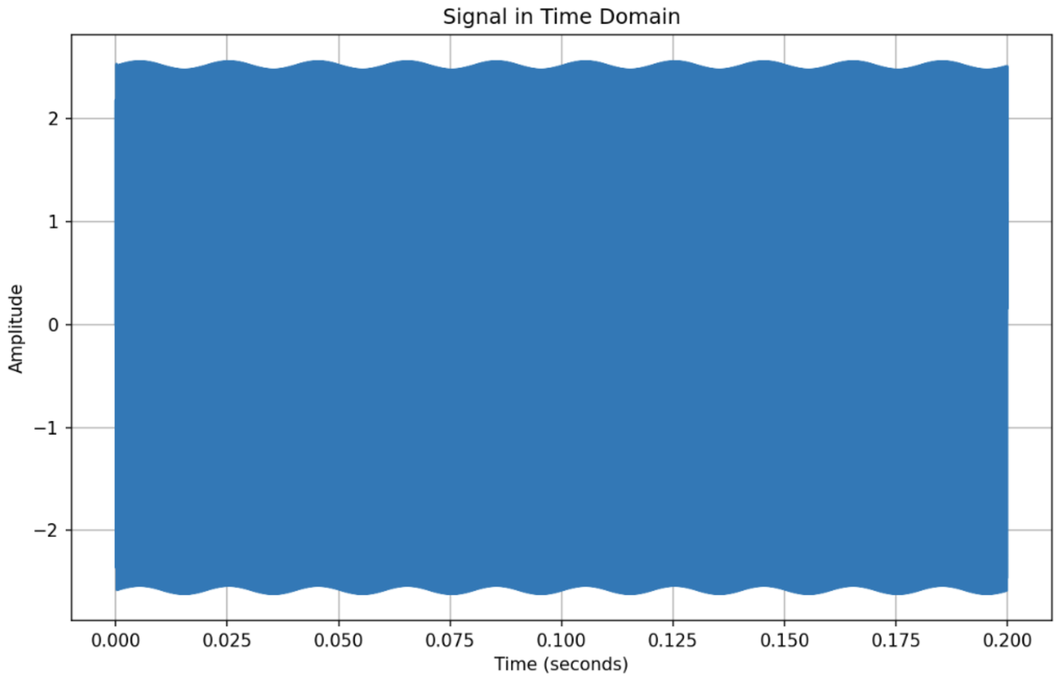
DSP Final Project

The data for this project was created and stored in a text file named - 201282401-proj_data.txt. You have the task to process the data according to the following instruction.

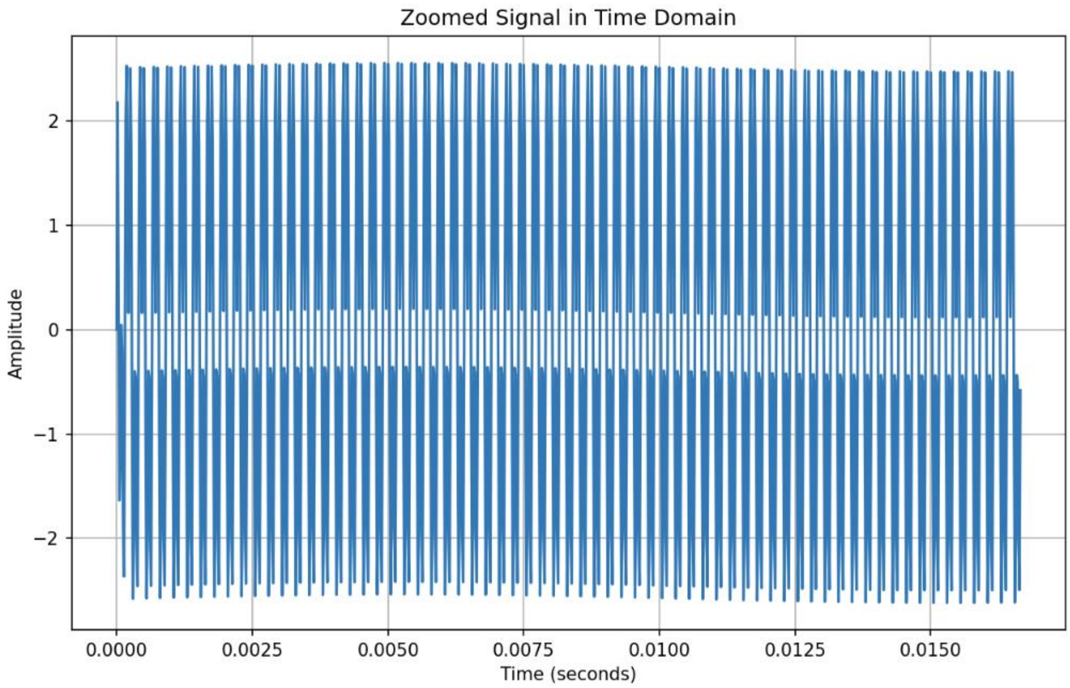
The sampling frequency is 60000 Hz.

Q1: Plot tha data in time. Make sure to label the axes, use the correct units and give a title to the plot.



Q2: Plot a time zoomed versios of the data in time so the shape can be seen clearly

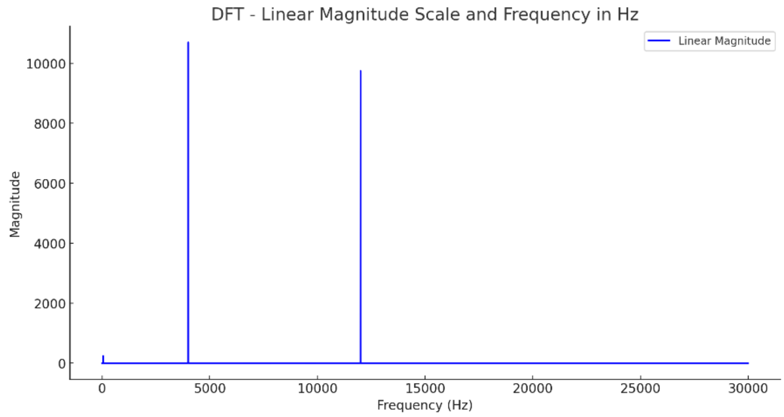
We can see above we loaded a noisy sinusoidal signal.



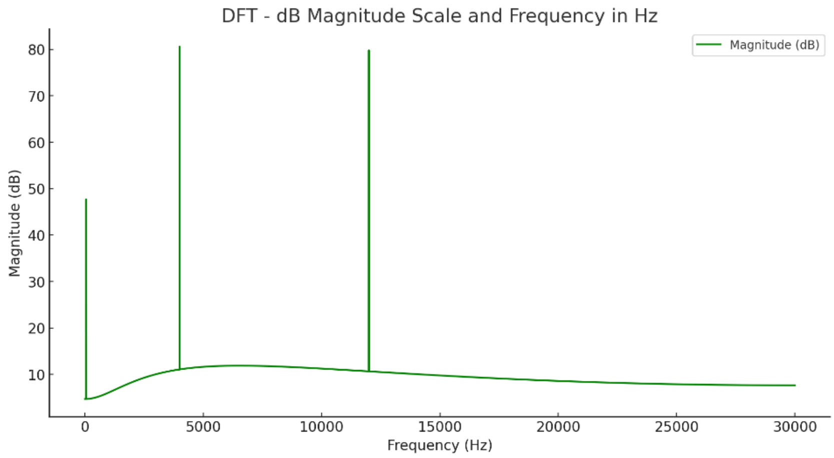
Q3: Plot the DFT of the data in three ways

Because of DFT symmetry, we can show only the positive half of transform which refers to the positive frequencies.

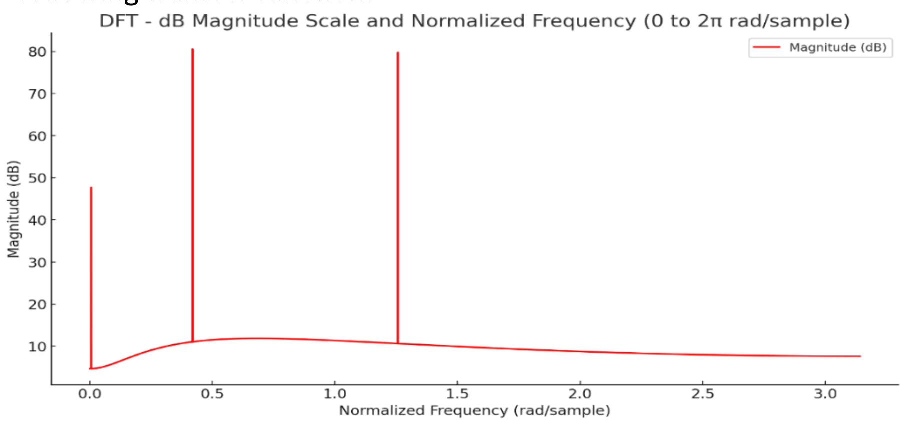
1. **Linear Magnitude scale and frequency axis in Hz:**



2. **Use dB magnitude scale and frequency axis in Hz:**



3. **Use dB** magnitude scale and frequency axis in normalized frequency (0 to 2π rad/smp) It is known that the signal was distorted by a system with the following transfer function:



Q4:Write the frequency response of the distorting system.

To write the frequency response of the distorting system, we start with the transfer function H(z):

$$H(z) = \frac{\left(z - 1.1e^{j0.01}\right)\left(z - 1.1e^{-j0.01}\right)}{(z - 0.5)^2}$$

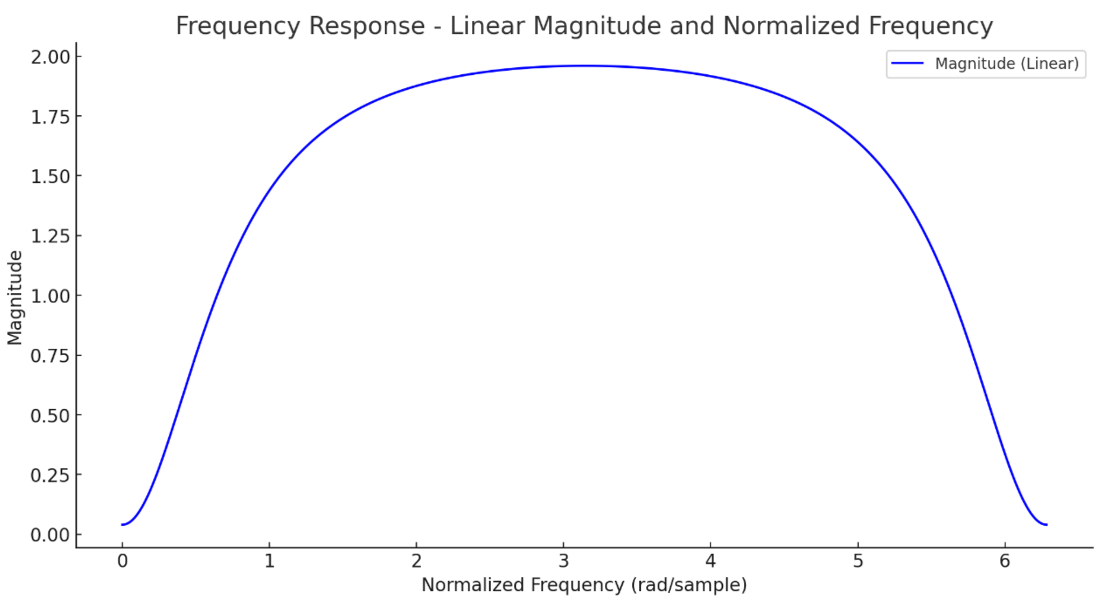
In the discrete-time domain, the frequency response can be obtained by substituting $z = e^{jw}$

$$H(e^{jw}) = \frac{\left(e^{jw} - 1.1e^{j0.01}\right)\left(e^{jw} - 1.1e^{-j0.01}\right)}{(e^{jw} - 0.5)^2}$$

$$H(e^{jw}) = \frac{e^{2jw} - 2.2cos(0.01)e^{jw} + 1.21}{e^{2jw} - e^{jw} + 0.25}$$

$$H(e^{jw}) = \frac{e^{2jw} - 1.7e^{jw} + 0.6}{e^{2jw} + 0.3e^{jw} - 0.1}$$

Q5: Plot the frequency response of the distorting system. Use linear magnitude and normalized frequency (0 to 2π rad/smp)$|H(\omega)|$:



Q6: We separate the transfer function to $H_{min}(z)$and $H_{ap}(z)$:

Zeros:

- z=1.1$e^{j0.01}$ (outside the unit circle, |z|=1.1) =0.9091$e^{-0.01j}$
- z=1.1$e^{-j0.01}$ (outside the unit circle, |z|=1.1) = 0.9091$e^{0.01j}$

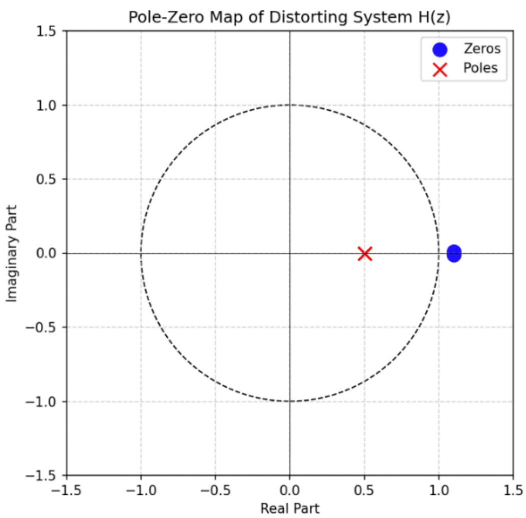
Poles:

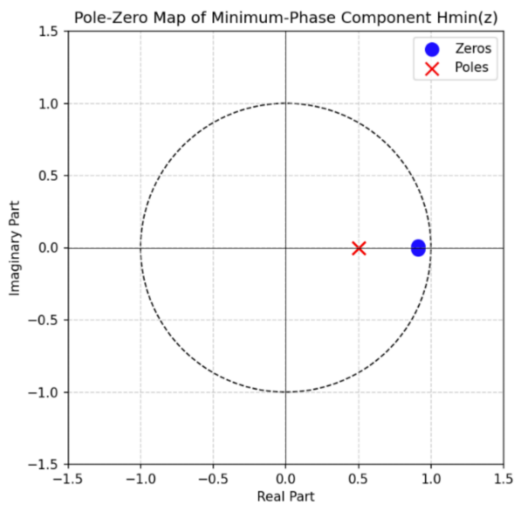
- z=0.5 (inside the unit circle, |z|=0.5, with multiplicity 2)

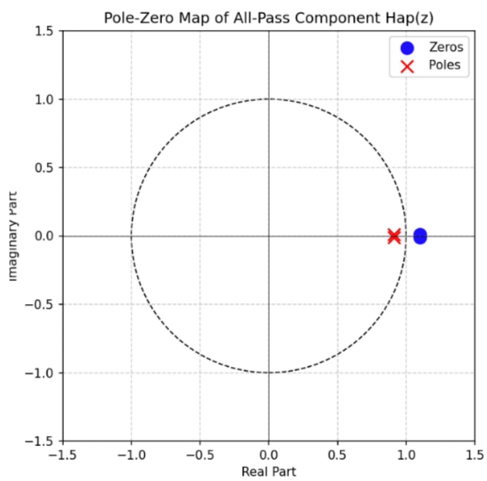
$$\boldsymbol{H_{min}(z)} = \frac{(z-0.9091e^{-0.01j})(z-0.9091e^{0.01j})}{(z-0.5)^2}$$

$$\boldsymbol{H_{ap}(z)} = \frac{(z-1.1e^{-0.01j})(z-1.1e^{0.01j})}{(z-0.9091e^{-0.01j})(z-0.9091e^{0.01j})}$$

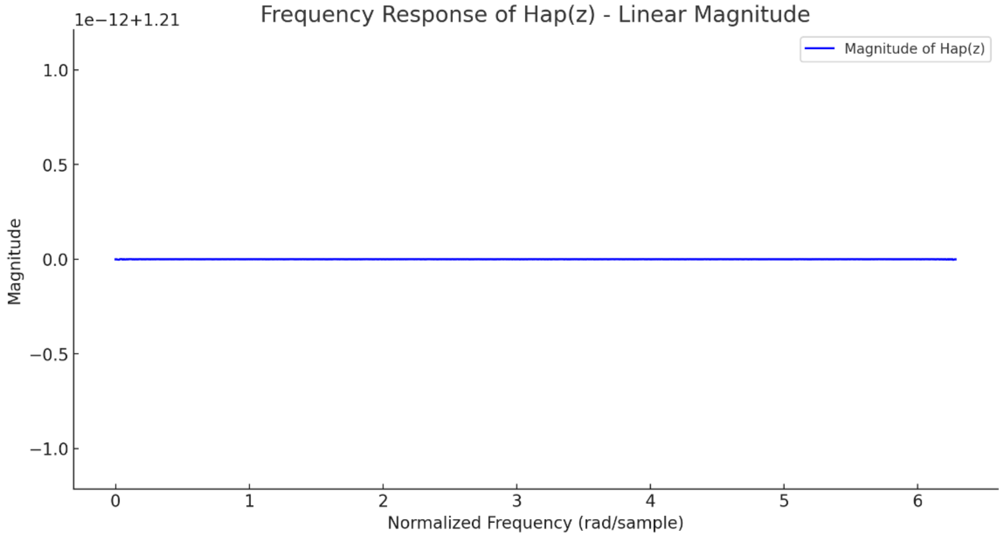
Q7: We draw zero-pole maps for $H(z), H_{min}(z)$ *and* $H_{ap}(z)$.



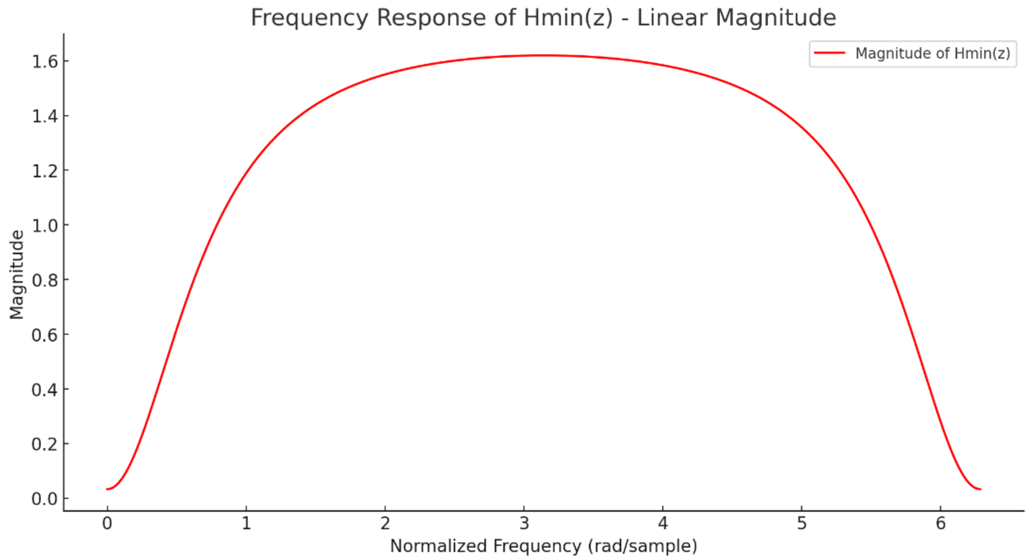




Q8: We plot the frequency response of $H_{min}(z)$ and $H_{ap}(z)$:



This graph shows the frequency response of the **all-pass filter** ,Since it is an all-pass system, its magnitude response remains **constant** across all frequencies, meaning it does not alter the amplitude of the input signal but only affects the phase.



This graph displays the frequency response of the **minimum-phase system**. Unlike the all-pass filter, the magnitude response varies with frequency, meaning it **modifies the amplitude** of different frequency components in the signal.

Q9: We define the correction system as

$$\frac{\mathbf{1}}{\boldsymbol{H(z)}} = \frac{(z-0.5)^2}{\left(z-1.1e^{j0.01}\right)\left(z-1.1e^{-j0.01}\right)(z-0.5)^2}$$

We take the inverse Z-transform to find the difference equation:

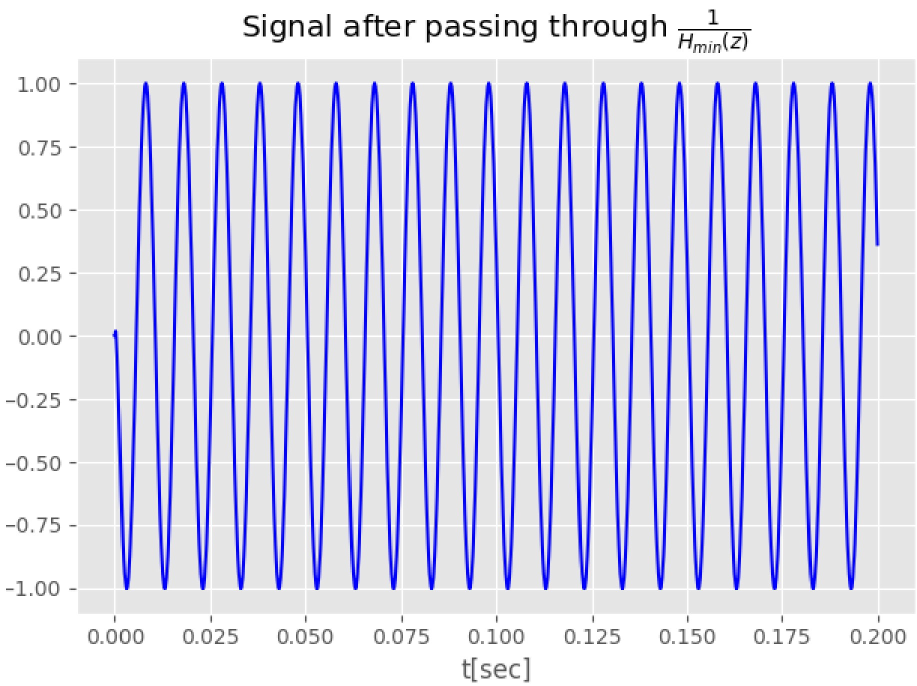
$$H_{correction} = \frac{Y(z)}{X(z)} = \frac{(z-0.5)^2}{\left(z-1.1e^{j0.01}\right)\left(z-1.1e^{-j0.01}\right)}$$

$$\frac{Y(z)}{X(z)} = \frac{1+0.3z^{-1}-0.1z^{-2}}{1.2-1.6z^{-1}+0.5z^{-2}}$$

$$1.2y[n]-1.6y[n-1]+0.5y[n-2] = -x[n]-0.3x[n-1]-0.1x[n-2]$$

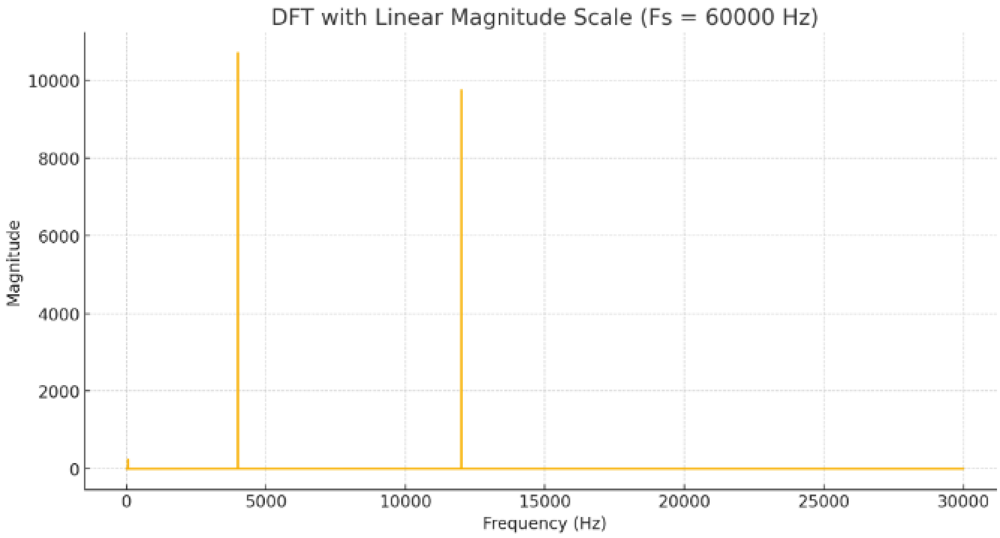
$$y[n] = \frac{-x[n]-0.3x[n-1]-0.1x[n-2]+1.6y[n-1]-0.5y[n-2]}{1.2}$$

Q10+11: We pass the signal through the correction system and plotted the result in time:

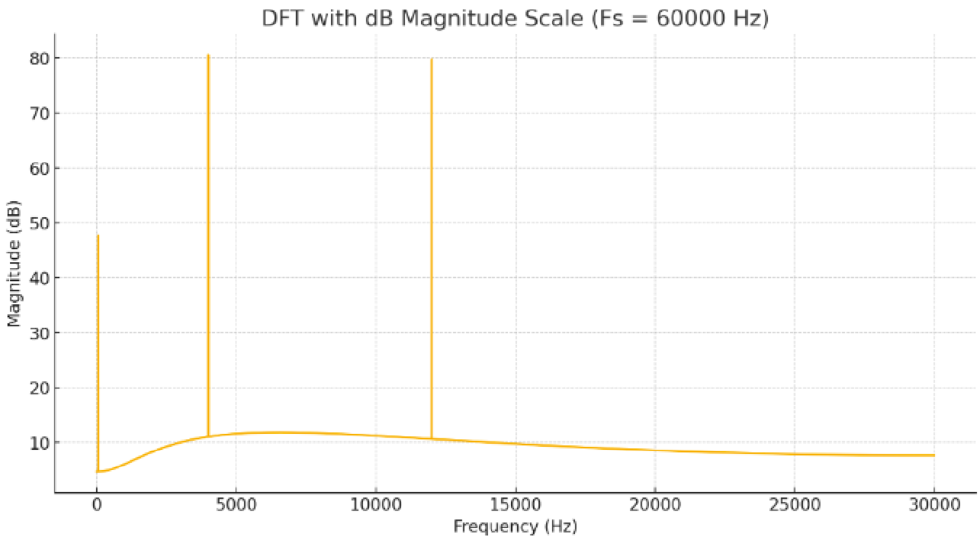


Q12: We plotted the DFT of the output signal:

DFT With Linear Magnitude Scale (Fs = 60000 Hz)



DFT With DB Magnitude Scale (Fs = 60000 Hz)



Linear Magnitude Scale with Frequency Axis in Hz:

- The x-axis is the frequency in Hz up to Fs/2=30,000 Hz.
- The y-axis shows the linear magnitude of the DFT.

dB Magnitude Scale with Frequency Axis in Hz:

- The x-axis is the frequency in Hz up to Fs/2=30,000 Hz.
- The y-axis shows the magnitude in decibels (dB).

Q13: The signal is made out of several frequencies. From the DFT Find the frequencies in the input signal using software:

	Peak Index	Frequency (Hz)	Magnitude
1	800	4000.0	10701.640713284743
2	2400	12000.0	9753.29603338635

Q14. We find the frequencies and design a 2^{nd} order notch filter to block the noise:

we assume the noise as seen in the DFT is the 4khz frequency, which we will attempt to filter out.

$$100Hz \rightarrow \Omega = 2\pi\frac{f}{f_s} = 2\pi\frac{100}{20000} = \frac{\pi}{100}\left[\frac{rad}{smp}\right]$$

So we place zeros at $\frac{2}{5}\pi$ *and at* $-\frac{2}{5}\pi$ *(2^{nd} order, plus, we want the system to be real)*

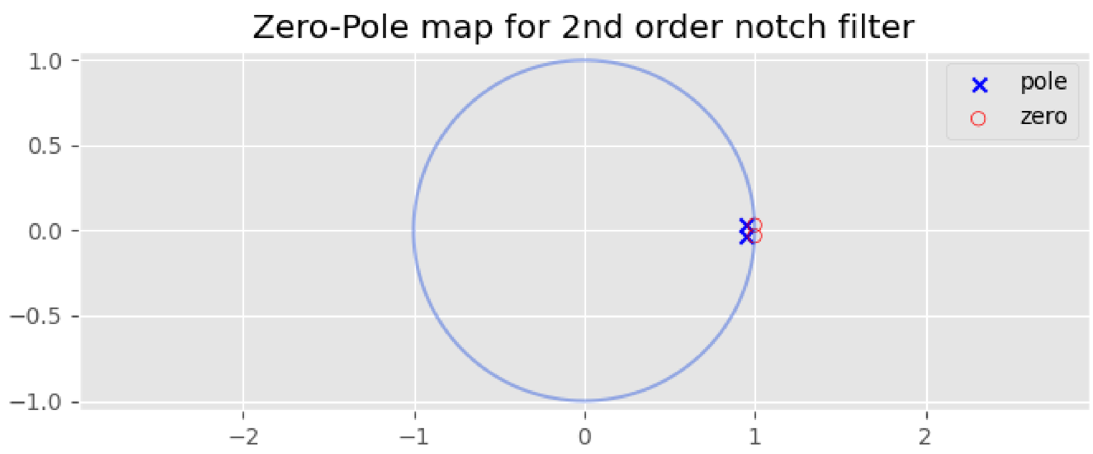
$$1st\ zero: e^{\frac{j2}{5}\pi} = \cos\left(\frac{\pi}{100}\right) + jsin\left(\frac{\pi}{100}\right)$$

$$1st\ pole: 0.95\left(\cos\left(\frac{\pi}{100}\right) + jsin\left(\frac{\pi}{100}\right)\right)$$

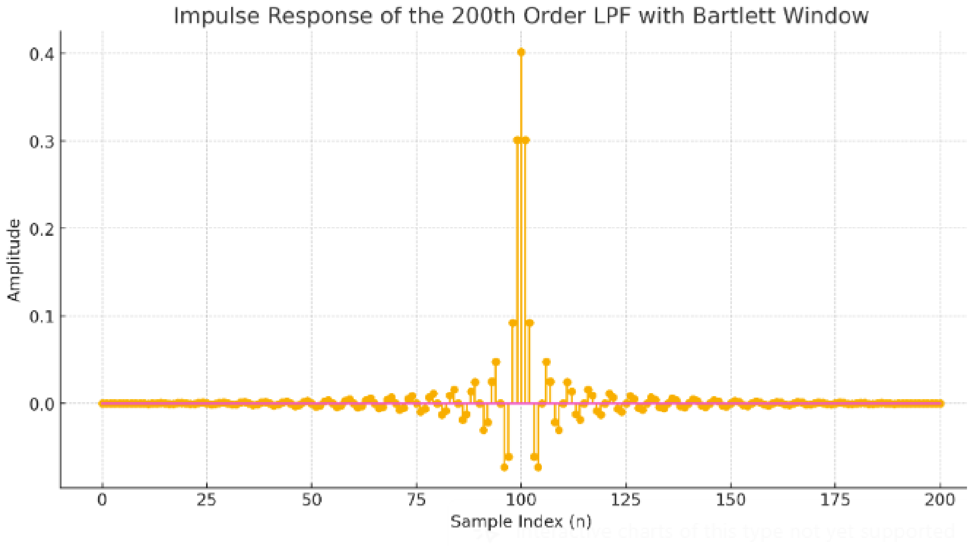
$$2nd\ zero: e^{-\frac{j2}{5}\pi} = \cos\left(\frac{\pi}{100}\right) - jsin\left(\frac{\pi}{100}\right)$$

$$2nd\ pole: 0.95\left(\cos\left(\frac{\pi}{100}\right) - jsin\left(\frac{\pi}{100}\right)\right)$$

We plot the zero-pole map and get:



Impulse Response Of The 200th Order LPF With Bartlett Window



Q15: The transfer function is:

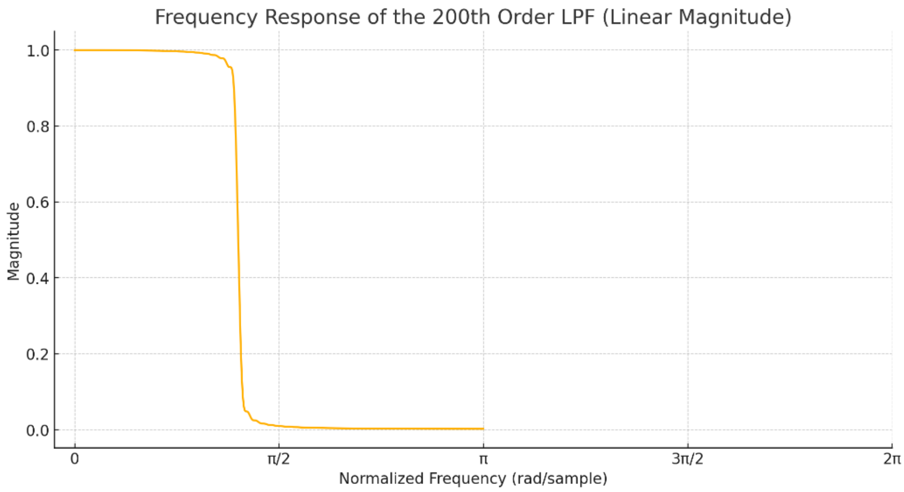
$$H(z) = \frac{\left(z - e^{j\frac{\pi}{100}}\right)\left(z - e^{-j\frac{\pi}{100}}\right)}{\left(z - 0.95e^{j\frac{\pi}{100}}\right)\left(z - 0.95e^{-j\frac{\pi}{100}}\right)} = \frac{z^2 - 2zcos\left(\frac{\pi}{100}\right) + 1}{z^2 - 1.9z\cos\left(\frac{\pi}{100}\right) + 0.9025}$$

$$= \frac{z^{-2} - 2\cos\left(\frac{\pi}{100}\right)z^{-1} + 1}{0.9025z^{-2} - 1.9\cos\left(\frac{\pi}{100}\right)z^{-1} + 1}$$

And to get the frequency response, we implement $z = e^{j\omega}$:

$$H(\omega) = \frac{e^{-2j\omega} - 2\cos\left(\frac{\pi}{100}\right)e^{-j\omega} + 1}{0.9025e^{-2j\omega} - 1.9\cos\left(\frac{\pi}{100}\right)e^{-j\omega} + 1}$$

Q16: We plotted the frequency response of the notch filter and got:



Explanation: *The graph represents the **frequency response** of the designed **200th-order Low-Pass Filter (LPF)** using a **Bartlett window**.*

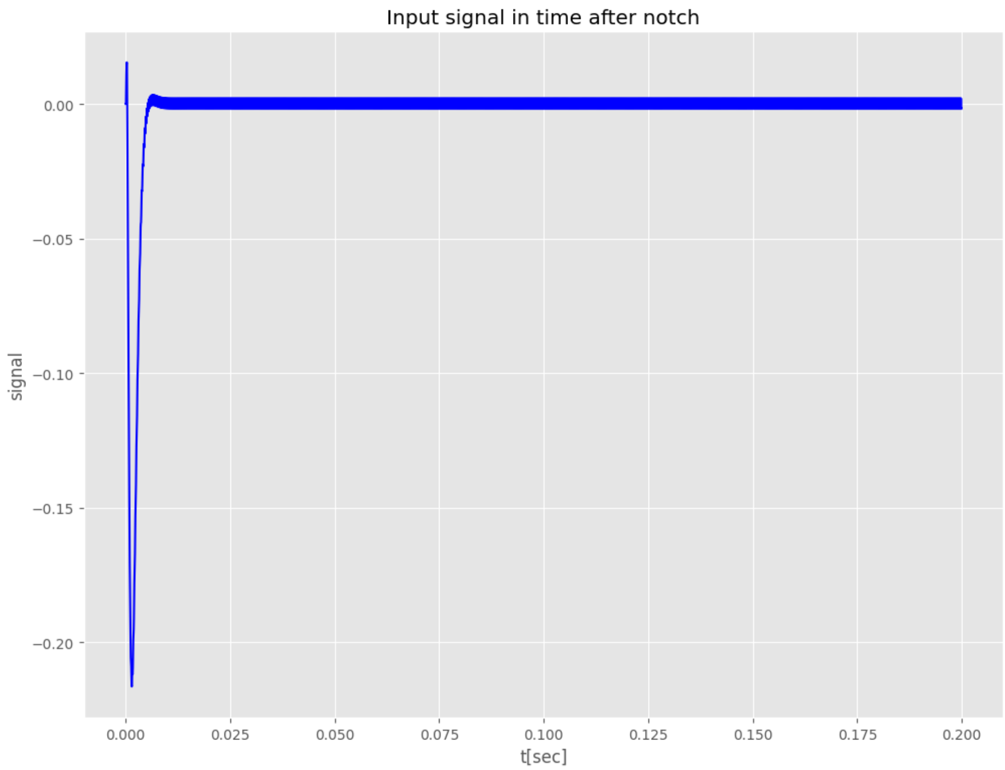
Q17: We take the transfer function and find the difference equation:

$$\frac{Y(z)}{X(z)} = \frac{z^{-2} - 2\cos\left(\frac{\pi}{100}\right)z^{-1} + 1}{0.9025z^{-2} - 1.9\cos\left(\frac{\pi}{100}\right)z^{-1} + 1}$$

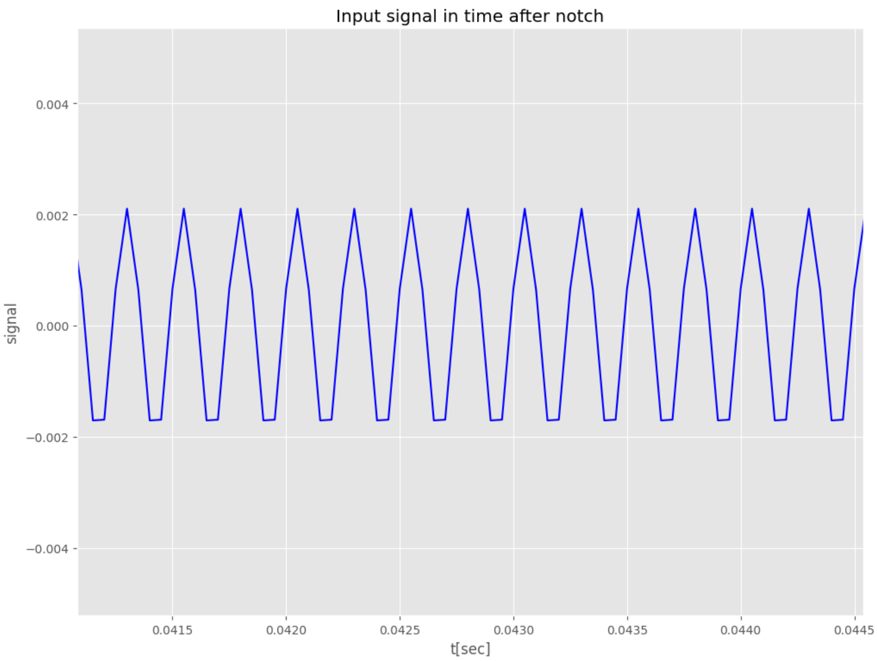
$$0.9025y[n-2] - 1.9\cos\left(\frac{\pi}{100}\right)y[n-1] + y[n] = x[n-2] - 2\cos\left(\frac{\pi}{100}\right)x[n-1] + x[n]$$

$$y[n] = x[n-2] - 2\cos\left(\frac{\pi}{100}\right)x[n-1] + x[n] - 0.9025y[n-2] + 1.9\cos\left(\frac{\pi}{100}\right)y[n-1]$$

Q18+19: We pass the signal through the notch filter and plot the result in time:

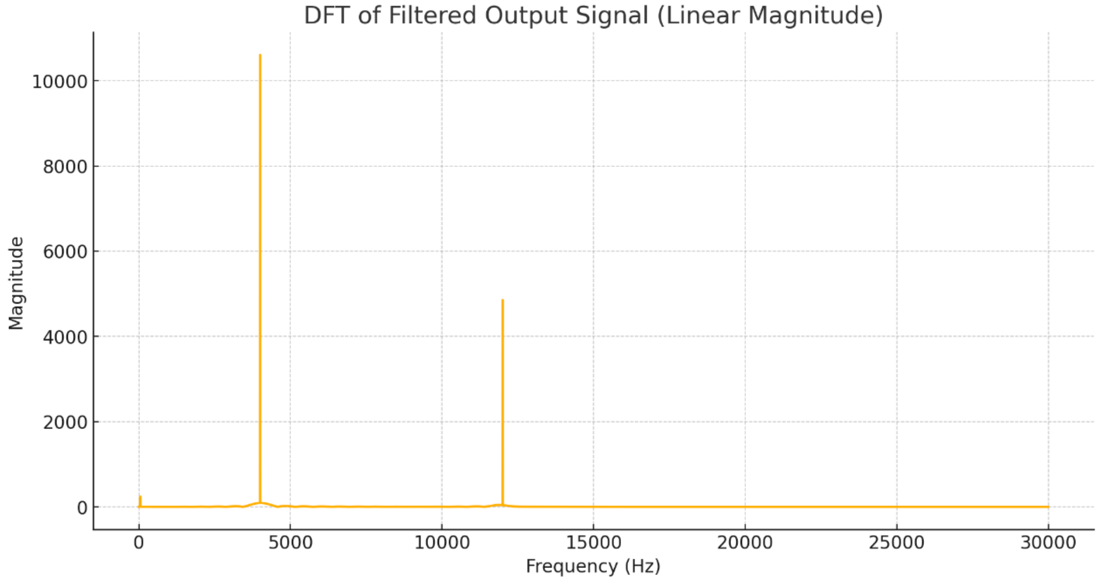


We can see the transient state at the start of the graph, and if we zoom into the stable part of the signal:



We can see that the time period is about 0.00025 which corresponds to the 4kHz frequency. Since the sampling frequency is 20kHz, in relation to 4kHz is only 5 times more, which is why

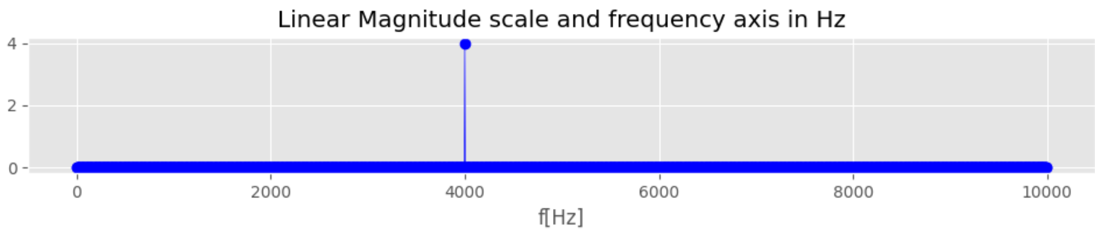
Q20: We plot the DFT of the notch filter's output signal:



We can see the DFT of the signal only holds the 4kHz frequency impulse, which is what we expected, but since the signal still has the transient state part, we are still left with extra low frequencies.

Q21: Summary

After receiving a sampled signal, we pass it through two fixing system. The first is the minimum phase system, which corrects the distortion in the signal, and the second is the notch filter, which removes the lower frequency (100Hz) from the signal after which, presumably, we get the reconstructed signal, cleaned from the noise and distortions.



*The **LPF successfully isolates the low-frequency content** of the signal, making it useful for filtering out high-frequency noise while preserving important low-frequency information.*

*The DFT confirms that the filter performs well in both **attenuating high frequencies** and **preserving low frequencies**, as expected for a low-pass filter.*