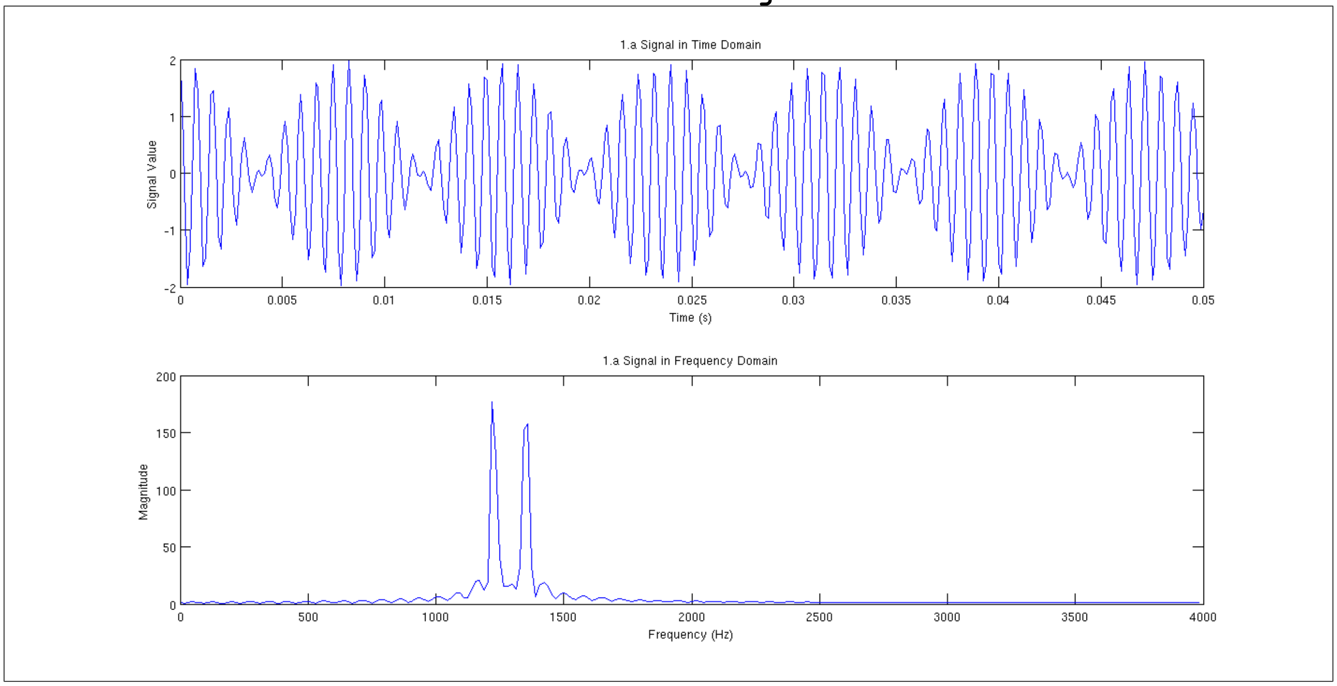
Lab 06
Christopher Bero
EE384

Problem 1

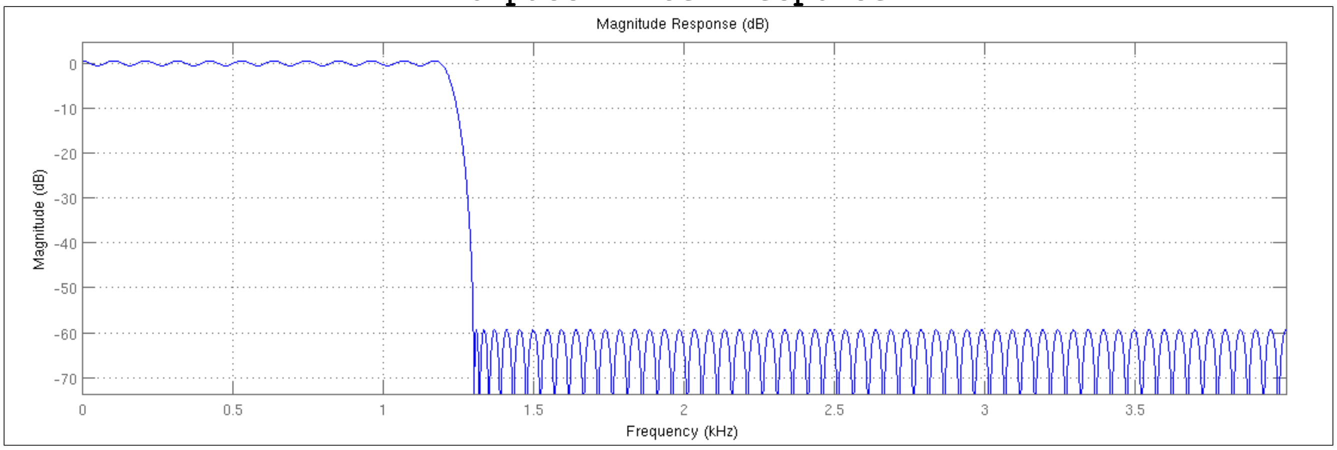
P1.a

Unfiltered Signal

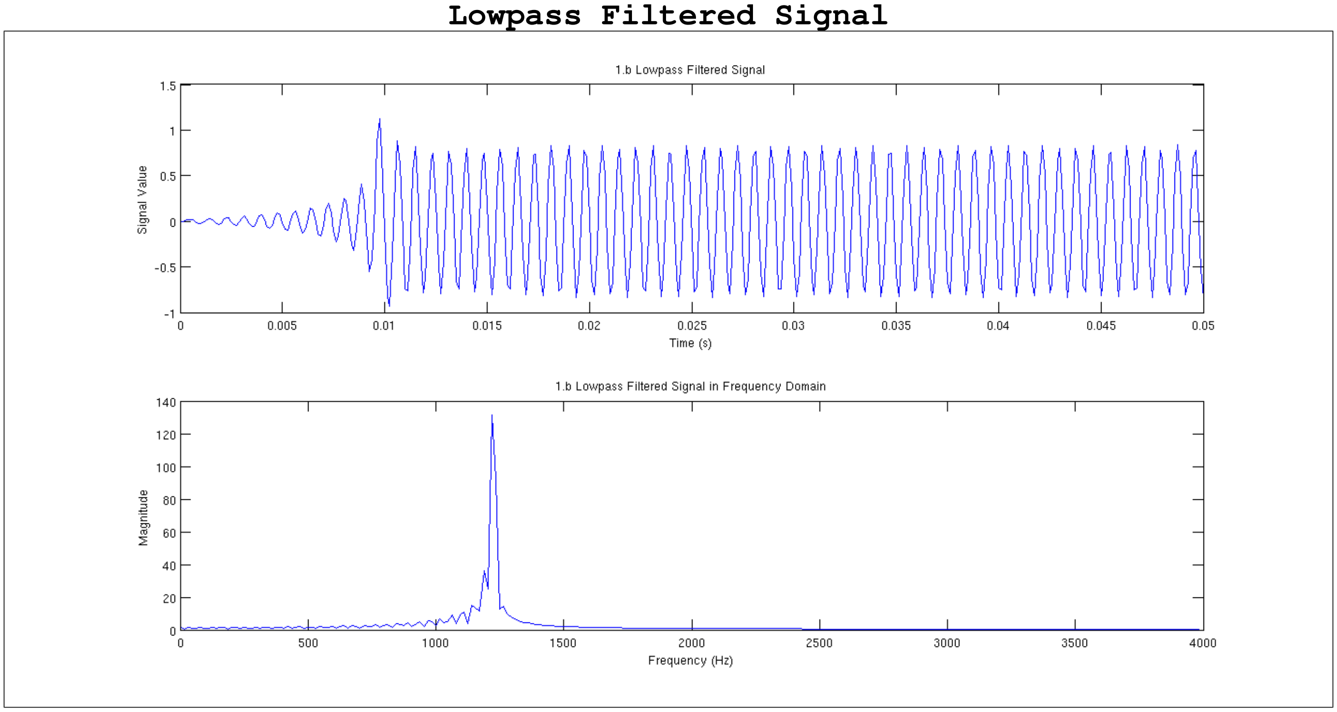


P1.b

Lowpass Filter Response

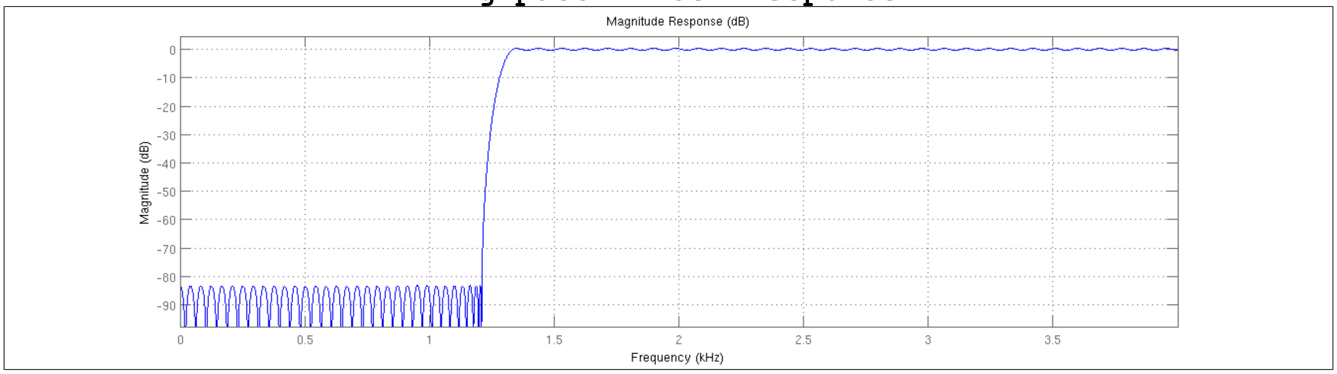


Lowpass Filtered Signal

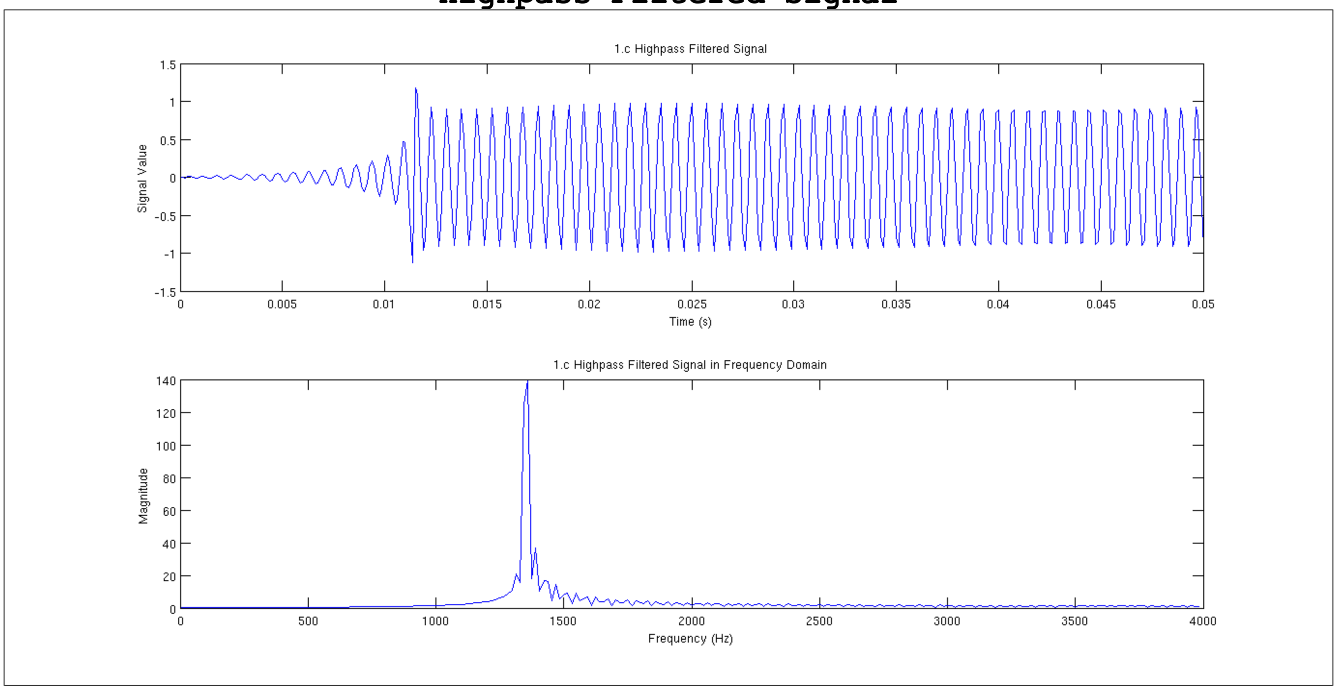


P1.c

Highpass Filter Response

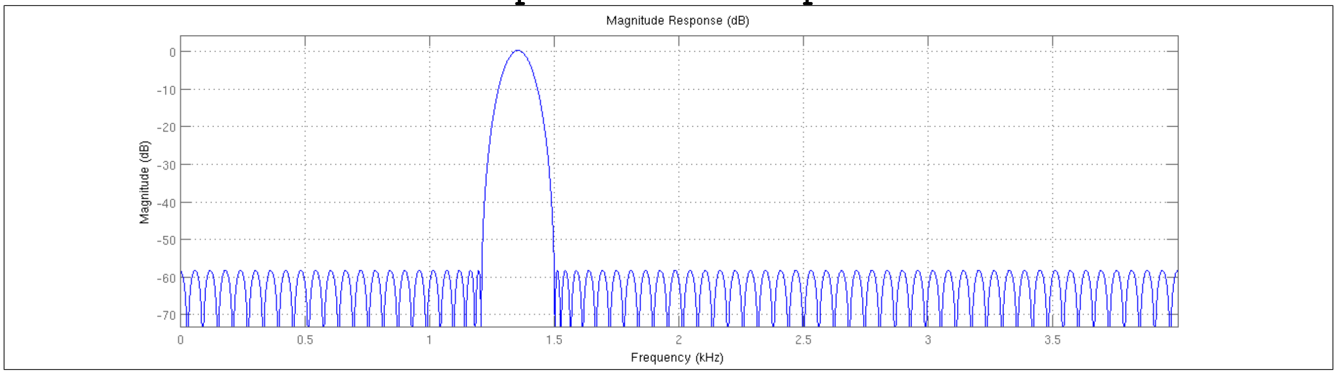


Highpass Filtered Signal

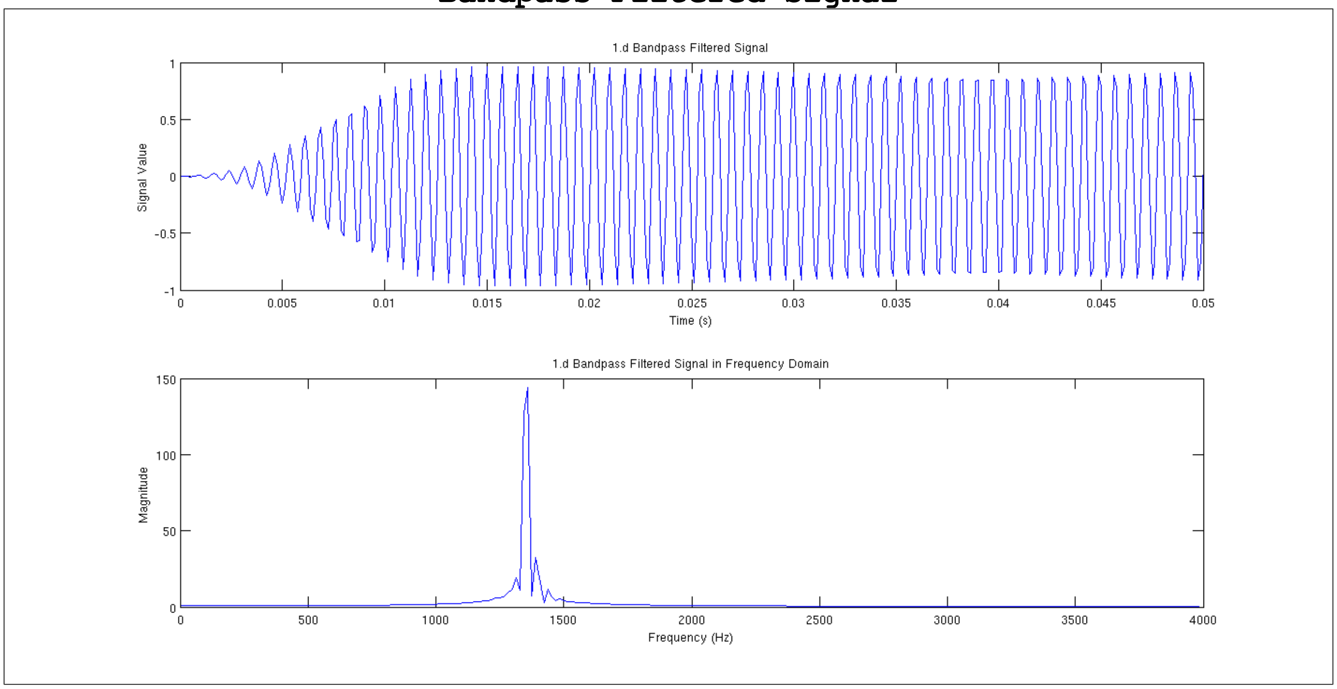


P1.d

Bandpass Filter Response



Bandpass Filtered Signal



```
% Lab 06 - Problem 1
% EE384
% Christopher Bero

% References:
%
ftp://ftp.eri.ucsb.edu/pub/org/lakemix/matlibrary~/users/emery/gener
al/other_peoples/signal/sigdemos/html/dfiltdemo.html
% http://www.gaussianwaves.com/2014/07/how-to-plot-fft-using-matlab-
fft-of-basic-signals-sine-and-cosine-waves/

% Setup
sfs=8000;
t=0:(1/sfs):0.050; % 50ms
f1=1209;
f2=1336;
y1=cos(2*pi*t*f1)+cos(2*pi*t*f2);

% P1.a
% Plot the signal (y1) in both time and freq domains
fig1=figure();
subplot(2,1,1);
plot(t,y1);
title('1.a Signal in Time Domain');
xlabel('Time (s)');
ylabel('Signal Value');

[y1f_vals,y1f_sig]=freqSpec_1s(y1,sfs);

subplot(2,1,2);
plot(y1f_vals,abs(y1f_sig)); % abs() will return magnitude of
complex num
title('1.a Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');

% P1.b
fig2=figure();
lowpass=p1_lowpass;
y2=filter(lowpass,y1);
subplot(2,1,1);
plot(t,y2);
title('1.b Lowpass Filtered Signal');
xlabel('Time (s)');
ylabel('Signal Value');

[y2f_vals,y2f_sig]=freqSpec_1s(y2,sfs);

subplot(2,1,2);
plot(y2f_vals,abs(y2f_sig));
title('1.b Lowpass Filtered Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');


% P1.c
```

```
fig3=figure();
highpass=p1_highpass;
y3=filter(highpass,y1);
subplot(2,1,1);
plot(t,y3);
title('1.c Highpass Filtered Signal');
xlabel('Time (s)');
ylabel('Signal Value');

[y3f_vals,y3f_sig]=freqSpec_1s(y3,sfs);

subplot(2,1,2);
plot(y3f_vals,abs(y3f_sig));
title('1.c Highpass Filtered Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');


% P1.d
fig4=figure();
bandpass=p1_bandpass;
y4=filter(bandpass,y1);
subplot(2,1,1);
plot(t,y4);
title('1.d Bandpass Filtered Signal');
xlabel('Time (s)');
ylabel('Signal Value');

[y4f_vals,y4f_sig]=freqSpec_1s(y4,sfs);

subplot(2,1,2);
plot(y4f_vals,abs(y4f_sig));
title('1.d Bandpass Filtered Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');
```

freqSpec_1s.m

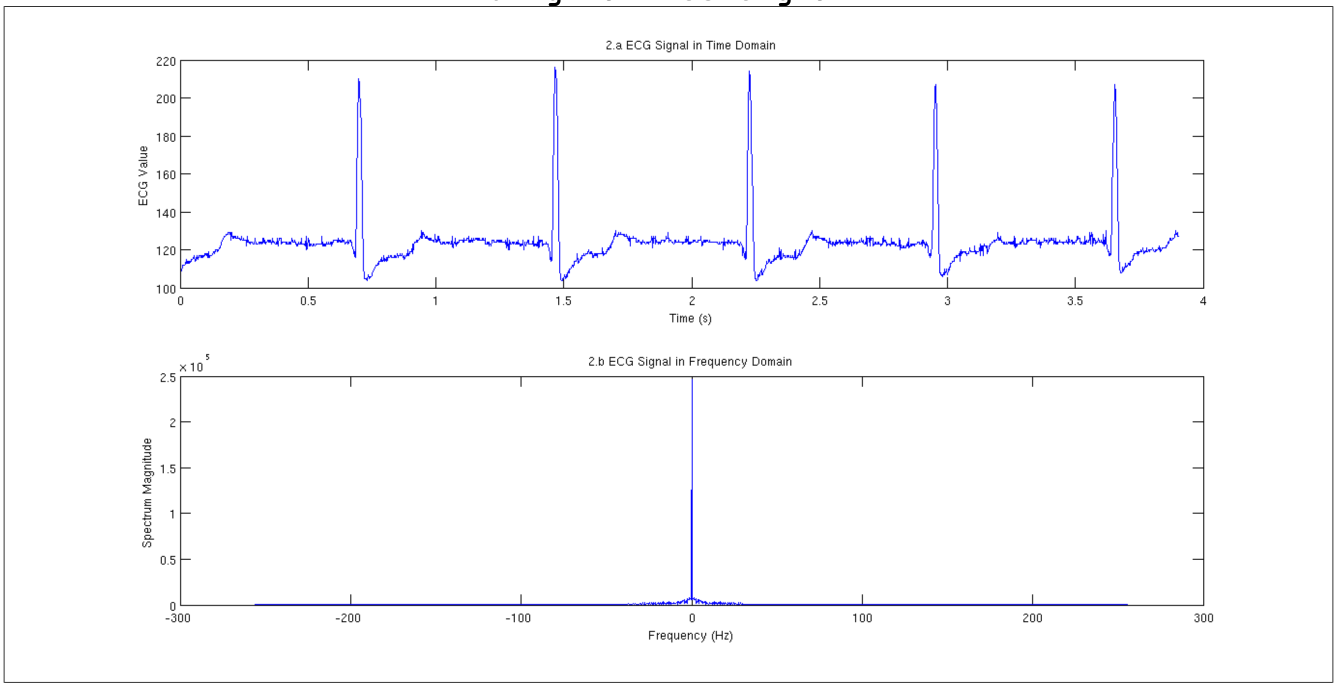
```
function [ f_vals, f_sig ] = freqSpec_1s( signal, sfs )
% freqSpec_1s: Generate one sided frequency spectrum for plotting
% Usage:    [x,y]=freqSpec_1s(time_signal,sample_freq);
%           plot(x,y);

t_len=length(signal); % number of samples
nfft=2^ceil(log2(t_len)); % number of FFT bins
%f_vals=sfs*(-nfft/2:nfft/2-1)/nfft; % Frequency range (w/ neg)
f_vals=sfs*(0:nfft/2-1)/nfft; % Frequency range (positive only)
y1_f=fftshift(fft(signal,nfft)); % Two sided, zero centered FFT
f_sig=y1_f(nfft/2:nfft-1); % Only positive frequencies

end
```

freqSpec_2s.m

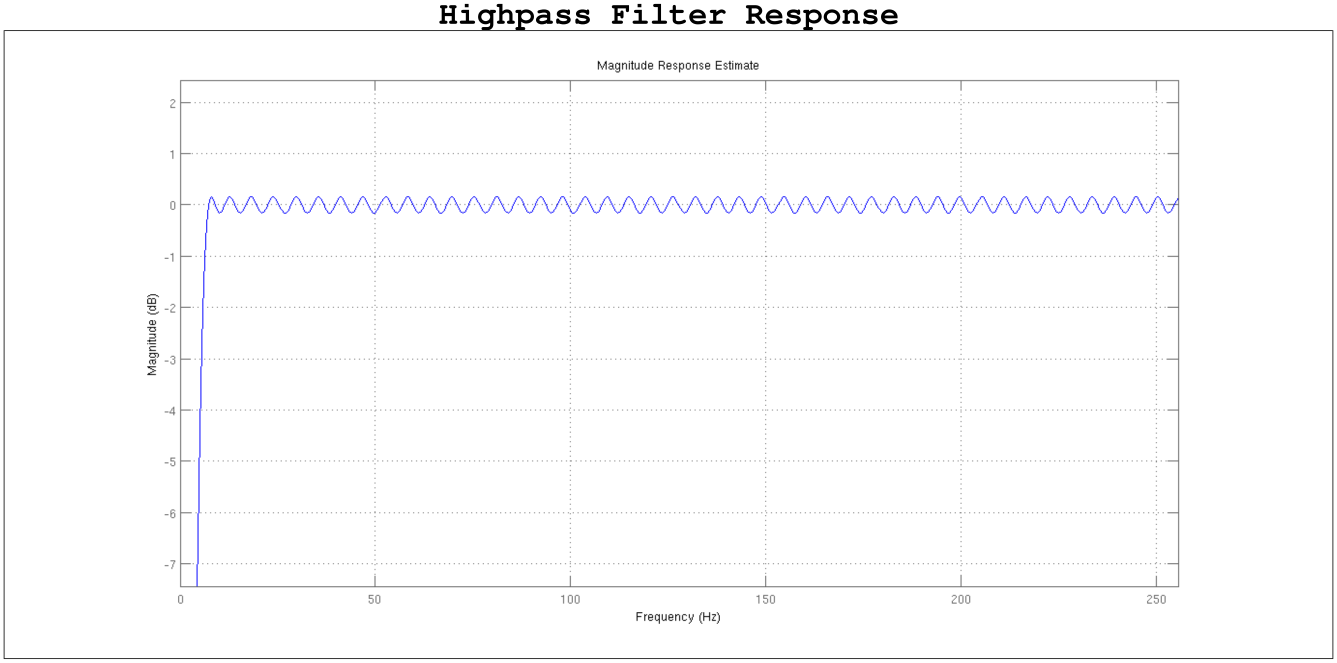
```
function [ f_vals, f_sig ] = freqSpec_2s( signal, sfs )
% freqSpec_2s: Generate two sided frequency spectrum for plotting
% Usage:    [x,y]=freqSpec_1s(time_signal,sample_freq);
%           plot(x,y);

t_len=length(signal); % number of samples
nfft=2^ceil(log2(t_len)); % number of FFT bins
f_vals=sfs*(-nfft/2:nfft/2-1)/nfft; % Frequency range (w/ neg)
f_sig=fftshift(fft(signal,nfft)); % Two sided, zero centered FFT

end
```

Problem 2

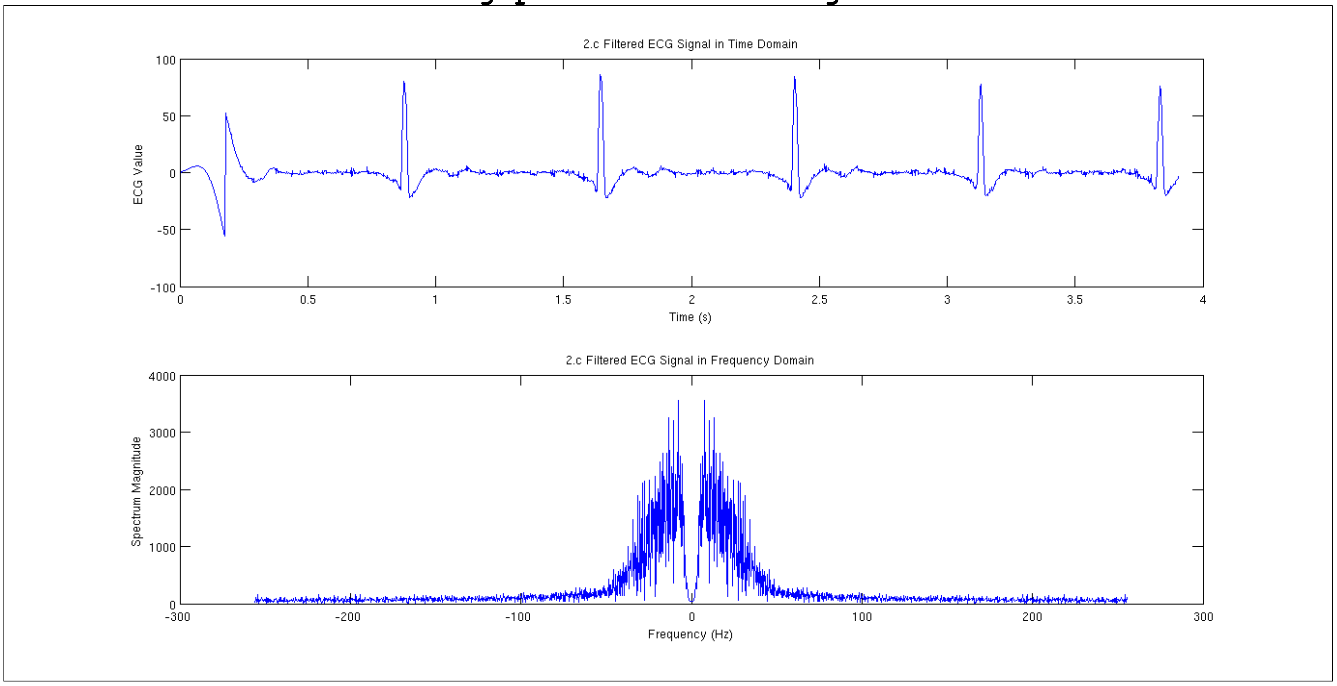
Original ECG Signal



Highpass Filter Response



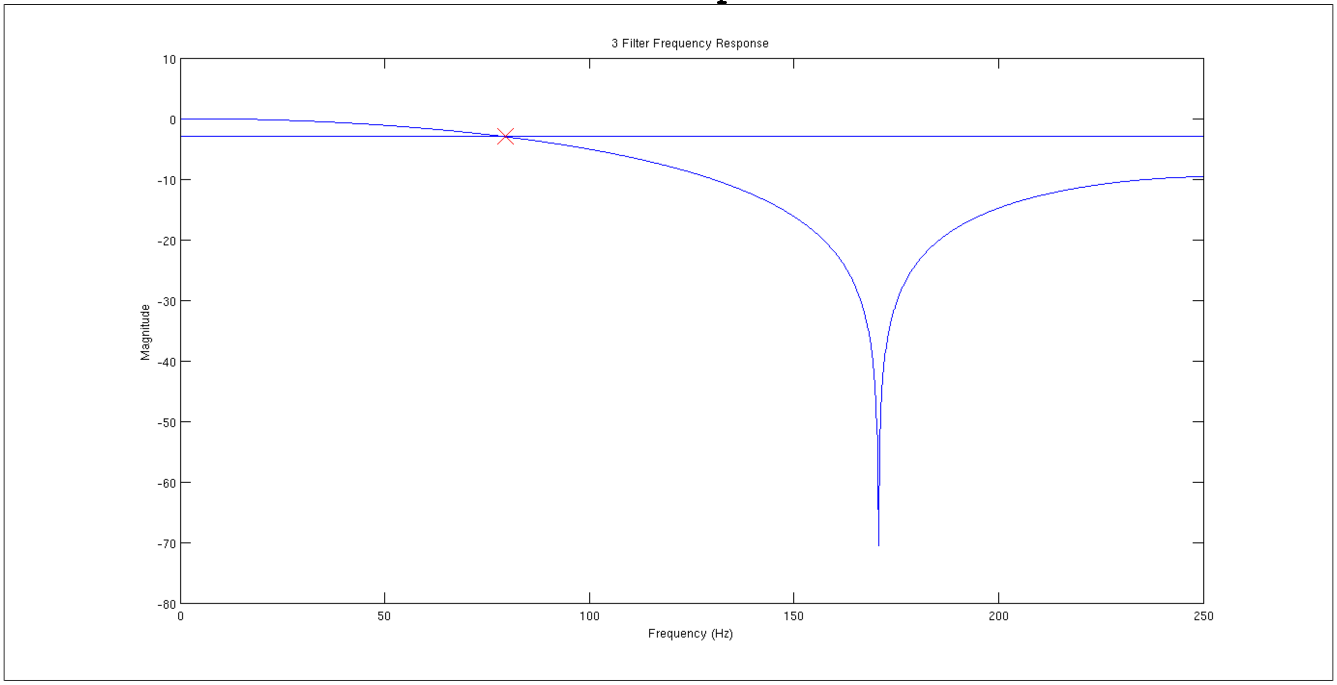
Highpass Filtered Signal



```
% Lab 06 - Problem 2
% EE384
% Christopher Bero

% P2.a
% Load and plot DCG_Data
load('/home/berocs/Documents/uah/ee384/SAMPLE_ECG.mat');
sfs=512;
t=0:1/512:(2000*(1/512))-(1/512);
fig_a=figure();
subplot(2,1,1);
plot(t,ECG_Data);
title('2.a ECG Signal in Time Domain');
xlabel('Time (s)');
ylabel('ECG Value');
% Heart rate appears to be ~80 BPM

% P2.b
% Plot Frequency (two sided) of the ECG signal
[ecgf_vals,ecgf_sig]=freqSpec_2s(ECG_Data,sfs);
figure(fig_a);
subplot(2,1,2);
plot(ecgf_vals,abs(ecgf_sig));
%axis([0 100 -50000 300000]);
title('2.b ECG Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Spectrum Magnitude');

% P2.c
% Apply high pass filter to the ECG signal
highpass=p2_highpass; % Filter everything below 1Hz
yc=filter(highpass,ECG_Data);
fig_b=figure();
subplot(2,1,1);
plot(t,yc);
title('2.c Filtered ECG Signal in Time Domain');
xlabel('Time (s)');
ylabel('ECG Value');
[ecgf_vals,ecgf_sig]=freqSpec_2s(yc,sfs);
figure(fig_b);
subplot(2,1,2);
plot(ecgf_vals,abs(ecgf_sig));
title('2.c Filtered ECG Signal in Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Spectrum Magnitude');
```

Problem 3

Filter Response




```
% Lab 06 - Problem 3
% EE384
% Christopher Bero

% Filter from ECG section of lab 02 had the form:
% y=(x(n)+x(n-1)+x(n-2))/3;
% Plot the frequency response of this filter,
% mark the cutoff frequency
% Estimate the cutoff frequency in Hz
% Sampling Frequency: 512Hz

% I'm not very confident with this solution, but since we're using
% an averaging filter, this should be the frequency response.
% http://www.mathworks.com/help/matlab/data_analysis/filtering-
data.html
%
http://www.mathworks.com/matlabcentral/newsreader/view_thread/128289
% https://en.wikipedia.org/wiki/Cutoff_frequency
% http://www.mathworks.com/matlabcentral/answers/24055-how-to-see-
the-frequency-response-of-difference-equation

a=1;
b=[1/3 1/3 1/3];
[h,f]=freqz(b,a,2048,512);
Db=20*log10(abs(h));
plot(f,Db);
axis([0 250 -80 10]);
title('3 Filter Frequency Response');
xlabel('Frequency (Hz)');
ylabel('Magnitude');
```

```
hold on;
cutoff=ones(length(f),1)*(-3);
plot(f,cutoff);
cutoff_freq=0;
for index=1:length(f)
    if Db(index) < cutoff(index)
        cutoff_freq=f(index);
        break;
    end
end
plot(cutoff_freq,cutoff,'--x','MarkerSize', 20, 'MarkerEdgeColor',
'r');

% This crashes MATLAB
%text_str=sprintf('[%d Hz, %d dB]', cutoff_freq, cutoff);
%text(cutoff_freq,cutoff(1), text_str, 'FontName', 'FixedWidth');

% Cutoff frequency: ~80Hz
```