الجامعة الإسلامية العالمية ماليزيا
INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA
يونيورسيتي إسلام انتارابغسا ملايسيا

# MECHATRONICS SYSTEM INTEGRATION

## EXPERIMENT 5: Programming Logic Controller Interfacing

| Name | Matric No. |
|---|---|
| Muhamad Imran Haziq Bin Haizam | 2216429 |
| Tanisha Rowshan Saleh | 2228290 |
| Muhammad Yusof Irfan Bin Abidin | 2210915 |
| Ariesha Daliela Hanim Binti Amir | 2316834 |

Date Of Submission: 19/04/25

# TABLE OF CONTENTS

## Abstract

Experiment 5 served to help understand the working principles of PLCs (programmable logic controllers). An Arduino board was set up with two push buttons connected to it, represented by a negated contact and a normally-open contact as their ladder-diagram equivalents. The software used to create a ladder diagram for this set-up is called OpenPLC Editor. The ladder diagram was designed to allow an LED connected to the Arduino board to light up upon one of the buttons being pushed, and turn off when the other button was pressed. However, once the set-up was completed, the LED lit up of its own accord, without being prompted by its designated turn-on push button. Aside from that, it couldn't be turned off by the turn-off push button either.

## Introduction

Programming logic controllers, also known as PLCs, are special computers used to control and automate both mechanical and electrical processes. They are commonly used in manufacturing, power generation and chemical processes, to name a few. In this experiment, an Arduino board was interfaced with OpenPLC Editor, a software used to program and simulate PLCs. This was done in order to get a better grasp of how PLCs function and to appreciate their versatility.

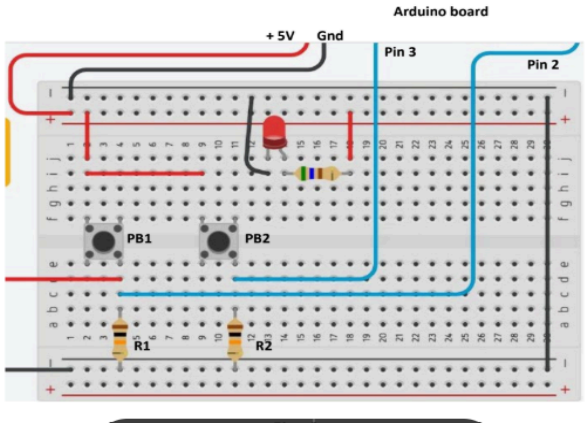
An LED was connected to the Arduino board, along with two push buttons to turn the LED on and off, to demonstrate how PLCs work.

## Materials And Equipment

- ★ OpenPLC Editor software
- ★ Arduino Board
- ★ 2 Push Button Switches
- ★ Jumper Wires
- ★ LED
- ★ Resistors (10k ohm and 560 ohm)
- ★ Breadboard

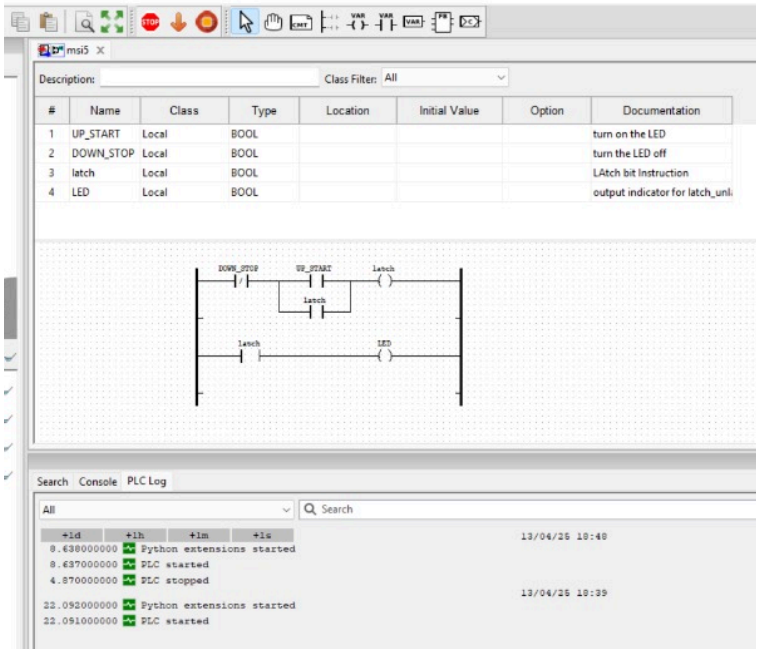
## Experimental Set-Up

The Arduino and bread boards were configured as shown in the figure below:




## Methodology

- ★ Configure a ladder diagram on OpenPLC Editor
- ★ Specify all the variables used in the diagram
- ★ Simulate the diagram
- ★ Upload it to the Arduino set-up
- ★ Select the correct communication port (COM)
- ★ Test the circuit




(Ladder diagram)

## Results

Despite configuring the circuit according to the circuit diagram and creating the ladder diagram accordingly, the LED lit up before the turn-on push button was pressed, and it didn't turn off when the turn-off button was pressed.

## Discussion

The goal of this experiment was to understand how programming logic controllers (PLC) function and to implement them with an Arduino set-up, to mimic real-life scenarios where PLCs are used.

The experiment failed to be successful, as the LED lit up before any of the push buttons was pressed, and it wouldn't turn off as well, when either of the buttons was pressed, despite the ladder diagram being configured correctly. This can be attributed to either the circuit being wired incorrectly or an issue with the push buttons.

The expected outcome was that the LED would light up when the turn-on button was pushed, and it would turn off when the turn-off button was pushed.

## Conclusion

This experiment was held to better understand programming logic controllers and their applications. An Arduino board was set up with two push buttons and an LED, and a corresponding ladder diagram was made using OpenPLC Editor. The ladder diagram was then interfaced with the Arduino set-up, with the expected outcome being that the LED would turn on or off, when either of the push buttons was pushed. However, this did not happen, and the LED turned on before any of the push buttons was pressed. It wasn't responsive to any of the push buttons and just stayed lit up. One of the factors causing this could have been incorrect wiring or a problem with the push buttons.

## Recommendation

1) Troubleshoot the components before building and testing the circuit
2) Rewire the circuit