

WinPLC-Engine

Version 1

User Manual - English

English

Software-PLC for SIEMENS STEP[®] 7 using TCP/IP



WinPLC-Engine user manual
All rights reserved.

The **WinPLC-Engine** application is subject to a software licensing agreement and shall only be used in accordance with the terms of this contract.
The license agreement is displayed during installation and the user must agree to the terms and conditions of the contract.

Trademarks
STEP®, SIMATIC®, S7-300®, S7-400® are registered trademarks of SIEMENS AG.
Revision level: 02/2010



Contents

1 Introduction	4
1.1 What is WinPLC-Engine ?	4
1.2 Applications	6
2 Installation	7
2.1 System requirement	7
2.2 Installation	7
2.3 Uninstall	7
3 Getting started	8
3.1 The STEP®7 software configuration	8
3.2 Loading mask	11
3.3 Operating the masks in cycle mode	11
3.4 Creating custom masks	13
4 User interface	20
4.1 Outline	20
4.2 Context menu	21
4.3 Keyboard shortcuts	21
5 Object	23
5.1 Text objects	23
5.2 Dynamic text objects	24
5.3 LED object	26
5.4 Analog input and output	27
6 Motion profiles	29
7 Configuration of the Software-PLC of WinPLC-Engine	30
7.1 Configuration with the WinPLC7-Engine dialog	30
7.1.1 PLC-Type	31
7.1.2 Net-Interface	31
7.1.3 Cycle	31
7.1.4 Remanence	31
7.1.5 Alarms	31
7.1.6 Time Alarms	31
7.1.7 Cycle Alarms	31
7.2 Configuration using STEP®7 V5.x	32
8 Settings	33
9 Technical specifications of the Software-PLC	34
10 External access to the Software-PLC	35
11 Call parameters	37

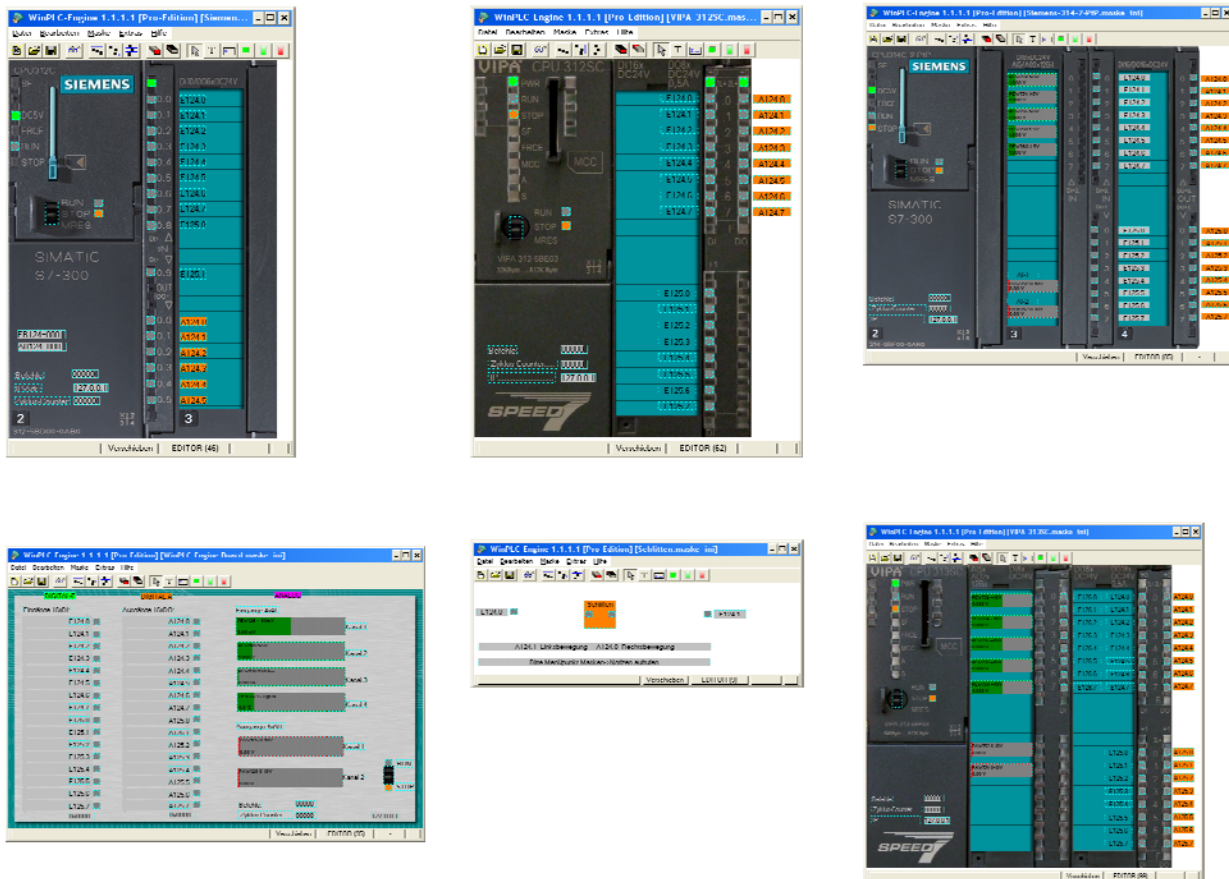


1 Introduction

1.1 What is WinPLC-Engine?

WinPLC-Engine is a Software-PLC for **STEP®7 programs**. Access to the Software-PLC is provided by TCP/IP. For STEP®7 V5.4 a driver was developed that may be selected and activated as an adapter in the Siemens dialog "**Extras->Set PC/PG-Interface**".

The user is free to change the WinPLC-Engine display by means of different objects (LED's, text, dynamic text, analog inputs, analog outputs, graphics, background image) as required. After the installation, a large number of masks are available:



It is possible to animate objects using **motion profiles**, i.e. you can, for instance, generate a proper simulation of an elevator or lift.

The configuration of the Software-PLC may be defined by means of the configuration dialog in WinPLC-Engine or with a hardware configuration in STEP®7. If you are entering the configuration into STEP®7, you must insert the same CPU that you have selected in WinPLC-Engine from the list in the settings dialog.



Different versions of WinPLC-Engine are available, including a **freeware Version** freeware version that you may use to load and simulate a **S7-300® 312C mask**. The **Pro-Edition** can be used to create and save customized masks.

Freeware-Edition	<p>The Freeware-Edition can be used to load and simulate the 312C mask. The memory of the Soft-PLC is limited to 1 KB of RAM.</p> <p>Restriction: When "WinPLC-Engine Freeware" is connected to the "ITS-PLC MHJ-Edition" (3D system simulation) a cycle can consist of approx. 150 commands. If you attempt to simulate larger programs the Software-PLC will go to STOP mode.</p>
Standard-Edition	<p>You can use the purchased standard edition to load and simulate all available PLC masks. You can insert text into PLC masks and modify the addresses of inputs and outputs. However, you cannot insert new objects. The memory of the Soft-PLC is limited to 32 KB RAM.</p> <p>Compatible with "ITS-PLC MHJ-Edition".</p>
Pro-Edition	<p>You can use the purchased Pro-edition to load and simulate all available PLC masks. In addition, the user can create new PLC masks.</p> <p>The memory of the Soft-PLC is limited to 240 KB of RAM.</p> <p>Compatible with "ITS-PLC MHJ-Edition".</p>

The online help of WinPLC-Engine details the differences.



1.2 Applications

WinPLC-Engine is used to simulate an S7 program. The application responds like a real S7 PLC. This means that you must first transfer the different blocks into the Software-PLC before you turn on the RUN mode of the Software-PLC. Now you can use all the diagnostic options provided the programming environment:

- monitoring blocks
- monitoring variables
- module status with diagnostic buffer, I-STACK and B-STACK
- etc.

In addition, you can define the hardware configuration using the STEP[®]7 hardware configurator and transferred the result into the Software-PLC of WinPLC-Engine by means of the TCP/IP link.

The Software-PLC of WinPLC-Engine is configured to use the **PG protocol** as well as the **OP protocol** via TCP/IP.

Using the OP protocol, **WinCC[®]** can easily co-operate with WinPLC-Engine. This means that you can comfortably test your graphic system representation using WinPLC-Engine.

What is more, the Pro-edition of WinPLC-Engine enables you to create customized masks. In this case you can insert an arbitrary background image and then place selected objects as necessary.

The following objects are available:

Static text	To include headings and labels in the mask
Dynamic text	To display the up-to-date value of any address.
LED	To represent a digital input or a digital output. The display can also consist of two graphic images that would represent status '1' or '0'. This object can also be used show a permanent graphic image, i.e. one that does not require an address.
Analog Input	Displays an analog input as a bar graph.
Analog Output	Displays an analog output as a bar graph

Each of these objects may be animated (motion profiles). The integrated collision check function can be used to control an input-LED.



2 Installation

2.1 System requirement

- PC with a CPU clock speed of 500 MHz or better
- Windows XP, Windows Vista, Windows7
- 512 MB RAM

2.2 Installation

Start the setup utility and follow the instructions that are displayed on screen. Once the installation has been completed, a “Start” icon appears on your Desktop. The **WinPLC-Engine** program group also contains a “Start” icon and the user manual as a PDF file.



Fig.: The WinPLC-Engine start icon

2.3 Uninstall

The Windows Uninstaller “Add or Remove Programs” contains the WinPLC-Engine icon to uninstall the application.

The uninstaller also removes the entry for the adapter from STEP®7 V5.x.



3 Getting started

3.1 The STEP®7 software configuration

WinPLC-Engine communicates with STEP®7 V5.x via TCP/IP.

Start by specifying the WinPLC-Engine IP address (menu **Extras->Properties of the Software-PLC**):

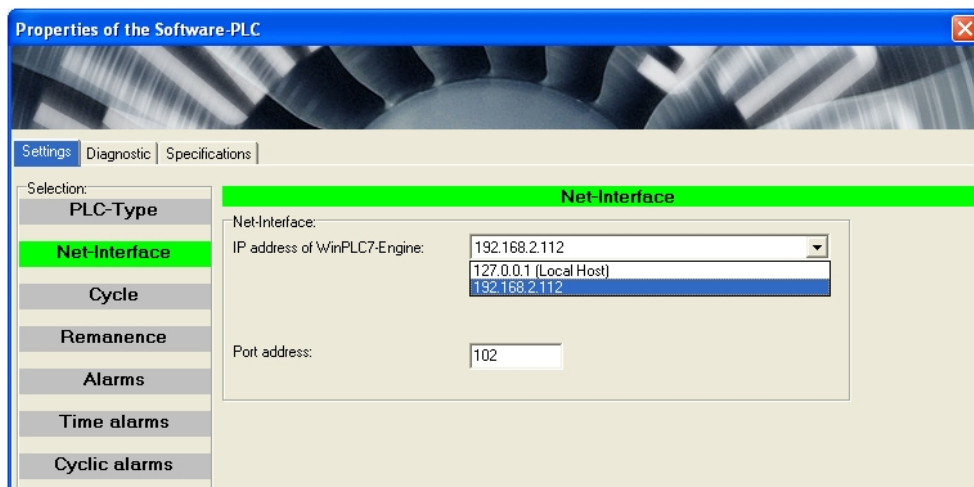


Fig.: The configuration of the WinPLC-Engine Software-PLC

Windows XP configuration:

Select **LocalHost** for this parameter if WinPLC engine and the programming software are executed on the **same PC**.

If the applications are installed on **different PCs**, enter the **IP address of the WinPLC-Engine PC**.

(in this fig. this is 192.168.2.112)

If **port 102** is occupied, select another port, e.g. 7000 – or any other unused port.

Windows Vista/Windows7 configuration:

For Windows Vista/Windows7 it is necessary to distinguish between "Computer is on a network" and "Computer is not on a network":

Computer is on a network:

Always enter the IP address of the WinPLC-Engine computer into WinPLC-Engine.
(you cannot use LocalHost here)

Computer is not on a network:

Always specify the LocalHost address in WinPLC-Engine on computers that are not on a network.

(no other options are available)

If port 102 is occupied, select another port, e.g. 7000 - or any other unused port.



STEP®7 V5.x configuration:

Now the IP address and the port number are defined. Open the PG/PC interface dialog to enable STEP®7 V5.x communications with WinPLC-Engine.

In STEP®7 V5.x, select menu item **Extras->Set PG/PC Interface**

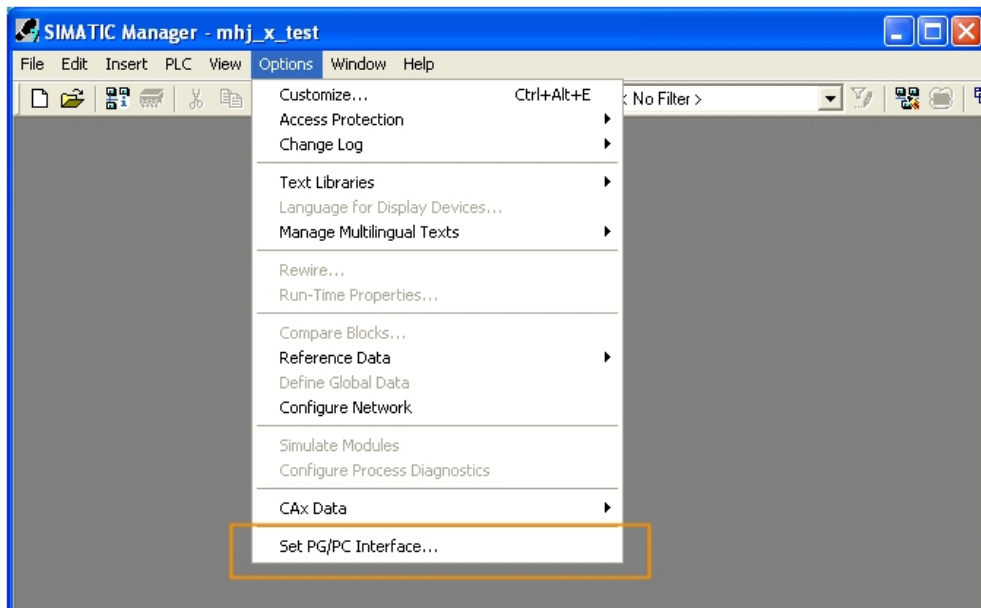


Fig.: Menu item "Extras->Set PG/PC Interface" of STEP®7 V5.x

Select the entry **WinPLC-Engine** and click on the properties button. Make sure the tick box "**Connect to real S7-300/400 and compatible PLCs**" was **not** checked:

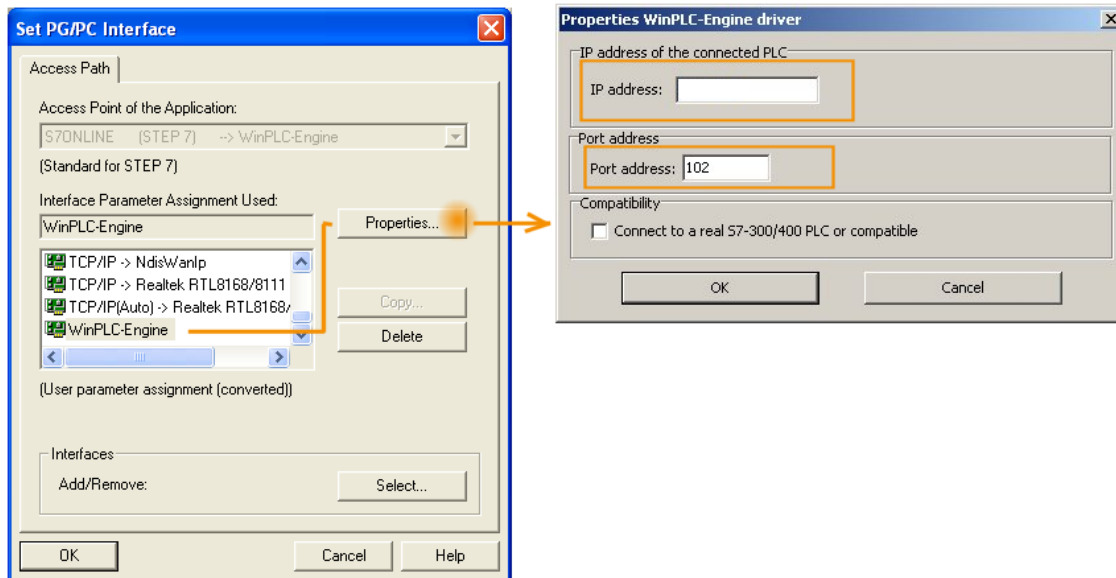


Fig.: Specifying the IP address

Enter the IP address and the port number that you have configured in WinPLC-Engine into this figure.

Confirm the dialog with the OK button. At this point, STEP[®]7 has been configured for WinPLC-Engine. If WinPLC-Engine was started, you can select menu item **PLC->Accessible nodes** in STEP[®]7. You have configured STEP[®]7 correctly if a window with MPI address 2 is displayed:

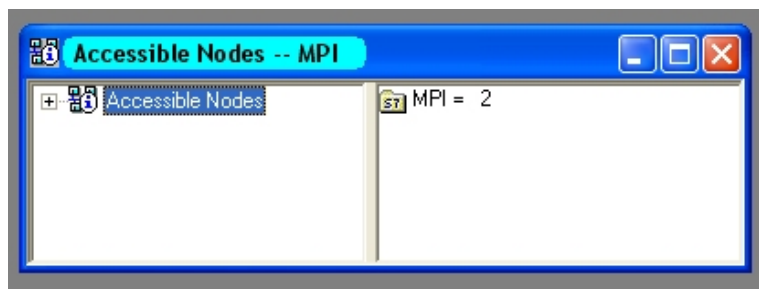


Fig.: Window "Accessible nodes" in STEP[®]7

Now the STEP[®]7 software is configured to always access the WinPLC-Engine. You may now generate blocks and transfer these into the Software-PLC of the WinPLC-Engine. Or open the module status (CTRL+D) to display information regarding the Software-PLC.



3.2 Loading mask

You can open a mask using menu item **File->Open**.

This function displays the folder that was selected in the configuration (Extras->User Settings) under "**Location for the projects**".

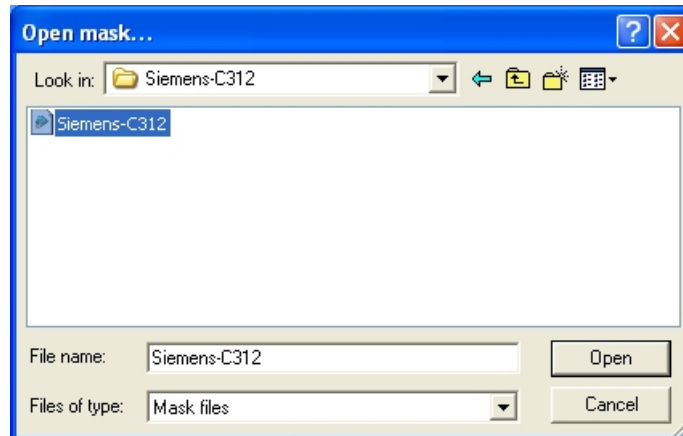


Fig.: Loading a mask

After you have loaded a mask, editor mode is activated. If you wish to change to the cycle mode you must select **Mask->Change to cycle / chance to editor mode** (keyboard shortcut CTRL+F7).

Only now will the Software-PLC be only fully operational.

The next section describes how you can operate the mask in "cycle" mode.

3.3 Operating the masks in cycle mode

Initially, the condition of any mask you load is in editor mode. To simulate the mask, you must turn on the cycle mode using menu item **Mask->Change to cycle / chance to editor mode** (keyboard shortcut CTRL+F7).

In cycle mode, OB1 -if available- is called cyclically when the status of the Software-PLC is RUN.

Each of the masks supplied contains a **RUN-STOP** switch.

Click this switch to change the mode of the Software-PLC to RUN.



Fig.: Software-PLC is in "RUN" mode

This means that the PLC program is executed and that it can be tested.



The mask shows **digital inputs and output** as LED's. Use the left mouse key to change the status of digital inputs.

If you have configured the LED as required you can also change it by means of a button.



Fig.: Changing a digital input with the mouse

You can change analog inputs by means of the mouse and the keyboard. First you must select the required analog input with the left mouse key. A red frame indicated the selected input.

Now you can change the analog input value by means of keyboard keys [+], [Page Up] and [-], [Page Down].

Use the mouse to change the value by clicking and dragging.

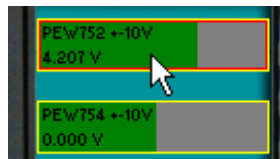


Fig.: Analog inputs, the upper one has a red frame and can be changed by means of the mouse or the keyboard.

You can assign a default value to digital inputs and analog inputs. This default value is defined in the configuration dialog for the input. The inputs are set to the default values when you select menu item **Mask->Set inputs to default values** (keyboard shortcut CTRL+F9).

At the start of a simulation (cycle mode) the inputs are set once to the default value.

It makes sense to assign default value '1' to "normally closed" contacts. This correctly configures the inputs when the simulation is started.



3.4 Creating custom masks

The pro-version of WinPLC-Engine can be used to create customized masks or schematic images.

Select menu item **File->New**. Then you save the mask using **File->Save** and choose a file name.

Hint:

As an alternative you may want to open an existing mask and save it with a different name. Modify the mask as required. The advantage is, that you do not have to create all the elements, e.g. the RUN-STOP switch.

Now you can select a background image using **Mask->Load background image**. The folder "**Pic**" in the main folder of WinPLC-Engine contains a number of background images.

You can now place different objects on the background image. Start by selecting the object type via the mask-menu or by means of the mouse keys.

The following are available:

- Static Label
- Dynamic Label
- LED (binary Input or Output)
- Analog Input
- Analog Output

Use the left mouse key to draw a rectangle (click the left mouse key and hold it, draw the rectangle, release the mouse key).

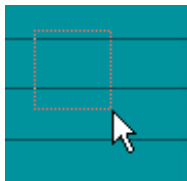


Fig.: Drawing a rectangle.

When you release the mouse key the configuration dialog will be displayed. Here you can define the properties.



Example: inserting 8 input LEDs

Here we will place **8 digital inputs** on a S7-300 312C.
This requires 1 step since you can define the number of LEDs to be generated.

Select the objekt LED:



Fig.: LED object was selected

Draw a rectangle at the correct position using the left mouse key:



Fig.: LED object was selected

As you can see in the figure we must now draw a rectangle around the input byte. This means that you can immediately place the 8 LEDs at the correct location.



Change on tabsheet
"Common" the address to
"124".

Change the number to "8".

New: LED

Common | Color | Interaction | Picture

Address:
☒ LED depends on PLC-operand:
Operand: 1 Byte address: 124 Bit: 0 DB-No: 0

☐ LED depends on variable: \$RUN

Default:
Default of the input signal: 0

Number: 8 How many LEDs are to be generated?

OK Cancel Help

In the "Color" tab you can
change the color of '1' and '0'.

New: LED

Common | Color | Interaction | Picture

Colors:
Color 1
Color 0

Preview:
[Green square]

In the "Interaction" tab you
should place a check mark
into "LEDs can be switched
with a left mouse click"

New: LED

Common | Color | Interaction | Picture

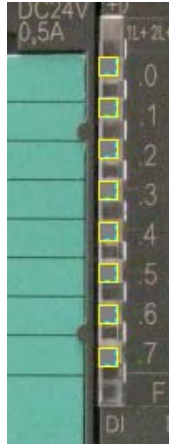
Interaction
☒ LEDs can be switched with the left mouse click

☐ Define hotkey: [] (A..Z, 0..9) ☐ Push button



The 8 LEDs are inserted when you close the dialog. All the LEDs are selected, i.e. you can position them immediately using the cursor keys (←→↑↓) .

If the vertical position is not correct you can modify it as follows: Move the first and the last LED to the correct position. Select all LEDs (draw a rectangle while the object 'Arrow' is active) and select menu item **Edit->Align vertically**



To change an object at a later stage, execute a double-click to open the respective dialog.

If you have selected the object, use the RETURN key to access the dialog.



Example: Inserting a RUN-STOP switch

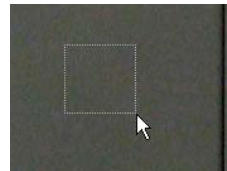
Insert a RUN-STOP switch into the mask to permit the PLC mode to be switched between RUN and STOP.

Alternatively, you may also switch the Software-PLC to RUN or STOP mode by means of the S7 programming software.

Select the object "LED"

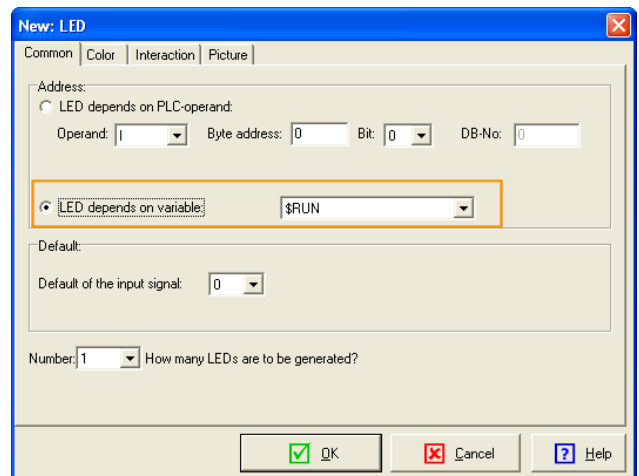


Draw a rectangle.

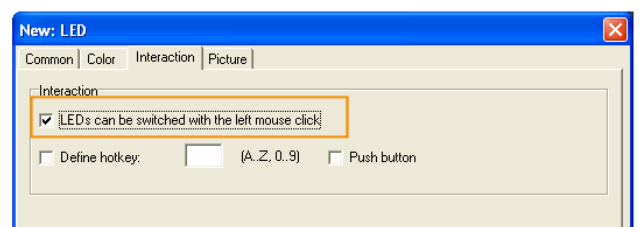


Important: for the address, select "LED depending on variable".

You must select variable \$RUN.



In the Interaction tab you must check tick box "LEDs can be selected with a left mouse click" to ensure that the RUN-STOP switch is available to the mouse.



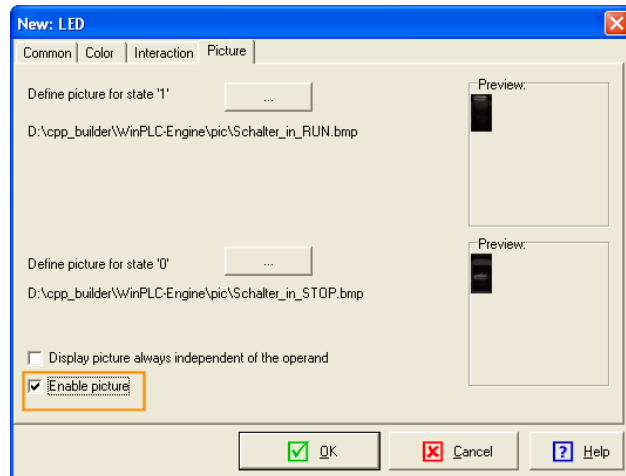
WinPLC-Engine • User manual • English



Select two symbols in the "Picture" tab.
First you select the symbol for
RUN-mode and then the one for
STOP-mode.

It is also important to remember: option
"Enable picture" must be selected.

You can use the graphic symbols that
are available in the "Pic" folder that is
located in the installation path of
WinPLC-Engine

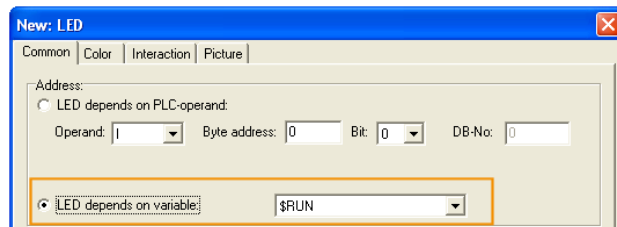


When you confirm this dialog the
RUN-STOP switch will be displayed.

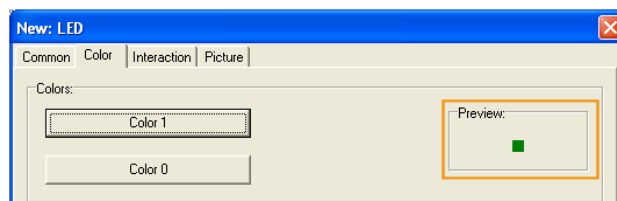


Insert two LEDs to make the mode
(RUN/STOP) visible. An orange LED for
STOP-mode and a green LED for
RUN-mode.

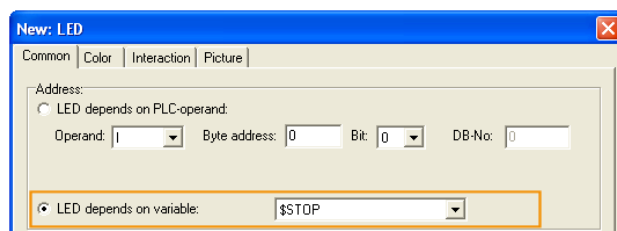
Start by selecting the LED-object and
continue by drawing a rectangle. The
configuration dialog will be displayed.



The color of the RUN-LED should be
green:

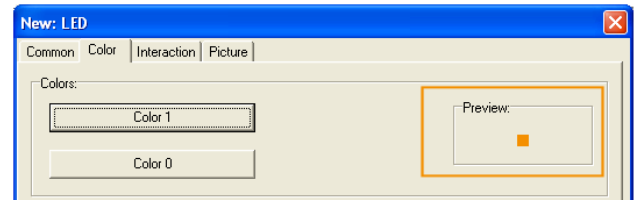


Next, we create the STOP-LED.
The status of the STOP-LED depends on
variable \$STOP.

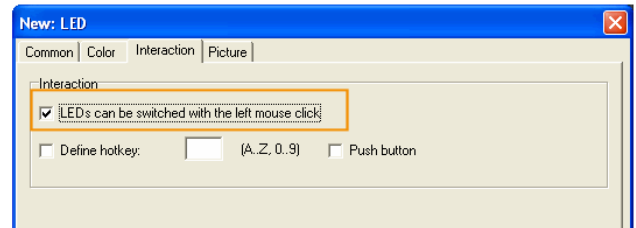




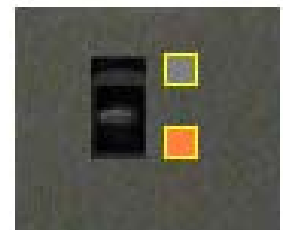
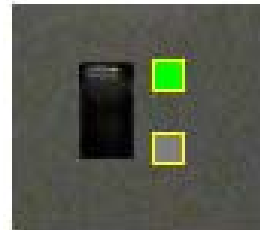
The color of the STOP-LED must be orange.



It must be possible to change the LED with the mouse.



The RUN-STOP switch is displayed as follows in "RUN" mode and in "STOP" mode. If you position this switch on the switch of a S7-300? symbol, the switch responds realistically when it is operated. Since the LEDs can also be changed by means of the mouse, you may change the mode using the graphic symbol or the LEDs.





4 User interface

4.1 Outline

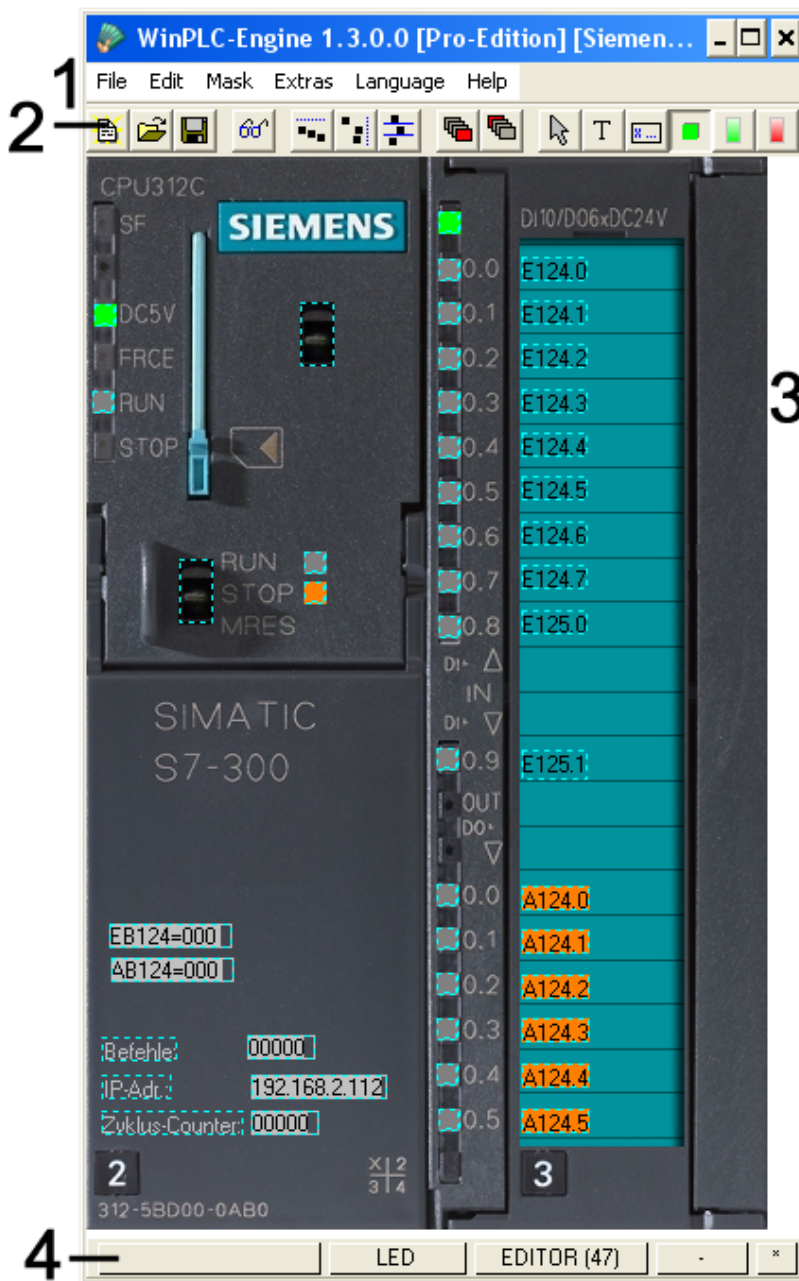


Fig.: WinPLC-Engine with a 312-mask

- 1 Menu bar
- 2 Mouse keys with the most important commands
- 3 User-defiend area
- 4 Status bar



4.2 Context menu

A click with the right mouse key displays the context menu. This menu contains many important commands:

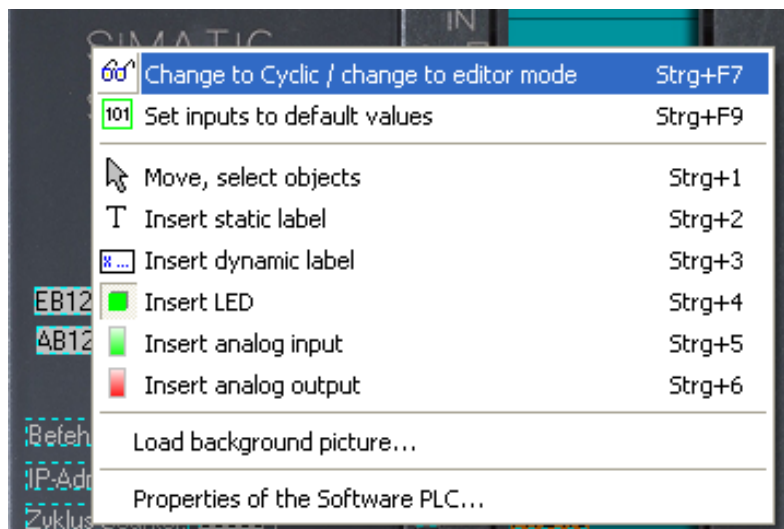


Fig.: WinPLC-Engine context menu

4.3 Keyboard shortcuts

Shortcut key	Function	Remarks
CTRL+O	Open a mask	
CTRL+N	Create a new mask	
CTRL+S	Save the current mask	
CTRL+B	"Motion profiles" dialog	
CTRL+C	Copy selected objects into the global clipboard	
CTRL+V	Insert objects from the clipboard	
CTRL+A	Select all objects	
DEL	Delete the selected object	
CTRL+D	"Properties of the Software-PLC" dialog. This dialog will also be displayed by a double-click in an empty area of the mask.	
CTRL+F	Bring the selected object into the foreground	Only a single object may be selected.
CTRL+H	Send the selected object to the background	Only a single object may be selected.
CTRL+ALT+H	Align horizontally	
CTRL+ALT+V	Align vertically	
CTRL+ALT+A	If you have selected several objects, the objects are arranged with the same vertically spacing	
CTRL+T	The respective .RTF description is opened. The file is displayed in WORDPAD or in WORD, if installed.	



CTRL+F7	Toggle the mask between simulation module or edit mode.	
CTRL+F9	Set all inputs to default values	Only in simulation mode
CTRL+U	Open the user user preferences dialog	
F1	Open the Help syste	
CTRL+1	Select object and move it to a new position	
CTRL+2	Object selection: static text	
CTRL+3	Object selection: dynamic text	
CTRL+4	Object selection: LED	
CTRL+5	Object selection: analog input	
CTRL+6	Object selection: analog output	
+ or [Page Up]	Increase the value of the selected analog input	
- or [Page Down]	Decrease the value of the selected analog input	



5 Object

5.1 Text objects

Use the text object to insert user-defined text into the mask.

The following settings are available:

- Text
- Text color
- Background color
- Transparency

You can also create multiple text elements in the same manner as described for the LEDs. This enables you to label an entire input byte in a single operation. For this purpose, change the "Number" field in the dialog. If you have checked the tick box "Increase address" the addresses in the text (e.g. E124.0) are incremented automatically.

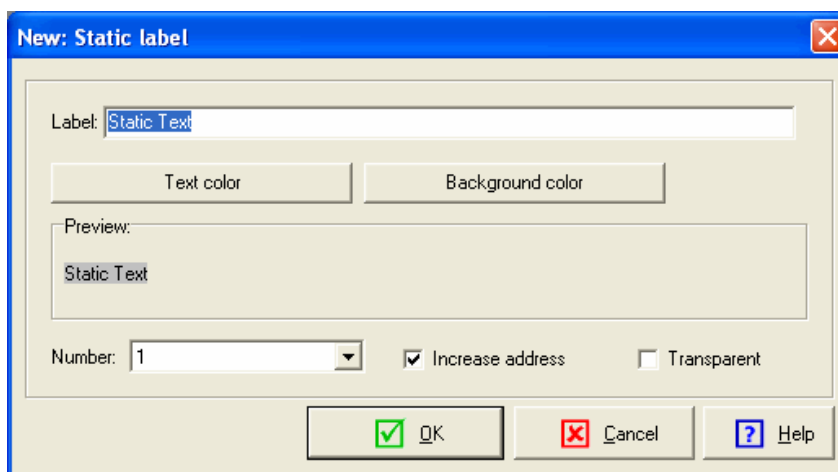


Fig.: Configuration dialog for LED objects

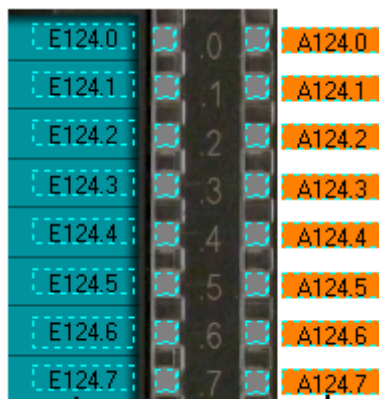


Fig.: Examples of static text.



Hint about text objects:

Normally, text is centered.

If you insert "\$r" before the text, it is right-justified on the mask.

If you insert "\$l" before the text, it is left-justified on the mask.

5.2 Dynamic text objects

Dynamic text objects can be used to display the actual values of the following addresses:

- Input (bit, byte, word, double-word)
- Output (bit, byte, word, double-word)
- Clock memory (bit, byte, word, double-word)
- Data (bit, byte, word, double-word)
- Peripheral inputs

The following notations are available:

- DECIMAL (without sig)
- DECIMAL (with sign)
- Hexadecimal
- Binary

It is also possible to include the actual values of certain variables:

\$CYCLIC_COUNTER	This is a counter in the Software-PLC. This counter is incremented with each cycle
\$COUNT_S7_COMMANDS	Number of STL statements in a cycle
\$IP_ADR	IP address of WinPLC-Engine.

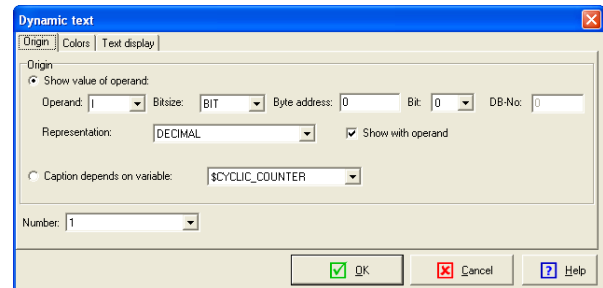
Dynamic text can also be used to display text that depends on a bit address.



The configuration dialog of a dynamic text object:

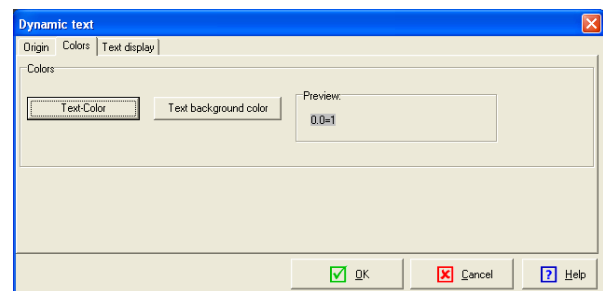
"Origin" tab:

Here you can specify whether you are displaying an address or a variable. To display addresses you must also specify the respective address and, for data blocks, the data block number. You may also specify the number of the objects to be generated when the object is being generated.



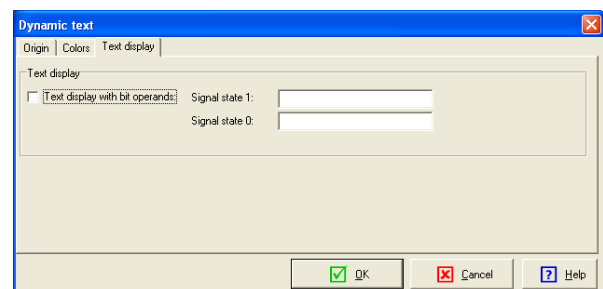
"Color" tab:

Here you can define the text color as well as the background color.

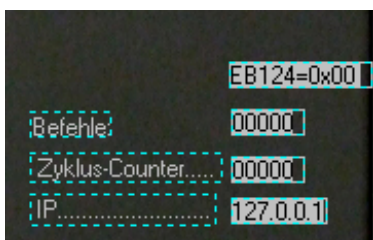


"Text display" tab:

Assigns two text items to bit addresses that are displayed when the status is '0' and when it is '1'.



Examples of dynamic text objects:





5.3 LED object

What can you do with a LED object?

- Display a bit address(I/O/M/D) as a LED

- the bit addresses (with the exception of outputs) can be changed with the mouse.
- Bit addresses may have a default value. This default value is applied when you start the simulation
- You may define a hotkey
- Key property or switch property

- Display an internal variable:

- \$RUN: status is '1' when the mode of the Software-PLC is RUN.
- \$STOP: status is '1' when the mode of the Software-PLC is STOP.
- \$WinSPSS7: status is '1' when WinSPS-S7 / WinPLC7 were started.
- \$SimaticNet: status is '1' if the Siemens STEP7 software was installed.
- \$TCP/IP: status is '1' when the TCP/IP interface of WinPLC-Engine is active.

- A display of two pictures that depend on the value of the address/variable.

- Display of a picture that does not depend on an address.

These properties are defined by means of a simple configuration dialog:

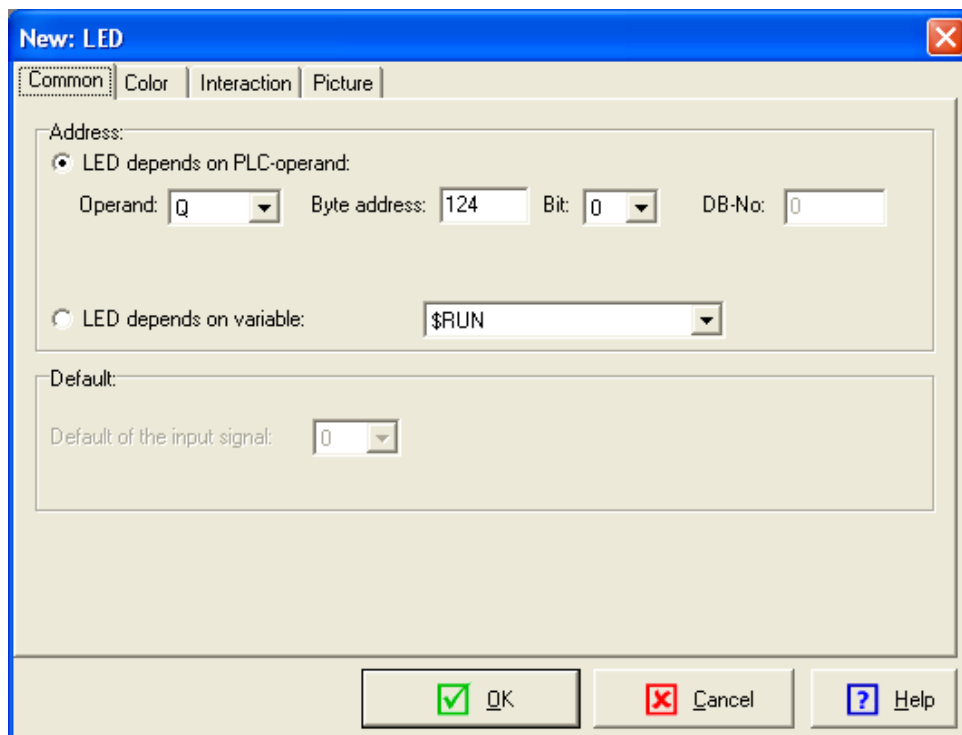


Fig.: Configuration dialog for LED objects



5.4 Analog input and output

Analog inputs and outputs are inserted as bar graphs. You can specify the orientation of the bar graph.

Example of an analog input and an analog output:

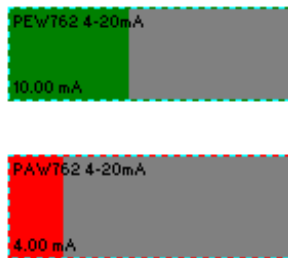


Fig.: Representation of an analog input and output.

Here the following parameters may be defined:

- Input and output address
- The orientation of the bar graph (right, left, top and bottom)
- Do you want to display the address in the bar graph?
- Do you want to display the actual value in the bar graph as well?
- Measurement range
- For inputs: the default value

If the mask is in cycle mode (CTRL+F7) the analog input can be changed with the mouse or the keyboard:

First, click on the required input with the mouse. A red frame appears around the input. Now you can increase the analog value using the keys [+] or [Page Up]: You can decrease the value using the keys [-] or [Page Down].

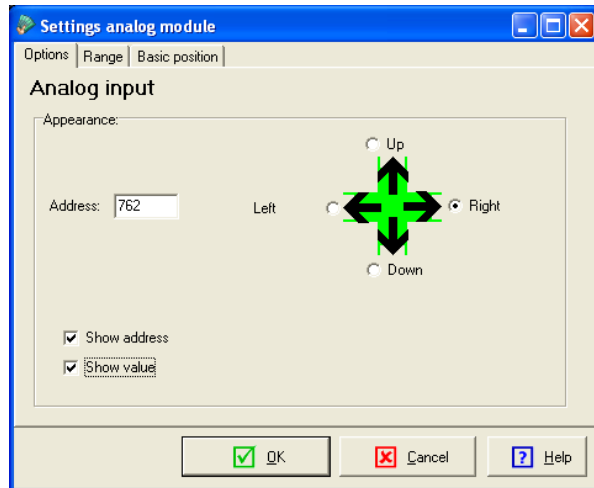
To change the value with the mouse, click and hold the input with the left mouse key. The value changes as you move the cursor with the mouse.



The configuration dialog of the analog input/output:

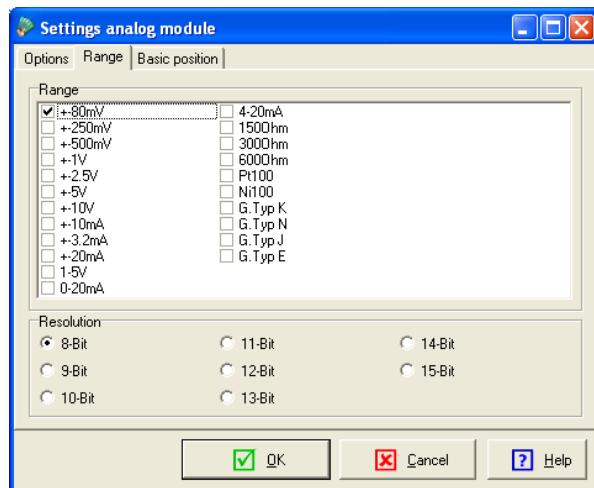
"Options" tab:

This is where you can define the input address and the orientation. You can also specify whether the address and the actual value should be displayed on the bar graph.



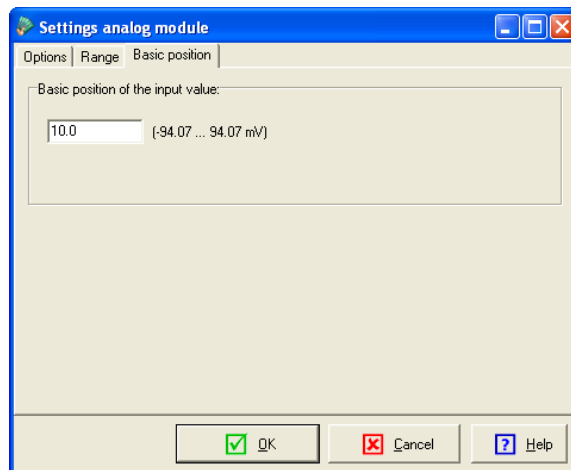
"Rang" tab:

Here you select the required measurement range.



"Basic Position" tab:

Here you can specify a default value for the respective inputs. This default value is applied when the simulation is started.





6 Motion profiles

You can use motion profiles to assign animation properties to any object.

You can configure the profile so that a collision with an input LED changes its state to '1'.

To assign a motion profile to an object (e.g., a static text object), select the object before you open the dialog **Mask->Motion profiles** (CTRL+B). Choose a motion profile in the list on the left.

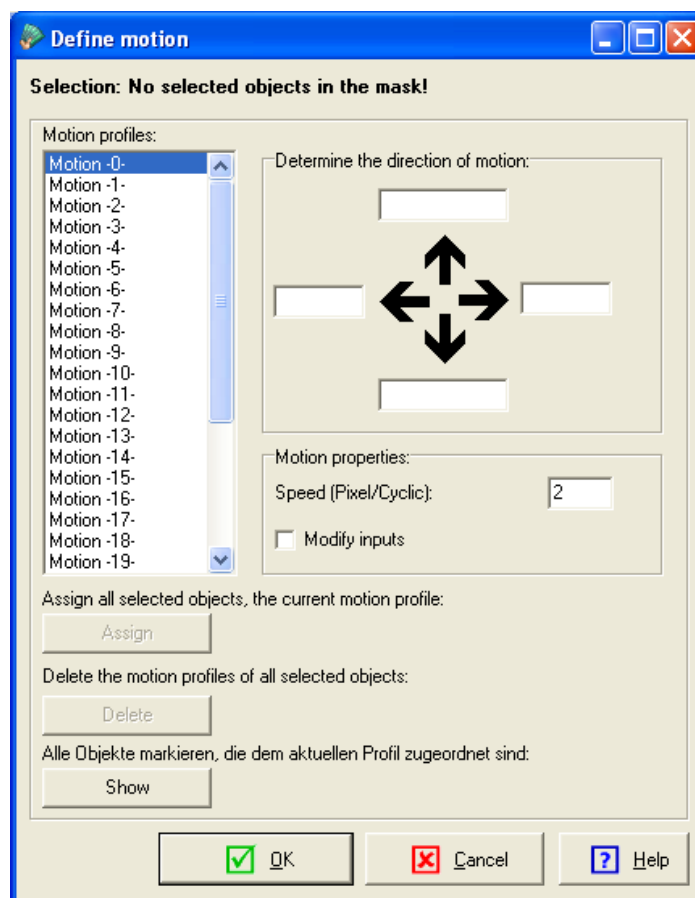


Fig.: Motion profiles dialog

Enter a bit address (e.g. A3.3) into the required **directions of motion**. If the address is valid the field turns green. The field turns red if the address is invalid or incomplete. In the field "**Speed (Pixel/Cycle)**" you may specify the speed at which the object should be moved (pixel per cycle).

If the moving object should control input LEDs check the tick box "**Modify Inputs**". Click the button "**Assign**" to complete your entries. The current motion profile is assigned to the selected objects.

In this manner, a total of 30 different motion profiles may be created and assigned.

The "**Delete**" button removes motion profiles from selected objects.

The "**Show**" button selects all the objects in the mask that are set to the current motion profile.



7 Configuration of the Software-PLC of WinPLC-Engine

You can either enter this configuration using the configuration dialog of WinPLC-Engine or with the STEP®7 V5.x hardware configurator.

7.1 Configuration with the WinPLC7-Engine dialog

Use menu item **Extras->Specifications** to open the configuration dialog of the Software-PLC:

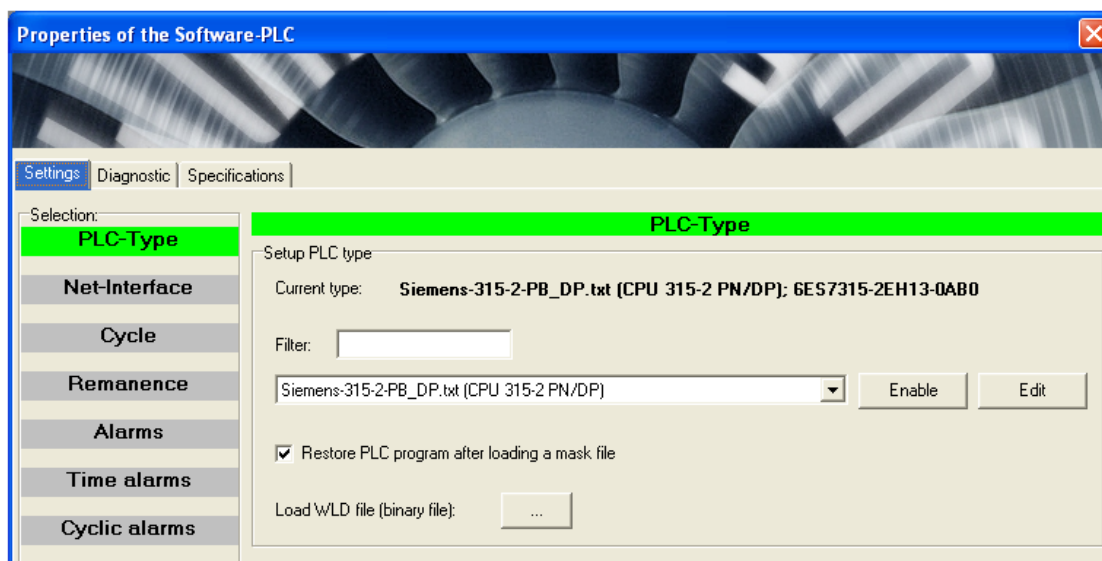


Fig.: Software-PLC related settings

This dialog is divided into **three tabs**:

Settings	In this tab you can define the properties of the PLC. These properties are subdivided as follows: <ul style="list-style-type: none">- PLC-Type- Net-Interface- Cycle- Remanence- Alarms- Time alarms- Cyclic alarms
Diagnostic	This tab lists the diagnostic messages that have occurred since the program was started. New messages are always at the top of the list.
Specifications	This table displays the technical specifications of the Software-PLC. The data may change, depending on the selected CPU type.



7.1.1 PLC-Type

Here you can select the PLC-type (CPU) from a list. If required, you can extend this list. Press the **Edit** button, modify the values and save the file with another name.

The PLC-type affects the properties of the Software-PLC. You can inspect the properties of the selected CPU in the tab "**Specifications**".

7.1.2 Net-Interface

Here you can specify the IP address that WinPLC-Engine uses to communicate. This entry must correspond with the entry in the Siemens dialog "**Set PG/PC Interface**".

7.1.3 Cycle

Here you can define the different settings for the cycle and the clock memory byte.

Under certain conditions, WinPLC-Engine may not provide all the SFB/SFC of the real CPU. Check the tick box "**Ignore not existing SFB/SFC**" if the software PLC should remain in RUN mode even if an unsupported SFB/SFC is encountered in the PLC program.

If the values are grayed out they may not be available for changing. This situation could, however, be modified by a service pack of WinPLC-Engine.

7.1.4 Remanence

At present, the remanence behavior of the current version of WinPLC-Engine is not available.

For this reason, the input fields in the dialog are not active.

7.1.5 Alarms

This section is used to configure the Alarm-OBs. At present, WinPLC-Engine only supports **delay alarms OB20, OB21, OB22 and OB23**.

7.1.6 Time Alarms

Here you can configure the time alarm OBs **OB10 to OB17**. At the appropriate execution time, a call is issued to the respective OB if it was loaded into the Soft-PLC.

7.1.7 Cycle Alarms

The cycle alarms (OB30 to OB38) can be used to implement periodic alarms.

For instance, on the CPU315 the OB35 may be programmed. Any OBs that are grayed out are not supported by the selected CPU.



7.2 Configuration using STEP®7 V5.x

You may also use the hardware configurator of STEP®7 to configure WinPLC-Engine. The type must be selected in the hardware configuration, depending on the selected CPU-type.

For the following example, the Siemens CPU 313C was selected in WinPLC-Engine:

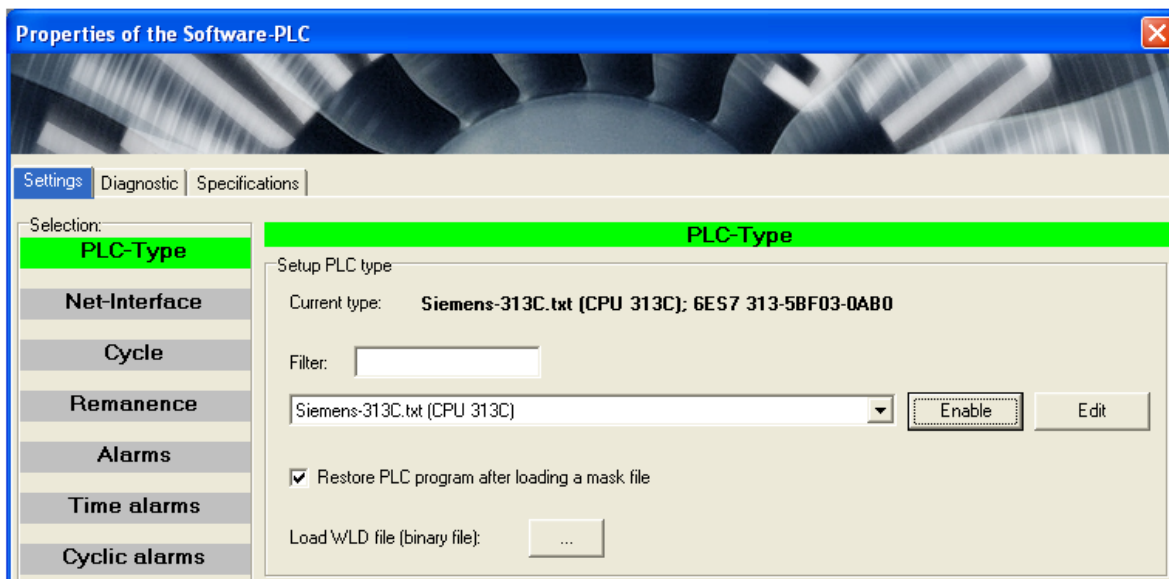


Fig.: CPU313C is selected

In the Siemens hardware configurator you must also insert the CPU with the order number

6ES7 313-5BF03-0AB0 into the rack.

Please ensure that you use a CPU with firmware rev. level 2.6:

SIMATIC 300-Station (Konfiguration) -- S7_Pro1							
(0) UR							
Steckplatz	Baugruppe	Bestellnummer	Firmware	MPI-Adresse	E-Adresse	A-Adresse	Kommentar
1							
2	CPU 313C	6ES7 313-5BF03-0AB0	V2.6	2			
2.2	DI24/DO16				124...126	124...125	
2.3	AI5/AO2				752...761	752...755	
2.4	Zählen				768...783	768...783	
3							
4							

Fig.: Hardware configurator with CPU 313C

Execute a double-click on the CPU to enter the configuration. You can transfer the configuration into the Soft-PLC of WinPLC-Engine via **PLC->Upload**.



8 Settings

The following dialog is available under **Extras->User settings**:

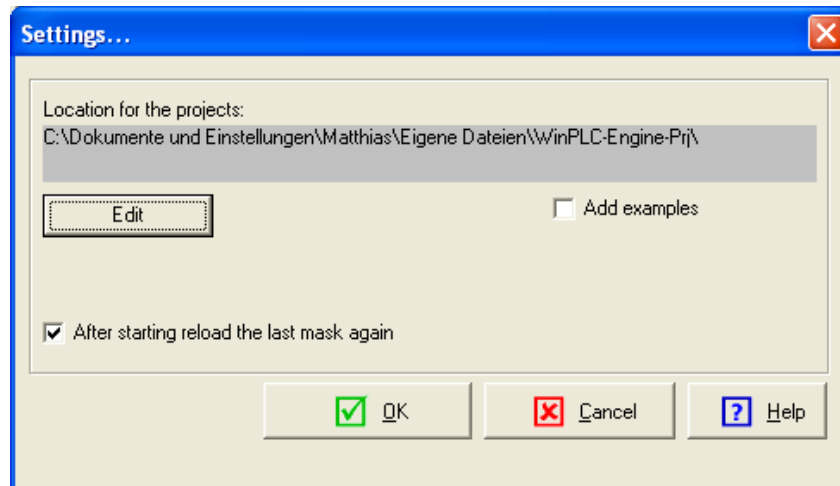


Fig.: User settings

The **"Edit"** button can be used to define the storage location for the masks. Select a path with write access privileges. This path should normally point to a folder in "My Documents".

The samples supplied with the package are copied into the selected storage location if you check the tick box **"Add examples"** and press OK.

Hint:

You can also define the storage location by specifying the respective path when you start WinPLC-Engine.

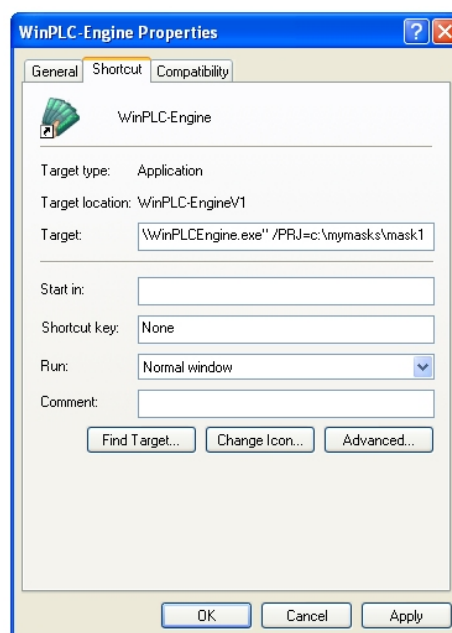
For this purpose, the start icon of WinPLC-Engine must be modified:

In input box "Target" you may add the storage location path following the Exe file using the parameter /PRJ.

Example:

C:\Program
Files\MHJ-Software\WinPLC-Engin
eV1\WinPLCEngine.exe
/PRJ="c:\MyMasks\Mask1"

In this case it is important to enclose the path in quotes ["].





9 Technical specifications of the Software-PLC

The following table shows the maximum **configuration level** of the Software-PLC in WinPLC-Engine. You can use AG files **to** limit the technical properties as required by the different types of PLCs.

The Software-PLC is designed to process the **object code** (MC7-code) of S7-controllers directly. For this reason, WinPLC-Engine is also able to process protected blocks **or S7-Graph blocks**.

RAM-size in byte	245.760
L-Stack size in byte	1.024
FC range of numbers	FC 0 - DC 2047
FB range of numbers	FB 0 - FB 2047
DB range of numbers	DB 0 - DB 1023
Input address space (process image)	EB 0 - EB 2047
Output address space (process image)	AB 0 - AB 2047
Peripheral address space inputs	PEB 0 - PEB 2047
Peripheral address space outputs	PAB 0 - PAB 2047
Clock memory address space	MB 0 - MB 2047
Timer address space	T 0 - T 255
Counter address space	Z 0 - Z 255
Accessible OB's	1, (cyclic operation) 10, 11, 12, 13, 14, 15, 16, 17, (timer alarms) 20, 21, 22, 23 (Verzögerungsalarme) 30, 31, 32, 33, 34, 35, 36, 37, 38, (cycle alarms) 40, 100, 101 (start-up)
Accessible SFC's	0, 1, 2, 3, 4, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 43, 46, 47, 64
Accessible SFB's	0, 1, 2, 3, 4, 5, 32
Communications	
TCP/IP - PG protocol	✓
TCP/IP - OG protocol	✓



10 External access to the Software-PLC

External programs can access WinPLC-Engine via the **integrated DLL interface**. This means that it is possible to share inputs and outputs or link external hardware to other software products.

The Software-PLC is part of the DLL **WS7_S7AG.DLL**. When you install WinPLC-Engine, this file is placed in the Windows System32 folder.

The following export functions are available:

BYTE* _export FAR PASCAL S7AG_GetPeriEPtr()

Returns a BYTE pointer for the peripheral space of the inputs.
The size of this space is 2048 bytes.

C-language example:

```
BYTE *pInputs=S7AG_GetPeriEPtr();
pInputs[0]=200; //write 200decimal into inputbyte 0
pInputs[2]=50; //write 50decimal into inputbyte 2
```

BYTE* _export FAR PASCAL S7AG_GetPeriAPtr();

Returns a BYTE pointer for the peripheral space of the outputs.
The size of this space is 2048 bytes.

C-language example:

```
BYTE *pOutputs=S7AG_GetPeriAPtr();

//reads output byte 0 and enters it into variable AB0
BYTE AB0=pOutputs[0];

//reads output byte 1 and enters it into variable AB1
BYTE AB1=pOutputs[1];
```



WORD _export FAR PASCAL S7AG_GetCPUZustand();

This function returns the status of the Software-PLC:

- 0: The Software-PLC is in STOP mode (i.e. the program in the controller is not being executed)
- 1: Software-PLC is starting up (i.e. the start-up blocks are currently being processed)
- 2: The Software-PLC is in RUN mode (i.e. the program in the controller is being executed)



11 Call parameters

WinPLC-Engine accepts different command line parameters when it is started:

Parameter	Description	Example
.maske_ini	Any mask file supplied with the call is opened immediately. The mask file must always be the first parameter.	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe d:\prj\MeineMaske.maske_ini
/NOMENU	This parameter hides the menu.	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe /NOMENU
/NOMOUSEBAR	This parameter hides the mouse buttons.	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe /NOMOUSEBAR
/RUN	This parameter starts an immediate simulation of the mask that was supplied.	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe /RUN
/READONLY	This parameter prevents changes to the masks that are supplied in the call	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe /READONLY
/PRJ=	This parameter defines the path to the user project. This is the path where mask files are stored. Important: Enclose the path in quotes ["].	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe /PRJ="c:\MeineWinPLCEngineProjekte"
/RUNTIME	The loaded project cannot be changed. The menu and the mouse buttons are hidden.	C:\Program Files\MHJ-Software\WinPLC-Engine\WinPLCEngine.exe d:\prj\MeineMaske.maske_ini /RUNTIME



<End of Doc>