LABORATORY REPORT

LAB 4:
UNDERSTANDING BOTH SOFTWARE AND HARDWARE ASPECTS OF PLC INTERFACING WITH MICROCONTROLLERS

GROUP B

PROGRAMME: MECHATRONIC ENGINEERING

GROUP MEMBERS:	MATRIC NO:
1. MUHAMMAD AFIQ BIN AHMAD	2115203
2. HANIS BINTI MOHD IZANI	2020590
3. SARAH AISHAH BINTI SA'AID HAZLEY	2117600
4. NUR AYU NAZIERA BINTI ROSLI	2119202
5. NOORUL GHOUSIAH BINTI NOORDEEN SAHIB	2118298

DATE OF SUBMISSION:
Wednesday, 15th November 2023

ABSTRACT

Students are introduced to the integration of microcontrollers and programmable logic controllers (PLCs) in mechatronic systems through this hands-on manual. Using the OpenPLC Editor, the emphasis is on comprehending the hardware and software components of this integration. The example shows how to simulate and create a ladder diagram for an LED that blinks, then upload it to an Arduino microcontroller. The creation of a Start-Stop Control Circuit is the next practical assignment in which students design, model, and transfer a ladder diagram to operate an Arduino-based circuit that has push buttons and LEDs. This hands-on method improves understanding of PLC-microcontroller integration and strengthens theoretical knowledge through actual applications.

TABLE OF CONTENTS

Table of Contents

INTRODUCTION

Mechatronics relies heavily on the combination of microcontrollers and Programmable Logic Controllers (PLCs) to automate and control electromechanical processes. The smooth coordination of industrial computers and microcontrollers made possible by this integration makes effective control possible in manufacturing facilities and other applications.

The OpenPLC Editor is a software application designed specifically for PLC-microcontroller interface, and this session focuses on its practical aspects. This experiment uses an Arduino board to show how ladder diagrams may be created, simulated, and uploaded into microcontrollers using the OpenPLC Editor.

In this experiment, we create and simulate a basic ladder diagram for a blinking LED using the OpenPLC Editor. The ladder diagram is then uploaded onto an Arduino microcontroller, offering a foundational understanding of PLC-microcontroller interaction in both software and hardware aspects.

The subsequent exercise involves constructing a Start-Stop Control Circuit using a ladder diagram in the OpenPLC Editor. We go through the steps of creating, compiling, and simulating the ladder diagram before uploading it to an Arduino board. This hands-on exercise provides practical experience in developing and testing a control system. This integrated approach, blending theoretical knowledge with practical application, equips learners with the necessary skills to navigate the complex integration between PLCs and microcontrollers in mechatronic systems. The steps provided and references serve as a guide, offering a comprehensive and insightful learning experience in Mechatronics System Integration.

PROCEDURE

Materials And Equipment

- OpenPLC Editor software
- Arduino Uno Board
- 2 Push Button Switches
- Jumper Wires
- LED
- Resistors 1kΩ and 250Ω
- Breadboard

Experimental Setup

The ladder diagram and the circuit connections are setup as shown in figures below.

Figure 1: the ladder diagram

Description: Class Filter: All

#	Name	Class	Type	Location	Initial Value	Option
1	DOWN_Stop	Local	BOOL	%IX0.1		
2	UP_Start	Local	BOOL	%IX0.0		
3	Latch	Local	BOOL			
4	Led	Local	BOOL	%QX0.0		

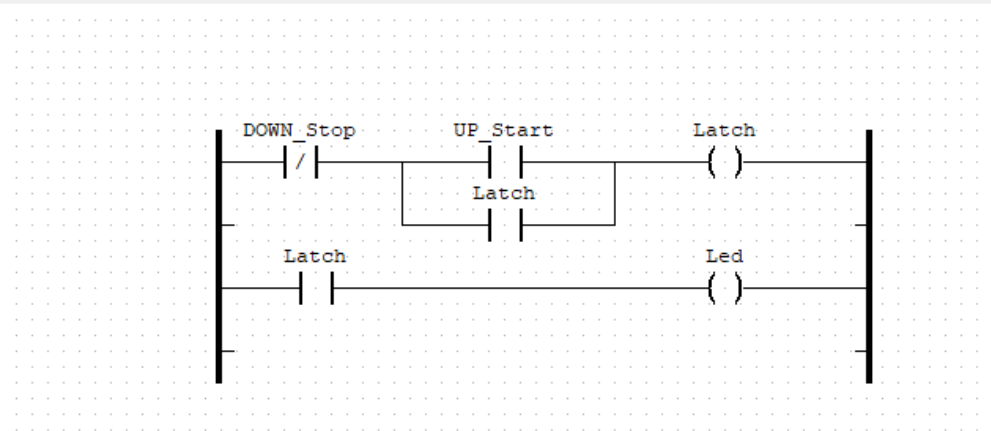


Figure 2: circuit setup of Arduino

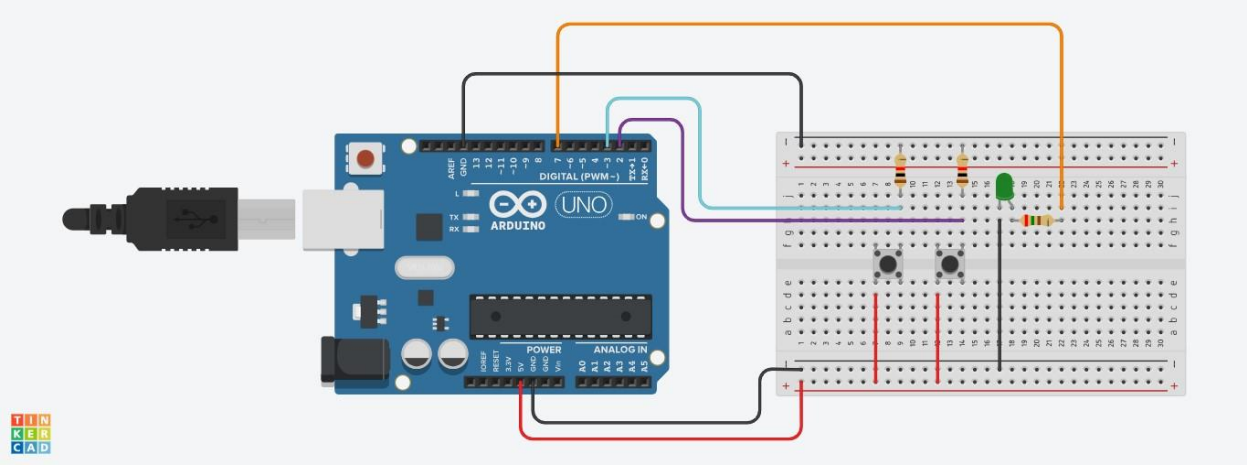
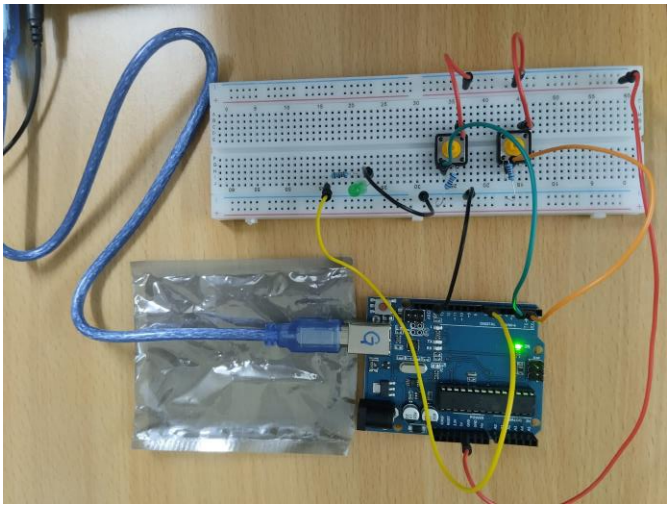


Figure 3: physical circuit setup of Arduino

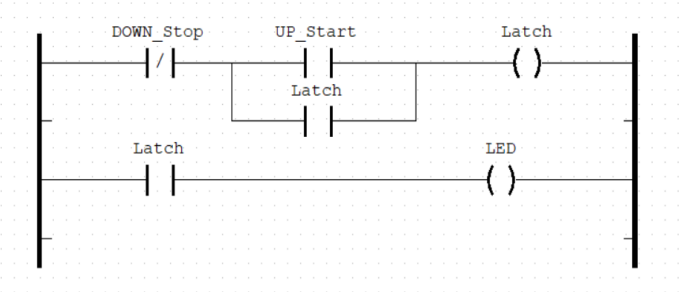


METHODOLOGY

1. The ladder diagram is created using OpenPLC Editor software as shown in Fig. 1.
2. All variables used in the ladder diagram is specified.
3. The ladder diagram is compiled and simulated in OpenPLC Editor.
4. The correct COM port number of Arduino is selected and all pin association between the OpenPLC variables and Arduino board is ensured to be correct.
6. The Arduino circuit is built as shown in Fig. 2.
7. The ladder diagram is uploaded to the Arduino board.
8. The functionality is tested.

DISCUSSION

In this experiment, we interfaced Programmable Logic Controllers (PLC) with microconreollers using OpenPLC and Arduino platform. We were able to create, simulate and upload ladder diagrams onto an Arduino board.



Down_stop is normally closed and the Up_start is normally open. Hence, if the button is not been pushed, the current will flow until it reaches to junction that have Up_start and latch. After the current flow through the Up_start, will then trigger the latch output and input. However when the latch is 1(true), it will act as a latch to keep the current to flow, and make both Latch inputs high. The Latch input will then send the high signal to the LED output, so that it will be turned on.

Besides, when down_stop is pressed, the switch cut the circuit and the current flow will break. Hence everything will be zero and turning off all the inputs and outputs. The LED will turn off.

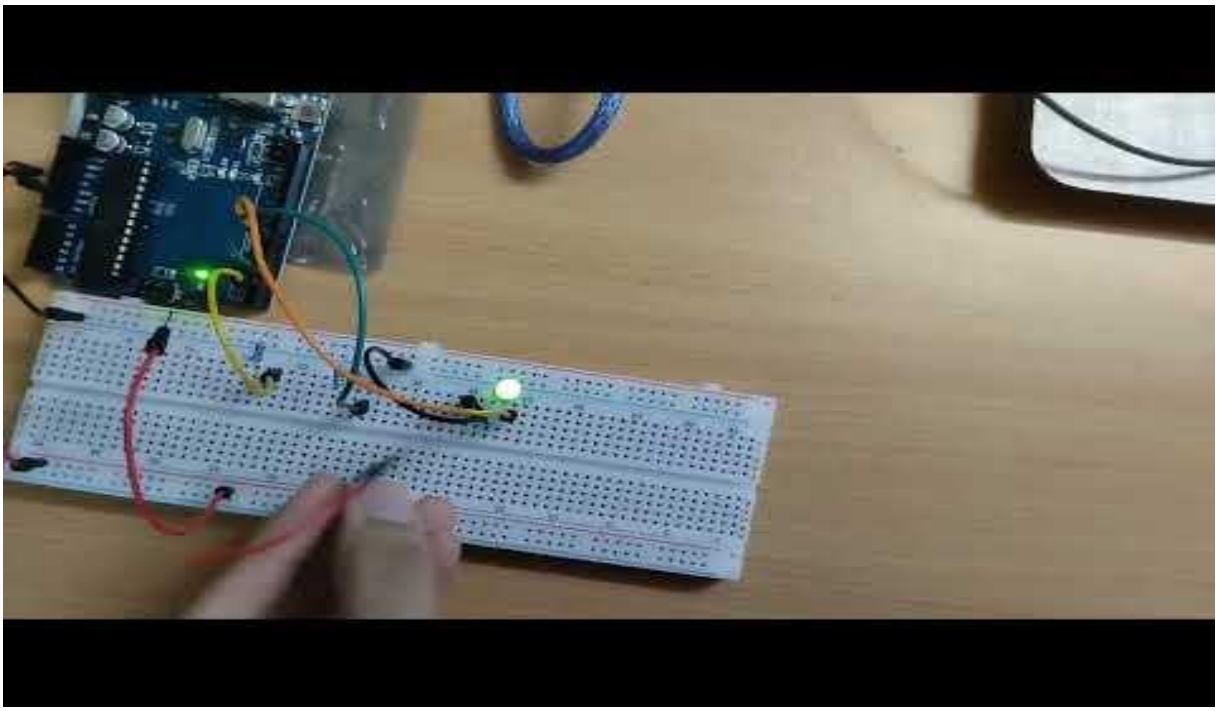
RESULTS

The experiment generated a correctly built start and stop control circuit. When the start button was pressed, the LED lit up even after the button was released, and the latch functioned as if the button had been pressed. By de-energizing the relay coil, pressing the stop button brought the LED to stop.

VIDEO

Using push buttons: https://youtube.com/shorts/zBFr95LkBc4?si=aqRYL1YZOc3QgpXA

Using jumpers as push buttons: https://youtu.be/mOwNMgEqSrc?si=upzMuq3gSy_XS5kf



CONCLUSION

Through this lab project, we have learned how to use two push buttons in order to turn on an LED lightbulb, using PLC to program the Arduino. With the aid of the opnenplc software, we have also learned the fundamentals of how the plc operates.

RECOMMENDATIONS:

This experiment can be improved in certain areas such as:

1) Signal Conditioning: To filter and condition analogue signals before they reach the PLC or Arduino, you should think about incorporating circuits for signal conditioning if your experiment uses such signals. Measurement precision and dependability may increase as a result.
2) Protection Circuitry: Provide protection against reverse polarity, overvoltage, and voltage spikes for the PLC and Arduino. Common protective components are fuses, diodes, and transient voltage suppressors. As our first attempt to use jumpers and push buttons we did not include resistors and the circuit did not work.
3) Colour coding the wiring, ensure that ground wires are black and red wires are red.
4) Use a newer and more efficient plc software.

REFERENCES

https://autonomylogic.com/docs/2-4-physical-addressing/

APPENDICES

ACKNOWLEDGEMENT

We would like to express our gratitude to Dr. Nadzril Bin Sulaiman, one of the lecturers of this course, mechatronic system integration 1 (MCTA 3203) for his guidance while the experiment is being carried out.

STUDENT'S DECLARATION

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 acknowledgement, and that the original work contained herein have not been untaken or done by unspecified sources or person.

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's contributor to the report.

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

Signature: *Ghousiah* Name: Noorul Ghousiah Binti Noordeen Sahib Matric Number: 2118298 Contribution: Procedure	Read Understand Agree	/ / /
Signature: *hanis* Name: hanis Matric Number: 2020590 Contribution: recommendation & conclusion	Read Understand Agree	/ / /
Signature: *Afiq* Name: Muhammad Afiq bin Ahmad Matric Number: 2115203 Contribution: result & discussion	Read Understand Agree	/ / /

Signature: *ayunaziera* Name: Nur Ayu Naziera Binti Rosli Matric Number: 2119202 Contribution: Abstract and Introduction	Read Understand Agree	/ / /
Signature: *Sarah* Name: Sarah Aishah binti Sa'aid Hazley Matric Number: 2117600 Contribution: Discussion	Read Understand Agree	/ / /