MODULE 05 HOMEWORK

3/5/18

EN.525.718.81.SP18 MULTIRATE SIGNAL PROCESSING

SPRING 2018

Contents

Problem 1

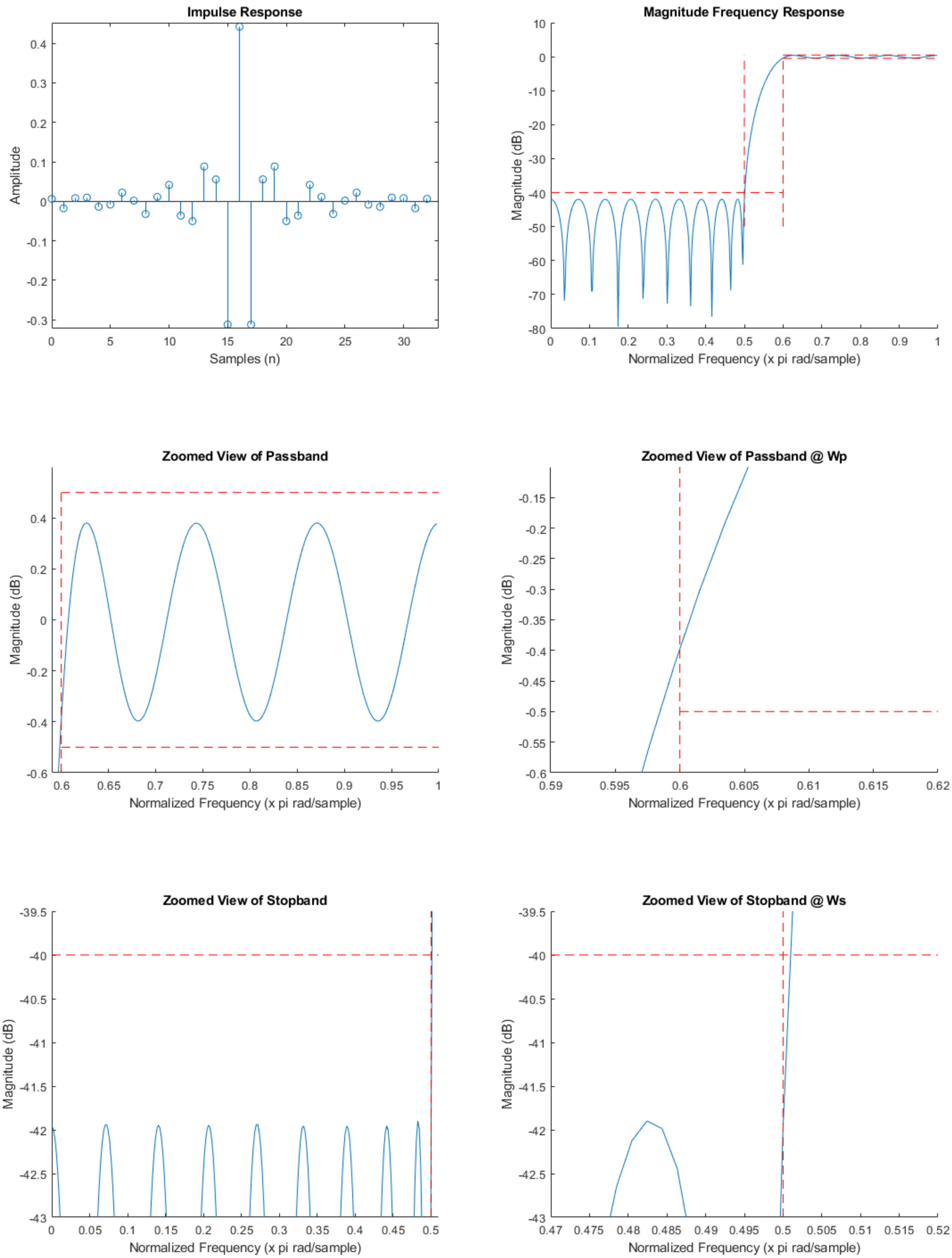
Using the Parks-McClellan algorithm, design a highpass filter with the following specifications: ωp=0.6π, ωs=0.5π, Rp=1dB and Rs=40dB. Plot the impulse response and the frequency response magnitude. Show clearly that your design meets all specifications by plotting the specification template on the frequency response graph. You will find the MATLAB functions firpm and firpmord helpful for this problem.

For this problem the given passband ripple parameter, Rp, was assumed to be the max peak to peak allowable passband ripple in dB. It can be seen in the script that the value of Rp is divided by 2 before being used with the MATLAB firpmord() function. (I asked for clarification in the module 5 discussion but never received a response. In order to play it safe I assumed it to be the max allowable peak to peak value because it would still satisfy the filter requirements).

```
% LPF passband/stopband specifications
wp = 0.6*pi;
ws = 0.5*pi;

% Rp was assumed to be the peak to peak passband ripple value as opposed to
% the peak ripple value. I asked in the for clarification on this in the
% module 5 discussion forum but did not recieve any responses.
Rp =  1;    %dB
Rs = 40;    %dB


% Change Rp to Peak ripple as opposed to Peak to Peak ripple
Rp = Rp/2;


% Calculate linear 1-ds and 1+ds ripple parameters. The smallest value of
% ds will be used in the firpmord() function.
% 20*Log(1-d1) = -Rp
% 20*Log(1+d1) =  Rp
Rp_linear = min([(1-10^(-Rp/20)) (10^(Rp/20)-1)]);


% Compute estimated filter order and other design params using firpmord()
[N,fo,mo,w] = firpmord([ws/pi wp/pi], [0 1], [10^(-Rs/20) Rp_linear]);


% Increase filter order by 2 since original order was under estimated
N = N+2;
b = firpm(N,fo,mo,w);


% Compute Frequency Response
[H,w] = freqz(b);


% Plot Impulse Response
figure(1)
stem((0:N),b);
axis([0 N+1 min(b)-0.01 max(b)+0.01]);
title('Impulse Response');
xlabel('Samples (n)'); ylabel('Amplitude');
```

```
% Plot Frequency Response
figure(2)
hold on
plot(w/pi, 20*log10(abs(H)));
title('Magnitude Frequency Response');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude (dB)');
line([wp/pi wp/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([ws/pi ws/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([0 wp/pi],[-Rs -Rs],'color','red','LineStyle','--');
line([wp/pi 1],[-Rp -Rp],'color','red','LineStyle','--');
line([wp/pi 1],[Rp Rp],'color','red','LineStyle','--');


% Zoomed view of passband
figure(3)
hold on
plot(w/pi, 20*log10(abs(H)));
title('Zoomed View of Passband');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude (dB)');
axis([wp/pi-0.01 1 -Rp-0.1 Rp+0.1]);
line([wp/pi wp/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([ws/pi ws/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([0 wp/pi],[-Rs -Rs],'color','red','LineStyle','--');
line([wp/pi 1],[-Rp -Rp],'color','red','LineStyle','--');
line([wp/pi 1],[Rp Rp],'color','red','LineStyle','--');


% Further zoomed view on passband at wp
figure(4)
hold on
plot(w/pi, 20*log10(abs(H)));
title('Zoomed View of Passband @ Wp');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude (dB)');
axis([0.59 0.62 -0.6 -0.1]);
line([wp/pi wp/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([ws/pi ws/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([0 wp/pi],[-Rs -Rs],'color','red','LineStyle','--');
line([wp/pi 1],[-Rp -Rp],'color','red','LineStyle','--');
line([wp/pi 1],[Rp Rp],'color','red','LineStyle','--');


% Zoomed view of stopband
figure(5)
hold on
plot(w/pi, 20*log10(abs(H)));
title('Zoomed View of Stopband');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude (dB)');
axis([0 ws/pi+0.01 -43 -39.5]);
line([ws/pi ws/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([0 wp/pi],[-Rs -Rs],'color','red','LineStyle','--');


% Further zoomed view of stopband at ws
figure(6)
hold on
plot(w/pi, 20*log10(abs(H)));
title('Zoomed View of Stopband @ Ws');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude (dB)');
axis([0.47 0.52 -43 -39.5]);
```

```
line([ws/pi ws/pi],[-Rs-10 Rp],'color','red','LineStyle','--');
line([0 wp/pi],[-Rs -Rs],'color','red','LineStyle','--');
```



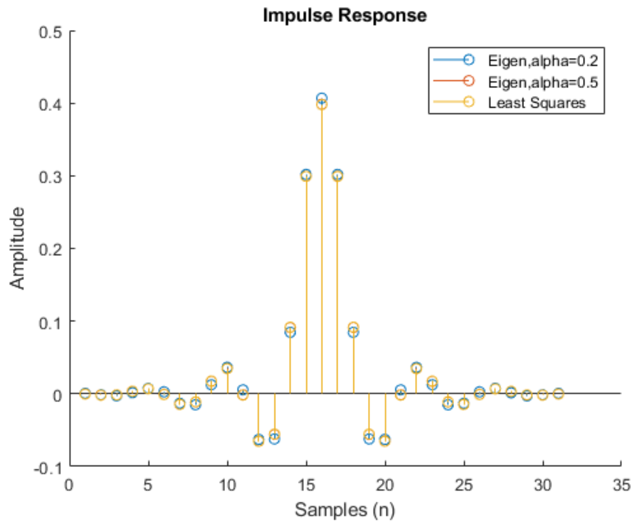
Problem 2

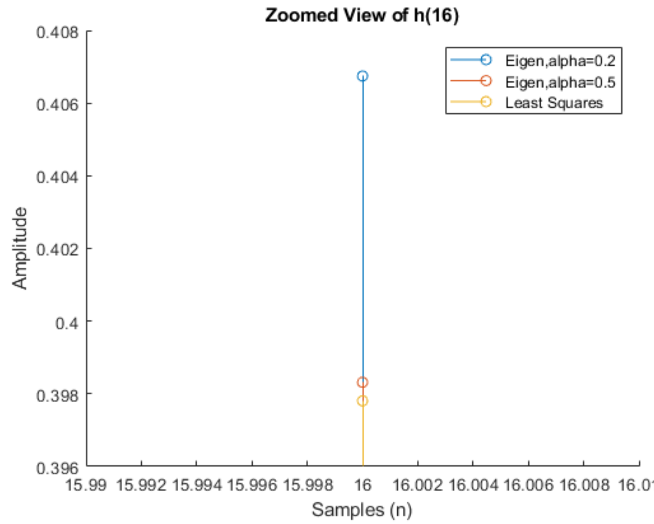
Create a MATLAB script that implements the eigenfilter design procedure. Design a lowpass eigenfilter with the following specifications: ωp=0.3π, ωs=0.5π, N=30 and α=0.2. Plot the impulse response and the frequency response magnitude. Repeat for α=0.5 and compare your filter with the result of using the MATLAB function firls (least-squares FIR design).

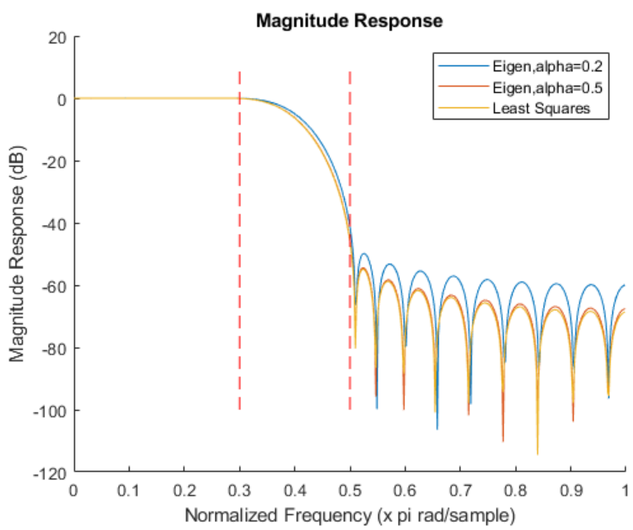
```
% Filter specifications
wp = 0.3*pi;
ws = 0.5*pi;
N = 30;
M = N/2;
alpha = [0.2 0.5];

h = zeros(length(alpha),N+1);

% Compute impulse response for alpha=0.2 and alpha=0.5
for a=1:length(alpha)
    % Compute Ps Matrix
    syms w;
    cw = cos((0:M)*w).';
    cw = cw*cw.';

    Ps = zeros(M+1,M+1);
    w = linspace(ws,pi,5000);

    for m=0:M
        for n=0:M
            % Evaluate cw(m,n) at each value of w
            c = eval(cw(m+1,n+1));

            % Handling cases where c = 1 or 0
            if c==1
                c = ones(length(w),1);
            elseif c==0
                c = zeros(length(w),1);
            end

            % Compute integral
            Ps(m+1,n+1) = (1/pi)*trapz(w,c);
        end
    end

    % Compute Pp Matrix
    syms w;
    cw = cos((0:M)*w).';
    cw = (1 - cw)*(1 - cw).';

    Pp = zeros(M+1,M+1);
    w = linspace(0,wp,5000);

    for m=0:M
        for n=0:M
            % Evaluate cw(m,n) at each value of w
            c = eval(cw(m+1,n+1));

            % Handling cases where c = 1 or 0
            if c==1
```

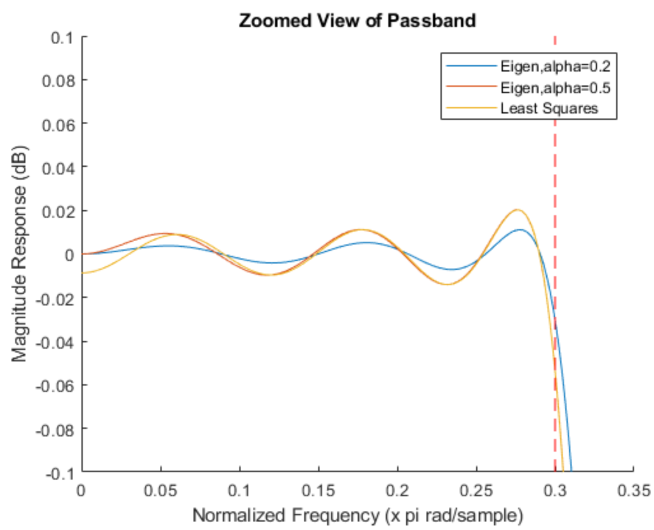
```
                c = ones(length(w),1);
            elseif c==0
                c = zeros(length(w),1);
            end

            % Compute Integral
            Pp(m+1,n+1) = (1/pi)*trapz(w,c);
        end
    end

    % Compute P
    P = alpha(a)*Ps + (1-alpha(a))*Pp;

    % Compute Eigen Vectors/Values of P
    [V,D] = eig(P,'vector');

    % Find index of smallest Eigen value in the Eigen value column vector
    ind = find(D==min(D));

    % Find Eigen Vector containing smallest Eigen value using the index
    b = V(:,ind);

    % Re-organize bn to get h(n)
    % h(M) = b(0)
    h(a,M+1) = b(1);

    % h(n) = b(n)/2 for n = 1 to M
    h(a,M+2:end) = b(2:M+1).'/2;
    h(a,1:M) = flip(b(2:M+1).'/2);

    h(a,:) = h(a,:)/sum(h(a,:));

end


% Least Squares Impulse Reponse and Frequency Response
hls = firls(30,[0 .3 0.5 1],[1 1 0 0]);
[Hls,wls] = freqz(hls);


% Magnitude Response for alpha = 0.2
h_02 = h(1,:);
[H_02,w02] = freqz(h_02);


% Magnitude Response for alpha = 0.5
h_05 = h(2,:);
[H_05,w05] = freqz(h_05);


% Plot Impulse Response for alpha=0.2, alpha=0.5, and
% least squares
figure(7)
hold on
stem(h_02);
stem(h_05);
stem(hls);
title('Impulse Response');
xlabel('Samples (n)'); ylabel('Amplitude');
legend('Eigen,alpha=0.2','Eigen,alpha=0.5','Least Squares');
```

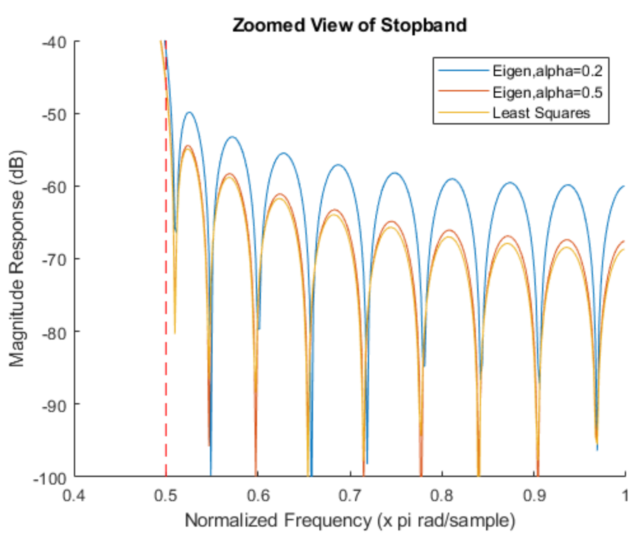
```matlab
% Zoomed view of some of the impulse response values
figure(8)
hold on
stem(h_02);
stem(h_05);
stem(hls);
axis([15.99 16.01 0.396 0.408]);
title('Zoomed View of h(16)');
xlabel('Samples (n)'); ylabel('Amplitude');
legend('Eigen,alpha=0.2','Eigen,alpha=0.5','Least Squares');


% Plot Magnitude Response for alpha=0.2, alpha=0.5, and
% least squares
figure(9);
hold on
p1 = plot(w02/pi,20*log10(abs(H_02)));
p2 = plot(w05/pi,20*log10(abs(H_05)));
p3 = plot(wls/pi,20*log10(abs(Hls)));
title('Magnitude Response');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude Response (dB)');
line([ws/pi ws/pi],[-100 10],'color','red','LineStyle','--');
line([wp/pi wp/pi],[-100 10],'color','red','LineStyle','--');
legend([p1 p2 p3],'Eigen,alpha=0.2','Eigen,alpha=0.5','Least Squares');


% Zoomed view of Magnitude Response (Passband)
figure(10);
hold on
p1 = plot(w02/pi,20*log10(abs(H_02)));
p2 = plot(w05/pi,20*log10(abs(H_05)));
p3 = plot(wls/pi,20*log10(abs(Hls)));
axis([0 0.35 -0.1 0.1]);
title('Zoomed View of Passband');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude Response (dB)');
line([wp/pi wp/pi],[-100 10],'color','red','LineStyle','--');
legend([p1 p2 p3],'Eigen,alpha=0.2','Eigen,alpha=0.5','Least Squares');


% Zoomed view of Magnitude Response (Stopband)
figure(11);
hold on
p1 = plot(w02/pi,20*log10(abs(H_02)));
p2 = plot(w05/pi,20*log10(abs(H_05)));
p3 = plot(wls/pi,20*log10(abs(Hls)));
axis([ws/pi-0.1 1 -100 -40]);
title('Zoomed View of Stopband');
xlabel('Normalized Frequency (x pi rad/sample)'); ylabel('Magnitude Response (dB)');
line([ws/pi ws/pi],[-100 10],'color','red','LineStyle','--');
legend([p1 p2 p3],'Eigen,alpha=0.2','Eigen,alpha=0.5','Least Squares');
```











Published with MATLAB® R2017b