Lab7 Report (Digital Filter Design)

12111008 屠耘诚

***Abstract*—This report delves into the design and implementation of digital filters, with a primary focus on FIR and IIR filters. It examines their applications in audio signal processing, particularly in enhancing speech signals and separating modulated signals from background noise. The report presents the theoretical underpinnings of these filters, their frequency responses, and the impact of different parameters on their performance. Practical implementations are demonstrated using MATLAB, showcasing the efficacy of these filters in manipulating audio signals. The use of lowpass filters and the truncation technique in designing FIR filters to improve the signal-to-noise ratio in audio processing is also explored. The findings highlight the crucial role of digital filters in modern audio signal processing applications.**

Keywords*—Digital Filters, FIR (Finite Impulse Response) Filter, IIR (Infinite Impulse Response) Filter, Audio Signal Processing, Frequency Response, Signal Interference, Lowpass Filter, Sinc Function, magnitude response, Noise

I. Introduction

In the evolving landscape of digital signal processing, the design and application of digital filters have become increasingly significant. This report aims to provide a comprehensive understanding of Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, two pivotal types of digital filters widely used in audio signal processing. The objective is to explore the theoretical principles underlying these filters, their frequency responses, and the various parameters influencing their performance. Practical aspects are addressed through MATLAB implementations, demonstrating the filters' effectiveness in manipulating audio signals for specific purposes. This includes the enhancement of speech signals and the separation of modulated signals from background noise. The exploration extends to the use of lowpass filters and the truncation technique in FIR filter design, highlighting their impact on improving the signal-to-noise ratio. This report is structured to provide insights into the versatility and capabilities of digital filters, offering a bridge between theoretical concepts and real-world applications in audio signal processing.

II. Background on Digital Filters

Digital signal processing often involves filtering to modify the amplitudes of certain frequency components or remove unwanted frequencies from a signal. The key tools for this are the z-transform and the discrete-time Fourier Transform (DTFT). Besides, Filters are defined by a difference equation, involving coefficients that parameterize the filter, and can be categorized as either infinite impulse response (IIR) or finite impulse response (FIR) based on their response duration. The z-transform is crucial for analyzing filter frequency responses, and a filter's transfer function is typically a ratio of polynomials.

III. 7.3 Design of a Simple FIR Filter

A. 7.3.1 $H_f(z)$ with three values of θ

To design this simple second-order and real-valued FIR (Finite Impulse Response) filter $H_f(z)$ and analyze its characteristics, we'll follow the steps outlined.

1) Deriving the Difference Equation

We will design a simple second order FIR filter with the two zeros on the unit circle.

Given the transfer function:

$$H_f(z) = (1 - z_1 z^{-1})(1 - z_2 z^{-1})$$

In order for the filter's impulse response to be real-valued, the two zeros must be complex conjugates of one another:

$$z_1 = e^{j\theta},\ z_2 = e^{-j\theta}$$

$$H_f(z) = (1 - e^{j\theta} z^{-1})(1 - e^{-j\theta} z^{-1})$$

which simplifies to

$$H_f(z) = 1 - 2cos(\theta) z^{-1} + z^{-2}$$

The difference equation relates the input $x[n]$ to the output $y[n]$ and can be derived from the transfer function. By replacing z^{-1} with the shift operator (which shifts the signal by one sample), the difference equation is

$$\boldsymbol{y[n] = x[n] - 2cos(\theta) x[n-1] + x[n-2]}$$

2) System Diagram

Figure 1 shows the system diagram of the impulse response for the filter $H_f(z)$

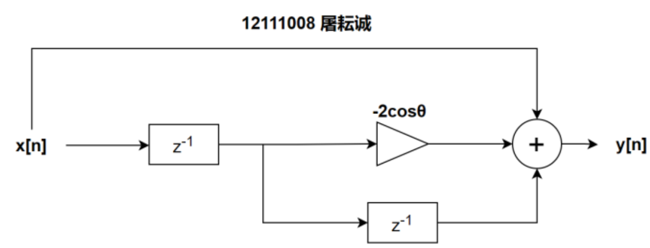

Fig. 1. system diagram of the impulse response for the filter $H_f(z)$

3) Analytical Expression of the Impulse Response h[n]

For this FIR filter, the impulse response is finite and equals the coefficients of the transfer function:

$$h[n] = \{1, -2cos(\theta), 1\}$$

for $n = 0,1,2$ respectively, and 0 otherwise.

4) Magnitude of the Frequency Response

We need to compute and plot$|\ H_f(e^{j\omega})\ |$ for $-\pi < \omega < \pi$ and for $\theta = \pi/6, \pi/3, \pi/2$, which is shown in Figure 2

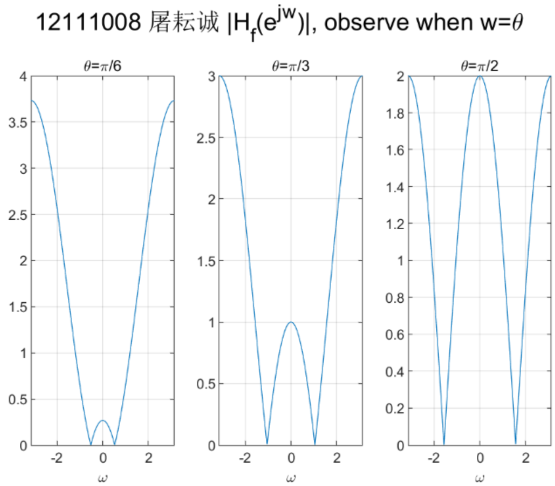

Fig. 2. Magnitude of the $H_f(e^{j\omega})$ for the three values of θ

5) *Analysis of the Effect of θ*

The plots will show how the location of the zeros in the frequency response $H_f(e^{j\omega})$ (which corresponds to θ) affects the filter's ability to pass or attenuate different frequencies.

For each value of θ, the filter will exhibit nulls (zero response) at the frequencies $\pm\theta$, effectively blocking sinusoidal components at these frequencies.

As θ changes, these nulls move along the frequency axis, demonstrating how the filter's frequency-selective nature can be controlled by the placement of its zeros. By adjusting θ, we can block unwanted frequency components.

B. *7.3.2 FIR Filter Application in Audio Signal Processing*

Now we continue with this FIR filter with two zeros on the unit circle, with the aim of removing specific frequency components associated with the interference from an audio signal.

1) *Time Domain Plot Analysis*

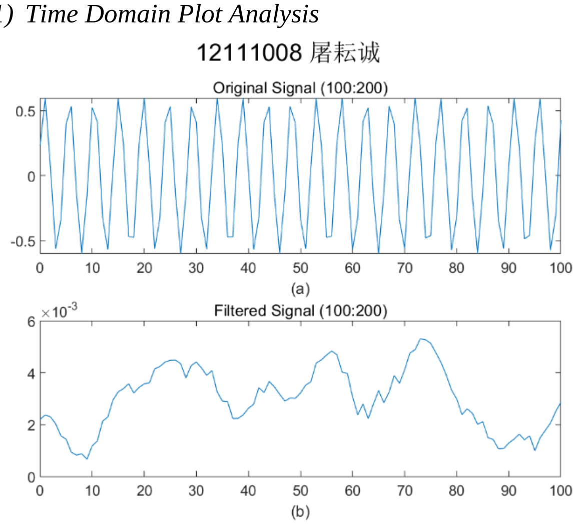

Fig. 3. Time Domain of Original Signal (a) and Filtered Signal (b) for 101 Samples

- Original Audio Signal: Figure 3 (a) shows 101 samples of the original audio signal in the time domain, covering the indices (100:200). This plot reveals the presence of sinusoidal interference superimposed on the speech signal.
- Filtered Audio Signal: Figure 3 (b) displays 101 samples of the filtered audio signal for the same indices. After filtering, high-frequency signals have been removed, resulting in a smoother waveform, making the underlying speech easier to recognize.

2) *Magnitude of the DTFT Analysis*

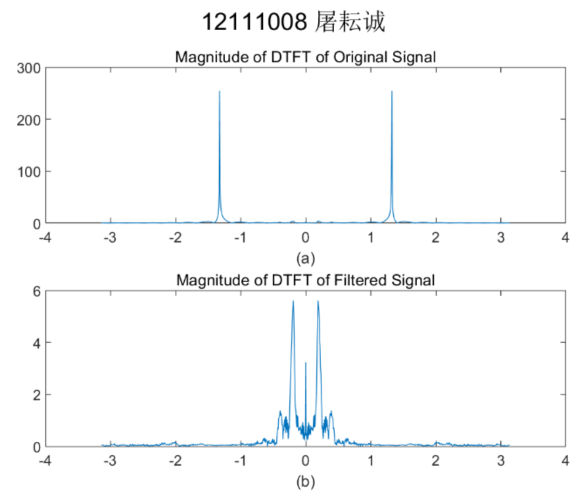

Fig. 4. Plot the Magnitude of the DTFT of Original Signal (a) and Filtered Signal (b) for 1001 Samples

- Original Audio Signal: Figure 4 (a) illustrates the magnitude of the DTFT for 1001 samples of the original audio signal (indices 100:1100). Two prominent peaks are visible, corresponding to the frequency of the sinusoidal interference.
- Filtered Audio Signal: Figure 4 (b) shows the magnitude of the DTFT for the same sample range of the filtered audio signal. The filtering process significantly reduces the magnitude at the interference frequencies, as evidenced by the diminished peaks.

3) *FIR Filter Function Code:*

The FIRfilter function, implemented in MATLAB, performs the filtering operation using convolution with the designed filter coefficients.

```
function y = FIRfilter(x)
    [X,w] = DTFT(x,0);
    [Xmax,Imax] = max(abs(X));
    h = [1 -2*cos(w(Imax)) 1];
    y = conv(x,h);
end
```

4) *Frequency Content Changes*

The filtering process effectively removes high-frequency noise components, making the speech signal more prominent. The amplitude variations in the time-domain waveform become smoother, indicating a reduction in interference.

5) *Type of Filter Used*

Based on the spectral analysis and filtering effects, the employed FIR filter acts as a bandstop filter. It primarily removes the frequency components at and around the measured value of θ, thus attenuating frequencies corresponding to the sinusoidal interference.

6) Audio Quality Changes

- Before Filtering: The original audio is dominated by a loud single-frequency noise, making the speech component almost inaudible.
- After Filtering: The filtering process successfully removes the noise, leaving behind a clear speech signal. The result is a much more intelligible and cleaner audio output, where the phrase "**Please get rid of this beep**" is distinctly audible.

IV. 7.4 Design of a Simple IIR Filter

A. 7.4.1 $H_i(z)$ with three values of r

To design a simple second-order IIR (Infinite Impulse Response) filter $H_i(z)$ and analyze its characteristics, we'll follow the steps outlined.

1) Deriving the Difference Equation

We will design a simple second order IIR filter with complex-conjugate poles:

The poles have the form

$$p_1 = re^{j\theta},\ p_2 = re^{-j\theta}$$

Given the transfer function

$$H_i(z) = \frac{1-r}{(1-p_1z^{-1})(1-p_2z^{-1})} = \frac{1-r}{(1-re^{j\theta}z^{-1})(1-re^{-j\theta}z^{-1})}$$

which simplifies to

$$H_i(z) = \frac{1-r}{1-2r\cos(\theta)z^{-1}+r^2z^{-2}}$$

Stability condition: $|r| < 1$, which makes poles all inside unit circle.

The difference equation relates the input $x[n]$ to the output $y[n]$ can be derived from the transfer function:

$$\boldsymbol{y[n] = (1-r)x[n] + 2r\cos(\theta)y[n-1] - r^2y[n-2]}$$

2) System Diagram

Figure 5 shows the system diagram of the impulse response for the filter $H_i(z)$

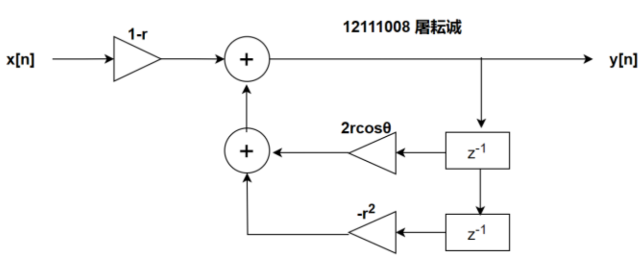

Fig. 5. system diagram of the impulse response for the filter $H_i(z)$

3) Analytical Expression of the Impulse Response h[n]

The impulse response h[n] of an IIR filter is of infinite duration. It is not straightforward to express h[n] analytically for IIR filters, but we can use difference equation.

$$\boldsymbol{h[n] = (1-r)\delta[n] + 2r\cos\theta h[n-1] - r^2h[n-2]}$$

4) Magnitude of the Frequency Response

We need to compute and plot $|H_i(e^{j\omega})|$ for $-\pi < \omega < \pi$ and $\theta = \pi/3$ and for $r = 0.99, 0.9, 0.7$, which is shown in Figure 6

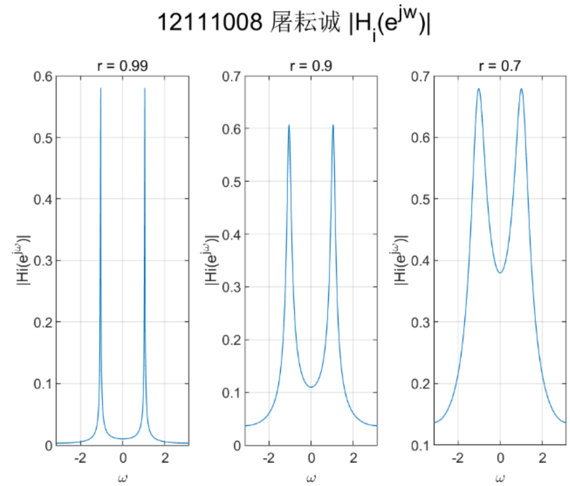

Fig. 6. Magnitude of the $H_i(e^{j\omega})$ for the three values of r

5) Effect of r on Magnitude Response

The value of r affects the sharpness and bandwidth of the filter's response. As r approaches 1, the poles get closer to the unit circle, making the filter's bandwidth narrower and its response at θ more pronounced, which makes the response nearly an ideal impulse at θ.

B. 7.4.2 Separate a Modulated Sinusoidal Signal from Background Noise

Now we continue with the IIR filter $H_i(z)$ to separate a modulated sinusoidal signal from background noise in the pcm audio signal.

1) Time Domain Plot Analysis

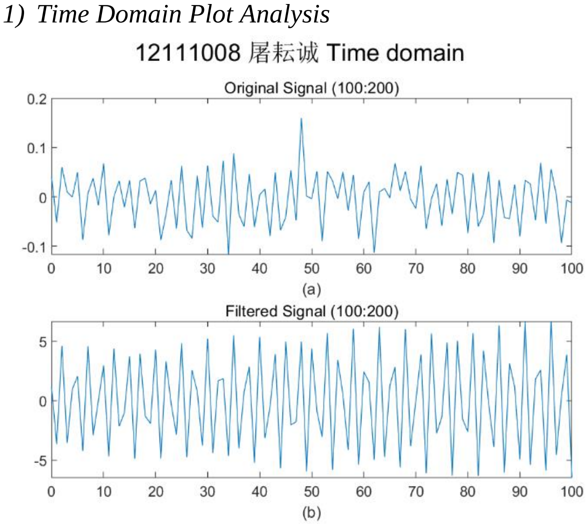

Fig. 7. (a) The Time-Domain Plot of 101 Samples of the Original Pcm Signal (indices 100-200) (b) The Time-Domain Plot of 101-sample plot of the Filtered Signal (indices 100-200)

- Original Audio Signal: Figure 7 (a) shows the presence of both the modulated sinusoidal signal and the background noise.
- Filtered Audio Signal: Figure 7 (b) After applying the IIRfilter, the 101-sample plot of the filtered signal demonstrates how the filter amplifies the modulated sinusoid while attenuating the background noise.

2) *Magnitude of the DTFT Analysis*

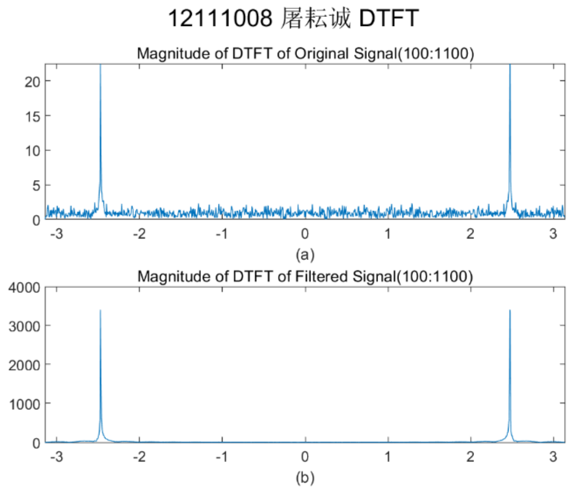

Fig. 8. Plot the Magnitude of the DTFT of Original Signal (a) and Filtered Signal (b) for 1001 Samples

- Original Audio Signal: Figure 8 (a) The DTFT magnitude plot of 1001 samples of the pcm signal reveals two peaks corresponding to the center frequency of the modulated signal, amidst wideband low-amplitude background noise.
- Filtered Audio Signal: Figure 8 (b) The DTFT magnitude plot of the filtered signal shows a pronounced peak at the center frequency, indicating effective amplification of the modulated signal and suppression of noise.

3) *DTFT Magnitude in the Narrow Frequency Range*

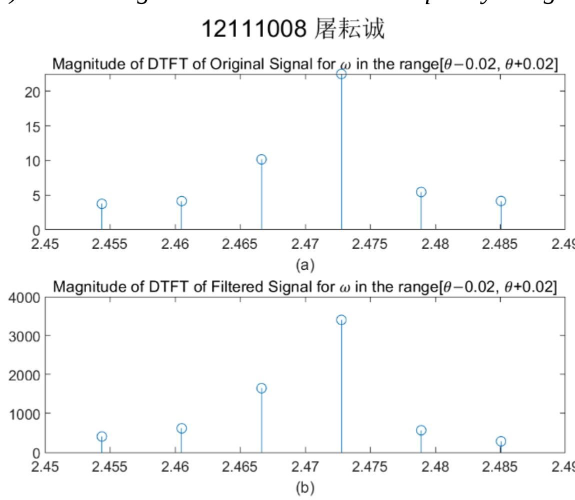

Fig. 9. Magnitude of DTFT of Original Signal (a) and Filtered Signal (b) for ω in the range$[\theta - 0.02, \theta + 0.02]$

The DTFT magnitude plots in the narrow range around θ (i.e., $\theta \pm 0.02$) for both the original and filtered signals. The filtered signal exhibits a more distinct peak at θ, demonstrating the filter's ability to isolate the modulated signal from the noise.

4) *FIR Filter Function Code:*

The IIRfilter function implements the IIR filter using a recursive difference equation $y[n] = (1 - r)x[n] + 2rcos(\theta)y[n - 1] - r^2y[n - 2]$.

Given θ and r, the function applies the IIR filter to an input signal x and outputs the filtered signal y. The code for the IIRfilter function is as follows:

```
function y = IIRfilter(x)
  theta = (3146/8000)*2*pi;
  r = 0.995;
  N = length(x);
  y = zeros(1, N);
  y(1) = x(1);
  y(2) = x(2) + 2*r*cos(theta);
  for i = 3:N
  y(i) =x(i)+2*r*cos(theta).*y(i-1)-(r^2).*y(i-2);
  end
end
```

5) *Comments on Filter Effects and Parameter r*

- **Signal Characteristics**: Before filtering, the pcm signal contains both the desired modulated signal and background noise. After filtering, the modulated signal becomes more prominent, and the noise is significantly reduced.
- **Filter Type**: The IIR filter used is a narrow-band amplifier, focusing on the frequency around θ and thereby acting as a bandpass filter in this context.
- **Effect of r**: Increasing r closer to 1 narrows the bandwidth of the filter, making it more selective around θ. However, setting r too close to 1 (e.g., r=0.9999999) might result in an unstable filter and can amplify noise or other undesired frequencies close to θ.
- **Audio Quality**: The filtered audio signal is clearer, with the modulated signal standing out more distinctly against a reduced noise background.

V. 7.5 LOWPASS FILTER DESIGN PARAMETERS

VI. 7.6 FILTER DESIGN USING TRUNCATION

Now we will filter out background noise from the speech signal in **nspeech2** using lowpass FIR filters

A. *Filter Design:*

To create a realizable filter, the ideal impulse response is truncate. We also will make the filter causal, the impulse response is shifted by M units, which is given by:

$$h[n] = \frac{\omega_c}{\pi} sinc(\frac{\omega_c}{\pi}(n - \frac{N-1}{2})), n = 0,1, \dots, N - 1$$

This shift introduces a linear phase term in the frequency response but does not affect its magnitude.

- Now we will design a lowpass filter with a cutoff frequency $\omega_c = 2.0$ using the truncated sinc function above. This filter aims to separate the low-frequency speech component from the high-frequency noise.

- Implement the **LPFtrunc** function in MATLAB to generate the filter's impulse response for different filter sizes.

```
function response = LPFtrunc(N)
    for i=1:N
      response(i)=2/pi*sinc(2/pi*(i-1-(N-1)/2));
    end
end
```

B. Plots of Magnitude Response:

- Plot the magnitude response (not in decibels) for the two filters with N=21 and N=101. Mark the passband, transition band, and stopband on each plot in Figure 10.
- Also, plot the magnitude response in decibels for the two filters in Figure 11.

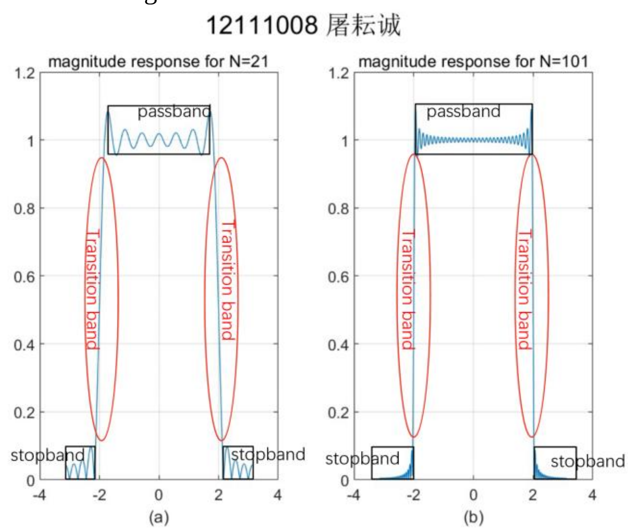

Fig. 10. Magnitude Response for the Two Filters (a) $N = 21$ (b) $N = 101$ not in dB

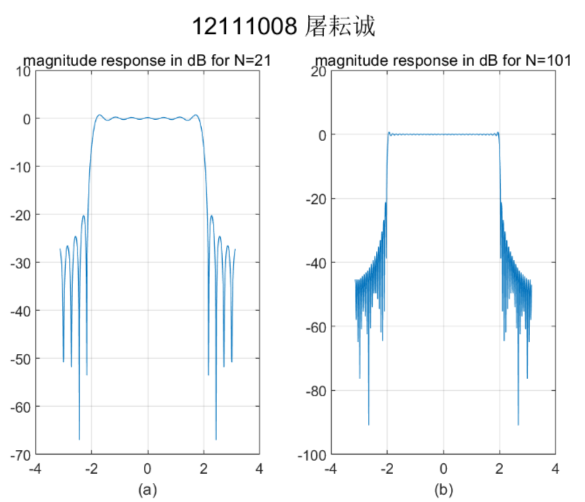

Fig. 11. Magnitude Response for the Two Filters (a) $N = 21$ (b) $N = 101$ in dB

The plots show how each filter attenuates frequencies above the cutoff. The transition from passband to stopband will be sharper for the larger-sized filter.

C. Effect of Filter Size on Stopband Ripple:

increase in filter size N results in more ripples in the stopband but with a lower magnitude, leading to better performance in attenuating unwanted frequencies..

D. Quality of Filtered Signals

Listen to both filtered signals and compare them with the unfiltered signal. We use command:

sound (conv (nspeech2, LPFtrunc(21)));

sound (conv (nspeech2, LPFtrunc(101)));

The filters isolate the speech component and suppress the background noise. The phrase "**If you can hear this clearly, you filter the signal correctly**" is now audible. But there are also slight distortions introduced by the filtering process.

Size effect: Generally, a larger filter size (N=101) may result in better noise reduction but may also affect the sharpness of the speech signal, which distort the speech signal slightly.

Anyways, overall, a larger size improves the restoration of speech signals and filters out noise, leading to better auditory results.

VII. Conclusion

This report has extensively explored the design and application of digital filters, specifically focusing on FIR (Finite Impulse Response) and IIR (Infinite Impulse Response) filters. The practical implementations demonstrated how these filters can effectively manipulate audio signals by removing unwanted frequencies or amplifying desired components. Key findings include the FIR filter's ability to eliminate high-frequency interference in audio signals, resulting in clearer speech, and the IIR filter's proficiency in amplifying a modulated sinusoidal signal while reducing background noise. Additionally, the exploration of lowpass filters using truncated sinc functions highlighted the impact of filter size on signal quality. Overall, the report underscores the versatility and effectiveness of digital filters in enhancing audio signal processing, offering valuable insights into their design and real-world applications.