# Assignment 1

Name: Claire Kim, Michael Zhu (Group G)

## 1. Abstract

This report investigates the design and implementation of six different filters—two Window FIR filters, two Frequency Sampling FIR filters, an Optimal FIR filter, and an IIR filter—to effectively remove noise from the given audio signal (groupG.wav). Through comprehensive time-domain plotting and spectrum analysis, each filter is carefully designed and evaluated to ensure optimal performance. The report also explores the practical aspects of IIR and FIR filter implementations using Python. FIR filters were found to generally outperform IIR filters in terms of stability and audio quality. Among the FIR filters, the Frequency Sampling FIR filter 2 (Homemade) emerged as the most effective, balancing noise reduction and signal preservation. Although the IIR filter showed the lowest filtered signal variance, auditory results indicated it was the least effective, likely due to phase distortions and retained noise. These findings emphasize the importance of combining quantitative and qualitative evaluations in assessing filter performance.

## 2. Introduction

This report focuses on detecting and removing noise from an audio signal. The process involves analysing the signal in time and frequency domains, identifying noise with FFT, and designing IIR and FIR filters in Python to remove the interference. The filters' performance is compared to determine the most effective noise removal method.

## 3. Comprehensive Analysis and Filtering of Audio Signals

### 3.1.

The sampled time-domain audio signal is plotted with the time axis in seconds to clearly show how the signal behaves over time. To enhance clarity and highlight finer details, the plot was zoomed in to focus on the portion of the signal between 0 and 0.08 seconds. This closer view makes it easier to observe specific patterns and variations that might not be as noticeable on a broader time scale. Figure 1 below illustrates the sampled time-domain audio signal, with a sine wave amplitude of one. The audio file has a total duration of nine seconds. The signal was sampled from a WAV file, and its amplitude values were normalized to a unitless scale. This time-domain representation helps us understand how the signal changes over time.

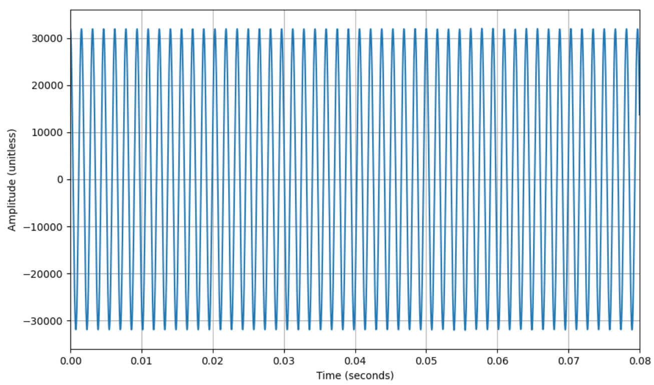


**Figure 1.** Sampled Time domain Plot (Zoomed in from 0 to 0.08 seconds)

## 3.2. Spectrum Analysis and Interference Detection

Figure 2 below shows the magnitude of the spectrum. To analyse the frequency content of the signal, the Fast Fourier Transform (FFT) was computed to convert the signal from the time domain to the frequency domain. The interference signal was found to be around 642.1 Hz based on the peak in the magnitude spectrum.

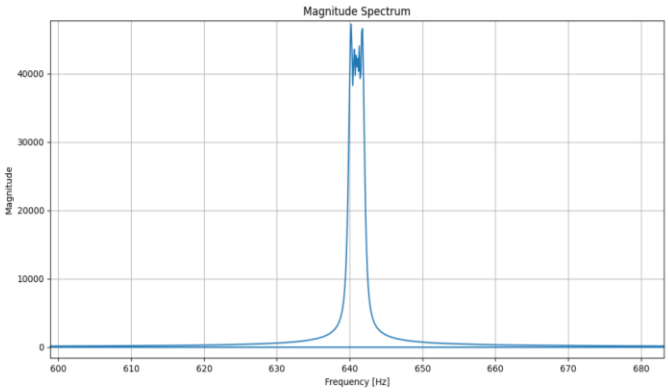


**Figure 21.** Sampled magnitude Plot

## 3.3. Analytical Design of IIR Filter for Noise Reduction

A biquad IIR 2-pole notch filter has second order realization. The design below:

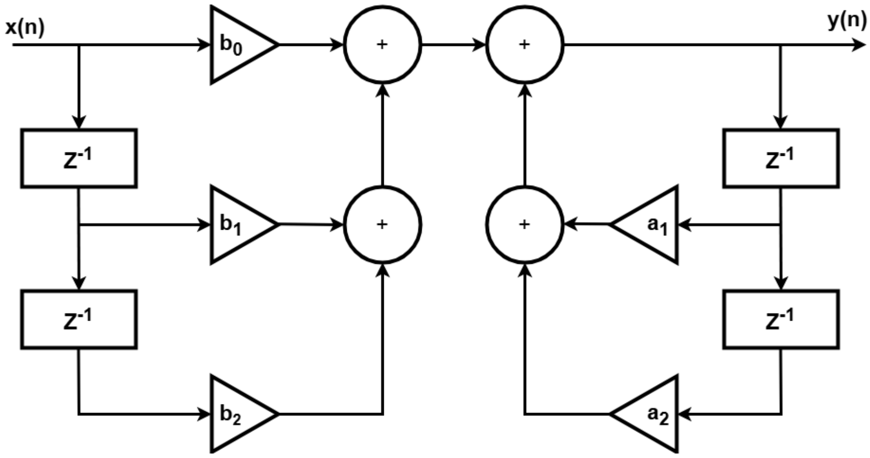


**Figure 3.** Second order realization block diagram

The transfer function of this digital filter is:

$$H(z) = \frac{Zeros}{Poles} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 z^{-2} + a_1 z^{-1} + a_2}$$

a0 is normalized to 1, The difference equation is:

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] - a_1 y[n-1] - a_2 y[n-2]$$

The coefficients $a$ and $b$ in the transfer function determine the placement of poles and zeros on the z-plane which shapes the notch filter. The poles of the filter are determined by coefficients $a$. Poles in the z-domain are placed in the pass band to create peaks in the frequency response. This isolates the interference frequency without affecting much of the surrounding frequencies.

The bandwidth of the notch filter determines the radius of the poles thus determining the placement of the poles. The equation of the relationship for the radius:

$$r = 1 - \frac{Bw}{f_s}\pi$$

Using the radius formula for pole placement a bandwidth that most effectively filters out the target signal can be chosen.

The zeros of the filter are determined by the numerator coefficients $b$. Zeros are placed at the stop band because they attenuate the target signal. Placing zeros on the unit circle at the angles corresponding to the target frequencies will create notches at those frequencies. The angle corresponding to the target frequency $f_a$ for sampling frequency $f_s$ is:

$$\theta_a = \frac{2\pi f_a}{f_s}$$

This is the normalised cut-off frequency of the filter. This frequency is then used to calculate the coefficients of the filter to determine the poles and zeros.

Using the Transfer function in frequency form:

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

We can substitute the parameters $r$ and $\theta$ and apply it to the original sampled signal to filter it. A Gain factor G can also be applied to b ensure unity in the passband.

## Pole Placement

The IIR filter must use complex conjugate pairs to keep coefficients real, so zeros are placed at angles corresponding to the target frequency and its negative counterpart. For the IIR design a bandwidth of 1400Hz was chosen which placed the poles at 0.897±0.082j and zeros at 0.996±0.091j. The pole-zero plot is shown below.

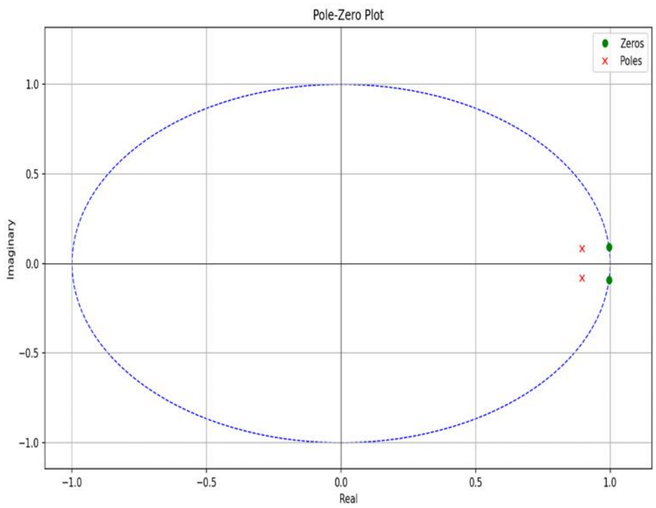


*Figure 2. Pole-Zero Plot*

The poles are placed close to the unit circle, so the notch filter has sharp peaks in the frequency response. The radius of poles formula has r > 0.9 to target the specific frequency. This ensures that the filter has a narrow bandwidth filtering the interference frequency with minimal impact on surrounding frequencies. The zeros are positioned on the unit circle to create a sharp notch at the target frequency. This will attenuate the interference signal. The radius of poles for this filter design is 0.9.

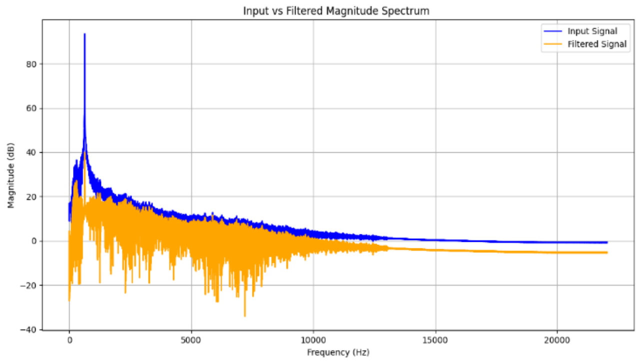


**Figure 5.** Magnitude Spectrum in dB

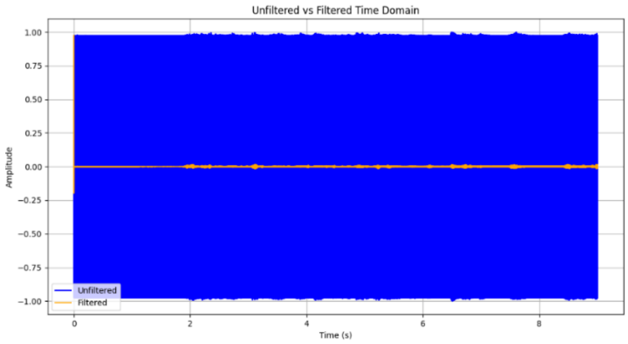


**Figure 6.** Time domain comparison plot

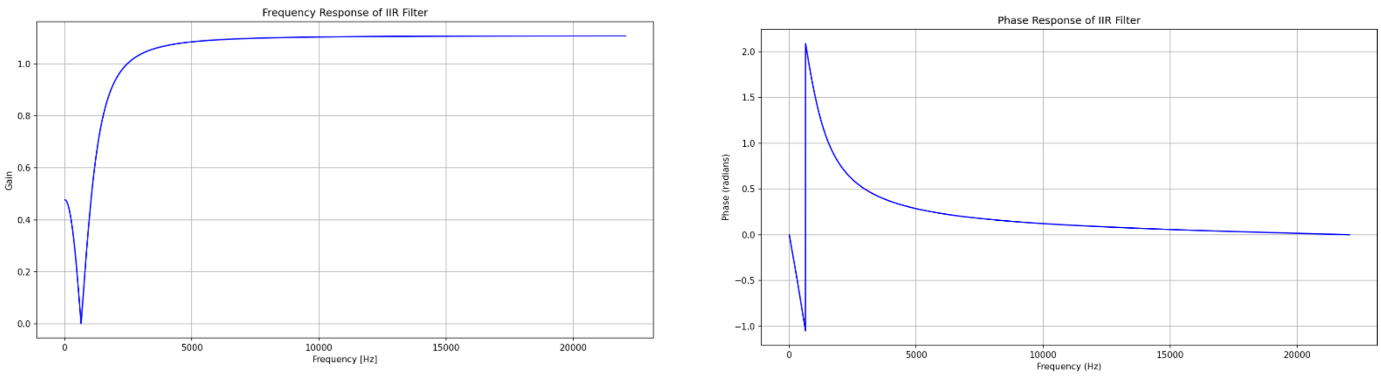


**Figure 7.** IIR Filter Magnitude and Phase response plots

# 3.5. Analytical Design of FIR Filters for Noise Reduction

## 3.5.1. Window-base FIR Notch filter design

In this section, we will show our window-base FIR notch filter to attenuate the interference centred approximately 640 Hz within the given audio signal (groupG.wav). The design process includes defining the specifications of the desired frequency, obtaining the ideal response, and applying a homemade window function while ensuring the filter achieves the requirements, including having an odd number of coefficients for a linear phase response.

### Specifications

In our group's audio signal sampled at a frequency of 44,100 Hz, the interference is centered around 640 Hz. To remove this interference successfully, a notch filter is designed with a narrow stopband around the interference frequency. We set the lower cutoff frequency of f1 and upper cutoff frequency of f2 to be 635 Hz and 645 Hz respectively. Firstly, these frequencies are normalised with respect to the sampling frequency.

$$f_{1,normalised} = \frac{635}{44100} \approx 0.0144$$

$$f_{2,normalised} = \frac{645}{44100} \approx 0.0146$$

### Ideal Impulse Response

The ideal impulse response for a notch filter can be determined by subtracting two ideal low-pass filters. The equation for this is shown below.

$$h_d(n) = \delta(n) - \frac{\sin(\omega_1 n)}{\pi n} + \frac{\sin(\omega_2 n)}{\pi n}, n \neq 0$$

$$h_d(0) = 1 - (f_2 - f_1)$$

Where $\omega_1$ and $\omega_2$ represent $2\pi f_1$ and $2\pi f_2$ respectively. These values correspond to the normalised cut-off frequencies for the stopband edges at 635 Hz and 645 Hz. $h_d(n)$ is the ideal impulse response for the notch filter.

This ideal impulse response generates a notch at the desired frequencies when convolved with the input signal, effectively removing the interference.

### Application of Hamming Window

To obtain a practical FIR filter, the ideal impulse response should be truncated to a finite length. This is achieved by applying a Hamming window, which smooths the transitions and reduces side lobes, thereby enhancing the frequency response of the filter. The Hamming window is defined as the following equation [1].

$$\omega(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right)$$

Where N is the number of coefficients used in the filter design.

Then the windowed impulse response is figured out as the following equation.

$$h(n) = h_d(n) \times \omega(n)$$

This operation makes sure that the resulting filter has a smooth and effective frequency response with minimal ripples.

### Filter Length and Transition Width

The number of coefficients, *N*, is an important factor in FIR filter design that influences the filter's performance. In this design, N is set to 2047 to create a sharp transition between the passband and stopband, effectively targeting the narrow interference band around 640 Hz. An odd number of coefficients ensures a linear phase response, which helps avoid phase distortion and preserves the signal's integrity.

While increasing N generally enhances stopband attenuation and transition sharpness, it also has trade-offs. A larger N increases the computational complexity and can introduce more latency, which may not be desirable in real-time applications [2]. Additionally, using a very high N can sometimes cause excessive passband ripple or make the filter too sensitive to minor signal variations.

For this design, setting N equals to 2047 strikes a balance between effectively suppressing the interference and keeping the filter manageable in terms of complexity. While the design is not perfect, it effectively removes the main noise from the signal, allowing the important content to still be recognised. This balance helps the filter work as needed without making it too complicated or causing extra issues.

#### Frequency Response of the Designed Filter

The frequency response of the FIR filter can be obtained by taking the Discrete Fourier Transform of the windowed impulse response. This frequency response should show a sharp attenuation (notch) around 640 Hz, effectively removing the interference from the audio signal.

$$H(\omega) = \sum_{n=0}^{N-1} h(n) e^{-j\omega n}$$

## 3.5.2 Frequency Sampling FIR Filter Design

The frequency sampling method is a powerful technique for FIR filter design, particularly when the desired frequency response can be specified at discrete frequency points. In this method, we define the desired frequency response in the frequency domain and then calculate the corresponding time-domain filter coefficients using the Inverse Discrete Fourier Transform (IDFT).

### Ensuring Linear Phase

Linear phase in FIR filters is important for preserving the shape of signals passing via the filter. Achieving linear phase needs the filter coefficients to show specific symmetry properties.

- Type I Filter: Symmetric coefficients with an odd number of taps
- Type II Filter: Symmetric coefficients with an even number of taps
- Type III and IV Filters: Non-symmetric coefficients which are less common for standard filtering tasks.

In our design, we will focus on a Type I filter to ensure symmetric coefficients and an odd number of taps to keep linear phase.

### Defining the Desired Frequency Response

The frequency points are sampled at intervals of $\frac{k \cdot f_s}{N}$ where $k$ is the index of the frequency bin and $f_s$ is the sampling frequency of 44,100 Hz. To find the value of $k$, we used the following equation.

$$k = round\left(\frac{f_0 \cdot N}{f_s}\right)$$

### Constructing the Desired Frequency Response

The desired frequency response $H_d(k)$ is set to make a notch at the interference frequency.

$$H_d(k) = \begin{cases} 0 & for\ k = 29, 30, 31 \\ 1 & otherwise \end{cases}$$

Our $H_d(k)$ creates a narrow notch centred around $k_0 = 30$.

### Adding Transition Samples

In order to optimise the frequency response, transition samples can be placed around the notch. These transition samples smooth the transition between passband and stopband. This results in reducing ringing effects in the frequency response, however, slightly widening the transition band.

The number of transition samples decides the trade-off between the width of the transition band and the sharpness of the notch. One transition sample is added at k= 28 and k= 32, with the value of 0.5.

### Calculating Time-Domain Filter Coefficients

The time-domain filter coefficients of h(n) are calculated using the Inverse Discrete Fourier Transform of $H_d(k)$.

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H_d(k)e^{j\pi kn/N}$$

Where n = 0, 1, 2, …, N-1.

This computation results in a symmetric impulse response to ensure the filter maintains a linear phase.

## 3.6. Implementation of FIR Filters Using Python

In this task, we will apply the filters we designed in the section 2.5 to the given audio signal and then evaluate their performance. By comparing the filtered outputs, our goal is to determine which filter design removes the noise effectively while preserving the true signal.

### 3.6.1. Hamming Window FIR Filter using firwin2

The Hamming FIR filter designed using the *firwin* function was constructed through a more automated process than the homemade Hamming window approach. The filter design began by specifying the lower and upper cutoff frequencies, which were normalised relative to the Nyquist frequency. The firwin function, which generates FIR filter coefficients, was used with a bandstop configuration to target the desired frequency range. The function automatically applied the Hamming window to the filter coefficients, handling the details of windowing and coefficient generation internally. This method simplifies the filter design process, making it easy to create a bandstop filter with minimal manual calculations, while still benefiting from the characteristics of the Hamming window. While this makes the process faster and simpler than custom window method, it offers less manual control over the specific details of the filter design.

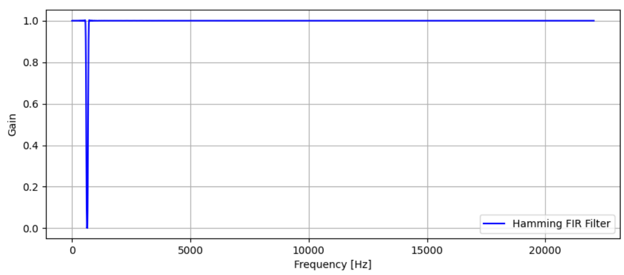


**Figure 8.a.** Frequency Response of the FIR Filter

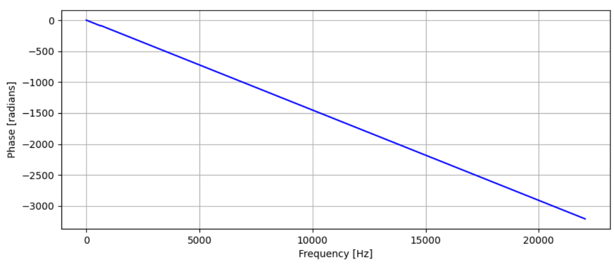


**Figure 8.a.** Phase Response of the FIR Filter

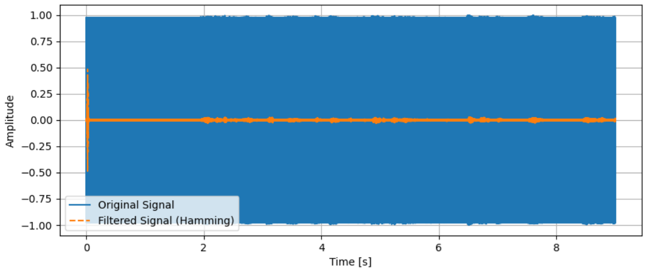


**Figure 8.c.** Time-Domain Signal (Original vs. Filtered)

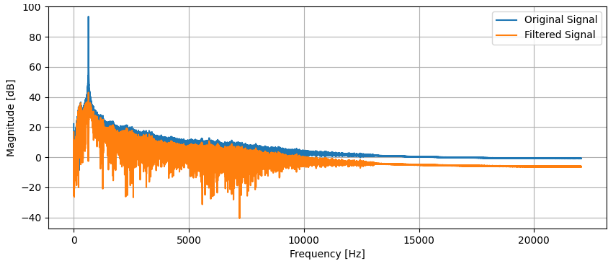


**Figure 8.d.** Magnitude Spectrum in dB

## 3.6.2. Frequency Sampling FIR Filter using firwin2

This frequency sampling FIR filter is designed using the firwin2 function, which simplifies the creation of custom filters by allowing users to specify key frequency points and corresponding gain values. In this case, the filter is configured as a bandstop (notch) filter targeting the frequency range between 600 Hz and 680 Hz. The *freqs* array defines the passbands with gains of 1 and the stopband with a gain of 0 around the cut-off frequencies. The *firwin2* function then interpolates between these points to generate the filter coefficients. Unlike the manual approach, which requires direct manipulation of the frequency domain and inverse Fourier transform calculations, firwin2 automates the process, making it more accessible and faster to implement. However, while the firwin2 method offers convenience and ease of use, the manual method provides more precise control over the filter's frequency response, allowing for finer customisation.

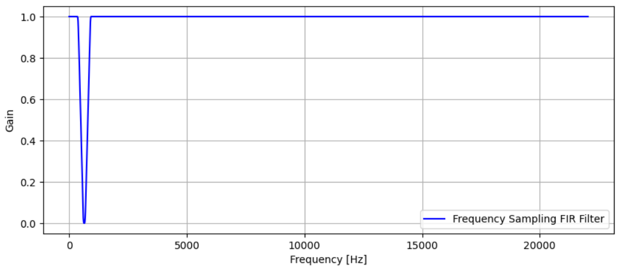


**Figure 9.a.** Frequency Response of the FIR Filter

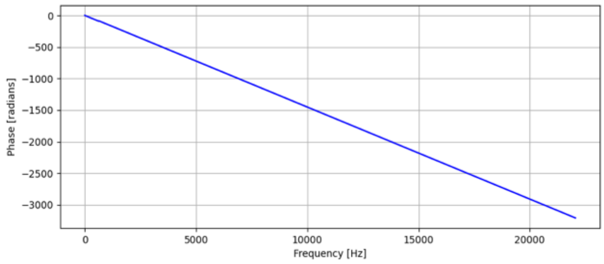


**Figure 9. b.** Phase Response of FIR Filter

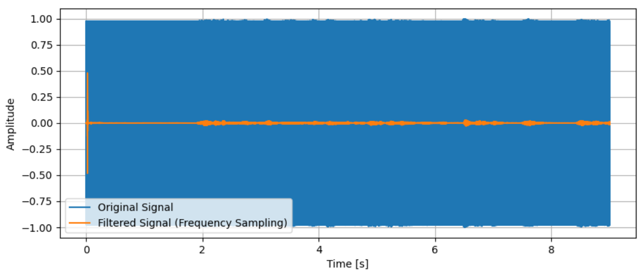


**Figure 9.c.** Time-Domain Signal (Original vs. Filtered)

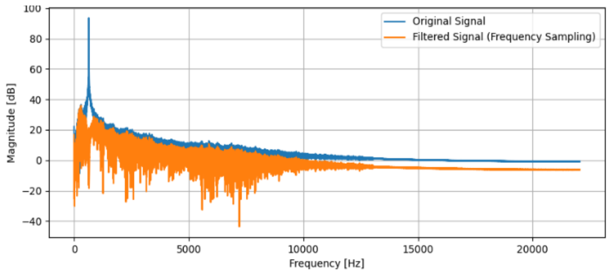


**Figure 9.d.** Magnitude Spectrum in dB

## 3.6.3. Optimal FIR Filter

The optimal FIR filter was designed using the Parks-McClellan algorithm, implemented via the *remez* function in Python. The filter is designed to attenuate the frequency band between 635 Hz and 645 Hz, while maintaining the passband gain outside this range. The filter parameters were initialised by defining the transition width and the band specifications relative to the Nyquist frequency. The number of coefficient was set to 999. The bands array specifies the edges of the passbands and stopbands, while the desired array defines the gain within each band, with '1' for the passband and '0' for the stopband. The weighting array assigns different importance to the bands, prioritising stopband attenuation.

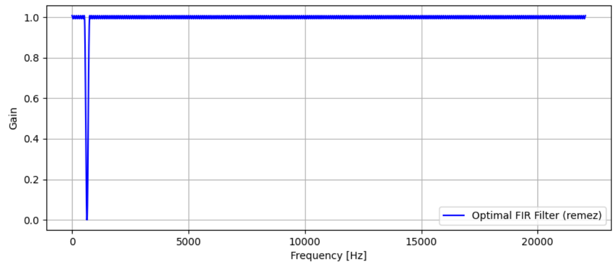


**Figure 10.a.** Frequency Response of the FIR Filter

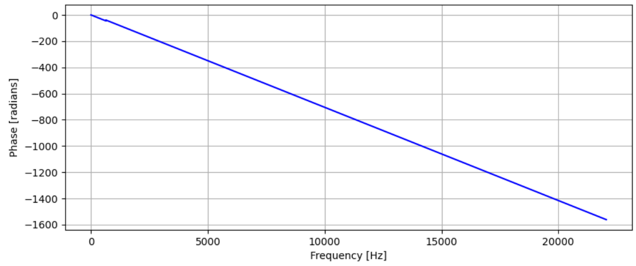


**Figure 10.b.** Phase Response of the FIR Filter

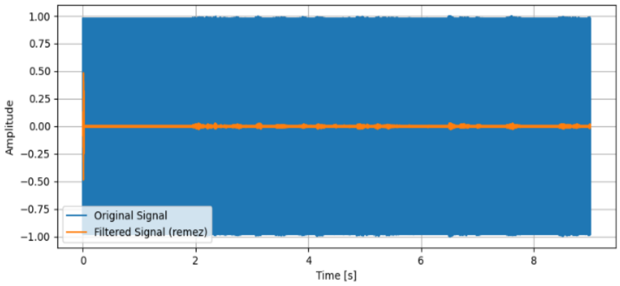


**Figure 10.c.** Time-Domain Signal (Original vs. Filtered)

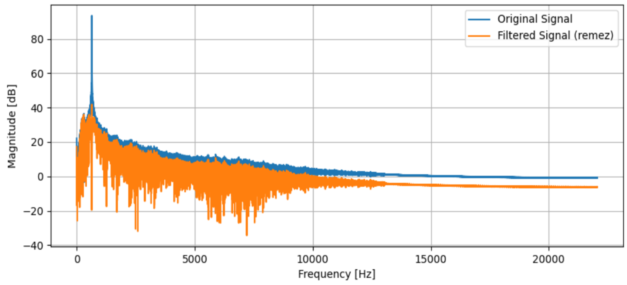


**Figure 10.d.** Magnitude Spectrum in dB

## 3.6.4. Homemade Frequency Sampling FIR Filter

The homemade Frequency Sampling FIR filter was designed by first loading and normalising the audio signal. The filter's parameters were initialised with a chosen number of coefficients of 345 and a specific frequency range where the interference was detected. This frequency range was targeted for attenuation, turning the filter into a notch filter. The frequency response was initially set as an all-pass filter, allowing all frequencies to pass without alteration. To create the notch filter, the frequency components within the identified interference range (635 Hz to 645 Hz) were set to zero, effectively eliminating those frequencies from the signal. Ensuring symmetry in the frequency response was important to maintain a real-valued impulse response, which is a fundamental characteristic of FIR filters. After defining the frequency response, the inverse Fourier transform was used to calculate the impulse response. The impulse response was normalised to ensure that the sum of its coefficients equalled one, preserving the signal's overall amplitude after filtering. This normalisation step is important to avoid altering the signal's volume unintentionally.

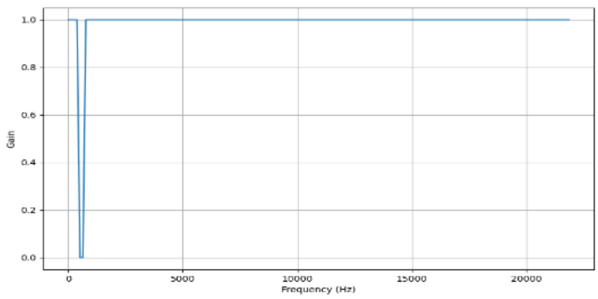


**Figure 11.a.** Frequency Response of the FIR Filter

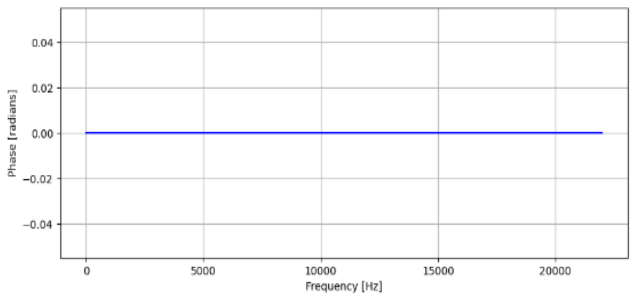


**Figure 11.b.** Phase Response of the FIR Filter

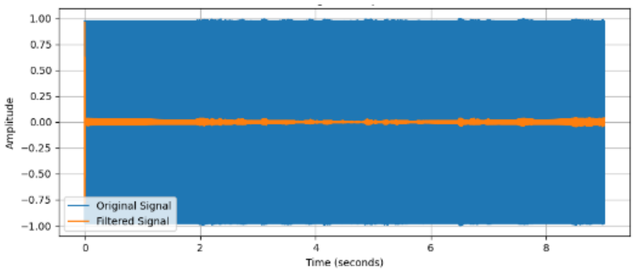


**Figure 11.c.** Time-Domain Signal (Original vs. Filtered)

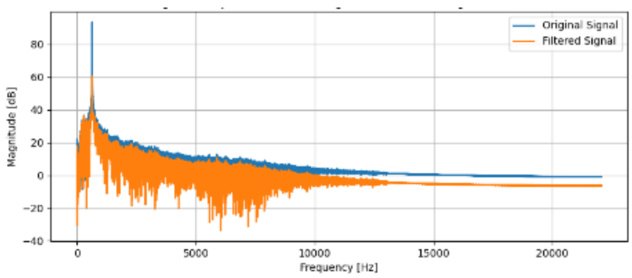


**Figure 11.d.** Magnitude Spectrum in dB

## 3.6.5. Homemade Hamming Window FIR Filter

The homemade Hamming FIR filter was constructed by first calculating the ideal impulse response for a notch filter targeting a specific frequency range. This process included manually defining the stopband by setting the desired frequency components to zero in the frequency domain, ensuring that the filter would effectively attenuate those frequencies. Once the ideal impulse response was established, a Hamming window was applied to smooth the transitions and reduce side lobes in the frequency response. The windowing process shapes the filter by tapering the impulse response, which helps control the frequency response more precisely. Finally, the filter coefficients were normalised to ensure that the filter preserved the overall signal amplitude after filtering. The homemade Hamming FIR filter provides more granular control over the filter design process compared to the 'firwin2' approach, requiring manual calculations and deeper understanding of FIR filter principles, including the direct manipulation of the impulse response and normalisation. This allows for more flexibility in shaping the filter's characteristics.

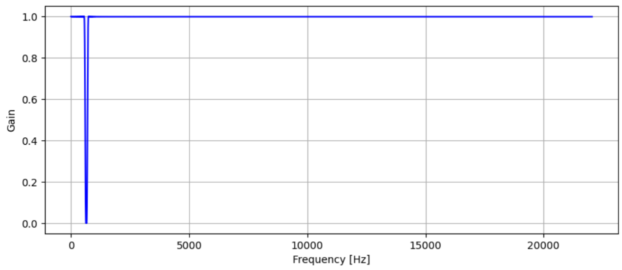


**Figure 12.a.** Frequency Response of the FIR Filter

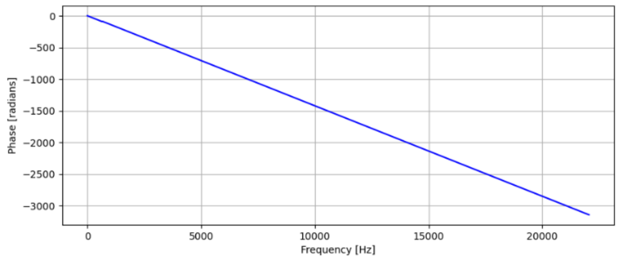


**Figure 12.b.** Phase Response of the FIR Filter

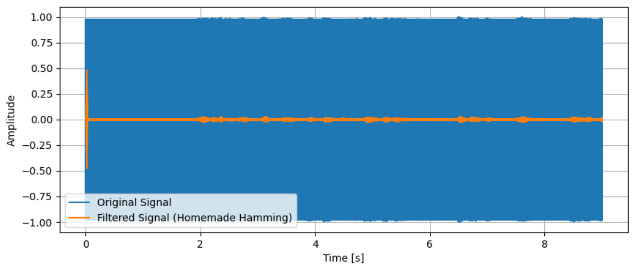


**Figure 12.c.** Time-Domain Signal (Original vs. Filtered)

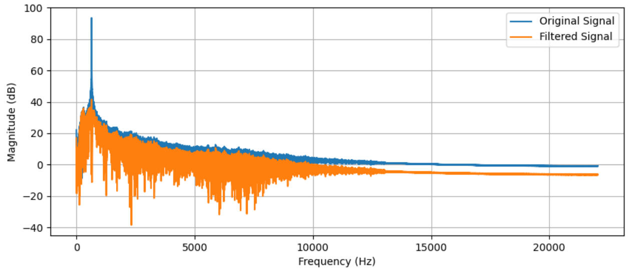


**Figure 12.d.** Magnitude Spectrum in dB

## 3.7. Variance Analysis of Interference and Signal Integrity Across Six Filters

The variance calculations for the filtered signals help us understand how much noise has been removed by each filter. When the signal and noise are uncorrelated, we can use the following equation.

$$Var(Filtered\ Signal + noise) = Var(Filtered\ Signal) + Var(noise)$$

In a perfect scenario, the variance of the filtered signal Var(filtered signal) should reflect the variance of the original signal after the noise has been removed, while the variance of the noise Var(noise) shows how much of the interference the filter was able to remove. The calculated the variance of the interference and the true (uncorrupted) signal are shown in Table 1 in the next page.

**Table 1.**

| | Given Signal Variance | Filtered Signal Variance | Noise Variance |
|---|---|---|---|
| **Frequency Sampling FIR filter 1** | 0.47503 | 2.72614e-05 | 0.47500 |
| **Frequency Sampling FIR filter 2 (Homemade)** | 0.47503 | 0.0002695 | 0.47476 |
| **Window FIR filter 1 (Hamming)** | 0.47503 | 7.84158e-05 | 0.47495 |
| **Window FIR filter 2 (Homemade)** | 0.47503 | 6.68941e-05 | 0.47496 |
| **Optimal FIR filter** | 0.47503 | 6.1336e-05 | 0.47497 |
| **IIR** | 0.47503 | 5.83716e-06 | 0.47502 |

By comparing the variances across different filters, we can assess which filter offers the best balance between retaining the true signal and minimising noise. However, since we do not have the exact variance values for the true signal and noise independently, the variance results from our calculations only provide estimates of filter performance. Without knowing the precise variance of the original signal or the exact noise level, it is challenging to definitively determine which filter is the most effective based solely on our variance calculations. While the results provides useful insights, they do not fully capture the effectiveness of each filter in terms of preserving the true signal versus eliminating noise.

While the IIR filter shows the lowest filtered signal variance (5.83716e-06), this doesn't necessarily translate into better audio quality. The low variance might indicate that the IIR filter removed more of the signal components along with the noise, resulting in a less effective audio output.

The higher noise variance for the IIR filter (0.47502) compared to the FIR filters indicates that more noise may have remained in the signal, which could explain why the IIR filter sounded least effective during your audio test. In contrast, FIR filters maintained a better balance between reducing noise and preserving the signal integrity.

## 3.8. Analysis and Comparison of Filter Performance on Audio Quote Identification

| **IIR** | **Attenuation(db)** | Stopband width(hz) | Transition Band width(hz) | Linearity |
|---|---|---|---|---|
| **IIR** | **50dB** | 1 | 3000 | Non linear |
| **Frequency Sampling FIR filter 1** | 68 | 52 | 258 | linear |
| **Frequency Sampling FIR filter 2 (Homemade)** | 30 | 130 | 129 | linear |
| **Window FIR filter 1** | 50 | 19 | 71 | Linear |

| | | | | |
|---|---|---|---|---|
| **(Hamming)** | | | | |
| **Window FIR filter 2 (Homemade)** | 32 | 126 | 124 | Linear |
| **Optimal FIR filter** | 52 | 10 | 100 | Linear |

### Magnitude spectrum comparison and Magnitude and phase frequency response

The magnitude spectrum of the IIR notch filter, as shown in Figure 5, demonstrates a substantial attenuation of approximately 50 dB at the target interference frequency of 640.2 Hz. Despite this significant reduction, a small residual spike remains, indicating that not all interference was entirely removed, resulting in some residual noise. The IIR Frequency Sampling FIR Filter 1 achieves a higher attenuation of 68 dB with a narrower stopband width of 52 Hz and a transition band width of 258 Hz, while the Frequency Sampling FIR Filter 2 (Homemade) provides 30 dB attenuation with a stopband width of 130 Hz and a transition band width of 129 Hz. The Window FIR Filter 1 (Hamming) offers 50 dB attenuation with a stopband width of 19 Hz and a transition band width of 71 Hz, and the Window FIR Filter 2 (Homemade) delivers 32 dB attenuation with a stopband width of 126 Hz and a transition band width of 124 Hz. The Optimal FIR Filter provides 52 dB attenuation with a stopband width of 10 Hz and a transition band width of 100 Hz. These FIR filters generally offer higher attenuation and narrower stopband widths, which might make them more effective for this application. The frequency response IIR filter achieves about -60 dB attenuation at the target frequency, with minimal ripple across other frequencies and a slight amplification of around 0.8 dB above 5000 Hz. The phase plot shows a significant phase shift around the interference frequency, which returns to 0 radians as the frequency moves away from this point. This indicates that while the IIR filter effectively targets the interference, it introduces some phase distortion due to its feedback. In contrast, the FIR filters maintain more consistent performance with less impact on the overall audio signal quality.

### Time domain comparison

The time-domain comparison of the original and all filtered signals all about the same except for the homemade hamming filter shown in figure 11.c. The original signal had an amplitude of approximately 1. This indicates strong signal strength. After filtering, the amplitude has been significantly reduced to around 0.02. The homemade hamming had a slightly larger amplitude at 0.06 compared to the rest which shows that there is still quite a bit of the interference signal left. This reduction in amplitude suggests that not only the interference but also the overall signal strength has been reduced.

The original waveform was a large block of large sine waves showing the interference. In the filtered signals, the amplitudes vary more significantly. This change in waveform structure indicates that the filters reduced the dominance of the interference frequency while preserving the characteristics of the movie quote signal.

The audio file under analysis contains the quote "Son, we live in a world that has walls, and those walls have to be guarded by men with guns. Who's goanna do it? You? You, Lt.Weinburgh?," spoken by the character Colonel Nathan R. Jessup, portrayed by Jack Nicholson in the movie “A Few Good Men”.

## 4. Conclusion

FIR filters have demonstrated superior performance over IIR filters in this assignment, both in terms of stability and phase response. While both filter types have their unique advantages, FIR filters proved more effective due to their inherent stability and linear phase response, which are critical for preserving the signal's integrity.

FIR filters are more stable because they do not have feedback loops, which are present in IIR filters and can lead to instability. This lack of feedback makes FIR filters more reliable, especially in applications where stability is essential. Additionally, FIR filters offer a linear phase response, meaning they delay all frequency components of the signal by the same amount. This characteristic is particularly beneficial in preserving the wave shape of the audio signal, helping to maintain sound quality. Among five different FIR filters, based on the variance table provided and the filtered audio, the most effective FIR filter appears to be the Frequency Sampling FIR filter 2 (Homemade).

On the other hand, IIR filters do not have a linear phase response, leading to phase distortion. This phase distortion alters the phase relationship between different frequency components, which can degrade the audio quality.

# 5. Statement Regarding AI Usage

In the completion of this assignment, ChatGPT was used to assist in refining explanations, and verifying concepts. The tool was used to improve my understanding and make sure clarity in the presentation of technical content. All contents generated by ChatGPT were reviewed, edited, and supplemented with additional research to align with the assignment requirements.

## List of Prompts Used

- What are the trade-offs of using a higher number of coefficients in FIR filter design?
- How can I make technical sentences more accessible and easier to understand?
- Check the grammar for the following sentences.
- What are the key advantages and limitations of using the frequency sampling method for FIR filter design?
- What are the practical considerations when choosing the number of coefficients N in FIR filters?
- Suggest useful resources and articles to improve my understanding of signal processing.

# 6. References


[1] Ifeachor, E.C. and Jervis, B.W., *Digital Signal Processing: A Practical Approach*, Prentice Hall, 2nd Edition, pp 459-62.

[2] A. Antoniou, "Digital Filters: Analysis, Design, and Applications," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 1, pp. 42-55, Jan. 1989.