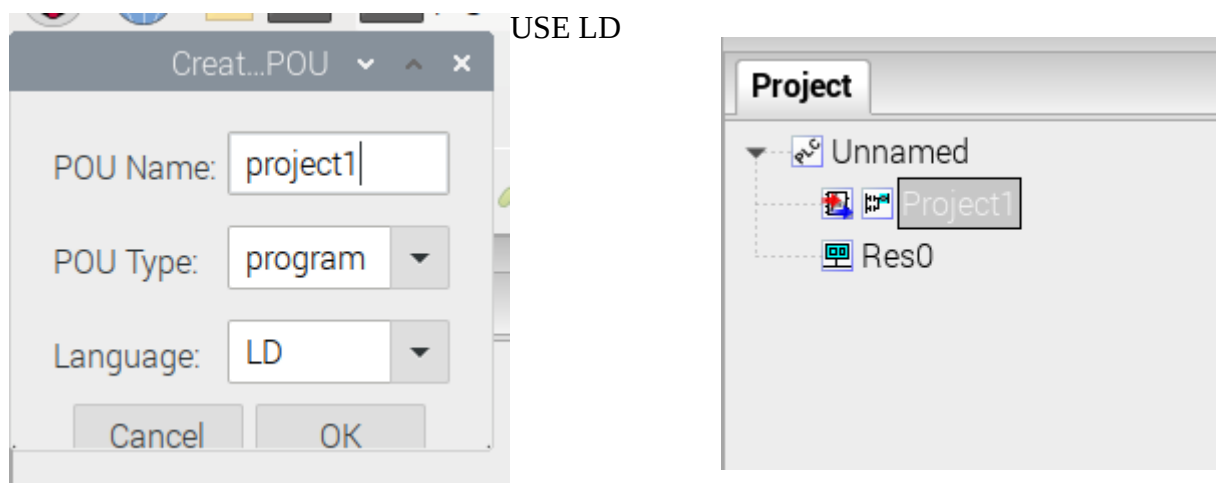


Matthew Townsend

2/28/23

## Lab 2

In this lab day we created went over the functionality of a plc circuit such as the on the openplc website documentation. From the voltage entering to it interfacing with the plc and then to the output. The PLC uses memory location with the scheme %IX0.0, %IX0.1, %QX0.0 %QX0.1 with I for input memory and q for output memory. This is the base for what the code we are writing will use to create a functional circuit. After opening openPLC we will go to the top right button that says file and create new project. It is named Project1, it will appear like this after completion.



Next in the center of the page that is the workspace to create the program. In this the top box is for variables such as input and output. You can double click Type to change the variable type, for a button and LED a simple bool will work. This will allow for the actual buttons on a physical bard to function with in a circuit.

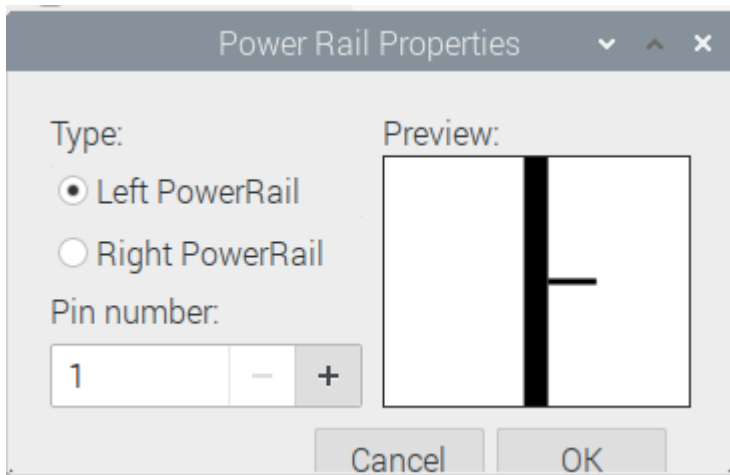
Description:		Class Filter:	All						
#	Name	Class	Type	Location	Initial Value	Option	Documentation		
1	input1	Local	BOOL	%IX0.0					
2	input2	Local	BOOL	%IX0.1					
3	LED	Local	BOOL	%QX0.0					

Below this is where a The actual ladder programming will be. To place an object in there is a bar along the top of the screen,

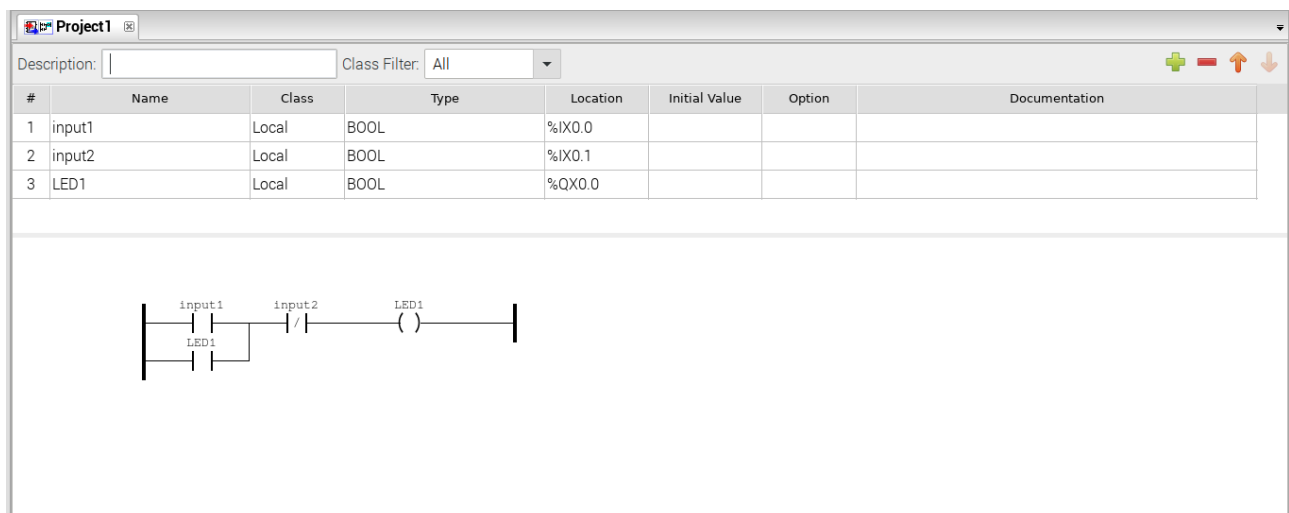


the ladder

Clicking one of the objects then clicking the lower box will create the object then prompt the user for input. Click ok then it will look like this. This will begin the circuit



After adding the functionality such as the contacts || and coils ( ), the program will look like this



The ladder will display as this because for the circuit to function properly it requires the PLC to translate this on to the board. If input 1 is pressed, electricity will travel to the LED and activate, press both and the light will stop, button two negates the circuit. The LED 1 contact below will be activated if the light is on, so if button 1 is no longer pressed, the light will continue because the electricity is passing through that contact now to the led. Button 2 will still kill the circuit even with the led contact on. To run the program, press one of these three buttons. We used the blue man



The screen will then look like this after running and you can interact with it through right clicking the objects and turning them on or off. Here are the examples

Description:

Class Filter: 

All

#	Name	Class	Type	Location	Initial Value	Option	Documentation
1	input1	Local	BOOL	%IX0.0			
2	input2	Local	BOOL	%IX0.1			
3	LED1	Local	BOOL	%QX0.0			

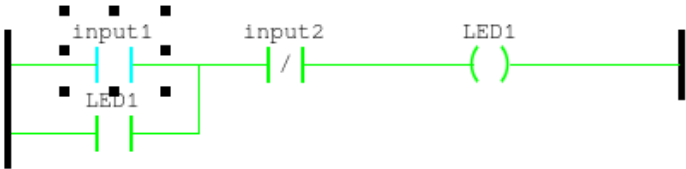
Debug: Config0.Res0.instance0

```
graph LR
    R1[ ] --- I1[input1]
    R1 --- I2[LED1]
    I1 --- I3[ ]
    I2 --- I3
    I3 --- I4[/ /]
    I4 --- O1[( )]
    O1 --- R2[ ]
```

Button 1 is pressed so it is completed

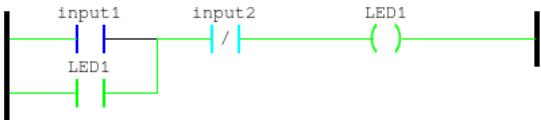
Debug: Config0.Res0.instance0

50



Button 1 is not pressed but led1 is because the led activated

Debug: Config0.Res0.instance0



Button

two is pressed and blocks the electricity from passing it

Debug: Config0.Res0.instance0

