BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

Pilani Campus

EEE C434/EEE F434: Digital Signal Processing

**Lab 3: Convolution and DFS**

**Note: Please write your MATLAB codes in this .doc file and save it. Capture and paste the snapshots of your plots, wherever required. Make sure you get it signed before leaving the lab.**

**PART A: Convolution**

Q1) Consider the system below:

y[n]

x[n]

h2[n]

h1[n]

x[n] = {10, -5, 2, 0, 7, 19, 3, 8, 4, 2,10, 12}

h1[n] = {1, 2, 3, 4}

h2[n] = {1, 1,1, 1}

What would be the output sequence y[n]? Plot your result. What are your observations?

x = [10, -5, 2, 0, 7, 19, 3, 8, 4, 2,10, 12];

h1 = [1 2 3 4];

h2 = [1 1 1 1];

w = conv(x,h1);

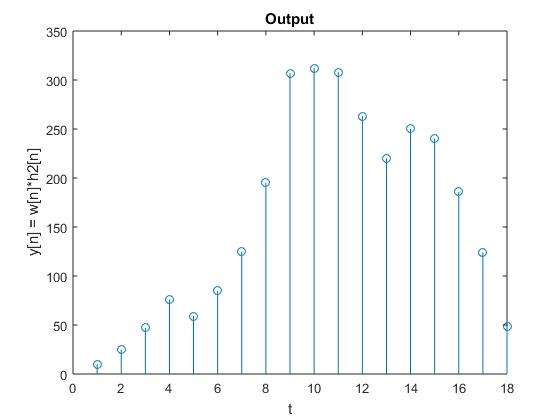
y = conv(w,h2);

stem(y);

title('Output');

xlabel('t');

ylabel('y[n] = w[n]\*h2[n]');



Observations:

subplot(5,1,1);

stem(x);

title('Input');

xlabel('t');

ylabel('x[n]');

hold on;

subplot(5,1,2);

stem(h1);

title('First system');

xlabel('t');

ylabel('h1[n]');

hold on;

subplot(5,1,3);

stem(w);

title('Intermediate result');

xlabel('t');

ylabel('w[n] = x[n]\*h1[n]');

hold on;

subplot(5,1,4);

stem(h2);

title('Second System');

xlabel('t');

ylabel('h2[n]');

hold on;

subplot(5,1,5);

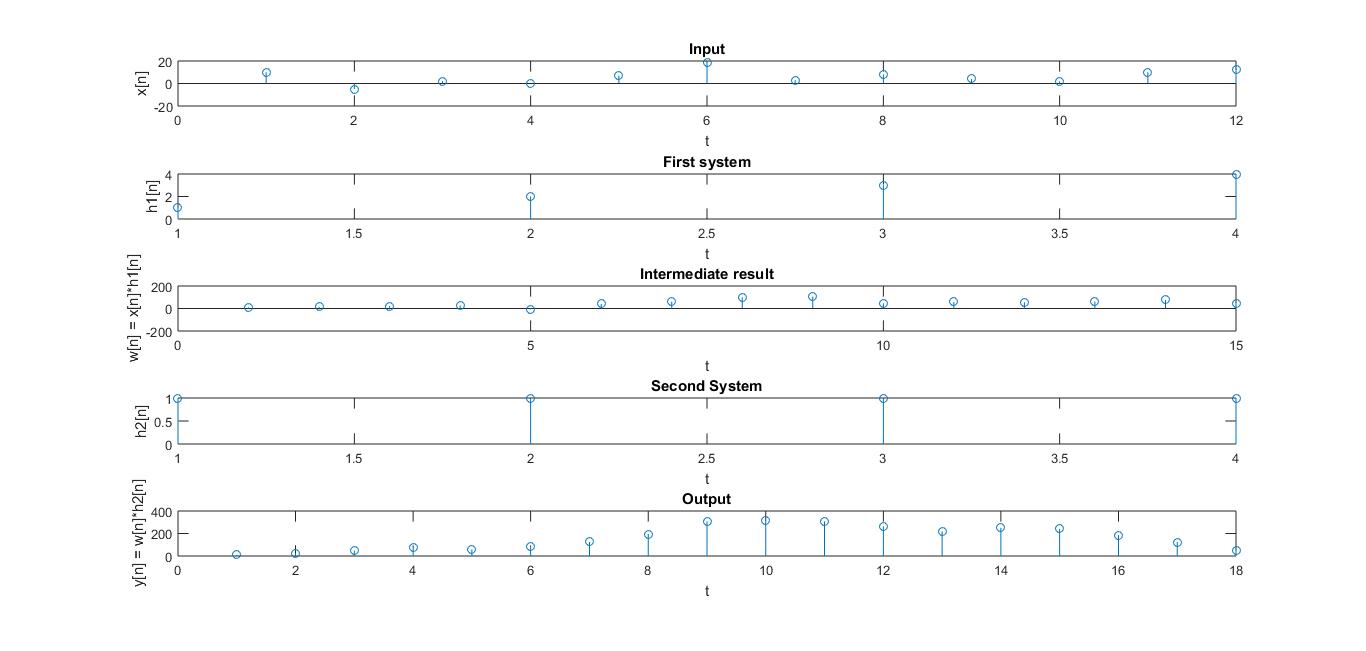
stem(y);

title('Output');

xlabel('t');

ylabel('y[n] = w[n]\*h2[n]');

hold off;



Q2) In a system, the input is an audio signal wave file used in lab2 (‘si1188.wav’). Read the file into MATLAB and assign it as x[n]. The impulse response of the system (h[n]) is a normal distributed sequence with mean 2 and variance 5 (i.e., h[n] ~ N(2,5)). Find the output sequence. Assume the length of the impulse response is 1000. Repeat the similar analysis for the case when h[n] is uniformly distributed between the range [-2 4].

To access the sound file (‘si1188.wav’):

<https://drive.google.com/file/d/0B7-qexRAlXuTUldYOUdPTXFYSW8/view?usp=sharing>

[x, fs] = audioread('si1188.wav');

% Case 1

mean = 2;

var = 5;

len = 1000;

rng(0,'twister');

h1 = mean.\*randn(len,1)+sqrt(var);

y1 = conv(x,h1);

% Case 2

h2 = -2 + (2+4)\*rand(1000,1);

y2 = conv(x,h2);

% Comparison

subplot(3,1,1);

plot(x);

title('input');

ylabel('x[n]');

xlabel('t\*fs');

subplot(3,1,2);

plot(y1);

title('output1');

ylabel('y1[n]');

xlabel('t\*fs');

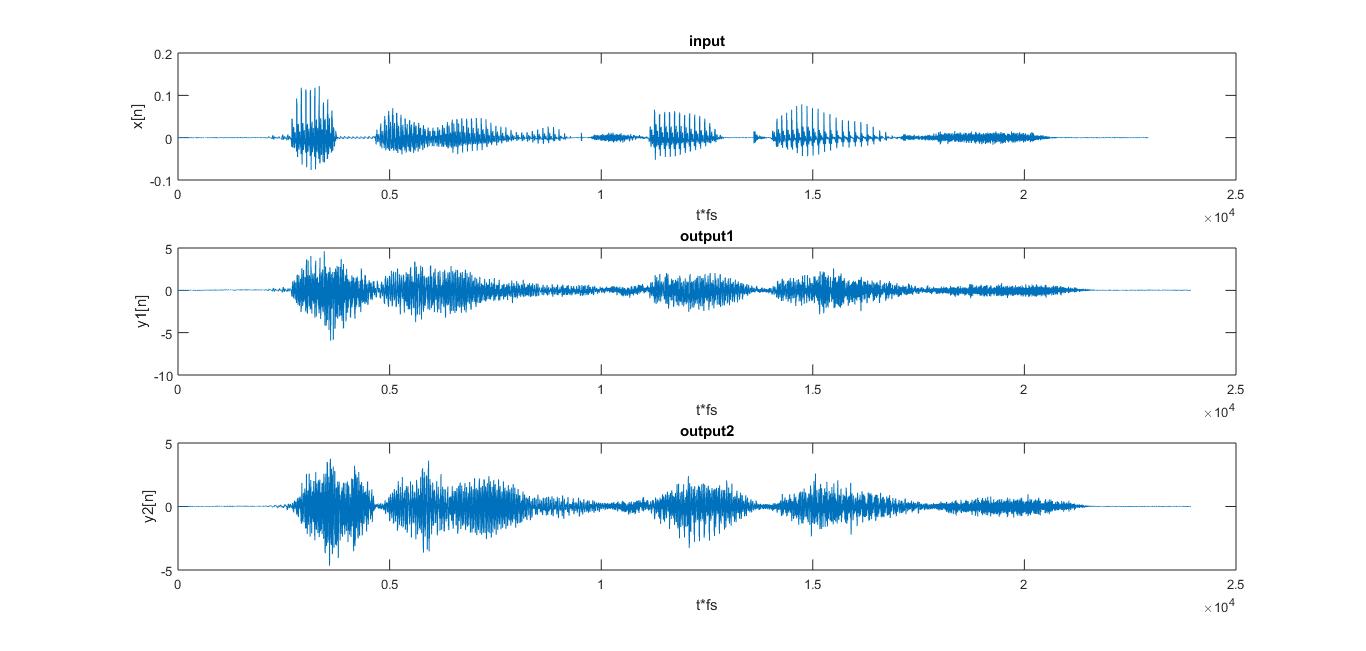
subplot(3,1,3);

plot(y2);

title('output2');

ylabel('y2[n]');

xlabel('t\*fs');



**PART B: DFS**

Q3a) Create a function which accepts a periodic sequence x ̃ [n] and period ‘N’, and returns the Discrete Fourier Series (DFS) coefficients of the periodic signal (**Don’t** **use any in-built function for computing DFS**).

function y = PartB\_1(x,N)

k = 1;

j = sqrt(-1);

y = zeros(1,N);

while k < N+1

m = 1;

while m < N+1

ynew = x(m)\*(exp(-1\*2\*j\*3.14\*(k-1)\*(m-1)/N));

y(k) = y(k) + ynew;

m = m+1;

end

k = k+1;

end

end

Q3b) Plot a Sine sequence of Frequency 10 Hz and Sampling Frequency 100 Hz for 5 periods (i.e., x ̃1[n]).

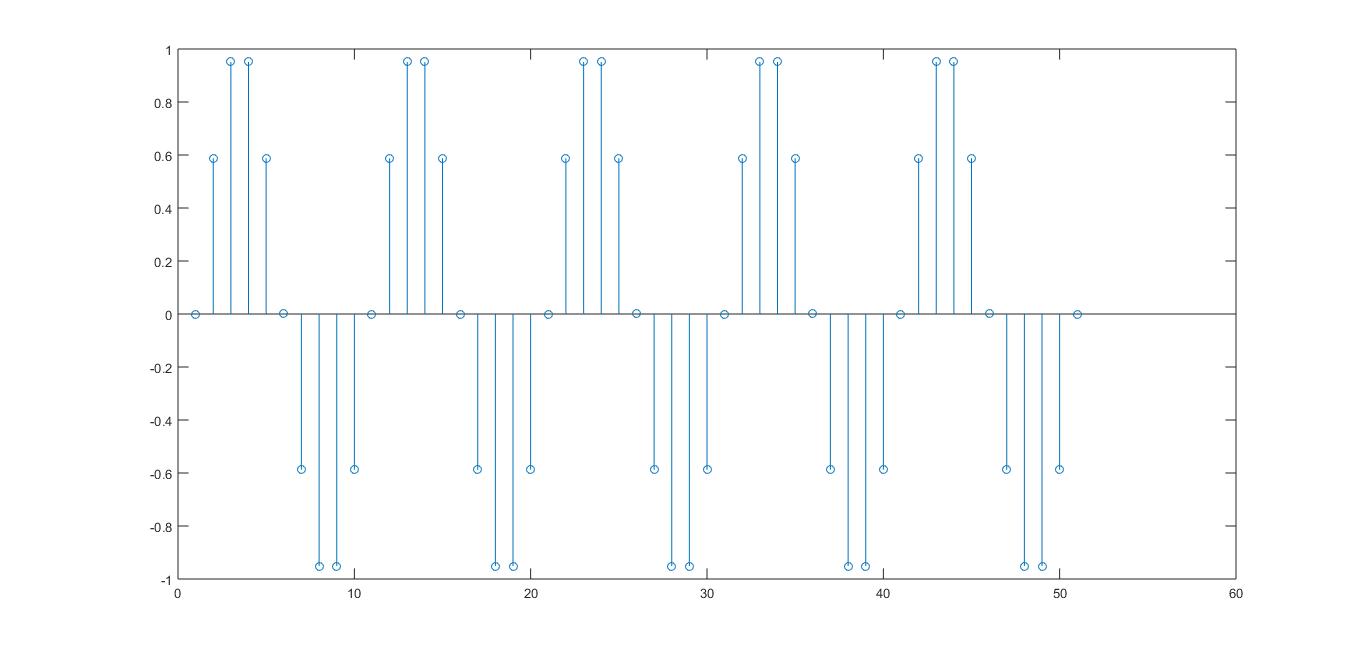
f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x = sin(10\*t);

stem(x);



Q3c) Create an another sine sequence (i.e., x ̃2[n]) and plot for 5 periods. Assume the same sampling frequency as before.

x ̃2[n] = ) ; f=10 Hz

1. Obtain DFS coefficients of x ̃1[n], x ̃2[n] and plot the magnitude spectrum (i.e., |X ̃1[k]| and |X ̃2[k]|).

f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x1 = sin(10\*t);

subplot(4,1,1);

stem(x1);

x2 = sin(10\*t + pi/4);

subplot(4,1,2);

stem(x2);

y1 = PartB\_1(x1,10);

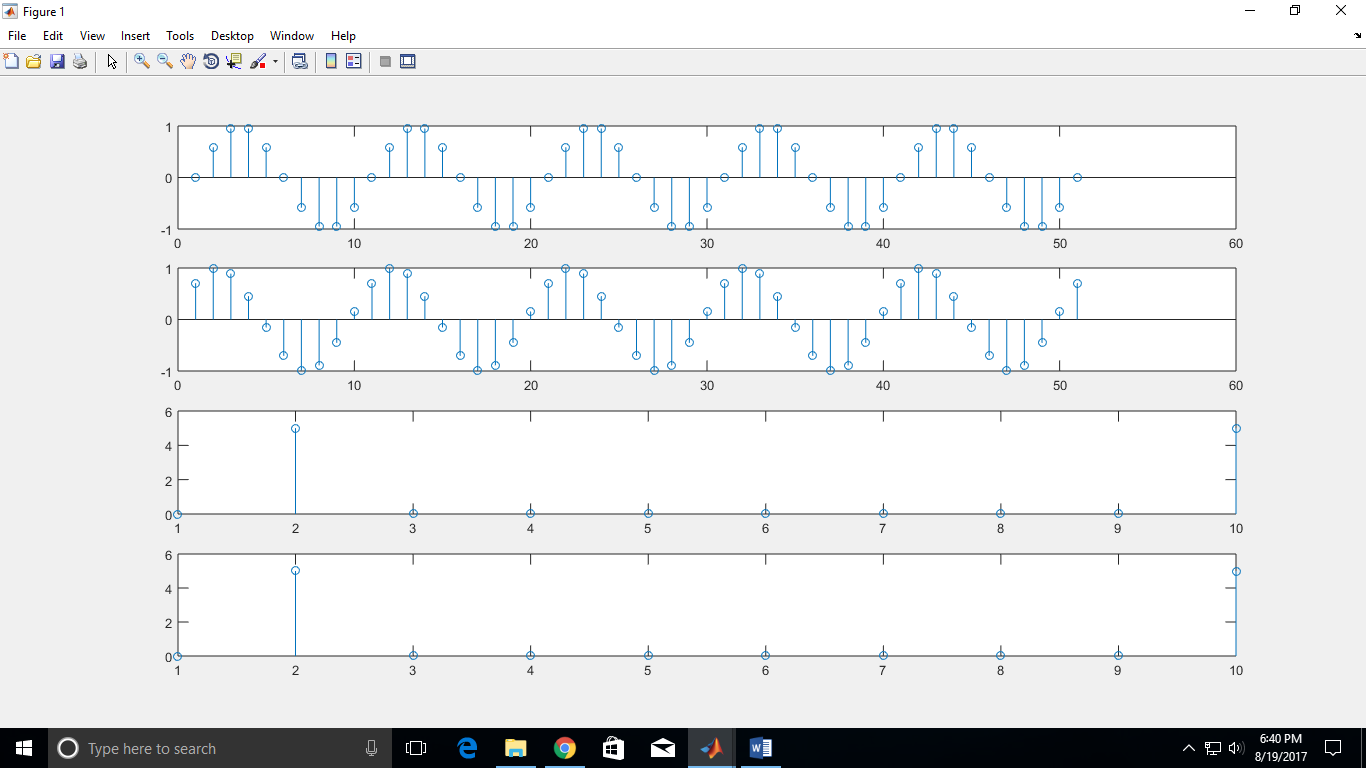
y2 = PartB\_1(x2,10);

subplot(4,1,3);

stem(abs(y1));

subplot(4,1,4);

stem(abs(y2));



1. x ̃3[n] = α x ̃1 [n] + β x ̃2 [n], where α=10, β=5. Plot x ̃3[n] and |X ̃3[k]|.

f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x1 = sin(10\*t);

x2 = sin(10\*t + pi/4);

x3 = 10\*x1 + 5\*x2;

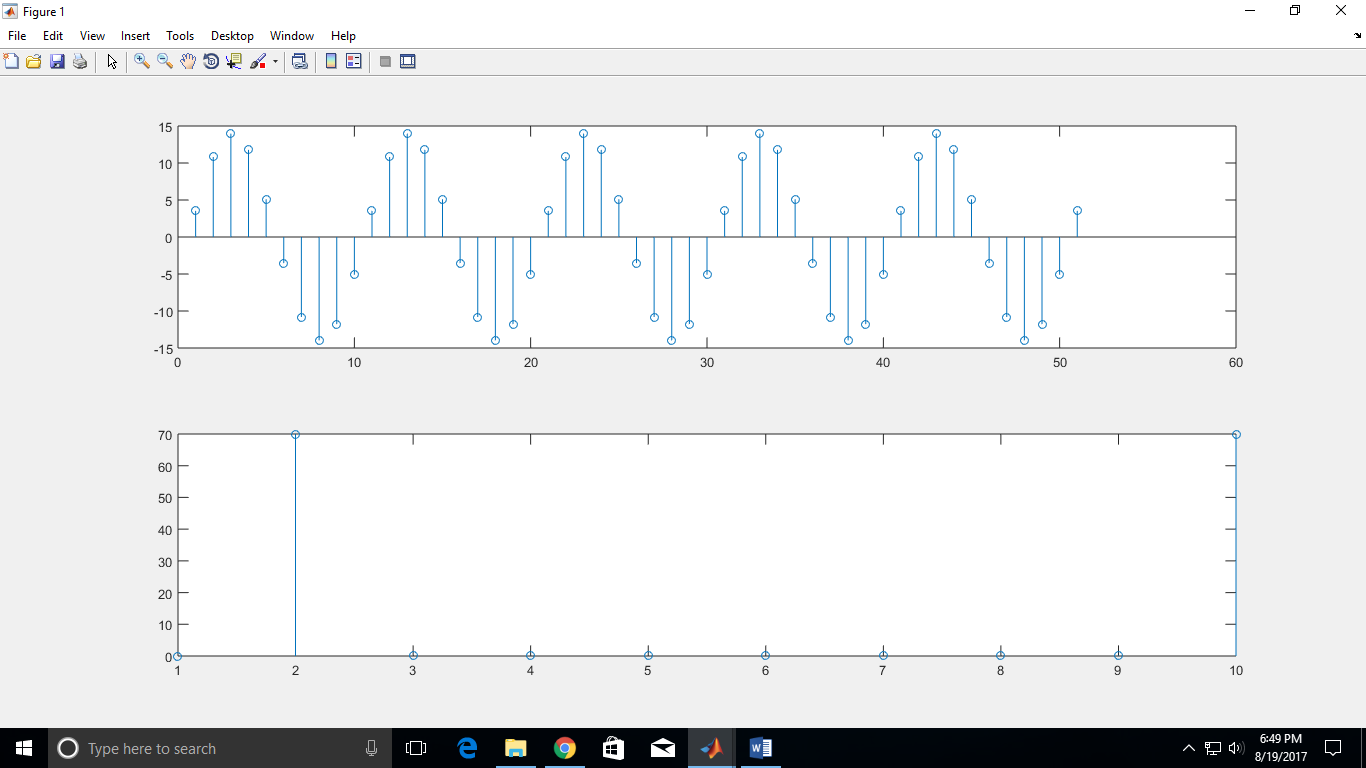
subplot(2,1,1);

stem(x3);

subplot(2,1,2);

mag = PartB\_1(x3,10);

stem(abs(mag));



1. Y ̃[k] = α X ̃1[k] + β X ̃2[k]. Plot |Y ̃[k]| and then compare your results with the above plot from (ii).

f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x1 = sin(10\*t);

x2 = sin(10\*t + pi/4);

x3 = 10\*x1 + 5\*x2;

subplot(2,1,1);

y3 = PartB\_1(x3,10);

stem(abs(y3));

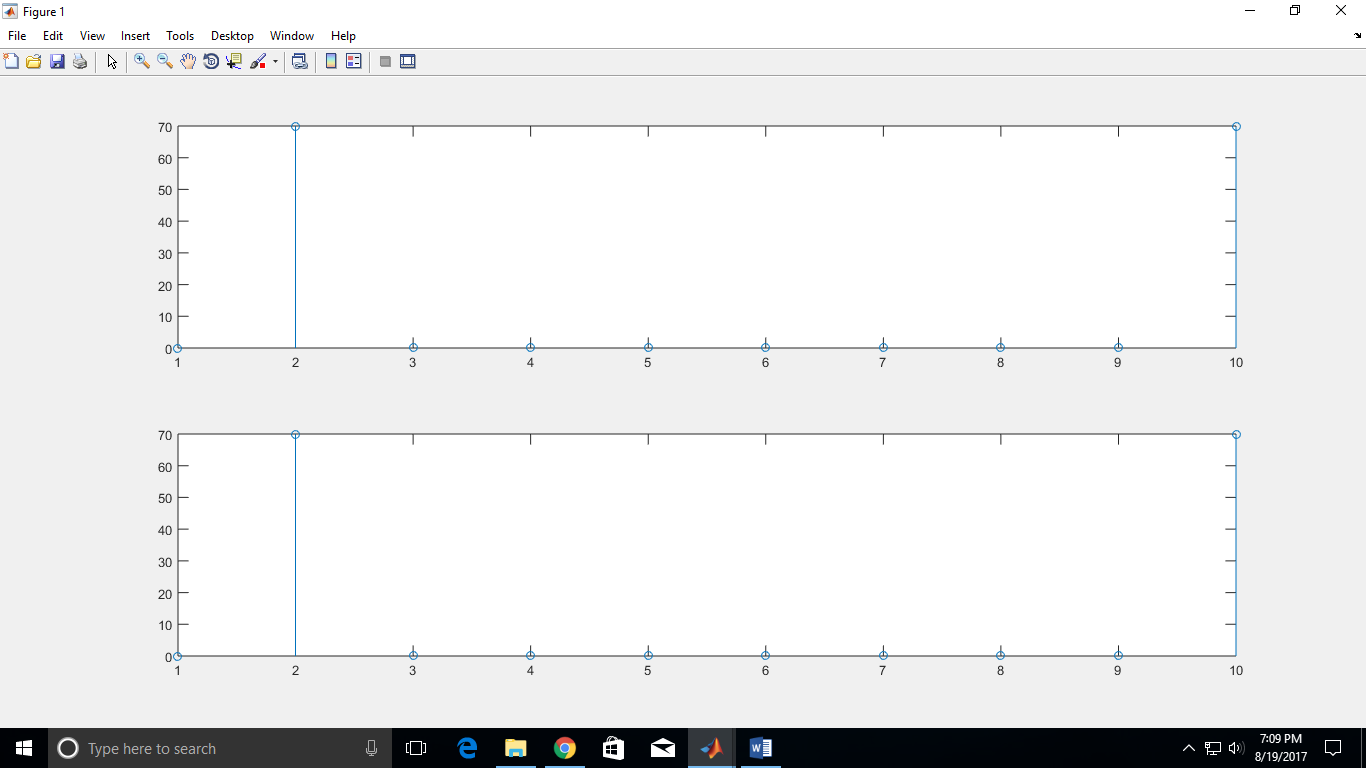
y1 = PartB\_1(x1,10);

y2 = PartB\_1(x2,10);

subplot(2,1,2);

z = 10\*y1 + 5\*y2;

stem(abs(z));



Note: The plots should be arranged as subplots (2 rows and 3 columns) showing x ̃1[n], x ̃2[n], x ̃ 3[n], |X ̃1[k]|, |X ̃2[k]|, |X ̃3[k]|. Another figure should show |X ̃3[k]| and |Y ̃3[k]| using the subplot (2 rows and 1 column).

1. Repeat the similar analyses (i.e., (ii) and (iii)) with the angle responses and using the same format for plotting as mentioned before.

f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x1 = sin(10\*t);

x2 = sin(10\*t + pi/4);

x3 = 10\*x1 + 5\*x2;

subplot(2,1,1);

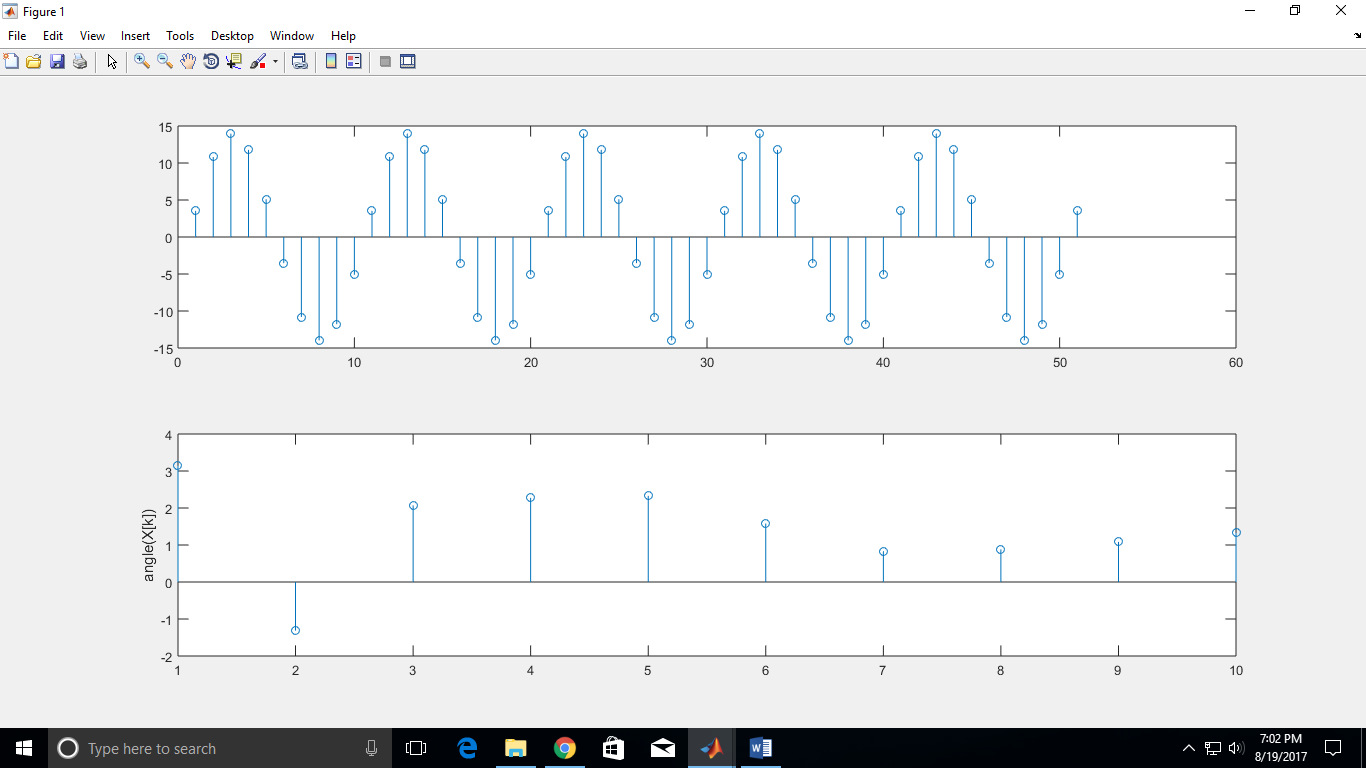
stem(x3);

subplot(2,1,2);

mag = PartB\_1(x3,10);

stem(angle(mag));

ylabel('angle(X[k])');



f = 10;

fs = 100;

t = 0:2\*pi/100:pi;

x1 = sin(10\*t);

x2 = sin(10\*t + pi/4);

x3 = 10\*x1 + 5\*x2;

subplot(2,1,1);

y3 = PartB\_1(x3,10);

stem(angle(y3));

y1 = PartB\_1(x1,10);

y2 = PartB\_1(x2,10);

subplot(2,1,2);

z = 10\*y1 + 5\*y2;

stem(angle(z));

