LABORATORY REPORT

MECHATRONICS SYSTEM INTEGRATION

MCTA 3203

SEMESTER 2 2023/2024

WEEK

LAB: SYSTEMS INTEGRATION

SUBMISSION DATE:

GROUP: 6

	NAME	MATRIC NUMBER
1.	MUHAMMAD AZIM ZUFAYRI BIN NORAZAM	2211973
2.	MUHAMMAD AMMAR FARIS BIN ARAFAS	2212645
3.	AHMAD ADAM BIN NORHELME	2219973
4.	AHMAD HAFIZULLAH BIN IBRAHIM	2216185

TABLE OF CONTENTS

ABSTRACT

This experiment explores the implementation of a low-pass filter (LPF) using an Arduino microcontroller to reduce high-frequency noise in sensor data. By connecting a sensor to the Arduino's analog input and employing a software-based LPF, the study demonstrates the effectiveness of filtering in enhancing signal quality. The results indicate that the LPF significantly smoothens the sensor signal, effectively mitigating high-frequency noise. This report discusses the methodology, results, and implications of LPF in signal processing applications, concluding with recommendations for further improvement.

INTRODUCTION

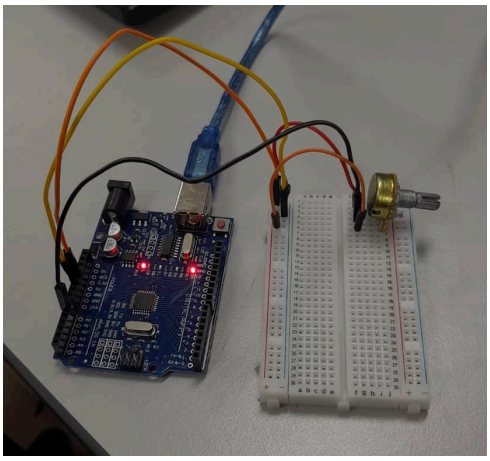
Maintaining data integrity is important in data acquisition and signal processing, especially in environments with significant high-frequency noise. Low-pass filters (LPF) are essential in such scenarios as they allow low-frequency signals to pass while attenuating high-frequency components. This experiment aims to demonstrate the practical application of an LPF using an Arduino microcontroller. By implementing a software-based LPF, we aim to showcase how signal quality can be improved in real-time data acquisition systems. The study's significance lies in its potential applications across various domains, including control systems, medical devices, and environmental monitoring.

MATERIALS AND EQUIPMENT

- Arduino Uno
- Breadboard
- Analog Input (Potentiometer)
- Jumper Wires (M2M)

METHODOLOGY

1. Connect the sensor to the analog input of the Arduino through LPF using the microcontroller.
2. Write Arduino code to read analog data from the sensor and implement a low-pass filter.
3. Upload the code to the Arduino board.
4. Monitor the output using serial monitor in the IDE.
5. Observe how the LPF smoothen the signal and reduces high-frequency noise.

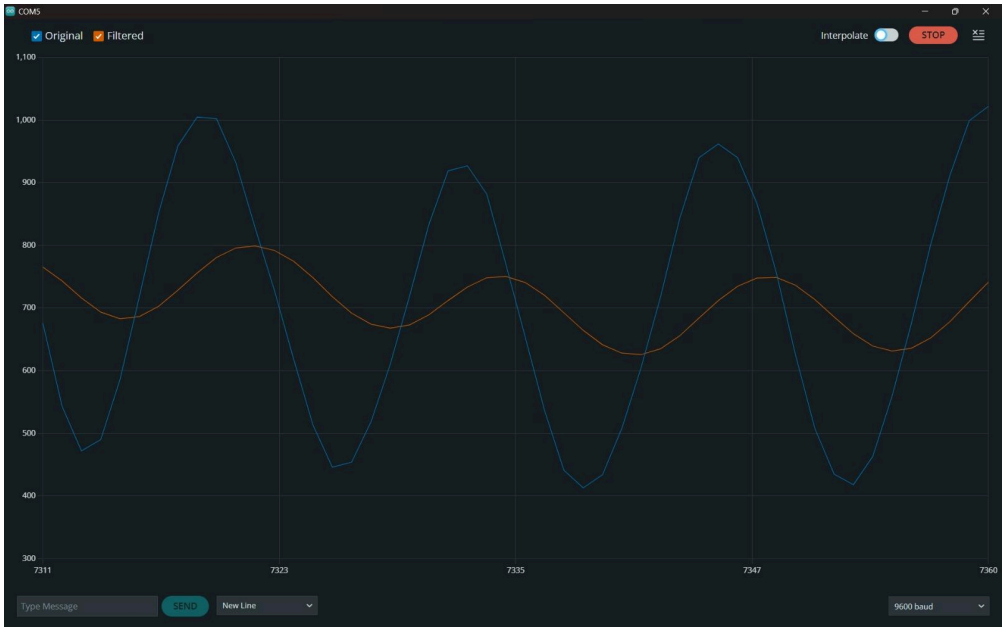


Arduino Code

```
const int analogPin = A0;
int sensorValue;
float filteredValue = 0.0;
float alpha = 0.1; // Adjust this value for filter strength

void setup() {
  Serial.begin(9600);
}

void loop() {
  // Read analog data from the sensor
  sensorValue = analogRead(analogPin);

  // Apply low-pass filter
  filteredValue = alpha * sensorValue + (1 - alpha) * filteredValue;

  // Print the results
  Serial.print("Original:");
  Serial.print(sensorValue);
 // Serial.print(",");
  Serial.print("\tFiltered:");
```

```
  Serial.println(filteredValue);

  delay(100); // Adjust the delay based on your sampling rate
}
```

RESULTS

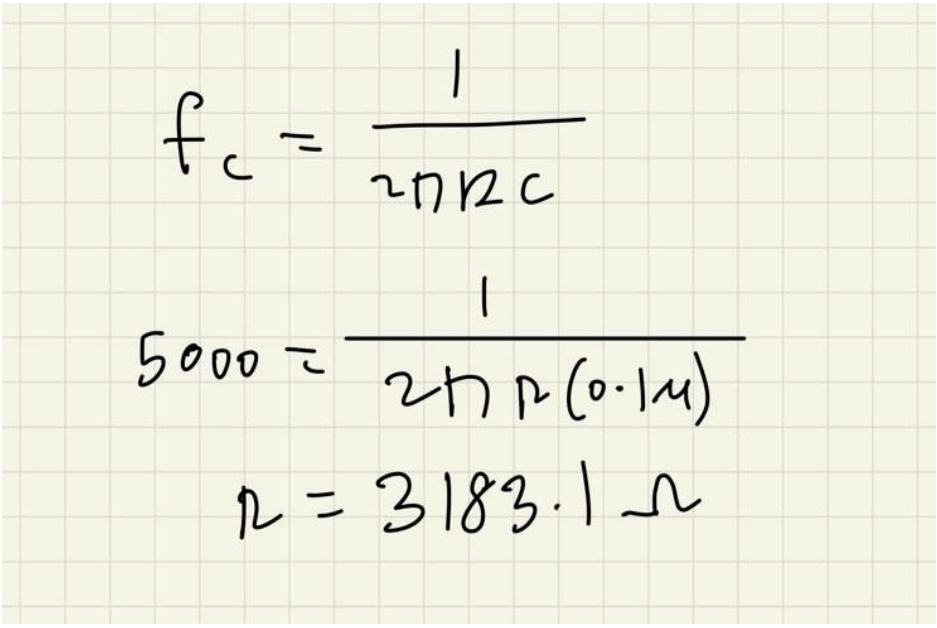


DISCUSSION

Low-pass filtering is crucial in applications where high-frequency noise can disturb the data accuracy. LPF improves signal quality, enhances accuracy, and is widely used in data acquisition, signal processing, and control systems. However, it can introduce latency, be ineffective against certain noise, and require careful component selection and circuit design.

QUESTION

PART A
CALCULATION



PART B
Discuss the principle of low-pass filter and its application. Consider the real-world application of digital filtering in areas like audio processing or sensor data filtering.

- A low-pass filter is a circuit that only passes signals below its cutoff frequency while attenuating all signals above it.
- The principle of low-pass filtering is based on frequency selectivity, and its applications are diverse and widespread.
- Application; LPFs are used in data communication systems to limit the bandwidth of transmitted signals, reducing interference and improving signal quality

CONCLUSION

The experiment successfully demonstrated the effectiveness of a software-based low-pass filter in reducing high-frequency noise from sensor data using an Arduino microcontroller. The LPF significantly smoothened the signal, enhancing its quality and reliability. This highlights the importance of LPFs in applications where accurate and clean data is critical. However, the study also acknowledges the limitations of LPFs, such as the potential introduction of latency and the necessity for careful tuning of filter parameters. Overall, the experiment validates the utility of LPFs in improving data integrity in noisy environments.

RECOMMENDATIONS

- Adjust the sampling rate to optimise filter performance and noise reduction efficacy.
- Implement hardware of an actual LPF to compare their performances.

ACKNOWLEDGEMENT

A special thanks go out to Dr Wahju Sediono, Dr Ali Sophian, Dr Zulkifli Bin Zainal Abidin, our teaching assistants and our peers for their invaluable help and support in finishing this report. Their advice and guidance have greatly improved the quality and understanding of the project. Their commitments are greatly appreciated.

STUDENT DECLARATION

We, hereby declare that this project is entirely our work except for the documents that were given as references. Any external sources utilized for reference or inspiration have been properly cited and credited.

azim

Muhammad Azim Zufayri Bin Norazam

ammar

Muhammad Ammar Faris Bin Arafas

hafiz

Ahmad Hafizullah Bin Ibrahim

adam

Ahmad Adam Bin Norhelme