**DEPARTMENT OF COMPUTER &
SOFTWARE ENGINEERING**

CE&ME, NUST, RAWALPINDI

EE-232 Signals and Systems

FINAL PROJECT REPORT

SUBMITTED TO:

Sir Haris Ata ullah

SUBMITTED BY:

Ayezah Maryam (366773)

Sara Imran (367711)

Musfira Aslam (382132)

DE- 43 Dept CE

Submission Date: 17/1/24

5 Band Graphical Equalizer

Abstract:

This project presents the development of a 5-band graphical equalizer implemented in MATLAB. The equalizer is designed to provide users with precise control over audio output by incorporating five distinct filters targeting specific frequency bands.

A user-friendly GUI facilitates real-time adjustments, enhancing the overall user experience. The project focuses on the importance of accurate frequency response shaping by designing and implementing filters. Experimental results showcase the effectiveness of the graphical equalizer in modifying the audio spectrum.

Introduction:

Audio equalization plays a pivotal role in shaping the tonal characteristics of audio signals, enabling users to tailor the sound to their preferences. In the realm of digital signal processing, graphical equalizers have become essential tools for audio enthusiasts, professionals, and hobbyists alike. This project introduces a 5-band graphical equalizer implemented in MATLAB, aiming to provide users with a versatile and intuitive platform for audio spectrum modification.

The primary objective of this project is to leverage digital signal processing techniques to design and implement a graphical equalizer capable of precise frequency response adjustments. By dividing the audio spectrum into five distinct bands, users gain the ability to selectively boost or attenuate specific frequency ranges, thereby customizing the audio output according to their preferences.

The development of this graphical equalizer involves the application of MATLAB, a powerful tool for numerical computing and signal processing. MATLAB's rich set of functions and libraries facilitate the creation of a user-friendly GUI, allowing real-time interaction with the equalization parameters. Users can visually identify and modify the impact of each band on the audio spectrum, making the equalizer an intuitive and effective tool for audio manipulation.

Literature Review:

The concept of audio equalization has a rich history in the field of audio engineering, with graphical equalizers being widely employed for shaping the frequency response of audio signals. A comprehensive literature review reveals the evolution of equalization techniques and the diverse applications of graphical equalizers in both professional audio processing and consumer electronics.

Historical Evolution of Equalization: Equalization, in its simplest form, involves adjusting the amplitude of different frequency components in an audio signal. The early days of audio equalization were characterized by analog circuits, including passive and active filters, designed to modify the frequency response of audio playback systems. The advent of parametric equalizers brought flexibility by allowing users to adjust center frequencies and bandwidths.

Graphical Equalizers: Graphical equalizers represent a significant advancement in audio equalization, providing a visual representation of the frequency response. Traditional graphic equalizers consist of a series of sliders, each corresponding to a specific frequency band. Users can intuitively adjust the sliders to boost or attenuate frequencies, offering a hands-on approach to tonal shaping. These devices became standard fixtures in recording studios, live sound reinforcement, and home audio systems.

Digital Signal Processing in Audio: The transition to digital signal processing (DSP) revolutionized audio equalization. Digital equalizers, both hardware and software-based, leverage the power of algorithms to process audio signals with precision. MATLAB, a prominent numerical computing environment, has become a preferred platform for implementing digital audio processing algorithms due to its versatility and ease of use.

Real-Time Audio Processing: The demand for real-time audio processing has grown with the increasing prevalence of digital audio workstations (DAWs) and live sound applications. Real-time equalization allows users to make instantaneous adjustments to the audio output, enhancing the user experience during music production, live performances, and multimedia applications.

User Interfaces for Audio Equalization: The design of user interfaces for graphical equalizers plays a crucial role in ensuring accessibility and ease of use. Researchers and developers explore various approaches to creating intuitive interfaces, considering factors such as visualization, interactivity, and responsiveness.

Application of Graphical Equalizers in MATLAB: MATLAB's capabilities in signal processing and graphical user interface (GUI) development make it a popular choice for implementing digital audio equalizers. Researchers and practitioners have utilized MATLAB to create interactive equalization tools, combining the flexibility of DSP algorithms with user-friendly interfaces.

In summary, the literature surrounding audio equalization reflects a dynamic evolution from analog circuitry to sophisticated digital processing. The integration of graphical interfaces, coupled with advancements in DSP and real-time processing, has enhanced the accessibility and effectiveness of audio equalization tools. This project contributes to this ongoing narrative by implementing a 5-band graphical equalizer in MATLAB, emphasizing both functionality and user interaction.

Objectives:

I.   Design Graphical Equalizer and observe the frequency response of equalizer.

II. Implement the equalizer in Simulink in MATLAB.

III. Design 5 filters to filter the sound at different frequencies and apply gain.

IV. Implement the equalizer using MATLAB GUI.

Methodology:

➢ Simulink Model:

The Simulink Model employs a multi-filter approach, with five filters targeting specific frequency bands (63 Hz, 250 Hz, 1000 Hz, 4000 Hz, 16000 Hz). Each filter's gain is dynamically adjustable through corresponding sliders, allowing users precise control over individual frequency components. The model visualizes both the filtered signals and the final equalized audio output using scopes.

Part I: Filter Design

The equalizer consists of 5 filters (1 low-pass, 1 high-pass and 3 band-pass filters). Low-pass incorporates all low frequency signals and high-pass allow all high frequency signals meanwhile band-pass allow selected range of frequencies. All filters are Butterworth filters as they have stable frequency response for the pass range that ensures noise cancellation. All filters have same Quality Factor (Q = 0.73) for better frequency response.
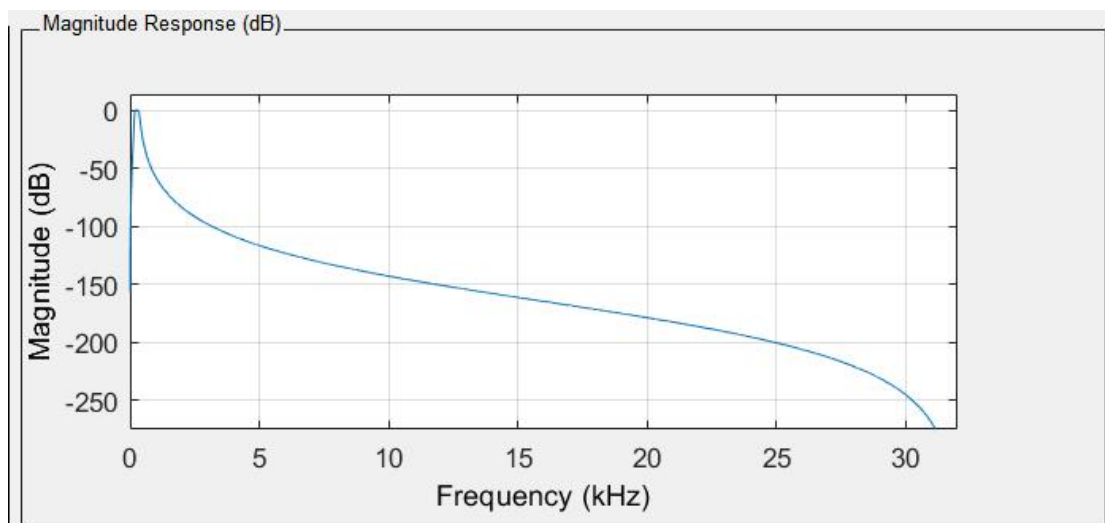
The details of 5 filters are given in the table.

$$Q = 1.47. \qquad F_s = 64000Hz \qquad Order = 8.$$

F_c(Hz)	63	250	1000	4000	16000
	(Low-Pass)				(High-pass)
F_c1(Hz)	63	177	710	2840	16000
F_c2(Hz)	63	355	1420	5680	16000

Filter 1 Response: **(Lowpass)**
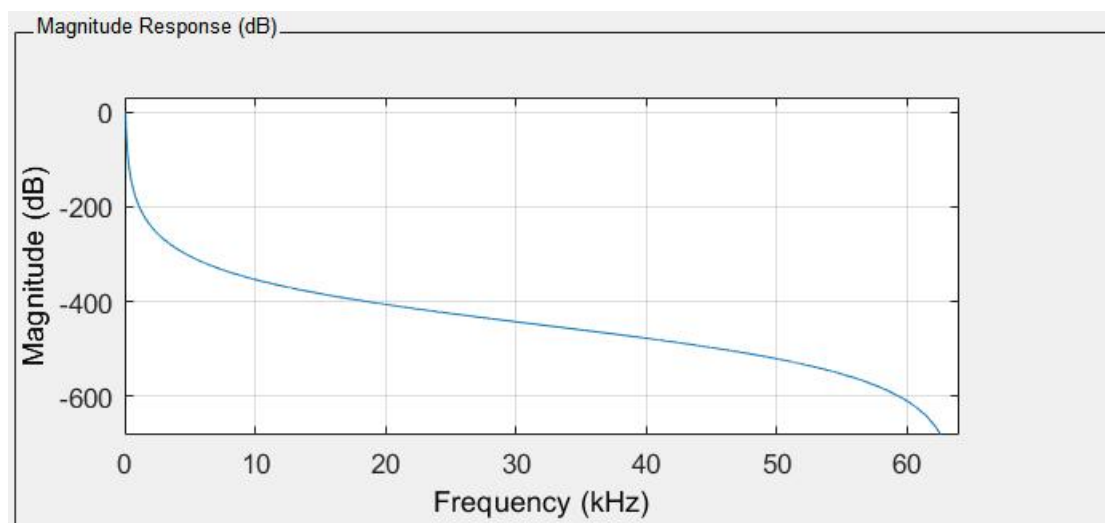
Filter 2 Response: **(Bandpass1)**
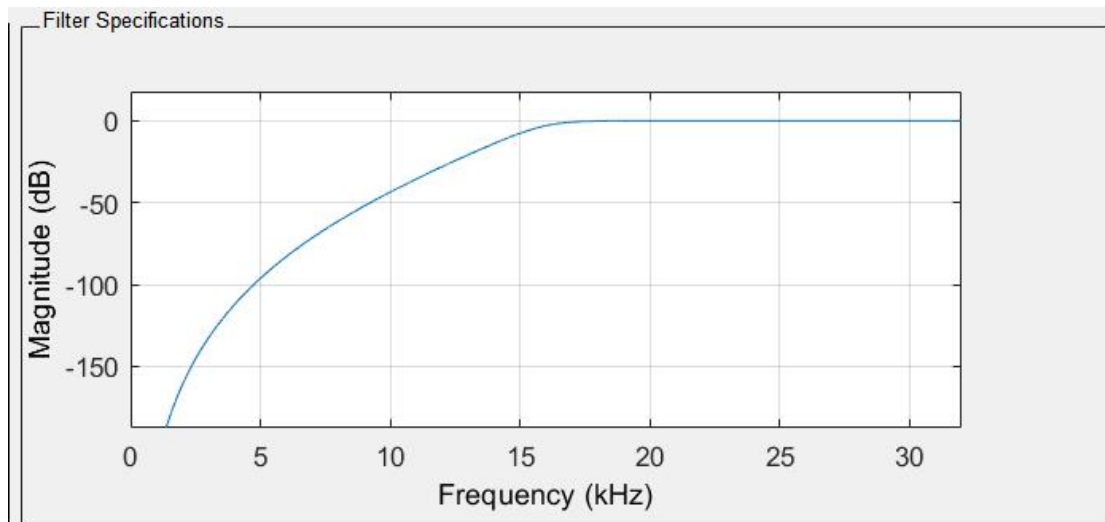

Filter 3 Response: **(Bandpass2)**
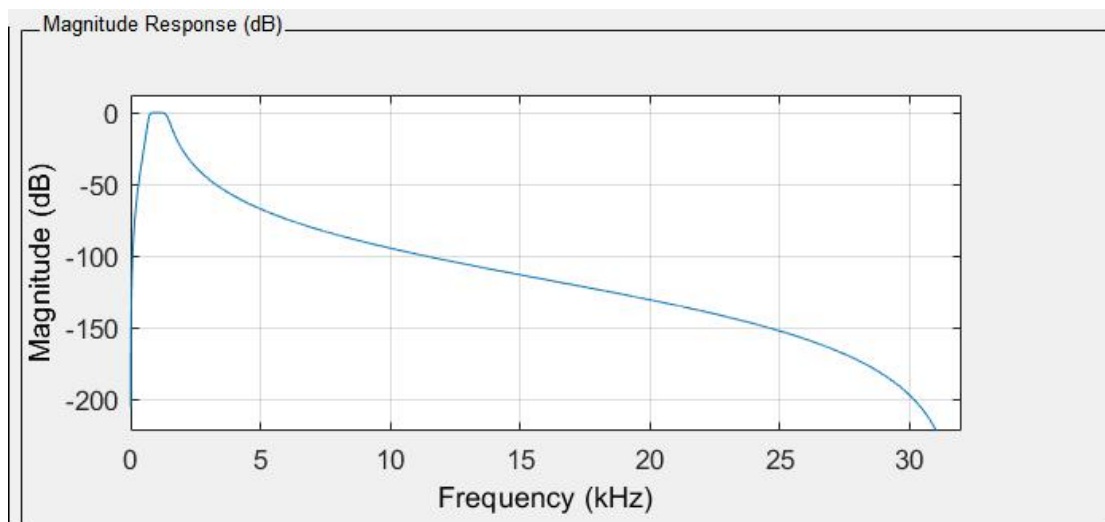
Filter 4 Response: **(Bandpass3)**
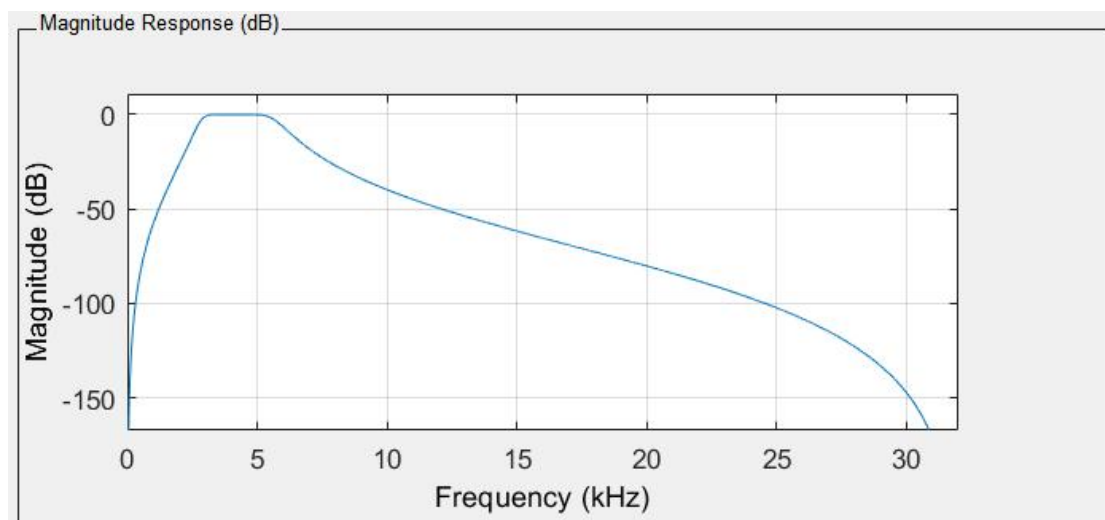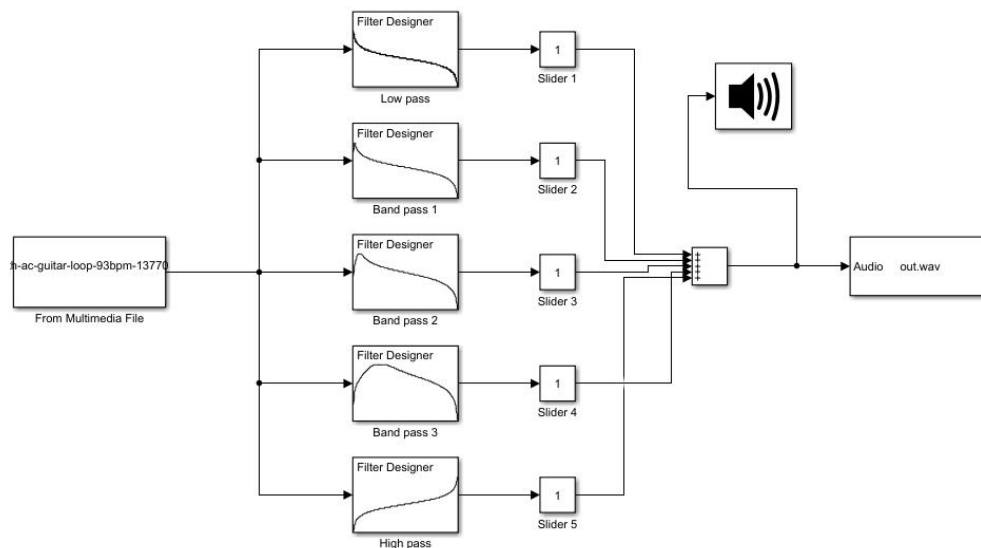
Filter 5 Response: (Highpass)

Simulink:

Butterworth Filter:

The MATLAB Butterworth function utilizes a five-step algorithm to filter signals. It starts by finding lowpass analog poles, zeros, and gain using 'buttap' and then transforms them into state-space form. The digital filter design involves a bilinear transformation, converting the state-space representation back into a transfer function with vector coefficients [b, a]. Input parameters include system order (n), cutoff frequency (w), and filter type.

For bandpass and bandstop filters, a two-element vector is required. The cutoff frequency adheres to the Nyquist theorem, with between 0 and 1 in units of π rad/sample. The quality factor (Q) of a bandpass filter influences

response width: a smaller Q yields a broader response, while a larger Q results in a narrower one.

Order:

Higher filter orders result in steeper roll-offs, impacting the filter's selectivity. However, excessively high orders may lead to increased processing complexity.

Lower orders provide gentler roll-offs, making them suitable for broad adjustments. The ideal order depends on desired precision and system constraints.

Quality Factor (Q):

The Quality Factor, denoted as Q, influences the bandwidth of the filter. A higher Q narrows this bandwidth for more focused frequency adjustments. Q is directly related to a filter's selectivity but inversely related to its bandwidth.

F_c (Center Frequency):

The center frequency FC represents the midpoint of a targeted frequency band. Usually 67% of max frequency.

F_s (Sampling Frequency):

The sampling frequency F_s should be greater than 2 times the max center frequency (Nyquist theorem). In our equilizer we used Fs = 64000Hz.

## Part II: GUI Design

➢ **GUI Model:**

The MATLAB App Designer GUI complements the Simulink Model by providing users with an intuitive interface for real-time interaction. Knobs enable users to select specific frequency bands, and sliders allow them to adjust gain levels. The GUI incorporates features for loading audio files and initiating playback, enhancing the overall user experience.
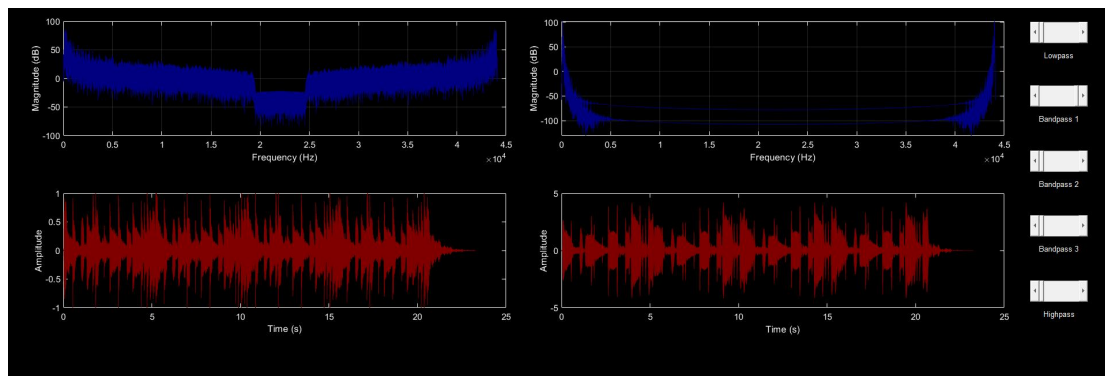
Design:
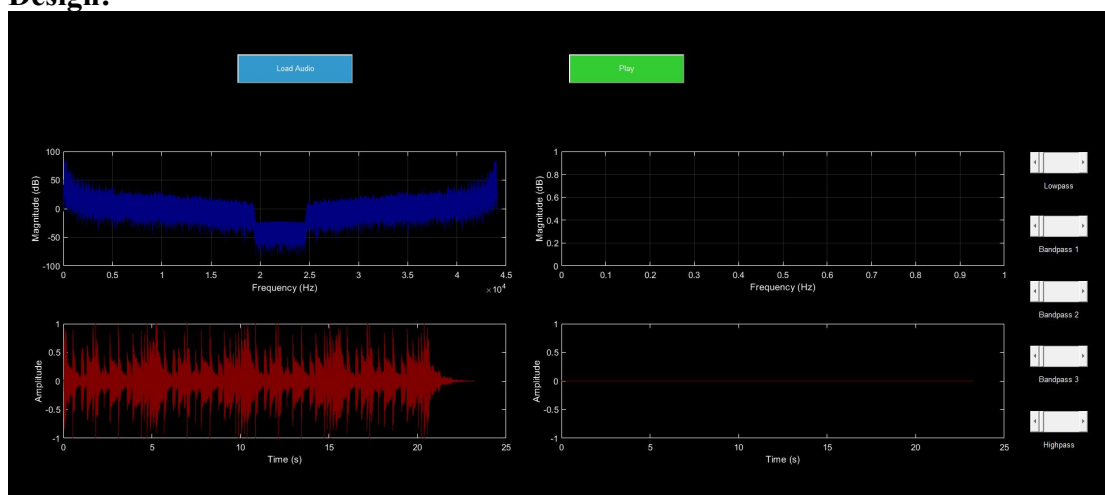
a. Load/Play audio button
b. Gain sliders   (val = 0 to 10)
c. Input/output signal
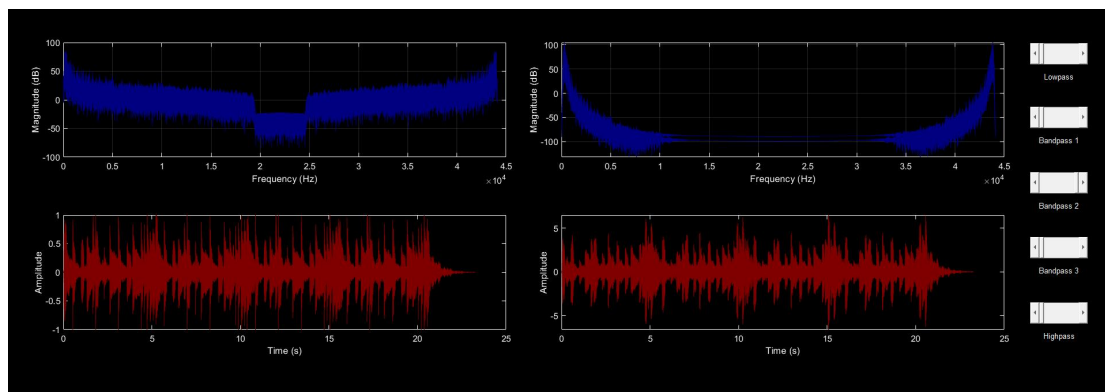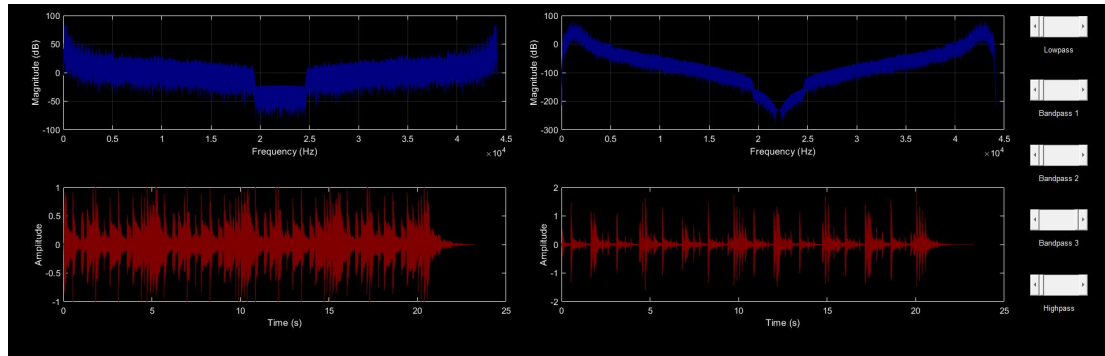d. Input/Output spectrum.

Results:
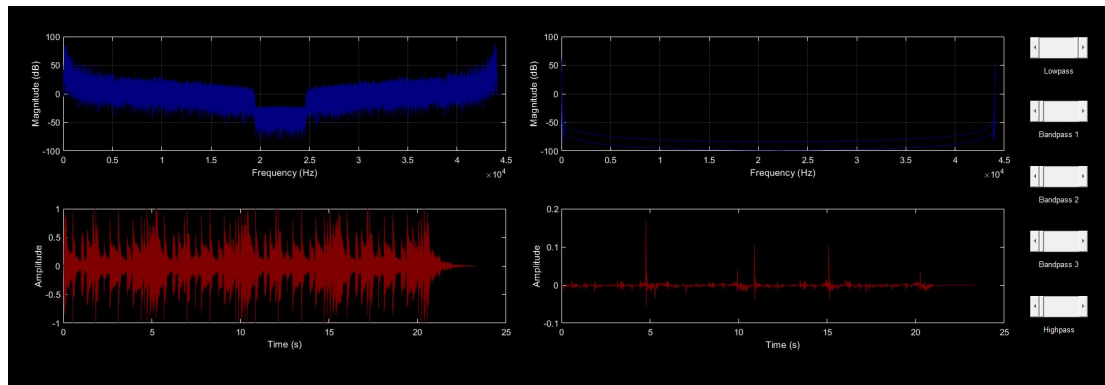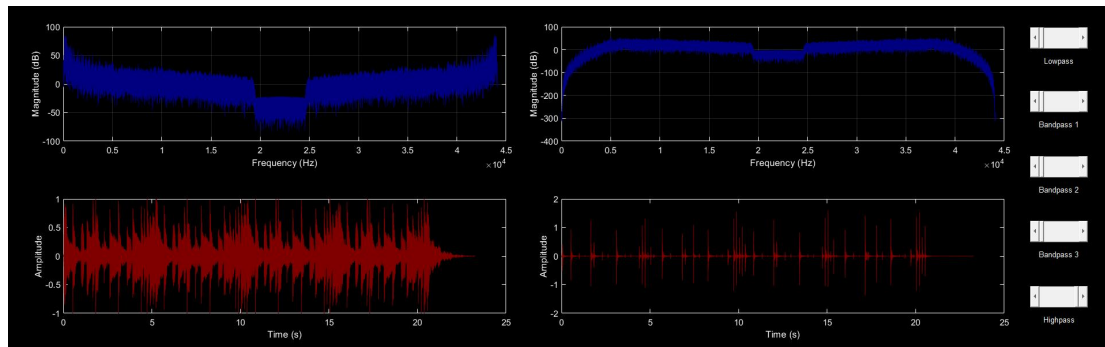

Full Gain to Lowpass

Full Gain to Bandpass 1

Full Gain to Bandpass 2

Full Gain to Bandpass 3

Full Gain to Highpass

Conclusion:

In conclusion, the project demonstrates the successful synthesis of theoretical design, simulation, and practical implementation, resulting in a functional 5-band GraphicEqualizer. We observed how an audio signal changes when different gains are applied to it. The frequency and time analysis depict the changes that occur on the audio and help us develop a better understanding of how an equalizer works. The combination of Simulink and GUI elements allows users to easily operate and observe the working of audio equalizer, meeting the outlined project goals effectively.

References:

1. Digital Audio Equalizer - File Exchange - MATLAB Central (mathworks.com)

2. Equalization - MATLAB & Simulink (mathworks.com)

3. Graphic Equalization - MATLAB & Simulink (mathworks.com)

4. https://www.techtarget.com/searchenterprisedesktop/definition/graphic-equalizer