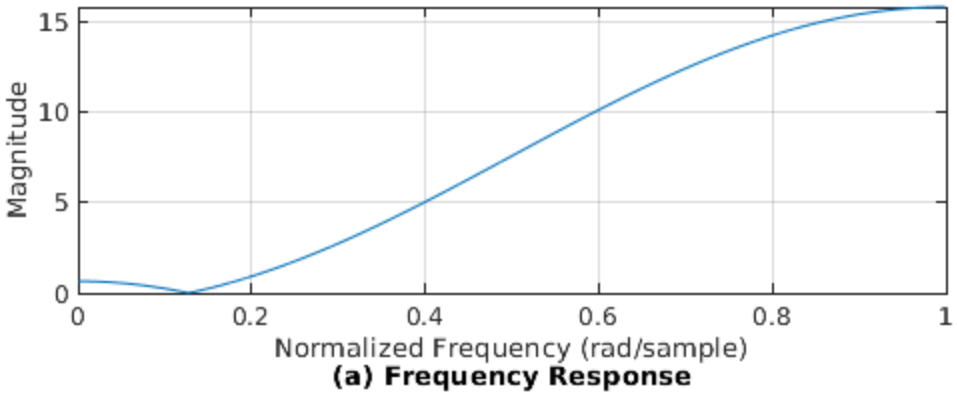
```
% HW11
% Q02

% ---------- reset -----------------
close all;
clear all;
clc;


% ---------- configs ----------------
w1 = 0.1;
w2 = 0.4;

% ------ generate the filter --------
[a b] = get_coeff_filter(w1, w2);  % generate the filter coeffs.


h = [a b a]; % actual 3 point filter coefficients

% ----------- generate the input signals ---------
n = 0: 99; % time index
A = 1; % amplitudes
B = 1;

x = A * cos(w1*n) + B * cos(w2*n); % input signal

% ----------- output = filter(input) -------------
y = filter(h, 1, x);

% ----------- Analyze the filter -----------------


% Compute frequency response of the filter
[H, w] = freqz(h, 1, 1024);

figure(1)
subplot(2,1,1)

% Plot the magnitude response
plot(w/pi, abs(H));
xlabel('Normalized Frequency (rad/sample)');
ylabel('Magnitude');
title('(a) Frequency Response','Units', 'normalized', 'Position',
 [0.5, -0.35, 0]);
grid on;

% Plot the phase response (optional)
subplot(2,1,2);
plot(w/pi, angle(H));
xlabel('Normalized Frequency (rad/sample)');
ylabel('Phase (radians)');
title('Phase Response');
```

```
title('(b) Frequency Response','Units', 'normalized', 'Position',
 [0.5, -0.35, 0]);
grid on;


saveas(gca, 'hw11_2a.eps', 'epsc');


% --------- Analyze I/O signals ---------

% Fourier Transform of Input Signal
N_fft = 1024;
X = fft(x, N_fft); % Compute the FFT of input signal
X = fftshift(X);
frequencies = 0: N_fft-1; % Define the frequency axis (normalized)
frequencies = frequencies/N_fft - 0.5;

frequencies = frequencies * 2*pi;

% Fourier Transform of Output Signal
Y = fft(y, N_fft); % Compute the FFT of output signal
Y = fftshift(Y);

% Magnitudes of the FFTs
magnitude_X = abs(X); % Magnitude of FFT of input signal
magnitude_Y = abs(Y); % Magnitude of FFT of output signal

% Plot the frequency spectrum
figure;

% Input signal spectrum
subplot(2, 1, 1);
plot(frequencies, magnitude_X);
title('(a) Frequency Spectrum of Input
 Signal','Units', 'normalized', 'Position', [0.5, -0.35, 0]);

xlabel('Normalized Frequency');
ylabel('Magnitude');
grid on;
xlim([-0.5,0.5])
xticks(-0.5:0.1:0.5);

% Output signal spectrum
subplot(2, 1, 2);
plot(frequencies, magnitude_Y);
title('(b)Frequency Spectrum of Output
 Signal', 'Units', 'normalized', 'Position', [0.5, -0.35, 0]);
xlabel('Normalized Frequency');
ylabel('Magnitude');
grid on;
xlim([-0.5,0.5])
xticks(-0.5:0.1:0.5);
```

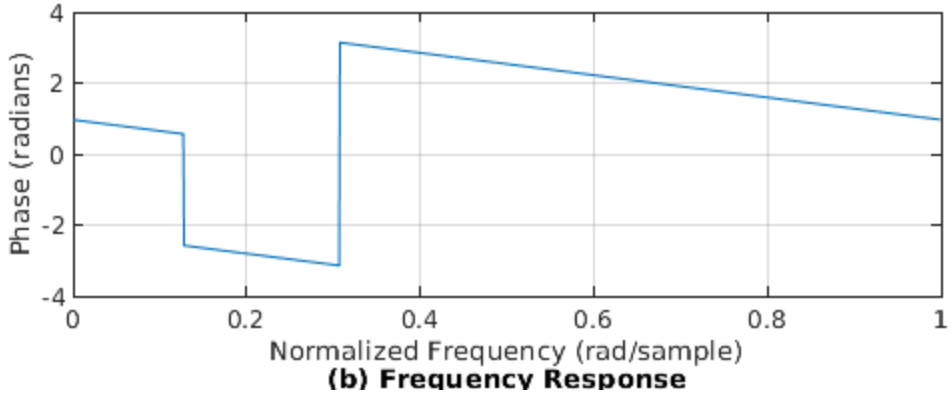
```
saveas(gca, 'hw11_2b.eps', 'epsc');



% ----- plot filter coeff ---------
h = [0 0 a b a 0 0];
n = [-2 -1 0 1 2 3 4];
h_mag = abs(h);
h_ph = angle(h);
figure(3)

subplot(2,1,1)
stem(n, h_mag, 'filled')
for i = 1:length(h)
    if h_mag(i) <= 1e-6
        continue
    end
    text(n(i), h_mag(i), sprintf(' %.2f',
 h_mag(i)), 'VerticalAlignment', 'bottom', 'HorizontalAlignment', 'center');
end
xlim([-2, 4])
ylim([0,10])
xlabel('Filter tap')
ylabel('Magnitude')
grid on
title('(a) Magnitude ','Units', 'normalized', 'Position', [0.5, -0.35,
 0]);


subplot(2,1,2)
stem(n, h_ph, 'filled')
for i = 1:length(h)
    if abs(h_mag(i)) <= 1e-6
        continue
    end
    if h_ph(i) >= 0
        text(n(i), h_ph(i), sprintf(' %.2f',
 h_ph(i)), 'VerticalAlignment', 'bottom', 'HorizontalAlignment', 'center');
    else
        text(n(i), h_ph(i), sprintf(' %.2f',
 h_ph(i)), 'VerticalAlignment', 'top', 'HorizontalAlignment', 'center');
    end
end
xlim([-2, 4])
ylim([-pi,pi])
xlabel('Filter tap')
ylabel('Phase (rads)')
grid on
title('(b) Phase','Units', 'normalized', 'Position', [0.5, -0.35, 0]);


saveas(gca, 'hw11_2c.eps', 'epsc');

% ---------------- functions -------------
```

```
function [a, b] = get_coeff_filter(w1, w2)
    term1 = 1 + exp(-1j * 2* w1);
    term2 = exp(1j*w2) + exp(-1j*w2);

    a = 1/(term1 - term2);
    b = -term2/(term1 - term2);


end
```

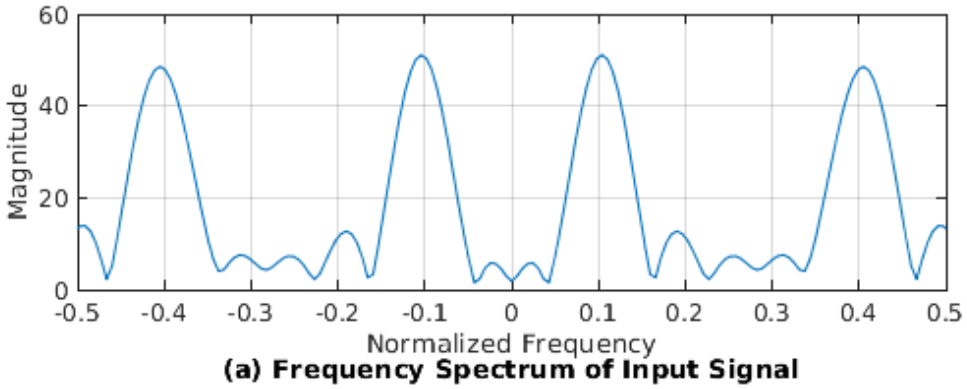
```
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
Couldn't create JOGL canvas--using painters
```



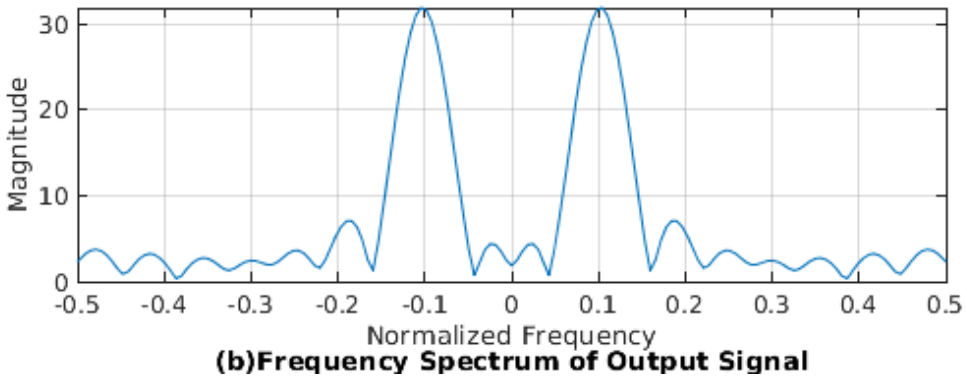
(a) Frequency Response



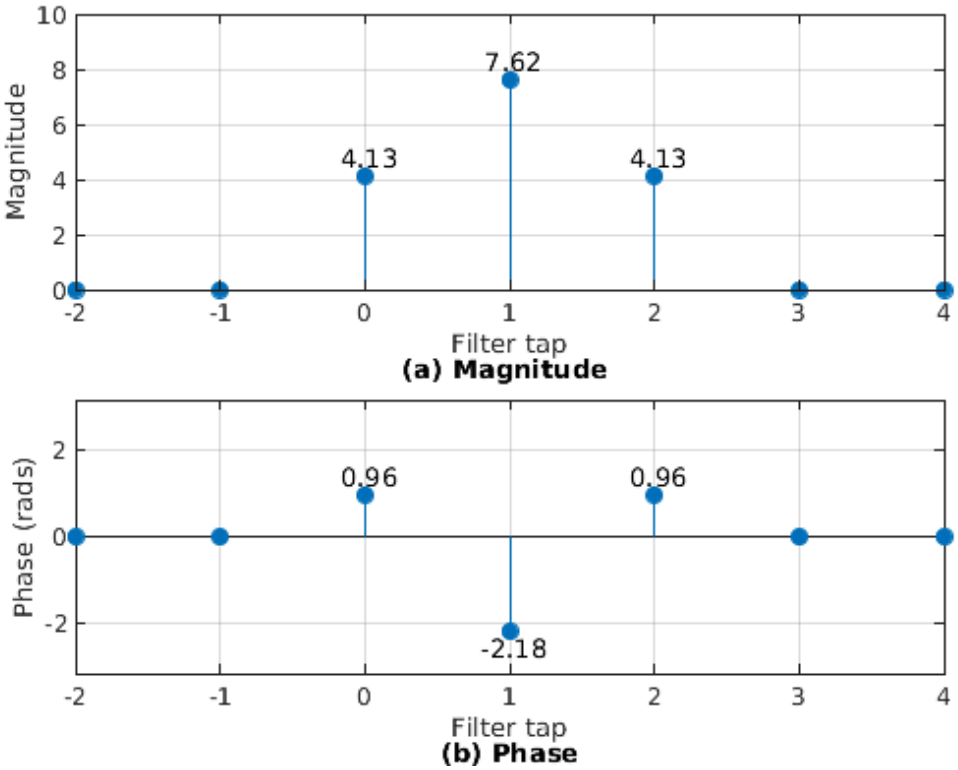
(b) Frequency Response



(a) Frequency Spectrum of Input Signal



(b)Frequency Spectrum of Output Signal



Published with MATLAB® R2018a