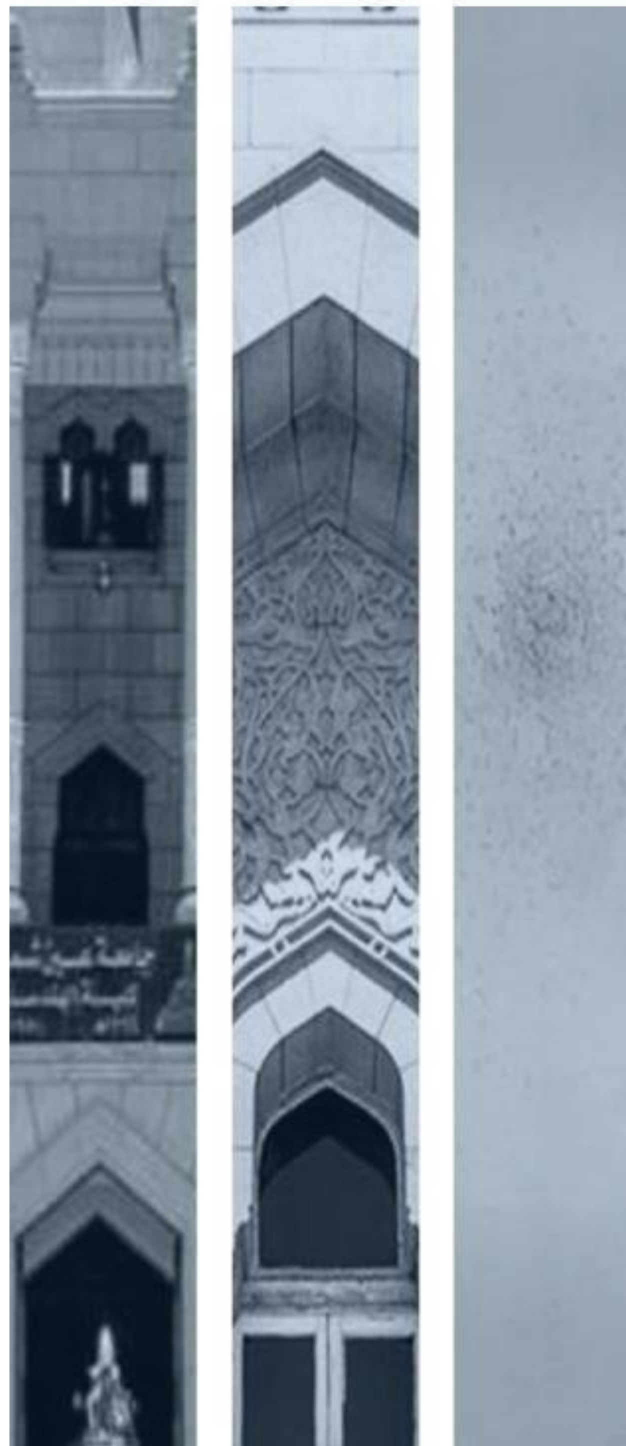
Submitted Assessment Task

Course: Signals and Systems Fundamentals

Course Code: ECE 251



Fall Semester – 2024

Report: Signal Processing with Butterworth Filters

Project Documentation

Submitted to
Dr. Michael Naiem Abdelmassih Ibrahim

Design Document Prepared by:

NAMES	ID
Ahmed Mohamed Tabbash	20P1076
Paula Emil Alexan Aziz	2200750
Ahmed Abdallah Hassan Nofull	22P0255
Kareem younis foad yehya	22P0136
Yaman Abdelhamid Elewa	2002078

The contribution of each member of the group:

1. **Ahmed Mohamed Tabbash** (Primary Contributor)
   - **Main Contributions**: Responsible for the overall project design and implementation of signal processing steps. led the development of the signal generation (`x(t)`), filtering techniques (Butterworth filters), and handling of frequency spectrum calculations. also integrated the audio file generation and worked on the energy computations of the signals.

2. **Paula Emil Alexan Aziz**
   - **Main Contributions**: Assisted in testing and validating the filter designs, including the Butterworth low-pass and high-pass filters. worked on the plotting functions for the frequency spectrum and energy computations. also contributed to the audio file generation process and verifying the output files for consistency.

3. **Ahmed Abdallah Hassan Nofull**
   - **Main Contributions**: Focused on debugging and ensuring the proper functioning of signal processing steps in GNU-Octave. worked on integrating the code sections related to frequency spectrum plotting and helped with error handling for any inconsistencies in the signals. Additionally, assisted in verifying the output of the filter designs.

4. **Kareem Younis Foad Yehya**
   - **Main Contributions**: Contributed to the coding of the Butterworth filter designs, including selecting appropriate filter orders and cut-off frequencies. was instrumental in implementing the `freqz` function to visualize the magnitude and phase responses of the filters. Additionally, helped in integrating the low-pass and high-pass filters into the signal processing chain.

5. **Yaman Abdelhamid Elewa**
   - **Main Contributions**: Contributed to the analysis and interpretation of the energy values of the signals before and after filtering. helped with the documentation of the project and ensuring all results were well-explained. also worked on preparing the final report by organizing and labeling all plots and including the audio files generated during the project.

Contents

1. Introduction

The purpose of this project was to design Butterworth low-pass and high-pass filters to process an audio signal. The signal consisted of a sum of four cosines with frequencies of 500 Hz, 1000 Hz, 1500 Hz, and 2000 Hz. We applied the filters to examine their effects in both the time and frequency domains. The project also involved calculating the energy of the signals before and after filtering, as well as generating audio files corresponding to the filtered signals.

2. Generated Signals

Original Signal (`x(t)`)

The original signal `x(t)` was defined as the sum of four cosine functions with different frequencies:

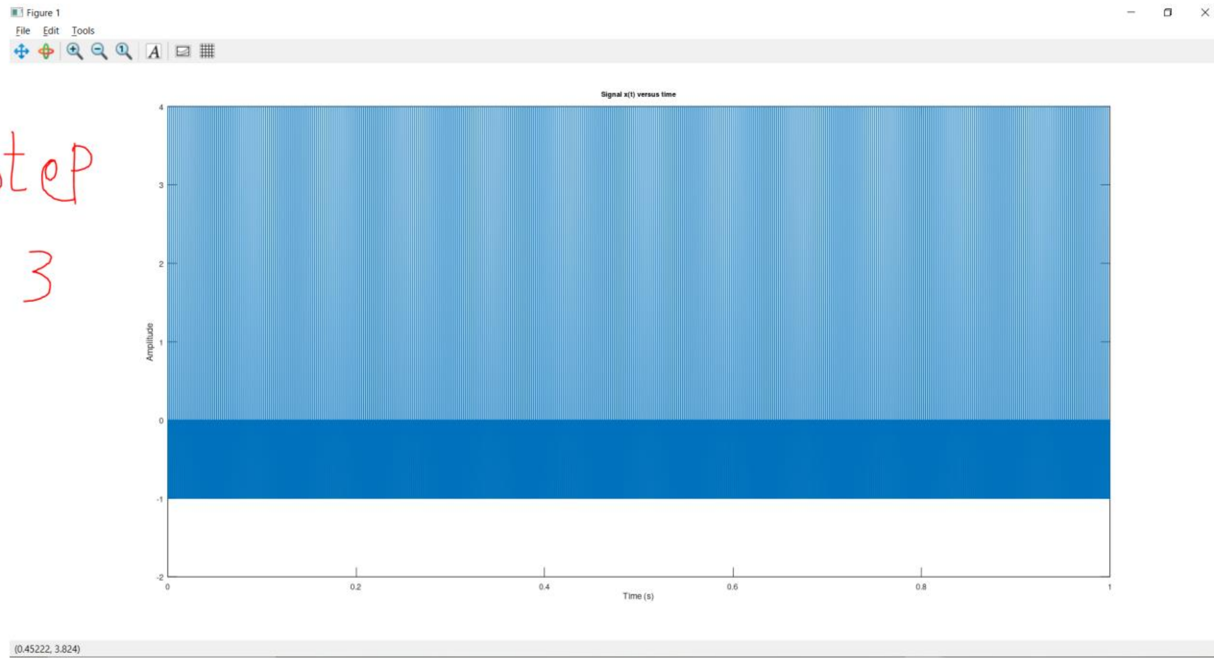
$$x(t) = \cos(2\pi f_1 t) + \cos(2\pi f_2 t) + \cos(2\pi f_3 t) + \cos(2\pi f_4 t)$$

Where the frequencies were

$f_1 = 500\,\mathrm{Hz}$, $f_2 = 1000\,\mathrm{Hz}$, $f_3 = 1500\,\mathrm{Hz}$, and $f_4 = 2000\,\mathrm{Hz}$.

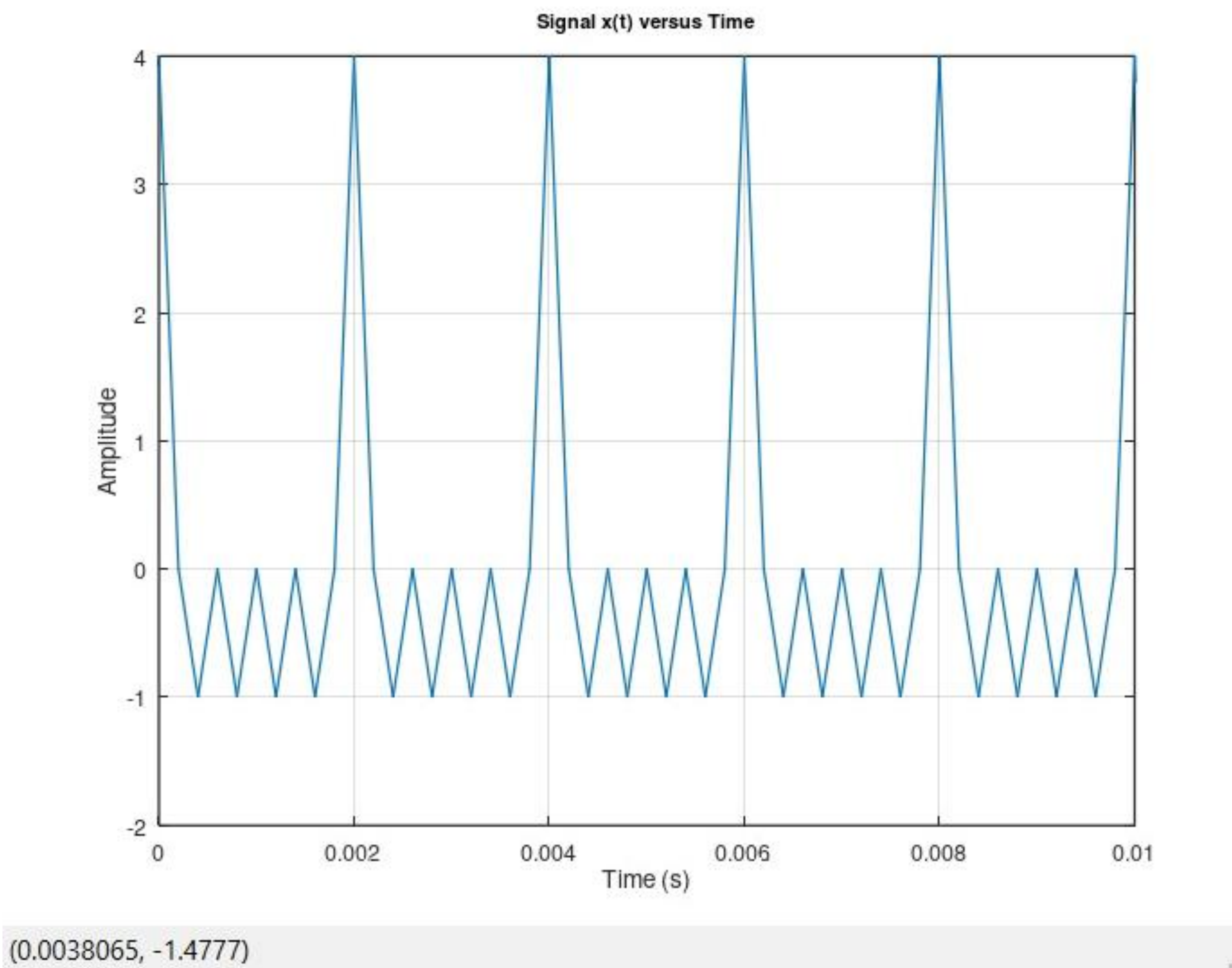
Plot of `x(t)` vs time:

step 3



To make the Graph more Clear we made a line of code to zoom on the graph:

```
xlim([0 0.01]); % Zoom in to see the oscillations
grid on; % Enable grid
```

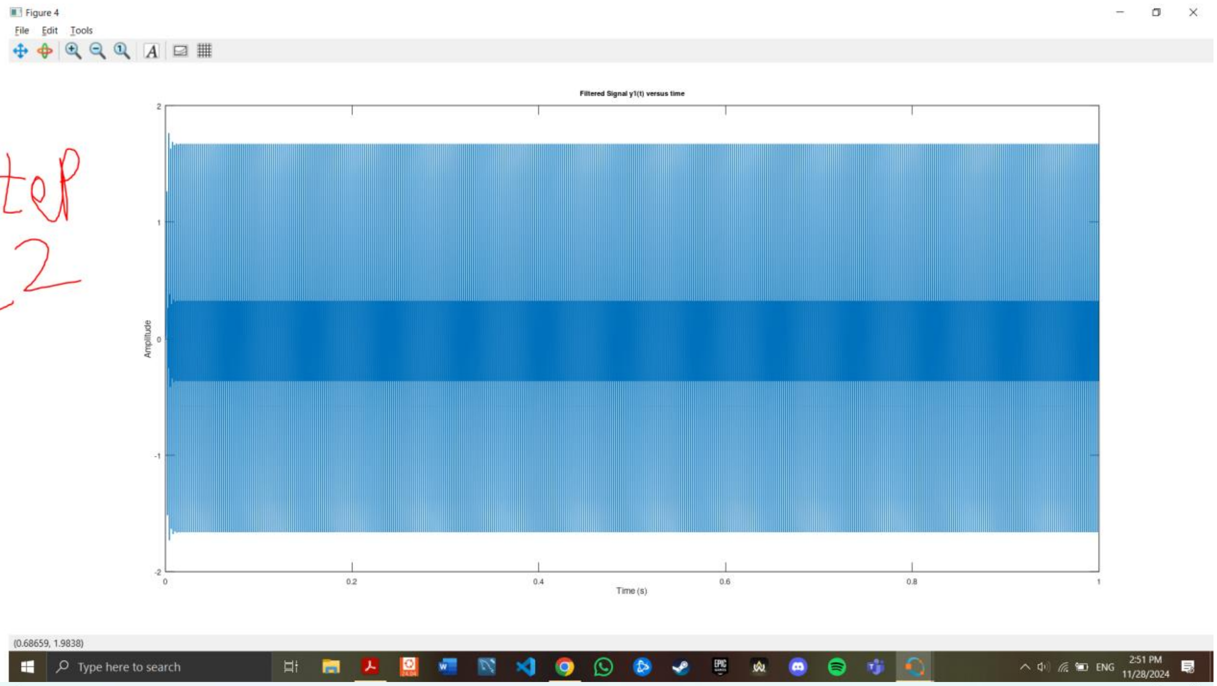


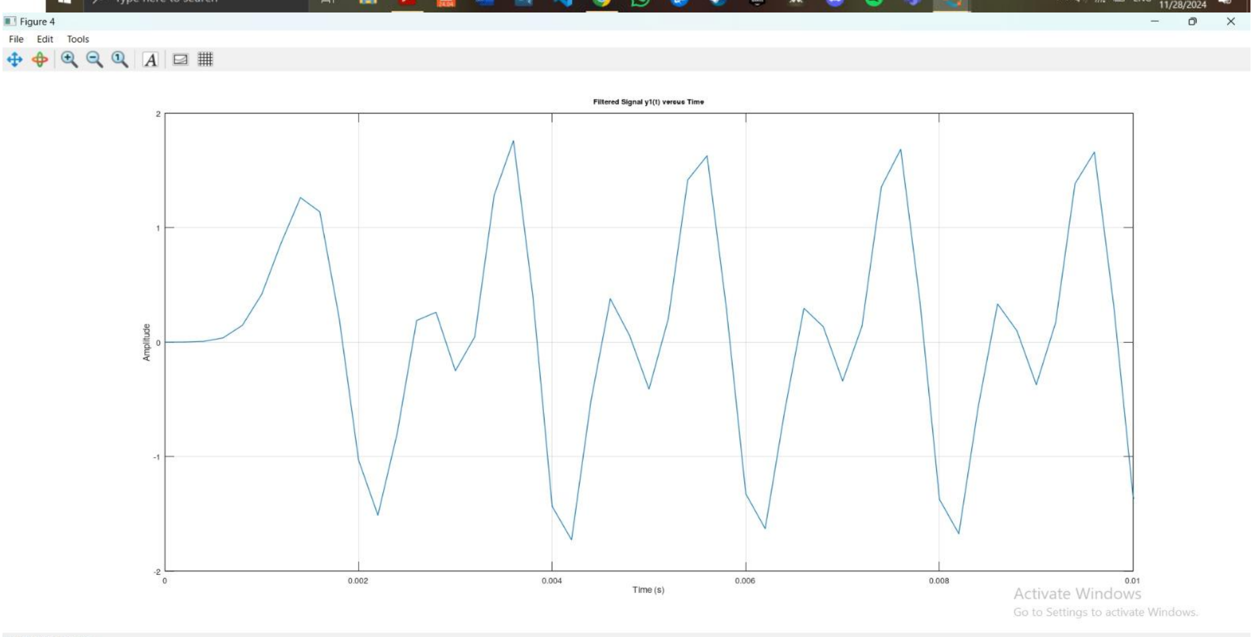
Low-Pass Filtered Signal (`y1(t)`)

After passing the original signal through a Butterworth low-pass filter with a cutoff frequency of 1.25 kHz, the resulting signal was denoted as `y1(t)`.

- **Plot of `y1(t)` vs time**:

step 12



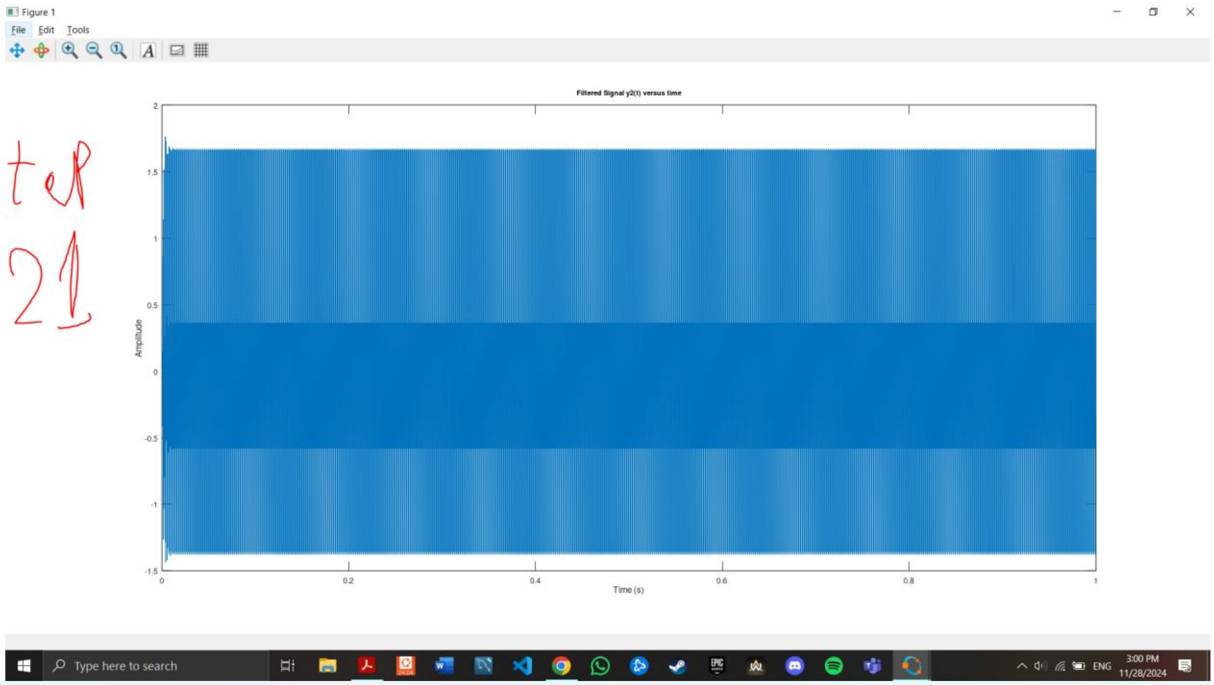


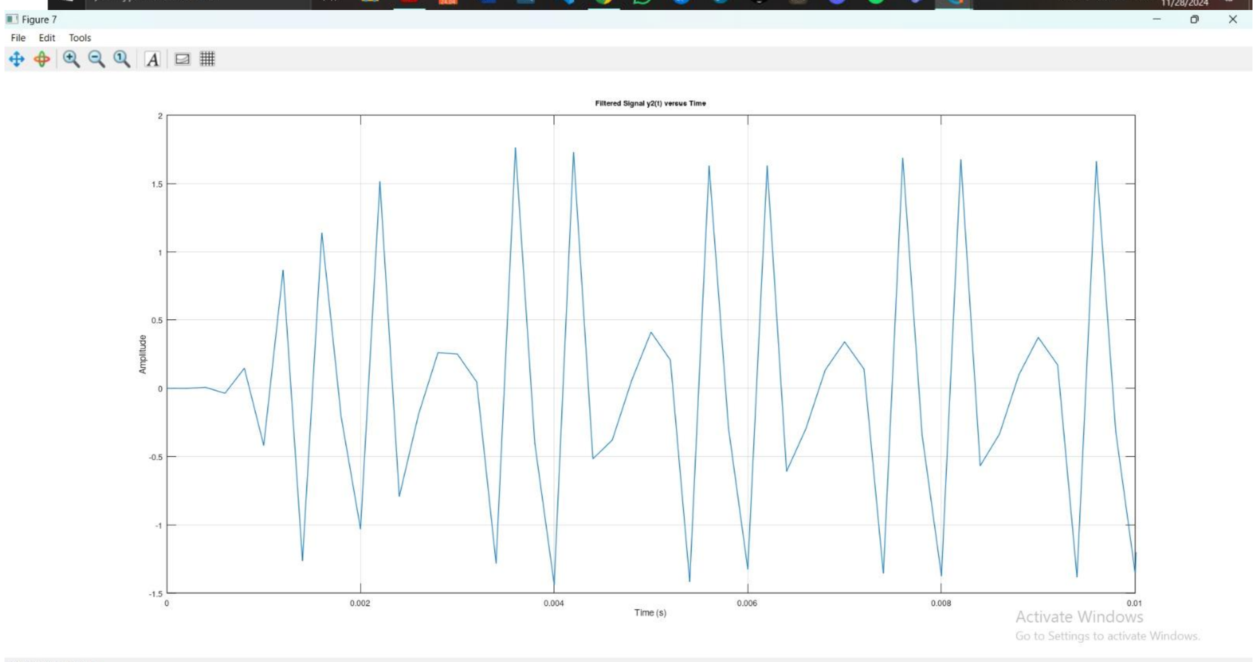
High-Pass Filtered Signal (`y2(t)`)

The high-pass filtered signal, denoted `y2(t)`, was obtained by passing the original signal through a Butterworth high-pass filter with a cutoff frequency of 1.25 kHz.

- **Plot of `y2(t)` vs time**:

step 21





3. Energy Computation

The energy of a signal is computed using the formula:

$$E = \int_{-\infty}^{\infty} |x(t)|^2 \, dt$$

where x(t)x(t) is the signal.

Energy of the Original Signal (`x(t)`)

The energy of the original signal `x(t)` was computed using the formula above, and the result was:

- **Energy of `x(t)`:**

Energy of the Low-Pass Filtered Signal (`y1(t)`)

Similarly, the energy of the low-pass filtered signal `y1(t)` was calculated.

- **Energy of `y1(t)`:**

Energy of the High-Pass Filtered Signal (`y2(t)`)

The energy of the high-pass filtered signal `y2(t)` was also calculated.

- **Energy of `y2(t)`:**

```
>> signal_processing
Energy of x(t): 1.9984
Energy from frequency spectrum: 10000
Energy of y1(t): 0.99864
Energy from frequency spectrum of y1(t): 4993.2487
Energy of y2(t): 0.99864
Energy from frequency spectrum of y2(t): 4993.2449
>>
```

4. Frequency Spectra

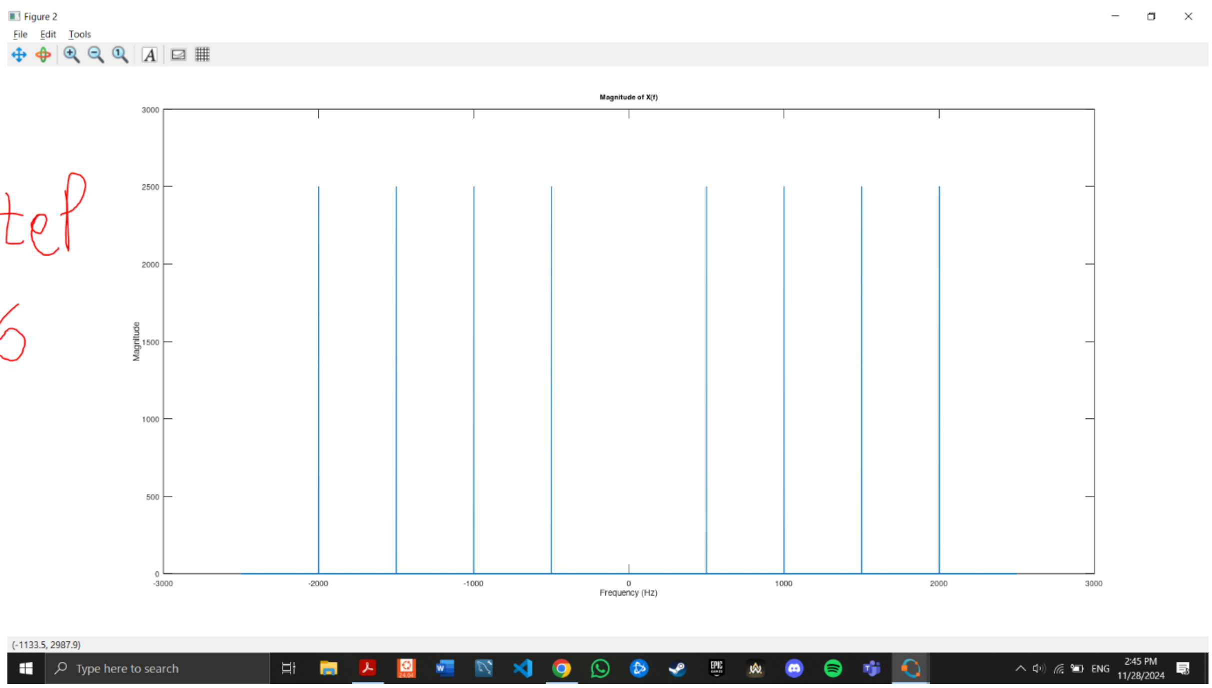
The frequency spectrum of a signal provides insight into the distribution of its energy across different frequencies. The magnitude of the Fourier transform of each signal was plotted.

Frequency Spectrum of `x(t)`

```
% Step 5: Compute the frequency spectrum X(f)
X_f = fft(x_t);
f_range = (-fs/2):(fs/length(x_t)):(fs/2 - fs/length(x_t));
X_f_shifted = fftshift(X_f); % Shift to center zero frequency
```

- **Plot of magnitude of `X(f)`:**

step 6

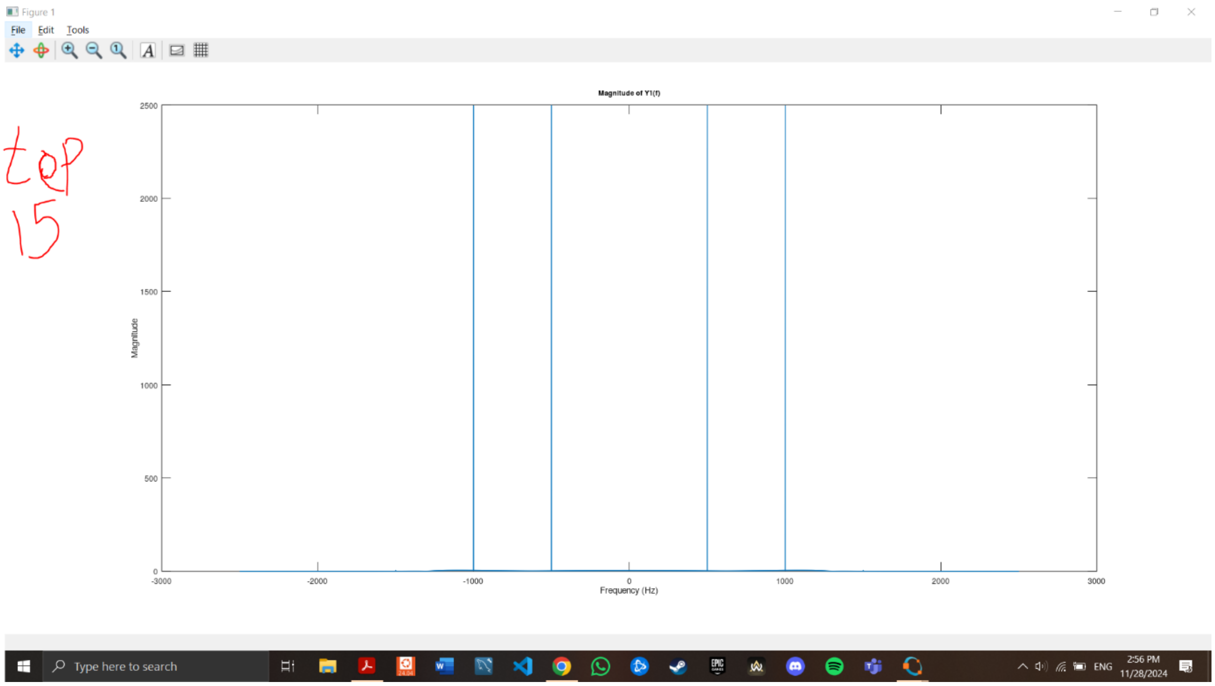


Frequency Spectrum of `y1(t)`

```
% Step 14: Compute the frequency spectrum Y1(f)
Y1_f = fft(y1_t);
Y1_f_shifted = fftshift(Y1_f); % Shift to center zero frequency
```

- **Plot of magnitude of `Y1(f)`:**

stop
15

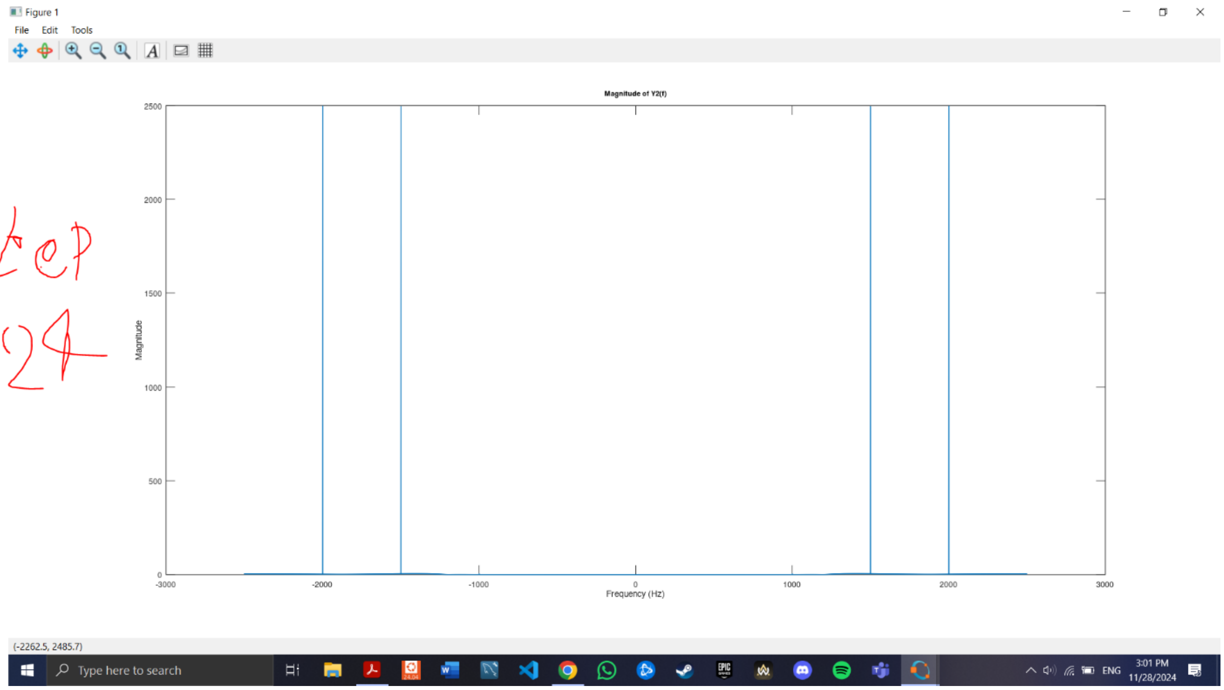


Frequency Spectrum of `y2(t)`

```
% Step 23: Compute the frequency spectrum Y2(f)
Y2_f = fft(y2_t);
Y2_f_shifted = fftshift(Y2_f); % Shift to center zero frequency
```

- **Plot of magnitude of `Y2(f)`:**

Step 24



5. Audio Files

The audio files corresponding to each signal were generated. These files can be played to hear the effect of the filters on the original signal.

Original Signal Audio (`x_t.wav`)

- **Audio file link**: https://drive.google.com/drive/folders/1fMjHXH08O6O2U9pGz2sKQ7vaQjfaCYFD?usp=sharing

Low-Pass Filtered Audio (`y1_t.wav`)

- **Audio file link**: https://drive.google.com/drive/folders/1fMjHXH08O6O2U9pGz2sKQ7vaQjfaCYFD?usp=sharing

High-Pass Filtered Audio (`y2_t.wav`)

- **Audio file link**: https://drive.google.com/drive/folders/1fMjHXH08O6O2U9pGz2sKQ7vaQjfaCYFD?usp=sharing

6. Plots and Analysis

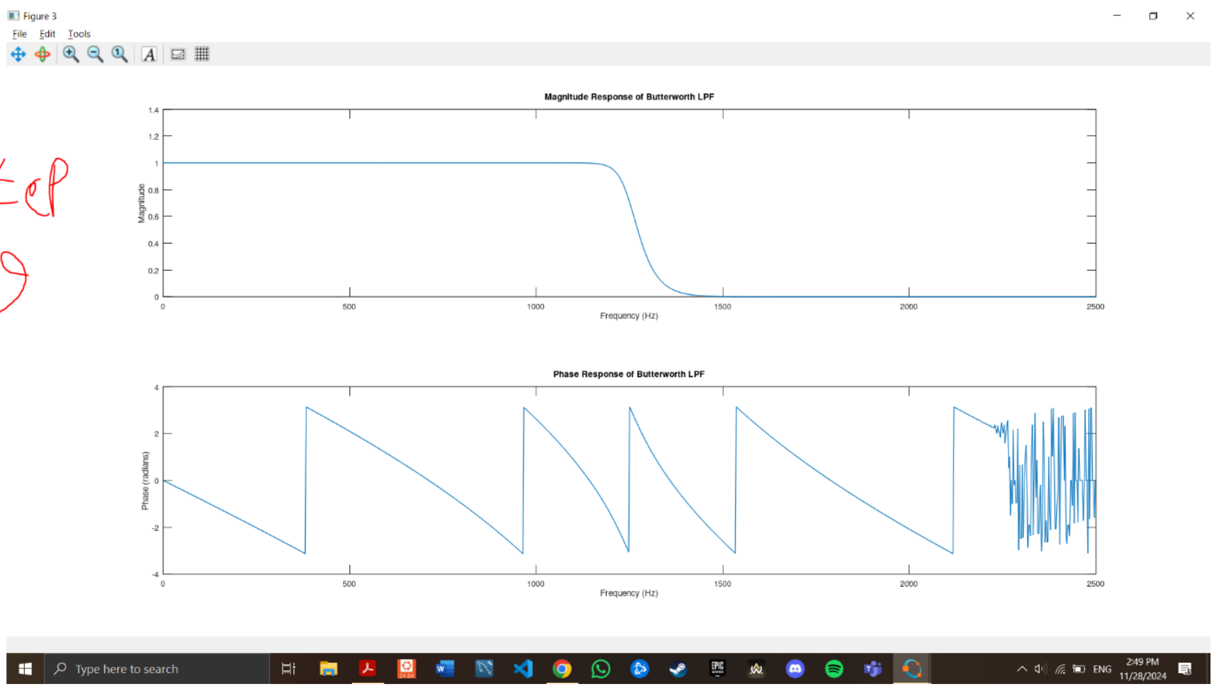
Magnitude and Phase Response of the Low-Pass Filter

The magnitude and phase response of the Butterworth low-pass filter were plotted to show the frequency characteristics of the filter.

```
% Step 8: Design a Butterworth low-pass filter (order 20, cutoff 1.25 kHz)
low_cutoff = 1250; % Cutoff frequency 1.25 kHz
[b, a] = butter(20, low_cutoff/(fs/2), 'low'); % Butterworth filter design
[b, a] = butter(20, low_cutoff/(fs/2), 'low'); % Butterworth filter design
```

- **Magnitude Response of Low-Pass Filter**:
- **Phase Response of Low-Pass Filter**:

step 9



Magnitude and Phase Response of the High-Pass Filter

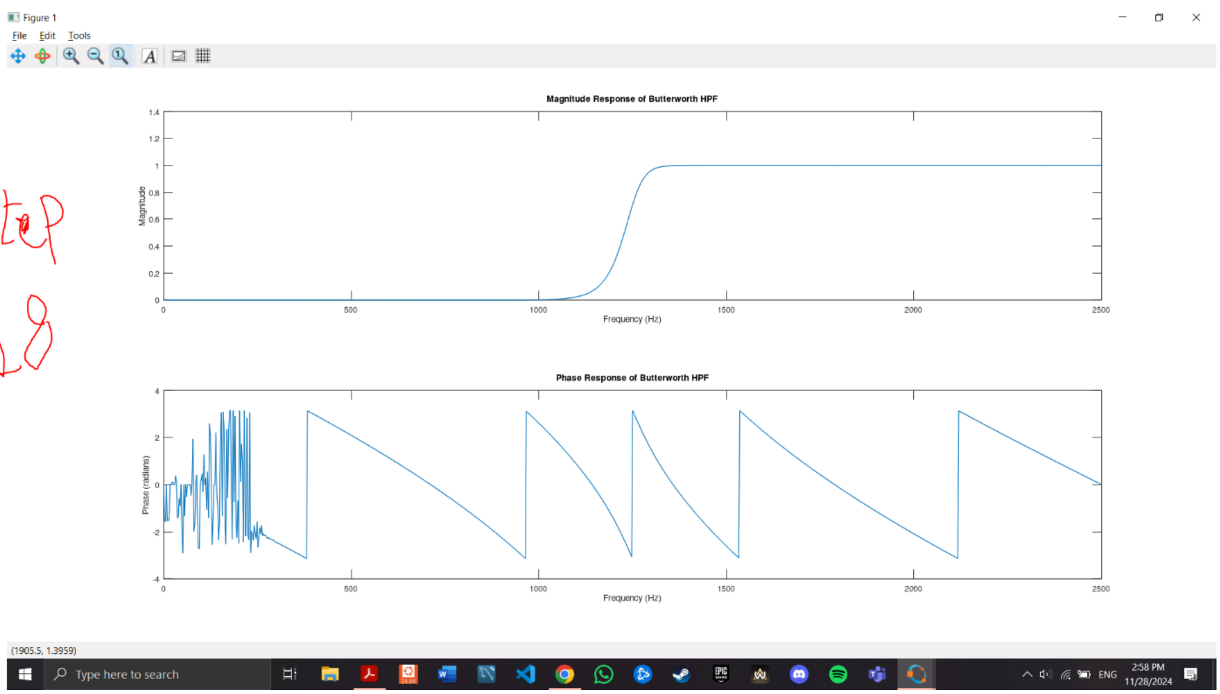
Similarly, the magnitude and phase response of the Butterworth high-pass filter were plotted.

```
% Step 16: Verify Parseval's theorem for y1(t) from Y1(f)
energy_from_spectrum_y1 = trapz(f_range, abs(Y1_f_shifted).^2) / fs;
disp(['Energy from frequency spectrum of y1(t): ', num2str(energy_from_spectrum_y1)]);

% Step 17: Design a Butterworth high-pass filter (order 20, cutoff 1.25 kHz)
[b_hp, a_hp] = butter(20, low_cutoff/(fs/2), 'high'); % High-pass filter design
```

- **Magnitude Response of High-Pass Filter**:
- **Phase Response of High-Pass Filter**:

step 18



7. Conclusion

In this project, we successfully designed and implemented Butterworth low-pass and high-pass filters. The original signal was processed, and the effects of filtering were observed in both the time and frequency domains. The energy values of the original and filtered signals were computed, providing insight into how the filtering affected the signal's power. Audio files corresponding to the original and filtered signals were generated, demonstrating the auditory differences.

The results show that the low-pass filter attenuates high-frequency components, while the high-pass filter removes low-frequency components, which is consistent with the theoretical behavior of these filters. The project successfully met its goals of understanding and applying signal filtering techniques using Butterworth filters.