DEPARTMENT OF MECHATRONICS ENGINEERING

Mechatronics System Integration (MCTA3203)

WEEK 6: DAQ interfacing with Microcontrollers

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: 22/11/2023

Prepared for: Dr. Nadzril bin Sulaiman

Abstract

This project explores interfacing Data Acquisition (DAQ) with Arduino Uno using PLX-DAQ software, integrating Light Dependent Resistor (LDR) and LM35 temperature sensor modules. The goal is to create a versatile data acquisition system for real-time monitoring of light intensity and temperature. The Arduino Uno reads analog data from the LDR and LM35, transmitting it to a computer via PLX-DAQ for logging and visualization in Excel. Calibration, sensor integration, and programming the Arduino for data collection were key steps. This project showcases a functional DAQ system, illustrating the potential for applications in environmental monitoring and automation while demonstrating the adaptability of Arduino-based setups for data analysis and acquisition.

Table of Contents

1.1 Introduction

Data Acquisition (DAQ) involves a measurement setup that requires sensors to measure a physical value and change into electrical signals, a DAQ device that serves as connection between the sensors and computer by converting analog signals to digital, and a computer to analyse data received. In this experiment, we will use Arduino as a DAQ Hardware and analyse potentiometer data and present it in an Excel Sheet using a software add-in , PLX-DAQ.

Objectives:

1. To use Arduino as DAQ hardware to collect data from a potentiometer and analyse using a computer utilising the PLX-DAQ software add-in.
2. To understand the utility of data acquisition and analysis by plotting graphs of data.

1.2 Materials & Equipments

1. Arduino board
2. PLX-DAQ Software
3. LDR
4. LM35
5. Resistors
6. Potentiometer
7. Jumpers
8. Breadboard
9.

1.3 Experimental setup

Hardware Setup:

1. Connect VCC and GND to the left and right pins of potentiometer.
2. Connect pin A1 to the middle pin of the potentiometer. (Figure 6.1)

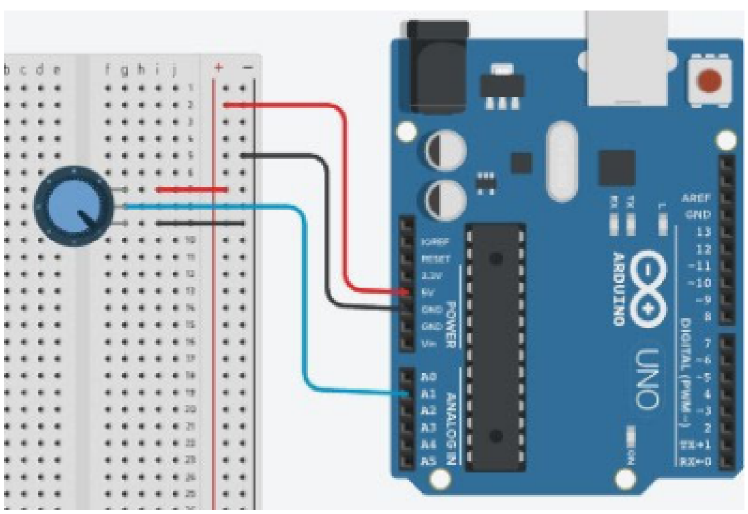


Figure 6.1: Circuit Setup Arduino to Potentiometer

Software Setup:

1. Launch the Arduino IDE and verify the example code (Figure 6.2)
2. Download and install PLX-DAQ from the internet.
3. Launch the PLX-DAQ Spreadsheet with the pop-out GUI window (Figure 6.3)
4. Select the correct COM port number and ensure the baud rate is the same as in code, 9600.
5. Press connect tab and observe received data. Use tools such as graphs in MS Excel for analysis. (Figure 6.4).

Figure 6.2: Example code for potentiometer to PLX-DAQ

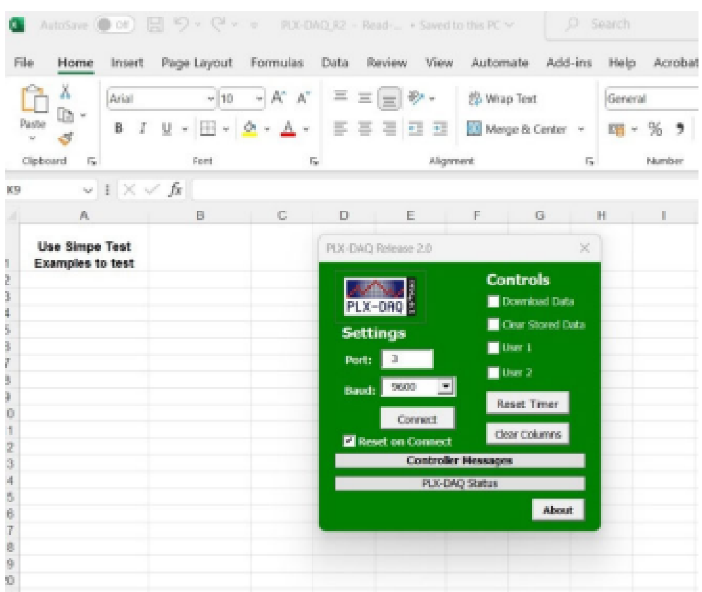


Figure 6.3: PLX-DAQ Spreadsheet with pop-out GUI window

Time	Started Time	Register value
13:57:42	0.01	1022
13:57:44	1.46	1021
13:57:45	2.95	1022
13:57:47	4.46	654
13:57:48	5.95	80
13:57:50	7.46	0
13:57:51	8.96	129
13:57:53	10.47	296
13:57:54	11.96	389
13:57:56	13.46	441
13:57:57	14.96	548
13:57:59	16.46	509
13:58:00	17.97	134
13:58:02	19.46	0
13:58:03	20.96	314
13:58:05	22.47	536
13:58:06	23.97	987
13:58:08	25.47	803
13:58:09	26.96	539
13:58:11	28.48	666
13:58:12	29.98	1022
13:58:14	31.47	789
13:58:15	32.97	771
13:58:17	34.48	480
13:58:18	35.98	699
13:58:20	37.48	883
13:58:21	38.97	880

Figure 6.4: Potentiometer data on the PLX-DAQ Spreadsheet

Once the PLX-DAQ Spreadsheet is successfully interfaced , proceed to collect data from the sensors LM35 and LDR by constructing the circuit below.

1. Connect Vout of the LM35 to analog pin A0. (Figure 6.5)
2. Connect the negative terminal of the LDR connected with the resistor to analog pin A1.

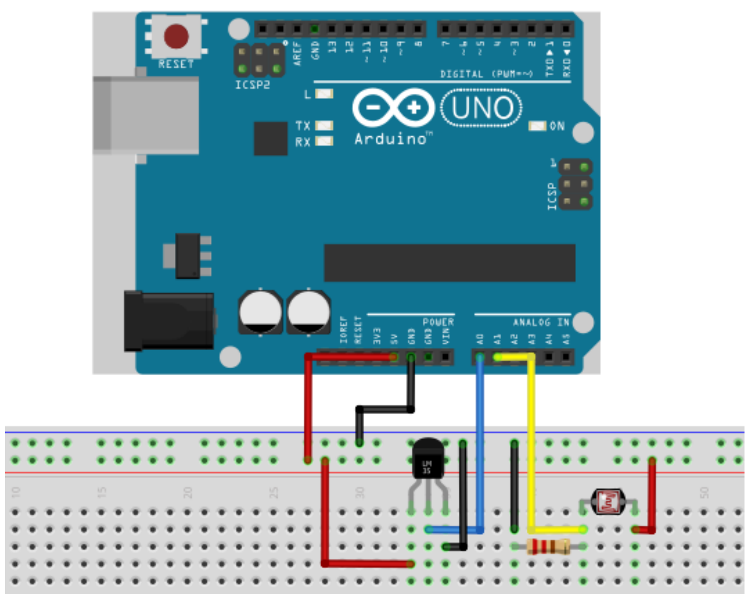


Figure 6.5: Sensor and LDR to Arduino UNO Circuit Setup

1.4 Methodology

1. After constructing the circuit, we launched Arduino IDE and we wrote code that allows Arduino to read analog signals from the LM35 and LDR and convert it to digital.

The picture below shows our code:

sketch_nov15a.ino

```
const int meter = A1;

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600); // start serial monitor with baud rate of 9600
  Serial.println("CLEARDATA"); // clear column a to j from row 2 onwards (labels in row 1 remain )
  Serial.println("LABEL,Time,Started Time,Register Value"); // labels at row 1 as Time,Started Time,Register Value
  Serial.println("RESETTIMER"); // resets the timer

void loop() {
  // put your main code here, to run repeatedly:
  int sensorValue = analogRead(meter); // reads the potentiometer values
  Serial.print("DATA,TIME,TIMER,"); // stores data TIME,TIMER and prints it in column A,B
  Serial.println(sensorValue); // prints potentiometer value at column C (under Register Value label)
  delay("1500"); // print reading every 1.5 seconds
}
```

3. We verified the codes and uploaded it to the Arduino board.
4. We launched the PLX-DAQ spreadsheet. Ensured the correct com port is selected and then generated the output from the sensors in the spreadsheet.

1.5 Data collection

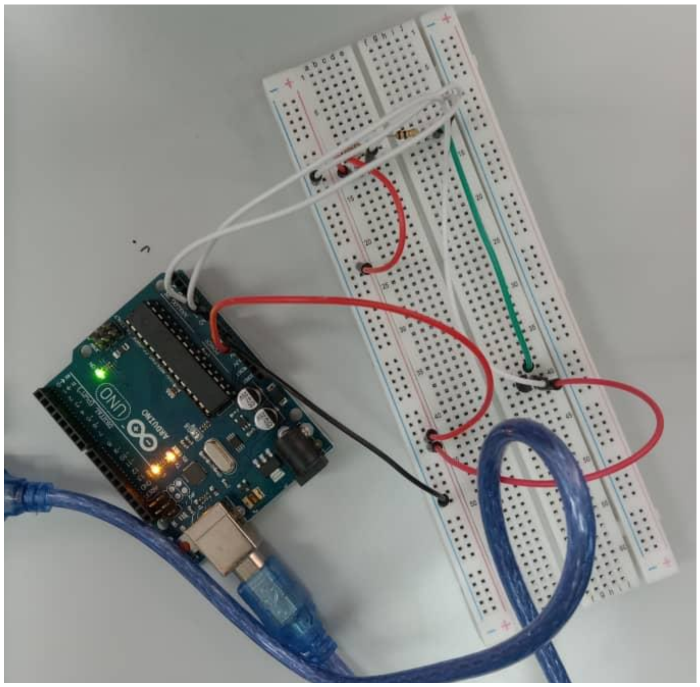
- No data collection

1.6 Data analysis

- No data analysis

1.7 Results

- We managed to plot the output of temperature and light intensity based on reading of LM35 and LDR.
- The result can be seen in demonstration video at link below with title "DAQ-demo-vid" in our github respiratory.



Link Video (refer to GitHub "DAQ-demo-vid"):
https://github.com/EimanAzzam/Group-E-Mecha-Integration/blob/main/Week6/DAQ-demo-vid.mp4

1.8 Discussion

- To test the functionality of sensor,we touch the LQM sensor and LDR sensor to varies the reading of sensor.
- Based on the video in the result,when the LQM sensor being touched,the sensor reading increase.While when LDR sensor being touched,the sensor reading will decrease.

1.9 Conclusion

- Based on the experiment, all the objectives were achieved. We learned how to interface the Arduino board using Excel software PLX-DAQ to read the data shown by potentiometer and sensors. We were also able to produce graphs of the data , and interpreted the conditions of when the signals are sending low or high measurements.

1.10 Recommendation

- Ensure to do the experiment at a suitable venue when using the sensor. This will help to get an exact value to read the data. Always recheck the circuit to prevent any errors when conducting the experiment. Improve the coding in arduino if the data are not reasonable. Make sure the table contents at the DAQ software are tally with the data shown. If there are any issues, check the COM port settings in both PLX-DAQ and Arduino sketch. Also, ensure that the baud rate matches.

1.11 References

1. AUTODESK Instructables - Arduino Temperature Sensor Using LM35
https://www.instructables.com/Arduino-Temperature-Sensor-Using-LM35/
2. Electronic Wings - LM35 Interfacing with Arduino UNO
https://www.electronicwings.com/arduino/lm35-interfacing-with-arduino-uno

1.12 Appendices

Refer to GitHub for coding and video

1.13 Acknowledgments

We would like to acknowledge our instructor, Dr. Nadzril Bin Sulaiman for teaching us how to interface the Arduino board by using PLX-DAQ software and answering all our tedious questions due to the difficulties we faced during the experiment. We hope to learn more in the future from such a talented instructor if the opportunity arises.

We would also like to give thanks to Allah S.W.T for giving us guidance in demonstrating this experiment successfully.

1.14 Student's 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: 22/11/2023

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