**# miniproject3\_DSP**

**Create Easy filter using python and some research about FIR filter**

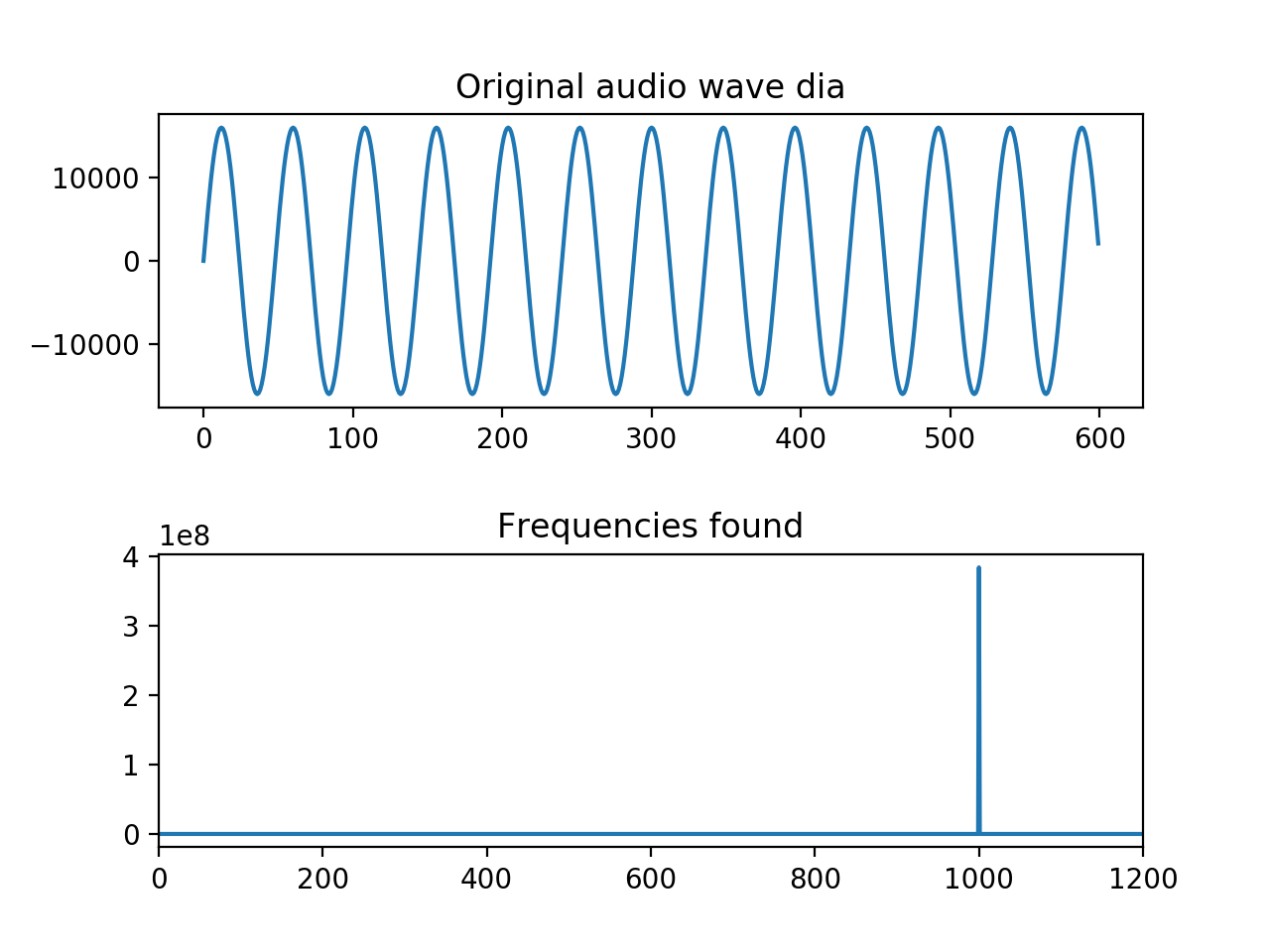
**Easy filter**

Library:

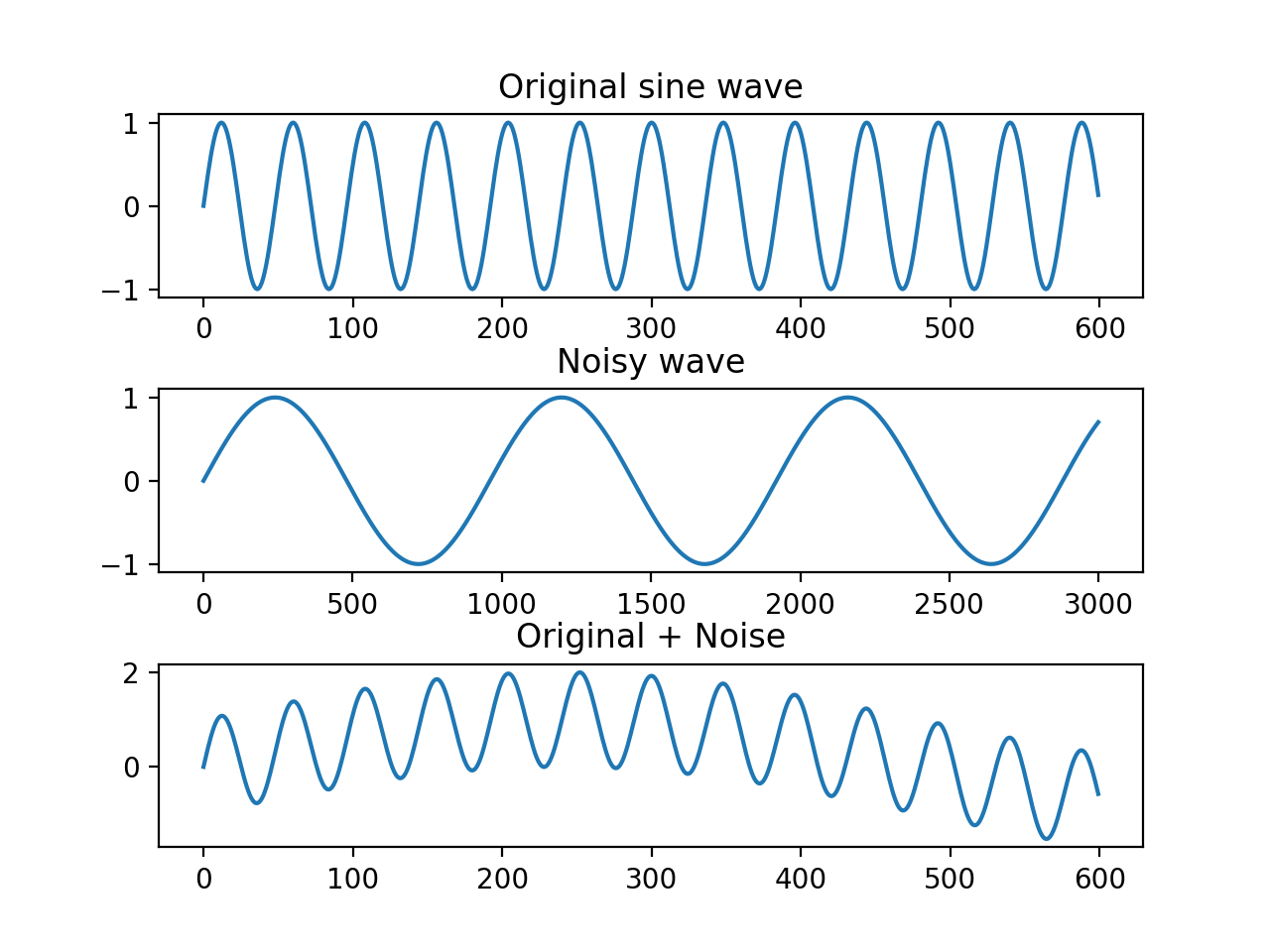
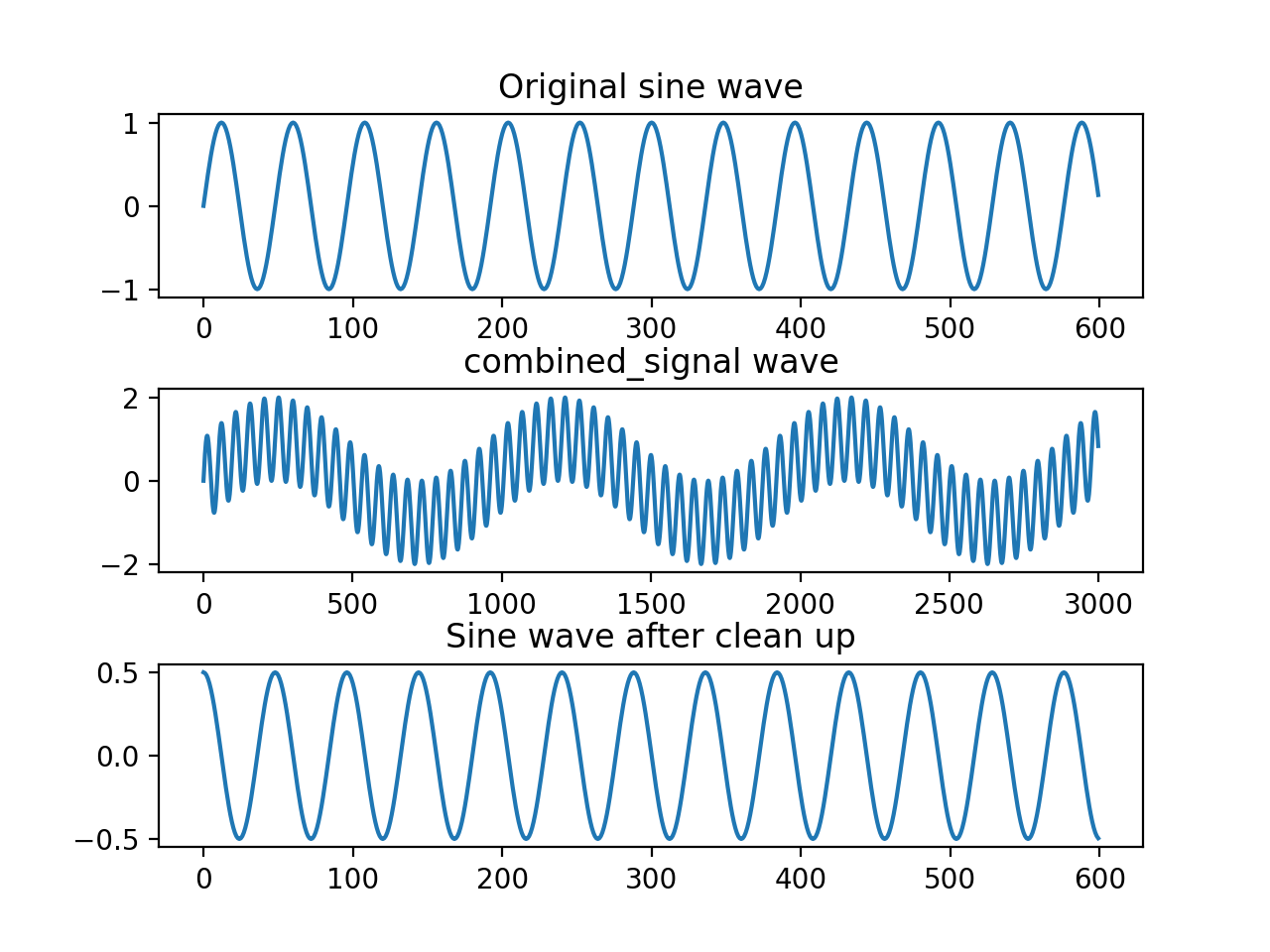
Matplotlib, wave, numpy.

First, create Sine signal with frequency of 1000HZ, amplitude of 16000, sampling rate of 48000, and pack the data with hexadecimal format so it produce 16 bit audio test.wav. Later, to check the frequency of radio, we unpack it

to readable format like binary. So we use FFT to get all the frequencies. Later we plot the diagram below.



Second, we create sine noise and add it with original sine wave to create new signal. Clearly as we can see, original sin wave becomes messy and from combinedfrequency.png we can see we have one more frequency of 50. Later we create a low pass filter to eliminate the noise which help us get the original signal. And from After\_filtering.png we can see the change of Frequency.



**FIR filter**

Library:

Matplotlib , scipy.signal , numpy

This is an easy test for difference of FIR filter on processing the signal integrally and separately. First we configurate the parameters of FIR filter , and set the parameters of signal with 44.1KHZ frequency and 1s frequency sweep. First, we directly process the signal using FIR, later, we cut the signal into 50 sets, processing every single set individually and combine them together after filtering. From the result we can see, there is no difference for FIR filter processing signal integrally and separately.

