

# ECE 311 Lab 7

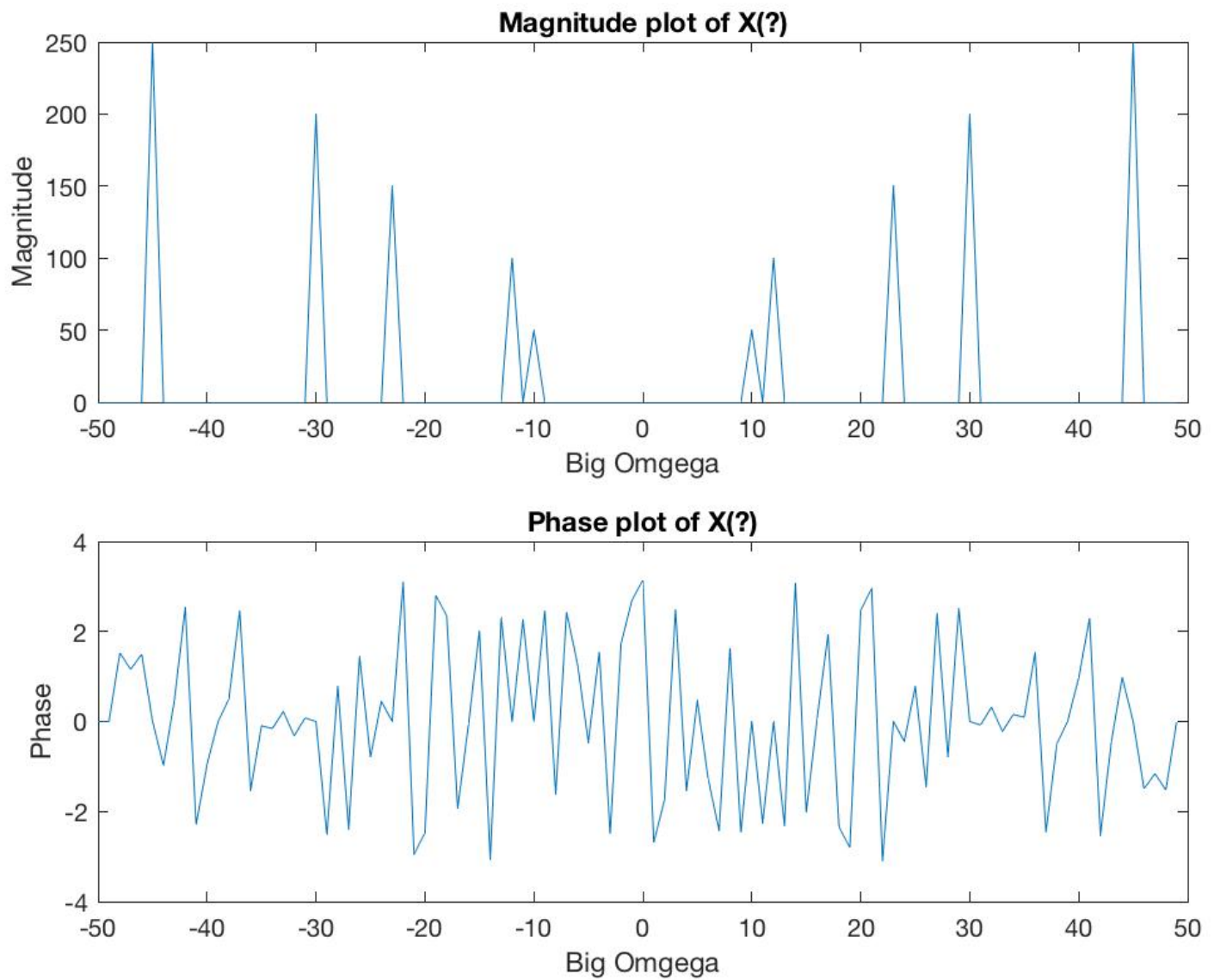
Jacob Hutter

May 7, 2017

Sakanaya

```
1 clear all;
2 clc;
3
4 load signal.mat;
5
6 N = length(x);
7
8 w = fftshift((0:N-1)/N*2*pi);
9 w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
10
11 % using w = Big w * T with T = 1/100, big omega = w * 100 / 2pi
12 w = w.*100/(2*pi);
13
14 x_w = fftshift(fft(x));
15
16 figure;
17 subplot(211);
18 plot(w,abs(x_w));
19 title('Magnitude plot of X(?)');
20 xlabel('Big Omega');
21 ylabel('Magnitude');
22
23 subplot(212);
24 plot(w,angle(x_w));
25 title('Phase plot of X(?)');
26 xlabel('Big Omega');
27 ylabel('Phase');
```

Sakanaya.m



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

## Chipotle

```
1 clear all;
2 clc;
3
4 load samplerate.mat;
5 fs = 40;
6 N = 40;
7
8 w = fftshift((0:N-1)/N*2*pi);
9 w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
10 x_w = fftshift(fft(x));
11 t = linspace(0,N-1,N) * 1/fs;
12 w = 40 * w/(2*pi);
13 figure;
14 subplot(211);
15 plot(w,abs(x_w));
16 title('fs = 40 magnitude plot');
17 xlabel('omega');
18 ylabel('magnitude');
19 subplot(212);
20 stem(t,x);
21 title('fs = 40 time plot');
22 xlabel('time');
23 ylabel('magnitude');
24
25 x_up = upsample(x,3);
26 x_up_w = fftshift(fft(x_up));
27 N = length(x_up);
28 t = linspace(0,N-1,N) .* 1/N;
29 w = fftshift((0:N-1)/N*2*pi);
30 w(1:N/2) = w(1:N/2) - 2*pi;
31 w = 120 * w/(2*pi);
32 figure;
33 subplot(211);
34
35 plot(w,abs(x_up_w));
36 title('fs = 120 magnitude plot');
37 xlabel('omega');
38 ylabel('magnitude');
39 subplot(212);
40 stem(t,x_up);
41 title('fs = 120 time plot');
42 xlabel('time');
43 ylabel('magnitude');
44
45 for i = 1:length(x_up)
46     if(abs(w(i)) > 25)
47         x_up_w(i) = 0;
48     end
49 end
50 figure;
51 subplot(211);
52 plot(w,abs(x_up_w));
53 title('fs = 120 filtered magnitude plot');
54 xlabel('omega');
55 ylabel('magnitude');
```

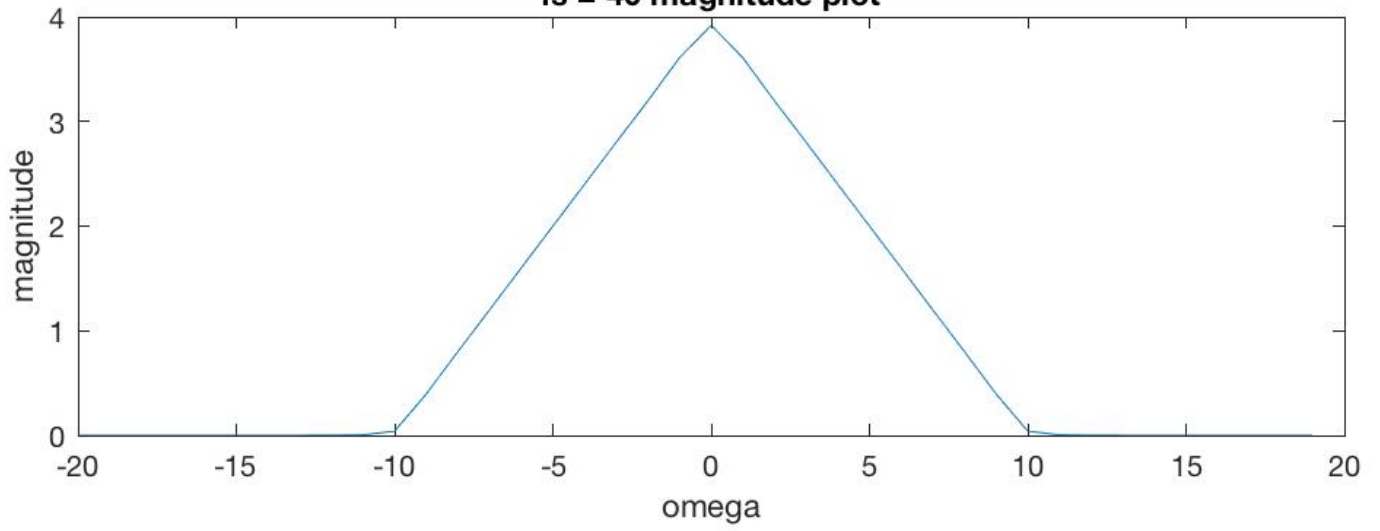
```

57 subplot(212);
   x_up = ifft(fftshift(x_up-w));
59 plot(t,abs(x_up));
   title('fs = 120 filtered time plot');
61 xlabel('time');
   ylabel('magnitude');
63
   x_down = downsample(x_up,2);
65 N = N/2;
   t = linspace(0,N-1,N)/N;
67 w = fftshift((0:N-1)/N*2*pi);
   w(1:N/2) = w(1:N/2) - 2*pi;
69 w = N * w/(2*pi);
   figure;
71 subplot(211);
   plot(t,abs(x_down));
73 title('fs = 60 filtered time plot');
   xlabel('time');
75 ylabel('magnitude');
   subplot(212);
77 plot(w,abs(fftshift(fft(x_down))));
   title('fs = 60 filtered magnitude plot');
79 xlabel('omega');
   ylabel('magnitude');

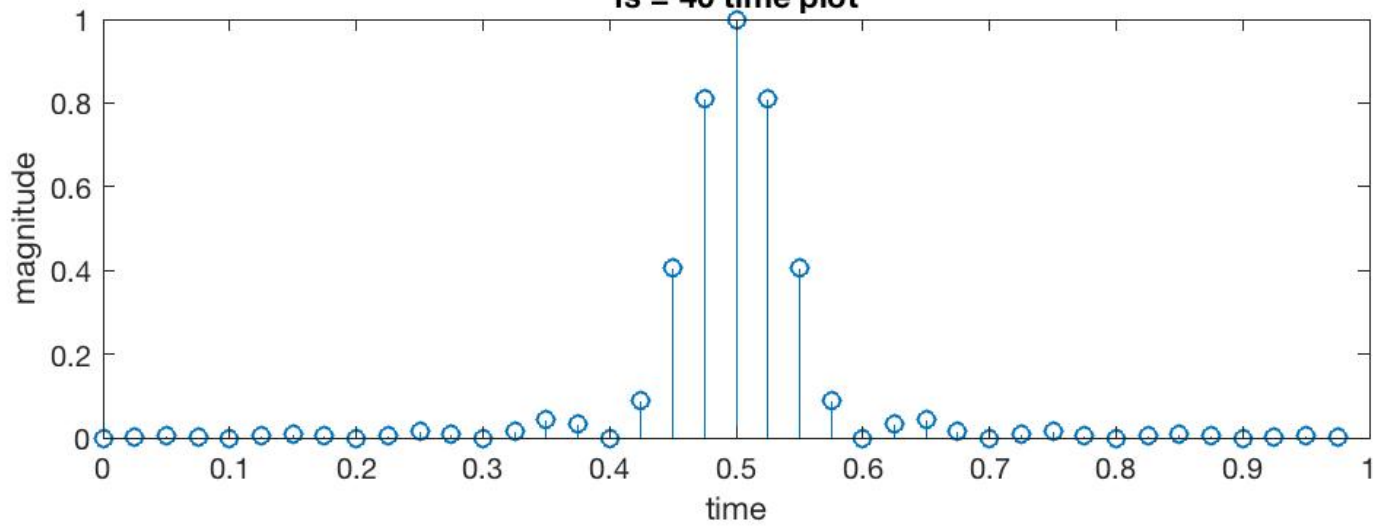
```

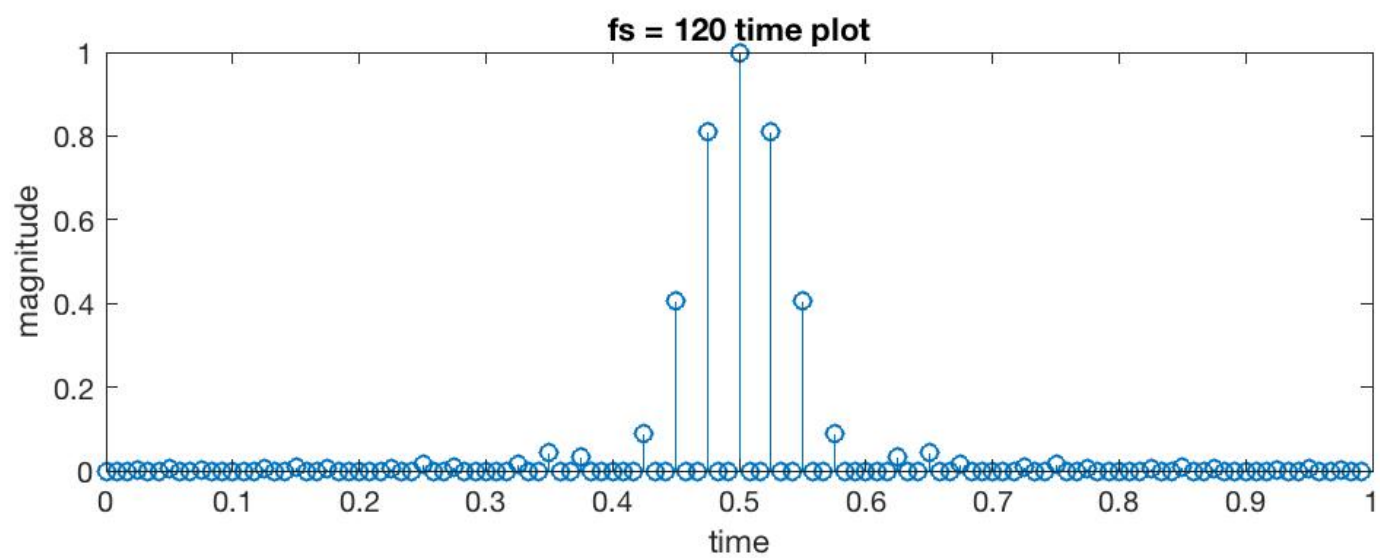
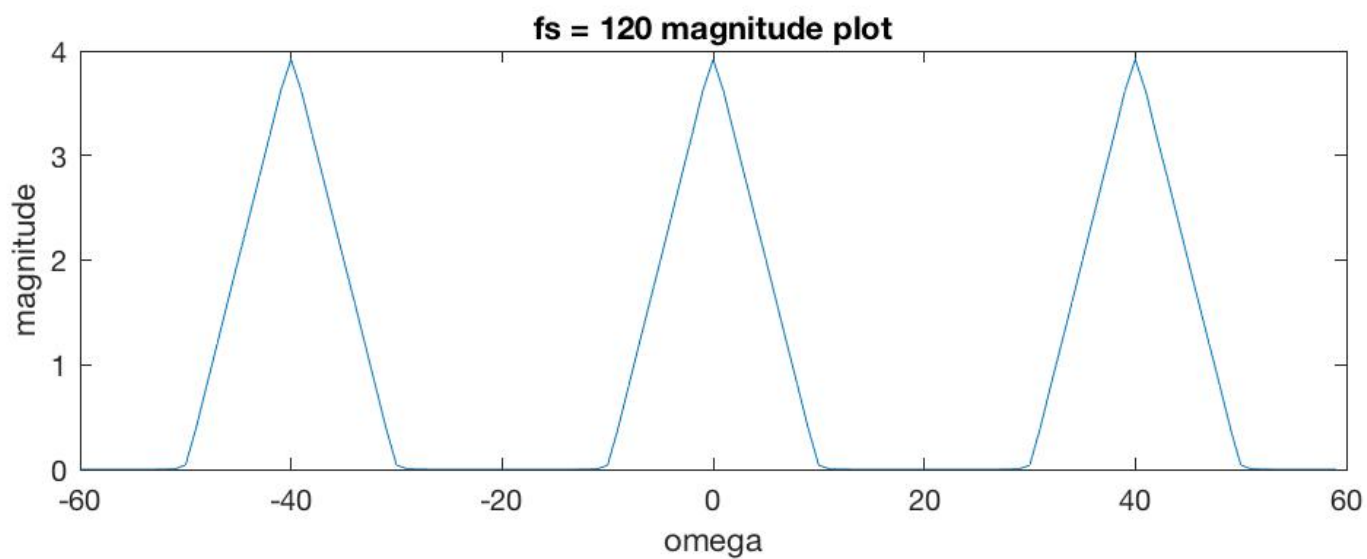
Chipotle.m

**fs = 40 magnitude plot**

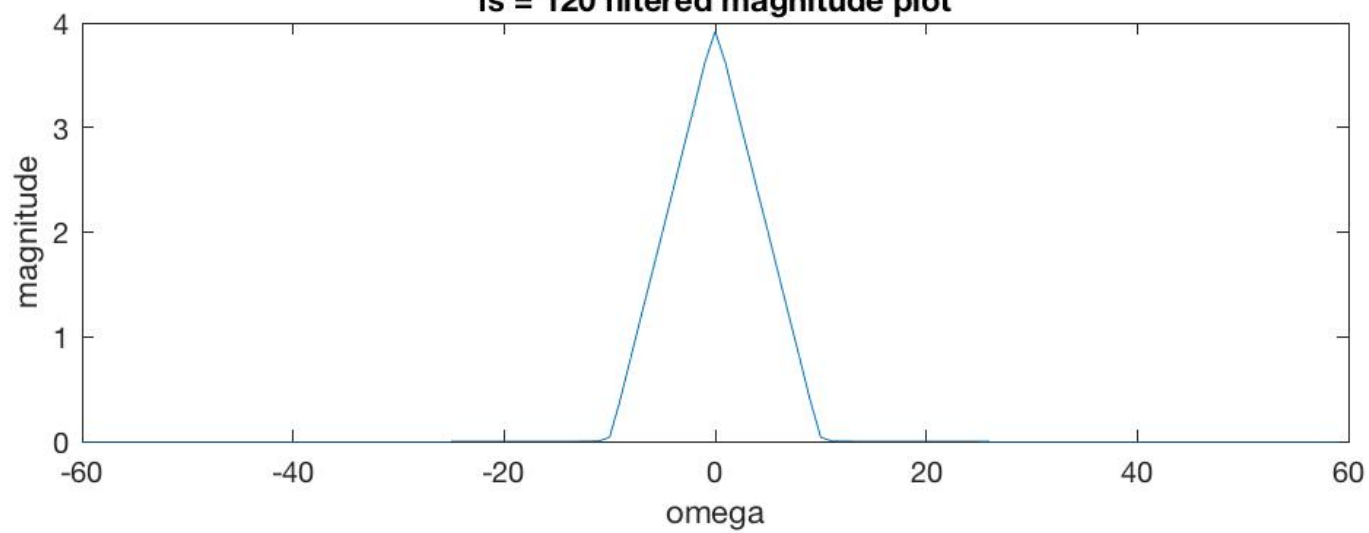


**fs = 40 time plot**

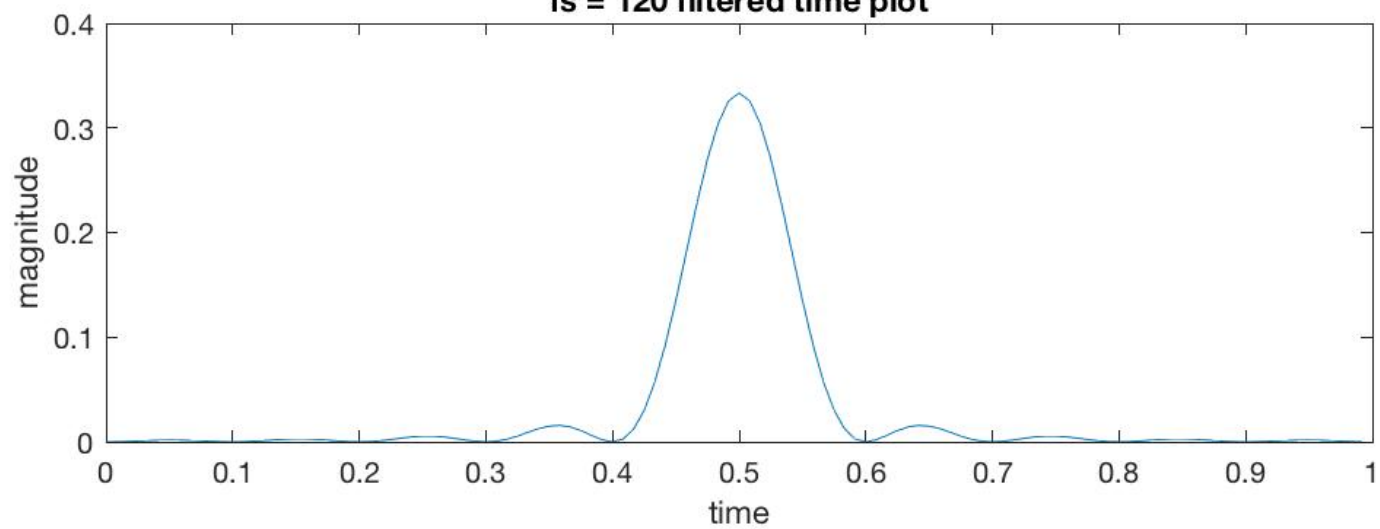


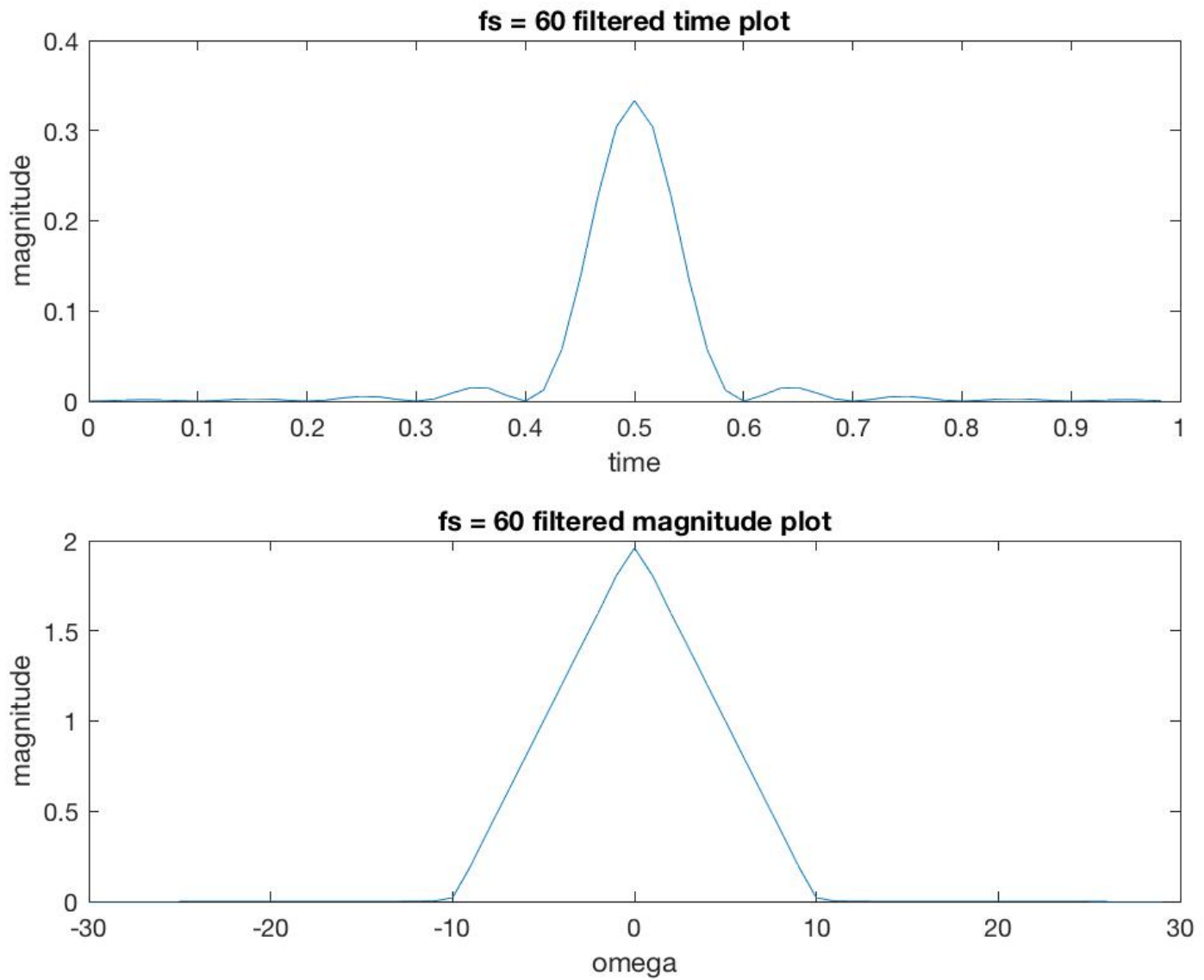


**fs = 120 filtered magnitude plot**



**fs = 120 filtered time plot**





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;
2 clc;

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

6 w = fftshift((0:N-1)/N*2*pi);
8 w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
w = w';
10 figure;
subplot(211);
12 x_w = fftshift(fft(x));
plot(w,abs(x_w));
14 title('Magnitude of X(w)');
xlabel('w');
16 ylabel('magnitude');
subplot(212);
18 plot(w,angle(x_w));
title('Phase of X(w)');
20 xlabel('w');
ylabel('phase');
22 %%%%%%%%%% end part 1 %%%%%%%%%%

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

40 %%%%%%%%%% end part 2 %%%%%%%%%%

42 % filter x
44 for i=1:length(x_w)
    if(w(i) < -2*pi/3 || w(i) > 2*pi/3)
46         x_w(i) = 0;
    end
48     if(abs(w(i)) < pi/3)
        x_w(i) = 0;
50     end
52 end

54 figure;
subplot(211);
56 plot(w,abs(x_w));
```

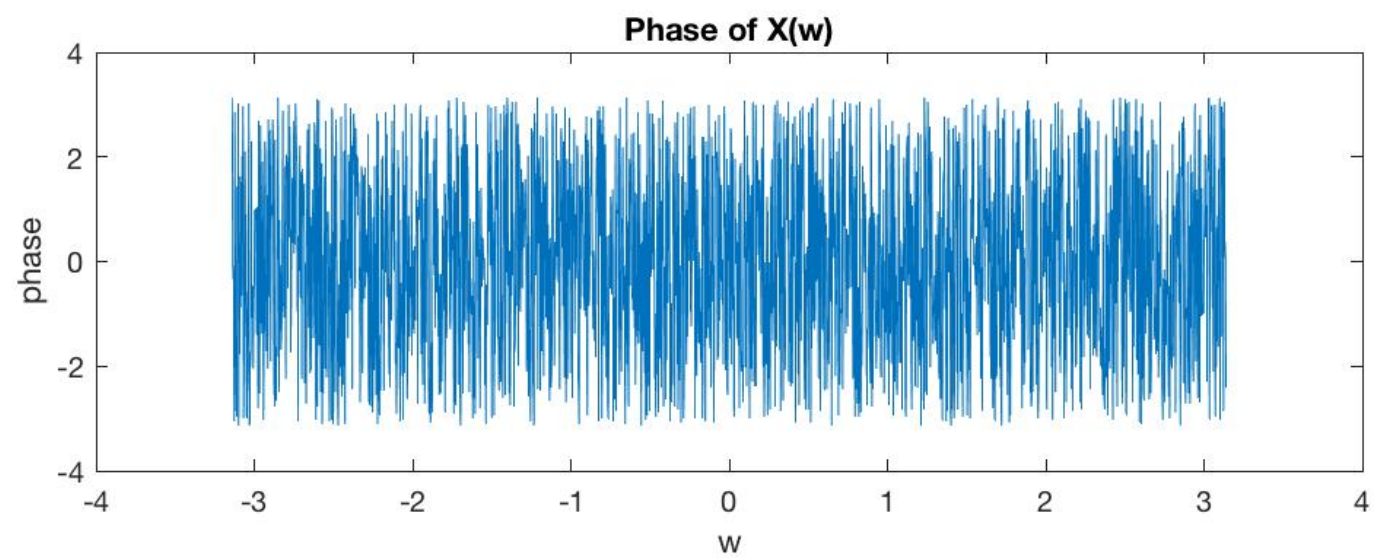
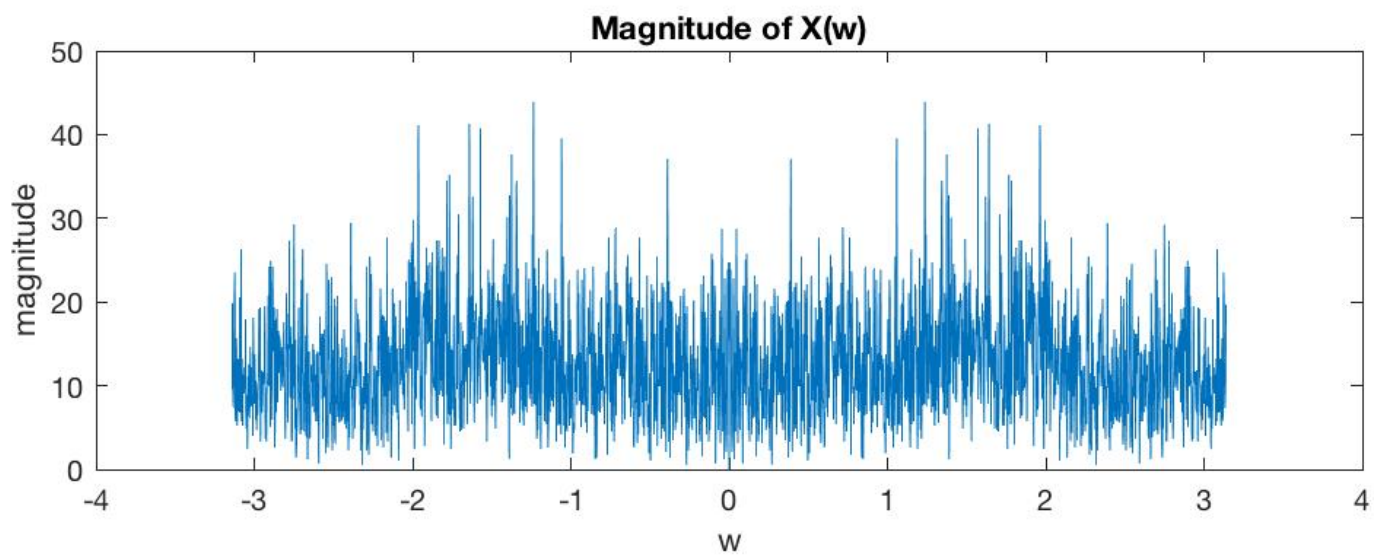
```

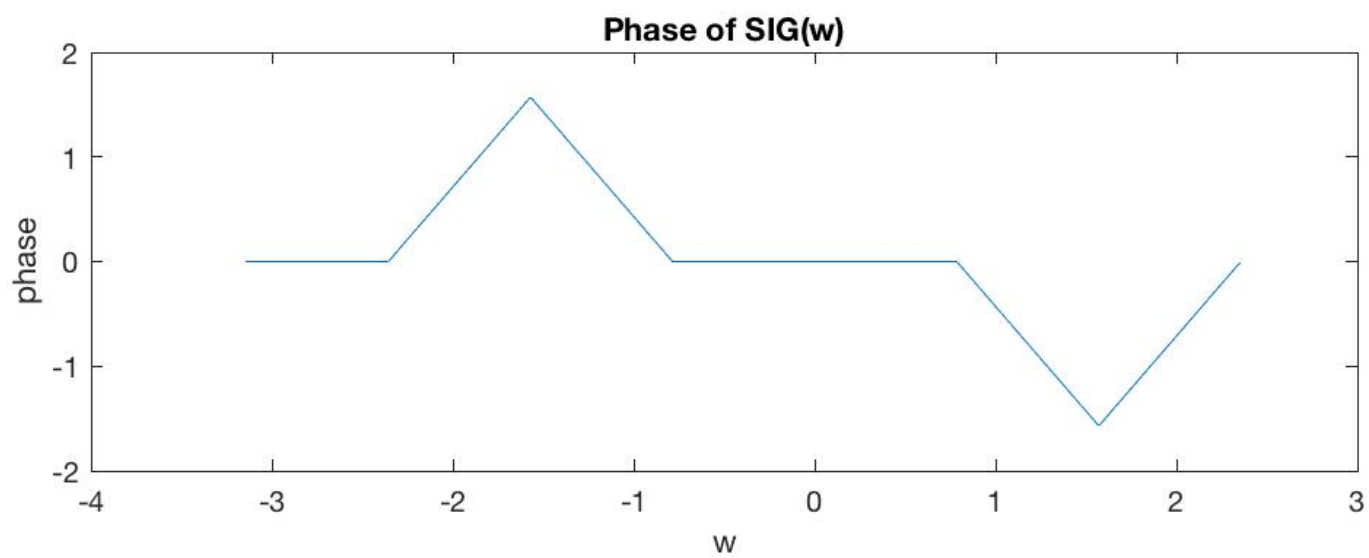
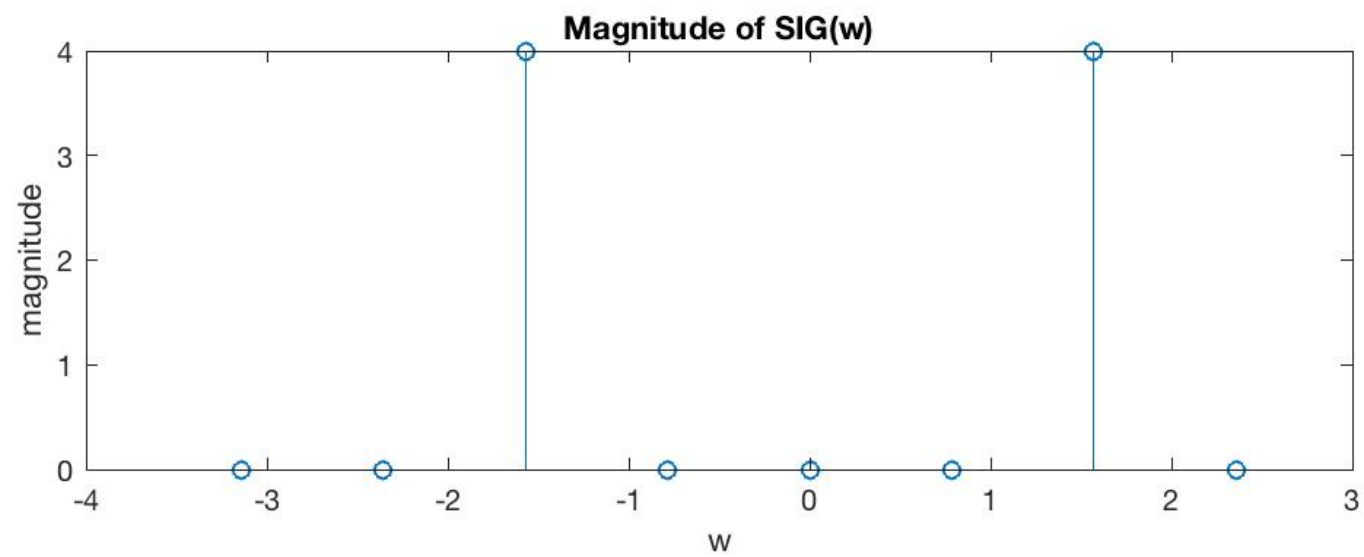
    title('Filtered magnitude X(w)');
58 xlabel('w');
   ylabel('phase');
60 subplot(212);
   plot(w, angle(x_w));
62 title('Filtered phase X(w)');
   xlabel('w');
64 ylabel('phase');

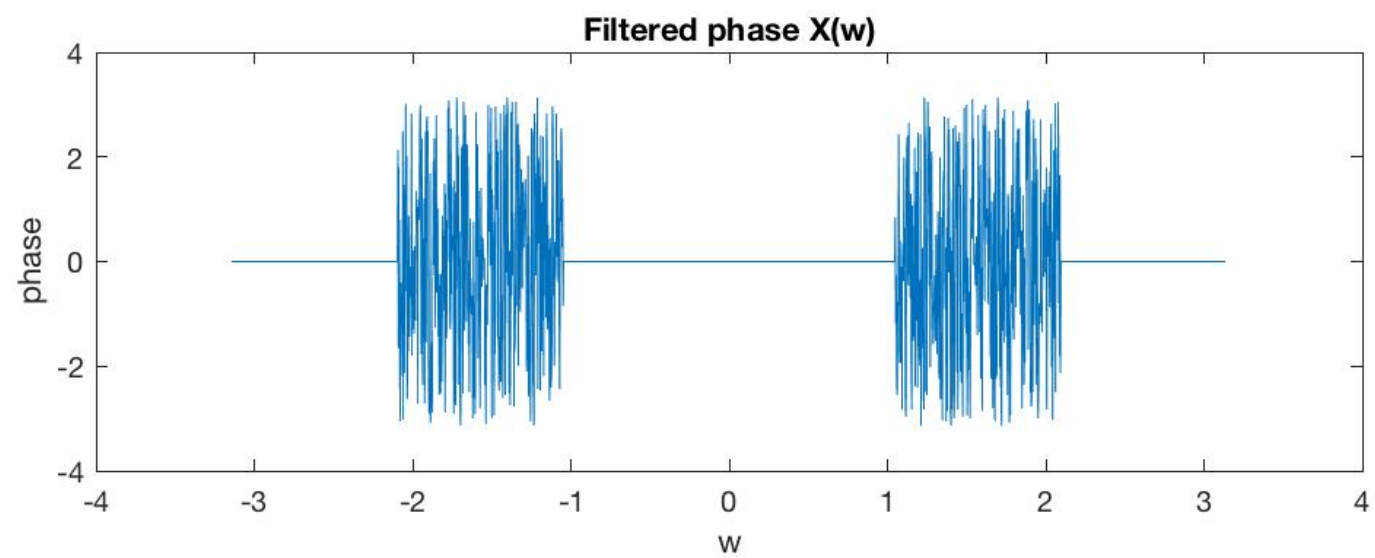
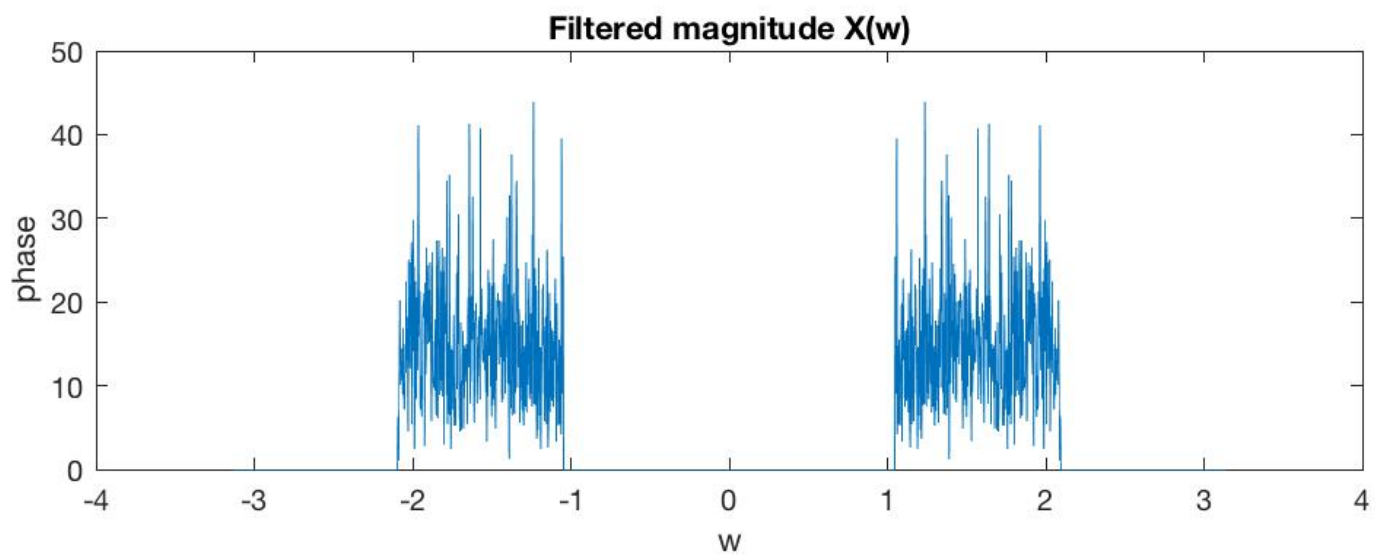
66
   x_new = ifft(fftshift(x_w));
68 figure;
   plot(x_new);
70 title('Filtered X(n)');
   xlabel('n');
72 ylabel('magnitude');

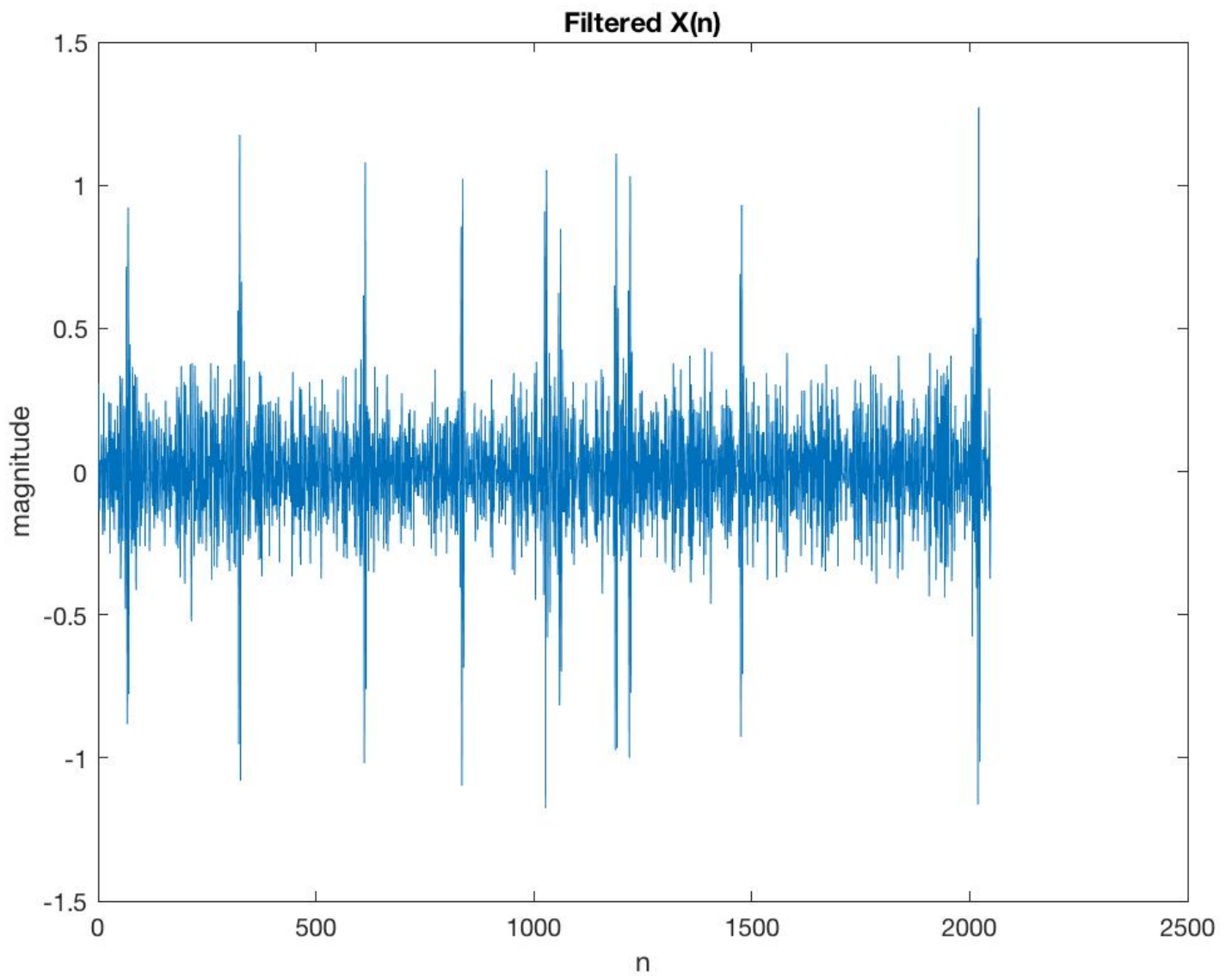
```

Legends.m







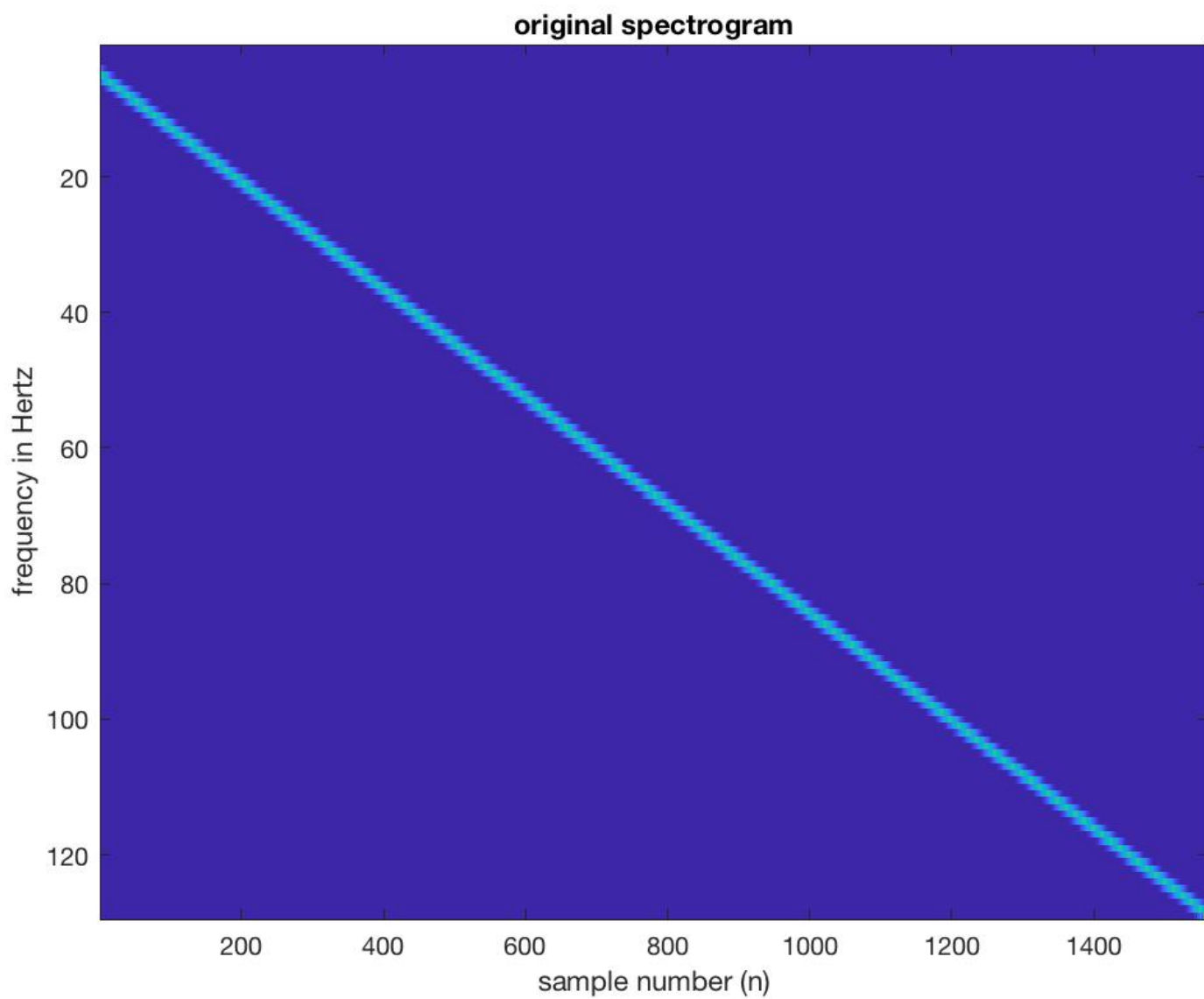


After filtering the correct frequencies seen from sig in frequency domain. We notice that there are 10 occurrences of sig in the time domain of x.

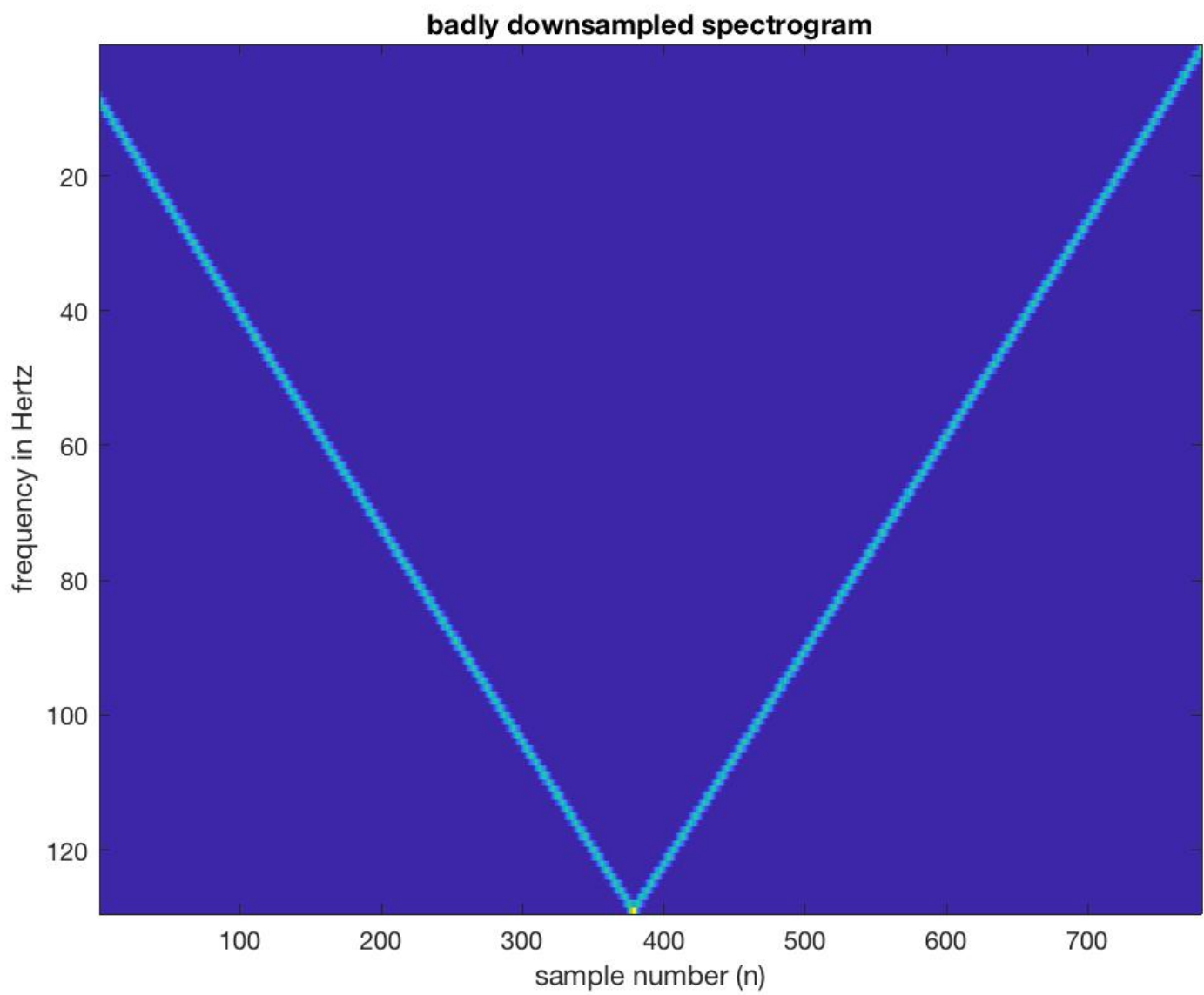
## Blackdog

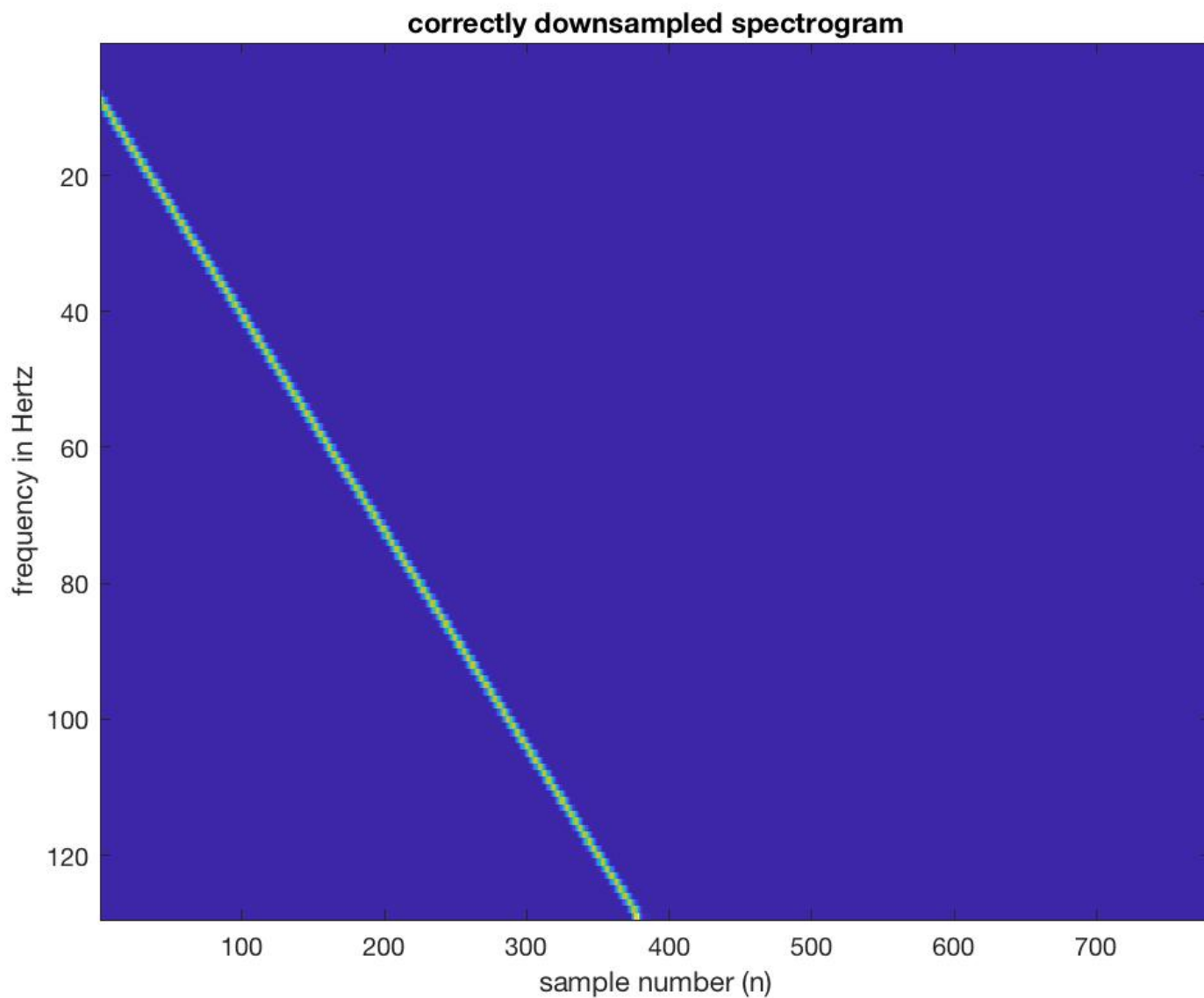
```
1 clear all;
2 clc;
3
4 load q2_signal.mat;
5
6 soundsc(x, fs);
7
8 s1 = spectrogram(x, hamming(256), 128);
9 figure;
10 imagesc(abs(s1));
11 title('original spectrogram');
12 xlabel('sample number (n)');
13 ylabel('frequency in Hertz')
14
15 % end part 1
16 xodd = x(1:2:length(x));
17 soundsc(xodd, fs);
18 s2 = spectrogram(xodd, hamming(256), 128);
19 figure;
20 imagesc(abs(s2));
21 title('badly downsampled spectrogram');
22 xlabel('sample number (n)');
23 ylabel('frequency in Hertz')
24 % end part 2
25
26 N = length(x);
27
28 x_w = fftshift(fft(x));
29
30 w = fftshift((0:N-1)/N*2*pi);
31 w(1:N/2) = w(1:N/2) - 2*pi; % get freq in radians
32 for i = 1:length(x) %lpf x_w
33     if(abs(w(i)) < pi/2)
34         x_w(i) = x_w(i);
35     else
36         x_w(i) = 0;
37     end
38
39 end
40 xright = ifft(ifftshift(x_w));
41 xright = downsample(xright, 2);
42 soundsc(xright, fs);
43 s3 = spectrogram(xright, hamming(256), 128);
44 figure;
45 imagesc(abs(s3));
46 title('correctly downsampled spectrogram');
47 xlabel('sample number (n)');
48 ylabel('frequency in Hertz')
```

Blackdog.m









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.