Open IE Communication: Data Exchange S7-300/400 <-> S7-1200

S7-1200

Configuration Example X18 • August 2010

Applications & Tools

Answers for industry.

SIEMENS

Warranty, Liability and Support

Note

The application examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The application examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are correctly used. These application examples do not relieve you of the responsibility of safely and professionally using, installing, operating and servicing equipment. When using these application examples, you recognize that Siemens cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these application examples at any time without prior notice. If there are any deviations between the recommendations provided in these application examples and other Siemens publications – e.g. Catalogs – then the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc. described in this application example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

It is not permissible to transfer or copy these application examples or excerpts of them without first having prior authorization from Siemens Industry Sector in writing.

For questions about this document please use the following e-mail address:

online-support.automation@siemens.com

Table of Contents

Warranty, Liability and Support 2

1 Automation Problem 4

1.1 Problem definitions 4
1.2 Configuration 5
List of components 6

2 Automation Solution 7

2.1 Connection diagram 7
2.2 Program structure 9
2.2.1 S7-300 block structure 9
2.2.2 S7-1200 block structure 10
2.3 Used blocks 12
2.3.1 Master (S7-300) 12
2.3.2 Slave (S7-1200) 14
2.3.3 Data consistency 15
2.4 Program flow 17
2.4.1 Program flow in the master controller 17
2.4.2 Program flow in the slave controller 19

3 Configuration 21

3.1 Installing the hardware and software 21
3.1.1 Installing and wiring the hardware 21
3.1.2 Installing the software 21
3.2 Configuring the hardware and network 22
3.2.1 Assigning the IP address of the PG/PC 22
3.2.2 Configuring the S7-300 24
Configuring the PG/PC interface 24
Creating a connection 25
Setting the synchronization time 28
Downloading the master project to the controller 30
3.2.3 Configuring the S7-1200 33
Creating a connection 33
Downloading the slave project to the controllers 34
3.3 Activating online mode 36
3.4 Live demo 39
3.4.1 Cyclic sequence 39
3.4.2 User data transfer 39
S7-300 master -> S7-1200 slaves 39
S7-1200 slave 1 -> S7-300 master 40
S7-1200 slave 2 -> S7-300 master 41
3.4.3 Time-of-day synchronization 42
Manual synchronization of one slave 42
Automatic synchronization of all slaves 43
3.4.4 Communication errors 44
3.4.5 Master power failure 45

4 Code Elements 46

5 History 47

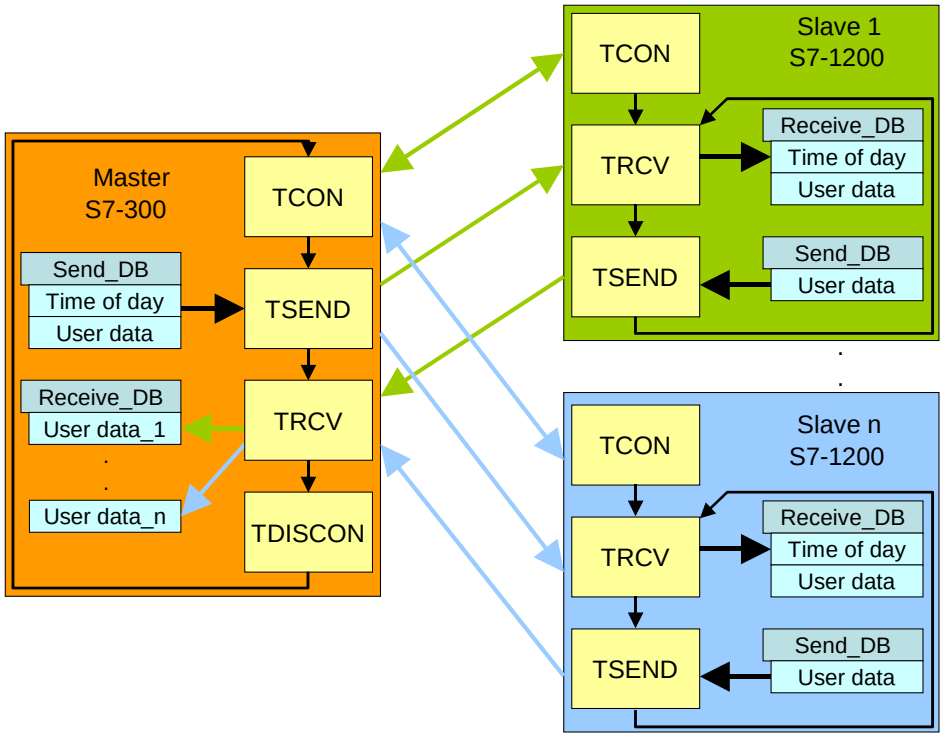
1 Automation Problem

1.1 Problem definitions

With the aid of open TCP/IP communication, deterministic data exchange (for example, for time-of-day synchronization) is to take place between one S7-300 master controller and several S7-1200 slave controllers via Industrial Ethernet.

Diagrammatic representation of the application task

Figure 1-1



Requirements for the application

Both the master and each slave have one send and one receive data block (Send_DB and Receive_DB).
Via the TCON block, the master sends a TCP/IP connection request to the first slave. To acknowledge connection establishment, the opposite side also executes the TCON block.
When there is a synchronization job, the master reads the system time and sends this time and the user data to the slave via the TSEND communication block. With the TRCV receive block, this block receives the data in the Receive_DB data block. The slave synchronizes its system time with the time of day received from the master.
Then slave 1 sends its user data to the master via the TSEND block. On the master side, the TRCV block is used to store the user data of slave 1 at a specified location in the receive data block.
Subsequently, the master disconnects the connection to slave 1 using the TDISCON block.

This procedure is repeated for the following slaves. After the master has exchanged data with the last slave, the master restarts data exchange with slave 1.

Once a connection has been established on the slave side, it remains reserved. TCON thus has to be called only for initialization.

1.2 Configuration

The automation problem is demonstrated using the example of data exchange between a CPU 315-2PN/DP as a master and two S7-1200 controllers (slave 1 and slave 2).

Schematic configuration

Figure 1-2

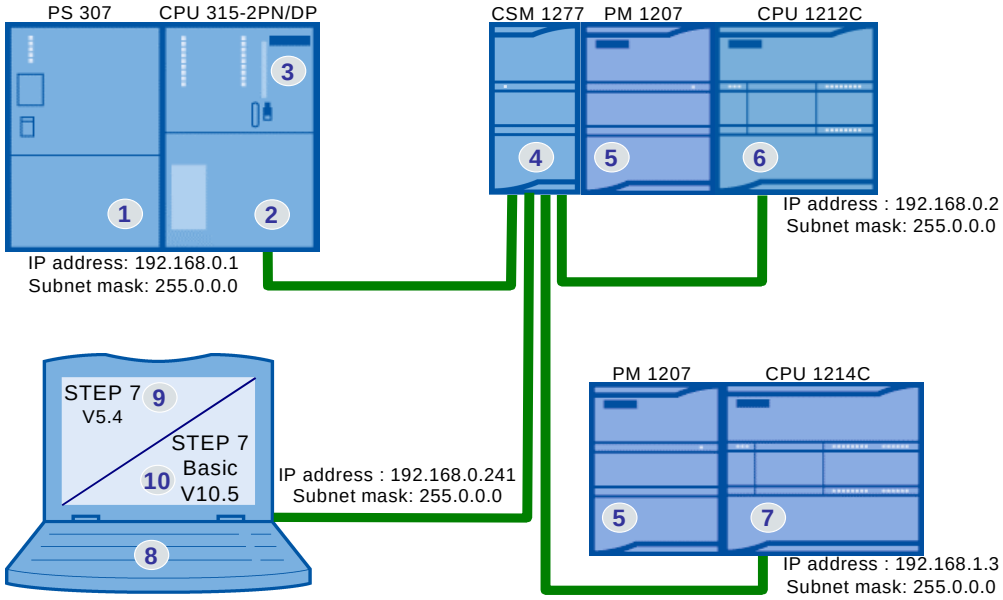


Figure 1-2 shows the basic configuration. The communicating CPUs and the programming unit with the “STEP 7 Basic V10.5” software for programming the S7-1200 and “STEP 7 V5.4” for programming the S7-300 are connected to the CSM 1277 using Ethernet cables.

Subnet mask

Since the CPU 1214C with IP address “192.168.1.3” is not in the same subnet as the other nodes with IP addresses “192.168.0.x”, this configuration is a Class B network. Communication of the nodes requires at least subnet mask “255.255.0.0” for Class B networks. However, Class A subnet mask “255.0.0.0” is selected, which allows communication with all nodes with IP address “192.x.x.x”.

List of components

Table 1-1

No.	Component	Qty.	MLFB/order number
1.	PS307 24V / 5A	1	6ES7307-1EA00-0AA0
2.	CPU315-2 PN/DP, 256 KB	1	6ES7315-2EH13-0AB0
3.	S7 MICRO MEMORY CARD, 8 MB	1	6ES7953-8LP10-0AA0
4.	CSM 1277 COMPACT SWITCH MODULE	1	6GK7277-1AA00-0AA0
5.	S7-1200 PM1207 POWER SUPPLY	2	6EP1332-1SH71
6.	S7-1200 CPU1212C	1	6ES7212-1AD30-0XB0
7.	S7-1200 CPU1214C	1	6ES7214-1AE30-0XB0
8.	PC/PG	1	
9.	STEP 7 V5.4	1	6ES7810-4CC08-0YA5
10.	STEP 7 V5.4 Service Pack 5	1	ID Number:36184684
11.	STEP 7 BASIC V10.5	1	6ES7822-0AA00-0YA0
12.	STEP 7 Basic V10.5 Service Pack 2	1	ID Number:39741113

2 Automation Solution

Both the S7-1200 and the S7-300/400 offer T communication blocks for open TCP/IP communication:

- TCON, TSEND, TRCV and TDISCON (with manual connecting and disconnecting)

The S7-1200 additionally offers T communication blocks with integrated connecting and disconnecting:

- TSEND_C and TRCV_C

The following protocols are supported for data exchange between S7-1200 and S7-300/400 via open TCP/IP communication:

- TCP native
- ISO-on-TCP (dynamic data length transfer)

Communication blocks with manual connecting and disconnecting are selected for both the S7-300 and the S7-1200:

- TCON for connecting
- TSEND for sending data
- TRCV for receiving data
- TDISCON for disconnecting

“ISO-on-TCP” is selected as protocol.
In the OSI model, “ISO-on-TCP” is based on TCP and offers the advantage of a message-oriented principle of operation, which is particularly useful for communication between SIMATIC systems.

The connection is configured in STEP 7 V5.4 using the Open Communication Wizard (OC Wizard).
The respective connection partner is identified by the IP address.
The OC Wizard reserves a connection resource and creates a relevant connection data block. The IP address of the partner is stored in this data block.

The maximum number of connections for open IE communication depends on the used CPU. For the used CPU 315-2 PN/DP, max. 8 simultaneous connections can be established using “ISO-on-TCP”.

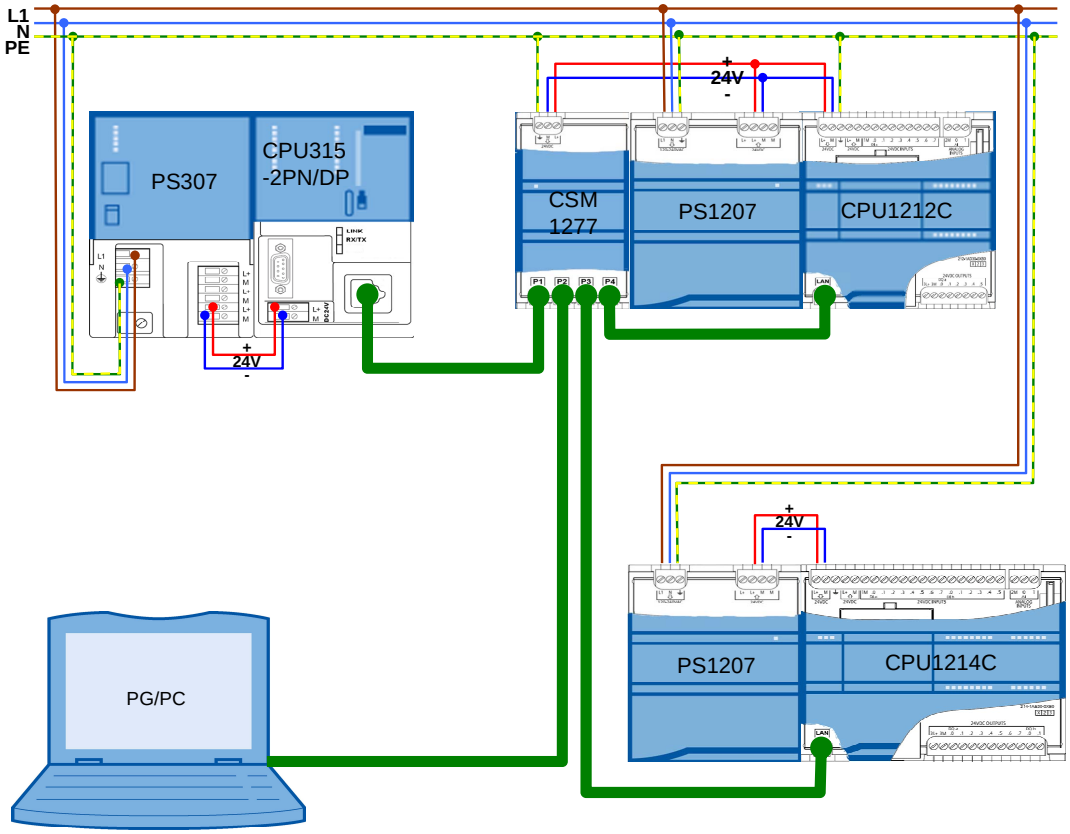
By changing the IP address in the connection data block, data can be successively exchanged with more than 8 different communication partners via the same connection resource.

“ISO-on-TCP” allows to transfer up to 8,192 bytes per job.

2.1 Connection diagram

For the list of components, please refer to chapter 1.2.

Figure 2-1



2.2 Program structure

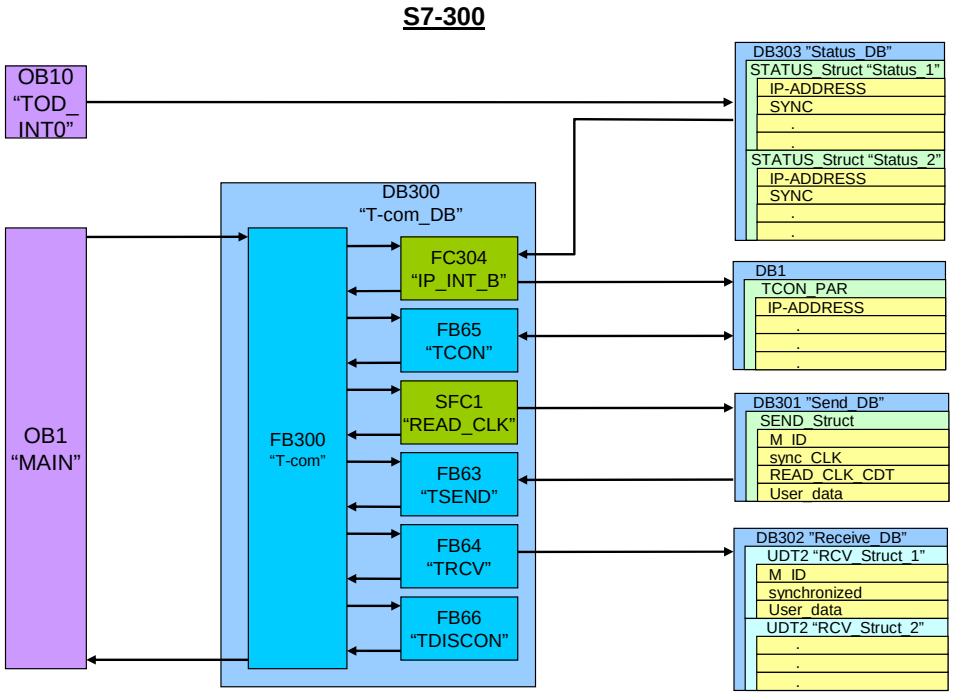
This chapter describes the program structure of the example at a functional and data block level of the automation system.

2.2.1 S7-300 block structure

Representation

Figure 2-2 shows the call hierarchy of the used blocks and the access to the data blocks used for the S7-300 master.

Figure 2-2



Description

In the hardware configuration of the CPU 315-2PN/DP, the OB10 “TOD_INT0” time-of-day interrupt is enabled and set to execute “every day”.

The DB303 “Status_DB” data block contains the IP address and control and status information from all S7-1200 slaves in the form of the “STATUS” data structure. Aside from the IP address and the “SYNC” synchronization request bit, this data structure includes information on the analysis of communication errors.

OB1 “MAIN” cyclically calls the FB300 “T-com” function block with its instance DB300 “T-com_DB”.

FC304 reads the IP address of the first slave from status DB 303 and writes it to the DB1 connection data block of the “TCON_PAR” data type.

FB65 “TCON” establishes a connection to the IP address stored in the connection data block.

When executing the OB10 “TOD_INT0” time-of-day interrupt, all “SYNC” synchronization request bits are set. Time-of-day synchronization, however, can also be executed individually for each slave using the variable table.

When setting the “SYNC” synchronization request for the first slave in DB303 “Status_DB”, the PLC time is read with the aid of SFC1 “READ_CLK” and stored in the DB301 “Send_DB” send data block together with the synchronization request.

The data type of the read “READ_CLK_CDT” time is DATE_AND_TIME.

The FB300 function block calls the FB63 “TSEND” send block. This block transfers the contents of the DB301 send data block to the first slave. Aside from the time-of-day synchronization information, “User_data” and an “M_ID” (message ID) are also transferred.

FB64 “TRCV” waits for data from slave 1 and stores the contents in the respective UDT2 “RCV_Struct_1” data type in DB 302 “Receive_DB”.

The received “M_ID” message ID is then compared to the sent message ID. Any discrepancy is stored in the status DB in the relevant “Status_1” data structure for slave 1.

After successful synchronization of slave 1 (signaled by the “synchronized” variable), the “SYNC” synchronization request bit is reset in “Status_1” of the status DB.

FB66 “TDISCON” disconnects the connection to slave 1.

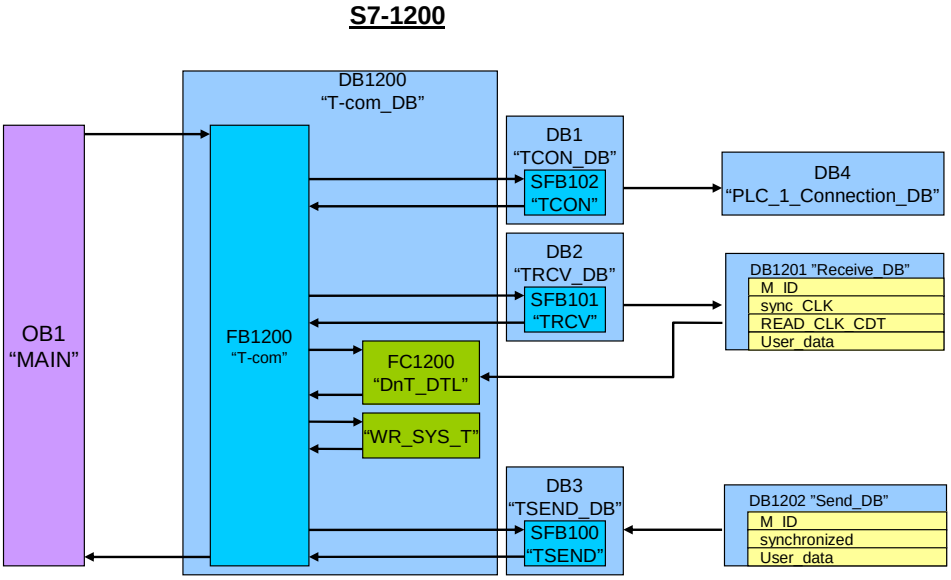
The “M_ID” message ID is increased and data exchange with slave 2 is handled in the same way.

2.2.2 S7-1200 block structure

Representation

Figure 2-3 shows the call hierarchy of the used blocks and the access to the data blocks used for the S7-1200 slaves.

Figure 2-3



Description

OB1 “MAIN” cyclically calls the FB1200 “T-com” function block with its instance DB1200 “T-com_DB”.

In the first cycle, FB1200 “T-com” calls SFB102 “TCON” with its instance DB 1. TCON receives the identification of the master from DB 4 “PLC_1_Connection-DB”. Once the connection has been established, it is maintained.

SFB101 “TRCV” with its instance DB 2 waits for data from the master and stores the contents in DB1201 “Receive_DB”.

When a “sync_CLK” synchronization request is issued, FB1200 “T-com” calls the FC1200 “DnT_DTL” function. This function converts the “READ_CLK_CDT” time of the S7-300 master of the DATE_AND_TIME type to the DTL data type. The “WR_SYS_T” function is used to write the converted time to the S7-1200 system time. After successful time-of-day synchronization, the “synchronized” bit is set in DB1202 “Send_DB”.

The “M_ID” message ID received from DB 1201 “Receive_DB” is mirrored to the DB 1202 “Send_DB” send data block.

With the aid of SFB100 “TSEND”, FB1200 “T-com” sends the contents of the DB1202 “Send_DB” send data block with its instance DB 3 to the master.

After successful send acknowledgement, slave 1 waits for new data from the master via TRCV and repeats the data exchange.

2.3 Used blocks

The following tables provide an overview of the blocks used on the master and slave side.

2.3.1 Master (S7-300)

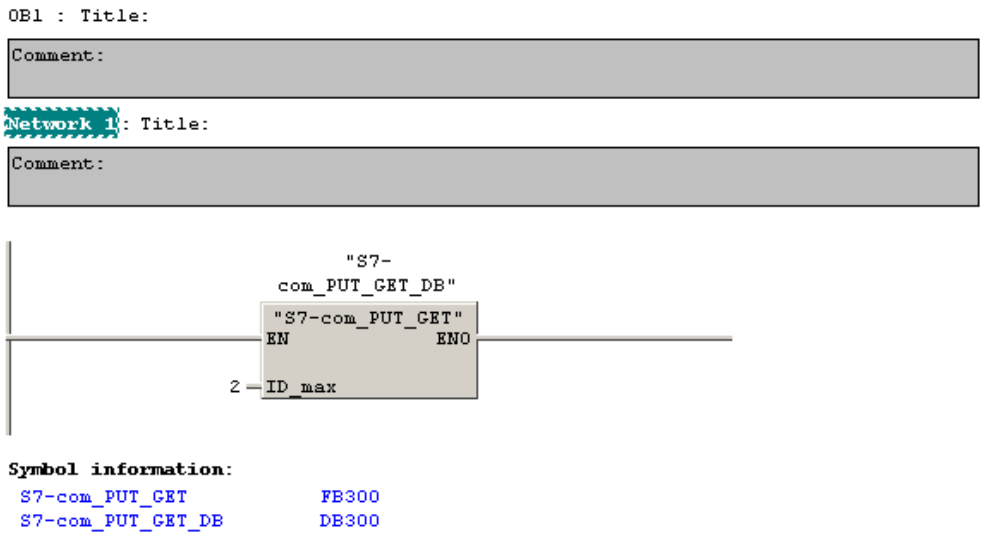
Table 2-1

Object name	Symbolic name	Description
OB1	MAIN	Cyclic organization block
OB10	TOD_INT0	Time-of-day interrupt
FB63	TSEND	T communication block for sending data
FB64	TRCV	T communication block for receiving data
FB65	TCON	T communication block for connecting
FB66	TDISCON	T communication block for disconnecting
FB300	T-com	Function block for deterministic data exchange with several slaves via the T communication blocks
FC301	read_bit	Function to read out the value of a bit using a pointer (used in FB300)
FC302	reset_bit	Function to reset a bit using a pointer (used in FB300)
FC303	output_bit	Function to output the value of a bit using a pointer (used in FB300)
FC304	IP_INT_B	Function to read out the octets of an IP address in integer format and to convert them to byte format (used in FB300)
DB300	T-com_DB	Multi-instance data block for FB300, FB63 to FB66
DB301	Send_DB	Send data block for FB15
DB302	Receive_DB	Receive data block for FB14
DB303	Status_DB	Status data block for all slaves
UDT2	RCV_STRUCT	Data type structure for receiving slave data

T-com (FB300)

The function block for deterministic data exchange with several slaves via the T communication blocks is called cyclically in OB1.

Figure 2-4



DB300 has been selected as an instance data block. It also contains the instances for the TCON, TSEND, TRCV and TDISCON T communication blocks.

As the only input, the maximum number of slaves “max” must be specified.

The relevant slave is identified by the “index” variable. Data exchange with the slaves is sequential.

The following static variables of FB300 provide configuration options via the initial value or the status evaluation of SFC1 “READ_CLK”.

Table 2-2

Name	Data type	Description
Receive_DB	Int	Number of the receive data block
RCV_STRUCT_size	Int	Size of the UDT2 receive data structure in bytes
Status_DB	Int	Number of the status data block
Status_size	Int	Size of the status data structure in bytes
TIMEOUT	S5Time	Maximum wait time until the step sequence in FB300 continues automatically
READ_CLK_ERROR	Bool	Error output of the READ_CLK block (SFC1)
READ_CLK_RET_VAL	Int	Status of the READ_CLK block (SFC1)

Status_DB (DB303)

The status DB consists of the two STATUS data structures for the two S7-1200 slaves with which the CPU 315-2PN/DP communicates. The structure includes 18 bytes and looks as follows:

Table 2-3

Name	Data type	Description
IP_ADDR	ARRAY[1..4] of INT	IP address of the slave (4 octets)
SYNC	Bool	Time-of-day synchronization request
TCON_ERROR	Bool	Error message of the TCON communication block
TCON_TIMEOUT	Bool	Execution time of the TCON communication block exceeded
TSEND_ERROR	Bool	Error message of the TSEND communication block
TRCV_ERROR	Bool	Error message of the TRCV communication block
TRCV_TIMEOUT	Bool	Execution time of the TRCV communication block exceeded
TDISCON_ERROR	Bool	Error message of the TDISCON communication block
M_ID_UNEQUAL	Bool	Sent and received M_IDs differ
TCON_ERROR_STATUS	Word	Status of the TCON block when the last error occurred
TSEND_ERROR_STATUS	Word	Status of the TSEND block when the last error occurred
TRCV_ERROR_STATUS	Word	Status of the TRCV block when the last error occurred
TDISCON_ERROR_STATUS	Word	Status of the TDISCON block when the last error occurred

2.3.2 Slave (S7-1200)

Table 2-4

Object name	Symbolic name	Description
OB1	Main	Cyclic organization block
FC1200	DnT_DTL	Function to convert the DATE_AND_TIME data type to the DTL data type
FB1200	T-com	Function block for data exchange with an S7-300 master via the T communication blocks
DB4	PLC_1_Connection_DB	Connection data block for TCON
DB1200	T-com_DB	Instance data block for T-com
DB1201	Receive_DB	Data block for receiving from the master
DB1202	Send_DB	Data block for sending to the master
DB1	TCON_DB	Instance data block for TCON
DB2	TRCV_DB	Instance data block for TRCV
DB3	TSEND_DB	Instance data block for TSEND

2.3.3 Data consistency

DB301 and DB1201

The send block of the master and the receive block of the slave must have the same length and structure. In the application example, they consist of 160 bytes and have the following structure:

Table 2-5

Name	Data type	Description
M_ID	Int	Message ID
sync_CLK	Bool	Time-of-day synchronization request
READ_CLK_CDT	DATE_AND_TIME or array of 8 bytes	Synchronization time of master (S7-300)
User_data	Array of 148 bytes	User data (S7-300 -> S7-1200)

DB302 and DB1202

The RCV_STRUCT receive structure (UDT2) of the master and the send block of the slave must be identical. Receive DB 302 consists of the two receive structures for the two S7-1200 slaves. The UDT2 receive structure or send DB 1202 consist of 160 bytes and look as follows:

Table 2-6

Name	Data type	Description
M_ID	Int	Mirrored message ID for acknowledgement
synchronized	Bool	Time-of-day synchronization acknowledgement
User_data	Array of 156 bytes	User data (S7-1200 -> S7-300)

The “User_data” can be changed individually. However, the data structure must be identical on the sending and receiving side.
Sequential processing of the send and receive jobs ensures program data consistency.
Status DB 303 offers the option of directly influencing communication errors.
Due to continuous data exchange between master and slaves, data consistency can be ensured for only one cycle.
Consequently, the user has to write consistent data to the send data blocks or read consistent data from the receive data blocks within one cycle.

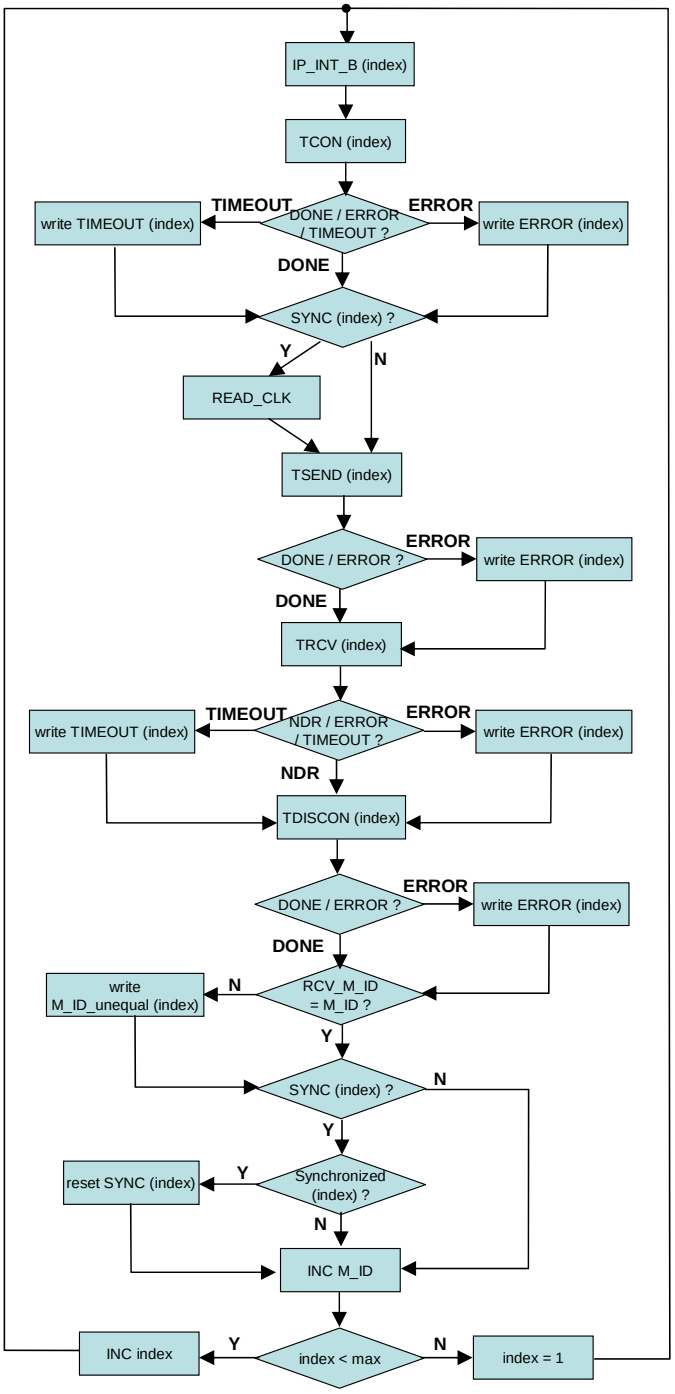
2.4 Program flow

2.4.1 Program flow in the master controller

Flowchart

The flowchart below shows the program flow in the master controller. The functionality is combined in FB300 “T-com”, which is called cyclically by OB1. FB300 is realized as a step sequence.

Figure 2-5



Flowchart description

The respective slave with which data is exchanged is identified by the “index” variable.

Starting with “index” = 1, the IP address of the first slave is read from the status DB using the “IP_INT_B” function and written to the connection data block.

TCON is used to send a connection request to this slave. Since the step sequence can only be continued after acknowledging the connection request, a maximum execution time, “TIMEOUT”, elapses.

This time or the “Error” feedback is output in the status DB depending on the index (slave).

Subsequently, the “SYNC” synchronization request is read from the status information for this slave. Depending on the request, the system time (“READ_CLK”) is read and written to the send data block.

The “TSEND” send data block is used to transfer the contents of the send data block to the slave. Aside from the time-of-day synchronization information, an “M_ID” message ID is also transferred.

In the event of an “ERROR” message of the “TSEND” send block, the error information is written to the status structure of the “index” slave in the status DB.

The “TRCV” receive block is used to receive the data of the “index” slave and to write it to the “index” receive structure in the receive DB.

Once the maximum execution time, “TIMEOUT”, is exceeded or if there is an “ERROR” message of the “TRCV” receive block, the error information is written to the status structure of the “index” slave in the status DB.

Subsequently, the connection to the slave is disconnected using the “TISCON” block. Errors when disconnecting are also output in the status DB.

Based on the receive data, the “RCV_M_ID” message ID mirrored by the slave is compared to the sent “M_ID”. If the IDs differ, this information is stored in the status structure of the “index” slave in the status DB (“M_ID_unequal”).

When a “SYNC” synchronization request of the “index” slave is issued, successful synchronization is checked using the receive data of the “index” slave (“synchronized”). When the result is positive, the “SYNC” synchronization request is reset for the “index” slave. Otherwise, time-of-day synchronization is repeated with this slave during the next communication.

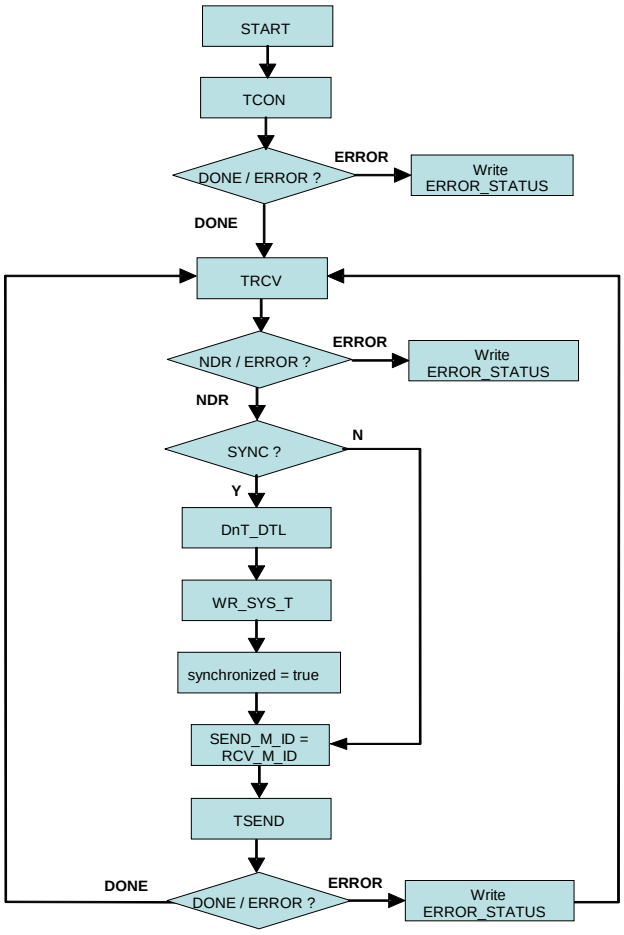
The message ID is increased (“INC M_ID”) and the “ID” is compared to the “ID_max”, i.e. the maximum number of slaves. The index is increased (“INC index”) until “max” is reached. Otherwise, the index is set to the initial value (“index = 1”).

2.4.2 Program flow in the slave controller

Flowchart

The flowchart below shows the program flow on the slave side. The functionality is combined in FB1200 “T-com”, which is called cyclically by OB1. FB1200 is realized as a step sequence.

Figure 2-6



Flowchart description

When initializing, TCON is used to send a connection request to the master. If an “ERROR” feedback is received, the status information is kept in the “TCON_ERROR_STATUS” variable. In the event of a positive acknowledgement of the connection establishment, “DONE”, the slave waits for data from the master via the “TRCV” block.

If an “ERROR” feedback is received from the receive block, the status information is kept in the “TRCV_ERROR_STATUS” variable.
When receiving new data, “NDR”, the received “SYNC” synchronization request of the master is queried.

When there is a request, the “DnT_DTL” function is used to convert the received system time of the DATE_AND_TIME data type to DTL format and “WR_SYS_T” is used to write it to the S7-1200 system time. Subsequently, successful synchronization is signaled in the send data of the slaves via the “synchronized” bit.

The “RCV_M_ID” message ID received from the master is mirrored in the “SEND_M_ID” send data. Using “TSEND”, the slave sends the send data to the master. If an “ERROR” message occurs, the “TSEND_ERROR_STATUS” error status information is stored. Finally, the slave is ready to receive again and waits for new data from the master.

When the master disconnects, the reserved connection resource to the master remains active. A new connection request using “TCON” is thus only necessary after a restart.

3 Configuration

3.1 Installing the hardware and software

3.1.1 Installing and wiring the hardware

Table 3-1

No.	Instruction	Note/screen shot
13.	Mount the S7-1200 modules onto a standard DIN rail.	
14.	Mount the S7-300 modules onto an S7-300 mounting rail.	
15.	Use an RJ45 Ethernet cable to connect the controllers and your programming unit to the CSM 1277.	See chapter “Connection diagram”
16.	Connect all ground connections to ground.	See chapter “Connection diagram”
17.	Connect the controllers to the voltage supply.	See chapter “Connection diagram”
18.	Insert the MICRO MEMORY CARD into the CPU 315-2PN/DP.	See Table 1-1

3.1.2 Installing the software

Table 3-2

No.	Instruction	Note/screen shot
1.	Install STEP 7 BASIC V10.5 on your programming unit.	See Table 1-1
2.	Install Service Pack 2 for STEP 7 BASIC V10.5 on your programming unit.	See Table 1-1
3.	Install STEP 7 V5.4 on your programming unit.	See Table 1-1
4.	Install Service Pack 5 for STEP 7 V5.4 on your programming unit.	See Table 1-1

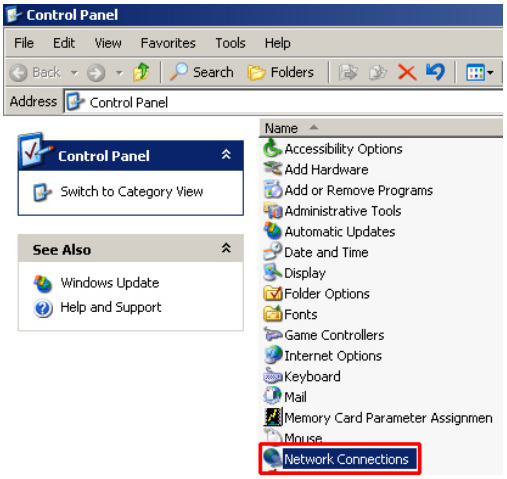
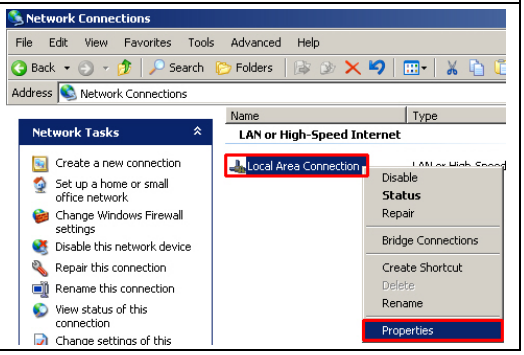
3.2 Configuring the hardware and network

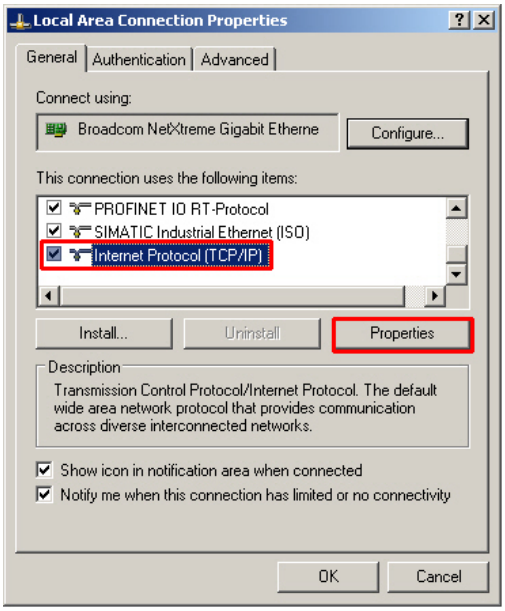
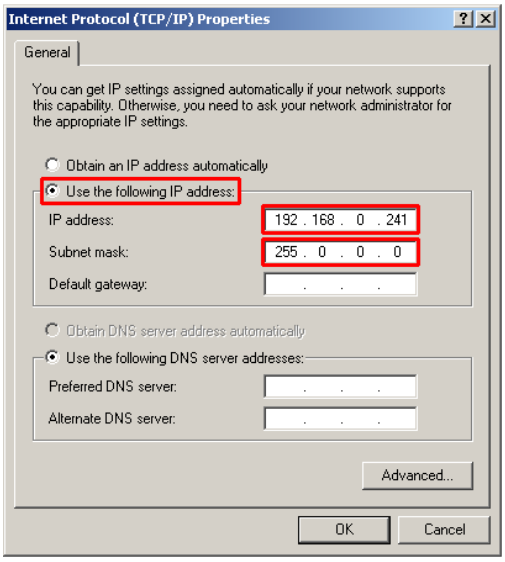
3.2.1 Assigning the IP address of the PG/PC

An IP address that is in the same subnet as the CPUs must be assigned to your PG/PC. The IP addresses of the individual nodes are shown in Figure 1-2.

To assign the IP address for your network card in the Windows XP operating system, proceed as follows:

Table 3-3

No.	Instruction	Note/screen shot
1.	Open the Windows Control Panel and select “Network Connections”.	
2.	Select the network card to be used and right-click to open “Properties”.	

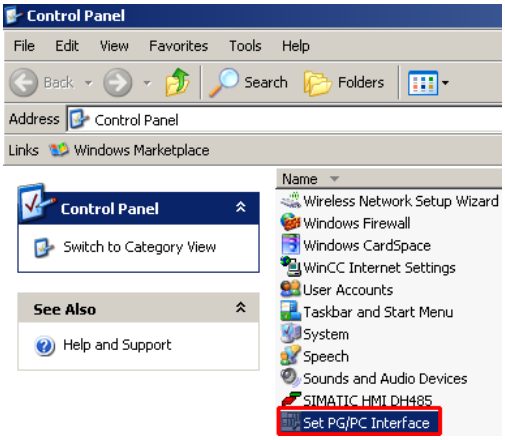
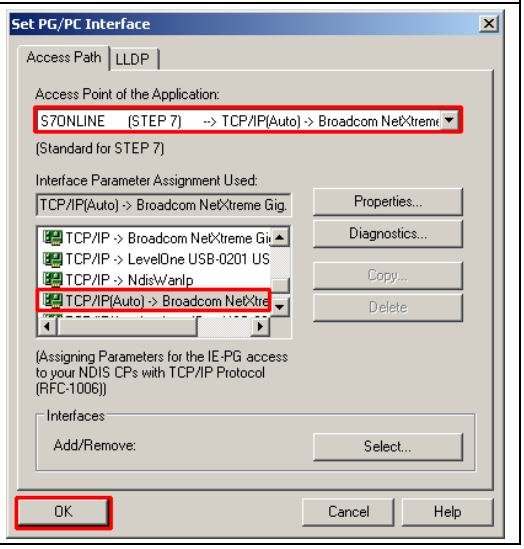
No.	Instruction	Note/screen shot
3.	Select “Internet Protocol (TCP/IP)” and click on the “Properties” button.	
4.	• Select “Use the following IP address”. • Enter the following IP address: “192.168.0.241” (see Figure 1-2). • Enter the following subnet mask: “255.0.0.0” (see Figure 1-2). • Click on “OK” to confirm the settings.	

3.2.2 Configuring the S7-300

Configuring the PG/PC interface

To download the project and to communicate with the CPU 315-2PN/DP online, the PG/PC interface must be set as follows:

Table 3-4

No.	Instruction	Note/screen shot
1.	• In the Windows Control Panel, select the “Set PG/PC Interface” dialog box.	
2.	• Select “S7ONLINE (STEP 7)” as access point of the application. • Select “TCP/IP(Auto)” as the interface parameter assignment used for your network card. • Click on “OK” to confirm the settings.	

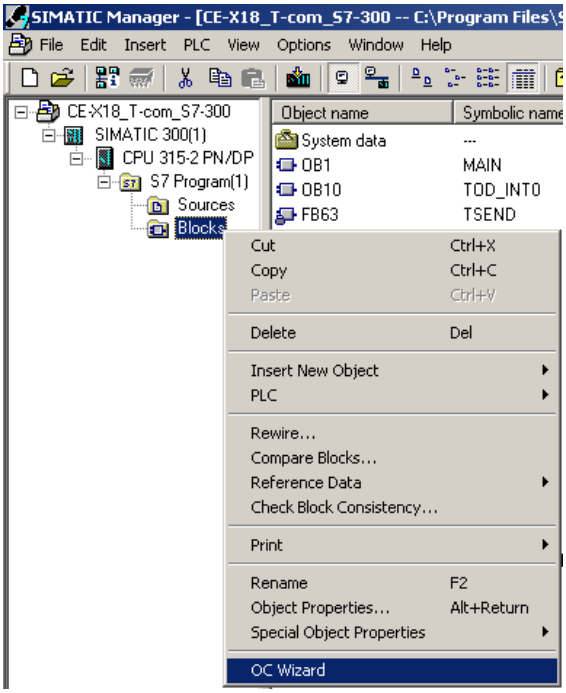
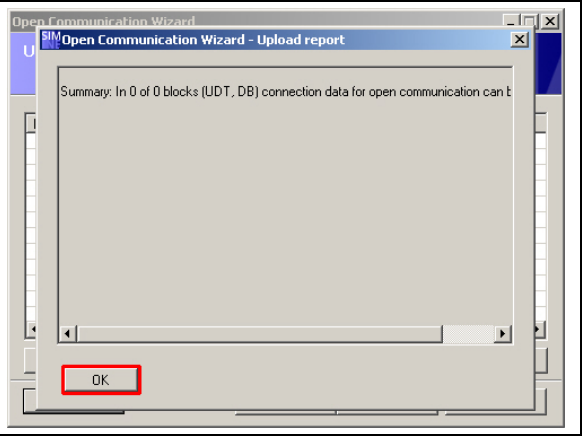
Creating a connection

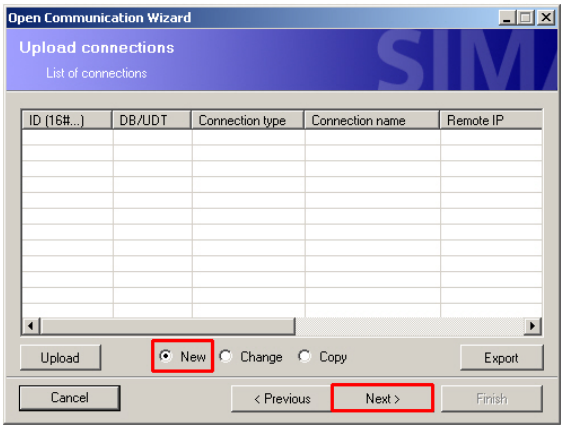
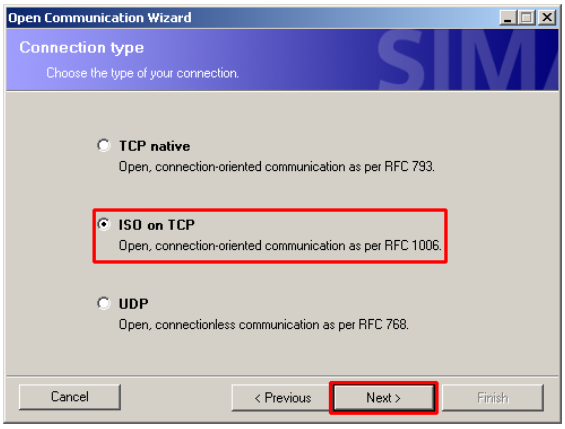
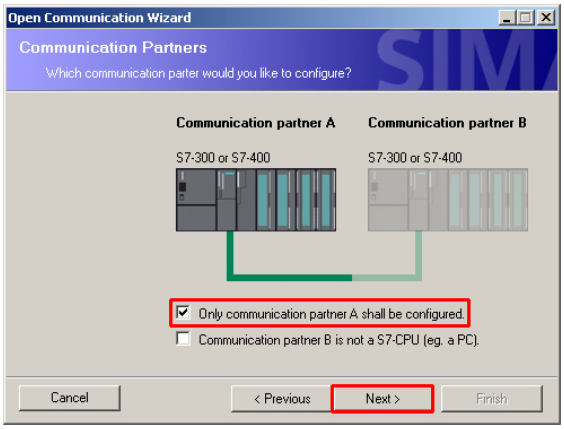
For open IE communication, the connection parameters are stored in a connection data block. A connection ID is assigned to each connection data set. The TCON connection block accesses this connection data block. The T communication blocks then use the connection ID to assign it to the connection data set.

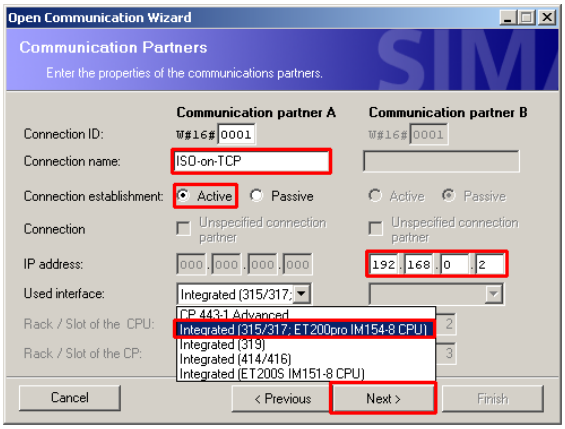
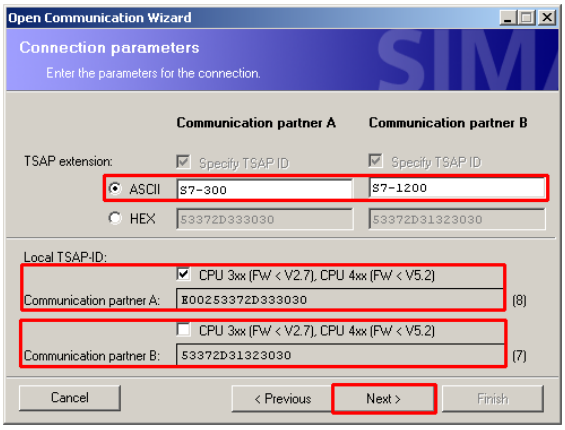
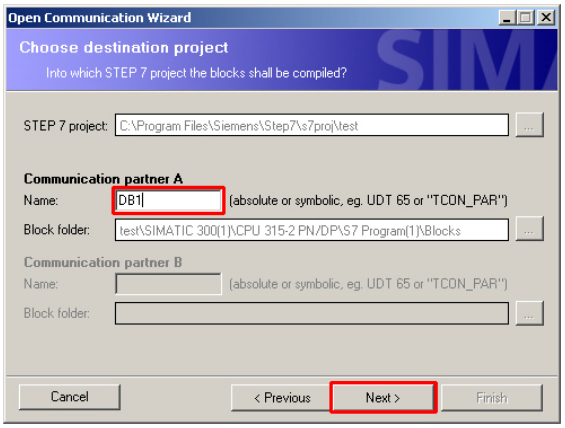
To configure the connection, the S7-300 offers the Open Communication Wizard.

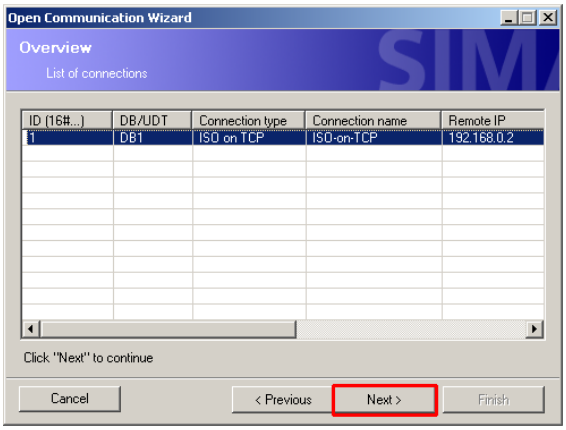
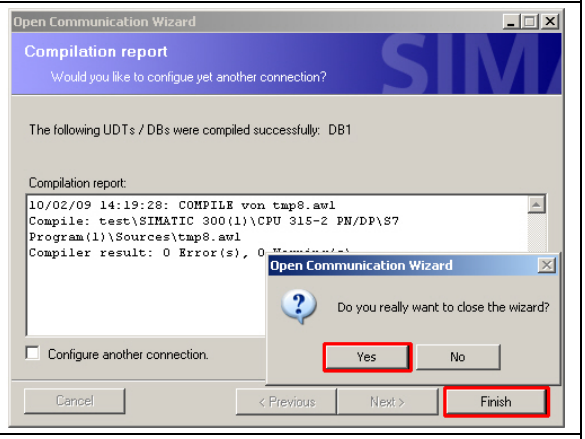
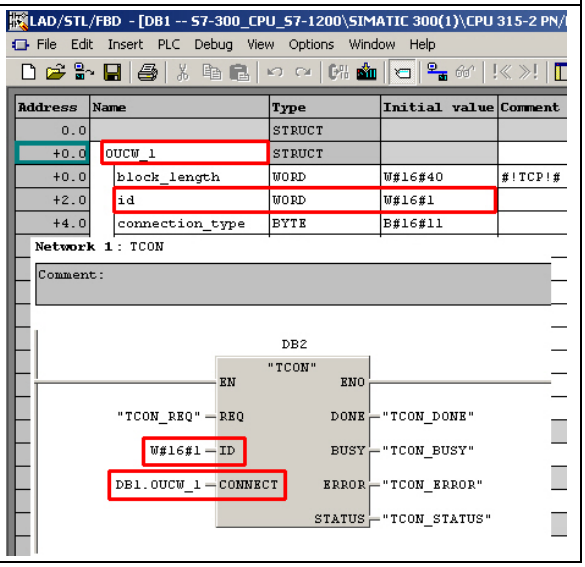
The following sections describe the configuration of an open IE connection using the OC Wizard.

Table 3-5

3.	In the SIMATIC Manager in the S7 program of your controller, select the “Blocks” folder and right-click to open the OC Wizard.	
4.	In the project, the OC Wizard searches for existing connection data blocks for open communication and displays the search result. Click on “OK” to confirm the display.	

5.	Select “New” and “Next >” to create a new connection.	
6.	Select the “ISO on TCP” protocol version and click on “Next>”.	
7.	Select “Only communication partner A shall be configured” and click on “Next >”.	

8.	• Assign a specific name to the connection. • Select “Active” for the connection establishment of communication partner A (S7-300). • Assign the IP address of communication partner B (S7-1200). • Select the integrated IE interface of your CPU 315-2PN/DP as the interface to be used.	
9.	• Assign TSAP IDs for both communication partners. • In this case, “S7-300” has been selected for partner A and “S7-1200” in ASCII characters has been selected for partner B. • For S7-300 CPU firmware revision levels < V2.7, partner A must be checked. The TSAP ID is then prefixed with the hexadecimal ID E002. • Partner B (the S7-1200) must not be checked.	
10.	As a name for the connection data block, enter a free DB (here: “DB1”).	

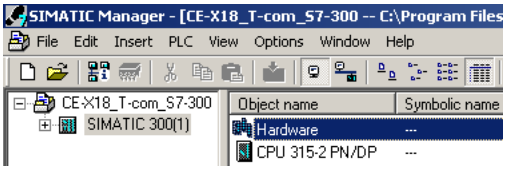
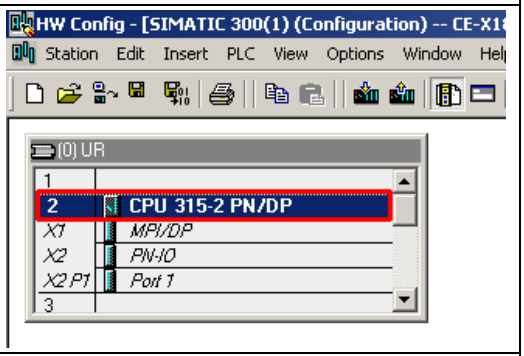
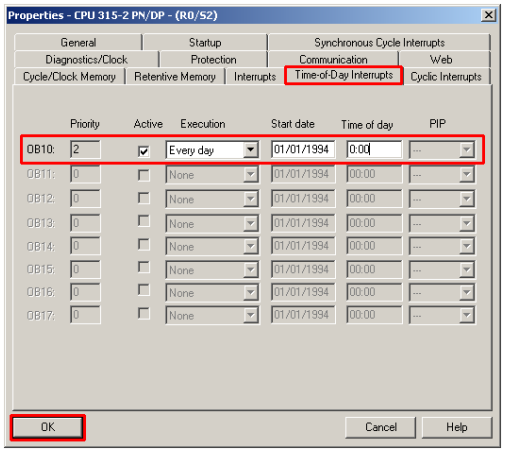
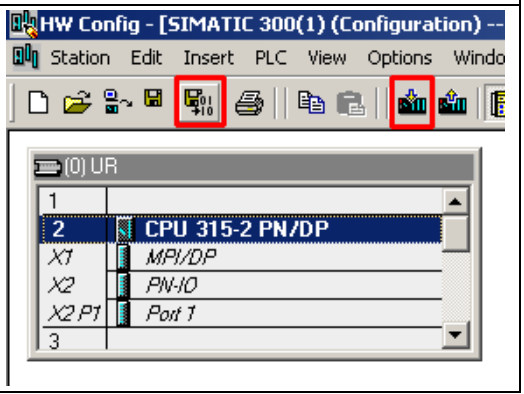
11.	Select “Next>” to confirm the overview of connections	
12.	Select “Finish” to confirm the compilation report and click on “Yes” to confirm the following message.	
13.	The connection data structure of the generated DB1 connection data block starts with the symbolic name “OUCW_1”. • Thus enter “DB1.OUCW_1” as an address for the connection data on the CONNECT parameter of the TCON connection block. • Parameterize the “ID” input with the assigned “W#16#1” connection ID. This connection ID also has to be specified for the other T communication blocks: TSEND, TRCV and TDISCON.	

Setting the synchronization time

The CPU 315-2PN/DP is the time-of-day master for synchronizing the S7-1200 slaves. The interval for time-of-day synchronization can be set by configuring the time-of-day interrupt. Aside from this automatic synchronization, each slave can also be synchronized manually using the variable table.

Note UTC time (Universal Time Coordinated) is the system time for both the S7-300 and the S7-1200.

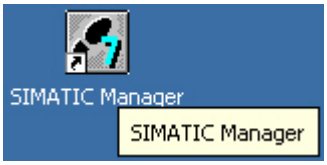
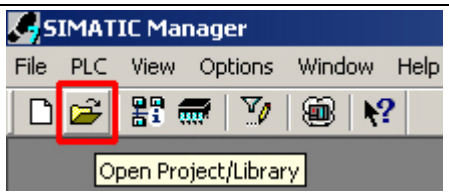
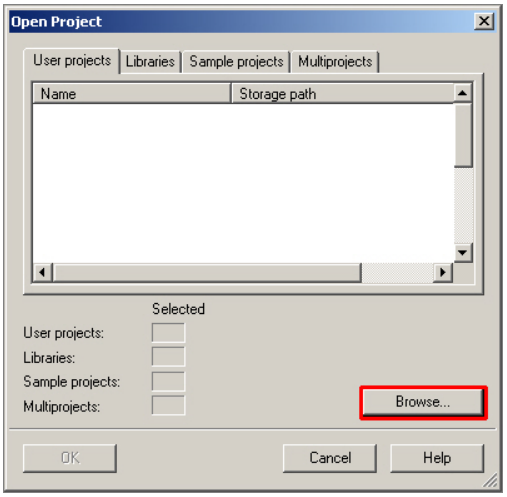
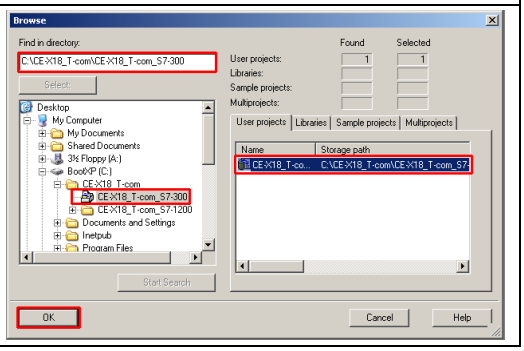
Table 3-6

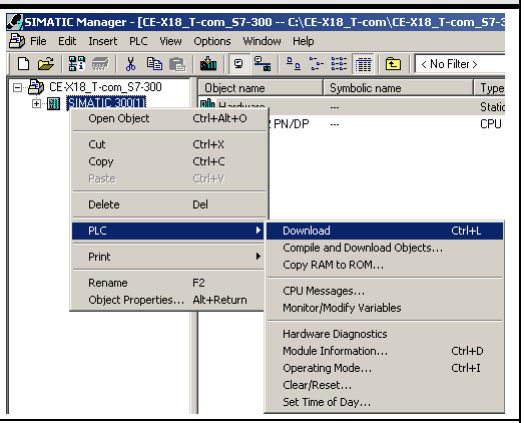
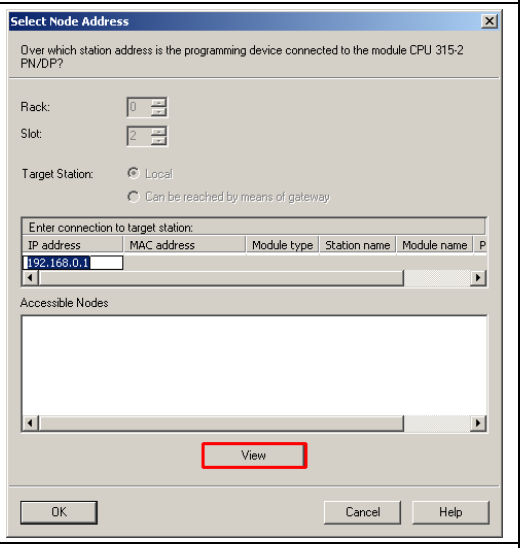
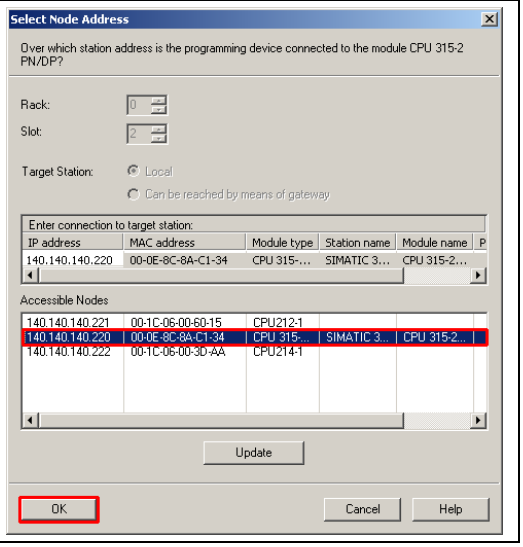
No.	Instruction	Note/screen shot
1.	In the “CE-X18B_Master_v1d2” STEP 7 project, double-click on the hardware configuration to open it.	
2.	Double-click on the CPU 315-2PN/DP to open the Properties.	
3.	• In the “Time-of-Day Interrupts” tab, activate OB10. • For Execution, select “Every day”. • Specify the time of day when the slaves are to be synchronized every day. • Click on “OK” to confirm the settings.	
4.	• Click on the “Save and Compile” button. • Finally, you have to download the changed hardware configuration to the CPU.	

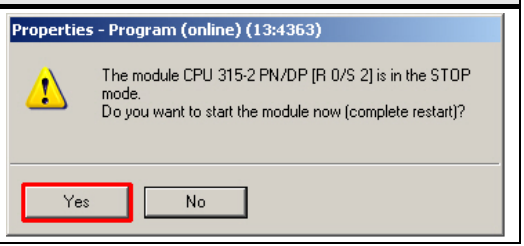
Downloading the master project to the controller

Unzip the “CE-X18B_Master_v1d2.zip” sample program to any directory on your hard drive.
The unzipped file includes the “CE-X18B_Master_v1d2” project for the CPU 315-2PN/DP.

Table 3-7

No.	Instruction	Note/screen shot
1.	• Open the “SIMATIC Manager”.	
2.	• Click on the “Open Project/Library” button.	
3.	• Click on the “Browse…” button.	
4.	• Navigate to the folder to which you have unzipped the sample project and select the “CE-X18B_Master_v1d2” S7-300 project. • Click on “OK” to confirm the opening of the project.	

No.	Instruction	Note/screen shot
5.	The SIMATIC Manager opens the S7-300 project. • Select the “SIMATIC 300(1)” station. • Right-click and select “PLC” -> “Download” to download the entire project.	
6.	The window for selecting the node address opens. • Click on the “View” button to display a list of accessible nodes.	
7.	• From the list of accessible nodes, select the S7-300 CPU (identified by the MAC address). • Click on “OK”.	

No.	Instruction	Note/screen shot
8.	• Once the project has been successfully downloaded, set the CPU to “RUN” mode.	

3.2.3 Configuring the S7-1200

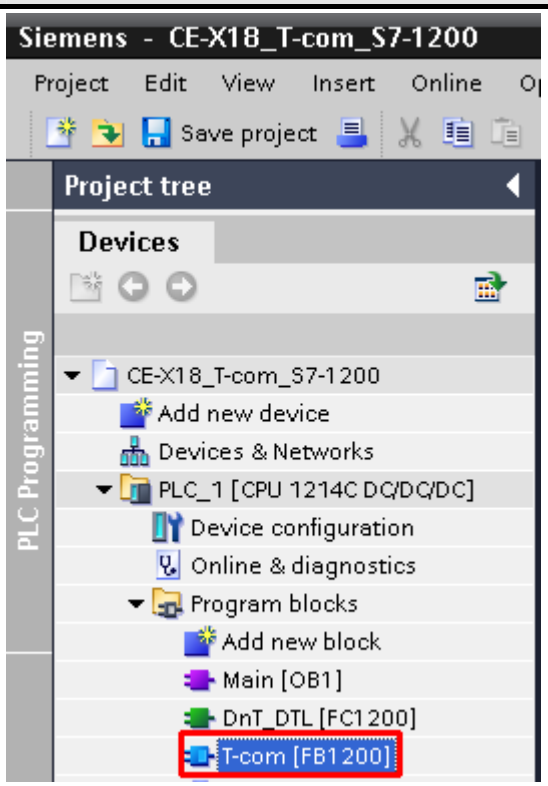
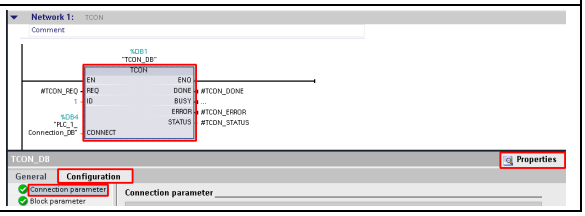
Creating a connection

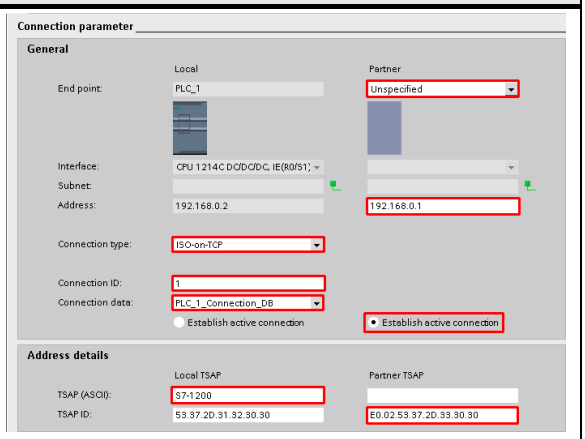
The connection parameters are stored in a connection data block. A connection ID is assigned to each connection data block. The TCON connection block accesses this connection data block. The TRCV, TSEND and TDISCON T communication blocks then access the connection parameters via the connection ID.

To configure the connection, the S7-1200 offers the Connection Wizard.

The following sections describe the configuration of an open IE connection using the Connection Wizard.

Table 3-8

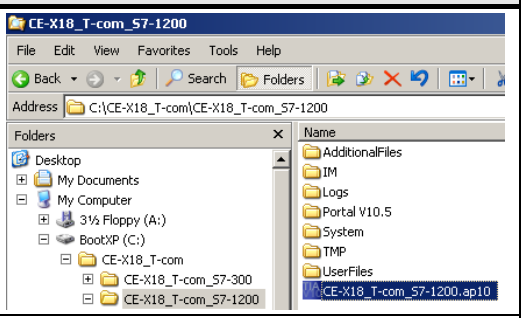
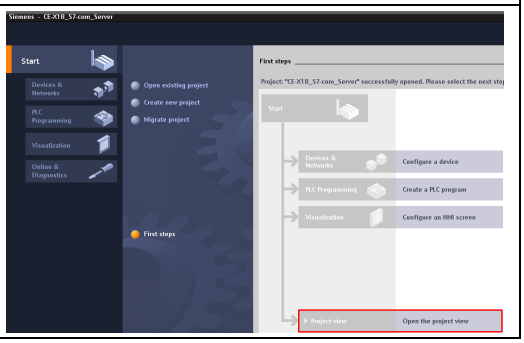
1.	• In the “CE-X18B_Slave_v1d2.ap10” project in the PLC for which the connection is to be created (here: “PLC_1”), open FB1200 “T-com”.	
2.	• Open network 1. • Select the “TCON” block. • Open the “Properties”. • In the “Configuration” tab, select “Connection parameter”.	

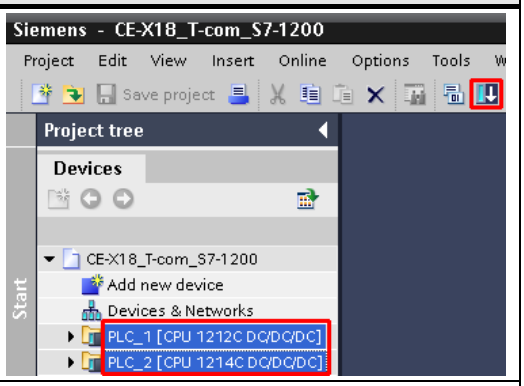
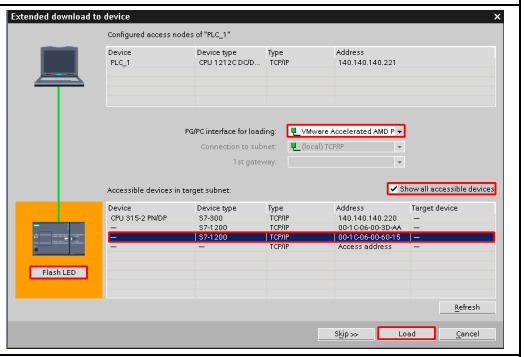
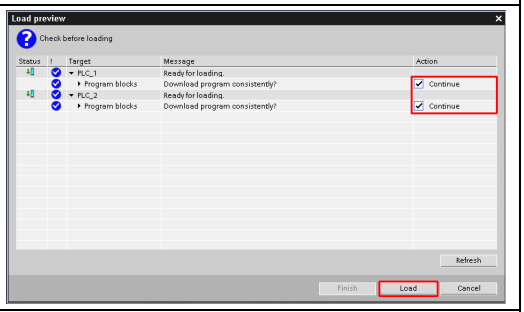
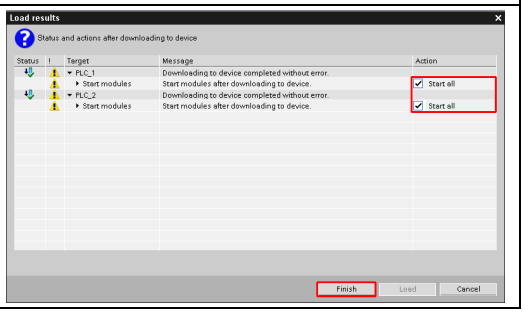
3.	• As an end point for the partner (S7-300), select “Unspecified”. • Enter the IP address of the partner. • Select the “ISO-on-TCP” connection type. • For connection ID, enter “1”. • In “Connection data”, select the symbolic name of the connection data block. • For the local TSAP, select the “S7-1200” ASCII string. • For the partner TSAP ID, select the hexadecimal string of communication partner A from step 9 in Table 3-5.	

Downloading the slave project to the controllers

Unzip the “CE-X18B_Slave_v1d2.zip” sample program to any directory on your hard drive.
The unzipped file includes the “CE-X18_ T-com_S7-1200” project for the two S7-1200 controllers.

Table 3-9

No.	Instruction	Note/screen shot
1.	• In Windows Explorer, navigate to the “CE-X18_ T-com_S7-1200.ap10” S7-1200 project and open it with a double-click.	
2.	The project opens in STEP 7 Basic. • Open the project view.	

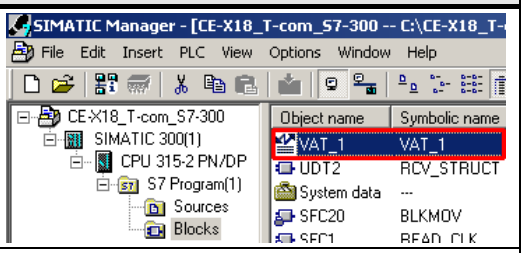
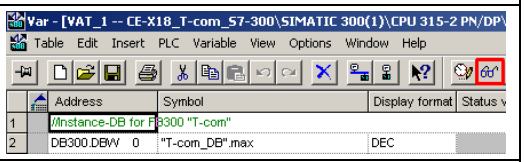
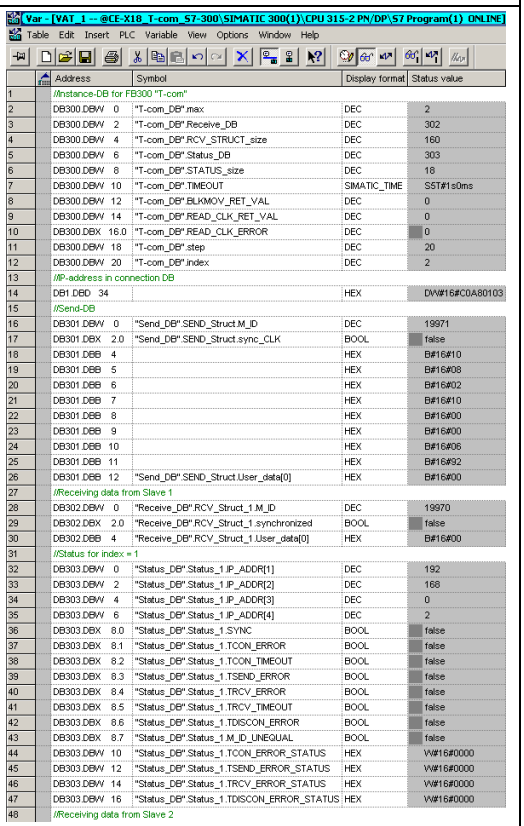
No.	Instruction	Note/screen shot
3.	• Select the two PLC folders “PLC_1 [CPU 1212C DC/DC/DC]” and “PLC_2 [CPU 1214C DC/DC/DC]”. • Click on the “Download to Device” button to download the entire projects to the controllers.	
4.	• Select the network card you are using. • Enable “Show all accessible devices”. • From the list of accessible devices, identify PLC_1 using the MAC address or “Flash LED”. • Select the desired controller and click on the “Load” button. Repeat these two steps to download PLC_2.	
5.	• Enable consistent downloading for both controllers. • Click on the “Load” button.	
6.	After transferring all program blocks to the controllers, a “Load results” window appears. • Check the “Start all” boxes to set both controllers to “Run” mode. • Click on the “Finish” button to finish the download.	

3.3 Activating online mode

To control and monitor communication, your PG/PC must be set to online mode for the S7-1200 and the S7-300 using the watch/variable table.

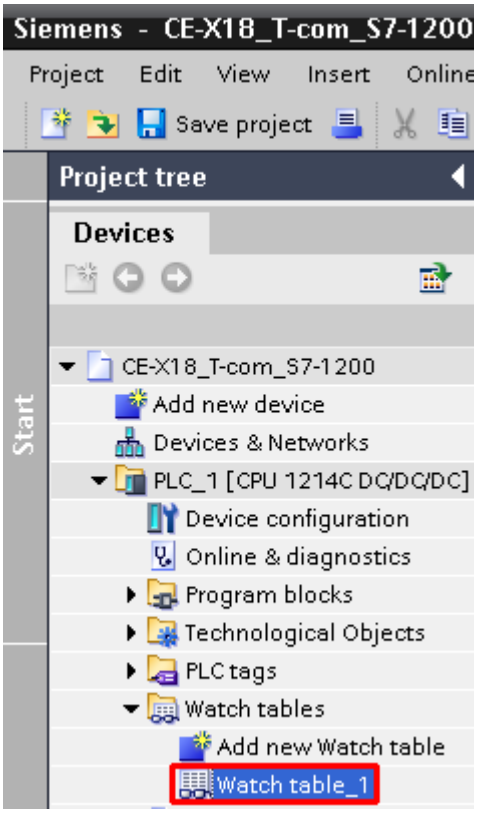
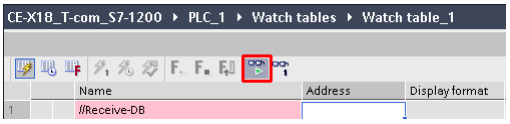
Activating the watch table for the S7-300 master

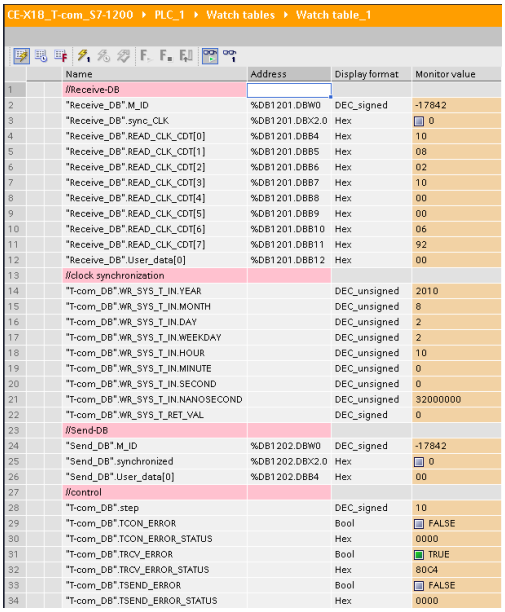
Table 3-10

No.	Instruction	Note/screen shot
1.	In the SIMATIC Manager in project name -> station name -> CPU -> “Blocks” container, select the “VAT_1” variable table.	
2.	Use the “Monitor variable” button to activate the variable table.	
3.	The variable table contains (row numbers in brackets): • Instance data block of FB300 (2-12) - Specified initial values (2-7) - Return values of the “BLKMOV” and “read system time” functions (8-10) - Step (11) - Index (12) • Hexadecimal IP address in the connection data block (14) • Send data block (16-26) - Message ID (16) - Synchronization request (17) - Master system time in DATE_AND_TIME format (18-25) - First user data byte (26) • Receive data of slave 1 (28-30) - Message ID (28) - Synchronization acknowledgement (29) - First user data byte (30) • Status information of slave 1 (32-47) - IP address (32-35) - Synchronization request (36) - Communication error analysis (37-47) • Receive data of slave 2 (49-51) • Status information of slave 2 (53-68)	

Activating the watch table for the S7-1200 slaves

Table 3-11

No.	Instruction	Note/screen shot
1.	• In the STEP 7 Basic project navigation, select “PLC_1” -> “Watch tables” -> “Watch table_1”.	
2.	• Use the “Monitor all” button to activate the watch table.	
3.	Repeat steps 1 and 2 for slave 2: • PLC_2 [CPU 1212C DC/DC/DC] • Watch table_2	

No.	Instruction	Note/screen shot
4.	Each watch table contains the following information (row numbers in brackets): • Receive data block (2-12) - Message ID (2) - Synchronization request (3) - Master system time in DATE_AND_TIME format (4-11) - First user data byte (12) • Time-of-day synchronization data (14-15) - Converted master system time in DTL format (14-21) - Return value of the “write system time” function (22) • Send data block (24-26) - Mirrored message ID (24) - Synchronization acknowledgement (25) - First user data byte (26) • Status information (28-34) - Step (28) - Communication error analysis (29-34)	

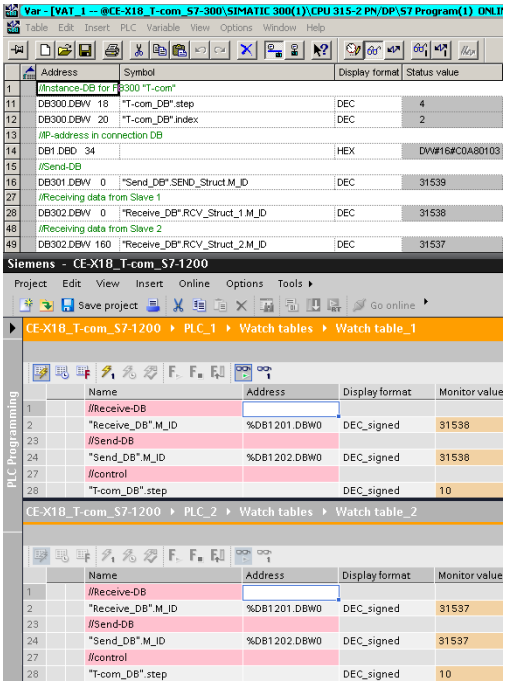
CE-X18_T-com_S7-1200 ▸ PLC_1 ▸ Watch tables ▸ Watch table_1

	Name	Address	Display format	Monitor value
1	//Receive-DB			
2	"Receive_DB".M_ID	%DB1201.DBW0	DEC_signed	-17842
3	"Receive_DB".sync_CLK	%DB1201.DBX2.0	Hex	0
4	"Receive_DB".READ_CLK_CDT[0]	%DB1201.DBB4	Hex	10
5	"Receive_DB".READ_CLK_CDT[1]	%DB1201.DBB5	Hex	08
6	"Receive_DB".READ_CLK_CDT[2]	%DB1201.DBB6	Hex	02
7	"Receive_DB".READ_CLK_CDT[3]	%DB1201.DBB7	Hex	10
8	"Receive_DB".READ_CLK_CDT[4]	%DB1201.DBB8	Hex	00
9	"Receive_DB".READ_CLK_CDT[5]	%DB1201.DBB9	Hex	00
10	"Receive_DB".READ_CLK_CDT[6]	%DB1201.DBB10	Hex	06
11	"Receive_DB".READ_CLK_CDT[7]	%DB1201.DBB11	Hex	92
12	"Receive_DB".User_data[0]	%DB1201.DBB12	Hex	00
13	//clock synchronization			
14	"T-com_DB".WR_SYS_T_IN.YEAR		DEC_unsigned	2010
15	"T-com_DB".WR_SYS_T_IN.MONTH		DEC_unsigned	8
16	"T-com_DB".WR_SYS_T_IN.DAY		DEC_unsigned	2
17	"T-com_DB".WR_SYS_T_IN.WEEKDAY		DEC_unsigned	2
18	"T-com_DB".WR_SYS_T_IN.HOUR		DEC_unsigned	10
19	"T-com_DB".WR_SYS_T_IN.MINUTE		DEC_unsigned	0
20	"T-com_DB".WR_SYS_T_IN.SECOND		DEC_unsigned	0
21	"T-com_DB".WR_SYS_T_IN.NANOSECOND		DEC_unsigned	32000000
22	"T-com_DB".WR_SYS_T_RET_VAL		DEC_signed	0
23	//Send-DB			
24	"Send_DB".M_ID	%DB1202.DBW0	DEC_signed	-17842
25	"Send_DB".synchronized	%DB1202.DBX2.0	Hex	0
26	"Send_DB".User_data[0]	%DB1202.DBB4	Hex	00
27	//control			
28	"T-com_DB".step		DEC_signed	10
29	"T-com_DB".TCON_ERROR		Bool	FALSE
30	"T-com_DB".TCON_ERROR_STATUS		Hex	0000
31	"T-com_DB".TRCV_ERROR		Bool	TRUE
32	"T-com_DB".TRCV_ERROR_STATUS		Hex	80C4
33	"T-com_DB".TSEND_ERROR		Bool	FALSE
34	"T-com_DB".TSEND_ERROR_STATUS		Hex	0000

3.4 Live demo

3.4.1 Cyclic sequence

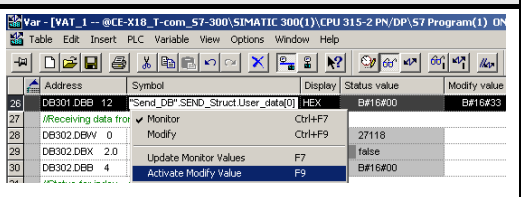
Table 3-12

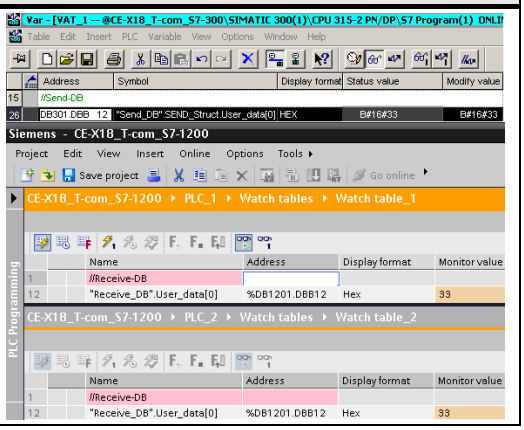
No.	Instruction	Note/screen shot
1.	• FB300 “T-com” of the master is called cyclically (which is indicated by the step display change in row 11). • The FB1200 “T-com” function blocks of the slaves are called cyclically (which is indicated by the step display change in rows 29). • The master continuously communicates with slaves 1 and 2 (indicated by the index change in row 12 and the hexadecimal IP address in row 14). • Message IDs with even numbers are sent to slave 1 where they are mirrored and received again. • Message IDs with uneven numbers are sent to slave 2 where they are mirrored and received again.	

3.4.2 User data transfer

S7-300 master -> S7-1200 slaves

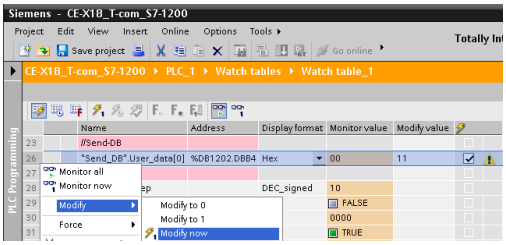
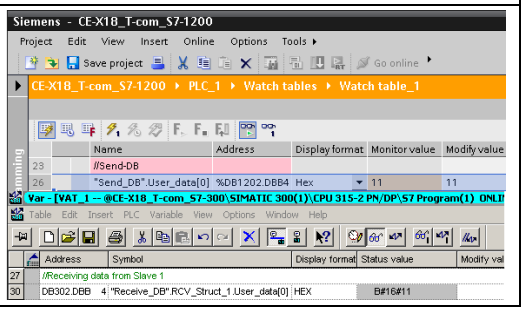
Table 3-13

No.	Instruction	Note/screen shot
1.	As an example of transferring user data from the master to the slaves, send byte 0 of the user data field in row 26 is to be changed: • Enter a value in the “Modify value” column in row 26. • Use F9 to apply the value.	

No.	Instruction	Note/screen shot
2.	The value is transferred to both slaves and written to receive byte 0 of the user data field in receive data block 1201 (as can be seen in rows 12 of the slave watch tables).	

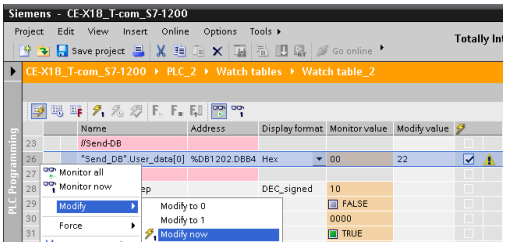
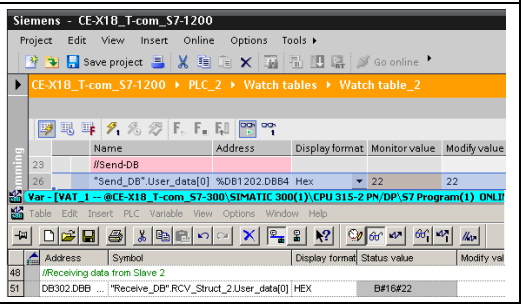
S7-1200 slave 1 -> S7-300 master

Table 3-14

No.	Instruction	Note/screen shot
1.	As an example of transferring user data from slave 1 to the master, send byte 0 of the user data field in row 26 is to be changed: • Open “Watch table_1” and enter a value in the “Modify value” column in row 26. • Right-click and select “Modify” -> “Modify now” to apply the value.	
2.	The value is transferred to the master and written to receive byte 0 of the user data field of the receive structure for slave 1 (as can be seen in row 30 of the master variable table).	

S7-1200 slave 2 -> S7-300 master

Table 3-15

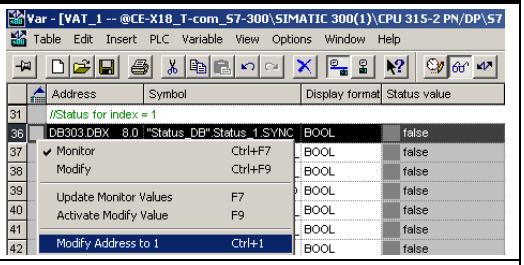
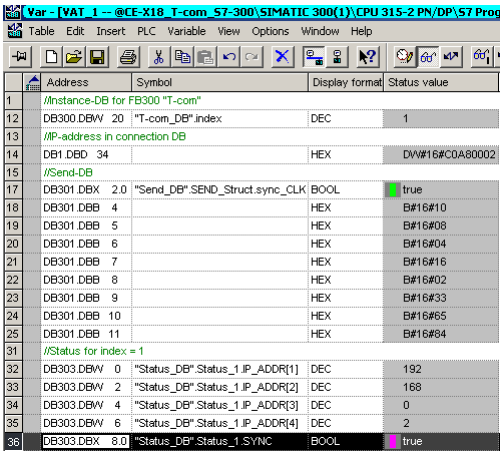
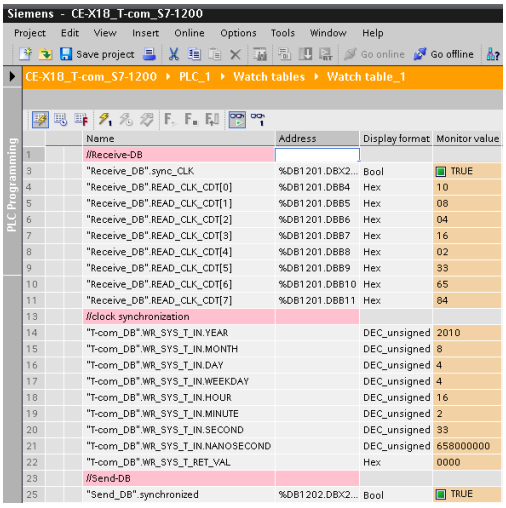
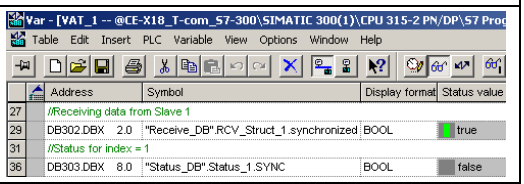
No.	Instruction	Note/screen shot
1.	As an example of transferring user data from slave 2 to the master, send byte 0 of the user data field in row 26 is to be changed: • In “Watch table_2”, enter a value in the “Modify value” column in row 26. • Right-click and select “Modify” -> “Modify now” to apply the value.	
2.	The value is transferred to the master and written to receive byte 0 of the user data field of the receive structure for slave 2 (as can be seen in row 51 of the master variable table).	

3.4.3 Time-of-day synchronization

Manual synchronization of one slave

Slave 1 is to be manually synchronized with the master system time.
Table 3-16 shows the procedure.
Slave 2 can be synchronized using the same procedure.

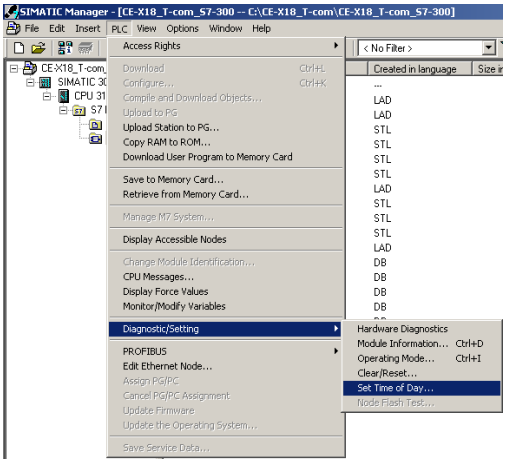
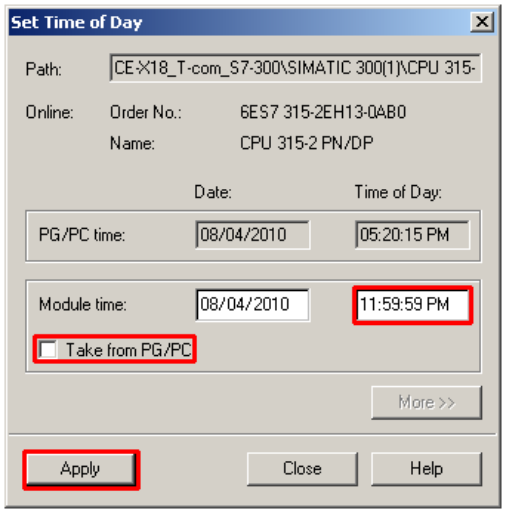
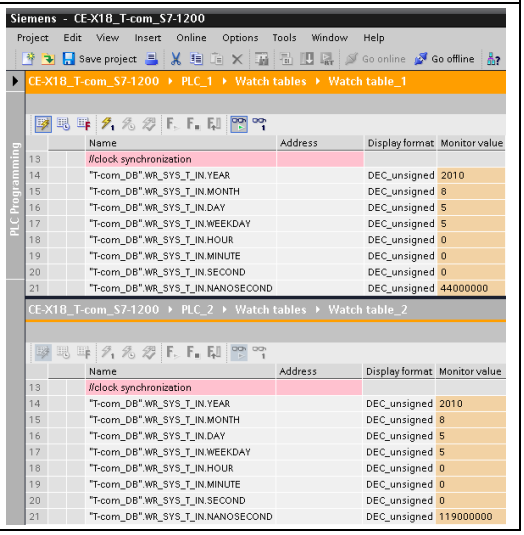
Table 3-16

1.	• Right-click and select -> “Modify Address to 1” (row 36 in the master variable table) to set the synchronization request in the status structure for slave 1.	
2.	• The system time is written to the send data in DATE_AND_TIME format (rows 18-25). • The synchronization request in the send data is set (row 17). • The send data is sent to slave 1, including index 1 (row 12).	
3.	• The time-of-day synchronization data is written to the receive block of slave 1 (“Watch table_1”, rows 3-11). • The converted synchronization time of the DTL data type is written to the S7-1200 system time (rows 14-22). • After successful time-of-day synchronization, the synchronization acknowledgement is set (row 25).	
4.	• On the master side, the synchronization acknowledgement is written to the receive structure of slave 1 (row 29). • The synchronization request in the status structure for slave 1 is reset (row 36).	

Automatic synchronization of all slaves

In chapter 3.2.2, the daily synchronization time of all slaves has been set to 00:00. To test this function, the master system time is set to 11:59 PM.

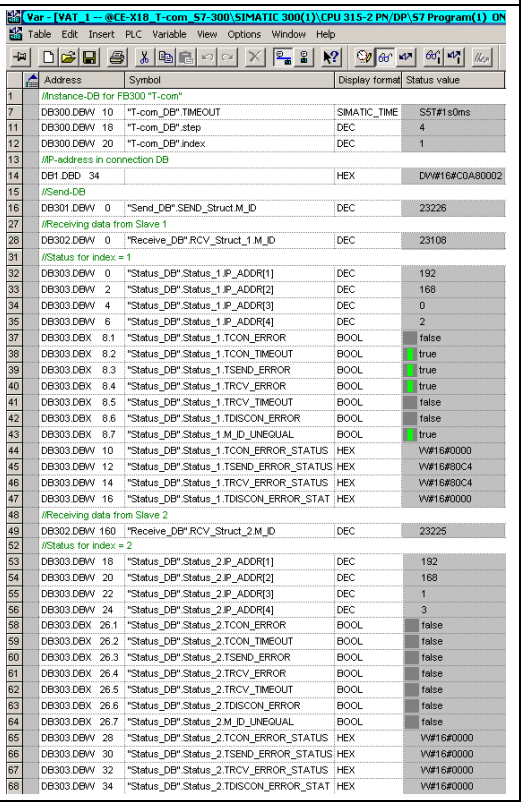
Table 3-17

No.	Instruction	Note/screen shot
1.	In Step 7 V5.4, select “PLC” -> “Diagnostic/Setting” -> “Set Time of Day…”.	
2.	• Uncheck “Take from PG/PC”. • Set the module time to “11:59:59 PM”. • Apply the settings.	
3.	The written system time of the slaves can be used to check whether the time-of-day synchronization of the slaves has been successful (rows 14-21 in “Watch table_1” and “Watch table_2”).	

3.4.4 Communication errors

The Ethernet cable of slave 1 is removed to demonstrate communication error analysis.
Table 3-16 describes the procedure.
The same procedure can be used to simulate and analyze the interruption of communication with slave 2.

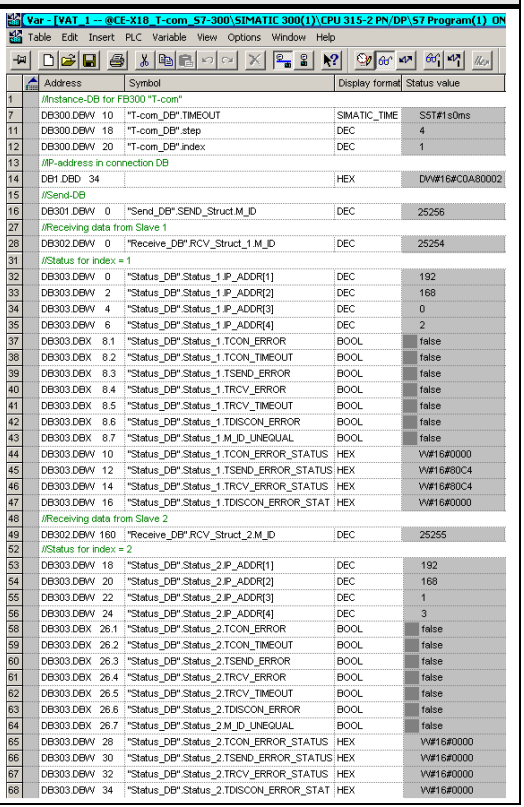
Table 3-18

No.	Instruction	Note/screen shot
1.	Remove the Ethernet cable from the LAN port of slave 1.	
2.	• The step sequence stops (row 11) when communicating with slave 1 (row 12) and waits for slave 1 to confirm the connection establishment. • Once the timeout time of one second (row 7) has elapsed, the step sequence continues and the timeout message occurs in the status structure for slave 1 (row 38). • The TSEND and TRCV communication blocks report errors (rows 39 and 40) with their error statuses (rows 45 and 46). • In addition, the discrepancy between the sent message ID (row 16) and the ID last received from slave 1 (row 24) is identified and output in row 28. • Data exchange with slave 2 still takes place correctly (rows 58-68). • While slave 1 is not accessible, a new attempt to establish a connection to slave 1 generates the same error profile.	
3.	Reconnect the Ethernet cable to the LAN port of slave 1.	

Var - [VAT_1 -- @CE-X18_T-com_S7-300\SIMATIC 300(1)\CPU 315-2 PN/DP\S7 Program(1) ON

Table Edit Insert PLC Variable View Options Window Help

	Address	Symbol	Display format	Status value
1		//Instance-DB for FB300 "T-com"		
7	DB300.DBW 10	"T-com_DB".TIMEOUT	SIMATIC_TIME	S5T#1s0ms
11	DB300.DBW 18	"T-com_DB".step	DEC	4
12	DB300.DBW 20	"T-com_DB".index	DEC	1
13		//IP-address in connection DB		
14	DB1.DBD 34		HEX	DW#16#C0A80002
15		//Send-DB		
16	DB301.DBW 0	"Send_DB".SEND_Struct.M_ID	DEC	23226
27		//Receiving data from Slave 1		
28	DB302.DBW 0	"Receive_DB".RCV_Struct_1.M_ID	DEC	23108
31		//Status for index = 1		
32	DB303.DBW 0	"Status_DB".Status_1.IP_ADDR[1]	DEC	192
33	DB303.DBW 2	"Status_DB".Status_1.IP_ADDR[2]	DEC	168
34	DB303.DBW 4	"Status_DB".Status_1.IP_ADDR[3]	DEC	0
35	DB303.DBW 6	"Status_DB".Status_1.IP_ADDR[4]	DEC	2
37	DB303.DBX 8.1	"Status_DB".Status_1.TCON_ERROR	BOOL	false
38	DB303.DBX 8.2	"Status_DB".Status_1.TCON_TIMEOUT	BOOL	true
39	DB303.DBX 8.3	"Status_DB".Status_1.TSEND_ERROR	BOOL	true
40	DB303.DBX 8.4	"Status_DB".Status_1.TRCV_ERROR	BOOL	true
41	DB303.DBX 8.5	"Status_DB".Status_1.TRCV_TIMEOUT	BOOL	false
42	DB303.DBX 8.6	"Status_DB".Status_1.TDISCON_ERROR	BOOL	false
43	DB303.DBX 8.7	"Status_DB".Status_1.M_ID_UNEQUAL	BOOL	true
44	DB303.DBW 10	"Status_DB".Status_1.TCON_ERROR_STATUS	HEX	W#16#0000
45	DB303.DBW 12	"Status_DB".Status_1.TSEND_ERROR_STATUS	HEX	W#16#80C4
46	DB303.DBW 14	"Status_DB".Status_1.TRCV_ERROR_STATUS	HEX	W#16#80C4
47	DB303.DBW 16	"Status_DB".Status_1.TDISCON_ERROR_STAT	HEX	W#16#0000
48		//Receiving data from Slave 2		
49	DB302.DBW 160	"Receive_DB".RCV_Struct_2.M_ID	DEC	23225
52		//Status for index = 2		
53	DB303.DBW 18	"Status_DB".Status_2.IP_ADDR[1]	DEC	192
54	DB303.DBW 20	"Status_DB".Status_2.IP_ADDR[2]	DEC	168
55	DB303.DBW 22	"Status_DB".Status_2.IP_ADDR[3]	DEC	1
56	DB303.DBW 24	"Status_DB".Status_2.IP_ADDR[4]	DEC	3
58	DB303.DBX 26.1	"Status_DB".Status_2.TCON_ERROR	BOOL	false
59	DB303.DBX 26.2	"Status_DB".Status_2.TCON_TIMEOUT	BOOL	false
60	DB303.DBX 26.3	"Status_DB".Status_2.TSEND_ERROR	BOOL	false
61	DB303.DBX 26.4	"Status_DB".Status_2.TRCV_ERROR	BOOL	false
62	DB303.DBX 26.5	"Status_DB".Status_2.TRCV_TIMEOUT	BOOL	false
63	DB303.DBX 26.6	"Status_DB".Status_2.TDISCON_ERROR	BOOL	false
64	DB303.DBX 26.7	"Status_DB".Status_2.M_ID_UNEQUAL	BOOL	false
65	DB303.DBW 28	"Status_DB".Status_2.TCON_ERROR_STATUS	HEX	W#16#0000
66	DB303.DBW 30	"Status_DB".Status_2.TSEND_ERROR_STATUS	HEX	W#16#0000
67	DB303.DBW 32	"Status_DB".Status_2.TRCV_ERROR_STATUS	HEX	W#16#0000
68	DB303.DBW 34	"Status_DB".Status_2.TDISCON_ERROR_STAT	HEX	W#16#0000

No.	Instruction	Note/screen shot
4.	• After connection recovery has been identified, the error bits are reset in the status structure for slave 1 (rows 37-43). • Data exchange with slave 1 has been restored. • The error statuses of the TSEND (row 45) and TRCV (row 46) communication blocks include the status information of the last errors.	

Var - [VAT_1 -- @CE-X18_T-com_S7-300\SIMATIC 300(1)\CPU 315-2 PN/DP\S7 Program(1) ON

Table Edit Insert PLC Variable View Options Window Help

	Address	Symbol	Display format	Status value
1		//Instance-DB for FB300 "T-com"		
7	DB300.DBW 10	"T-com_DB".TIMEOUT	SIMATIC_TIME	S5T#1s0ms
11	DB300.DBW 18	"T-com_DB".step	DEC	4
12	DB300.DBW 20	"T-com_DB".index	DEC	1
13		//IP-address in connection DB		
14	DB1.DBD 34		HEX	DW#16#C0A80002
15		//Send-DB		
16	DB301.DBW 0	"Send_DB".SEND_Struct.M_ID	DEC	25256
27		//Receiving data from Slave 1		
28	DB302.DBW 0	"Receive_DB".RCV_Struct_1.M_ID	DEC	25254
31		//Status for index = 1		
32	DB303.DBW 0	"Status_DB".Status_1.IP_ADDR[1]	DEC	192
33	DB303.DBW 2	"Status_DB".Status_1.IP_ADDR[2]	DEC	168
34	DB303.DBW 4	"Status_DB".Status_1.IP_ADDR[3]	DEC	0
35	DB303.DBW 6	"Status_DB".Status_1.IP_ADDR[4]	DEC	2
37	DB303.DBX 8.1	"Status_DB".Status_1.TCON_ERROR	BOOL	false
38	DB303.DBX 8.2	"Status_DB".Status_1.TCON_TIMEOUT	BOOL	false
39	DB303.DBX 8.3	"Status_DB".Status_1.TSEND_ERROR	BOOL	false
40	DB303.DBX 8.4	"Status_DB".Status_1.TRCV_ERROR	BOOL	false
41	DB303.DBX 8.5	"Status_DB".Status_1.TRCV_TIMEOUT	BOOL	false
42	DB303.DBX 8.6	"Status_DB".Status_1.TDISCON_ERROR	BOOL	false
43	DB303.DBX 8.7	"Status_DB".Status_1.M_ID_UNEQUAL	BOOL	false
44	DB303.DBW 10	"Status_DB".Status_1.TCON_ERROR_STATUS	HEX	W#16#0000
45	DB303.DBW 12	"Status_DB".Status_1.TSEND_ERROR_STATUS	HEX	W#16#80C4
46	DB303.DBW 14	"Status_DB".Status_1.TRCV_ERROR_STATUS	HEX	W#16#80C4
47	DB303.DBW 16	"Status_DB".Status_1.TDISCON_ERROR_STAT	HEX	W#16#0000
48		//Receiving data from Slave 2		
49	DB302.DBW 160	"Receive_DB".RCV_Struct_2.M_ID	DEC	25255
52		//Status for index = 2		
53	DB303.DBW 18	"Status_DB".Status_2.IP_ADDR[1]	DEC	192
54	DB303.DBW 20	"Status_DB".Status_2.IP_ADDR[2]	DEC	168
55	DB303.DBW 22	"Status_DB".Status_2.IP_ADDR[3]	DEC	1
56	DB303.DBW 24	"Status_DB".Status_2.IP_ADDR[4]	DEC	3
58	DB303.DBX 26.1	"Status_DB".Status_2.TCON_ERROR	BOOL	false
59	DB303.DBX 26.2	"Status_DB".Status_2.TCON_TIMEOUT	BOOL	false
60	DB303.DBX 26.3	"Status_DB".Status_2.TSEND_ERROR	BOOL	false
61	DB303.DBX 26.4	"Status_DB".Status_2.TRCV_ERROR	BOOL	false
62	DB303.DBX 26.5	"Status_DB".Status_2.TRCV_TIMEOUT	BOOL	false
63	DB303.DBX 26.6	"Status_DB".Status_2.TDISCON_ERROR	BOOL	false
64	DB303.DBX 26.7	"Status_DB".Status_2.M_ID_UNEQUAL	BOOL	false
65	DB303.DBW 28	"Status_DB".Status_2.TCON_ERROR_STATUS	HEX	W#16#0000
66	DB303.DBW 30	"Status_DB".Status_2.TSEND_ERROR_STATUS	HEX	W#16#0000
67	DB303.DBW 32	"Status_DB".Status_2.TRCV_ERROR_STATUS	HEX	W#16#0000
68	DB303.DBW 34	"Status_DB".Status_2.TDISCON_ERROR_STAT	HEX	W#16#0000

3.4.5 Master power failure

After power recovery of the master, the step sequence of FB300 “T-com” continues from the last executed position.

4 Code Elements

This example uses the following program codes:

Table 4-1

No.	File name	Contents
1.	CE-X18B_Master_v1d2.zip • CE-X18_ T-com_S7-300	Zip file with the S7-300 master project for deterministic T communication
2.	CE-X18B_Slave_v1d2.zip • CE-X18B_Slave_v1d2.ap10	Zip file with the S7-1200 slave project for deterministic T communication

5 History

Table 5-1

Version	Date	Modification
V1.0	01/13/10	T-communication via the integrated S7-300 CPU interface (task A) and via a S7-300 CP (task B)
V1.1	02/10/10	Extensions in chapter 2.3: S7-1200 Data transfer
V1.2	08/31/10	Modification of the automation task in deterministic data exchange via T communication (task B)