MMI 503/MMI 603 - Audio Signal Processing 2

Project 1 Report

Author: Ashay Dave

Date: 02/07/2023

Project Description and Problem Statement:

In order to represent the frequency content of the signal which provides essential information about the characteristics of the signal such as the presence of certain frequencies and their locations, their magnitude, energy distribution, pitch and harmonic features - spectral analysis (in the frequency domain) needs to performed to extract said characteristics of a digital audio signal. These characteristics are extremely important while designing filters and audio signal processing methodologies.

The project is aimed at understanding how spectral analyses is done on a digital audio signal using the fundamental concepts for signal processing with built-in MATLAB functions, and creating custom functions to generate, analyze and study audio signals particularly in the frequency domain.

Methodology: Describe the functions you are developing.

<u>Function 1: rms_loudness</u>

This function takes an input signal “x” and returns the root-mean squared loudness value of that input signal in decibels.

The built-in MATLAB function for root-mean square is used which returns the RMS value of that input, which is then converted to a decibel value by multiplying the “log10” function by 20.

<u>Function 2: spectral_analyzer</u>

This function takes two inputs – input signal “x” and a sampling rate “fs”, and returns the magnitude spectrum in decibels and frequency bin values. A Discrete Fourier transform is applied on the input signal to extract the frequency components.

First, the number of samples that will undergo the Fourier transform are assigned to be the length of the input signal “x”, which is basically the full signal.

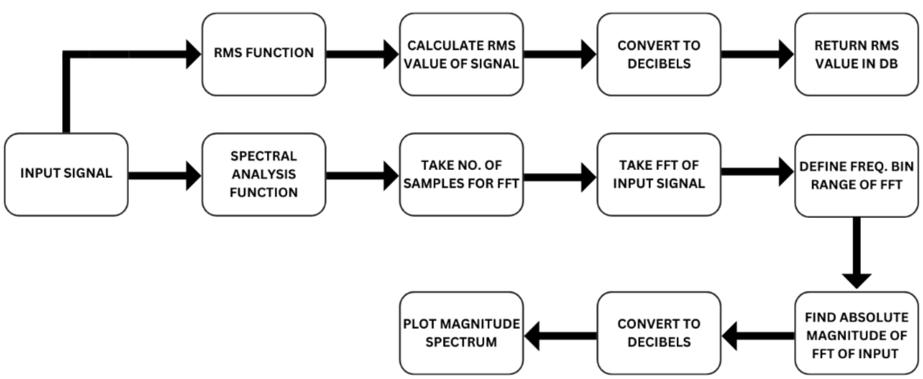
A Discrete Fourier transform is done on the input signal using the built-in MATLAB function “fft()” on the input signal, with the length of the transform being the sampling rate “fs”. DFT is used to analyze the frequency components of the signal by representing the signal in the frequency domain.

The bin width is calculated by dividing the sample rate by the length of the FFT (which is the length of the signal initially). The frequency range is then calculated by going from 0 to the length of the DFT minus one sample (the span of the bin index because the length of the DFT is equal to the number of samples), and multiplying it with the bin width.

The magnitude spectrum is converted to the decibel scale by taking the absolute value of the input divided by the number of samples then using the log10 function on that value.

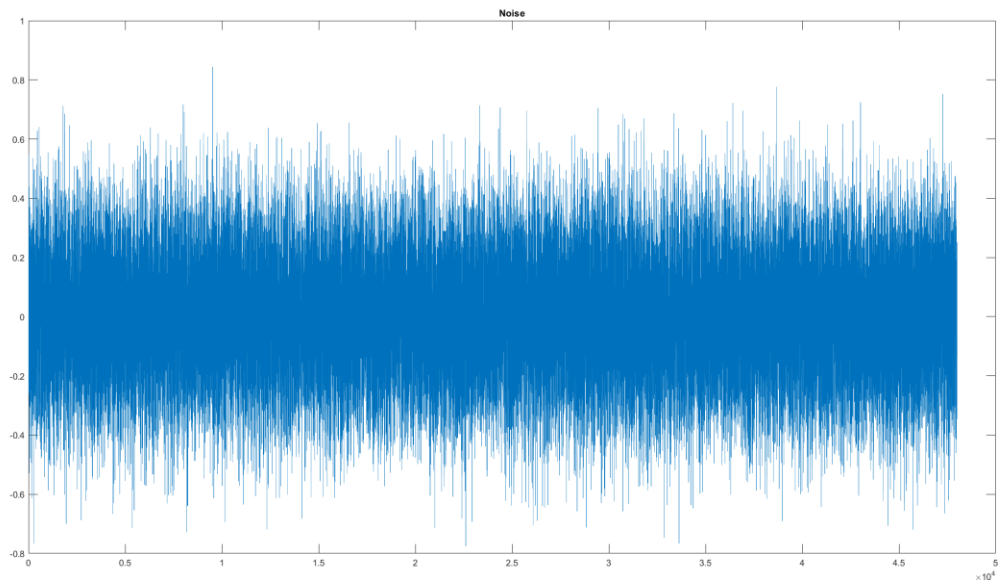
This particular function takes the bin size to be 1 (divides the frequencies into bins of 1Hz), so a frequency of 1Hz would be in the bin range of 0 to 1.

Block Diagram:



Results:

1) Noise Waveform & Spectral Waveform using spectral_analyzer:



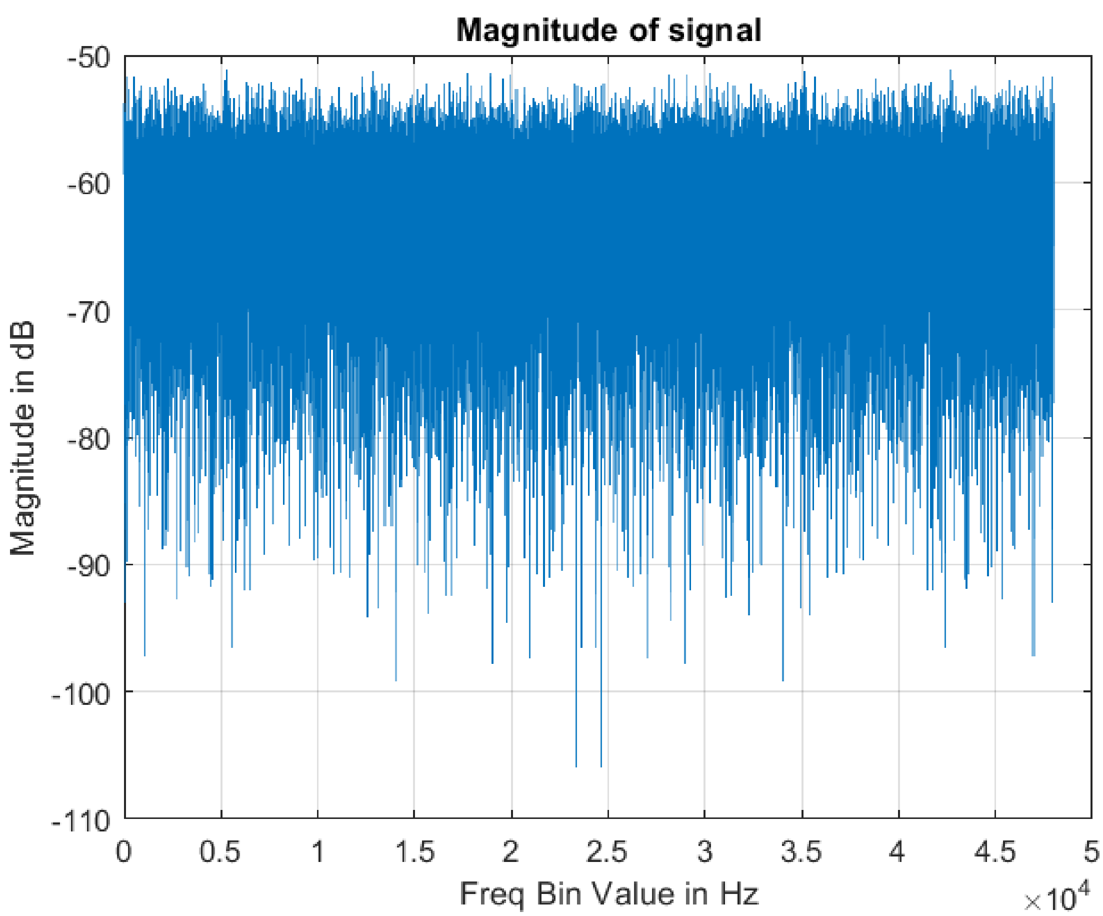


Fig. 1(a) and 1(b)

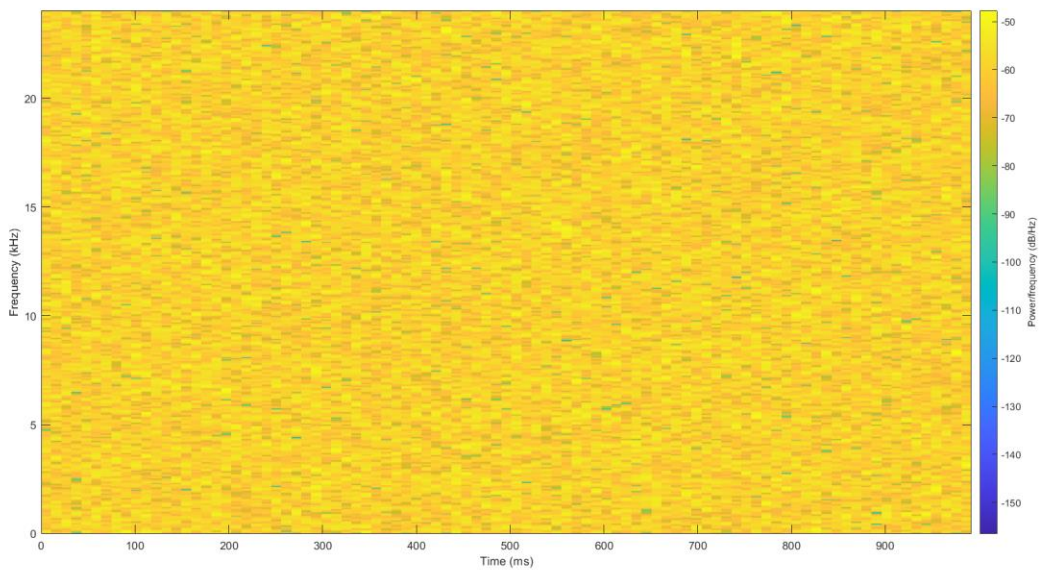


Fig. 1(c) White Noise Spectrogram

RMS of this noise signal comes out to be approximately -14 dB or -13.9215 dB to precise.

2) Sine Tone @ 1KHz

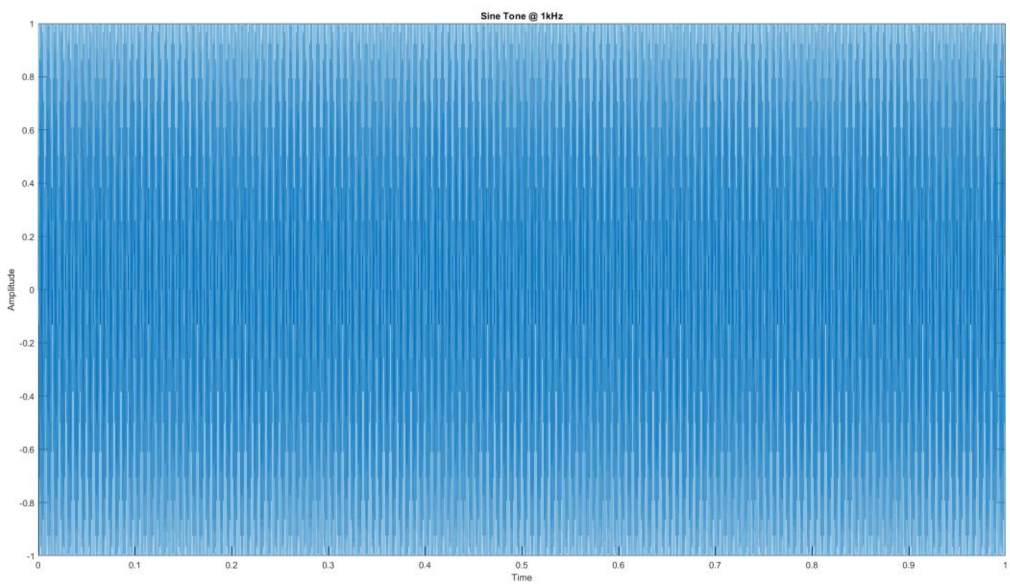


Fig. 2(a)

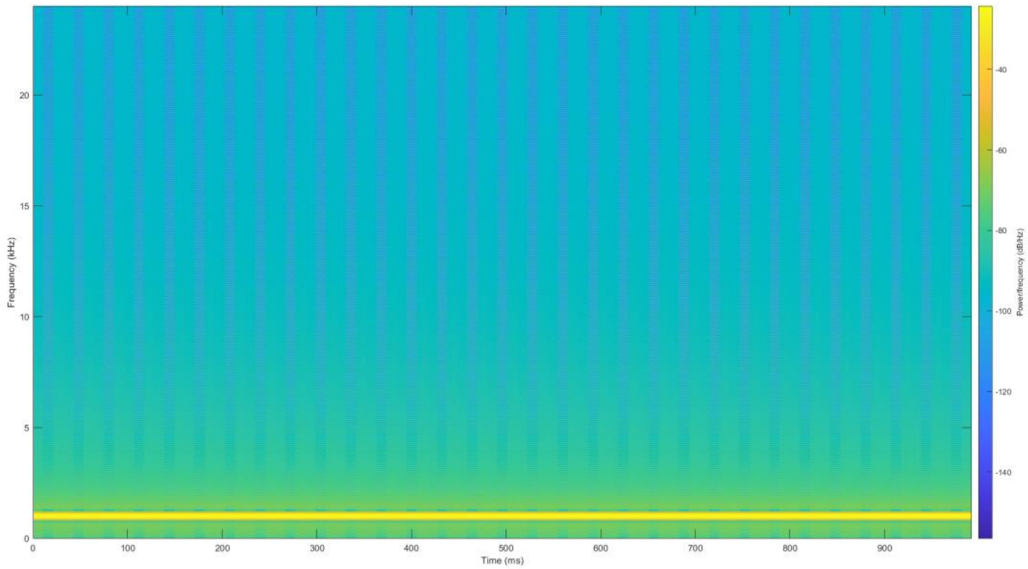


Fig. 2(b) Sine tone at 1 KHz Spectrogram

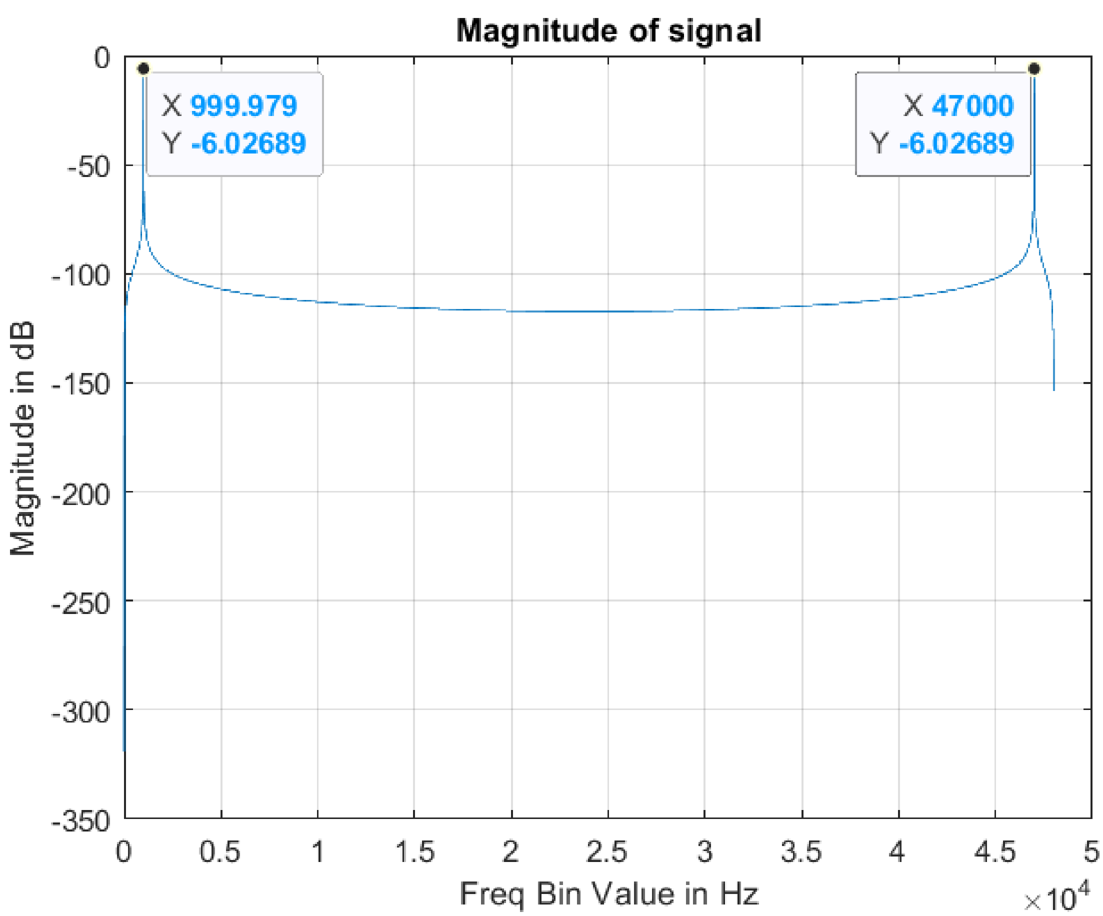


Fig. 2(c) Magnitude Response of 1 kHz Sine Tone

Using the spectral analysis function, the 1 KHz can be seen in the frequency bin value of 999.979 which is approximately 1000. Its counterpart can be seen at 47,000.

RMS of a 1KHz sine tone comes out to be -3.01 dB.

3) Exponential Sine Sweep (3 Second):

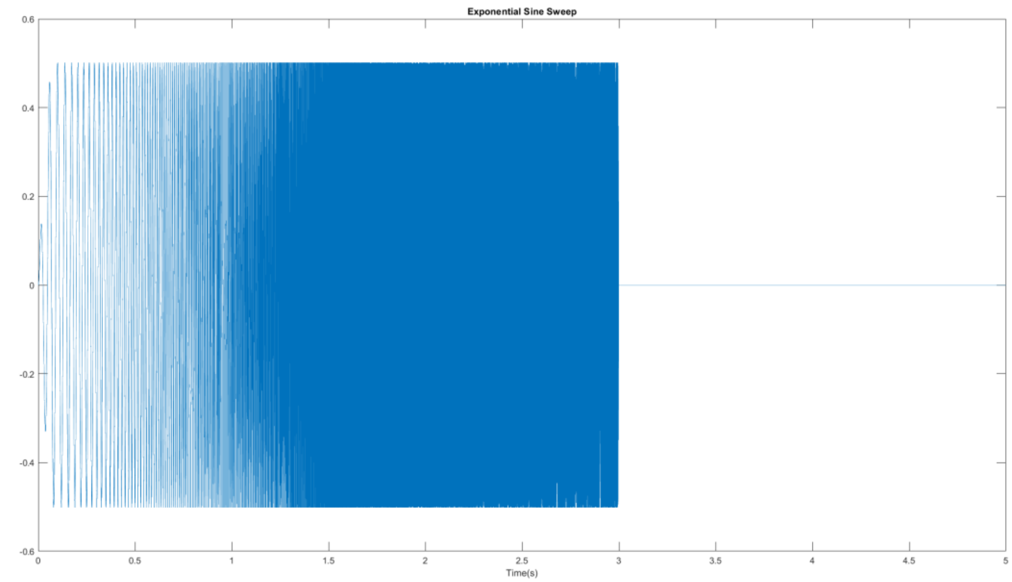


Fig. 3(a)

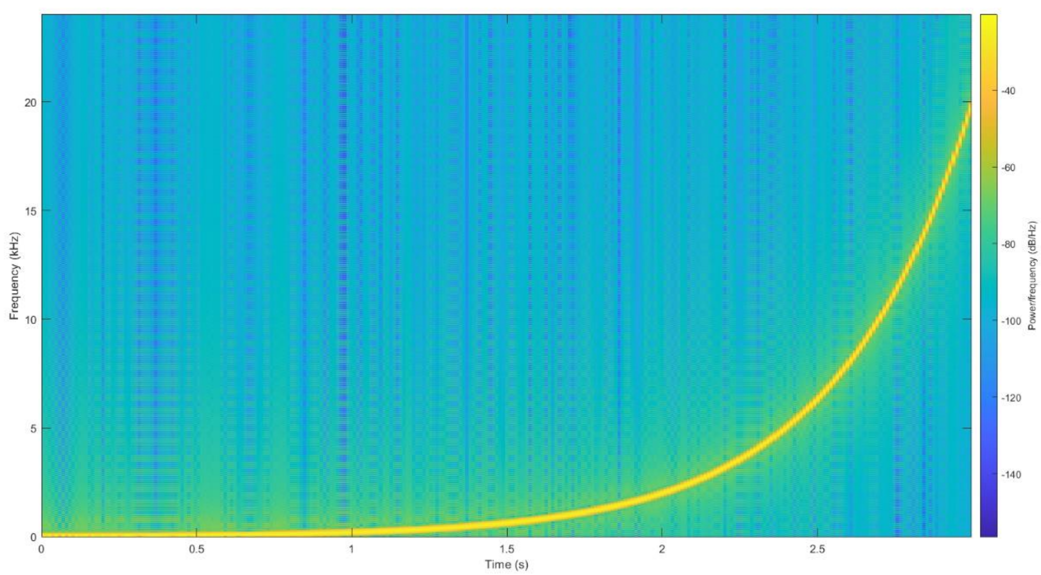


Fig. 3(b) Exponential Sine Sweep Spectrogram

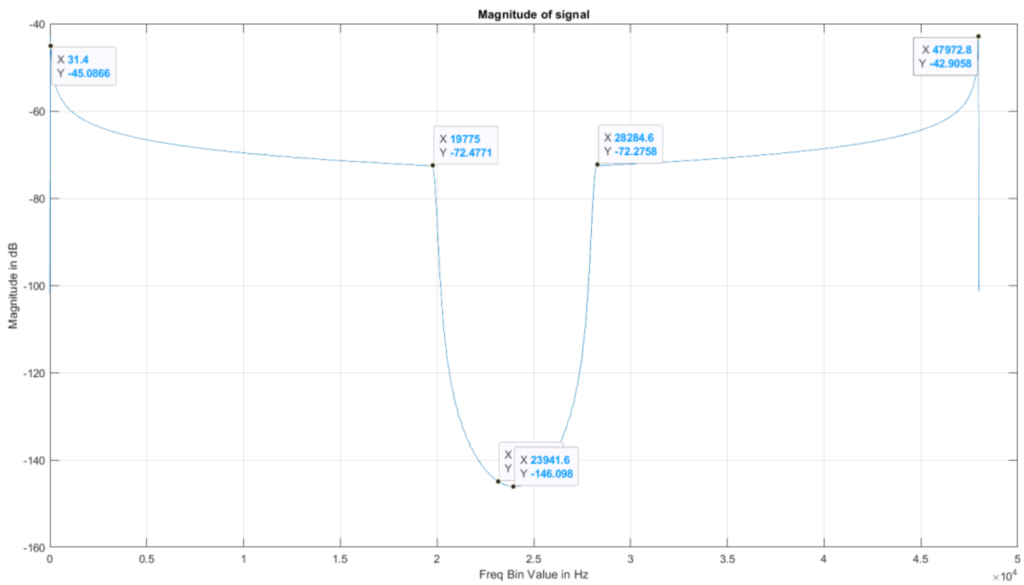


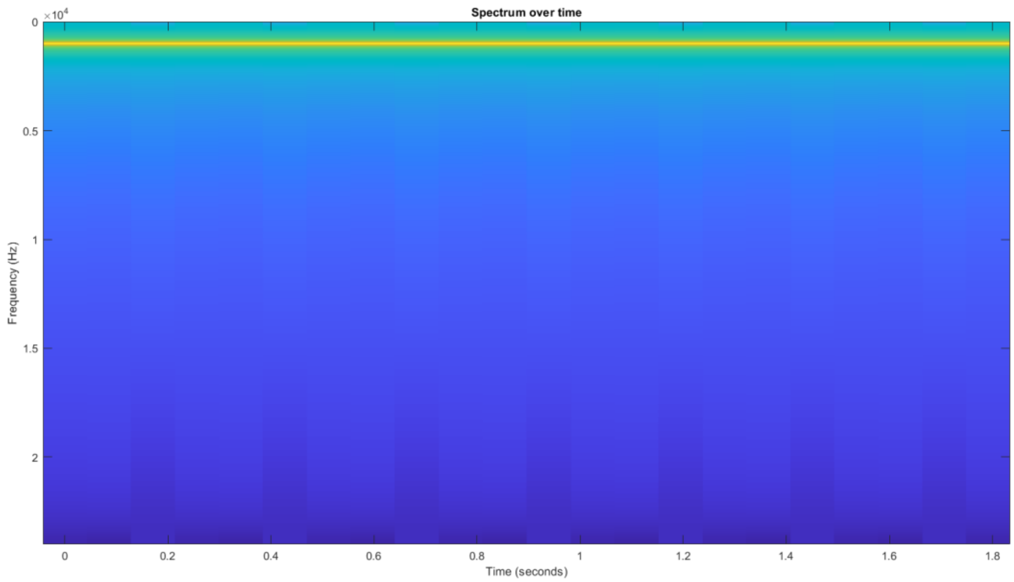
Fig. 3(c) Magnitude Response of ESS

Using spectral analysis, the behavior of the exponential sine sweep that goes from 20 Hz to 20 KHz is shown on this plot (figure 3(c)). It is observed how the sweep goes back up near the Nyquist frequency to form the conjugate of the original frequencies.

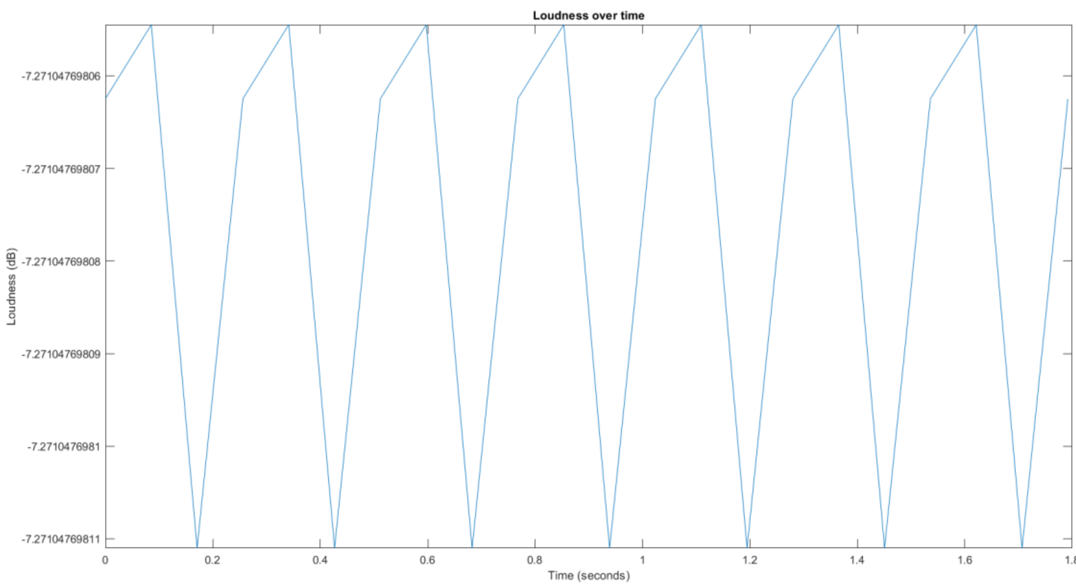
RMS of a 3 second ESS is found to be -11.3 dB.

<u>4) Analyzing the signals using buffer size of 4096 and an overlap of 2048:</u>

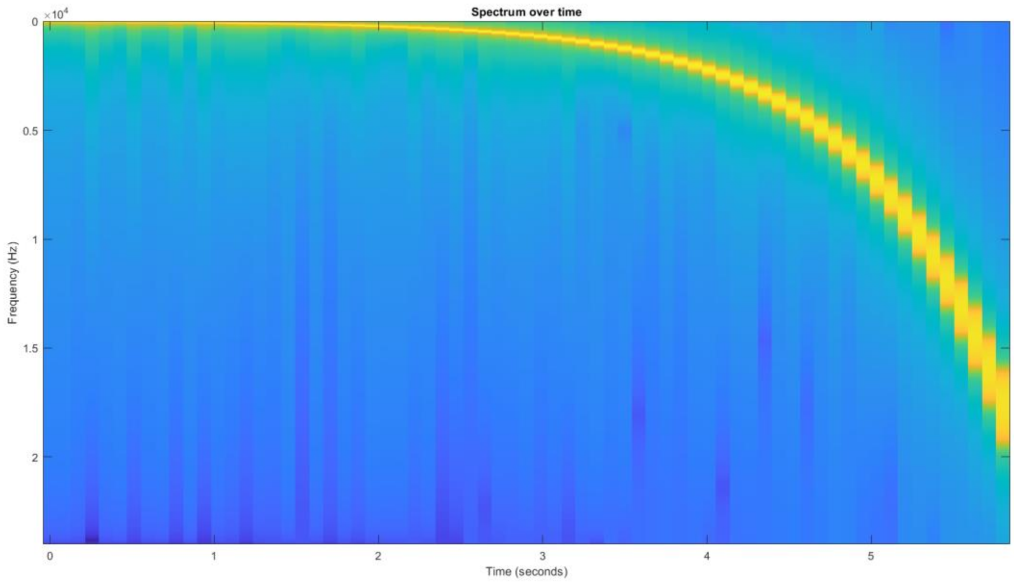
a) Sine Tone @ 1 kHz Spectrum Over Time:



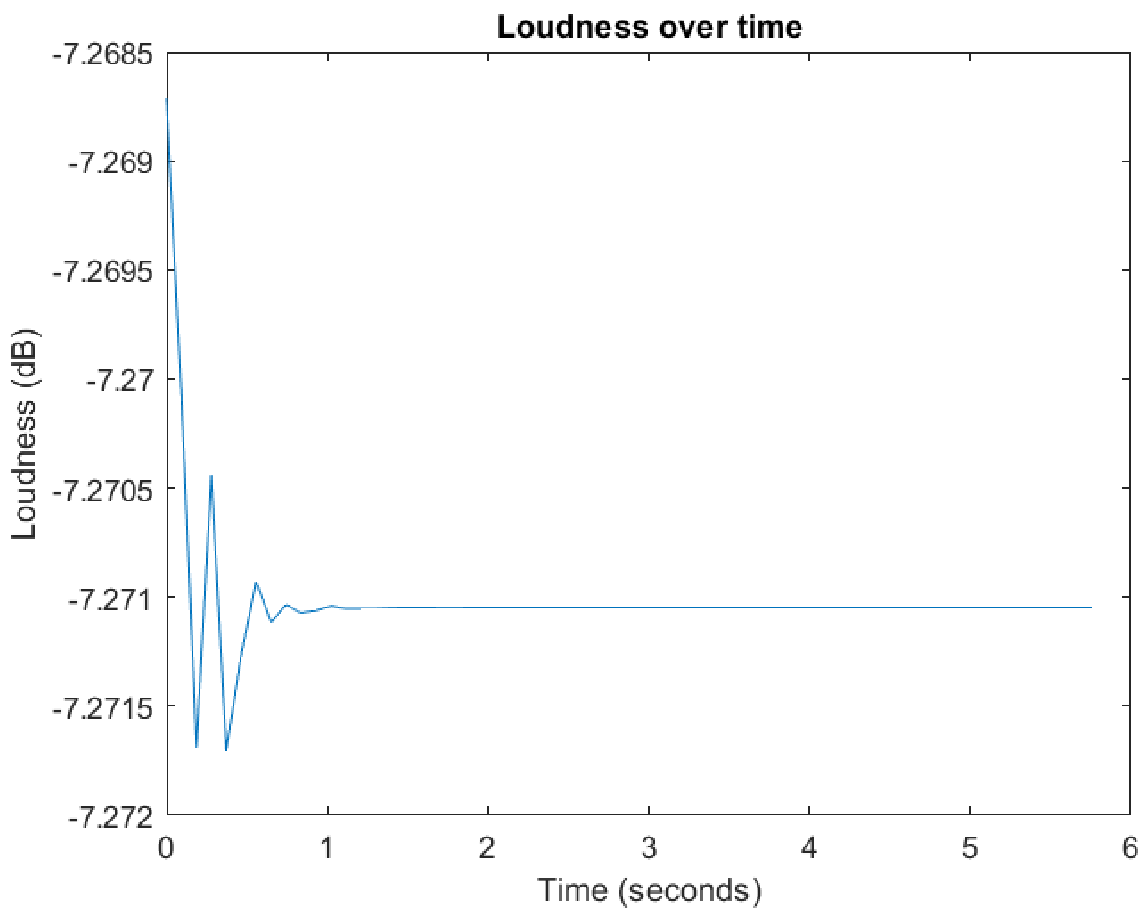
Sine Tone @ 1 kHz Loudness Over Time:



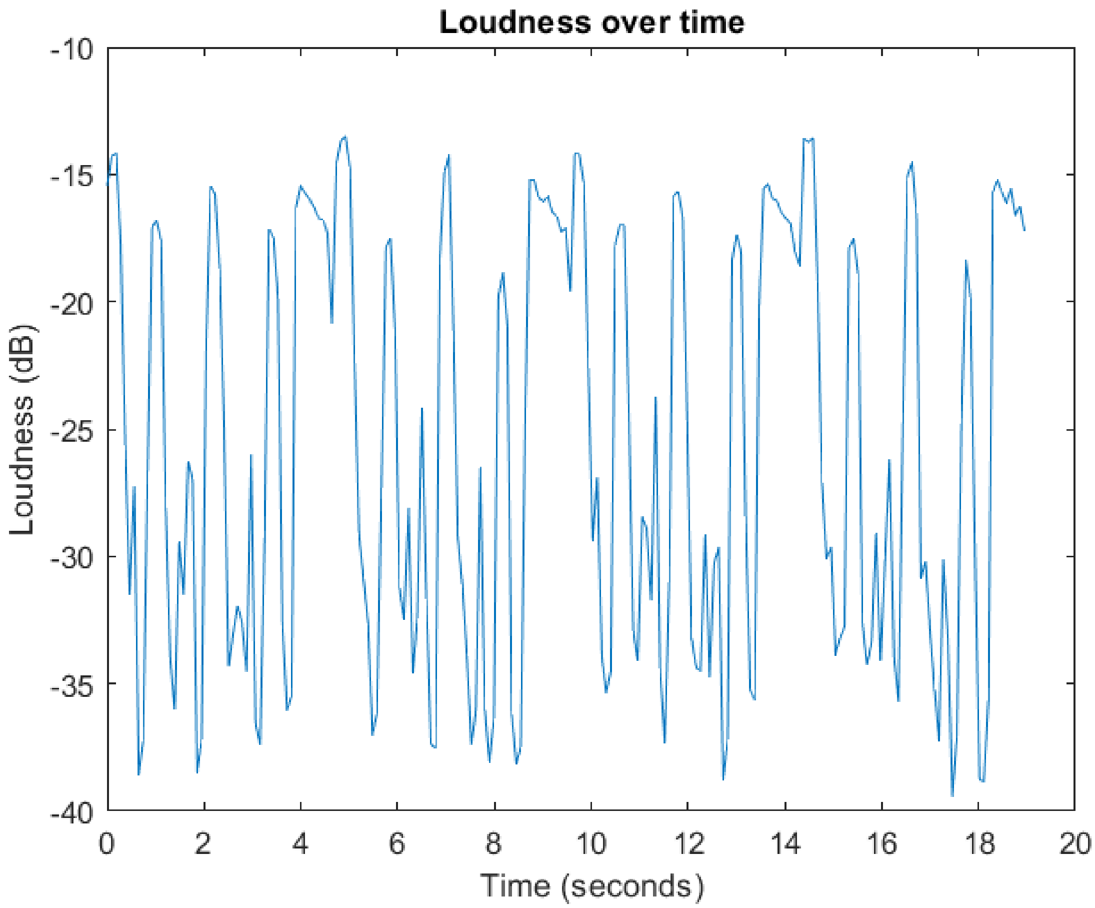
b) ESS Spectrum over time:

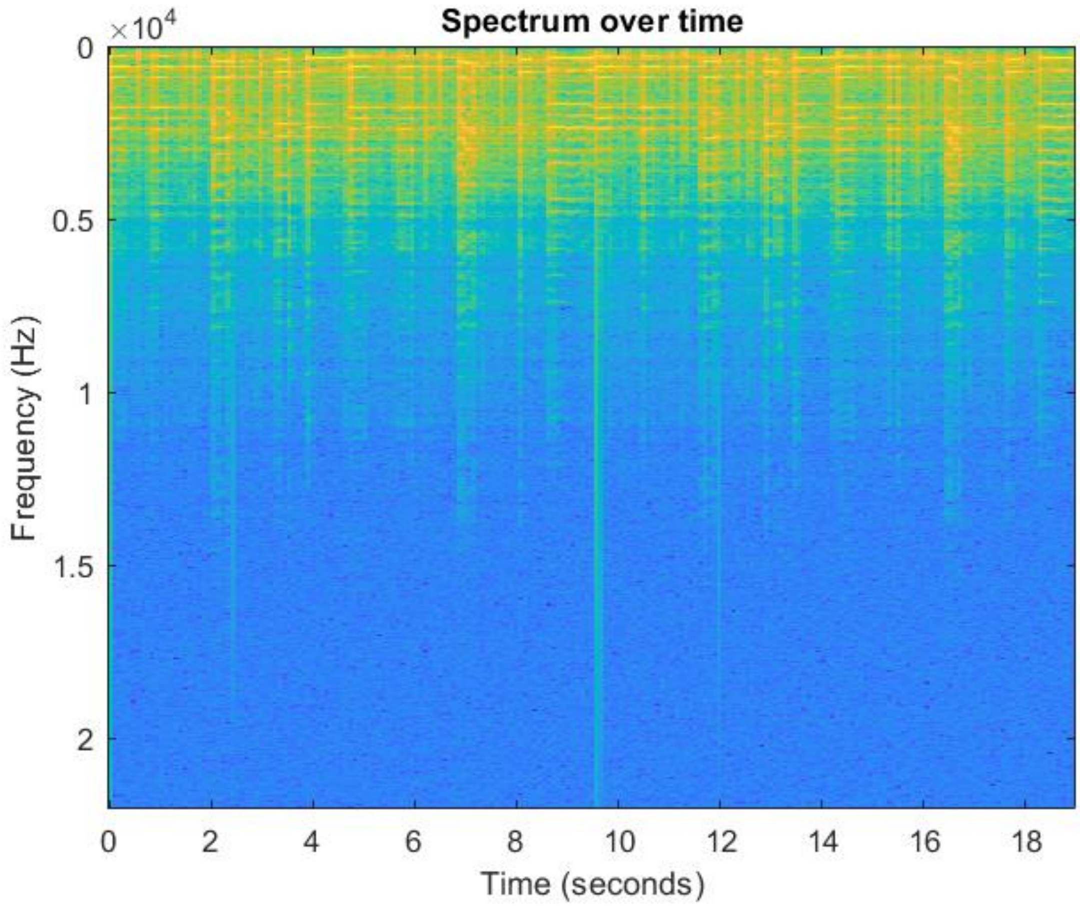


ESS Loudness over time:



c) .WAV File of choice (Guitar Clip 9 seconds long):





Discussion:

These type of analyses systems and functions present a wide range of characteristics of a digital audio signal to be manipulated and studied, and being in the frequency domain, allows for better control over parameters that directly affect how a digital audio system may be designed.

Spectral processing techniques provide valuable information and insights such as the frequency characteristics of a signal such as where particular frequencies are present, which is utilized in applications such as filtering and equalization.

For example, a spectrum analyzer plugin used in a DAW provides the user with highly accurate representations of the frequencies present in a particular audio track, which would be pivotal in sound designing. Spectral processing is also a fundamental technique for separating frequency contents of different sounds – also referred to as source separation.

Frequency domain representation of a digital signal using spectral processing also exhibits the harmonic structure and how energy (loudness) is distributed across the spectrum. This type of information is highly influential in applications that involve pitch detection, compression, noise reduction, etc. For example, identifying the frequency/spectral content of noise which can be removed by using filters.

Spectral processing increases computing speeds, and also allows for designing a magnitude response for a filter, giving specific control over filter design. Higher level analysis can also be done to yield a spectrogram (visualization of a spectrum over time).