LAB REPORT

MCTA 3203

SECTION 1

GROUP D

EXPERIMENT 5

Understanding both software and hardware aspects of PLC interfacing with Microcontrollers.

MATIC NO.	NAME
2111633	MUHAMMAD FARID BIN JAFRI
2110765	MUHAMMAD ARIF SYAZWAN BIN MOHD RIZAL
2112969	MUHAMMAD HAZIQ HAIKAL BIN SUAIB
2115207	MUHAMMAD AMIRUL SYAFIQ BIN RUSLANI
2116281	MUHAMMAD FADHLUL WAFI BIN AHMAD NAIM

SUBMISSION DATE: 15/11/2023

INSTRUCTOR: DR. NADZRIL BIN SULAIMAN

Abstract

Programmable Logic Controller or PLC is known as a specialized industrial computer used for automating processes, particularly in manufacturing environments. PLCs are programmable and can control a wide range of machinery and processes.For example, in a manufacturing plant, a PLC might be used to control the assembly line, managing tasks such as controlling robotic arms, monitoring sensors, and ensuring that each step in the production process occurs in the correct sequence.

Arduino refers to an open-source electronics platform based on easy-to-use hardware and software. It consists of a programmable circuit board (microcontroller) and a development environment, making it accessible for hobbyists, students, and professionals to create interactive electronic projects. Arduino boards are widely used for prototyping and building various electronic devices due to their simplicity and versatility.

In this modern world, we can interface the PLC with the arduino to create something useful for all. There are various applications of PLC with Arduino that we can do, and of it is by using OpenPLC software.

Table of Contents

Introduction 4
Materials and Equipment 5
Experimental Setup 6
Methodology 8
Results (Observation) 10
Discussions 11
Conclusion 12
Recommendations 13
References 14
Student's Declaration 15

Introduction

Arduino microcontrollers and programmable logic controllers (PLCs) are now essential parts of automation and control systems. They offer flexible platforms that enable the design and implementation of a broad range of applications, ranging from home automation to industrial automation, and they enable the control and monitoring of numerous devices and processes. By combining these two potent technologies, this experiment seeks to combine the advantages of PLCs and Arduino, producing a flexible and affordable solution for control and automation tasks.

We are going to explore the benefits and features of both PLCs and Arduino in this experiment, as well as how they may complement one another to solve practical automation problems. We can take advantage of both the flexibility and programmability of the Arduino and the powerful industrial control capabilities of the PLC by linking these two components. This combination creates new and interesting opportunities for designing efficient, flexible control systems that are tailored to individual requirements.

Upon completion of this experiment, we will possess the skills and knowledge necessary to take on challenging automation jobs in a variety of industries and applications. We will also have a stronger understanding of how to construct advanced control systems that take advantage of the capabilities of both PLCs and Arduino. This integration is a significant advancement in the field of automation and control, making it a useful ability for technicians, engineers, and enthusiasts alike.

Materials and Equipment

Material needed:

1. OpenPLC Editor software
2. Arduino Board
3. 2 Push Button Switches
4. Jumper Wires
5. LED
6. Resistors
7. Breadboard

Equipment:

1. Computer with OpenPLC Editor software for ladder diagram construction.

Experimental Setup

1. A ladder diagram was created as shown in Fig. 1.
2. All variables used have been specified in the ladder diagram.
3. The ladder diagram has been compiled and simulated in OpenPLC Editor.
4. The ladder diagram was uploaded to the Arduino board.
5. COM port number and all pin associations between the OpenPLC variables and Arduino board have been correctly ensured.
6. A circuit was built as shown in Fig. 2.
7. The functionality has been tested.

#	Name	Class	Type	Location
1	UP_Start	Local	BOOL	%IX0.0
2	DOWN_Stop	Local	BOOL	%IX0.1
3	Latch	Local	BOOL	
4	LED	Local	BOOL	%QX0.0

Documentation
Will Turn ON the LED
Will Turn OFF the LED
Latch Bit Instruction
Output indicator for Latch_Unlatch Program

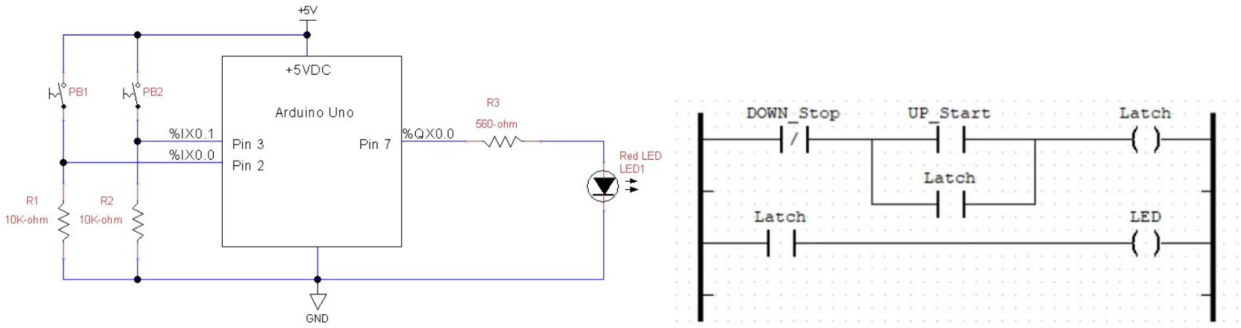


Fig. 1

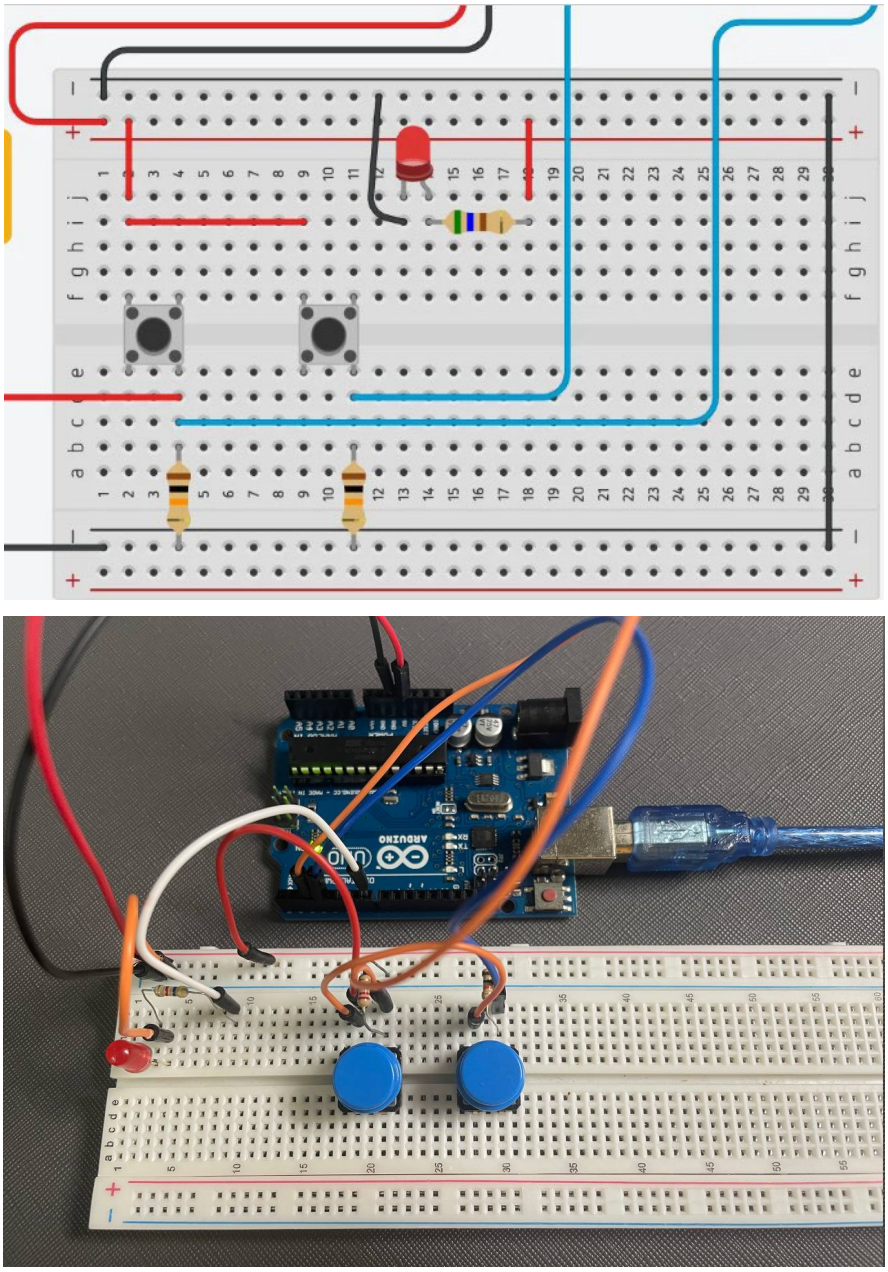
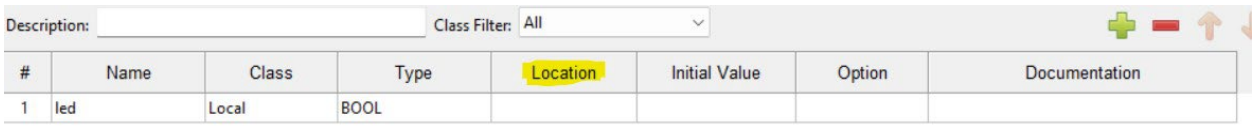


Fig. 2

Methodology

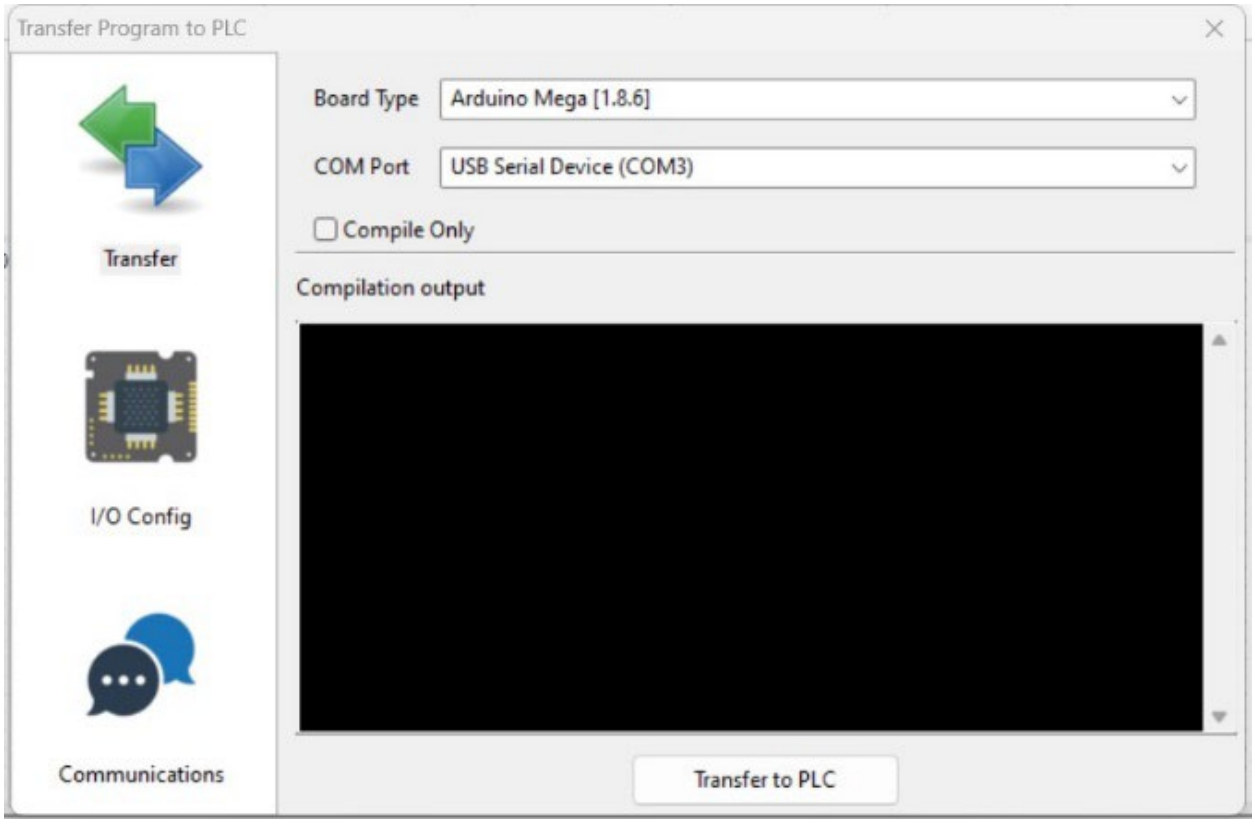
Procedures:

1. Started a new project (File → New → New Folder and named it → Selected Folder).
2. The project was named in the Create a new POU pop up window, and the language was selected as LD (ladder diagram).
3. We pressed the button  to create a new variable and name it. (Class Filter was set to local and Type was BOOL).
4. We started creating the Ladder Diagram by right clicking and choosing Add.
5. The Left the Right power rails, Contact (negated), and Coil were created as shown in Fig. 2.
6. Once the ladder diagram was created, it was compiled by clicking the icon  .
7. The blinking led simulation was now seen in the editor.
8. The variables that were created in the Ladder Diagram were associated with the led pin on the Arduino board. You could check the associated pin between open PLC variables and various types of microcontrollers from the OpenPLC Editor website. On the Documentation menu, I chose OpenPLC Runtime and then Physical Addressing. The correct microcontroller type and digital out pins were referred to.
9. Using an Arduino Mega board, the led was connected to pin no. 14, the associated Digital Out pin on the OpenPLC Editor was %QX0.0. This needed to be typed into the Location of the variable in the editor.



10. The Arduino board was connected to your laptop and the COM port number was noted.
11. Once connected, the ladder diagram program was compiled to the Arduino board by clicking the icon  . A window popped up, I selected the correct board, typed in the

correct COM port number, and proceeded by pressing the Transfer to PLC button.



12. The LED was now blinking according to the written ladder diagram program.

Results (Observation)

Once the ladder diagram program is compiled to the Arduino board, the LED will blink according to the button pressed. If the left button is pressed, the LED will turn on and if the right button is pressed, the LED will turn off.

Link for the video:
https://drive.google.com/file/d/18KyVWekUjfQkJa-q65wtWgXHbS8uDp0p/view

Discussions

The start-stop control circuit designed and simulated using OpenPLC demonstrated effective functionality in both the simulation environment and on the Arduino board. The ladder diagram, created with OpenPLC, accurately represented the logic required for initiating and halting a process.

Upon careful observation, it was noted that the start-stop control circuit responded promptly to simulated inputs and physical button presses on the Arduino board. This responsiveness is crucial for applications requiring precise control over the initiation and termination of processes. The ladder diagram's execution on the Arduino platform showcased the adaptability and reliability of the implemented control logic.

A notable aspect of the discussion involves the comparison between simulation and actual implementation. While the simulation environment provided a controlled setting for testing the ladder logic, the transition to the Arduino board brought forth considerations such as real-world delays and hardware constraints. Analyzing any disparities between simulation and actual results allowed for a deeper understanding of the circuit's behavior and potential areas for optimization.

Conclusion

In conclusion, this experiment effectively showcased the practical application of a start and stop control circuit using a ladder diagram within OpenPLC software, exemplifying its relevance in industrial automation. We've gained insights into integrating software and hardware elements of PLCs with Microcontrollers. The constructed circuit, tailored to regulate an LED, consistently responded to the On and Off buttons. Activation of the On button successfully latched the connected circuit, resulting in continuous LED illumination until the Off button disengaged the circuit, creating an open circuit on the breadboard.

The experiment also emphasized crucial features essential in electronic integrated circuits, such as response time and feedback mechanisms. A rapid response time demonstrated the circuit's reliability for real-time control applications. Additionally, timers were utilized in earlier experiments to illustrate their function within the OpenPLC software, allowing users to preset them based on experimental outcomes. Short-duration timers showcased the software's ability to process received data promptly in real-time. The successful functionality of the push buttons underscored the effective integrated feedback system between Arduino, circuit, and software.

Recommendations

To enhance our experiment's efficacy, we can explore integrating a greater variety of components into our ladder system. By introducing additional elements like multiple LEDs, switches, resistors, and servo motors, we can expand the capabilities of our programmable codes. This expansion will enable us to develop a more intricate system capable of emulating real-life scenarios, such as traffic light simulations or pedestrian crossing signals, automatic doors with lights indicator and sensor, thus reflecting practical applications in our daily lives. Furthermore, we can elevate our experimentation by simulating more intricate systems and circuits within the PLC software to explore more complex and sophisticated PLC applications.

References

1) *AMCI : Advanced Micro Controls Inc :: What is a PLC?* (n.d.). https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/what-plc/#:~:text=A%20PROGRAMMABLE%20LOGIC%20CONTROLLER%20

2) *Welcome to OpenPLC | OpenPLCProject*. (n.d.). https://openplcproject.gitlab.io/

3) PLC ladder logic on an Arduino: Building a start-stop circuit - technical articles. Control. (n.d.). https://control.com/technical-articles/plc-ladder-logic-on-an-arduino-building-a-start-stop-circuit/

Student's Declaration

Certificate of Originality and Authenticity

This is to certify that we are responsible for the work submitted in this report, that the original work is our own except as specified in the references and acknowledgment, and that the original work contained herein has not been untaken or done by unspecified sources or persons.

We hereby certify that this report has not been done by only one individual and all of us have contributed to the report. The length of contribution to the reports by Each individual is noted within this certificate.

We also hereby certify that we have read and understand the content of the total report and no further improvement on the reports is needed from any of the individual contributor to the report.

We, therefore, agreed unanimously that this report shall be submitted for marking and this final printed report has been verified by us

Signature: *arifsyazwan* — Read ✔️
Name: Muhammad Arif Syazwan Bin Mohd Rizal — Understand ✔️
Matric Number:2110765 — Agree ✔️

Signature: *haziqhaikal* — Read ✔️
Name: Muhammad Haziq Haikal bin Suaib — Understand ✔️
Matric Number: 2112969 — Agree ✔️

Signature: *amirul* — Read ✔️
Name: Muhammad Amirul Syafiq Bin Ruslani — Understand ✔️
Matric Number: 2115207 — Agree ✔️

Signature: *Wafi* — Read ✔️
Name: Muhammad Fadhlul Wafi Bin Ahmad Naim — Understand ✔️
Matric Number: 2116281 — Agree ✔️

Signature: *faridjafri* — Read ✔️
Name: Muhammad Farid Bin Jafri — Understand ✔️
Matric Number: 2111633 — Agree ✔️