INSTITUTE OF ENGINEERING

ADVANCED COLLEGE OF ENGINEERING AND MANAGEMENT

KALANKI, KATHMANDU

(AFFILIATED TO TRIBHUVAN UNIVERSITY)

LAB REPORT

SUBJECT: DIGITAL SIGNAL ANALYSIS AND PROCESSING (DSAP)

LAB NO: 01

SUBMITTED BY:

NAME: DIPESH DHUNGANA

ROLL NO: ACE077BCT035

DATE: MAY 13, 2024

SUBMITTED TO:

DEPARTMENT OF COMPUTER AND ELECTRONICS

TITLE: SIGNAL GENERATION AND MANIPULATION WITH INTRODUCTION TO DIGITAL SIGNAL PROCESSING (DSP) TOOLS.

OBJECTIVE:

TO GENERATE DIFFERENT SIGNALS AND MANIPULATE IT WITH THE HELP OF DIGITAL SIGNAL PROCESSING (DSP) TOOLS.

THEORY:

MATLAB

MATLAB (Matrix Laboratory) is a powerful software environment widely used in digital signal analysis and processing. It offers a high-level programming language and an extensive suite of built-in functions specifically designed for numerical computation, visualization, and application development. MATLAB excels in handling matrix and vector operations, making it ideal for processing and analyzing complex signals. Its intuitive interface and versatile toolboxes simplify tasks such as filtering, Fourier analysis, and signal visualization, making it an essential tool for engineers and researchers in signal processing.

Command Window in MATLAB:

This is the main area where you can interact with MATLAB directly. Here, you can enter commands, run scripts, and see the results immediately. It is useful for quick computations and testing small code snippets.

Editor Window in MATLAB:

This window is used for writing, editing, and debugging scripts and functions. It provides features like syntax highlighting, code folding, and error checking, which help in developing and managing more complex code efficiently.

In the MATLAB Command Window, you can enter commands directly to perform computations, execute scripts, manage files, and more. Here are some commonly used **MATLAB commands:**

Variable Management (Assigning Values)

```
x = 10;
y = [1, 2, 3, 4, 5]; % Row vector
z = [1; 2; 3; 4; 5]; % Column vector
```

Arithmetic Operations

- a = 5 + 3; % Addition
- b = 10 - 2; % Subtraction
- c = 4 * 3; % Multiplication
- d = 8 / 2; % Division
- e = 2^3; % Exponentiation

Matrix Operations

- A = [1, 2; 3, 4]; % Define a matrix
- B = [5, 6; 7, 8];
- C = A + B; % Matrix addition
- D = A * B; % Matrix multiplication
- E = A .* B; % Element-wise multiplication

Complex Number Operations

- z1 = 3 + 4i; % Using 'i'
- z2 = 2 - 5j; % Using 'j'
- z3 = z1 + z2; % Addition
- z4 = z1 - z2; % Subtraction
- z5 = z1 * z2; % Multiplication
- z6 = z1 / z2; % Division
- realPart = real(z1); % Extracts the real part
- imagPart = imag(z1); % Extracts the imaginary part
- magnitude = abs(z1); % Calculates the magnitude
- phase = angle(z1); % Calculates the phase angle in radians
- conjZ = conj(z1); % Calculates the complex conjugate
- expZ = exp(z1); % Calculates the complex exponential

Basic MATLAB Commands (Examples)

- ➔ a = 2
- ➔ b = 3
- ➔ a + b

  ans = 5

- ➔ c = a+b
- ➔ c

  c = 5

- ➔ a-b

  ans = -1

- ➔ a*b

  ans = 6

- ➔ a/b

  ans = 1.5

- ➔ a\b

  ans = 0.6667

- ➔ a^b

  ans = 8

- ➔ b^a

  ans = 9

- ➔ z = 1+1*i
- ➔ real(z)

  ans = 1

- ➔ imag(z)

  ans = 1

- ➔ conj(z)

  ans = 1.00-1.00*i

- ➔ abs(z)

  ans = 1.4142

- ➔ angle(z)

  ans = 0.7854

- ➔ degtorad(z)

  ans = 0.0175 + 0.175 * i

- ➔ radtodeg(z)

  ans = 45

- ➔ a = [2 1; 3 4]

  $a = \begin{matrix} 2 & 1 \\ 3 & 4 \end{matrix}$

- ➔ b = [4 1; 6 2]

$b = \begin{matrix} 4 & 1 \\ 6 & 2 \end{matrix}$

➔ a+b

$ans = \begin{matrix} 6 & 2 \\ 9 & 6 \end{matrix}$

➔ a-b

$ans = \begin{matrix} -2 & 0 \\ 3 & 2 \end{matrix}$

➔ a*b

$ans = \begin{matrix} 14 & 4 \\ 36 & 11 \end{matrix}$

➔ a/b

$ans = \begin{matrix} -1 & 1 \\ -9 & 6.5 \end{matrix}$

➔ a’

$ans = \begin{matrix} 2 & 1 \\ 3 & 4 \end{matrix}$

➔ b’

$ans = \begin{matrix} 4 & 6 \\ 1 & 2 \end{matrix}$

Signal: A Signal refers to a quantity or function that can vary over time or in space. Signals can be categorized into various types. Example: audio signal, video signal, radio signal, etc.

Continuous signal: A signal that can take any value within a certain range and can change at any point in time is called a continuous signal. They are generally represented as $x(t)$.

Discrete signal: A signal that is only defined at specific points in time is called a discrete signal. It is generally represented as $x[n]$.

Sine wave: It is a periodic signal that is fundamental in may areas of science and engineering. Mathematically, it is represented as:

$x(t) = A\, sin(2\pi f t + \emptyset)$

Or,

$x[n] = A\sin(2\pi f nT + \emptyset)\ where, T = Sampling\ Time = \frac{1}{f_s}$

Cosine wave: A cosine wave is a type of periodic wave that is mathematically described by the cosine function. Like the sine wave, the cosine wave is fundamental in various fields, including physics, engineering, and signal processing.

Mathematically. it is represented as:

$x(t) = A\, cos(2\pi f t + \emptyset)$

Or,

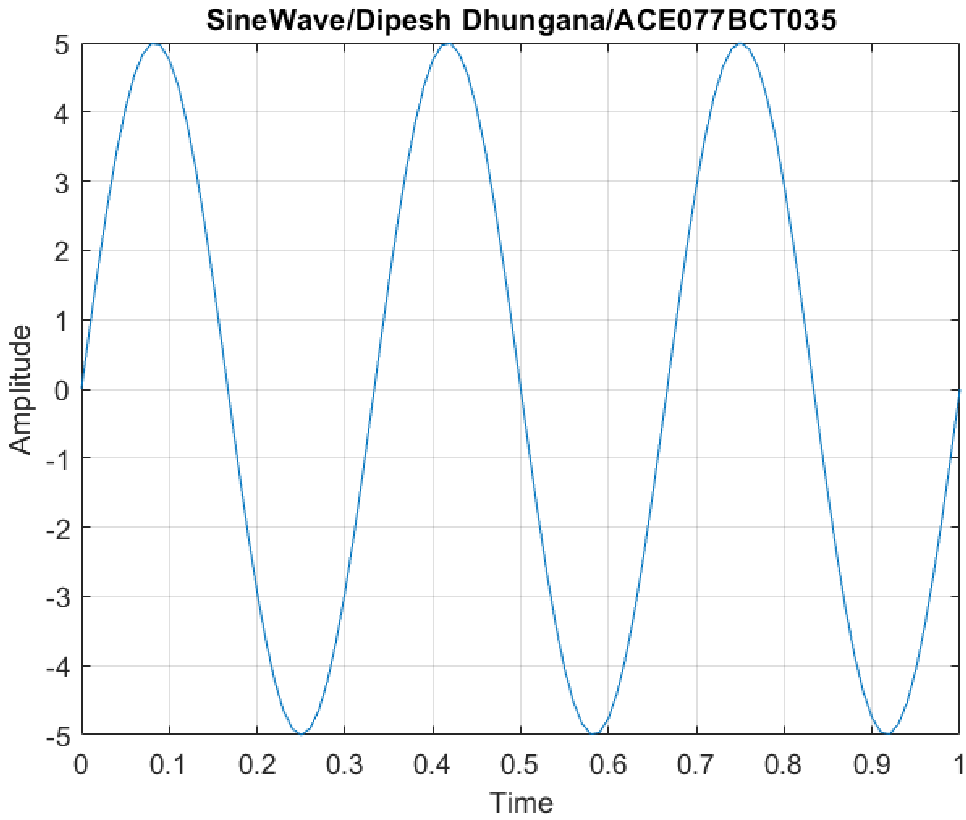
$x[n] = A\, cos\ (2\pi f nT + \emptyset)\ where, T = Sampling\ Time = \frac{1}{f_s}$

Q1. Construct a continuous sine wave having amplitude 5 units and frequency size 3HZ. Plot the signal.

<u>Source code (Continuous Sine Wave)</u>

```
clc;
close all;
clear all;
a=5;
f=3;
t=0:0.01:1;
x=a*sin(2*pi*f*t);
plot(t,x);
xlabel('Time');
ylabel('Amplitude');
title('SineWave/Dipesh Dhungana/ACE077BCT035');
grid on;
```

<u>Output (Continuous Sine Wave)</u>

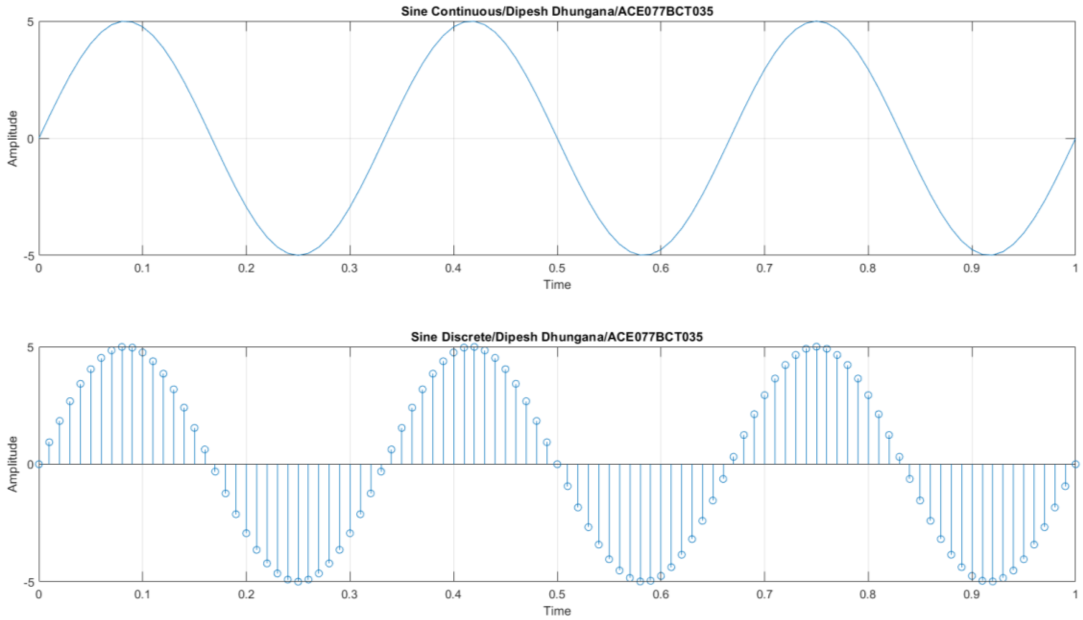


Q2. Construct both continuous and discrete sine wave having amplitude 5 units and frequency 3HZ. Plot the signal in one single screen.

<u>Source code (Continuous and Discrete Sine Wave)</u>

```
clc;
close all;
clear all;
a=5;
f=3;
t=0:0.01:1;
x=a*sin(2*pi*f*t);
subplot(2,1,1)
plot(t,x)
xlabel('time');
ylabel('amplitude')
title('Sine Continuous/Dipesh Dhungana/ACE077BCT035');
grid on;
subplot(2,1,2)
stem(t,x)
xlabel('time');
ylabel('amplitude')
title('Sine Discrete/Dipesh Dhungana/ACE077BCT035');
grid on;
```

Output (Continuous and Discrete Sine Wave)



Q3. Construct both continuous and discrete cosine wave having amplitude 5 units and frequency 3HZ. Plot the signal in one single screen.

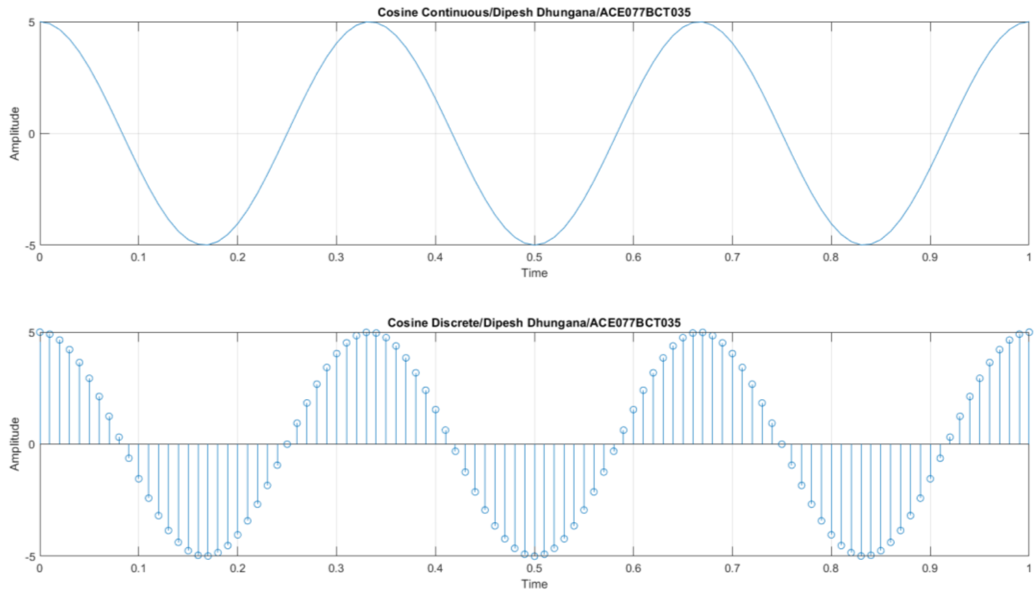
Source code (Continuous and Discrete Cosine Wave)

```
clc;
close all;
clear all;
a=5;
f=3;
t=0:0.01:1;
y=a*cos(2*pi*f*t);
subplot(2,1,1)
plot(t,y)
xlabel('Time');
ylabel('Amplitude')
title('Cosine Continuous/Dipesh Dhungana/ACE077BCT035');
grid on;
subplot(2,1,2)
stem(t,y)
xlabel('Time');
ylabel('Amplitude')
```

title('Cosine Discrete/Dipesh Dhungana/ACE077BCT035');

grid on;

Output (Continuous and Discrete Cosine Wave)



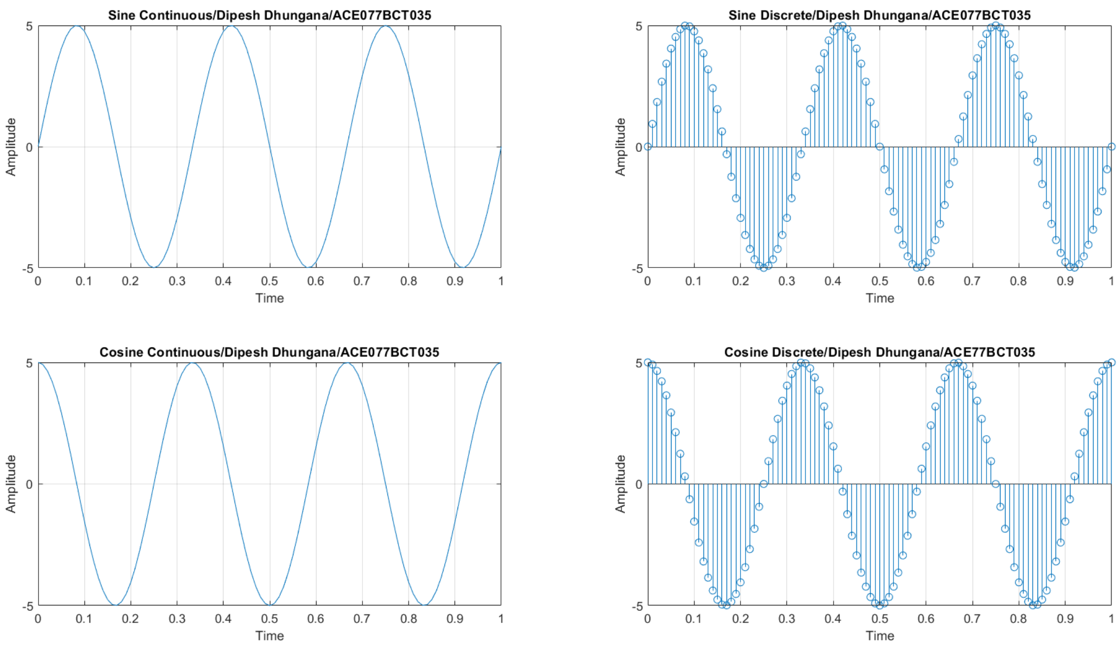
Q4. Construct both continuous and discrete sine and cosine wave having amplitude 5 unit and frequency size 3HZ. Plot all these in one screen.

Source code (Continuous and Discrete Sine and Cosine Wave)

```
clc;
close all;
clear all;
a=5;
f=3;
t=0:0.01:1;
x=a*sin(2*pi*f*t);
y=a*cos(2*pi*f*t);
subplot(2,2,1)
plot(t,x)
xlabel('Time');
ylabel('Amplitude')
title('Sine Continuous/Dipesh Dhungana/ACE077BCT035');
grid on;
subplot(2,2,2)
```

```
stem(t,x)
xlabel('Time');
ylabel('Amplitude')
title('Sine Discrete/Dipesh Dhungana/ACE077BCT035');
grid on;
subplot(2,2,3)
plot(t,y)
xlabel('Time');
ylabel('Amplitude')
title('Cosine Continuous/Dipesh Dhungana/ACE077BCT035');
grid on;
subplot(2,2,4)
stem(t,y)
xlabel('Time');
ylabel('Amplitude')
title('Cosine Discrete/Dipesh Dhungana/ACE77BCT035');
grid on;
```

<u>Output (Continuous and Discrete Sine and Cosine Wave)</u>



DISCUSSION AND CONCLUSION:

In this lab, we explored signal generation and manipulation using MATLAB. By creating and visualizing sine and cosine waves in both continuous and discrete forms, we demonstrated MATLAB's capabilities in handling complex signal processing tasks. Continuous signals, which are defined at every point in time, and discrete signals, which are defined only at specific intervals, were both generated and analyzed. This distinction is crucial for understanding analog and digital signal processing. The lab utilized MATLAB commands and functions to generate and plot these signals, highlighting the software's ease of use and powerful visualization tools.

This lab exercise effectively showcased the generation and manipulation of signals using MATLAB. It reinforced our understanding of continuous and discrete signals, which is fundamental for digital signal processing. Through hands-on practice, we became more proficient with MATLAB, enhancing our ability to perform complex signal processing tasks. The skills learned in this lab are directly applicable to various fields, including telecommunications and audio engineering. Overall, this lab provided a solid foundation in digital signal processing using MATLAB, preparing us for more advanced applications in engineering and research.