

Industry Online Support

-

NEWS

S7 Communication between SIMATIC S7-1200 and SIMATIC S7-300

STEP 7 V16 / STEP 7 V5.6 / PUT / GET

https://support.industry.siemens.com/cs/ww/en/view/82212115

Siemens Industry Online Support



This entry is from the Siemens Industry Online Support. The general terms of use (<u>http://www.siemens.com/terms_of_use</u>) apply.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks. In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept.

Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.

Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

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

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

Table of content

1	Introdu	ction	5
2	Configu	Iration of SIMATIC S7-1200	6
	2.1 2.1.1 2.1.2 2.1.3	Create the Hardware Configuration Set the IP address of the CPU Set the IP address of the CP Permit Access with PUT/GET Communication from Remote Partner.	6 6 7
	2.1.4 2.2 2.2.1 2.2.2 2.2.3	Enable S7 communication to the CPU Install SIMATIC S7-1200 as Client Configuration of the S7 Connection Create Send and Receive Data Areas Create Variables for Parameters of Function Blocks "PutData"	8 9 9 13
	2.2.4 2.3 2.3.1 2.4	and "GetData" Create a User Program Install SIMATIC S7-1200 as Server Create Send and Receive Data Areas Download the Hardware Configuration and User Program	17 20 27 27 33
3	Configu	Iration of SIMATIC S7-300 in STEP 7 V16	37
	3.1 3.1.1 3.2 3.2 3.2.1 3.2.2 3.2.3	Create the Hardware Configuration Set the IP address of the CPU Set the IP address of the CP Install SIMATIC S7-300 as Client Configuration of the S7 Connection Create Send and Receive Data Areas Create Variables for Parameters of Function Blocks "PutData"	37 37 38 39 39 43
	3.2.4 3.3 3.3.1 3.4	and "GetData" Create a User Program Install SIMATIC S7-300 as Server Create Send and Receive Data Areas Download the Hardware Configuration and User Program	47 50 57 57 61
4	Configu	iration of SIMATIC S7-300 in STEP 7 V5.6	65
	4.1 4.1.1 4.1.2	Create the Hardware Configuration Open the Hardware Configuration Define IP Address and Assign Subnet to the PROFINET interface of the CPU	65 65 67
	4.1.3 4.2 4.2.1 4.2.2	Define IP Address and Assign Subnet to the PROFINET interface of the CP Configure the SIMATC S7-300 as Client Configuration of the S7 Connection Create Send and Receive Data Areas	69 71 71 76
	4.2.3 4.2.4 4.3 4.3.1 4.4	Create Variables for Parameters of Function Blocks "PutData" and "GetData" Create a User Program Configure the SIMATC S7-300 as Server Create a User Program Download the Hardware Configuration and User Program	79 82 90 90 94
5	Error Ha	andling	96
	5.1 5.2	FB "PutData" FB "GetData"	96 96
6	Operati	ng and Monitoring	97
	6.1	SIMATIC S7-1200 Writes and Reads Data as Client	97

6.1.1	Write Data to the S7-300	97
6.1.2	Read Data from the S7-300	99
6.2	SIMATIC S7-300 Writes and Reads Data as Client (STEP 7	
	V16)	101
6.2.1	Write Data to the S7-1200	101
6.2.2	Read Data from the S7-1200	103
6.3	SIMATIC S7-300 Writes and Reads Data as Client (STEP 7	
	V5.6)	105
6.3.1	Write Data to the S7-1200 CPU	105
6.3.2	Read Data from the S7-1200 CPU	107

1 Introduction

This document shows you how to exchange data between an S7-1200 and an S7-300 using S7 Communication.

The SIMATIC S7-1200 is configured in STEP 7 V16 (TIA Portal V16).

The SIMATIC S7-300 can be configured in STEP 7 V16 (TIA Portal V16) or in STEP 7 V5.6.

Two S7 connections are established.

For the first S7 connection the S7-300 acts as a client and the S7-1200 as server. This means that the S7-300 actively establishes the connection. The function blocks "GET" and "PUT" are called in the user program of the S7-300 to read data from the S7-1200 and write data to the S7-1200.

For the second S7 connection the S7-1200 acts as a client and the S7-300 as server. This means that the S7-1200 actively establishes the connection. The instructions "GET" and "PUT" are called in the user program of the S7-1200 to read data from the S7-300 and write data to the S7-300.



2 Configuration of SIMATIC S7-1200

- Create a new project or open an existing project in STEP 7 V16 (TIA Portal V16).
- 2. Add the S7-1200 CPU and create the hardware configuration.
- 3. Install the S7-1200 CPU as client or server.
 - Client: Configure a new S7 connection and create the user program. In the user program you call the "PUT" instruction to write data to the S7-300 CPU. In the user program you call the "GET" instruction to read data from the S7-300 CPU. Add the data blocks for saving the Send and Receive data.
 - Server: Create the user program. In the user program you add data blocks in which the sent and received data is stored.

2.1 Create the Hardware Configuration

2.1.1 Set the IP address of the CPU

1. In the Device view or Network view of the "Devices & networks" editor you mark the S7-1200 CPU.

The properties of the S7-1200 CPU are displayed in the inspector window.

- 2. In the "General" tab under "PROFINET interface [X1] > Ethernet addresses" you enter the IP address and the subnet mask of the S7-1200 CPU.
 - IP address: 172.16.43.2
 - Subnet mask: 255.255.0.0
- 3. Assign a subnet to the Ethernet interface of the S7-1200 CPU. You have two options:
 - Click the "Add new subnet" button to add a new subnet.
 - Select an existing subnet.



2.1.2 Set the IP address of the CP

- In the Device view or Network view of the "Devices & networks" editor you mark the CP in the S7-1200.
 The properties of the CP are displayed in the inspector window.
 - The properties of the CP are displayed in the inspector window.
- 2. In the "General" tab under "Ethernet interface [X1] > Ethernet addresses" you enter the IP address and the subnet mask of the CP.
 - IP address: 172.16.43.40
 - Subnet mask: 255.255.0.0
- 3. Assign a subnet to the Ethernet interface of the CP. You have two options:
 - Click the "Add new subnet" button to add a new subnet.
 - Select an existing subnet.

CP 1243-1 [CP 1243-1]			🔍 Proper	ties 🛄 li	nfo 追 🖪 Diagno
General IO tags Sys	tem constants	Texts			
General General	Ethernet address	es			
Ethernet interface [X1]	Interface netv	vorked wit	h		
General Ethernet addresses		Sub	onet: PN/IE 1		
Time-of-day synchronization			Add new	subnet	
 Advanced options 					
Web server access	IP protocol	\sim			
DNS configuration	ii protocor				
 Communication with the CPU 			Set IP addres	ss in the proje	t
Watchdog bit		2~~			
CP time			IP add	dress: 172 .	16 . 43 . 40
CP diagnostics			Subnet r	mask: 255 .	255.0.0
SNMP			Use router		

2.1.3 Permit Access with PUT/GET Communication from Remote Partner

In the S7-1200 CPU you enable the function "Permit access with PUT/GET communication from remote partner". This enables the partner CPU to access the data in the S7-1200 CPU using the "PUT" and "GET" instructions.

1. In the Device view or Network view of the "Devices & networks" editor you mark the S7-1200 CPU.

The properties of the S7-1200 CPU are displayed in the inspector window.

 In the "General" tab, under "Protection & Security > Connection mechanisms" you enable the "Permit access with PUT/GET communication from remote partner" function.



2.1.4 Enable S7 communication to the CPU

With the S7-1200, enable S7 communication to the CPU in the CP. In order to release the function of S7 communication with the assigned CPU and the S7 routing.

If you configure an S7 connection that runs via the CP of the S7-1200, then you have to activate this option.

- 1. In the Device view or Network view of the "Devices & networks" editor you mark the CP in the S7-1200.
- 2. In the "General" tab, under "Communication types" you enable the "Enable S7 communication to the CPU" option.

CP 1243-1 [CP 1243-1]		🔍 Propert	ties	i Info	1 Diagnostics
General IO tags Sys	tem constants	Texts			
General	Communication	tunos			
Communication types	Communication types				
Ethernet interface [X1]					
DNS configuration	Enable telecontrol communication			control communication	
 Communication with the CPU 	Protocol type: TeleControl Basic		sic		
Watchdog bit					
CP time	I Enable online functions			ne functions	
CP diagnostics				ommunication to the CPU	
SNMP					

2.2 Install SIMATIC S7-1200 as Client

2.2.1 Configuration of the S7 Connection

- 1. In the Network view you click the "Connections" icon to enable the Connection mode.
- 2. Select "S7 connection" as the connection type in the adjacent drop-down list box.

In the Network view, all devices that are available for an S7 connection are highlighted in color.

- 3. Right-click the S7-1200 CPU. The pop-up menu opens.
- 4. Select the "Add new connection" item. The "Add new connection" dialog opens.

Network		t.
PLC_1 CPU 1214C		
	Y Device configuration Change device	
PN/IE_1	🗶 Cut	Ctrl+X
	E Copy	Ctrl+C
	📋 Paste	Ctrl+V
	X Delete	Del
	Rename	F2
	Assign to new DP master / IO controller	
	Disconnect from DP master system / IO system	
	Highlight DP master system / IO system	
	Go to topology view	
	Add new connection	
	Highlight connection partr	•
	Compile	•

- 5. Specify the following connection partner: "Unspecified".
- Select the local interface. In this example, the integrated interface of the S7-1200 CPU is used. If the S7 connection is to be established via a CP, select the CP.
- 7. Enter the local ID for the S7 connection, 100 (hex), for example.
- 8. Click the "Add" button to add the unspecified S7 connection.
- 9. Click the "Close" button to end the dialog.

Add new connection	×
Please select connection partner for PLC_1:	Type: 57 connection
Local ID (hex): 100	♥ Establish active connection ♥ One-way
	8 Add Close

Result

- An unspecified S7 connection is created.
- The connection path is displayed highlighted in the graphical area of the Network view.

Network Connections S7 connect	on 🔽 👯 🖫 🗐 🖽 🛄 🔍 ±
	4 Highlighted: Connection
PLC_1 CPU 1214C	
S7 connection one-way	

• The S7 connection is entered in the "Connections" table in the table area of the Network view.

Network overview	Connections	I/O commun	ication	VPN	Tele	Control			
Y Local connection name	Local end point		Local ID (hex) Partne	er ID (hex)	Partner		Connection type	
S7 connection one-way	PLC_1 [CPU 1]	214C DC/DC/DC]	100	•		🔢 Unknowr	י 🗖	S7 connection	

Setting the connection parameters

- 1. In the "Connections" table, mark the unspecified S7 connections. The Properties of the S7 connection are displayed in the inspector window.
- 2. Under "General" the "General" tab displays the general connection parameters that identify the local endpoint. Under Partner you enter the IP address of the S7-300 CPU, for example: 172.16.43.37. If you use a CP as partner instead of the integrated interface of the S7-300 CPU, enter the IP address of the CP.

\$7 connection one-way [\$	7 connection]	Risperties	🗓 Info 追 📱 Di	iagnostics
General IO tags	System constants Texts			
General Local ID	General			
Special connection propertie	s Connection			
Address details	Name: [S7 connection one-way		
	Connection path			
	_	Local		Partner
	-	-	-	?
	End point:	PLC_1 [CPU 1214C DC/D0	[/DC]	Unknown
	Interface:	PLC_1, PROFINET interfac	e_1[X1 : PN(LAN 🔻	Unknown
	Interface type:	Ethernet		Ethernet
	Subnet:	PN/IE_1	I	
	Address:	172.16.43.2	0	172.16.43.37

 In the "General" tab under "Local ID" you see the local ID of the module from which the connection is viewed (local partner). You can change the local ID. You specify the local ID later at the "ID" input parameter of the "PUT" or "GET" instruction.

In this example we use the local ID with the value 100 (hex).

57 connection one-way [57 connection]							
General	IO tags	System constants	Texts				
General		Local ID					
Special conn	ection properti	es Block param	eters				
Address deta	ils						
			Local ID (hex):	100			
			ID:	W#16#100			

4. In the "General" under "Special connection properties" you see that the S7 connection is configured unilaterally. Unilaterally means that the connection partner is server for this connection and cannot actively send or receive. The S7 CPU that is used as the local endpoint for the S7 connection, actively establishes the S7 connection as a client and actively sends and receives.

\$7 connection one-way [\$7 co	nnection]
General IO tags Sys	tem constants Texts
General Local ID	Special connection properties
Special connection properties Address details	Local end point
	☑ One-way
	Active connection establishment
	The active connection establishment cannot be deactivated if the Partner TSAP in the address details has the value 3.
	Send operating mode messages

 In the "General" tab under "Address details" you enter the partner TSAP. Since the S7 connection is configured unilaterally to an unspecified partner, the partner TSAP is composed as follows: 03 <slot of CPU or CP >, for example:

S7-300 CPU in slot 2: 03.02

S7 connection one-way [S7 co	nnection]	Sector Properties	s 🗓 Info 追 🗓 Diag
General IO tags Sys	tem constants Texts		
General Local ID	Address details		
Special connection properties			
Address details	Local		Partner
	End point: PLC_1 [CPU 12	14C DC/DC/DC]	Unknown
	Rack/slot: 0	1	0 2
	Connection res. (hex):		03
	TSAP: SIMATIC-ACC10	0001	03.02
			SIMATIC-ACC
	Subnet ID: TA43-0001		•

2.2.2 Create Send and Receive Data Areas

Add data block for saving the Receive data

The data received from the S7-300 CPU is stored in data block DB1 "RecvDataClient" of the S7-1200 CPU.

- In the project tree you navigate to the device folder of the S7-1200 CPU. The device folder contains structured objects and actions that belong to the device.
- In the device folder you navigate to the "Program blocks" subfolder and double-click the "Add new block" action. The "Add new block" dialog opens.



- 3. Click the "Data block (DB)" button.
- 4. Enter the name of the data block (DB), "RecvDataClient", for example.
- 5. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 6. Enable the "Add new and open" function.
- 7. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "RecvDataClient" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "RecvDataClient" DB opens in the working area of STEP 7 V16.

Siemens AG 2020 All rights reserved

Add new block					
Haa new block					^
Name:					
RecvDataClient	4				
	T				
	pe:	🥃 Global DB			
OB	Language:	DB	-		
Organization block	Number:	1	×		
		🔘 Manual			
		 Automatic 	6		
FB	Description:				
Function block	Data blocks (DB	s) save program data.			
FC					
Function					
	a				
Data block					
	more				
> Additional information	ation				
					Const
Add new and open		9		OK	

8. Alternatively, you can open the "RecvDataClient" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-1200 CPU and double-clicking the "RecvDataClient" DB.



9. In the DB "RecvDataClient" you define the variable "data" of the data type Array[0..199] of Byte.

RecvDataClient						
Name	Data type	Offset	Start value			
🕣 🔻 Static						
🕣 = 🕨 data	Array[0199] of Byte	0.0				
Add new>						

10. In the Properties of the DB1 "RecvDataClient", under "Attributes" you disable the "Optimized block access" function.

RecvDataClient [DB1]		×
General Text	ts	
General Information Time stamps Compilation Protection Attributes Download without re	einitializati	
	OK Cancel	
	Calicer	

Add data block for saving the Send data

The data that is transferred to the S7-300 CPU is stored in data block DB2 "SendDataClient" of the S7-1200 CPU.

- 1. Add another DB for saving the Send data.
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "SendDataClient", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "SendDataClient" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "SendDataClient" DB opens in the working area of STEP 7 V16.

Add new block			x
Name:			
SendDataClient	3		
		🧧 Global DB 📃 🔻	
	Language:	DB	
-0B			
block	Number:	2	
		O Manual	
		Automatic	
	Description		
FB	Description:		
Function block	Data blocks (DB	Bs) save program data.	
FC			
Eunction			
	2		
Data block			
	more		
> Additional info	rmation		
Add new and one		OKN	Cancel

 Alternatively, you can open the "SendDataClient" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-1200 CPU and double-clicking the "SendDataClient" DB.



8. In the DB "SendDataClient" you define the variable "data" of the data type "Array[0..199] of Byte".

S	endDataClient							
		Na	me		Data type	Offset	Start value	
	01	•	St	atic				
	01	•	٠	data	Array[0199] of Byte	0.0		
				<add new=""></add>	E			

9. In the Properties of DB1 "SendDataClient", under "Attributes" you disable the "Optimized block access" function.

SendDataClient [DB2]	×
General Texts	
General Information Time stamps Compilation Protection Attributes Download without reinitializati	Attributes Only store in load memory Data block write-protected in the device Optimized block access Data block accessible from Data block accessible from Data block accessible via Web Server
	OK Cancel

2.2.3 Create Variables for Parameters of Function Blocks "PutData" and "GetData"

- 1. Add another DB to create variables for the parameters of function blocks "PutData" and "GetData".
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "GeneralData", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "GeneralData" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "GeneralData" DB opens in the working area of STEP 7 V16.

Add new block			×
Name:			
GeneralData	3		
)_		
	lype:	GIODAT DB	
-OB	Language:	DB	
Organization	Number:	10	
		O Manual	
		Automatic	
	Description:		
-FB	Data blocks (DBs)	save program data	
Function block	(,		
FC			
Function			
	2		
Data block			
	more		
Additional information	ation		
🛃 Add new and open	5		OK Cancel
-			

- Alternatively, you can open the "GeneralData" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-1200 CPU and double-clicking the "GeneralData" DB.
 - PLC_1 [CPU 1214C DC/DC/DC]
 Device configuration
 Online & diagnostics
 Program blocks
 Add new block
 Main [OB1]
 GetData [FB13]
 PutData [FB12]
 GeneralData [DB10]
 InstGetData [DB13]
 InstPutData [DB12]
- 8. Create the following variables to assign the input and output parameters of the FBs "PutData" and "GetData". PLC data types are used as data type.

Variable	PLC data type	Description
put	typePut	Variables for parameterizing FB12 "PutData".
get	typeGet	Variables for parameterizing FB13 "GetData".
diagnostic	typeDiagnostic	Variables to store the status of FBs "PutData" and "GetData" in case of error.

PLC Data Type "typePut"

The following table shows the structure of "typePut" data type.

Table 2-1

Parameter	Data type	Start value	Description
execute	Boolean	false	Control parameter for FB12 "PutData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Boolean	false	Status parameters
busy	Boolean	false	Status parameters
error	Boolean	false	Status parameters
status	Word	16#0	Status parameters

PLC Data Type "typeGet"

The following table shows the structure of the PLC data type "typeGet". Table 2-2

Parameter	Data type	Start value	Description
execute	Boolean	false	Control parameter for FB13 "GetData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Boolean	false	Status parameters
busy	Boolean	false	Status parameters
error	Boolean	false	Status parameters
status	Word	16#0	Status parameters
execute connectionId done busy error status	Boolean Boolean Boolean Boolean Word	false 16#0 false false false 16#0	Control parameter for FB13 "GetData". Addressing parameter for specifying the connection to the communication partner. Status parameters Status parameters Status parameters Status parameters Status parameters

PLC Data Type "typeDiagnostic"

The following table shows the structure of the "typeDiagnostic" data type.

Гab	le	2-3	

Parameter	Data type	Start value	Description
statusPut	Word	16#0	Parameter to store the status of FB12 "PutData".
statusGet	Word	16#0	Parameter to store the status of FB13 "GetData".

2.2.4 **Create a User Program**

Overview

The following figure shows an overview of the user program in the S7-1200 CPU and S7-300 CPU.

Figure 2-1



Entry-ID: 82212115, V 3.0, 11/2020

FB12 "PutData"

The FB12 "PutData" calls the "PUT" instruction to execute the following functions:

- Send data to the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Write job is running, it is not possible to trigger a new Write job.
- Output the status of the FB and data transmission at the "status" output.

FB12 "PutData" is called cyclically in OB1.

The following figure shows the call of the FB12 "PutData" in OB1.

Figure 2-2

			_
	PutData		
Bool —	execute	done —	Bool
Word —	connectionId	busy —	Bool
		error —	Bool
		status —	
Remote	ADDR_1		Remote
Remote	ADDR_2		Remote
Remote	ADDR_3		
Remote	ADDR_4		Remote
Variant —	SD_1		
Variant —	SD_2		
Variant —	SD_3		Variant
Variant —	SD_4		Variant
vallalit			- vanant

The following table shows the parameters of FB12 "PutData".

Table 2-4

Name	P type	Data type	Comment	
execute	IN	Bool	Control parameter:	
			Enables the Write job on a rising edge.	
connectionId	IN	Word	Local ID: Addressing parameter for specifying the connection to the partner CPU. Note The local ID is available in the Properties of the configured S7 connection. The local ID 100 (hex) is used in this example.	
done	OUT	Bool	TRUE: The Write job was executed error-free.	
busy	OUT	Bool	TRUE: The Write job is running.	
error	OUT	Bool	Status parameters "error" and "status":	
status	OUT	Word	 "error" = 0: "status" = 0000 (hex): neither warning nor error "status" <> 0000 (hex): Warning The "status" parameter provides detailed information. "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error. 	
ADDR_1	IN_OUT	Remote	Pointer to the area to be written to in the partner CPU.	
ADDR_2	IN_OUT	Remote	Note	
ADDR_3	IN_OUT	Remote	I he following parameters are hidden and are preset with the value "NULL":	
ADDR_4	IN_OUT	Remote	 "ADDR_2" "ADDR_3" "ADDR_4" The following minimum user data size for the "PUT" instruction with 1 – 4 variables is guaranteed: 160 bytes 	
SD_1	IN_OUT	Variant	Pointer to the area in your own S7 CPU that contains the data to	
SD_2	IN_OUT	Variant	be sent.	
SD_3	IN_OUT	Variant	The following parameters are hidden and are preset with the	
SD_4	IN_OUT	Variant	 value "NULL": "SD_2" "SD_3" "SD_4" The following minimum user data size for the "PUT" instruction with 1 – 4 variables is guaranteed: 160 bytes 	

Assign the variables created in section 2.2.3 to the parameters of FB12 "PutData". The following table shows the assignment of the parameters of the "put" variable to the parameters of FB12 "PutData".

Parameters of FB12 "PutData"	Parameters of the "put" variable	Start value	Note
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Write job. The Write job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Write job is completed with "done" = true or "error" = true.
connectionId	connectionId	16#100	Enter the local ID of the connection that you defined during the connection configuration.
done	done	false	-
busy	busy	false	-
error	error	false	-
status	status	16#0	-
ADDR_1	-	P#DB1.DBX0.0 BYTE 100.	Only absolute addressing is permitted.
SD_1	-	P#DB2.DBX0.0 BYTE 100	-

FB13 "GetData"

The FB13 "GetData" calls the "GET" instruction to execute the following functions:

- Receive data from the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Read job is running, it is not possible to trigger a new Read job.
- Output the status of the FB and data transmission at the "status" output.

FB13 "GetData" is called cyclically in OB1.

The following figure shows the call of the FB13 "GetData" in OB1.

Figure 2-3

			_
	GetData		
Bool —	execute	done —	Bool
Word —	connectionId	busy —	Bool
		error —	Bool
		status —	
Remote	ADDR_1		Remote
Remote	ADDR_2		
Remote	ADDR_3		Remote
Remote	ADDR_4		Remote
Variant —	RD_1		
Variant —	RD_2		
Variant —			Variant
Variant	RD_4		Variant
variant			- variant

Table 2-6

Name	P type	Data type	Comment
execute	IN	Bool	Control parameter:
			Enables the Read job on a rising edge.
connectionId	IN	Word	Local ID: Addressing parameter for specifying the connection to the partner CPU.
			Note The local ID is available in the Properties of the configured S7 connection.
			The local ID 100 (hex) is used in this example.
done	OUT	Bool	TRUE: The Read job was executed error-free.
busy	OUT	Bool	TRUE: The read job is running.
error	OUT	Bool	Status parameters "error" and "status":
status	OUT	Word	• "error" = 0:
			 "status" = 0000 (hex): neither warning nor error
			 "status" <> 0000 (hex): Warning The "status" parameter provides detailed information.
			 "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error.
ADDR_1	IN_OUT	Remote	Pointer to the area to be read in the partner CPU.
ADDR_2	IN_OUT	Remote	Note
ADDR_3	IN_OUT	Remote	I he following parameters are hidden and are preset with the value "NULL":
ADDR_4	IN_OUT	Remote	• "ADDR_2"
			• "ADDR_3"
			• "ADDR_4"
			The following minimum user data size for the "GET" instruction with $1 - 4$ variables is guaranteed: 160 bytes
RD_1	IN_OUT	Variant	Pointer to the area in your own S7 CPU in which the read data
RD_2	IN_OUT	Variant	is stored.
RD_3	IN_OUT	Variant	The following parameters are hidden and are preset with the
RD_4	IN_OUT	Variant	 value "NULL": "RD_2" "RD_3" "RD_4" The following minimum user data size for the "GET" instruction
			with $1 - 4$ variables is guaranteed: 160 bytes

Assign the variables created in section <u>2.2.3</u> to the parameters of FB13 "GetData". The following table shows the assignment of the parameters of the "get" variable to the parameters of FB13 "GetData". Table 2-7

Parameters of FB13 "GetData"	Parameters of the "get" variable	Start value	Note				
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Read job. The Read job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Read job is completed with "done" = true or "error" = true.				
connectionId	connectionId	16#100	Enter the local ID of the connection that you defined during the connection configuration.				
done	done	false	-				
busy	busy	false	-				
error	error	false	-				
status	status	16#0	-				
ADDR_1	-	P#DB4.DBX0.0 BYTE 100	Only absolute addressing is permitted.				
RD_1	-	P#DB1.DBX0.0 BYTE 100	-				

Siemens AG 2020 All rights reserved

2.3 Install SIMATIC S7-1200 as Server

2.3.1 Create Send and Receive Data Areas

In the user program of the S7-1200 CPU you add the data blocks for saving the Send and Receive data.

Add data block for saving the Receive data

- In the project tree you navigate to the device folder of the S7-1200 CPU. The device folder contains structured objects and actions that belong to the device.
- In the device folder you navigate to the "Program blocks" subfolder and double-click the "Add new block" action. The "Add new block" dialog opens.



- 3. Click the "Data block (DB)" button.
- 4. Enter the name of the data block (DB), "RecvDataServer", for example.
- 5. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 6. Enable the "Add new and open" function.
- 7. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "RecvDataServer" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "RecvDataServer" DB opens in the working area of STEP 7 V16.

Add new block			×
Name:			
RecvDataServer			
	AI)		
	pe:	Global DB 🔻	
OB	Language:	DB	
Organization	Number:	3	
		Manual	
		Automatic	
	Description:		
-FB	Data blocks (DBs)	save program data.	
Function block			
FC			
Function			
	9		
	Ť.		
Data block			
	more		
> Additional inform	ation		
Add new and open			OK Cancel
L			

8. Alternatively, you can open the "RecvDataServer" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-1200 CPU and double-clicking the "RecvDataServer" DB.



S7 communication between SIMATIC S7-1200 and SIMATIC S7-300 with PUT/GET Entry-ID: 82212115, V 3.0, 11/2020 9. In the DB "RecvDataServer" you define the variable "data" of the data type "Array[0..199] of Byte".

Ree	RecvDataServer						
	Na	me		Data type	Offset	Start value	
-	•	Sta	atic				
	•	٠	data	Array[0199] of Byte	0.0		
	Add new>		<add new=""></add>] (

10. Because the communication partner, SIMATIC S7-300, for example, does not support DBs with optimized block access, you have to disable the "Optimized block access" option under "Attributes" in the Properties of the DB. DBs with standard access have a fixed structure. The data elements in the declaration include both a symbolic name and a fixed address in the block. The address is displayed in the "Offset" column. You can address the variables in this block both symbolically and absolutely.

RecvDataServer [DB3]	×
General Texts	
General Information Time stamps Compilation Protection Attributes Download without reinitialization	Attributes
	OK Cancel

Add data block for saving the Send data

- 1. Add another DB for saving the Send data.
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "SendDataServer", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "SendDataServer" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "SendDataServer" DB opens in the working area of STEP 7 V16.

Add new block			×
Name:			
SendDataServer	3		
	AI	🥃 Global DB 🔍 🔻	
OB	Language:	DB	
Organization block	Number:	4	
		Manual	
=	Description:		
Function block	Data blocks (DB	s) save program data.	
Function			
	2		
Data block			
	more		
Additional inform	nation		
Add new and open		0	OK Cancel

 Alternatively, you can open the "SendDataServer" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-1200 CPU and double-clicking the "SendDataServer" DB.



8. In the DB "SendDataServer" you define the variable "data" of the data type "Array[0..199] of Byte".

SendDa	ataServer			
Nam	e	Data type	Offset	Start value
🕣 💌 S	tatic			
	data	Array[0199] of Byte	0.0	
	<add new=""></add>			

9. Because the communication partner, SIMATIC S7-300, for example, does not support DBs with optimized block access, you have to disable the "Optimized block access" option under "Attributes" in the Properties of the DB. DBs with standard access have a fixed structure. The data elements in the declaration include both a symbolic name and a fixed address in the block. The address is displayed in the "Offset" column. You can address the variables in this block both symbolically and absolutely.

General Texts	
General Information Time stamps Compilation Protection Attributes Download without reinitialization	Attributes Only store in load memory Data block write-protected in the device Optimized block access Data block accessible from OPC UA Data block accessible via Web s

2.4 Download the Hardware Configuration and User Program

Requirements

You have already assigned the configured IP address and subnet mask to the S7-1200 CPU and possibly the CP.

Compile

- 10. In the project tree you mark the device folder of the S7-1200 CPU.
- 11. Click the "Compile" button in the toolbar. The hardware configuration and the software of the S7-1200 are compiled.



Download

- 1. In the project tree you mark the device folder of the S7-1200 CPU.
- 2. Click the "Download to device" button in the toolbar. The "Extended download to device" or "Load preview" dialog opens automatically.



- 3. The "Extended download to device" dialog opens automatically only if the access path from the PG/PC to the S7-1200 CPU has to be set. Make the following settings:
 - Type of the PG/PC interface: PN/IE
 - PG/PC interface: Network card of the PG/PC
 - Connection to interface/subnet: Subnet to which the S7-1200 CPU is connected
- 4. In the drop-down list box, you select "Show all compatible devices".
- 5. Click "Start Search".
- 6. The S7-1200 CPU is displayed in the "Select target device:" list. Mark the S7-1200 CPU.
- 7. Click the "Load" button.

	Configured	access nodes of "PLC_	1*			
	Device	Device typ	oe Slot	Interface type	Address	Subnet
-	PLC_1	CPU 1214	C DC/D 1 X1	PN/IE	192.168.178.2	PN/IE_1
	1	Type of the Connection to in	PG/PC interface: PG/PC interface: terface/subnet: 151 gateway.	PN/IE Wwxnet3 Ethe PN/IE_1	rnet Adapter	
	Select targ	et device:	Interface type	Address	Show all compatible d	evices
····	PLC_2	CPU 1214C DC/DC/DC	PN/IE	172.16.43.2		
Flash LED	sql1200 CPU-1214 —	CPU 1212C DC/DC/DC CPU 1214C DC/DC/DC 	ON IF	172.16.62.14 172.16.66.12 Access address	 CPU-1214 	
Inline status inform	ation			ſ	Display only error m	5 <u>Start search</u>
Found accessible	le device vm-z003	1axk		L. L.	_ clopicy only chorin	
Scan completed	l. 3 compatible de e information	vices of 20 accessible	devices found.			[

8. In the "Load preview" dialog you click the "Load" button to start the loading procedure.

Status	1	Target	Message		Acti	on
ŧ.	<u> </u>	▼ PLC_1	Ready for	loading.		Load 'PLC_1'
		 Protection 	n Protection	n from unauthorized access		
	4		Devices o internet r access, e more info http://ww	connected to an enterprise network or must be appropriately protected again e.g. by use of firewalls and network seg romation about industrial security, plea w.siemens.com/industrialsecurity	directly to the st unauthorized mentation. For ise visit	
	v	Stop mo	dules The modu	ules are stopped for downloading to de	evice.	Stop all
	0	Device c	onfigurati Delete ar	nd replace system data in target		Download to device
	0	Software	Download	d software to device		Consistent download
<	•		• •			· · ·) >

9. In the "Load results" dialog you enable the "Start module" action and click the "Finish" button to terminate the loading procedure.


3

Configuration of SIMATIC S7-300 in STEP 7 V16

- 1. Create a new project or open an existing project in STEP 7 V16 (TIA Portal V16).
- 2. Add the S7-300 CPU and create the hardware configuration.
- 3. Install the S7-300 CPU as client or server.
 - Client: Configure a new S7 connection and create the user program. In the user program you call the "PUT" instruction to write data to the S7-1200 CPU. In the user program you call the "GET" instruction to read data from the S7-1200 CPU. Add the data blocks for saving the Send and Receive data.
 - Server: Create the user program. In the user program you add data blocks in which the sent and received data is stored.

3.1 Create the Hardware Configuration

3.1.1 Set the IP address of the CPU

1. In the Device view or Network view of the "Devices & networks" editor you mark the S7-300 CPU.

The properties of the S7-300 CPU are displayed in the inspector window.

- 2. In the "General" tab under "PROFINET interface [X1] > Ethernet addresses" you enter the IP address and the subnet mask of the S7-300 CPU.
 - IP address: 172.16.43.37
 - Subnet mask: 255.255.0.0
- 3. Assign a subnet to the Ethernet interface of the S7-300 CPU. You have two options:
 - Click the "Add new subnet" button to add a new subnet.
 - Select an existing subnet.



3.1.2 Set the IP address of the CP

- In the Device view or Network view of the "Devices & networks" editor you mark the CP in the S7-300.
 The properties of the S7-300 CPU are displayed in the inspector window.
 - The properties of the S7-300 CPU are displayed in the inspector window.
- 2. In the "General" tab under "PROFINET interface [X1] > Ethernet addresses" you enter the IP address and the subnet mask of the CP.
 - IP address: 172.16.43.41
 - Subnet mask: 255.255.0.0
- 3. Assign a subnet to the Ethernet interface of the CP. You have two options:
 - Click the "Add new subnet" button to add a new subnet.
 - Select an existing subnet.

CP 343-1_1 [CP 343-1]		Properties	🗓 Info 追 🗓 Diagno
General IO tags	System constants Texts		
General Ontions	Ethernet addresses		
 PROFINET interface [X1] General 	Interface networked with		
Ethernet addresses	Subnet:	PN/IE_1 Add new subnet	
Time-of-day synchron: Operating mode	ation		
Diagnostics addresse	ISO protocol		
IP access protection	Use ISO protocol		
SNMP Web server Web discretion	MAC address:	08 -00 -06 -01 -00	- 00
I/O addresses	IP protocol		
		• Set IP address in the	project
		IP address: Subnet mask:	172 . 16 . 43 . 41 255 . 255 . 0 . 0

3.2 Install SIMATIC S7-300 as Client

3.2.1 Configuration of the S7 Connection

- 1. In the Network view you click the "Connections" icon to enable the Connection mode.
- 2. Select "S7 connection" as the connection type in the adjacent drop-down list box.

In the Network view, all devices that are available for an S7 connection are highlighted in color.

- 3. Right-click the S7-1200 CPU. The pop-up menu opens.
- 4. Select the "Add new connection" item. The "Add new connection" dialog opens.

Network	nnections 57 connection	
CPU 315-2 PN/DP	Device configuration Change device Migrate to \$7-1500	
PN/IE_1	↓ K Cut	Ctrl+X
	💼 Сору	Ctrl+C
	📋 Paste	Ctrl+V
	🗙 Delete	Del
	Rename	F2
	Assign to new DP master / IO controller	
	Disconnect from DP master system / IO system	
	Highlight DP master system / IO system	
	🚽 Go to topology view	
	Add new connection	
	Highlight connection partners	•

- 5. Specify the following connection partner: "Unspecified".
- 6. Select the local interface. In this example, the integrated interface of the S7-300 CPU is used. If the S7 connection is to be established via a CP, select the CP.
- 7. Enter the local ID for the S7 connection, 1 (hex), for example.
- 8. Click the "Add" button to add the unspecified S7 connection.
- 9. Click the "Close" button to end the dialog.

Add new connection	×
Please select connection partner for PLC_3:	Type: S7 connection
Unspecified	
Local ID (hex): 1	✓ Establish active connection ✓ One-way
	Add Close

Result

- An unspecified S7 connection is created.
- The connection path is displayed highlighted in the graphical area of the Network view.

Network	💌 🕮 📲 💷 🃩 🚍
	4 Highlighted: Connection
PLC_3 CPU 315-2 PN/DP	
PN/IE_1	

• The S7 connection is entered in the "Connections" table in the table area of the Network view.

Network overview	Connections	I/O communication		VPN	VPN TeleControl		
Y Local connection nam	e Local end point		Local ID (hex)	Partner ID (h	ex) Partner		Connection type
S7 connection one-w	ay 🚺 PLC_3 [CPU 3	15-2 PN/DP]	1 🔳		🔳 📔 Unkno	own 🕒	 S7 connection

Setting the connection parameters

- 1. In the "Connections" table, mark the unspecified S7 connections. The Properties of the S7 connection are displayed in the inspector window.
- 2. Under "General" the "General" tab displays the general connection parameters that identify the local endpoint. Under Partner you enter the IP address of the S7-1200 CPU, for example: 172.16.43.2. If you use a CP as partner instead of the integrated interface of the S7-1200 CPU, enter the IP address of the CP.

S7 connection one-way [S7 c	connection]		🗟 Proper	rties 🚺 Info	追 🗓 Diagno
General IO tags Sy	stem constants	Texts			
General Local ID	General				
Special connection properties	Connection				
Address details		Name: S7	connection one-way		
	Connection p	bath			
		Lo	cal		Partner
			-	-	?
	, Enc	d point: PL	C_3 [CPU 315-2 PN/DP]		Unknown
	Int	erface: PL	C_3, PROFINET-Schnittstel	le_1[X2] 🔻	Unknown
	Interfac	e type: Et	nernet		Ethernet
	2	ubnet: PN	/IE_1		
	A	ddress: 17	2.16.43.37		172.16.43.2
				AI	

 In the "General" tab under "Local ID" you see the local ID of the module from which the connection is viewed (local partner). You can change the local ID. You specify the local ID later at the "ID" input parameter of the "PUT" or "GET" instruction.

In this example we use the local ID with the value 1 (hex).

S7 connectio	🔍 Properties					
General	IO tags	System constar	nts	Texts		
General		Local ID				
Special conr Address deta	es Block	param	eters			
				Local ID	(hex):	1
					ID:	W#16#1

4. In the "General" under "Special connection properties" you see that the S7 connection is configured unilaterally. Unilaterally means that the connection partner is server for this connection and cannot actively send or receive. The S7 CPU that is used as the local endpoint for the S7 connection, actively establishes the S7 connection as a client and actively sends and receives.

S7 connection one-way [S7 con	nection] 📴 Properties 🚺 Info 🚯 🖫 Diagnostics 📑 🗖 🖃
General IO tags Syste	em constants Texts
General Local ID	Special connection properties
Special connection properties Address details	Local end point
	✓ One-way
	Active connection establishment
	The active connection establishment cannot be deactivated if the Partner TSAP in the address details has the value 3.
	Send operating mode messages

- In the "General" tab under "Address details" you enter the partner TSAP. Since the S7 connection is configured unilaterally to an unspecified partner, the partner TSAP is composed as follows: 03 <slot of CPU or CP>, for example:
 - S7-1200 CPU in slot 1: 03.01

S7 connection one-way [S7 con	nection]	🔍 Properties	🗓 Info 🔒 📱 Diagnostics
General IO tags Syste	em constants Text	S	
General	Address details		
Local ID			
Special connection properties			
Address details		Local	Partner
	End point:	PLC_3 [CPU 315-2 PN/DP]	Unknown
	Rack/slot:	0 2	0 1
	Connection res.		
	(hex):	10	▼ 03
	TSAP:	10.02	03.01
		SIMATIC-ACC	
	Subnet ID:	1A43 - 0001	·

3.2.2 Create Send and Receive Data Areas

Add data block for saving the Receive data

The data received from the S7-1200 CPU is stored in data block DB1 "RecvDataClient" of the S7-300 CPU.

- In the project tree you navigate to the device folder of the S7-300 CPU. The device folder contains structured objects and actions that belong to the device.
- In the device folder you navigate to the "Program blocks" subfolder and double-click the "Add new block" action. The "Add new block" dialog opens.



- 3. Click the "Data block (DB)" button.
- 4. Enter the name of the data block (DB), "RecvDataClient", for example.
- 5. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 6. Enable the "Add new and open" function.
- 7. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "RecvDataClient" DB is added to the "Program blocks" folder of the S7-300 CPU.
 - If the "Add new and open" function is enabled, the "RecvDataClient" DB opens in the working area of STEP 7 V16.

Add new block					×
Name:					
RecvDataClient	0				
(A	I				
	pe:	🧧 Global DB	-		
OB	Language:	DB			
Organization	Number:	1	1 V		
		O Manual			
		 Automatic 			
	Description:				
Eventing black	Data blocks (DBs) save program data.			
Function block					
					
FC					
Function					
Data block					
	more				
> Additional information	tion				
Add new and open)		ОК	Cancel
)		(🚹	
		7			

8. Alternatively, you can open the "RecvDataClient" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-300 CPU and double-clicking the "RecvDataClient" DB.

PLC_3 [CPU 315-2 PN/DP]
Device configuration
🛂 Online & diagnostics
🔻 ⋥ Program blocks
📫 Add new block
📲 Main [OB1]
🚛 GetData [FB13]
🚛 PutData [FB12]
🥃 GeneralData [DB10]
📒 InstGetData [DB13]
📒 InstPutData [DB12]
🧧 RecvDataClient [DB1]
🥃 RecvDataServer [DB3] 🍐 🏹 🎽
🧧 SendDataClient [DB2]
🥃 SendDataServer [DB4]

9. In the DB "RecvDataClient" you define the variable "data" of the data type Array[0..199] of Byte.

RecvDataClient										
	Name			Data type	Offset	Start value				
	•	Sta	atic							
		٠	data	Array[0199] of Byte	0.0					
	•		<add new=""></add>							

Add data block for saving the Send data

The data that is transferred to the S7-1200 CPU is stored in data block DB2 "SendDataClient" of the S7-300 CPU.

- 1. Add another DB for saving the Send data.
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "SendDataClient", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "SendDataClient" DB is added to the "Program blocks" folder of the S7-1200 CPU.
 - If the "Add new and open" function is enabled, the "SendDataClient" DB opens in the working area of STEP 7 V16.

Add new block			X
Neme			
SendDataClient	N -		
	AI	🧧 Global DB 💌	
-OB	Language:	DB	
Organization block	Number:	2	
Function block	Description: Data blocks (D	Manual Automatic Bs) save program data.	
FC			
Data block	D		
	more		
> Additional info	ormation		
Add new and op	en 🚺	6	OK Cancel

 Alternatively, you can open the "SendDataClient" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-300 CPU and double-clicking the "SendDataClient" DB.

PLC_3 [CPU 315-2 PN/DP]
Device configuration
🗓 Online & diagnostics
🔻 🛃 Program blocks
💣 Add new block
💶 Main [OB1]
💶 GetData [FB13]
💶 PutData [FB12]
🥃 GeneralData [DB10]
🥃 InstGetData [DB13]
🥃 InstPutData [DB12]
🥃 RecvDataClient [DB1]
📒 RecvDataServer [DB3]
🥃 SendDataClient [DB2]
SendDataServer [DB4]

8. In the DB "SendDataClient" you define the variable "data" of the data type "Array[0..199] of Byte".

SendDataClient						
Name Data type Offset Start value						Start value
	•	Sta	atic			
-		•	data	Array[0199] of Byte	0.0	
			<add new=""></add>			

3.2.3 Create Variables for Parameters of Function Blocks "PutData" and "GetData"

- 1. Add another DB to create variables for the parameters of function blocks "PutData" and "GetData".
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "GeneralData", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "GeneralData" DB is added to the "Program blocks" folder of the S7-300 CPU.
 - If the "Add new and open" function is enabled, the "GeneralData" DB opens in the working area of STEP 7 V16.

Add now black				×
Add new block				
Name:				
GeneralData	3			
	Type:	🧧 Global DB	¥	
OB	Language:	DB	T	
Organization block	Number:	10	-	
		🔿 Manual		
		 Automatic 		
FB	Description:			
Function block	Data blocks (DBs)	save program data.		
Function				
	2			
Data block				
	more			
> Additional information	ation		_	
Add new and open				OK Cancel

 Alternatively, you can open the "GeneralData" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-300 CPU and double-clicking the "GeneralData" DB.

PLC_3 [CPU 315-2 PN/DP]
Device configuration
🛂 Online & diagnostics
🔻 🛃 Program blocks
📫 Add new block
📲 Main [OB1]
🖅 GetData [FB13]
🖅 PutData [FB12]
🥃 GeneralData [DB10]
🧧 InstGetData [DB13] 🍐 🏹 🎽
📒 InstPutData [DB12]

8. Create the following variables to assign the input and output parameters of the FBs "PutData" and "GetData". PLC data types are used as data type.

Variable	PLC data type	Description
put	typePut	Variables for parameterizing FB12 "PutData".
get	typeGet	Variables for parameterizing FB13 "GetData".
diagnostic	typeDiagnostic	Variables to store the status of FBs "PutData" and "GetData" in case of error.

PLC Data Type "typePut"

The following table shows the structure of "typePut" data type.

Table 3-1

Parameter	Data type	Start value	Description
execute	Bool	false	Control parameter for FB12 "PutData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Bool	false	Status parameters
busy	Bool	false	Status parameters
error	Bool	false	Status parameters
status	Word	16#0	Status parameters

PLC Data Type "typeGet"

The following table shows the structure of the PLC data type "typeGet".

able 3-2	
----------	--

Parameter	Data type	Start value	Description
execute	Bool	false	Control parameter for FB13 "GetData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Bool	false	Status parameters
busy	Bool	false	Status parameters
error	Bool	false	Status parameters
status	Word	16#0	Status parameters

PLC Data Type "typeDiagnostic"

The following table shows the structure of the "typeDiagnostic" data type.

Table	3-3
-------	-----

Parameter	Data type	Start value	Description
statusPut	Word	16#0	Parameter to store the status of FB12 "PutData".
statusGet	Word	16#0	Parameter to store the status of FB13 "GetData".

3.2.4 Create a User Program

Overview

The following figure shows an overview of the user program in the S7-300 CPU and S7-1200 CPU.

Figure 3-1



FB12 "PutData"

The FB12 "PutData" calls the "PUT" instruction to execute the following functions:

- Send data to the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Write job is running, it is not possible to trigger a new Write job.
- Output the status of the FB and data transmission at the "status" output.

FB12 "PutData" is called cyclically in OB1.

The following figure shows the call of the FB12 "PutData" in OB1.

Figure 3-2

			_
	PutData		
Bool —	execute	done —	Bool
Word —	connectionId	busy —	Bool
		error —	Bool
		status —	
Any —	ADDR_1		— Any
Any —	ADDR_2		— Any
Any —	ADDR_3		— Any
Any —	ADDR_4		- Any
Anv —	SD_1		- Anv
Anv —	SD_2		Anv
Δην —	SD_3		Δnv
, (1)	SD 4		
Any —			- Any

The following table shows the parameters of FB12 "PutData".

Table	3-4
-------	-----

Name	P type	Data type	Comment
execute	IN	Bool	Control parameter:
connectionId	IN	Word	Local ID: Addressing parameter for specifying the connection to the partner CPU. Note The local ID is available in the Properties of the configured S7 connection. The local ID 1 (hex) is used in this example.
done	OUT	Bool	TRUE: The Write job was executed error-free.
busy	OUT	Bool	TRUE: The Write job is running.
error	OUT	Bool	Status parameters "error" and "status":
status	OUT	Word	 "error" = 0: "status" = 0000 (hex): neither warning nor error "status" <> 0000 (hex): Warning The "status" parameter provides detailed information. "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error.
ADDR_1	IN_OUT	Any	Pointer to the area to be written to in the partner CPU.
ADDR_2	IN_OUT	Any	Note
ADDR_3	IN_OUT	Any	I he following parameters are hidden and are preset with the value "NULL":
ADDR_4	IN_OUT	Any	 "ADDR_2" "ADDR_3" "ADDR_4" The following minimum user data size for the "PUT" instruction with 1 – 4 variables is guaranteed: 160 bytes
SD_1	IN_OUT	Any	Pointer to the area in your own S7 CPU that contains the data
SD_2	IN_OUT	Any	to be sent.
SD_3	IN_OUT	Any	The following parameters are hidden and are preset with the
SD_4	IN_OUT	Any	 value "NULL": "SD_2" "SD_3" "SD_4" The following minimum user data size for the "PUT" instruction with 1 – 4 variables is guaranteed: 160 bytes

Assign the variables created in section <u>3.2.3</u> to the parameters of FB12 "PutData". The following table shows the assignment of the parameters of the "put" variable to the parameters of FB12 "PutData".

Parameters of FB12 "PutData"	Parameters of the "put" variable	Start value	Note
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Write job. The Write job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Write job is completed with "done" = true or "error" = true.
connectionId	connectionId	16#1	Enter the local ID of the connection that you defined during the connection configuration.
done	done	false	_
busy	busy	false	-
error	error	false	-
status	status	16#0	-
ADDR_1	-	P#DB1.DBX0.0 BYTE 100.	Only absolute addressing is permitted.
SD_1	-	P#DB2.DBX0.0 BYTE 100	-

FB13 "GetData"

The FB13 "GetData" calls the "GET" instruction to execute the following functions:

- Receive data from the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Read job is running, it is not possible to trigger a new Read job.
- Output the status of the FB and data transmission at the "status" output.

FB13 "GetData" is called cyclically in OB1.

The following figure shows the call of the FB13 "GetData" in OB1.

Figure 3-3

			_
	GetData		
Bool —	execute	done —	Bool
Word —	connectionId	busy —	Bool
		error —	Bool
		status —	
Any —	ADDR_1		— Any
Any —	ADDR_2		— Any
Anv —	ADDR_3		- Anv
Anv —	ADDR_4		- Anv
Anv —	RD_1		— Anv
Any —	RD_2		- Anv
	RD_3		
	RD 4		Any
Any —			- Any

3 Configuration of SIMATIC S7-300 in STEP 7 V16

The following table shows the parameters of FB13 "GetData".

Table 3-6

Name	P type	Data type	Comment
execute	IN	Bool	Control parameter:
			Enables the Read job on a rising edge.
connectionId	IN	Word	Local ID: Addressing parameter for specifying the connection to the partner CPU. Note The local ID is available in the Properties of the configured S7 connection. The local ID 1 (hex) is used in this example.
done	OUT	Bool	TRUE: The Read job was executed error-free.
busy	OUT	Bool	TRUE: The read job is running.
error	OUT	Bool	Status parameters "error" and "status":
status	OUT	Word	 "error" = 0: "status" = 0000 (hex): neither warning nor error "status" <> 0000 (hex): Warning The "status" parameter provides detailed information. "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error.
ADDR_1	IN_OUT	Any	Pointer to the area to be read in the partner CPU.
ADDR_2	IN_OUT	Any	Note
ADDR_3	IN_OUT	Any	I he following parameters are hidden and are preset with the value "NULL":
ADDR_4	IN_OUT	Any	 "ADDR_2" "ADDR_3" "ADDR_4" The following minimum user data size for the "GET" instruction with 1 – 4 variables is guaranteed: 160 bytes
RD_1	IN_OUT	Any	Pointer to the area in your own S7 CPU in which the read data
RD_2	IN_OUT	Any	is stored.
RD_3	IN_OUT	Any	The following parameters are hidden and are preset with the
RD_4	IN_OUT	Any	 value "NULL": "RD_2" "RD_3" "RD_4" The following minimum user data size for the "GET" instruction with 1 – 4 variables is guaranteed: 160 bytes

Assign the variables created in section 3.2.3 to the parameters of FB13 "GetData". The following table shows the assignment of the parameters of the "get" variable to the parameters of FB13 "GetData".

Parameters of FB13 "GetData"	Parameters of the "get" variable	Start value	Note		
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Read job. The Read job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Read job is completed with "done" = true or "error" = true.		
connectionId	connectionId	16#100	Enter the local ID of the connection that you defined during the connection configuration.		
done	done	false	-		
busy	busy	false	-		
error	error	false	-		
status	status	16#0	-		
ADDR_1	-	P#DB4.DBX0.0 BYTE 100	Only absolute addressing is permitted.		
RD_1	-	P#DB1.DBX0.0 BYTE 100	-		

Table 3-7

3.3 Install SIMATIC S7-300 as Server

3.3.1 Create Send and Receive Data Areas

In the user program of the S7-300 CPU you add the data blocks for saving the Send and Receive data.

Add data block for saving the Receive data

- In the project tree you navigate to the device folder of the S7-300 CPU. The device folder contains structured objects and actions that belong to the device.
- In the device folder you navigate to the "Program blocks" subfolder and double-click the "Add new block" action. The "Add new block" dialog opens.



- 3. Click the "Data block (DB)" button.
- 4. Enter the name of the data block (DB), "RecvDataServer", for example.
- 5. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 6. Enable the "Add new and open" function.
- 7. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "RecvDataServer" DB is added to the "Program blocks" folder of the S7-300 CPU.
 - If the "Add new and open" function is enabled, the "RecvDataServer" DB opens in the working area of STEP 7 V16.

Add now black			×
Add new block			^^
Name:			
RecvDataServer	-0		
/ [AT		
	e:	🧧 Global DB 🛛 💌	
	Language:	DB	
Organization	Number:	3	
		🔿 Manual	
		Automatic	
	Description		
FB	Description:		
Function block	Data blocks (DB	s) save program data.	
Eunction			
Tunction			
DB 📈	3		
Data block			
> Additional inform	ation (
		6	
🛃 Add new and open	Le contraction de la contracti		OK Cancel

8. Alternatively, you can open the "RecvDataServer" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-300 CPU and double-clicking the "RecvDataServer" DB.



9. In the DB "RecvDataServer" you define the variable "data" of the data type "Array[0..199] of Byte".

Re	RecvDataServer								
	Name				Data type		Offset	Start value	1
-	 Static 								
	•	•	data		Array[0199] of Byte		0.0		
			<add n<="" th=""><th>ew></th><th></th><th></th><th></th><th></th><th></th></add>	ew>					

Add data block for saving the Send data

- 1. Add another DB for saving the Send data.
- 2. In the "Add new block" dialog you click the "Data block (DB)" button.
- 3. Enter the name of the DB, "SendDataServer", for example.
- 4. Enable the "Automatic" option to have the number of the DB assigned automatically by STEP 7.
- 5. Enable the "Add new and open" function.
- 6. Click the "OK" button to apply the settings and close the dialog.
 - In the project tree the "SendDataServer" DB is added to the "Program blocks" folder of the S7-300 CPU.
 - If the "Add new and open" function is enabled, the "SendDataServer" DB opens in the working area of STEP 7 V16.

Add new block			×
Name			
SendDataServer			
	AI		
	e:	GIODAT DB	
-OB	Language:	DB	
Organization	Number:	4	
		O Manual	
		Automatic	
FB	Description:		
Function block	Data blocks (DB	s) save program data.	
FC			
Function			
DB Data block	2		
	more		
> Additional inform	nation		
Add new and open		0	OK Cancel

 Alternatively, you can open the "SendDataServer" DB in the working area of STEP 7 V16 by going in the project tree to the "Program blocks" folder of the S7-300 CPU and double-clicking the "SendDataServer" DB.



8. In the DB "SendDataServer" you define the variable "data" of the data type "Array[0..199] of Byte".

Se	SendDataServer						
	Name			Data type	Offset	Start value	
	•	St	atic				
-	•	•	data	Array[0199] of Byte	0.0		
			<add new=""></add>				

3.4 Download the Hardware Configuration and User Program

Requirements

You have already assigned the configured IP address and subnet mask to the S7-300 CPU and possibly the CP.

Compile

- 1. In the project tree you mark the device folder of the S7-300 CPU.
- 2. Click the "Compile" button in the toolbar. The hardware configuration and the software of the S7-300 are compiled.

Vîè	Siemens - D:\Projekte\82212115_S7communication_S7-300\S7communication_S7-300\S
Pr	roject Edit View Insert Online Options Tools Window Help
	출 🞦 🗔 Save project 📑 🐰 🧃 🗎 🗙 🍤 ± (주 ± 🔩 🗓 🖸 🖳 🙀 💋 Go online
	Project tree
	Devices
	▼ 🗋 S7communication_S7-300
Ħ	🗳 Add new device
Ľ,	🖶 Devices & networks
	▶ []] PLC_3 [CPU 315-2 PN/DP]
	🕨 🔛 Ungrouped devices
	🕨 📷 Security settings
	Cross-device functions

Download

- 1. In the project tree you mark the device folder of the S7-300 CPU.
- 2. Click the "Download to device" button in the toolbar. The "Extended download to device" or "Load preview" dialog opens automatically.



- 3. The "Extended download to device" dialog opens automatically only if the access path from the PG/PC to the S7-300 CPU has to be set. Make the following settings:
 - Type of the PG/PC interface: PN/IE
 - PG/PC interface: Network card of the PG/PC
 - Connection to interface/subnet: Subnet to which the S7-300 CPU is connected
- 4. In the drop-down list box, you select "Show all compatible devices".
- 5. Click "Start Search".
- 6. The S7-300 CPU is displayed in the "Select target device:" list. Mark the S7-300 CPU.
- 7. Click the "Load" button.

3 Configuration of SIMATIC S7-300 in STEP 7 V16

	Device	Device typ	e Slo	t Interface type	Address	Subnet
—	PLC_1	CPU 1214	CDC/D 1>	(1 PN/IE	192.168.178.2	PN/IE_1
9		Type of the F F Connection to int	G/PC interfac G/PC interfac erface/subne	e: PN/IE e: PN/IE_1 y-	ernet Adapter	
	Select targe	et device:	Interface tv	ne Address	Show all compatible d	evices
···· —]	PLC_2	CPU 1214C DC/DC/DC	PN/IE	172.16.43.2		
Flash LED	sq11200 CPU-1214 	CPU 1212C DC/DC/DC 	PINIE 6	172.16.66.12 Access address	 CPU-1214 	
					(5 <u>Start set</u>
ne status information	1:				Display only error m	essages
Found accessible de	vice vm-z003	1axk	laviene farme			
scan completed. 3 c	ompatible de	vices of 20 accessible (levices tound		-	

8. In the "Load preview" dialog you click the "Load" button to start the loading procedure.

Status	1	Target	Message	Action
4I	<u> </u>	▼ PLC_3	Ready for loading.	Load 'PLC_3'
		 Protection 	Protection from unauthorized access	
	A		Devices connected to an enterprise network or directly to the internet must be appropriately protected against unauthorized access, e.g. by use of firewalls and network segmentation. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity	
	v	Stop modules	The modules are stopped for downloading to device.	Stop all
	0	 Device configurati. 	. Delete and replace system data in target	Download to device
	0	Software	Download software to device	Consistent download
٢ 🗌	_			<u> </u>

9. In the "Load results" dialog you enable the "Start module" action and click the "Finish" button to terminate the loading procedure.

tatus !	Target	Message	Action
+ (✓ ▼ PLC_3	Downloading to device completed without error.	Load 'PLC_3'
(Start modules	Start modules after downloading to device.	Start module
1			

4

Configuration of SIMATIC S7-300 in STEP 7 V5.6

- 1. Create a new project or open an existing project in STEP 7 V5.6.
- 2. Add an S7-300 station and create the hardware configuration.
- 3. Install the S7-300 CPU as client or server.
 - Client: Configure a new S7 connection and create the user program. In the user program you call the "PUT" instruction to write data to the S7-300 CPU. In the user program you call the "GET" instruction to read data from the S7-300 CPU. Add the data blocks for saving the Send and Receive data.
 - Server: Create the user program. In the user program you add data blocks in which the sent and received data is stored.

4.1 Create the Hardware Configuration

4.1.1 Open the Hardware Configuration

- 1. In the SIMATIC Manager you mark the SIMATIC S7-300 station that you have added to your STEP 7 project.
- 2. Double-click the "Hardware" item. The Hardware Configuration opens.



3. Drag and drop the relevant modules like Power Supply, CPU, CP etc. from the hardware catalog into the S7-300 profile channel.

📲 HW Config - [SIMATIC 315 (Configuration) PUT_GET]					
💵 Station	Edit Insert PLC View Options Window Help				
] 🗅 😅 🔓	5 🗣 🕾 🛍 🏜 🎼 📼 🗏 😥				
🚍(0) UR					
1					
2	CPU 315-2 PN/DP				
X1	MPI/DP				
X2	РN-Ю				
X2 P1 R	Port 1				
X2 P2 R	Port 2				
3					
4	DI16xDC24V				
5					

4.1.2 Define IP Address and Assign Subnet to the PROFINET interface of the CPU

1. Double-click the PROFINET interface of the S7-300 CPU. The Properties dialog of the PROFINET interface opens.

🖳 HW Config	g - [SIMATIC 315 (Configuration) PUT_GET]
Station	Edit Insert PLC View Options Window Help
] 🗅 😅 🔓	🖩 🖏 🚑 🖻 🛍 🕍 👔 🖪 🔛 🎇 💦
🚍(0) UR	
1	
2	CPU 315-2 PN/DP
X1	MPI/DP
X2	PN-10
X2P1R	Port 1 (🔊)
X2 P2 R	Port 2
3	
4	DI16xDC24V
5	

2. Click the "Properties" button to change the IP address and subnet mask and assign the subnet.

Properties - PN-IO (R0/S2.2)		×
Media redun General	dancy Addresses	Time-of-Day Synchronization PROFINET I-Device	Options Synchronization
Short description: Device name: Use different m	PN-IO PN-IO nethod to obtain device	name	
Interface	e replacement without	xchangeable medium	
Device Address:	0 172.16.43.37		
Networked: Comment:	yes	Properties	
			< >
ОК		Canc	el Help

- 3. Enter the IP address and subnet mask, for example:
 - IP address: 172.16.43.37
 - Subnet mask: 255.255.0.0
- 4. Click the "New..." button to create a new subnet.
- 5. Select the new subnet or another existing subnet.
- 6. Click the "OK" button to apply the IP address and subnet mask and assign the selected subnet to the PROFINET interface of the S7-300 CPU.

Properties - Ethernet interface PN-IO (R0/S2.2)	\times
General Parameters	1
IP address: Gateway	
Subnet mask: 255.255.0.0	
Use different method to obtain IP address Address:	
Subnet:	
not networked New	0
Properties Delete	5
OK Cancel Help	

4.1.3 Define IP Address and Assign Subnet to the PROFINET interface of the CP

- 1. Double-click the PROFINET interface of the CP. The Properties dialog of the PROFINET interface opens.
- 2. Click the "Properties" button to change the IP address and subnet mask and assign the subnet.

Properties - PN-IO-1	I (R0/S5.1)			×
General Addresses	s IP Configuration	PROFINET I-Device	Synchronization Me	dia redundancy
Short description:	PN-IO			
Device name:	PN-IO-1			
🔲 Use different m	ethod to obtain dev	ice name		
Support device	e replacement withou	ut exchangeable medium		
Type:	Ethernet			
Device	0			
Address:	172.16.43.41			
Networked:	yes	Properties		
Comment:				
				^
				~
ОК			Cancel	Help

- 3. Enter the IP address and subnet mask, for example:
 - IP address: 172.16.43.41
 - Subnet mask: 255.255.0.0
- 4. Click the "New..." button to create a new subnet.
- 5. Select the new subnet or another existing subnet.
- 6. Click the "OK" button to apply the IP address and subnet mask and assign the selected subnet to the PROFINET interface of the CP.

Properties - Ethernet interface PN-IO (R0/S5.1)	Х
General Parameters	_
Set MAC address / use ISO protocol	
MAC address: If a subnet is selected, the next available addresses are suggested.	
IP address: 172.16.43.41 Subnet mask: 255.255.0.0 Gateway © Do not use router Use router All Address:	
Subnet: not networked New New	0
Properties.)
OK Cancel Help	

4.2 Configure the SIMATC S7-300 as Client

The S7-300 station actively sets up the S7 connection. In this case you must configure an S7 connection for the S7-300 CPU and call the following function blocks (FBs) in the user program:

- FB14 "GET": You use FB14 "GET" to read data from a partner CPU. With an S7-400 CPU you use the system function block SFB14 "GET".
- FB15 "PUT": You use FB15 "PUT" to write data to a partner CPU. With an S7-400 CPU you use the system function block SFB15 "PUT".

4.2.1 Configuration of the S7 Connection

Add an S7 Connection

 In the SIMATIC Manager you open the "NetPro" tool via the menu "Options > Configure Network". In "NetPro" you configure the S7 connection for the S7-300 CPU.

SIMATIC Manager - PUT_GET		
File Edit Insert PLC View	Options Window Help	
🗅 🚅 🎛 🛲 X 🖻 🖻	Customize	Ctrl+Alt+E
PUT_GET D:\Projekte\PutC	Access Protection Change Log	> >
SIMATIC 315 - SIMATIC 315 - SI CPU 315-2 PN/DP - SI S7 program - Sources	Text Libraries Language for Display Devices Manage Multilingual Texts	>
Elocks	Rewire Run-Time Properties	
	Compare Blocks Reference Data Define Global Data	>
	Configure Network	
	Simulate Modules Configure Process Diagnostics	
	CAx Data	>

- 2. Mark the CPU in the SIMATIC S7-300 station.
- 3. Open the "Insert New Connection" dialog via the menu "Insert > New Connection...".

Siemens AG 2020 All rights reserved



- 4. Select the "Unspecified" item under "Connection Partner".
- 5. Select "S7 connection" as the connection type.
- 6. Click "Apply". The Properties dialog of the S7 connection opens.
| Insert New Conr | nection | × |
|---------------------------------|--|------|
| Connection Pa | artner
ccurrent project
UT_GET
(Unspecified)
All broadcast statio
All multicast statio
known project | |
| Project:
Station:
Module: | (Unspecified) | ₹ |
| Connection
Type: | S7 connection | |
| ОК | Apply Cancel | Help |

Define connection parameters

In the Properties dialog of the S7 connection you define the connection parameters.

- 1. Enable the "Establish an active connection" function.
- Select the local interface. In this example, the integrated interface of the S7-300 CPU is used. If the S7 connection is to be established via a CP, select the CP.
- Enter the IP address of the partner CPU. In this example you enter the IP address of the S7-1200 CPU: 172.16.43.2.
 If you use a CP as partner instead of the integrated interface of the S7-1200 CPU, enter the IP address of the CP.
- 4. You specify the local ID of the connection in the user program at the input parameter "ID" of the function blocks FB/SFB14 "GET" and FB/SFB15 "PUT".
- 5. Click the "Address Details..." button. The "Address Details" dialog opens.

Properties - S7 conn	ection	×
General Status Info	omation	
Local Connection ☐ Configured dy ☑ Configured at ☑ Establish an at ☑ Send operation	The End Point	}
Connection Path	Local Partner SIMATIC 315/	_
Interface:	CPU 315-2 PN/DP	- -
Subnet:	Ethemet(1) [Industrial Ethemet] [Industrial Ethemet]	
Address:	172.16.43.37 Address Details	
ОК	Cancel	elp

- 6. Enter the rack and slot of the partner CPU or CP. The S7-1200 CPU uses Rack 0 and Slot 1.
- Select the connection resource 03(hex) for the partner CPU because the S7 connection is only configured unilaterally in the S7-300 CPU. With these settings the TSAP has the value 03.01 in the partner CPU.
- 8. Apply the settings with "OK".

Address Details			\times
	Local	Partner	
End Point:	SIMATIC 315/ CPU 315-2 PN/DP	Unknown	_
Rack/Slot:	0 2	0 1	
Connection Resource (hex):	10 💌		
TSAP:	10.02	03.01	
			_
S7 Subnet ID:	007C - 0012		
ОК		Cancel Help	

- 9. Likewise apply the settings in the Properties dialog of the S7 connection with "OK".
- 10. In the "Insert New Connection" dialog you click the "OK" button to close the dialog. The S7 connection is inserted in "NetPro".

Load the S7 Connection

1. In "NetPro" you mark the S7 CPU in the SIMATIC S7-300 station. The connection table shows all the connections configured for the S7-CPU.



- 2. In "NetPro" you mark the SIMATIC S7-300 station.
- 3. In the toolbar you click the "Save and Compile" button to save and compile the connection configuration.
- 4. In the toolbar you click the "Load marked station(s)" button to load the connection configuration into the S7 CPU.



4.2.2 Create Send and Receive Data Areas

Add data block for saving the Receive data

The data received from the S7-1200 CPU is stored in data block DB1 "RecvDataClient" of the S7-300 CPU.

- 1. In your STEP 7 project you right-click the "Blocks" folder. The pop-up menu opens.
- You select the menu "Insert New Object > Data Block". The Properties dialog of the DB opens.



- 3. You enter the name and type of block that is to be created, DB1, for example.
- 4. Enter a symbolic name for the block, "RecvDataClient", for example.
- 5. Apply the settings with "OK". Block DB1 "RecvDataClient" is added to the "Blocks" folder.

Properties - Data Block		×
General - Part 1 General	Part 2 AII) Attributes	
Name:	DB1 (AI)	
Symbolic Name:	RecvDataClient	
Symbol Comment:		
Created in Language:	DB	
Project Path:	PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program\Blocks \DB1	-
Storage location of project:	D:\Projekte\PutGet	-
Date created:	Code Interface 08/06/2020 09:18:11 AM	
Last modified:	08/05/2020 03:45:49 PM 08/05/2020 03:45:49 PM	
Comment:		-
ОК	Cancel Help	

6. Double-click DB1 "RecvDataClient" in the "Blocks" folder. The DB opens.

🍠 SIMATIC Manager - PUT_GET			
File Edit Insert PLC View Option	ns Window Hel	lp	
🗅 😅 🖁 🛲 🕺 🛍 💼	g <u>e</u> <u>e</u>	🔠 🏢 🔁 < No Filter >	· 💽 💆 📲
PUT_GET D:\Projekte\PutGet PUT_GET SIMATIC 315 CPU 315-2 PN/DP S7 program Sources	Object name Systemdaten OB1 FB1	Symbolic name CYCL_EXC PutData	Created in language FBD FBD
Blocks	FB14 FB15	GetData GET PUT RecvDataClient	FBD STL STI
		Sendo atabient RecvDataServer SendDataServer GeneralData InstPutData InstGetData typePut typeGet typeDiagnostic Client Server	DB DB DB STL STL STL
	🛃 SFB15		STL

7. In the DB "RecvDataClient" you define the variable "data" of the data type "Array[0..199] of Byte".

Ħ	K L	AD/ST	L/F	BD -	DB1	- "RecvDa	ataClien	t"						
	File	Edit	Ir	nsert	PLC	Debug	View	Opti	ons	Windo	w	Help		
	D	产 🔓	2		s 3	K 🖻 🕻	3 0	α	6 % s	b E	- S	. 60	r !	$\ll \gg!$
:	DB1 "RecvDataClient" PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\\DB1													
10														
	Add	lress	N	lame		Туре			Init	ial v	alu	e		
		0.0				STRUCT								
		+0.0	D	dat	a	ARRAY [0199	1	B#16	5#0				
		*1.0	D			BYTE								
	-	=200.0	D			END_ST	RUCT							
ľ														

Add data block for saving the Send data

The data that is transferred to the S7-1200 CPU is stored in data block DB2 "SendDataClient" of the S7-300 CPU.

- 1. Add another DB for saving the Send data.
- 2. You enter the name and type of block that is to be created, DB2, for example.
- 3. Enter a symbolic name for the block, "SendDataClient", for example.
- 4. Apply the settings with "OK". Block DB2 "SendDataClient" is added to the "Blocks" folder.

Properties - Data Block		×
General - Part 1 General	- Part 2 AI Attributes	
Name:	DB2	
Symbolic Name:	SendDataClient	
Symbol Comment:		
Created in Language:	DB	
Project Path:	PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program\Blocks \DB2	
Storage location of project:	D:\Projekte\PutGet	
Date created:	Code Interface 08/05/2020 03:44:01 PM	
Last modified:	08/05/2020 03:45:49 PM 08/05/2020 03:45:49 PM	
Comment:		-
OK	Cancel Help	

5. Double-click DB2 "SendDataClient" in the "Blocks" folder. The DB opens.

····• • • • • • • • • • • • • • • •			
SIMATIC Manager - PUT_GET			
File Edit Insert PLC View Option	ns Window He	lp	
🗅 🕞 🎛 🛲 🐰 🖻 🛍		1 No Filter >	- 7/ 8
BUT_GET D:\Projekte\PutGet			
□- 🎒 PUT_GET	Object name	Symbolic name	Created in language
SIMATIC 315	🚔 Systemdaten		
🖻 📓 CPU 315-2 PN/DP	🖬 OB1	CYCL_EXC	FBD
⊡ S7 program	🕀 FB1	PutData	FBD
	🖬 FB2	GetData	FBD
BIOCKS	🗗 FB14	GET	STL
	🛃 FB15	PUT	STL
	STR DB1		DB
	🖬 DB🚧	SendDataClient	DB
	DB3	RecyDataServer	DR / 💽 🕺
	DB4	SendDataServer	DB
	DB10	GeneralData	DB
	DB11	InstPutData	DB
	DB12	InstGetData	DB
	DDT1	typePut	STL
	UDT2	typeGet	STL
		typeDiagnostic	STL
		Client	
	Server	Server	
	SFB14		SIL
	SFB15		SIL

6. In the DB2 "SendDataClient" you define the variable "data" of the data type "Array[0..99] of Byte".

💫 LAD/STL/FBD - DB2 "SendDataClient"								
File Edit	Insert	PLC	Debug	View	Options	Window	Help	
🗋 🚔 🔓		🚭 3	K 🖻 f	3 0	$\simeq \left \theta_{0}^{0} \right $	🏜 🖂	≌∎ & !≪ ≫!	
DB2 "SendDataClient" PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\\DB2								
Address Name Type Initial value								
Address	Name	•	1	Туре		Init	ial value	
Address 0.0	Name	2		Type STRUCI	:	Init	ial value	
Address 0.0 +0.0	Name	ta	, , ,	Type STRUCT ARRAY (0199]	Init: B#16	ial value	
Address 0.0 +0.0 +1.0	Name dat	ta	, , ,	Type STRUCT ARRAY (BYTE	0199]	Init: B#163	ial value	
Address 0.0 +0.0 *1.0 =200.0	Name dat	ta		Type STRUCT ARRAY (BYTE END_ST	0199] RUCT	Init: B#163	ial value #0	

4.2.3 Create Variables for Parameters of Function Blocks "PutData" and "GetData"

- 1. Add another DB to create variables for the parameters of function blocks "PutData" and "GetData".
- 2. You enter the name and type of block that is to be created, DB10, for example.
- 3. Enter a symbolic name for the block, "GeneralData", for example.
- 4. Apply the settings with "OK". Block DB10 "GeneralData" is added to the "Blocks" folder.

Properties - Data Block		×
General - Part 1 General	- Part 2 Attributes	
Name:	DB10	
Symbolic Name:	GeneralData	
Symbol Comment:		-
Created in Language:	DB	
Project Path:	PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program\Blocks \DB10	
Storage location of project:	D:\Projekte\PutGet	-
Date created:	Code Interface 08/05/2020 03:48:06 PM	
Last modified:	08/13/2020 09:29:53 AM 08/13/2020 09:29:53 AM	
Comment:		
ОК	Cancel Help	

5. Create the following variables to assign the input and output parameters of the FBs "PutData" and "GetData". UDTs are used as data types.

Variable	UDT	Description
put	UDT1 "typePut"	Variables for parameterizing FB1 "PutData".
get	UDT2 "typeGet"	Variables for parameterizing FB2 "GetData".
diagnostic	UDT3 "typeDiagnostic"	Variables to store the status of FBs "PutData" and "GetData" in case of error.

UDT1 "typePut"

The following table shows the structure of UDT1 "typePut".

Table 4-1

Parameter	Data type	Start value	Description
execute	Boolean	false	Control parameter for FB1 "PutData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Bool	false	Status parameters
busy	Bool	false	Status parameters
error	Bool	false	Status parameters
status	Word	16#0	Status parameters

UDT2 "typeGet"

The following table shows the structure of UDT2 "typeGet".

Table 4-2

Parameter	Data type	Start value	Description
execute	Bool	false	Control parameter for FB2 "GetData".
connectionId	Word	16#0	Addressing parameter for specifying the connection to the communication partner.
done	Bool	false	Status parameters
busy	Bool	false	Status parameters
error	Bool	false	Status parameters
status	Word	16#0	Status parameters

UDT3 "typeDiagnostic"

The following table shows the structure of UDT3 "typeDiagnostic".

Table 4-3

Parameter	Data type	Start value	Description
statusPut	Word	16#0	Parameter to store the status of FB1 "PutData".
statusGet	Word	16#0	Parameter to store the status of FB2 "GetData".

4.2.4 Create a User Program

In the user program of the S7-300 CPU you call the function blocks FB14 "GET" and FB15 "PUT" for data transfer.

If you are using the integrated PROFINET interface of the S7-300 CPU for data communication via S7 connections, then utilize function blocks FB14 "GET" and FB15 "PUT" from the Standard Library under "Communication Blocks -> Blocks" with the family "CPU_300".

If you are using an Industrial Ethernet CP or PROFIBUS CP for data communication via S7 connections in the S7-300 station, then utilize the function blocks FB14 "GET" and FB15 "PUT" from the "SIMATIC_NET_CP" library under "CP 300 > Blocks".

Copy the FBs from the Standard Library or from the "SIMATIC_NET_CP" library and insert them in your STEP 7 project.

Note The system function blocks SFB14 "GET" and SFB15 "PUT" for the S7-400 are located in the Standard Library under "System Function Blocks".

The following figure shows the block calls in the S7-300 CPU and S7-1200 CPU. Figure 4-1





FB1 "PutData"

The FB1 "PutData" calls the FB15 "PUT" to execute the following functions:

- Send data to the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Write job is running, it is not possible to trigger a new Write job.
- Output the status of the FB and data transmission at the "status" output.

FB1 "PutData" is called cyclically in OB1.

The following figure shows the call of the FB1 "PutData" in OB1.



Name	P type	Data type	Comment
execute	IN	Bool	Control parameter: Enables the Write job on a rising edge.
connectionId	IN	Word	Reference to the local connection description (preset by the connection configuration in in "NetPro"). In this example the value w#16#1 is preset by the configuration connection.
done	OUT	Bool	TRUE: The Write job was executed error-free.
busy	OUT	Bool	TRUE: The Write job is running.
error	OUT	Bool	Status parameters "error" and "status":
status	OUT	Word	 "error" = 0: "status" = 0000 (hex): neither warning nor error "status" <> 0000 (hex): Warning The "status" parameter provides detailed information. "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error.

FB1	"PutData"	has	the	following	parameters.
-----	-----------	-----	-----	-----------	-------------

Table 4-4

Assign the variables created in section <u>4.2.3</u> to the parameters of FB1 "PutData".

The following table shows the assignment of the parameters of the "put" variable to the parameters of FB1 "PutData".

Table 4-5

Parameters of FB1 "PutData"	Parameters of the "put" variable	Start value	Note
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Write job. The Write job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Write job is completed with "done" = true or "error" = true.
connectionId	connectionId	16#1	Enter the local ID of the connection that you defined during the connection configuration.
done	done	false	-
busy	busy	false	-
error	error	false	_
status	status	16#0	_

S7 communication between SIMATIC S7-1200 and SIMATIC S7-300 with PUT/GET Entry-ID: 82212115, V 3.0, 11/2020

Open FB1 "PutData" to define the Receive and Send data areas in FB15 "PUT".

Receive data area

At input "ADDR_1" of FB15 "PUT" you refer to the area in the partner CPU (S7-1200 CPU, for example) to which you want to write.

• P#DB1.DBX0.0 BYTE 100.

Note The following minimum user data size for the "PUT" instruction is guaranteed: 160 bytes

Send data area

At input "SD_1" of FB15 "PUT" you refer to the area in your own S7 CPU (S7-300 CPU) that contains the data to be sent.

- P#DB2.DBX0.0 BYTE 100
- **Note** The following minimum user data size for the "PUT" instruction is guaranteed: 160 bytes

FB 2 "GetData"

The FB2 "GetData" calls the "GET" instruction to execute the following functions:

- Receive data from the partner via the configured connection as soon as the input "execute" recognizes a positive edge. When the Read job is running, it is not possible to trigger a new Read job.
- Output the status of the FB and data transmission at the "status" output.

FB2 "GetData" is called cyclically in OB1.

The following figure shows the call of the FB2 "GetData" in OB1. Figure 4-3



Name	P type	Data type	Comment
execute	IN	Bool	Control parameter: Enables the Read job on a rising edge.
connectionId	IN	Word	Reference to the local connection description (preset by the connection configuration in in "NetPro"). In this example the value w#16#1 is preset by the configuration connection.
done	OUT	Bool	TRUE: The Read job was executed error-free.
busy	OUT	Bool	TRUE: The read job is running.
error	OUT	Bool	Status parameters "error" and "status":
status	OUT	Word	 "error" = 0: "status" = 0000 (hex): neither warning nor error "status" <> 0000 (hex): Warning The "status" parameter provides detailed information. "error" = 1: An error has occurred. The "status" parameter provides detailed information about the type of error.

FB2 "PutData" has the following parameters.

Table 4-6

Assign the variables created in section <u>4.2.3</u> to the parameters of FB2 "GetData".

The following table shows the assignment of the parameters of the "get" variable to the parameters of FB2 "GetData".

Та	ble	4-	7
	~~~		•

Parameters of FB2 "GetData"	Parameters of the "get" variable	Start value	Note
execute	execute	false	Set the "execute" parameter to the value "TRUE" to start the Read job. The Read job is started on a rising edge. Reset the "execute" parameter to the value "FALSE" if the Read job is completed with "done" = true or "error" = true.
connectionId	connectionId	16#1	Enter the local ID of the connection that you defined during the connection configuration.
done	done	false	-
busy	busy	false	-
error	error	false	_
status	status	16#0	-

Open FB2 "GetData" to define the Receive and Send data areas in FB14 "GET".

### Send data area

At input "ADDR_1" of FB14 "GET" you refer to the area in the partner CPU (S7-1200 CPU, for example) which you want to read.

- P#DB4.DBX0.0 BYTE 100
- **Note** The following minimum user data size for the "GET" instruction is guaranteed: 160 bytes

#### Receive data area

At input "RD_1" of FB14 "GET" you refer to the area in your own S7 CPU (S7-300 CPU) that contains the data to be read.

- P#DB1.DBX0.0 BYTE 100
- **Note** The following minimum user data size for the "GET" instruction is guaranteed: 160 bytes

### 4.3 Configure the SIMATC S7-300 as Server

### 4.3.1 Create a User Program

In the user program of the S7-300 CPU you add the data blocks for saving the Send and Receive data.

### Add data block for saving the Send data

- 1. In your STEP 7 project you right-click the "Blocks" folder. The pop-up menu opens.
- 2. You select the menu "Insert New Object > Data Block". The Properties dialog of the DB opens.

🎝 SIMATIC Manager - PUT_GET				
File Edit Insert PLC View Optio	ons Window Help	)		
🗋 🗅 😅 📲 🛲 🛛 👗 🗈 💼 🕍		🏥 🏢 🔁 🛛 < No Filter >	- y 4	🖁 🎟 🖻 🗖 😽
DUT_GET D:\Projekte\PutGet				
PUT_GET     SIMATIC 315     G CPU 315.2 PN/DP     Sources     Sources     Blocks	Object name                Systemdaten                 GB1                 FB1                 FB1                 FB1                 FB1                 FB2                 FB14                 FB15                 DB2                 DB3                 DB10                 DB12                 UDT1                 UDT2                 UDT3                 SFB14                    SFB14                 SFB15	Symbolic name          CYCL_EXC       PutData       GetData       GET       PUT       Open Object       Cut       Copy       Paste       Delete       Insert New Object       Compare Blocks       Reference Data       Print       Rename       Object Properties       Snecial Object Properties	Created in language FBD FBD STL STL Ctrl+Alt+O Ctrl+X Ctrl+C Ctrl+V Del Ctrl+V FBD SEL Ctrl+Return S	Size in the work me Type SDB 426 Organiz 528 Functio 527 Functio 527 Functio 528 Functio 529 Functio 520 Data B 236 Data B 237 Data B 238 Data B 239 Data B 230 Data

- 3. You enter the name and type of block that is to be created, DB4, for example.
- 4. Enter a symbolic name for the block, "SendDataServer", for example.
- 5. Apply the settings with "OK".

Block DB4 "SendDataServer" is added to the "Blocks" folder.

Properties - Data Block		×
General - Part 1 General	- Part 2 AII) Attributes	
Name:	DB4 (AI)	
Symbolic Name:	SendDataServer 4	1
Symbol Comment:		1
Created in Language:	DB	
Project Path:	PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program\Blocks \DB4	
Storage location of project:	D:\Projekte\PutGet	
Date created:	Code Interface 08/06/2020 12:21:46 PM	
Last modified:	08/06/2020 12:22:35 PM 08/06/2020 12:22:35 PM	
Comment:		Ľ
ОК	Cancel Help	

6. Double-click DB4 "SendDataServer" in the "Blocks" folder. The DB opens.

🍠 SIMATIC Manager - PUT_GET			
File Edit Insert PLC View Optio	ns Window Hel	р	
🗋 D 🚅   🎛 🛲   🐰 🖻 💼   🎪	9 <u>9</u> <u>9</u> <u>5</u>	🟥 🏢 🔁 🔤 No Filter >	▼ 20 8
PUT_GET D:\Projekte\PutGet			
PUT_GET	Object name	Symbolic name	Created in language
SIMATIC 315	🚵 Systemdaten		
⊡ ··· 📓 CPU 315-2 PN/DP	🖬 0B1	CYCL_EXC	FBD
Erist S7 program	🕞 FB1	PutData	FBD
	🖬 FB2	GetData	FBD
DIOCKS	<table-of-contents> FB14</table-of-contents>	GET	STL
	<table-of-contents> FB15</table-of-contents>	PUT	STL
	B DB1	RecvDataClient	DB
	DB 🖉	SendDataClient	DB
	DB3	RecyDataServer	DB
	🖬 DB4	SendDataServer	DB
	- DRI0	GeneralData	
	DB11	InstPutData	DB
	DB12	InstGetData	DB
	DDT1	typePut	STL
	UDT2	typeGet	STL
	UDT3	typeDiagnostic	STL
	Client	Client	
	Server	Server	
	SFB14		STL
	SFB15		STL

7. In the DB4 "SendDataServer" you define the variable "data" of the data type "Array[0..199] of Byte".

#	🖏 L	AD/STL	/FBD	- DB4	- "SendD	ataServe	er"					
ł	File	Edit	Insert	PLC	Debug	View	Options	Wind	low	Help	)	
	D	🖻 🔓		<b>s</b>   3	X 🖻 🛙		$\simeq \left   \theta_{0}^{n} \right.$	<b>ša</b>   [*		9 <b>6</b> 6	r !≪	$\gg!$
1	⊐ D	B4 "S	endDa	itaServe	r" PU1	r_get∖si	MATIC 315	5\CPU	315-2	2 PN/I	DP\\D	B4
	Add	ress	Name		1	Туре		In	itia	al va	lue	
	Add	ress 0.0	Name	-		Type STRUCT		In	itia	al va	lue	
	Add	ress 0.0 +0.0	Name dat	:a	1	Type STRUCT ARRAY [(	0199]	In B‡	itia 16‡0	al va	lue	
	Add	ress 0.0 +0.0 *1.0	Name dat	a		Type STRUCT ARRAY [( BYTE	0199]	In B‡	itia 16#0	al va	lue	

### Add data block for saving the Receive data

- 1. Add another DB for saving the Receive data.
- 2. You enter the name and type of block that is to be created, DB3, for example.
- 3. Enter a symbolic name for the block, "RecvDataServer", for example.
- 4. Apply the settings with "OK". Block DB3 "RecvDataServer" is added to the "Blocks" folder.

Properties - Data Block	Properties - Data Block				
General - Part 1 General - Part 2 AI) Attributes					
Name:	DB3				
Symbolic Name:	RecvDataServer				
Symbol Comment:			-		
Created in Language:	DB				
Project Path:	PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program\Blocks \DB3				
Storage location of project:	D:\Projekte\PutGet				
	Code	Interface			
Date created:	08/06/2020 12:18:38 PM				
Last modified:	08/06/2020 12:20:12 PM	08/06/2020 12:20:12 PM			
Comment:		^			
		~			
ОК		Cancel Help			

5. Double-click DB3 "RecvDataServer" in the "Blocks" folder. The DB opens.

SIMATIC Manager - PUT_GET			
File Edit Insert PLC View Optio	ns Window Help		
🗅 😅 🖁 🛲 🕹 🛍 💼		🗄 🏢   🔁   < No Filter >	- V B
PUT GET D:\Projekte\PutGet			
	Object name	Symbolic name	Created in language
SIMATIC 315	Sustemdaten		
📄 🖳 🚺 CPU 315-2 PN/DP		CYCL EXC	FBD
🔄 🔄 🔄 S7 program	FB1	PutData	FBD
B Sources	FB2	GetData	FBD
Blocks	🗩 FB14	GET	STL
	🗗 FB15	PUT	STL
	DB1	RecvDataClient	DB
		SendDataClient	DR
	🖬 DB3	RecvDataServer	DB
	1 DB4	SendDataServer	
	🕀 DB10	GeneralData	DB
	🕀 DB11	InstPutData	DB
	DB12	InstGetData	DB
	DDT1	typePut	STL
	UDT2	typeGet	STL
		typeDiagnostic	STL
	Client	Client	
	Server	Server	0.71
	SFB14		SIL
	Preio		SIL

6. In the DB3 "RecvDataServer" you define the variable "data" of the data type "Array[0..99] of Byte".

	🙀 LAD/STL/FBD - DB3 "RecvDataServer"								
File	Edit	Insert	PLC	Debug	View	Options	Window	Help	
D	🖻 🔓	• 日		X 🖻 🛙	9   M	$\simeq    \pmb{\theta}_0^0  $	🏜   🖂	º <u>∎</u> &   !≪ ≫!	
•	DB3 "RecvDataServer" PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\\DB3								
Ad	dress	s Name			Туре		Initia	Initial value	
	0.0				STRUCT				
	+0.0	dat	a	i	ARRAY[0199]		B#16#0	0	
	*1.0			1	BYTE				
	=200.0			1	END_STRUCT				

## 4.4 Download the Hardware Configuration and User Program

### Requirements

You have already assigned the configured IP address and subnet mask to the S7-300 CPU.

### Instructions

1. In the SIMATIC Manager, you select the menu "Options > Set PG/PC Interface...". The "Set PG/PC Interface" dialog opens.

SIMATIC Manager - PUT_GET		
File Edit Insert PLC View	Options Window Help	
🗋 🗅 😅 📲 🚟 🖌 🛍 💼	Customize	Ctrl+Alt+E
	Access Protection	>
	Change Log	>
	Text Libraries	
🚊 🖓 🚺 CPU 315-2 PN/DP	Language for Display Devices	
🖻 💼 S7 program	Manage Multilingual Texts	>
	Manage Matinguar rexts	
_	Rewire	
	Run-Time Properties	
	Compare Blocks	
	Reference Data	>
	Define Global Data	
	Configure Network	
	Simulate Modules	
	Configure Process Diagnostics	
	CAx Data	>
	Block Privacy	
	S7-Web2PLC	
	Set PG/PC Interface	

- 2. As access point for the application you set the network card with TCP/IP protocol via which the PC is connected to the S7-300 CPU and via which you access the S7-300 CPU.
- 3. Apply the settings with "OK".

Set PG/PC Interface		×					
Access Path LLDP / DCP PNIO Adapter Info							
Access Point of the Application: S7ONLINE (STEP 7)> vmxnet3 Ethemet (Standard for STEP 7)	Adapter.TCPIP.A	uto.1 💌					
Interface Parameter Assignment Used:							
vmxnet3 Ethemet Adapter.TCPIP.Auto.1 <	Properties	s					
wmxnet3 Ethernet Adapter.ISO.2	Diagnostic	xs					
vmxnet3 Ethernet Adapter. TCPIP.1	Сору						
Vmxnet3 Ethemet Adapter.TCPIP.A	2 Delete						
(Parameter assigment for the IE-PG access to your NDIS CPs with TCP/IP Protocol (RFC-1006))							
OK	Cancel	Help					

- 4. In the SIMATIC Manager you mark the S7-300 station.
- 5. Click the "Load" button in the toolbar. The hardware configuration and the user program are downloaded to the S7-300 CPU.



# 5 Error Handling

# 5.1 FB "PutData"

If an error occurs in FB "PutData", the cause of the error is written to the "status" output parameter.

Table 5-1

error	status	Description	Remedy	
0	16#7000	FB "PutData" is not active.	Status information Enable FB "PutData" by setting the "execute" input to "true".	
0	16#7001	FB "PutData" is initialized.	Status information	
0	16#7002	Write job is running.	Status information	
1	<>16#0000	Status display of the "PUT" instruction.	<ul> <li>Detailed information is available in the STEP 7 Online Help or in the following manuals:</li> <li><u>"SIMATIC STEP 7</u> <u>Basic/Professional V16 and SIMATIC WinCC V16"</u>.</li> <li><u>SIMATIC System Software for S7-300/400 System and Standard Functions - Volume 1/2</u></li> </ul>	

## 5.2 FB "GetData"

If an error occurs in FB "GetData", the cause of the error is written to the "status" output parameter.

error	status	Description	Remedy	
0	16#7000	FB "GetData" is not active.	Status information Enable FB "GetData" by setting the "execute" input to "true".	
0	16#7001	FB "GetData" is initialized.	Status information	
0	16#7002	Read job is running.	Status information	
1	<>16#0000	Status display of the "GET" instruction.	<ul> <li>Detailed information is available in the STEP 7 Online Help or in the following manuals:</li> <li><u>"SIMATIC STEP 7</u> <u>Basic/Professional V16 and</u> <u>SIMATIC WinCC V16"</u>.</li> <li><u>SIMATIC System Software for S7-300/400 System and Standard</u> <u>Functions - Volume 1/2</u></li> </ul>	

Tabl	e 5	-2
I abr	50	~~

# 6 Operating and Monitoring

Operating and monitoring is done via a variable table in STEP 7 V5.6 and STEP 7 V16.

- Client":
  - The variable table is for starting the Read job and for monitoring the data that the S7 CPU as client reads from the server.
  - The variable table is for starting the Write job and for monitoring the data that the S7 CPU writes to the server.
- "Server":
  - The variable table is for monitoring the data that the S7 CPU as server receives from the client.
  - The variable table is for controlling the data that the client reads from the S7 CPU.

### 6.1 SIMATIC S7-1200 Writes and Reads Data as Client

### 6.1.1 Write Data to the S7-300

- 1. In STEP 7 V16 you open the device folder of the S7-1200 CPU.
- 2. In the project tree you open the device folder of the S7-1200 CPU.
- 3. In the "Watch and force tables" folder you double-click the variable table "Client".

Project tree
Devices
<ul> <li>S7communication_S7-1200</li> </ul>
📫 Add new device
🚠 Devices & networks
▼ 1 PLC_1 [CPU 1214C DC/DC/DC]
🕎 Device configuration 🖉 🎽
🛂 Online & diagnostics
🕨 🔙 Program blocks
🕨 🙀 Technology objects
External source files
🕨 🔚 PLC tags
PLC data types
🔻 詞 Watch and force tables
Add new watch table
🔛 Client
Force table
Server

The variable table opens in the working area of STEP 7 V16.

- Click the "Monitor all" button in the toolbar. The "Monitor value" column displays the current values stored in the Send data area (DB1) and Receive data area (DB2).
- 5. In the "Modify value" column you enter the values that are to be put in the Send data area (DB2) of the S7-1200 CPU and written to the S7300 CPU.
- 6. Set the "GeneralData" variable to the value "True" to start the Write job. Note

The variable "GeneralData".put.execute is reset to the value "False" in the user program as soon as the Write job is completed with "done" = true or "error" = true.

7. In the toolbar you click the "Modify all enabled values once and immediately" button.

<b>\$7</b> 0	S7communication_S7-12							
<b>1</b>	<b>1</b>	* 1 <b>9</b> 10 <b>9</b> 70 7						
_	1	Name 4	Address	Display format	Monitor value	Modify value		
1	// Send	Data (						
2		"GeneralData".put.execute		Bool	FALSE	TRUE		
3		"SendDataClient".data[0]	%DB2.DBB0	DEC+/-	15 AT	15		
4		"SendDataClient".data[1]	%DB2.DBB1	DEC+/-	16 AI	16		
5		"SendDataClient".data[2]	%DB2.DBB2	DEC+/-	17	17		
6		"SendDataClient".data[3]	%DB2.DBB3	DEC+/-	18	18		
7		"SendDataClient".data[4]	%DB2.DBB4	DEC+/-	19	19		
8		"SendDataClient".data[5]	%DB2.DBB5	DEC+/-	20	20		
9		"SendDataClient".data[6]	%DB2.DBB6	DEC+/-	21	21		
10		"SendDataClient".data[7]	%DB2.DBB7	DEC+/-	22 9	22		
11		"SendDataClient".data[8]	%DB2.DBB8	DEC+/-	23 (AL)	23		
12		"SendDataClient".data[9]	%DB2.DBB9	DEC+/-	24	24		
13		"SendDataClient".data[99]	%DB2.DBB99	DEC+/-	99	99		
14		"GeneralData".diagnostic.statusPut		Hex	16#0000	16#0000		
15	// Recei	ve Data						
16		"GeneralData".get.execute		Bool	FALSE	TRUE		
17		"RecvDataClient".data[0]	%DB1.DBB0	DEC+/-	17			
18		"RecvDataClient".data[1]	%DB1.DBB1	DEC+/-	18			
19		"RecvDataClient".data[2]	%DB1.DBB2	DEC+/-	19			
20		"RecvDataClient".data[3]	%DB1.DBB3	DEC+/-	20			
21		"RecvDataClient".data[4]	%DB1.DBB4	DEC+/-	21			
22		"RecvDataClient".data[5]	%DB1.DBB5	DEC+/-	22			
23		"RecvDataClient".data[6]	%DB1.DBB6	DEC+/-	23			
24		"RecvDataClient".data[7]	%DB1.DBB7	DEC+/-	24			
25		"RecvDataClient".data[8]	%DB1.DBB8	DEC+/-	25			
26		"RecvDataClient".data[9]	%DB1.DBB9	DEC+/-	32			
27		"RecvDataClient".data[99]	%DB1.DBB99	DEC+/-	33			
28		"GeneralData".diagnostic.statusGet		Hex	16#0000			
29			<add new=""></add>					

### Result

The modified values are put in the Send data area (DB1) of the S7-1200 CPU and written to the S7-300 CPU.

### 6.1.2 Read Data from the S7-300

- 1. In STEP 7 V16 you open the S7-1200 CPU project.
- 2. In the project tree you open the device folder of the S7-1200 CPU.
- 3. In the "Watch and force tables" folder you double-click the variable table "Client".

The variable table opens in the working area of STEP 7 V16.

Project tree
Devices
11 International
<ul> <li>S7communication_S7-300</li> </ul>
💣 Add new device
Devices & networks
PLC_3 [CPU 315-2 PN/DP]
🛛 Device configuration
🖫 Online & diagnostics
🕨 🚘 Program blocks
🕨 🚂 Technology objects
External source files
🕨 🚂 PLC tags
PLC data types
🗢 🥅 Watch and force tables
Add new watch table
Client 3
🔛 Forcetal 🌔
Server

- 4. Click the "Monitor all" button in the toolbar. The "Status value" column displays the current values stored in the Send data area (DB2) and Receive data area (DB1).
- 5. Set the "GeneralData" variable to the value "True" to start the Read job. **Note**

The variable "GeneralData".get.execute is reset to the value "False" in the user program as soon as the Read job is completed with "done" = true or "error" = true.

6. In the toolbar you click the "Modify all enabled values once and immediately" button.

S7communication_S7-12						
🥐 u	ž 😼 🇓 👂 🧞 🎘 📭 📭					
i	Name	Address	Display format	Monitor value	Modify value	
// Send	Data ( 🍋 )					
	"GeneralData".put.execute		Bool	FALSE	TRUE	
	"SendDataClient".data[0]	%DB2.DBB0	DEC+/-	15	15	
	"SendDataClient".data[1]	%DB2.DBB1	DEC+/-	16	16	
	"SendDataClient".data[2]	%DB2.DBB2	DEC+/-	17	17	
	"SendDataClient".data[3]	%DB2.DBB3	DEC+/-	18	18	
	"SendDataClient".data[4]	%DB2.DBB4	DEC+/-	19	19	
	"SendDataClient".data[5]	%DB2.DBB5	DEC+/-	20	20	
	"SendDataClient".data[6]	%DB2.DBB6	DEC+/-	21	21	
	"SendDataClient".data[7]	%DB2.DBB7	DEC+/-	22	22	
	"SendDataClient".data[8]	%DB2.DBB8	DEC+/-	23	23	
	"SendDataClient".data[9]	%DB2.DBB9	DEC+/-	24	24	
	"SendDataClient".data[99]	%DB2.DBB99	DEC+/-	99	99	
	"GeneralData".diagnostic.statusPut		Hex	16#0000	16#0000	
// Recei	ve Data					
	"GeneralData".get.execute		Bool	FALSE	TRUE	
	"RecvDataClient".data[0]	%DB1.DBB0	DEC+/-	17		
	"RecvDataClient".data[1]	%DB1.DBB1	DEC+/-	18		
	"RecvDataClient".data[2]	%DB1.DBB2	DEC+/-	19		
	"RecvDataClient".data[3]	%DB1.DBB3	DEC+/-	20		
	"RecvDataClient".data[4]	%DB1.DBB4	DEC+/-	21		
	"RecvDataClient".data[5]	%DB1.DBB5	DEC+/-	22		
	"RecvDataClient".data[6]	%DB1.DBB6	DEC+/-	23		
	"RecvDataClient".data[7]	%DB1.DBB7	DEC+/-	24		
	"RecvDataClient".data[8]	%DB1.DBB8	DEC+/-	25		
	"RecvDataClient".data[9]	%DB1.DBB9	DEC+/-	32		
	"RecvDataClient".data[99]	%DB1.DBB99	DEC+/-	33		
	"GeneralData".diagnostic.statusGet		Hex	16#0000		
	-					
	Commu	communication_S7-12 PLC_1 [CPU Name // Send Data "GeneralData".put.execute "SendDataClient".data[0] "SendDataClient".data[1] "SendDataClient".data[2] "SendDataClient".data[3] "SendDataClient".data[3] "SendDataClient".data[5] "SendDataClient".data[6] "SendDataClient".data[8] "SendDataClient".data[9] "SendDataClient".data[9] "GeneralData".get.execute "RecvDataClient".data[1] "RecvDataClient".data[1] "RecvDataClient".data[2] "RecvDataClient".data[3] "RecvDataClient".data[1] "RecvDataClient".data[3] "RecvDataClient".data[4] "RecvDataClient".data[4] "RecvDataClient".data[5] "RecvDataClient".data[4] "RecvDataClient".data[4] "RecvDataClient".data[5] "RecvDataClient".data[6] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9] "RecvDataClient".data[9]	Communication_S7-12 PLC_1 [CPU 1214C DC/DC Address // Send Data GeneralData".put.execute SendDataClient".data[0] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[2] SendDataClient".data[3] SendDataClient".data[4] SendDataClient".data[5] SendDataClient".data[5] SendDataClient".data[6] SendDataClient".data[7] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[1] SendDataClient".data[2] SendDataClient".data[3] SendDataClient".data[4] SendDataClient".data[4] SendDataClient".data[5] SendDataClient".data[4] SendDataClient".data[5] SendDataClient".data[6] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] SendDataClient".data[9] Se	Communication_S7-12       UC_1 [CPU 1214C DC/DC/DC] > Watch         Image: Communication_S7-12       Umage: Communication S7-12       Umage: Communication S7-12         Image: Communication S7-12       Umage: Communication S7-12       Umage: Communication S7-12       Variable S7-12         Image: Communication S7-12       Umage: Communication S7-12       Umage: Communication S7-12       Umage: Communication S7-12       Variable S7-12       Variable S7-12       Variable S7-12         Image: Communication S7-12       Variable S7-12       Variable S7-12       Variable S7-12       Variable S7-12       Variable S7-12         Image: Communication S7-12       Variable S7	Image       Image <thimage< th=""> <thimage< th=""> <thi< td=""></thi<></thimage<></thimage<>	

### Result

The "Monitor value" column displays the data read from the S7-300 and stored in the Receive data area (DB1).

# 6.2 SIMATIC S7-300 Writes and Reads Data as Client (STEP 7 V16)

### 6.2.1 Write Data to the S7-1200

- 1. In STEP 7 V16 you open the S7-300 CPU project.
- 2. In the project tree you open the device folder of the S7-300 CPU.
- 3. In the "Watch and force tables" folder you double-click the variable table "Client".

The variable table opens in the working area of STEP 7 V16.

Project tree					
Devices					
<ul> <li>S7communication_S7-300</li> </ul>					
📫 Add new device					
🛗 Devices & networks					
▼ 📺 PLC_3 [CPU 315-2 PN/DP]					
🛛 Device configuration					
🖫 Online & diagnostics					
🕨 🚘 Program blocks					
🕨 🚂 Technology objects					
External source files					
🕨 🚂 PLC tags					
PLC data types					
🔻 🥅 Watch and force tables					
Add new watch table					
Client 3					
🔚 Forcetab ( 🏹 )					
🔛 Server					

- 4. Click the "Monitor all" button in the toolbar. The "Monitor value" column displays the current values stored in the Send data area (DB1) and Receive data area (DB2).
- 5. In the "Modify value" column you enter the values that are to be put in the Send data area (DB2) of the S7-300 CPU and written to the S7-1200 CPU.
- 6. Set the "GeneralData" variable to the value "True" to start the Write job. **Note**

The variable "GeneralData".put.execute is reset to the value "False" in the user program as soon as the Write job is completed with "done" = true or "error" = true.

7. In the toolbar you click the "Modify all enabled values once and immediately" button.

### 6 Operating and Monitoring

S7communication_S7-30C_3 [CPU 315-2 PN/DP] > Watch and force tables > Client								
<b>*</b>	🥐 d	1 1/ 10 1 70 27 PP PP						
_	i	Name	4 ddress	Display format	Monitor value	Modify value		
1	// send	data ( 🍋						
2		"GeneralData".put.execute	%DB10.DBX0.0	Bool	FALSE	TRUE		
3		"SendDataClient".data[0]	%DB2.DBB0	DEC+/-	17 6	17		
4		"SendDataClient".data[1]	%DB2.DBB1	DEC+/-	18 ( AĬ )	18		
5		"SendDataClient".data[2]	%DB2.DBB2	DEC+/-	19	19		
6		"SendDataClient".data[3]	%DB2.DBB3	DEC+/-	20	20		
7		"SendDataClient".data[4]	%DB2.DBB4	DEC+/-	21	21		
8		"SendDataClient".data[5]	%DB2.DBB5	DEC+/-	33	33		
9		"SendDataClient".data[6]	%DB2.DBB6	DEC+/-	34	34		
10		"SendDataClient".data[7]	%DB2.DBB7	DEC+/-	35 6	35		
11		"SendDataClient".data[8]	%DB2.DBB8	DEC+/-	36 Aĭ	36		
12		"SendDataClient".data[9]	%DB2.DBB9	DEC+/-	37	37		
13		"SendDataClient".data[99]	%DB2.DBB99	DEC+/-	49	49		
14		"GeneralData".diagnostic.statusPut	%DB10.DBW16	Hex	16#0000			
15								
16	// receiv	ve data						
17		"GeneralData".get.execute	%DB10.DBX8.0	Bool	FALSE	TRUE		
18		"RecvDataClient".data[0]	%DB1.DBB0	DEC+/-	26			
19		"RecvDataClient".data[1]	%DB1.DBB1	DEC+/-	9			
20		"RecvDataClient".data[2]	%DB1.DBB2	DEC+/-	19			
21		"RecvDataClient".data[3]	%DB1.DBB3	DEC+/-	81			
22		"RecvDataClient".data[4]	%DB1.DBB4	DEC+/-	24			
23		"RecvDataClient".data[5]	%DB1.DBB5	DEC+/-	10			
24		"RecvDataClient".data[6]	%DB1.DBB6	DEC+/-	20			
25		"RecvDataClient".data[7]	%DB1.DBB7	DEC+/-	18			
26		"RecvDataClient".data[8]	%DB1.DBB8	DEC+/-	3			
27		"RecvDataClient".data[9]	%DB1.DBB9	DEC+/-	11			
28		"RecvDataClient".data[99]	%DB1.DBB99	DEC+/-	60			
29		"GeneralData".diagnostic.statusGet	%DB10.DBW18	Hex	16#0000			
30			<add new=""></add>					

### Result

The modified values are put in the Send data area (DB1) of the S7-300 CPU and written to the S7-1200 CPU.

### 6.2.2 Read Data from the S7-1200

- 1. In STEP 7 V16 you open the S7-300 CPU project.
- 2. In the project tree you open the device folder of the S7-300 CPU.
- 3. In the "Watch and force tables" folder you double-click the variable table "Client".

The variable table opens in the working area of STEP 7 V16.

Project tree					
Devices					
Ĩ					
<ul> <li>S7communication_S7-300</li> </ul>					
📑 Add new device					
Devices & networks					
▼ 📺 PLC_3 [CPU 315-2 PN/DP]					
🛐 Device configuration					
🖳 Online & diagnostics					
🕨 📴 Program blocks					
🕨 🚂 Technology objects					
🕨 🔙 External source files					
🕨 📜 PLC tags					
PLC data types					
🔻 🥅 Watch and force tables					
🗳 Add new watch table					
Client Client					
Forcetab 🖌					
Server Server					

- 4. Click the "Monitor all" button in the toolbar. The "Status value" column displays the current values stored in the Send data area (DB2) and Receive data area (DB1).
- 5. Set the "GeneralData" variable to the value "True" to start the Read job. **Note**

The variable "GeneralData".get.execute is reset to the value "False" in the user program as soon as the Read job is completed with "done" = true or "error" = true.

6. In the toolbar you click the "Modify all enabled values once and immediately" button.

### 6 Operating and Monitoring

S7communication_S7-30								
	i	Name	4 ddress	Display format	Monitor value	Modify value		
1	// send	data (		·				
2		"GeneralData".put.execute	%DB10.DBX0.0	Bool	FALSE	TRUE		
3		"SendDataClient".data[0]	%DB2.DBB0	DEC+/-	17	17		
4		"SendDataClient".data[1]	%DB2.DBB1	DEC+/-	18	18		
5		"SendDataClient".data[2]	%DB2.DBB2	DEC+/-	19	19		
6		"SendDataClient".data[3]	%DB2.DBB3	DEC+/-	20	20		
7		"SendDataClient".data[4]	%DB2.DBB4	DEC+/-	21	21		
8		"SendDataClient".data[5]	%DB2.DBB5	DEC+/-	33	33		
9		"SendDataClient".data[6]	%DB2.DBB6	DEC+/-	34	34		
10		"SendDataClient".data[7]	%DB2.DBB7	DEC+/-	35	35		
11		"SendDataClient".data[8]	%DB2.DBB8	DEC+/-	36	36		
12		"SendDataClient".data[9]	%DB2.DBB9	DEC+/-	37	37		
13		"SendDataClient".data[99]	%DB2.DBB99	DEC+/-	49	49		
14		"GeneralData".diagnostic.statusPut	%DB10.DBW16	Hex	16#0000			
15								
16	// receiv	e data						
17		"GeneralData".get.execute	%DB10.DBX8.0	Bool	FALSE	TRUE		
18		"RecvDataClient".data[0]	%DB1.DBB0	DEC+/-	26			
19		"RecvDataClient".data[1]	%DB1.DBB1	DEC+/-	9			
20		"RecvDataClient".data[2]	%DB1.DBB2	DEC+/-	19	<b>A</b> I)		
21		"RecvDataClient".data[3]	%DB1.DBB3	DEC+/-	81			
22		"RecvDataClient".data[4]	%DB1.DBB4	DEC+/-	24			
23		"RecvDataClient".data[5]	%DB1.DBB5	DEC+/-	10			
24		"RecvDataClient".data[6]	%DB1.DBB6	DEC+/-	20			
25		"RecvDataClient".data[7]	%DB1.DBB7	DEC+/-	18			
26		"RecvDataClient".data[8]	%DB1.DBB8	DEC+/-	3			
27		"RecvDataClient".data[9]	%DB1.DBB9	DEC+/-	11			
28		"RecvDataClient".data[99]	%DB1.DBB99	DEC+/-	60			
29		"GeneralData".diagnostic.statusGet	%DB10.DBW18	Hex	16#0000			
30			<add new=""></add>					

### Result

The "Monitor value" column displays the data read from the S7-300 and stored in the Receive data area (DB1).

# 6.3 SIMATIC S7-300 Writes and Reads Data as Client (STEP 7 V5.6)

### 6.3.1 Write Data to the S7-1200 CPU

- 1. In STEP 7 V5.6 you open the S7-300 CPU project.
- 2. In the "Blocks" folder you double-click the variable table "Client". The variable table opens.



- 3. Click the "Monitor variable" button in the toolbar. The "Status value" column displays the current values stored in the Send data area (DB2) and Receive data area (DB1).
- 4. In the "Modify value" column you enter the values that are to be put in the Send data area (DB2) of the S7-300 CPU and written to the S7-1200 CPU.
- 5. Set the "GeneralData" variable to the value "True" to start the Write job. **Note**

The variable "GeneralData".put.execute is reset to the value "False" in the user program as soon as the Write job is completed with "done" = true or "error" = true.

6. Click the "Activate modify values" button in the toolbar.

42 14-	r Client							
Ring var - Client								
Tabelle Bearbeiten Einfügen Zielsystem Variable Ansicht Extras Fenster Hilfe								
-Fai	D 🖻 🔒 🎒		<b>≗ </b> <u></u> <u></u>	Sy Gr	47 60 Y 140			
8	Client @PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program_ONLINE							
	Operand	Symbol	Anzeigeformat	Statuswort	Steuerwert			
1	DB10.DBX 6.0	"GeneralData".put.execute	BOOL	false	true			
2	//Send data		<u>-</u>					
3	DB2.DBB 0	"SendDataClient".data[0]	DEZ	26	26			
4	DB2.DBB 1	"SendDataClient".data[1]	DEZ	9	9			
5	DB2.DBB 2	"SendDataClient".data[2]	DEZ	19	19			
6	DB2.DBB 3	"SendDataClient".data[3]	DEZ	81	81			
7	DB2.DBB 4	"SendDataClient".data[4]	DEZ	14	16			
8	DB2.DBB 5	"SendDataClient".data[5]	DEZ	2 ( AĬ )	2			
9	DB2.DBB 6	"SendDataClient".data[6]	DEZ	20	20			
10	DB2.DBB 7	"SendDataClient".data[7]	DEZ	13	13			
11	DB2.DBB 8	"SendDataClient".data[8]	DEZ	3	3			
12	DB2.DBB 9	"SendDataClient".data[9]	DEZ	10	10			
13	DB2.DBB 99	"SendDataClient".data[99]	DEZ	1	1			
14	DB10.DBW 12	"GeneralData".diagnostic.statusPut	HEX	W#16#0000				
15								
16	DB10.DBX 0.0	"GeneralData".get.execute	BOOL	false	true			
17	//Receive Data							
18	DB1.DBB 0	"RecvDataClient".data[0]	DEZ	11				
19	DB1.DBB 1	"RecvDataClient".data[1]	DEZ	12				
20	DB1.DBB 2	"RecvDataClient".data[2]	DEZ	13				
21	DB1.DBB 3	"RecvDataClient".data[3]	DEZ	14				
22	DB1.DBB 4	"RecvDataClient".data[4]	DEZ	15				
23	DB1.DBB 5	"RecvDataClient".data[5]	DEZ	16				
24	DB1.DBB 6	"RecvDataClient".data[6]	DEZ	17				
25	DB1.DBB 7	"RecvDataClient".data[7]	DEZ	18				
26	DB1.DBB 8	"RecvDataClient".data[8]	DEZ	19				
27	DB1.DBB 9	"RecvDataClient".data[9]	DEZ	20				
28	DB1.DBB 99	"RecvDataClient".data[99]	DEZ	21				
29	DB10.DBW 14	"GeneralData".diagnostic.statusGet	HEX	W#16#0000				
30								

### Result

The modified values are put in the Send data area (DB1) of the S7-300 CPU and written to the S7-1200 CPU.

### 6.3.2 Read Data from the S7-1200 CPU

- 1. In STEP 7 V5.6 you open the S7-300 CPU project.
- 2. In the "Blocks" folder you double-click the variable table "Client". The variable table opens.



- 3. Click the "Monitor variable" button in the toolbar. The "Status value" column displays the current values stored in the Send data area (DB2) and Receive data area (DB1).
- 4. Set the "GeneralData" variable to the value "True" to start the Read job. **Note**

The variable "GeneralData".get.execute is reset to the value "False" in the user program as soon as the Read job is completed with "done" = true or "error" = true.

Tabelle       Bearbeiten       Linfügen       Zielsystem       Variable       Anscht       Extras       Fenster       Hiffe         Image: Status       Image: Status <th colspan="7">🕍 Var - Client</th>	🕍 Var - Client						
Image: Second Symbol         Anzeigeformat         Status         Steve           Image: Symbol         Steve         BOOL         faise         True           Image: Symbol         Steve         Steve         BOOL         faise         Steve           Image: Symbol Biol Symbol         Steve         Steve         BOOL         Imagee: Steve <t< td=""><td colspan="7">Tabelle Bearbeiten Einfügen Zielsystem Variable Ansicht Extras Fenster Hilfe</td></t<>	Tabelle Bearbeiten Einfügen Zielsystem Variable Ansicht Extras Fenster Hilfe						
Client @PUT_GET\SIMATIC 315\CPU 315-2 PN/DP\S7 program ONLINE       Status       Status         1       DB10.DBX 6.0       'GeneraData".put execute       BOOL       faise       true         2       //Send data	▱▯▰▯◓▯฿฿๚฿ฅ๛๛Ⴟฃ๚๚๛						
Control         Symbol         Anzeigeformat         Status         Steu           1         DB10.DBX         6.0         "GeneralData", put.execute         BOOL         faise         true           3         DB2.DBB         0         "SendDataClient".data[0]         DEZ         26         26           4         DB2.DBB         1         "SendDataClient".data[1]         DEZ         9         9           5         DB2.DBB         2         "SendDataClient".data[2]         DEZ         19         19           6         DB2.DBB         3         "SendDataClient".data[3]         DEZ         81         81           7         DB2.DBB         3         "SendDataClient".data[4]         DEZ         20         20           9         DB2.DBB         5         "SendDataClient".data[5]         DEZ         2         2           9         DB2.DBB         6         "SendDataClient".data[6]         DEZ         10         10           10         DB2.DBB         8         "SendDataClient".data[9]         DEZ         1         1           11         DB2.DBB         8         "SendDataClient".data[9]         DEZ         1         1           13         DB2	14 <u>8 c</u>	lient @PUT_GF	T\SIMATIC 315\CDU 315-2 PN/DD\S7	program ONUN	3		
Image         Operand         Symbol         All zergetormal         Status         Status         Status           1         DB10.DBX         6.0         "GeneralData".put.execute         BOOL         faise         true           2         ///Send data		l Oceaned					
1       DB10.0BX 0.0       GeneralData (Jule XeCure       DOL       Iake       Ittle         2       ///Send data		Operand DB10 DBV 6.0	"ConorralData" put execute	Anzeigerormat	Status	Steus	
2       Indent data         3       DB2.DBB 0       "SendDataClient".data[0]       DEZ       28       26         4       DB2.DBB 1       "SendDataClient".data[1]       DEZ       9       9         5       DB2.DBB 2       "SendDataClient".data[3]       DEZ       19       19         6       DB2.DBB 3       "SendDataClient".data[3]       DEZ       81       81         7       DB2.DBB 4       "SendDataClient".data[6]       DEZ       2       2         9       DB2.DBB 5       "SendDataClient".data[6]       DEZ       20       20         10       DB2.DBB 7       "SendDataClient".data[7]       DEZ       13       13         11       DB2.DBB 9       "SendDataClient".data[9]       DEZ       10       10         13       DB2.DBB 9       "SendDataClient".data[9]       DEZ       1       1         14       DB10.DBX       0.0       "GeneralData".data[9]       DEZ       11       1         14       DB1.DBB 1       "RecvDataClient".data[1]       DEZ       13       2       1       1         19       DB1.DBB 2       "RecvDataClient".data[1]       DEZ       11       I       I       I       I       I<		UBTU.DBA 0.0	GeneralData .put.execute	BUUL	laise	uue	
3       DB2.DB0       0       DB4.DB1       "SendDataClient".data[1]       DEZ       9       9         5       DB2.DB8       2       "SendDataClient".data[2]       DEZ       19       19         6       DB2.DB8       3       "SendDataClient".data[2]       DEZ       81       81         7       DB2.DB8       3       "SendDataClient".data[3]       DEZ       81       81         7       DB2.DB8       4       "SendDataClient".data[4]       DEZ       16       16         8       DB2.DB8       5       "SendDataClient".data[5]       DEZ       2       2         9       DB2.DB8       6       "SendDataClient".data[6]       DEZ       13       13       13         11       DB2.DB8       9       "SendDataClient".data[9]       DEZ       1       1         13       DB2.DB8       9       "SendDataClient".data[9]       DEZ       1       1         14       DB1.DB8       0       "RecvDataClient".data[0]       DEZ       11       1         14       DB1.DB8       1       "RecvDataClient".data[1]       DEZ       11       I         19       DB1.DB8       1       "RecvDataClient".data[2]       D	3	DB2 DBB 0	"SendDataClient" data[0]	DEZ	26	26	
Image: Second	4	DB2 DB8 1	"SendDataClient" data[1]	DEZ DEZ	20 9	9	
0         DB2.DB8_1         "SendDataClient".data[3]         DEZ         81         81           6         DB2.DB8_3         "SendDataClient".data[4]         DEZ         16         16           7         DB2.DB8_4         "SendDataClient".data[5]         DEZ         2         2           9         DB2.DB8_6         "SendDataClient".data[6]         DEZ         20         20           10         DB2.DB8_6         "SendDataClient".data[7]         DEZ         13         13           11         DB2.DB8_8         "SendDataClient".data[9]         DEZ         10         10           13         DB2.DB8_9         "SendDataClient".data[9]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".diagnostic.statusPut         HEX         W#16#0000           15	5	DB2 DBB 2	"SendDataClient" data[2]	DEZ	19	19	
0       DBL.DB 0       0       DBL.DB 0       0         7       DB2.DB 4       "SendDataClient".data[4]       DEZ       16       16         8       DB2.DB 5       "SendDataClient".data[5]       DEZ       2       2         9       DB2.DB 6       "SendDataClient".data[6]       DEZ       20       20         10       DB2.DB 7       "SendDataClient".data[6]       DEZ       13       13         11       DB2.DB 8       "SendDataClient".data[9]       DEZ       10       10         13       DB2.DB 9       "SendDataClient".data[9]       DEZ       1       1         14       DB10.DBW 12       "GeneralData".get.execute       BOOL       false       true         17       //Receive Data       "RecvDataClient".data[0]       DEZ       11       AI         18       DB1.DB8       "RecvDataClient".data[1]       DEZ       13       AI         20       DB1.DB8       "RecvDataClient".data[2]       DEZ       14       AI         21       DB1.DB8       "RecvDataClient".data[2]       DEZ       11       AI         22       DB1.DB8       "RecvDataClient".data[2]       DEZ       13       AI         22       DB	6	DB2 DBB 3	"SendDataClient" data[3]	DEZ	81	81	
1       00120100       10       10       10       10         8       DB2.DB8       5       "SendDataClent".data[6]       DEZ       20       20         9       DB2.DB8       6       "SendDataClent".data[6]       DEZ       20       20         10       DB2.DB8       7       "SendDataClent".data[7]       DEZ       13       13         11       DB2.DB8       8       "SendDataClent".data[8]       DEZ       3       3         12       DB2.DB8       9       "SendDataClent".data[9]       DEZ       10       10         13       DB2.DB8       9       "SendDataClent".data[9]       DEZ       1       1         14       DB10.DBW       12       "GeneralData".get.execute       BOOL       false       true         15	7	DB2 DBB 4	"SendDataClient" data[4]	DEZ	16	16	
9         DB2.DB8         6         "SendDataClient".data[6]         DEZ         20         20           10         DB2.DB8         7         "SendDataClient".data[6]         DEZ         13         13           11         DB2.DB8         8         "SendDataClient".data[7]         DEZ         13         13           12         DB2.DB8         8         "SendDataClient".data[8]         DEZ         10         10           13         DB2.DB8         9         "SendDataClient".data[9]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".get.execute         BOOL         false         true           15	8	DB2 DB8 5	"SendDataClient" data[5]	DEZ	2	2	
10         DB2.DBB         7         "SendDataClient".data[7]         DEZ         13         13           11         DB2.DBB         8         "SendDataClient".data[8]         DEZ         3         3           12         DB2.DBB         8         "SendDataClient".data[9]         DEZ         10         10           13         DB2.DBB         9         "SendDataClient".data[9]         DEZ         10         10           13         DB2.DBB         9         "SendDataClient".data[9]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".get.execute         BOOL         false         true           16         DB10.DBX         0.0         "GeneralData".get.execute         BOOL         false         true           17         //Receive Data	9	DB2.DBB 6	"SendDataClient" data[6]	DEZ	20	20	
11         DB2.DB8         "SendDataClient".data[8]         DEZ         3         3           12         DB2.DB8         "SendDataClient".data[9]         DEZ         10         10           13         DB2.DB8         "SendDataClient".data[9]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".diagnostic.statusPut         HEX         W#16#0000           15	10	DB2.DBB 7	"SendDataClient" data[7]	DEZ	13	13	
12         DB2.DBB         9         "SendDataClient".data[9]         DEZ         10         10           13         DB2.DBB         99         "SendDataClient".data[99]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".diagnostic.statusPut         HEX         W#16#0000           15	11	DB2.DBB 8	"SendDataClient".data[8]	DEZ	3	3	
13         DB2.DBB         99         "SendDataClient".data[99]         DEZ         1         1           14         DB10.DBW         12         "GeneralData".diagnostic.statusPut         HEX         W#16#0000           15	12	DB2.DBB 9	"SendDataClient".data[9]	DEZ	10	10	
14         DB10.DBW         12         "GeneralData".diagnostic.statusPut         HEX         W#16#0000           15	13	DB2.DBB 99	"SendDataClient".data[99]	DEZ	1	1	
15	14	DB10.DBW 12	"GeneralData".diagnostic.statusPut	HEX	W#16#0000		
16         DB10.DBX         0.0         "GeneralData".get.execute         BOOL         false         true           17         //Receive Data	15						
17       //Receive Data         18       DB1.DBB 0       "RecvDataClient".data[0]       DEZ       11         19       DB1.DBB 1       "RecvDataClient".data[1]       DEZ       12         20       DB1.DBB 2       "RecvDataClient".data[2]       DEZ       13         21       DB1.DBB 3       "RecvDataClient".data[3]       DEZ       14         22       DB1.DBB 4       "RecvDataClient".data[4]       DEZ       15         23       DB1.DBB 5       "RecvDataClient".data[5]       DEZ       16         24       DB1.DBB 6       "RecvDataClient".data[6]       DEZ       17         25       DB1.DBB 7       "RecvDataClient".data[8]       DEZ       18         26       DB1.DBB 8       "RecvDataClient".data[9]       DEZ       19         27       DB1.DBB 9       "RecvDataClient".data[9]       DEZ       20	16	DB10.DBX 0.0	"GeneralData".get.execute	BOOL	false	true	
18         DB1.DBB         0         "RecvDataClient".data[0]         DEZ         11           19         DB1.DBB         1         "RecvDataClient".data[1]         DEZ         12           20         DB1.DBB         2         "RecvDataClient".data[2]         DEZ         13           21         DB1.DBB         3         "RecvDataClient".data[3]         DEZ         14           22         DB1.DBB         4         "RecvDataClient".data[4]         DEZ         15           23         DB1.DBB         5         "RecvDataClient".data[5]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	17	//Receive Data	1	. <u>.</u>			
19         DB1.DBB         1         "RecvDataClient".data[1]         DEZ         12           20         DB1.DBB         2         "RecvDataClient".data[2]         DEZ         13           21         DB1.DBB         3         "RecvDataClient".data[3]         DEZ         14           22         DB1.DBB         4         "RecvDataClient".data[4]         DEZ         15           23         DB1.DBB         5         "RecvDataClient".data[6]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         7         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	18	DB1.DBB 0	"RecvDataClient".data[0]	DEZ	11		
20         DB1.DBB         2         "RecvDataClient".data[2]         DEZ         13           21         DB1.DBB         3         "RecvDataClient".data[3]         DEZ         14           22         DB1.DBB         4         "RecvDataClient".data[4]         DEZ         15           23         DB1.DBB         5         "RecvDataClient".data[5]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[6]         DEZ         17           25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[9]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	19	DB1.DBB 1	"RecvDataClient".data[1]	DEZ	12		
21         DB1.DBB         3         "RecvDataClient".data[3]         DEZ         14           22         DB1.DBB         4         "RecvDataClient".data[4]         DEZ         15           23         DB1.DBB         5         "RecvDataClient".data[5]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[6]         DEZ         17           25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[9]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	20	DB1.DBB 2	"RecvDataClient".data[2]	DEZ	13		
22         DB1.DBB         4         "RecvDataClient".data[4]         DEZ         15           23         DB1.DBB         5         "RecvDataClient".data[5]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[6]         DEZ         17           25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	21	DB1.DBB 3	"RecvDataClient".data[3]	DEZ	14		
23         DB1.DBB         5         "RecvDataClient".data[5]         DEZ         16           24         DB1.DBB         6         "RecvDataClient".data[6]         DEZ         17           25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	22	DB1.DBB 4	"RecvDataClient".data[4]	DEZ	15		
24         DB1.DBB         6         "RecvDataClient".data[6]         DEZ         17           25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	23	DB1.DBB 5	"RecvDataClient".data[5]	DEZ	16		
25         DB1.DBB         7         "RecvDataClient".data[7]         DEZ         18           26         DB1.DBB         8         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	24	DB1.DBB 6	"RecvDataClient".data[6]	DEZ	17		
26         DB1.DBB         "RecvDataClient".data[8]         DEZ         19           27         DB1.DBB         "RecvDataClient".data[9]         DEZ         20	25	DB1.DBB 7	"RecvDataClient".data[7]	DEZ	18		
27         DB1.DBB         9         "RecvDataClient".data[9]         DEZ         20	26	DB1.DBB 8	"RecvDataClient".data[8]	DEZ	19		
	27	DB1.DBB 9	"RecvDataClient".data[9]	DEZ	20		
28 DB1.DBB 99 "RecvDataClient".data[99] DEZ 21	28	DB1.DBB 99	"RecvDataClient".data[99]	DEZ	21		
29 DB10.DBW 14 "GeneralData".diagnostic.statusGet HEX W#16#0000	29	DB10.DBW 14	"GeneralData".diagnostic.statusGet	HEX	W#16#0000		
30	30						

5. Click the "Activate modify values" button in the toolbar.

### Result

The "Status value" column displays the data read from the S7-1200 and stored in the Receive data area (DB1).

S7 communication between SIMATIC S7-1200 and SIMATIC S7-300 with PUT/GET Entry-ID: 82212115, V 3.0, 11/2020