Process Report | DSP

Introduction

In this document, I will discuss the most important concepts I learned in my Digital Signal Processing (DSP) course, accompanied by relevant images.

Workshop 1 - DSP Introduction

Key Takeaways

- A definition of Digital Signal Processing (DSP) and its applications.
- An understanding of the difference between continuous and discrete signals and systems.
- An understanding of the Nyquist sampling theorem and how it relates to aliasing.
- An overview of some of the tools used in DSP, such as MATLAB.

Challenges

Working with MATLAB

During the first workshop of the digital signal processing subject, I learned how MATLAB works and how to use it. By completing the provided tutorial, I gained a solid understanding of the software's functionalities and capabilities. At the end of the workshop, I received a certificate, which is included in my portfolio.

https://prod-files-secure.s3.us-west-2.amazonaws.com/d1d3989f-b8c8-40c6-a850-7da1b7d3e359/62473de1-628e-49ef-b60d-dd4d95019aa0/MathLab-OnRamp-Certificate.pdf

Workshop 2 - Tools workshop

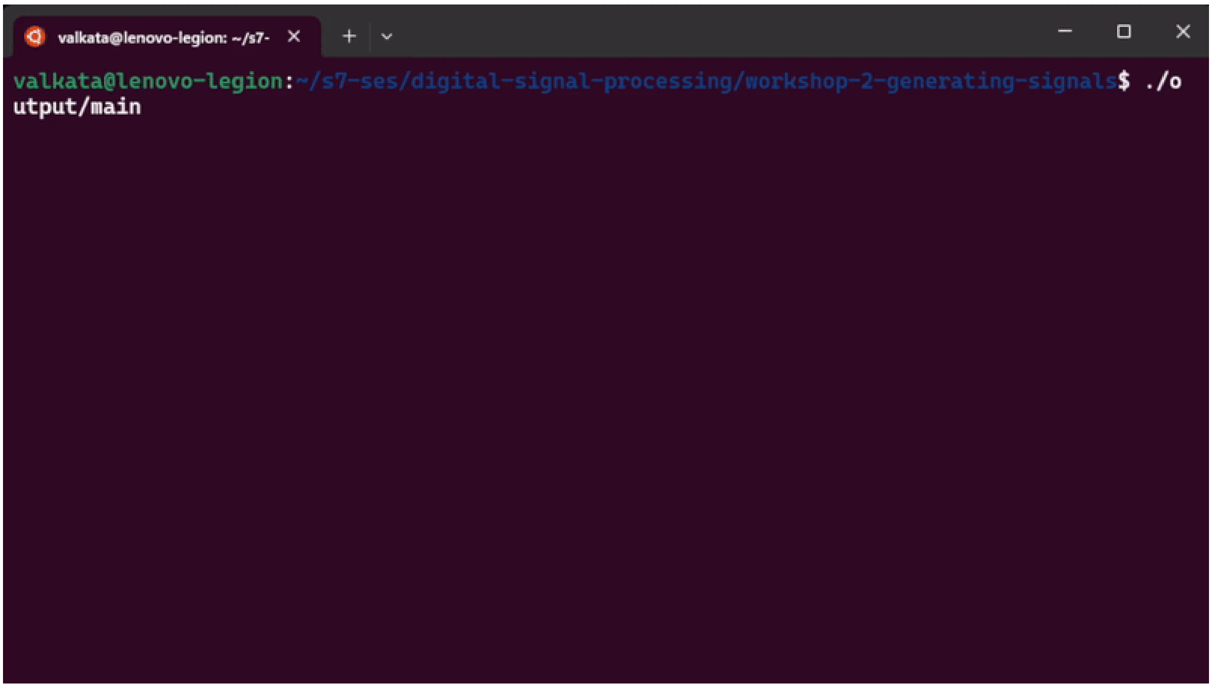
Key Takeaways:

- How to build a sine generator in C++ using only multiplication.
- How to mix audio files in MATLAB.
- Complex number notation, multiplication, and application.
- Continuous and discrete time complex exponentials parameters.

Challenges:

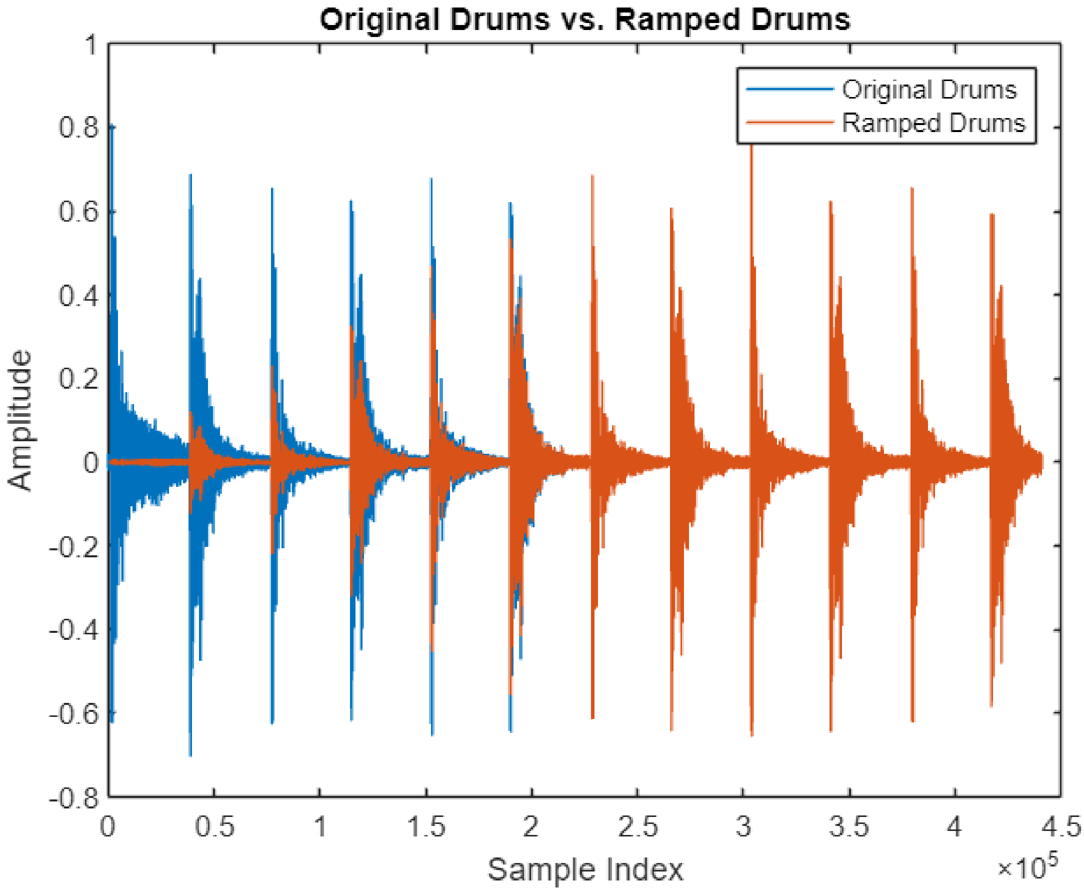
Sine generator

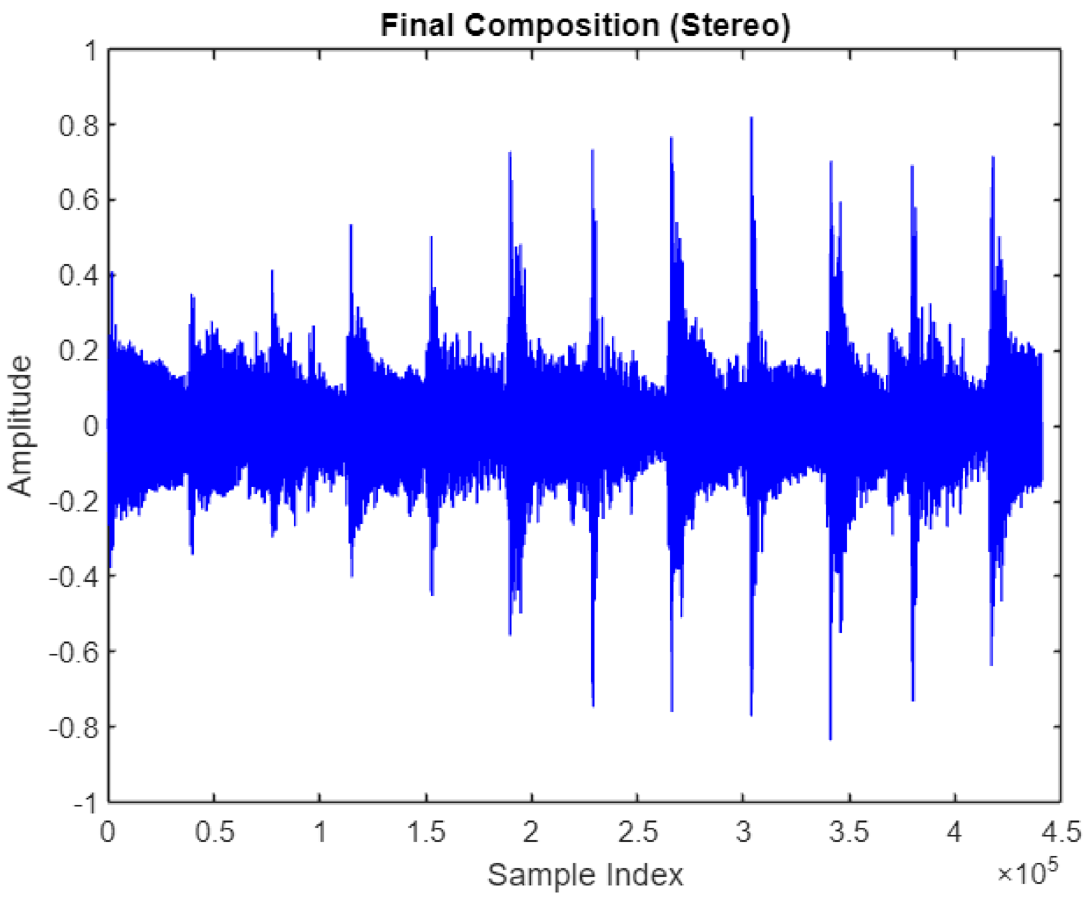
I successfully generated sine and cosine waves using complex numbers in C++. This involved understanding the relationship between complex exponentials and trigonometric functions. By manipulating complex numbers, I could control the frequency, amplitude, and phase of the generated waveforms. The following gif shows how the complex number change on real and imaginary axes during a frequency sweep.



Band Mixing

The band mixing challenge involved using MATLAB to import and manipulate audio files. This included tasks like adjusting volume levels, applying effects, and potentially combining multiple tracks. The goal was to gain practical experience with audio processing techniques in MATLAB and understand how to create a final audio mix.


Original Drums vs. Ramped Drums
Original Drums
Ramped Drums
Amplitude
Sample Index
×10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5


Final Composition (Stereo)
Amplitude
Sample Index
×10^5
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5

Workshop 3 - LTI systems in time domain

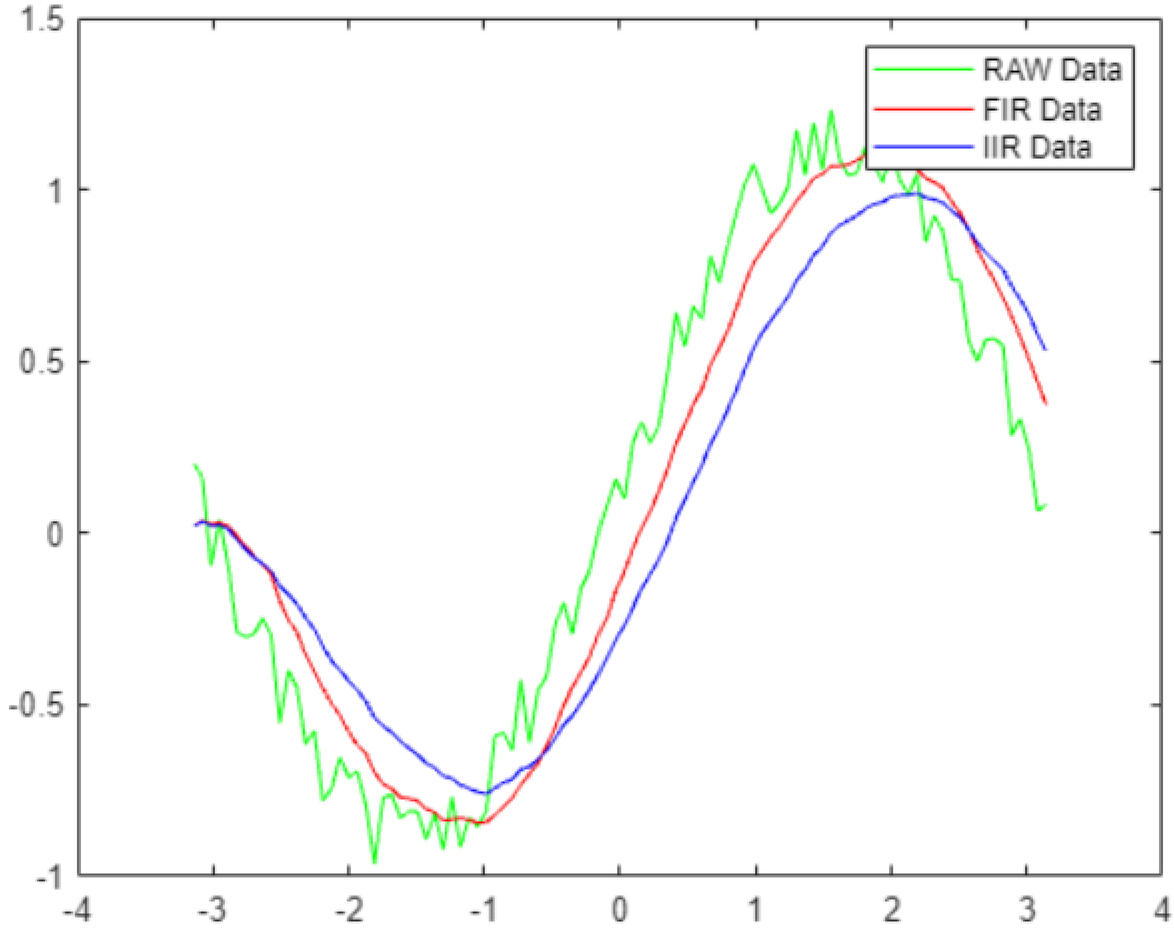
Key Takeaways:

- **Linear Time-Invariant (LTI) Systems:** Understanding the properties and significance of LTI systems in signal processing.
- **Impulse Response:** The concept of impulse response as a fundamental characteristic of LTI systems, describing their behavior in response to a brief input signal.
- **Convolution:** The role of convolution as the mathematical operation that describes how an LTI system modifies an input signal over time.
- **FIR and IIR Filters:** Introduction to Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, two common types of digital filters used in signal processing.
- **Moving Average (MA) Filters:** Understanding MA filters as a specific type of FIR filter and how they can be implemented using both direct and recursive approaches.

Challenges

Comparing 2 moving average filters

I implemented both Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters to remove random noise from a sine wave. FIR filters have finite-length impulse responses, while IIR filters have infinite-length impulse responses due to feedback. I learned how to design and implement these filters using difference equations and observed their effects on the noisy signal.



Workshop 4 - LTI systems in frequency domain

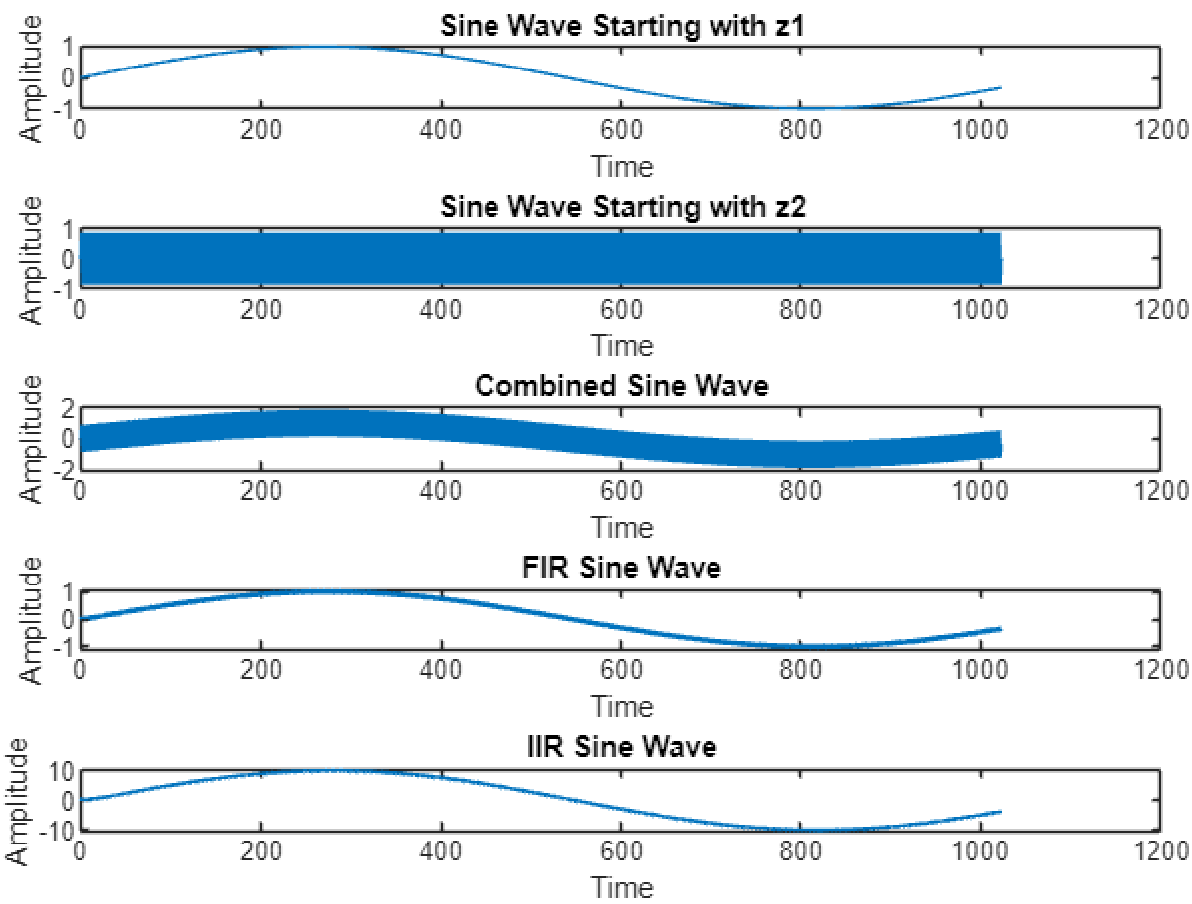
Key Takeaways:

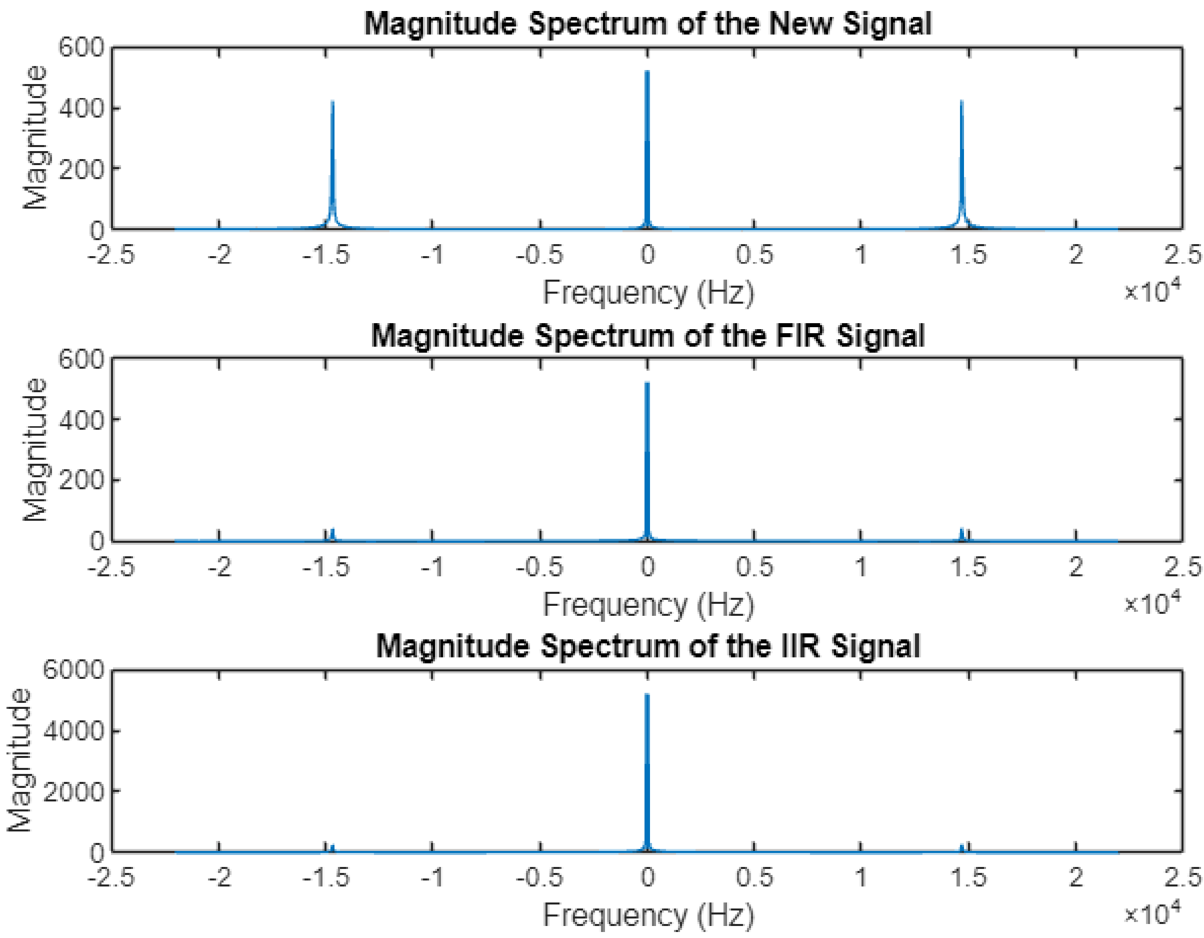
- **Fourier Analysis**: The concept of decomposing a signal into its constituent frequency components using the Fourier transform, providing insights into its spectral content.
- **Frequency Response:** The relationship between the input and output spectra of an LTI system, characterizing how the system modifies different frequencies.
- **Signal Filtering:** Applying filters in the frequency domain to modify specific frequency components of a signal, such as removing noise or isolating particular frequencies.

Challenges

Frequency analyzer

I added two sine waves and plotted the Fast Fourier Transform (FFT) of the resulting signal. The FFT is a fundamental tool in DSP for analyzing the frequency content of signals. By plotting the FFT, I could visualize the individual frequency components of the combined sine waves.





Workshop 5 - Filtering in the frequency domain

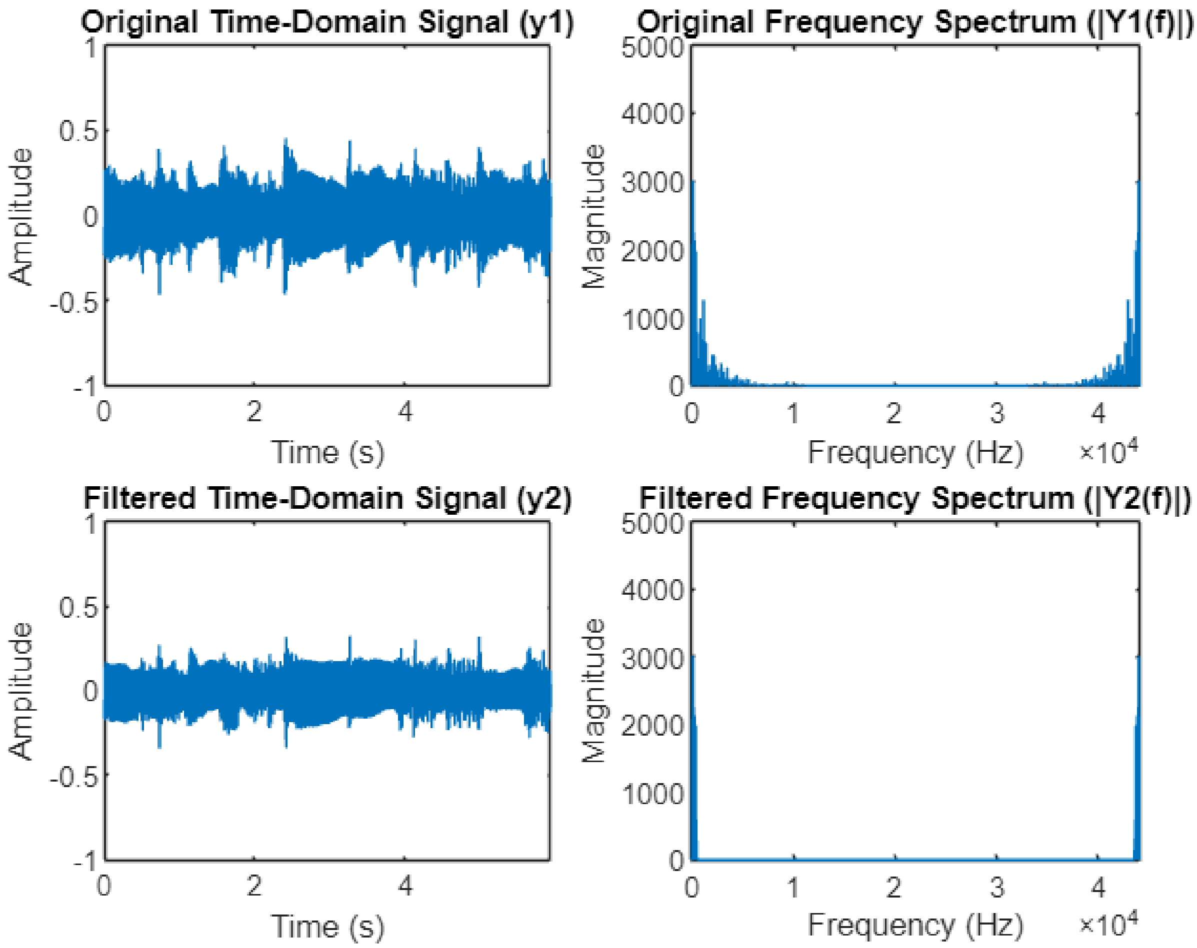
Key Takeaways:

- **Frequency Domain Filtering:** Understanding the process of filtering signals by manipulating their frequency components in the frequency domain.
- **Ideal Filters:** Exploring the concept of ideal filters, which have perfect passband and stopband characteristics, and their limitations in practical implementations.
- **Filter Design:** Learning about different filter design techniques, such as windowing methods and filter transformations, to create filters with desired frequency responses.
- **Applications:** Understanding the applications of frequency domain filtering in various fields, such as audio processing, image enhancement, and communication systems.

Challenges

Filtering in the frequency domain

I loaded a guitar sample and applied filtering in the frequency domain. This involved transforming the time-domain signal into the frequency domain using the FFT, applying a filter to modify specific frequency components, and then transforming the filtered signal back to the time domain using the inverse FFT. I listened to the filtered result and observed how different filters affected the sound of the guitar sample.



Workshop 6 - Filtering in the time domain

Key Takeaways:

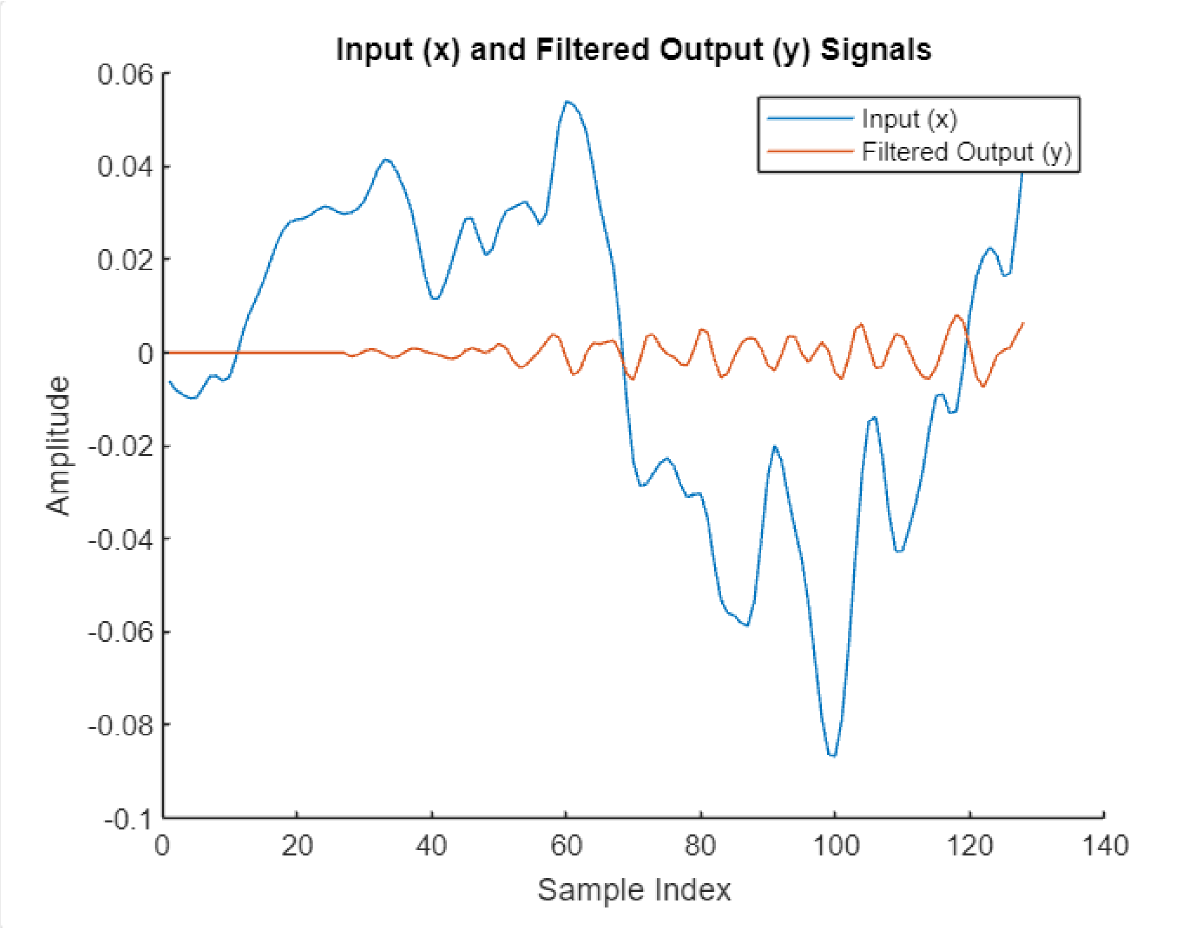
- **Time Domain Filtering:** Understanding the process of filtering signals by convolving them with a filter's impulse response in the time domain.

- **Filter Types:** Exploring different types of filters, such as low-pass, high-pass, band-pass, and band-stop filters, and their effects on signals.
- **Filter Design:** Learning about filter design techniques in the time domain, such as the window method and the bilinear transform, to create filters with desired characteristics.
- **Filter Implementation:** Understanding how to implement filters in software or hardware using difference equations or block diagrams.

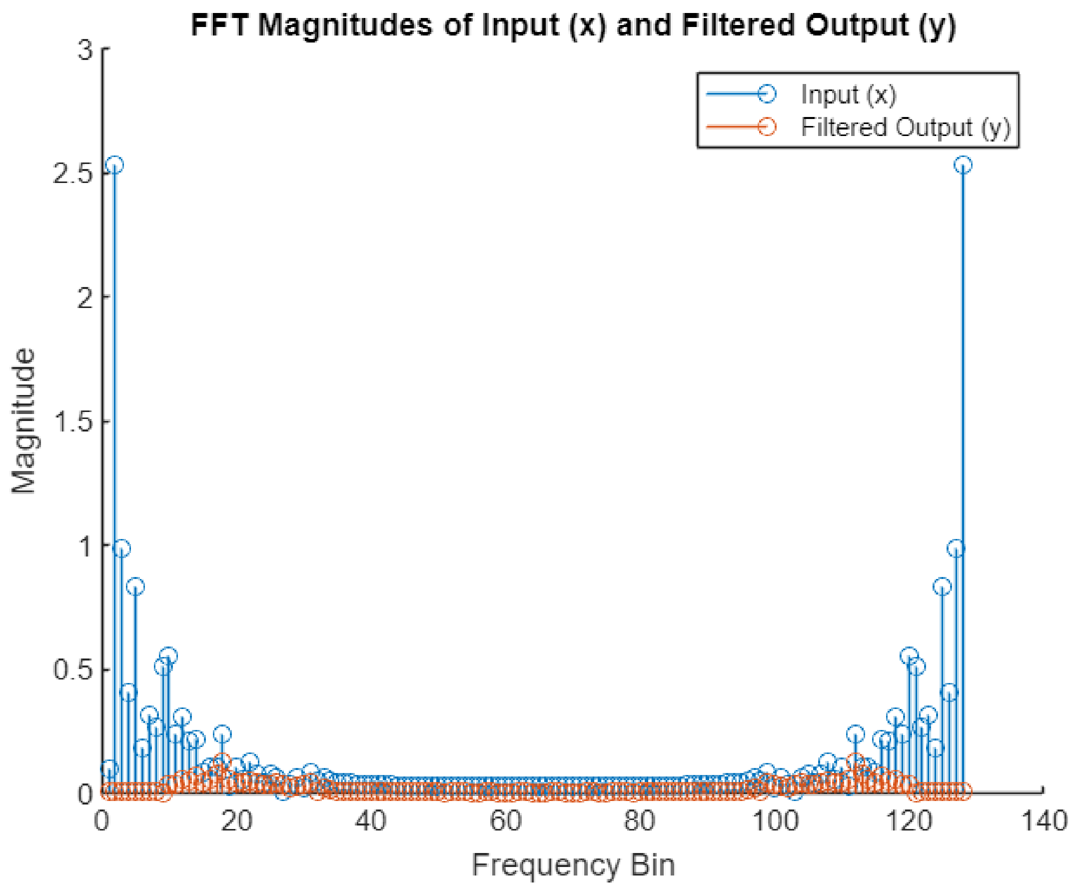
Challenges

Filtering in the time domain

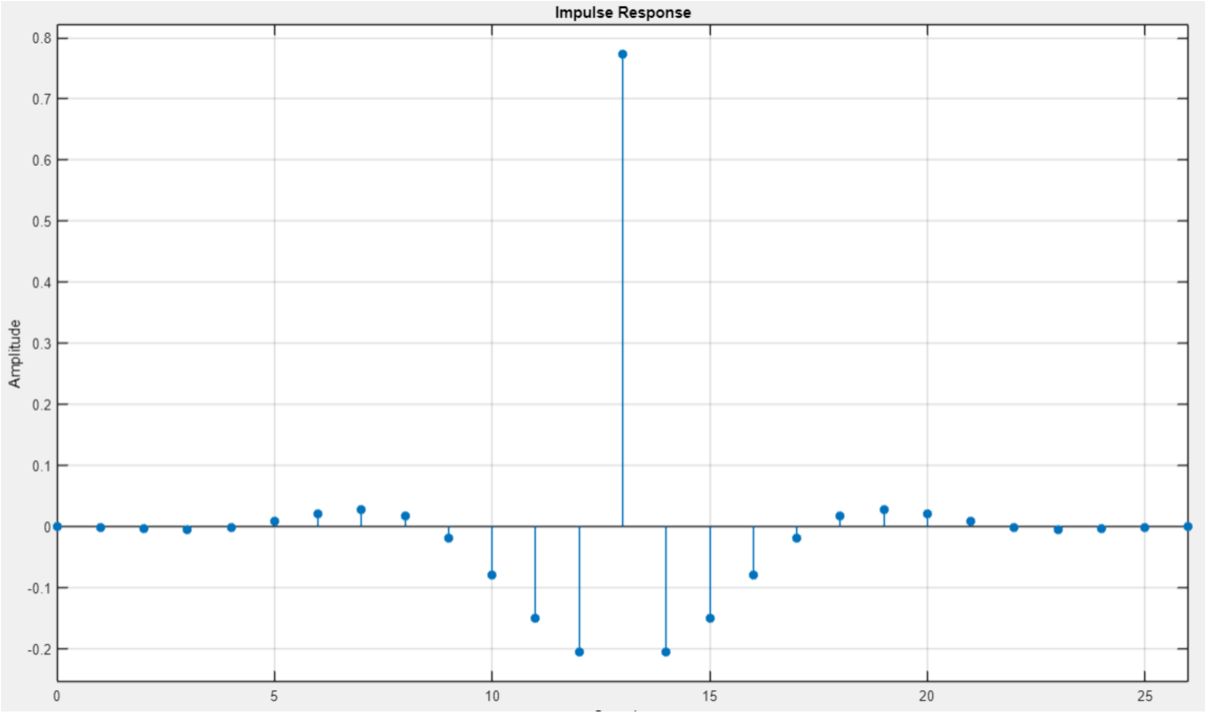
I delved into the concept of convolution, which is a mathematical operation that describes how a linear time-invariant (LTI) system modifies an input signal. I learned that the output of an LTI system is the convolution of the input signal with the system's impulse response. By understanding convolution, I gained insights into how filters operate in the time domain.



Input (x) and Filtered Output (y) Signals
Input (x)
Filtered Output (y)
Amplitude
Sample Index
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
-0.08
-0.1
0
20
40
60
80
100
120
140



FFT Magnitudes of Input (x) and Filtered Output (y)
Input (x)
Filtered Output (y)
Magnitude
Frequency Bin
3
2.5
2
1.5
1
0.5
0
0
20
40
60
80
100
120
140



Workshop 7 - Amplitude modulation

Key Takeaways:

- **Amplitude Modulation (AM):** Understanding the process of modulating a carrier signal's amplitude with a message signal to transmit information.
- **Modulation Index:** The concept of modulation index, which determines the extent of amplitude variation in AM signals.
- **Demodulation:** The process of recovering the original message signal from the modulated carrier signal.
- **Applications:** Exploring the applications of AM in radio broadcasting, analog communication systems, and other areas.

Conclusion

Throughout this course, I gained a solid foundation in DSP principles and techniques. I learned how to analyze and manipulate signals in both the time and frequency domains. I also gained practical experience in implementing filters, generating waveforms, and using MATLAB for DSP tasks. These skills are essential for various applications, including audio processing, image processing, and communication systems.

In conclusion, this DSP course has equipped me with valuable knowledge and skills in the field of digital signal processing. I am excited to apply this knowledge to future projects and continue exploring the fascinating world of DSP.