ECE 311 Lab 7

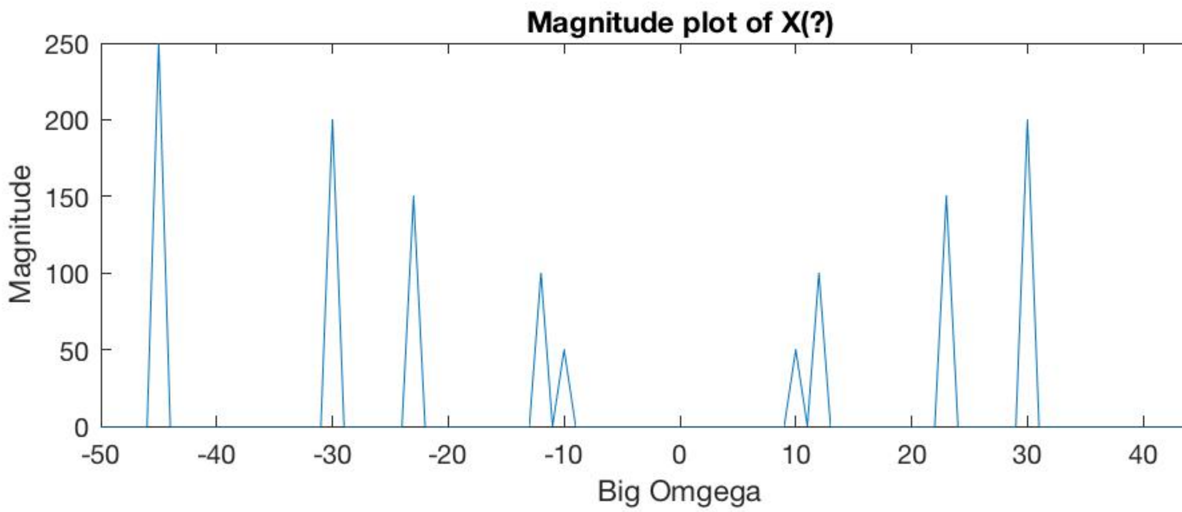
Jacob Hutter

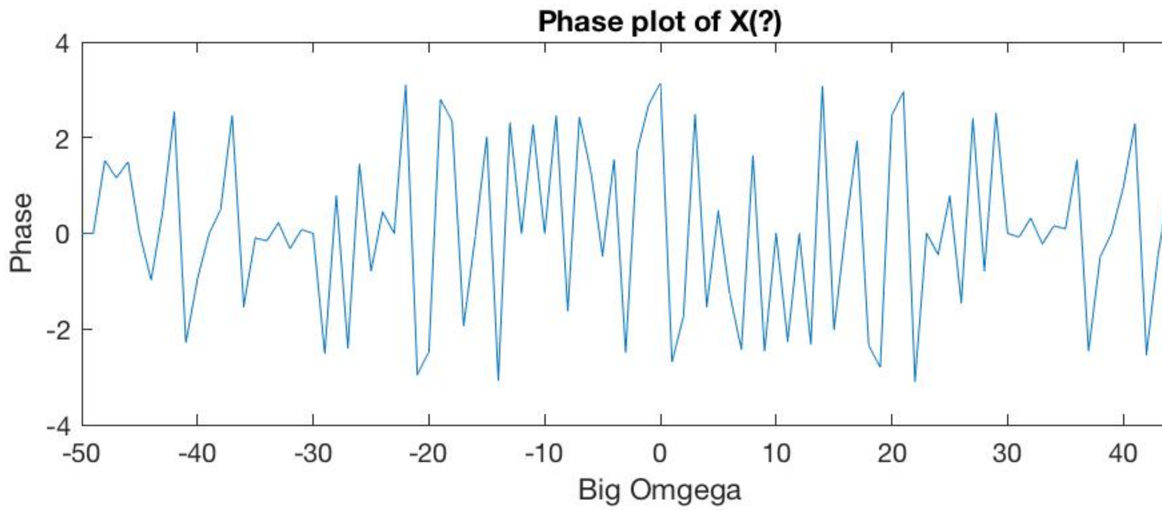
May 7, 2017

Sakanaya

```
clear all;
clc;

load signal.mat;

N = length(x);

w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians

% using w = Big w * T with T = 1/100, big omega = w * 100 / 2pi
w = w.*100/(2*pi);

x_w = fftshift(fft(x));

figure;
subplot(211);
plot(w,abs(x_w));
title('Magnitude plot of X(?)');
xlabel('Big Omgega');
ylabel('Magnitude');

subplot(212);
plot(w,angle(x_w));
title('Phase plot of X(?)');
xlabel('Big Omgega');
ylabel('Phase');
```

Sakanaya.m





From the plot we see that there are tones at $+/-10, 12, 22, 30 and 45Hz$ so a total of 10 tones.

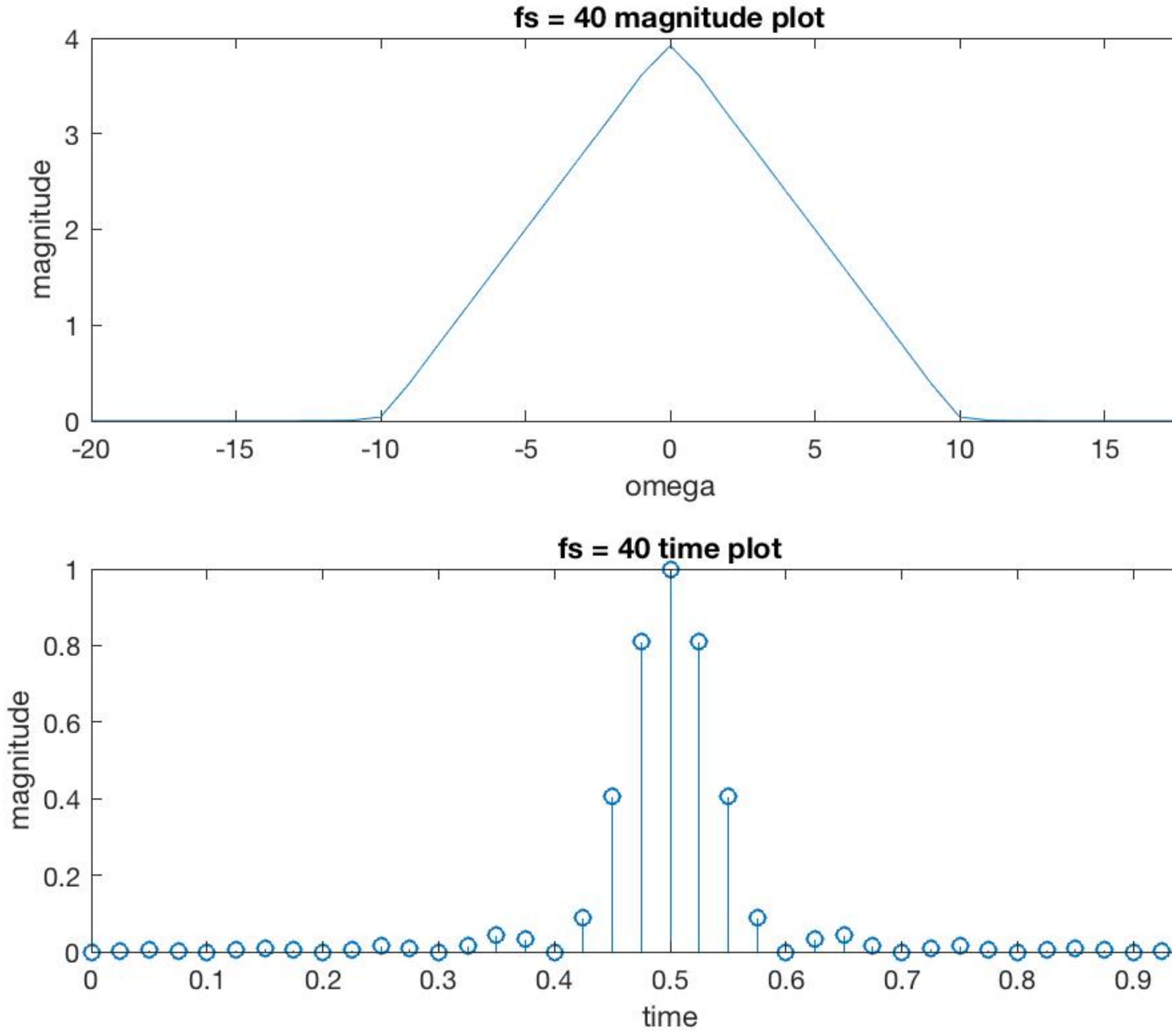
Chipotle

```
clear all;
clc;

load samplerate.mat;
fs = 40;
N = 40;

w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
```

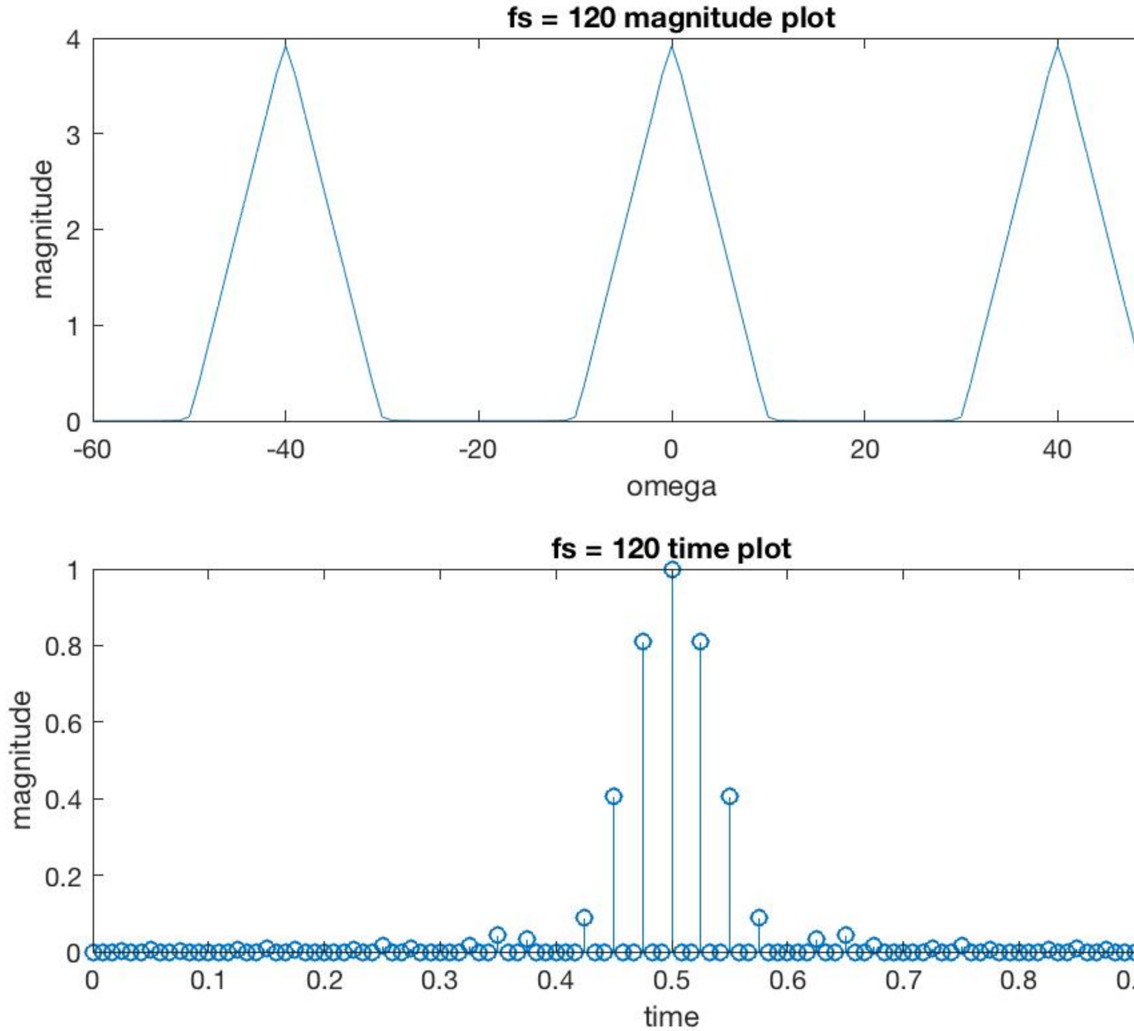
```
x_w = fftshift(fft(x));
t = linspace(0,N-1,N) * 1/fs;
w = 40 * w/(2*pi);
figure;
subplot(211);
plot(w,abs(x_w));
title('fs = 40 magnitude plot');
xlabel('omega');
ylabel('magnitude');
subplot(212);
stem(t,x);
title('fs = 40 time plot');
xlabel('time');
ylabel('magnitude');


x_up = upsample(x,3);
x_up_w = fftshift(fft(x_up));
N = length(x_up);
t = linspace(0,N-1,N) .* 1/N;
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi;
w = 120 * w/(2*pi);
figure;
subplot(211);

plot(w,abs(x_up_w));
title('fs = 120 magnitude plot');
xlabel('omega');
ylabel('magnitude');
subplot(212);
stem(t,x_up);
title('fs = 120 time plot');
xlabel('time');
ylabel('magnitude');

for i = 1:length(x_up)
    if(abs(w(i)) > 25)
        x_up_w(i) = 0;
    end
end
figure;
subplot(211);
plot(w,abs(x_up_w));
title('fs = 120 filtered magnitude plot');
xlabel('omega');
ylabel('magnitude');
subplot(212);
x_up = ifft(ifftshift(x_up_w));
plot(t,abs(x_up));
title('fs = 120 filtered time plot');
xlabel('time');
ylabel('magnitude');

x_down = downsample(x_up,2);
N = N/2;
t = linspace(0,N-1,N)/N;
```

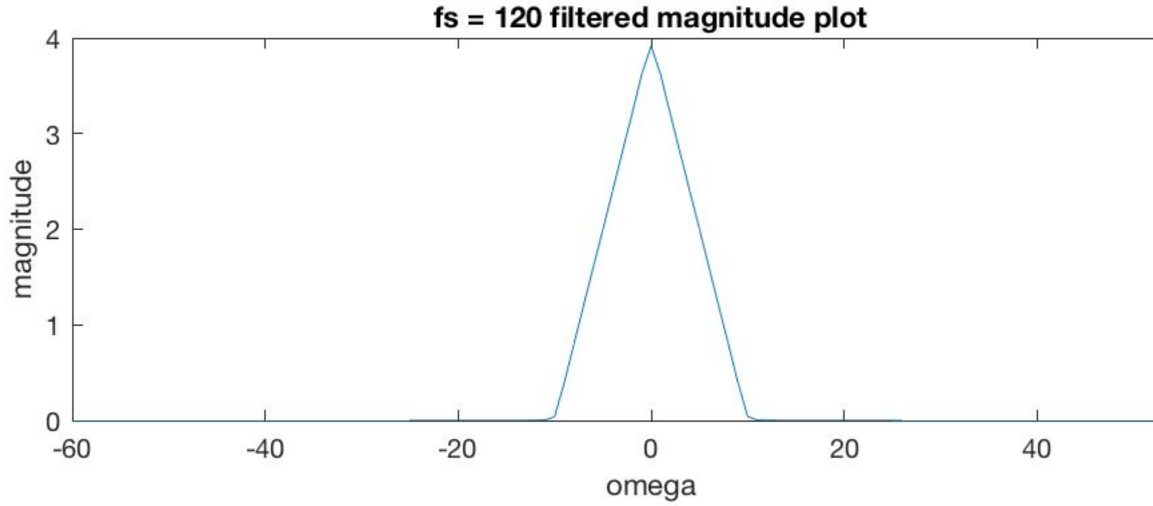
```
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi;
w = N * w/(2*pi);
figure;
subplot(211);
plot(t,abs(x_down));
title('fs = 60 filtered time plot');
xlabel('time');
ylabel('magnitude');
subplot(212);
plot(w,abs(fftshift(fft(x_down))));
title('fs = 60 filtered magnitude plot');
xlabel('omega');
ylabel('magnitude');
```

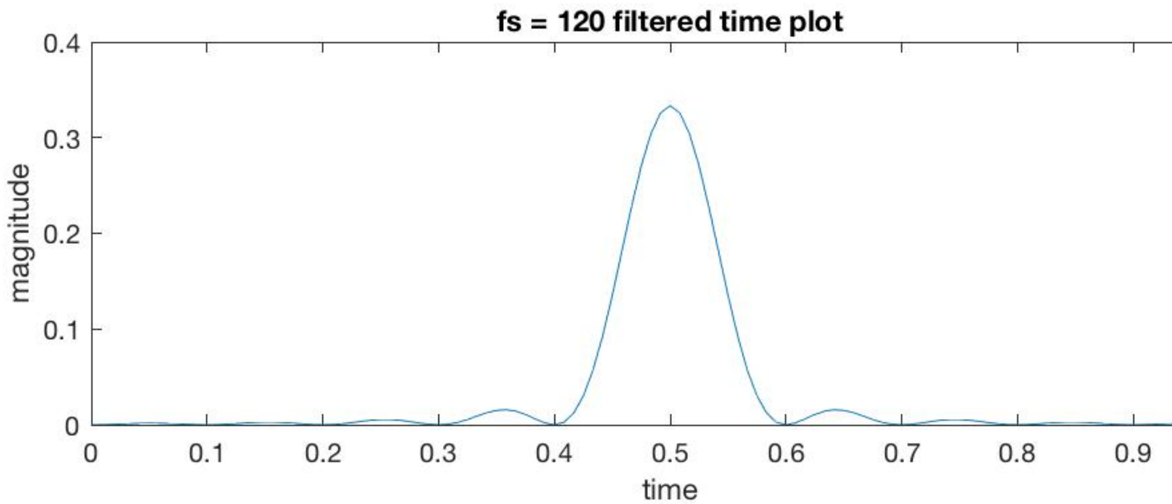
Chipotle.m

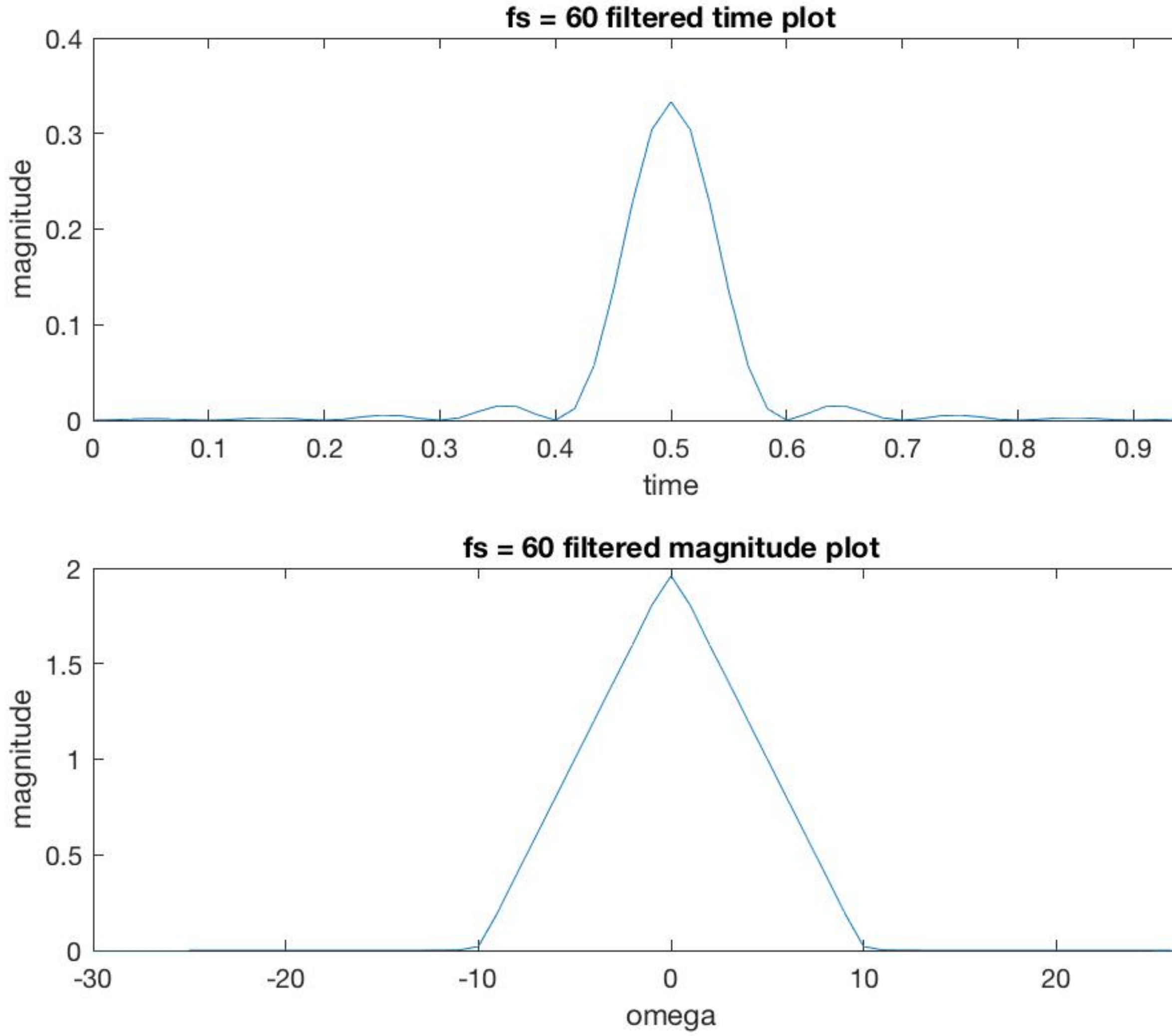


fs = 40 magnitude plot
magnitude
omega
fs = 40 time plot
magnitude
time


fs = 120 magnitude plot
magnitude
omega
fs = 120 time plot
magnitude
time


fs = 120 filtered magnitude plot
magnitude
omega
4
3
2
1
0
-60
-40
-20
0
20
40


fs = 120 filtered time plot
magnitude
time
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9



After upsampling, we see the frequency effect because of the time and frequency scaling property of the Fourier transform. The maximum we could have downsampled in order to avoid aliasing is 4.

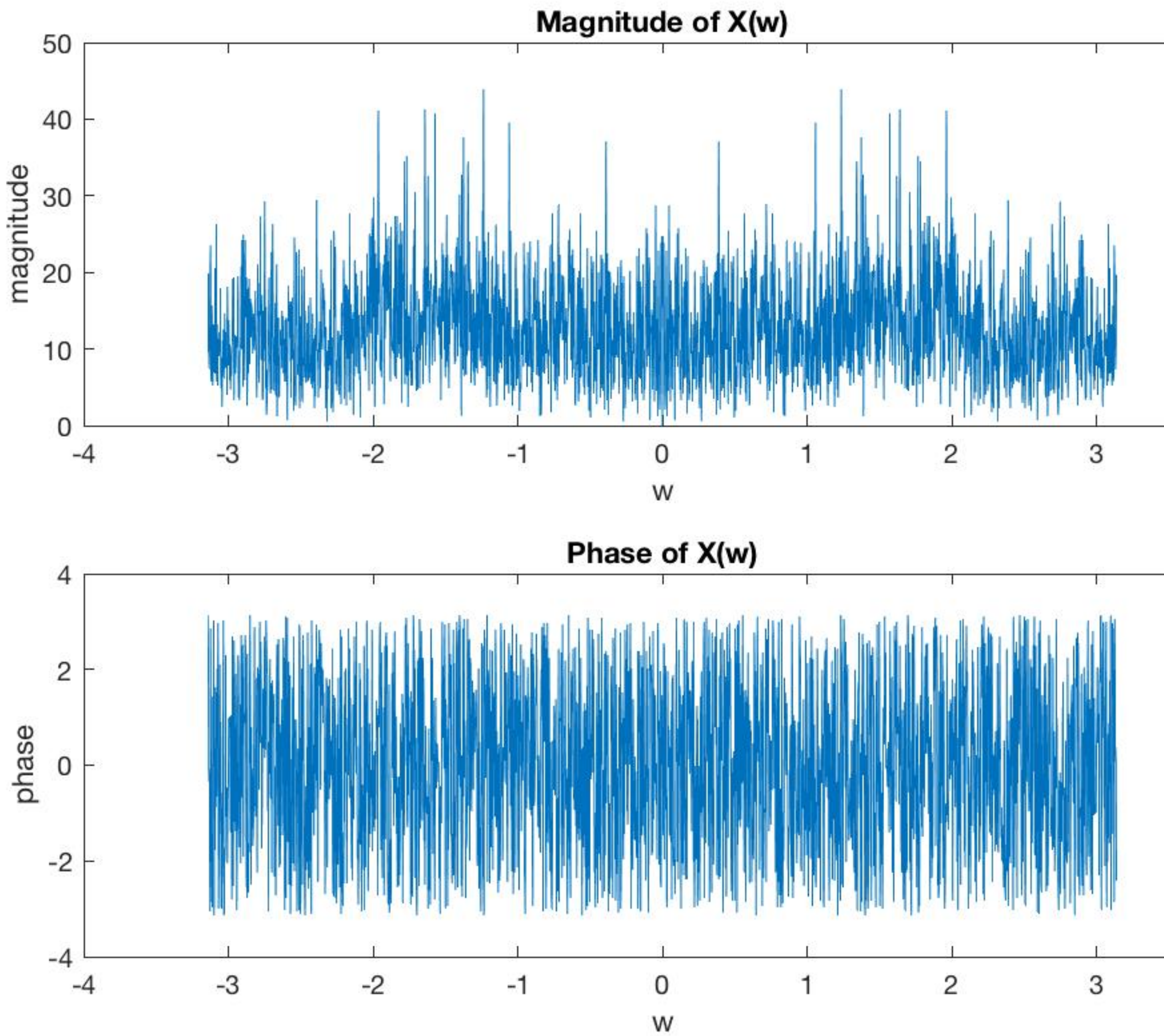
Legends

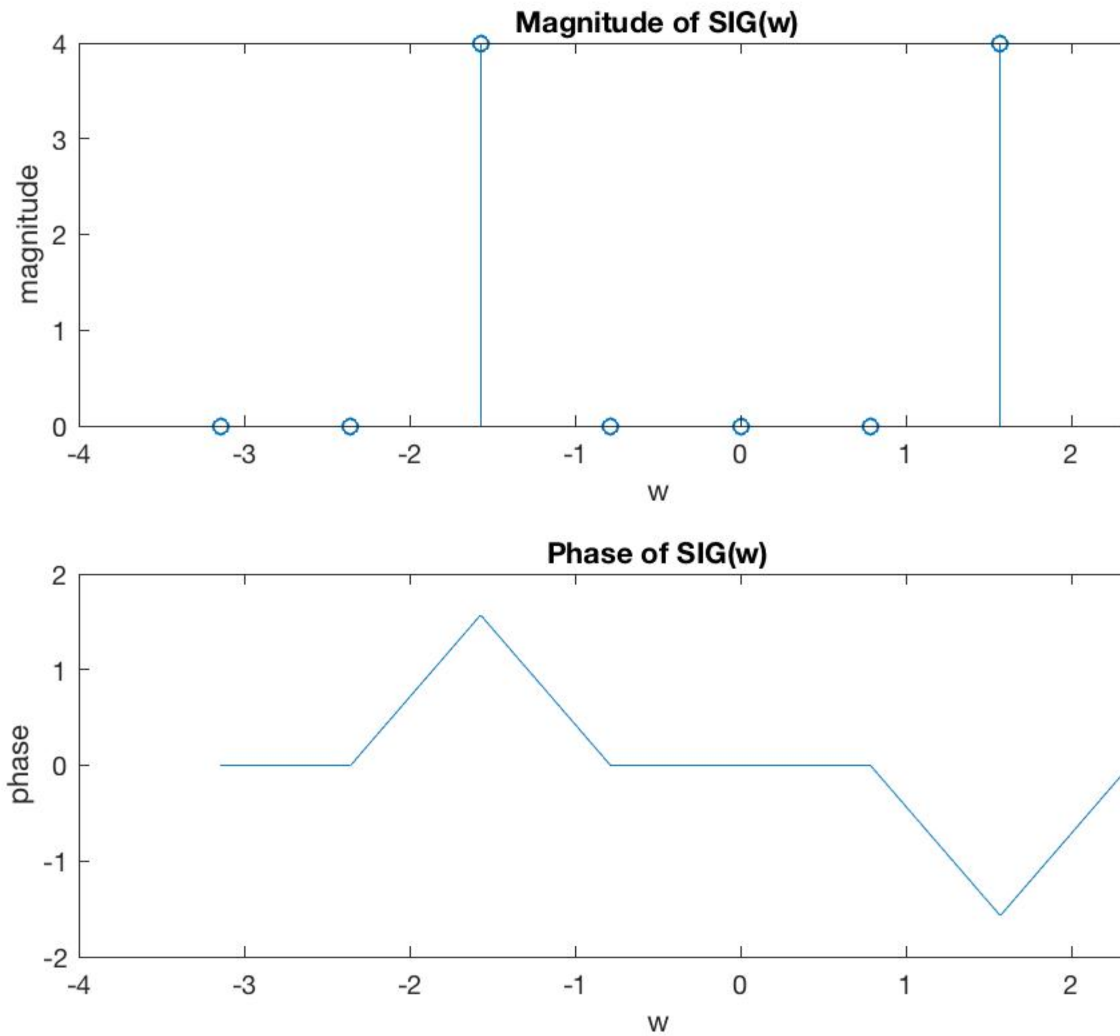
```
clear all;
clc;

load q1_signal.mat;
N = length(x);

w = fftshift((0:N-1)/N*2*pi);
```

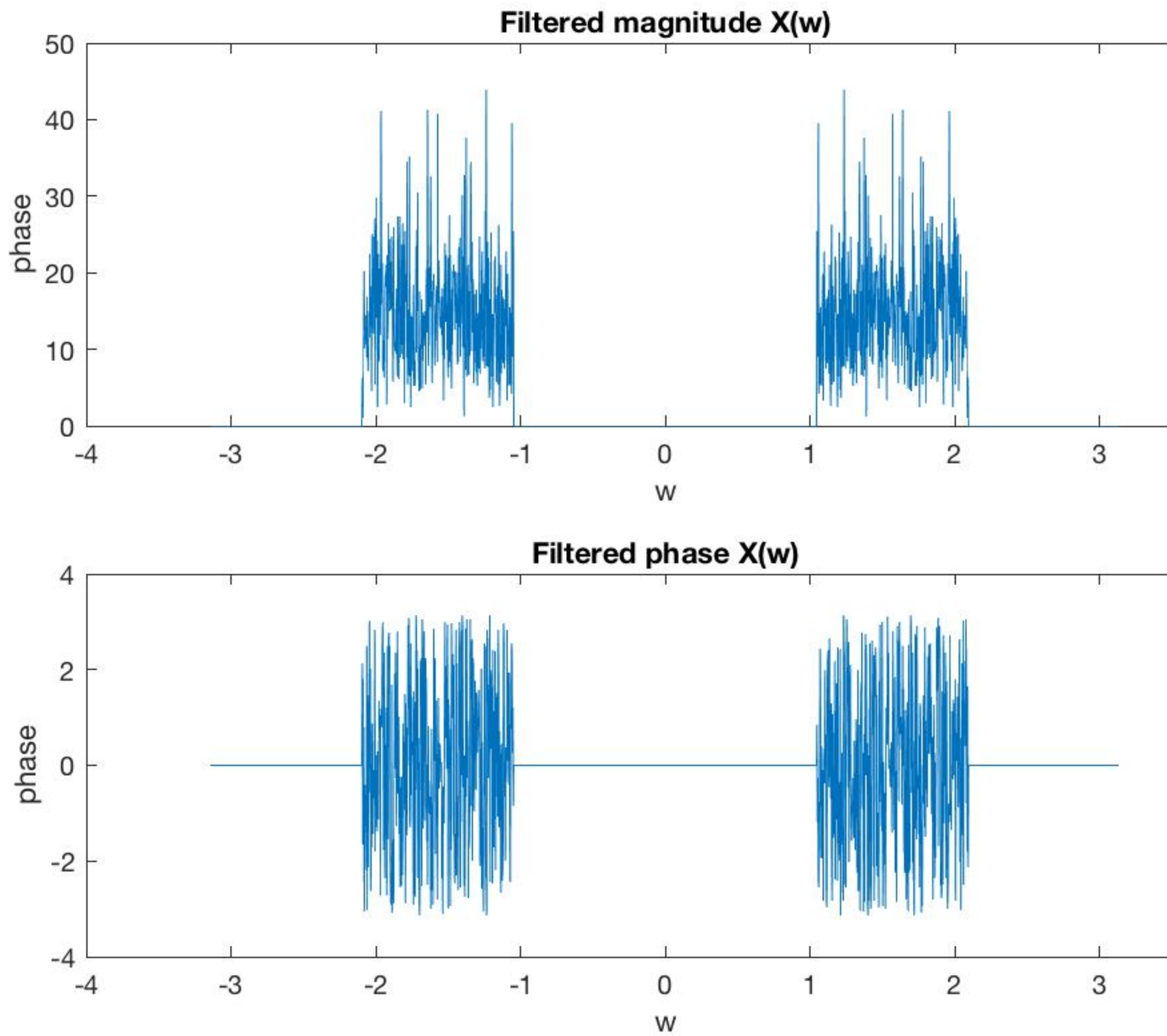
```
w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
w = w';
figure;
subplot(211);
x_w = fftshift(fft(x));
plot(w,abs(x_w));
title('Magnitude of X(w)');
xlabel('w');
ylabel('magnitude');
subplot(212);
plot(w,angle(x_w));
title('Phase of X(w)');
xlabel('w');
ylabel('phase');
%%%%%%%%%%%%%%% end part 1 %%%%%%%%%%%%%

sig_w = fftshift(fft(sig));
N2 = length(sig);
w2 = fftshift((0:N2-1)/N2*2*pi);
w2(1:N2/2) = w2(1:N2/2) - 2*pi; % get freq in radians
figure;
subplot(211);
stem(w2,abs(sig_w));
title('Magnitude of SIG(w)');
xlabel('w');
ylabel('magnitude');
subplot(212);
plot(w2,angle(sig_w));
title('Phase of SIG(w)');
xlabel('w');
ylabel('phase');

%%%%%%%% end part 2 %%%%%%%%%%%%%%%%


% filter x
for i=1:length(x_w)
    if(w(i) < -2*pi/3 || w(i) > 2*pi/3)
        x_w(i) = 0;
    end
    if(abs(w(i)) < pi/3)
        x_w(i) = 0;
    end

end

figure;
subplot(211);
plot(w,abs(x_w));
title('Filtered magnitude X(w)');
xlabel('w');
ylabel('phase');
subplot(212);
plot(w,angle(x_w));
title('Filtered phase X(w)');
xlabel('w');
ylabel('phase');
```

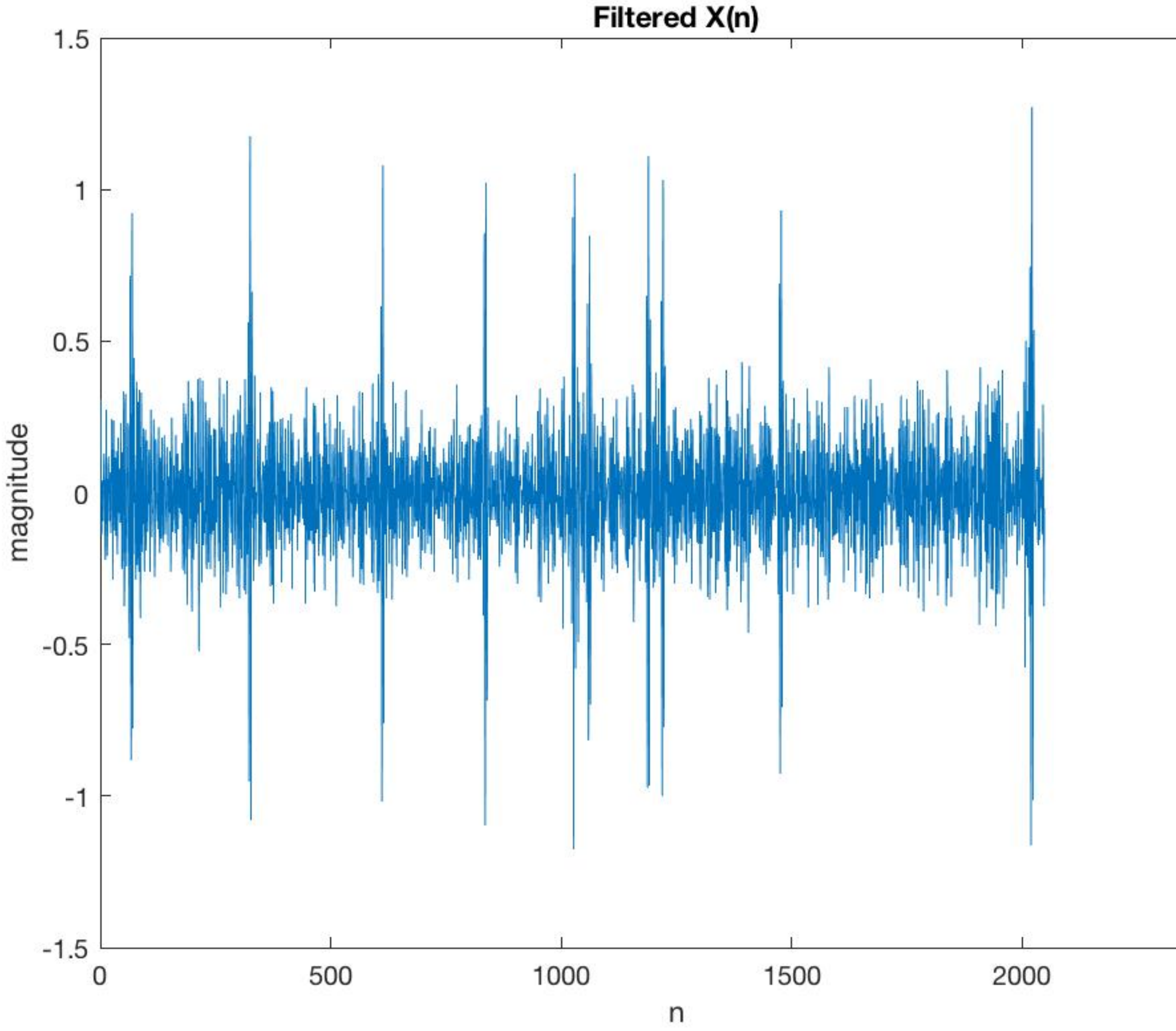
```

x_new = ifft(ifftshift(x_w));
figure;
plot(x_new);
title('Filtered X(n)');
xlabel('n');
ylabel('magnitude');
```

Legends.m




Magnitude of SIG(w)
magnitude
w
Phase of SIG(w)
phase
w


Filtered magnitude X(w)
phase
w
Filtered phase X(w)
phase
w



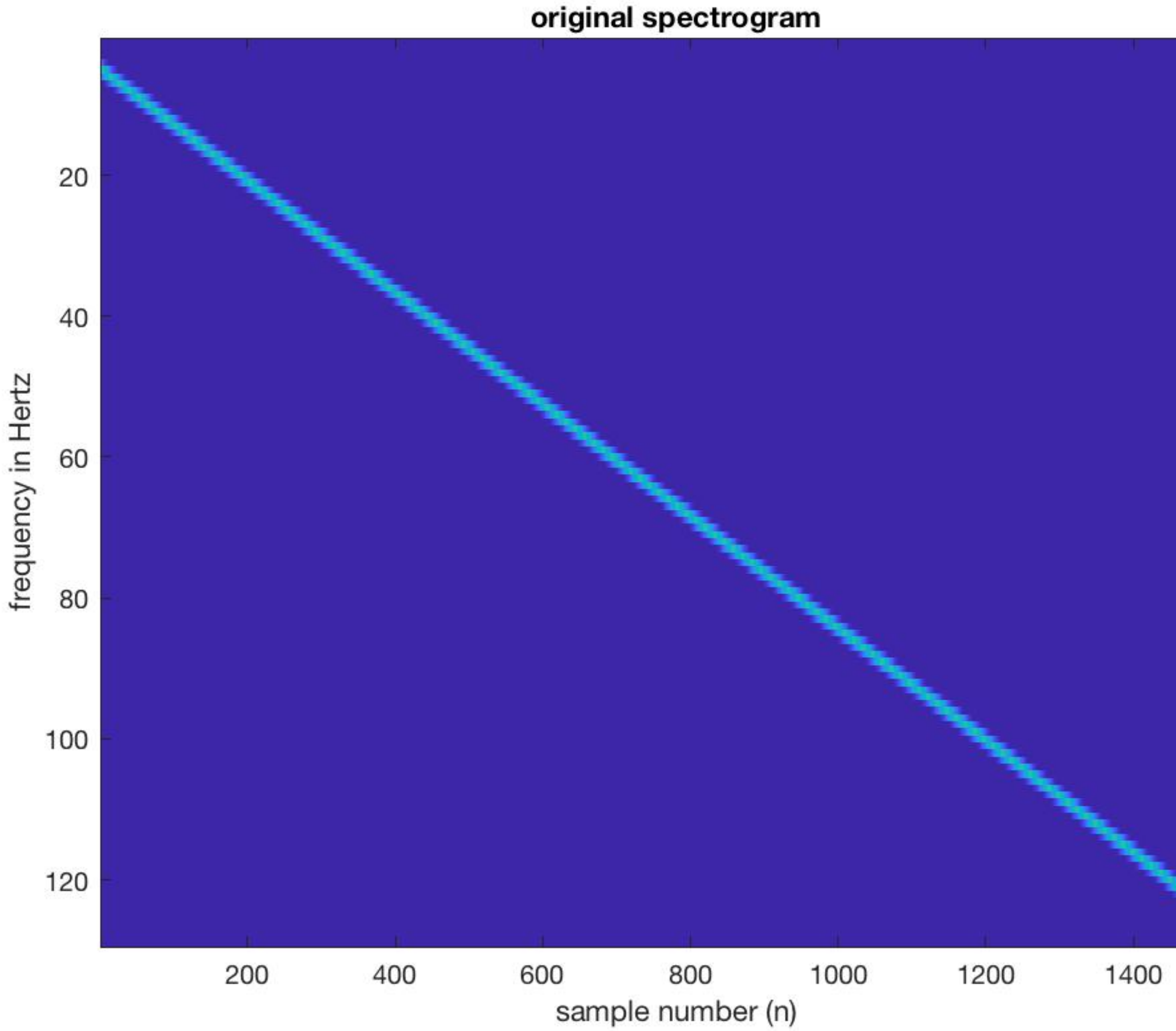
After filtering the correct frequencies seen from sig in frequency domain. We notice that there are 10 occurences of sig in the time domain of x.

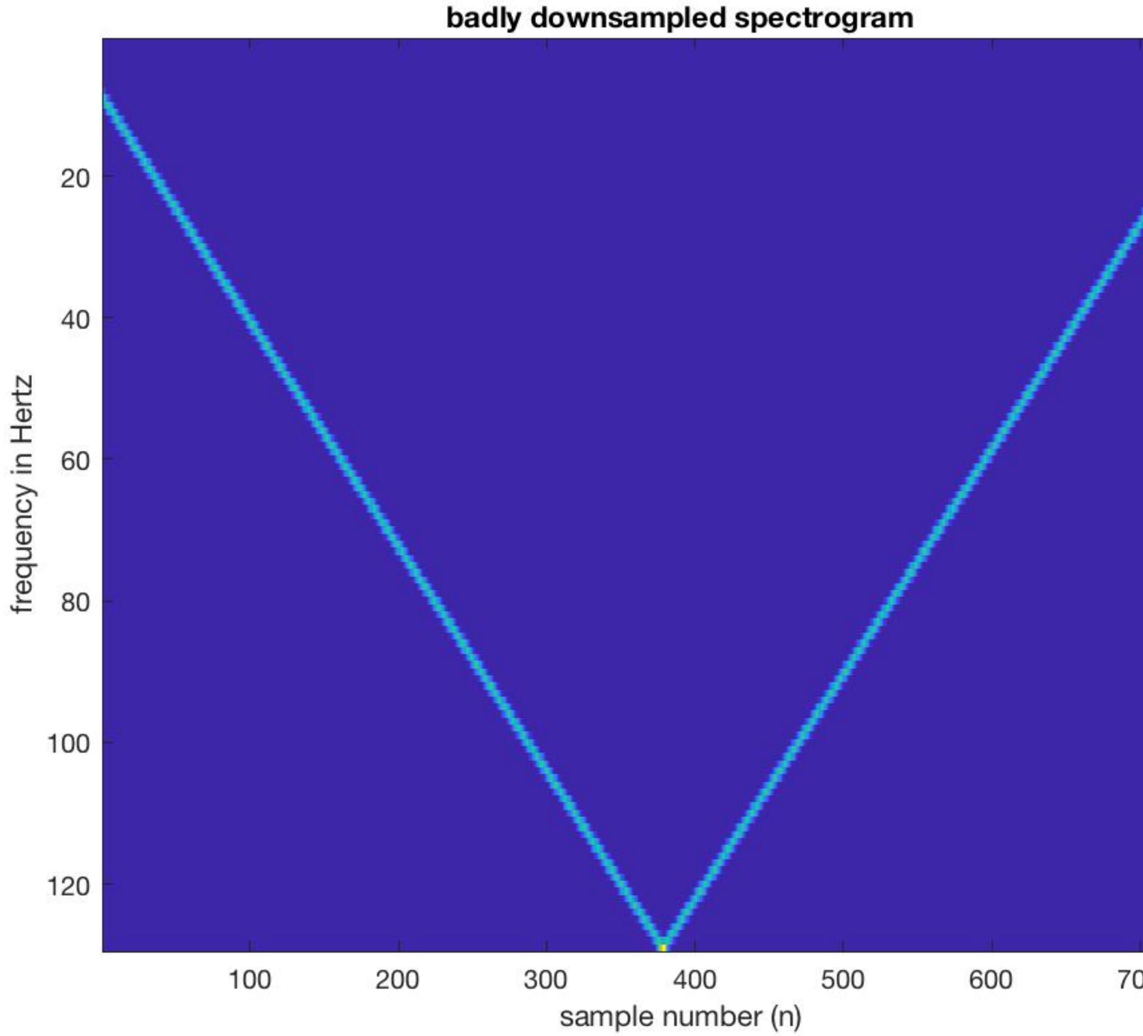
Blackdog

```
clear all;
clc;

load q2_signal.mat;

soundsc(x,fs);

s1 = spectrogram(x,hamming(256),128);
figure;
```

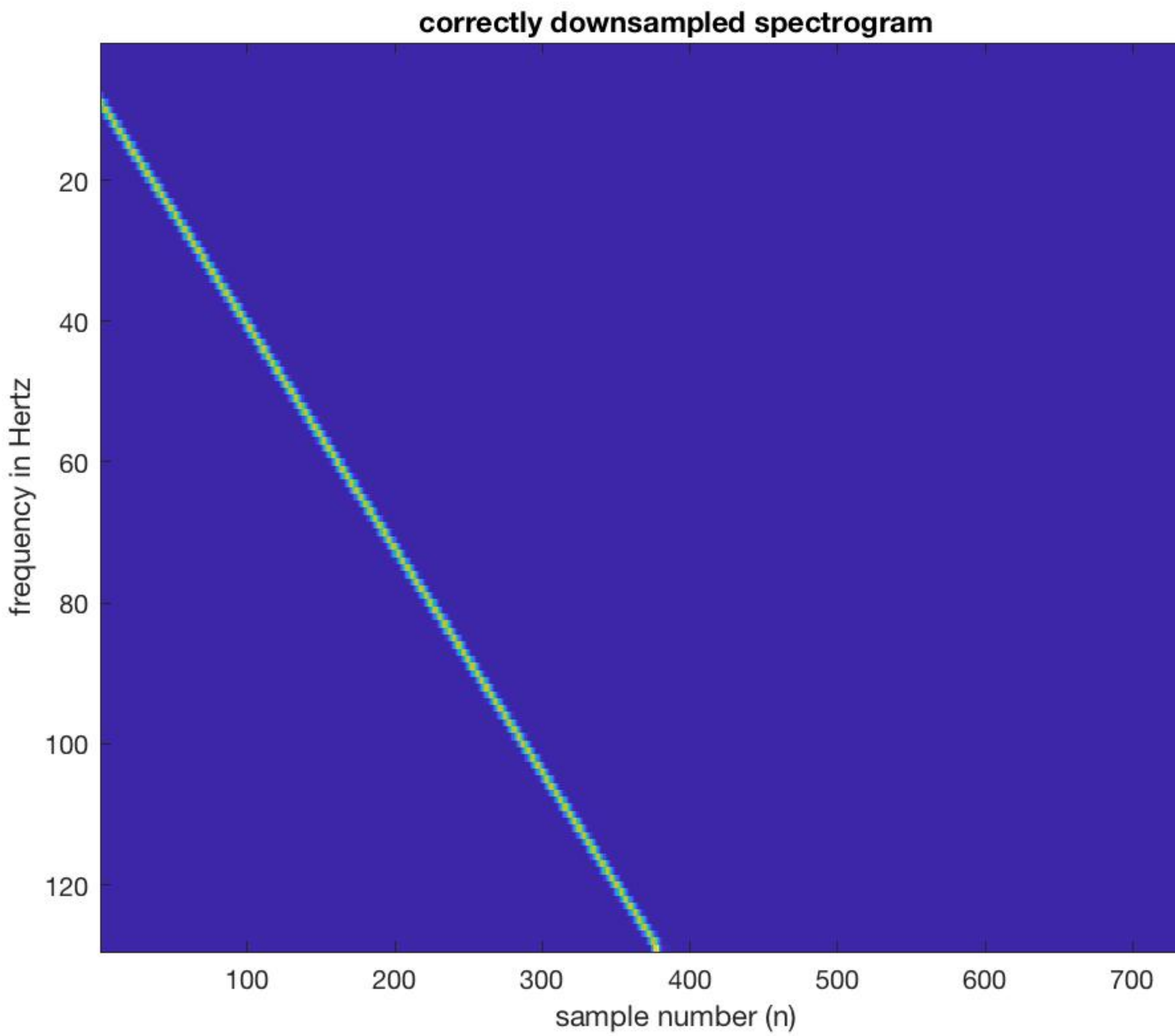
```
imagesc(abs(s1));
title('original spectrogram');
xlabel('sample number (n)');
ylabel('frequency in Hertz')

% end part 1
xodd = x(1:2:length(x));
soundsc(xodd,fs);
s2 = spectrogram(xodd,hamming(256),128);
figure;
imagesc(abs(s2));
title('badly downsampled spectrogram');
xlabel('sample number (n)');
ylabel('frequency in Hertz')
% end part 2

N = length(x);

x_w = fftshift(fft(x));

w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
for i = 1:length(x) %lpf x_w
    if(abs(w(i)) < pi/2)
        x_w(i) = x_w(i);
    else
        x_w(i) = 0;
    end

end
xright = ifft(ifftshift(x_w));
xright = downsample(xright,2);
soundsc(xright,fs);
s3 = spectrogram(xright,hamming(256),128);
figure;
imagesc(abs(s3));
title('correctly downsampled spectrogram');
xlabel('sample number (n)');
ylabel('frequency in Hertz')
```

Blackdog.m


original spectrogram
frequency in Hertz
20
40
60
80
100
120
200
400
600
800
1000
1200
1400
sample number (n)



badly downsampled spectrogram
frequency in Hertz
sample number (n)
20
40
60
80
100
120
100
200
300
400
500
600



After listening to the sound, we notice a linear increase in frequency that is seen by the spectrogram. The friend incorrectly downsampled because he did not low pass filter after downsampling which introduced copies into the mix. Therefore, we heard a shortened sound and a reverse version. After we lowpass filtered, We got the correct shortened version of the original signal.