Ahsanullah University of Science and Technology

Project

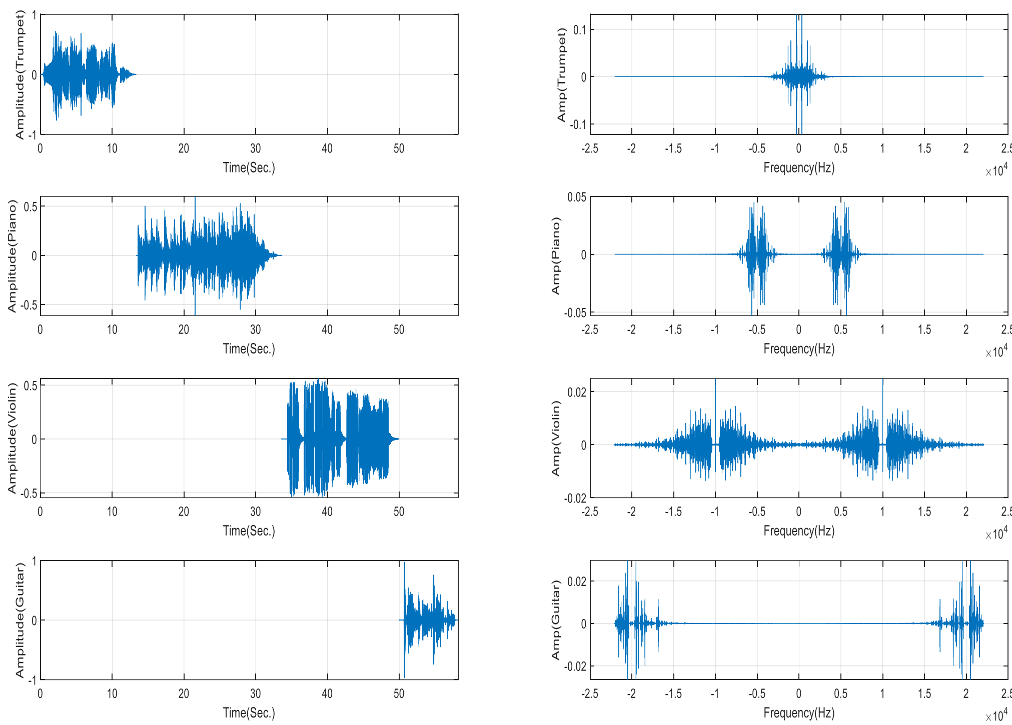
Course No: EEE 3218
Course Name: Digital Signal Processing Lab.

Date of Submission: 23/02/2023

Submitted By:

Name: Md. Nujratul Haque
ID: 190205131
Year: 3, Semester: 2
Department of Electrical and Electronics Engineering.

1.The Spectrum of each of the instrument’s sound from ‘NonOverlapping.wav’:



The spectrum of each of the instruments are given above. We have split the given audio signal (NonOverlapping.wav) into their corresponding time frame. And find out their frequency spectrum using FFT. Now, we can easily find the frequency range from it.

Frequency ranges:

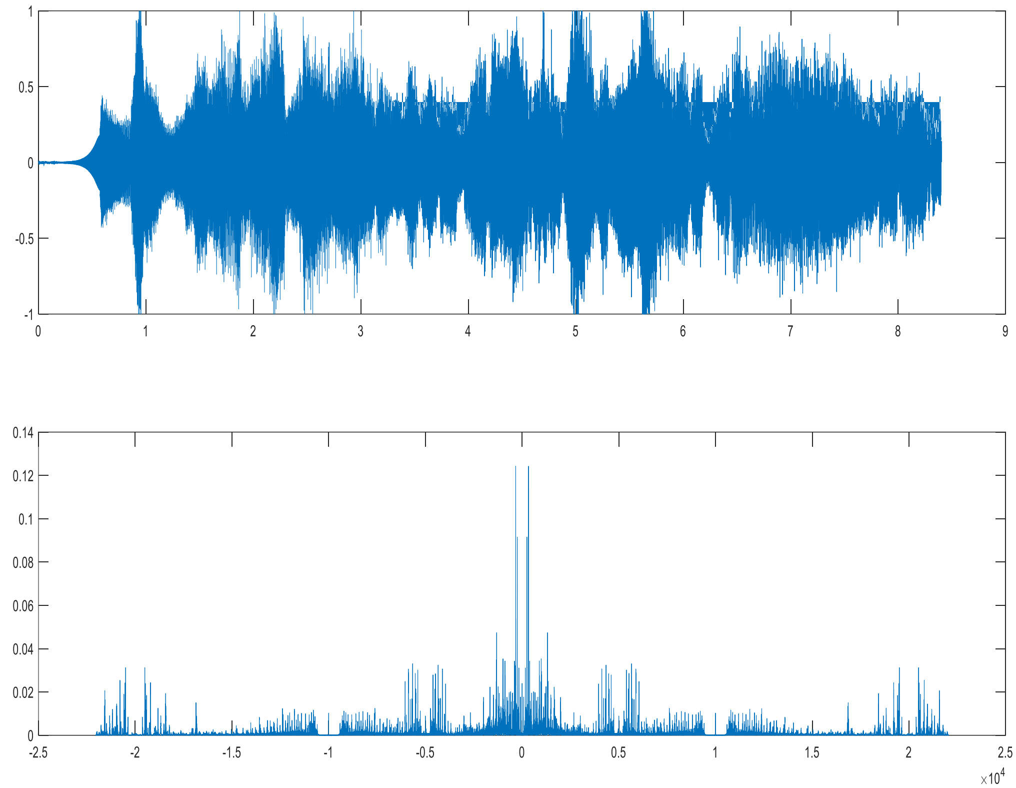
Trumpet: 0 - 2050

Piano: 3400 – 6700 Hz

Violin: 5851 – 15202 Hz

Guitar: 18000- 22000 Hz

2.The Spectrum of each of the combined sound signal from ‘final.wav’:

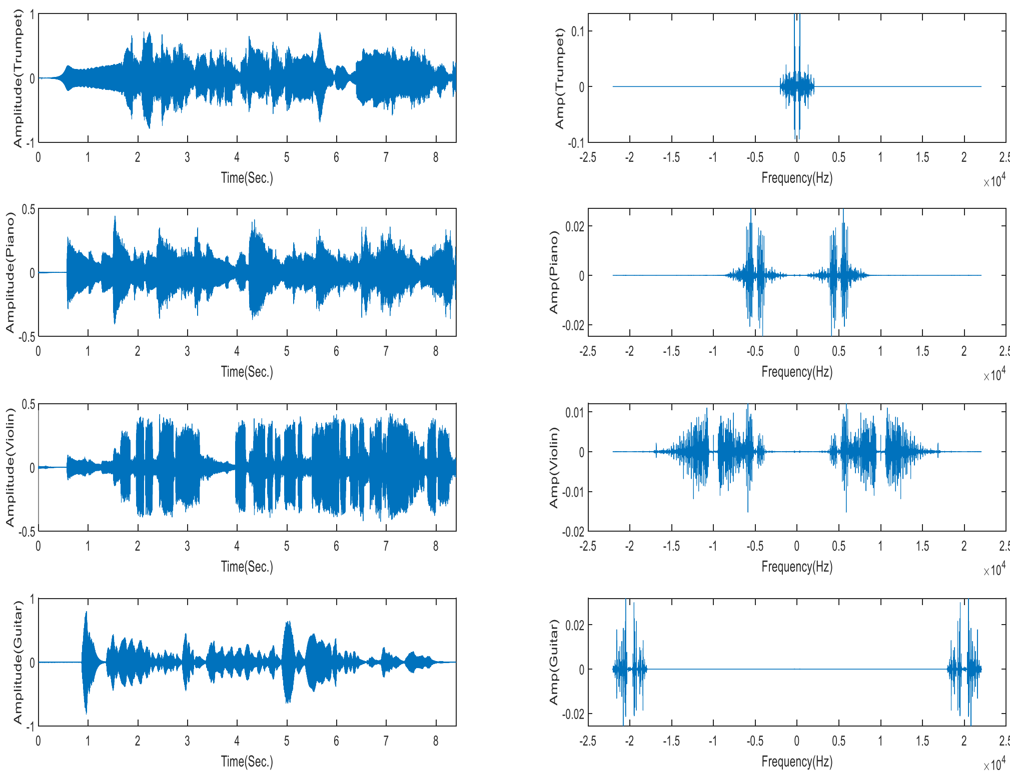


This is the frequency spectrum of overlapped signals. From the spectrum we can find the frequency range, which is 0~21992Hz approximately.

3. Designing filter that would separate the sound of each instrument from ‘final.wav’:

In my opinion, the filtering approach will not suffice to separate the sound of each instrument from the overlapping signal(final.wav). If there are any overlapping in frequency domain, it will be hard to separate the sound of each instrument by filtering approach.

Here, we will use four bandpass FIR filters of different frequency ranges to separate the sounds of different instruments. But this will not give us the exact separations.



4. Extraction of each instrument sound in separate '.wav' file:

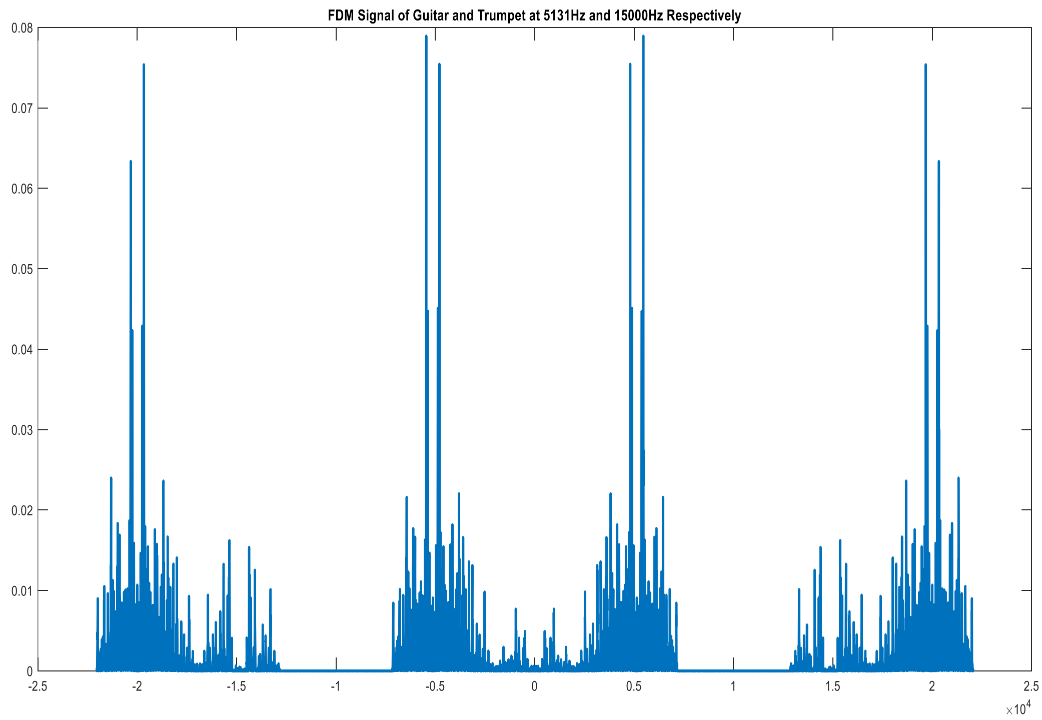
This is done by filtering and is attatched in Zip file.

5. Passing the individual wav files separately through a channel of bandwidth 0 to 10KHz:

We can say the channel as Low Pass Filter. This channel will block the frequencies above 10KHz. So, the channel will block the frequencies of Violin(partially) and Guiter.

6. Sending any two (Guitar and Trumpet) of the four separated signals through a two channel FDM link:

Here, we have taken the channel frequencies of 5131Hz and 15000Hz. We have tried to optimize the FDM link bandwidth cleverly while keeping the signal fidelity as high as possible.



7. The signal extracted from the ‘final.wav’ audio file lacks melody. Converting this into an overlapping yet melodious one with proper synchronization of octaves using MATLAB code:

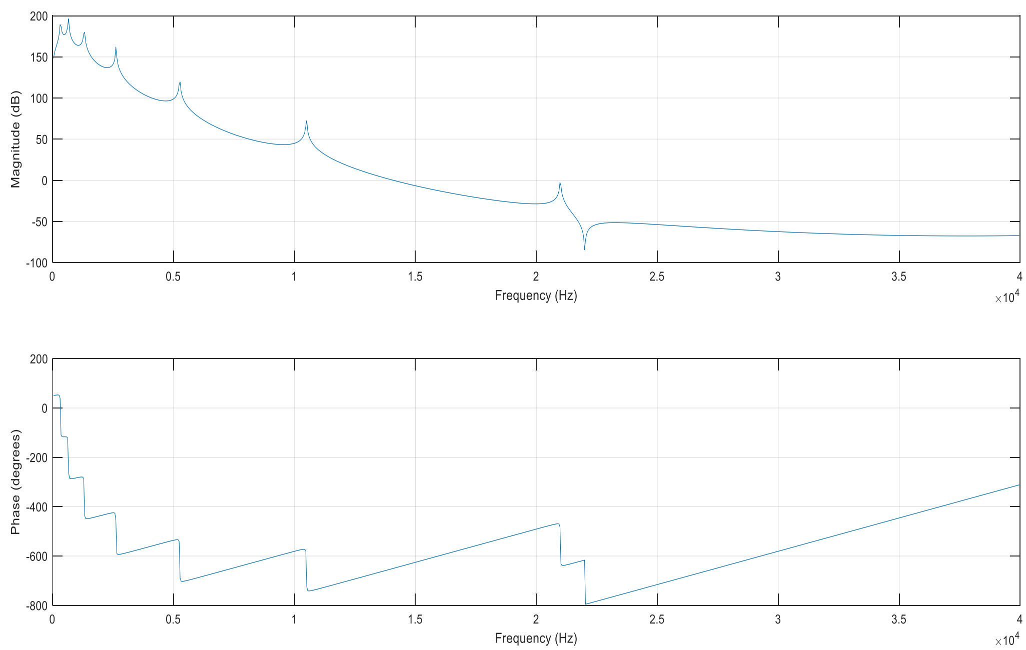
For octave synchronization, we have to pass the audio file through a multiband pass filter which will only pass the octave frequencies and will block the other frequency components.

We have designed the multiband FIR filter with the Pole-zero placement method. In this manner, we did not get the exact filter, but we have tried to get as close as possible.

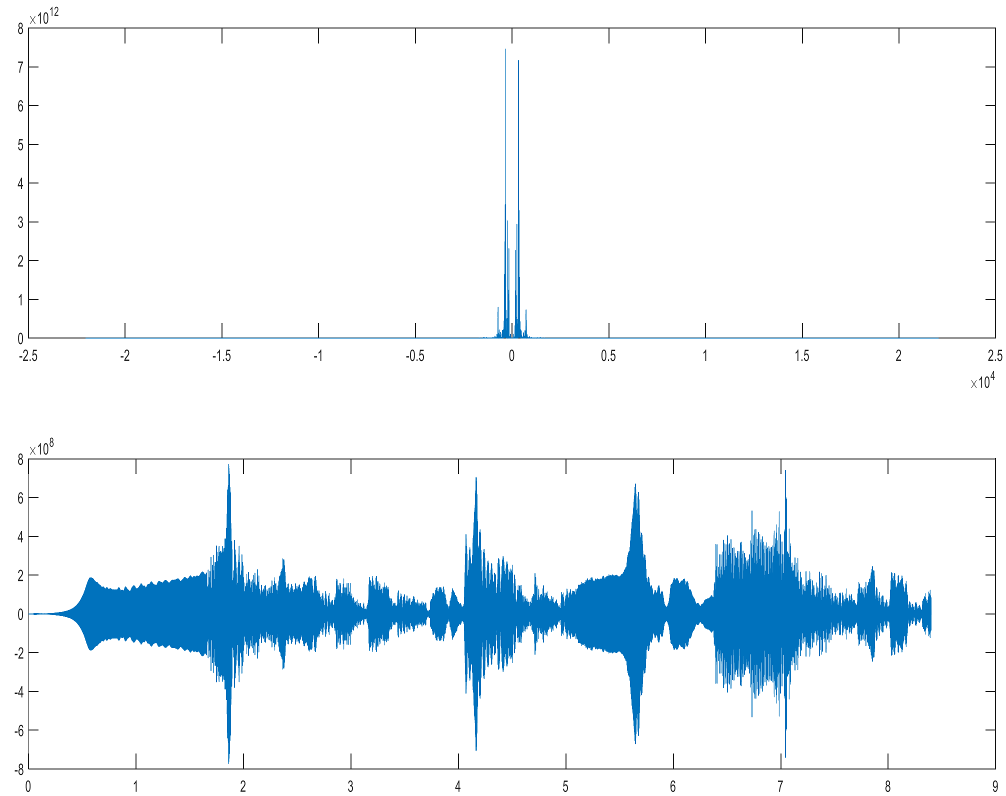
After passing through the filter, we have saved the filter output as an audio file with the name ‘overlappig_yet_melodious.wav’.

While designing filter, we have taken the lower frequency as 328Hz. And the calculated octave frequencies are: 328Hz, 656Hz, 1312Hz, 2624Hz, 5248Hz, 10496Hz, 20992Hz.

Designed Filter Response:



Filtered Output at Time and Frequency Domain:



Discussion: Through this project, we have got the experience of real-life signal processing. While doing this project, we have to design different filters, use various functions, and have to go through a lot of studies. We have faced the biggest challenges while designing the multiband filter for goal 7. The instructions provided for the project was sometimes vague for us, as a student. As for example, the point 1 in instruction, we have been said to find the frequency spectrum of each of the instrument's sound from 'NonOverlapping.wav'. But not being overlapped at time domain, doesn't guarantee of being not overlapped in frequency domain.