Freeway Exercise

Solutions for OEMs, FreeStudio

Thermostat exercise

Schneider Electric

Table of Contents

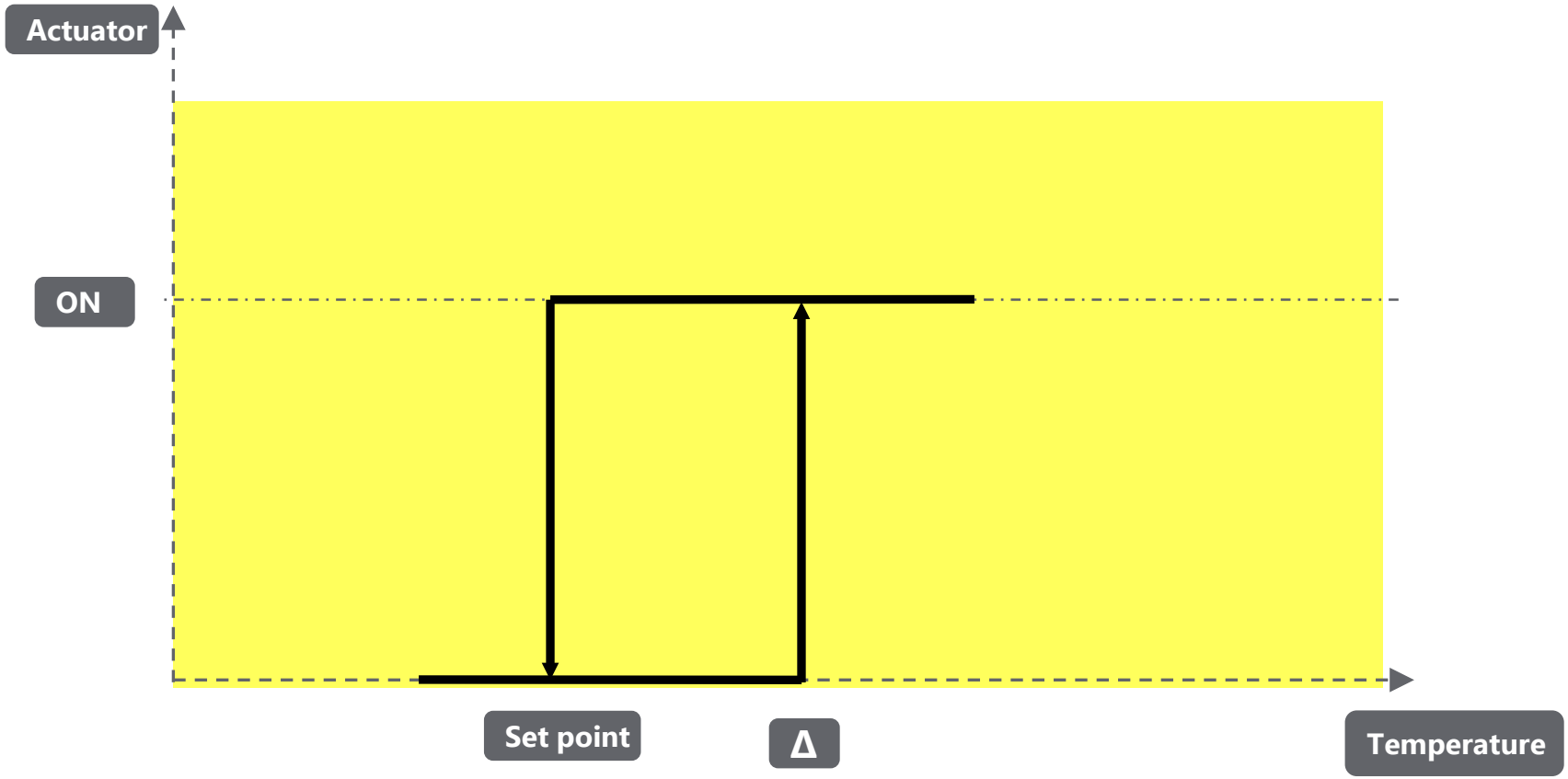
Chapter 1

Function Description

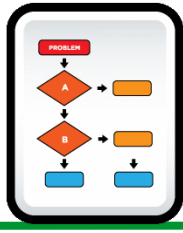
Goal:

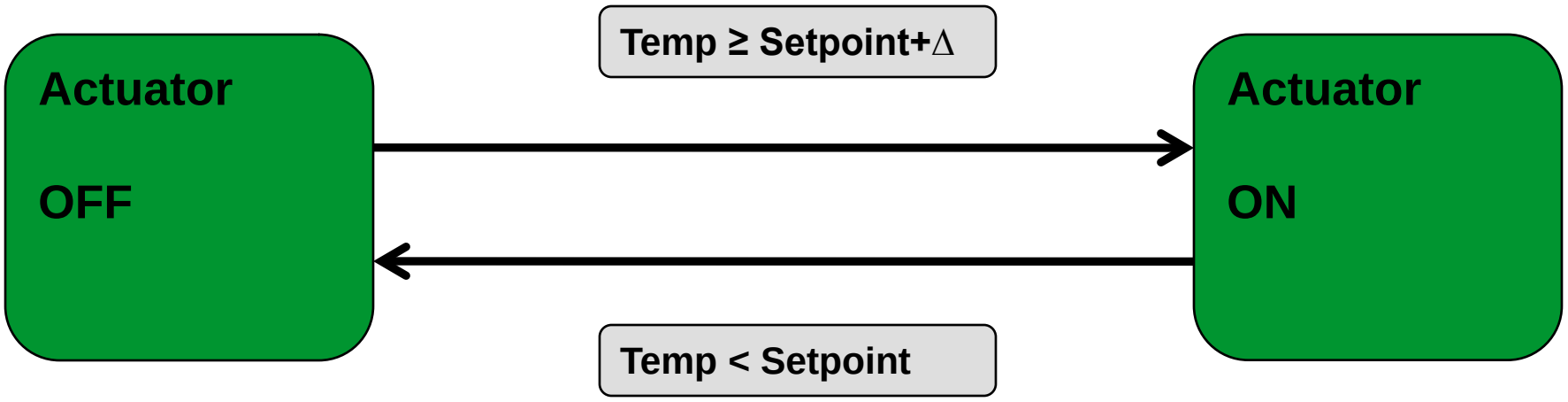
Describe thermostat flow chart

Thermostat cycle



Thermostat states (function description)





Chapter 2

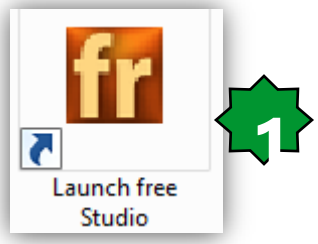
Programming

Goal:

- Familiarizing with programming environment
- Creating Thermostat Function Block

Creating New project

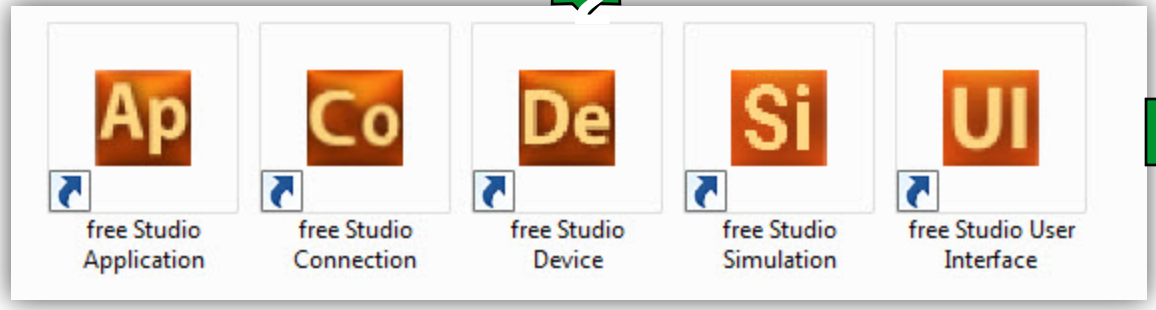
2*click



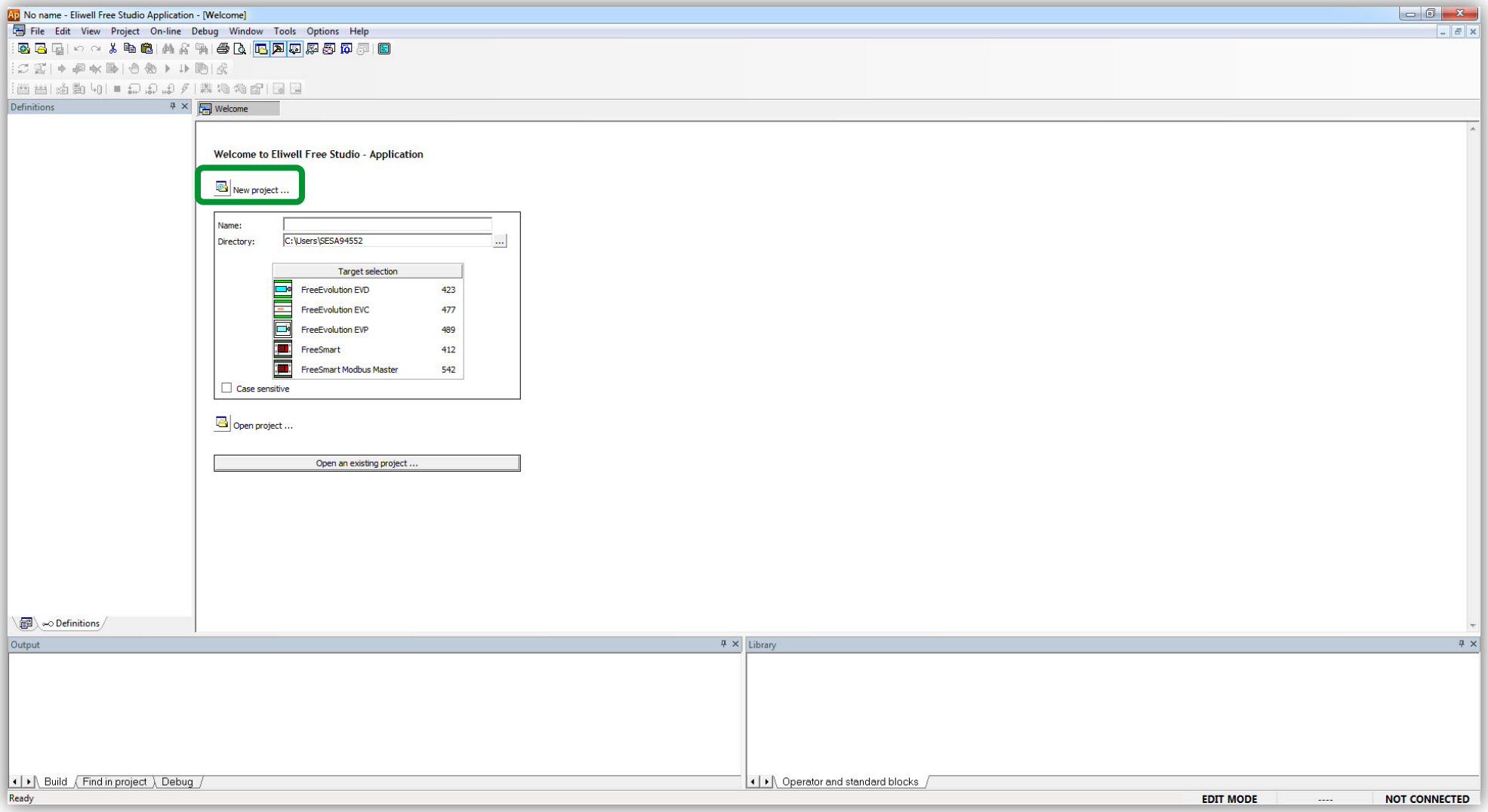
1

2*click

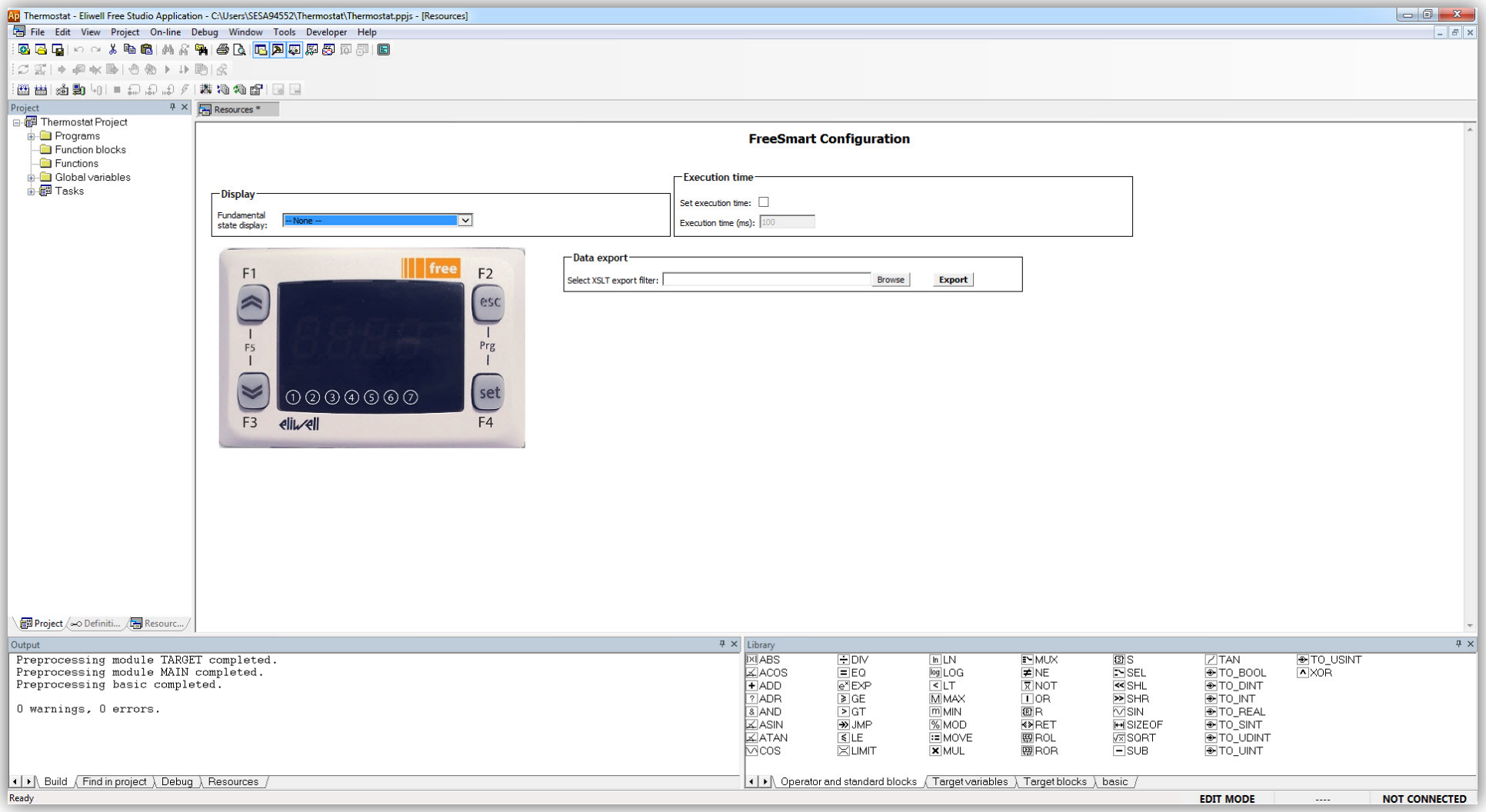
2



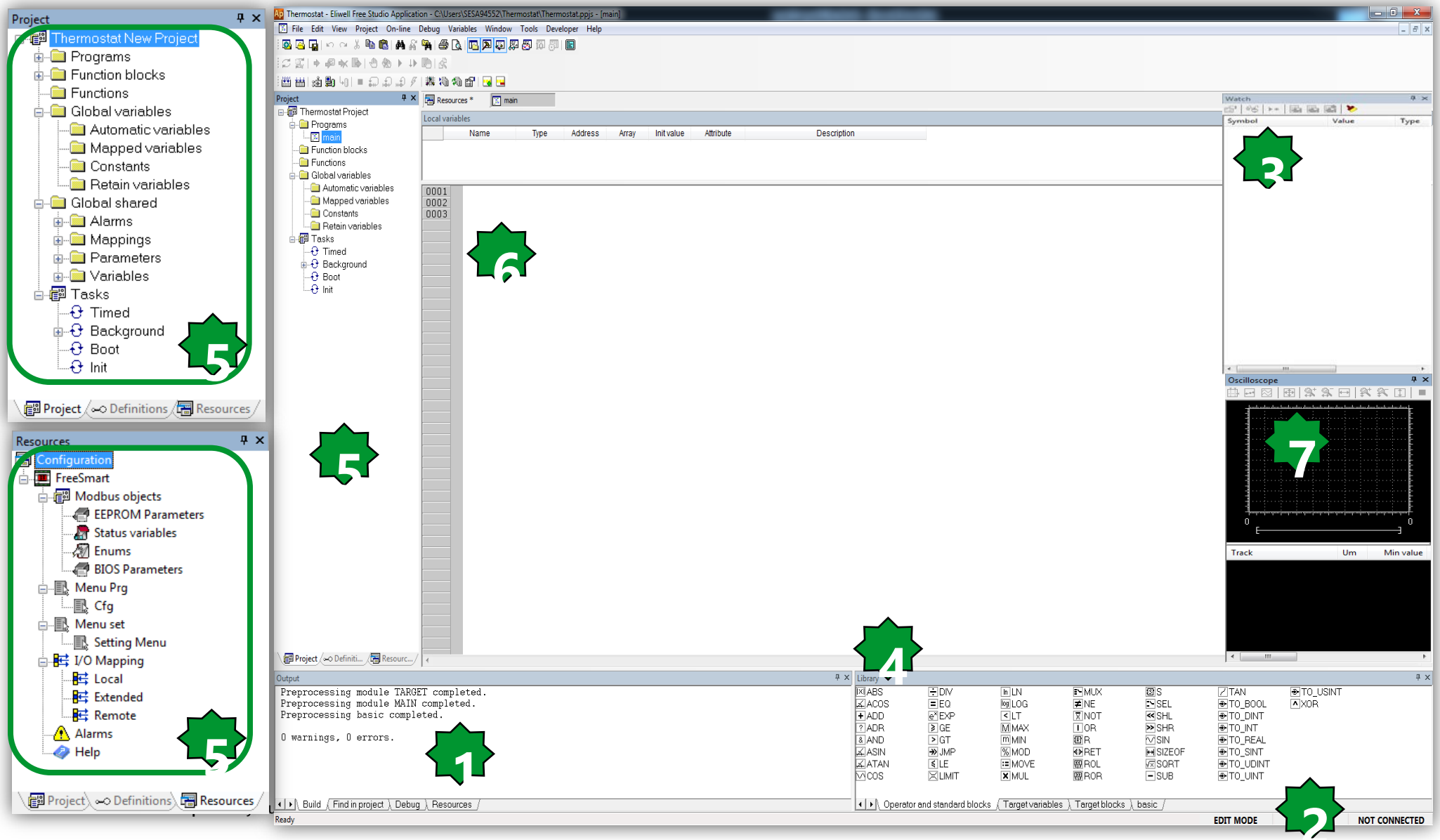
Creating New project



Creating New project

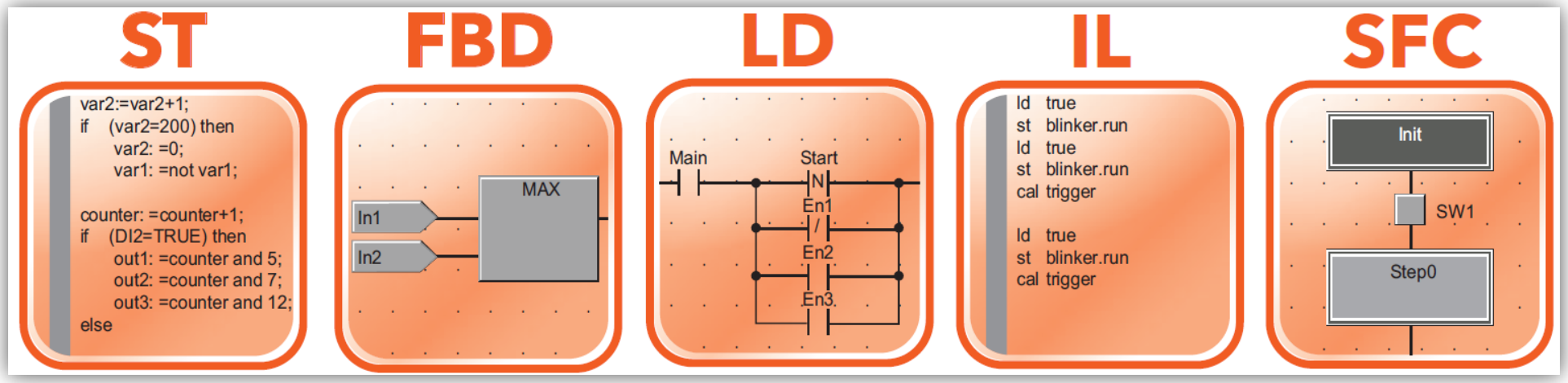


Programming environment



Programming Languages

The **FREE STUDIO platform is compatible with all 5 standard programming languages (IEC 61131-3).**



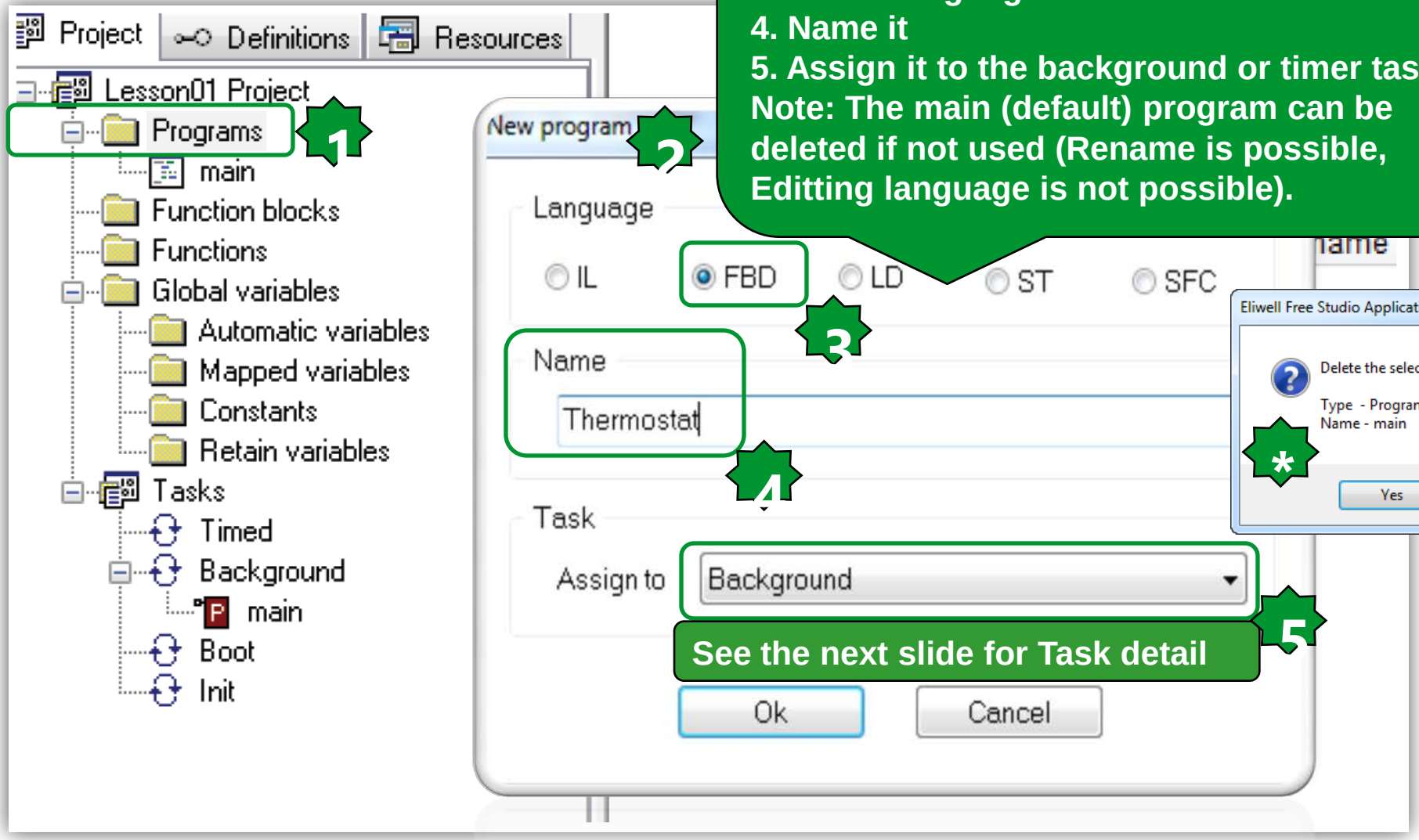
5 programming languages, 2 text-based and 3 graphics-based:

- **ST, Structured Text,language text**
- **FBD, Functional Block Diagram language graphical**
- **LD, Ladder language graphical**
- **IL, Instruction List language text**
- **SFC, Sequential Function Chart language graphical**

Thermostat program

1. Right click on the programs
2. New program
3. Select language
4. Name it
5. Assign it to the background or timer task

Note: The main (default) program can be deleted if not used (Rename is possible, Editting language is not possible).



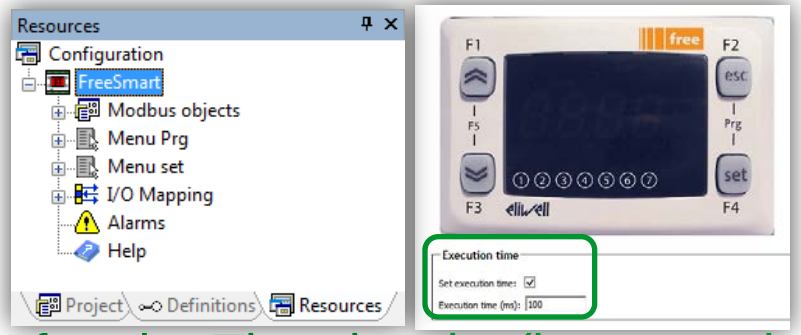
Associating a program to a task

For a program to run, it must be associated to a task.

There are various types of tasks:

- **BOOT Task** runs once only at system start-up.
- **Init. Task** runs each time the application is downloaded and after **BOOT**.
- **Timed. Task** runs at regular intervals which can be set by the developer.

The default setting is 100ms.



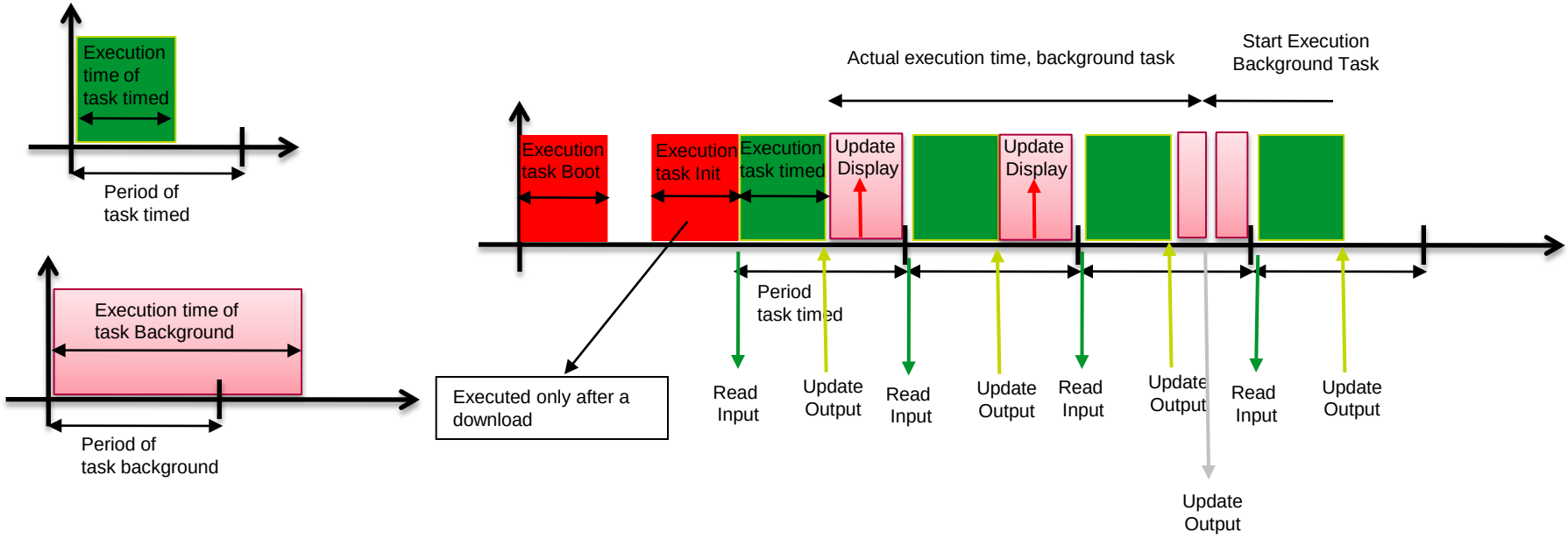
- **Background. Task** runs with low priority after the Timed tasks (between the end of one Timed task and the start of the next), it can be interrupted in case of long execution or executed more than 1time in case short execution.

Note Each new project has the main program associated to the background task (the main program can still be eliminated and/or associated to other tasks).

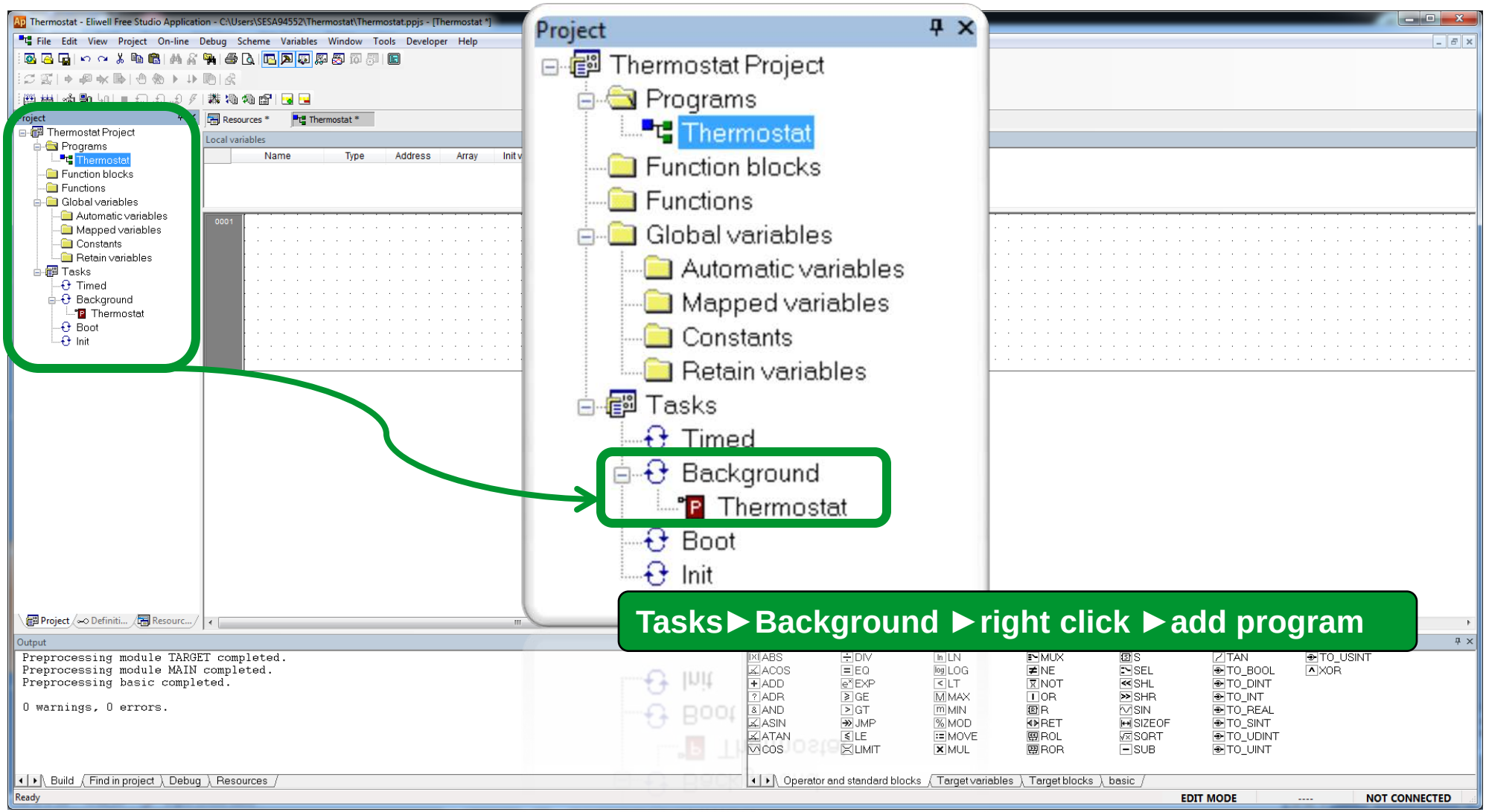
To activate a task, go to the task you want, right-click and select Add program

Tasks

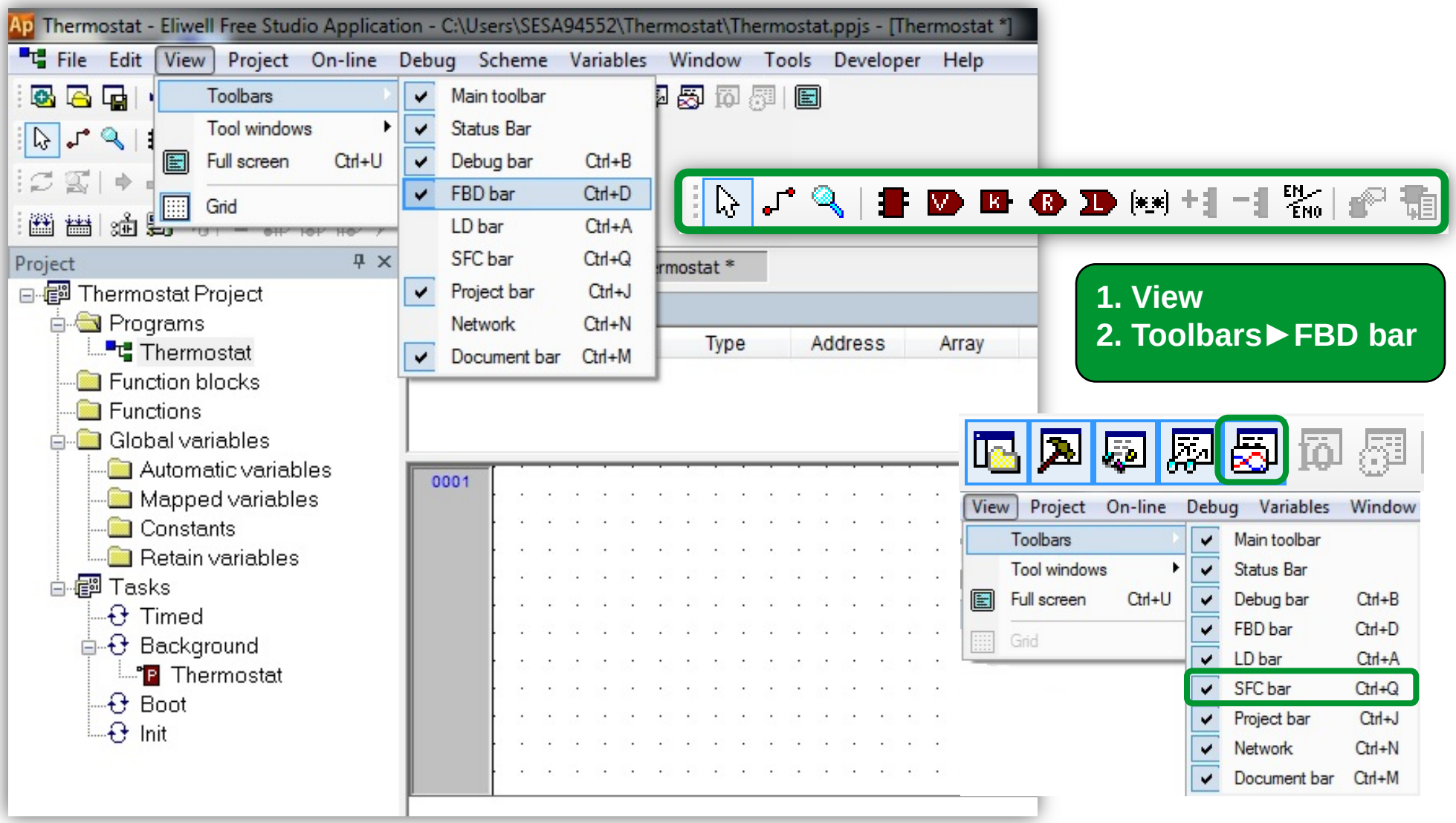
- **BOOT Task** runs once only at system start-up.
- **Init. Task** runs each time the application is downloaded and after **BOOT**.
- **Timed. Task** runs at regular intervals which can be set by the developer. The default setting is 100ms.
- **Background. Task** runs with low priority after the Timed tasks (between the end of one Timed task and the start of the next).



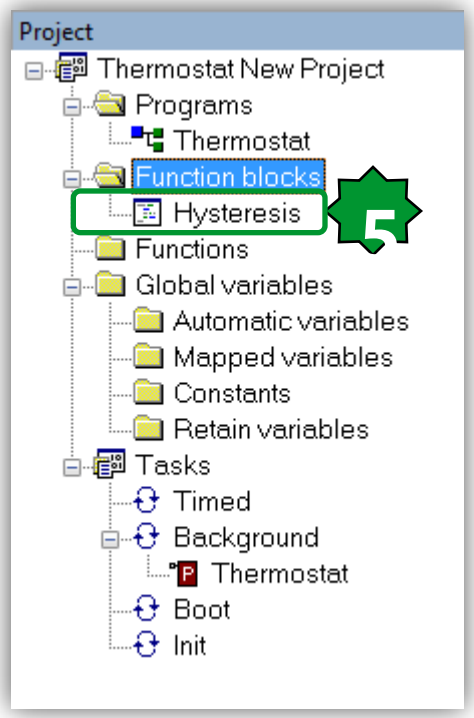
Assigning program to the task



View FBD toolbar

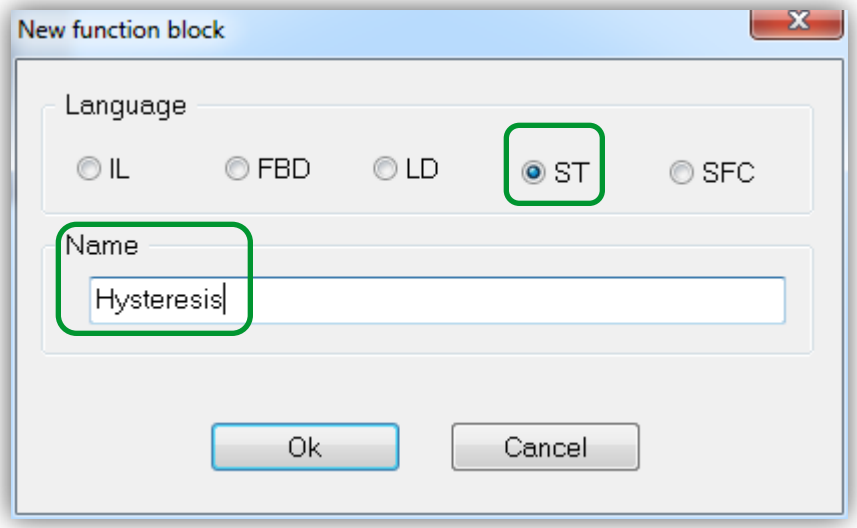


New function block creation

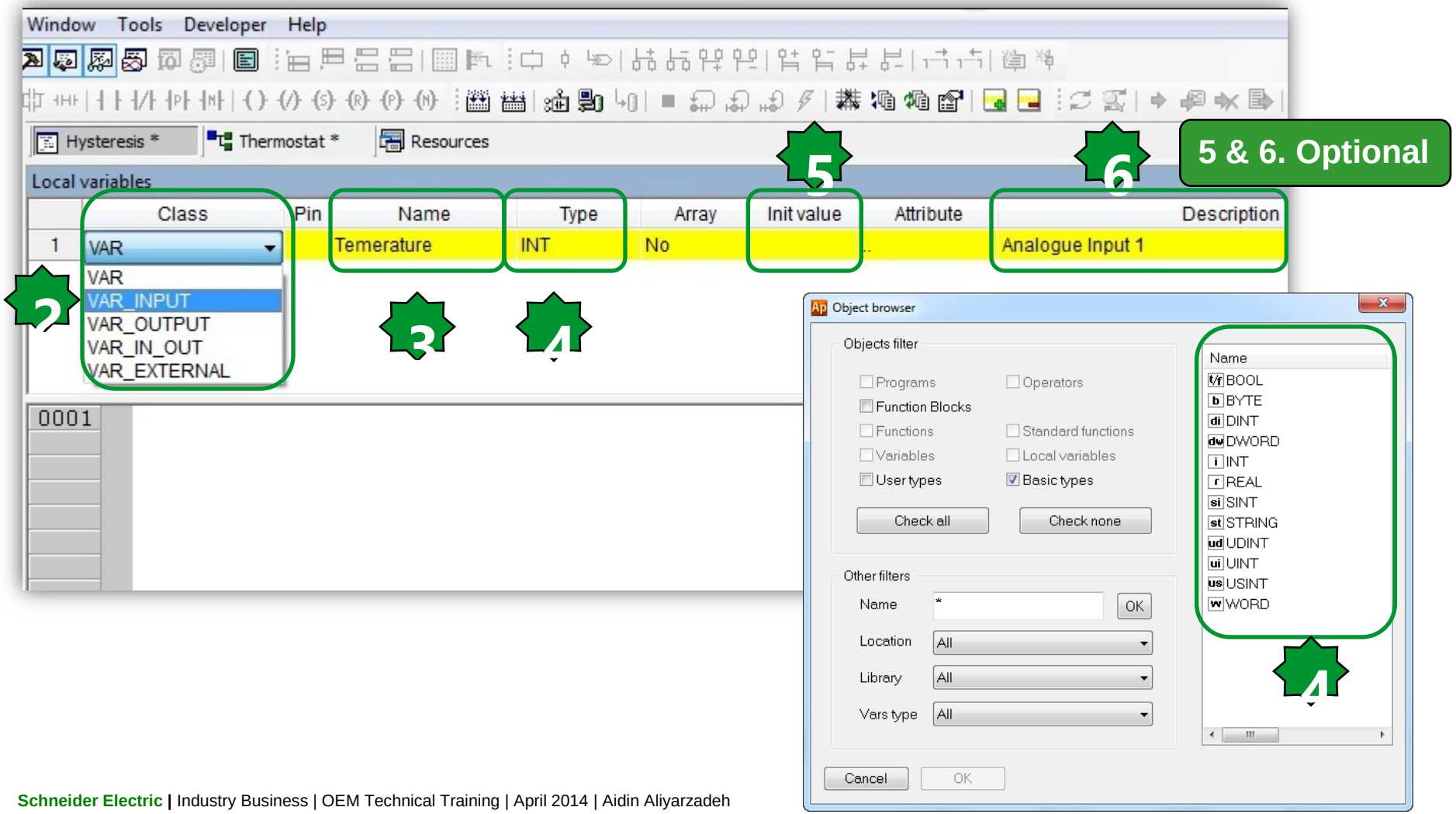


1. Right Click
2. New function block
3. Select the language
4. Assign a name
5. Double click on the Hysteresis to open the editor related to the selected language.

Note: Try to create function if the block does not require static RAM, it will optimize the RAM usage



Insert record



Inside Hystersis FBD

Local variables

	Class	Pin	Name	Type	Array	Init value	Attribute	Description
1	VAR_INPUT	0	Temperature	INT	No		..	Analogue Input 1
2	VAR_INPUT	1	Setpoint	INT	No		..	Set point
3	VAR_INPUT	2	Differentiation	INT	No		..	Δ
4	VAR_OUTPUT	0	Alarm	BOOL	No		..	Probe Alarm
5	VAR_OUTPUT	1	Output	BOOL	No		..	Actuator

```
0001  (* Hystersis FBD *)
0002
0003  if Temperature >= Setpoint + Differentiation then
0004      Output := TRUE;
0005  end_if;
0006
0007  if Temperature < Setpoint then
0008      Output := FALSE;
0009  end_if;
0010
0011  (* Probe disconnection detector *)
0012
0013  if Temperature = -32768 then
0014      Alarm := TRUE;
0015      else Alarm := FALSE;
0016  end_if;
0017
0018
```

Compile result is valid as soon as FBD used in the program

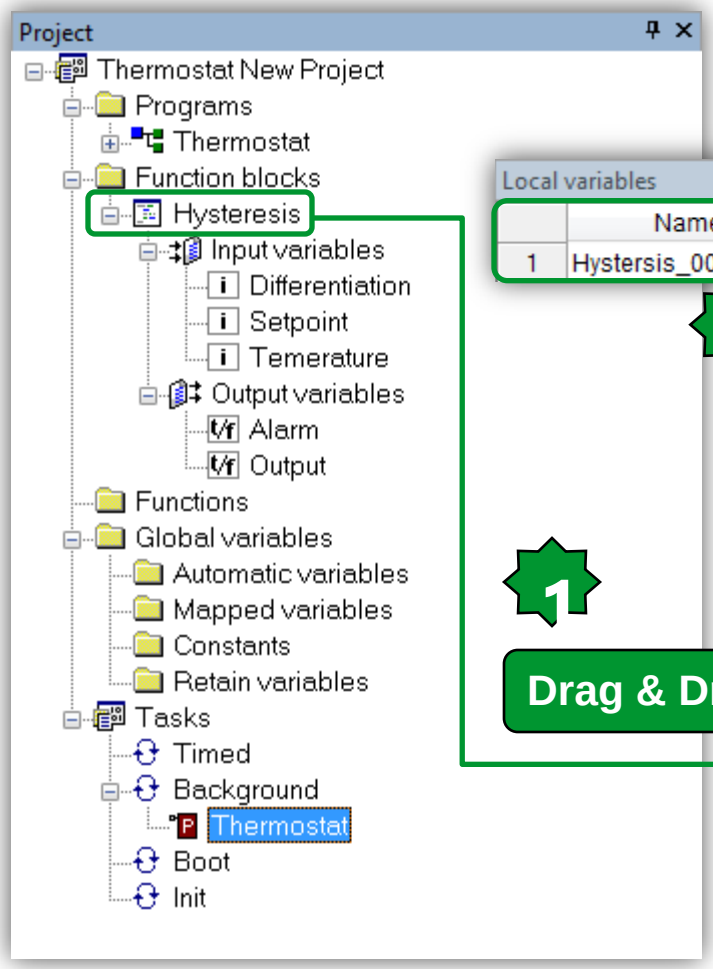
Output

```
Free code space:          2F1E0h    (   188 KByte)

Data space:                 8C0h    (     2 KByte)
Free data space:            8ABh    (     2 KByte)


0 warnings, 0 errors.
```

Build | Find in project | Debug | Resources

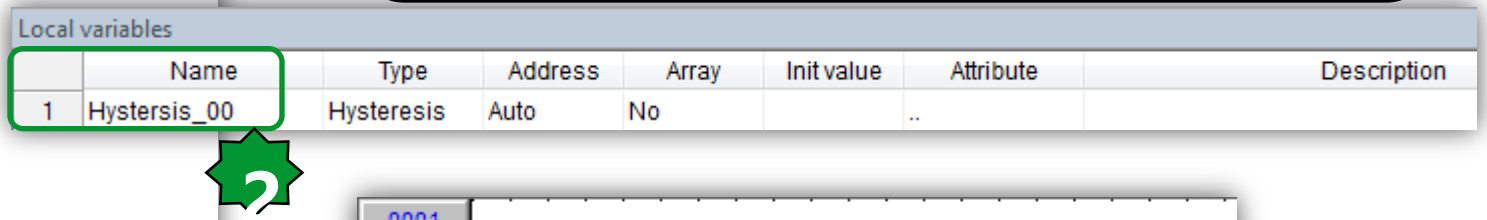
FBD in Background

1. Double click on the Thermostat programt to open the editor that is related to the program
2. drag & drop
3. Name it
4. Add new network

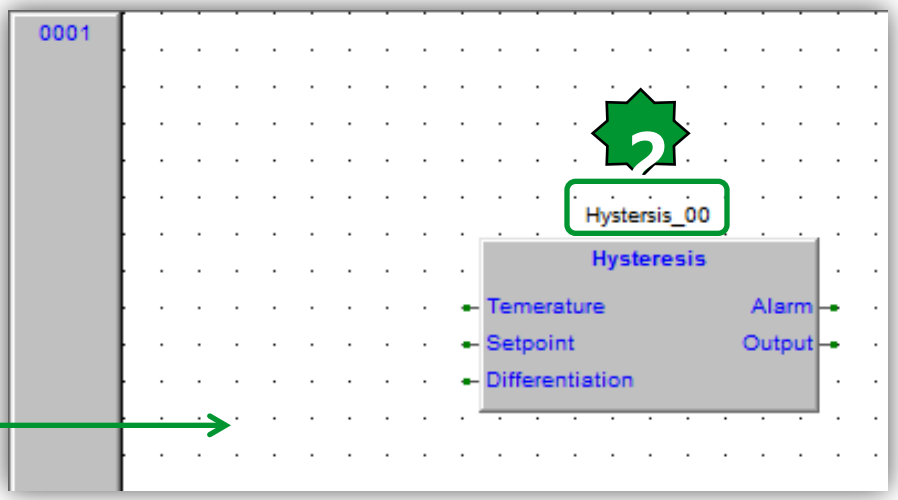


1

Drag & Drop



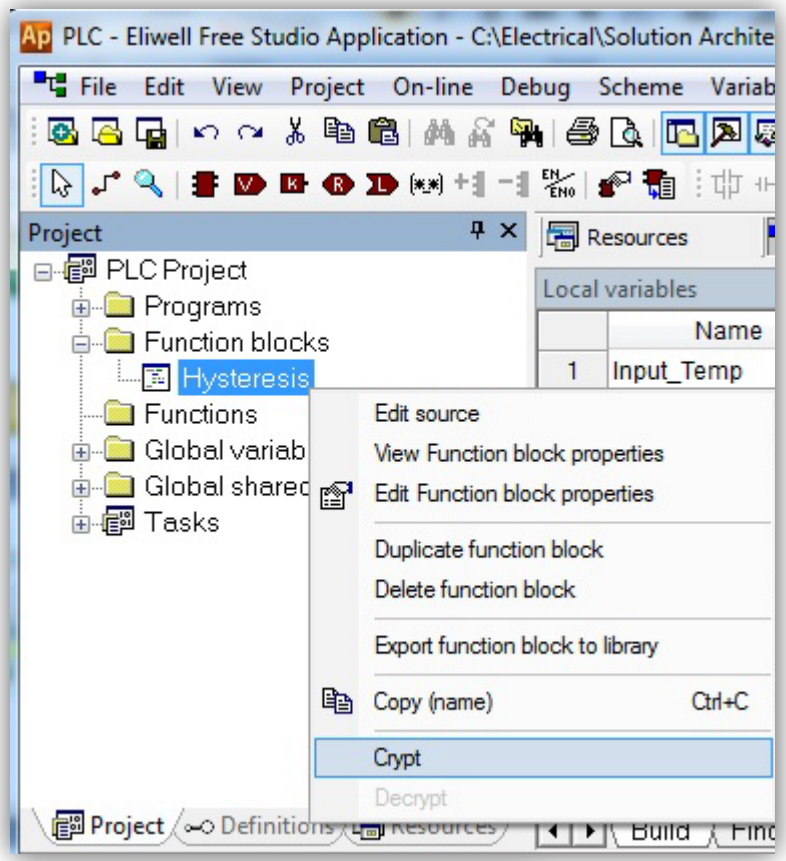
2

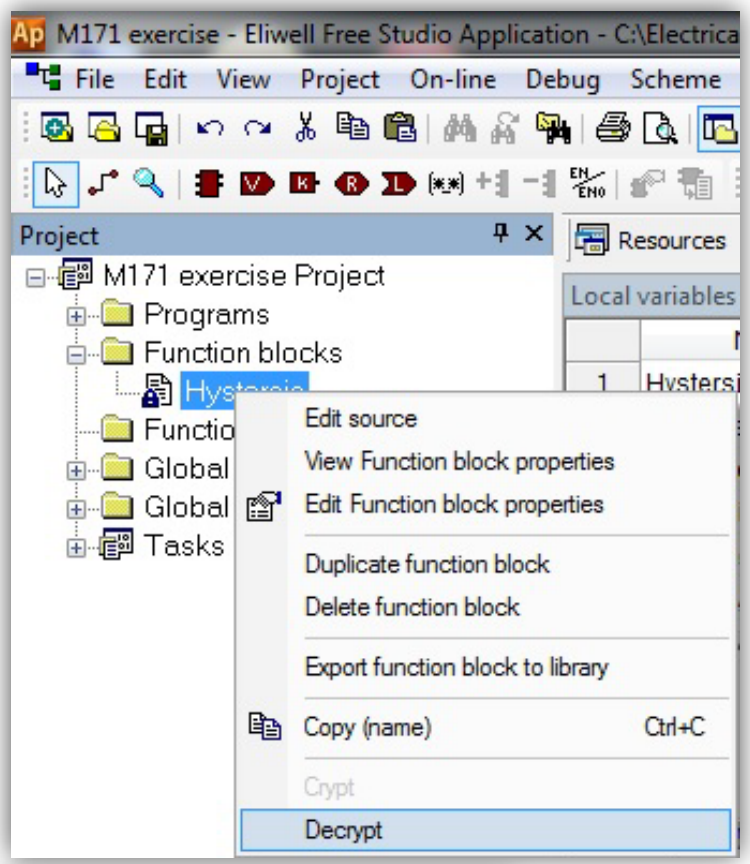


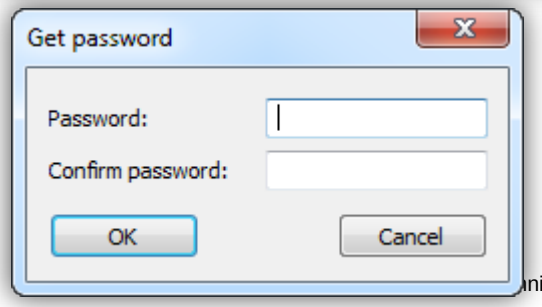
2

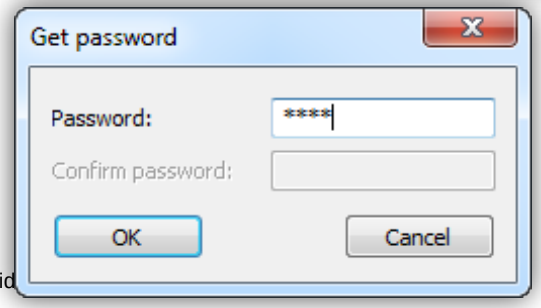
3

Set password for written FB



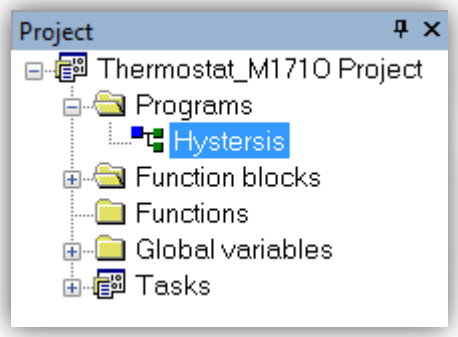






You can prevent access to your written codes inside of FBD by cript.

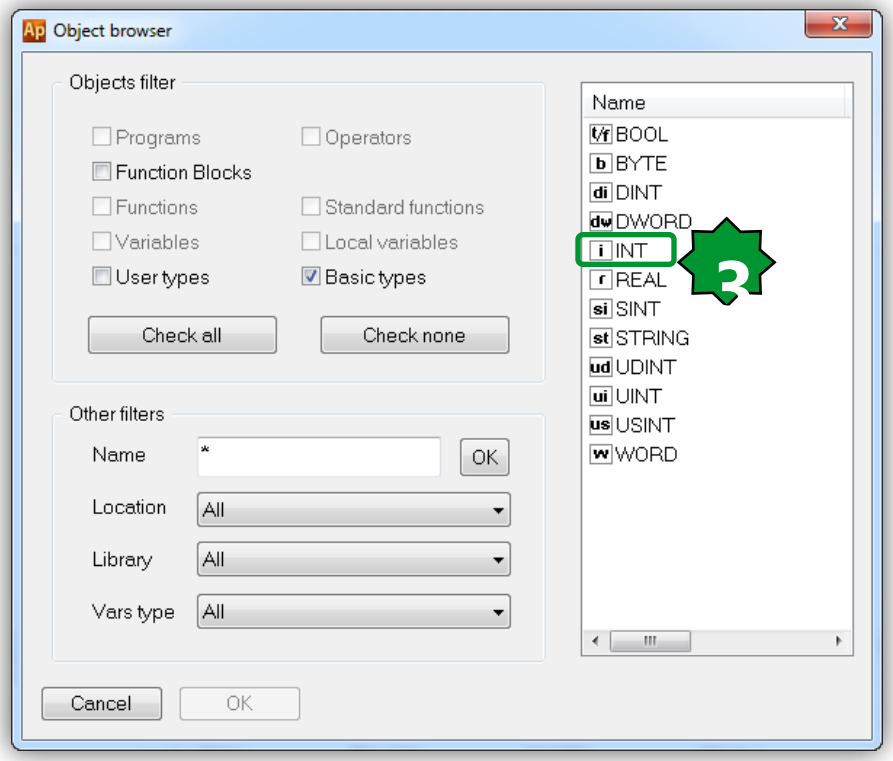
Assigning local variable to the FBD



	Name	Type	Address	Array	Init value	Attribute	Description
1	Hystersis_00	Hysteresis	Auto	No		..	
2	Input_Temp	INT ..	Auto	No		..	

2 3

1. **Add new record**
2. **Name it**
3. **Define the type from object browser**



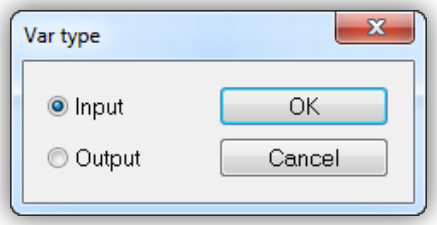
Connecting Variables to the FBD

Local variables

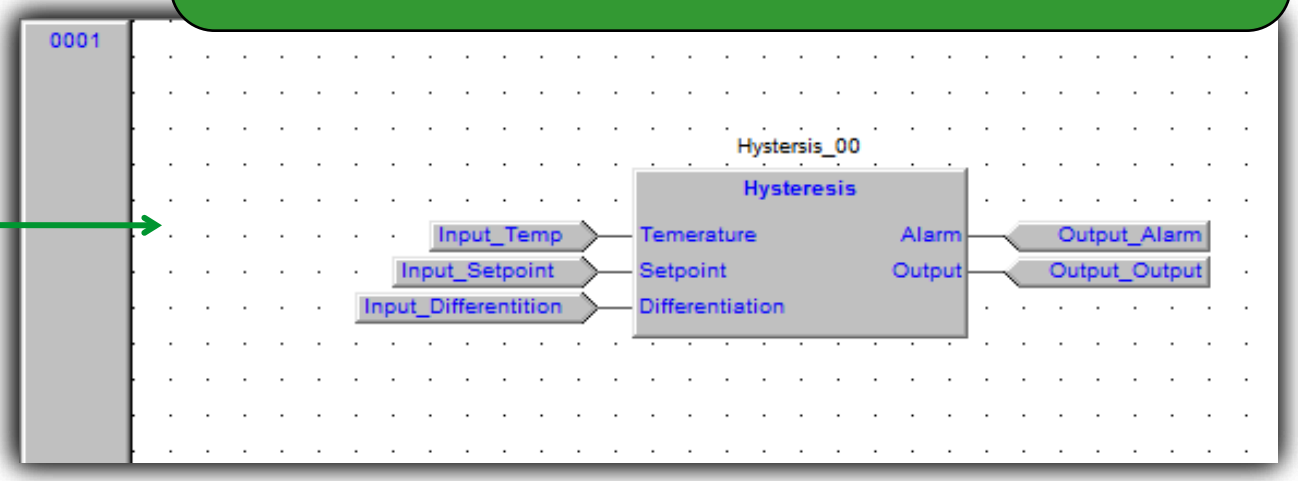
	Name	Type	Address	Array	Init value	Attribute	Description
1	Input_Temp	INT	Auto	No		..	
2	Input_Setpoint	INT	Auto	No		..	
3	Input_Differentiation	INT	Auto	No		..	
4	Output_Alarm	BOOL	Auto	No		..	
5	Output_Output	BOOL	Auto	No		..	
6	Hysteresis_00	Hysteresis	Auto	No			

Shortcuts:
Space: Connecting Mode ◄► Insert/Move Mode
Ctrl+Left Mouse Button: Multiple Selection
Ctrl+Shift+F: Find in project

Drag & Drop



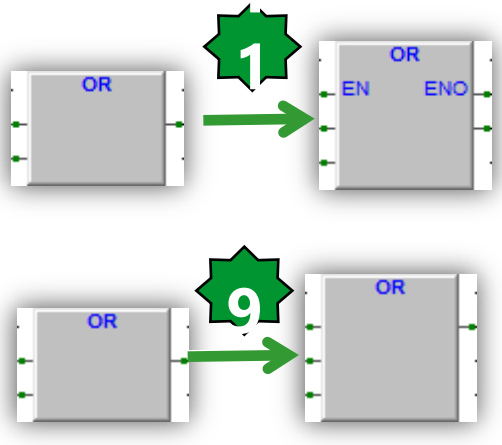
Select Input/output



FBD toolbar…

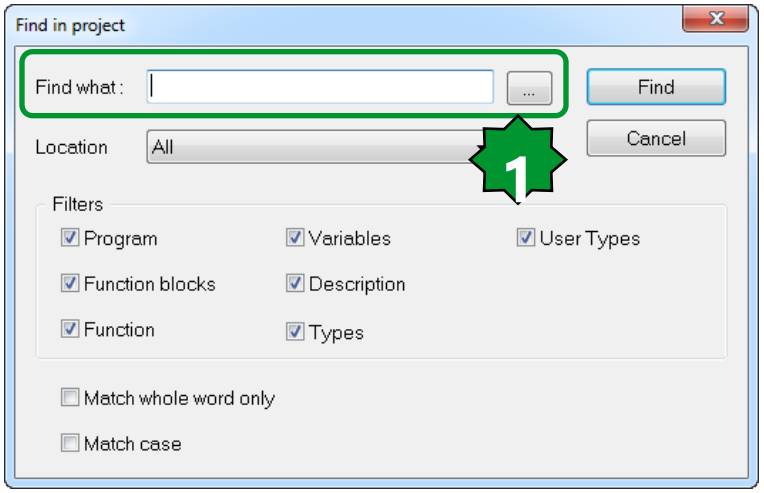
1. Connection
2. Watch
3. New block
4. Variable
5. Constant
6. Return
7. Jump
8. Comment
9. Increase number of pins
10. Decrease number of pins
11. Display enable I/O pins
12. FBD properties
13. View source

11. The output will not update if En=False



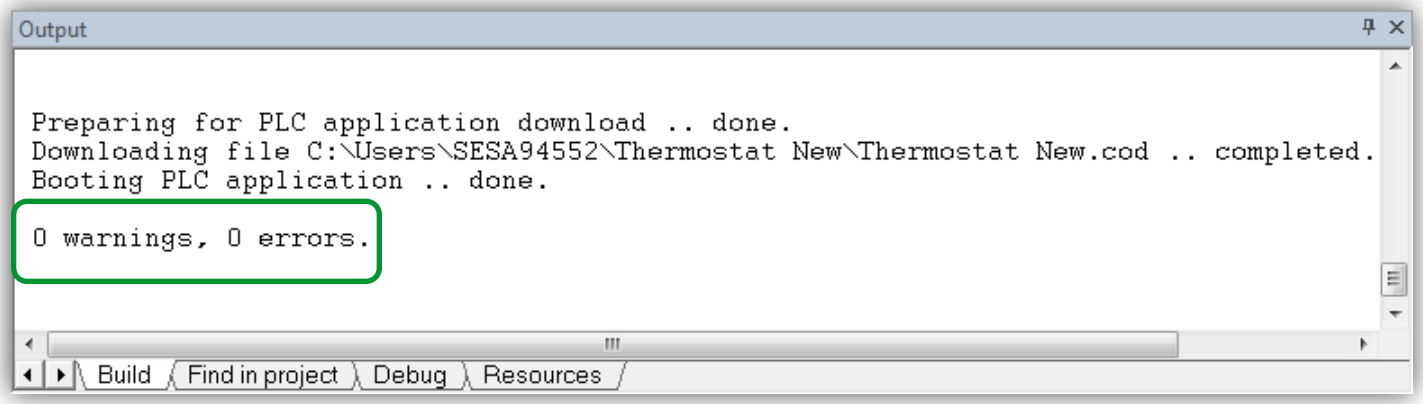
Cross Reference

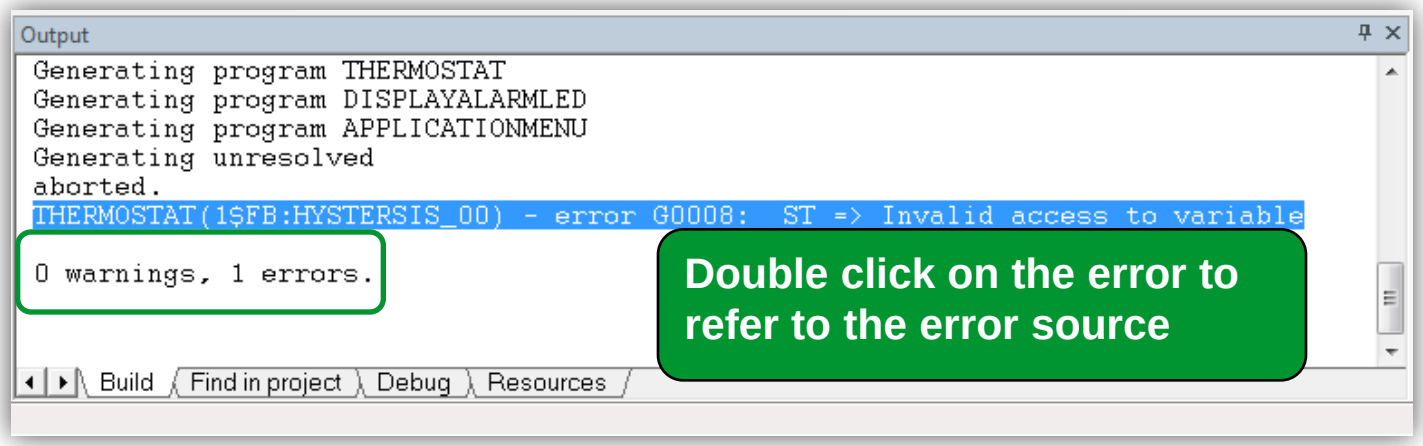
1. Find in project (cross reference)



Compile/Build

Compile





Double click on the error to refer to the error source

Chapter 3

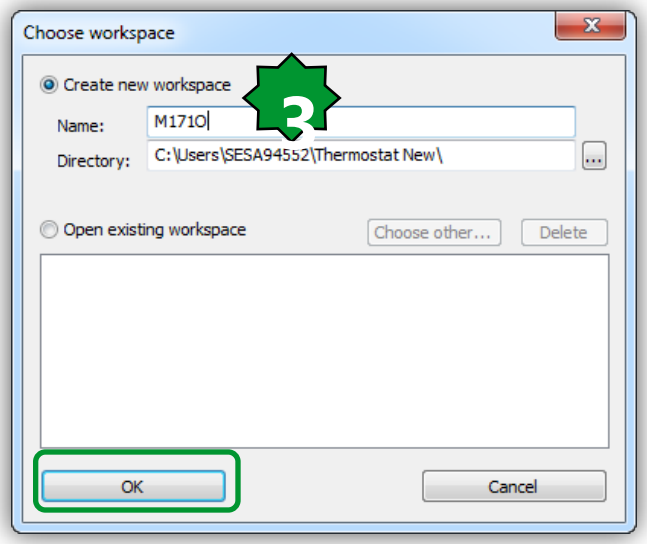
Simulation/Debugging – Part 1

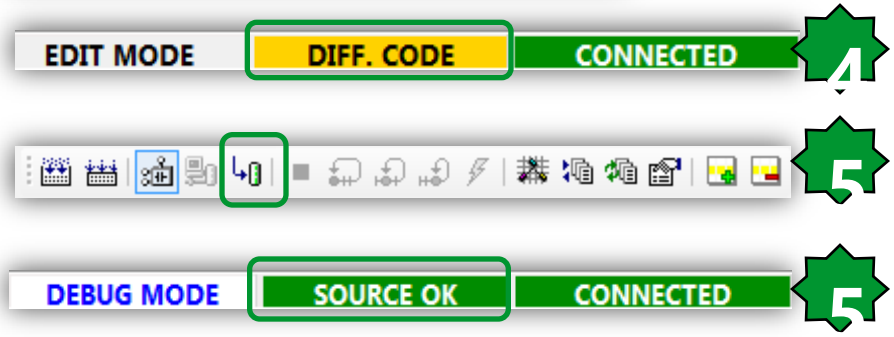
Goal:

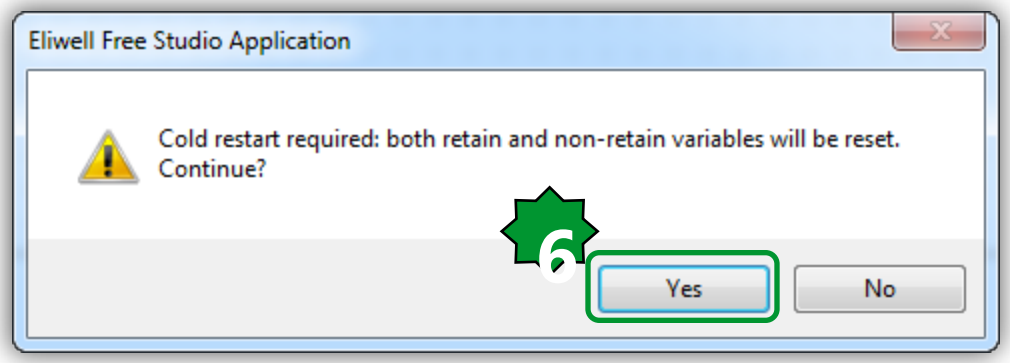
Debugging created FB by different off-line simulation tools such as Watch or Oscilloscope

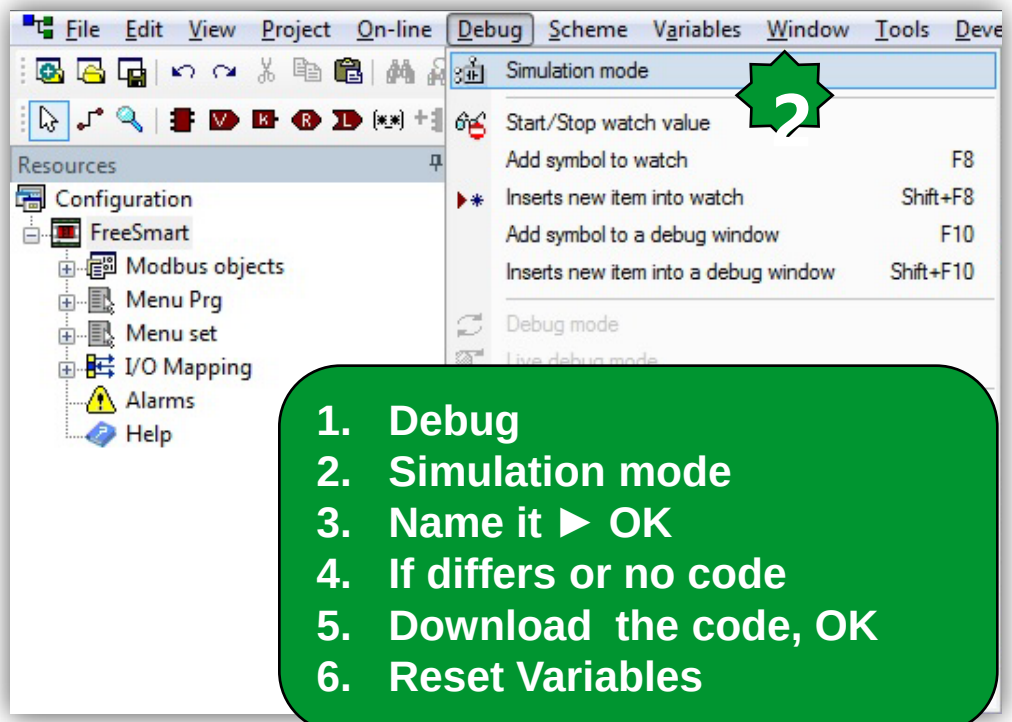
Off line simulation mode

Debug ► Simulation mode ►









1. Debug
2. Simulation mode
3. Name it ► OK
4. If differs or no code
5. Download the code, OK
6. Reset Variables

On-Line Status / Application Status

The state of communication is shown in a small box next to the right border of the **Status bar.**

If you have not yet attempted to connect to the target, the state of communication is set to **Not connected.**

When you try to connect to the target device, the state of communication becomes one of the following:

-Error: the communication cannot be established. You should check both the physical link and the communication settings.

-Connected: the communication has been established

On-Line Status / Application Status

Next to the communication status there is another small box which gives information about the status of the application currently executing on the target device.

When the connection status is Connected, the application status takes on one of the following values.

-No code: no application is executing on the target device.

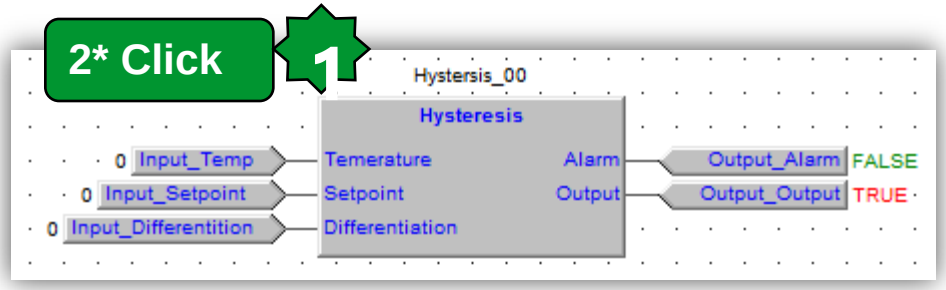
--Diff. code: the application currently executing on the target device is not the same as the one currently open in the IDE; moreover, no debug information consistent with the running application is available: thus, the values shown in the watch window or in the oscilloscope are not reliable and the debug mode cannot be activated.

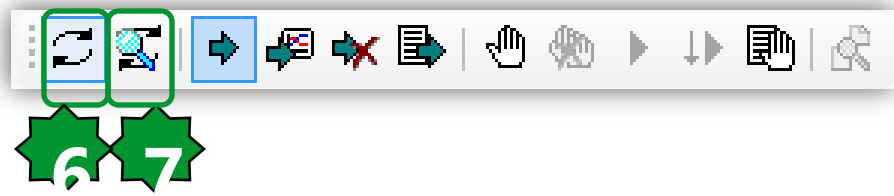
On-Line Status / Application Status

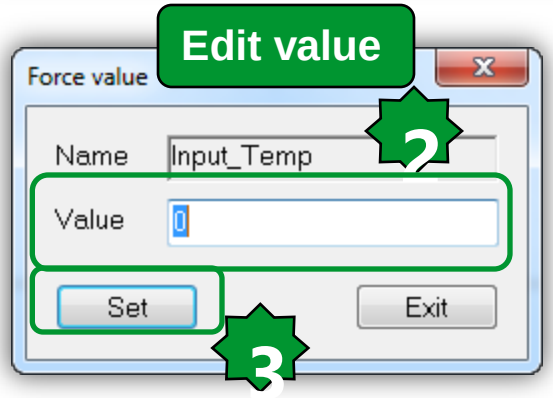
--Diff. code, Symbols OK: the application currently executing on the target device is not the same as the one currently open in the IDE; however, some debug information consistent with the running application is available (for example, because that application has been previously downloaded to the target device from the same PC): the values shown in the watch window or in the oscilloscope are reliable, but the debug mode still cannot be activated.

-Source OK: the application currently executing on the target device is the same as the one currently open in the IDE: the debug mode can be activated.

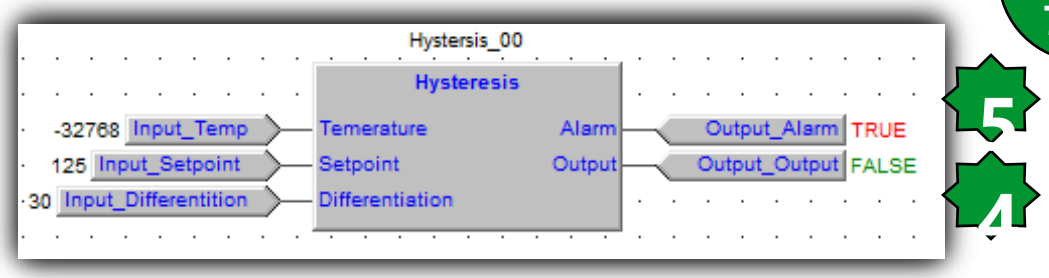
Debug mode/Changing values







1. 2*click on required variable
2. Edit the value
3. Set the values
4. Check the Output status
5. Check the Alarm status
   In probe disconnection, short circuited or broken the value= - 32768
6. Debug mode (optional)
7. Live (continuous) debug mode (optional)



Watch configuration

Local variables

	Name	Type	Address	Array	Init value	Attribute	Description
1	Hystersis_00	Hysteresis	Auto	No		..	
2	Input_Temp	INT	Auto	No		..	
3	Input_Setpoint	INT	Auto	No		..	
4	Input_Differentition	INT	Auto	No		..	
5	Output_Alarm	BOOL	Auto	No		..	
6	Output_Output	BOOL	Auto	No		..	

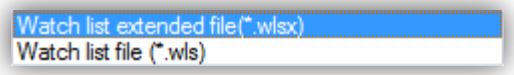
Watch list is independant from live debug mode

Drag & Drop

Watch

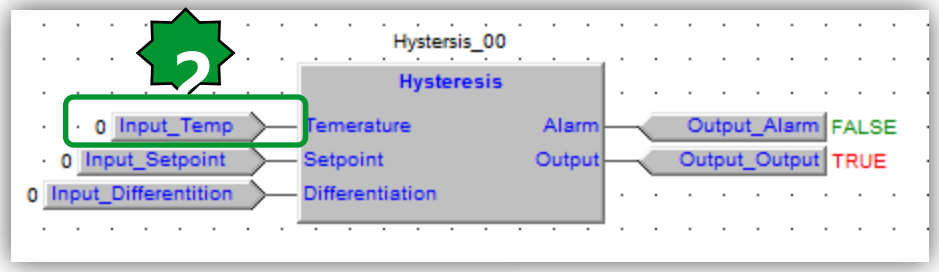
Symbol	Value	Type	Location
INPUT_TEMP	179	INT	@BACKGROUND:THERMOSTAT
INPUT_SETPOINT	125	INT	@BACKGROUND:THERMOSTAT
INPUT_DI...	30	INT	@BACKGROUND:THERMOSTAT
OUTPUT_ALARM	FALSE	BOOL	@BACKGROUND:THERMOSTAT
OUTPUT_OUTPUT	TRUE	BOOL	@BACKGROUND:THERMOSTAT

Save watch list

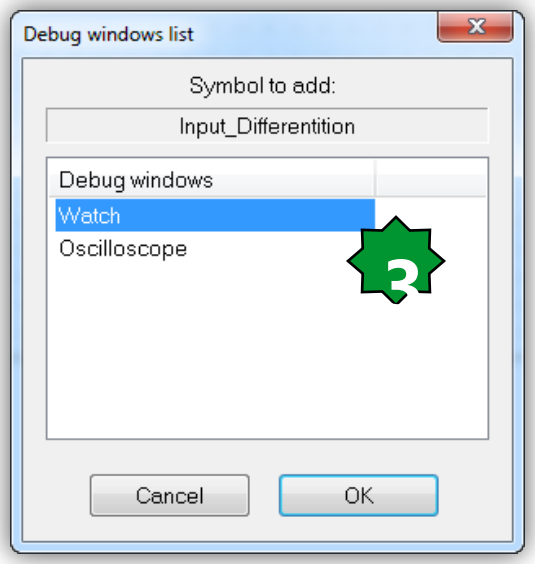


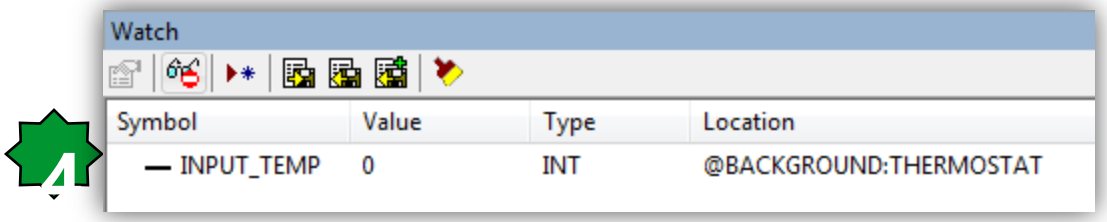
Watch list formats

Watch/ drag & drop



1. Select watch icon
2. Select the variable
3. Select the watch
4. ► OK
5. It adds to watch list





Watch Configuration/ST language

```
0001  (* Hystersis FBD *)
0002
0003  if Temperature >= Setpoint + Differentiation then
0004      Output := TRUE;
0005  end_if;
0006
0007  if Temperature < Setpoint then
0008      Output := FALSE;
0009  end_if;
0010
0011  (* Probe disconnection detector *)
0012
0013  if Temperature = -32768 then
0014      Alarm := TRUE;
0015      else Alarm := FALSE;
0016  end_if;
0017
0018
```

1. Select the variale
2. Double click
3. Drag & drop it directly to the watch properties

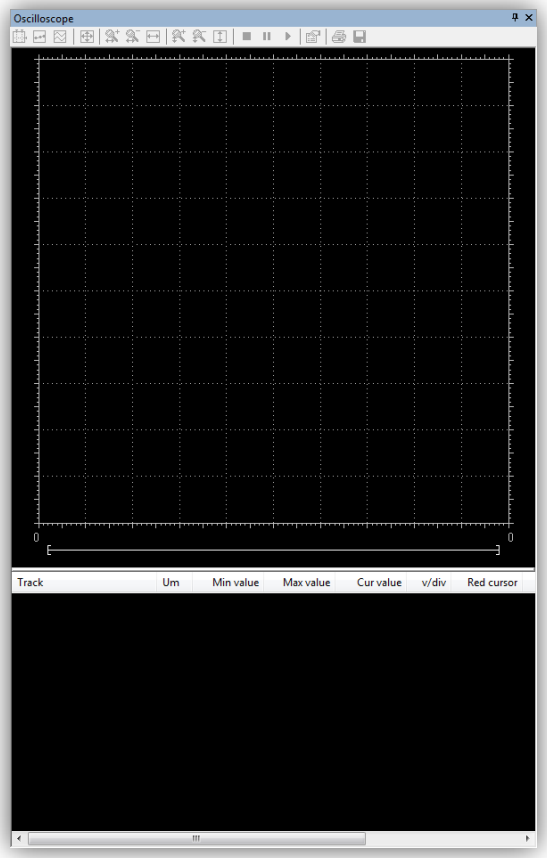
```
0001  (* Hystersis FBD *)
0002
0003  if Temperature >= Setpoint + Differentiation then
0004      Output := TRUE;
0005  end_if;
0006
0007  if Temperature < Setpoint then
0008      Output := FALSE;
0009  end_if;                    120  *
0010
0011  if Temperature = -32768 then
0012      Alarm := TRUE;
0013      else Alarm := FALSE;
0014  end_if;
0015
```

Watch

Symbol	Value	Type	Location
HYSTERESIS_00.TEMPERATURE	0	INT	@BACKGROUND:THERMOSTAT

Oscilloscope

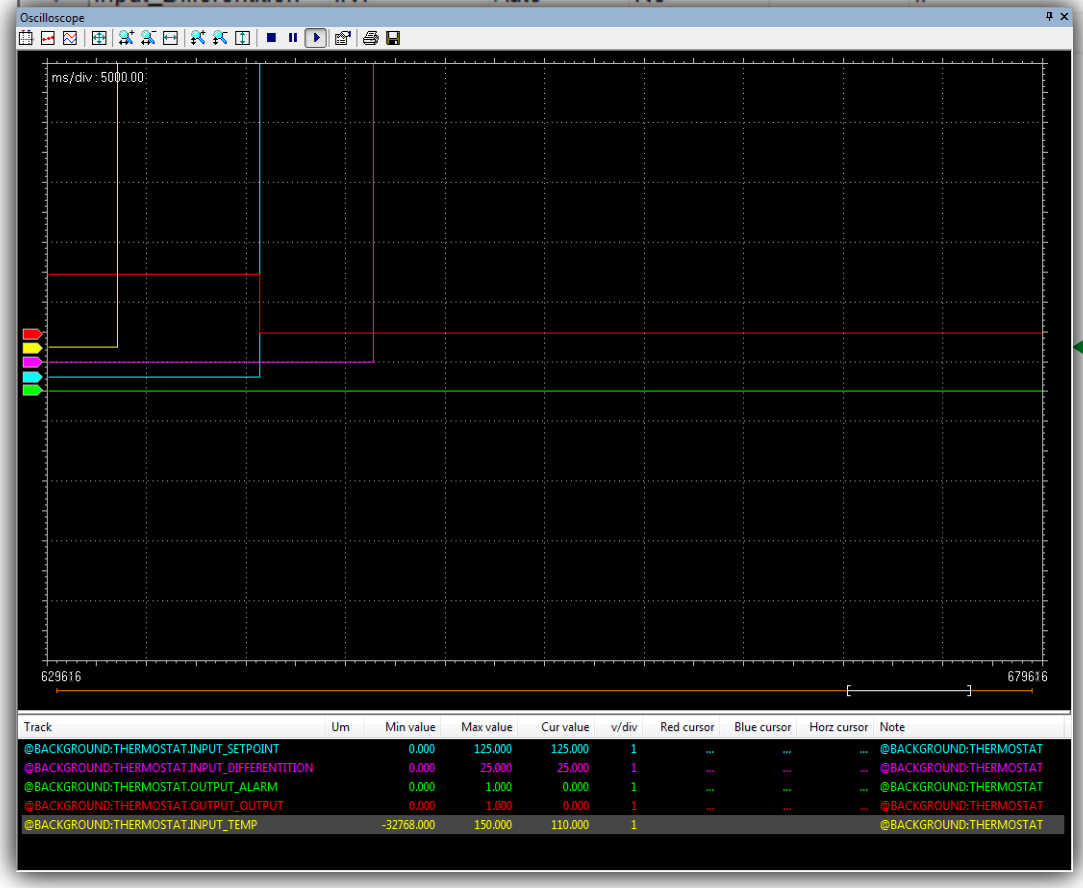
View ►Tool windows ► Async graphic windows ►



Assigning variable to the oscilloscope

Local variables — Thermostat

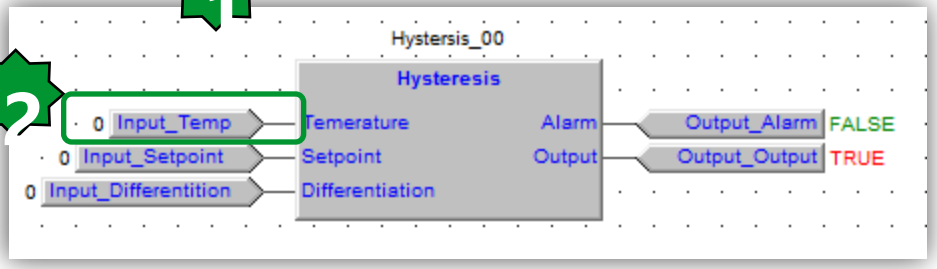
	Name	Type	Address	Array	Init value	Attribute	Description
1	Hysteresis_00	Hysteresis	Auto	No		..	
2	Input_Temp	INT	Auto	No		..	
3	Input_Setpoint	INT	Auto	No		..	
4	Input_Differentition	INT	Auto	No		..	



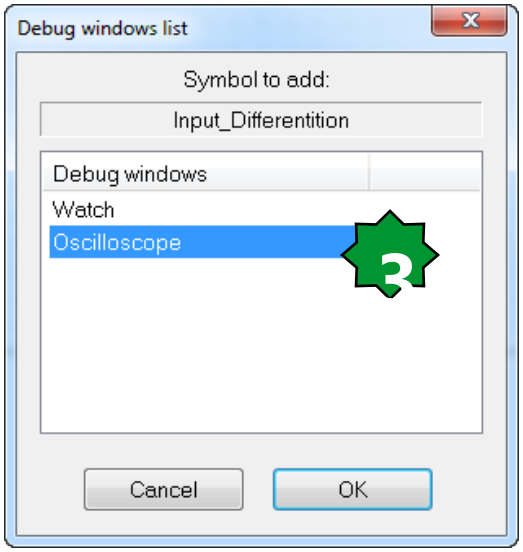
Drag & Drop

Up to 8 channels could be traced simultaneously

Assigning variable to the oscilloscope



1. Select watch icon
2. Select the variable
3. Select the Oscilloscope
4. It adds to oscilloscope list



Oscilloscope tools/starting & stopping data acquisition

When you add a variable to the Oscilloscope, data acquisition begins immediately.

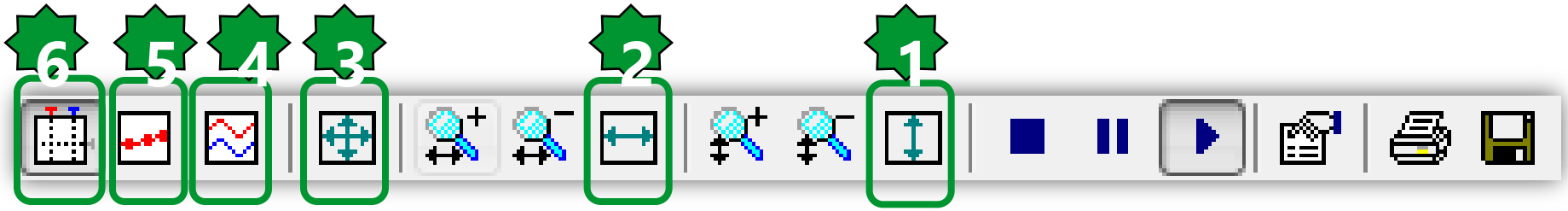
However, you can suspend the acquisition by clicking on **Pause acquisition.**

The curve freezes (while the process of data acquisition is still running in background), until you click on **Restart acquisition.**

In order to stop the acquisition you may click on **Stop acquisition.**

In this case, when you click on **Restart acquisition, the evolution of the value of the variable is plotted from scratch.**

Oscilloscope tools/ Vertical split



1.Selected track‘s vertical show all

2. Horizental show all

3. Show all values

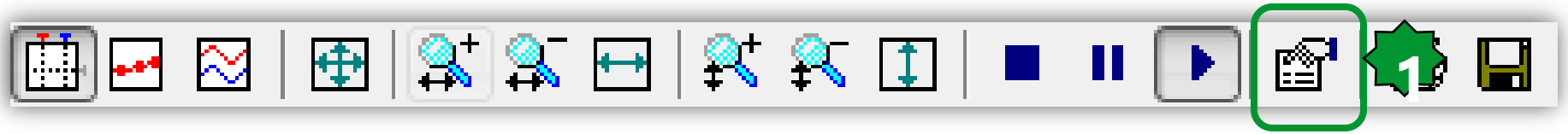
4. When you are watching the evolution of two or more variables, you may want to split the respective tracks.

5. The tool highlights the single values detected during data acquisition.

You can click on the same item again, in order to go back to the default view mode.

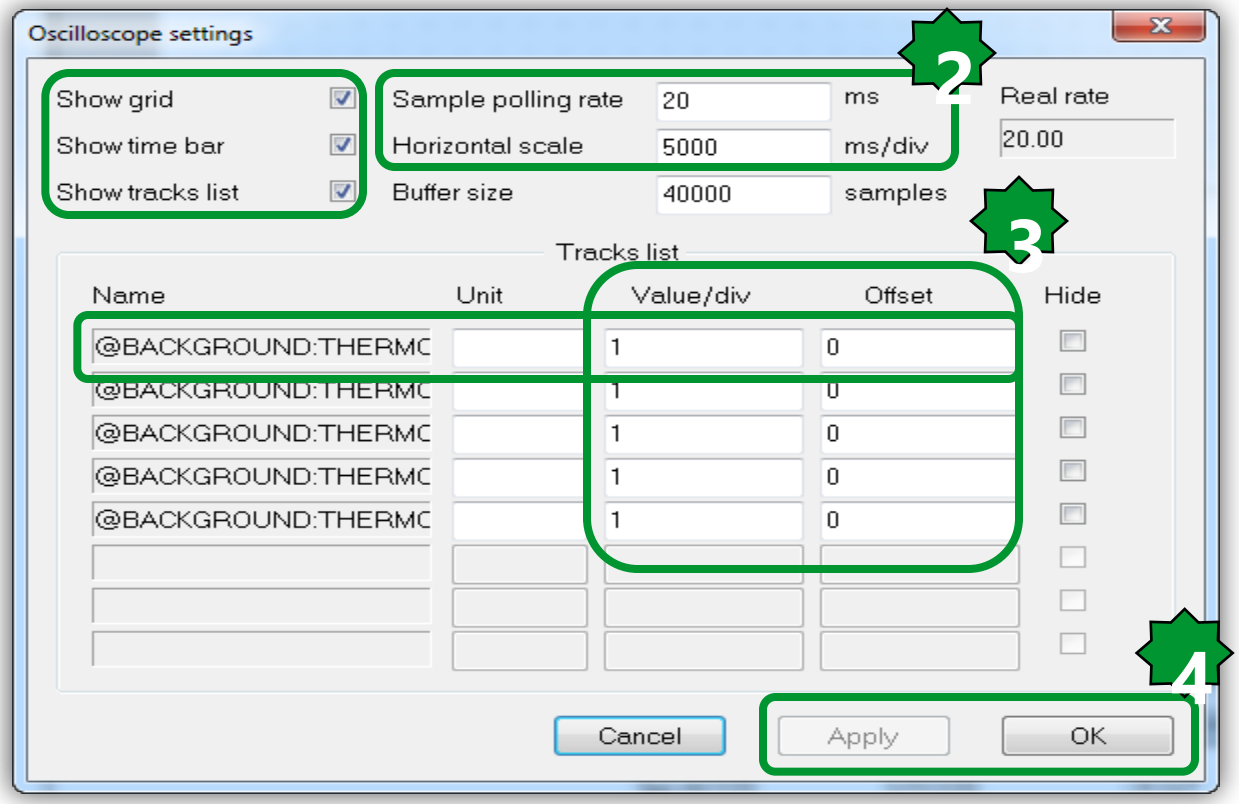
6. The Oscilloscope includes two measure bars, which can be exploited to take some measures on the chart.

Oscilloscope tools/ Setting the scale of the axes



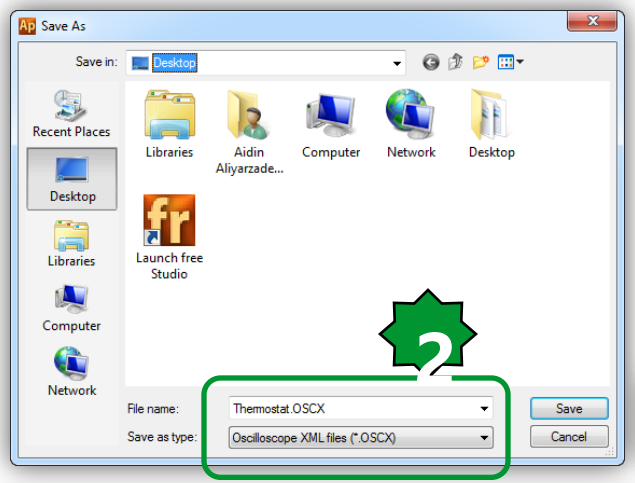
When you open the Oscilloscope, Application applies a default scale to the axes. However, if you want to set a different scale, you may follow this procedure:

1) Open the graph properties
2) Set the scale of the horizontal axis & sampling polling rate
3) Specify a distinct scale for the vertical axis.
4) Confirm your settings.



Oscilloscope/export

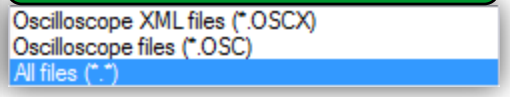
1



2

2

Available formats



1. Save icon
2. Name & format defining
   - -OSC: simple plain-text file, containing time and value of each sample
   - -OSCX: XML file, that includes more complete information
3. Open it via Excel (OSCX)

	A	B	C	D	E	F	G	H	I	J
1	hscale	triggerpos	name	um	vscale	offset	color	note	sample	time
2	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870186
3	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870205.9
4	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870226.2
5	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870246.1
6	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870266.2
7	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870286.2
8	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870306.1
9	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870326
10	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870346.1
11	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870366
12	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870386.2
13	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870406.1
14	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870426.2
15	55536.85563	1.79769313486232E+308	@BACKGROUND:THERMOSTAT.NTC_PROBE		25323.57143	111271.0714	65535	@BACKGROUND:THERMOSTAT	0	232870445.9

3

Chapter 4

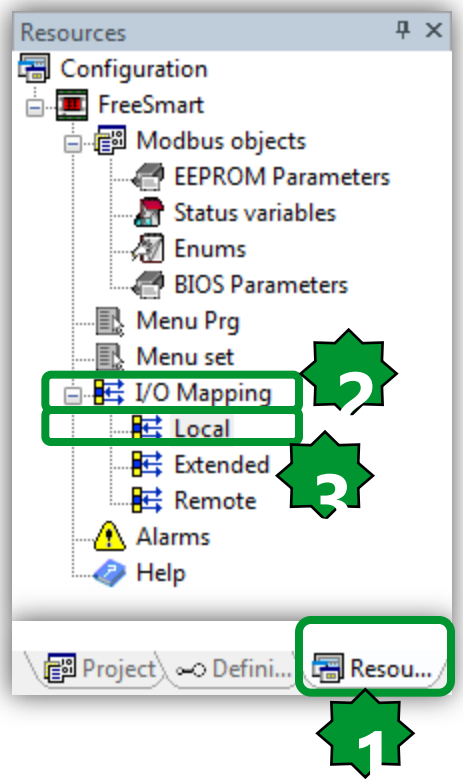
Resources

Goal:

Defining the resources:

- Assigning physical Input/output
- EEPROM parameters
- Status variables
- Menu definition and navigation

Physical I/O Mapping (Base Unit)...



FreeSmart Local I/O Mapping

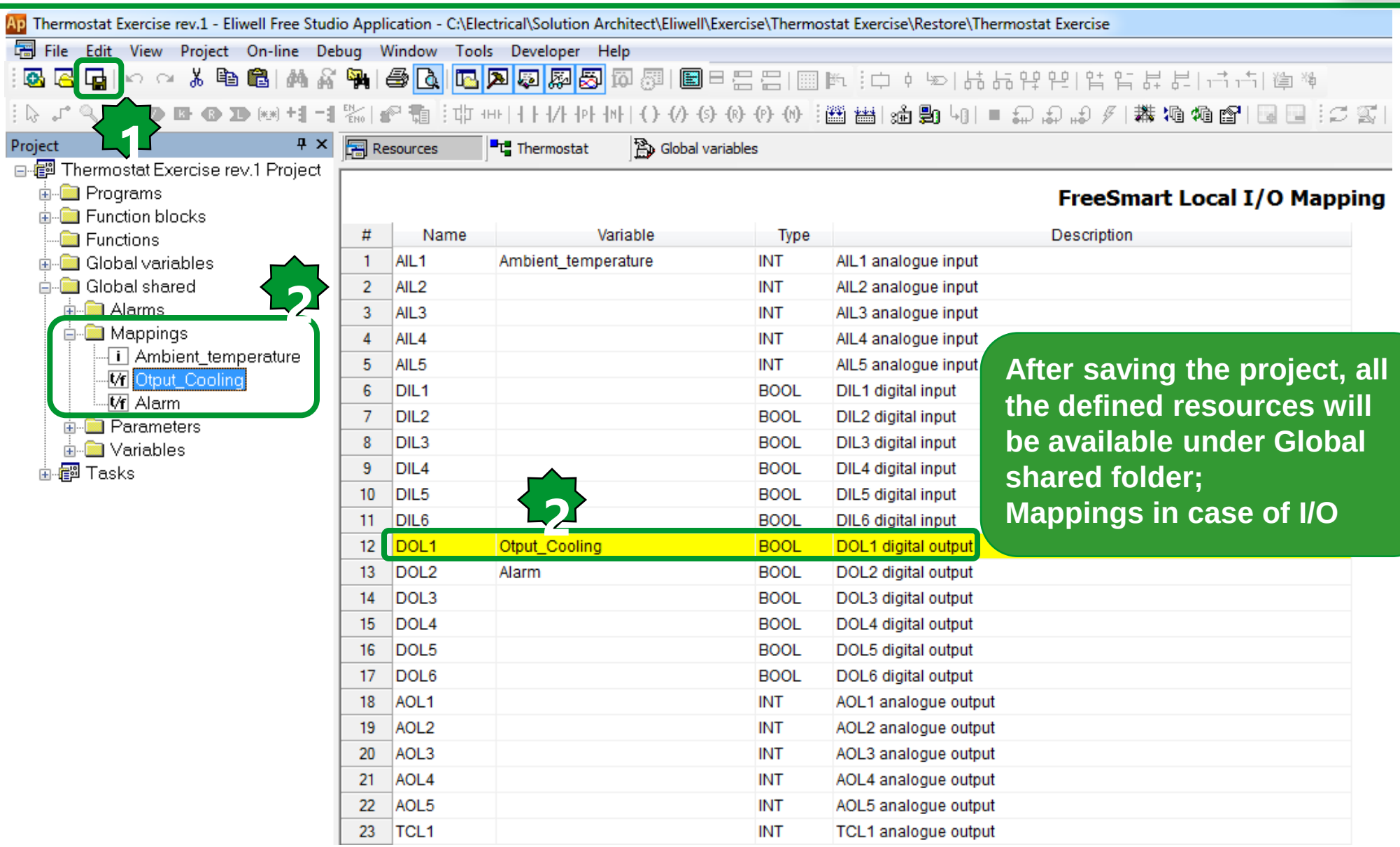
#	Name	Variable	Type	Description
1	AIL1	NTC_Probe	INT	AIL1 analogue input
2	AIL2		INT	AIL2 analogue input
3	AIL3		INT	AIL3 analogue input
4	AIL4		INT	AIL4 analogue input
5	AIL5		INT	AIL5 analogue input
6	DIL1		BOOL	DIL1 digital input
7	DIL2		BOOL	DIL2 digital input
8	DIL3		BOOL	DIL3 digital input
9	DIL4		BOOL	DIL4 digital input
10	DIL5		BOOL	DIL5 digital input
11	DIL6		BOOL	DIL6 digital input
12	DOL1	Output_Cooling	BOOL	DOL1 digital output
13	DOL2	Alarm	BOOL	DOL2 digital output
14	DOL3		BOOL	DOL3 digital output
15	DOL4		BOOL	DOL4 digital output
16	DOL5		BOOL	DOL5 digital output
17	DOL6		BOOL	DOL6 digital output
18	AOL1		INT	AOL1 analogue output
19	AOL2		INT	AOL2 analogue output
20	AOL3		INT	AOL3 analogue output
21	AOL4		INT	AOL4 analogue output
22	AOL5		INT	AOL5 analogue output
23	TCL1		INT	TCL1 analogue output

2. I/O Mapping definition:

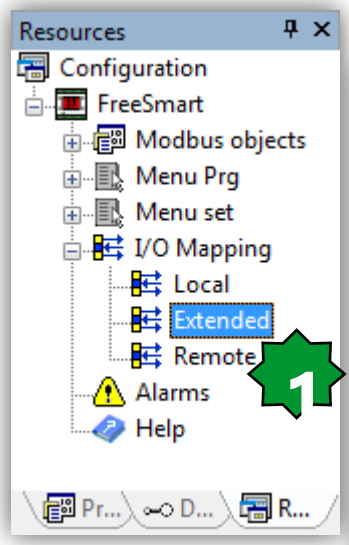
Local: Base I/O
Extend: Expansion
Remote: Keyboard

1. **Resources**
2. **I/O mapping**
3. **Local**
4. **Name variables**

...Physical I/O Mapping (Base Unit)...

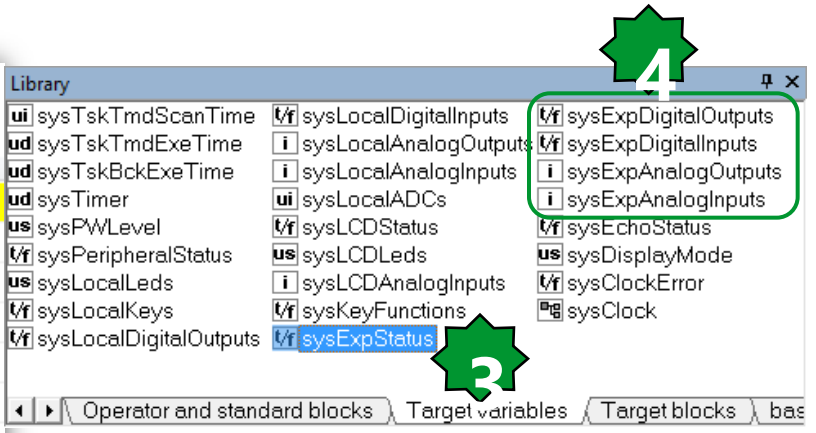


...Physical I/O Mapping (Expansion)



#	Na...	Variable	Type	Description
1	AIE1		INT	AIE1 analogue input
2	AIE2		INT	AIE2 analogue input
3	AIE3		INT	AIE3 analogue input
4	AIE4		INT	AIE4 analogue input
5	AIE5		INT	AIE5 analogue input
6	DIE1		BOOL	DIE1 digital input
7	DIE2		BOOL	DIE2 digital input
8	DIE3		BOOL	DIE3 digital input
9	DIE4		BOOL	DIE4 digital input
10	DIE5		BOOL	DIE5 digital input
11	DIE6		BOOL	DIE6 digital input
12	DOE1		BOOL	DOE1 digital output
13	DOE2		BOOL	DOE2 digital output
14	DOE3		BOOL	DOE3 digital output
15	DOE4		BOOL	DOE4 digital output
16	DOE5		BOOL	DOE5 digital output
17	DOE6		BOOL	DOE6 digital output
18	AOE1		INT	AOE1 analogue output
19	AOE2		INT	AOE2 analogue output
20	AOE3		INT	AOE3 analogue output
21	AOE4		INT	AOE4 analogue output
22	AOE5		INT	AOE5 analogue output
23	TCE1		INT	TCE1 analogue output

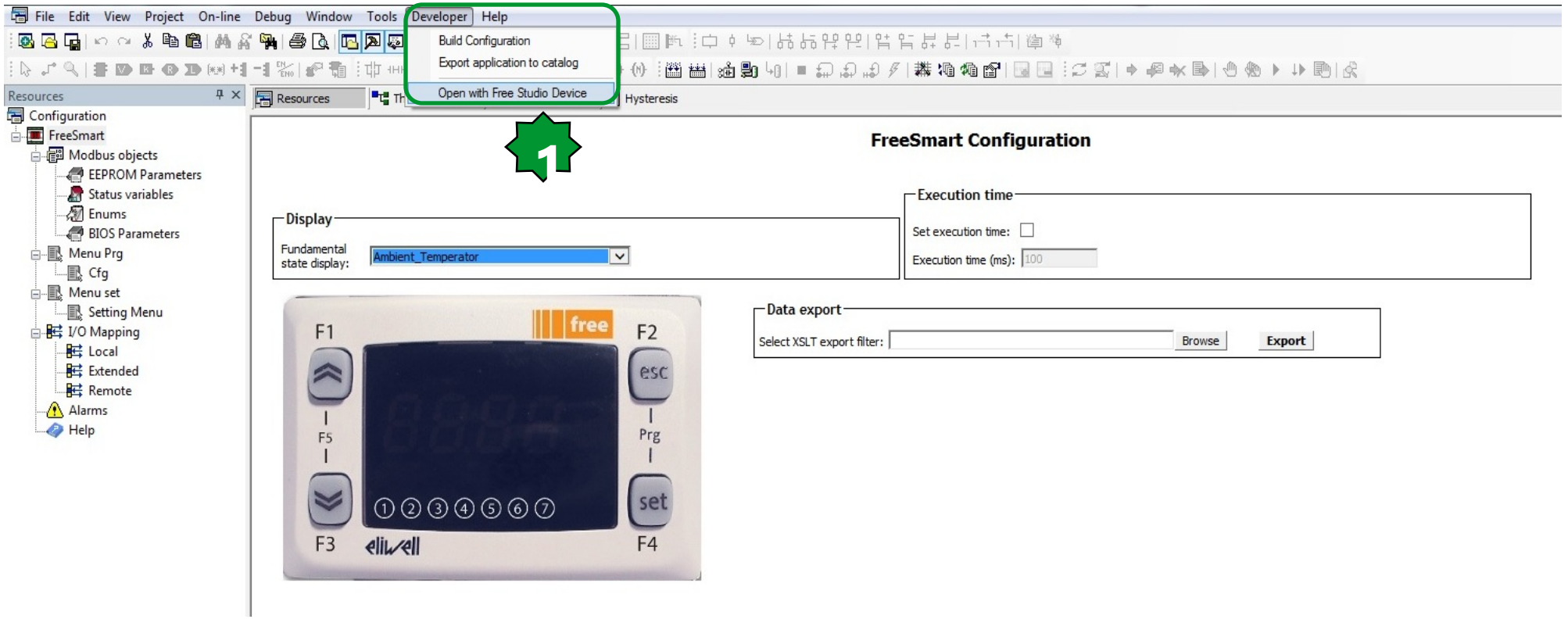
2



Note.
In case of loosing communiucation between Base & expansion:
All DO‘s = 0
All DI‘s = False
All Probes = -32768

3. It can use as communication alarm variable
4. They can only use in watch

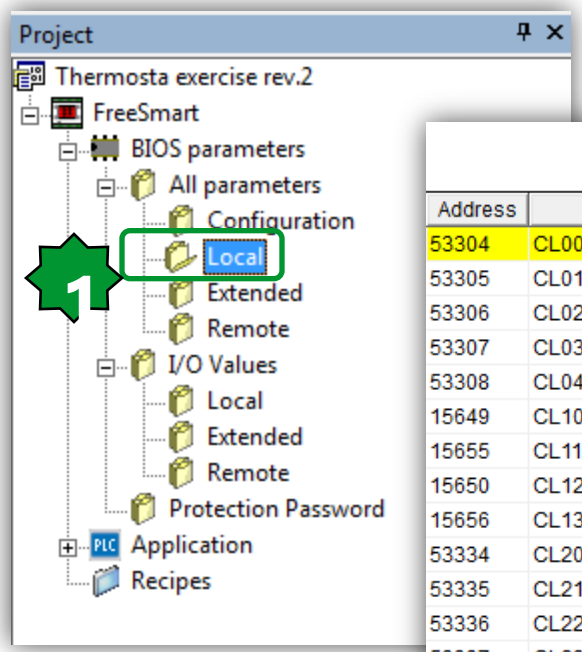
How to configure I/O types, range?



1. Menu Developer ► Open with free studio device

Note: BIOS parameters are also available on the installation manual

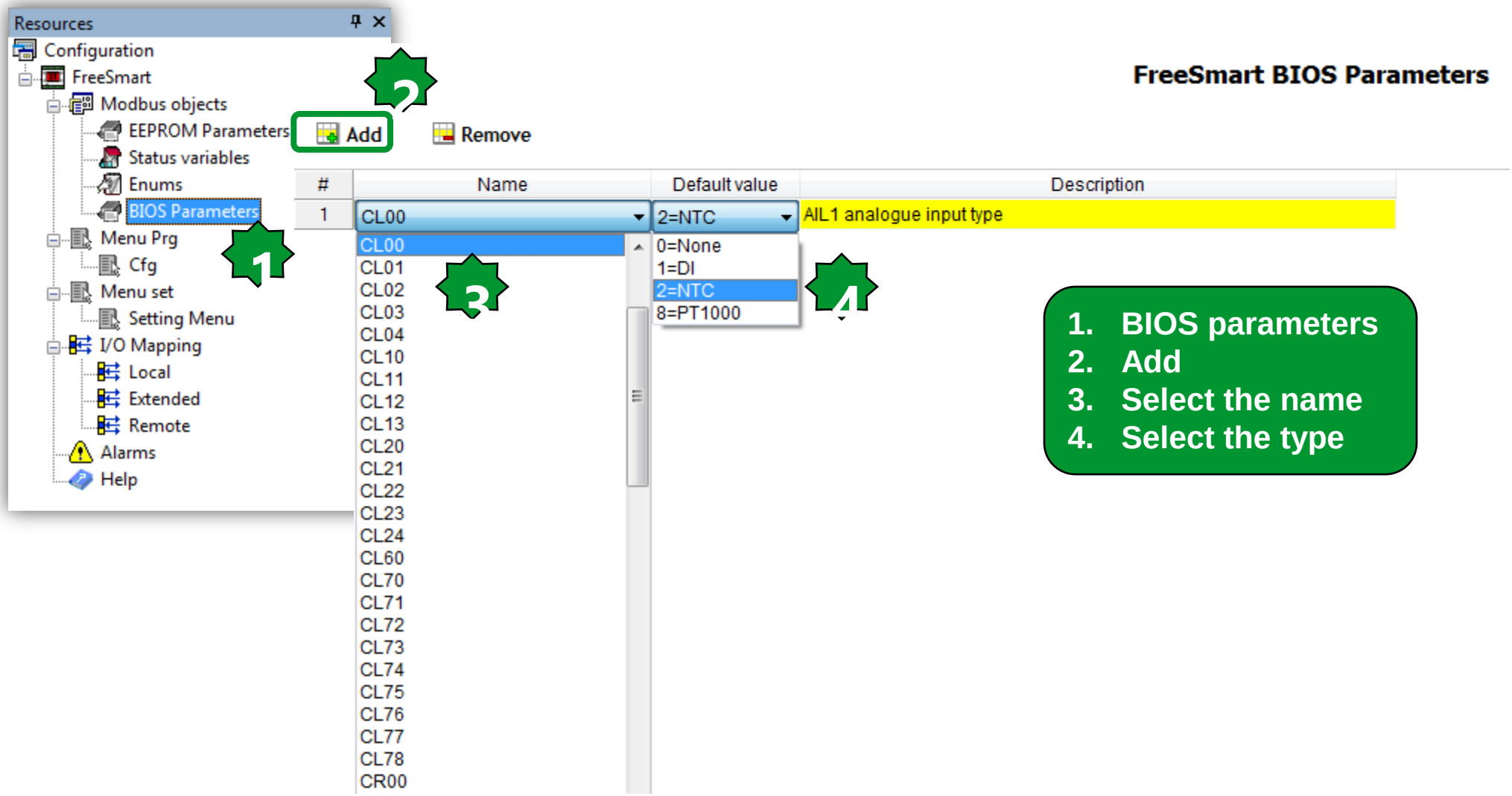
Check FS Device parameters description...



Local

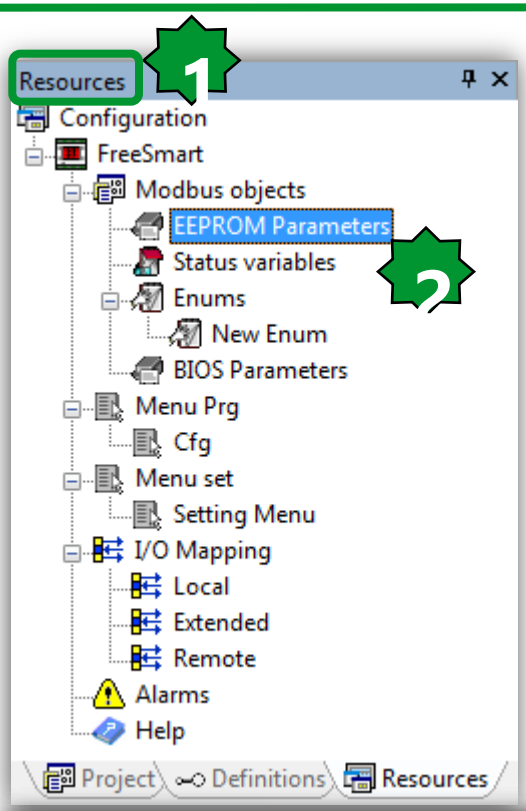
Address	Name	Value	Um	Default	Min	Max	Description
53304	CL00	2=NTC	num	2=NTC	0	8	AIL1 analogue input type
53305	CL01	2=NTC	num	2=NTC	0	8	AIL2 analogue input type
53306	CL02	2=NTC	num	2=NTC	0	7	AIL3 analogue input type
53307	CL03	2=NTC	num	2=NTC	0	7	AIL4 analogue input type
53308	CL04	2=NTC	num	2=NTC	0	8	AIL5 analogue input type
15649	CL10	500	°C/Bar	500	-9999	9999	AIL3 analogue input full scale value
15655	CL11	0	°C/Bar	0	-9999	9999	AIL3 analogue input start of scale value
15650	CL12	500	°C/Bar	500	-9999	9999	AIL4 analogue input full scale value
15656	CL13	0	°C/Bar	0	-9999	9999	AIL4 analogue input start of scale value
53334	CL20	0	°C	0	-120	120	AIL1 analogue input differential
53335	CL21	0	°C	0	-120	120	AIL2 analogue input differential
53336	CL22	0	°C/Bar	0	-120	120	AIL3 analogue input differential
53337	CL23	0	°C/Bar	0	-120	120	AIL4 analogue input differential
53338	CL24	0	°C	0	-120	120	AIL5 analogue input differential
53344	CL60	0=0-20mA	num	0=0-20mA	0	2	AOL5 analogue output type
53346	CL70	0=Disable	num	0=Disable	0	2	Enable TCL1 analogue output
53347	CL71	0=Disable	num	0=Disable	0	2	Enable AOL1 analogue output
53348	CL72	1=Enable	num	1=Enable	0	2	Enable AOL2 analogue output
53349	CL73	27	Deg	27	0	90	Phase shift TCL1 analogue output
53350	CL74	27	Deg	27	0	90	Phase shift AOL1 analogue output
53351	CL75	27	Deg	27	0	90	Phase shift AOL2 analogue output
53352	CL76	10	69 µsec	10	5	40	TCL1 analogue output pulse length
53353	CL77	10	69 µsec	10	5	40	AOL1 analogue output pulse length
53354	CL78	10	69 µsec	10	5	40	AOL2 analogue output pulse length

...and define the Application BIOS Default



1. BIOS parameters
2. Add
3. Select the name
4. Select the type

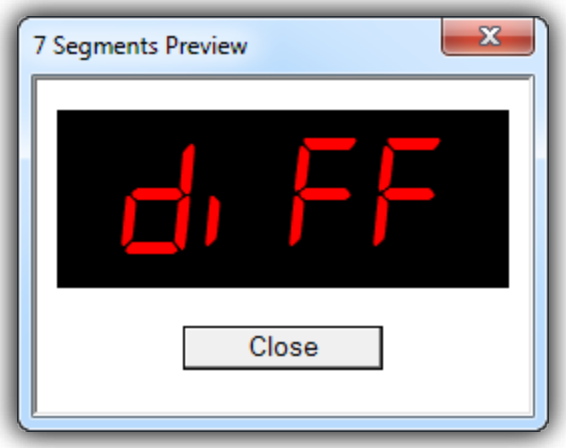
EEPROM parameters



1. Resources
2. EEPROM parameters
3. Add
4. Define required „retain“ data
5. 1% resolution
   Format XXX.Y

Note:Default values are written to the target only by Free Studio Device

App. Type : IEC variable type
Device Type : FS Device / Display type
App. Type = Scale x Device Type + Offset



FreeSmart EEPROM Parameters

5

Add Remove Recalc

#	Address	Name	Display label	Device type	Application type	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description	Note
1	16384	Setpoint	SetP	Signed 16-bit	INT	180	150	300	1	0	°C	XXX.Y	Always visible		
2	16385	Differentiation	Diff	Signed 16-bit	INT	20	5	50	1	0	°C	XXX.Y	Always visible		

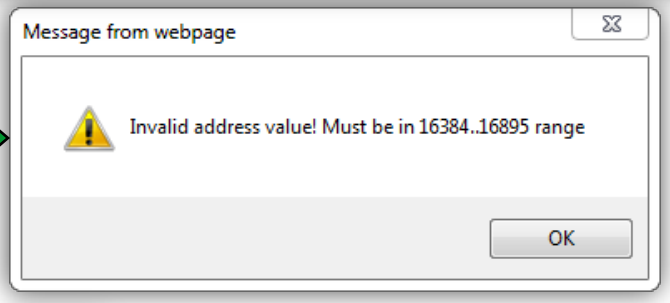
EEPROM Properties

FreeSmart EEPROM Parameters

Add | Remove | Recalc

#	Address	Name	Display label	Device type	Application type	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Description
1	16384	Setpoint	SetP	Signed 16-bit	INT	180	150	300	1	0	°C	XXX.Y	Always visible	
2	16385	Differentiation	Diff	Signed 16-bit	INT	20	5	50	1	0	°C	XXX.Y	Always visible	

1



2

°C
bar
%R.H.

3

XXX.Y
XX.YY
%04x
HH:MM

4

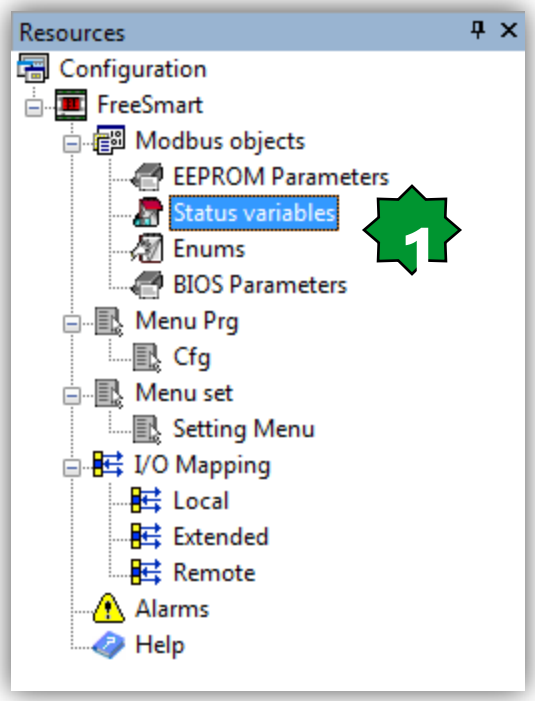
Never visible
Level 1
Level 2
Always visible

EEPROM Parameters:

App. Type : IEC variable type
Device Type : FS Device / Display type
App. Type = Scale x Device Type + Offset

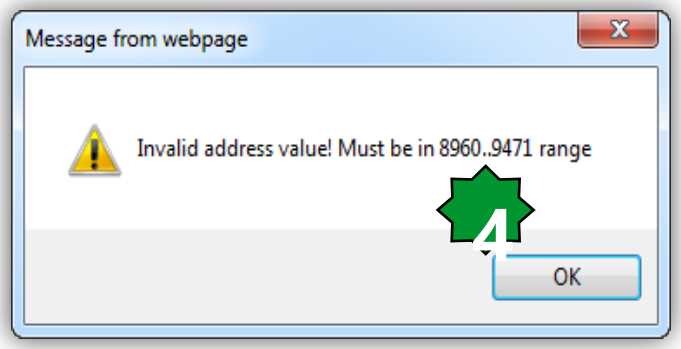
Note.
Dynamic setpoint by the other parameters Min/Maxis possible.

Status Variables

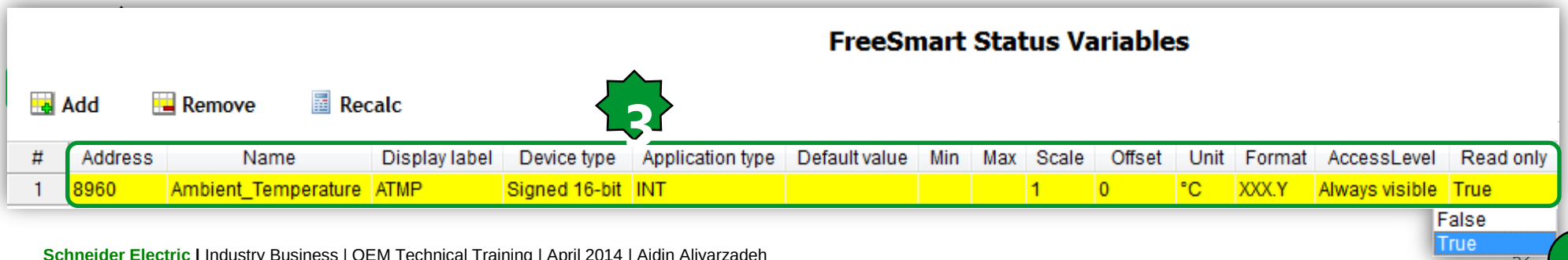


1. Resources ►Status variable
2. Add new record
3. Define required RAM data
4. Valid range

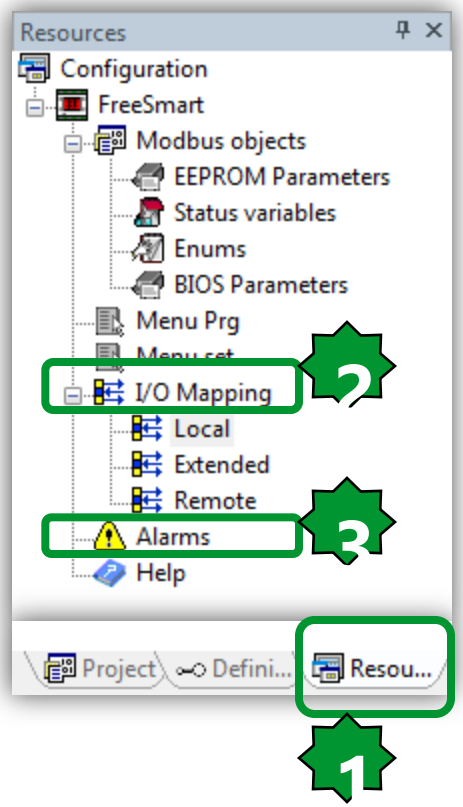
Note: Same properties as EEPROM + read only

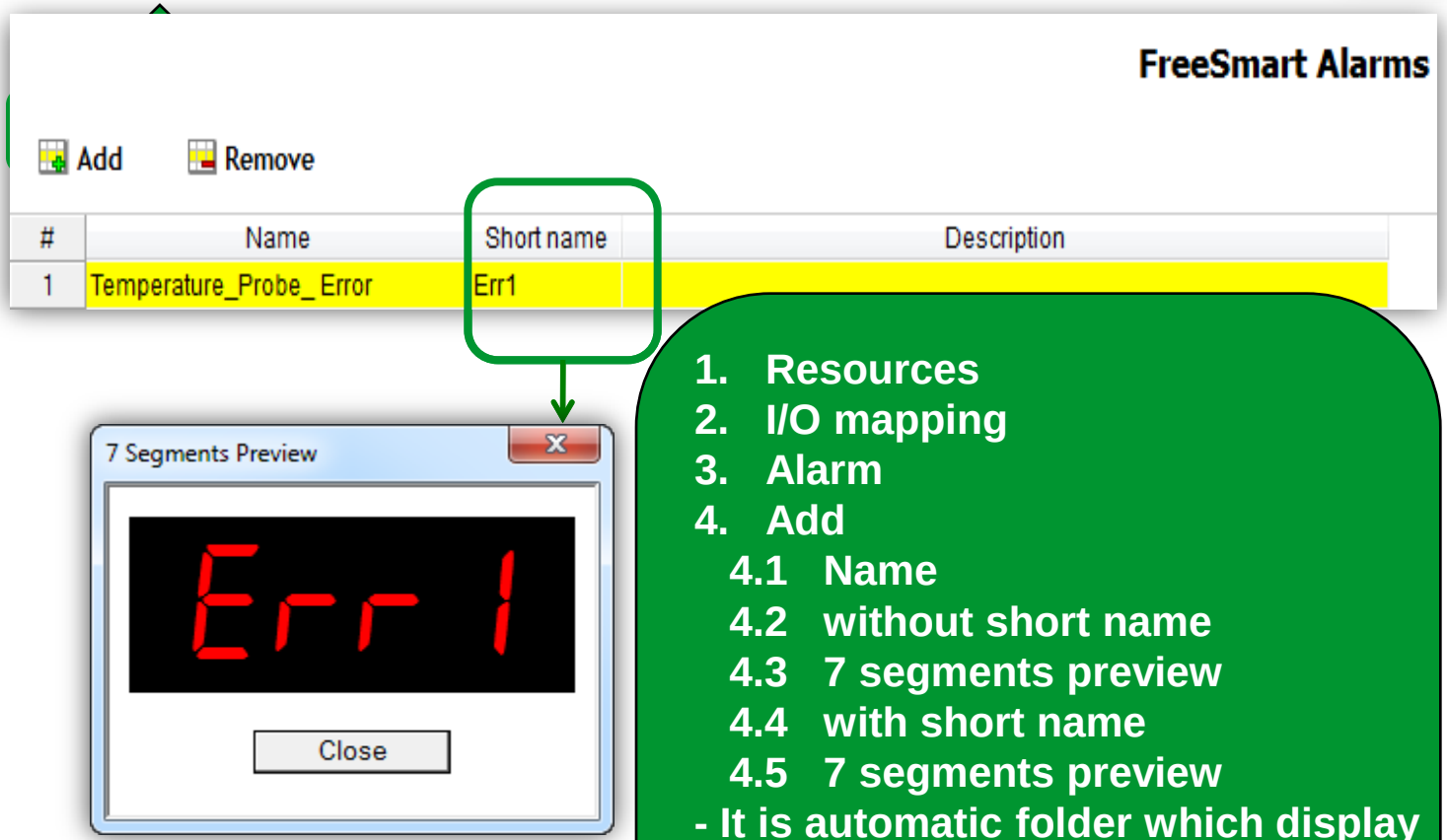


App. Type : IEC variable type
Device Type : FS Device / Display type
App. Type = Scale x Device Type + Offset



Alarms

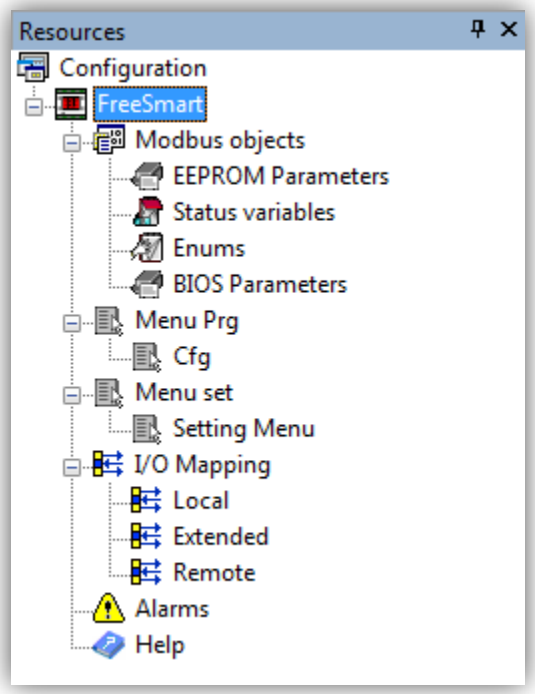


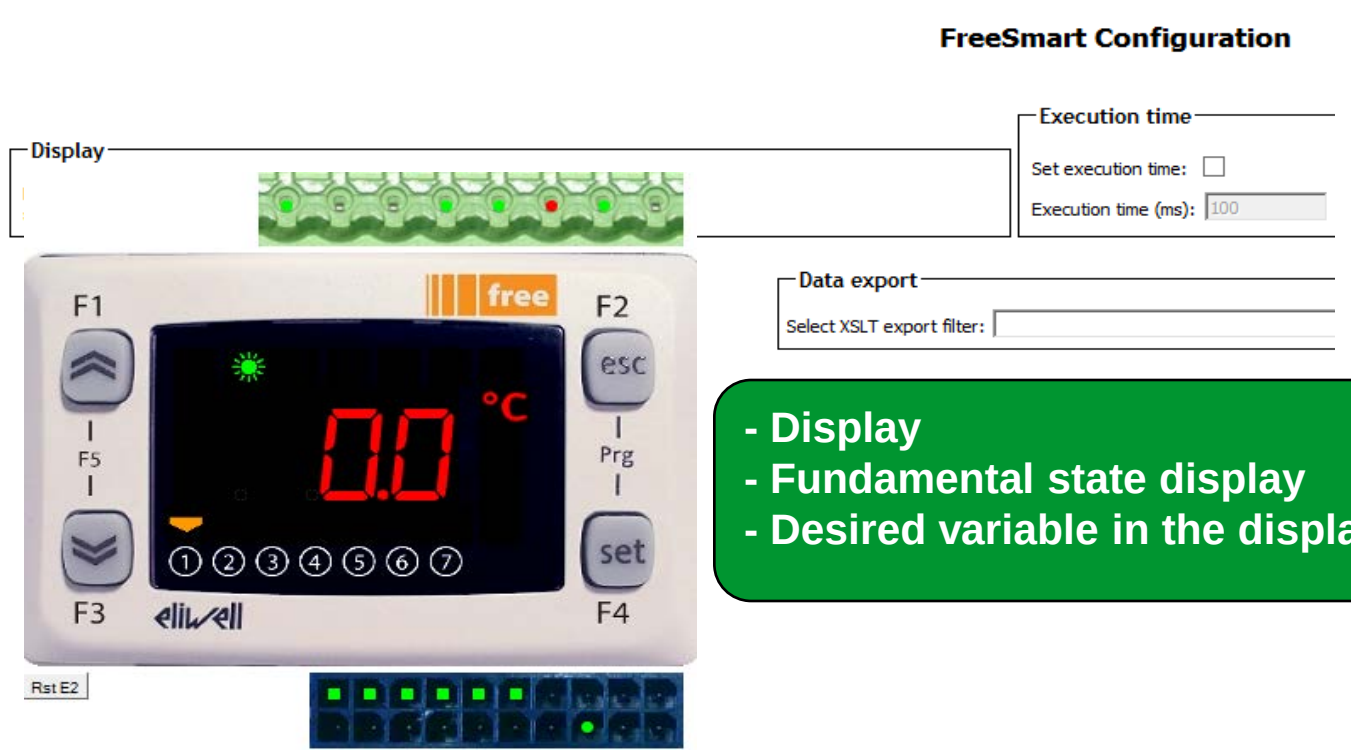


1. Resources
2. I/O mapping
3. Alarm
4. Add
   - 4.1 Name
   - 4.2 without short name
   - 4.3 7 segments preview
   - 4.4 with short name
   - 4.5 7 segments preview

- It is automatic folder which display the Alarms, if value is 0, no display In AL folder if alarm value is 1, displays the short name. blinks if value> than 1.
- In case of any alarms, the red triangle icon in the display is on.

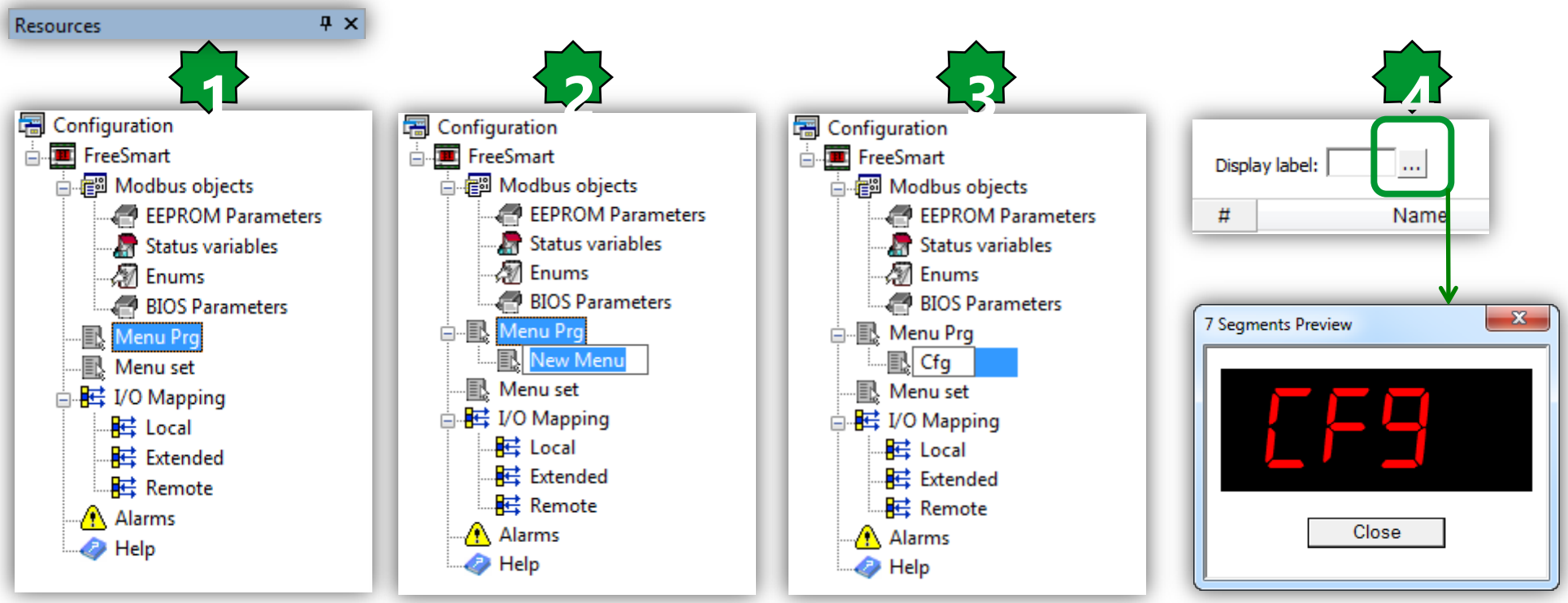
Fundamental state display configuration





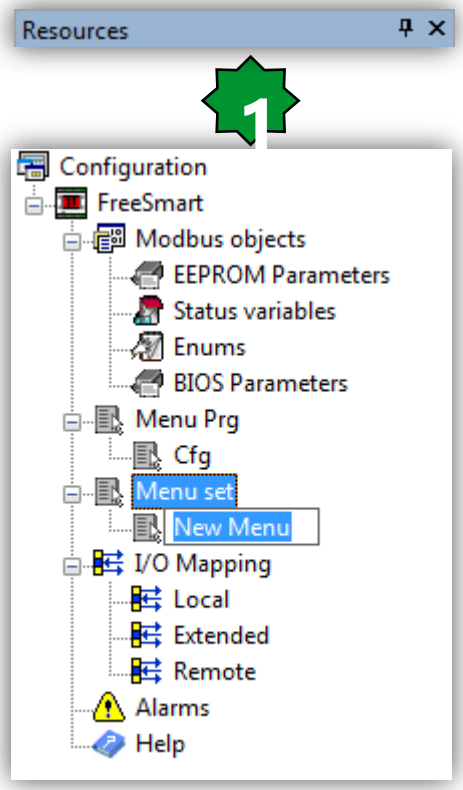
- Display
- Fundamental state display
- Desired variable in the display

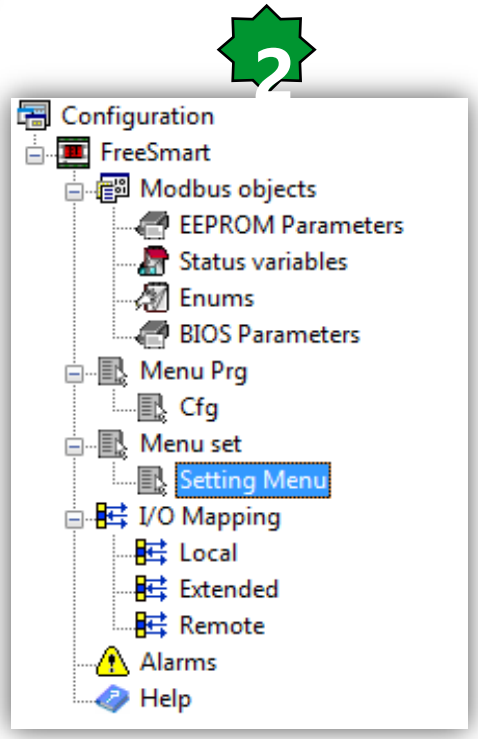
Menu Program – Add Folder

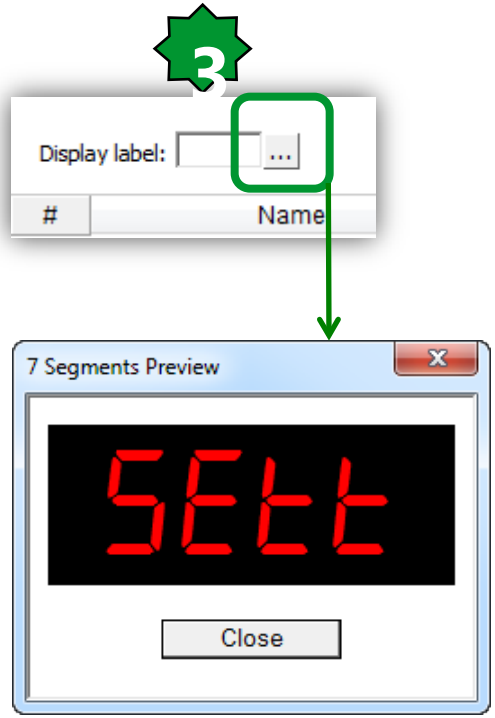


1. **Menu Prg.**
2. **Add Menu**
3. **New Menu, name it (Cfg)**
4. **7 segment preview**

Menu Set – Add Folder

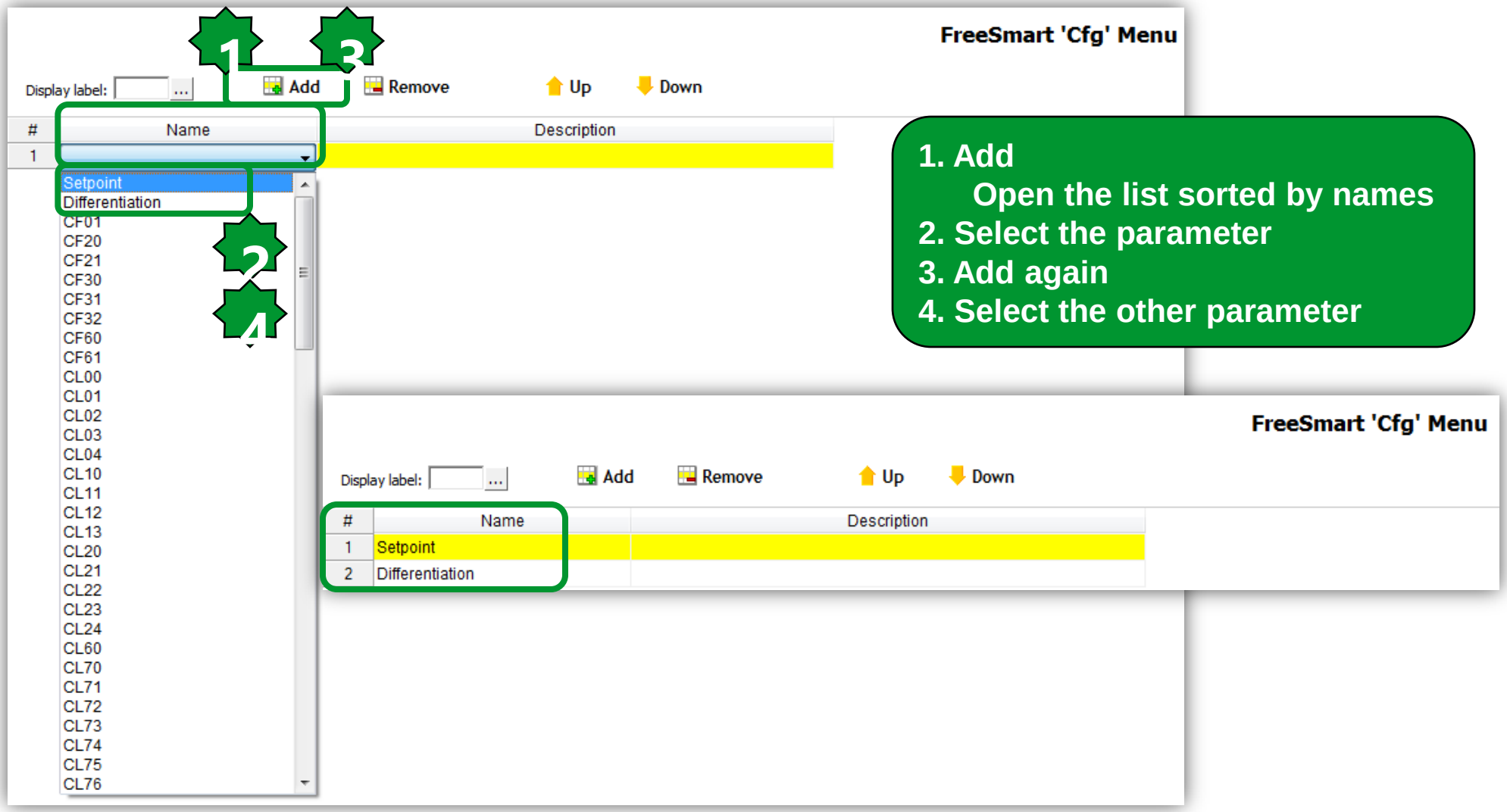




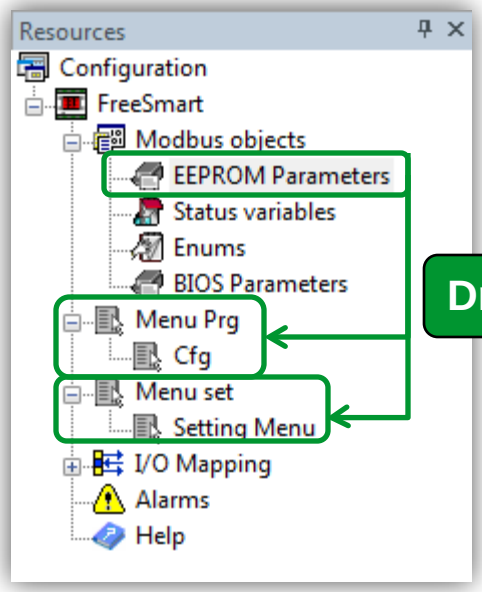


1. **Menu Set, Right Click Add Menu**
2. **New Menu, name it (Setting Menu)**
3. **7 segments preview**

Add/Remove elements to folder



Add/Remove elements by drag & drop



Drag & Drop

FreeSmart 'Cfg' Menu

Display label: ... Add Remove Up Down

#	Name	Description
1	Setpoint	
2	Differentiation	

FreeSmart 'Setting Menu' Menu

Display label: ... Add Remove Up Down

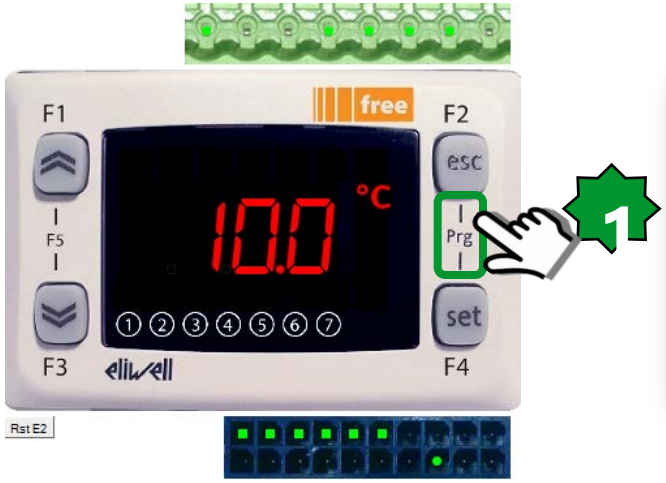
#	Name	Description
1	Setpoint	
2	Differentiation	
3	Ambient_Temperator	

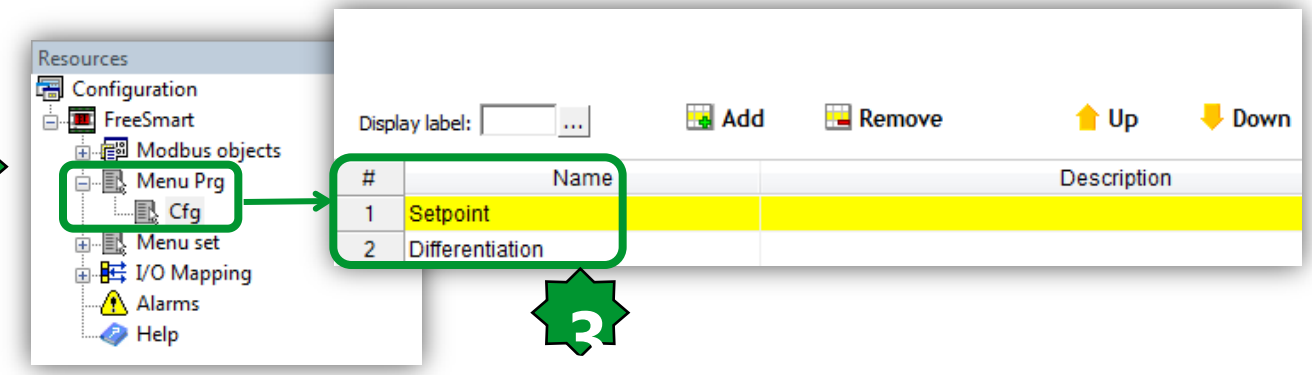
FreeSmart EEPROM Parameters

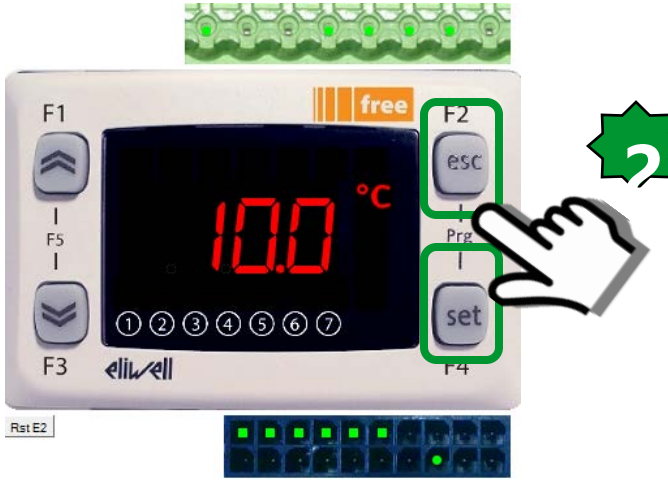
Add Remove Recalc

#	Address	Name	Display label	Device type	Application type	Default value	Min	Max	Scale	Offset	Format	AccessLevel
1	16384	Setpoint	SetP	Signed 16-bit	INT	180	150	300	1	0	XXX.Y	Always visible
2	16385	Differentiation	Diff	Signed 16-bit	INT	20	5	50	1	0	XXX.Y	Always visible

Menu Program – How to Access



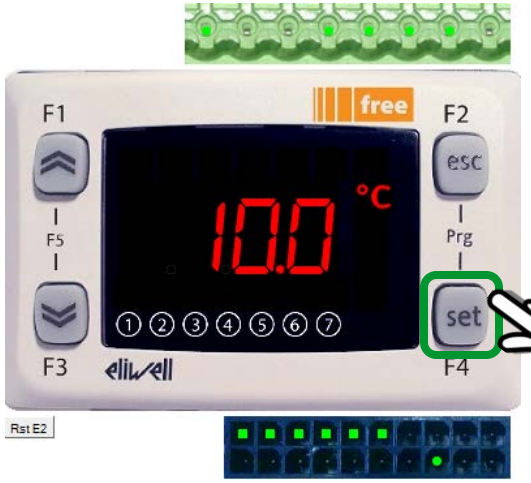


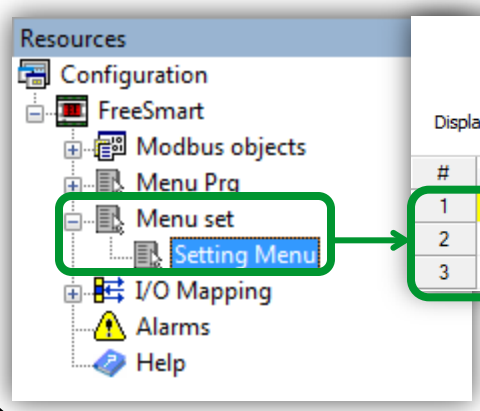


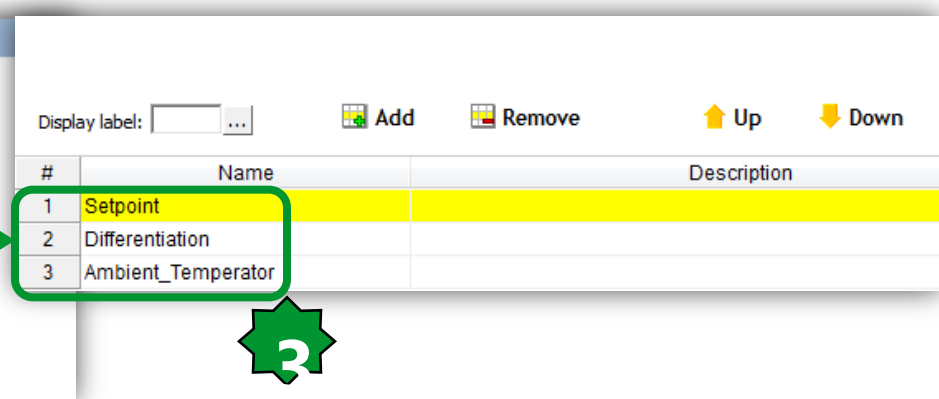
To form & define the program button function:

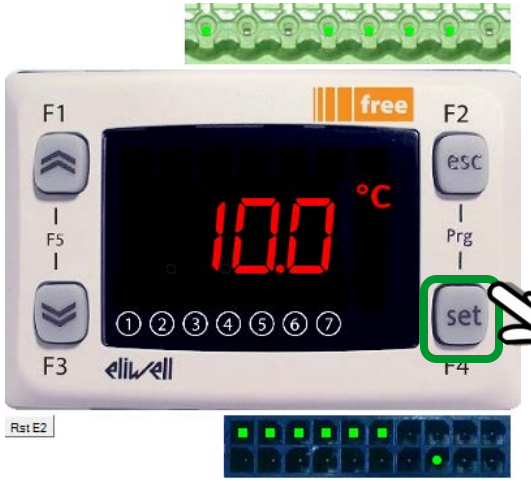
1. **Menu Prg, access by pressing Prg button in simulation**
2. **Menu Prg, access by pressing esc.& set buttons simultaniously on the product**
3. **Scroll up or down into the defined variables ▲▼**

Menu Set – How to Access



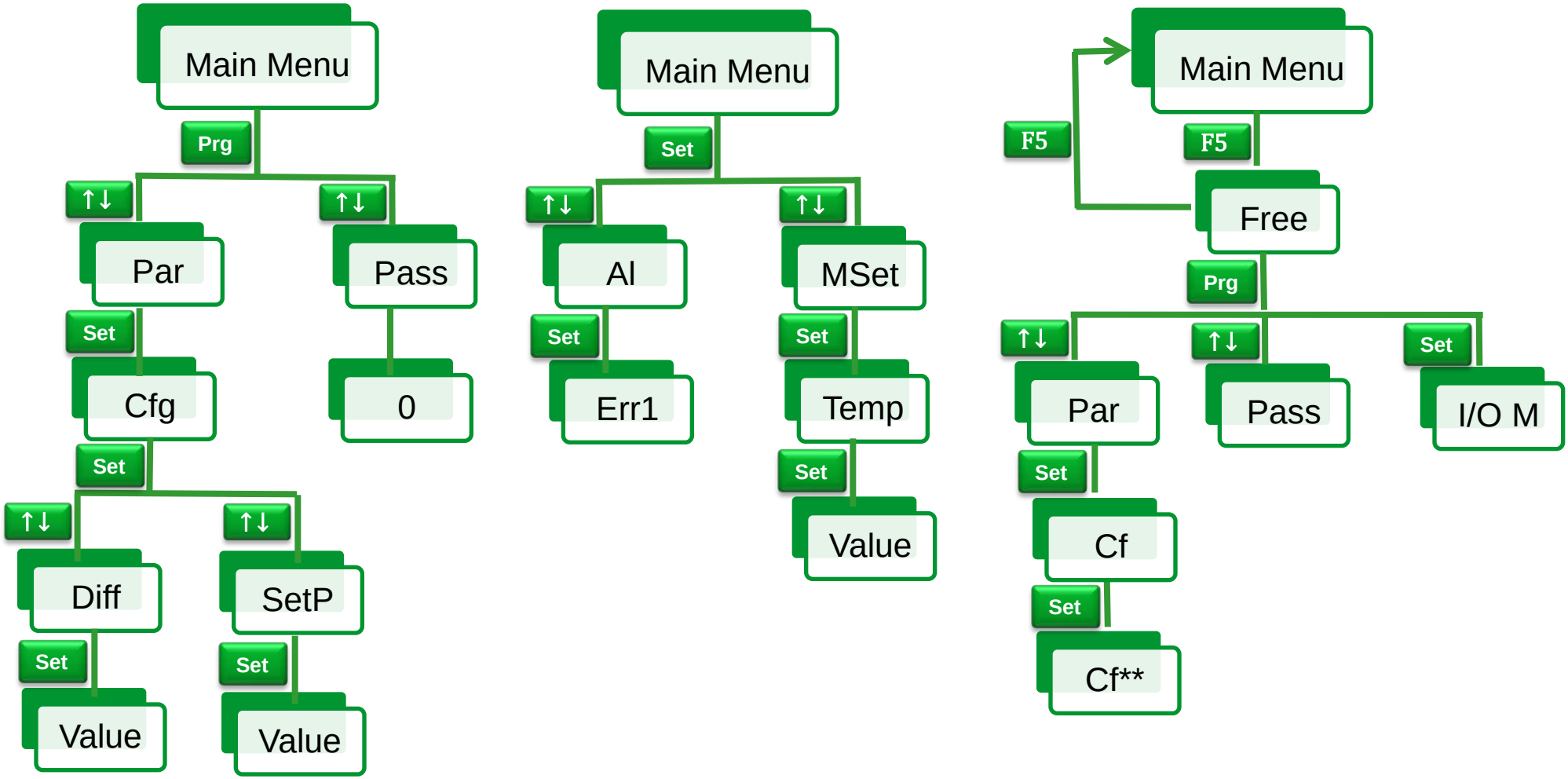






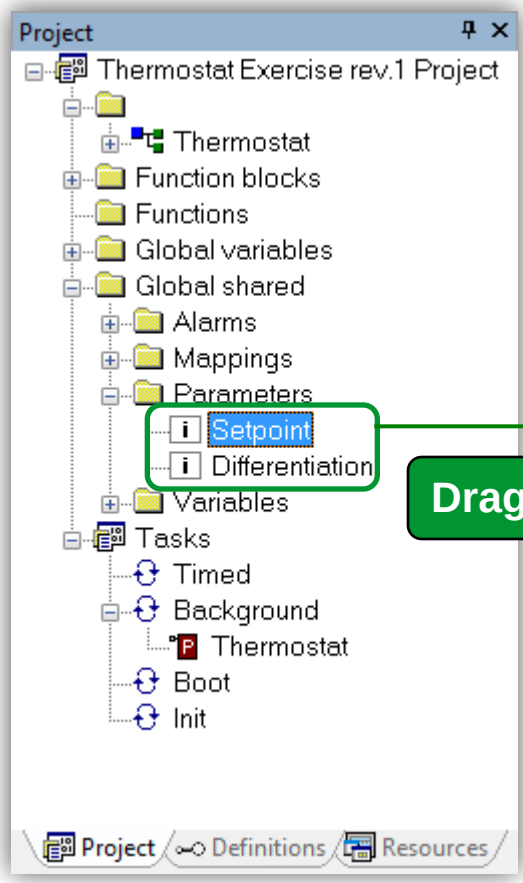
To form & define the set button function:
1. Access to the set menu by pressing set in simulation
2. Access to the set menu by pressing set on the product
3. Scroll up or down into the defined variables ▼▲

Menu architecture

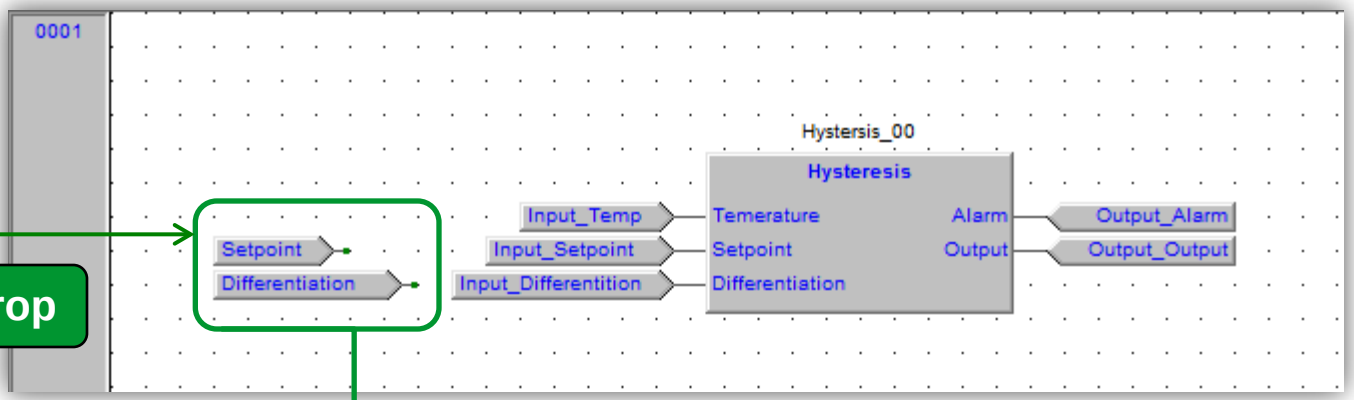


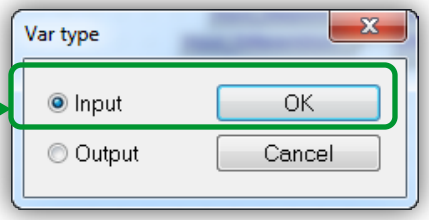
Project / Global Shared

All the resources are available in the project tab under Global Shared folder (after saving or compiling the project)

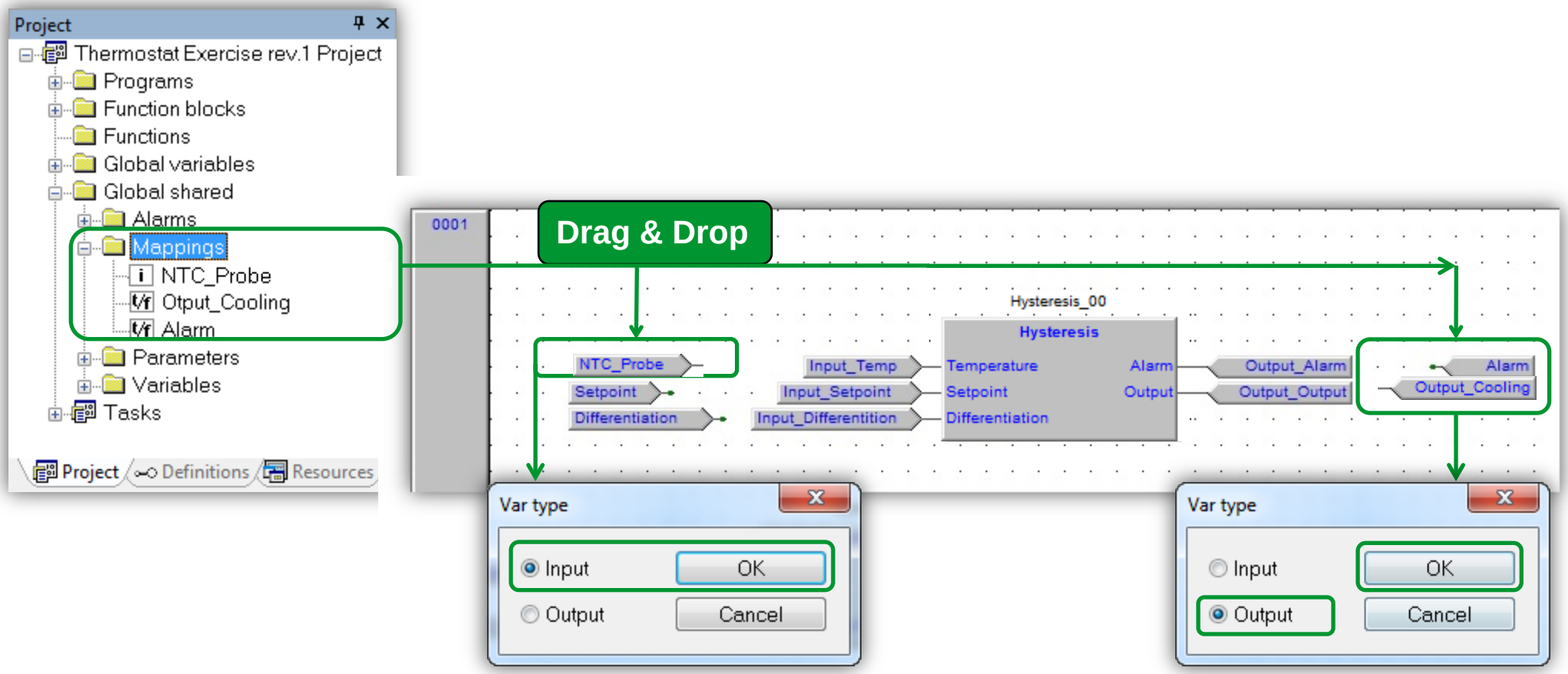


Drag & Drop

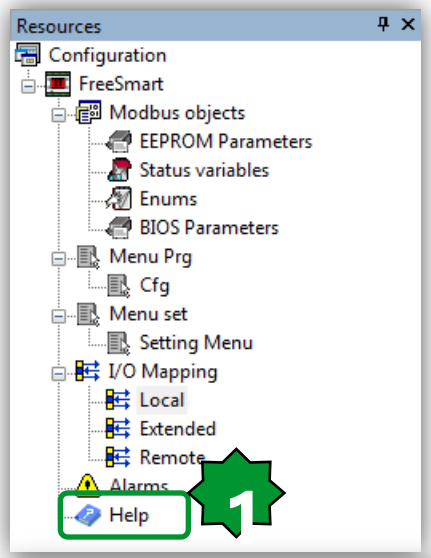


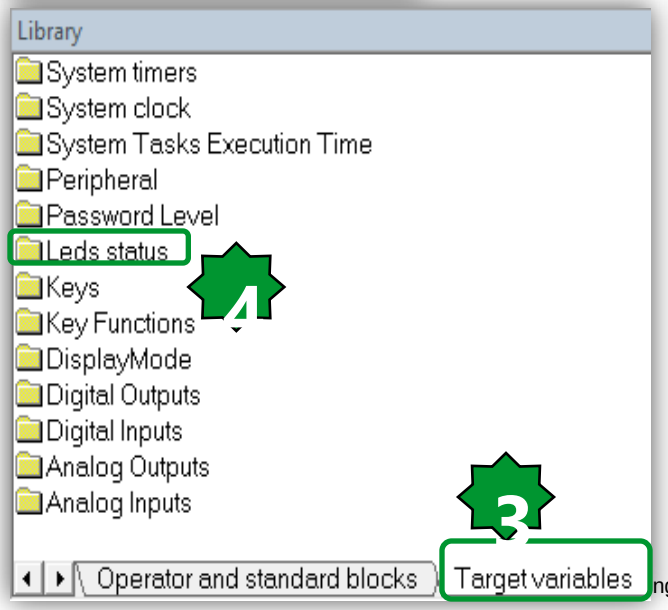


Using physical I/O



System LED setting





2

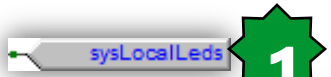
LED reference for the developer

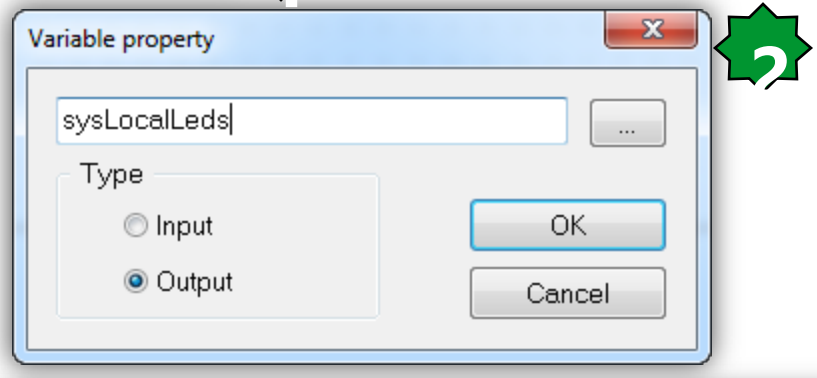
The IEC developer can turn on (either steady or blinking) and off the whole range of local display LEDs, by properly setting the array SYSLED.

LED number	Symbol or icon	Description	Off	On (steady)	On (blinking)
0	:	Colon	SYSLED[0]=0	SYSLED[0]=1	SYSLED[0]=2
1	%R.H.	%RH	SYSLED[1]=0	SYSLED[1]=1	SYSLED[1]=2
2		Defrost	SYSLED[2]=0	SYSLED[2]=1	SYSLED[2]=2
3	Bar	Bar	SYSLED[3]=0	SYSLED[3]=1	SYSLED[3]=2
4		Stand-by	SYSLED[4]=0	SYSLED[4]=1	SYSLED[4]=2
5	°C	°C	SYSLED[5]=0	SYSLED[5]=1	SYSLED[5]=2
6		Cooling	SYSLED[6]=0	SYSLED[6]=1	SYSLED[6]=2
7		Clock (RTC)	SYSLED[7]=0	SYSLED[7]=1	SYSLED[7]=2
8		Heating	SYSLED[8]=0	SYSLED[8]=1	SYSLED[8]=2
9		User-defined 1	SYSLED[9]=0	SYSLED[9]=1	SYSLED[9]=2
10		User-defined 2	SYSLED[10]=0	SYSLED[10]=1	SYSLED[10]=2
11		User-defined 3	SYSLED[11]=0	SYSLED[11]=1	SYSLED[11]=2
12		User-defined 4	SYSLED[12]=0	SYSLED[12]=1	SYSLED[12]=2
13		User-defined 5	SYSLED[13]=0	SYSLED[13]=1	SYSLED[13]=2
14		User-defined 6	SYSLED[14]=0	SYSLED[14]=1	SYSLED[14]=2
15		User-defined 7	SYSLED[15]=0	SYSLED[15]=1	SYSLED[15]=2
16		Alarm	SYSLED[16]=0	SYSLED[16]=1	SYSLED[16]=2
17	ABC	Menu	SYSLED[17]=0	SYSLED[17]=1	SYSLED[17]=2
18		Economy	SYSLED[18]=0	SYSLED[18]=1	SYSLED[18]=2

Some of the LEDs - for example, LED number 0, 1, 3, 5, and 7 (in green) - cannot be used by the IEC developer when BIOS menu is active.

System local LED assigning

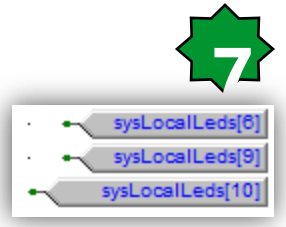


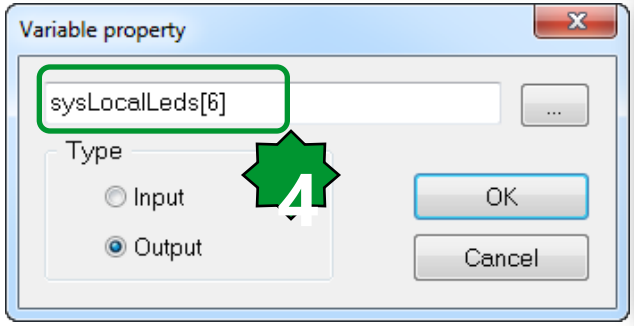


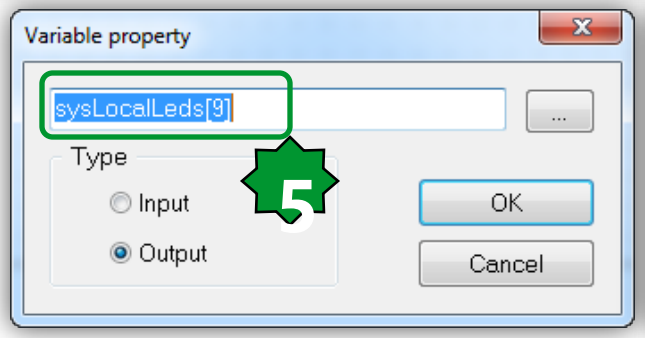
1. 2*click
2. Variable property
3. Dedicate required LED
4. Type [6] ► Output ► OK
5. Type [9] ► Output ► OK
6. Type[10] ► Output ► OK
7. Dedicated LED‘s

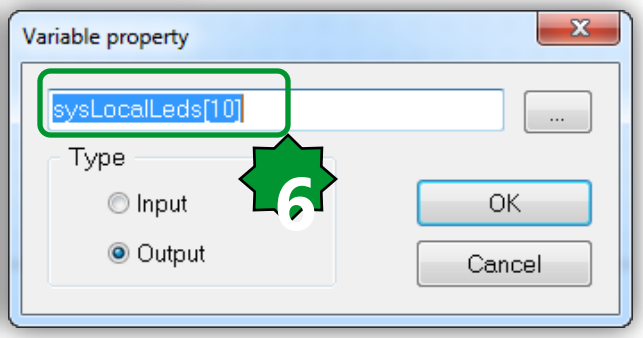
3

LED number	Symbol or icon	Description	Off	On (steady)	On (blinking)
6		Cooling	SYSLED[6]=0	SYSLED[6]=1	SYSLED[6]=2
9		User-defined 1	SYSLED[9]=0	SYSLED[9]=1	SYSLED[9]=2
10		User-defined 2	SYSLED[10]=0	SYSLED[10]=1	SYSLED[10]=2





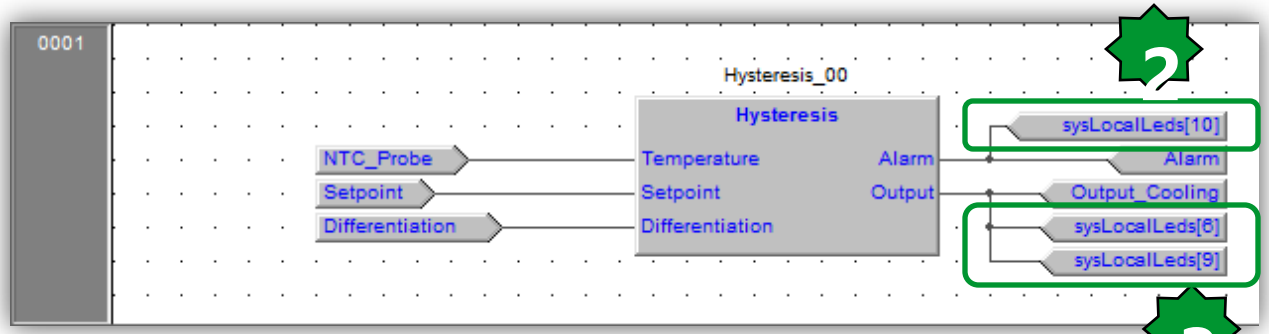




Connecting dedicated LED‘s to the FBD

1

2



3

Digital output map

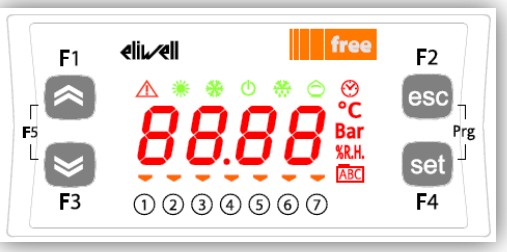
5

Digital Input map

4

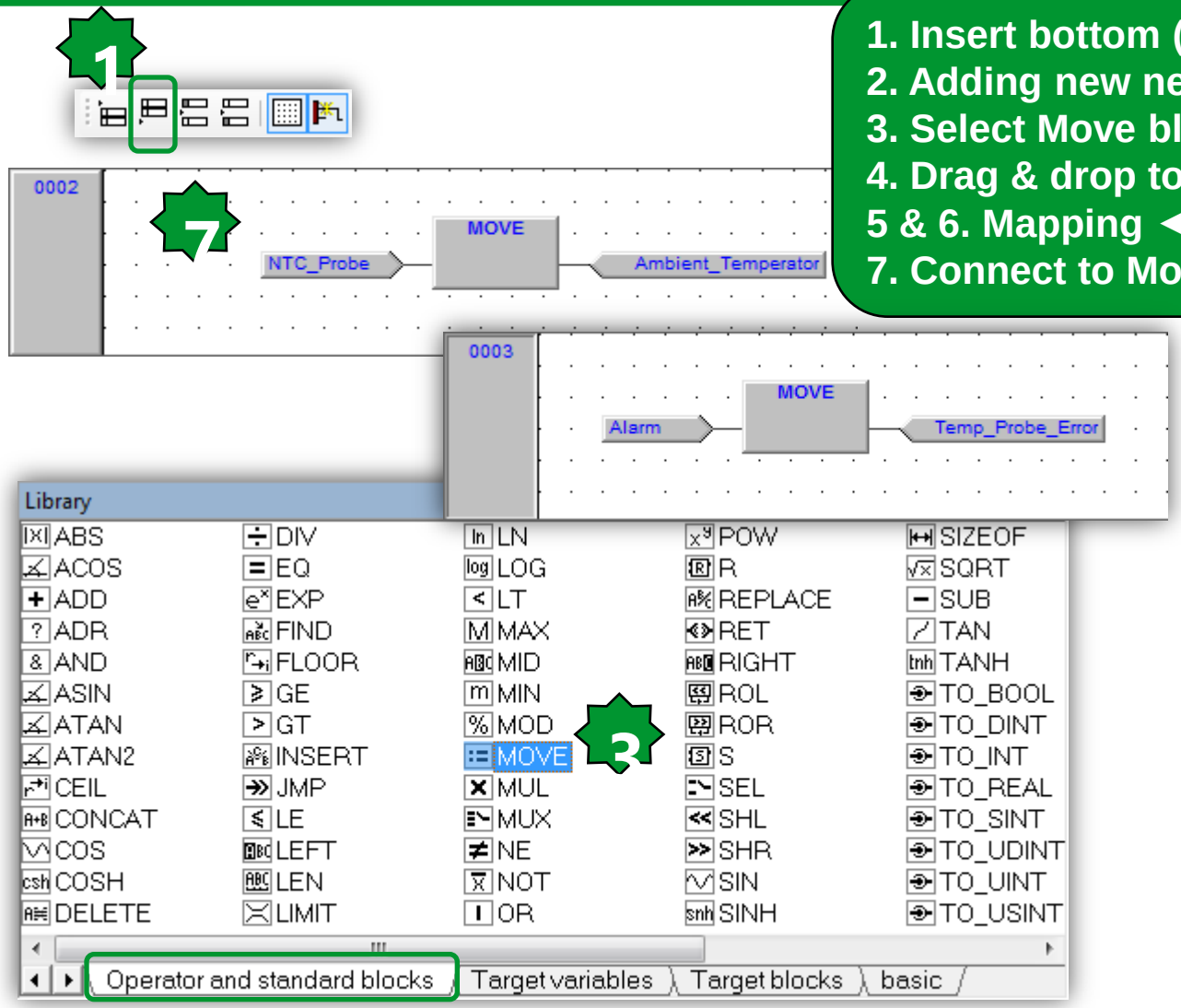
4

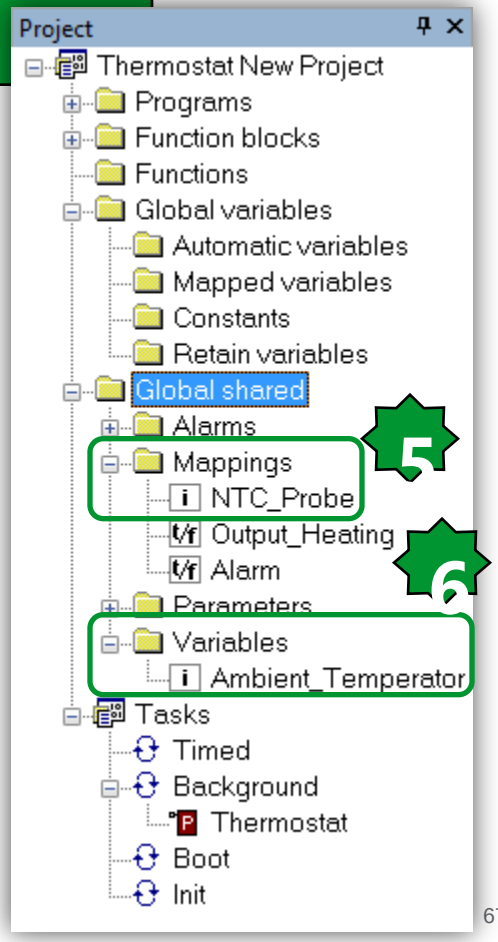
LED number	Symbol or icon	Description	Off	On (steady)	On (blinking)
6	❄	Cooling	SYSLED[6]=0	SYSLED[6]=1	SYSLED[6]=2
9	▬	User-defined 1	SYSLED[9]=0	SYSLED[9]=1	SYSLED[9]=2
10	▬	User-defined 2	SYSLED[10]=0	SYSLED[10]=1	SYSLED[10]=2



Valorize Fundamental State Display

1. Insert bottom (new network)
2. Adding new network
3. Select Move block
4. Drag & drop to new network
5 & 6. Mapping ◄► Variables
7. Connect to Move block

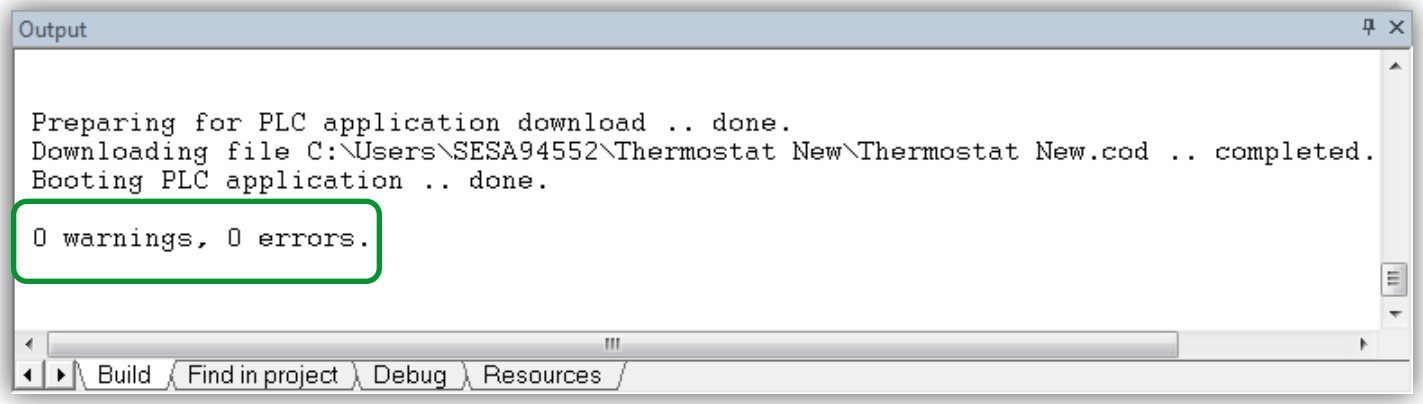


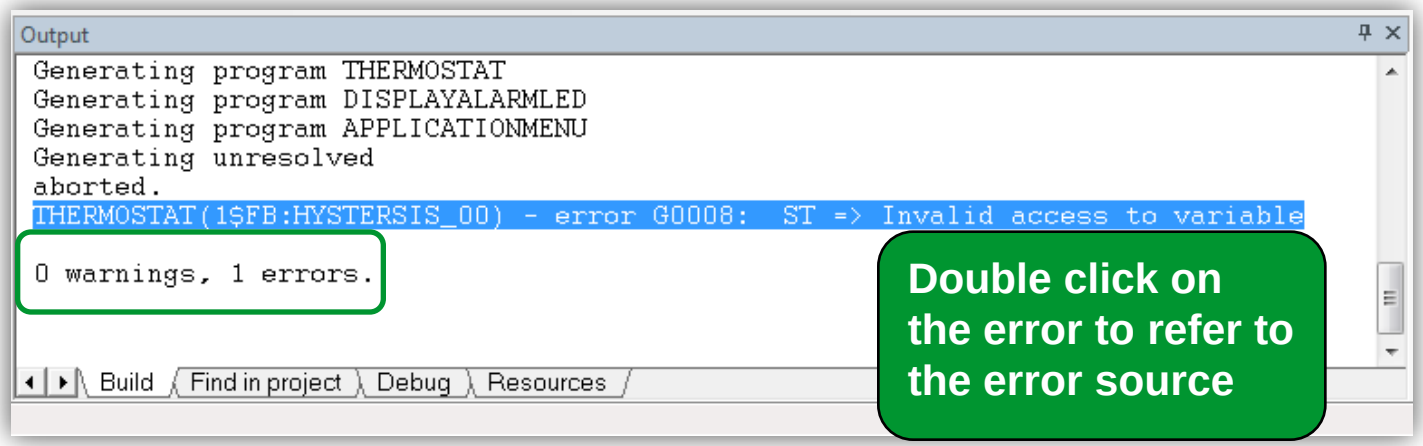


*** Move command is mandetory to connect two variables.**

Compile/Build

Compile





Double click on the error to refer to the error source

Chapter 5

Simulation and Debugging – Part 2

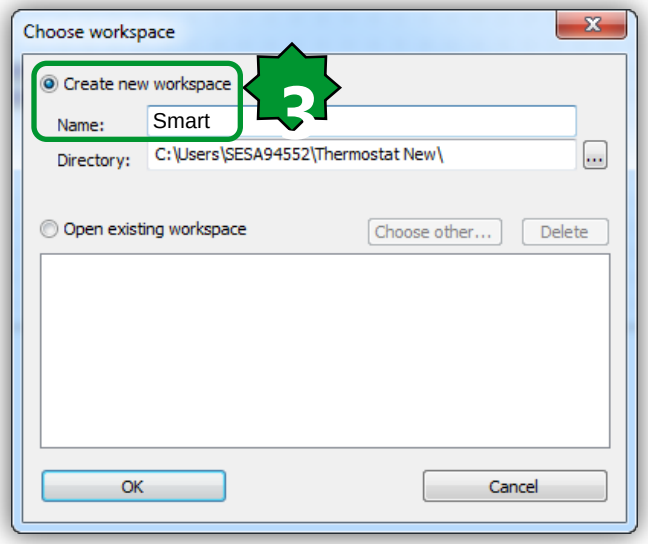
Goal:

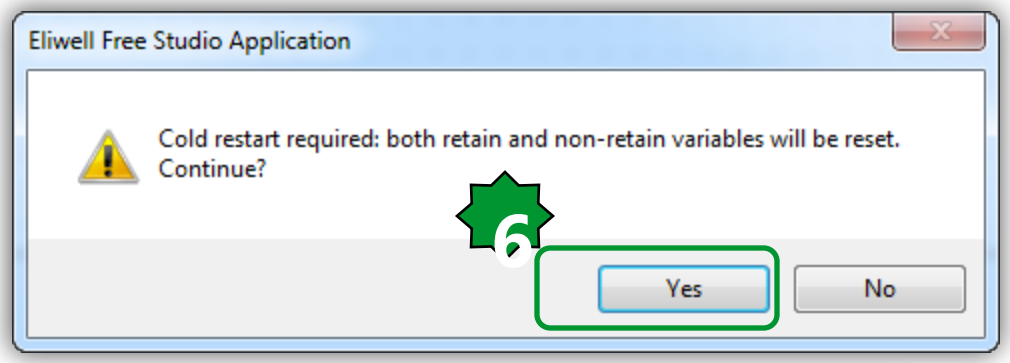
On-Line simulation mode, testing of:

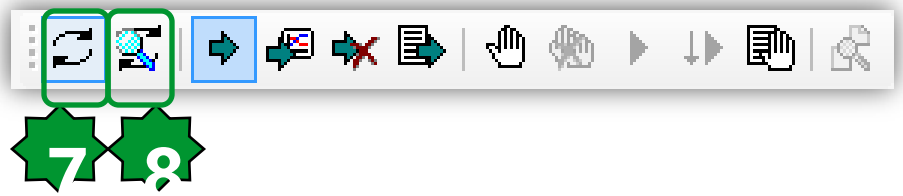
- Physical I/O
- 7 segment display

Off line simulation mode

Debug ► Simulation mode ►

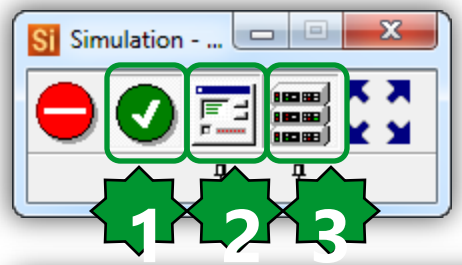




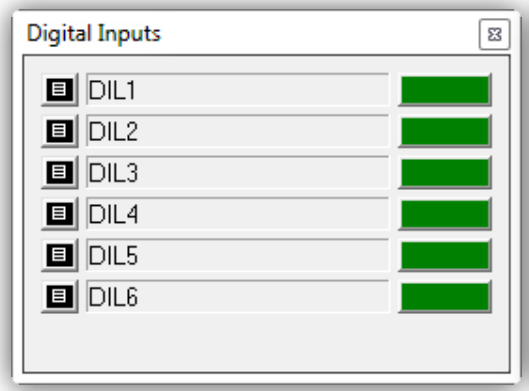


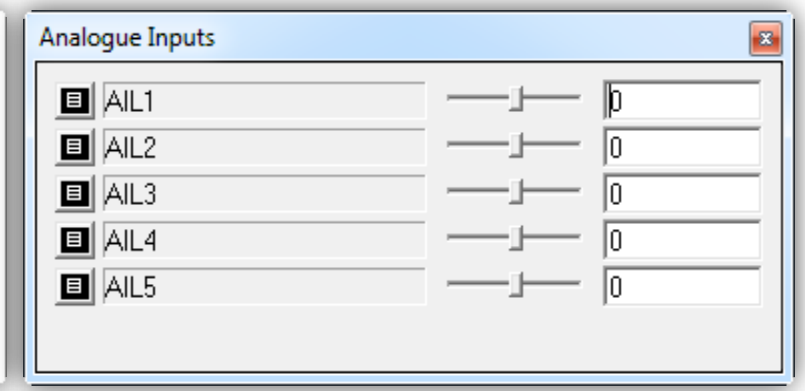
1. Debug
2. Simulation mode
3. Name it ► OK
4. If differs or no code
5. Download code, OK
6. Reset Variables
7. Debug mode (optional)
8. Live debug (Continuous) mode (optional)

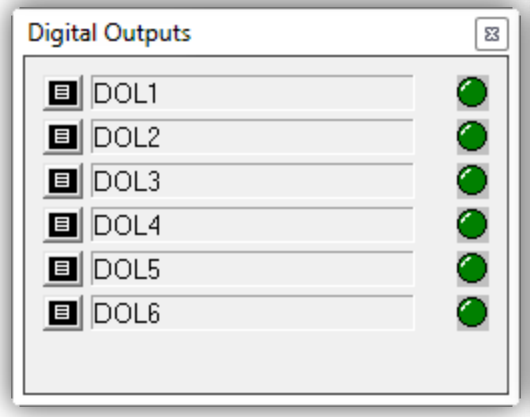
Simulation tools

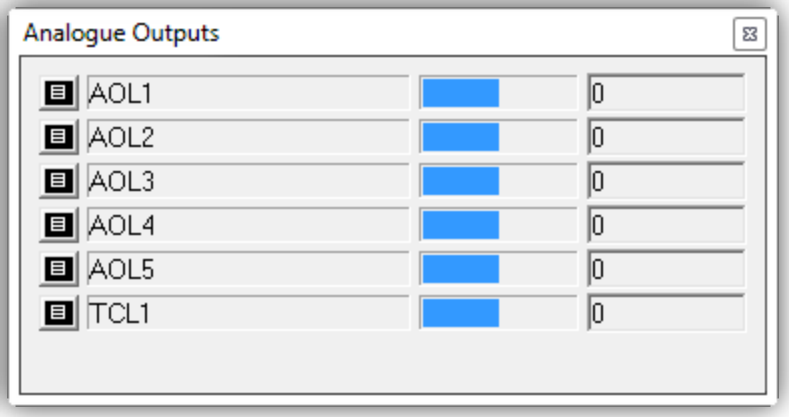


1. Active code execution
2. Show I/O panels
3. Show HMI window

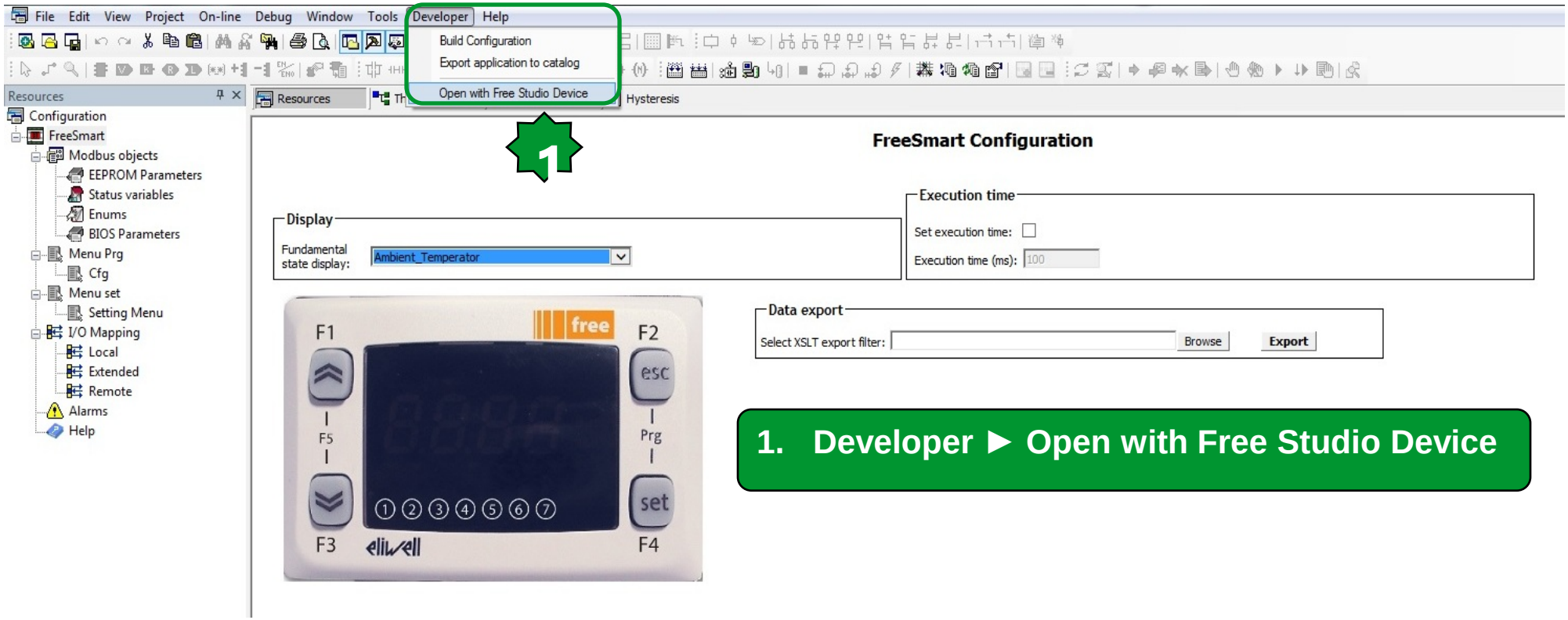






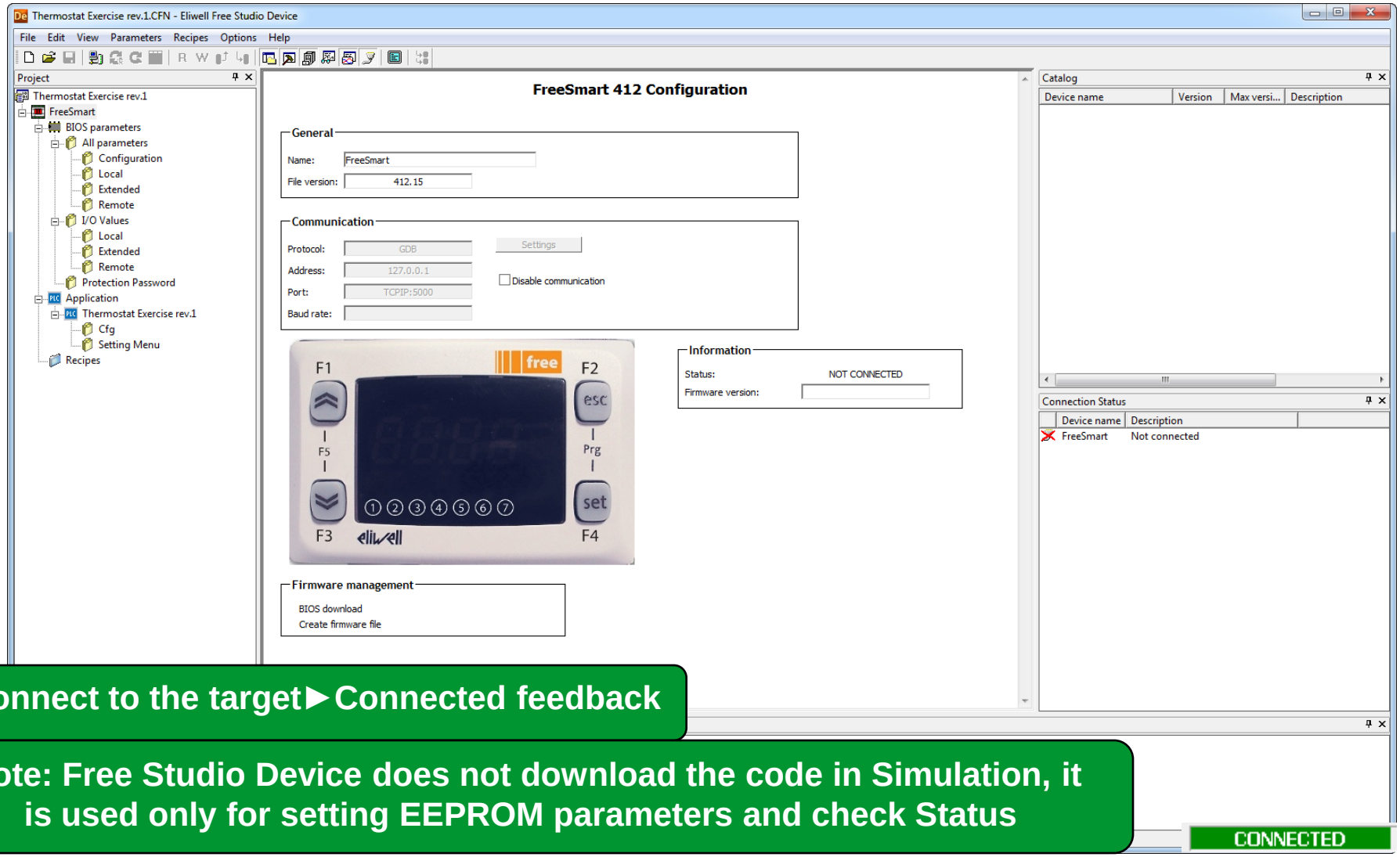


Open Free Studio Device from Application



1. Developer ► Open with Free Studio Device

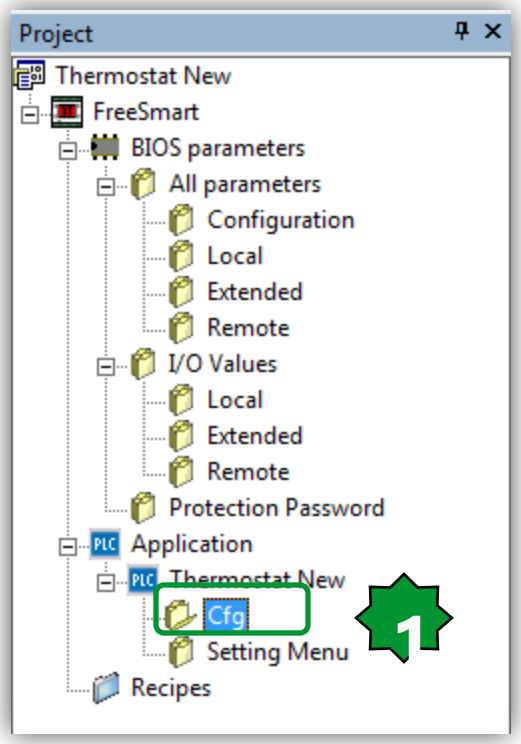
Free Studio Device (Simulation Target)

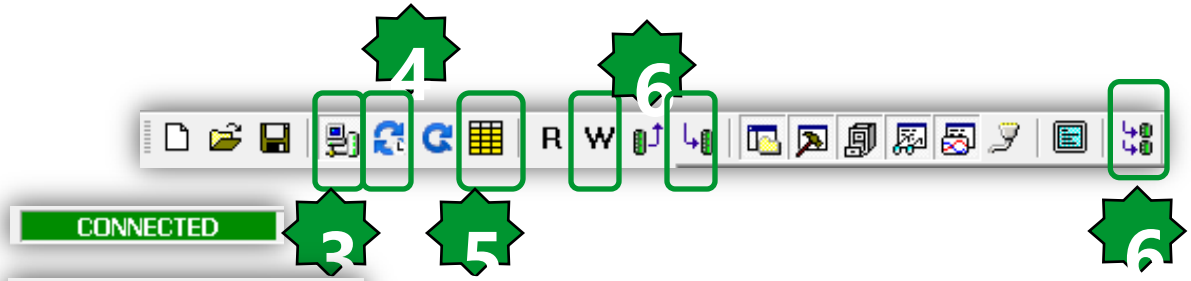


Connect to the target►Connected feedback

Note: Free Studio Device does not download the code in Simulation, it is used only for setting EEPROM parameters and check Status

Read / Write Values





Connection Status

	Device name	Description
	FreeSmart	Connected

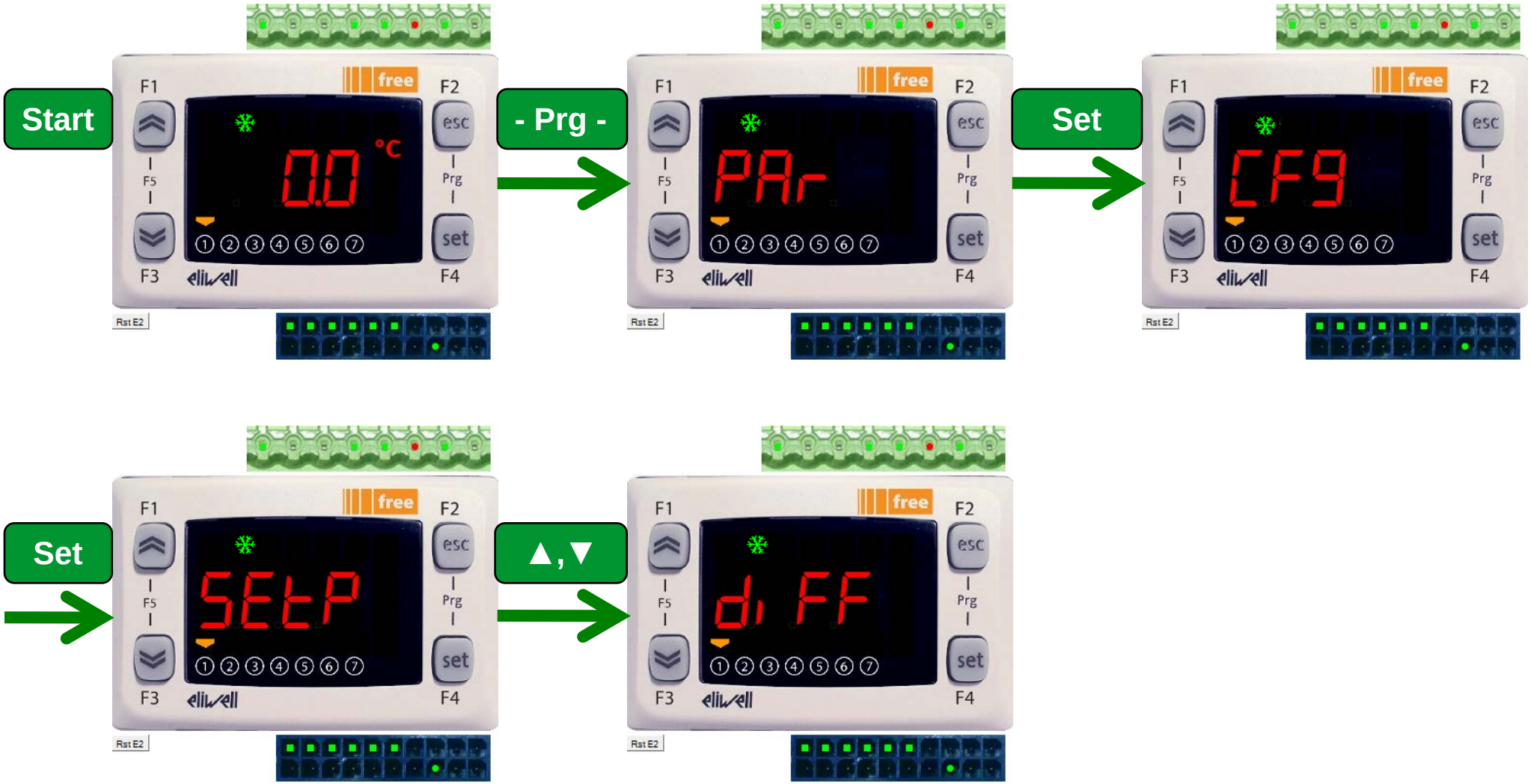
1. Connect to the target►Connected feedback
2. Application►Project►Cfg
3. Cfg menu►non aligned value with target (Red)
4. Auto refresh mode (optional)
5. Select all parameters (or selective)
6. Write parameter (all or selective)+ download all
7. Aligned values with the target (black)

7

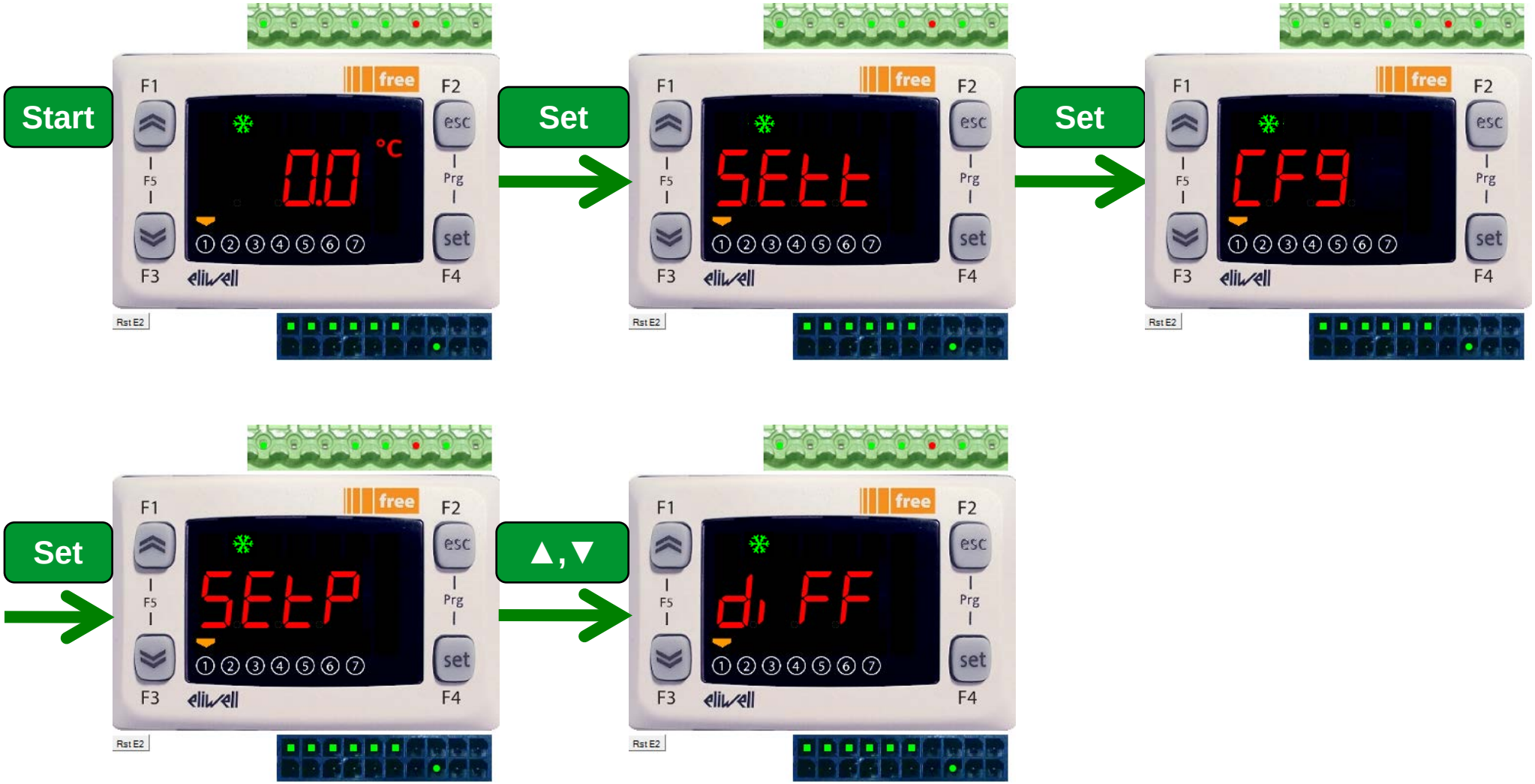
Cfg

Address	Name	Value	Um	Default	Min	Max	Description
16384	Setpoint	180.0	°C	180.0	150.0	300.0	
16385	Differentiation	20.0	°C	20.0	5.0	50.0	

Menu Navigation



Menu Navigation



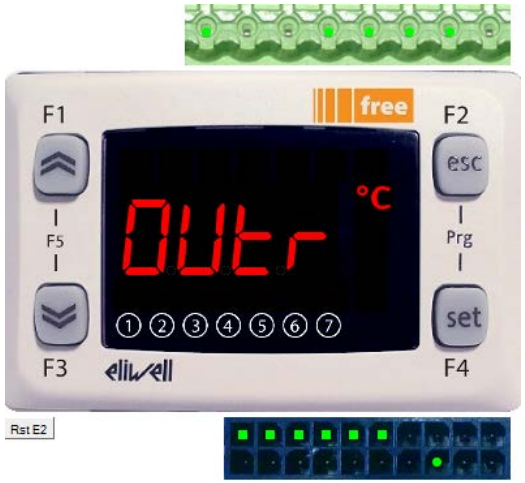
Setting the setpoint

Defaul value aligned with the Free Studio Device

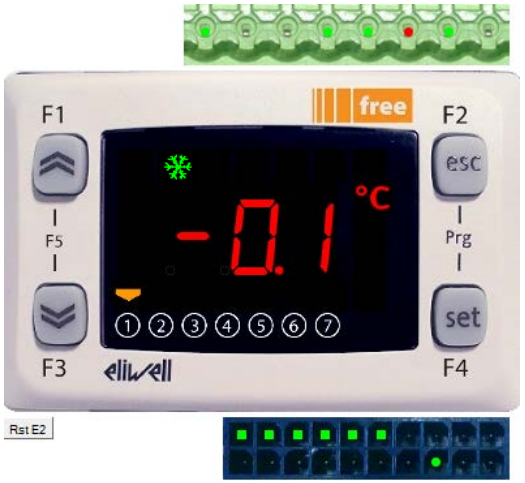
Press Set to validate
Press esc to cancel

Setting the differentiation

Message



Out of range message
Only can disply:
- 99.9.....999.9



Free Studio Device does not write default values

Testing program/applying values

- **Setpoint=26.0, Differentiation=10.0 & Ambient_Tempereature =37.0**

⇒ DOL1= ON & ☼ = ON

- **Setpoint=26.0, Differentiation=10.0 & Ambient_Tempereature =25.0**

⇒ DOL1= OFF & ☼ = OFF

- **Setpoint=26.0, Differentiation=10.0 & 26.0<Ambient_Tempereature<36.0**

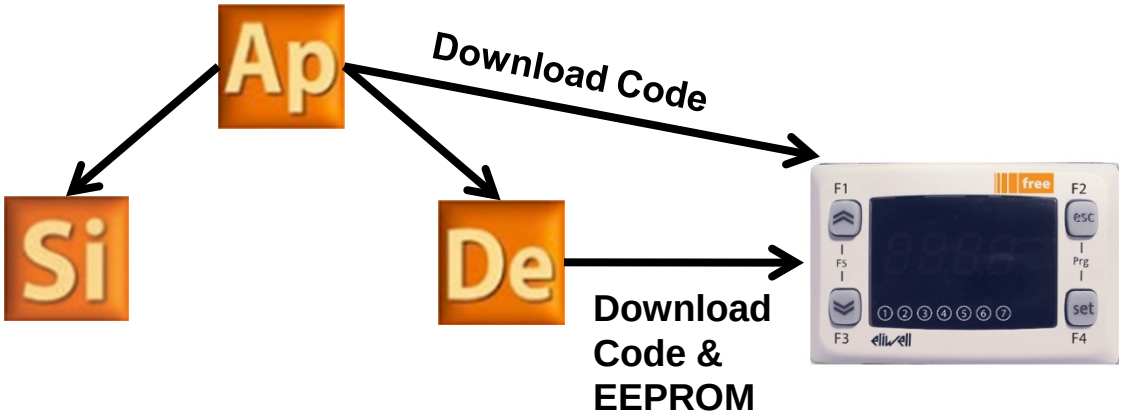
⇒ DOL1= ON & ☼ = ON

- **Ambient_Tempereature =-32768**

=> DOL1= OFF , ☼ = OFF & DOL2= ON (probe disconnection alarm= ON)

Smart Project Architecture

- Application is the programming starting point.
- Device is used to download the overall compiled project and it is the only tool able to write EEPROM parameters.
- From Application it will always be possible to open Device directly without having to launch the program using the FREE Studio icon.

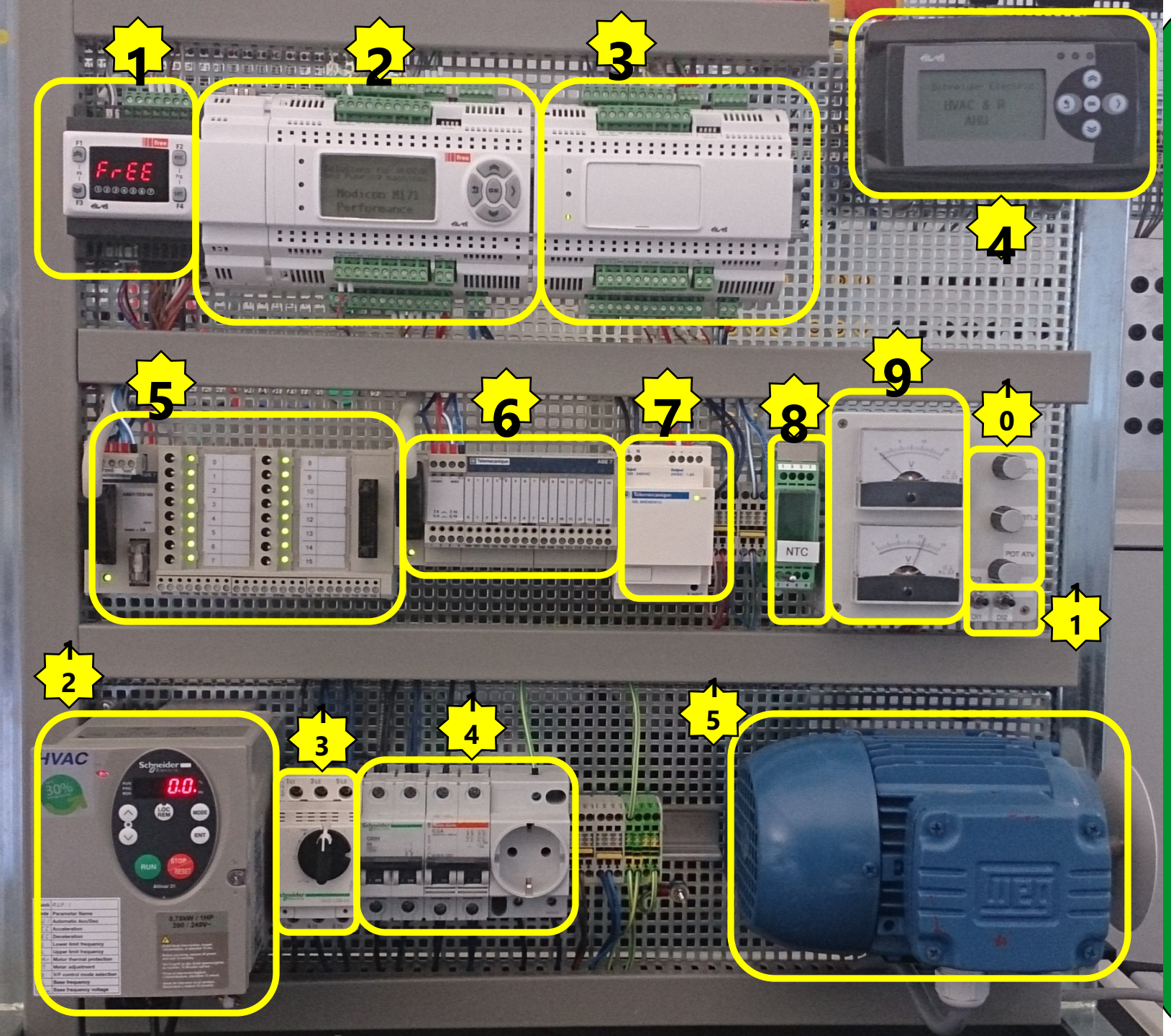


Chapter 6

Hardware

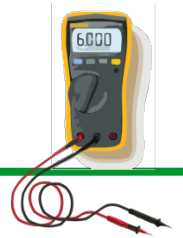
Goal:

Introduction of products that are used in training stand and target pin-out



1. M171O/SMART
2. M171P/EVOLUTION
3. EVE*
4. EVK1000
5. Digital input
   - I0-I7 ► M171P
   - I8-I15 ► EVE*
6. Digital Output
   - Q0-Q3 ► M171O
   - Q4-Q9 ► M171P
   - Q10-Q15 ► EVE*
7. 24VDC power supply
8. NTC probes (AI1*)
9. Analogue Output
   - AO1 ► M171O
   - AO2 ► M171P
10. Analogue Input
    - AI1 ► M171O
    - AI2 ► M171P
    - AI3 ► ATV21
11. Digital input
    - I0 & I1(level) ► M171O
    - I2 & I3 (edge)► M171O
12. Variable Speed Drive ATV21 (1 to 3 phases)
13. Short circuit ptotection (GV3P)
14. LV distribution & protection
15. Asynchronous motor

Training Stand I/O wiring diagram

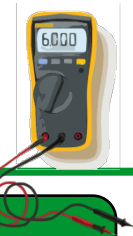


M171O	Description	Label
DI1	Switch DI 1 (level)	DI0-O
DI2	Switch DI 2 (Level)	DI1-O
DI3	Switch DI 3 (Pulse)	DI2-O
DI4		
DI5		
DI6		
AI 1	NTC 1	AI1-O
AI 2		
AI 3	Potentiometer 1	AI3-O
AI 4		
AI 5		
DO 1	Telefast DO 0	DO0-O
DO 2	Telefast DO 1	DO1-O
DO 3	Telefast DO 2	DO2-O
DO 4		
DO 5		
DO 6		
AO 1		
AO 2		
AO 3	Meter 1	AO1-O
AO 4		
AO 5		

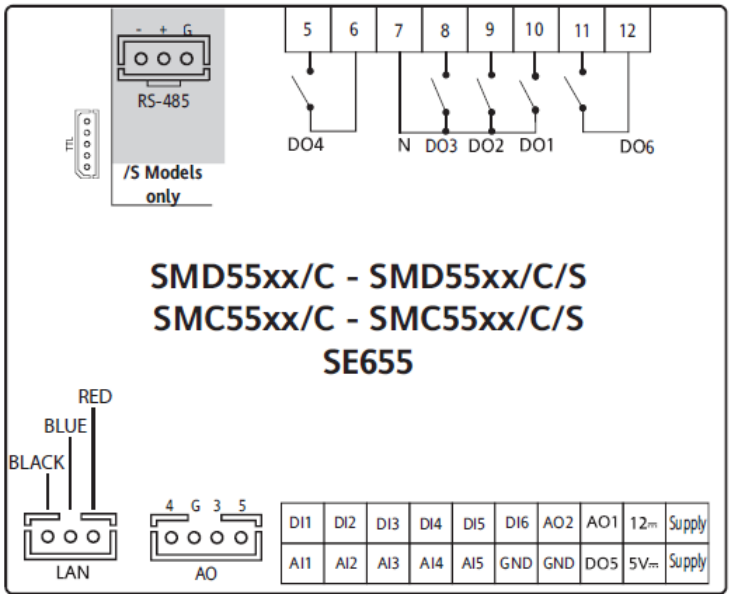
M171P	Description	Label
DI 1	Telefast DI 0	DI1-P
DI 2	Telefast DI 1	DI2-P
DI 3	Telefast DI 2	DI3-P
DI 4	Telefast DI 3	DI4-P
DI 5	Telefast DI 4	DI5-P
DI 6	Telefast DI 5	DI6-P
DI 7	Telefast DI 6	DI7-P
DI 8	Telefast DI 7	DI8-P
AI 1	NTC 2	AI1-P
AI 2		
AI 3	Potentiometer 2	AI3-P
AI 4		
AI 5		
AI 6		
DO 1		
DO 2		
DO 3	Telefast DO3	DO3-P
DO 4	Telefast DO4	DO4-P
DO 5	Telefast DO5	DO5-P
DO 6	Telefast DO6	DO6-P
DO 7	Telefast DO7	DO7-P
AO 1	Meter 2	AO1-P
AO 2		
AO 3		
AO 4		
AO 5		

M171E	Describtion	Label
DI 1	Telefast DI 8	DI1-E
DI 2	Telefast DI 9	DI2-E
DI 3	Telefast DI 10	DI3-E
DI 4	Telefast DI 11	DI4-E
DI 5	Telefast DI 12	DI5-E
DI 6	Telefast DI 13	DI6-E
DI 7	Telefast DI 14	DI7-E
DI 8	Telefast DI 15	DI8-E
AI 1	NTC 3	AI1-E
AI 2		
AI 3		
AI 4		
AI 5		
AI 6		
DO 1	Telefast DO9	DO1-E
DO 2	Telefast DO10	DO2-E
DO 3	Telefast DO11	DO3-E
DO 4	Telefast DO12	DO4-E
DO 5	Telefast DO13	DO5-E
DO 6	Telefast DO14	DO6-E
DO 7	Telefast DO15	DO7-E
AO 1		
AO 2		
AO 3		
AO 4		
AO 5		

Wiring Diagram

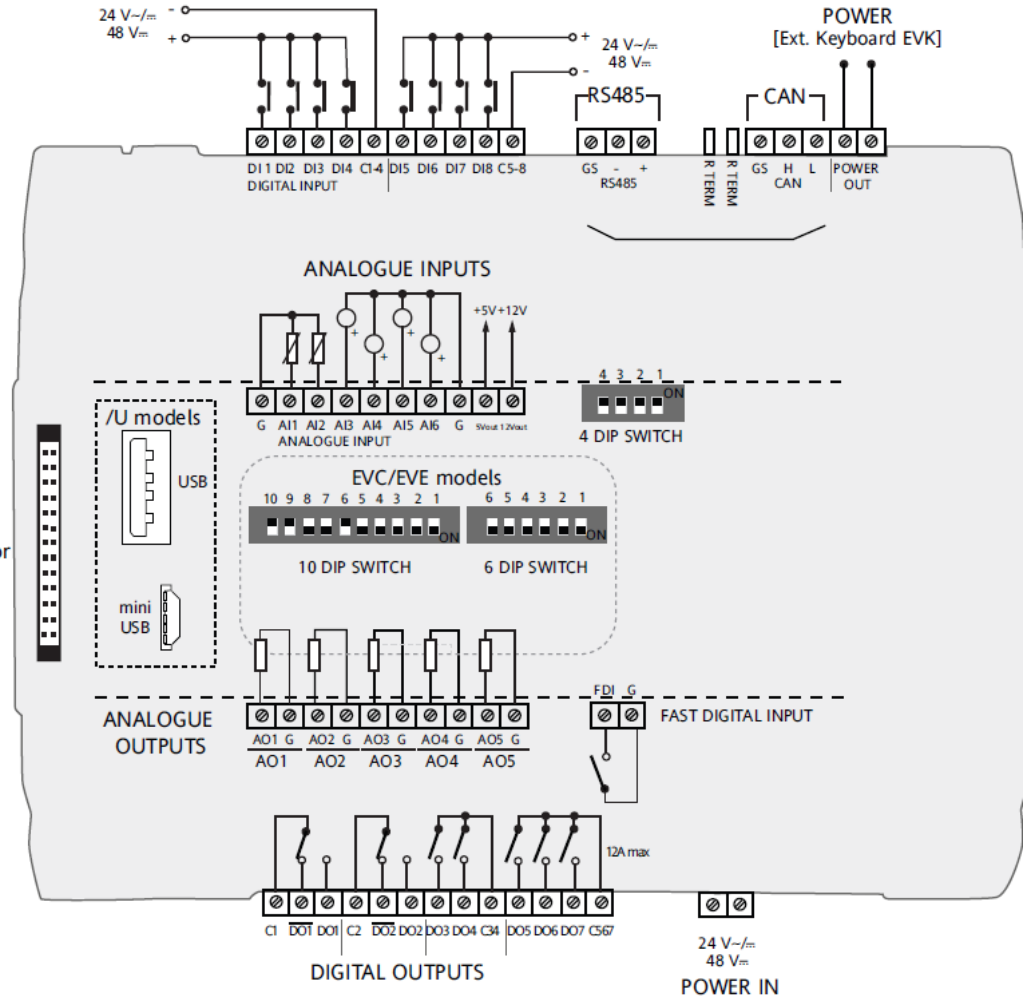


M171O
SMD5500/C/S

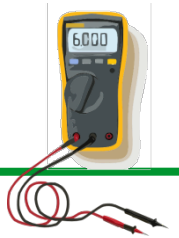


EVS Plugin → PLUG-IN Connector

M171P
EVD7500/C/U



NTC probe,Type: 103AT/NTCNK103

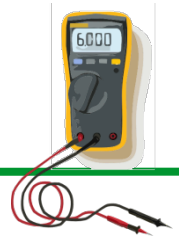


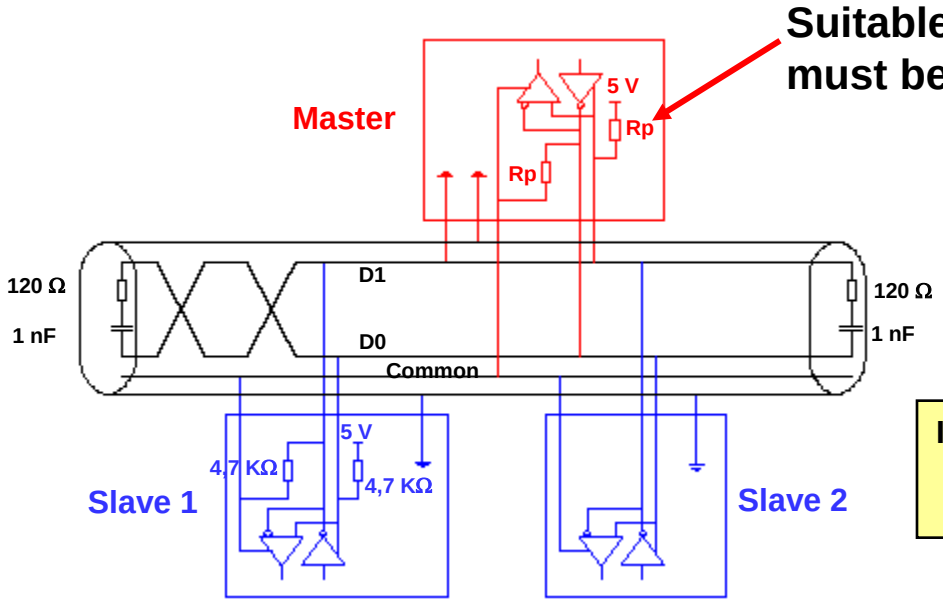
Temperature probes (*)	SN691150	NTC 103AT probe, 1.5m (plastic cap, 2-wire cable)
	SN8DED11502C0	NTC103AT 5X20 1.5mt TPE IP68
	SN8DED13002C0	NTC103AT 5X20 3mt TPE IP68
	SN8DAD11502C0	NTC103AT 6X20 1.5mt TPE IP68
	SN8DAD13002C0	NTC103AT 6X20 3mt TPE IP68

	NTC* -50..+100°C	**0/4...20 mA**	**0-10V**	**0-5V**	**0-1V**	**DI**
Resolution	0.1°C	0.1	0.1	0.1	0.1	
Accuracy	1% e.o.s. f.s.	1% e.o.s. 1% f.s.	1% e.o.s. 1% f.s.	1% e.o.s. 1% f.s.	2% e.o.s. 2% f.s.	
Impedence		100Ohm	21KOhm	110KOhm	110KOhm	

	NTC* -50..+100°C	**0/4...20 mA**	**0-10V**	**0-5V**	**0-1V**	**DI**
AI1	✔	-	-	-	-	✔
AI2	✔	-	-	-	-	✔
AI3	✔	✔	✔	✔	✔	✔
AI4	✔	✔	✔	✔	✔	✔
AI5	✔	-	-	-	-	✔

Modbus RS485 standard schematic





If the master is fitted with a 470 Ω polarisation, it 's possible to connect a maximum of 18 slaves with 4,7 KΩ polarisation

Maximum length of bus	1000 m at 19200 bps
Maximum number of stations (without repeater)	At most 32 stations (depending on Rp and the number of 4,7 KΩ
Maximum length of tap links	20 m for one tap link 40 m divided by the number of tap links
Bus polarisation	Rp should be validated by calculating the equivalent polarisation Re according to the polarisation of the master and slave stations. Re must be between 162 Ω and 650 Ω
Line terminator	120 Ω - 0,25Wm in series with 1nF 10V
Common polarity	Yes (Common) connected to the PG

Chapter 7

Connection to Smart

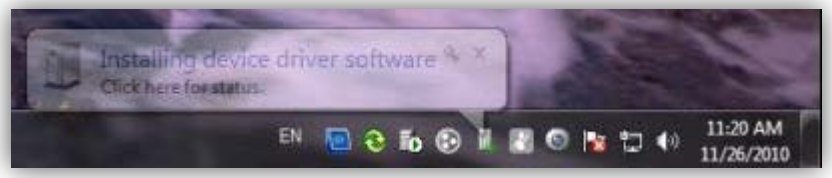
Goal:

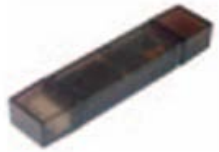
DMI interface driver installation and connect to the target

DMI interface setup WIN 7

- As soon as the DM interface is connected, the Windows 7 operating system recognizes the newly connected hardware. The steps to be followed are described below.

1. Once the hardware is connected, the message shown in the figure will appear:





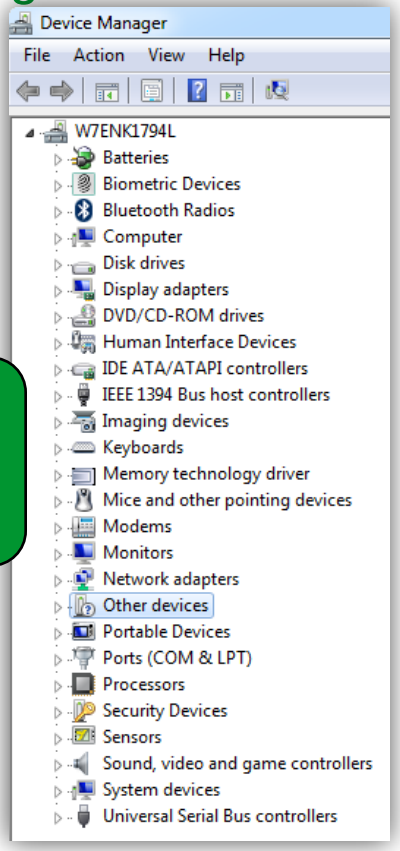
Click on the message to start the Guided installation procedure

Or you can manualy find it at:

Control Panel ▸ All Control Panel Items ▸ System

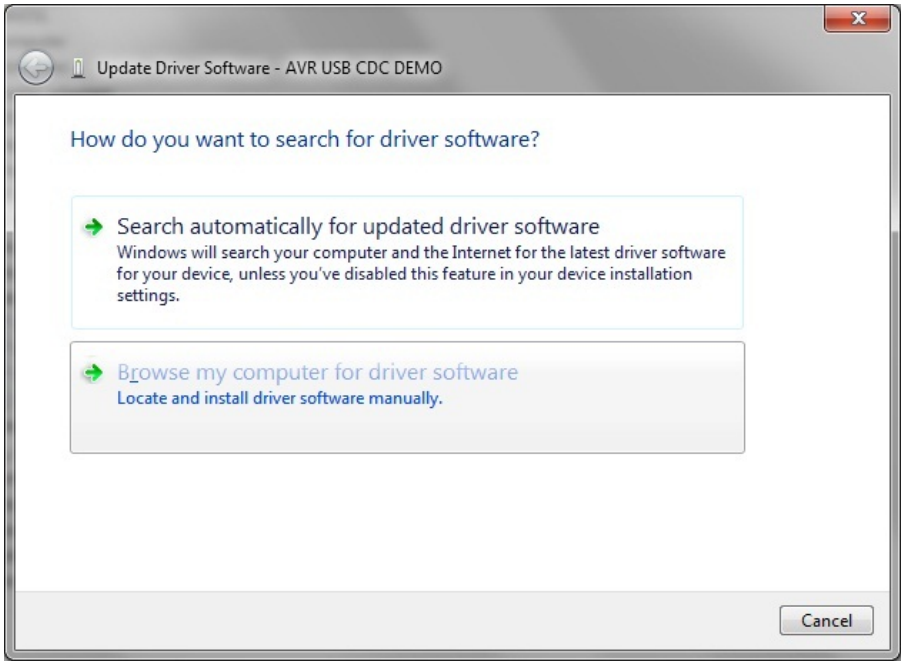
Device manager ► other devices ►

Note: Connection procedure:
Connection: first USB then TTL
Disconnection: first TTL then USB



Update driver software

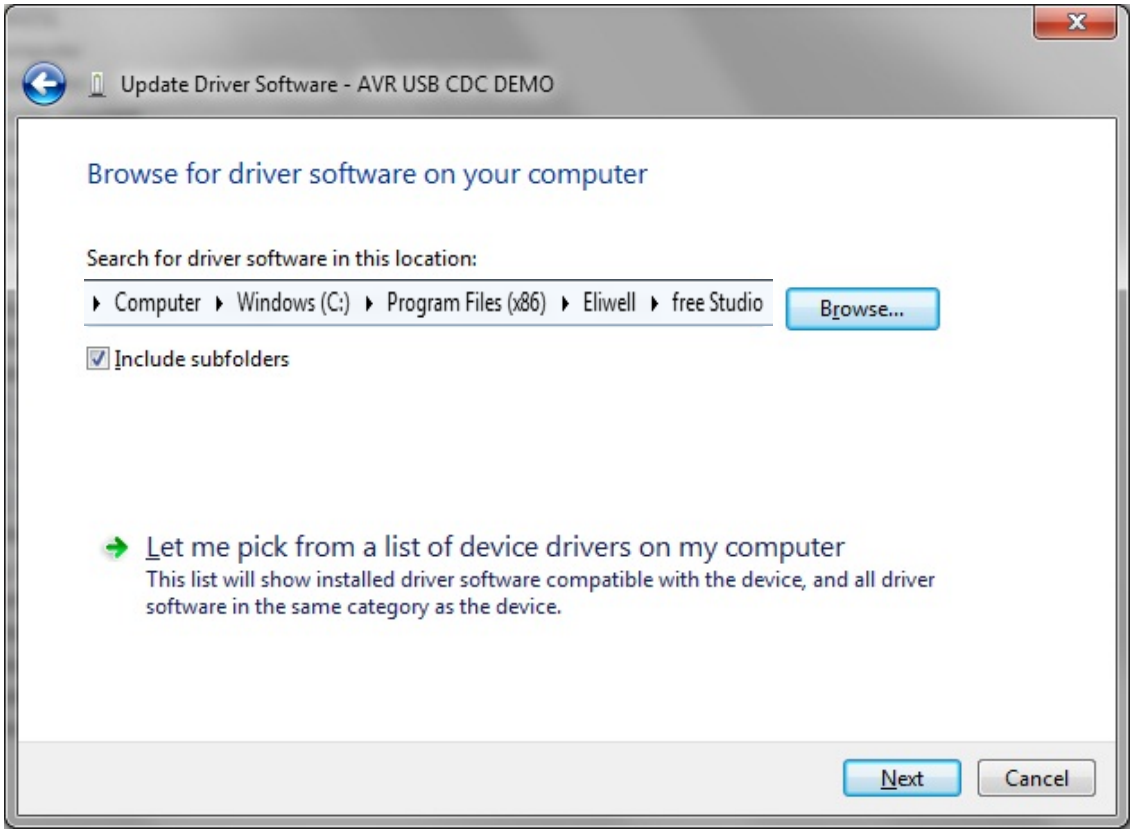
2. The screen shown below will appear: select the second option to identify the driver



3. In the next screen, select the installation path for the FREE Studio program. Unless changed during installation, the path will be as shown in the next figure.

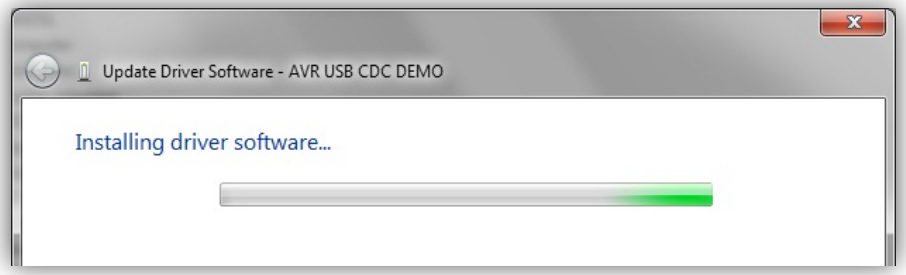
Update driver software

4. Once you have selected the correct path, the screen shown below will appear: select **Install this driver software anyway**



Update driver software

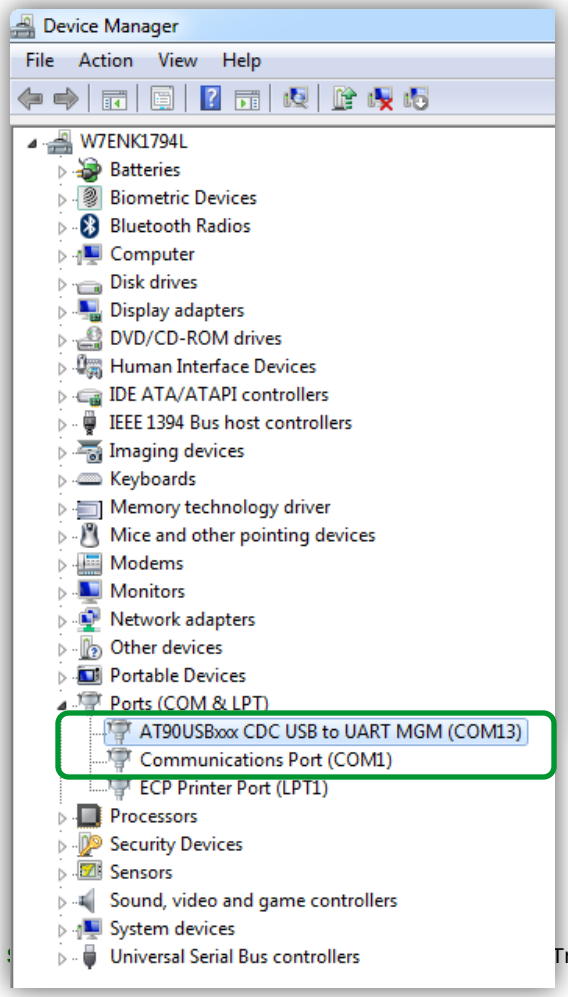
5. The screen shown below will appear, indicating that the action has been performed.

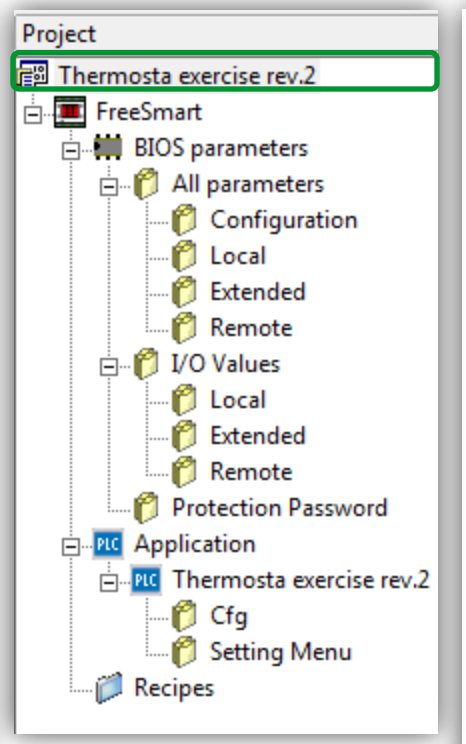


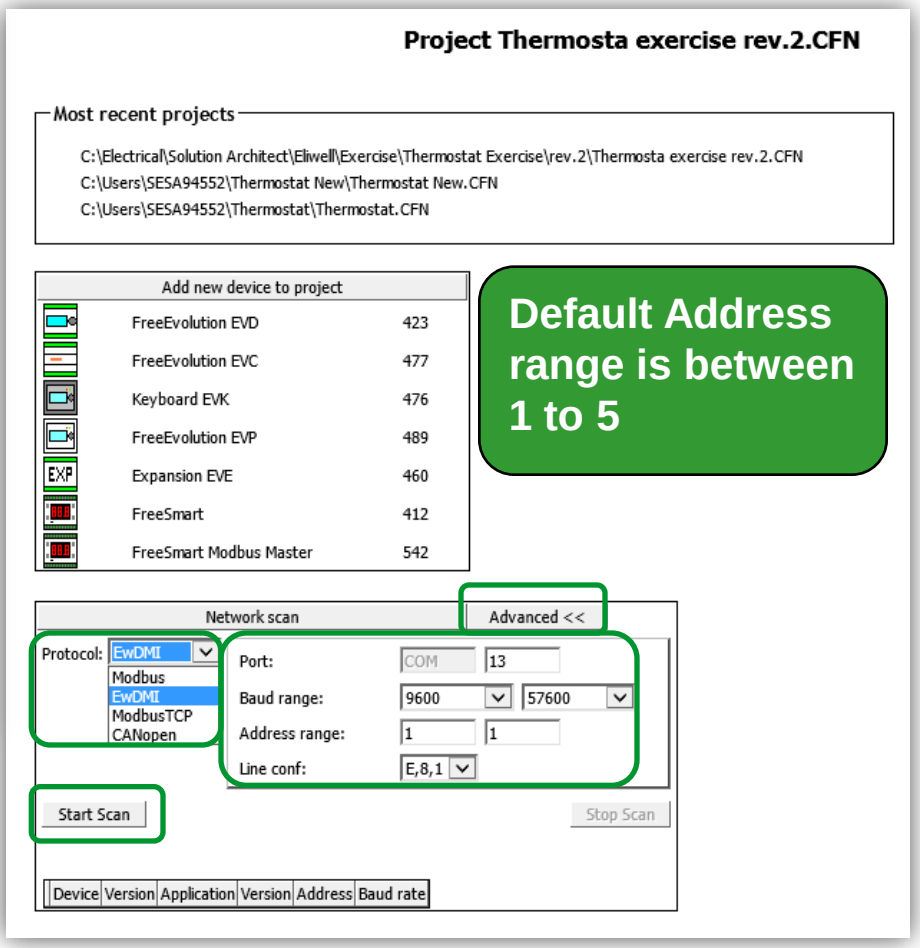
6. On completion of the process, the screen shown below will appear, then close.

DMI Test via FS Device

To check correct installation of the driver and the port to which the hardware has been allocated, check the Windows screens shown below:





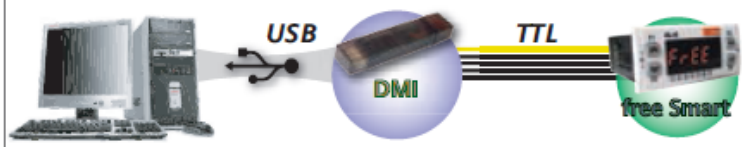


Connection to Smart

To download the IEC applications of **Studio from the personal computer to the Smart target device,**

Direct

PC → **← Smart**



Use yellow TTL cable for DMI - target connection

direction Data downloading	→	←
Parameter map	✓	✓
IEC application	✓	-
BIOS	✓	-

Note: in “Direct”, **Smart must not be connected to earth**

- it can switch on Smart without external power supply

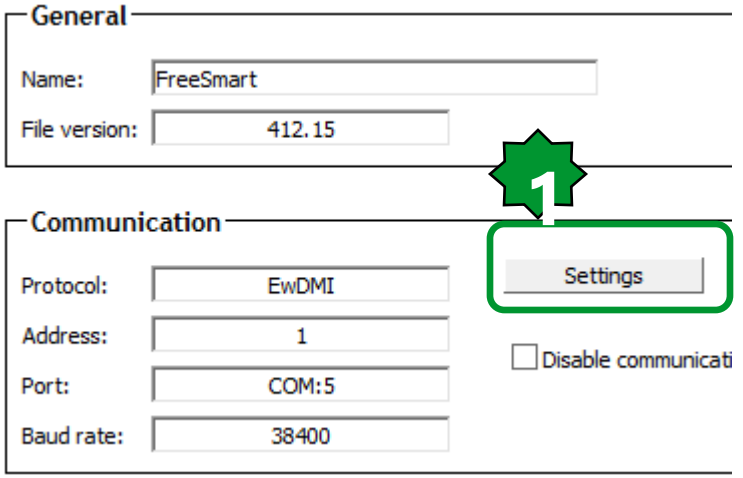
Connection to Smart

Smart

Preliminary operations

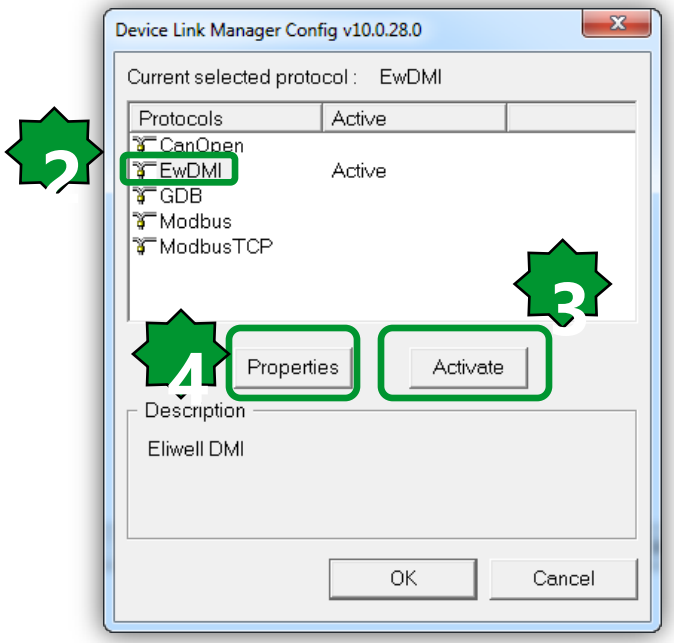
In order to download the application correctly:

1. connect the DMI hardware interface to the PC.
2. Make sure that the driver is installed



Press **Settings**

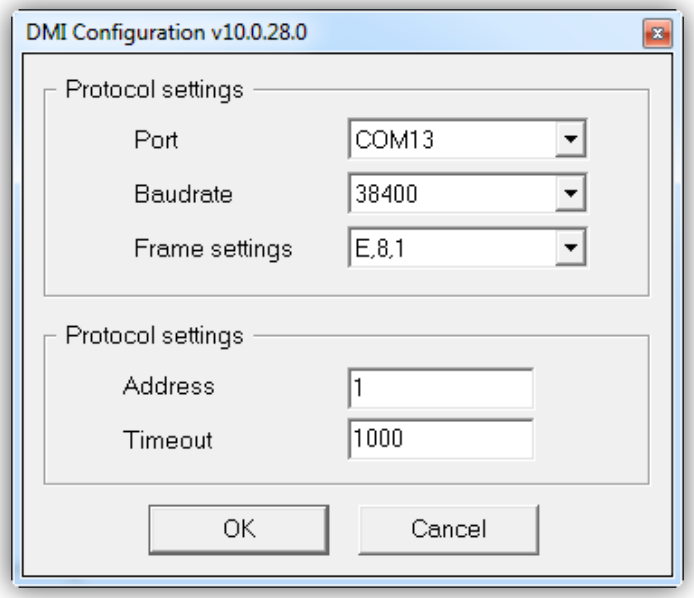
The COM port must previously have been read/set in “Peripherals Management” (see Reading the DMI interface COM port) to be recognized. If there are errors, refer to the paragraph “**DMI interface connection error”.**



* the COM settings must be set on all of the workspaces: Application, Device and User Interface

Protocol Configuration

- For **Smart select EWDMI or Modbus*. If the protocol is not activated press the Activate button**
- The value selected for the COM port will be saved and will reappear each time the program is accessed, until it is changed.
- The properties are visible and can be edited from the panel **Communication > Settings > Properties****



*** in the case of Modbus for /S models only with maximum speed 19200 baud. TTL not for use. NOT POSSIBLE TO UPDate the BIOS.**

**** obviously, the protocol must be activated beforehand**

Factory default configuration:

Address:1, Baud rate: 9600
E,8,1 (CF30=1, CF31=3, CF32=1)

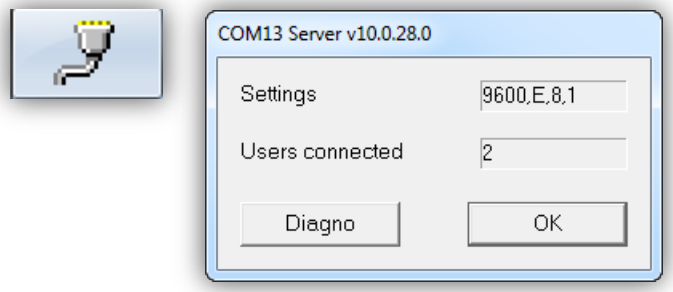
DMI interface connection error

Error opening serial port

If the “Error opening serial port” message appears, proceed as follows:

1. Check that the COM port setting in the program is the same as one read in the COM port reading by the DMI interface.

2. Check if Com Server is opened when you try to connect to Evolution. If not disconnect TTL cable, USB port and reconnect first USB and then TTL.



3. Repeat the DMI Detection function.

Target and Free Studio

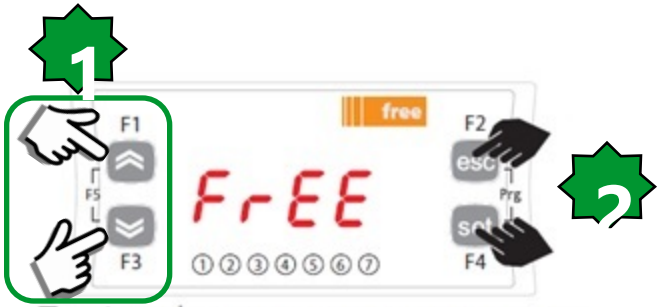
Parameters needed for correct connection between the **Smart target and Free Studio.**

parameter	description	values	default	visibility	notes
CF30	Modbus protocol controller address	1...255	1	3	Check that the set values correspond to those defined by the panel **Communication > Settings > Properties**
CF31**	Modbus protocol baud rate	0,1, 2 = not used 3 = 9600 baud 4 = 19200 baud 5 = 38400 baud 6 = 57600 baud 7 = 115200 baud	3	3	
CF32	Modbus protocol controller parity	1= EVEN 2 = NONE 3 = ODD	1	3	
*COM1 = TTL / RS485 (/S models only): cannot be used simultaneously					
**CF31			5=38400 baud (RS485: not supported) 6=57600 baud (RS485: non supported) 7=115200 baud (RS485: non supported)		

Customize Smart Baud Rate

Smart parameters in the CF folder manages the connection between the target and Studio

If the target is “empty”, i.e. there is no IEC application on the device, Smart will display the message FrEE, otherwise fundamental state is displayed (Press F5 to switch to FrEE menu)



To view the parameter menu, press the Esc and Set keys at the same time. This will open the PAr menu.

The parameters menu PAr contains all controller folders. Press the set key to view folders.

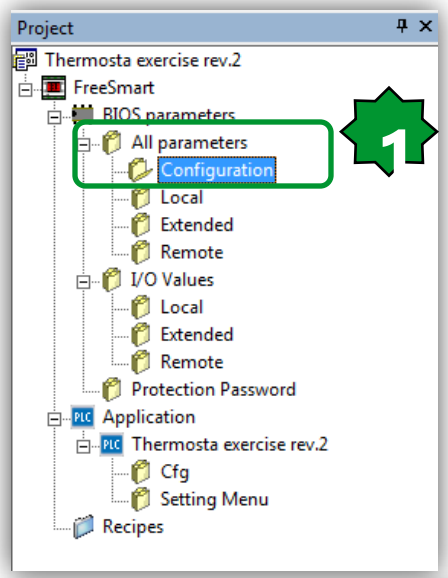
The first folder shown is the CF configuration folder. Press the set key to view the folder parameters.

The first parameter shown is CF30. To view the value of the parameter press the set key.

Use the UP and DOWN keys to change the value if necessary. To confirm the value press the set key.
To exit press Esc

Use the UP and DOWN keys to scroll the other parameters and repeat the procedure to view the values and - if necessary - edit them.

Customize Smart Baud Rate by FS Device



Only After Connection has been estabilished:

1. Project ►BIOS parameters►All parameters►Configuration
2. CF31 editing►38400 bits/Sec.

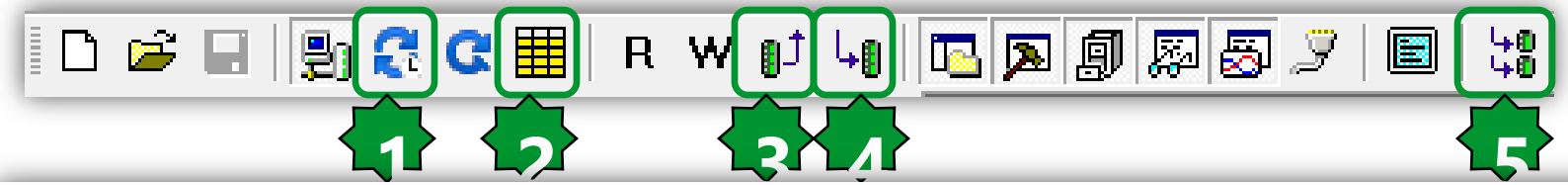
Protocol parameters are loaded at power up, remember to switch off controller after changing them.

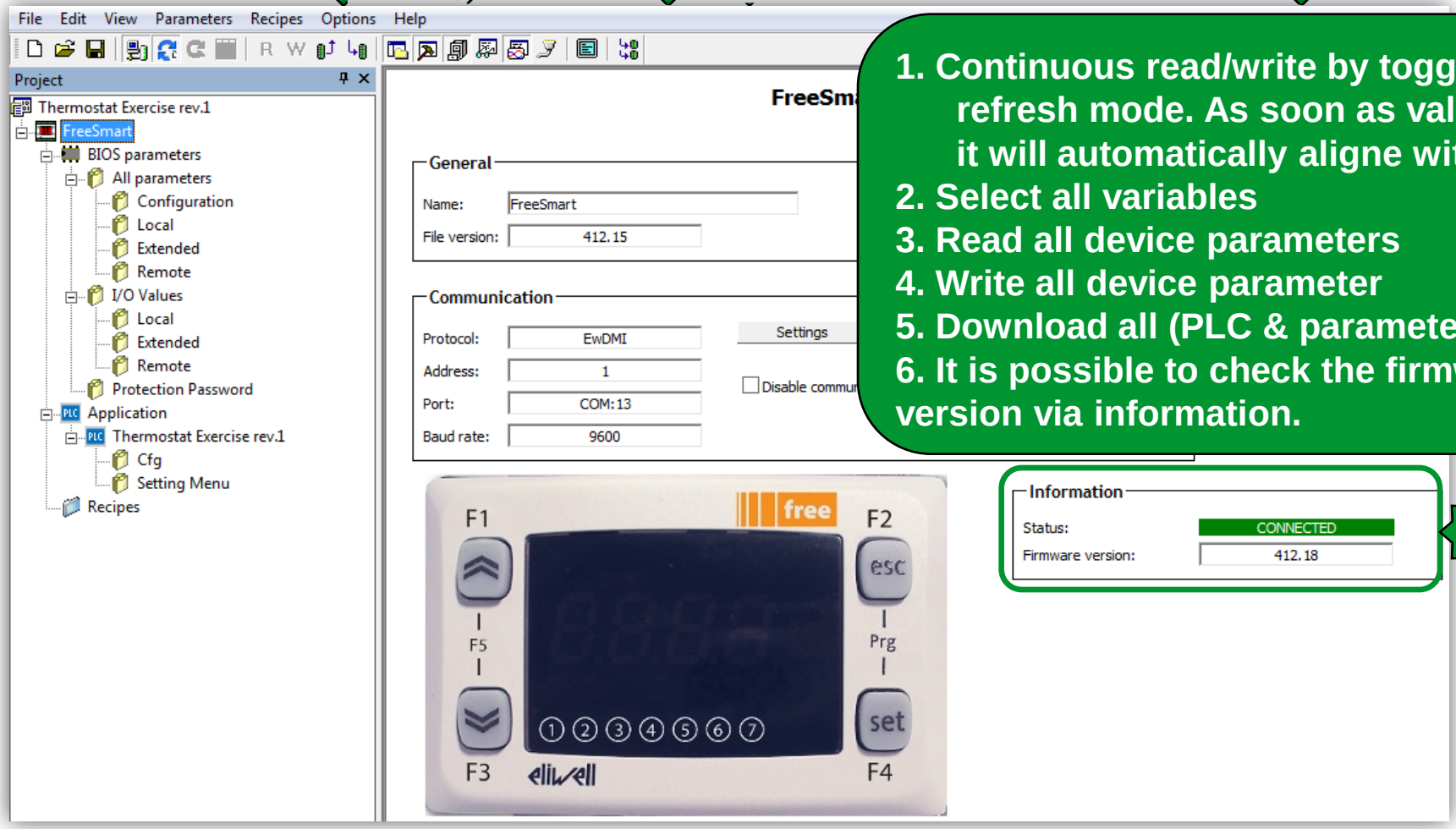
Configuration

Address	Name	Value	Um	Default	Min	Max	Description
53265	CF01	1	num	1	0	1	Select COM1 protocol
53272	CF20	0	num	0	0	14	Eliwell protocol controller address
53273	CF21	0	num	0	0	14	Eliwell protocol controller family
53274	CF30	1	num	1	1	255	Modbus protocol controller address
53275	CF31	5=38400	num	3=9600	0	7	Modbus baud rate protocol
53276	CF32	1=2400 2=4800 3=9600 4=19200 5=38400 6=57600 7=115200	num	1=Even	1	3	Modbus parity protocol
15639	CF60		num	0	0	999	Customer code 1
15640	CF61		num	0	0	999	Customer code 2
53456	CF50		num	1=Present	0	1	RTC present
15715	Ui26		4ms	350	0	999	Key hold time to enable function
15744	Ui27		num	1	0	255	Installation engineer password
15745	Ui28	2	num	2	0	255	Manufacturer password
15636	Par_POLI	1026	num	0	0	65535	Polycarbonate code

2

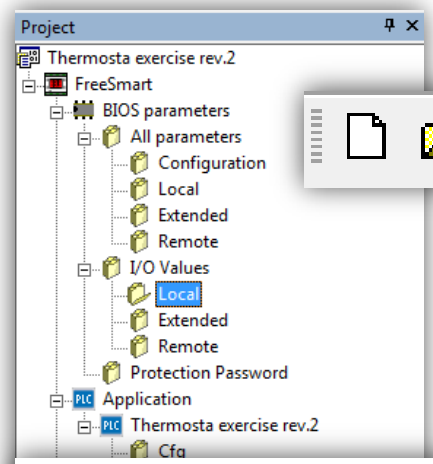
Free Studio Device - Main icons

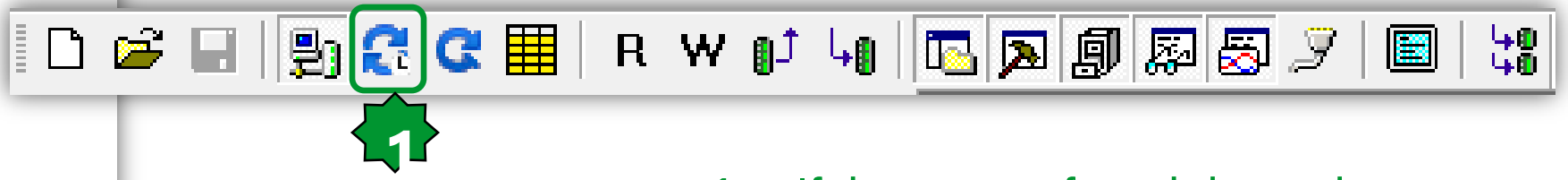




1. Continuous read/write by toggle auto refresh mode. As soon as value changes, it will automatically aligne with the target.
2. Select all variables
3. Read all device parameters
4. Write all device parameter
5. Download all (PLC & parameter)
6. It is possible to check the firmware version via information.

Free Studio Device - Colors





Local

Address	Name	Value	Um	Default	Min	Max	
8336	AIL1	0.0		0.0			AIL1 analogue input
8337	AIL2	0.0		0.0			AIL2 analogue input
8338	AIL3	0.0		0.0			AIL3 analogue input
8339	AIL4	0.0		0.0			AIL4 analogue input
8340	AIL5	0.0		0.0			AIL5 analogue input
8192	DIL1	False		False			DIL1 digital input
8193	DIL2	False		False			DIL2 digital input
8194	DIL3	False		False			DIL3 digital input
8195	DIL4	False		False			DIL4 digital input
8196	DIL5	False		False			DIL5 digital input
8197	DIL6	False		False			DIL6 digital input
8528	DOL1	False		False			DOL1 digital output
8529	DOL2	False		False			DOL2 digital output
8530	DOL3	False		False			DOL3 digital output
8531	DOL4	False		False			DOL4 digital output
8532	DOL5	False		False			DOL5 digital output
8533	DOL6	False		False			DOL6 digital output
8448	AOL1	0.0		0.0			AOL1 analogue output
8449	AOL2	0.0		0.0			AOL2 analogue output
8450	AOL3	0.0		0.0			AOL3 analogue output
8451	AOL4	0.0		0.0			AOL4 analogue output
8452	AOL5	0.0		0.0			AOL5 analogue output
8453	TCL1	0.0		0.0			TCL1 analogue output

1. If the auto referesh is on the column value is continiously updated and values are written as soon as you change them.

Color meanings:

Red: not aligned with the target

Grey: read only data

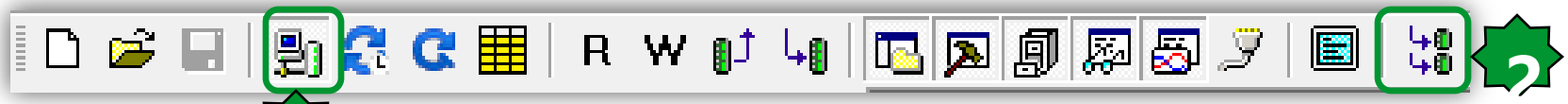
Blue: value is different from default

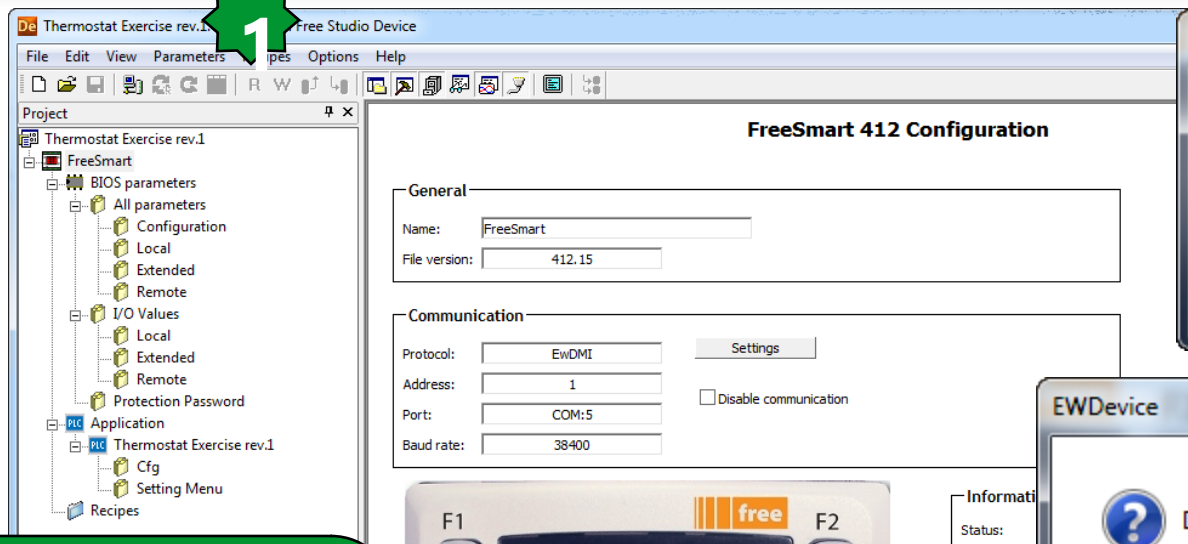
Green: data is not visible in the target

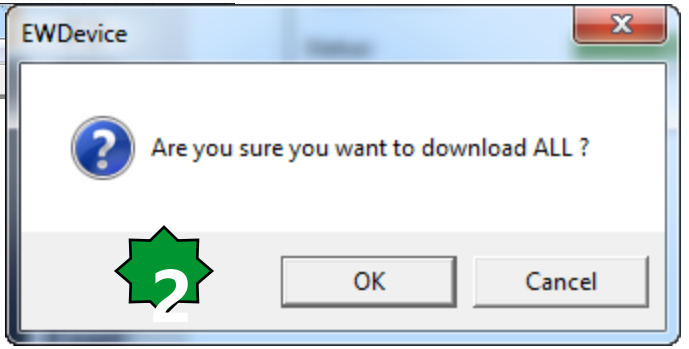
Black: aligned with the target

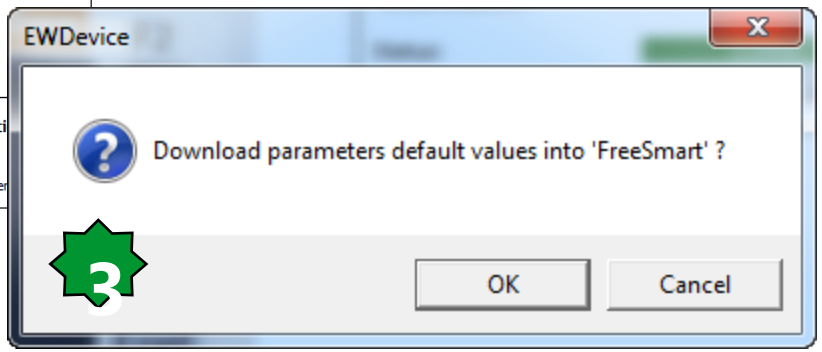
(if auto referesh is enabled) 1

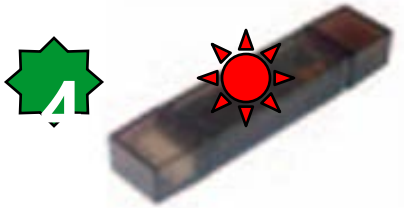
Connect to the target and Download All











1. Connect
2. Download All
3. Write the default parameter values
4. DMI Blink: Communicating

Editing value

1. Be sure to be Connected
2. Set or change the value (Red, not applied)
3. Write the parameter (Black, applied)

CONNECTED

Setting Menu

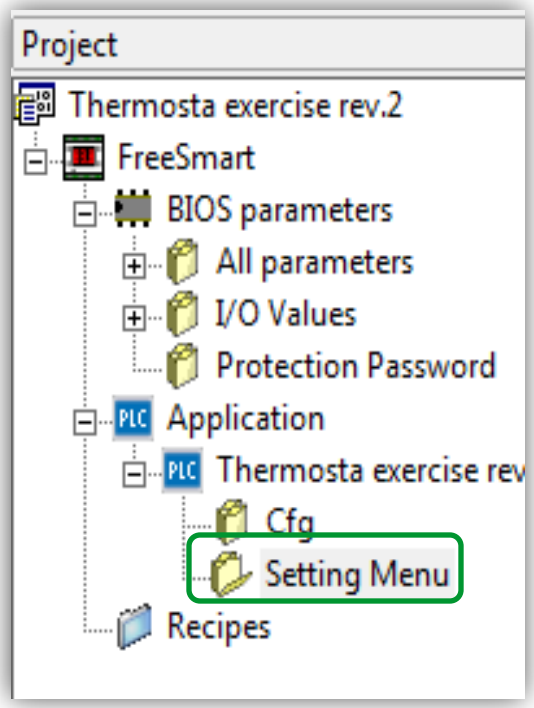
Address	Name	Value	Um	Default	Min	Max
16384	Setpoint	21.0	°C	18.0	15.0	30.0
16385	Differentiation	2.0	°C	2.0	0.5	5.0
8960	Ambient_Temperator	22.9	°C			

Setting Menu

Address	Name	Value	Um	Default	Min	Max
16384	Setpoint	23.0	°C	18.0	15.0	30.0
16385	Differentiation	2.0	°C	2.0	0.5	5.0
8960	Ambient_Temperator	22.9	°C			

Setting Menu

Address	Name	Value	Um	Default	Min	Max
16384	Setpoint	23.0	°C	18.0	15.0	30.0
16385	Differentiation	2.0	°C	2.0	0.5	5.0
8960	Ambient_Temperator	26.3	°C			

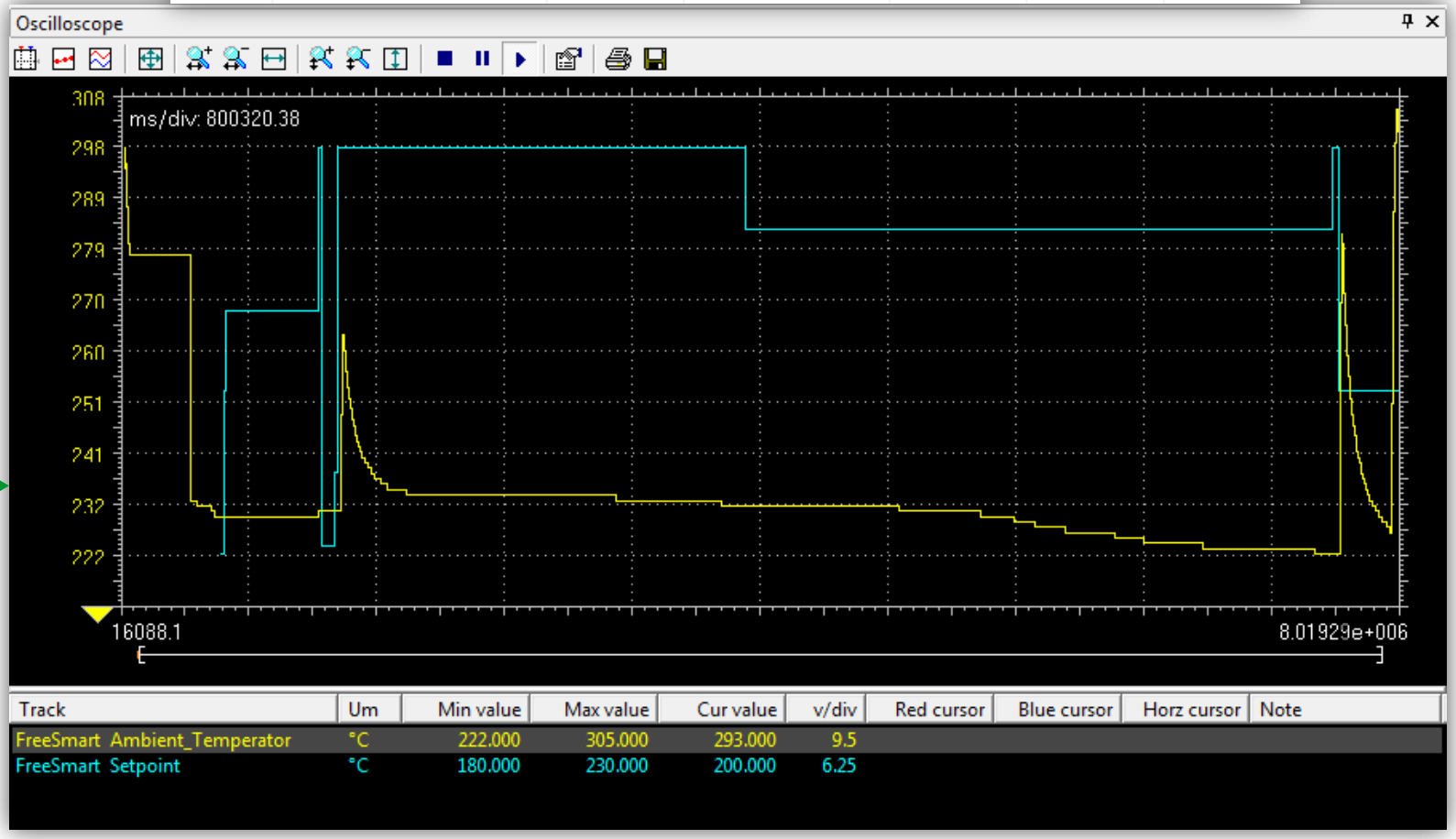


Oscilloscope

Drag & Drop

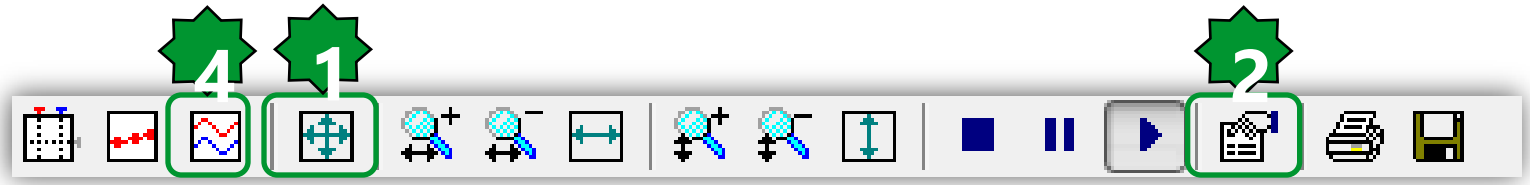
Setting Menu

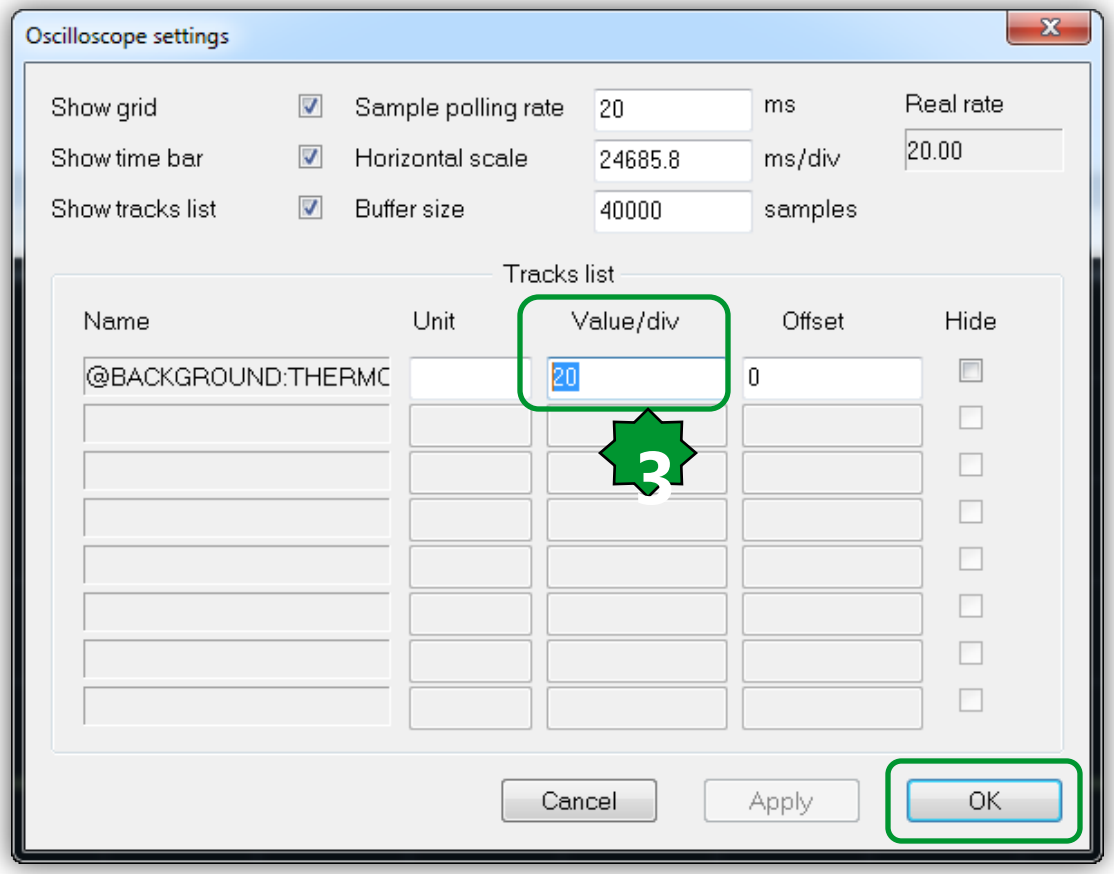
Address	Name	Value	Um	Default	Min	Max
16384	Setpoint	20.0	°C	18.0	15.0	30.0
16385	Differentiation	2.0	°C	2.0	0.5	5.0
8960	Ambient_Temperator	23.1	°C			



Track	Um	Min value	Max value	Cur value	v/div	Red cursor	Blue cursor	Horz cursor	Note
FreeSmart Ambient_Temperator	°C	222.000	305.000	293.000	9.5				
FreeSmart Setpoint	°C	180.000	230.000	200.000	6.25				

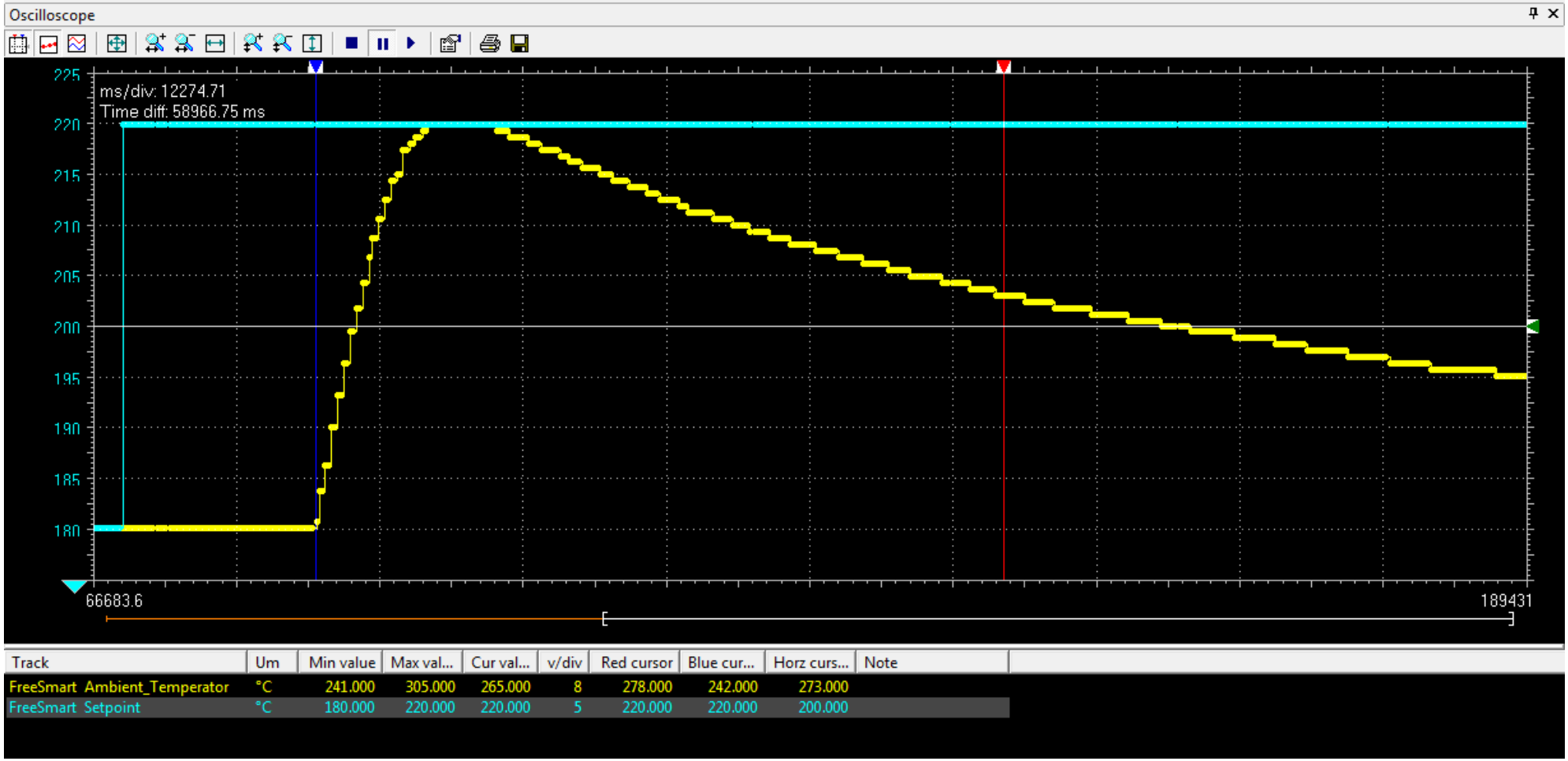
Oscilloscope



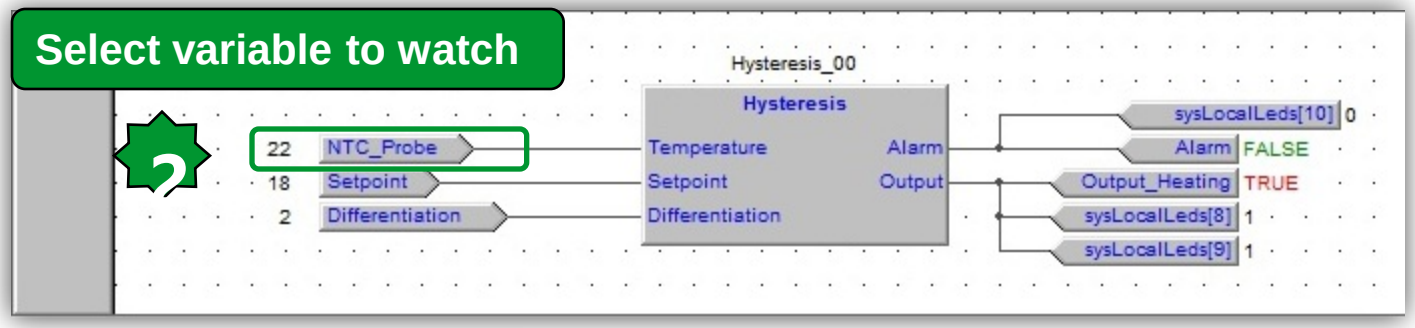


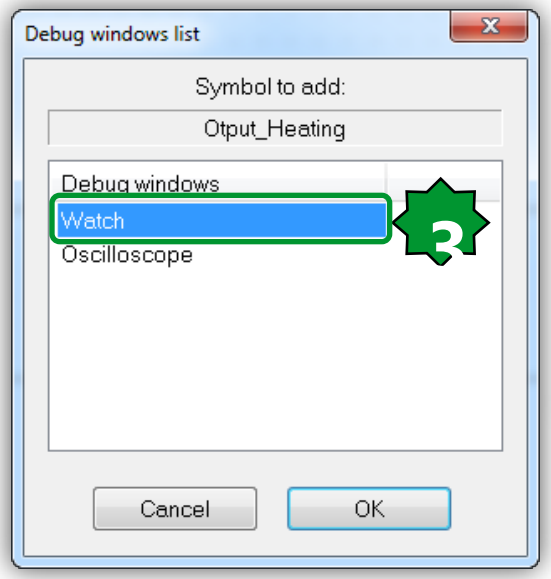
1. **Show all values**
2. **Graph properties**
3. **Value/div=20 ► OK**
4. **Vertical split**

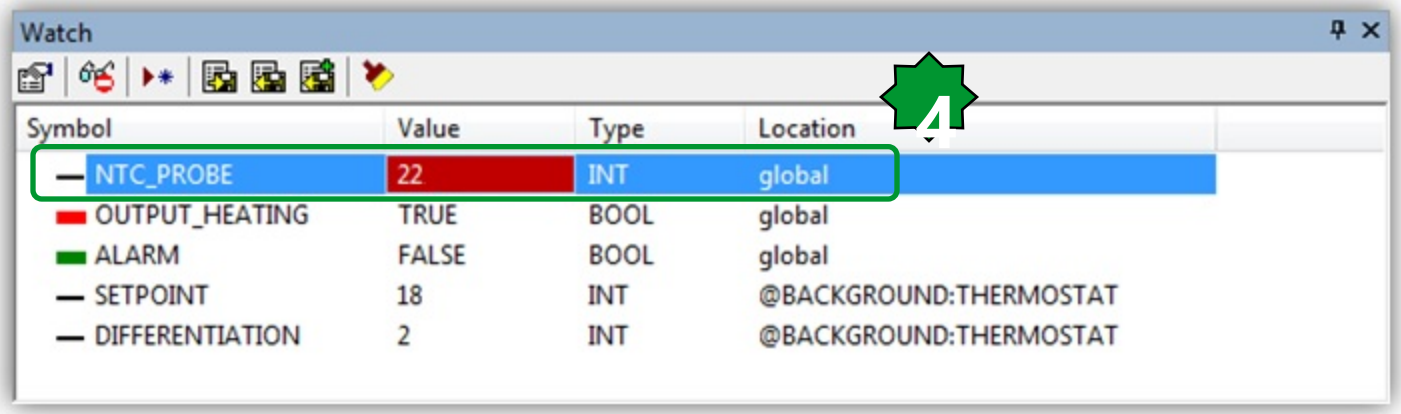
Oscilloscope



Debug on-line/Watch







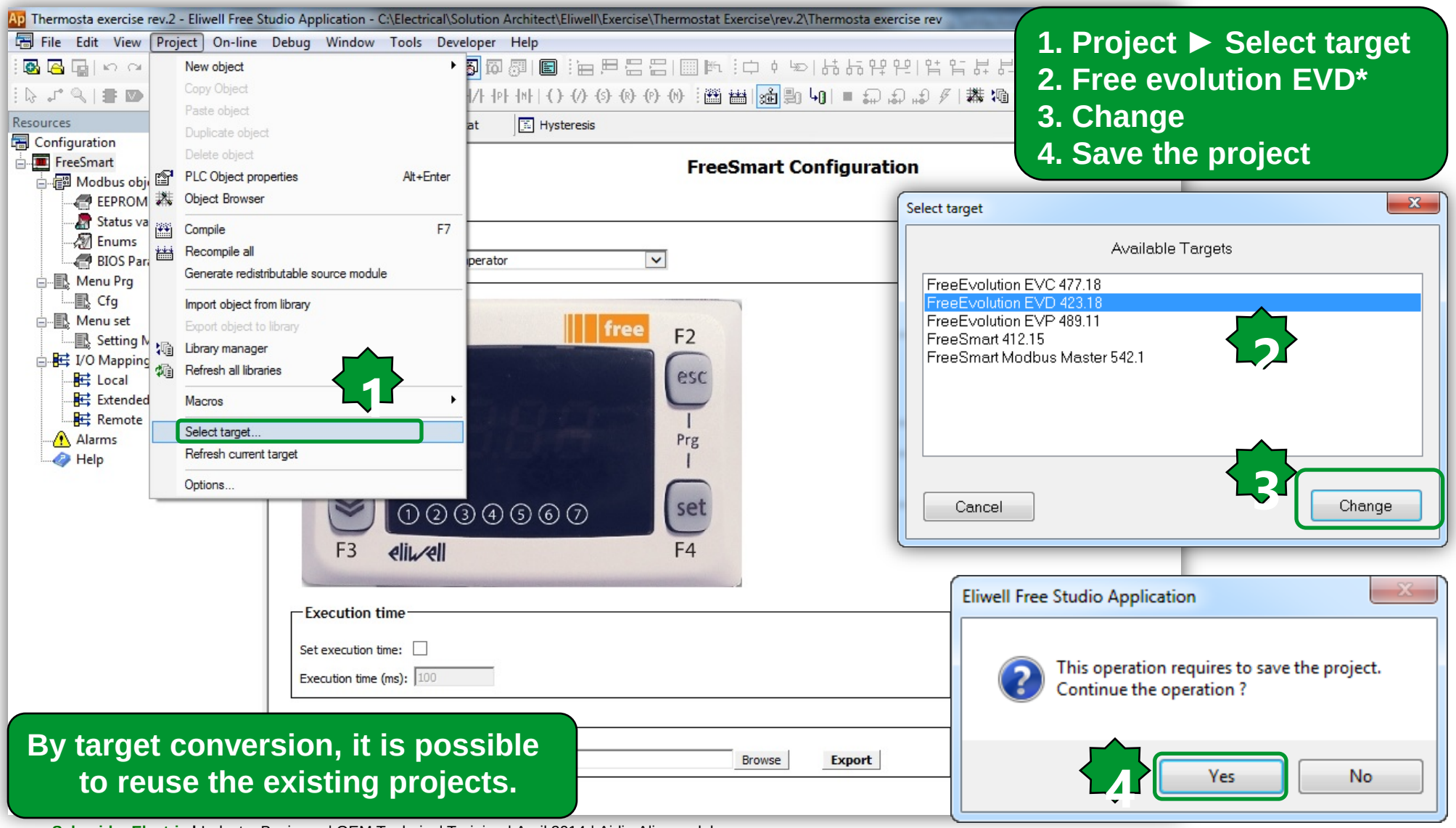
Chapter 8

Target conversion and code import

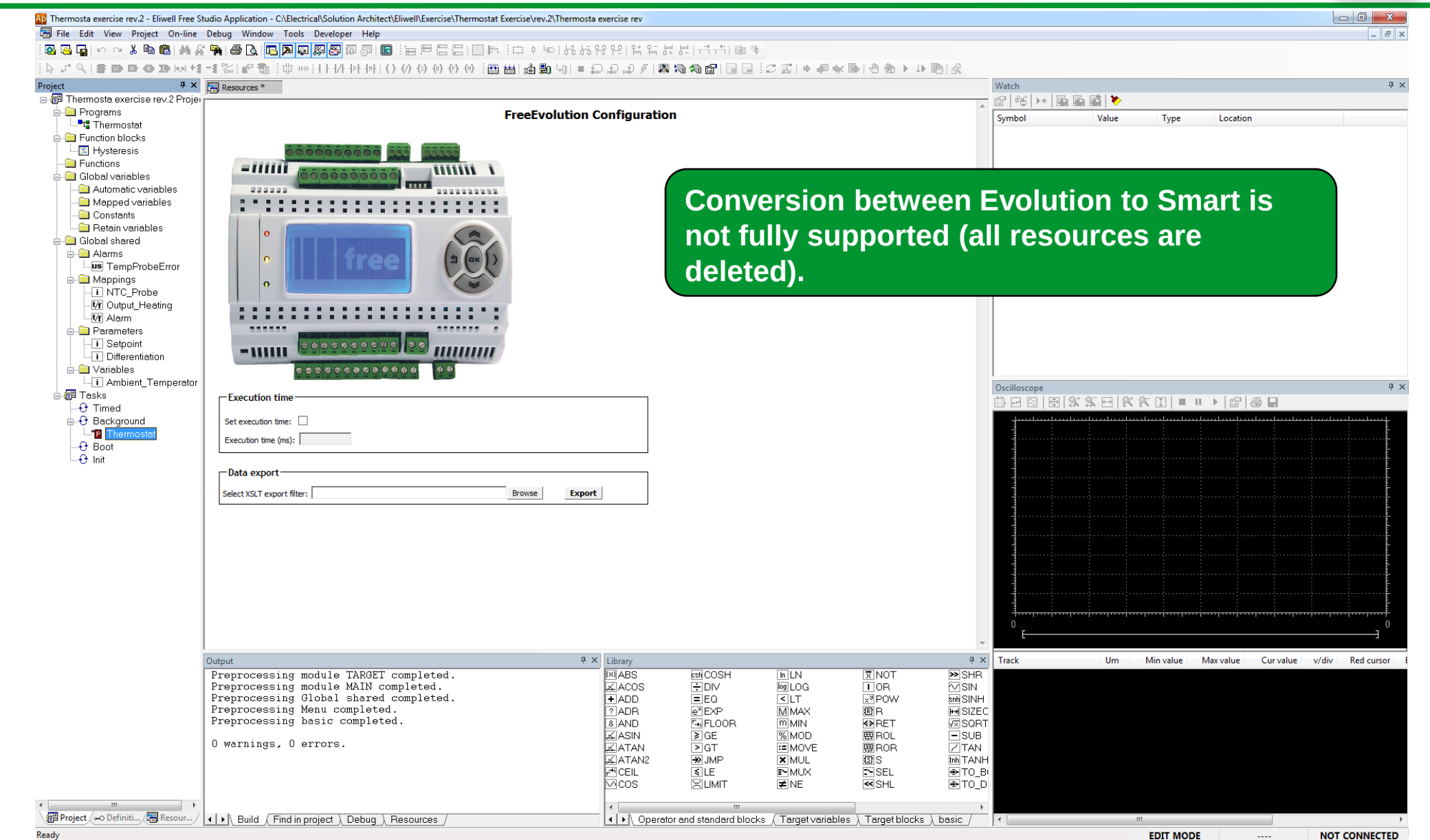
Goal:

Reuse of existing code and libraries

Convert project from Smart to Evolution

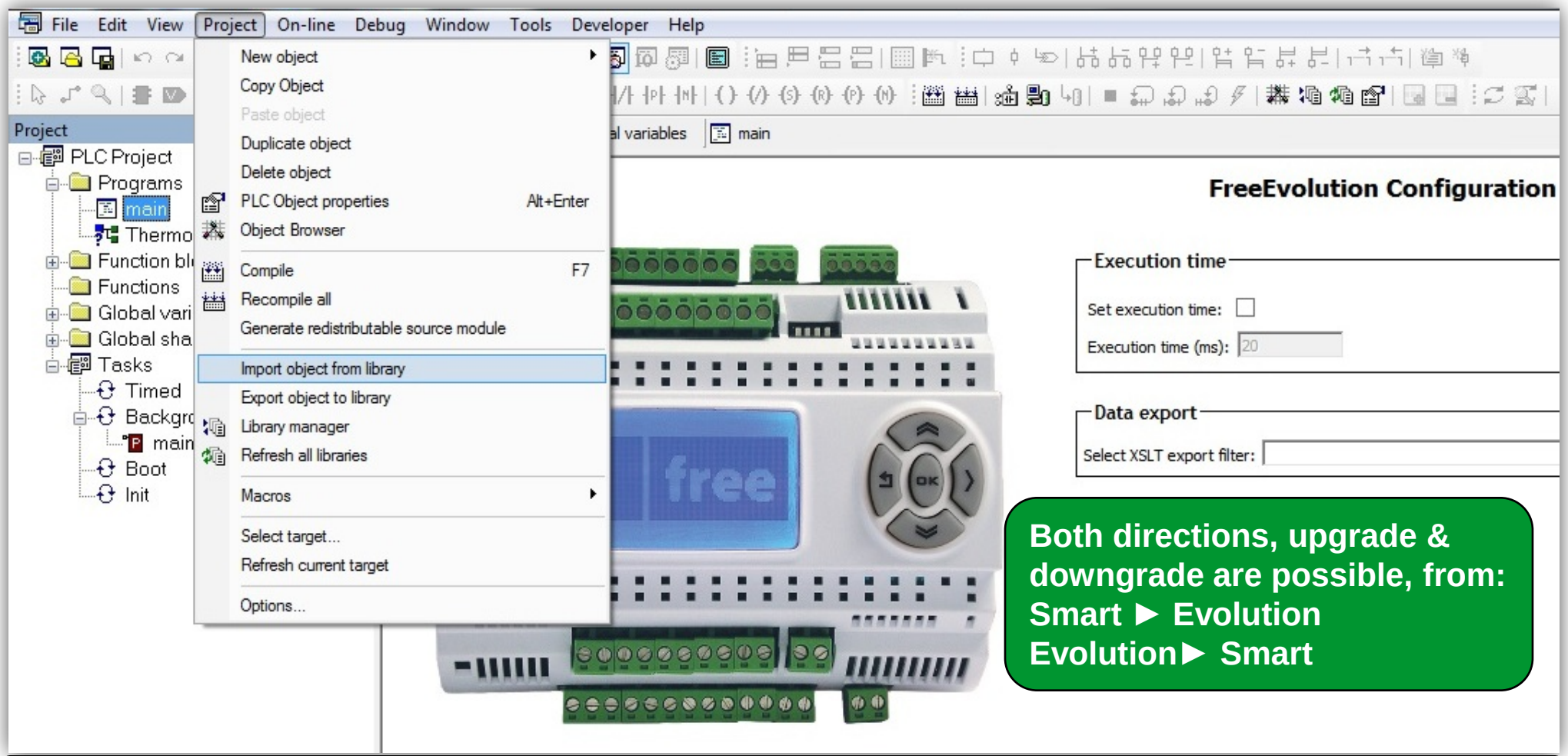


Converted project from Smart to Evolution



Conversion between Evolution to Smart is not fully supported (all resources are deleted).

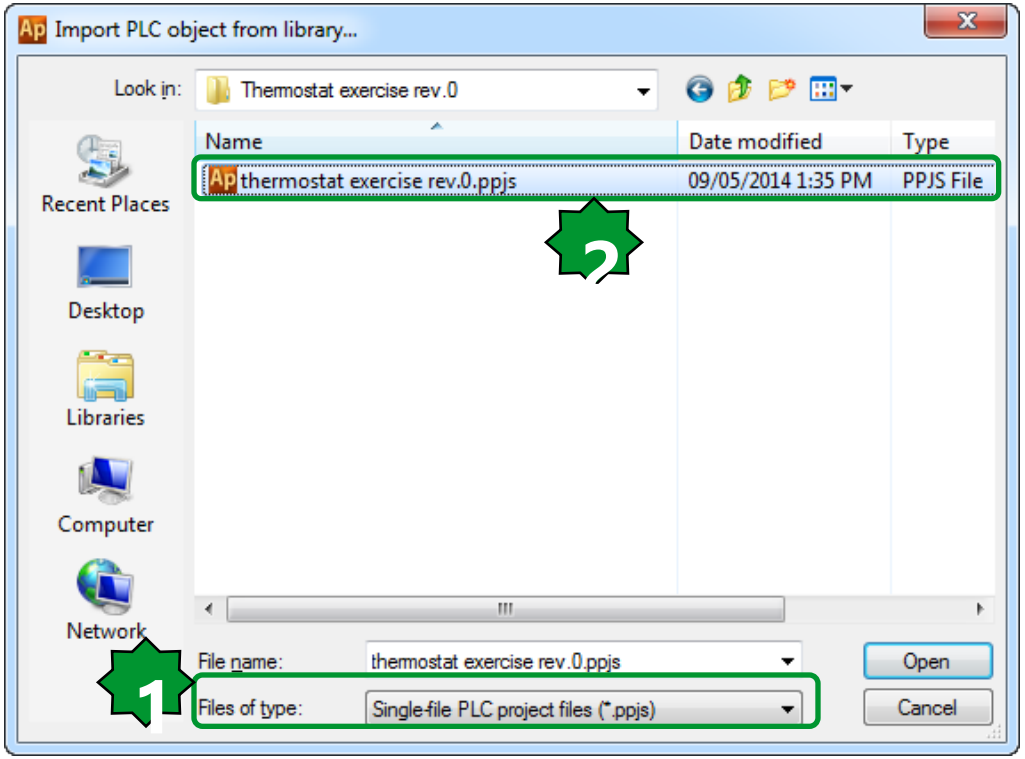
Import Objects from library (or Project)

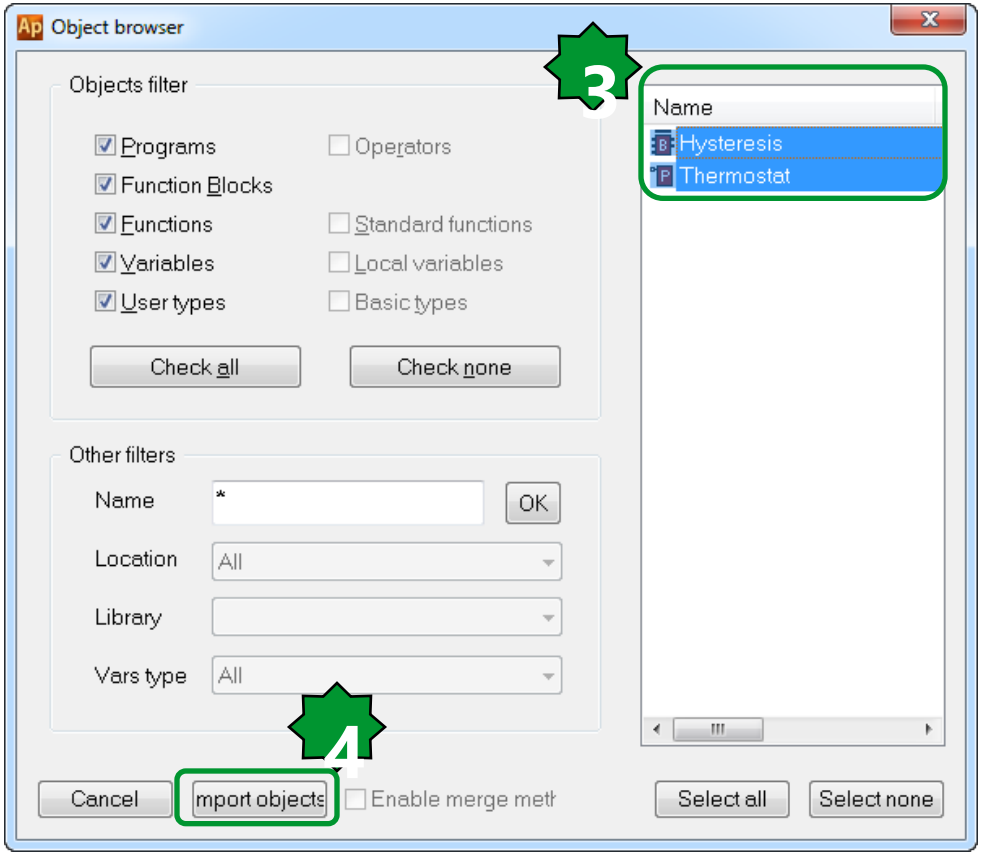


Both directions, upgrade & downgrade are possible, from:
Smart ► Evolution
Evolution► Smart

It allows also to import programs, FB, functions from other projects regardless the related target.

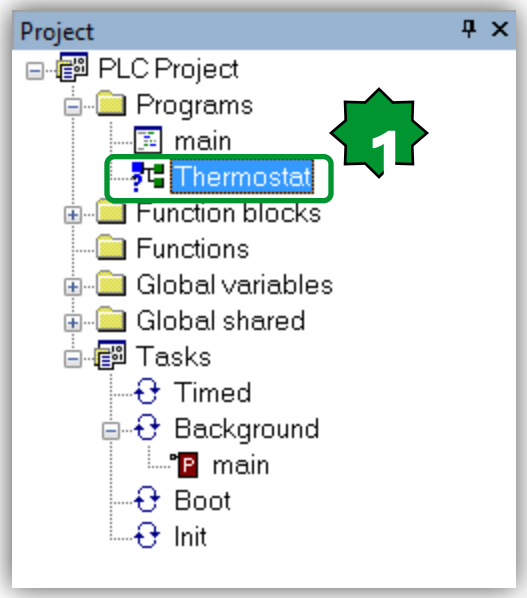
Import Objects from Project...

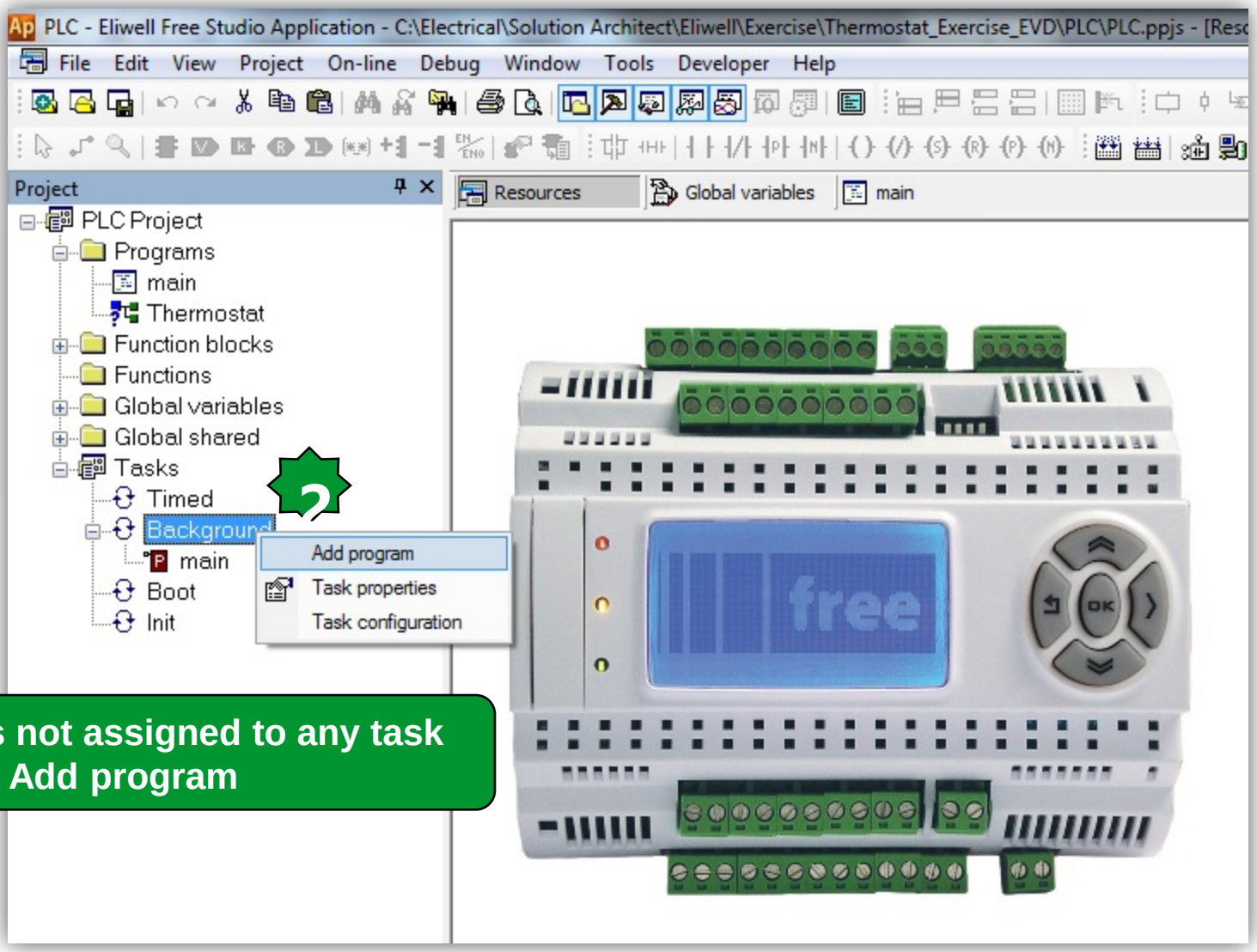




1. Select *.ppjs file type
2. Select desired project (Smart)
3. Select desired program & FBD
4. Import Objects

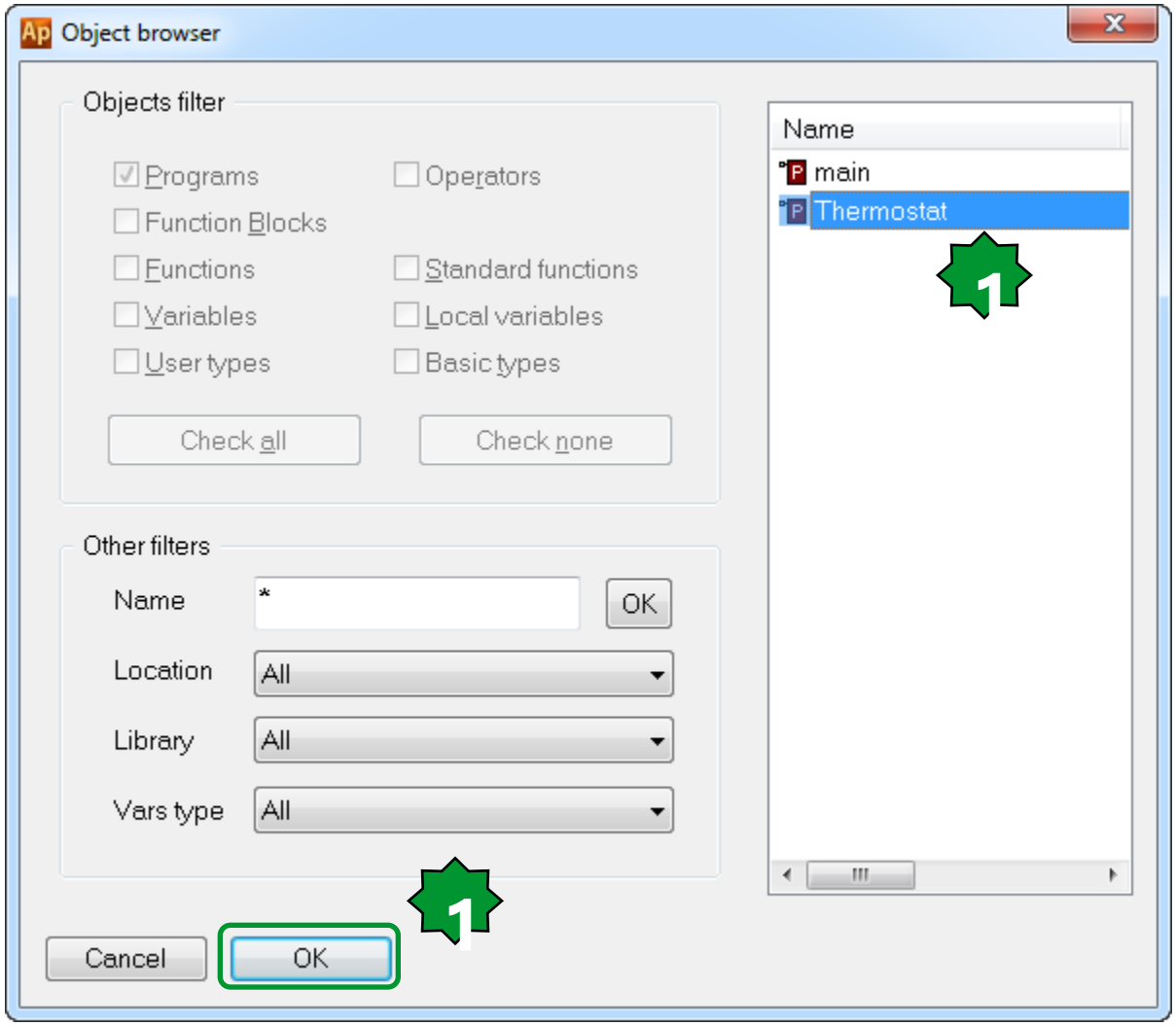
...Assign to Task (in case of program)

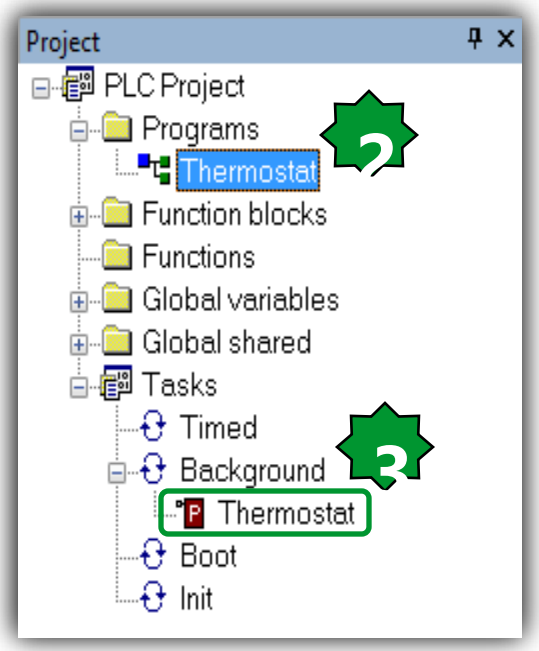




1. ? Means that program is not assigned to any task
2. Tasks ► Background ► Add program

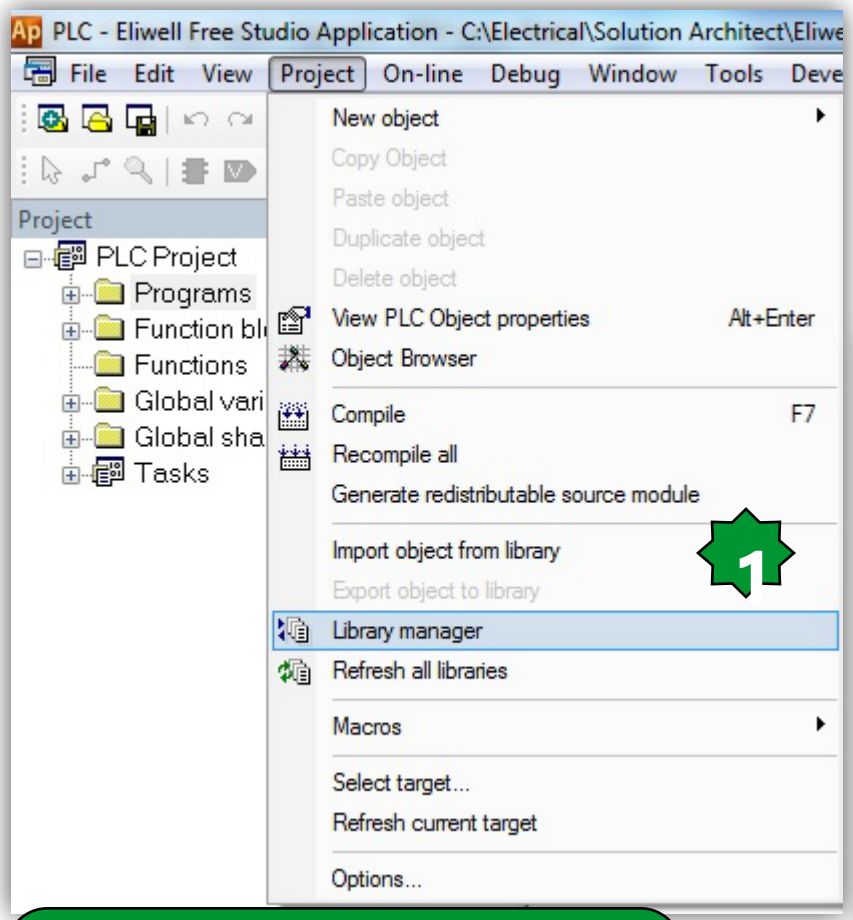
Assigning imported program to the task

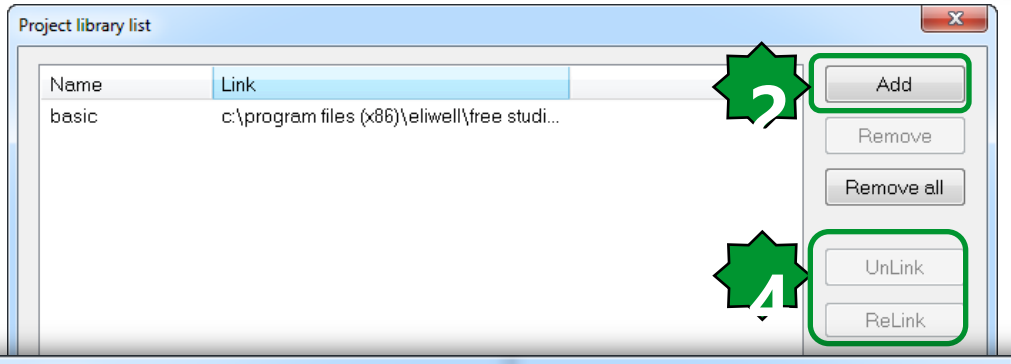


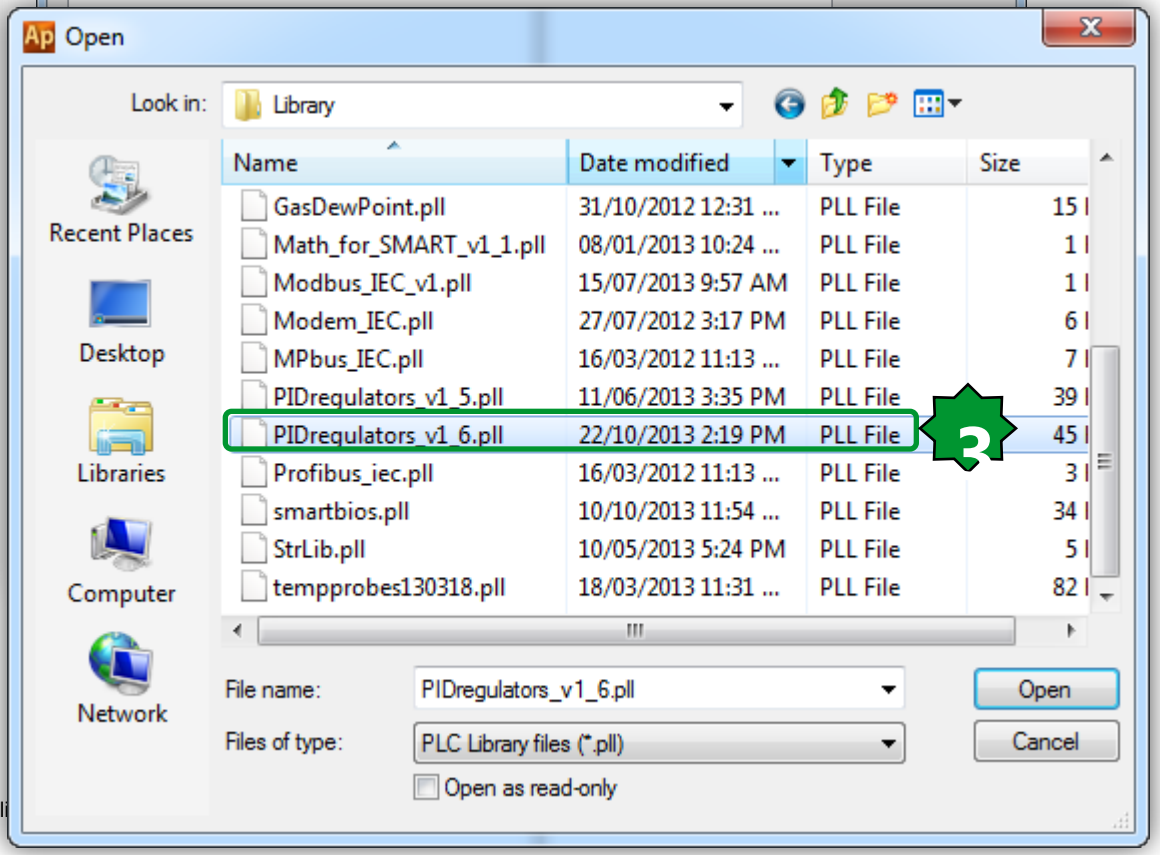


1. Select the desired program name ► OK
2. The ? Disapeares in prgrams
3. It will assign to to desired task (delete non-required programmes)

Link libraries...

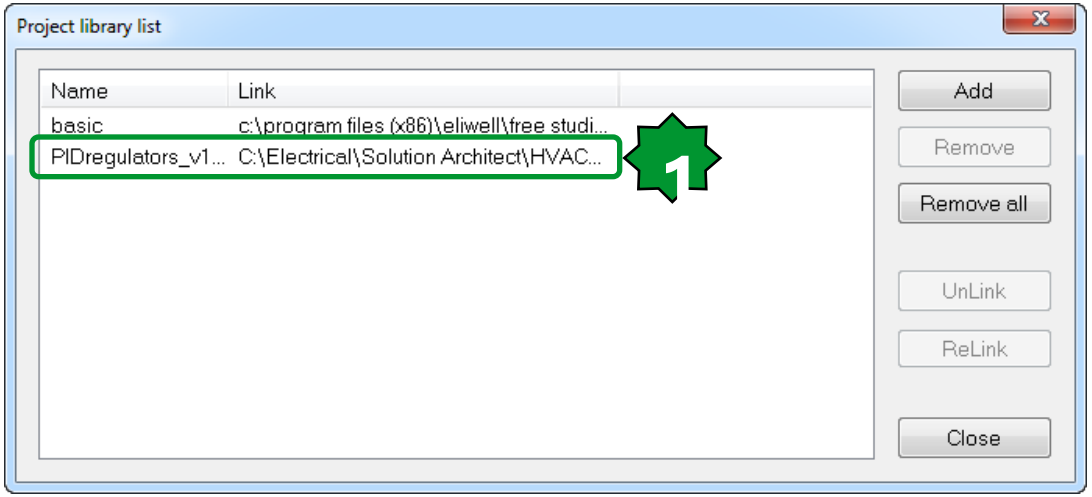


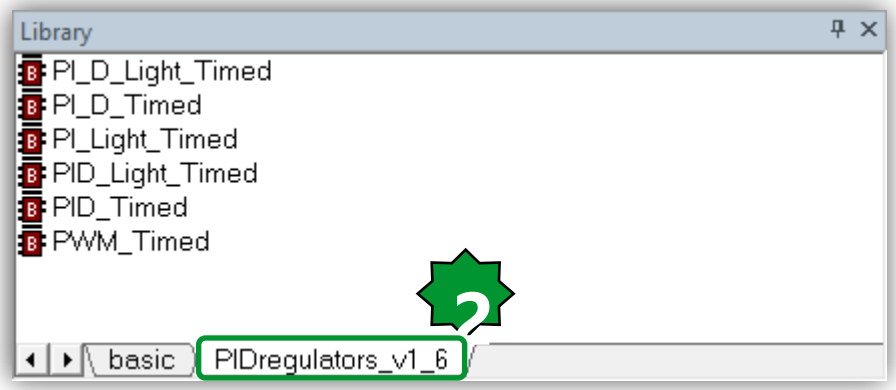




1. Project ► Library Manager
2. Project Library list
3. add/remove
4. Unlink/Relink

...Link Libraries

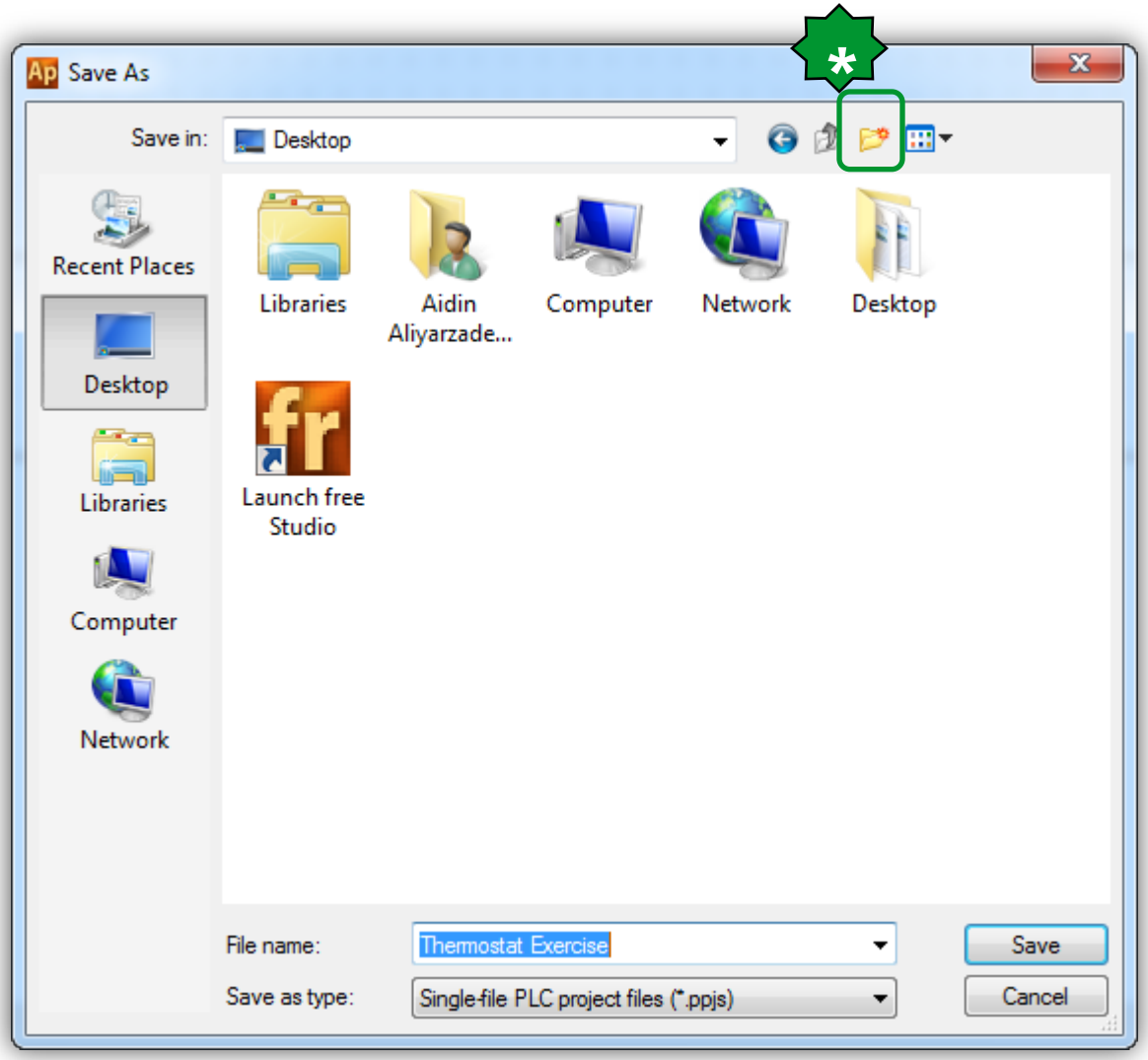




Save a project as:

File ► save a project as :

* Create a folder for the project before saving



Chapter 9

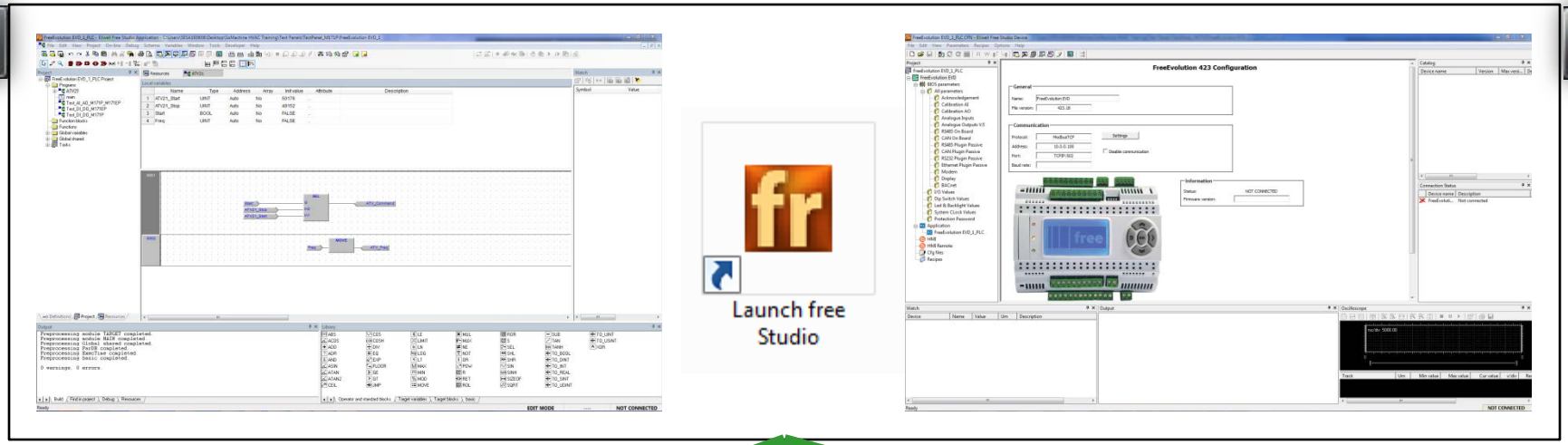
Methodology:

Goal:

Navigation between the SW, application, device, connection & familiarizing with their abilities

Free studio/unique programming software

In Unique software suite for Smart and Evolution



Smart

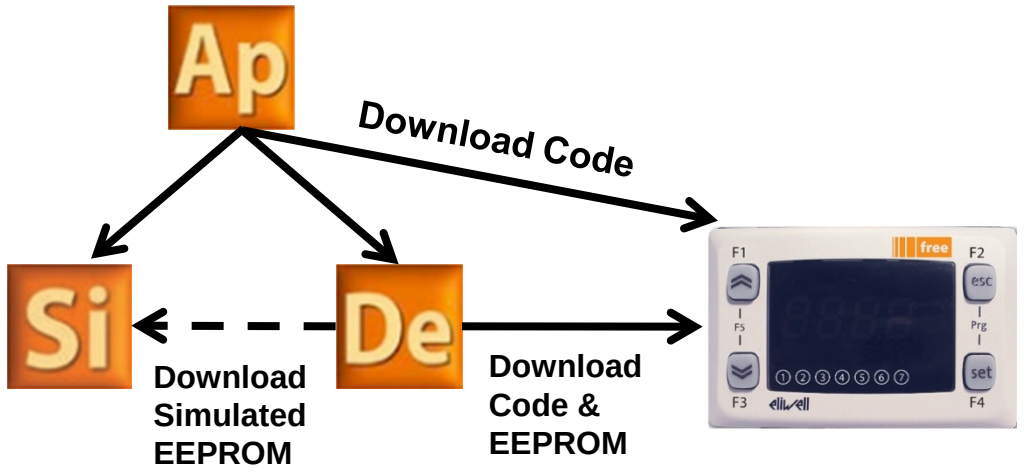
Evolution

Software suite presentation

M171P + M171O		M171P	
icon (link)	description	icon (link)	description
Ap	Application development tool for Smart & Evolution	Co	Connection development tool for Evolution
De	Device development tool for Smart & Evolution	UI	User Interface development tool for Evolution
Si	Simulation development tool for Smart & Evolution		

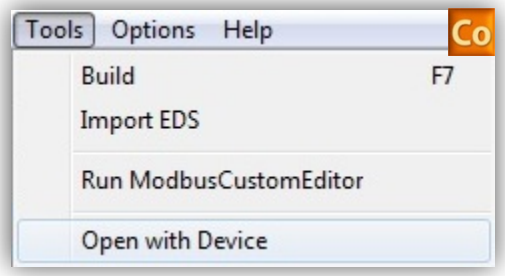
Smart Project Architecture

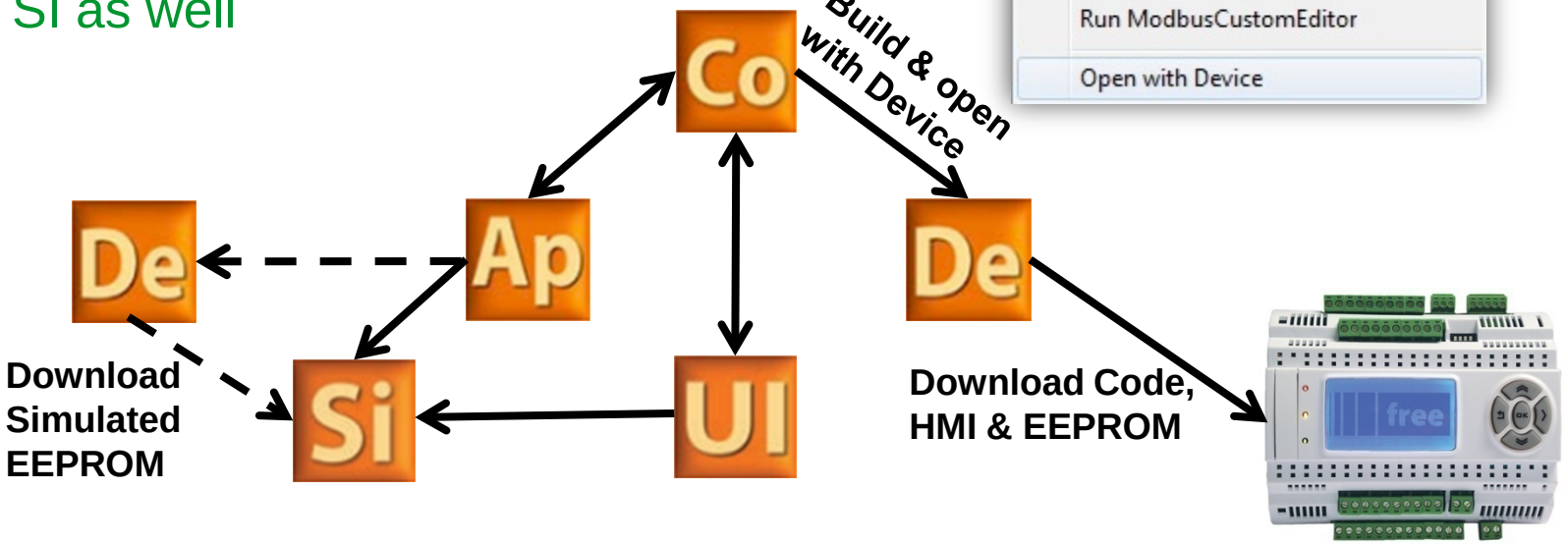
- Application is the programming starting point.
- Device is used to download the overall compiled project and it is the only tool able to write EEPROM parameters.
- From Application it will always be possible to open Device directly without having to launch the program using the FREE Studio icon.



Evolution Project Architecture

- Connection is the entry point for all development activities.
- Device is used to download the overall compiled project and it is the only tool able to write EEPROM parameters and the master connectivity configuration
- Application can download only the algorithm and the EEPROM parameters and Status Variable definition.
- UI can download only the HMI to EVD
- UI can open SI as well





Chapter 10

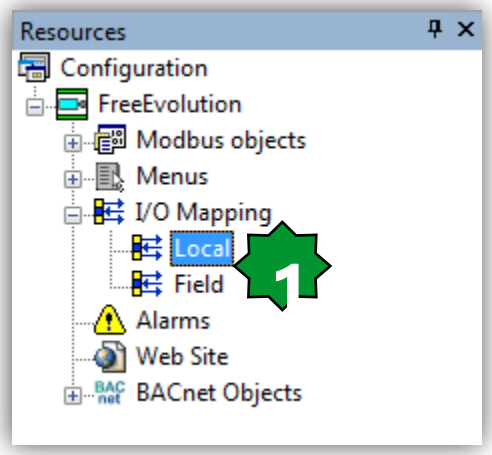
Fan Management

Goal:

Manage 3 fans base on analogue input configuration

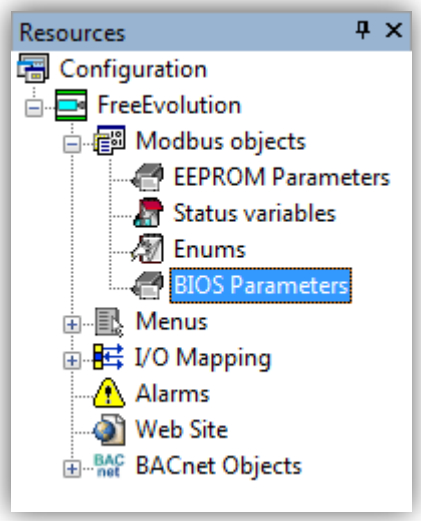
Physical I/O assignment

Fan Management Function Description:
Fan Management enable by DIL1=True
If AI3P > 3.3 V => Fan1=ON
If 3.3 V < AI3P < 6.6 V => Fan1 & 2 = ON
If AI3P > 6.6 V => Fan1,2 & 3 =ON
If AI3P = - 32768 => Alarm=ON & Fans=False
Monitoring AI3P by the gauge that is connected to the AO1P (0-10 V).



#	Name	Variable	Type	
1	AIL1	NTC_Probe	INT	AIL1 analogue input
2	AIL2		INT	AIL2 analogue input
3	AIL3	Potentiometer_AI3P	INT	AIL3 analogue input
4	AIL4		INT	AIL4 analogue input
5	AIL5		INT	AIL5 analogue input
6	AIL6		INT	AIL6 analogue input
7	DIL1	Fan_Start_Stop	BOOL	DIL1 digital input
8	DIL2		BOOL	DIL2 digital input
9	DIL3		BOOL	DIL3 digital input
10	DIL4		BOOL	DIL4 digital input
11	DIL5		BOOL	DIL5 digital input
12	DIL6		BOOL	DIL6 digital input
13	DIL7		BOOL	DIL7 digital input
14	DIL8		BOOL	DIL8 digital input
15	DOL1		BOOL	DOL1 digital output
16	DOL2		BOOL	DOL2 digital output
17	DOL3	Output_Cooling	BOOL	DOL3 digital output
18	DOL4	Alarm	BOOL	DOL4 digital output
19	DOL5	Fan1	BOOL	DOL5 digital output
20	DOL6	Fan2	BOOL	DOL6 digital output
21	DOL7	Fan3	BOOL	DOL7 digital output
22	AOL1	Guage_AO1P	INT	AOL1 analogue output
23	AOL2		INT	AOL2 analogue output
24	AOL3		INT	AOL3 analogue output
25	AOL4		INT	AOL4 analogue output
26	AOL5		INT	AOL5 analogue output
27	FDI_counter		UDINT	FDI Input counter
28	FDI_frequency		UDINT	FDI Input frequency
29	FDI_value		BOOL	FDI Input value
30	FDI_reset_counter		BOOL	FDI reset input counter value

BIOS Parameters/AI* Configuration

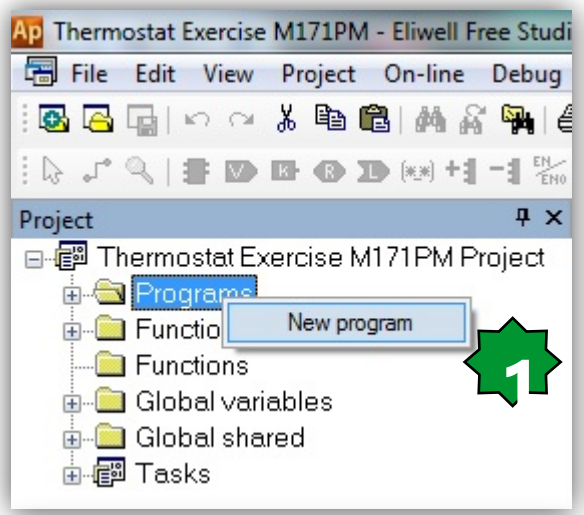


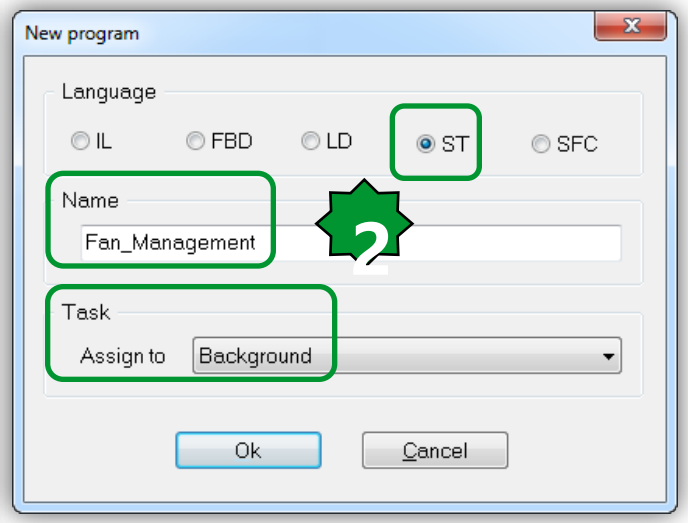
FreeEvolution BIOS Parameters

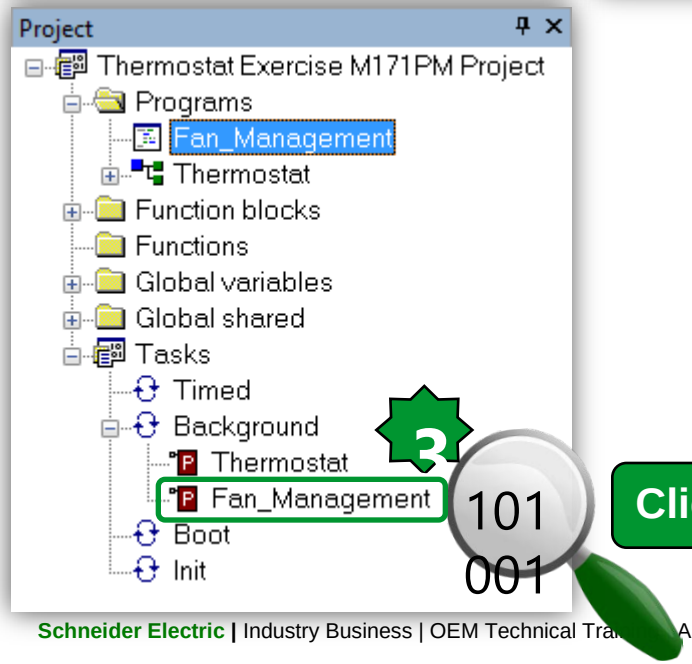
Add Remove

#	Name	Default value	Description
1	Cfg_AI3	4=0÷10V	Type of analogue input AI3
2	FullScaleMin_AI3	0	First value analogue input AI3 scale
3	FullScaleMax_AI3	10000	Last value analogue input AI3 scale
4	Cfg_AI1	2=NTC(103AT)	Type of analogue input AI1
5	Cfg_AO1_AO5	2=Voltage modulation	Type of analogue output AO1/AO5

New program creation





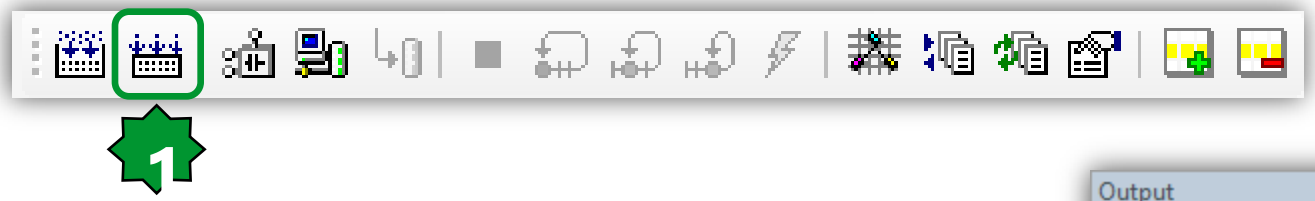


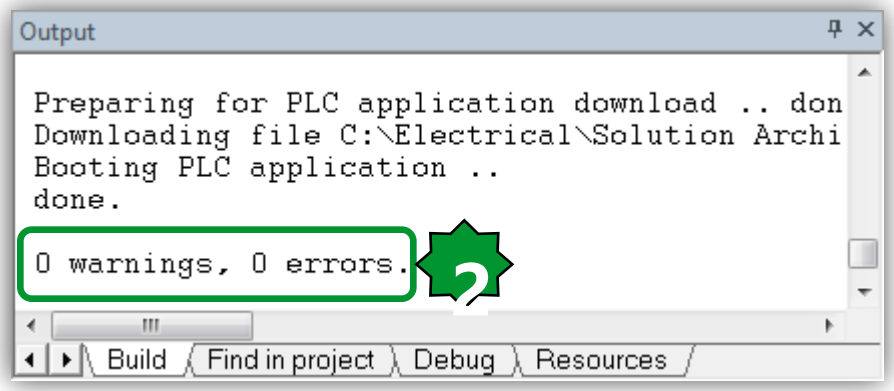
Click on the magnifying-glass to write the codes.

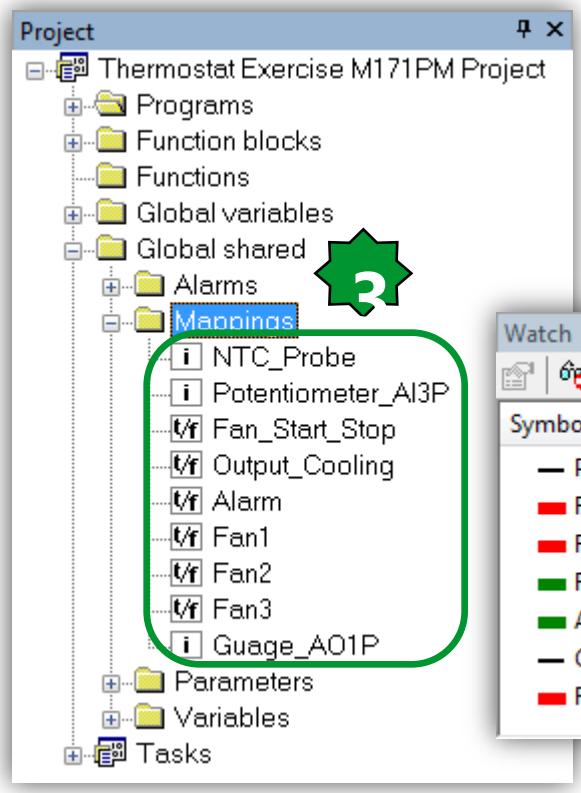
```
0001  (* Enable/Disable of Fan management by Start/Stop the digital Input (DIL1) *)
0002  If Fan_Start_Stop = False then
0003      Fan3:= FALSE;
0004      Fan2:= FALSE;
0005      Fan1:= FALSE;
0006  end_if;
0007
0008
0009  (* If the AI3P value is less than 3.33 volts, then Fan1=ON *)
0010
0011  if Potentiometer_AI3P <= 333 and Fan_Start_Stop = True then
0012     Fan1:= True;
0013     else Fan1 := FALSE;
0014  End_If;
0015
0016  (* If the AI3P value is less or equal than 6.66 volts or greater than 3.33 Volts, then Fan1=ON & Fan2=ON*)
0017
0018  if Potentiometer_AI3P <= 666 and Potentiometer_AI3P > 333 and Fan_Start_Stop = True then;
0019     Fan2:= True;
0020     Fan1:= True;
0021     else Fan2 := FALSE;
0022    End_If;
0023
0024  (* If the AI3P value is grater than 6.66 volts, then Fan1=ON & Fan2=ON & Fan3=ON*)
0025
0026  if Potentiometer_AI3P >666 and Fan_Start_Stop = True then;
0027     Fan3:= True;
0028     Fan2:= True;
0029     Fan1:= True;
0030     else Fan3 := FALSE;
0031     End_If;
0032
0033  (* AI3P disconnection detector *)
0034
0035  if Potentiometer_AI3P = -32768 then
0036      Alarm:= TRUE;
0037      Fan3:= FALSE;
0038      Fan2:= FALSE;
0039      Fan1:= FALSE;
0040      else Alarm := FALSE;
0041  end_if;
0042
0043  (* Monitoring AI3P by AO1P via the 0-10 Volts gauge *)
0044
0045  Guage_AO1P := Potentiometer_AI3P;
```

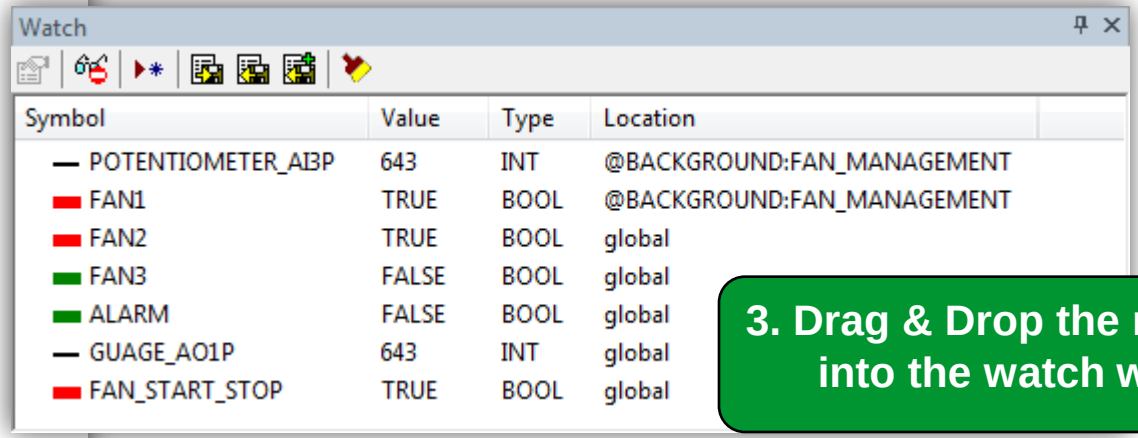
Fan Management,
Enable/Disable Outputs,
AI monitoring by AO (0-10 V),
AI disconnection detection codes.

Debugging/Watch



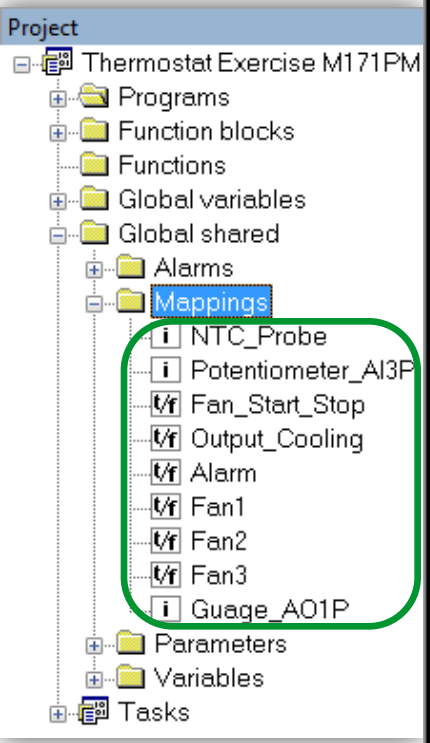


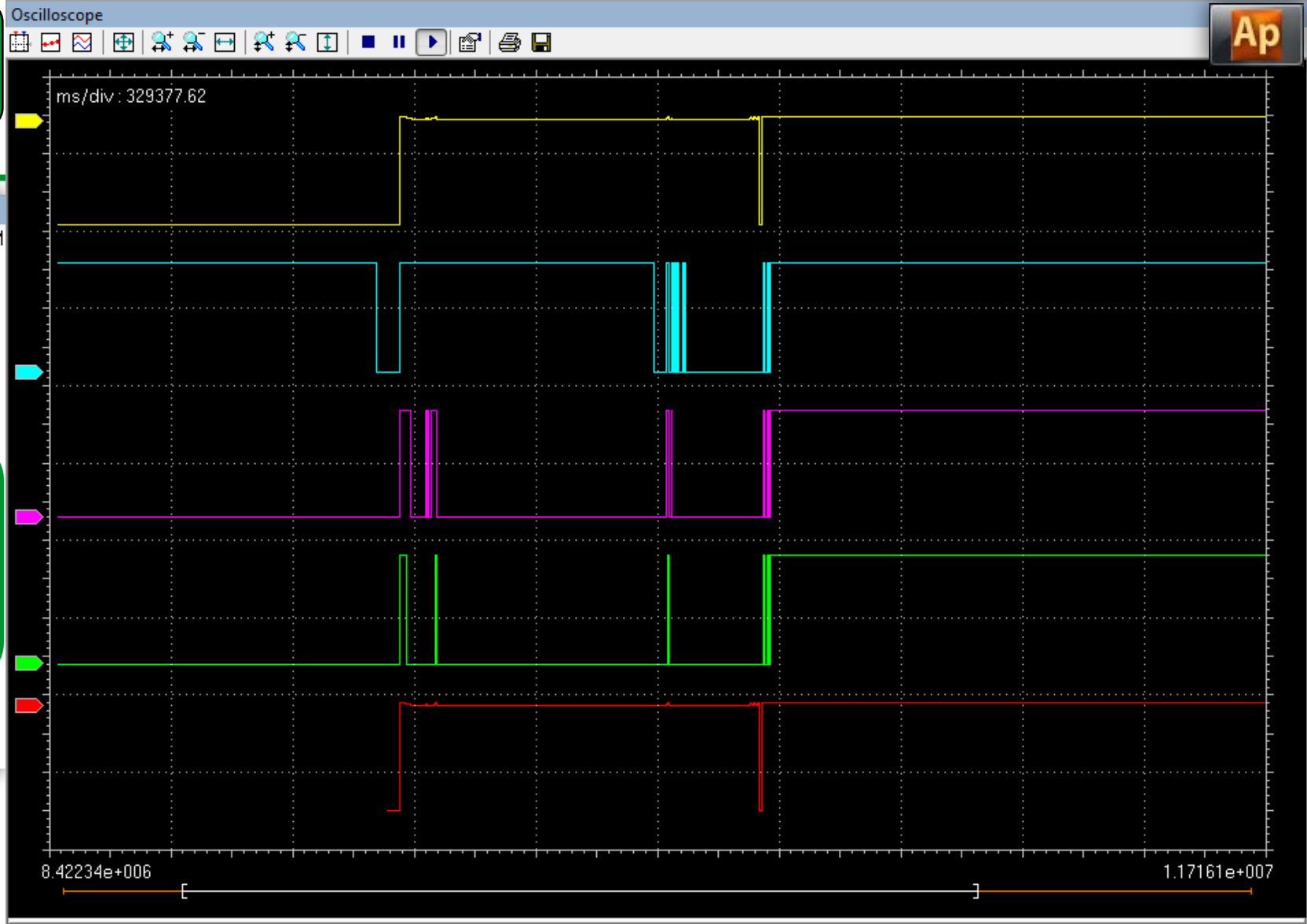




3. Drag & Drop the mappings into the watch window.

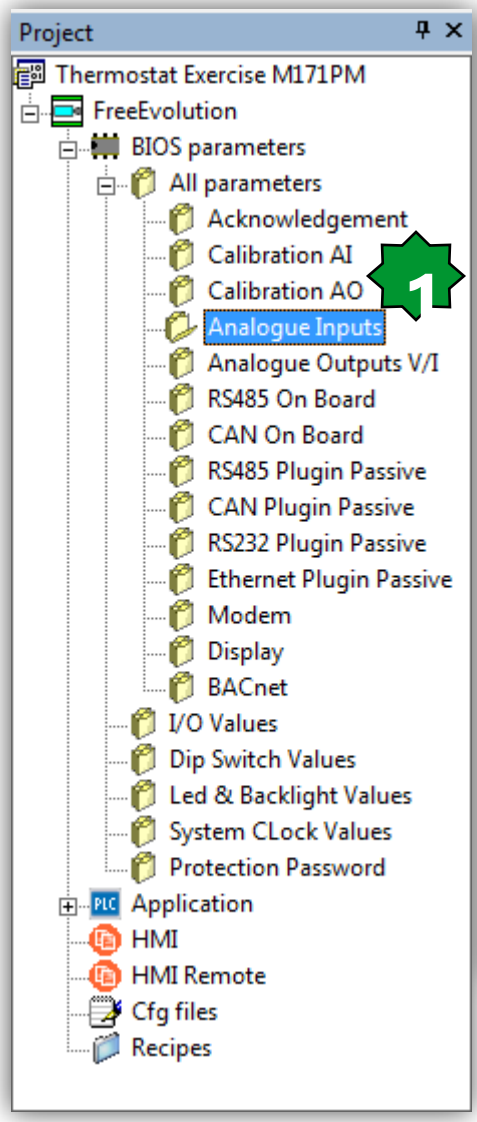
Debugging /Oscilloscope:





Track	Um	Min value	Max value	Cur value	v/div	Red ...	Blue...	Horz cursor	Note
@BACKGROUND:FAN_MANAGEMENT.POTENTIOMETER_AI3P		-32768.000	1048.000	643.000	2415...	...	...	...	@BACKGROUND:...
@BACKGROUND:FAN_MANAGEMENT.FAN1		0.000	1.000	1.000	0.714...	...	...	...	@BACKGROUND:...
@BACKGROUND:FAN_MANAGEMENT.FAN2		0.000	1.000	1.000	0.714...	...	...	...	@BACKGROUND:...
@BACKGROUND:FAN_MANAGEMENT.FAN3		0.000	1.000	0.000	0.714...	...	...	...	@BACKGROUND:...
@BACKGROUND:FAN_MANAGEMENT.GUAGE_AO1P		-32768.000	1022.000	643.000	2413...	...	...	...	@BACKGROUND:...

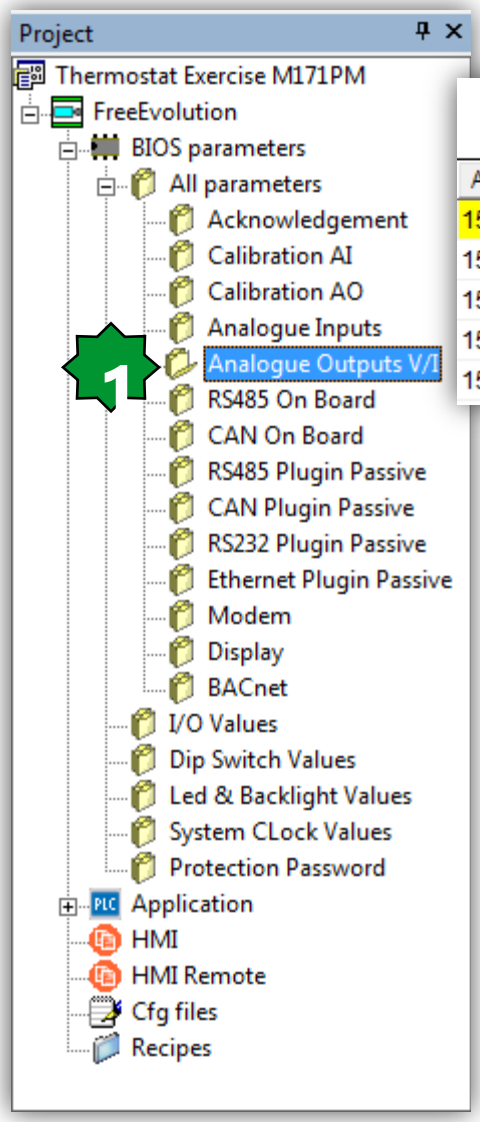
Analogue Input Configuration



Analogue Inputs

Address	Name	Value	Um	Default	Min	Max	Description
15725	Temp_UM	0=°C	num	0=°C	0	1	Unit of temperature measurement
15726	Cfg_AI1	2=NTC(103AT)	num	2=NTC(103AT)	0	2	Type of analogue input AI1
15727	Cfg_AI2	2=NTC(103AT)	num	2=NTC(103AT)	0	2	Type of analogue input AI2
15728	Cfg_AI3	4=0÷10V	num	3=4÷20mA	0	8	Type of analogue input AI3
15729	Cfg_AI4	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI4
15730	Cfg_AI5	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI5
15731	Cfg_AI6	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI6
15736	FullScaleMin_AI3	0	digit	0	-9999	9999	First value analogue input AI3 scale
15737	FullScaleMax_AI3	1000	digit	1000	-9999	9999	Last value analogue input AI3 scale
15738	FullScaleMin_AI4	0	digit	0	-9999	9999	First value analogue input AI4 scale
15739	FullScaleMax_AI4	1000	digit	1000	-9999	9999	Last value analogue input AI4 scale
15740	FullScaleMin_AI5	0	digit	0	-9999	9999	First value analogue input AI5 scale
15741	FullScaleMax_AI5	1000	digit	1000	-9999	9999	Last value analogue input AI5 scale
15742	FullScaleMin_AI6	0	digit	0	-9999	9999	First value analogue input AI6 scale
15743	FullScaleMax_AI6	1000	digit	1000	-9999	9999	Last value analogue input AI6 scale
15748	Calibration_AI1	0	°C/10,°F/10	0	-180	180	Analogue input AI1 differential
15749	Calibration_AI2	0	°C/10,°F/10	0	-180	180	Analogue input AI2 differential
15750	Calibration_AI3	0	digit	0	-1000	1000	Analogue input AI3 differential
15751	Calibration_AI4	0	digit	0	-1000	1000	Analogue input AI4 differential
15752	Calibration_AI5	0	digit	0	-1000	1000	Analogue input AI5 differential
15753	Calibration_AI6	0	digit	0	-1000	1000	Analogue input AI6 differential

Analogue Output Configuration



Analogue Outputs V/I

2

Address	Name	Value	Um	Default	Min	Max	Description
15758	Cfg_AO1_AO5	2=Voltage modulatio	num	0=Current modulation	0	2	Type of analogue output AO1/AO5
15759	Cfg_AO2	0=Current modulation	num	0=Current modulation	0	2	Type of analogue output AO2
15760	Cfg_AO3	0=Current modulation	num	0=Current modulation	0	2	Type of analogue output AO3
15761	Cfg_AO4	0=Current modulation	num	0=Current modulation	0	2	Type of analogue output AO4
15762	SubCfg_AO5	0	num	0	0	1	Subtype of analogue output AO5

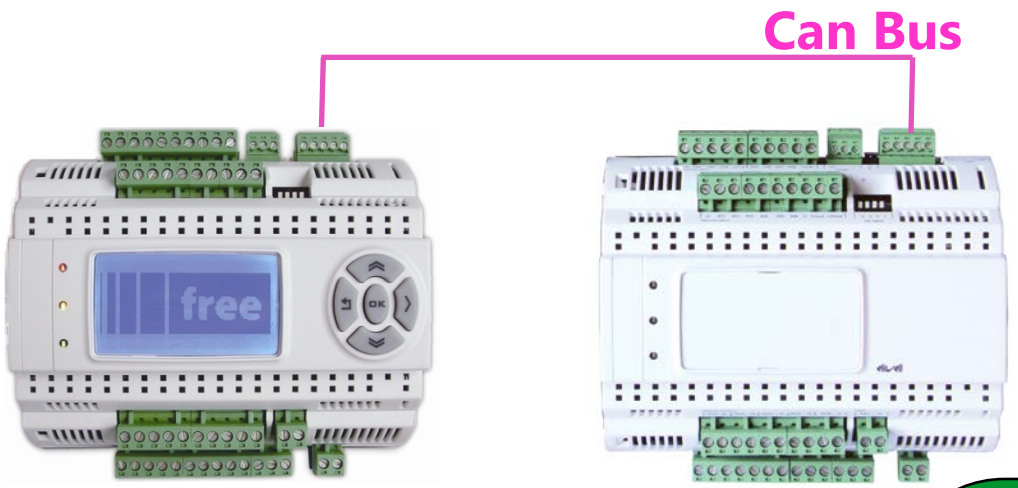
Chapter 11

Network

Goal:

Expansion connection to the base unit via CAN BUS

Evolution Networking Exercise



EVD1

EVE1

Address 2
➔
Dip Switch 1 = ON

Goals:

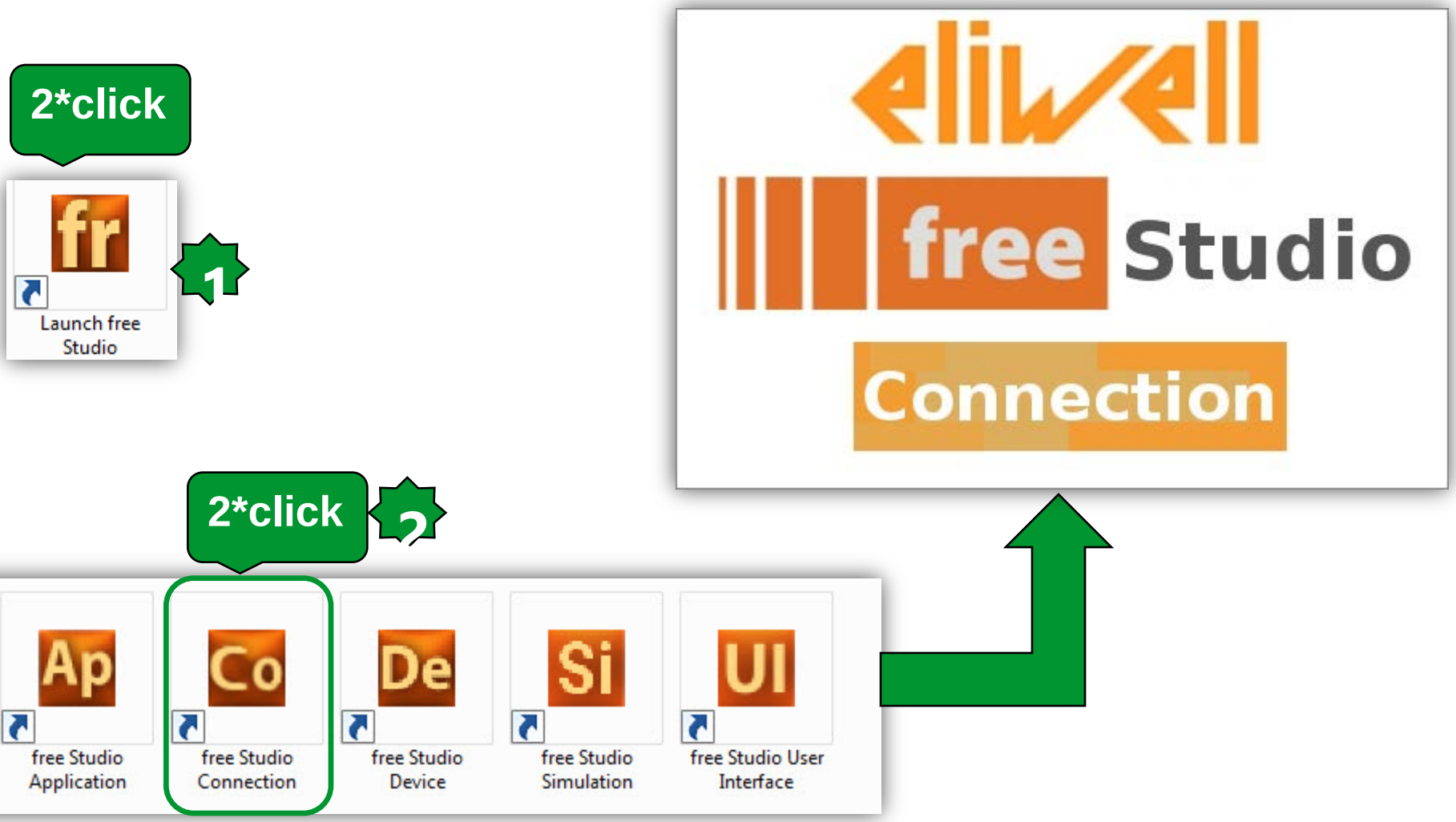
- **Connect an expansion EVE1 to Can Bus,**
- **Configure the AI3 of the EVD as 0-10V input**
- **Read AI3 and DI1 of the EVD.**
- **Write AO1 and DO5, 6 & 7 of the EVD.**

Note: Connect GS,H,L of EVD to GS,H,L of EVE respectively

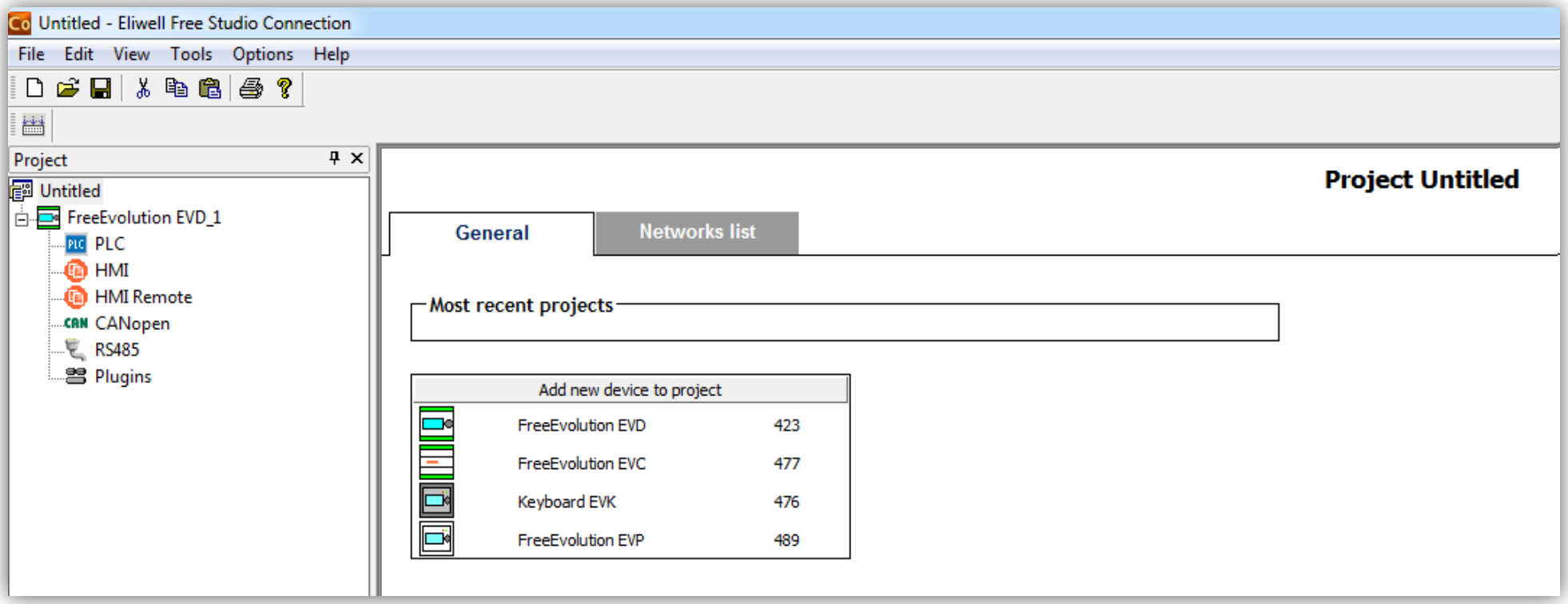
DO NOT CONNECT together POWER OUT, POWER OUT is just for powering the remote EVK1000 display.

NOTE: Leave the CanOpen end resistor jumpers only to the endline Devices, in this case EVD1 and EVE1

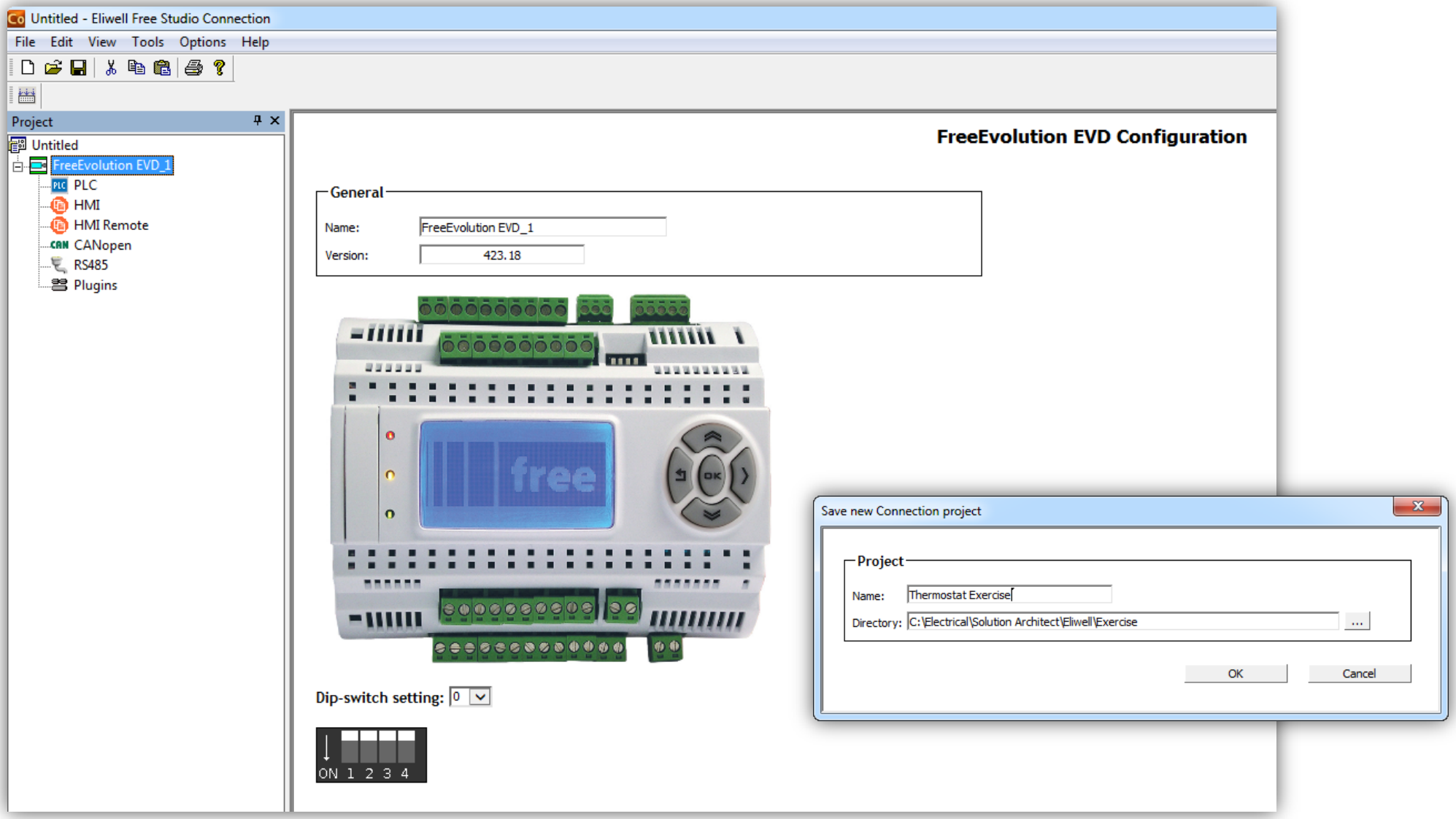
Creating New project‘s connection



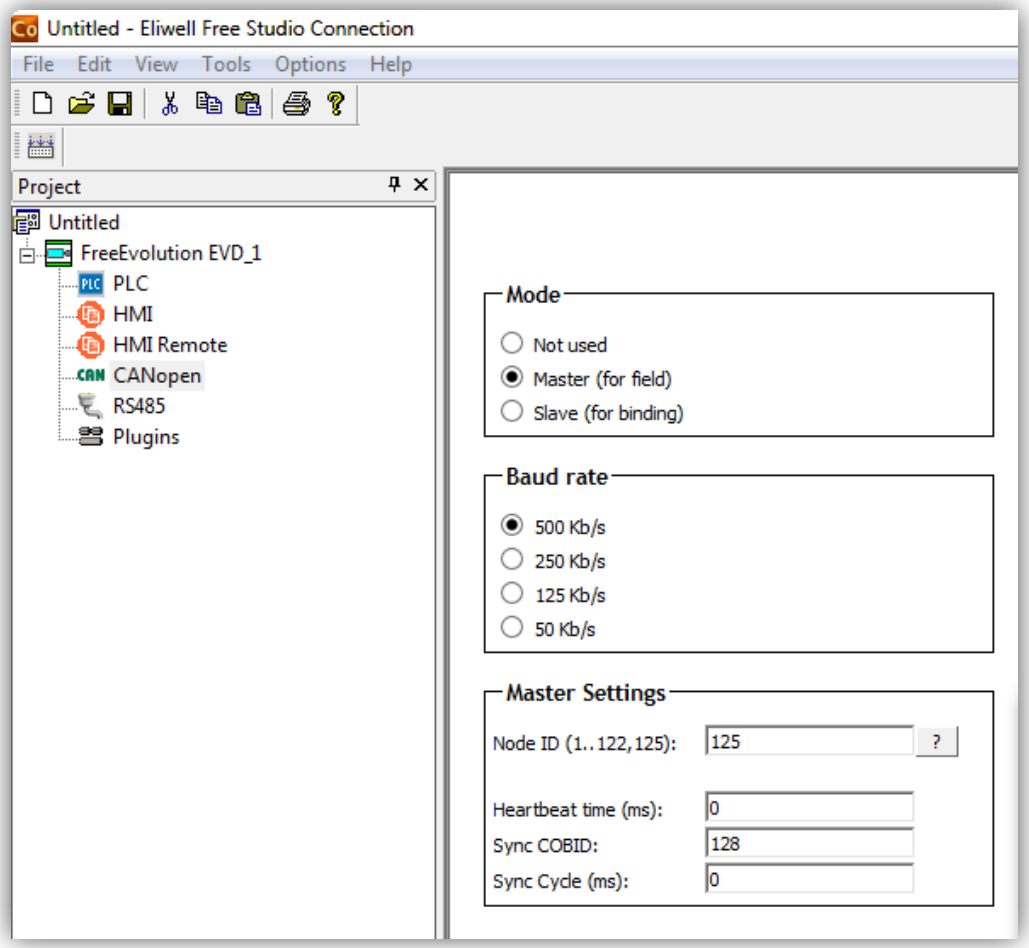
Define the Project Architecture



FreeEvolution EVD configuration

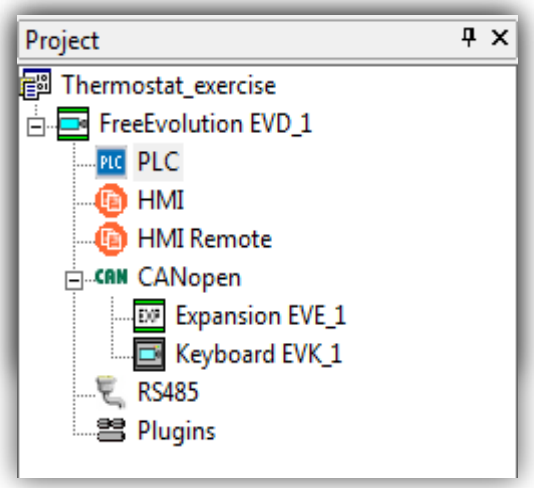


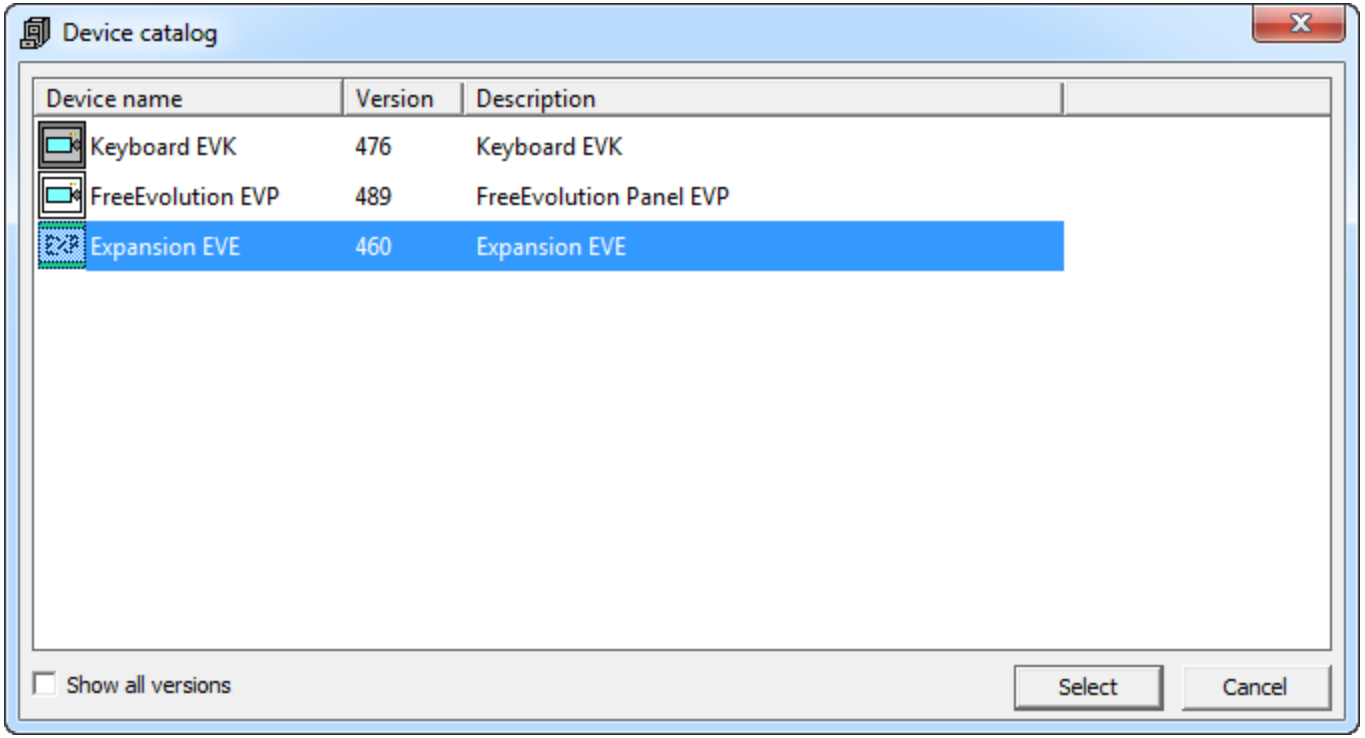
CANopen configuration



- **The CanOpen address of EVD is 125, it will written in the CONNEC.PAR file, CAN On Board parameters are not valid if the Evolution is Master on CAN.**
- **EVE must be set at the speed defined here (if changed the devices must be restarted, Factory default is 500Kb/s)**

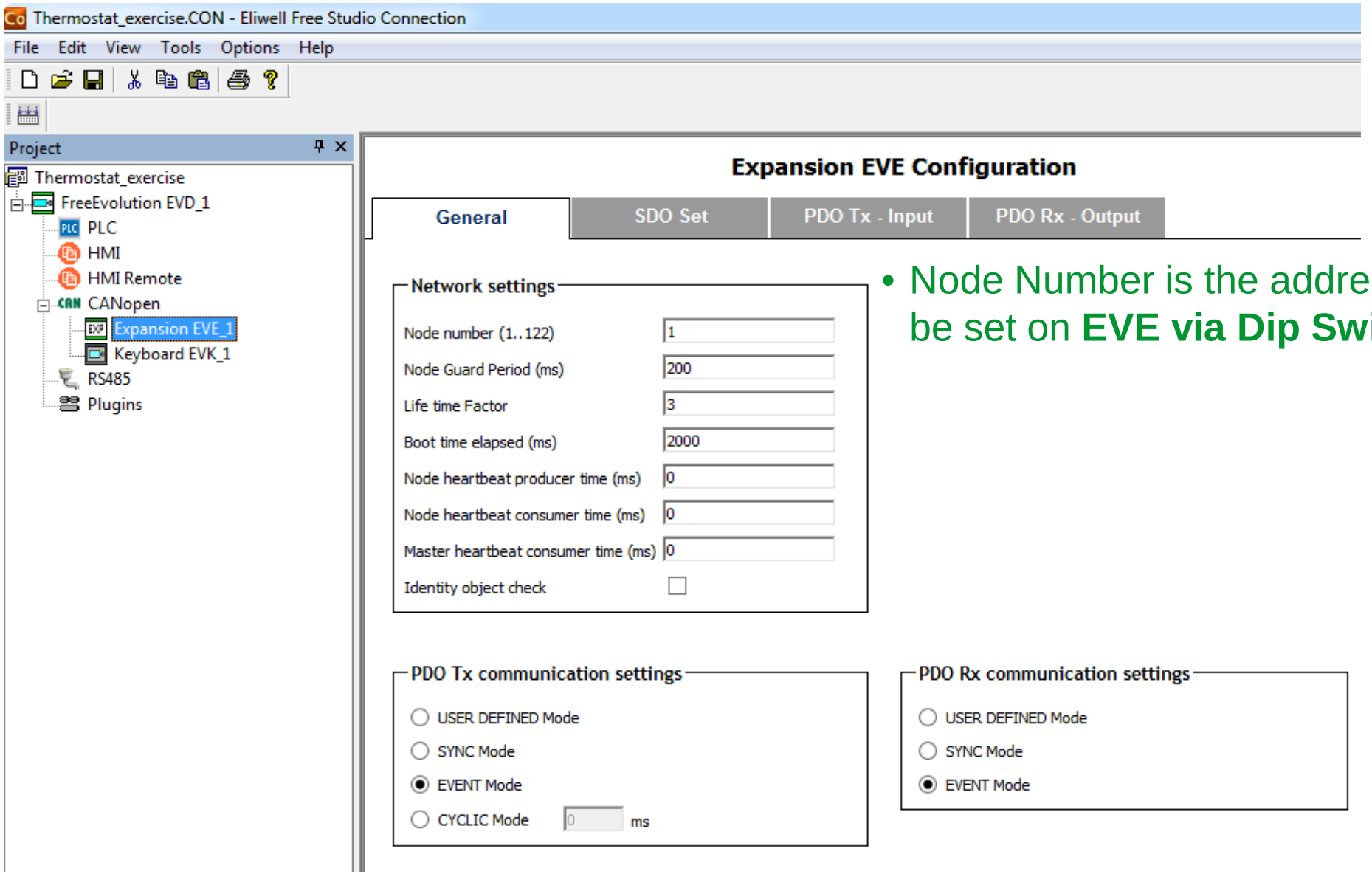
Add an expansion





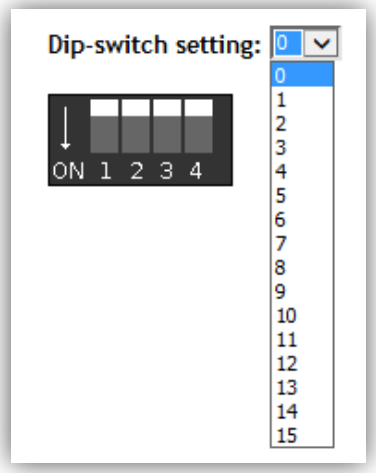
Drag & drop from device catalogue to the CANopen
Or
CANopen►Add►Device catalogue►Select the target

Expansion EVE configuration

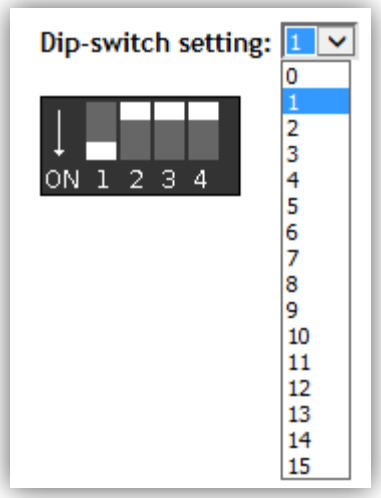


- Node Number is the address to be set on **EVE via Dip Switch.**

Dip switch setting ► EVE Address defining



Address=Dip switch +1
1=0+1



Address=Dip switch +1
2=1+1

It is possible to connect up to 12 expansion+2 keyboard

Expansion EVE configuration

- Pressing Add you can eventually define the I/O configuration of EVE that EVD will send at powerup
- For example: EVE1 AI3 is set as 0-10V.

Expansion EVE Configuration

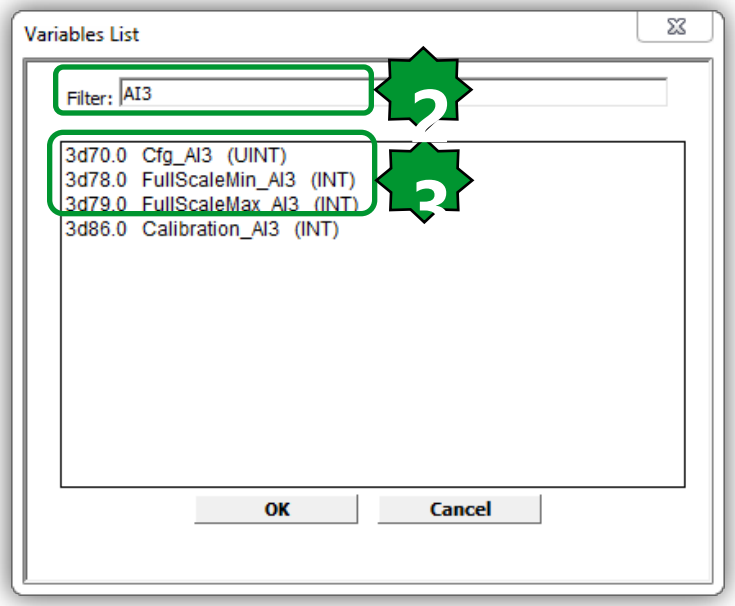
1

General | SDO Set | PDO Tx - Input | PDO Rx - Output

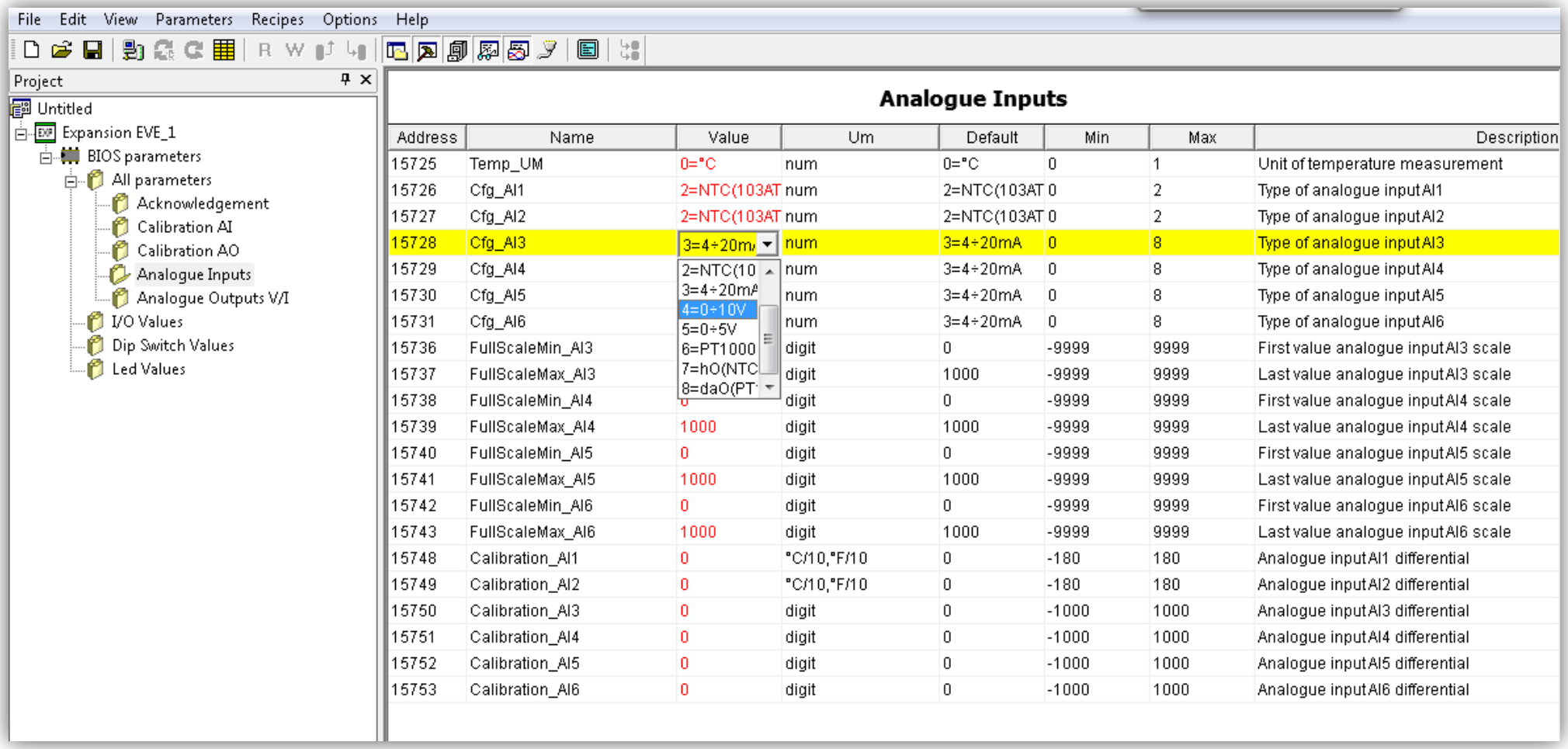
Add | Remove

#	Label	Index	SubIndex	Type	Value	Timeout
1	COB-ID	1404	1	UDINT	$NODEID+0x40000500	1000
2	COB-ID	1804	1	UDINT	$NODEID+0x40000480	1000
3	Transmission Type	1800	2	USINT	255	100
4	Event Timer	1800	5	UINT	0	100
5	Transmission Type	1801	2	USINT	255	100
6	Event Timer	1801	5	UINT	0	100
7	Transmission Type	1802	2	USINT	255	100
8	Event Timer	1802	5	UINT	0	100
9	Transmission Type	1804	2	USINT	255	100
10	Event Timer	1804	5	UINT	0	100
11	Transmission Type	1400	2	USINT	255	100
12	Transmission Type	1401	2	USINT	255	100
13	Transmission Type	1402	2	USINT	255	100
14	Transmission Type	1404	2	USINT	255	100
15	Cfg_AI3	3d70	0	UINT	4	100
16	FullScaleMin_AI3	3d78	0	INT	0	100
17	FullScaleMax_AI3	3d79	0	INT	1000	100

4



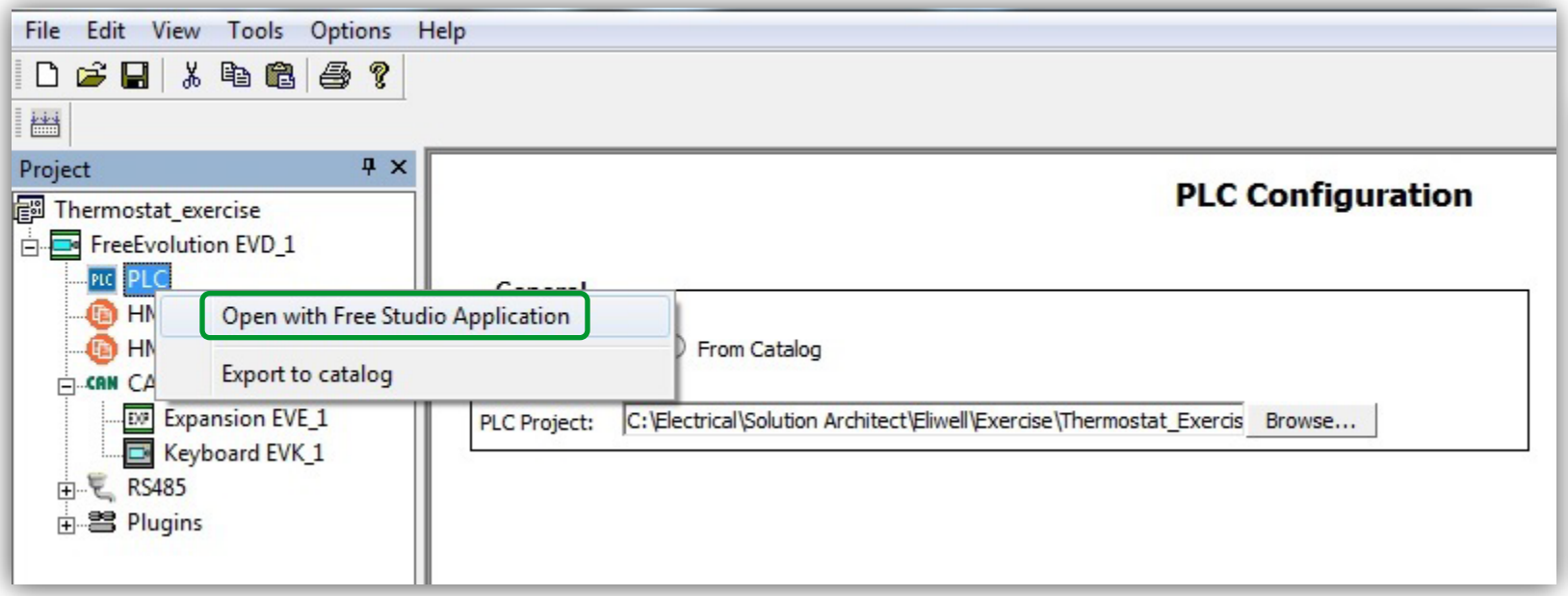
Check FS Device for parameter enumeration

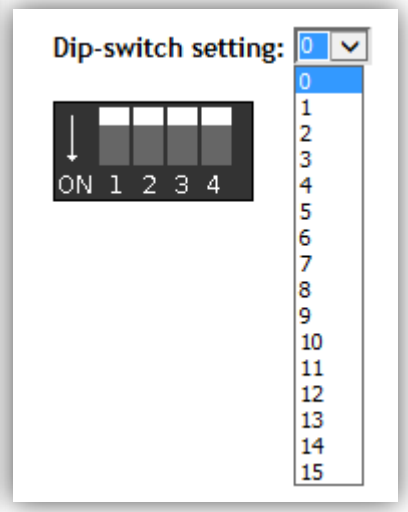


Address	Name	Value	Um	Default	Min	Max	Description
15725	Temp_UM	0=°C	num	0=°C	0	1	Unit of temperature measurement
15726	Cfg_AI1	2=NTC(103AT	num	2=NTC(103AT	0	2	Type of analogue input AI1
15727	Cfg_AI2	2=NTC(103AT	num	2=NTC(103AT	0	2	Type of analogue input AI2
15728	Cfg_AI3	3=4÷20m	num	3=4÷20mA	0	8	Type of analogue input AI3
15729	Cfg_AI4		num	3=4÷20mA	0	8	Type of analogue input AI4
15730	Cfg_AI5		num	3=4÷20mA	0	8	Type of analogue input AI5
15731	Cfg_AI6		num	3=4÷20mA	0	8	Type of analogue input AI6
15736	FullScaleMin_AI3		digit	0	-9999	9999	First value analogue input AI3 scale
15737	FullScaleMax_AI3		digit	1000	-9999	9999	Last value analogue input AI3 scale
15738	FullScaleMin_AI4		digit	0	-9999	9999	First value analogue input AI4 scale
15739	FullScaleMax_AI4	1000	digit	1000	-9999	9999	Last value analogue input AI4 scale
15740	FullScaleMin_AI5	0	digit	0	-9999	9999	First value analogue input AI5 scale
15741	FullScaleMax_AI5	1000	digit	1000	-9999	9999	Last value analogue input AI5 scale
15742	FullScaleMin_AI6	0	digit	0	-9999	9999	First value analogue input AI6 scale
15743	FullScaleMax_AI6	1000	digit	1000	-9999	9999	Last value analogue input AI6 scale
15748	Calibration_AI1	0	°C/10,°F/10	0	-180	180	Analogue input AI1 differential
15749	Calibration_AI2	0	°C/10,°F/10	0	-180	180	Analogue input AI2 differential
15750	Calibration_AI3	0	digit	0	-1000	1000	Analogue input AI3 differential
15751	Calibration_AI4	0	digit	0	-1000	1000	Analogue input AI4 differential
15752	Calibration_AI5	0	digit	0	-1000	1000	Analogue input AI5 differential
15753	Calibration_AI6	0	digit	0	-1000	1000	Analogue input AI6 differential

PLC project linking/creation

- Already existing PLC/HMI project can be linked through the related project field.

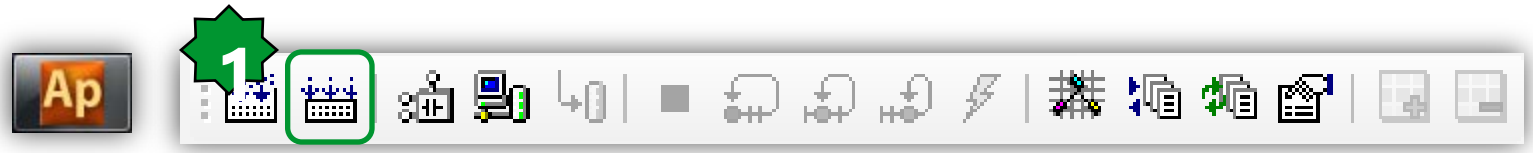


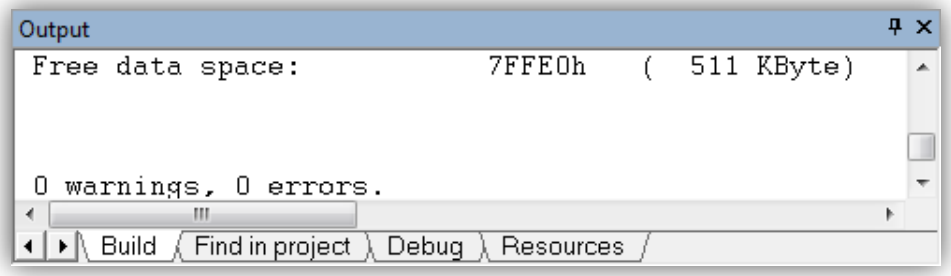


- If it is empty, a new project will be created and saved in a folder placed where the connection project is saved

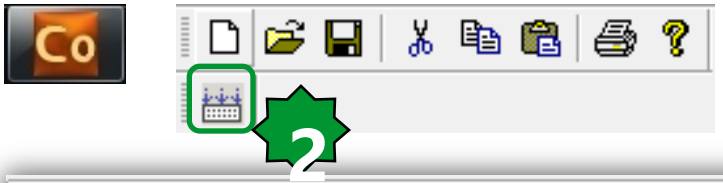
Dictionay Organization:
When a project is created & saved from CO a directory will be create; then if AP created by new from CO; a directory of the project is created inside the CO directory

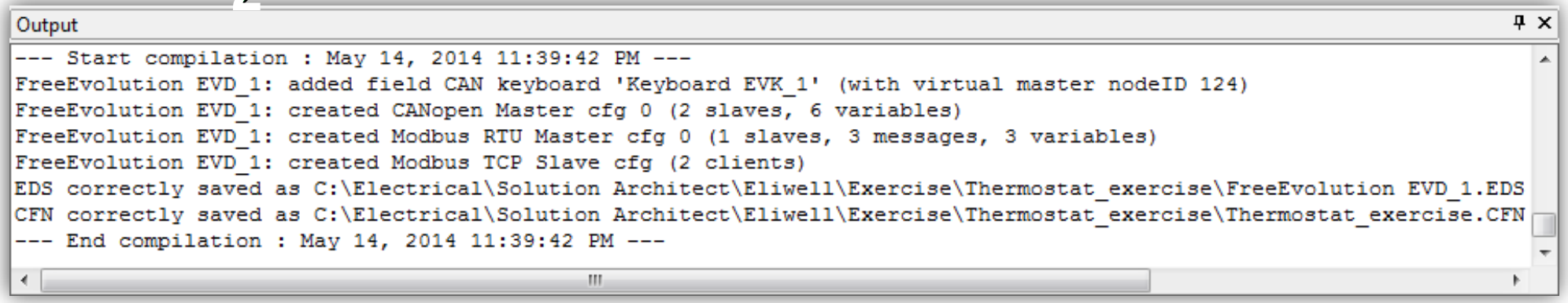
Build the connection





Note:
To apply the changes to the network, free studio asks you to reboot.



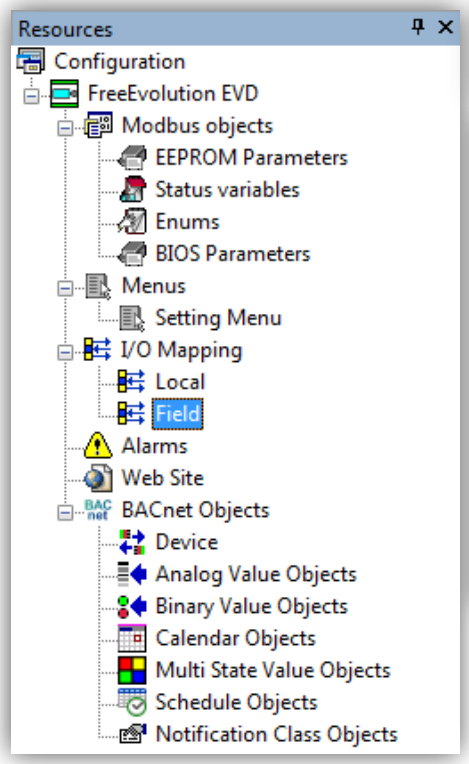


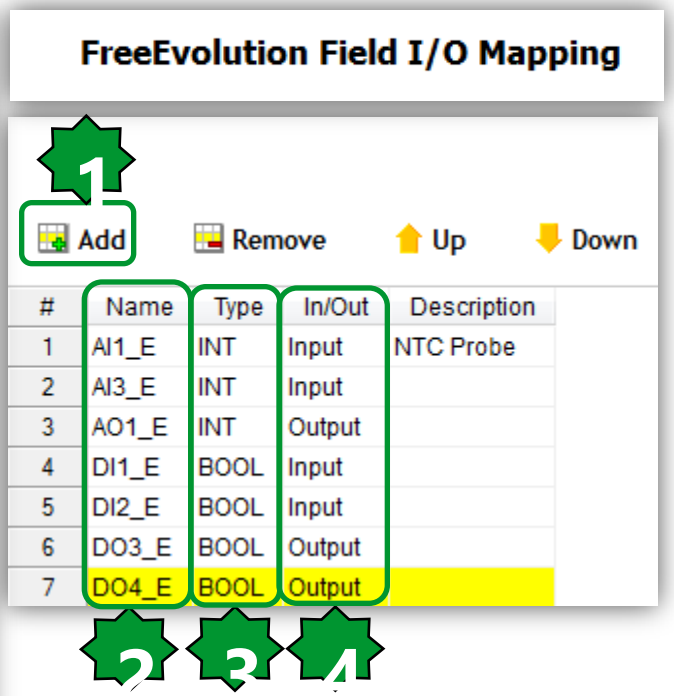
Define Application Variables to be linked to Physical I/O of EVE

The set of PLC objects you can read or write is made of:

- Status variables, created with FREE Studio Application (not BIOS).
- Field variables, created with FREE Studio Application.

1. Add
2. Name it
3. Define the type
4. Define the In/Out

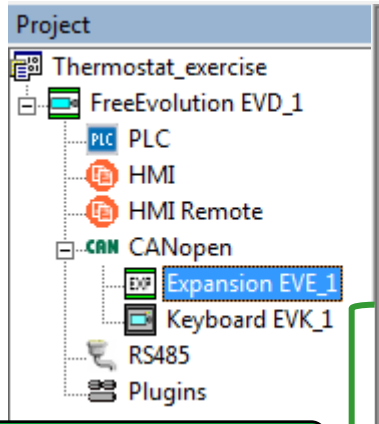




- NOTE: If the Status Variables is defined in order to be linked to an EVE input

it must be set as not READ ONLY

EVE Expansion configuration



Expansion EVE Configuration

General | SDO Set | PDO Tx - Input | PDO Rx - Output

Assign UnAssign

#	Idx	Sub	PDO	Bit	COBID	Object Name	Type	Size	Label	DataBlock
1	6000	1	1	0	181	Read Input 1h to 8h	BOOL	1		
2	6000	1	1	1	181	Read Input 1h to 8h	BOOL	1		
3	6000	1	1	2	181	Read Input 1h to 8h	BOOL	1		
4	6000	1	1	3	181	Read Input 1h to 8h	BOOL	1		
5	6000	1	1	4	181	Read Input 1h to 8h	BOOL	1		
6	6000	1	1	5	181	Read Input 1h to 8h	BOOL	1		
7	6000	1	1	6	181	Read Input 1h to 8h	BOOL	1		
8	6000	1	1	7	181	Read Input 1h to 8h	BOOL	1		
9	6000	2	1	8	181	Read Input 9h to 16h	BOOL	1		
10	6000	2	1	9	181	Read Input 9h to 16h	BOOL	1		
11	6000	2	1	10	181	Read Input 9h to 16h	BOOL	1		
12	6000	2	1	11	181	Read Input 9h to 16h	BOOL	1		
13	6401	1	2	0	281	Analogue Input 1	INT	16		
14	6401	2	2	16	281	Analogue Input 2	INT	16		
15	6401	3	2	32	281	Analogue Input 3	INT	16		
16	6401	4	2	48	281	Analogue Input 4	INT	16		
17	6401	5	3	0	381	Analogue Input 5	INT	16		
18	6401	6	3	16	381	Analogue Input 6	INT	16		
19	2230	0	5	0	481	Counter	UDINT	32		
20	2232	0	5	32	481	Frequency	UDINT	32		

Digital Inputs

Dip switches

Analogue Inputs

Fast Digital Input

- Select the Analogue Input 1 of EVE_1, PDO Tx – Input
- Press Assign
- Link the Physical input to the desired Application variable
- Repeat this for each EVE Input used in your project
- Use PDO Rx – Output for EVE Output

Assign/UnAssign of physical I/O

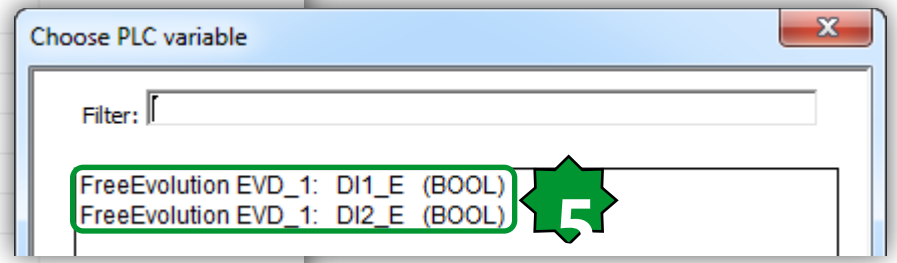
Expansion EVE Configuration

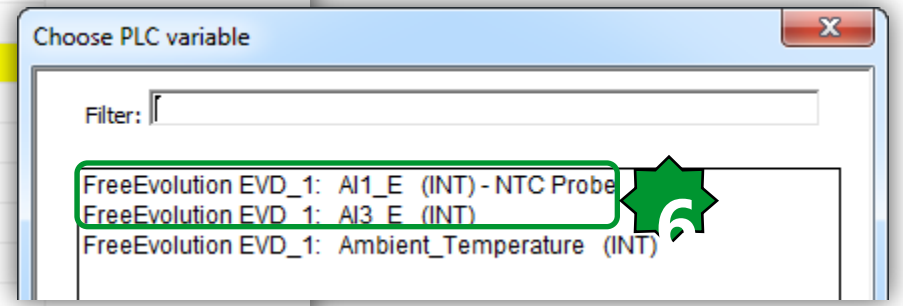
General | SDO Set | PDO Tx - Input | PDO Rx - Output

Assign | UnAssign

#	Idx	Sub	PDO	Bit	COBID	Object Name	Type	Size	Label	DataBlock
1	6000	1	1	0	181	Read Input 1h to 8h	BOOL	1	DI1_E	IX10.0
2	6000	1	1	1	181	Read Input 1h to 8h	BOOL	1	DI2_E	IX10.1
3	6000	1	1	2	181	Read Input 1h to 8h	BOOL	1		
4	6000	1	1	3	181	Read Input 1h to 8h	BOOL	1		
5	6000	1	1	4	181	Read Input 1h to 8h	BOOL	1		
6	6000	1	1	5	181	Read Input 1h to 8h	BOOL	1		
7	6000	1	1	6	181	Read Input 1h to 8h	BOOL	1		
8	6000	1	1	7	181	Read Input 1h to 8h	BOOL	1		
9	6000	2	1	8	181	Read Input 9h to 16h	BOOL	1		
10	6000	2	1	9	181	Read Input 9h to 16h	BOOL	1		
11	6000	2	1	10	181	Read Input 9h to 16h	BOOL	1		
12	6000	2	1	11	181	Read Input 9h to 16h	BOOL	1		
13	6401	1	2	0	281	Analogue Input 1	INT	16	AI1_E	IW11.0
14	6401	2	2	16	281	Analogue Input 2	INT	16		
15	6401	3	2	32	281	Analogue Input 3	INT	16	AI3_E	IW11.1
16	6401	4	2	48	281	Analogue Input 4	INT	16		
17	6401	5	3	0	381	Analogue Input 5	INT	16		
18	6401	6	3	16	381	Analogue Input 6	INT	16		
19	2230	0	5	0	481	Counter	UDINT	32		
20	2232	0	5	32	481	Frequency	UDINT	32		

1. PDO Tx-Intput
2. Choose PLC variable DI
3. Choose PLC variable AI
4. Assign





Assign/UnAssign of physical I/O

1

Expansion EVE Configuration

General | SDO Set | PDO Tx - Input | PDO Rx - Output

4

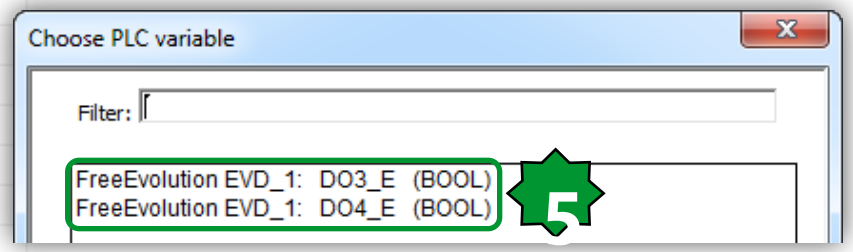
Assign | UnAssign

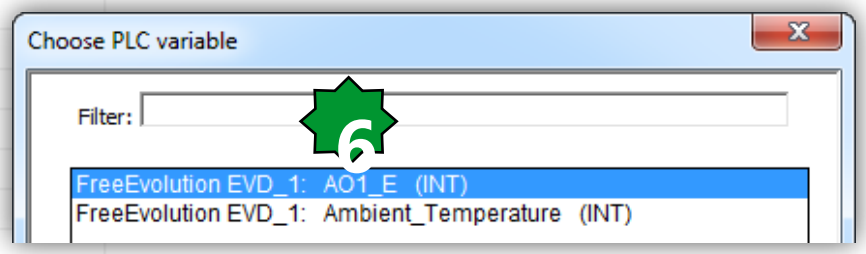
#	Idx	Sub	PDO	Bit	COBID	Object Name	Type	Size	Label	DataBlock
1	6200	1	1	0	201	Write Output 1h to 8h	BOOL	1		
2	6200	1	1	1	201	Write Output 1h to 8h	BOOL	1		
3	6200	1	1	2	201	Write Output 1h to 8h	BOOL	1	DO3_E	QX11.0
4	6200	1	1	3	201	Write Output 1h to 8h	BOOL	1	DO4_E	QX11.1
5	6200	1	1	4	201	Write Output 1h to 8h	BOOL	1		
6	6200	1	1	5	201	Write Output 1h to 8h	BOOL	1		
7	6200	1	1	6	201	Write Output 1h to 8h	BOOL	1		
8	6411	1	2	0	301	Analogue Output 1	INT	16	AO1_E	QW10.0
9	6411	2	2	16	301	Analogue Output 2	INT	16		
10	6411	3	2	32	301	Analogue Output 3	INT	16		
11	6411	4	2	48	301	Analogue Output 4	INT	16		
12	6411	5	3	0	401	Analogue Output 5	INT	16		
13	21c0	0	5	0	501	LED1	USINT	8		
14	21c1	0	5	8	501	LED2	USINT	8		
15	21c2	0	5	16	501	LED3	USINT	8		

2

3

1. PDO Rx-Output
2. Choose PLC variable DO
3. Choose PLC variable AO
4. Assign

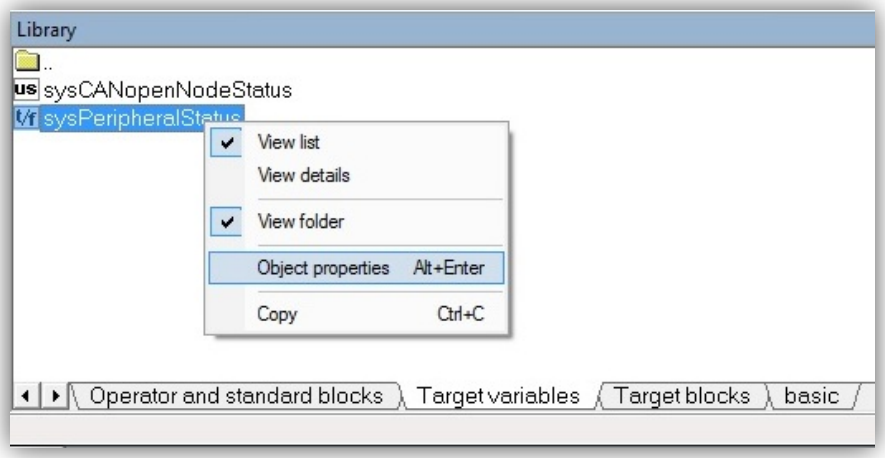


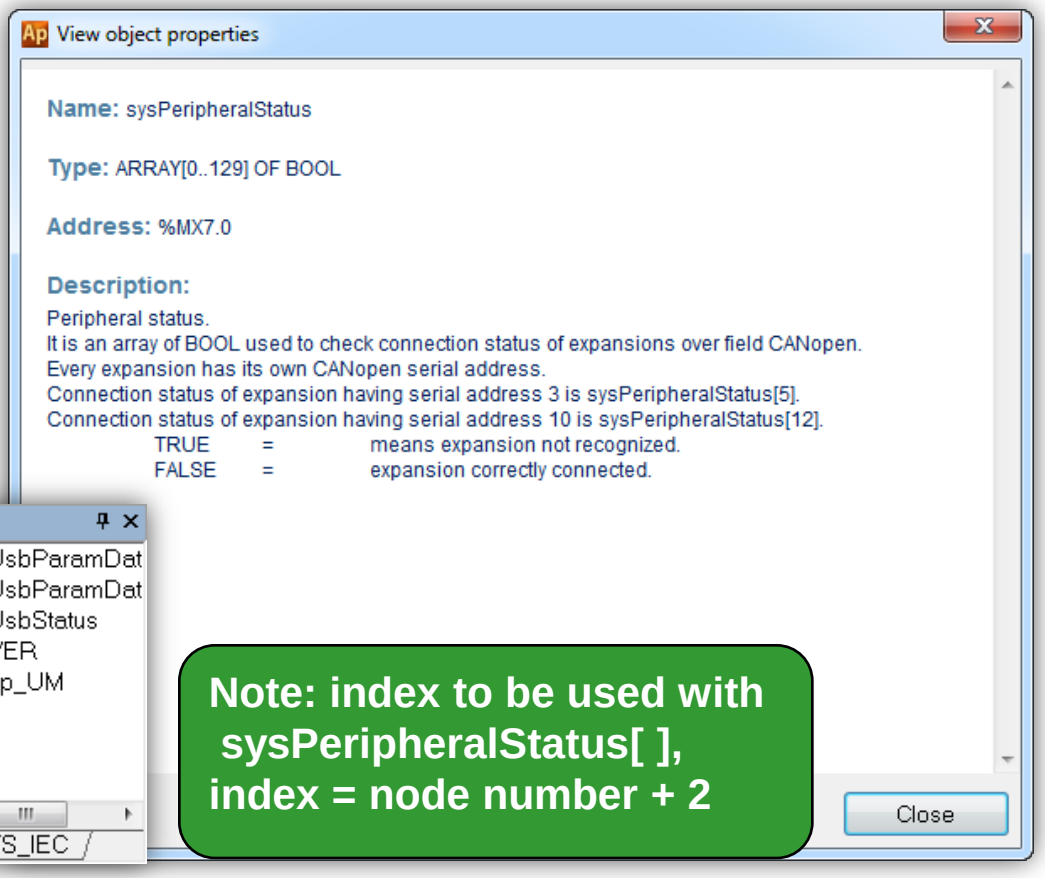


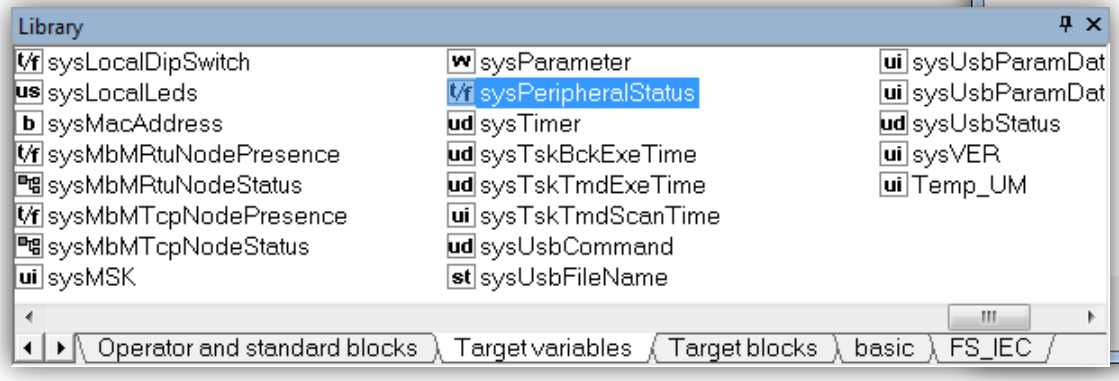
Application Project

You can create your Application project in the usual way using local and field I/O

- sysPeripheralStatus[3] tells the communication status with EVE_1

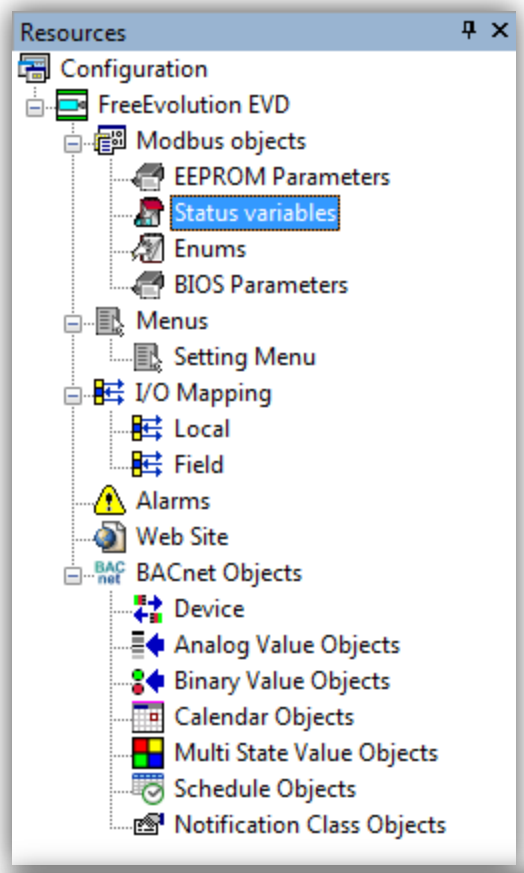






Note: index to be used with sysPeripheralStatus[ ], index = node number + 2

Status Variable



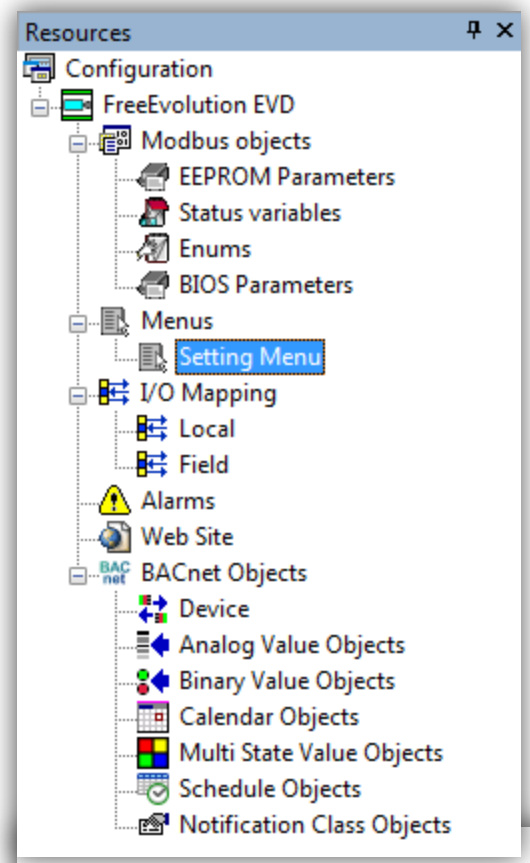
- Create Status Variables readable via Modbus

FreeEvolution Status Variables

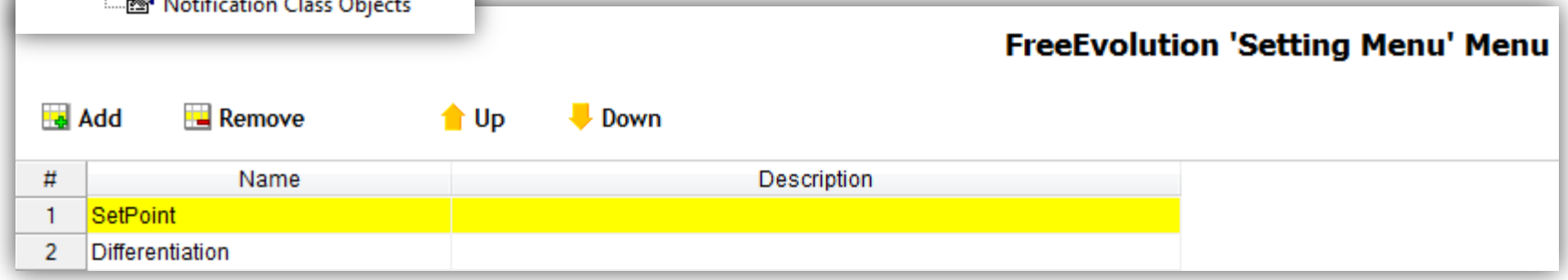
Add Remove Recalc

#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Read only
1	8960	Ambient_Temperature	Signed 16-bit	INT					1	0	°C	XXX.Y	Always visible	True

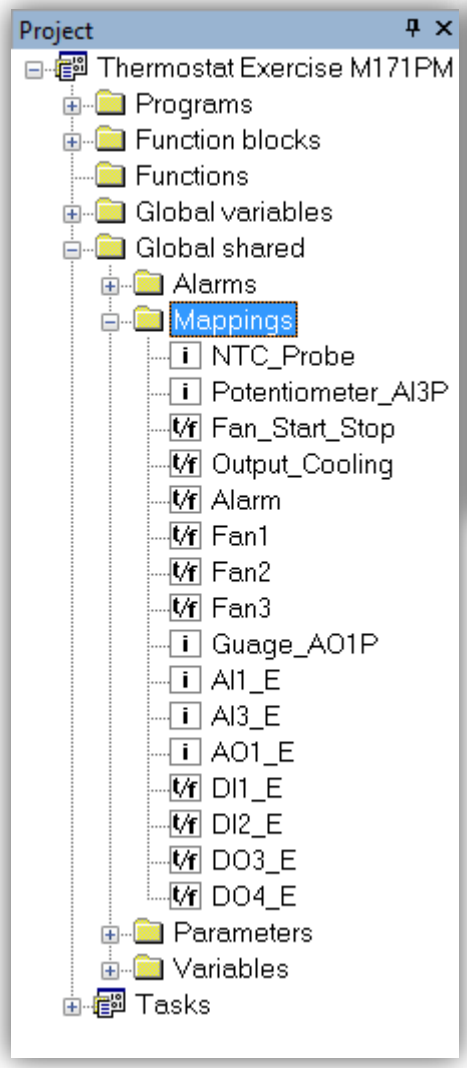
Setting Menu

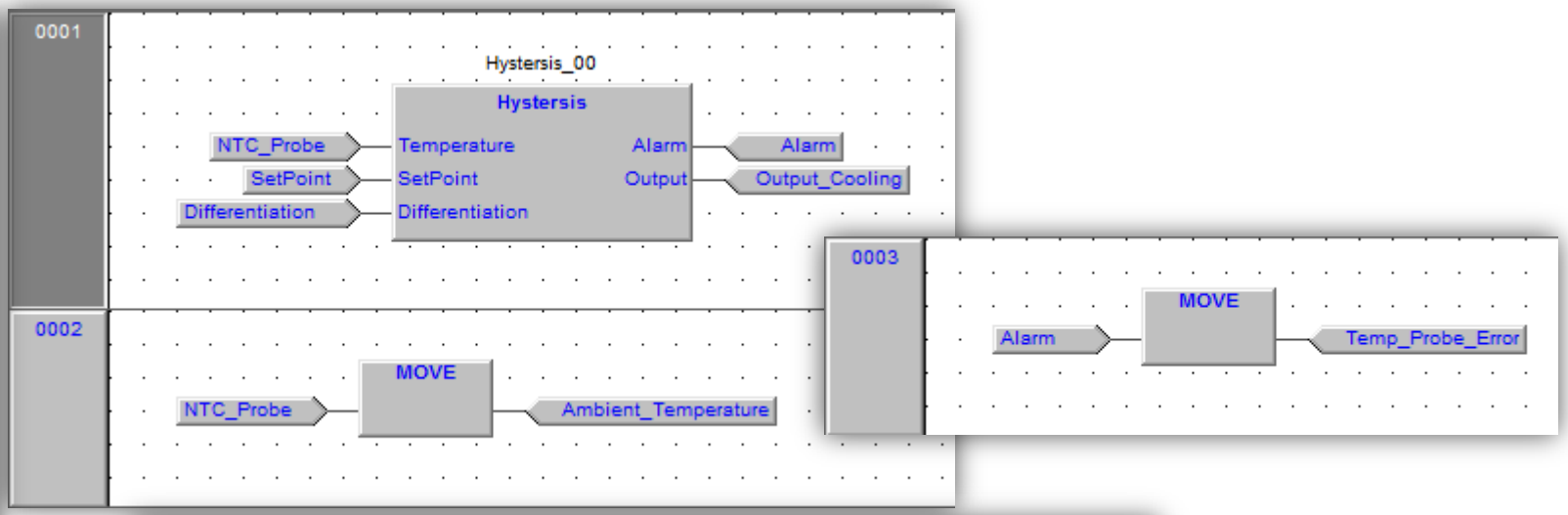


- Create folder to be shown on Free Studio Device



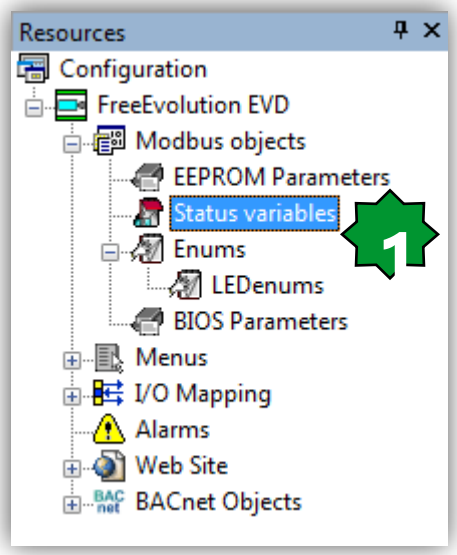
Communication Alarm Checking, Link I/O of EVE






```
0001  (* Hystersis FBD *)
0002
0003  if Temperature >= Setpoint + Differentiation then
0004      Output := TRUE;
0005  end_if;
0006
0007  if Temperature < Setpoint then
0008      Output := FALSE;
0009  end_if;
0010
0011  (* Probe disconnection detector *)
0012
0013  if Temperature = -32768 then
0014      Alarm := TRUE;
0015      else Alarm := FALSE;
0016  end_if;
0017
0018
```

Green LED management



1. Resources ► Status variables
2. Add
3. Define: Name, device & application types

2

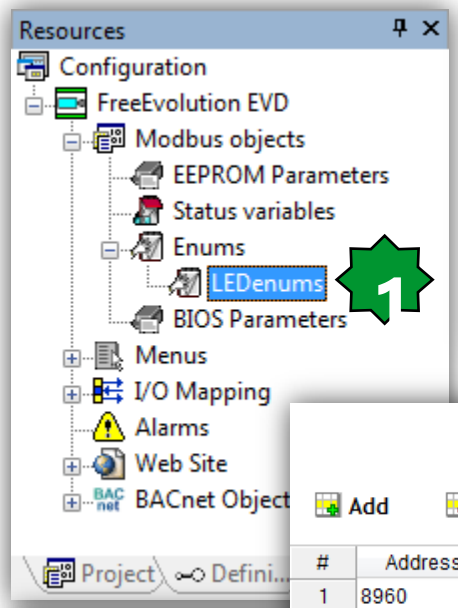
FreeEvolution Status Variables

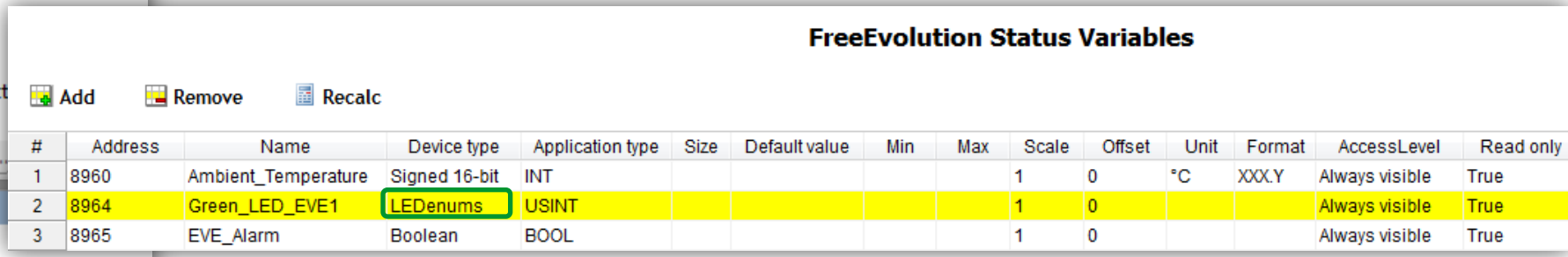
Add | Remove | Recalc

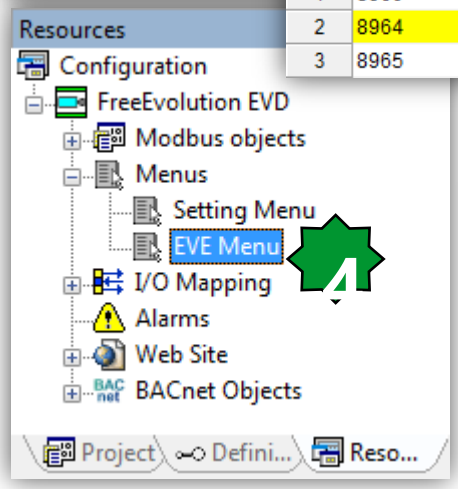
#	Address	Name	Device type	Application type	Size	Default value	Min	Max	Scale	Offset	Unit	Format	AccessLevel	Read only
1	8960	Ambient_Temperature	Signed 16-bit	INT					1	0	°C	XXX.Y	Always visible	True
2	8964	Green_LED_EVE1	Unsigned 8-bit	USINT					1	0			Always visible	True
3	8965	EVE_Alarm	Boolean	BOOL					1	0			Always visible	True

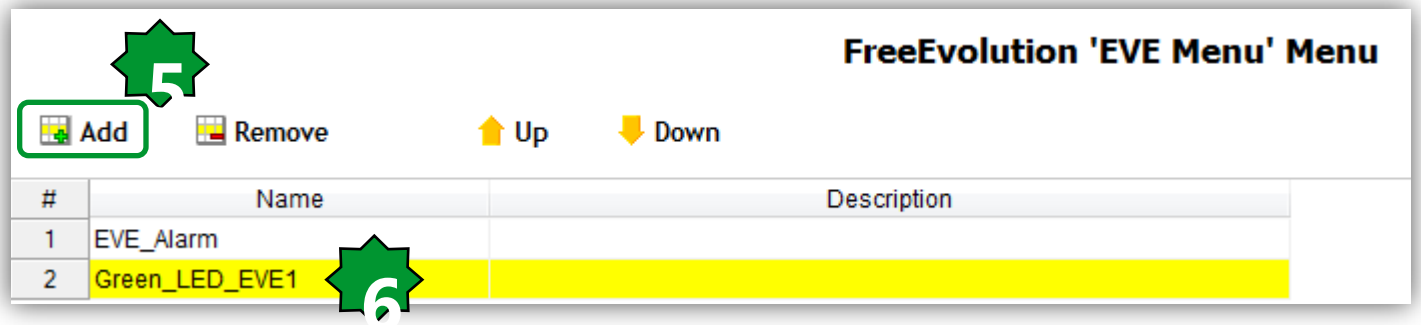
3

Green LED management

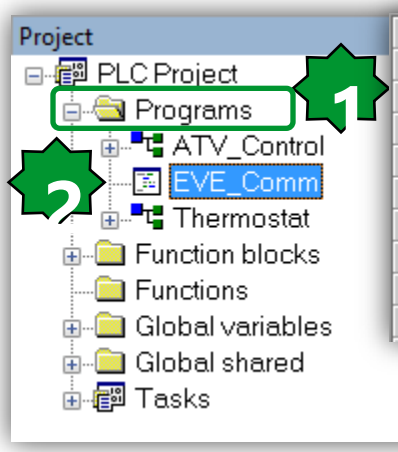






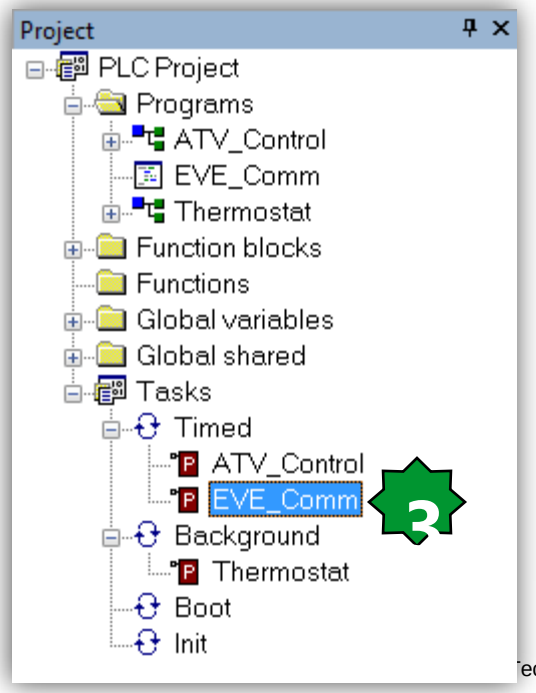


Green LED management




```
0001  (* Green Led will blink when communication works *)
0002  Green_LED_EVE1 := 2;
0003
0004  (* DO1_E of EVE IS ON when AI1_E is measuring more than 20 degree *)
0005  DO1_E := (AI1_E >= 200);
0006  (* DO2 of EVE is ON when AI1_E probe is disconnected *)
0007  DO2_E := (AI1_E = -32768);
0008  (* EVE communication alarm *)
0009  EVE_Alarm := sysPeripheralStatus[3];
```

4



Name: sysPeripheralStatus

Type: ARRAY[0..129] OF BOOL

Address: %MX7.0

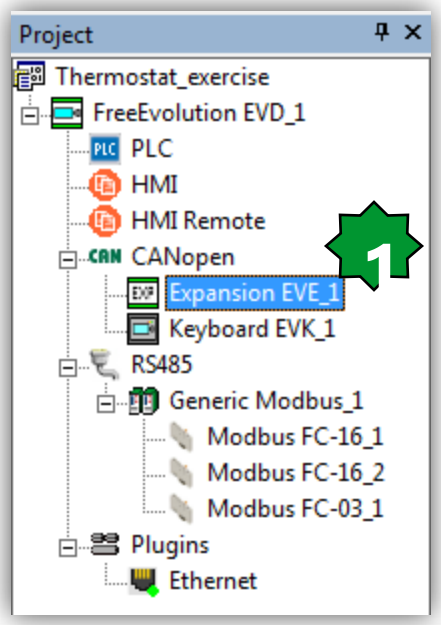
Description:

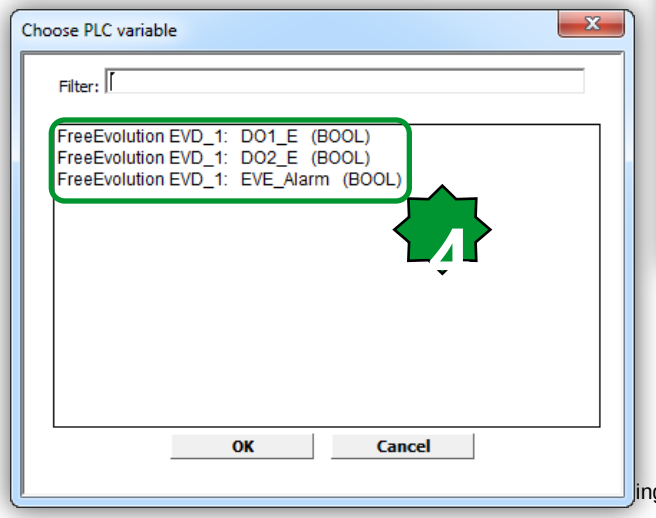
Peripheral status.
It is an array of BOOL used to check connection status of expansions over field CANopen.
Every expansion has its own CANopen serial address.
Connection status of expansion having serial address 3 is sysPeripheralStatus[5].
Connection status of expansion having serial address 10 is sysPeripheralStatus[12].

TRUE	=	means expansion not recognized.
FALSE	=	expansion correctly connected.

1.Project ► Programs ► new program
2. Name it as: EVE_Comm
3. Assign it to the timed task
4. Program
5. Communication configuration ► OK

Green LED management/Output





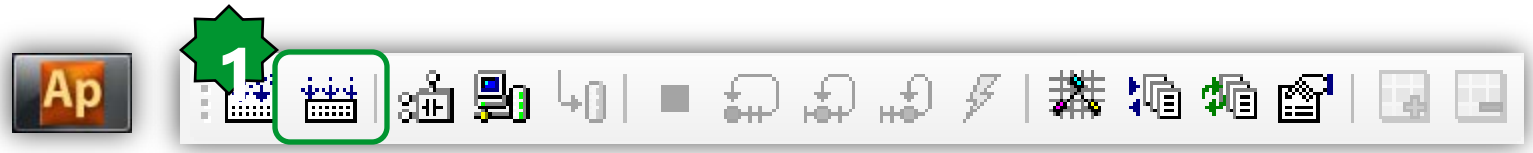
Expansion EVE Configuration

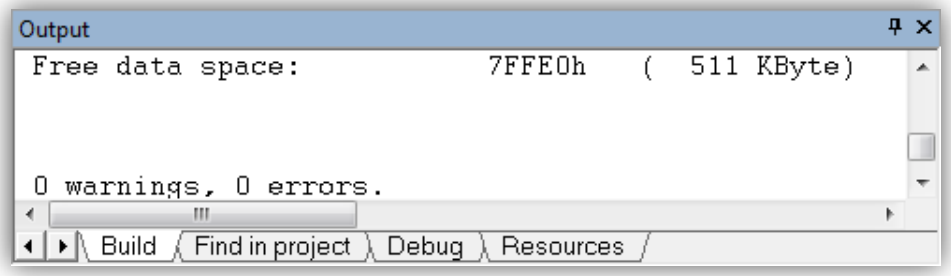
General | SDO Set | PDO Tx - Input | PDO Rx - Output

3 Assign UnAssign 2

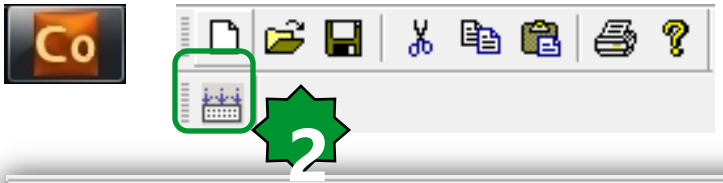
#	Idx	Sub	PDO	Bit	COBID	Object Name	Type	Size	Label	DataBlock
1	6200	1	1	0	201	Write Output 1h to 8h	BOOL	1	DO1_E	QX11.0
2	6200	1	1	1	201	Write Output 1h to 8h	BOOL	1	DO2_E	QX11.1
3	6200	1	1	2	201	Write Output 1h to 8h	BOOL	1		
4	6200	1	1	3	201	Write Output 1h to 8h	BOOL	1		
5	6200	1	1	4	201	Write Output 1h to 8h	BOOL	1		
6	6200	1	1	5	201	Write Output 1h to 8h	BOOL	1		
7	6200	1	1	6	201	Write Output 1h to 8h	BOOL	1		
8	6411	1	2	0	301	Analogue Output 1	INT	16		
9	6411	2	2	16	301	Analogue Output 2	INT	16		
10	6411	3	2	32	301	Analogue Output 3	INT	16		
11	6411	4	2	48	301	Analogue Output 4	INT	16		
12	6411	5	3	0	401	Analogue Output 5	INT	16		
13	21c0	0	5	0	501	LED1	USINT	8	Green_LED_EVE1	MW110.4
14	21c1	0	5	8	501	LED2	USINT	8		
15	21c2	0	5	16	501	LED3	USINT	8		

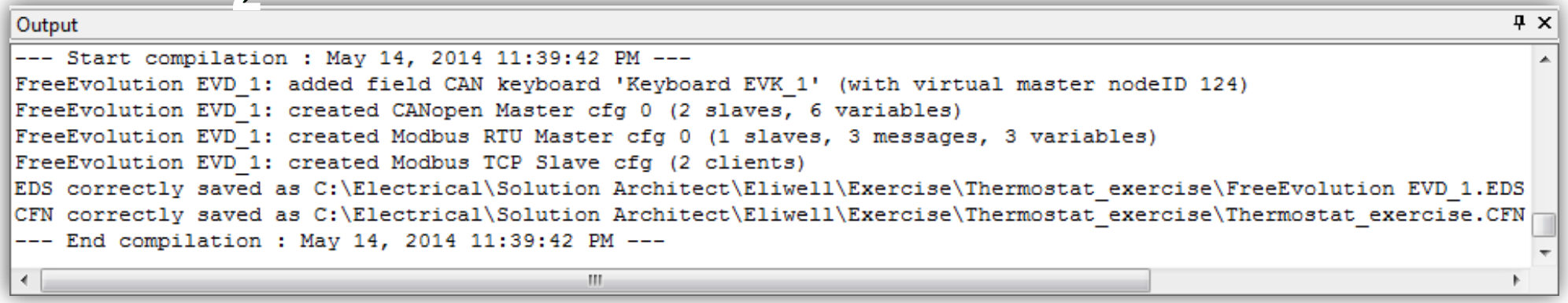
Build the connection



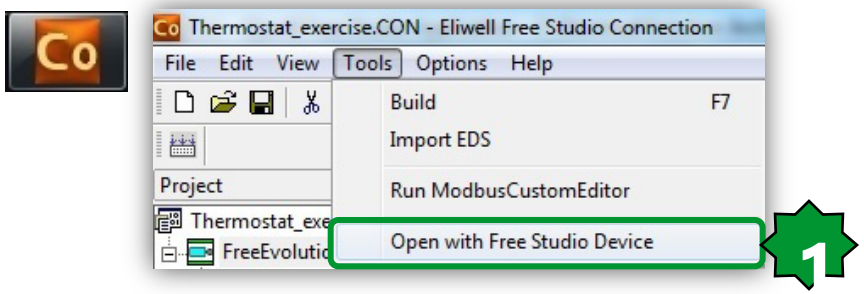


Note:
To apply the changes to the network, free studio asks you to reboot.

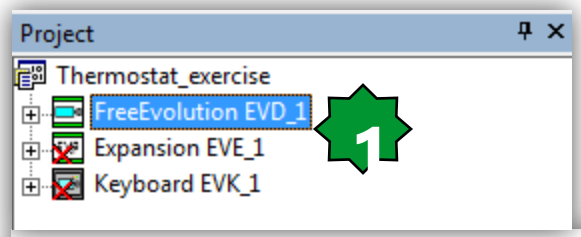


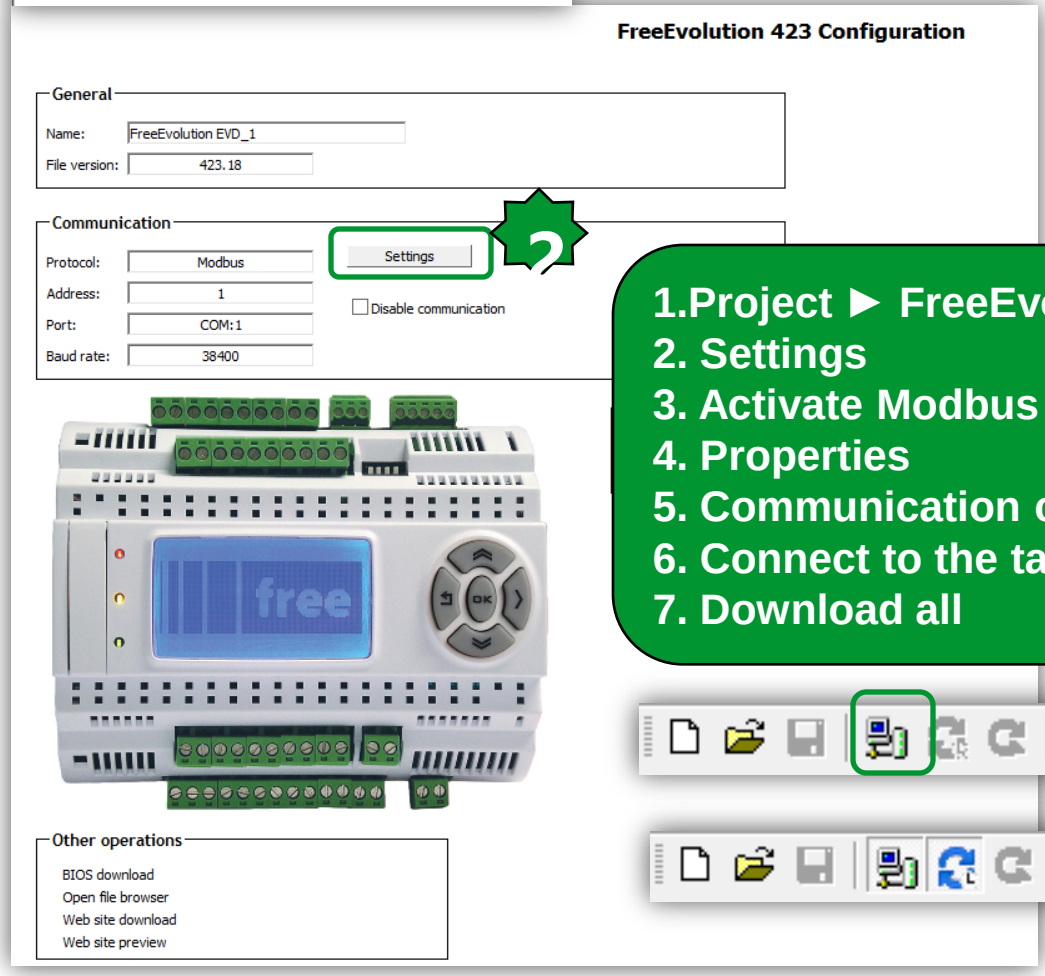


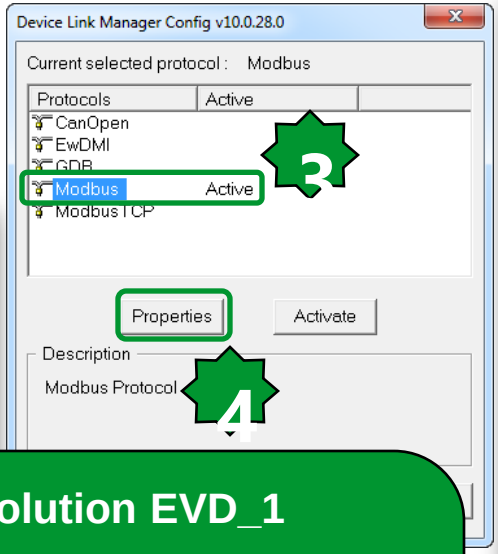
Open with free studio device

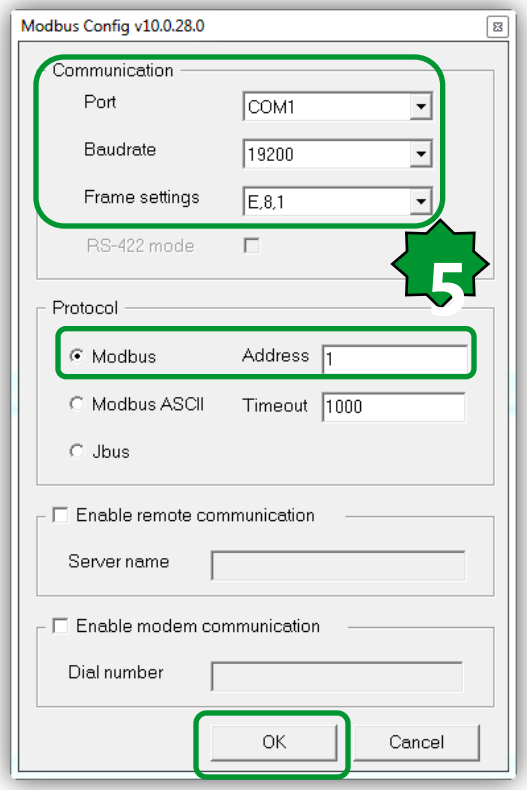


Download via RS485

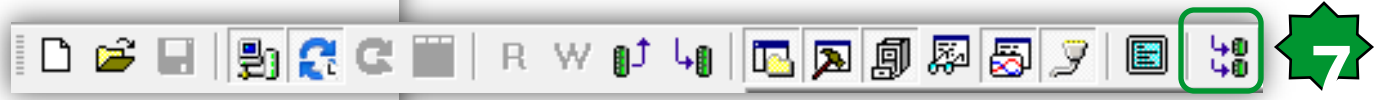




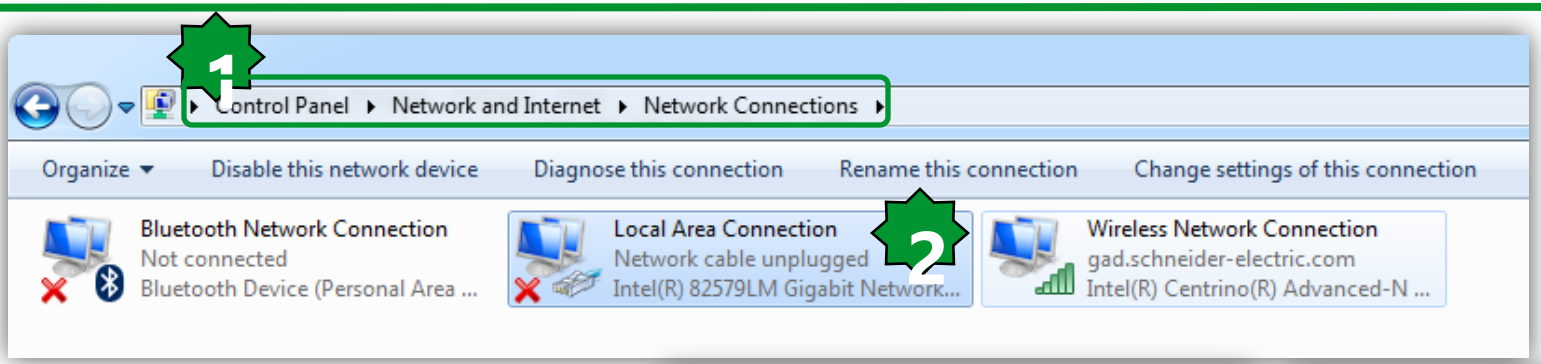


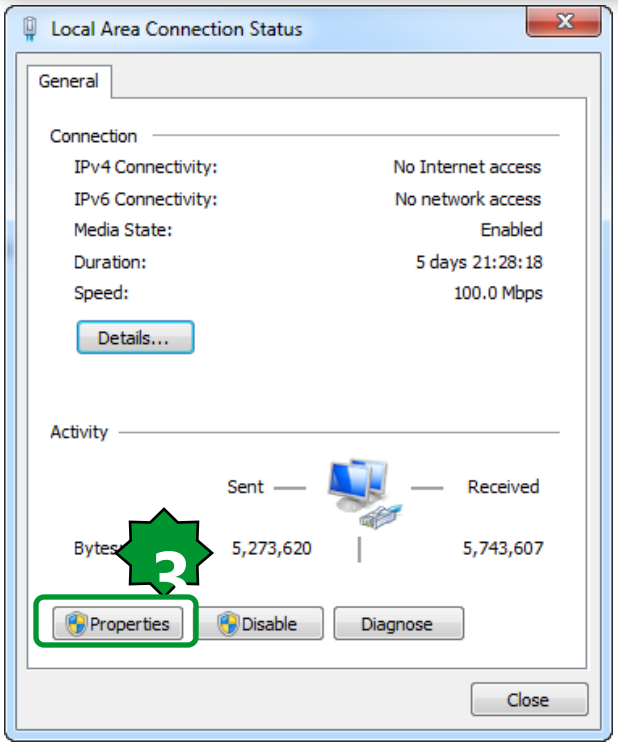


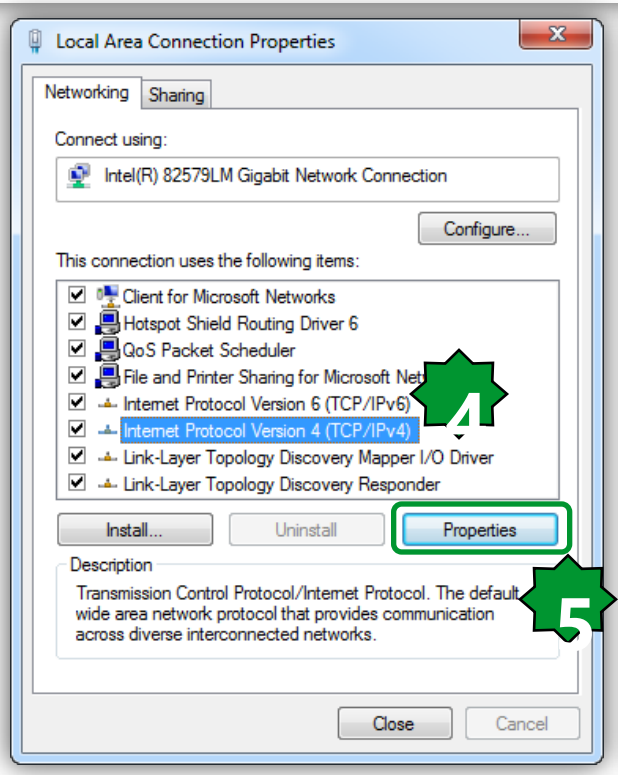
1.Project ► FreeEvolution EVD_1
2. Settings
3. Activate Modbus
4. Properties
5. Communication configuration ► OK
6. Connect to the target
7. Download all

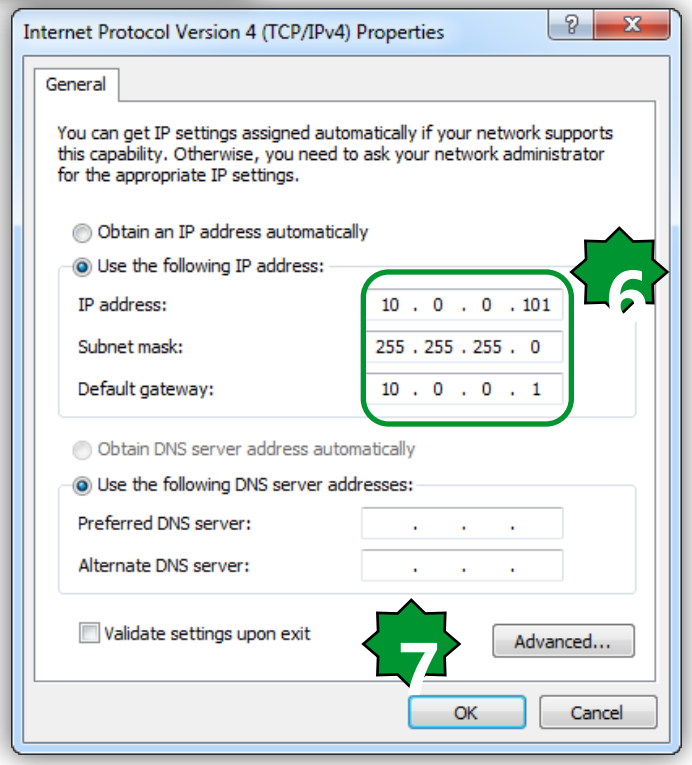


PC TCP/IP configuration

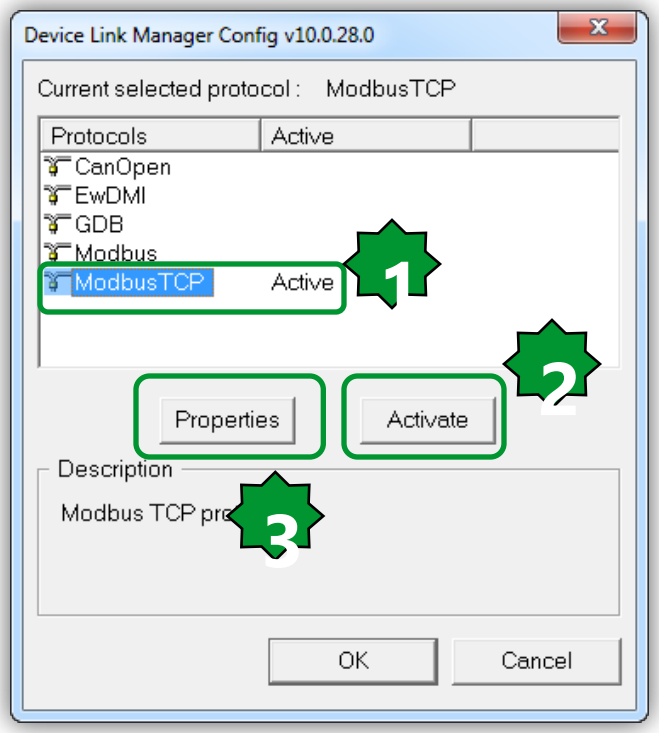


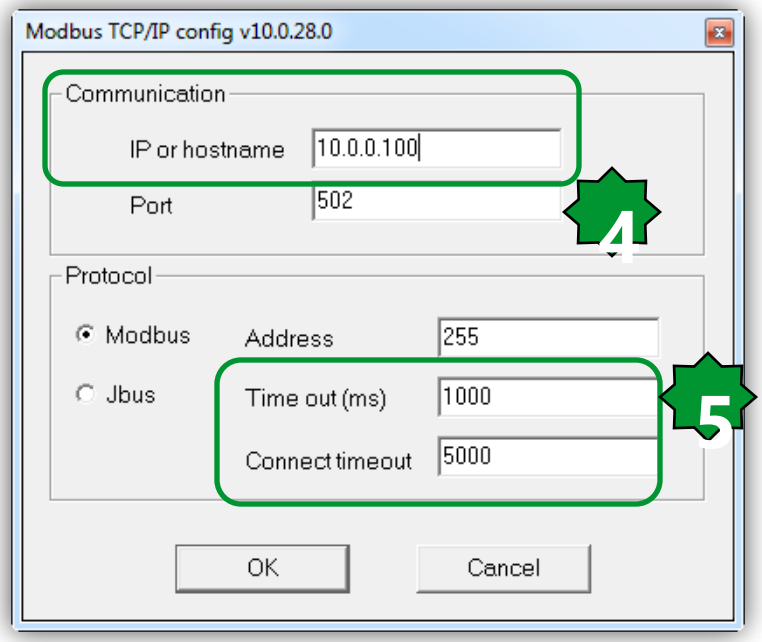




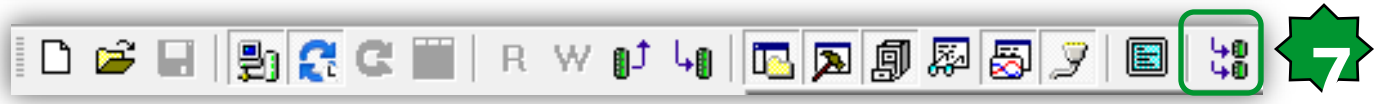


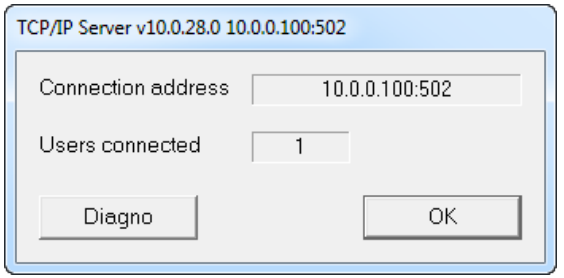
Modbus TCP/download





5.change Timeout and Connect Timeout based on the Ethernet band available. If the connection it is not direct it is better to increase both of them 10000-20000





Chapter 12

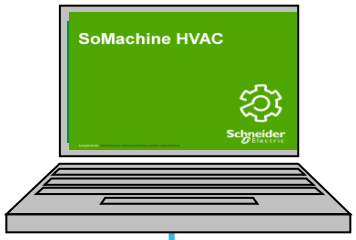
Modbus Communication

Goal:

Connection ATV21/212 to the Evolution via Modbus serial line, write the command + speed reference and read the output frequency

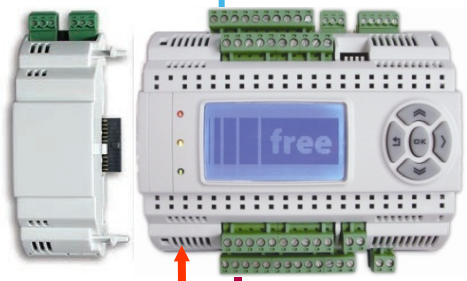
Machines architecture

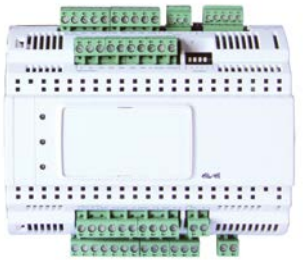
- On Board RS485 SL can be either Master or Slave
- Plugin RS485 SL can be either Master or Slave
- They can't be both master at the same time



Program. port

EVD*





Modbus Master

Other Machines

Modbus Slave USB/RS485

Magelis Family Range

or

BMS Modbus Master

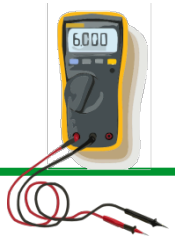
Altivar Drives e.g ATV212/21

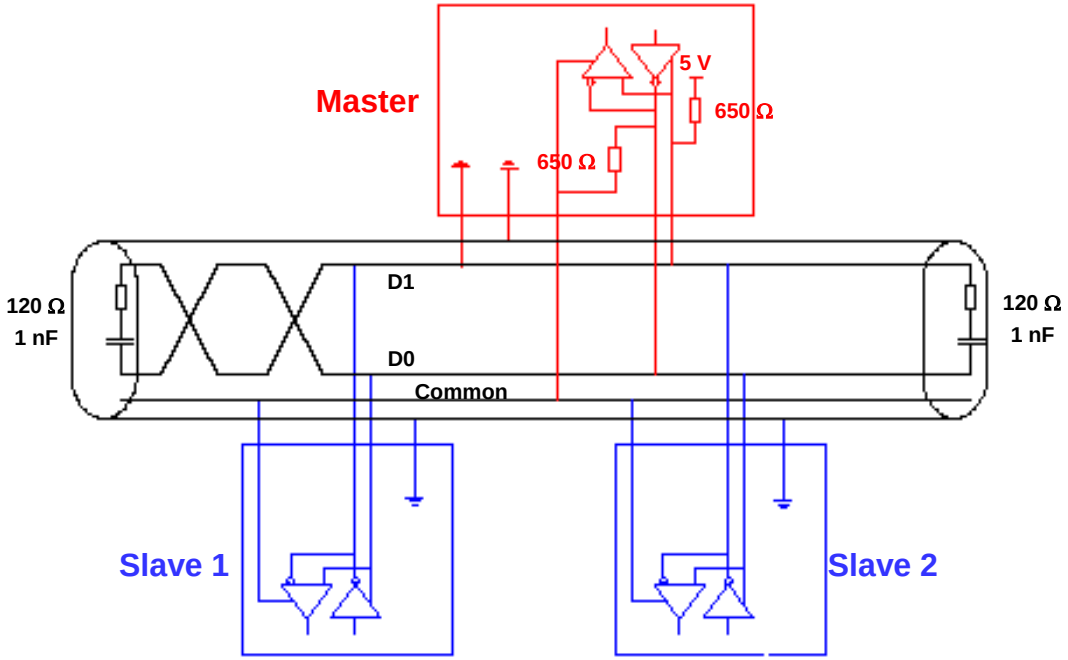
3rd Party Modbus slave device

ATV 21/212 via Modbus

Set ups at ATV21:
-LOC / REM = off
-CN0d = 2 Enables Start / Stop control via network
-FN0d = 4 Enables frequency reference to be Controlled by network.
-F800: 1 (default) = 19200bps
-F801: 1 (default) = even parity
-F802: 1 = Address of ATV21
-F803: 4 = Timeout in seconds
-F829: 1 (default) = Protocol Modbus RTU
-F851: 1 = Communication fault setting
-(Last commanded operation continues)

Modbus RS485 standard schematic

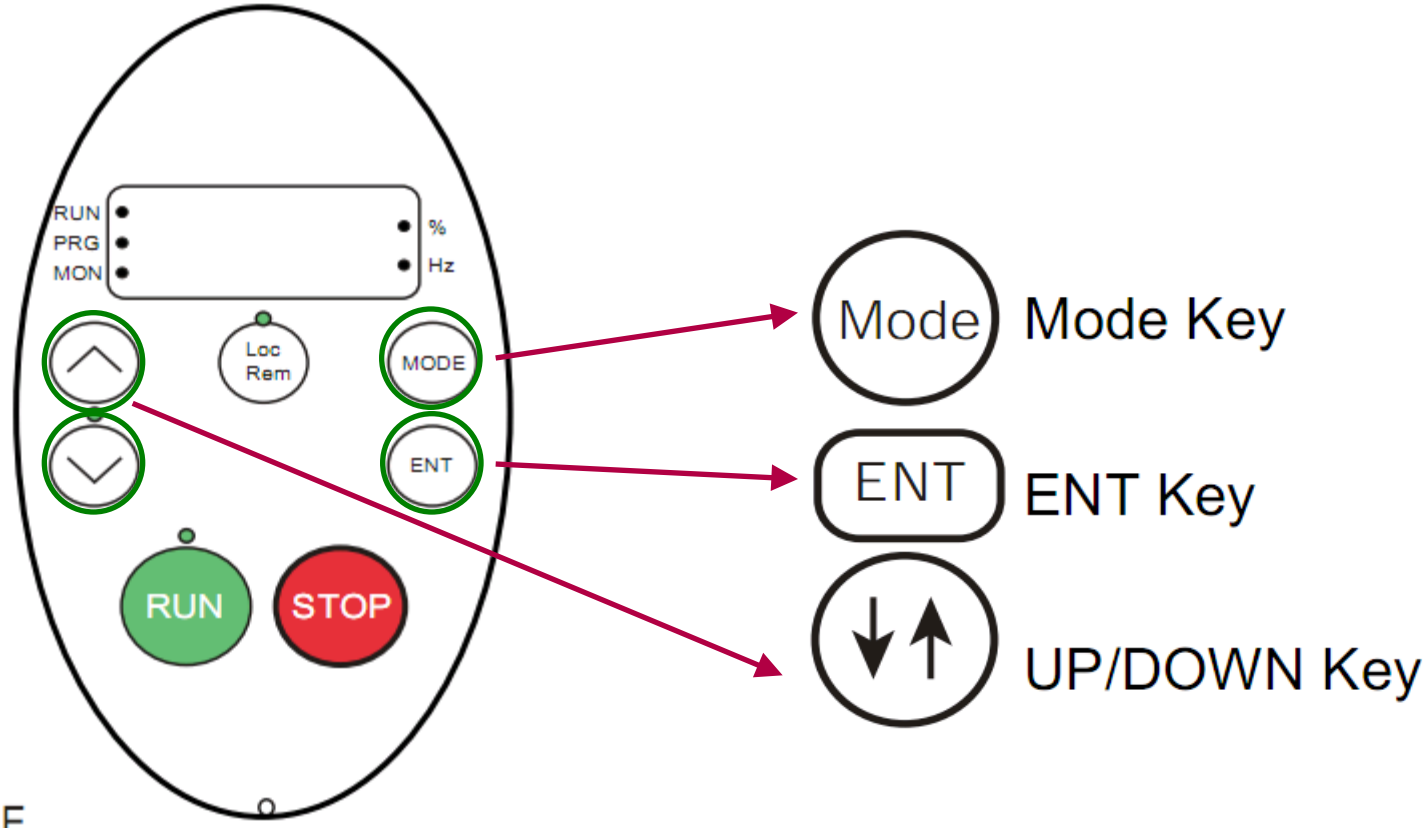




Maximum length of bus	1000 m at 19200 bps
Maximum number of stations (without repeater)	32 (31 slaves)
Maximum length of tap links	20 m for one tap link 40 m divided by the number of tap links
Bus polarisation	650 Ω at 5V and common for the master
Line terminator	120 Ω - 0,25Wm in series with 1nF 10V
Common polarity	Yes (Common) connected to the PG

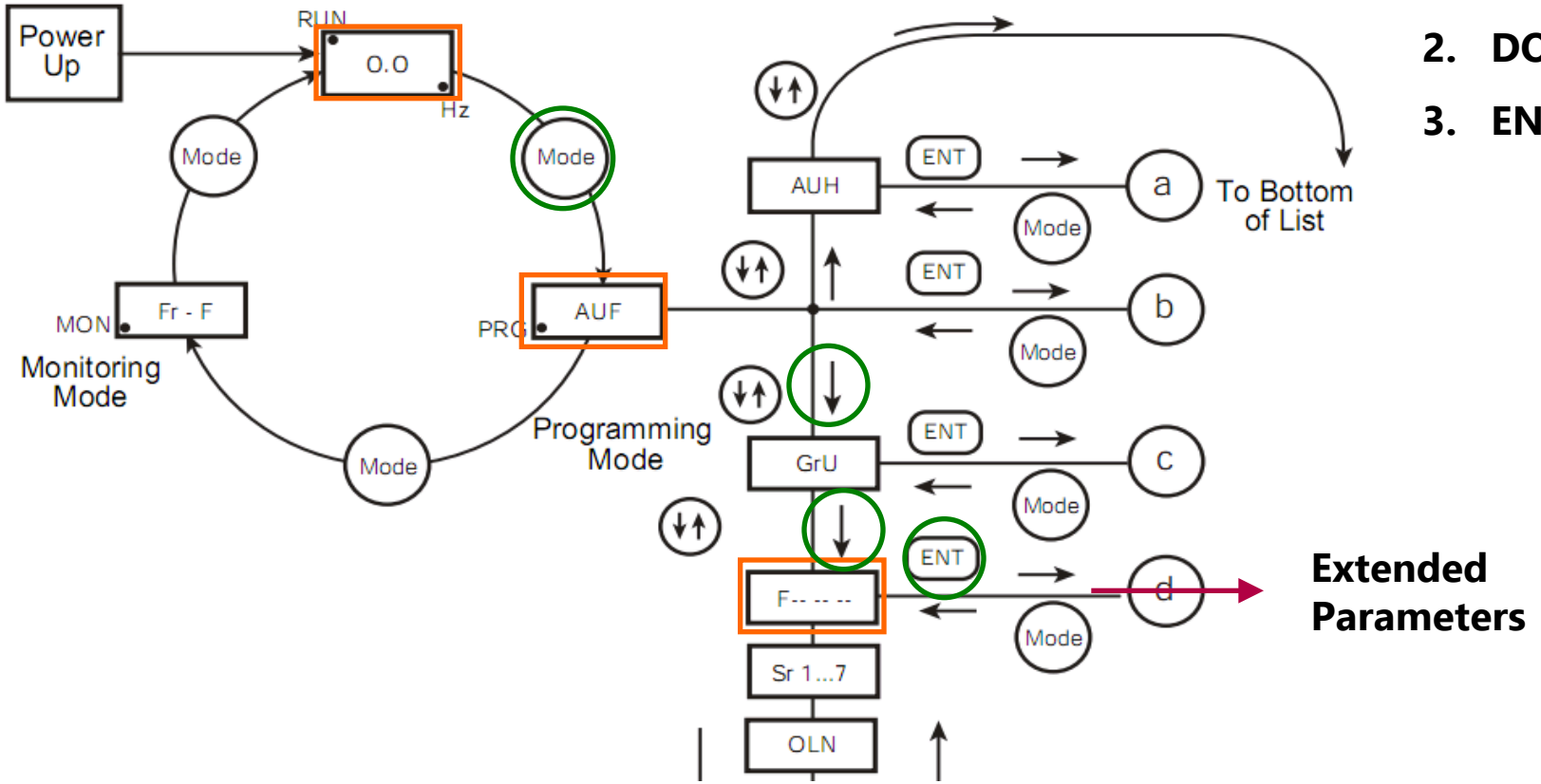
Modbus – ATV21/212

- Drive (slave)
  - use Drive Keypad for setting Modbus parameters



Modbus – ATV21/212 navigation

- Drive (slave)
  - Drive Menu

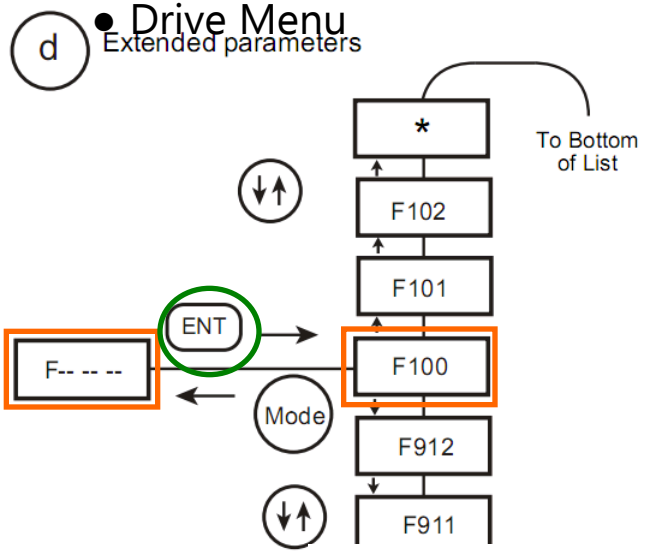


1. **MODE**
2. **DOWN KEY 2x**
3. **ENTER on F---**

Extended Parameters

Modbus – ATV21/212 configuration

- Drive (slave)



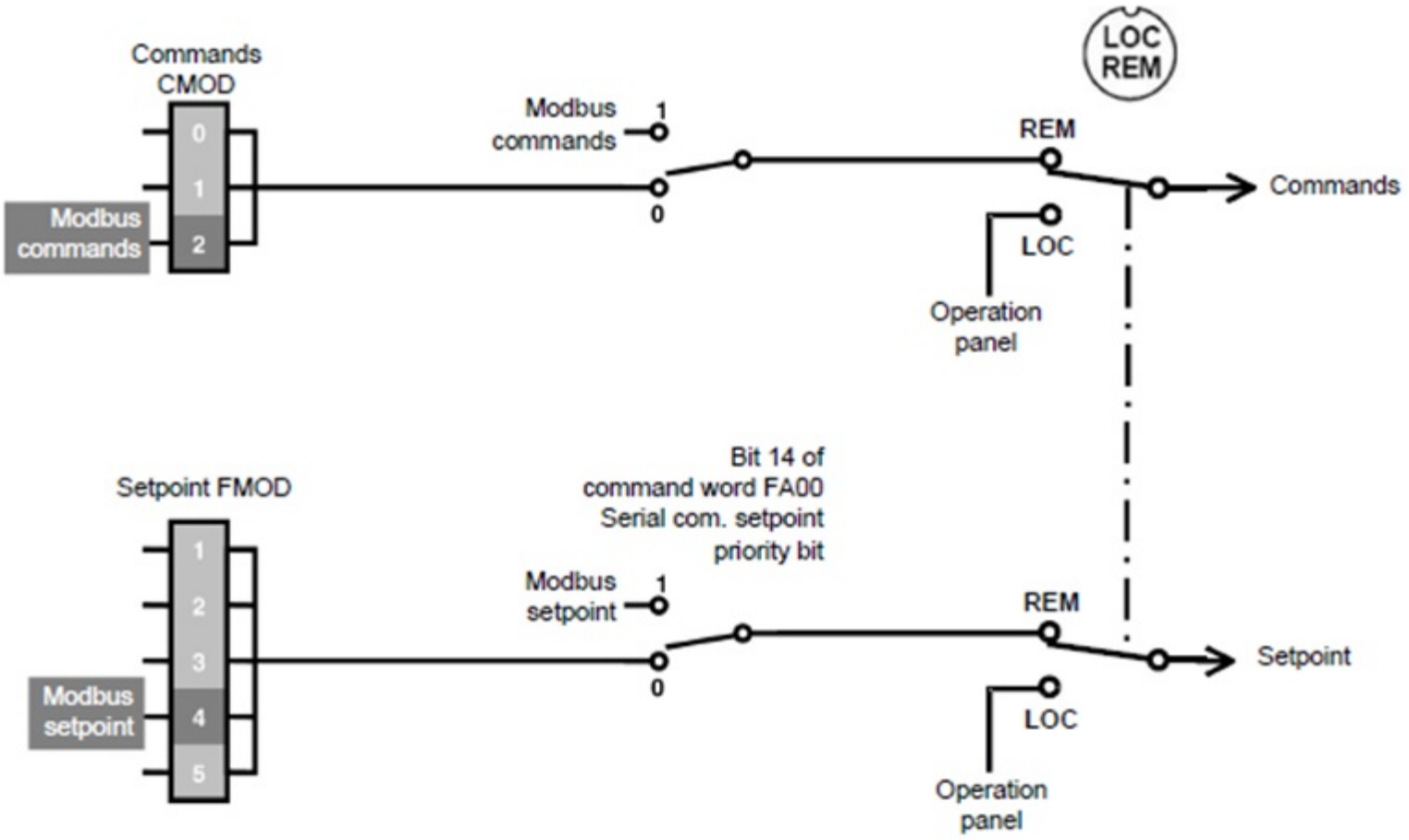
1. Press **DOWN** key repeatedly advance to **F829**
2. press **ENTER** key to change setting
3. with **UP/DOWN** keys change value
4. press **ENTER** to confirm and exit
5. repeat for parameters listed

Important: Cycle power on drive after changing Modbus settings for changes to become effective.

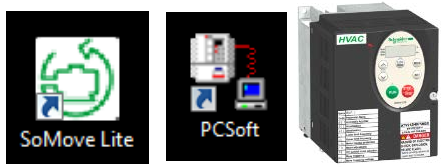
Title	Communication Number	Function	Adjustment range	Unit	Default setting	Valid
F829	0829	Selection of communication protocol	0 ... 4 1: Modbus-RTU protocol	-	1	After reset
F800	0800	Baud rate	0: 9600 bps 1: 19200 bps	-	1	After reset
F801	0801	Parity	0: NON (No parity) 1: EVEN (Even parity) 2: ODD (Odd parity)	-	1	After reset
F802	0802	Modbus address	0 ... 247	-	1	After setting

Modbus – ATV21/212 configuration Command & Speed Reference

Parameter title	Function number	Function description
CNOD	2	Serial communication
FNOD	4	Serial communication



Modbus – ATV21/212 configuration Command & Speed Reference



Code	Name/Description	Adjustment range	Factory setting
CMOd	❑ **Remote Mode Start/Stop Control**	-	0

The setting of parameter CMOd determines the source of start, stop, forward, and reverse operation commands when the drive is in remote mode.

The drive must be stopped to make changes to parameter CMOd.

See diagram on page 31 for more information on the source of the drive's operation commands.

0 / 1 / 2

- ☐ Control terminal logic inputs.
- ☐ Graphic display terminal.
- ☐ Serial communication

1 Mode ► AUF ► 4*▲ ► cmod ►ENT ► edit value ► ENT to validate

FMOd	❑ **Remote Mode Primary Speed Reference Source**	-	1

The setting of parameter FMOd determines the source of the drive's speed reference when the drive is in remote mode.

The drive must be stopped to make changes to parameter FMOd.

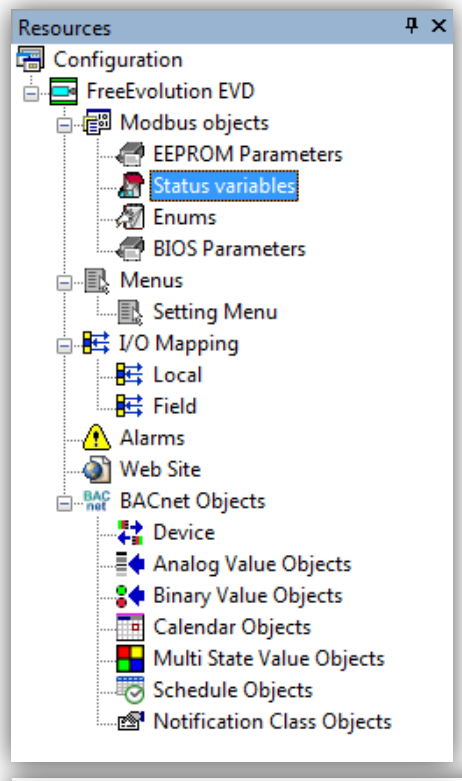
See diagram on page 31 for more information on the source of the drive's speed reference.

1 / 2 / 3 / 4 / 5

- ☐ VIA
- ☐ VIB
- ☐ Graphic display terminal
- ☐ Serial communication
- ☐ +/- Speed

3 Mode ► AUF ► 5*▲ ► fmod ►ENT ► edit value ► ENT to validate

Modbus link



- Developer must define a set of variables corresponding to what he wants to read or write via Modbus using Status Variables.

- In case of 32-bit Modbus registers to be read/write, developer should define 2 16-bit Status variable and merge them later.

Registers:

- Command= 64000+1 (W ►FC16)
- Frequency = 64001+1 (W►FC16)
- Output Frequency = 64768+1 (R►FC03)

Message for ATV command:

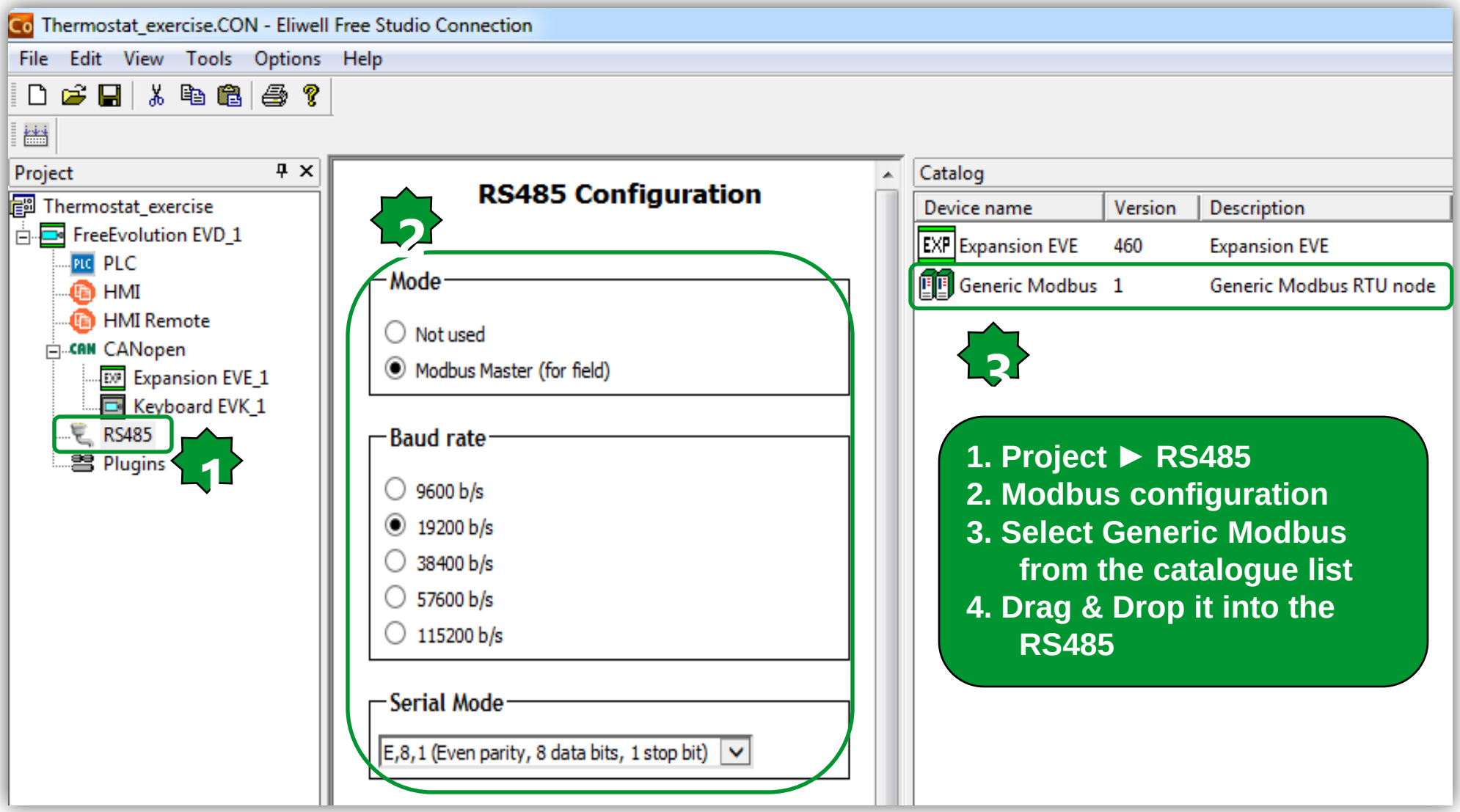
- Start Command= 50176
- Stop Command= 49152

FreeEvolution Status Variables

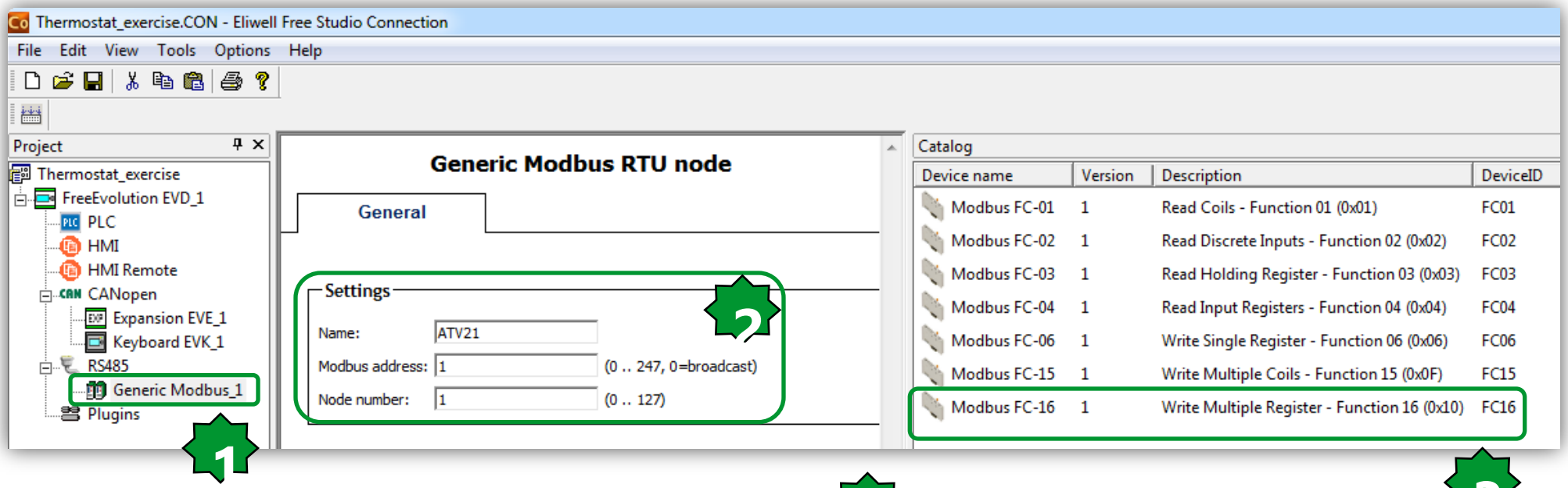
Add Remove Recalc

#	Address	Name	Device type	Application type	AccessLevel	Read only	Description
1	8960	Ambient_Temperature	Signed 16-bit	INT	Always visible	True	
2	8961	ATV_Command	Signed 16-bit	INT	Always visible	False	
3	8962	ATV_Speed_Reference	Signed 16-bit	INT	Always visible	False	
4	8963	ATV_Output_Frequency	Signed 16-bit	INT	Always visible	True	

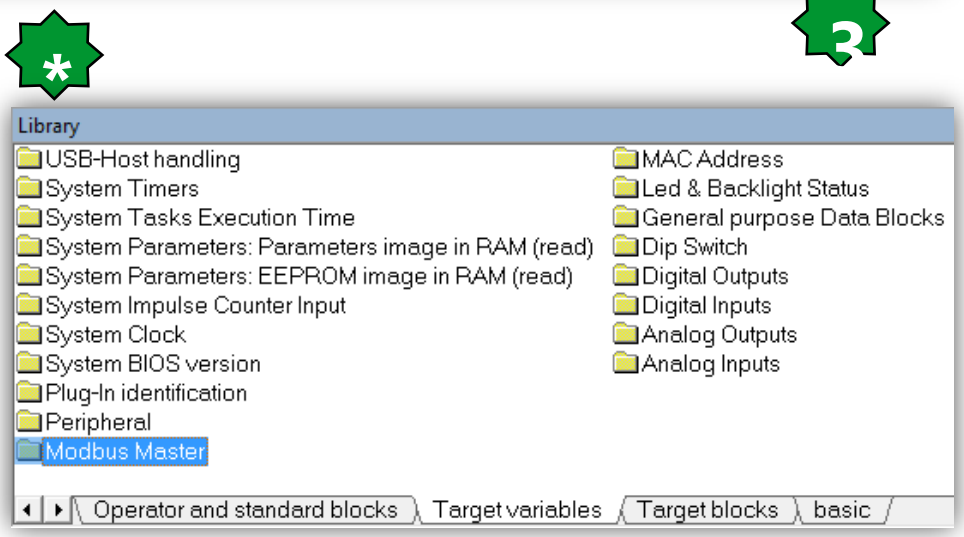
Modbus Master Configuration



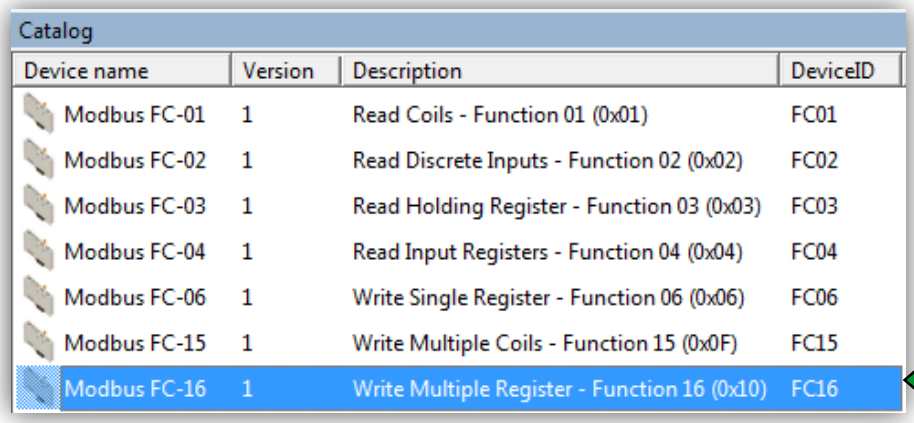
Generic Modbus

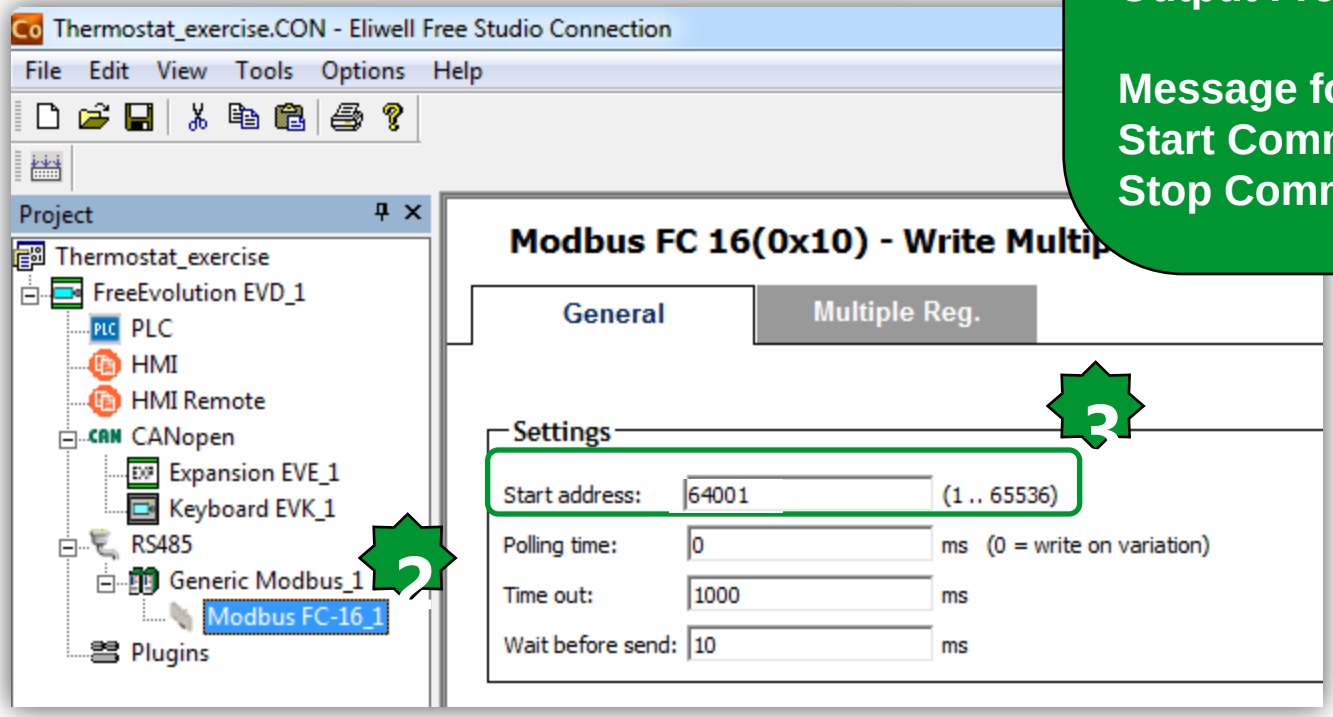


1. After drag & drop
2. Name it & define the Modbus address.
It is recommended to set the Node number the same as the Modbus address
3. Select the desired function code from catalogue
* Note: Vectors sysMbRtu* in the folder Modbus Master are addressed by node number



Modbus Function Code



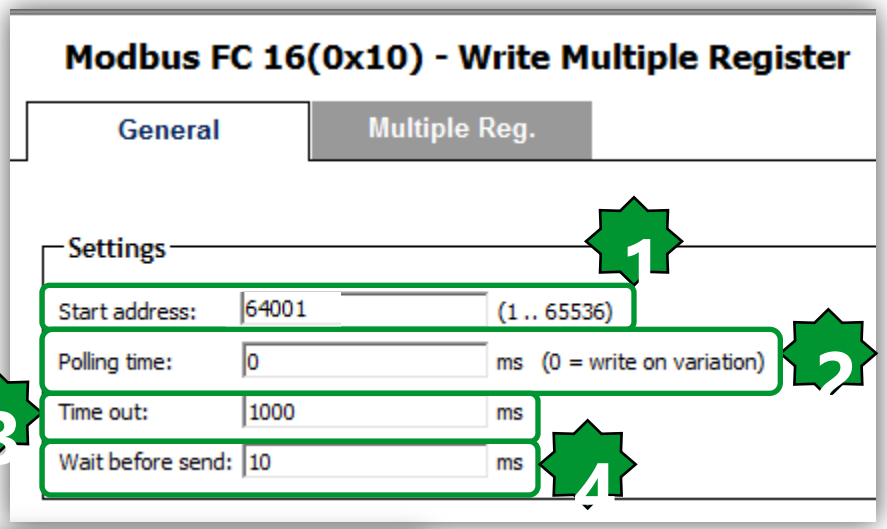


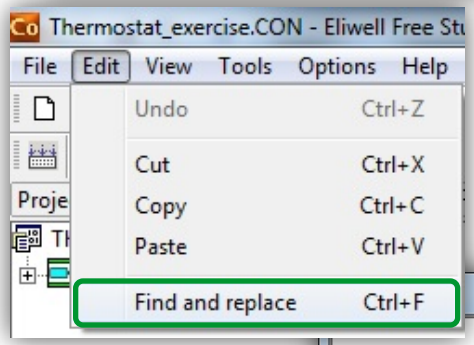
1. Select the required function code
2. Drag & drop it into the Project
► RS485 ►Generic Modbus 1
3. Do the general settings
Registers:
Command= 64000+1 (W ➔ FC16)
Frequency = 64001+1 (W ➔ FC16)
Output Frequency = 64768+1 (R ➔FC03)

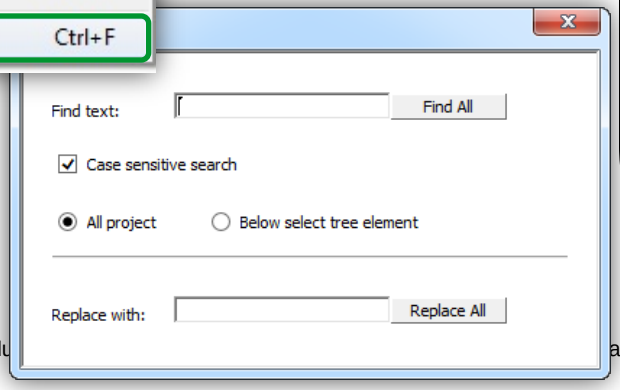
Message for ATV command:
Start Command= 50176
Stop Command= 49152

Important: Some slave devices requires an offset of one to register Address: ATV requires it, Evolution doesn‘t need it.

Modbus Function Code/Settings







1. Start address: address of the first modbus object to read or write (1..65536).

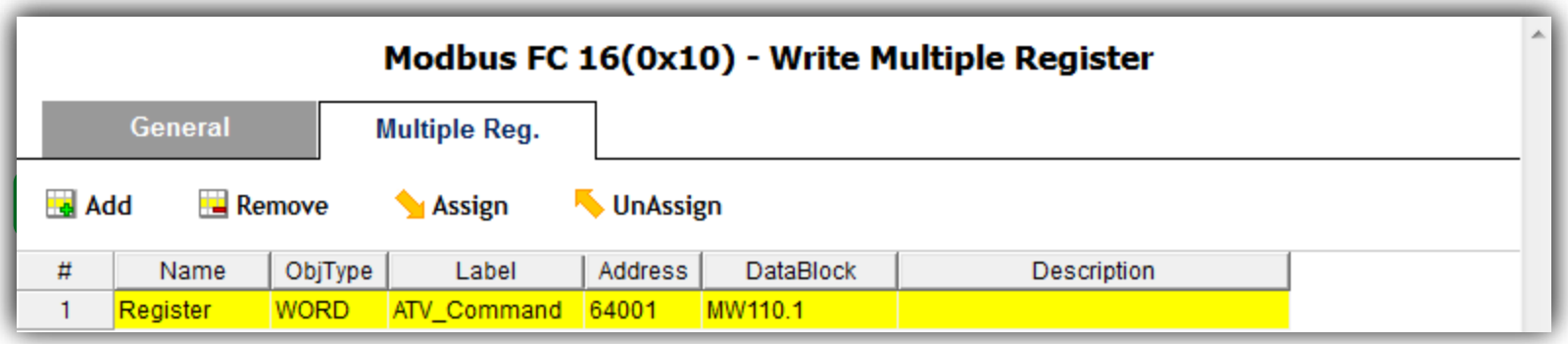
2. Polling time: minimum waiting period between 2 message processing (ms); for writing operations, 0 means to write it only on variation of the value, for reading operations 0 means maximum speed.

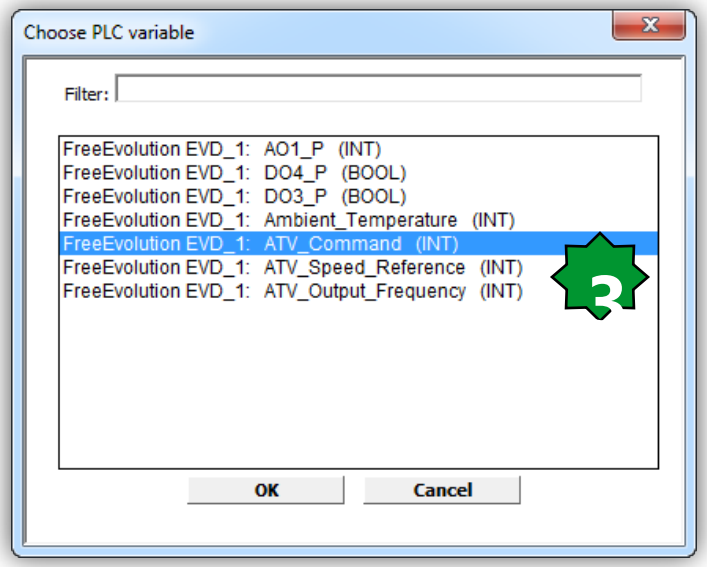
3. Timeout: the operation will fail when this time-out expires (ms).

4. Wait before send: Waiting time after end of previous message response (suggested time >=10ms).

Note: Follow below order in Modbus:
1. Write (FC16)
2. Read (FC3)

Multiple Reg.





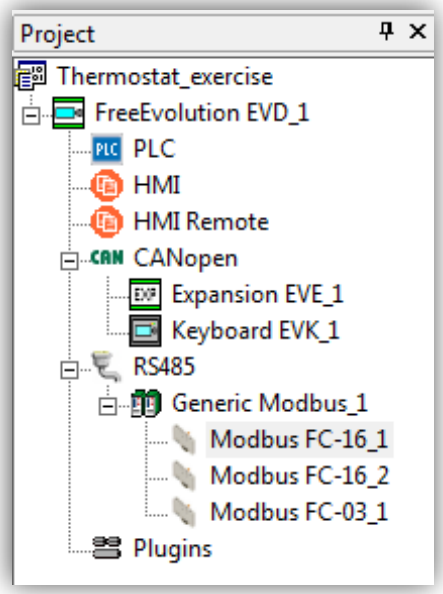
1. **Press Add/Remove in order to define how many registers should be written**
2. **Assign**
3. **Choose PLC variable ► OK**

Note:

- ATV does not support more than 1 read/ write register with the same message

- Waiting time after end of previous message response (suggested time >=10ms)

Generic Modbus\Register



If a status variable is used to write a value on variation and to read the same value, the related write message must be listed in Connection before the corresponding read message

Modbus FC 16(0x10) - Write Multiple Register

General | Multiple Reg.

Add | Remove | Assign | UnAssign

#	Name	ObjType	Label	Address	DataBlock	Description
1	Register	WORD	ATV_Command	64001	MW110.1	

Modbus FC 16(0x10) - Write Multiple Register

General | Multiple Reg.

Add | Remove | Assign | UnAssign

#	Name	ObjType	Label	Type	Address	DataBlock	Description
1	Register	WORD	ATV_Output_Frequency	INT	64002	MW110.3	

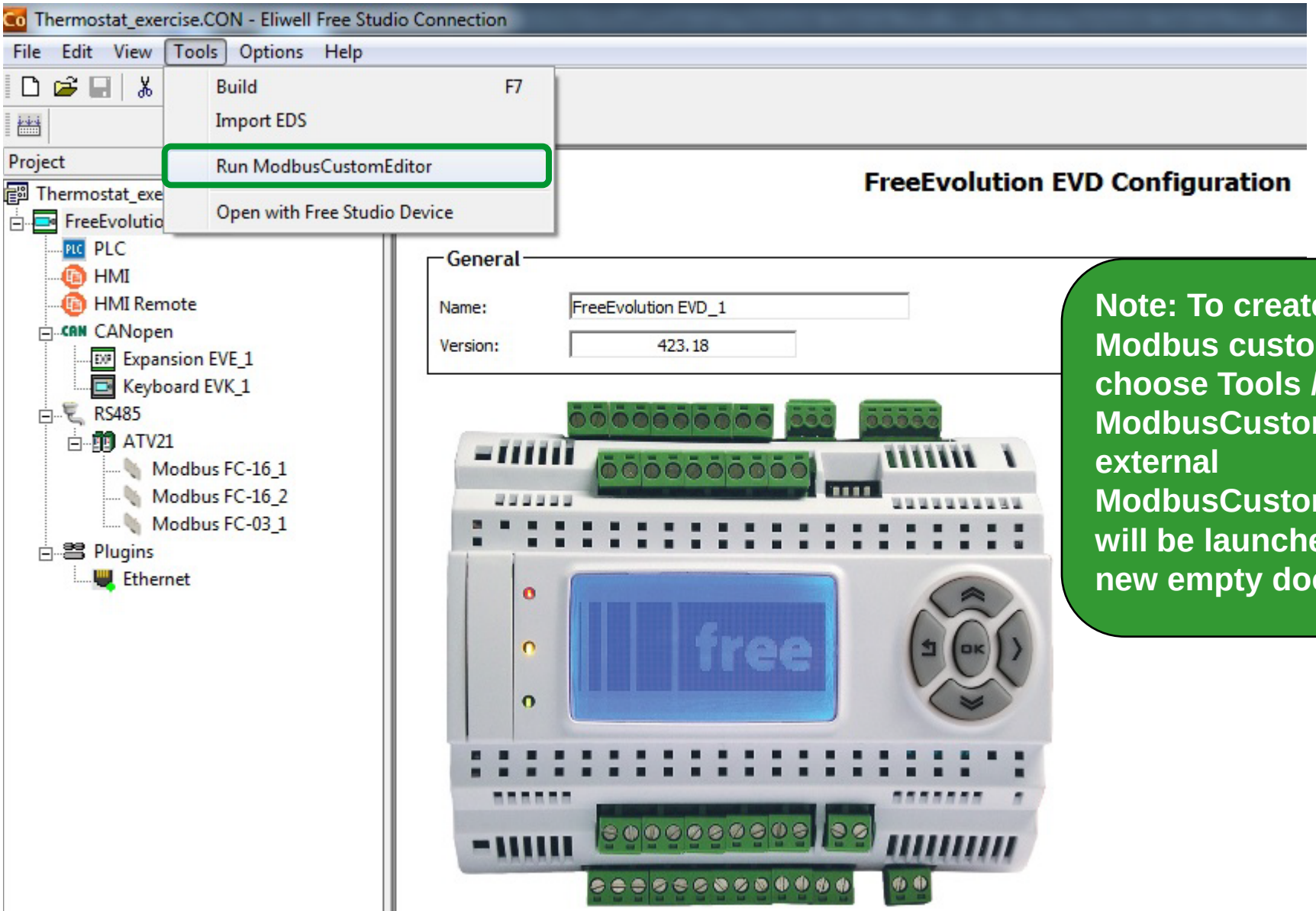
Modbus FC 03(0x03) - Read Holding Register

General | Holding Reg.

Add | Remove | Assign | UnAssign

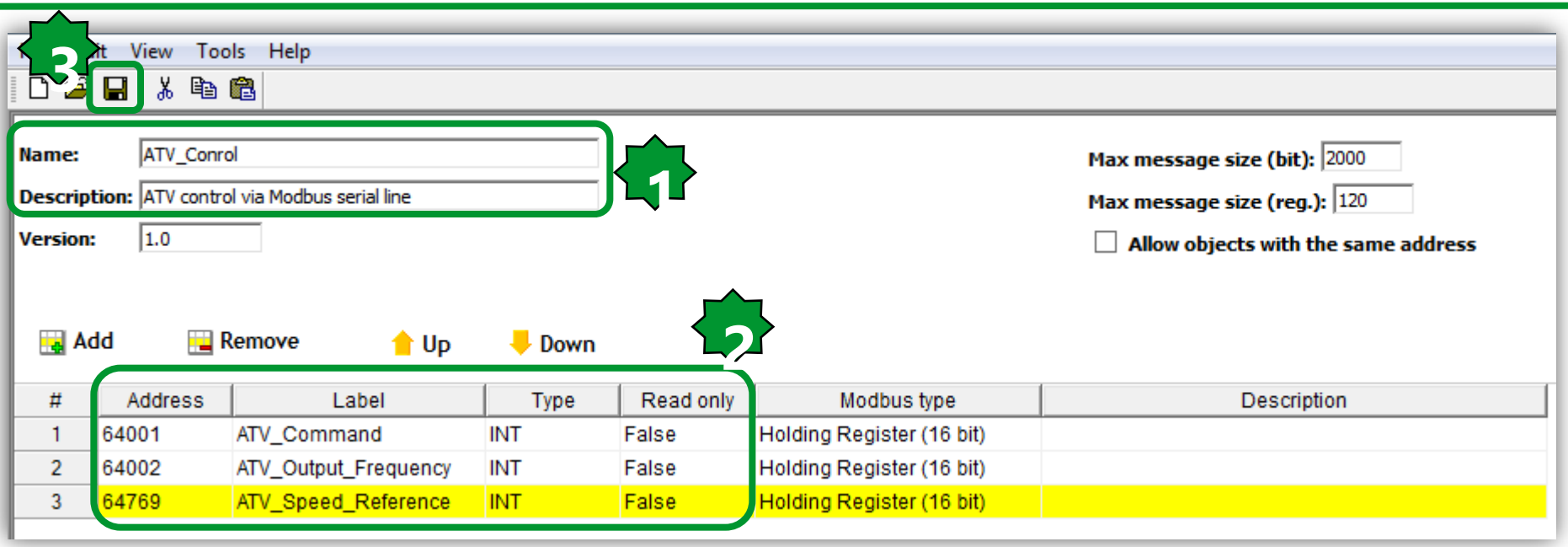
#	Name	ObjType	Label	Type	Address	DataBlock	Description
1	Register	WORD	ATV_Output_Frequency	INT	64769	MW110.3	

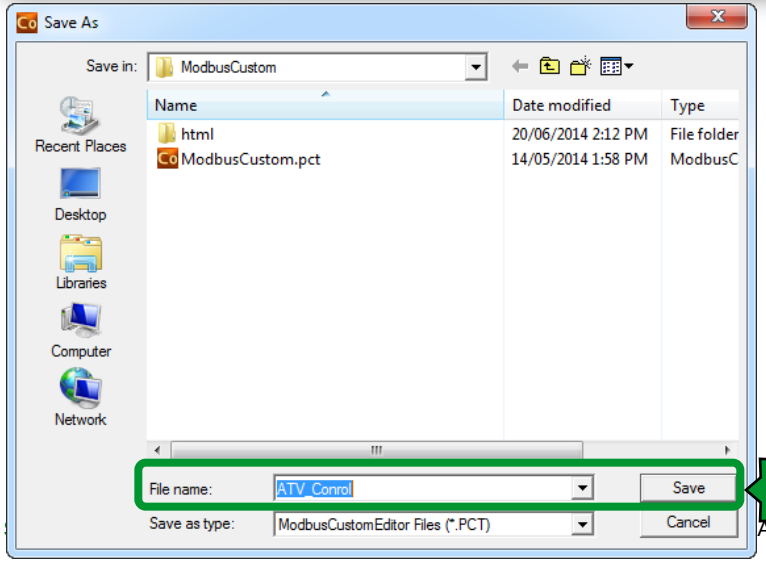
…Creating a new Modbus custom device



Note: To create a new Modbus custom device, choose Tools / Run ModbusCustomEditor; the external ModbusCustomEditor tool will be launched, with a new empty document.

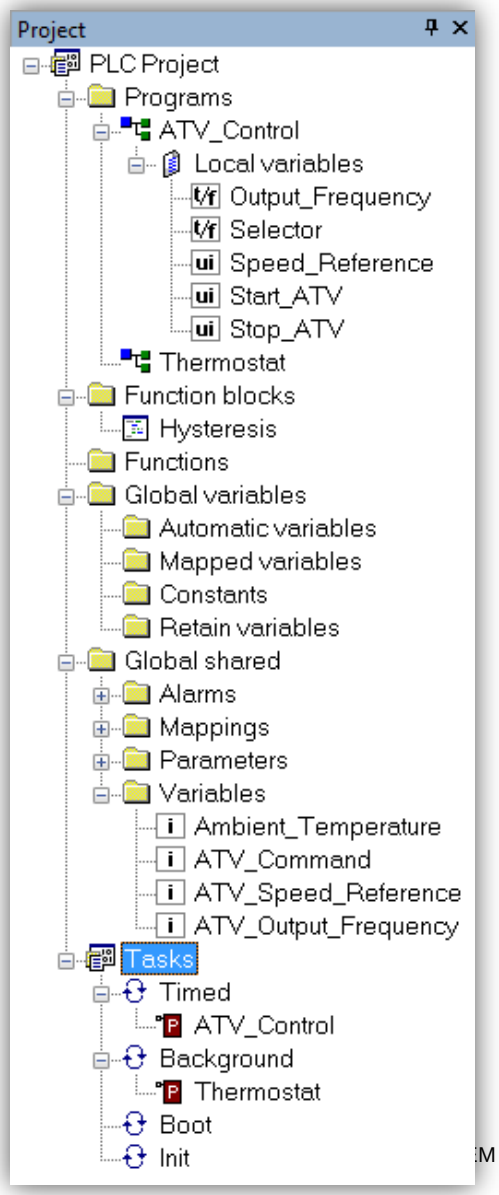
Creating a new Modbus custom device…





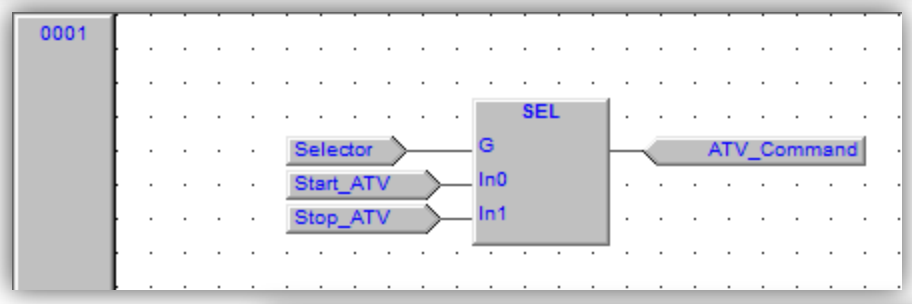
Note:
Advantage: Easier to be used in Connection
Disadvantage: Message are not optimized & common polling time for all messages

ATV control, Local variable definition

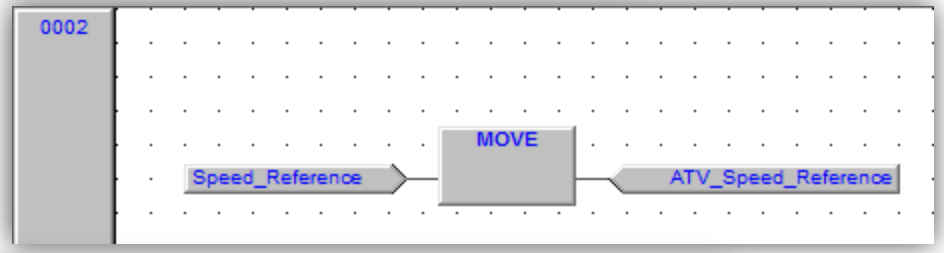


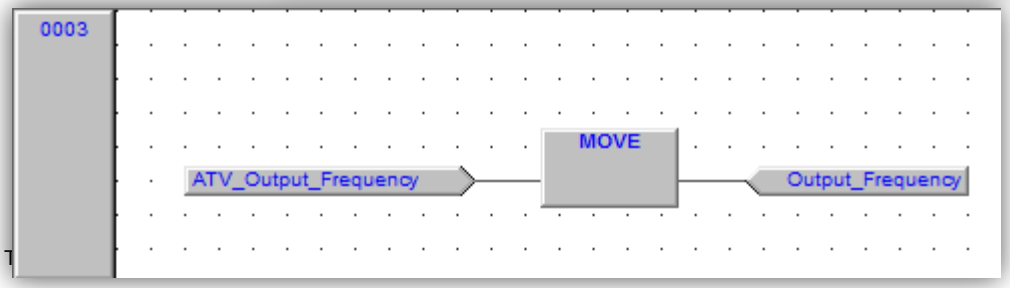
Local variables

	Name	Type	Address	Array	Init value	Attribute
1	Selector	BOOL	Auto	No	False	..
2	Speed_Reference	INT	Auto	No		..
3	Output_Frequency	INT	Auto	No		..
4	Start_ATV	UINT	Auto	No	50176	CONSTANT
5	Stop_ATV	UINT	Auto	No	49152	CONSTANT

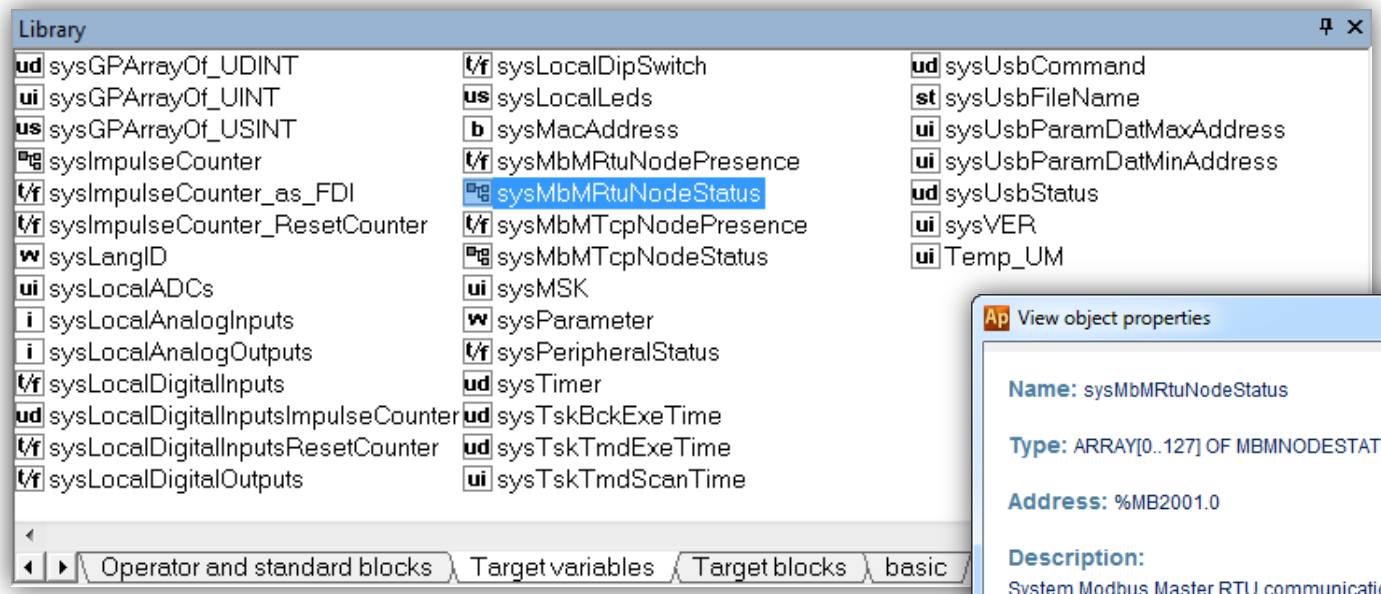


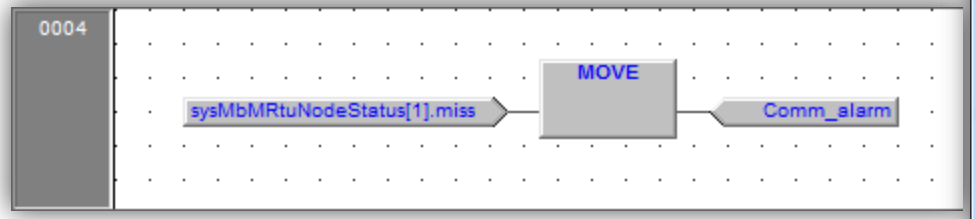
Note: Two independant group of blocks must be place in two separate networks



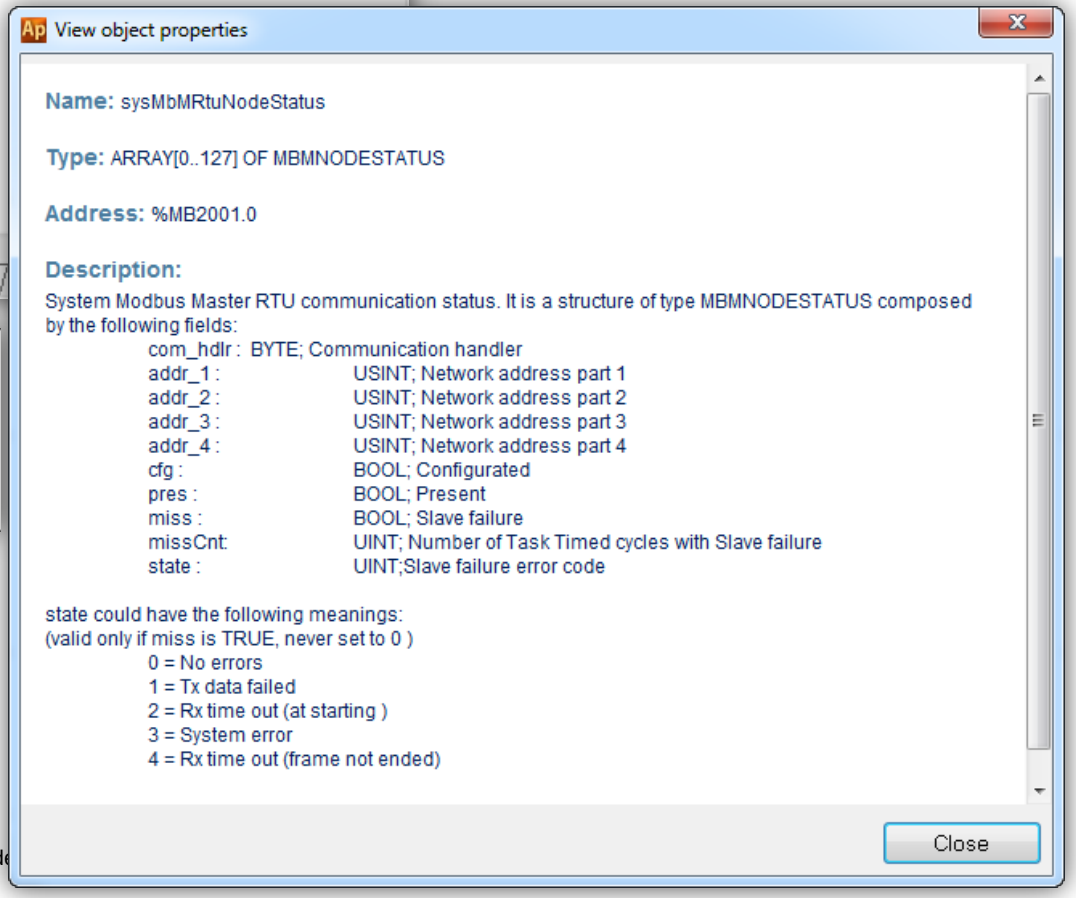


Modbus Communication Alarm

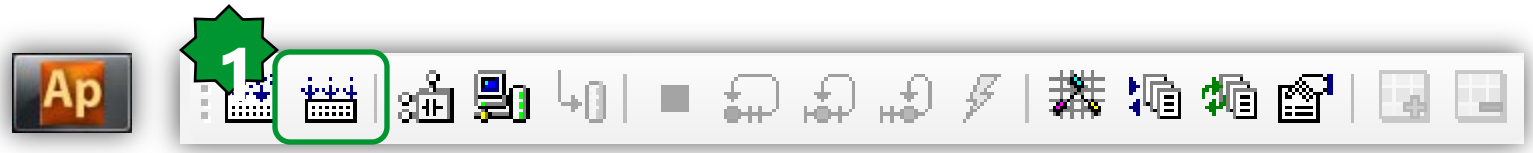


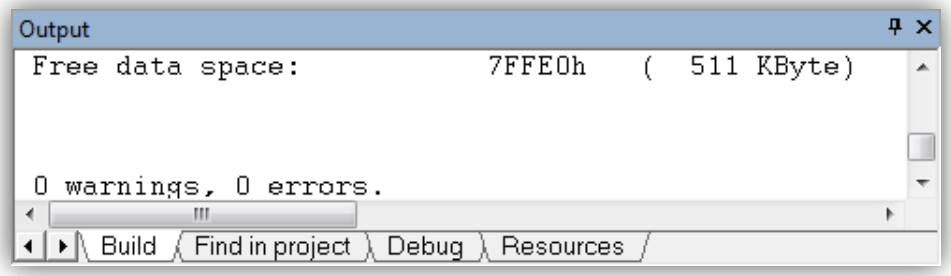


See next chapter for download details

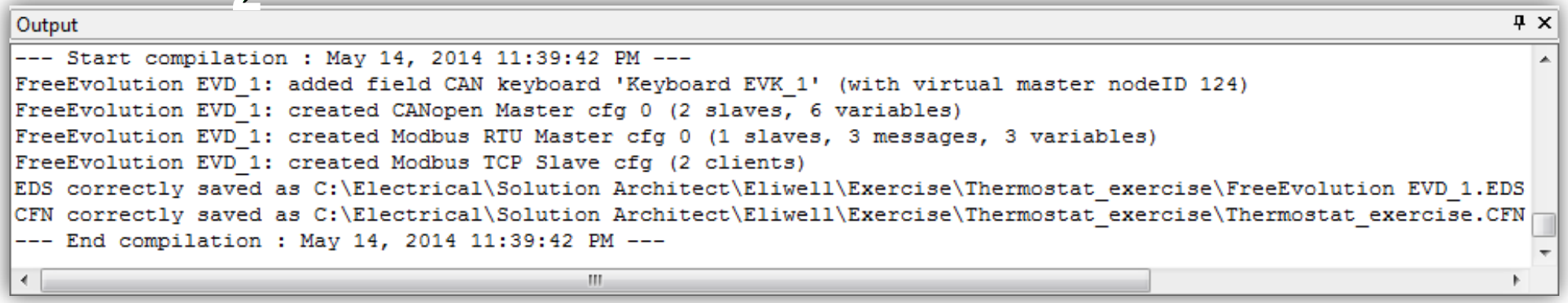


Recompile & Build

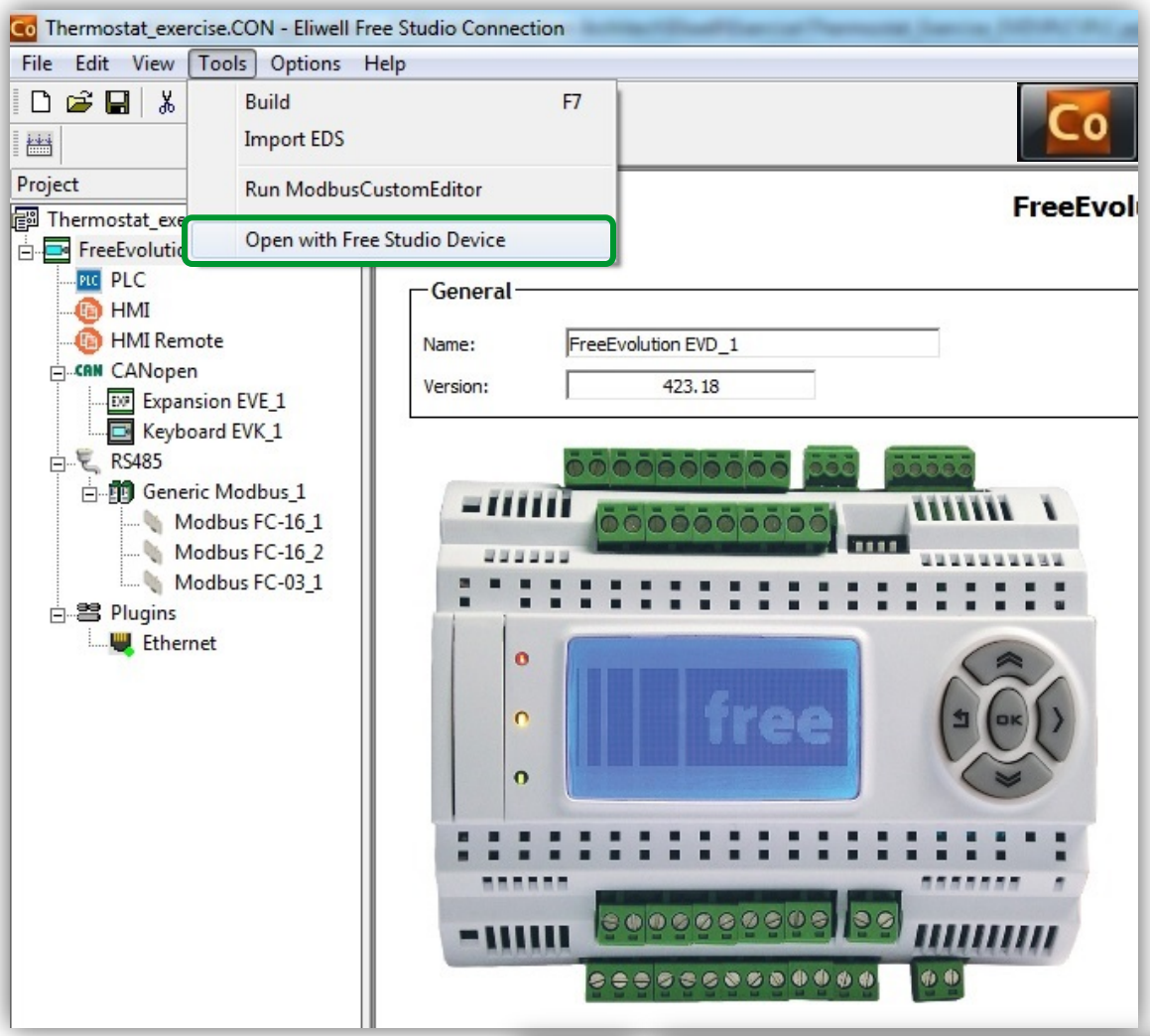




Note:
To apply the changes to the network, free studio asks you to reboot.
Yes: if you want to validate it.
Cancel: if you want to dismiss



Open with free studio device



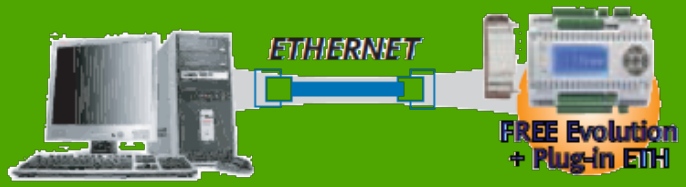
See next chapter for further details on download via TCP

Chapter 13

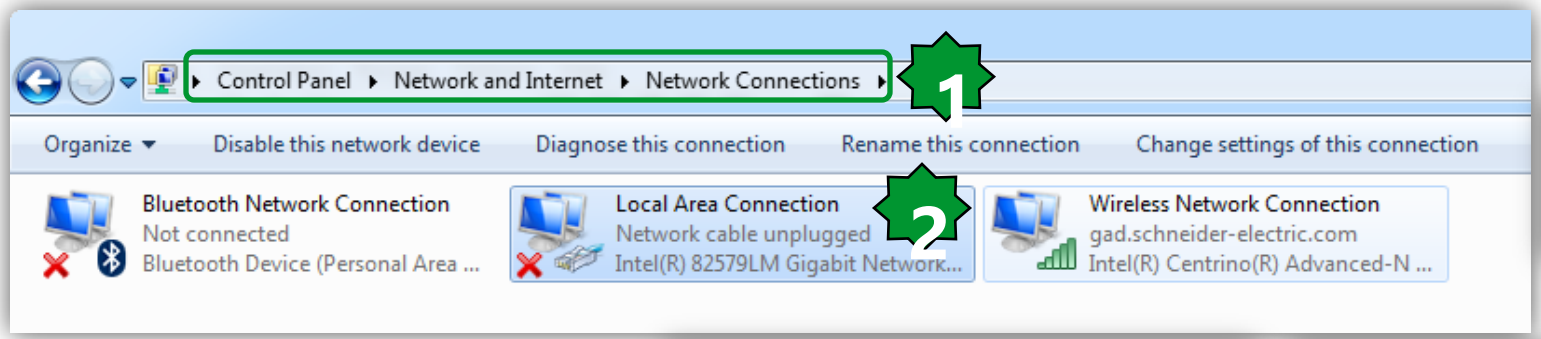
Modbus TCP

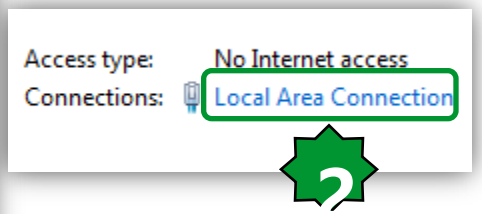
Goal:

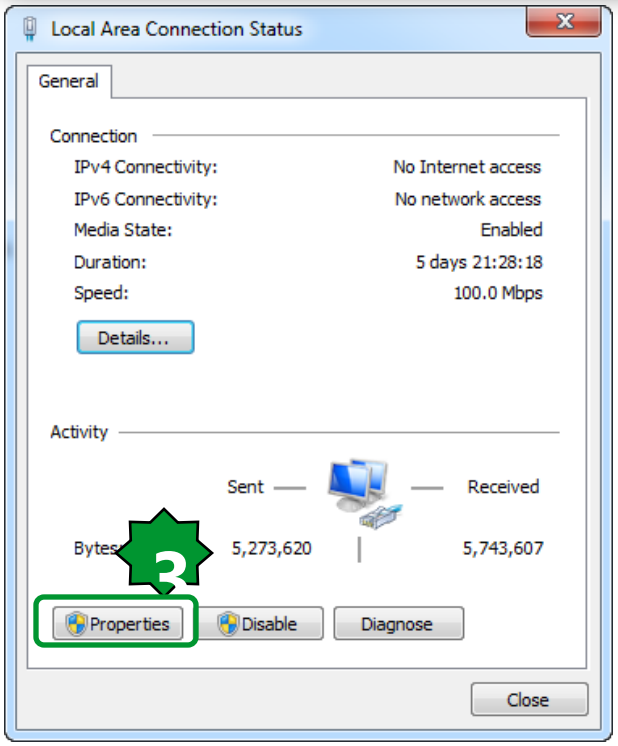
Modbus TCP configuration, project download and socket management

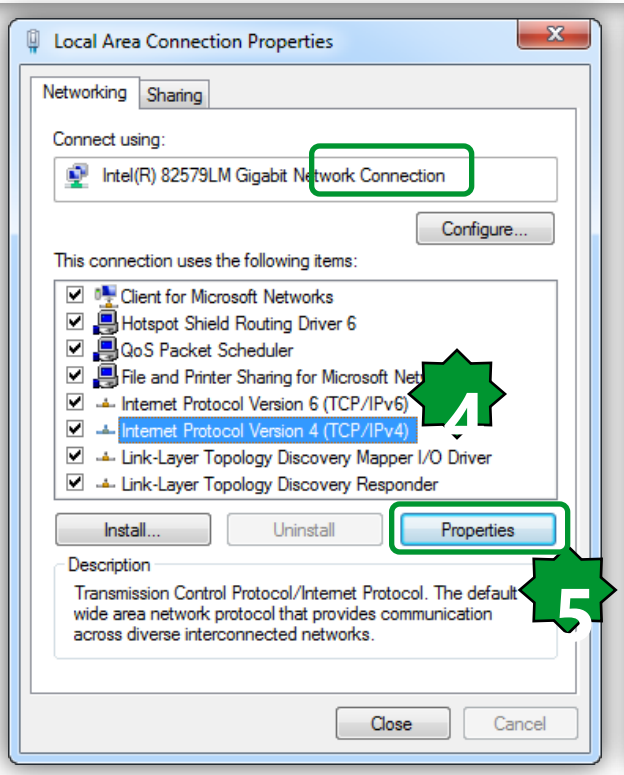


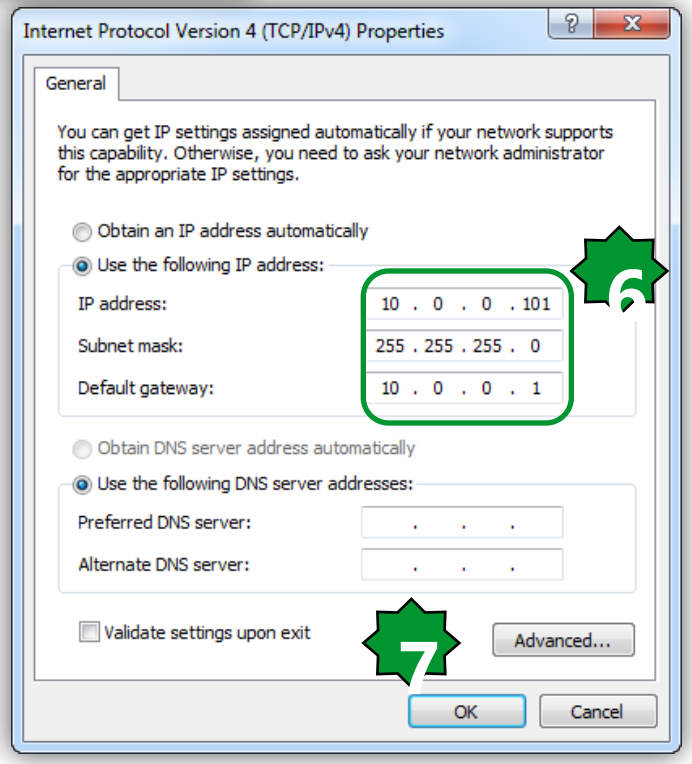
PC TCP/IP configuration



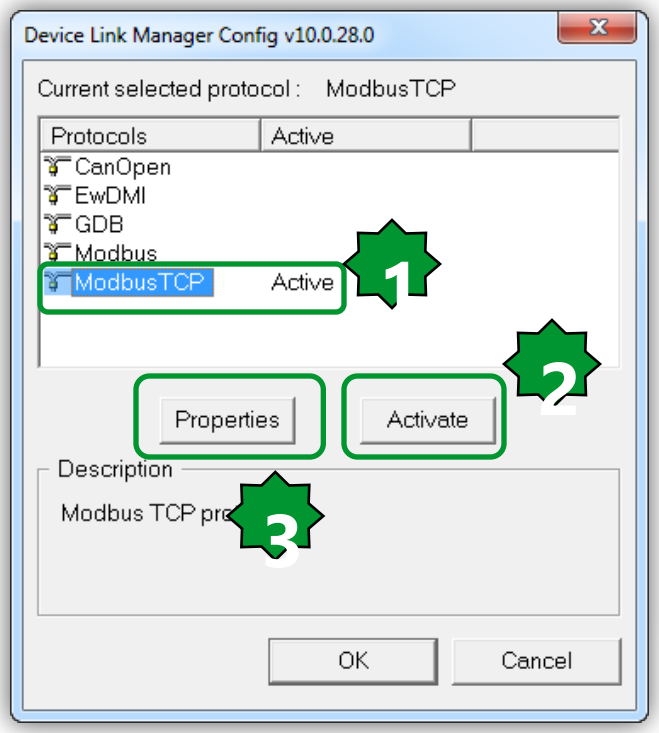


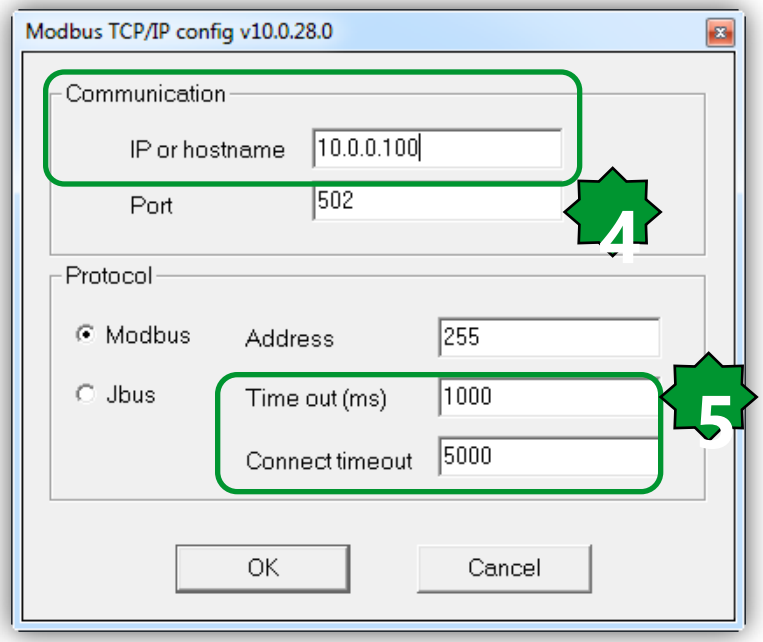




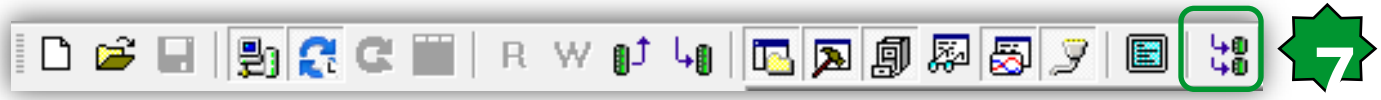


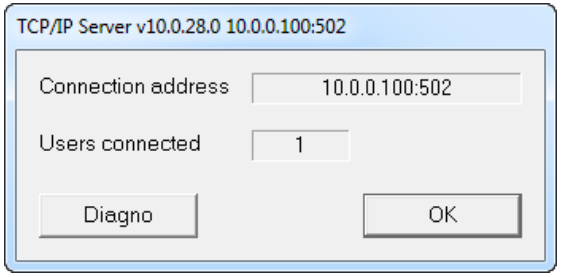
Modbus TCP/download



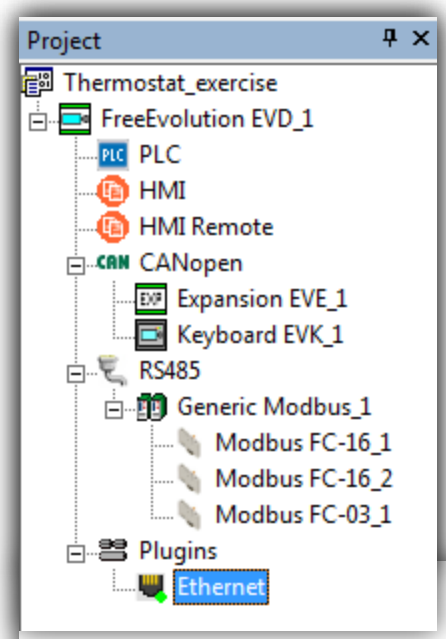


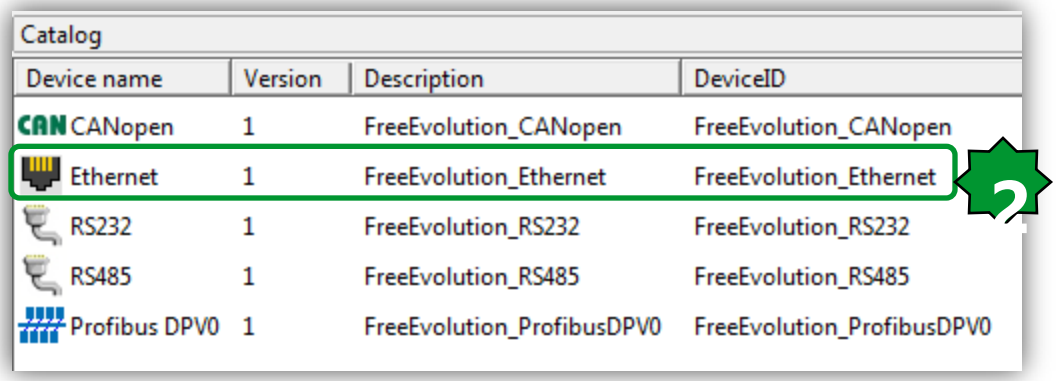
5.change Timeout and Connect Timeout based on the Ethernet band available. If the connection it is not direct it is better to increase both of them 10000-20000

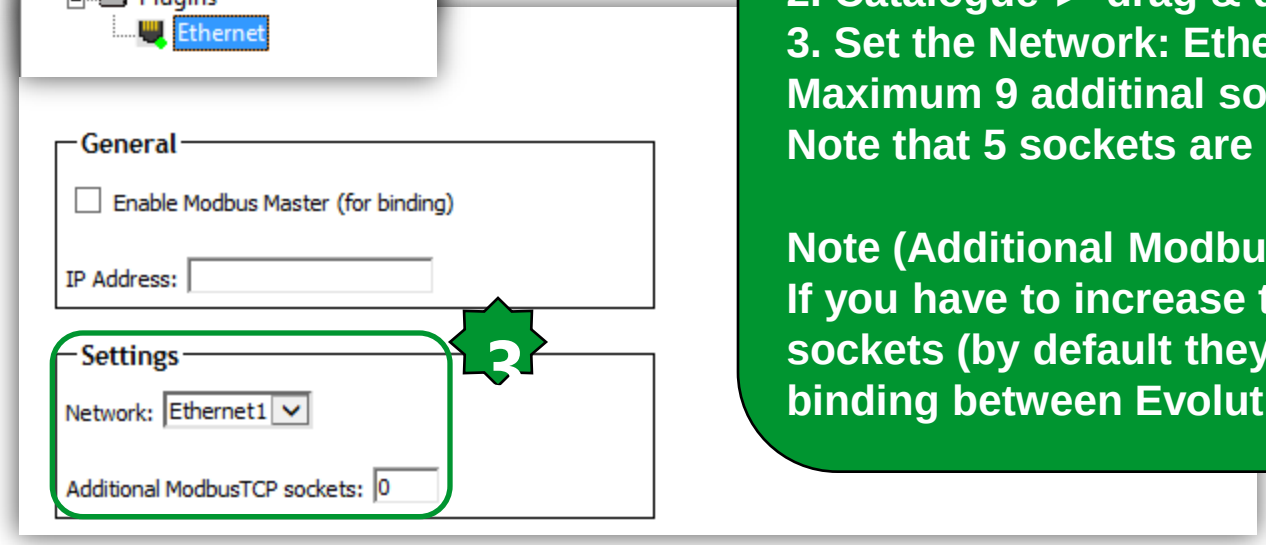




Modbus TCP socket



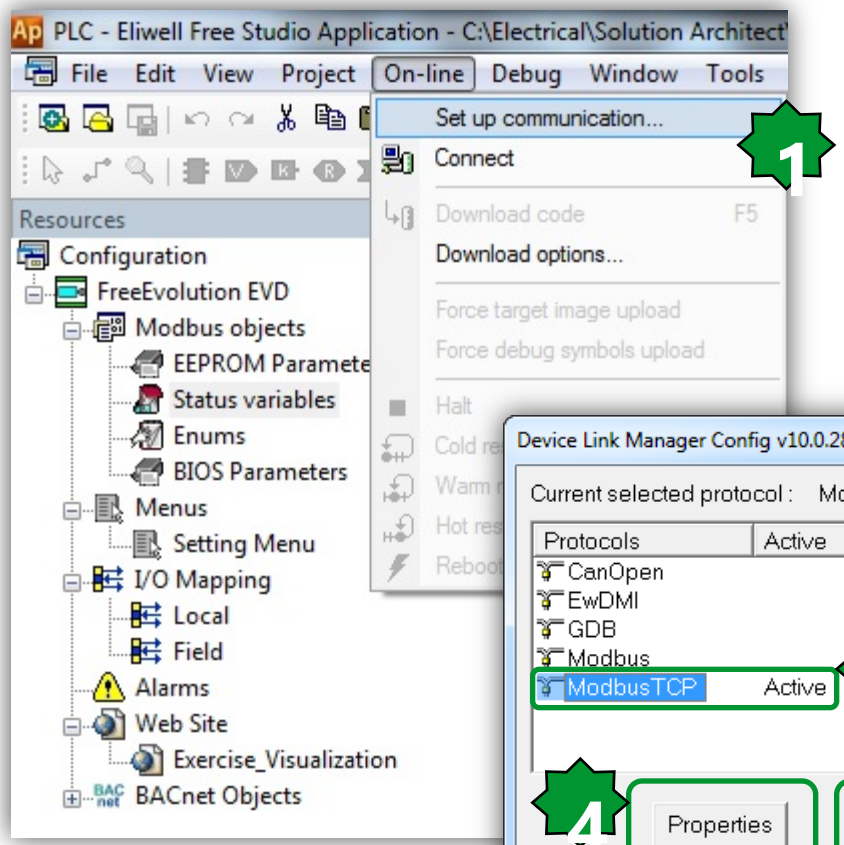


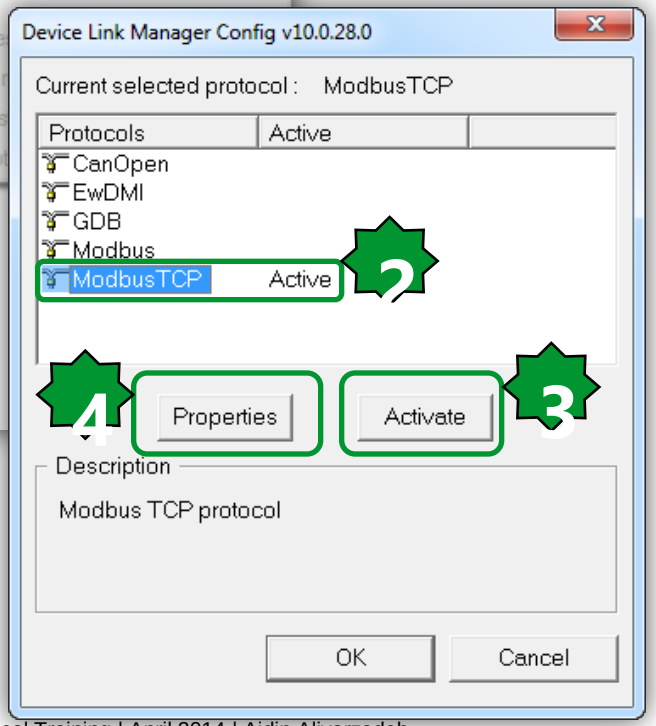


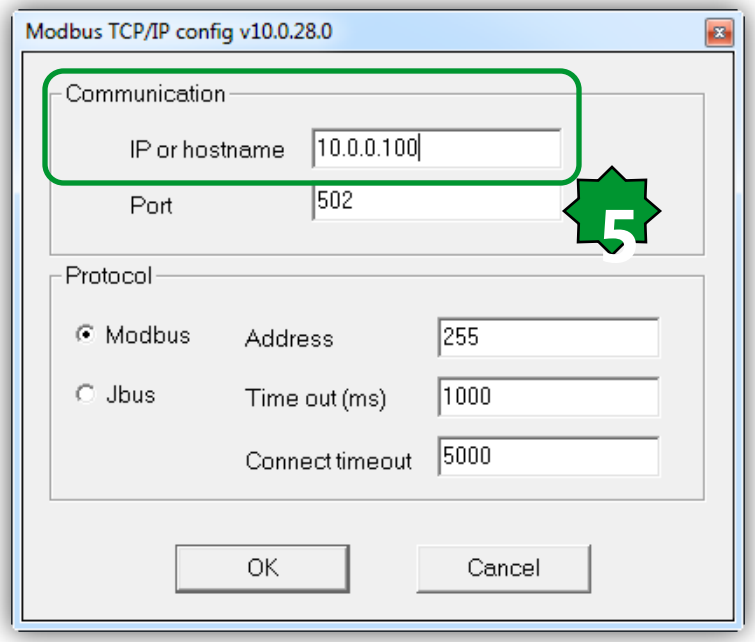
1.Plugins
2. Catalogue ► drag & drop Ethernet into the plugins part
3. Set the Network: Ethernet1
Maximum 9 additinal sockets(+2 opened by default).
Note that 5 sockets are related toWebserver

Note (Additional Modbus TCP sockets):
If you have to increase the number of Modbus/TCP sockets (by default they are 3) or if you have to implement binding between Evolution.

Modbus TCP/debugging







Chapter 14

Modbus Slave

Goal:

Configuration of Free Studio to creat connection between Vijeodesigner & Evolution

Modbus Slave

- Status variables and EEPROM parameters have a modbus address and they are all Holding Registers, regardless the type of variable defined into Device Type

FreeEvolution Status Variables

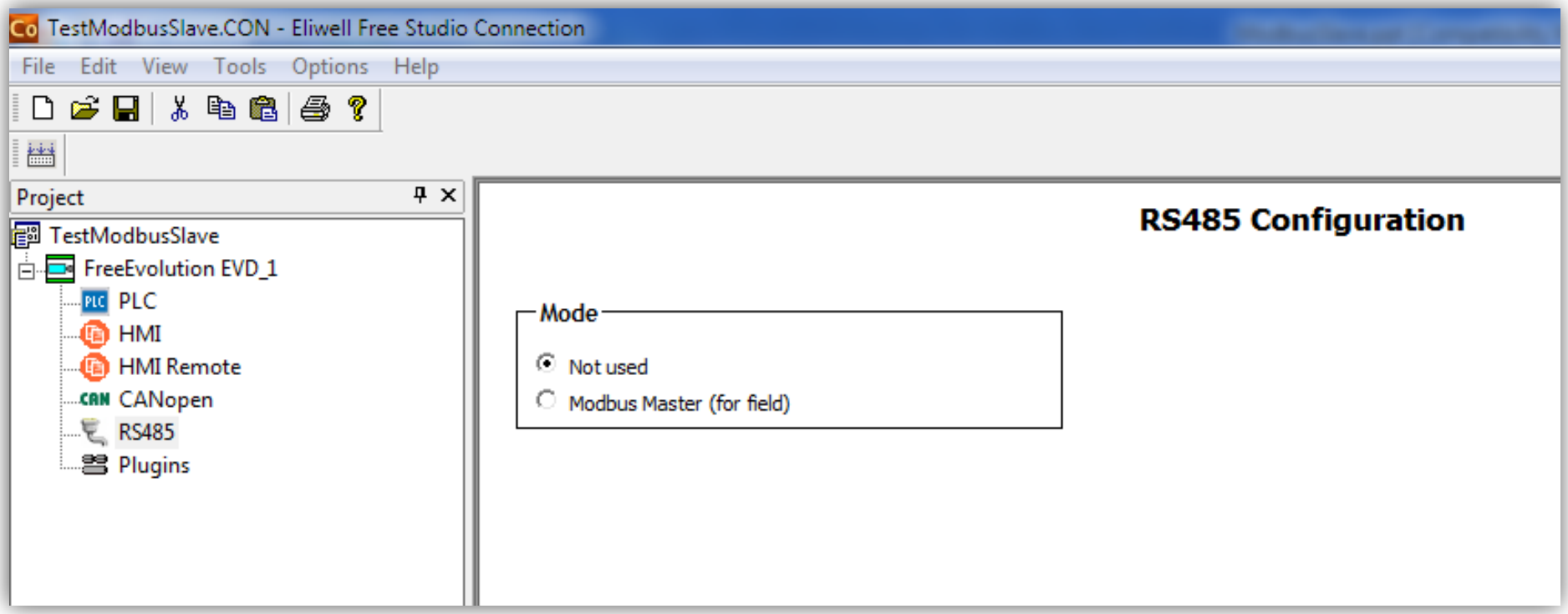
Add Remove Recalc

#	Address	Name	Device type	Application type	Size	Offset	Unit	AccessLevel	Read only
1	8960	TestWord	Unsigned 16-bit	UINT		0		Always visible	False
2	8961	TestBit	Boolean	BOOL		0		Always visible	False

- EEPROM parameters are always R/W
- Status Variables are RO by default
  - Set to False Read Only in case of R/W Status Variable

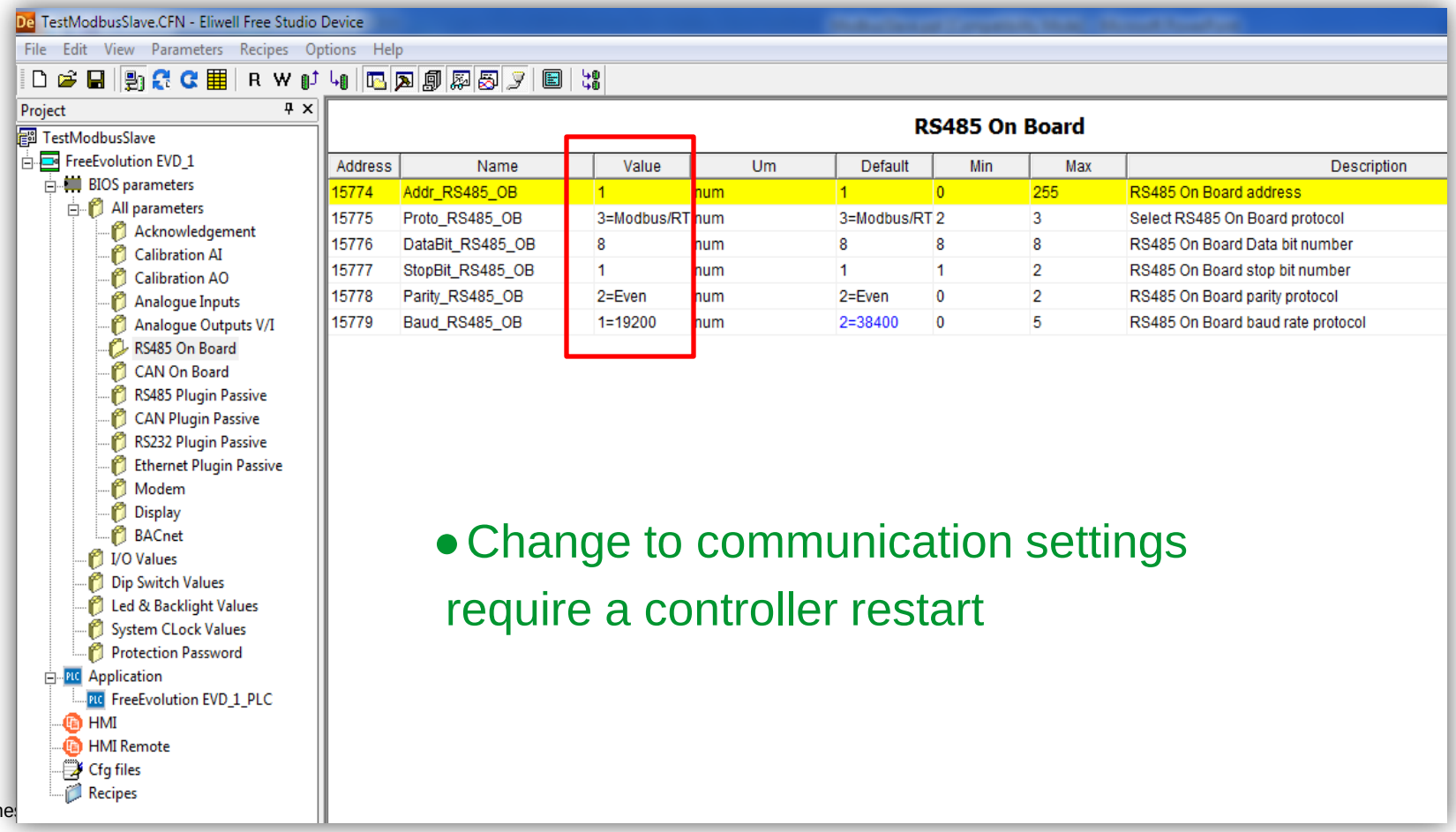
Modbus Slave

- In Free Studio Connection, set the Mode of RS485 of the controller to Not used
  - ➔ It means that the RS485 on board is configured as a slave port



Modbus Slave

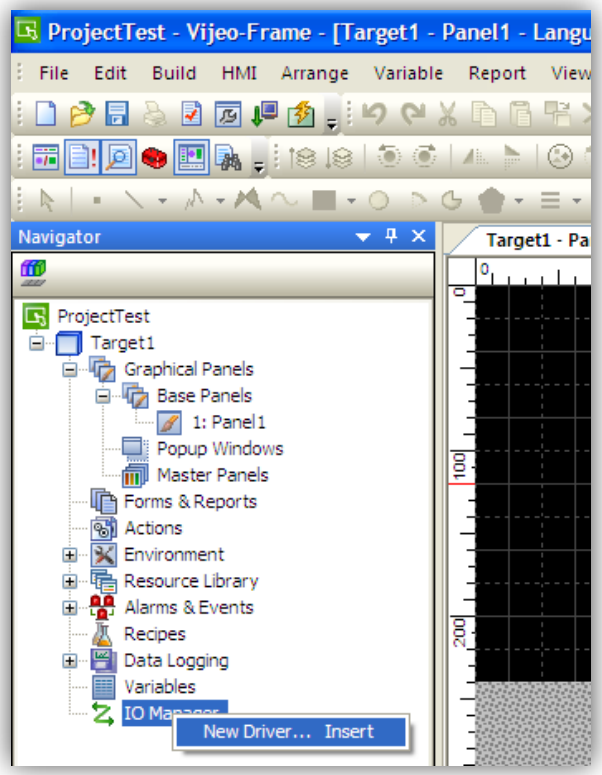
- In Free Studio Device, in Bios parameters, click on RS485 On Board
- Configure the Modbus communication:
  - ➔ Address of the controller
  - ➔ Baudrate
  - ➔ Stop bits
  - ➔ Parity



- Change to communication settings require a controller restart

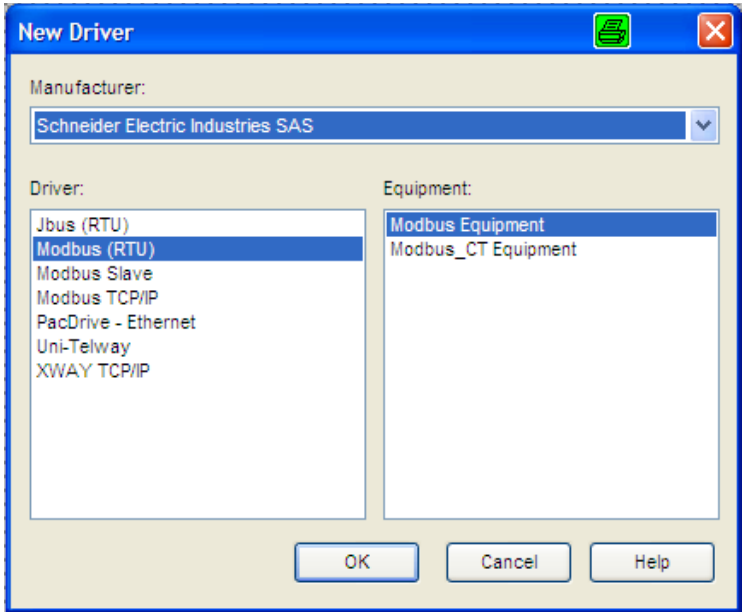
How to connect Freeway to VijeoDesigner target

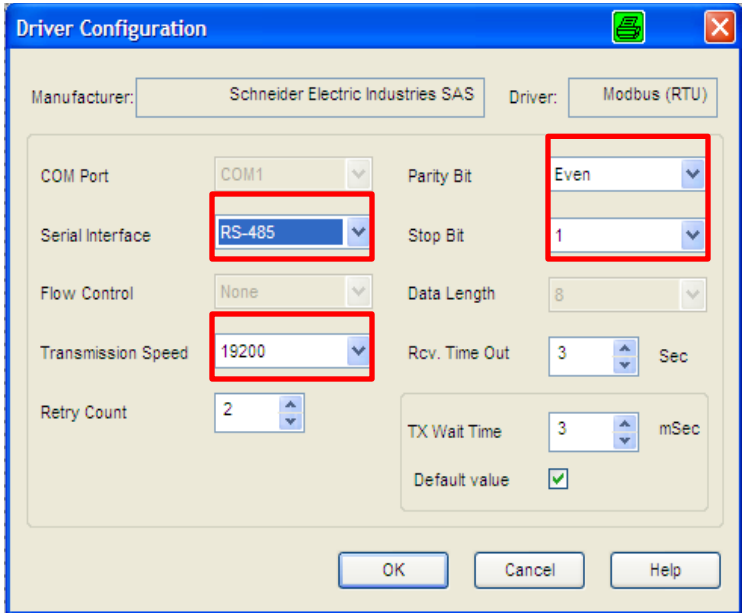
- Create a new driver

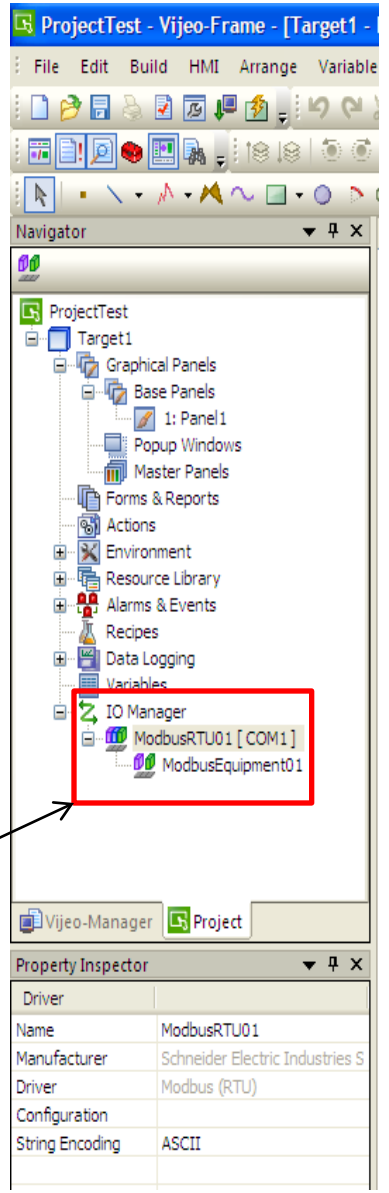


Vijeo Designer

- Configure the driver as configured in Free Studio



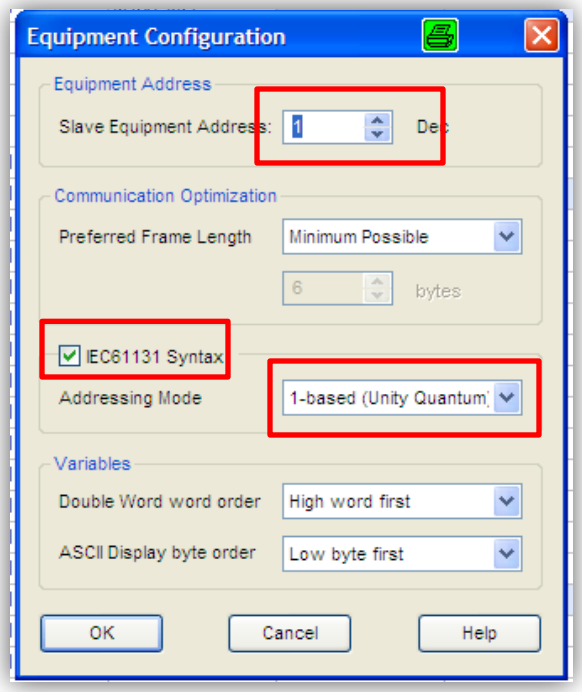


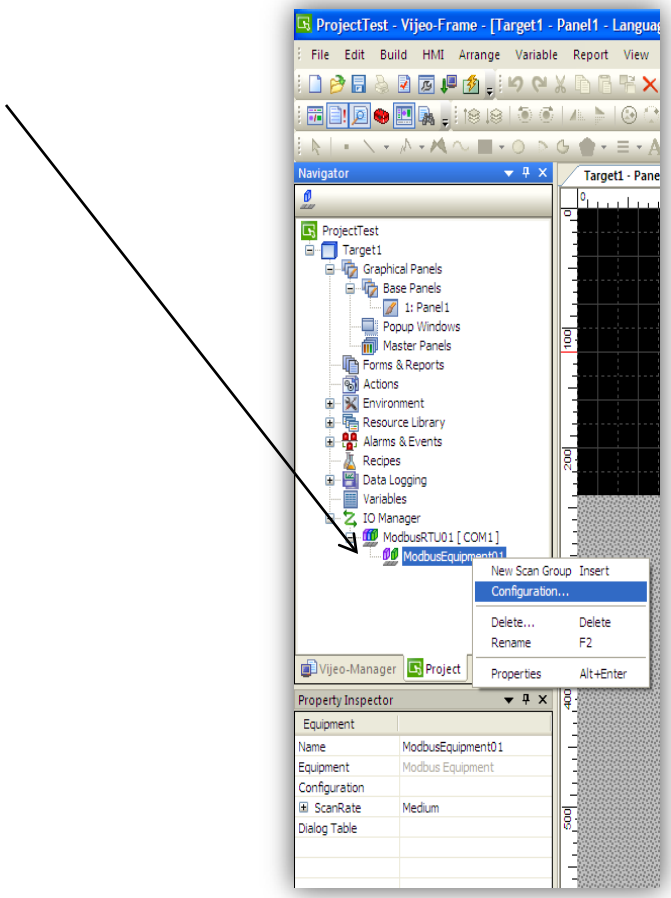


- A new Modbus equipment has been created

Vijeo Designer

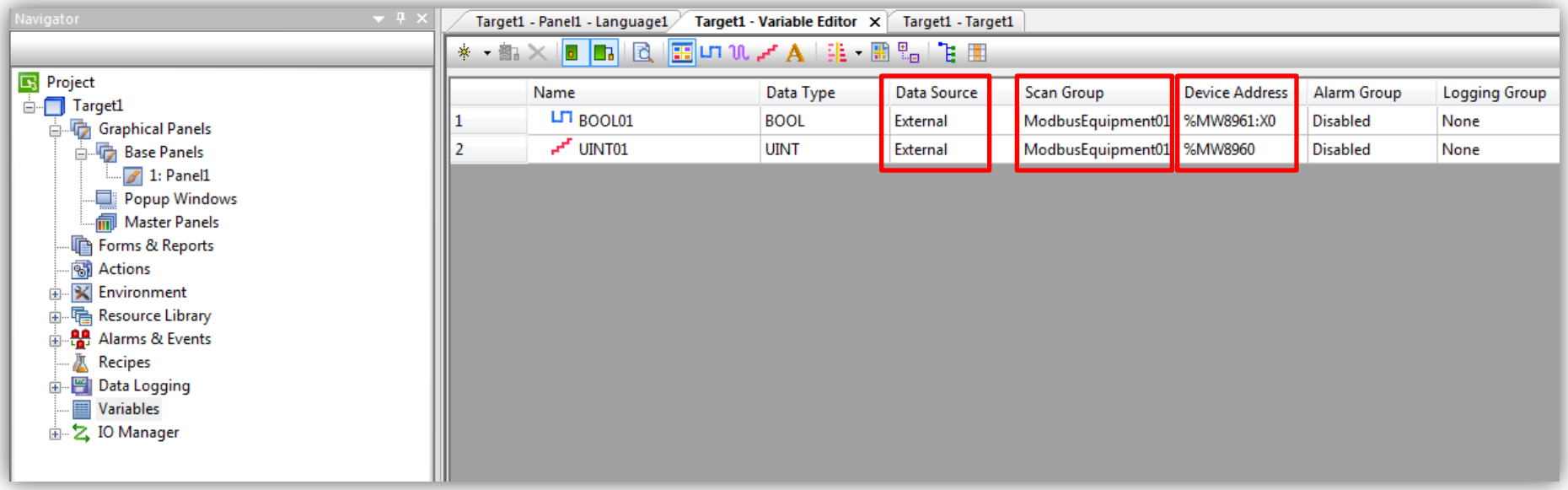
- Configure the Modbus equipment (right click):
  - Enter the address as set in Free Studio
  - Select IEC61131 Syntax
  - Adressing mode: 1-based (Unity Quantum)





Vijeo Designer

- Create your variables:



  - Define the Data Source: External
  - Defin the Scan Group: name of your Modbus equipment you have created
  - Specify the register address of the variable

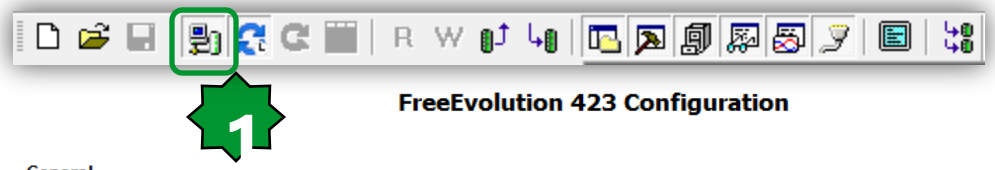
Chapter 15

Web server

Goal:

Embedded & customized web pages

Web visualization



FreeEvolution 423 Configuration

General

Name: FreeEvolution EVD

File version: 423.18

Communication

Protocol: ModbusTCP — Settings

Address: 10.0.0.100

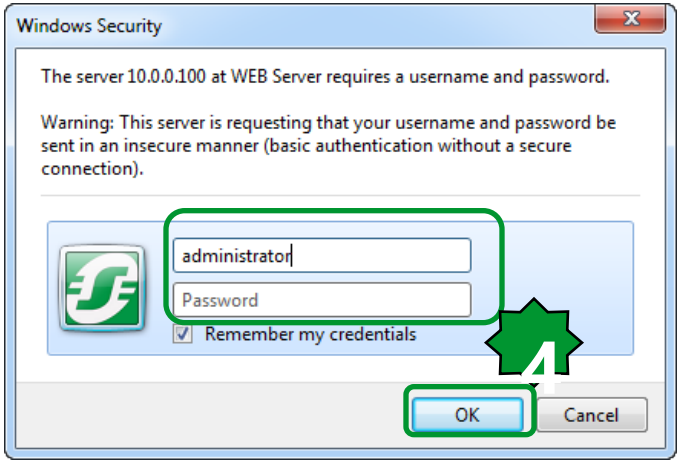
Port: TCPIP:502 — Disable communication

Baud rate:

Information

Status: CONNECTED

Firmware version: 423.16



Other operations

BIOS download

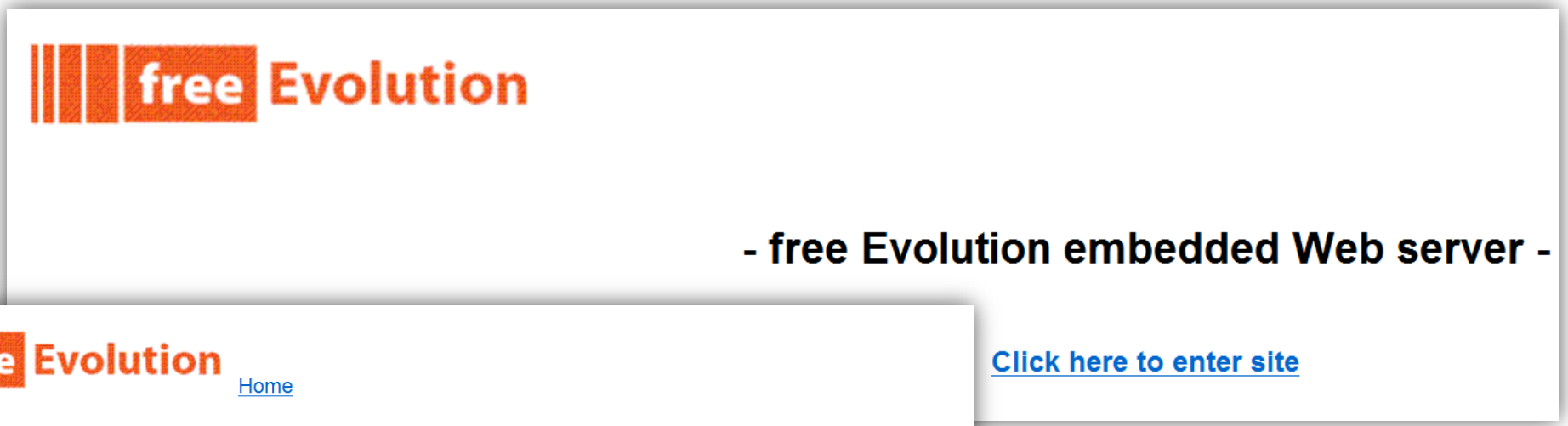
Open file browser

Web site download

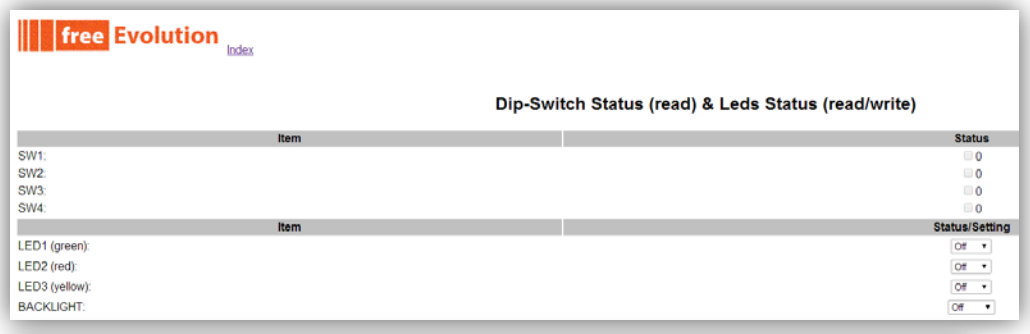
Web site preview

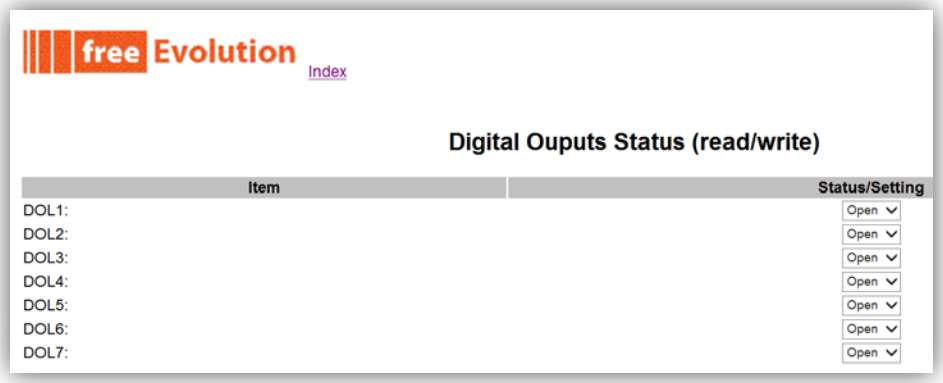
1. Connect to the target ► FreeEvolution EVD_1
2. Open internet browser (Google Chorome)
3. Type 10.0.0.100 in the address bar
4. In the windows security pop-up:
Default Username: administrator
Default Password: password
► OK

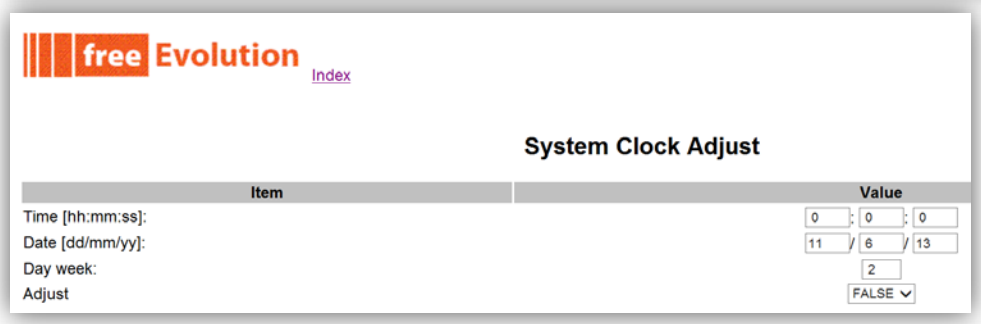
Embedded web pages

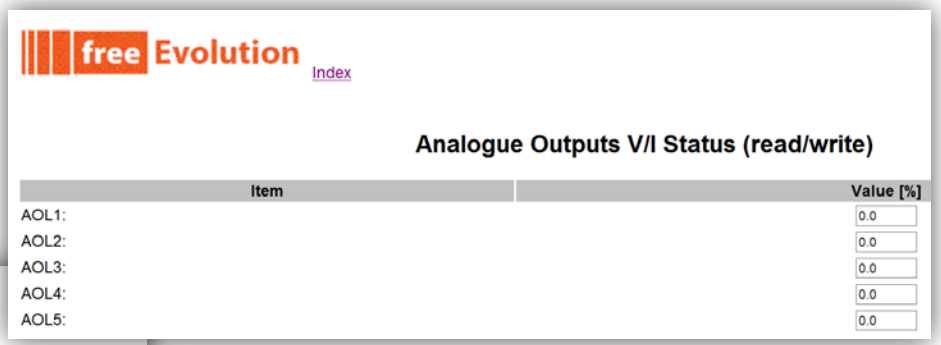


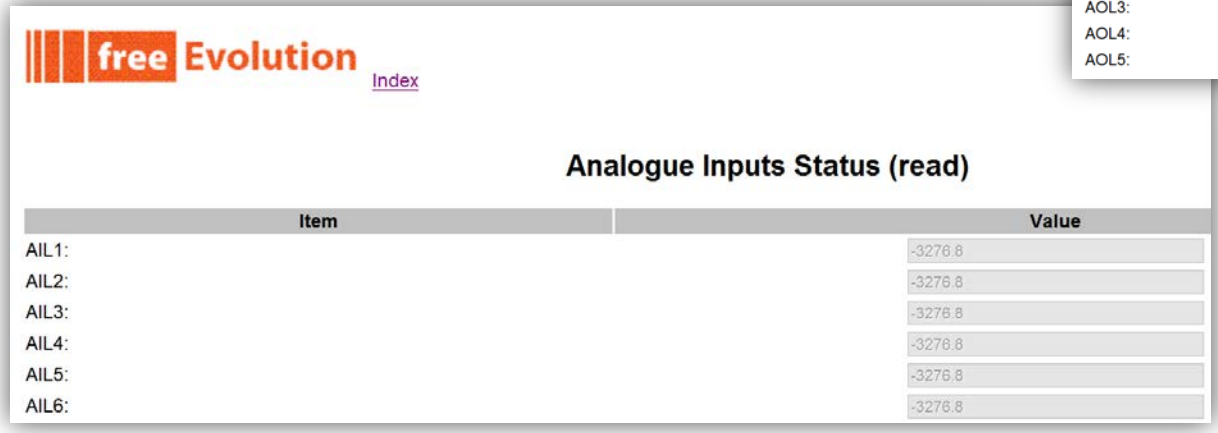
Embedded web pages

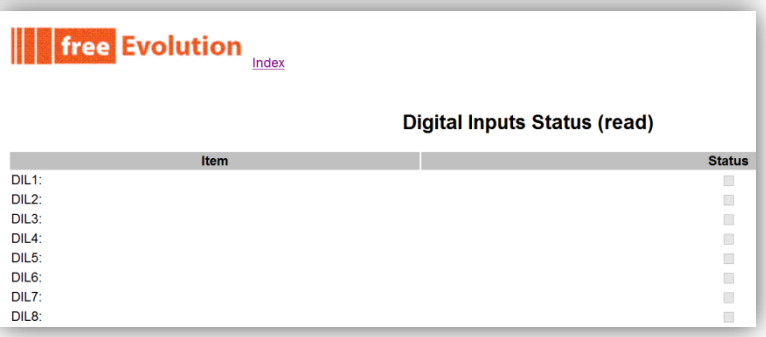




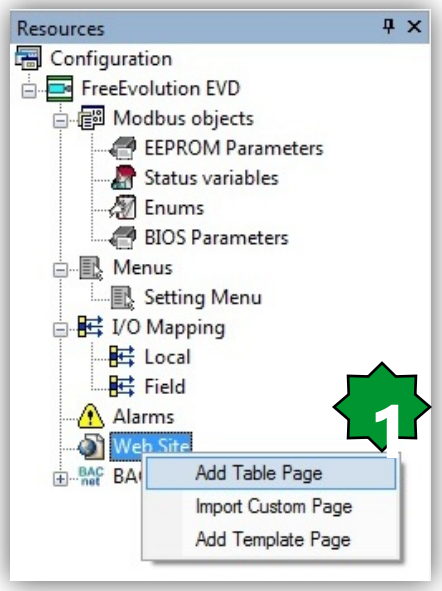


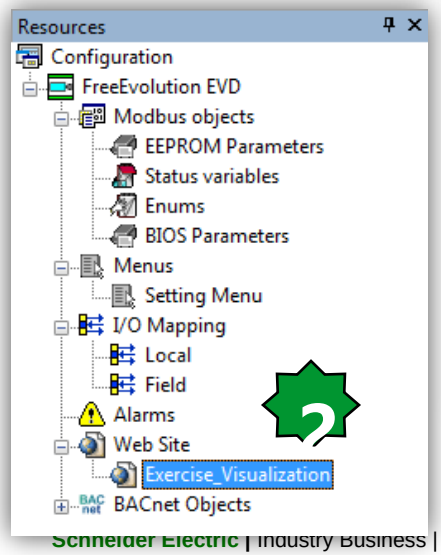


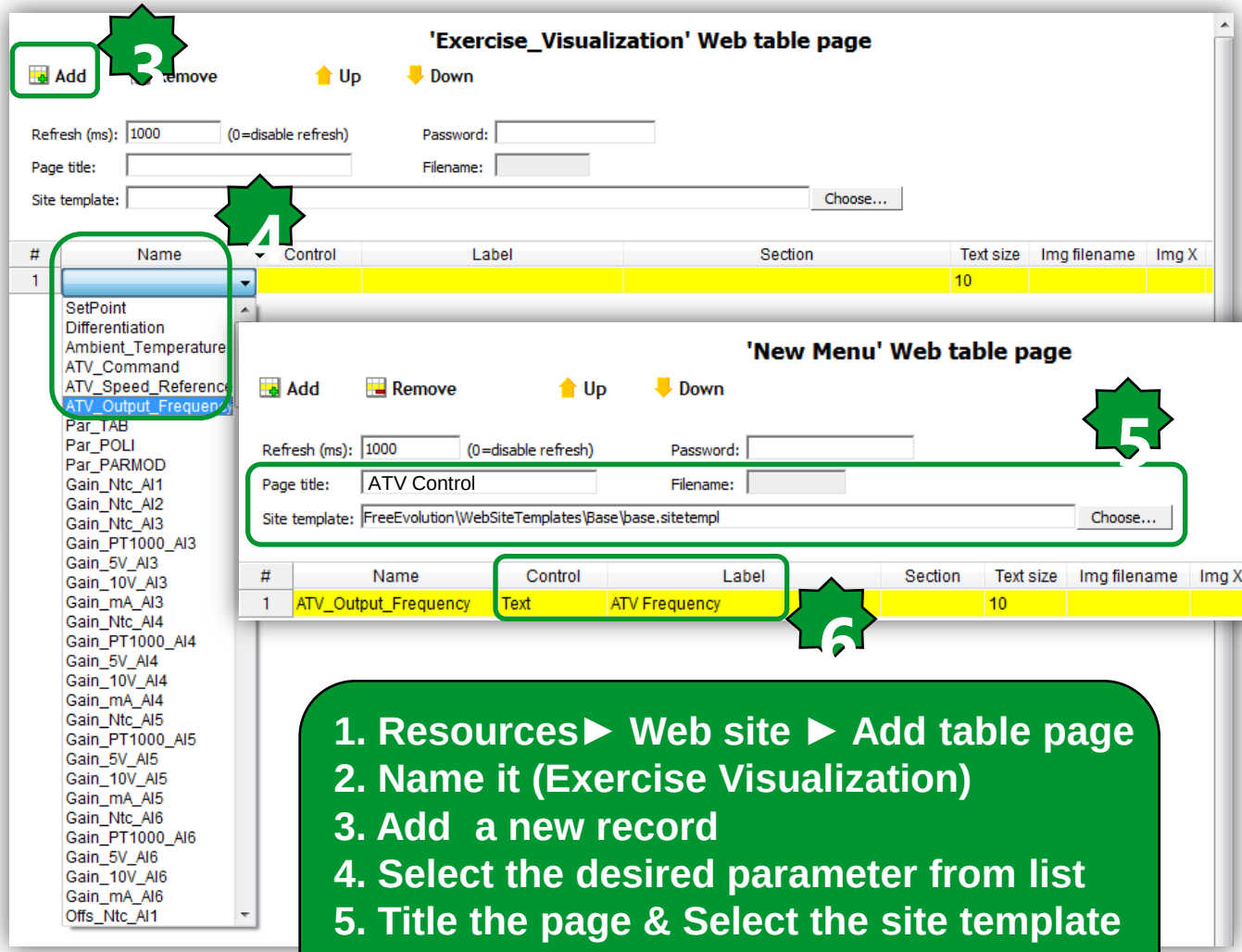




Application web pages

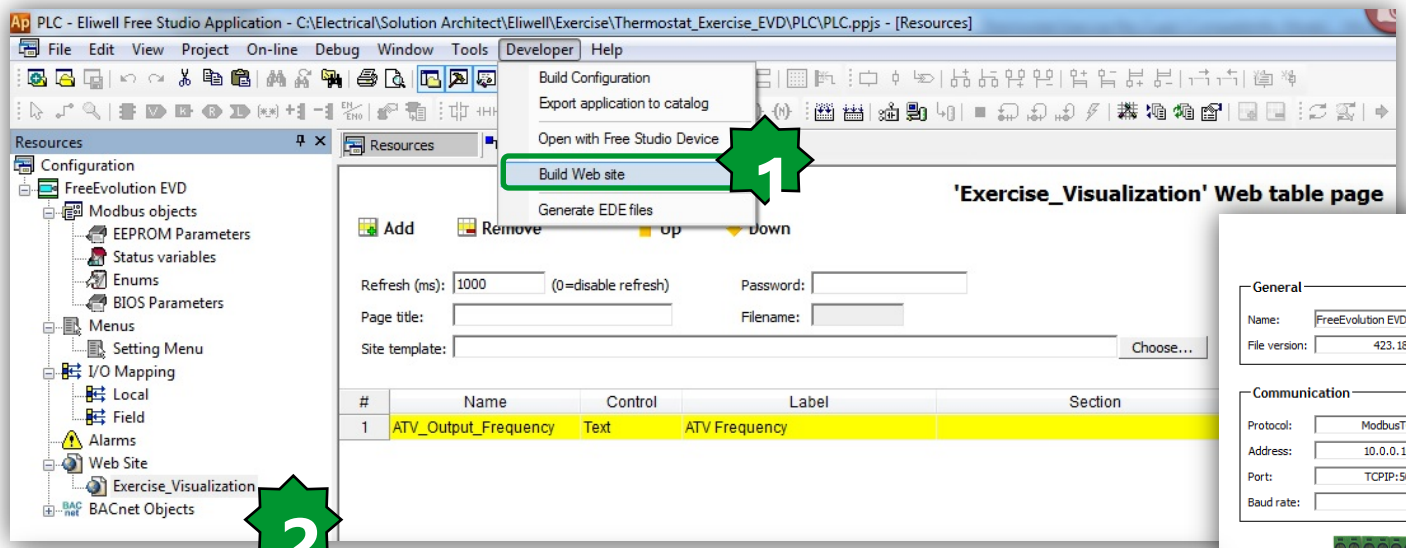


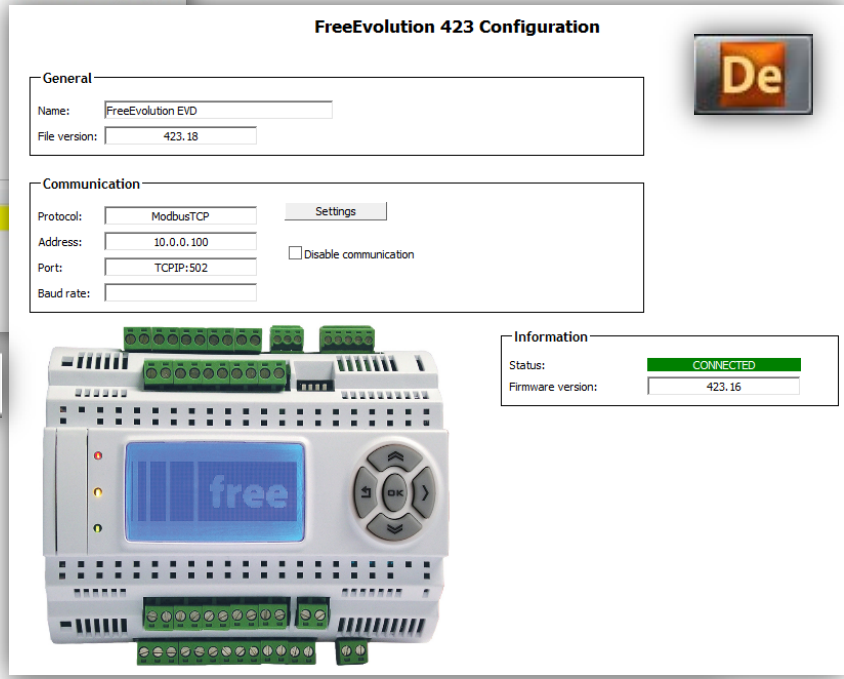


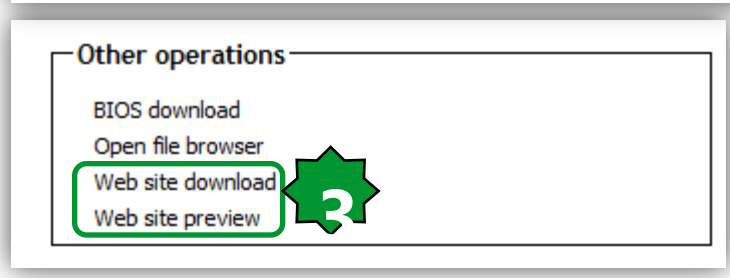


1. Resources► Web site ► Add table page
2. Name it (Exercise Visualization)
3. Add a new record
4. Select the desired parameter from list
5. Title the page & Select the site template
6. Define the control type & Label it

Web visualization/customized page



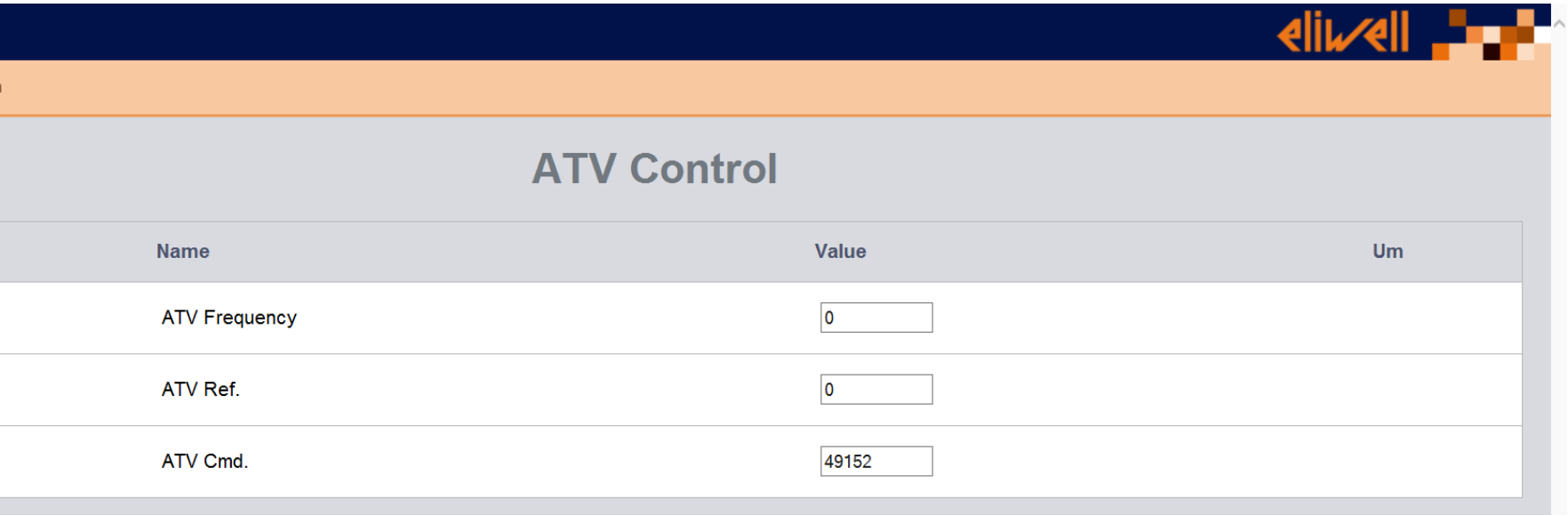




1. Developer ► Build Web site
2. Connect ►Device ► FreeEvolution Configuration
   Web site download
   Web site preview

Note: Embedded Web Pages are no longer available if Application Web Pages are used

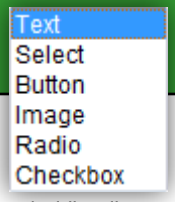
Custom web page preview



Use below address to return back to the embedded page: http://10.0.0.100/evoindex.htm

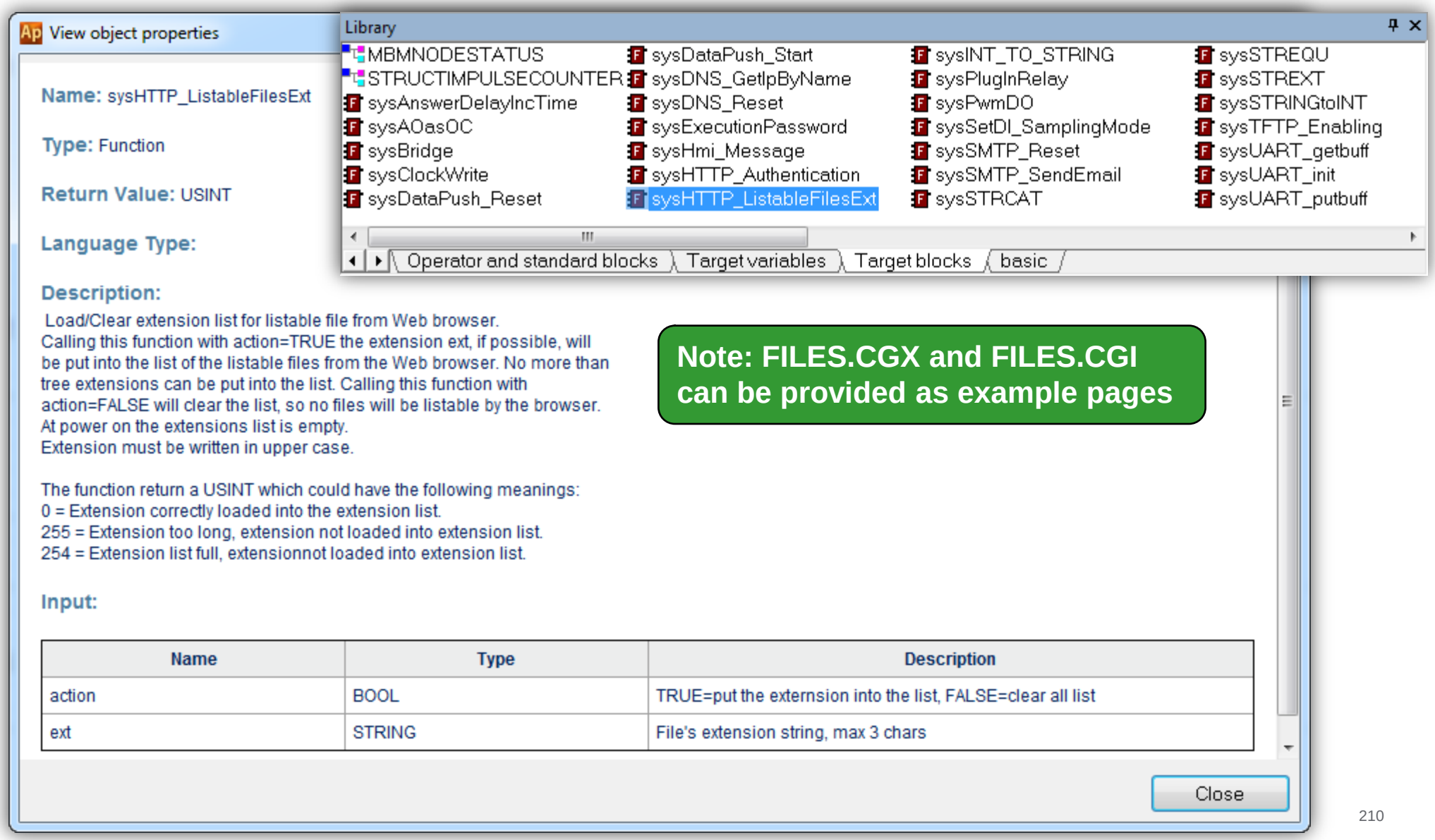
Custom page address: http://10.0.0.100/index.htm

If the device type is Boolean or Enums other types of control type are selectable.



Web folder is located in PLC ► Web to edit the header

sysHTTP_ListableFilesExt library



Note: FILES.CGX and FILES.CGI can be provided as example pages

Chapter 16

Firmware update

Goal:

Updating Smart & Evolution firmwares by Free Studio Device

Firmware update

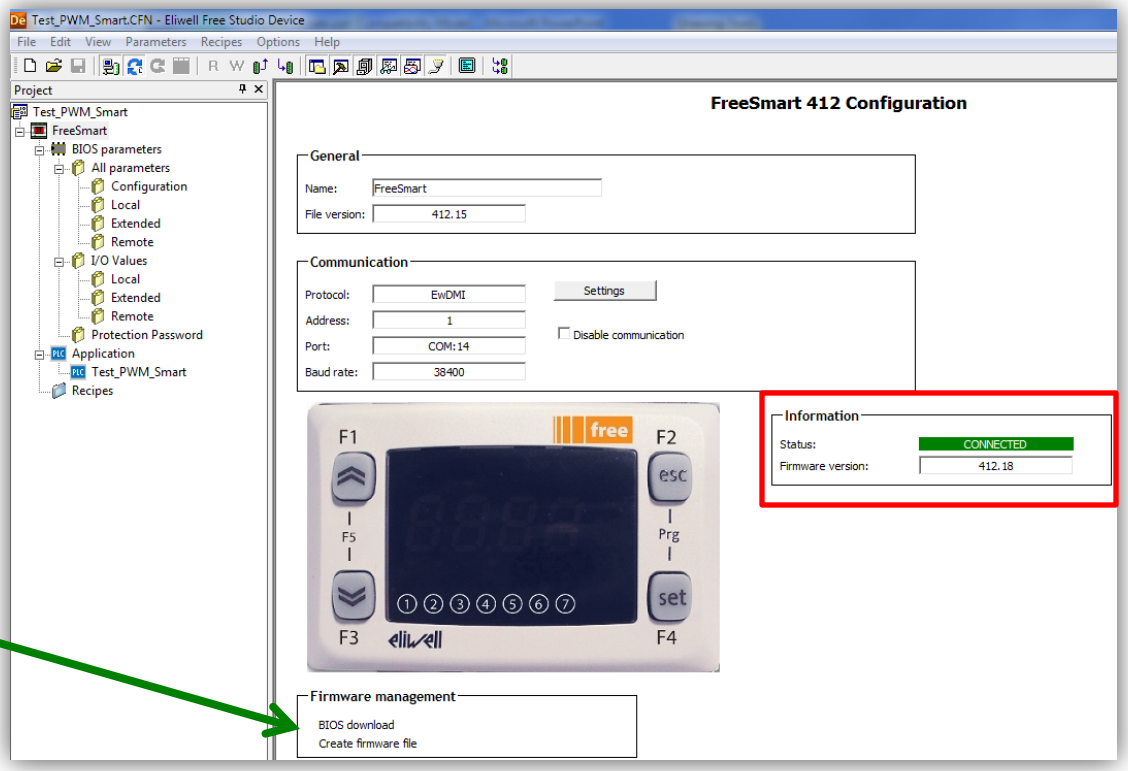
In Free Studio Device

➔Click on “Connects to Target”

➔Once the device is connected,

➔ Check the firmware version

➔If the firmware version is not the last one, click on BIOS download

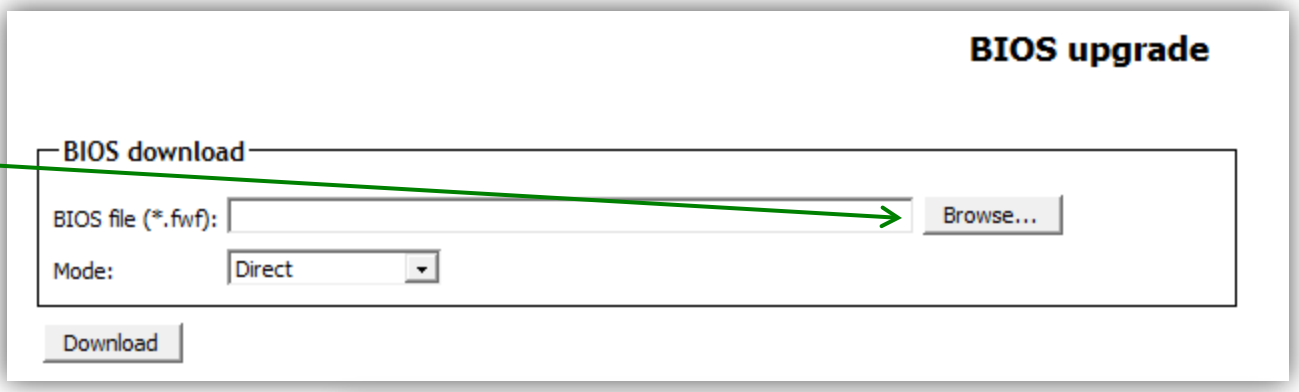


Note.
Smart: Application Lost
Evolution: Application kept

Note: Smart, the controller must be powered only by DMI.

Firmware update

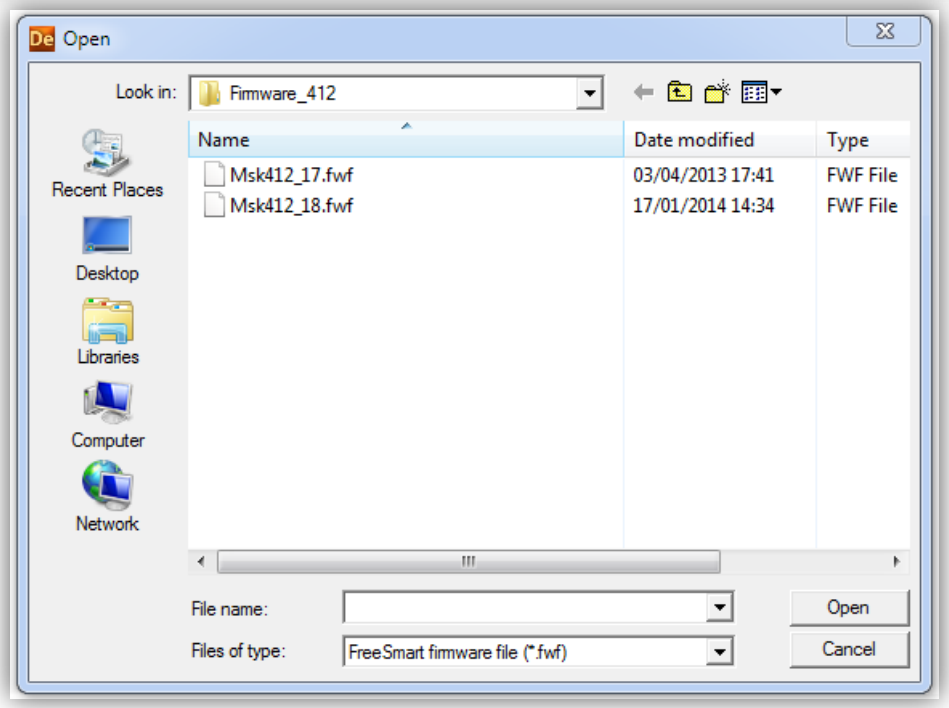
- Click on Browse



- The default folder opened contains the last firmware version released with the software.

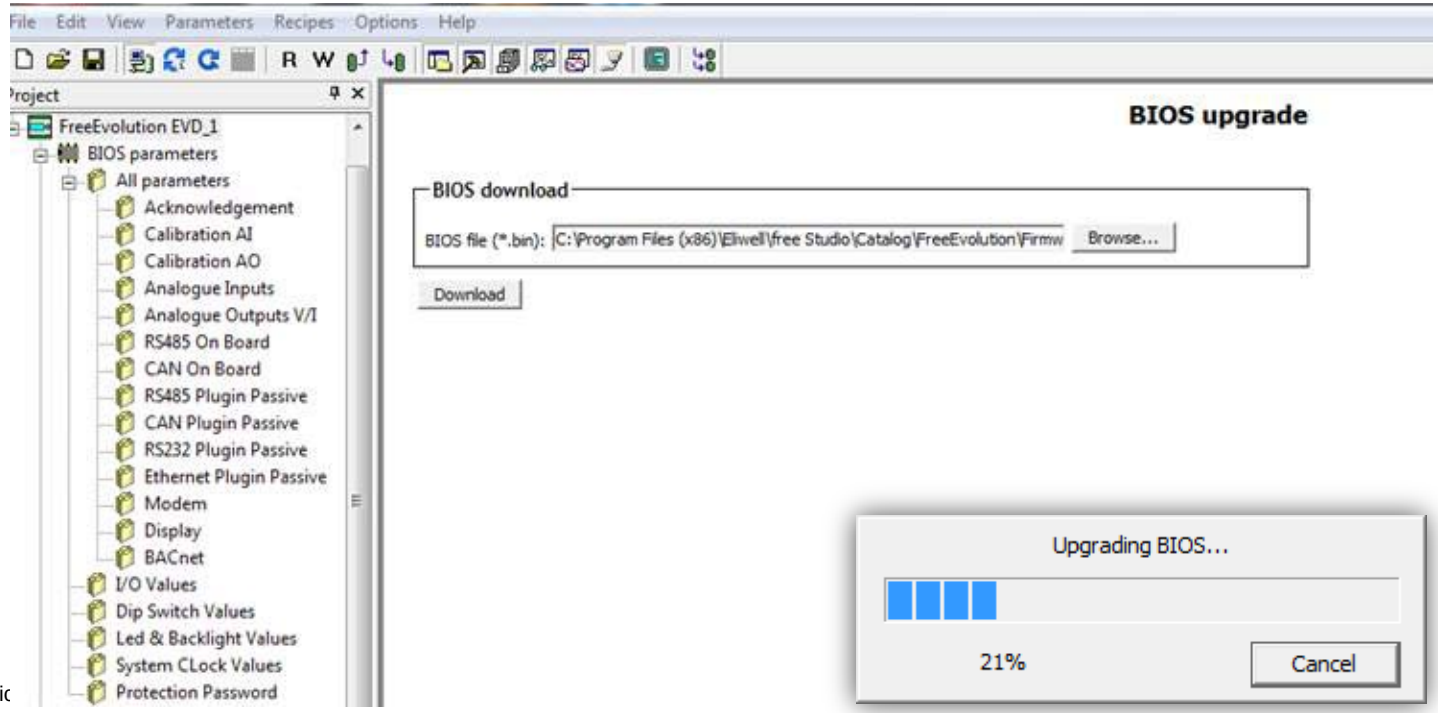
➔Select the last firmware version

➔Click on Open



Firmware update

- Select Direct Mode
- Click on Download
- The same procedure can be applied to Evolution except for Mode selection (not available) and power supply to be provided.



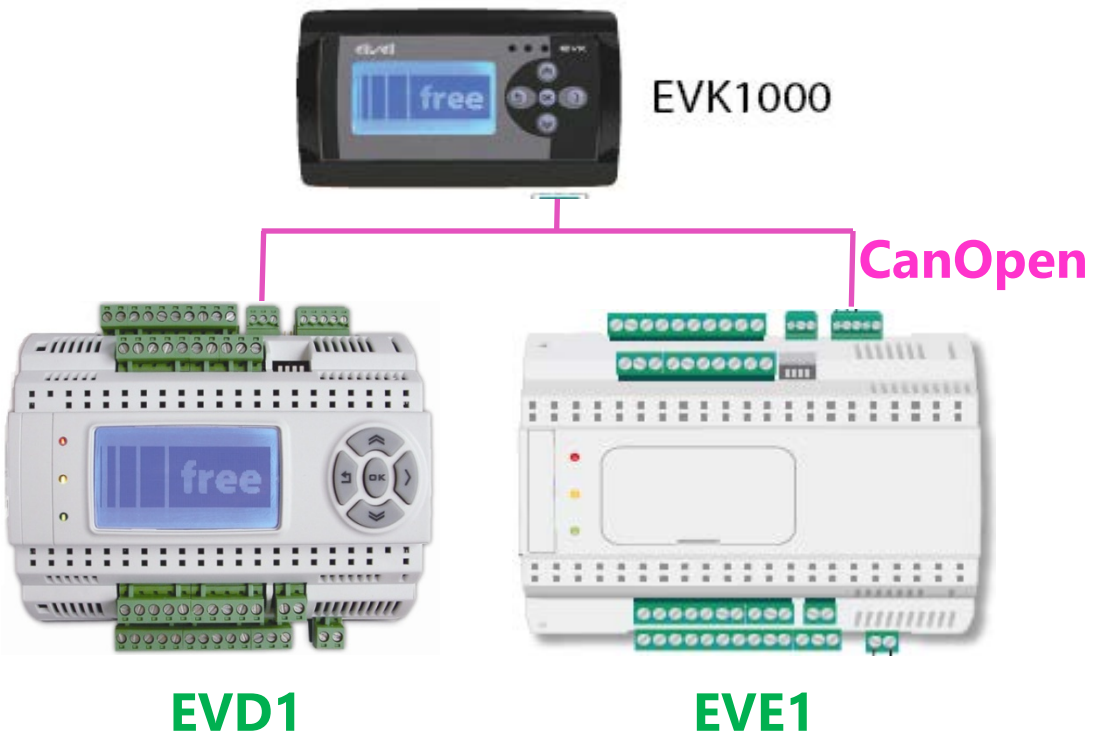
Chapter 17

User Interface

Goal:

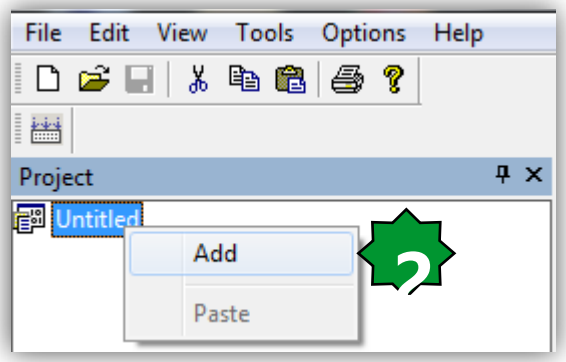
Local & Remote display programming by User Interface

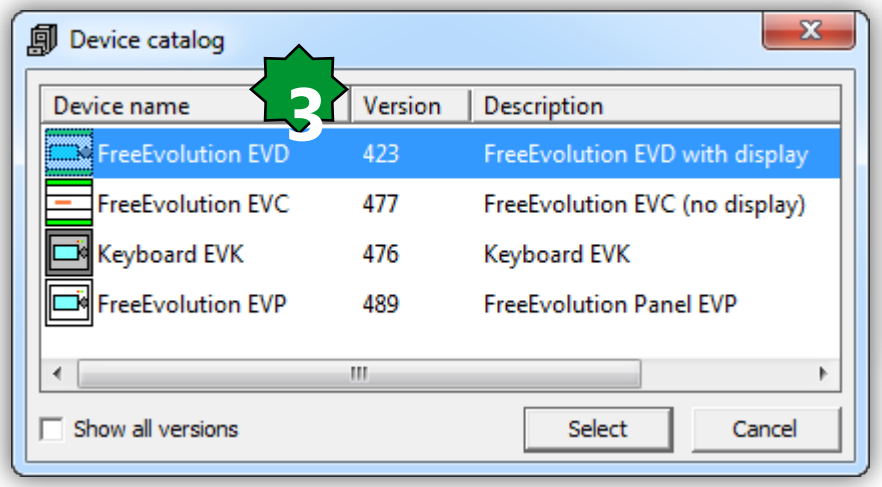
DemoField Example

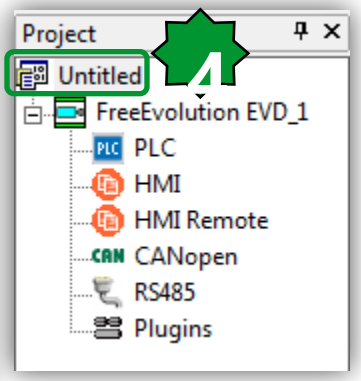


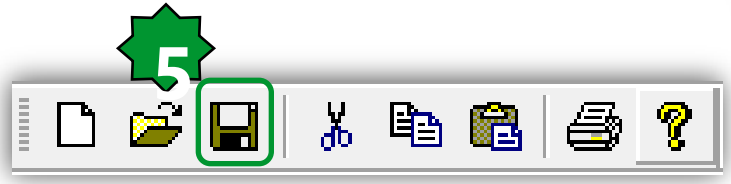
NOTE: Leave the CanOpen end resistor jumpers only to the endline Devices, in this case EVD and EVE1

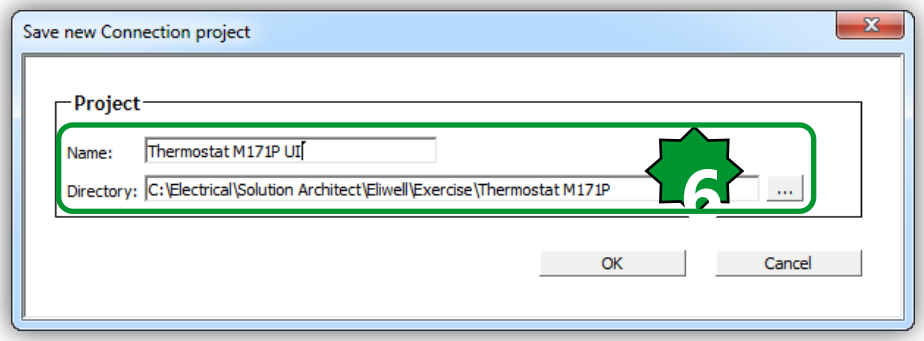
Keyboard EVK configuration

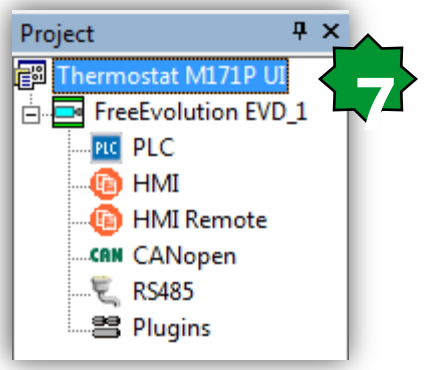












Keyboard EVK configuration

Thermostat_exercise.CON - Eliwell Free Studio Connection

File Edit View Tools Options Help

Project

- Thermostat_exercise
  - FreeEvolution EVD_1
    - PLC
    - HMI
    - HMI Remote
    - CANopen
      - Expansion EVE_1
      - Keyboard EVK_1
    - RS485
    - Plugins

Keyboard EVK Configuration

General

Name: Keyboard EVK_1

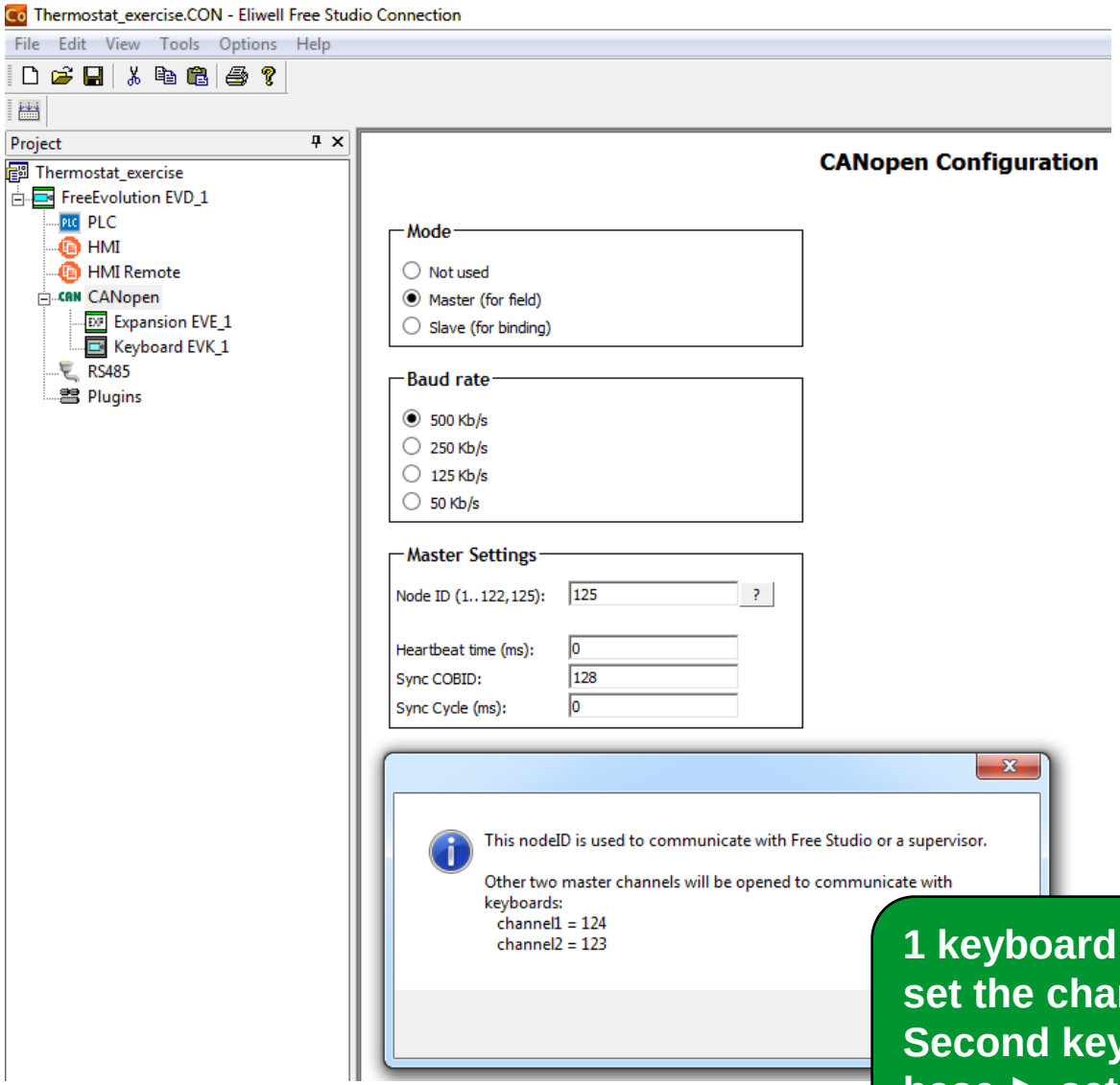
Version: 476.11

Network settings

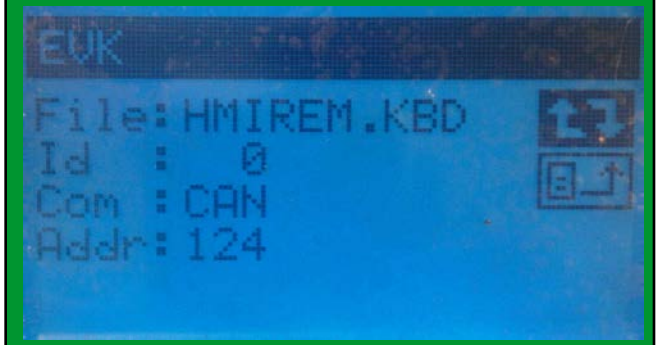
Node number (126,127): 127

Network settings should be aligned to the real address of the EVK (127 is the factory default).

CANopen configuration



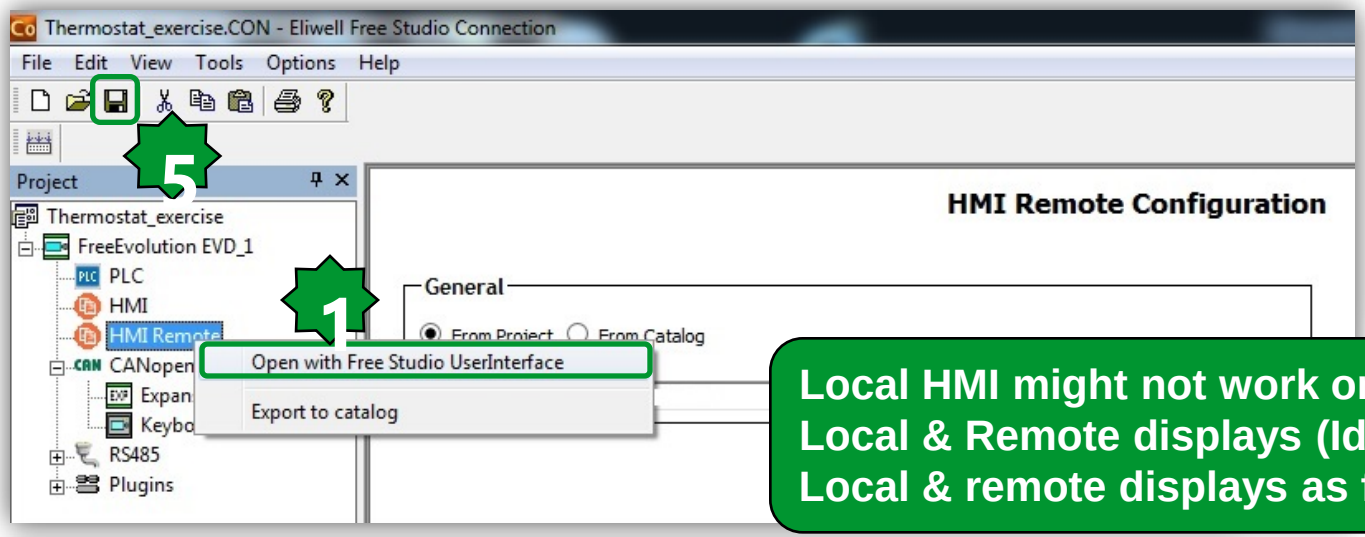
- **124 or 123 are the addresses to be used in the HMI management menu of EVK**

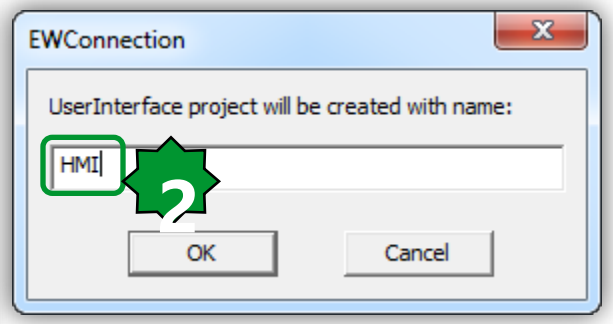


- **124 is the factory default**

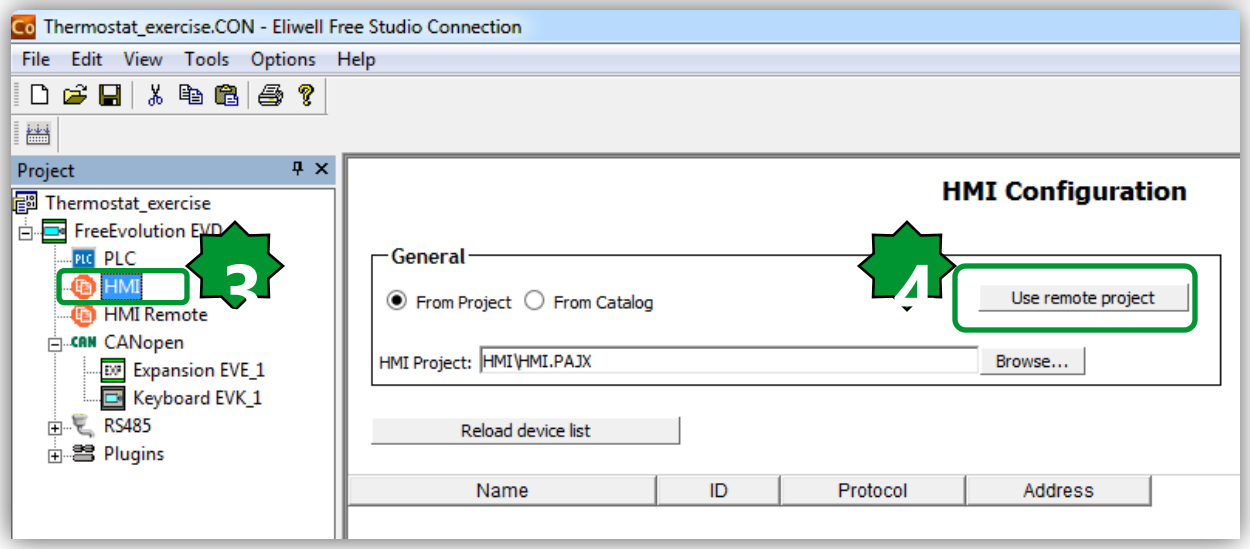
1 keyboard to communicate with base ► set the channel=124
Second keyboard to communicate with base ► set the channel=123

HMI project linking/creation





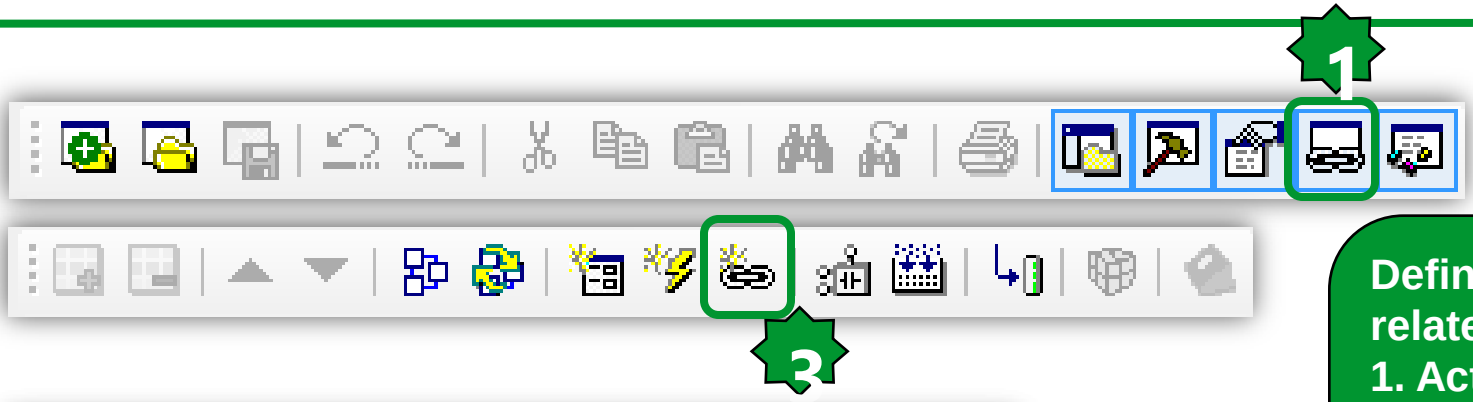
Local HMI might not work on remote display
Local & Remote displays (Identical) ► HMI remote
Local & remote displays as future option► HMI remote



HMI:Local Display
HMI Remote: EVK1000

2. Name it ► OK ► User Interface project starts automatically
3 & 4. Local display will have the same HMI of the remote one.

Preliminary: Actions



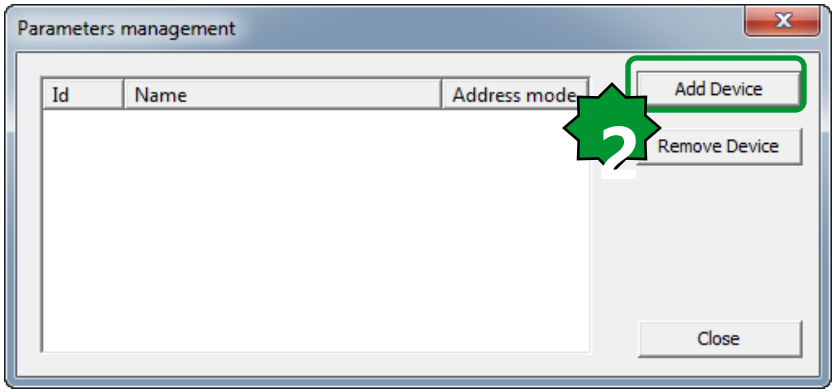
Actions

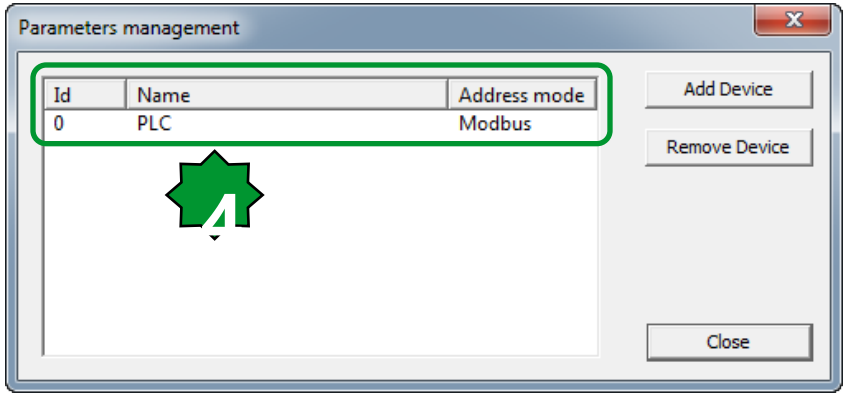
Local actions | Global actions

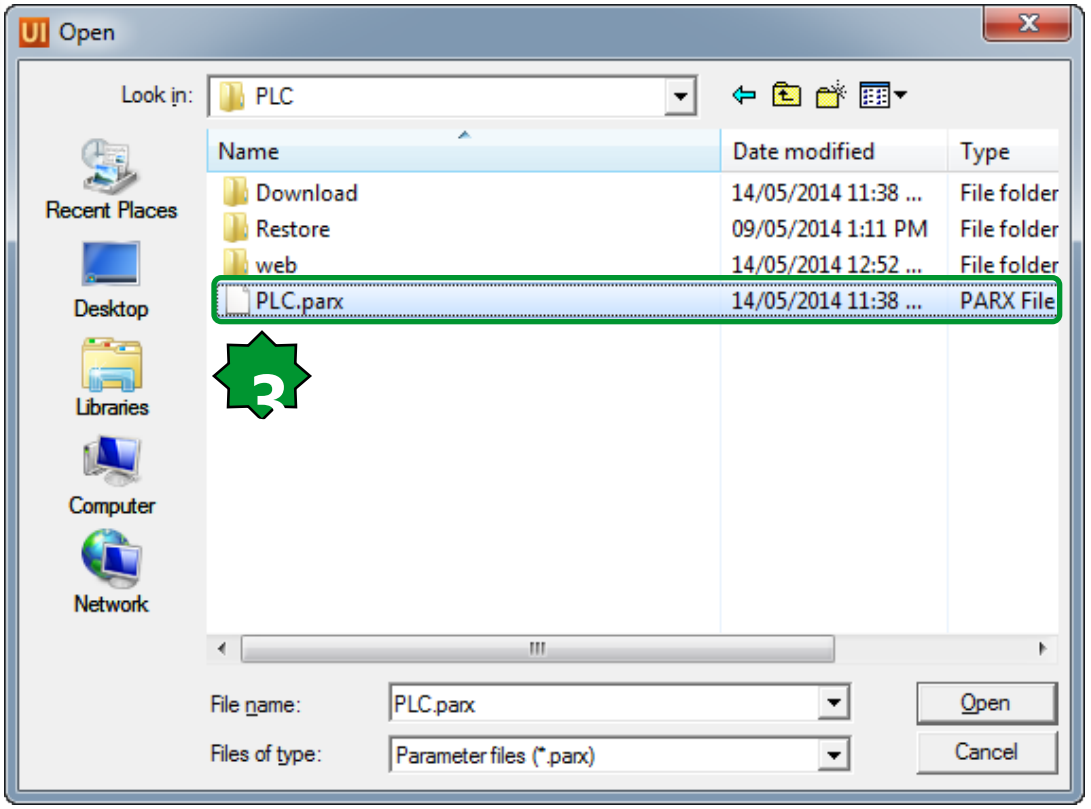
Key	Action	Link
Enter	Edit	
Left	PrevField	
Right	NextField	
Up	PrevField	
Down	NextField	
LongLeft	Close	

Define the global action related to the buttons
1. Activate action bar icon
2. Global actions
3. New action
3. Define the Key
4. Define the Action

Preliminary: Link parameter file

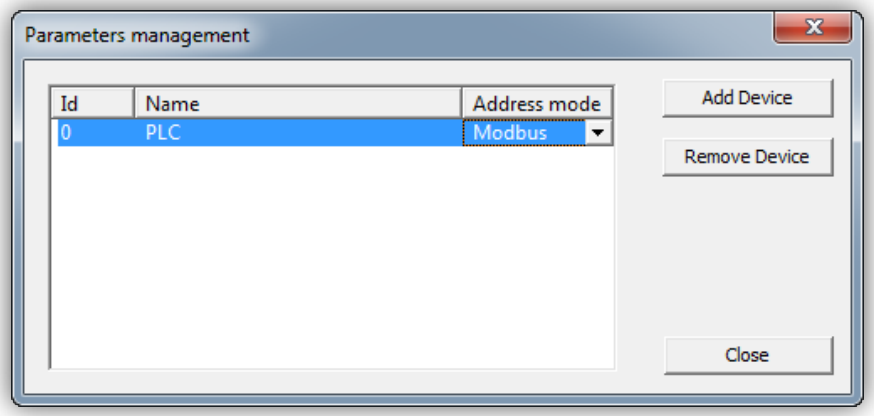


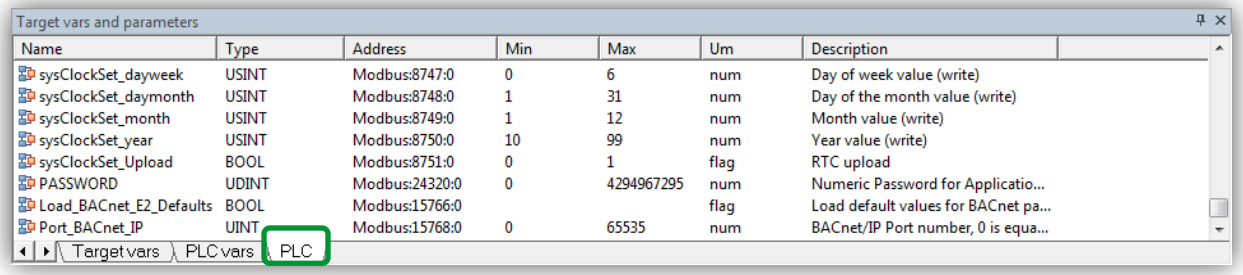




- Link the parameters file of your application project.
- Select short name to prevent long variable names

Target Vars & parameters





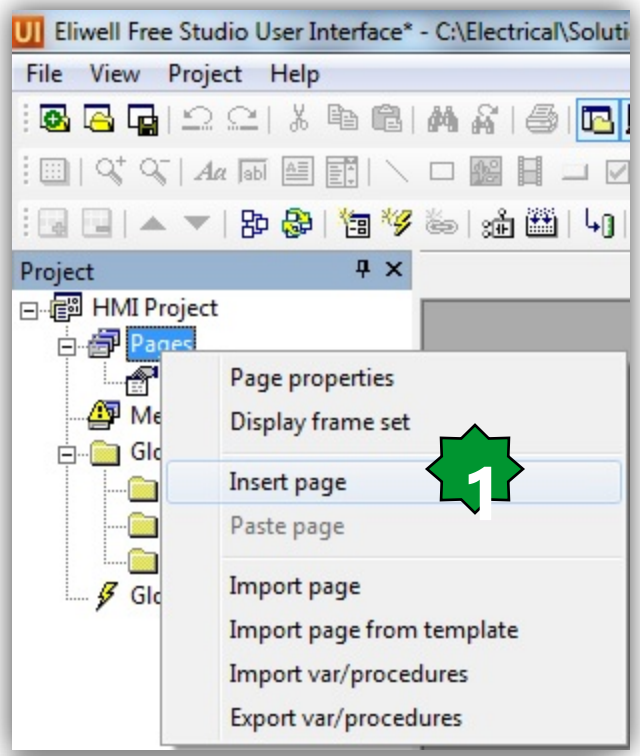
- **Name can be eventually changed manually in order to adapt the same UI to different project without changing the set already defined (@ syntax)**
- **The list of parameters/status variables will appear**
- **Note: Remember to refresh parameters file every time you change the related Application project**

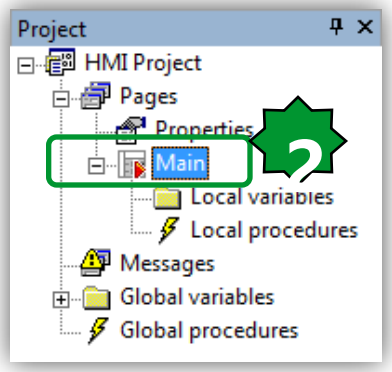
Toolbar description

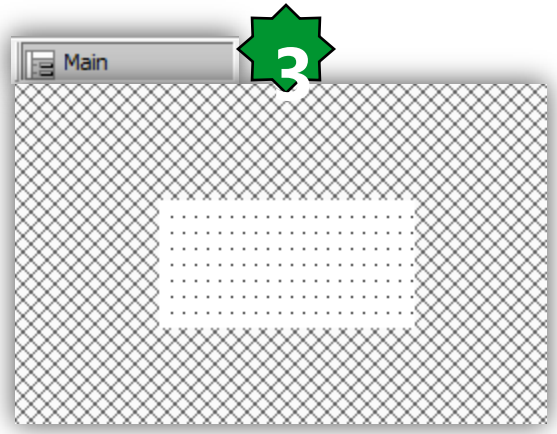
1. Insert static
2. Insert new edit
3. Insert new image
4. Insert new animation
5. Insert new button
6. Insert new check box
7. Insert new progress
8. Bring to front
9. Sent to back

We will see how to manage all these objects…

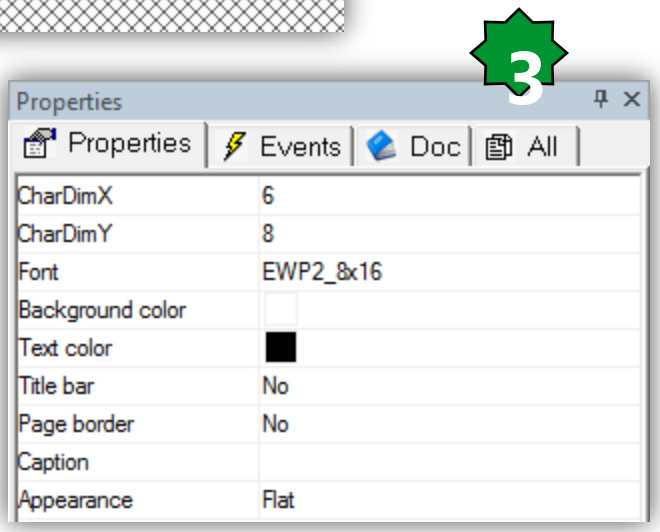
Page creation & page properties

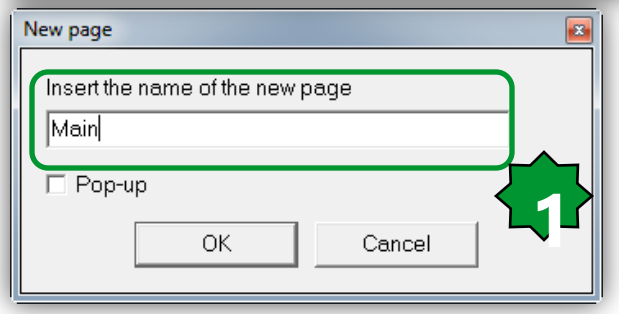


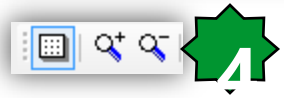




Note.
Press Enter to apply the changes otherwise it will be lost.

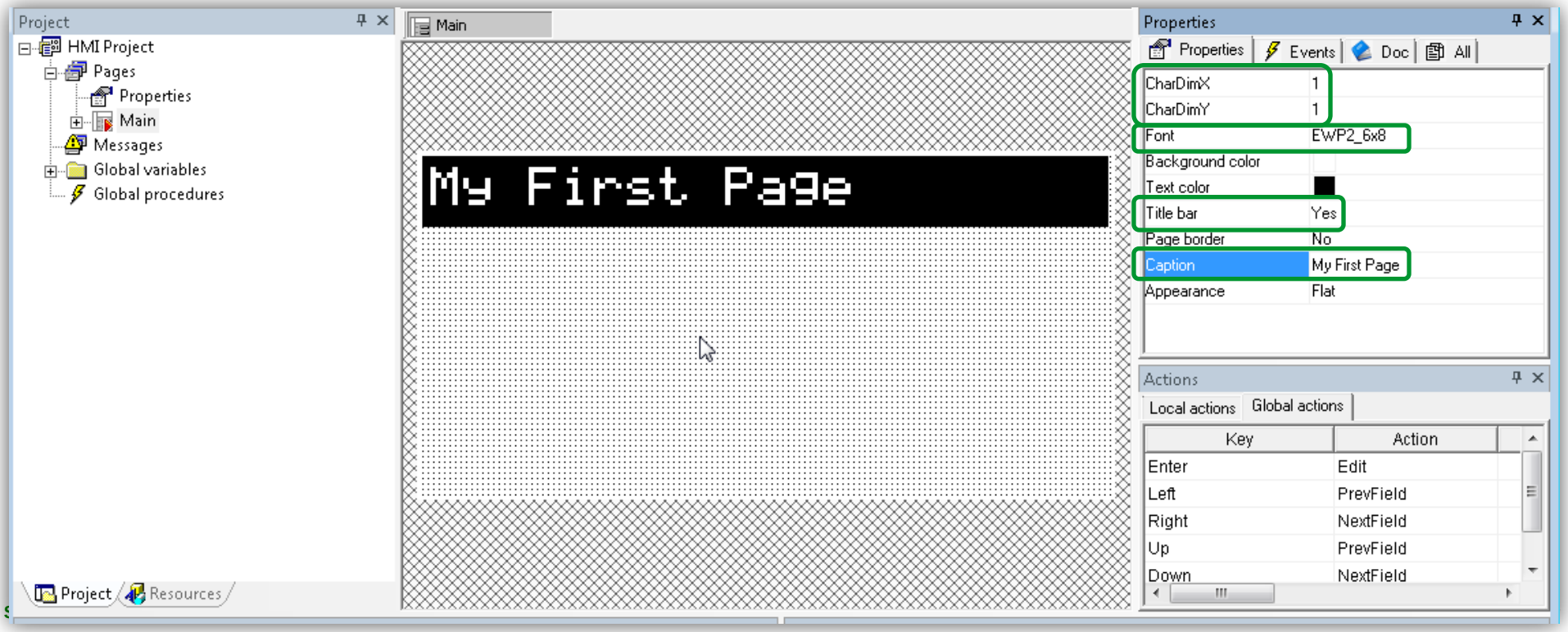






Page Customization

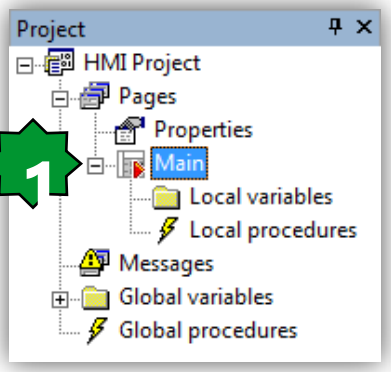
- Main Page Properties
  - Customize grid
  - Insert Title Bar
  - Define Font size
  - Define Title

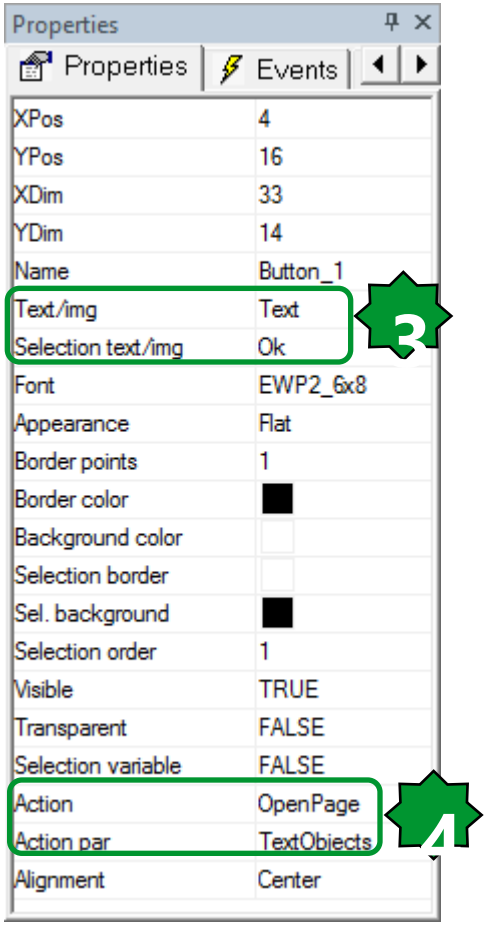


Page Navigation by Graphic Button

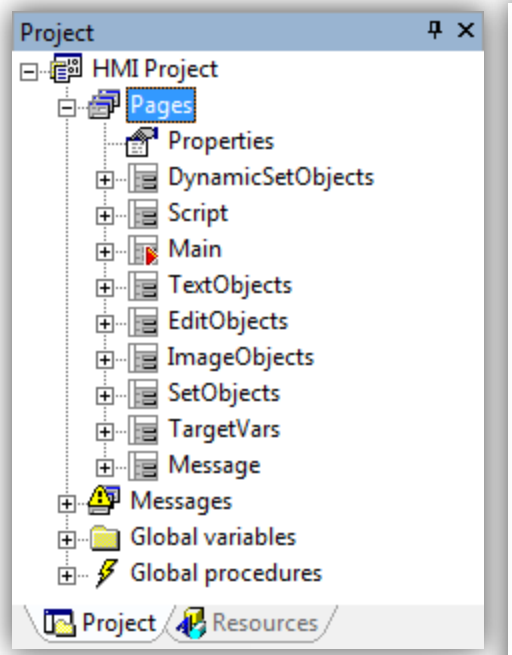
1. Define a new page
2. Create a button in the source page

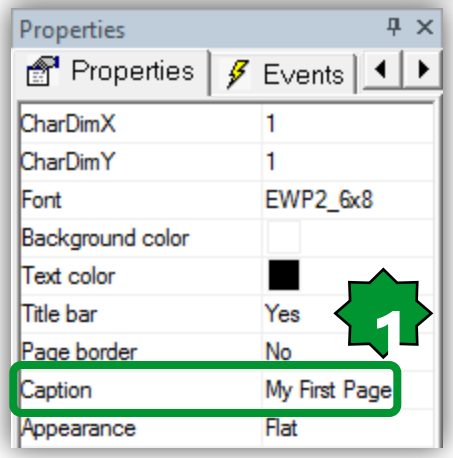
3. Text/img can be changed base on selection status
4. Define the Action and Action Par properties of the button



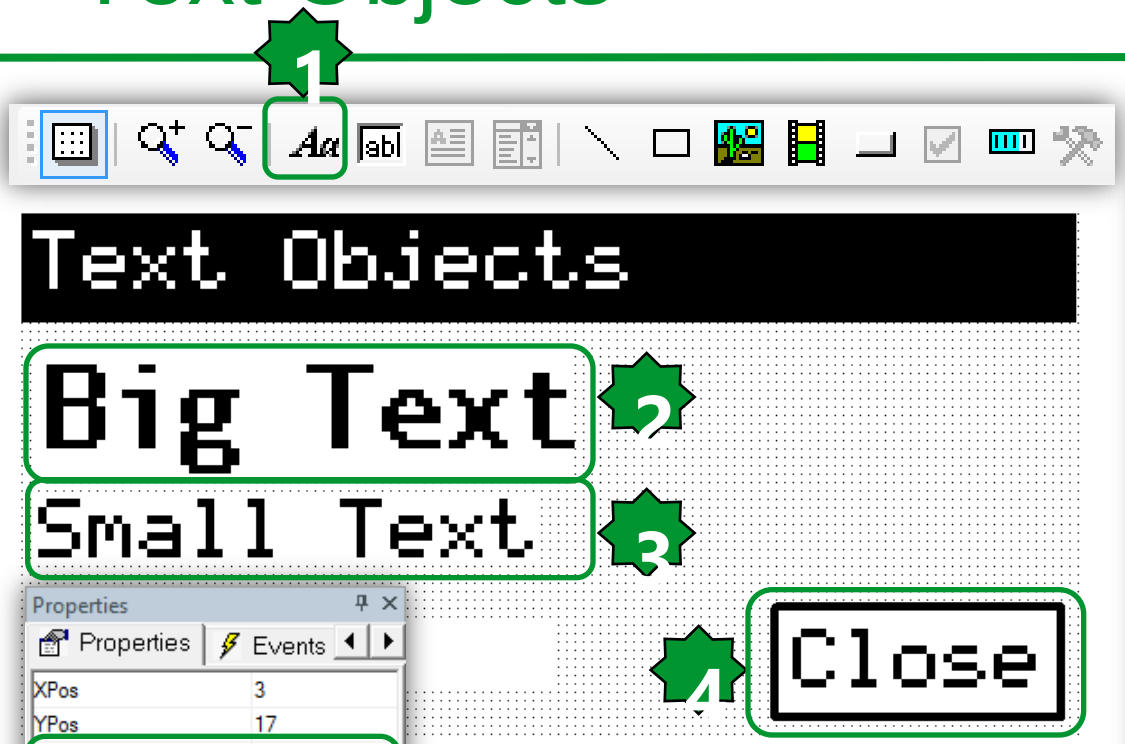


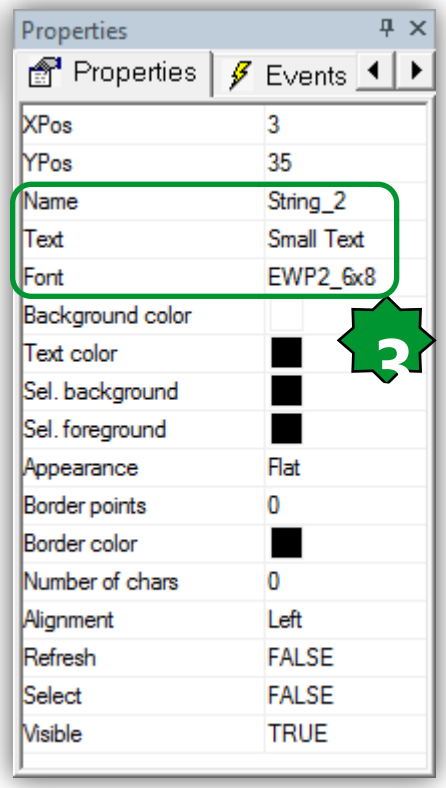
Main/My First Page…

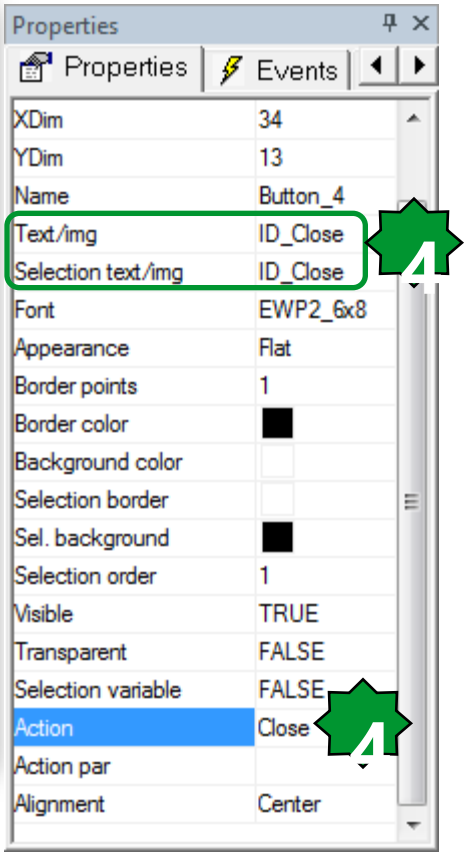




Text Objects





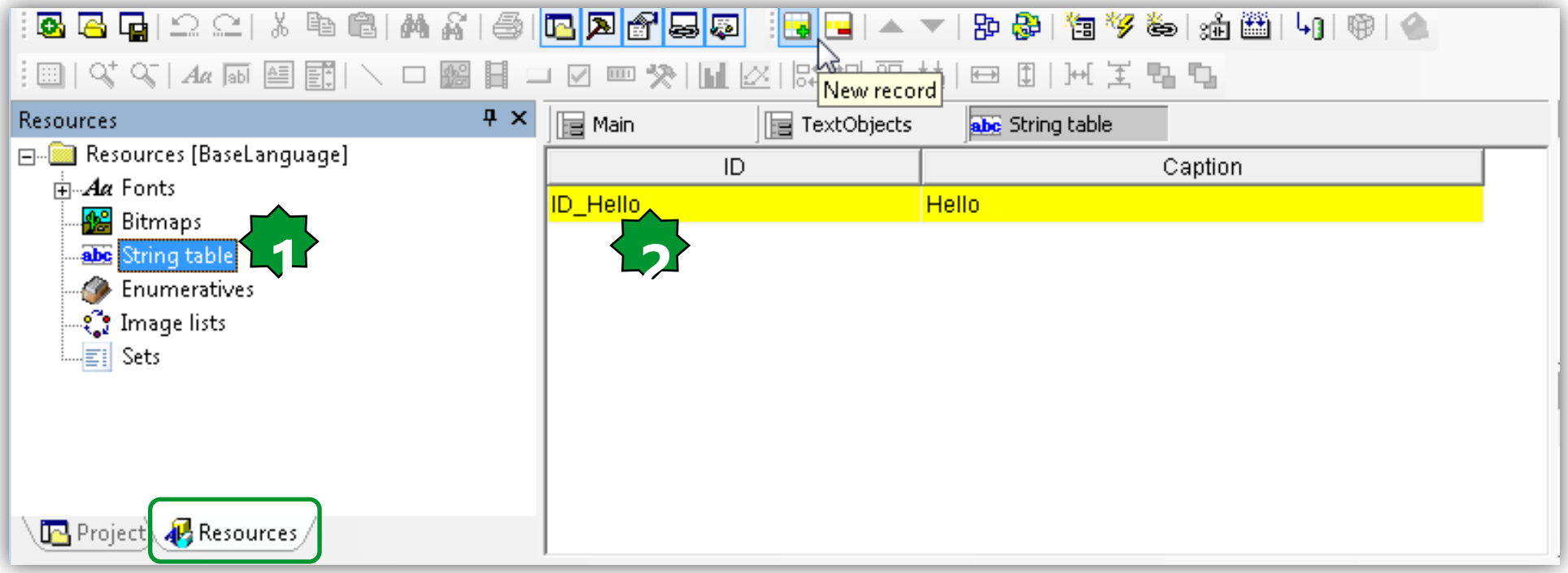


1. **Insert static text**

Define the required font size (2 sizes are available)

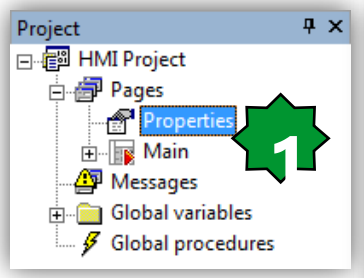
Text Objects...

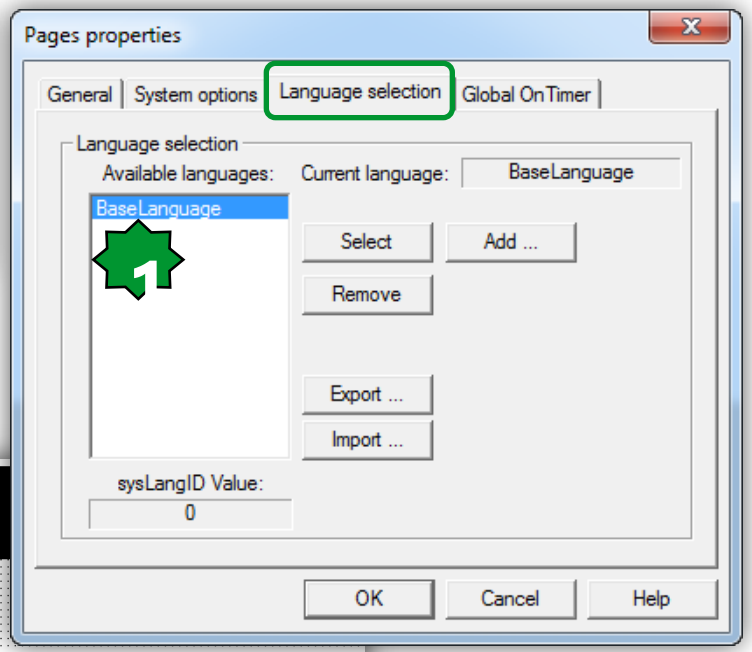
- Text to be translated:
  1. Define the string ID and the string in the current language
  2. Use the ID instead of static string

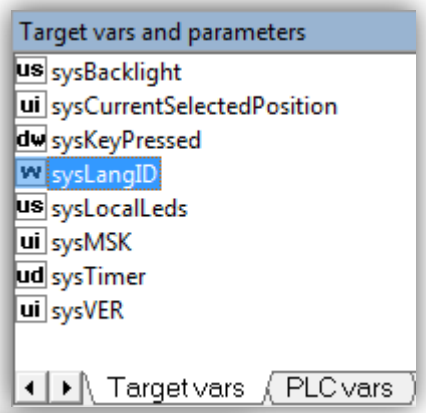


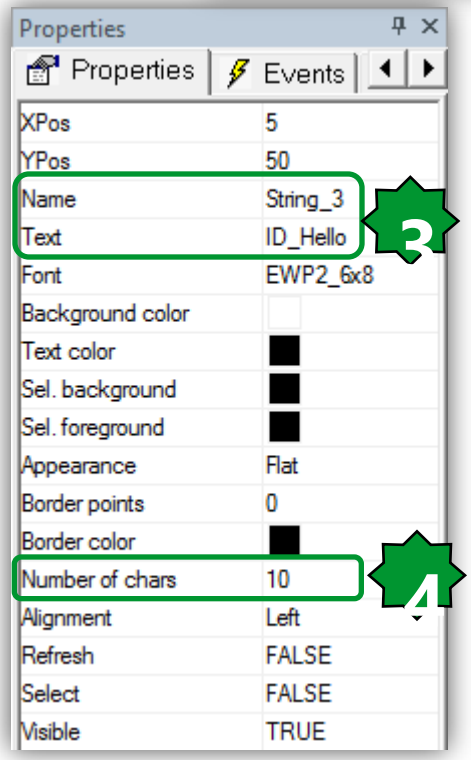
...Text Objects

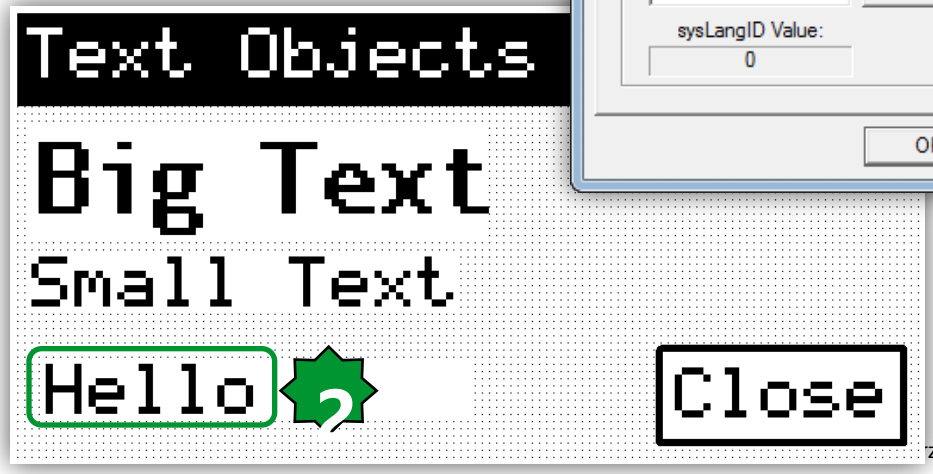
- Text to be translated:
  1. Languages are defined in the language selection tab
  2. String table can be exported/imported and translated





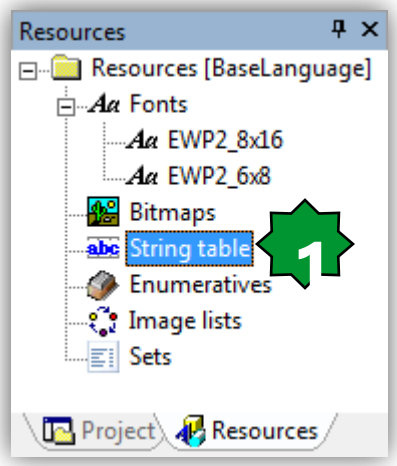






4. Max = 19 Chars

String Table



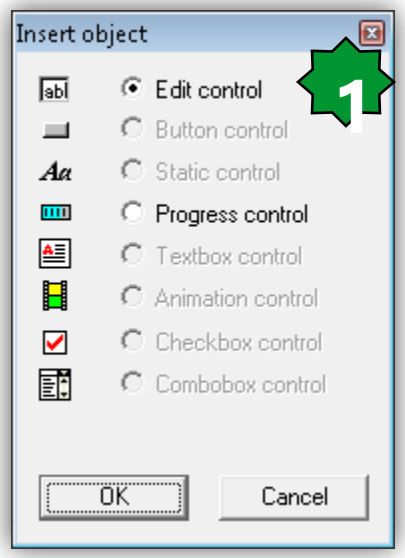
ID	Caption
ID_Hello	Hello
ID_EditObjectTitle	Edit
ID_AmbientTemp	Ambient:
ID_Setpoint	Setpoint:
ID_GreenLed	Green Led:
ID_Close	Close
ID_Image	Image
ID_Set	Set
ID_Script	Script
ID_Differential	Differential:
ID_EveAlarm	Er01-EVE Alarm
ID_DIL2Alarm	Er02-DIL2
ID_DIL3Alarm	Er03-DIL3
ID_DIL4Alarm	Er04-DIL4
ID_DynSet	Dyn.Set

2

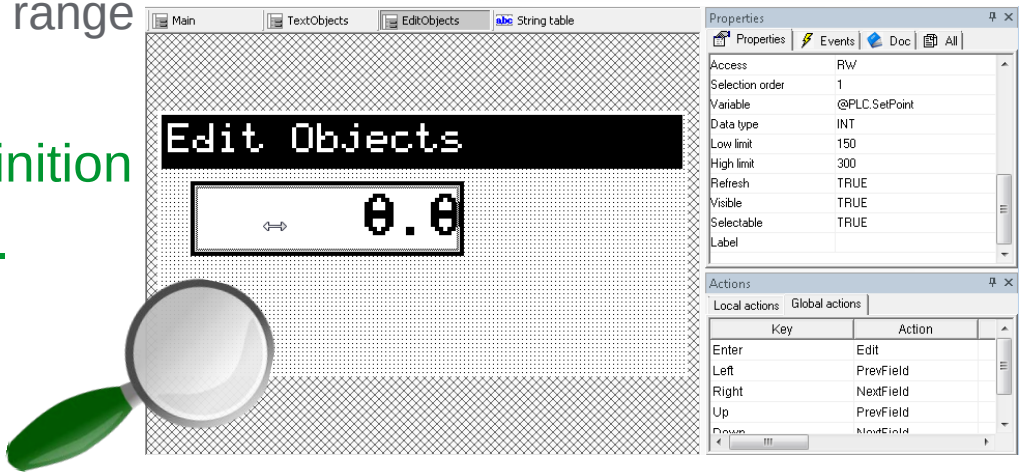
Let‘s define our String table…

Edit Objects - Values

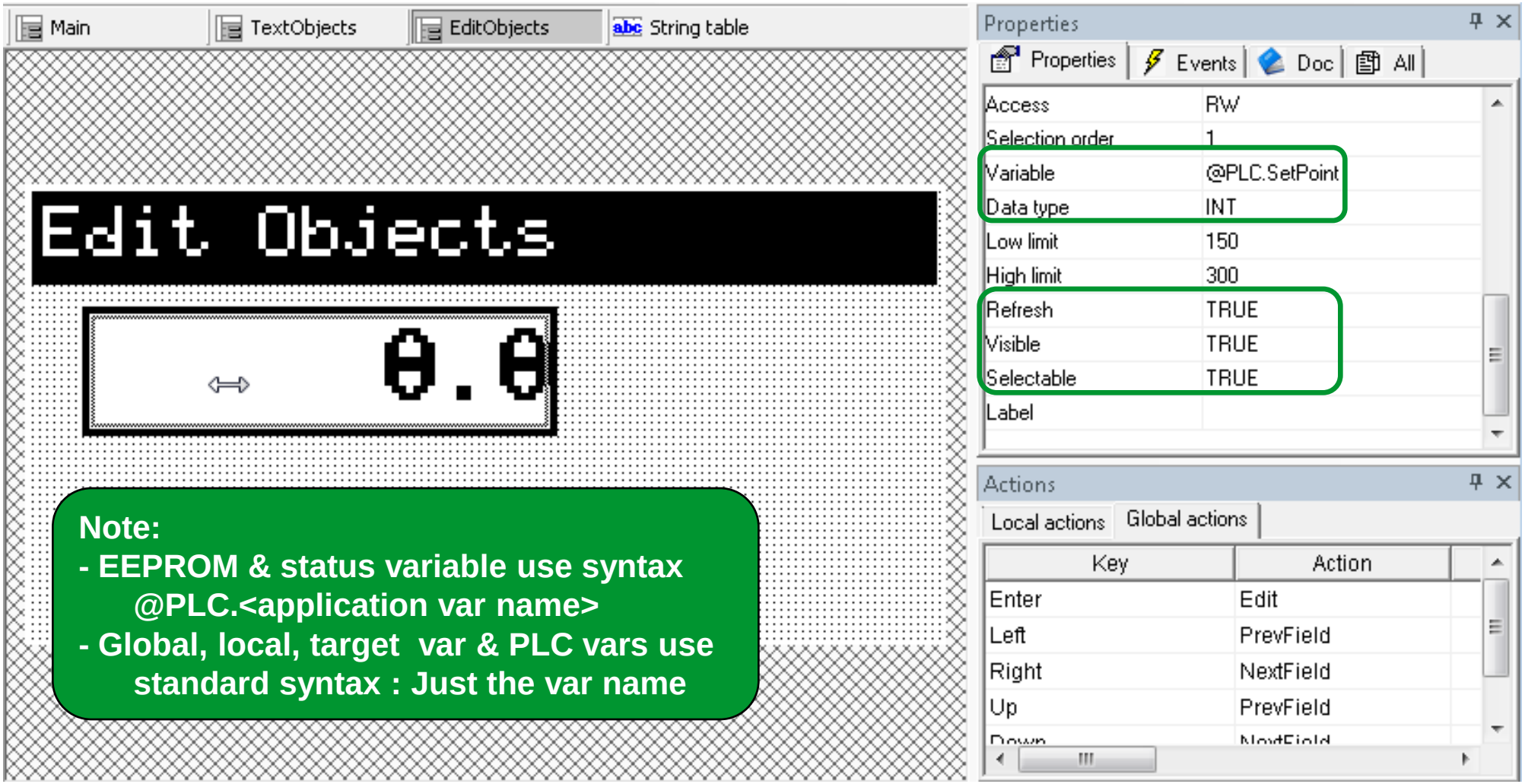
1. Select object or Drag and drop desired variable from from PLC tab



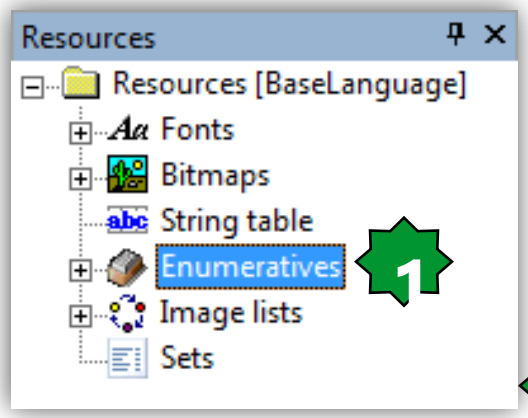
- Main Properties
  - Variable: syntax used is: @PLC.<application var name>
  - Format
  - Selectable: True for R/W, False for RO
  - Refresh
  - High/Low Limit: shows ----- outside the range
- Select the @PLC.<var> desired
- Default Properties comes from Ap definition
- Selectable: False means, not editable.

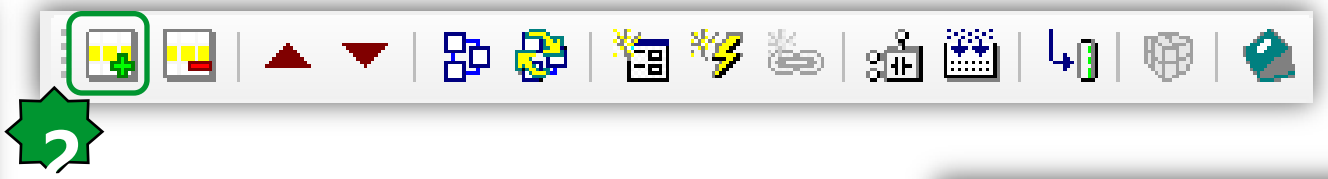


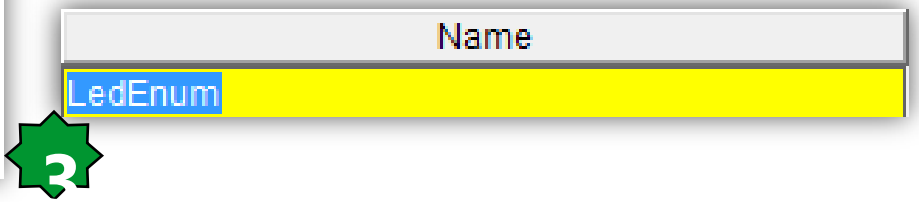
Edit Objects - Values

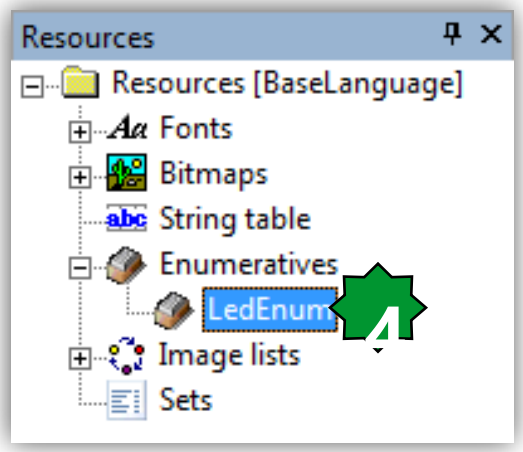


Enumerative creation



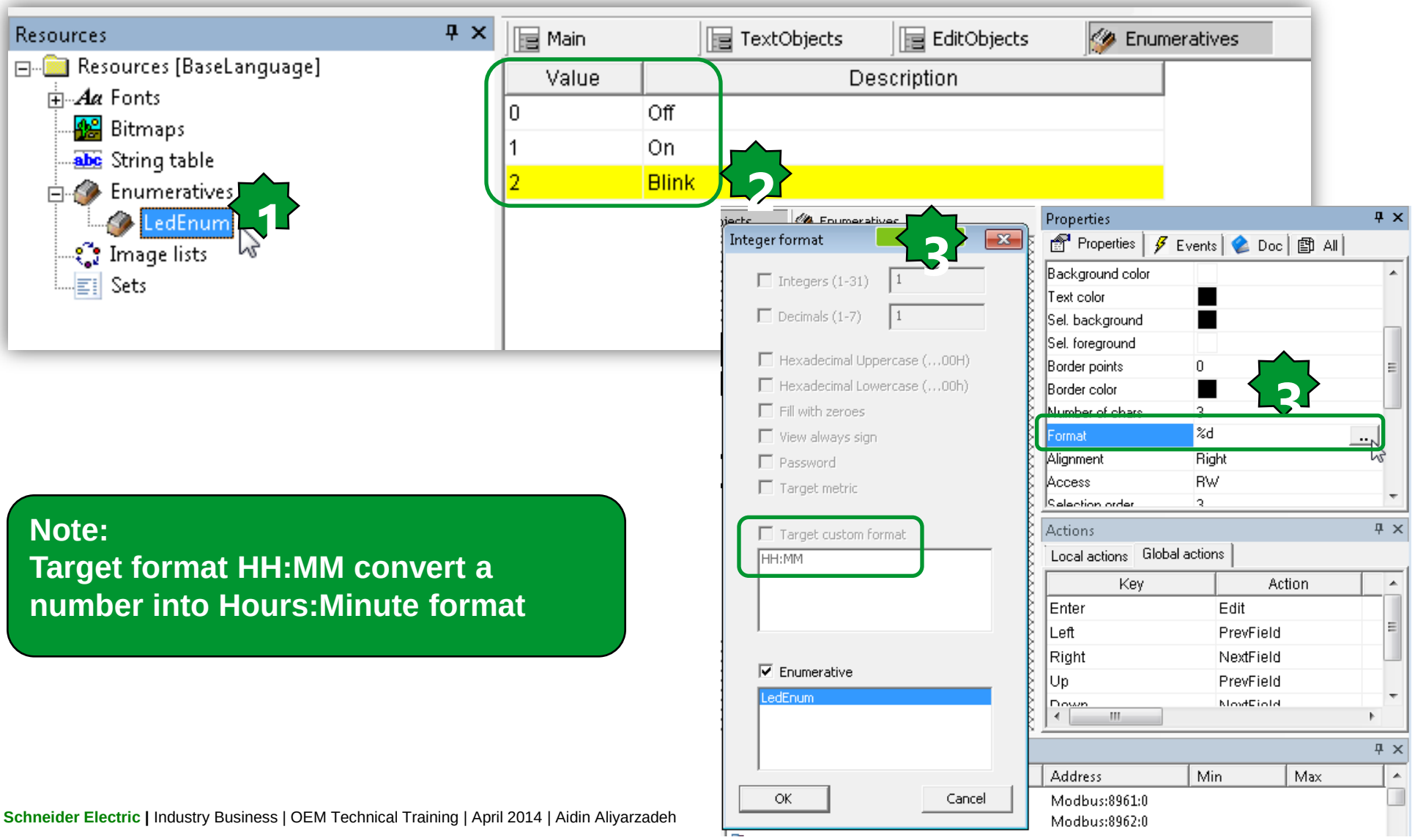






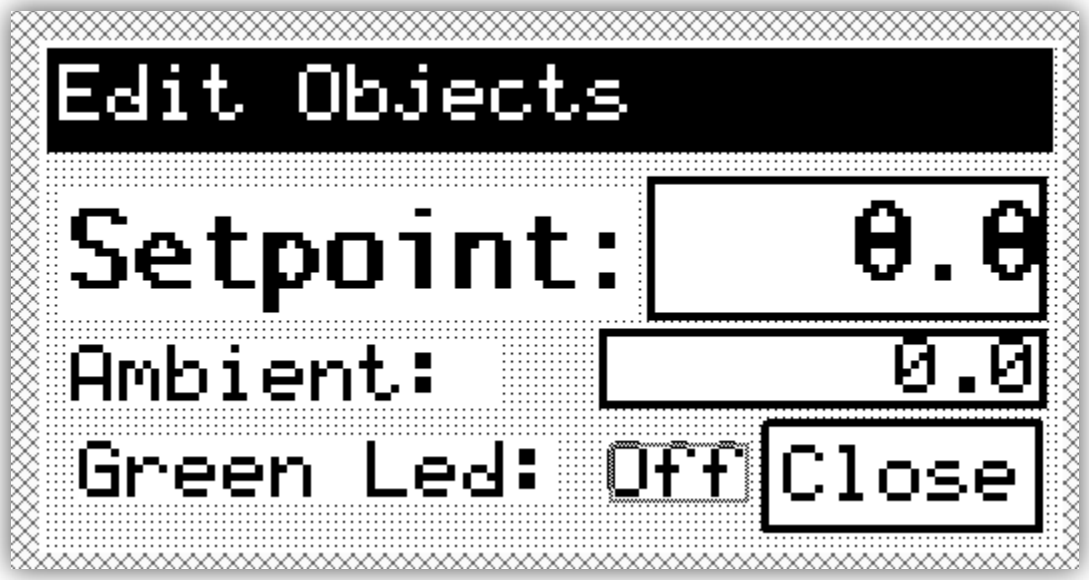
1. Resources► Enumeratives
2. Add new
3. Name/Rename it ► Open
4. Enumeratives tree

Enumeratives

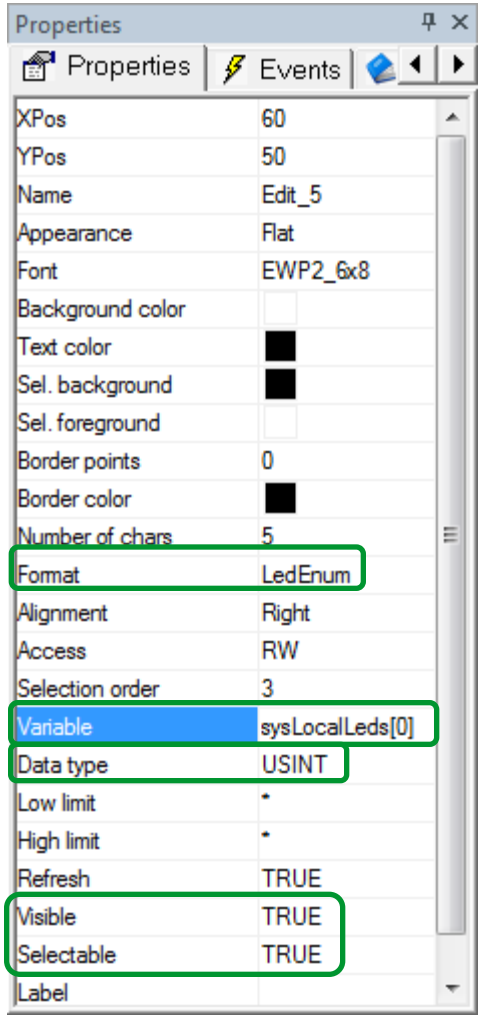


Note:
Target format HH:MM convert a number into Hours:Minute format

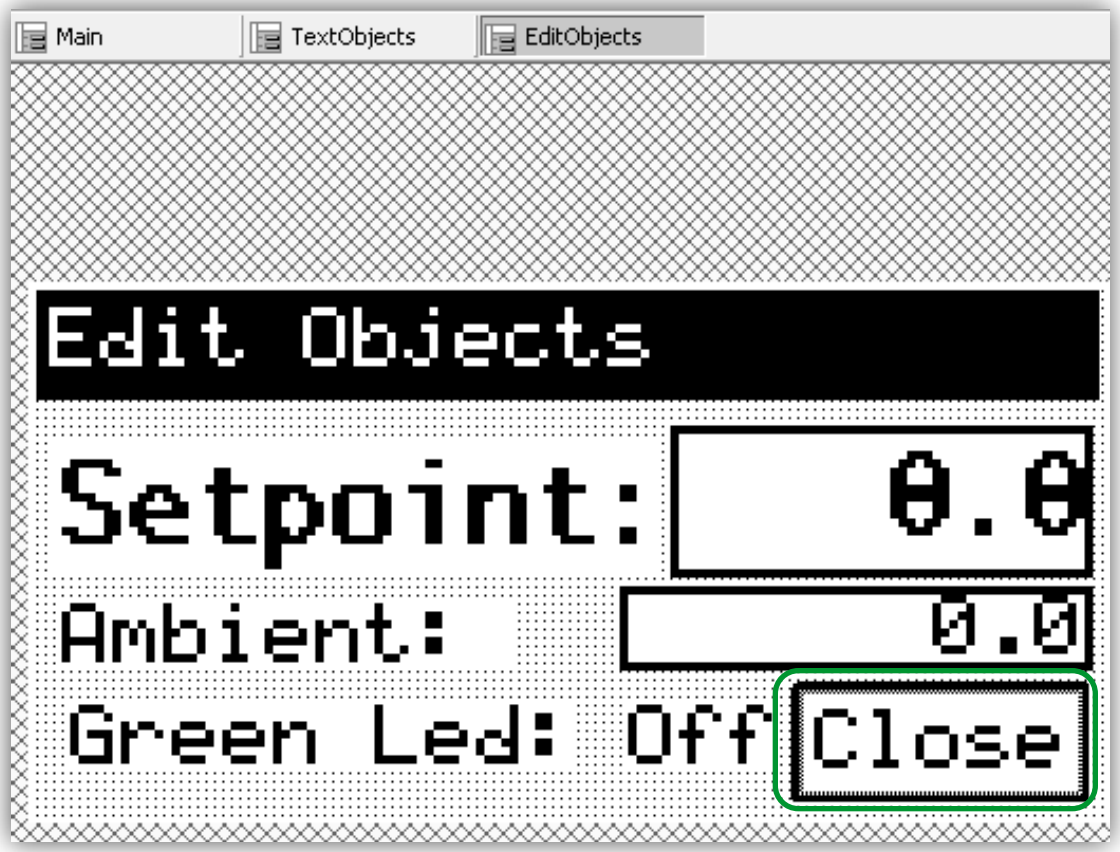
Enumeratives

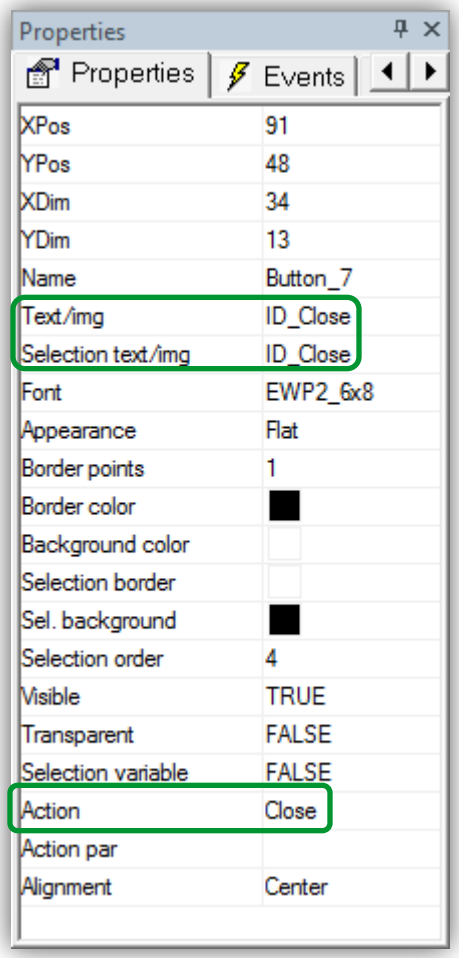


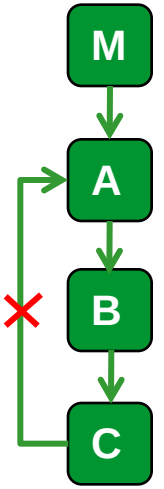
Enumeratives are translatable



Page Navigation – Close Action

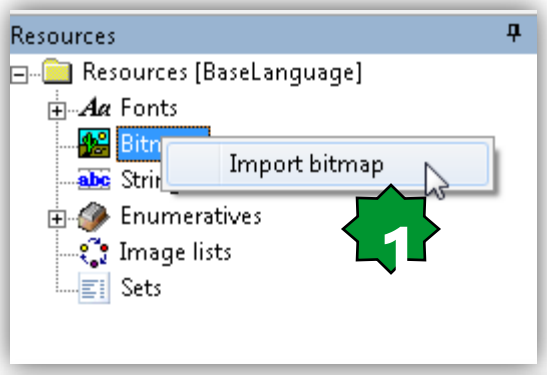


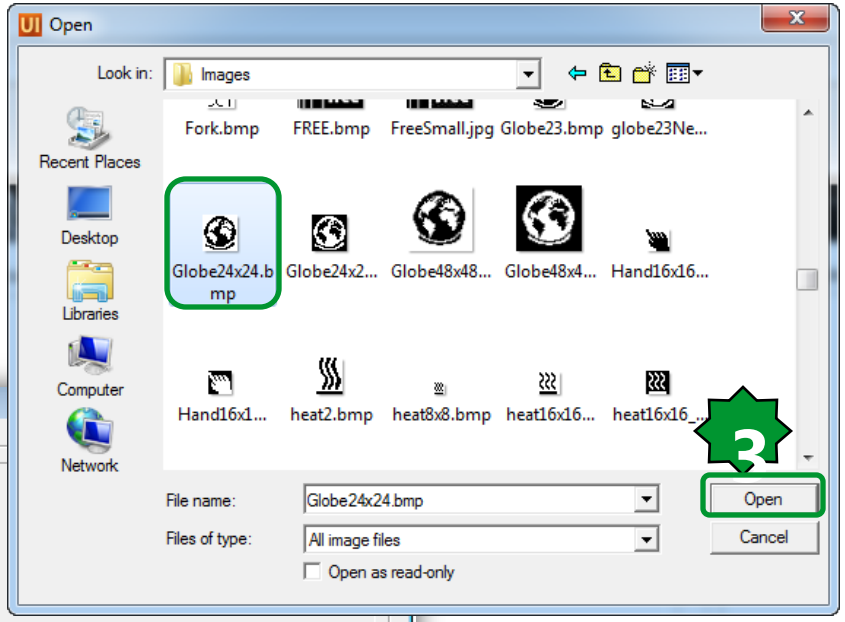


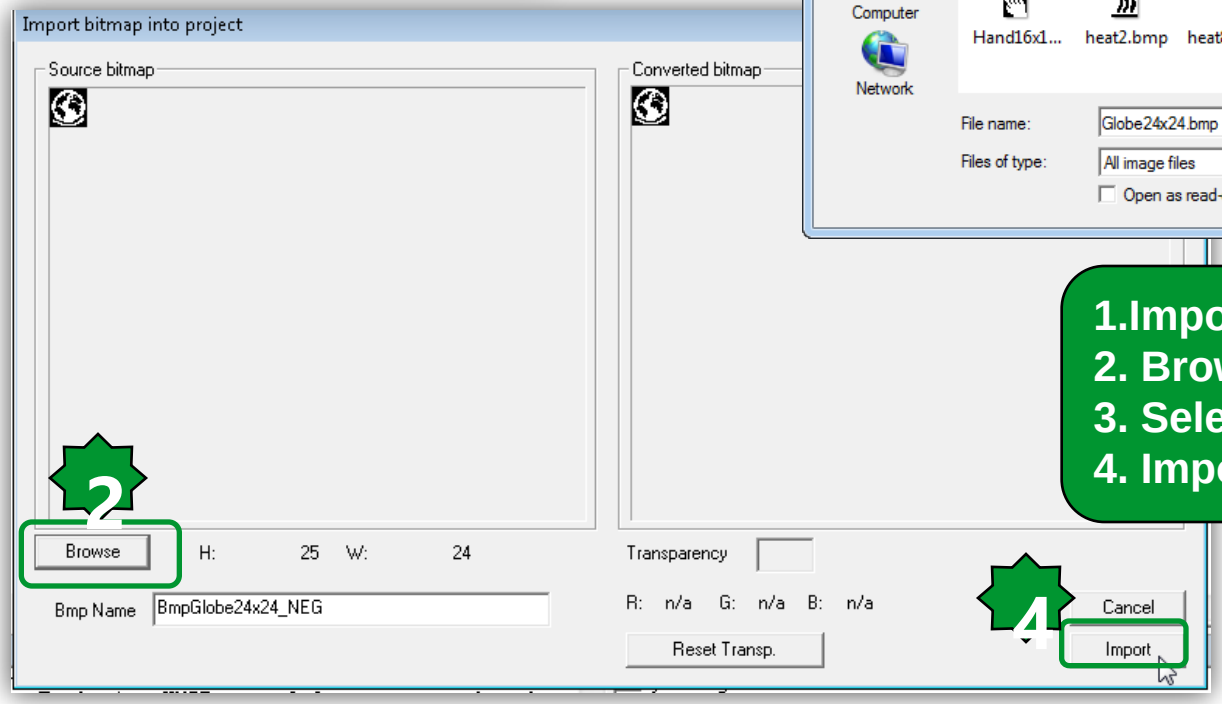


Note:
- An already opened page can't be opened: i.e. from this page it is not possible to open Main.
- Main can be reached only through a Close Action (close current and open previous page)

Image Object

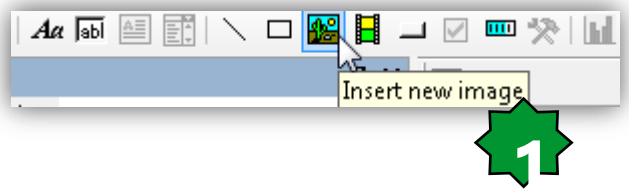




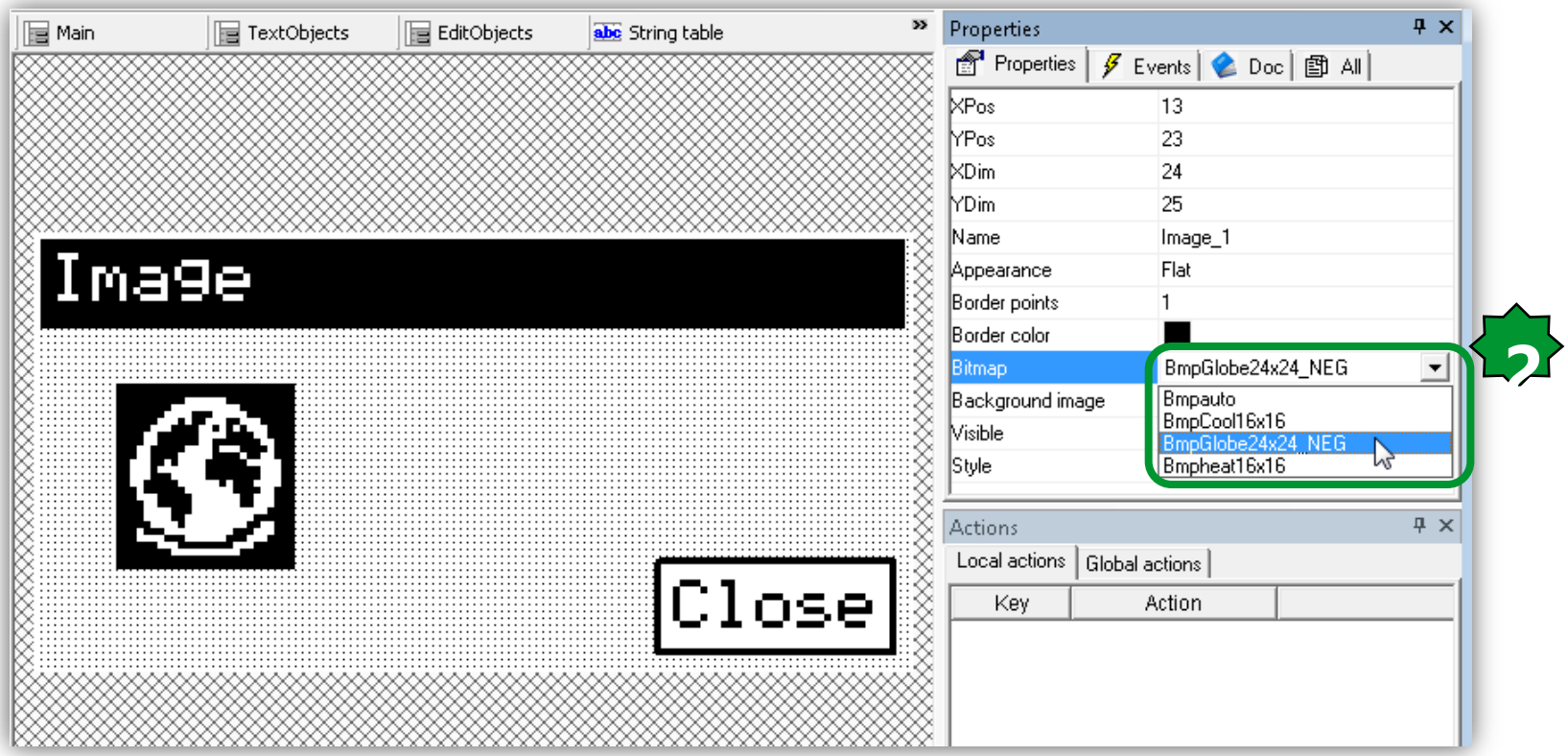


1.Import bitmap
2. Browse ► select form image library
3. Select the desired image ► Open
4. Import

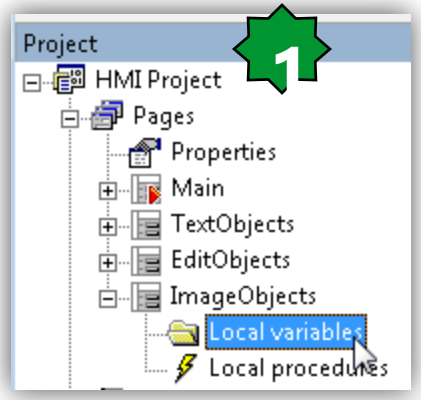
Image Object

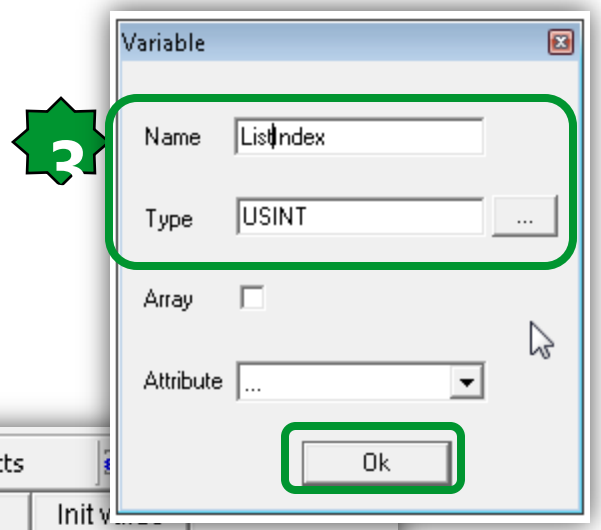


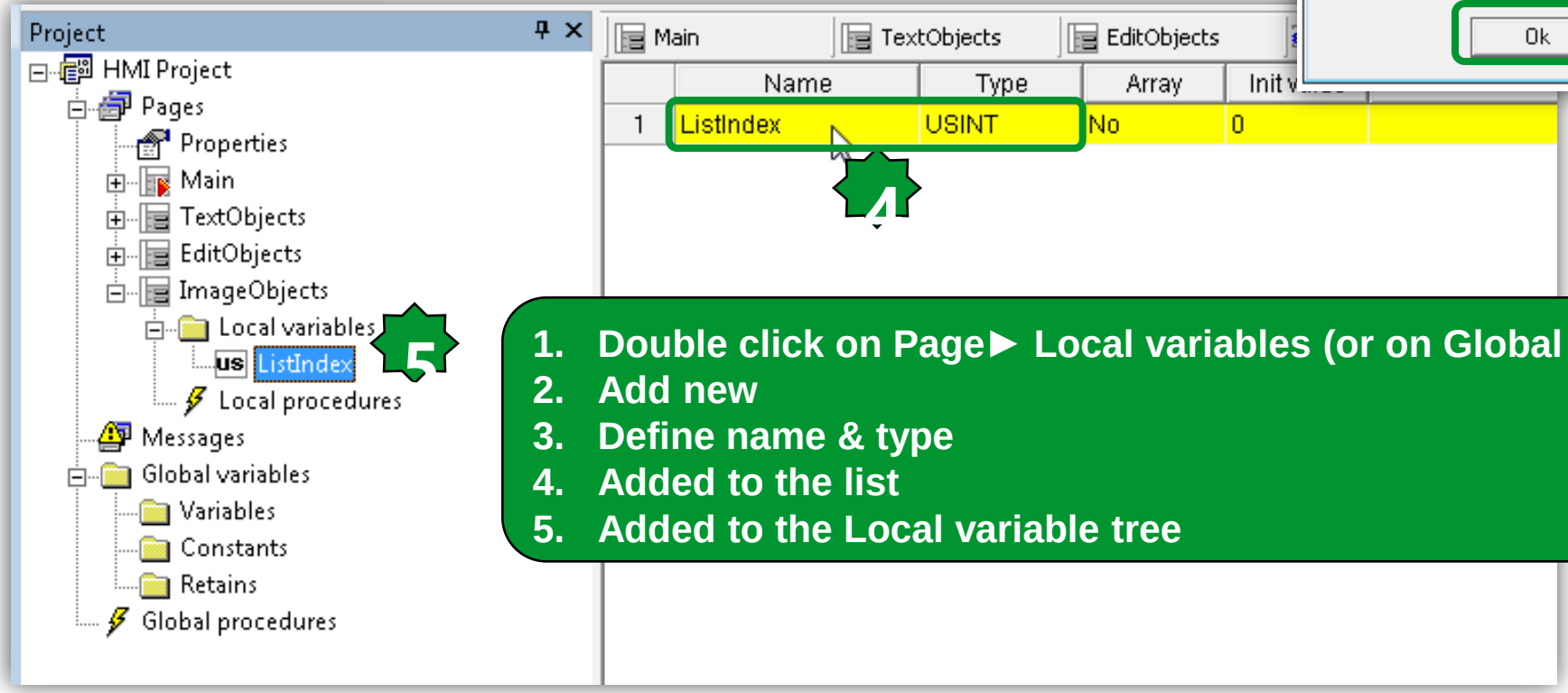
1. Inset new image
2. Properties ► Bitmap ► select the desired image from imported list



Local and Global Variables

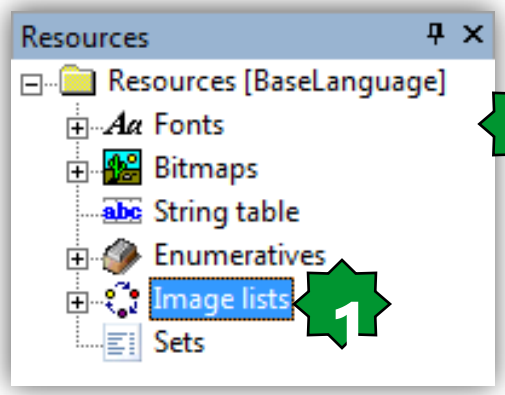


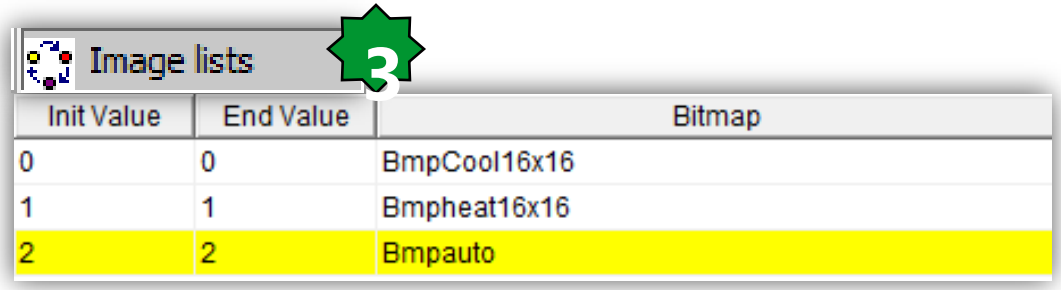


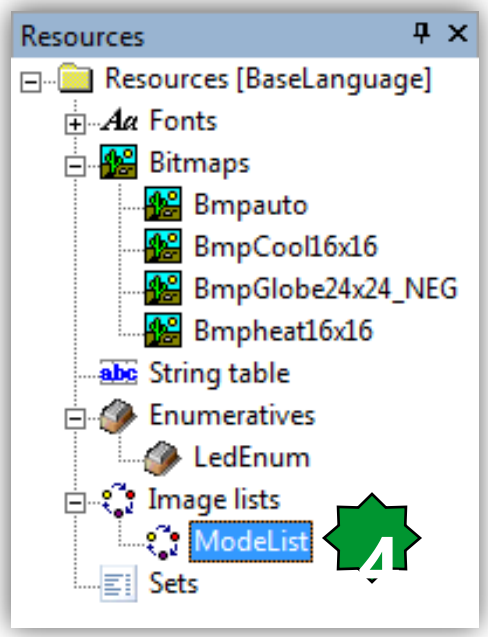


1. Double click on Page► Local variables (or on Global variables)
2. Add new
3. Define name & type
4. Added to the list
5. Added to the Local variable tree

Animation-Image lists







1. Resources► Image lists
2. Add new
3. Name/Rename it
4. Image lists tree

Animation

1

Property definition

Variable selection

@PLC.sysClock_dayweek	@PLC.Temp_UM
@PLC.sysClock_Error	ListIndex
@PLC.sysClock_hours	sysBacklight
@PLC.sysClock_minutes	sysCurrentSelectedPosition
@PLC.sysClock_month	sysKeyPressed
@PLC.sysClock_seconds	sysLangID
@PLC.sysClock_year	sysLocalLeds
@PLC.sysClockSet_daymonth	sysMSK
@PLC.sysClockSet_dayweek	sysTimer
@PLC.sysClockSet_hours	sysVER
@PLC.sysClockSet_minutes	
@PLC.sysClockSet_month	
@PLC.sysClockSet_seconds	
@PLC.sysClockSet_Upload	
@PLC.sysClockSet_year	

4

Filter: All

Value selection: None / Variable

Properties

Properties

XPos	108
YPos	21
XDim	16
YDim	17
Name	Animation_3
Appearance	Flat
Border points	0
Border color	
Image list	ModeList
Animation variable	ListIndex
Data type	USINT
Visible	TRUE

2 3 4

5. Filter variables as page locals

Properties — Properties | Events

XPos	85
YPos	22
Name	Edit_4
Appearance	Flat
Font	EWP2_6x8
Background color	
Text color	
Sel. background	
Sel. foreground	
Border points	1
Border color	
Number of chars	3
Format	%d
Alignment	Right
Access	RW
Selection order	1
Variable	ListIndex
Data type	USINT
Low limit	0
High limit	2
Refresh	TRUE
Visible	TRUE
Selectable	TRUE
Label	

5

Main | TextObjects | Edit

Image

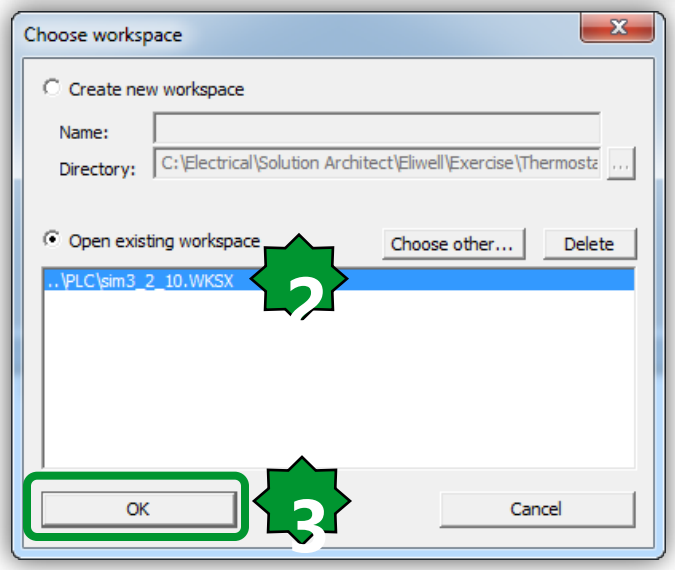
Test: 0

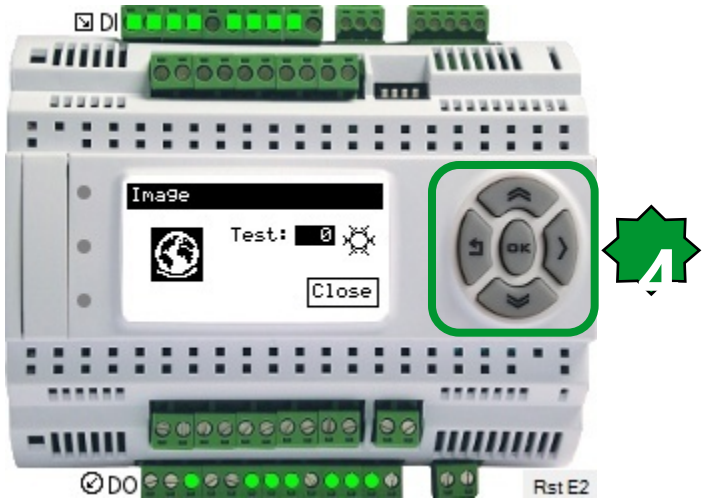
5 2

Close

1. Insert new animation
2. Rename it if needed (optional)
3. Select from Image list
4. Select from property definition

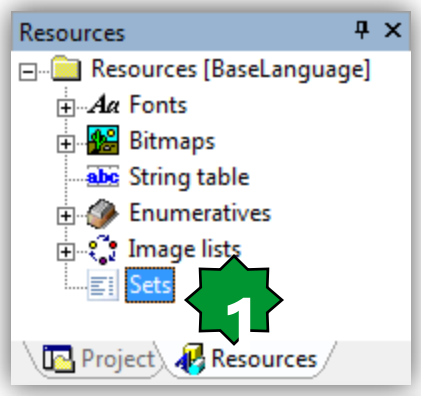
UI simulation

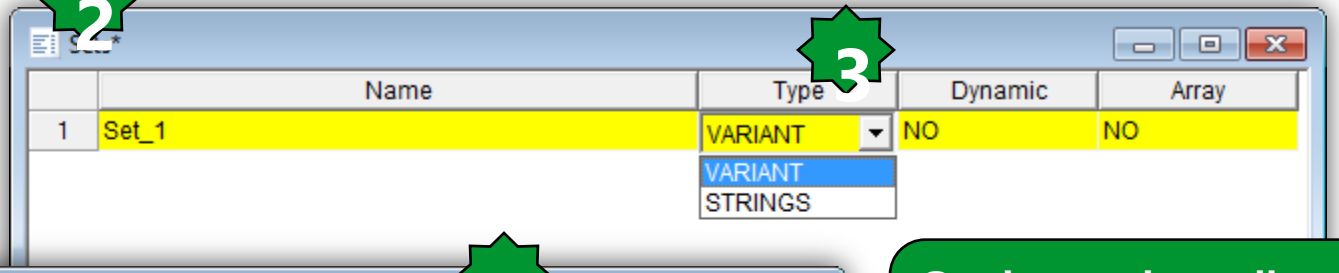


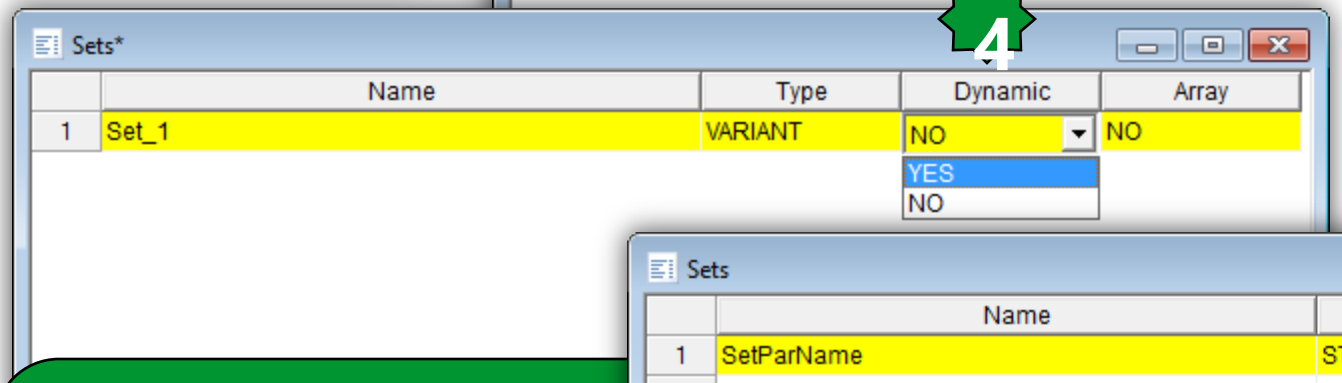


There are two parallel, identical ways to simulate, via UI or Ap

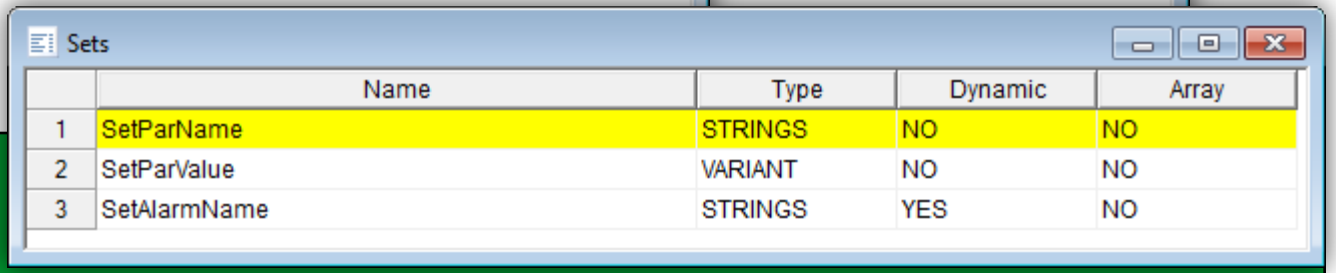
Sets creation





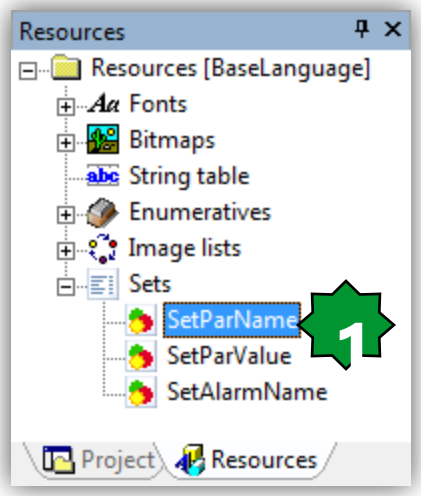


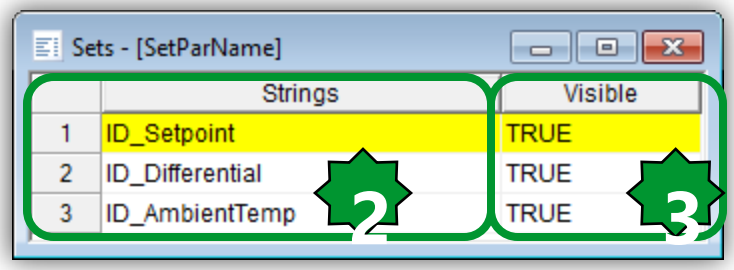
Goal: creating a list of variables & strings with scroll up/down ability



1. Double clicks
2. Add
3. Type
   3.1 Variant: Variable/parameter sets even of not equal type
   3.2 Strings: text
4. Dynamic: compresses the list when invisible variable/parameter are used

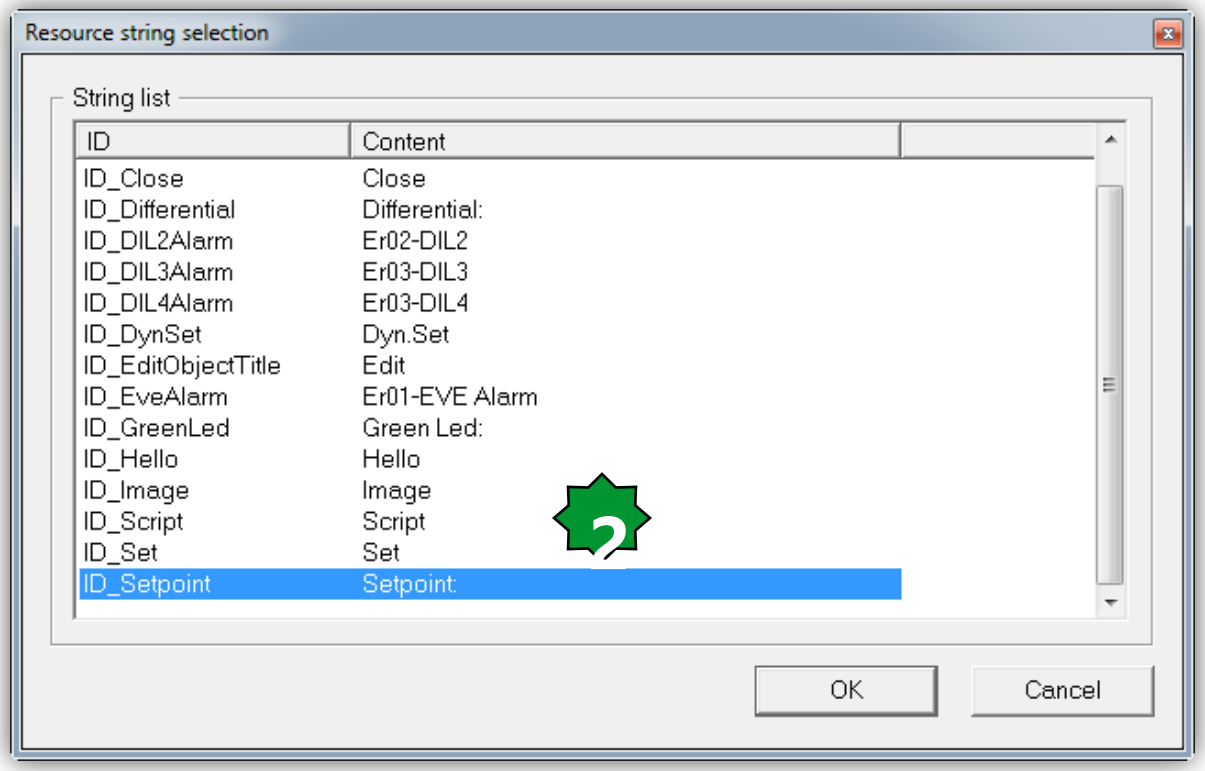
Strings Set filling



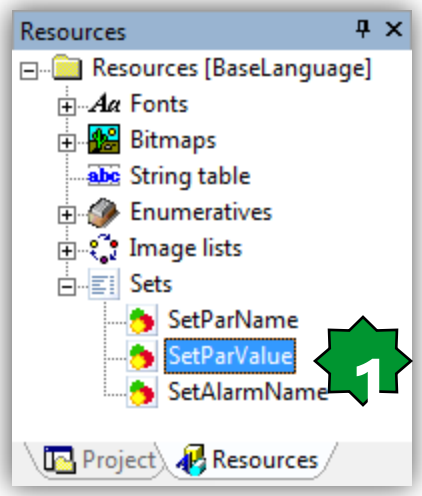


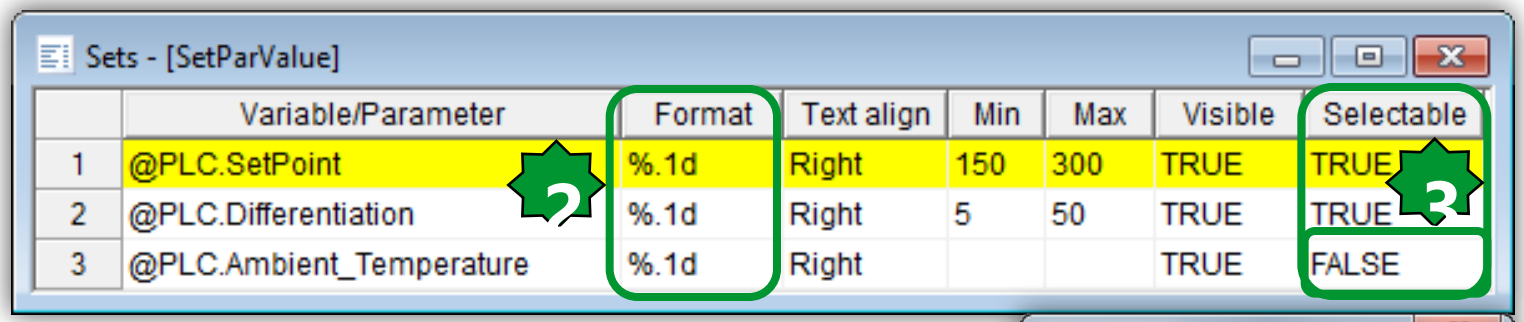
1. Double clicks
2. Select the string form the list
3. Define the visible field

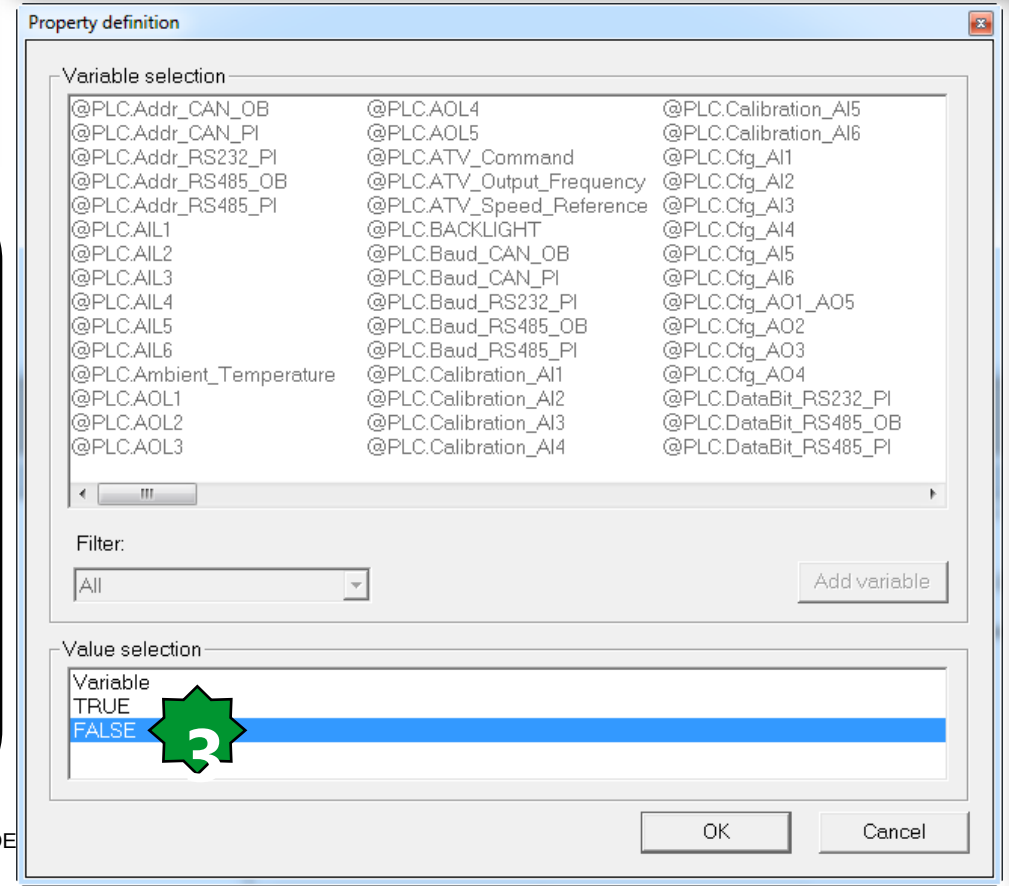
Dynamic visibility:
If the visible field is False, then empty parameter's rows will be removed from the list and the list could be shrink pack.

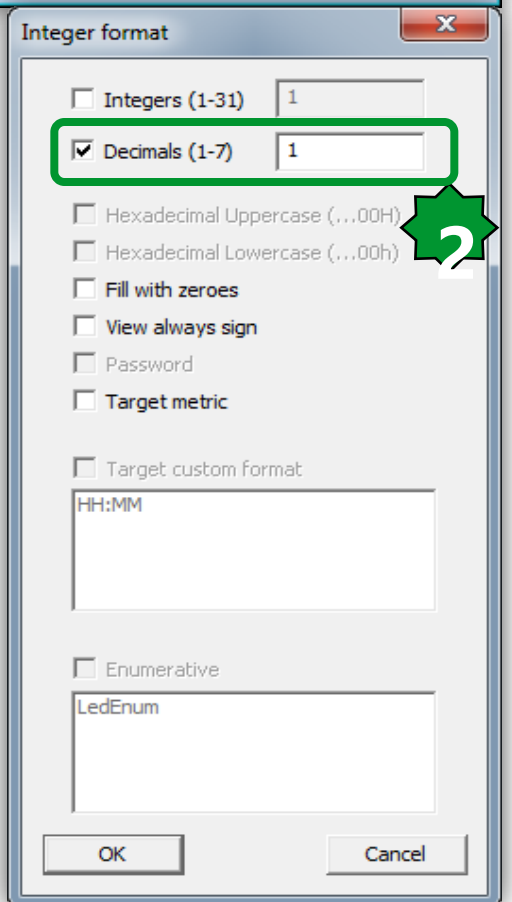


Variant Set filling



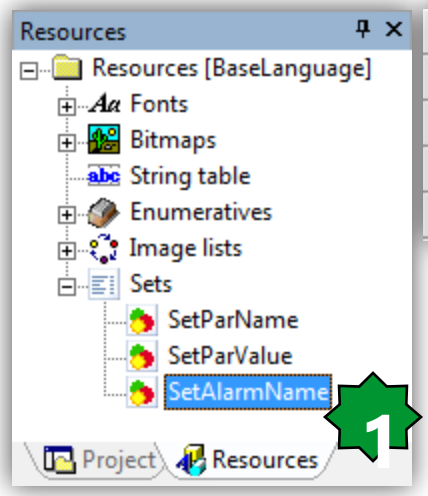






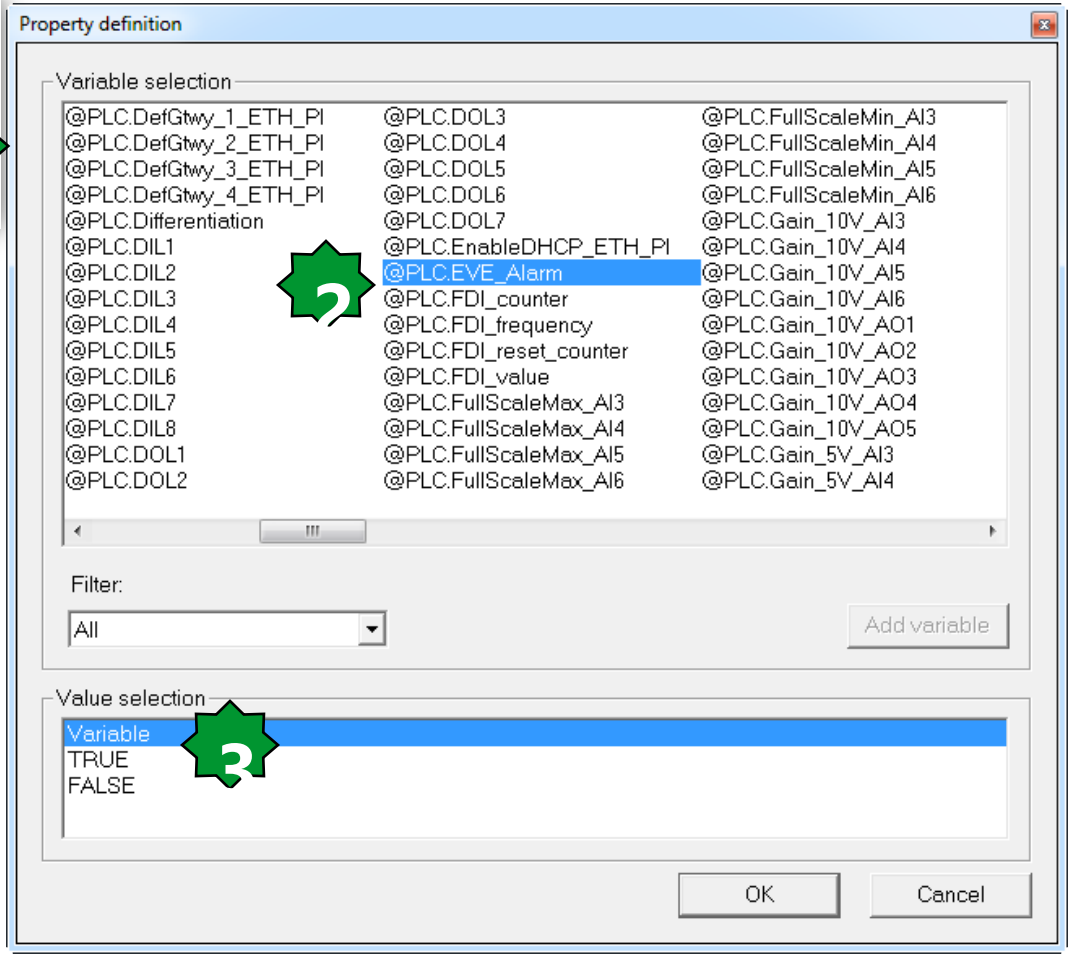
1. Sets
2. Format: One decimal point
3. Selectable False in case of RO elements

Dynamic Set filling



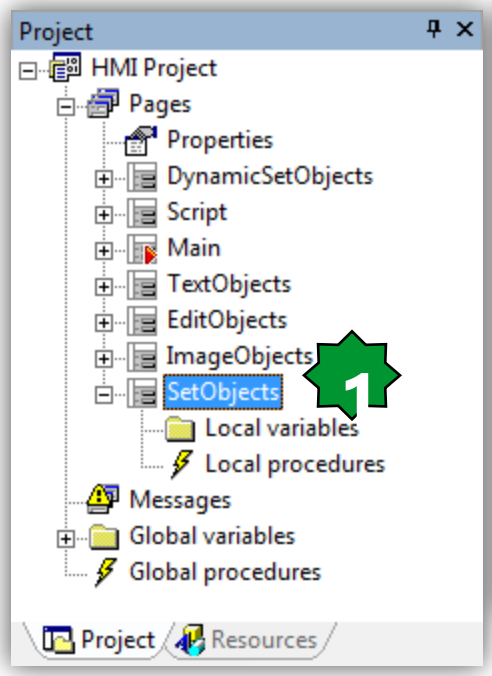
	Strings	Visible
1	ID_EveAlarm	@PLC.EVE_Alarm
2	ID_DIL2Alarm	@PLC.DIL2
3	ID_DIL3Alarm	@PLC.DIL3
4	ID_DIL4Alarm	@PLC.DIL4

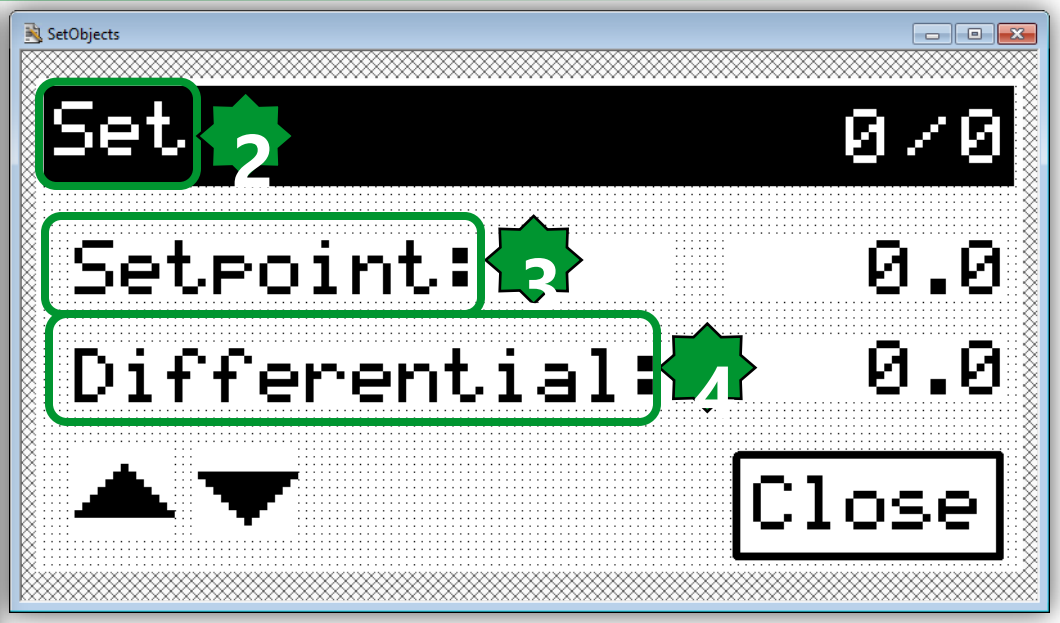
4

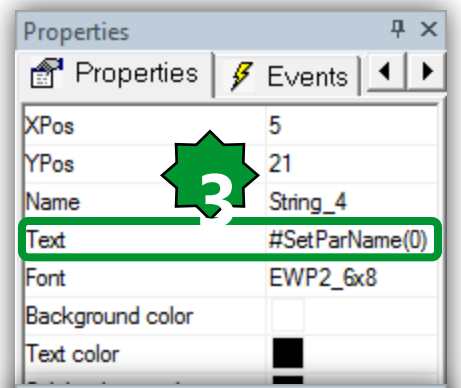


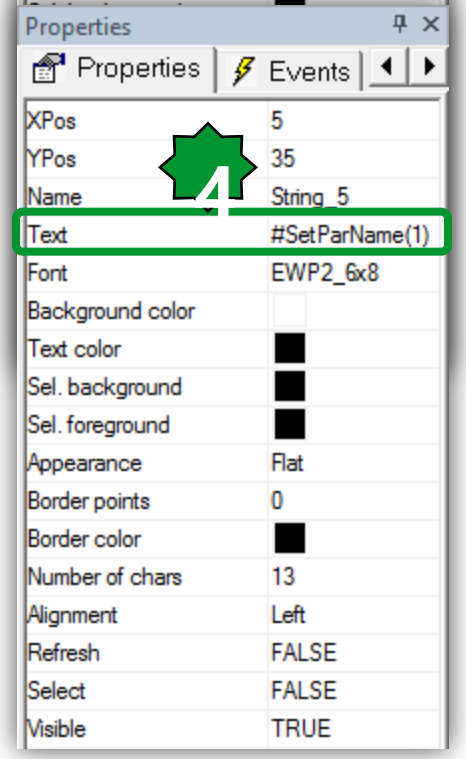
Note: the item is displayed only if Visible field is TRUE

Set Objects…









Note: The number of elements of the sets shown in the same page should be equal

Syntax:

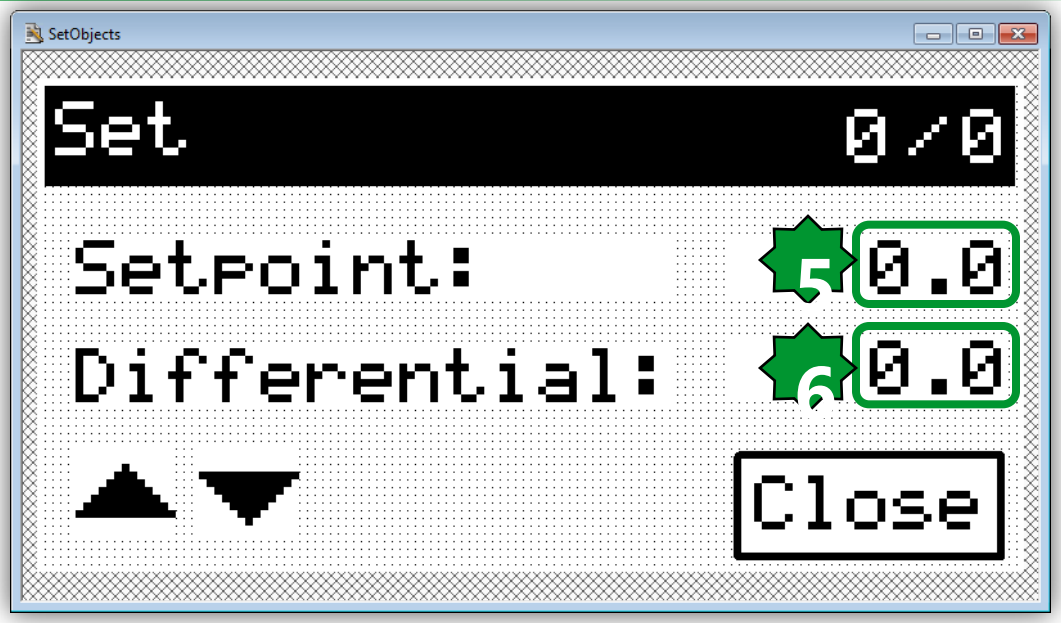
String:

#Setparname(0), #Setparname(1), …, #Setparname(element x page -1)

Variant:

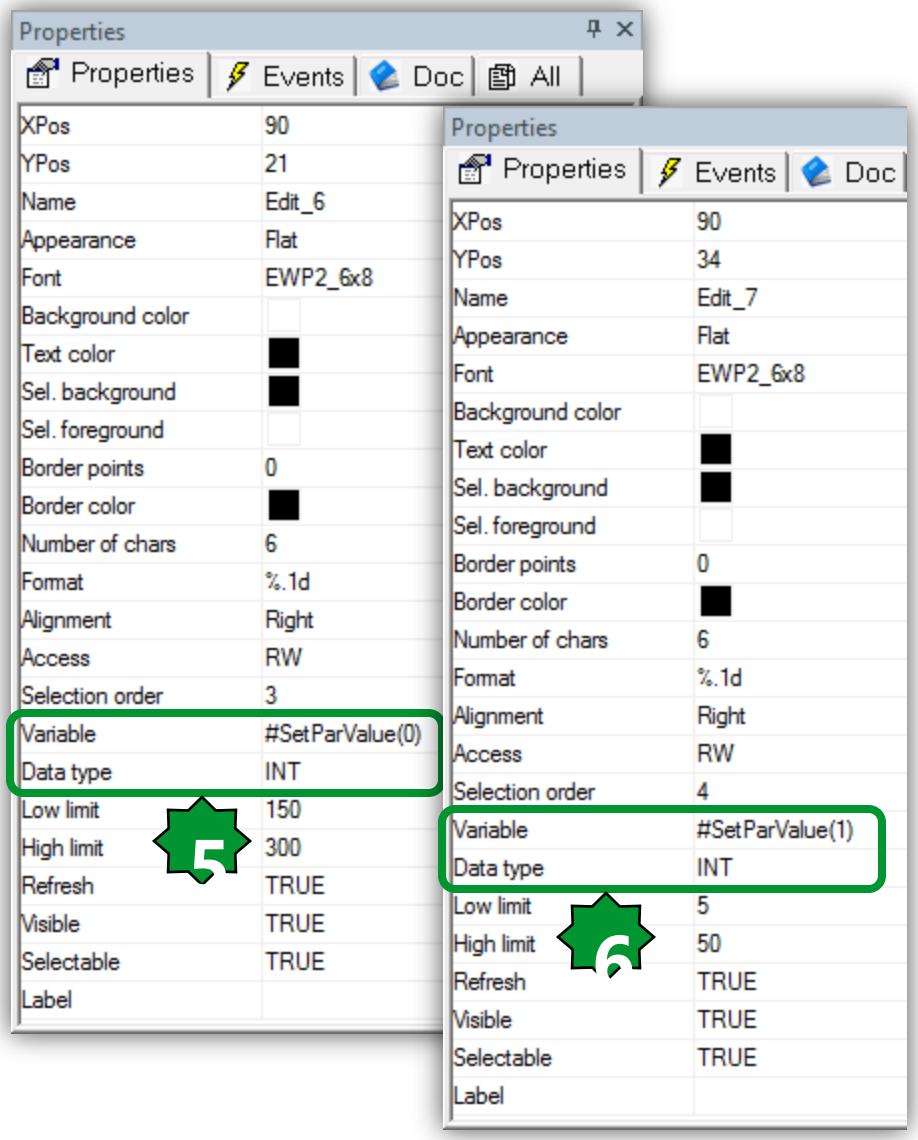
#Setparvalue(0), #Setparvalue(0), …, #Setparvalue(element x page -1)

Set Objects…

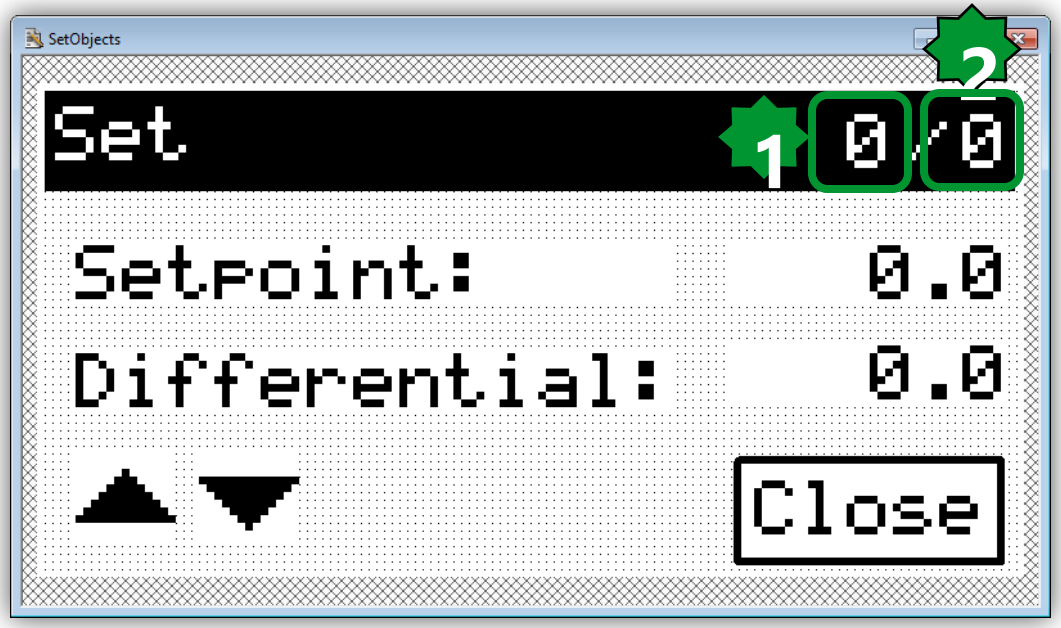


Syntax Variant:
#Setparvalue(0), #Setparvalue(0), …,
#Setparvalue(element x page -1)

- Size the text dimension based on the longest string to be displayed
- Size the Edit dimension base on the biggest digits to be displayed

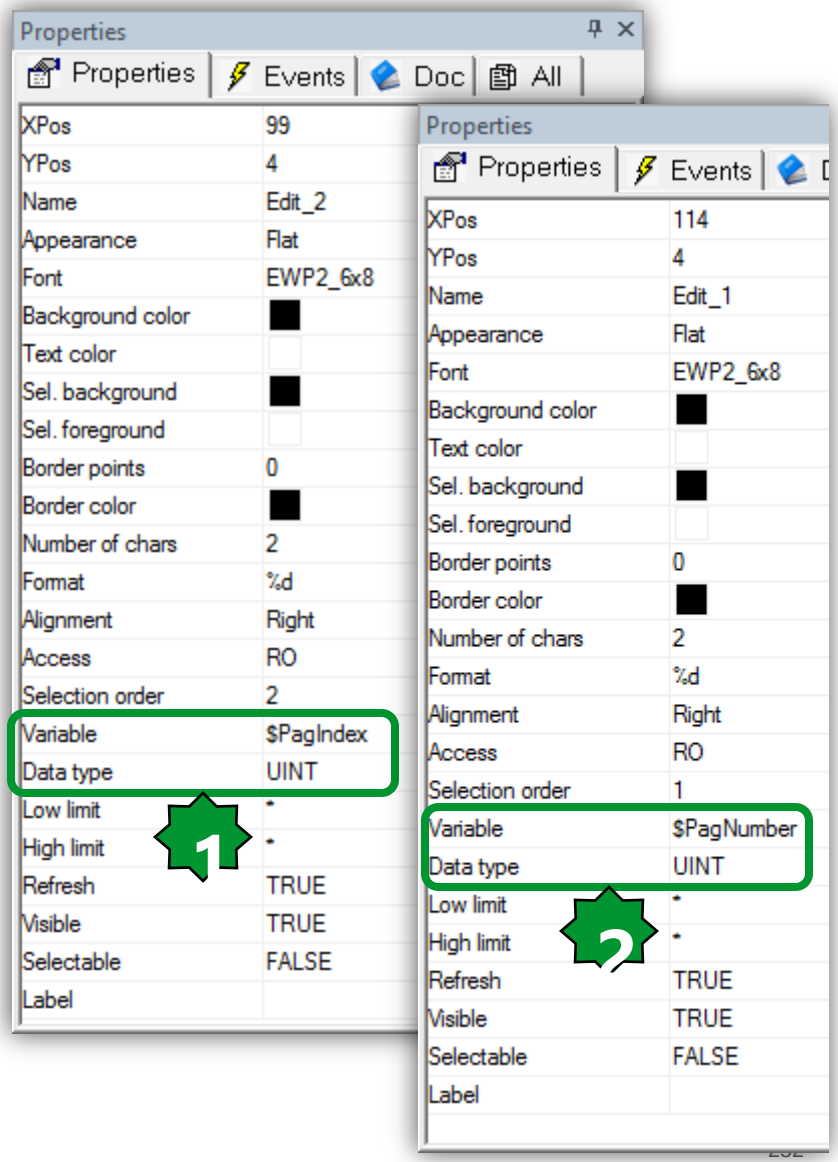


Set/Objects/Page x out of Y

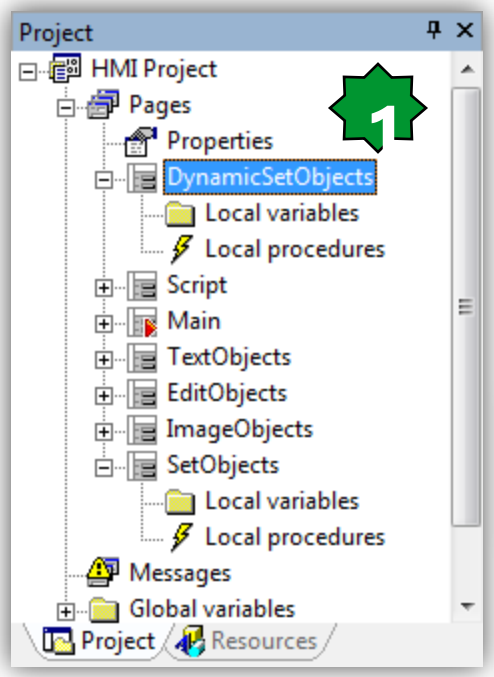


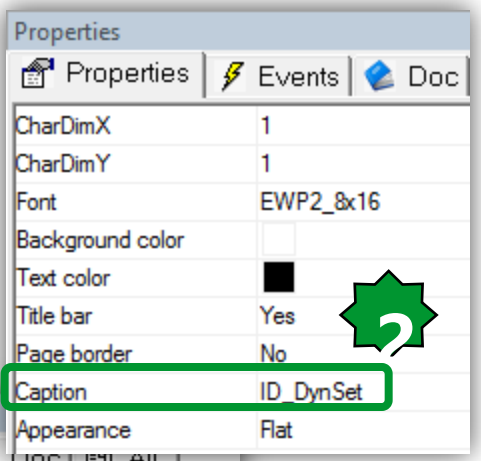
Note:
Data type will be define automatically as soon as variable recognized, if not there is a mistake in the variable name.

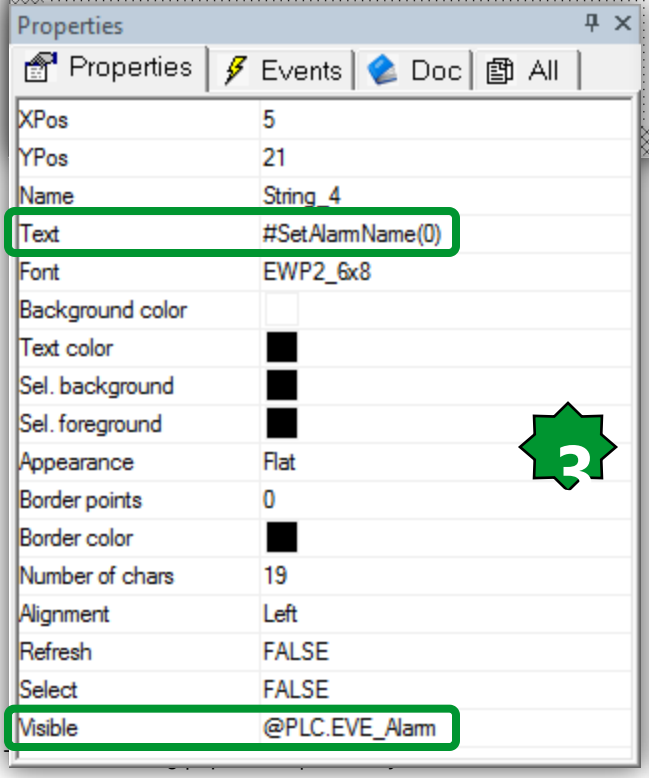
1. **$ PagIndex=Current Page Number**
2. **$PagNumber= Total pages number**

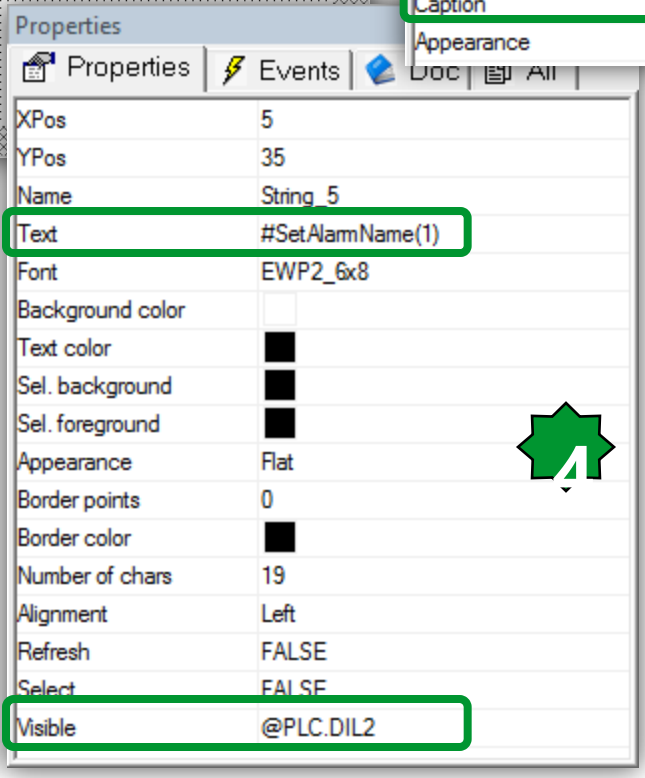


Dynamic Set Objects

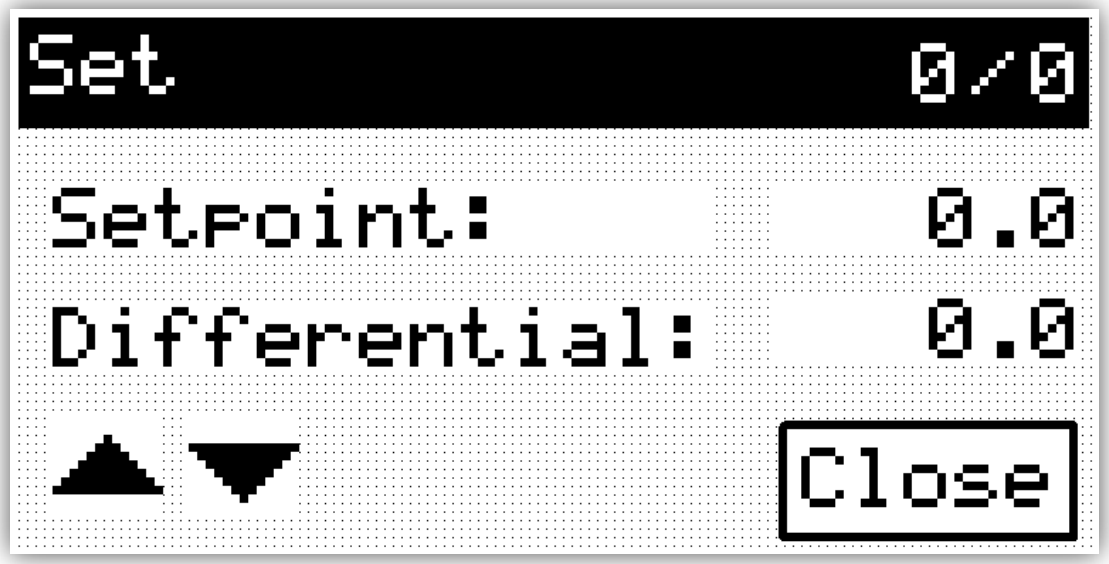


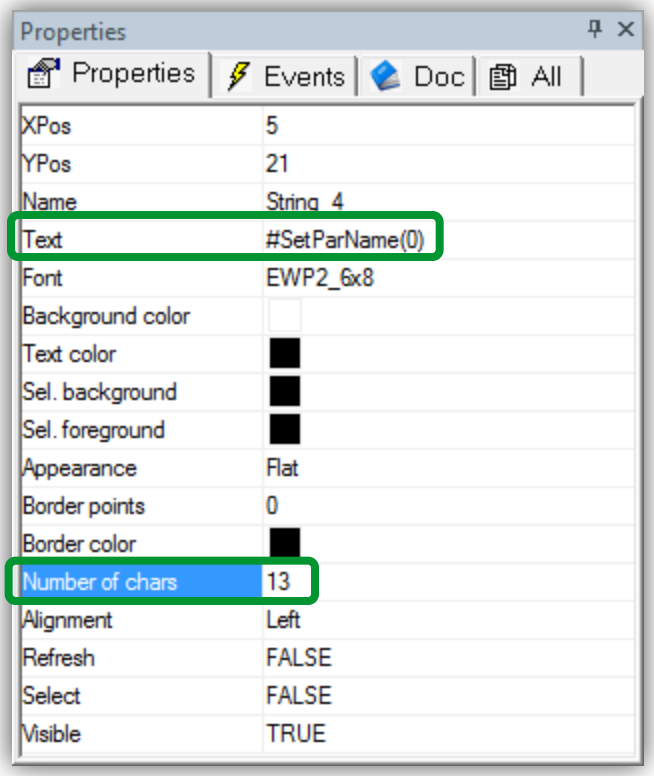






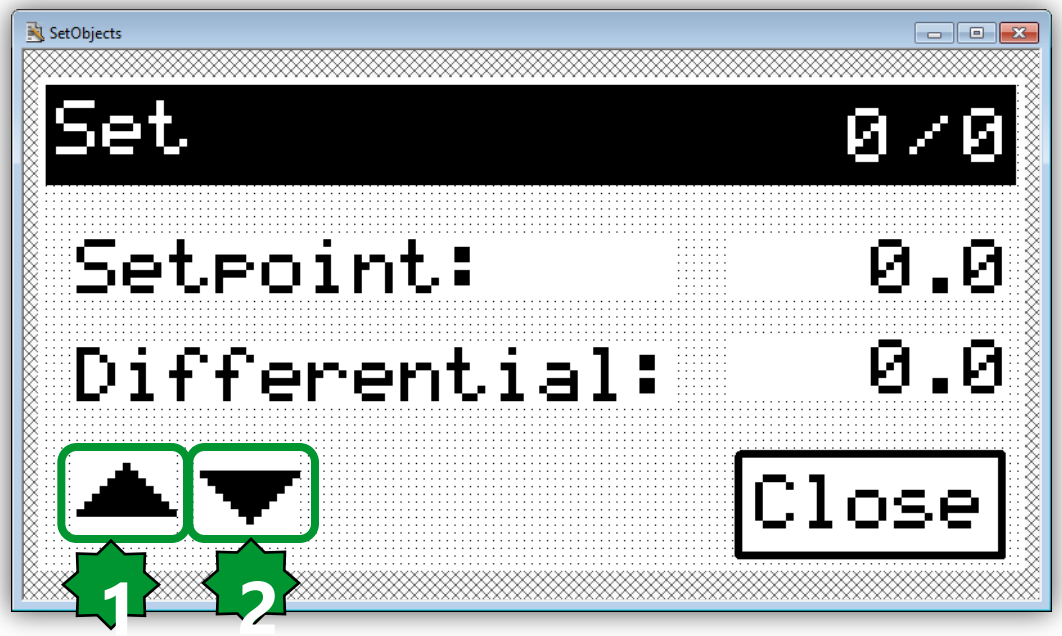
Maximum number of charachters

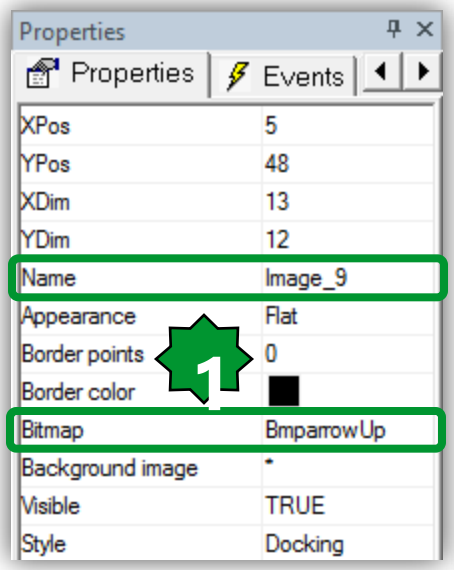


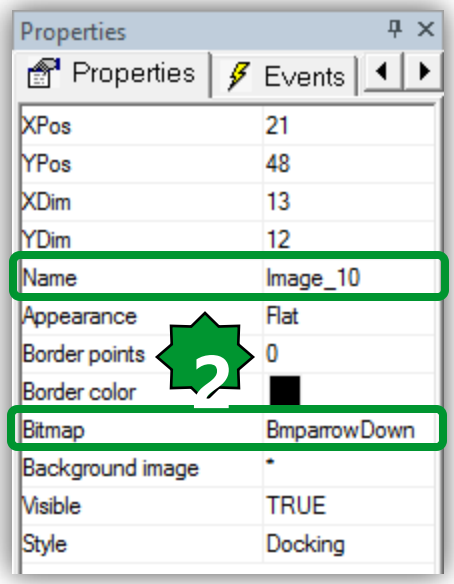


Number of Chars:
If you do not resize the related window will be 0, and you might face with refresh problem.
It is possible to set it maximum 21 charachters (>21 Chars=> out of display)

Set Objects/up & down arrows



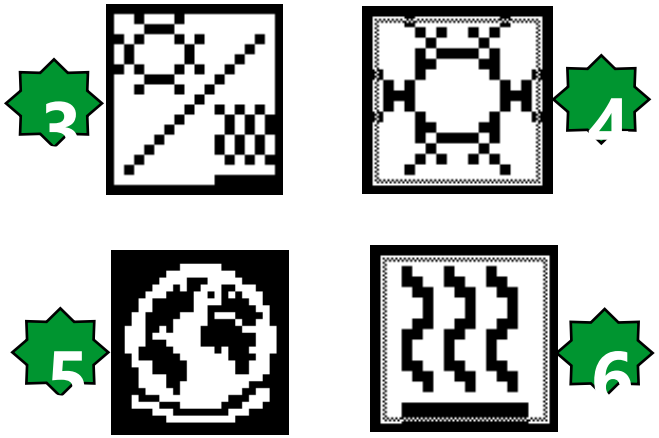




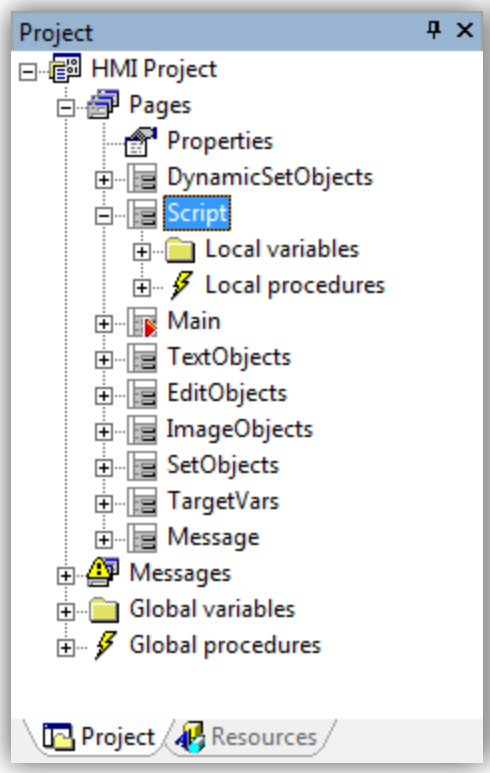
Bitmap

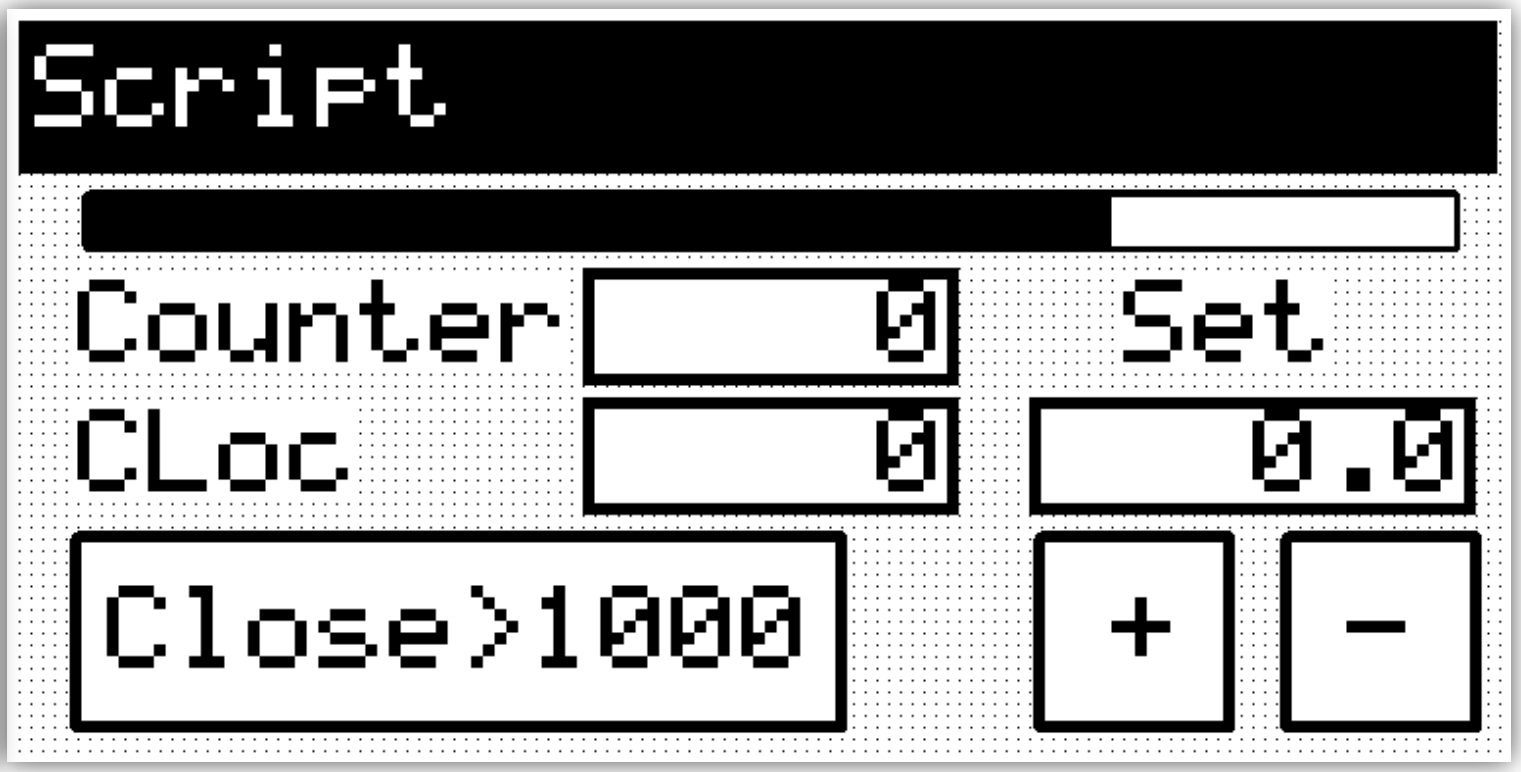
BmparrowDown
BmparrowUp
Bmpauto
BmpCool16x16
BmpGlobe24x24_
Bmpheat16x16

1. **BmparrowDown:**
2. **BmparrowUp:**
3. **Bmpauto:**
4. **BmpCool 16*16:**
5. **BmpGlobe 24*24:**
6. **Bmpheat 16*16:**

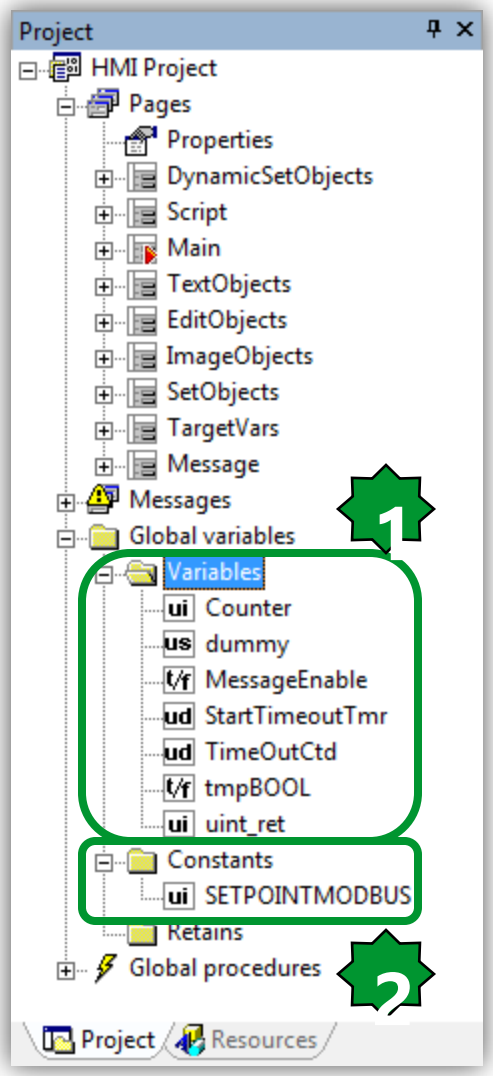


Script


Project
HMI Project
Pages
Properties
DynamicSetObjects
Script
Local variables
Local procedures
Main
TextObjects
EditObjects
ImageObjects
SetObjects
TargetVars
Message
Messages
Global variables
Global procedures
Project
Resources


Script
Counter
0
Set
CLoc
0
0.0
Close>1000
+
-

Global Variables To Be Defined



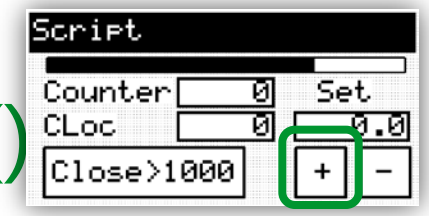
	Name	Type	Address	Group	Array	Init value	Attribute
1	Counter	UINT	Auto		No	0	..
2	StartTimeoutTmr	UDINT	Auto		No	0	..
3	dummy	USINT	Auto		No	0	..
4	uint_ret	UINT	Auto		No	0	..
5	TimeOutCtd	UDINT	Auto		No	0	..
6	MessageEnable	BOOL	Auto		No	FALSE	..
7	tmpBOOL	BOOL	Auto		No	FALSE	..

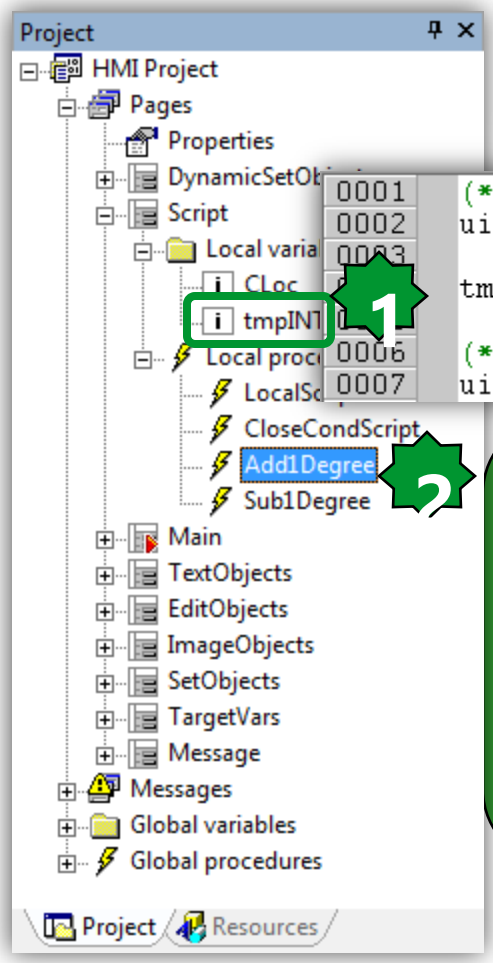
1

	Name	Type	Address	Group	Array	Init value	Attribute
1	Counter	UINT	Auto		No	0	..
2	StartTimeoutTmr	UDINT	Auto		No	0	..
3	dummy	USINT	Auto		No	0	..
4	uint_ret	UINT	Auto		No	0	..
5	TimeOutCtd	UDINT	Auto		No	0	..
6	MessageEnable	BOOL	Auto		No	FALSE	..
7	tmpBOOL	BOOL	Auto		No	FALSE	..
8	SETPOINTMODBUS	UINT	Auto		No	16384	CONSTANT

2

Video_GetParam() Video_SetParam()

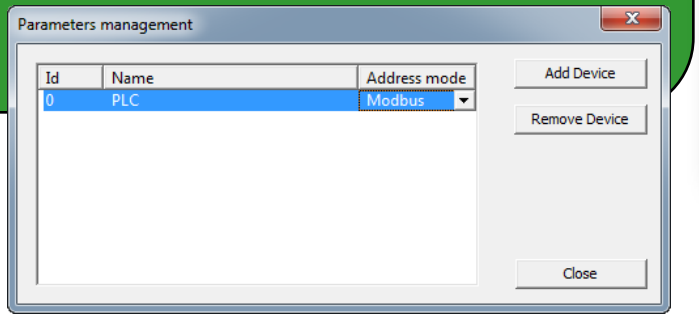


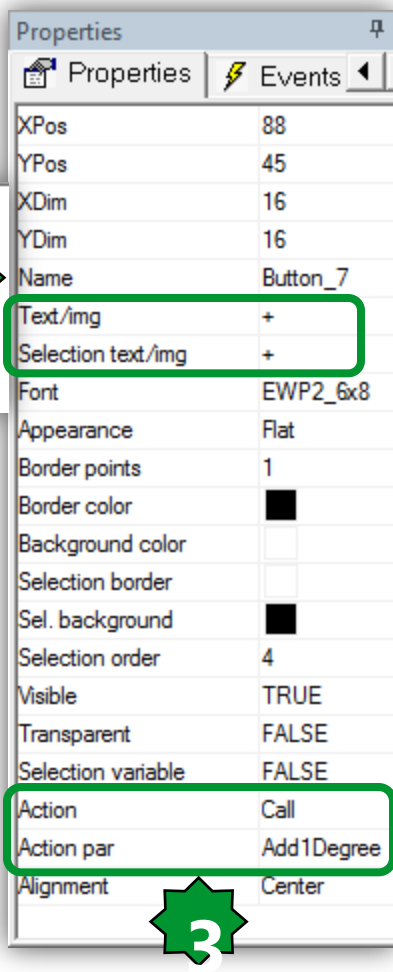


	Name	Type	Array	Init value	Description
1	CLoc	INT	No	0	
2	tmpINT	INT	No	0	

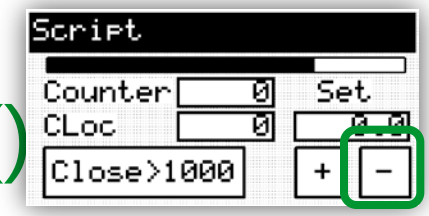
```
0001  (* Read Modbus Address SETPOINTMODBUS and copy into tmpINT *)
0002  uint_ret:= Video_GetParam( 0, SETPOINTMODBUS, 0, ?tmpINT, tyInt );
0003
0004  tmpINT := tmpINT+10;
0005
0006  (* tmpINT is written to Modbus Address SETPOINTMODBUS  *)
0007  uint_ret:= Video_SetParam( 0, SETPOINTMODBUS, 0, ?tmpINT, tyInt );
```

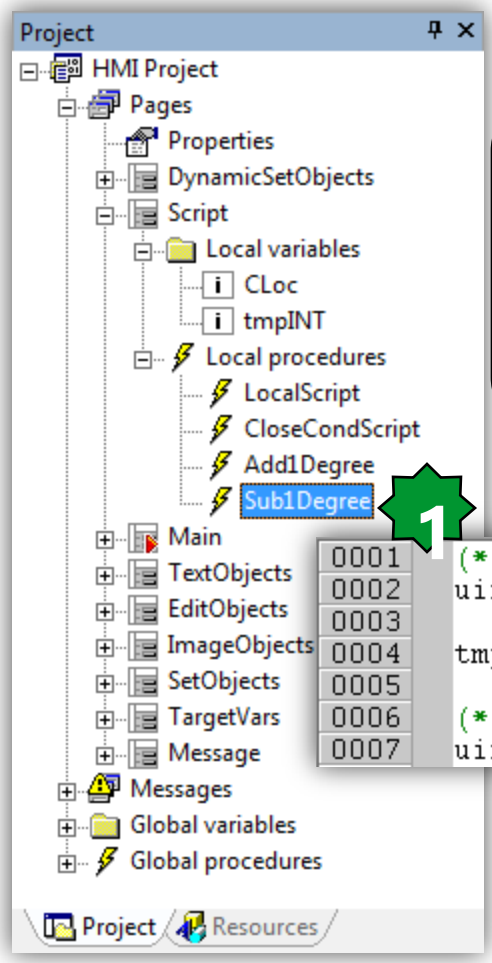
- **Video_GetParam() read and value from the parameter file Id, with address SETPOINTMODBUS. Result is copied into tmpINT.**
- **Video_SetParam() write tmpINT**





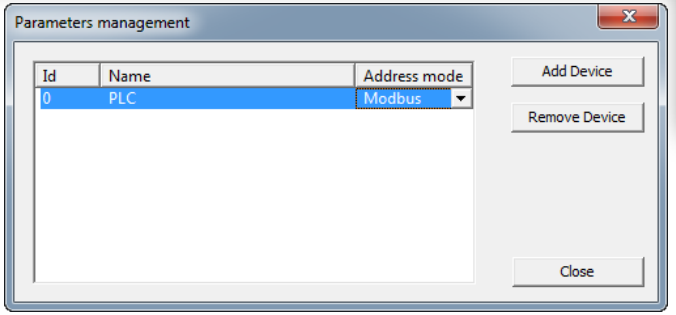
Video_GetParam() Video_SetParam()

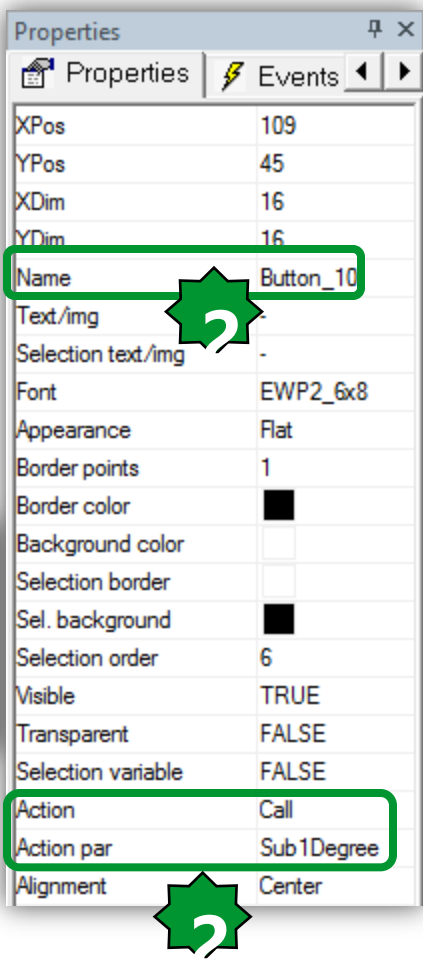




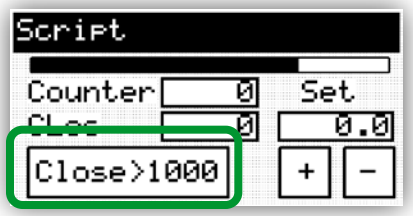
- **Video_GetParam() read and value from the parameter file Id, with address SETPOINTMODBUS. Result is copied into tmpINT.**
- **Video_SetParam() write tmpINT**

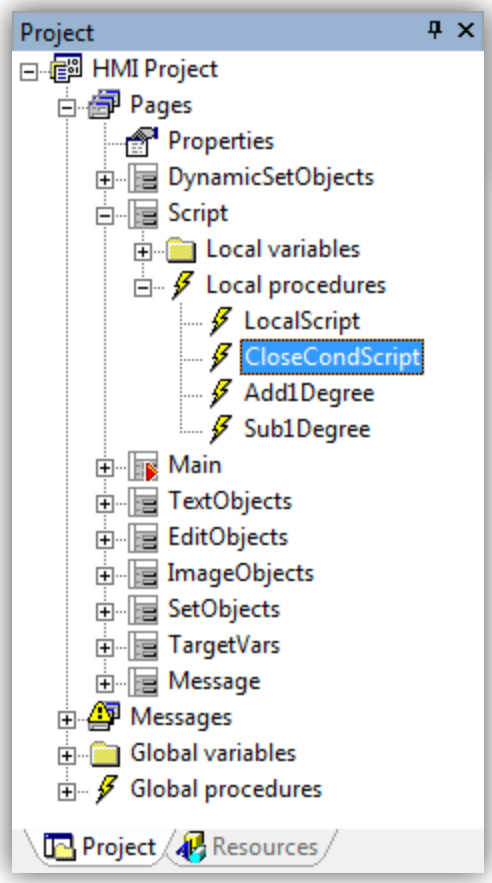
```
0001  (* Read Modbus Address SETPOINTMODBUS and copy into tmpINT *)
0002  uint_ret:= Video_GetParam( 0, SETPOINTMODBUS, 0, ?tmpINT, tyInt );
0003
0004  tmpINT := tmpINT-10;
0005
0006  (* tmpINT is written to Modbus Address SETPOINTMODBUS  *)
0007  uint_ret:= Video_SetParam( 0, SETPOINTMODBUS, 0, ?tmpINT, tyInt );
```





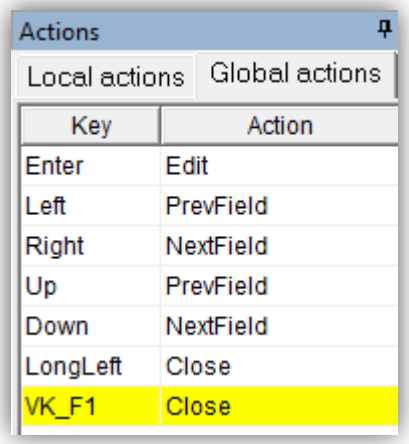
Script: Video_SendEvent()

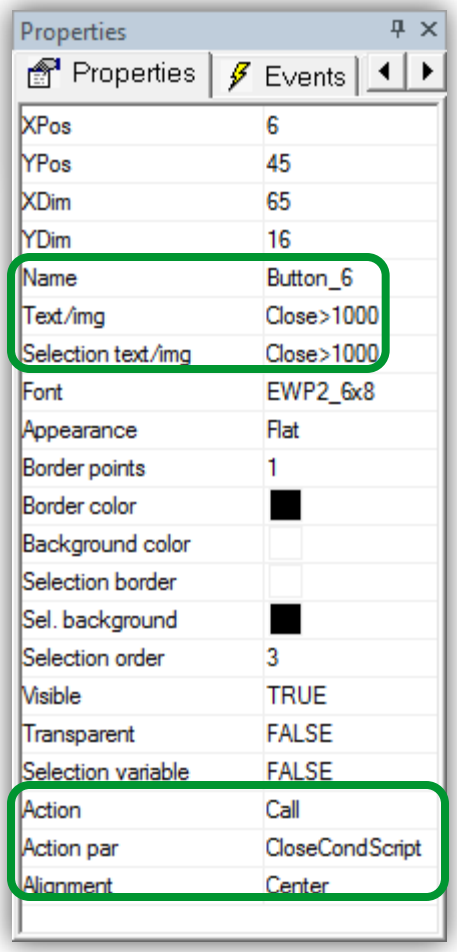




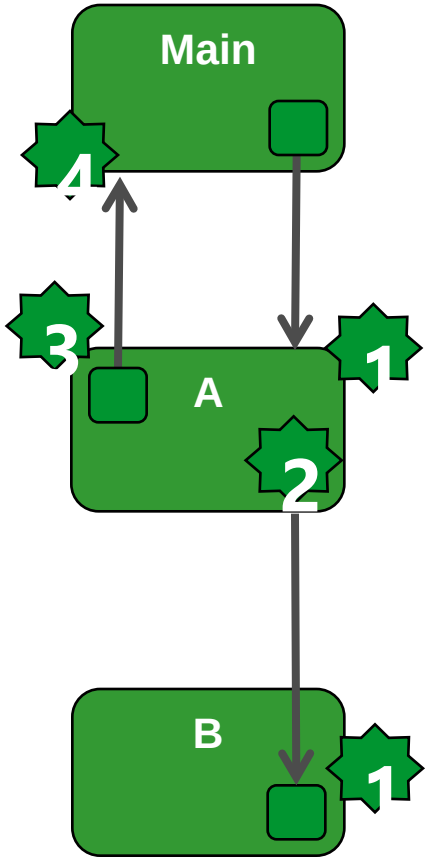

```
0001  if Counter>1000 then
0002      (*Close Current Page for Timeout *)
0003      dummy:=Video_SendEvent(kWM_KEY,kKEY_LongLeft);
0004      Counter := 0;
0005  end_if;
```

- **When counter is greater than 1000 the script emulates a Long Left button press using function Video_SendEvent().**
- **Long Left action causes a page Close**

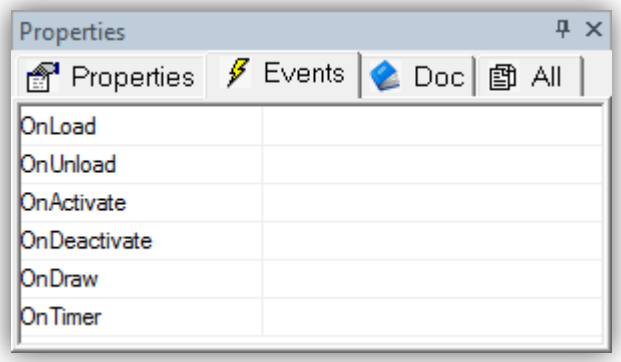




Pages Events

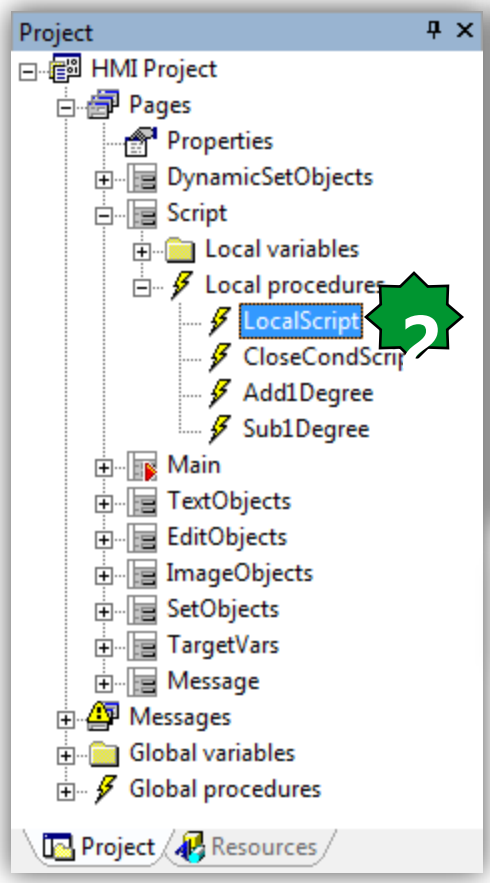


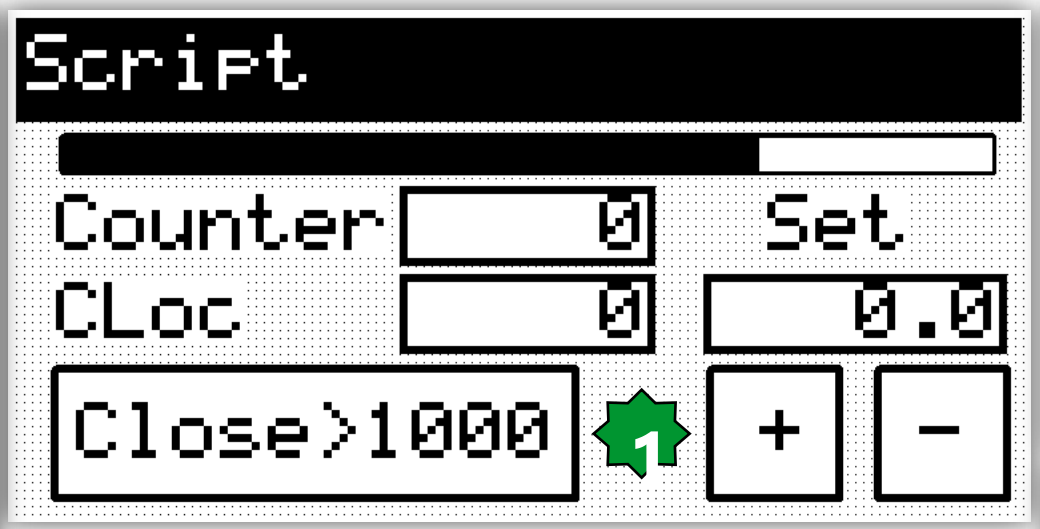
1. OnLoad
2. OnDeactivated
3. OnUnload
4. OnActivate

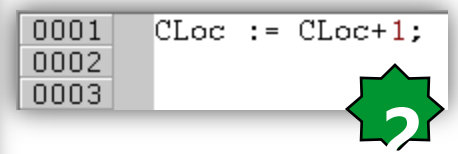


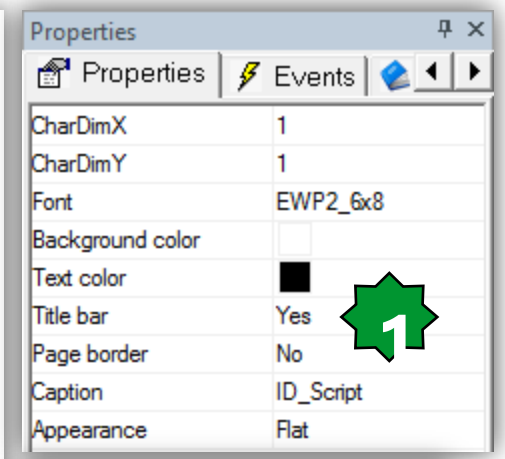
Event	Description
OnLoad	On loading this page, i.e. when calling from parent page.
OnUnload	On closing this page, when the page returns and the parent page will be restored.
OnDeactivate	On calling a child page and the current page is no more active. This event does not exist in main page.
OnActivate	When the previous opened child page will be closed. This event does not appear in leaf page, i.e in the pages which do not call child pages.
OnDraw	When the page starts drawing all the objects. The page has just drawn border, background, and title.
OnTimer	Asynchronous event. The user can link a procedure and it will be executed cyclically.

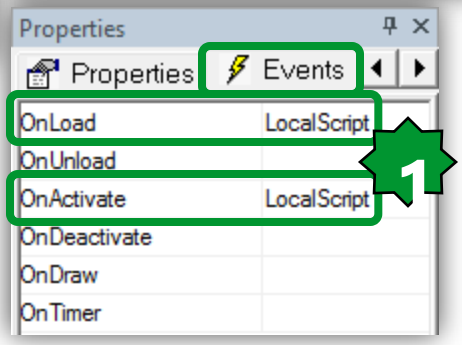
Pages Events





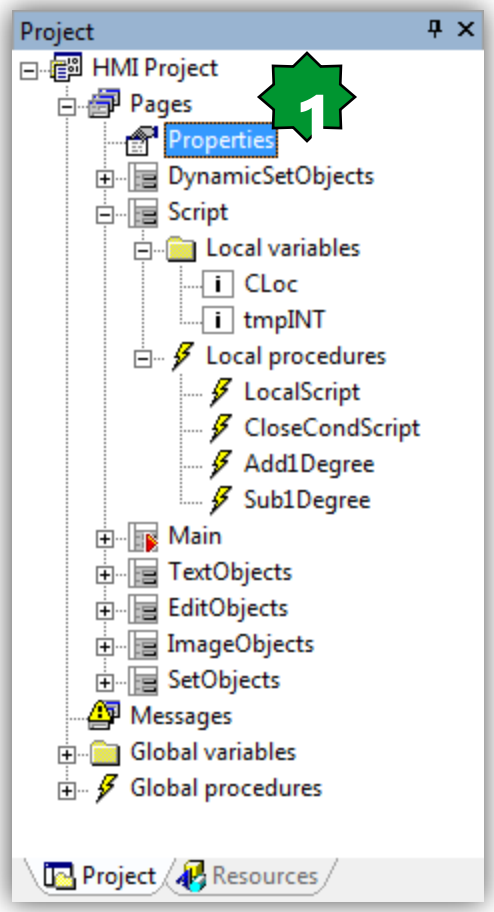


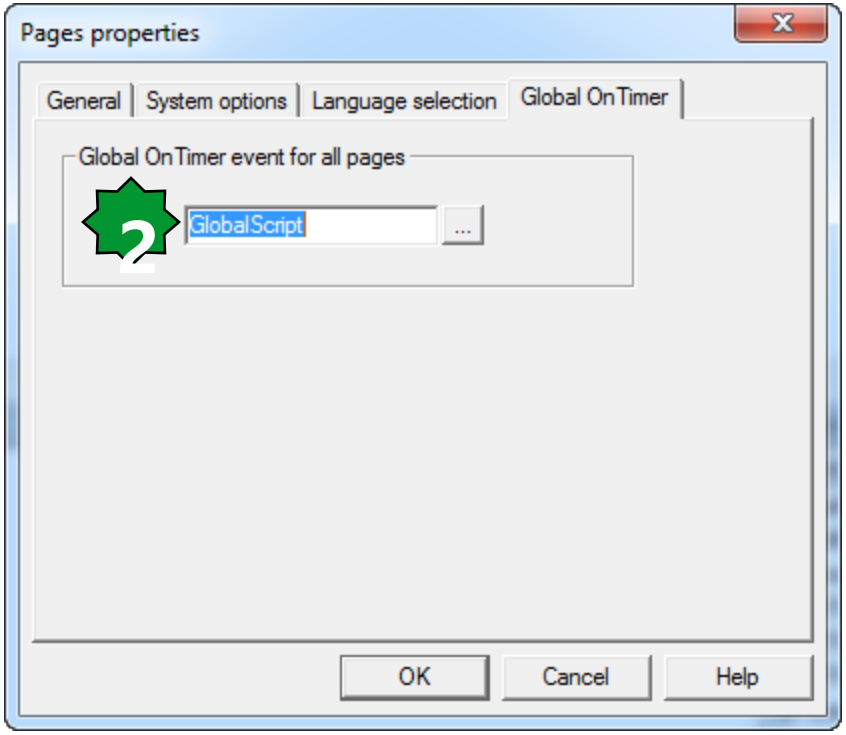


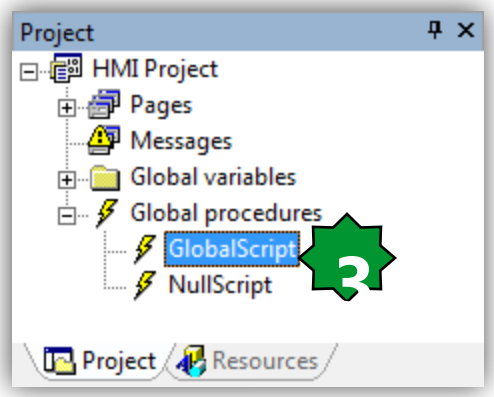


Event: CLoc counts the number of times Script page is displayed

Global On Timer Script

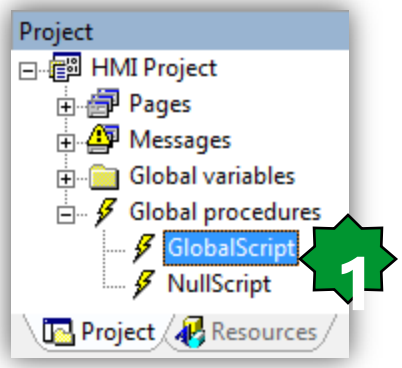




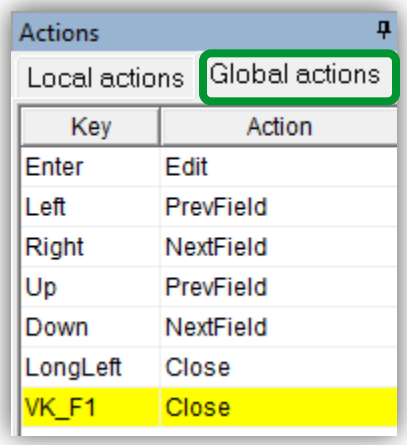


Note:
Global On Timer always runs in background

Global On Timer Script – Page Timeout



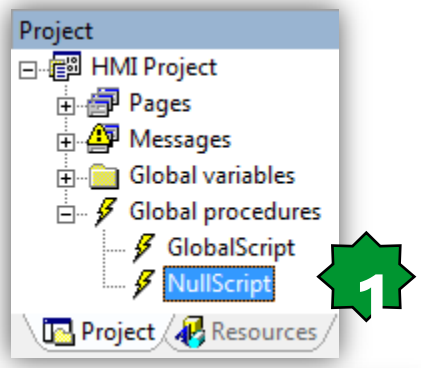
Force a page close if no button has been pressed for 20sec



1

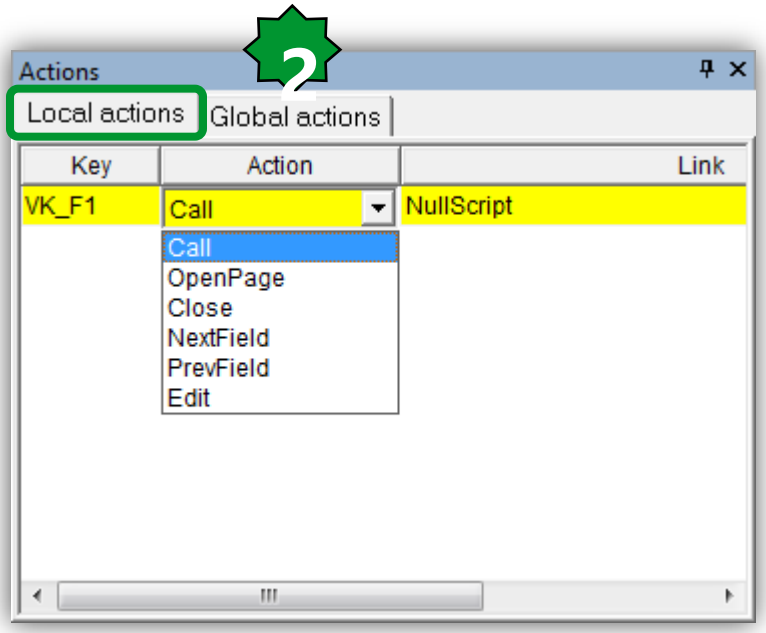
```
0001
0002  (*-----------------------------------------------------------*)
0003  (* 20sec Timeout Inactivity based on sysTimer                *)
0004  (*-----------------------------------------------------------*)
0005
0006  IF (sysKeyPressed AND 2)=0 AND sysKeyPressed<>0 THEN
0007
0008      (* A key has been pressed and not yet detected *)
0009      StartTimeoutTmr := sysTimer;
0010
0011  END_IF;
0012
0013
0014  IF (sysTimer-StartTimeoutTmr) > 20000 THEN
0015
0016      (*Close Current Page for Timeout *)
0017      dummy:=Video_SendEvent(kWM_KEY,kKEY_VK_F1);
0018      StartTimeoutTmr := sysTimer;
0019
0020  END_IF;
0021
0022  (* Raise the second bit in order to detect new key press     *)
0023  (* In this way sysKeyPressed can be used also by other scripts  *)
0024
0025  sysKeyPressed := sysKeyPressed OR 2;
0026  (*-----------------------------------------------------------*)
0027
0028  (*-----------------------------------------------------------*)
0029  (* Seconds to next timeout event [sec/10]                   *)
0030  (*-----------------------------------------------------------*)
0031  TimeOutCtd := (20000-(sysTimer-StartTimeoutTmr))/100;
0032  (*-----------------------------------------------------------*)
0033
0034  (*-----------------------------------------------------------*)
0035  (* Counter used in Script Page                              *)
0036  (*-----------------------------------------------------------*)
0037  IF Counter<10000 THEN
0038      Counter := Counter + 1;
0039  END_IF;
```

Global On Timer Script – Page Timeout



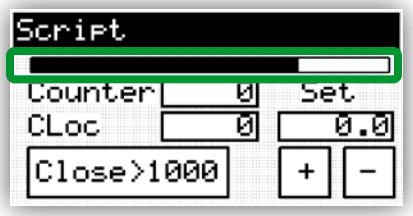

```
0001  Counter:=Counter;
0002
```

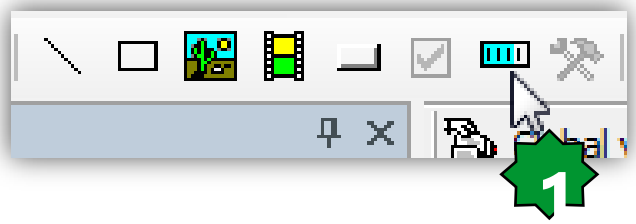
Script can not be empty

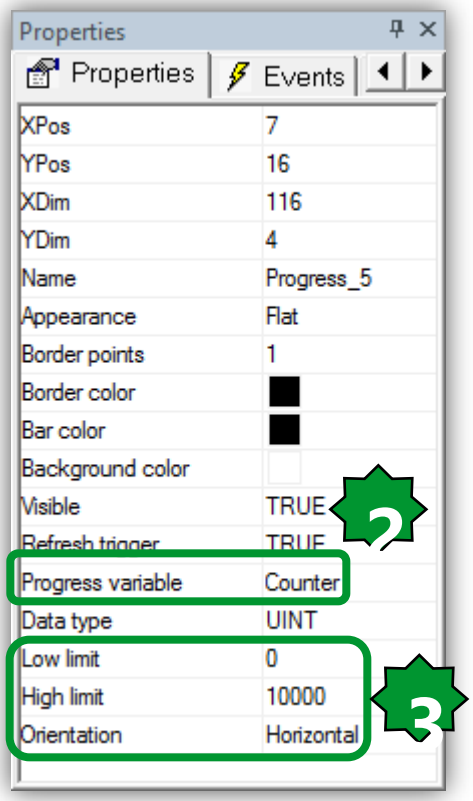


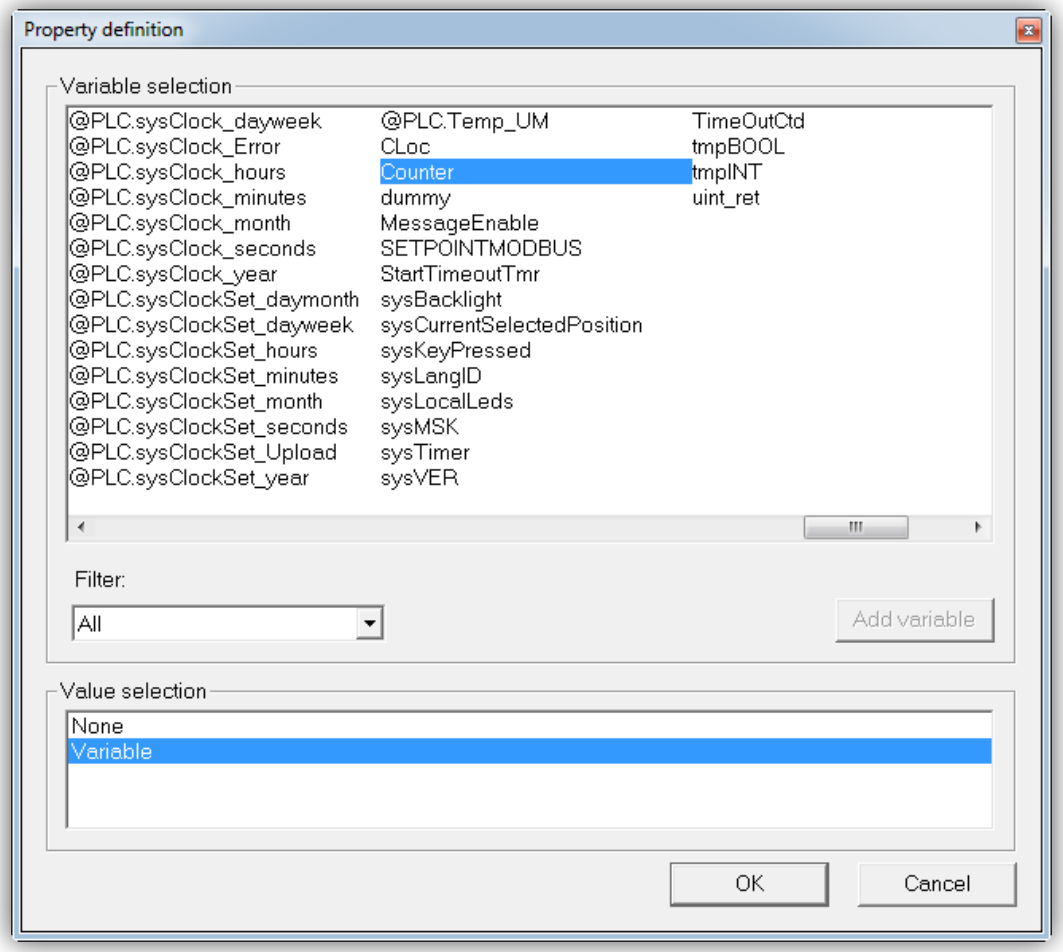
Main Page can‘t be closed: Global actions is bypassed by local ones

Progress Bar

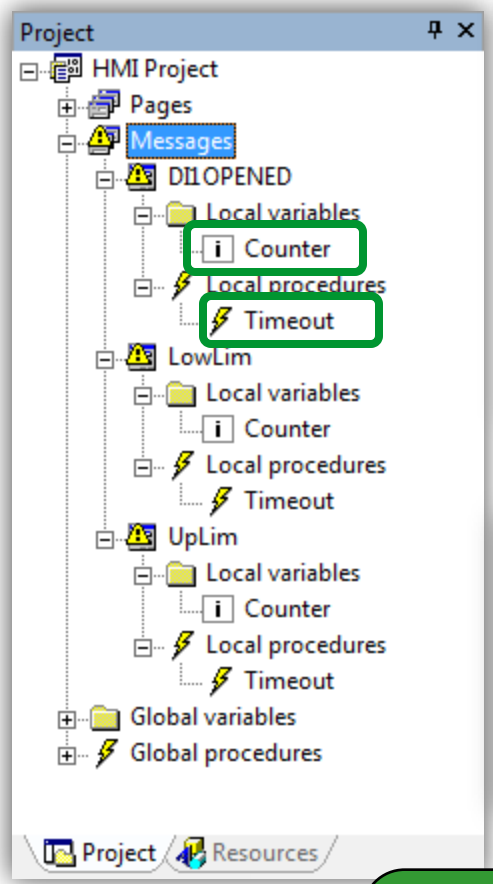


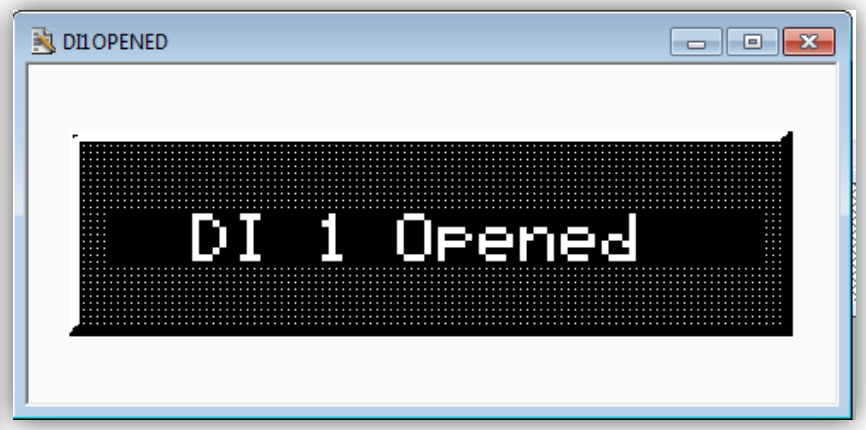


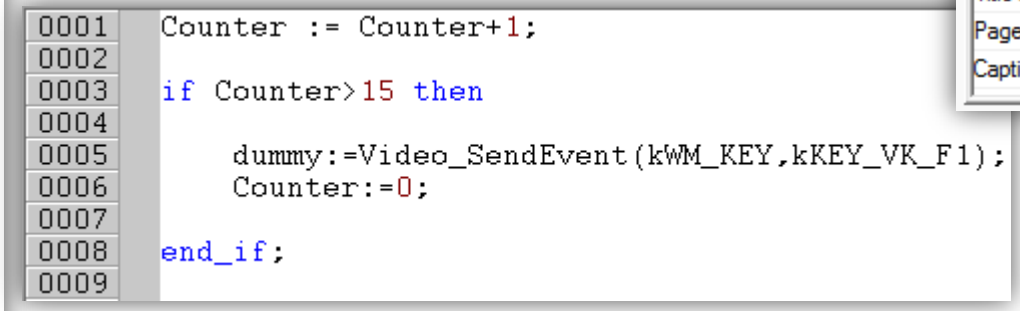




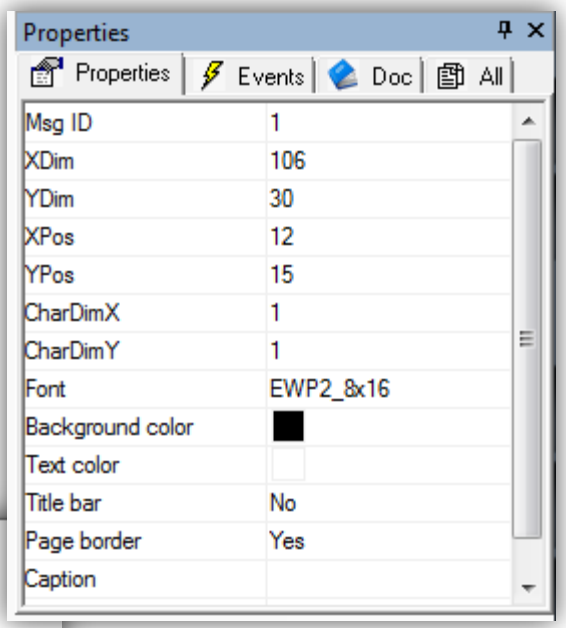
Messages...





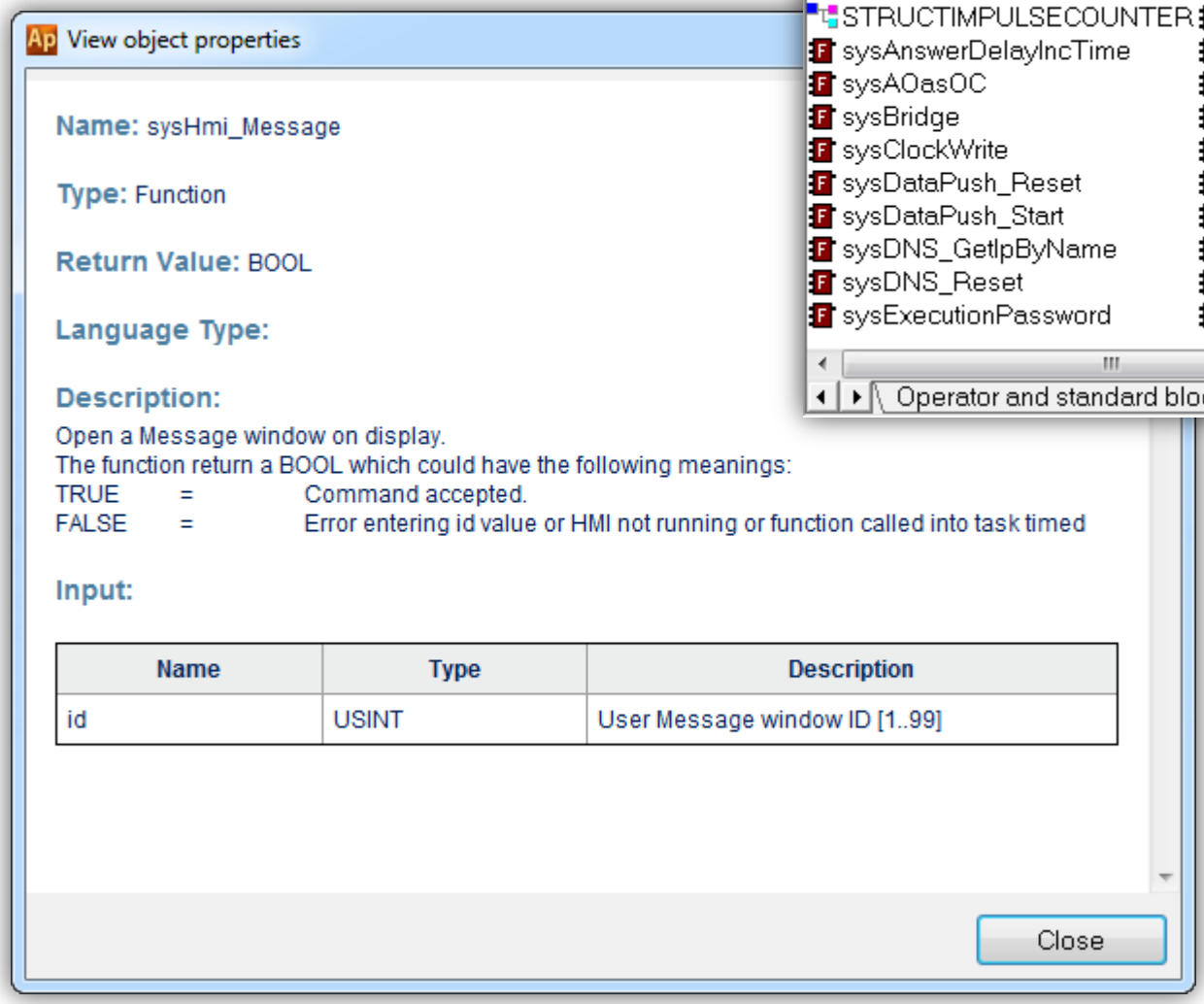


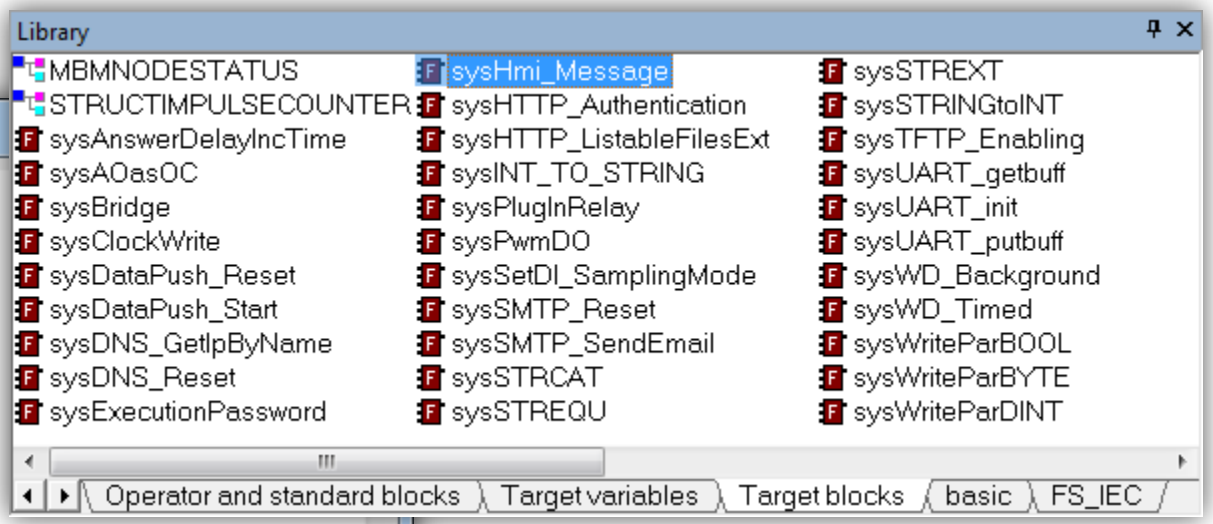
```
0001  Counter := Counter+1;
0002
0003  if Counter>15 then
0004
0005      dummy:=Video_SendEvent(kWM_KEY,kKEY_VK_F1);
0006      Counter:=0;
0007
0008  end_if;
0009
```



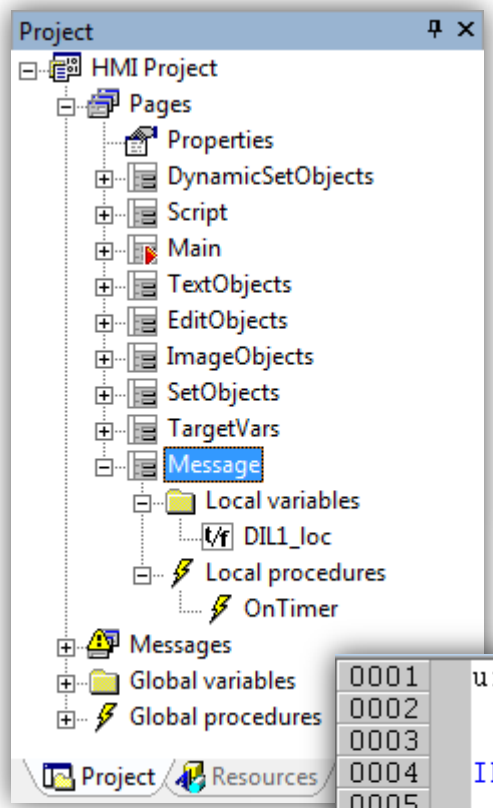
- **Message can be opened from every page but it can‘t have child page**
- **Message is identified by its Msg ID**
- **Message can be opened using Video_SendEvent(kWM_MSG, MsgID)**
- **Msg ID 101-102 are automatically opened with low/up range error**

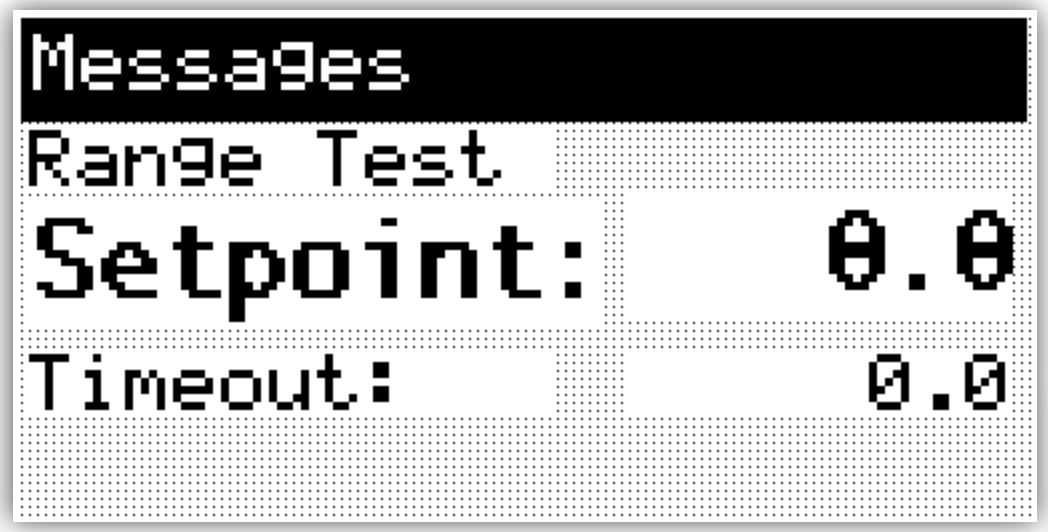
... Message


View object properties
Name: sysHmi_Message
Type: Function
Return Value: BOOL
Language Type:
Description:
Open a Message window on display.
The function return a BOOL which could have the following meanings:
TRUE = Command accepted.
FALSE = Error entering id value or HMI not running or function called into task timed
Input:
Name
Type
Description
id
USINT
User Message window ID [1..99]
Close


Library
MBMNODESTATUS
STRUCTIMPULSECOUNTER
sysAnswerDelayIncTime
sysAOasOC
sysBridge
sysClockWrite
sysDataPush_Reset
sysDataPush_Start
sysDNS_GetIpByName
sysDNS_Reset
sysExecutionPassword
sysHmi_Message
sysHTTP_Authentication
sysHTTP_ListableFilesExt
sysINT_TO_STRING
sysPlugInRelay
sysPwmDO
sysSetDI_SamplingMode
sysSMTP_Reset
sysSMTP_SendEmail
sysSTRCAT
sysSTREQU
sysSTREXT
sysSTRINGtoINT
sysTFTP_Enabling
sysUART_getbuff
sysUART_init
sysUART_putbuff
sysWD_Background
sysWD_Timed
sysWriteParBOOL
sysWriteParBYTE
sysWriteParDINT
Operator and standard blocks
Target variables
Target blocks
basic
FS_IEC

Testing Messages



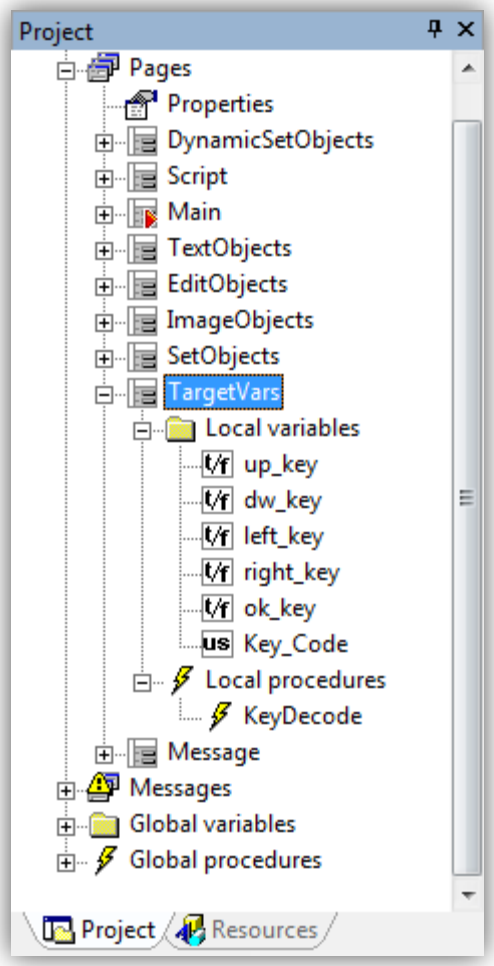


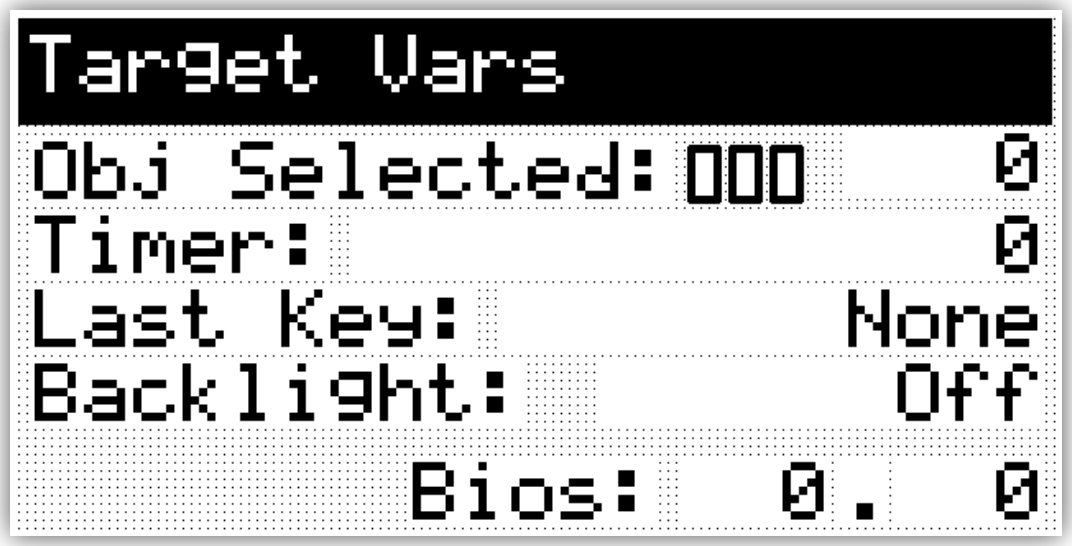
Access	RW
Selection order	1
Variable	@PLC.SetPoint
Data type	INT
Low limit	150
High limit	300
Refresh	FALSE
Visible	TRUE
Selectable	TRUE

Access	RW
Selection order	2
Variable	TimeOutCtd
Data type	UDINT
Low limit	*
High limit	*
Refresh	TRUE
Visible	TRUE
Selectable	FALSE

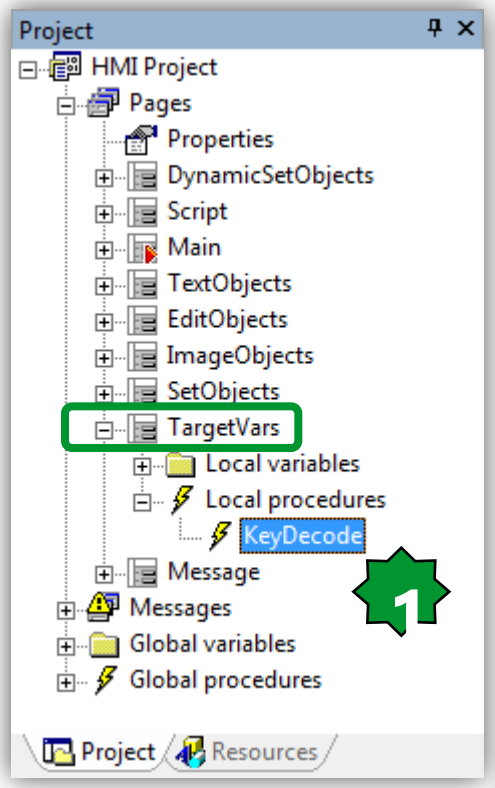
```
0001  uint_ret:= Video_GetParam( 0, 8192, 0, ?tmpBOOL, tyBool );
0002
0003
0004  IF NOT(tmpBOOL) AND DIL1_loc THEN
0005      (* Open Message window with msgid=1 *)
0006      uint_ret:=Video_SendEvent( kWM_MSG, 1 );
0007  END_IF;
0008
0009  DIL1_loc := tmpBOOL;
0010
0011
```

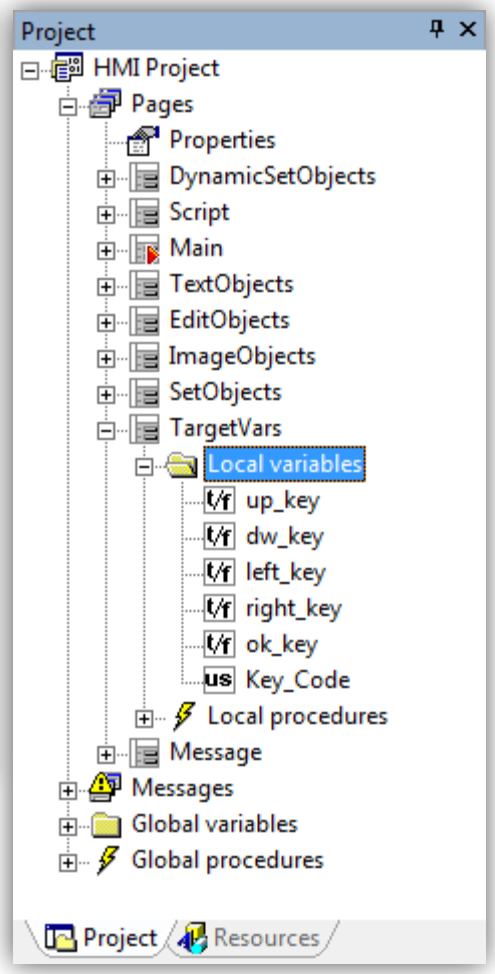
Target Vars


Project
Pages
Properties
DynamicSetObjects
Script
Main
TextObjects
EditObjects
ImageObjects
SetObjects
TargetVars
Local variables
up_key
dw_key
left_key
right_key
ok_key
Key_Code
Local procedures
KeyDecode
Message
Messages
Global variables
Global procedures
Project
Resources


Target Vars
Obj Selected: 000 0
Timer: 0
Last Key: None
Backlight: Off
Bios: 0. 0

Target Vars: sysKeyPressed

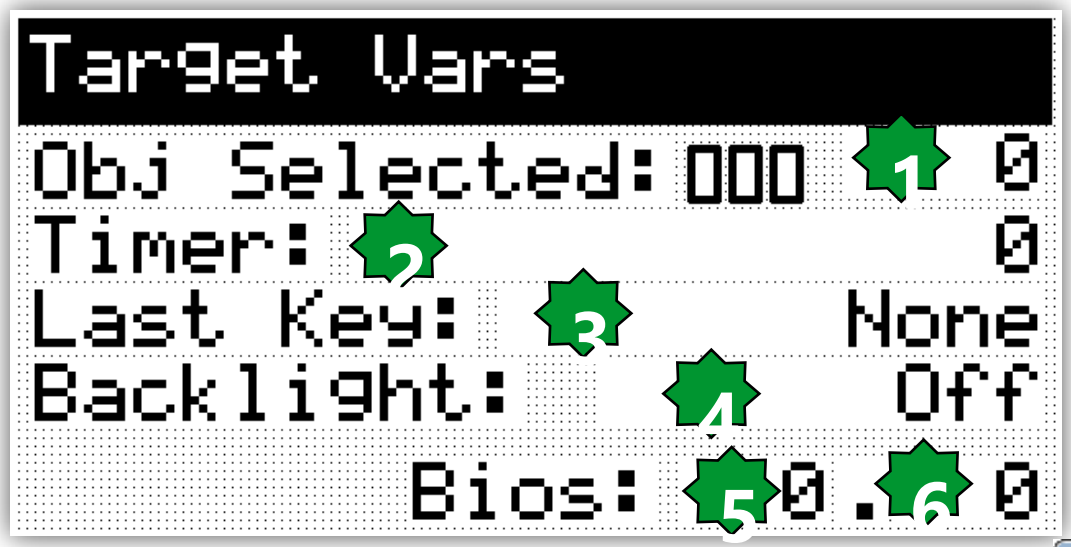





```
(*
    sysKeyPressed

    KEY_NULL            = 0x00000000
    KEY_UP              = 0x00100001
    KEY_OK              = 0x00200001
    KEY_RIGHT           = 0x00800001
    KEY_DOWN            = 0x01000001
    KEY_LEFT            = 0x02000001
    KEY_UP_LONG         = 0x80100001
    KEY_OK_LONG         = 0x80200001
    KEY_RIGHT_LONG      = 0x80800001
    KEY_DOWN_LONG       = 0x81000001
    KEY_LEFT_LONG       = 0x82000001
*)


up_key      := (sysKeyPressed AND 16#00100000) <>0;
ok_key      := (sysKeyPressed AND 16#00200000) <>0;
right_key   := (sysKeyPressed AND 16#00800000) <>0;
dw_key      := (sysKeyPressed AND 16#01000000) <>0;
left_key    := (sysKeyPressed AND 16#02000000) <>0;

IF (sysKeyPressed AND 16#80000000) <>0 THEN

    Key_Code := 6*TO_USINT(up_key)+
                7*TO_USINT(ok_key)+
                8*TO_USINT(right_key)+
                9*TO_USINT(dw_key)+
                10*TO_USINT(left_key);

ELSE

    Key_Code := TO_USINT(up_key)+
                2*TO_USINT(ok_key)+
                3*TO_USINT(right_key)+
                4*TO_USINT(dw_key)+
                5*TO_USINT(left_key);

END_IF;
```

2

Target Vars



1. sysCurrentSelectedPosition changed value based on the current selected object
2. sysTimer works as in Ap
3. Key_Code (see previous slide)
4. sysBackLight
5. sysMSK is the firmware mask
6. sysVER is the firmware version

Enumeratives - [BacklightEnum]

Value	Description
0	Off
1	On
2	Blink
3	Timed
4	Timed Run

Enumeratives - [KeyCodeEnum]

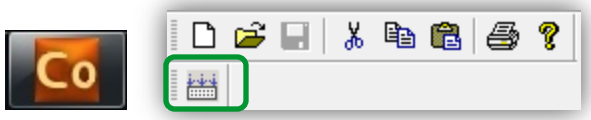
Value	Description
0	None
1	Up
2	Ok
3	Right
4	Down
5	Left
6	L-Up
7	L-Ok
8	L-Right
9	L-Down
10	L-Left

Remember to align Ap UI and Co projects

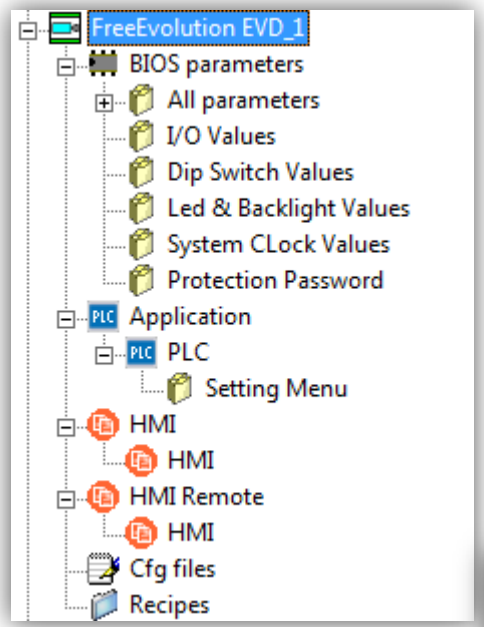
- “Recompile all”the Application Project

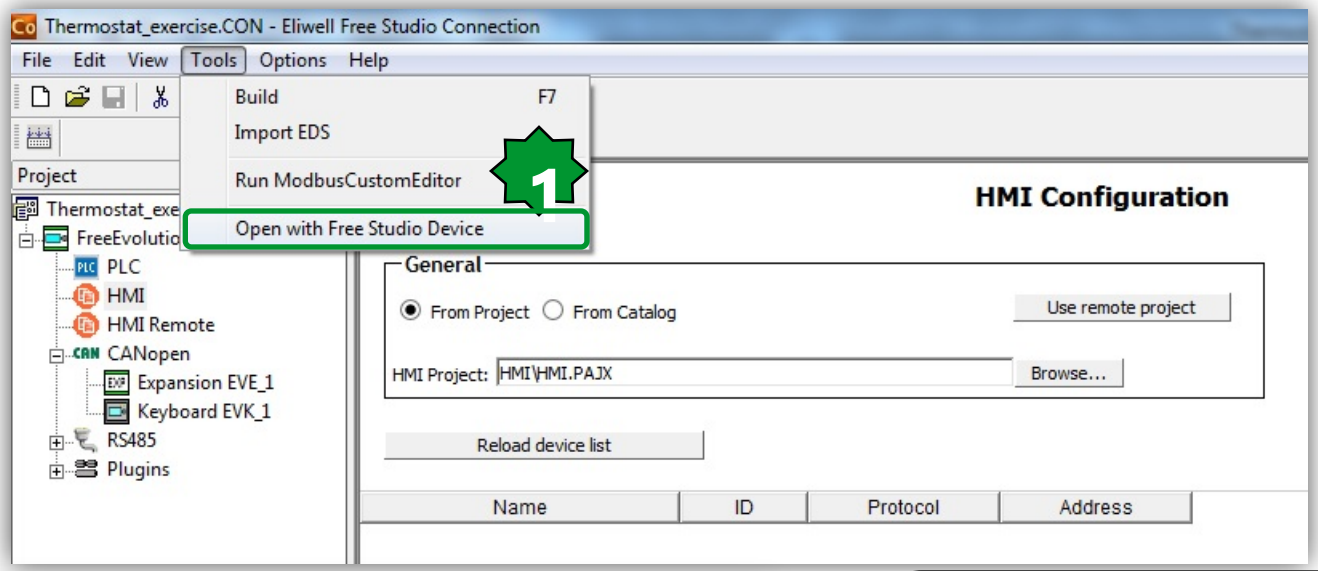
- Compile the User Interface project

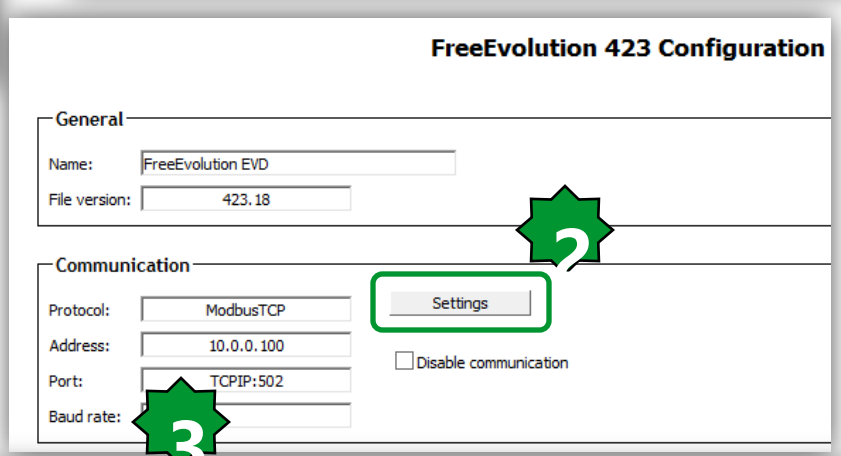
- Build the Connection project in order to align it to the linked Ap and UI project



Open Free Studio Device







1.Tools►Open with free Studio Device
2. Define settings
3. Connect to the EVD
4. Download all

Upload HMI from EVD to EVK...

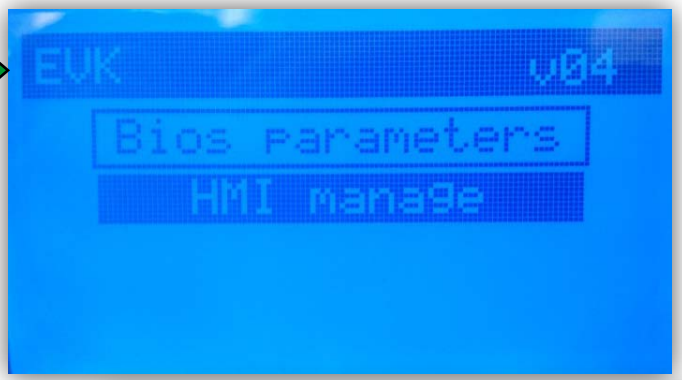
In the BIOS Menu of EVK (Long press of down + left if a HMI is already loaded).

In order to run HMI:

1. Select HMI Manage
2. upload
3. Press OK to confirm
4. Press to run HMI

Upload HMI from EVD to EVK...

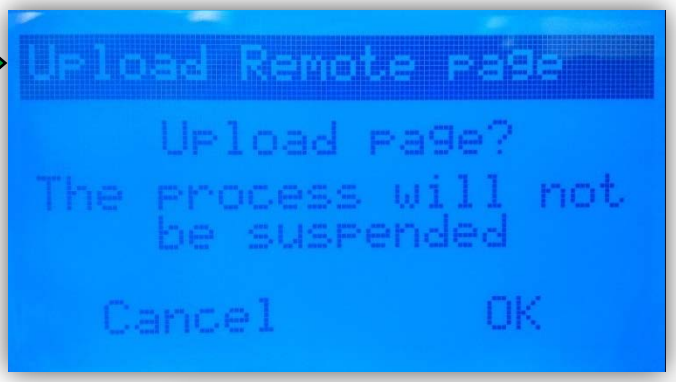
1



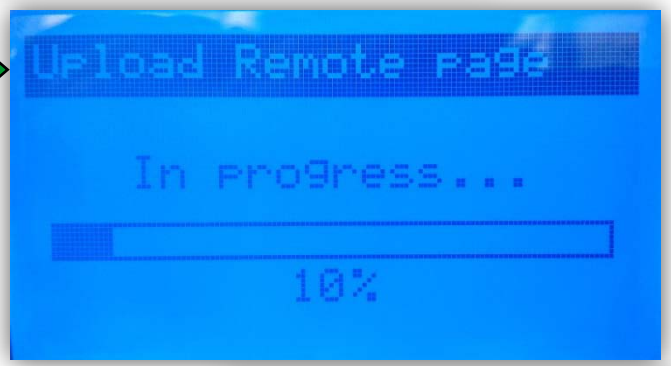
2

3

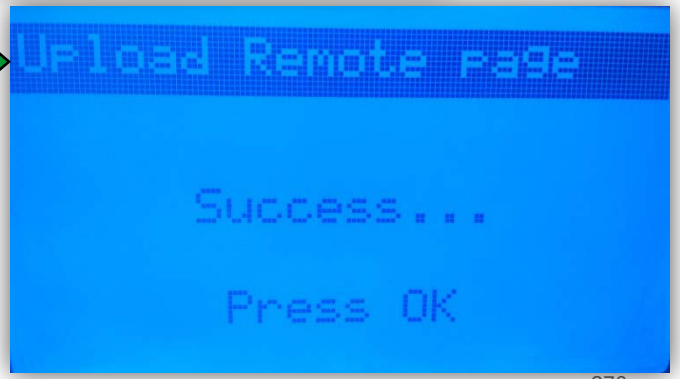
4



5



6



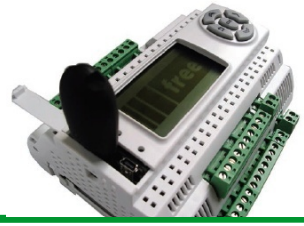
Chapter 18

Evolution USB

Goal:

- DownLoad/UpLoad via USB
- Retain variable configuration

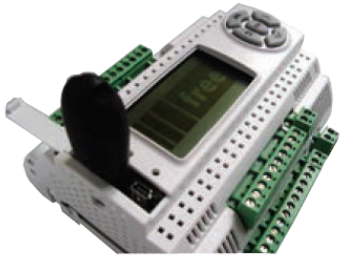
USB device (PC ◄►target)



- **Type A USB (HOST)**. Used to connect a standard USB to download the application/BIOS.
- **Type B mini USB (DEVICE)**. Used to connect FREE Evolution to a PC or third party device via mini A/B USB cable to up/download the application, files or documentation. This can be done from a PC or other device. [1]

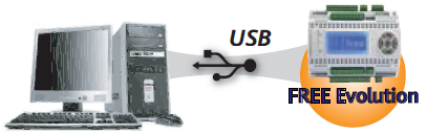
USB Formated FAT32

CASE 1
USB Host
USB → ← FREE



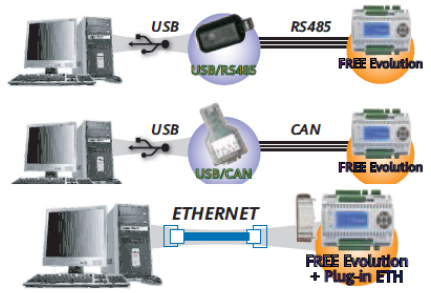
Data downloading direction	→	←
Parameter map	✓	✓
IEC application	✓	-
HMI application	✓	-
Data file	✓	✓
BIOS	✓	-

CASE 2
USB device
PC → ← FREE



Data downloading direction	→	←
Parameter map	-	-
IEC application	✓	✓
HMI application	✓	✓
Data file	✓	✓
BIOS	-	-

CASE 3
USB-RS485 /USB-CANOpen
ETHERNET + Plugin
PC → ← FREE



Data downloading direction	→	←
Parameter map	✓	✓
IEC application	✓	-
HMI application	✓	-
Data file	✓	✓
BIOS	✓	-

What is inside Evolution Filesystem or USB ?

Inside both Evolution/USB Pen Drive:

PLCIEC.COD : Application binary file
HMIIEC.COD : User Interface binary file (not mandatory)
HMIREM.KBD : Remote User Interface binary file (not mandatory)
CONNEC.PAR : Master Connectivity settings (not mandatory)

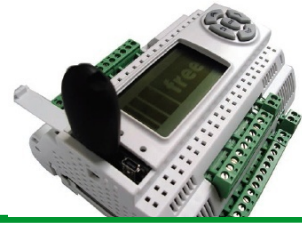
Inside Evolution:

Webserver files
Logging file
Others…

Inside USB Pen Drive:

PARAM.DAT (.RAW) : Parameter Map file

Parameter map file



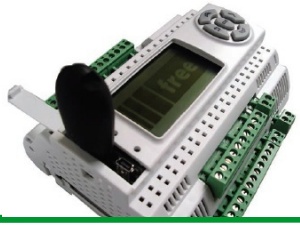
PARAM.DAT file includes a set of Evolution BIOS& IEC parameter values.

PARAM.DAT can be renamed as PARAM.raw in order to skip parameters' range limit check (used in case of par limited by other pars).

PARAM.DAT file can be created via IEC code using the target var (see next slides)

PARAM.DAT (.RAW) can be manually created/modified in order to contain even a subset of the full map.

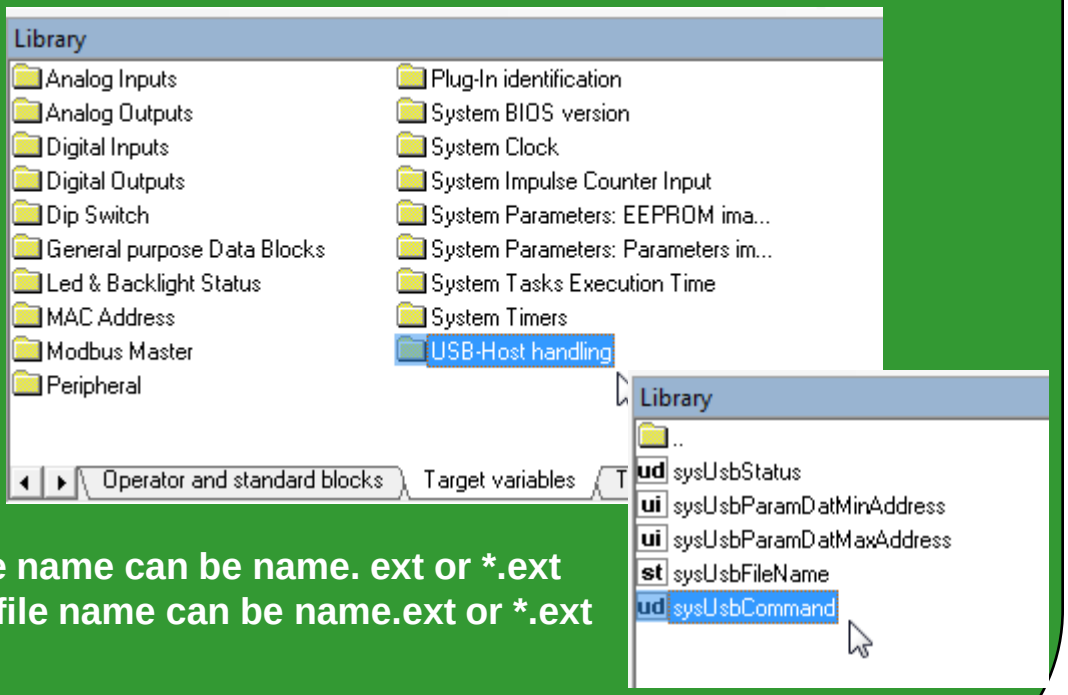
Use USB host from IEC code



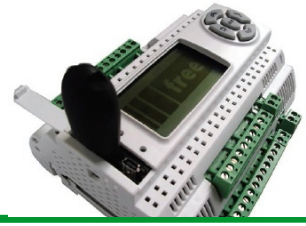
- **Upload an application from the pen drive to Evolution**
- **Upload/Download a parameter map from evolution to the pen drive.**

sysUSBCommand is the system command to upload/download to/from USB-Host:

7 = load PARAM.BIN from USBH
8 = load PLCIEC.COD from USBH
9 = load HMIIEC.COD from USBH
10 = load PARAM.DAT from USBH
11 = save PARAM.DAT to USBH
12 = load CONNEC.PAR from USBH
13 = load HMIREM.KBD from USBH
14 = save sysUsbFileName file to USBH, file name can be name. ext or *.ext
15 = load sysUsbFileName file from USBH, file name can be name.ext or *.ext



Automatic Upload via USB pen drive



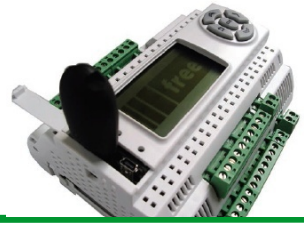
Uploading automatically an application via USB pen drive
- **Copy into a pen drive the COD/PAR/DAT files**
- **Edit an UPLOAD.TXT file containing the list of the files to be uploaded**

Note. PARAM.DAT (.RAW) file can be uploaded only if FREE Evolution has been rebooted with related application, therefore PARAM.DAT (.RAW) cannot be uploaded at the same time of PLCIEC.COD

The upload file can have a prefix from 00 to 15, for example 03UPLOAD.TXT:
- **Copy into a pen drive the UPLOAD.TXT (03UPLOAD.TXT) files as well**

Files with numeric prefix are uploaded only if the Evolution dip-switches match the prefix; in this way it is possible to store on the same USB pen drive one or more Evolution applications.

USB-LED status during upload



The upload process starts when the pen drive is plugged and can be monitored through the led status which, during the upload process, are controlled directly by Evolution bios.

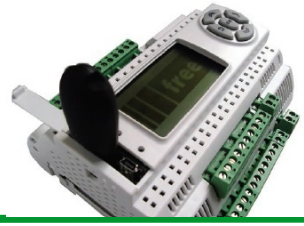
The process results which will switch on the red led are the ones related to a value of sysUsbStatus>1.

After the process, Evolution must be restarted in order to run the new application. File PARAM.DAT is uploaded by an Evolution only if the Bios Mask and Par_POLI7 of the Evolution that has generated the PARAM.DAT are the same as the destination Evolution.

The parameters' map update does not require to switch off Evolution.

LED		Upload
RED	Blinking 2 seconds	Failed
YELLOW	On	Underway
GREEN	Blinking	Completed successfully

Firmware Update By USB



How To Update:

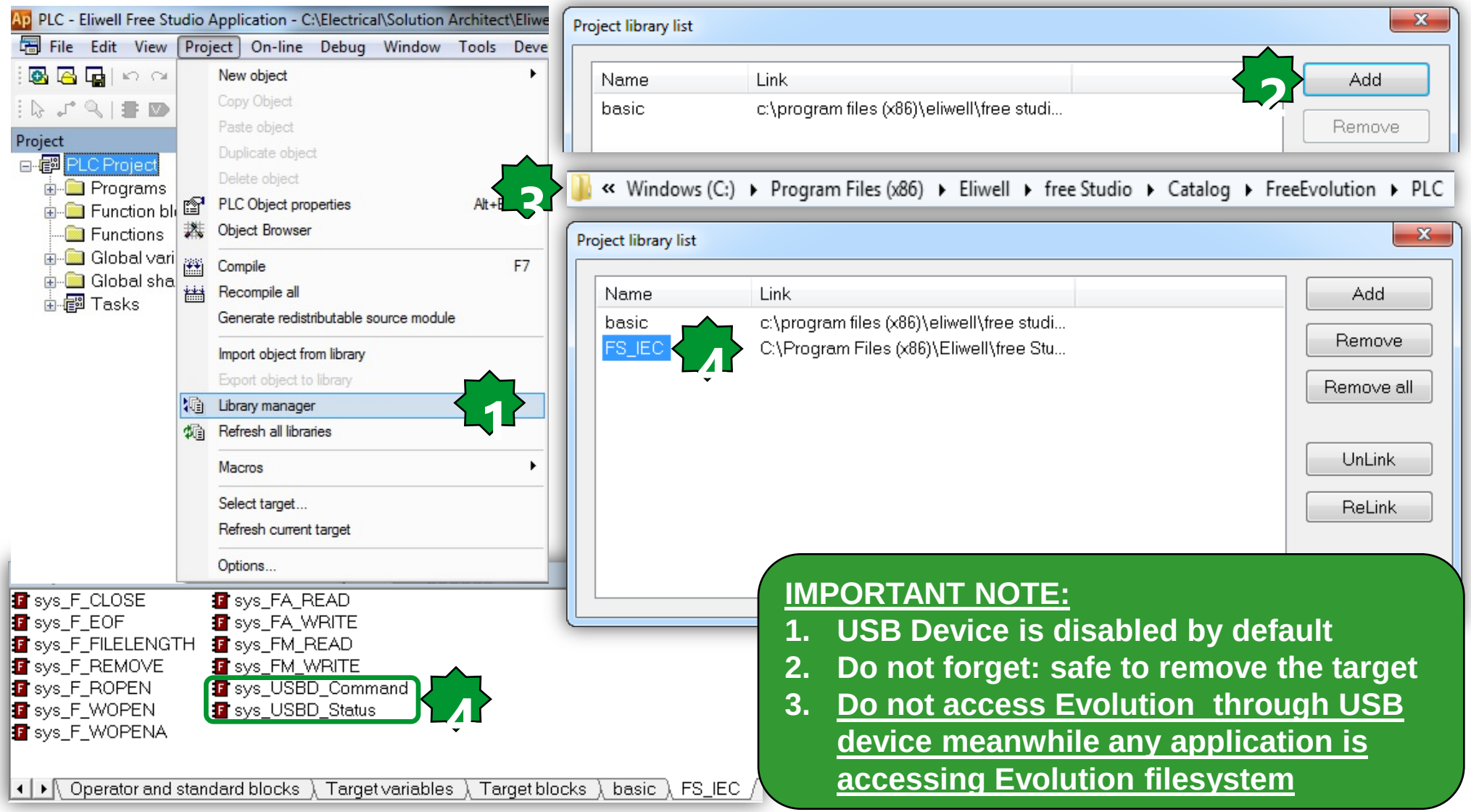
- **Copy the relevant .bin file into a USB pen drive (e.g. msk423_18.bin)**
- **Connect USB pen drive to Evolution**
- **Firmware will be downloaded into Evolution**
- **Yellow LED will blink during download.**
- **Remove USB pen drive as soon as Yellow LED will switch off**
- **Evolution will automatically reset and will reboot**

BIOS are available @<C:\Programs>\Eliwell\free Studio\Catalog\FreeEvolution\<firmware>
<firmware> = msk423 for EVD, msk477 for EVC.

Please Note: a SYSTEM FAULT message will appear - DO NOT CONSIDER -
BIOS upgrade has been completed successfully

Note: Evolution make a filter based on the filename in order to prevent user mistakes

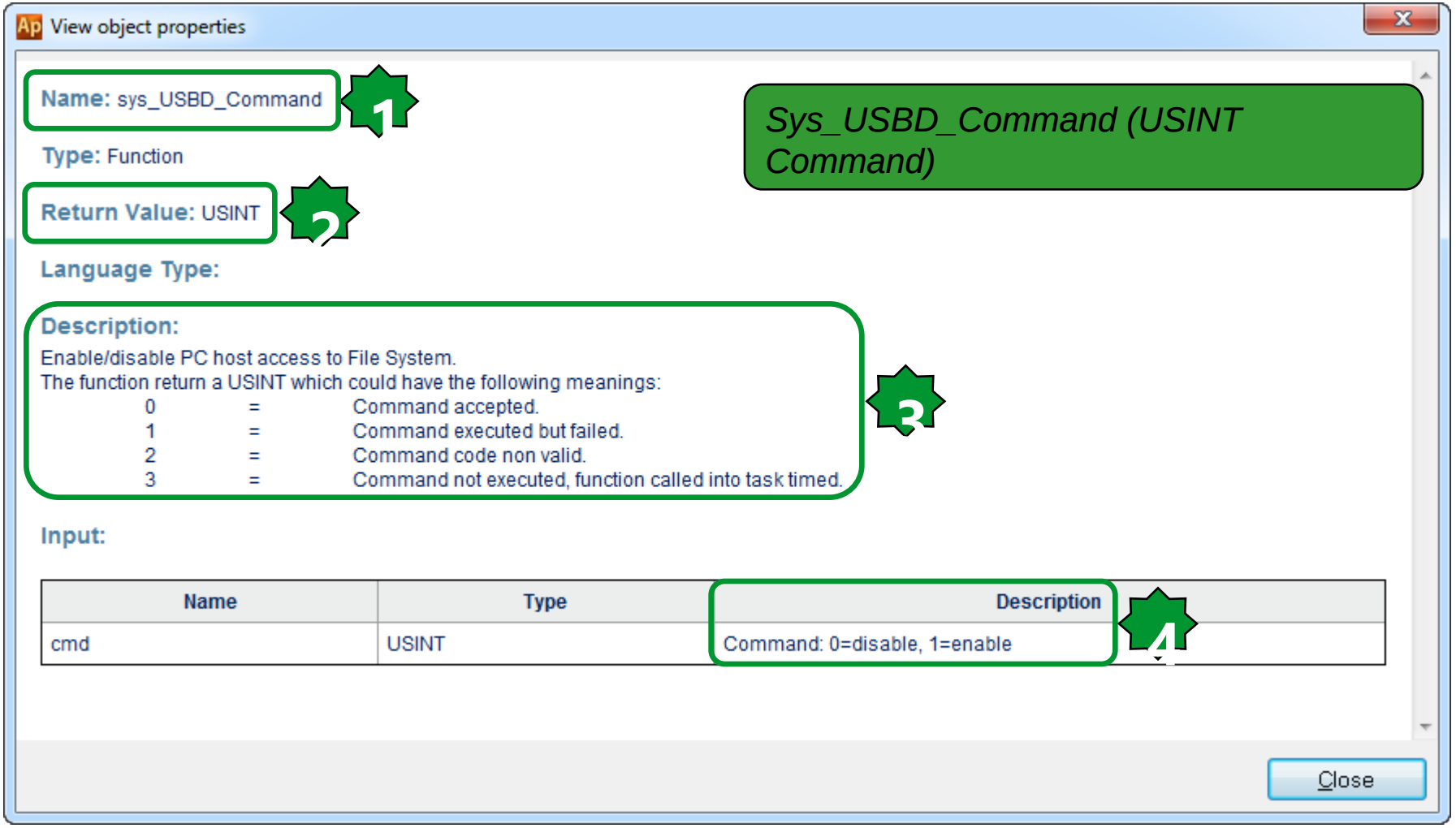
Using the USB Device - Adding library



IMPORTANT NOTE:

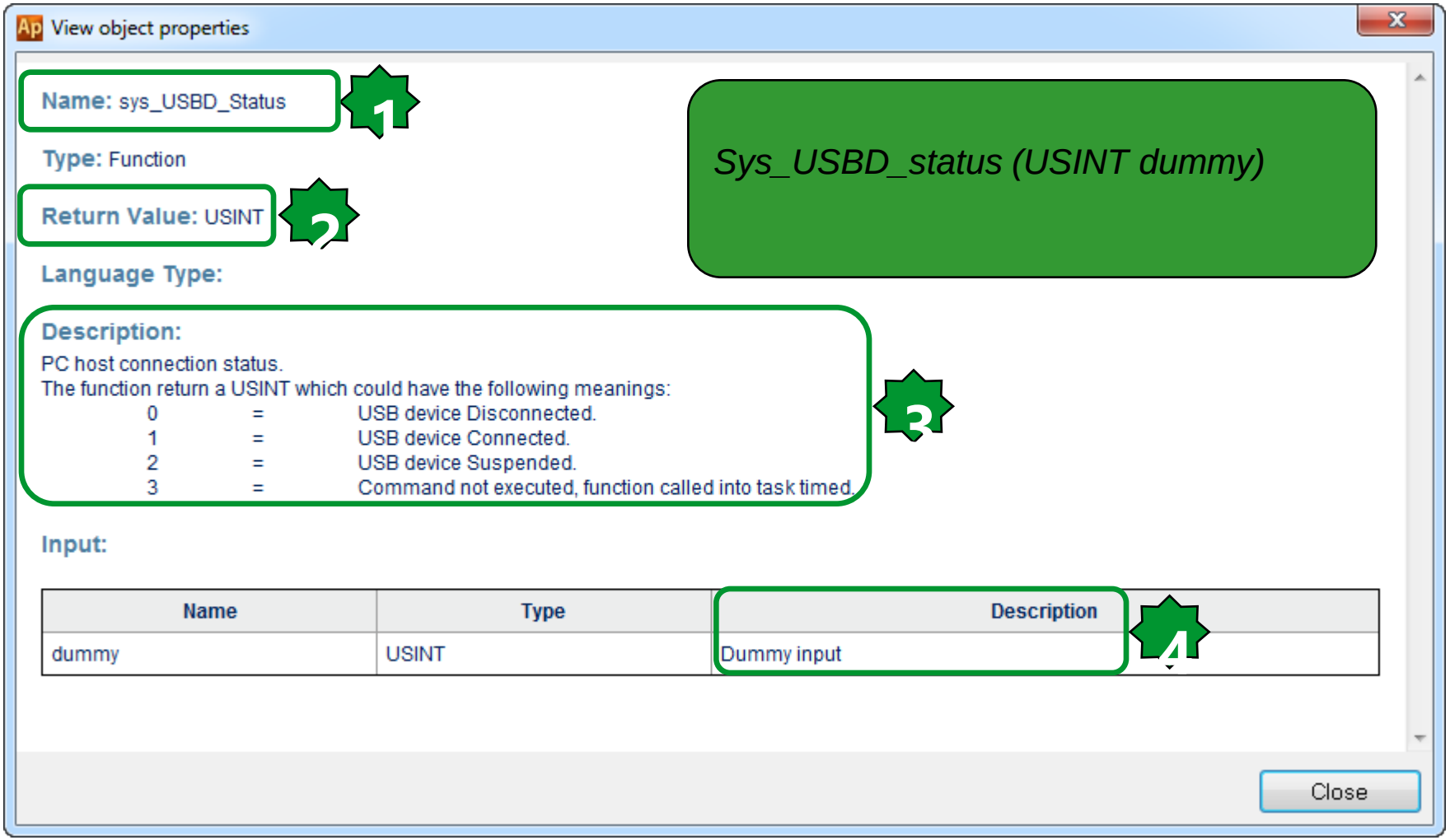
1. **USB Device is disabled by default**
2. **Do not forget: safe to remove the target**
3. **Do not access Evolution through USB device meanwhile any application is accessing Evolution filesystem**

Enable/Disable PC host access to file System Function



Sys_USBD_Command (USINT Command)

PC host connection status function

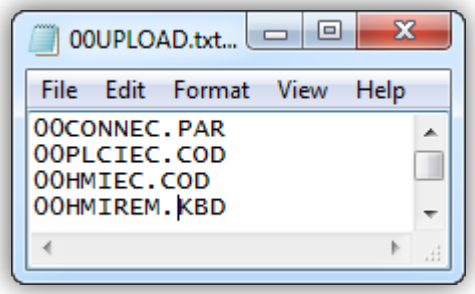


USB application download workflow

Cfg_Node name.par @ project root

Rename it as CONNEC:PAR

Create Text file as shown



Copy them to the USB stick

Project name(PLC).bin @project root ► PLC ► Download

Rename it as PLCIEC.COD

Power cycle to apply them into the RAM

Project name(HMI).bin @project root ► HMI ► Download

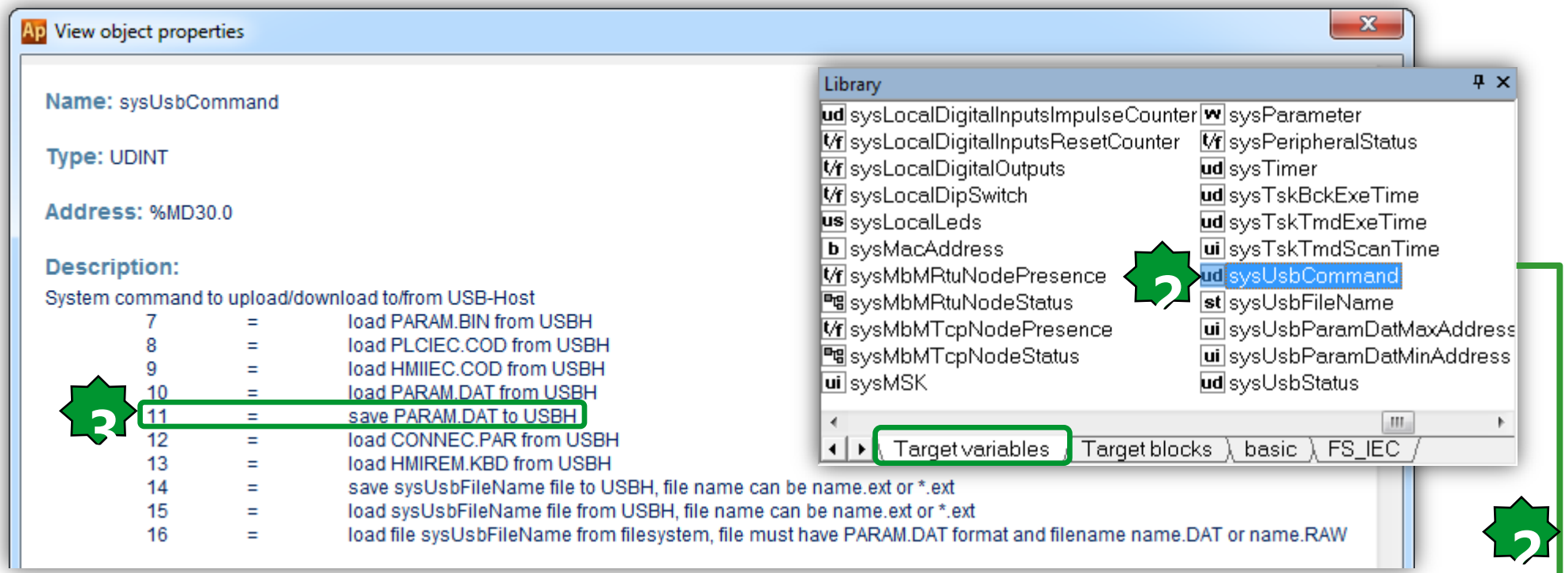
Rename it as HMIIEC.COD

Project name(HMI).bin @project root ► HMI ► Download

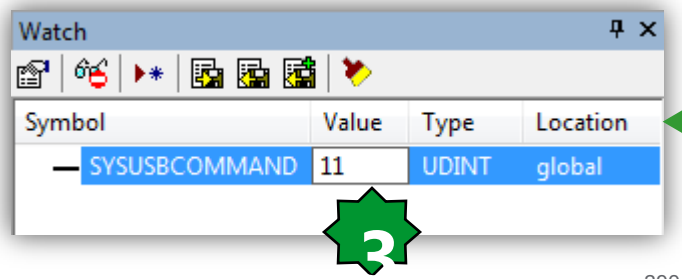
Rename it as HMIREM.KBD

Note.
Always use Capital letter in renaming TXT files

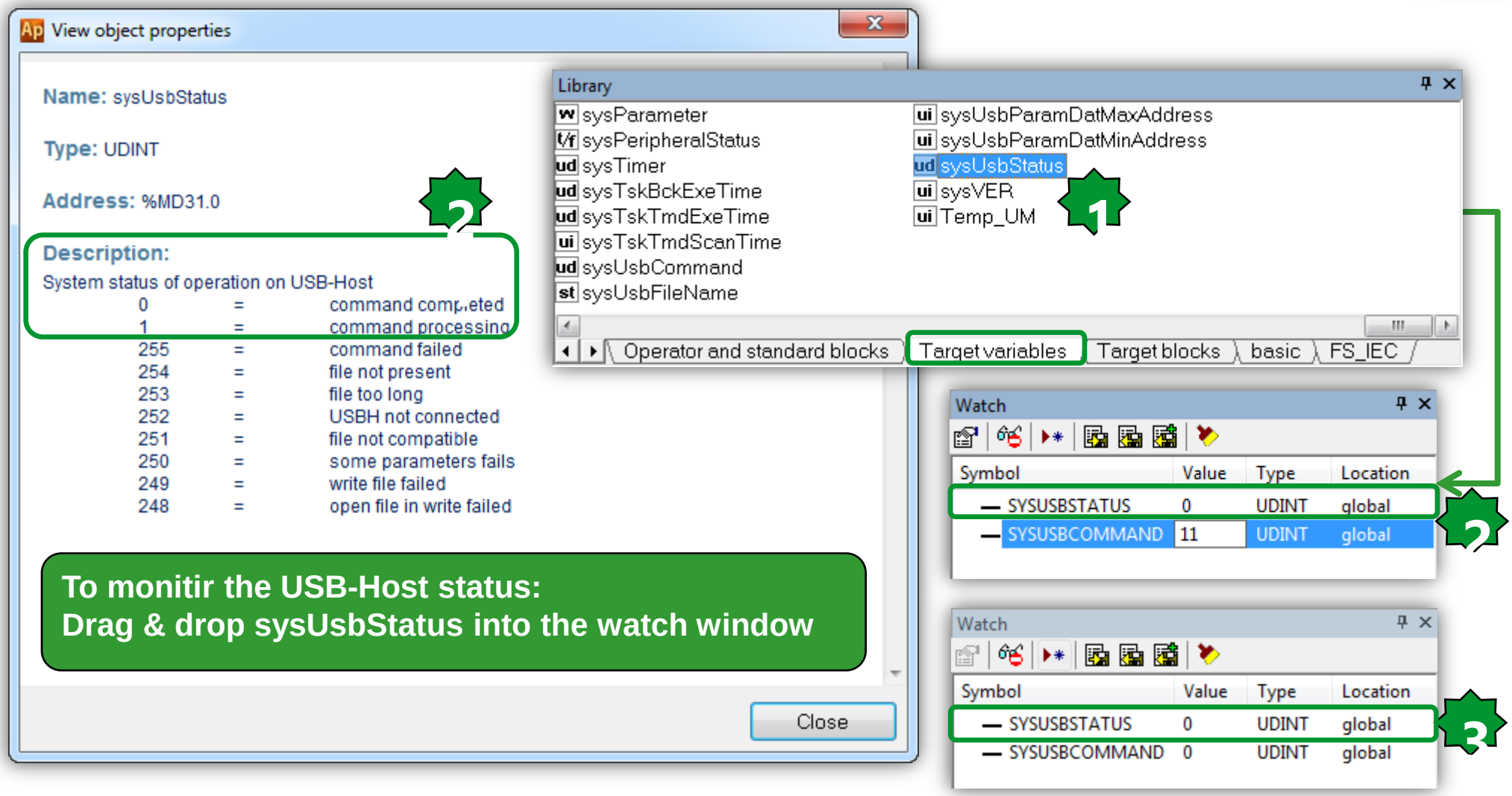
USB data upload workflow...



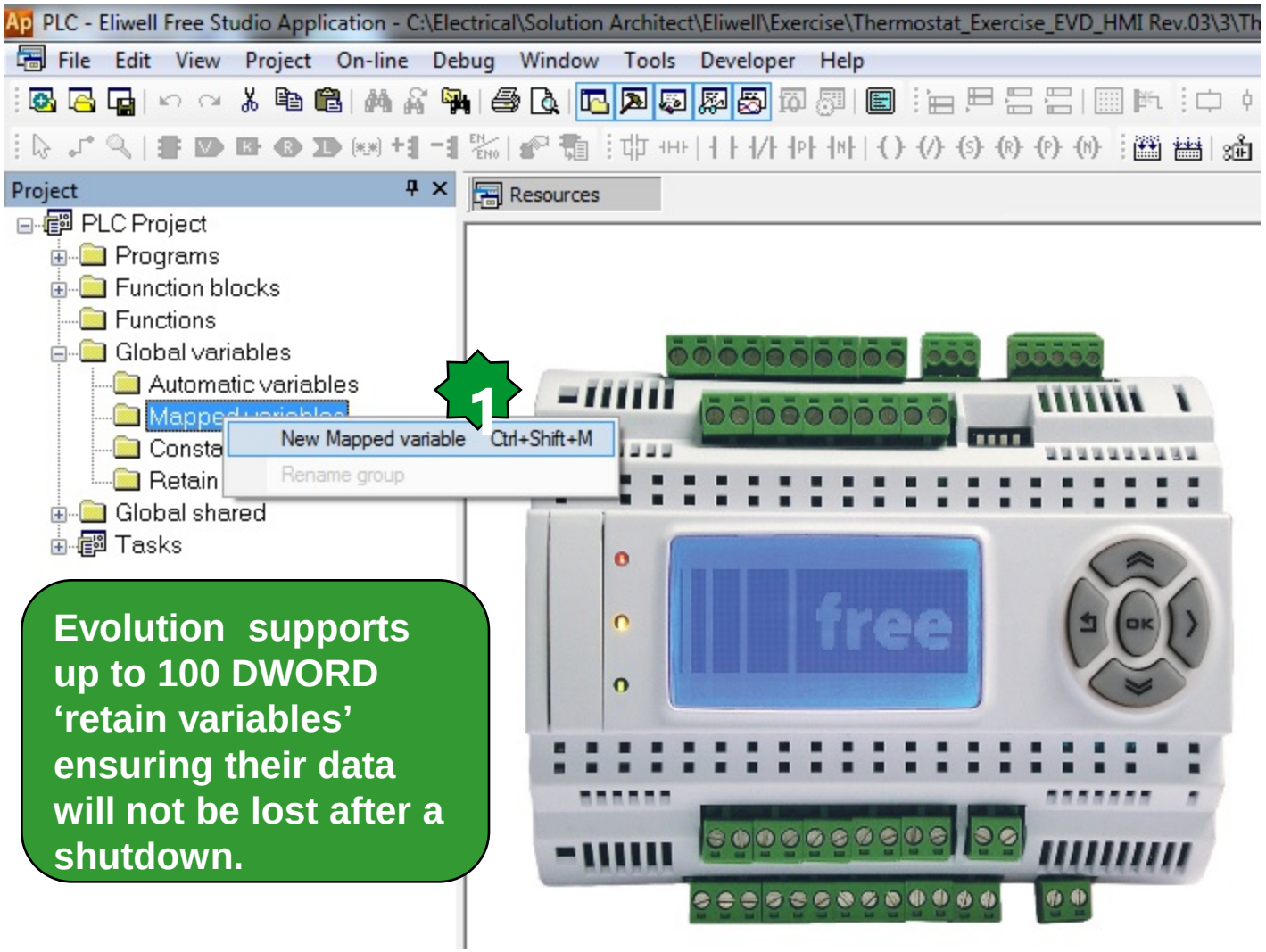
1.Connect to the target via Ap
2. Drag & drop sysUsbCommand into the watch window
3. Write value=11



...USB data upload workflow



Evolution - Retain Variables



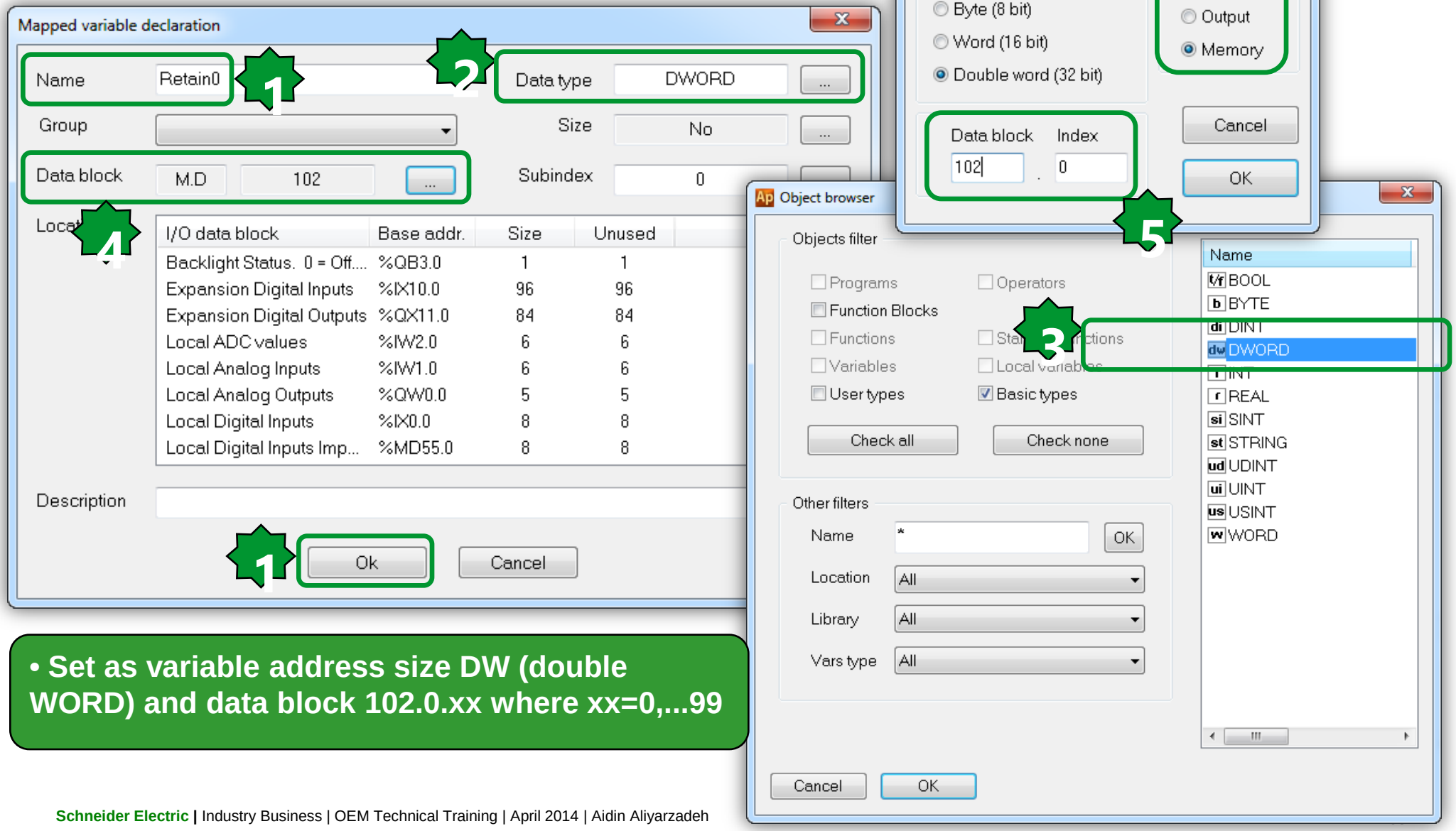
A RETAIN variable indicates that the variables within the structuring elements are retentive, i.e. they keep their value even after the target device has been reset or switched off.

Evolution supports up to 100 DWORD 'retain variables' ensuring their data will not be lost after a shutdown.

Retain variable values can be changed several times without affecting internal memory performance.

Note: RETAIN variables cannot be displayed in the Watch window

Evolution - Retain Variables



- **Set as variable address size DW (double WORD) and data block 102.0.xx where xx=0,...99**

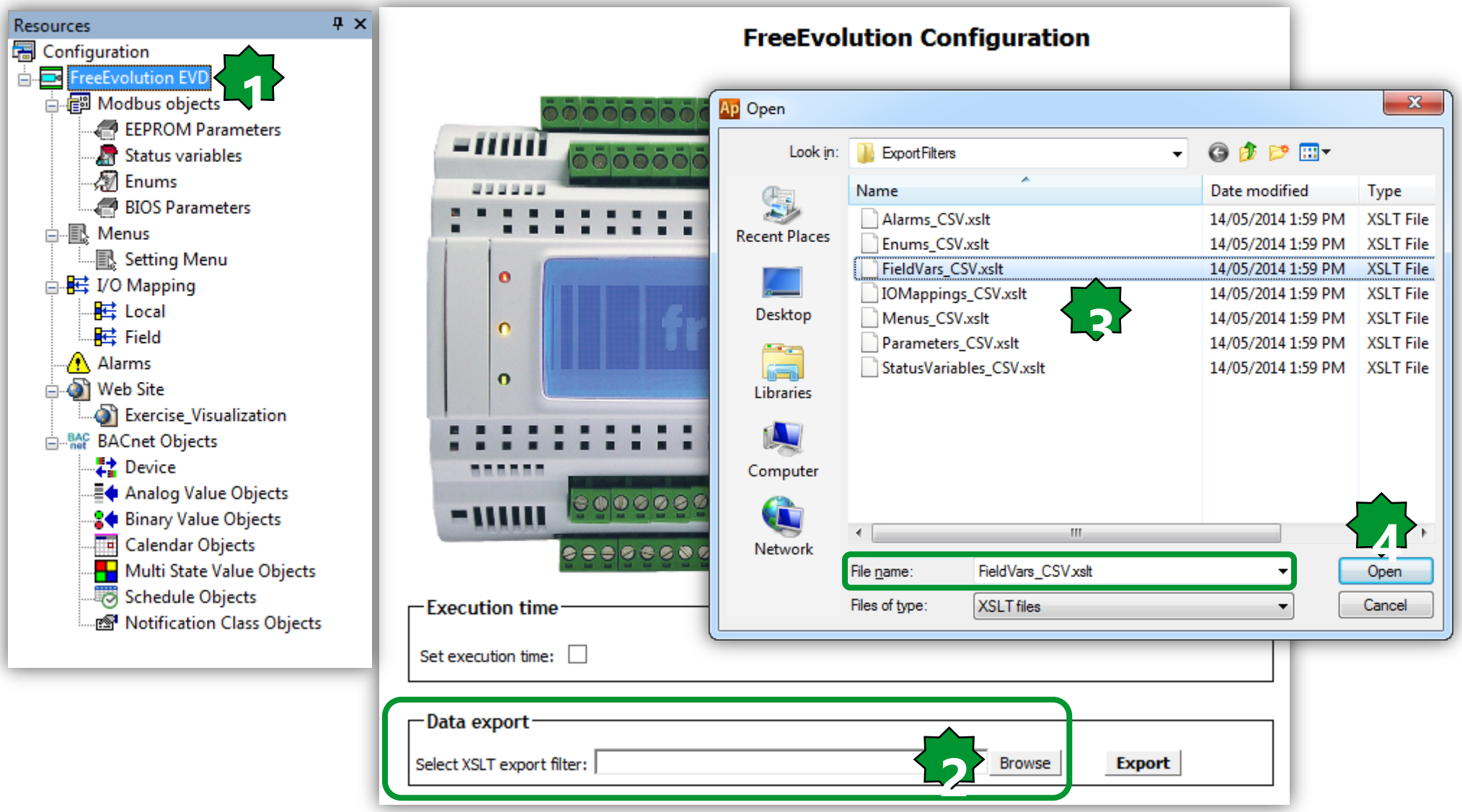
Chapter 19

Documentation

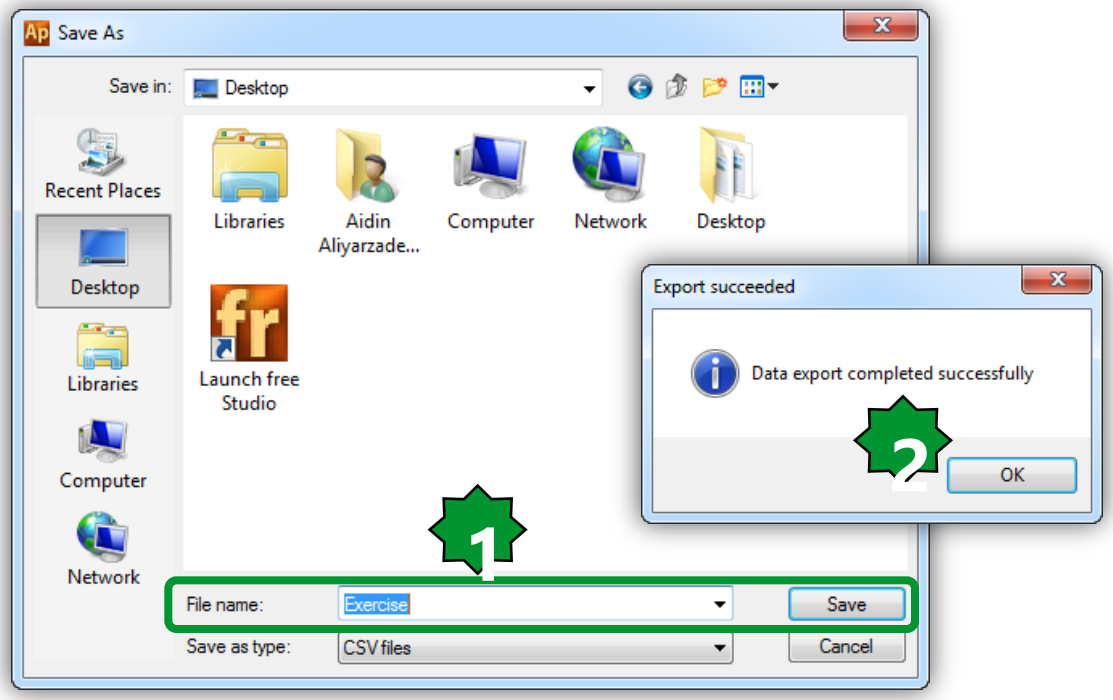
Goal:

Creating document and exporting by Application, Device & User Interface as report or as useing them by other products such as Vijeodesigner

Application - Export to Excel...



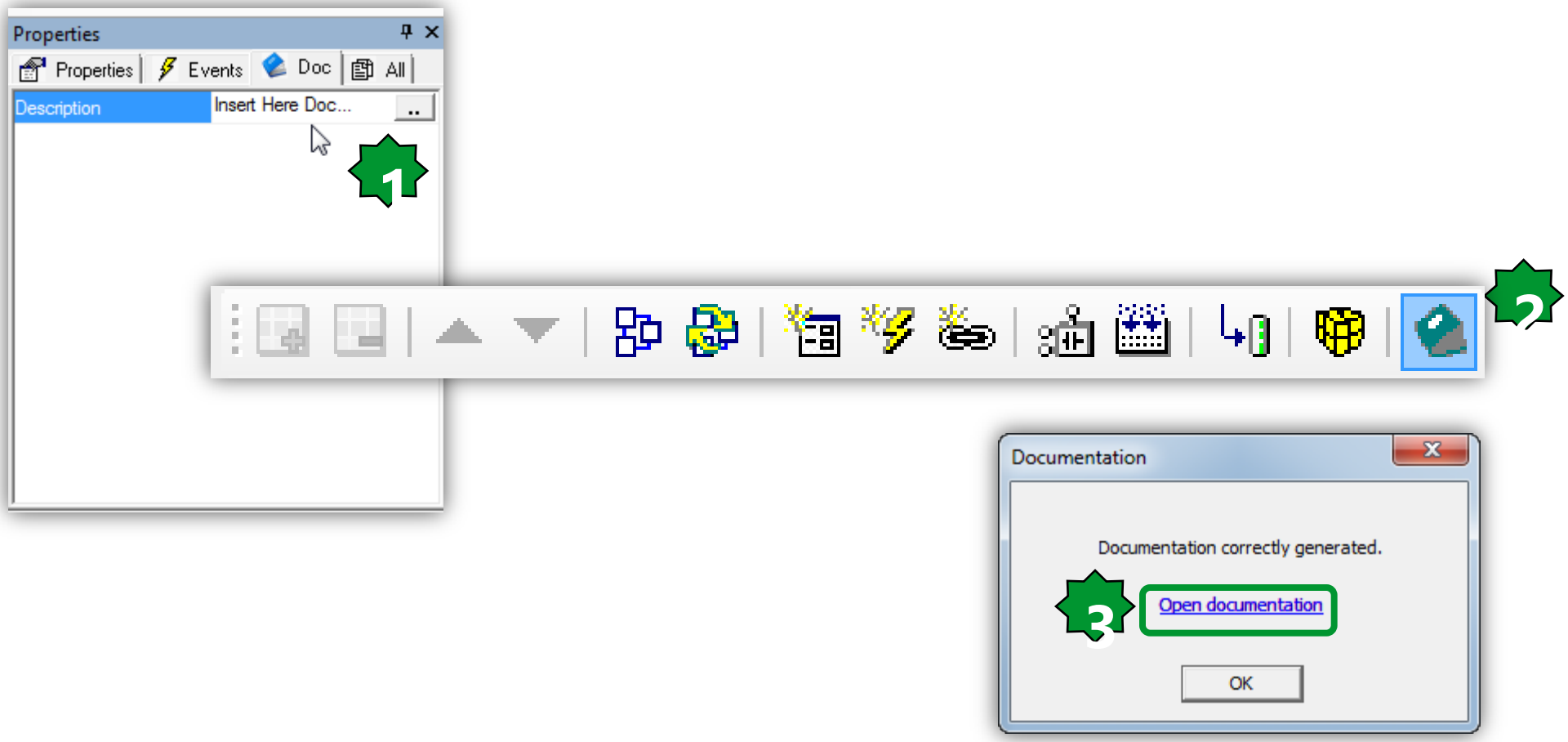
... Export to Excel



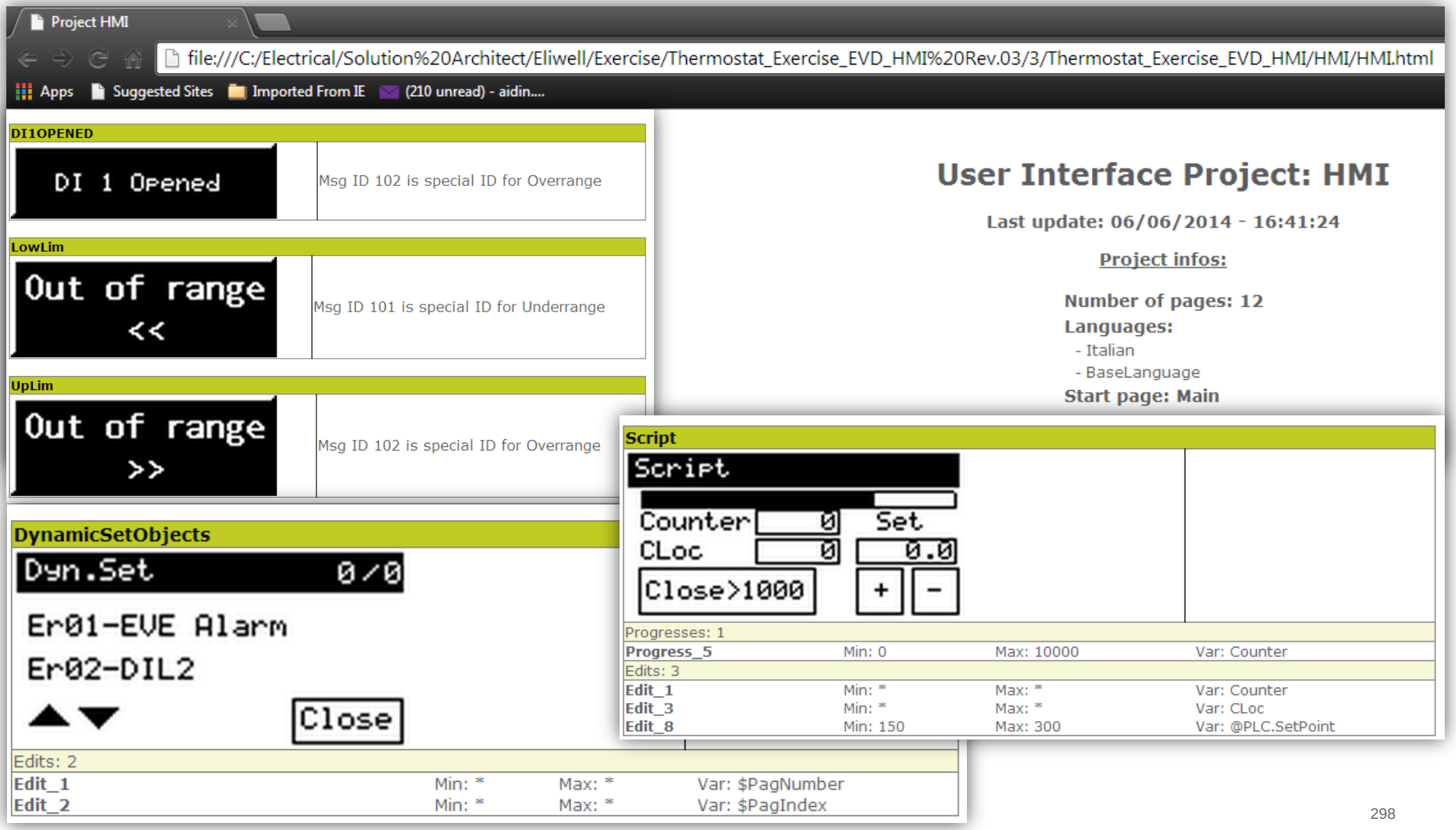
Name	Type	In/Out	Description
AI1_E	INT	in	
DI1_E	BOOL	in	
DI2_E	BOOL	in	
DO1_E	BOOL	out	
DO2_E	BOOL	out	

3

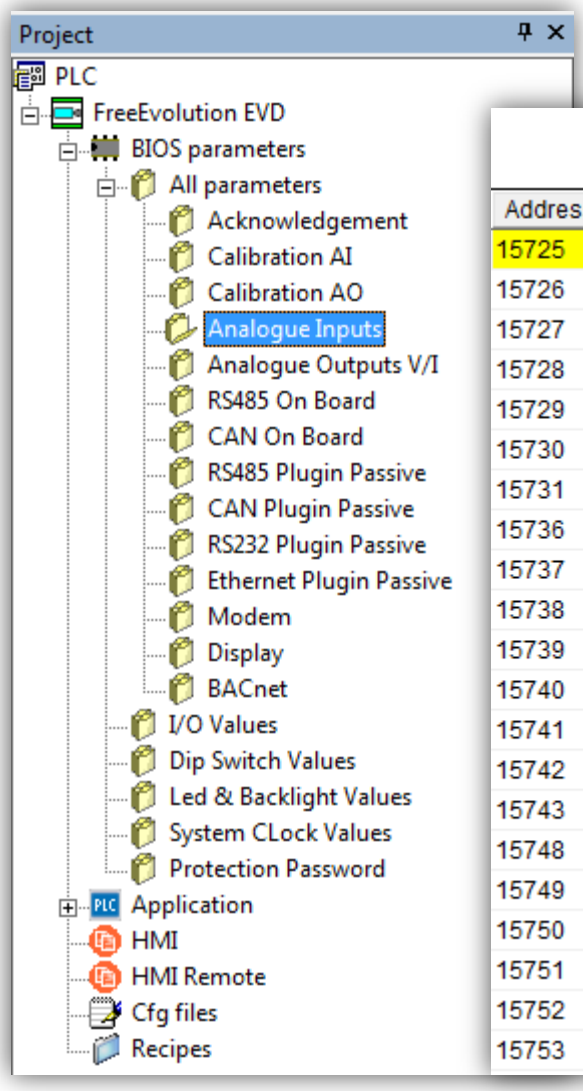
User Interface - Documentation



User Interface - Documentation



Device - Export to Text file...

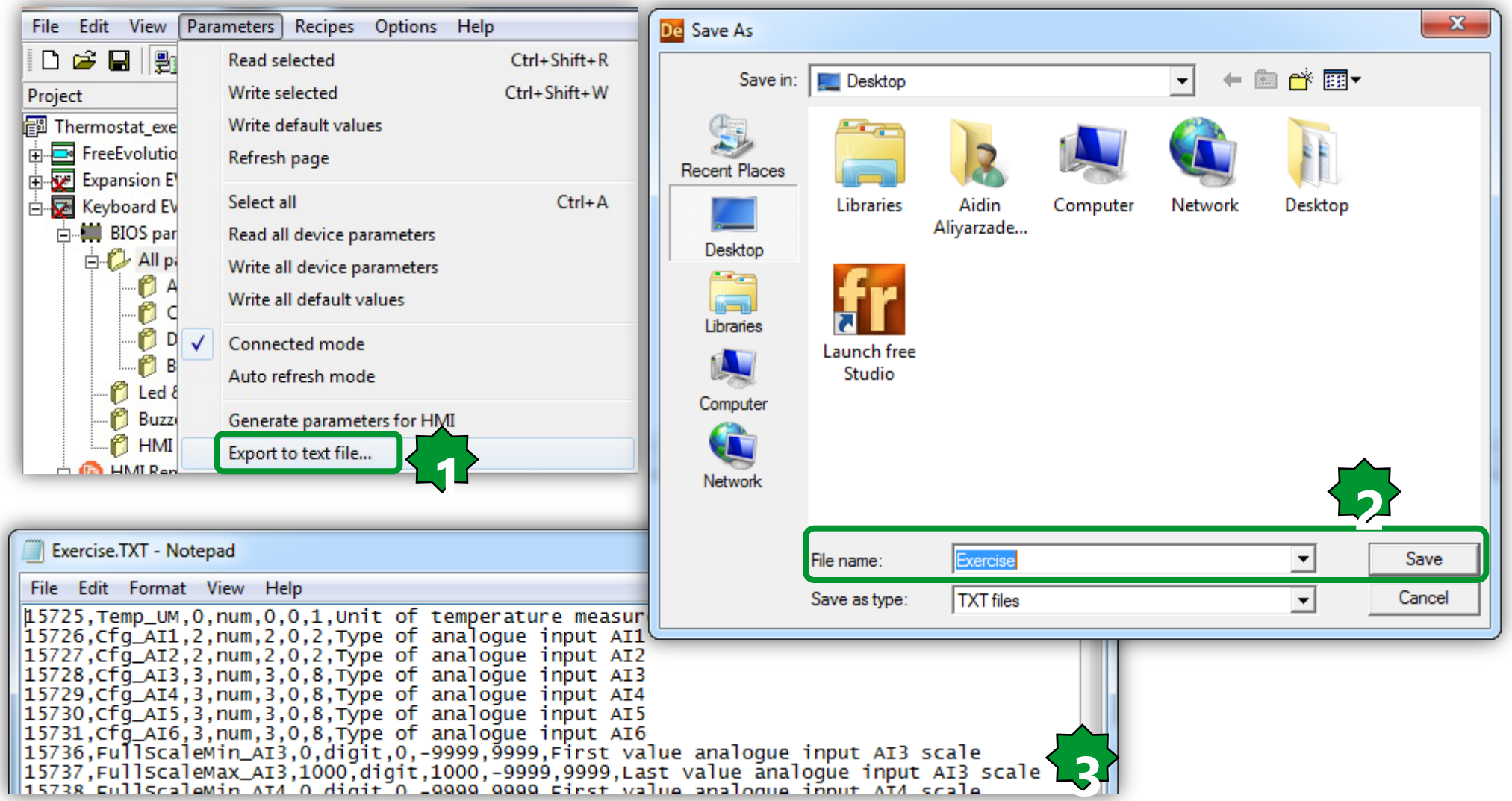


Select a Table

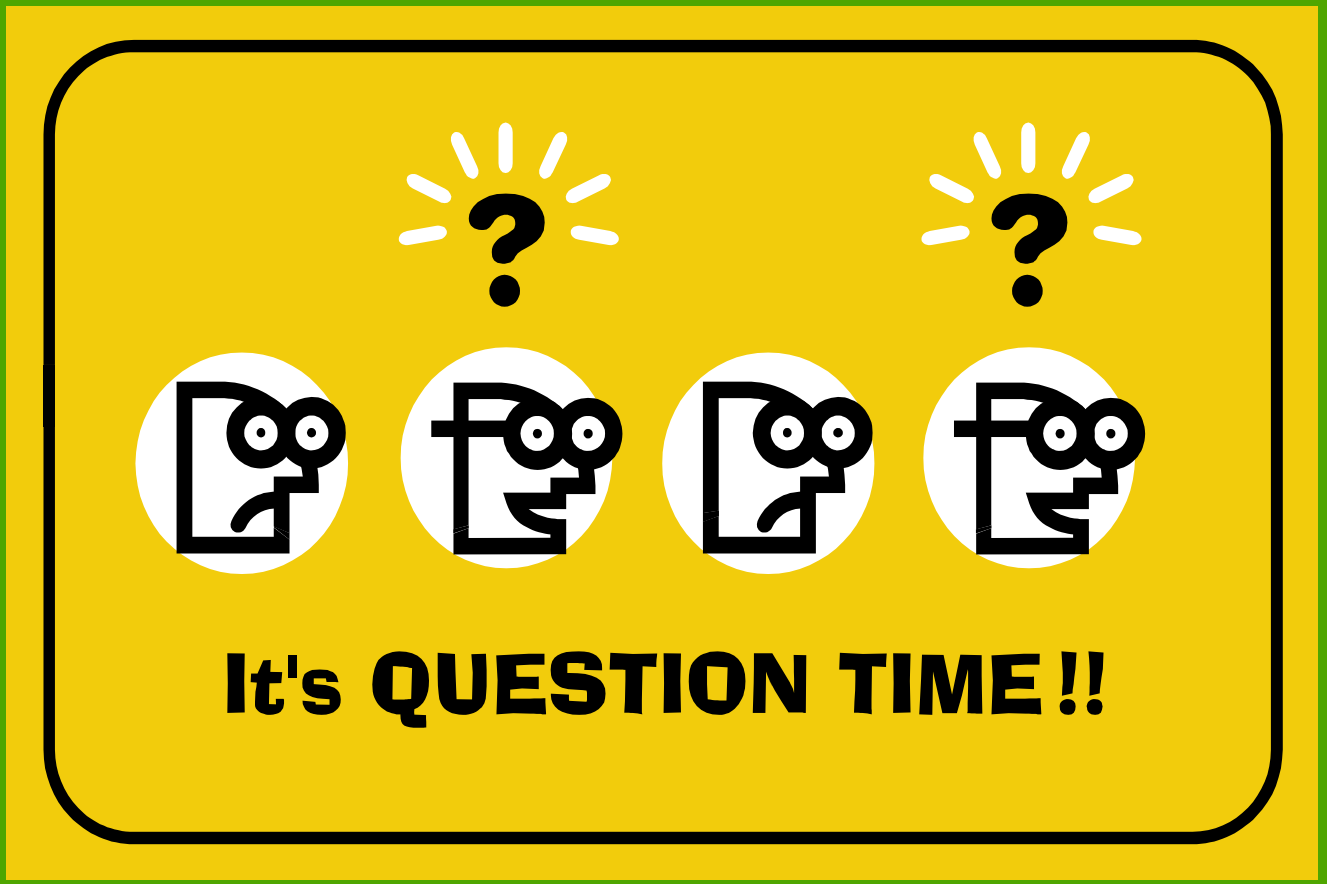
Analogue Inputs

Address	Name	Value	Um	Default	Min	Max	Description
15725	Temp_UM	0=°C	num	0=°C	0	1	Unit of temperature measurement
15726	Cfg_AI1	2=NTC(103AT	num	2=NTC(103AT	0	2	Type of analogue input AI1
15727	Cfg_AI2	2=NTC(103AT	num	2=NTC(103AT	0	2	Type of analogue input AI2
15728	Cfg_AI3	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI3
15729	Cfg_AI4	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI4
15730	Cfg_AI5	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI5
15731	Cfg_AI6	3=4÷20mA	num	3=4÷20mA	0	8	Type of analogue input AI6
15736	FullScaleMin_AI3	0	digit	0	-9999	9999	First value analogue input AI3 scale
15737	FullScaleMax_AI3	1000	digit	1000	-9999	9999	Last value analogue input AI3 scale
15738	FullScaleMin_AI4	0	digit	0	-9999	9999	First value analogue input AI4 scale
15739	FullScaleMax_AI4	1000	digit	1000	-9999	9999	Last value analogue input AI4 scale
15740	FullScaleMin_AI5	0	digit	0	-9999	9999	First value analogue input AI5 scale
15741	FullScaleMax_AI5	1000	digit	1000	-9999	9999	Last value analogue input AI5 scale
15742	FullScaleMin_AI6	0	digit	0	-9999	9999	First value analogue input AI6 scale
15743	FullScaleMax_AI6	1000	digit	1000	-9999	9999	Last value analogue input AI6 scale
15748	Calibration_AI1	0	°C/10,°F/10	0	-180	180	Analogue input AI1 differential
15749	Calibration_AI2	0	°C/10,°F/10	0	-180	180	Analogue input AI2 differential
15750	Calibration_AI3	0	digit	0	-1000	1000	Analogue input AI3 differential
15751	Calibration_AI4	0	digit	0	-1000	1000	Analogue input AI4 differential
15752	Calibration_AI5	0	digit	0	-1000	1000	Analogue input AI5 differential
15753	Calibration_AI6	0	digit	0	-1000	1000	Analogue input AI6 differential

Device - Export to Text file...



Q & A



Schneider Electric

Thanks

Schneider Electric