SINAMICS V: Speed Control of a V20 with S7-1500 and ET 200SP via USS® Protocol, with HMI connection

SINAMICS V20, SIMATIC S7-1500, SIMATIC ET 200SP, as of TIA-Portal V12 SP1 Update 4

https://support.industry.siemens.com/cs/ww/en/view/90468030

Siemens Industry Online Support

Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.
Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

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

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.
The Siemens terms of use (https://support.industry.siemens.com) shall also apply.

Security information

Siemens provides products and solutions with Industrial Security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.
Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.
For additional information on industrial security measures that may be implemented, please visit https://www.siemens.com/industrialsecurity.
Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: https://www.siemens.com/industrialsecurity.

Table of contents

Legal information **2**

1 Task **5**

1.1 Controlling a SINAMICS V20 (process data exchange) 6
1.2 Reading/writing parameters 7
1.3 HMI for convenient operating and monitoring 7

2 Components and Structure **8**

2.1 Hardware 8
2.1.1 Hardware components 8
2.1.2 Installation and wiring 9
2.1.3 Bus connection 10
2.2 Software 12
2.2.1 Standard software 12
2.2.2 Firmware version of the SINAMICS V20 12
2.2.3 User software and documentation 13

3 Commissioning **14**

3.1 Requirements 14
3.2 Instruction 14
Changing inverter parameters via the BOP 14
Step table for commissioning the application example 15

4 Operation **20**

4.1 Screen navigation 20
4.2 “Drive control” screen 22
Drive control 22
USS activation 22
4.3 “Read/write parameters” screen 23
Reading parameters 23
Writing parameters 24
4.4 “Actual data” screen 25
Reading a set of current status parameters 25
4.5 “Special features” screen 26
Reading out inverter functions 26
Writing inverter functions 26
Info buttons 27
4.6 “Error (USS_Port_Scan / USS_Drive_Control)” screen 27
Info buttons 28
4.7 “Error (USS_Read_Param / USS_Write_Param)” screen 28
Info buttons 28

5 Programming **29**

5.1 Hardware configuration 29
5.2 Relevant code blocks 32
USS instructions 32
User function blocks 33
5.3 Program structure 34
Calling the V20_USS_Control_1 [FB1] in two OBs 34
Supplying the formal parameters of the V20_USS_Control [FB1] 35
Framework program (Main [OB1] and HMI_DB_1 [DB10]) 35
5.4 V20_USS_Control [FB1] function block 36
5.4.1 Parameterization 36
5.5 V20_USS_Param [FB2] function block 38
5.5.1 Operating modes 38
ACTIVATE_USS 38

RW_PARAM ... 39
ACTUAL_STATE ... 40
SET_FUNCTIONS ... 40
5.5.2 Parameterization ... 40
5.5.3 Start of a function ... 45

6 Time Behavior ... 46

6.1 Call intervals of the USS system function blocks ... 46
USS_Port_Scan calls ... 46
Calls USS_Drive_Control, USS_Read_Param, USS_Write_Param ... 46
6.2 Measuring the communication time ... 47
6.3 Frame time out of the SINAMICS V20 ... 47

7 Expansion to Several Drives ... 49

7.1 Expansion to up to 16 drives per port ... 49
7.2 Expansion to several ports ... 49
7.2.1 Expansion of hardware to several CM PtP per ET 200SP ... 49
7.2.2 Expansion of the hardware to several ET 200SP stations ... 50
7.2.3 Expansion of the user program to several ports ... 50

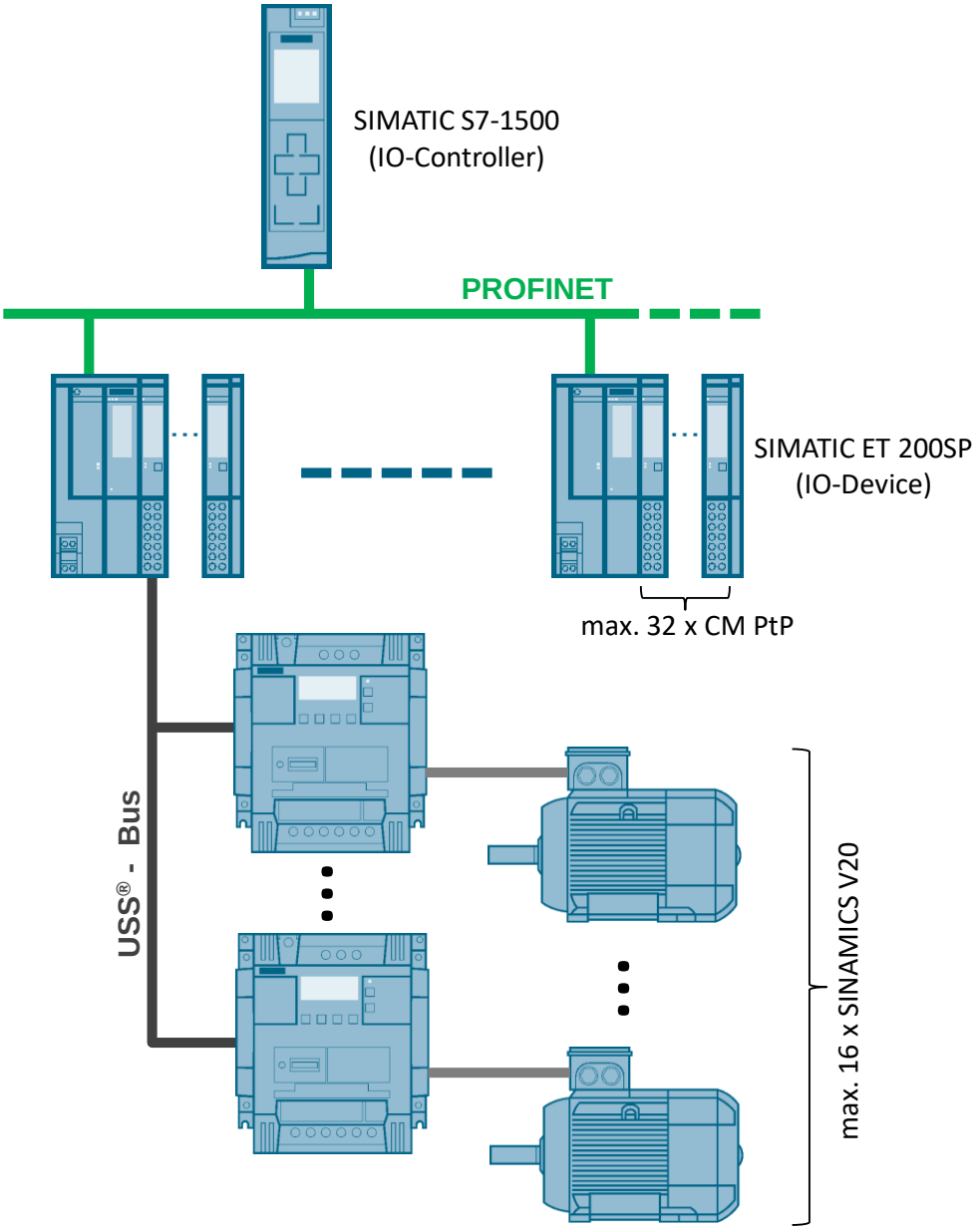
8 Literature ... 54

9 History ... 54

1 Task

SINAMICS V20 drives are to exchange data via the RS485 interface and via USS® (**U**niverselles **S**erielles **S**chnittstellenprotokoll – universal serial interface protocol) with a SIMATIC S7-1500 controller. The USS® bus is to be linked via a CM PtP communication module of a SIMATIC ET 200SP distributed I/O station (as PROFINET IO device) with the SIMATIC controller (as PROFINET IO controller). On a RS485 port or CM PtP up to 16 drives can be operated. A SIMATIC ET 200SP in turn, can operate a maximum of 32 communication modules.

Figure 1-1: Bus structure



It requires the functionality described in the sections 1.1 to 1.3.

1.1 Controlling a SINAMICS V20 (process data exchange)

- A SINAMICS V20 is to be controlled via the following input signals:[1]
  - **RUN [RUN]** – start bit of SINAMICS V20 (STW1, bit 0)
    If this parameter has the value TRUE, this input will enable the operation of the V20 at the preset speed.
  - **OFF2 [OFF2]** – coasting to a standstill (STW1, bit 1)
    If this parameter has the value FALSE, this bit will cause the SINAMICS V20 to coast to a standstill, without braking.
  - **OFF3 [OFF3]** – fast stop bit (STW1, bit 2)
    If this parameter has the value FALSE, this bit will cause a fast stop by braking the SINAMICS V20.
  - **F_ACK [F_ACK]** – error acknowledgement bit (STW1, bit 7)
    With this bit you reset the error bit of the SINAMICS V20 once you have removed the drive error. Thus, the V20 detects that the error no longer has to be reported.
  - **DIR [DIR]** – direction control of the SINAMICS V20 (STW1, bit 11) [2]
    This bit has to be set when the V20 is to run in forward direction (when $n_{setpoint}$ is positive).
  - **$n_{setpoint}$ [SPEED_SP]** – setpoint speed value
    This is the speed in percentages of the SINAMICS V20 to the configured frequency. When entering a positive value, the V20 will run forward (if DIR has the value TRUE).
- A SINAMICS V20 is to continuously transfer the following data to the controller:
  - **RUN_EN [RUN_EN]** – enable operation (ZSW1, bit 2)
    This bit reports whether the SINAMICS V20 is running.
  - **D_DIR [D_DIR]** – direction of drive (ZSW1, bit 14)
    This bit reports whether the SINAMICS V20 is running forward.
  - **INHIBIT [INHIBIT]** – SINAMICS V20 blocked
    (ZSW1, bit 6)
    This bit reports the status of the inhibit bit for the SINAMICS V20.
  - **FAULT [FAULT]**– drive error (ZSW1, bit 3)
    This bit reports whether an error occurred in the SINAMICS V20. The user has to remove the fault and set F_ACK (STW1, bit 7) in order to delete this bit.
  - **n_{actual} [SPEED]** – actual value drive speed (scaled value of PZD2)
    The date is the value of the current speed as a percentage to the configured speed.
- If there is a communication error, the error status is to be displayed.

[1] The signal names in square brackets correspond to the formal parameter names of the respective blocks.
[2] In STW1 the DIR direction bit is stored negated.

1.2 Reading/writing parameters

The required parameter accesses are summarized in four selectable modes:[3]

- **USS Activation ACTIVATE_USS** – enabling the USS communication channel in order to control a SINAMICS V20 via the USS communication.
- **Reading/writing parameters [RW_PARAM]** – reading and writing any inverter parameter by specifying the parameter number and the parameter index.
- **Displaying current data ACTUAL_STATE** – reading a set of selected status information from the inverter.
  - Actual value of the filtered output frequency [Hz] (r0024)
  - Actual value of the output voltage [V] (r0025)
  - Actual value of the smoothed link voltage [V] (r0026)
  - Actual value of the output current [A] (r0027)
  - Actual value of the total setpoint value [Hz] (r1078)
  - Actual value of the energy saving [kWh, currency, CO_2] (r0043[0..2])
  - Current connection macro (p0717)
  - Current application macro (p0717)
- **Selecting special functions [SET_FUNCTIONS]** – specifying a set of selected operating modes.
  - Continuous operation for DDS1…DDS3 (P0503[0..2])
  - Flying start (P1200)
  - Holding brake active (P1215)
  - Energy saving mode for DDS1…DDS3 (P2365[0..2])
  - Torque pulse mode for DDS1…DDS3 (P2365[0..2])

Note The functions provided to you here, are to support you in configuring your user software and in commissioning your inverter application. The inverter configuration itself is not subject of this application example.

1.3 HMI for convenient operating and monitoring

The controller project is to include a KTP600 operator panel (touch panel) for operating and monitoring which can also run as simulation on the development system (PG/PC). This makes very fast commissioning and demonstrating of the application example possible. The user can furthermore accept the operator panel configuration either fully or partly in own projects.

[3] The signal names in square brackets correspond to the formal parameter names of the appropriate blocks or the names of the respective data structures.

2 Components and Structure

2.1 Hardware

2.1.1 Hardware components

The application was created with the following components and assumes a configuration with one SINAMICS V20. If there are several inverters, the number of the affected components has to be adjusted.

Table 2-1: Hardware components[4]

Position	Component	No.	Article number	Note
	SIMATIC S7-300 Controller			
1	SIMATIC S7-1500 CPU 1511-1 PN	1	6ES7 511-1AK00-0AB0	Other S7-1500 CPU also possible (Application example created with firmware V1.1 and V1.5 [5])
2	SIMATIC S7 Memory Card 24MB	1	6ES7954-8LF02-0AA0	all Memory Cards 6ES7954-… possible
3	SIMATIC PM 1507 controlled power supply Input: 120/230 V AC output: DC 24 V/3 A	1	6EP1332-4BA00	for CPU, ET 200SP, KTP600; other suitable power supply also possible
4	SIMATIC S7-1500 160 mm DIN rail	1	6ES7590-1AB60-0AA0	All DIN rails 6ES7590-1… possible
	SIMATIC ET 200SP distributed I/O			
5	IM 155-6 PN ST incl. bus adapter 2xRJ45 (6ES7193-6AR00-0AA0)	1	6ES7155-6AA00-0BN0	ET 200SP Interface module (Application example created with firmware V1.1.)
6	Server module	1	-	Included in position 5
7	BASEUNIT BU15-P16+A0+2D	1	6ES7193-6BP00-0DA0	ET 200SP BaseUnit
8	CM PtP (Freeport, 3964R, USS, Modbus RTU)	1	6ES7137-6AA00-0BA0	ET 200SP communication module
9	SIMATIC standard 35 mm DIN rail, length 483 mm	1	6ES5710-8MA11	for ET 200SP, all 6ES5710-8MA.. DIN rails possible
	SINAMICS V20 drive and motor			
10	SINAMICS V20 1AC200-240V, 0.75kW	1	6SL3210-5BB17-5AV0	every other SINAMICS V20 is also possible
11	Asynchronous motor	1	1LA7083-4AA60	Example
	HMI			
12	SIMATIC Panel KTP600 Basic color PN	1	6AV6647-0AD11-3AX0	optional[6]

[4] Small parts such as wire and other installation material are not included in this table.
[5] See Table 2-3 regarding dependency between CPU firmware and required TIA version.
[6] Not required if you simulate the touch panel in TIA Portal.

Position	Component	No.	Article number	Note
	Other			
13	RS485 bus terminating network	1	6SL3255-0VC00-0HA0	Package content: 50 pieces.
14	Ethernet cable with 2 RJ45 plugs	3	6XV1850-2Hxxx xxx = E50 → 0.5 m = H10 → 1 m = H20 → 2m = H60 → 6m = N10 → 10m	S7-1500 ⇔ ET 200SP S7-1500 ⇔ PG/PC S7-1500 ⇔ KTP600[7]
15	Shield terminal/support	1	6ES7193-6SC00-1AM0	For shield support of the USS® bus cable on the ET 200SP BaseUnit

2.1.2 Installation and wiring

As a general rule, the installation and wiring instructions are based on the manuals (\3\, \5\, \8\, \10\) of the individual components.

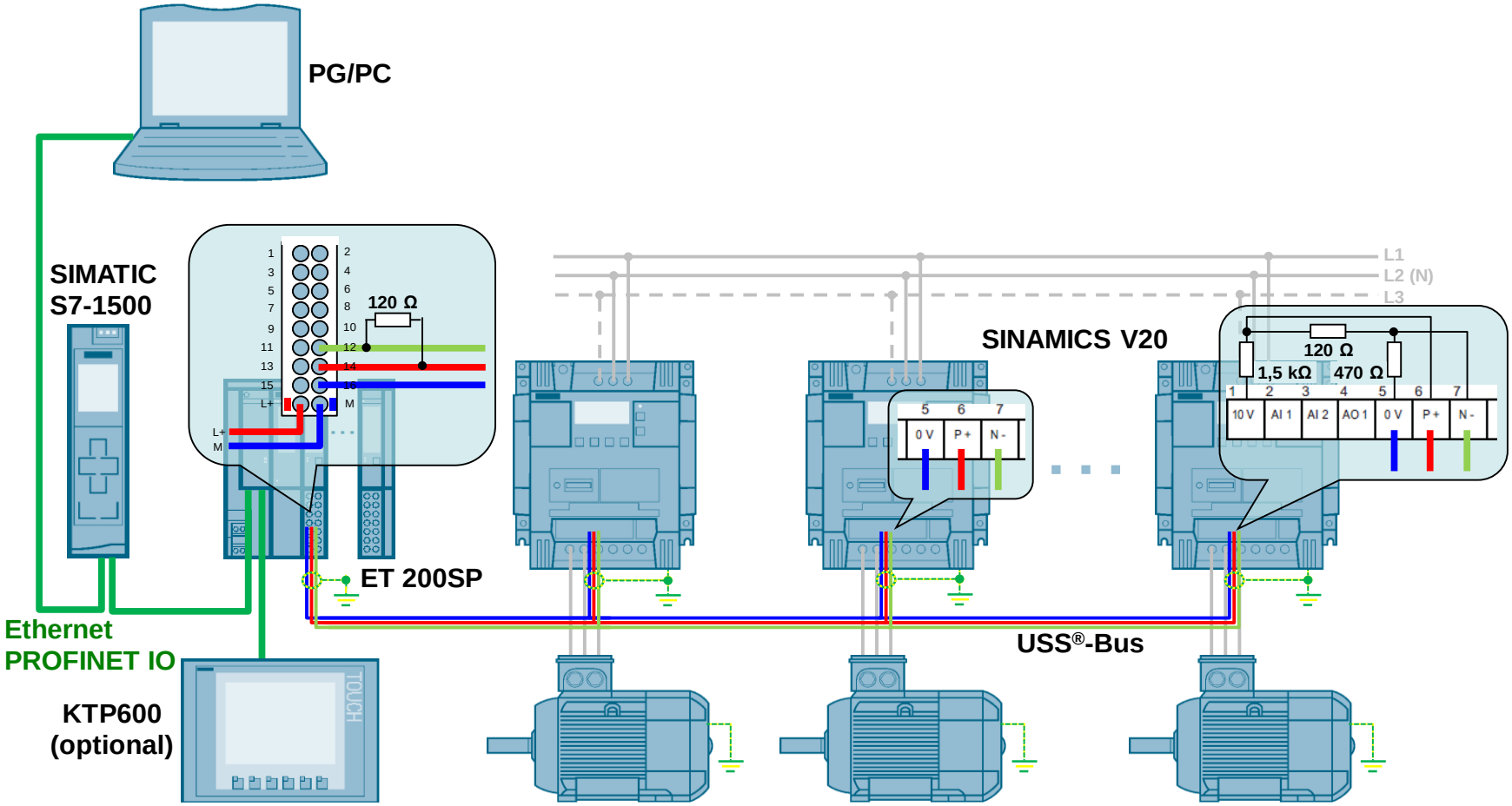
Table 2-2: Instructions on installation and wiring

	Instruction
1.	Install the SIMATIC S7-1500 CPU (pos. 1) and the SIMATIC PM 1507 power supply (pos. 3) onto the DIN rail (pos. 4). An instruction can be found in library \3\ under "Automation System S7-1500 / 5 Installation".
2.	Install the interface module (pos. 5), the base unit (pos. 7) and the server module (pos. 6) of the ET 200SP in the order specified from left to right onto the standard DIN rail (pos. 9). An instruction can be found in library \5\ under "System / ET 200SP distributed I/O system / 4 Installation".
3.	Insert the communication module (pos. 8) into the BaseUnit (pos. 7).
4.	Wire the 24V supply: • SIMATIC PM 1507 (pos. 3), see library \3\ under "Load current supply/ load current supply module PM 70 W / 3 Connection" • S7-1500 CPU (pos. 1), see library \3\ under "Automation system S7-1500 / 6 Connection / 6.5 Connecting power supply to the S7 CPU"; • ET 200SP Interface module (pos. 5), see library \5\ under "Interface module / Interface module IM 155-6 PN ST / 3 Connection / 3.1 Pin assignment" • ET 200SP BaseUnit (pos. 7), see library \5\ under "BaseUnits / BaseUnits (6ES7193-6BP…) / 3 BU15-P16+A10+2D / 3.2 Connection"; • KTP600 (Pos. 12), see operating instruction \10\ under "3.3 Connecting operating device;
5.	Establish the PROFINET IO / Ethernet connection of the respective components via Ethernet patch cable (pos. 14) according to Figure 2-1.
6.	Wire the USS® bus. Bus wiring is explained in detail in the following chapter.

[7] Not required if you simulate the touch panel in TIA Portal.

2.1.3 Bus connection

Figure 2-1:Wiring example USS® bus



The bus cable consists of three wires and has to be shielded:

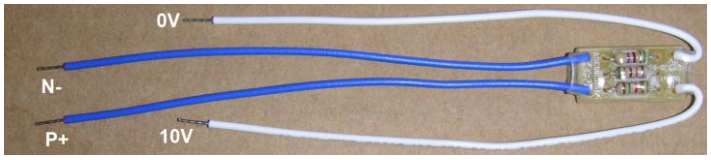
- Bus cable A (N-)
  Is displayed green in Figure 2-1.
- Bus cable B (P+)
  Is displayed red in Figure 2-1.
- Potential equalization cable
  Connects the bus voltage reference points (0V) of all bus nodes with each other and is displayed in blue in Figure 2-1.

The shield of the bus line of each bus node is to be connected with its protective earth connection. Connect the shield via a shield terminal/support (pos. 15) with the DIN rail on the BaseUnit (pos. 7) of the ET 200SP.

The correct bus termination has to be observed on both sides. Bridge the bus lines A and B with a 120Ω resistor (in Figure 2-1 on the ET 200SP) on the bus end and attach a termination network on the other bus end on the last SINAMICS V20, according to the display in Figure 2-1.

The bus termination network is available as accessory. Article number, see Table 2-1.

Figure 2-2: Bus termination network



Note

You can also use the bus termination network from Table 2-1 as simple terminating resistor for the other bus side by only connecting the blue wires and removing the white ones, if necessary.

Observe the following literature when establishing the bus connection:

- Library \5\: "I/O module / communication module (CM) / communication module CM PtP / 3 Connection"
- Library \5\: "System / ET 200SP distributed I/O system / 5.6 Connecting cable shields".
- Operating instruction SINAMICS V20 \8\: "6 Communication with the PLC"
- USS® specification \9\: "Chapter B: physical interface and bus structure"

2.2 Software

2.2.1 Standard software

Table 2-3: Standard software components

Component	Article number	Note
SIMATIC STEP 7 Professional V12 SP1 Floating License	6ES7822-1AA02-0YA5	For S7 1500 CPUs with firmware <V1.5 Update 4 mandatory.
Updates for STEP 7 V12 SP1	free download \6\	
SIMATIC STEP 7 Professional V13 Floating License	6ES7822-1AA03-0YA5	For S7-1500 CPUs with firmware ≥V1.5 Update 1 mandatory.
Updates for STEP 7 V13	free download \7\	

To simulate the operator panel in the TIA Portal, no additional software is required.

2.2.2 Firmware version of the SINAMICS V20

The firmware version of your SINAMICS V20 is stored in parameters r0018 and r0964 and can be read out.

Parameter r0018: Firmware version

The value has data type REAL

Example: r0018 = 3.51

Parameter r0964: Firmware version data

Table 2-4: Firmware of the SINAMICS V20

r0964 (firmware version)		
Index	**Meaning**	**Value**
[0]	Company (Siemens = 42)	42
[1]	Product type	8001
[2]	Firmware version	351
[3]	Firmware date (year)	2012
[4]	Firmware date (day/month)	1012
[5]	Number of inverter objects	1
[6]	Firmware version	500

The application example was created with the above firmware version.

2.2.3 User software and documentation

The following table includes all downloadable files of this application.

Table 2-5: Projects, libraries and documentation

Component	Note
90468030_V20_at_ET200SP_USS_proj_V13_Vxdy.zip[8] (archive file) V20_at_ET200SP_USS_proj_V13 (project folder)	STEP 7 V13 Project
90468030_V20_at_ET200SP_USS_lib_V13_Vxdy.zip[8] (archive file) V20_at_ET200SP_USS_lib_V13 (library folder)	STEP 7 V13 Library
90468030_V20_at_ET200SP_USS_proj_V12_Vxdy.zip[8] (archive file) V20_at_ET200SP_USS_proj_V12 (project folder)	STEP 7 V12 Project
90468030_V20_at_ET200SP_USS_lib_V12_Vxdy.zip[8] (archive file) V20_at_ET200SP_USS_lib_V12 (library folder)	STEP 7 V12 Library
90468030_V20_at_ET200SP_USS_Vxdy_en.pdf[8]	This document
90468030_V20_at_ET200SP_USS_Vxdy_SHORT-DOCU_en.pdf[8]	Short Documentation

The core of the application example and the STEP 7 project are two function blocks. They are additionally stored in Table 2-5 in the listed global STEP 7 library for separate use by the user. If you are working with the project, you do not require the library.

[8] Vxdy = Version ID

3 Commissioning

3.1 Requirements

1. The application example uses the hardware components according to Table 2-1. Structure and wiring is according to chap. 2.1.
2. The example configuration includes one SINAMICS V20 and one SIMATIC ET 200SP, which only has a CM PtP as the only I/O module. For expansion see chap. 7.
3. For the example you do not necessarily need a motor. However, if you so connect one, you have to set the correct motor parameters in the inverter. (This is performed in step 3 of the following step table.)
4. Use the TIA Portal software from Table 2-3 or newer.
5. Make sure that the firmware versions of your hardware components used, corresponds to the ones specified in chap. 2 or newer.
6. The instructions below assume that the inverter is in delivery state or was reset to factory settings.
7. You should have sufficient basic knowledge on SINAMICS inverters, SIMATIC S7-1500 controllers and TIA Portal.

CAUTION	**Observe all security-relevant notes regarding commissioning and operation in the manuals of the components used.** **If the inverter configuration and the data of a connected motor do not match, inverter and/or motor could be damaged or destroyed.**

3.2 Instruction

Changing inverter parameters via the BOP

Proceed as described in "Table 3-2: Step table for commissioning the application example" when changing parameters, as described in the following table.

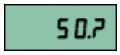
Table 3-1: Changing inverter parameters via the BOP (general)

	Step
1.	You already went to the setup or parameter menu via the respective step in Table 3-2.
2.	Select the parameter number with the ▲ ▼ arrow buttons and press OK .
3.	If parameter index is available: Select the index with the ▲ ▼ arrow buttons and press OK .
4.	Select the parameter value with the ▲ ▼ arrow buttons and press OK .

Step table for commissioning the application example

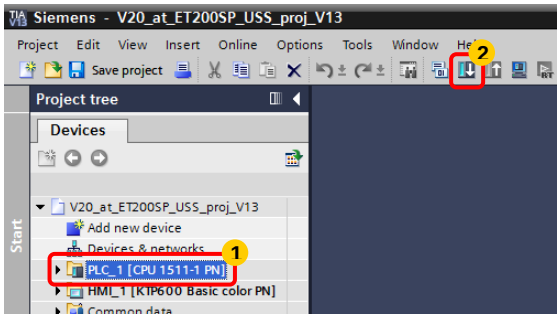
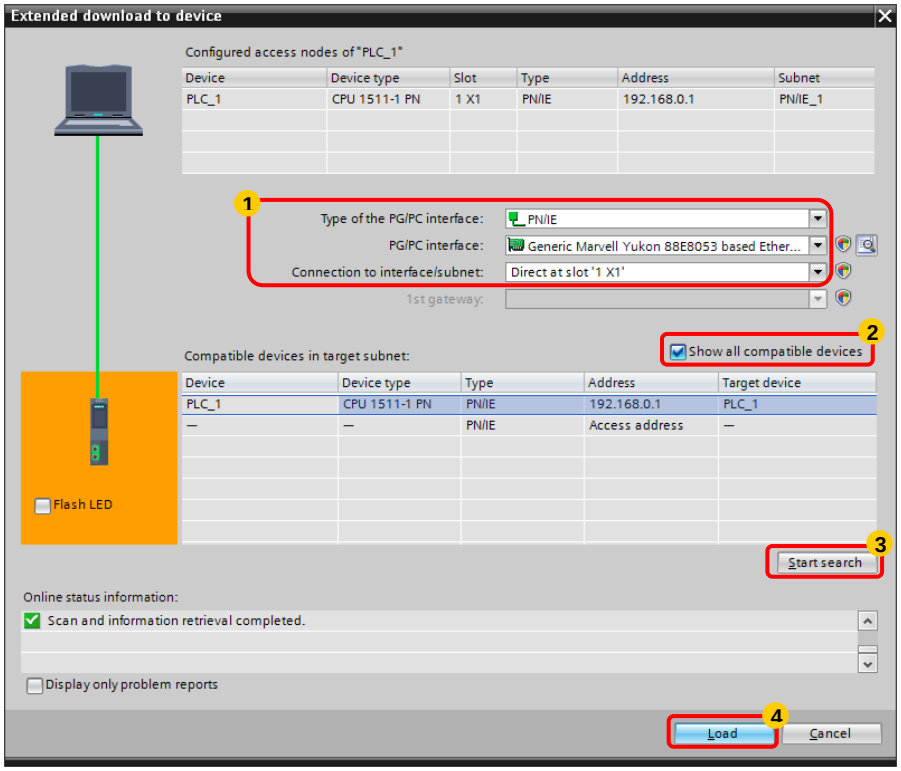
On the primary side apply 3-phase 400V~ or 1-phase 230V~ on the SINAMICS V20 – depending on the inverter type used- and connect the supply voltage on SIMATIC S7-1500 CPU and the SIMATIC ET 200SP. Subsequently, follow the steps in the instruction below:

Table 3-2: Step table for commissioning the application example

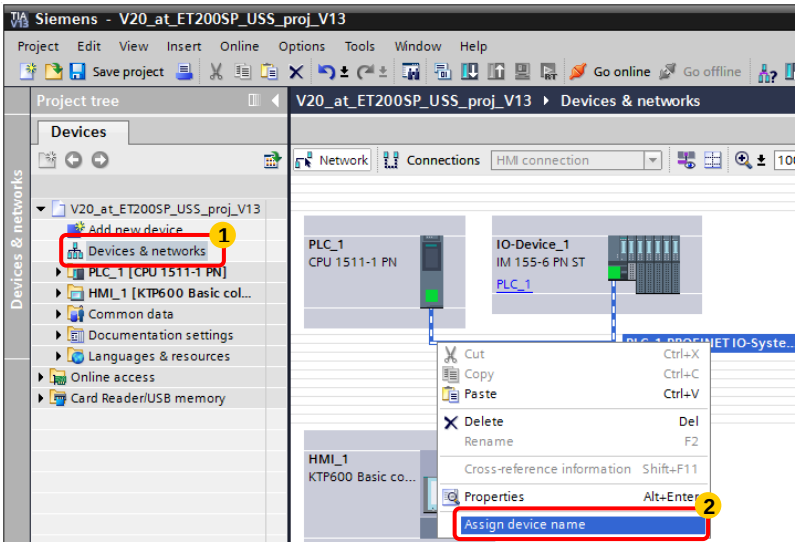
	Implementing the application example
	BOP configuration of the SINAMICS V20
	General: • Below, WE stands for "factory settings" (= delivery state). If not mentioned otherwise, press buttons [M] and [OK] <2s.
1.	Reset all parameters to factory settings if the SINAMICS V20 is no longer in delivery state. For this purpose, go from the display to the parameter menu with [M] and change the following parameters: Access level P0003 ⇨ 1 (WE: 1) Commissioning parameter P0010 ⇨ 30 (WE: 0) Reset to factory setting[9] P0970 ⇨ 21 (WE: 0)
2.	 is shown on the display. Select the motor base frequency (50 Hz or 60 Hz) that is suitable for your region with the arrow buttons [▲] [▼] and the dimension for the power settings (kW or PS) and exit the screen with [OK] (chap.5.3 in\8\).
3.	You are now in the setup menu and you can enter the motor parameters, starting with P0304, if you are executing the example with connected motor. Press [M] (>2s) once you have finished entering the motor parameters and the motor calculations or if you do not want to enter any motor parameters now[10]. You are now in the display menu again.
4.	Go to the parameter menu with [M] .
5.	Control or change the following parameters:[11] Access level P0003 ⇨ 3 (WE: 1) Baud rate P2010[0] ⇨ 8[12] (WE: 8) USS address P2011[0] ⇨ 1[13] (WE: 0) PIV length P2013[0] ⇨ 4 (WE: 127) Select RS485 protocol P2023 ⇨ 1 (WE: 1)
6.	Transfer the parameter values from RAM to EEPROM: RAM to EEPROM P0971 ⇨ 21 (WE: 0) Access level P0003 ⇨ 1

[9]When resetting to factory settings, the connection parameters P2010, P2011, P2023 used for the USS protocol are not automatically reset in the process. However, they are supplied in step 5 of the table.
[10] Complete basic commissioning with entering the motor parameters and motor calculations is not subject of this application.
[11]In the application example, the connection parameters in SINAMICS V20 are not set by means of the Cn010 connection macro (for USS), but individually.
[12] The application uses default value 8 (38400 bps). If you want to change it, you also have to change the respective IN_P2010 variable in V20_USS_Control_1_DB.
[13] For several drives they are to be numbered without any gaps from 1 onward.

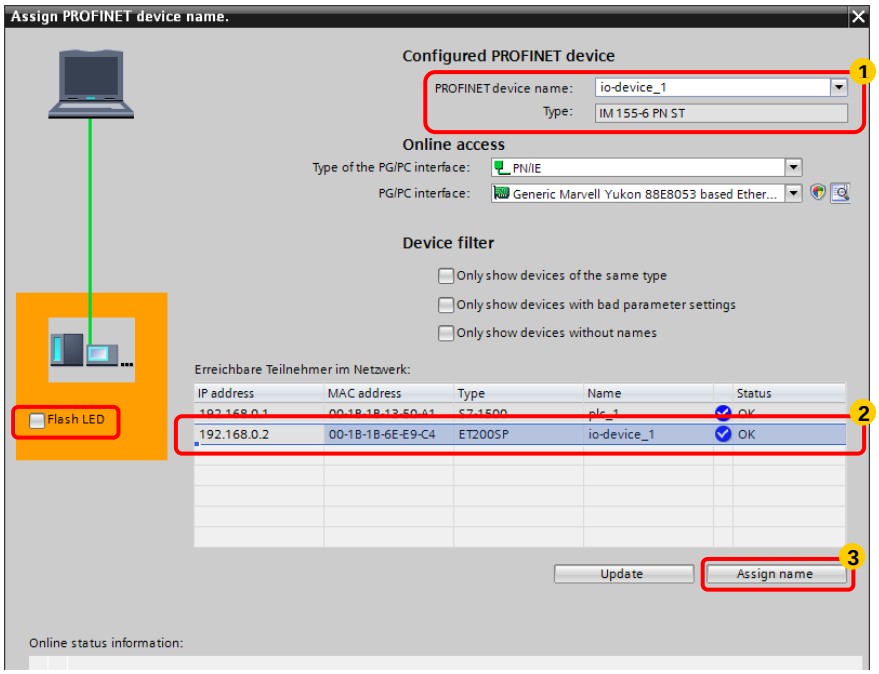
	Implementing the application example
7.	When the parameter entry is completed, go back to the display menu by pressing M (>2s).
8.	Switch the SINAMICS V20 off and back on again. Wait until the LED or the display goes out after switch off (may take some seconds), before you switch the device back on.
Loading the SIMATIC program	
9.	Start the TIA Portal and open the project **V20_at_ET200SP_USS_proj_V13** (in TIA Portal V13) or **V20_at_ET200SP_USS_proj_V12** (in TIA Portal V12), which you have downloaded and unzipped from the Siemens Industry Online Support pages.

	Implementing the application example
10.	Load the controller project into the CPU.  If the window for further download appears, select … • Type of the PG/PC interface: ⇨ PN/IE • PG/PC interface: ⇨ select interface card • Connection to interface/subnet: ⇨ Directly on slot '1 X1' or PN/IE_1 …and start the search. Make sure that the is a tick at "Show all compatible nodes" (default setting), so that your CPU can be found when the IP address so far does not correspond to the address configured in the application example. Subsequently click "Load".  Follow the load process on the …window 1. "Load preview" (continue with the "Load" button) and 2. "Load results" (continue with the "Finish" button).

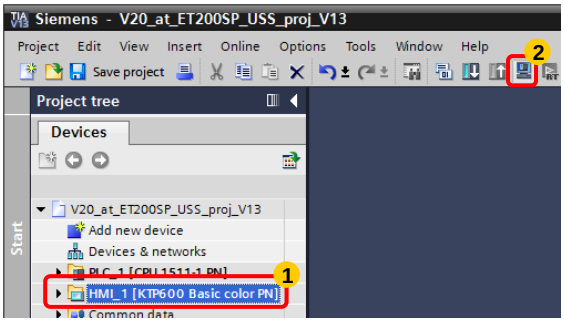
	Implementing the application example
11.	Assign the configured device name to the ET 200SP IO device.



For this purpose, right click the PROFINET IO system in the “Devices & networks” editor on and select “Assign device name” from the context menu.



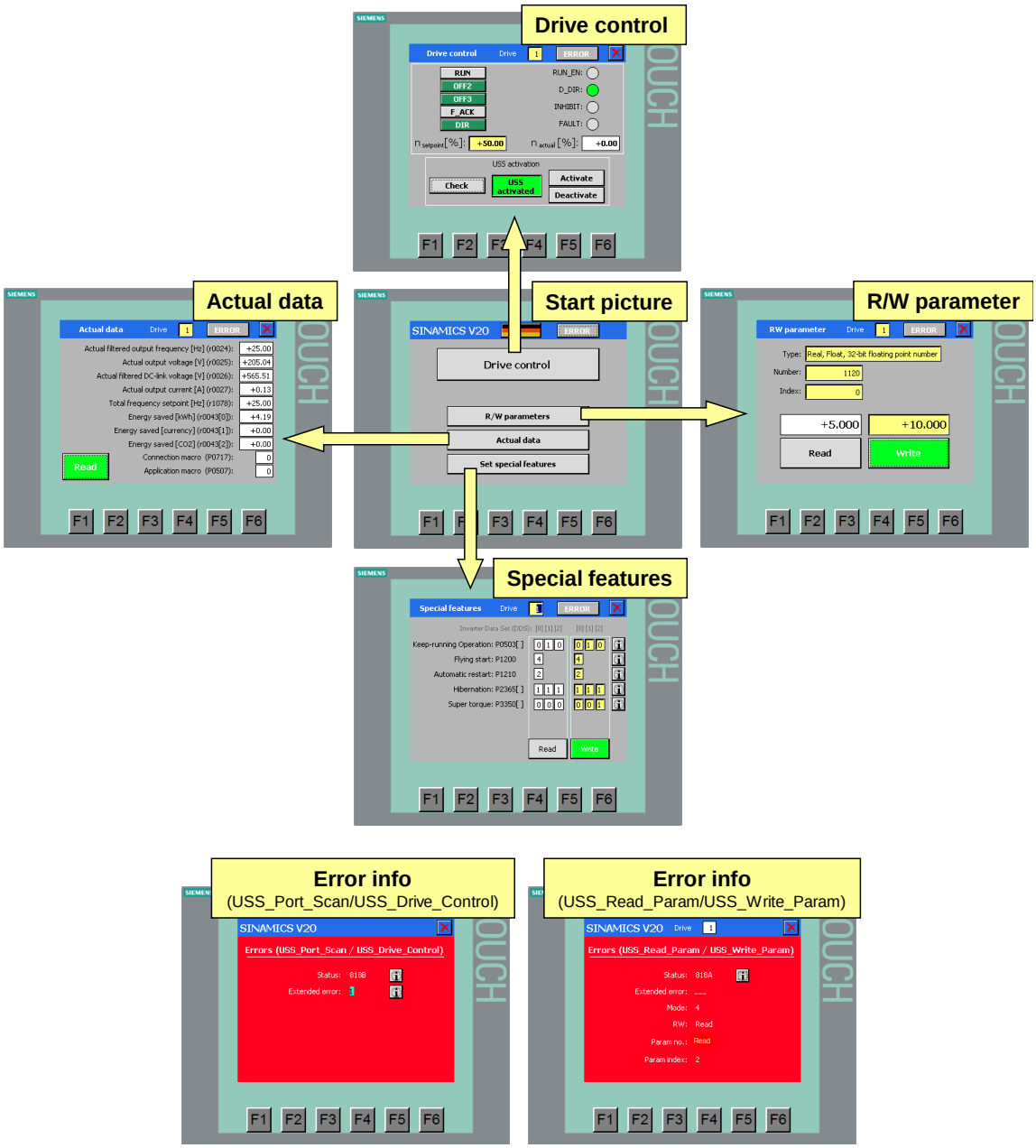
1. Select the configured PROFINET device – here the interface module of ET 200SP – from the dropdown menu.
2. Select the affected reached node that you want to assign the device name. You can identify it based on its IP address, its previous device name, its MAC address or by having it flash.
3. Click “Assign device name“.

	Implementing the application example
12.	Start the simulation of the operator panel.  **NOTICE** Set PG/PC Interface (32-bit) If you are working with TIA V12 make sure that the interface via which you communicate with the controller is selected in the Windows® system control. After completed compilation, the start window of the simulated operator panel will open on the screen.  **Note** The warnings while compiling the HMI can be ignored. They only point to object overlaps. Object arrangements in several layers however is common practice.

4 Operation

4.1 Screen navigation

Figure 4-1: Screen navigation



Specify the desired language, English or German, in the start screen

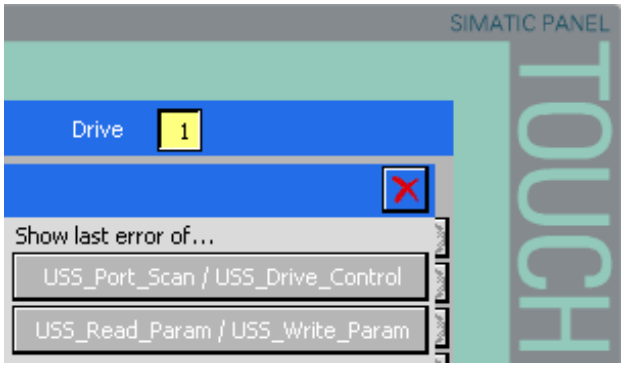
and select the desired function. You get back to the start screen via the button in the respective function screen.

Select the desired drive by entering the drive address in the header of the appropriate function screen[14].

If there is a communication error, the ERROR button in the header will flash red-white. By pressing the button you get to the respective screen of the error information. With the button in the error information screen, you get back to the previous screen.

If there is no current error and the ERROR button does not flash red-white, you can get the error information of the last error back on the screen by clicking it from any screen:

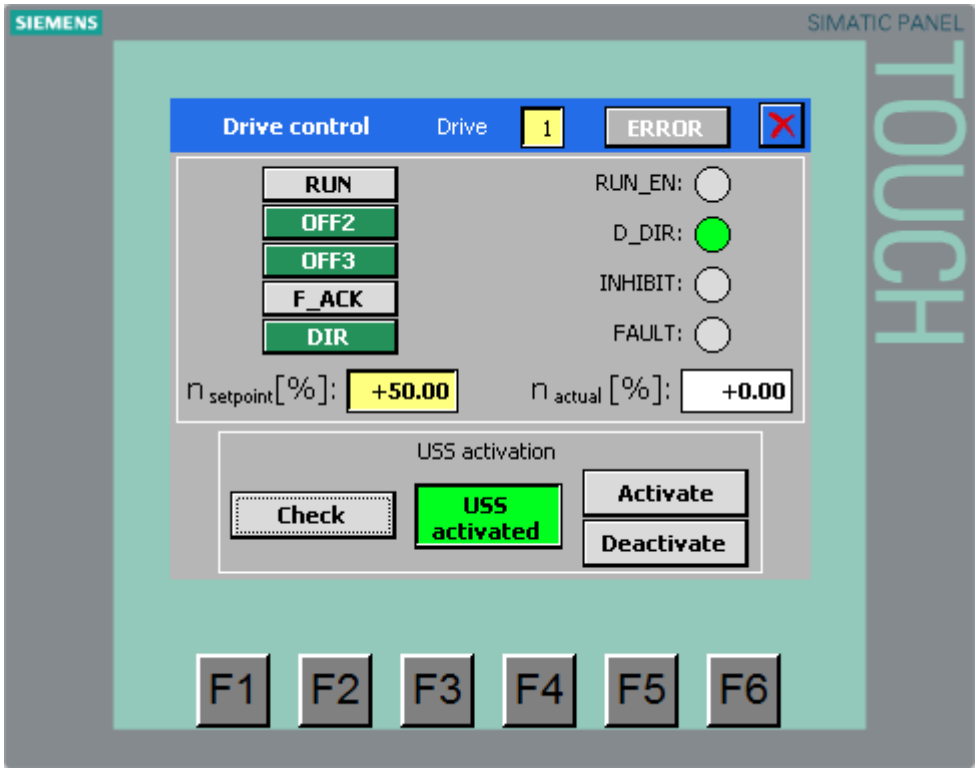
Figure 4-2: Calling last error information



[14]The entry is limited to the number of configurable drives ("V20_USS_Control_DB_1".IN_Number_of_drives).

4.2 “Drive control” screen

Figure 4-3: “Drive control” screen



Drive control

In the top part, the screen includes the following elements:

- **Buttons**
  They correspond to the control bits listed in chap. 1.1. For the buttons RUN, OFF2, OFF3 and DIR, the logic status is inverted when clicking. F_ACK supplies “true”, as long as the button is pressed. The green color of the button signifies the “true” signal state. If there is a restart of the controller, OFF2, OFF3 and DIR are set to “true” by default.
- **Entry field for speed**
  It corresponds to the setpoint speed demanded in chap. 1.1 and is entered in %. Possible values are -100.0…+100.0.
- **Illuminated display**
  They correspond to the status bits listed in chap. 1.1. The green color of the respective display signifies the “true” signal state.
- **Output field for speed**
  It corresponds to the actual speed value demanded in chap. 1.1 and is displayed in %.

USS activation

Before the SINAMICS V20 can be controlled via the USS communication, the command and frequency setpoint source have to be changed to USS by changing the parameters (see chap. 5.5.1). In the bottom part of the screen, you can carry out the following three functions of the parameter operations of the ACTIVATE_USS mode at the touch of a button:

- **Check**
  Query whether USS is enabled for the drive control.
- **Activate**
  Enabling USS for the drive control.
- **Deactivate**
  Disabling USS regarding the drive control. A re-parameterization to the command and frequency setpoint source that was previously set before changing to USS is carried out.

The execution of the according event is temporarily displayed in a display field with details regarding the event.

Note If the SINAMICS V20 cannot be operated in the “Drive control” screen, get some information on the USS enabling status first by pressing the check button.

4.3 “Read/write parameters” screen

Figure 4-4: “Read/write parameters” screen

Reading parameters

In order to read any parameter, make the appropriate entries in the yellow input fields: type, number and index. The input field for the data type is a drop-down list for selecting the format. The required formats you will find in the parameter list in the SINAMICS V20 manual (\8\). Then subsequently press the “Read” button. The

parameter value read from the inverter is entered in the white output field above the read button. The execution of the event is displayed by a temporary green color of the read button.

Writing parameters

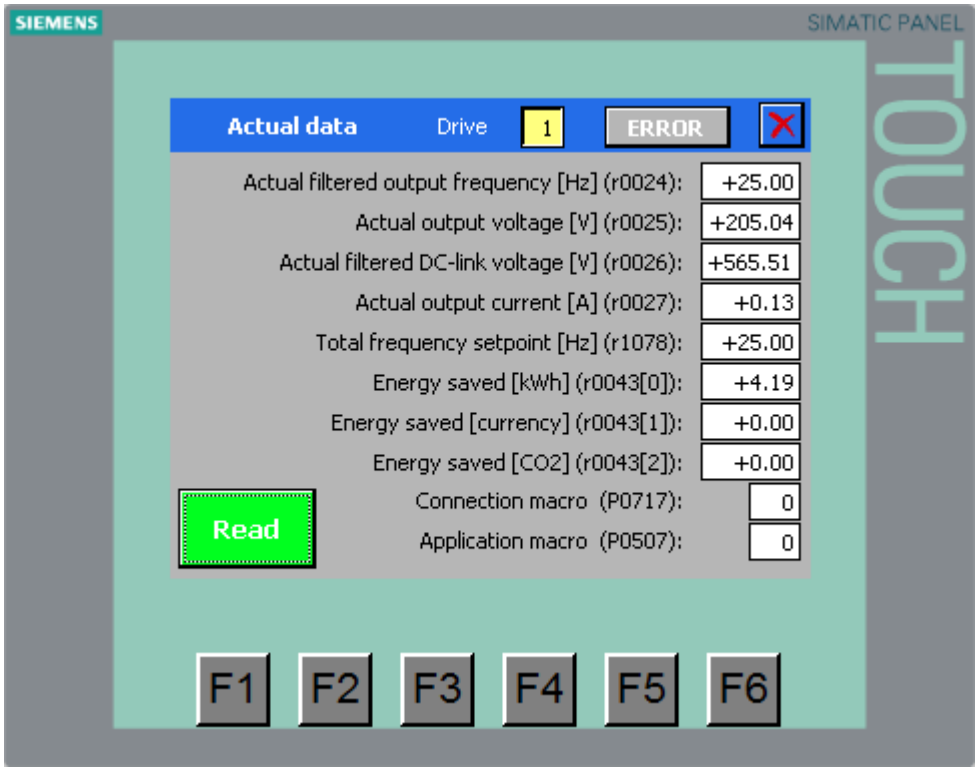
In order to write any parameter, make the appropriate entries in the yellow input fields: type, number and index. Write the new parameter value in the yellow input field above the write button. Then, press the “Write” button. This writes the new parameter in the RAM[15] of the inverter, for control purposes it is read out straight away again[16] and entered in the white output field, above the read button. The execution of the event is displayed by a temporary green color of the write button.

NOTICE	**Entering an incorrect data format (type), does not necessarily cause an error message. An incorrect parameter value may be written or read.**

[15] In the application example the RW_PARAM.EEPROM FB parameter of V20_USS_Param [FB2] is permanently assigned to “false”.
[16] In the application example the RW_PARAM.FUNCTION FB parameter of V20_USS_Param [FB2] is permanently assigned to “2”.

4.4 “Actual data” screen

Figure 4-5: “Actual data” screen



The parameters to be read correspond to the demands from chap. 1.2.

Note

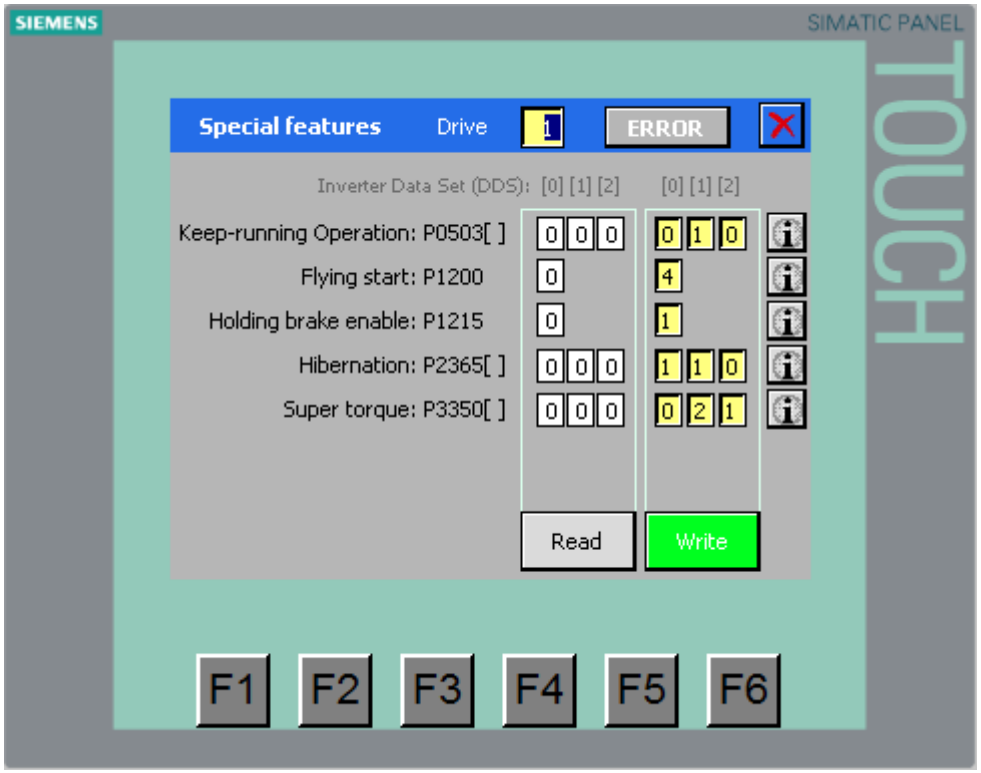
Connection macros and application macros provide preallocated sets of parameters for a quick commissioning. Being able to belatedly modify separate parameters of this sets, the macro values do not necessarily indicate the actual communication and application parameters.

Reading a set of current status parameters

Press the “Read” button. The set of parameters read from the inverter is entered in the white output fields. The execution of the event is displayed by a temporary green color of the read button.

4.5 “Special features” screen

Figure 4-6: “Special features” screen



The function parameters to be read/written correspond to the demands from chap. 1.2.

Reading out inverter functions

In order to read the function parameters, click the “Read” button. The set of parameters read from the inverter is entered in the white output fields above the read button. At the same time, the yellow input fields with the read out values are preassigned. The execution of the event is displayed by a temporary green color of the read button.

Writing inverter functions

To write a set of parameters, enter the respective values in the yellow input fields. Afterwards, press the “Write” button below the input fields. This writes the new function parameters in the RAM[17] of the inverter, for control purposes they are read out straight away again[18] and are entered in the white output fields above the read button. The execution of the event is displayed by a temporary green color of the write button. A parameter value to write, which exceeds its permitted value range, causes an error reaction of the SINAMICS V20. At the end of the complete read/write sequence, the last failed write attempt is displayed according to chap. 4.7. An easy limitation to a valid parameter range at HMI side would be

[17] In the application example the SET_FUNCTIONS.EEPROM FB parameter of V20_USS_Param [FB2] is permanently assigned to “false”.
[18] In the application example the SET_FUNCTIONS.FUNCTION FB parameter of V20_USS_Param_1 [FB2] is permanently assigned to “2”.

possible. But we abstained from this for the benefit of being able to demonstrate the drive’s reaction.

Note

When writing the parameters, it is always the entire set of parameters that is transferred, therefore also values that you explicitly do not want to change. For this reason, proceed as follows to prevent an undesired overwriting of individual parameters.

1. Read out the set of parameters. With the read out values, the yellow input fields are automatically preassigned.
2. Enter the parameter or parameters to be changed into the yellow input fields.
3. Click the “Write” button.

Info buttons

Below the info buttons you can find explanations on the codes of the parameter values. As long as you hold the respective button down, the text is visible.

4.6 “Error (USS_Port_Scan / USS_Drive_Control)” screen

Figure 4-7: “Error (USS_Port_Scan / USS_Drive_Control)” screen

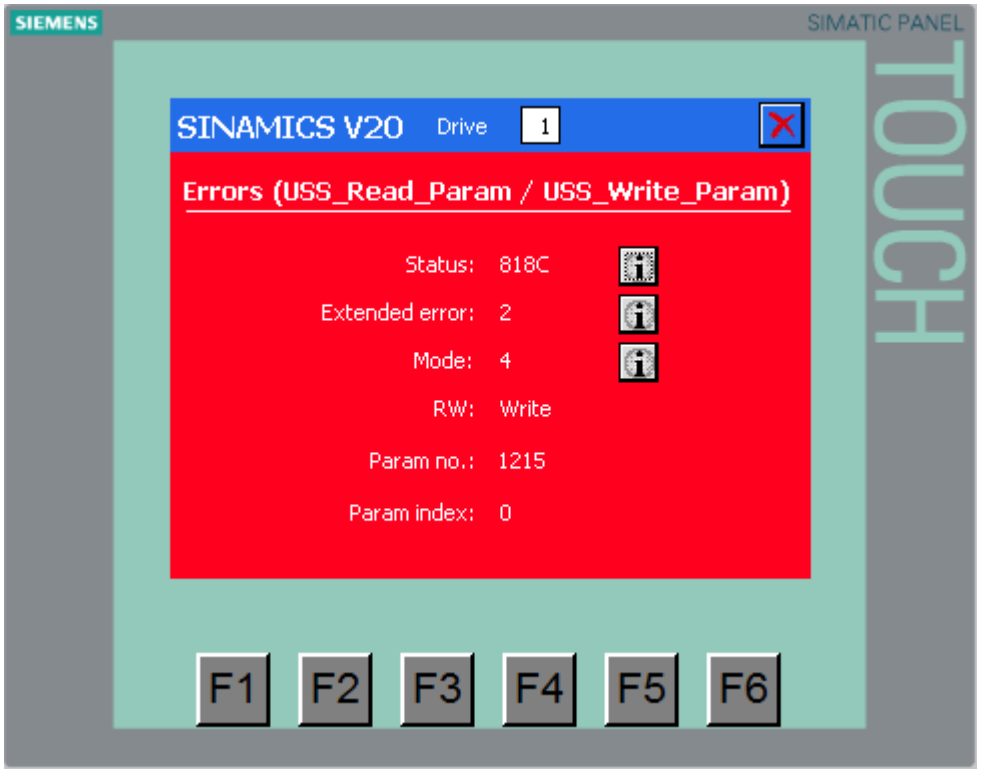
The error status and for some errors an expanded error information of the instruction creating the error is displayed which includes the address of the affected SINAMICS V20. The information always relates to the error that occurred last at the USS_Port_Scan or USS_Drive_Control. A new error overwrites the previous error information.

Info buttons

Below the info buttons you can find explanations on the error codes. As long as you hold the respective button down, the text is visible. The info button to the “Extended error” is only visible if respective error information is available.

4.7 “Error (USS_Read_Param / USS_Write_Param)” screen

Figure 4-8: “Error (USS_Read_Param / USS_Write_Param)” screen



The number of the SINAMICS V20 with the communication failure is written in the header of the error information screen (only output field). Apart from the error status and the expanded error information that is included for some errors, the mode, the data direction (read or write) and the parameter address and index are displayed to be able to locate the error better. The screen always shows the error information of the error that occurred last in a parameter operation. A new error overwrites the previous error information.

Info buttons

Below the info buttons you can find explanations on the error codes. As long as you hold the respective button down, the text is visible. The info button to the “Extended error” is only visible if respective error information is available.

5 Programming

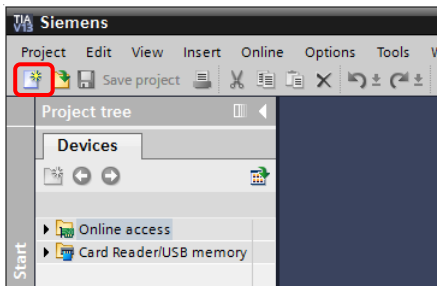
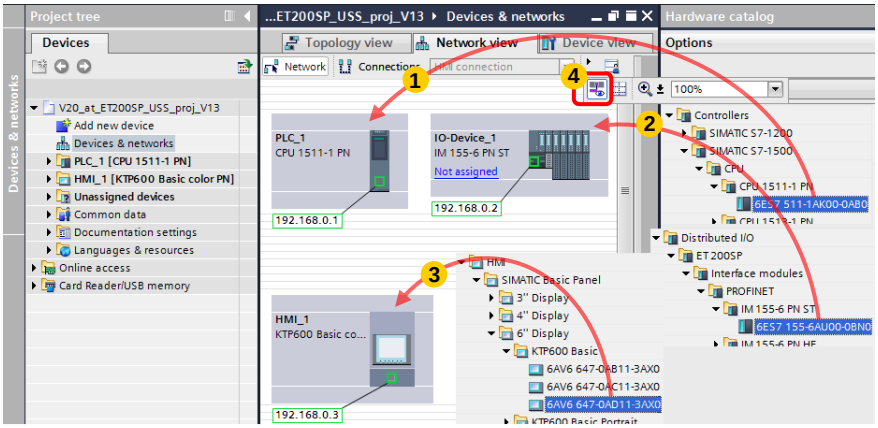
This chapter describes the hardware configuration and the structure of the STEP 7 program. Furthermore, the specific functions are explained in more detail in chap. 1 Task. The chapter is to help you to deepen your knowledge on the functionality of the SINAMICS V20 ⇔ SIMATIC S7-1500 communication via the USS protocol.

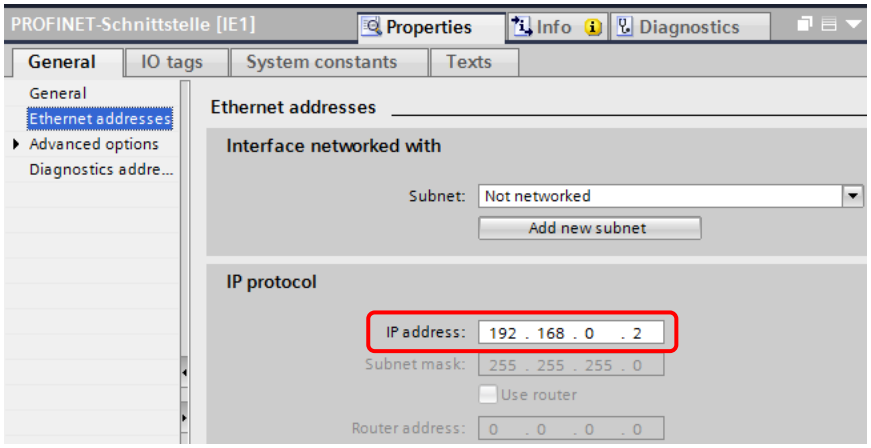
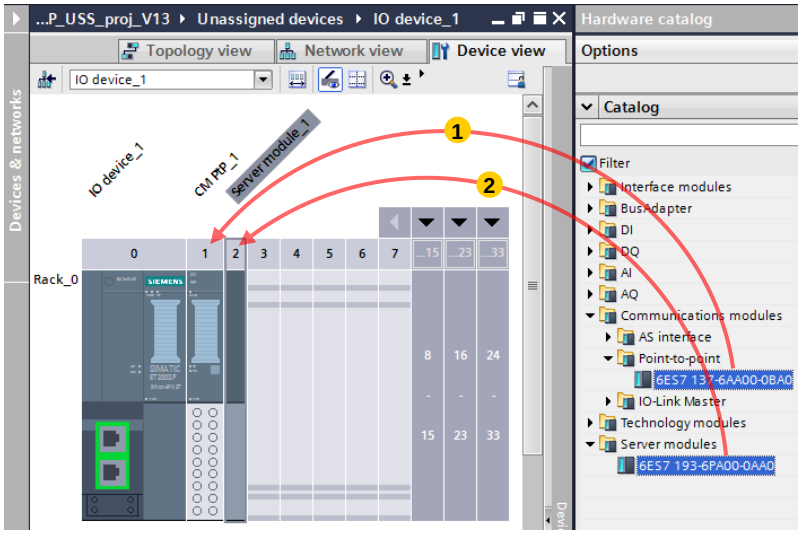
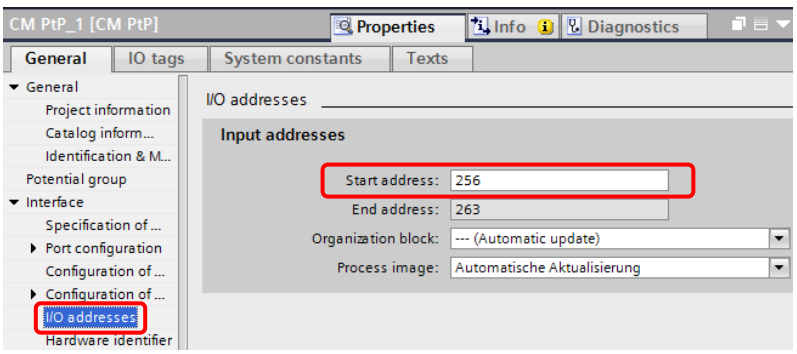
The content of this section is not necessarily required for implementing and operating the application example.

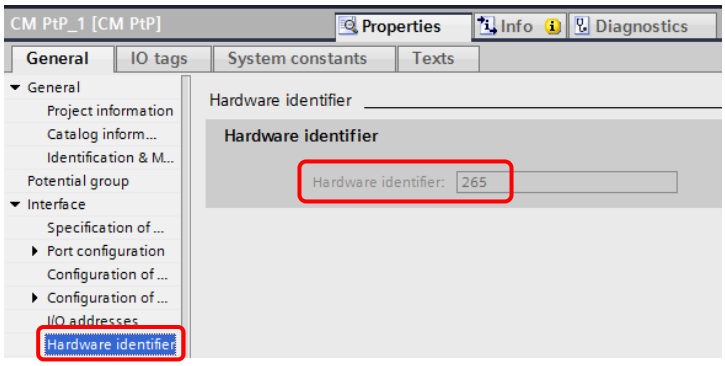
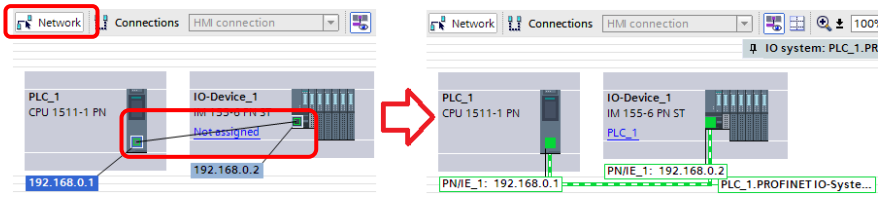
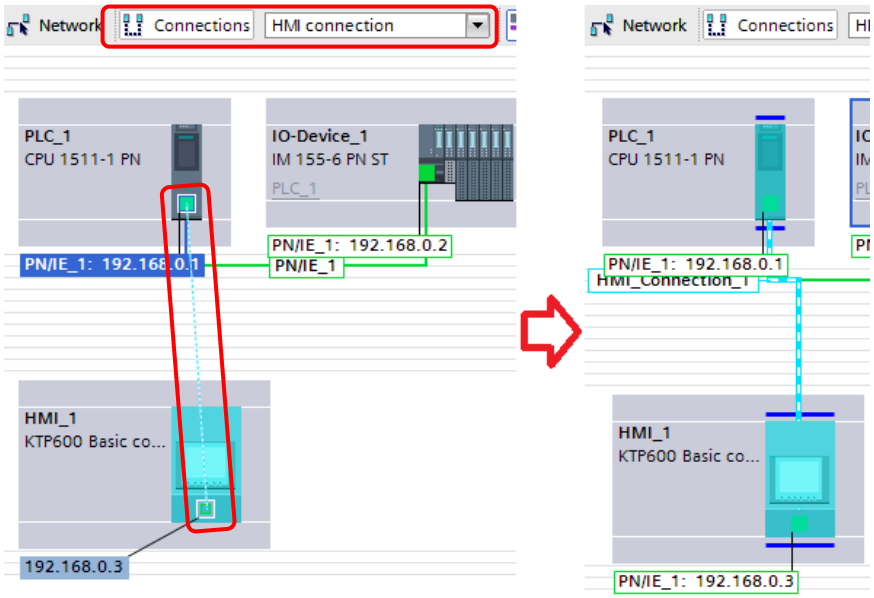
5.1 Hardware configuration

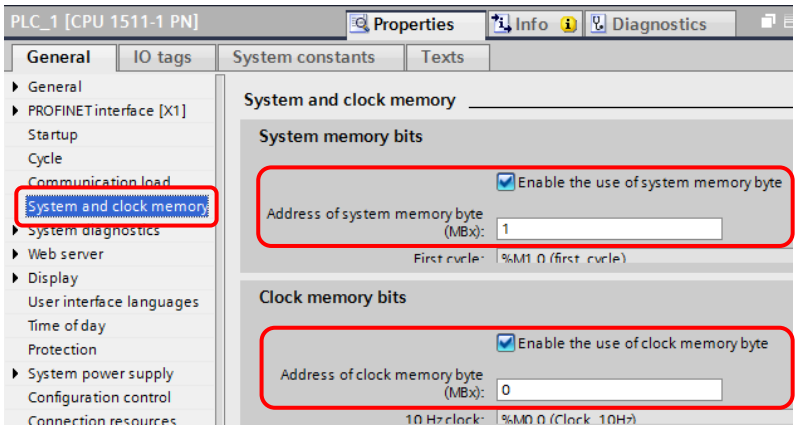
The hardware configuration is part of the loadable application example. If you want to create it yourself, proceed according to the following step table.

Table 5-1: Configuring the hardware

No.	Instruction
1.	Start the TIA Portal in the project view and create a new project. 
2.	Drag the required components from the catalog to the graphic area of the network view in the "Devices and networks" editor (1..3).  Display the IP addresses (4).

No.	Instruction
3.	With a mouse click on the IP address or interface of the affected component you get to its properties. This is where you adjust the IP address of the component to your requirements. 
4.	By double click, for example, you get to the device view of the IO Device_1. Drag the PtP communication module and the server module with the mouse from the catalog to the graphic area of the device view. 
5.	Open the properties of the CM PtP_1 communication module in the device view, for example, by double click. Put the input addresses (8 bytes) to a free area (in the application example 256..263).  Further configuration of the interface is not required here, since this is done via the system function blocks.

No.	Instruction
6.	Remember the hardware identifier of the entered CM.  The correlation between the code blocks of the user program and the respective communication module is only made via the hardware identifier. It is to be entered into the respective instance DB V20_USS_Control_DB_n in the IN_HW_Id variable.
7.	Network the CPU with the ET 200SP by selecting “Network” and connecting the interfaces graphically with the mouse. 
8.	Establish a HMI connection between the CPU and the KTP600 by selecting “Connections > HMI connection” and connect the interfaces graphically with the mouse. 

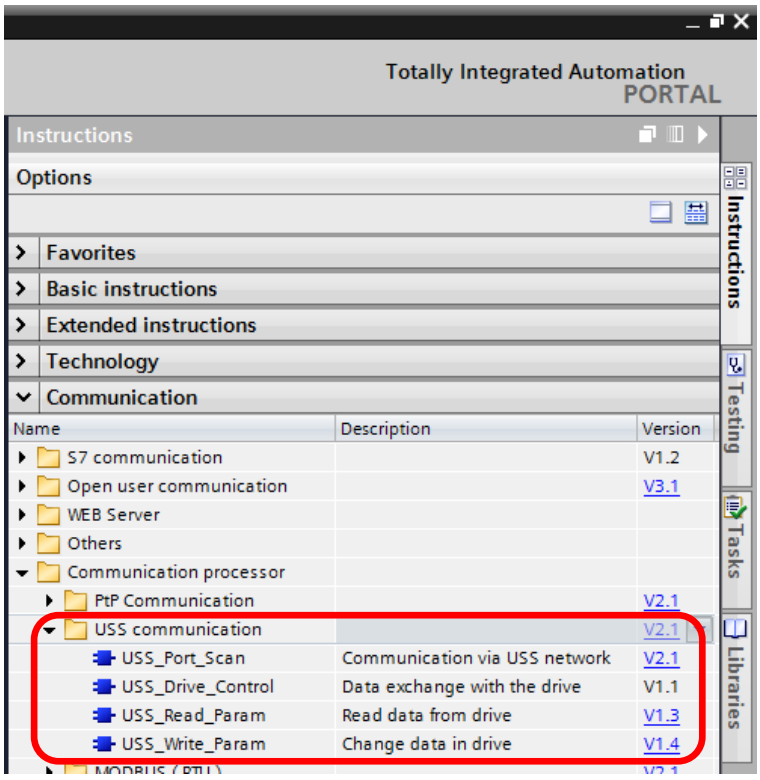
No.	Instruction
9.	Adjust the CPU properties to your requirements. For example, make a right click in the network view on the CPU and select “Properties” from the context menu. The application example does not require any special settings. Only the system memory bits (MB1) and the clock memory bits (MB0) have to be released. 

5.2 Relevant code blocks

USS instructions

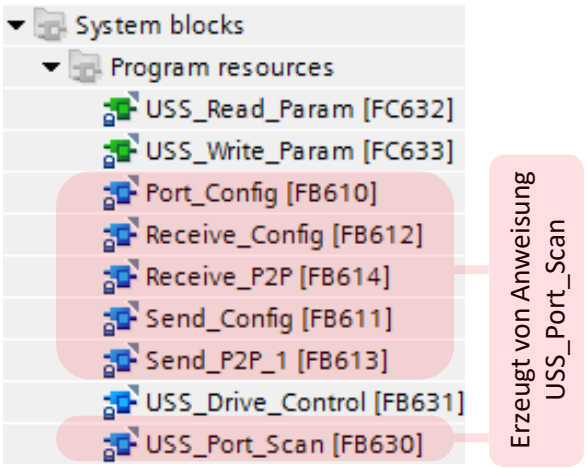
The functions specified in chap. 1 Task use the following four instructions provided by the TIA Portal as a basis and which you can find in the “Instructions” task card.

Figure 5-1: USS instructions



- **USS_Port_Scan (TIA V13: V2.1; TIA V12: V1.2)**
  When inserting into the program, the instruction generates the system function blocks displayed in Figure 5-2 along with the instance data[19] The FB USS_Port_Scan processes the communication via the USS network.
- **USS_Drive_Control (TIA V13: V1.1; TIA V12: V1.0)**
  When inserting it in the program, the instruction generates a system block with the same name along with the instance data block[20]. See Figure 5-2. The USS_Drive_Control exchanges data with the SINAMICS V20 by generating request messages and evaluating the reply messages of the V20.
- **USS_Read_Param (TIA V13: V1.3; TIA V12: V1.2)**
  When inserting in the program, the instruction generates a system function with the same name. See Figure 5-2. The FC USS_Read_Param reads a parameter from the SINAMICS V20.
- **USS_Write_Param (TIA V13: V1.4; TIA V12: V1.3)**
  When inserting in the program, the instruction generates a system function with the same name. See Figure 5-2. The FC USS_Write_Param changes a parameter in the SINAMICS V20.

Figure 5-2: USS system FCs/FBs



Details to the above instructions can be found in the Online help in the TIA Portal.

User function blocks

The USS communication is divided in a cyclic and an acyclic part.

Cyclic communication
Within the framework of the cyclic communication, data for drive control is exchanged between the PLC and SINAMICS V20. All drives of a port are operated one after the other. After the last drive, follows the first one again. The data, specified in chapter 1.1 is exchanged. The acyclic communication is realized in **V20_USS_Param [FB1]**.

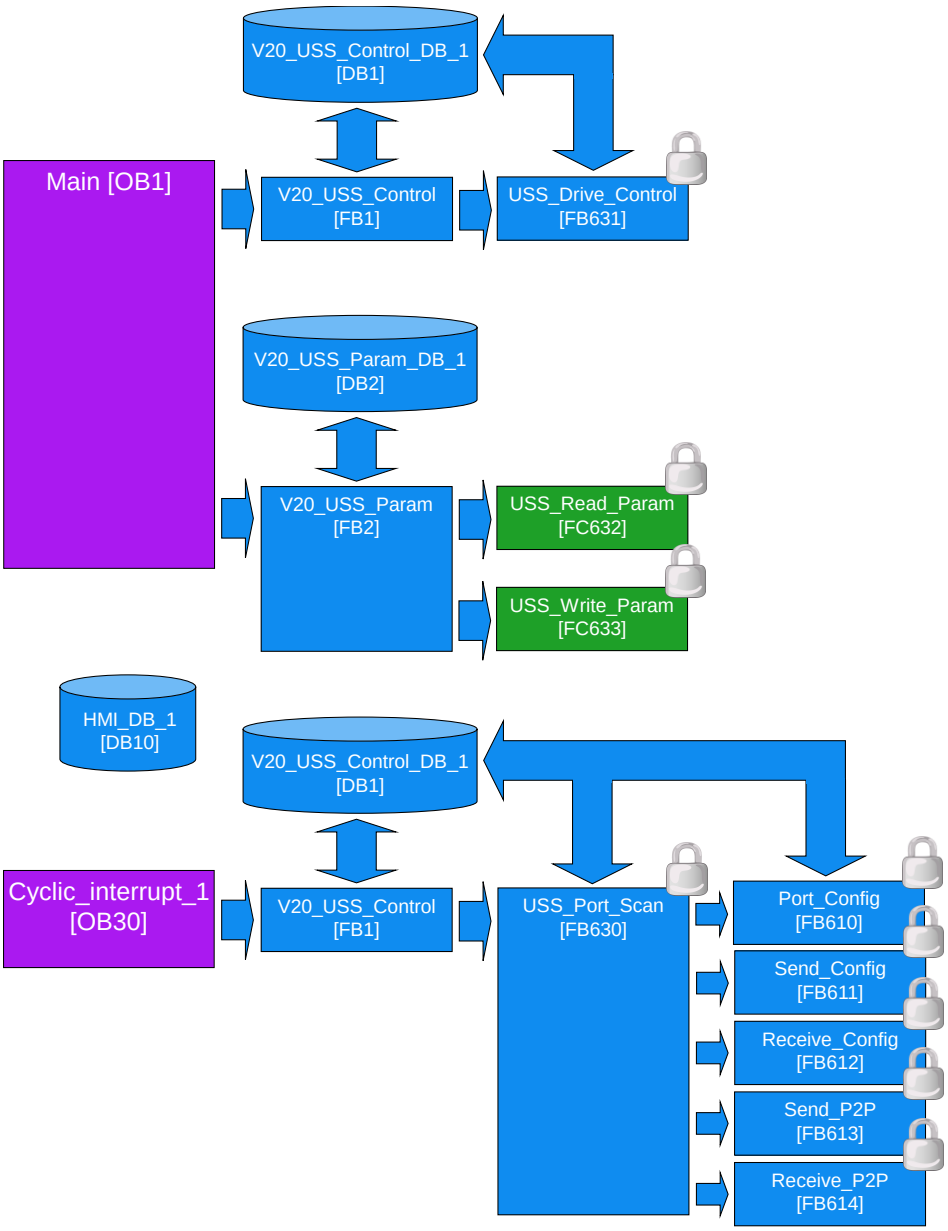
Acyclic communication
Within the framework of the acyclic communication, one or several inverter parameters of a selected SINAMICS V20 are read and/or written once upon request (not cyclically). The user selects one of four available modes, according to the specifications in chapter 1.2. The acyclic communication is realized in

[19] In the application example the instance data is stored in V20_USS_Control_DB_1 [DB1] as USS_Port_Scan_Instance (multiinstance).
[20] In the application example the instance data is stored as USS_Drive_Control_Instance (multi instance) in V20_USS_Control_DB_1 [DB1].

V20_USS_Param [FB2]. Due to the USS system blocks, it is necessary to also call the V20_USS_Control [FB1] in the program when using V20_USS_Param [FB2].

5.3 Program structure

Figure 5-3: Program structure



Calling the V20_USS_Control_1 [FB1] in two OBs

As you can see in Figure 5-3, the V20_USS_Control [FB1] is called in the cyclic program part (OB1) as well as in a cyclic interrupt OB (OB30). The reason for this is the fact that that part of the FB1 that processes the communication via the USS network and calls the USS_Port_Scan [FB630] system function is to run in OB30 and that part that calls the USS_Drive_Control [FB631] system function block and which exchanges data with the SINAMICS V20 is to be processed in OB1. The decision which program part of the FB1 is to be processed in which OB is specified by an input parameter of the FB1 (see chap. 5.4.1).

How often the cyclic interrupt OB and therefore the USS_Port_Scan is called can be configured by the user in V20_USS_Control [FB1].The call interval determines the speed of the communication and has to be seen in correlation with the cycle load of the entire user program. Furthermore, call intervals of OB30 and frame time out of the inverter (P2014) have to be adjusted to each other. Although a call of USS_Port_Scan in OB1 would also be possible, when calling from a cyclic interrupt OB a more constant time behavior of the frame transmission can be guaranteed. In the application example the OB30 is called every 10ms. For more information, see chapter 6.

Supplying the formal parameters of the V20_USS_Control [FB1]

When calling the V20_USS_Control [FB1] in two places, its formal parameter bar also has to be provided with the same parameter values twice. To minimize the effort for the user and to avoid configuration errors, FB1 has only one single formal parameter that can be accessed from outside, in particular the specification whether it is called in OB1 or in OB30. The remaining configuration is performed once directly in the appropriate instance DB V20_USS_Control_DB_1. The parameters to be supplied or removed are stored as static data. They are identified by name and in the comment as user parameter.

Framework program (Main [OB1] and HMI_DB_1 [DB10])

Main [OB1]
The OB has the following tasks:

- Call of V20_USS_Control [FB1] (NW1)
- Call of V20_USS_Param [FB2] (NW2)
- Preparing the feedback signals (done messages) of the V20_USS_Param [FB2], to display them on the operator panel (NW3, NW4)
- HMI screen switching to error masks for communication errors (whilst using control jobs) (NW5)

HMI_DB_1 [DB10]
The block includes the following data:

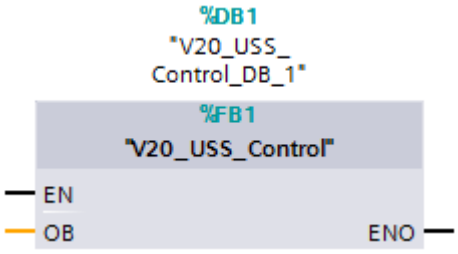
- All actual parameters (INPUT and OUTPUT) that can be specified/displayed via the HMI, of V20_USS_Param [FB2] that are pending at its formal parameter bar. DB10 is therefore used as interface for the operator panel of FB2.[21]
- Data that is required to display the feedback signals (done messages) of V20_USS_Param [FB2] on the operator panel and that was generated in OB1.
- Control bits to coordinate the screen switching in the event of communication errors.
- Mailbox for control jobs for HMI screen switching for communication errors.

[21] Via the HMI, block parameters in the V20_USS_Param_DB_1 [DB2] instance DB can be accessed directly – without detours via the DB10. In this case, the FB2 formal parameters would be unprovided if it is called in Main [OB1]. In view of a transparent display of the program code, it was worked with the DB10.

5.4 V20_USS_Control [FB1] function block

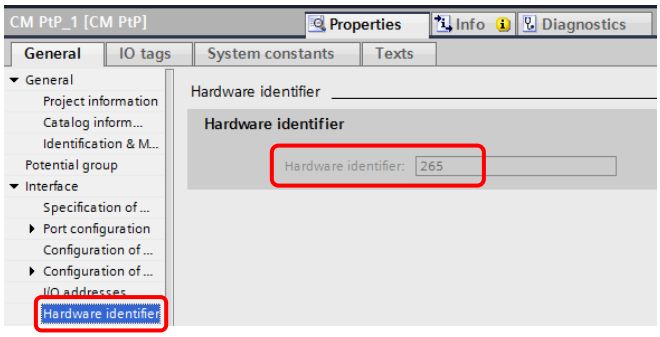
5.4.1 Parameterization

Figure 5-4: Call of V20_USS_Control [FB1] (FB1)



The block has the variable OB as the only parameter of the “Input” data segment. It now appears in the left formal parameter bar of the FB. The further configuration is not performed by supplying the formal parameters when calling the block but in the static data of the appropriate instance DB (see page 35). Each parameter that has to be provided or removed by the user has the IN_... or OUT_... prefix in the variable name or in one of its structural components. In addition, the line comment of an IN parameter starts with “?“ and the line comment of an OUT parameter with “!“. Variables with higher address offset than those in the table below, meaning data that is further down in the DB, must not be changed by the user.

Table 5-2: Parameter of V20_USS_Control [FB1]

Name	IN/ OUT	Type	Explanation
OB *Only parameter of the “Input” data segment!*	IN	USInt	**Call ID** = 1, if the FB is called in the cyclic user program (e.g. by Main [OB1]; ≠ 1, if the FB is called by the cyclic interrupt OB;
IN_P2010	IN	USInt	**Baud rate** Coding is identical with that of the V20 parameter P2010 (values: 6…12). See /8/.
IN_HW_Id	IN	PORT	**Hardware identifier of the communication module** You can find the value in the properties of the CM PtP_1 communication module of ET 200SP. 

Name	IN/ OUT	Type	Explanation
IN_CYCLIC_INTERRUPT. OB_NR	 IN	Struct OB_CYCLIC	**Cyclic interrupt OB** Number of the cyclic interrupt OB, in which the FB V20_USS_Control also has to be called.
CYCLE	IN	UDInt	Call interval of the cyclic interrupt OB
PHASE	IN	UDInt	Phase shift of the cyclic interrupt interval. If several ports are configured and their FBs USS_Port_Scan are called in cyclic interrupt OBs with the same clock cycles, a phase shift helps to achieve a better equal time distribution of the communication. The default value is 0 and can be left as only one configured port.
IN_Number_of_drives	IN	USInt	**Number of drives** Number of drives that are connected to the respective port (1…16).
IN_STW[n]	IN	Array of Word	**Control/status data of the SINAMICS V20** Control word drive [1..16] The V20_USS_Control_1 [FB1] sends the following STW bits to the inverter: ON_OFF Bit 00 OFF2 Bit 01 OFF3 Bit 02 Ack_fault Bit 07 Direction_reversal Bit 11 The remaining STW bits cannot be influenced by the FB1.
IN_SPEED_SP[n]	IN	Array of Real	Setpoint speed value drive [1..16] in %
OUT_ZSW[n]	OUT	Word	Status word drive [1..16] The V20_USS_Control_1 [FB1] receives the following ZSW bits from the inverter: Operation_enabled Bit 02 Motor_rotates_fwd Bit 14 Closing_lockout_active Bit 06 Fault_present Bit 03 The remaining ZSW bits cannot be influenced by FB1.
OUT_SPEED[n]	OUT	Real	Actual speed value drive [1..16] in %
OUT_ERROR	OUT	Bool	**Error** The bit is set for a processing cycle of the FB if an error was detected by the system blocks USS_Port_Scan or USS_Drive_Control.
OUT_ERROR_INFO. DRIVE_ADDR	 OUT	Struct USInt	**Error information**[22] Drive address Address of the SINAMICS V20, where an error was signaled through OUT_ERROR.

[22] It is always only the error information of the error that occurred last that is saved in the instance DB. A new error – signalized by the OUT_ERROR bit – overwrites the previous error information.

Name	IN/ OUT	Type	Explanation
STATUS	OUT	Word	Processing status Error code of USS_Port_Scan or USS_Drive_Control, if OUT_ERROR signals an error. You can find the error codes in the online help to the respective instructions in the TIA Portal.
EXTENDED_ERROR_available	OUT	Bool	Extended error information available From the instance data of the system FB s USS_Drive_Control, an extended error information can be called if OUT_ERROR signals an error.
EXTENDED_ERROR	OUT	UInt	Extended error information The address of the SINAMICS V20 where the communication error occurred can be found in the expanded error information.

5.5 V20_USS_Param [FB2] function block

5.5.1 Operating modes

The parameter accesses carried out with FB2 are summarized in the four operating modes ACTIVATE_USS, RW_PARAM, ACTUAL_STATE and SET_FUNCTIONS in section 1.2. The desired operating mode is specified by the FB2 input parameter MODE (see Table 5-5) that you specify in the application example by selecting the HMI screen (Figure 4-1).

Table 5-3:Operating mode ⇔ HMI screen

Mode	HMI caption
ACTIVATE_USS	Drive control
RW_PARAM	RW parameter
ACTUAL_STATE	Actual data
SET_FUNCTIONS	Special features

ACTIVATE_USS

In this mode you enable the USS communication channel in order to control a SINAMICS V20 via the USS communication. In the process the following V20 parameters are set:

Table 5-4: Inverter parameter for USS communication

Function	No.	Value	Dim.
Selection of command source	P0700	⇨ 5	-
Selection of frequency setpoint source	P1000	⇨ 5	-
USS telegram time out	P2014	⇨ application specific, (230 ms in application example)	ms

In SINAMICS V20 a so called frame time out can be set. Here, the drive monitors respectively the time between two consecutive communication requests. If it is

larger than the configured frame time out, the drive will go to fault with fault code F72. The frame time out to be selected depends on the baud rate, the data volume to be transferred, the number of drives on the port and the frame repetition rate. If the frame time out is configured with 0 ms, no fault code is created in the event of a frame failure.

You can select between the following functions that can be selected with the ACTIVATE_USS.FUNCTION parameter:

- **Check**
  It is checked whether the USS communication has already been selected, meaning whether the three parameter values from the above table are already enabled in the inverter. For this purpose, the current parameters are read out from the inverter. If one of the parameters P0700 and P1000 does not have the value 5, the current inverter parameters P0700, P1000 and P2014 are saved in the instance DB and the FB sets the USS_selected Boolean parameter to false. If P0700 and P1000 both have the value 5, the FB USS_selected is set to true. If, in this case, the current value of P2014 differs from the configured value then the configured value is written in the inverter. At the end the FB delivers a DONE pulse as done message.
- **Activate**
  When enabling, it is checked first whether the USS communication has already been selected, meaning whether the three parameter values from the above table are already active in the inverter. For this purpose, the current parameters are read out from the inverter. The current inverter parameters P0700, P1000 and P2014 are saved in the instance DB if one of the parameters P0700 and P1000 does not have the value 5. Subsequently, all three parameters are written into the inverter according to Table 5-4. The FB sets the “USS_selected” Boolean parameter to true and delivers a DONE pulse as done message.
- **Deactivate**
  When disabling, it is checked first whether the USS communication has already been selected, meaning whether the three parameter values from the above table are already enabled in the inverter. For this purpose, the current parameters are read out from the inverter. The current inverter parameters P0700, P1000 and P2014 are saved in the instance DB if one of the parameters P0700 and P1000 does not have the value 5. Subsequently the parameters P0700, P1000 and P2014, saved in the instance DB, are written into the inverter in order to re-establish the status before enabling the USS communication[23]. The FB delivers a DONE pulse as done message.

RW_PARAM

With this, you can read or write any SINAMICS V20 parameter. Access to parameter numbers >2047 through the usual entry of parameter numbers and index is also possible.

You can select between the following functions that can be selected with the RW_PARAM.FUNCTION FB parameter:

- **Read**
  The value of the inverter parameter is read into a variable that corresponds to the RW_PARAM_VALUE_OUT (OUT parameter) structure in

[23] If the controller cannot get the previous connection status from the inverter, in the event of an enabling or disabling action – e.g. after a restart – because it has already been configured for USS communication (P0700 = P1000 = 5), the next disabling is carried out with the default values P0700=1, P1000=1 and P2014=2000.

V20_USS_Param_DB_1. Enter the data type of the inverter parameter in the RW_PARAM.FORMAT FB input parameter.

- **Write**
  The value of the inverter parameter is retrieved from a variable that corresponds to the RW_PARAM_VALUE_IN (IN parameter) structure in V20_USS_Param_1_DB. Enter the data type of the inverter parameter in the RW_PARAM.FORMAT FB input parameter. A further FB input parameter specifies whether you want to write to the RAM or EEPROM of the inverter.
- **Write&read**
  Combination of the above two functions. The inverter parameter is written into the SINAMICS V20 and subsequently read out straight away for control purposes.

ACTUAL_STATE

With this, you read out a set of current inverter values from the SINAMICS V20 that are represented by parameters. The inverter values involved are specified in chapter 1.2. All data read out from the V20 is united in the ACTUAL_STATE_VALUES_OUT (Struct) FB output parameter.

SET_FUNCTIONS

With this, you can select or switch the inverter functions on and off, that are stored in the inverter parameters, via the USS communication. The inverter functions involved are explained in chapter 1.2.

You can select between the following functions that can be selected with the SET_FUNCTIONS.FUNCTION FB parameter:

- **Read**
  A defined set of inverter functions is read out from the inverter in the SET_FUNCTIONS_VALUES_OUT FB output parameter in V20_USS_Param_1_DB.
- **Write**
  A defined set of inverter functions is written from the SET_FUNCTIONS_VALUES_IN FB input parameter in V20_USS_Param_DB_1 to the inverter. A further FB input parameter specifies whether you want to write to the RAM or EEPROM of the inverter.
- **Write&read**
  Combination of the above two functions. The set of inverter parameter is written into the SINAMICS V20 and subsequently read out again straight away for control purposes.

5.5.2 Parameterization

The configuration is either performed by supplying the formal parameters for the FB call or by directly accessing the parameters in the “Input” or “Output” data segments of the appropriate instance DB. The line comment of one of the IN parameters to be written by the user starts with “?“ that of a OUT parameter to be read with “!“. Variables with higher address offset than those in the table below, meaning variables in the data segment of the statistic data that are further down in the DB, must not be changed by the user.

Figure 5-5: Call of V20_USS_Param [FB2]

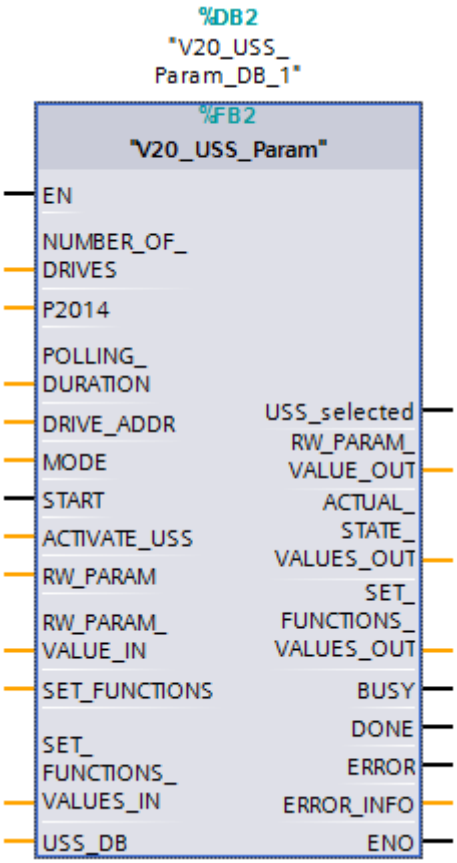


Table 5-5: Parameter of V20_USS_Param [FB2]

Name	IN/ OUT	Type	Explanation
NUMBER_OF_DRIVES	IN	USInt	**Number of drives on the port** Possible values: 1…16
P2014	IN	UInt	**Frame time out [ms]** This parameter, multiplied with the number of drives on the port is transferred to the selected drive in the ACIVATE_USS mode when enabling the USS communication.
POLLING_DURATION	IN	Time	**Polling duration** In the acyclic communication, “DONE” does not necessarily indicate that the communication channel is free again for another parameter transmission. This is why the next parameter transfer may be rejected with status 818A until the communication channel is free again. Only when this time exceeds the POLLING_DURATION value, is the 818A status displayed as error on the HMI.

Name	IN/ OUT	Type	Explanation
MODE	IN	USInt	**Mode of the block** 1 = ACTIVATE_USS 2 = RW_PARAM 3 = ACTUAL_STATE 4 = SET_FUNCTION Detailed contents of the various modes, see chapter 1.2.
START	IN	Bool	**Function start** The function selected is executed by START with a positive edge.
ACTIVATE_USS FUNCTION	 IN	Struct USInt	**Enabling USS** Function within the mode 0 = Deactivate 1 = Activate 2 = Check Details can be found in the description of the operating mode in this chapter.
RW_PARAM. FORMAT	 IN	Struct USInt	**Reading/writing parameters** Data type 3 = Int, I16, 16-bit integer 4 = DInt, I32, 32-bit integer 6 = UInt, U16, 16-bit unsigned 7 = UDInt, U32, 32-bit unsigned 8 = Real, Float, 32-bit floating point number Specifying a data type is required because PLC does not know what data format the inverter parameter to be read or written has. For the data type see the parameter list in the SINAMICS V20 manual /8/.
PARAM	IN	UInt	Parameter number The number can be found in the SINAMICS V20 operating instruction /8/.
INDEX	IN	UInt	Parameter index The index can be found in the SINAMICS V20 operating instruction /8/.
EEPROM	IN	Bool	Writing to EEPROM false = The written value is only temporarily saved and gets lost the next time the SINAMICS V20 is switched on. true = The value written in the drive parameter is saved in the EEPROM of the SINAMICS V20.
FUNCTION	IN	USInt	Function within the mode 0 = Reading parameters 1 = Writing parameters 2 = Writing & reading parameters Details can be found in the description of the operating mode in this chapter.

Name	IN/ OUT	Type	Explanation
RW_PARAM_VALUE_IN. Int DInt USInt UInt UDInt Real	IN	Struct Int DInt USInt UInt UDInt Real	**Parameter value to be written** The inverter parameter value to be written is to be stored in the FB formal parameter whose data format corresponds to the specification in the RW_PARAM.FORMAT FB input parameter.
SET_FUNCTIONS EEPROM	IN	Struct Bool	**Inverter functions** Writing parameter set to EEPROM false = The written set of inverter parameters is only temporarily saved and gets lost the next time the SINAMICS V20 is switched on. true = The inverter set of parameters written in the set of drive parameters is saved in the EEPROM of the SINAMICS V20.
FUNCTION	IN	USInt	Function within the mode 0 = Reading set of parameters 1 = Writing set of parameters 2 = Writing & reading set of parameters Details can be found in the description of the operating mode in this chapter.
SET_FUNCTIONS_VALUES_IN. Keep_running_Op[n] Flying_start Automatic_restart Hibernation[n] Super_torque[n]	IN/ OUT	Struct UInt UInt UInt UInt UInt	**Function values (IN)** Continuous operation P0503[n] Flying Start P1200 Automatic restart P1210 Energy saving mode P2365[n] Torque pulse mode P3350[n] n=0..2 indicates the drive data set Information on the meaning and value range of the functions can be found in the SINAMICS V20 operating instruction /8/.
USS_DB	IN/ OUT	P2P_ USS_ BASE	**Structure variable USS_DB** Interconnect the parameter with… *"V20_USS_Control_DB_n". USS_Drive_Control_Instance.USS_DB* …here, *V20_USS_Control_DB_n* is the instance DB of the respective port for FB V20_USS_Control.
USS_selected	OUT	Bool	**Enabling USS – event bit** false = USS communication not enabled true = USS communication enabled The bit is only valid in the cycle in which the DONE message is pending in a function started with MODE=1.
RW_PARAM_VALUE_OUT. Int DInt USInt UInt UDInt Real	OUT	Struct Int DInt USInt UInt UDInt Real	**Read parameter value** The read inverter parameter value is stored in the FB formal parameter whose data format corresponds to the specification in the RW_PARAM.FORMAT FB input parameter.

Name	IN/ OUT	Type	Explanation
ACTUAL_STATE_VALUES_OUT. FREQ_OUTPUT OUTPUT_VOLTS CURRENT DC_BUS_VOLTS TOTAL_FREQ_SP ENERGY_SAVED[n] CONNECTION_MACRO APPLICATION_MACRO	OUT	Struct Real Real Real Real Real Real UInt UInt	**Actual values** Act. filtered output frequency r0024 [Hz] Act. output voltage r0025 [V] Act. output current r0027 [A] Act. smoothed link voltage r0026 [V] Act. total setpoint r1078 [Hz] Act. energy saving r0043[0..2] [kWh,€,CO2] connection Macro last selected p0717 user Macro last selected p0507 Further information on the meaning and value range of the parameters can be found in the SINAMICS V20 operating instruction /8/.
SET_FUNCTIONS_VALUES_OUT. Keep_running_Op Flying_start Automatic_restart Hibernation[n] Super_torque[n]	OUT	Struct UInt UInt UInt UInt UInt	**Function values (OUT)** Continuous operation P0503[n] Flying Start P1200 Automatic restart P1210 Energy saving mode P2365[n] Torque pulse mode P3350[n] n=0..2 indicates the drive data set Information on the meaning and value range of the functions can be found in the SINAMICS V20 operating instruction /8/.
BUSY	OUT	Bool	**Block in process** Processing the block requires several cycles. false = not in process true = in process As long as BUSY is pending, the parameters pending in the parameter bar to be transferred or the transferred parameters must not be changed.
DONE	OUT	Bool	**Done message** When the processing of the block is completed without errors after its start (with START=true), a ready message will be output in form of a DONE pulse for the duration of one cycle. Afterwards, the parameters read by the converter may be removed and the function can be restarted with a restart of the START edge.
ERROR	OUT	Bool	**Error message** When the processing of the block is completed with errors after its start (with START=true), an error message will be output in form of an ERROR pulse for the duration of one cycle. Afterwards the error information stored in the ERROR_INFO output parameter can be read out and a respective error response can be initiated. With the error pulse is ready again for a new start command.
ERROR_INFO. DRIVE_ADDR	 OUT	Struct USInt	**Error information** Address of the SINAMICS V20 where the error occurred.
STATUS	OUT	Word	Error information of the instructions USS_Read_Param and USS_Write_Param. Details on this can be found in the Online help in the TIA Portal.

Name	IN/ OUT	Type	Explanation
EXTENDED_ERROR_available	OUT	Bool	For .STATUS=818C$_{hex}$ the instructions USS_RPM and USS_WPM store an expanded error information in the USS_DRV_DB_1 system DB that is provided to the user as EXTENDED_ERROR output parameter of the V20_USS_Param_1 FB. false = no .EXTENDED_ERROR available true = .EXTENDED_ERROR available The .EXTENDED_ERROR may only be evaluated by the user if .EXTENDED_ERROR_available is true.
EXTENDED_ERROR	OUT	UInt	For .EXTENDED_ERROR_available =true available additional error information. The error code can be found in chap. 6.1 of the SINAMICS V20 operating instruction /8/ (table: Error numbers in response ID 7 (request cannot be processed).
MODE		USInt	Mode of the block at which the error occurred.
RW		Bool	false = read error true = write error
PARAM		UInt	Number of the parameter at which the error occurred.
INDEX		UInt	Index of the parameter at which the error occurred.

5.5.3 Start of a function

Table 5-6: Start of a function

	Instruction
1.	Setting the MODE FB parameter to desired value.
2.	Setting the FUNCTION FB parameter to the desired value in that structure that corresponds to the selected MODE.
3.	Depending on the MODE, if required, provide other FB input parameters that specify the function.
4.	Depending on MODE, if required, provide source data (parameter values) on the appropriate FB parameter.
5.	Create positive edge on the START FB parameter. (This can take place in the same cycle as the previous points.)
6.	For done message by DONE FB parameter: • Depending on MODE, if required, remove target data (parameter values) on the respective FB parameter. • Ready for other parameter operation

6 Time Behavior

6.1 Call intervals of the USS system function blocks

USS_Port_Scan calls

The USS_Port_Scan instruction can be called from any OB. It is usually called from a cyclic interrupt OB in order to guarantee a constant time behavior when mailing the frames. If the execution of the USS_Port_Scan is interrupted by another instruction, unexpected errors may occur.

If the frame exchange to a drive is performed without repeated attempts, the following time requirement can be roughly estimated:

$$t_{comm} = 5^{*} \cdot t_{cycle} + t_{send} + t_{rcv}$$

t_{comm} *= time requirement of an error-free communication to a drive*

t_{cycle} *= = call interval of the cyclic interrupt OB*

t_{send} *= runtime of the send frame*

t_{rcv} *= runtime of the receive frame*

* At least 5 cycles are required for the sending+receiving of frames.

The above formula applies for the frame runtimes:

$$t_{send} = t_{rcv} = n_{char} \cdot \frac{11^{*}}{v_{data}}$$

n_{char} *= number of characters per frame (frame structure see \|9\|)*

v_{data} *= data transmission speed (baud rate in bit/s)*

* *11 bits per character (USS is one 11bit protocol)*

Thus, the smaller you select the call interval of the cyclic interrupt OB, by calling USS_Port_Scan, the faster your communication.However, if the call interval is too small, the communication requirement can no longer be satisfied and you receive error message 8281H “Negative acknowledgement when writing the module”.

Example:
For the application example with the following values …

t_{cycle} *= 10ms*

n_{char} *= 16 (for 2 PZD words)*

v_{data} *= 38400 bit/s*

…a time requirement of $t\_comm = 59ms$ for the communication to a drive is the result.

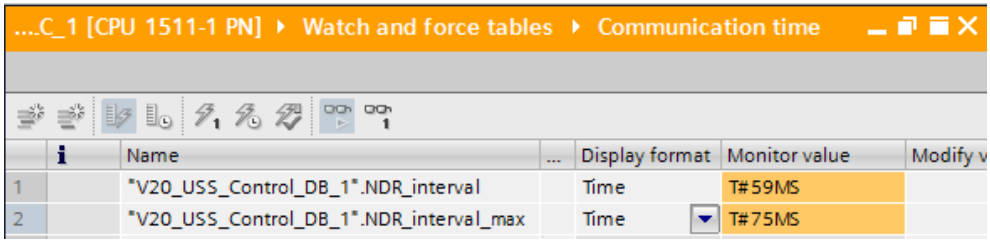
Calls USS_Drive_Control, USS_Read_Param, USS_Write_Param

Only call USS_Drive_Control, USS_Read_Param and USS_Write_Param from a cycle OB of the main program. Do not use this instruction in an OB with a higher priority than the respective USS_Port_Scan instruction. For example, do not add USS_Port_Scan into the main program and USS_Read_Param into a cyclic interrupt OB.

6.2 Measuring the communication time

The NDR output bit parameter of the USS_Drive_Control system function block changes every time from FALSE to TRUE if the data is provided in a new communication request. In network 7 of FB V20_USS_Control the time distance of the positive NDR edges is measured (NDR_interval variable). In addition, the maximum value of all measured values is saved. Both values can be viewed in the application example in the “Communication time” watch table.

Table 6-1: Communication time



The value of the NDR_interval variable corresponds to the calculated value from the example in chap. 6.1.

If NDR_interval_max is significantly larger than NDR_interval then the respective drive has not received the communication request of the controller or did not understand it and it will be repeated[24] up to two times. You can overwrite NDR_interval_max again and again with 0ms in the watch table. The frequency/rarity with which a time larger than NDR_interval occurs, is a measurement for the quality of the communication.

6.3 Frame time out of the SINAMICS V20

As most drives with USS communication, the SINAMICS V20 also has a configurable frame time out (P2014). If the drive does not receive a communication request within a certain period of time, it goes to fault with error F72. The frame time out to be configured on the drive should not be below the following value – in order to guarantee fault-free operation:

$$t_{out-min} = n_{drives} \cdot \left(5^{*} \cdot t_{cycle} + t_{send} + t_{timeout}\right) \cdot n_{send}$$

$t_{out-min}$ = *smallest frame time out*
n_{drives} = *number of drives on the port*
t_{cycle} = *call interval of the cyclic interrupt OB*
t_{send} = *runtime of the send frame*
$t_{timeout}$ = *time out of the receipt frame*
n_{send} = *number of send attempts*
* At least 5 cycles are required for the sending+receiving of frames.

The time out of the receive frame $t_{timeout}$ depends on the baud rate and is made up of two times that are stored in the instance data of the FB USS_Port_Scan:

[24] Default setting in the instance data of USS_Port_Scan, RETRIES_MAX variable.

Table 6-1: Time out times of the receipt frame

Baud rate [bit/s]	Receive_Conditions.END.RCVTIME	Receive_Conditions.END.MSGTIME
115200	25	25
57600	29	29
38400	33	33
19200	56	56
9600	72	72
4800	100	124
2400	100	240
1200	100	460

Case1:

If no reply arrives as a response to the send frame by the drive within the RCVTIME, the send frame is repeated or if the configured number of repeat attempts has been reached, it is passed on to the next drive. Therefore, the following would apply: $t_{timeout}$ = RCVTIME

Case2:

If an incomplete or faulty reply arrives as a response to the send frame by the drive shortly before the lapse of RCVTIME, it is waited for the MSGTIME and the send frame is repeated or if the configured number of repeat attempts has been reached, it is passed on to the next drive. Therefore, the following would apply: $t_{timeout}$ = RCVTIME + MSGTIME. This presents the worse case.

Example:
For the application example with the following values …

$n_{drives} = 1$

$t_{cycle} = 10ms$

$n_{char} = 16$ *(2 PZD words)*

$v_{data} = 38400$ *bit/s*

$t_{timeout} = 66ms$ *(RCVTIME + MSGTIME at 38400 bit/s from Table 6-1)*

…a configurable WorstCase frame time out in SINAMICS V20
$t_{out-min} = 362ms$ for the communication to a drive is the result. For several drives, the value would have to be multiplied with its number.

NOTICE	**The selection of the frame time out depends on the security requirements of your application. From the above calculated estimate, a relatively large value results which ensures that switching off the inverter is largely avoided due to the frame time out.** **A more sensitive behavior is achieved when you orientate yourself on the NDR_interval_max value measured from chap. 6.2.** **Under no circumstances disable the frame time out by configuring with 0ms. A running drive could then no longer be switched off in the event of a communication failure.**

7 Expansion to Several Drives

7.1 Expansion to up to 16 drives per port

The application example operates one SINAMICS V20. However, via a port up to 16 drives can be operated. This has already been taken into consideration in the application example and especially in the FBs V20_USS_Control [FB1] and V20_USS_Param [FB2]. To increase the number of drives, proceed as follows:

Table 7-1: Expansion to up to 16 drives

	Instruction
1.	Supplement your configuration by the number of the desired drives according to Figure 2-1.
2.	Configure the added SINAMICS V20 according to points 1 to 8 of Table 3-2 via the incorporated BOP. From “2” onward, the drive addresses have to be assigned continuously.
3.	Enter the new number of drives as start value in the IN_Number_of_drives variable in the static data of the V20_USS_Control_1 [DB1], load the DB into the CPU and restart it.

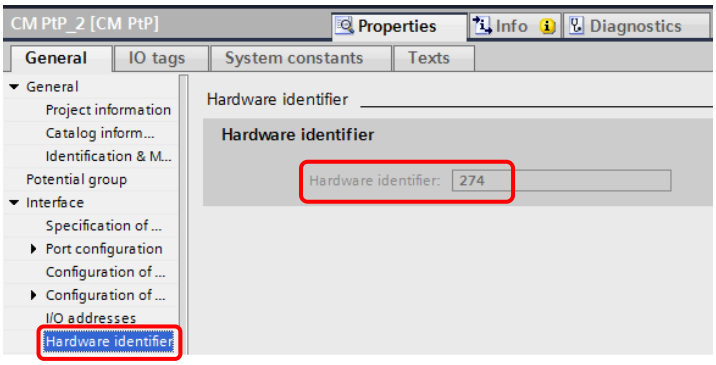
By entering the drive address in the header of the respective operating screen, you can select the drive for your monitoring or operation.

7.2 Expansion to several ports

7.2.1 Expansion of hardware to several CM PtP per ET 200SP

Table 7-2: Expansion of ET 200SP to several ports

No.	Instruction
1.	Go to step 4**Fehler! Verweisquelle konnte nicht gefunden werden.** according to Table 5-1 in the device view of that ET 200SP, into which you want to add one or several PtP communication modules.
2.	Slide the server module with the mouse to the right by the number of CMs to be added.
3.	Copy the already existing CM to the places/place that have/has become free or add the CM_PtP from the catalog as in Table 5-1 step 4.
4.	Place the input addresses to a free area according to Table 5-1 step 5 .

No.	Instruction
5.	Remember the hardware identifier of the newly added CM.  The correlation between the code blocks of the user program and the respective communication module is only made via the hardware identifier. It is to be entered into the respective V20_USS_Control_DB_n instance DB in the IN_HW_Id variable.
6.	To expand the user program, continue with chap. 7.2.3.

7.2.2 Expansion of the hardware to several ET 200SP stations

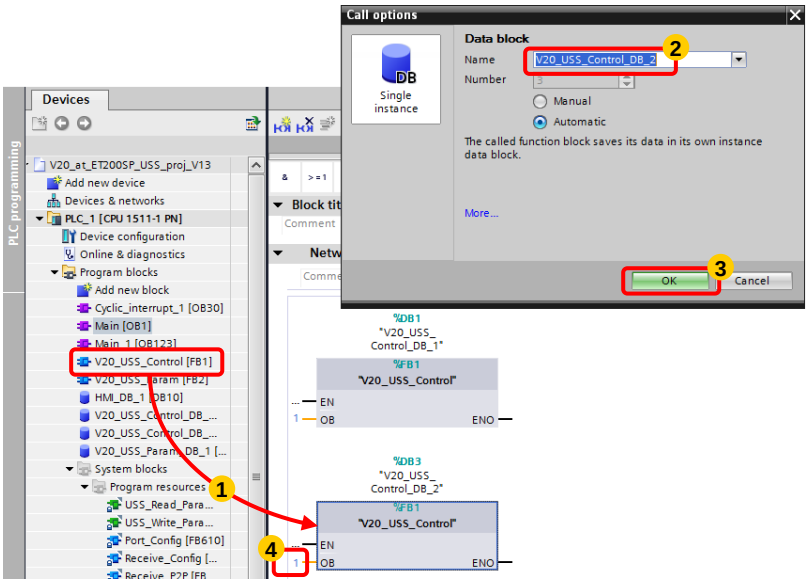
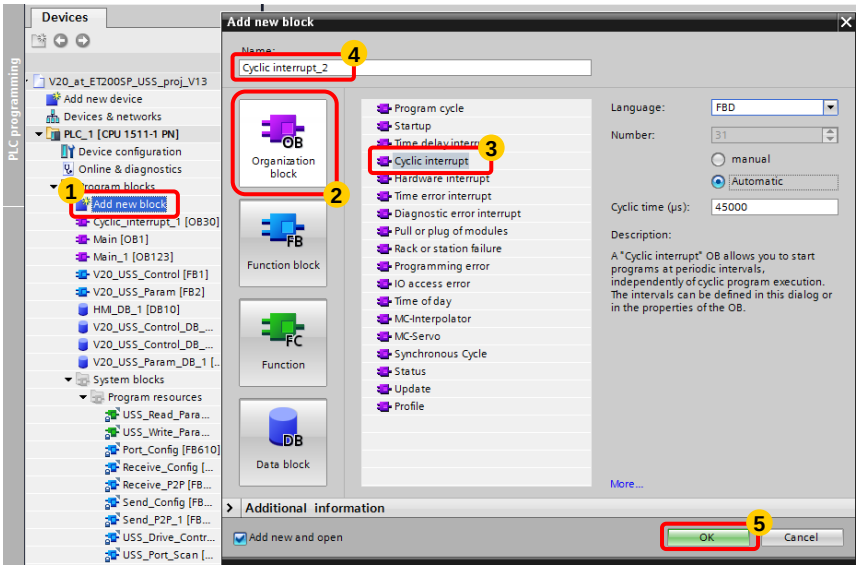
Table 7-3: Expansion to several ET 200SP stations

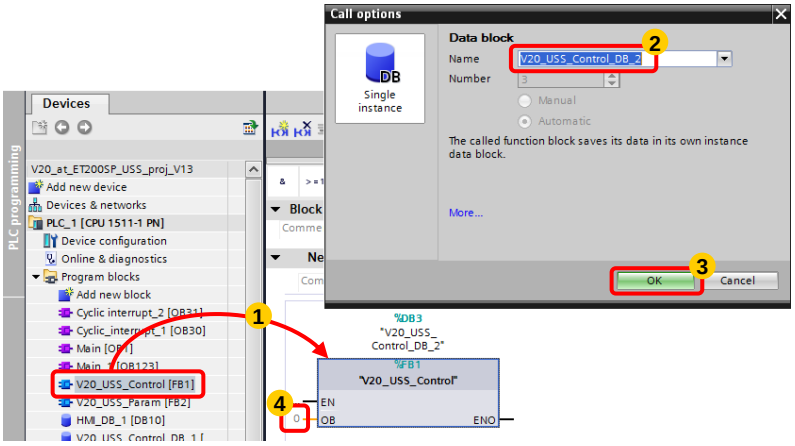
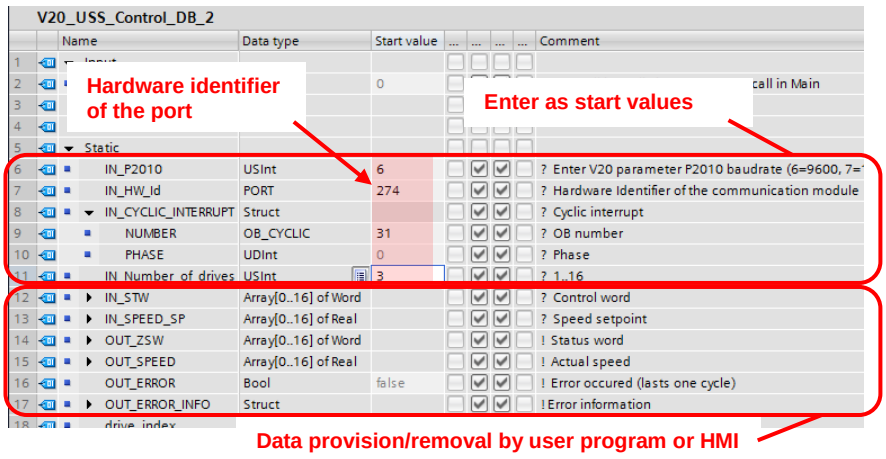
No.	Instruction
1.	Correspondingly, proceed as described in Table 5-1 steps 2 to 7. If you want to add several communication modules into the appropriate ET 200SP, proceed according to Table 7-2.
2.	To expand the user program, continue with chap. 7.2.3.

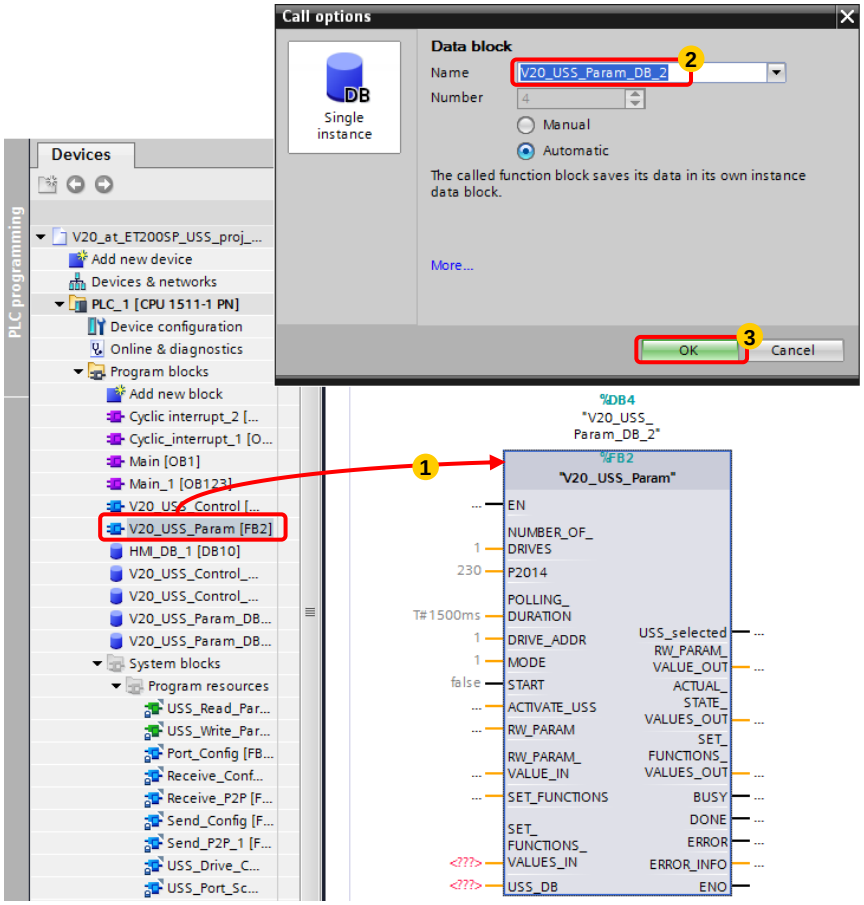
7.2.3 Expansion of the user program to several ports

This way, the distribution of the CM PtP communication module to several ET 200SP stations is irrelevant for the user program. For addressing it is solely the hardware identifier of the respective CM that is decisive. The instruction of the hardware expansion according to chap. 7.2.1 as well as of chap. 7.2.2 leads to the expansion of the user program according to chap. 7.2.3.

Table 7-4: Expansion of the user program by a port

No.	Instruction
1.	1. Drag the call of V20_USS_Control [FB1] from the project navigation into the cyclic part of your user program.  2. Change the name of the created instance DB to your wishes (e.g. to V20_USS_Control_DB_2). 3. Close the window „Call options“ with OK. 4. Assign a 1 to the OB input parameter (see Table 5-2).
2.	1. Add a new cyclic interrupt OB.  2. Select an organization block. 3. Select a cyclic interrupt. 4. Modify the block’s name in accordance with your wishes. 5. Close the window „Add new block“ with OK. The required time cycle is set by the user program (see IN_CYCLIC_INTERRUPT.CYCLE input parameter of the FBs V20_USS_Control in Table 5-2). In the above “Add new block” window you do not have to make any entries in this case.

No.	Instruction
3.	1. Drag the call of V20_USS_Control [FB1] from the project navigation into the just created cyclic interrupt OB.  2. Assign the call the instance data of the new instance. 3. Close the window „Call options“ with OK. 4. Assign the OB input parameter a value ≠1 or leave the default setting 0 (see Table 5-2).
4.	Configure the new instance by supplying their static variables, which start with "?" (input parameter) and by removing those that start with "!" (output parameter).  Input parameters that can no longer be changed during runtime can be directly entered as start values in the DB. The other parameters have to be provided or evaluated by your user program or the HMI.

No.	Instruction
5.	1. Drag the call of V20_USS_Param [FB2] from the project navigation into the cyclic part of your user program.  2. Change the name of the created instance DB to your wishes (e.g. to V20_USS_Param_DB_2). 3. Close the window „Call options“ with OK.
6.	Supply and remove the formal parameters of the just created FB call (see Table 5-5). The reference to the newly installed port is established via the USS_DB parameter. Interconnect it with the static variable with the same name from the USS_Drive_Control instance data of the new port. If the V20_USS_Control_DB_2 instance DB was created for the new port, then this actual parameter "V20_USS_Control_DB_2".USS_Drive_Control_Instance.USS_DB has to be connected to the USS_DB formal parameter of FB V20_USS_Param.

Note

For ports with identical baud rates the FBs V20_USS_Control may possibly be called in a joint cyclic interrupt OB. Creating a new cyclic interrupt OB would be superfluous in this case. However, you can only achieve a more balanced time behavior by using separate cyclic interrupt OBs with the same clock cycle but different phasing even for identical baud rates (see Parameter description in Table 5-2).

8 Literature

Table 8-1

	Topic	Title
\1\	Siemens Industry Online Support	http://support.automation.siemens.com
\2\	Download page of the entry	http://support.automation.siemens.com/WW/view/en/90468030
\3\	SIMATIC S7-1500	SIMATIC S7-1500 / ET 200MP Library http://support.automation.siemens.com/WW/view/en/86140384
\4\		Automating with SIMATIC S7-1500 Author: Hans Berger Published by: Publicis Publishing ISBN: 978-3-89578-404-0
\5\	ET 200SP	SIMATIC ET 200SP Library http://support.automation.siemens.com/WW/view/en/84133942
\6\	TIA Portal	Updates for STEP 7 V12 SP1 http://support.automation.siemens.com/WW/view/en/78683919
\7\		Updates for STEP 7 V13 http://support.automation.siemens.com/WW/view/en/90466591
\8\	SINAMICS V20	SINAMICS V20 Inverter – operating instruction http://support.automation.siemens.com/WW/view/en/67267484
\9\	USS® Bus	Universal serial interface protocol (USS® protocol) http://support.automation.siemens.com/WW/view/en/24178253
\10\	HMI	SIMATIC HMI control panels Basic Panels http://support.automation.siemens.com/WW/view/en/31032678

9 History

Table 9-1

Version	Date	Changes
V1.0	05/2014	First version
V1.0.1	08/2019	Corrected wiring of the potential equalization cable in chapter 2.1.3