Training Document for Comprehensive Automation Solutions

Totally Integrated Automation (T I A)

MODULE E05

PROFINET with

IO-Controller CP343-1 Advanced

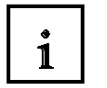
and

IO-Device ET 200S

We would like to thank the following: Michael Dziallas Engineering, the teachers at vocational schools, and all others who helped to prepare this document.

PAGE

The following symbols are provided as a guide through Module E5:

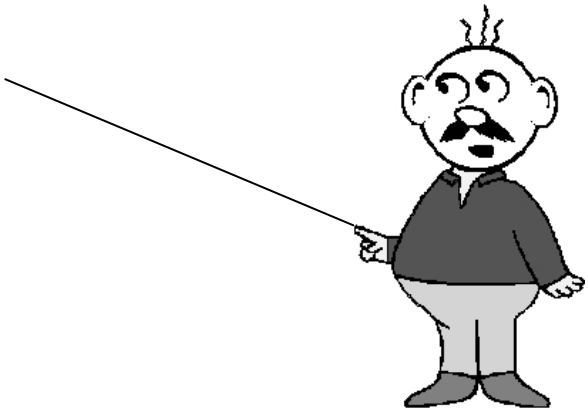


Information

Programming

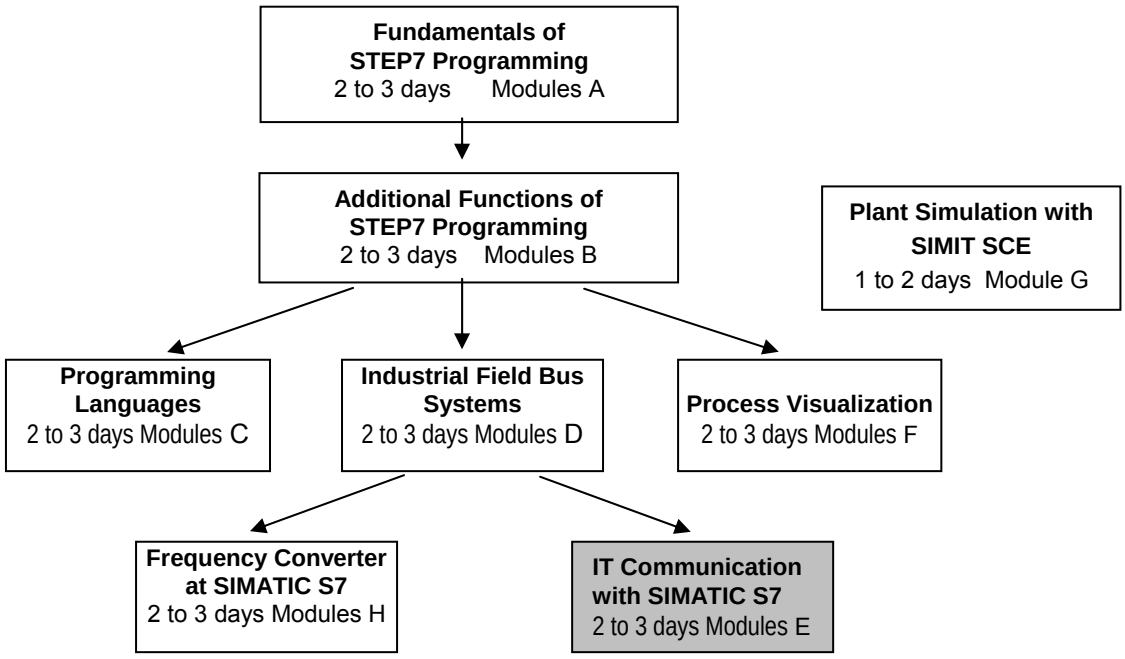
Sample Exercise

Notes



1. PREFACE

In terms of its contents, Module E05 is part of the teaching unit entitled **'IT Communication with SIMATIC S7'**.



Learning Objective:

In Module E05, the reader learns how PROFINET is started up with the CP343-1 Advanced as IO controller and the ET200S as IO device. Module E05 demonstrates the method in principle, using a short example.

Prerequisites:

To successfully work through Module E05, the following knowledge is assumed:

- Knowledge in handling Windows
- Fundamentals of PLC programming with STEP 7 (for example, Module A3 - 'Startup' PLC Programming with STEP 7)
- Fundamentals of network engineering (for example, Appendix V - Basics of Network Engineering)

Hardware and software required

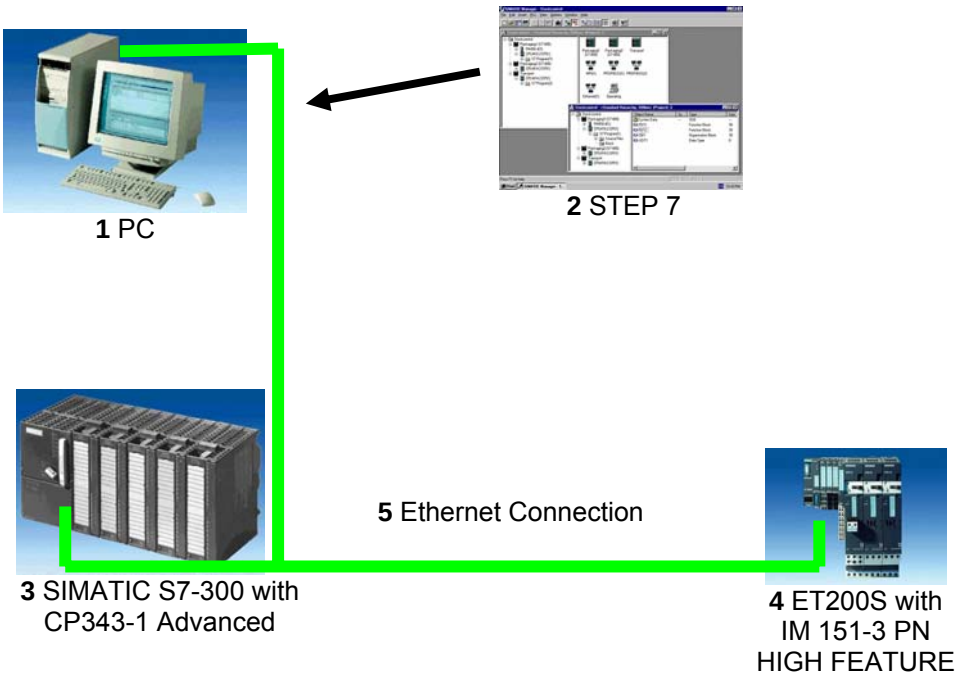
1 PC, operating system Windows 2000 Professional starting with SP4/XP Professional starting with SP1/Server 2003 with 600MHz and 512RAM, free hard disk storage 650 to 900 MB, MS Internet Explorer 6.0 and network card

2 Software STEP7 V 5.4

3 PLC SIMATIC S7-300 with CP343-1 Advanced
Sample configuration:
- Power supply: PS 307 2A
- CPU: CPU 314C-2DP
- CP: CP343-1 Advanced

4 Distributed periphery ET 200S for PROFINET with 2 digital inputs and 4 digital outputs
Sample Configuration:
- Interface Module IM151-3 PN HIGH FEATURE (HF)
- Power module PM-E DC 24V …48V/AC 24V…230V
- Electronic module: 2DI Standard DC 24V
- Electronic module: 4DO Standard DC 24V/0.5A

5 Ethernet connection between PC, CPU 343-1 Advanced and ET200S with IM 151-3 PN HF



2. NOTES ON USING THE CP343-1 ADVANCED

The CP343-1 Advanced is the communication module of the SIMATIC S7-300 for Industrial Ethernet with PROFINET functionality.
Having its own processor, it relieves the CPU of communication tasks, and makes additional connections possible.
The parameters are assigned and the Industrial Ethernet and PROFINET are configured with the Software STEP7. This provides the user with a uniform configuring tool for central as well as distributed configurations.

The CP343-1 Advanced allows for connecting the SIMATIC S7-300 to the Industrial Ethernet with the following capabilities:

- 10/100 Mbit/s Full/Half duplex connection with auto-sensing for automatic switching
- Connection by means of RJ45
- TCP/UDP transport protocol
- PG/OP communication
- Network overreaching PG/OP communication through S7 routing
- S7 Communication
- S5 compatible communication
- Multicast for UDP
- IT communication:

  HTTP communication permits access to process data using the Web browser; FTP communication allows for program-controlled FTP client communication, access to data blocks by means of FTP servers, data handling of own data system by means of FTP, Email.
- Remote programming and initial startup by means of the network
- IP address assignment by means of DHCP, simple PC tool, or programming block with HMI
- Access protection based on IP addresses
- Time of day synchronization of the CPU by means of NTP or SIMATIC procedure
- Integration into network management systems by the support of SNMP

With programmed FC block calls, the user program triggers the transfer of data areas for communication, and successful execution is monitored. The FC blocks needed for communication are stored in the library **"SIMATIC_NET_CP"**. To use these functions, they have to be integrated in (copied to) your "own" project.

Note:
- In Module E05, the CP343-1 Advanced is used on the PROFINET as IO controller.

3. NOTES ON USING THE ET200S WITH THE IM 151-3 PN HF

The SIMATIC ET200S is a decentral IO device configured in a highly modular mode. It can be operated with different interface modules:

IM 151-1 BASIC, IM 151-1 STANDARD and **IM 151-1 FO STANDARD** for connecting a maximum of 63 IO modules (all types except PROFIsafe) to the PROFIBUS DP; alternatively, bus connection with RS 485 Sub-D connector or by means of an integrated fiber-optic connection

IM 151-1 HIGH FEATURE for connecting a maximum of 63 IO modules (all types, including clocked mode for PROFIsafe) to PROFIBUS DP; bus connection with RS485 Sub-D connector

IM 151-3 PN for connecting a maximum of 63 IO modules (all types, including the clocked mode for PROFIsafe) to PROFINET IO controllers; bus connection by means of RJ45 connector

IM 151-3 PN HF (HIGH FEATURE) for connecting a maximum of 63 IO modules (all types; including the clocked mode for PROFIsafe) to PROFINET IO controllers; bus connection with 2 x RJ45 connector

IM 151-7/F-CPU, IM 151-7/CPU or **IM 151-7/CPU FO** for connecting a maximum of 63 IO modules (all types; PROFIsafe only with IM151-7/F CPU) to PROFIBUS DP; alternatively bus connection with RS 485 Sub-D connector or by means of an integrated fiber-optic connection; with integrated CPU 314 of the SIMATIC S7-300, for preprocessing process data.

The following IO modules can be used:

Power modules for individual grouping of load and encoder supply voltages and their monitoring

Digital electronic modules for connecting digital sensors and actuators

Analog electronic modules for connecting analog sensors and actuators

Sensor module for connecting IQ sense sensors

Technology modules Electronic modules with integrated technological functions, such as counting, positioning, data exchange, etc.

Frequency converters and motor starter modules

For training purposes, an integrated system is provided, suitable for teaching many technologies

Notes:

- In Module E05, the interface module IM151-3 PN HF is used as PROFINET IO device.
- A micro memory card is required for running the IM151-3 PN HF!

4. STARTING UP PROFINET

(IO CONTROLLER CP343-1 ADVANCED/IO DEVICE ET200S)

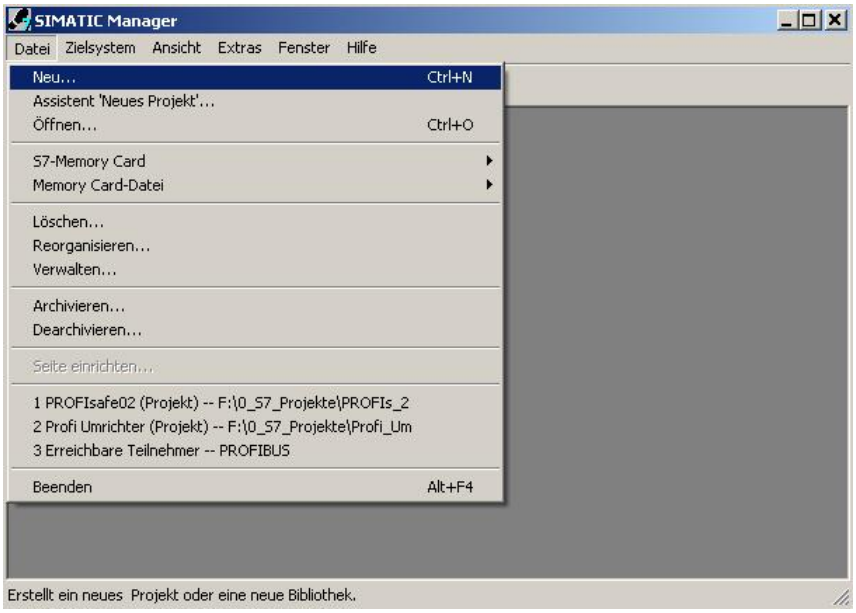
Below, the startup of a PROFINET network with the CP343-1 Advanced as IO controller and the ET200S as IO device is described.
To test the configuration, a program is written. In this program, an indicator lamp P1 is activated when two buttons, S0 and S1, are operated simultaneously.

Assignment List:

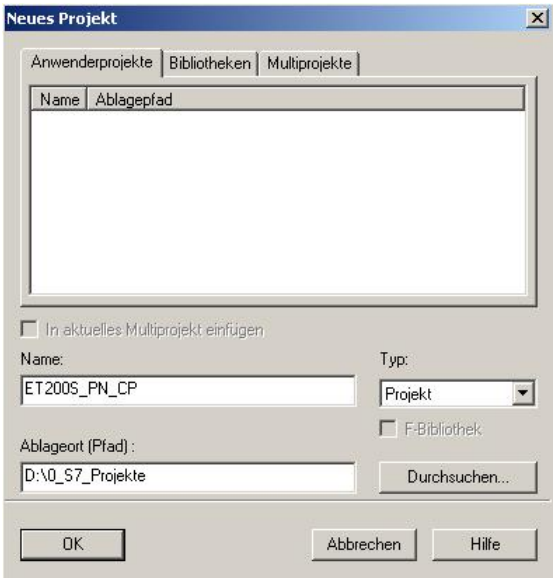
I0.0	S0	Button Selection 1
I0.1	S1	Button Selection 2
O0.0	P1	Indicator lamp

1. The central tool in STEP 7 is the **'SIMATIC Manager'**. It is called here with a double click. (→ SIMATIC Manager)

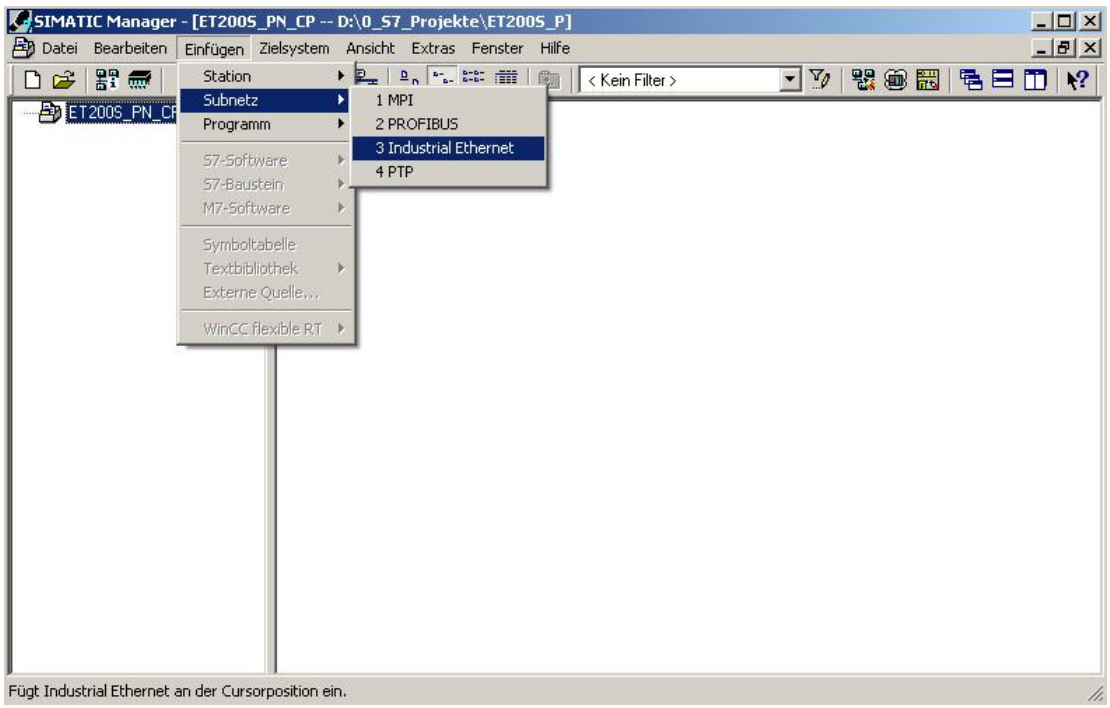
2. STEP 7 programs are managed in projects. We are now setting up such a project (→ File → New)



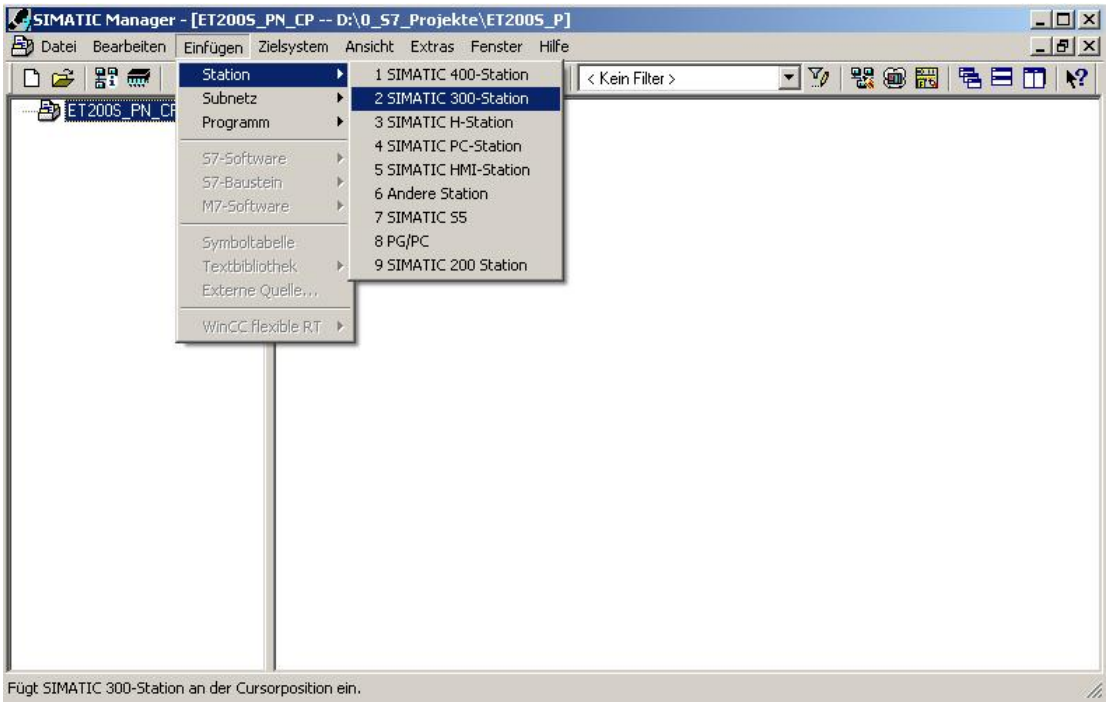
3. The project is now assigned the **'Name' 'ET200S_PN_CP'** (→ ET200S_PN_CP → OK)



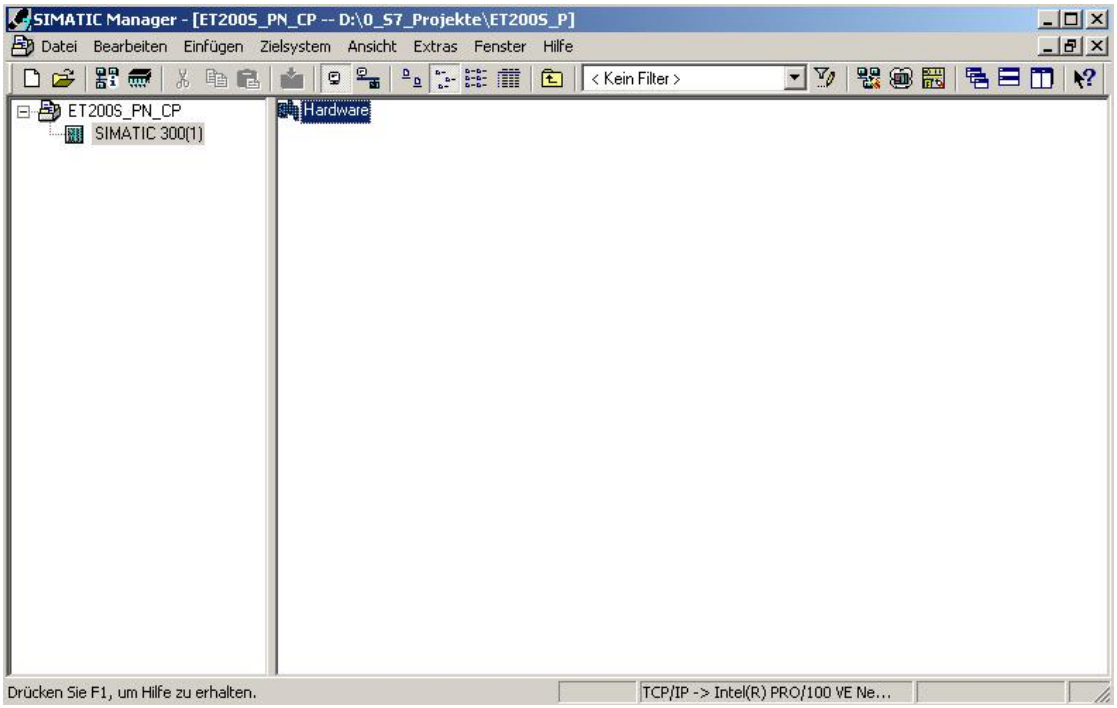
4. Highlight your project and insert an **'Industrial Ethernet Subnet'** (→ ET200S_PN_CP → Insert → Subnet → Industrial Ethernet).



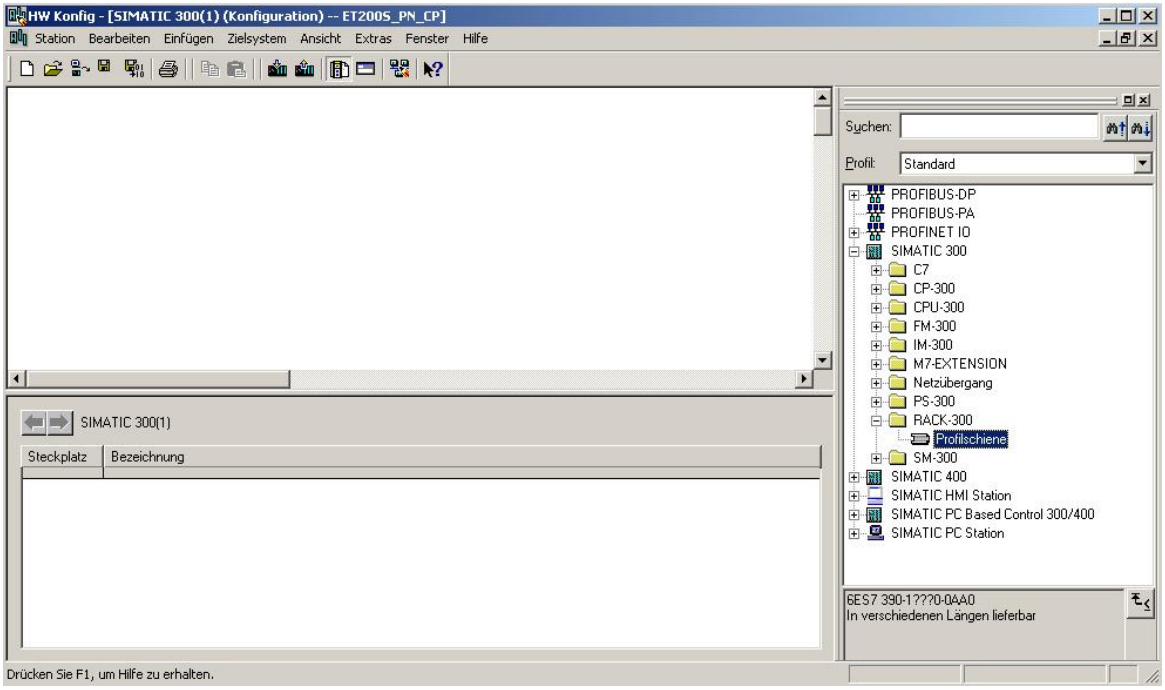
5. Then, insert a **'SIMATIC 300 Station'**. (→ Insert → Station → SIMATIC 300 Station)



6. With a double click, open the configuration tool for the **'Hardware'** (→ Hardware)

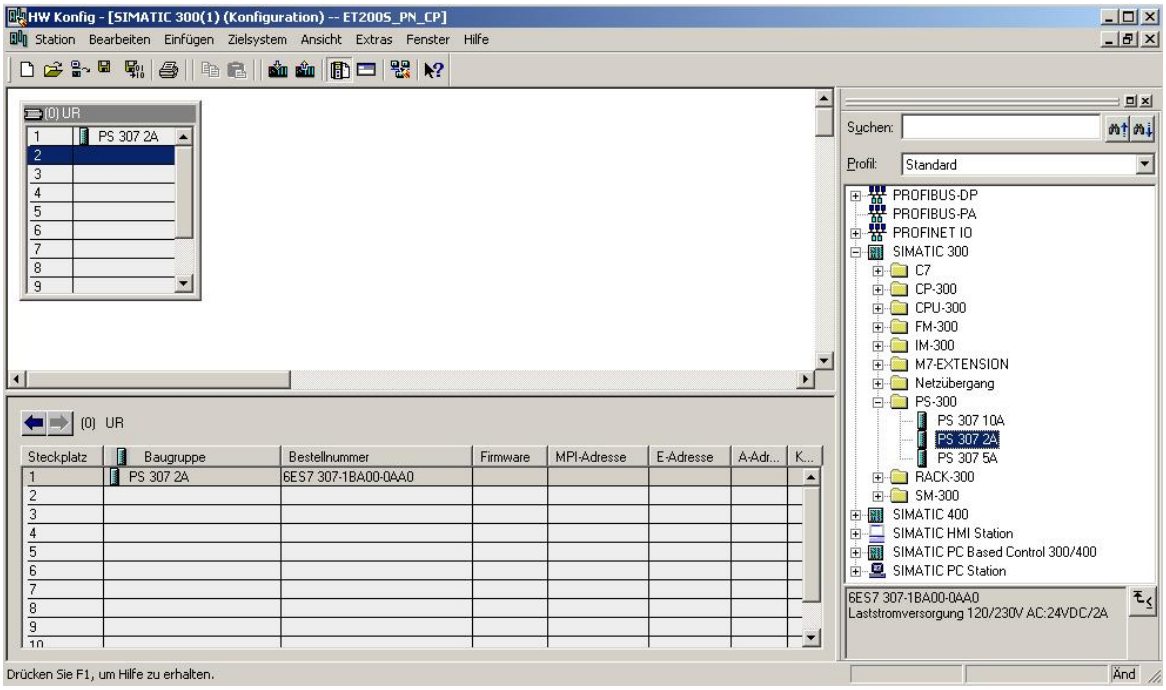


7. Open the hardware catalog by clicking on '  '. (→ )
   There, arranged in the following directories:
   PROFIBUS DP, PROFIBUS PA, PROFINET IO, SIMATIC 300, SIMATIC 400,
   SIMATIC PC Based Control, and SIMATIC PC Station,
   all racks, modules and interface modules are provided for configuring your hardware.
   Insert **'Rail'** with a double click. (→ SIMATIC 300 → RACK-300
   → Rail)



Note: After that, a configuration table is displayed automatically for configuring Rack 0.

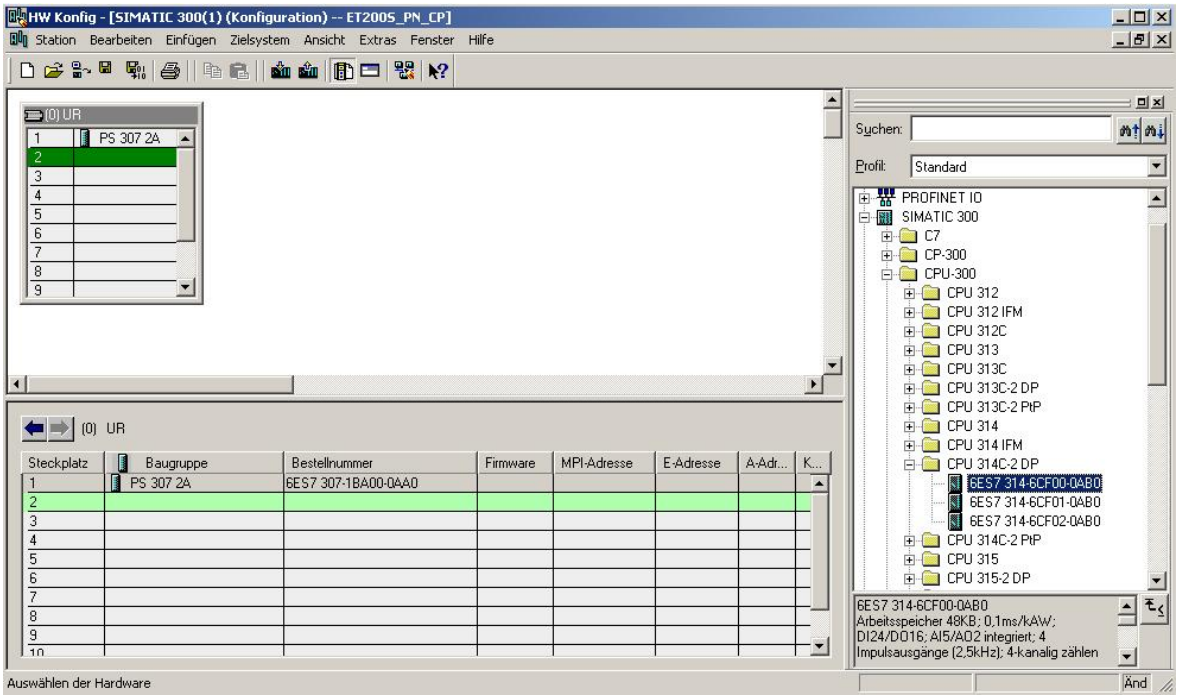
8. From the hardware catalog, you can now select all modules that are also in your real rack, and insert them in the configuration table.
   To this end, click on the name of the respective module, hold the mouse key and drag the module to a line in the configuration table.
   We are starting with the power unit **'PS 307 2A'.** (→ SIMATIC 300 → PS-300 → PS 307 2A)



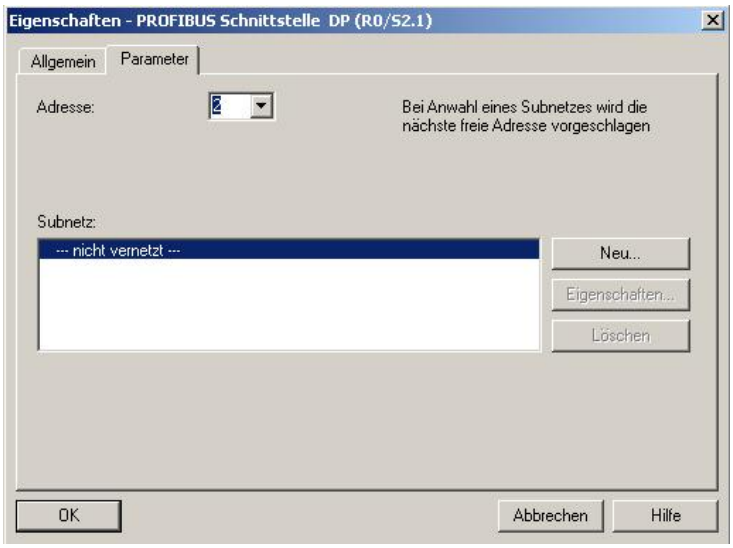
Note: If your hardware differs from the one displayed here, simply select the corresponding modules from the catalog and insert them in your rack. The order numbers of the individual modules -that are also indicated on the components- are displayed in the footer of the catalog.

.

9. Next, we drag the **'CPU314C-2 DP'** to the second slot. The order number and the version of the CPU can be read off the front of the CPU. (→ SIMATIC 300 → CPU-300 → CPU 314C-2 DP → 6ES7 314-6CF00-0AB0)

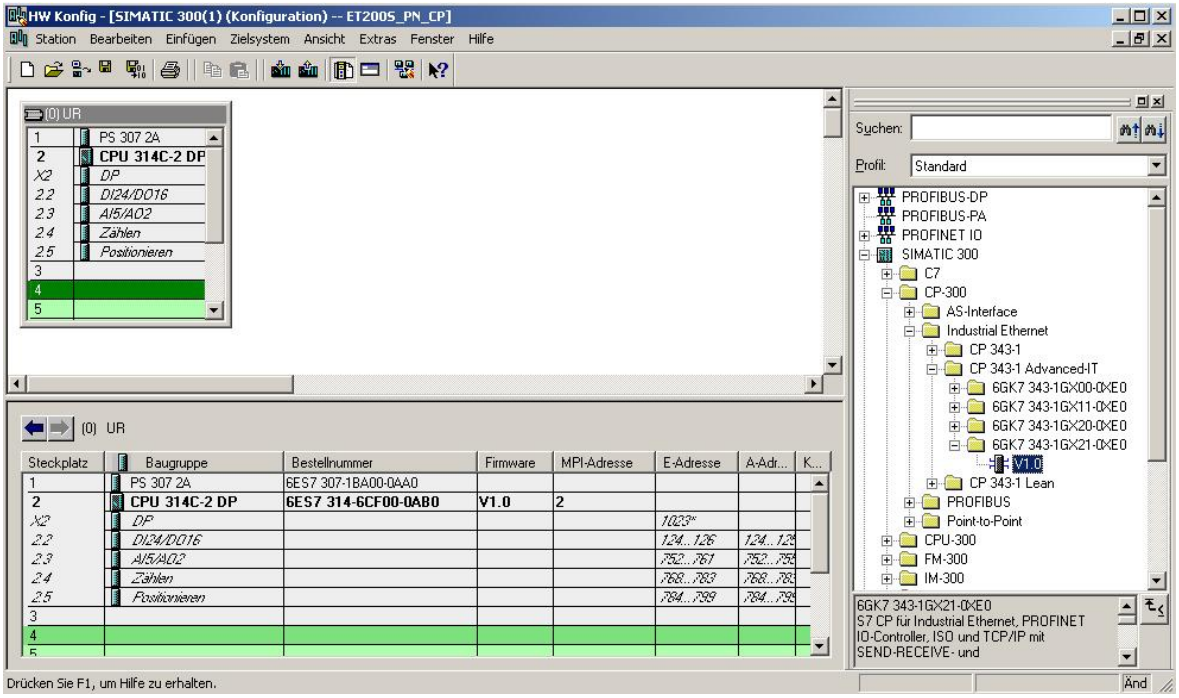


10. When entering the CPU, the following window appears. In this window, you can assign a **'PROFIBUS DP Address'** to the CPU 314C-2 DP.
Since we are not using the PROFIBUS, we leave the settings unchanged and accept with **'OK'** (→ OK)

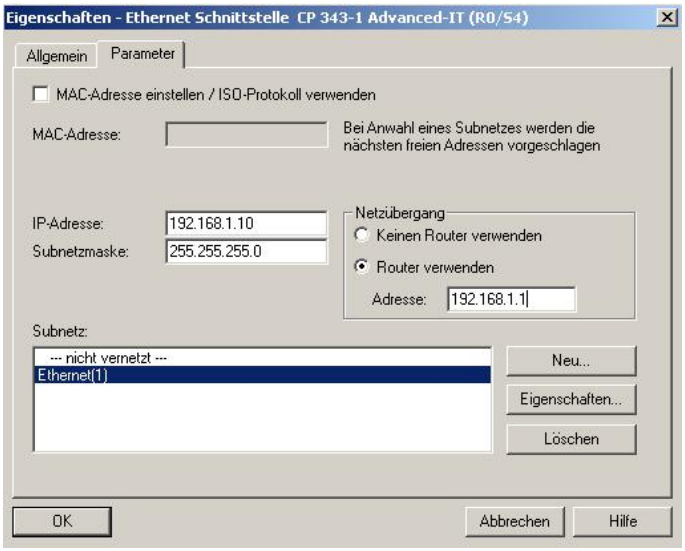


11. Next, we drag the **'CP343-1 Advanced IT'** to the 4th slot. Here also, the order number and the version of the CP can be read off the front. (→ SIMATIC 300 → CP-300 → CP 343-1 Advanced-IT → 6GK7 343-1GX21-0XE0 → V1.0)

Note: Slot 3 is reserved for interface modules, and remains empty for that reason. The module's order number is indicated in the footer of the catalog.



12. When entering the CP, the window below appears. In this window, do the following:
Assign an **'IP Address'** to the CP343-1 Advanced-IT, specify the **'Subnet Screen Form'**, and select the **'Ethernet'** network that has already been generated.
Optional: a **'Router Address'** can also be selected for network-overreaching communication.
Confirm your input with **'OK'** (→ IP Address: 192.168.1.10 → Subnet Mask: 255.255.255.0 → Ethernet(1) → Use Router → Address: 192.168.1.1 → OK)



Notes on Networking on the Ethernet (additional information is provided in Appendix V of the training document):

MAC Address:
The MAC address consists of a permanent and a variable part. The permanent part ("Basis MAC Address") identifies the manufacturer (Siemens, 3COM, ...). The variable part of the MAC address differentiates the different Ethernet stations, and should be assigned uniquely world-wide. On each module, a MAC address specified by the factory is imprinted.

Value range for the IP address:
The IP address consists of 4 decimal numbers from the value range 0 to 255 which are separated by a period; for example 141.80.0.16

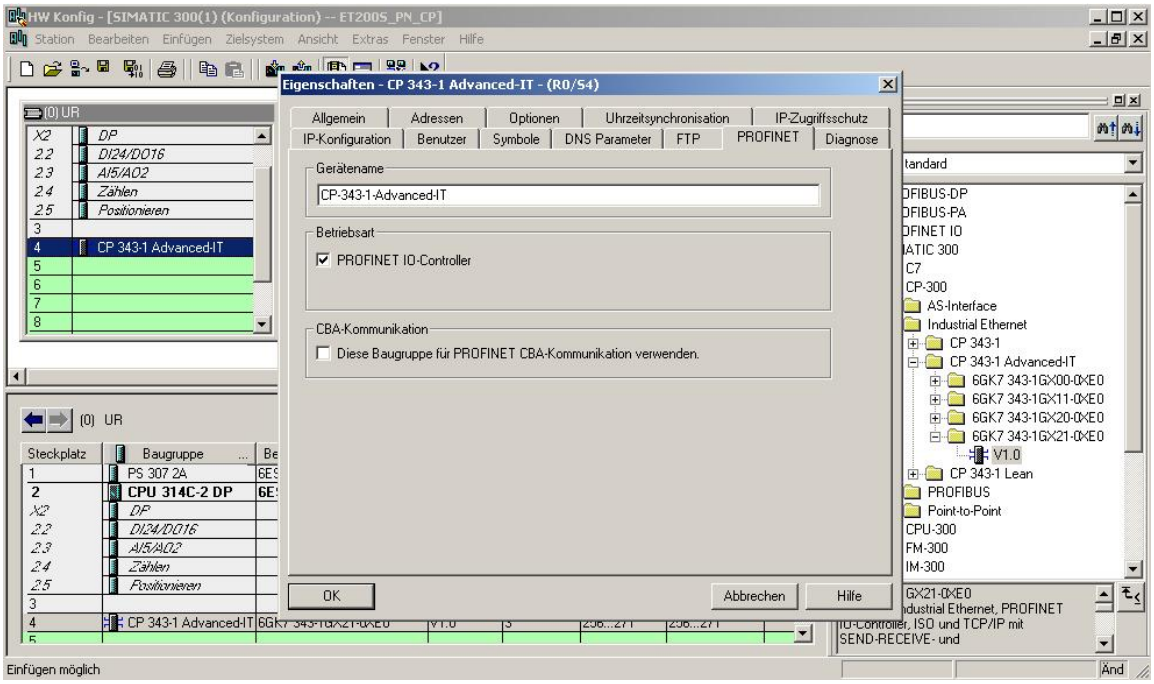
Value range for the subnet screen form:
This screen form is used in order to recognize whether a station or its IP address is part of the local subnet, or can be accessed only by means of a router.
The subnet screen form consists of 4 decimal numbers from the value range 0 to 255 which are separated by a period; for example, 255.255.0.0
In their binary representation, the 4 decimal numbers of the subnet screen form have to contain from the left a series of gapless values "1" and from the right a series of gapless values "0".
The values "1" determine the area of the IP address for the network number. The values "0" determine the area of the IP address for the station address.
Example:

Correct values:	255.255.0.0 decimal = 1111 1111.1111 1111.0000 0000.0000 0000 binary
	255.255.128.0 decimal = 1111 1111.1111 1111.1000 0000.0000 0000 binary
	255.254.0.0 decimal = 1111 1111.1111 1110.0000 0000.0000.0000 binary
Incorrect value:	255.255.1.0 decimal = 1111 1111.1111 1111.0000 0001.0000 0000 binary

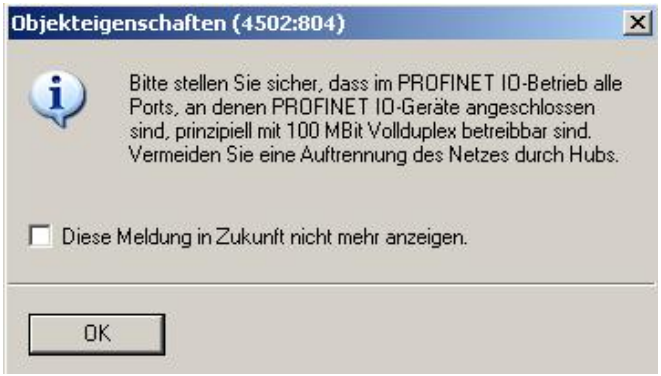
Value range for the address of the network transition (Router):
The address consists of 4 decimal numbers from the value range 0 to 255 which are separated by a period; for example, 141.80.0.1.

Relationship of IP addresses, router address, and subnet screen form:
The IP address and the address of the network transition may differ only in positions that have a "0" in the subnet screen form.
Example:
You entered: for the subnet screen form 255.255.255.0; for the IP address 141.30.0.5, and for the router address 141.30.128.1.
The IP address and the address for the network transition are to have a different value only in the 4th decimal number. In the example, however, the 3rd position already differs.
In the example, you have to change alternatively:
- the subnet screen form to: 255.255.0.0 or
- the IP address to: 141.30.128.5 or
- the address of the network transition to: 141.30.0.1

13. By double clicking with the mouse on the **'CP343-1 Advanced IT'**, you get to its properties. In this example, under **'PROFINET'**, set the CP's **'Operating Mode'** to **'PROFINET IO Controller'**. The other settings are retained. Close the window with **'OK'**. (→ CP 343-1 Advanced IT → PROFINET → Operating Mode → PROFINET IO Controller → OK)

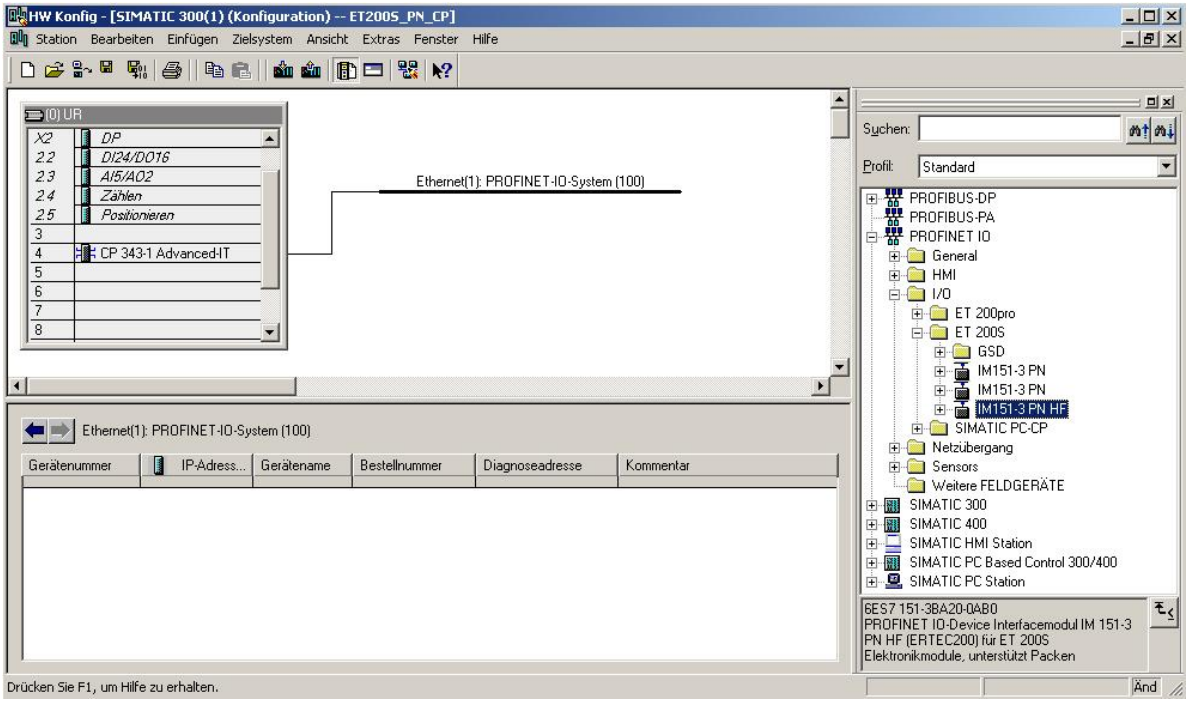


14. Confirm the note regarding network performance of PROFINET **'OK'**. (→ OK).

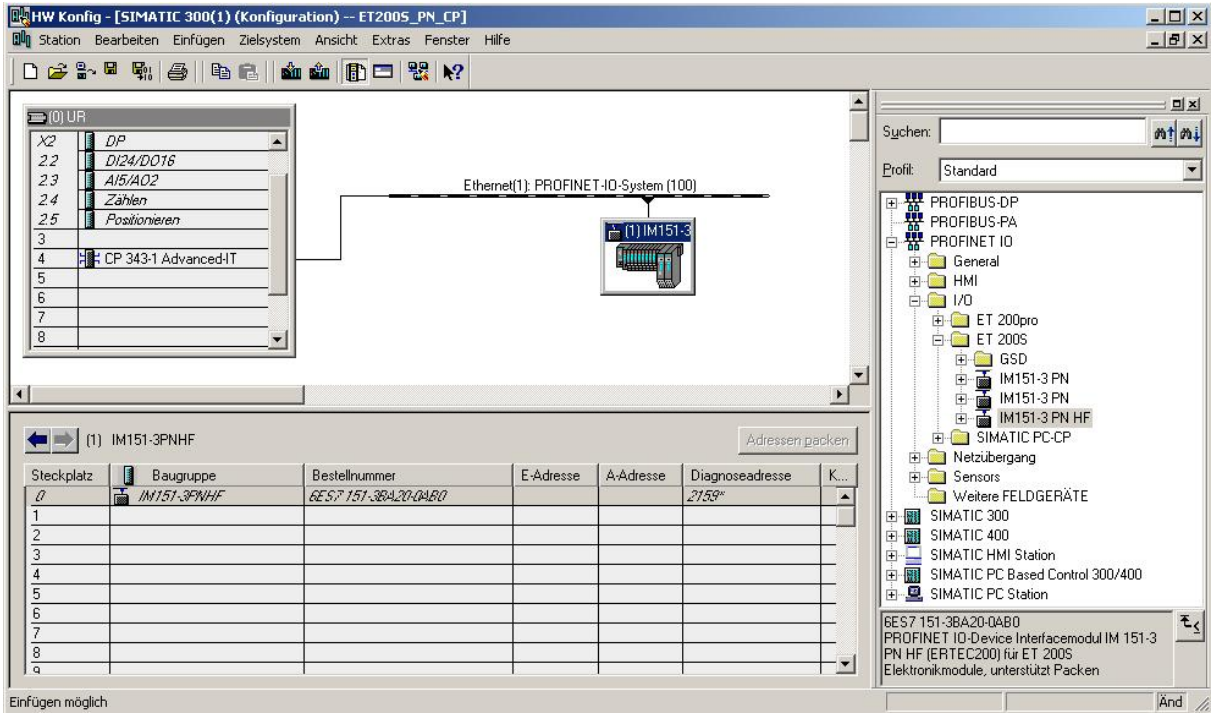


<<Object Properties (4502:804)
Please make sure that in the PROFINET IO mode, all ports to which PROFINET IO devices are connected can, in principle, be operated with 100Mbit full duplex. Avoid opening the network with hubs.
Don't show this message again in the future.>>

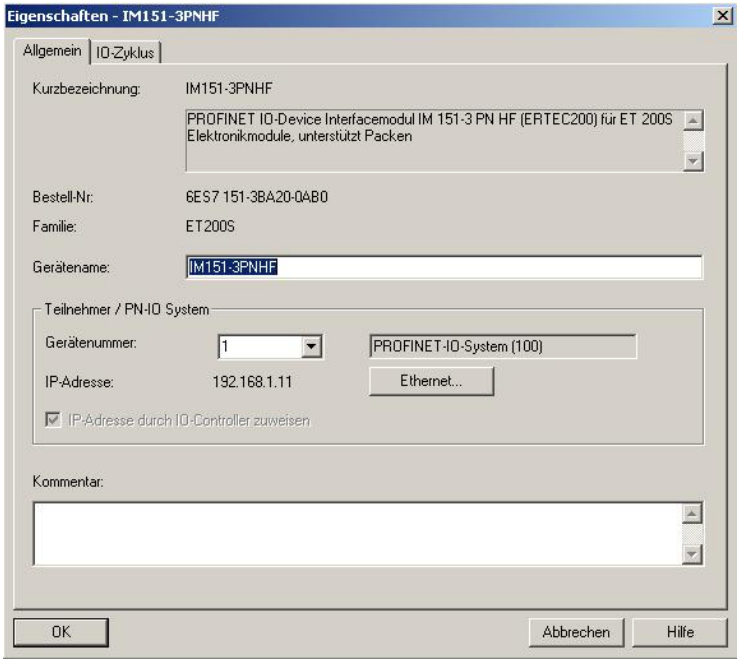
15. After you have accepted the network settings, a bar appears to the right of the CP343-1 Advanced IT, the **'PROFINET IO System'** where you can arrange PROFINET IO devices. This is done by clicking on the desired module (here the **'ET200S'** with **'IM151-3PN HF'**.) and dragging it to the **'PROFINET IO System'**. (→ PROFINET IO → I/O → ET 200S → IM151-3PN HF).



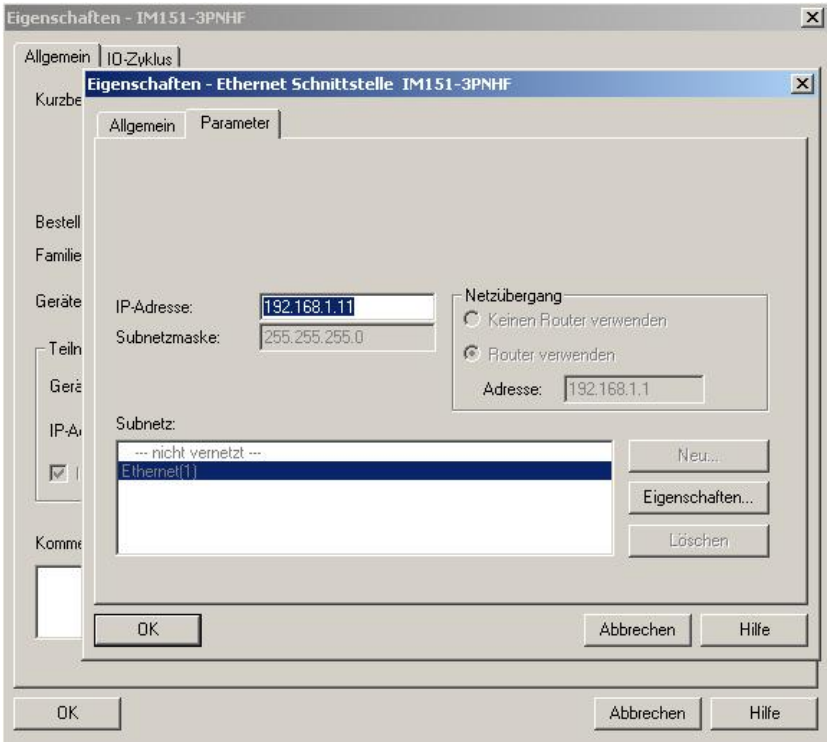
16. By double clicking on the **'IM151-3 PN HF'**, you open its properties.
(→ IM151-3 PN HF)



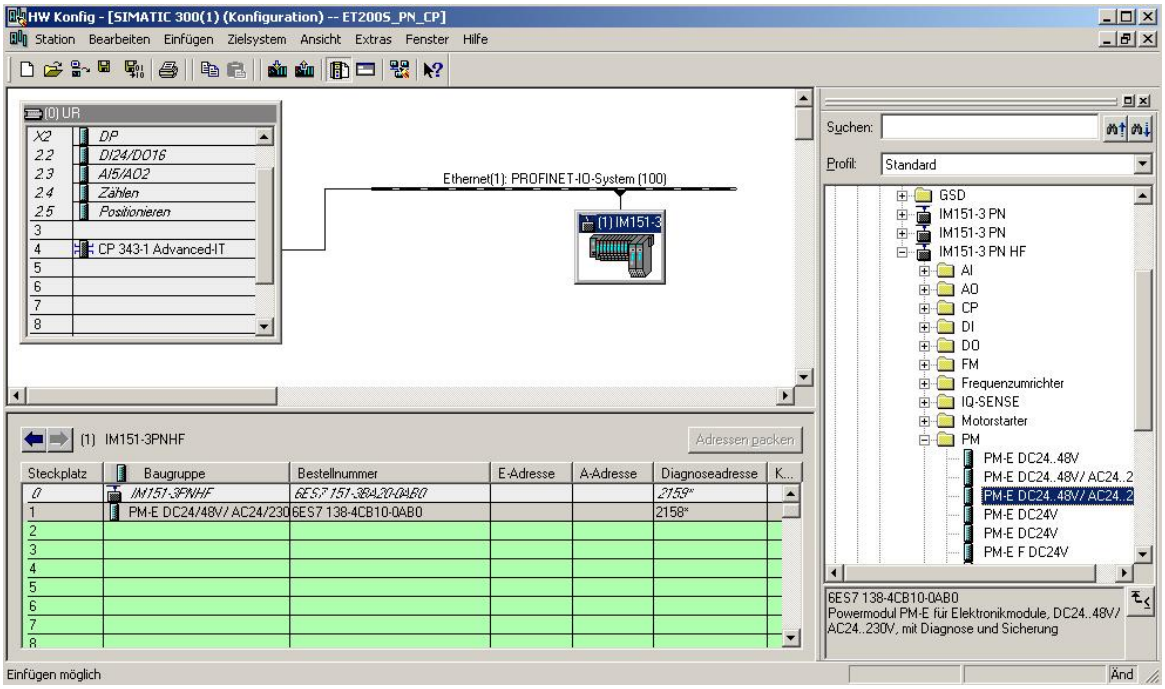
17. To each IO device, a **'Device Name'** that is unique within the PROFINET IO system has to be assigned, and an IP address on the **'Ethernet'**. (→ Device name: IM151-3PNHF → Ethernet)



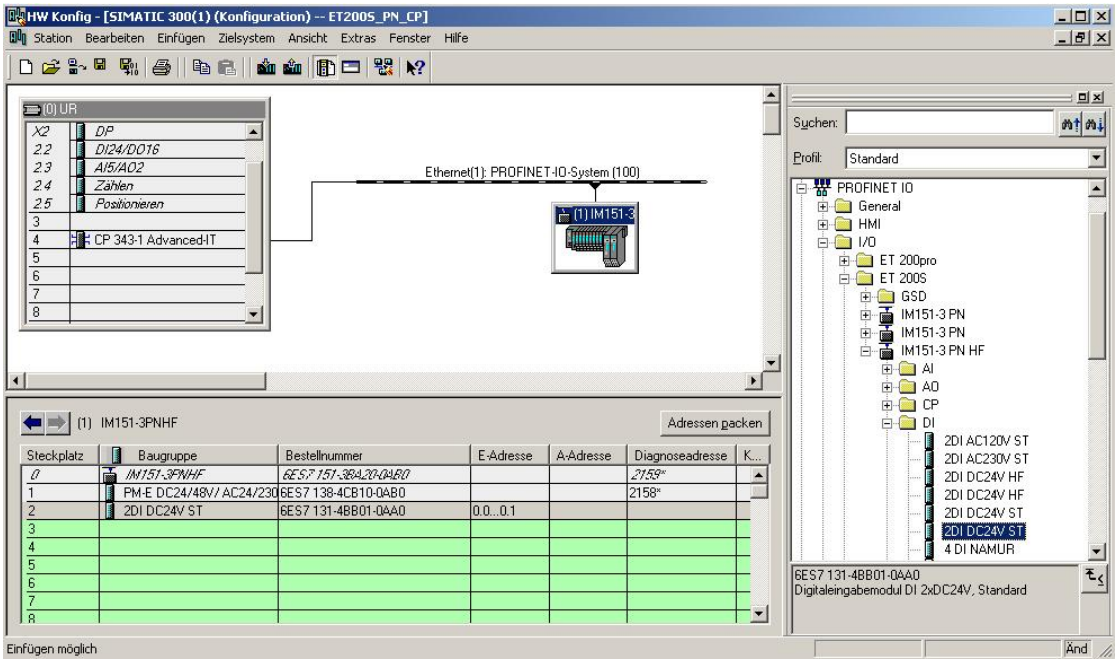
18. After you assigned the **'IP Address'**, accept with **'OK'**.
(→ IP-Address: 192.168.1.11 → OK → OK)



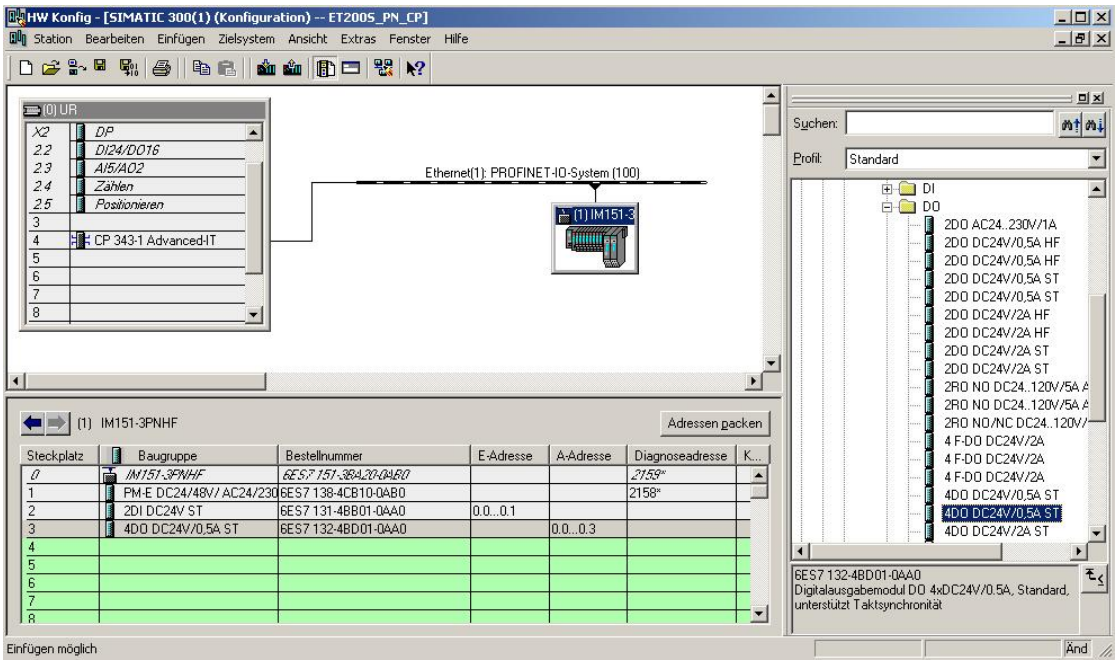
19. From the hardware catalog, you can now select all modules that are also in your real ET200S, and insert them in the configuration table.
To this end, click on the name of the respective module, hold the mouse key and drag the module to a line in the configuration table.
We are starting with the power unit **'PM-E DC 24V ... 48V/AC 24...230V**. (→ PROFINET IO → I/O → ET 200S → IM151-3 PN HF → PM → PM-E DC 24V...48V/AC 24...230V)



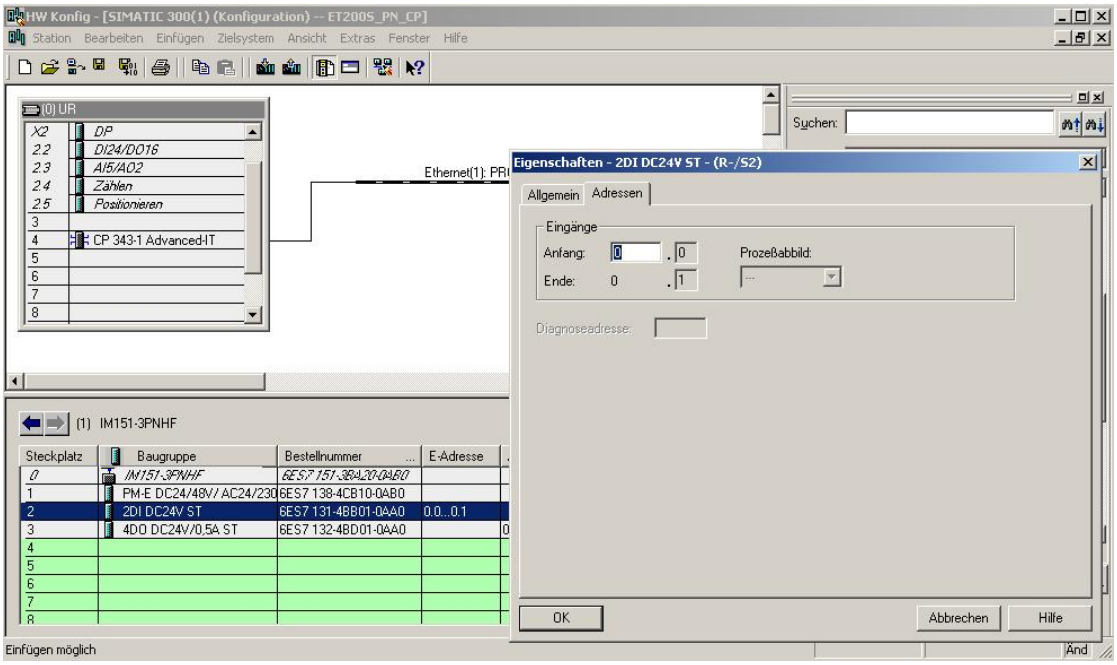
20. Next, we drag the digital input module **'2DI DC 24V ST'** to the 2nd slot. The order number and the version can be read off the module. (→ PROFINET IO → I/O → ET 200S → IM151-3 PN HF → DI → 2DI DC 24V ST)



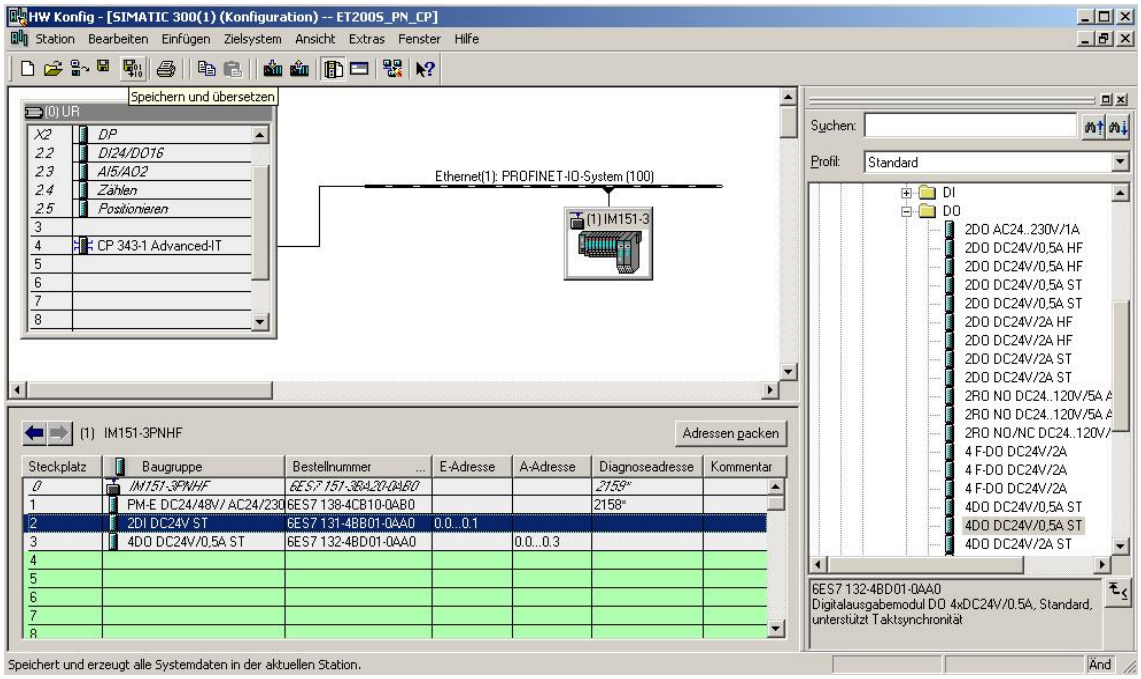
21. Then we drag the digital output module **'4 DO DC 24V/0.5A ST'** to the 3rd slot. The order number and the version can be read off the module. (→ PROFINET IO → I/O → ET 200S → IM151-3 PN HF → DO → 4 DO DC 24V/0.5A ST)



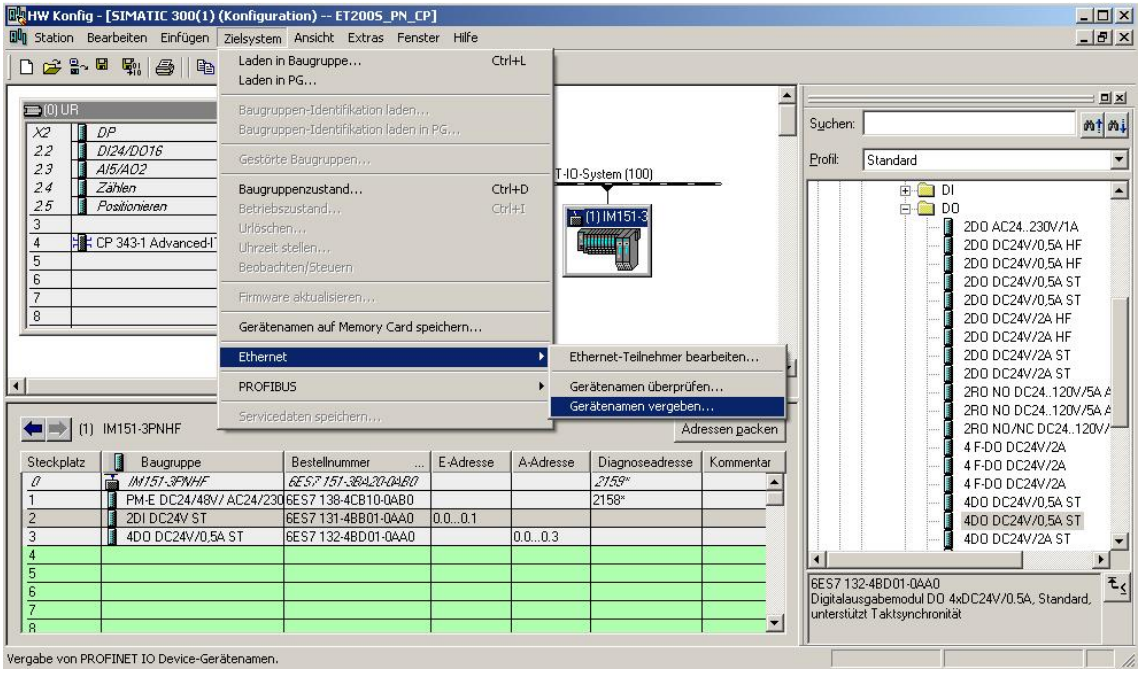
22. Now, the addresses of the inputs and outputs of the ET200S can be changed. To this end, we double click on the corresponding input and output modules in the ET200S, and set them under the tab **'Addresses'**. These addresses should be noted down in any case. Addresses are assigned automatically in the sequence in which the modules are entered. (→ 2DI DC 24V ST → Addresses → OK)



23. By clicking on '  ', the configuration table is stored and converted. (→ )



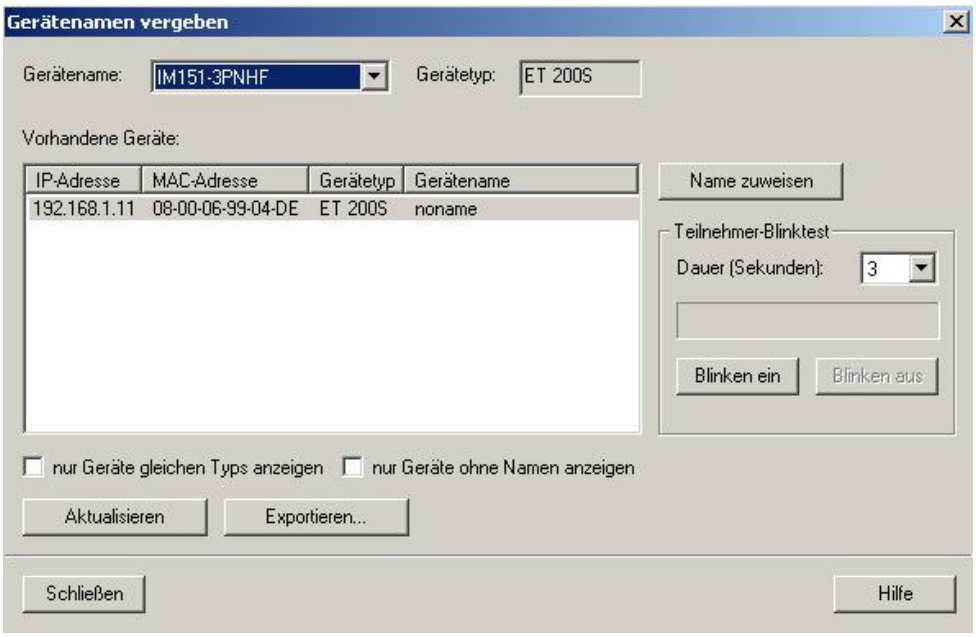
24. Now, after it was highlighted, the IO device has to be assigned a device name **'Assign device name'**. (→ IM151-3PN HF → PLC → Ethernet → Assign device name)



Note: A precondition for this is that the PG/PC interface is set to TCP/IP and the PC's network card is configured correctly. For example: IP address 192.168.1.99, Subnet 255.255.255.0 and Router address 192.168.1.1. (refer to Module E02!)

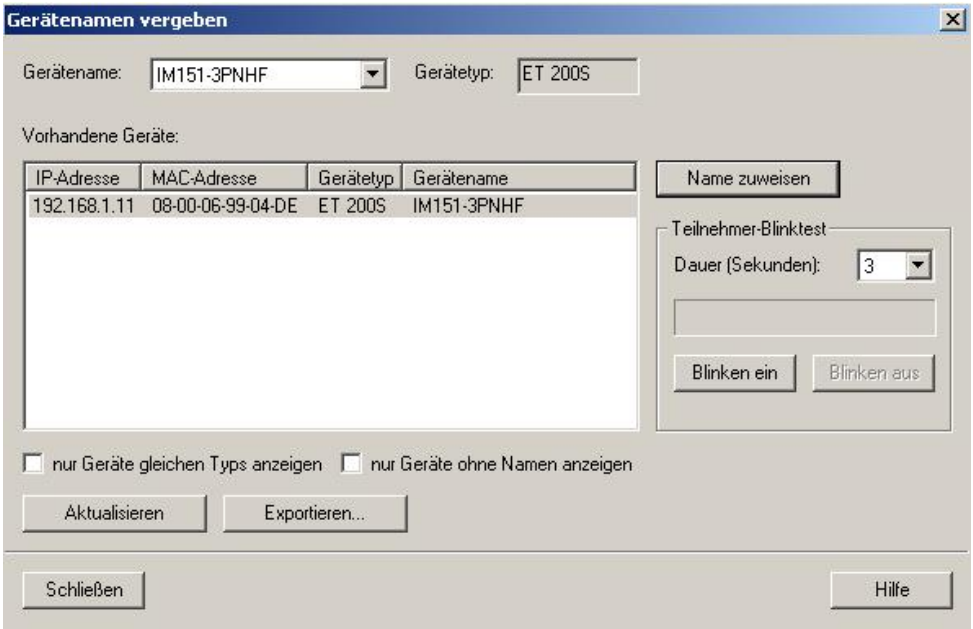
Note: Make sure your programming device is connected to the ET200S via the Ethernet!

25. Now, the ET200S has to be selected in order to assign a name **'Assign name'**. (→ ET200S → Assign name)

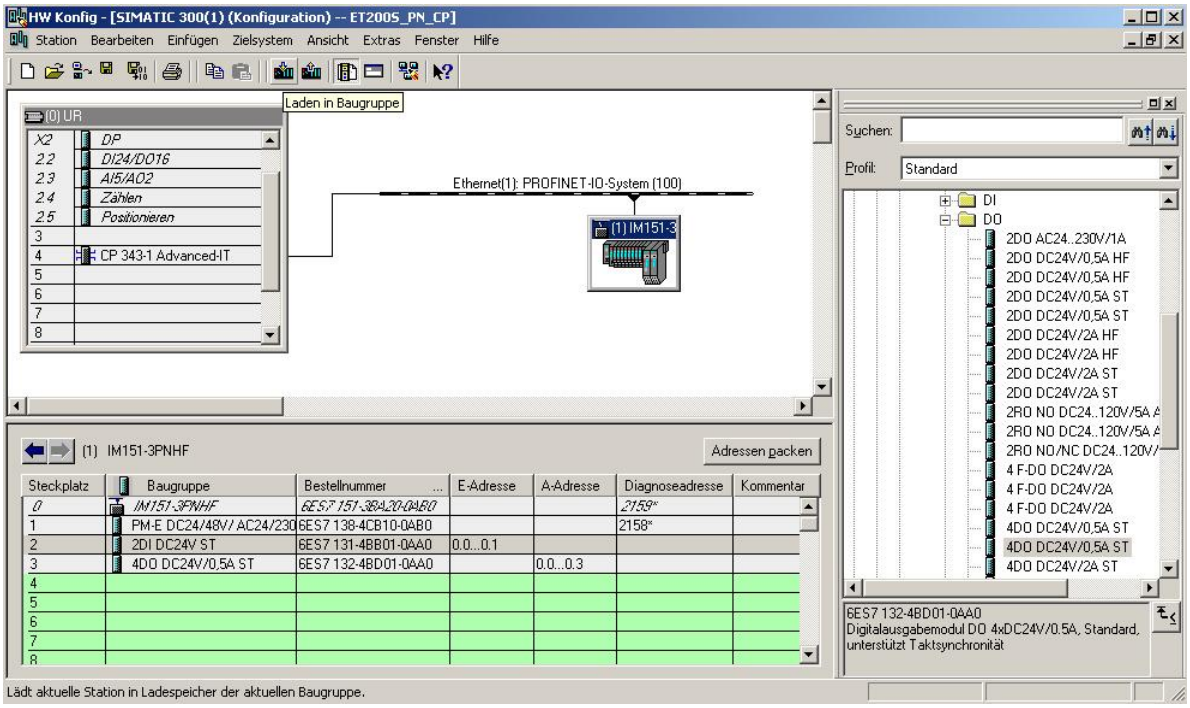


Note: If there are several IO devices in the network, the device can be identified with the imprinted MAC address.

26. The new device name is then displayed in the area **'Existing Devices'**. Then, **'Close'** the dialog. (→ Close)



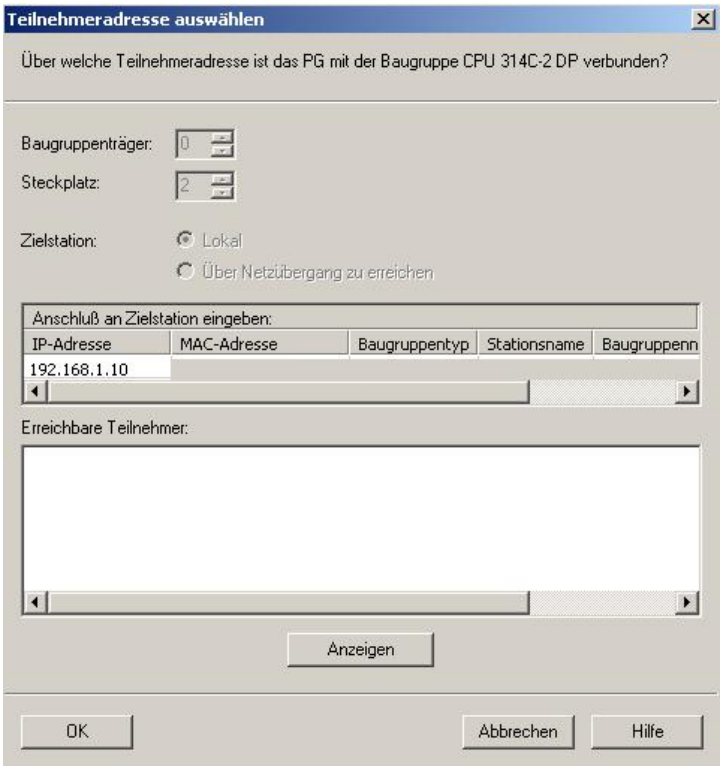
27. By clicking on '  ', you can now load the configuration table to the PLC. The CPU's operating mode switch should be on Stop! (→ 



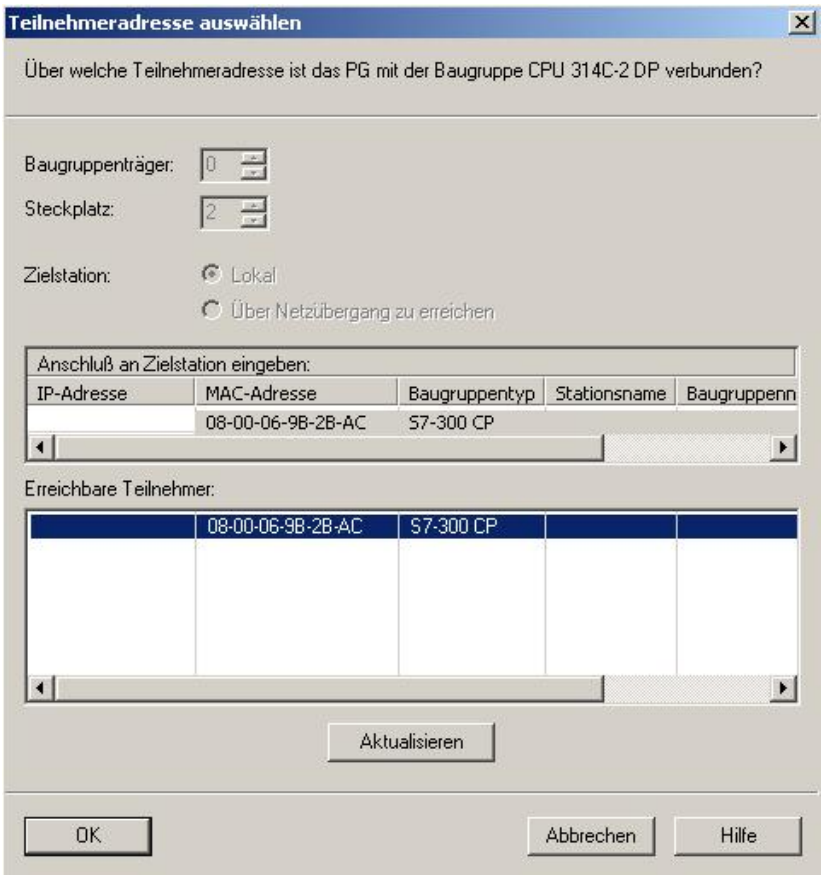
Note: Make sure your programming device is connected to the CP via the Ethernet!

28. CPU 314C-2 DP is confirmed as the destination module for loading the configuration table. (→ OK)

29. In the dialog below, you can have the devices that are connected to the network **'Display'**ed. (→ Display)



30. Then, select the MAC address of the CP on the Ethernet. If you are connected to only one CPU, just accept with **'OK'**. (→ OK)

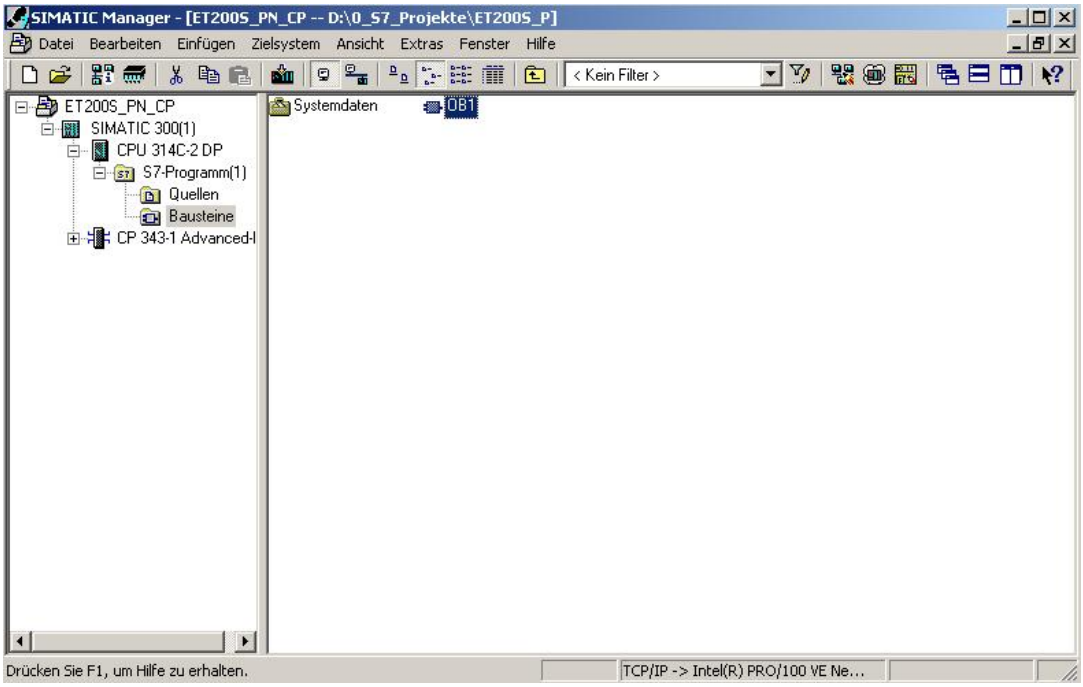


Note: If several IO controllers are on the network, the device can be identified with the imprinted MAC address.

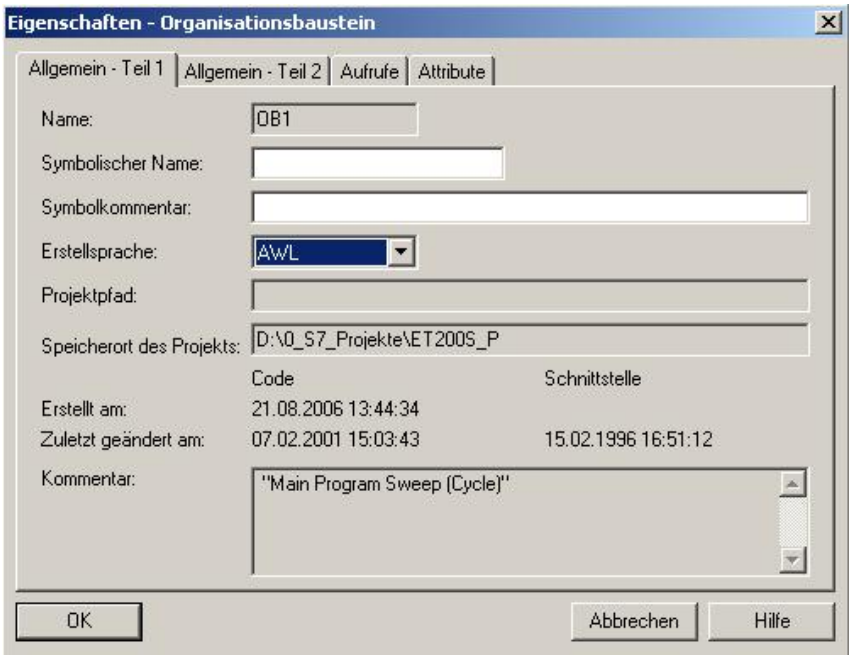
31. Now, the correct IP address has to be assigned to the controller if the address has not been set correctly yet. Confirm this in the dialog below with 'Yes'. (→ Yes)

<<The selected station does not have an IP address. Do you want to assign address 192.168.1.10 now?>>

32. After you loaded the hardware configuration, you can start setting the program. From the **'SIMATIC Manager'**, open the block **'OB1'** with a double click. (→ OB1)



33. Select the **'Programming language STL'** and accept with **'OK'**. (→ STL →OK)



By means of programmed FC block calls, the user program triggers the transfer of the data areas for the inputs and outputs of the IO devices. These FCs also monitor the successful execution.
The FC blocks needed for communication are stored in the library **"SIMATIC_NET_CP"**. To use these functions, they have to be integrated into (copied to) your "own" project.

The FC block **PNIO SEND** transmits data from the user program in the CPU to the CP343-1 Advanced:
The block transfers the data of a specified output area of the CPU to the CP343-1 Advanced for output at the IO devices.

When calling the FC block **PNIO SEND**, the following parameters have to be entered in succession:

Name	Type	Value Range	Comment
CPLADDR	WORD		Module start address of the CP (in STEP7, is provided in the configuration table).
LEN	INT	1...2160	Number of bytes that are to be send from the data area specified in SEND.
IOCS	ANY	1…270 bytes	Status information for the output area - 1 bit per byte input data - Address gaps are transferred with 0 == GOOD - 0 == GOOD/1 == BAD The length of this pointer results from the variable LEN for: (LEN/8 + 1) byte The minimum length is 1 byte.
DONE	BOOL	0: - 1: New data transferred	The status parameter indicates whether sending was successful
ERROR	BOOL	0: - 1: Error	Error
STATUS	WORD		Status display for the diagnostic and error bit. Regarding error analysis, important notes are provided in Online Help!
CHECK_ICPS	BOOL	0: - 1: Error	Auxiliary bit that indicates whether IOCS should be evaluated
SEND	ANY		Specifies address and length of the data area in the CPU from which the data that is to be sent to the IO devices is to be fetched. The address can refer to IO areas, flag areas, and data block areas.

The FC block **PNIO-RECV** receives data from the CP343-1 Advanced, and writes it to the user program of the CPU.
The block accepts process data of the IO devices as well as status information from the CP343-1 Advanced into a specified input area of the CPU.

When calling the FC block **PNIO-RECV**, the following parameters have to be entered in succession:

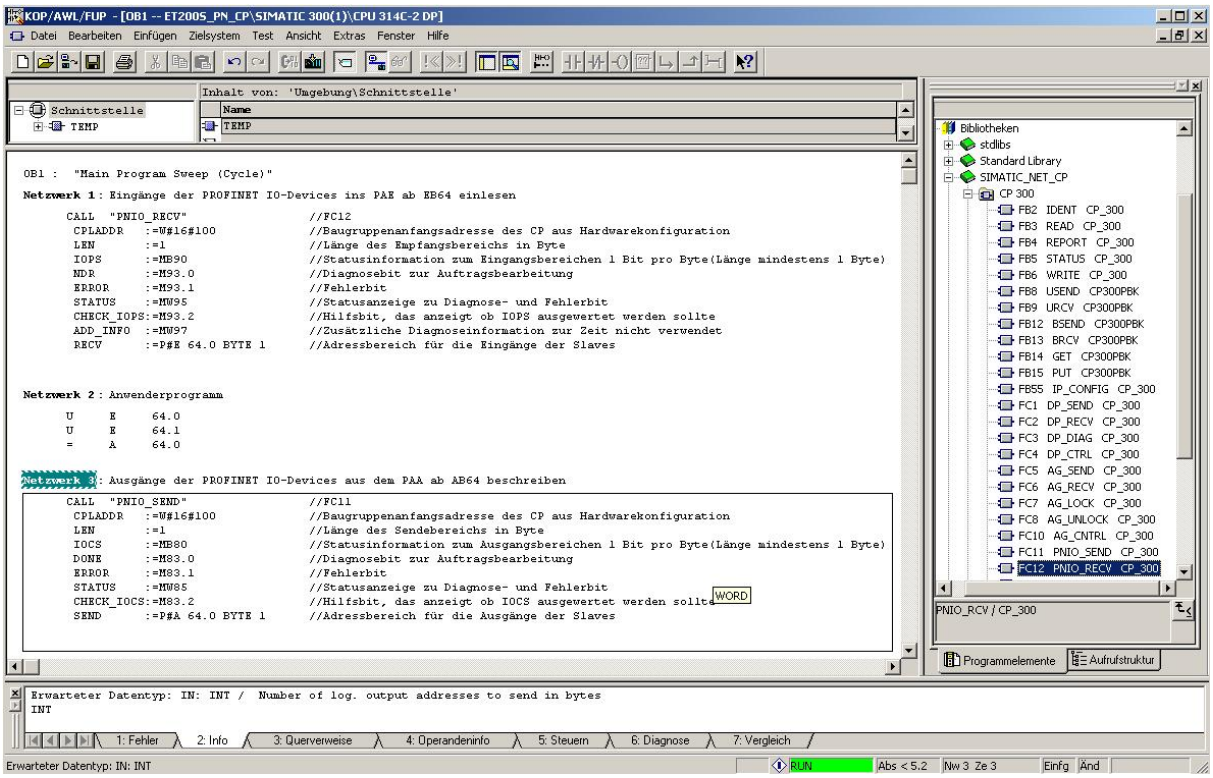
Name	Type	Value Range	Comment
CPLADDR	WORD		Module start address of the CP (in STEP7, is provided in the configuration table).
LEN	INT	1...2160	Number of bytes that are to be received from the data area specified in RECV.
IOPS	ANY	1…270 Byte	Status information for the input area - 1 bit per byte input data - Address gaps are transferred with 0 == GOOD - 0 == GOOD/1 == BAD The length of this pointer results from the variable LEN for: (LEN/8 + 1) byte The minimum length is 1 byte.
NDR	BOOL	0: - 1: New data accepted	The status parameter indicates whether new data was accepted
ERROR	BOOL	0: - 1: Error	Error bit
STATUS	WORD		Status display for the diagnostic and error bit. Regarding error analysis, important notes are provided in Online Help!
CHECK_IOPS	BOOL	0: - 1: Error	Auxiliary bit that indicates whether IOPS should be evaluated
ADD_INFO	WORD		Additional diagnostic information, presently not used and therefore always 0.
RECV	ANY		Specifies address and length of the data area in the CPU in which the data received from the IO devices is to be stored. The address can refer to IO areas, flag areas, and data block areas.

34. With **'LAD, STL, FBD – Program S7 Blocks'**, you now have an editor with which you can generate your STEP7 program accordingly. To this end, OB1 has already been opened with the first network. To generate your initial operations, highlight the first network. Now you can write your first STEP7 program. In STEP7, individual programs are usually arranged in networks. A new network is opened by clicking on the network symbol '  '.

Here, in Network 1, the inputs of the IO devices are entered with the block **'PNIO_RECV'**. In the catalog, from the **'Library' 'SIMATIC_NET**_CP', you can drag this block to your network. (→ Libraries → SIMATIC_NET_CP → CP 300 → FC12 PNIO_RECV)

In Network 3, the outputs of the IO devices are written to with the block **'PNIO_SEND'**. In the catalog, from the **'Library' 'SIMATIC_NET_CP'**, you can drag this block to your network. (→ Libraries → SIMATIC_NET_CP → CP 300 → FC11 PNIO_SEND)

Now save OB1 '  '. (→ )



STEP7 Program in OB1:

Network 1: Enter inputs of the PROFINET IO devices to the PII, starting with IB64

```
CALL  "PNIO_RECV"                    //FC12
   CPLADDR      :=W#16#100           //Module start address of the CP from
                                     //hardware configuration
   LEN          :=1                  //Length of the receive area in bytes
   IOPS         :=MB90               //Status information for the input area 1 bit per
                                     //byte (length at least one byte)
   NDR          :=M93.0              //Diagnostic bit for request processing
   ERROR        :=M93.1              //Error bit
   STATUS       :=MW95               //Status display for diagnostic and error bit
   CHECK_IOPS   :=M93.2              //Auxiliary bit that indicates whether IOPS should be
                                     //evaluated
   ADD_INFO     :=MW97               //Additional diagnostic information. Presently not
                                     //used
   RECV         :=P#E 64.0 BYTE 1    //Address area for the inputs of the slaves
```

Network 2: User Program

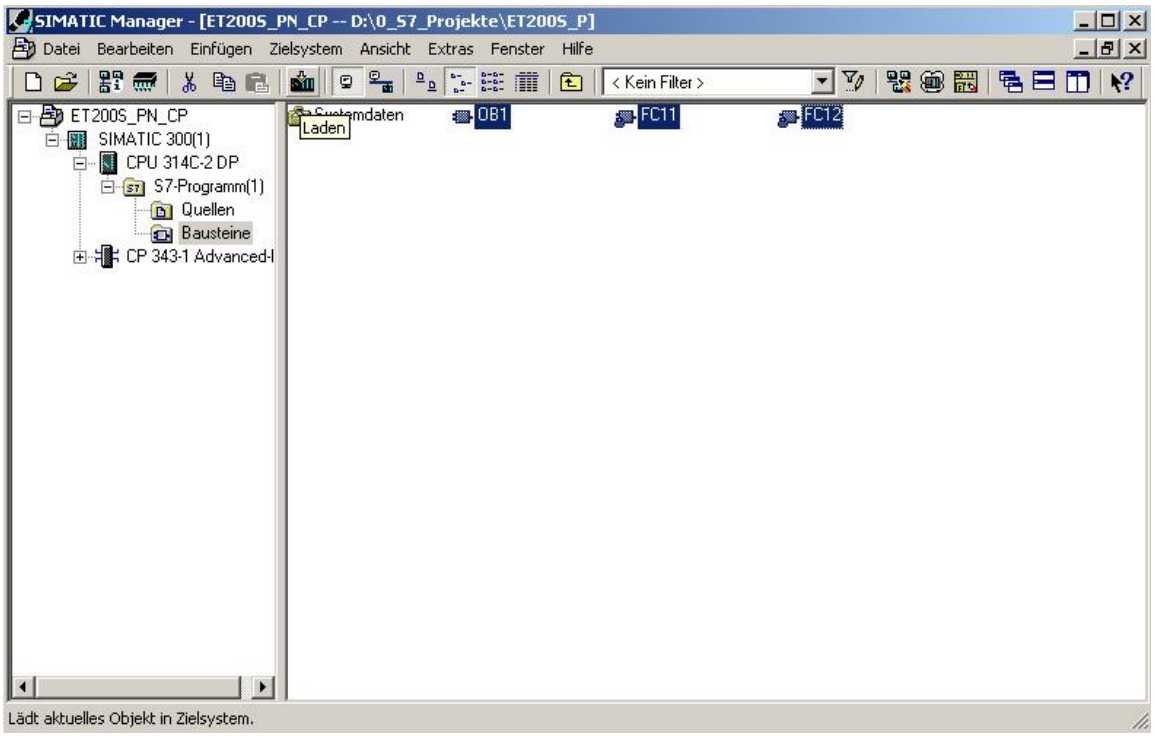
```
   U   I   64.0
   U   I   64.1
   =   O    64.0
```

Network 3: Writing to outputs of the PROFINET IO devices from the PIO, starting with OB64

```
CALL  "PNIO_SEND"                    //FC11
   CPLADDR      :=W#16#100           //Module start address of the CP from the
                                     //hardware configuration
   LEN          :=1                  //Length of the send area in bytes
   IOCS          :=MB80              //Status information for the output range 1 bit per
                                     //byte (length at least 1 byte)
   DONE         :=M83.0              //Diagnostic bit for request processing
   ERROR        :=M83.1              //Error bit
   STATUS       :=MW85               //Status display for diagnostic and error bit
   CHECK_IOCS   :=M83.2              //Auxiliary bit whether IOCS should be
                                     //evaluated
   SEND         :=P#A 64.0 BYTE 1    //Address area for the outputs of the slaves
```

Note: Here, an ET200S with 1 byte input data and 1 byte outputs data is incorporated on Slot 4 (module start address decimal: 256/hexa-decimal 100) by means of a CP343-1 Advanced.
The input data is to be located in the input area starting with IB 64, and the output area is to write the data to the ET200S starting with OB 64.
It is important that the data of all IO devices that are defined in the hardware configuration is integrated with the blocks PNIO _RECV and PNIO _SEND, whereby all IO devices are combined in one PNIO _RECV and one PNIO _SEND.
The area that is read/written to may be larger, but not smaller than the address area of the IO devices!!
The addresses of individual modules are provided in the hardware configuration.

35. Now, the STEP7 program has to be loaded to the PLC. In our case, this is done from the **'SIMATIC Manager'**. There, highlight **'OB1'** and the FCs **'FC11'** and **'FC12'** in the folder 'Blocks', and then click on Load '  '. The CPU's key switch should be on STOP! (→ SIMATIC Manager → Blocks → OB1 → FC11 → FC12 → )



Note: Make sure that the CPU is connected to the CP via Ethernet!

36. By setting the key switch to RUN, the program is started.