Table of Contents


```
% plotMrmRetLog.m
% This script prompts the user for a MRM-RET logfile of background and target
% reads, parses, and detection lists in the logfile
clear all; close all; clc
```

Query user for logfile

```
[fnmb,dnmb] = uigetfile('*.csv');
fprintf('Reading logfile1 %s\n',fullfile(dnmb,fnmb));
[cfgb,reqb,scnb,detb] = readMrmRetLog(fullfile(dnmb,fnmb));

[fnmt,dnmt] = uigetfile('*.csv');
fprintf('Reading logfile2 %s\n',fullfile(dnmt,fnmt));
[cfgt,reqt,scnt,dett] = readMrmRetLog(fullfile(dnmt,fnmt));
```

```
Reading logfile1 C:\Users\tonka\OneDrive\Documents\MATLAB\384_Lab\cw11_scans
\RetLog_Background004.csv
Reading logfile2 C:\Users\tonka\OneDrive\Documents\MATLAB\384_Lab\cw11_scans
\RetLog_Target001.csv
```

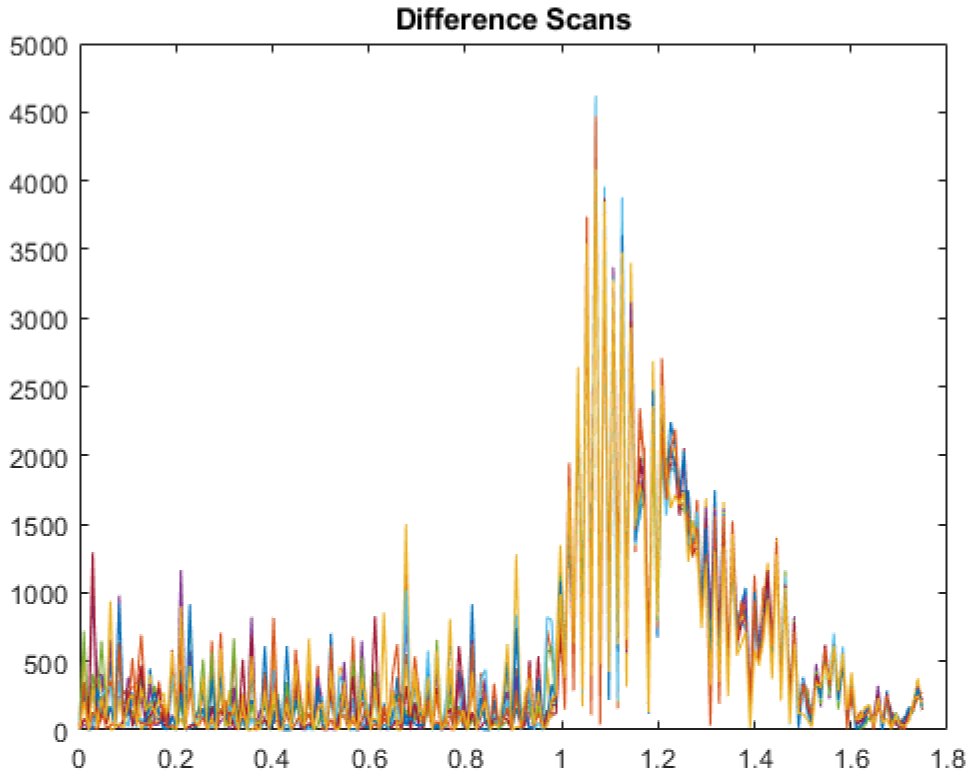
Pull out the raw scans

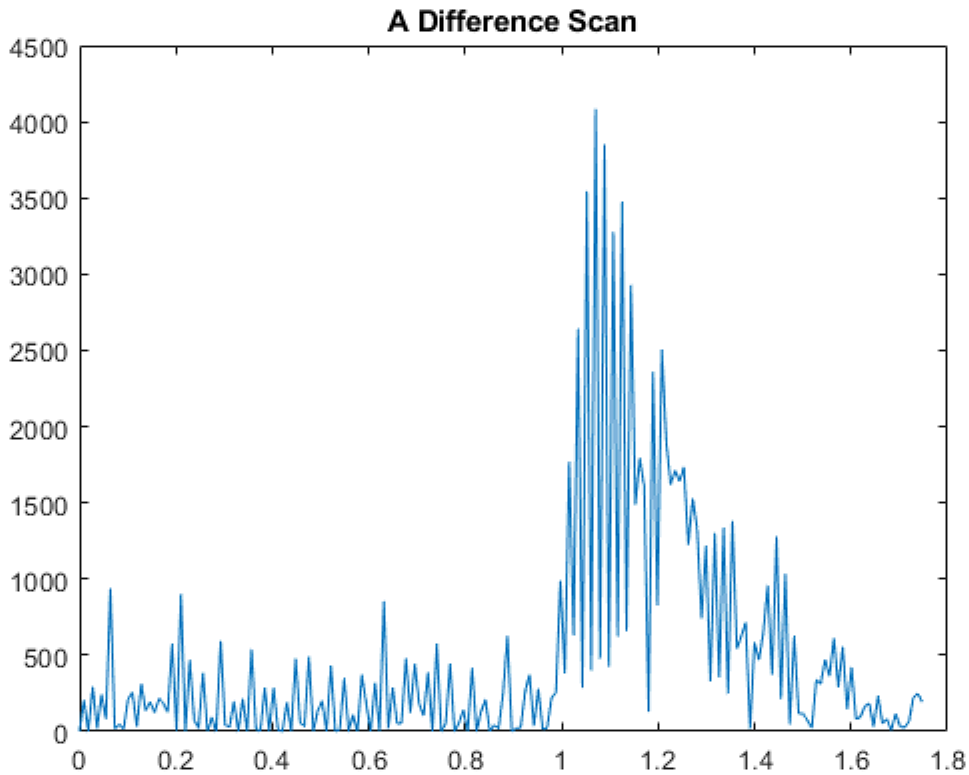
```
rawscansI = find([scnb.Nfilt] == 1);
rawscansV_background = reshape([scnb(rawscansI).scn],[],length(rawscansI))';

rawscansI1 = find([scnt.Nfilt] == 1);
```

```
rawscansV_target = reshape([scnt(rawscansI).scn],[],length(rawscansI))';

scan_difference = abs(rawscansV_background-rawscansV_target);
```

Create the horizontal and vertical axes

```
Tbin = 32/(512*1.024); % ns
T0 = 0; % ns
c = 0.29979; % m/ns
Rbin = c*(Tbin*(0:size(scan_difference(1,:),2)-1) - T0)/2; % Range Bins in
 meters

figure(1);plot(Rbin,scan_difference), title("Difference Scans")
figure(2);plot(Rbin,scan_difference(10,:)), title("A Difference Scan")
```



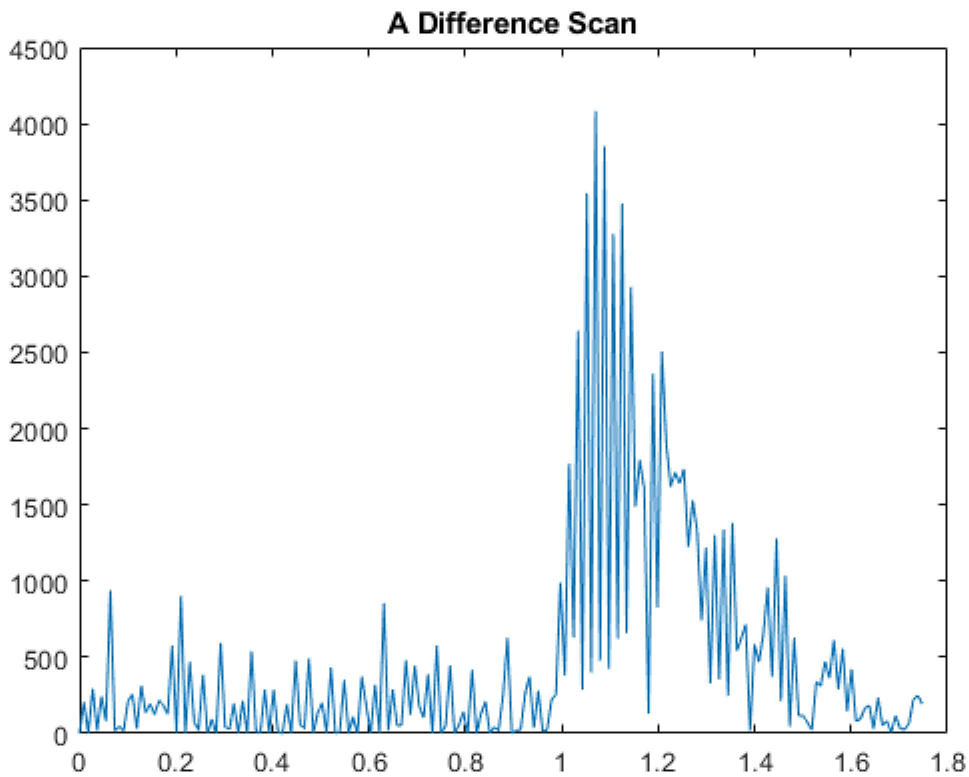


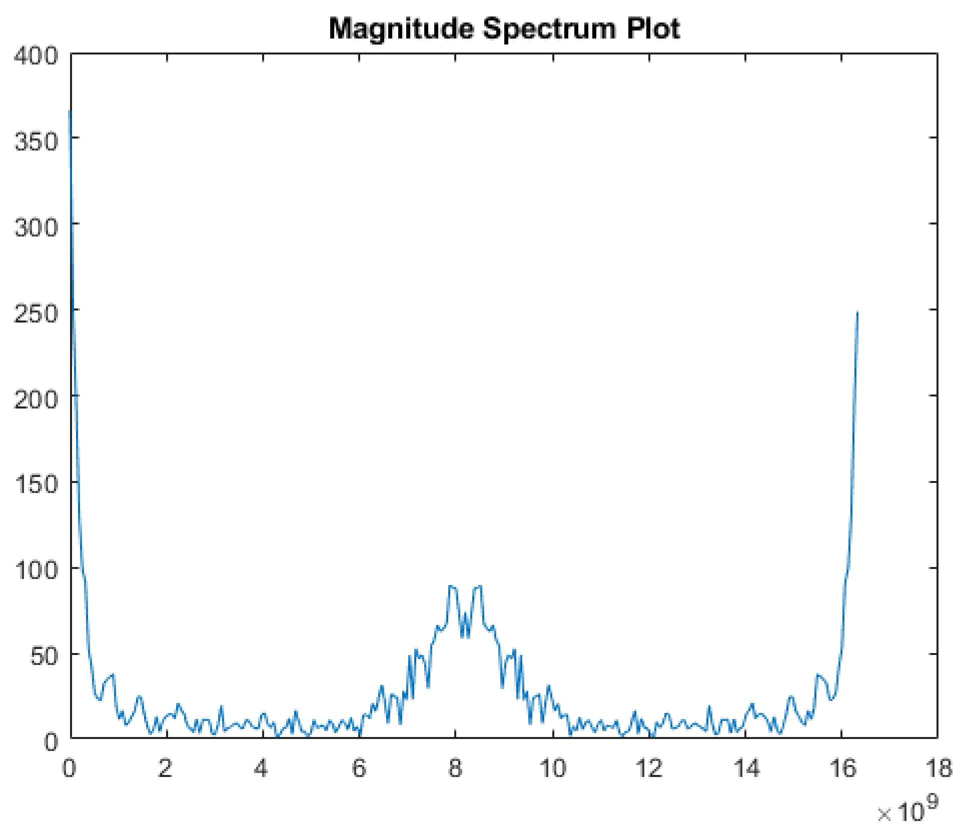
6. To plot the spectrum of the signal

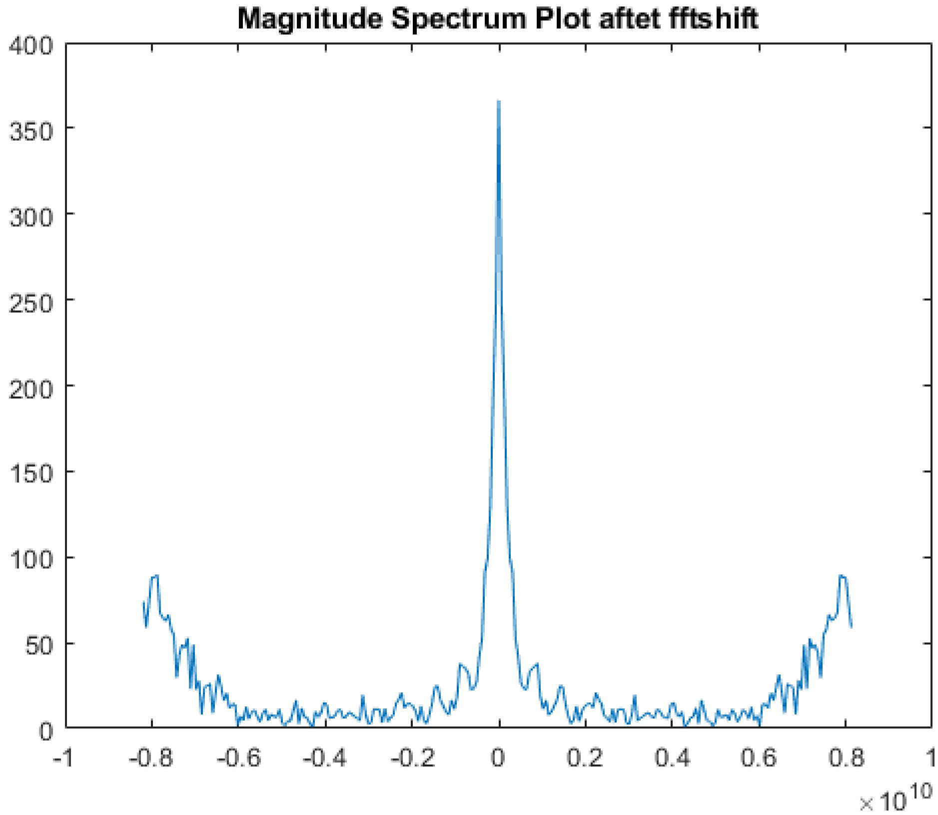
```
[a,i] = max(scan_difference(10,:));
distance = Rbin(i) % 6b
x = scan_difference(10,:);  % Sample Data
L = length(x);              % Window length
n = pow2(nextpow2(L));      % Next power of 2 from length of y
y_dft= fft(x,n);            % DFT
y_s = fftshift(y_dft);      % Rearrange y values
Ts = Tbin*1e-9;             % Tbin is given in nsec
fs = 1/Ts;
f = (-n/2:n/2-1)*(fs/n);    % 0-centered frequency range
fnz=(0:n-1)*(fs/n);         % Nonzero-centered frequency range
figure(3); plot(fnz,abs(y_dft)/n), title("Magnitude Spectrum Plot")  %6c(i)
figure(4); plot(f,abs(y_s)/n), title("Magnitude Spectrum Plot aftet fftshift")
      %6c(iii)
```

distance =

1.0704


A Difference Scan
4500
4000
3500
3000
2500
2000
1500
1000
500
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8

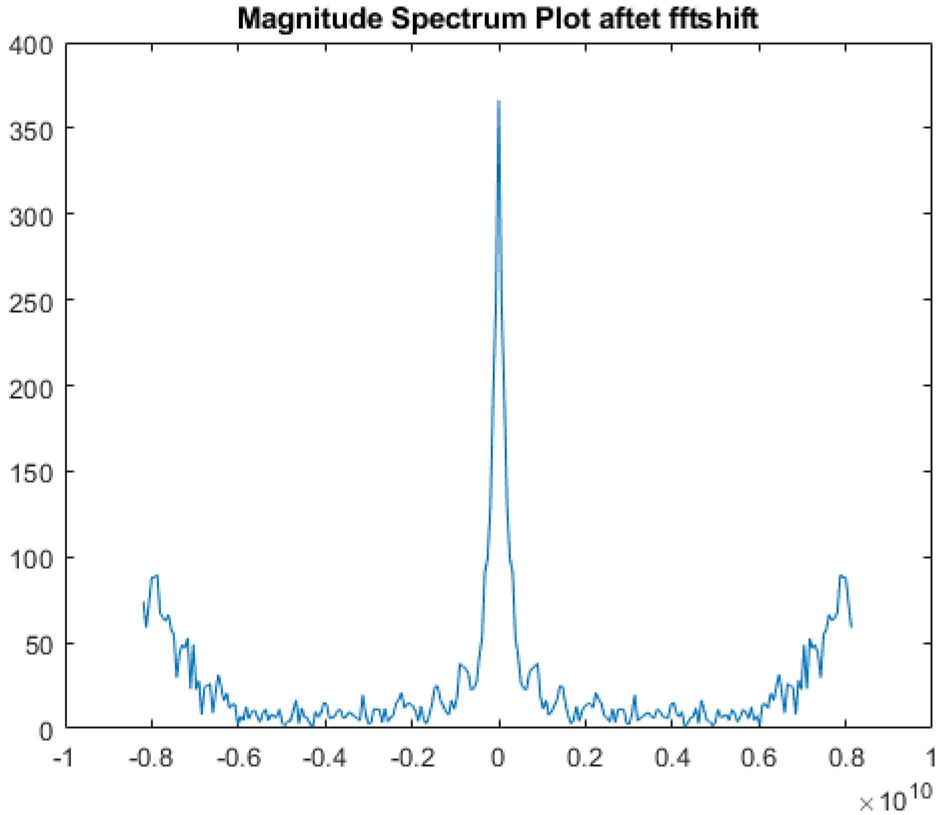

Magnitude Spectrum Plot
400
350
300
250
200
150
100
50
0
0
2
4
6
8
10
12
14
16
18
×10^9

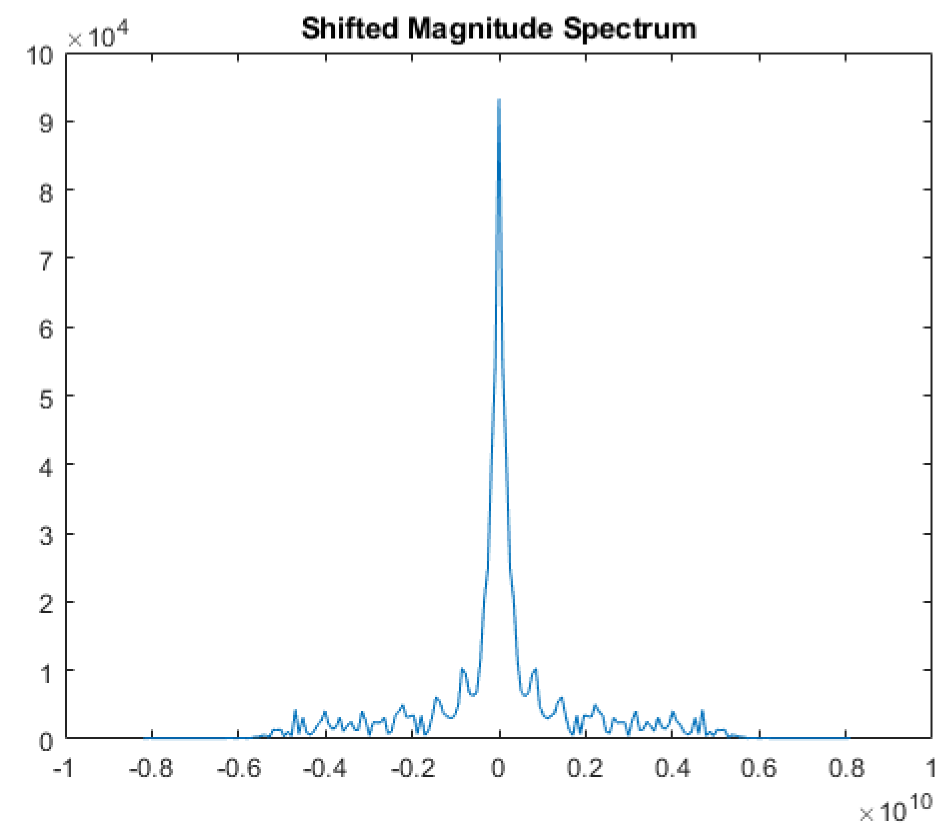


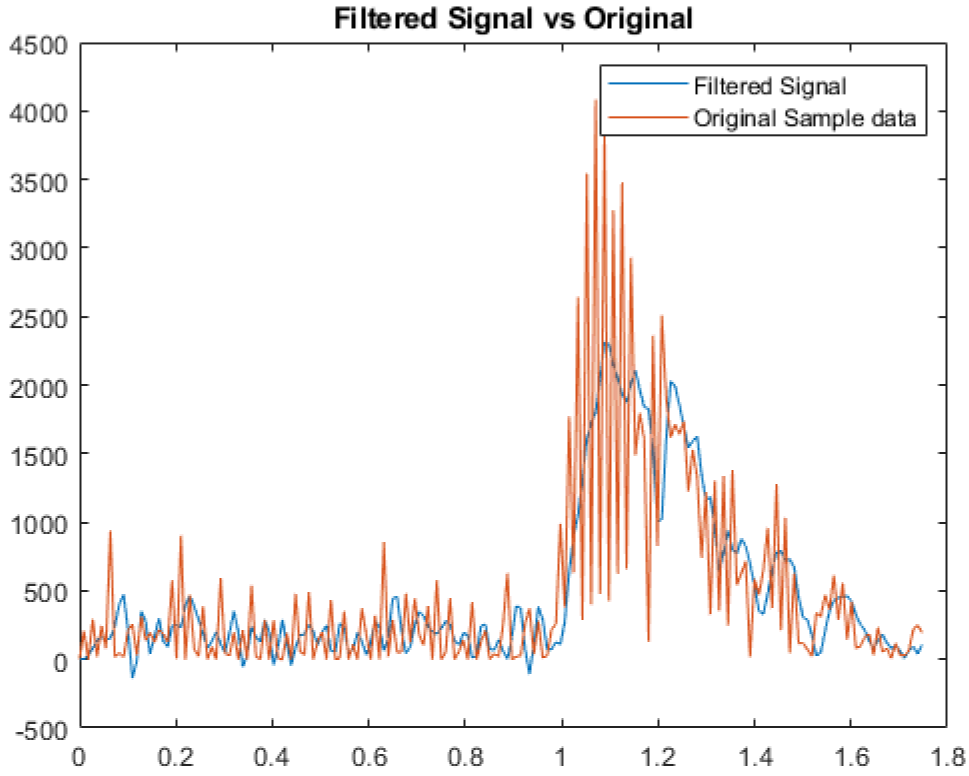
7. Butterworth Lowpass Filter Design

Drop-off is very gradual

```
fpass = 4.9e9; fstop = 5.6e9;
Wp = (2*fpass)/fs;
Ws = (2*fstop)/fs;
Rp = 1;
Rs = 20;
[n,Wn] = buttord(Wp,Ws,Rp,Rs);
[b,a] = butter(n,Wn,'low');
yf = filter(b,a,x);

%magnitude specturm
Y = fft(yf);
Y_shift = fftshift(Y);
L = length(Y_shift);
f = (-L/2 : L/2-1) * (fs/L);

figure(5); plot(f, abs(Y_shift)), title("Shifted Magnitude Spectrum");
figure(6); plot(Rbin,yf,Rbin,x), title("Filtered Signal vs Original"); % 7b
legend('Filtered Signal','Original Sample data')
```


Magnitude Spectrum Plot aftet fftshift
400
350
300
250
200
150
100
50
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
$\times 10^{10}$

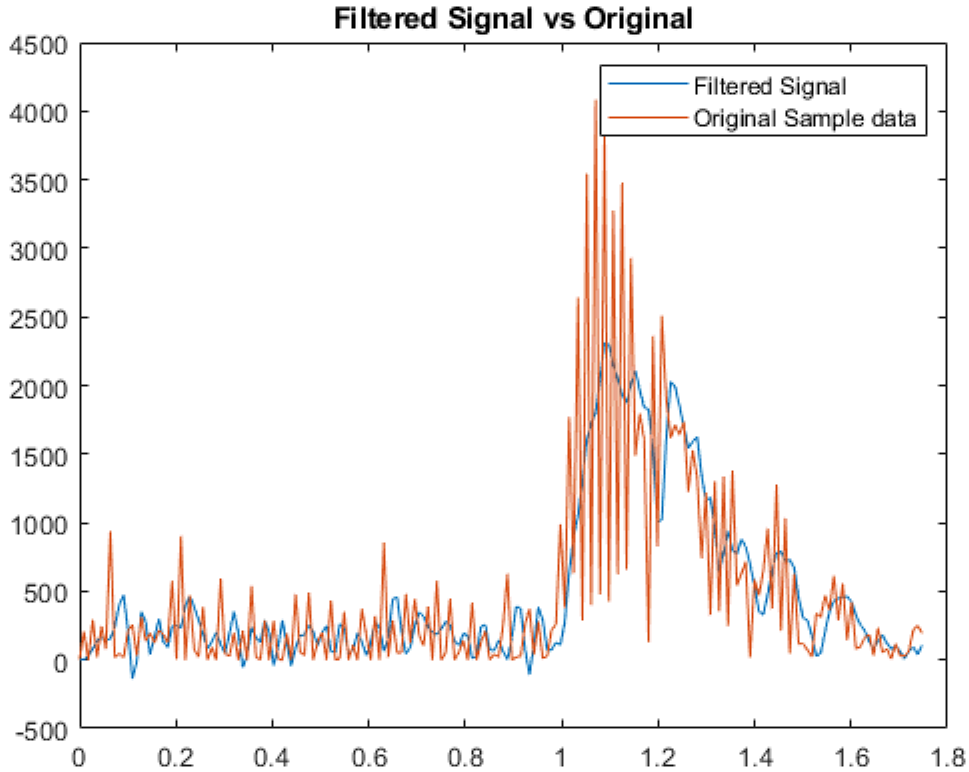

Shifted Magnitude Spectrum
$\times 10^{4}$
10
9
8
7
6
5
4
3
2
1
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
$\times 10^{10}$

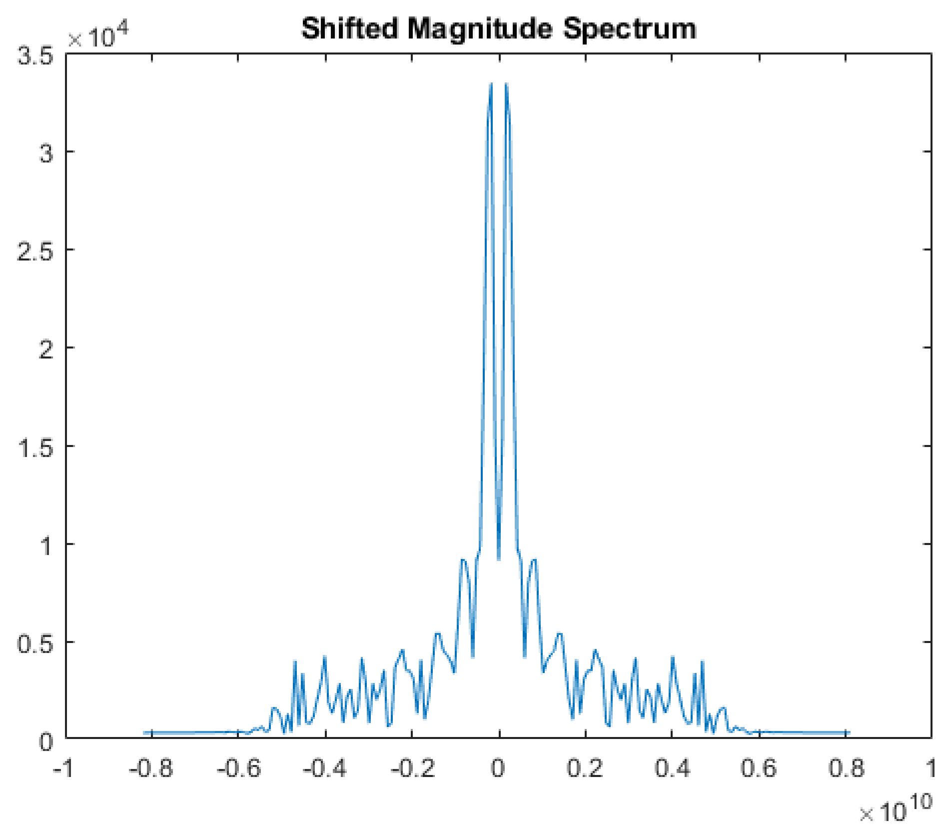


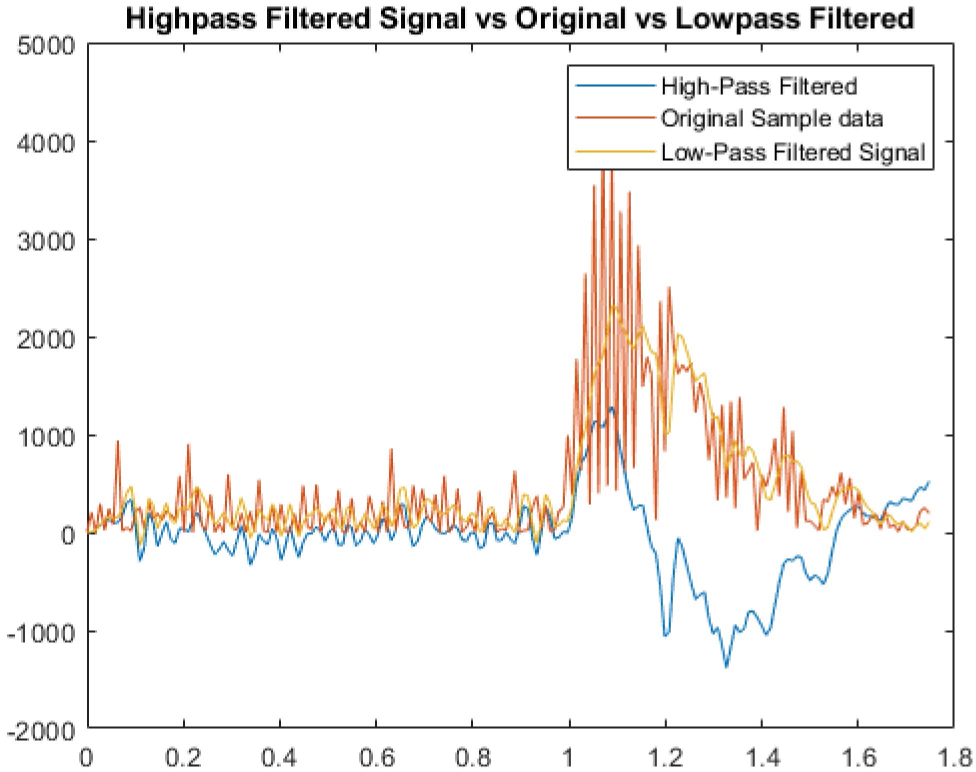
8. Butterworth Highpass Filter Design

Drop-off is very gradual

```
fpass = .1e9; fstop = .05e9;
Wp = (2*fpass)/fs;
Ws = (2*fstop)/fs;
Rp = 1;
Rs = 20;
[n,Wn] = buttord(Wp,Ws,Rp,Rs);
[b,a] = butter(n,Wn,'high');
yfh = filter(b,a,yf);

%magnitude specturm
Y = fft(yfh);
Y_shift = fftshift(Y);
L = length(Y_shift);
f = (-L/2 : L/2-1) * (fs/L);

figure(7); plot(f, abs(Y_shift)), title("Shifted Magnitude Spectrum");
figure(8); plot(Rbin,yfh,Rbin,x,Rbin,yf), title("Highpass Filtered Signal vs
 Original vs Lowpass Filtered");
legend('High-Pass Filtered','Original Sample data','Low-Pass Filtered Signal')
```

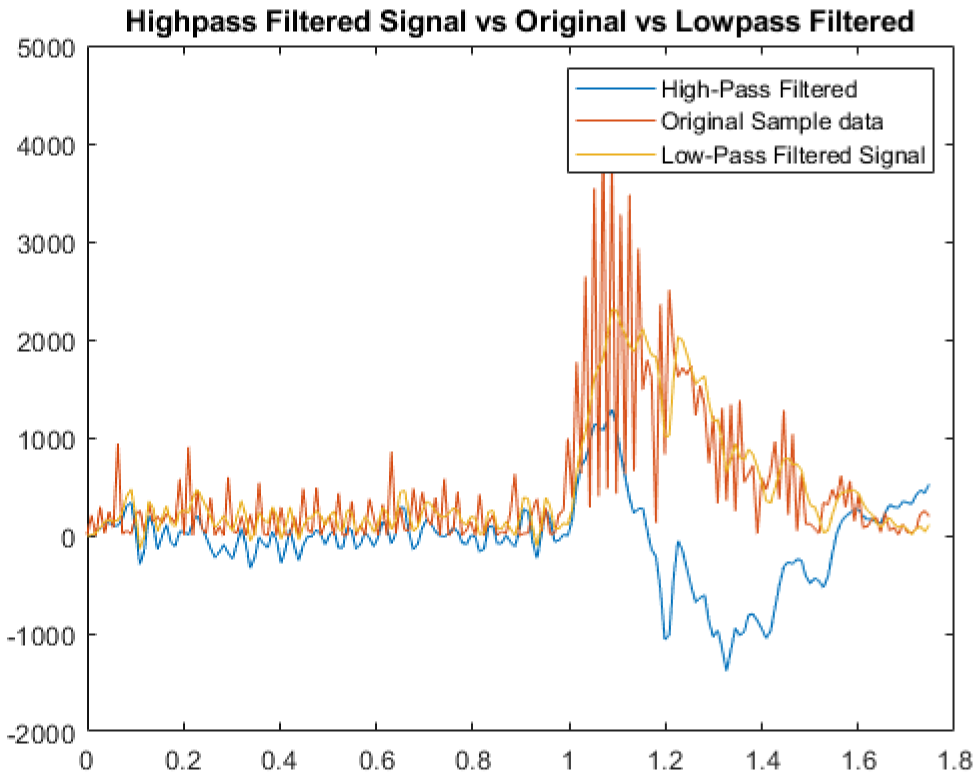

Filtered Signal vs Original
Filtered Signal
Original Sample data
4500
4000
3500
3000
2500
2000
1500
1000
500
0
-500
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8

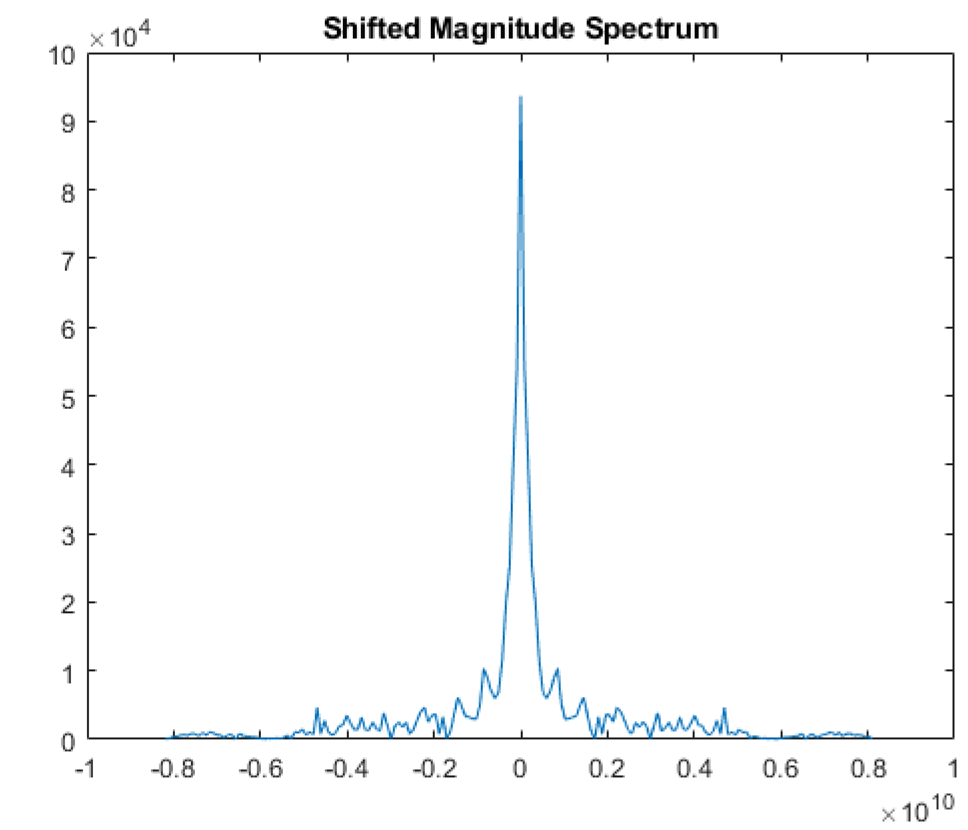

Shifted Magnitude Spectrum
×10^4
3.5
3
2.5
2
1.5
1
0.5
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
×10^10

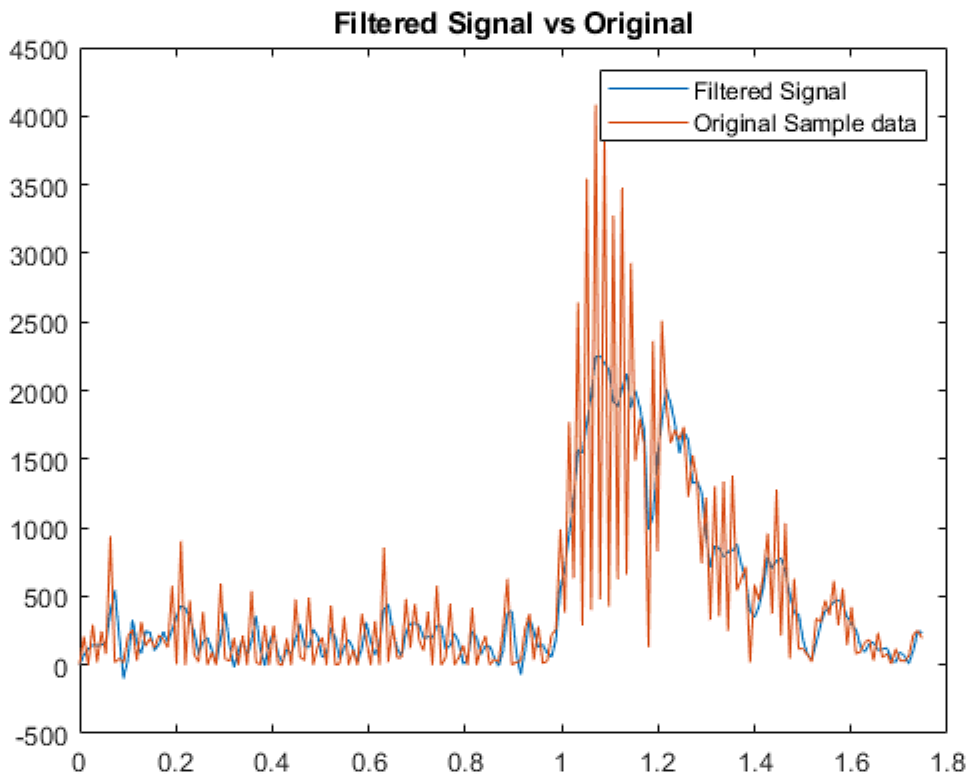


9a Elliptic Low Pass

```
fpass = 4.9e9; fstop = 5.6e9;
Wp = (2*fpass)/fs;
Ws = (2*fstop)/fs;
Rp = 1;
Rs = 20;
[n,Wn] = ellipord(Wp,Ws,Rp,Rs);
[b,a] = ellip(n,Rp, Rs,Wn, 'low');
yf = filter(b,a,x);
%magnitude specturm
Y = fft(yf);
Y_shift = fftshift(Y);
L = length(Y_shift);
f = (-L/2 : L/2-1) * (fs/L);

figure(9); plot(f, abs(Y_shift)), title("Shifted Magnitude Spectrum");
figure(10); plot(Rbin,yf,Rbin,x), title("Filtered Signal vs Original"); % 9a
legend('Filtered Signal','Original Sample data')
```

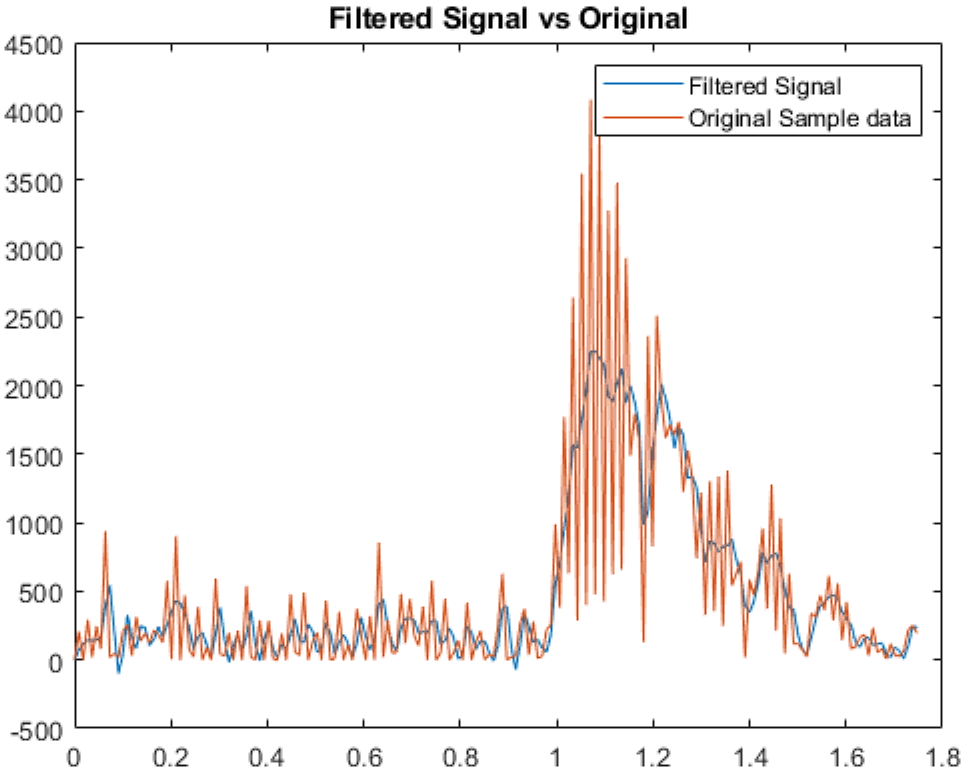

Highpass Filtered Signal vs Original vs Lowpass Filtered
High-Pass Filtered
Original Sample data
Low-Pass Filtered Signal
5000
4000
3000
2000
1000
0
-1000
-2000
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8

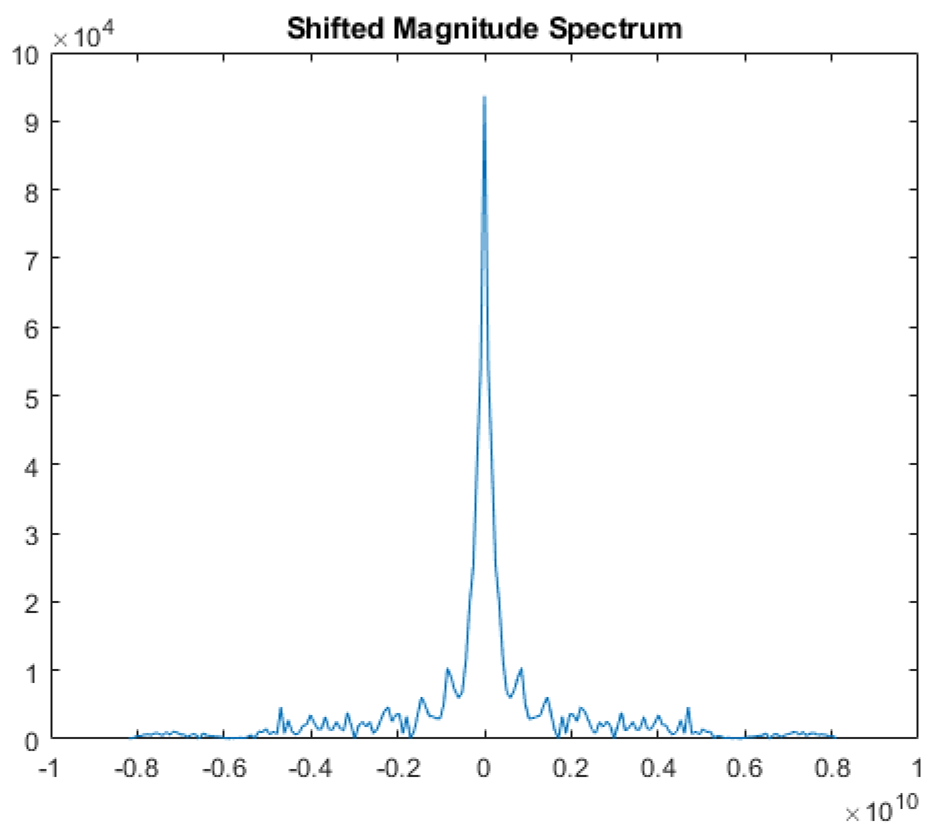

Shifted Magnitude Spectrum
×10^4
10
9
8
7
6
5
4
3
2
1
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
×10^10

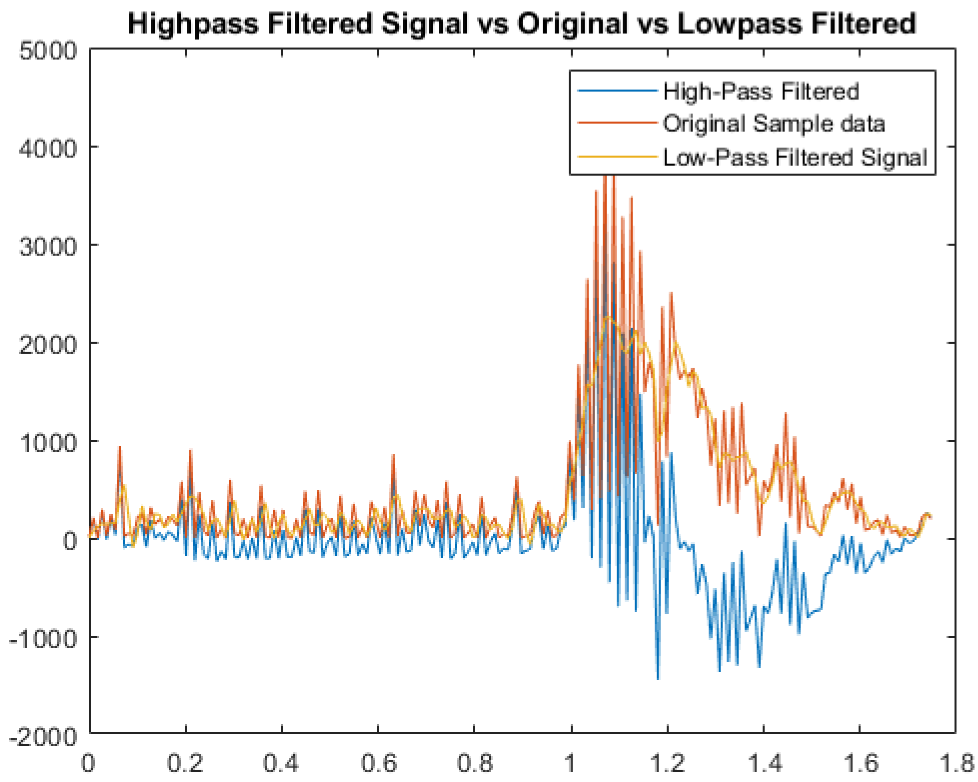


9b Elliptic High Pass

```
fpass = .1e9; fstop = .05e9;
Wp = (2*fpass)/fs;
Ws = (2*fstop)/fs;
Rp = 1;
Rs = 20;
[n,Wn] = ellipord(Wp, Ws, Rp, Rs);
[b,a] = ellip(n, Rp, Rs, Wn,'high');
yfh = filter(b, a, x);

%magnitude specturm
Y = fft(yf);
Y_shift = fftshift(Y);
L = length(Y_shift);
f = (-L/2 : L/2-1) * (fs/L);

figure(11); plot(f, abs(Y_shift)), title("Shifted Magnitude Spectrum");
figure(12); plot(Rbin,yfh,Rbin,x,Rbin,yf), title("Highpass Filtered Signal vs
 Original vs Lowpass Filtered"); % 9b
legend('High-Pass Filtered','Original Sample data','Low-Pass Filtered Signal')
```


Filtered Signal vs Original
Filtered Signal
Original Sample data
4500
4000
3500
3000
2500
2000
1500
1000
500
0
-500
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8


Shifted Magnitude Spectrum
$\times 10^4$
10
9
8
7
6
5
4
3
2
1
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
$\times 10^{10}$



Questions

6.c.i

The range of frequencies presents looks to be about 16.39 GHz to fs/2 (8.1920e9)

7b & c

The shifted magnitude specturm has a smaller range of frequences than the one in 6.c.iii and has a larger magnitude at 0 Hz. The filter attenuated the higher frequencies (ones above 20 dB)

8

Compared to 6.c.iii this magnitude spectrum has two spikes instead of 1. As far as range goes they both stop around the same frequency which is likely due to the fact that the high pass signal filtered was the filtered low pass signal. Comparing all three signals, the high pass filter almost flips the low pass signal once you get close to the distance that scan read for the object it saw

9

The magnitude spectrum for both elliptic filters look fairly the same in comparison to 6.c.iii. It does attenuate the lowest and highest frequencies on the filtered plots in comparison. The filtered plots themselves follow the same trends as the other filtered plots.

1.

The frequency spectrum you see for part 6.c.i is the actual frequencies present in the signal . The appearance of frequency above 8.2 GHz is from aliasing distortion because frequencies higher than Fs/2 will fold back and appear as lower.

2.

The fftshift commands makes it so the frequencies are rearranged so that 0 is in the middle and then the negative and positive frequencies respectively fall around it. So the plot appears more symmetric.

3

In the time domain higher frequencies are elimated or reduced and in the fequency doman there's a gradual roll off as higher frequencies are attenuated

4

The high pass filter does the same as the low pass but in reverse, thus in the plots its line appears to be similar but just lower than the low pass signal.

5

Running the program with breakpoints after n is calculated the order values are: 10, 5, 3, 3. So the elliptic filters both use an order of 3 and the butterworth ones use 10 and 5. Between the low pass filters there wasn't much of a difference however the butterworth looked to to have smoother lines when attenuating the higher frequencies probably due to the higher filter order it used. The high pass filters had a bigger difference. The elliptic had a lot more noise.

Published with MATLAB® R2022b