**ET3270: Signal Processing**

**Name of the student: Chetan Shinde Roll No. 05**

**Div: D Batch: D1**

**Date of performance: 25-01-2023**

**Lab No.3**

**Title: PLOT Basic ELEMENTARY SIGNALS on Matlab**

**AIM :**To plot Multiple signals in same window using subplot and hold on. Also to find the frequency domain transform and phase domain transform using fft(), abss(), angle()

**Learning objectives:**

**Software used:** MATLAB R2022b

**Theory:**

1. **Explaining Elementary Signals** 🡪

In any engineering application, if we want to analyze the characteristics of any system that can be a process or a physical device, we must test or analyze it with a certain set of inputs. But every time it is not possible to apply these directly to that physical device. Hence, we model the physical system into a suitable mathematical model and analyze it with a set of signals that replicate the character of the input we want to apply to this system and record its response by using various mathematical tools such as Fourier transforms, Laplace transforms, eigenfunctions, etc. There are some elementary signals those feature regularly in the analysis of signals and systems. They are namely, impulse signal, unit step signal, sinusoidal signal, exponential signal, ramp signal, parabolic signal, rect, etc.

* **Impulse 🡪** An ideal impulse signal is a signal that is zero everywhere but at the origin (t = 0), it is infinitely high. Although, the area of the impulse is finite. The unit impulse signal is the most widely used standard signal used in the analysis of signals and systems.
* **Unit Step 🡪** The step signal or step function is that type of standard signal which exists only for positive time and it is zero for negative time. In other words, a signal x(t) is said to be step signal if and only if it exists for t > 0 and zero for t < 0.
* **Unit Ramp 🡪** A ramp function or ramp signal is a type of standard signal which starts at 𝑡 = 0 and increases linearly with time. The unit ramp function has unit slop. Widely used in signal processing, the function forms a building block for more complex signals.
* **Sin 🡪** A sine wave is a geometric waveform that oscillates (moves up, down, or side-to-side) periodically, and is defined by the function *x(t) = sin(2piFt)*.Sinusoidal signals are important in both electrical and electronic engineering domains. According to Fourier Series Theory, any

signal (Periodic Signal) can be written in terms of only sine and cosine Signals of different frequencies.

1. **Important Matlab Functions:**

* **clc 🡪** clears all the text from the Command Window, resulting in a clear screen. After running clc , one cannot use the scroll bar in the Command Window to see previously displayed text. We can use the up-arrow key ↑ in the Command Window to recall statements from the command history.
* **clear all 🡪** To clear all variables from the current workspace
* **close all 🡪** clr performs: clear all; close all; clc; This clears your workspace, closes all figures, and clears command window.
* **stem 🡪** stem( X , Y ) plots the data sequence, Y , at values specified by X . The X and Y inputs must be vectors or matrices of the same size. Additionally, X can be a row or column vector and Y must be a matrix with length(X) rows. If X and Y are both vectors, then stem plots entries in Y against corresponding entries in X.

**# The stem functions return plot of each point .**

* **plot 🡪** plot( X , Y ) creates a 2-D line plot of the data in Y versus the corresponding values in X . To plot a set of coordinates connected by line segments, specify X and Y as vectors of the same length. To plot multiple sets of coordinates on the same set of axes, specify at least one of X or Y as a matrix.

**# The plot functions returns joining all the points plotted of each point.**

**Plotting sine wave and cosine wave Using subplot fusnction:**

**Code:-** clc;

clear all;

close all;

F=2;

Fs=100;

Ts=1/Fs;

t=0:Ts:1;

a=sin(2\*pi\*F\*t);

b=cos(2\*pi\*F\*t);

subplot(2,1,1)

plot(a,'r')

xlabel('Time')

ylabel('Amplitude')

title('Sine wave')

subplot(2,1,2)

plot(b,'b')

xlabel('Time')

ylabel('Amplitude')

title('Cosine wave')

**WaveForm:-**

**Chart, line chart

Description automatically generated**

**2) plotting stem() and plot function of two different functions**

**Code:** clc;

clear all;

close all;

F=2;

Fs1=100;

Ts1=1/Fs1;

t1=0:Ts1:1;

a=sin(2\*pi\*F\*t1);

plot(a)

xlabel('Time')

ylabel('Amplitude')

title('Sine wave')

figure;

stem(a)

xlabel('Time')

ylabel('Amplitude')

title('Sine wave')

figure;

F=2;

Fs1=100;

Ts1=1/Fs1;

t2=0:Ts1:2;

b=sin(2\*pi\*F\*t2);

plot(b)

xlabel('Time')

ylabel('Amplitude')

title('Sine wave')

figure;

stem(b)

xlabel('Time')

ylabel('Amplitude')

title('Sine wave')

**Chart, line chart

Description automatically generated**

**Chart, histogram

Description automatically generated**

**Chart, line chart

Description automatically generated**

**Chart

Description automatically generated**

**3) Plotting two function in same Window:-**

**Code:-** clc;

clear all;

close all;

F=2;

Fs=100;

Ts=1/Fs;

t=0:Ts:1;

a=sin(2\*pi\*F\*t);

b=cos(2\*pi\*F\*t);

plot(a)

xlabel('Time')

ylabel('Amplitude')

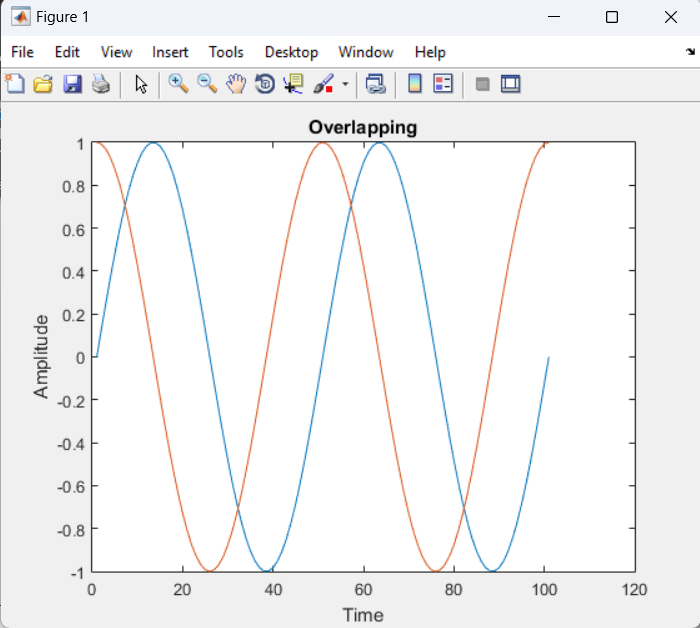
hold on;

plot(b)

xlabel('Time')

ylabel('Amplitude')

title('Overlapping')

**Graph:- **

**4) Plotting frequency transform of normal signal in magnitude and phase transform:-**

**Code:-** clc;

clear all;

close all;

F=2;

Fs=100;

Ts=1/Fs;

t=0:Ts:1;

a=sin(2\*pi\*F\*t);

afft=fft(a);

afftmag=abs(afft);

afftphase=angle(afft);

subplot(3,1,1);

plot(a,'.');

xlabel('Time');

ylabel('Amplitude');

title('Sine wave');

subplot(3,1,2);

plot(afftmag,'g');

xlabel('Frequency');

ylabel('Magnitude');

title('Magnitude');

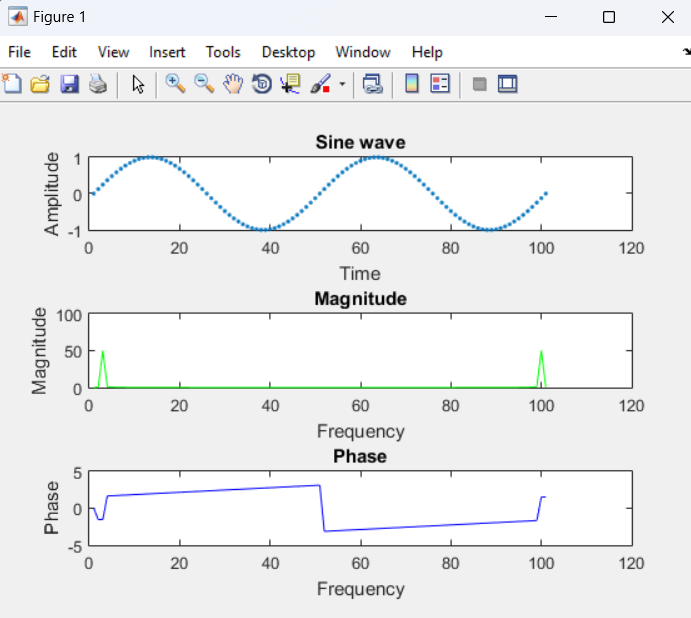
subplot(3,1,3);

plot(afftphase,'b');

xlabel('Frequency');

ylabel('Phase');

title('Phase');

**Graph:- **

**Conclusion:**

After plotting the graphs we learnt to plot the multiple graphs in same window. The frequency transform, magnitude and phase transform shows output of corresponding graphs in same window.