Department of Electrical Engineering
Indian Institute of Technology Kharagpur

Digital Signal Processing Laboratory (EE39203)

Autumn, 2022-23

Experiment 5 **Digital Filter Design**

Slot: Date:

Student Name: Roll No.:

Grading Rubric

	Tick the best applicable per row			Points
	Below Expectation	Lacking in Some	Meets all Expectation	
Completeness of the report				
Organization of the report (5 pts) *With cover sheet, answers are in the same order as questions in the lab, copies of the questions are included in report, prepared in LaTeX*				
Quality of figures (5 pts) *Correctly labelled with title, x-axis, y-axis, and name(s)*				
Understanding and implementation of simple FIR filter (35 pts) *Difference eq., flow diagram, impulse response, plots of magnitude response, plots of original and filtered signals and their DTFT, matlab code, questions*				
Understanding and implementation of simple IIR filter (35 pts) *Difference eq., flow diagram, impulse response, plots of magnitude response, plots of original and filtered signals and their DTFT, matlab code, questions*				
Understanding parameters of lowpass filter design (20 pts) *Magnitude response plots with marked regions, questions.*				
			TOTAL (100 pts)	

Total Points (100): TA Name: TA Initials:

Digital Signal Processing Laboratory (EE39203)
Experiment 5: Digital Filter Design

Name: Sujay Vivek
Roll Number: 22EE30029

October 16th 2024

1. Learning Objective

In digital signal processing, filtering is a crucial process for modifying the relative amplitudes of a signal's frequency components or for removing undesired frequencies. Digital filters, which can be implemented as either finite impulse response (FIR) or infinite impulse response (IIR) filters, are widely used in various applications such as audio systems, where they allow for adjustments to bass and treble frequencies. FIR filters have a finite-duration response, as they involve only the present and past values of the input, whereas IIR filters have an impulse response that extends indefinitely due to feedback from past output values.

The design and analysis of digital filters involve both frequency and time-domain techniques. Filters are typically described in the frequency domain, using tools like the Z-transform and the discrete-time Fourier Transform (DTFT), but are implemented in the time domain with difference equations. The Z-transform, in particular, is essential for analyzing the frequency response and stability of filters. A general linear time-invariant (LTI) filter's behavior is governed by a rational transfer function, which represents the relationship between the input and output in terms of the filter's poles and zeros. By evaluating this transfer function on the unit circle in the complex plane, one can derive the filter's frequency response, enabling the design of filters suited to specific applications.

2. Design of a Simple FIR Filter

The design of a two-zero FIR filter with both zeros on the unit circle will be covered in this section. The two zeros in the filter must be complex conjugates of one another, with,in order for the impulse response to be real-valued. This filter's transfer function is given by:

$$H_f(z) = (1 - z_1 z^{-1})(1 - z_2 z^{-1}) = 1 - 2\cos(\theta) z^{-1} + z^{-2}$$

This corresponds to the difference equation:

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

The system diagram for this FIR filter is shown in Figure **??**.

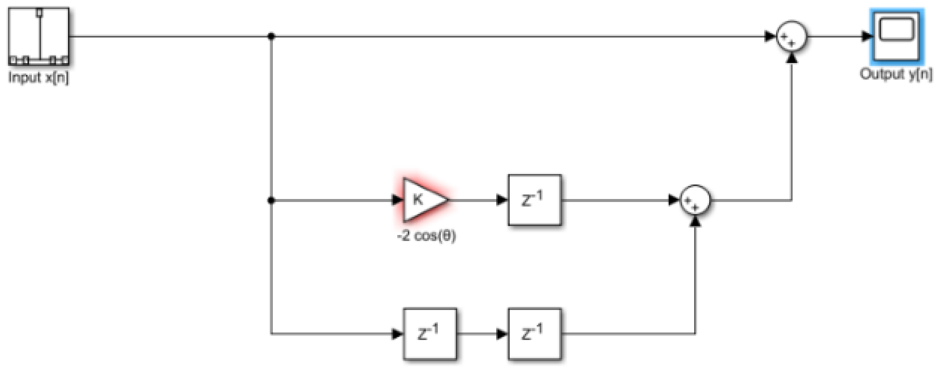


Figure 1: Magnitude Response of the Filter for $\theta = \pi/6, \pi/3, \pi/2$

The impulse response $h[n]$ of this filter can be obtained by applying an impulse input $\delta[n]$:

$$h[n] = \begin{cases} 1, & n = 0 \\ -2\cos(\theta), & n = 1 \\ 1, & n = 2 \\ 0, & \text{otherwise} \end{cases}$$

Therefore,

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

The magnitude of the filter's frequency response as a function of ω over the interval $-\pi < \omega < \pi$ is calculated and plotted using MATLAB for the following three values of θ: $\theta = \pi/6$ [BLUE], $\theta = \pi/3$ [ORANGE], and $\theta = \pi/2$ [YELLOW].

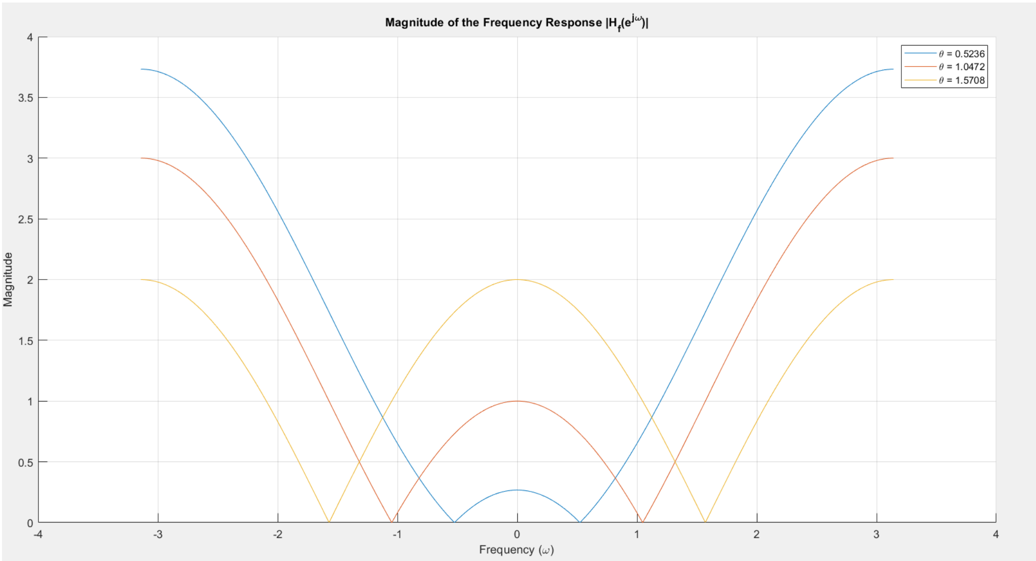
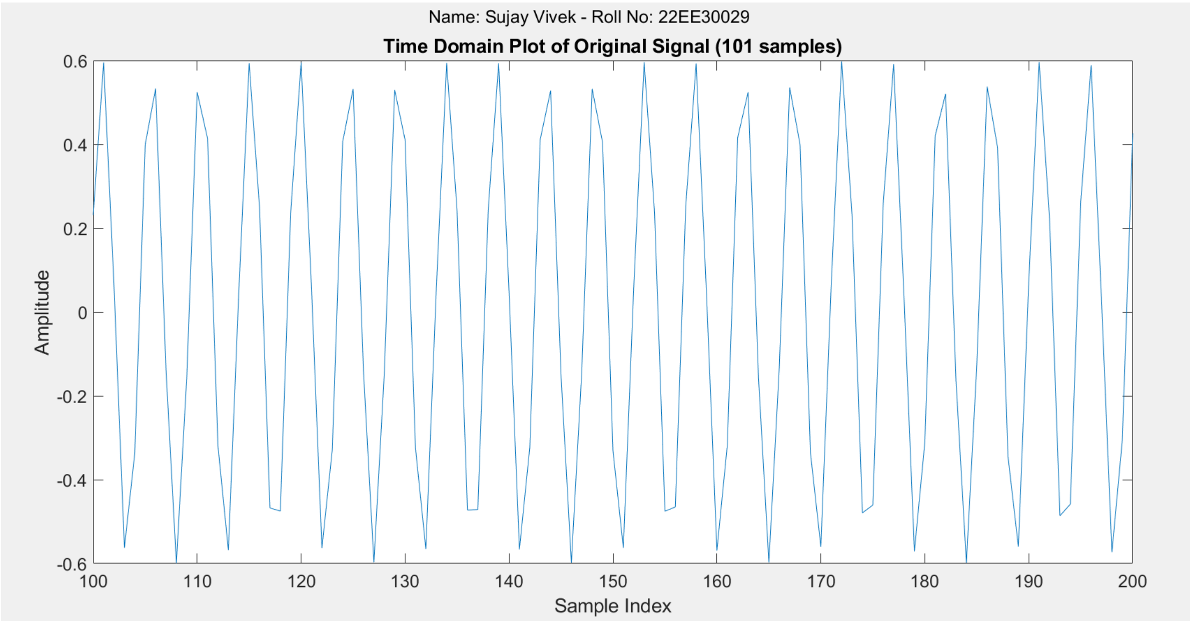


Figure 2: Magnitude Response of the Filter for $\theta = \pi/6, \pi/3, \pi/2$

As the value of θ increases, the magnitude response of the filter from the frequency range 0 to θ increases. In other words, an increase in θ improves the system's performance as a low-pass filter. Another observation is that when θ grows, the amplitude response at θ falls in value. The value of our three graphs moves from 3.73 to 3 to 2.

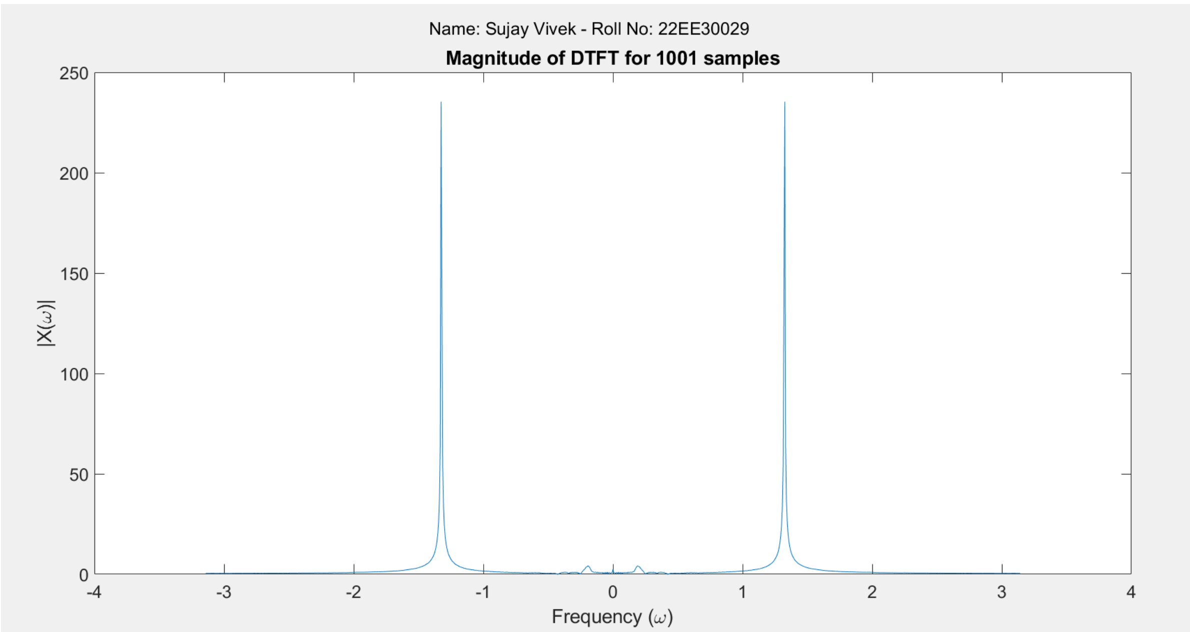
Audio Signal Analysis

Firstly, we plot 101 samples of the signal nspeech1.mat in MATLAB:



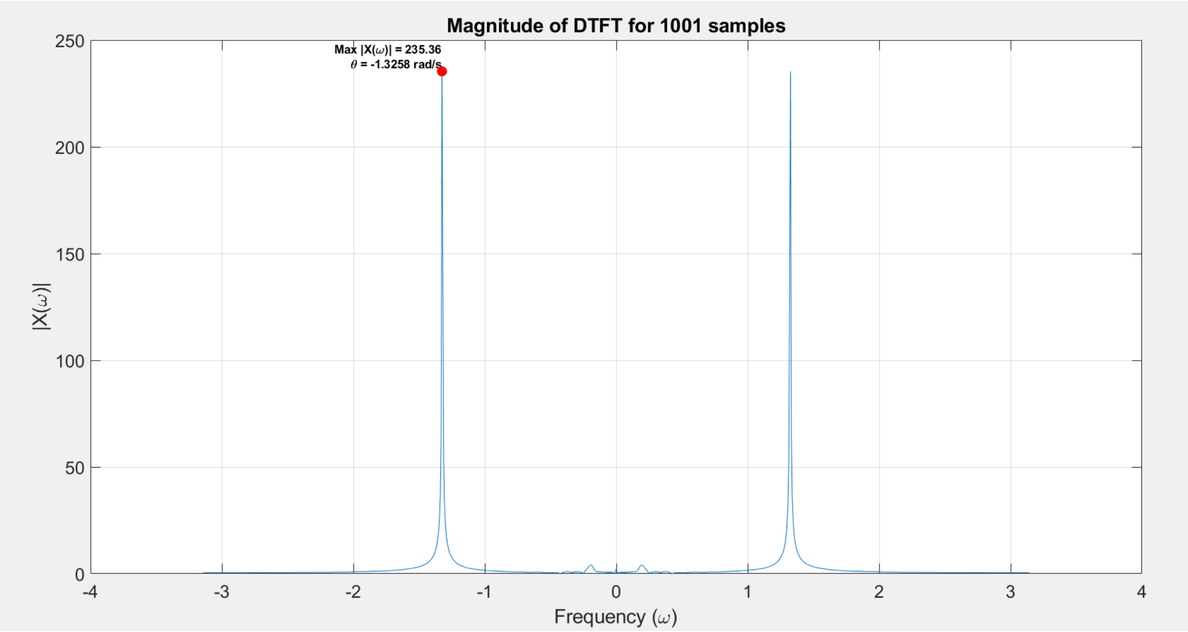
(a) Time Domain Plot of Original Signal

Next, we plot the magnitude of the DTFT samples versus frequency for $\pi < w < \pi$



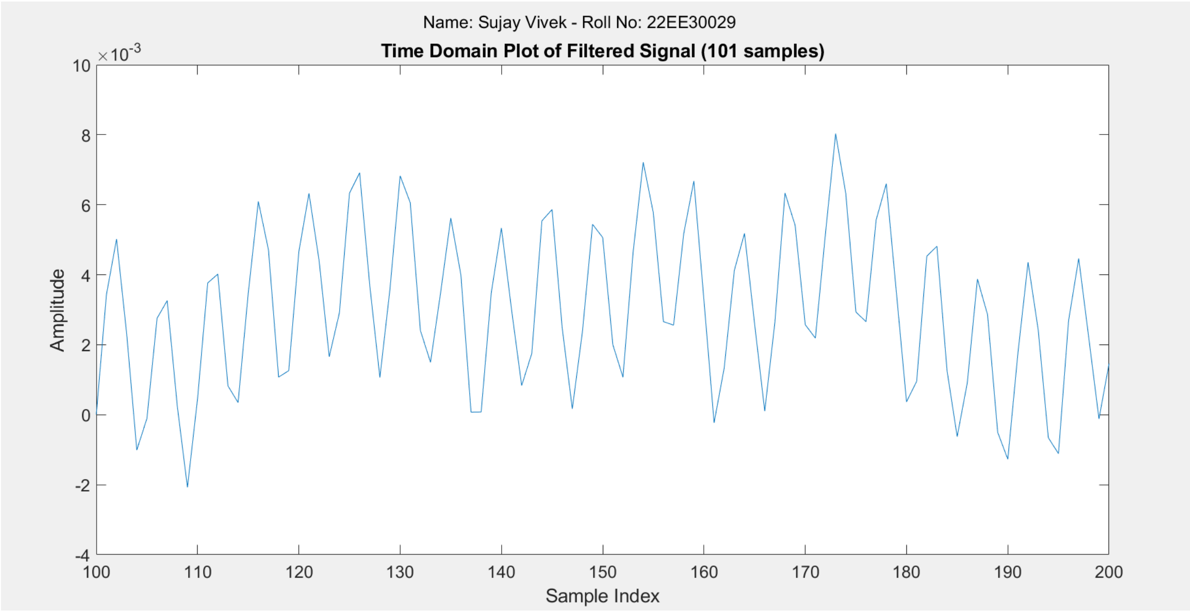
(a) Magnitude of DTFT for 1001 Samples

We also determine the index and magnitude of the largest value point in the DTFT magnitude plot. The frequency is calculated to be $= \pm 1.3258$ radians per second.



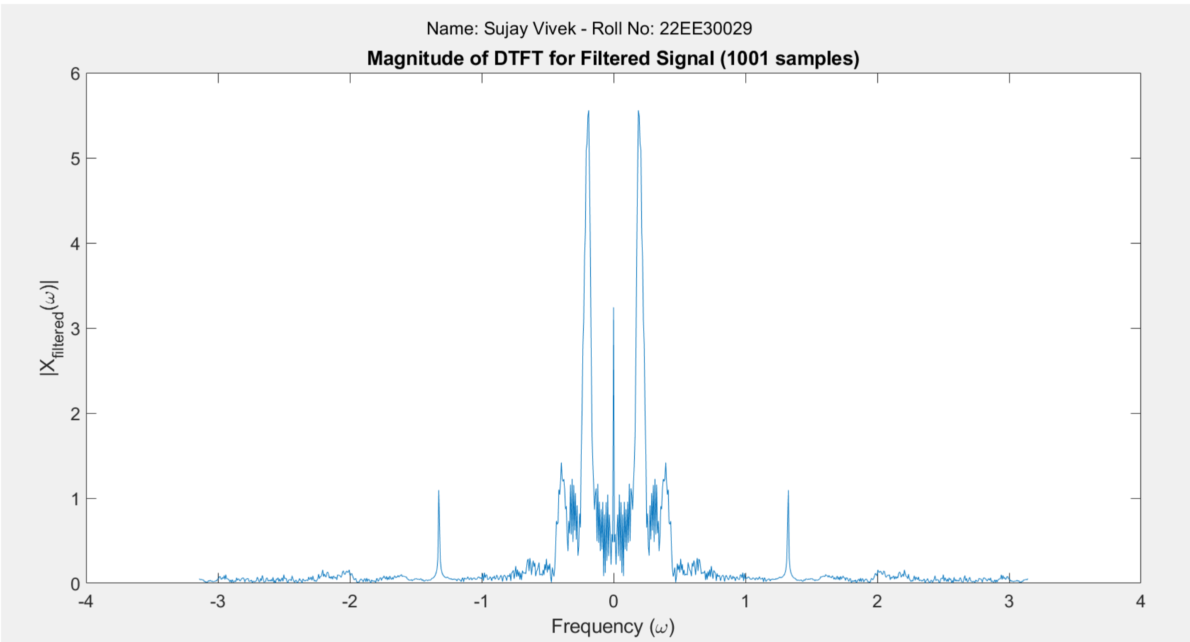
(a) Magnitude of DTFT for 1001 Samples along with Max Frequency

Then, using the measured value of θ, we create a Matlab function called FIRfilter(x) that applies the filter Hf(z) and outputs the filtered signal. The corresponding MATLAB code is provided below:



(a) Plot of 101 samples of filtered audio signal

Additionally, for the filtered signal, we plot the magnitude of the DTFT samples versus frequency for $\pi < w < \pi$.



(a) Magnitude of DTFT of Filtered Signal

After filtering, the frequency composition of the stream changed more towards lower frequencies. This is demonstrated by the sharp spikes acquired at frequency 0.19 rad/s, as well as the larger height of these peaks when compared to those obtained at Imax (=1.3258 rad/s). As a result, our FIR filter is a lowpass filter.

We can hear distinct speech after filtering if we play the generated audio signal in MATLAB. The person speaking has a lower pitch than the frequency of the original audio nspeech1.mat. This also demonstrates that our filter is a lowpass filter.

4. Design of a Simple IIR Filter

The transfer function for the second-order IIR filter with complex-conjugate poles is given by:

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

To find the difference equation, expand $H_i(z)$:

$$Y(z)\left(1-2r\cos(\theta)z^{-1}+r^2z^{-2}\right) = X(z)\cdot(1-r)$$

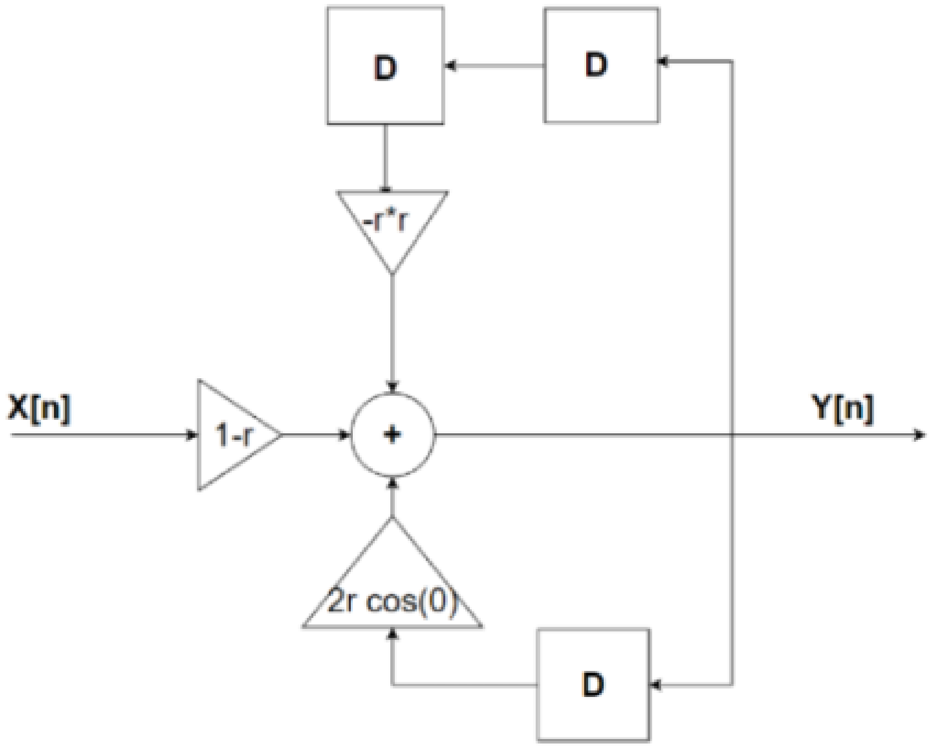
Expanding this equation into the time domain gives the difference equation:

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

Analytical Expression:

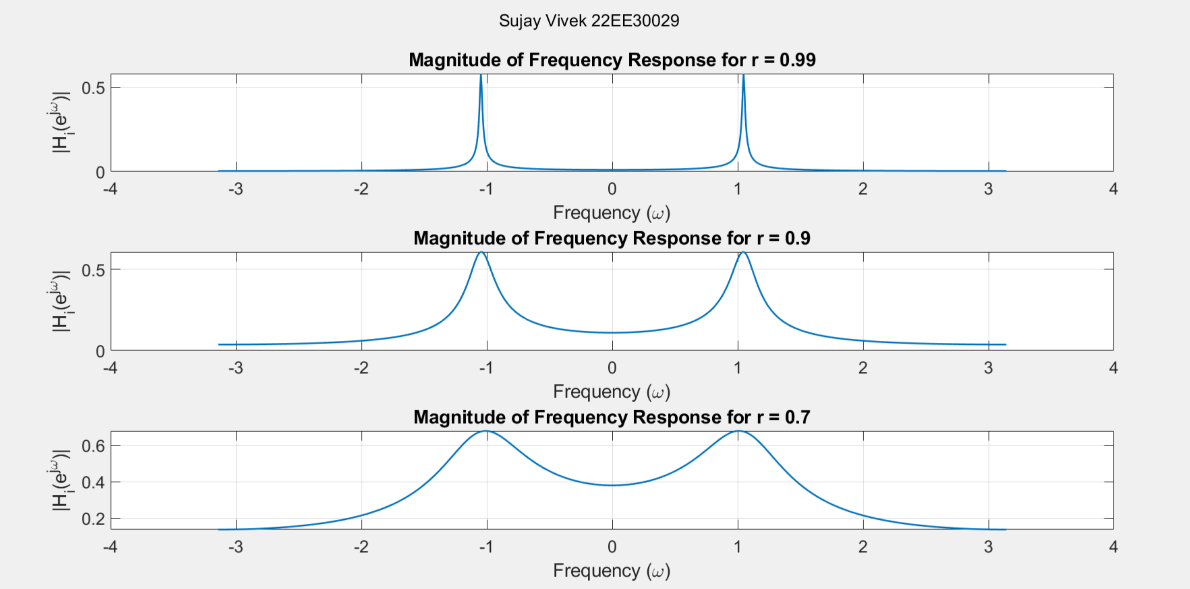
$$\begin{aligned}
H_i(z) &= \frac{1-r}{1-2r\cos(\theta)z^{-1}+r^2z^{-2}} \\
H_i(z) &= \frac{1-r}{1-2r\cos(\theta)e^{-j\omega}+r^2e^{-2j\omega}} \\
h_i[n] &= \frac{1-r}{\sin\theta}r^n\sin((n+1)\theta)u[n]
\end{aligned}$$

The system diagram for this IIR filter involves feedback loops due to the $y[n-1]$ and $y[n-2]$ terms, and a feedforward path for $x[n]$. Here's how the system diagram components are structured:



(a) System Diagram of Transfer Function

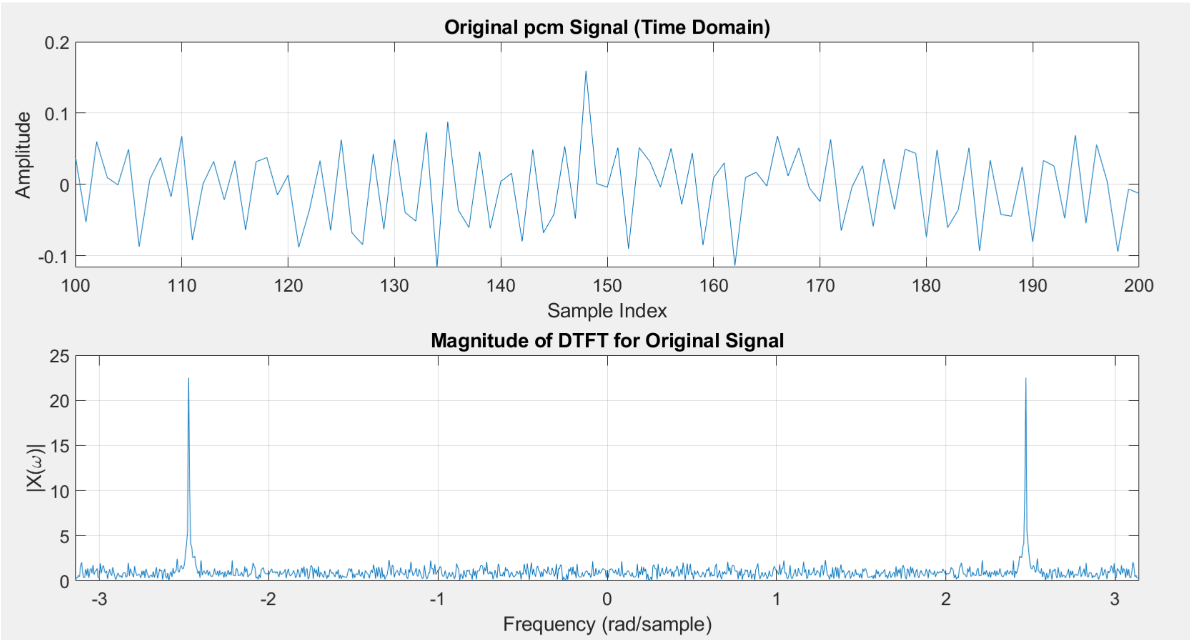
The Magnitude of Frequency Response is plotted with the help of Matlab for different values of R:



(a) Magnitudes of Frequency Responses

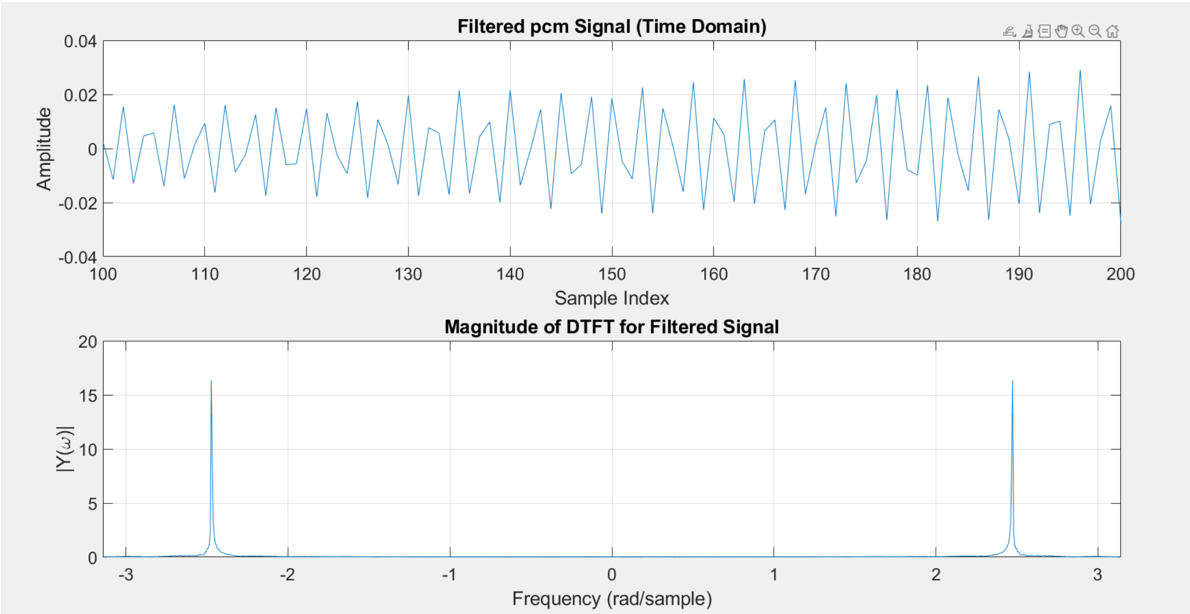
Use of IIR Filter

We'll utilise an IIR filter in this section to distinguish a modulated sinusoid from background noise:



(a) Original PCM Signal in Time and Frequency Domain

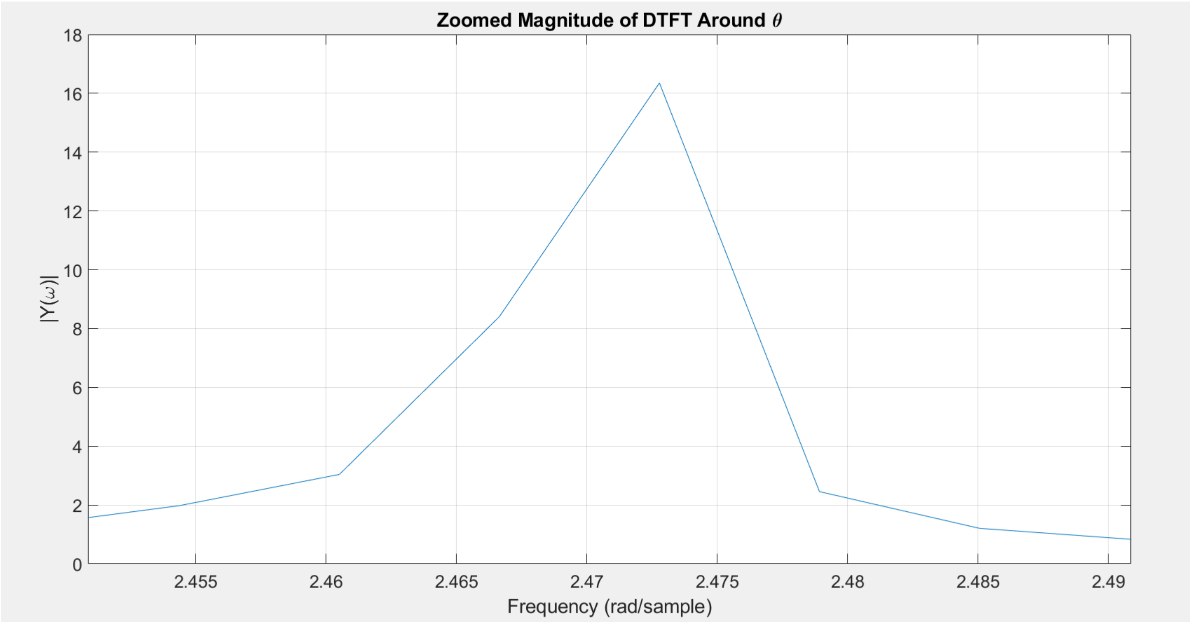
Above is the Time Domain Plot of the Original PCM Signal for 101 points, and the Magnitude of DTFT of Original Signal in the frequency domain for 1001 Points.



(a) Filtered PCM Signal in Time and Frequency Domain

Above is the Time Domain Plot of the Filtered PCM Signal for 101 points, and the Maginitude of DTFT of Original Signal in the frequency domain for 1001 Points.

The plot of the magnitude of the DTFT for w in the range $[\theta - 0.2, \theta + 0.2]$ is given below.



(a) Zoomed Magnitude of DTFT Around θ

After filtering the audio signal, all frequencies other than theta were deleted. As a result, the noise in the signal was removed.

If we move r in the direction of 1, the filter rejects more frequencies and reduces noise. If we raise the value of r above one, the filter will accept more frequencies, resulting in more noise in the filtered audio.

Choosing a value that is really close to 1 (such as r = 0.9999999) is not a good idea because it will waste a lot of memory and computation time while affecting the audio signal only slightly. r = 0.9999, on the other hand, will use 103 times the memory and computing time. More numeric values must be considered as r approaches zero, as h[n]=0.

5. Lowpass Filter Design Parameters

The figures below show the Filtered Signal and the Original Signal. Filtered Signals through both the different sized filters of size 21 and 101 are depicted below.

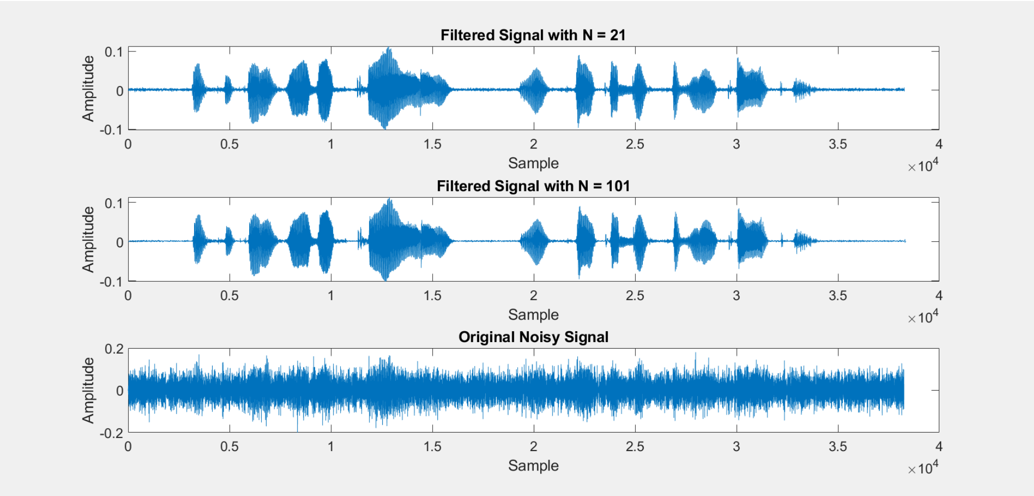


Figure 13: Filtered and Original Signals for different filter sizes)

Figures below show the magnitude responses of the truncated low-pass filters with sizes $N = 21$ and $N = 101$, respectively. These plots are in linear scale, and we can observe that the filter with $N = 101$ has a steeper transition band compared to the filter with $N = 21$.

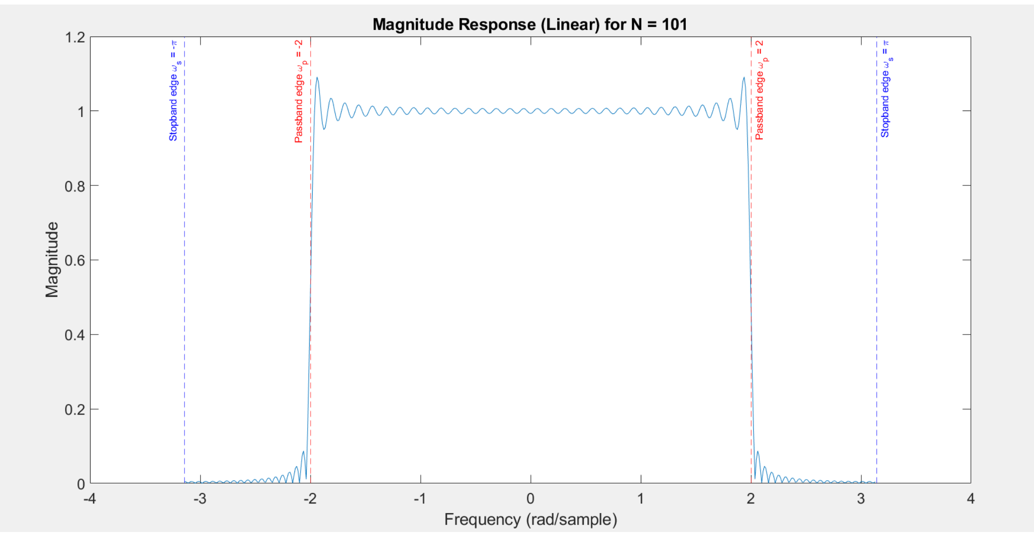


Figure 14: Filtered and Original Signals for different filter sizes)

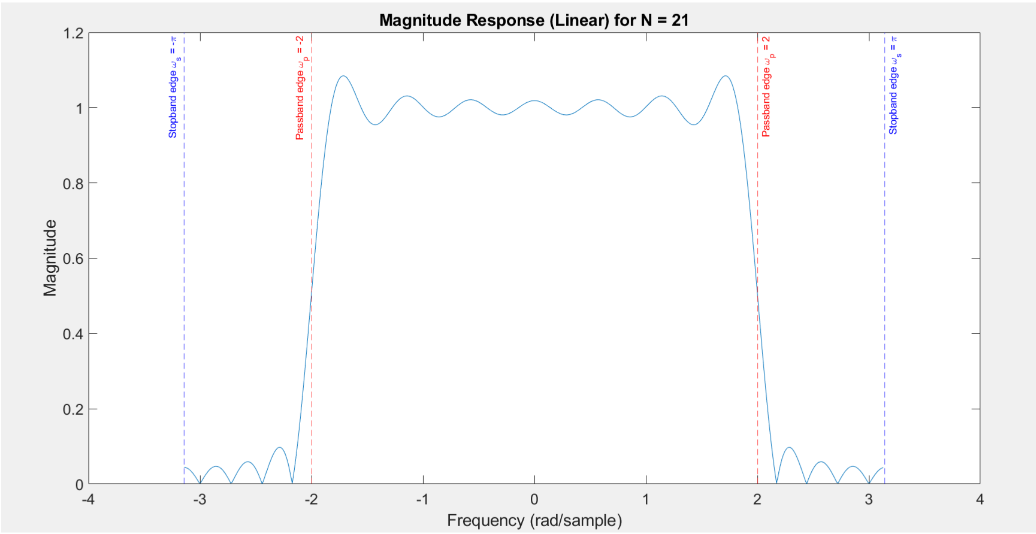


Figure 15: Filtered and Original Signals for different filter sizes)

Similarly, to observe the plots of in Decibels

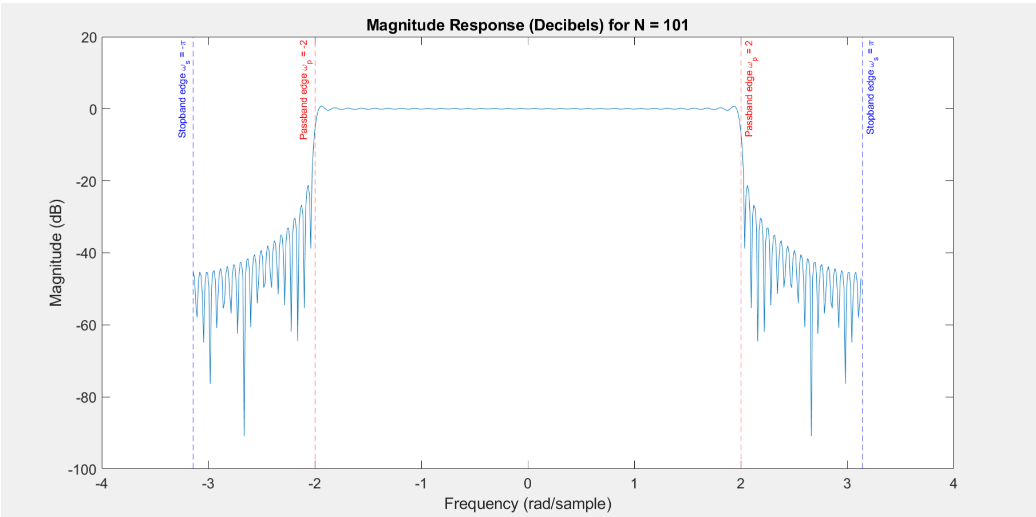


Figure 16: Filtered and Original Signals for different filter sizes)

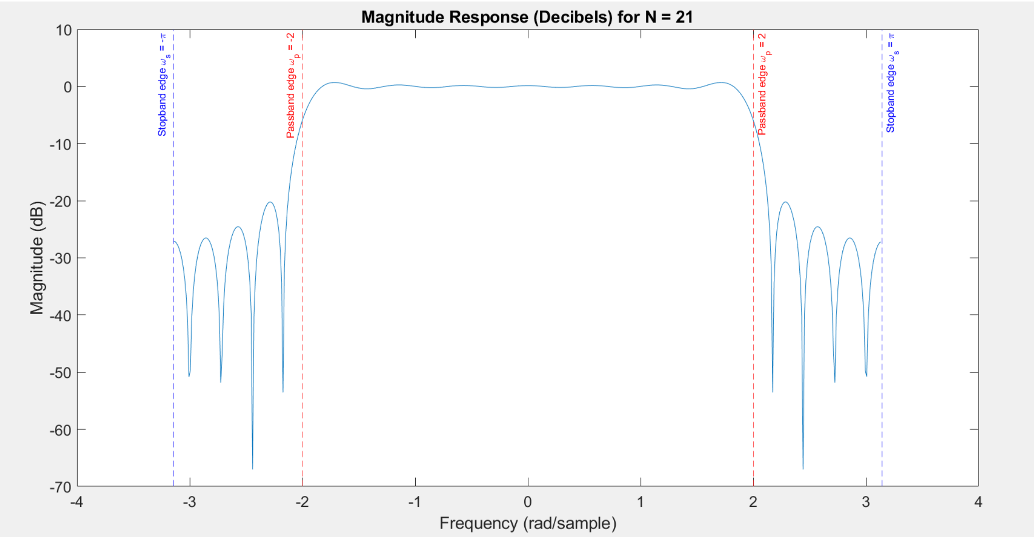


Figure 17: Filtered and Original Signals for different filter sizes)

Some observations

Increasing the filter size (N) has a noticeable effect on the stopband ripple. With a larger N, the stopband ripples decrease, as seen in the decibel plots. This is due to the better approximation of the ideal low-pass filter characteristics, where the transition between passband and stopband is sharper. Therefore, a larger filter size results in a narrower transition band and better suppression of frequencies in the stopband.

After applying the filters to the noisy speech signal `nspeech2.mat`, the audio quality was improved due to the reduction of high-frequency noise. Comparing the two filtered signals, the one filtered with $N = 101$ demonstrated superior noise suppression, resulting in clearer speech. This indicates that the larger filter size provides better quality filtering by attenuating unwanted noise more effectively. However, the larger filter size also introduces slight delays due to the increased number of taps in the filter.

In conclusion, this demonstrates the effect of filter size on the frequency characteristics of a truncated low-pass filter. Increasing the filter size decreases the stopband ripples and improves the transition band sharpness, which enhances the filter's ability to attenuate noise in the stopband. This makes the larger filter more effective at isolating the desired signal from noise, as demonstrated by the audio results.