I

DIGITAL SIGNAL PROCESSING (DSP) INTERFACING

GROUP 4

MCTA 3203

SEMESTER 2 2023/2024

MECHATRONICS SYSTEM INTEGRATION

DATE OF SUBMISSION: 5/6/2024

	NAME	MATRIC NUMBER
1.	NUR SHAHEERA BINTI KAMIS	2214040
2.	NORAMIRA HUSNA BT NORKHAIRANI	2214496
3.	MUHAMMAD EIQBAL BIN HASBOLLAH	2216911

TABLE OF CONTENT

ABSTRACT

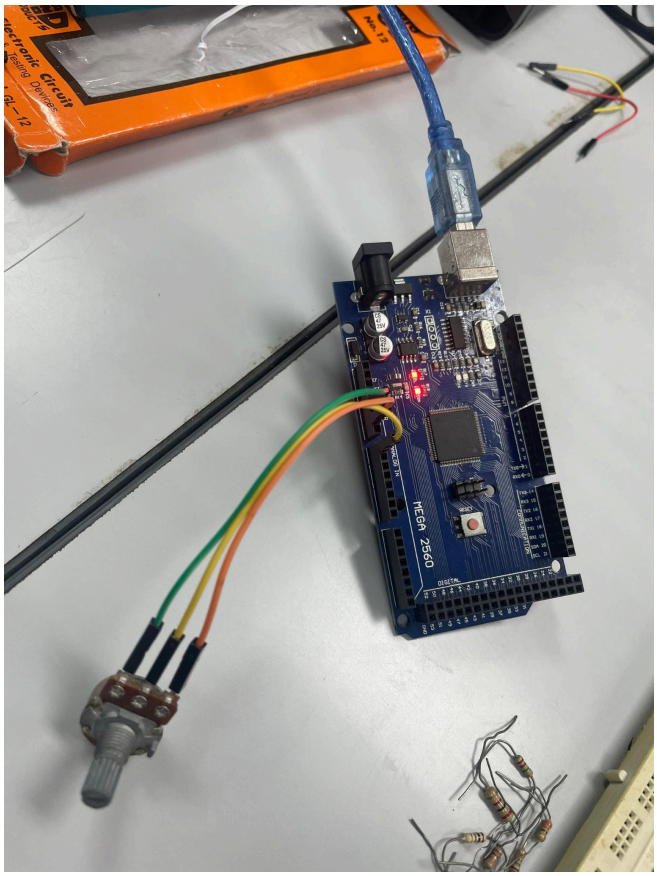
This experiment explores how low-pass and high-pass filters work using an Arduino board, focusing on reducing noise in signals. By connecting a sensor to the Arduino's input through filter circuits and writing code, it shows how these filters smooth signals and reduce unwanted noise. The method involves choosing resistor and capacitor values to set cut-off frequencies and checking the filtered outputs through the serial monitor. Results indicate that low-pass filters remove high-frequency noise, while high-pass filters eliminate low-frequency components, useful for audio and sensor data filtering. Suggestions for improvement include using high-precision components, digital filters for flexibility, and advanced data analysis tools for better understanding filter performance.

Keywords: Arduino, low-pass filter, high-pass filter, signal processing, noise reduction, digital filtering, analog input, cut-off frequency, sensor data, Arduino IDE.

MATERIAL & EQUIPMENT

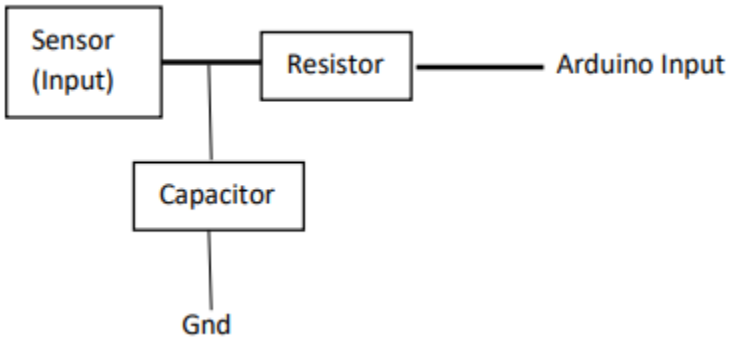
- Arduino board
- Arduino IDE
- Breadboard and jumper wires
- Analog input (sensors/ function generator etc.)
- Resistor and Capacitor (for LPF and HPF)

EXPERIMENTAL SETUP



METHODOLOGY/PROCEDURE

1. Connect the sensor to the Arduino's analog input through a low-pass filter (LPF) as illustrated in Fig. 2 below.

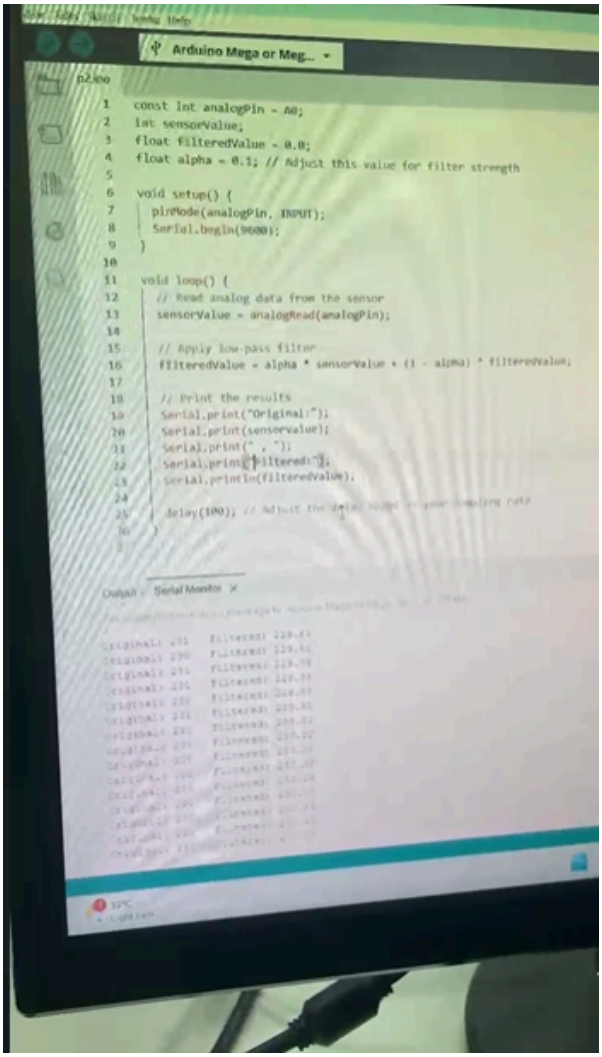


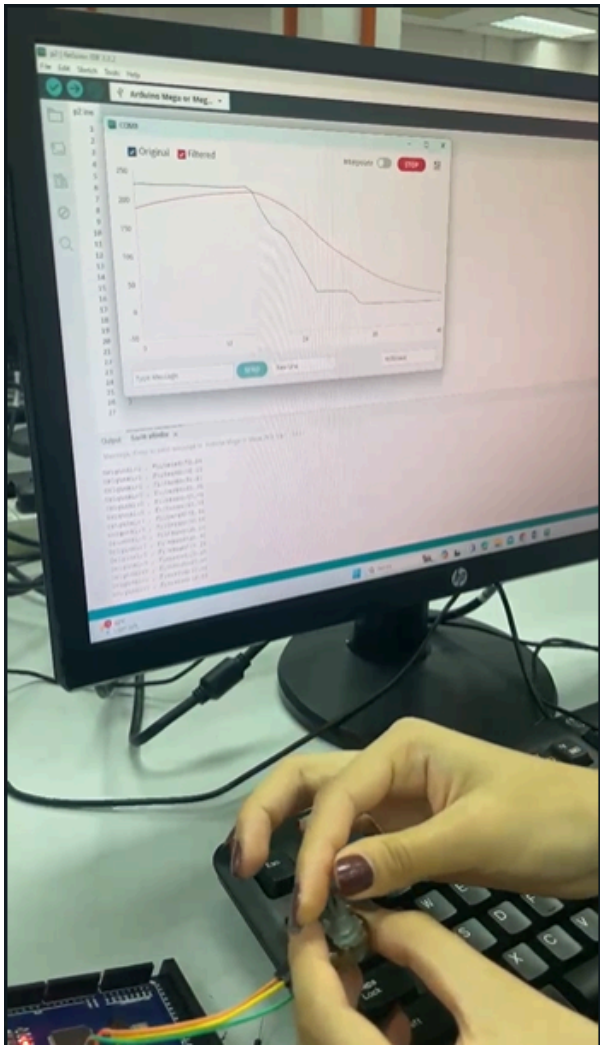
[Fig. 2]

2. The resistor and capacitor together create a low-pass filter circuit that follows the cut-off frequency formula: $f_c = 1/2pRC$. Select the resistor and capacitor values based on the desired cut-off frequency.
3. Write Arduino code to read analog data from the sensor and apply a low-pass filter.
4. Upload the code to the Arduino board.
5. Use the serial monitor in the IDE to observe the output.
6. Notice how the LPF smooths the signal and reduces high-frequency noise.
7. Discuss the principles of low-pass filtering and its applications. Explore real-world uses of digital filtering in fields such as audio processing or sensor data filtering.

*Repeat the above steps using a high-pass filter instead of a low-pass filter. Demonstrate and observe the high-pass filter's functionality.

RESULTS





DISCUSSION

1. **HARDWARE**

The principle of a low-pass filter (LPF)

A low-pass filter (LPF) is an electrical circuit that selectively permits the transmission of signals with frequencies below a designated cutoff frequency, while simultaneously reducing the strength of signals with frequencies above the cutoff frequency. The cutoff frequency is determined by the values of the resistor (R) and capacitor (C) in the circuit, according to the formula $f_c = 1 / (2\pi RC)$. This approach is essential in several applications where it is necessary to eliminate high-frequency noise in order to provide a clear and steady signal.

Components and Connections of Hardware

Arduino Board: The Arduino board acts as the core unit for processing the analog signal from the sensor and applying the low-pass filter.

Breadboard and Jumper Wires: The breadboard and jumper wires aid in the assembly of the LPF circuit and the establishment of connections between the sensor, the LPF, and the Arduino.

Analogue input: Refers to the use of a sensor, function generator, or audio signal to provide input to a system. A source of analogue signals, such as a sensor, function generator, or audio signal, supplies the input signal that requires filtering.

The resistor and capacitor: Constitute the low-pass filter circuit. The selection of their values is based on the intended cutoff frequency in order to efficiently filter out high-frequency components.

2. SOFTWARE

Arduino Code

The software part involves writing an Arduino sketch to read analog data from the sensor, apply a digital low-pass filter, and display the filtered signal on the serial monitor. Here's a step-by-step explanation and the corresponding code:

Initialize Variables:

Define variables to store the raw and filtered signal values. Set the initial values and constants needed for the low-pass filter.

Setup Function:

Initialize the serial communication. Set the analog input pin mode.

Loop Function:

Continuously read the analog signal from the sensor. Apply a simple digital low-pass filter to the signal. Print the raw and filtered signal values to the serial monitor for visualization.

3. ELECTRICAL

Arduino Board

The Arduino board works as the core unit for processing the analog signal from the sensor, applying the low-pass filter, and displaying the filtered signal on the serial monitor.

Breadboard with Jumper Wires

These are used to build the circuit without soldering, permitting quick connections between components to produce the low-pass filter circuit.

Analog Input (Sensor, Function Generator, or Audio Signal)

Provides the analog signal that has to be filtered, such as a temperature sensor, function generator, or audio source.

Resistor

A fundamental component of the low-pass filter, the resistor value, along with the capacitor, sets the cutoff frequency. For instance, a 1 kΩ resistor may be utilised.

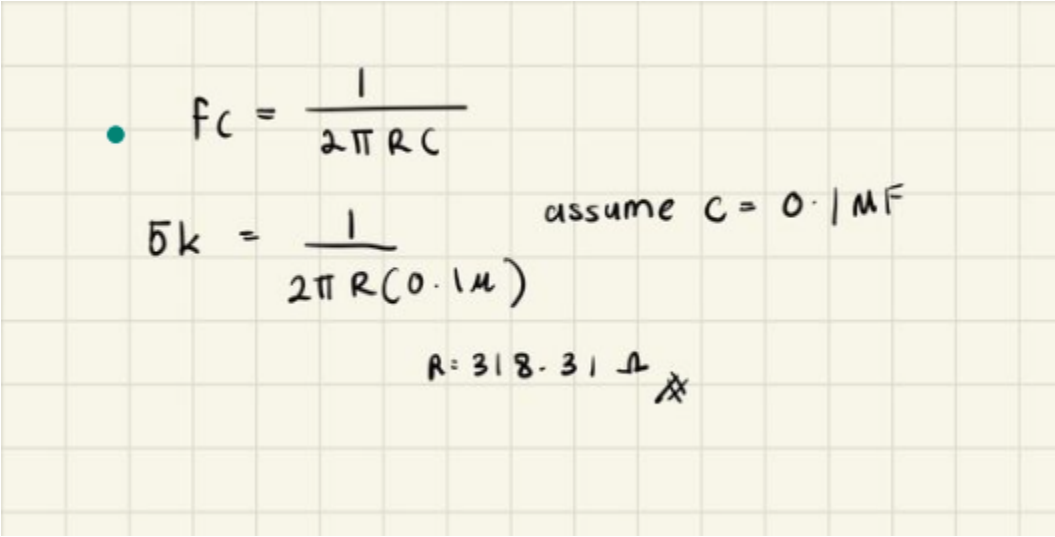
Capacitor

Along with the resistor, the capacitor controls the cutoff frequency of the low-pass filter. For example, a 15.9 µF capacitor might be utilised.

Circuit Design

The sensor output is linked to the resistor, and the junction of the resistor and capacitor is connected to the Arduino's analog input pin. The opposite end of the capacitor is grounded.

CALCULATION



CONCLUSION

This experiment highlighted the importance of filtering in electronic signal processing. By carefully selecting the resistor and capacitor values, we were able to tailor the filter characteristics to meet specific requirements. The practical implementation using an Arduino board and the straightforward coding approach demonstrated how digital microcontroller platforms can be effectively utilized for analog signal processing tasks.

Overall, the experiment provided valuable insights into the principles and applications of both low-pass and high-pass filters, showcasing their significance in modern electronic systems.

RECOMMENDATION

1. **Use Higher Precision Components**:

   Using high-precision resistors and capacitors with 1% or 0.1% tolerance significantly improves the accuracy and consistency of cut-off frequencies, ensuring more reliable and stable filter performance by minimizing variability due to component tolerances.

2. **Implement Digital Filters**:

   Implementing digital low-pass and high-pass filters using software algorithms within Arduino code offers greater flexibility and precision compared to analog filters. Digital filters allow for real-time adjustments and experimentation with various filter types, enabling fine-tuning of the filtering process without needing physical component changes.

3. **Enhance Data Analysis and Visualization**:

   Using advanced tools for data analysis and visualization, like logging data or using MATLAB or Python, provides deeper insights into filter performance. This allows better visualization of frequency response, noise reduction, and signal stability, enabling more informed filter design adjustments.

REFERENCES

1. Harvie, L. (2023, October 18). *Digital Signal Processing (DSP) in Embedded Systems*. Medium. https://medium.com/@lanceharvieruntime/digital-signal-processing-dsp-in-embedded-systems-68b4649ff441
2. *Getting into Digital Signal Processing: A Basic Introduction*. (n.d.). Www.rs-Online.com. Retrieved June 6, 2024, from https://www.rs-online.com/designspark/getting-into-digital-signal-processing-a-basic-introduction

ACKNOWLEDGEMENT

A special thanks goes out to Dr. Wahju Sediono, Dr. Ali Sophian, Dr. Zulkifli Bin Zainal Abidin, my teaching assistant, and my peers for their invaluable help and support in finishing this report. Their advice, feedback, and experience have greatly influenced the level of quality and understanding of this work. Their time, patience, and commitment to supporting my academic success are greatly appreciated.

STUDENT 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: 	**Read**	/
Name: Noramira Husna Bt Norkhairani	**Understand**	/
Matric No: 2214496	**Agree**	/

Signature: 	**Read**	/
Name: Nur Shaheera Bt Kamis	**Understand**	/
Matric No:2214040	**Agree**	/

Signature: 	**Read**	/
Name: Muhammad Eiqbal bin Hasbollah	**Understand**	/
Matric No: 2216911	**Agree**	/