LAB REPORT 4: SOFTWARE AND HARDWARE ASPECTS OF PLC INTERFACING WITH MICROCONTROLLER

GROUP 1

MCTA 3203

SEMESTER 1 2024/2025

MECHATRONICS SYSTEM INTEGRATION

NO	NAME	MATRIC NO
1.	ABDUL A'LIM BIN MOHD RAJIB	2119687
2.	AHMAD HAZAMI BIN MOHD RAZIP	2211203
3.	ABDUL HADI BIN ZAWAWI	2210739
4.	AIN MAISARA BINTI ABDULLAH	2217856
5.	ADIBAH BINTI MOHD AZILAN	2212670

DATE OF SUBMISSION: 13/11/2024

ABSTRACT

This experiment focuses on the design and implementation of a start-stop control circuit using ladder diagram programming in OpenPLC Editor, integrated with an Arduino board for hardware deployment. The objective was to develop a functional ladder diagram to control the circuit, simulate the program for validation, and transfer the compiled program to the Arduino for real-time operation. The setup involved key components such as push buttons for starting and stopping, switches, jumper wires for connections, an LED for output indication, resistors for current limiting, and a breadboard for constructing the circuit.

The ladder logic was created and tested in the OpenPLC Editor software to ensure logical correctness before hardware integration. The compiled program was then transferred to the Arduino board, which facilitated the control of circuit operations based on the ladder logic design. The experiment successfully demonstrated the integration of software-based ladder diagrams with microcontroller-based hardware, highlighting practical applications of ladder logic in electronic control circuits. This approach showcases the potential for automated control systems using open-source tools and microcontroller platforms.

TABLE OF CONTENTS

1. INTRODUCTION

The development of automated control systems is an essential part of modern electronics and industrial automation. Ladder logic, a popular programming language in industrial control systems, allows for the design of control circuits in an intuitive, visual format. OpenPLC Editor, an open-source software, facilitates the creation and simulation of ladder diagrams, making it accessible for both educational and practical applications. This experiment aims to leverage OpenPLC Editor and an Arduino board to design and implement a start-stop control circuit, a fundamental configuration commonly used in industrial automation for managing machine operations.

The circuit comprises essential components including push buttons for initiating and stopping the process, switches for control, an LED for visual output, resistors for current regulation, jumper wires for connections, and a breadboard for constructing the physical setup. By designing the circuit in OpenPLC Editor, simulating it for logical validation, and transferring the compiled program to an Arduino board, this experiment bridges the gap between theoretical understanding and practical application.

This introduction lays the groundwork for exploring the practical implications of ladder logic and microcontroller integration, showcasing how software-based programming can effectively control real-world hardware components. The hands-on nature of this experiment provides insights into the process of developing control circuits, from conceptual design to hardware implementation, reinforcing the understanding of both electronic and programming fundamentals.

DESIGN AND IMPLEMENTATION OF A START-STOP CONTROL CIRCUIT USING LADDER DIAGRAMS IN OPEN PLC AND ARDUINO INTEGRATION

2. MATERIAL AND EQUIPMENT

- Arduino Board
- OpenPLC editor software
- 2 Push Button Switches
- Jumper Wires
- LED
- Resistors
- Breadboard

3. EXPERIMENTAL SETUP

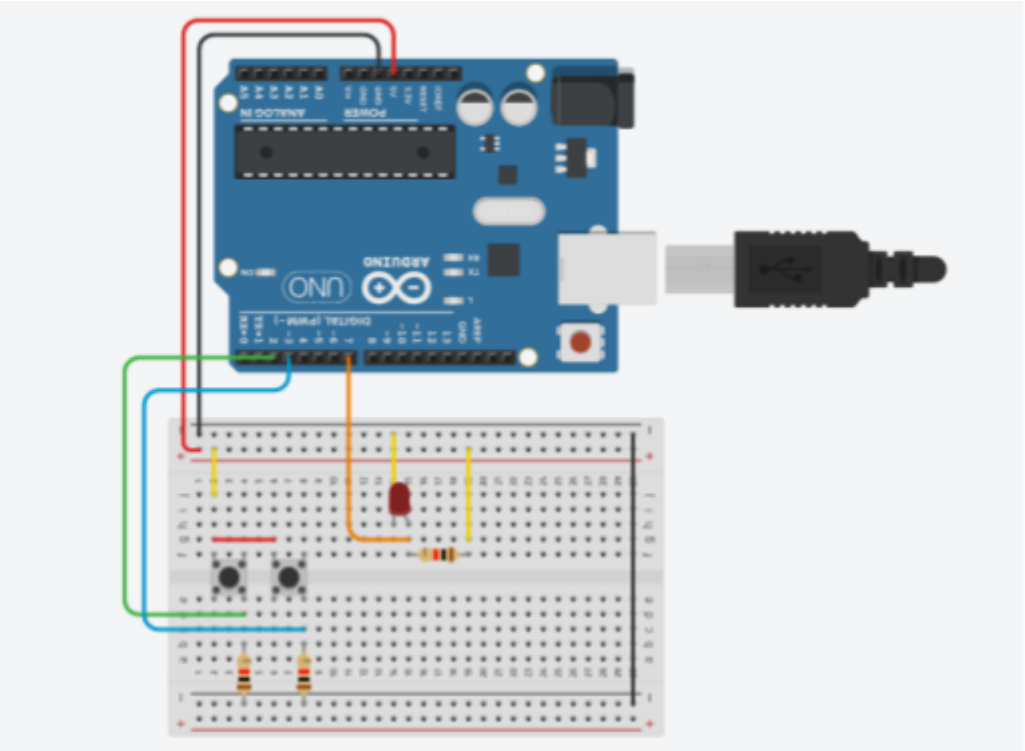


Figure 1: Arduino circuit diagram

4. METHODOLOGY

PROCEDURES:

a. Create the ladder diagram shown in Fig. 5.

b. Specify all variables used in the ladder diagram.

c. Compile and simulate the ladder diagram in OpenPLC Editor.

d. Upload the ladder diagram to the Arduino board.

e. Ensure to select correct COM port number and all pin association between the OpenPLC variables and Arduino board.

f. Build the circuit as shown in Fig. 6.

g. Test the functionality.

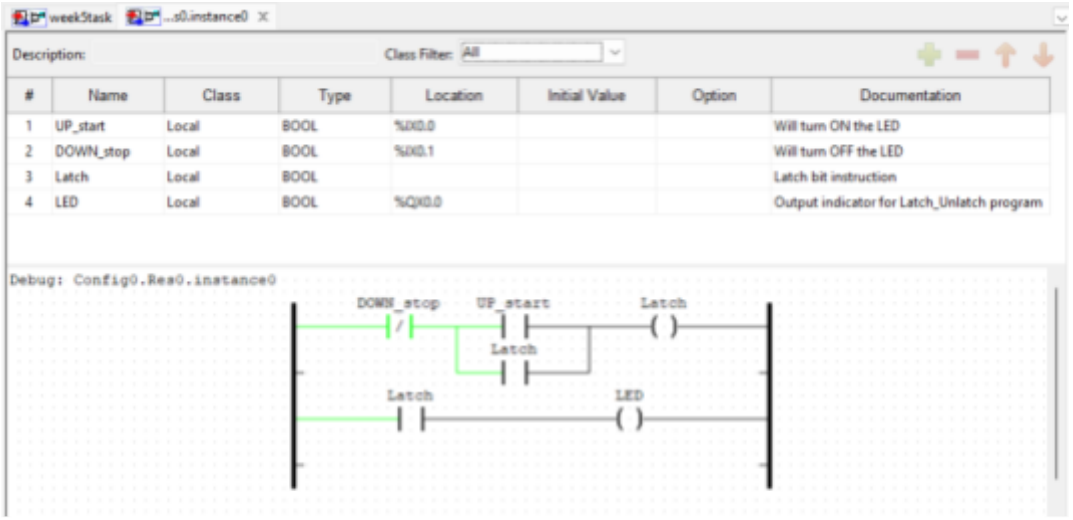


Figure 2: Ladder Diagram for the Start-Stop Control Circuit

5. DATA COLLECTION

PUSH BUTTON 1 (DOWN_stop)	PUSH BUTTON 2 (UP_start)	LED
False	False	Off
False	True	On
True	False	Off

Table 1

6. DATA ANALYSIS

The system successfully operated according to the ladder diagram logic, where the LED turned on when the UP_start button was pressed. The LED turned off when the DOWN_stop button was pressed. This confirms that the Start-Stop control mechanism was implemented correctly. The two push buttons served as inputs to the ladder diagram logic, and the state of the LED was the expected output in all conditions**.**

7. RESULTS

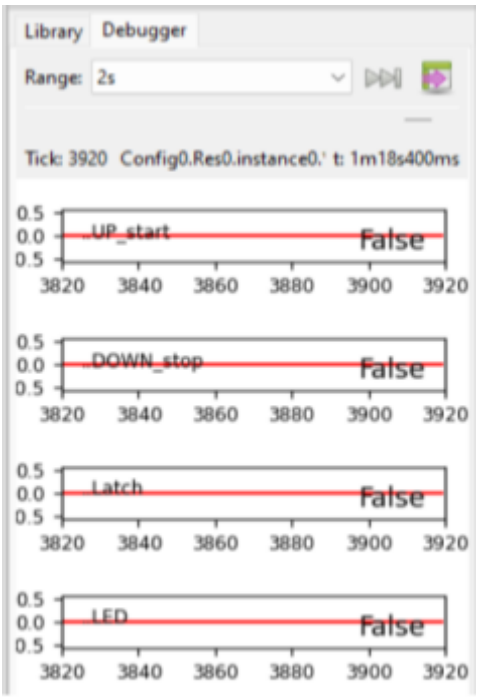


Figure 3: LED off

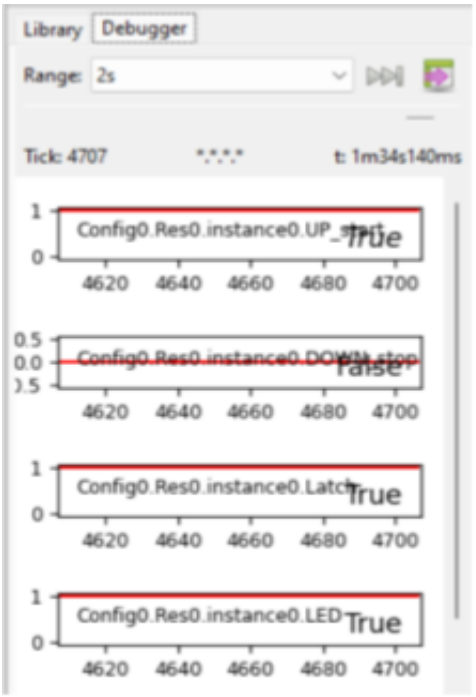


Figure 4: LED on

8. DISCUSSION

Using ladder logic programming in the OpenPLC Editor and an Arduino microcontroller, the experiment successfully showed how to design and implement a start-stop control circuit. Based on the data collected, we observed that the LED operated as intended; it turned on when the UP_start button was pressed and turned off when the DOWN_stop button was

activated. This behaviour aligns perfectly with the expected outcomes based on the designed ladder diagram logic, confirming that the Start-Stop control mechanism was effectively implemented.

Despite the successful operation, minor discrepancies such as potential sources of error, including wiring mistakes or inherent limitations of mechanical components, were still noted. Variability in wiring connections or incorrect pin assignments could lead to unexpected findings. Careful verification of connections is crucial in such setups to avoid any human error. Additionally, while OpenPLC Editor is effective for simulation, it may not account for all real-world conditions like electrical noise. Overall, the experiment met its primary objectives while highlighting areas for improvement in future iterations.

9. CONCLUSION

In a nutshell, this activity confirmed that ladder logic can effectively control hardware components through microcontrollers, as evidenced by the successful operation of the start-stop control circuit. We illustrated the integration of ladder logic programming with microcontroller hardware through the development of a start-stop control circuit using OpenPLC Editor and an Arduino board. The LED responded correctly to the UP_start and DOWN_stop buttons, supporting the initial hypothesis that software-based programming can translate into effective hardware control. These outcomes emphasise the importance of integrating theoretical knowledge with practical experience in automation technology. As automation becomes increasingly prevalent in manufacturing and other sectors, understanding how to design and implement control circuits using accessible tools like OpenPLC Editor will empower engineers and technicians to create efficient and reliable systems. Furthermore, this experiment emphasizes the importance of bridging theoretical knowledge with practical skills, preparing students and professionals for real-world

challenges in modern technology. Future work could explore more complex control systems or incorporate additional sensors and actuators to further enhance system capabilities and applications in various industrial contexts.

10. RECOMMENDATION

For this Start-Stop control experiment, we can consider adding debounce logic for push buttons to minimize signal noise and enhance reliability. Next, we can extend the control circuit with timed delays or extra indications and introduce more advanced logic principles. We can also include safety devices, such as an emergency stop, which would highlight the significance of workplace safety. To avoid problems, detailed documentation of variable functions is required, as is precise mapping between ladder logic and Arduino pins. This experiment effectively teaches students about PLC-microcontroller interfacing and ladder logic, which are both useful for understanding industrial control systems and prepares them for real-world applications.

11. REFERENCES

1. Wilcher, D. (2022, August 30). *PLC ladder logic on an Arduino: Introduction to OpenPLC*. Control.com. https://control.com/technical-articles/plc-ladder-logic-on-an-arduino-introduction-to-openplc/
2. *2.4 Physical Addressing – Autonomy*. (n.d.). https://autonomylogic.com/docs/2-4-physical-addressing/

APPENDICES

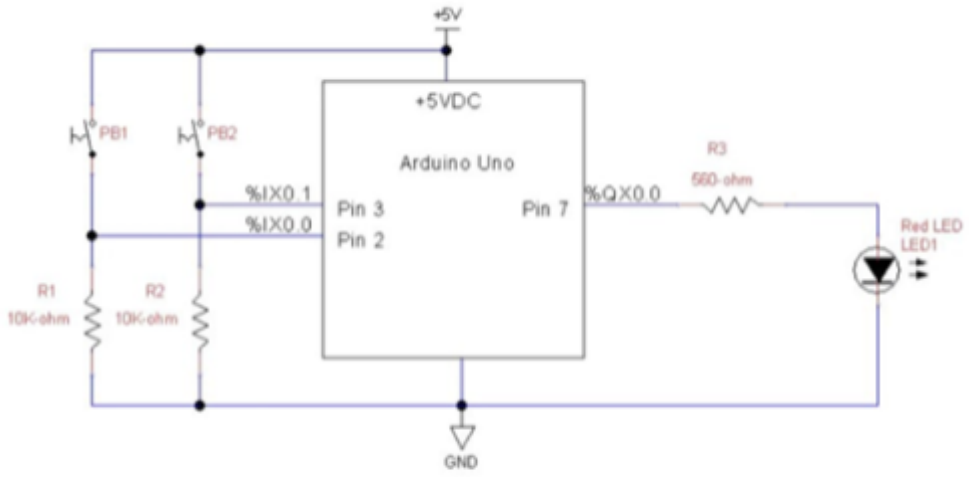


Figure 3: Start-Stop Control Circuit

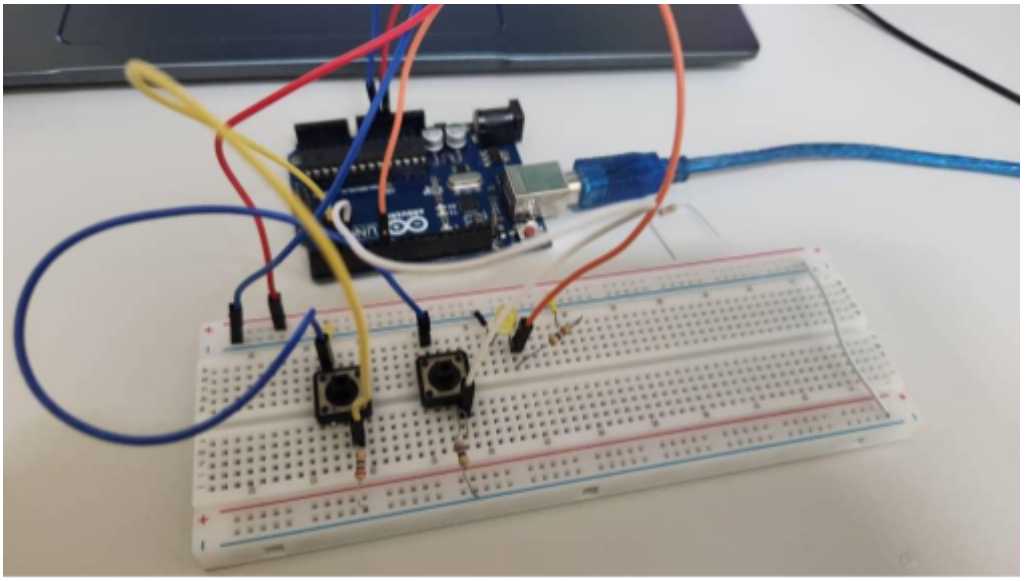
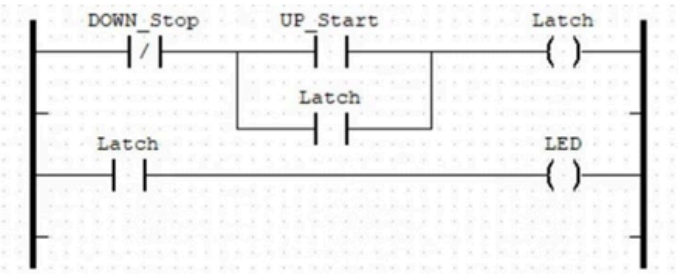


Figure 4: Arduino circuit



#	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. 5: Ladder Diagram for the Start-Stop Control Circuit

task5.ino

```
#define UPSTART_PIN 2      // Pin for the Upstart button (NO)
#define DOWNSTOP_PIN 3     // Pin for the Downstop button (NC)
#define LED_PIN 7        // Pin for the LED

bool latch = false;              // Latch state for holding LED on
bool lastUpstartState = LOW;     // Last state of the Upstart button
bool lastDownstopState = HIGH;   // Last state of the Downstop button (NC)

void setup() {
  pinMode(UPSTART_PIN, INPUT_PULLUP);  // Set Upstart button as input (NO button)
  pinMode(DOWNSTOP_PIN, INPUT_PULLUP); // Set Downstop button as input (NC button)
  pinMode(LED_PIN, OUTPUT);           // Set LED pin as output

  Serial.begin(9600);  // For debugging
  Serial.println("Start/Stop Control System");
}

void loop() {
  // Read the current state of the Upstart button (active LOW)
  bool upstartState = digitalRead(UPSTART_PIN) == LOW;
  // Read the current state of the Downstop button (active HIGH)
  bool downstopState = digitalRead(DOWNSTOP_PIN) == HIGH;

  // Detect Upstart button press (turn LED ON and latch state)
  if (upstartState && !lastUpstartState) {
    latch = true;                 // Activate latch to keep LED ON
    digitalWrite(LED_PIN, HIGH);  // Turn the LED ON
    Serial.println("LED ON (Upstart pressed)");
  }

  // Detect Downstop button press (turn LED OFF)
  if (!downstopState && lastDownstopState) {
    latch = false;                // Reset latch to turn LED OFF
    digitalWrite(LED_PIN, LOW);   // Turn the LED OFF
    Serial.println("LED OFF (Downstop pressed)");
  }


  // Maintain LED ON if latch is true
  if (latch && !upstartState) {
    digitalWrite(LED_PIN, HIGH);  // Keep LED ON (even if Upstart button is released)
    Serial.println("LED ON (Holding state)");
  }

  // Update the last button states for next loop iteration
  lastUpstartState = upstartState;
  lastDownstopState = downstopState;

  delay(50);  // Small delay to debounce the buttons
}

```

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to our lecturers, Dr. Wahju Sediono and Dr. Zulkifli bin Zainal Abidin for their invaluable guidance in this experiment. Their expertise in the mechatronics field and encouragement have been a big support in the successful integration of our system. Not forgetting our teaching assistants in the lab for helping us indirectly.

To add, we also wish to extend a special thank you to our fellow team members for their collaboration and hard work during the course of this project. Their contributions were pivotal in driving our efforts forward and enhancing the overall quality of our work.

STUDENTS' 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 acknowledgement, and that the original work contained herein have 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 that no further improvement on the reports is needed from any of the individual contributors 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: **A'LIM**	**Read**	/
Name: ABDUL A'LIM BIN MOHD RAJIB	**Understand**	/
Matric No: 2119687	**Agree**	/

Signature: hazami	**Read**	/
Name: AHMAD HAZAMI BIN MOHD RAZIP	**Understand**	/
Matric No: 2211203	**Agree**	/

Signature: 	**Read**	/
Name: ABDUL HADI BIN ZAWAWI	**Understand**	/
Matric No: 2210739	**Agree**	/

Signature: 	**Read**	/
Name: AIN MAISARA BT. ABDULLAH	**Understand**	/
Matric No: 2217856	**Agree**	/

Signature: 	**Read**	/
Name: ADIBAH BINTI MOHD AZILAN	**Understand**	/
Matric No: 2212670	**Agree**	/