WinPLC7

Programming
Simulation
Diagnostics

Version 6
User Manual

Revision: 03.08.2015

1 Introduction

1.1 What features does WinPLC7 offer?

WinPLC7 is a complete programming system for S7-300®, S7-400® and compatible PLCs (VIPA S7-PLCs).
WinPLC7 includes a **Software-PLC** that simplifies the simulation of S7 programs. You can monitor digital and analog inputs and outputs by means of a graphical S7-300® -screen (PLC Mask). At the same time, you can also monitor a block in the editor.

The hardware configurator provides special support for S7-PLCs supplied by **VIPA GmbH**. It offers a simple user interface for the configuration of VIPA-CPUs and modules.
Most of the hardware modules that are currently available from VIPA GmbH have already been implemented in the hardware configurator, i.e. it is not necessary to import these using GSD files.

1.2 Notes for WinPLC7 V1 users

If you have only worked with version 1 of WinPLC7, note the following:

- In Version 1 you can only save one PLC program in a project. Since version 2, you can store multiple PLC programs in a project.
- In Version 5, the project is called a **solution**. You can store several projects in a solution - each with an independent PLC program.
- To import a V1-project into WinPLC7 V5 proceed as follows:
  1. Create a new solution with V5
  2. Select File->Import->WinPLC7 V1 project

1.3 Notes for WinPLC7 V2-V5 users

If you have worked with version 2-5 of WinPLC7, note the following:

- The project structure is still the same, i.e. you can simply open your old projects (file extension .WS7) with version 6.
- The definition of a project has changed:
  A WinPLC7 project is referred to as a "solution", and a sub-project is now called a "project".

1.4 New features in version 6

- New Licensing-Manager. Now you can deactivate your license on the current PC and activate the license on another PC.
  You can even remove an activation from a PC that is not available any more (damaged or vanished). This is possible because the activation is fully cloud based. Only when activating or deactivating an internet connection is required.
- A number of small changes so that WinPLC7 is compatible with Windows 10.

1.5 New features in version 5

- Auto-completion "S7-Intellisense"
  During programming, the matching operands, symbols or variables are displayed.
- Collapsible networks
  to provide a better overview when editing large blocks.
- Moving networks with drag and drop
- Inserting and deleting networks easily, quickly and intuitive: to the left of each network you will see a number of symbols that you can use to paste, copy and delete networks.
- New function "Go to network title"
- Convenient ToDo list
- Intelligent detailed block comparison
- Voice recognition navigation of the user interface
- Voice output from the variable monitoring screen
- Status variable via web server: Watch the variables on a smartphone via WLAN
- Live view of the address locations
- Block synchronization
  with a single click, you can load the modified blocks into the PLC.
- Use the CPU-control center screen to monitor the most important CPU information at a glance.
- The integrated S7-Soft-PLC is accessible via TCP / IP.

1.6 New features in version 4

- Consistency-check with straightforward trouble shooting facilities
- Global dialog-controlled CALL-statement processing
- Sequencer wizard
- New project-manager with context-sensitive quick-start buttons
- Network view in the hardware configurator
- Object list for OFF-LINE and ON-LINE views
- Stations are accessible via Ethernet with temporary assignment of IP parameters
- A new and comprehensive online help system
- Compatible with SIEMENS MPI adapters
- Remote maintenance via analog/ISDN telephone lines is available when using SIEMENS Teleservice V6 (e.g. TS a**dapter II)**
- Support for Windows Vista, Windows 7

1.7 System requirements

WinPLC7 is compatible with the following systems:

Hardware:

- 500 MB RAM or more
- Approx. 200 MB of hard disk space

Operating system:

- Windows10, all editions, 32- or 64 Bit.
- Windows8, all editions, 32- or 64 Bit.
- Windows7, all editions, 32- or 64 Bit.
- Windows XP (not recommended)

A working version of the Internet Explorer is necessary, since this is required for the help system.

1.8 Installation

Separate installation hints are supplied with the software.

The installation requires a serial number.
This is included with the package that is shipped to the end use.

1.9 Information about this manual

The sample programs in this manual are based on the English syntax.

In the "Language" tab, you can select your specific syntax settings under *Extras->Settings,* provided that you have not yet saved any blocks in the project.

2 Quick start and brief tutorial

2.1 Overview of the user interface

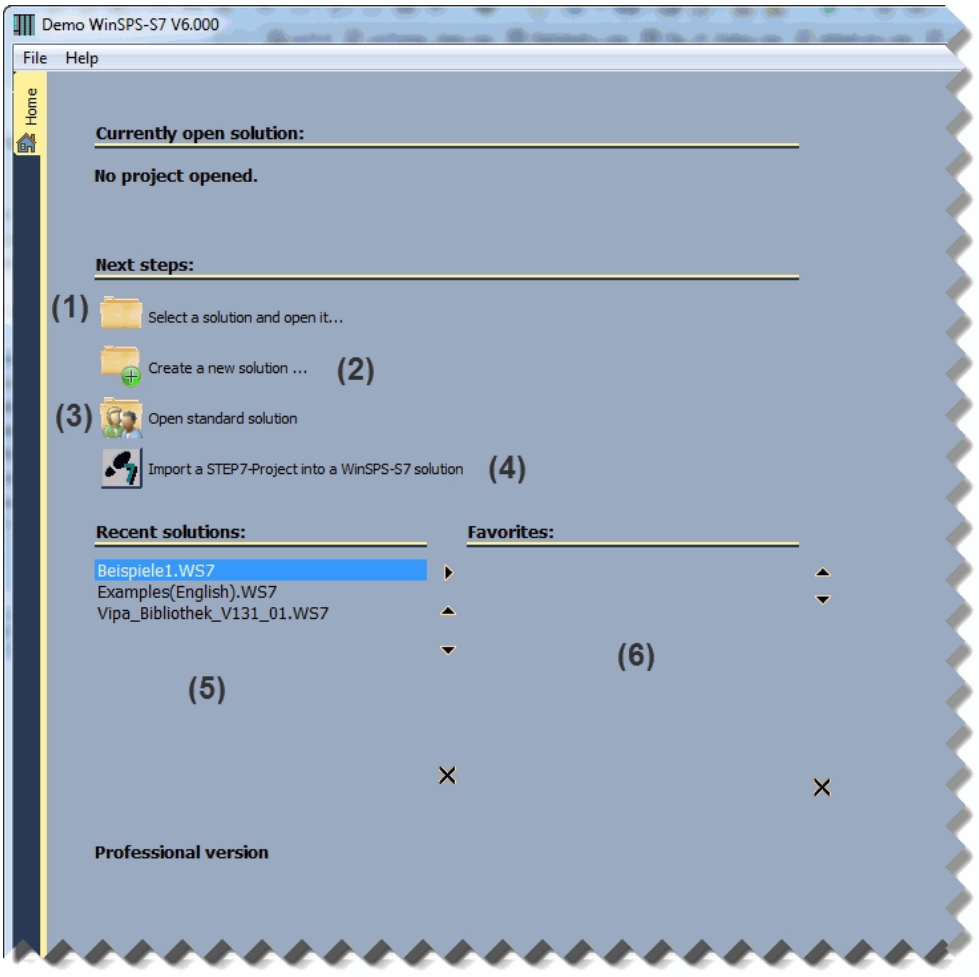


Fig.: The home screen of WinPLC7 V5

When WinPLC7 starts, it displays the **home screen** with the following items:

1. **Select a solution and open it...**

   you can select and open a WinPLC7 solution using the windows open dialog.
   Hint for Windows7: you can also use the search field in the open dialog to search for *.WS7 files.

2. **Create a new solution**
   you can create a new, blank solution. You can also specify the save location in the dialog.

3. **Open standard solution opens** the standard user solution. This is located in the "WinPLC7" folder and contains your own files.

4. **Import a STEP7 project into a WinPLC7 solution**
   you can import a STEP7 project into a WinPLC7 solution. The STEP7 project can be zipped.

5. List of the recent solutions
   This is the list of recently used solutions.
6. List of the favorite solutions
   you can collect your favorite WinPLC7 solutions in this list.

Note:
The file extension **.WS7** is associated with WinPLC7 so that you can open WinPLC7 solutions with a double click.
In this case, WinPLC7 must be closed.
You can also use the Windows search method to locate for the file extension **.WS7** and open it with a double click.

An open solution displays the following user interface:

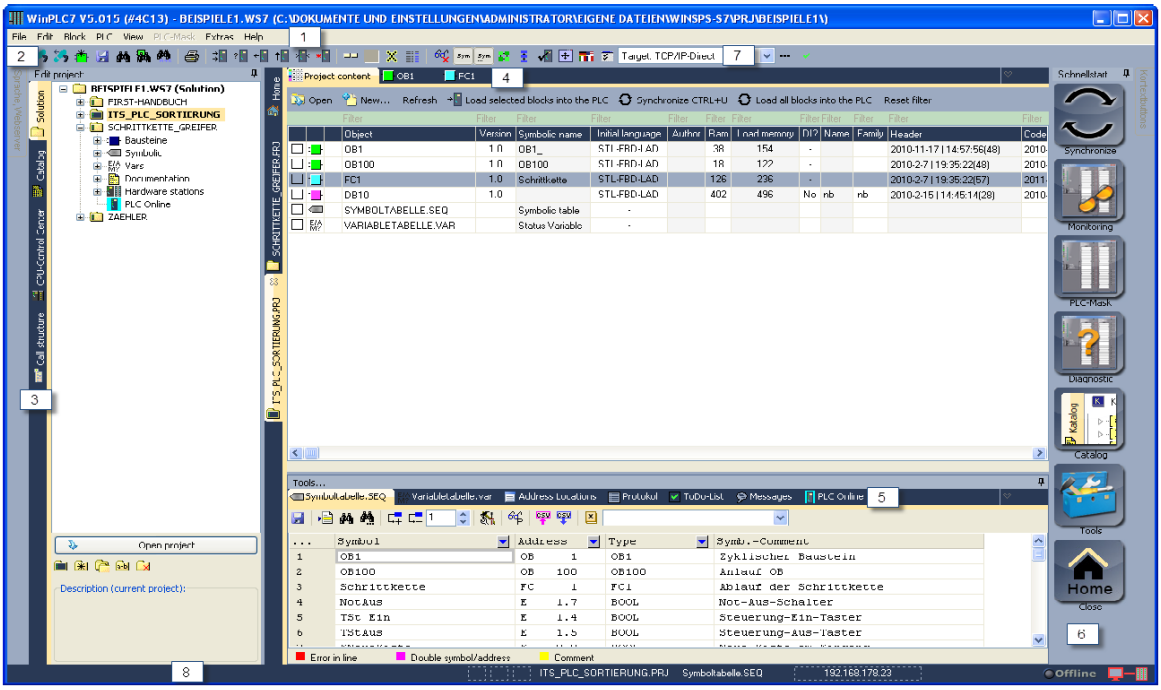


Fig.: WinPLC7 with an open solution

The most important control elements are:

1. **Main menu**
2. **Speed bar**
3. **Project window with the following tab sheets:**
   - Solution: solution tree: open, delete, duplicate projects in the solution.
   - Catalog: to use with LAD and FBD
   - CPU Control Center: information about the connected PLC
   - CALL structure: Displays the structure of the PLC program.
4. **Editor area with the following tab sheets:**
   - project content
   - OB1
   - FC1
5. **Tool window with the following tab sheets:**
   - symbolic editor
   - monitoring variable window
   - address locations
   - protocol
   - ToDo list
   - messages
   - PLC-Online
6. **Quick start panel** with the icons: synchronize, block monitoring, PLC mask window, diagnostics (PLC state), catalog, tools-window and "close solution"
7. **Target settings "Software-PLC" or "PLC-external".** If you select the mode "Software-PLC", you can work with the Software-PLC of WinPLC7 to simulate a PLC program.
8. **Status bar with short hints and messages**

Additional information about the target settings:
In this list you can select whether you want to work with the internal simulator or with an external S7 PLC.
Here you can also define the communication path (Serial RS232, Netlink-Pro, Netlink, Ethernet, Simatic® Net, etc.) that will be used to access the external S7 PLC.

2.2 Quick start tutorial on the based on a small program

This section describes how you would go about creating and simulating a WinPLC7 program and how you would go about transferring it to the S7 PLC.

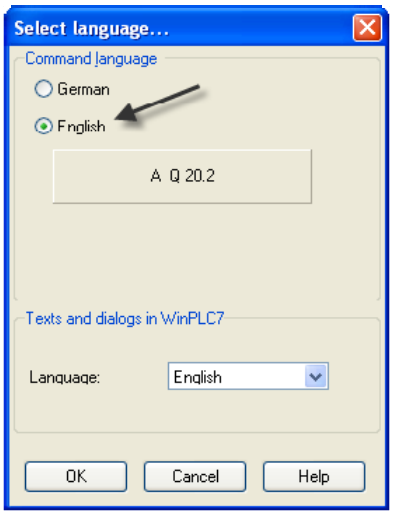
Procedure in telegram style:

1. Create new solution
2. Define the symbolic names for the input/output/etc.
3. Create the block FC1
4. Create the block OB1
5. Select the Software-PLC as the target system and define the clock memory byte.
6. Transfer blocks to the target system
7. Change to PLC to RUN mode
8. Monitor OB1
9. Status variable (monitor the variable)
10. Connect to external PLC
11. Hardware configuration of a real S7-PLC

The following paragraphs contain a detailed explanation of the different items.

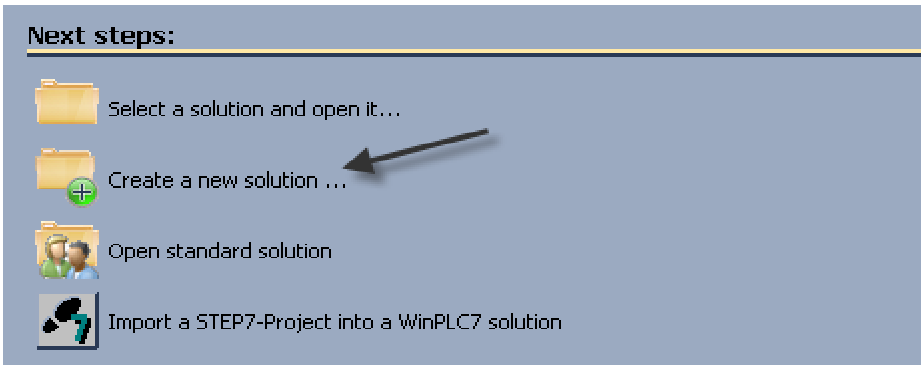
2.2.1 Create a new solution

Before we start, check if the English language settings are active.
Close any open solution.
Select menu item **File->Language**. In the dialog, check the "English" radio box. The sample code will display the STEP7 command "A Q 20.2":

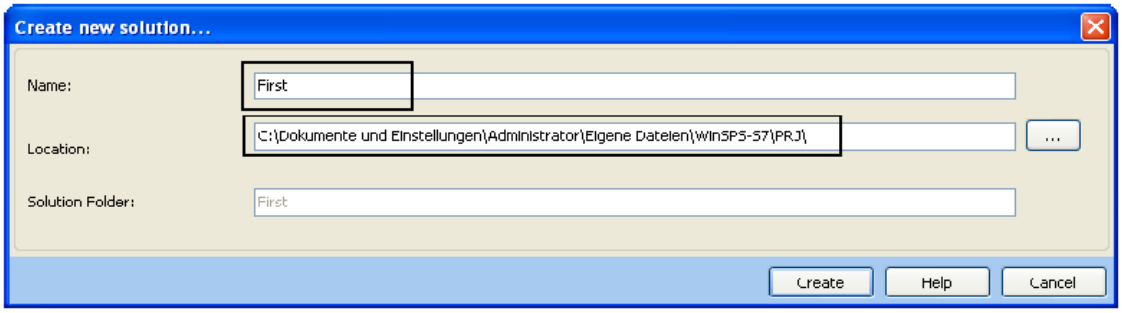


Next we begin to create a new solution:

When WinPLC7 starts, the home screen is displayed (see paragraph 2.1). Click on the button "**Create a new solution**":



An input window is displayed:



Enter the name of the new project (e.g. "First") and specify the location. Then press the button "**Create**".
This creates and opens "First" as the new solution. WinPLC7 appears as follows:

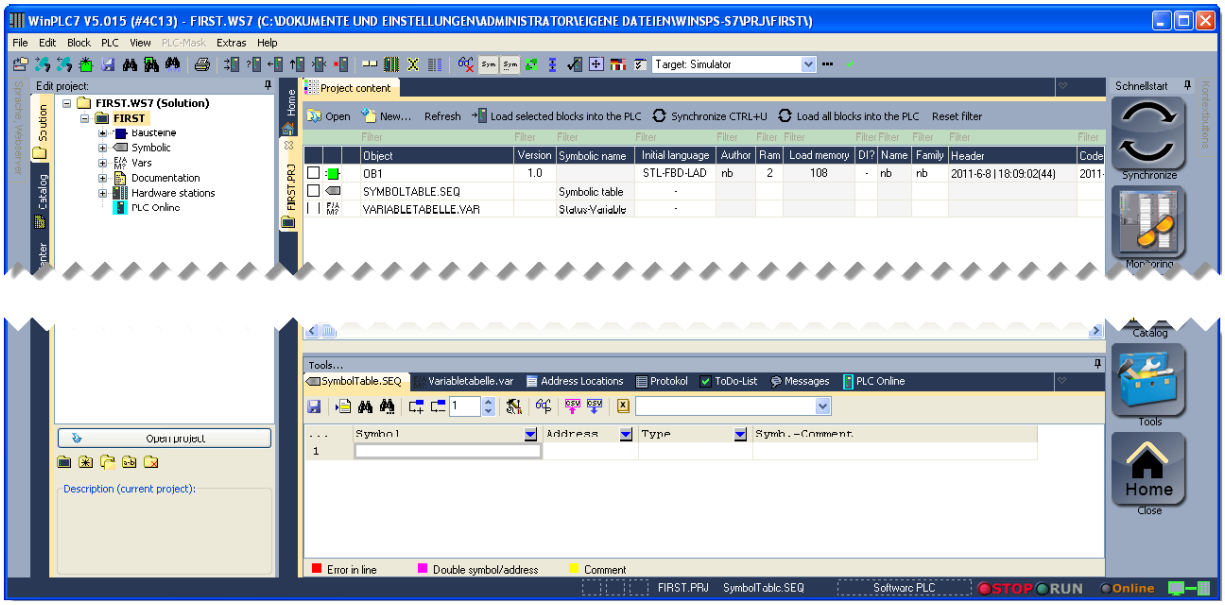


Fig.: The new solution was created.

The project window with the Solution tree is visible at the left. Since you have created a new solution, only one (1) project with the name "First" exists. The name of the solution is always assigned to the first project.
If you right click the project window with the mouse, you can create or delete additional projects. More information will follow later.

Summary:
We have created a new solution with the name "First". The solution "First" contains a project "First" where we will now create our blocks.

2.2.2 Create the symbolic table

It is common practice to start by creating a symbolic table. This increases the legibility of the program and enables others to quickly follow the flow of the program. In the symbolic table a symbolic name (e.g. switch1) is assigned to absolute addresses (e.g. I0.0).

With the new project an empty symbolic table is available:

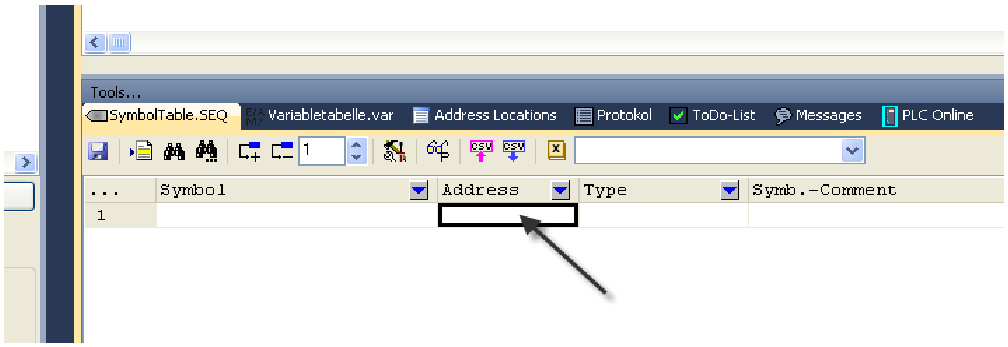


Fig.: The empty symbolic editor

Select the Address column and enter "M255.0" via the keyboard. Next, select the Symbol column and enter "Sensor1":

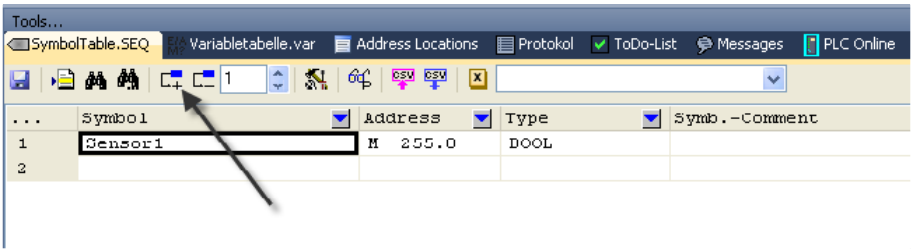


Fig.: Press the button with the plus sign 2 times

The resulting window appears as follows:

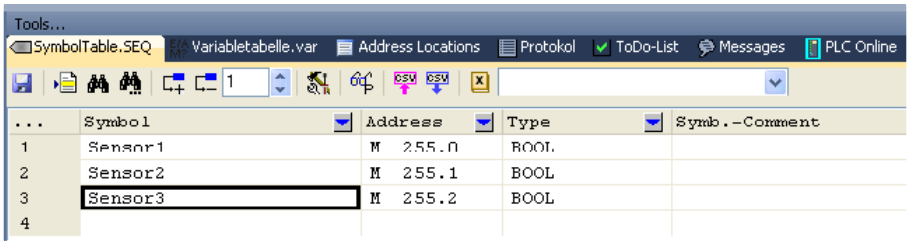


Fig.: 3 symbols were created

Using the "+" button to create the following icons quickly and logically.
Select the Address column that is located below the last entry. Enter "QW0", select the "Symbol" column and enter "Counter1".
You must change the Type from "WORD" to "INT"
Continue by pressing the "+" button twice.

As a result the table looks as follows:

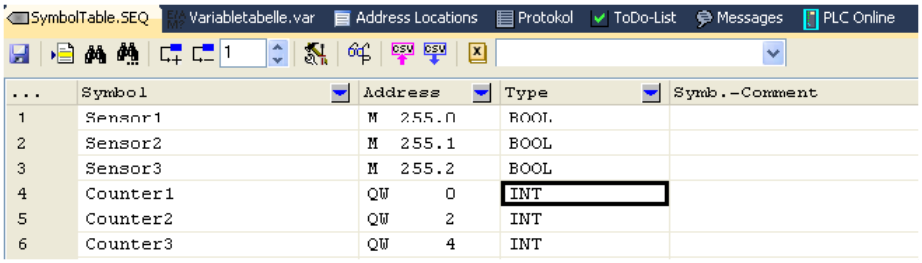


Fig.: At this point you have created a total of 6 symbols.

Hint:
Click the right mouse key to display a context-sensitive menu with suitable commands for the symbolic editor.

Summary:
We have created a symbolic table. The symbols that were created can now be used for programming purposes.

2.2.3 Create FC1

We will continue by creating block FC1. In this block we will write the PLC program for a counter. FC1 is then called repeatedly in OB1.
Click the button "New" in the "Project content" tab sheet. Then choose the item "FC":

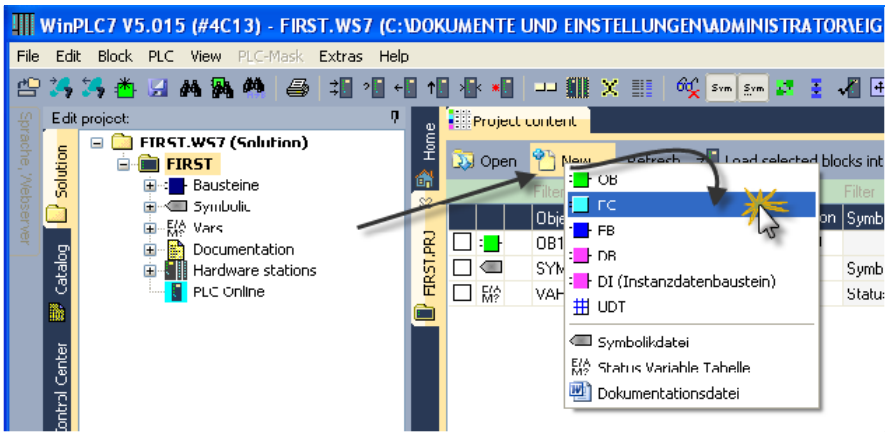


Fig.: Create a new FC1

The software determines now the next free FC block number. In our case this is "FC1". To create the FC1, simply click on the button "Create block FC1":

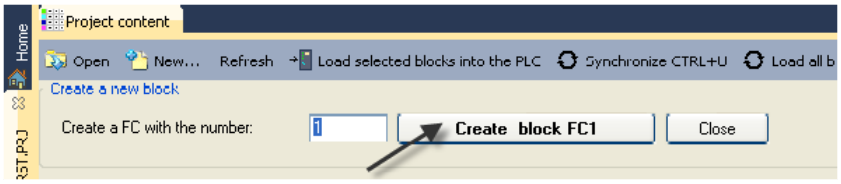


Fig.: Create block FC1

Now we add three block parameters to FC1:

1. Pulse (declaration=IN, Type=BOOL)
2. Counter (declaration=OUT, Type=INT)
3. Flag (declaration=IN_OUT, Type=BOOL)

Click in column "Name" and follow the instructions:

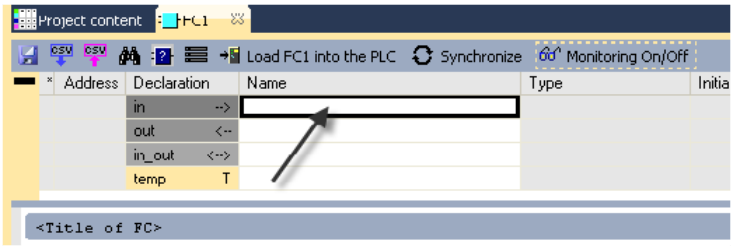


Fig.: Header of FC1 (empty)

1. Enter "Pulse" in the first line of column "Name" and press the RETURN key
2. The cursor jumps to column "Type". Enter "bool" and press the RETURN key again.
3. The cursor jumps to the "Comment" column . Hit the RETURN key and the cursor is back to the "Name" column.
4. Move the cursor to the declaration "out" to column "Name" and enter the next parameter: "MyCounter". Press RETURN and enter "int".
5. Move the cursor to the declaration "in_out" and define the variable "Flag" with data type "bool"

The third variable completes the block header:

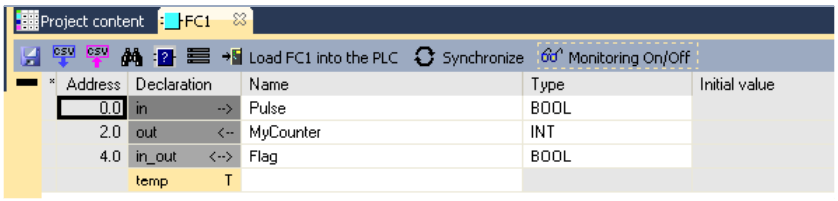


Fig.: Header of FC1 (finished)

Make sure that the variables are defined in the correct declaration area: Pulse (in), MyCounter (out) and Flag (in_out). Otherwise the PLC program will not work properly.

We will now create three networks in FC1.
The first network (network 1) is created in FBD (Function block diagram). In network 1 click on "FBD" representation:

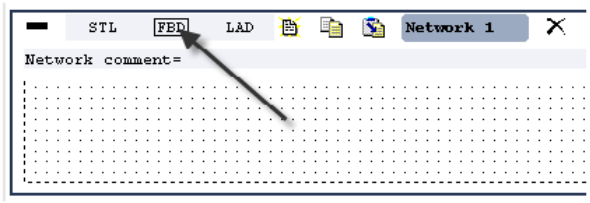


Fig.: Selecting FBD representation

Next, execute a double click on item "ADD_I" in the catalog:

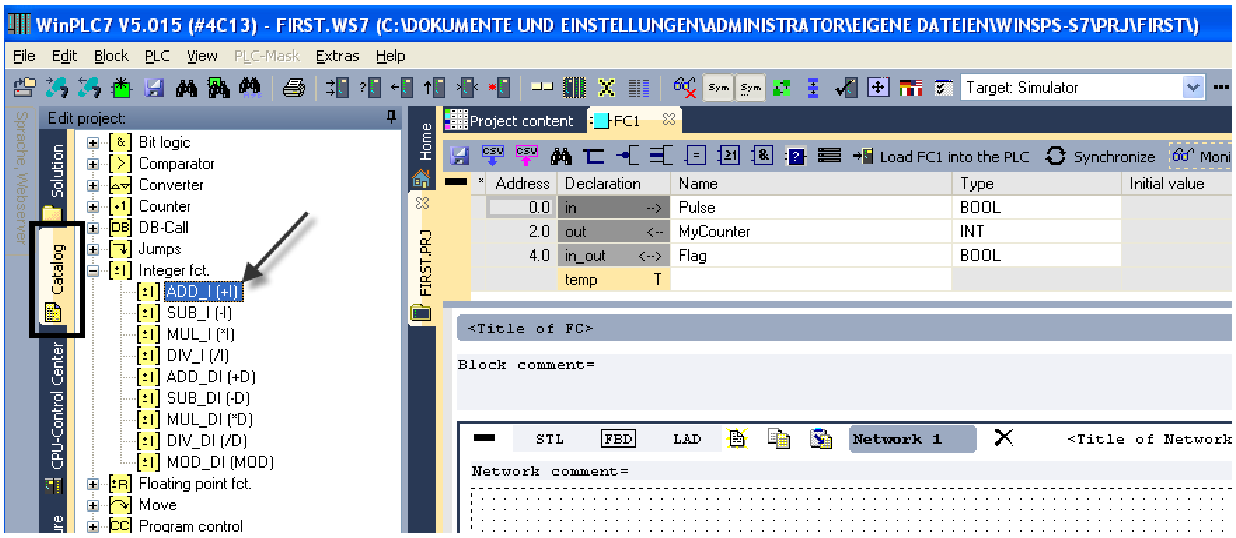


Fig.: Inserting a ADD block into network 1 with a double click

The "ADD" block is displayed:

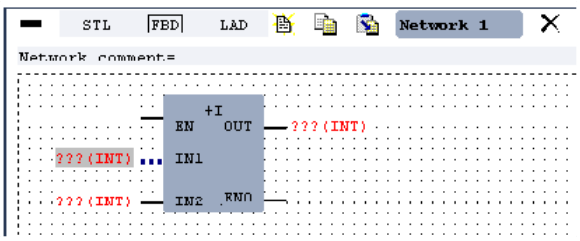
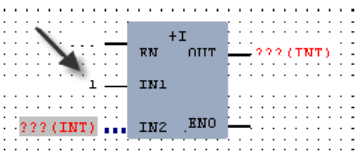
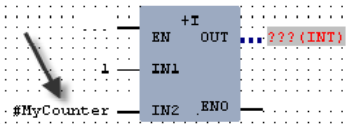


Fig.: New "ADD" block

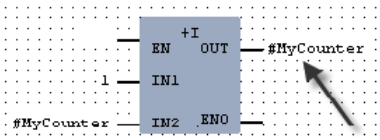
The next steps describes how you can complete network 1:



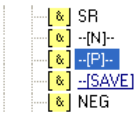
Click on the "???" at input "in1" and enter "1". Press ENTER to confirm.



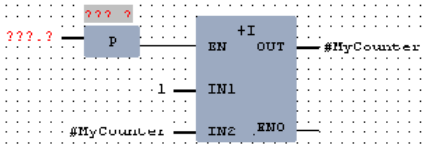
Click on the "???" at input "in2" and enter "#". The IntelliSense window appears and shows suitable variables. In this case, only the "#MyCounter" variable appears. Simply press ENTER to insert "#MyCounter".



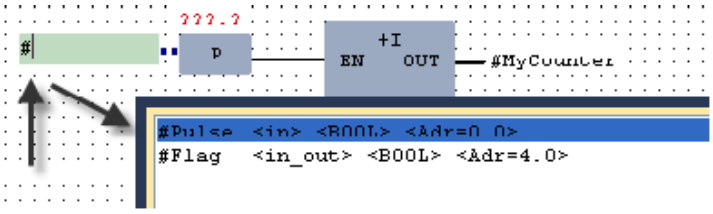
Click on the "???" at output "OUT" and enter "#". Insert the "#MyCounter" variable.



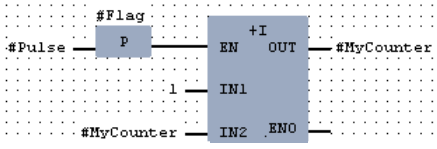
Select the EN-input line and doubleclick the item "**--[P]--**" in the catalog. You can find this item in the "Bit logic" node.



The box "**--[P]--**" was inserted.

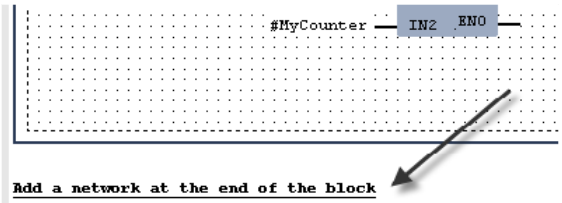


Now insert the variable "Pulse" on the left side of the P-Box . For this purpose, enter "#" and select "#Pulse". Then press enter. Replace the "???" above the box with the variable "#Flag".



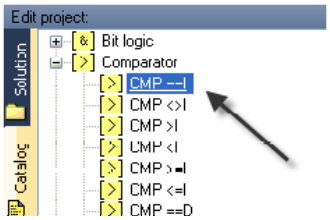
The completed network.

The result of this network is: when variable "#Pulse" creates a positive edge, the value of "#MyCounter" is increased by 1.

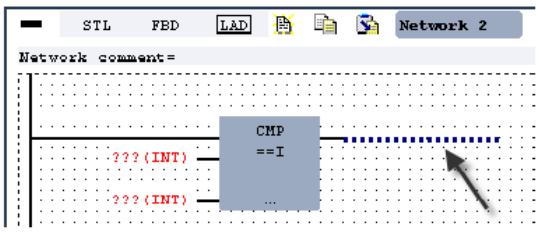


To add two networks, click twice on this text button.

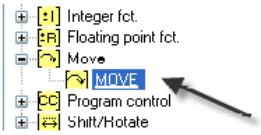
Change network 2 to "LAD" representation



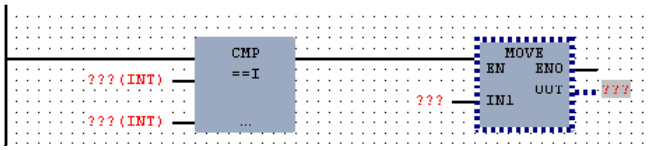
Insert a comparator with a double click on the "Compare CMP==I" node in the catalog



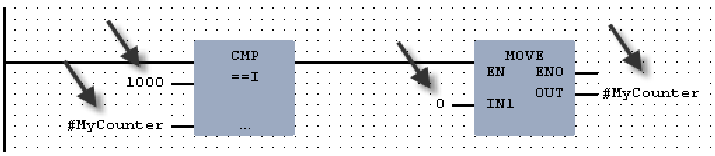
Click on the output line of the comparator box



Insert a "Move" block with a double click. (you can also drag and drop this item)

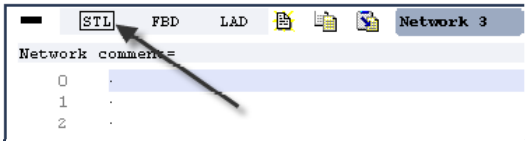


"Move" block is inserted



Edit the two input lines of the compare block. Change the first input to "1000" and the second input to "#MyCounter".
Enter "0" on the left side of the "Move" block and "#MyCounter" on the right side

Now we will complete the last nework: network 3
We use "STL" representation to edit this network.
Click on the "STL" button of network 3:



Insert these STL code:

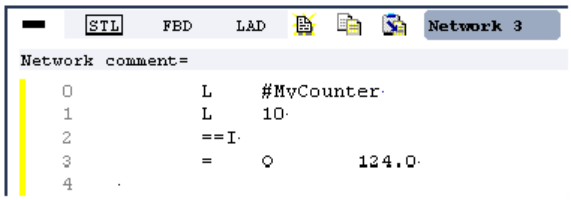


Fig.: Network 3

When completed, FC1 looks like this:

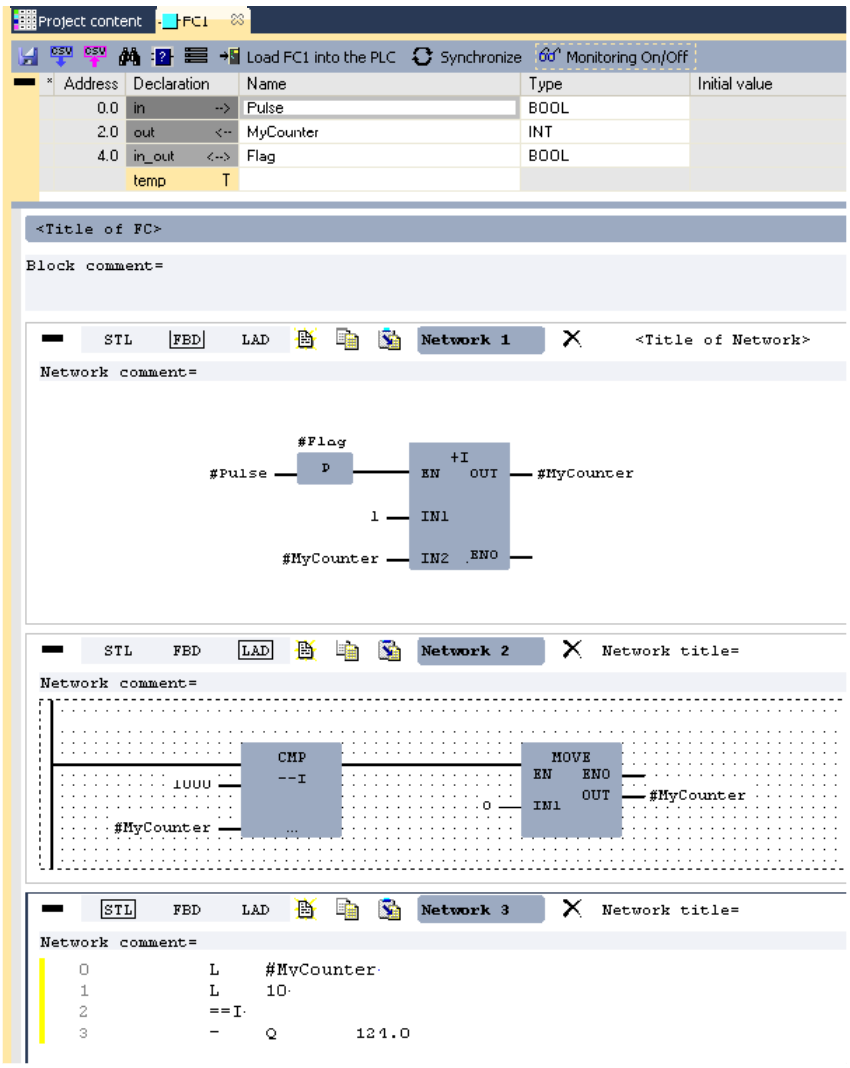


Fig.: FC1, finished

2.2.4 Edit the OB1

The next step is to edit the block OB1. Click on the tab sheet "**Project content**" to display the blocks of the project.
OB1 (blank) already exists. To open it, double click on the item "OB1":

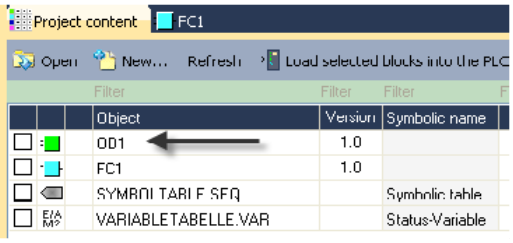


Fig.: FC1, completed

Change the representation of network 1 to "STL" and enter:

CALL FC

When you insert "CALL FC", the IntelliSense Window appears and shows all the FCs in the project:

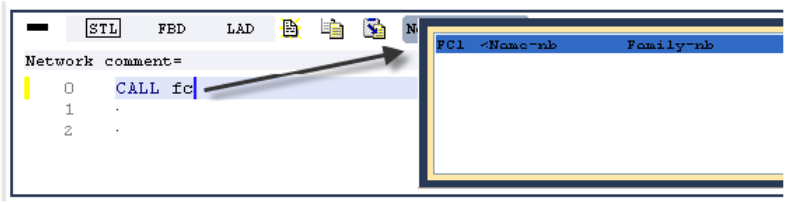


Fig.: IntelliSense window

Press the ENTER key on the keyboard to let the IntelliSense function complete the STL line. Press ENTER again to insert the block parameter of FC1:

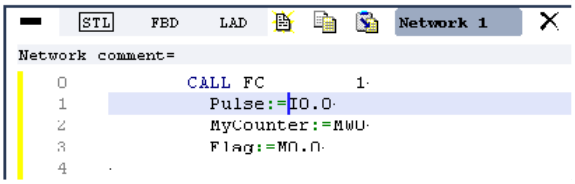


Fig.: Parameter of FC1 was inserted

FC1 is inserted with the current default parameters and address 0. Now we must replace the default parameters. Modify the actual parameter as shown in the next image:

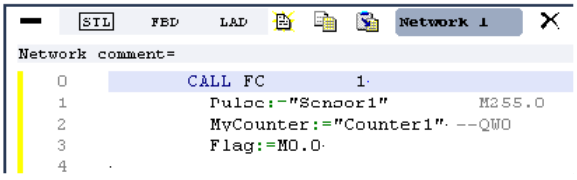
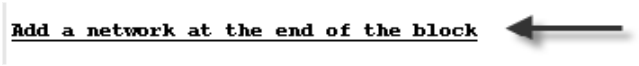


Fig.: new current parameters of the "CALL FC1"

Since OB1 needs three networks, click twice on the "Add" button located at the end of the OB1:



Change the representation of network 2 to "FBD" and network 3 to "LAD".
Click in the empty code area of network 2 to select this network. Insert the CALL to FC1 with a double click on the node "FC1":

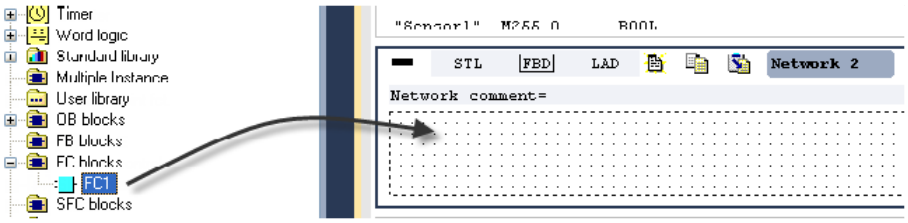


Fig.: Insert the CALL FC1 statement with a double click

The FC1 block as it appears in network two:

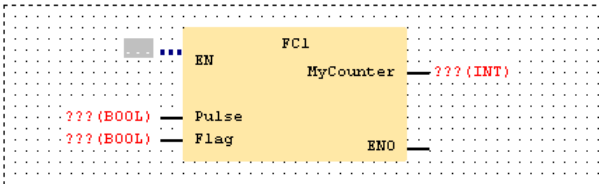


Fig.: "CALL FC1" is inserted

Change the actual parameters as follows:

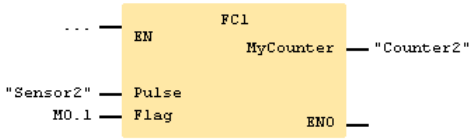


Fig.: "CALL FC1" is inserted

Click in the empty code area of network 3 and insert the FC1 again using a double click on node "FC1":

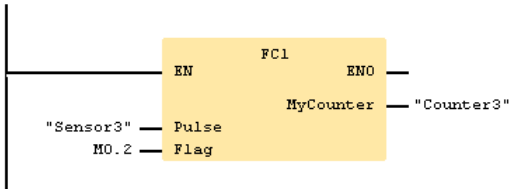


Fig.: "CALL FC1" in the network 3

The default parameters were previously changed to suitable parameters.

At this point, OB1 has been completed.
To make sure that the PLC program works properly, check the input and output parameters of the "CALL FC1" of each network again:

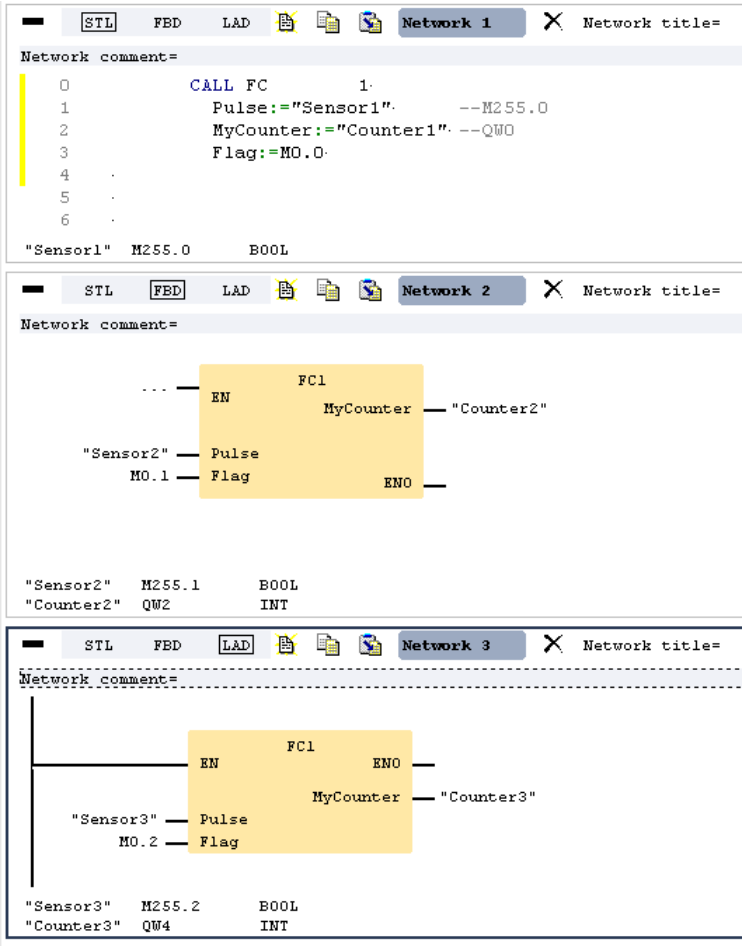


Fig.: OB1 with three networks

2.2.5 Change to simulator mode and set the clock memory byte

In our program we use flag byte MB255. We now set the "clock memory byte" of the Software-PLC to this address. The result is that the bits of MB255 are toggled at different frequencies.

In this example we use the Software-PLC, so you must ensure that the target setting is set to "Simulator". Continue by pressing the "..." button located on the right:

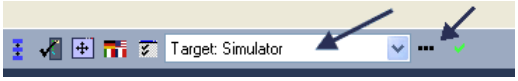


Fig.: Target setting

The Software-PLC settings will be displayed.

Change the clock memory byte from "-1" to "255":

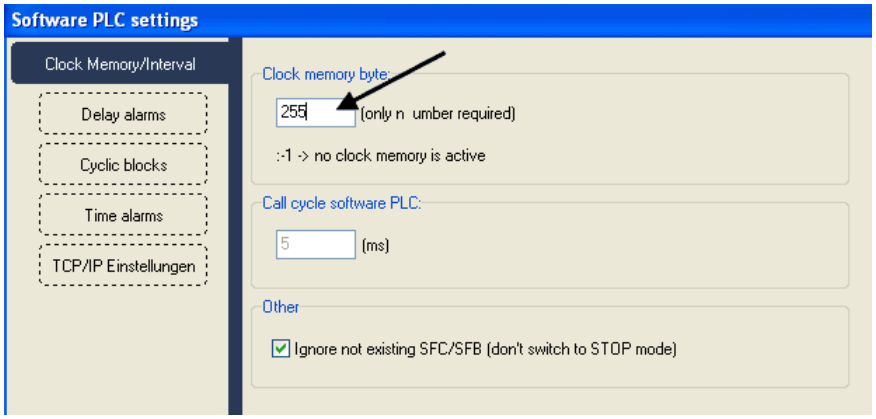


Fig.: Software-PLC settings

Confirm the dialog with the OK button.

2.2.6 Load blocks (OB1, FC1) into the PLC

Now we can simulate the PLC program. First we must load the blocks into the PLC. To do this, click the button "**Synchronize**":

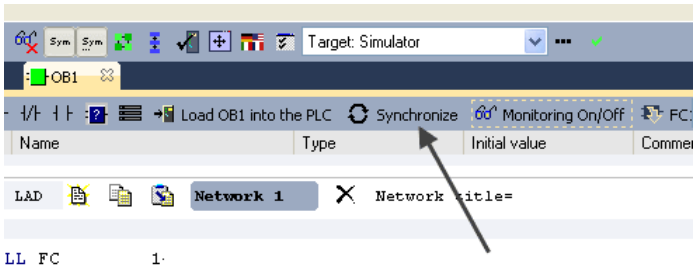


Fig.: button synchronize

When you have pressed this button you can verify the execution of the process in the message tab:

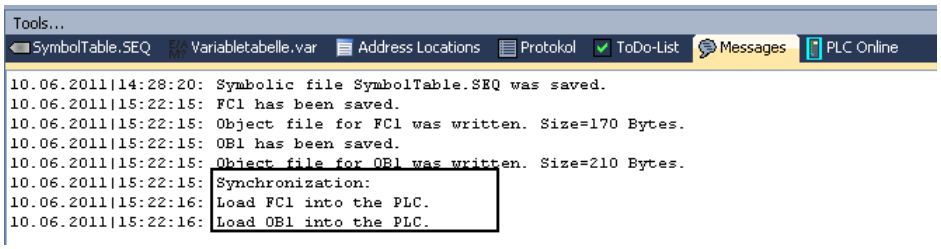


Fig.: Different messages in the message window

2.2.7 Switch PLC to RUN mode

Click on the tab sheet "PLC Control Center".
You can change the PLC to RUN mode with the "RUN" button:

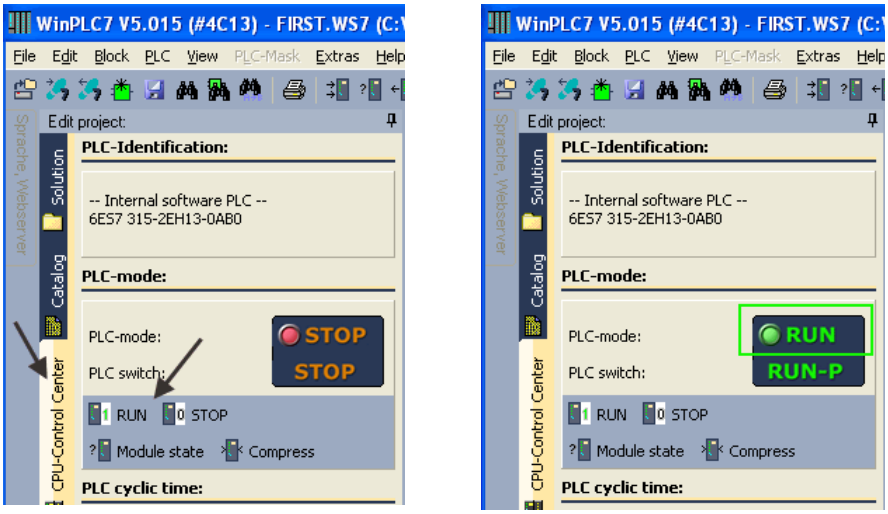


Fig.: Left side: PLC in STOP mode, right side PLC in RUN mode.

2.2.8 Block monitoring

To obtain a better view of the most important windows, maximize the editor window by clicking on these buttons:

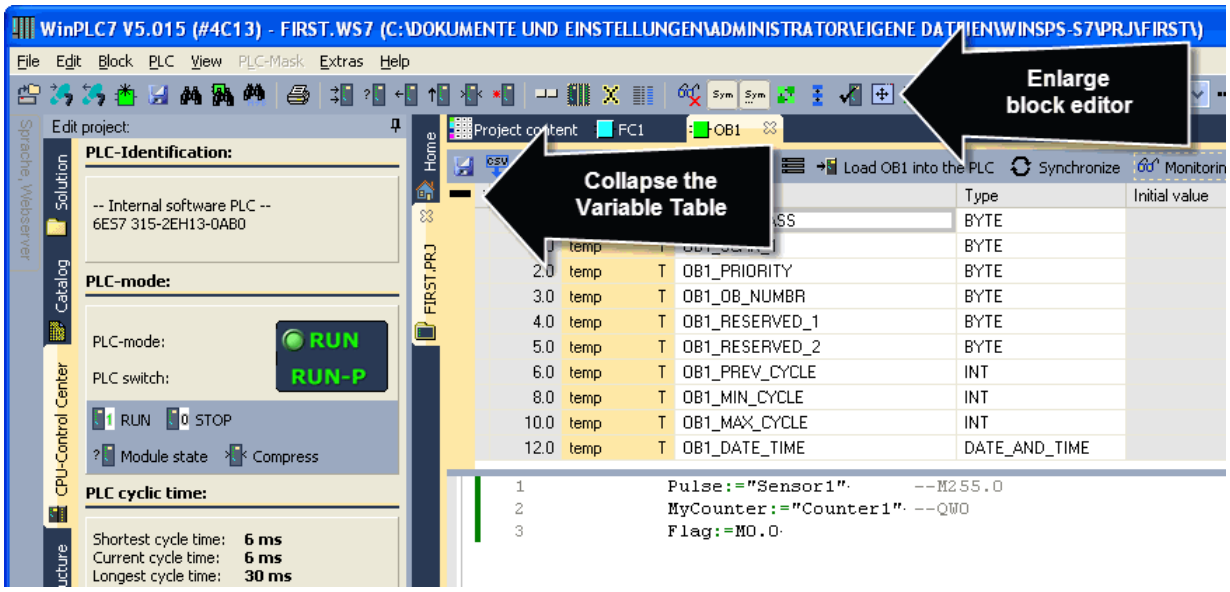


Fig.: Zoom in to the block editor

This provides more space for the block editor:

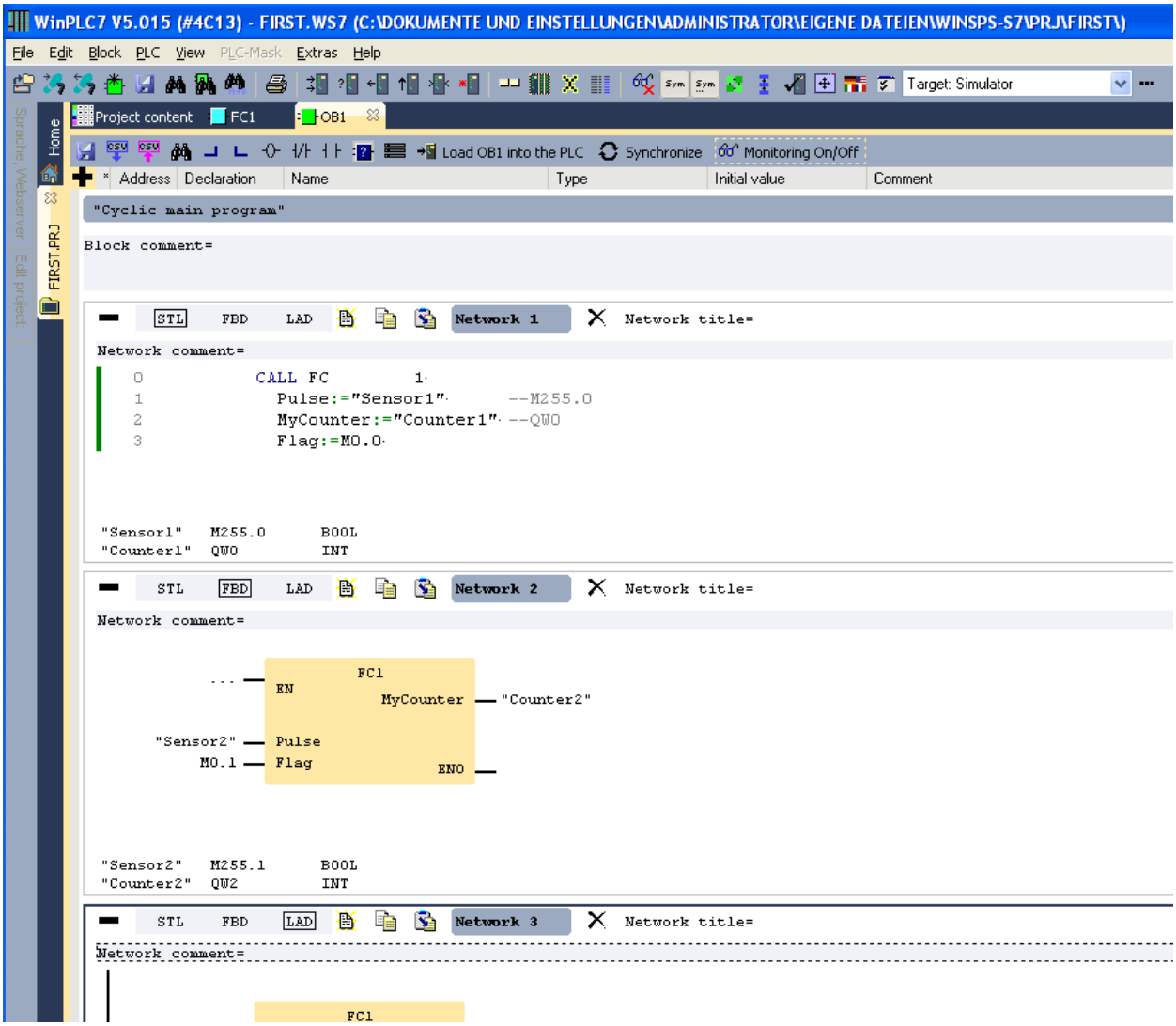


Fig.: Block OB1

At this point you can turn on block monitoring mode by clicking on the following icon:

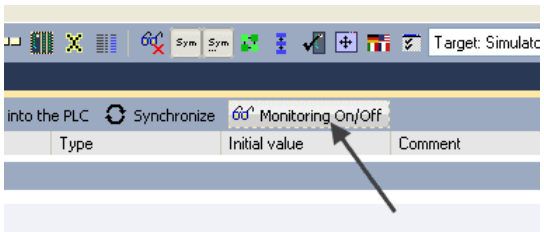


Fig.: Enable block monitoring mode

OB1 is displayed in monitoring mode:

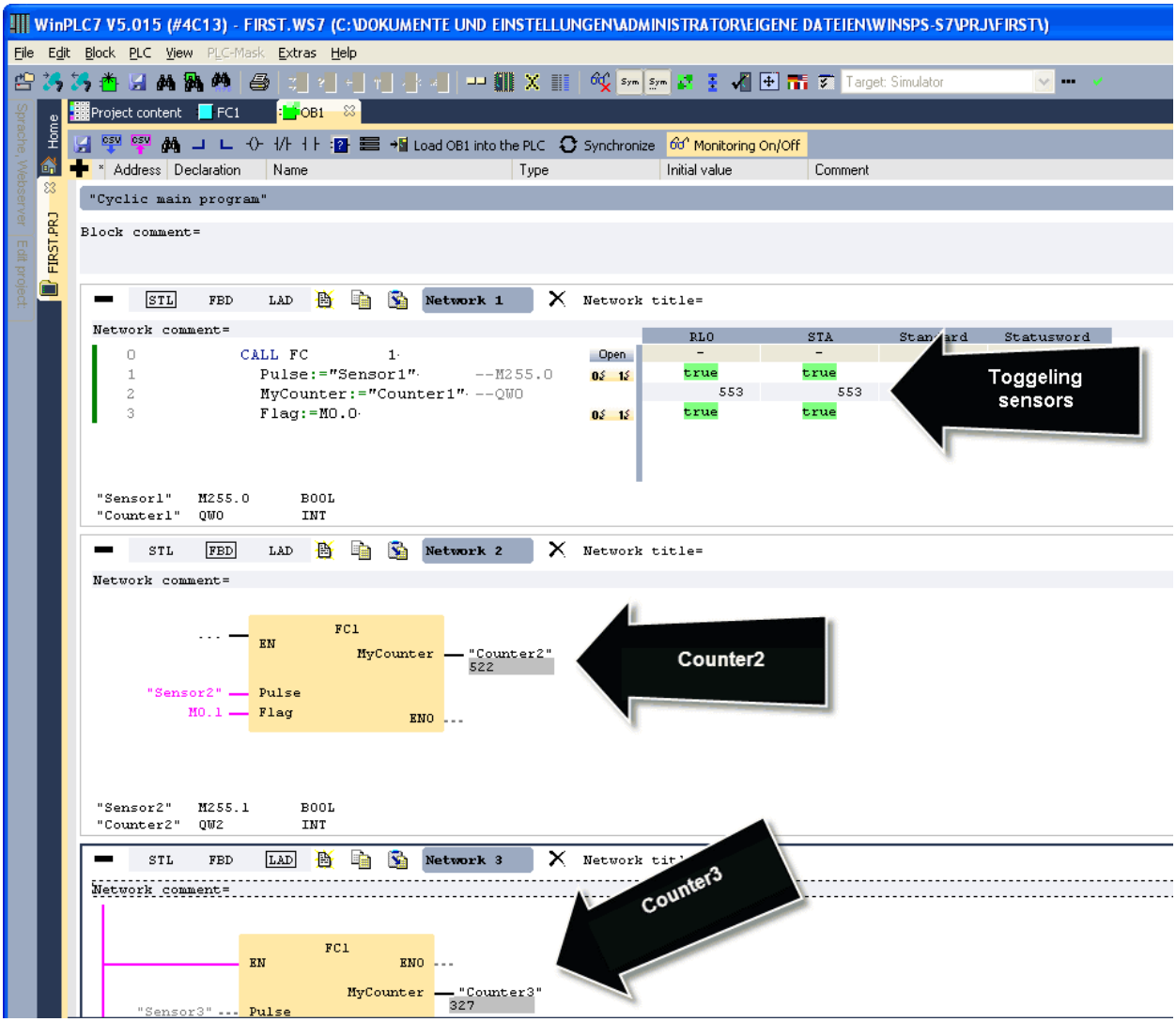


Fig.: OB1 in monitoring mode

Press the button "Monitoring ON/OFF" again to quit from the monitoring mode.

Press the  button again to zoom out of the block editor window.
In the next section we use the window "variable monitoring" window.

2.2.9 Variable monitoring

The variable monitoring window is available in a tab sheet of the tool window.
Now it currently blank:

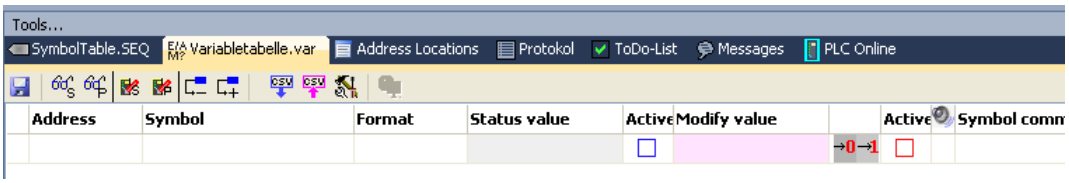


Fig.: Blank variable monitoring window

To add some variables, click on the column "Address" and enter "QW0".
Then press the "+" button twice to complete the table:

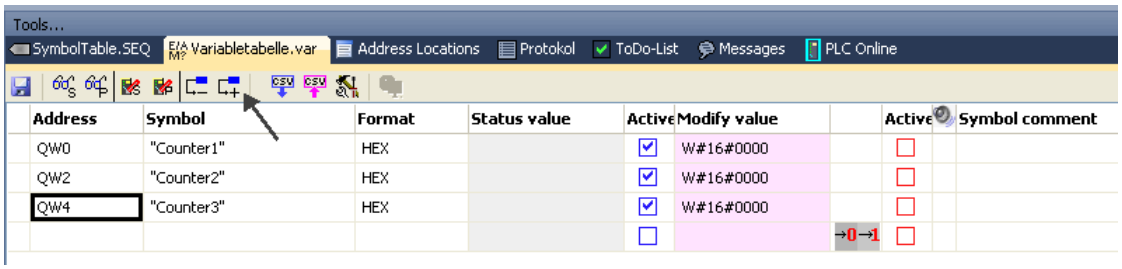


Fig.: window with 3 items

You can enable the monitoring mode with the button "Monitoring (permanently)". This displays the changing counter as follows:

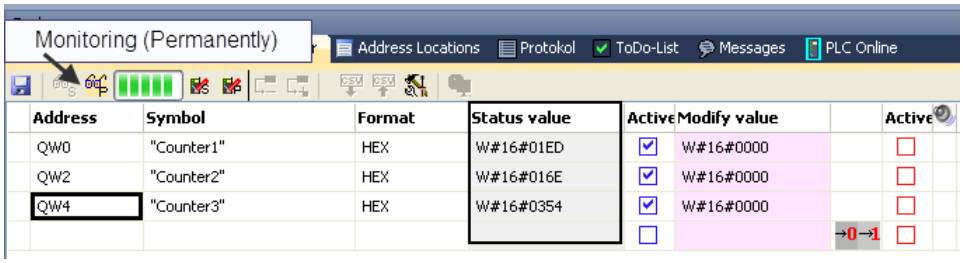


Fig.: Monitoring is ON

You can enable block monitoring and variable monitoring simultaneously.

Press the button "Monitoring (permanently)" again to close monitoring.

Hint:
If you have enabled monitoring mode in more than one window, you can close the monitoring mode in all windows by pressing the mouse button: 

2.2.10 Connect to a real S7-PLC

Until now we have only established a connection with the internal S7-Software-PLC. But now we will connect to a real S7-PLC.
First we must define the correct interface in the target settings:

Target: RS232	The PC is connected to the PLC via a serial MPI adapter or a USB MPI adapter (with a virtual COM-port)
Target: MHJ-Netlink	The PC is connected to the PLC via the MHJ Netlink adapter, Netlink-Lite or IBH-link.
Target TCP/IP Direct	The PC is connected to the PLC are connected via a standard Ethernet patch cable.
Target: MHJ-Netlink TS	The PC is connected to the PLC via the MHJ-Netlink adapter, Netlink-Lite or IBH-Link. Use this setting to establish the connection via the Internet.
Target: NETLink-PRO TCP/IP	The PC is connected to the PLC via the Netlink-PRO or the **NETLink-Pro Compact** adapter.
Target: SIMATIC-NET	The PC is connected to the PLC via the Simatic® Net-driver. With this setting you select the interface by means of the PG/PC interface dialog of the Simatic® manager or the Teleservice software V6.

To edit the settings of a specific target, press the "..." on the right of the combo box:

Fig.: Edit the target settings

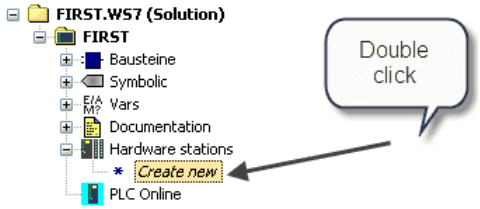
If you want to upload this example into a real PLC, you must enable the clock memory byte of the PLC using the hardware configurator. Otherwise the program in the PLC will not work. See also the next section.

Important note regarding the "Simatic-Net" target:
You can only use this target, if the SIMATIC-NET drivers are installed on the PC. For example, you want to use the SIEMENS interface "PC-Adapter USB" (972-0CB20-0XA00). You can only use this interface with the target setting "SIMATIC NET" if you have installed the driver for this interface.

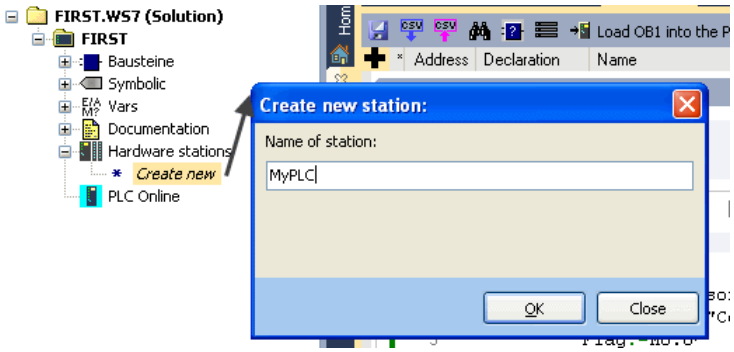
2.2.11 Create a hardware station (hardware configuration)

The following steps are required to create a hardware station and load the hardware settings into a PLC.
This example shows, how to change the "clock memory byte".

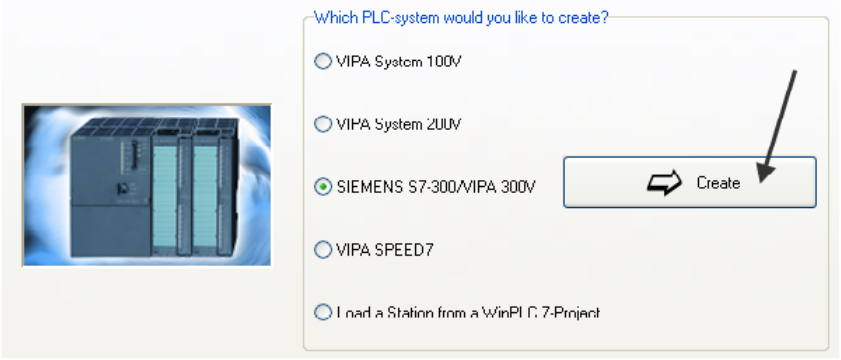
1. Create a new station in the node "Hardware stations" of the project tree



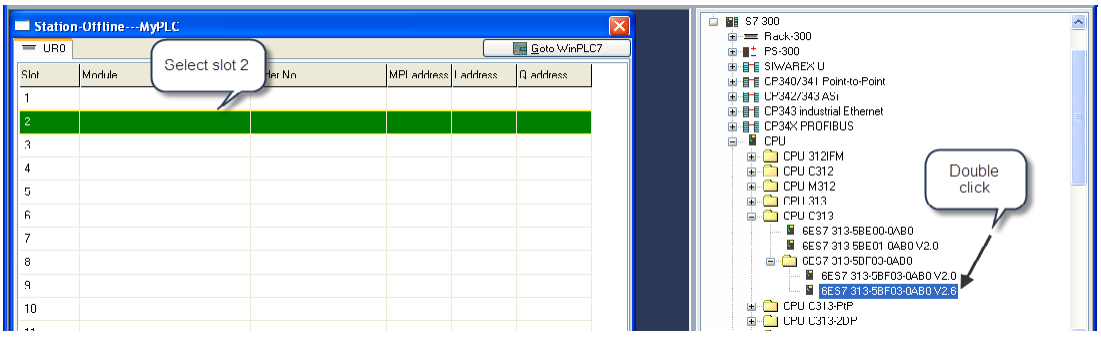
2. Define the name of the station



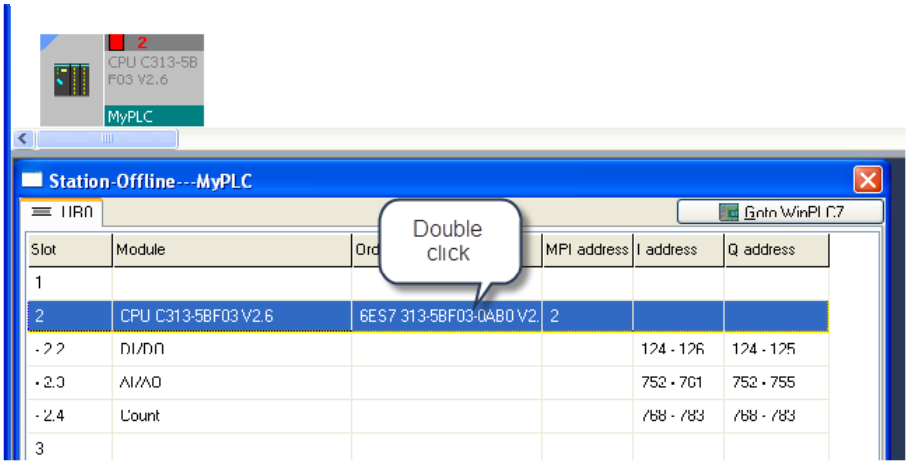
3. Select the PLC system



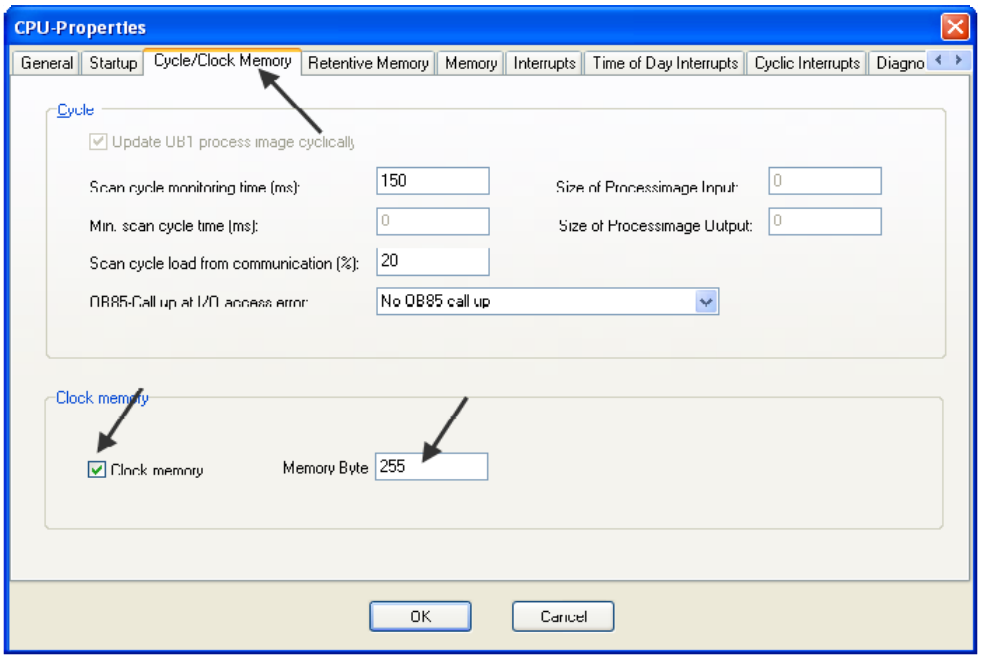
4. Insert the correct PLC into the hardware table



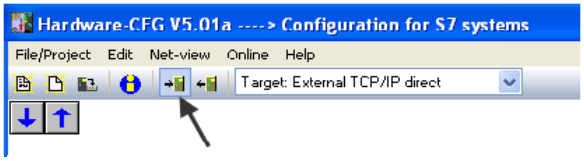
5. Edit the settings of the PLC

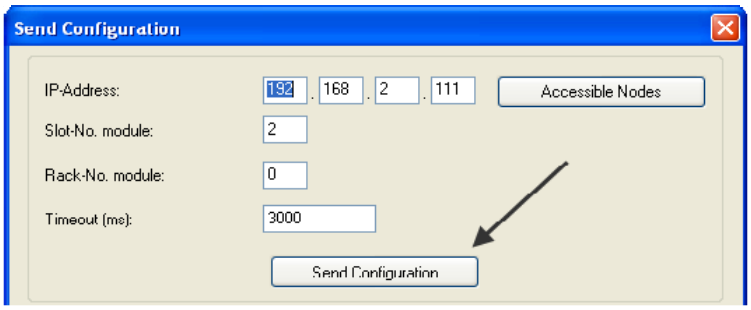


6. Set the clock memory byte to "255"



7. Load the hardware settings into the PLC





8. Close the hardware configurator

2.2.12 Summary

In this section you have read the following:

1. Creating a new solution
2. Adding symbols in the symbolic table
3. Creating blocks (FC1 and OB1)
4. Opening blocks
5. Synchronizing blocks with the PLC
6. Simulating the PLC program with the Software-PLC
7. Connecting to an external PLC
8. Creating a hardware configuration

2.3 PLC mask simulation

The PLC mask simulation is displayed when you have activated the target setting "Simulator". The PLC mask simulation displays a virtual S7-300 PLC. The PLC mask simulation is only provided for simulation purposes.

The PLC mask does not concern the hardware configuration, i.e. it can be used independently.

You may add the following modules to the virtual CPU module:

- 8-bit or 16-bit digital input module
- 8-bit or 16-bit digital output module
- Analog input module (1 channel each) with a slider
- Analog output module (1 channel each) as a bar graph
- BCD input module
- BCD output module

Open the PLC mask window using menu item View->PLC mask or with the following mouse button:

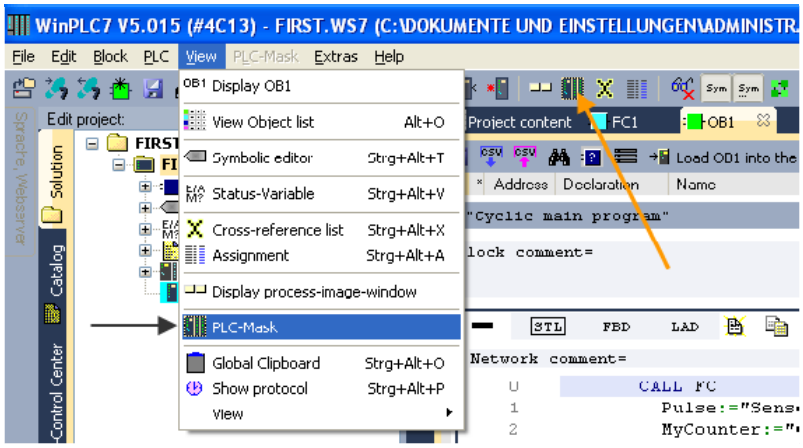


Fig.: Open the PLC mask window; only if the target is set to "Simulator"

The first time you open the PLC mask of your project it is displayed as follows::

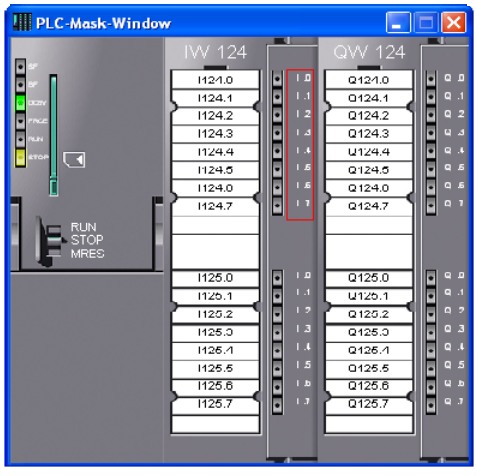


Fig.: PLC mask window, the default view

The PLC mask is initially equipped with one input module and one output module.

Here you can access the following configuration options:

- A double click on the address changes the module settings (address, labels, ...)
- If the bits do not have a label the symbols are displayed. If no symbols exists, the absolute address of the bit is displayed.
- You can use the modules using drag and drop. To start dragging, select the module address.
- You can delete a module by dropping it on the CPU module.
- You can add or delete modules by clicking the right mouse button.
- You can use the mouse to adjust the size of the window.

Digital input modules can be toggled in two ways:

1. By clicking to the LED with the mouse
2. By pressing the key '0'-'7'. Only the active input byte can be toggled via the keyboard. The active input byte is marked with a red rectangle. You can use the cursor keys (left and right) to move the red rectangle.

Example:

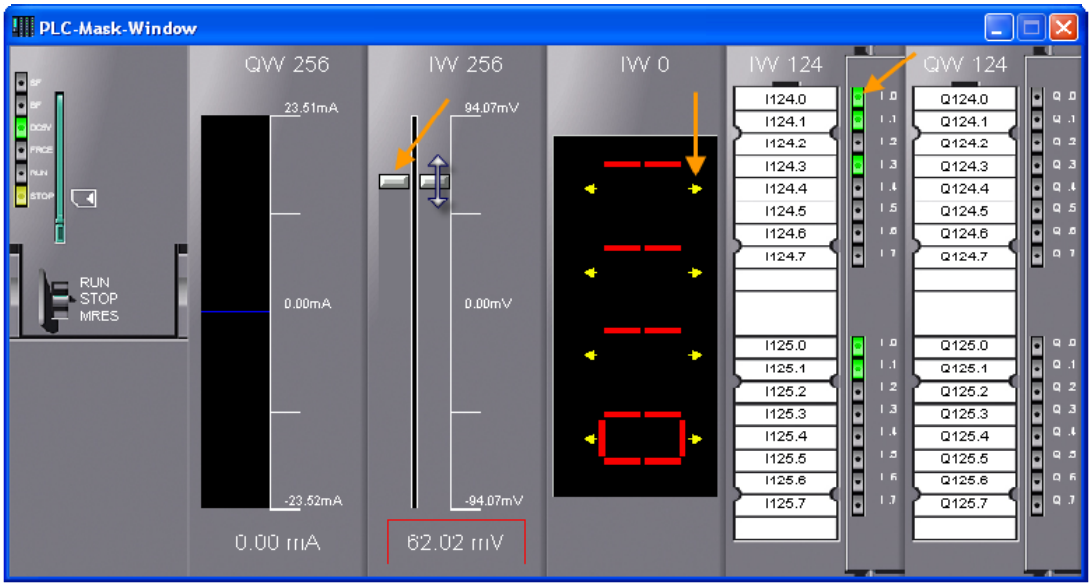


Fig.: PLC mask window with different modules

2.4 Status variable (monitor variable)

You can use the "Status variable" to monitor and control (modify) any number of addresses of the PLC.

Features of the status variable window:

1. Display the current value
2. Modify the value of an address
3. Voice output of the current value
4. Built-in Web server to monitor the current values on a smartphone or similar device.

The status variable table is displayed in the tool window in the tab sheet "Status variable":

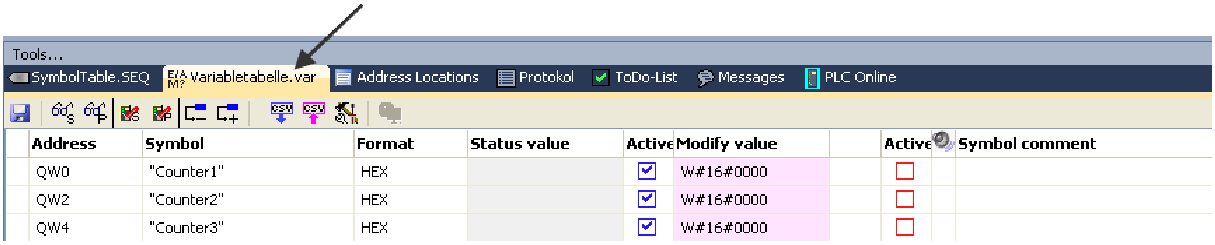


Fig.: Empty window "Status/Modify variable"

This window contains the following columns:

Address	The address that must be monitored or controlled.
Symbol	Displays the respective symbol of the address (if it exists).
Format	Select the format of the address
Status value	Indicates the current value of the PLC.
Active (status)	The blue tick specifies whether the actual value should be displayed or not.
Modify value	Enter the target value that will be saved to the PLC.
Toogle cell	Click on "1" to change the operand to "1". Click on "0" to change the operand to "0". Only available for binary operands.
Active (modify value)	A red tick specifies whether the target value should be saved or not.
Speaker	Click on this column and the speaker symbol will be visible. When the value changes, it is announced via voice output.
Symbol comment	Displays the symbol comment for the respective address (if it exists).

Explanation of the buttons:

	Save	Save the table
	Monitoring (permanent)	Permanent monitoring of all selected addresses
	Monitoring (single)	The actual value is read once from the PLC and displayed.
	Modify (permanent)	The set-point value is transferred repeatedly to the PLC.
	Modify (single)	The set-point value is transferred once to the PLC.
	Duplicate (+)	Duplicate the current entry and increment the address
	Duplicate (-)	Duplicate the current entry and decrement the address
	CSV import	You can import a table from a CSV file.
	CSV output	You can save the table to a CSV file. You can edit this file using Microsoft Excel.
	Settings	Displays the settings dialog.
	Web server	Enable and disable the web server. See section "Web server"

Press the mouse button "**Monitoring (permanent)**" to enable monitoring mode.

In monitoring mode, only the variables with a blue check will be monitored.

While monitoring, some of the functions of WinPLC7 are blocked. So don't forget to disable monitoring mode again.
To disable monitoring, click the mouse button "Monitoring (permanent)" again.

You can also press the following mouse button in the menu bar: 

This button disables all active monitoring windows.

Important editing operations:

Insert rows	Press the [Ins] key or the menu item with the right mouse key
Delete rows	Select two or more rows and press the [Del] key.
Copy a row into the clipboard	[CTRL] + [C]
Insert a row from the clipboard	[CTRL] + [V]

The "Project content window" shows all variable tables:

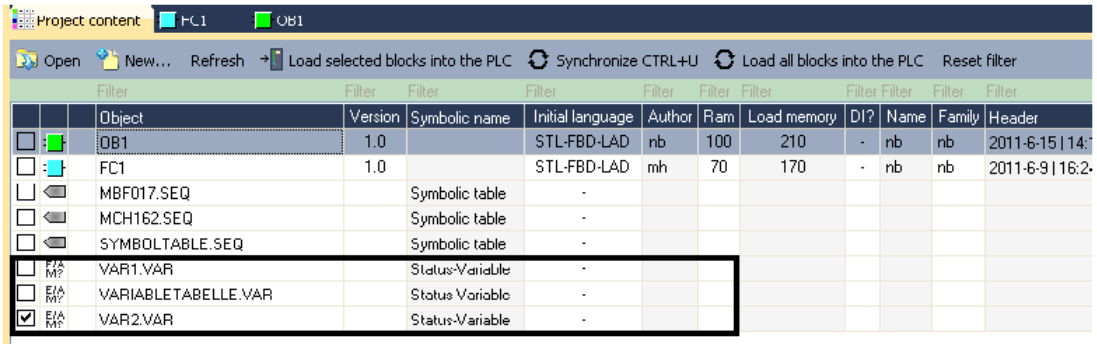


Fig.: Project content window with 3 variable tables

Click on the button "New.." to create a new variable and select the item "Status variable table" in the context menu:

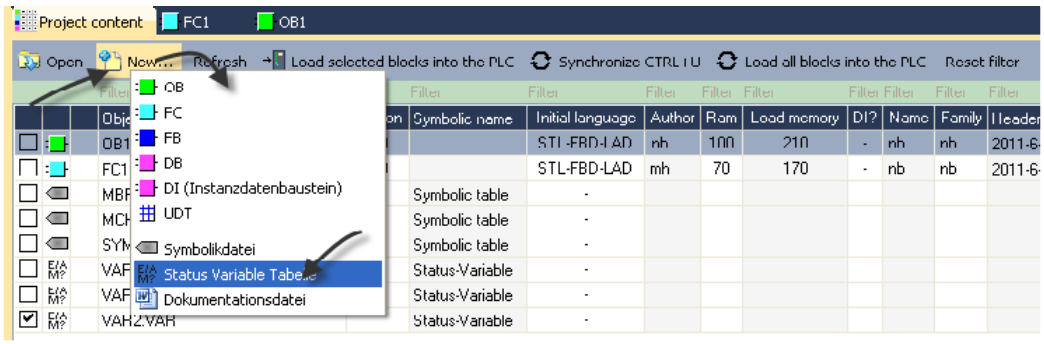


Fig.: Create a new status variable file

Hint:
You can filter the table if you enter ".var" in the edit field above the column labeled "Object". As a result, only the variable files will be visible in the table.

2.5 Web server in the status variable window

You can enable a web server in the status variable window. This feature enables you to view the current values of the variables on your smartphone. The smartphone and the PC are connected via WLAN. You can also use a second PC or a laptop that is connected via the local wired network.

This is helpful during commissioning. You can walk through your plant and actuate your sensor as required while you monitor the value returned on your smart phone to see whether the value of the sensor changes in the PLC.
This is available for binary and analog sensors.

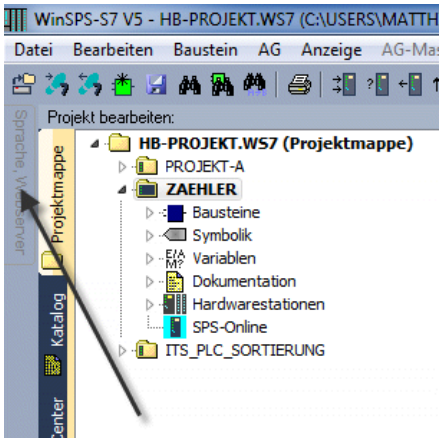
Procedure

1. Requirements: WLAN interface in the notebook or PC
2. First you must create an "AdHoc network". You can use the freeware "Virtual Router" or "Connectify" for this purpose.
3. Select the virtual network adapter that you created with the freeware software and start the web server.
4. Connect your smart phone to the WLAN network you have created.
5. When the connection is established, start your smartphone's browser and enter the correct IP address (IP address of the virtual network interface, see 3.)

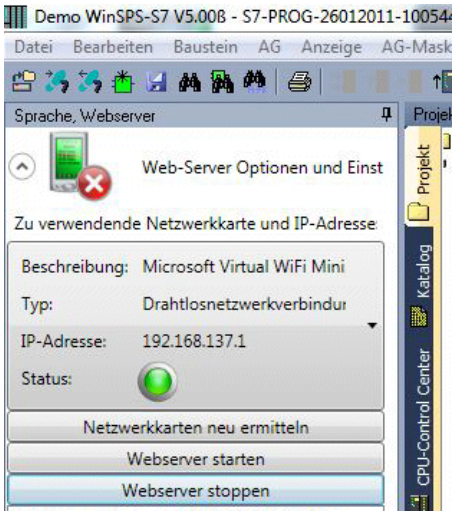
The detailed procedure follows below. In this case, the software "Virtual Router" is being used:

1. Download the software "Virtual Router" (Google search "Virtual router codeplex").
2. Install the software
3. Start the "Virtual Router"
4. Enter the "Network name SSID" and the password and press the button "Start virtual Router"

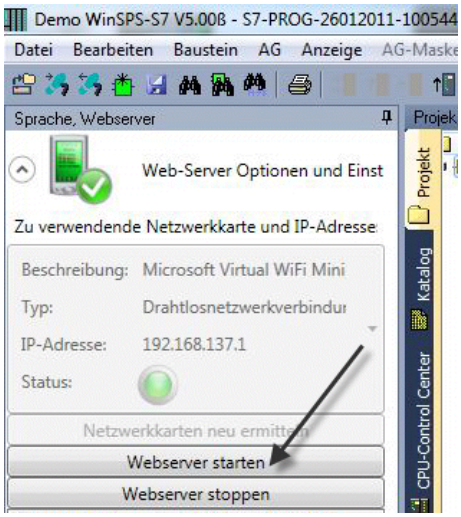
5. In WinPLC7, click on the tab sheet "Language, Web server"



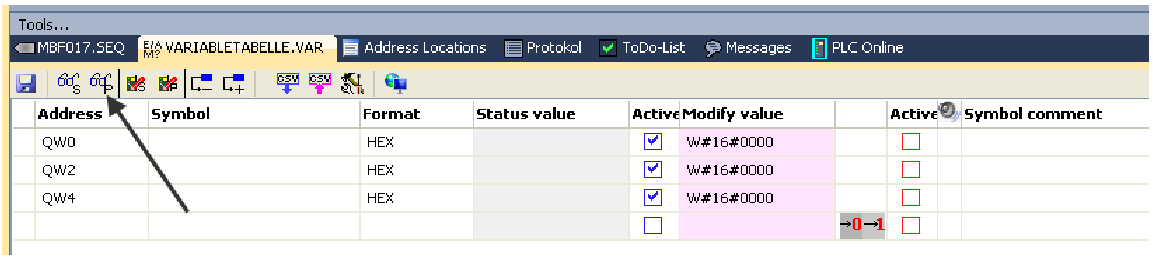
6. Select the network adapter "Microsoft Virtual WiFi Mini". A green LED should be visible.



7. Press the button "Start web server"



8. Establish a connection with the PLC and press the button "Monitoring (Permanent) in tab sheet "Status variable":



9. At this point everything is ready for monitoring with the smartphone. Connect your smartphone to the WLAN network (see 4.) you have created. Open the browser and enter the correct IP address. The status variable table will be visible on your smart phone:

2.6 Projects in WinPLC7 V5

WinPLC7 employs a solution and a project.
A solution contains one or more projects. Every project is an independent PLC program with a symbolic file, variable tables, etc.

The tab sheet "solution" displays the open solution:

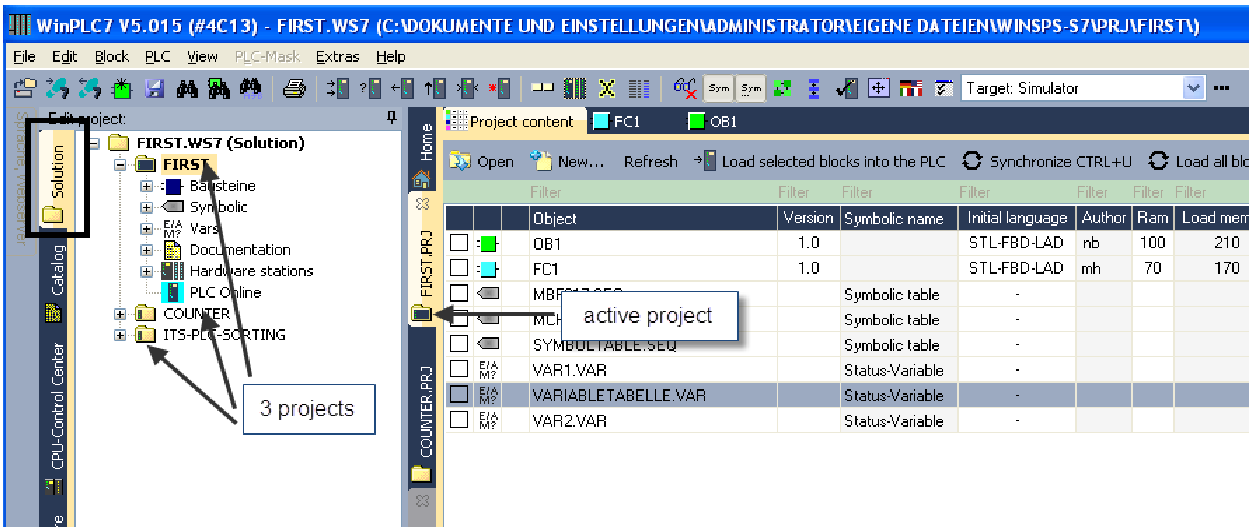


Fig.: A solution with 3 projects

The solution in the figure have 3 projects:

1. First
2. Counter
3. ITS-PLC-Sorting

The selected project is the active project. In the vertical tab sheet, the active project is the visible project.

One or more projects is open and visible when you open a solution. You can use the context menu **to open a second project** of the selected solution. Press the right mouse button to open the context menu:

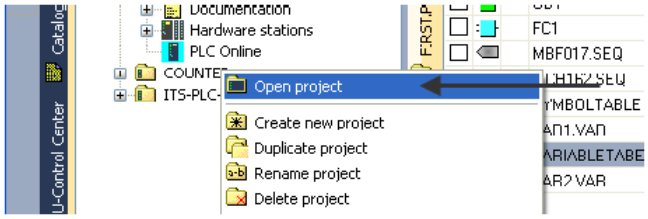


Fig.: open a project of the solution

You can also open a project by double clicking a block of the project:

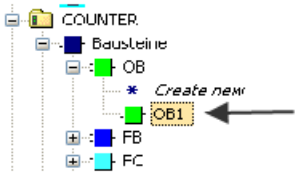


Fig.: When you double click a block, the respective project is opened

Open projects are displayed in the vertical tab:

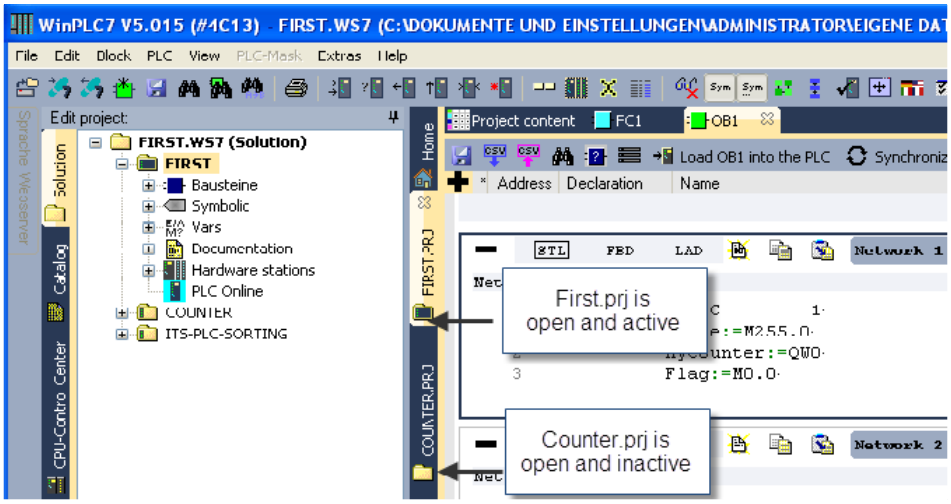
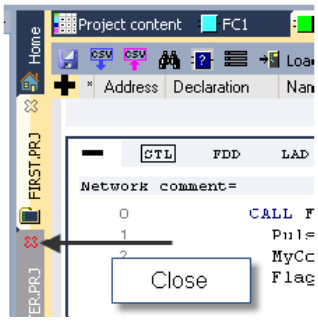


Fig.: When you double click to a block, the respective project is opened

You have the following options In the project tree:

- You can move objects (e.g. a block) by dragging and dropping them from one project to another project.
- Using the context menu, you can:
  - open a project
  - duplicate a project
  - rename a project
  - delete a project
  - move a project into another solution
  - import a project from another solution



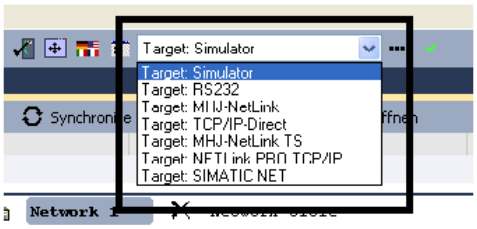
You can close a project with a click on the close icon

Components of a project:

Node	Description
Blocks	This node contains the blocks of the project.
Symbolic	This node contains the symbol files of the project. Only one symbol file can be active (visible).
Vars	This node lists the variable tables of the project. Only one table can be visible at any time.
Documentation	This node lists the documentation files are listed. You can copy these into the project using "drag and drop".
Hardware stations	Here you will find all the hardware stations of the project. You can edit a hardware station with a double click.
PLC-Online	This icon displays the tab "PLC-Online". This tab displays all the blocks of the connected PLC (or Software-PLC).

2.7 Target: External or internal simulator

WinPLC7 is provided with an integrated simulator that enables you to test PLC programs without a hardware PLC.
The selection that defines whether you will work with the simulator or with an external S7 PLC is defined in target:



If you wish to work with the simulator, select **"Target: Simulator"**.
All the other parameters apply to an external S7 PLC:

Target: RS232	PC and PLC are connected via MPI adapter serial or MPI adapter USB (with a virtual COM-port)
Target: MHJ-Netlink	PC and PLC are connected via the MHJ Netlink adapter, Netlink-Lite or IBH-link.
Target TCP/IP Direct	PC and PLC are connected via a standard Ethernet patch cable.
Target: MHJ-Netlink TS	PC and PLC are connected via the MHJ-Netlink adapter, Netlink-Lite, IBH-Link. Use this setting to establish the connection via the Internet.
Target: NETLink-PRO TCP/IP	PC and PLC are connected via the Netlink-PRO, **NETLink-Pro Compact** adapter.
Target: SIMATIC-NET	PC and PLC are connected via the Simatic® Net-driver. With this setting, you select the interface by means of the PG/PC interface dialog of the Simatic® manager or the Teleservice software V6.

2.8 Importing and exporting a STEP®7 project

Import a SIEMENS STEP®7 project as follows

On the HOME screen you can import a STEP7 project as a ZIP file or as an unpacked project:

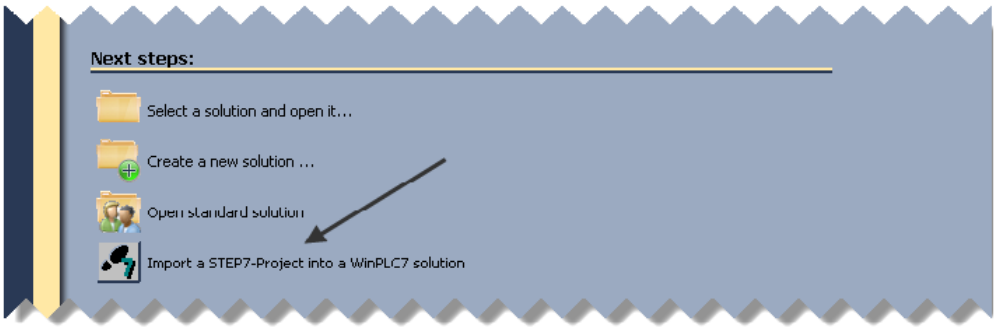


Fig.: The import button in the home screen

1. Select the ZIP file or the S7P-file (SIEMENS project file)
2. Define the new project name and the target folder
3. Select the projects that you wish to import
4. Import the selected projects. The result appear in the project tree of WinPLC7.

You can also import a STEP7 project, when a WinPLC7 solution is open. In this case, select the menu item **File->Import->STEP7-Project...**

Export a SIEMENS STEP®7 project as follows

First, open the WinPLC7 solution that you want to export. You can only export an active project.
Select menu item File->Export->Export S7P.
The export dialog will be displayed:

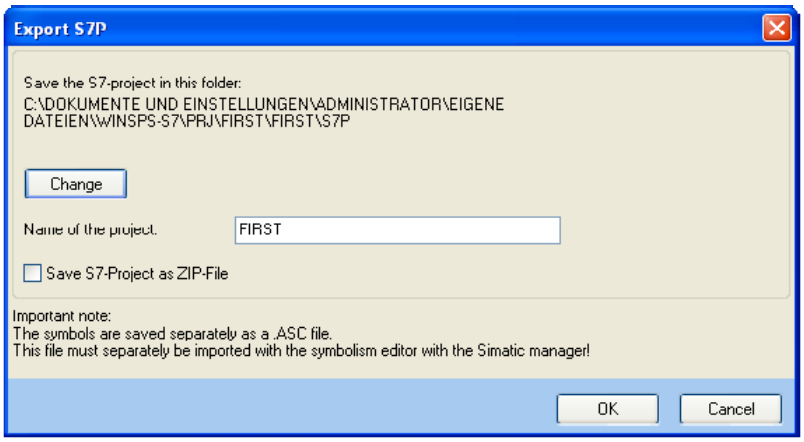


Fig.: The export dialog

Use the "change" button to select the target folder. If you want to create a ZIP file, mark the corresponding check box.

The exported project includes:

- blocks
- symbol file
- comments

For more information, read the text file "notes_hdw_export.txt". This file is generated in the root directory of the project being created.

2.9 Sending a project by e-mail

Because a single project consists of many different files, you should save the project to a ZIP file. It is then a simple matter to send the complete project via e-mail.

If the recipient **works with the SIEMENS STEP®7 software**, you should generate a ZIP file of the Siemens project. The previous paragraph describes how you can create such a ZIP file.

If the recipient uses WinPLC7 as well, you should choose menu item ***File->Save project as ZIP file*** and select format **WinPLC7 V2**.
The advantage is that you can pack multiple projects into the ZIP file.

2.10 Simulate your S7 Program

You can turn the WinPLC7 simulator on by means of the target specification:

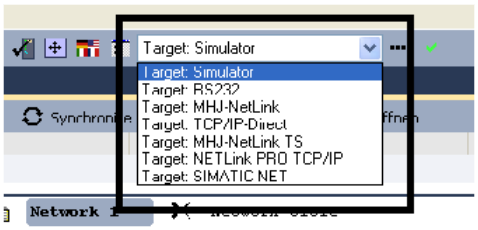
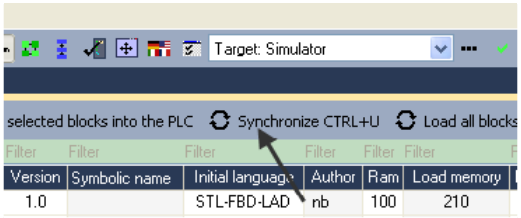


Fig.: Simulator is turned on.

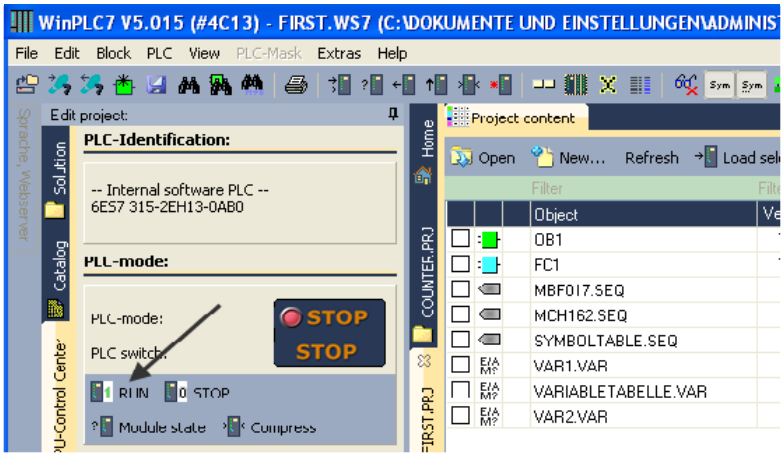
If the simulator is on, all the commands in menu item "PLC" refer to the simulator (the Software PLC).

Since the simulator behaves like an external S7 PLC, you must complete the following steps to simulate a PLC program:

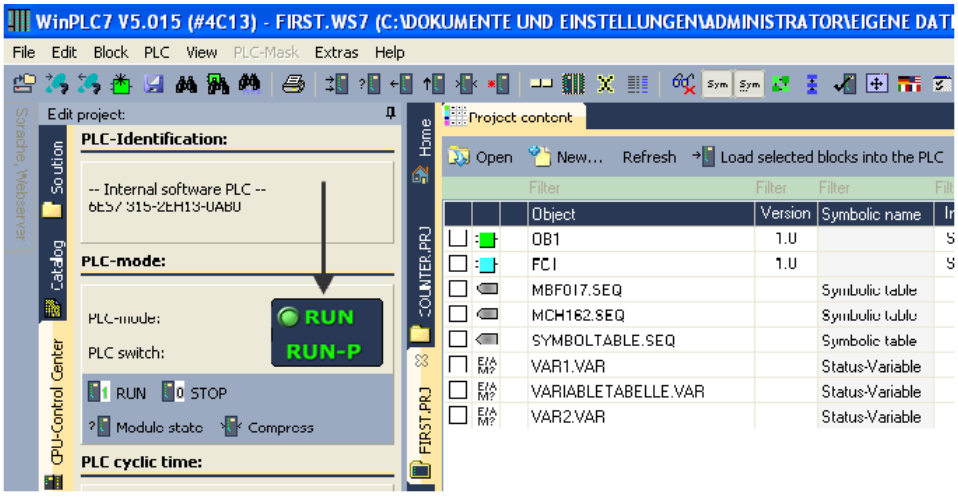
1. Create a PLC program
2. Transfer the PLC program (e.g. by means of menu item *PLC->Synchronize*)



3. Switch the simulator to RUN mode (e.g. by means of the "CPU Control Center")

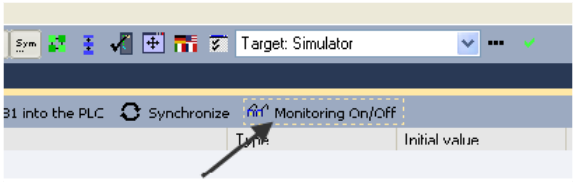


4. If there are no errors in the PLC program, the simulator should be in **RUN** mode:

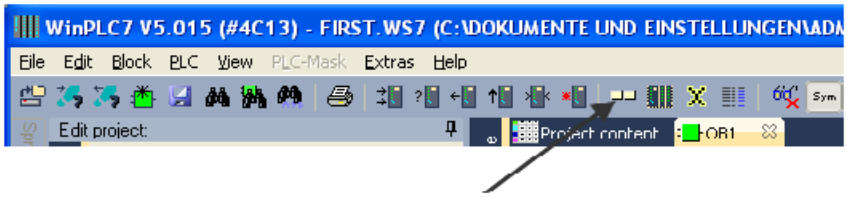


In simulator mode you have many different options to monitor and to control the PLC program, i.e.:

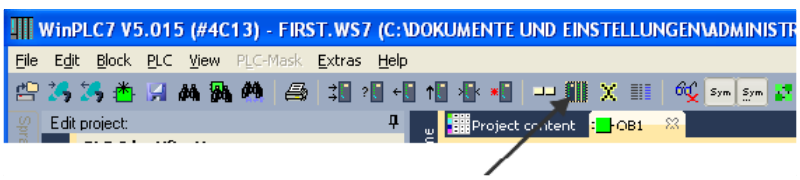
1. You can open a block to **monitor** the program (menu item *Block->Monitoring On/Off* or you can click on the icon with the glasses)



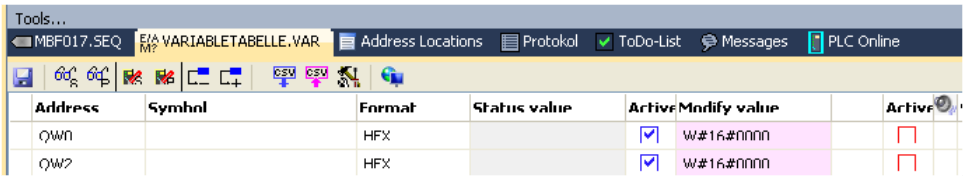
2. Display the **PIO windows**. These are small fields that you can use to monitor inputs, outputs, bit memories, timers, counters, etc. You can control the inputs with the mouse or the keyboard (numbers 0-7).
You can turn the PIO windows on (and off again) via menu item *View->Process image window*.



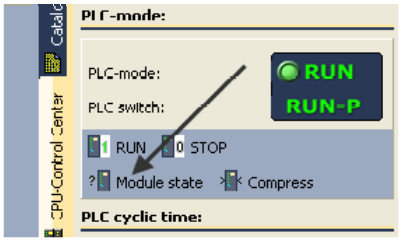
3. The **PLC Mask Simulation** displays a S7-300®-CPU. Here you can insert modules and you can modify the addresses of the modules.
You can also display analog inputs and analog outputs.



4. The **status variable window** enables you to monitor and control addresses.



5. If an error occurs that causes the simulator to change to stop mode, you can determine the reason by means of the **module state**. The "Diagnostic" tab displays the most recent diagnostic messages. You can display the module state via menu item *PLC->Module State* or with the button "Modul state" in the CPU control center.



The memory size (RAM) of the simulator is **250 KB**.

The addressing ranges are:

			Range	
Process image (input)	16384	(Bit)	I0.0	I2047.7
Process image (output)	16384	(Bit)	Q0.0	Q2047.7
Bit memory	32768	(Bit)	M0.0	M4095.7
Timers	512		T0	T511
Counters	512		C0	C511
Local data	1024	(Byte)		

The chapter on the Software-PLC of WinPLC7 contains all the relevant technical data.

2.11 Connecting to a S7 PLC

In order to connect to an external S7 PLC, the target must be set to External.
The type of MPI interface depends on the target:

MPI-Interface	**Target definition**
Serial MPI-adapter	Target: RS232
USB MPI-adapter with a virtual COM-port	Target: RS232
GreenCable supplied by VIPA, only suitable for VIPA-CPUs.	Target: RS232 The baud rate must be set to 38400.
MHJ-Netlink, Netlink-Lite, IBH-Link	Target: MHJ-Netlink
NETLink-PRO TCP/IP **NETLink-PRO Compact TCP/IP**	Target: NETLink-PRO TCP/IP
Siemens MPI-Adapter 5.1 serial 6ES7972-0CA23-0XA0	Target: RS232
Siemens MPI adapter USB (*) 6ES7972-0CB20-0XA0	Target: SIMATIC-NET
Siemens TS adapter 5.2 serial 6ES7972-0CA34-0XA0	Target: SIMATIC-NET
Siemens Notebook adapter CP5512 6GK1551-2AA0	Target: SIMATIC-NET
Siemens PCI adapter CP5611 6GK1561-1AA01	Target: SIMATIC-NET
Siemens Teleservice adapter II 6ES7972-0CB35-0XA0	Target: SIMATIC-NET
Ethernet patch cable	Target: TCP/IP direct

(*) "RS232" is not a valid target since this USB adapter does not provide a virtual COM port.

When the target is "SIMATIC-NET", the "PG/PC interface dialog" of SIEMENS must be available on the PC.
The respective start icon is located in the Windows control panel.
This dialog exists when you have installed a SIEMENS interface driver, SIEMENS STEP®7 or the SIEMENS Teleservice software.

You can define the properties of the target by means of the [...] button located to the right of the target selection box:

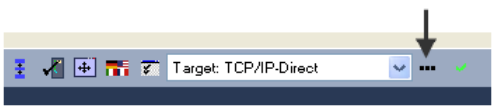


Fig.: [...] button for the targets

Hints about the "RS232" target:

A successful connection depends on the COM port number and the baud rate:

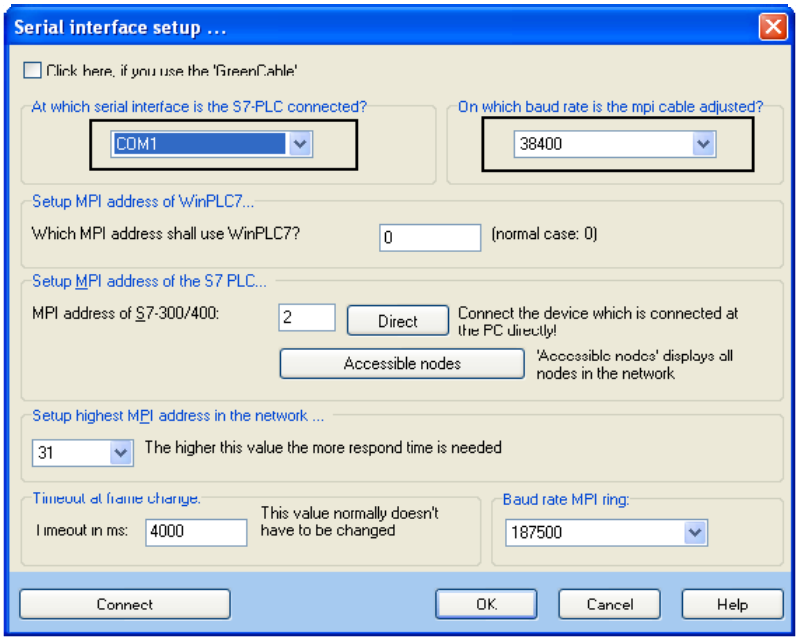


Fig.: Settings for the RS232 target

The SIEMENS serial adapter only supports baud rates of 19200 and 38400. These baud rates are selected by means of a micro switch on the interface.
If you select the wrong baud rate a connection cannot be established.

The serial adapters supplied by MHJ-Software and other vendors support all the available baud rates.

USB adapters that support a virtual COM port usually provide a COM port no. that is >=4. The correct COM port number is displayed by the Windows device manager (Control Panel->System->Device manager).

Hints regarding the targets "MHJ-Netlink, NETLink-PRO, Netlink-Pro Compact and TCP/IP direct":
It is important, that the PC has access to the IP address if the PLC must be controlled via an IP address.
I.e. the IP address of the PC must be located in the same subnet as the IP address of the PLC or the adapter.

You can check whether the PLC is accessible via the specified IP address using the **ping command**.
At the Windows command prompt, type:

ping xxx (replace xxx with the IP address)

If you do not receive a reply, then the IP address is bad, or the IP address is not accessible from the PC.
Solution: change the IP address of the interface or the IP address of the PC.

2.12 Modifying the program in a S7 PLC

You have **two options** to modify a block in the PLC:

1. If you have saved the PLC program in a project on the PC: open the block in the project, modify the block and transfer it back to the PLC.
2. Open the online window of the PLC, then open the block with a double-click. Modify the block and save it to the PLC.

Of these, the first option is **preferred**:
You have saved the PLCs program in a project.
When you have modified the block and transferred it back to the PLC, the blocks in the PLC and the project are identical.

Since **text/comments are not saved** to the PLC, **no** comments or original variable names are available.
When you use this approach, the block is only modified in the PLC and not in the project.

If you don't have the original project, start by creating a new solution. Continue to load all the blocks of the PLC into the new project. Duplicate the project in the project tree. Then work with the copied project. Now you can make changes and comment these. In this manner you can progressively create a PLC program that includes comments.

Tab "PLC-Online" in the tool window:
Press the "Get block list" button to display the blocks of the connected PLC. Here you can select all the blocks and copy them into the current project (button "Copy into project").

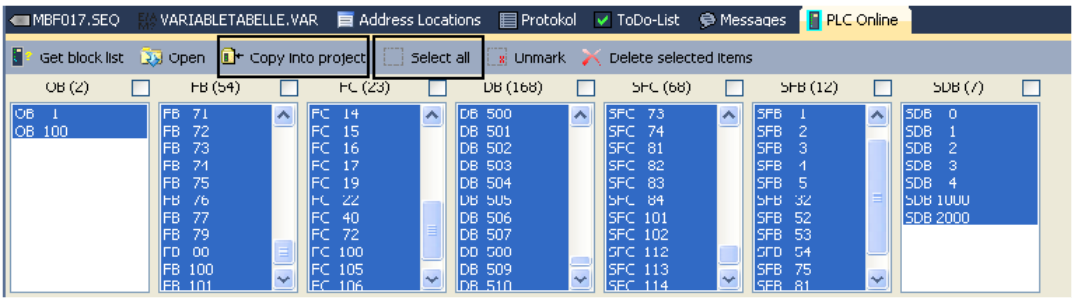
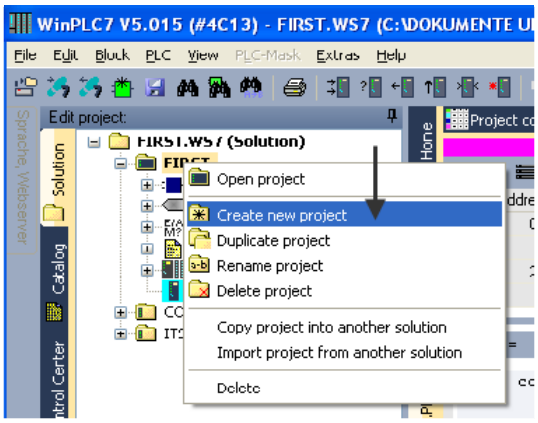


Fig.: "PLC Online tab"

You can also open the PLC block directly with a double click.
You should, however, avoid this because of the reasons mentioned above.

2.13 Creating a new S7 program

You can either create a new PLC program in a new project (in the current solution) or in a new solution.
If the new PLC program belongs to the current solution, create a new project in the open solution:



Click on the project tree with the right mouse key.
The context-sensitive menu will be displayed. Select the entry: *Create new project*.

To create a completely new solution, close the current solution and press the button "**Create a new solution**" in the HOME screen:

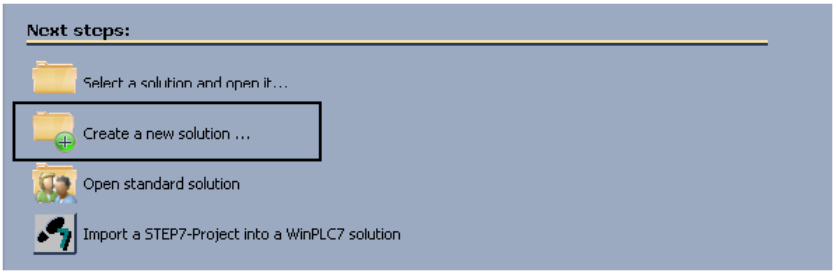
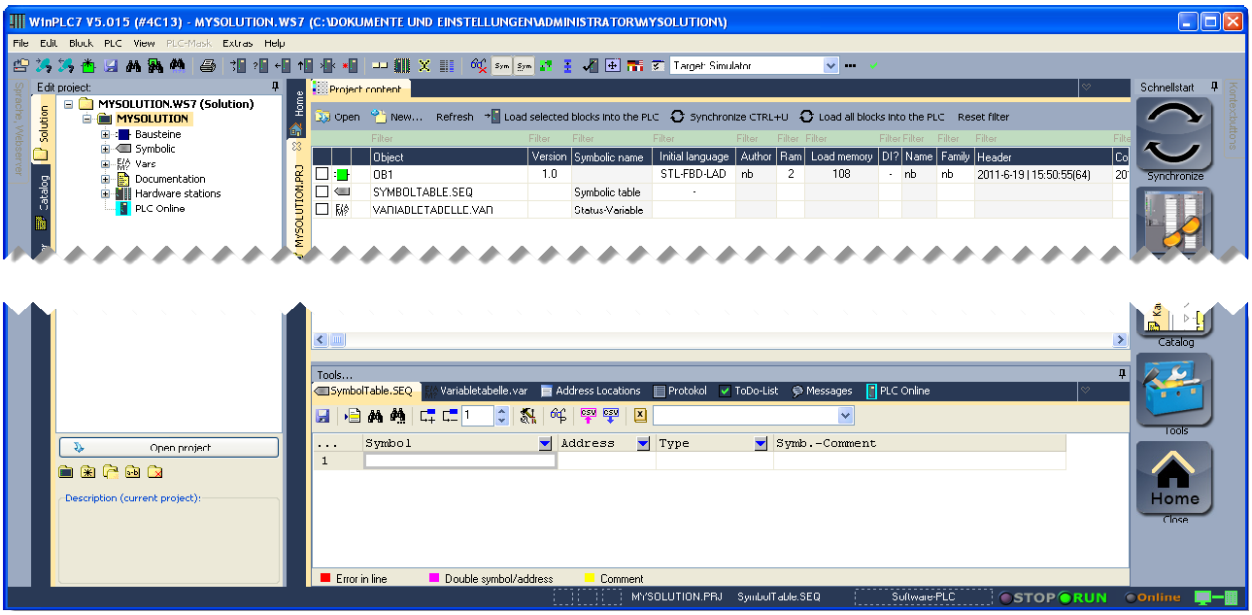


Fig.: Create a new solution

Now you can define the name and the location of the new solution.

The new solution contains an empty block "OB1", a symbol table and a variable table:



The solution already has one project with the same name as the solution. You can rename this project with a right mouse click in the project tree.

You might continue with the following steps:

- select the target system (Simulator or a real PLC) in the combo box under the menu bar.
- open the block OB1 and edit it.
- select the symbolic table tab and edit it.
- create a new hardware station in the project tree and edit it
- make a backup of the connected PLC (see tab "PLC online" in the tool window)

3 Project content window

The project content window displays all the objects of a single project in the solution. These can be:

- blocks
- symbol tables
- variable tables
- documentation files (WORD, EXCEL, images, ...)

You can use the project content window to open, delete, rename and create new objects.

3.1 Open the project content window

The project content window is displayed in the first tab of the editor area:

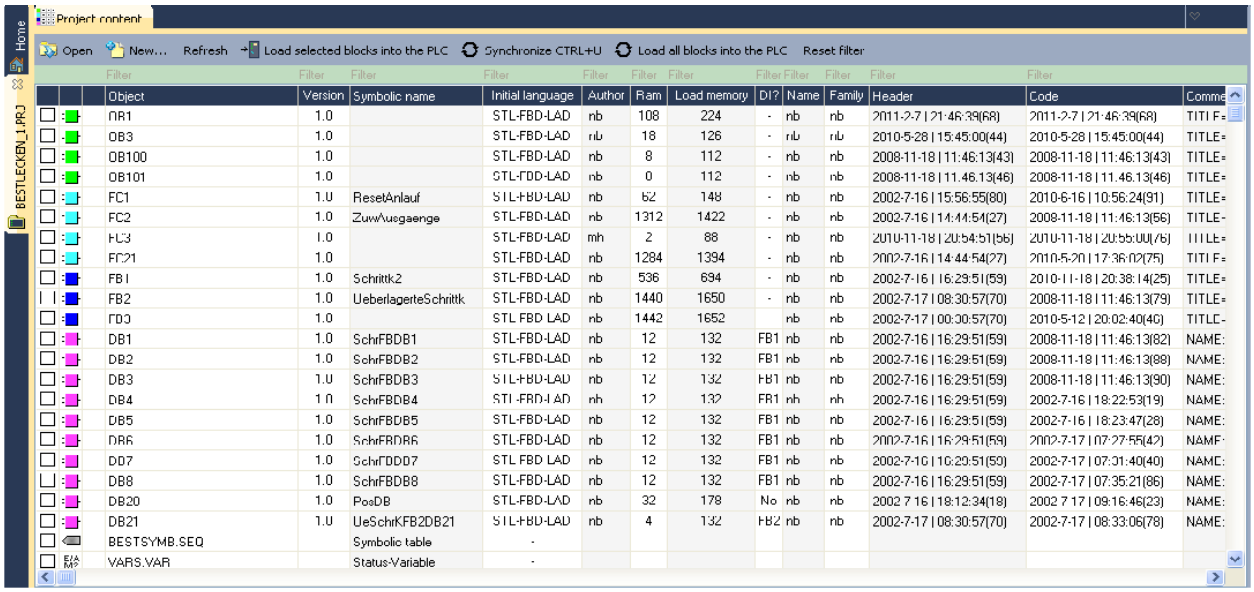


Fig.: Project content window

A filter is available at the top of every column. For instance, if you enter the text "FC" into the filter "Object", only the FC blocks will be visible in the table. If you add the filter "Author", only FCs will be displayed that were created by a specific person.

You can create a new object using the button "New...":

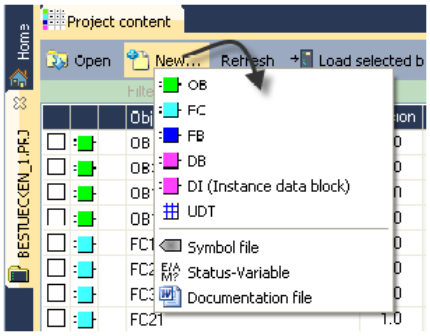


Fig.: Create a new object

For more options, open the context menu with the right mouse button:

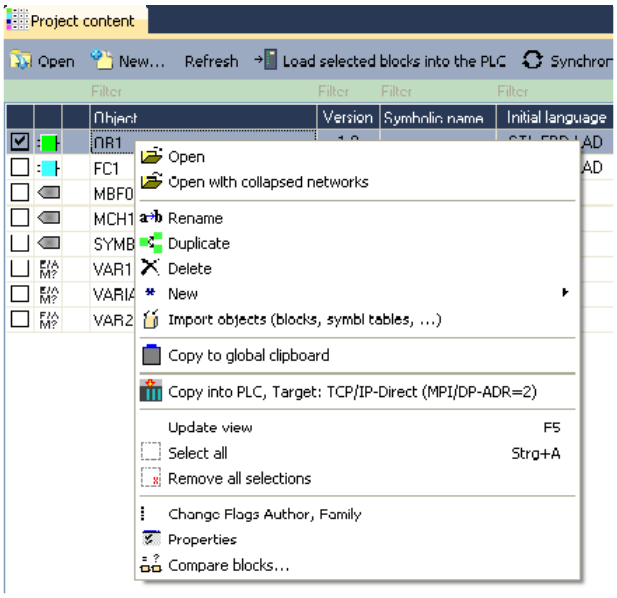


Fig.: The context menu of the "Project content window"

3.2 Create a documentation file

You can create a new documentation file using the context menu.

This can be:

- a WORD document
- an EXCEL document
- an ASCII file
- etc.

Choose menu item New->Documentation file in the context menu:

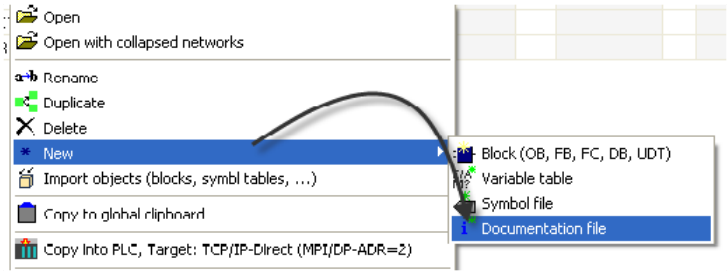


Fig.: The context menu of the project content window

Then enter the filename with the correct extension:

Document	File name extension
WORD document	.DOC or .DOCX
EXCEL document	.XLS or .XLSX
ASCII file	.TXT

The file extension is optional. In the table, a suitable icon identifies the file extensions. Unidentified file extensions are identified by a standard icon.

A double-click on the icon starts the Windows application that is linked to this file extension is started.

3.3 Add a documentation file using "drag and drop"

You can also add a documentation file using "drag and drop". Simply drop the file on the project content window. The dropped file will be copied into the project folder.

If you zip the project (WinPLC7-format) the documentation files are also included in the resulting ZIP file.

Note:
If you export the project to a SIEMENS project, the documentation files are not exported, because the SIEMENS project does not support these files.

4 Symbolic programming

4.1 Absolute programming

Absolute programming makes use of absolute addresses (e.g. M50.0).
This means that operands and addresses are immediately recognizable in the PLC program. However, the disadvantage is that the function behind an address is not visible:

Absolute programming:

```
A   I 124.0
A   I 124.1
A   I 124.2
=   M 10.0
```

With appropriate comments, this code becomes a lot clearer:

```
A   E 124.0   //door closed
A   E 124.1   //emergency-off sequence ok
A   E 124.2   //safety light barrier
=   M 10.0    //safety chain exists
```

Advantages of absolute programming:
The operand and the address of the operation are immediately visible.
For instance, if you want to check the wiring, you can instantaneously compare the level at the input module.

Disadvantages of absolute addressing:
Readability suffers, since the relationship to the process is not directly visible.
This requires a manual comment.

For this reason, it is common practice to create an symbolic file to allow the person viewing the code to choose whether the symbols should be displayed or not.

The following paragraph shows the same statements as a symbolic program.

4.2 Symbolic programming

Symbolic programming:

```
A   "door closed"
A   "emergency-off chain"
A   "safety light barrier"
=   "safety chain"
```

These STL lines are easily readable without any comments and the absolute address can be displayed to the right of the symbol.

The symbol assignment is defined in the symbolic editor where every absolute address is associated with a symbol.

You see the symbolic editor in a tab-sheet of the tool-window:

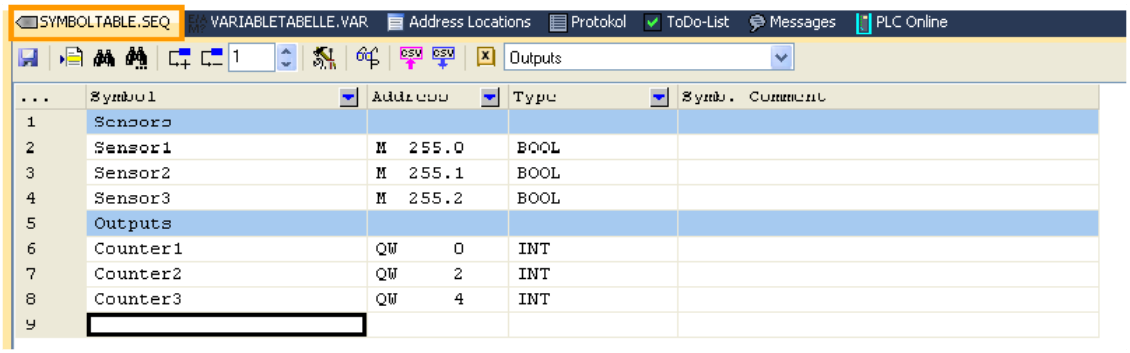


Fig.: Symbolic editor

When you enter addresses and symbols, you should start in the address column. When you have specified a row, you can continue to add addresses quickly by means of the mouse key [..+]:

A click with the right mouse key displays the context-sensitive menu. This menu contains many different commands for the symbolic editor:

- insert row
- delete row
- search functions
- cross reference list of a symbols/addresses
- etc.

The most important keyboard commands are:

Key	**Description**
Ins	Insert row
Del	Delete the selected row (At least two columns must be selected.)
CTRL+C	Copy the selected row into the clip board
CTRL+V	Insert a row from the clip board
F2 or RETURN	Edit the cell
CTRL+S	Save

The cells of addresses or symbols that were **duplicated** by mistake are **highlighted in color**.
This means that you can see any possible error before you save the symbolic file.

The resulting STL network could be displayed as follows:

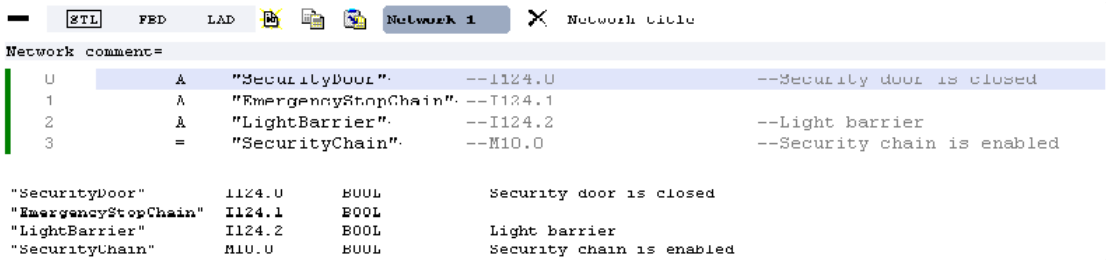


Fig.: An STL network with symbols

The absolute addresses are displayed to the right of the STL. All the symbols that were used in the network are summarized below the network.

You can use the mouse button to determine whether the summary of the symbols is displayed below each network:

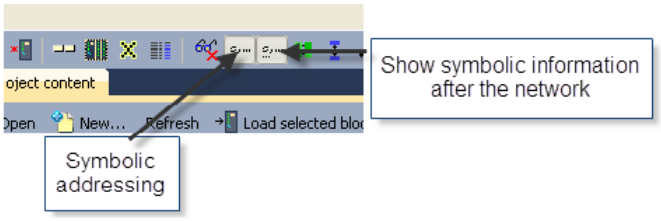


Fig.: Icon "View with symbolic information"

It is also possible to display the network in a ladder diagram by pressing the LAD switch:

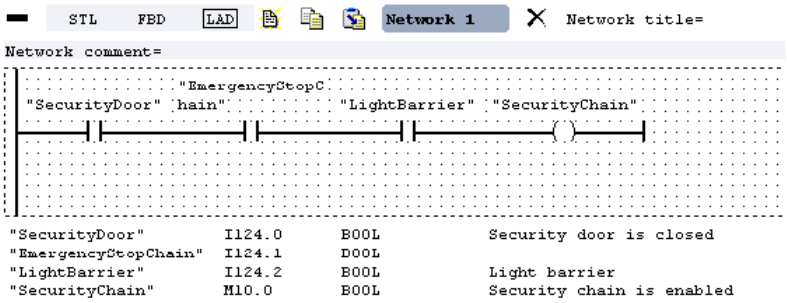


Fig.: Network in LAD representation

And here is the same network in FBD representation:

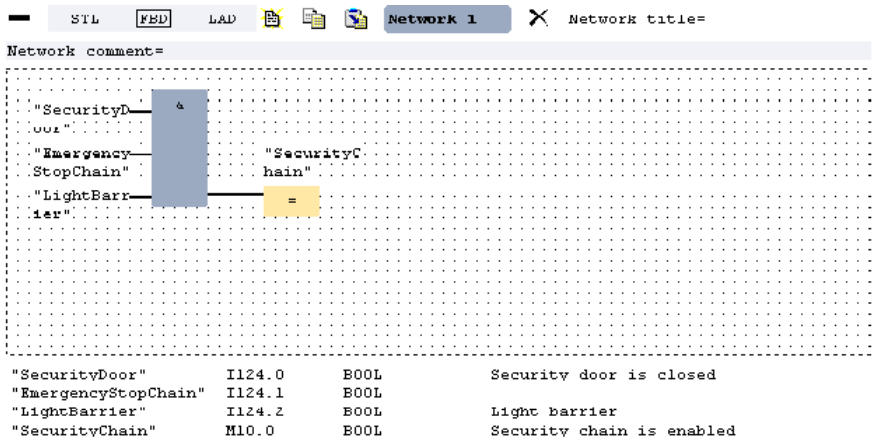


Fig.: Network in FBD representation

You can find all the settings regarding "symbols" in menu item *Extras->Settings, tab Symbolic settings*:

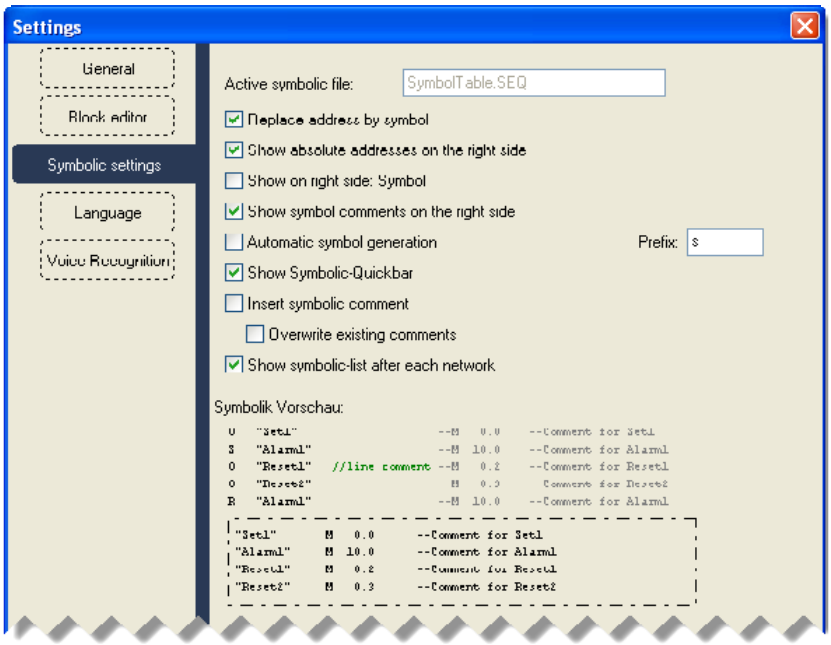


Fig.: Symbolic settings

4.3 Bookmarks in the symbol editor

The  button enables you to easily insert bookmarks.
If you press this button, a new line will be inserted at the current line of the symbolic editor. The "Symbol" column will contain the string "//{{".
Now you can enter the name of the bookmark after these characters. When you press RETURN, you will see the colored bookmark in the table and in the combo box above the symbolic editor.

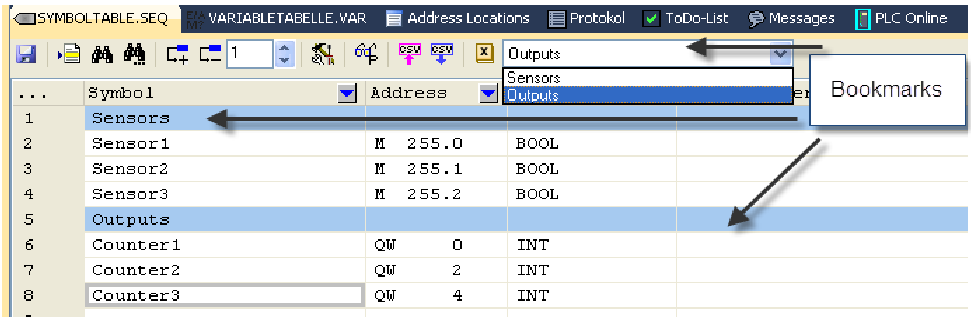


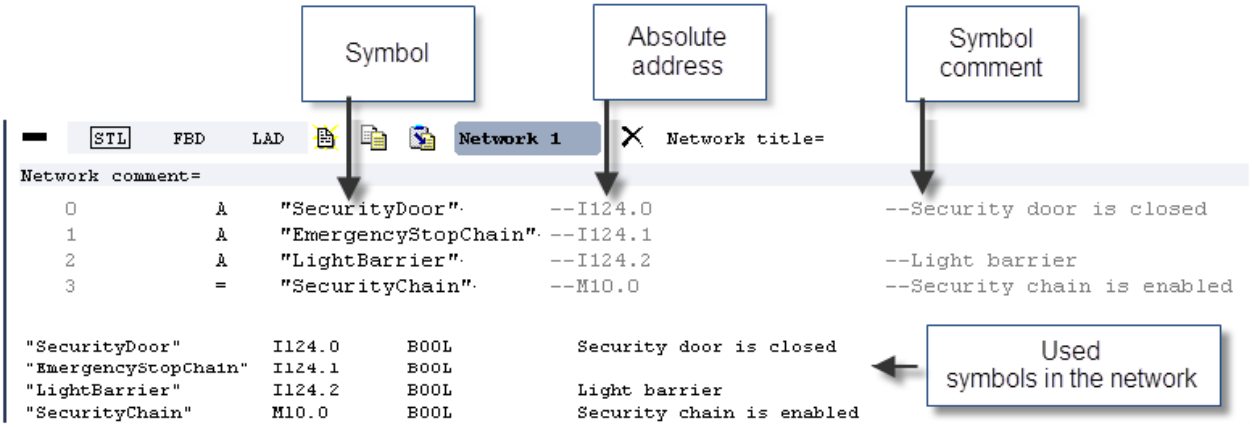
Fig.: Bookmarks in the symbolic editor

Bookmarks result in a clearer symbolism table.

4.4 Symbolic settings

The symbolic settings define how the symbols are displayed in the network.

Example of a STL network:



You can change the symbolic settings via **Extras->Settings** (Symbolic section).

4.5 Automatic symbol generation

At the start of a new project, you would normally enter the known inputs and outputs into the symbolic table.

While programming, you also need other addresses (flags, timer, counter, etc.). Later, you must add these operands to the symbolic table.

WinPLC7 V5 simplifies this task.

With the **symbolic quickbar** you can quickly change the symbolic name and the symbolic comment quickly.

How does it works?

In STL mode, if you enter the line...

```
S       M10.0
```

...and this flag does not exists in the symbolic table, then WinPLC7 generates the Symbol "sM10.0" for the flag M10.0.

When you press RETURN, the resulting STL line is displayed as follows:

```
S       "sM10.0"
```

Now you can change the symbol name and symbolic comment with the **F6 key**:

First, change the symbolic name:

Then press the **TAB key** and enter the symbolic comment:

Then press ENTER to change the line in the STL editor:

Benefits of the symbolic quick bar:

- edit the symbolic name and the symbolic comment quickly and easily.
- you can use it for new symbols as well as existing symbols
- It works with STL, LAD and FBD

You can enable and disable the symbolic quickbar in the settings (Extras->settings):

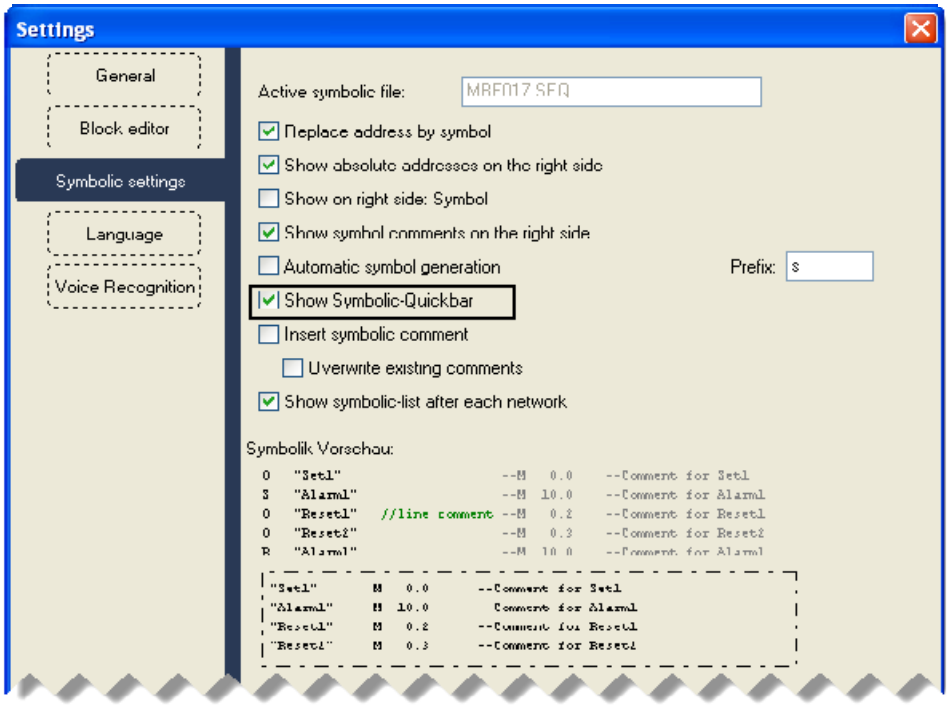


Fig.: Symbolic settings

4.6 Monitoring in the symbolic editor

Press the button  in the symbolic editor to **enable** monitoring mode.

In this mode, a new column appears. This column contains the current values of the symbols:

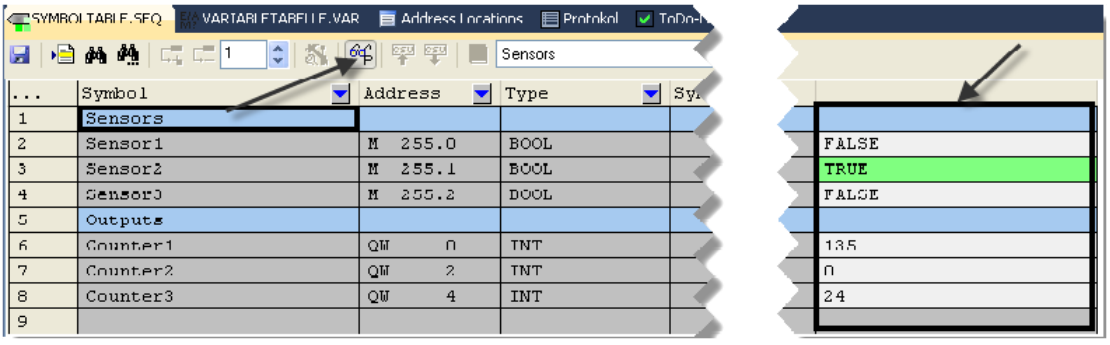


Fig.: Symbolic settings

Press the button  again to **disable** monitoring mode.

Hint:
You can monitor a block and a symbolic table simultaneously.

5 Creating an FC, FB and DB

FCs and FBs are blocks that are accessible by means of a CALL statement.
Block parameters can be used to transfer constants or addresses to the blocks. In this way, subroutines can be created that are supplied with different parameters as required.

5.1 Creating an FC (function)

Before you can call an FC or an FB it must first be created.

Open the project content window and press the button "New...". Then select "FC":

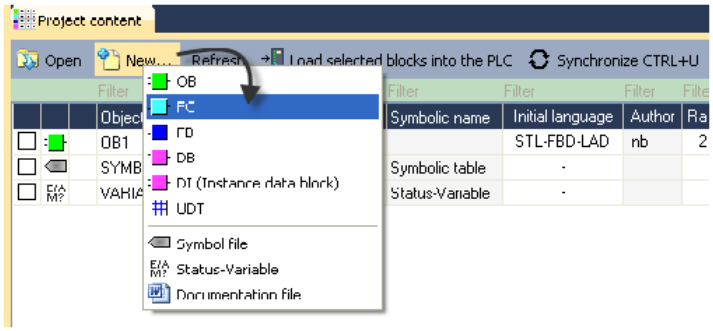


Fig.: Creating a new FC

The next unused FC number will be determined. Press the button "Create..." to create the FC block:

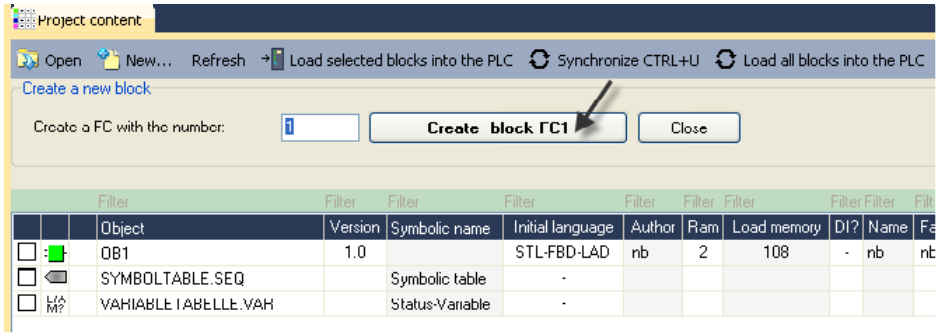


Fig.: Creating a new FC

The empty FC1 is displayed as follows:

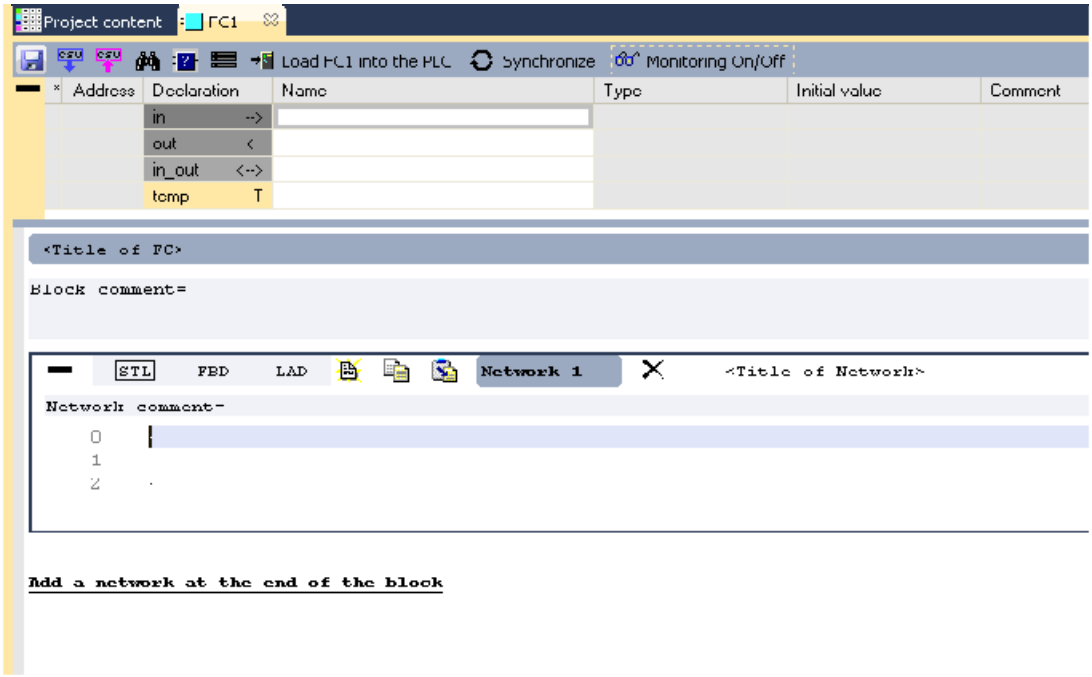


Fig.: The empty FC1

Functions (FCs) can have the following block parameters:

1. In parameters (IN)
2. Out parameters (OUT)
3. IN-OUT parameters (IN_OUT)
4. Temp parameters (Temp)

These block parameters are declared in the so-called block header. This defines the addresses of the constants that can be transferred to the block.
The so-called **Temp parameters** will not be transferred, but they are used as buffers by the block.

Every parameter has the properties "Name" and "Type". The respective initial value can not be defined for FCs.

Specifying the block parameters

When you create a block parameter, start by defining the name. For instance, enter "Switch1" into the column "Name" and press the return key once. The cursor jumps to the "TYPE" column. Here you may enter the type directly, or you can press the RETURN key again. In this case, a list with all the available types will be displayed. You may either type "bool" or select "BOOL" in the list. Press the RETURN key again. The cursor jumps to the comment column. Enter a meaningful comment for the parameter.

A row will be inserted when you press the RETURN key again. Now you can define the second input parameter:

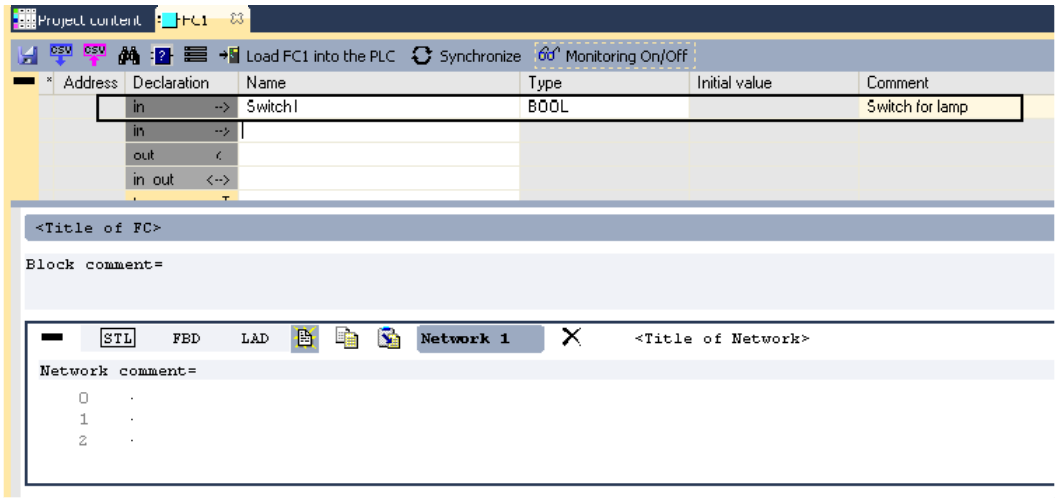


Fig.: Parameter "Switch1" was defined.

At this point we will define parameter "Switch2" in the "IN" area followed by output parameter "Lamp1":

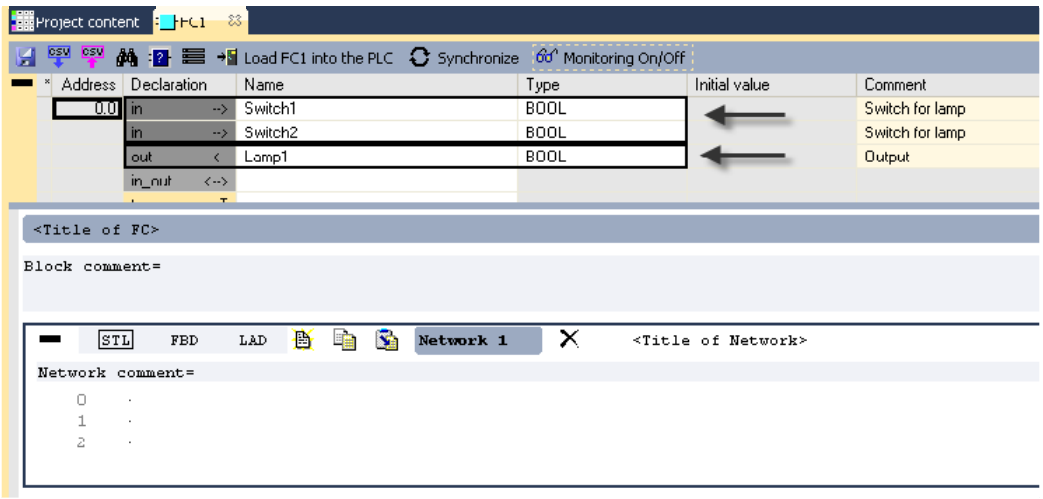


Fig.: The block header of FC1

Make sure that you have defined **Switch1 and Switch2** in the **IN** area. It often happens that an input parameter is unintentionally defined in the "OUT" area.

Now you can make use of the parameters in the code of the block. For this purpose, enter the "#" symbol at the start of each parameter.

Make sure that the network is in STL mode:

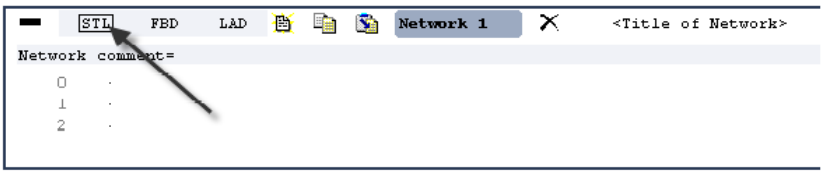


Fig.: Network 1 in STL mode

Then enter the code for network 1:

```
A    #Switch1
A    #Switch2
=    #Lamp1
```

After you enter "A #", the IntelliSense Window shows the valid parameters:

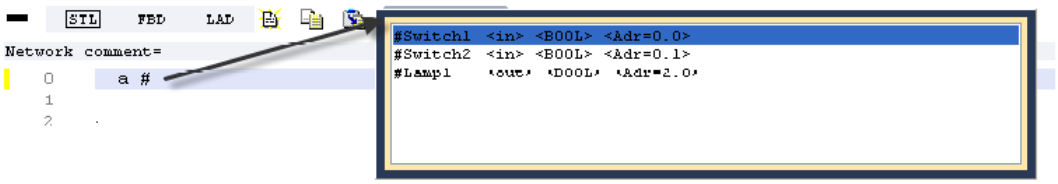


Fig.: IntelliSense Window

Select the correct address and press enter. In this way you can complete the network quickly and without typing errors.

FC1 now looks as follows:

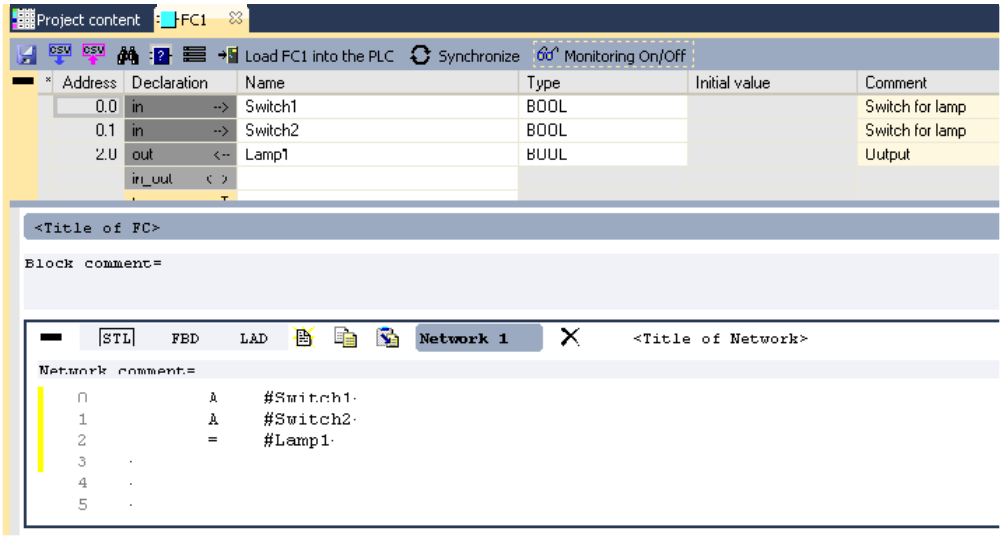


Fig.: FC1 with network 1

We now have created a function with two input parameters and one output parameter. The purpose of this function is to turn the lamp on when both switches are pressed.
We have only used the parameters in the code area. The advantage is that the function is easily called several times in succession using different parameters.

We now want to use (call) this function in OB1.
For this purpose we open OB1 (double-click the project content window).

In network 1 we enter the STL statement:

```
CALL FC1                          ->press RETURN
```

The block parameters are automatically inserted into the STL:

```
CALL FC        1
  Switch1:=I0.0
  Switch2:=I0.0
  Lamp1:=Q0.0
```

The editor automatically inserts the standard addresses. You may now replace with the required addresses.
Modify the CALL statement as follows:

```
CALL FC        1
  Switch1:=I0.0
  Switch2:=I0.1
  Lamp1:=Q0.0
```

Insert two new networks. For instance, **press** the text button "Add a new network at the end of the block" **twice**:

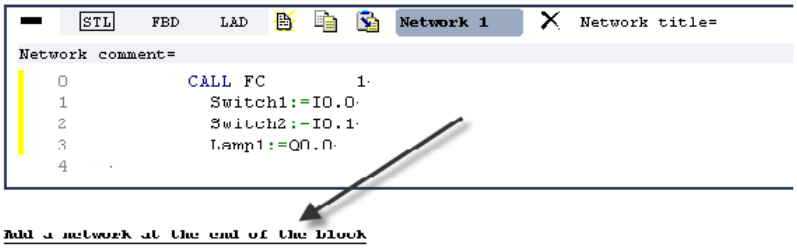
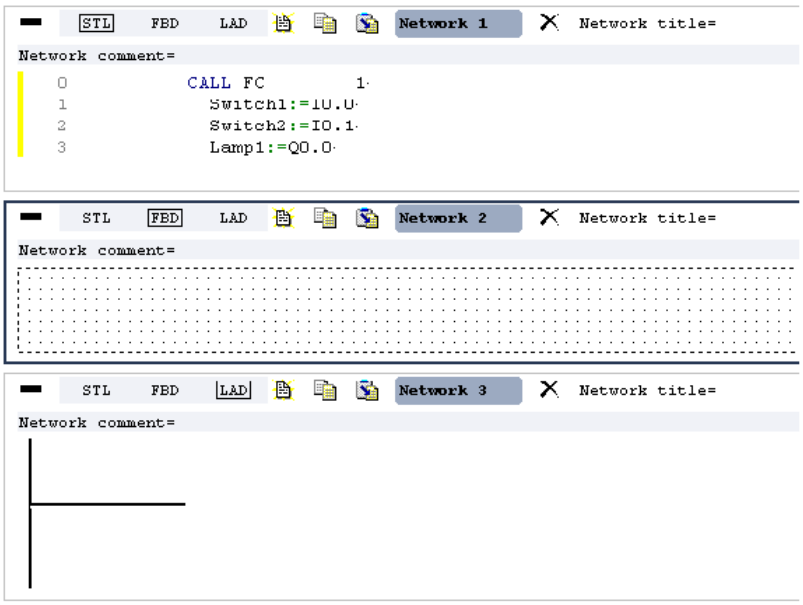


Fig.: Add a new network

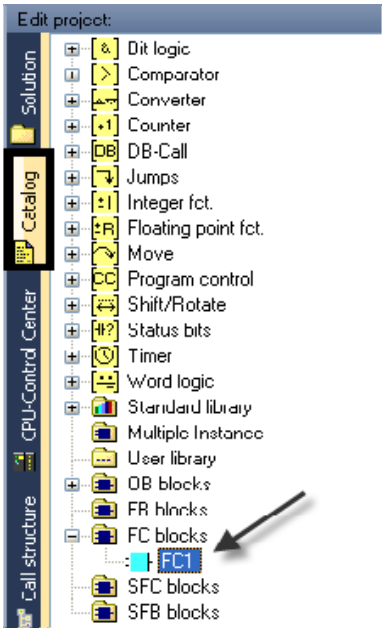
We adjust new Network 2 to "FBD" and network 3 we adjust to "LAD" representation.

Click on the code area of network 2 with the mouse.

Block must now appear as follows:



Now we add FC1 to the function diagram of network 2.



Open the node **"FCs blocks"** in the catalog. FC1 is displayed. Double-click the entry with the mouse or drag and drop FC1 into network 2.

Now the call to FC1 is located in network 2:

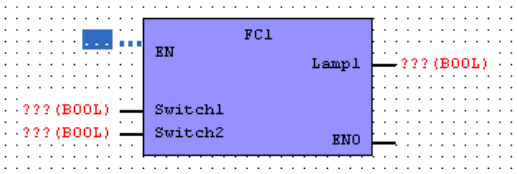


Fig.: FC1 was called

Continue by replacing the question marks with addresses.
Assign input I0.2 to parameter "Switch1", input "I0.3" to parameter "Switch2". Q0.1 is assigned to output parameter "Lamp1":

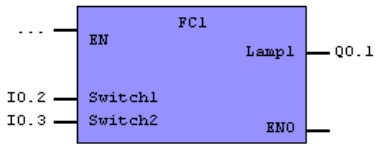


Fig.: the "wired" FC1 call

Note:
FC1 still has one **EN-input**. If this input is not connected anywhere, the function will always be called (same as in network 1),
For instance, if you connect bit memory M10.0 to this input, then the function is only called if the status of bit memory M10.0 is '1' (conditional call).
In network 3 you can insert FC1 into the LAD representation once more:

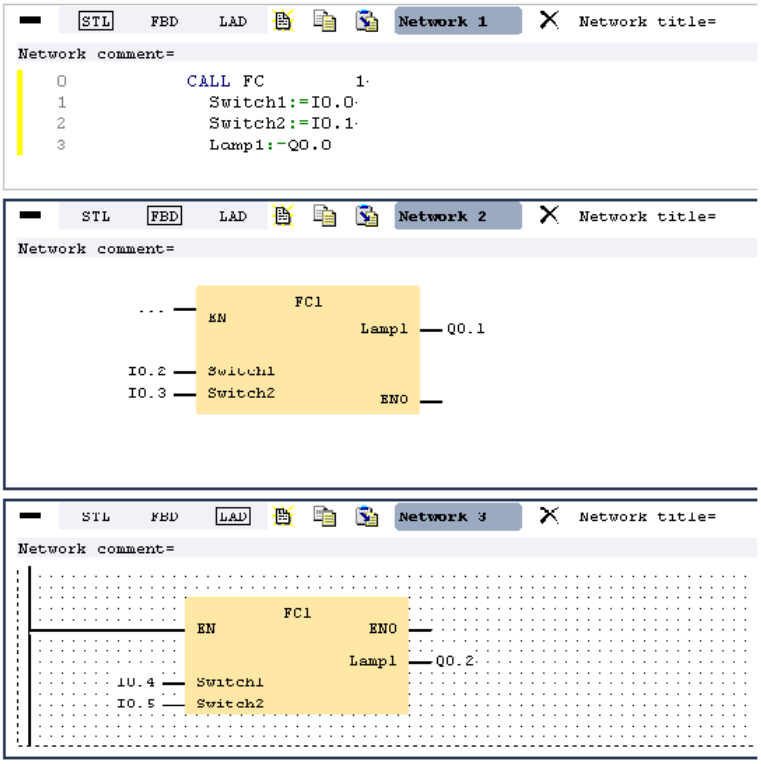


Fig.: Network 1 to 3

At this stage we have called FC1 3 times in OB1 – but each time with different parameters. The status of output Q0.0 can be changed to '1' using inputs I0.0 and I0.1. I0.2 and I0.3 change the state of output Q0.1 to '1' and I0.4, I0.5 change output Q0.2 to '1'.

Chapter 2 contains a detailed explanation of how a program may be simulated; you can now also simulate this example:

1. Load OB1 and FC1 into the Software-PLC
2. Set the Software-PLC to RUN mode
3. Test the program with the PLC mask, block monitoring, PLC-Mask or process image window.

5.2 Creating a FB (Function block) with the respective DB (data block)

In contrast to functions, function blocks require a data block where the block parameters are saved.
Advantages when compared with a function: The contents of the parameters are not lost. An FB has another section in the block header: **static local data**.
In this area you can declare variables that will retain the data during the following cycle. This means that you can use this area to save data permanently.

For the following example we will create a new project with the name **"Counter"**.
Right-click the project tree with the mouse and select menu item "Create new project".

You can create a FB like a FC using the button "New..." in the project content window:

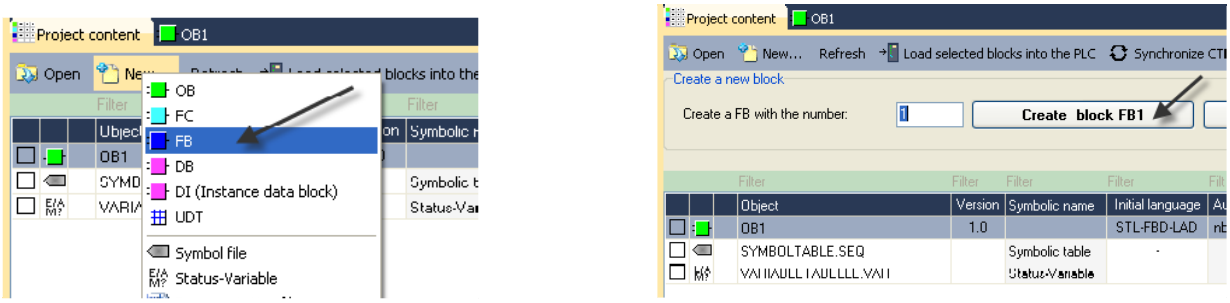


Fig.: First press the button **"New..."** then the button **"Create block FB1"**

The empty FB is displayed as follows:

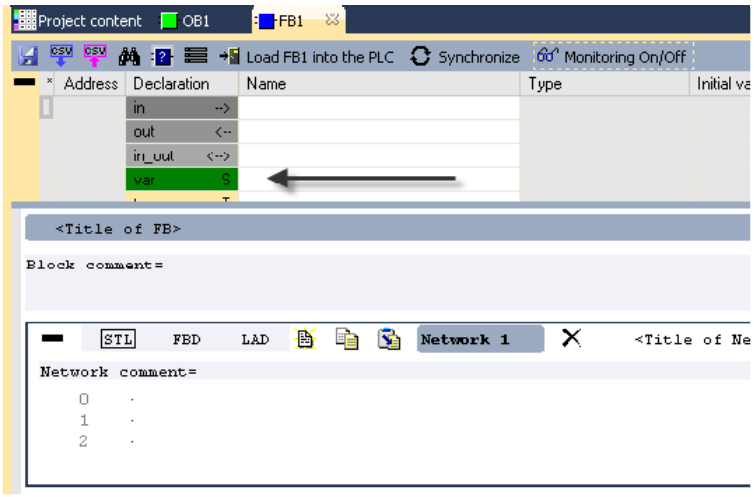


Fig.: The empty FB1

The block header contains the declaration area "VAR". This area is only available for function blocks (FB).

As an example we want to program a counter that can count from 0 to 0xFFFF FFFF.

This counter requires the following **inputs**:

Parameter	**Type**	**Explanation**
CU	BOOL	When a rising edge is applied to this parameter the counter is incremented by 1.
Reset	BOOL	When a '1' is applied to this input the counter is reset

The counter requires the following **outputs**:

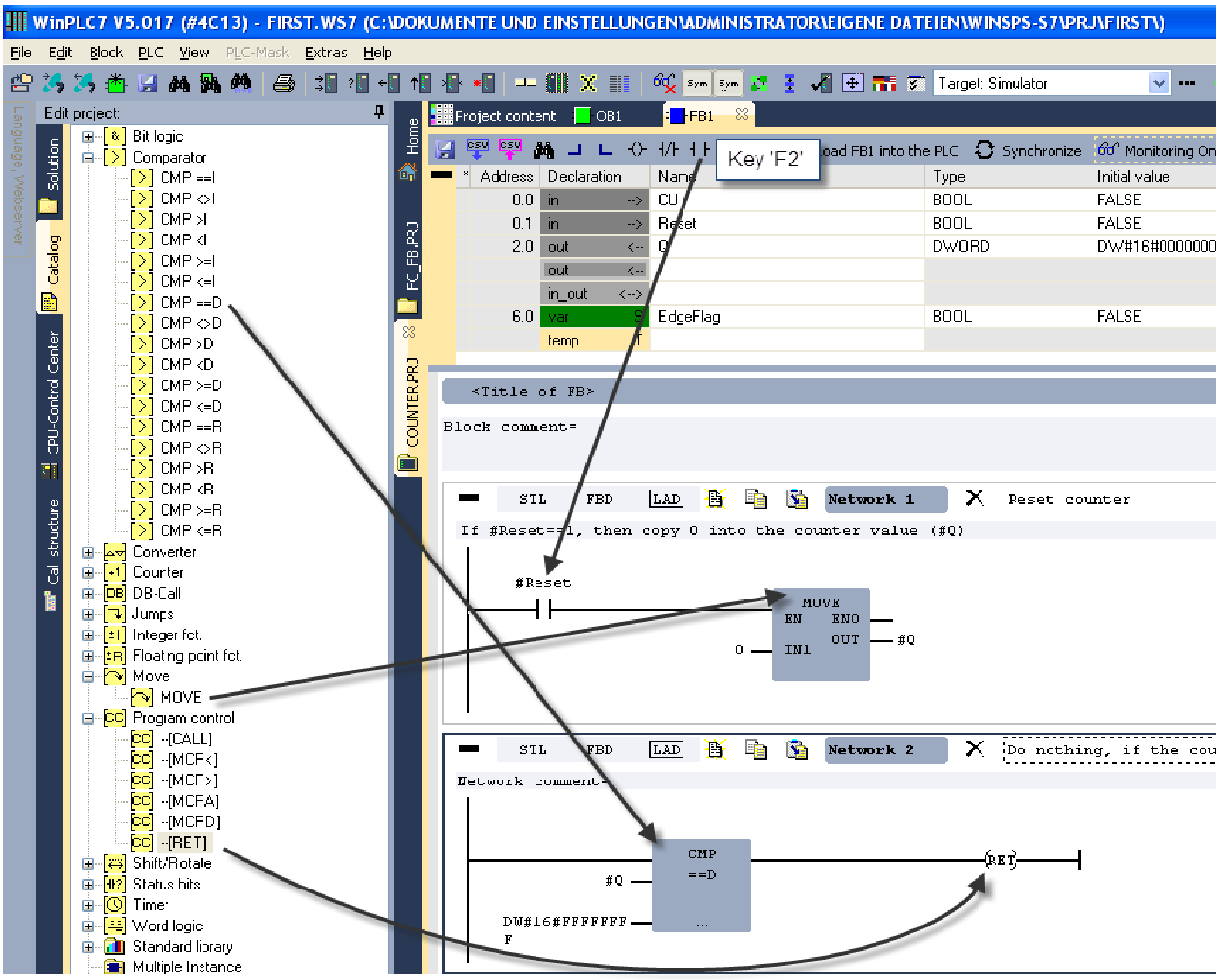
Parameter	**Type**	**Explanation**
Q	DWORD	Returns the current count

The respective block header of our FB must look as follows:

Adress	Declaration	Name	Type	Initial value	Comment
0.0	in -->	CU	BOOL	FALSE	Count up input
0.1	in -->	Reset	BOOL	FALSE	Set Counter to 0
	in -->				
2.0	out <--	Q	DWORD	DW#16#00000000	Current value of the counter
	in_out <-->				
	var S				
	temp T				

Fig.: The block header of FB1

Network 1 and 2:



The counter in **network 1** is reset when the status of parameter "Reset" is '1'.

In **network 2,** the count is checked whether it has reached the maximum value. If this is true, the routine exits from the block with the "BEC" statement.

In **network 3,** the counter is incremented by 1 when a rising edge is applied to the CU-input:
In network 3 the variable #EdgeFlag is required to detect the edge. This is created in the static area of the block header:

Adress	Declaration	Name	Type	Initial value	Comment
0.0	in -->	CU	BOOL	FALSE	Count up input
0.1	in -->	Reset	BOOL	FALSE	Set Counter to 0
	in -->				
2.0	out <--	Q	DWORD	DW#16#00000000	Current value of the counter
	in_out <-->				
6.0	var S	EdgeFlag	BOOL	FALSE	
	temp T				

Fig.: New block header of FB1

The contents of network 3:

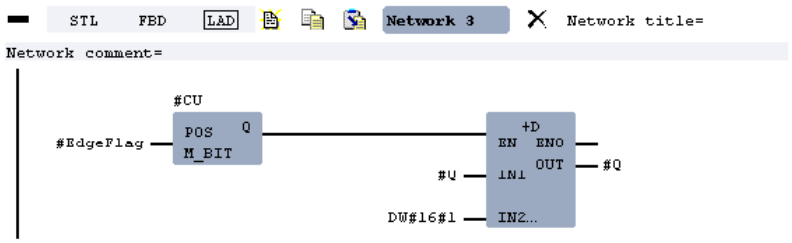


Fig.: Network 3

This completes FB1 and it can now be used in OB1.

In the FBD representation, FB1 is inserted by means of the catalog (node "**FBs blocks**"):

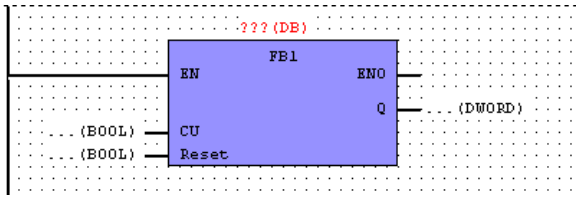


Fig.: A new FB1 was inserted.

Above the block you must insert a DB; enter "DB1" here.
Parameter "CU" is assigned to E1.0 and the "Reset" input assigned to E1.1.
The Q-output is assigned to MD10:

The completed FB1 is displayed as follows:

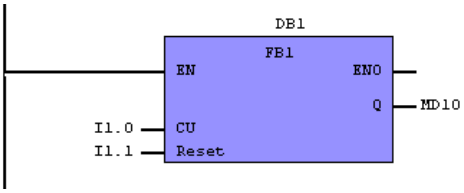


Fig.: The completed FB1.

Now you can simulate the program:

1. Transfer all the components into the simulator (menu item *PLC->Synchronize*)
2. Switch the simulator to RUN mode (menu item *PLC->Operating mode*)
3. Turn on the block status of OB1 (menu item *Block->Monitoring on/off*)
4. Enable the process image window (menu item *View->Process image Window*)
5. Change E1.0 to '1', the counter is incremented (MD10)

If you want to issue the call to FB1 at another location in OB1, you must specify another data block; otherwise the edge detection will not function properly.

The data block that is specified in the CALL is also referred to as **instance data block**. The instance data block has the same block header as the respective function block.

You can create an instance data block using menu item "File->Create new block”.

In this case, if you specify a DB you will have to specify whether an instance data block (data block with associated function block) or a global data block (data block) should be created:

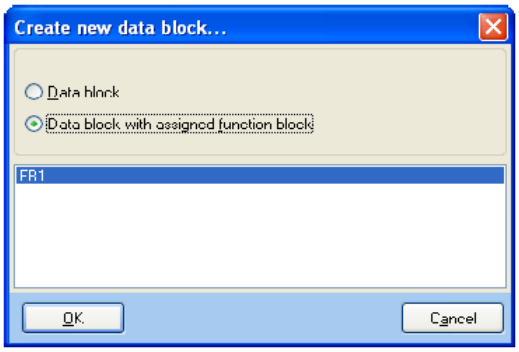


Fig.: Create a new data block

Hint:
The instance data block of a FC is generated automatically, when you enter the DB above the FB block.

6 Creating a global data block

Global data blocks serve as a global storage area that can be accessed by all other blocks.

The organization or the structure of a global data block is defined in the block header. Here you can define bits, bytes, words, double words, arrays and structures.

You can create a global data block using the button "New..." in the project content window:

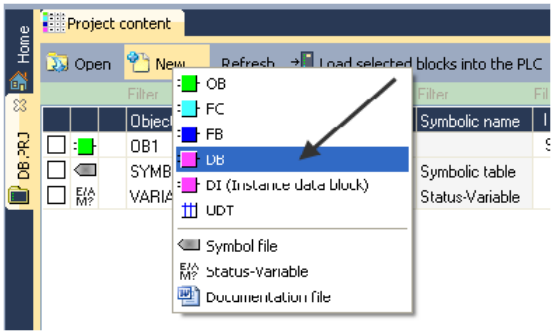


Fig.: Select "DB" in the context menu

Change the DB-number to "10" and press the button "Create...":

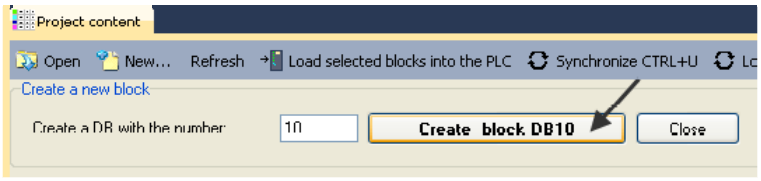


Fig.: We will create DB10

The empty DB is displayed as follows:

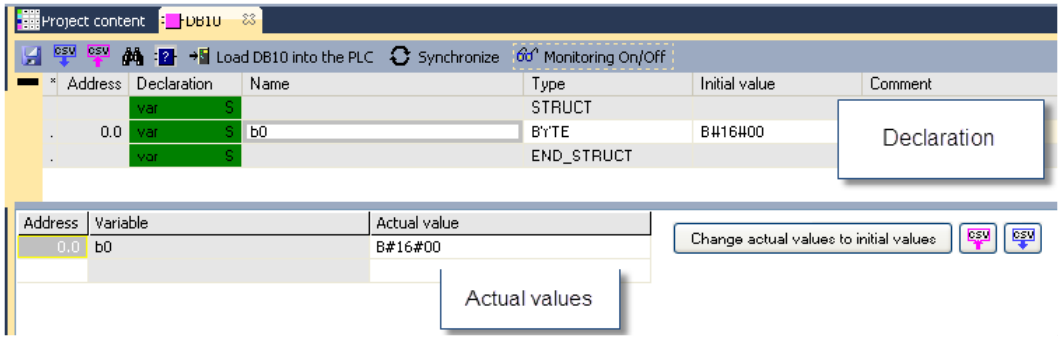


Fig.: DB10 after creation

The data block window is divided **into two areas**. The top half shows the **declaration section**. This section defines the structure of the data block. The bottom half is the section that contains the **actual values**. This displays the real content of the data block. The address of the variables is displayed to the left of the actual values. You must use this address if you want to access the variable directly (without variable name). You are only permitted to modify the section at the top. The actual values are always updated when the block is saved.
The default parameter of the DB is a byte variable with the name "b0". This can be deleted or renamed.

To insert a row.
Press the [Ins] key on your keyboard to insert a new line.

To delete a row.
Select 2 or more columns and press the [Del] key on your keyboard.

Alternatively, you can also press the right mouse key. As a result, a context-menu will be displayed. Here you can also insert and delete rows.

Edit the block header in the same manner as you edited the block header of a standard block:
When you confirm the entries in a cell with the **RETURN key**, the cursor automatically jumps to the next column.
If you confirm the entries in the last column with a RETURN, a new row is inserted and the cursor jumps to the "Name" column.

In the following example, three variables have been created in DB10:

- Delay_ms (DWORD)
- Duration_ms (DWORD)
- A1 (Array)

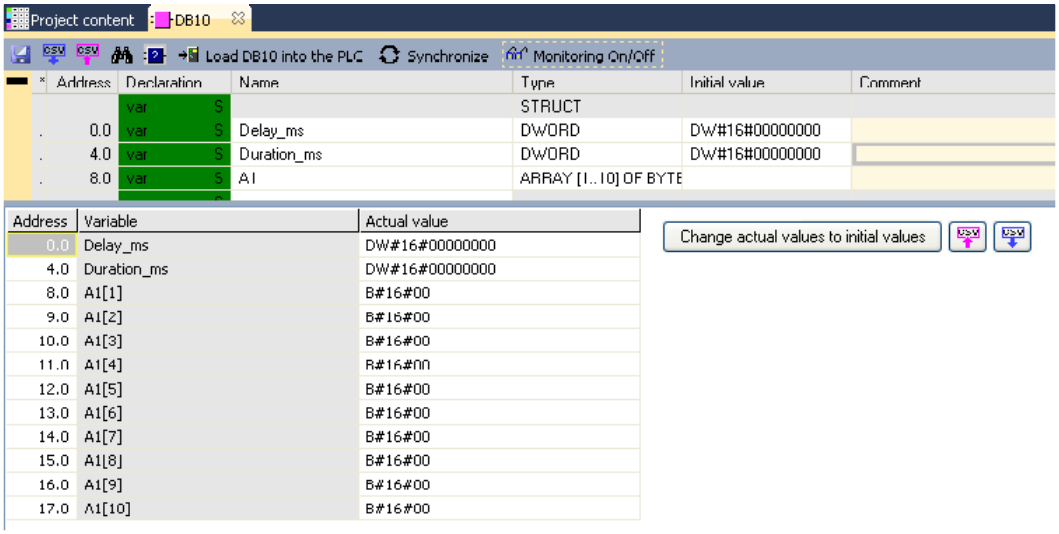


Fig.: DB10

You can access the variable in a block (OB, FB, FC) as follows:

```
L     DB10.Delay_ms
L     DB10.Duration_ms
L     DB10.A1[1]
```

If you enter "L DB" in the STL editor, the IntelliSense Windows shows the data blocks that exist in the project:

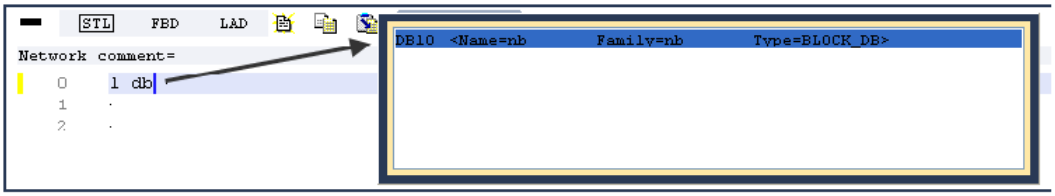


Fig.: DB10 loaded from the PLC with standard names

You can select the DB and press RETURN. If you enter the "."- character, the IntelliSense Windows appears and displays all the digital variables of data block DB10:

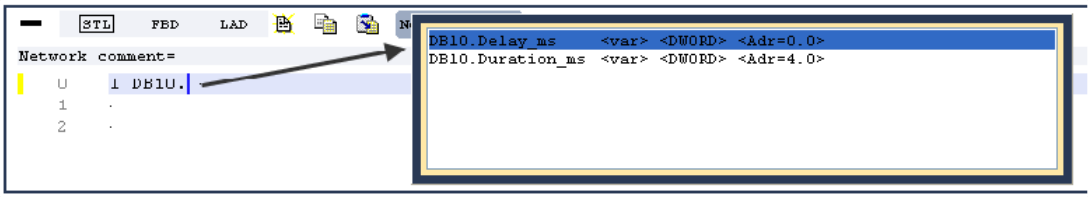


Fig.: Digital variables of the DB10

Select a variable and press RETURN to insert the variable into the STL editor.

Instead of the variable names you can also access the bytes of the DB directly (this this is not recommended):

```
L     DB10.DBD0          //access to Delay_ms
L     DB10.DBD4          //access to Duration_ms
L     DB10.DBB8          //access to A1[1]
L     DB10.DBB9          //access to A1[2]
```

If you load a data block from a PLC when the original project no longer exists, then the variable names are replaced with standard names.

The following DB10 is identical to the one on the last page. However, in this case the original names were replaced by standard names:

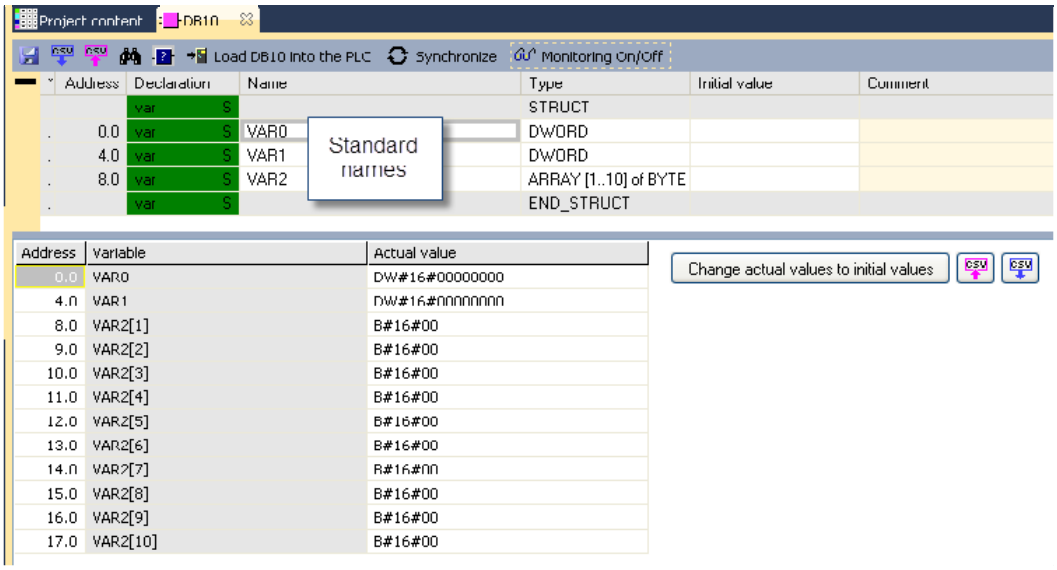


Fig.: DB10 loaded from the PLC with standard names

6.1 Change actual values to initial values

The button "Change actual values to initial values" is located to the right of the actual values.
If you press this button, the actual values (lower section of the window) are replaced by the initial values in the block header.

This may become necessary if you load a data block from a PLC and you want to restore the saved values of the DB to their original state.

7 Loading and testing the program

This chapter explains how you can load a PLC program into the S7 PLC and how it may be tested.

7.1 Establishing an online connection

First, the target must be defined correctly to enable the connection with the PLC.
The target settings depend on the MPI interface being used:

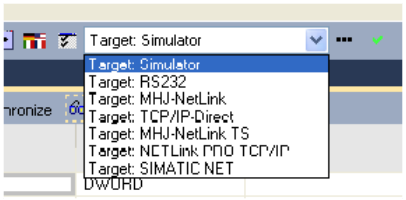


Fig.: Target selection

The following target table contains information about the target that you must select.

MPI-Interface	**Target definition:**
Serial MPI-adapter	Target: RS232
USB MPI-adapter with a virtual COM-port	Target: RS232
GreenCable supplied by VIPA, only suitable for VIPA-CPUs.	Target: RS232 The baud rate must be set to 38400.
MHJ-Netlink, Netlink-Lite, IBH-Link	Target: MHJ-Netlink
NETLink-PRO TCP/IP **NETLink-PRO Compact TCP/IP**	Target: NETLink-PRO TCP/IP
SIEMENS MPI-adapter 5.1 serial 6ES7972-0CA23-0XA0	Target: RS232
SIEMENS MPI-adapter USB (*) 6ES7972-0CB20-0XA0	Target: SIMATIC-NET
SIEMENS TS-adapter 5.2 serial 6ES7972-0CA34-0XA0	Target: SIMATIC-NET
Siemens Notebook adapter CP5512 6GK1551-2AA0	Target: SIMATIC-NET
Siemens PCI adapter CP5611 6GK1561-1AA01	Target: SIMATIC-NET
Siemens Teleservice adapter II 6ES7972-0CB35-0XA0	Target: SIMATIC-NET
Ethernet patch cable	Target: direct TCP/IP

When you select "SIMATIC-NET" as the target, the "PG/PC interface dialog" of SIEMENS must be available on the PC.
The respective start icon is located in the Windows control panel.
This dialog is available when you have installed a SIEMENS Interface driver, SIEMENS STEP®7 or the SIEMENS Teleservice software.

You can define the properties of the target by means of the [...] button located on the right of the target selection box:

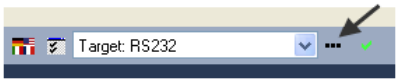


Fig.: [...] target button

7.2 Loading the program into the target system

When you have selected and configured the "target" correctly, you can access the S7 PLC.

You can use menu item *PLC->Module State* (hotkey = CTRL + D) to verify that a connection can be established with the S7 CPU is or not.

You have several options to transfer blocks into the PLC:

- Menu item ***PLC->Synchronize***
  transfers only the modified blocks/non existent blocks into the PLC
- Menu item ***PLC->Send blocks***
  displays a list where you can select the blocks that should be transferred.
- Menu item ***PLC->Send all blocks***
  transfers all the blocks of the current project.

The "PC->Send blocks" dialog is always displayed when you send blocks. Here you can verify that every component was transmitted correctly:

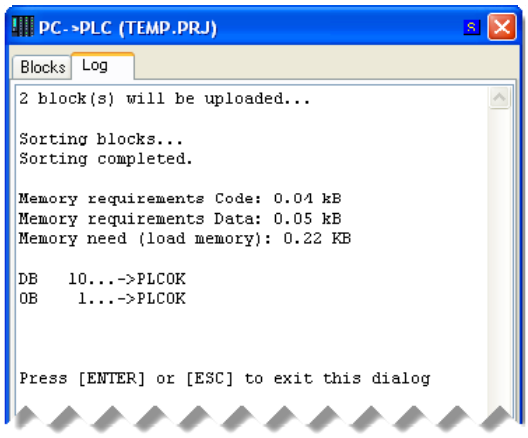


Fig.: Verification dialog when sending blocks

If you use the "Synchronize" method to load the blocks into the PLC you can watch the results in the message window:

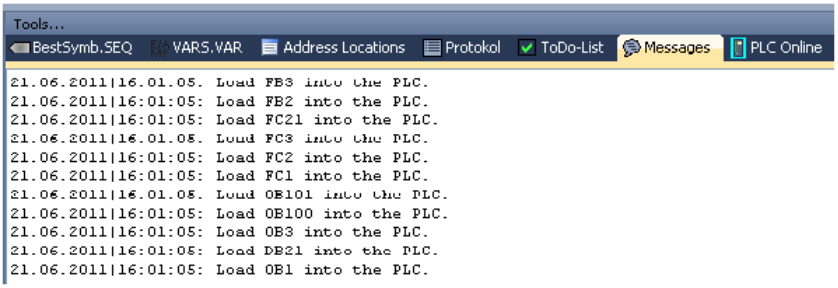


Fig.: Message log

Hint:
When a block is transferred to the S7 PLC it may be rejected by the CPU.

This could occur if the block contains a statement that is not valid for the S7 PLC or that is corrupt. For example, this can occur if you access an address (bit memory) that does not exist in the PLC.
You can locate these errors by means of the function ***Extras->Check PLC program***.

7.3 Changing the PLC to RUN mode

You can control the PLC and change its mode to RUN with the **CPU Control Center** or via menu item *PLC->Operating mode*. If the PLC does **not** change to RUN mode, a problem exists in the PLC program or in the hardware.

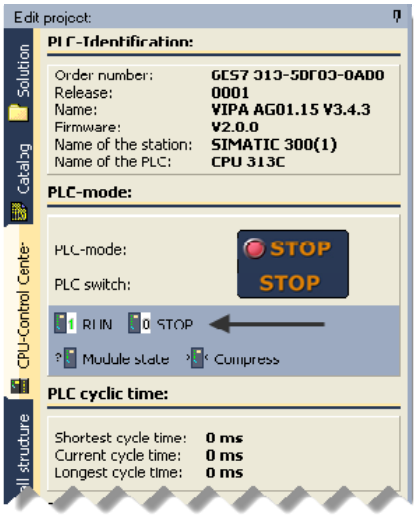


Fig.: CPU Control Center

If the PLC does not change to RUN mode you should inspect the **diagnostic buffer** first: menu item *PLC->Module status, Diagnostics tab* (Hotkey=CTRL+D).

This displays a list with the current messages of the CPU. If you require advanced diagnostics, you can display the **ISTACK/BSTACK** (ISTACK/BSTACK tab in the module status dialog).
The **ISTACK** (interrupt stack) displays the latest contents of the CPU registers (RLO, ACCU, etc.). If the problem is related to a statement in a block, the statement that was processed last is displayed.
The **BSTACK** (block stack or process stack) shows the calling hierarchy of the blocks. Here you can investigate which block calls were used to access the last block.

7.4 Testing a program

You can test the PLC program if the S7 CPU is in RUN mode.
The following options are available for this purpose:

- status block (block monitoring)
- status variable (variable monitoring)
- status variable (in the symbolic editor)

7.4.1 Status block window

Open the block that you want to monitor. Select menu item *Block->Monitoring ON/OFF.*
You can also press the mouse button with the glasses (above the block editor):

Fig.: toggle status block on or off.

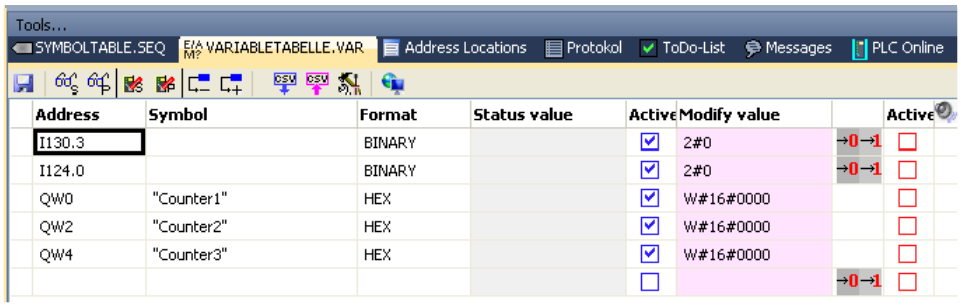
For more detailed information refer to chapter "Block monitoring".

7.4.2 Status/Modify variable window

You can use the status/modify variable window to monitor and control the contents of any address (inputs, outputs, bit memories, timers, counters, data, etc.).

Enter the respective addresses into a table in the status/modify variable window.

This window is displayed as a tab in the tool window*:*



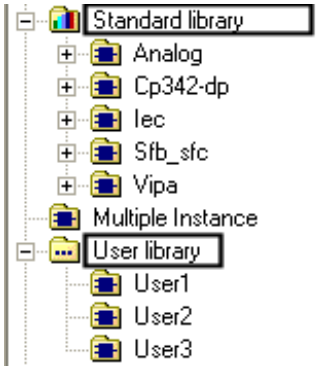
Address	Symbol	Format	Status value	Active	Modify value		Active
I130.3		BINARY		☑	2#0	→0→1	☐
I124.0		BINARY		☑	2#0	→0→1	☐
QW0	"Counter1"	HEX		☑	W#16#0000		☐
QW2	"Counter2"	HEX		☑	W#16#0000		☐
QW4	"Counter3"	HEX		☑	W#16#0000		☐
				☐		→0→1	☐

Fig.: Status variable window

For more detailed information refer to chapter "Variable monitoring".

8 Working with the library

The catalog contains the standard library and the user library:



The standard library contains the blocks that are required for ANALOG, CP342-DP, IEC, SFB_SFC and VIPA.

You can create and extend the user library by yourself.

We recommend that you use the **global clipboard** to create your own library, because this allows you to create blocks as well as STL-lines or networks.
See the next chapter on the "Global clipboard"

Follow these steps if you want to call/use a block from the library:

Open a STL, FBD- or a LAD network. Click into the code area of the network. Now the network is in edit mode.
Double-click a block in the library. This inserts a CALL statement or CALL box into the editor.
In case of a STL network:
A list containing the parameters of the block is displayed when you confirm the statement with the RETURN key.

The block that was inserted from the library will now also appear in the current project.

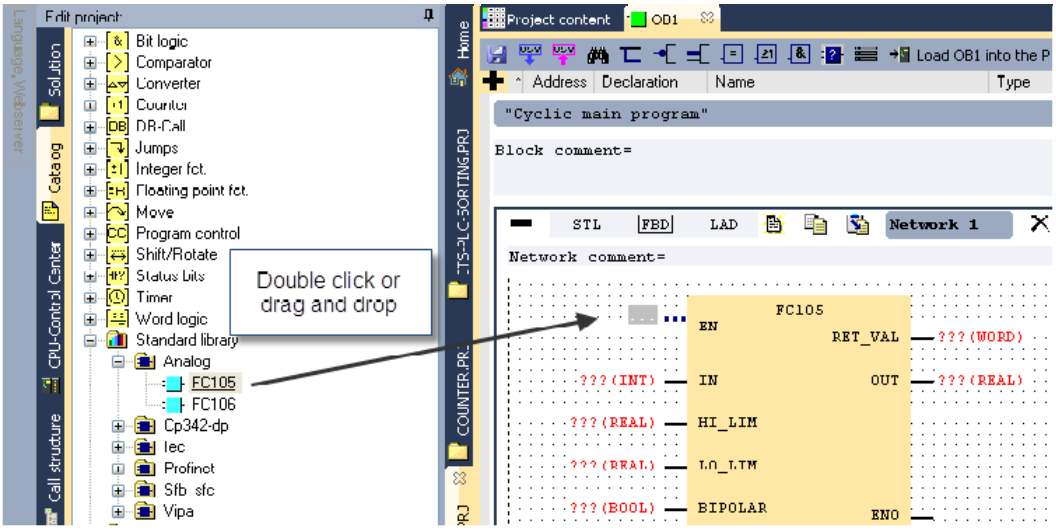


Fig.: FC105 from the library ANALOG is inserted into the project.

Now FC105 is also included in the project:

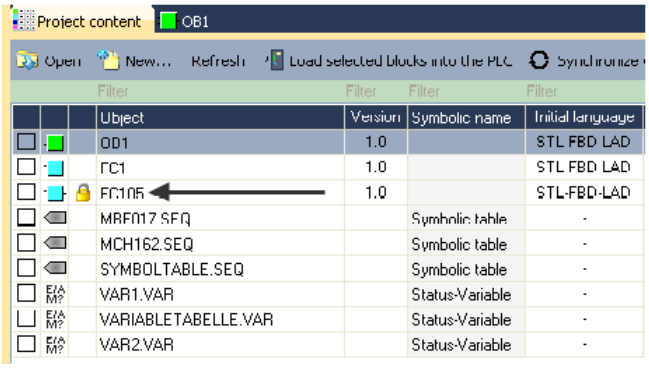


Fig.: FC105 in the window "Objectlist OFFLINE".

8.1 Extending the library

You can use drag and drop to add blocks to the USER library.

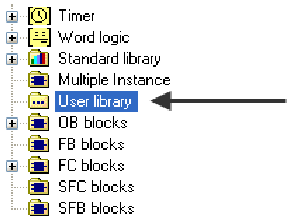


Fig.: Empty user library

First, we create a new node. Select the node "User library" and press the right mouse button:

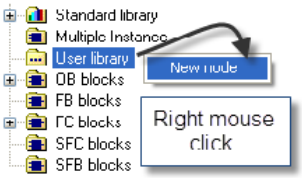


Fig.: Add a new node

Select "New node". A new node with the default name "User0" will appear. You can rename the node with a left mouse click.

Now you can add a block using "drag and drop":

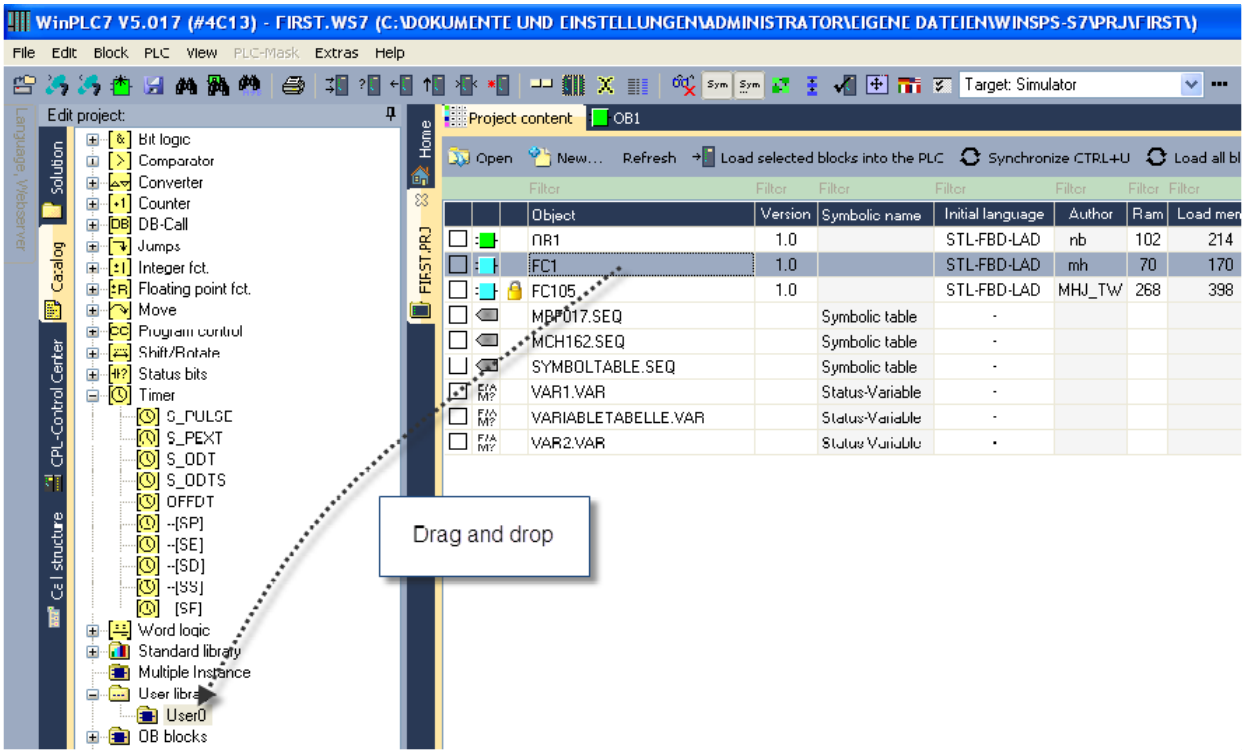
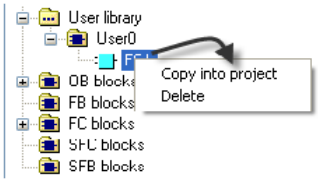


Fig.: Adding a FC block to the user library



The context menu enables you to:
- insert the object (block) into the currently active project
- delete the object from the user library

A description of the block is displayed below the catalog. This is the block title. For this reason it is recommended that you define the block title correctly:

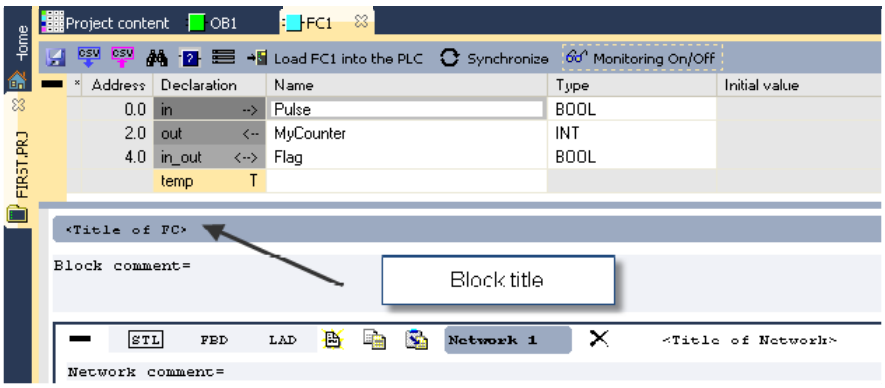


Fig.: The block title

9 Working with the global clipboard

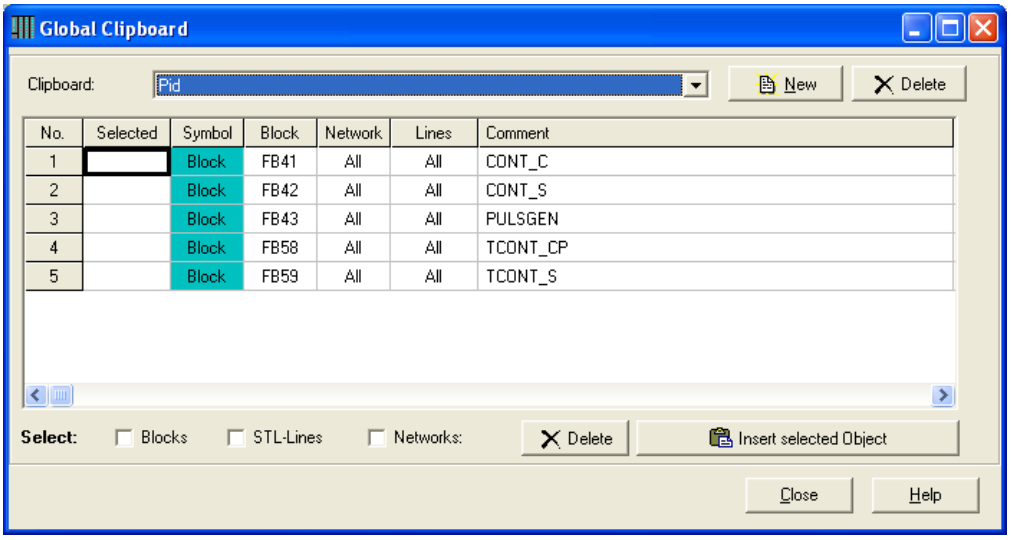


Fig.: global clipboard

The global clipboard can be used to store blocks, STL-lines and networks that must be accessible to projects and other solutions.

The global clipboard makes use of files that are retained after the PC is turned off.

The "New" button creates a new clipboard with the name that you have entered.
The "Delete" button permanently deletes the current clipboard.

9.1 Copying objects into the project

Copy an object into the current project as follows:

1. Open the global clipboard with menu item **View->Global clipboard.**
2. Select the clipboard that you want to use.
3. Select the objects that you want to insert:
   Blocks are copied into the current project.
   Networks are always inserted before the selected network.
   STL-lines are inserted into the current STL-network. The STL editor must be active, i.e. the text cursor must be visible.
4. Press the "Insert selected Obj" button

Note:
Only one object type can be inserted at a time. Therefore, it is not possible to simultaneously select STL-lines and networks in the global clipboard.

9.2 Copying objects into the global clipboard

To copy objects (STL-lines, networks, blocks) into the global clipboard, open this by means of menu item *View->Global clipboard*.
Continue by selecting the clipboard into which you want to insert the objects or create a new clipboard:

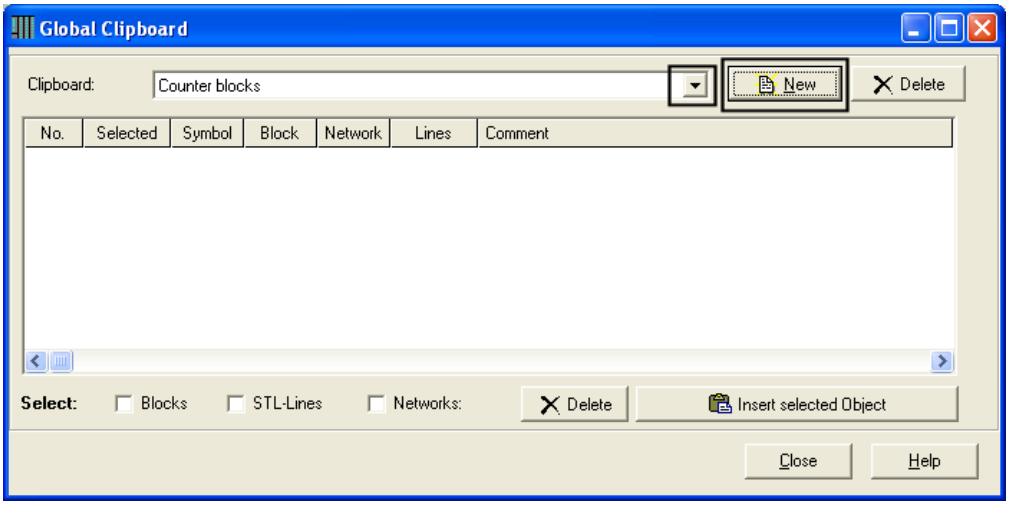


Fig.: here a new clipboard "Counter blocks" was created.

Inserting blocks into the global clipboard

You can insert blocks from the "Project Content Window" into the global clipboard:
Select the blocks that should be inserted. Continue by pressing the right mouse key.
Select "Copy to global clipboard" in the object menu.

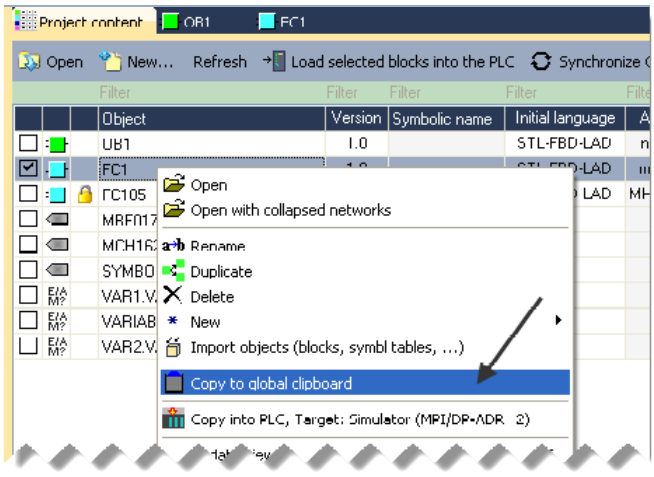


Fig.: copy a block into the global clipboard.

Inserting networks into the global clipboard

To insert a network, open a block and continue by selecting the network number of a network; then press the right mouse key.
A context-sensitive menu will be displayed; select "Copy network x into global clipboard":

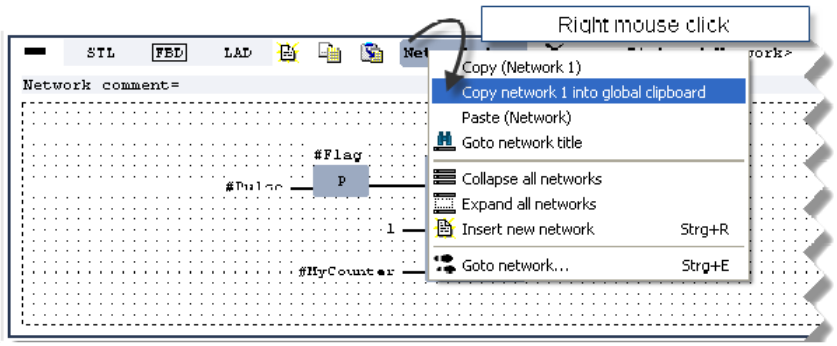


Fig.: copy a network into the global clipboard.

Inserting STL-lines into the global clipboard

Open the block and go to the network from which you want to copy the STL-lines into the global clipboard.

Note:
Change FBD or LAD networks to STL representation so that you can copy STL-lines from them.

Click on the STL with the mouse to start the text editor.
Continue by selecting the required STL-lines and press the right mouse key. Select *Copy STL-lines into the global clipboard* in the context-sensitive menu:

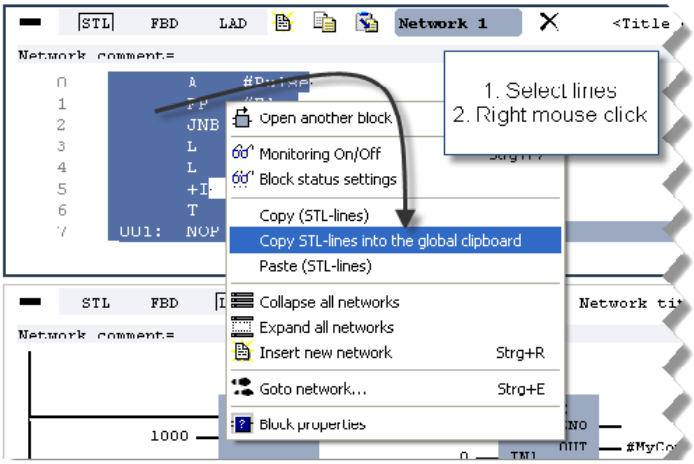


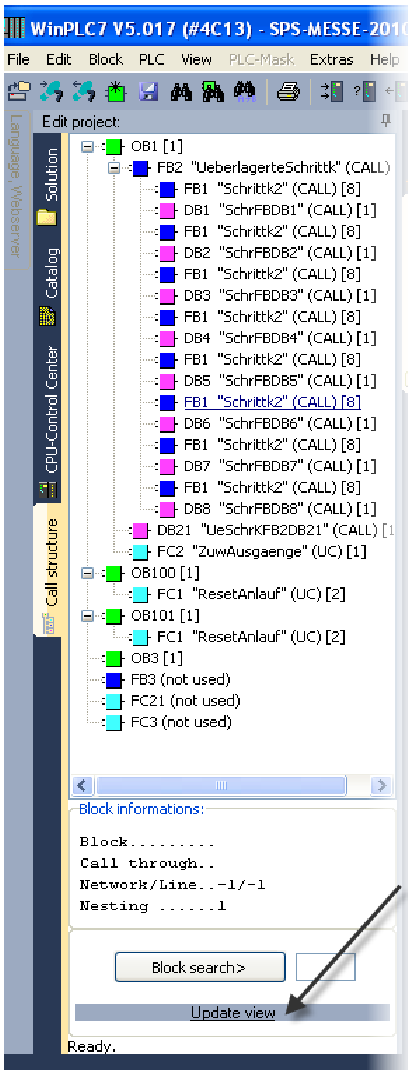
Fig.: STL-lines are copied into the global clipboard.

10 Checking the PLC program

WinPLC7 offers various options to check the operation of a PLC program:

10.1 Program structure

The program structure is displayed in the tab "**Call structure**" at the left edge of the screen:



Press the "Update view" button to generate or update the "Program structure".

The program structure shows the call-level structure of the blocks, starting with OB1.

Every item is numbered. This indicates how often the block is called from within the PLC program.

You can jump to any item by means of the context-sensitive menu (press the right mouse key). For this purpose, open the respective block editor or display it in the foreground.

10.2 Assignment plan

The assignment plan provides information about the **inputs, outputs, flags, timers and counters** that are used in the PLC program.

In addition, you can recognize whether an address is being used in a bit, byte, word or double word operation.

This display also shows whether an overlap takes place when the addresses are being accessed.

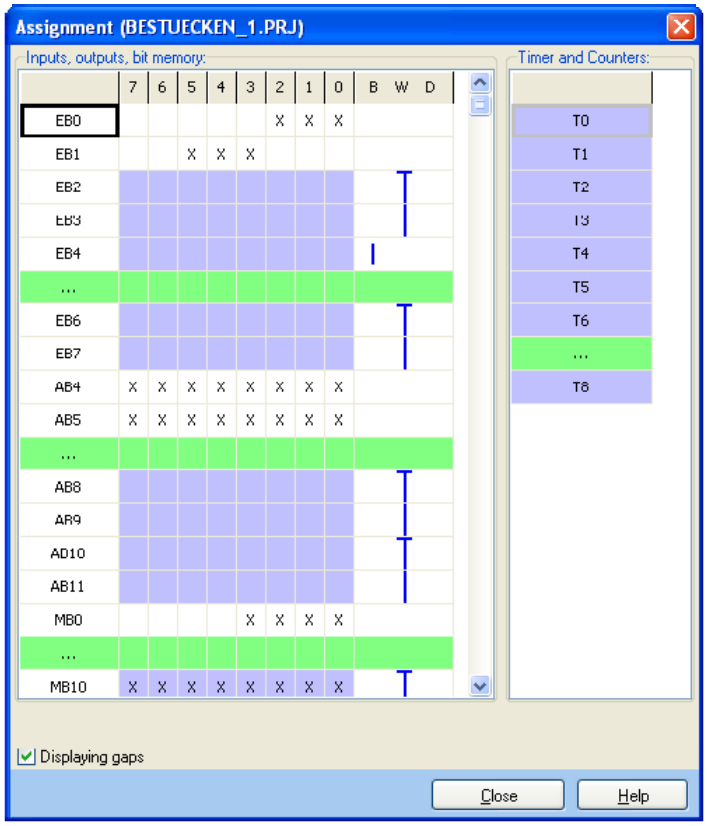


Fig.: assignment plan showing unused addresses

An "X" in columns "0 to "7" indicates that the respective bit is being read or written by a bit operation.

A vertical line in column "B" means that the byte is being read or written by a byte operation (e.g., L EB10).

A vertical line in column "W" means that the byte is being read or written by a word operation (e.g., L EW10).

A vertical line in column "D" means that the byte is being read or written by a double word operation (e.g., L ED10).

10.3 Cross-reference list

The cross-reference list shows where the different addresses are being accessed.

Using the cross-reference list you can determine the following information for every address:

- Does the PLC program use the respective address?
- Which block makes use of the address?
- Which operation is being used for the address?

The configuration dialog can be started by means of menu item *View->Cross reference list*.
Here you can specify the addresses and the blocks that must be included in the search. If the search should include the entire PLC program as well as all addresses, select "All blocks" and "All" in the "Filter" range.

The tick box "**With Diagnostic**" provides additional information. If you have selected this option, warnings are displayed when the system detects **typical programming mistakes** (e.g., double assignments, address overlaps).
Examine the resulting warnings to decide yourself whether the mistake is deliberate, or whether it must be corrected.

The configuration dialog:

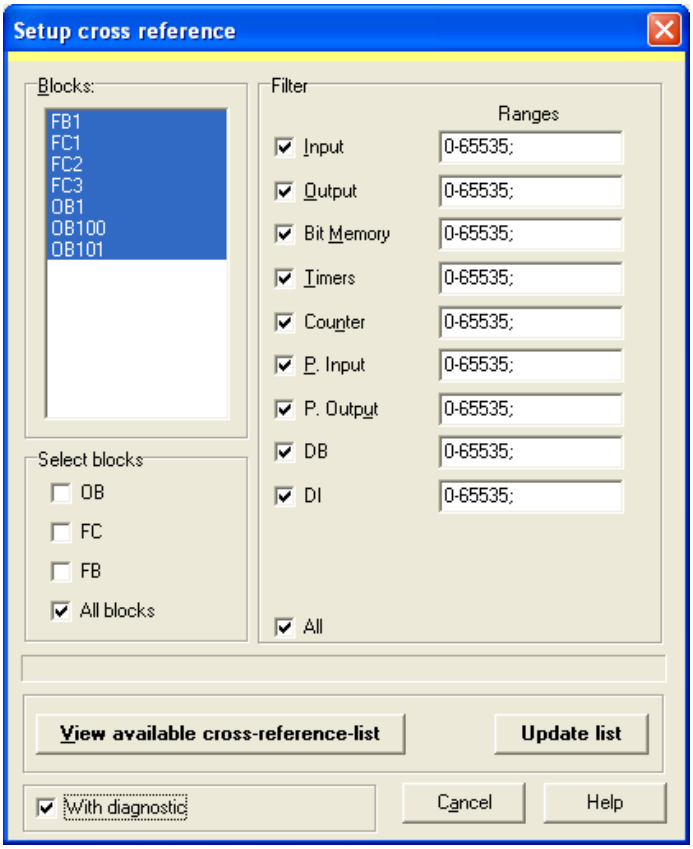


Fig.: specifying the properties of the cross-reference

The configuration dialog is always displayed first. Now you can decide whether the cross-reference that has already been determined should be displayed ("View available cross-reference list" button), or whether a new one should be created ("Update list" button).

The cross-reference list is displayed when you press the button "Create new" or "View available cross-reference list":

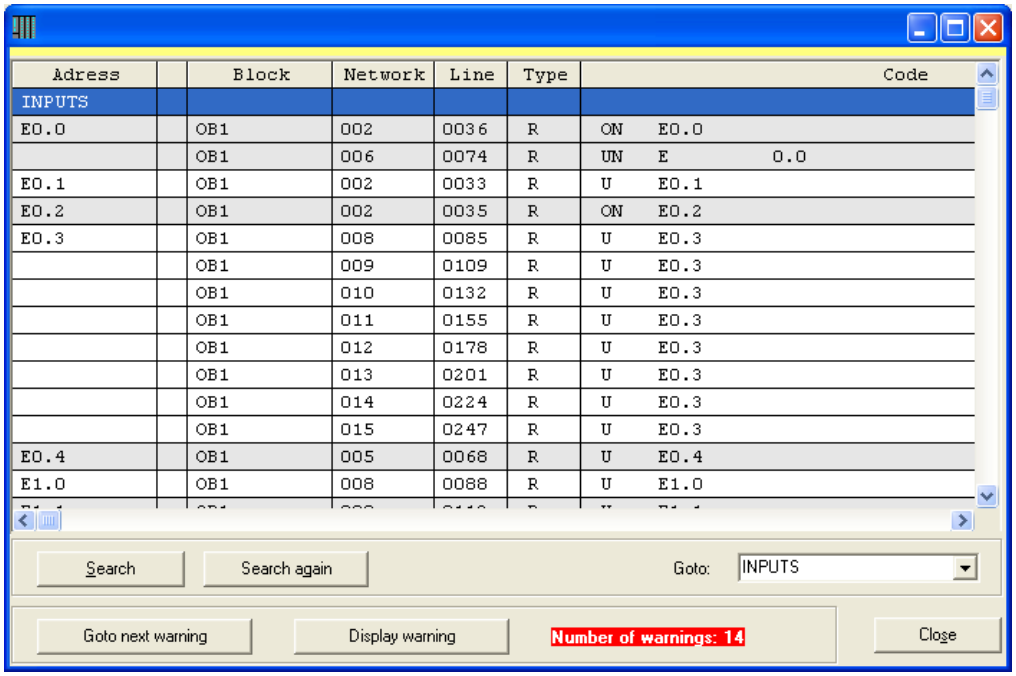


Fig.: cross-reference list

Analysis of the cross-reference list

An address is accessed in every line of the table. Each line indicates the location where the address is being accessed:

- Block
- Network
- Row
- Type:
  R=Read=read access
  W=Write=write access

The code in the last column provides information about the command used to access the address.
You can search the entire list for a specific search string by means of the "**Search**" button. This searches all the columns of the table.

A **double click** on any entry opens the respective block and displays the location where it is being used.

You can check the warnings using the button "**Goto next warning**" and "**Display warning**".

Hints:
You can use the context-sensitive menu in the symbolic editor to create a cross reference for single address.
You can also create a cross reference for a specific address in a STL or FUP/LAD network using the context-sensitive menu.

10.4 Check PLC program

The function "**Check PLC program**" compares the PLC program with the properties of a specific S7 CPU.

If the function detects incompatible commands, it will create a list detailing the errors. In this manner you can quickly determine whether a PLC program can be executed on a specific S7 CPU or not.

An ONLINE connection is necessary to access the properties of the CPU. This determines the properties of the CPU.

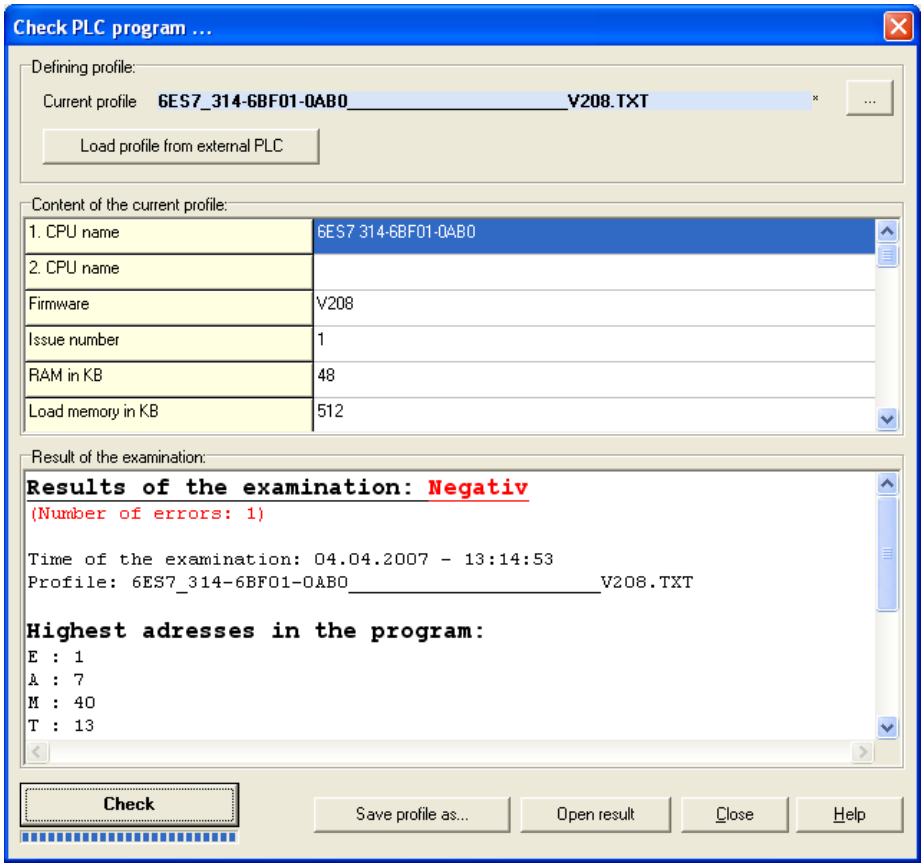


Fig.: check PLC program dialog

Procedure:

1. Connect the S7 PLC and check the connection
2. Open menu item *Extras->Check PLC program*
3. Press the "**Load profile from external PLC**" button and wait until the properties have appeared in the table.
4. Press the "Check" button
5. The result of the check is displayed in the field "Result of the examination".

The following properties are checked:

1. Access to inputs in the invalid area
2. Access to outputs in the invalid area
3. Access to bit memories in the invalid area
4. Access to timers in the invalid area
5. Access to counters in the invalid area
6. Does the CPU support all the OBs that are used in the program?
7. Does the CPU support all the SFCs and SFBs that are used in the program?
8. Are all the FBs, FCs and DBs located in the valid number range?

10.5 Consistency check

You can start the consistency check via menu item *Extras->Consistency check*.
The entire PLC program is checked for the following:

- Do all the instance data blocks have the latest revision level?
- Does the program contain conflicting time stamps (in CALLs)?
- Is every CALL still correct?
- Are all the block headers that employ UDTs still current?

For more detailed information refer to chapter "Consistency check".

10.6 Displaying the UDTs that are being used

Menu item Extras->Show used UDTs produces a list containing all the UDTs that are being used by the PLC program in a dialog.

This function provides an overview of whether and where the UDTs have been used.

UDT = **U**ser **D**efined **T**ype

An **UDT** summarizes variables in a single type.

10.7 Displaying local instances in FBs

This function provides information of whether and where local instances have been used in the PLC program.

The function is accessed via menu item *Extras->Show local instances in FBs*.
A dialog lists all available local instances.

11 Programming of a multiple instance

11.1 Creating an instance in a FB

Open a FB to create an instance in a FB.
You can now define the instance in the "VAR" area of the block header:

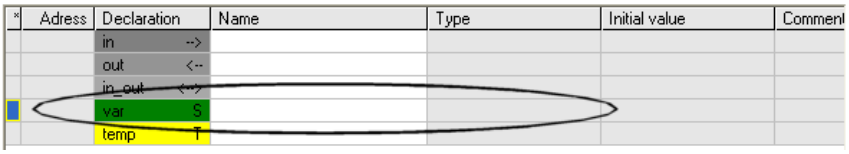


Fig.: VAR-area of the FB

If you want to create an instance of type **FB10**, specify a variable name in the column "Name" and enter "**FB10**" into the “Type” column:

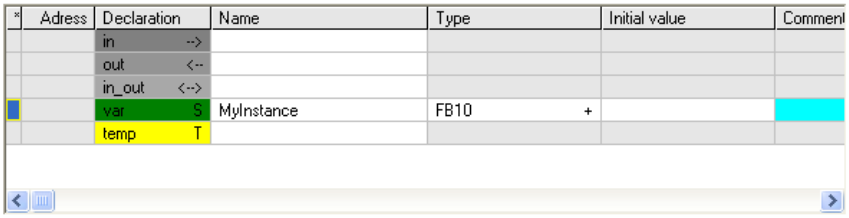
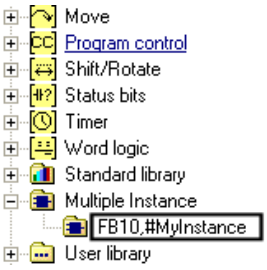


Fig.: a local instance "MyInstance" was created.

If you save the block using CTRL+S, the instance is entered into node "Multiple Instance" in the catalog:



The next paragraph explains how you can call the instance.

Note:
The column "Type" in the block header shows a plus sign to the right of "FB10". You can open FB10 with a click on the plus sign with the mouse.

11.2 Calling the instance in the FB

You can call instance "MyInstance" that was created in the previous paragraph in the code section of the same FB as follows:

```
CALL #MyInstance
```

If you confirm the line by pressing return, the parameters of the instance (in this case the parameters of FB10) will appear in a list:

```
CALL #MyInstance
  In1:=I0.0
  In2:=I0.0
  Error:=Q0.0
```

You can now assign addresses to the standard parameters (to the right of ":=").

You can also insert the call to the instance in FBD or LAD. Simply double click the entry in the catalog:

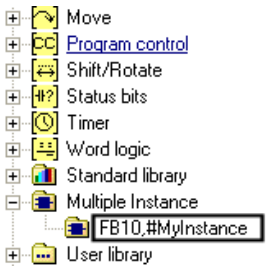


Fig.: the instance in the catalog

You can also **drag and drop** the instance from the catalog into a FBD or LAD network.

Background info:
The local instance (here "MyInstance") occupies memory in the same data block as the other variables in the block header.
Advantage of local instances:
Only one data block is required to store the data.

12 Monitoring blocks

You can observe the operation of a block with the status block function (monitor block). This will display the current value of the addresses, the result of logic operation (RLO), the accumulators, etc.

12.1 Monitoring the STL

A STL network being monitored can be displayed as follows:

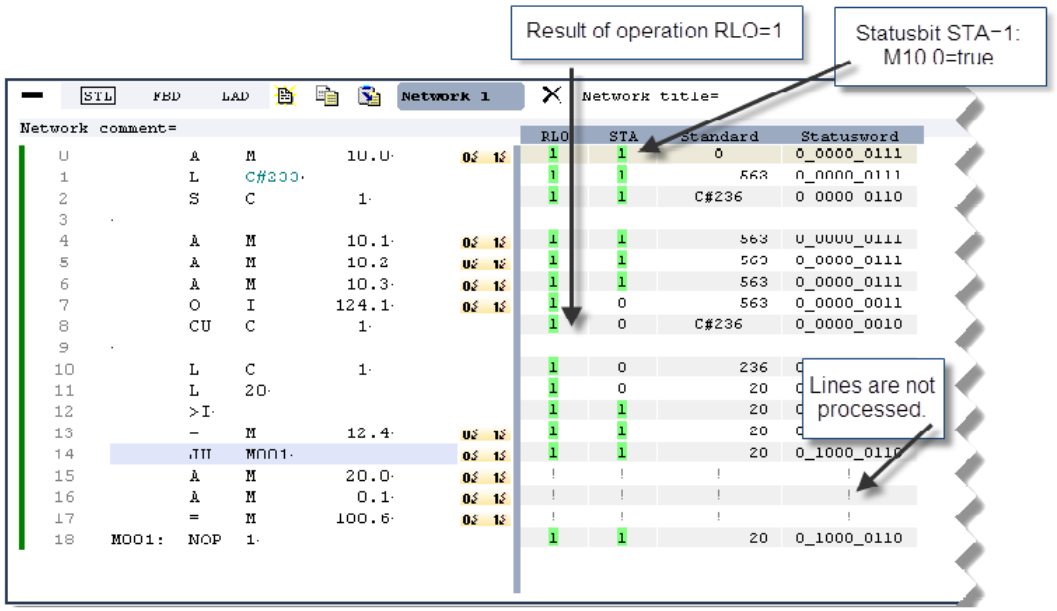


Fig.: an STL network in status mode

STL lines that have not been processed or that are **not associated with status information** are identified by an **exclamation mark**.
In the example above, three STL lines are by-passed by a jump statement. No status information is available for the bypassed STL lines because the S7 CPU did not process these lines.

Important:
The S7 CPU can only supply a limited number of status information items in a cycle. For this reason it often happens that the status information is only displayed up to a certain STL line.
In the figure above the green arrow indicates the line number that contains the first status line. The blue dots indicate lines that contain a STEP®7 command (i.e. blank lines do not have blue dots).
You can now move the STL line containing the first status information to the bottom with a mouse click on a blue dot.

You can also move the visible window using the scroll bar. The start of the status is adjusted automatically.

You can modify the block status settings using the context-sensitive menu (right mouse key).

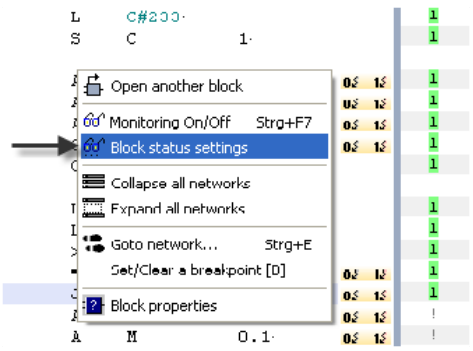


Fig.: the context-sensitive menu

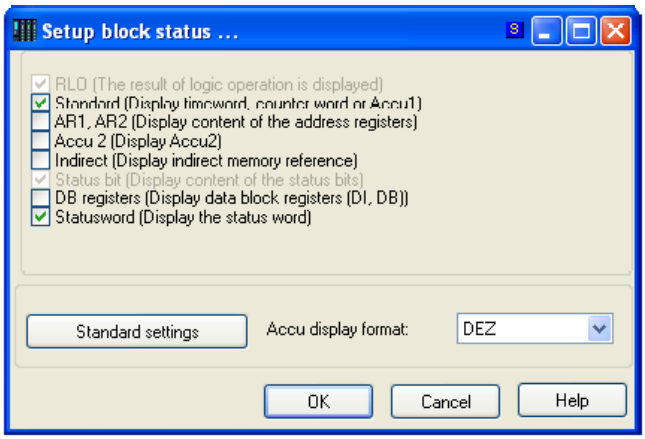


Fig.: the block status settings

In the settings you can change the Accu number format or you can choose to display additional registers of the S7 CPU.

The more information you add, the lower the number of STL lines that can be displayed in the status.

12.2 Monitoring the FBD/LAD

The FBD network is displayed in the block status as follows:

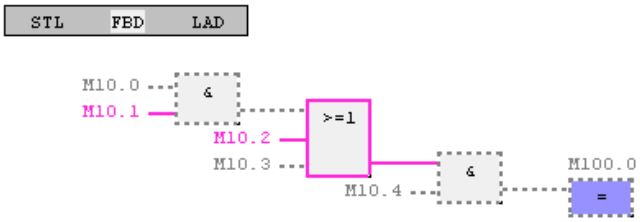


Fig.: FBD in status operation

If a bit address is **gray**. it has the state **'0'**.
If it is displayed in color, its state is **'1'**. If the bit address is **black**, **no status information** is available.

The following is a LAD network in status operation:

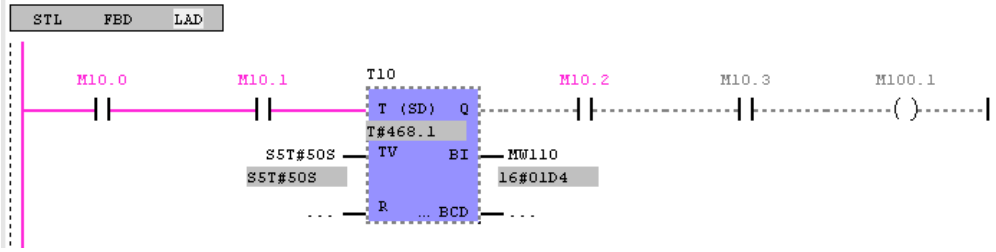


Fig.: LAD in status operation

If the line in the LAD is displayed in **color and as a solid line**, this corresponds to status **'1'**. If the line is **gray and dashed**, this corresponds to status **'0'**.

In the following network the actual values of MW110, MW112 and MW114 cannot be displayed because no status information is available.

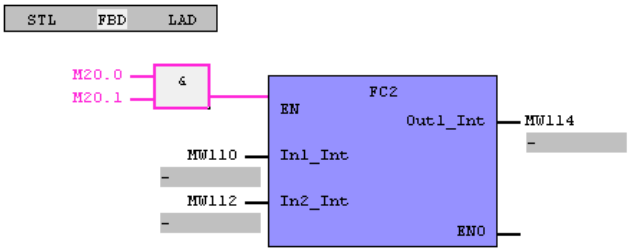


Fig.: actual values of FC2 cannot be displayed.

When you click on first address in the network (M20.0) the status will be displayed, starting with this address:

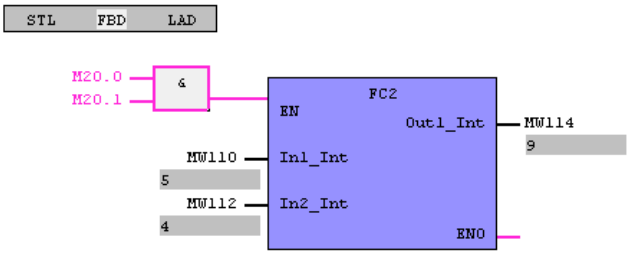


Fig.: now the actual addresses are displayed.

12.3 Changing the block while monitoring is active

To change the block while monitoring is active, open the context-sensitive menu with a click of the right mouse key.
Select menu item *Open another block*:

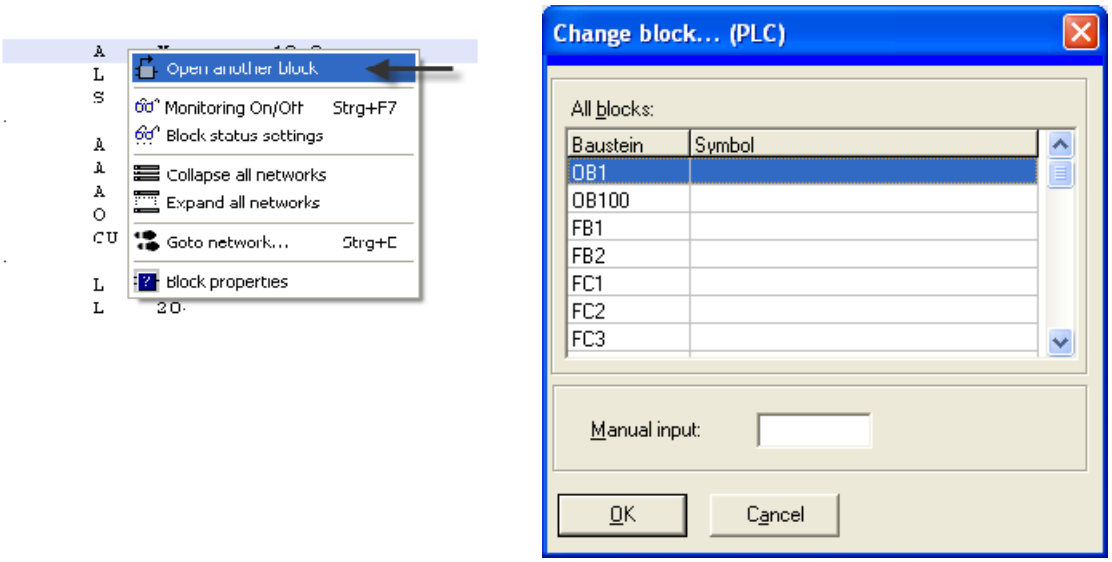


Fig.: changing the block while monitoring is active

You can then select the new block from a list.

13 Useful functions for STEP®7 beginners

If you are not yet acquainted with the STEP®7 programming language, then this section contains useful hints that can simplify the task somewhat.

13.1 Function to check a PLC program

When you have completed a PLC program and you want to transfer this to a PLC it is possible, that certain blocks are rejected by the PLC.
An error message informs you that the block could not be sent to the PLC.
However, the real reason cannot be displayed at this point.

This often happens when an invalid command is encountered in a block.
For example, a S7 PLC only supports 256 timers (0-255) and you may be using timer T256 in your PLC program.

If you transfer the block the following dialog is displayed:

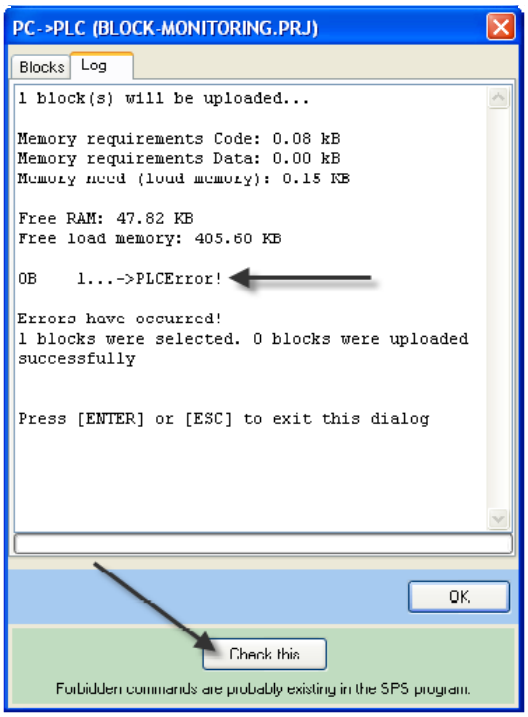


Fig.: communication error

The dialog displays the “Check this” button. If you press this button, the PLC program will be examined for these errors.

In our case the result is as follows:

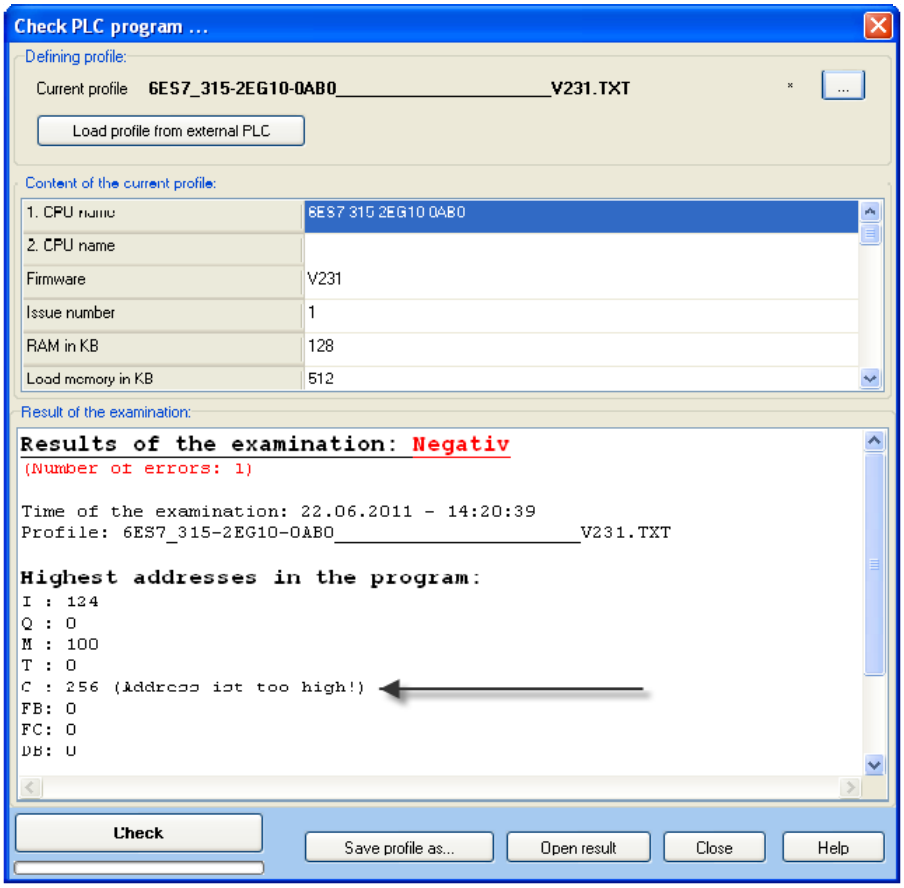


Fig.: the result of the error check.

If this feature did not exist, you would have to open the assignment plan or the cross-reference, and then compare the maximum value assigned to the addresses with the properties of the CPU (see module status).
Function *Check PLC program* is also available via menu item "Extras".

13.2 Analyzing the PLC program in the cross-reference list

When you are generating the cross-reference (menu item *View->Cross reference list*), you can activate tick box **"With diagnostic"**.
When the cross-reference list is being displayed, this function also includes typical programming errors as warnings in the resulting display. These are, for instance:

- Double assignment of addresses
- Duplicate utilization of timers, counters
- Address overlaps in bit, byte, word and double word operations

If you have made one of these mistakes in your program, your PLC program behaves strangely and not at all as you would expect it to behave.
In this case you should examine these warnings one by one.

You can find more information about this in chapter 10 "Check PLC program".

14 Working with the Software PLC of WinPLC7

The integrated Software PLC enables you to verify the operation of your PLC program. The software PLC behaves exactly like a real S7 PLC when it encounters errors in a program. For this reason, you can also use the "**Module state**" dialog in an attempt to determine the reason for a STOP.

The simulation environment of WinPLC7 offers the following options:

- Monitoring of blocks in the block editor.
- Monitoring of variables using the status variable window.
- The virtual graphic PLC, i.e. the PLC mask simulation, graphically displays a S7-300®. You can also add modules to this PLC. It is also possible to display analog inputs and analog outputs.
  You can control the inputs (analog or digital) by means of the mouse or the keyboard.
- The process-image-window can display a compact image of inputs, outputs, bit memories, timers, counters and data block addresses.
- Single-stepping through the PLC program (debug mode).

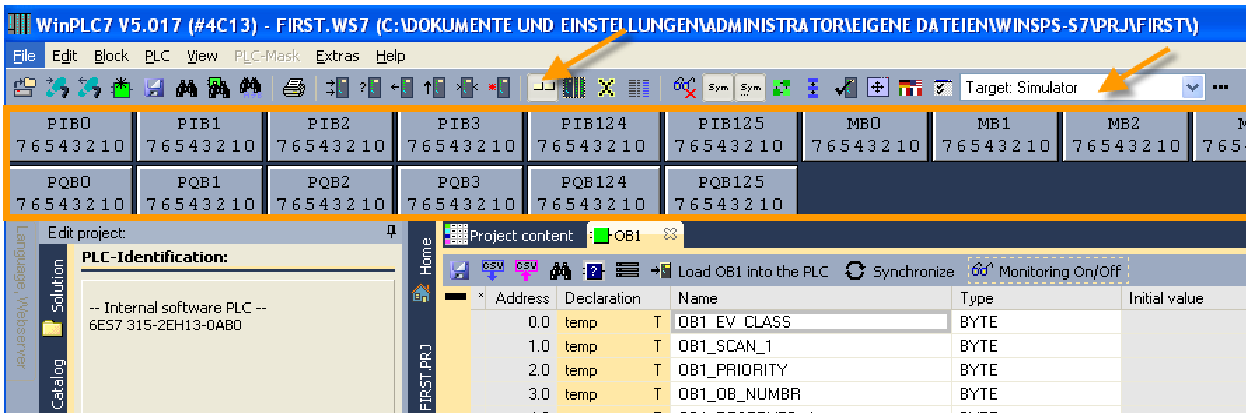


Fig.: Process-image-window (menu item View process-image-window)

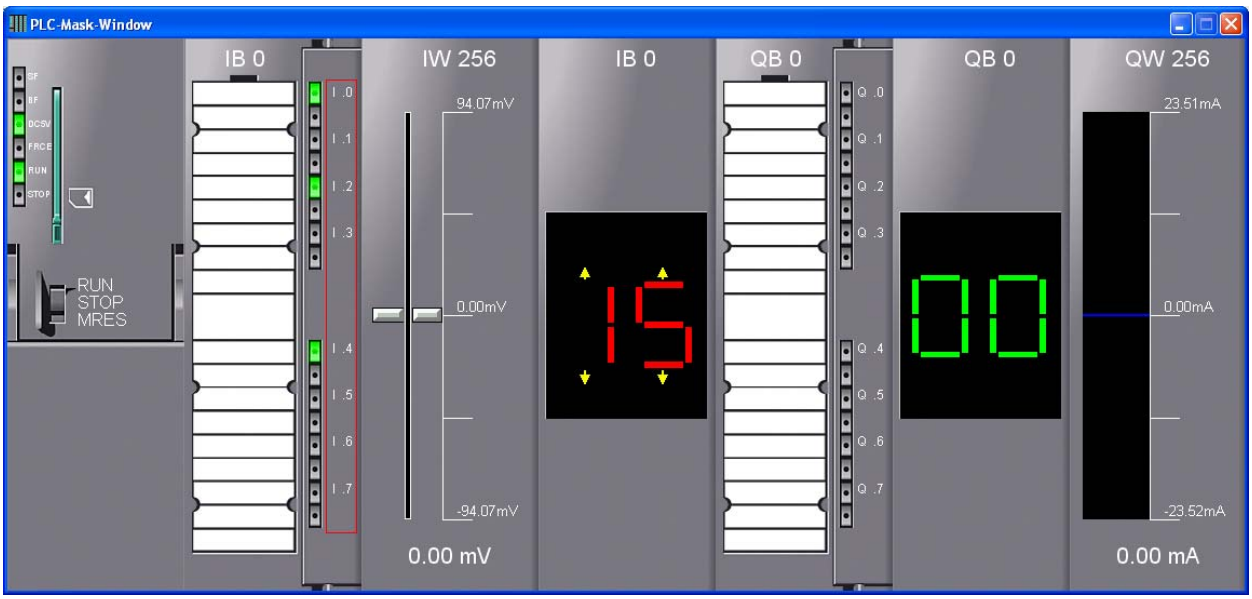


Fig.: PLC mask simulation with various modules

14.1 Turning the Software-PLC on

To activate the Software-PLC, the target must be set to "Target: Simulator":

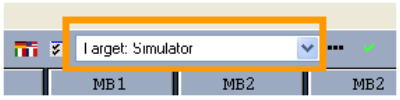


Fig.: simulator was activated

All commands now refer to the integrated Software-PLC. The following steps are necessary to simulate a program:

1. Transfer the blocks into the simulator, e.g. using *PLC->Synchronize*
2. Change the Software-PLC to "RUN" mode by means of menu item *PLC->Operating mode* or with the tab *"CPU Control Center"*
3. Now you can start the simulation:
   - Open OB1 and press the “glasses” icon in the mouse bar of the block editor.
   - Call the PROCESS-IMAGE-WINDOW using **View->Display process-image-window**
   - Display the PLC mask simulation using *View->PLC mask simulation*
   - Monitoring of variables using *View->Status variable*
4. If the simulator should not go to RUN mode or immediately go to STOP mode, you can use the status of the module (*PLC->Module status*) to localize the error.

Note:
Chapter 2 "Quick start" also introduced the simulator with a brief example.

14.2 Technical data of the Software-PLC

The technical specifications for the Software-PLC of WinPLC7 are as follows:

Process image of the inputs	2048 Byte	I0.0 to I2047.7
Process image of the outputs	2048 Byte	Q0.0 to Q2047.7
Bit memory	4096 Byte	M0.0 to M4095.7
Timers	512	T0 to T511
Counters	512	C0 to C511
Local data	1024 Byte	L0.0 to L1023.7
OBs	OB1, OB10-17, OB20-OB23, OB30-OB38,OB40, OB 41, OB100, OB101	
SFCs	SFC0,SFC1,SFC2,SFC3,SFC4,SFC20,SFC21,SFC22,SFC23,SFC24,SFC25,SFC26,SFC27,SFC28,SFC29,SFC30,SFC31,SFC32,SFC33,SFC34,SFC43,SFC46,SFC47,SFC64	
SFBs	-	
Range of numbers FC	0-2047	
Range of numbers FB	0-2047	
Range of numbers DB	1-2047	
Available working memory	262140 Byte	

The Software-PLC supports nearly all STEP®7 commands.
The following commands are not supported:
Commands with master control relay.

Following event-driven OBs are supported:

Time-of-day interrupts	OB 10 to OB 17
Cyclic interrupts	OB 30 to OB 38
Delay alarms	OB 20 to OB 23

The respective settings are available via menu item
Extras->Target properties: Simulator.

Note: This menu item is only visible if the target was set to "Simulator”.

14.3 Access the Software-PLC of WinPLC7 over TCP/IP

If the TCP/IP Interface is active, you can connect the Software-PLC of WinPLC7 via Ethernet.
You have local access (within the same PC) or via the local network using a separate PC.

You can also connect a OP/TP with an Ethernet interface to the Software-PLC of WinPLC7.

To activate it, choose the target "Simulator" and press the setting button:

Fig.: Open the settings of the Software-PLC

Then select "TCP/IP Settings" on the left side of the dialog :

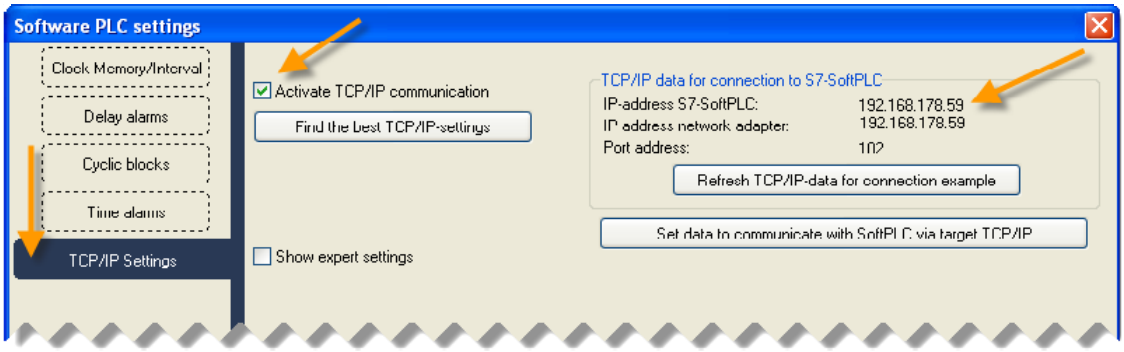


Fig.: The TCP/IP settings

Tick the checkbox "Activate TCP/IP communication". On the right you see the IP-Address of the Software-PLC. Use this IP-address to communicate with the Software-PLC. Press the OK button
Now, if the Windows firewall asks for permission, you must confirm this.
When the TCP/IP interface is active you see this label in the mouse bar:

Fig.: TCP/IP is active

You can now change the target to "**Target TCP/IP direct**" and enter the above IP address. Now you can access the Software-PLC over TCP/IP:

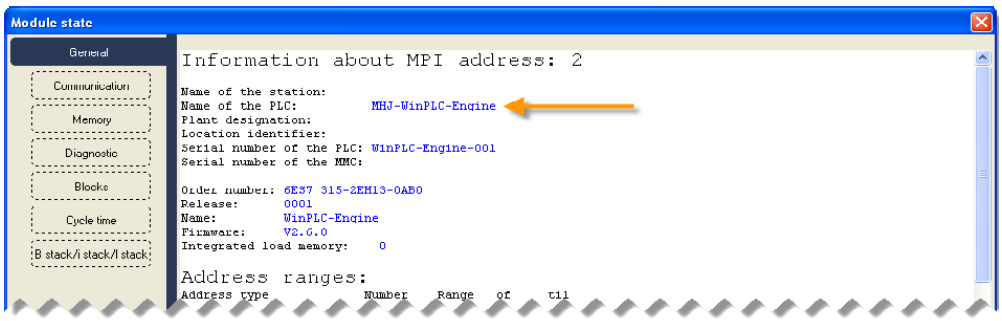


Fig.: Module state of the Software-PLC

15 Working efficiently with WinPLC7

This chapter shows the operating elements that enable you to work quickly with WinPLC7.

15.1 IntelliSense (Autocomplete)

The IntelliSense window helps you to quickly find the right operand while programming.

If the IntelliSense window appears, you can also simply enter the next characters. The list in the IntelliSense window is filtered automatically.

The IntelliSense feature is turned **on by default**. You can toggle it ON and OFF with the **hotkey CTRL+ALT+I**.
The current status is displayed in the status bar of the block editor:

Fig.: IntelliSense ON/OFF indication

The IntelliSense window will appear automatically when programming in STL or FBD/LAD when you type the first characters of an operand.

Examples of STL-progamming:

Enter into the STL editor: A "

Now the IntelliSense window appears with a list of symbols of the type "BOOL":

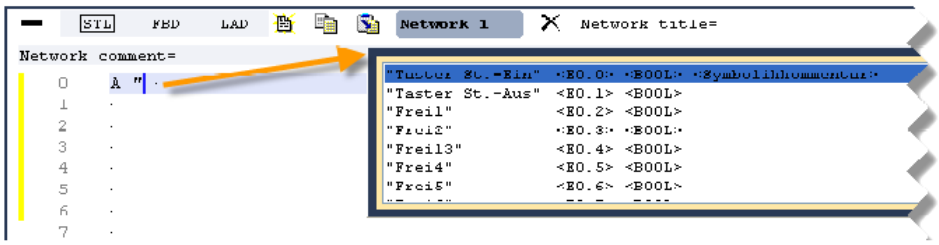


Fig.: IntelliSense ON/OFF indication

You can now filter the list by entering the next character of the symbol.
When you see a suitable symbol, press ENTER and the symbol appears in the code editor.

You can use the **IntelliSense window** with the following objects:

- Symbols
  Example:
  A " ---> all symbols of type BOOL appear
- Local block Parameters
  L # ---> all local parameter of the digital type appear
- Variables of a data block
  L DB1. ---> all BOOL DB1-variables appear
- Blocks of the project
  CALL FB ---> all FB blocks of the project appear
- System Function Blocks
  CALL SFB ---> all SFB blocks appear
  Note: after this, you may enter "send" for example to filter the list
- System Functions
  CALL SFC ---> all SFC blocks appear
  Note: after this, you may enter "move" for example to filter the list

Important note:
After displaying the IntelliSense window, you can filter the list by entering more characters.
This is particularly useful to quickly find the required SFC or SFB block.

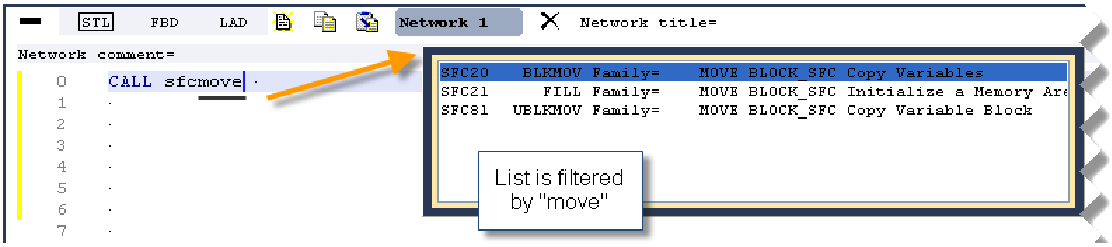


Fig.: Filtering the list

15.2 Automatic address entries in the symbolic editor, status variable window

You can use the following assistant to enter consecutive addresses into the symbolic editor or into the status variable window.

Example:
You want to enter bit memories M10.0 to M10.7 into the status variable window.
Enter M10.0, select the line and press the following button 7 times:

Bit memories M10.0 to M10.7 are now located in the window.
This also works for byte, word and double word addresses.

15.3 Important keyboard shortcuts (hotkeys)

15.4 Hotkeys related to a block editor

CTRL+S	Save the block
CTRL+E	Goto network
CTRL+T	Goto network title
CTRL+R	Insert a new network
CTRL+F7	Monitoring on/off toggle
CTRL+1	Convert all networks to LAD representation
CTRL+2	Convert all networks to STL representation
CTRL+3	Convert all networks to FDB representation
CTRL+J	Display list of symbols
ALT+BACK	Undo in the STL, FDB or LAD network
CTRL+A	Select everything in the STL editor
CTRL+F	Search
CTRL+ALT+K	STL: mark the selected STL lines as comments. You can press this hotkey again to toggle the selection.
CTRL+ALT+K	STL: convert comment lines back to STL lines

15.5 Global hotkeys

CTRL+O	Open/create a project
CTRL+N	Create a new block
CTRL+P	Open the print dialog
CTRL+L	Send active block
CTRL+U	Synchronize
CTRL+Q	Show tab "CPU Control Center"
CTRL+W	Show tab "Solution"
CTRL+Y	Show tab "Program structure"
CTRL+K	Show tab "Catalog"
ALT + PgUp	Open "Send blocks" dialog
ALT + PgDn	Open "Receive blocks" dialog
CTRL+I	Display "Operating mode" dialog
CTRL+D	Display "Module status" dialog
CTRL+ALT+T	Display the symbolic editor
CTRL+ALT+O	Display the global clipboard
CTRL+ALT+P	Display "Display protocol" dialog
CTRL+ALT+X	Display "Cross Reference" dialog
CTRL+K	Show/hide catalog
CTRL+ALT+F	Extended search
CTRL+AKT+U	Rewire
CTRL+I	IntelliSense ON/OFF
CTRL+ALT+M	Show tab sheet "Messages"
CTRL+ALT+T	Show tab sheet "Symbolic Editor"
CTRL+ALT+V	Show tab sheet "Status Variable"
CTRL+ALT+D	Show tab sheet "ToDo"
CTRL+ALT+H	Show tab sheet "Online PLC"
CTRL+ALT+Y	Show tab sheet "Address locations"
ALT+1	Activate tab sheet "Edit Project" (left side)
ALT+2	Activate tab sheet "Open Projects"
ALT+3	Activate tab sheet "Edit blocks"
ALT+4	Activate tab sheet "Tools"
STRG+TAB	Activate the next tab

15.6 Hotspots that are only accessible with the mouse

A Hotspot is an area of the WinPLC7 window that you can click with the mouse to start a particular operation:

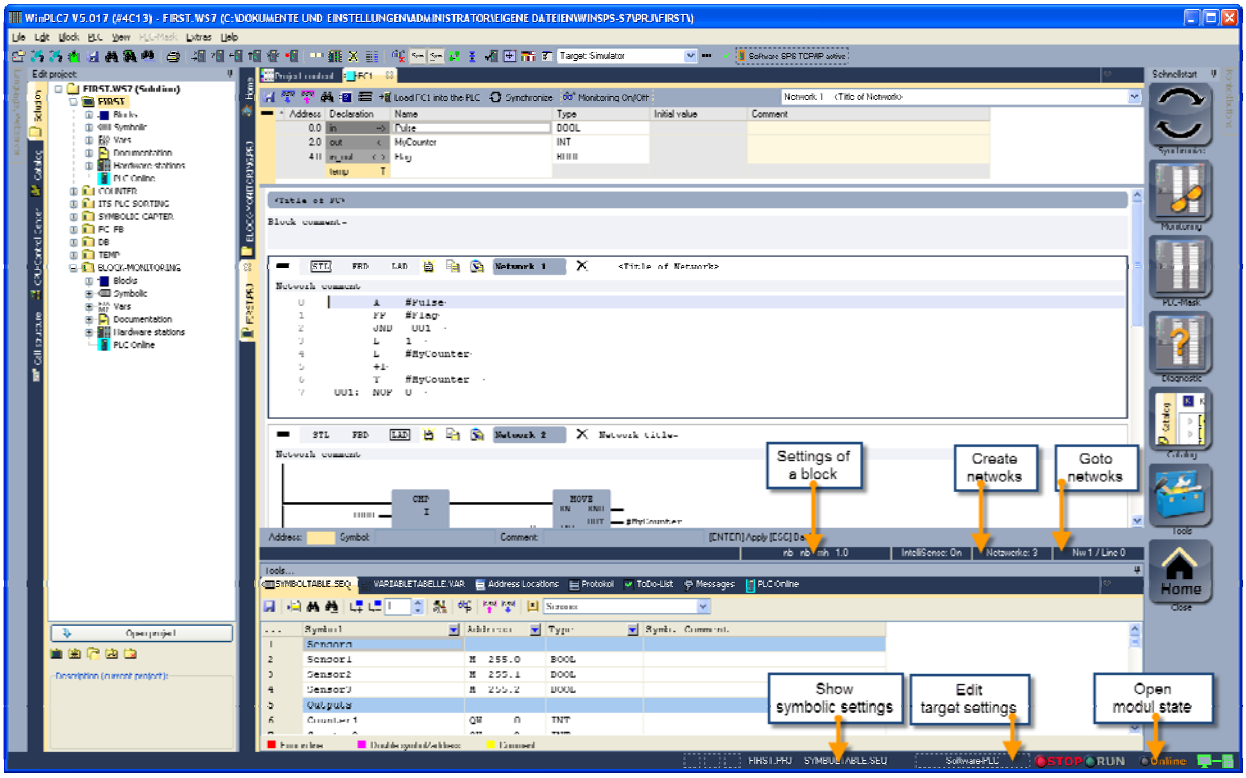


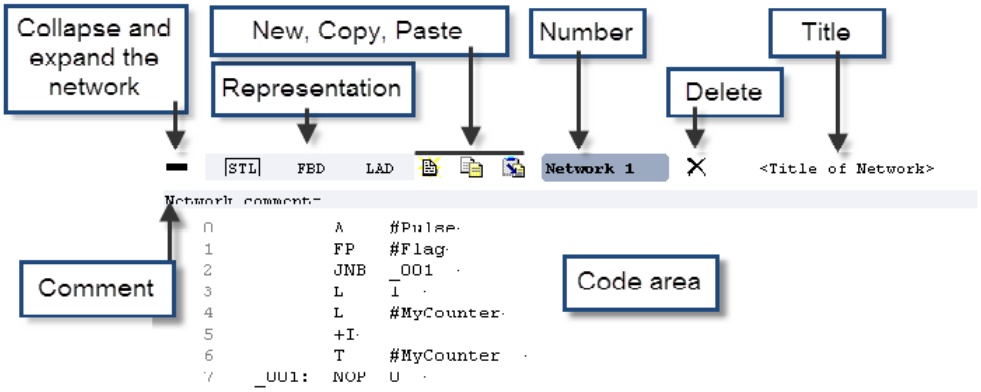
Fig.: Hotspots

16 STL programming

16.1 Using the STL editor

You can create any network of a block in STL, FBD or LAD. To program in STL, click on STL:

Every network has the following properties and capabilities:



To edit the code of a network, click on the code area. Now you can enter STEP®7 commands. It is important that a space is inserted between the operation and the address.
If you enter:

```
A M10.0
```

and confirm the line with the RETURN key, it will be formatted as follows:

```
A    M        10.0
```

You can enter a comment to the right of the operation:

```
A    M        10.0  //STL comment
```

You can also enter a comment on a separate line:

```
//Is safety chain present?
      A    I        10.0                //Safety door
      A    I        10.1                //Light barrier L2
      A    I        10.2                //Lock ok
      =    M        10.0                //Safety chain
```

Press [ESC] to quit from the STL editor.
The lines are numbered consecutively in the left column of the STL.

If you make a typing error in the STL editor, the line is not formatted when you press RETURN, and an error message with a red background appears in the status bar of WinPLC7:

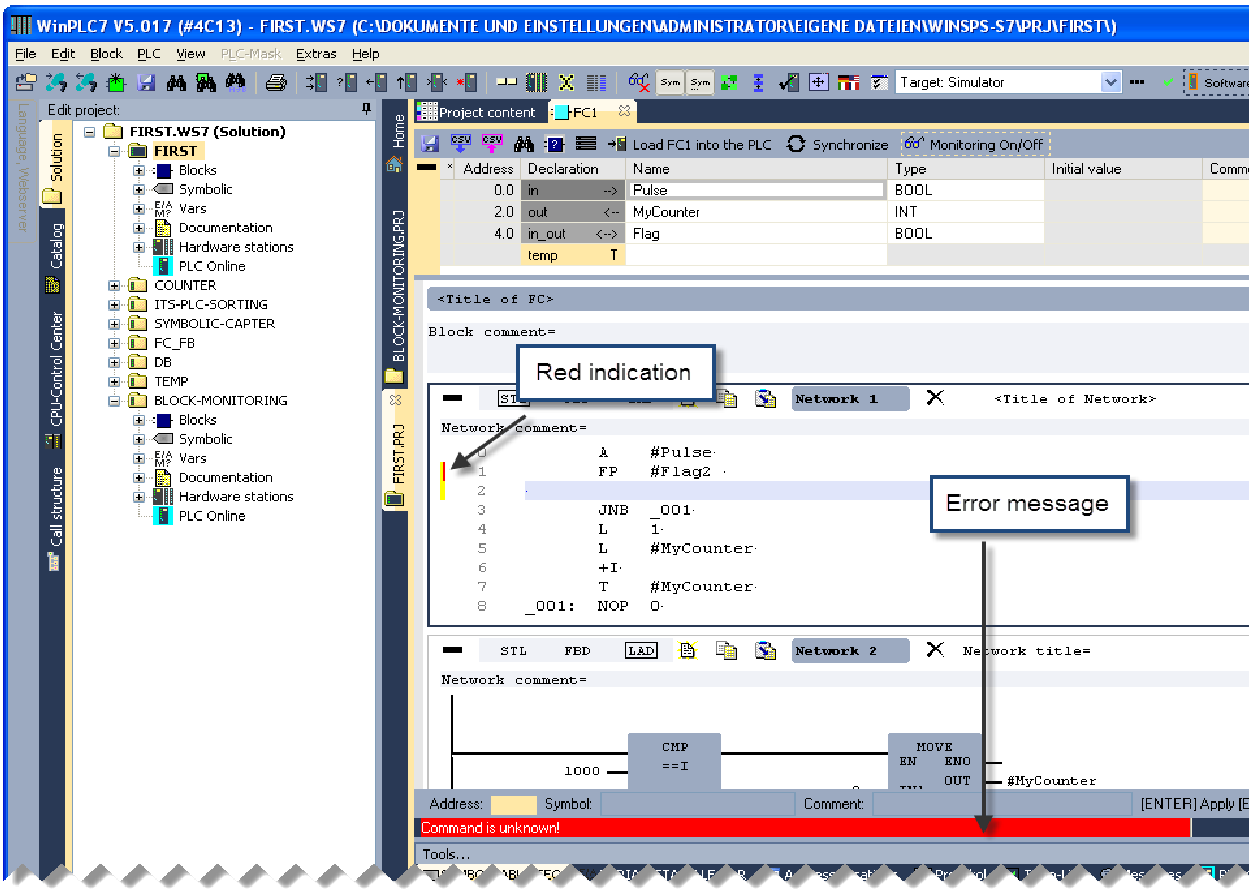


Fig.: error message displayed for a syntax error

If you save a block (CTRL+S) that contains errors, a red error message is displayed and the location of the error is displayed on a red background:

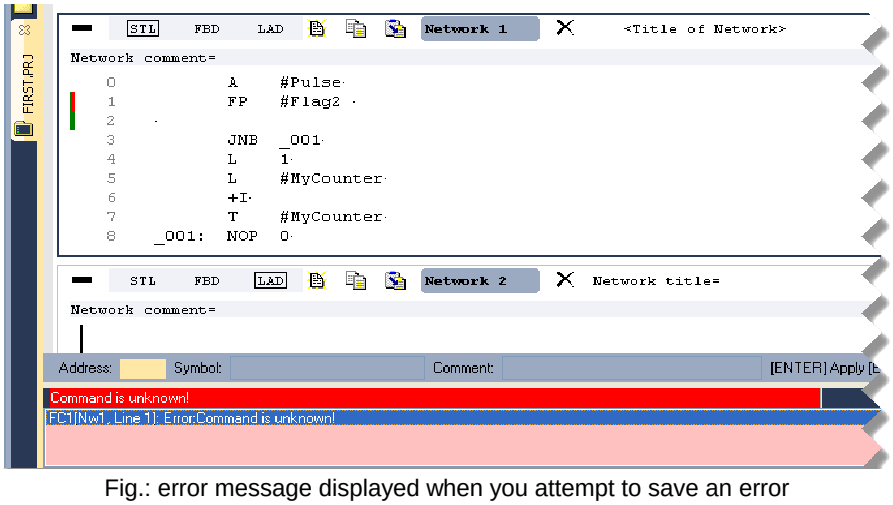


Fig.: error message displayed when you attempt to save an error

16.2 Hints regarding the STL editor

Commenting out lines:
To deactivate certain commands, you can comment out individual STL lines using a hotkey.
For this purpose, select the respective lines (press and hold the SHIFT key and press the up/down arrow) and press CTRL+ALT+K together.

To restore the lines, select the lines again and press CTRL+ALT+L.

Copying lines to the global clipboard:
Select the lines that you want to copy into the global clipboard. Continue with a right click of the mouse and select "Copy STL lines to global clipboard" in the context-sensitive menu.
First, you should open the global clipboard (View->Global clipboard) and determine which clipboard you want to used.
You can use this method to collect individual STL lines in a kind of library.

The context-sensitive menu of the STL network
The context-sensitive menu opens when you click the STL with the **right mouse key**:

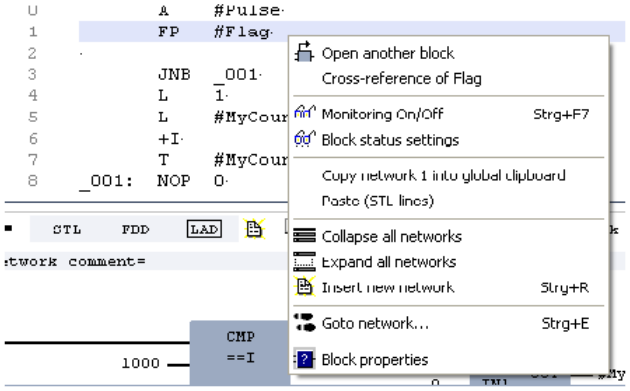


Fig.: the context-sensitive menu of the STL network

The following options are available:

- Open the other block
- Toggle monitoring on/off
- Monitor block with call path
- Cross-reference (local cross-reference) of the address in the STL line
- Insert new network
- Call "Goto" network dialog
- Call block properties (block flags)
- Collapse all networks
- Expand all networks

17 FBD programming

17.1 Principles

Select "FBD" with a click of the mouse (see figure) to switch the network to "FBD" representation:

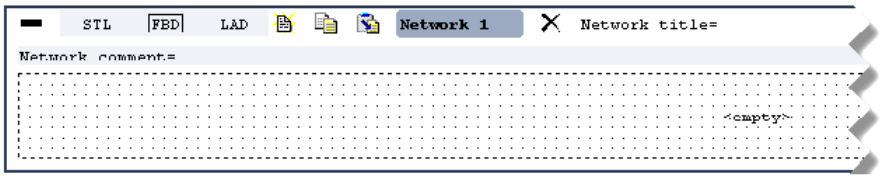


Fig.: empty FBD network

Click on the empty area if the code section is not dotted. Now the network is ready for programming.
To streamline your entries, the most important commands are available with the following shortcut keys.

Important keyboard shortcuts related to FBD programming:

F2	Insert an AND operation
F3	Insert an OR operation
F7	Insert an assignment
F9	Negation
F8	Add an input to an AND/OR/XOR
F11	Insert a branch
Del	Delete the selected object
ALT+Back	Cancel the last operations

Let us enter the following figure:

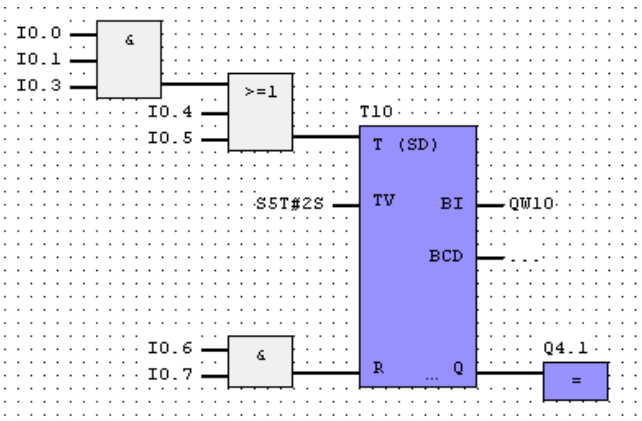
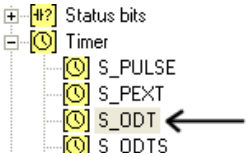
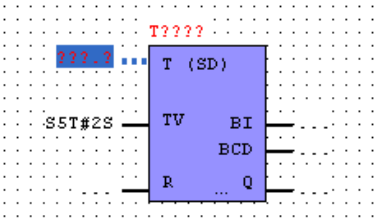


Fig.: FBD network

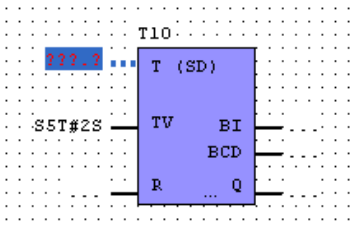
In this case it does not matter whether you begin with the AND gate on the left or with the timer.
Note:
If you make a mistake while you edit the FBD you can always press the ALT+BACK keys to cancel the mistake.



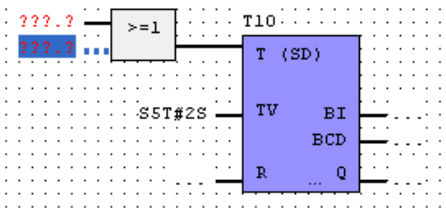
We will start with the timer. Select the entry "**S_ODT**" in "**Timer**" column of the catalogue on the right side of the screen



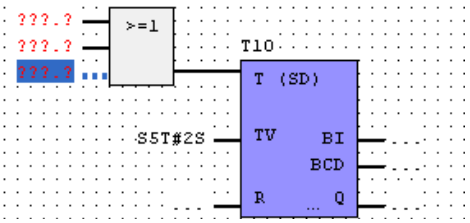
Double click this entry or drag and drop the entry onto the surface of the FBD.



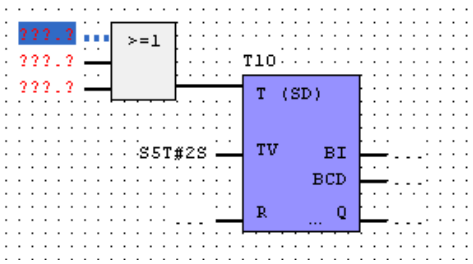
Now we replace the placeholder "T???" with T10



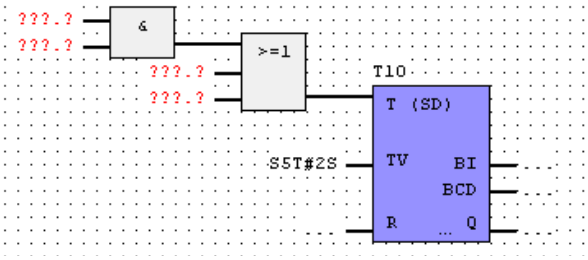
Because the input of the timer has already been selected, we now press the F3 key to insert an OR block in front of the input.



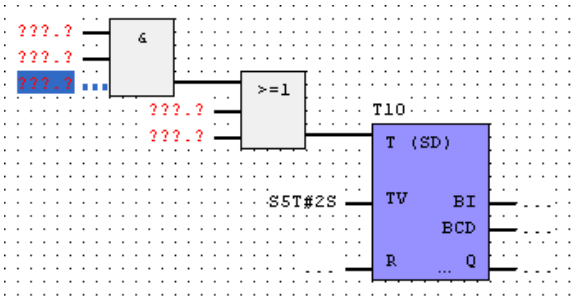
Press the F8 key once to add an input.



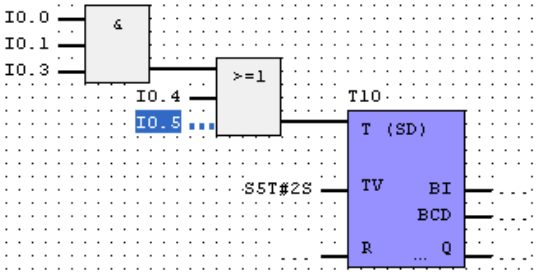
We now want to insert the AND gate in front of the OR gate. Click on the first input of the OR gate.



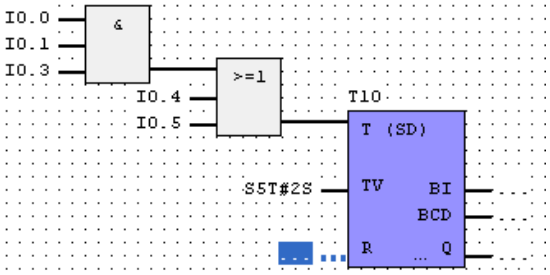
Press the F2 key to insert the AND gate.



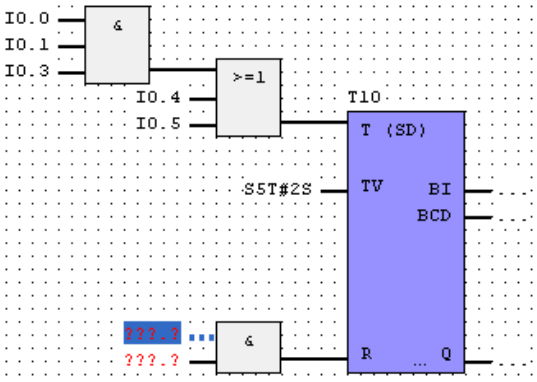
Press the F8 key once again to insert an additional input.



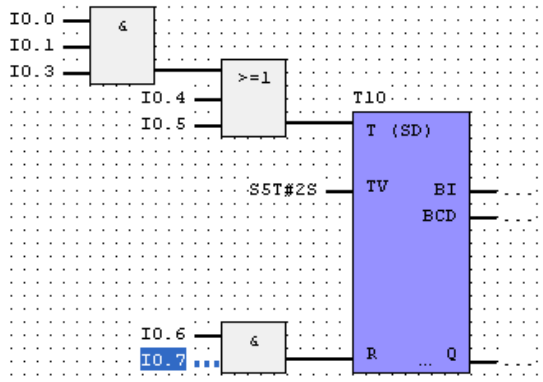
Now you can replace all the "???" placeholders with addresses. Select the placeholder and enter the address immediately. Confirm every entry with the RETURN key.
Note:
to edit an existing address, you may press the RETURN key first.



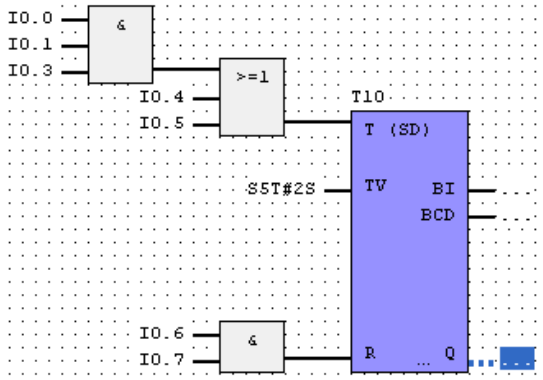
Click with the mouse on the "R" input of the timer.



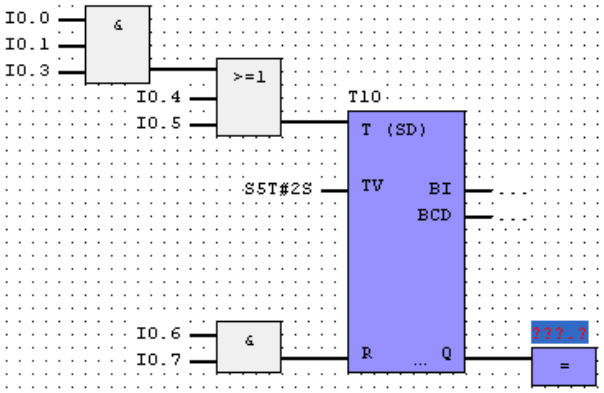
Press F2 to insert the AND gate.



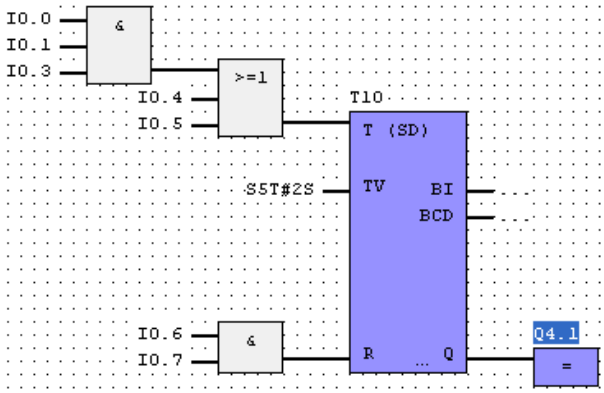
Replace the placeholders.



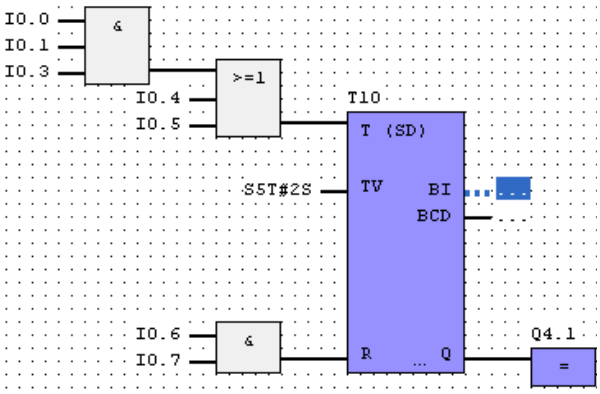
Select output (Q) of the timer.



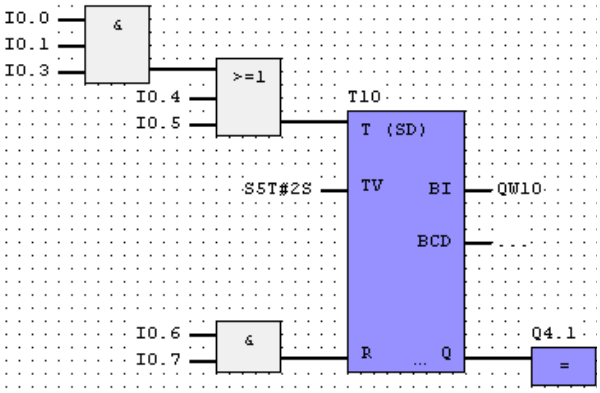
Press the F7 key to insert the assignment.



Replace the placeholder.



Select output “BI”.



and type "QW10".

This completes the network.

To modify an address, start with a click on the address and press the RETURN key. Now you can edit the address.

If do not press RETURN before you enter the new address, the old one is overwritten.

The **Delete key** deletes an address or an entire gate.
You can cancel this operation by pressing **ALT+BACK**.

17.2 Hints about FBD programming

The network that you have created above employs a timer. Changing the type of timer in the FBD is very difficult.

However, in WinPLC7 you can change the type of timer with a simple double click. This displays the dialog "Change the box…:

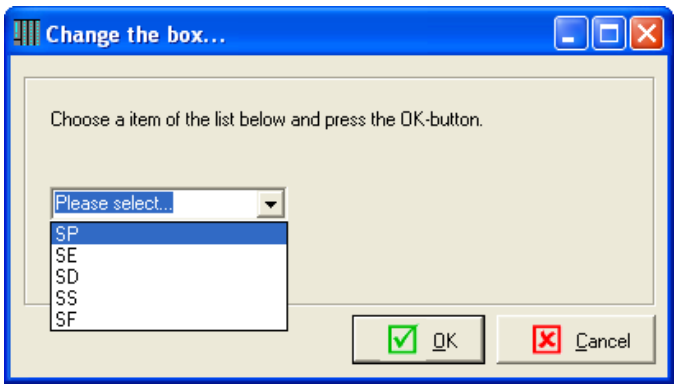


Fig.: change the type of timer

Select the desired timer and press OK. The type of the timer is changed in the FBD.

You can change the properties of the following FBD objects with a double click:

- Timer
- Memory: RS gates, SR gates
- Comparators
- Converters
- Counters (CU or CD)
  This is not possible when the entire counter has been assigned
- Fixed point arithmetic
- Floating point arithmetic
- Shift/rotate
- Word logic blocks

An alternative would be to change to "STL" representation and to insert the required modifications there.
You should always keep this possibility in mind, because it may be quicker and simpler to make changes in the STL. However, you should avoid deleting placeholders (NOP 0) in the STL, since the resulting STL can no longer be represented in FBD/LAD mode.

18 LAD programming

18.1 Principles

The LAD representation of the FBD configuration that we have created in the previous chapter looks as follows:

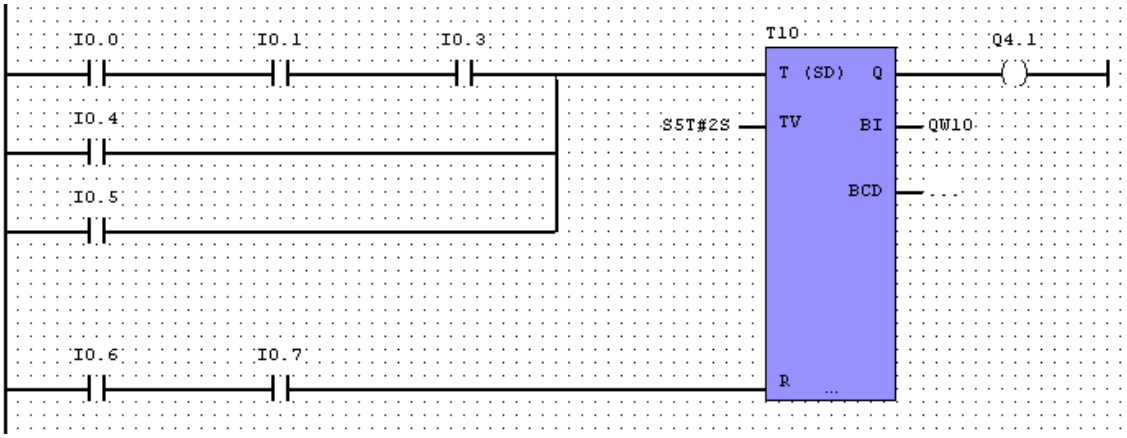


Fig.: network in LAD representation.

We will now continue to create this network. You can cancel the most recent change to the LAD using the keyboard shortcut ALT+Back (ALT+BACKSPACE).

The most important keyboard shortcuts with regard to LAD programming are:

F2	Insert normally open contact
F3	Insert normally closed contact
F7	Insert a coil
F8	Open branch
F9	Close branch
Del	Delete the selected object
ALT+Back	Cancel the last operations

We will start with the timer. However, you could also start with input I0.0.

Switch the network to LAD representation:

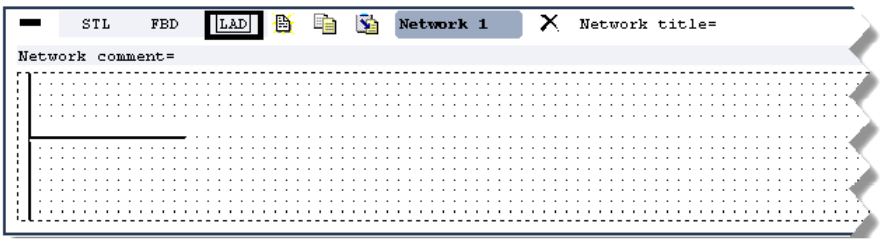
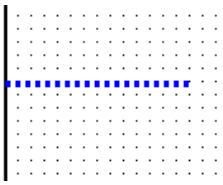
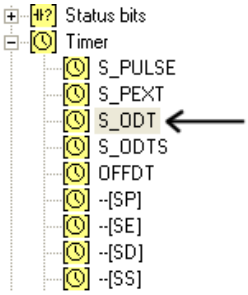


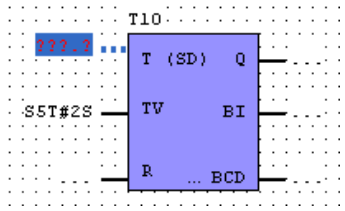
Fig.: Empty LAD network



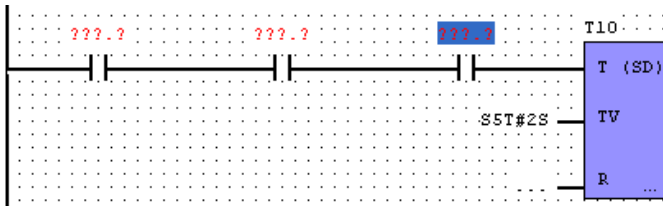
Click on the horizontal line in the empty LAD.



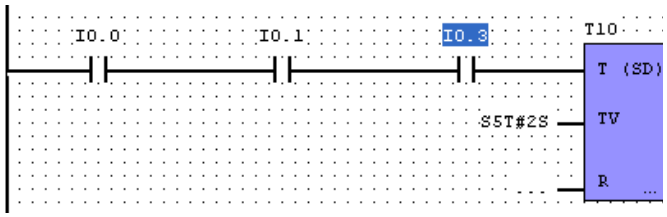
In the catalog (on the right side of the screen) select entry "**S_EVERTZ (SE)**" in the column "**Timers**". Double click this entry or drag and drop the entry onto the horizontal line in the empty LAD.



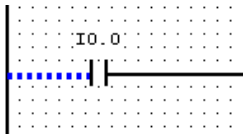
The timer is inserted. Click on placeholder "T???" and enter: T10. You can also press RETURN before you change the entry to "T10". Select the input of the timer with a click of the mouse.



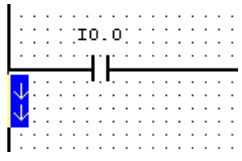
Press the "F2" key 3 times to insert three normally open contacts.



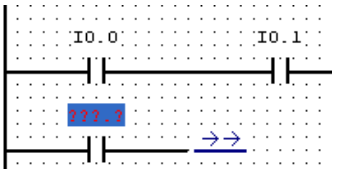
Label the switches with the appropriate addresses.



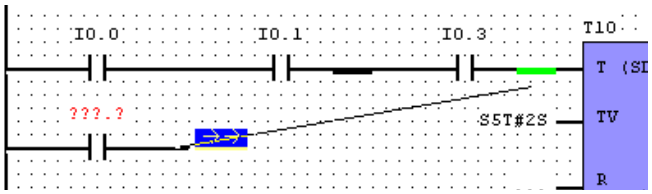
We will now insert the parallel connections for inputs I0.0, I0.1 and I0.3. Click on the left side of switch I0.0 with the mouse.



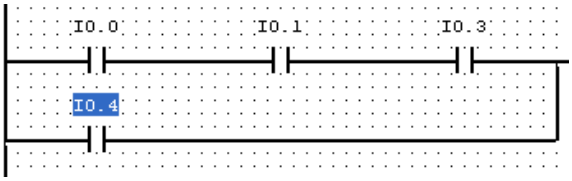
Then press the F8 key for "Open branch".



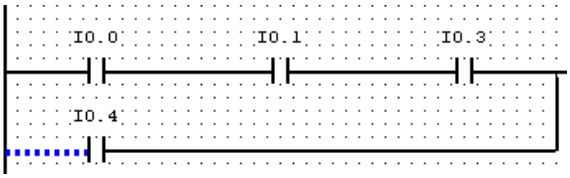
Press the F2 key to insert an additional switch.



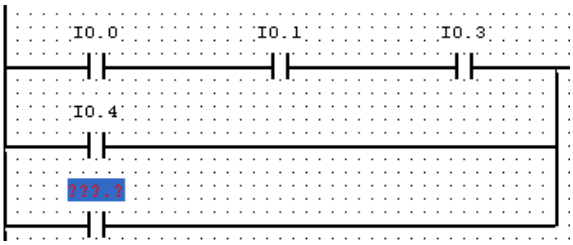
Now click and hold the symbol with two arrows with the mouse (hold the mouse key) and drag it to the right side of input I0.3.



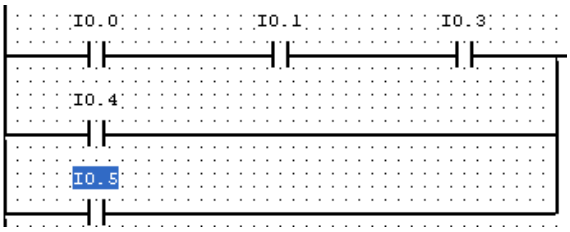
Replace the placeholder with I0.4; at this point you have completed the parallel connection.



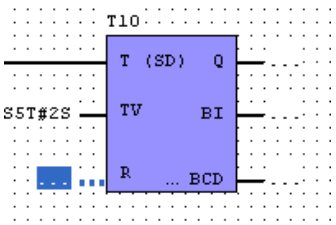
Select the left side of the new switch. Press the following in the indicated sequence:
F8 (open branch), F2 (new normally open contact) and F9 (close branch)



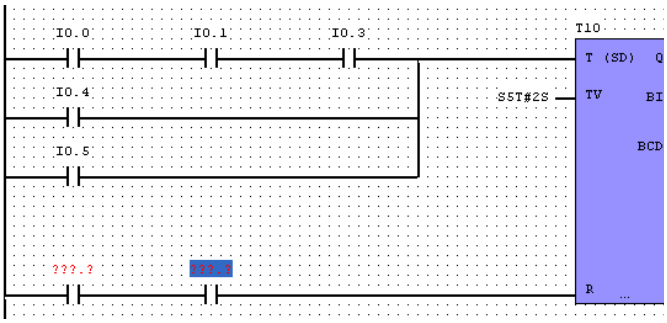
Replace placeholder ???.? with I0.5; this completes the second parallel connection.



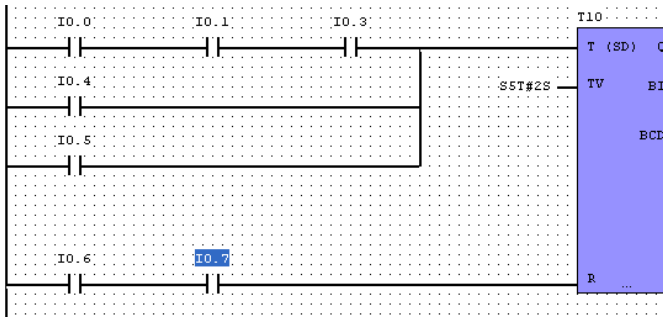
Label the new switch I0.5.



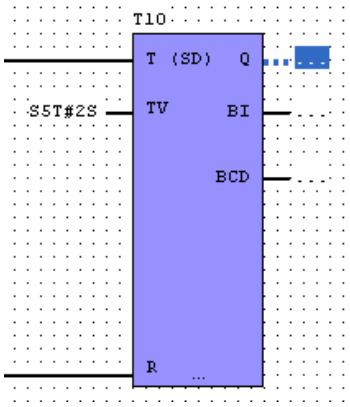
We continue with the two normally open contacts at the reset input of the timer. Click on this input with the mouse to select the input.



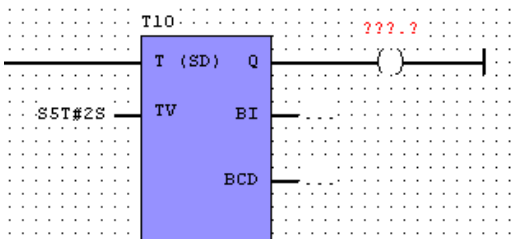
Press the "F2" key twice (insert normally open contacts).



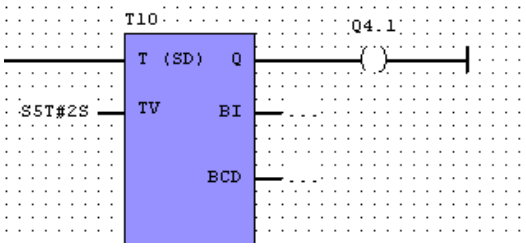
Label the new normally open contacts as I0.6 and I0.7.



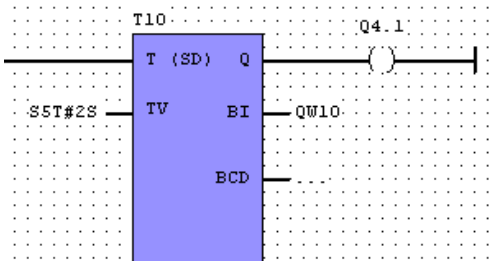
Select output (Q) of the timer.



Press the "F7" key to "Insert coil".



Label the coil as A4.1



Change the DUAL output to "AW10".

This completes the LAD network:

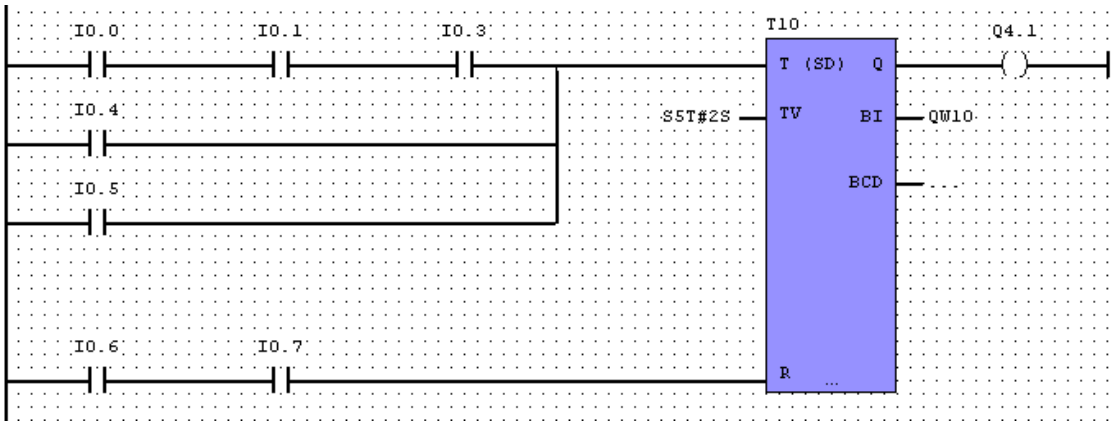


Fig.: the completed LAD network

18.2 Hints about LAD programming

Changing the properties of a LAD object with a double click

As previously described for FBD programming, it is also possible to change the properties of a LAD object by means of a dialog that is opened with a double click. You can use this to change an SE timer to an SI timer.

This function is supported by the following LAD objects:

- Timer, memory: RS gates, SR gates
- Comparator, converter
- Counters (CU OR CD)
  This is not possible when the entire counter has been assigned
- Fixed point arithmetic, floating point arithmetic
- Shift/rotate, Word logic blocks

Cancel recent changes

You can cancel any change to the LAD by means of keyboard shortcut ALT+Back.

Extending the parallel connection with a normally open contact

To insert a normally open contact into the series between M10.0 / M10.1 and M10.2/M10.3 proceed as follows:

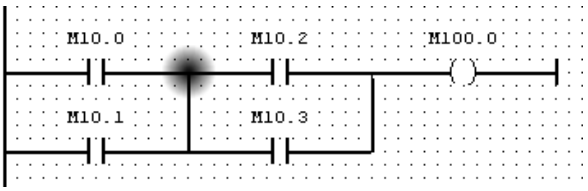


Fig.: a normally open contact must be inserted at the selected location.

Select the left side of bit memory M10.2. **Hold the CTRL key and click on the normally open contact with the mouse**:

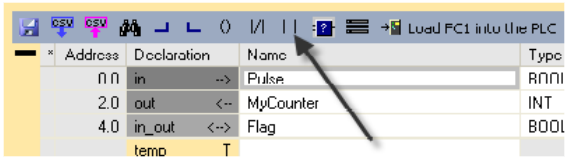


Fig.: inserting a normally open contact using the mouse symbol

A new switch is inserted into the series circuit:

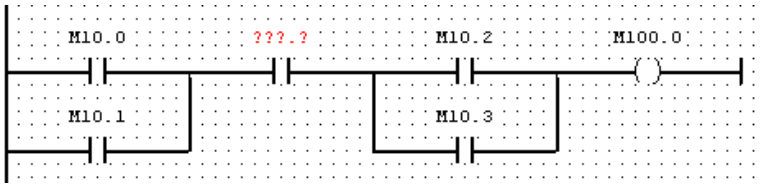


Fig.: the inserted normally open contact

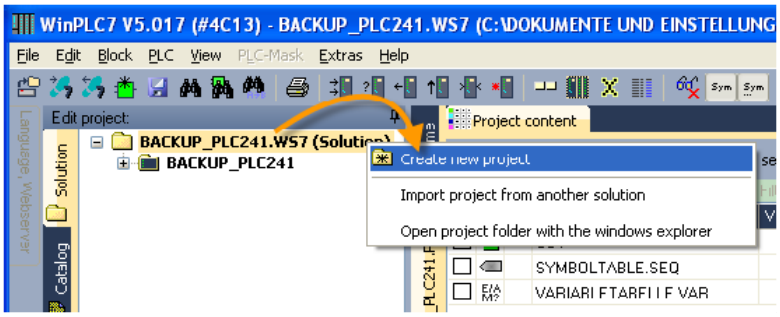
19 PLC program backup and restore

19.1 Backup the blocks in the PLC

Decide whether you want to create a new solution or use the current solution.

Proceed as follows to create a backup of the blocks in the S7 PLC:

1. Create a project in the solution tree (with a right mouse click)



2. Establish a connection with the S7 PLC (select and activate target). To test the connection, execute "Module State"- **Hotkey CTRL+D**
3. Load all the blocks from the PLC using PLC->Receive blocks

The blocks are now located in the current project. The blocks will be visible in the tab "Project content".

This procedure has created a backup of all the blocks in the PLC (including the hardware configuration).

19.2 Restore (recover) the blocks in the PLC

If you want to restore a PLC program from a backup, you must transfer both, the **user blocks** as well as the **system data blocks** (hardware configuration).

Proceed as follows:

1. Select the project where the blocks are located.
2. Establish a connection with the PLC (select and activate target)
3. Change the PLC to stop mode using menu item ***PLC->Operating mode*.**
4. Delete all the blocks in the PLC via ***PLC->Delete blocks***
5. Compress the memory contents in the PLC via menu item ***PLC->Compress***
6. Transfer all user blocks via menu item ***PLC->Send all blocks***
7. Transfer all system data blocks via menu item ***PLC->Restore hardware configuration***
   If you have a hardware station in the project, you can also restore the hardware configuration with the hardware configurator. In this case, double click to the hardware station in the project tree to start the hardware configurator.
8. Change the PLC to RUN mode using menu item ***PLC->Operating mode*.**

Important note, if you don't have a hardware station in the project tree:

To change the hardware configuration before you transmit it to the PLC, proceed as follows:

1. Start the hardware configurator (create a new hardware station in the project tree)
2. Select menu item **File/Project->Load station from WinPLC7 project**

The configuration is loaded and can be modified.
However, after you have made changes you must transfer the hardware configuration using ***Online->Send configuration*** and not via WinPLC7.

19.3 Backup of a project

You can duplicate a project in the project tree. Execute a click with the right mouse key and select menu item "Duplicate project".

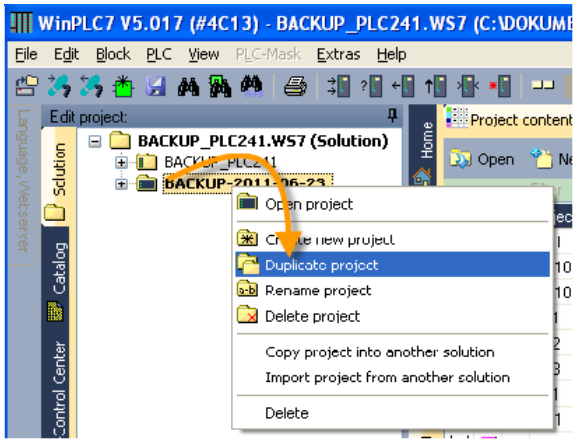


Fig.: duplicate project

To save a project to a CD-ROM, you should create a ZIP-file of the project and write it to the CD-ROM.
You can save a project in a ZIP file using ***File->Export->Save project as a ZIP file***.

19.4 Restoring the project

You can import a project from a ZIP file as follows:

1. Open an existing solution or create a new solution
2. Select menu item ***File->Import->ZIP file (STEP7-projekt or WinPLC7-project)***
   In the dialog, select the ZIP file by means of the button "**...**". The list now includes the projects in the ZIP file.
3. Select all the projects that you want to import and press the "**Start**" button.
4. The selected projects are inserted into the current solution.

20 The sequencer wizard

Based on your definitions, the sequencer wizard generates a complete PLC program that you can use as the basic framework for a new PLC program.

You can complete the PLC program much quicker by specifying different blocks as well as the symbol table.

20.1 Principles

Options offered by the sequencer wizard:

- You can create sequencers in a FC or FB that you can edit in STL, FBD or LAD.
- You can use flags or data bits from a data block as step addresses. If you define flags, then the sequencer is generated in a FC. If you define data bits, then the sequencer is generated in a FB with instance data blocks.
- You can insert an EMERGENCY-OFF switch
- You can prepare a CONTROL SYSTEM ON/ CONTROL SYSTEM OFF mechanism.
- You can define the conditions for every sequencer step as STL-lines.
- You can associate every step with a network title and network comment.
- Automatic creation of a symbolic file

Requirements

To be able to create a sequencer, the current project must be empty, i.e. no blocks may exist in the current project.

Procedure

1. **"Settings" tab:**
   Specify whether you want to use flags or a data blocks as the memory for the steps (step address).
   If required, tick the check box "Use emergency off switch"
   If required, tick the check box "Use ‘Control system ON/OFF’"
2. **"Sequencer" tab:**
   Generate the necessary steps for your system by means of the button "New step".
   In the "Settings of the step” area you define the title of the network, the network comment and the conditions for the step.
3. **"Symbols" tab:**
   Press the button "Create symbol table". A list with all the addresses used in the sequencer is displayed. Change symbol name and the symbol comment as required.
4. **"Create sequencer" tab:**
   Select the desired representation (STL, FBD, LAD).
   Note: You may change the representation in the block. Press the button "Create sequencer" to create the necessary blocks. As a result, different blocks will be created.

What remains to be done?

The following tasks must still be completed when the blocks have been created:

- you must reset the last step.
- the step addresses (flags or data bits) must be assigned to the outputs.

20.2 Example

Create the PLC program for the following technology scheme by means of the sequencer wizard:

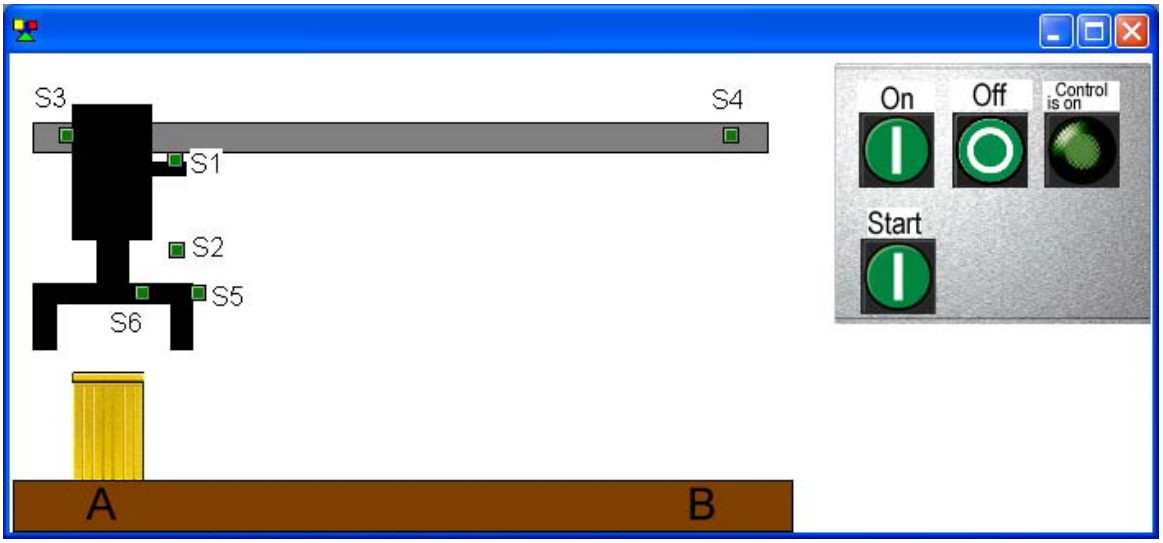


Fig.: system with gripper

A gripper arm must carry a box from location A to location B. To prepare the system, start the "Control-IsOn" mechanism in the PLC program by pressing the "ON" button. The lamp "Control is on" is turned on. The following process is started when you press the "Start" button:

1. Start pressed, the gripper is lowered
2. Close gripper
3. Raise gripper
4. Move gripper to the right
5. Lower the gripper
6. Open gripper (the box is dropped)
7. Raise gripper
8. Move gripper to the left starting position

These are also the steps in the sequencer.
That means 8 steps.

The following assignment table applies:

Address	Assigned to
I0.0	Control system **ON** button (NO contact)
I0.1	Control system **OFF** button (NC contact)
I0.2	Start button (NO contact)
I0.6	Top stop position (S1)
I0.7	Bottom stop position (S2)
I1.0	Left stop position (S3)
I1.1	Right stop position (S4)
I1.2	Gripper open (S5)
I1.3	Gripper closed (S6)
Q4.0	Move gripper to the right
Q4.1	Move gripper to the left
Q4.2	Move gripper to the bottom
Q4.3	Move gripper to the top
Q4.4	Open gripper
Q4.5	Close gripper
Q4.6	"Control is on" lamp

You should first consider the conditions that apply to each step:

Step No.	Step description	Condition
Step 1	Gripper moving down	Start button: **A I 0.2**
Step 2	Close gripper	Gripper at bottom position: **A I 0.7**
Step 3	Gripper moving up	Gripper closed: **A I 1.3**
Step 4	Gripper moving right	Gripper at top position: **A I 0.6**
Step 5	Gripper moving down	Gripper at the right limit: **A I 1.1**
Step 6	Open gripper	Gripper at lower stop position: **A I 0.7**
Step 7	Gripper moving up	Gripper is open: **A I 1.2**
Step 8	Move gripper to the left home position	Gripper at top stop position: **A I 0.6**

Fig.: table of conditions

Now that all the required information is available you can enter this information into the sequencer wizard.

Create a new project or a new solution. Then select menu item **Extras->Sequencer Wizard**

Remove the tick mark from "Use emergency off switch" in the **"Settings" tab**, since the example does not make use of an emergency off switch. Enter the addresses for Control system ON/OFF as shown in the figure:

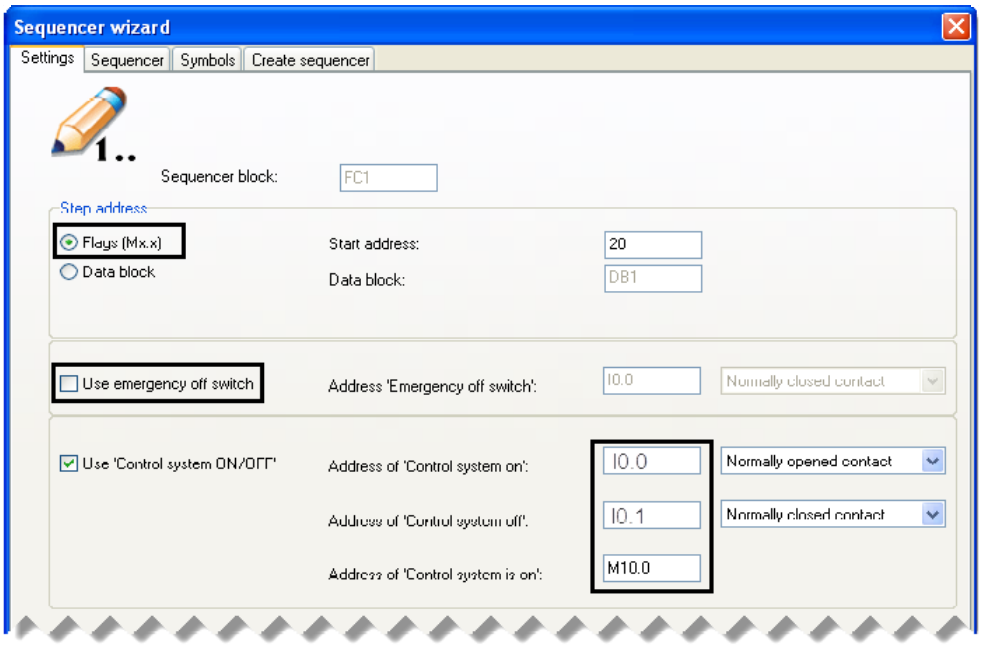


Fig.: settings tab

Press **"Continue"** button to select the next tab.
A total of **8 steps** must be created in the "Sequencer" tab using button "**New step**". Then select each step and define the settings of the step. Each step must be associated with a condition. The STL condition is available from the "table of conditions" shown on the previous page. You can also enter a meaningful text for the title of the network and the network comment.

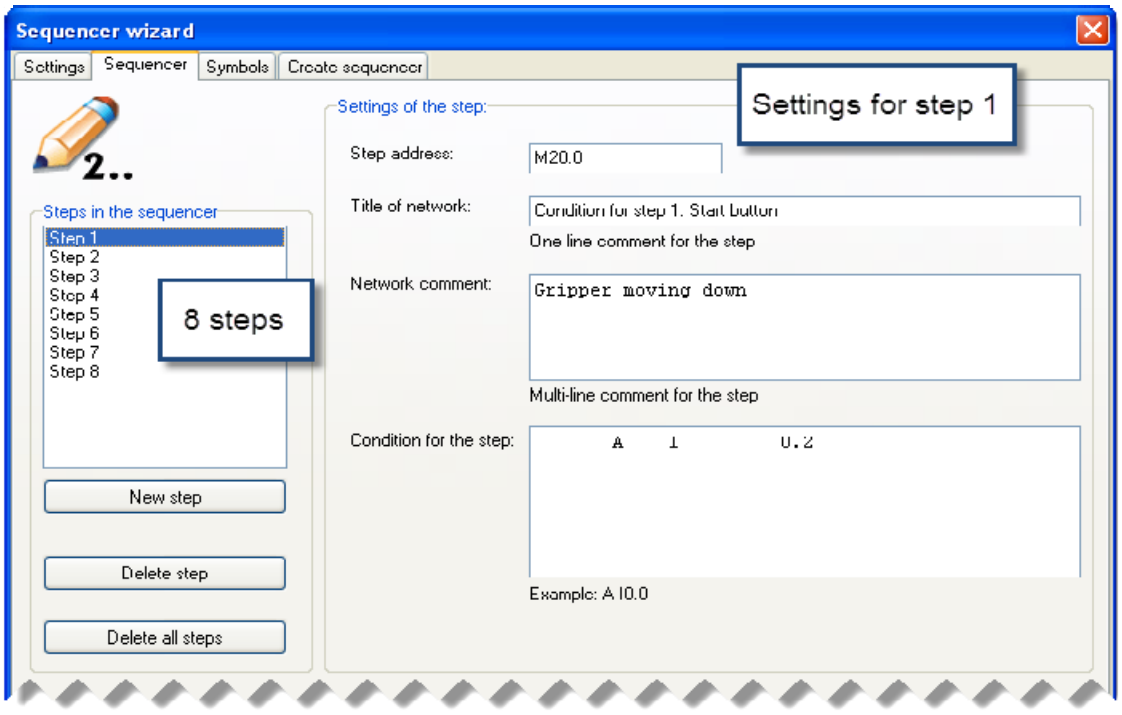


Fig.: sequencer tab

You can create symbols in the **"Symbols" tab** by means of the button "**Create symbol table**". You can modify the symbol as well as the symbol comment as required.

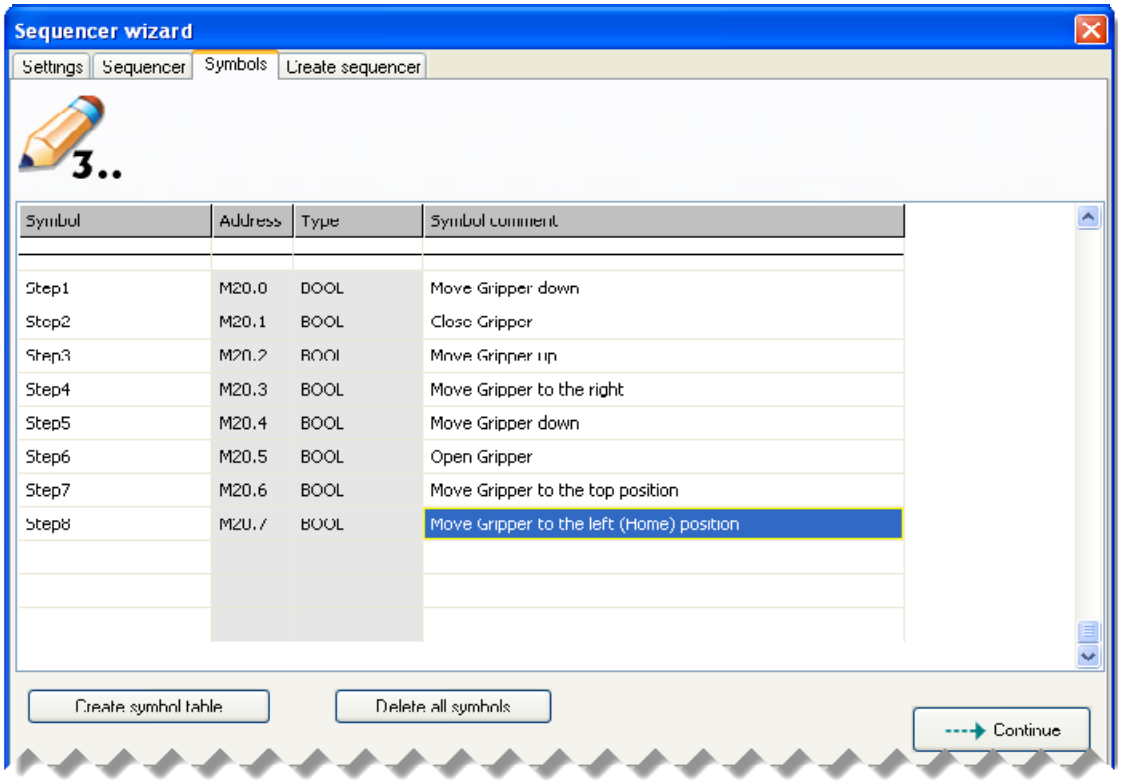


Fig.: "Symbols" tab

You can select the type of representation that will be used to create the blocks in the **"Create sequencer"** tab. When you press the "**Create sequencer**" button the blocks are transferred to the current project:

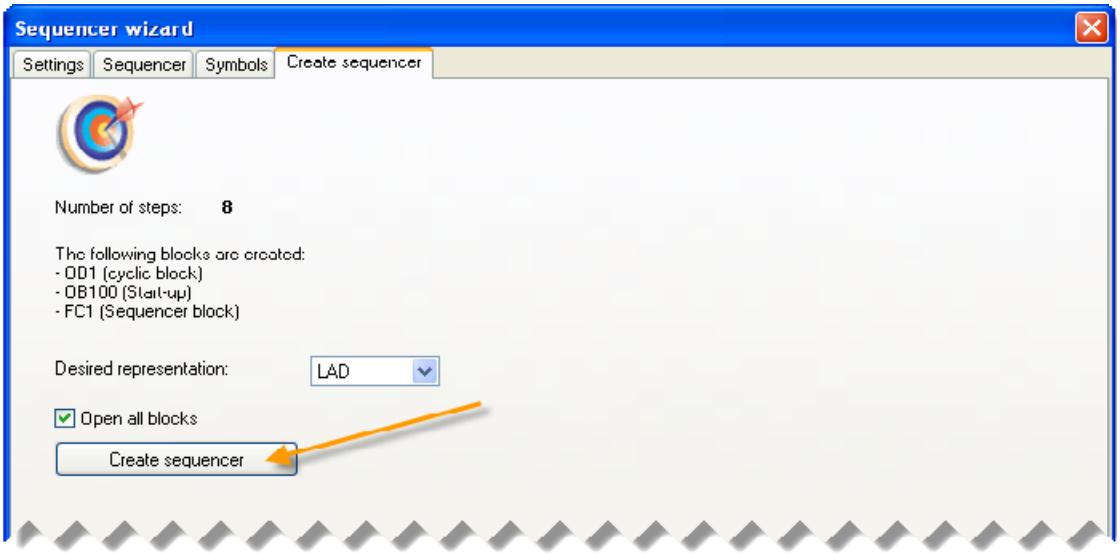


Fig.: “Create sequencer” tab

The following blocks were created (see Project Content Window):

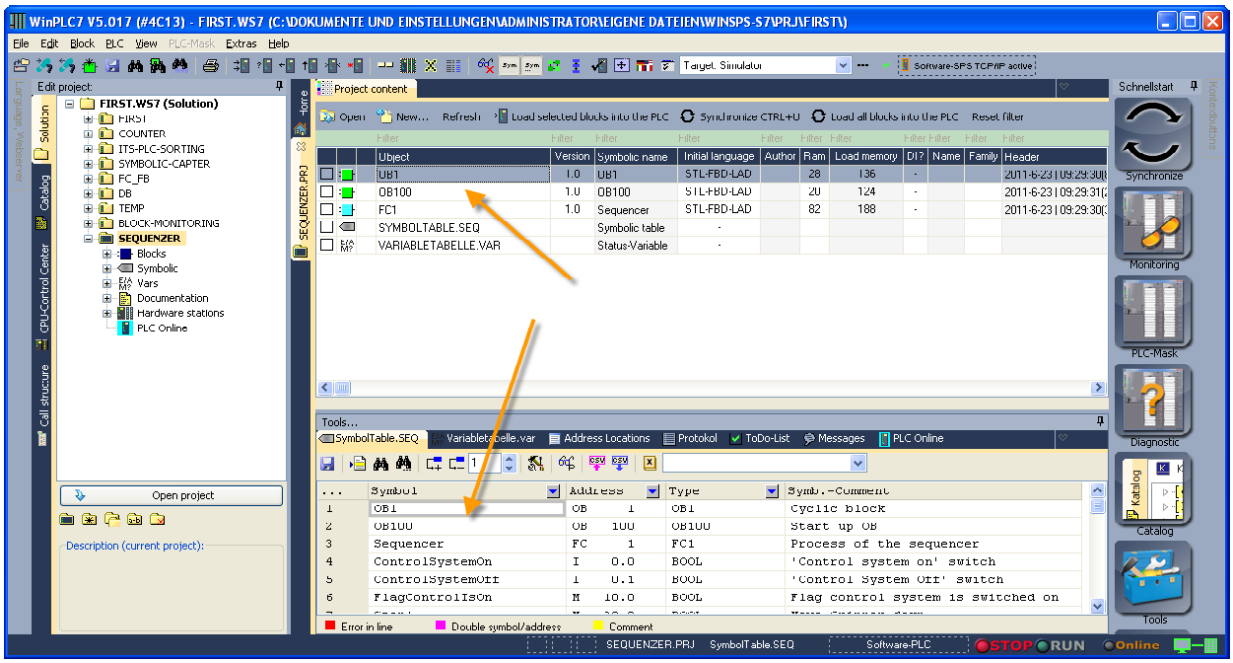


Fig.: Project Content Window and Symbolic Table after creation

Adding the symbols for the inputs into the symbolic table:

SymbolTable.SEQ | Variabletabelle.var | Address Locations | Protokol | ToDo-List | Messages | PLC Online

...	Symbol	Address	Type	Symb.-Comment
1	OB1	OB 1	OB1	Cyclic block
2	OB100	OB 100	OB100	Start up OB
3	Sequencer	FC 1	FC1	Process of the sequencer
4	ControlSystemOn	I 0.0	BOOL	'Control system on' switch
5	ControlSystemOff	I 0.1	BOOL	'Control System Off' switch
6	FlagControlIsOn	M 10.0	BOOL	Flag control system is switched on
7	Step1	M 20.0	BOOL	Move Gripper down
8	Step2	M 20.1	BOOL	Close Gripper
9	Step3	M 20.2	BOOL	Move Gripper up
10	Step4	M 20.3	BOOL	Move Gripper to the right
11	Step5	M 20.4	BOOL	Move Gripper down
12	Step6	M 20.5	BOOL	Open Gripper
13	Step7	M 20.6	BOOL	Move Gripper to the top position
14	Step8	M 20.7	BOOL	Move Gripper to the left (Home) posit
15	StartButton	I 0.2	BOOL	
16	TopStop	I 0.6	BOOL	
17	BottomStop	I 0.7	BOOL	
18	LeftStop	I 1.0	BOOL	
19	RightStop	I 1.1	BOOL	
20	GripperOpen	I 1.2	BOOL	
21	GripperClosed	I 1.3	BOOL	

OB1:

In "Network 1" of OB1 the "ControlSystem" is turned on and in "Network 2" a call is issued to block "FC1".

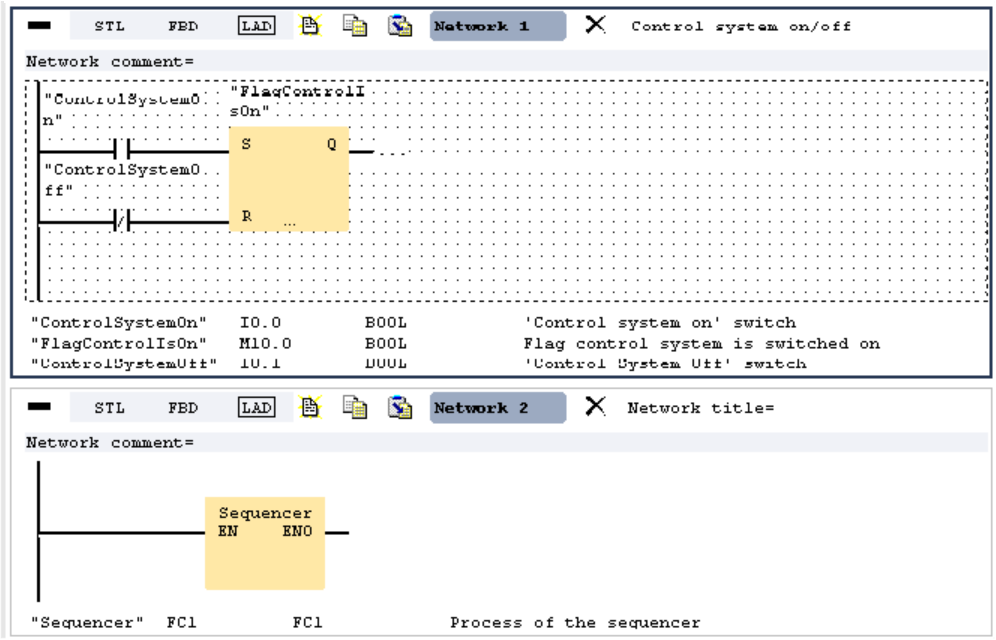


Fig.: OB1

OB100:

The sequencer is reset in the restart-OB:

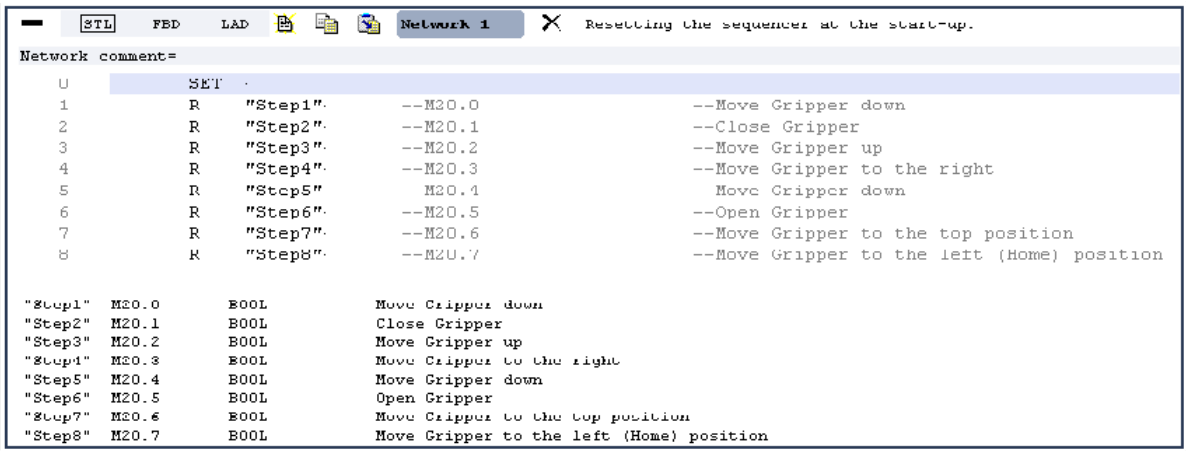
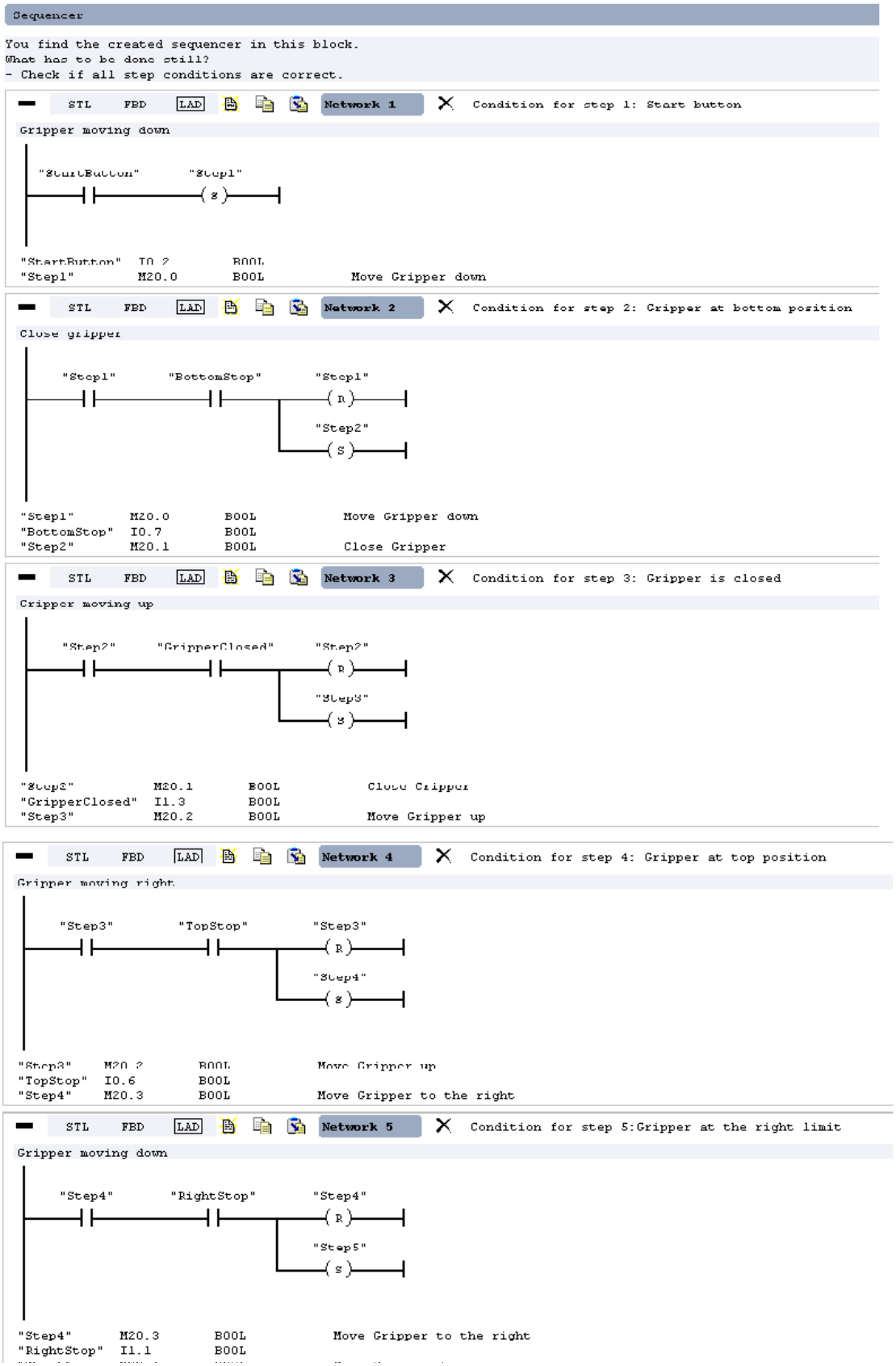


Fig.: OB100

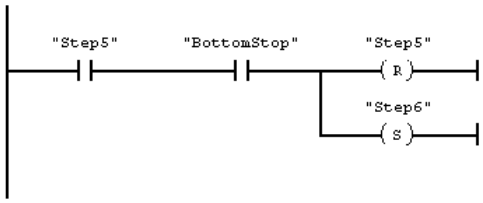
FC1:

The actual sequencer is contained in FC1.
Only Network 9 and Network 11 were changed.
All the other networks are used directly as they were received from the sequencer wizard.



STL FBD LAD **Network 6** Condition for step 6: Gripper at the lower stop position

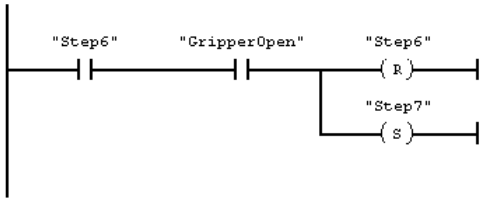
Now open the gripper and drop the box



"Step5"	M20.4	BOOL	Move Gripper down
"BottomStop"	I0.7	BOOL	
"Step6"	M20.5	BOOL	Open Gripper

STL FBD LAD **Network 7** Condition for step 7: Gripper is open

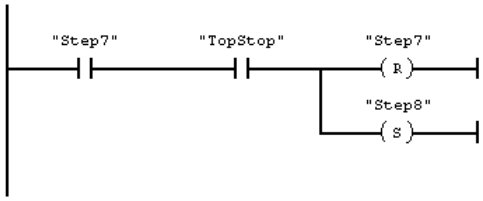
Move now the gripper to top position



"Step6"	M20.5	BOOL	Open Gripper
"GripperOpen"	I1.2	BOOL	
"Step7"	M20.6	BOOL	Move Gripper to the top position

STL FBD LAD **Network 8** Condition for step 8: Gripper is on top position

Now moving back to start (home) position



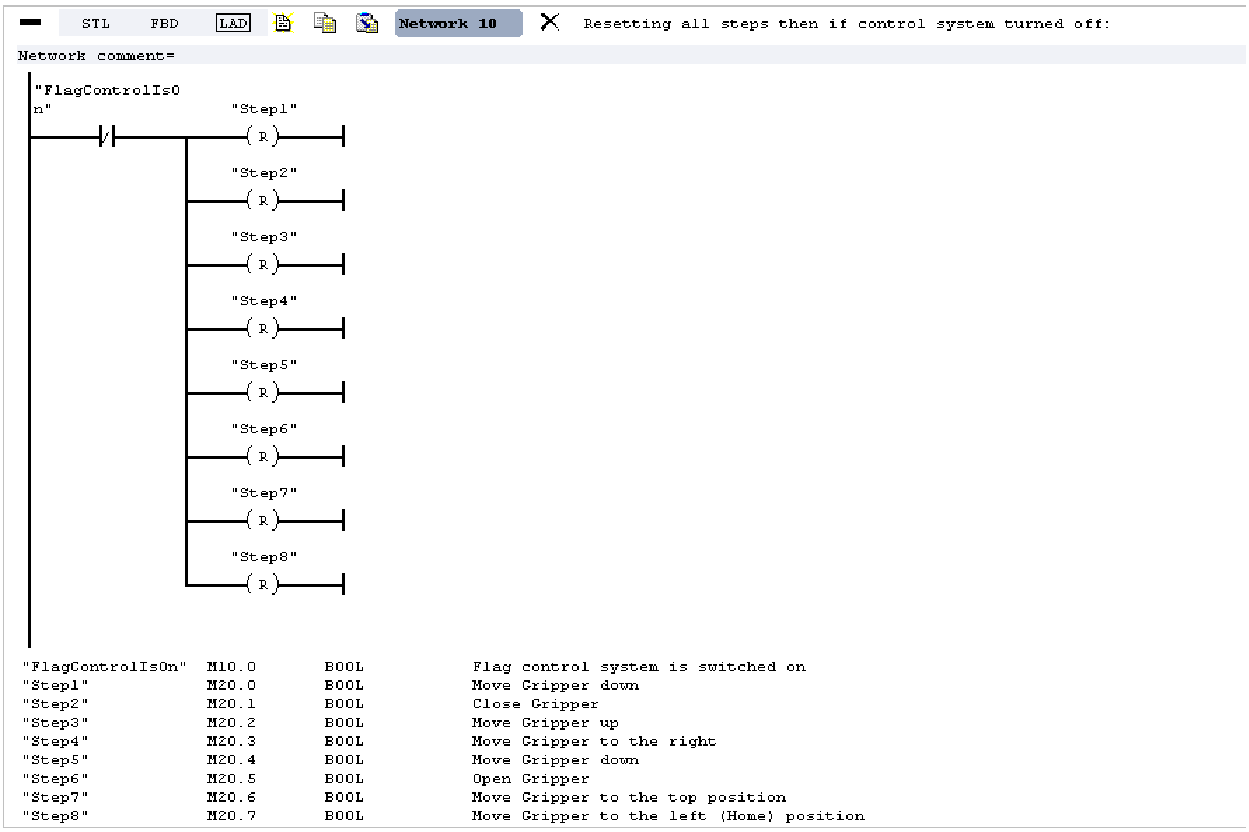
"Step7"	M20.6	BOOL	Move Gripper to the top position
"TopStop"	I0.6	BOOL	
"Step8"	M20.7	BOOL	Move Gripper to the left (Home) position

STL FBD LAD **Network 9** Resetting last step:

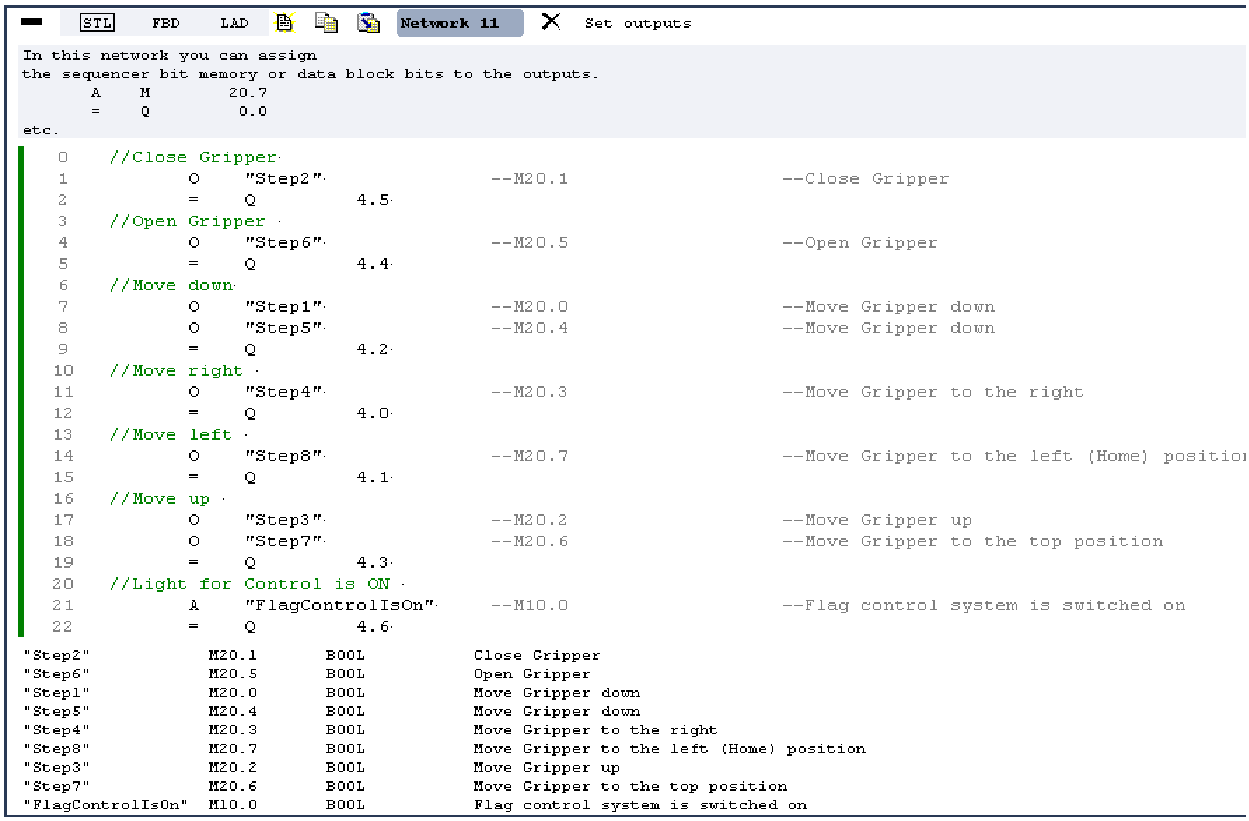
Network comment=

```
0     A     "Step8"     --M20.7                --Move Gripper to the left (Home) position
1     A     "LeftStop"  --I1.0
2     R     "Step8"     --M20.7                --Move Gripper to the left (Home) position
```

"Step8"	M20.7	BOOL	Move Gripper to the left (Home) position
"LeftStop"	I1.0	BOOL	



In network 11, the bit memories were assigned to the outputs:



21 Check consistency

The consistency check examines the entire PLC program for the following errors and inconsistencies:

- time-stamp errors (if the time-stamp of the block header is more recent than the code in the block from where the call to the block is issued).
- CALL error: do the transferred parameters still match the block header?
- are all the instance blocks still up to date?
- do time-stamp conflicts exist with respect to UDTs?

You can start the check via menu item *Extras->Check consistency*:

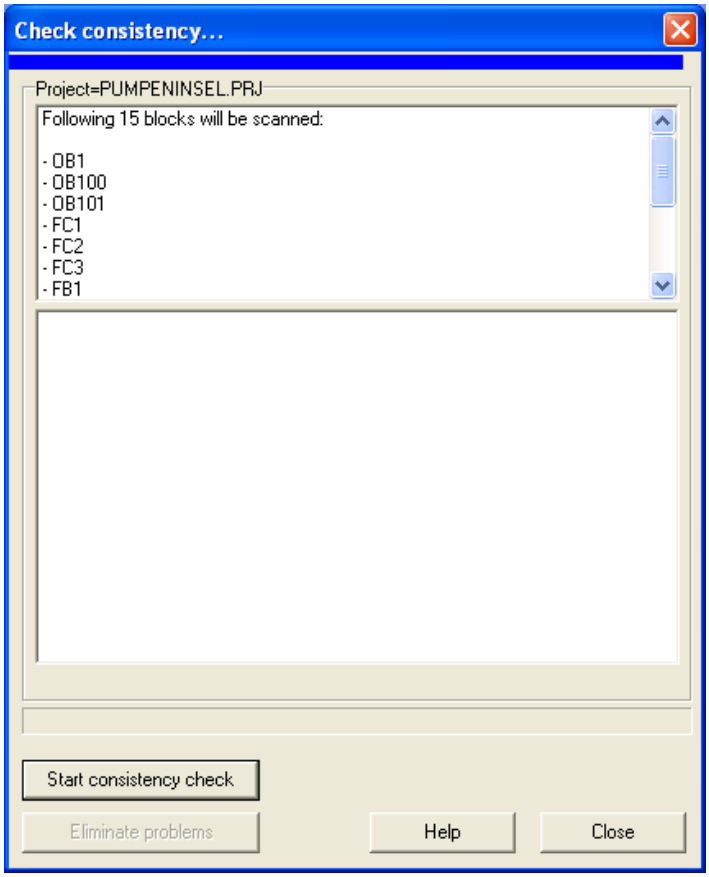


Fig.: "Check consistency" dialog

When you press the "Start consistency check" button, all the blocks in the current project will be verified.

The display will have the following appearance if the check does not return any errors:

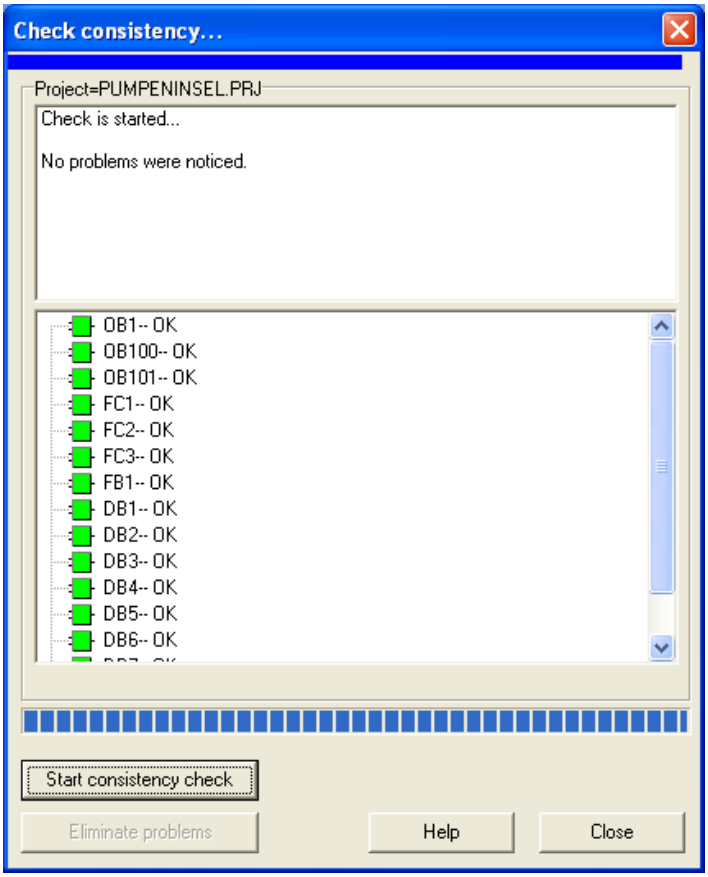


Fig.: consistency check without errors

21.1 Error type: "Time-stamp error in CALL"

This error indicates that the block header of a block with parameters was changed at a later stage; however, the CALL command is still correct (this may occur if you have made changes to the static local variables of a FB).
If this refers to a FB, all the instance data blocks of the FB must be modified.
This error is corrected automatically by the dialog when you press the button "Eliminate problems".

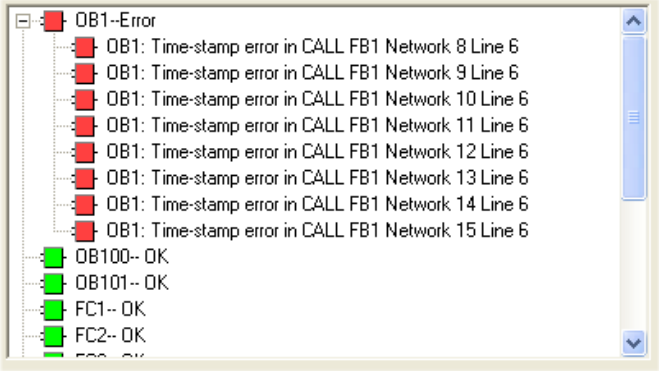


Fig.: time stamp error

21.2 Error type: "Structure of call xy is wrong"

This error indicates that a CALL command is bad because parameters are missing or have been added.

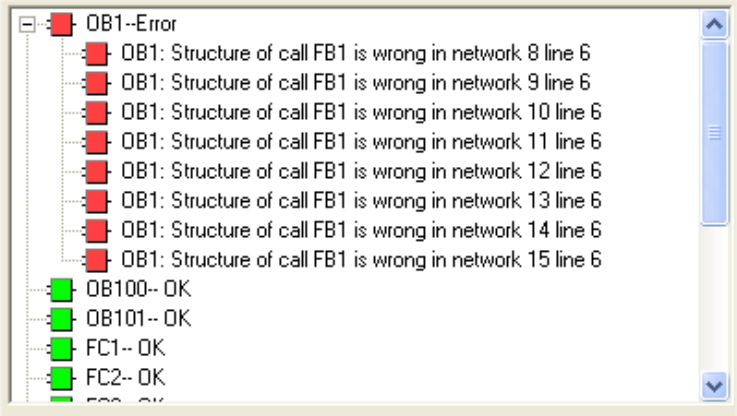


Fig.: error in the structure of a call

If **parameters have been added**, these parameters must also be added to every call in the PLC program.
The button "Eliminate problems" in the "Check consistency" dialog provides you with additional support:
The "Edit problems" dialog is displayed:

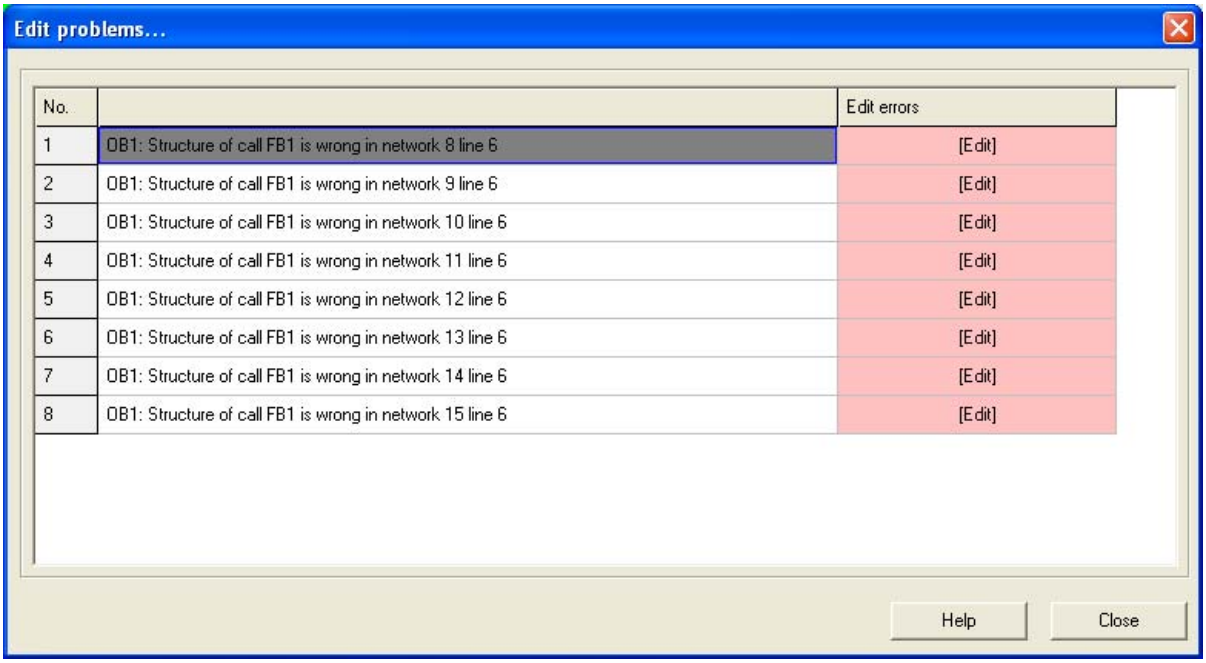


Fig.: “Edit problems” dialog

Click on the "Edit" cell. A dialog will be displayed that shows the call command in a table:

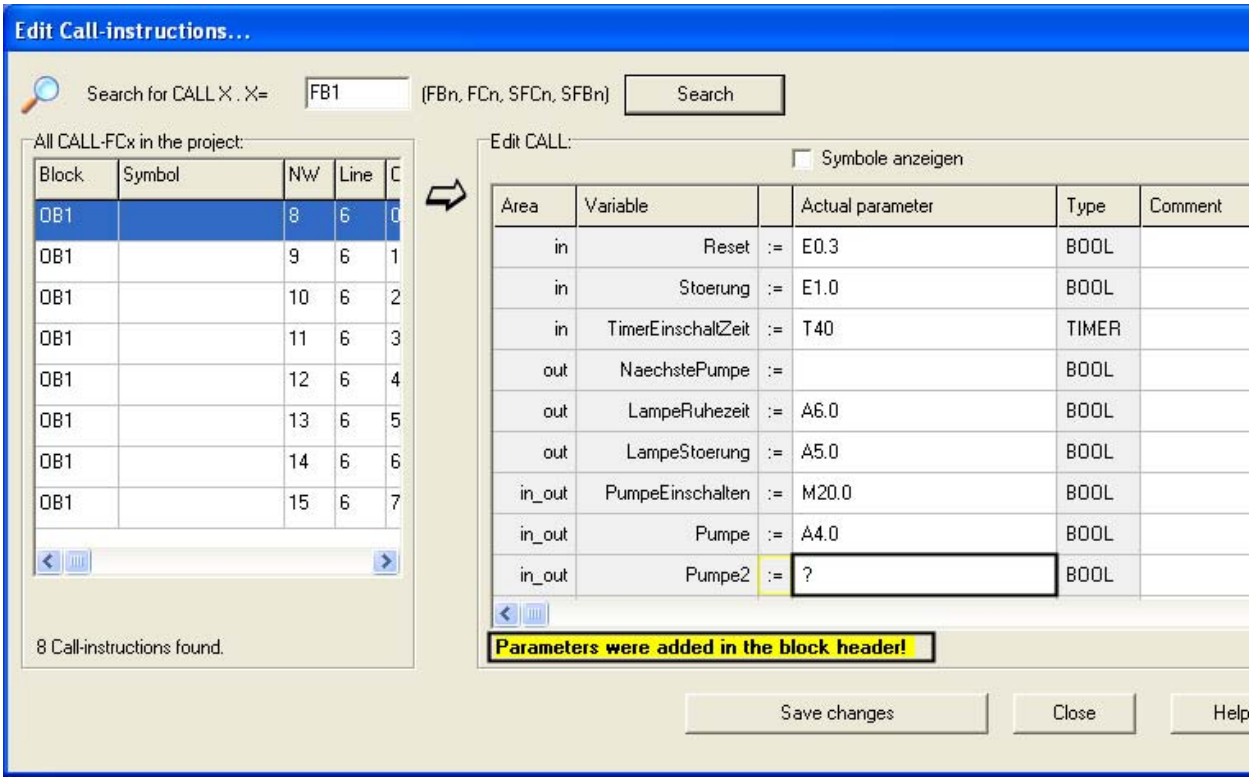


Fig.: The CALL FB1- command in the form of a table

A question mark identifies all the additional parameters in the “Actual parameter” column. In this specific case, a parameter is **not** necessary since this refers to a CALL FB.
The list on the left side shows all "CALL FB1" commands in the PLC program. Now you can edit all the call commands one by one.
When you have edited all the CALLs, press the "Save changes" button.

The "Edit problems" dialog indicates that all the problems have been corrected:

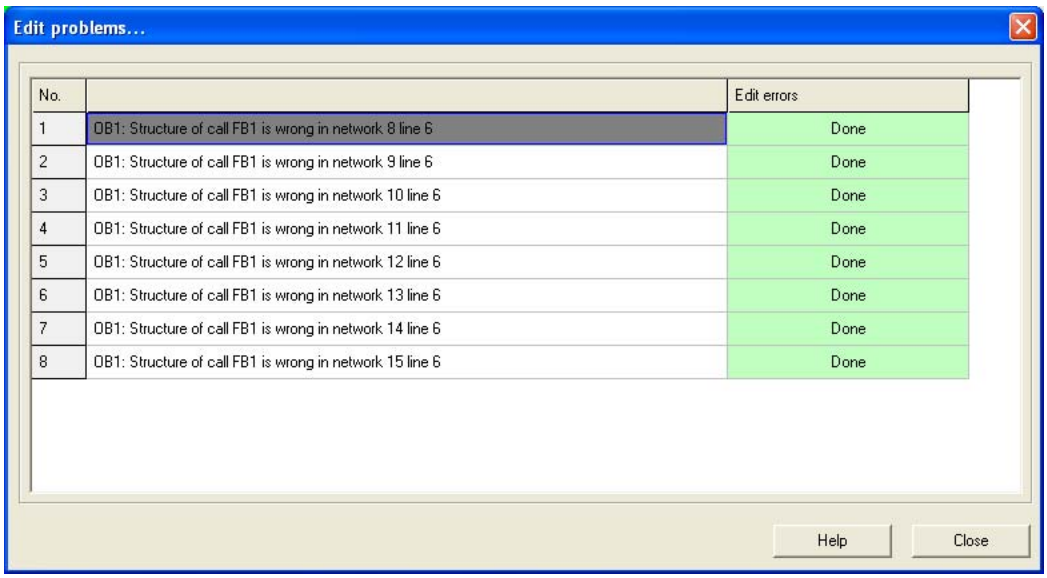
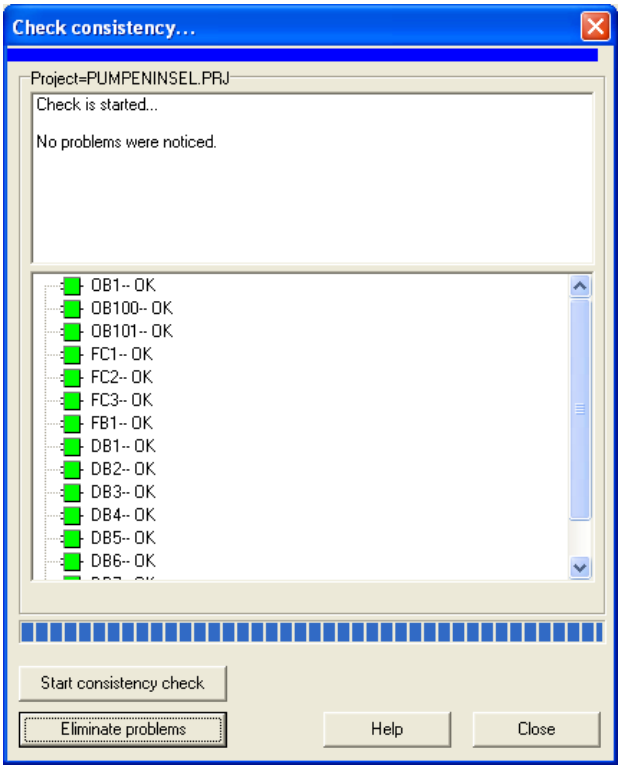


Fig.: "Edit problems" - dialog

Press the "Close" button. The "Check consistency" dialog is displayed without the errors:



"Check consistency" dialog

If **parameters were deleted from the block header**, the necessary changes can be made automatically and without manual intervention.
In this case the parameter is deleted from every call.

21.3 Dialog: "Edit call"

You can also access the "Edit call" dialog of the consistency check via menu item *Extras->Edit call*.

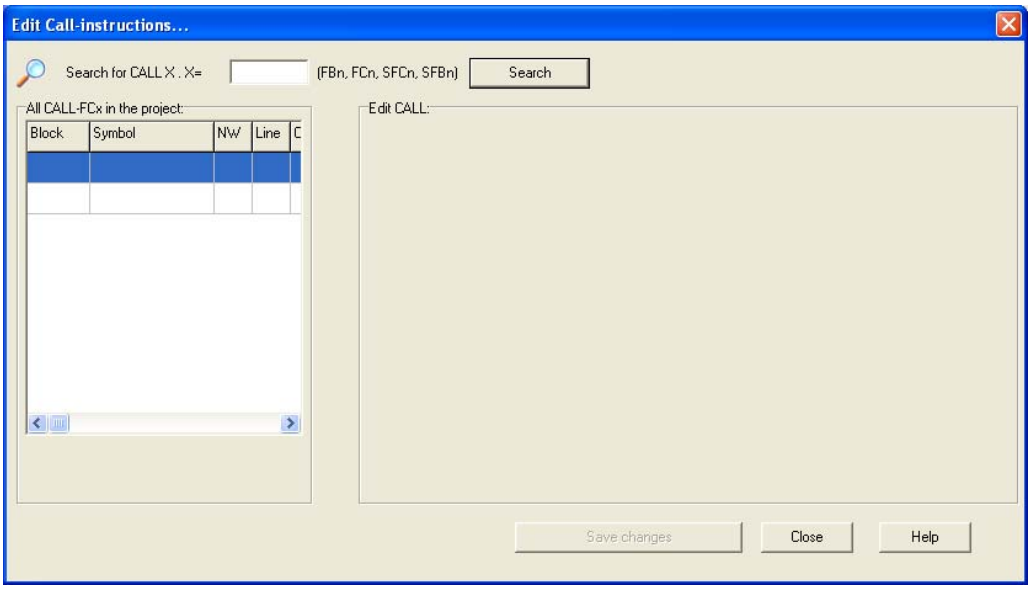


Fig.: "Edit Call-instructions" dialog

Use this dialog to edit a call-instruction that appears several times in the PLC program. You can also use this dialog to quickly verify the parameters that are supplied to a specific CALL.

22 Determining the reason for a stop (diagnostic options)

If the S7 PLC unexpectedly goes to STOP mode, this can have several reasons:

- a hardware problem related to the connected modules
- a PLC program error

You can investigate the error using the following tools:

1. Dialog "Module state", tab "Diagnostic"
2. Dialog "Module state", tab "ISTACK/BSTACK"

22.1 Diagnostic messages in the module state dialog

The diagnostic buffer is the first location to investigate when problems occur. Open the module state via menu item *PLC->Module state*
(hotkey= CTRL+D).
The "Diagnostic" tab lists all the diagnostic messages of the S7 CPU.

Example 1 of an error:

An intentional error that results in an endless loop was inserted into the CPU.
The CPU goes to STOP mode and the diagnostic buffer appears as follows:

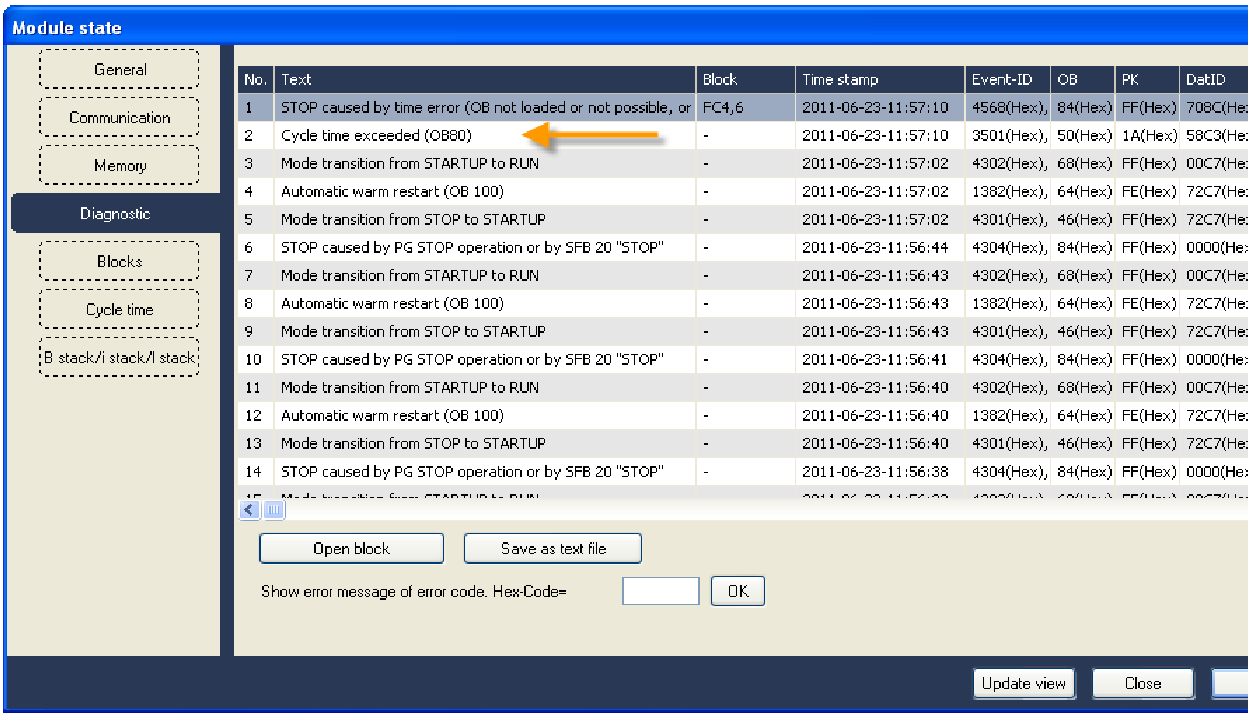


Fig.: diagnostic messages

Line No. "1" shows the error "**STOP caused by time error (OB not loaded or…)**". The next column identifies the location of the error in the PLC program: FC4, address 6 (decimal). Press the "**Open block**" button to jump directly to the location of the error.
Line No. "2" shows the actual error: "**Cycle time exceeded OB80**".

22.2 Interrupt stack (ISTACK) and block stack (BSTACK) in the dialog module state

The dialog ISTACK and BSTACK provide additional information when a program error is detected:

- register contents (status word, accumulators, AR1, AR2)
- point of interrupt (block, rel. byte address)
- open data blocks

Example 2 of an error:

An intentional error was inserted into the PLC program. Data word 210 (DBW210) was accessed, although the size of the data block is 200 bytes.

The diagnostic buffer now appears as follows:

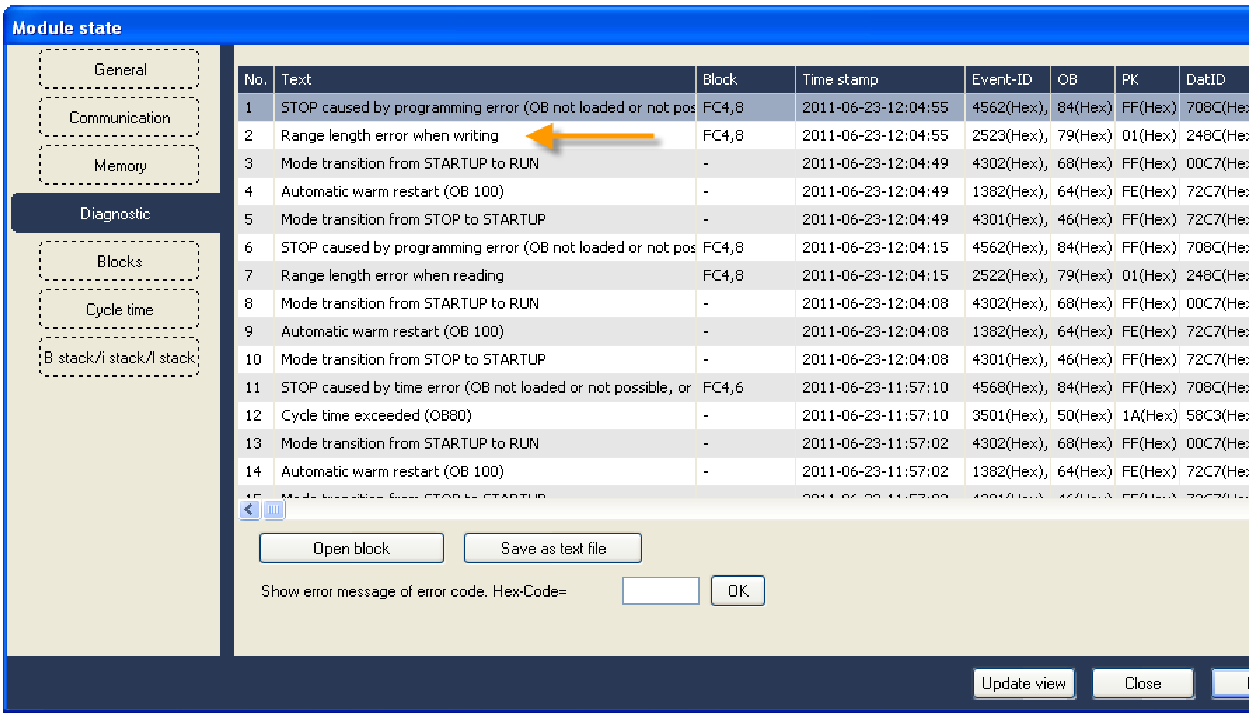


Fig.: diagnostic buffer

Line No. "1" contains: " **STOP caused by time error-OB**". The next column to the right identifies the location of the error: block FC4 and byte address 8 (decimal).
Line No. "2" shows the actual error: "**Range length error when writing**".
Here you can also press the "**Open block**" button to jump directly to the location of the error in FC4.

Open the tab "BStack/IStack/LStack" to display the block stack (BSTACK):

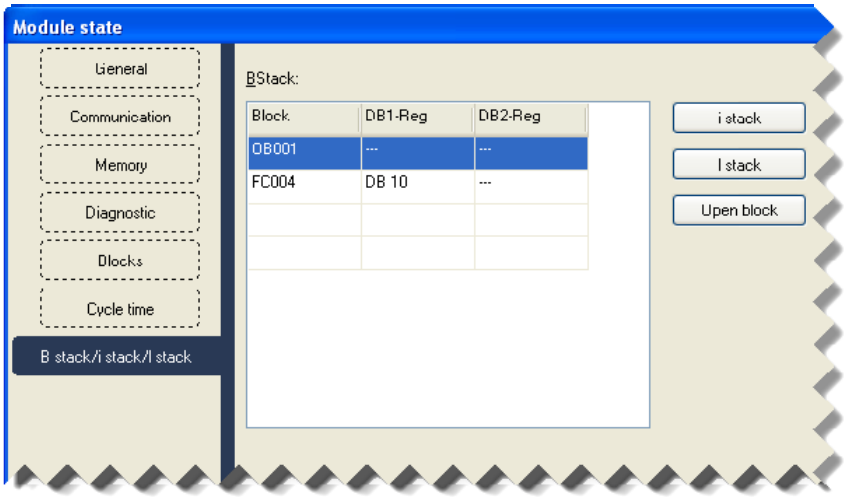


Fig.: displaying the BSTACK

The block stack indicates from which block the call to the error block was issued (call stack).
This information is very important if calls to the block with the error are issued several times in the PLC program.

Press the "ISTACK" button. The following dialog appears:

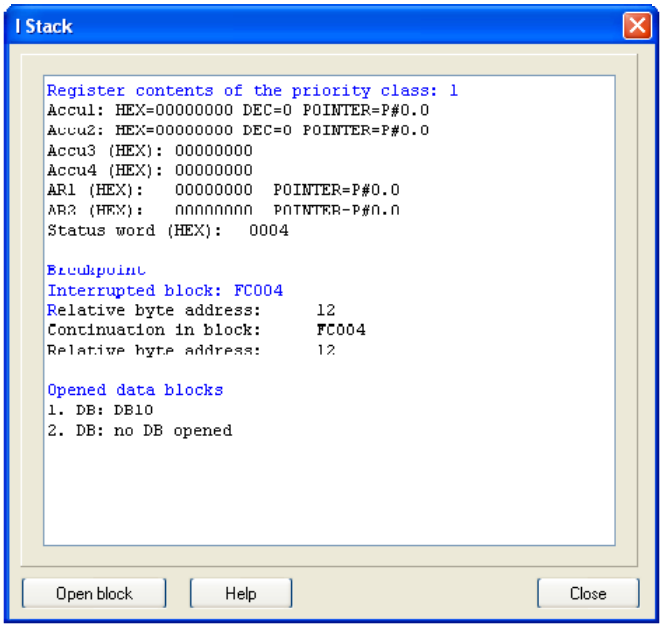


Fig.: ISTACK display

The ISTACK indicates the contents of the registers, the point of interruption and the open data blocks.
The diagnostic buffer does not provide all of this information. The ISTACK is very important with regard to indirect addressing, because it also indicates the contents of address registers AR1 and AR2.

Press the "Open block" button to jump directly to the location of the error in the PLC program.

22.3 Go to byte address

Both, the diagnostic buffer and the ISTACK display addresses within blocks.
This is the byte address relative to the beginning of the block.

This dialog also has the "Open block" button to enable you to jump to this address in the block. The button opens the block and jumps to respective byte address.

To access another location in the block, open the block first and then select menu item ***Edit->Goto byte address***:

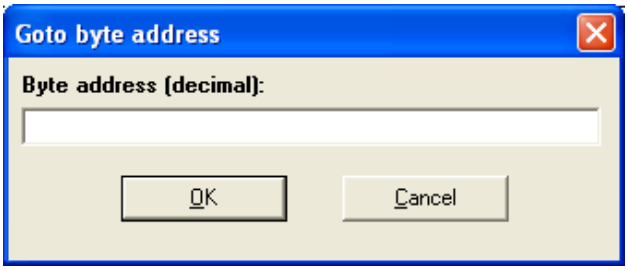


Fig.: Goto byte address

Enter the byte address (in decimal format).
Press "OK" to jump to the respective location in the block.

Notes:

23 Remote maintenance with WinPLC7

23.1 Accessing a S7 PLC via the internet

Remote maintenance can employ a telephone line as well as the Internet. The following diagram illustrates the principle of such a connection.

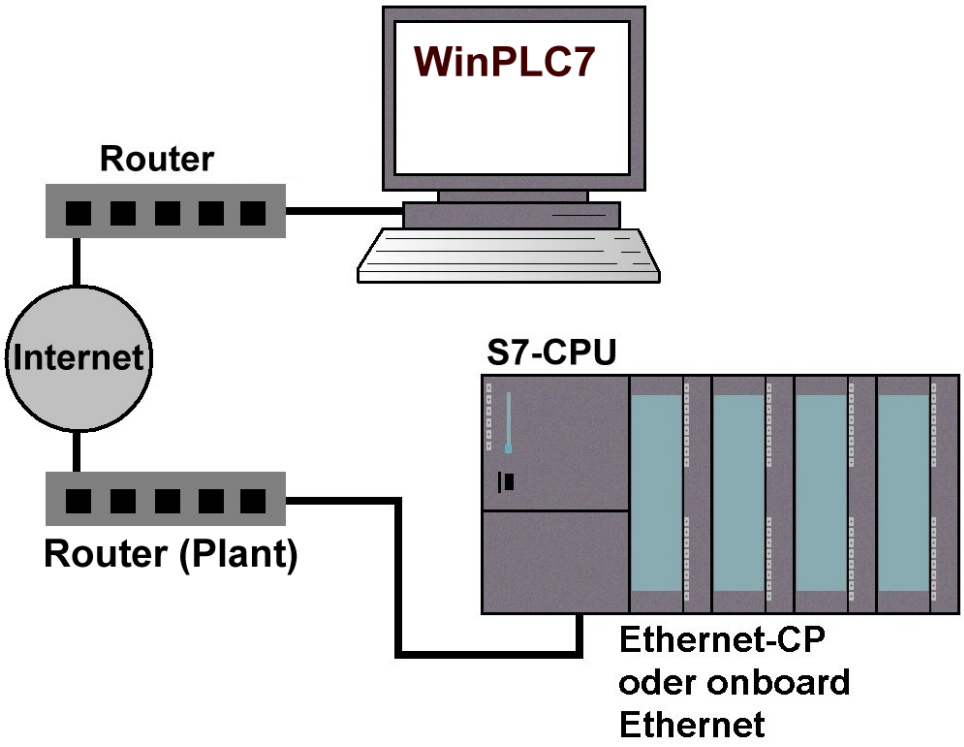


Fig.: Diagram to illustrate how a S7-300/400 is accessed via the internet

In the diagram WinPLC7 is running on a PC. This PC has an Ethernet connection to a router. The router establishes the connection with the Internet.
The system is also connected to the internet by means of a router. A number of different options exist to connect the router on the system side with the PLC.
These are:

- NetLink: A NetLink (NetLink Lite, IBH link) connection plugged into the router and connected to the MPI or Profibus-DP-interface of the CPU.
- NetLink PRO: If you are using NetLink PRO, this is plugged into the router and connected to the MPI or Profibus-DP-interface of the CPU.
- Ethernet CP34X/44X or an on-board Ethernet interface of the CPU: If the S7 PLC is equipped with an Ethernet interface it can be connected directly to the router. This connection requires a standard Ethernet cable.

The following paragraph describes the settings that are required for the different connections.

23.1.1 Direct TCP/IP connection

If the S7 PLC has an on-board Ethernet interface (e.g., SPEED7, 315PN/DP, etc.) or an Ethernet-CP (e.g., CP343), this can be connected directly to the router on the system side by means of an Ethernet cable.

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: TCP/IP-Direct".

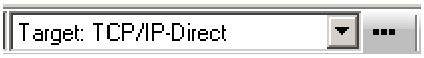


Fig.: TCP/IP-Direct selected

In the "Ethernet properties" dialog you specify the IP address of the router located on the system side, i.e. the IP address that the router uses to log in to the Internet. The following figure shows the dialog in WinPLC7. The IP address of the system router is 84.163.183.85.

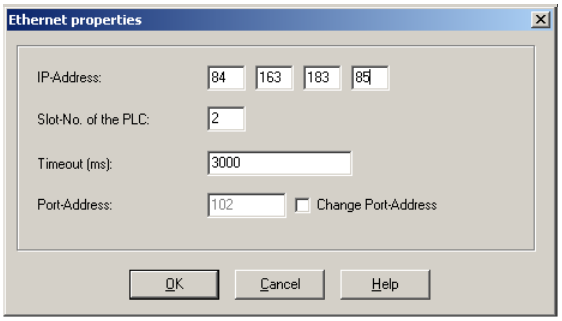


Fig.: Ethernet data dialog with the WAN IP address of the system router

These are all the settings that are required in WinPLC7.

Router settings required on the system side:

The system router has a specific IP address (in this example the address is 84.163.183.85). This IP address is accessed by WinPLC7. However, the router is not yet aware that it should transfer this request to the S7 PLC.
The request of WinPLC7 is transferred to the IP address as well as a specific port. The default port is 102 (available via the "Ethernet properties" dialog in WinPLC7). For this reason the system router must be configured to transfer any request that arrives via port 102 to the S7 PLC. On most routers this type of configuration is available under the "Virtual Server" settings. This consists of a table that specifies where the router should transfer requests arriving from the Internet (via a defined port).
The following figure shows one entry in a table; here the S7 PLC has the IP address 192.168.2.130 on the system LAN.

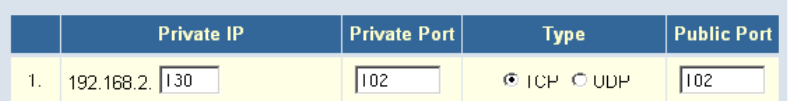


Fig.: "Virtual of server" entry in the system router

This entry determines that the request from WinPLC7 is transferred to IP address 192.168.2.130, provided that the inquiry arrives via port 102.

Settings in the S7 PLC:

Let us now turn to the settings on the S7 PLC. The PLC must have an Ethernet interface. This interface must be configured with an IP address that is located in a subnet of the system router. In our example we have used the IP address 192.168.2.130. If the S7 PLC is accessed by another station with an IP address in the same subnet, then the PLC can respond directly to the respective station since it can communicate directly with the station.
However, when a request for remote maintenance arrives the PLC receives the request from a station (i.e. from WinPLC7 via the Internet) that has an IP address that is not within the subnet. This means that the PLC is not able to respond directly to the requesting station. The response must be routed via a "gateway". It is therefore necessary to specify the address of the gateway in the hardware configuration of the Ethernet interface on the S7 PLC. The gateway is the system router, because this knows where the request to the S7 PLC originated and it can therefore return the response correctly (i.e. to WinPLC7 via the Internet).
The figure below shows the hardware configuration required for a CP343.

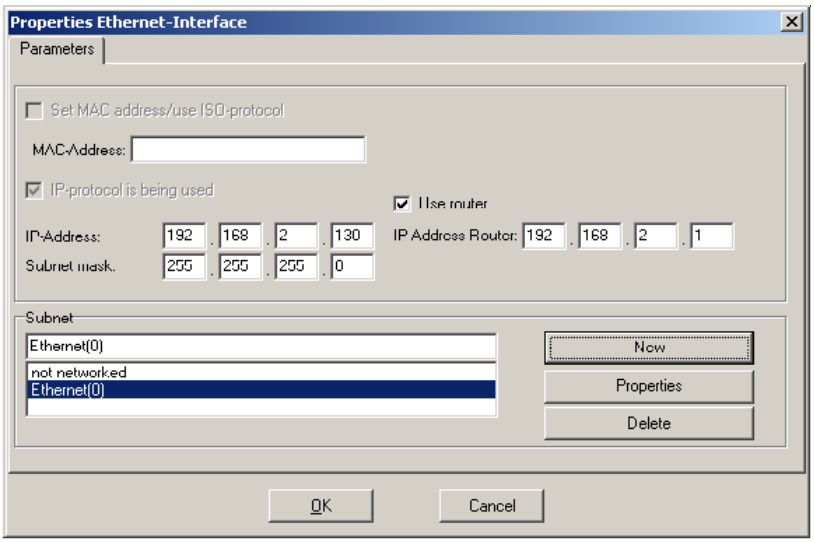


Fig.: configuration dialog for the Ethernet CP of the S7 PLC

You must make sure that you have selected the tick box "Use router" in the dialog. In addition the IP address of the system router is specified. In the example above this is 192.168.2.1.

Result:

Now we have completed all the settings that are required to allow WinPLC7 to access the S7 PLC via the internet.
It would be most desirable if the system router was connected to the internet via a dedicated line with a fixed public IP address. Otherwise a service like DDNS (Dynamic DNS) can supply the current IP address of the system router at any time.

23.1.2 NetLink PRO connection

NetLink PRO can be used to establish an Ethernet to MPI/Profibus-DP connection. In the remote maintenance example the NetLink PRO is inserted into the system router. On the S7 PLC the NetLink PRO is connected to the MPI or the Profibus-DP-interface of the CPU (also refer to the diagram at the beginning of this chapter).

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: NetLink PRO TCP/IP".

Fig.: Selected target in WinPLC7 "NetLink PRO TCP/IP"

Enter the IP address of the system router into the "NetLink PRO" dialog. In this example the address is 84.163.183.85. In addition, tick box "No baud rate detection" is selected. However, this means that the baud rate defined in the dialog matches the baud rate specified for the S7 PLC.

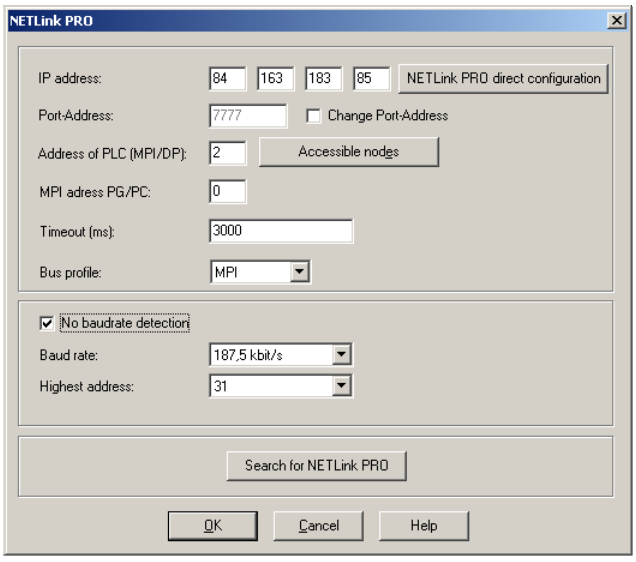


Fig.: "NetLink PRO" dialog with the WAN IP address of the system router

This completes the settings for WinPLC7.

Router settings required on the system side:

The system router has a specific IP address (in this example the address is 84.163.183.85). This IP address is accessed by WinPLC7. However, the router is not yet aware that it should transfer this request to the S7 PLC.
The request of WinPLC7 is transferred to the IP address as well as a specific port. A NetLink PRO is accessed via port 7777 (available from the "NetLink PRO" dialog in WinPLC7). For this reason the system router must be configured to transfer any request that arrives via port 7777 to the NetLink PRO (i.e. to the S7 PLC). Most routers refer to this type of configuration as "Virtual Server". This consists of a table that specifies where the router should transfer requests arriving from the Internet (via a defined port).
The following figure shows one entry in a table; here the NetLink PRO has the IP address 192.168.2.141 on the system LAN.

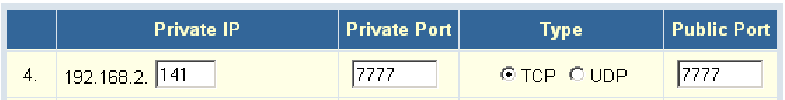


Fig.: "Virtual of server" entry in the system router

This entry determines that the request from WinPLC7 is transferred to IP address 192.168.2.141, provided that the inquiry arrives via port 7777.

NetLink PRO settings required on the system side:

Let us continue with the system side settings for the NetLink PRO. The IP address of the NetLink PRO is configured to lie within the subnet of the system router. In our example we have used the IP address 192.168.2.141. If the NetLink PRO is accessed by another station with an IP address in the same subnet, then it can respond directly to the respective station since the NetLink PRO can communicate directly with the station. However, when a request for remote maintenance arrives, the NetLink PRO receives the request from a station (i.e. from WinPLC7 via the Internet) that has an IP address that is not within its subnet. This means that the NetLink PRO cannot respond directly to the requesting station. The response must be routed via a “gateway”. It is therefore necessary to specify the address of the gateway in the configuration of the NetLink PRO. The gateway is the system router, because this knows where the request to the NetLink PRO (i.e. to the S7 PLC) originated and it can therefore return the response correctly (i.e. to WinPLC7 via the Internet).
The figure below shows the configuration required for the NetLink PRO. In WinPLC7 the configuration dialog can be accessed via menu item "Tools->Target properties: NetLink PRO TCP/IP". In the first dialog, press the button "Netlink PRO Direct Configuration". If the IP address of the NetLink PRO is not available you can press the button "List all NetLink PRO" followed by the "Settings" button.

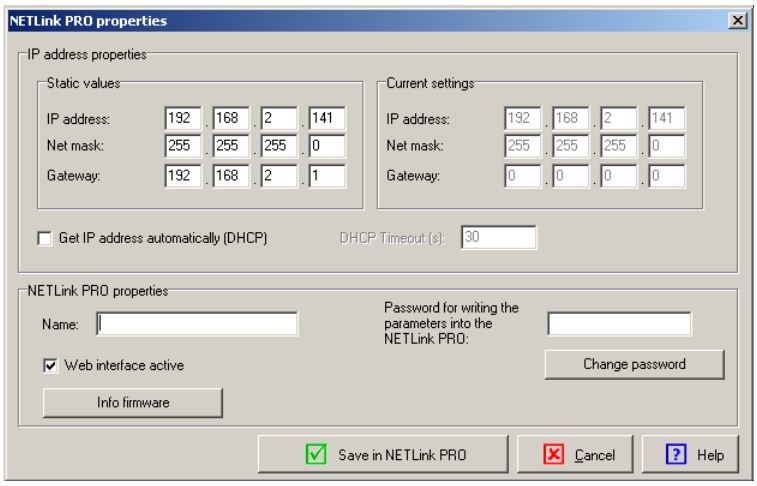


Fig.: NetLink PRO configuration on the system side

You must make sure that you have entered the address of the gateway in the dialog. In the example this address is 192.168.2.1 which corresponds to the IP address of the system router. A click of the button "Save NETLink PRO saves the settings to the NetLink PRO.

Result:
Now we have completed all the settings that are required to allow WinPLC7 to access the NetLink PRO (and therefore the S7 PLC) via the internet.
It would be most desirable if the system router was connected to the internet via a dedicated line with a fixed public IP address. Otherwise a service like DDNS (Dynamic DNS) can supply the current IP address of the system router at any time.

23.1.3 NetLink connections (NetLink Lite, IBH Link)

NetLink can be used to establish an Ethernet to MPI/Profibus-DP connection. In the remote maintenance example the NetLink is inserted into the system router. On the S7 PLC the NetLink is connected to the MPI or the Profibus-DP-interface of the CPU (also refer to the diagram at the beginning of this chapter).
Please note that the NetLink offers the lowest remote maintenance performance in comparison to the two other versions, i.e. TCP/IP-direct and NetLink PRO. However, this still outperforms a connection via an analog telephone line.

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: MHJ-NetLink TS". It is important that you do not use "Target: Net link MHJ" as the type of connection. This setting is only valid for a direct NetLink connection (this is required for the NetLink configuration below).

Fig.: selected type of connecting "Target: MHJ-NetLink TS"

Enter the IP address of the system router into the "MHJ-NetLink" dialog. In the example this is 84.163.183.85.

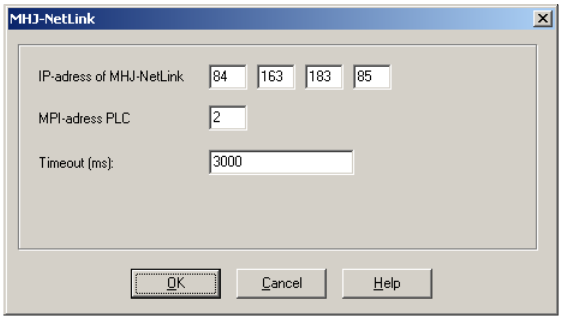


Fig.: "MHJ-NetLink" dialog with the IP address of the system router

Router settings required on the system side:

Thesystem router has a specific IP address (in this example the address is 84.163.183.85). This IP address is accessed by WinPLC7. However, the router is not yet aware that it should transfer this request to the S7 PLC or the NetLink.
The request of WinPLC7 is transferred to the IP address as well as a specific port. The NetLink is accessed via port 1099. For this reason the system router must be configured to transfer any request that arrives via port 1099 to the NetLink (i.e. to the S7 PLC). Most routers refer to this type of configuration as "Virtual Server". This consists of a table that specifies where the router should transfer requests arriving from the Internet (via a defined port).
The following figure shows one entry in a table; here the NetLink has the IP address 192.168.2.115 on the system LAN**.**

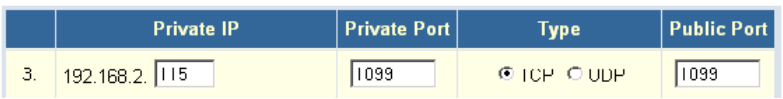


Fig.: "Virtual of server" entry in the system router

This entry determines that the request from WinPLC7 is transferred to IP address 192.168.2.130, provided that the inquiry arrives via port 1099.

NetLink settings on the system side:

Let us now turn to the settings required on the NetLink on the system side. The NetLink must be configured with an IP address that is located in a subnet of the system router. In our example we have specified an IP address of 192.168.2.115. If the NetLink is accessed by another station with an IP address in the same subnet it can respond directly to the respective station, since it can communicate directly with the station.
However, when a request for remote maintenance arrives, then the NetLink receives the request from a station (i.e. from WinPLC7 via the Internet) that has an IP address that is not within the subnet. This means that the NetLink is not able to respond directly to the requesting station. The response must be routed via a so-called gateway. It is therefore necessary to specify the address of the gateway in the configuration of the NetLink. The gateway is the system router, because this knows where the request to the NetLink originated and it can therefore return the response correctly (i.e. to WinPLC7 via the Internet).
The figure below shows the configuration required for a NetLink. In WinPLC7 the configuration dialog can be accessed via menu item "Tools->Target properties: MHJ-NetLink".

Note:
Please remember to select "Target: MHJ-NetLink" when you configure the connection type, and not "Target: MHJ-NetLink TS" that is required for access by the actual remote maintenance.

The figure below illustrates the NetLink dialog with the required settings:

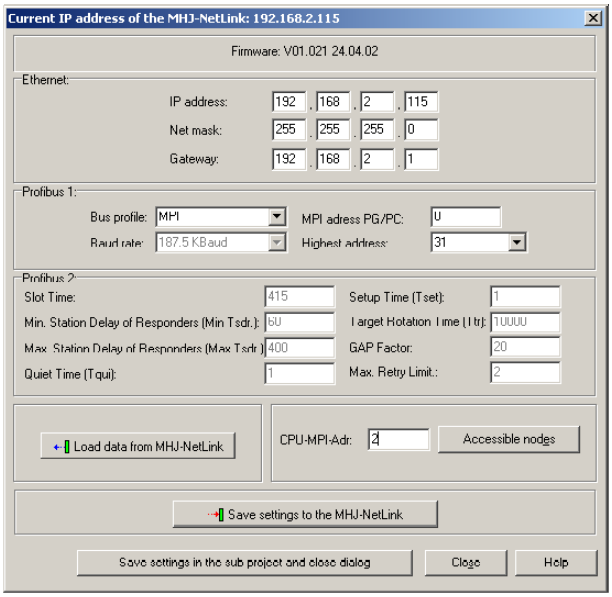


Fig.: the NetLink configuration dialog

Please note that the address of the gateway was entered into this dialog. The address is 192.168.2.1 and in our example this corresponds to the IP address of the system router. A click of the button "Save settings to the MHJ-NetLink saves the settings to the NetLink.

Result:
Now we have completed all the settings that are required to allow WinPLC7 to access the NetLink (and therefore the S7 PLC) via the internet. Let us repeat again, you must have selected "MHJ-NetLink TS" as the connection type in WinPLC7.
The best solution would be to connect the system router to the internet via a dedicated line using a fixed public IP address. Otherwise a service like DDNS (Dynamic DNS) can supply the current IP address of the system router at any time**.**

23.1.4 Summary regarding the internet access to a S7 PLC

The description above illustrated how to configure a S7 CPU for access via the internet. The following basic settings were explained. These settings enable good performance access to the S7 PLC for remote maintenance purposes. In addition we recommend that you consider a VPN connection. Please refer to the relevant literature in this respect.

23.2 Accessing a S7 PLC via ISDN by means of LAN to LAN routers

The last chapter explained how you could implement access via the internet for remote maintenance purposes. The respective connection types were TCP/IP-direct, NetLink PRO and NetLink. The connection was established via the router and the internet. It is also possible to establish a connection by means of an ISDN line and ISDN routers that support LAN-to-LAN traffic routing.
The following diagram illustrates the principle of such a connection:

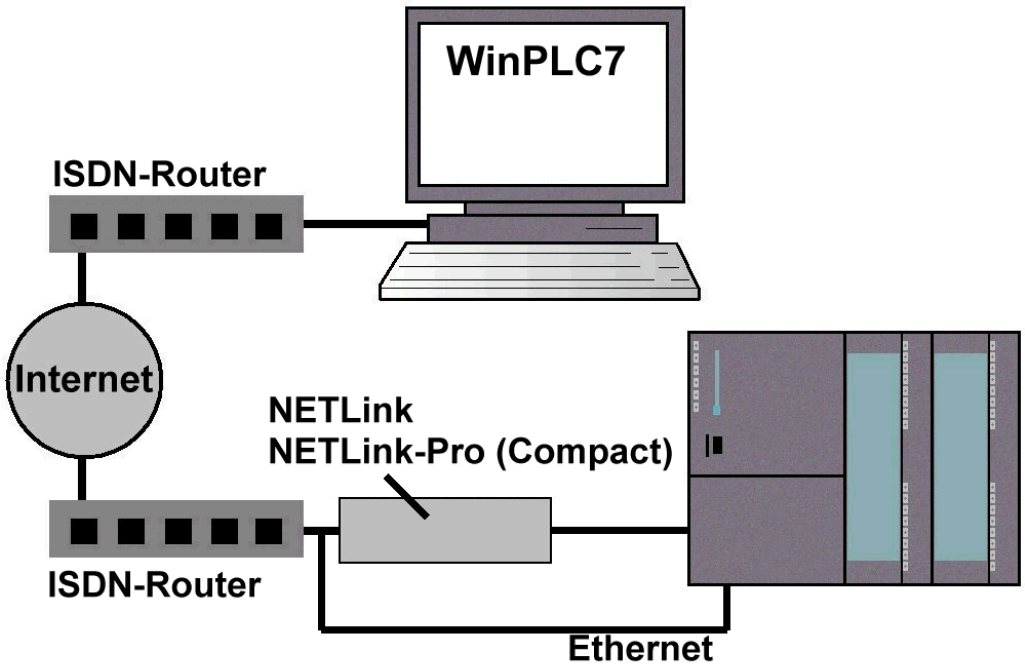


Fig.: remote maintenance via ISDN and LAN-to-LAN routers

It is obvious that you can use the same connections here as the ones that you used to enable the internet connectivity.
The configuration is similar to the one described in the previous chapter, which is why we will not describe all the details here. We recommend that you read the preceding chapter in preparation for the procedures outlined in this chapter. The settings for the ISDN routers differ, depending on the respective make. However, the basic settings or the principles are always the same.

23.2.1 How does the connection operate and what are the conditions that apply to establish a connection

A LAN-to-LAN connection must be defined in the settings of the two ISDN routers. Make sure that different IP addresses are used for the two subnets that manage the routers. In the example the router of the WinPLC7 network manages IP addresses from 192.168.2.100 and the routers IP address is 192.168.2.1.
The system router manages the IP addresses from 192.168.3.100 and the IP address of the router is 192.168.3.1.
A dial-out connection is defined on the router where the PC running WinPLC7 is connected. The respective settings include the telephone number of the system router as well as the start IP-address of the address space that is managed by the system router.
A dial-in connection is defined on the system router. Here you can also specify the telephone number of the router that should have exclusive dial-in rights.
The WinPLC7 router will establish an ISDN connection to the system router if WinPLC7 issues an access request to an IP address that is located in the address space of the system router. This means that the connection request is routed via the ISDN network.

The following passages assume that the IP address of the respective S7 PLC, the NetLink or NetLink PRO is 192.168.3.110.

23.2.2 Direct TCP/IP connection

If the S7 PLC has an on-board Ethernet interface (e.g., SPEED7, 315PN/DP, etc.) or an Ethernet-CP (e.g., CP343), this can be connected directly to the ISDN router on the system side by means of an Ethernet cable.

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: TCP/IP-Direct”. Enter the IP address 192.168.3.110 as the target address in the "Ethernet properties" dialog.

Settings in the S7 PLC:

The Ethernet interface on the S7 PLC must have an IP address that is located within the subnet of the system router. In the example this is 192.168.3.110. The IP address of the system router must also be configured (select tick box "Use router"). In the example this is 192.168.3.1.

Result:

The router will establish an ISDN connection to the system router if WinPLC7 issues a request to IP address 192.168.2.110. This means that the request is routed to the Ethernet interface of the S7 PLC via the ISDN network.

23.2.3 NetLink PRO connection

NetLink PRO can be used to establish an Ethernet to MPI/Profibus-DP connection. In the remote maintenance example the NetLink PRO is inserted into the system router. On the S7 PLC the NetLink PRO is connected to the MPI or the Profibus-DP-interface of the CPU (also refer to the diagram at the beginning of this chapter).

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: NetLink PRO TCP/IP".

Enter the IP address of the NetLink PRO into the "NetLink PRO" dialog. In this example the address is 192.168.3.110. In addition, tick box "No baud rate detection" is selected. However, this means that the baud rate defined in the dialog matches the baud rate specified for the S7 PLC.

NetLink PRO settings on the system side:

Let us now turn to the settings required on the NetLink PRO located on the system side. The NetLink PRO must be configured with an IP address that is located in a subnet of the system router. In our example we have specified an IP address of 192.168.3.110. If the NetLink PRO is accessed by another station with an IP address in the same subnet it can respond directly to the respective station, since it can communicate directly with the station.
However, when a request for remote maintenance arrives, then the NetLink PRO receives the request from a station (i.e. from WinPLC7 via the Internet) that has an IP address that is not within the subnet. This means that the NetLink PRO is not able to respond directly to the requesting station. The response must be routed via a so-called gateway. It is therefore necessary to specify the address of the gateway in the configuration of the NetLink PRO. The gateway is the system router, because this knows where the request to the NetLink PRO originated and it can therefore return the response correctly (i.e. to WinPLC7 via ISDN).

For this reason you must specify both, IP address 192.168.3.110 of the NetLink PRO as well as the IP address of the gateway 192.168.3.1 (system router) in the configuration dialog of the NetLink PRO.

Result:

The router will establish an ISDN connection to the system router if WinPLC7 issues a request to IP address 192.168.2.110. This means that the request is routed via the ISDN network to the NetLink PRO on the S7 PLC.

23.2.4 NetLink connections (NetLink Lite, IBH Link)

NetLink can be used to establish an Ethernet to MPI/Profibus-DP connection. In the remote maintenance example the NetLink is inserted into the system router. On the S7 PLC the NetLink is connected to the MPI or the Profibus-DP-interface of the CPU (also refer to the diagram at the beginning of this chapter).

WinPLC7 settings:

The selected type of connection will be WinPLC7 "Target: MHJ-NetLink TS". It is important that you do not use "Target: Net link MHJ" as the type of connection. This setting is only valid for a direct NetLink connection (this is required for the NetLink configuration below).

NetLink settings on the system side:

Let us now turn to the settings required on the NetLink on the system side. The NetLink must be configured with an IP address that is located in a subnet of the system router. In our example we have specified an IP address of 192.168.3.110. If the NetLink is accessed by another station with an IP address in the same subnet it can respond directly to the respective station, since it can communicate directly with the station.
However, remote maintenance requests are issued by stations (i.e. WinPLC7 via ISDN) that have IP addresses that are not located within the subnet of the NetLink. This means that the NetLink is not able to respond directly to the requesting station. The response must be routed via a so-called gateway. It is therefore necessary to specify the address of the gateway in the configuration of the NetLink. The gateway is the system router, because this knows where the request to the NetLink (and therefore to the S7 PLC) originated, and it can therefore return the response correctly (i.e. to WinPLC7 via ISDN).
In WinPLC7 the configuration dialog can be accessed via menu item "Tools->Target properties: MHJ-NetLink".

Note:
Please remember to select "Target: MHJ-NetLink" when you configure the connection type, and not "Target: MHJ-NetLink TS" that is required for access by the actual remote maintenance.

You must also make sure that you specify the IP address of the gateway when you configure the NetLink. The address is 192.168.3.1 and in our example this corresponds to the IP address of the system router. A click of the button "Save settings to the MHJ-NetLink saves the settings to the NetLink.

Result:
The router will establish an ISDN connection to the system router if WinPLC7 issues a request to IP address 192.168.2.110. This means that the request is routed via the ISDN network to the NetLink on the S7 PLC**.**

23.2.5 Summary of Accessing a S7 PLC via ISDN Routers

The description above illustrated how it is possible to access a S7 CPU via ISDN. The following basic settings were explained. These settings enable good performance access to the S7 PLC for remote maintenance purposes.

23.3 Remote maintenance by means of the Siemens Teleservice software with the Teleservice adapter

Fig.: Siemens Teleservice software and hardware
Source: image database at www.automation.siemens.com

The Teleservice software is normally used with STEP®7 of SIEMENS or Prodave of SIEMENS.
Since WinPLC7 provides support for the "Simatic-Net" communication path it also provides support for remote maintenance via the Siemens Teleservice software.

The following are required to execute remote maintenance:

1. WinPLC7 Version 6
2. Teleservice of SIEMENS (order number 6ES7842-0CC10-0YA5)
3. Teleservice adapter
4. Standard modem in the PC

Many Teleservice adapters are available:

Adapter	**Order number**
TS-adapter II analog with integrated modem from SIEMENS	6ES7972-0CB35-0XA0
TS-adapter II isdn with integrated modem from SIEMENS	6ES7972-0CC35-0XA0
TS-adapter serial (without modem)	6ES7972-0CA34-0XA0

23.3.1 Preparations

The figure below illustrates the principle of the communication path:

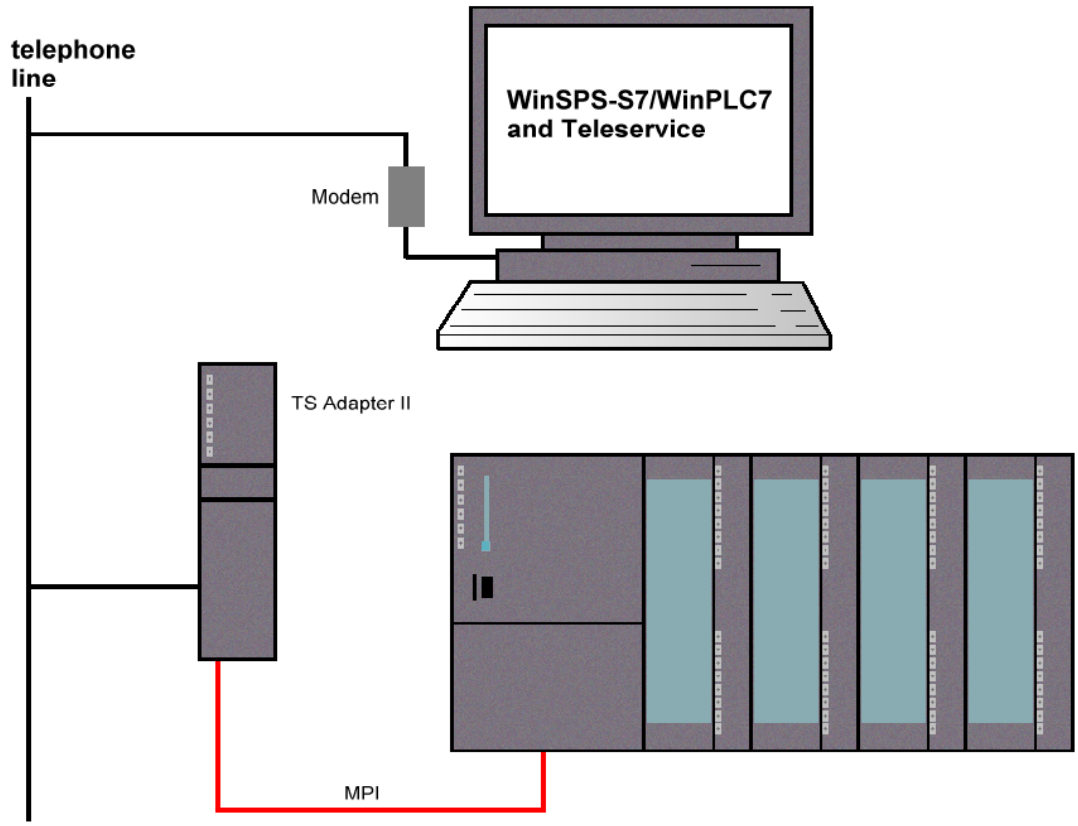


Fig.: WinPLC7 and SIEMENS Teleservice

WinPLC7 is connected to the telephone line via a **commercially available modem**. The **Teleservice II-adapter** is connected to the telephone line. This establishes the connection with the S7 PLC using the MPI interface.

23.3.2 The SIEMENS Teleservice software

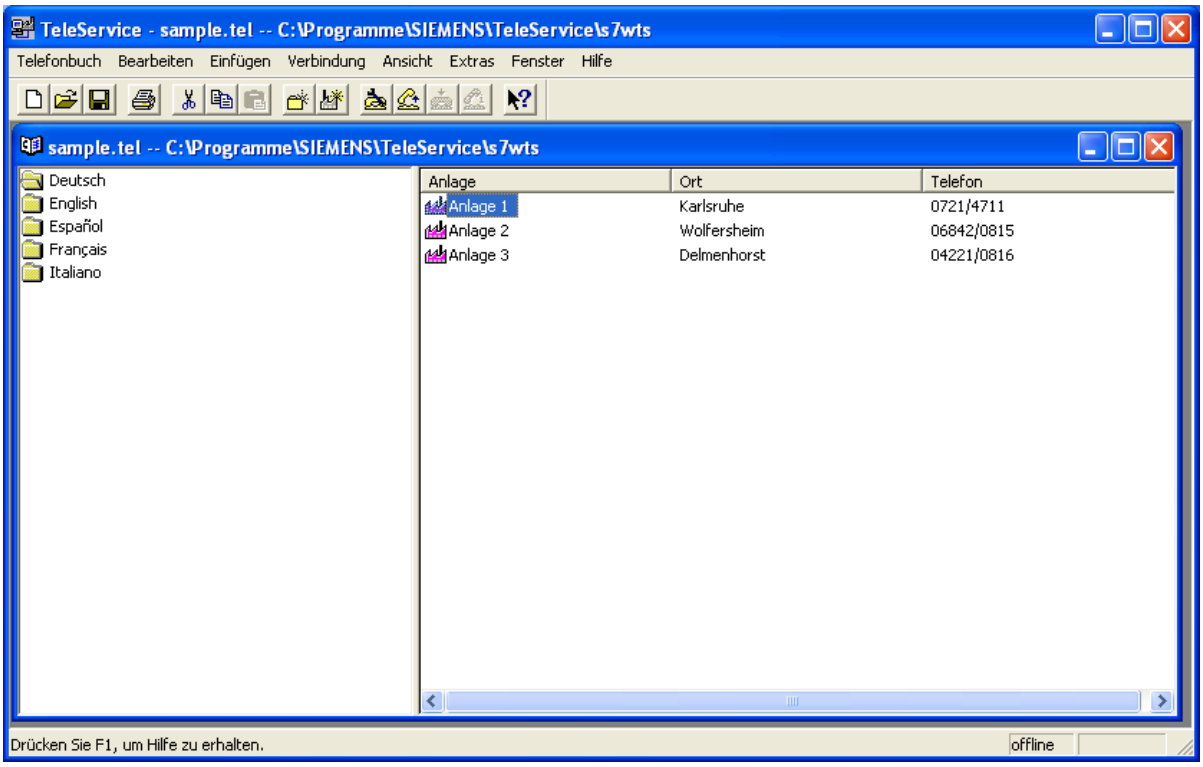


Fig.: TeleService of SIEMENS

At this point a new system must be created and configured in the SIEMENS Teleservice software.

It is important that the Teleservice Adapter is selected in the **Set PG/PC interface** dialog:

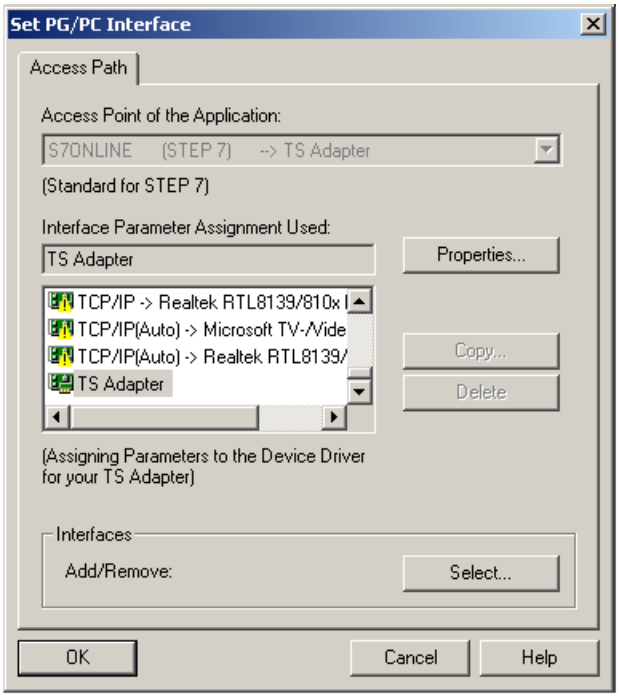


Fig.: The "Teleservice" entry must have been selected.

Now you can initiate the connection in the Teleservice software by means of menu item ***Connection->Connect***.
Set the target in WinPLC7 to **Simatic®-Net**:

Fig.: Target set to Simatic®-Net.

When the connection has been established the S7 PLC is accessible via the phone line. All the functions in menu "PLC" are active.

The connection can be terminated by means of menu item ***Connection->Terminate*** in the SIEMENS Teleservice software.

24 Communicating with a PLC via the MAC address

S7 PLCs that have not been configured are usually not accessible via Ethernet. However, since the MAC address is defined in the hardware, it is possible to access the S7 PLC using the MAC address.

The following requirements must be satisfied:

1. The target in WinPLC7 is configured as "**Target: TCP/IP direct**".
2. A network connection exists between the PC and the PLC.
3. The "WinPCap" driver is installed.
   (The installation routine is available in the WinPLC7 directory:
   WinPcap_4_1_1.exe or similar)

Open menu item "PLC->Accessible nodes". The dialog "Accessible nodes via Ethernet" is displayed:

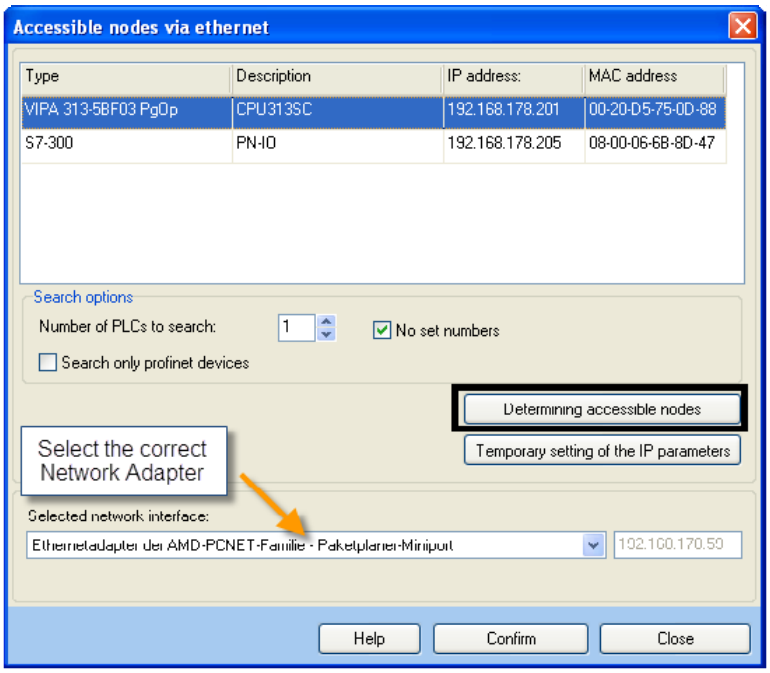


Fig.: Accessible nodes via Ethernet.

Important:
WinPLC7 must be started with **administrator rights** to be able to locate the nodes on the network.

Press the button "**Determine accessible nodes**". After a short pause all the nodes will be displayed in the window. The IP settings of the module can be changed by means of the button "**Temporary setting of IP parameters**". **However, these settings are temporary and will be lost when the power is removed from the module.**
When you have entered the IP settings and transferred them to the module you can access the module as usual via Target "TCP/IP direct".
The “Help” button of the dialog displays a detailed description of this function.

25 Working with the hardware configurator

WinPLC7 includes a hardware configurator that you can use to configure the SIEMENS S7-300® systems and the systems 100V, 200V, 300V, 300S of VIPA.

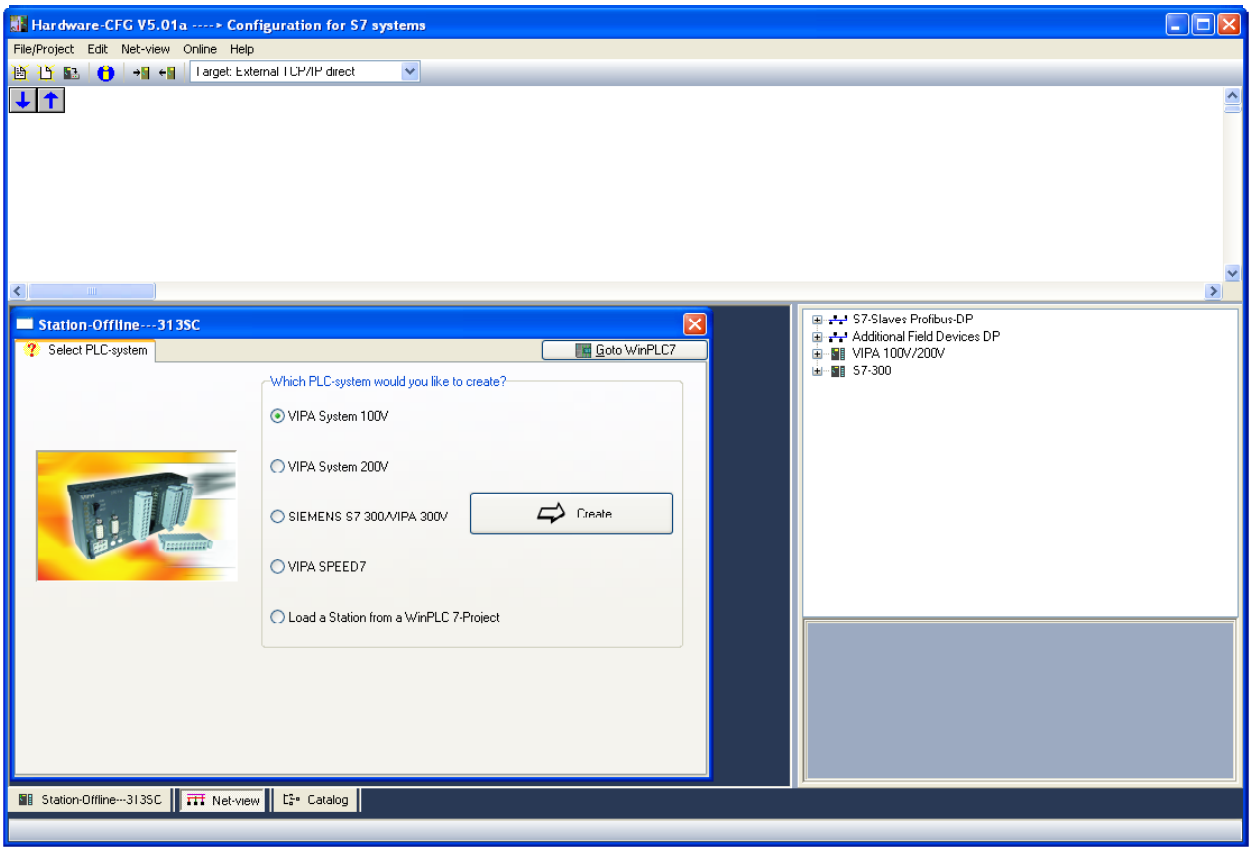
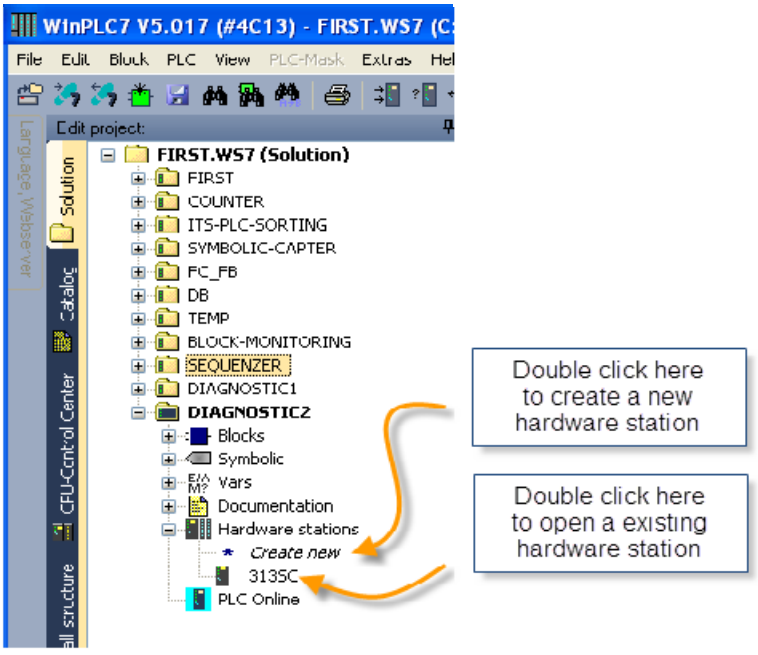


Fig.: The hardware configurator of WinPLC7

You can start the hardware configurator in the project tree. You can create a new hardware station or open a existing hardware station:



25.1 Selecting the system

If you create a new station, a system selection dialog is displayed:

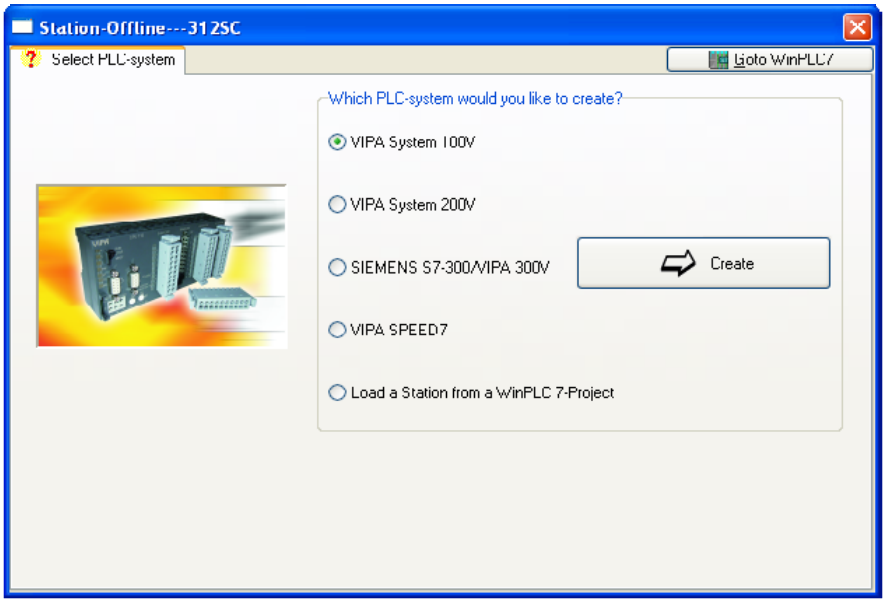


Fig.: Selecting the PLC system

The hardware configurator can be used to configure the following PLC systems:

Selection	System
VIPA System 100V	System family "100V" of VIPA GmbH
VIPA System 200V	System family "200V" of VIPA GmbH
Siemens S7-300®/ VIPA 300V	System family S7-300® of SIEMENS or system family "300V" of VIPA GmbH
VIPA SPEED7	System family "S7-300S" (SPEED7) of VIPA GmbH

25.2 Example: Configuration of a SIEMENS S7-300® CPU

The first example explains the configuration of a CPU 313C. In addition to the CPU, two digital input modules and two digital output modules must be installed and configured. A power supply module provides the necessary voltages to the CPU. The CPU is equipped with on-board peripherals, i.e. the CPU enclosure includes digital and analog inputs and outputs. We will use the digital I/Os as well as two analog inputs that must be configured for a range of "0-10V".
The following settings are also necessary:

- The bit memory byte of the CPU must be set to MB100.
- A time-of-day interrupt must be defined that is triggered on a daily basis, starting with the 31.5.2006 (May 31, 2006) at 12:30.
- A write protect feature must be configured on the CPU to protect the blocks against unauthorized overwriting.

Select an external target after you have opened a solution and the respective project. For example, select "Target: RS232".

Fig.: WinPLC7 external access settings

Then you start the WinPLC7 hardware configurator from the project tree by creating a new hardware station.

The stations created in the hardware configurator are always associated with the current WinPLC7 project. I.e. when you change to another project in WinPLC7, you should also close the hardware configurator and open it again in the new project.

After the hardware configurator has started the “Select PLC-System” dialog is displayed. Here you select "SIEMENS S7-300/VIPA 300V" and you press the “Create” button. As a result a new station is created with an appropriate rack for the S7-300. Slot 2 is selected. In the catalog the location "CPU" of the "S7-300" family is selected.

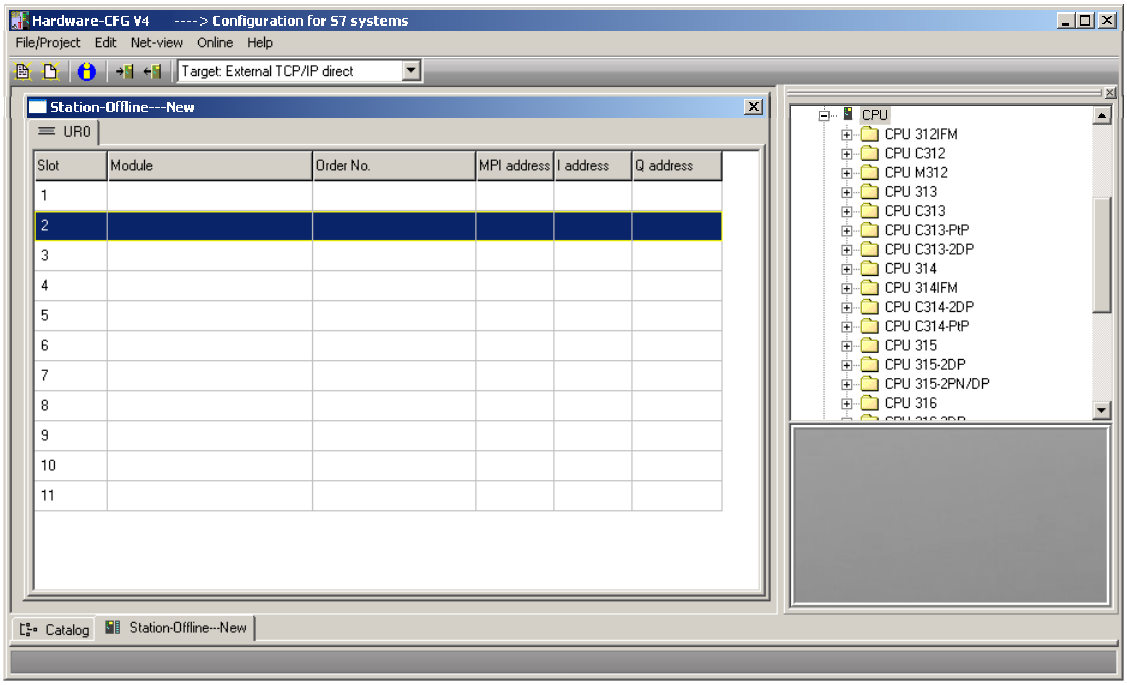


Fig.: Empty station with S7-300 rack

The station has 11 slots and the name of the rack is UR0.
You might assume that you could install any S7-300® modules in the rack. However, this is not true since certain limitations apply to the order in which the modules are installed.
These are:

- Slot 1 is reserved for the power supply module (PS module). This slot must remain empty if the rack is not equipped with a PS module.
- Slot 2 is reserved for the CPU module.
- Slot 3 is reserved for so-called IM modules that are required to interconnect several different racks. Slot 3 must remain empty if the rack is not equipped with an IM module.

It is obvious that these limitations reduce the number of available signal modules (digital input modules, digital output modules, etc.) to a maximum of 8 per rack. A separate example explains how a larger number of signal modules can be accommodated.

25.2.1 Installing a PS module in the rack

Start with slot 1 where the PS module must be installed. Select the slot in the rack.

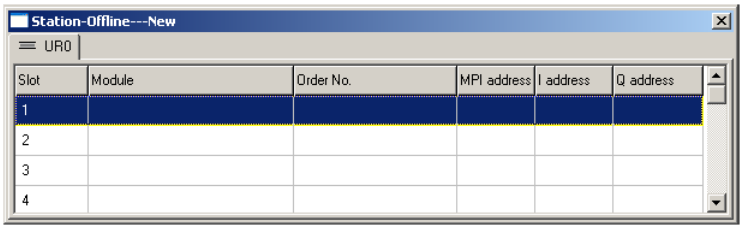


Fig.: rack with slot 1 selected

Select the required module in the catalog of modules; this is located in the category "PS-300".

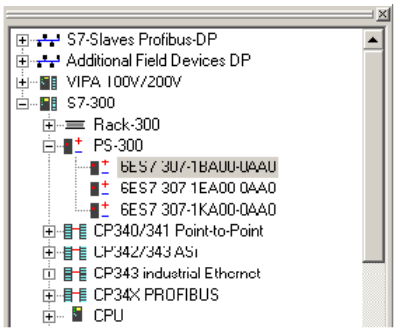


Fig.: catalog of modules with the selected PS module

When you select the module in the catalog, the appropriate slots for the module are highlighted in color. In case of the PS module this means that slot 1 is colored in, because this module can only be installed in this slot.
A double-click selects the module in the catalog and transfers it into the rack.

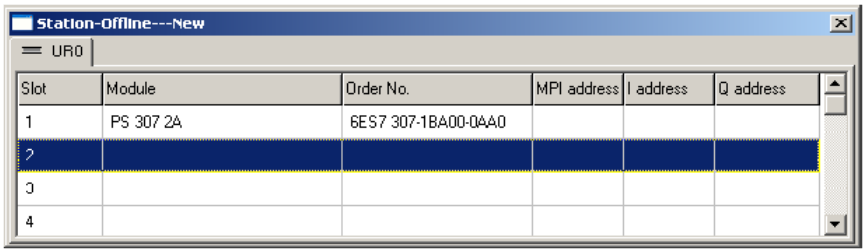


Fig.: rack with inserted power supply module (PS module)

25.2.2 Installing the CPU module

After the installation above slot 2 is selected as the active slot. Install the CPU into this slot. At the start of the example we stated that a CPU 313C would be used. For this reason we select this CPU in the catalog of modules.

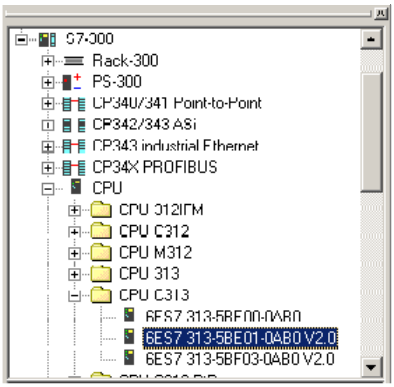


Fig.: catalog with CPU C313

When you click on the CPU in the catalog slot 2 is highlighted in color since this is the slot that is reserved for the CPU. A double-click selects the module in the catalog and transfers it into the appropriate slot.

The figure below shows the rack after the CPU was inserted. This figure also shows a number of additional slots that were inserted when the CPU was inserted. These are the slots numbered 2.2, 2.3 and 2.4.

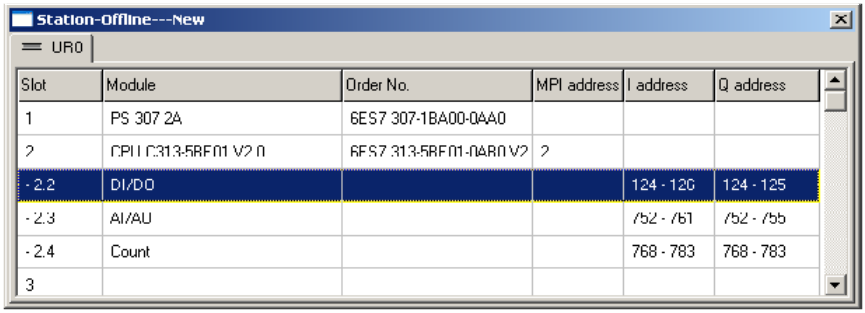


Fig.: rack with the inserted CPU

The additional slots belong directly to the CPU, because a CPU of the type 313C equipped with on-board peripherals. In our case this means that the CPU has three digital input bytes, two digital output bytes, five analog inputs and two analog outputs. The CPU also features fast counters that could, for instance, be used as period counter and for pulse width modulation purposes.

The CPUs that are equipped with this type of on-board peripherals are referred to as compact CPUs and in the catalog these are identified by the letter "C" (e.g., C313).

25.2.3 Inserting digital input and output modules

In the next step we will insert the first one of the two digital input modules into the rack. The rules for the installation of modules that were mentioned at the beginning of this example state that these modules may be installed starting with slot 4. For this reason slot 4 in the rack is selected the active slot. Simply click on the respective slot with the mouse.
The module is located in the category "DI-300" and its order number is "6ES7-321-1BH01-0AB0".

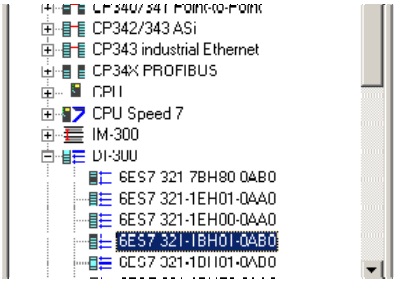
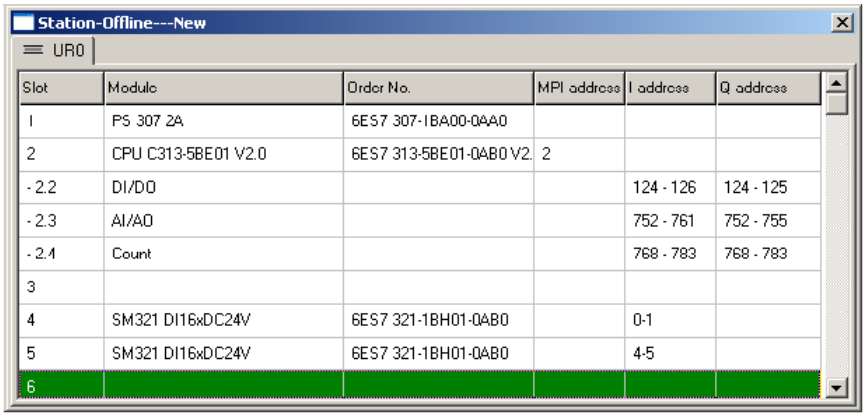


Fig.: DI module in the catalog

When you select this module slots 4 to 11 are highlighted in color to indicate that this module can be inserted into these slots.
A double-click with the mouse inserts the module into the rack and activates the next slot. Since we require two digital input modules of the same type we can execute another double click. As a result slots 4 and 5 are now equipped with digital input modules.
At this stage the rack appears as follows:



Slot	Module	Order No.	MPI address	I address	Q address
1	PS 307 2A	6ES7 307-1BA00-0AA0			
2	CPU C313-5BE01 V2.0	6ES7 313-5BE01-0AB0 V2.	2		
- 2.2	DI/DO			124 - 126	124 - 125
- 2.3	AI/AO			752 - 761	752 - 755
- 2.4	Count			768 - 783	768 - 783
3					
4	SM321 DI16xDC24V	6ES7 321-1BH01-0AB0		0-1	
5	SM321 DI16xDC24V	6ES7 321-1BH01-0AB0		4-5	
6					

Fig.: rack with both digital input modules

Only the two digital output modules are still required. The first module is inserted into slot 6. In the catalog the digital output modules are located in category "DO-300". The module required for the example has the order number "6ES7-322-1BH01-0AA0" and it is shown below.

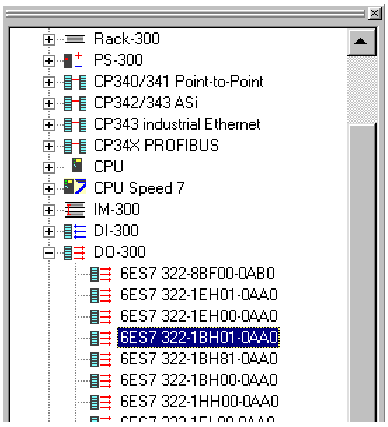


Fig.: catalog with DO module

A double-click of the mouse inserts the module into the active slot in the rack. Since we require two modules of the same type we can execute another double click. At this stage two DO modules are installed in slots 6 and 7.

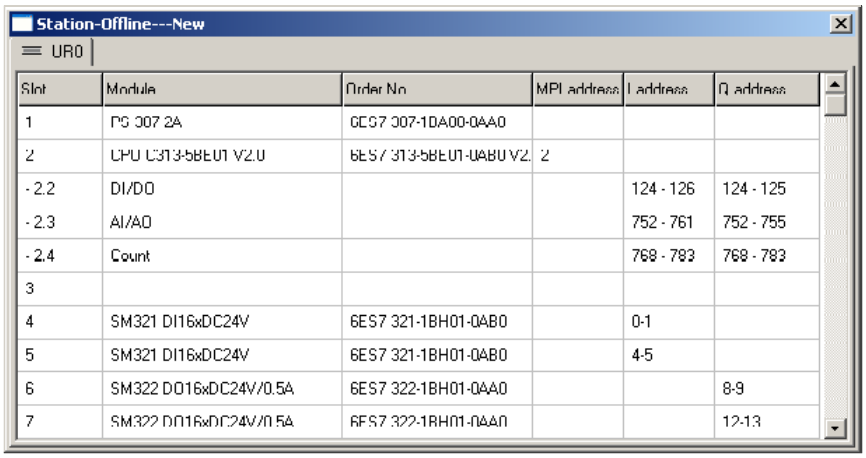


Fig.: complete rack

Now all the modules required for this example have been inserted into the rack.

25.2.4 Changing the input and output addresses

When the input modules were inserted into the rack, the start addresses of the modules were automatically determined and defined by the hardware configurator. The configurator has attempted to use addresses that would also be valid for a slot-oriented CPU. A different address will only be used if the determined address is already occupied.
As a result of this address allocation the DI module in slot 5 will have start address EB4. However, the DI module in slot 4 only occupies addresses EB0 and EB1, and therefore addresses EB2 and EB3 are still available.
The reason for this behavior is that slot-oriented CPUs generally result in 4 bytes being reserved for the slot of a digital module (16 bytes for an analog module). Digital modules can occupy a maximum of 4 bytes, which is why the reservation of 4 bytes does not require an address offset if a 16 bit module was exchanged against a 32 bit module.
These gaps in the addresses interfere with the programming and therefore we try to eliminate the gaps when non-slot-oriented CPUs are used.
We also want to apply this to the example. The start address of the DI module in slot 5 should be changed to EB2. Execute a double click on slot 5. As a result the "DI-200 module properties" dialog appears.
The "Addresses" tab of the dialog contains a field for the start address of the module.
This field is write-protected as long as check box "System selection" is active. You can change the start address if you remove the tick mark in this tick box.

In the example we enter start address 2.

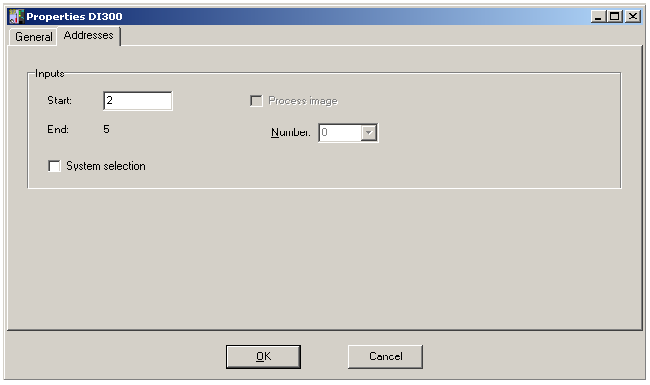


Fig.: dialog with modified start address

Click OK to confirm this dialog box. As a result the modified rack will be displayed. The modified address space is displayed in the slot of the module.

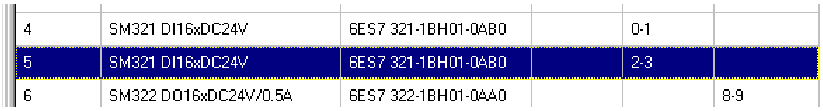


Fig.: Slot 5 in the rack with the modified address space.

The input modules occupy bytes EB0 to EB3 without any gaps.
The addresses of the output modules must now be changed in the same manner. The address on the bracket shows that the base address of the module in slot 6 starts with AB8. This address must be changed, because we want to start with a base address of zero and we want make sure that the address space does not include gaps.
Execute a double click on slot 6. Dialog "DO300-Module dialog" will be displayed which is similar to the dialog for the DI-300 modules. The "Address" tab of Dialog DO-300 also contains a field with the start address. As soon as you remove the tick mark from "System selection" you can change the contents of the address field. In the example we enter the address 0 and exit from the dialog with a click on OK.
The new address will now be displayed on the rack. Slot 6 occupies addresses AB0 and AB1.
We will now change slot 7. Execute a double-click on this slot and change the address to AB2. When you have confirmed your entries slots 6 and 7 appear as follows:

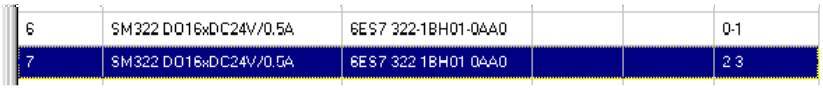


Fig.: Slots 6 and 7 with modified base addresses

The I/O modules that we have installed occupy the required address space. However, the digital inputs and outputs of the on-board peripherals of the CPU 313C have a default address that starts with 124. This should also be changed.
The approach is the same as for the "normal" slots. Execute a double-click on the slot with the name "-2.2". The configuration dialog will be displayed. In addition to the address settings, the "DI/DO300 Module dialog" offers extra configuration options. The dialog contains an "Inputs" tab that provides various input settings.

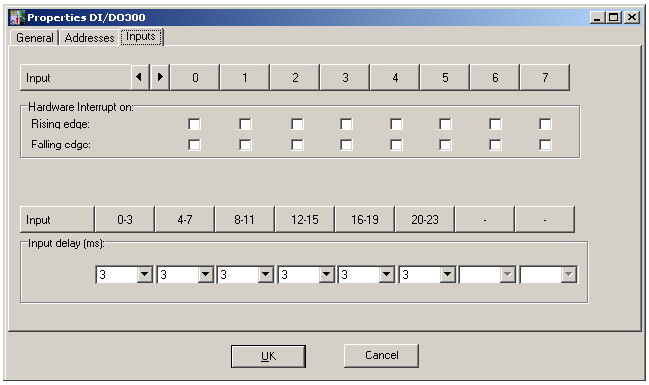


Fig.: configuration versions of the on-board inputs

For example, you can specify that the inputs should be used as alarm inputs and to trigger a process alarm on a rising or falling edge. The operating system of the CPU can access and analyze this alarm in the respective alarm-OB.
Here you can also define input delays.

Depending on the CPU type, these settings may vary; a description of the respective settings is available from the CPU reference manual.
The basic settings in the example should not be changed; we only want to adapt the addresses in accordance with our requirements. For this reason we select the "Addresses" tab of the dialog. Note that you can access the input as well as output addresses via this dialog. The reason is that this module is a combination of an input and an output module, i.e. the module is equipped with inputs and outputs.

You are already aware of the necessary procedure, i.e. remove the tick from the "System selection" tick box for the inputs and the outputs. Enter starting address 4 for the inputs. The same applies to the output addresses. Click OK to confirm your entries.

Fig.: the altered addresses of the on-board inputs and outputs

These changes have specified that the input address space is EB0 to EB6 and the output address space is AB0 to AB5. The address space does not contain any gaps.

You could have left the original addresses unchanged; however a continuous address space without gaps simplifies programming.

25.2.5 Analog input configuration

The next step involves the configuration of the first two channels of the analog input of the CPU which must be adjusted to the required input range.
Open the configuration dialog with a double-click on slot "-2.3".
The "Properties AI/AO300" dialog will be displayed. This dialogue has 4 tabs with that provide general information on the on-board module as well as access to the address settings and the configuration settings of the input and output ranges.
In our example the first two channels of the analog inputs must be set to a range of "0..10V". Unused channels must be disabled.
Open the "Input" tab to enter these settings. In this dialog tab you can select the required measurement range, provided that the measuring ranges of the different channels are independent of each other.

The type of measurement is preset to "U" for voltage; therefore you must merely set the measurement range of channels 0 and 1 to "0..10V".

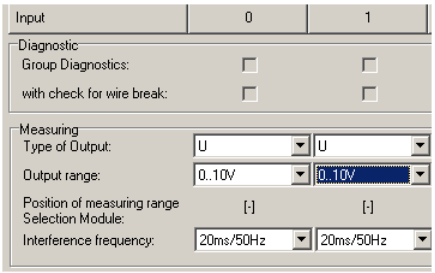


Fig.: extract from the dialog with the configured measurement range for channels 0 and 1

Channels 2, 3, and 4 are not used at present; you should therefore deactivate these channels. This reduces the conversion time required for these channels. To deactivate these channels you simply select "..." for the "Type of" measurement of the respective channel. During execution the measurement range is automatically set to the same entry. Once confirmed, the channels are displayed as follows:

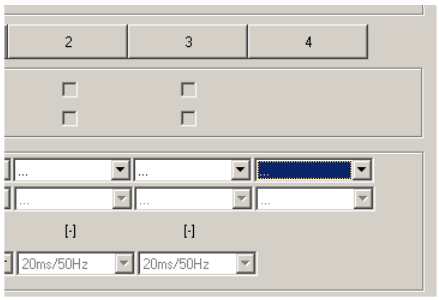


Fig.: deactivated input channels 2, 3 and 4

Let us now proceed to the outputs that are currently also not being used. For this reason they should also be deactivated. Open the “Outputs” tab of the dialog. As with the deactivation of the input channels, the output types are also set to "...". As a result the output range is also changed to this setting.

Fig.: deactivating the output channels

This completes all the settings that are required for the analog channels; you can close the dialog by pressing the "OK" button.

The properties of the counter inputs can be also be modified in the same way as the properties of the on-board digital and analog inputs or outputs. Simply double-click slot "-2.4" and enter the required settings into the dialog. Refer to the CPU reference manual for a detailed description of the settings.

25.2.6 Configuring the properties of the CPU

We can proceed to enter the settings for the CPU when the properties and the addresses of the signal modules have been defined in accordance with our specifications.
According to concept the clock memory of the CPU should be set to clock memory byte MB100. In addition, a time-of-day interrupt must be defined that is triggered on a daily basis, starting with the 31.5.2006 (May 31, 2006) at 12:30. And finally the CPU should be protected with a password to guarantee that the CPU is write protected.

Double-click the slot of the CPU to open the "CPU properties" dialog. The appearance of the dialog may vary, because the different CPUs offer different settings, depending on their performance. It is obvious that the available range of settings is vast. These settings have been divided thematically in accordance with the tab names that are used as headings. For example, the "Start-up" tab contains the CPU settings that concern the start-up procedure (change from CPU-stop to CPU-run). The different descriptions of the options are usually self-explanatory. Please refer to the reference manuals of the CPU for detailed descriptions.

Clock memory

We will start by specifying the clock memory settings. The clock memory byte is a setting where the different bits of the clock memory byte change their state at different frequencies. With a clock memory byte it is very simple to implement a flashing light (1.25Hz). The following table shows the frequencies of the different bits:

Bit	0	1	2	3	4	5	6	7
Frequency in Hz	10	5	2.5	2	1.25	1	0.625	0.5

Select the tab "Cycle/clock memory" for this setting. This dialog contains the settings for the clock memory. Start by activating the "Clock memory" option. Then you can enter the address of the clock memory byte into the address field. The available clock memory area depends on the CPU. For example, if you should enter the address 500 you would receive the error message before the display changes to another dialog page. The error message states this address is not located in the valid range since the CPU only supports clock memory bytes 0 to 255.
The example should use clock memory byte MB100, i.e. you must enter the address 100. In the PLC program you should remember that this clock memory byte can only be read, because any write access would generally be overwritten by the operating system of the CPU. As a programmer you should normally always use the same address for the clock memory byte to avoid this source of errors as far as possible.
Here follow the settings within the dialog:

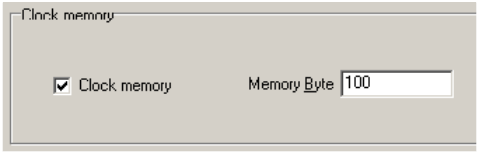


Fig.: Setting the clock memory byte to MB100

Parameter settings for the time-of-day alarms

The settings for the time-of-day alarm are entered in the tab with the same name. Here you can see that the CPU only supports a single time-of-day alarm. Here OB10 was called by the operating system of the CPU when the alarm was triggered.
A priority is required, when you suspect that another alarm with higher priority could be processed when this alarm has occurred, i.e. OB10 is only started after the high-priority alarm was processed. If this is a problem, then you should modify the priority classes accordingly. However, this is not an option that is supported by all CPUs; the CPU 313C employed in the example does not support this modification. This does not have any consequences for our example since we have not defined any additional alarms.
The time-of-day alarm should be executed daily at a certain time; therefore we select the "Every day" in the execution settings.
The starting date should be 31.5.2006 (May 31, 2006), the daily execution time is defined as 12:30. Finally, the alarm must be activated.

Fig.: Setting the time-of-day alarm

Defining write protection

The CPU should be protected with a password to prevent the PLC program being overwritten within the CPU.
Write protection provides sufficient security in this respect. Once this is activated the password is requested with every write access. Access is granted when the correct password is entered, if the password is not available, access is refused.
Even with the activated write protection it is possible to read the PLC program. To prevent this as well the Write/Read protection must be activated.

In "Protection" tab contains the elements that are necessary to define password protection. Select "Write-protection" as the protection option and enter a password with a max. of 8 characters. Repeat the password in the field "Password again" to prevent mistakes in the first entry. If these entries are not identical an error message will be displayed.

The following figure shows the relevant portion of the dialog:

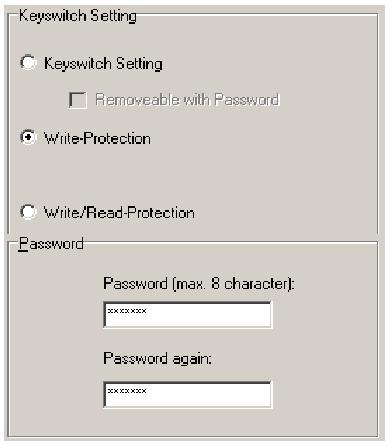


Fig.: defining write protection

This completes all the settings that are required for the CPU; you can close the dialog by pressing the "OK" button.

25.2.7 Transferring the configuration to the CPU

This concludes the entire hardware configuration. You can save the station using menu item "File/Project->Save active station" or via keyboard shortcut [CTRL] + [S].
To make the configuration active it must be transferred into the CPU. Start by selecting the required communication path which defaults to the setting that was defined as default for the WinPLC7 project. When we started the hardware configurator we selected "RS232" as destination, i.e. the serial interface. If you require another communication path (e.g., NetLink, NetLink PRO, TCP/IP-direct, etc.) you can change this setting via the following selection box.

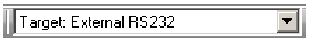


Fig.: selected communication path to the CPU

Execute menu item "Online->Send configuration". Depending on the communication path, different dialogs will be displayed to define the communication parameters of the respective interface.
The figure below illustrates the RS232 communication dialog:

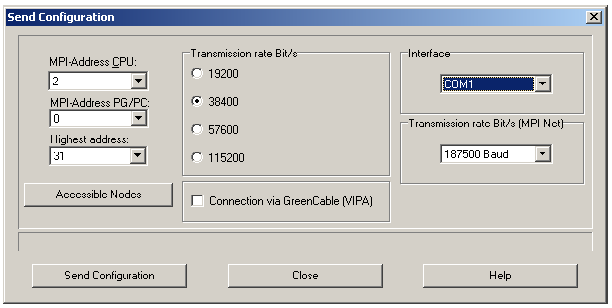


Fig.: Send configuration (RS232)

Start by selecting the transmission baud rate. Current MPI lines offer a minimum transmission rate of 38400 Baud; others also support higher rates of 57600 or 115200 Baud. Pleas refer to the reference manual for the MPI line. When the GreenCable is used with a VIPA CPU the baud rate must be set to 38400 Baud.
Then you must select the respective serial interface to which the MPI adapter was connected. If the MPI adapter consists of a USB MPI adapter that creates a virtual serial interface during the installation, you can find the number of the interface in the Windows device manager provided that the MPI adapter is inserted.

If the interface was selected correctly you can determine the MPI address of the connected CPU via the "Accessible nodes" button. The dialog "Accessible nodes" is displayed with a list of the respective MPI addresses. The figure below shows a possible display:

Fig.: a possible display of accessible nodes

The dialog shows that the CPU has an MPI address of 2; this is also the default address of the CPU when it has not been configured.
Select the line with the MPI address of the CPU and press OK to close the dialog. The MPI address selected above is transferred into the field "CPU MPI address" of the "Send configuration" dialog.
This completes the necessary settings. The configuration is transmitted to the CPU by means of the "Send configuration" button.
Close the dialog.
This also completes our work with the hardware configurator so we can also close this using menu item "File/Project->Exit".

25.2.8 Summary of example 1 in respect of the hardware configuration

The example explained how you would configure a S7-300® PLC. The rules applying to the installation of the different types of modules were detailed and it was shown, how the addresses and parameters of the signal modules should be defined.
The necessary settings in the CPU were implemented and the configuration was finally transferred to the PLC.

25.3 Second example regarding the hardware configuration

The first example only used a small number of signal modules. This meant that the available slots in the rack could accommodate all the modules. Since signal modules can only be installed in slots 4 to 11 the number of signal modules is limited to a maximum of 8 per rack. This covers the requirements of a large number of systems; however, it is also possible that a larger number of signal modules are required. This means that additional racks are necessary.
Not all S7-300® CPUs provide support for multiple racks. The CPUs reference manual can provide information whether a CPU is suitable for the operation with multiple modules, or you could refer to the brief description of the different CPUs in the catalog of modules that is available in the hardware configurator. The figure below shows such a display:

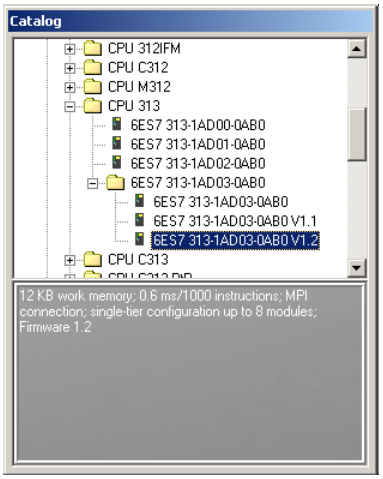


Fig.: catalog of modules with a brief description of the selected CPU

The brief description of the CPU indicates ".. single-tier configuration up to 8 modules". From this you can infer that this CPU only supports a single rack and it is therefore not suitable for multi-rack systems.

The following figure shows another CPU display:

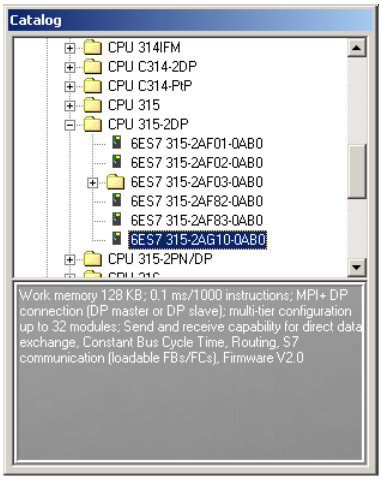


Fig.: catalog of modules with a brief description of the selected CPU

This brief description indicates ".. multi-tier configuration up to 32 modules". This means that this CPU supports multiple racks.

The following example will employ the CPU 315-2DP indicated above. A PS-module is installed into the rack and, in addition, two analog input modules and four analog output modules will be used. We also require three 32-bit digital input modules and four 16-bit digital output modules.
The number of racks required therefore results from the number of the signal modules. The 13 signal modules require two racks.

25.3.1 Starting the hardware configurator

Create a new project In WinPLC7. Continue by selecting an external target (e.g. RS232) and create a new hardware station in the project tree. The hardware configurator is started and displays an empty station window.
The first steps are the same as described in the previous example. Start by selecting the family "SIEMENS S7-300/VIPA 300 V" as the system that should be created. Press the "Create" button.

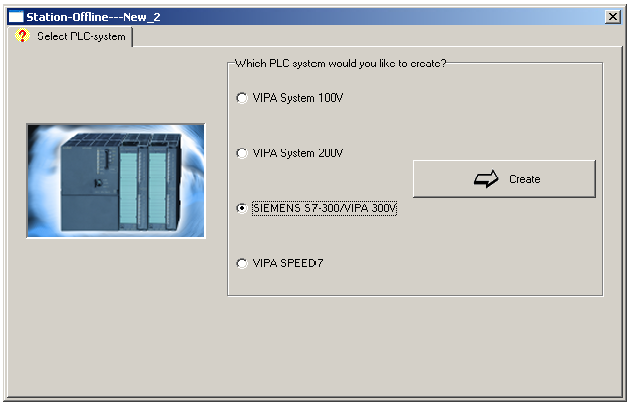


Fig.: new station with the selection of the system family that must be created

The S7-300 rack is inserted into the station and the 300-type CPUs are preselected in the catalog of modules.
In the next step we install the PS module. This module must be installed in slot 1. For this reason we select this slot in the rack and insert the PS module with a double-click.

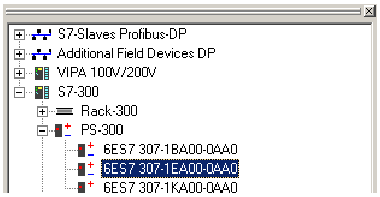


Fig.: PS module from the catalog

The CPU must be installed into slot 2. In the example we use the CPU 315-2AG10.

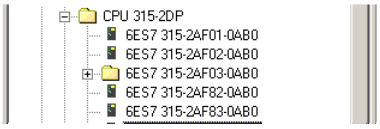


Fig.: the CPU in the catalog

After these activities the rack appears as follows:

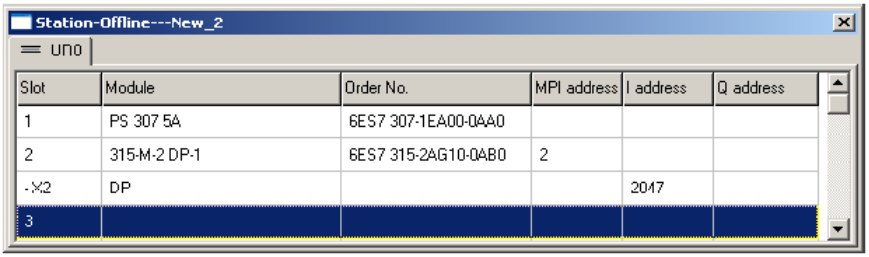


Fig.: station with PS module and CPU installed

In contrast to CPU 313C that was used in the previous example, the CPU 315-2-DP does not have any on-board peripherals.

Slot 3 is reserved for the IM module. In the previous example this slot remained empty, because IM modules are required to interconnect multiple racks. The previous example employed a single rack; therefore no IM module was necessary.

This is different in this example. Here we have already recognized that two racks are necessary to accommodate the necessary number of signal modules.
The IM modules provide the connection between the backplanes bus of the different racks the CPU employs to communicate with the signal modules. If you inspect category "IM-300" in the catalog of modules you will note that several different versions of IM modules are available. These are as follows:

- IM360: this is the interface that can be installed in rack 0 (UR0). This module establishes a connection between rack UR0 and another rack that must be equipped with the IN361. The maximum distance permitted between an IM360 and an IM361 is 10m.
- IM361: the counterpart to the IM360. The IM361 is used in racks 1 to 3 (UR1 to UR3). This module has one IN and one OUT interface; the OUT interface can be used to provide the connection to the next IM361 interface. The distance to the next interface may not exceed 10m.
- IM365: this interface provides the connection between rack 0 (UR0) and rack 1 (UR1). The IM365 consists of a sender that is installed on UR0 and a receiver for UR1. It is not possible to extend the connection, e.g. to a UR2. The IM365 can be used when two racks provide sufficient space. The K-bus is not linked to the UR1, i.e. certain CP modules cannot be installed in the UR1. Also remember that the power supply is limited to 1.2A and that each individual rack must not exceed a max. load of 0.8A.

Because we only need two racks in the example we are able to use an IM365. The figure below shows the module in the catalog.

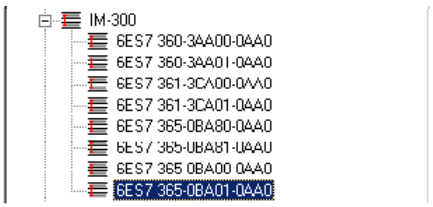


Fig.: utilized IM modules

A double-click inserts the IM into slot 3 that was preselected automatically in the station.

Now the rack can be extended with an additional rack. To install this rack, we double-click the entry for the rack the beginning of category "Rack-300".

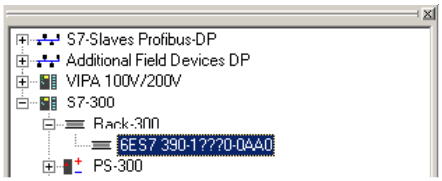


Fig.: the rack to insert the second rack

When this operation has been completed you may install another rack. If you confirmed this request with "Yes" the installation will take place. The upper section of the station shows another tab with the name UR1.

Fig.: new rack UR1

If you activate this rack with a click on the tab UR1 an additional 11 slots will be displayed.
This rack is subject to the same rules as UR0. This means that signal modules can only be installed from slot 4 onwards. Slot 3 is reserved for the IM module. Normally, slots 1 and 2 are not used, or slot 1 can be used to install a PS module.
In UR1 we must now install the counterpart to the IM-365 of UR0. Start by selecting slot 3 and then select the IM365 in the catalog of modules. You must make sure that this is the same module as installed in UR0.

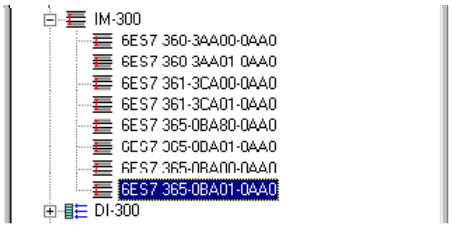


Fig.: IN-365 for UR1

UR1 will appear as follows after the IM has been installed:

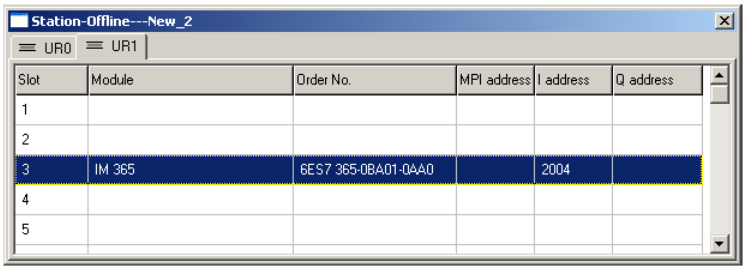


Fig.: UR1 with IM module

This completes the installation of the interfaces and we can begin to install the signal modules into carrier UR0.
Change to UR0 and select slot 4. Two analog input modules must be installed into this slot and the next slot. These modules are located in category "AI-300" which is shown below.

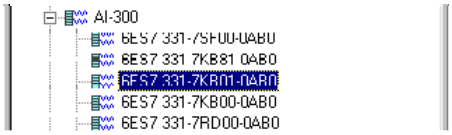


Fig.: the utilized analog input modules

Insert the two modules one after the other using two double-clicks. Now slot 6 is the active slot. From this slot onwards we must install 4 analog output modules.

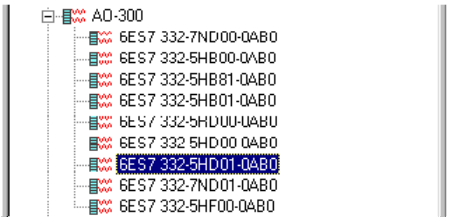


Fig.: analog output modules

These can also be installed directly one after the other.
This means that two unused slots for signal modules remain in UR0. These are slots 10 and 11 where we must install digital input modules.

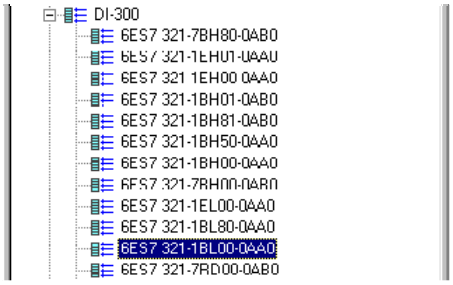


Fig.: the digital input modules

When the DI modules have been installed you should change their respective addresses. Double-click slot 10 and set the start address to 0. This allocates input addresses EB0 to EB3. For this reason the start address of the module in slot 11 is set to 4.

The resulting image of the rack appears as follows.

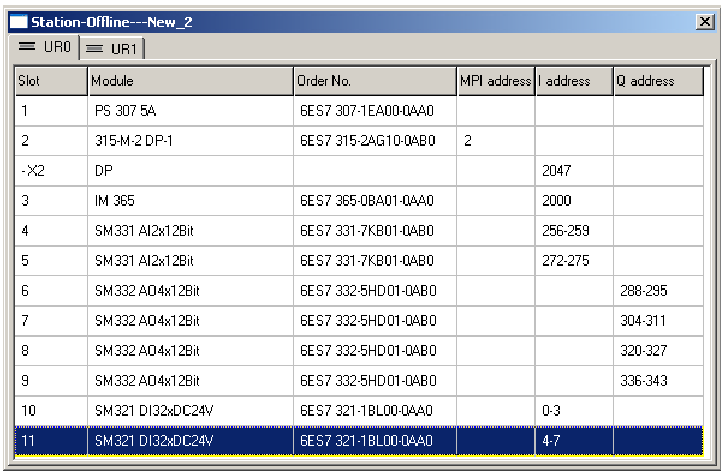


Fig.: organization of UR0

At this stage another 5 modules are required which must therefore be installed in rack UR1, starting with slot 4. For this reason we select rack UR1 and activate slot 4.
Here we install another DI module of the type with start address 8.

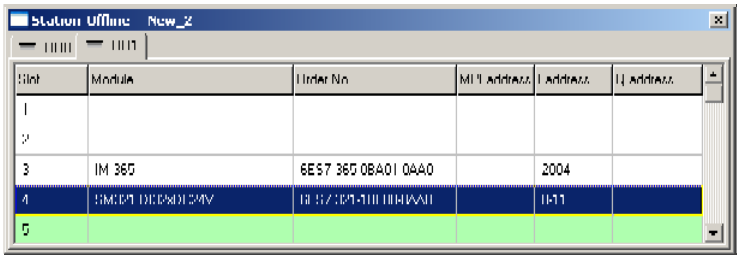


Fig.: the first signal module in rack UR1

Finally we must install the remaining DO-300 modules in slots 5 to 8. These addresses must also be modified, starting from zero. No gaps should exist in the address space.
The figure below shows the type of module that must be used.

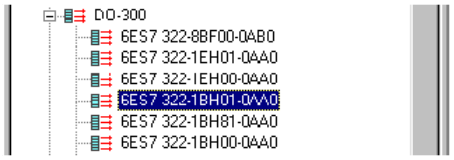


Fig.: the digital output modules that must be used

These modules are installed one after the other into the rack and then the start addresses are modified by means of the configuration dialog. The following figure shows the rack after these actions.

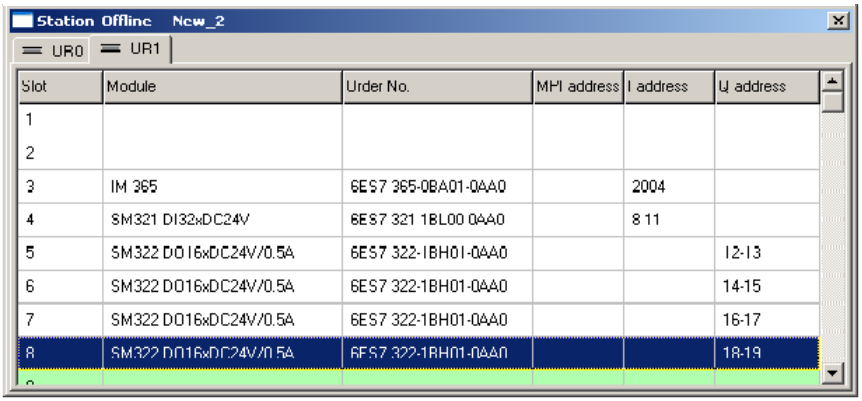


Fig.: the completed rack UR1

This completes the configuration and the station can be saved by means of keyboard shortcut [CTRL] + [S].

When the configuration has been transferred to the CPU the hardware configurator is closed. The procedure to transfer the configuration is available from the previous example.

25.3.2 Summary of example 2 in respect of the hardware configuration

The second example demonstrates the use of multiple racks. Multiple racks are required when you require more than the 8 slots that are available for the signal modules in rack UR0. Not all CPUs provide support for several racks.
The chapter also introduced the IM modules which are required to interconnect the different racks. The different types of IM modules and their applications were explained.

25.4 Example: Configuration of a VIPA 100V CPU 114

To configure a VIPA CPU114 select "VIPA System100V" in the system selection tab:

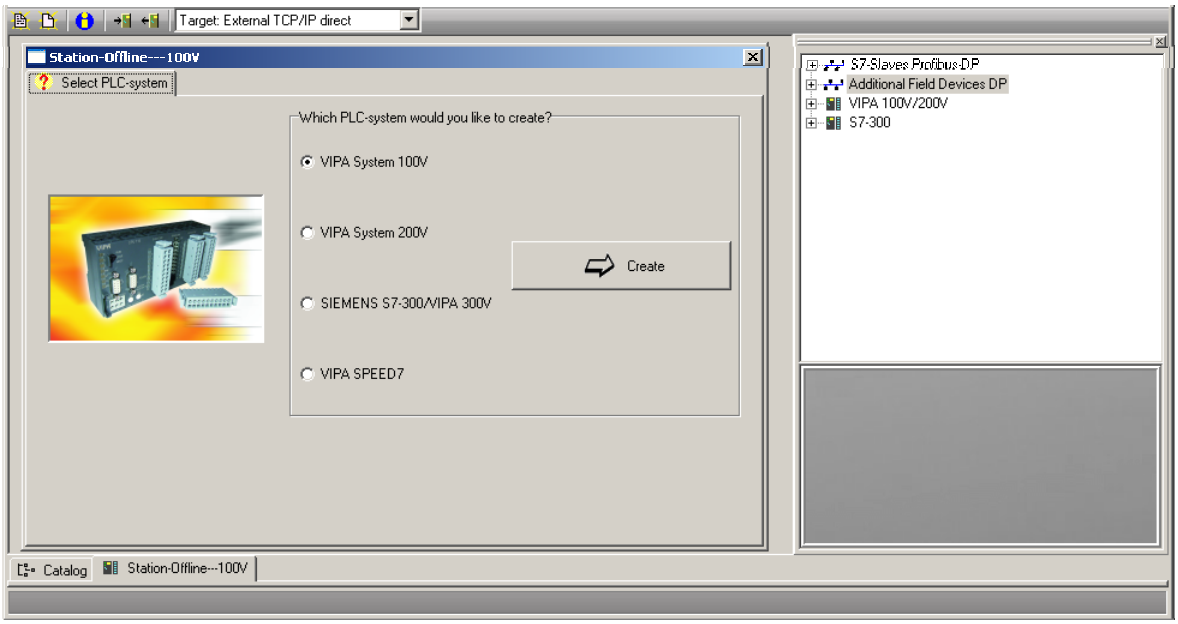


Fig.: configuration of a CPU114 of system100V family

When you press the “Create” button, a new station is created and the CPUs of the 100V family are opened in catalog on the right:

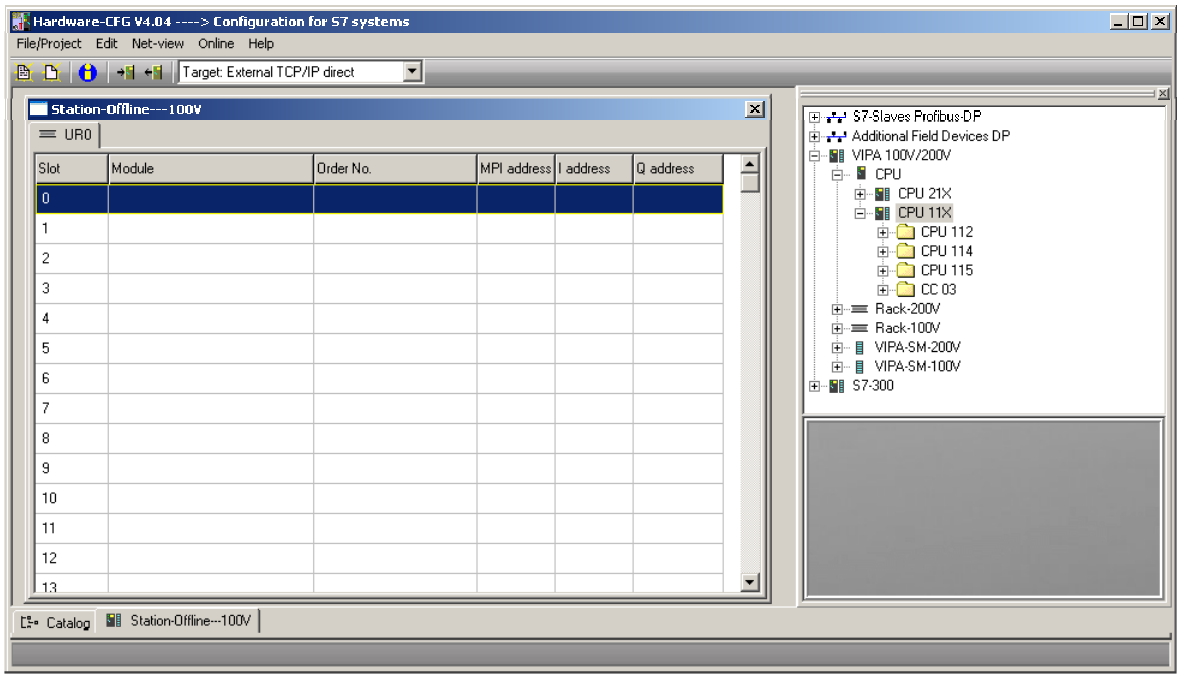


Fig.: station for CPU114

Now the CPU can be installed in slot 0. Open the node "CPU114" and double-click the entry "114-6BJ02" .
The CPU is now located in slot 0:

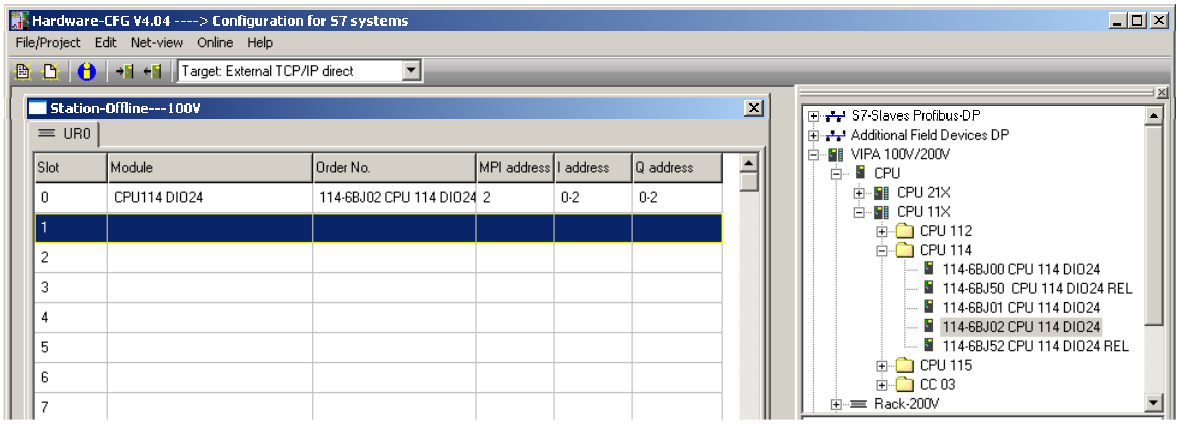


Fig.: CPU 114 is located in slot 0

You can now change the settings of the CPU. Double-click slot 0 to edit the CPU properties. The "CPU properties" dialog appears:

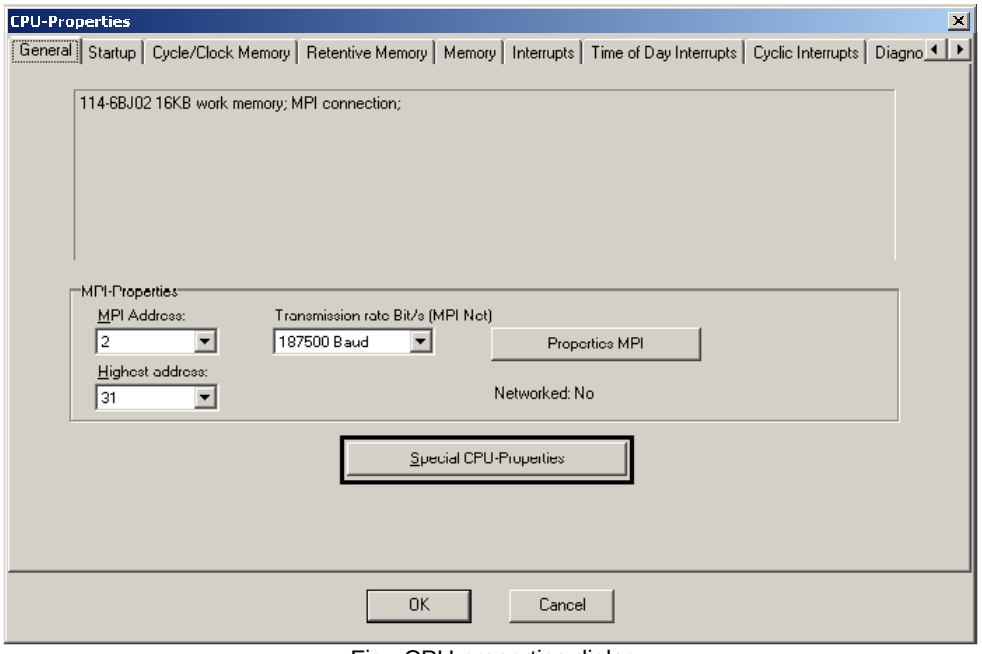


Fig.: CPU properties dialog

The properties of the general CPU are accessible via the different tabs.

You can configure the integrated inputs and outputs of the CPU by means of the button "Special CPU properties":

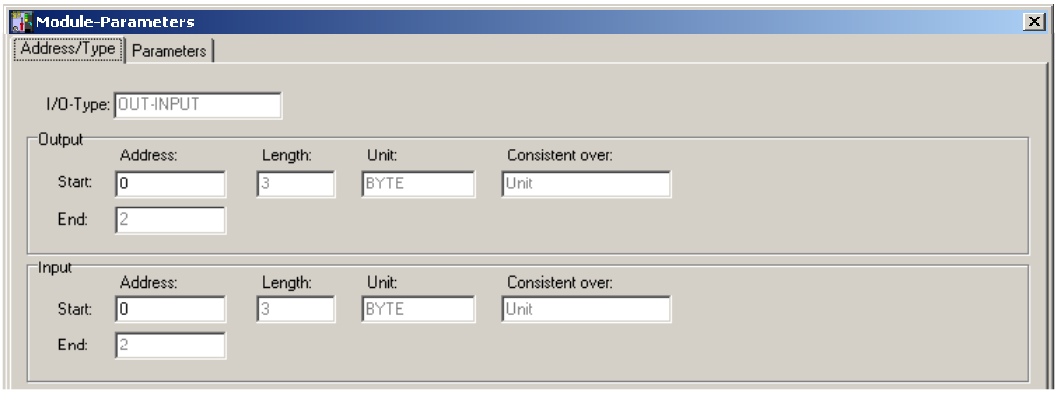


Fig.: defining the addresses of inputs and outputs

Press shortcut keys CTRL+S to save the station.

Now you can transfer the configuration to the CPU. For this purpose you select menu item *Online->Send configuration.*

25.5 Configuration of VIPA SPEED7, VIPA 200V, Profibus-DP-configuration and Ethernet configuration

The online help of the hardware configurator contains additional examples with respect to:

- the configuration of VIPA SPEED7
- VIPA 200V
- Profibus-DP configuration
- Ethernet configuration

These examples are available via the "Help" menu tree of the hardware configurator.