Ethernet Communication: Data Exchange S7-1200 <-> S7-1200

S7-1200

Configuration Example X17 • September 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 configuration, equipment 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 used correctly. These application examples do not relieve you of the responsibility to use sound practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we 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 this application example and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

We accept no liability for 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 thereof without express authorization from Siemens Industry Sector.

If you have any questions regarding this document, please send us an e-mail to the following address:

online-support.automation@siemens.com

Table of Contents

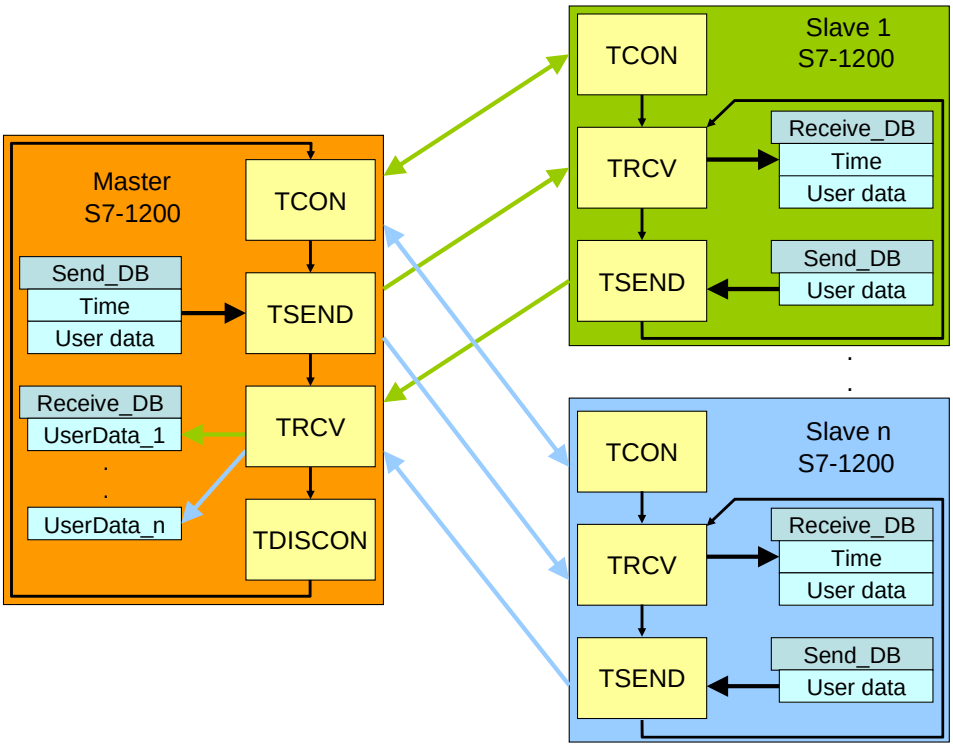
1 Automation Task

1.1 Tasks

Data exchange (e.g. for time synchronization) shall be enabled between an S7-1200 master controller and several S7-1200 slave controllers.

Layout of the application task

Figure 1-1



Application requirements

Master and slaves have one send and one receive data block each (Send_DB and Receive_DB).
The master sends a TCP/IP connection request to the first slave via the TCON block. To acknowledge the established connection the partner also executes the TCON block.
For a synchronization request the master reads the system time and sends it to the slave via the TSEND communication block together with the user data. With the TRCV receive block the slave receives the data in the Receive_DB data block.
The slave synchronizes its system time with the received clock time of the master.
Then slave 1 sends its user data to the master via the TSEND block. The user data of slave 1 are stored on the master side via the TRCV block at the given location in the receive data block.
Then the master disconnects from slave 1 via the TDISCON block.

This procedure is repeated with the subsequent slaves. After the data exchange between master and the last slave the master starts the data exchange with slave 1.

On the slave side a once established connection remains reserved. So TCON must only be called for initialization.

1.2 Setup

The automation task is demonstrated at the example of data exchange between a master CPU and two slave controllers (slave 1 and slave 2).

Schematic layout

Figure 1-2

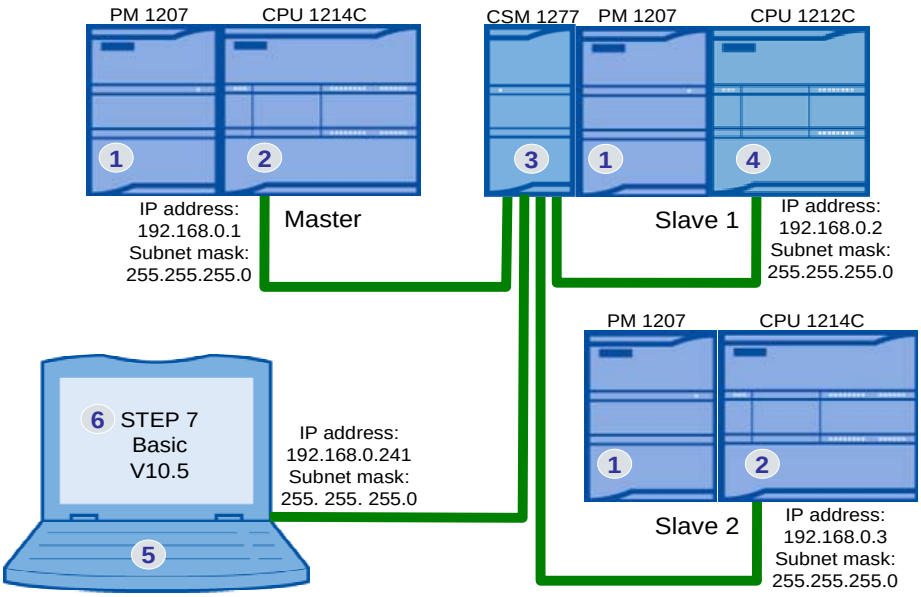


Figure 1-2 shows the principal setup. The communicating CPUs as well as the programming device with the “STEP 7 Basic V10.5” software for programming the S7-1200 are each connected with the CSM 1277 switch via Ethernet cable.

Subnet mask

The IP addresses of the communication nodes only differ in the last octet (192.168.0.x), which makes it a class C network. As a subnet mask “255.255.255.0” is selected for a class C network.
To expand the slave node in the example program further IP addresses can be used within this class C network (192.168.0.x).

List of components

Table 1-1

No.	Component	Qty.	MLFB/order number
1.	POWER SUPPLY S7-1200 PM1207	3	6EP1332-1SH71
2.	S7-1200 CPU1214C	2	6ES7214-1AE30-0XB0
3.	COMPACT SWITCH MODULE CSM 1277	1	6GK7277-1AA00-0AA0
4.	S7-1200 CPU1212C	1	6ES7212-1AD30-0XB0
5.	PC/PG	1	

No.	Component	Qty.	MLFB/order number
6.	STEP 7 BASIC V10.5	1	6ES7822-0AA00-0YA0
7.	STEP 7 Basic V10.5 Service Pack 2	1	Entry ID:39741113

2 Automation Solution

For data exchange via Ethernet the S7-1200 provides the open TCP/IP communication with the T communication block:

- TCON, TSEND, TRCV and TDISCON (with explicit execution of the connecting and disconnecting process)

and

- TSEND_C and TRCV_C (with integrated connecting and disconnecting process).

The following protocols are supported by the T communication blocks:

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

On the master as well as the slave side the communication blocks with explicit execution of the connecting and disconnecting process are selected:

- TCON for establishing the connection
- TSEND for sending the data
- TRCV for receiving the data
- TDISCON for disconnecting

The selected protocol is “ISO-on-TCP”.
In the OSI model the “ISO-on-TCP” protocol is added to the TCP protocol and provides the advantage of a message-orientated approach, which is especially helpful for the communication between SIMATIC systems.

For the connection configuration in STEP 7 Basic V10.5 the respective connection parameter is identified via the IP address. For a configured connection a connection resource is reserved and the connection parameters are stored in a connection data block. The IP address of the partner is also stored here.

The maximum number for the open T communication is limited to a maximum of 8 simultaneous connections.

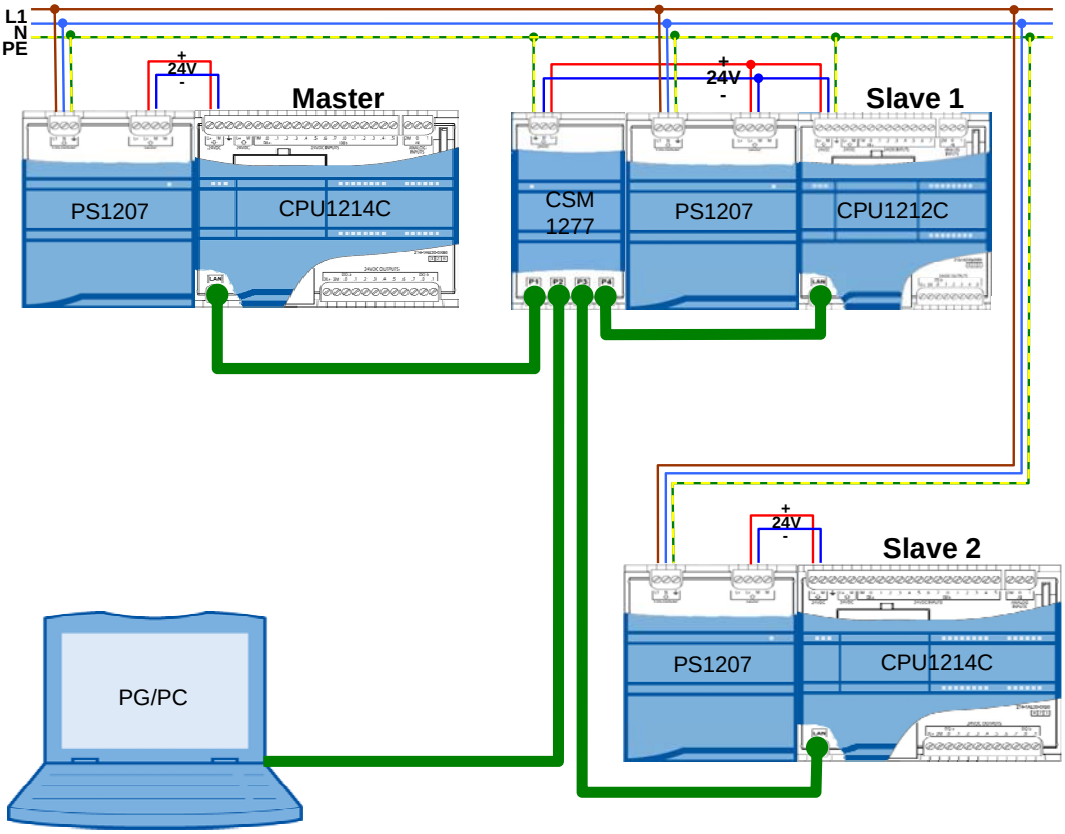
Changing the IP addresses in the connection data block enables data exchange between more than 8 different communication partners via the same connection resource.

With the “ISO-on-TCP” protocol up to 8.192 bytes per job can be transferred.

2.1 Wiring diagram

The components list is available in chapter 1.2.

Figure 2-1



2.2 Program structure

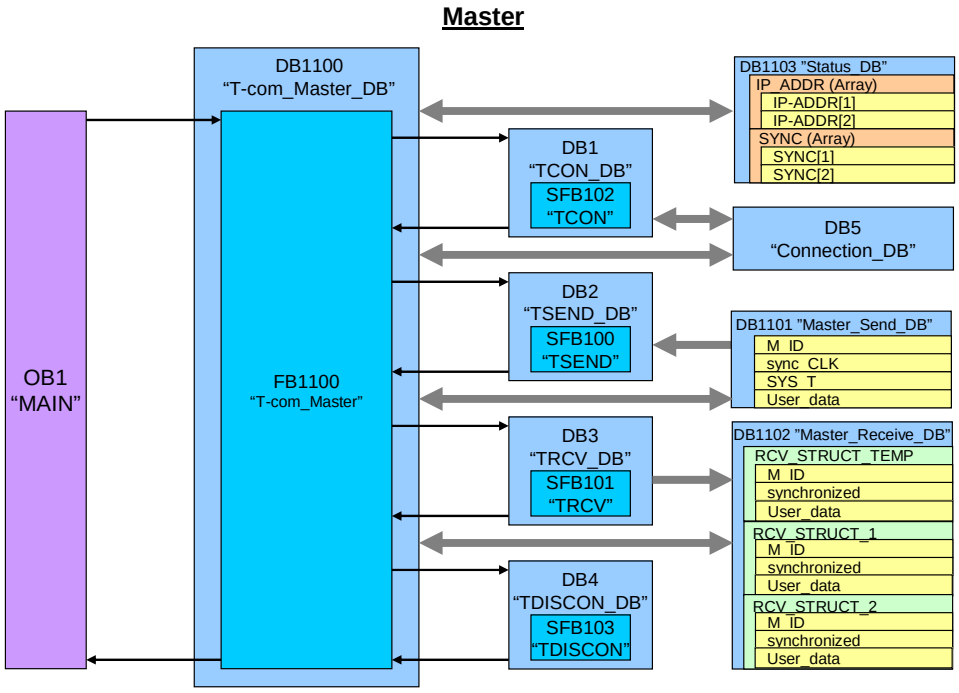
This chapter describes the program structure of the example on the function and data block level of the automation system.

2.2.1 Block structure of the master

Representation

Figure 2-2 shows the call hierarchy of the used blocks as well as the access to the used data blocks for the master.

Figure 2-2



Description

Data block DB1103 "Status_DB" contains the IP address, as well as control and status information of all slaves in form of arrays. Figure 2-2 displays the arrays for the IP address information "IP-ADDR" and synchronization request "SYNC" with 2 elements for 2 slaves. Depending on which slave is to be addressed the respective element is accessed.

OB1 "MAIN" cyclically calls function block FB1100 "T-com_Master" with its instance DB1100 "T-com_Master_DB".

"T-com_Master" reads the IP address information of the first slave from the status DB 1103 and writes it to the connection data block DB5 "Connection_DB".

SFB102 "TCON" with its instance DB1 establishes a connection with the IP address stored in the connection DB5.

In FB1100 the local time is read cyclically and compared with a given daily synchronization time. At this time all synchronization request bits of the "SYNC"

array are set in the status DB. The clock synchronization can also be executed individually for each slave via the watch table.

Setting the synchronization request “SYNC[1]” for the first slave in DB1103 “Status_DB” causes reading the system time “SYS_T” and storing in send data block DB1101 “Send_DB” together with the synchronization request “sync_CLK”.

Function block FB1100 calls the send block SFB100 “TSEND” with its instance DB2. It transmits the content of the send data block DB1101 “Master_Send_DB” to the first slave. Apart from the clock time synchronization information, “User_data” and a message ID “M_ID” are also transferred.

SFB101 “TRCV” with its instance DB3 waits to receive from slave 1 and saves the content to the temporary receive data structure “RCV_STRUCT_TEMP” in the receive DB1102 “Master_Receive_DB”.

The temporary receive data “RCV_STRUCT_TEMP” is copied to the respective receive structure in the receive data block (“RCV_STRUCT_1” for slave 1), depending on which slave they are from.

The received message ID “M_ID“ is compared with the sent ID. A deviation is noted in the status DB of the respective element [1] of the “M_ID_UNEQUAL” array for slave 1.

After successful synchronization of slave 1 (signaled by the “synchronized” tag) the synchronization request bit “SYNC[1]” for slave 1 is reset in the status DB.

SFB103 “TDISCON“ with its instance DB4 disconnects from slave 1.

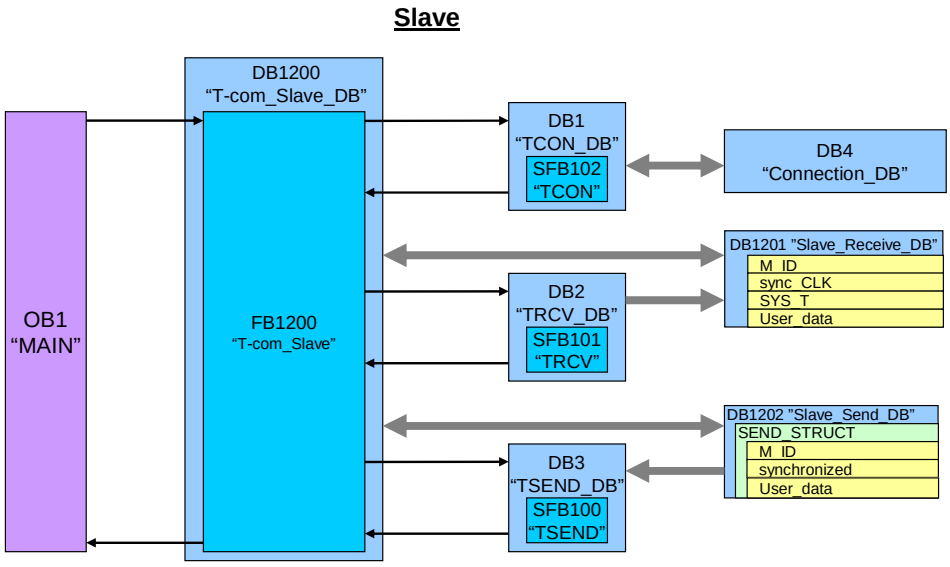
The message ID “M_ID” is increased and the data exchange with slave 2 is handled in the same way.

2.2.2 Block structure of the slave

Representation

Figure 2-3 shows the call hierarchy of the used blocks as well as the access to the used data blocks for the slave.

Figure 2-3



Description

OB1 “MAIN” cyclically calls function block FB1200 “T-com_Slave” with its instance DB1200 “T-com_Slave_DB”.

In the first cycle, FB1200 “T-com_Slave” calls the SFB100 “TCON” with its instance DB1. The identification of the master receives TCON from the connection DB4 “Connection-DB”. If the connection has been established it is maintained.

SFB101 “TRCV” with its instance DB2 waits to receive from the master and saves the content to DB1201 “Slave_Receive_DB”.

At a synchronization request “sync_CLK”, the FB1200 “T-com” synchronizes its system time with the system time “SYS_T” received from the master. After successful clock synchronization the “synchronized” bit is set in the send structure “SEND_STRUCT” in the send DB1202 “Slave_Send_DB”.

The received message ID “M_ID” from DB1201 “Receive_DB” is mirrored in the send block DB1202 “Send_DB”.

FB1200 “T-com_Slave” sends the content of the send structure “SEND_STRUCT” using the SFB100 “TSEND” with its instance DB3 to the master.

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

2.3 Used blocks

2.3.1 Master

The following table gives you an overview of the used blocks on the master side.

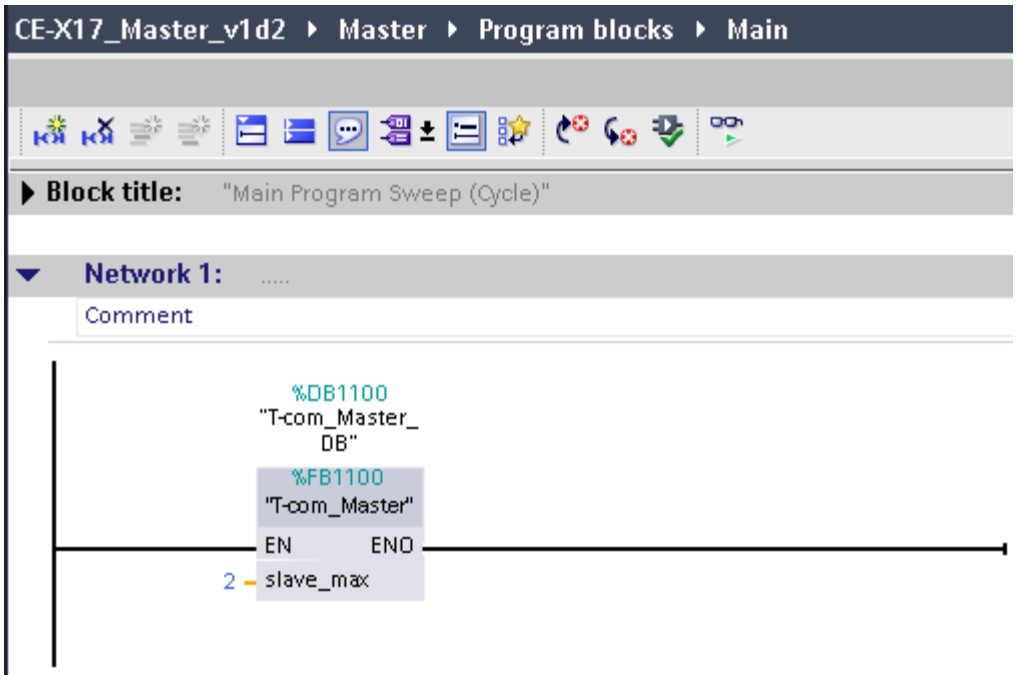
Table 2-1

Object name	Symbolic name	Description
OB1	Main	Cyclic organization block
FB1100	T-com_Master	Function block for deterministic data exchange with several slaves via their function block “T-com_Slave”
SFB100	TSEND	T-communication block for sending data
SFB101	TRCV	T-communication block for receiving data
SFB102	TCON	T-communication block for establishing a connection
SFB103	TDISCON	T-communication block for disconnection
DB5	Connection_DB	Connection data block for TCON
DB1102	Master_Receive_DB	Receive data block for TRCV
DB1101	Master_Send_DB	Send data block for TSEND
DB1103	Status_DB	Status data block for all slaves
DB1100	T-com_Master_DB	Instance data block for “T-com_Master”
DB1	TCON_DB	Instance data block for TCON
DB4	TDISCON_DB	Instance data block for TDISCON
DB3	TRCV_DB	Instance data block for TRCV
DB2	TSEND_DB	Instance data block for TSEND

T-com_Master (FB1100)

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

Figure 2-4



The selected instance data block is DB1100.

As the only input the maximal number of slaves “slave_max” must be specified.

The respective slave is identified via the “slave” tag. Data exchange with the slaves is a sequential process.

The following static tags of the FB300 provide configuration options via the initial value or the status evaluation.

Table 2-2

Name	Data type	Description
hour	USInt	Hour specification of the daily local synchronization time (value range: 0 – 23)
minute	USInt	Minute specification of the daily local synchronization time (value range: 0 – 59)
TIMEOUT	Time	Maximum wait time until the step chain in FB1100 is automatically switched further (default: 500 ms)

Status_DB (DB1103)

The status DB consists of the following arrays with respectively 2 elements for 2 slaves. For the expansion of the slave number the arrays must be increased accordingly (see chapter 2.3.4).

Table 2-3

Name	Element type	Description
IP_ADDR	USInt	4th octet of the IP address of the slave
SYNC	Bool	Clock synchronization request
TIMEOUT	Bool	Exceeding the processing time when connecting or receiving data
M_ID_UNEQUAL	Bool	Sent and received M_ID unequal
TCON_ERROR	Bool	Error message of the communication block TCON
TCON_ERROR_STATUS	Word	Status of the TCON block during last error
TSEND_ERROR	Bool	Error message of the TSEND communication block
TSEND_ERROR_STATUS	Word	Status of the TSEND block during the last error
TRCV_ERROR	Bool	Error message of the TRCV communication block
TRCV_ERROR_STATUS	Word	Status of the TRCV block during last error
TDISCON_ERROR	Bool	Error message of the TDISCON communication block
TDISCON_ERROR_STATUS	Word	Status of the TDISCON block during last error

2.3.2 Slave

The following table gives you an overview of the used blocks on the slave side.

Table 2-4

Object name	Symbolic name	Description
OB1	Main	Cyclic organization block
FB1200	T-com_Slave	Function block for data exchange with one master via its function block “T-com_Master”
SFB100	TSEND	T-communication block for sending data
SFB101	TRCV	T-communication block for receiving data
SFB102	TCON	T-communication block for establishing a connection
DB4	Connection_DB	Connection data block for TCON
DB1201	Slave_Receive_DB	Data block for receiving from the master
DB1202	Slave_Send_DB	Data block for sending to the master
DB1200	T-com_Slave_DB	Instance data block for “T-com_Slave”
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

“Master_Send_DB” and “Slave_Receive_DB”

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	Clock synchronization request
SYS_T	DTL	Synchronization time of the master
User_data	Array of 144 bytes	User data (master -> slave)

The user data “User_data” can be changed individually. However, the data structure must be identical on the sending and receiving side.

“SEND_STRUCT” and “RCV_STRUCT_x”

The send structure of the “SEND_STRUCT” slave and the receive structure of the “RCV_STRUCT_x” master must be identical. The receive DB 1102 “Master_Receive_DB” consists of 3 receive structures (one temporary and 2 for 2 slaves). This structure consists of 34 bytes and has the following design:

Table 2-6

Name	Data type	Description
M_ID	Int	Mirrored message ID for acknowledgement
synchronized	Bool	Clock synchronization acknowledgement
User_data_Byte	Byte	User data (slave -> master)
User_data_Char	Char	User data (slave -> master)
User_data_Word	Word	User data (slave -> master)
User_data_DWord	DWord	User data (slave -> master)
User_data_DInt	DInt	User data (slave -> master)
User_data_Real	Real	User data (slave -> master)
User_data_Time	Time	User data (slave -> master)
User_data_USInt	USInt	User data (slave -> master)
User_data_UInt	UInt	User data (slave -> master)
User_data_UDInt	UDInt	User data (slave -> master)
User_data_SInt	SInt	User data (slave -> master)

The user data “User_data” can be changed individually. However, the data structure must be identical on the sending and receiving side.
When increasing the slave number an appropriate number of receive structures “RCV_STRUCT_x” must be defined in the receive DB “Master_Receive_DB” (see chapter 2.3.4).

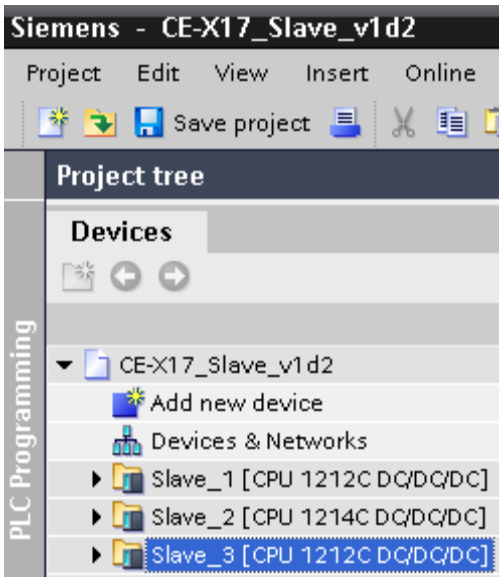
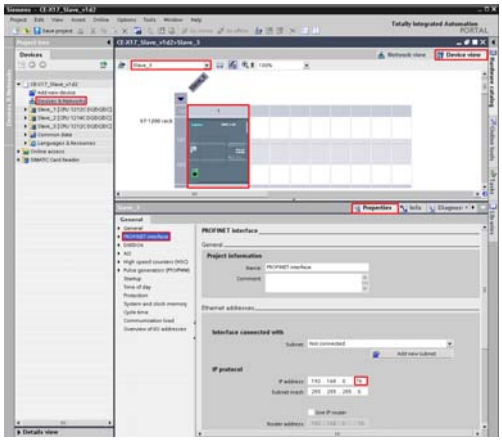
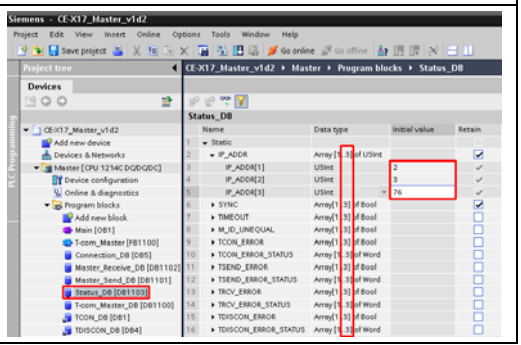
Data consistency is on the program side guaranteed by the sequential processing of the send and receive jobs.
Status DB 1103 provides the option of directly influencing the communication errors.
Due to the continued data exchange between master and slaves the data consistency can only be provided for one cycle.
Consistent data must therefore be written into the send data blocks by the user within one cycle or be read from the receive data blocks.

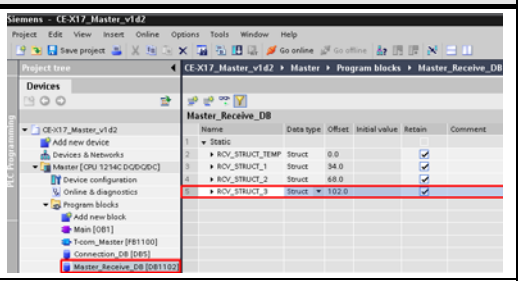
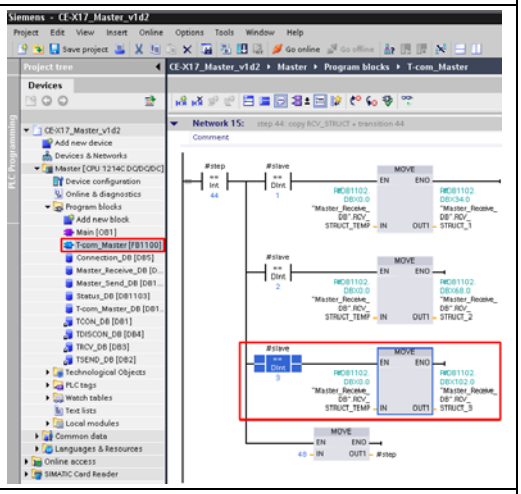
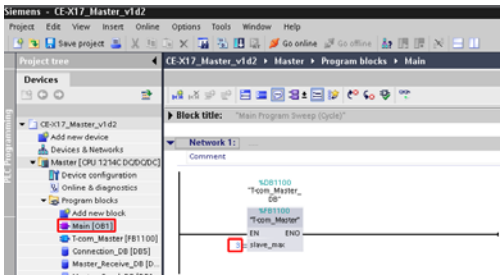
2.3.4 Expanding the slave number

To adjust the master and the slave project of an increased number of subordinate slave controllers proceed as follows:

The expansion is displayed at the example of 3 slaves.

Table 2-7

No.	Instruction	Note/picture
1.	• In the project “CE-X17_Slave_v1d2“, you duplicate one of both controller folders “Slave_1” or “Slave_2” via “Copy & Paste”.	
2.	• In the menu item “Devices & Networks“, you open the device view for the newly created “Slave_3”. • Mark the controller and open the settings of the “PROFINET interface” • Adjust the 4th octet of the IP address to the newly created controller (here: “76”). Here you also exchange the CPU from the hardware catalog (if necessary).	
3.	• In the master project “CE-X17_Master_v1d2” you exchange the status DB1103. • Expand all arrays to 3 elements. • When changing the arrays “IP_ADDR” the initial values of all elements must be created new. • For the IP address of the newly created slave 3 you enter the 4th octet of the IP address from step 2 (line 5: “76”).	

No.	Instruction	Note/picture
4.	• In the master project “CE-X17_Master_v1d2” you open the receive DB1102. • Duplicate one of the existing receive structures “RCV_STRUCT_x” and name the newly create structure (here: “RCV_STRUCT_3”).	
5.	• In the master project “CE-X17_Master_v1d2” you open FB1100 “T-com_Master”. • In network 15 you expand the copy instruction of the temporary receive structure “RCV_STRUCT_TEMP” at the location of the newly created structure “RCV_STRUCT_3” for the reception of slave 3.	
6.	• In the master project “CE-X17_Master_v1d2” you open OB1 “Main”. • Change the number of slaves with which the master shall communicate (“slave_max”) to “3”.	

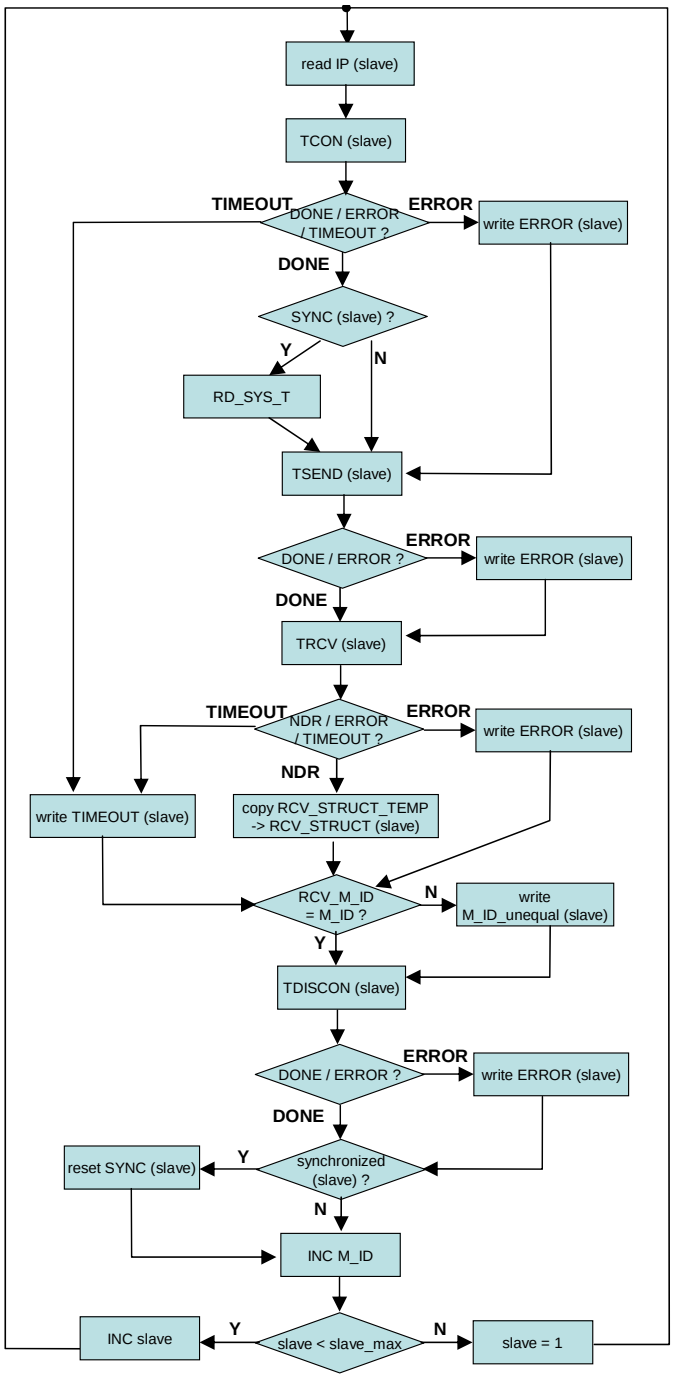
2.4 Program sequence

2.4.1 Program sequence in the master controller

Flow chart

The following flowchart illustrates the program sequence on the master side. The functionality is bundled in FB1100 “T-com_Master”, which is called cyclically by OB1. FB1100 is realized as step chain.

Figure 2-5



Description of the flow chart

The respective slave used for exchanging the data is identified via the “slave” index.

Starting with “slave” = 1 the IP address (4th octet) of the first slave is read from the status DB and written to the connection data block.

TCON sends a connection request to this slave. Since the step chain is only continued after acknowledgement of the connection request, a maximum processing time “TIMEOUT” elapses.

The information that this time has elapsed or the “Error” feedback message is output in the status DB depending on the communication partner (“slave”).

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

With the send block “TSEND” the content of the send data block is transferred to the slave. Apart from the clock time synchronization information, a message ID “M_ID” is also transferred.

For an “ERROR“ message of the “TSEND” send block the error information is written to the elements “TSEND_ERROR(slave)” and “TSEND_ERROR_STATUS(slave)” in the status DB.

Data of the slave are received with the receive block “TRCV” and written to the temporary receive structure “RCV_STRUCT_TEMP” in the receive DB.

After the maximum processing time “TIMEOUT” has elapsed or upon “ERROR” message of receive block “TRCV” the error information is written to the respective “slave” elements of the arrays in the status DB.

For positive acknowledgement of receipt “TRCV_NDR“ the temporary receive structure “RCV_STRUCT_TEMP“ is copied to the respective slave “RCV_STRUCT(slave)”.

From the temporary receive data the message ID “M_ID” mirrored by the slave is compared with the sent “M_ID”. If it deviates this is noted in the “slave” element of the array (“M_ID_unequal”) in the status DB.

Subsequently, the connection with the slave is established with the “TISCON” block. Errors which occur during the disconnecting process are also output in the status DB.

Now the successful synchronization from the temporary receive data of the slave are checked (“synchronized”). In the positive case the synchronization request “SYNC” is reset for the slave. Otherwise the clock synchronization is repeated at the next communication with this slave.

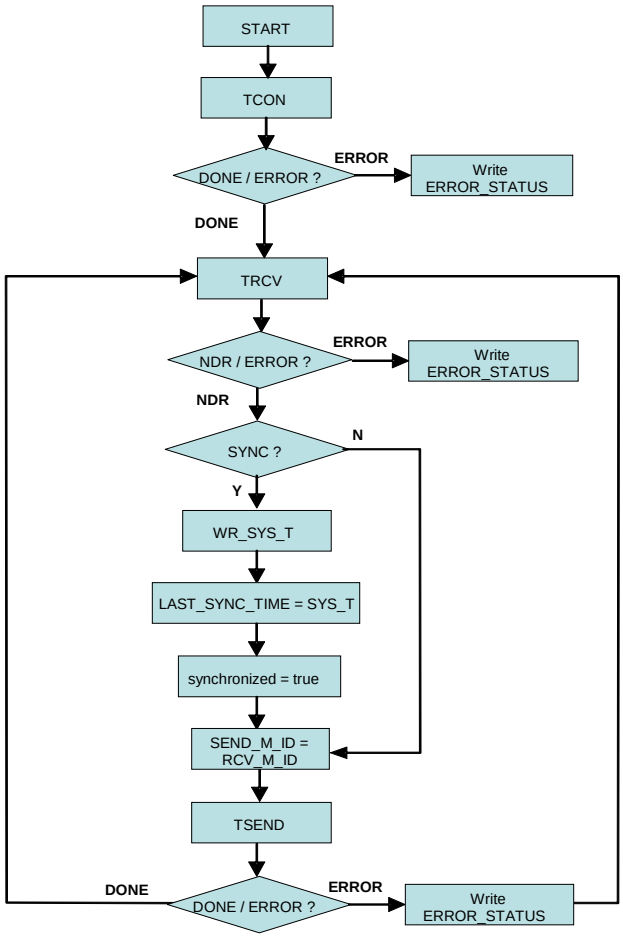
The message ID is increased (“INC M_ID”) and the current communication node “slave” is compared with the maximum slave number “slave_max”. As long as “slave_max” has not been reached, the communication node number is increased (“INC slave”). Otherwise, the index is set to the initial value (slave = “1”).

2.4.2 Program sequence in the slave controller

Flow chart

The following flowchart illustrates the program sequence on the slave side. The functionality is bundled in FB1200 “T-com_Slave”, which is called cyclically by OB1. FB1200 is realized as a step chain.

Figure 2-6



Description of the flow chart

Upon initialization the TCON sends a connection request to the master. For an “ERROR” feedback message the status information in the “TCON_ERROR_STATUS” tag is maintained. For positive acknowledgement of the established connection “DONE”, the slave waits for reception by the master via the “TRCV” block.

For a faulty “ERROR” feedback message of the receive block the status information in the “TRCV_ERROR_STATUS” tag is maintained.
When receiving new data “NDR”, the received synchronization request “SYNC” of the master is polled.

If it is given, the received system time “SYS_T” is written to the system time of the S7-1200 with “WR_SYS_T” and this time is stored as the last synchronization time “LAST_SYNC_TIME”. Subsequently the successful synchronization is signaled in the send data of the slave via the “synchronized” bit.

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

When disconnected by the master, the reserved connection resource with the master remains. A renewed connection request via “TCON” is therefore only required after a restart.

3 Configuration

3.1 Hardware and software installation

3.1.1 Installing and wiring the hardware

Table 3-1

No.	Instruction	Note/picture
1.	Mount the S7-1200 module to a standard top-hat rail.	
2.	Connect the controller and your programming device with the switch CSM 1277 via RJ45 Ethernet cable.	See chapter “Wiring diagram”
3.	Connect all ground connections to ground.	See chapter “Wiring diagram”
4.	Supply power to the controller.	See chapter “Wiring diagram”

3.1.2 Software installation

Table 3-2

No.	Instruction	Note/picture
1.	Install STEP 7 Basic V10.5 on your programming device.	See Table 1-1
2.	Install Service Pack 2 for STEP 7 Basic V10.5 on your programming device.	See Table 1-1

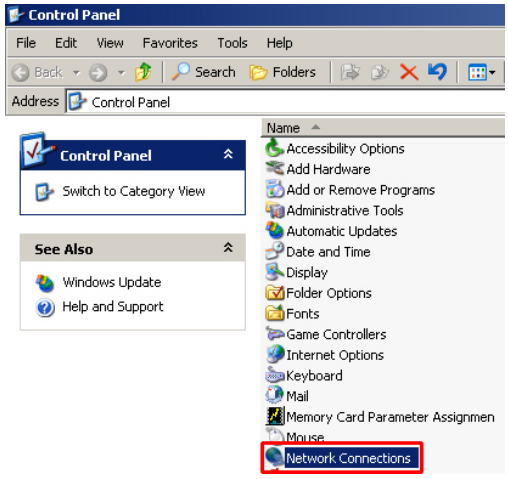
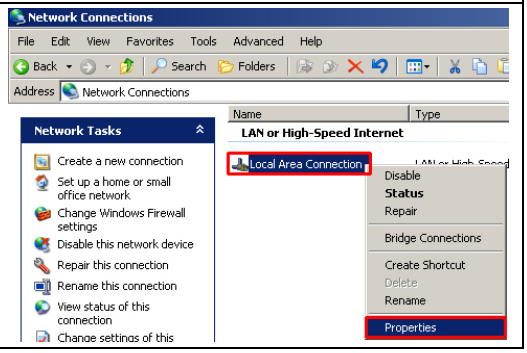
3.2 Hardware and network configuration

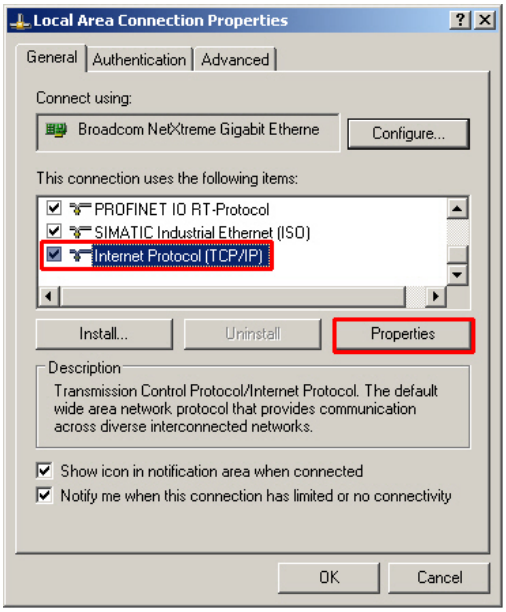
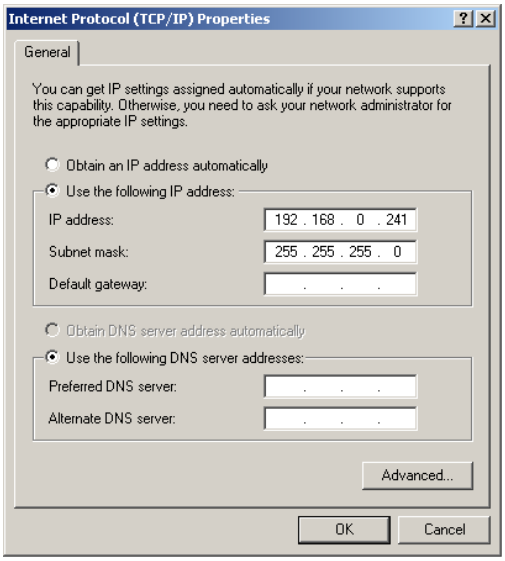
3.2.1 Assigning the IP address of the PG/PC

Your PG/PC must have an IP address assigned to it in the same subnet as the CPUs. The IP addresses of the individual nodes are displayed in Figure 1-2.

In order to assign the IP address for your network card in the Windows XP operating system, please proceed as follows:

Table 3-3

No.	Instruction	Note/picture
1.	Select the “Network Connections” option in the Control Panel.	
2.	Select the network card to be used and open the properties via right-click.	

No.	Instruction	Note/picture
3.	Select the element “Internet Protocol (TCP/IP)” and open its properties.	
4.	• Select “Use the following IP address” • Enter “192.168.0.241” as an IP address (see Figure 1-2). • Enter “255.255.255.0” as a subnet mask (see Figure 1-2). • Confirm the settings with “OK”.	

3.2.2 Configuring the master

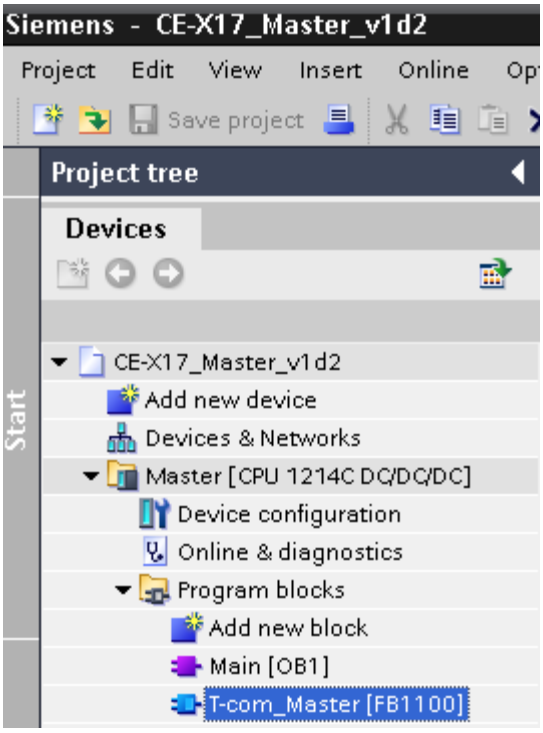
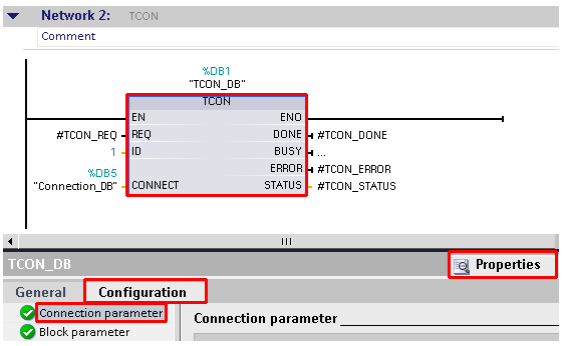
Establishing a connection

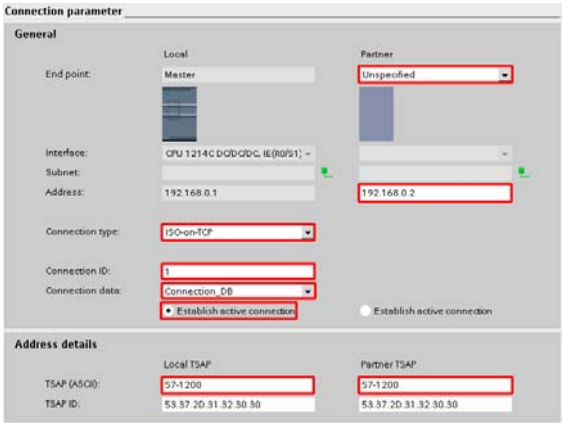
The connection parameters are stored in a connection data block. A connection ID is assigned to each connection data block. The connection block TCON accesses this connection data block. The access of the T communication blocks TRCV, TSEND and TDISCON to the connection parameters then occurs via the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

The S7-1200 offers the connection wizard for configuring the connection.

The configuration of an open IE connection with the connection wizard is described below.

Table 3-4

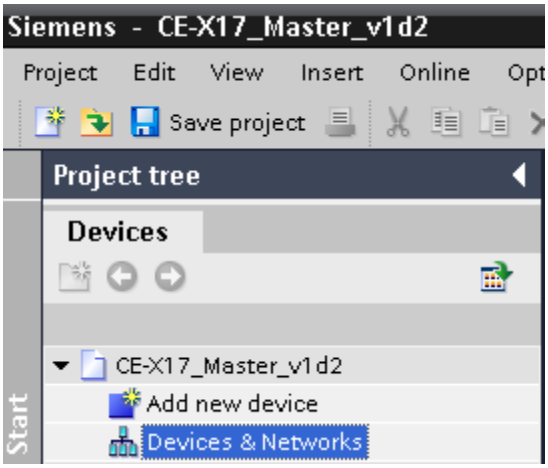
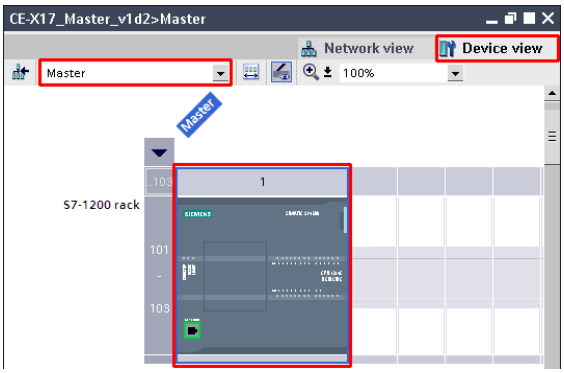
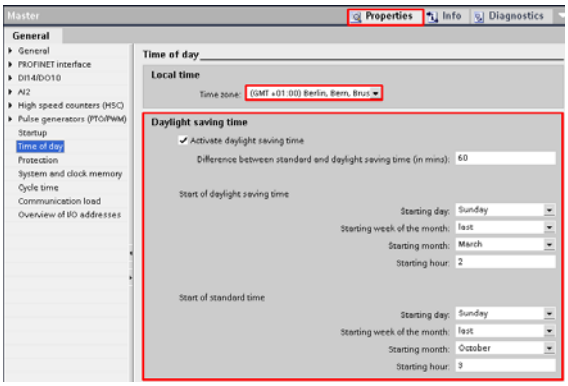
1.	• Open FB1100 “T-com_Master” in the “Master” controller of the “CE-X17_Master_v1d2.ap10” project.	
2.	• Open the Network 2. • Select the “TCON” block. • Open the “Properties” dialog. • Select “Connection parameter” from the “Configuration” tab.	

3.	• Select “Unspecified” as the end point for the partner (slave). • Enter the IP address of the first slave: • Selected the connection type “ISO-on-TCP”. • Enter “1” as the connection ID. • Select the symbolic name of the connection data block in “Connection data”. • Select “Establish active connection” on the Local side. • Select the ASCII characters “S7-1200” for the local TSAP. • For the partner TSAP-ID you also select the ASCII characters “S7-1200”.	

Time settings

The master controller synchronizes the system time of the slaves once per day. The time of synchronization is given as local time. Correct execution of the synchronization requires the time settings for the CPU.

Table 3-5

1.	• In the master project you open “Devices & Networks”.	
2.	• Open the “Device view” dialog. • Select the “Master” controller. • Select the CPU.	
3.	• In the “Properties” you select “Time of day”. • Select your “Time zone” for calculating the local time. • If necessary you activate the daylight-saving time with all required settings.	

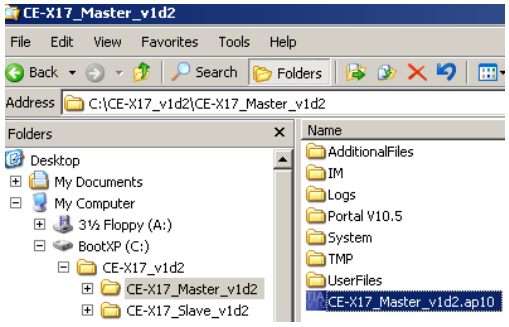
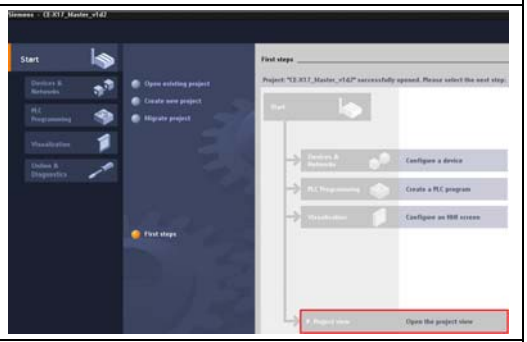
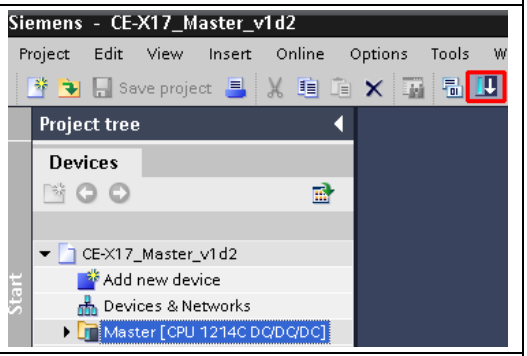
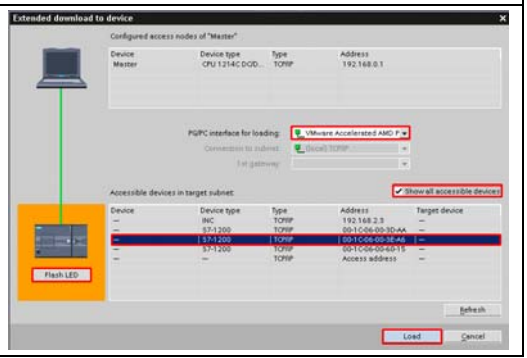
The local daily synchronization time can be set via the initial value assignment of the parameters “hour” and “minute” in FB1100 “T-com_Master” or in the watch table (see Table 2-2).

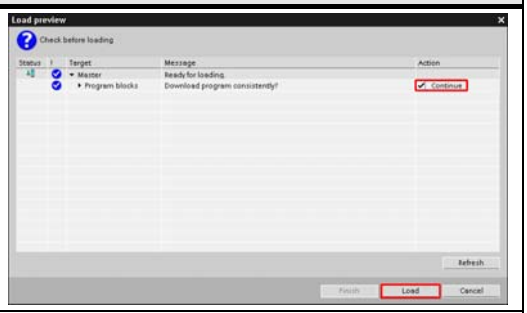
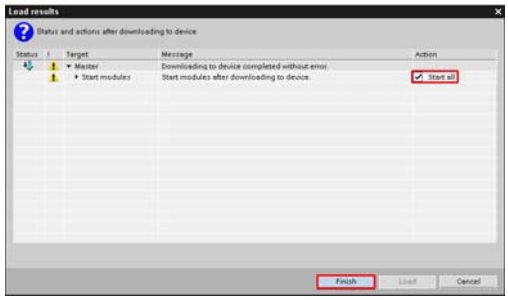
Note For S7-1200 the system time is the UTC time (Universal Time Coordinated).

Download the master project to the controller

Unzip the example program “CE-X17_Master_v1d2.zip” into any directory on your hard drive.

Table 3-6

No.	Instruction	Note/picture
1.	• In the Windows Explorer you navigate to the S7-1200 project “CE-X17_Master_v1d2.ap10” and open it via double-click.	
2.	The project is opened in STEP 7 Basic. • Open the Project view.	
3.	• Select the controller folder “Master [CPU 1214C DC/DC/DC]”. • Press the “Extended download to device” button for downloading the entire project to the controller.	
4.	• Select the used network card. • Activate the display of all accessible nodes. • Identify your master controller from the list of accessible nodes via the MAC address or via “Flash LED”. • Mark the selected controller and press the “Load” button.	

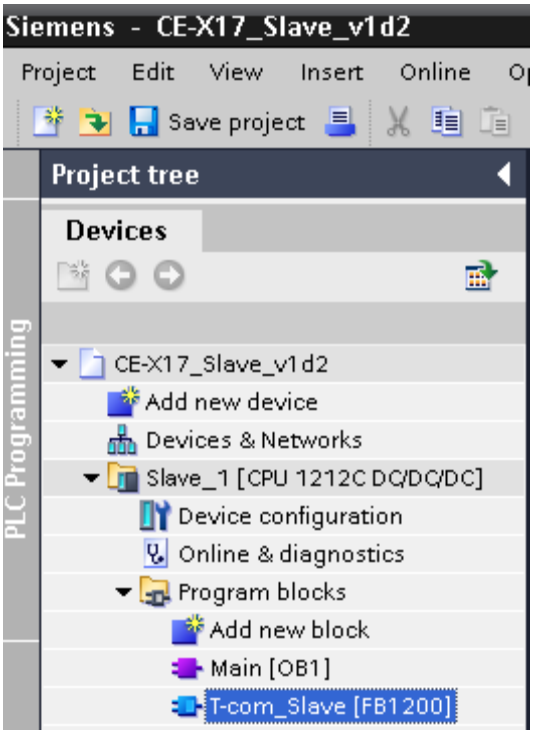
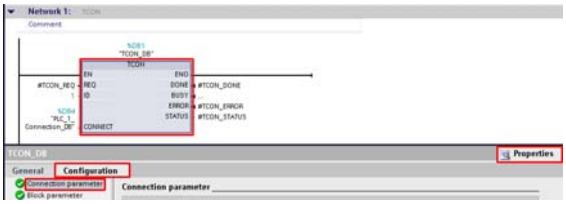
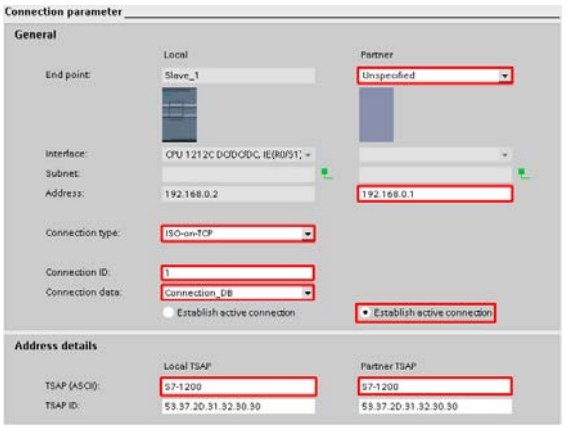
No.	Instruction	Note/picture
5.	• Activate the consistent loading of the program blocks. • Press the “Load” button.	
6.	After transferring all program blocks to the controller a window appears with the “Download result”. • Select the field “Start all” to set the controller to “Run” mode. • Terminate the download via the “Finish” button.	

3.2.3 Configuration of the slaves

Establishing a connection

The connection parameter for the slave must fit the given values of the master.

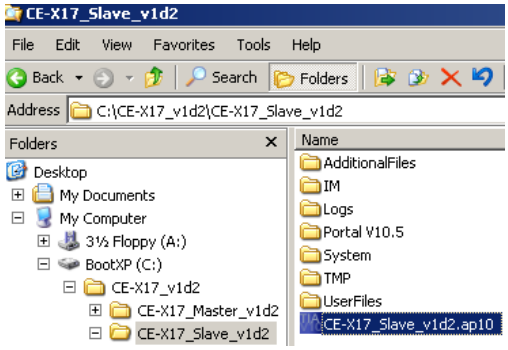
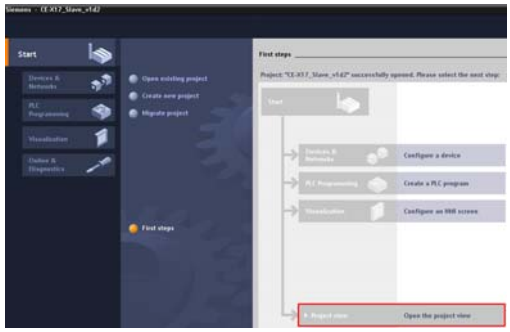
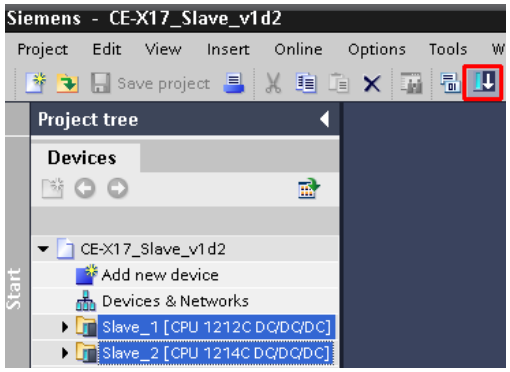
Table 3-7

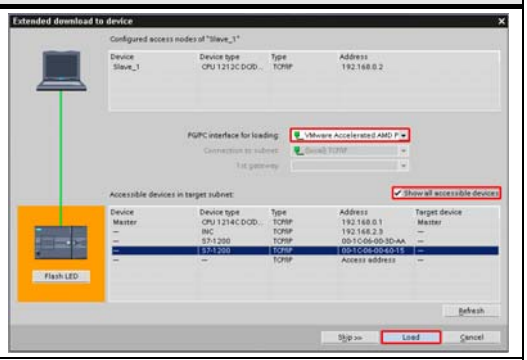
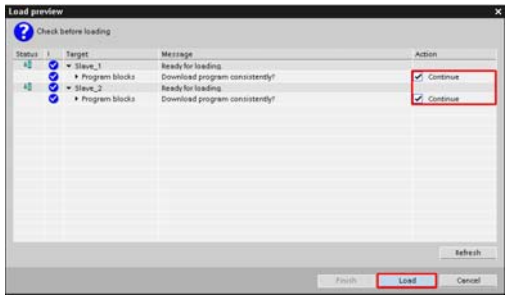
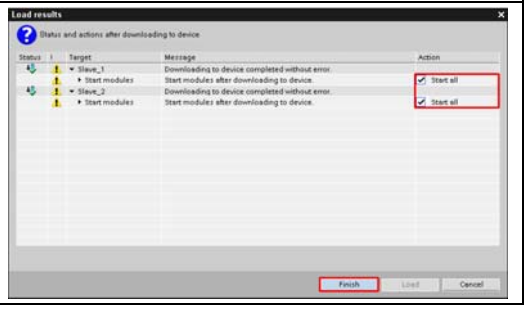
1.	• Open FB1200 “T-com_Slave” in the “CE-X17_Slave_v1d2.ap10” project under the controller for which the connection is to be established (here: “Slave_1”).	
2.	• Open the Network 1. • Select the “TCON” block. • Open the “Properties” dialog. • Select “Connection parameter” from the “Configuration” tab.	
3.	• Select “Unspecified” as the end point for the partner (S7-300). • Enter the IP address of the master. • Selected the connection type “ISO-on-TCP”. • Enter “1” as the connection ID. • Select the symbolic name of the connection data block in “Connection data”. • Select “Establish active connection” on the Partner side. • Select the ASCII characters “S7-1200” for the local and the partner TSAP.	

Download the slave project to the controllers

Unzip the example program “CE-X17_Slave_v1d2.zip” into any directory on your hard drive.
The unzipped file contains the “CE-X17_ Slave_v1d2” project for both slave controllers.

Table 3-8

No.	Instruction	Note/picture
1.	• In the Windows Explorer you navigate to the S7-1200 project “CE-X17_Slave_v1d2.ap10” and open it via double-click.	
2.	The project is opened in STEP 7 Basic. • Open the Project view.	
3.	• Select both controller folders “Slave_1 [CPU 1212C DC/DC/DC]” and “Slave_2 [CPU 1214C DC/DC/DC]”. • Press the “Extended download to device” button for downloading the entire projects to the controller.	

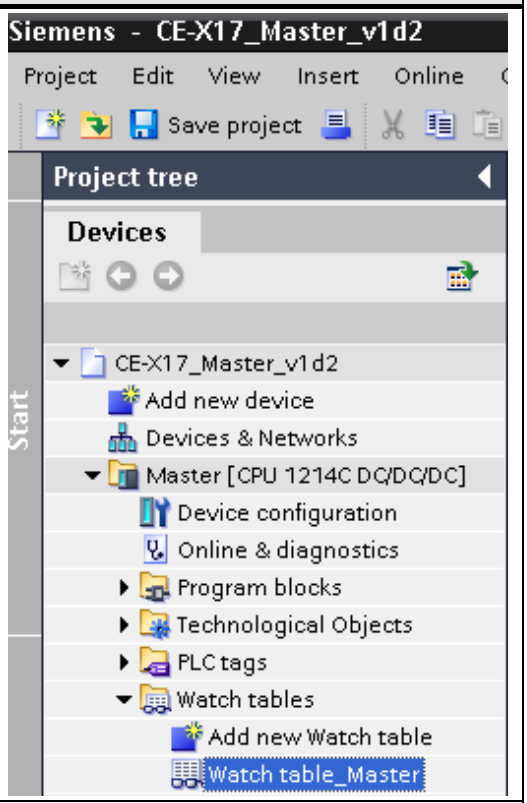
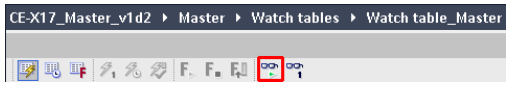
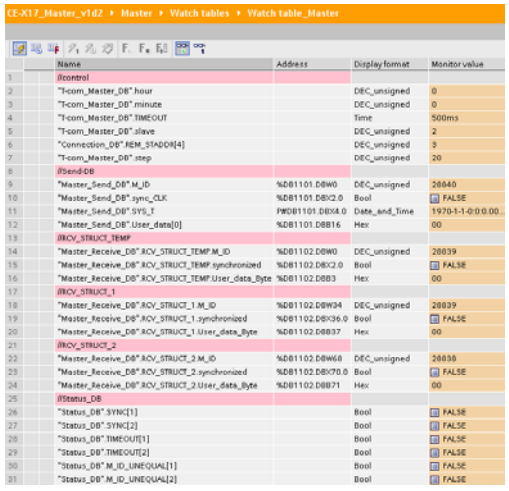
No.	Instruction	Note/picture
4.	• Select the used network card. • Activate the display of all accessible nodes. • Identify the controller “Slave_1” from the list of accessible nodes via the MAC address or via “Flash LED”. • Mark the selected controller and press the “Load” button. Repeat those two points for the download of “Slave_2”.	
5.	• Activate the consistent loading for both controllers. • Press the “Load” button.	
6.	After transferring all program blocks to the controller a window appears with the “Download result”. • Select the “Start all” fields to set both controllers to “Run” mode. • Terminate the download via the “Finish” button.	

3.3 Activate online mode

For control and monitoring of the communication your PG/PC must be switched online to the master and the slaves via the watch table.

Activate the watch table for the master

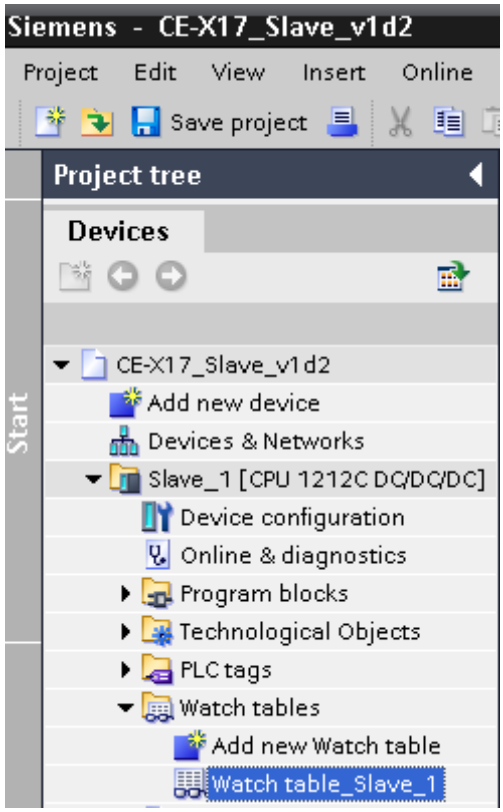
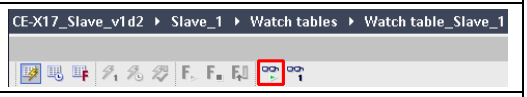
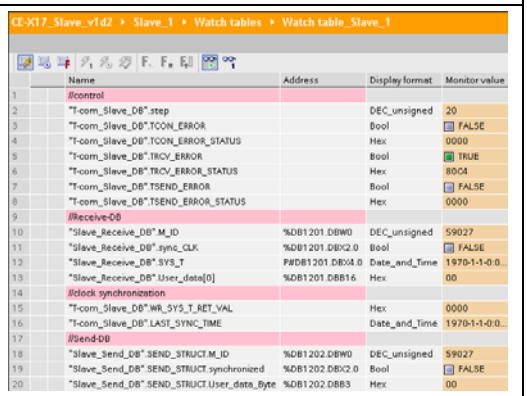
Table 3-9

No.	Instruction	Note/picture
1.	• In the project navigation of STEP 7 Basic under the controller “Master” -> “Watch tables” you open the table “Watch table_Master”.	
2.	• Activate the watch table via the “Watch all” button.	
3.	The watch table contains (line numbers in brackets): • Monitoring information (2-7) - Default local synchronization time (2-3) - Maximum processing time (4) - Currently addressed slave (5) - Currently addressed IP address (6) - Step detail (7) • Send data block (9-12) - Sent message ID (9) - Synchronization request (10) - Synchronization system time (11) - First byte of the user data (12) • Temporary receive data (14-16)	

No.	Instruction	Note/picture
	- Mirrored message ID from the slave (14) - Synchronization acknowledgement (15) - First byte of the user data (16) • Last receive data from Slave_1 (18-20) - Mirrored message ID (18) - Synchronization acknowledgement (19) - First byte of the user data (20) • Last receive data from Slave_2 (22-24) - Mirrored message ID (22) - Synchronization acknowledgement (23) - First byte of the user data (24) • Status information for both slaves (26-31) - Synchronization request (26-27) - Exceeding of processing time (28-29) - Unequal message ID (30-31)	

Activate watch table for the S7-1200 slaves

Table 3-10

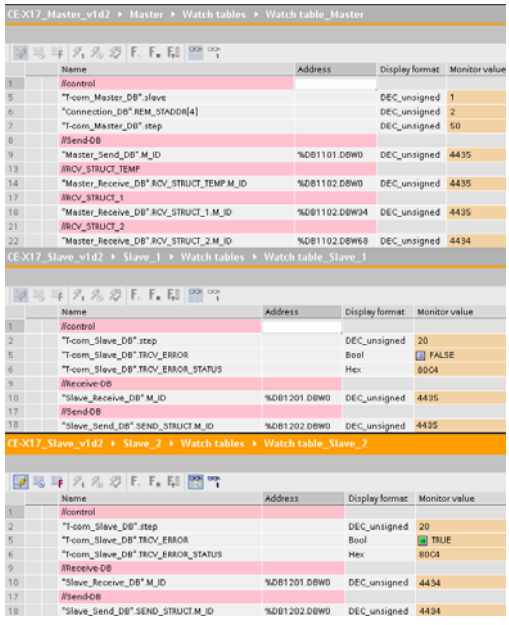
No.	Instruction	Note/picture
1.	• In the project navigation of STEP 7 Basic under the controller “Slave_1” -> “Watch tables” you open the table “Watch table_Slave_1”.	
2.	• Activate the watch table via the “Watch all” button.	
3.	Repeat step 1 and 2 for Slave_2: • Slave_2 [CPU 1214C DC/DC/DC] • Watch table_Slave_2	
4.	The watch tables contain (lines numbers in brackets): • Monitoring information (2-8) - Step detail (2) - Communication error evaluation (3-8) • Receive data block (10-13) - Message ID (10) - Synchronization request (11) - Master system time (12) - First byte of the user data (13) • Time synchronization data (15-16) - Return value of the function “Write system time” (15) - Last synchronized system time (16) • Send data block (18-20) - Mirrored message ID (18)	

No.	Instruction	Note/picture
	- Synchronization acknowledgement (19) - First byte of the user data (20)	

3.4 Live Demo

3.4.1 Cyclical sequence

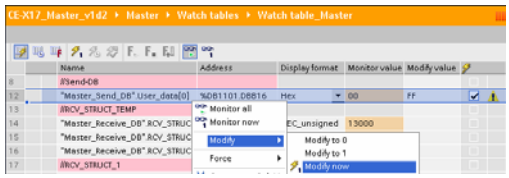
Table 3-11

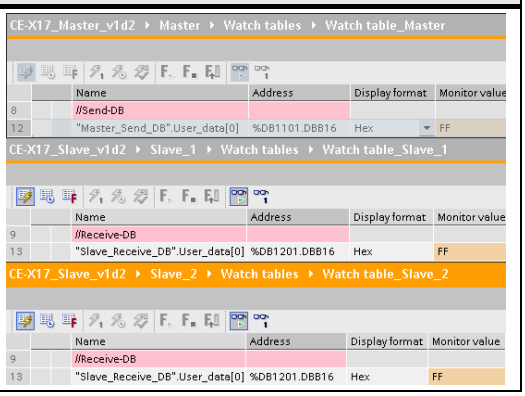
No.	Instruction	Note/picture
1.	• FB1100 “T-com_Master” of the master is called cyclically (apparent by the changed step display in line 7) • The FB1200s “T-com_Slave” of the slaves are called cyclically (apparent by the changed step display in lines 2) • The master continuously communicates with slave 1 and 2 (apparent by the changed index in line 5 and the 4th octet of the IP address in line 6). • The odd message IDs are sent to slave 1, where they are mirrored, and received again. • The even message IDs are sent to slave 2, where they are mirrored, and received again. • During the communication of the master with one slave, the receive block of the other slave displays the hexadecimal communication error “80C4” (line 5 and 6).	

3.4.2 User data transfer

Master -> Slaves

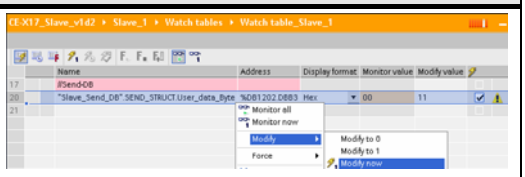
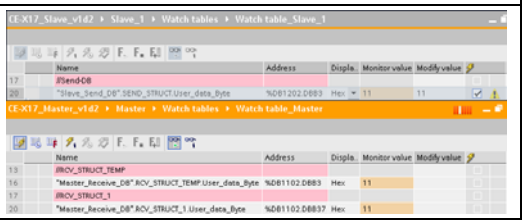
Table 3-12

No.	Instruction	Note/picture
1.	As an example for the user data transmission from the master to the slaves the send byte 0 of the user data field in line 12 shall be changed: • Enter a value in line 12 of the Modify value column. • Accept the value by right-clicking “Modify” -> “Modify now”.	

No.	Instruction	Note/picture
2.	The value is transferred at both slaves and written to the receive byte 0 of the user data field in the receive data block 1201 (apparent in lines 13 of the slave watch table).	

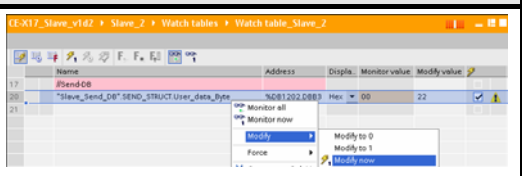
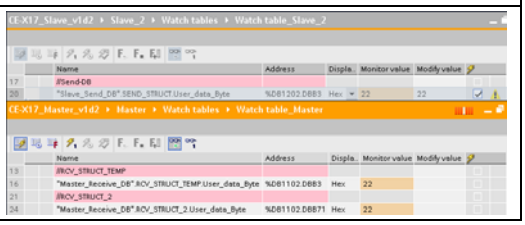
Slave 1 -> Master

Table 3-13

No.	Instruction	Note/picture
1.	As an example for the user data transmission from slave 1 to the master the user data byte of the send structure in line 20 shall be changed: • In the monitoring table “Watch table_Slave1” you enter a value in line 20 of the Modify value column. • Accept the value by right-clicking “Modify” -> “Modify now”.	
2.	The value is transferred to the master and written to the temporary receive structure (line 16), as well as to the receive structure for slave 1 (apparent in line 20 of the master watch table).	

Slave 2 -> Master

Table 3-14

No.	Instruction	Note/picture
1.	As an example for the user data transmission from slave 2 to the master the user data byte of the send structure in line 20 shall be changed: • In the monitoring table “Watch table_Slave2” you enter a value in line 20 of the Modify value column. • Accept the value by right-clicking “Modify” -> “Modify now”.	
2.	The value is transferred to the master and written to the temporary receive structure (line 16), as well as to the receive structure for slave 2 (apparent in line 24 of the master watch table).	

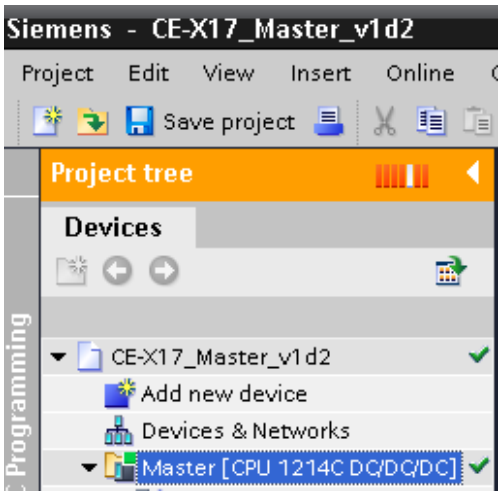
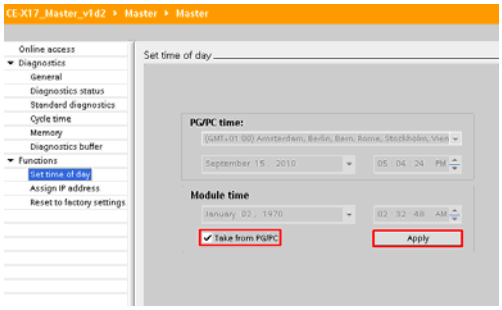
3.4.3 Time synchronization

Setting the master time

To synchronize the slaves with the system time of the master, the time of the master must be set. This can be realized manually or by adopting the local time from the PC/PG. The system time is then calculated depending on the time zone and summer/winter time settings and written to the system time of the CPU.

To set the master time proceed as follows:

Table 3-15

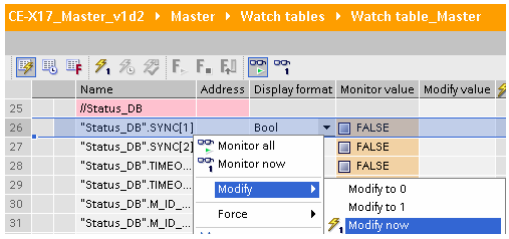
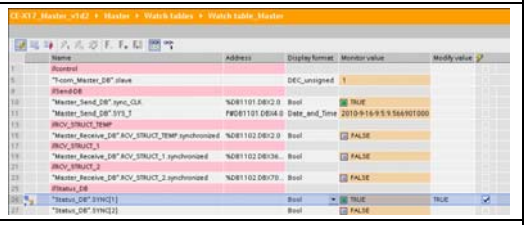
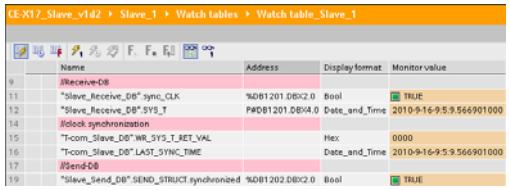
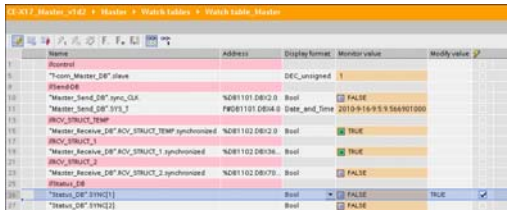
No.	Instruction	Note/picture
1.	• In the project navigation of the master project you select the master CPU.	
2.	• Open the “Online” menu and select the item “Online & diagnostics”.	
3.	• Open the “Set time of day” function. • Activate the “Take from PG/PC” option. • “Apply” the local time from the PG/PC.	

Note A more comfortable and precise setting of the time is provided by receiving the time via GPS tracking (entry ID: 45057335).

Manual synchronization of a slave

Slave 1 shall be synchronized manually with the system time of the master.
Table 3-16 shows the procedure.
With the respective procedure slave 2 can also be synchronized.

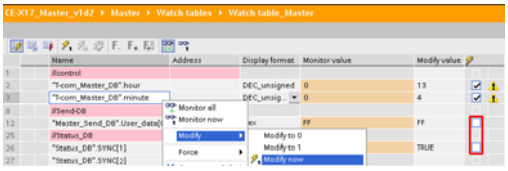
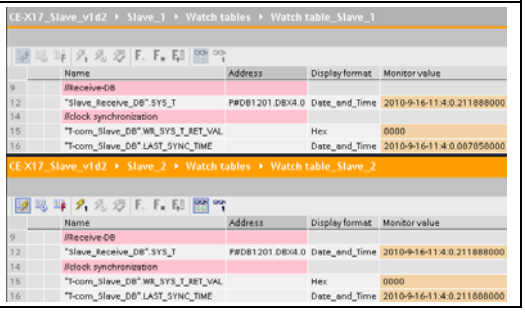
Table 3-16

No.	Instruction	Note/picture
1.	• Set the synchronization request for slave 1 in the status array “SYNC” via right-click -> “Modify” -> “Modify to 1” (line 26 in the master watch table).	
2.	• The system time is written to the send data (line 11) • The synchronization request in the send data is set (line 10) • The send data are sent to slave 1 (line 5)	
3.	• The time synchronization data are written to the receive block of slave 1 (“watch table_Slave_1”, line 11 – 12) • The synchronization time is written to the system time of slave 1 and fixed as last synchronization time (line 16). • After successful time synchronization the synchronization acknowledgement is set (line 19)	
4.	• The synchronization acknowledgement is on the master side set to the temporary receive structure (line 15) and to that of slave 1 (line 19). • The synchronization request for slave 1 is reset in the status DB (line 26)	

Automatic synchronization of all slaves

The daily local synchronization time of all slaves can be set via the initial default value in FB1100 “T-com-Master” or via the watch table.
The respective parameters “hour” and “minute” are kept remnant.

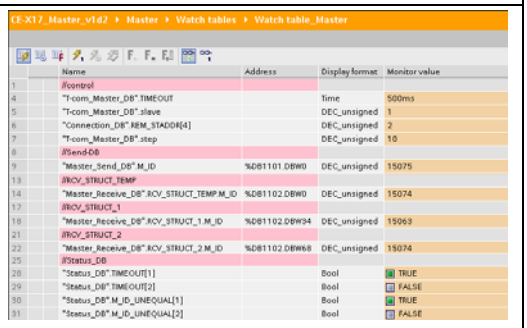
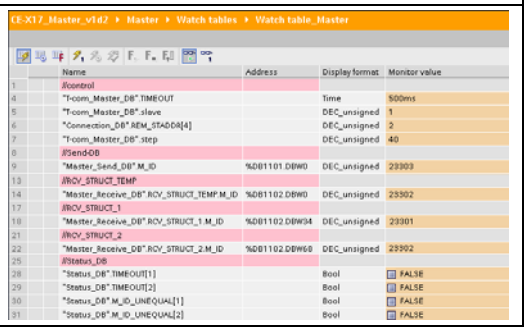
Table 3-17

No.	Instruction	Note/picture
1.	• Deactivate the checkboxes of all previously modified tags (line 12 and 26). • Enter a local synchronization time in the close future to the control values in line 2 “hour” and line 3 “minute” (here: 13:04). • Accept the values by right-clicking “Modify” -> “Modify now”.	
2.	• The successful time synchronization of the slaves can be checked via the last written system time of the slaves (line 16 in the watch tables “Watch table_Slave_1” and “Watch table_Slave_2”). • Due to the time zone and the summer time the local time in the represented case differs by 2 hours from the system time. The last synchronized system times therefore show 11:04.	

3.4.4 Communication errors

Pulling the Ethernet cable from slave 1 the communication error evaluation shall be demonstrated.
Table 3-18 shows the procedure.
With the respective procedure a communication interruption with slave 2 can be simulated and evaluated.

Table 3-18

No.	Instruction	Note/picture
1.	Pull the Ethernet cable from the LAN connection of slave 1.	
2.	• The step chain stops (line 7) for the communication with slave 1 (line 5) and waits for a confirmation of an established connection from slave 1. • After a timeout of 500 milliseconds (line 4) the step-chain is switched forward and the timeout message appears in the TIMEOUT array for slave 1 (line 28). • Additionally the sent message ID (line 9) being unequal with that last received by slave 1 (line 18) is detected and output in line 30. • Data exchange with slave 2 still occurs without error (line 29 and 31). • The renewed connecting attempt with slave 1 delivers the same error image as long as slave 1 cannot be reached.	
3.	Reconnect the Ethernet cable with the LAN connection of slave 1.	
4.	• After detecting the reconnection the error bits in the status arrays for slave 1 are reset (line 28 and 30). • Data exchange with slave 1 has been restored.	

3.4.5 Synchronization time of the master

After voltage recovery of the master the step chain of the function block 1100 “T-com_Master” starts reconnecting with the last connected slave.

4 Code Elements

In the example on hand the following program codes are used.

Table 4-1

No.	File name	Contents
1.	CE-X17_Master_v1d2.zip • CE-X17_Master_v1d2.ap10	Zip-file with the S7-1200 master project for the deterministic T communication
2.	CE-X17_Slave_v1d2.zip • CE-X17_Slave_v1d2.ap10	Zip-file with the S7-1200 slave project for the deterministic T communication

5 History

Table 5-1

Version	Date	Changes
V1.0	15.10.2009	Manual communication via the integrated T-blocks TSEND_C and TRCV_C (task A) and via conventional T-blocks TCON, TSEND, TRCV and TDISCON (task B)
V1.1	10.02.2010	Supplementation in chapter 2.3.2: Symbolic addressing
V1.2	17.09.2010	Changing the task in the deterministic data exchange via the conventional T-communication blocks (TCON, TSEND, TRCV und TDISCON)