MMI 503/MMI 603 - Audio Signal Processing 2

Project Report Template

Author: Ashay Dave

Date: 04/18/2023

Project Description and Problem Statement: The project involves designing three different time modulated and filter effects using a modulo counter, circular buffer and *insert something here.* An ESS, sine tone and a song of choice is passed through these effects which are a Modulated Parametric EQ, a Long Delay effect, and a Compressor. The primary objective of the project is to analyze how these effects change the signal, and how they can be shaped to achieve creative results.

Methodology: There are three effects that are designed.

1. Modulated Parametric Biquad Filter: A Q parametric biquad filter is first designed which allows us to adjust the center frequency, Q, and gain. For this example, a non-constant Q filter is chosen. The resulting filtered output is then modulated with a LFO using a modulo counter.

   Using equations for the parameters and co-efficients for the filter, the co-efficients are computed based on the center frequency, Q factor, gain and the sampling rate. Then using a biquad filter, these co-efficients are passed to the difference equation. The resulting output is passed to the modulo function which modulates the filtered output of the parametric biquad filter using a triangular LFO wave.

   The modulo counter returns to zero after a certain amount. Since it is unipolar, it counts from 0 to 1 and back to 0 once it crosses 1 plus the distance it crossed when it exceeded 1. The counter counts up with an increment which is the modulating frequency over the sampling rate. This creates a saw wave, and if we use the formula 2*mod(saw) – 1, we get a triangle wave LFO.

   Purpose: This type of effect is more geared towards creative use on audio signals. A standalone parametric EQ can be used in designing a boost or a cut filter, while a modulating LFO can be used to create effects like a vibrato or a tremolo and to also modulate parameters of different functions or objects.

2. Long Delay Using a Circular Buffer: A long delay uses a buffer that stores the incoming signal, that enter one end of the buffer and exit the other end after certain samples of delay. Then a feedback path is used which creates an echo or delayed effect. A difference equation is utilized – $y(n) = x(n-D) + fb*y(n-D)$ where D is the number of delayed samples and fb is the feedback gain.

   To create this delay effect, a circular buffer is used which wraps the index (pointer) back to the top once it equals the end of the buffer length as the index keeps incrementing. This is a basic explanation of how the circular buffer is used:

   - Calculate the current write index in the circular buffer.
   - Store the current sample in the buffer at the write index.
   - Calculate the read index in the buffer using the delay time and the sampling rate.
   - Get the delayed sample from the buffer at the read index.
   - Add the delayed sample to the current sample with some feedback gain to create the delay effect.
   - Store the processed sample in the output signal.
   - If the write index equals the end of the buffer length, set it back to 0.

   Purpose: A long delay can be utilized to create effects like echo and reverb. Using the similar concepts of circular buffers, we can create pluck sound models.

3. Hard Knee Compressor: A compressor automatically reduces the gain or amplitude of the incoming audio signal once it crosses a certain threshold. How the compressor limits or boosts the gain depends on the attack time, release time and the ratio of compression (which determines how hard the compressor should control the gain).

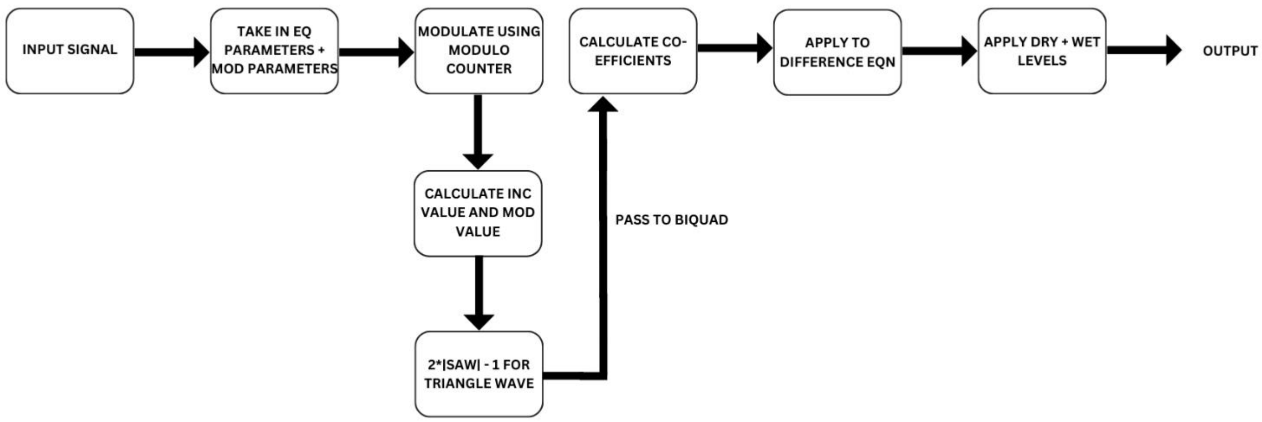
   To implement a compressor, an audio envelope detector is made to calculate the RMS values of the signal based on attack and release times. These values are converted to dB and the appropriate gain is calculated depending on the threshold, knee width and ratio values which determines how much compression should be applied.

   In a hard knee compression, the signal is either above the threshold or below it and the compression is applied accordingly. A soft knee compression allows for a smoother more gradual shift into compression.

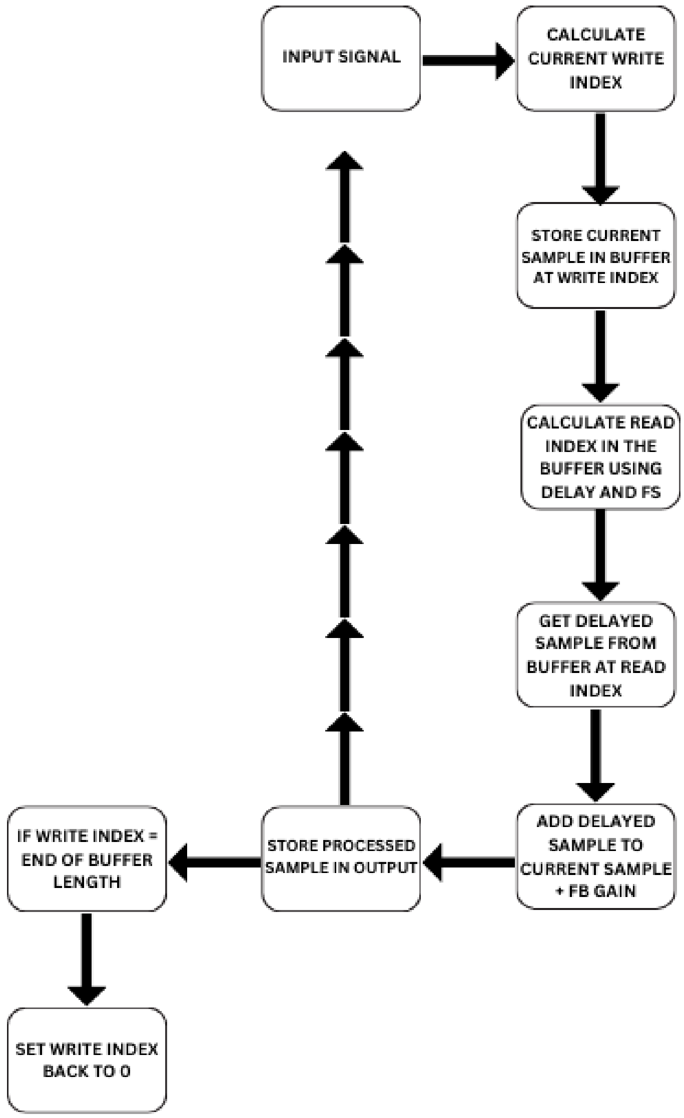
   Purpose: A compressor is primarily utilised to control the gain or amplitude of a signal, and is one the primary components in a mastering and mixing chain. A compressor allows for better control over unnecessary or unannounced peaks, and helps to normalize an audio signal.

<u>Block Diagram</u>: Insert a block diagram

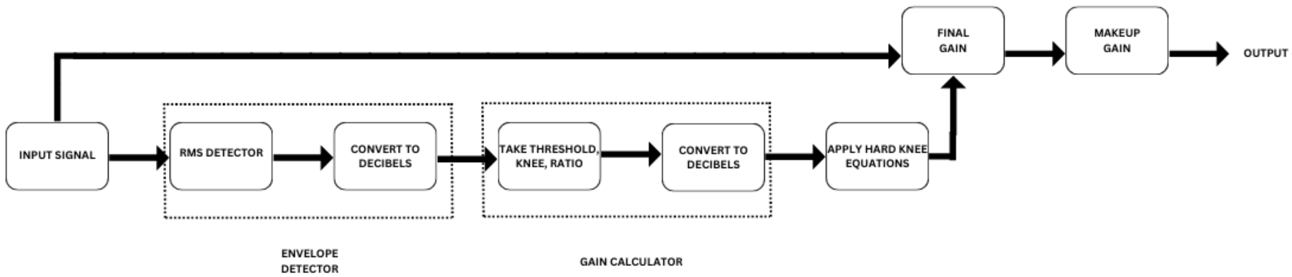
1. Modulated Parametric Biquad Filter:



2. Long Delay Using a Circular Buffer:



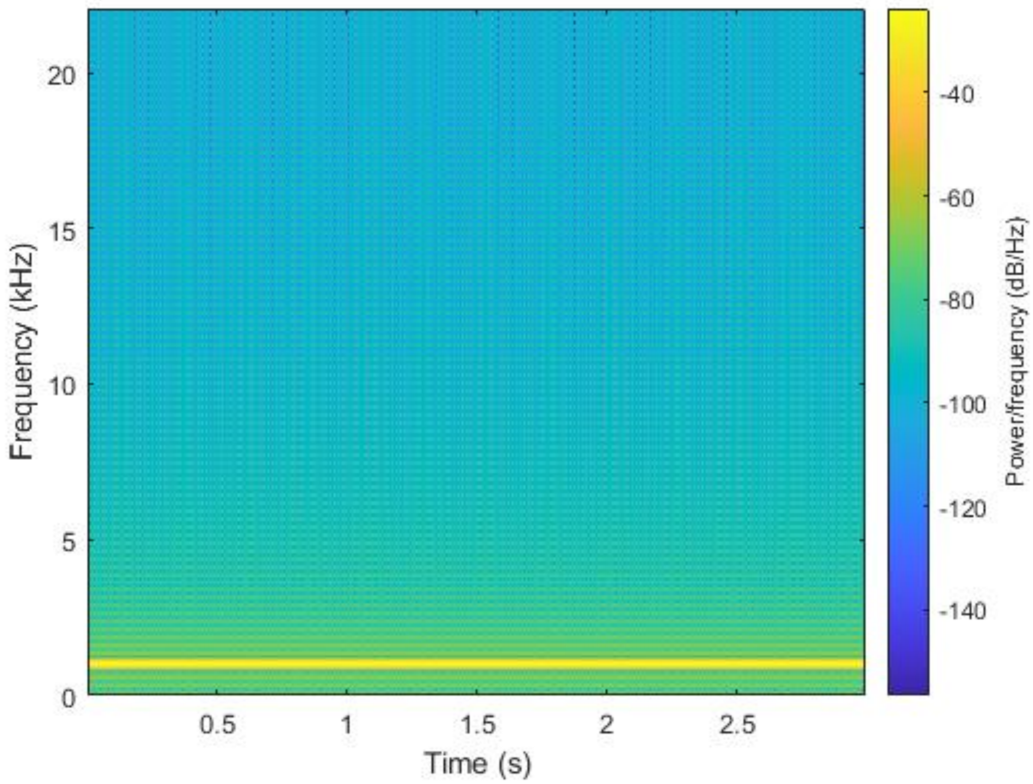
3. Hard Knee Compressor


INPUT SIGNAL
RMS DETECTOR
CONVERT TO DECIBELS
TAKE THRESHOLD, KNEE, RATIO
CONVERT TO DECIBELS
APPLY HARD KNEE EQUATIONS
FINAL GAIN
MAKEUP GAIN
OUTPUT
ENVELOPE DETECTOR
GAIN CALCULATOR

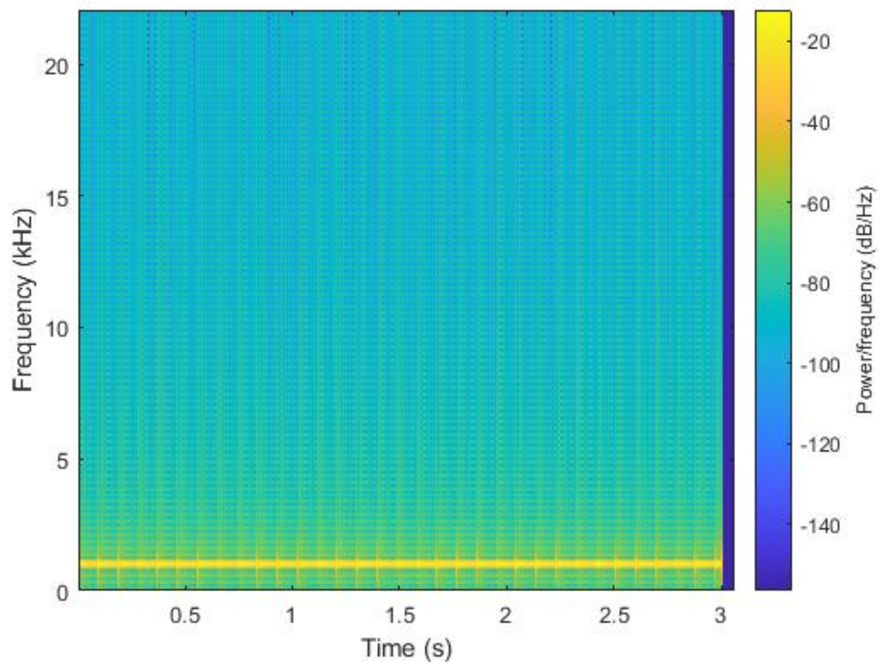
Results: Insert values and/or plots provided by your MATLAB code.

1. **Modulated Parametric Non Constant Q Biquad Filter**

   a. <u>Sine Tone @ 1kHz</u>

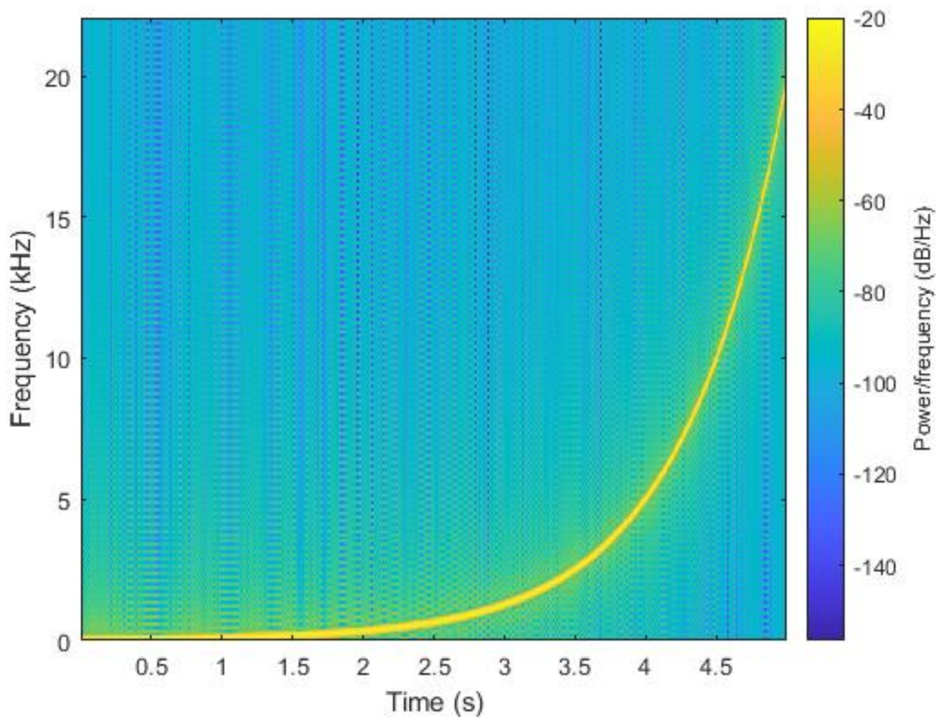


Sine tone spectrogram @ 1kHz

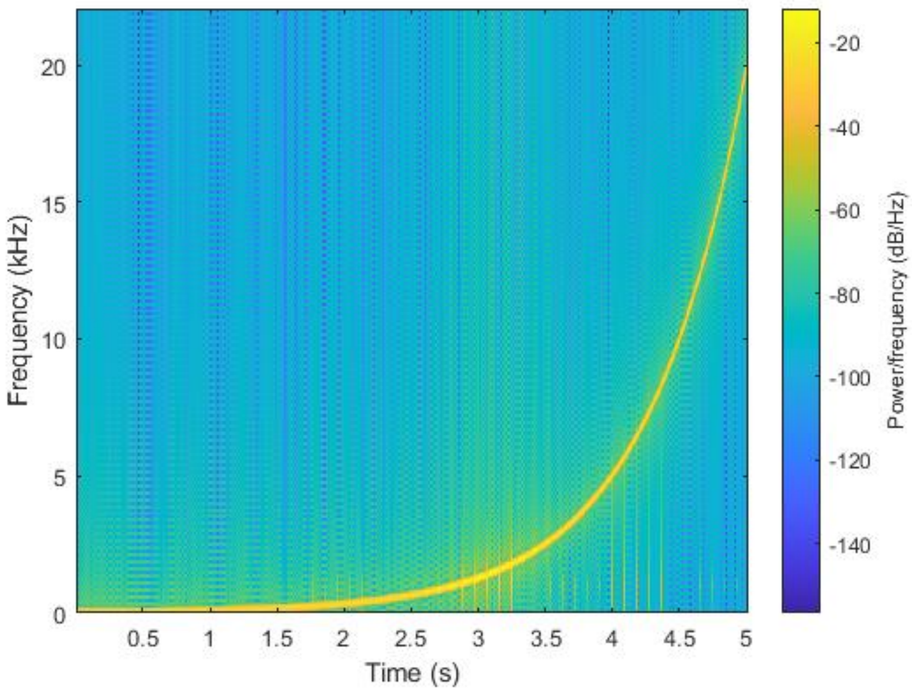


Sine tone spectrogram with Modulated Parametric EQ effect

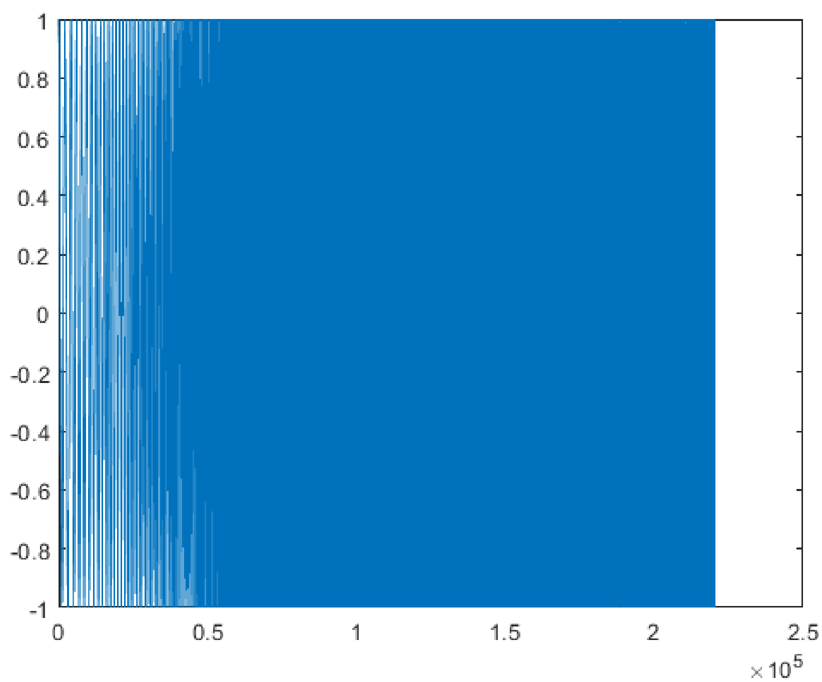
b. <u>ESS</u>



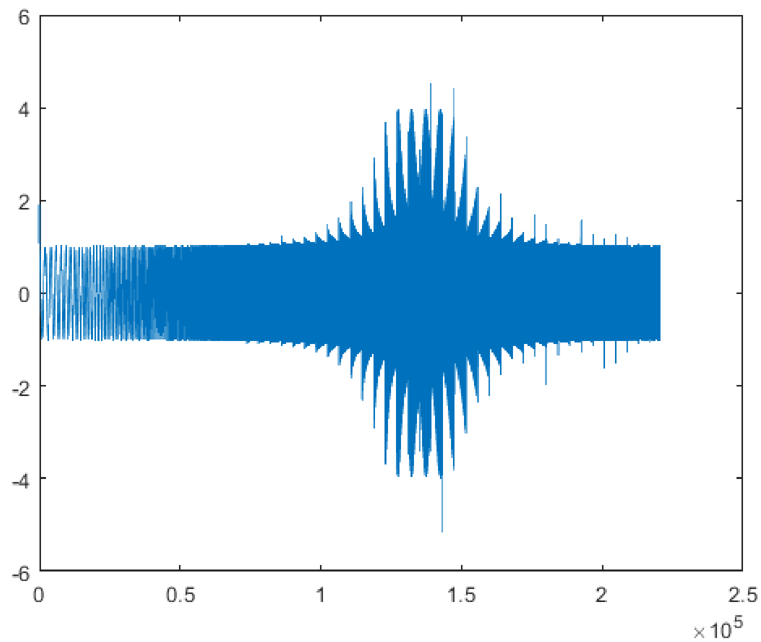
5 Second ESS spectrogram



ESS Spectrogram with Modulated Parametric EQ effect

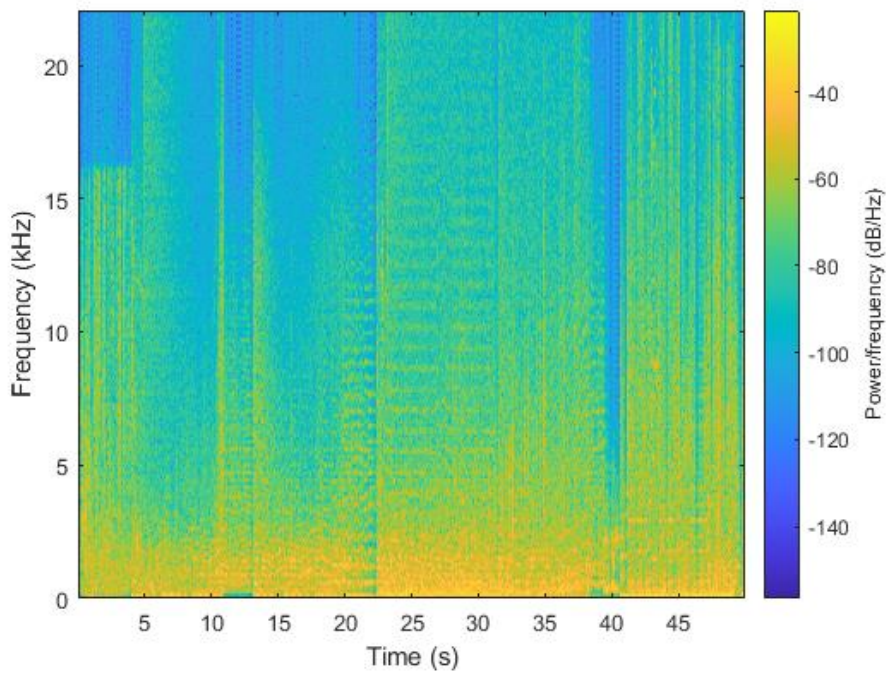


Plot for ESS (5 Second)

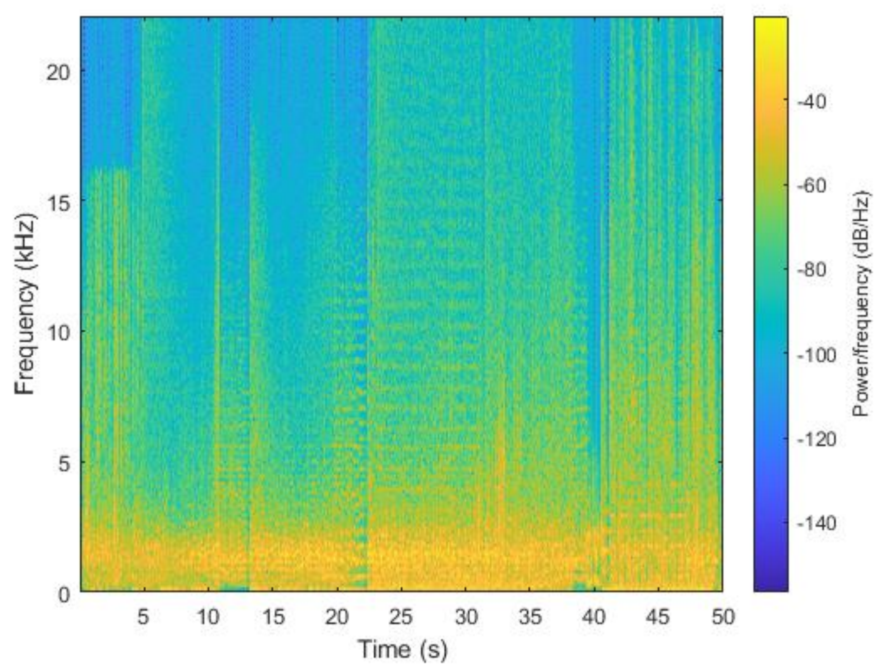


ESS Plot after Modulated Parametric EQ effect

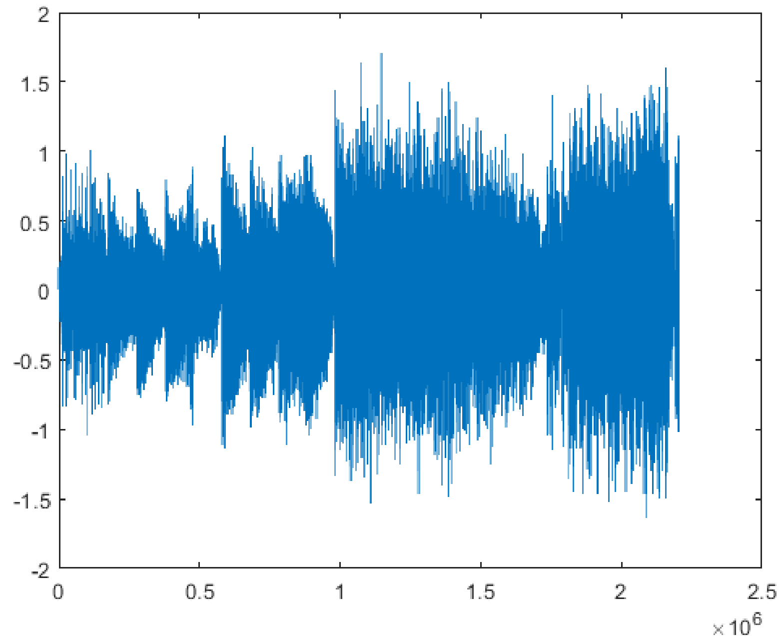
c. Audio file



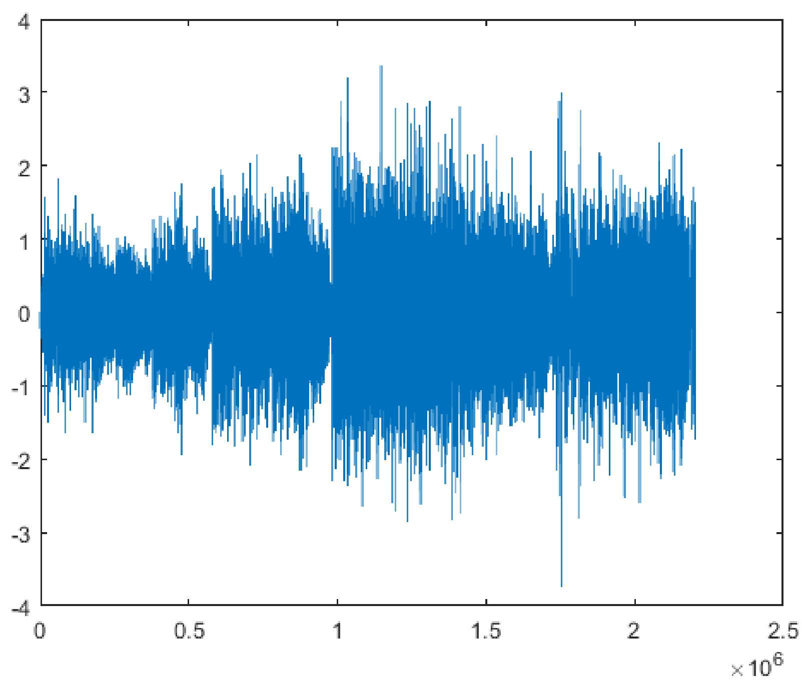
Song spectrogram



Song with Modulated Parametric EQ effect: A boost near the 1kHz region is observed



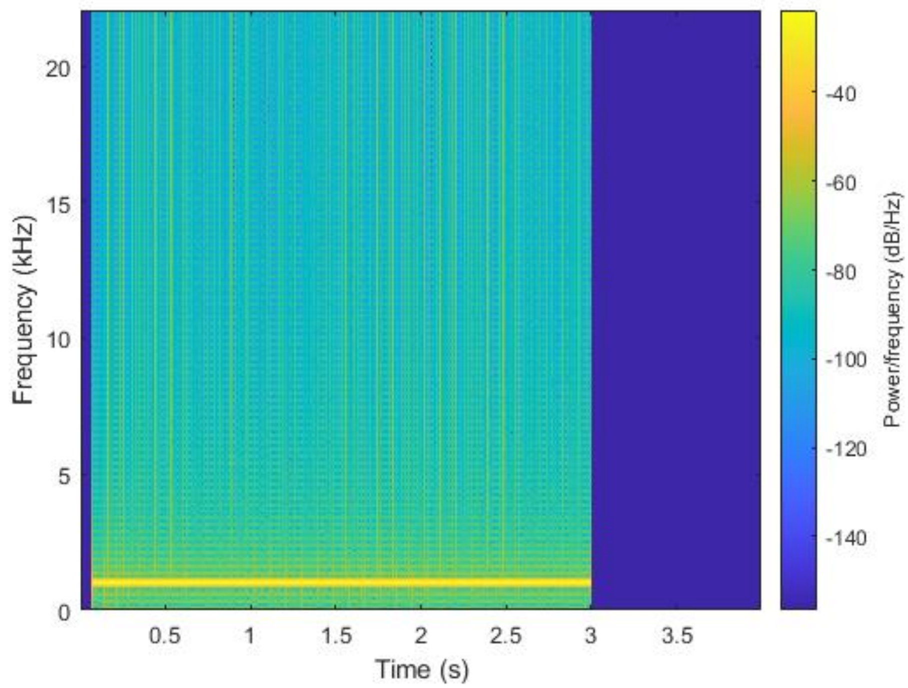
Plot for song file



After applying Modulated Parametric EQ effect

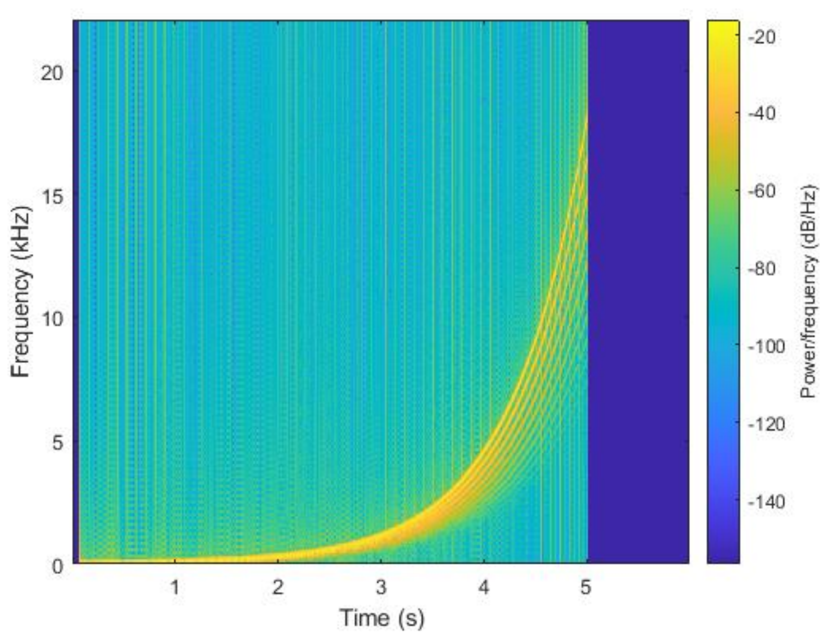
2. Long Delay Using a Circular Buffer:

a. <u>Sine Tone @ 1kHz</u>

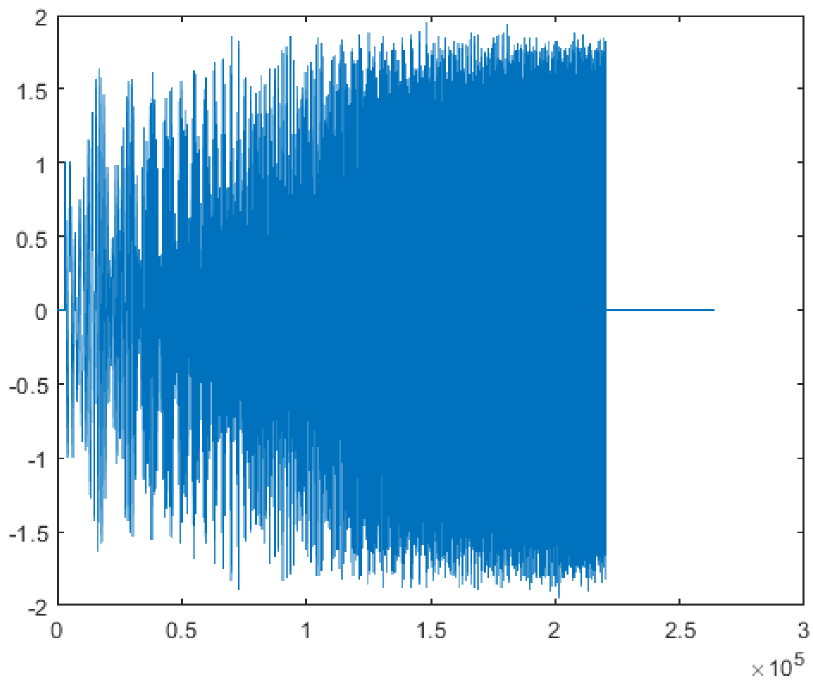


Delayed Sine tone

b. <u>ESS</u>

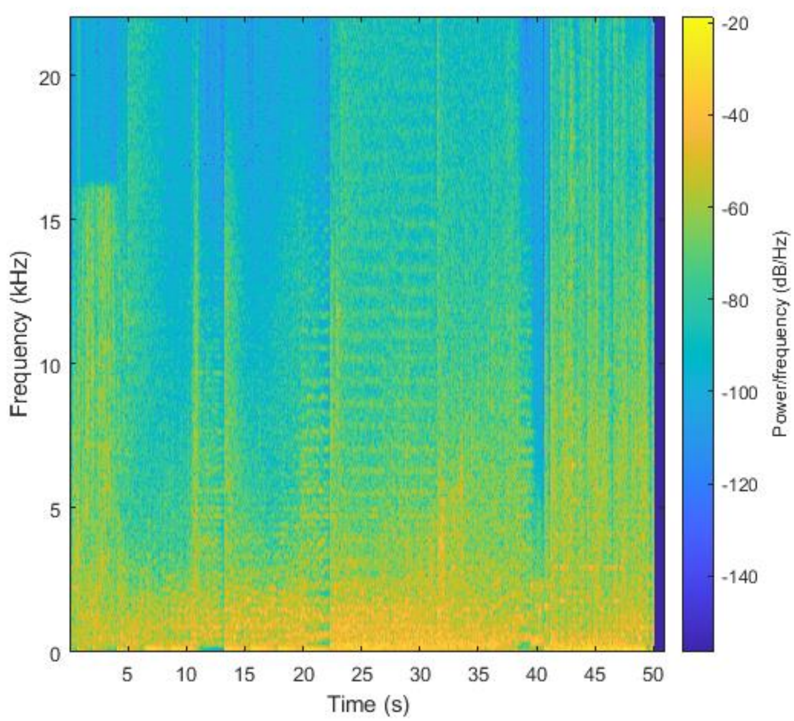


Delayed ESS

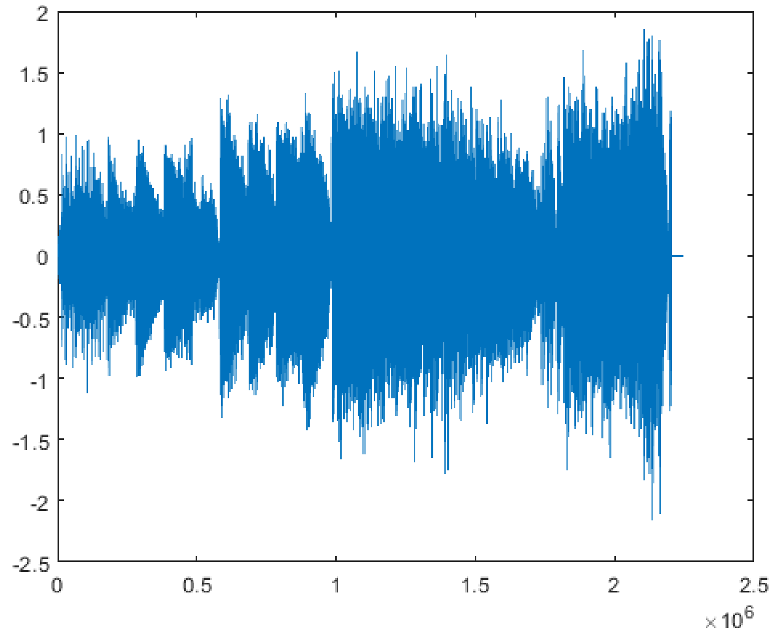


Plot for Delayed ESS

c. <u>Audio File</u>



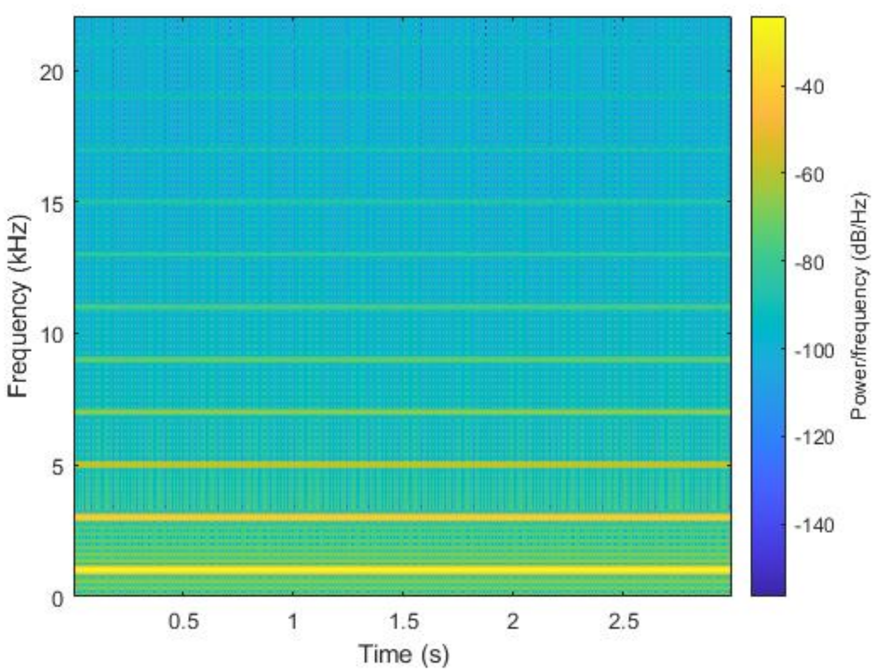
Delay effect on the song



Plot of the delay effect on the song. Some transient peaks can be seen when the transients inside the song are delayed.

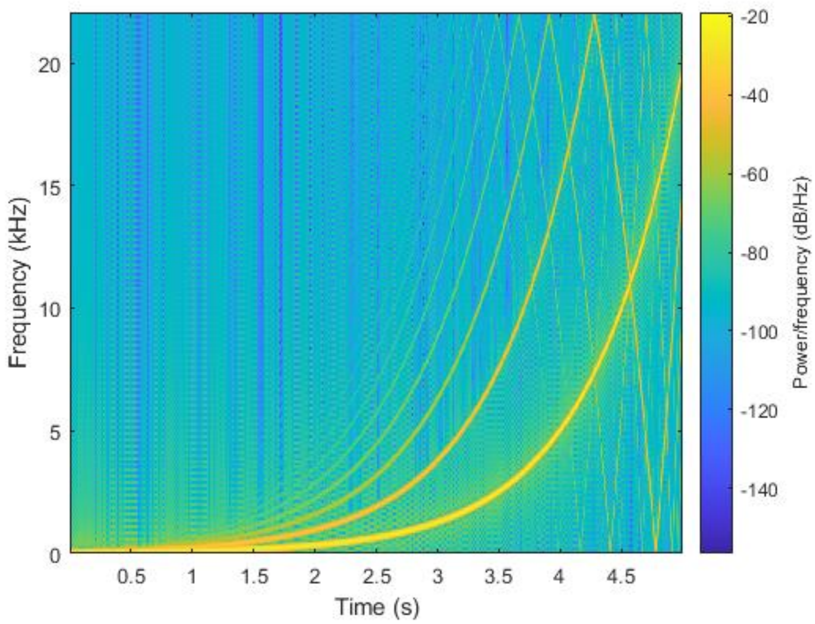
3. Compressor (Hard Knee)

a. <u>Sine Tone</u>

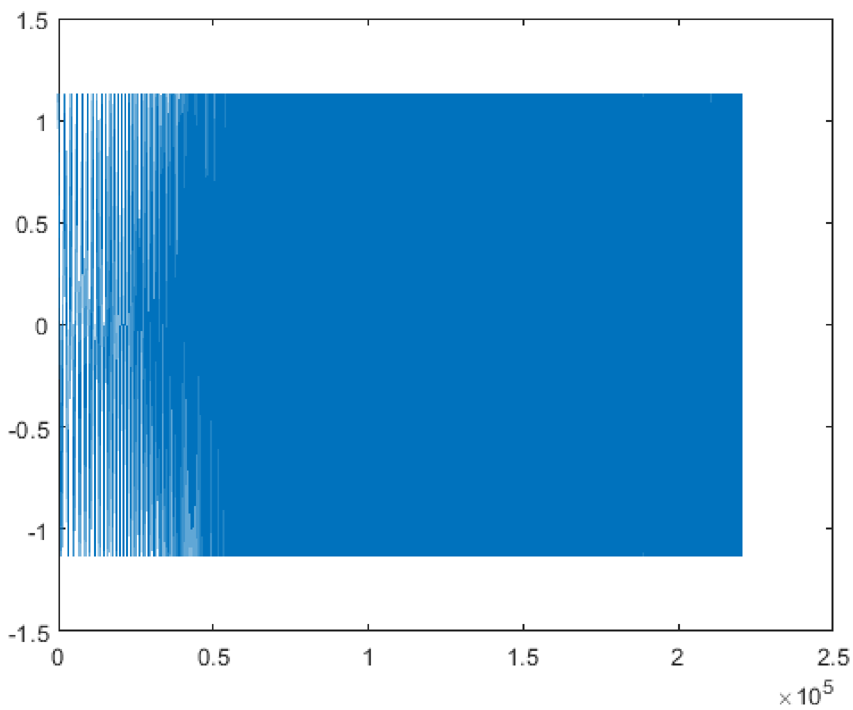


Compressed Sine Tone: Harmonics are observed.

b. ESS

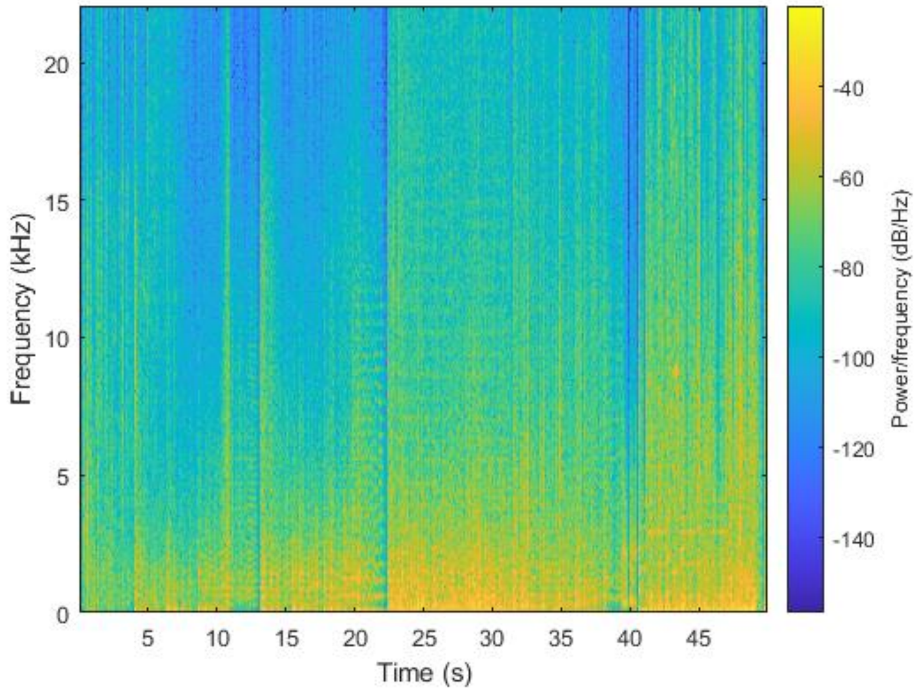


ESS Compressed

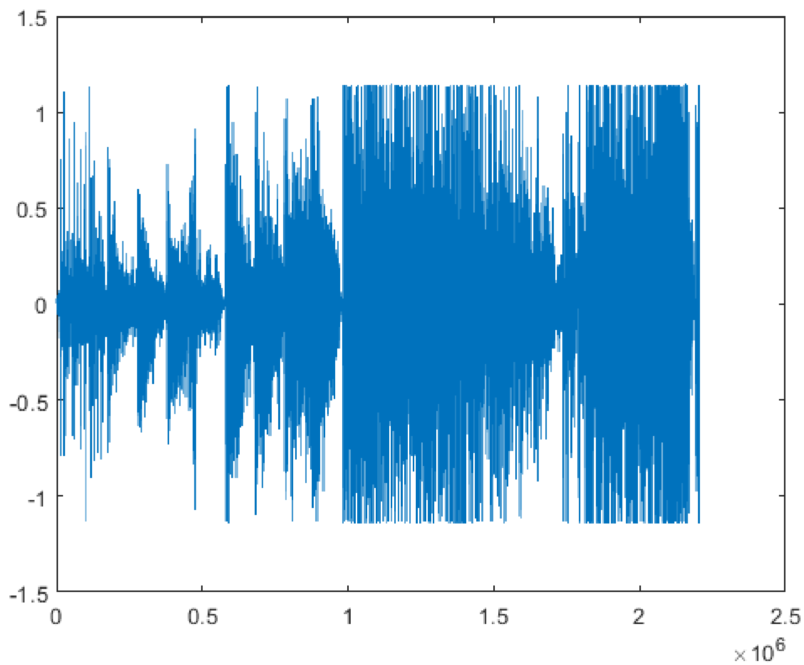


Plot for Compressed ESS

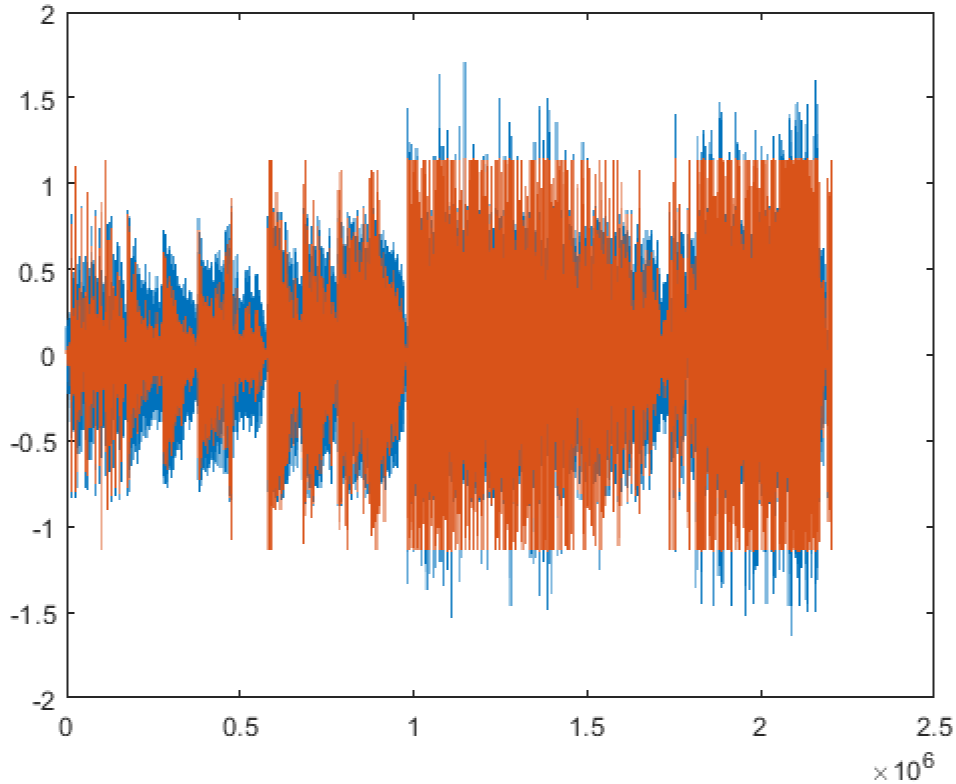
c. <u>Audio File</u>



Compressed Audio File



Plot of Compressed Audio File



Plot of Compressed (red) vs Uncompressed (blue)

Discussion: The functions created in this project are useful primarily in a creative audio engineering environment such as mixing, mastering or post production. The effects described and implemented are also important in understanding the variety of ways a filter can be designed to achieve creative results in the context of music production.

Other than creative audio engineering, these effects are also implemented in algorithms that require noise shaping and/or removal, gain controlling (compression), spatializing sound, and in cases where the signal needs to be filtered (EQed) to give a concise boost or cut in certain frequency bands.

Compression/expansion algorithms are also common on streaming services, where the music is compressed to follow a standard loudness limit. The functions implemented in this project can be used as a good starting point to understand how compression works in a real-world scenario. Similarly for applications involving Spatial or 3D audio, delay effects are much more common albeit highly tweaked to achieve “spatialization” of the sound signal.