3TP3 Lab 4 Report

Rui Qiu (400318681)

Shabab Rahman (400291441)

Instructor: Terry Todd

Section: T01

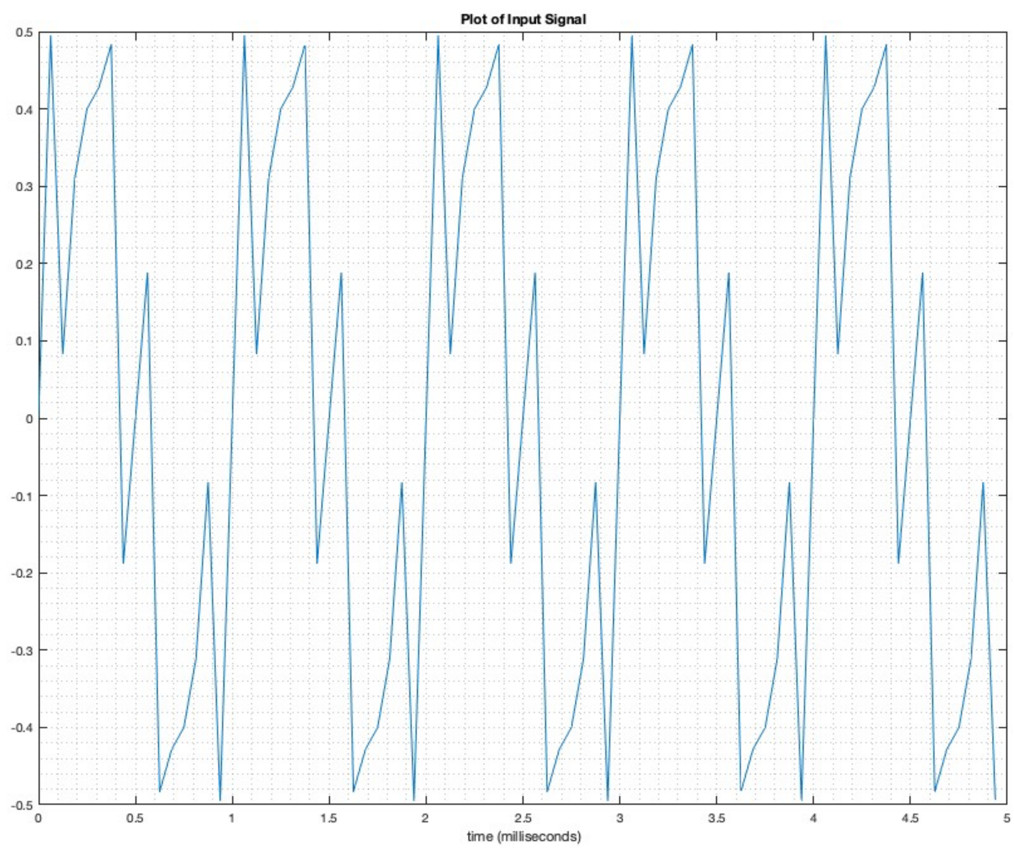
Part1:

2. I hear a high pitch of sound with a tiny up and down frequency changes for 10seconds.

3. The code for displaying the sound waveform for 5 msec.

```
% Read in the signal from the audio file
[signal, Fs] = audioread("tones2022.wav");
L = length(signal);
T = 1/Fs;
t = (0:L-1)*T;

% Plot the signal for t_plot msec
t_plot = 5;
msec_per_sec = 1000;
numSamples = t_plot*Fs/msec_per_sec;
plot(msec_per_sec*t(1:numSamples), signal(1:numSamples))
title('Plot of Input Signal')
xlabel('time (milliseconds)')
grid('minor');
```

The plot:

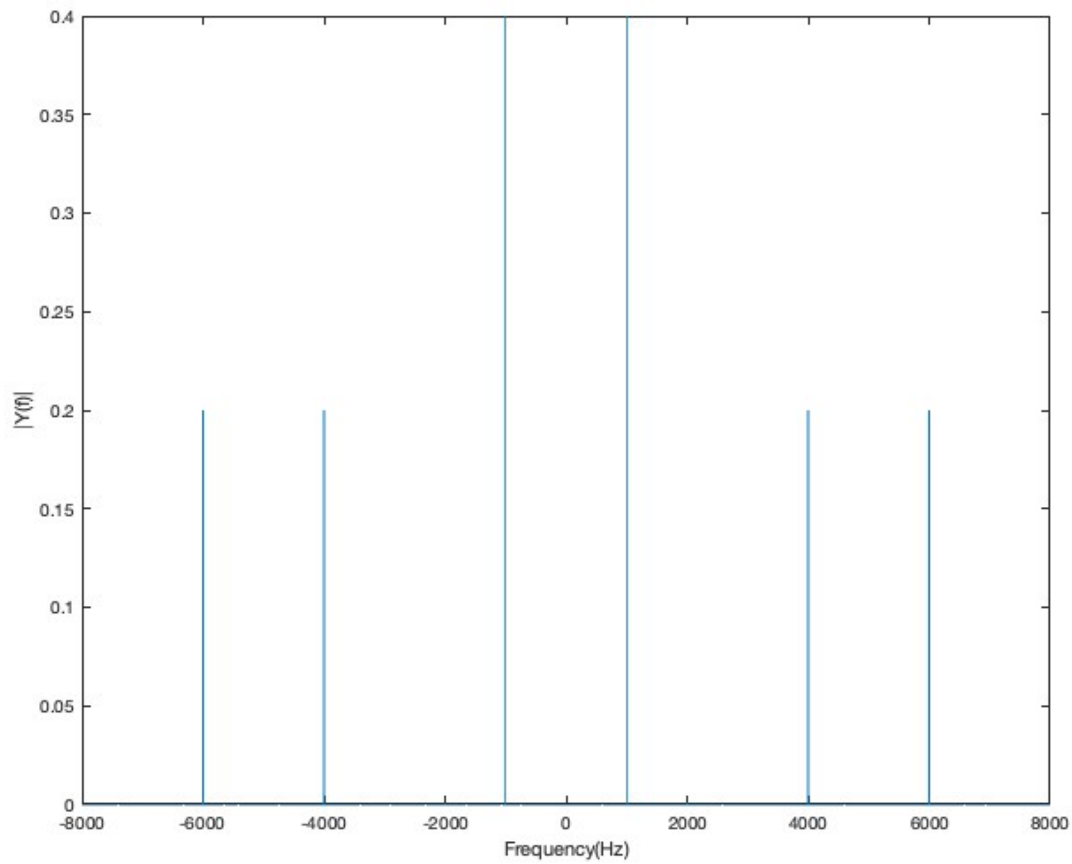


4. Based on the plot from Q3. I would estimate this signal is a combination of three frequencies. As the plot shown above, there are approximately three spikes above zero and those spikes also follows a pattern which will repeat themselves for every one millisecond. As a result, this signal could be a combination of three different sine frequencies with different periods and amplitudes.

5.

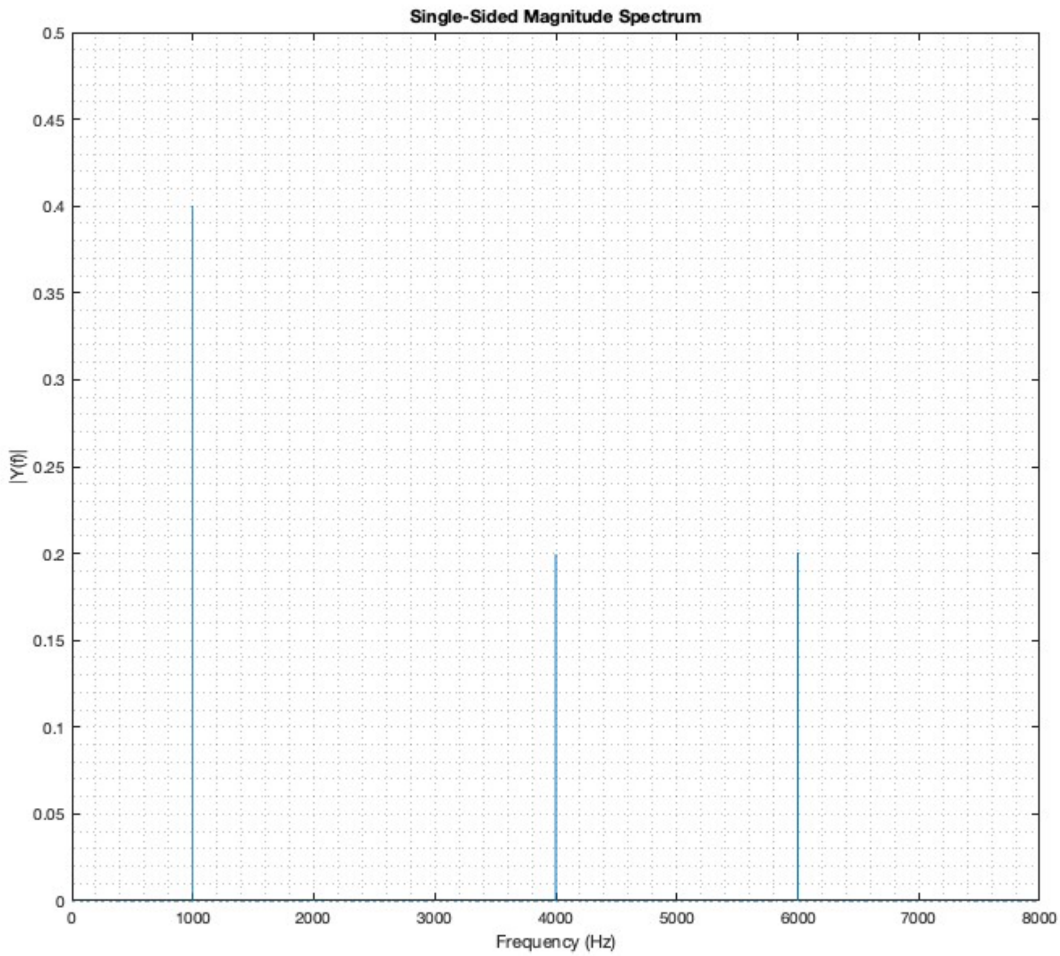
```
clc;
clear;
% Read in the signal from the audio file
[signal, Fs] = audioread("tones2022.wav");
L = length(signal);
T = 1/Fs;
t = (0:L-1)*T;

% Plot the signal for t_plot msec
t_plot = 5;
msec_per_sec = 1000;
numSamples = t_plot*Fs/msec_per_sec;


X = fft(signal);
Y = fftshift(X);
fshift = (-L/2:L/2-1)*(Fs/L); % zero-centered frequency range
P2 = 2.*abs(Y/L);

plot(fshift,P2);
xlabel("Frequency(Hz)");
ylabel("|Y(f)|");
```

The plot:



The matlab fft function is used to compute the discrete Fourier transform(DFT) of signal using a fast Fourier transform algorithm. It will output the DFT of a signal within the interval [0, fs]. Since the spectrum is extended in a periodic order. So the spectrum in the negative interval will symmetrical as the y axis which to observe the fully plot, ffshift can be used to shift zero-frequency component to center of spectrum.

Q6:
The plot:

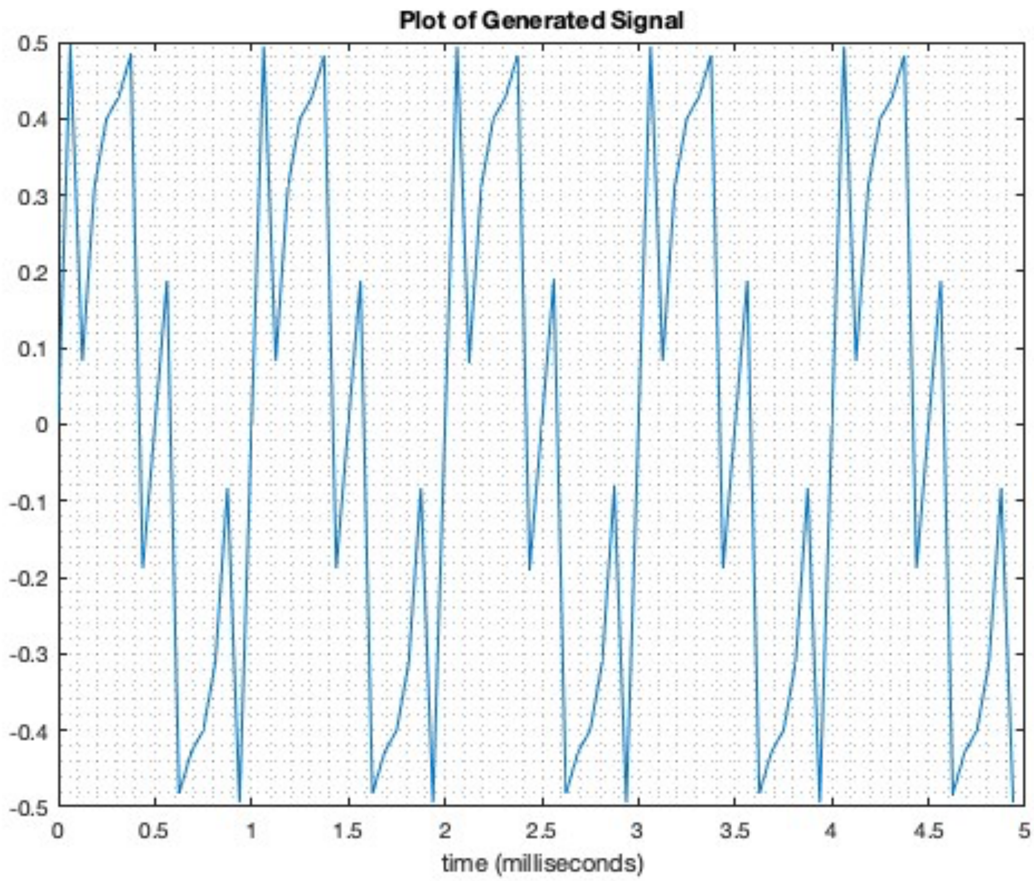


7.
Based on the plot result from Q6, we can conclude the frequency and magnitude of this sinusoids are 1000 Hz, 4000 Hz, and 6000 Hz with magnitudes of 0.4, 0.2 and 0.2 respectively.

8. Code:
```
clc;
clear;

Fs=16000;
L = Fs * 10;
T = 1/Fs;
t = (0:L-1)*T;

freqs = [1000, 4000, 6000];
amps = [0.4, 0.2, 0.2];

%combine signals
y = sum(amps .* sin(2 * pi * freqs .* t'), 2)';
```

```
t_plot = 5;
msec_per_sec = 1000;
numSamples = t_plot*Fs/msec_per_sec;
plot(msec_per_sec*t(1:numSamples), y(1:numSamples))
title('Plot of Generated Signal')
xlabel('time (milliseconds)')
grid('minor');
```

Plot:



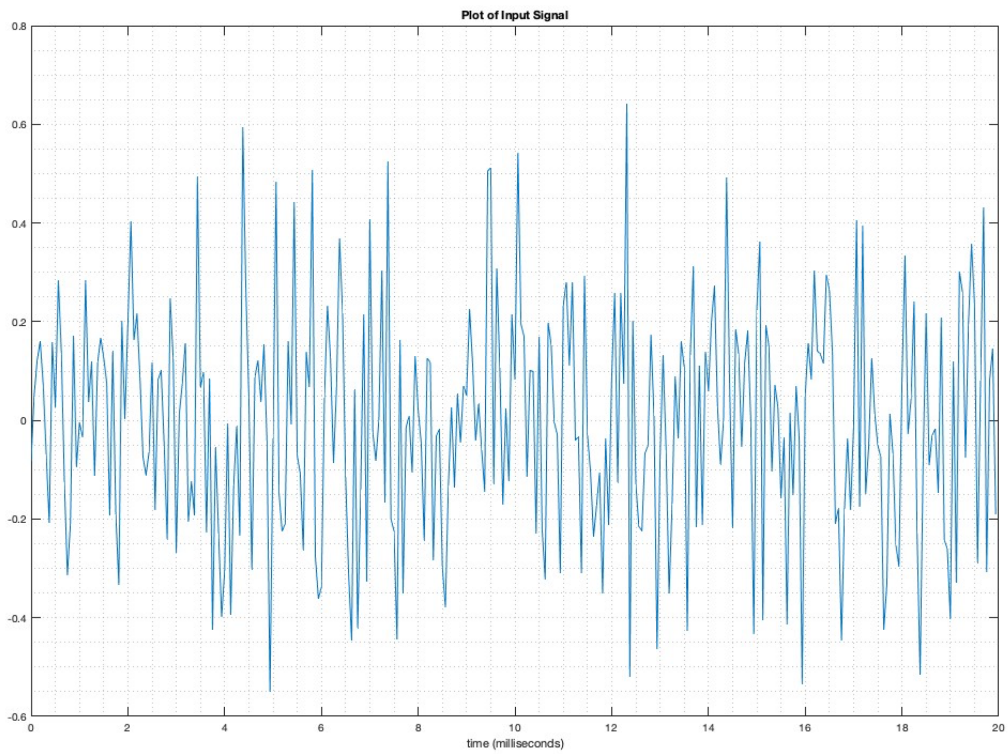
Compare the plot to the input signal from Q3, we can find they have the same shape with the same spike. Which it can conclude the estimate that the input signal is a combination of three sine signals where with the frequency 1000 Hz, 4000 Hz, and 6000 Hz and magnitudes of 0.4, 0.2 and 0.2 respectively. We are using FFT matlab algorithm to find the DFT frequency and amplitude and prove that the sum of these three frequencies are exactly the same as the input frequency.

Part2:

2. I hear the SecretMessage2022.wav was a continuous, hoarse sound with a lot of frequency changes that lasted 1 minute and 16 seconds.

3.

```
[signal, Fs] = audioread("SecretMessage2022.wav");
L = length(signal);
T = 1/Fs;
t = (0:L-1)*T;

% Plot the signal for t_plot msec
t_plot = 20;
msec_per_sec = 1000;
numSamples = t_plot*Fs/msec_per_sec;
plot(msec_per_sec*t(1:numSamples), signal(1:numSamples))
title('Plot of Input Signal')
xlabel('time (milliseconds)')
grid('minor');
```

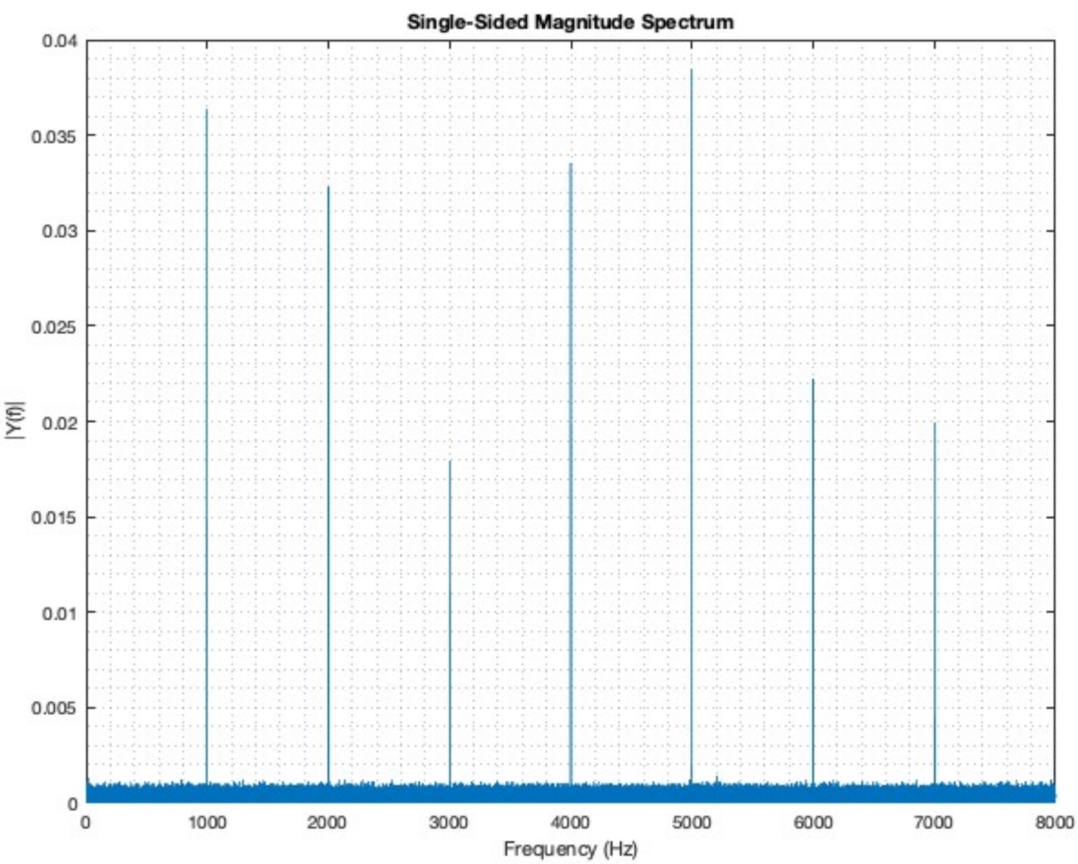


Up is the plot of the input signal for 20 milliseconds, unlike the plot in part 1, it is difficult to distinguish the pattern from the spike changes. So I will use the FFT to find the DFT of the input signal.

Using the code for part1 Q6 to find the DFT.

```
[signal, Fs] = audioread("SecretMessage2022.wav");
L = length(signal);
T = 1/Fs;
t = (0:L-1)*T;
```

```
% Plot the signal for t_plot msec
t_plot = 20;
msec_per_sec = 1000;
numSamples = t_plot*Fs/msec_per_sec;
plot(msec_per_sec*t(1:numSamples), signal(1:numSamples))
title('Plot of Input Signal')
xlabel('time (milliseconds)')
grid('minor');

% Take the DFT
Y = fft(signal)/L;
f = Fs/2*linspace(0,1,L/2+1);
% Plot the single-sided magnitude spectrum.
plot(f,2*abs(Y(1:L/2+1)));
title('Single-Sided Magnitude Spectrum')
xlabel('Frequency (Hz)')
ylabel('|Y(f)|')
% axis([0 Fs/2 0 .5]);
grid('minor');
```



Q4.
To extract the signal from 76 seconds into 1 second per each. We observed that by using vector indexing, we want the signal to get incrementing by 16000 per second as following the algorithm shown below.

```
y = signal(1:Fs);
q = signal(Fs+1:2*Fs);
d = signal(2*Fs+1:3*Fs);
```

Therefore, we can write a for loop to extract each second signal by using DFT.

Code Below:

```
clc;
clear;

[signal, Fs] = audioread('SecretMessage2022.wav');
L = length(signal);
T = 1/Fs;
freqs = zeros(76, 4);
set(gcf,'position',[10,10,2500,1200]);
loops_time = L/Fs;
time = 0;
% y = signal(1:Fs);
% q = signal(Fs+1:2*Fs);
% d = signal(2*Fs+1:3*Fs);
for i = 0:loops_time-1
    Y = fft(signal((1+i*Fs):(i+1)*Fs))/Fs;
    [B, I] = maxk(abs(Y(1:Fs / 2 + 1)), 4); %use maxk function to find the
largest four amplitudes
    freqs(i + 1, :) = I - 1; %record the horizontal of largest amp index
    f = Fs/2*linspace(0,1,Fs/2+1);
    time = time+1;
    subplot(10,8,time);
    plot(f,2*abs(Y(1:Fs/2+1)));
    title(time+"s");
    xlabel('Frequency (Hz)')
    ylabel('|Y(f)|')
    axis([0 Fs/2 0 .1]);
    grid('minor');

end

freqs = sort(freqs, 2); %sort the matrix from low to high
dec_msg = "";

for i = 1:loops_time
    if isequal(freqs(i, :),[1000, 2000, 3000, 4000])
        dec_msg = strcat(dec_msg, "A");
    elseif isequal(freqs(i, :),[1000, 2000, 3000, 5000])
        dec_msg = strcat(dec_msg, "B");
    elseif isequal(freqs(i, :) , [1000, 2000, 3000, 6000])
        dec_msg = strcat(dec_msg, "C");
    elseif isequal(freqs(i, :) , [1000, 2000, 3000, 7000])
        dec_msg = strcat(dec_msg, "D");
    elseif isequal(freqs(i, :) , [1000, 2000, 4000, 5000])
```

```
        dec_msg = strcat(dec_msg, "E");
    elseif isequal(freqs(i, :) , [1000, 2000, 4000, 6000])
        dec_msg = strcat(dec_msg, "F");
    elseif isequal(freqs(i, :) , [1000, 2000, 4000, 7000])
        dec_msg = strcat(dec_msg, "G");
    elseif isequal(freqs(i, :) , [1000, 2000, 5000, 6000])
        dec_msg = strcat(dec_msg, "H");
    elseif isequal(freqs(i, :) , [1000, 2000, 5000, 7000])
        dec_msg = strcat(dec_msg, "I");
    elseif isequal(freqs(i, :) , [1000, 2000, 6000, 7000])
        dec_msg = strcat(dec_msg, "J");
    elseif isequal(freqs(i, :) , [1000, 3000, 4000, 5000])
        dec_msg = strcat(dec_msg, "K");
    elseif isequal(freqs(i, :) , [1000, 3000, 4000, 6000])
        dec_msg = strcat(dec_msg, "L");
    elseif isequal(freqs(i, :) , [1000, 3000, 4000, 7000])
        dec_msg = strcat(dec_msg, "M");
    elseif isequal(freqs(i, :) , [1000, 3000, 5000, 6000])
        dec_msg = strcat(dec_msg, "N");
    elseif isequal(freqs(i, :) , [1000, 3000, 5000, 7000])
        dec_msg = strcat(dec_msg, "O");
    elseif isequal(freqs(i, :) , [1000, 3000, 6000, 7000])
        dec_msg = strcat(dec_msg, "P");
    elseif isequal(freqs(i, :) , [1000, 4000, 5000, 6000])
        dec_msg = strcat(dec_msg, "Q");
    elseif isequal(freqs(i, :) , [1000, 4000, 5000, 7000])
        dec_msg = strcat(dec_msg, "R");
    elseif isequal(freqs(i, :) , [1000, 4000, 6000, 7000])
        dec_msg = strcat(dec_msg, "S");
    elseif isequal(freqs(i, :) , [1000, 5000, 6000, 7000])
        dec_msg = strcat(dec_msg, "T");
    elseif isequal(freqs(i, :) , [2000, 3000, 4000, 5000])
        dec_msg = strcat(dec_msg, "U");
    elseif isequal(freqs(i, :) , [2000, 3000, 4000, 6000])
        dec_msg = strcat(dec_msg, "V");
    elseif isequal(freqs(i, :) , [2000, 3000, 4000, 7000])
        dec_msg = strcat(dec_msg, "W");
    elseif isequal(freqs(i, :) , [2000, 3000, 5000, 6000])
        dec_msg = strcat(dec_msg, "X");
    elseif isequal(freqs(i, :) , [2000, 3000, 5000, 7000])
        dec_msg = strcat(dec_msg, "Y");
    elseif isequal(freqs(i, :) , [2000, 3000, 6000, 7000])
        dec_msg = strcat(dec_msg, "Z");
    elseif isequal(freqs(i, :) , [2000, 4000, 5000, 6000])
        dec_msg = strcat(dec_msg, " ");
    elseif isequal(freqs(i, :) , [2000, 4000, 5000, 7000])
        dec_msg = strcat(dec_msg, ".");
    end
end
dec_msg
```

Outpu:

```
dec_msg =

    "NEVER LET THE FEAR OF STRIKING OUT KEEP YOU FROM PLAYING THE GAME.BABE RUTH."
```

DFT graph for each second signal input: