer of the operating system is being disconnected. A receive job over this connection is not possible.
- The interface is being reconfigured.

Table 2-19

Output parameters	Value	Description	Remedy
status	16#80A1	Status display of the "TURCV" instruction	The connection must be established to receive data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Temporary communication errors

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-20](#) if a temporary communication error occurs:

- The interface is being reconfigured.

Table 2-20

Output parameters	Value	Description	Remedy
status	16#80C4	Status display of the "TURCV" instruction	The connection must be established to receive data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

IP address of remote connection endpoint in parameter "addrTurcv" invalid

If the IP address of the remote endpoint in parameter "addrTurcv" is invalid or matches the own IP address, the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-21

Output parameters	Value	Description	Remedy
status	16#80A4	Status display of the "TURCV" instruction	Check that you have defined the IP address of the remote connection endpoint in the "addrTurcv" parameter.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Invalid remote IP addresses are among others:

- Broadcast addresses, e.g. 192.168.0.255
- Network addresses, e.g. 192.168.0.0

Parameter "rcvData" was changed before the current job was finished.

If you change the parameter "rcvData" before the current receive job has been completed, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-22

Output parameters	Value	Description	Remedy
status	16#80B1	Status display of the "TURCV" instruction	Initiate the connection setup again via the parameter enable = 1.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

2.5.3 Error sending data

Parameterized length of transmission data invalid

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-23](#) if the following conditions are given:

- The parameterized length of the send data is greater than the largest permissible value.
- The parameter "sendLen" of the FB "LOpenUserComm_Udp" has the value "0".

Table 2-23

Output parameters	Value	Description	Remedy
status	16#8085	Status display of the "TUSEND" instruction	Specify the length of the data to be sent at the "sendLen" parameter. Detailed information about the maximum amount of user data that can be transferred with UDP can be found in the following FAQ: 18909487 .
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Wrong parameter setting of transmission range

The output parameters of the function block "LOpenUserComm_Udp" are set for a cycle as in [Table 2-24](#) if the following conditions are met:

- Send area too small
- Value at parameter "sendLen" is greater than the send area specified at parameter "sendData"

Table 2-24

Output parameters	Value	Description	Remedy
status	16#8088	Status display of the "TUSEND" instruction	The value at parameter "sendLen" must not be larger than the transmission range specified at parameter "sendData".
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-25](#) if one of the following communication errors occurs:

- The specified connection between user program and communication layer of the operating system has not yet been established.
- The specified connection between the user program and the communication layer of the operating system is being disconnected. Transmission over this connection is not possible.
- Reinitializing the interface.

Table 2-25

Output parameters	Value	Description	Remedy
status	16#80A1	Status display of the "TUSEND" instruction	The connection must be established to send data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Temporary communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for a cycle as in [Table 2-26](#) if one of the following temporary communication errors occurs.

- The connection between the user program and the communication layer of the operating system cannot be established at this time.
- The interface is being reconfigured.
- Temporarily no receive resource is available at the connection partner. The connection partner is not ready to receive.

Table 2-26

Output parameters	Value	Description	Remedy
status	16#80C4	Status display of the "TUSEND" instruction	The connection must be established to send data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

IP address of remote connection endpoint in parameter "addrTusend" invalid

If the IP address of the remote endpoint in parameter "addrTusend" is invalid or matches the own IP address, the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-27

Output parameters	Value	Description	Remedy
status	16#80A4	Status display of the "TUSEND" instruction	Check whether you have defined the IP address of the remote connection endpoint in the "addrTusend" parameter.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Invalid remote IP addresses are among others:

- Broadcast addresses, e.g. 192.168.0.255
- Network addresses, e.g. 192.168.0.0

2.5.4 Error when disconnecting

Order to terminate connection with "TDISCON" was not executed (watchdog timer expired)

The watchdog timer is started as soon as a connection release order is started with TDISCON (REQ = 1). A disconnect request is started when one of the following conditions is met:

- A falling edge is detected at the "enable" input of the FB "LOpenUserComm_Udp".
- An error occurs when establishing a connection with TCON, so that the connection must first be terminated before it can be re-established.
- An error occurs when receiving or sending data.

If the order to terminate the connection was executed with "TDISCON" with DONE = 1 or ERROR = 1, the watchdog timer is reset.

If the job to terminate the connection with "TDISCON" has not been completed after 30 s, the watchdog timer runs out and the FB "LOpenUserComm_Udp" changes to the state "STATE_IDLE", so that the connection can be established again with enable = 1.

If you try to establish the connection with enable = 1 while the watchdog timer is running, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-28

Output parameters	Value	Description	Remedy
status	16#8104	Connection could not be terminated (watchdog timer (30 s) expired)".	Initiate the connection setup again via the parameter enable = 1.
statusId	1	Internal error in function module	
Error	1	Error display: 1: Error identified	

Connection is being terminated

If the connection has been successfully terminated with TDISCON (DONE = 1), it can take up to 60 s until the connection has been terminated. Therefore a timer of 60 s is started after a successful disconnection.

If you try to establish the connection with enable = 1 while the timer is running, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-29

Output parameters	Value	Description	Remedy
status	16#8103	Connection is being terminated (60 s).	Initiate the connection setup again via the parameter enable = 1.
statusId	1	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Connection does not exist or has already been terminated

If the connection referenced via the parameter "ID" of the parameter data set "TCON_IP_V4" does not exist or has already been terminated, the output parameters of the FB "LOpenUseComm_Udp" for one cycle are set as follows.

Table 2-30

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TDISCON" instruction	Initiate the connection setup again via the parameter enable = 1.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Connection ID is outside the permitted range

If the parameter "ID" of the parameter data set "TCON_IP_V4" is outside the permissible range, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-31

Output parameters	Value	Description	Remedy
status	16#8086	Status display of the "TDISCON" instruction	In the parameter data set "TCON_IP_V4", enter a value for "ID" that lies in the value range from 1 to 4095.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Temporary communication errors

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-32](#) if one of the following temporary communication errors occurs:

- Interface is re-parameterized
- Connection is being established

Table 2-32

Output parameters	Value	Description	Remedy
status	16#80C4	Status display of the "TDISCON" instruction	The FB's trying to disconnect again.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

2.6 Error handling S7-300 and S7-400

In the FB "LOpenUserComm_Udp" some error states are intercepted exemplarily and reactions to them are programmed. However, you can also catch all error states that the OUC statements supply here and implement your own reactions according to this schema, depending on your requirements.

Note Since in the event of an error, the values at the output parameters are only output for one cycle, the values of "status" and "statusId" must be stored if error = 1.

2.6.1 Error when establishing connection

Job to establish connection with "TCON" could not be executed successfully (watchdog timer expired)

The watchdog timer is started as soon as a rising edge is detected at the "enable" input of the FB "LOpenUserComm_Udp".

If the order to establish the connection was successfully executed with "TCON" (DONE = 1), the watchdog timer is reset, but there is no acknowledgement from the communication partner that the communication connection has really been established successfully.

If the job to establish the connection with "TCON" has not been successfully completed after 30 s, the watchdog timer runs out and the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-33

Output parameters	Value	Description	Remedy
status	16#8102	Connection could not be established (watchdog timer (30 s) expired).	Check whether an error occurred when establishing a connection with "TCON" (ERROR = 1): Evaluate the output parameters "statusID" and "status" of the FB "LOpenUserComm_Udp" and, if necessary, correct the error when establishing the connection with "TCON". Initiate the connection setup again via the parameter enable = 1.
statusId	1	Internal error in function module	
Error	1	Error display: 1: Error identified	

Identifier for the type of connection not permitted

With UDP, only a passive connection setup is permitted. If you use the value "TRUE" for "active_est" in the parameter data set "TCON_PAR", the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-34

Output parameters	Value	Description	Remedy
status	16#80B5	Status display of the "TCON" instruction	Enter the value "FALSE" for "active_est" in the parameter data set "TCON_PAR". Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Parameterization error in parameter "connection_type"

If there is a parameterization error in the parameter "connection_type" of the parameter data set "TCON_PAR", the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-35

Output parameters	Value	Description	Remedy
status	16#80B6	Status display of the "TCON" instruction	Enter the value 0x13 (hex) for "connection_type" in the parameter data set "TCON_PAR". Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Length parameter in parameter data set "TCON_PAR" wrong parameterized

The output parameters of the FB "LOpenUserComm_Udp" are set for a cycle as in [Table 2-36](#), if the following length parameters in the parameter data set "TCON_PAR" are incorrectly parameterized:

- block_length: Length of the parameter data set "TCON_PAR": 64 bytes (fixed)
- local_tsap_id_len: used length of the parameter local_tsap_id (local port)
 - 2: 2 bytes
- rem_subnet_id_len: Parameter is currently not used: B#16#00
- rem_staddr_len: used length of the parameter rem_staddr
 - 0: Parameter rem_staddr is irrelevant
- rem_tsap_id_len: Used length of the parameter rem_tsap_id
 - 0: Parameter rem_tsap_id is irrelevant
- next_staddr_len: Length of parameter used next_staddr
 - 0: Parameter rem_staddr is irrelevant

Table 2-36

Output parameters	Value	Description	Remedy
status	16#80B7	Status display of the "TCON" instruction	Check the local port in the parameter data set "TCON_PAR". The valid value range is 1 to 49151. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection or port are already occupied

Each connection is uniquely defined by the connection ID and port number. If you use the same port number for several connections, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-37

Output parameters	Value	Description	Remedy
status	16#80A1	Status display of the "TCON" instruction	Use a different port number for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Local or remote port used by the system

If you define a port number for the local or remote port in the parameter data set "TCON_PAR" that is used by the system, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-38

Output parameters	Value	Description	Remedy
status	16#80A2	Status display of the "TCON" instruction	The following ports are reserved by the system: 20, 21, 80, 102, 135, 161, 162, 443, 34962, 34963, 34964 and the range 49152 to 65535 For the local and remote ports, use a port number that is not reserved by the system. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection ID is already in use

Each connection is uniquely defined by the connection ID and port number.

1. If you use the same connection ID for several connections, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-39

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TCON" instruction	Use a different connection ID for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

2. If the connection ID is already used by an established connection that uses the same parameter record at the udpConnParam parameter, an attempt is made to establish an existing connection. The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-40

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TCON" instruction	Use a separate parameter data set with different connection ID and port number for each connection. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Connection ID is outside the permitted range

If the parameter "id" of the parameter data set "TCON_PAR" is outside the permissible range, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-41

Output parameters	Value	Description	Remedy
status	16#8086	Status display of the "TCON" instruction	In the parameter data set "TCON_PAR", enter a value for "id" that lies in the value range from W#16#0001 to 16#0FFF. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

"local_device_id" does not match the CPU

If the "local_device_id", which is specified in the parameter data set "TCON_PAR", does not match the CPU, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-42

Output parameters	Value	Description	Remedy
status	16#809B	Status display of the "TCON" instruction	Check whether the "local_device_id", which is specified in the parameter data set "TCON_PAR", matches the CPU. Detailed information about the parameter "local_device_id" can be found in the following FAQ 51339682 . Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Communication error: "TDISCON" was executed before "TCON" was finished

If the instruction "TCON" has finished the connection establishment with DONE = 1 or ERROR = 1, a job to terminate the connection may be initiated.

If the connection establishment is aborted prematurely by calling a "TDISCON", the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-43

Output parameters	Value	Description	Remedy
status	16#80A7	Status display of the "TCON" instruction	Since the connection setup was aborted prematurely by calling a "TDISCON", the connection is terminated by the FB. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

Order to establish connection is initiated during disconnection

If the job to establish a connection is triggered while the connection is being terminated, an attempt is made to establish an existing connection. The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-44

Output parameters	Value	Description	Remedy
status	16#80A3	Status display of the "TCON" instruction	Since you are trying to re-establish an existing connection, the connection is terminated by the FB. Initiate the connection setup again via the parameter enable = 1.
statusId	2	Error when establishing a connection with "TCON"	
Error	1	Error display: 1: Error identified	

2.6.2 Error when receiving data

Parameterized length of received data invalid

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-45](#) if the following conditions are given:

- The parameterized length of the receive data is greater than the largest permissible value.
- The value of the parameter "rcvLen" or "rcvData" was changed after the first call.

Table 2-45

Output parameters	Value	Description	Remedy
status	16#8085	Status display of the "TURCV" instruction	Enter the length of the data to be received at the parameter "rcvLen". Specify the reception range at the "rcvData" parameter. Detailed information about the maximum amount of user data that can be transferred with UDP can be found in the following FAQ: 18909487 .
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Receive range wrong parameterized

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-46](#) if the following conditions are given:

- Receive area is too small
- Value at parameter "rcvLen" greater than the receive range specified at parameter "rcvData"

Table 2-46

Output parameters	Value	Description	Remedy
status	16#8088	Status display of the "TURCV" instruction	The value at parameter "rcvLen" must not be larger than the receive range specified at parameter "rcvData".
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-47](#) if one of the following communication errors occurs:

- The specified connection between user program and communication layer of the operating system has not yet been established.
- The specified connection between the user program and the communication layer of the operating system is being disconnected. A receive job over this connection is not possible.
- The interface is being reconfigured.

Table 2-47

Output parameters	Value	Description	Remedy
status	16#80A1	Status display of the "TURCV" instruction	The connection must be established to receive data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Temporary communication errors

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-48](#) if a temporary communication error occurs:

- The interface is being reconfigured.

Table 2-48

Output parameters	Value	Description	Remedy
status	16#80C4	Status display of the "TURCV" instruction	The connection must be established to receive data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

IP address of remote connection endpoint in parameter "addrTurcv" invalid

If the IP address of the remote endpoint in parameter "addrTurcv" is invalid or matches the own IP address, the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-49

Output parameters	Value	Description	Remedy
status	16#80A4	Status display of the "TURCV" instruction	Check that you have defined the IP address of the remote connection endpoint in the "addrTurcv" parameter.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

Invalid remote IP addresses are among others:

- Broadcast addresses, e.g. 192.168.0.255
- Network addresses, e.g. 192.168.0.0

Parameter "rcvData" was changed before the current job was finished.

If you change the parameter "rcvData" before the current receive job has been completed, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

- Table 2-50

Output parameters	Value	Description	Remedy
Status	16#80B1	Status display of the "TURCV" instruction	Initiate the connection setup again via the parameter enable = 1.
statusId	3	Error when receiving data with "TURCV".	
Error	1	Error display: 1: Error identified	

2.6.3 Error sending data

Parameterized length of transmission data invalid

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-51](#) if the following conditions are given:

- The parameterized length of the transmit data is greater than the largest permissible value.
- The parameter "sendLen" of the FB "LOpenUserComm_Udp" has the value "0".

Table 2-51

Output parameters	Value	Description	Remedy
Status	16#8085	Status display of the "TUSEND" instruction	Specify the length of the data to be sent at the "sendLen" parameter. Detailed information about the maximum amount of user data that can be transferred with UDP can be found in the following FAQ: 18909487 .
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Wrong parameter setting of transmission range

The output parameters of the function block "LOpenUserComm_Udp" are set for a cycle as in [Table 2-52](#) if the following conditions are met:

- Transmission range too small
- Value at parameter "sendLen" is greater than the send area specified at parameter "sendData"

Table 2-52

Output parameters	Value	Description	Remedy
Status	16#8088	Status display of the "TUSEND" instruction	The value at parameter "sendLen" must not be larger than the send area specified at parameter "sendData".
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-53](#) if one of the following communication errors occurs:

- The specified connection between user program and communication layer of the operating system has not yet been established.
- The specified connection between the user program and the communication layer of the operating system is being disconnected. Transmission over this connection is not possible.
- Reinitializing the interface.

Table 2-53

Output parameters	Value	Description	Remedy
Status	16#80A1	Status display of the "TUSEND" instruction	The connection must be established to send data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Temporary communication error

The output parameters of the FB "LOpenUserComm_Udp" are set for a cycle as in [Table 2-54](#) if one of the following temporary communication errors occurs.

- The connection between the user program and the communication layer of the operating system cannot be established at this time.
- The interface is being reconfigured.
- Temporarily no receive resource is available at the connection partner. The connection partner is not ready to receive.

Table 2-54

Output parameters	Value	Description	Remedy
Status	16#80C4	Status display of the "TUSEND" instruction	The connection must be established to send data. In the event of a communication error, the connection is terminated by the FB so that it can be re-established.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

IP address of remote connection endpoint in parameter "addrTusend" invalid

If the IP address of the remote endpoint in parameter "addrTusend" is invalid or matches the own IP address, the output parameters of the FB "LOpenUser_Udp" for one cycle are set as follows.

Table 2-55

Output parameters	Value	Description	Remedy
Status	16#80A4	Status display of the "TUSEND" instruction	Check whether you have defined the IP address of the remote connection endpoint in the "addrTusend" parameter.
statusId	4	Error when sending data with "TSEND".	
Error	1	Error display: 1: Error identified	

Invalid remote IP addresses are among others:

- Broadcast addresses, e.g. 192.168.0.255
- Network addresses, e.g. 192.168.0.0

2.6.4 Error when disconnecting

Order to terminate connection with "TDISCON" was not executed (watchdog timer expired)

The watchdog timer is started as soon as a connection release order is started with TDISCON (REQ = 1). A disconnect request is started when one of the following conditions is met:

- A falling edge is detected at the "enable" input of the FB "LOpenUserComm_Udp".
- An error occurs when establishing a connection with TCON, so that the connection must first be terminated before it can be re-established.
- An error occurs when receiving or sending data.

If the order to terminate the connection was executed with "TDISCON" with DONE = 1 or ERROR = 1, the watchdog timer is reset.

If the job to terminate the connection with "TDISCON" has not been completed after 30 s, the watchdog timer runs out and the FB "LOpenUserComm_Udp" changes to the state "STATE_IDLE", so that the connection can be established again with enable = 1.

If you try to establish the connection with enable = 1 while the watchdog timer is running, the output parameters of the FB "LOpenUserComm_Udp" for one cycle are set as follows.

Table 2-56

Output parameters	Value	Description	Remedy
Status	16#8104	Connection could not be terminated (watchdog timer (30 s) expired)".	Initiate the connection setup again via the parameter enable = 1.
statusId	1	Internal error in function module	
Error	1	Error display: 1: Error identified	

Connection is being terminated

If the connection has been successfully terminated with TDISCON (DONE = 1), it can take up to 60 s until the connection has been terminated. Therefore a timer of 60 s is started after a successful disconnection.

If you try to establish the connection with enable = 1 while the timer is running, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-57

Output parameters	Value	Description	Remedy
Status	16#8103	Connection is being terminated (60 s).	Initiate the connection setup again via the parameter enable = 1.
statusId	1	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Connection does not exist or has already been terminated

If the connection referenced via the parameter "id" of the parameter data set "TCON_PAR" does not exist or has already been terminated, the output parameters of the FB "LOpenUseComm_Udp" for one cycle are set as follows.

Table 2-58

Output parameters	Value	Description	Remedy
Status	16#80A3	Status display of the "TDISCON" instruction	Initiate the connection setup again via the parameter enable = 1.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Connection ID is outside the permitted range

If the parameter "id" of the parameter data set "TCON_PAR" is outside the permissible range, the output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as follows.

Table 2-59

Output parameters	Value	Description	Remedy
Status	16#8086	Status display of the "TDISCON" instruction	In the parameter data set "TCON_PAR", enter a value for "id" that lies in the value range from W#16#0001 to 16#0FFF.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

Temporary communication errors

The output parameters of the FB "LOpenUserComm_Udp" are set for one cycle as in [Table 2-60](#) if one of the following temporary communication errors occurs:

- Interface is re-parameterized
- Connection is being established

Table 2-60

Output parameters	Value	Description	Remedy
Status	16#80C4	Status display of the "TDISCON" instruction	The FB's trying to disconnect again.
statusId	5	Error when disconnecting with "TDISCON".	
Error	1	Error display: 1: Error identified	

2.7 Explanations on the state machine

The state machine contains the following states:

- STATE_IDLE
- STATE_PARAM
- STATE_CONNECT
- STATE_CONNECTED
- STATE_RECEIVE
- STATE_SEND
- STATE_ERROR

2.7.1 STATE_IDLE

In the idle state "STATE_IDLE" all parameters (static variables and outputs of the FB "LOpenUserComm_Udp") are reset.

The FB "LOpenUserComm_Udp" waits in the state "STATE_IDLE" until it detects a positive edge at the input parameter "enable".

2.7.2 STATE_CONNECT

The order to establish the connection is triggered if the following conditions are fulfilled:

- The input parameter "enable" is set to the value "true" so that the FB recognizes a positive edge at the input parameter "enable".
- The connection has not yet been established.
- In order for the FB to be able to correct an error independently, the connection setup is restarted internally.

Connection successfully established

If the connection with "TCON" is successfully established, the FB "LOpenUserComm_Udp" changes to the state "STATE_CONNECTED". The connection remains established until the connection with "TDISCON" is terminated again.

Error when establishing connection.

If an error occurs during connection establishment, the following actions are executed in the FB "LOpenUserComm_Udp":

- The error information is stored in the static variable "statStatus" of the data type "Word".
- The state in which the error occurs is stored in the static variable "statStatusId" of the data type "UInt".
- The FB changes to the state "STATE_ERROR".

The values of the variables "statStatus" and "statStatusId" are transferred in the state "STATE_ERROR" to the output parameters "status" and "statusId" of the FB "LOpenUserComm_Udp".

Table 2-61 shows the values and meaning of the output parameters "status" and "statusId".

2.7.3 STATE_CONNECTED

The reception of data with "TURCV" is enabled.

Successful reception of new data

If new data is received with "TURCV", the FB changes to the state "STATE_RECEIVE".

Error when receiving data with "TURCV".

If an error occurs when receiving the data with "TURCV", the following actions are executed in the FB "LOpenUserComm_Udp":

- The error information is stored in the static variable "statStatus" of the data type "Word".
- The state in which the error occurs is stored in the static variable "statStatusId" of the data type "UInt".
- The FB changes to the state "STATE_ERROR".

The values of the variables "statStatus" and "statStatusId" are transferred in the state "STATE_ERROR" to the output parameters "status" and "statusId" of the FB "LOpenUserComm_Udp".

Table 2-61 shows the values and meaning of the output parameters "status" and "statusId".

2.7.4 STATE_RECEIVE

The received data can be processed individually.

The FB "LOpenUserComm_Udp" changes back to the state "STATE_CONNECTED" without further switching condition.

2.7.5 STATE_SEND

A new send request is triggered if the following conditions are met:

- Positive edge at input parameter " sendRequest".
- No send request is active

If a send request is active, the following actions are executed in the FB "LOpenUserComm_Udp":

- A new send request cannot be triggered.

Send job successfully completed

If a send job is successfully completed with "TUSEND", the following actions are executed in the FB "LOpenUserComm_Udp":

- The output parameters "done" and "busy" are set to the following values for one cycle:
 - done = 1
 - busy = 0
- The FB changes back to the state "STATE_CONNECTED".

Error when sending data with "TUSEND".

If an error occurs when sending the data with "TUSEND", the following actions are executed in the FB "LOpenUserComm_Udp":

- The error information is stored in the static variable "statStatus" of the data type "Word".
- The state in which the error occurs is stored in the static variable "statStatusId" of the data type "UInt".
- The FB changes to the state "STATE_ERROR".

The values of the variables "statStatus" and "statStatusId" are transferred in the state "STATE_ERROR" to the output parameters "statStatus" and "statStatusId" of the FB "LOpenUserComm_Udp".

Table 2-61 shows the values and meaning of the output parameters "status" and "statusId".

2.7.6 STATE_DISCONNECT

The order to terminate the connection is triggered when the connection is established or when another attempt to establish a connection is to be initiated.

A watchdog timer (30 s) is started. If the job to terminate the connection is not completed, the timer runs out and the FB "LOpenUserComm_Udp" changes to the state "STATE_IDLE", so that a new job to establish the connection can be started with enable = 1.

Connection successfully established

If the connection is successfully terminated, the FB "LOpenUserComm_Udp" changes to the idle state "STATE_IDLE".

Since it can take up to 60 s until the connection with TDISCON is terminated, a timer of 60 s is started. A new job for establishing a connection can only be started after the timer has expired. If you try to establish the connection before the timer expires, the value "16#8103" is output at the "status" output.

Error when disconnecting

If an error occurs during connection establishment, the following actions are executed in the FB "LOpenUserComm_Udp":

- The error information is stored in the static variable "statStatus" of the data type "Word".
- The state in which the error occurs is stored in the static variable "statStatusId" of the data type "UInt".
- The FB changes to the state "STATE_ERROR".

The values of the variables "statStatus" and "statStatusId" are transferred in the state "STATE_ERROR" to the output parameters "status" and "statusId" of the FB "LOpenUserComm_Udp".

Table 2-61 shows the values and meaning of the output parameters "status" and "statusId".

2.7.7 STATE_ERROR

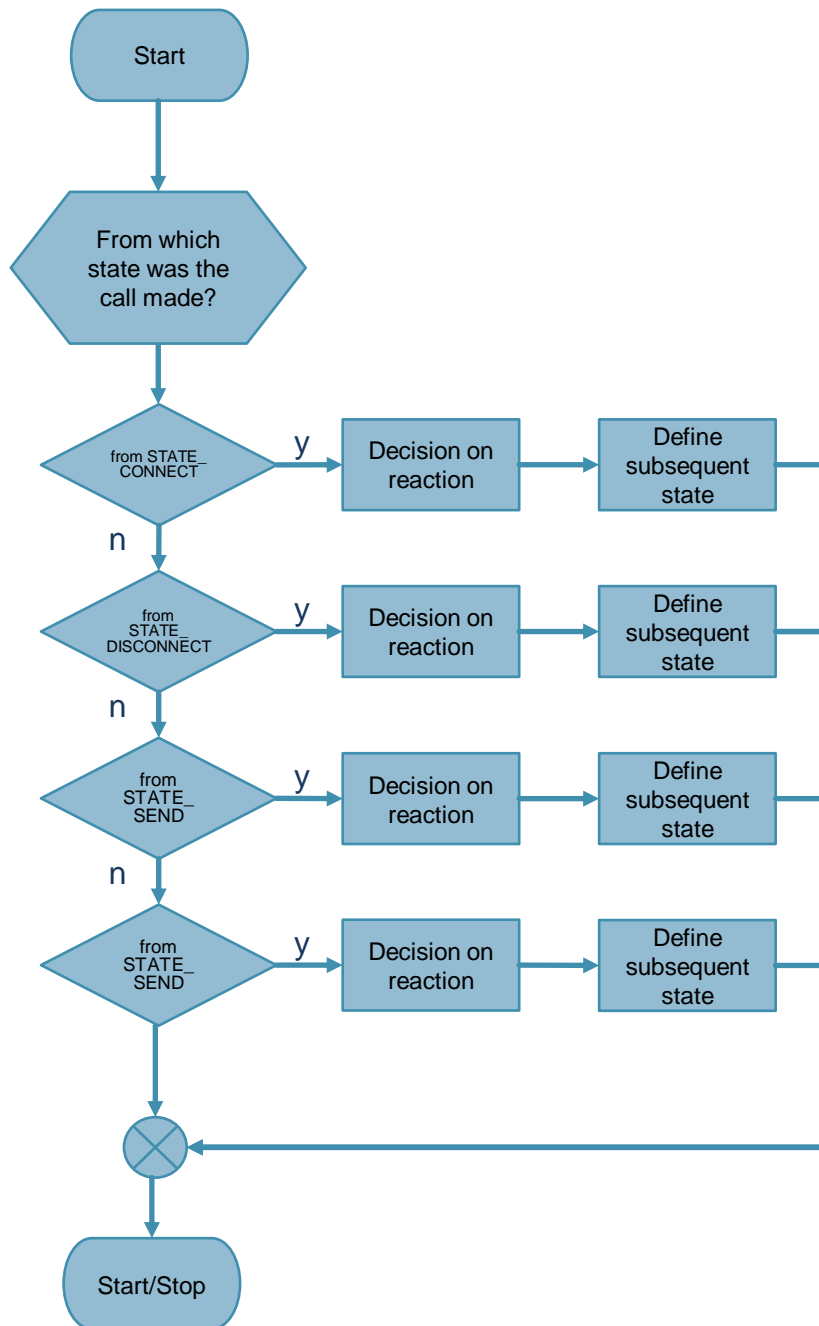
The state "STATE_ERROR" evaluates the most important error information of the OUC statements ("TON", "TDISCON", "TUSEND" and "TURCV") and shows the user how to react to this error information.

The user has the option of extending the "STATE_ERROR" state according to this schema as follows:

- Analyze further OUC error messages and realize individual reactions.
- Perform your own user-specific error analyses, e.g. if the received data does not correspond to what the program expects.

The following figure shows the general scheme according to which this state is realized.

Figure 2-5



The program block "Decision on reaction" defines how the FB "LOpenUserComm_Udp" reacts in the event of an error. The reaction depends on the state in which the error occurred. In this example, the following reactions are possible, depending on the cause of the error:

- If an error occurs during connection establishment, the following reactions are realized in the FB "LOpenUserComm_Udp":
 - In order to remedy the error independently, the FB changes to the state "STATE_DISCONNECT" or "STATE_CONNECT".
If an established connection must first be terminated before the connection establishment can be restarted internally, the FB changes to the state "STATE_DISCONNECT".
If the connection partner refuses to establish the connection or has actively terminated the connection, the FB changes to the state "STATE_CONNECT" in order to restart the connection establishment internally.
 - If an error occurs that must be corrected by the user, the FB switches to the idle state "STATE_IDLE". This error occurs, e.g. if the connection parameterization is faulty or there is a network error, so that the connection partner cannot be reached. The user must correct the error and then trigger the connection setup again via the "enable" parameter.
- If an error occurs while sending or receiving data, the following reactions are realized in the FB "LOpenUserComm_Udp":
 - If the specified connection between the user program and the communication layer of the operating system has not yet been established or is currently being terminated, the FB changes to the "STATE_DISCONNECT" state.
 - If the interface is re-parameterized, the FB changes to the state "STATE_DISCONNECT".
 - If an error occurs that must be corrected by the user, the FB changes to the state "STATE_DISCONNECT". This error occurs if the input parameter "rcvLen" or "sendLen" does not specify the correct length or the input parameter "rcvData" or "sendData" does not specify the correct send or receive buffer. The connection is terminated. The user must correct the error and then trigger the connection setup again via the "enable" parameter.
 - If the connection was terminated via the parameter "enable", the FB changes to the state "STATE_IDLE".
- If an error occurs during disconnection, the following reactions are realized in the FB "LOpenUserComm_Udp":
 - To start another attempt to terminate the connection, the FB changes to the state "STATE_DISCONNECT". If the connection is successfully terminated, the system waits in state "STATE_IDLE" until a positive edge is detected at input parameter "enable".

Note

If an error occurs, the output parameter "error" is set to the value "1" for one cycle and the corresponding error information is output at the output parameter "status". The output parameter "statusId" provides you with information on the status in which the error occurred.

The following table shows the values and meaning of the output parameters "status" and "statusId" of the function module "LOpenUserComm_Udp".

Table 2-61

statusId Wert (dec)	Meaning	Status Wert (hex)	Comment
1	internal error in function module	16#8101	Function module could not be activated
1	internal error in function module	16#8102	Connection setup failed. The watchdog timer (30 s) has expired.
1	internal error in function module	16#8103	Connection termination is still active. Connection setup cannot yet be started.
1	Internal error in function module	16#8104	Connection cannot be disconnected. Watchdog timer (30 s) running. Restart connection setup with enable = 1.
2	Error at TCON	-	The status of TCON is output at the output parameter "status" of the function block.
3	Error at TURCV	-	The status of TURCV is output at the output parameter "status" of the function module.
4	Error at TUSEND	-	The status of TUSEND is output at the output parameter "status" of the function module.
5	Error at TDISCON	-	The status of TDISCON is output at the output parameter "status" of the function module.

3 Useful information

3.1 Basics

3.1.1 Basics of the UDP protocol

Further information about the UDP protocol can be found in the following FAQ:

<https://support.industry.siemens.com/cs/ww/en/view/26171811>

3.1.2 Structure of the parameter data set "TCON_IP_V4".

In order to parameterize the communication connections for UDP, a connection description DB with a structure according to "TCON_IP_V4" is used for CPUs of the S7-1200 from V4.0 and S7-1500. The fixed data structure of "TCON_IP_V4" contains the necessary parameters that are required to establish the connection.

The connection parameter "udpConnParam" of the FB "LOpenUserComm_Udp" contains a reference to the used data block.

Table 3-1

Byte	Parameters	Data type	Value	Description
0 to 1	InterfaceId	HW_ANY	64	Hardware identification of the local interface (value range: 0 to 65535)
2 to 3	ID	CONN_OUC	1	Connection number (value range 0 to 4095)
4	ConnectionType	BYTE	19	Connection type: <ul style="list-style-type: none"> • 19: UDP (19 dec = 0x13 hex)
5	ActiveEstablished	BOOL	FALSE	Identifier for the manner in which the connection is established: You must assign FALSE to this parameter.
6 to 9	RemoteAddress	ARRAY [1..4] of BYTE	-	<ul style="list-style-type: none"> • IP address of the partner endpoint, e.g. for 192.168.0.2: <ul style="list-style-type: none"> - ADDR[1] = 192 - ADDR[2] = 168 - ADDR[3] = 0 - ADDR[4] = 2 • Multicast address of an IPv4 multicast group for S7-1500 CPUs V2.0 and later
10 to 11	RemotePort	UINT	2000	Port address of the remote connection partner (Value range: 1 to 49151)
12 to 13	LocalPort	UINT	2000	Port of the local connection partner (value range 1 to 49151)

Note

We recommend that you use the same port for the local and remote connection partners.

3.1.3 Structure of the "TCON_PAR" parameter data set

To parameterize the communication connections at UDP, create a DB for the S7-300 and S7-400 CPUs, which contains the data structure from the UDT 65 "TCON_PAR". This data structure contains the necessary parameters that you need to set up the connection.

The connection parameter "udpParam" of the FB "LOpenUserComm_Udp" contains a reference to the used data block.

Table 3-2

Byte	Parameters	Data type	Start value	Description
0 to 1	block_length	WORD	W#16#40	Length of UDT 65: 64 bytes (fixed)
2 to 3	id	WORD	W#16#0001	Connection number: (Value range: W#16#0001 to W#16#0FFF)
4	connection_type	BYTE	B#16#13	Protocol variant B#16#13: UDP
5	active_est	BOOL	FALSE	Identifier for the manner in which the connection is established: You must assign FALSE to this parameter.
6	local_device_id	BYTE	B#16#02	Detailed information on the parameter "local_device_id" can be found in the article 51339682 .
7	local_tsap_id_len	BYTE	B#16#02	Used length of the local parameter local_tsap_id (local port): 2 bytes
8	rem_subnet_id_len	BYTE	B#16#00	This parameter is not used. Assign B#16#00 to it.
9	rem_staddr_len	BYTE	B#16#00	This parameter is not used. Assign B#16#00 to it.
10	rem_tsap_id_len	BYTE	B#16#00	This parameter is not used. Assign B#16#00 to it.

3 Useful information

11	next_staddr_len	BYTE	B#16#00	This parameter is not used. Assign B#16#00 to it.
12 to 27	local_tsap_id	ARRAY [1..16] of BYTE	-	Local port number local_tsap_id[1] = high byte of the port number in hexadecimal notation local_tsap_id[2] = low byte of port number in hexadecimal notation local_tsap_id[3-16] = B#16#00
28 to 33	rem_subnet_id	ARRAY [1..6] of BYTE	-	This parameter is not used. Assign B#16#00 to it.
34 to 39	rem_staddr	ARRAY [1..6] of BYTE	-	This parameter is not used. Assign B#16#00 to it.
40 to 55	rem_tsap_id	ARRAY [1..16] of BYTE	-	This parameter is not used. Assign B#16#00 to it.
56 to 61	next_staddr	ARRAY [1..6] of BYTE	-	This parameter is not used. Assign B#16#00 to it.
62 to 63	Spare	WORD	W#16#0000	Reserve Assign W#16#0000 to this parameter.

3.1.4 Structure of the address information of the communication partner according to "TADDR_Param".

When using UDP connections in S7-1500 CPUs and S7-1200 CPUs, the address information of the communication partner is stored in the system data type "TADDR_Param":

- At the parameter "addrTusend" you pass the address information of the recipient via "TADDR_Param".
The stored address data of the remote partner are read from the system data type by the "TUSEND" instruction.
- At the parameter "addrTurcv" you get the address of the sender via "TADDR_Param".
The address data is written from the instruction "TURCV" to the system data type.

The system data type "TADDR_Param" contains the address information of the communication partner, consisting of the IP address and the port number.

Table 3-3

Byte	Parameters	Data type	Value	Description
0 to 3	REM_IP_ADDR	ARRAY [1..4] of BYTE	-	<ul style="list-style-type: none"> • IP address of the communication partner, e.g. 192.168.0.2 <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#00 (0) - REM_IP_ADDR[4] = B#16#02 (2) <p>The IP address can be found in the "Devices & Networks" view in the properties of the interface of the communication partner.</p> <ul style="list-style-type: none"> • Multicast address of an IPv4 multicast group: For the instruction "TUSEND" at S7-1500 CPUs from firmware version V2.0. • The "TURCV" statement enters the IP address of the sender in this parameter if the sender is in an IPv4 multicast group.
4 to 5	REM_PORT_NR	UINT	2000	Remote port number (Value range: 1 to 49151)
6 to 7	RESERVED	UINT	0	Not used. Leave this parameter at the value "0".

3.1.5 Structure of the address information of the communication partner according to "TADDR_PAR".

When using UDP connections in S7-300 CPUs and S7-400 CPUs, the address information of the communication partner is stored in the data structure UDT 66 "TADDR_PAR":

- At the parameter "addrTusend" you pass the address information of the recipient via "TADDR_PAR".
The stored address data of the remote partner are read from the system data type by the "TUSEND" instruction.
- At the parameter "addrTurcv" you get the address of the sender via "TADDR_PAR".
The address data is written from the instruction "TURCV" to the system data type.

The system data type "TADDR_PAR" contains the address information of the communication partner, consisting of the IP address and the port number.

Table 3-4

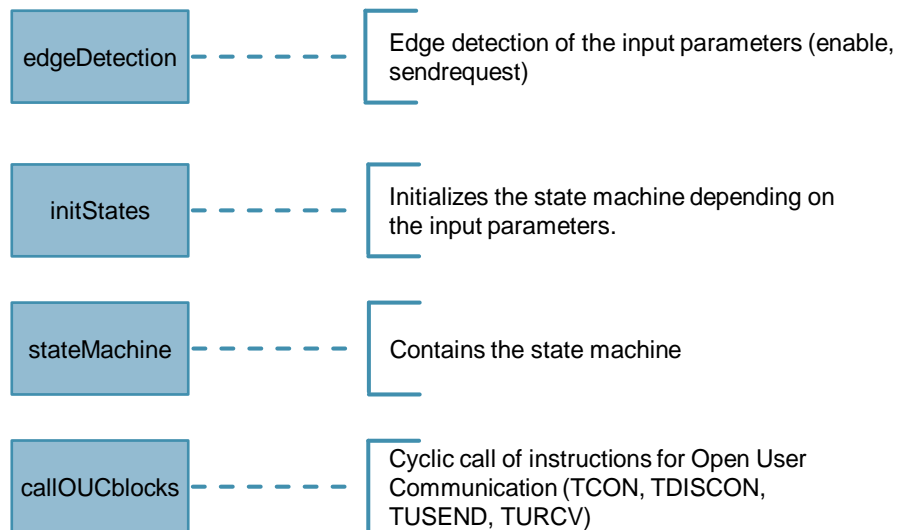
Byte	Parameters	Data type	Value	Description
0 to 3	rem_ip_addr	ARRAY [1..4] of BYTE	-	<ul style="list-style-type: none"> • IP address of the communication partner, e.g. 192.168.0.2 <ul style="list-style-type: none"> - REM_IP_ADDR[1] = B#16#C0 (192) - REM_IP_ADDR[2] = B#16#A8 (168) - REM_IP_ADDR[3] = B#16#00 (0) - REM_IP_ADDR[4] = B#16#03 (3) <p>The IP address can be found in the "Devices & Networks" view in the properties of the interface of the communication partner.</p>
4 to 5	rem_port_nr	ARRAY [1..2] of BYTE	2001	<p>Remote port number, e. g. 2001 (dec) = 7D1 (hex)</p> <ul style="list-style-type: none"> • rem_port_nr[1] = high byte of the port number in hexadecimal notation = B#16#07 • rem_port_nr[2] = low byte of the port number in hexadecimal notation = B#16#D1
6 to 7	Spare	ARRAY [1..2] of BYTE		<p>Reserve</p> <p>Assign "0" to this parameter.</p>

3.2 Details on functionality

3.2.1 Structure of the program

The following figure shows the structure of the FB "LOpenUserComm_Udp". The program consists of four regions.

Figure 3-1



3.2.2 Explanation of the region "edgeDetection"

The region "edgeDetection" contains the edge evaluation of the following input parameters:

- "enable": Positive and negative edges are evaluated
- "sendRequest": Positive edge are evaluated

3.2.3 Explanation of the region "initStates"

If a positive edge is detected at the input parameter "enable", the following actions are executed in the FB "LOpenUserComm_Udp":

- Output parameter "busy" is set to the value "1".
- State machine is initialized with the state "STATE_PARAM" in order to read in the connection parameters and initiate a job for establishing the connection.
- State machine is initialized with the state "STATE_DISCONNECT" to terminate the connection if it is not successfully established within 30 s.

If a negative edge is detected at the input parameter "enable", the state machine is initialized with the state "STATE_DISCONNECT" in order to initiate a job to terminate the connection.

If a positive edge is detected at the input parameter "sendRequest" and the connection is established, the state machine is initialized with the state "STATE_SEND" in order to initiate a send job.

3.2.4 Explanation of the region "stateMachine"

A detailed description of the state machine can be found in section [2.7](#).

3.2.5 Explanation of the region "callOUCblocks".

The following OUC instructions are called cyclically in the FB "LOpenUserComm_Udp":

- TCON to establish the UDP connection
- TDISCON to terminate the UDP connection
- TUSEND to send data to the communication partner via the UDP connection.
- TURCV to receive data from the communication partner via the UDP connection.

4 Appendix

4.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:

<https://support.industry.siemens.com>

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:

www.siemens.com/industry/supportrequest

SITRAIN – Training for Industry

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:

www.siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:

<https://support.industry.siemens.com/cs/sc>

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for Apple iOS, Android and Windows Phone:

<https://support.industry.siemens.com/cs/ww/en/sc/2067>

4.2 Links and Literature

Table 4-1

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to the entry page of the application example https://support.industry.siemens.com/cs/ww/de/view/109747710
\3\	SIMATIC STEP 7 Basic/Professional V15 and SIMATIC WinCC V15 https://support.industry.siemens.com/cs/ww/en/view/109764516
\4\	SIMATIC STEP 7 Basic/Professional V15.1 and SIMATIC WinCC V15.1 https://support.industry.siemens.com/cs/ww/en/view/109755202

4.3 Change documentation

Table 4-2

Version	Date	Change
V1.0	07/2019	First version