# S7 Communication:
# Data Exchange S7-300 <-> 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 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 for damages arising from a breach of a duty under this 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 questions regarding this document, please send an e-mail to:

online-support.automation@siemens.com

# Table of Contents

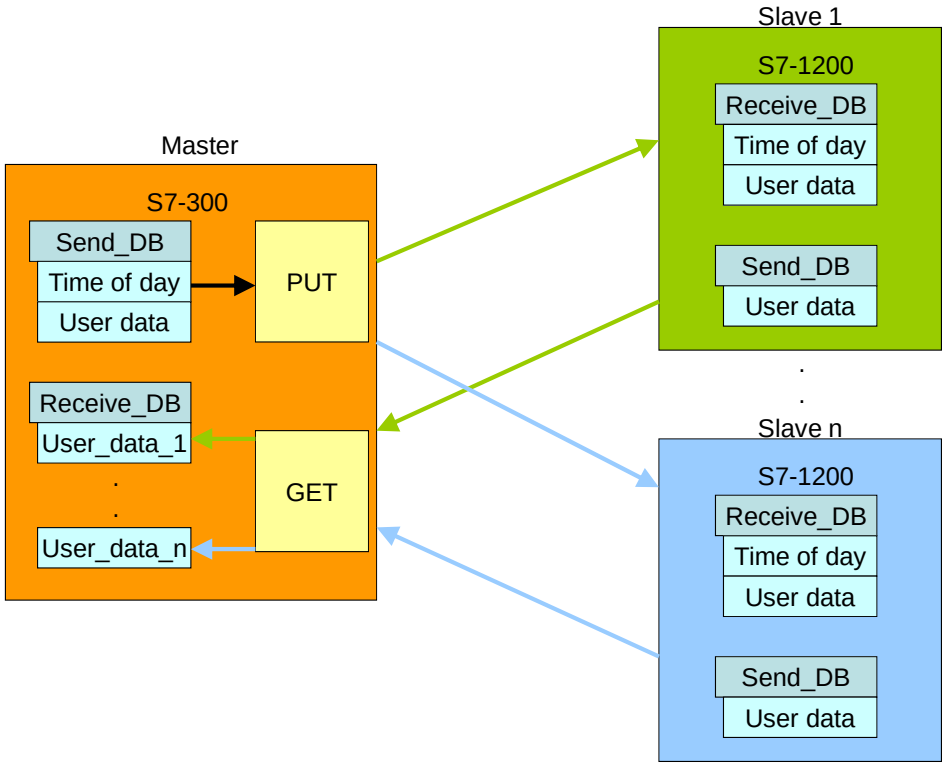
# 1 Automation Task

## 1.1 Task description

Data exchange (e.g. for time synchronization) shall be effected in a deterministic approach via Ethernet with the help of S7 communication between an S7-300 master and several S7-1200 slave controllers.

### Scheme of the application task

Figure 1-1




### Requirements of the application

The master, as well as the slaves include a send and a receive block (Send_DB and Receive_DB). After receipt of the synchronization command, the master reads the system time and sends this information and the user data to the first slave via the PUT block for S7 communication. The PUT block synchronizes its own system time with the time-of-day information received from the master.

Then the master polls the user data of slave 1 via the GET communication block. This user data of slave 1 is then stored at the relevant location in the master's receive block.

This procedure is repeated for all subsequent slave units. After data exchange between master and the last slave is completed, the master unit starts data exchange with slave 1 again.

# 1.2 Structure

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

### Schematic structure

Figure 1-2

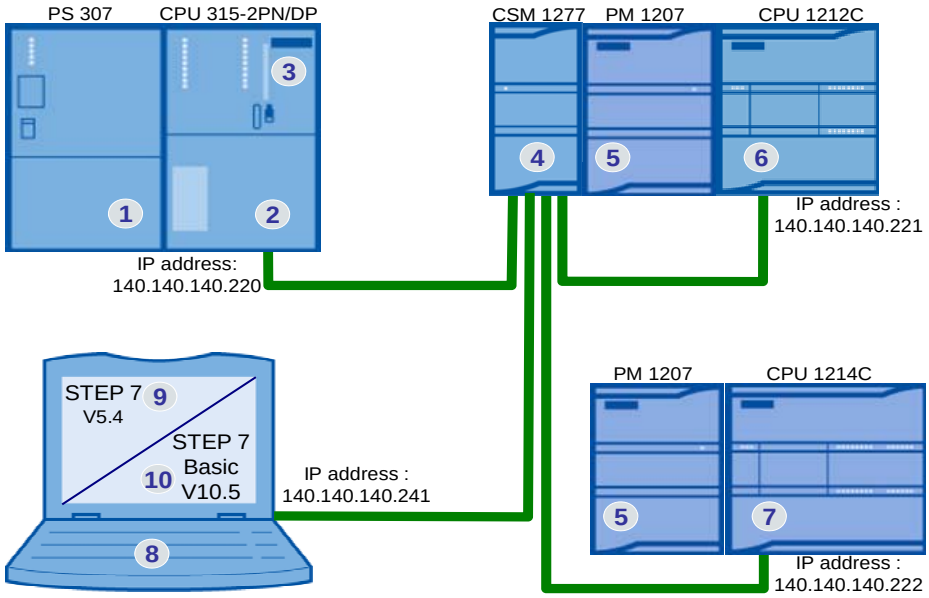



Figure 1-2 shows the principle of this structure. The communicating CPUs, as well as the programming unit with the softwares "STEP 7 Basic V10.5" for S7-1200 programming and "STEP 7 V5.4" for S7-300 programming are connected to the switch CSM 1277 by means of Ethernet cables.

### 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, 8MB | 1 | 6ES7953-8LP10-0AA0 |
| 4. | COMPACT SWITCH MODULE CSM 1277 | 1 | 6GK7277-1AA00-0AA0 |
| 5. | POWER SUPPLY S7-1200 PM1207 | 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 | Entry ID:36184684 |
| 11. | STEP 7 BASIC V10.5 | 1 | 6ES7822-0AA00-0YA0 |
| 12. | STEP 7 Basic V10.5 Service Pack 2 | 1 | Entry ID:39741113 |

# 2 Automation Solution

The S7-1200 PLC offers the passive server functionality for the S7 communication. In doing so, the S7-1200 allows read-and-write access to the data.

Configuration is performed by the S7-300 client via the PUT and GET blocks. The PUT block is used to write data from the S7-300 to the S7-1200 and the GET block retrieves data from the S7-1200 and writes them to the S7-300.

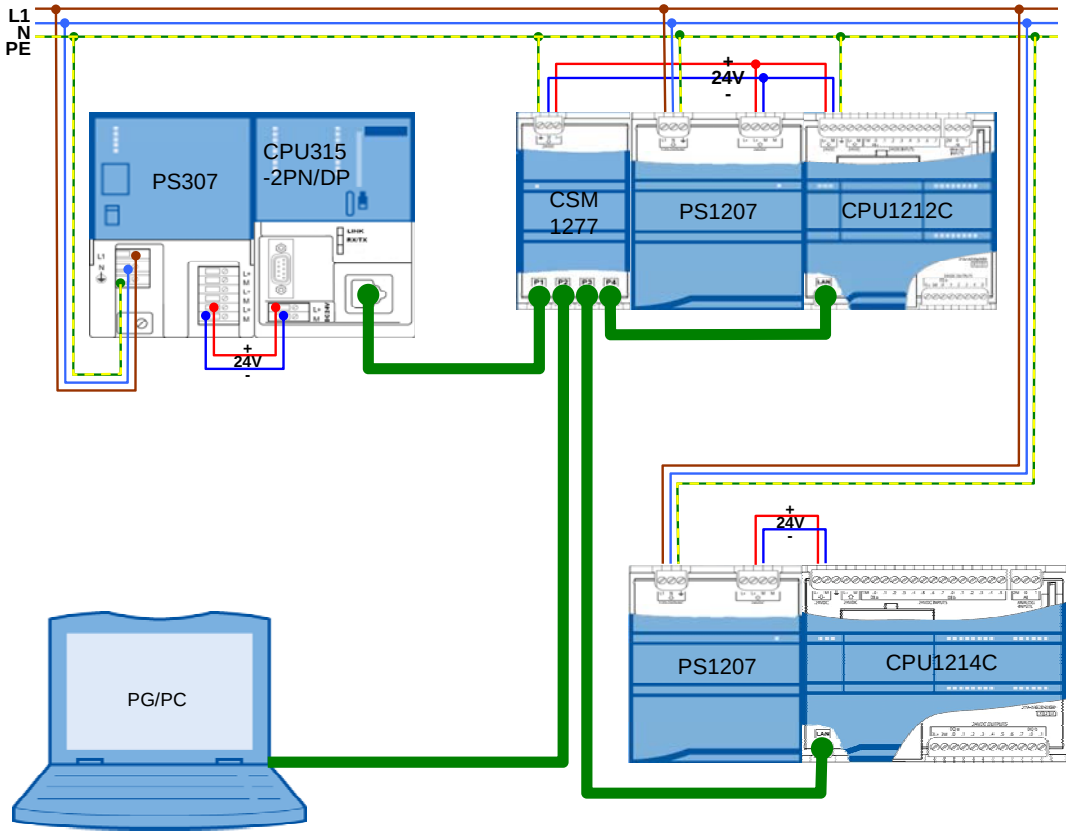
The connection is configured in STEP 7 V5.4 in NetPro. The relevant connection partner is defined by the specific IP address. Exactly one ID is assigned for each connection to an S7 server. This ID is then transferred to the S7 communication blocks PUT and GET.

The maximum number of configurable connections in NetPro depends on the type of S7-300 CPU used. The CPU 315-2 PN/DP is suitable for a maximum of 14 S7-connections in NetPro.

## 2.1 Cabling diagram

Please also refer to the list of components in Chapter 1.2.

Figure 2-1



# 2.2 Program structure

This chapter describes the program structure used in this example for the automation system on function and data block level.

## 2.2.1 Overview of the block structure

Figure 2-2 and Figure 2-4 show the block-call hierarchy, as well as the access to the data blocks used for the S7-300 client and the S7-1200 servers.

Figure 2-2

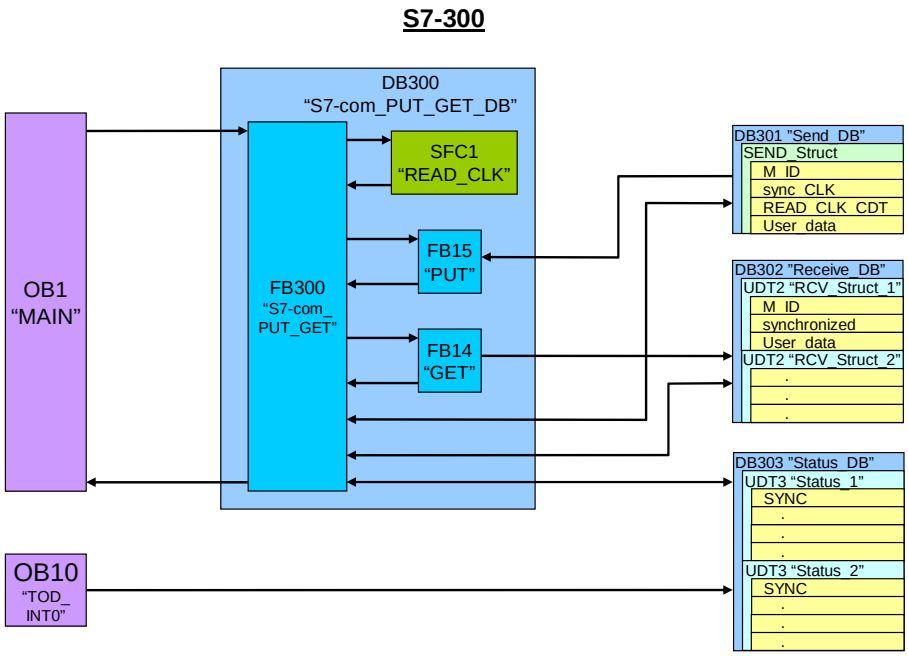
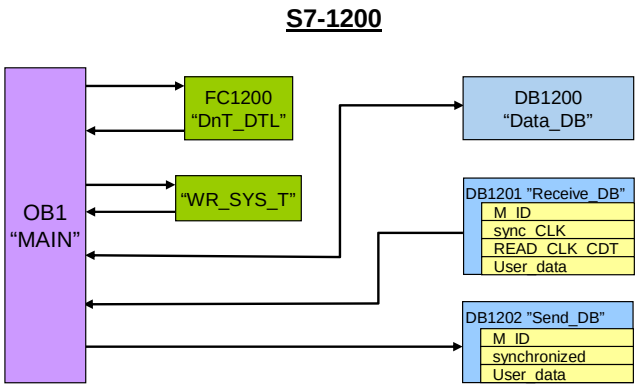



Figure 2-3



### 2.2.2 Description of the block structure

The time-of-day interrupt OB10 “TOD_INT0” in the hardware structure of the CPU 315-2PN/DP is activated and set to execution at daily intervals.

The “Status_DB” data block DB303 contains the control and status information of all S7-1200 servers in the form of the data type UDT3 “STATUS”. Apart from the “SYNC” synchronization request bit, this data type structure also includes information for the analysis of communication errors.

When the time-of-day interrupt OB10 “TOD_INT0” is executed, all “SYNC” synchronization request bits will be set. Time synchronization, however, may also be executed individually for each server by means of the table of variables.

OB1 calls the function block FB300 “S7-com_PUT_GET” at cyclic intervals by using its instance DB300 “S7-com_PUT_GET_DB”.

When the synchronization request “SYNC” of the first S7-1200 server is set in the DB303 “Status_DB”, the PLC time will be read with the help of SFC1 “READ_CLK” and stored in the send data block DB301 “Send_DB”, together with the synchronization request.

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

Function block FB300 calls send block FB14 “PUT”. This block transmits the contents of the send data block DB301 to the receive data block DB1201 “Receive_DB” of the first server. Apart from the information for time synchronization, transmission also includes the "User_data" and a message ID “M_ID”.

When a synchronization request “sync_CLK” is issued, OB1 “MAIN” of the server calls the function FC1200 “DnT_DTL”. This function is used to convert the “READ_CLK_CDT” time information of the S7-300 clients of a DATE_AND_TIME type into a DTL data type. All variables are stored in the DB1200 “Data_DB”. The function “WR_SYS_T” is used to write the converted time information into the system time of the S7-1200. After successful time synchronization, the “synchronized” bit in the send DB1202 “Send_DB” is set.

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

After data transmission with the help of the “PUT” communication block, the contents of send data block DB 1202 “Send_DB” are retrieved from the first server via “GET” and then stored in the relevant data type UDT2 “RCV_Struct_1” in the “Receive_DB” DB302.

The message ID “M_ID” received is then compared with the ID sent. Any discrepancy will be stored in the status DB in the relevant data type structure "Status_1" for server 1.

After successful synchronization of server 1 (signalled by the “synchronized” variable), the synchronization request bit “SYNC” in “Status_1” of the status DB will be reset.

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

## 2.3 Blocks used

The following tables provide an overview of all blocks used on the client and server side.

### 2.3.1 Client (S7-300)

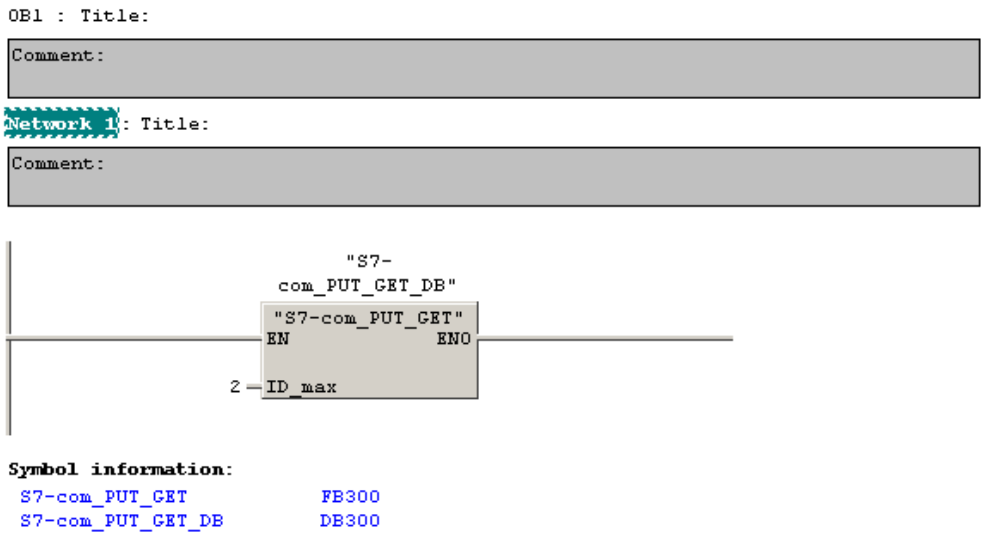
Table 2-1

| Object name | Symbolic name | Description |
|---|---|---|
| OB1 | MAIN | Cyclic organization block |
| OB10 | TOD_INT0 | Time-of-day interrupt |
| FB14 | GET | S7 communication block for data retrieval |
| FB15 | PUT | S7 communication block for data transfer |
| FB300 | S7-com_PUT_GET | Function block for deterministic data exchange with several servers via the S7 communication blocks PUT and GET |
| FC301 | read_bit | Function to read the value of a bit by using a pointer (used in FB300) |
| FC302 | reset_bit | Function to reset a bit by using a pointer (used in FB300) |
| FC303 | output_bit | Function for the output of a value by using a pointer (used in FB300) |
| DB300 | S7-com_PUT_GET_DB | Multi-instance data block for FB300, FB14 and FB15 |
| DB301 | Send_DB | Send data block for FB15 |
| DB302 | Receive_DB | Receive data block for FB14 |
| DB303 | Status_DB | Status data block for all servers |
| UDT2 | RCV_STRUCT | Data type structure for the receipt of server data |
| UDT3 | STATUS | Data type structure for the status information of the servers |

### S7-com_PUT_GET (FB300)

This function block is used for a deterministic data exchange with several servers via the S7 communication blocks PUT and GET; it is called at cyclic intervals in OB1.

Figure 2-4




DB300 has been selected as instance data block. It also includes the instances for the S7 communication blocks PUT and GET.

As being the only input, the maximum number of servers “ID_max” must be stated. For the CPU 315-2PN/DP used in this example, a maximum of 14 S7 connections can be configured in NetPro.

The index variable used to identify the relevant server is the connection ID. Data exchange with the servers is performed in a sequential manner.

**WARNING**

**A dynamic change of the ID for the S7 communication blocks PUT and GET is supported only by the S7-300 controller.**
**An S7-400 controller requires a static ID for each communication block.**

The following static variables of FB300 offer configuration options via the initial value or the status analysis 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 receive data structure UDT2 in bytes |
| Status_DB | Int | Number of the status data block |
| Status_size | Int | Size of the status data structure UDT3 in bytes |
| READ_CLK_ERROR | Bool | Error output of the block READ_CLK (SFC1) |
| READ_CLK_RET_VAL | Int | Status of the block READ_CLK (SFC1) |

**Status_DB (DB303)**

This status DB consists of 14 data type structures UDT3 STATUS for a maximum of 14 servers for communication with the CPU 315-2PN/DP. The structure includes 6 bytes as follows:

Table 2-3

| Name | Data type | Description |
|---|---|---|
| SYNC | Bool | Time synchronization request |
| PUT_ERROR | Bool | Error message of the PUT communication block |
| GET_ERROR | Bool | Error message of the GET communication block |
| M_ID_UNEQUAL | Bool | Discrepancy in the M_IDs received and sent |
| PUT_ERROR_STATUS | Word | Status of the PUT block when the last error occurred |
| GET_ERROR_STATUS | Word | Status of the GET block when the last error occurred |

## 2.3.2 Server (S7-1200)

Table 2-4

| Object name | Symbolic name | Description |
|---|---|---|
| OB1 | Main | Cyclic organization block |
| FC1200 | DnT_DTL | Function to convert the data type DATE_AND_TIME into a DTL data type |
| DB1200 | Data_DB | Variables data block |
| DB1201 | Receive_DB | Data block for reception from a client |
| DB1202 | Send_DB | Data block for transmission to a client |



## 2.3.3 Data consistency

**DB301 and DB1201**

The send block of the client and the receive block of the server must be identical in length and structure. In this application example, they consist of 160 bytes with the following structure:

Table 2-5

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

### DB302 and DB1202

The receive structure RCV_STRUCT (UDT2) of the client and the send block of the server must be identical. The receive DB 302 consists of 14 receive structures for the maximum number of server connections for the CPU 315-2PN/DP. The receive structure UDT2 or the send DB 1202 consist of 160 bytes with the following structure:

Table 2-6

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

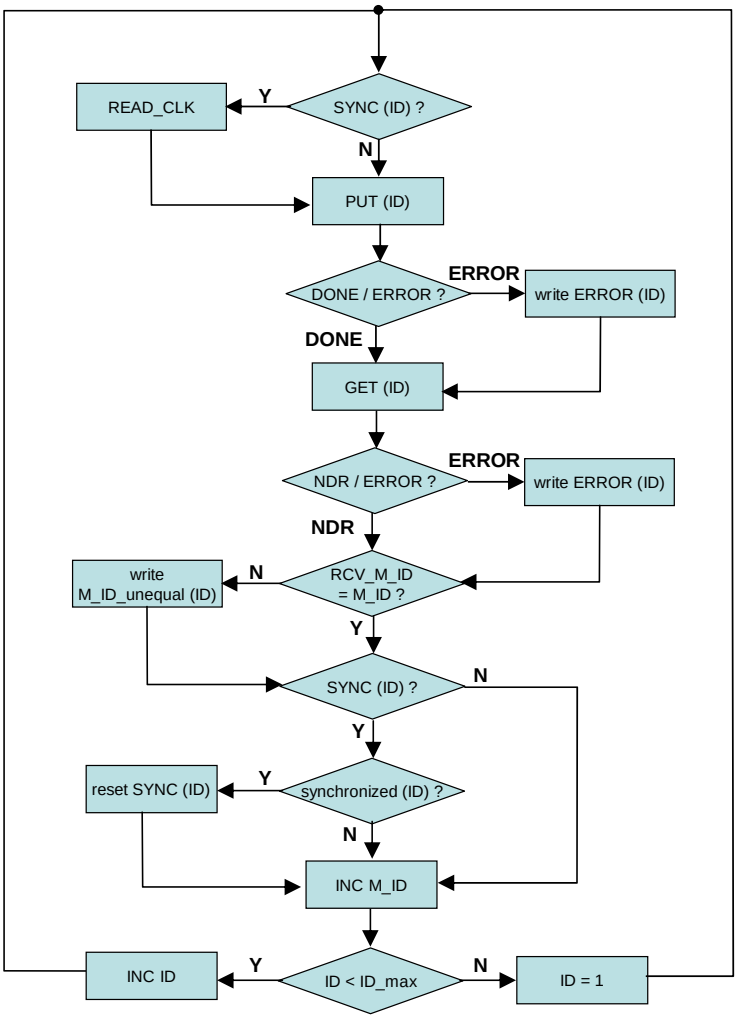
The "User_data" can be adapted individually. The data structure, however, must be identical on the sender and receiver side.
Program-related data consistency is ensured through sequential processing of the send and receive jobs.
The status DB 303 offers direct influence on communication errors.
Through the continuous data exchange between the client and the servers, data consistency can be ensured only for one cycle.
Consequently, the consistent data must be written to the send data blocks, or be read from the receive data blocks, respectively, within one cycle.

## 2.4 Program sequence in the client station

### Flowchart

The flowchart below shows the program sequence in the client station. The functionality is combined in FB300 “S7-com_PUT_GET” which is called by OB1 at cyclic intervals. FB300 is realized in the form of a step sequence.

Figure 2-5



**Description of the flowchart**

The “ID” is the index used to identify the relevant server to be used for data exchange.

Depending on the “ID”, the synchronization request “SYNC” is read from the status information for the server “ID” of “Status_DB” 303. Depending on the request, the system time (“READ_CLK”) is read and then written to the send data block.

The send block “PUT” is used to transmit the contents of the send data block to the “ID” server. Apart from the time synchronization information, a message ID “M_ID” is transmitted also.

In case of an “ERROR” message from the “PUT” send block, the error information is written to the status structure of the “ID” server in the status DB.

The receive data block “GET” is used to receive the data from the “ID” server and to write them into the “ID” receive structure in the receive DB.

In case of an “ERROR” message from the “GET” receive block, the error information is written to the status structure of the “ID” server in the status DB.

On the basis of this receive data, the message ID “RCV_M_ID” mirrored by the server is compared with the “M_ID” sent. Any discrepancies will be stated in the status structure of the “ID” server in the status DB (“M_ID_unequal”).

When a synchronization request “SYNC” of the “ID” server is issued, successful synchronization is checked on the basis of the receive data of the “ID” server (“synchronized”). If the result is positive, the synchronization request “SYNC” for the “ID” server will be reset. Otherwise, time synchronization will be repeated during the next communication with this server.

The message ID is increased (“INC M_ID”) and the “ID” is compared with “ID_max”, i.e. the maximum number of servers. The ID continues to increase (“INC ID”) until “ID_max” is reached. Otherwise the ID will be reset to the initial ID (“ID = 1”).

# 3 Configuration

## 3.1 Hardware and software installation

### 3.1.1 Hardware installation and cabling

Table 3-1

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Mount the S7-1200 modules to a “top-hat” DIN rail. | |
| 2. | Mount the S7-300 modules to a S7-300 mounting rail. | |
| 3. | Us an RJ45 Ethernet cable to connect the controllers and the associated programming unit with the switch CSM 1277. | See Chapter “Cabling diagram” |
| 4. | Connect all grounding terminals to ground. | See Chapter “Cabling diagram” |
| 5. | Connect the controllers to voltage. | See Chapter “Cabling diagram” |
| 6. | Insert the MICRO MEMORY CARD into the CPU 315-2PN/DP. | See Table 1-1 |



### 3.1.2 Software installation

Table 3-2

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Install the software STEP 7 BASIC V10.5 on your programming unit. | See Table 1-1 |
| 2. | Install the 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 the Service Pack 5 for STEP 7 V5.4 on your programming unit. | See Table 1-1 |

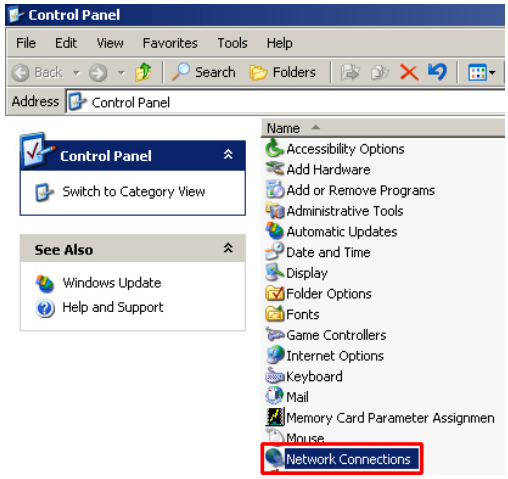
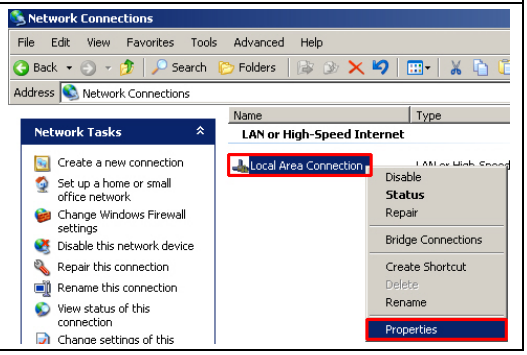
# 3.2 Hardware and network configuration

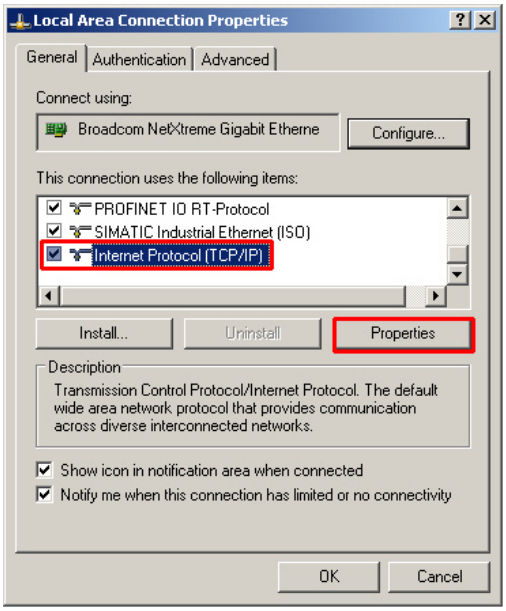
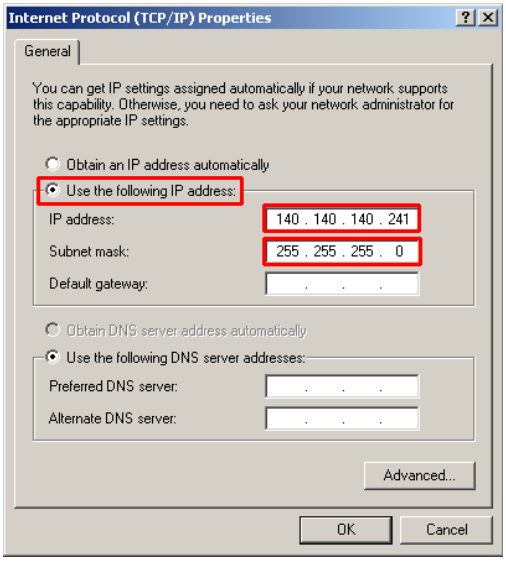
## 3.2.1 Assignment of an IP address for the PG/PC

Your PG/PC must be allocated an IP address in the same subnetwork as the CPUs. The IP addresses of the individual stations can be seen in Figure 1-2.

Proceed as described below to assign an IP address for your network card in the Windows XP operating system:

Table 3-3

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Open the Windows Control Panel and select “Network Connections”. |  |
| 2. | Select the network card to be used and click your right mouse button to open the associated “Properties”. |  |

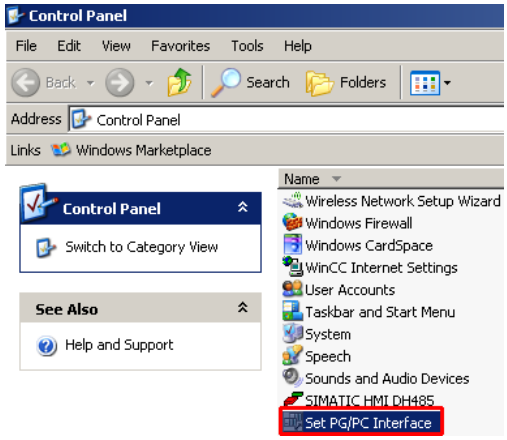
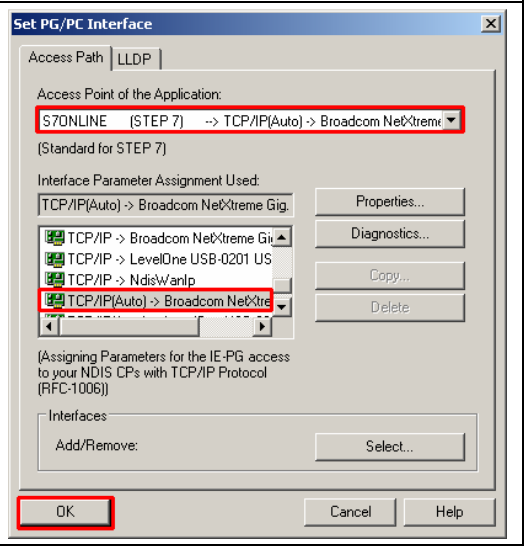
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 3. | Select the element “Internet Protocol (TCP/IP)” and click the “Properties” button. |  |
| 4. | • Select “Use the following IP address”<br>• Enter the IP address “140.140.140.241” (see Figure 1-2).<br>• Enter the subnet mask “255.255.255.0”.<br>• Click “OK” to confirm your settings. |  |

## 3.2.2 Configuration of the S7-300

### Configuring the PG/PC interface

For project download and online communication with the CPU 315-2PN/DP, the PG/PC interface must be configured as follows:

Table 3-4

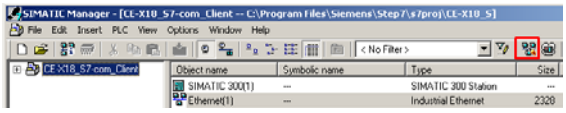
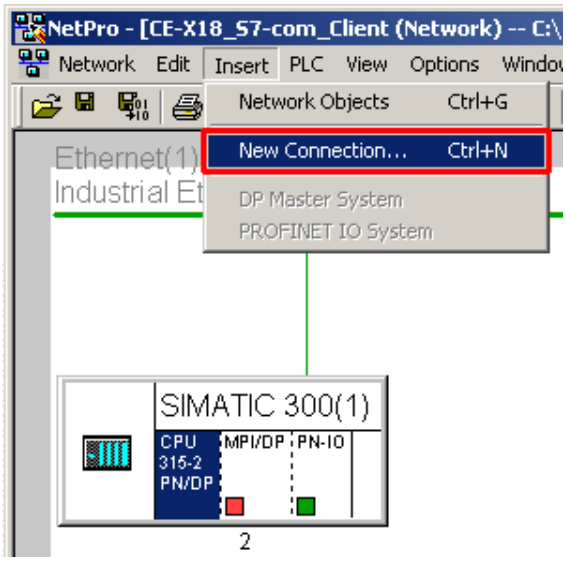
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Open the Windows Control Panel and select “Set PG&PC Interface”. |  |
| 2. | • Select “S7ONLINE (STEP 7)” as access point of the application.<br>• Select “TCP/IP(Auto)” as the interface parameter assignment to be used for your network card.<br>• Click “OK” to confirm your settings. |  |

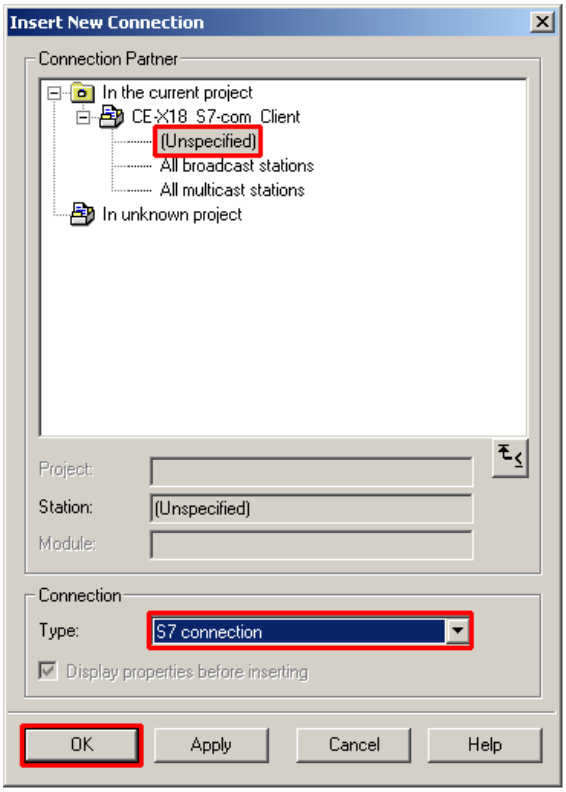
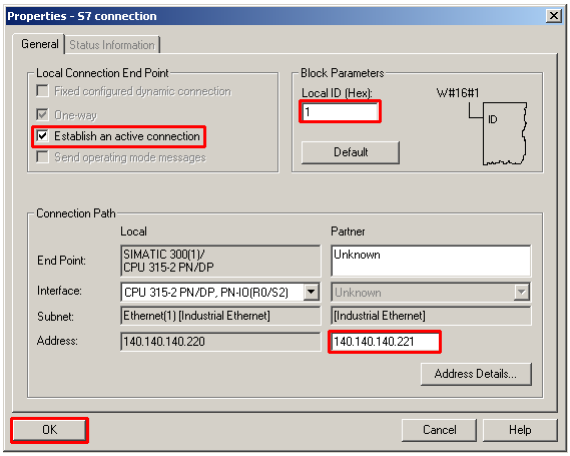
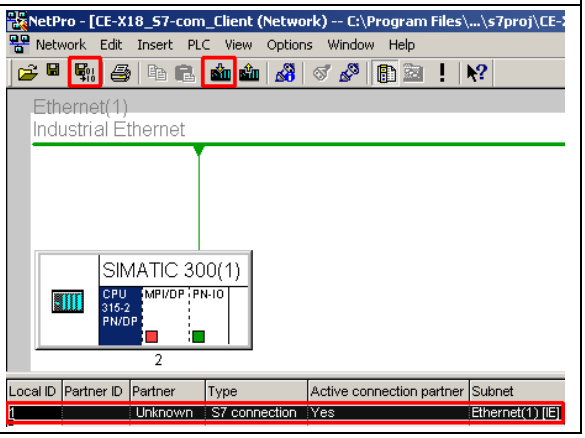
### Creating a new connection

The S7-1200 PLC offers a passive server functionality for S7 communication via Ethernet. Communication is configured only on the S7-300 via the PUT and GET blocks. Consequently, the connection is also configured only in STEP 7 V5.4 in NetPro. This requires the IP addresses of the S7-1200 servers.

The table below describes how an S7 connection is configured in NetPro.

Table 3-5

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Open the SIMATIC Manager and select the project “CE-X18A_Client_v1d2”.<br>• Click the “Configure Network” button to open NetPro. |  |
| 2. | • Select the CPU in the NetPro window and use the menu command “Insert” to create a “New Connection”. |  |

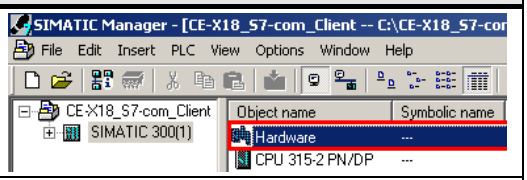
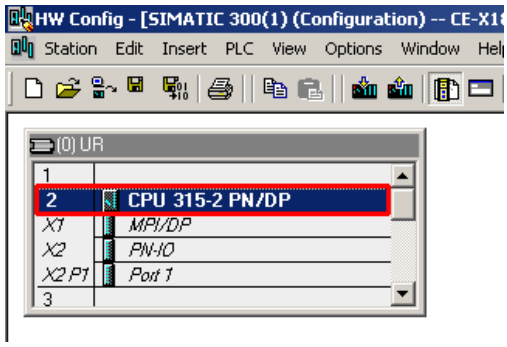
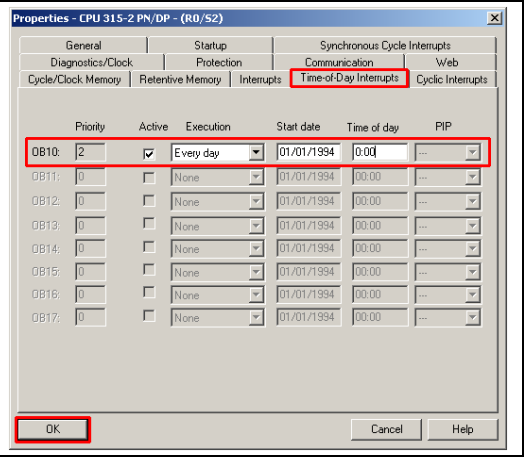
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 3. | • Select an unspecified “S7 connection” as connection partner and click “OK” to confirm your settings. |  |
| 4. | • Open the “General” tab in the “Properties” dialog window and activate the option “Establish and active connection”.<br>• The connection ID for transfer to the PUT and GET blocks is predefined, however, it may also be allocated individually as available.<br>• Enter the IP address “140.140.140.221” of the first S7-1200 server (see Figure 1-2).<br>• Click “OK” to confirm your settings. |  |
| 5. | • In NetPro, the defined connection is shown next to the highlighted CPU.<br>• Click the “Save and Compile” button.<br>• Click the “Download the selected station(s)” button to load the modified network configuration. |  |

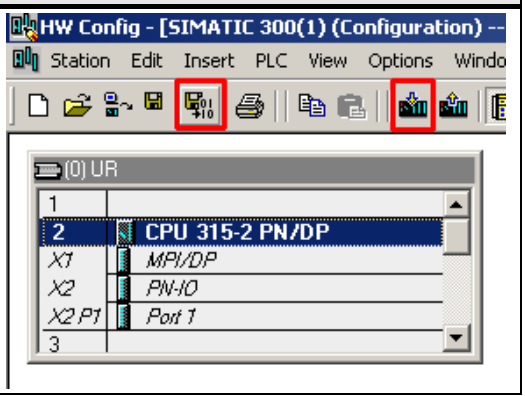
### Setting the synchronization time

The client CPU 315-2PN/DP is the clock master for synchronization of the S7-1200 servers. The interval for time synchronization can be set by configuration of the time-of-day interrupt. Apart from the automatic synchronization, each server may also be synchronized manually via the table of variables.

**Note** The UTC time (Universal Time Coordinated) is the system time of both the S7-300 and the S7-1200 PLC.

Table 3-6

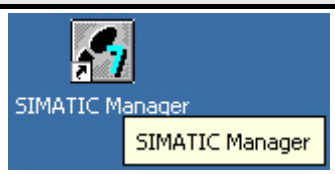
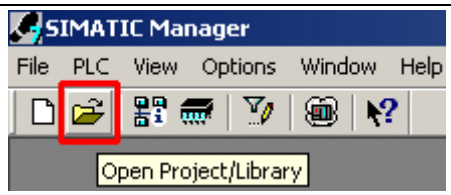
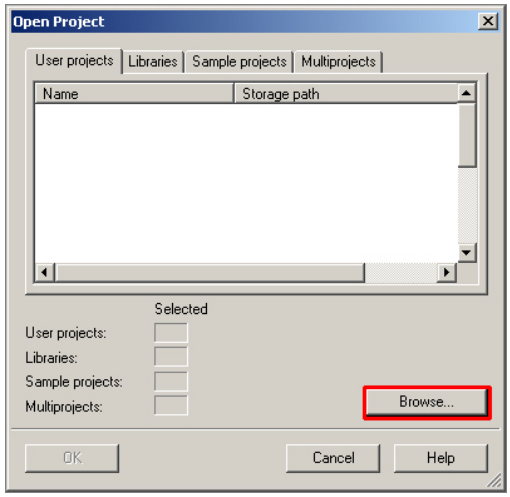
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Double-click the Hardware configuration in the STEP 7-Projekt “CE-X18A_Client_v1d2”. |  |
| 2. | Double-click the CPU 315-2PN/DP to open the Properties. |  |
| 3. | • Select the “Time-of-Day Interrupts” tab and activate OB10.<br>• Select execution “Every day”.<br>• Define the time of day at which the server shall be synchronized every day.<br>• Click “OK” to confirm your settings. |  |

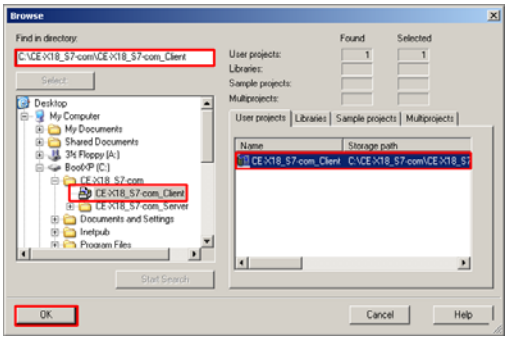
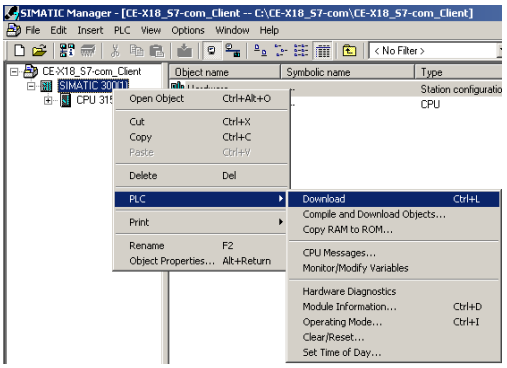
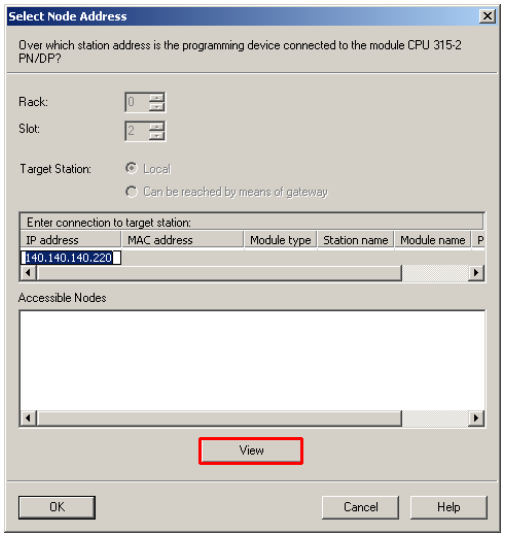
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 4. | • Click the “Save and Compile” button.<br>• Finally, the modified hardware configuration must be loaded into the CPU. |  |

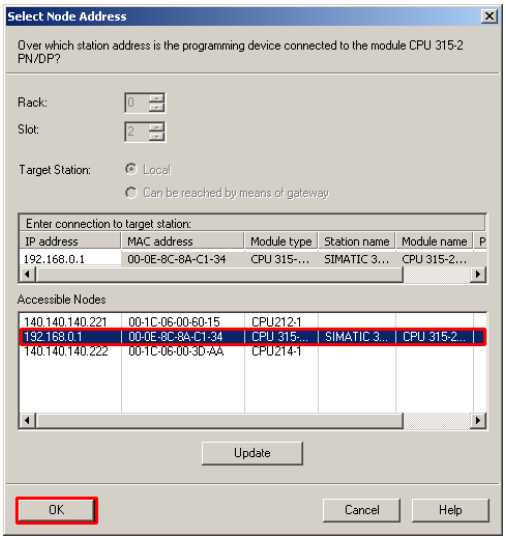
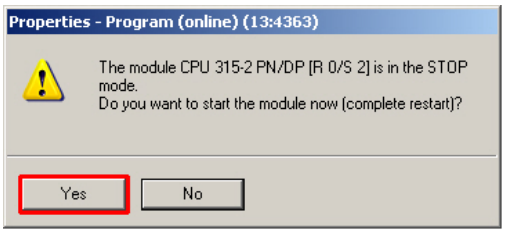
### Loading the client project into the controller

Unzip the example program “CE-X18A_Client_v1d2.zip” into any desired directory on your harddisk.
The unzipped file includes the project “CE-X18A_Client_v1d2” for the CPU 315-2PN/DP.

Table 3-7

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Open the “SIMATIC Manager”. |  |
| 2. | • Click the “Open Project/Library” button. |  |
| 3. | • Click the “Browse” button. |  |

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 4. | • Navigate to the folder where you have unzipped the example project and select the S7-300 project “CE-X18A_Client_v1d2”.<br>• Click “OK” to confirm opening of the project. |  |
| 5. | The SIMATIC Manager opens the S7-300 project.<br>• Select the station “SIMATIC 300(1)”.<br>• Click your right mouse button to open the context menu and select “PLC” -> “Download” to download the whole project. |  |
| 6. | A dialog window opens where you can select the node address.<br>• Click the “View” button to show a list of all accessible nodes. |  |

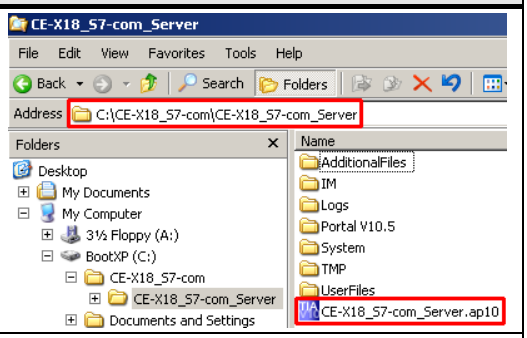
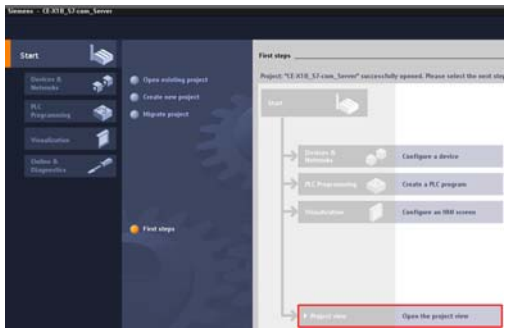
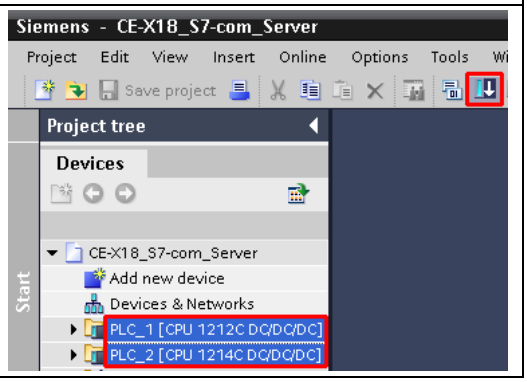
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 7. | • Select the S7-300 from the list of accessible nodes (identified via the MAC address).<br>• Click “OK” to confirm your settings. |  |
| 8. | • After successful project download, set the CPU to the operating mode “RUN”. |  |

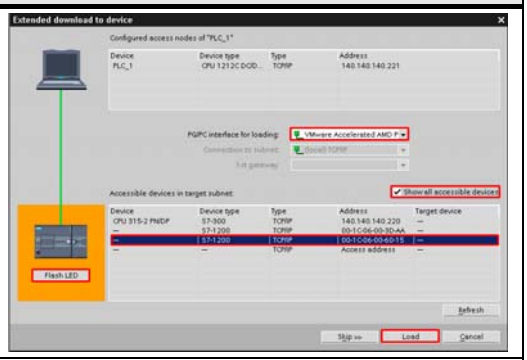
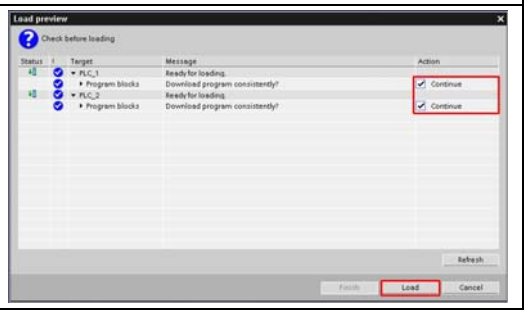
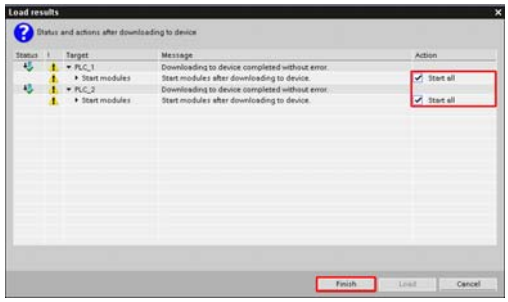
### 3.2.3 Configuration of the S7-1200

**Loading the server project into the controllers**

Unzip the example program “CE-X18A_Server_v1d2.zip” into any desired directory on your harddisk.
The unzipped file includes the project “CE-X18A_Server_v1d2” for the two S7-1200 controllers.

Table 3-8

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Open the Windows Explorer, navigate to the S7-200 project “CE-X18A_Server_v1d2.ap10” and open the project with a double-click. |  |
| 2. | The project opens in STEP 7 Basic.<br>• Open the project view. |  |
| 3. | • Select the two controller folders “PLC_1 [CPU 1212C DC/DC/DC]” and “PLC_2 [CPU 1214C DC/DC/DC]”.<br>• Click the “Download to Device” button to load the projects completely into the controllers. |  |

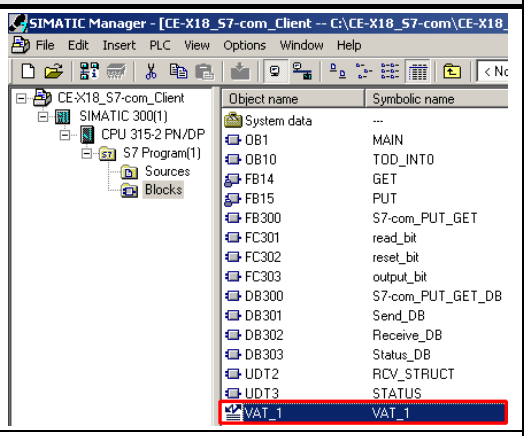
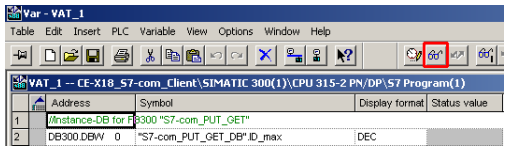
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 4. | • Select the network card to be used.<br>• Activate the relevant option to show all accessible devices.<br>• Identify the PLC_1 controller from the list of accessible devices via the MAC address or “Flash LED”.<br>• Select the desired controller and click the “Load” button.<br>Repeat the last two steps to download PLC_2. |  |
| 5. | • Activate the continuous loading action option for both controllers.<br>• Click the “Load” button. |  |
| 6. | When the transfer of all program blocks to the controllers is completed, a window with the “Load results” appears.<br>• Select the option fields “Start all” to set both controllers into the operating mode “Run”.<br>• Click the “Finish” button to complete the download operation. |  |

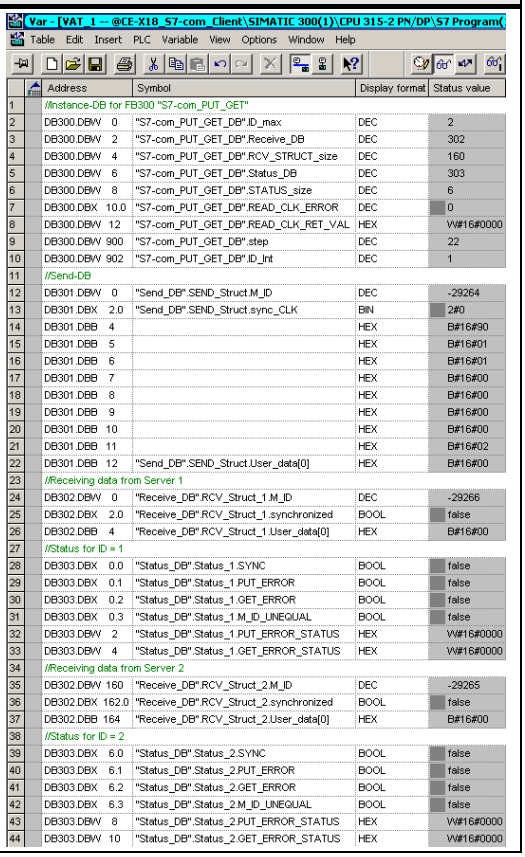
## 3.3 Online mode activation

For communication control and monitoring, your PG/PC must be switched to online mode for the S7-1200 and the S7-300 via the monitoring/variables table.

**Activating the monitoring table for the S7-300 client**

Table 3-9

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Open the SIMATIC Manager and select the project name -> station name -> CPU -> “Blocks” container and then the variables table “VAT_1”. |  |
| 2. | Activate the variables table by clicking the “Watch variable” button. |  |

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 3. | The variables table includes (row numbers in brackets):<br>• Instance data block of FB300 (2-10)<br>- specified initial values (2-6)<br>- return value of the “read system clock” function (7-8)<br>- step indication (9)<br>- ID for connection (10)<br>• Send data block (12-22)<br>- message ID (12)<br>- synchronization request (13)<br>- master system time in the format DATE_AND_TIME (14-21)<br>- first byte of user data (22)<br>• Receive data of server 1 (24-26)<br>- message ID (24)<br>- synchronization acknowledgement (25)<br>- first byte of user data (26)<br>• Status information of server 1 (28-33)<br>- synchronization request (28)<br>- communication error analysis (29-33)<br>• Receive data of server 2 (35-37)<br>• Status information of server 2 (39-44) |  |

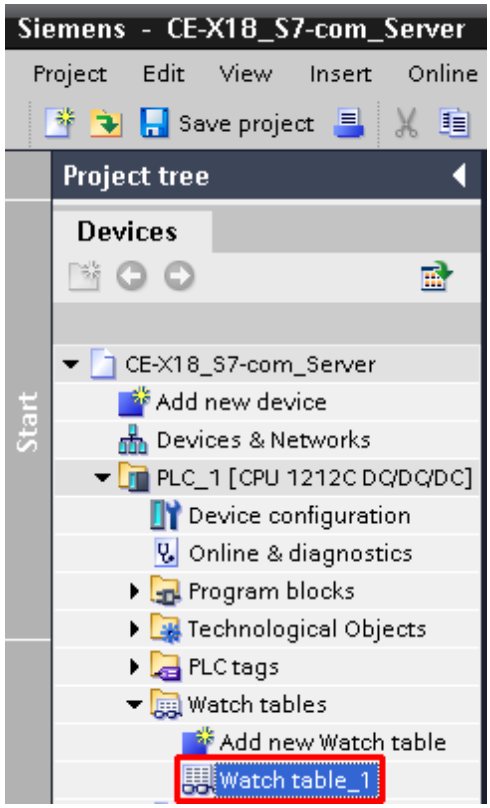
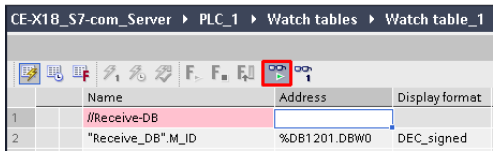
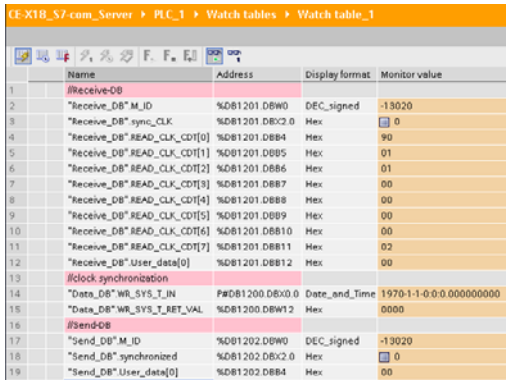
Var - [VAT_1 -- @CE-X18_S7-com_Client\SIMATIC 300(1)\CPU 315-2 PN/DP\S7 Program(

Table Edit Insert PLC Variable View Options Window Help

| | Address | Symbol | Display format | Status value |
|---|---|---|---|---|
| 1 | | //Instance-DB for FB300 "S7-com_PUT_GET" | | |
| 2 | DB300.DBW 0 | "S7-com_PUT_GET_DB".ID_max | DEC | 2 |
| 3 | DB300.DBW 2 | "S7-com_PUT_GET_DB".Receive_DB | DEC | 302 |
| 4 | DB300.DBW 4 | "S7-com_PUT_GET_DB".RCV_STRUCT_size | DEC | 160 |
| 5 | DB300.DBW 6 | "S7-com_PUT_GET_DB".Status_DB | DEC | 303 |
| 6 | DB300.DBW 8 | "S7-com_PUT_GET_DB".STATUS_size | DEC | 6 |
| 7 | DB300.DBX 10.0 | "S7-com_PUT_GET_DB".READ_CLK_ERROR | DEC | 0 |
| 8 | DB300.DBW 12 | "S7-com_PUT_GET_DB".READ_CLK_RET_VAL | HEX | W#16#0000 |
| 9 | DB300.DBW 900 | "S7-com_PUT_GET_DB".step | DEC | 22 |
| 10 | DB300.DBW 902 | "S7-com_PUT_GET_DB".ID_Int | DEC | 1 |
| 11 | | //Send-DB | | |
| 12 | DB301.DBW 0 | "Send_DB".SEND_Struct.M_ID | DEC | -29264 |
| 13 | DB301.DBX 2.0 | "Send_DB".SEND_Struct.sync_CLK | BIN | 2#0 |
| 14 | DB301.DBB 4 | | HEX | B#16#90 |
| 15 | DB301.DBB 5 | | HEX | B#16#01 |
| 16 | DB301.DBB 6 | | HEX | B#16#01 |
| 17 | DB301.DBB 7 | | HEX | B#16#00 |
| 18 | DB301.DBB 8 | | HEX | B#16#00 |
| 19 | DB301.DBB 9 | | HEX | B#16#00 |
| 20 | DB301.DBB 10 | | HEX | B#16#00 |
| 21 | DB301.DBB 11 | | HEX | B#16#02 |
| 22 | DB301.DBB 12 | "Send_DB".SEND_Struct.User_data[0] | HEX | B#16#00 |
| 23 | | //Receiving data from Server 1 | | |
| 24 | DB302.DBW 0 | "Receive_DB".RCV_Struct_1.M_ID | DEC | -29266 |
| 25 | DB302.DBX 2.0 | "Receive_DB".RCV_Struct_1.synchronized | BOOL | false |
| 26 | DB302.DBB 4 | "Receive_DB".RCV_Struct_1.User_data[0] | HEX | B#16#00 |
| 27 | | //Status for ID = 1 | | |
| 28 | DB303.DBX 0.0 | "Status_DB".Status_1.SYNC | BOOL | false |
| 29 | DB303.DBX 0.1 | "Status_DB".Status_1.PUT_ERROR | BOOL | false |
| 30 | DB303.DBX 0.2 | "Status_DB".Status_1.GET_ERROR | BOOL | false |
| 31 | DB303.DBX 0.3 | "Status_DB".Status_1.M_ID_UNEQUAL | BOOL | false |
| 32 | DB303.DBW 2 | "Status_DB".Status_1.PUT_ERROR_STATUS | HEX | W#16#0000 |
| 33 | DB303.DBW 4 | "Status_DB".Status_1.GET_ERROR_STATUS | HEX | W#16#0000 |
| 34 | | //Receiving data from Server 2 | | |
| 35 | DB302.DBW 160 | "Receive_DB".RCV_Struct_2.M_ID | DEC | -29265 |
| 36 | DB302.DBX 162.0 | "Receive_DB".RCV_Struct_2.synchronized | BOOL | false |
| 37 | DB302.DBB 164 | "Receive_DB".RCV_Struct_2.User_data[0] | HEX | B#16#00 |
| 38 | | //Status for ID = 2 | | |
| 39 | DB303.DBX 6.0 | "Status_DB".Status_2.SYNC | BOOL | false |
| 40 | DB303.DBX 6.1 | "Status_DB".Status_2.PUT_ERROR | BOOL | false |
| 41 | DB303.DBX 6.2 | "Status_DB".Status_2.GET_ERROR | BOOL | false |
| 42 | DB303.DBX 6.3 | "Status_DB".Status_2.M_ID_UNEQUAL | BOOL | false |
| 43 | DB303.DBW 8 | "Status_DB".Status_2.PUT_ERROR_STATUS | HEX | W#16#0000 |
| 44 | DB303.DBW 10 | "Status_DB".Status_2.GET_ERROR_STATUS | HEX | W#16#0000 |

### Activating the monitoring tables for the S7-1200 servers

Table 3-10

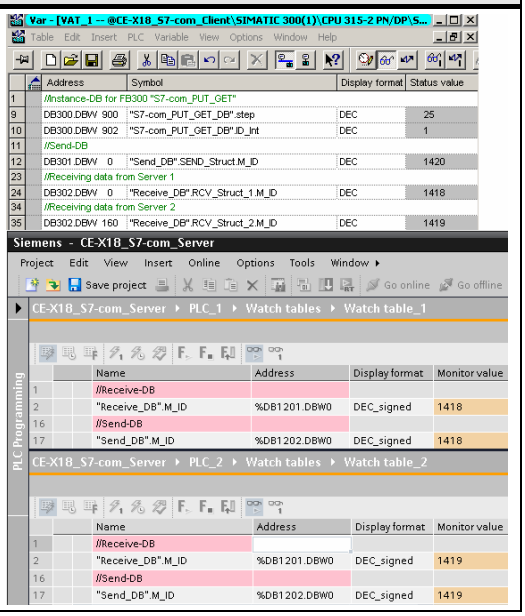
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Go to the STEP 7 Basic project tree and select “PLC_1” -> “Watch tables” -> “Watch table_1”. |  |
| 2. | • Activate the monitoring table by clicking the “Watch all” button. |  |
| 3. | Repeat steps 1 and 2 for server 2:<br>• PLC_2 [CPU 1214C DC/DC/DC]<br>• Watch table_2 | |
| 4. | Each monitoring table includes the following information (row numbers in brackets):<br>• Receive data block(2-12)<br>- message ID (2)<br>- synchronization request (3)<br>- master system time in the format DATE_AND_TIME (4-11)<br>- first byte of user data (12)<br>• Variables data block (14-15)<br>- converted master system time in DTL format (14)<br>- return value of the “write system time” function (15)<br>• Send data block (17-19)<br>- mirrored message ID (17)<br>- synchronization acknowledgement (18) |  |

| No. | Instruction | Note/Screenshot |
|---|---|---|
| | - first byte of user data (19) | |

# 3.4 Live demonstration

## 3.4.1 Cyclic operation

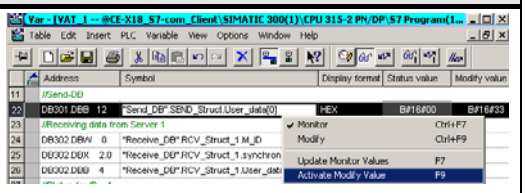
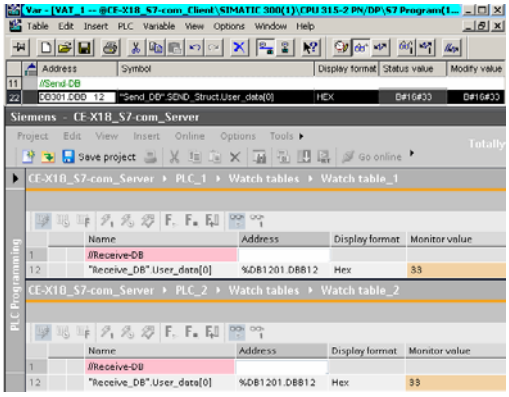
Table 3-11

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • The FB300 “S7-com_PUT_GET” of the client is called at cyclic intervals (which can be seen by the step changes in row 9)<br>• It communicates continuously with the servers 1 and 2 (indicated by the ID change in row 10).<br>• Message IDs with even numbers are transmitted to server 1 where they are mirrored and then received again.<br>• Message IDs with uneven numbers are transmitted to server 2 where they are mirrored and then received again. |  |

## 3.4.2 Transmission of user data

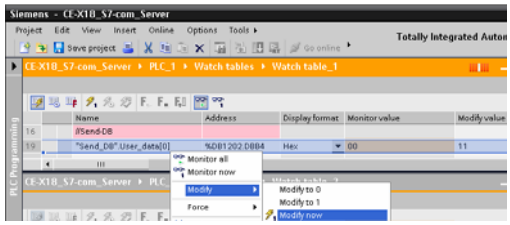
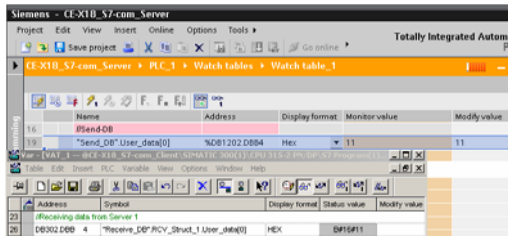
### S7-300 client -> S7-1200 server

Table 3-12

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | In this example for the transmission of user data from the client to the servers, the send byte 0 of the user data field in row 22 shall be changed:<br>• Enter a value in the “Status value” column in row 22.<br>• Click F9 to confirm activation of the modified value. |  |
| 2. | The value will be transmitted to both servers and written into the receive byte 0 of the user data field in the receive data block 1201 (as shown in row 12 of the server watch tables). |  |

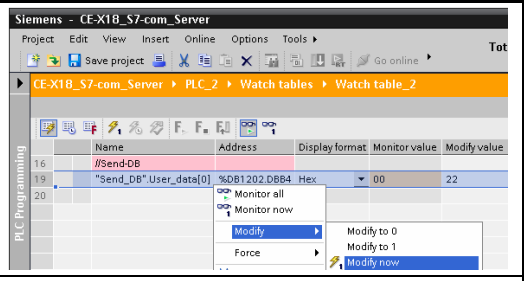
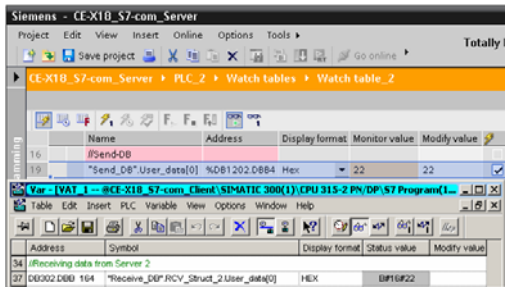
### S7-1200 server 1 -> S7-300 client

Table 3-13

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | In this example for the transmission of user data from server 1 to the client, the send byte 0 of the user data field in row 19 shall be changed:<br>• Open “Watch table_1” and enter a value in the “Monitor value” column in row 19<br>• Click your right mouse button to confirm the changed value by selecting “Modify” -> “Modify now”. |  |
| 2. | The value will be transmitted to the client and written into the receive byte 0 of the user data field in the receive structure for server 1 (as shown in row 26 of the client variables table). |  |

### S7-1200 server 2 -> S7-300 client

Table 3-14

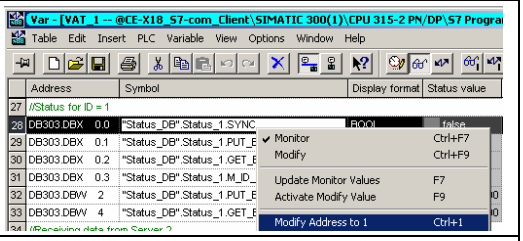
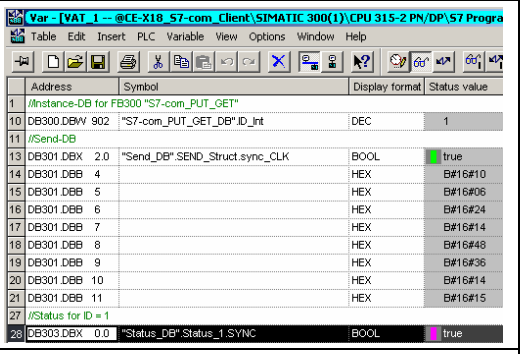
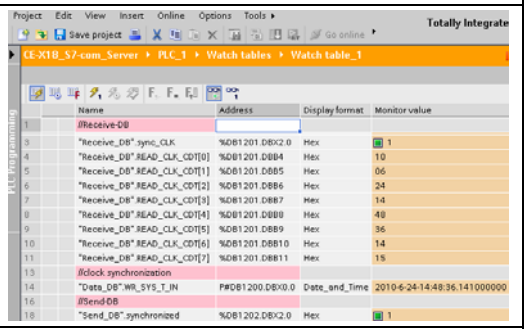
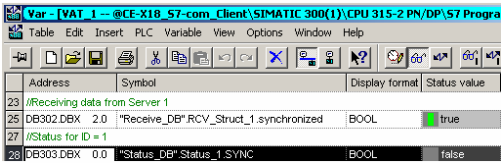
| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | In this example for the transmission of user data from server 2 to the client, the send byte 0 of the user data field in row 19 shall be changed:<br>• Open “Watch table_2” and enter a value in the “Monitor value” column in row 19<br>• Click your right mouse button to confirm the changed value by selecting “Modify” -> “Modify now”. |  |
| 2. | The value will be transmitted to the client and written into the receive byte 0 of the user data field in the receive structure for server 2 (as shown in row 37 of the client variables table). |  |

### 3.4.3 Time synchronization

#### Manual server synchronization

Server 1 shall be synchronized manually with the client's system time. How to proceed is described in Table 3-15.
The same procedure can also be used for the synchronization of sever 2.

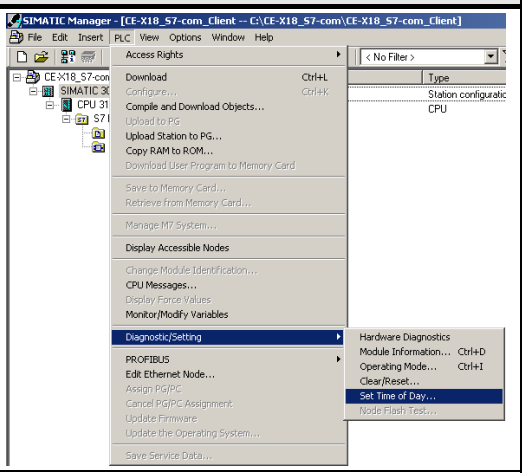
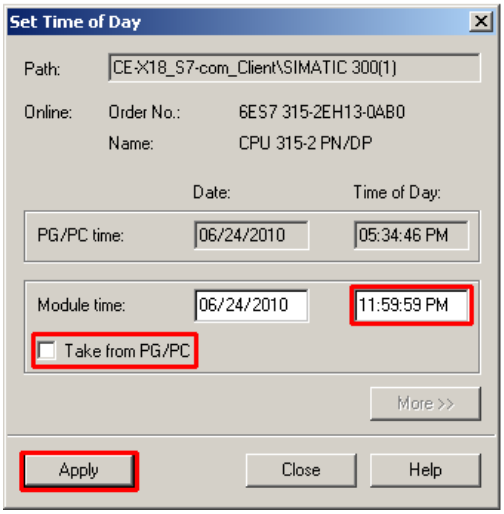
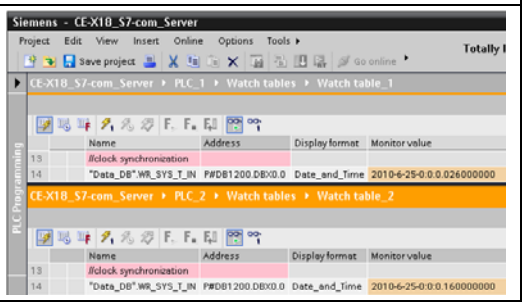
Table 3-15

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | • Select the synchronization request in the status structure for server 1 by clicking your right mouse button and then -> "Modify Address to 1" (row 28 in the client's variables table). |  |
| 2. | • The system time is written to the send data in a DATE_AND_TIME format (rows 14 – 21)<br>• The synchronization request is set in the send data (row 13).<br>• The send data is transmitted to server 1 including connection ID 1 (row 10). |  |
| 3. | • The time synchronization data is written to the receive block of server 1 ("Watch table_1", rows 3 – 11)<br>• The converted synchronization time in a DTL data type is written to the system time of the S7-1200 (row 14).<br>• After successful time synchronization, synchronization will be acknowledged (row 18). |  |
| 4. | • On the client side, the synchronization acknowledgement is written into the receive structure of server 1 (row 25)<br>• The synchronization request in the status structure for server 1 is reset (row 28). |  |

### Automatic synchronization of all servers

In chapter 3.2.2, the daily synchronization time of all servers has been set to 00:00. In order to check proper function, the system time of the client is set to 23:59 pm.

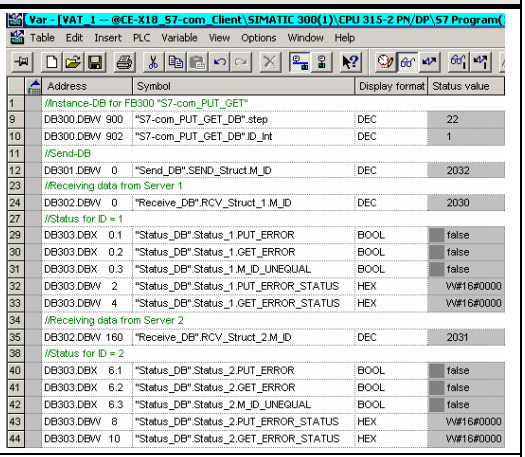
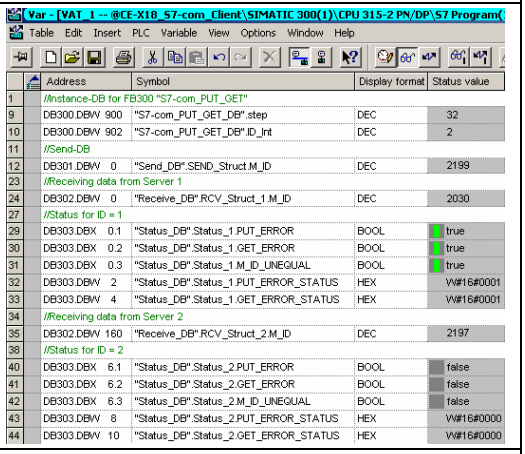
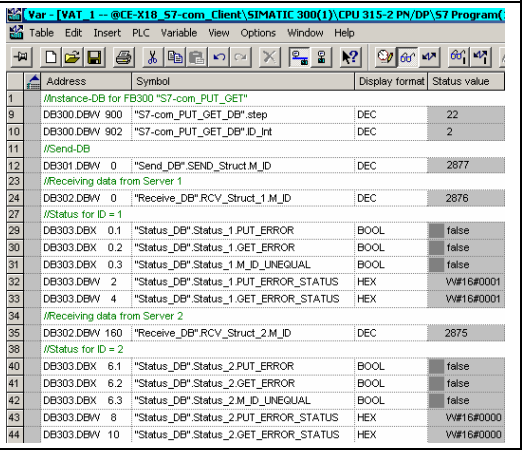
Table 3-16

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Open Step 7 V5.4 and select “PLC” -> “Diagnostic/Setting” -> “Set Time of Day…”. |  |
| 2. | • Deactivate the option “Take from PG/PC”.<br>• Set the module time to “11:59:59 PM”.<br>• Click “Apply” to confirm your settings. |  |
| 3. | Successful clock synchronization of the servers can be checked by means of the stated system time of the servers (row 14 in the monitoring tables “Watch table_1” and “Watch table_2”). |  |

### 3.4.4 Communication errors

Pull the Ethernet cable from server 1 to demonstrate the communication error analysis function.
How to proceed is described in Table 3-15.
The same procedure can be used to simulate and analyze an interruption in the communication with server 2.

Table 3-17

| No. | Instruction | Note/Screenshot |
|---|---|---|
| 1. | Pull the Ethernet cable from the LAN port of server 1.<br>• The step sequence stops (row 9) during communication with server 1 (row 10), since there is no confirmation or error message from the S7 communication blocks PUT or GET. |  |
| 2. | • After the internal timeout period of the communication blocks has elapsed, an error message is issued and the step sequence continues.<br>• In the status structure for server 1, the communication errors of the blocks PUT (row 29) and GET (row 30) are issued including the relevant error states (rows 32 and 33).<br>• Furthermore, the discrepancy between the transmitted message ID (row 12) and the last ID received from server 1 (row 24) is identified and indicated in row 31. |  |
| 3. | Reconnect the Ethernet cable with the LAN port of server 1.<br>• After recovery of the connection has been identified, the error bits in the status structure for server 1 will be reset (rows 29 – 31).<br>• Data exchange with server 1 has been restored.<br>• The error states of the communication blocks PUT (row 32) and GET (row 33) include the status information of the last errors. |  |

### 3.4.5 Power failure of the client

After power recovery of the client, the step sequence of function block 300 “S7-com_PUT_GET” starts from the last position of execution.

# 4 Code Elements

In the example described in this document uses the following program codes:

Table 4-1

| No. | File name | Contents |
|---|---|---|
| 1. | CE-X18A_Client_v1d2.zip<br>• CE-X18A_Client_v1d2 | Zip file including the S7-300 client project for deterministic S7 communication via the PUT and GET blocks |
| 2. | CE-X18A_Server_v1d2.zip<br>• CE-X18A_Server_v1d2.ap10 | Zip file including the S7-1200 server project for deterministic S7 communication via the PUT and GET blocks |

# 5 History

Table 5-1

| Version | Date | Revisions |
|---|---|---|
| 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 S7 communication (task A) |
| | | |