Programming Assignment

Team Members

Bhuminjay Soni(B20CS009)

Siddharth Singh(B20EE067)

1. One of the many applications of Internet of Things (IoT) consists of continuous monitoring of temperature in an area. To that end, several temperature sensors are installed at different locations. These sensors measure and store the recorded value of temperature over time. However, due to limitations of hardware, the sensor memory needs to be cleared periodically and this is done by transmitting the stored values to a base unit. Assume that $x[n]$ denotes the samples of the true value of temperature recorded by a sensor. However, it is found that the received signal $y[n]$ at the base unit suffers from blur distortions and noise (additive). Hence, the signal $y[n]$ needs to be first processed so that we can recover $x[n]$ from it. Assume that blur happens via a system characterized an impulse response $h[n] = \frac{1}{16}\begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix}$ (assume that the center value of $\frac{6}{16}$ corresponds to $n = 0$). Then, implement the following two approaches to recover the original signal $x[n]$ from distorted signal $y[n]$.

   1. First remove noise and then sharpen (deblur). Let the resulting signal be $x_1[n]$.
   2. First sharpen (deblur) and then remove noise. Let the resulting signal be $x_2[n]$.

   Now, compare $x_1[n]$ and $x_2[n]$ with $x[n]$. What conclusions can you draw from your observations? Also, explain your observations from a theoretical perspective if possible.

<u>Solution</u>

Following concepts and techniques, we have used in this assignment:

- ## <u>Denoising:</u>

The problem in signal denoising is that instead of the actual signal samples, Y, a noisy version of the corresponding observations, *y*, is available; that is, y=Y+η, where *η* is the vector of noise samples. In the noisy case, however, this is a problem. The reason for this is that noise is, by definition, high-frequency information. So when we try to deconvolve Y, the noise is amplified to an almost infinite extent.

We have used local mean signal denoising or averaging to remove noise from the signal. Since neighboring signal values generally (except at ends) have similar values. Therefore, we have replaced a signal value with the average of its neighbors.

So to Denoise the signal :

Replacing y[n] = (y[n-2] + y[n-1] + y[n] + y[n+1] + y[n+2] /5)

First & Last values are taken care off by appending them 2 times at the beginning and the end of the array respectively.

• Deblurring :

We have used the convolution property for deblurring. If x[n], h[n], and y[n] are the input, impulse response, and the output, respectively of an LTI system, so that

y[n] = h[n] * x[n]

Here in the Assignment we are given h[n] and y[n] we have to recover x[n] from it.

So to Recover x[n] we applied following approach:

We applied the Convolution property of Fourier transform :

$$\boxed{Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega}),}$$

Then after we can get x[n] by taking inverse Discrete-time Fourier transform of X(e∧jw).

For finding DTFT and Inverse DTFT we have used the following expressions :

1. The DTFT y[k] of length N of the length - N sequence x[n] is defined as :

$$y[k] = \sum_{n=0}^{N-1} e^{-2\pi j \frac{kn}{N}} x[n]$$

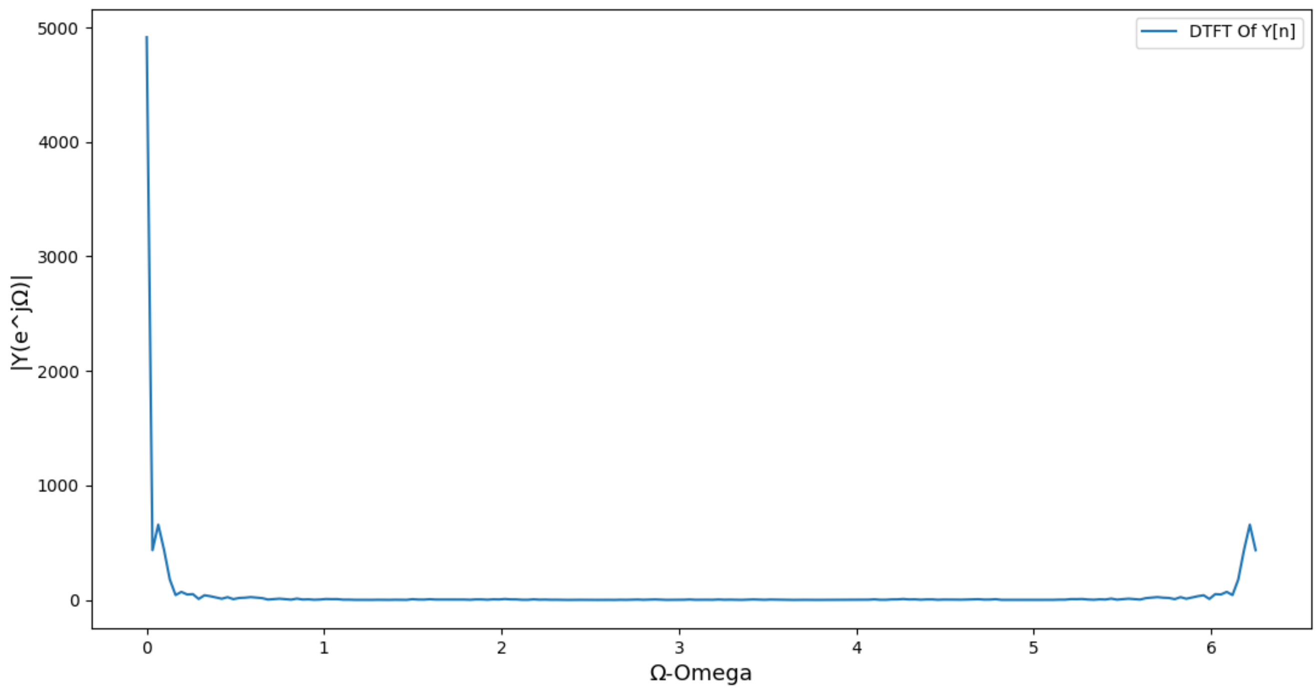
2. The inverse transform is defined as follows :

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} e^{2\pi j \frac{kn}{N}} y[k]$$

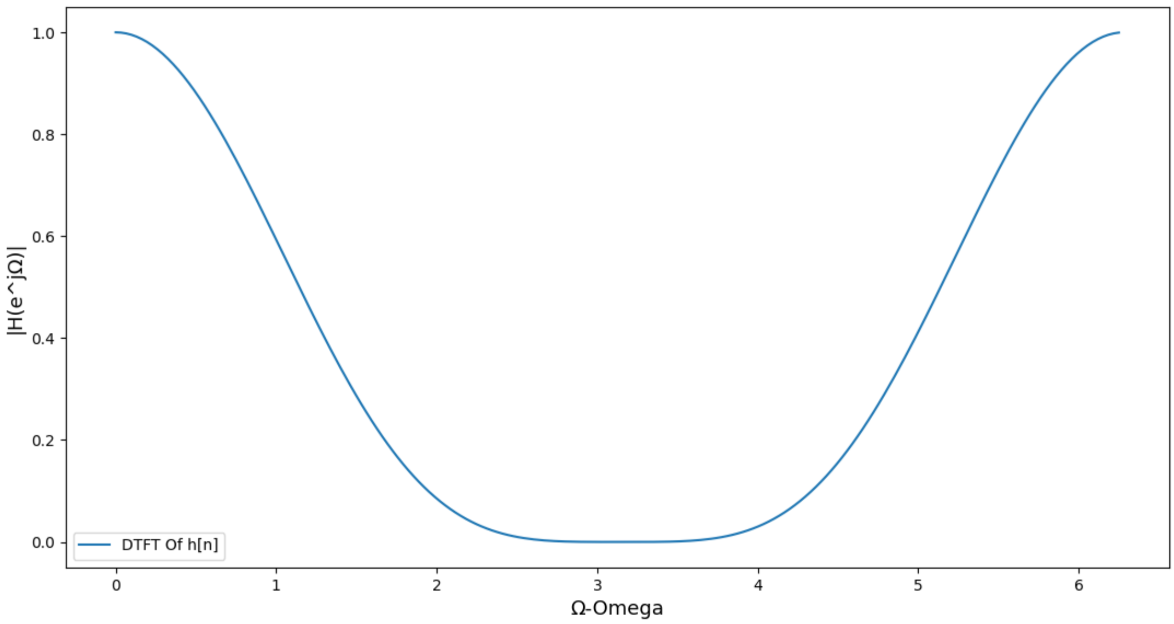
We have used following approaches:

1. First Denoised then Deblurred:

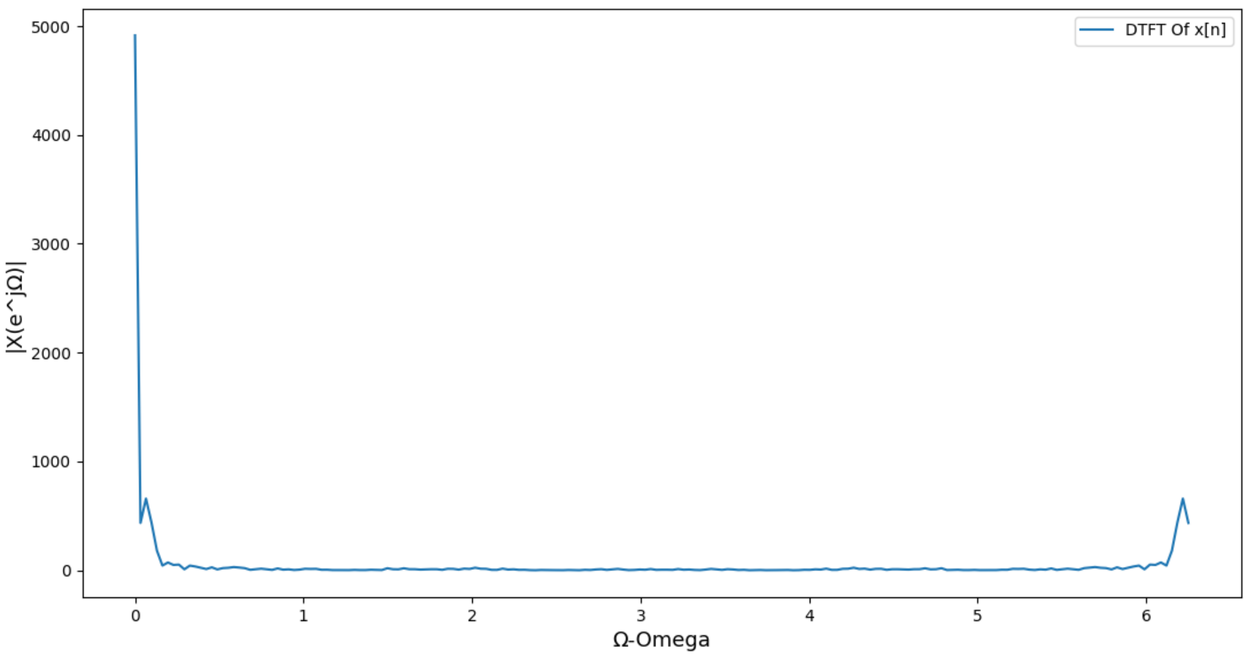
DTFT of Denoised y[n]:



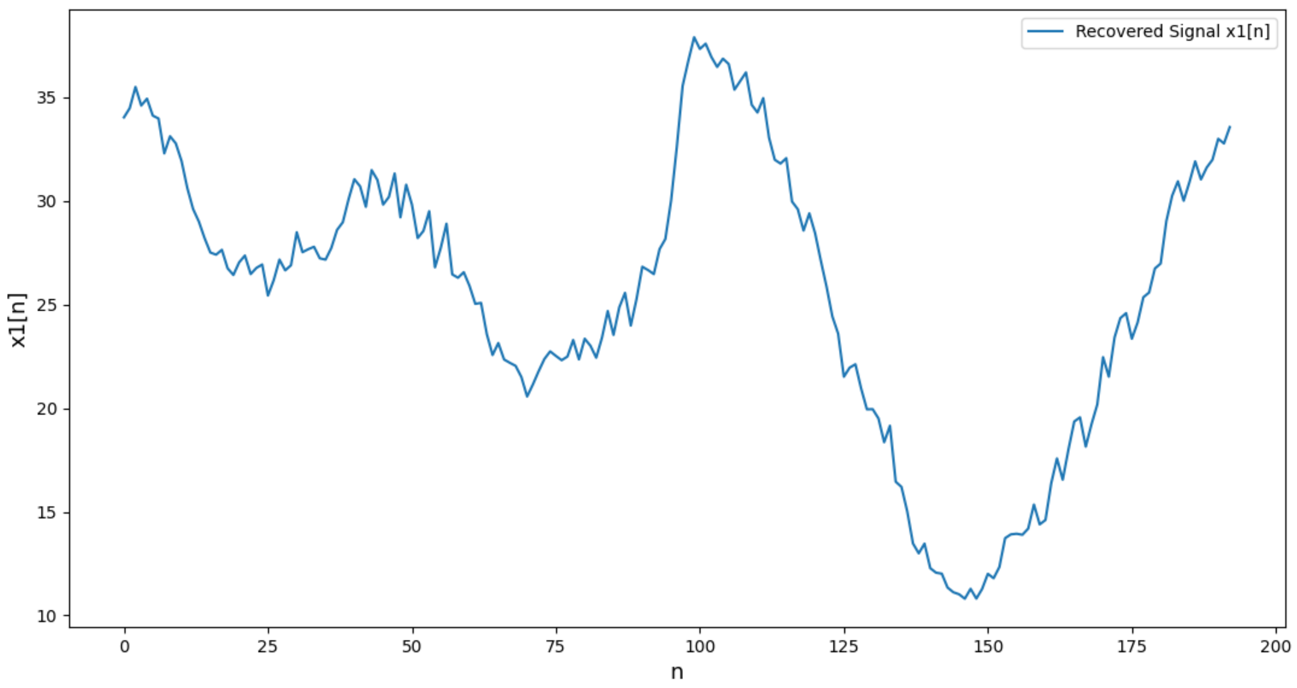
DTFT of h[n]:



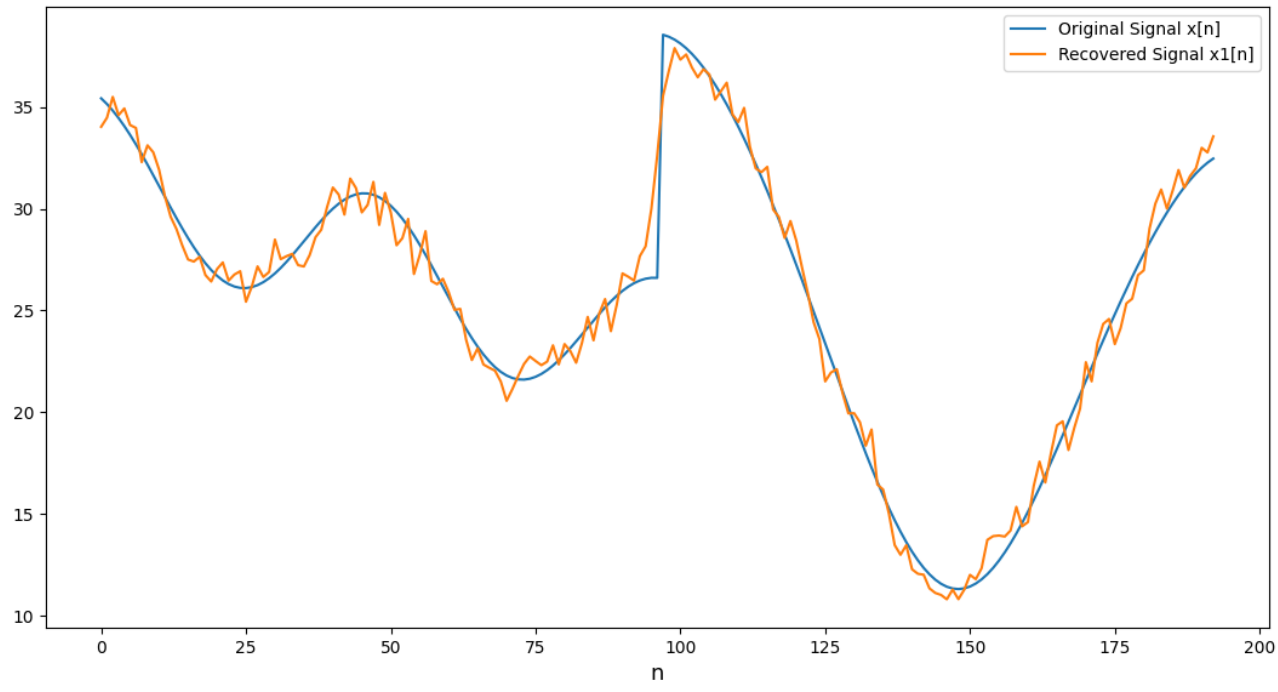
DTFT of x1[n] :



Recovered signal x1[n] :

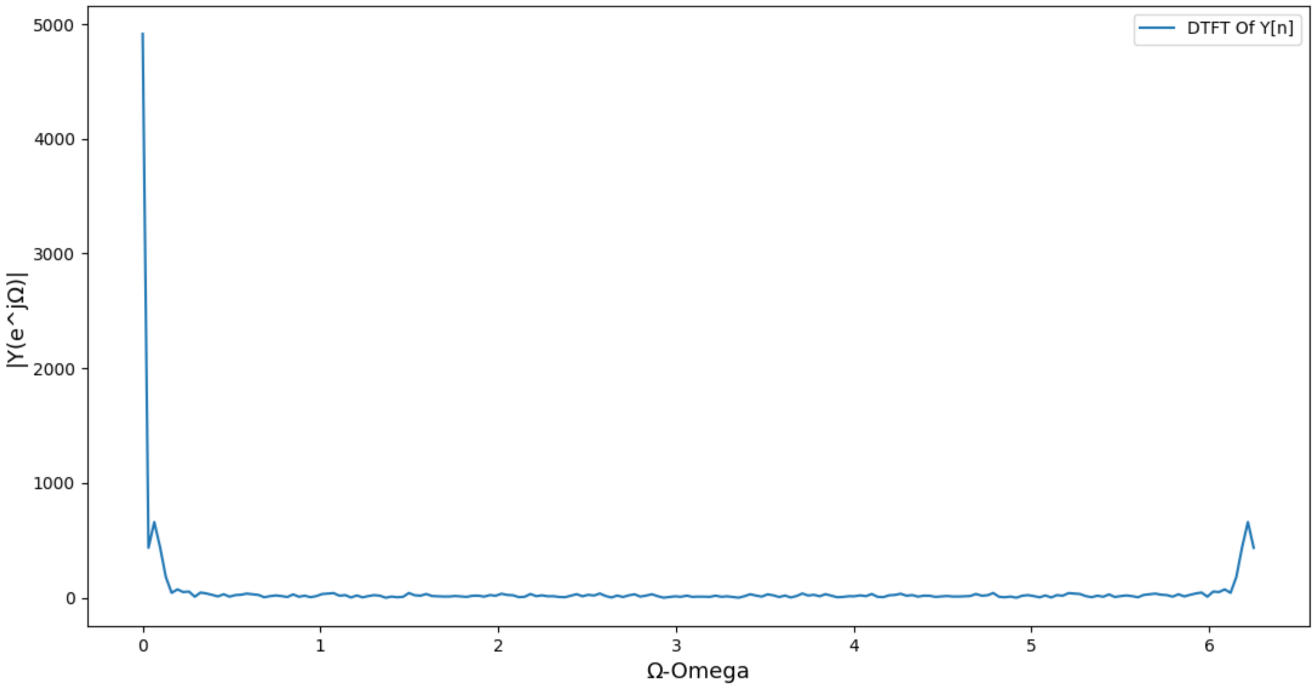


Comparision of x1[n] and x[n] :

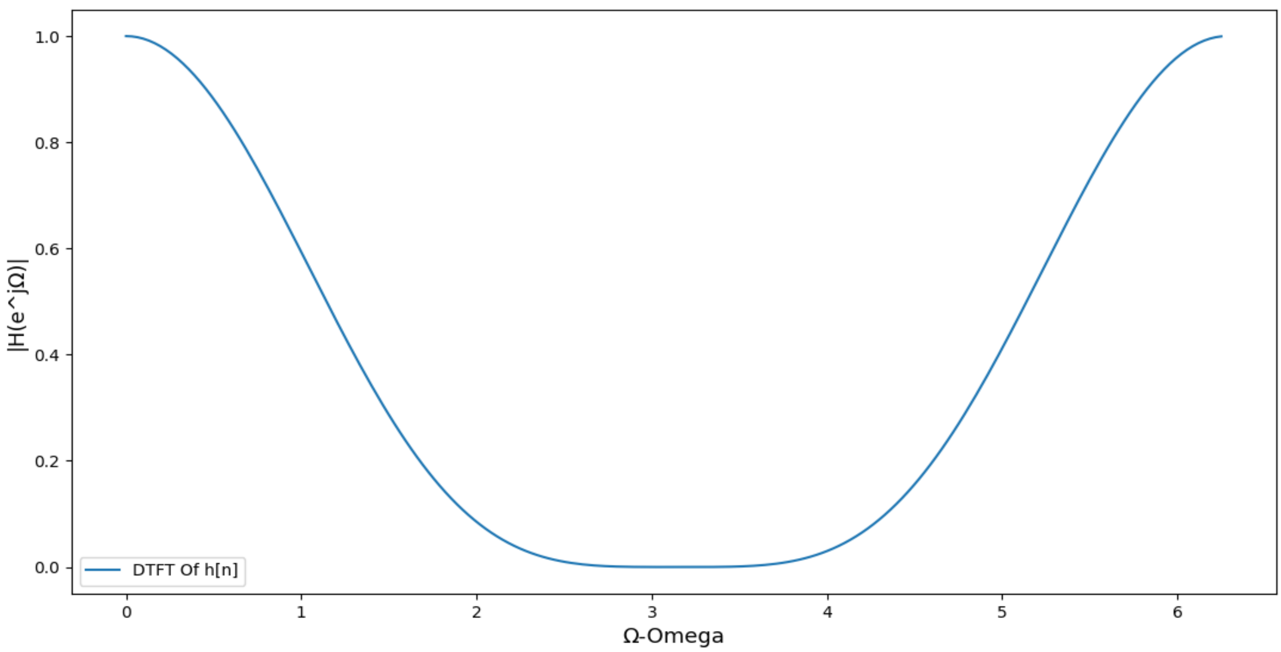


2. Deblurred first then Denoised :

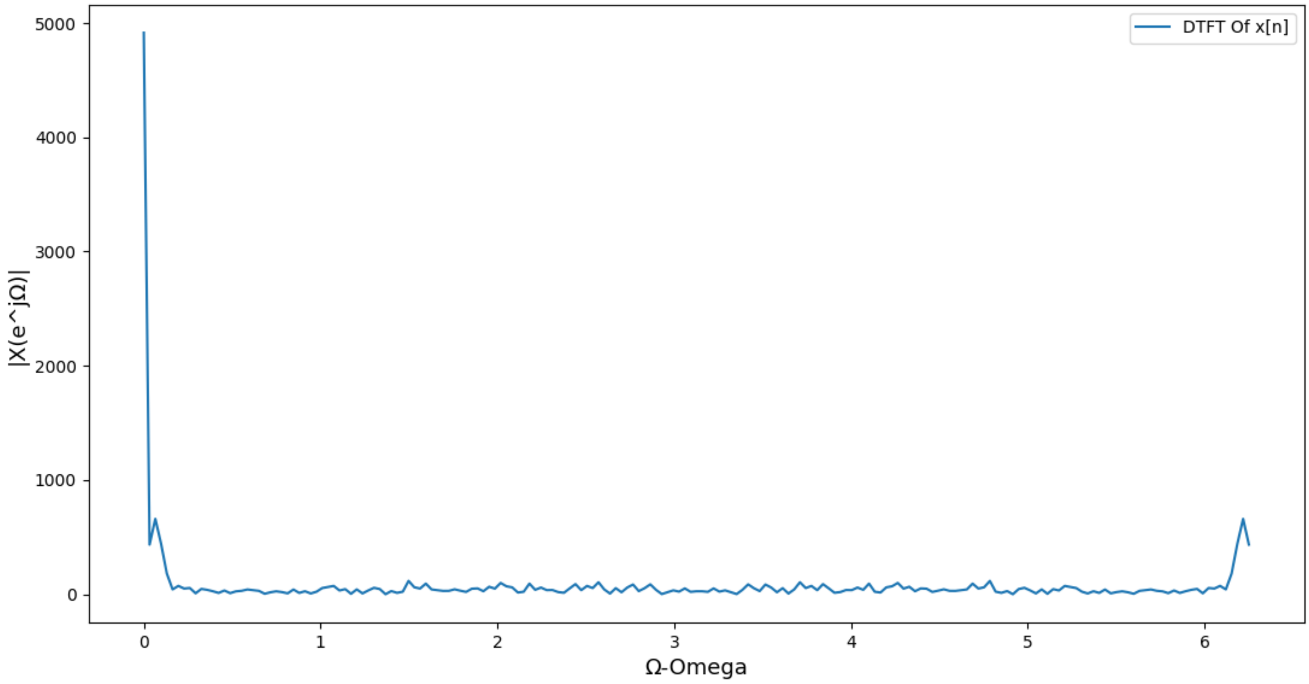
DTFT of y[n]:



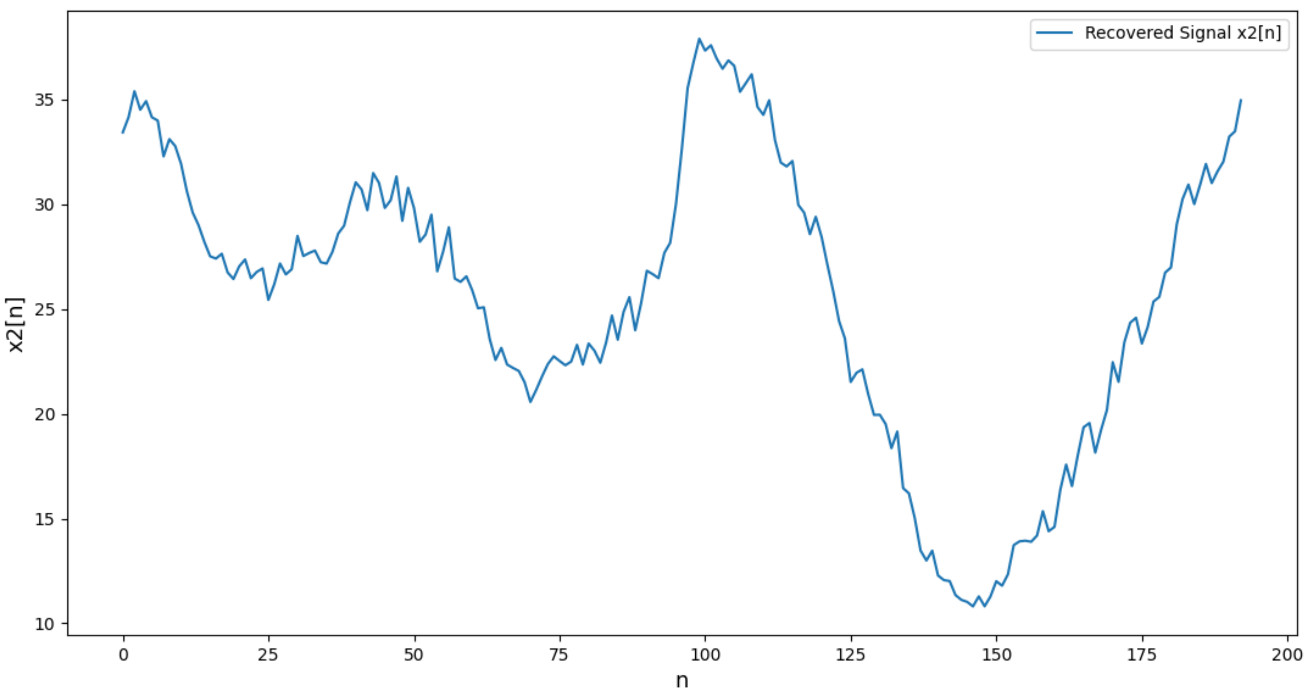
DTFT of h[n]:



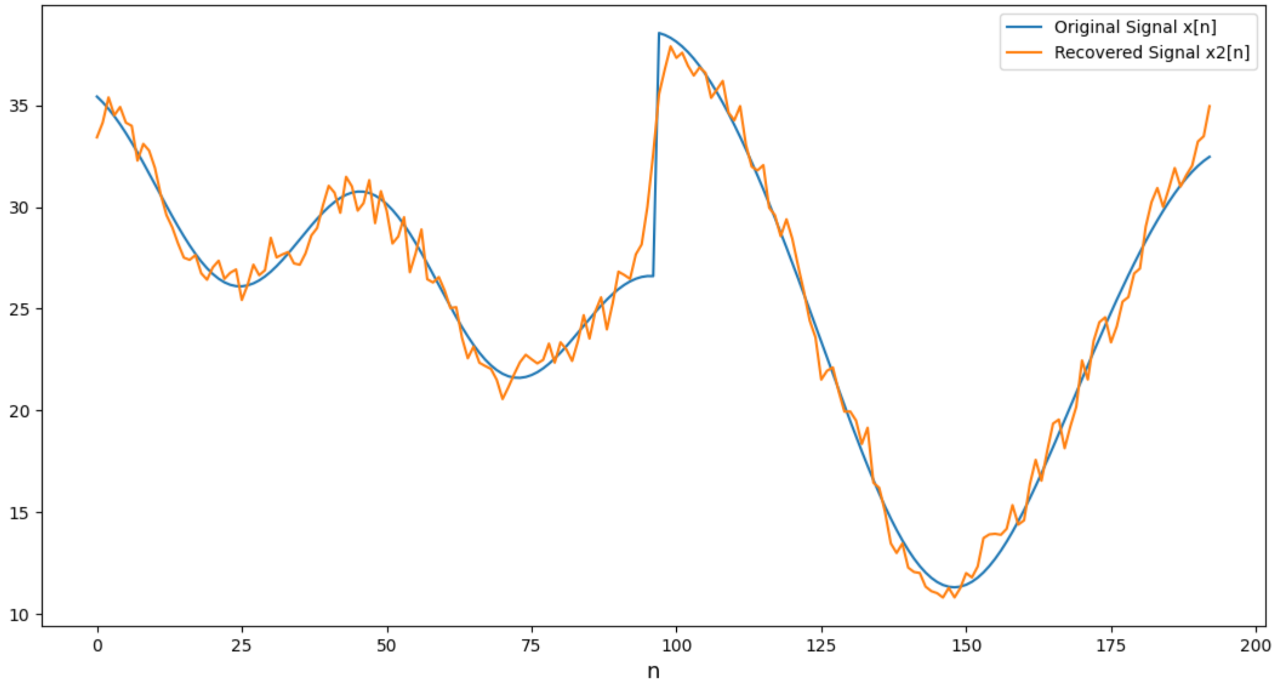
DTFT of x2[n](Noised signal) :



Recovered signal:



Comparison of x[n] and x2[n] :



Conclusion

We are given the received signal y[n]. However, it is found that the received signal y[n] at the base unit suffers from blur distortions and noise(additive). So, we processed y[n] by following the two approaches:

1. We first removed noise and then sharpen(deblur) and obtained signal x1[n] .

   Resulting Mean Squared Error (MSE) between x[n] and x1[n] = 0.927

2. In the second approach, we first sharpen (deblur) and then remove noise. We obtained x2[n] by this approach.

   Resulting Mean Squared Error (MSE) between x[n] and x2[n] = 0.974

We finally conclude that approach 1(Denoising first then Deblurring) is better than approach 2(Deblurring first then Denoising) in recovering the original signal.

Contribution of Team members

We both contributed equally in this programming assignment.

Siddharth Singh (B20EE067)

Bhuminjay Soni (B20CS009)