EasyPLC v.5 Quick Guide

EasyPLC Quick Guide

The purpose of this guide is that the user can start using EasyPLC quickly without the need to read the entire help manual.

Here is showed how to create and simulate our first logic program using the Ladder language. Please refer to the EasyPLC User Manual in order to know how to create programs in Script, Function Blocks and Grafcet languages and to see all the possibilities of the software.

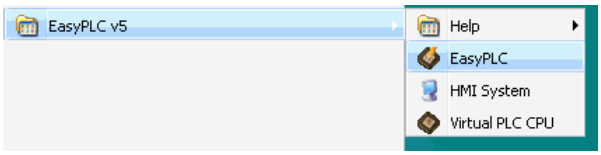
Also is showed how to compile and send the logic programs to Virtual PLC CPU, the utility designed to emulate the real PLC operation.

Finally is explained how to:

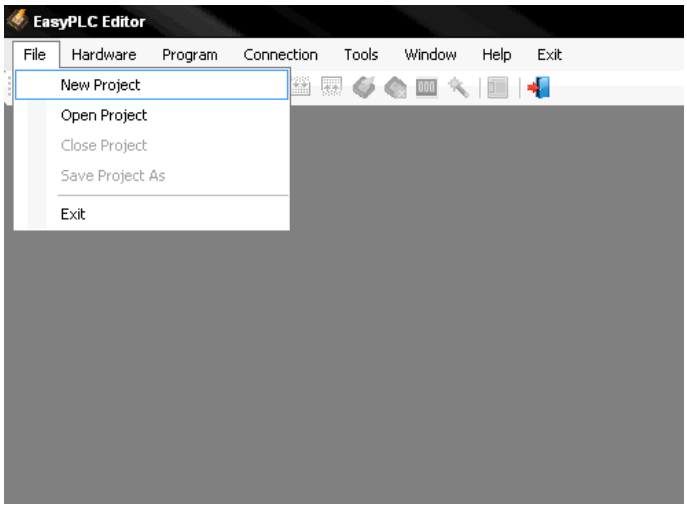
- Simulate the logic programs using the Simulated I/O Driver, useful when the users have not a physical hardware needed to read/write digital/analogic inputs/outputs.
- Execute a logic program using a real I/O interface.

Creating our logic program

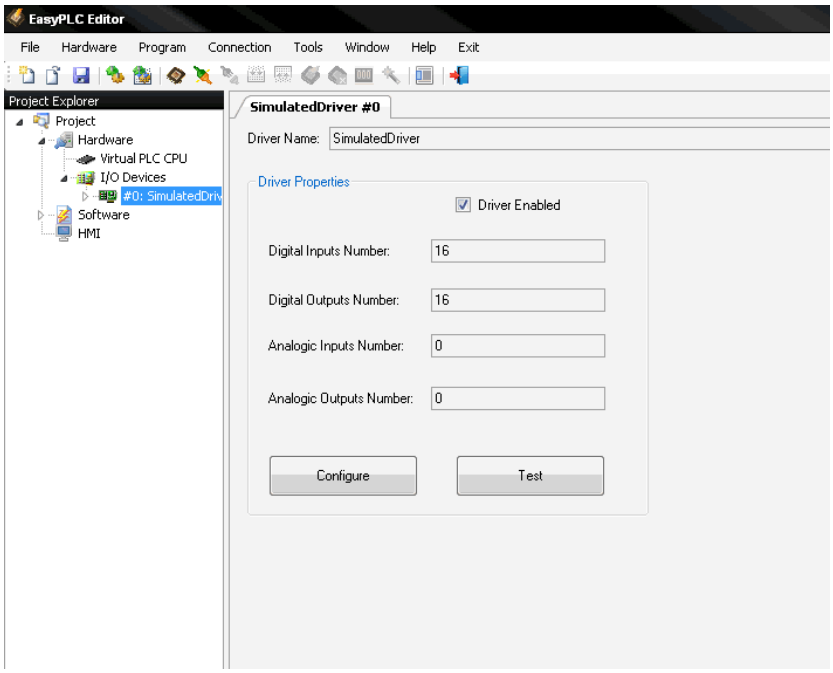
Open EasyPLC Editor.



Click on File -> New Project



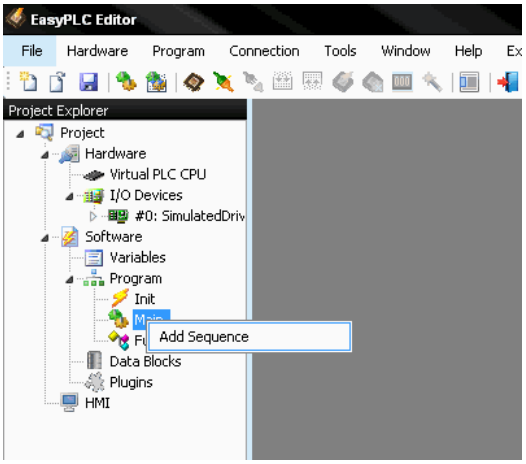
Click on File -> New Project



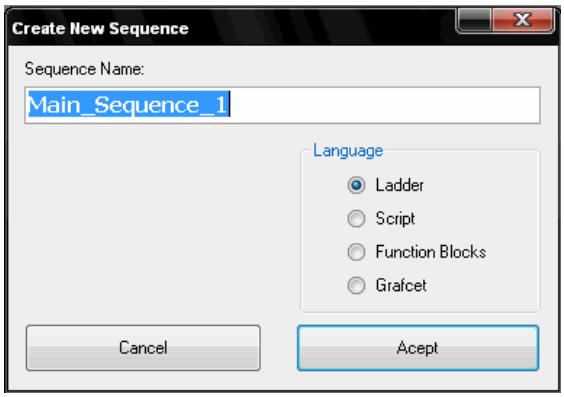
Then a new empty Project will be created. Click on the I/O Device of the Hardware node of the left tree of the Project Explorer. Then you see how a Driver has been automatically created. By default when a new Project is created, a Simulated I/O Driver is added to the Project.

This driver has 16 digital inputs and 16 digital outputs to simulate.

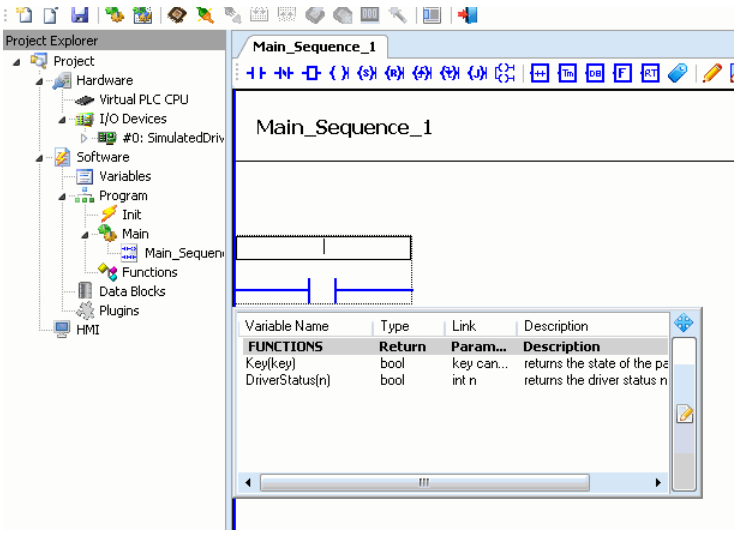
Now make click on Software -> Program -> Main node of the Project Explorer tree. Once the node is selected, press right mouse button, a contextual menu appears, select Add Sequence.

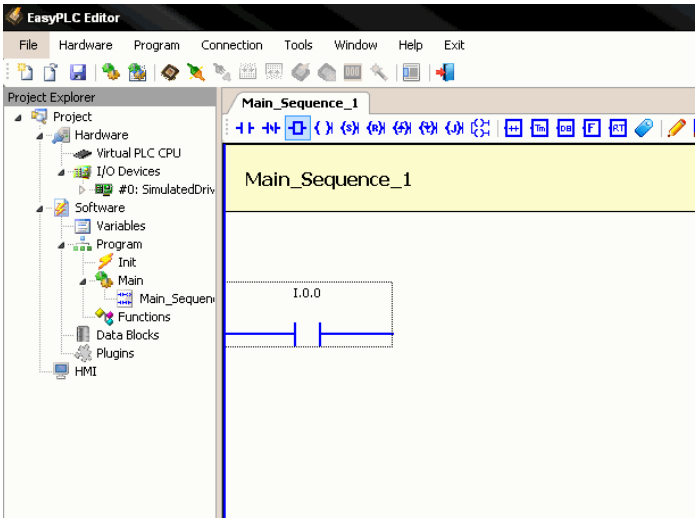


In the Create New Sequence window, let all the options as default and press Accept button. A new Ladder sequence is created in the project.

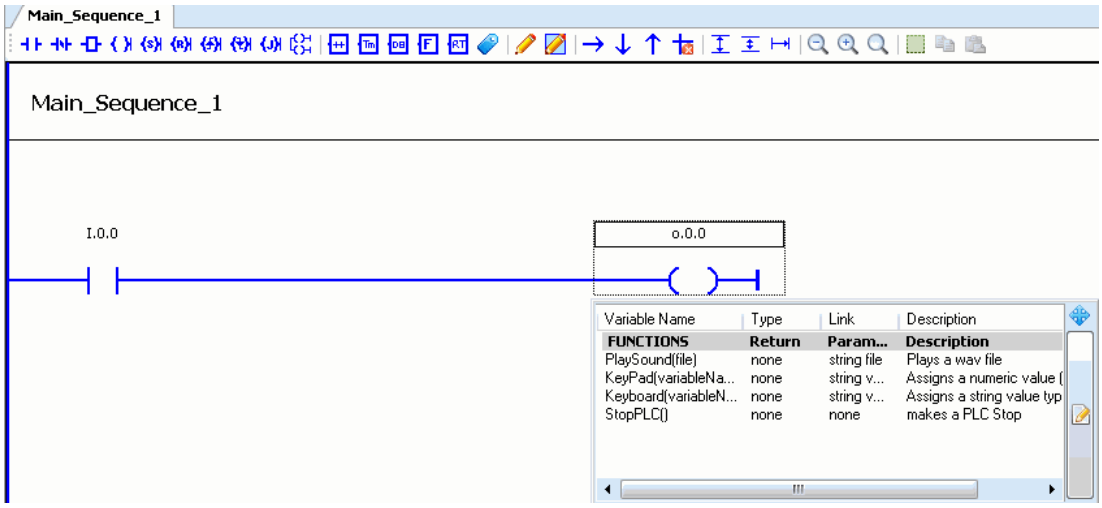


Click on a cell located in the first column and press insert Key, then a new contact is inserted, type I.0.0, this means that the contact will be activated when the input number 0 of the driver number 0 will be active. And press Enter key.

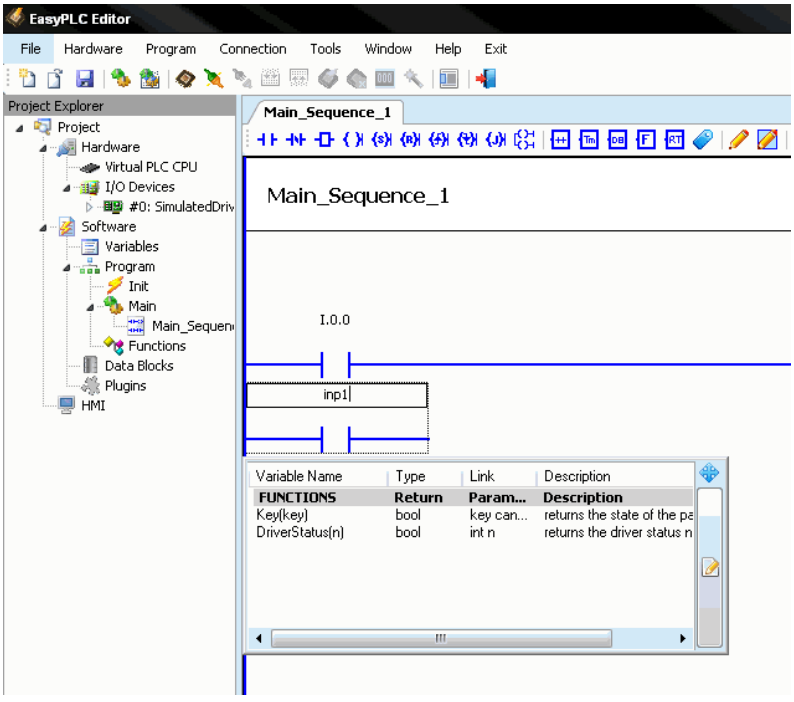




Now press Shift + C key twice, then you have created a wire, now press Shift + B Key to insert a coil, then press O.0.0 and press Enter Key. Now you have created your first segment or rung. This segment works in the following way, if input 0 of driver 0 is on, the output 0 of driver 0 will be also on, when input 0.0 will be off, output 0.0 will be also off.

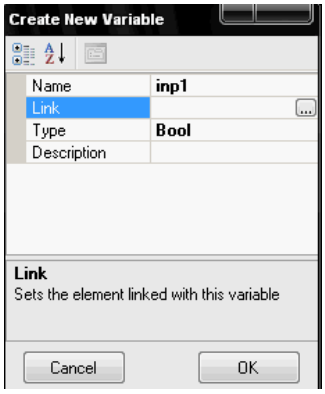


Let's add other segment, place the Mouse cursor down the first created segment and press Insert key or press the contact button form the upper tool bar, now type the following text: inp1 and press Enter.

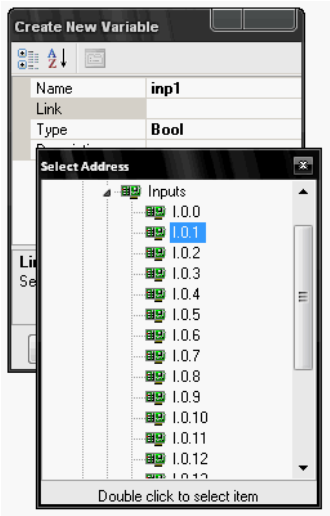


Automatically it will be added all the methods that implements this interface. You must program each method properly in order to make the driver work. Next are explained all the methods to be programmed:

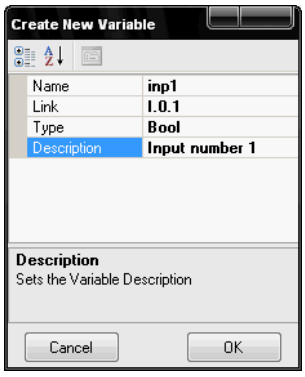
EasyPLC Editor does not recognize the word ‘inp1’ the opens the New Variable window. In this new contact we’re going to assign a variable, then click on Link cell, and press the button available in the right side.



From the Select Address Window, click on Inputs, and make double click on Input 0.1 (I.0.1).

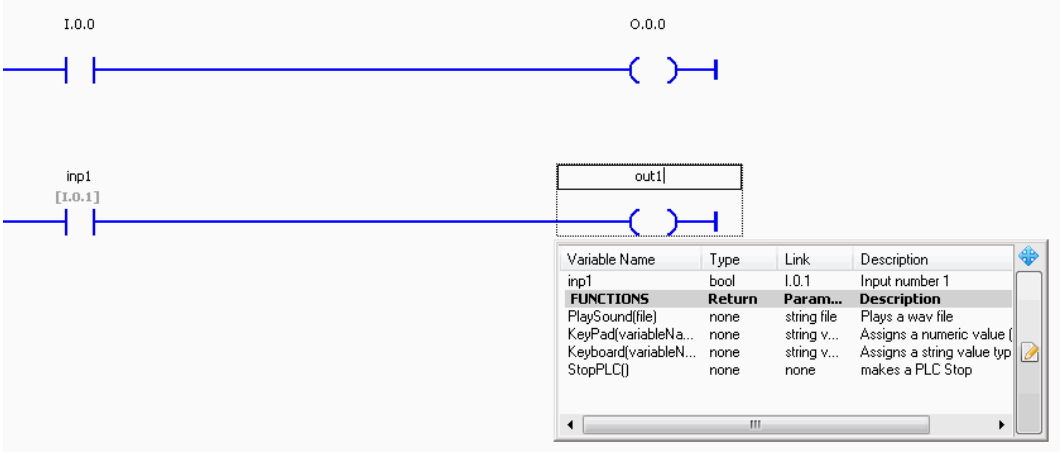


Also is possible to add a description, type the showed text and press OK button.

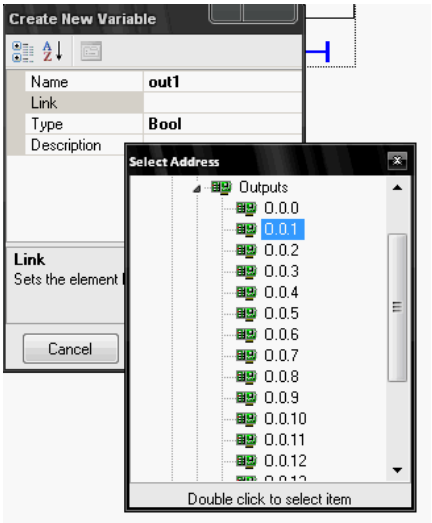


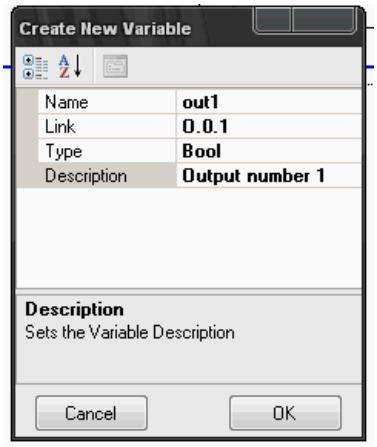
Now we have used a variable, the variables are very useful because they facilitate the programs writing. Now the variable name inp1 is linked to the state of the digital input number 1 of driver number 0 (simulated in this case).

Use Shift + C keys or cable toolbar button in order to create a wire, then add a contact and type out1, press Enter key.

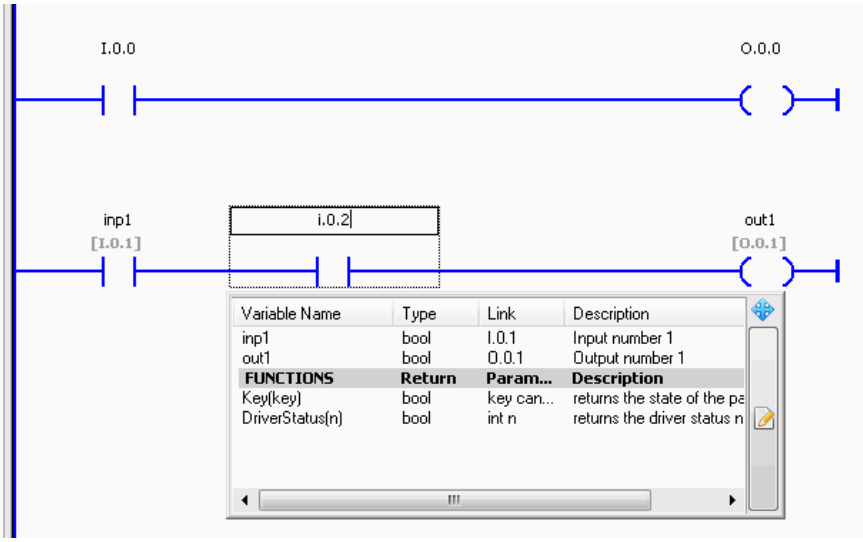


Same as before, the Create New Variable window appears, select Link cell and press the right button, select Outputs -> O.0.1 in the Select Address window. Finally add a description and press OK button.

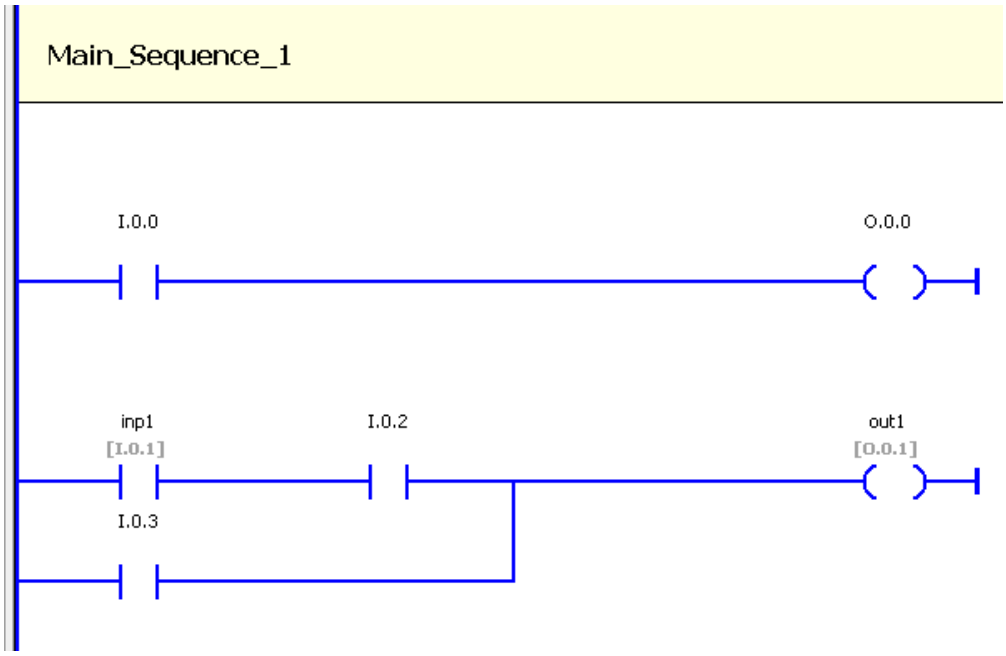




Now select the cell at the right of the inp1 contact and add a new contact, type I.0.2 and press Enter key.



Now select the cell below inp1 contact and add a new contact, type I.0.3. Use the cable tools to connect in the same way that the figure:



We have finished our first program, and works in this way:

Digital output 0 of driver number 0 will have the same value the digital input number 0 of driver number 0.

Digital output 1 of driver 0 will be on if:

- Are on the inputs 0.1 and 0.2
- Or if is on the input number 0.3

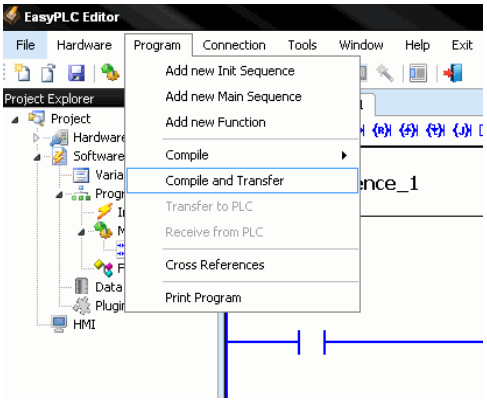
In other case output o.1 will be false.

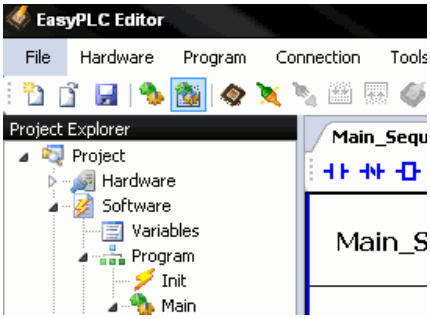
Compiling & send logic program to PLC

Once the logic program is written, is time to compile and send it to the PLC emulator. The PLC emulator program is called Virtual PLC CPU.

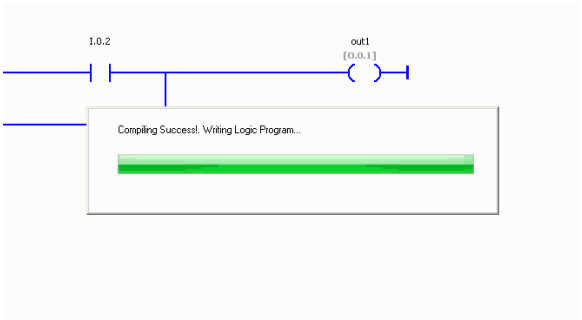
In order to compile the program select from the EasyPLC Editor, ***Program -> Compile and Transfer***.

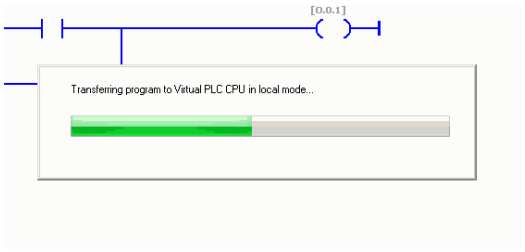
Or click on the EasyPLC Tool bar button ***Compile & Send Program***.



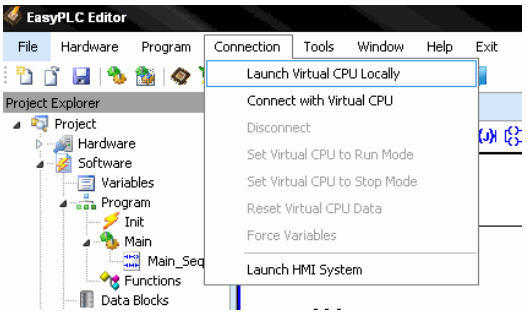


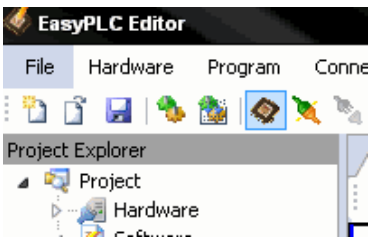
Once the compilation is started appears a progress bar indicating the compilation status, if the logic program has some errors, will inform about the sequence where the error is located, if there are no errors, the message: ***Compiling Success!, Writing Logic Program..*** is showed, and next appears other progress bar indicating the transferring process of the logic program to the PLC.



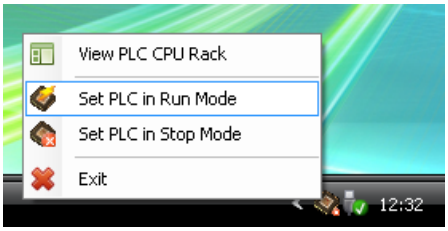


Now, we must launch Virtual PLC CPU in order to run the logic program, to do it, select ***Launch Virtual CPU Locally*** from the EasyPLC menu, or make click on the ***Virtual PLC PCU*** toolbar button.





When Virtual PLC CPU program is running, shows an icon in the Windows Tray area, by default the plc mode is stopped, then a red cross is showed indicating the state, click on the icon with the mouse right button, and from the contextual menu, select ***Set PLC in Run Mode***.

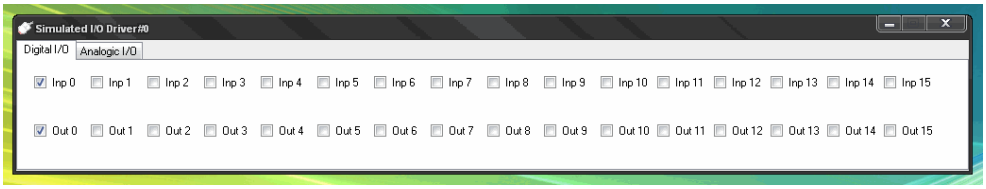


Then the Virtual PLC CPU icon changes, showing a thunder, this means that the PLC is in Run Mode (executing the logic program).

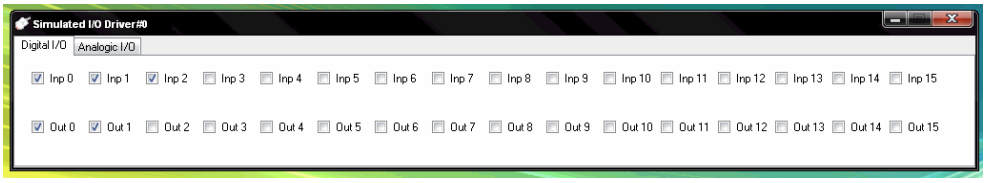
Also a new minimized window appears, the Simulated I/O driver, this window is showed due the Simulated I/O driver is included in our project; make click on the window title to maximize.

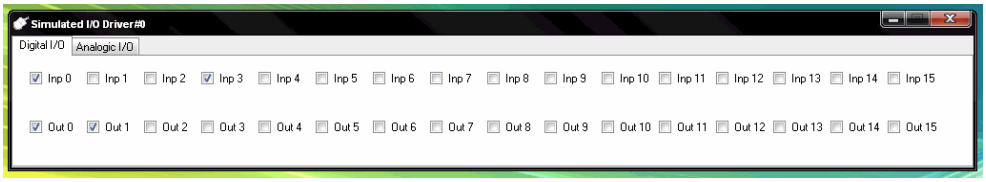
In this window is possible to force the digital inputs and to read the digital outputs. To force an input make click on it.

In order to see our logic program operation make click on the first input (***Inp 0***), then you see how the output 0 (***Out 0***) copy the ***Inp 0*** value.



Now let's test the second segment, force the input 1 and the input 2, then you see how output 1 is activated. Now deactivate input 1 and input 2 and activate input 3, see how output 1 is also activated.

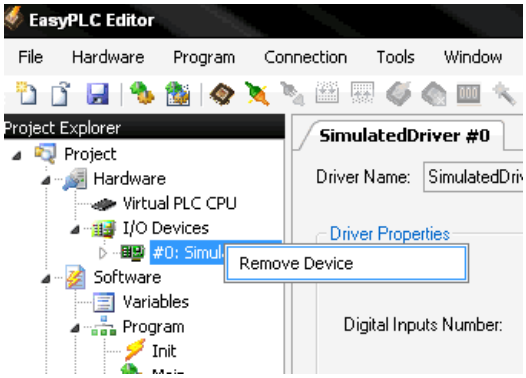




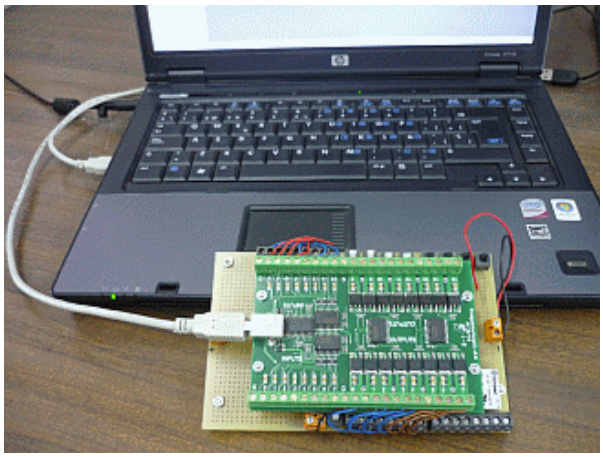
Using a real I/O device

In the previous example we have seen how to simulate a logic program, now we're going to use the same logic program but with a real I/O device, for example the Phidgets Interface Kit, in order to activate real outputs.

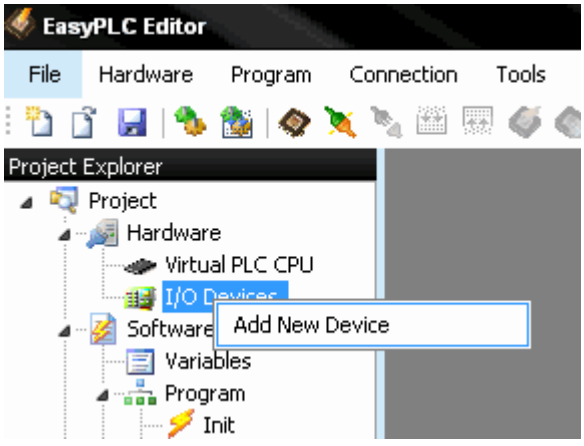
We're going to use the same logic program, only we change the I/O driver, then make mouse right click on the driver number 0, SimulatedDriver node (Project - > Hardware -> I/O Devices). From the contextual menu, select ***Remove Device***.



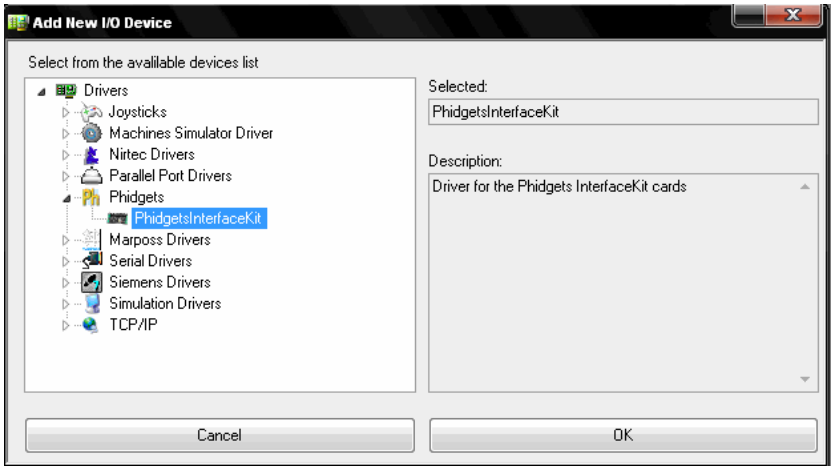
Now connect the Phidgets Interface Kit in one free USB port of your computer:



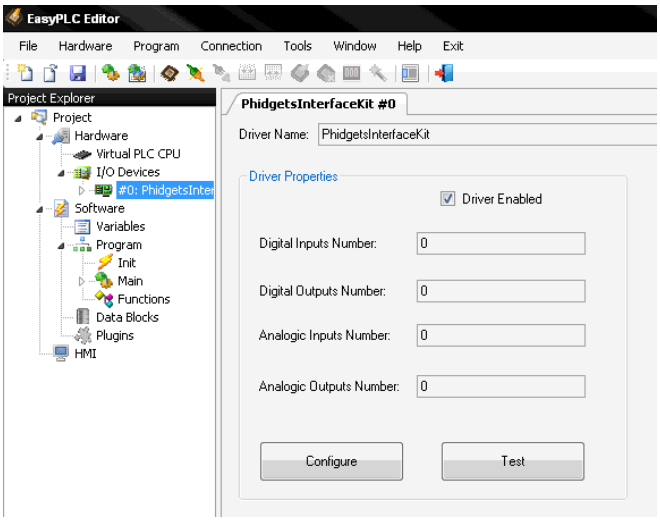
Mouse right click on I/O Devices, and select ***Add New Device***



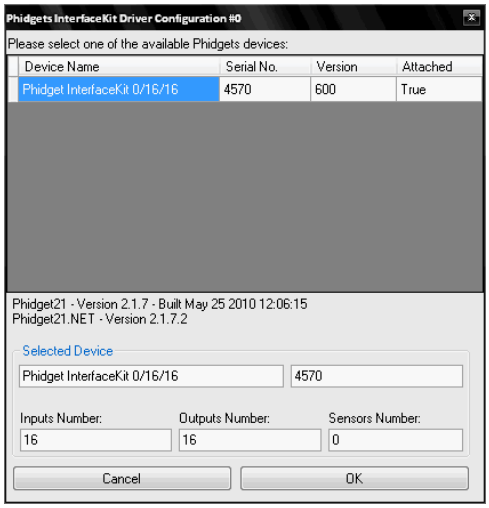
From the Add New I/O Device Window, select Phidgets -> PhidgetsInterfaceKit, and press Ok button.



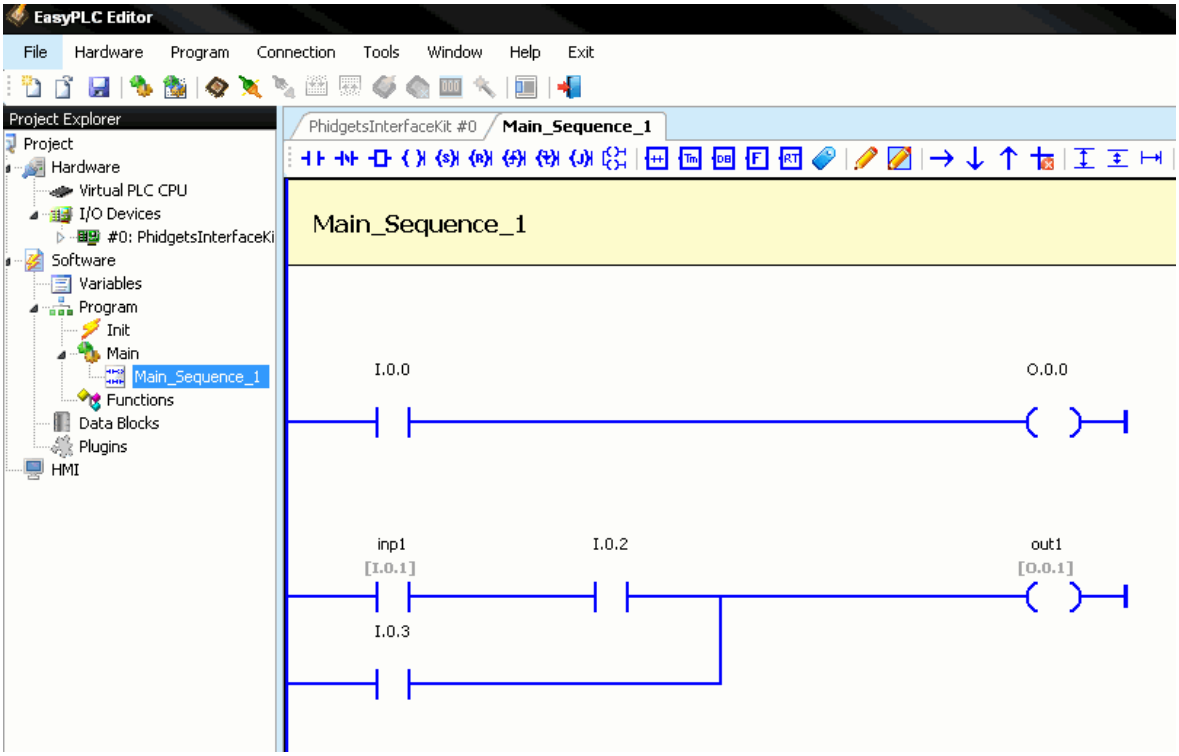
The driver is included in the project, now press Configure Button in order to select the Phidgets Interface Kit card to be used.



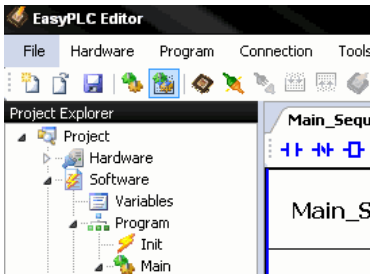
In the Configuration window, will appear the detected card connected in your computer; make click in the Device Name cell in order to select it. Press Ok button.



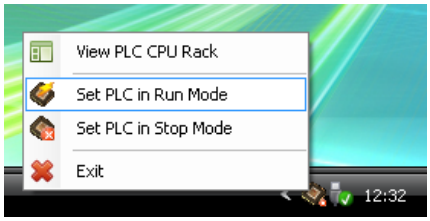
Now we have configured the hardware, we're using the same logic program, but only change the Driver:



We must to compile and send the program to Virtual PLC CPU.

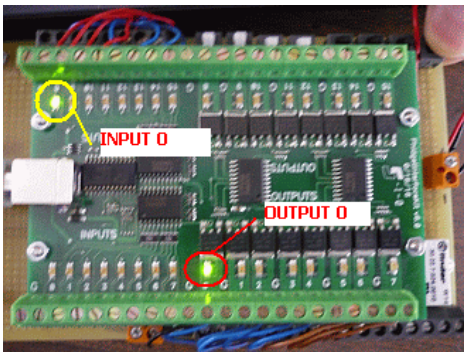


Launch Virtual PLC CPU program and change to Run Mode.

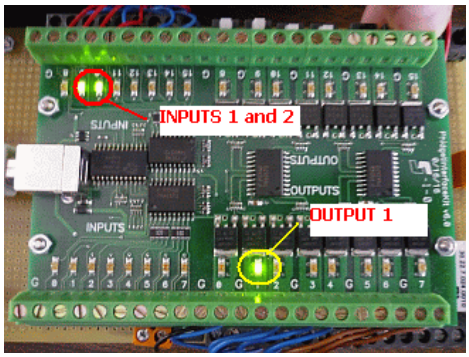


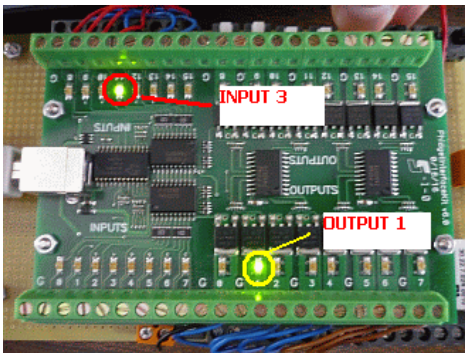
Let’s see how the program works:

If input 0 is activated, output 0 is activated, false otherwise.



If input 1 and input 2 are activated together or input 3 is activated, output 1 is activated, false otherwise.





OK, now we have made our first logic program, we have compiled and transfer to the PLC.

We have seen how to call to Virtual PLC CPU in order to execute and/or simulate the logic program.

Please read the EasyPLC User Manual in order to know all the possibilities of the software.

Good luck in your programming!

Idea, Design & Programming by Rafael Izquierdo
Valencia (Spain)

www.nirtec.com