Table of Contents

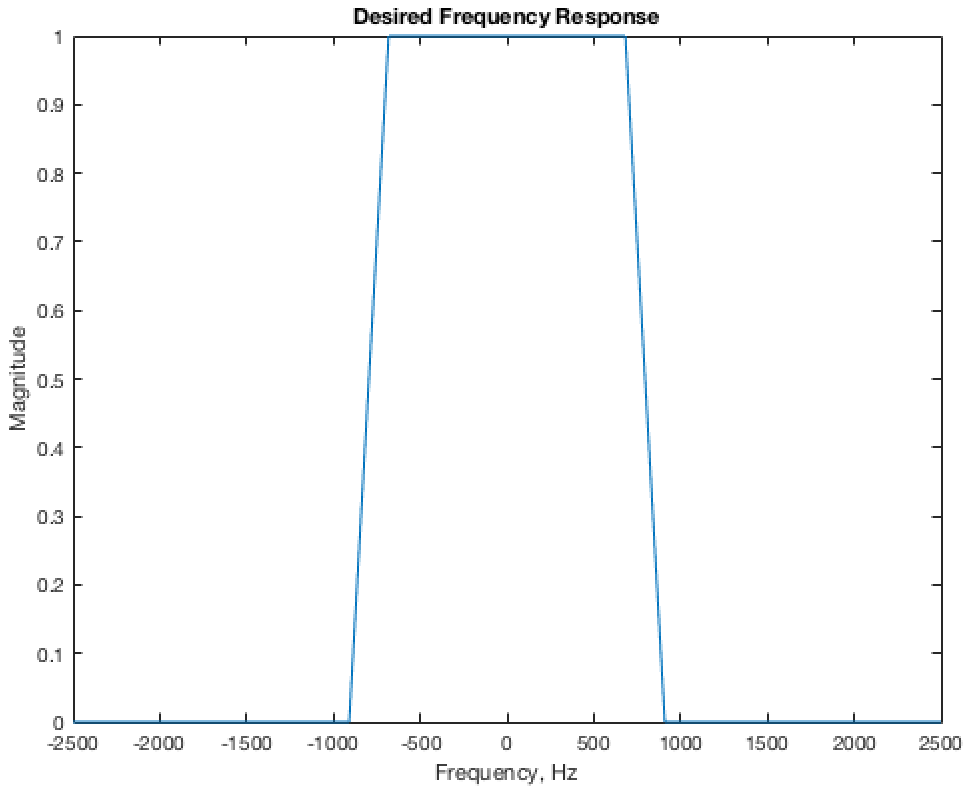
EE125: Digital Signal Processing Matlab Project 6 Alexander Christenson

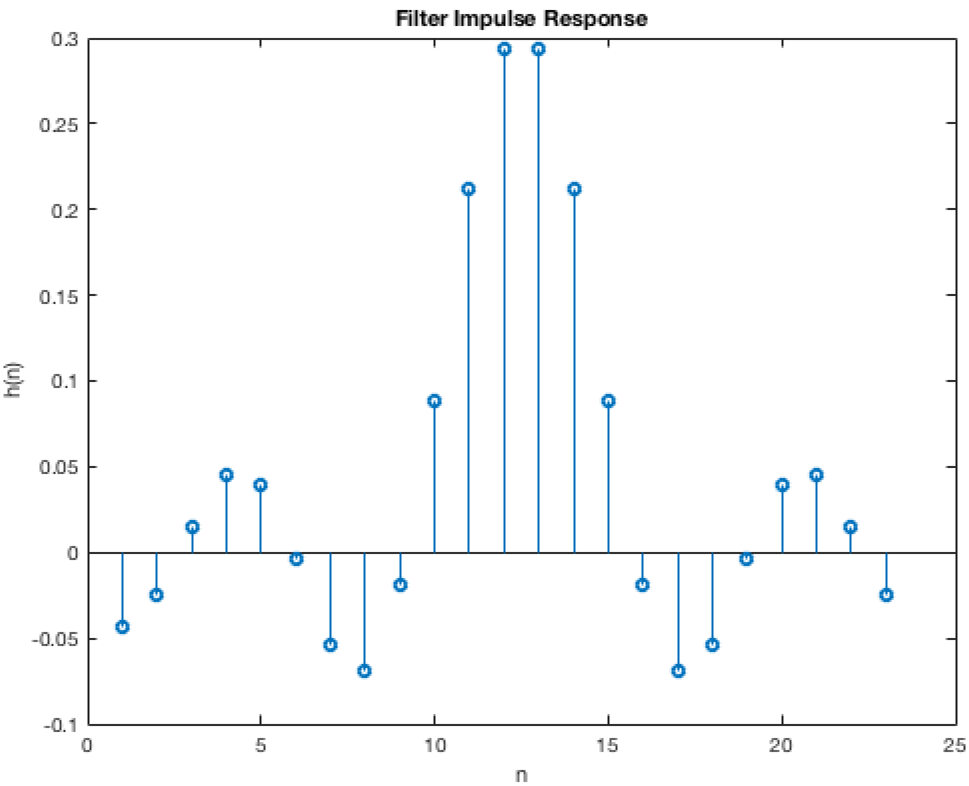
```
clear all, close all
```

Designing a filter by frequency sampling

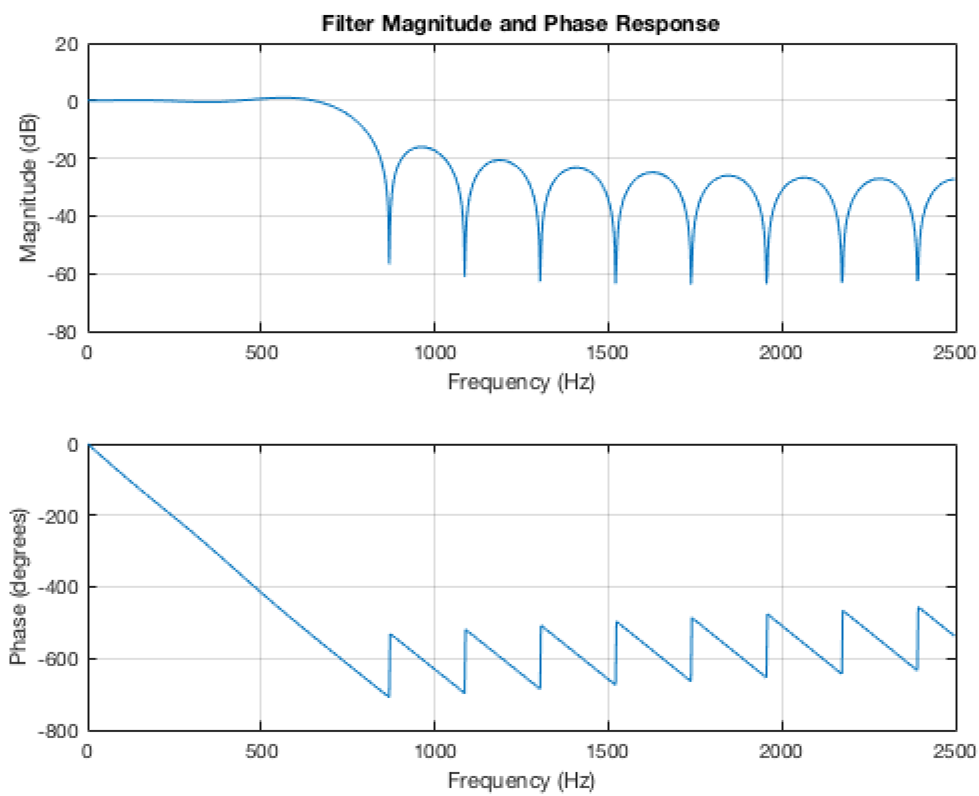
```
% Plugging in the given information
fs = 5000; % Sampling Freq, Hz
L = 23; % Filter length
fc = 750;

% Creating the frequency vector
df = fs/(L-1);
M = (L-1)/2;
f = [(0:M)*df (-M:-1)*df];

% Creating the magnitude of the filter frequency response
Hmag = double(abs(f)<fc);

% Plotting the frequency response to check
figure
plot(fftshift(f),fftshift(Hmag));
title('Desired Frequency Response')
xlabel('Frequency, Hz')
ylabel('Magnitude');

% Creating the linear phase
w = f*2*pi/fs;
Hphase = exp(1i*w*M);
H = Hmag.*Hphase;
h = real(ifft(H));
figure
stem(1:L,h)
```

```
title('Filter Impulse Response')
xlabel('n')
ylabel('h(n)')

figure
freqz(h,1,1024,fs);
title('Filter Magnitude and Phase Response')
% Yes, this is the frequency and phase responses I was expecting.
 Since the
% magnitude response is plotted in decibels it makes sense that you
 couldnt
% have that represented in the response because that equates to -inf!!
```





Filter Impulse Response
h(n)
n

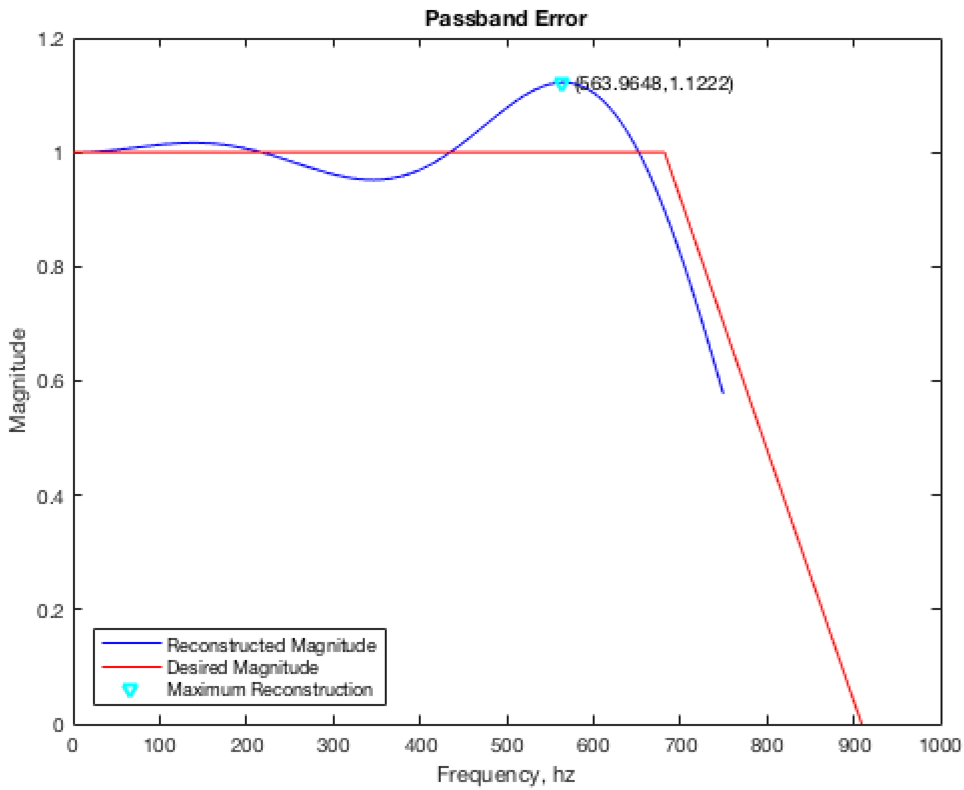


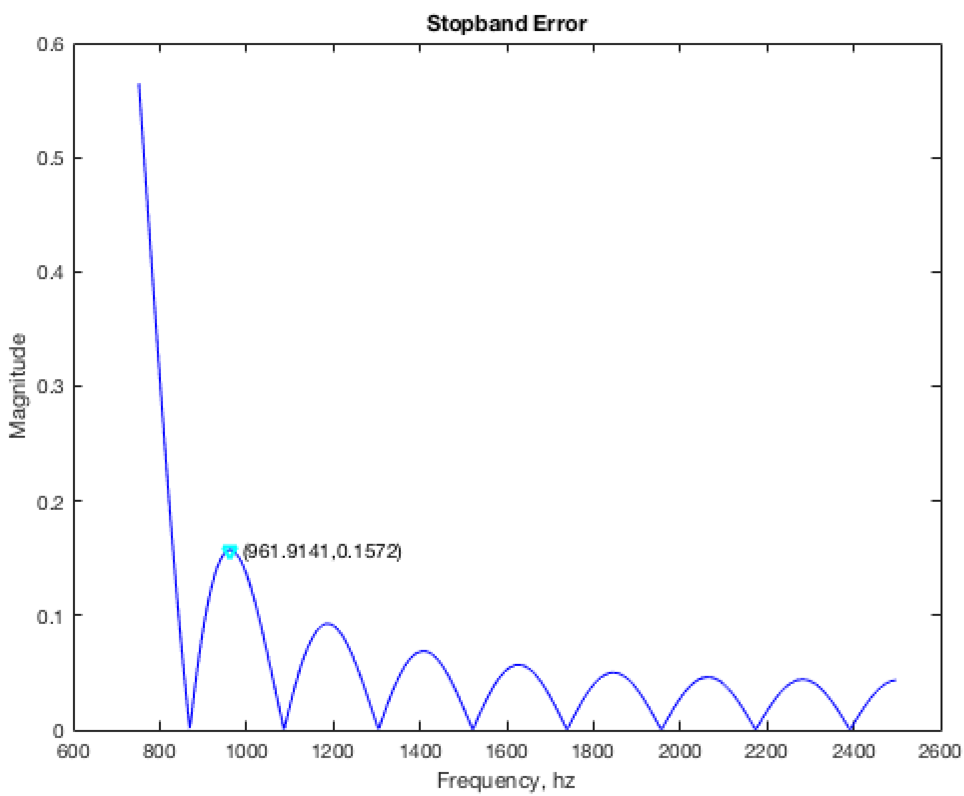
Filter Magnitude and Phase Response
Magnitude (dB)
Frequency (Hz)
Phase (degrees)
Frequency (Hz)

Effect of the transition band

```
[Herr,ferr] = freqz(h,1,1024,fs);
pb = find(ferr<fc);
Hpb = abs(Herr(pb));
figure
plot(ferr(pb),Hpb,'b')
[maximum,idx] = max(Hpb);
hold on
f0 = find(f==0);
H0 = find(Hmag==0);
plot(f(f0:H0),Hmag(f0:H0),'r')
plot(ferr(idx),Hpb(idx),'vc')
hold off
title('Passband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
legend('Reconstructed Magnitude','Desired Magnitude',...
    'Maximum Reconstruction','Location','southwest')
txt = sprintf('  (%.4f,%.4f)',ferr(idx),Hpb(idx));
text(ferr(idx),Hpb(idx),txt)

pbErr = Hpb(idx)-1 % This error is the largest, 0.12, at ~564 Hz

sb = find(ferr>=fc);
Hsb = abs(Herr(sb));
fsb = ferr(sb);
figure
plot(fsb,Hsb,'b')
hold on
[peaks,pts] = findpeaks(Hsb);
[maximum,idx] = max(peaks);
idx = pts(idx);
txt = sprintf('  (%.4f,%.4f)',fsb(idx),Hsb(idx));
text(fsb(idx),Hsb(idx),txt)
plot(fsb(idx),Hsb(idx),'vc')
title('Stopband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
hold off

sbErr = Hsb(idx) % This maximum stopband error of 0.157 is at ~962 Hz
```

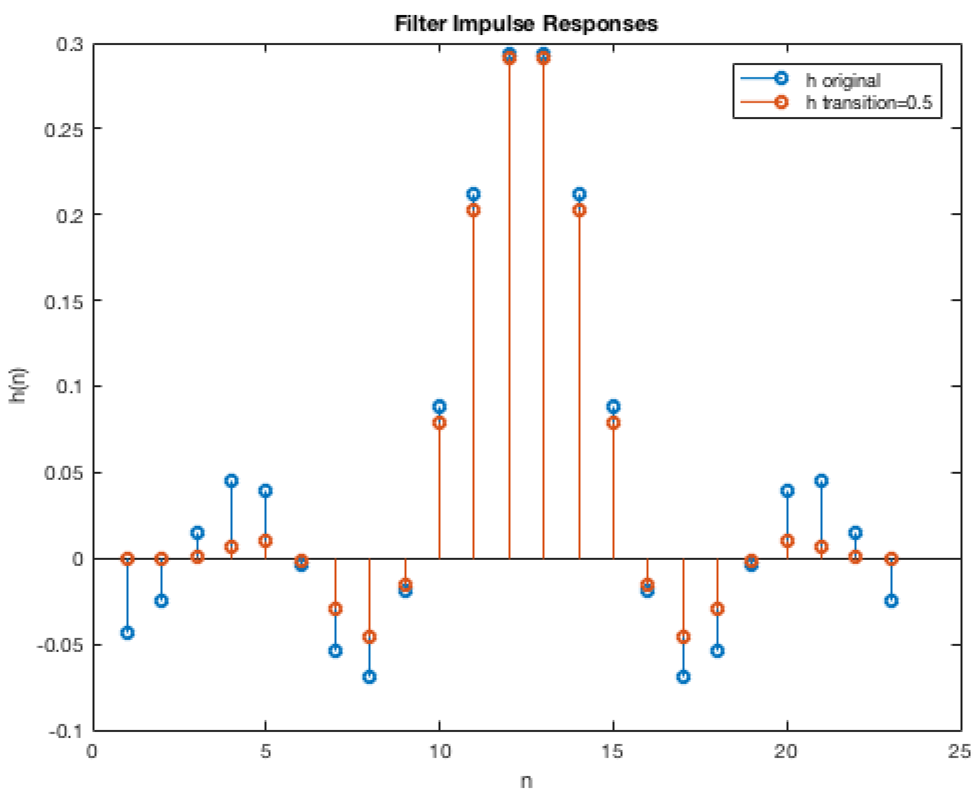
```
pbErr =

    0.1222


sbErr =

    0.1572
```


Passband Error
(563.9648,1.1222)
Reconstructed Magnitude
Desired Magnitude
Maximum Reconstruction
Magnitude
Frequency, hz
0
0.2
0.4
0.6
0.8
1
1.2
0
100
200
300
400
500
600
700
800
900
1000


Stopband Error
(961.9141,0.1572)
Magnitude
Frequency, hz
0
0.1
0.2
0.3
0.4
0.5
0.6
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600

Adding the transition band

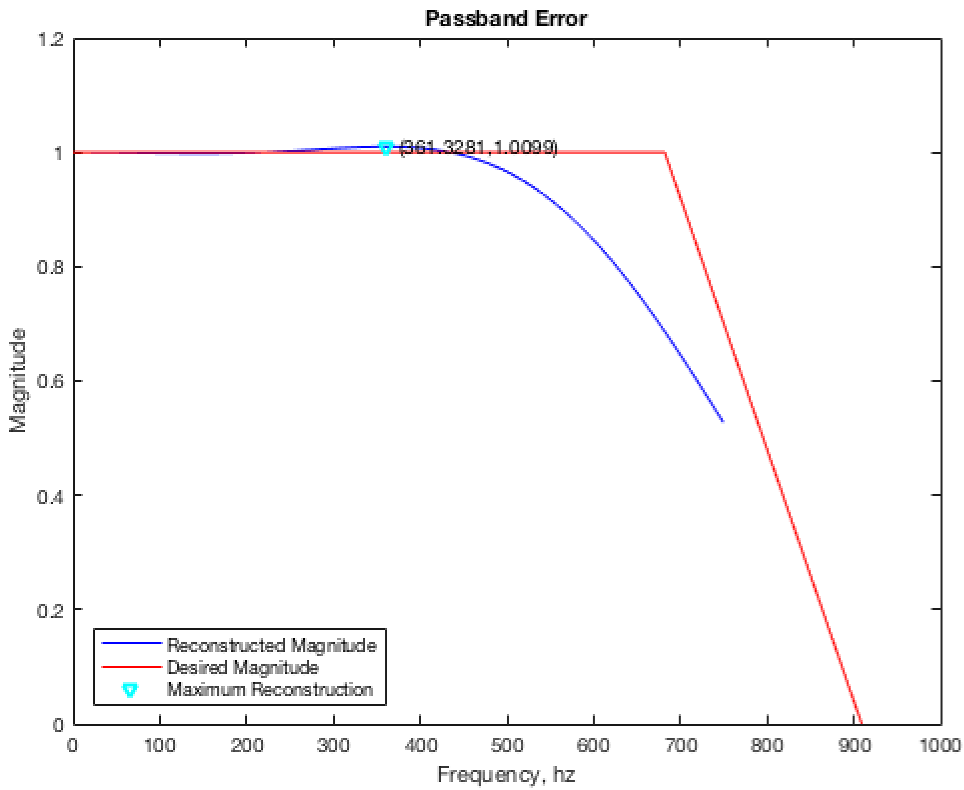
```
[minimum, trans] = min(abs(f-fc));
Hmag1 = Hmag;
Hmag1(trans) = 0.5; % Set trans value
Hmag1(L+1-trans) = 0.5; % Do it for the negative side too

% Using the same phase, invert the Fourier transform
H1 = Hmag1.*Hphase;
h1 = real(ifft(H1));

% Get an plot filter coefficients
figure
stem(1:L,h)
hold on
stem(1:L,h1)
title('Filter Impulse Responses')
legend('h original','h transition=0.5')
xlabel('n')
ylabel('h(n)')
hold off

% This looks like the original impulse response, however, it is a
 little
% more vertically squished, but it does still look sinc-like
```

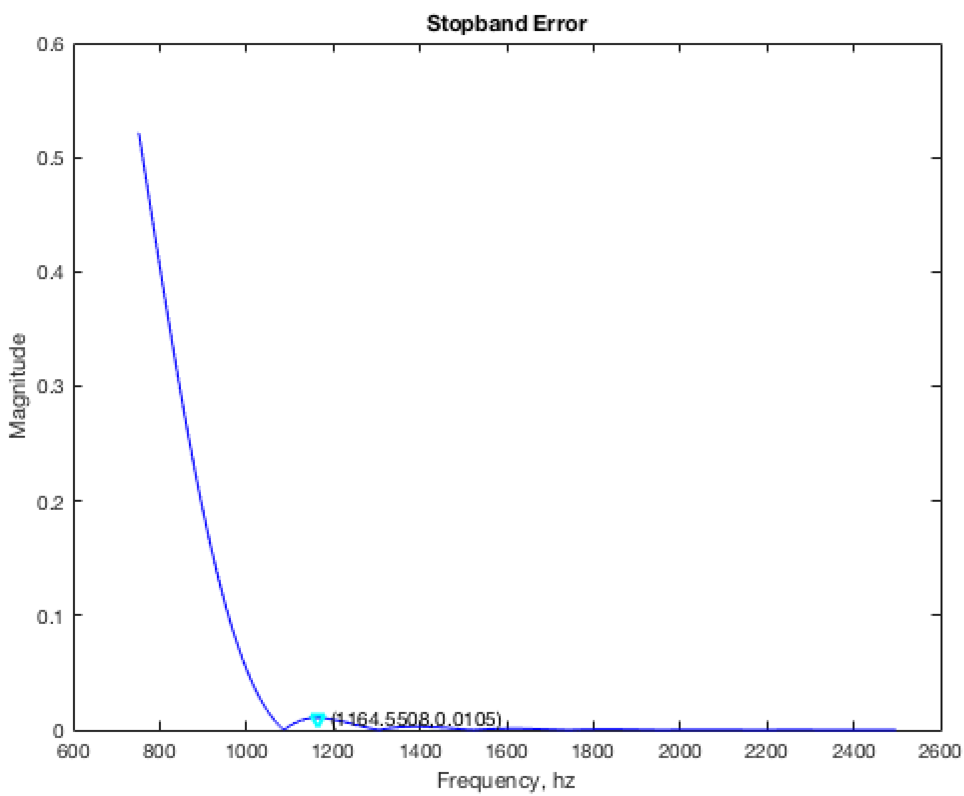


Repeating previous steps w trans band

```
[Herr,ferr] = freqz(h1,1,1024,fs);
pb = find(ferr<fc);
Hpb = abs(Herr(pb));
figure
plot(ferr(pb),Hpb,'b')
[maximum,idx] = max(Hpb);
hold on
f0 = find(f==0);
H0 = find(Hmag==0);
plot(f(f0:H0),Hmag(f0:H0),'r')
plot(ferr(idx),Hpb(idx),'vc')
hold off
title('Passband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
legend('Reconstructed Magnitude','Desired Magnitude',...
    'Maximum Reconstruction','Location','southwest')
txt = sprintf('  (%.4f,%.4f)',ferr(idx),Hpb(idx));
text(ferr(idx),Hpb(idx),txt)

pbErrTrans = Hpb(idx)-1 % Max err of 0.0099 at ~361 Hz

sb = find(ferr>=fc);
Hsb = abs(Herr(sb));
fsb = ferr(sb);
figure
plot(fsb,Hsb,'b')
hold on
[peaks,pts] = findpeaks(Hsb);
[maximum,idx] = max(peaks);
idx = pts(idx);
txt = sprintf('  (%.4f,%.4f)',fsb(idx),Hsb(idx));
text(fsb(idx),Hsb(idx),txt)
plot(fsb(idx),Hsb(idx),'vc')
title('Stopband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
hold off

sbErrTrans = Hsb(idx) % Max err of 0.0105 at ~1165 Hz
```

```
pbErrTrans =

    0.0099


sbErrTrans =

    0.0105
```


Passband Error
(361.3281,1.0099)
Reconstructed Magnitude
Desired Magnitude
Maximum Reconstruction
Magnitude
Frequency, hz
0
0.2
0.4
0.6
0.8
1
1.2
0
100
200
300
400
500
600
700
800
900
1000


Stopband Error
(1164.5508,0.0105)
Magnitude
Frequency, hz
0
0.1
0.2
0.3
0.4
0.5
0.6
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600

Test and tabulate various transition value errors

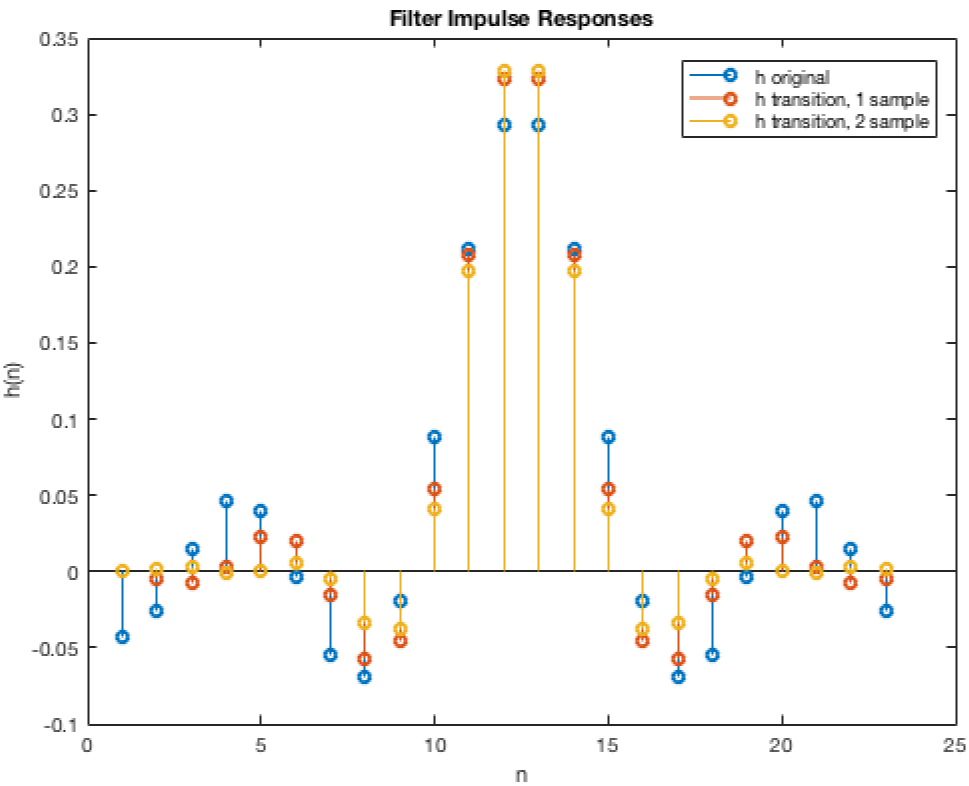
```
pbErrTab = zeros(1,9);
sbErrTab = zeros(1,9);
for t = 0.1:0.1:0.9
    Hmag1(trans) = t;
    Hmag1(L+1-trans) = t;
    H1 = Hmag1.*Hphase;
    h1 = real(ifft(H1));
    HH = freqz(h1,1,1024,fs);
    pbErrTab(round(t*10)) = max(abs(HH(pb)))-1;
    sbErrTab(round(t*10)) = max(findpeaks(abs(HH(sb))));
end
Passband_Error = pbErrTab';
Stopband_Error = sbErrTab';
Transition_Values =
 {'0.1';'0.2';'0.3';'0.4';'0.5';'0.6';'0.7';'0.8';'0.9'};
TGap1 = table(Passband_Error,
 Stopband_Error, 'RowNames',Transition_Values)

% From the table there is a clear tradeoff between the transition
 value and
% the pass/stop band error. As the transition value is shifted right
 the
% passband error decreases and the stopband error increases. So for
 good
% stopband performance choose a low transition value and for good
 passband
% performance choose a high transition value.
```

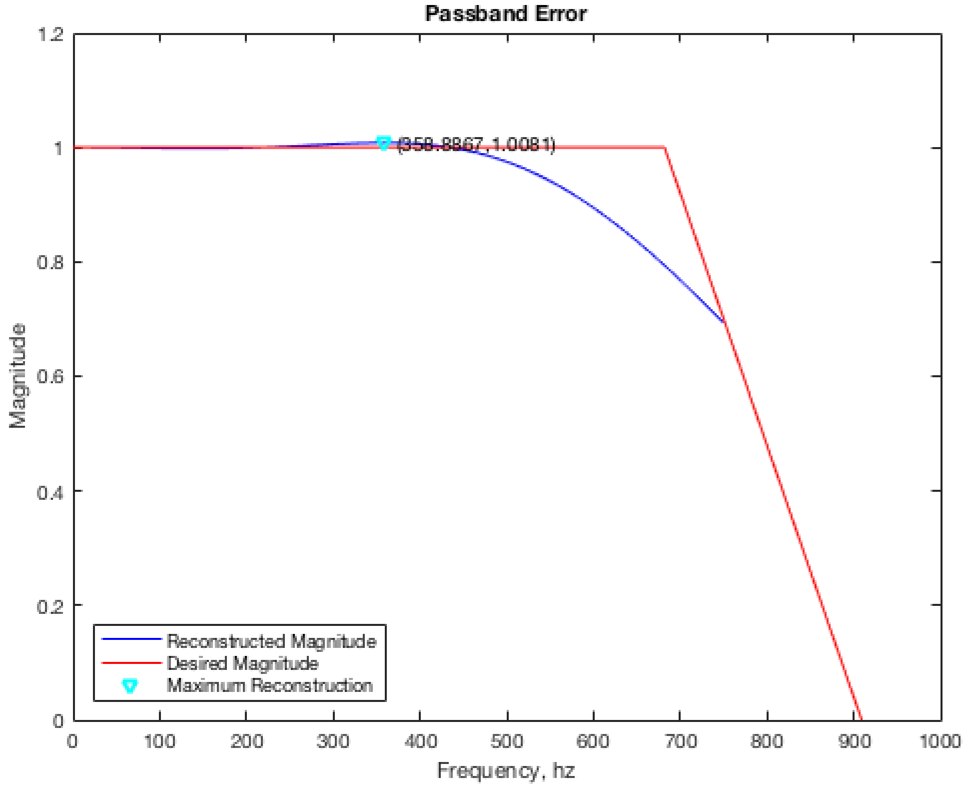
```
TGap1 =

  9×2 table

           Passband_Error    Stopband_Error
           ______________    ______________

    0.1      0.025604          0.0089399
    0.2      0.021676          0.0026562
    0.3       0.01775          0.0021327
    0.4      0.013829          0.0062425
    0.5     0.0099102           0.010484
    0.6     0.0060083            0.01475
    0.7     0.0021896           0.019025
    0.8     0.0009212           0.023303
    0.9     0.0086479           0.027581
```

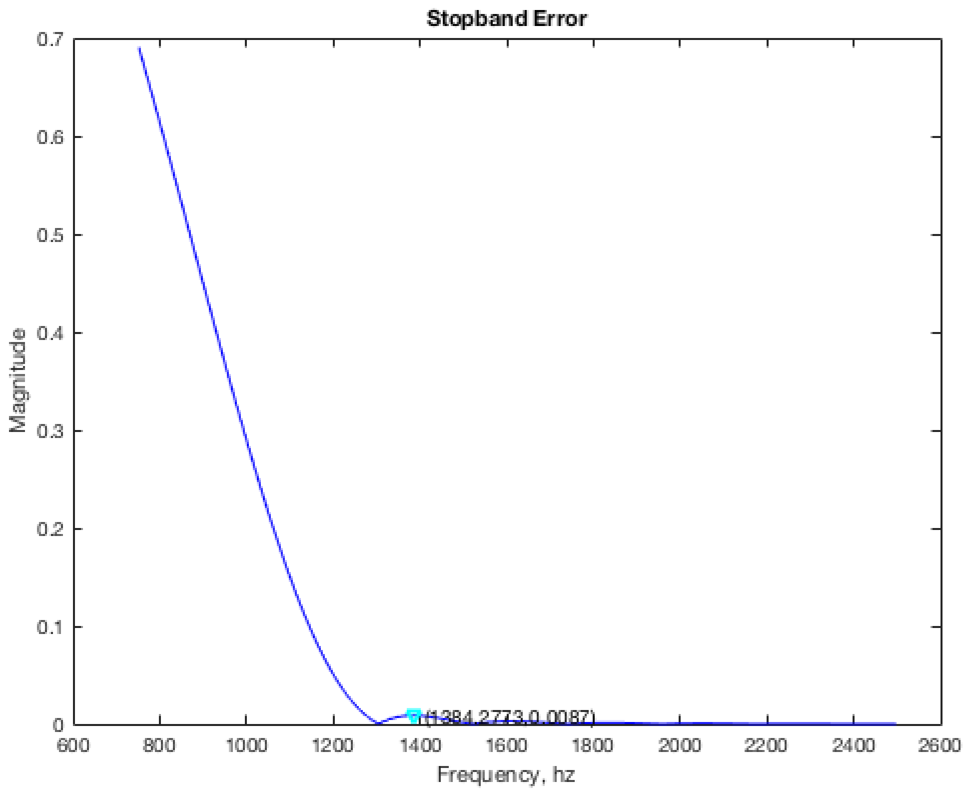
Repeat for a 2 sample transition band

```
Hmag2 = Hmag;
Hmag2(trans:trans+1) = [2 1]/3;
Hmag2(L-trans:L+1-trans) = [1 2]/3;
H2 = Hmag2.*Hphase;

% Using the same phase, invert the Fourier transform
h2 = real(ifft(H2));

% Get an plot filter coefficients
figure
stem(1:L,h)
hold on
stem(1:L,h1)
stem(1:L,h2)
title('Filter Impulse Responses')
legend('h original','h transition, 1 sample','h transition, 2 sample')
xlabel('n')
ylabel('h(n)')
hold off

% Repeating previous steps w trans band
[Herr,ferr] = freqz(h2,1,1024,fs);
pb = find(ferr<fc);
Hpb = abs(Herr(pb));
figure
plot(ferr(pb),Hpb,'b')
[maximum,idx] = max(Hpb);
hold on
f0 = find(f==0);
H0 = find(Hmag==0);
plot(f(f0:H0),Hmag(f0:H0),'r')
plot(ferr(idx),Hpb(idx),'vc')
hold off
title('Passband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
legend('Reconstructed Magnitude','Desired Magnitude',...
    'Maximum Reconstruction','Location','southwest')
txt = sprintf('   (%.4f,%.4f)',ferr(idx),Hpb(idx));
text(ferr(idx),Hpb(idx),txt)

pbErrTrans2 = Hpb(idx)-1

sb = find(ferr>=fc);
Hsb = abs(Herr(sb));
fsb = ferr(sb);
figure
plot(fsb,Hsb,'b')
hold on
[peaks,pts] = findpeaks(Hsb);
[maximum,idx] = max(peaks);
```

```
idx = pts(idx);
txt = sprintf('  (%.4f,%.4f)',fsb(idx),Hsb(idx));
text(fsb(idx),Hsb(idx),txt)
plot(fsb(idx),Hsb(idx),'vc')
title('Stopband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
hold off

sbErrTrans2 = Hsb(idx)

% Notice that using the linear phase 2 sample transition band causes
 the
% measured error in both the stop and pass band to be smaller than any
 of
% the 1 sample transition band errors from the table. (compare table
 with
% pbErrTrans2 and sbErrTrans2 for the passband and stopband errors
 with 2
% samples)


pbErrTrans2 =

    0.0081


sbErrTrans2 =

    0.0087
```


Filter Impulse Responses
h original
h transition, 1 sample
h transition, 2 sample
h(n)
n
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
0
5
10
15
20
25

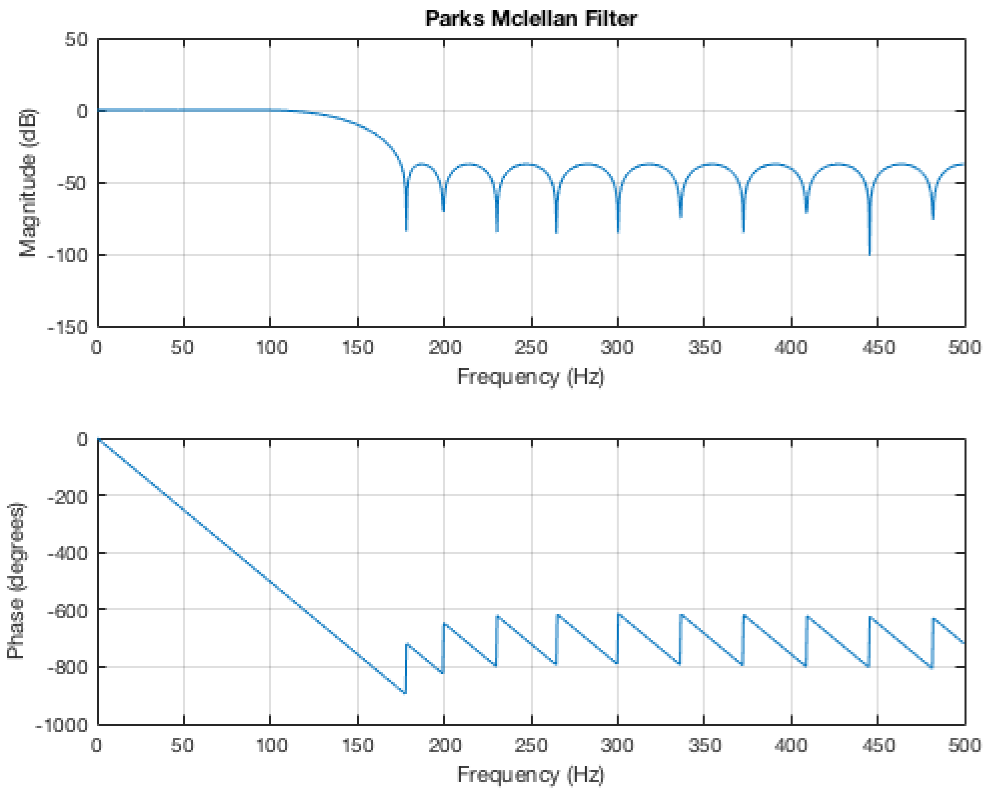

Passband Error
(358.8867,1.0081)
Reconstructed Magnitude
Desired Magnitude
Maximum Reconstruction
Magnitude
Frequency, hz
1.2
1
0.8
0.6
0.4
0.2
0
0
100
200
300
400
500
600
700
800
900
1000

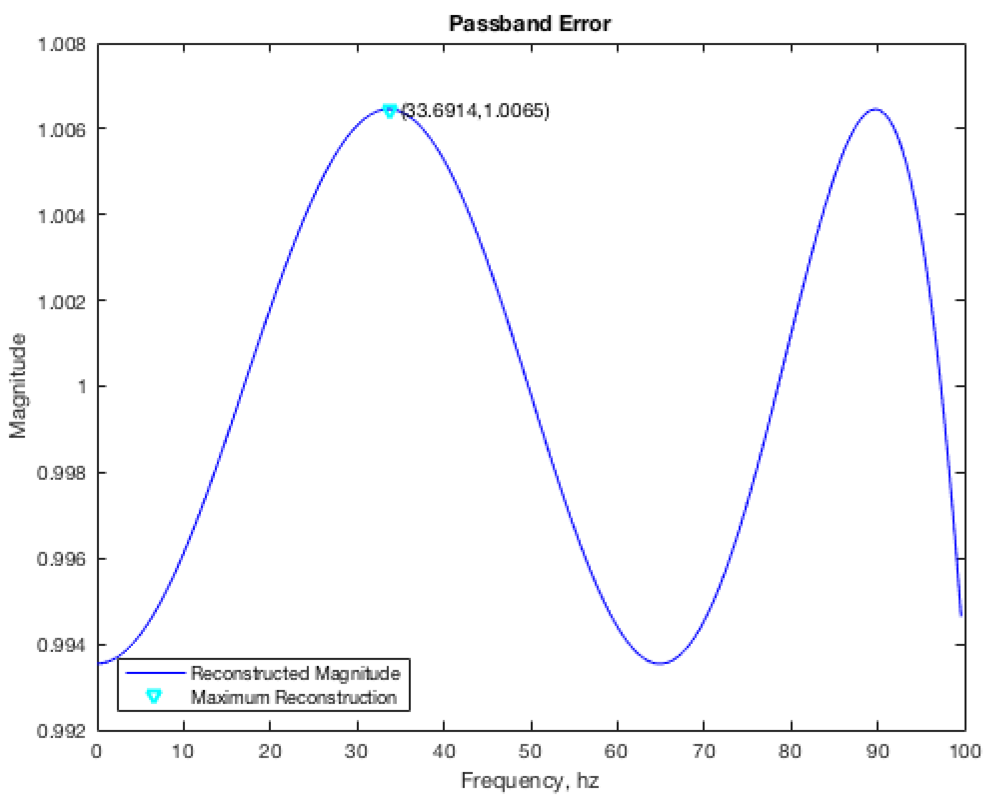


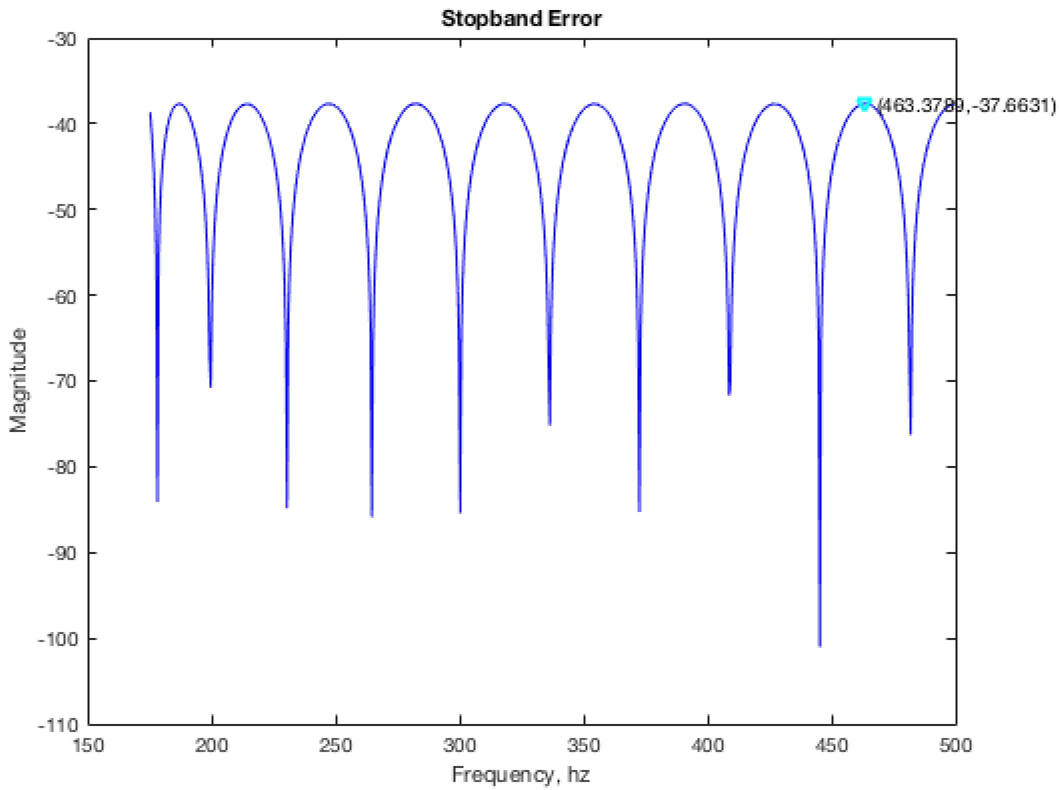
Parks Mclellan

```
% From firpm doc
rp = 10*log10(1.02);    % Passband ripple
rs = 40;                % Stopband ripple
fs = 1000;              % Sampling frequency
f = [100 175];          % Cutoff frequencies
a = [1 0];              % Desired amplitudes

dev = [(10^(rp/20)-1)/(10^(rp/20)+1)  10^(-rs/20)];
[n,fo,ao,w] = firpmord(f,a,dev,fs);
b = firpm(n,fo,ao,w);
freqz(b,1,1024,fs)
title('Parks Mclellan Filter')

[Herr,ferr] = freqz(b,1,1024,fs);
pb = find(ferr<f(1));
Hpb = abs(Herr(pb));
figure
plot(ferr(pb),Hpb,'b')
[maximum,idx] = max(Hpb);
hold on
plot(ferr(idx),Hpb(idx),'vc')
hold off
title('Passband Error')
```

```
xlabel('Frequency, hz')
ylabel('Magnitude')
legend('Reconstructed Magnitude',...
    'Maximum Reconstruction','Location','southwest')
txt = sprintf('  (%.4f,%.4f)',ferr(idx),Hpb(idx));
text(ferr(idx),Hpb(idx),txt)

pbErrPM = Hpb(idx) - 1;

sb = find(ferr>f(2));
Hsb = mag2db(abs(Herr(sb)));
fsb = ferr(sb);
figure
plot(fsb,Hsb,'b')
hold on
[peaks,pts] = findpeaks(Hsb);
[maximum,idx] = max(peaks);
idx = pts(idx);
txt = sprintf('  (%.4f,%.4f)',fsb(idx),Hsb(idx));
text(fsb(idx),Hsb(idx),txt)
plot(fsb(idx),Hsb(idx),'vc')
title('Stopband Error')
xlabel('Frequency, hz')
ylabel('Magnitude')
hold off

sbErPM = Hsb(idx) + rs;

% No adjustment necessary
```


Parks Mclellan Filter
Magnitude (dB)
Frequency (Hz)
Phase (degrees)
Frequency (Hz)

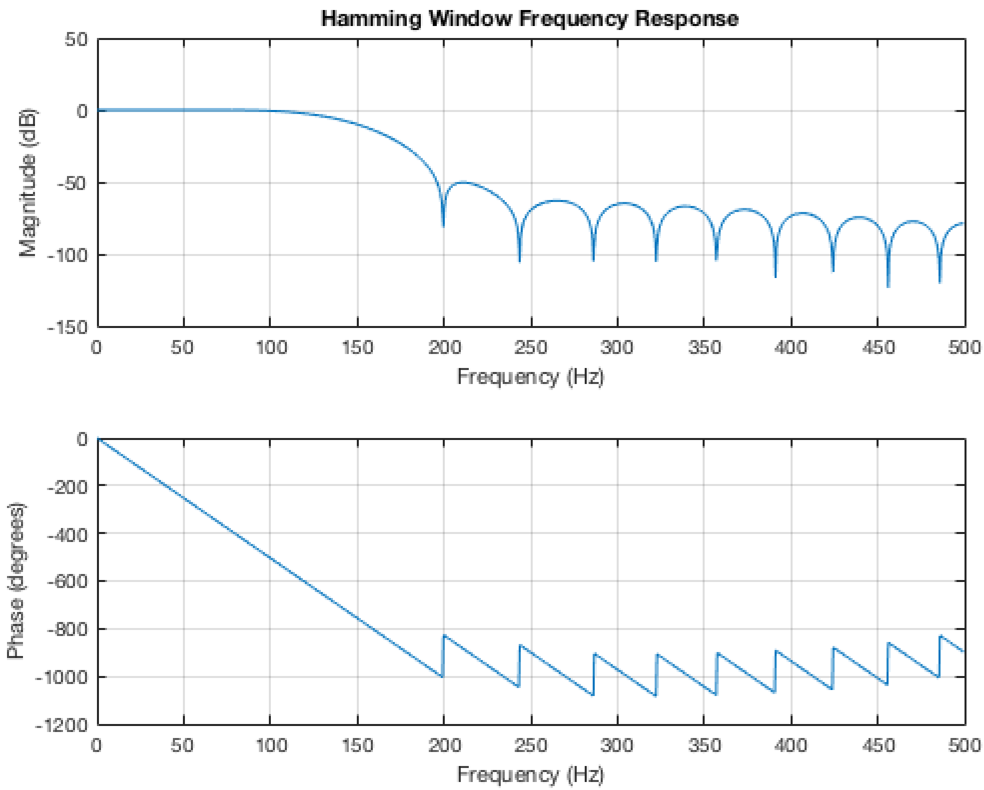

Passband Error
(33.6914,1.0065)
Magnitude
Reconstructed Magnitude
Maximum Reconstruction
Frequency, hz

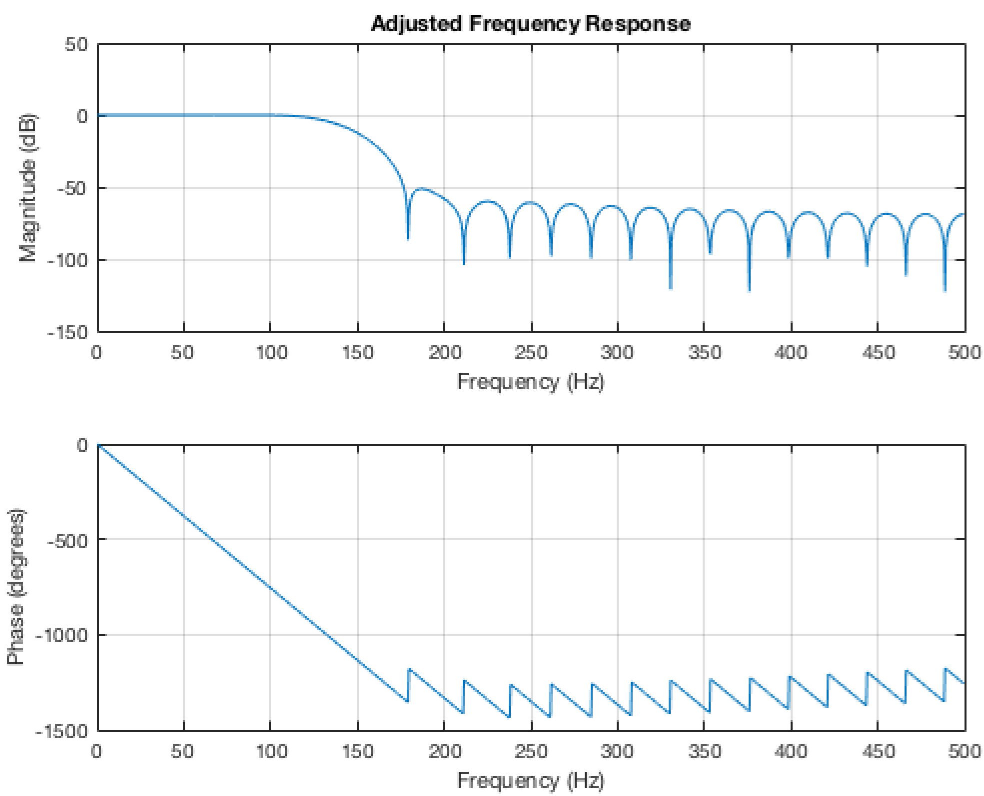


Filter Design By Windowing

1. Hamming window cutoff -> 2fc/fs - middle of trans band

```
hamm = fir1(n,(f(1)+f(2))/fs,'low');
figure
freqz(hamm,1,1024,fs)
title('Hamming Window Frequency Response')
% Doesn't meet specs

hamm = fir1(1.5*n,(f(1)+f(2))/fs,'low');
figure
freqz(hamm,1,1024,fs)
title('Adjusted Frequency Response')
% Looks like it meets the specs
```

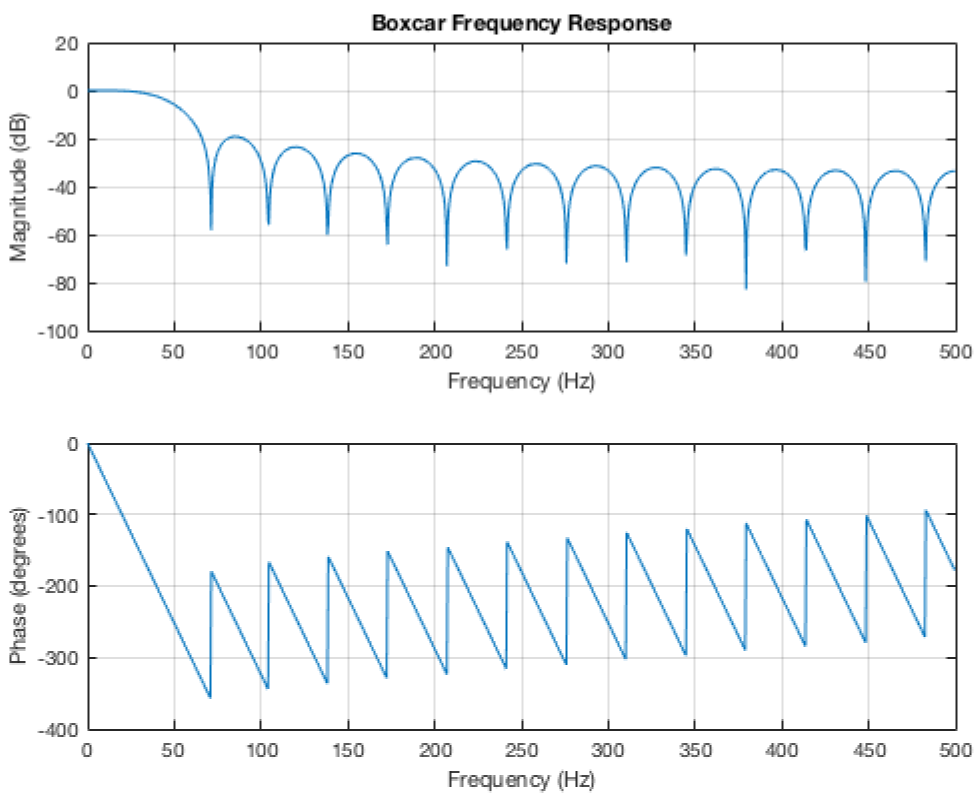

Hamming Window Frequency Response
Magnitude (dB)
Frequency (Hz)
Phase (degrees)
Frequency (Hz)

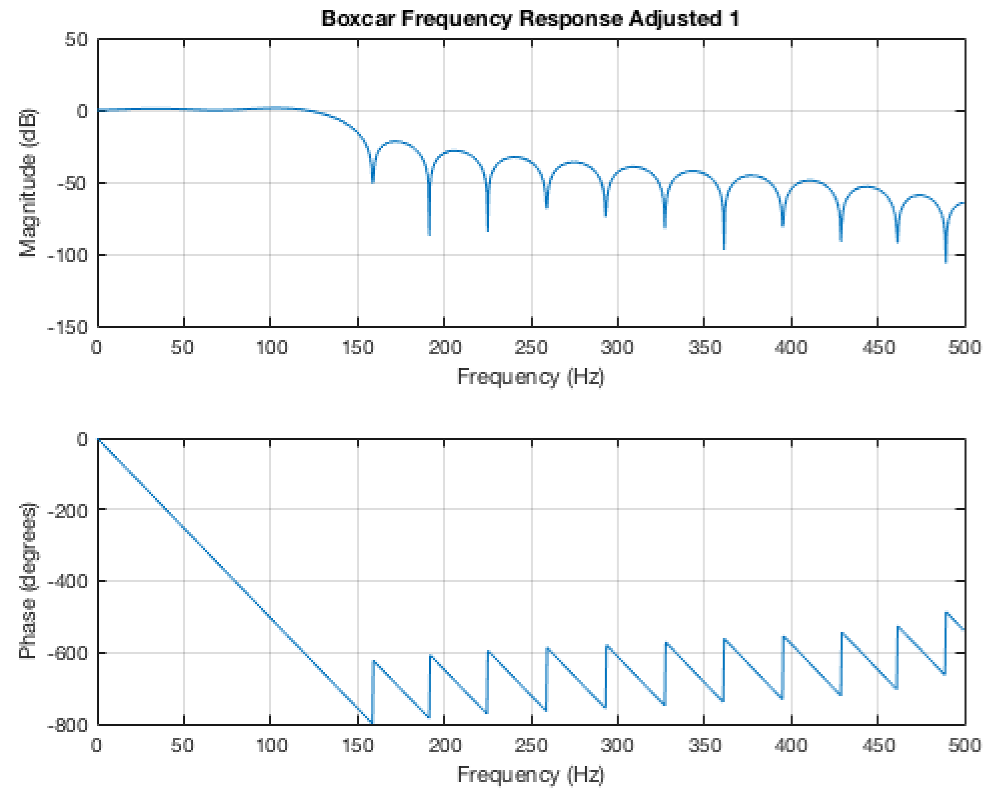

Adjusted Frequency Response
Magnitude (dB)
Frequency (Hz)
Phase (degrees)
Frequency (Hz)

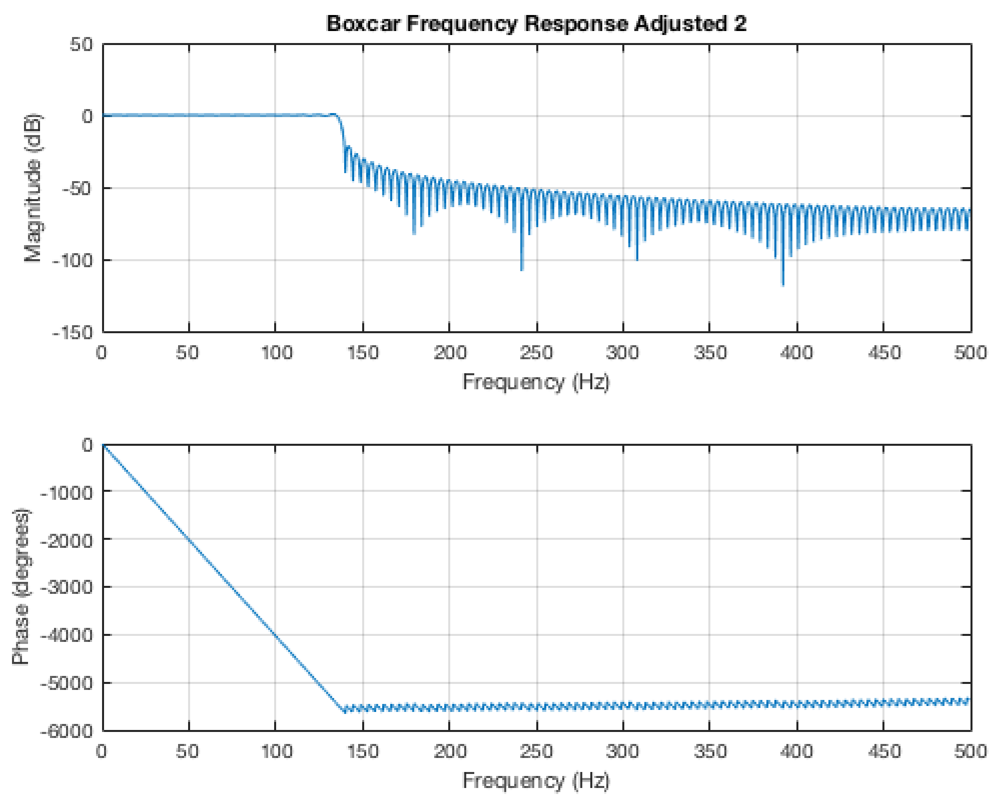
2 Boxcar window

Try making the boxcar with the start at the low cutoff freq

```
bc = fir1(n,f(1)/fs,rectwin(n+1));
figure
freqz(bc,1,1024,fs)
title('Boxcar Frequency Response')
% Doesn't meet specs - cutoff too quick
% Try using the transition point
bc = fir1(n,(f(1)+f(2))/fs,rectwin(n+1));
figure
freqz(bc,1,1024,fs)
title('Boxcar Frequency Response Adjusted 1')
% This is a better frequency to change the window length from
% Improving the response by changing the window length to 8x
bc = fir1(8*n,(f(1)+f(2))/fs,rectwin(8*n+1));
figure
freqz(bc,1,1024,fs)
title('Boxcar Frequency Response Adjusted 2')
% Yay this meets the specs :)
```




Boxcar Frequency Response Adjusted 1
Magnitude (dB)
50
0
-50
-100
-150
0
50
100
150
200
250
300
350
400
450
500
Frequency (Hz)
Phase (degrees)
0
-200
-400
-600
-800
Frequency (Hz)


Boxcar Frequency Response Adjusted 2
Magnitude (dB)
50
0
-50
-100
-150
0
50
100
150
200
250
300
350
400
450
500
Frequency (Hz)
Phase (degrees)
0
-1000
-2000
-3000
-4000
-5000
-6000
Frequency (Hz)

3

FFT has complexity of nlogn

```
% Calculating complexity for equiripple
EC = n*log(n) % = 93.3017
HC = 2*n*log(2*n) % = 225.4197 - ~2.4x slower
BC = 8*n*log(8*n) % = 1212.2 - ~13x slower

% Calculate percent savings
HE = (HC-EC)/HC * 100 % ~58% Savings
BE = (BC-EC)/BC * 100 % ~92% Savings wow thats a lot of savings

% Shoutout to Richard Preston for guidance in some places
```

```
EC =

   93.3017


HC =

  225.4197


BC =

   1.2122e+03


HE =

   58.6098


BE =

   92.3032
```

Published with MATLAB® R2017b