Project Report

Title: Synthesis of Audio with Interferences and Echo, and Recovery of Original Audio

Project Description:

In this project, the audio signal is synthesized with the interference of signals with original audio and the original audio echo is also added to signal. The project's original audio is 'HappyBirthday' sound and two interference signals are 'Bird_Chirping' and 50 Hz Electrical Grid Interference. The echo of the original audio (delayed signal) is also interfering with the audio. The aim is to recover the original audio from all of these interferences to achieve distinguishable audio recovery and minimal MSE (mean square error).

Introduction:

In this project, we were required to synthesize audio signal with added interferences and echo. The project code is divided into two main code files, one generates the audio and one processes the generated audio and aims to recover the original signal from corrupted synthesized signal. The processing is done with digital signal processing techniques and mean square error is calculated and plotted for each step to visualize the improvement in recovered audio signal.

Project Files:

The project files consist of code files and audio files.

Code Files:

1. **Synthesize_Received_Audio.m:** It synthesizes the corrupted audio signal by adding interferences and echo.

2. **Process_Received_Audio.m:** It processed the received corrupted signal to recover the original audio

3. **Freq_Plot_1.m:** It plots the frequency domain of the signal in logarithmic scale.

4. **MSE.m:** It calculates the Mean Square Error between two signals.

Audio Files:

1. **HappyBirthday.mp3:** It is the original Happy Birthday audio.

2. **Bird_Chirping.mp3:** It is the bird chirping interference audio.
3. **Synthesized_Received_Audio.wav:** It is the synthesized received corrupted audio.

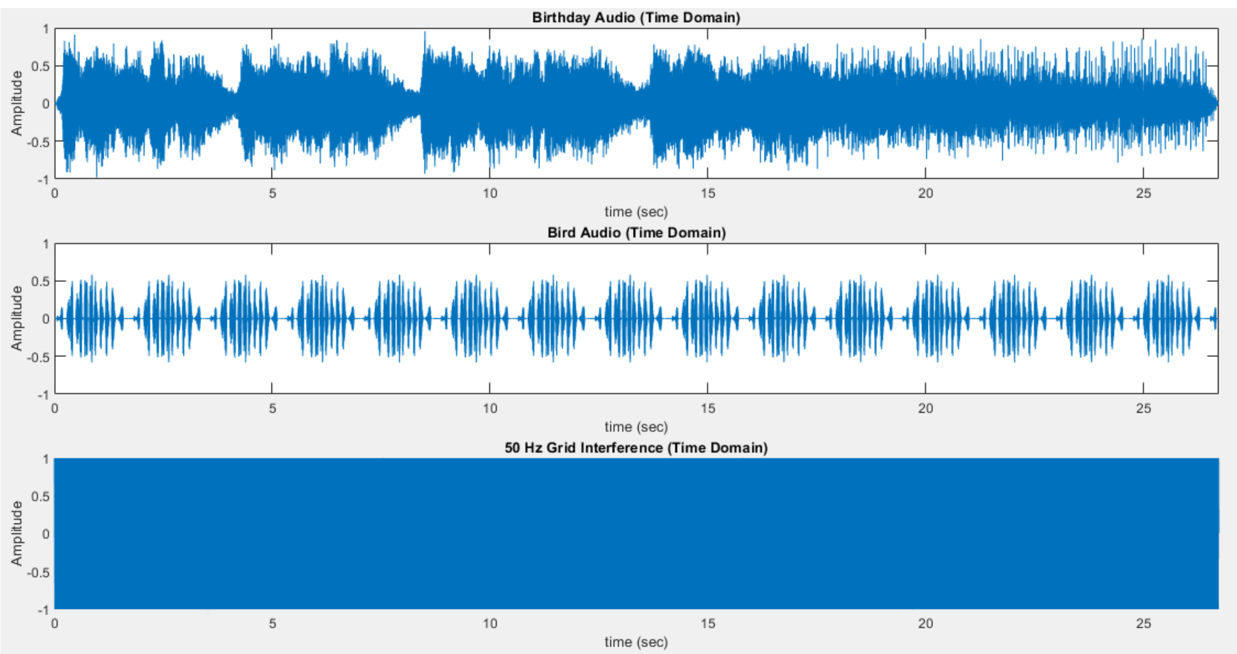
4. **Recovered_Received_Audio.wav:** It is the recovered received audio after processing.

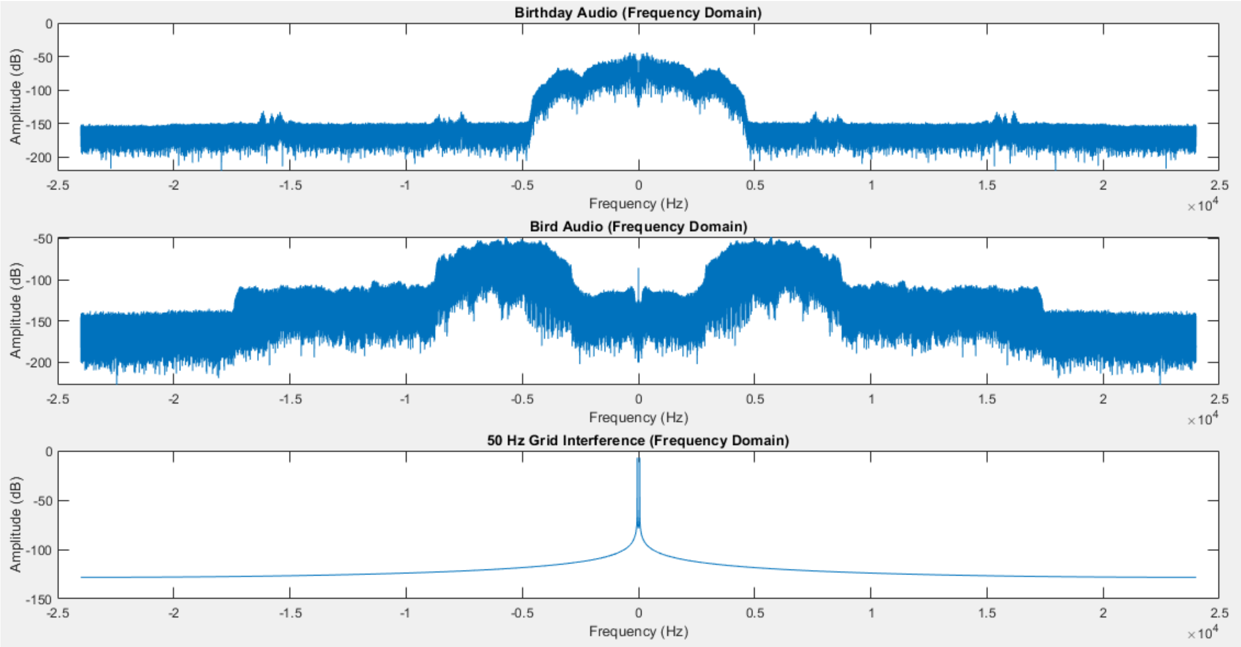
Algorithm Description:

The algorithm is distributed into two parts

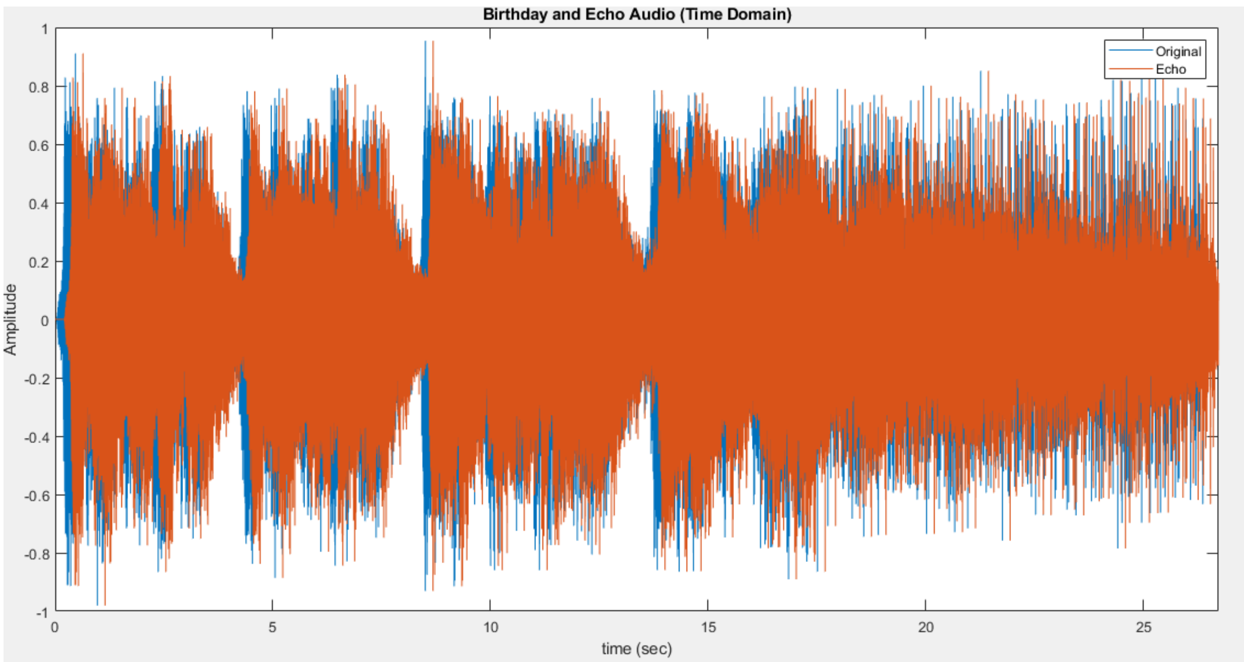
Synthesis of Corrupted Received Audio:

1. The audio files are loaded and the length of both files are made same by looping the shorter audio of bird chirping
2. The 50 Hz sinusoidal interference signal is generated with code.
3. The audio signals are plotted in both time and frequency domain for analysis.





4. The echo signal is simulated by delaying the original signal by fix number of samples calculated from echo reflection distance and time.



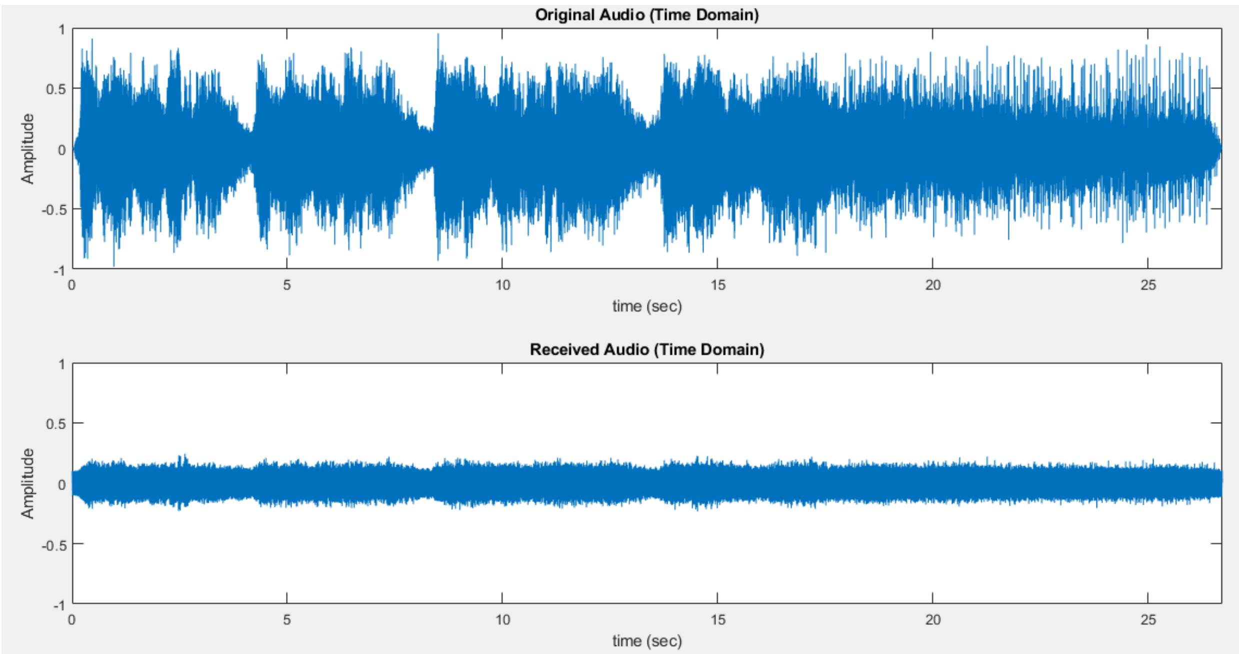
5. The original signal, interfering signals, and the echo signals are added and properly scaled to produce a corrupted synthesized received audio signal.
6. The synthesized signal is saved as a wav file for analysis and use in processing code.

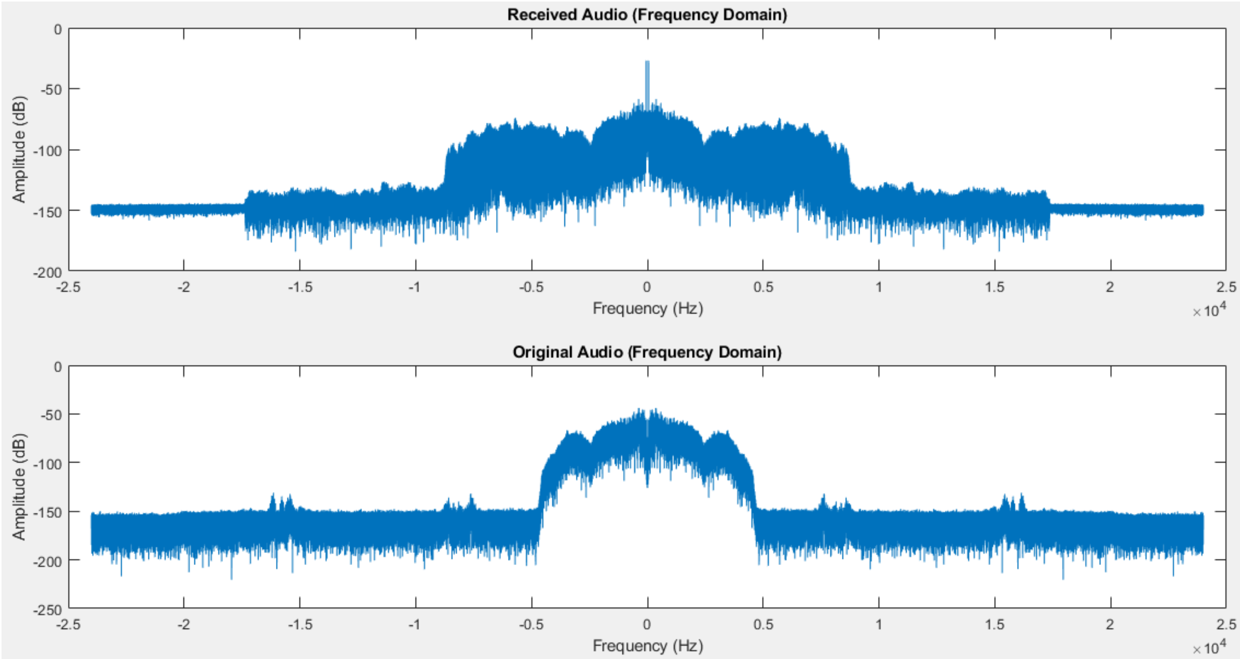
Recovery of Corrupted Received Audio:

1. The Audio files are loaded and the MSE (Mean Square Error) [1] is calculated between original sound and corrupted sound.

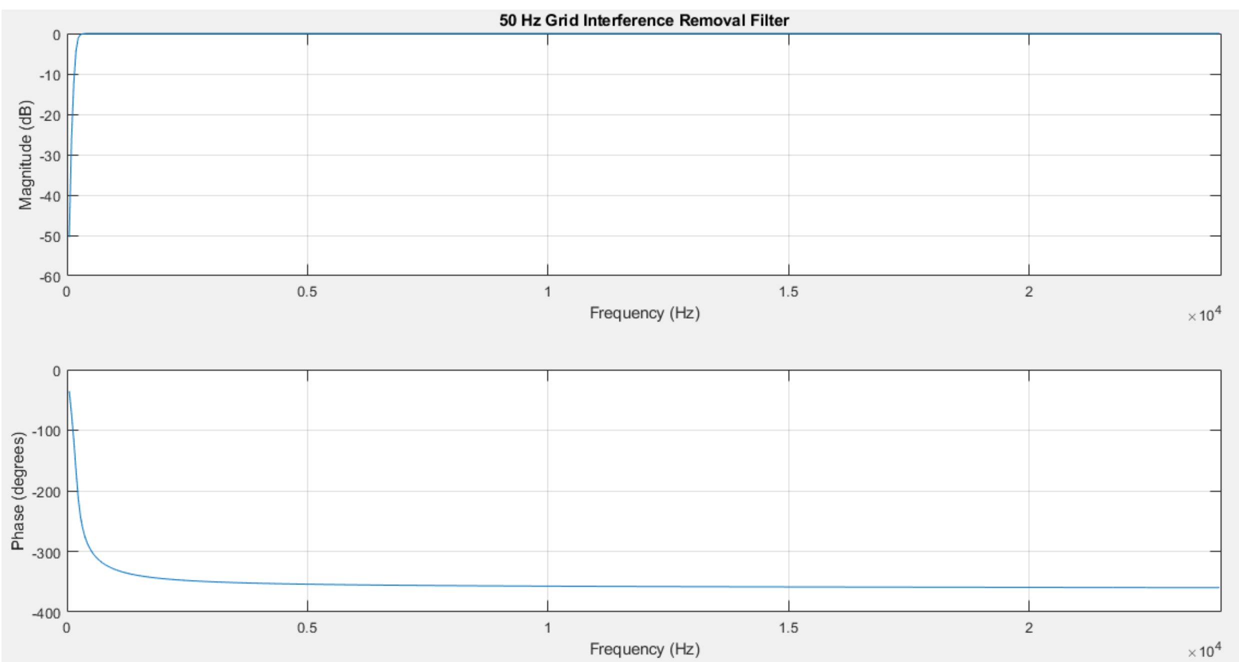
$$MSE = \frac{1}{N}\sum_{n=1}^{N}\left(x[n] - x_{corrupt}[n]\right)^2$$

2. The original and corrupted signals are plotted in time and frequency domain for analysis.

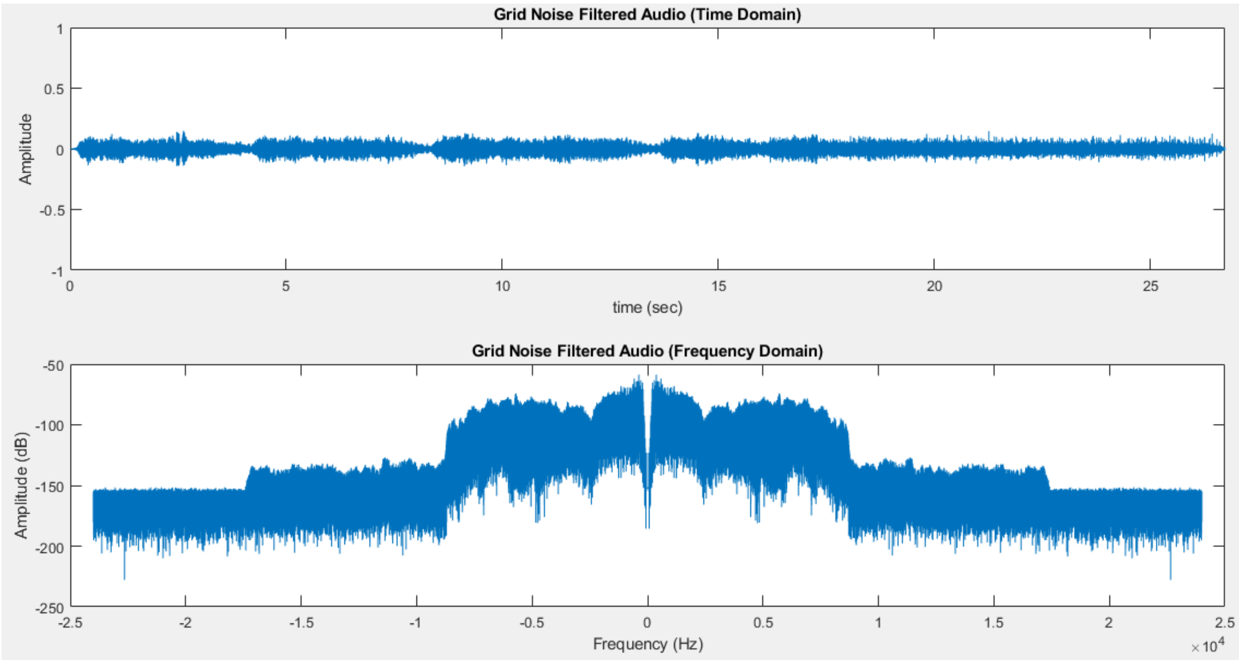




3. A 50 Hz Electrical Interference Removal filter is designed which is a high pass filter with cutoff = 200 Hz, filter order = 4, and filter type = butterworth. IIR [2] butterworth is chosen because it can provide good response at v low orders compared to FIR filters.

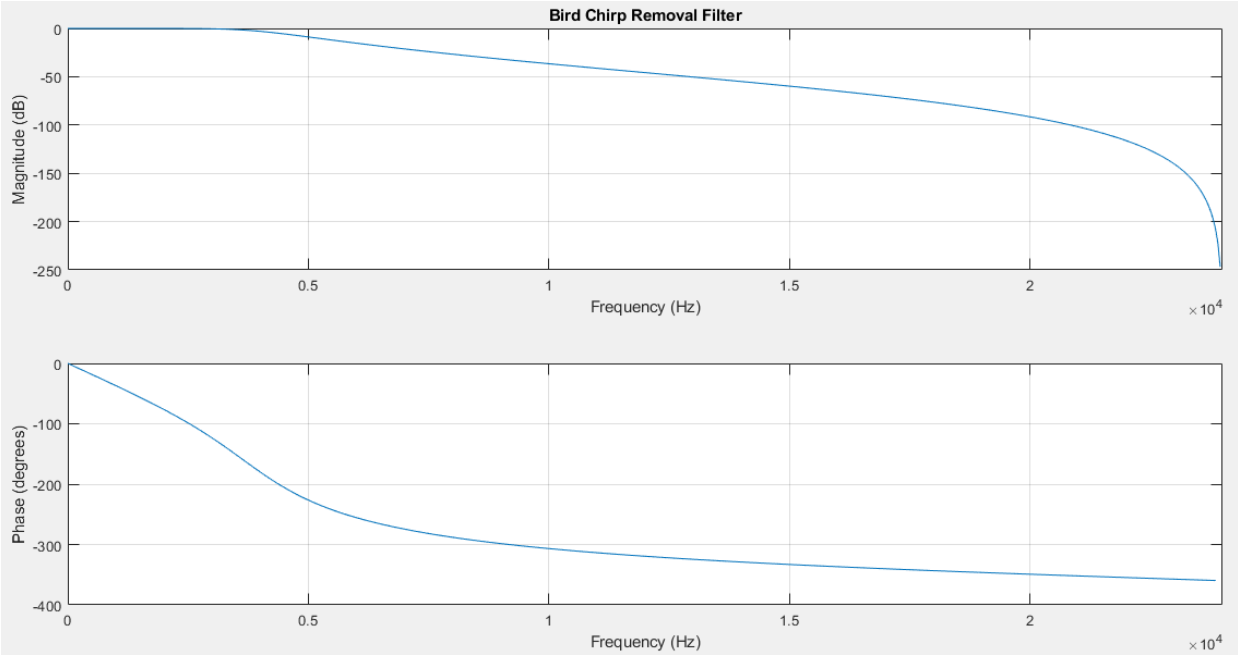


4. The designed filter is used to filter the signal for removal of 50 Hz interference. The zero phase filtering [3] is done to avoid phase shift which can cause difficulty in echo removal.
5. The filtered signal is plotted in time and frequency domain and MSE is calculated.

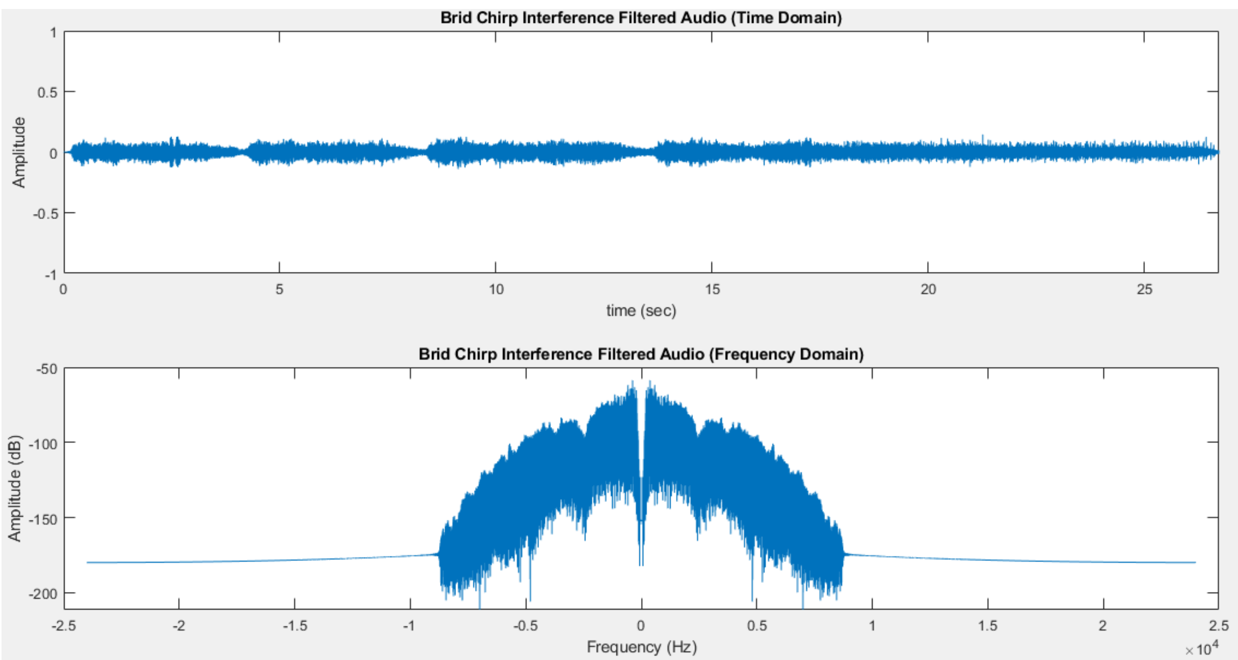


6. The Bird Chirp Interference removal filter is designed which is a low pass filter with cutoff at 4000 Hz since the bird chirping is a high frequency signal starting at almost 4000 Hz.

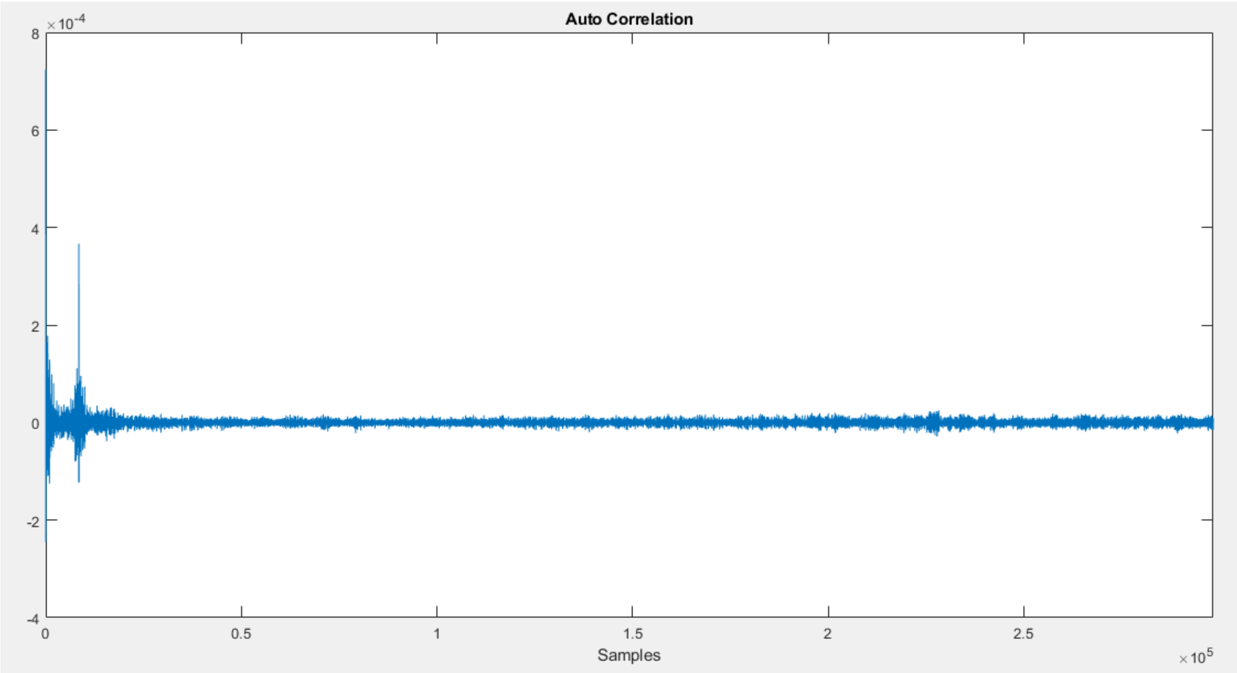
7. The designed filter response is plotted and the signal is filtered with zero phase filtering.



8. The filtered signal is plotted in time and frequency domain and MSE is calculated.



9. The Echo time delay needs to be determined. So the autocorrelation [4] of the signal is done. The location of the peak of autocorrelation tells the delay in samples of the echo signal.

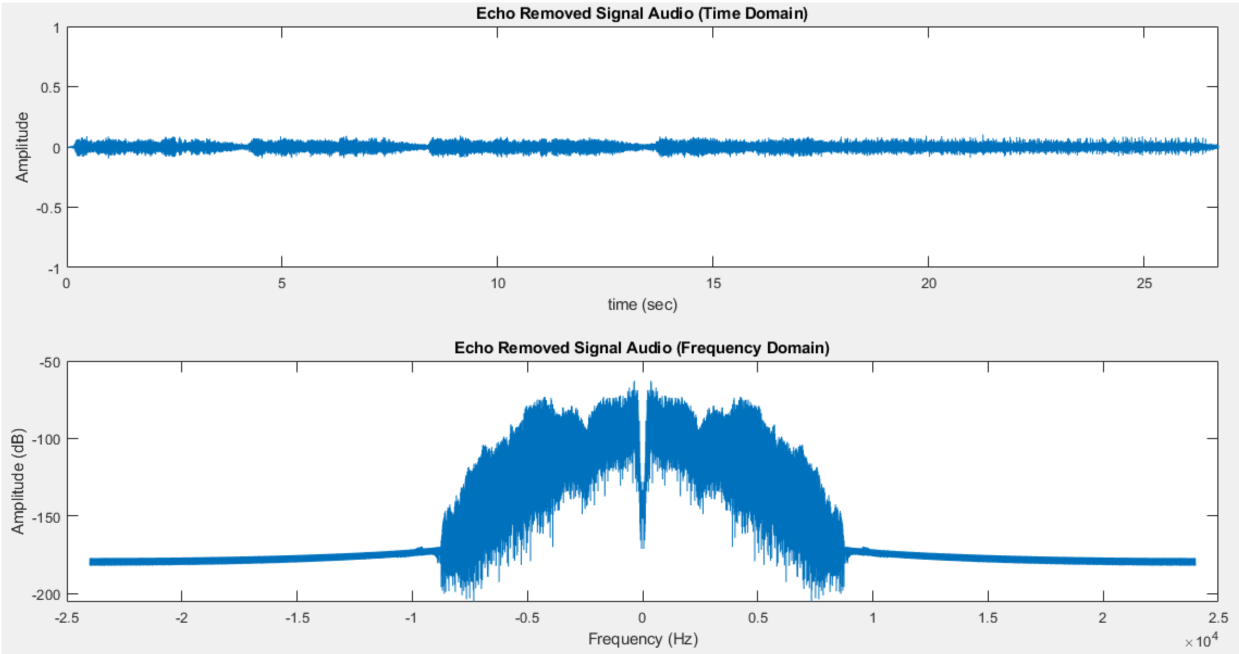


$$x[n] + \alpha x[n-\Delta] = x_{corrupt}[n]$$

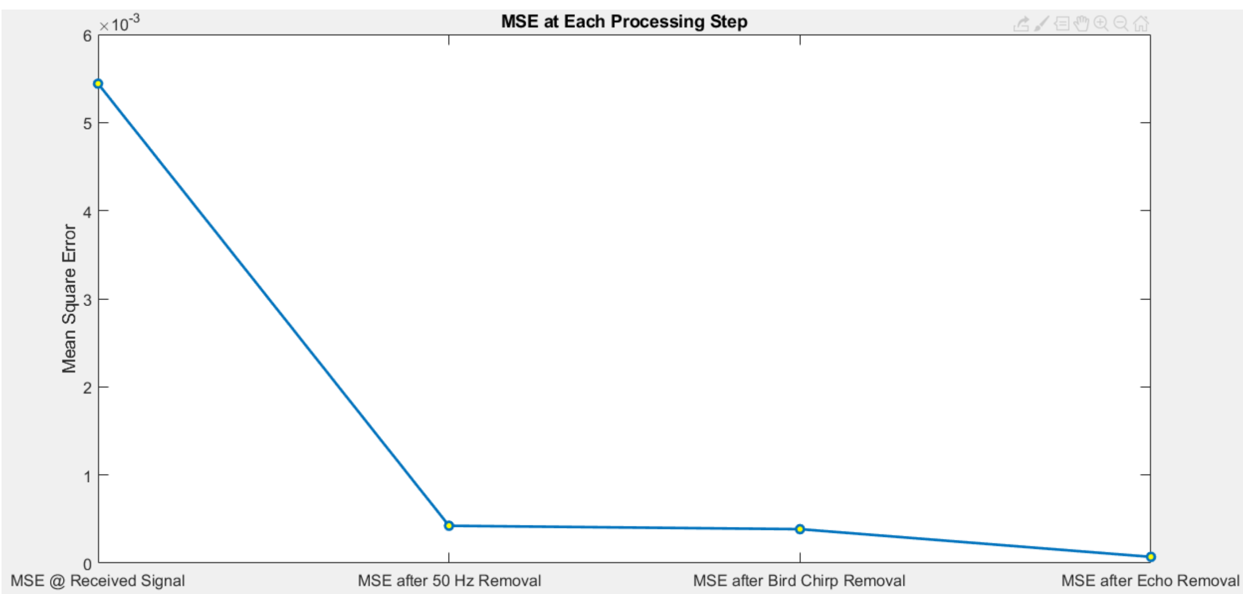
10. An IIR filter is designed which subtracts the echo signal from the signal by making the coefficient at the same number of samples equal to alpha.

$$X(Z)\left(1+\alpha Z^{-\Delta}\right) = X_{corrupt}(Z) \rightarrow X(Z) = \frac{X_{corrupt}(Z)}{(1+\alpha Z^{-\Delta})}$$

11. The designed filter is used to filter out the echo signal. The filtered signal is plotted in time and frequency domain and MSE is calculated.



12. The Mean Square Errors are plotted at each step to show that MSE is decreasing and signal SNR (Signal to Noise Ratio) [5] is improving.



13. The recovered audio signal is saved as a wav file subsequent listening comparison.

Conclusion:

In this project, we created a corrupted signal by adding interferences in the original sound and including the echo of the signal simulating the reflection of audio. The processing digital signal processing algorithms with zero phase IIR (Infinite Impulse Response) filters, autocorrelation and customized digital filters were used to recover the original audio. The Mean Square Error was plotted at each step which showed that signal quality improved and MSE decreased with each signal processing step to recover the original audio.