DEPARTMENT OF MECHATRONICS ENGINEERING

Mechatronics System Integration (MCTA3203)

WEEK 10: Systems integration (Microcontroller, PLC and Computer Systems)

Section 1

(Group E)

Adam Mikhail Bin Khairul Effendy	2117613
Muhammad Afiq bin Mazhalimi	2110491
Muhammad Fahmi Ikhwan bin Ahmad Suparjo	2111167
Eiman Bin Azzam	2110565
Zulhilmi bin Abdul Rahman	2117679

Date of submission: 10/1/2024
Prepared for: Dr. Nadzril bin Sulaiman

Abstract

This experiment focuses on creating a cohesive industrial automation system by integrating Programmable Logic Controllers (PLC), microcontrollers, sensors, and computers through the widely adopted Modbus communication protocol. Modbus serves as the linchpin for enabling seamless communication between devices within the same network, typically in a master-slave configuration. The experiment entails configuring sensors to collect data, connecting them to a microcontroller programmed for data interpretation and processing, and subsequently utilizing Modbus to facilitate communication between the microcontroller and a computer system. The computer, equipped with specialized software, receives and analyzes data from the sensors through the microcontroller, allowing for real-time monitoring and control. This integrated setup showcases practical implementation, demonstrating the effectiveness of Modbus in enabling efficient data exchange and control among diverse components in an industrial automation context.

Table of Contents

1.1 Introduction

This experiment aims to demonstrate the integration of a Microcontroller Processing Unit (MPU) with a range of sensors including a touch sensor, LM35 temperature sensor, and LDR (Light Dependent Resistor) sensor. The objective is to showcase the functionality and versatility of the MPU in interpreting data from these diverse sensors and facilitating communication with a computer system using the Modbus communication protocol. By harnessing the MPU's capabilities, this experiment emphasizes the practical implementation of a comprehensive system architecture that enables efficient data collection, processing, and transmission between sensors and computer systems, highlighting the significance of standardized communication protocols in industrial automation and control systems.

Objectives:

- To showcase the functionality and versatility of the MPU in interpreting data from different sensors.
- To learn how different devices can communicate using the Modbus protocol

1.2 Materials and Equipment

1. TTP223 Capacitive Touch Sensor
2. IR sensor
3. LDR (Light Dependent Resistor)
4. Raspberry Pi
5. Arduino Uno Board
6. Resistors
7. LED

1.3 Experimental Setup

Hardware Setup

- Raspberry Pi
  - Configured as Modbus master.
  - Communicates with Arduino (Modbus slave) to read sensors' data.
- Arduino
  - Configured as slave.
  - Read data from sensors.

Software Setup

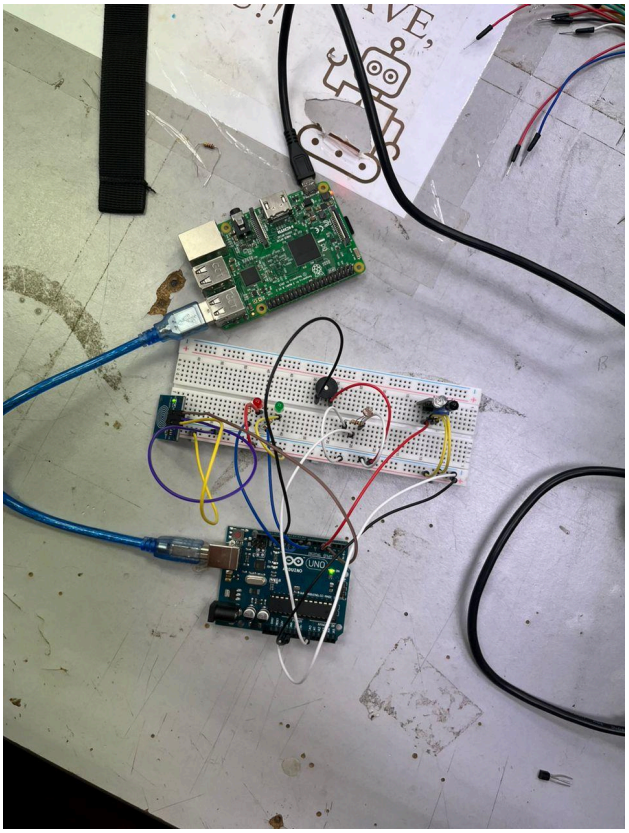
- OpenPLC
  - Configured as slave.
  - Received data from Raspberry Pi and controlled the output based on the received data.
  - Used to control the output based on the sensors' data received from Arduino.
- Modbus
  - Radzio

1.4 Methodology

1. Selected sensors (LDR, IR sensor, TTP223), and chose Modbus for communication.
2. Connected sensors to Arduino and established Arduino-Raspberry Pi communication.
3. Programmed Arduino to read sensor data and implemented Modbus communication for data transmission to Raspberry Pi.
4. Developed a ladder diagram using graphical symbols to depict logical control sequences for operating and managing the system components (sensors, Arduino, Raspberry Pi) based on input conditions and defined actions.
5. Installed the system in the lab, validated data accuracy, and confirmed real-time display using OpenPLC Runtime

1.5 Results

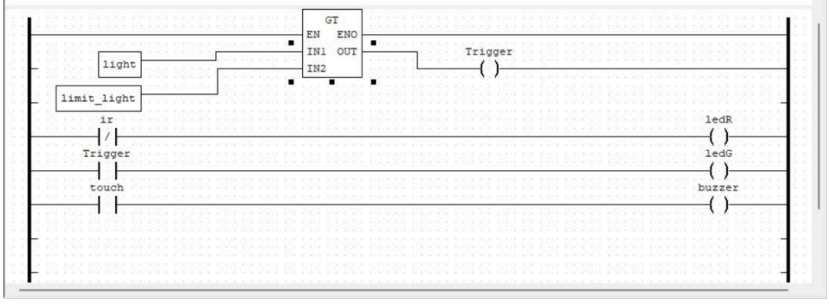
We successfully developed a system that consists of an LDR, IR sensor, TTP223 Capacitive Touch Sensor, a Raspberry Pi and an Arduino using Modbus as communication protocol. We also were able to monitor all the sensor's data in realtime using OpenPLC Runtime.



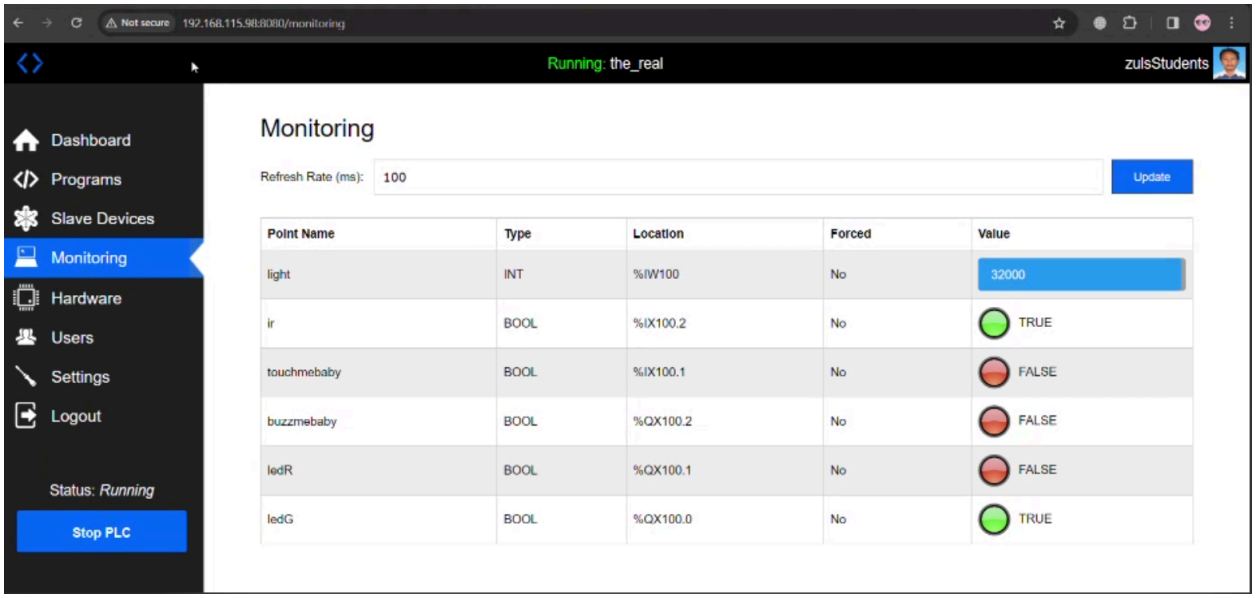
Hardware setup

Description: Class Filter: All

#	Name	Class	Type	Location	Initial Value	Option	Documentation
2	Trigger	Local	BOOL				
3	ir	Local	BOOL	%IX100.2			
4	limit_light	Local	INT		20000		
5	touch	Local	BOOL	%IX100.1			
6	limit_temp	Local	INT		100		
7	buzzer	Local	BOOL	%QX100.2			
8	ledR	Local	BOOL	%QX100.1			
9	ledG	Local	BOOL	%QX100.0			



Ladder logic programming



Realtime data monitoring using OpenPLC Runtime

1.6 Discussion

1. The red LED lights light up when the IR sensor gives a false value.
2. The buzzer turns on when the touch sensor gives a true value.
3. The green LED turns off when the LDR value is below 20000, otherwise, the green LED stays on

1.7 Conclusions

In conclusion, we learned that this experiment highlights the integration of Modbus protocol as an essential component in putting the Industrial Internet of Things (IIoT) at the forefront of industrial plants. This is also a way to realise the 4th industrial revolution. Implementing IIoT provides a lot of benefits in managing plants, it can improve the plant's efficiency, productivity, and safety besides also allowing remote monitoring and possibly remote control of field sites.

1.8 Recommendation

To enhance our system, it's advisable to explore alternative communication methods beyond Modbus for potential improvements in data transfer efficiency. Additionally, fine-tuning sensor calibration could notably enhance the accuracy of collected data. Implementing backup protocols to address sensor failures or communication disruptions is crucial for ensuring continuous operation. Improving the user interface of OpenPLC Runtime or integrating supplementary visualization tools could enhance user interaction and facilitate better monitoring

of real-time sensor data. Strengthening security measures to safeguard data transmission within our setup and considering scalability options for future sensor integrations are important. Investigating energy-saving techniques and ensuring compatibility with other systems could further optimize our system's functionality and applicability. Regular maintenance and updates will be essential to keep the system operating at its best.

1.9 References

1. https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-andsupervisory-system/the-modbus-protocol-in-depth.html
2. https://funprojects.blog/2021/11/18/openplc-on-a-raspberry-pi/
3. OpenPLC Runtime, https://autonomylogic.com/docs/installing-openplc-runtime-onwindows/
4. Raspberry pi with OpenPLC and modbus, https://www.youtube.com/watch?v=iC5s4CEiOB4
5. https://www.youtube.com/watch?v=dQ8GcfpGDuA
6. Aduino with OpenPLC, https://www.youtube.com/watch?v=d39Al4w1sL0
7.https://control.com/technical-articles/how-to-building-a-plc-project-with-modbuscommunication/

1.10 Acknowledgements

We would like to acknowledge our instructor, Dr. Nadzril bin Sulaiman for your expert guidance, encouragement, and unwavering support throughout the project and also for your invaluable support and provision of resources crucial to the successful execution of the experiment on Systems integration (Microcontroller, PLC and Computer Systems). We hope to learn more from such a talented instructor if the opportunity arises.

We would also like to thank Allah S.W.T. for guiding us in successfully demonstrating this experiment.

1.11 Students'Declaration

Declaration:

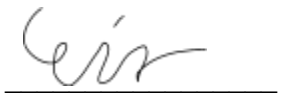
We certify that this project/assignment is entirely our own work, except where we have given fully documented references to the work of others, and that the material in this assignment has not previously been submitted for assessment in any formal course of study.

_adam_____ Date: 10/1/2024

Name: Adam Mikhail Bin Khairul Effendy

fahmi_____

Name: Muhammad Fahmi Ikhwan Bin Ahmad Suparjo



Name: Eiman Bin Azzam

afiq

Name: Muhammad Afiq bin Mazhalimi

_Zul_____

Name: Zulhilmi bin Abdul Rahman