**AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH**

# Faculty of Engineering

## Laboratory Report Cover Sheet

***Students must complete all details except the faculty use part.***

Please submit all reports to your subject supervisor or the office of the concerned faculty.

Laboratory Title: Study of signal frequency, spectrum, bandwidth, and quantization using MATLAB

Experiment Number: 2   Due Date: 12/10/2022   Semester: Fall 2022-23

Subject Code: COE3103   Subject Name: Data Communication   Section: K

Course Instructor: Dr. Shuvra Mondal   Degree Program: B.Sc. in CSE

### Declaration and Statement of Authorship:

1. I/we hold a copy of this report, which can be produced if the original is lost/ damaged.
2. This report is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgement is made.
3. No part of this report has been written for me/us by any other person except where such collaboration has been authorized by the lecturer/teacher concerned and is clearly acknowledged in the report.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the School for review and comparison, including review by external examiners.

### I/we understand that

7. Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.
8. Enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy your work

Group Number (if applicable): 01   ☐ Individual Submission   ☐ Group Submission

| No. | Student Name | Student Number | Student Signature | Date |
|---|---|---|---|---|
| **Submitted by:** | | | | |
| **1** | Md. Monzurul Kabir Zhanda (CSE) | 20-43495-1 | | |
| **Group Members:** | | | | |
| **2** | Jarin Tasnim Monisha (SE) | 16-32185-2 | | |
| **3** | Muhammad Shahriar Zaman (CSE) | 20-41840-1 | | |
| **4** | Md. Fahad Khan (CSE) | 20-43328-1 | | |
| **5** | H. M. Saifullahil Mazid (CSE) | 20-44018-2 | | |
| **6** | Sandip Misra(CSE) | 21-44506-1 | | |

***For faculty use only:***

**Total Marks: _______   Marks Obtained: _______**

**Faculty comments**____________________________________________________________

# American International University- Bangladesh

## Department of Computer Engineering

Data Communication Laboratory

**Title of the experiment:** Study of signal frequency, spectrum, bandwidth, and quantization using MATLAB.

**Abstract:** This lab experiment is aimed at understanding how to use MATLAB for solving communication engineering problems and to develop the understanding of MATLAB environment, syntax and commands.

## Introduction:

**I. Frequency:** The frequency of a wave describes how many waves go past a certain point in one second. Frequency is measured in Hertz (usually abbreviated Hz), and can be calculated using the formula:

$V = f\lambda$

where V is the velocity of the wave (in $ms^{-1}$), f is the frequency of the wave (in Hz), and $\lambda$ (the Greek letter lambda) is the wavelength of the wave (distance from one peak / trough to the next, in m). Frequency is the rate of change with respect to time. Change in a short span of time means high frequency. Change over a long span of time means low frequency.

**II. Spectrum:** Usually we represent signals in time domain. But signals can be represented in frequency domain as well. When signals are represented in frequency domain they are called spectrum.

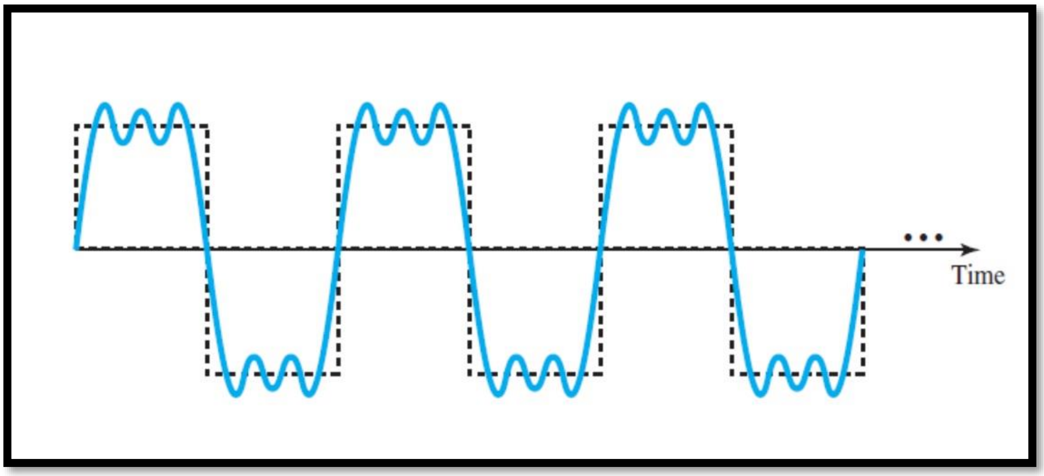


**Fig.: A composite periodic signal**

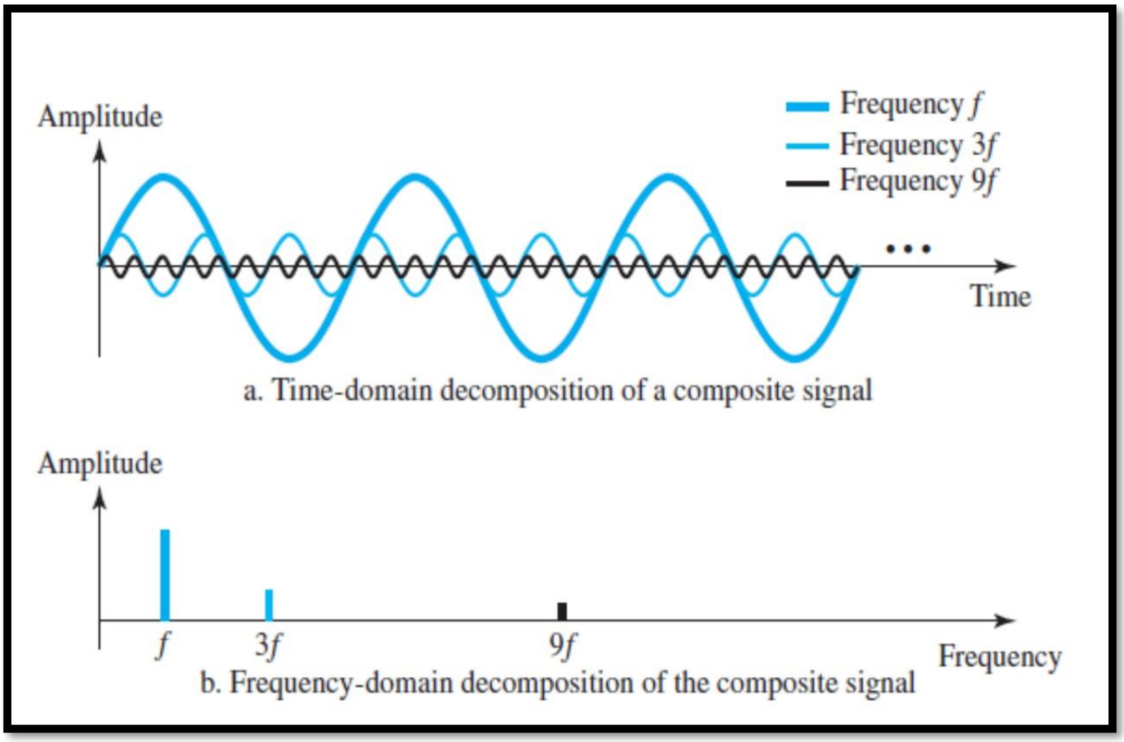


**Fig.: Decomposition of a composite periodic signal in the time and frequency domains**

**III. Bandwidth:** Bandwidth is the range of frequency a signal contains in it. If a composite signal is made up of multiple sinusoids of 100, 250, 300, and 400 Hz. Then its bandwidth is the difference of the highest and lowest frequency components. So here the bandwidth of the signal is (400-100) = 300 Hz.

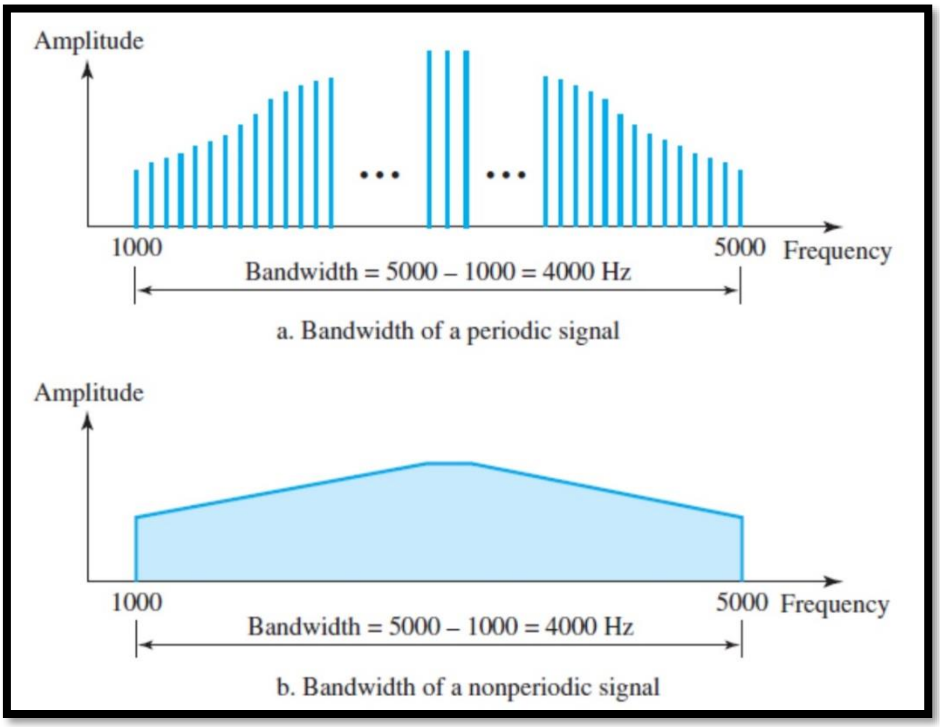


**Fig.: Bandwidth**

**IV. Quantization:** The digitization of analog signals involves the rounding off of the values which are approximately equal to the analog values. The method of sampling chooses a few points on the analog signal and then these points are joined to round off the value to a near stabilized value. Such a process is called as Quantization.

Quantizing an Analog Signal:

The analog-to-digital converters perform this type of function to create a series of digital values out of the given analog signal. The following figure represents an analog signal. This signal to get converted into digital has to undergo sampling and quantizing.

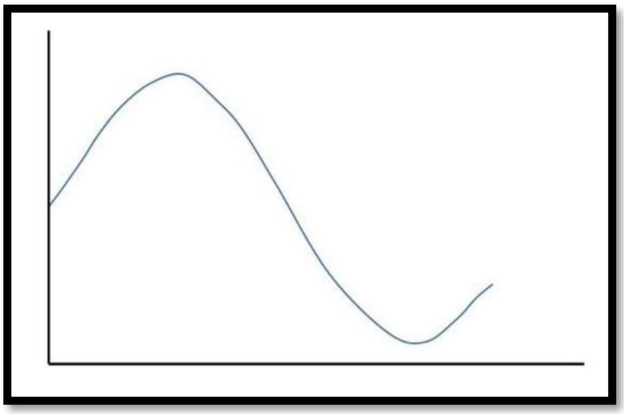

**Fig.: Example of an analog signal**

The quantizing of an analog signal is done by discretizing the signal with a number of quantization levels. Quantization is representing the sampled values of the amplitude by a finite set of levels, which means converting a continuous-amplitude sample into a discrete-time signal.

Both sampling and quantization result in the loss of information. The quality of a Quantizer output depends upon the number of quantization levels used. The discrete amplitudes of the quantized output are called as representation levels or reconstruction levels. The spacing between the two adjacent representation levels is called a quantum or step-size.

The following figure shows the resultant quantized signal which is the digital form for the given analog signal.

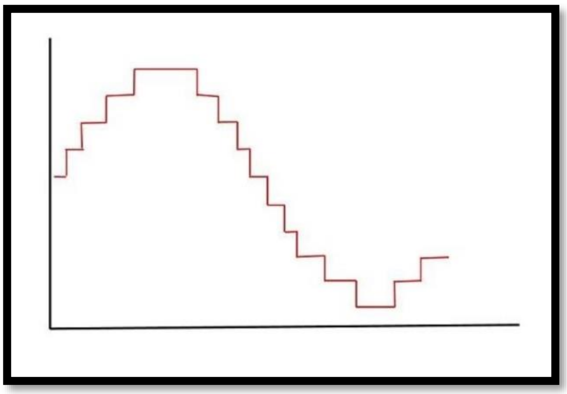

**Fig.: A quantized signal**

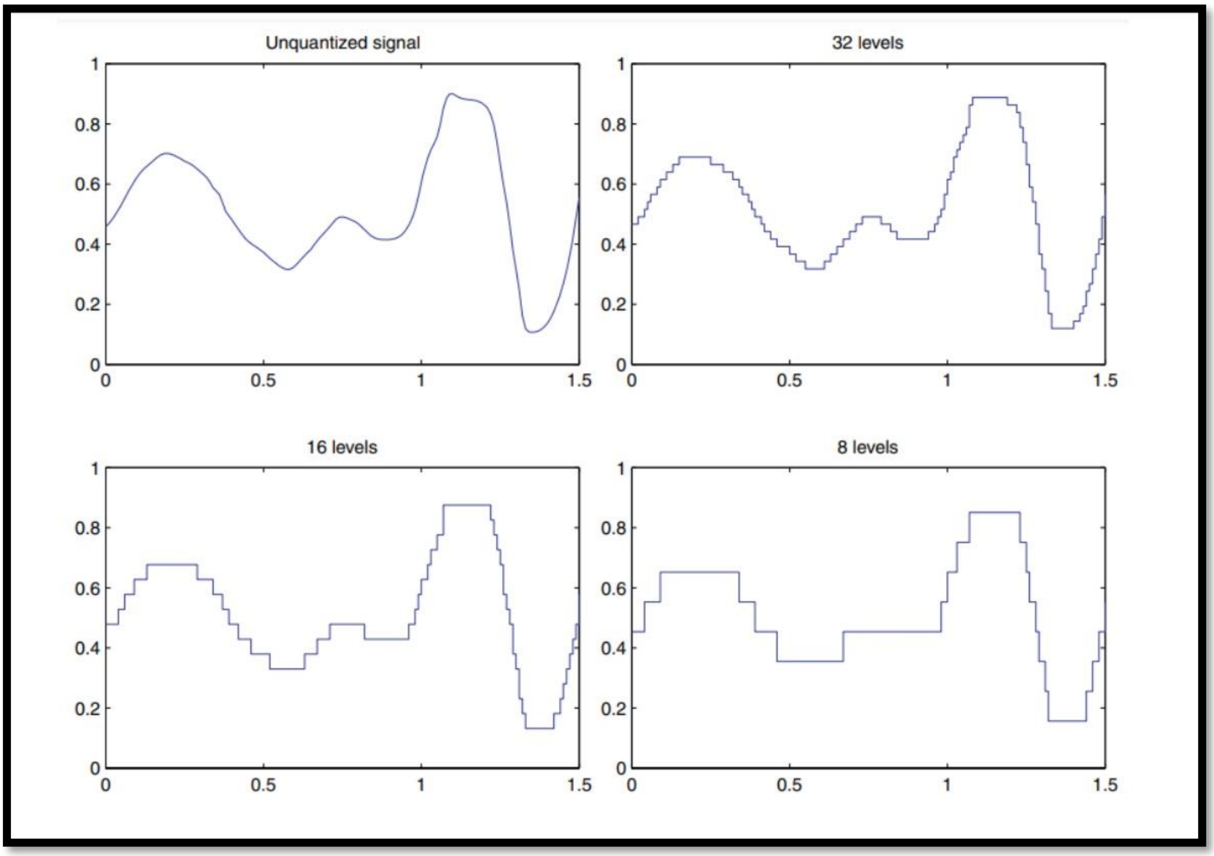


**Fig.: Quantized versions of an analog signal**

A simple method of quantization is given below:

$\Delta = \frac{x_{max} - x_{min}}{L - 1}$; $\Delta$= Step size

L = 2m; m= number of bits

i = round$\{\frac{x - x_{min}}{\Delta}\}$

xq = $x_{min}$ + i*Δ; i = 0, 1, …, L-1

where $\mathbf{x_{max}}$ and $\mathbf{x_{min}}$ are the maximum value and minimum values, respectively, of the analog input signal **x**. The symbol **L** denotes the number of quantization levels, where **m** is the number of bits used in ADC. The symbol **Δ** is the step size of the quantizer or the ADC resolution. Finally, **xq** indicates the quantization level, and **i** is an index corresponding to the binary code.

# Simulation:

### 1) Generating sinusoidal signals with different frequencies:

**Code:**

```
clc
clear all
close all
fs = 500; % Sampling frequency
t = 0:1/fs:0.5; % Time duration
f1 = 12; % Frequency of first signal
f2 = 6; % Frequency of second signal
A1 = 1.5; % Amplitude of first signal
A2 = 1.1; % Amplitude of second signal
x1 = A1*sin(2*pi*f1*t); % First Signal
x2 = A2*sin(2*pi*f2*t); % Second Signal
%Plotting both signals in time domain
plot(t,x1,'k--o','LineWidth',1)
hold on
plot(t,x2,'b-*','LineWidth',1)
hold off
xlabel('time in seconds')
ylabel('Amplitude in volts')
title('Signals of different Frequencies')
legend('Signal x1','Signal x2')
```

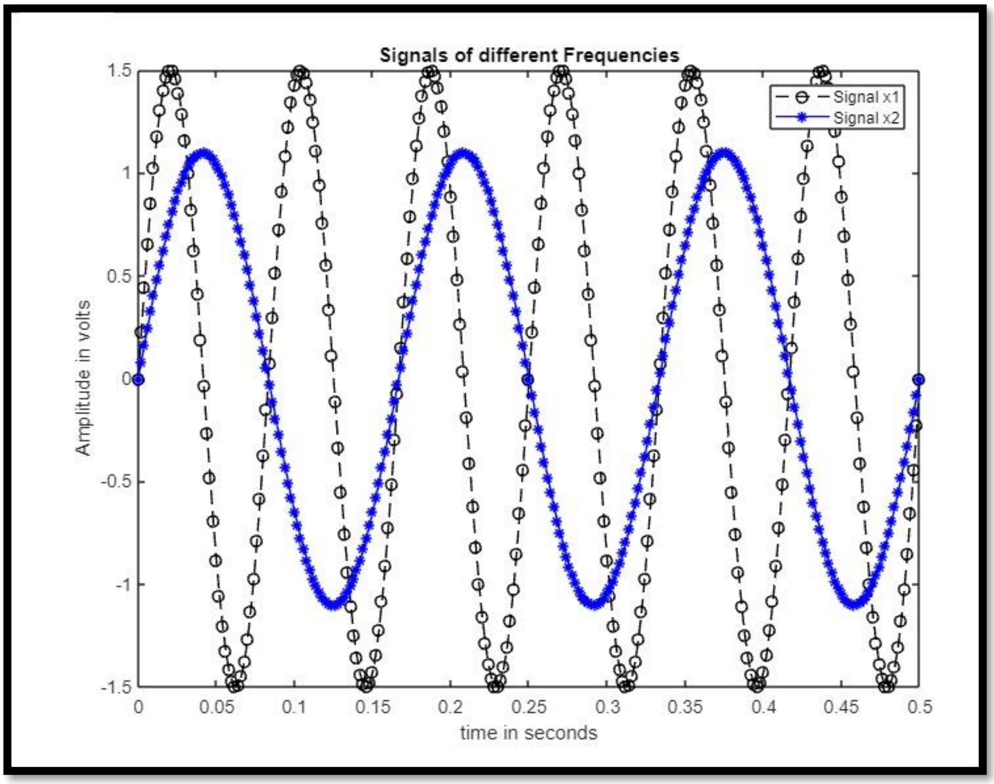


**Fig.: Signals with different frequencies**

**2) Representing signals in frequency domain:**

**Code:**

```
clc
clear all
close all
fs = 5000; % Sampling frequency
t = 0:1/fs:2; % Time duration
f1 = 12; % Frequency of first signal
f2 = 6; % Frequency of second signal
A1 = 2; % Amplitude of first signal
A2 = 3; % Amplitude of second signal
x1 = A1*sin(2*pi*f1*t); % First Signal
x2 = A2*sin(2*pi*f2*t); % Second Signal
nx = length(t); % Total number of samples
%Taking fourier transform
fx1 = fft(x1); % Frequency analysis is done here
fx2 = fft(x2); % Applying fftshift to put it in the form we are used to (see % documentation)
fx1 = fftshift(fx1)/(nx/2); % Axis correction and scaling are being done here
fx2 = fftshift(fx2)/(nx/2); % Next, we calculate the frequency axis, which is defined by the  sampling rate
f = linspace(-fs/2,fs/2,nx); % fft function in Matlab returns complex numbers that has both
% frequency and phase information
% we will only plot absolute values of the fft transformed variables
% to see the frequency domain representations
bar(f, abs(fx1),2,'k') % to create black bar graph across frequency axis
hold on
bar(f, abs(fx2),2,'r') % to create red bar graph across frequency axis
hold off
axis([-50 50 0 4])
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title('Frequency Domain Representation of Different Signals');
legend('Signal x1','Signal x2')
```

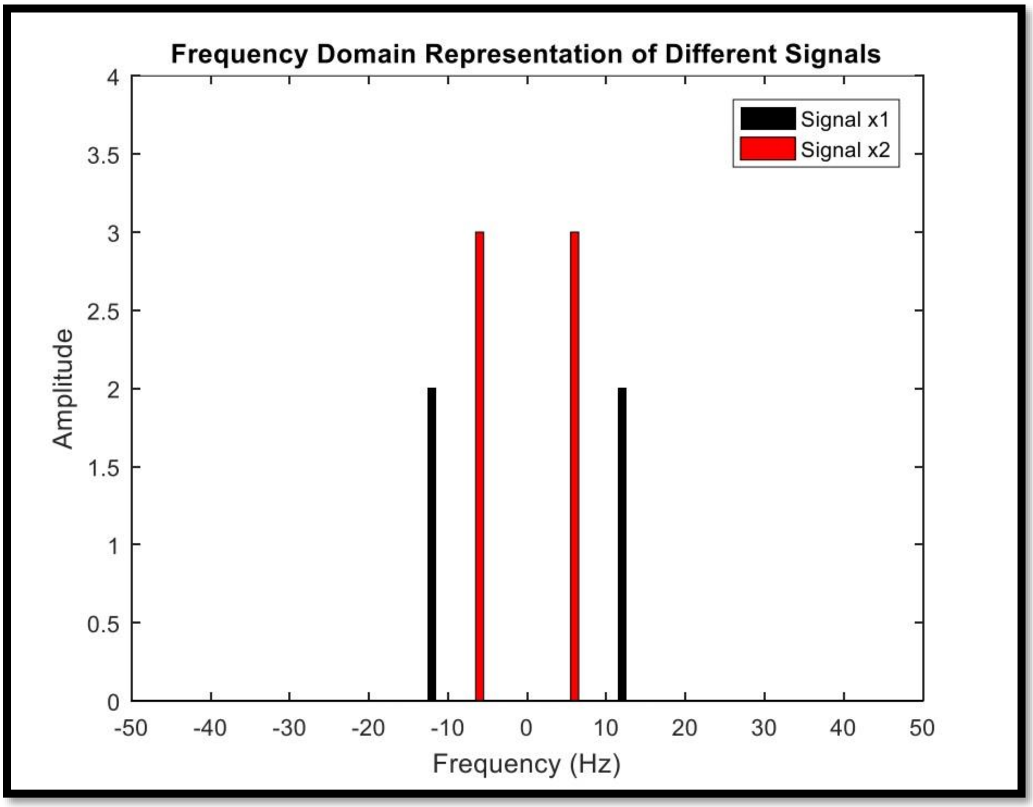


**Fig.: Frequency domain representation of different signals**

**3) Example of Bandwidth Calculation:**

**Code:**

```
fs = 8000; % Sampling frequency
t = 0:1/fs:1-1/fs; % Time duration
cx = 1.1*sin(2*pi*100*t) + 1.3*cos(2*pi*300*t) + 1.5*sin(2*pi*2000*t);
bandwidth = obw(cx,fs)  % obw is the occupied bandwidth function, it shows difference in frequency
```

**Command Window Output:**

bandwidth = 1.9010e+03

**4. Example showing signal processing (noise reduction) is convenient in frequency domain:**

**Code:**

```
close all;

clc;

%Define number of samples to take

fs = 8000;

f = 4; %Hz

%Define signal

t = 0:1/fs:2;

signal = 2*sin(2*pi*f*t);

nx = length(t); % Total number of samples

%Plot to illustrate that it is a sine wave

plot(t, signal,'linewidth',1);

title('Time-Domain Representation of Signal');

xlabel('Time (s)');

ylabel('Amplitude');

% Take fourier transform fftSignal =
fft(signal);

% Apply fftshift to put it in the form

% we are used to (see documentation) fftSignal =
fftshift(fftSignal)/(nx/2);

% Scaling done by dividing with (fs/2)

% Next, calculate the frequency axis,

% which is defined by the sampling rate f = linspace(-
fs/2,fs/2,nx);

% Since the signal is complex, we need to plot the magnitude to get it to

% look right, so we use abs (absolute value)

figure;

plot(f, abs(fftSignal),'linewidth',2);

title('Frequency-Domain Representation of Signal'); xlabel('Frequency (Hz)');
ylabel('Amplitude');

xlim([-20 20]) %noise

sd = 2;

noise = sd*randn(size(signal)); % noise power = sd^2 figure

plot(t,noise, 'linewidth', 1) xlabel('Time (s)');
ylabel('Amplitude');

title('Time-Domain Representation of Noise'); fftNoise = fft(noise);
```

```
fftNoise = fftshift(fftNoise)/(nx/2); figure

plot(f,abs(fftNoise), 'linewidth', 2)

title('Frequency-Domain Representation of Noise'); xlabel('Frequency (Hz)');
ylabel('Amplitude');

xlim([-20 20])

%noisy signal

noisySignal = signal + noise;

figure

plot(t,noisySignal, 'linewidth', 1) xlabel('Time (s)');
ylabel('Amplitude');

title('Time-Domain Representation of Noisy Signal'); fftNoisySignal =
fft(noisySignal); fftNoisySignal = fftshift(fftNoisySignal)/(nx/2); figure

plot(f,abs(fftNoisySignal), 'linewidth', 2)

title('Frequency-Domain Representation of Noisy Signal');

xlabel('Frequency (Hz)');

ylabel('Amplitude');

xlim([-20 20])
```

**Generated Graph plots for the above code:**

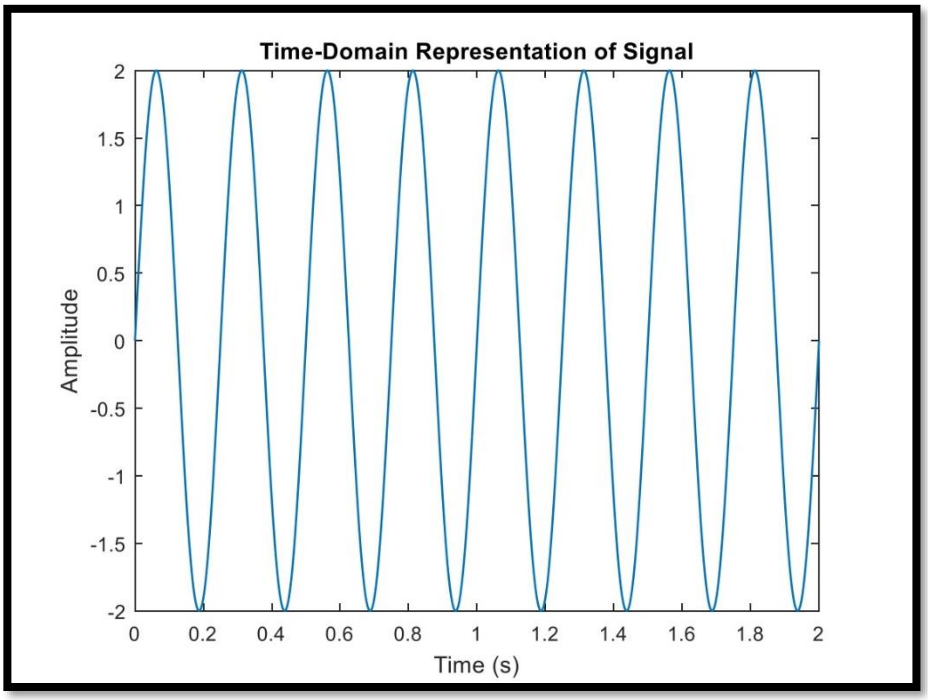


**Fig.: Time domain representation of a signal**

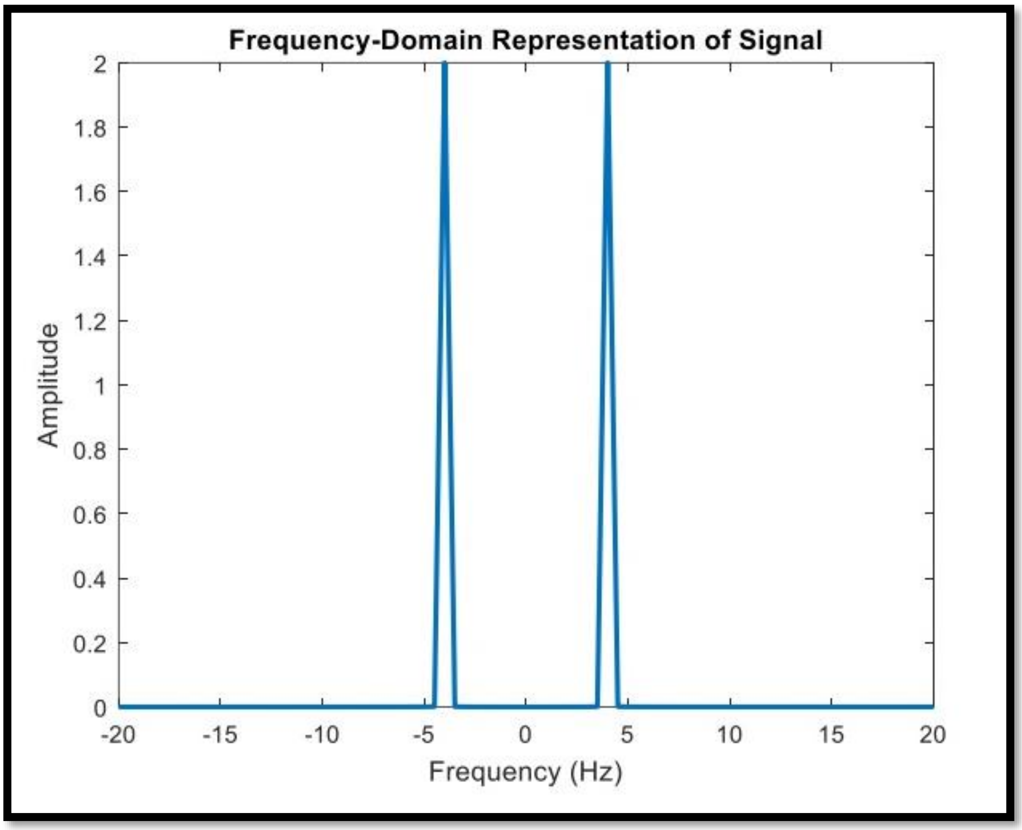


**Fig.: Frequency domain representation of a signal**

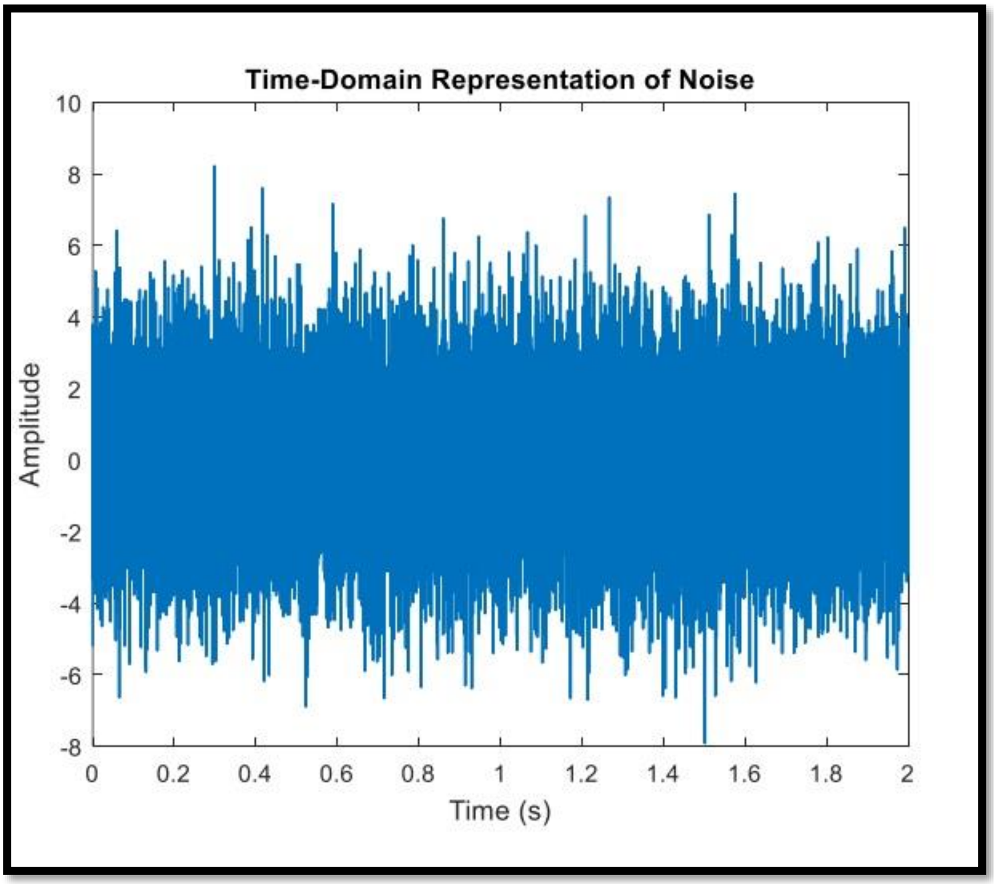


**Fig.: Time domain representation of noise**

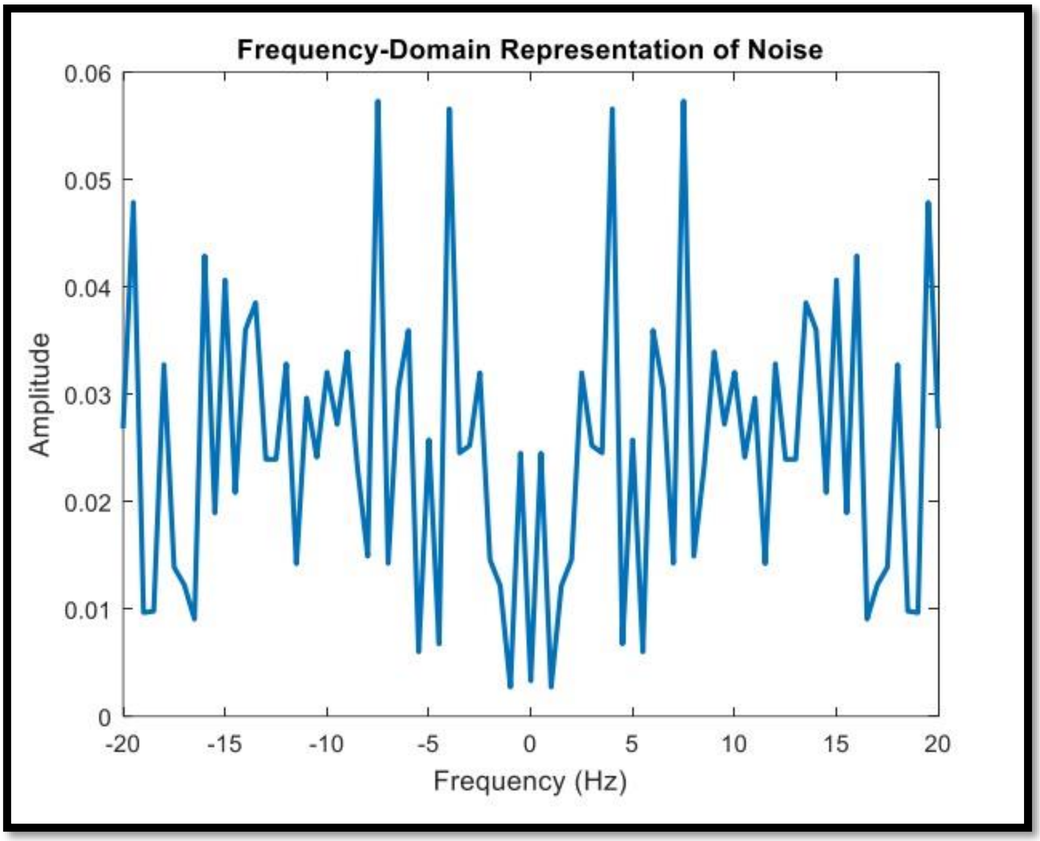


**Fig.: Frequency domain representation of noise**

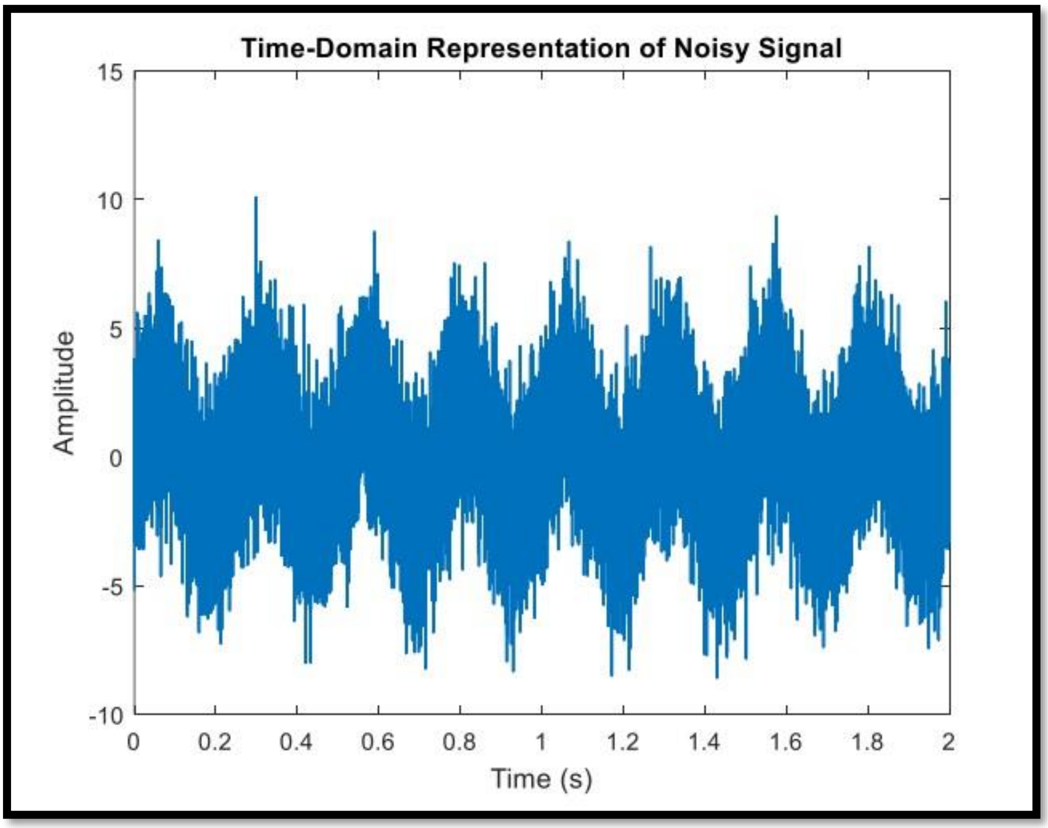


**Fig.: Time domain representation of noisy signal**

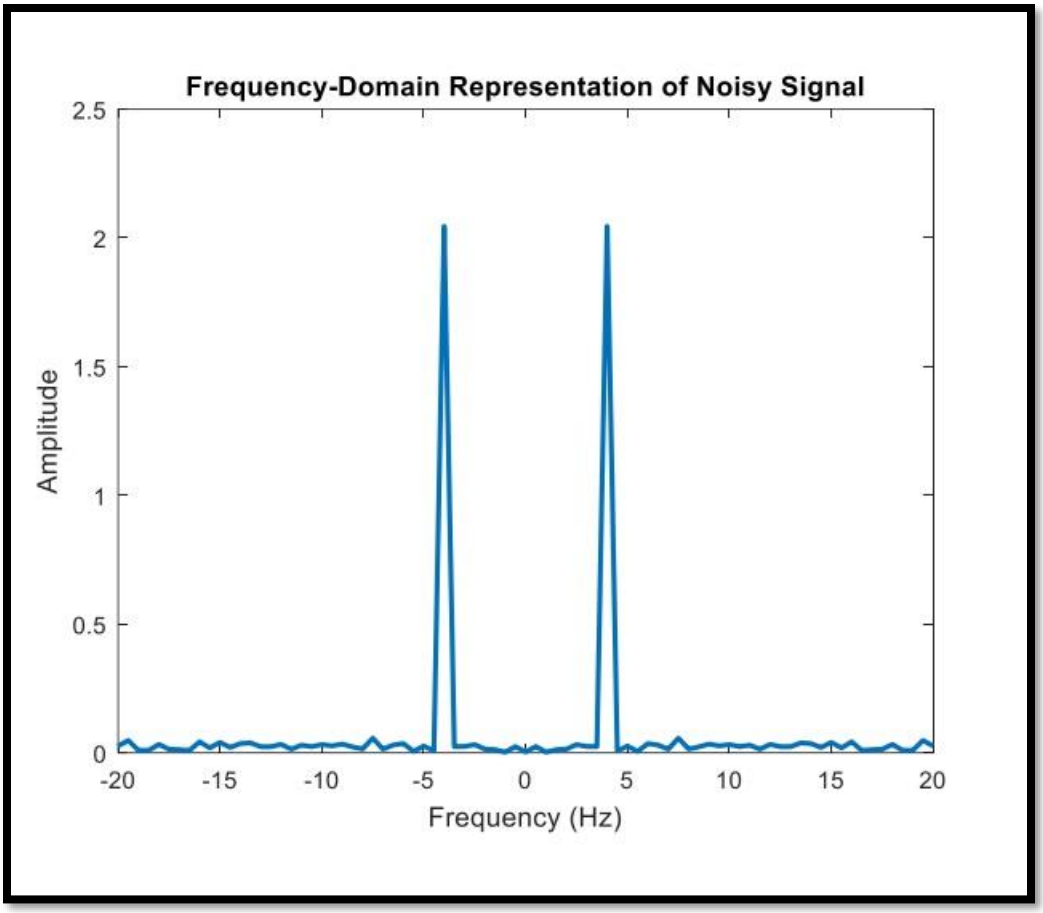


**Fig.: Frequency domain representation of noisy signal**

**5. Showing an example of Quantization using Matlab's bulit-in function 'quantiz'**

**Code:**

```
fs = 10000;
t = [0:1/fs:0.1];

f = 10; % Times at which to sample the sine function
sig = 2*sin(2*pi*f*t); % Original signal, a sine wave
partition = -1.5:1.5; % Length 4, to represent 5 intervals
codebook = -2:2; % Length 5, one entry for each interval
[index,quants] = quantiz(sig,partition,codebook); % Quantize.


figure
plot(t,sig,'x',t,quants,'.')
legend('Original signal','Quantized signal');
```

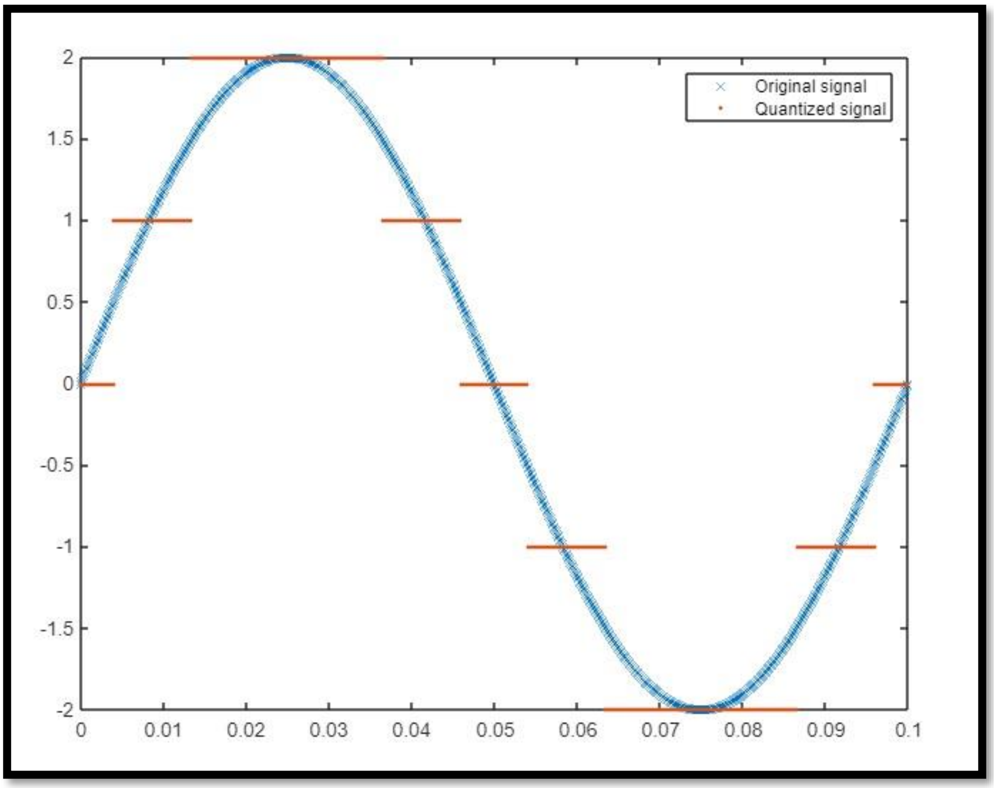


**Fig.: Quantization using built in MATLAB function**

**5. Showing an example of Quantization without using Matlab's bulit-in function 'quantiz'**

**Code:**

```
clc

close all

fs = 40e3;% sampling frequency

f = 50;% frequency of the signal

t = 0:1/fs:1/f;%discrete time

A=2;

x   = A*sin(2*pi*f*t);% discrete signal %--------
Quantization------------%

n = 3;

L =(2^n); delta=(max(x)-min(x))/(L-1);

xq = min(x)+(round((x-min(x))/delta)).*delta; %--------END------------
% plot(t,x,'r-.', 'linewidth',1.5);

hold on;

plot(t,xq,'k-.', 'linewidth',1.5); %plotting wave forms.
xlabel('time') ylabel('amplitude')

title('example of manual quantization') legend('Original
signal','quantized signal')
```

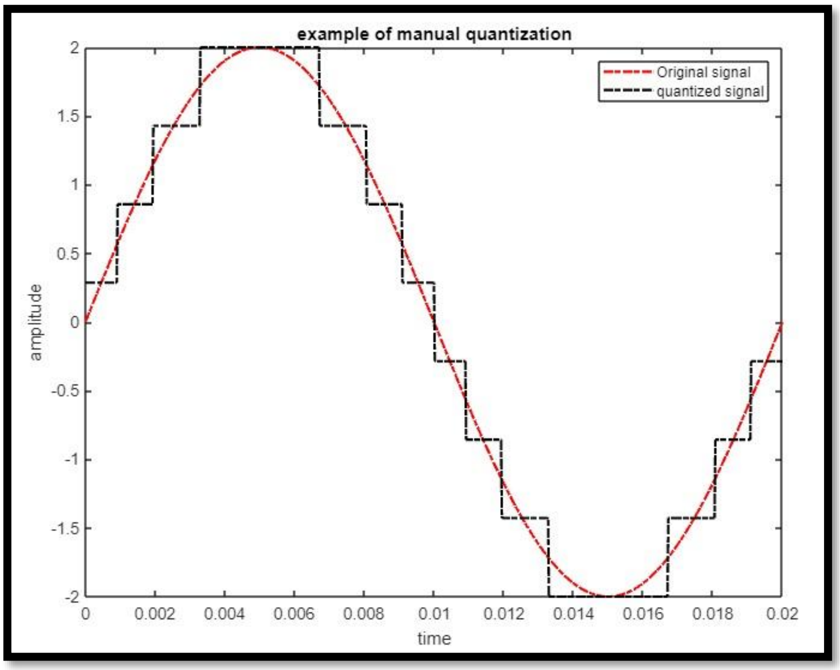


**Fig.: Quantization without using the built in MATLAB function**

## Software:

MATLAB2016a or newer versions

**Performance Task for Lab Report:**

**a) Creating a signal from student ID and showing the time and frequency domain representation of it.**

My ID=20-41840-1

So, a1=G+1=1; a2=F+2=6; a3= E+3 =11;

Again, f1=E+1=; f2=F+2=; f3= G+3 =;

**Code:**

```
fs = 1000; % sampling frequency
t = 0:1/fs:1; % will plot 1 second in the abscissa

 a1=1; %calculated according to student id
 f1=9; %calculated according to student id
 x1=a1*cos(2*pi*f1*t);
 a2=6; %calculated according to student id
 f2=6; %calculated according to student id
 x2= a2*cos(2*pi*f2*t);
 a3=11; %calculated according to student id
 f3=3;  %calculated according to student id
 x3=a3*cos(2*pi*f3*t);

 signal_x=x1+x2+x3; %composite of the 3 signals

%------Plotting the first graph
subplot(2,1,1);

 plot(t,signal_x); %plotting using the plot function
 xlabel('time'); %labelling the axes
 ylabel('amplitude');  %labelling the axes
 title('Time domain representation of a signal'); %title

%------Plotting the second graph

nx = length(t); % Total number of samples

fsx = fft(signal_x); %Taking fourier transform

%      Applying fftshift to put it in the form we are used to

ffsx = fftshift(fsx )/(nx/2); % Axis correction and scaling are % done here

%      Next, we calculate the frequency axis, which is defined by the
%      sampling rate
f = linspace(-fs/2,fs/2,nx);

subplot(2,1,2)
bar(f, abs(ffsx),2,'k')     %fft function in Matlab returns complex numbers that has both
                            %frequency and phase information

                            %we have only plotted absolute values of the fft
                            %transformed variables

axis([-40 40 0 20])
xlabel('Frequency (Hz)'); %labelling the axes

ylabel('Amplitude');

title('Frequency Domain Representation');   %title
```

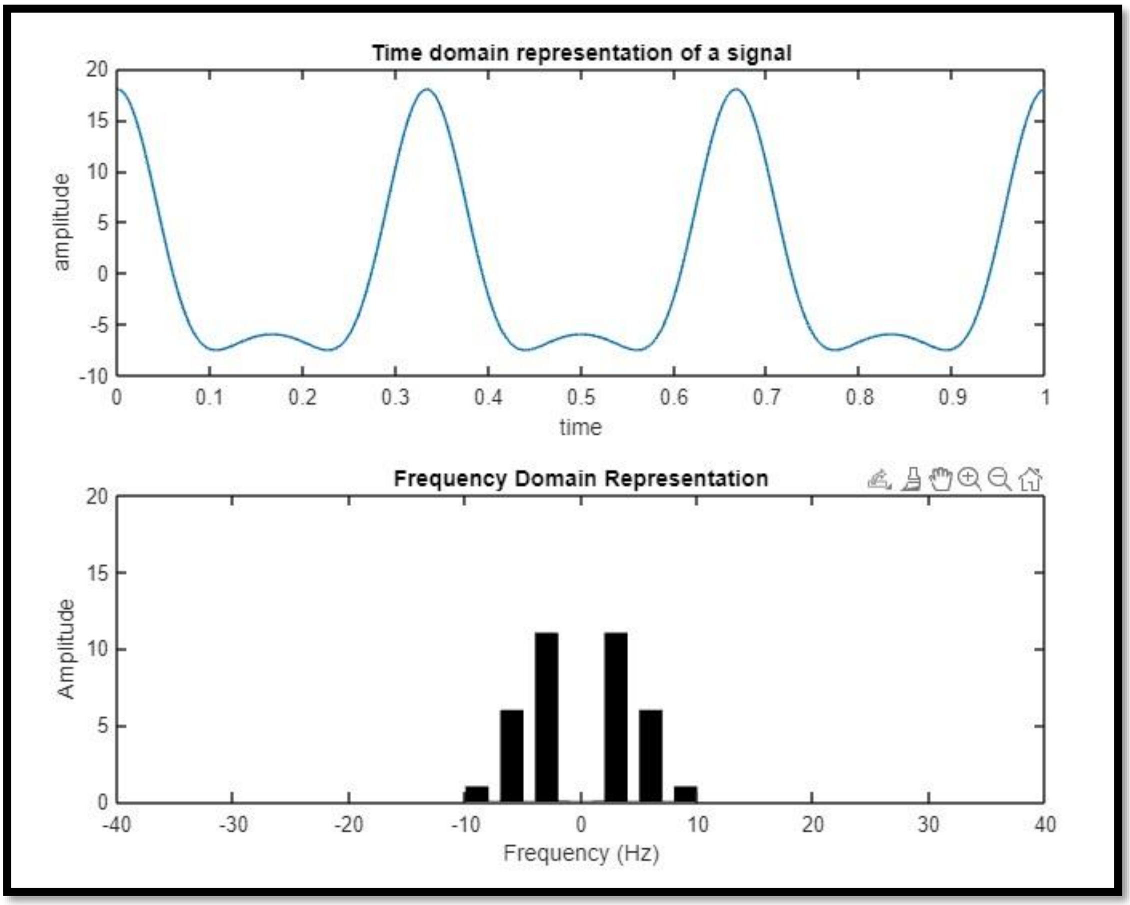


**Fig.: Time and frequency domain representation**

**(P.T.O.)**

**b) Quantizing the signal into 4 levels using the MATLAB built-in function**

**code:**

```
fs = 1000; % sampling frequency

t = [0:1/fs:1]; % will plot 1 second in the abscissa

 a1=1; %calculated according to student id
 f1=9; %calculated according to student id
 x1=a1*cos(2*pi*f1*t);
 a2=6; %calculated according to student id
 f2=6; %calculated according to student id
 x2= a2*cos(2*pi*f2*t);
 a3=11; %calculated according to student id
 f3=3;  %calculated according to student id
 x3=a3*cos(2*pi*f3*t);

 signal_x=x1+x2+x3; %composite of the 3 signals

 partition  =-7:12:17; % Length 3 to represent 4 intervals
 codebook = -8:8.66:18; % Length 4, one entry for each interval
                    %the partition matrix defines the range and the
                    %codebook matrix assigns corresponding values to
                    %those range intervals

 [index,quants] = quantiz(signal_x,partition,codebook);
 % Quantization using the quantization() function

plot(t,signal_x,'-',t,quants,'-'); %plotting the original and quantized signal

 xlabel('time'); %labels for the axes
 ylabel('amplitude');

title("Quantization using quantiz() function") %title on the top
legend('Original signal','Quantized signal'); %legend to specify the signals
axis([0 0.33 -10 20]); %set to show one full cycle of this wave
```

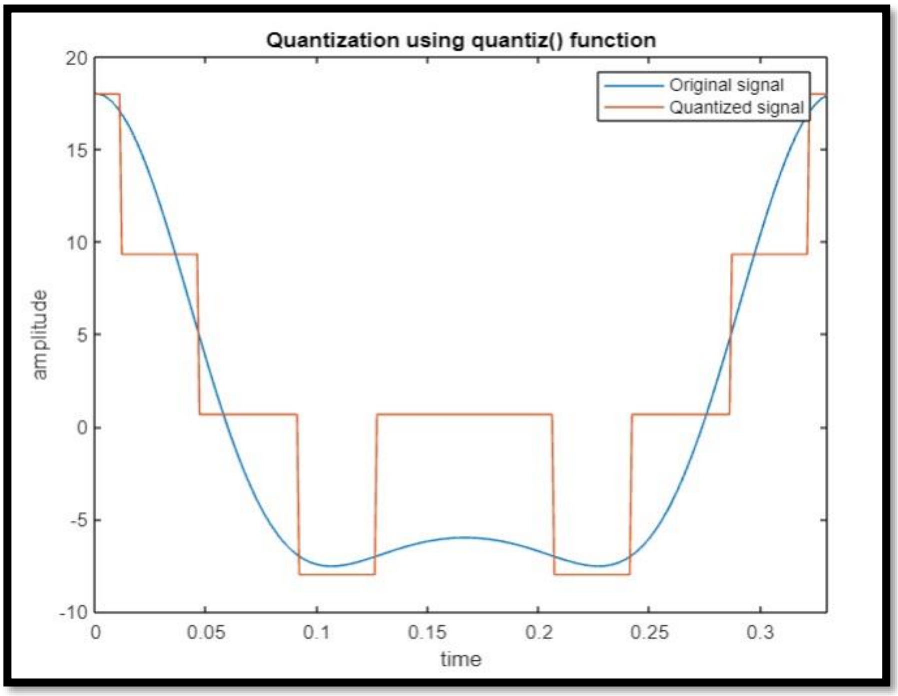


**Fig.: Quantizing the signal into 4 levels using the quantiz() function**

### c) Quantizing the signal into 8 levels manually

**code:**

```
fs = 1000;% sampling frequency

t = 0:1/fs:1;%will plot for 1 second in the time domain

a1=1; %calculated according to student id
f1=9; %calculated according to student id
x1=a1*cos(2*pi*f1*t);
a2=6; %calculated according to student id
f2=6; %calculated according to student id
x2= a2*cos(2*pi*f2*t);
a3=11; %calculated according to student id
f3=3; %calculated according to student id
x3=a3*cos(2*pi*f3*t);
signal_x=x1+x2+x3; %composite of the 3 signals


n = 3;
L =(2^n); delta=(max(signal_x)-min(signal_x))/(L-1);
signal_xq = min(signal_x)+(round((signal_x-min(signal_x))/delta)).*delta;

%the above equations will quantize the input signal into 8 deiscrete levels

plot(t, signal_x,'r-.', 'linewidth',1.5); %plotting the original signal
hold on; %retaining the plot
plot(t,signal_xq,'k-.', 'linewidth',1.5); %plotting the quantized signal on the same plot
hold off; %setting the plot free

xlabel('time') %labelling the axes
ylabel('amplitude')

title('example of manual quantization') %title
legend('Original signal','quantized signal')%specifying the signals
axis([0 0.33 -10 20]); %showing one full cycle
```

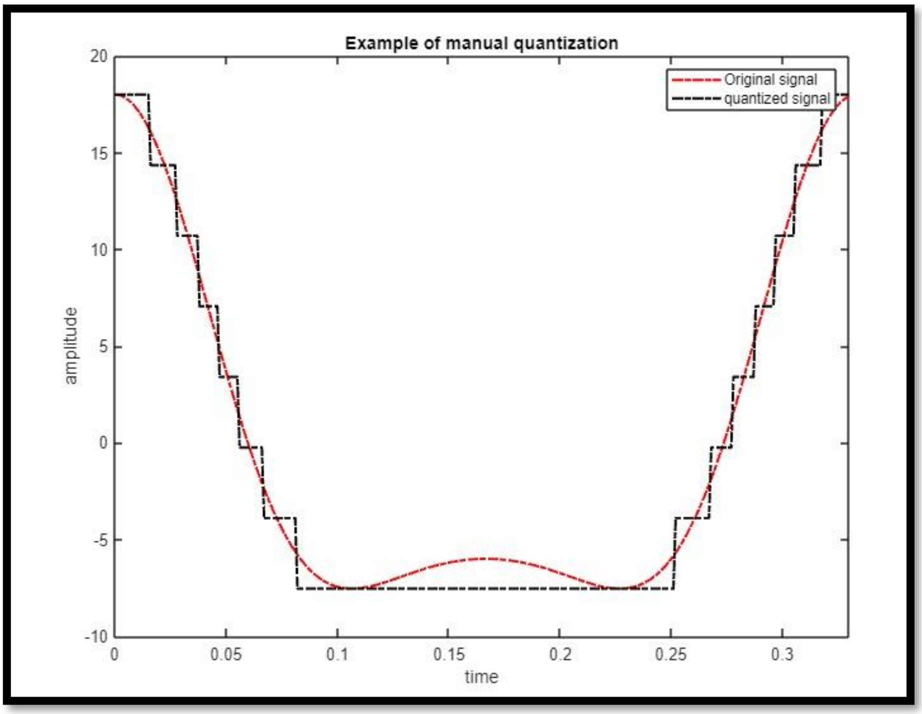


**Fig.: Quantizing the signal into 8 levels manually**

**Results and Discussion:** For the lab performance task we have successfully represented a composite signal in time and frequency domains, performed quantization on the signal by both the built-in function and manual codes.

One issue we faced was with the double quotations, because MATLAB identifies these symbols (“ ”) as errors. This issue was resolved by replacing the double quotations with single quotations(‘ ’),

Overall meticulous care was taken to avoid errors and the outputs were obtained perfectly during experiment time. We therefore believe the experiment has been completed successfully.

**Conclusion:** Through this experiment we have successfully demonstrated the study of signal frequency, spectrum, bandwidth, and quantization using MATLAB.

**References:**

1. MATLAB user guide.
2. Prof. Dr.-Ing. Andreas Czylwik, “MATLAB for Communications”
3. AIUB Student Lab Manual