Training document for the company-wide automation solution Totally Integrated Automation (T I A)

MODULE A4

Programming of the CPU 315-2DP

We thank the company Michael Dziallas Engineering and the instructors of vocational schools as well as further persons for the support with the production of the document.

PAGE:

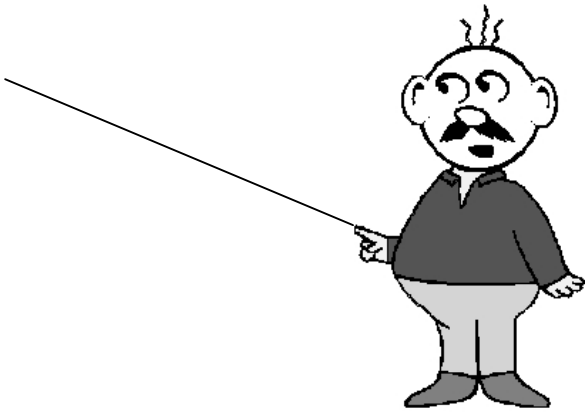
The following symbols stand for the specified modules:

Information

Programming

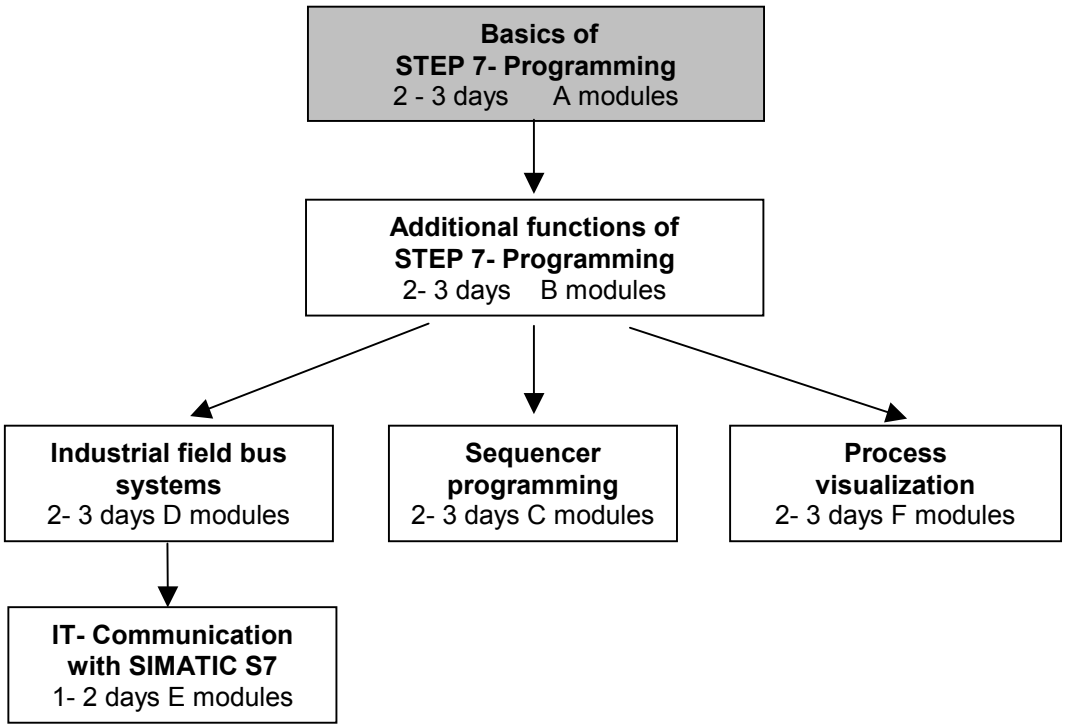
Example exercise

Notes



1. FORWARD

The module A4 is assigned content wise to the **Basics of STEP 7- Programming**.



Learning goal:

In this module, the reader will learn about the generation of the hardware configuration for the CPU 315-2DP and the writing and debugging of a STEP 7-Program. The module arranges the principle procedure and shows them in the following steps by means of a detailed example.

- Application of a STEP 7- Project
- Generation of the hardware configuration for the CPU 315-2DP
- Writing of a STEP 7- Program
- Debugging of a program

Requirements:

For the successful use of this module the following knowledge is assumed:

- Knowledge in the use of Windows 95/98/2000/ME/NT4.0
- Basics of PLC- Programming with STEP 7 (e.g. Module A3 – 'Startup' PLC- Programming with STEP 7)

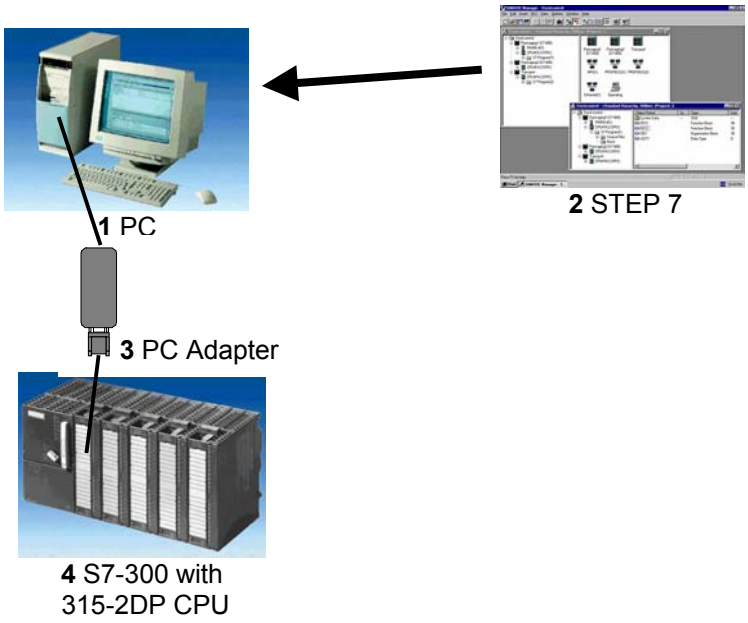
Required hardware and software

1 PC, Operating system Windows 95/98/2000/ME/NT4.0 with
- Minimal: 133MHz and 64MB RAM, approx. 65 MB free hard disk space
- Optimal: 500MHz and 128MB RAM, approx. 65 MB free hard disk space

2 Software STEP7 V 5.x

3 MPI- Interface for the PC (e.g. PC- Adapter)

4 PLC SIMATIC S7-300 with the CPU 315-2DP and a minimum of one digital In- and Output device. The inputs must be led out of a switch bay.
Example configuration:
- Power supply: PS 307 2A
- CPU: CPU 315-2DP
- Digital inputs: DI 16x DC24V
- Digital outputs: DO 16x DC24V / 0.5 A



2. NOTES FOR THE APPLICATION OF THE CPU 315-2DP

The CPU 315-2DP is a CPU that is combined with an integrated PROFIBUS DP- Interface.

The following PROFIBUS- Protocol profiles are available for the CPU 315-2DP:

- DP-Interface as Master in accordance with EN 50170.
- DP-Interface as Slave in accordance EN 50170.

The PROFIBUS-DP (Subsidiary Peripherals) is the protocol for the connection from the subsidiary peripherals/field equipment with a quick reaction time.

A further characteristic is that the addresses of the input and output modules can be parameterized by this CPU.

The operation efficiency is sufficient for training purposes in each case with the following data:
- 16K statements. 48Kbyte workspace 80Kbyte build space.
- 1024 Byte DI/DO
- 128 Byte AI/AO
- 0,3 ms / 1K instructions
- 64 counters
- 128 Timers
- 2048 bit memories

3. GENERATION OF THE HARDWARE CONFIGURATION FOR THE CPU 315-2DP

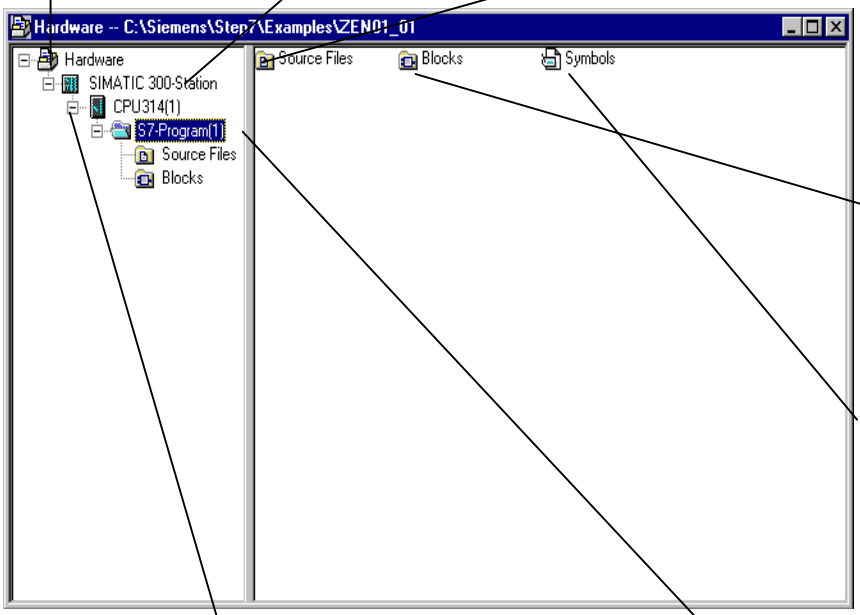
File management takes place in STEP 7 with the **SIMATIC Manager**. Here e.g. program blocks can be copied or be called for further processing with other tools by clicking with the mouse. The operation corresponds to the standards usually seen in WINDOWS 95/98/2000/ME/NT4.0. (in such a way e.g. With one right click from the mouse button, one is able to receive the selection menu to each module).
In the folders **SIMATIC 300 station** and **CPU**, the structure of the hardware of the PLC is illustrated. Therefore such a project can always be seen as hardware specific.

In STEP 7, each project is put into a firmly given structure. The programs are stored in the following directories:

Project:
The directory contains the hardware (e.g. SIMATIC 300 Station) and the sub structure (e.g. MPI and PROFIBUS).

SIMATIC 300 Station:
Stored here are the appropriate hardware configuration (Hardware/SC*1) and CPU data.

Source Files/SO*[1]:
Sources are placed here (e.g. SCL- Source Files). They can be converted into executable programs by translation.



Blocks/AP-off*[1]:
Stored here are the program blocks (OB, FB, FC, SFB, SFC, DB etc.).

Symbols/SY*[1]:
Stored here are the symbol lists for symbolic addressing.

CPU:
The S7 program and the interlaced connecting partners (Connection/CO*[1]) are registered here.

S7-Program: The user programs (Blocks/AP-off*[1]), symbol tables (Symbols/SY*[1]), and Source files(Source files/SO*[1]) are administered here.

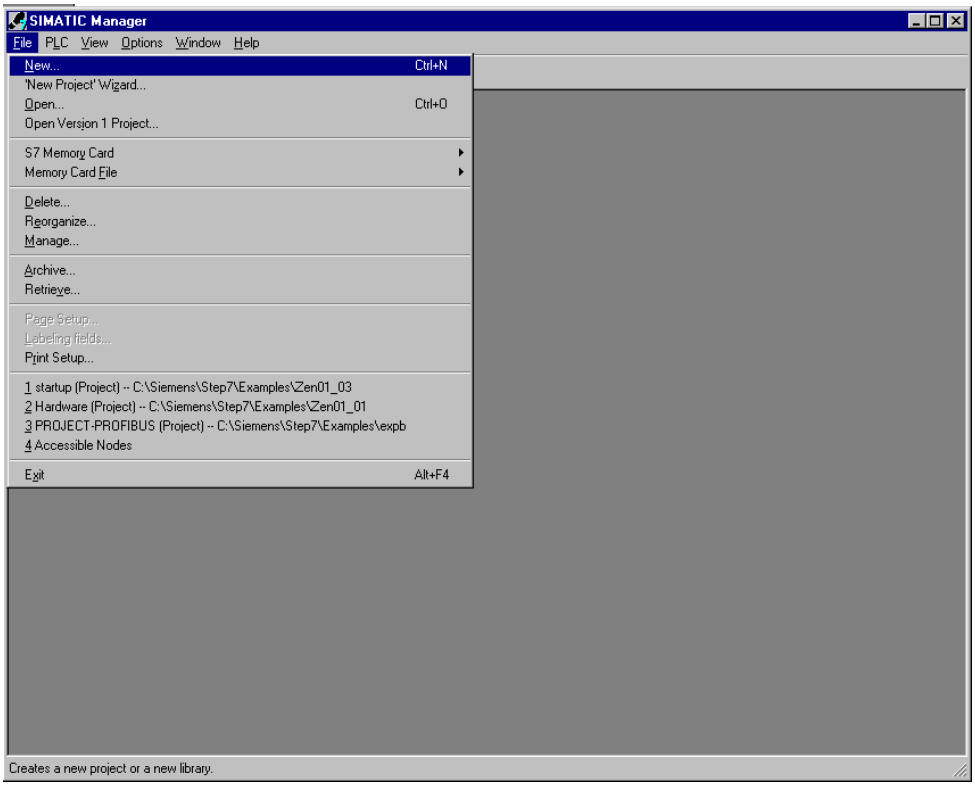
*[1] Terms are from STEP 7 Version 2.x

In the folders **SIMATIC 300 station** and **CPU**, the structure of the hardware of the PLC is illustrated. This is done here for the special case of a configuration with the CPU 315-2DP. Another clock memory should still be configured and the addresses of the inputs and output modules should be adjusted.

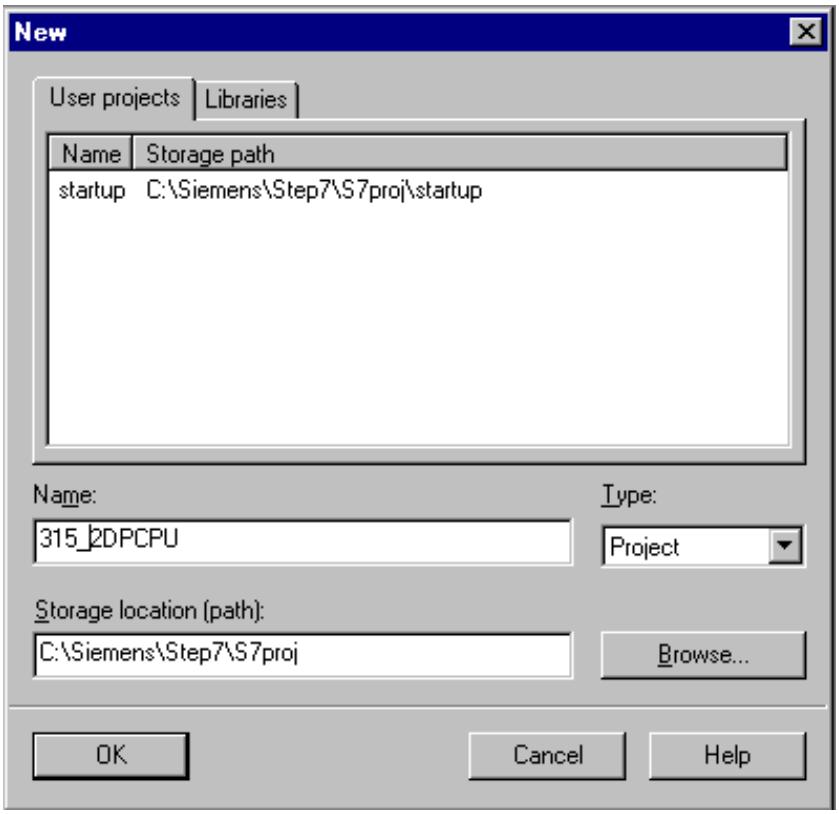
The user must implement the following steps in order to provide a project in which the solution program can be written.

1. The main tool in STEP 7 is the **SIMATIC Manager**, which can be opened with a double click on the icon (→ SIMATIC Manager).

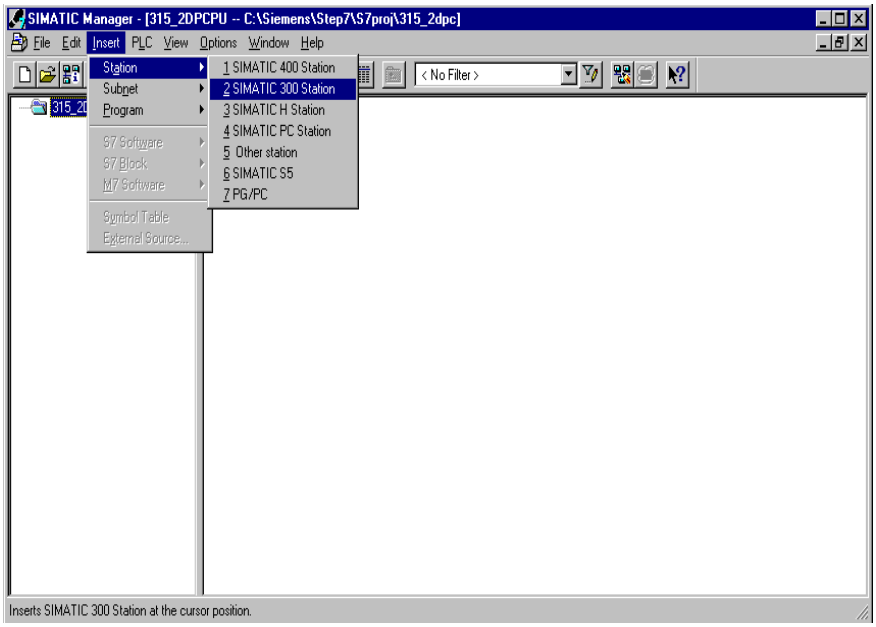
2. STEP 7- Programs are managed in projects. Each project can be newly created (→ File → New).



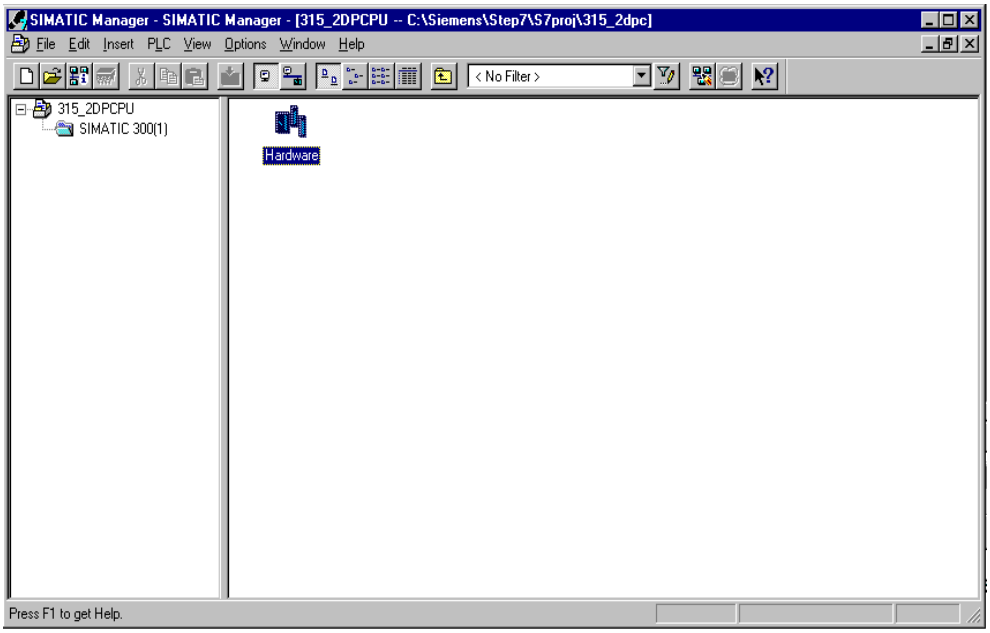
3. Give the project the **Name 315_2DPCPU** (→ 315_2DPCPU → OK).



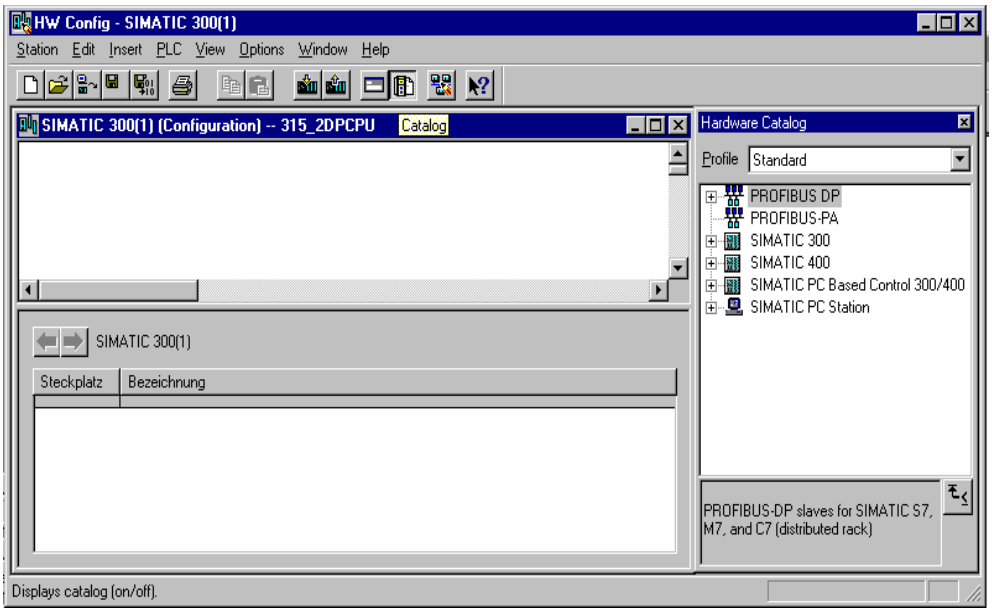
4. Insert a **SIMATIC 300-Station** (→ Insert → Station → SIMATIC 300-Station).



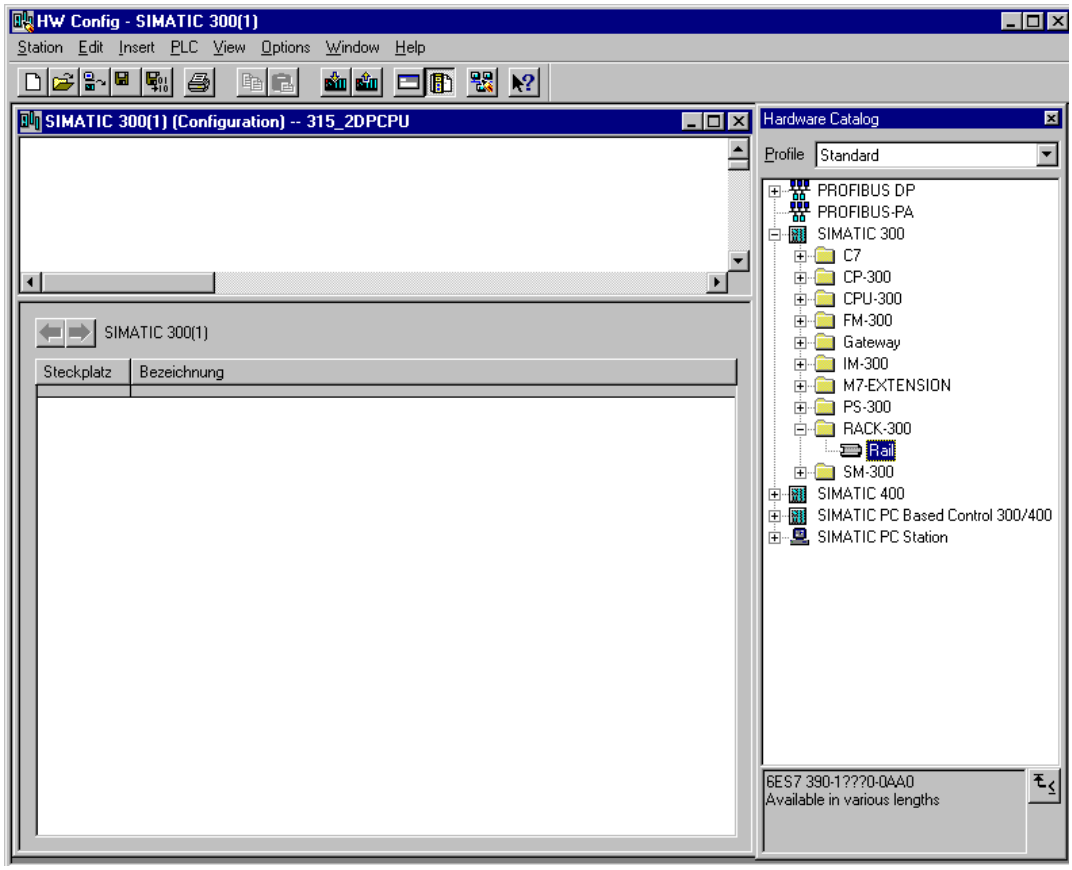
5. Open the configuration kit by double clicking on ‘Hardware’ (→ Hardware).



6. Open the hardware catalog by clicking on the symbol  (→ ).
The contents are divided into the following modules:
- PROFIBUS-DP, SIMATIC 300, SIMATIC 400 and SIMATIC PC Based Control,
All components, blocks and interface modules for your assembled project are shown there.

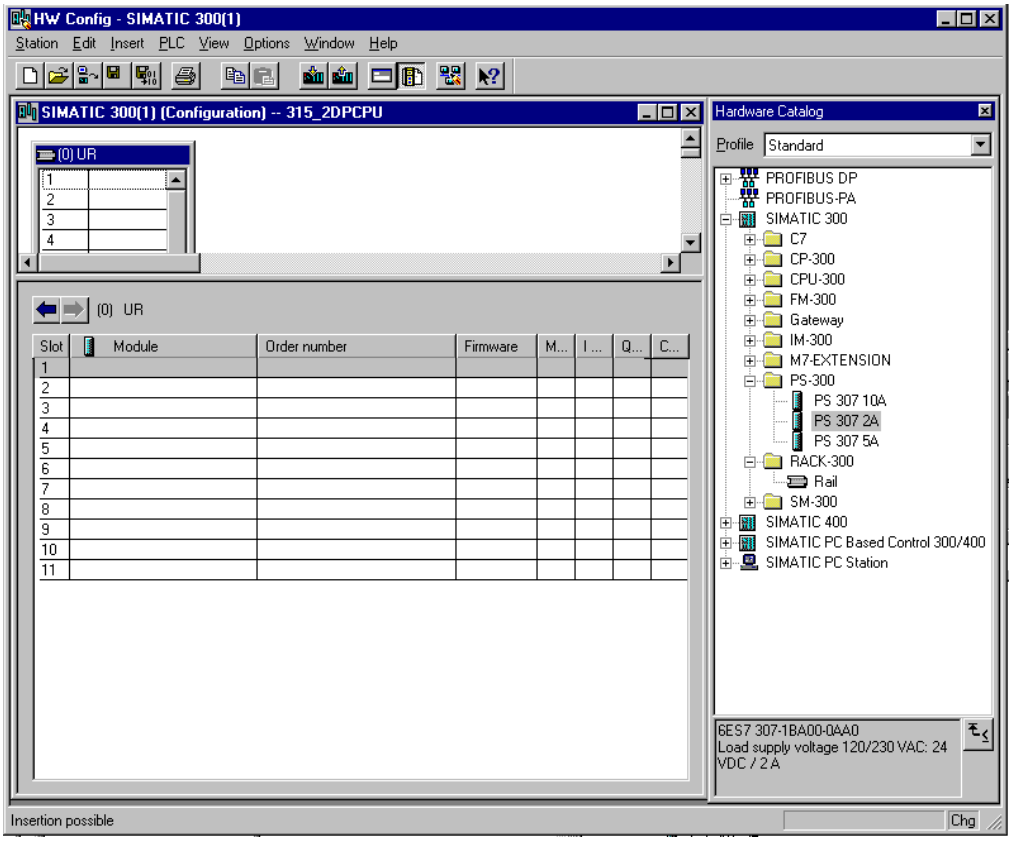


7. Double click on **Rail** (→ SIMATIC 300 → RACK-300 → Rail).



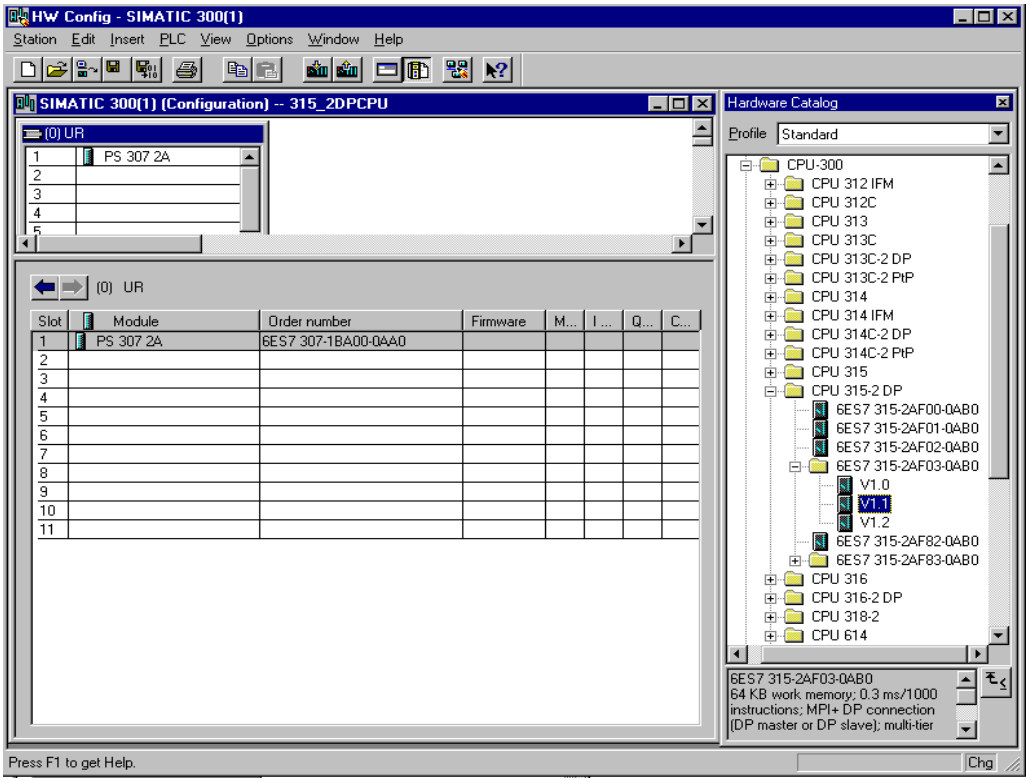
Afterwards, a configuration table for the structure of the RACK 0 is blended in automatically.

8. Now all modules which are in the material rack can be selected and inserted in the configuration table from the hardware catalog. To perform this action, you must click on the indicator of the respective module, hold the mouse button and drag & drop the item into the configuration table. We will begin with the power supply **PS 307 2A** (→ SIMATIC 300 → PS-300 → PS 307 2A).

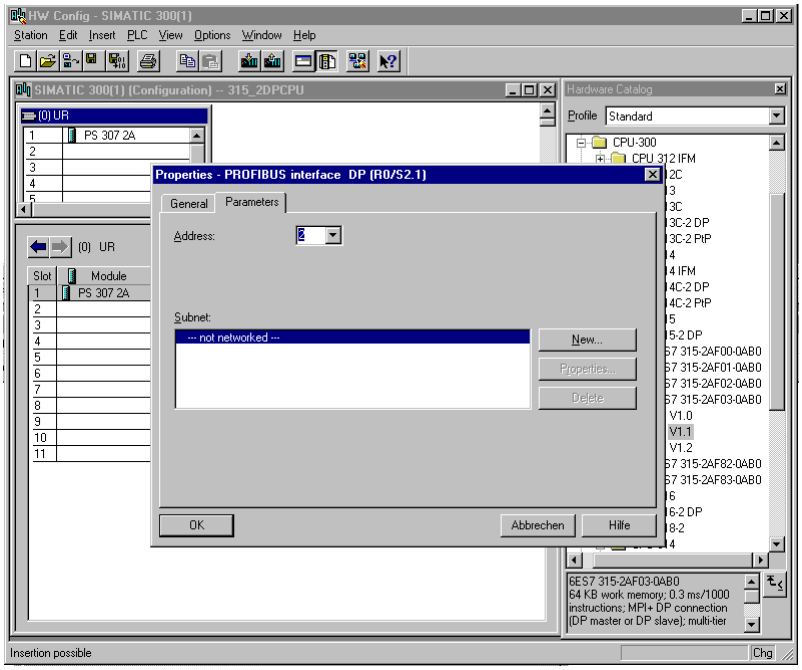


Note: If your hardware differs from what is shown above, then you must select the appropriate modules from the catalog and insert them into the rack.
The part numbers of the individual modules, which are found on the components, are indicated in the footer of the catalog.

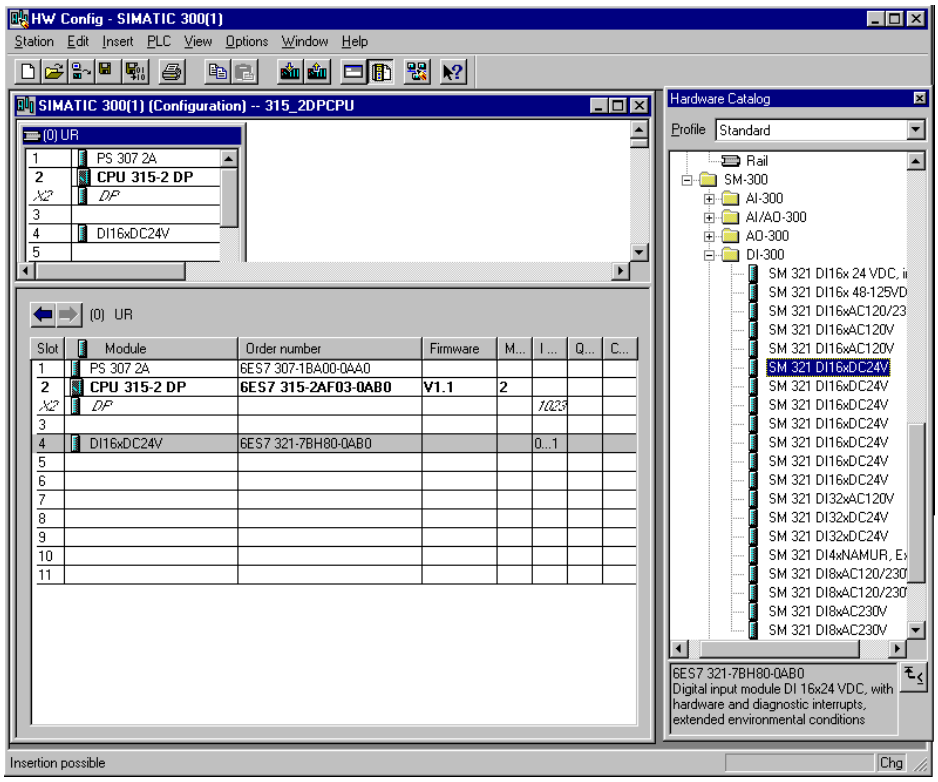
9. In the next step, we drop the CPU 315-2DP into the second card location. This allows for the part number and version of the CPU to be read off. (→ SIMATIC 300 → CPU-300 → CPU 315-2DP → 6ES7 315-2AF03-0AB0 → V1.1).



10. In the following dialog the integrated PROFIBUS- Interface is adjusted. We will not be adjusting anything here, so click **OK** (→ OK).

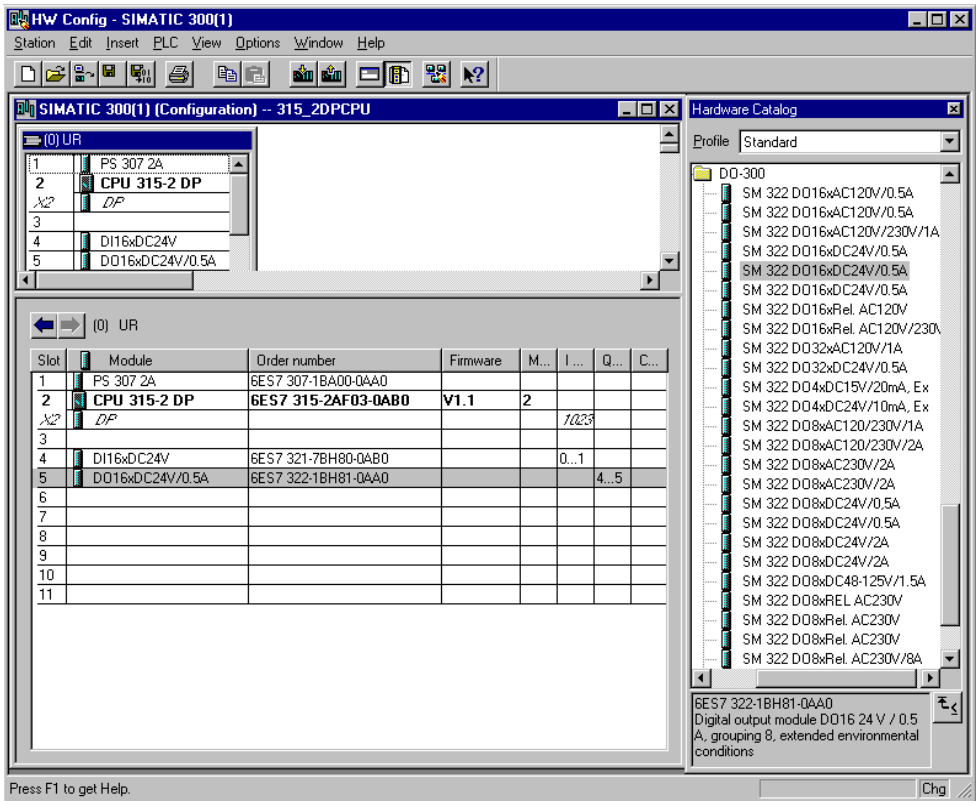


11. In the next step we drag the input module for 16 inputs into the fourth card location. The part number of the module can be read off on the front (→ SIMATIC 300 → SM300 → DI-300 → SM 321 DI16xDC24V).



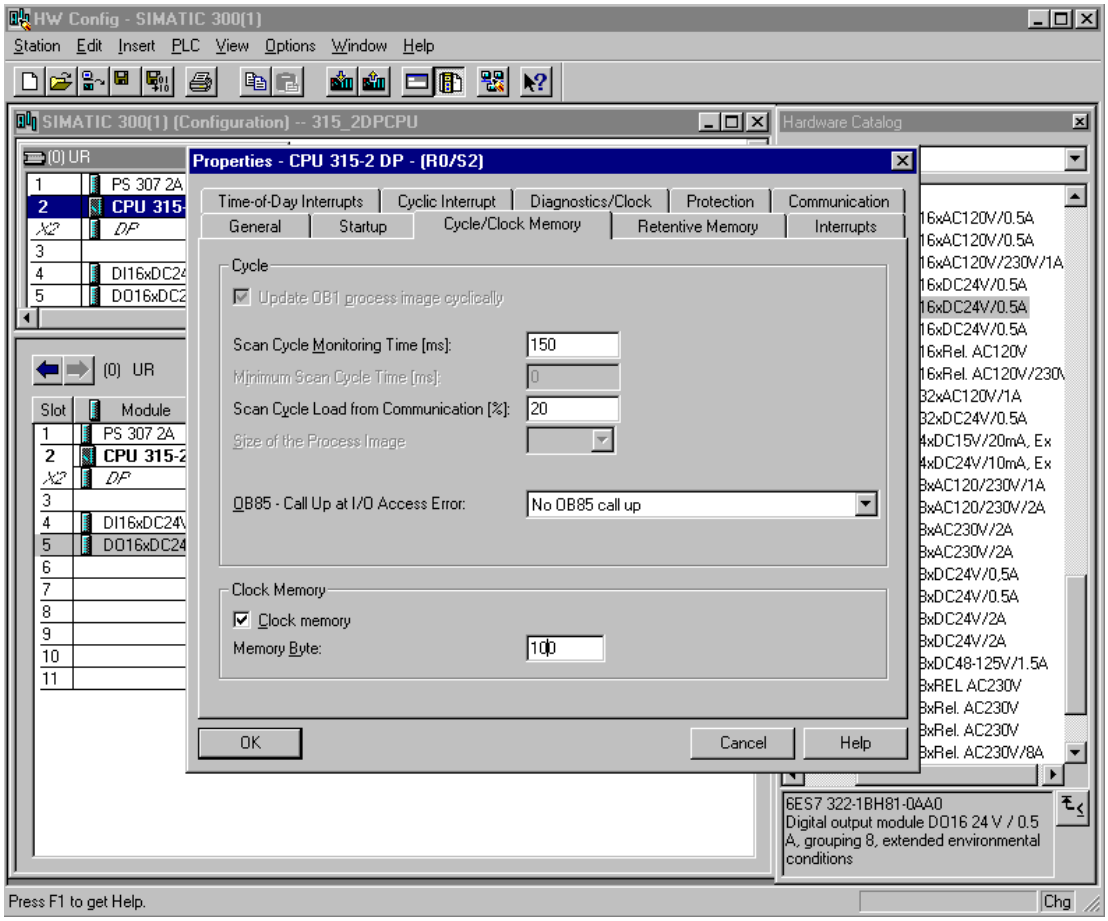
Note: Card location number 3 is reserved for connection modules and must always remain empty. The part number of the module is indicated in the footer of the catalog.

12. In the next step, we drag the output module for 16 exits into the fifth card slot. The part number of the module can be read off on the front (→ SIMATIC 300 → SM300 → DO-300 → SM 322 DO16xDC24V/0,5A).

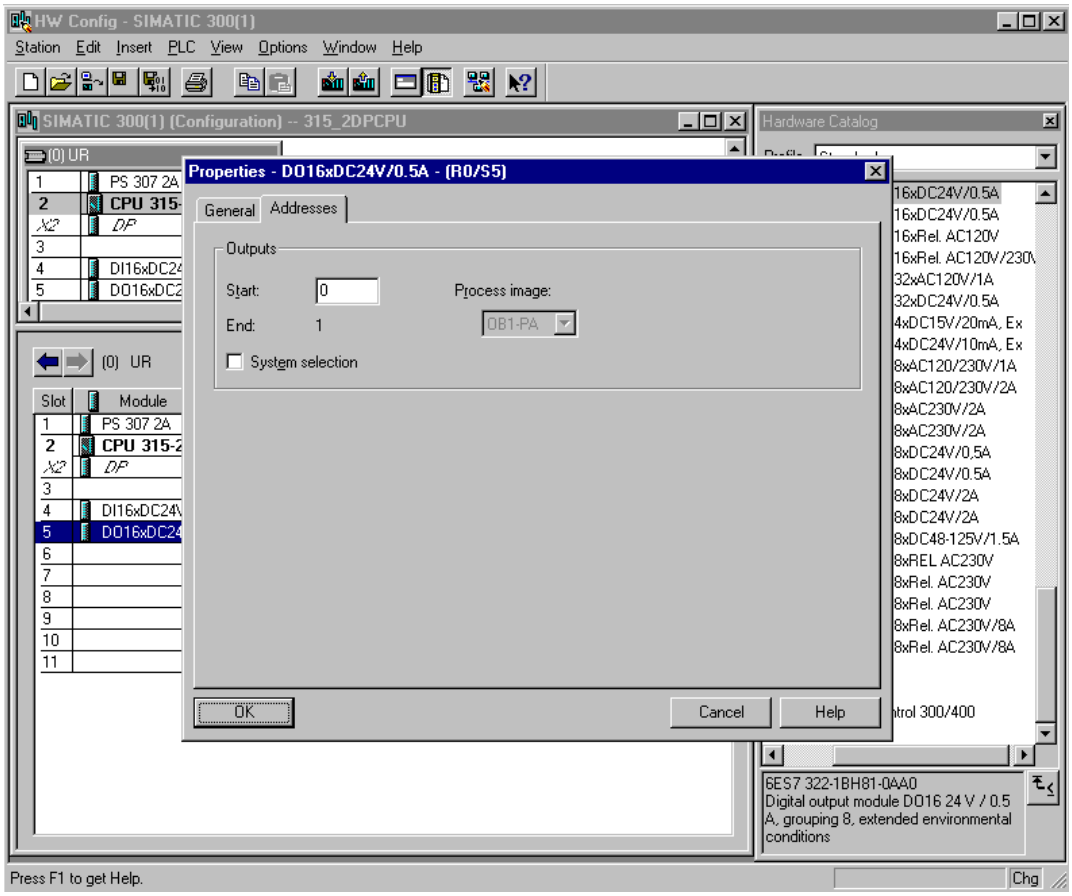


Note: The part number of the module is indicated in the footer of the catalog.

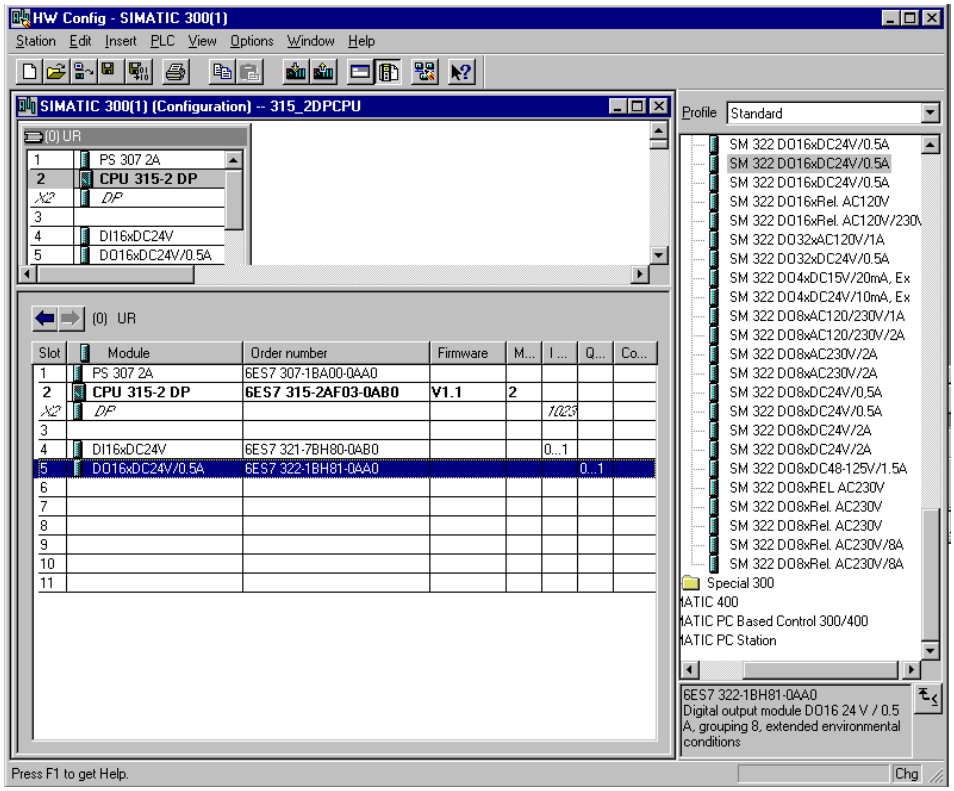
13 The characteristics of some modules can be changed.
(→ Right click CPU 315-2DP module→ insert→object properties → OK).
e.g. A clock memory can be fixed by all CPUs (→ Cycle/Clock memory → v Clock memory → Memory byte 100).



14. The addresses of the I/O modules can be changed only by S7-300 CPUs with integrated PROFIBUS- Interface.
This occurs when the appropriate modules are double clicked and adjusted in the register ‘Addresses‘. These addresses should be noted in each case (otherwise automatic address assignment affects card location-bindings) (→ DO 16xDC24V/0.5A → Addresses → uncheck System selection → 0 → OK).



15. The configuration table can be saved, translated and then downloaded into the PLC by clicking and . The code switch by the CPU should remain on STOP! (→ →)



4. WRITING OF A STEP 7- PROGRAM

The program which can be debugged is written in the statement list (STL) and contains only two lines.
The frequencies of the cycle memory byte MB100 in the activated hardware are given out to an output byte here.

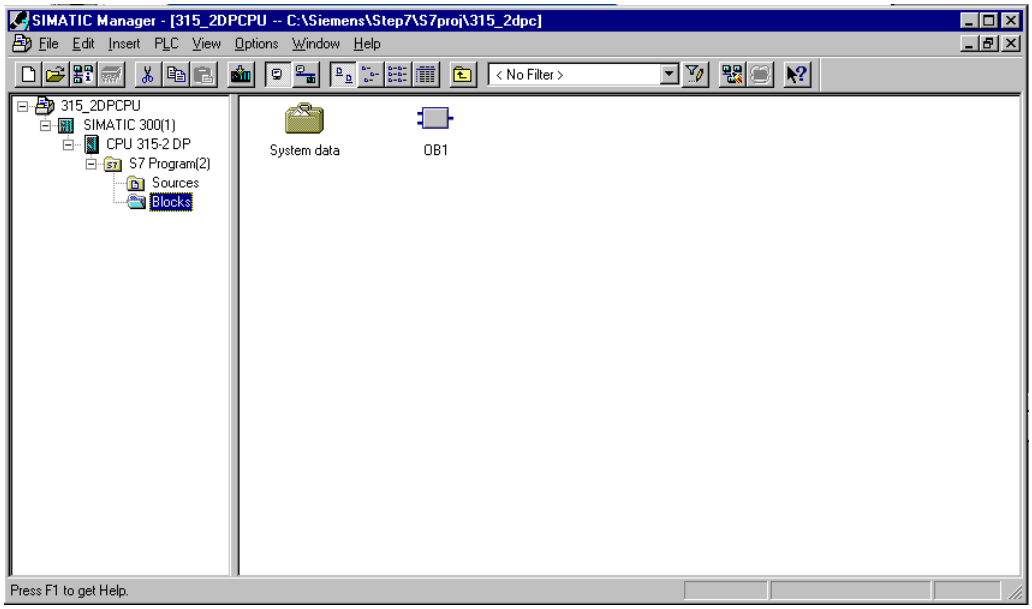
Symbol table:

MB100	clock	clock memory byte
QB0	QB	output display

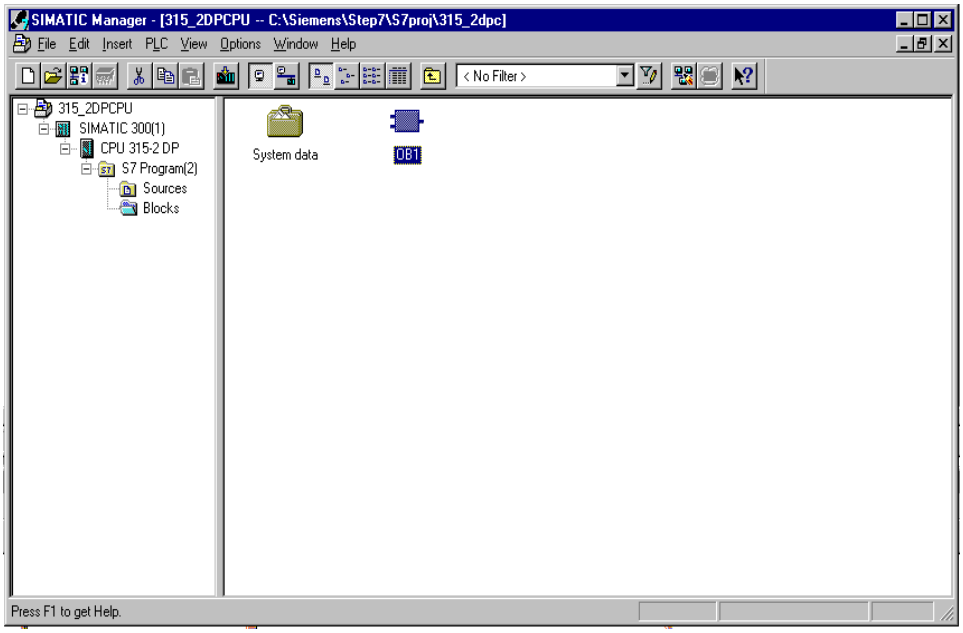
A period length/frequency is assigned to each bit of the clock memory. The following assignments apply:

Bit:	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Period length (s):	2	1.6	1	0.8	0.5	0.4	0.2	0.1
Frequency (Hz):	0.5	0.625	1	1.25	2	2.5	5	10

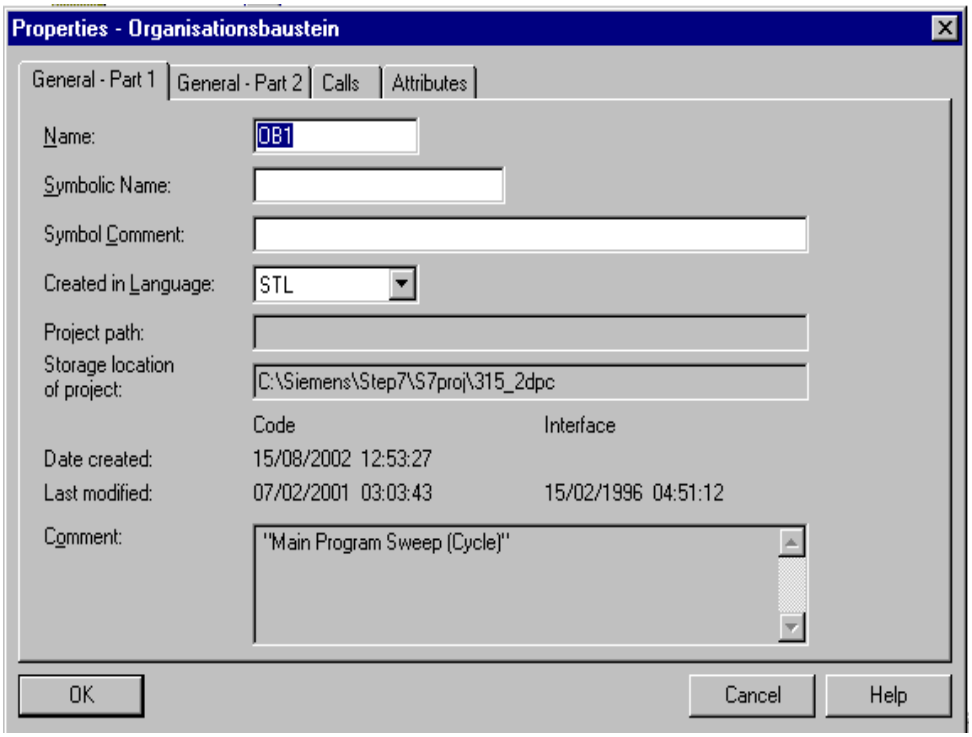
16. In the **SIMATIC Manager** select the folder **Blocks**.(→ SIMATIC Manager →Blocks)



17. In the SIMATIC Manager, double click the block **OB1** (→ OB1).

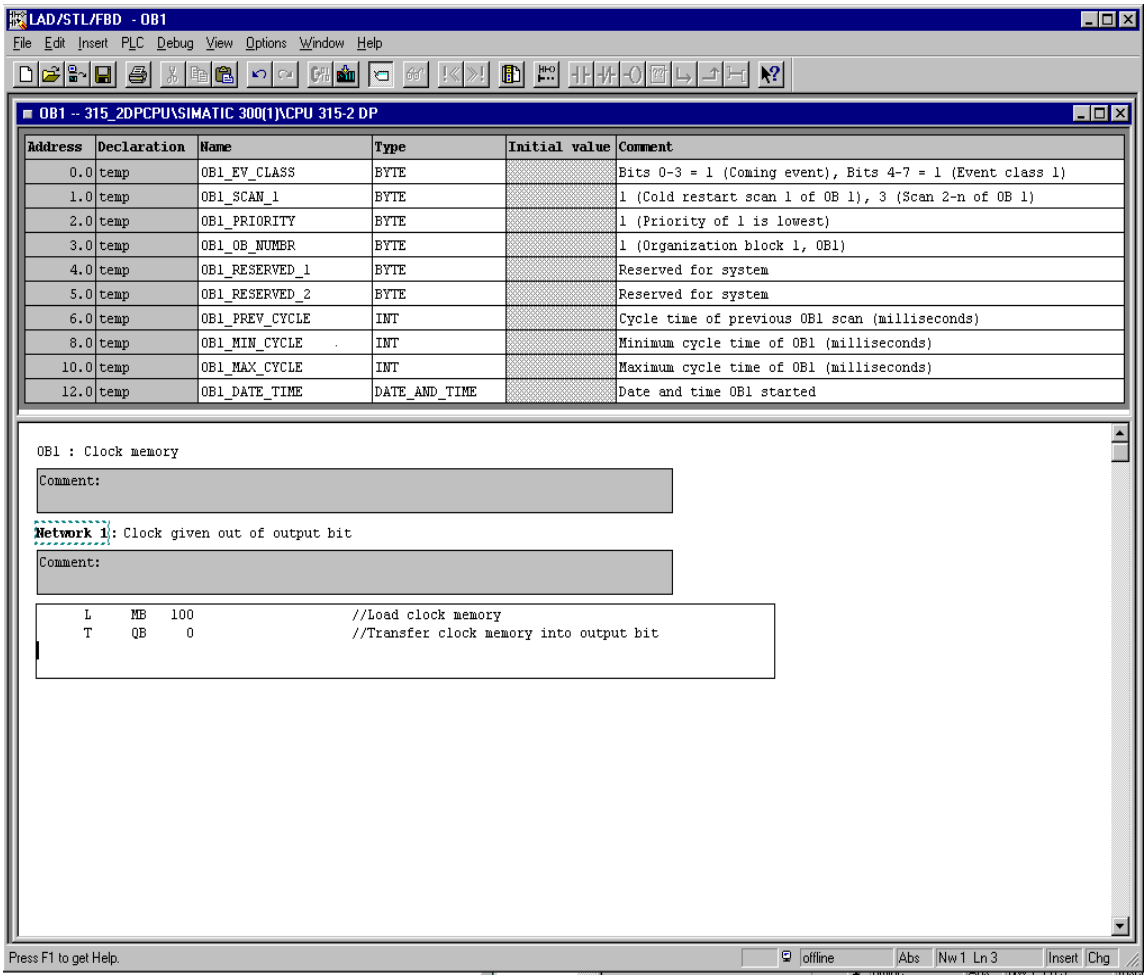


18. Accept the options of the OB1 block with **OK**. (→OK).



19. There is an editor provided with **LAD, STL, FBD: Program blocks** that gives you the possibility to edit your STEP 7- Program accordingly. In order to do this, the organization block OB1 should already be opened within the first network. The first network must be highlighted in order to create the first operation. Now you can write your first STEP 7- Program. Individual programs in STEP 7 are usually divided into networks. A new network can be opened by clicking on the network symbol  .

Note: Comments on program documentation are separated from the program instructions by the character sequence '//'.



In the network

```
L   MB 100   //Line 1
T   QB 0     //Line 2
```

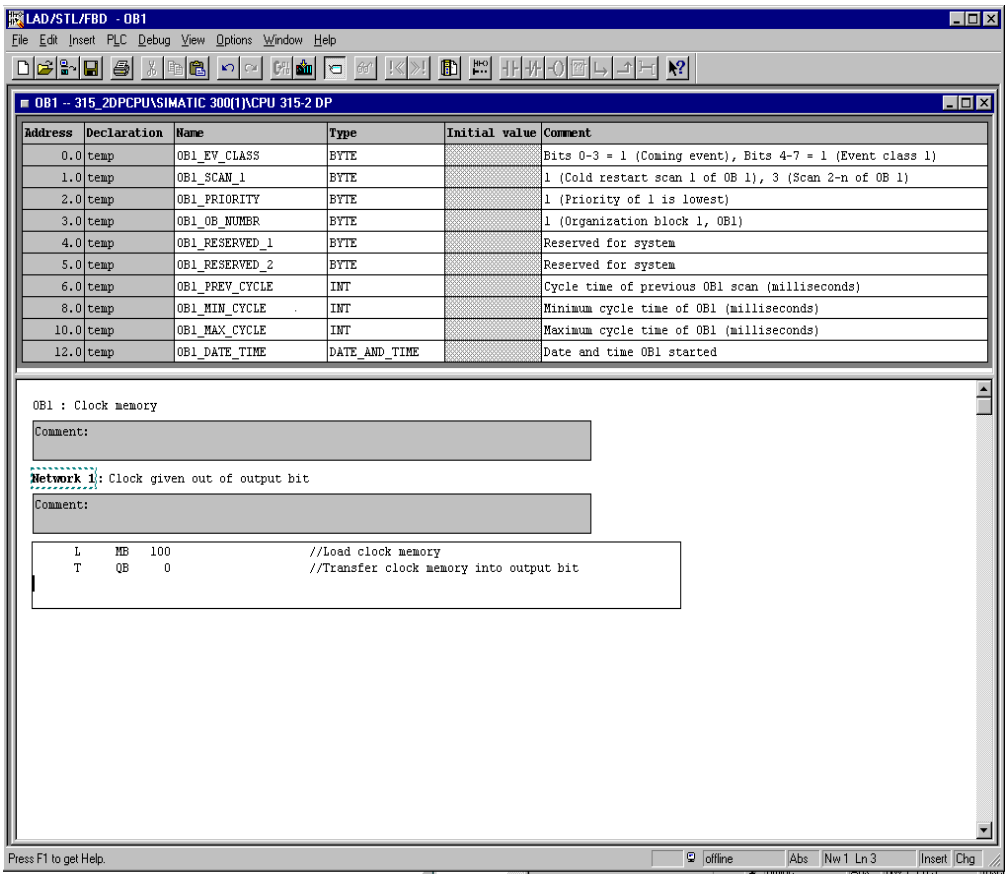
line 1 activates the clock memory byte and line 2 transfers the index into the output byte. The 8 bits of the output byte in the different frequencies of the clock memory bit should flash.

Note: The address of the output byte can be different depending on hardware configuration.

5. DEBUGGING OF THE STEP 7- PROGRAM

The STEP 7- Program to be debugged can now be loaded into the PLC. For this example, only OB1 will be debugged.

19. Save the organization block by clicking and download the program by clicking . The code switch by the CPU should remain on STOP! (→ , →)



20. By switching the code switch to ‘RUN‘, the program can be started. After it is started, the program in ‘OB1‘ can be observed by clicking the symbol  (→ ).