



SIDDHARTHA
INSTITUTE OF TECHNOLOGY & SCIENCES
Korremula Road, Narapally, Ghatkesar Mandal, Ranga Reddy District.



LABORATORY MANUAL

For

BASIC SIMULATION

(II B.Tech. ECE- I Semester- R18 .AY:2021 - 22)

Prepared by

- 1. Mr M RANJITH REDDY, Asst. Professor**
- 2. Mr K RAJESH, Asst. Professor**

Department of
Electronics & Communication Engineering
2022

www.siddhartha.co.in

SIDDHARTHA INSTITUTE OF TECHNOLOGY & SCIENCES

CONTENTS

S.No	Description	Page No.
1	Vision & Mission of the Institute	i
2	Vision & Mission of the Department	ii
3	Program Outcomes	iii
4	Rules and Regulations of Lab	iv
5	Basic Operations on Matrices.	1
6	Generation of Various Signals and Sequences (Periodic and Non periodic), such as Unit Impulse, Unit Step, Square, Saw tooth, Triangular, Sinusoidal, Ramp, Sinc.	5
7	Operations on Signals and Sequences such as Addition, multiplication, Scaling, Shifting, Folding, Computation of Energy and Average Power	12
8	Finding the Even and Odd parts of Signal/Sequence and Real and Imaginary parts of Signal	20
9	Convolution between Signals and sequences	25
10	Auto Correlation and Cross Correlation between Signals and Sequences	29
11	Verification of Linearity and Time Invariance Properties of a given Continuous/Discrete System	33
12	Computation of Unit sample, Unit step and Sinusoidal responses of the given LTI system and verifying its physical reliability and stability properties.	37
13	Gibbs Phenomenon	40
14	Finding the Fourier Transform of a given signal and plotting its magnitude and phase spectrum	42
15	Waveform Synthesis using Laplace Transform.	45
16	Locating the Zeros and Poles and plotting the Pole-Zero maps in S-plane and Z Plane for the given transfer function.	49
17	Generation of Gaussian noise (Real and Complex), Computation of its mean, M.S. Value and its Skew, Kurtosis, and PSD, Probability Distribution Function	53
18	Sampling Theorem Verification	55
19	Removal of noise by Autocorrelation / Cross correlation	58

20	Extraction of Periodic Signal masked by noise using Correlation	62
21	Verification of Weiner-Khinchine Relations.	66
ADDITIONAL EXPERIMENTS		
22	Generate a DTS of N=1000 i.i.d uniformly distributed random no.'s in the interval (-0.5,-0.5) and compute the autocorrelation of the seq.	67
23	Obtain and plot the power spectrum of the o/p process when a white random process is passed through a filter with specific impulse resp.	68

SIDDHARTHA INSTITUTE OF TECHNOLOGY & SCIENCES

VISION & MISSION OF THE INSTITUTE

VISION:

To be a Centre of Excellence in Technical Education and to become an epic center of Research for creative solutions.

MISSION:

To address the Emerging Needs through Quality Technical Education with an emphasis on practical skills and Advanced Research with social relevance.

OBJECTIVES:

- To translate our vision into action and accomplish our mission, we strive to provide state-of-art infrastructure.
- Recruit, Motivate and develop faculty of high caliber and with multiple specialization.
- Continuously review, innovate and experiment teaching methodologies and learning processes.
- Focus on research, training and consultancy through an Integrated Institute-Industry symbiosis.

SIDDHARTHA INSTITUTE OF TECHNOLOGY & SCIENCES

VISION & MISSION OF THE DEPARTMENT

VISION :

To provide innovative teaching and learning methodologies for excelling in a high-value career, higher education and research to the students in the field of Electronics and Communication Engineering to meet the needs of the industry and to be a part of the advancing technological revolution.

MISSION :

- To create engineers of high quality on par with international standards by providing excellent infrastructure and well qualified faculty.
- To establish centers of excellence to enhance collaborative and multidisciplinary activities to develop human and intellectual qualities.
- To provide technical expertise to carry out research and development.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS) :

Graduates shall apply the fundamental, advanced and contemporary knowledge of

1. Electronics, Communication and allied Engineering, to develop efficient solutions and systems, to meet the needs of the industries and society.
2. Graduates will get employed or pursue higher studies or research.
3. Graduates will have team spirit, good communication skills and ethics with lifelong learning attitude.

SIDDHARTHA INSTITUTE OF TECHNOLOGY & SCIENCES













PROGRAM OUTCOMES:

Engineering Graduates will be able to:





1. **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

RULES AND REGULATIONS OF LAB

All students must observe the Dress Code while in the laboratory.

-  All bags must be placed at rack.
-  The lab timetable must be strictly followed.
-  Be PUNCTUAL for your laboratory session.
-  Program/experiment must be executed within the given time.
-  Workspace must be kept clean and tidy at all time.
-  Handle the systems and interfacing kits with care.
-  All students are liable for any damage to the accessories due to their own negligence.
-  All interfacing kits connecting cables must be RETURNED if you taken from the lab supervisor.
-  Students are strictly PROHIBITED from taking out any items from the laboratory.
-  Students are NOT allowed to work alone in the laboratory without the Lab Supervisor
-  USB Ports have been disabled if you want to use USB drive consult lab supervisor.
-  Report immediately to the Lab Supervisor if any malfunction of the accessories, is there

Before leaving the lab

-  Place the chairs properly.
-  Turn off the system properly
-  Turn off the monitor.
-  Please check the laboratory notice board regularly for updates.

Experiment No-1

BASIC OPERATIONS ON MATRICES

AIM: Generate a matrix and perform basic operation on matrices using MATLAB software.

Software Required: MATLAB software

Theory:

MATLAB treats all variables as matrices. Vectors are special forms of matrices and contain only one row or one column. Whereas scalars are special forms of matrices and contain only one row and one column. A matrix with one row is called row vector and a matrix with single column is called column vector.

The first one consists of convenient matrix building functions, some of which are given below.

1. eye - identity matrix
2. zeros - matrix of zeros
3. ones - matrix of ones
4. diag - extract diagonal of a matrix or create diagonal matrices
5. triu - upper triangular part of a matrix
6. tril - lower triangular part of a matrix
7. rand - ran

commands in the second sub-category of matrix functions are

1. size- size of a matrix
2. det -determinant of a square matrix
3. inv- inverse of a matrix
4. rank- rank of a matrix
5. rref- reduced row echelon form
6. eig- eigenvalues and eigenvectors
7. poly- characteristic polynomial

Program:

% Creating a column

vector >> a=[1;2;3]

a =

1 2

3

% Creating a row vector

>> b=[1 2 3]

b=

1 2 3

% Creating a matrix

```
>> m=[1 2 3;4 6 9;2 6
```

```
9] m =
```

```
1 2 3
```

```
4 6 9
```

```
2 6 9
```

% Extracting sub matrix from matrix

```
>> sub_m=m(2:3,2:3)
```

```
sub_m =
```

```
6 9
```

```
6 9
```

% extracting column vector from

```
matrix >> c=m(:,2)
```

```
c =
```

```
2
```

```
6
```

```
6
```

% extracting row vector from

```
matrix >> d=m(3,:)
```

```
d =
```

```
2 6 9
```

% creation of two matrices a and b

```
>> a=[2 4 -1;-2 1 9;-1 -1 0]
```

```
a =
```

```
2 4 -1
```

```
-2 1 9
```

```
-1 -1 0
```

```
>> b=[0 2 3;1 0 2;1 4 6]
```

```
b =
```

```
0 2 3
```

```
1 0 2
```

```
1 4 6
```

% matrix multiplication

```
>> x1=a*b
```

x1 =

3 0 8

10 32 50

-1 -2 -5

% element to element

multiplication >> x2=a.*b

x2 =

0 8 -3 -

2 0 18

-1 -4 0

% matrix addition

```
>> x3=a+b
```

x3 =

2 6 2

-1 1 11

0 3 6

% matrix subtraction

```
>> x4=a-b
```

x4 =

2 2 -4

-3 1 7

-2 -5 -6

% matrix division

```
>> x5=a/b
```

x5 =

-9.0000 -3.5000 5.5000

12.0000 3.7500 -5.7500

3.0000 0.7500 -1.7500

% element to element division

```
>> x6=a./b
```

Warning: Divide by zero.

```
x6 =  
Inf 2.0000 -0.3333  
-2.0000 Inf 4.5000  
-1.0000 -0.2500 0  
  
% inverse of matrix a  
>> x7=inv(a)  
x7 =  
-0.4286 -0.0476 -1.7619  
0.4286 0.0476 0.7619 -  
0.1429 0.0952 -0.4762  
  
% transpose of matrix a  
>> x8=a'  
x8 =  
2 -2 -1  
4 1 -1  
-1 9 0
```

RESULT: Matrix operations are performed using Matlab software.

VIVA QUESTIONS:-

1. Expand MATLAB? And importance of MATLAB?
2. What is clear all and close all will do?
3. What is disp() and input()?
4. What is the syntax to find the eigen values and eigenvectors of the matrix?
5. What is the syntax to find rank of the matrix?

Experiment No-2

Generation of signals and sequences

AIM: Generate various signals and sequences (Periodic and aperiodic), such as Unit Impulse, Unit Step, Square, Saw tooth, Triangular, Sinusoidal, Ramp, Sinc.

Software Required: Matlab software

Theory: If the amplitude of the signal is defined at every instant of time then it is called continuous time signal. If the amplitude of the signal is defined at only at some instants of time then it is called discrete time signal. If the signal repeats itself at regular intervals then it is called periodic signal. Otherwise they are called aperiodic signals.

EX: ramp, Impulse, unit step, sinc- Aperiodic signals

square, sawtooth, triangular sinusoidal – periodic signals.

Ramp sinal: The **ramp function** is a unitary real function, easily computable as the mean of the independent variable and its absolute value. This function is applied in engineering. The name *ramp function* is derived from the appearance of its graph.

$$r(t) = \begin{cases} t & \text{when } t \geq 0 \\ 0 & \text{else} \end{cases}$$

Unit impulse signal: One of the more useful functions in the study of linear systems is the "unit impulse function." An ideal impulse function is a function that is zero everywhere but at the origin, where it is infinitely high. However, the *area* of the impulse is finite

$$Y(t) = \begin{cases} 1 & \text{when } t=0 \\ 0 & \text{other wise} \end{cases}$$

Unit step signal: The unit step function and the impulse function are considered to be fundamental functions in engineering, and it is strongly recommended that the reader becomes very familiar with both of these functions.

$$u(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t > 0 \\ \frac{1}{2} & \text{if } t=0 \end{cases}$$

Sinc signal: There is a particular form that appears so frequently in communications engineering, that we give it its own name. This function is called the "Sinc function".

The Sinc function is defined in the following manner:

$$\text{sinc}(x) = \frac{\sin \pi x}{\pi x} \quad \text{if } x \neq 0 \text{ and } \text{sinc}(0) = 1$$

The value of sinc(x) is defined as 1 at x = 0, since

$$\lim_{x \rightarrow 0} \text{sinc}(x) = 1$$

PROCEDURE:-

- Open MATLAB
- Open new M-file
- Type the program
- Save in current directory
- Compile and Run the program
- For the output see command window\ Figure window

PROGRAM:

```
% Generation of signals and
sequences clc;
clear all; close all;
%~~~~~
%generation of unit impulse signal
t1=-1:0.01:1

y1=(t1==0);
subplot(2,2,1);
plot(t1,y1); xlabel('time');
ylabel('amplitude');
title('unit impulse signal');
%generation of impulse
sequence subplot(2,2,2);
stem(t1,y1);

xlabel('n');
ylabel('amplitude');
title('unit impulse sequence');
%~~~~~
~

%generation of unit step signal
t2=-10:1:10;
y2=(t2>=0);
subplot(2,2,3);
plot(t2,y2);
xlabel('time');
ylabel('amplitude');
title('unit step signal');
%generation of unit step sequence
subplot(2,2,4);
stem(t2,y2);
xlabel('n');
ylabel('amplitude');
title('unit step sequence');
%~~~~~

%generation of square wave signal
t=0:0.002:0.1;
y3=square(2*pi*50*t);
```

```

figure;
subplot(2,2,1);
plot(t,y3);
axis([0 0.1 -2 2]);
xlabel('time');
ylabel('amplitude');
title('square wave signal');
%generation of square wave sequence
subplot(2,2,2);
stem(t,y3);
axis([0 0.1 -2 2]);
xlabel('n');
ylabel('amplitude');
title('square wave sequence');
%~~~~~

%generation of sawtooth signal
y4=sawtooth(2*pi*50*t);
subplot(2,2,3);
plot(t,y4);
axis([0 0.1 -2 2]);
xlabel('time');
ylabel('amplitude');
title('sawtooth wave signal');
%generation of sawtooth sequence
subplot(2,2,4);
stem(t,y4);
axis([0 0.1 -2 2]);
xlabel('n');
ylabel('amplitude');
title('sawtooth wave sequence');
%~~~~~

%generation of triangular wave
signal y5=sawtooth(2*pi*50*t,.5);
figure;
subplot(2,2,1);
plot(t,y5);
axis([0 0.1 -2 2]);
xlabel('time');
ylabel('amplitude');
title(' triangular wave signal');
%generation of triangular wave sequence
subplot(2,2,2);
stem(t,y5);
axis([0 0.1 -2 2]);
xlabel('n');
ylabel('amplitude');
title('triangular wave sequence');
%~~~~~

```

```

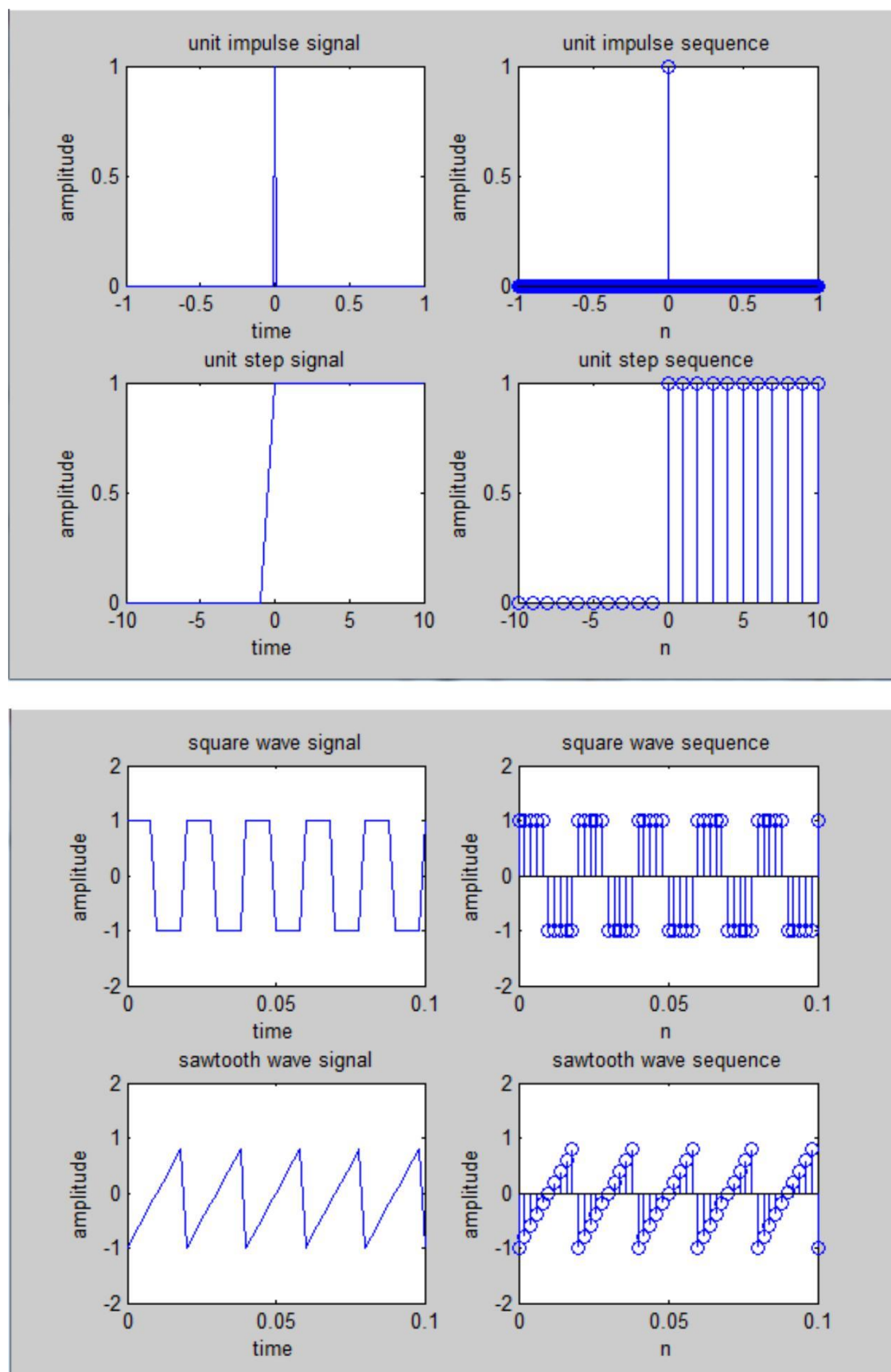
%generation of sinsoidal wave signal
y6=sin(2*pi*40*t);
subplot(2,2,3);
plot(t,y6);
axis([0 0.1 -2 2]);
xlabel('time');
ylabel('amplitude');
title(' sinsoidal wave signal');
%generation of sin wave sequence
subplot(2,2,4);
stem(t,y6);
axis([0 0.1 -2 2]);
xlabel('n');
ylabel('amplitude');
title('sin wave sequence');
%~~~~~

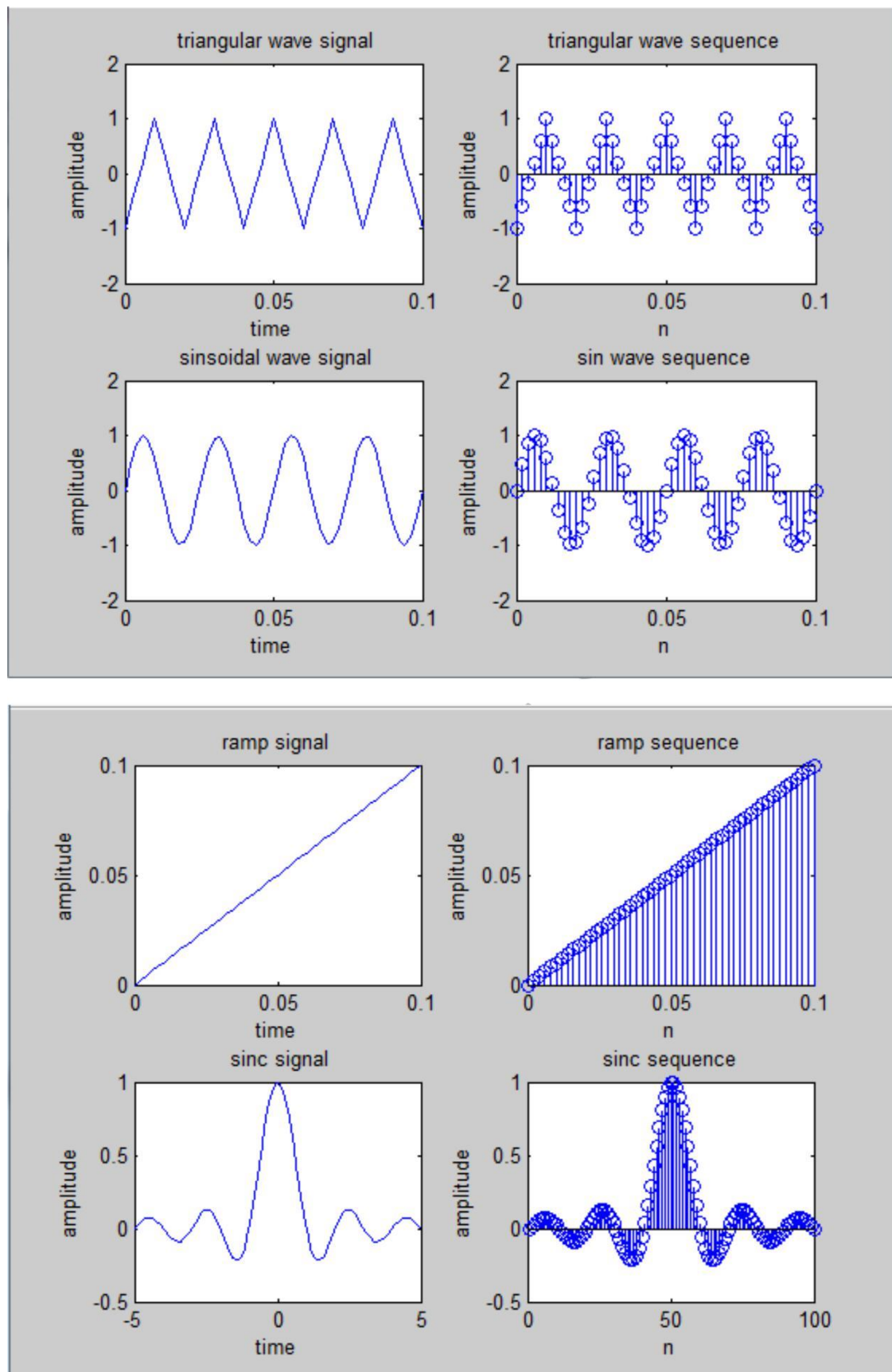
%generation of ramp signal
y7=t;
figure;
subplot(2,2,1);
plot(t,y7);
xlabel('time');
ylabel('amplitude');
title('ramp signal');
%generation of ramp sequence
subplot(2,2,2);
stem(t,y7);
xlabel('n');
ylabel('amplitude');
title('ramp sequence');
%~~~~~

%generation of sinc signal
t3=linspace(-5,5);
y8=sinc(t3);
subplot(2,2,3);
plot(t3,y8);
xlabel('time');
ylabel('amplitude');
title(' sinc signal');
%generation of sinc sequence
subplot(2,2,4);
stem(y8);
xlabel('n');
ylabel('amplitude');
title('sinc sequence');

```

Result: Various signals & sequences generated using Matlab software.

output:



VIVA QUESTIONS:-

1. Define Signal?
2. Define continuous and discrete Signals?
3. State the relation between step, ramp and Delta Functions?
4. Differentiate saw tooth and triangular signals?
5. Define Periodic and aperiodic Signal?

Experiment No-3

Basic Operations on Signals and sequences

AIM: perform the operations on signals and sequences such as addition, multiplication, scaling, shifting, folding and also compute energy and power.

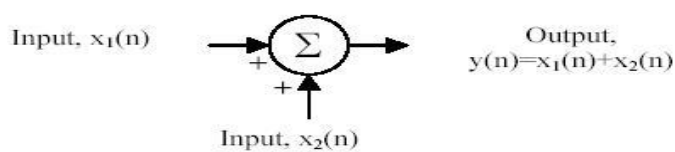
Software Required: Matlab software.

Theory:

Signal Addition

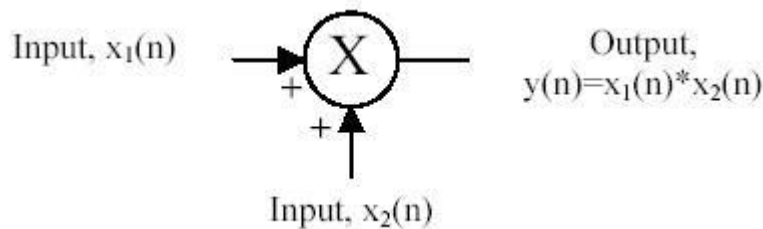
Addition: any two signals can be added to form a third signal,

$$z(t) = x(t) + y(t)$$



Multiplication :

Multiplication of two signals can be obtained by multiplying their values at every instants . $z(t) = x(t) y(t)$



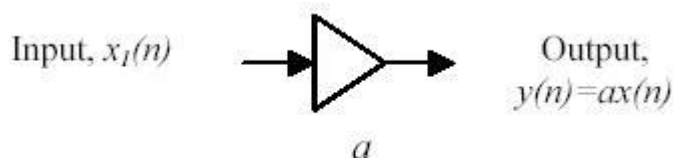
Time reversal/Folding:

Time reversal of a signal $x(t)$ can be obtained by folding the signal about $t=0$.

$$Y(t) = x(-t)$$

Signal Amplification/Scaling : $Y(n) = ax(n)$ if $a < 1$ attnuation

$a > 1$ amplification



Time shifting: The time shifting of $x(n)$ obtained by delay or advance the signal in time by using $y(n) = x(n+k)$

If k is a positive number, $y(n)$ shifted to the right i.e the shifting delays the signal

If k is a negative number, $y(n)$ it gets shifted left. Signal Shifting advances the signal

Energy:

$$E[n] = \lim_{N \rightarrow \infty} \sum_{n=-N}^N |x[n]|^2 = \sum_{n=-\infty}^{\infty} |x[n]|^2$$

Average power:

$$P[n] = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N |x[n]|^2$$

Program:

```

clc;
clear all;
close all;
%~~~~~

% generating two input
signals t=0:.01:1;
x1=sin(2*pi*4*t);
x2=sin(2*pi*8*t);
subplot(2,2,1);
plot(t,x1);
xlabel('time');
ylabel('amplitude');
title('input signal 1');
subplot(2,2,2);
plot(t,x2);
xlabel('time');
ylabel('amplitude');
title('input signal 2');

% addition of
signals y1=x1+x2;
subplot(2,2,3);
plot(t,y1);
xlabel('time');
ylabel('amplitude');
title('addition of two signals');

% multiplication of
signals y2=x1.*x2;
subplot(2,2,4);
plot(t,y2);
xlabel('time');
ylabel('amplitude');
title('multiplication of two signals');

```

```

% scaling of a
signal1 A=2;
y3=A*x1;
figure;
subplot(2,2,1);
plot(t,x1); xlabel('time');
ylabel('amplitude');
title('input signal')
subplot(2,2,2); plot(t,y3);
xlabel('time');
ylabel('amplitude');
title('amplified input signal');

```

```

% folding of a signal1
h=length(x1); nx=0:h-1;
subplot(2,2,3);
plot(nx,x1);
xlabel('nx');
ylabel('amplitude');
title('input signal')
y4=fliplr(x1); nf=-
fliplr(nx);
subplot(2,2,4);
plot(nf,y4);
xlabel('nf');
ylabel('amplitude');
title('folded signal');

```

```

%shifting of a signal 1
figure;
subplot(3,1,1);
plot(t,x1);
xlabel('time t');
ylabel('amplitude');
title('input signal');
subplot(3,1,2);
plot(t+2,x1);
xlabel('t+2');
ylabel('amplitude');
title('right shifted signal');
subplot(3,1,3);
plot(t-2,x1);
xlabel('t-2');
ylabel('amplitude');
title('left shifted signal');
%~~~~~

```

```

%operations on sequences
n1=1:1:9;
s1=[1 2 3 0 5 8 0 2 4];
figure;
subplot(2,2,1);
stem(n1,s1);
xlabel('n1');
ylabel('amplitude');
title('input sequence1');
s2=[1 1 2 4 6 0 5 3 6];
subplot(2,2,2);
stem(n1,s2);
xlabel('n2');
ylabel('amplitude');
title('input sequence2');

% addition of sequences
s3=s1+s2; subplot(2,2,3);
stem(n1,s3); xlabel('n1');
ylabel('amplitude');
title('sum of two
sequences');

% multiplication of sequences
s4=s1.*s2;
subplot(2,2,4);
stem(n1,s4);
xlabel('n1');
ylabel('amplitude')
;
title('product of two sequences');
%~~~~~

% program for energy of a sequence
z1=input('enter the input sequence');
e1=sum(abs(z1).^2);
disp('energy of given sequence is');e1

% program for energy of a
signal t=0:pi:10*pi;
z2=cos(2*pi*50*t).^2;
e2=sum(abs(z2).^2);
disp('energy of given signal is');e2

% program for power of a sequence p1=
(sum(abs(z1).^2))/length(z1);
disp('power of given sequence is');p1

% program for power of a signal
p2=(sum(abs(z2).^2))/length(z2);
disp('power of given signal is');p2

```

OUTPUT:

enter the input sequence[1 3 2 4 1]

energy of given sequence is

$$e1 = 31$$

energy of given signal is

$$e2 = 4.0388$$

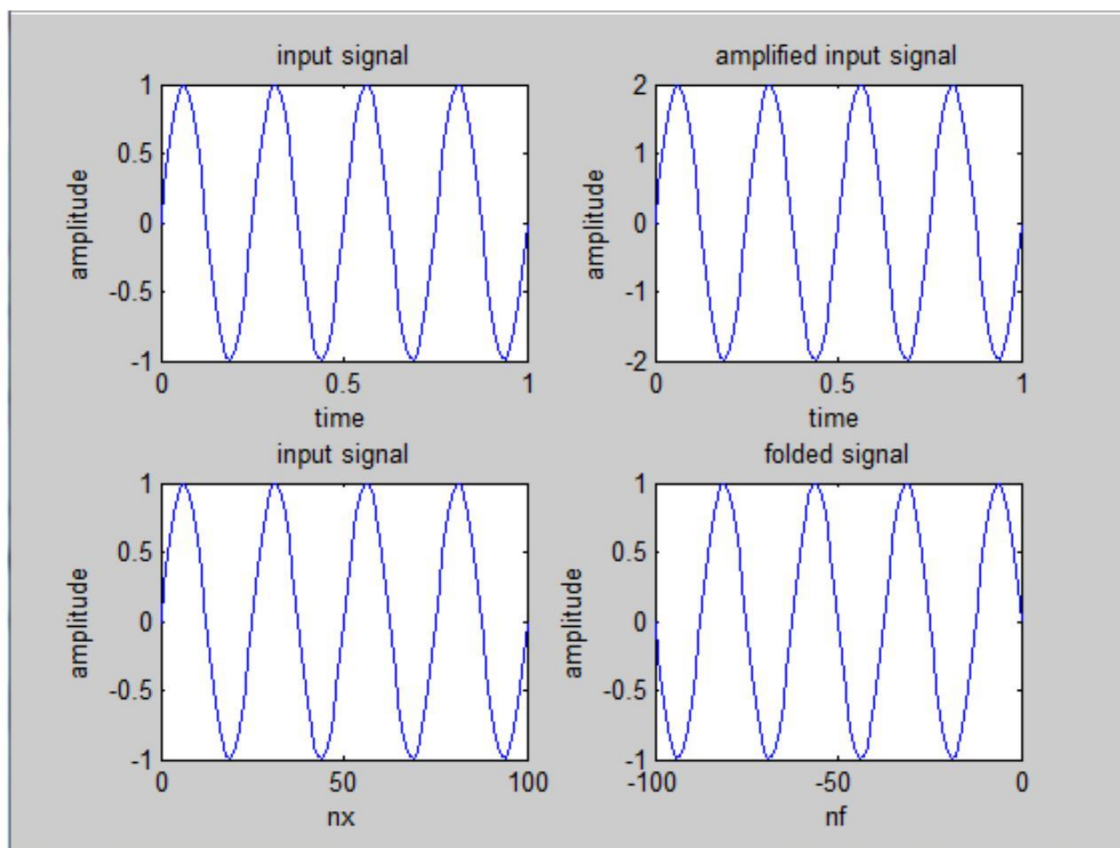
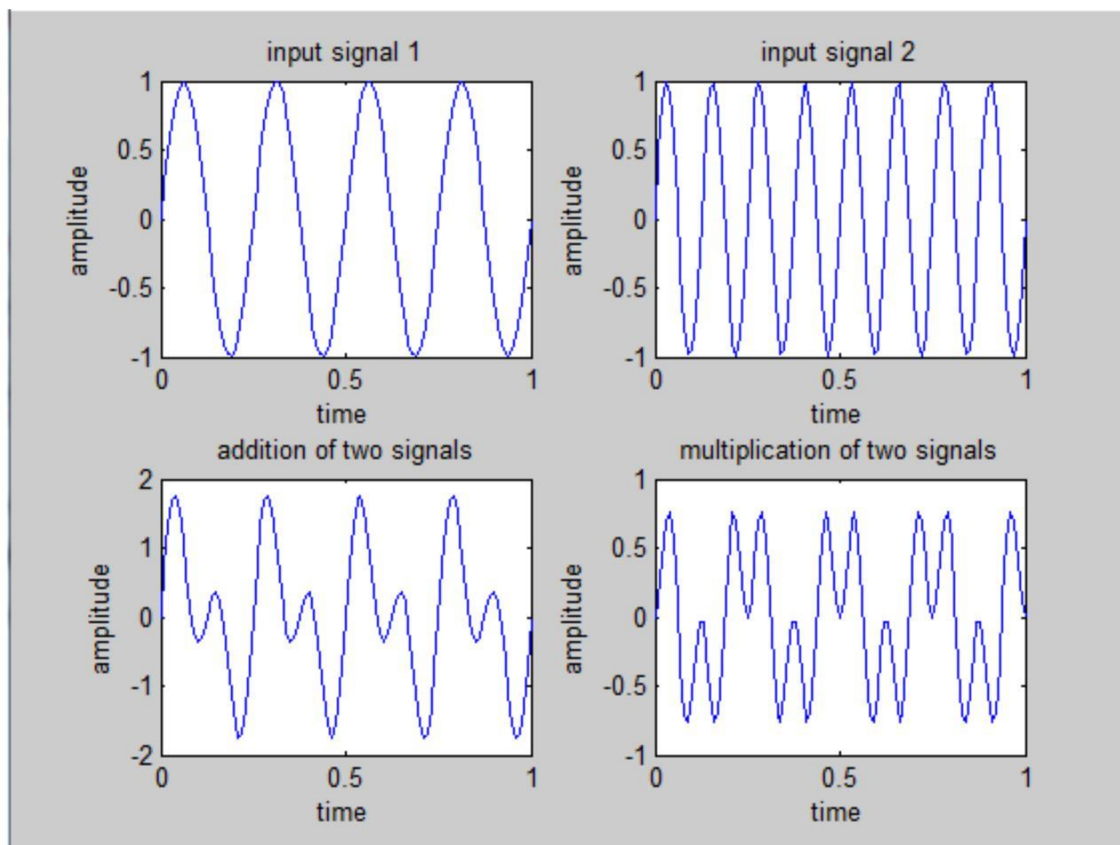
power of given sequence is

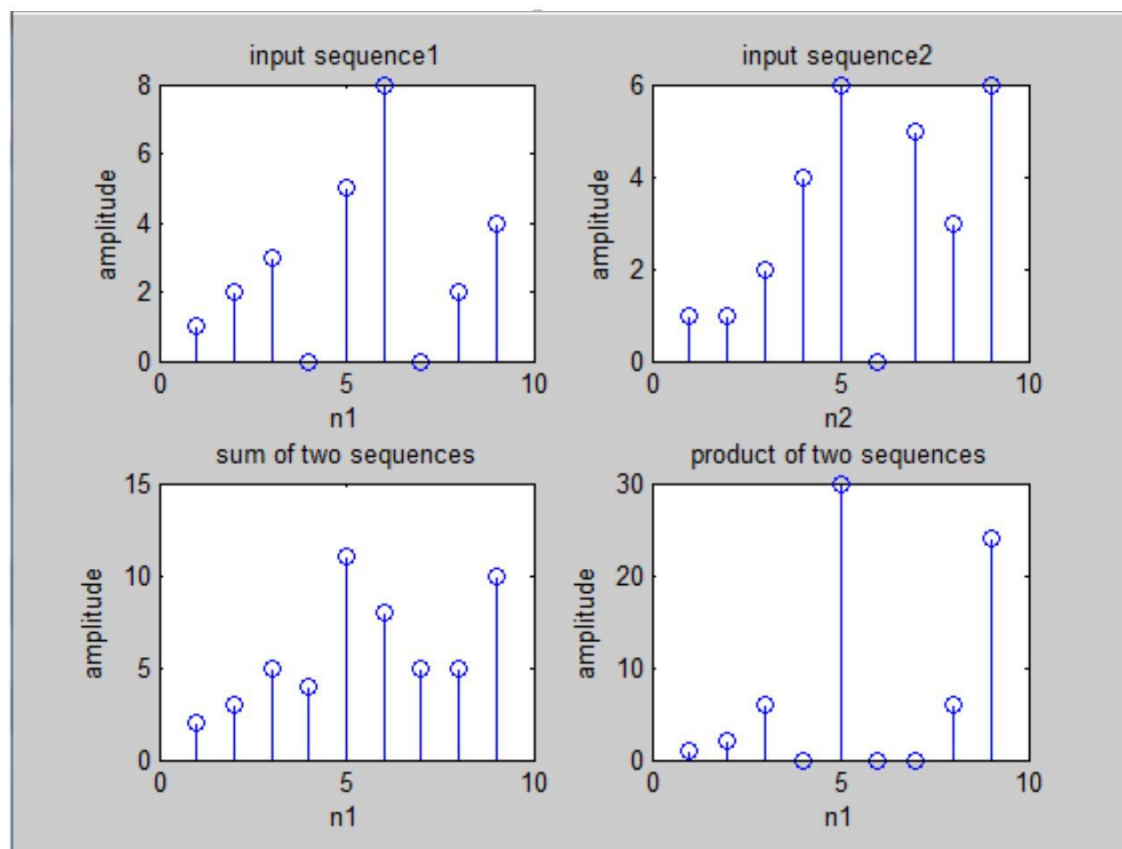
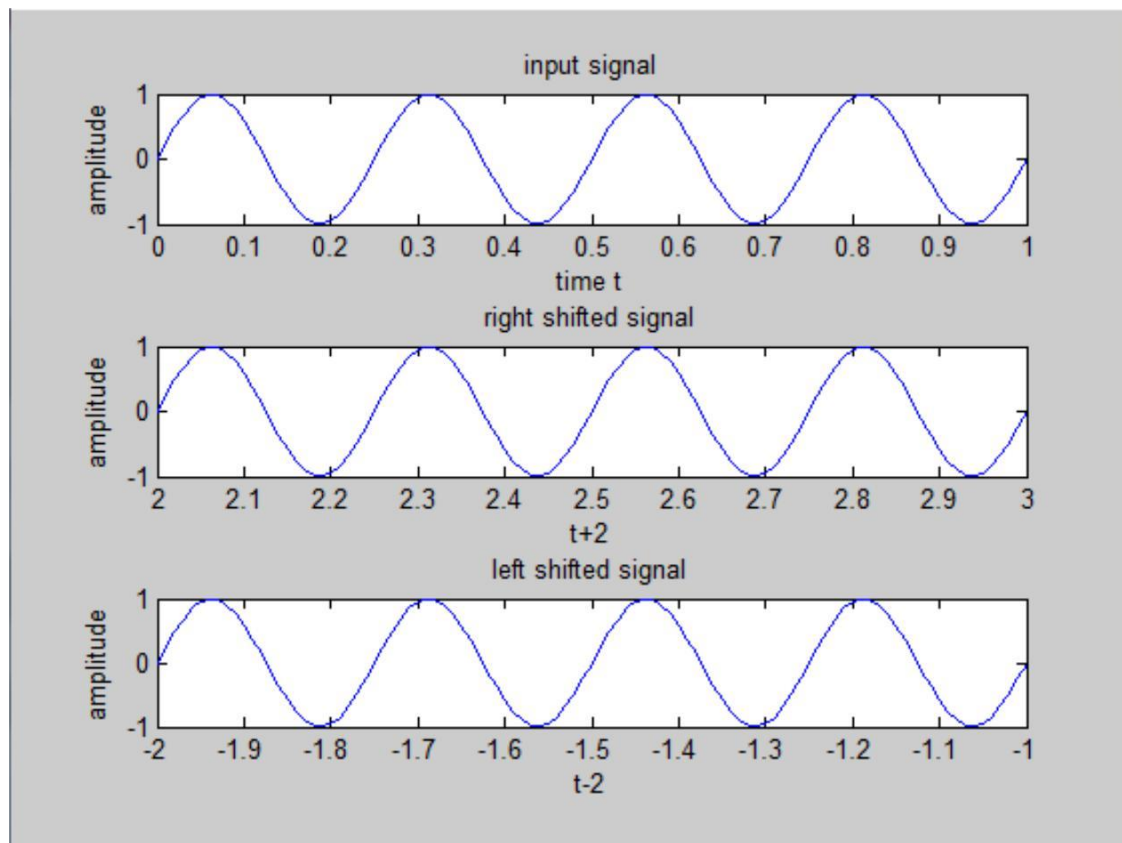
$$p1 = 6.2000$$

power of given signal is

$$p2 = 0.3672$$

Result: Various operations on signals and sequences are performed.

Output:



VIVA QUESTIONS:-

1. Define Symmetric and Anti-Symmetric Signals?
2. Define Continuous and Discrete Time Signals?
3. What are the Different types of representation of discrete time signals?
4. What are the Different types of Operation performed on signals?
5. What is System?

Experiment No-4

Even and odd parts of signal and sequence & Real and imaginary parts of Signal

AIM: Finding even and odd part of the signal and sequence and also find real and imaginary parts of signal.

Software Required: Matlab software

Theory: One of characteristics of signal is symmetry that may be useful for signal analysis. Even signals are symmetric around vertical axis, and Odd signals are symmetric about origin.

Even Signal: A signal is referred to as an even if it is identical to its time-reversed counterparts; $x(t) = x(-t)$.

Odd Signal: A signal is odd if $x(t) = -x(-t)$.

An odd signal must be 0 at $t=0$, in other words, odd signal passes the origin.

Using the definition of even and odd signal, any signal may be decomposed into a sum of its even part, $x_e(t)$, and its odd part, $x_o(t)$, as follows

Even and odd part of a signal: Any signal $x(t)$ can be expressed as sum of even and odd components i.e.,

$$x(t) = x_e(t) + x_o(t)$$

$$x_e(t) = \frac{1}{2} \{x(t) + x(-t)\}$$

$$x_o(t) = \frac{1}{2} \{x(t) - x(-t)\}$$

$$x(t) = x_e(t) + x_o(t)$$

$$= \frac{1}{2} \{x(t) + x(-t)\} + \frac{1}{2} \{x(t) - x(-t)\}$$

Program:

```
clc
close all;
clear all;

%Even and odd parts of a signal
t=0:.001:4*pi;
x=sin(t)+cos(t);      % x(t)=sin(t)+cos(t)
subplot(2,2,1)
plot(t,x)
xlabel('t');
ylabel('amplitude')
title('input signal')

y=sin(-t)+cos(-t);    % y(t)=x(-t)
subplot(2,2,2)
plot(t,y)
xlabel('t');
ylabel('amplitude')
title('input signal with t= -t')
even=(x+y)/2;
subplot(2,2,3)
plot(t,even)
xlabel('t');
```

```

ylabel('amplitude')
title('even part of the signal')
odd=(x-y)/2;
subplot(2,2,4)
plot(t,odd)
xlabel('t');
ylabel('amplitude');
title('odd part of the signal');

% Even and odd parts of a
sequence x1=[0,2, -3,5,-2,-1,6];
n=-3:3;
y1= fliplr(x1);%y1(n)=x1(-n)
figure;
subplot(2,2,1);
stem(n,x1); xlabel('n');
ylabel('amplitude');
title('input sequence');
subplot(2,2,2);
stem(n,y1); xlabel('n');
ylabel('amplitude');

title('input sequence with n= -
n'); even1=.5*(x1+y1);
odd1=.5*(x1-y1);

% plotting even and odd parts of the
sequence subplot(2,2,3);
stem(n,even1); xlabel('n');
ylabel('amplitude');
title('even part of sequence');
subplot(2,2,4); stem(n,odd1);

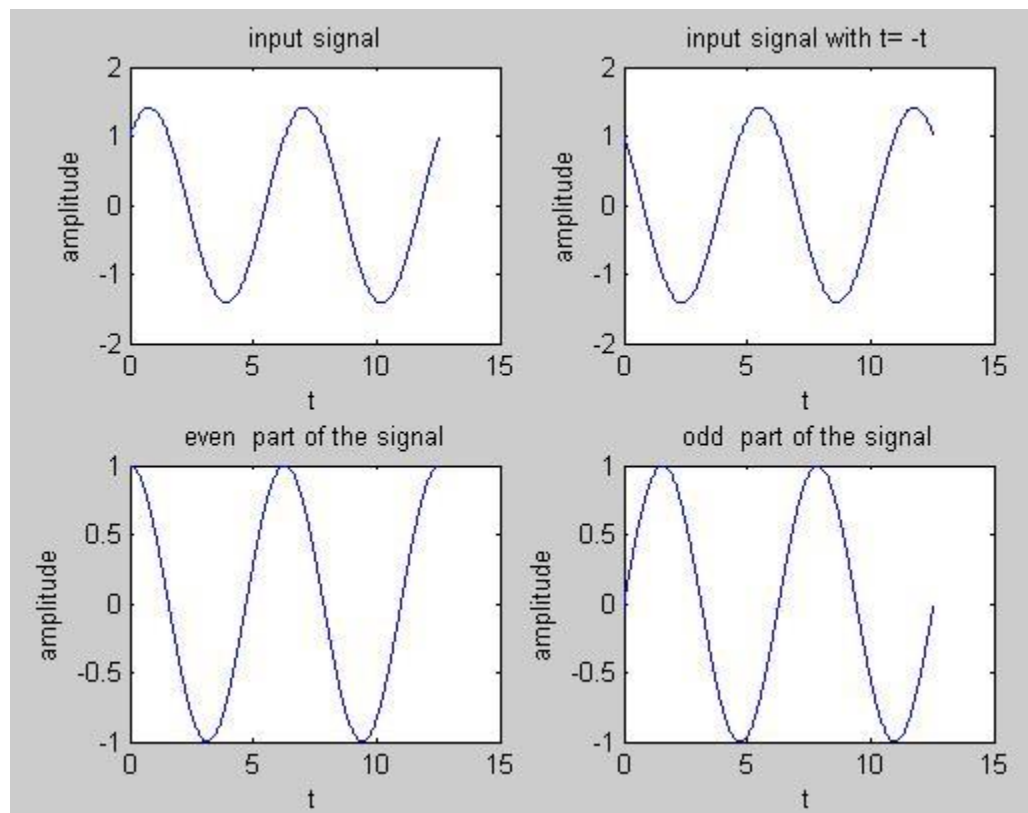
xlabel('n');
ylabel('amplitude'); title('odd
part of sequence');

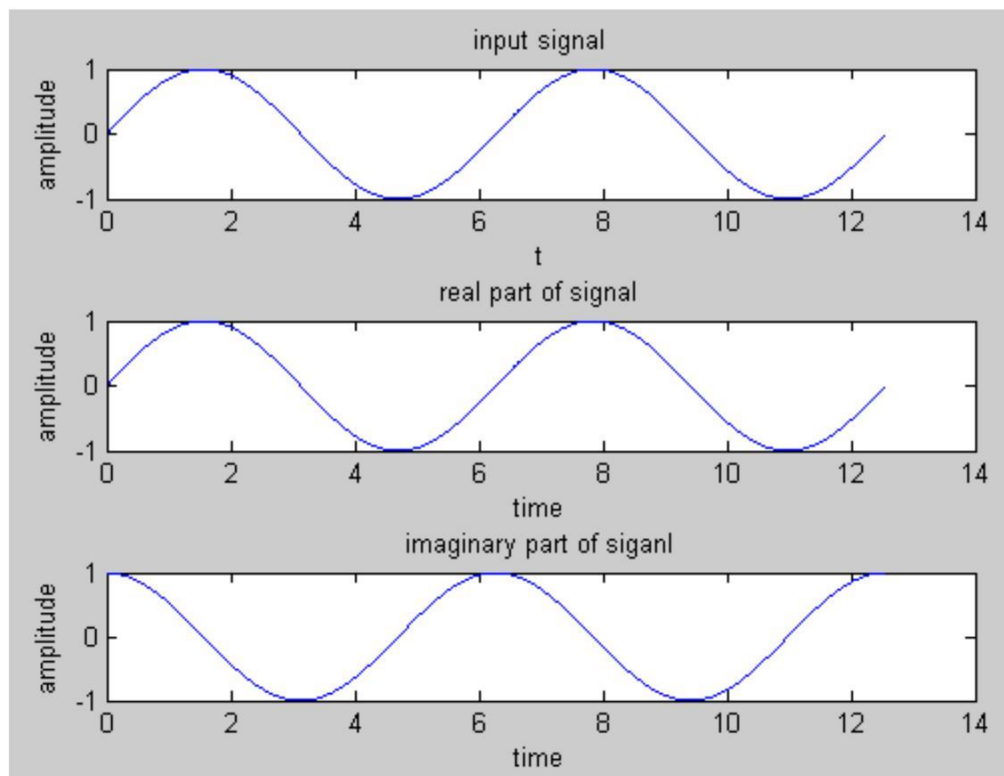
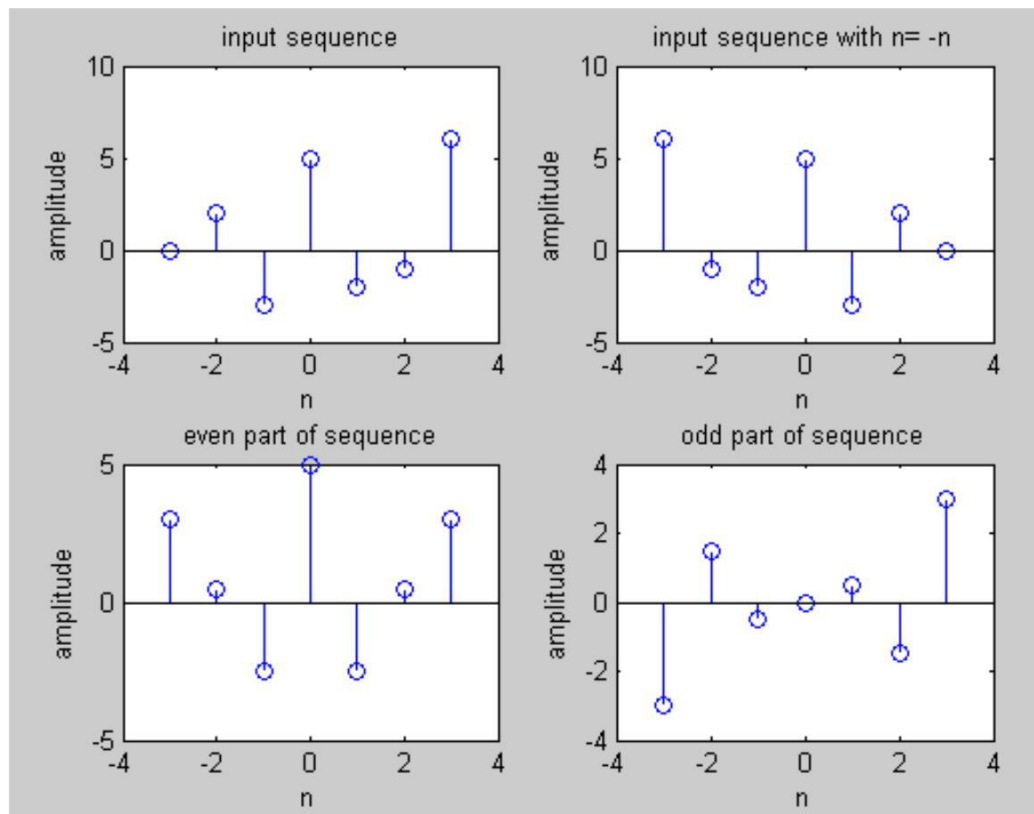
% plotting real and imaginary parts of the
signal x2=sin(t)+j*cos(t);
figure;
subplot(3,1,1);
plot(t,x2); xlabel('t');
ylabel('amplitude'); title('input
signal'); subplot(3,1,2)
plot(t,real(x2)); xlabel('time');
ylabel('amplitude'); title('real
part of signal'); subplot(3,1,3)
plot(t,imag(x2)); xlabel('time');
ylabel('amplitude');
title('imaginary part of siganl');

```

RESULT: Even and odd part of the signal and sequence, real and imaginary parts of signal are computed.

Output:





VIVA QUESTIONS:-

1. What is the formula to find odd part of signal?
2. What is Even Signal?
3. What is Odd Signal?
4. What is the formula to find even part of signal?
5. What is the difference b/w stem & plot?

Experiment No-5

Convolution between signals & sequences

Aim: Write the program for convolution between two signals and also between two sequences.

Software Required: Matlab software

Theory:



Convolution involves the following operations.

1. Folding
2. Multiplication
3. Addition
4. Shifting

$$y[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot \delta(n-k)$$

These operations can be represented by a Mathematical Expression as follows:

$x[n]$ = Input signal Samples

$h[n-k]$ = Impulse response co-efficient.

$y[n]$ = Convolution output.

n = No. of Input samples

h = No. of Impulse response co-efficient.

Example : $X(n) = \{1 \ 2 \ -1 \ 0 \ 1\}$, $h(n) = \{1, 2, 3, -1\}$

Program:

```

clc;
close all;
clear all;

%program for convolution of two
sequences x=input('enter input sequence: ');
h=input('enter impulse response: ');
y=conv(x,h);
subplot(3,1,1);
stem(x);
xlabel('n');
ylabel('x(n)');
title('input sequence')
subplot(3,1,2);
stem(h);
xlabel('n');
ylabel('h(n)');
title('impulse response sequence')
subplot(3,1,3);
stem(y);
xlabel('n');
ylabel('y(n)');
title('linear convolution')
disp('linear convolution y=');
  
```



```
disp(y)
%program for signal convolution
t=0:0.1:10;
x1=sin(2*pi*t);
h1=cos(2*pi*t);
y1=conv(x1,h1);
figure;
subplot(3,1,1);
plot(x1);
xlabel('t');
ylabel('x(t)');
title('input signal')
subplot(3,1,2);
plot(h1);
xlabel('t');
ylabel('h(t)');
title('impulse response')
subplot(3,1,3);
plot(y1);
xlabel('n');
ylabel('y(n)');
title('linear convolution');
```

RESULT: convolution between signals and sequences is computed.

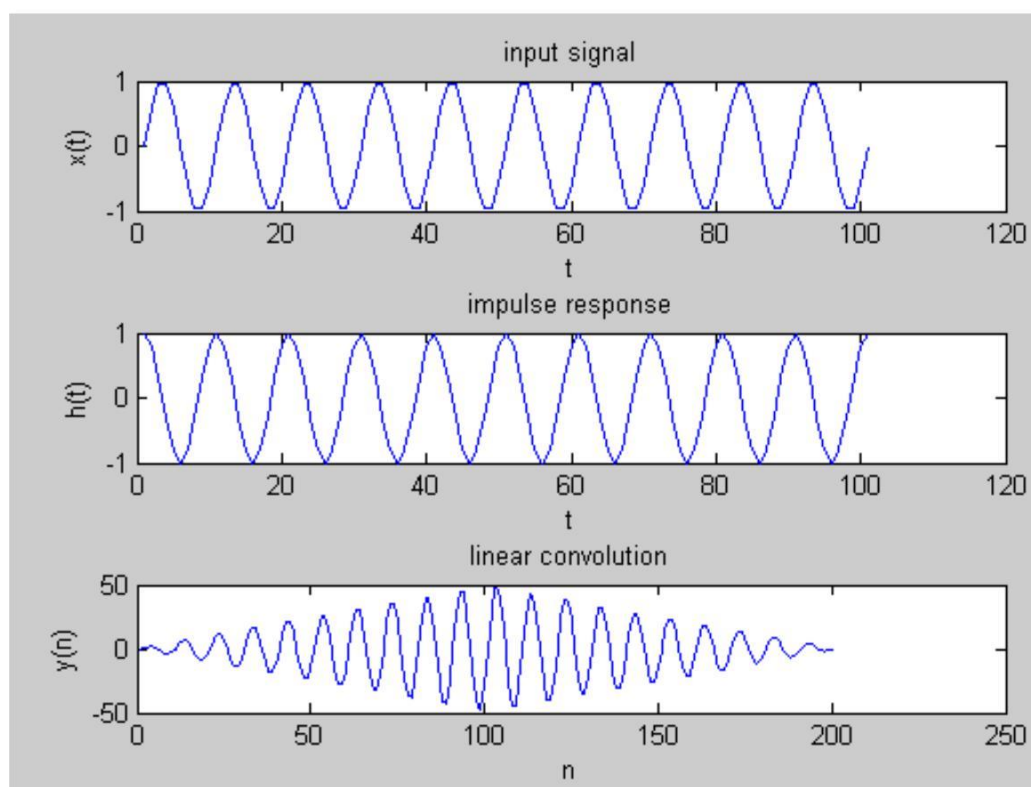
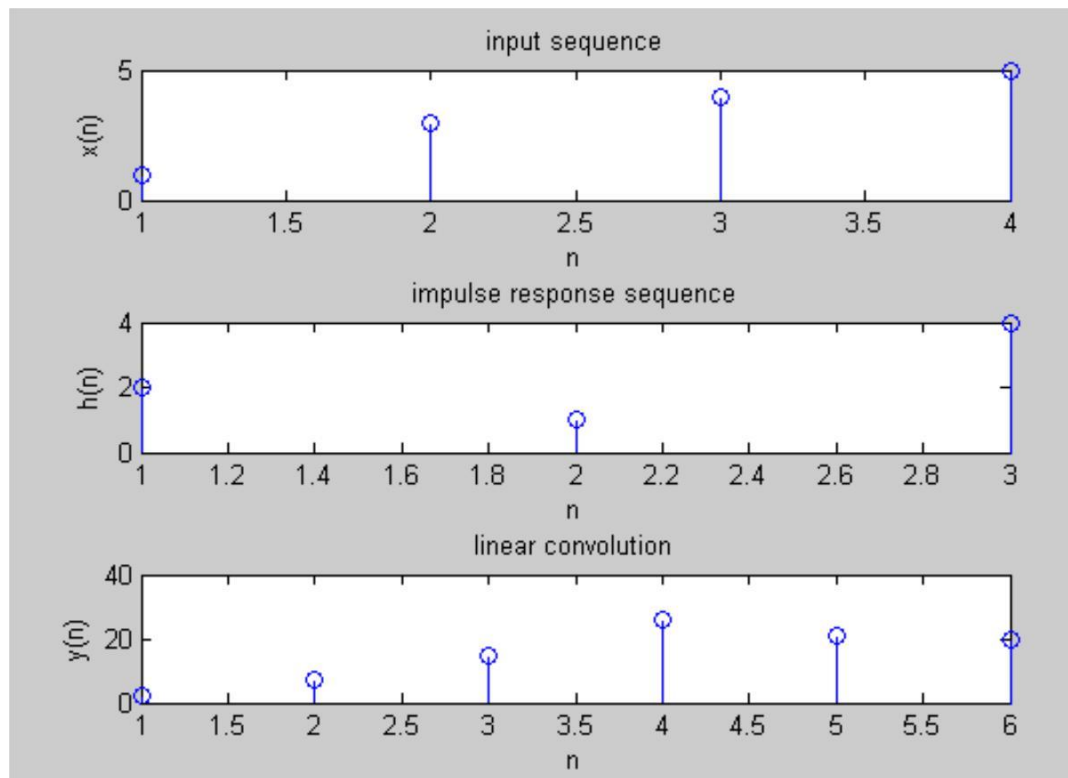
Output:

enter input sequence: [1 3 4 5]

enter impulse response: [2 1 4]

linear convolution y=

2 7 15 26 21 20



VIVA QUESTIONS:-

1. Define Convolution?
2. Define Properties of Convolution?
3. What is the Difference between Convolution& Correlation?
4. Define impulse response?
5. What is Half Wave Symmetry?

Experiment No-6

Auto correlation and Cross correlation

Aim: To compute Auto correlation and Cross correlation between signals and sequences.

Software Required: Mat lab software

Theory:

Correlations of sequences:

It is a measure of the degree to which two sequences are similar. Given two real-valued sequences $x(n)$ and $y(n)$ of finite energy,

Convolution involves the following operations.

1. Shifting
2. Multiplication
3. Addition

These operations can be represented by a Mathematical Expression as follows:

Cross correlation

$$r_{x,y}(l) = \sum_{n=-\infty}^{+\infty} x(n) y(n-l)$$

The index l is called the shift or lag parameter

Autocorrelation

$$r_{x,x}(l) = \sum_{n=-\infty}^{+\infty} x(n) x(n-l)$$

Program:

```
clc;
close all;
clear all;

% two input sequences
x=input('enter input sequence');
h=input('enter the impulse
sequence'); subplot(2,2,1);
stem(x);
xlabel('n');
ylabel('x(n)');
title('input sequence');
subplot(2,2,2);
stem(h);
xlabel('n'); ylabel('h(n)');
title('impulse
sequence');

% cross correlation between two sequences
```

```

y=xcorr(x,h);
subplot(2,2,3);
stem(y);
xlabel('n');
ylabel('y(n)');
title(' cross correlation between two sequences ');
% auto correlation of input sequence
z=xcorr(x,x);
subplot(2,2,4);
stem(z);
xlabel('n');
ylabel('z(n)');
title('auto correlation of input sequence');

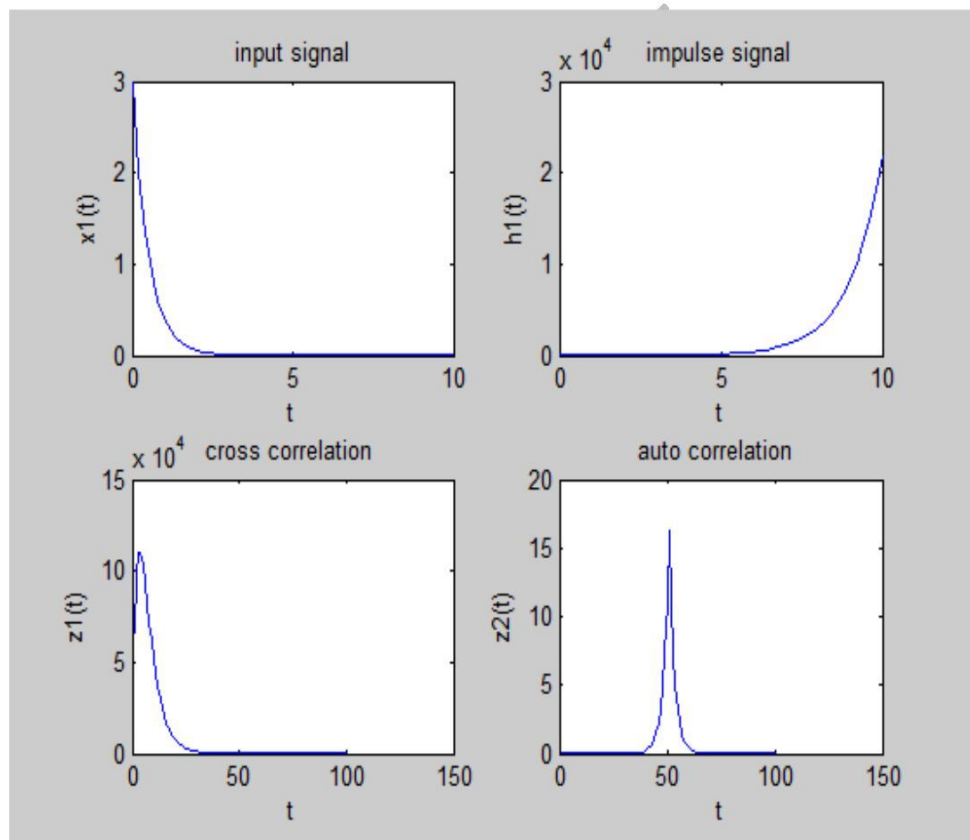
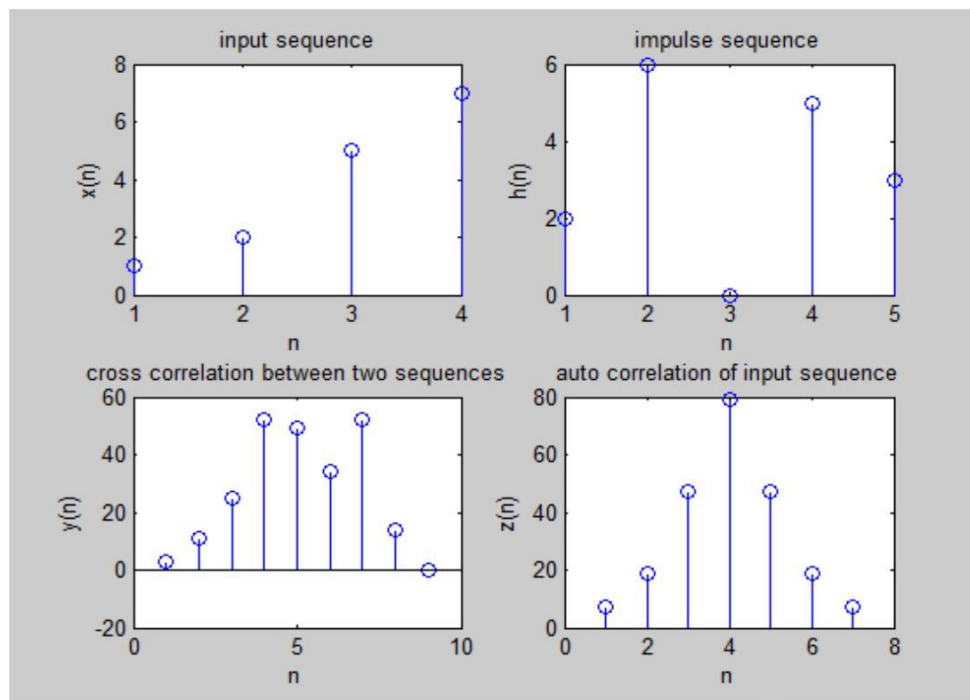
% cross correlation between two signals
% generating two input
signals t=0:0.2:10;
x1=3*exp(-2*t);
h1=exp(t);
figure;
subplot(2,2,1);
plot(t,x1);
xlabel('t');
ylabel('x1(t)');
title('input signal');
subplot(2,2,2);
plot(t,h1); xlabel('t');
ylabel('h1(t)');
title('impulse signal');

% cross correlation
subplot(2,2,3);
z1=xcorr(x1,h1);
plot(z1);
xlabel('t');
ylabel('z1(t)');
title('cross correlation ');
% auto correlation
subplot(2,2,4);
z2=xcorr(x1,x1);
plot(z2); xlabel('t');
ylabel('z2(t)');
title('auto correlation ');

```

Result: Auto correlation and Cross correlation between signals and sequences is computed.

Output: enter input sequence [1 2 5 7]
 enter the impulse sequence [2 6 0 5 3]



VIVA QUESTIONS:-

1. Define Correlation? And its properties?
2. Define Auto-Correlation?
3. Define Cross-Correlation?
4. What is the importance of correlation?
5. What is the difference b/w correlation and convolution?

Experiment No-7(a)

Verification of Linearity of a Discrete System

AIM: Verify the Linearity of a given Discrete System.

Software Required:

Mat lab software 7.0 and above

Theory:

LINEARITY PROPERTY:

Any system is said to be linear if it satisfies the superposition principal. superposition principal state that Response to a weighted sum of input signal equal to the corresponding weighted sum of the outputs of the system to each of the individual input signals.

If $x(n)$ is a input signal and $y(n)$ is a output signal then

$$y(n)=T[x(n)]$$

$$y_1(n)=T[x_1(n)] \quad \text{and} \quad y_2(n)=T[x_2(n)]$$

$$x_3=[a*x_1(n) +b *x_2(n)]$$

$$Y_3(n)= T [x_3(n)]$$

$$T [a*x_1(n)+b*x_2(n)] = a y_1(n)+ b y_2(n)$$

Program:

```
% Verification of Linearity of a given System.
% a)  $y(n)=nx(n)$    b)  $y=x^2(n)$ 
clc;
clear all;
close all;

n=0:40;
a1=input('enter the scaling factor a1=');
a2=input('enter the scaling factor a2=');
x1=cos(2*pi*0.1*n);
x2=cos(2*pi*0.4*n);
x3=a1*x1+a2*x2;
%y(n)=n.x(n);
y1=n.*x1;
y2=n.*x2;
y3=n.*x3;
yt=a1*y1+a2*y2;
yt=round(yt);
y3=round(y3);
if y3==yt
    disp('given system [y(n)=n.x(n)]is Linear');
else
    disp('given system [y(n)=n.x(n)]is non Linear');
end
%y(n)=x(n).^2
x1=[1 2 3 4 5];
x2=[1 4 7 6 4];
```



```
x3=a1*x1+a2*x2;  
y1=x1.^2;  
y2=x2.^2;  
y3=x3.^2;  
yt=a1*y1+a2*y2;  
if y3==yt  
    disp('given system [y(n)=x(n).^2 ]is Linear');  
else  
    disp('given system is [y(n)=x(n).^2 ]non Linear');  
end
```

Result: The Linearity of a given Discrete System is verified.

Output:

```
enter the scaling factor a1=3  
enter the scaling factor a2=5  
given system [y(n)=n.x(n)]is Linear  
given system is [y(n)=x(n).^2 ]non Linear
```

VIVA QUESTIONS:-

1. Define linear system with example?
2. Define non- linear system with example?
3. Define super position principle?
4. Give mathematical expression for Linearity.
5. Identity the system $y(n)=[x(n)]^3$ is linear or non-linear system.

Experiment No -7(b) Verification of Time Invariance of a Discrete System

AIM: Verify the Time Invariance of a given Discrete System.

Software Required: Mat lab software

Theory:

TIME INVARIANT SYSTEMS(TI):

A system is called time invariant if its input – output characteristics do not change with time

$X(t)$ ---- input : $Y(t)$ --- output
 $X(t-k)$ ----- delay input by k seconds : $Y(t-k)$ ----- Delayed output by k seconds

If $Y(t)=T[X(t)]$ then $Y(t-k)=T[X(t-k)]$ then system is time invariant system.

Program:

```
% Verification of Time Invariance of a Discrete System
% a) y=x^2(n)    b) y(n)=nx(n)
clc;
clear all;
close all;
n=1:9;
x(n)=[1 2 3 4 5 6 7 8 9];
d=3; % time delay
xd=[zeros(1,d),x(n)];%x(n-k)
y(n)=x(n).^2;
yd=[zeros(1,d),y];%y(n-k)
disp('transformation of delay signal yd:');disp(yd)
dy=xd.^2; % T[x(n-k)]
disp('delay of transformation signal dy:');disp(dy)
if dy==yd
    disp('given system [y(n)=x(n).^2] is time invariant');
else
    disp('given system is [y(n)=x(n).^2] not time invariant');
end
y=n.*x;
yd=[zeros(1,d),y(n)];
disp('transformation of delay signal yd:');disp(yd);
n1=1:length(xd);
dy=n1.*xd;
disp('delay of transformation signal dy:');disp(dy);
if yd==dy
    disp('given system [y(n)=nx(n)] is a time invariant');
else
    disp('given system [y(n)=nx(n)] not a time invariant');
end
```

Result: The Time Invariance of a given Discrete System is verified.

Output:

transformation of delay signal yd:

0 0 0 1 4 9 16 25 36 49 64 81

delay of transformation signal dy:

0 0 0 1 4 9 16 25 36 49 64 81

given system $[y(n)=x(n).^2]$ is time invariant

transformation of delay signal yd:

0 0 0 1 4 9 16 25 36 49 64 81

delay of transformation signal dy:

0 0 0 4 10 18 28 40 54 70 88 108

given system $[y(n)=nx(n)]$ not a time invariant

VIVA QUESTIONS:-

1. Define time invariant system with example?
2. Define time variant system with example?
3. Define LTI system?
4. Give mathematical expression for time invariant system?
5. Give another name for time invariant system and time variant system ?

Experiment No-8**Unit sample, unit step and sinusoidal response of the given LTI system and verifying its stability**

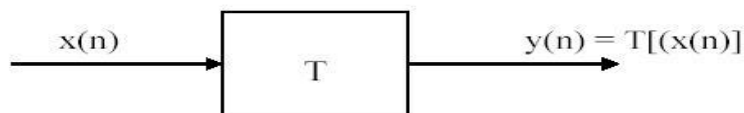
AIM: Compute the Unit sample, unit step and sinusoidal response of the given LTI system and verifying its stability

Software Required:

Mat lab software 7.0 and above

Theory:

A discrete time system performs an operation on an input signal based on predefined criteria to produce a modified output signal. The input signal $x(n]$ is the system excitation, and $y(n]$ is the system response. The transform operation is shown as,



If the input to the system is unit impulse i.e. $x(n) = \delta(n)$ then the output of the system is known as impulse response denoted by $h(n)$ where,

$$h(n) = T[\delta(n)]$$

we know that any arbitrary sequence $x(n)$ can be represented as a weighted sum of discrete impulses. Now the system response is given by,

$$y(n) = T[x(n)] = T\left[\sum_{k=-\infty}^{\infty} x(k) \delta(n-k)\right]$$

For linear system (1) reduces to

$$y(n) = \sum_{k=-\infty}^{\infty} x(k) T[\delta(n-k)]$$

% given difference equation $y(n) - y(n-1) + 0.9y(n-2) = x(n)$;

$$H(Z) = \frac{\sum_{k=0}^M b_k Z^{-(n-k)}}{\sum_{k=1}^N a_k Z^{-(n-k)}}$$

$$H(z) = \frac{b_0 + b_1 Z^{-1} + b_2 Z^{-2} + \dots + b_{N-1} Z^{-(N-1)} + b_N Z^{-N}}{1 + a_1 Z^{-1} + a_2 Z^{-2} + \dots + a_{N-1} Z^{-(N-1)} + a_N Z^{-N}}$$

Program:

```

%given difference equation  $y(n)-y(n-1)+.9y(n-2)=x(n)$ ; clc;
clear all;
close all;
b=[1];
a=[1,-1,.9];
n =0:3:100;

%generating impulse signal
x1=(n==0);
%impulse response
y1=filter(b,a,x1);
subplot(3,1,1);
stem(n,y1);
xlabel('n');
ylabel('y1(n)');
title('impulse response');

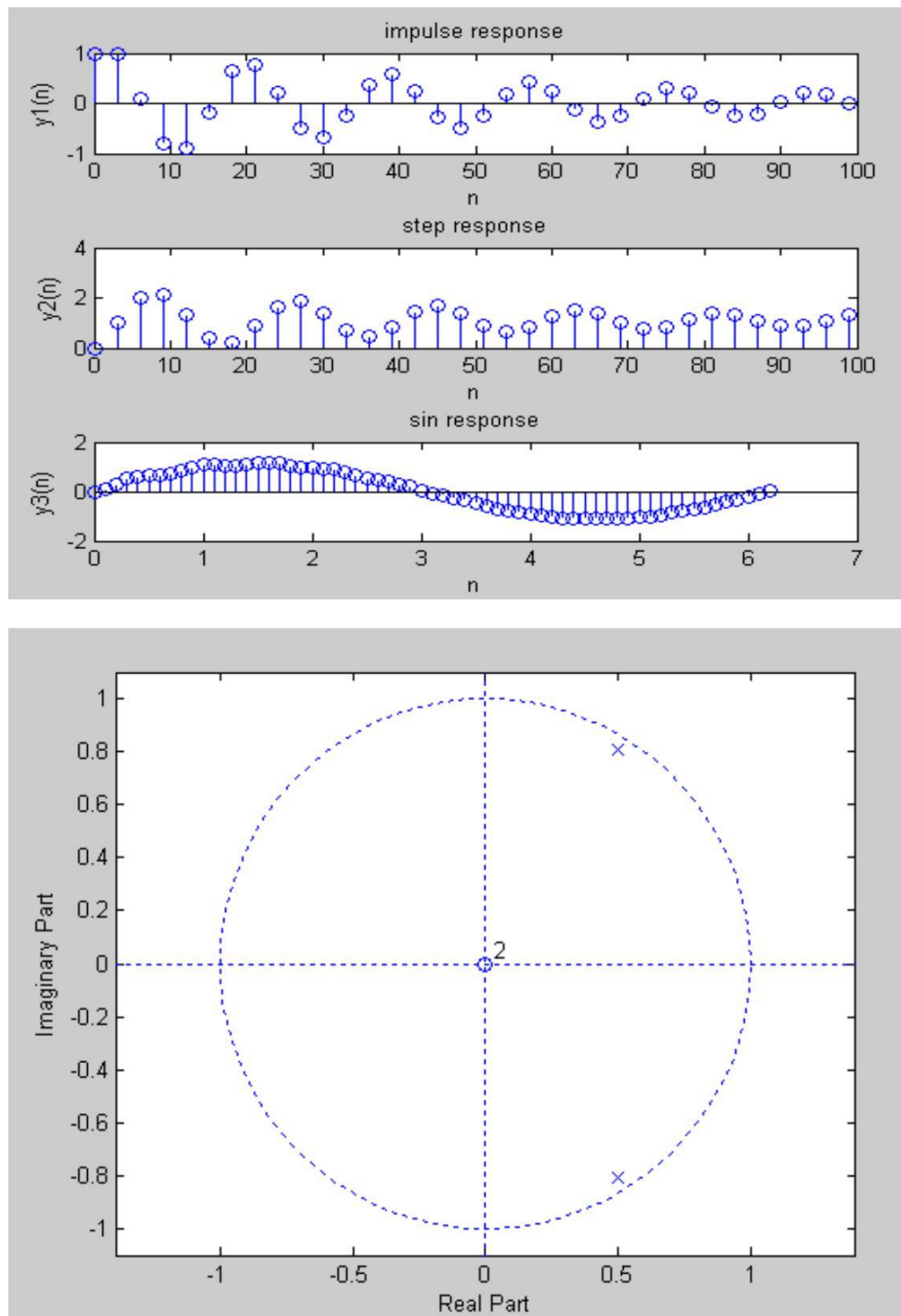
%generating step signal
x2=(n>0);
% step response
y2=filter(b,a,x2);
subplot(3,1,2); stem(n,y2);
xlabel('n'); ylabel('y2(n)');
title('step response');
%generating sinusoidal
signal t=0:0.1:2*pi;

x3=sin(t);
% sinusoidal response
y3=filter(b,a,x3);
subplot(3,1,3);
stem(t,y3);
xlabel('n');
ylabel('y3(n)');
title('sin response');

% verifying stability
figure;
zplane(b,a);

```

Result: The Unit sample, unit step and sinusoidal response of the given LTI system is computed and its stability verified. Hence all the poles lie inside the unit circle, so system is stable.

Output:**VIVA QUESTIONS:-**

1. What operations can be performed on signals and sequence?
2. Define causality and stability?
3. Define step response and impulse response of the system.
4. Define poles and zeros of the system?
5. What is the function of filter?

Experiment No-9

Gibbs phenomenon

AIM: Verify the Gibbs phenomenon.

Software Required: Matlab software

Theory:

The **Gibbs phenomenon**, the Fourier series of a piecewise continuously differentiable periodic function behaves at a jump discontinuity. The n the approximated function shows amounts of ripples at the points of discontinuity. This is known as the Gibbs Phenomena . partial sum of the Fourier series has large oscillations near the jump, which might increase the maximum of the partial sum above that of the function itself. The overshoot does not die out as the frequency increases, but approaches a finite limit

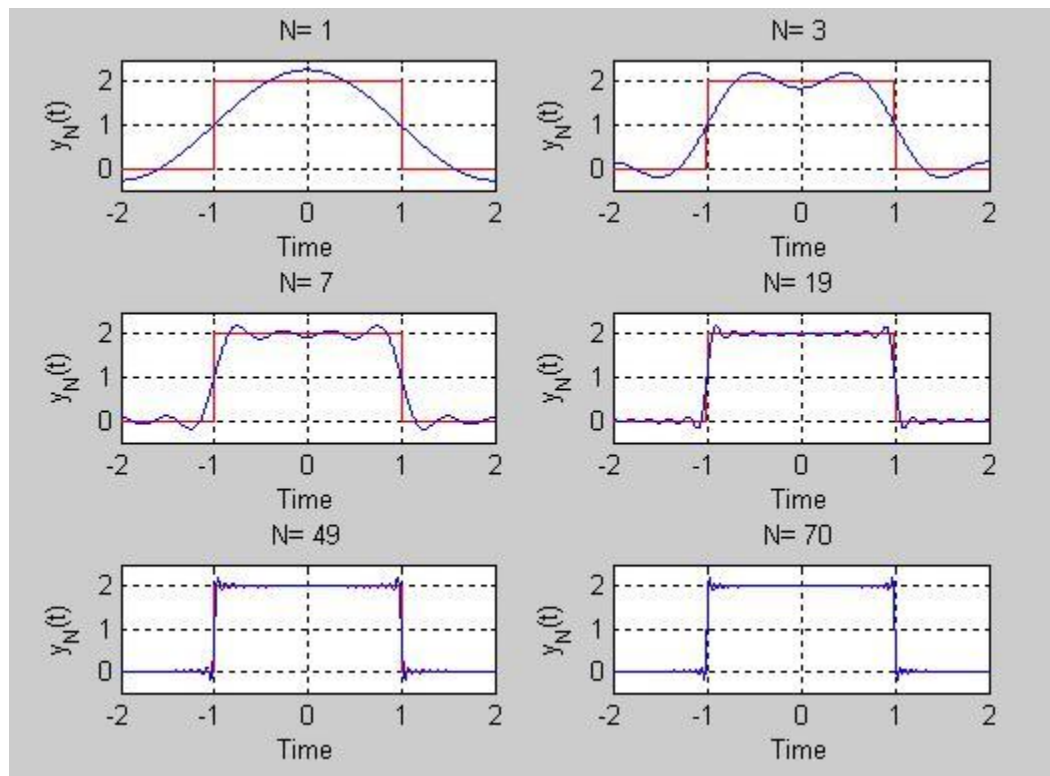
The Gibbs phenomenon involves both the fact that Fourier sums overshoot at a jump discontinuity, and that this overshoot does not die out as the frequency increases.

Program:

```
% Gibb's
phenomenon.. clc;
clear all; close all; t=linspace(-2,2,2000);
u=linspace(-2,2,2000);
sq=[zeros(1,500),2*ones(1,1000),zeros(1,500)]
; k=2;

N=[1,3,7,19,49,70]
; for n=1:6;
an=[];
for m=1:N(n)
    an=[an,2*k*sin(m*pi/2)/(m*pi)];
end;
fN=k/2; for
m=1:N(n)
    fN=fN+an(m)*cos(m*pi*t/2);
end;
nq=int2str(N(n));
subplot(3,2,n),plot(u,sq,'r');hold on; plot(t,fN);
hold off; axis([-2 2 -0.5 2.5]);grid;
xlabel('Time'), ylabel('y_N(t)');title(['N=
',nq]); end;
```

Result: In this experiment Gibbs phenomenon have been demonstrated using MATLAB.

Output:**VIVA QUESTIONS:-**

1. Define Gibb's Phenomenon?
2. What is the importance of Gibb's Phenomenon?
3. What is Static and Dynamic System?
4. Define Fourier series ?
5. What is Causality Condition of the Signal?

Experiment No-10

Finding the Fourier Transform of a given signal and plotting its magnitude and phase spectrum

AIM: To find the Fourier Transform of a given signal and plotting its magnitude and phase spectrum.

Software Required: Matlab software

Theory:

Fourier Transform:

The Fourier transform as follows. Suppose that f is a function which is zero outside of some interval $[-L/2, L/2]$. Then for any $T \geq L$ we may expand f in a Fourier series on the interval $[-T/2, T/2]$, where the "amount" of the wave $e^{2\pi i n x/T}$ in the Fourier series of f is given by

By definition Fourier Transform of signal $f(t)$ is defined as

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$

Inverse Fourier Transform of signal $F(\omega)$ is defined as

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega$$

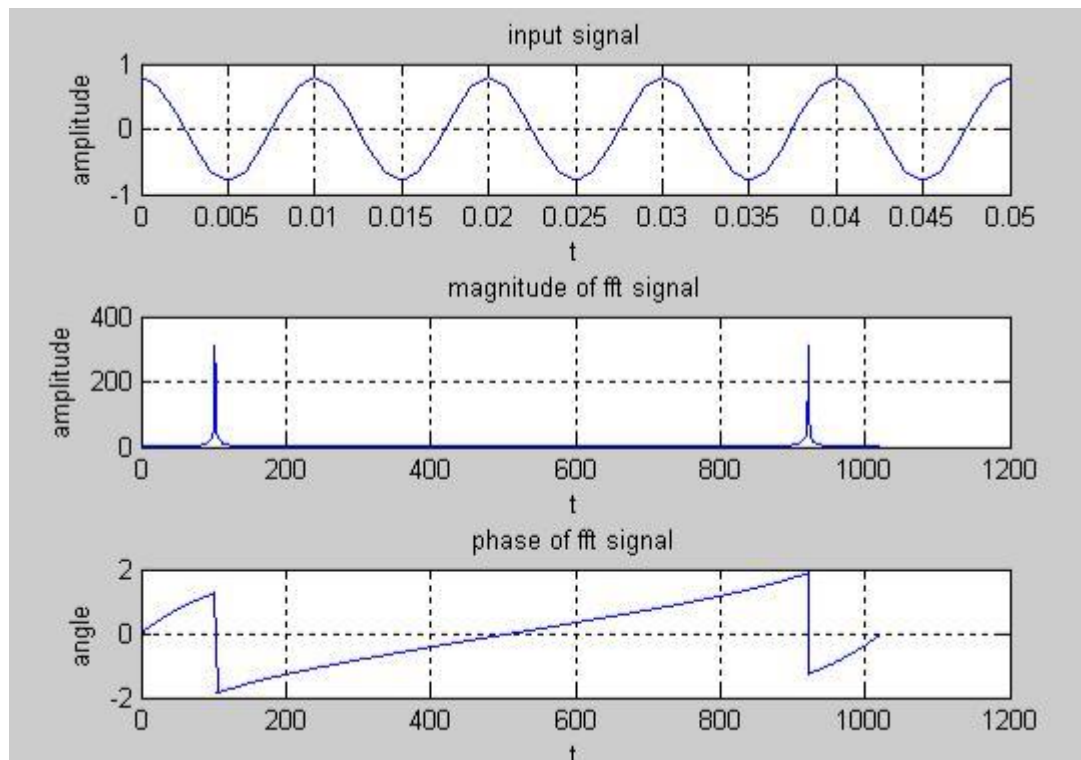
Program:

```
clc;
clear all;
close all ;
fs=1000;
N=1024; % length of fft sequence
t=[0:N-1]*(1/fs);
% input signal
x=0.8*cos(2*pi*100*t);
subplot(3,1,1);
plot(t,x);
axis([0 0.05 -1 1]);
grid; xlabel('t');
ylabel('amplitude');
title('input signal');

% Fourier transform of given
signal x1=fft(x);
% magnitude spectrum
k=0:N-1;
Xmag=abs(x1);
subplot(3,1,2);
```

```
plot(k,Xmag);  
grid;  
xlabel('t');  
ylabel('amplitude');  
title('magnitude of fft signal')  
%phase spectrum  
Xphase=angle(x1);  
subplot(3,1,3);  
plot(k,Xphase);  
grid;  
xlabel('t');  
ylabel('angle');  
title('phase of fft signal');
```

Result: Magnitude and phase spectrum of FFT of a given signal is plotted.

Output:**VIVA QUESTIONS:**

1. Define convolution property of Fourier transform?
2. What are the properties of Continuous-Time Fourier transform?
3. What is the sufficient condition for the existence of F.T?
4. Define the F.T of a signal?
5. What is the difference b/w F.T&F.S?

Experiment No-11

Waveform Synthesis Using Laplace transforms

AIM: Finding the Laplace transform & Inverse Laplace transform of some signals.

Software Required: Matlab software

Theory:

Bilateral Laplace transforms:

The Laplace transform of a signal $f(t)$ can be defined as follows:

$$F(s) = \mathcal{L}\{f(t)\} = \int_{-\infty}^{\infty} e^{-st} f(t) dt.$$

Inverse Laplace transform

The inverse Laplace transform is given by the following formula :

$$f(t) = \mathcal{L}^{-1}\{F(s)\} = \frac{1}{2\pi i} \lim_{T \rightarrow \infty} \int_{\gamma-iT}^{\gamma+iT} e^{st} F(s) ds,$$

Program:

```
clc;
clear all;
close all;
%representation of symbolic variables
syms f t w s;
%laplace transform of t
f=t;
z=laplace(f);
disp('the laplace transform of f = '); disp(z);

% laplace transform of a signal
%f1=sin(w*t); f1=-1.25+3.5*t*exp(-2*t)+1.25*exp(-2*t); v=laplace(f1);

disp('the laplace transform of f1 = '); disp(v);
lv=simplify(v);
pretty(lv)
%inverse laplace
transform y1=ilaplace(z);
disp('the inverse laplace transform of z = ');
disp(y1);
y2=ilaplace(v);
disp('the inverse laplace transform of v = ');
disp(y2);
ezplot(y1);
figure;
ezplot(y2)
```

Output:

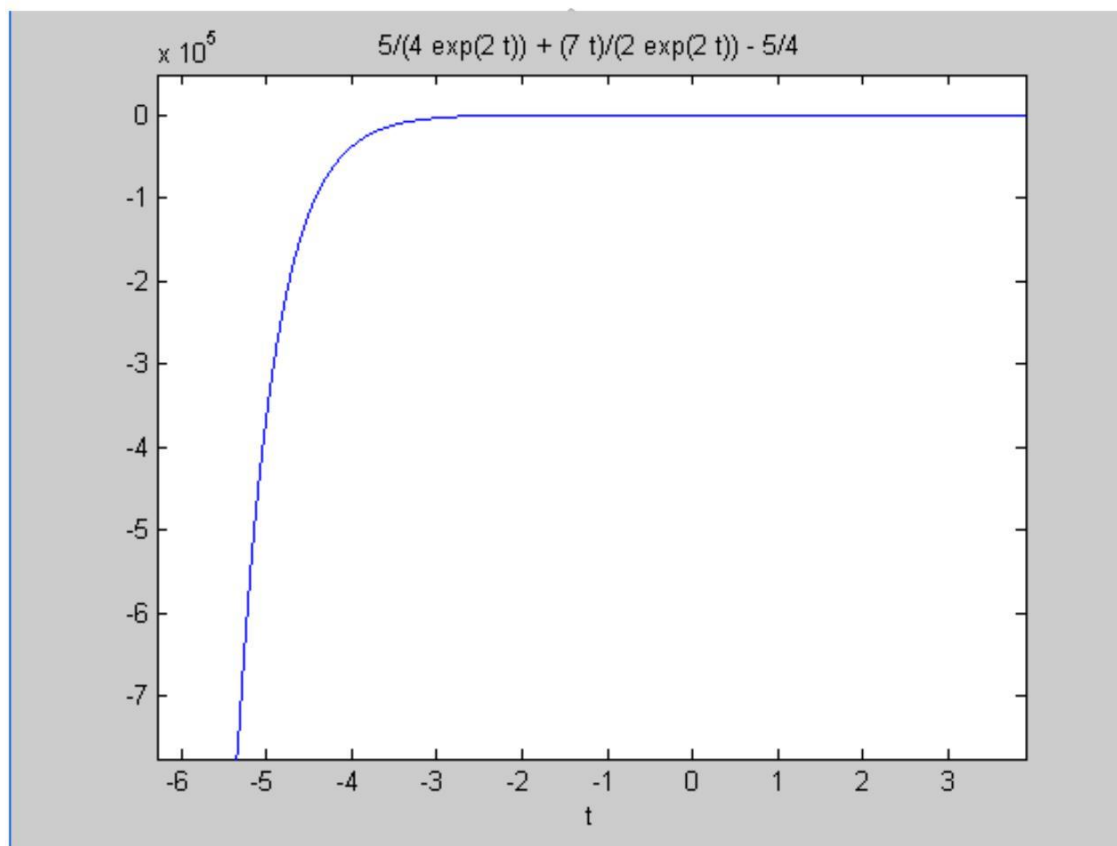
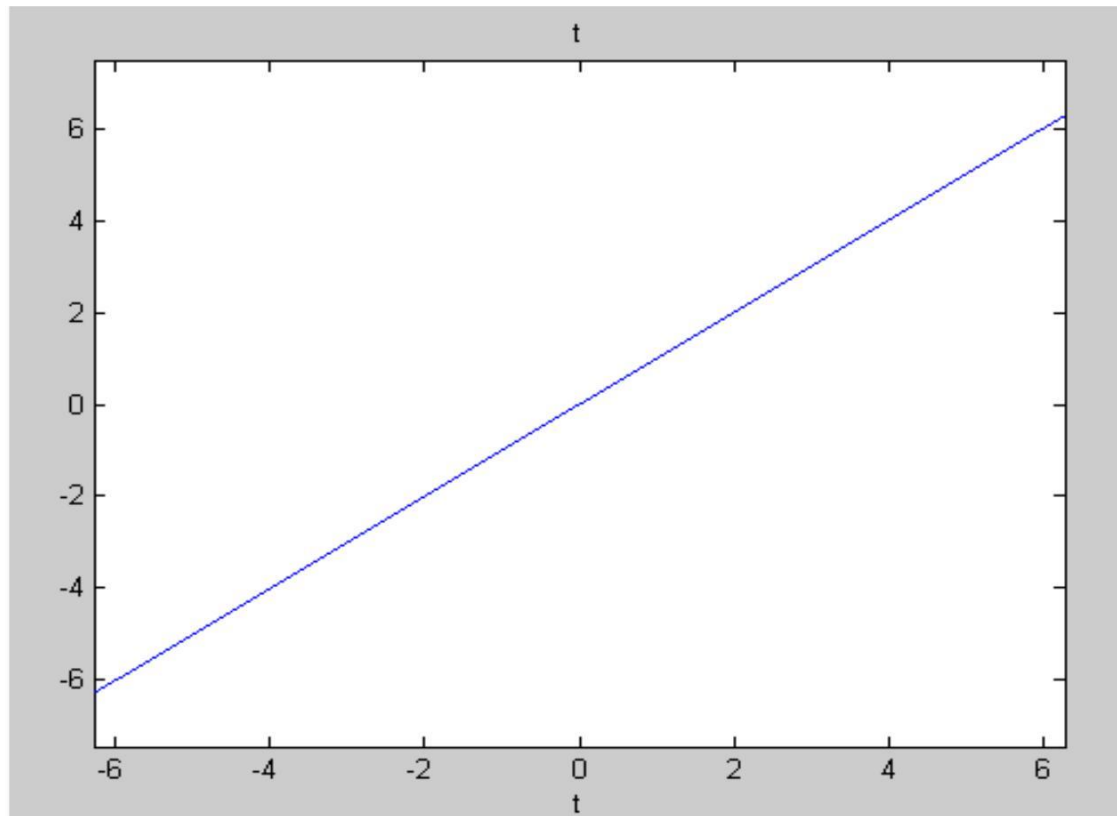
the laplace transform of f =
 $1/s^2$

the laplace transform of f1 =
 $5/(4*(s + 2)) + 7/(2*(s + 2)^2) - 5/(4*s)$

$$\frac{s - 5}{s(s + 2)^2}$$

 the inverse laplace transform of z =
 t

the inverse laplace transform of v =
 $5/(4*\exp(2*t)) + (7*t)/(2*\exp(2*t)) - 5/4$

Output:

VIVA QUESTIONS:-

1. Define Laplace-Transform?
2. What is the Condition for Convergence of the L.T?
3. What is the Region of Convergence (ROC)?
4. State the Shifting property of L.T?
5. State convolution Property of L.T?

Experiment No-12

Locating Poles and Zeros in s-plane & z-plane

AIM: Write the program for locating poles and zeros and plotting pole-zero maps in s-plane and z-plane for the given transfer function.

Software Required: Matlab software

Theory:

Z-transforms

The Z-transform, like many other integral transforms, can be defined as either a *one-sided* or *two-sided* transform.

Bilateral Z-transform

The *bilateral* or *two-sided* Z-transform of a discrete-time signal $x[n]$ is the function $X(z)$ defined as

$$X(z) = \mathcal{Z}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

Unilateral Z-transform

Alternatively, in cases where $x[n]$ is defined only for $n \geq 0$, the *single-sided* or *unilateral* Z-transform is defined as

$$X(z) = \mathcal{Z}\{x[n]\} = \sum_{n=0}^{\infty} x[n]z^{-n}$$

In signal processing, this definition is used when the signal is causal.

where $z = r.e^{j\omega}$

$$X(z) = \frac{P(z)}{Q(z)}$$

The roots of the equation $P(z) = 0$ correspond to the 'zeros' of $X(z)$

The roots of the equation $Q(z) = 0$ correspond to the 'poles' of $X(z)$

Example:

The zeros are: $\{-1\}$

$$H(z) = \frac{z+1}{(z-\frac{1}{2})(z+\frac{3}{4})}$$

The poles are: $\left\{ \frac{1}{2}, -\left(\frac{3}{4}\right) \right\}$

Program:

```
clc;
clear all;
close all;
%enter the numerator and denominator coefficients in square
brackets num=input('enter numerator co-efficients');
den=input('enter denominator co-efficients');
```



```
den=input('enter denominator co-efficients');

% find poles and zeros
poles=roots(den)
zeros=roots(num)

% find transfer function
H(s) h=tf(num,den);

% plot the pole-zero map in s-
plane sgrid;
pzmap(h)
; grid
on;
title('locating poles and zeros on s-plane');

%plot the pole zero map in z-plane
figure
zplane(poles,zeros);
grid on;
title('locating poles and zeros on z-plane');
```

Result: Pole-zero maps are plotted in s-plane and z-plane for the given transfer function.

Output:

enter numerator co-efficients[1 -1 4 3.5]

enter denominator co-efficients[2 3 -2.5 6]

poles =

-2.4874

0.4937 + 0.9810i

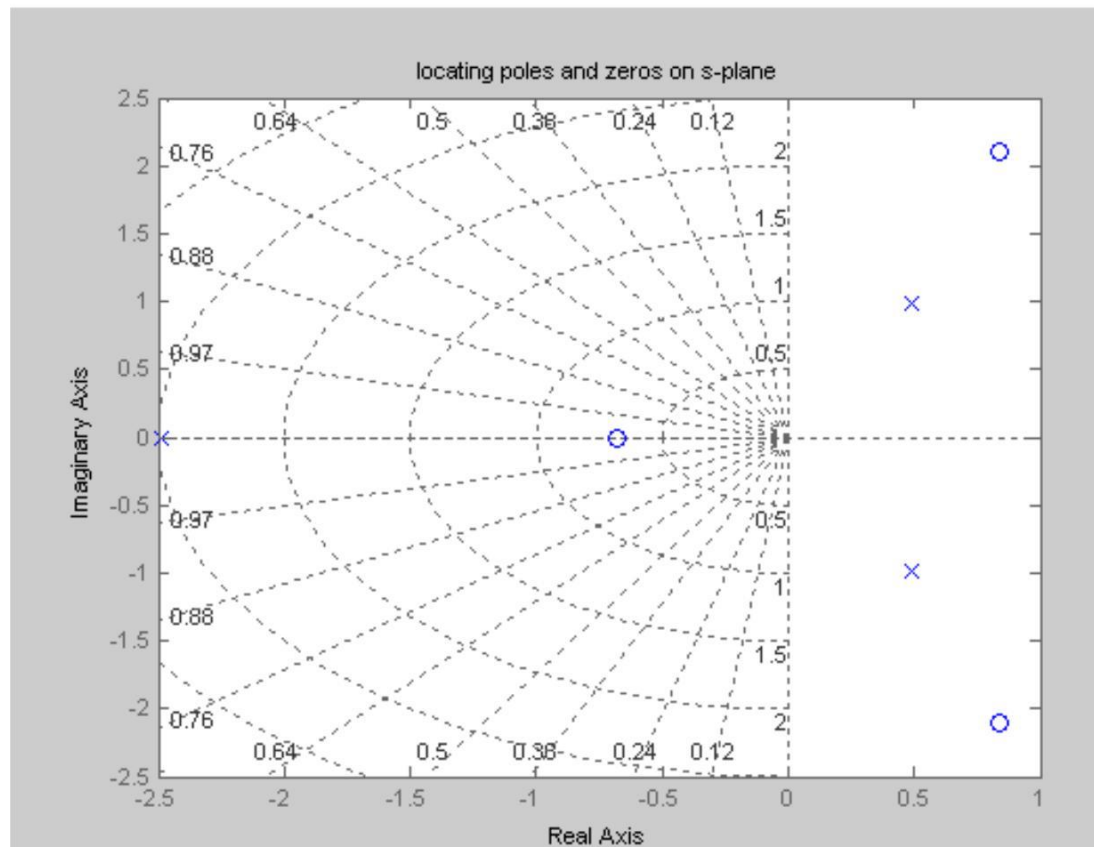
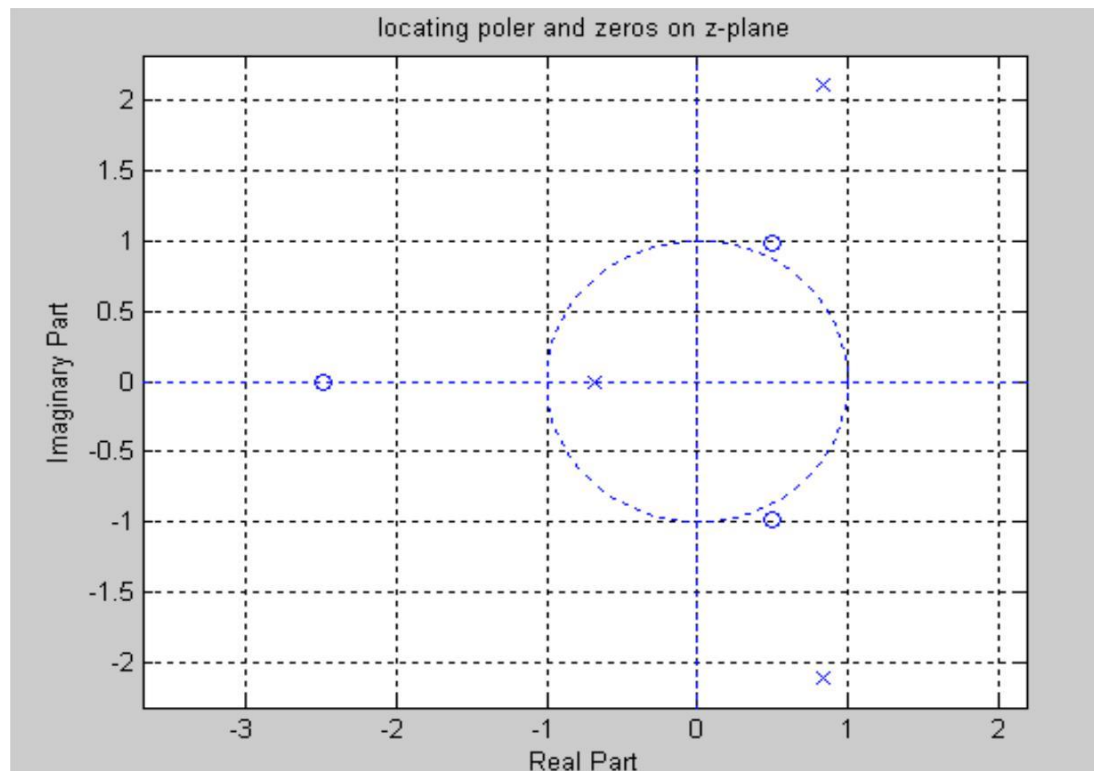
0.4937 - 0.9810i

zeros =

0.8402 + 2.1065i

0.8402 - 2.1065i

-0.6805



VIVA QUESTIONS:-

1. Study the details of pzmap() and zplane() functions?
2. What are poles and zeros?
3. How you specify the stability based on poles and zeros?
4. Define S-plane and Z-plane?
5. Define transfer function of the system?

Experiment No-13

Generation of Gaussian Noise

AIM: Write the program for generation of Gaussian noise and computation of its mean, mean square value, standard deviation, variance, and skewness.

Software Required: Matlab software

Theory:

Gaussian noise is statistical noise that has a probability density function (abbreviated pdf) of the normal distribution (also known as Gaussian distribution). In other words, the values that the noise can take on are Gaussian-distributed. It is most commonly used as additive white noise to yield additive white Gaussian noise (AWGN). Gaussian noise is properly defined as the noise with a Gaussian amplitude distribution. It says nothing of the correlation of the noise in time or of the spectral density of the noise. Labeling Gaussian noise as 'white' describes the correlation of the noise. It is necessary to use the term "white Gaussian noise" to be correct. Gaussian noise is sometimes misunderstood to be white Gaussian noise, but this is not the case.

Program:

```
clc;
clear all;
close all;

%generates a set of 2000 samples of Gaussian distributed random numbers
x=randn(1,2000);
%plot the joint distribution of both the sets using dot.
subplot(211)
plot(x, '.');
title('scatter plot of gaussian distributed random numbers');
ymu=mean(x)
ymsq=sum(x.^2)/length(x)
ysigma=std(x)
yvar=var(x)
yskew=skewness(x)

p=normpdf(x,ymu,ysigma);
subplot(212);
stem(x,p);
title(' gaussian distribution');
```

Output:

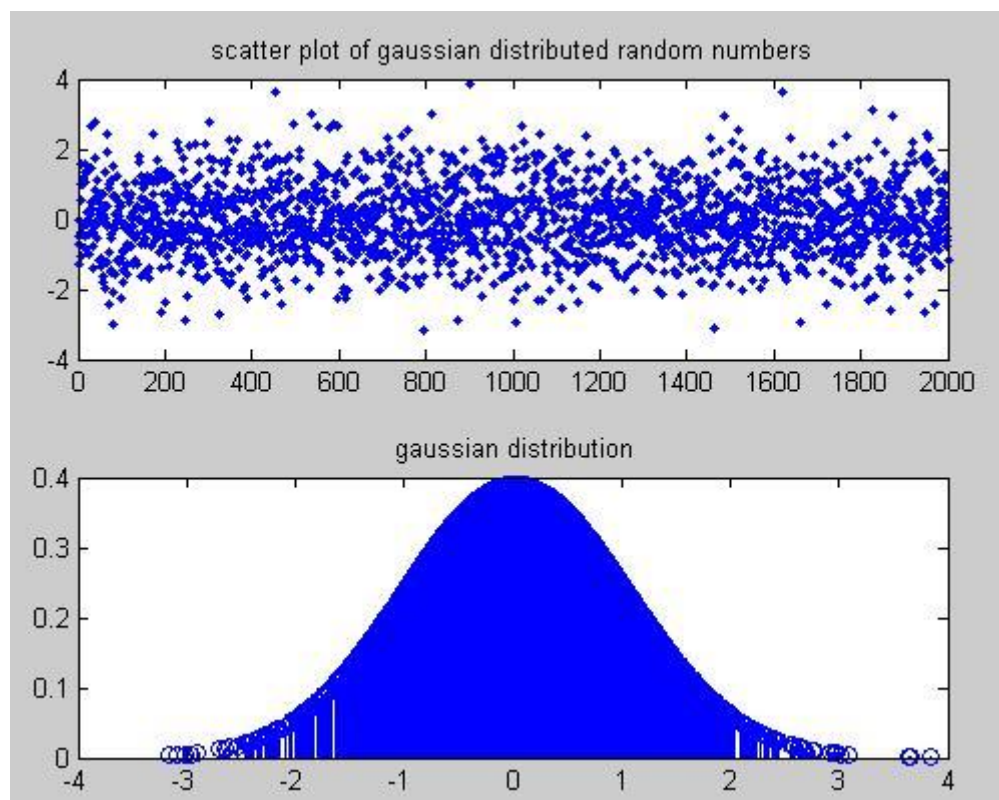
ymu = 0.0403

ymsq = 0.9727

ysigma = 0.9859

yvar = 0.9720

yskew = 0.0049

**VIVA QUESTIONS:-**

1. What is a noise and how many types of noises are there?
2. What is Gaussian noise?
3. What is correlation? How many types of correlation are there?
4. State Parseval's energy theorem for a periodic signal?
5. What is Signum function?

Experiment No-14

Sampling theorem verification

AIM: Verify the sampling theorem.

Software Required: Matlab software

Theory:

Sampling Theorem:

"A bandlimited signal can be reconstructed exactly if it is sampled at a rate atleast twice the maximum frequency component in it." Figure 1 shows a signal $g(t)$ that is bandlimited.

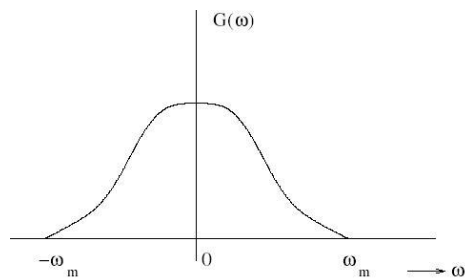


Figure 1: Spectrum of band limited signal $g(t)$

The maximum frequency component of $g(t)$ is f_m . To recover the signal $g(t)$ exactly from its samples it has to be sampled at a rate $f_s \geq 2f_m$.

The minimum required sampling rate $f_s = 2f_m$ is called 'Nyquist rate'

Proof: Let $g(t)$ be a bandlimited signal whose bandwidth is f_m ($\omega_m = 2\pi f_m$).



Figure 2: (a) Original signal $g(t)$ (b) Spectrum $G(\omega)$

$\delta(t)$ is the sampling signal with $f_s = 1/T > 2f_m$.

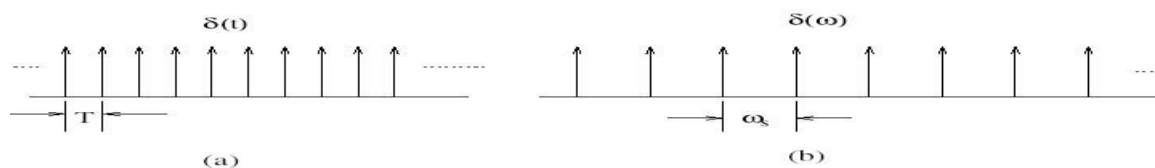
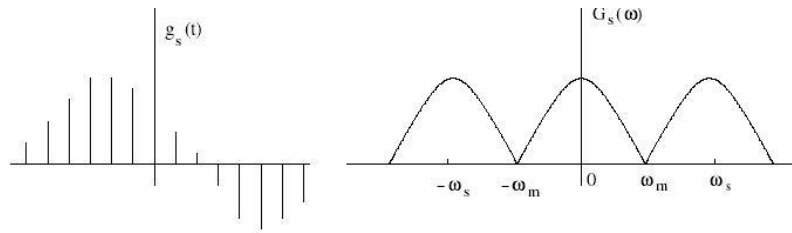


Figure 3: (a) sampling signal $\delta(t)$ (b) Spectrum $\delta(\omega)$

Figure 4: (a) sampled signal $g_s(t)$ (b) Spectrum $G_s(\omega)$

To recover the original signal $G(\omega)$:

1. Filter with a Gate function, $H_{2\omega_m}(\omega)$ of width $2\omega_m$
Scale it by T .

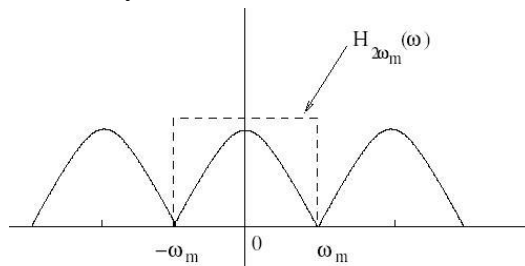
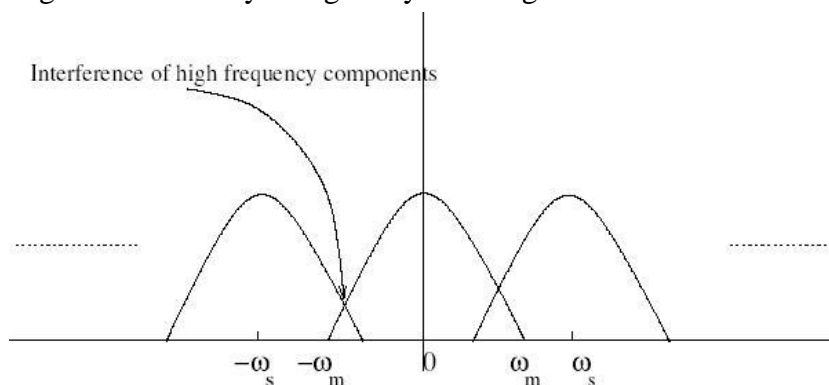
Figure 5: Recovery of signal by filtering with a filter of width $2\omega_m$ Aliasing $\omega_s < 2\omega_m$.

Figure 6: Aliasing due to inadequate sampling

Aliasing leads to distortion in recovered signal. This is the reason why sampling frequency should be at least twice the bandwidth of the signal. Oversampling $\omega_s > 2\omega_m$. This condition avoids aliasing.

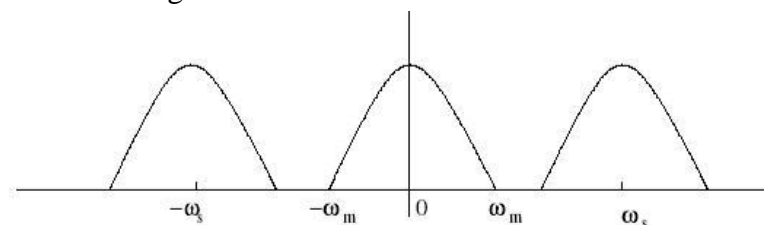


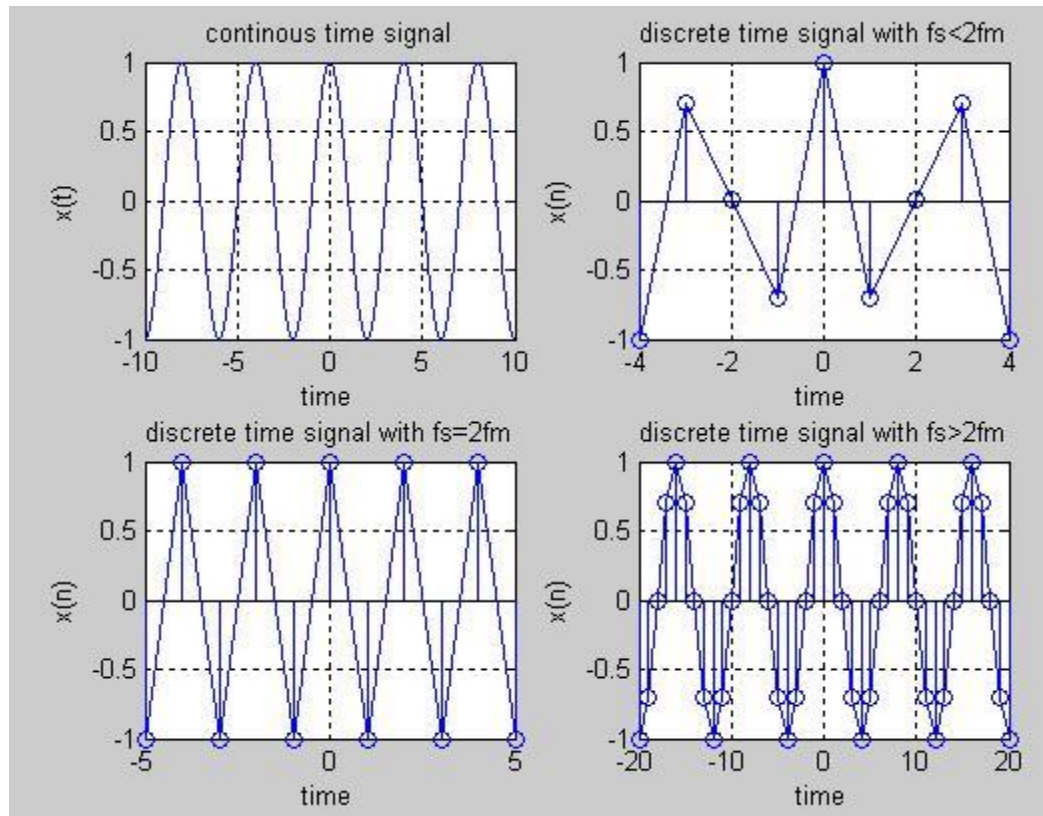
Figure 7: Oversampled signal-avoids aliasing

Program:

```
clc;
clear all;
close all;
t=-10:.01:10;
T=4;
fm=1/T;
x=cos(2*pi*fm*t);
subplot(2,2,1);
plot(t,x);
xlabel('time');
ylabel('x(t)');
title('continuous time signal');
grid;
n1=-4:1:4;
fs1=1.6*fm;
fs2=2*fm;
fs3=8*fm;
x1=cos(2*pi*fm/fs1*n1);
subplot(2,2,2);
stem(n1,x1);
xlabel('time');
ylabel('x(n)');
title('discrete time signal with fs<2fm');
hold on;
subplot(2,2,2);
plot(n1,x1);
grid;
n2=-5:1:5;
x2=cos(2*pi*fm/fs2*n2);
subplot(2,2,3);
stem(n2,x2);
xlabel('time');
ylabel('x(n)');
title('discrete time signal with fs=2fm');
hold on;
subplot(2,2,3);
plot(n2,x2);
grid;
n3=-20:1:20;
x3=cos(2*pi*fm/fs3*n3);
subplot(2,2,4);
stem(n3,x3);
xlabel('time');
ylabel('x(n)');
title('discrete time signal with fs>2fm');
hold on;
subplot(2,2,4);
plot(n3,x3);
grid;
```

Result: Sampling theorem is verified.

OUTPUT:



VIVA QUESTIONS:-

1. State Parseval's energy theorem for a periodic signal?
2. Define sampling Theorem?
3. What is Aliasing Effect?
4. What is under sampling?
5. What is over sampling?

EXP.No:15**REMOVAL OF NOISE BY AUTO CORRELATION/CROSS CORRELATION**

AIM: Write the program for Removal of noise by using auto correlation.

Software Required: Matlab software

Theory:

Detection of a periodic signal masked by random noise is of great importance .The noise signal encountered in practice is a signal with random amplitude variations. A signal is uncorrelated with any periodic signal. If $s(t)$ is a periodic signal and $n(t)$ is a noise signal then

$$\lim_{T \rightarrow -\infty} \frac{1}{T} \int_{-T/2}^{T/2} S(t)n(t-T) dt = 0 \quad \text{for all } T$$

$Q_{sn}(T)$ = cross correlation function of $s(t)$ and $n(t)$ Then $Q_{sn}(T) = 0$

Detection of noise by Auto-Correlation:

$S(t)$ = Periodic Signal (Transmitted) , mixed with a noise signal $n(t)$.

Then $f(t)$ is received signal is $[s(t) + n(t)]$

Let $Q_{ff}(T)$ = Auto Correlation Function of $f(t)$

$Q_{ss}(t)$ = Auto Correlation Function of $S(t)$

$Q_{nn}(T)$ = Auto Correlation Function of $n(t)$

$$\begin{aligned} Q_{ff}(T) &= \lim_{T \rightarrow -\infty} \frac{1}{T} \int_{-T/2}^{T/2} f(t)f(t-T) dt \\ &= \lim_{T \rightarrow -\infty} \frac{1}{T} \int_{-T/2}^{T/2} [s(t)+n(t)] [s(t-T)+n(t-T)] dt \\ &= Q_{ss}(T) + Q_{nn}(T) + Q_{sn}(T) + Q_{ns}(T) \end{aligned}$$

The periodic signal $s(t)$ and noise signal $n(t)$ are uncorrelated

$$Q_{sn}(t) = Q_{ns}(t) = 0 ;$$

Then $Q_{ff}(t) = Q_{ss}(t) + Q_{nn}(t)$

The Auto correlation function of a periodic signal is periodic of the same frequency and the Auto correlation function of a non periodic signal is tends to zero for large value of T since $s(t)$ is a periodic signal and $n(t)$ is non periodic signal so $Q_{ss}(T)$ is a periodic where as $Q_{nn}(T)$ becomes small for large values of T Therefore for sufficiently large values of T $Q_{ff}(T)$ is equal to $Q_{ss}(T)$.

Program:

```
clc;
clear all;
close all;
t=0:0.2:pi*8;
%input signal
s=sin(t);
subplot(6,1,1);
plot(s);
title('signal s');
xlabel('t');
ylabel('amplitude');
%generating noise
n = randn([1 126]);
%noisy signal
f=s+n;
subplot(6,1,2)
plot(f);
title('signal f=s+n');
xlabel('t');
ylabel('amplitude');
%autocorrelation of input signal
as=xcorr(s,s);
subplot(6,1,3);
plot(as);
title('auto correlation of s');
xlabel('t');
ylabel('amplitude');
%autocorrelation of noise signal
an=xcorr(n,n);
subplot(6,1,4)
plot(an);
title('auto correlation of n');
xlabel('t');
ylabel('amplitude');
%autocorrelation of transmitted signal
cff=xcorr(f,f);
subplot(6,1,5)
```

```

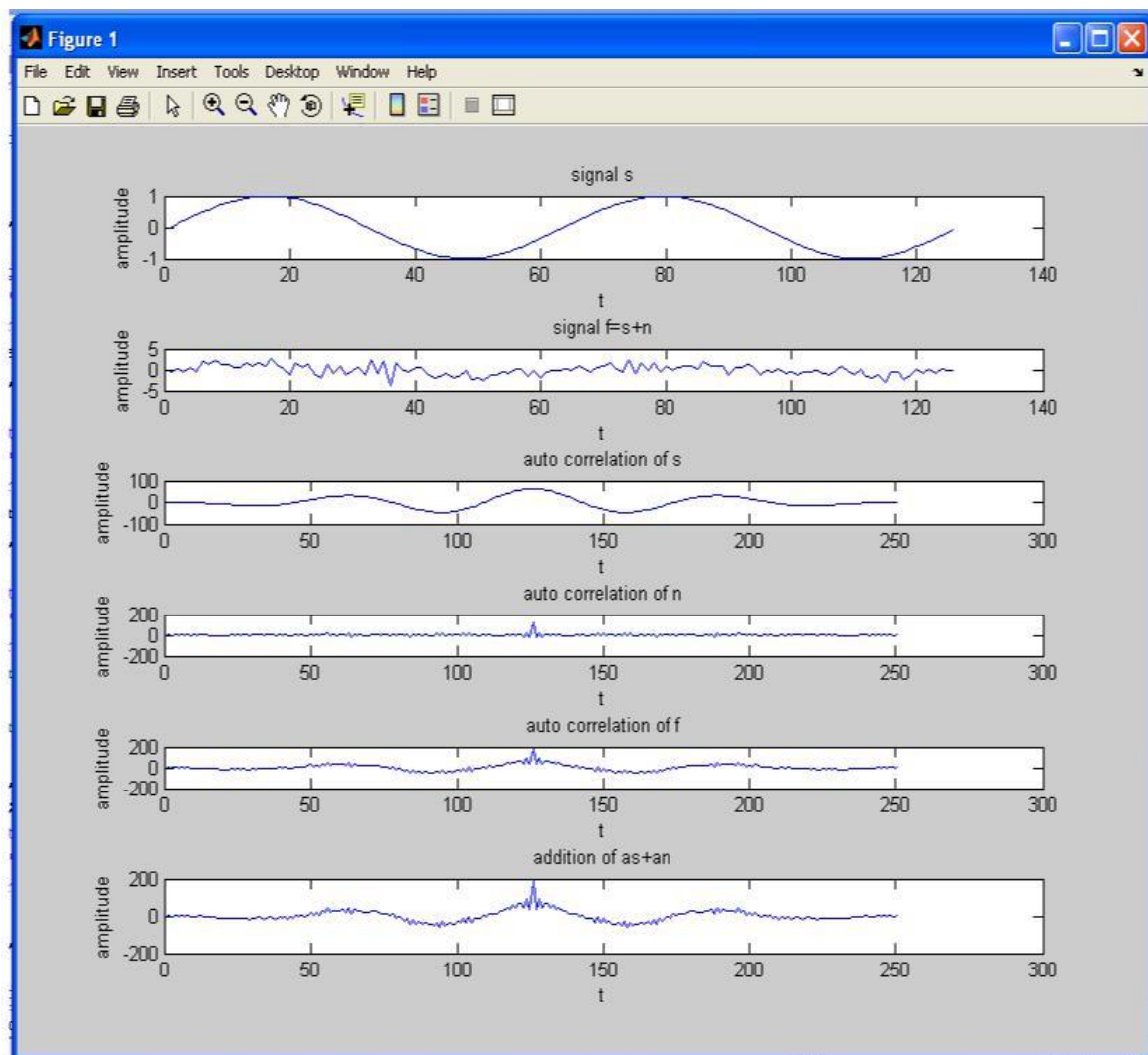
plot(cff);
title('auto correlation of f');
xlabel('t');
ylabel('amplitude');

% autocorrelation of received signal
hh=as+an;
subplot(6,1,6)
plot(hh);
title('addition of as+an');
xlabel('t');
ylabel('amplitude');

```

Result: Removal of noise using autocorrelation is performed.

Output:



VIVA QUESTIONS:-

1. Define Auto correlation function?
2. What are the Different types of noise signals?
3. Define cross correlation function?
4. What is Signum function?
5. What is Static and Dynamic System?

EXP.No:16
EXTRACTION OF PERIODIC SIGNAL MASKED BY NOISE USING
CORRELATION

AIM: Write the program for extraction of periodic signal using correlation.

Software Required: Matlab software

Theory:

A signal is masked by noise can be detected either by correlation techniques or by filtering. Actually, the two techniques are equivalent. The correlation technique is a measure of extraction of a given signal in the time domain whereas filtering achieves exactly the same results in frequency domain.

Program:

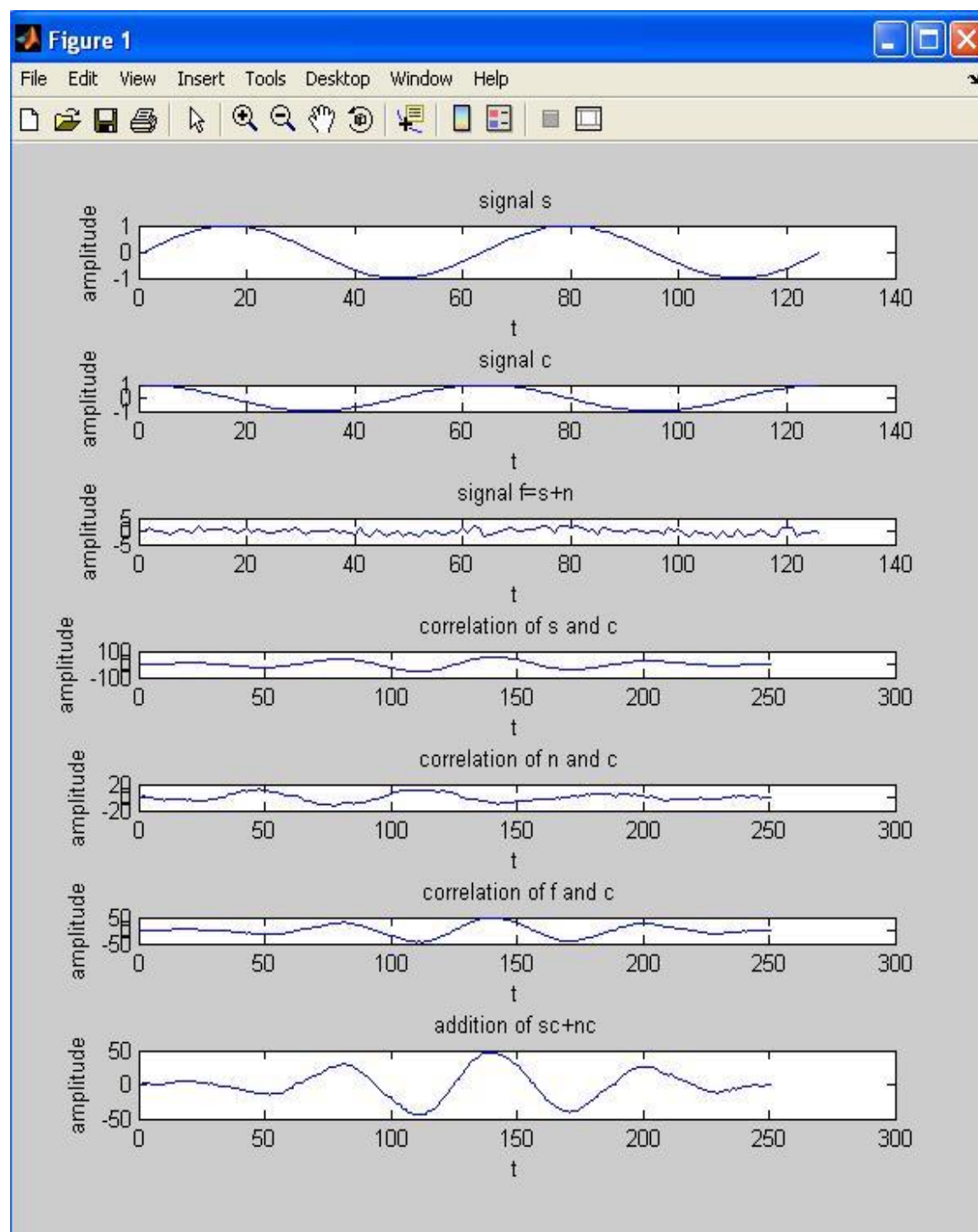
```
clear all;
close all;
clc;
t=0:0.1: pi*4;
%input signal1
s=sin(t);
subplot(7,1,1)
plot(s);
title('signal s');
xlabel('t');
ylabel('amplitude');
%input signal2
c=cos(t);
subplot(7,1,2)
plot(c);
title('signal c');
xlabel('t');
ylabel('amplitude');

%generating noise signal
n = randn([1 126]);
%signal+noise
f=s+n;
subplot(7,1,3);
plot(f);
title('signal f=s+n');
xlabel('t');
ylabel('amplitude');

%crosscorrelation of signal1&signal2
```

```
asc=xcorr(s,c);
subplot(7,1,4)
plot(asc);
title(' correlation of s and c');
xlabel('t');
ylabel('amplitude');
%crosscorrelation of noise&signal2
anc=xcorr(n,c);
subplot(7,1,5)
plot(anc);
title(' correlation of n and c');
xlabel('t');
ylabel('amplitude');
%crosscorrelation of f&signal2
cfc=xcorr(f,c);
subplot(7,1,6)
plot(cfc);
title(' correlation of f and c');
xlabel('t');
ylabel('amplitude');
%extracting periodic signal
hh=asc+anc;
subplot(7,1,7)
plot(hh);
title('addition of sc+nc');
xlabel('t');
ylabel('amplitude');
```

Result: Periodic signal is extracted by using correlation.

Output:**VIVA QUESTIONS:-**

1. State the relationship between PSD and ACF?
2. What is the integration of ramp signal?
3. Difference between vectors and signals?
4. Define PSD?
5. Define Hilbert transforms?

EXP.No:17**VERIFICATION OF WIENER-KHINCHIN RELATION**

AIM: Verification of wiener-khinchin relation**Software Required:** Matlab software**Theory:**

The wiener-khinchin theorem states that the power spectral density of a wide sense stationary random process is the flourier transform of the corresponding autocorrelation function.

$$\text{PSD} = S_{XX}(\omega) = FT[R_{XX}(\tau)] = \int_{-\infty}^{\infty} R_{XX}(\tau) e^{-j\omega\tau} d\tau$$

$$\text{ACF} = R_{XX}(\tau) = IFT[S_{XX}(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{XX}(\omega) e^{j\omega\tau} d\omega$$

Program:

```

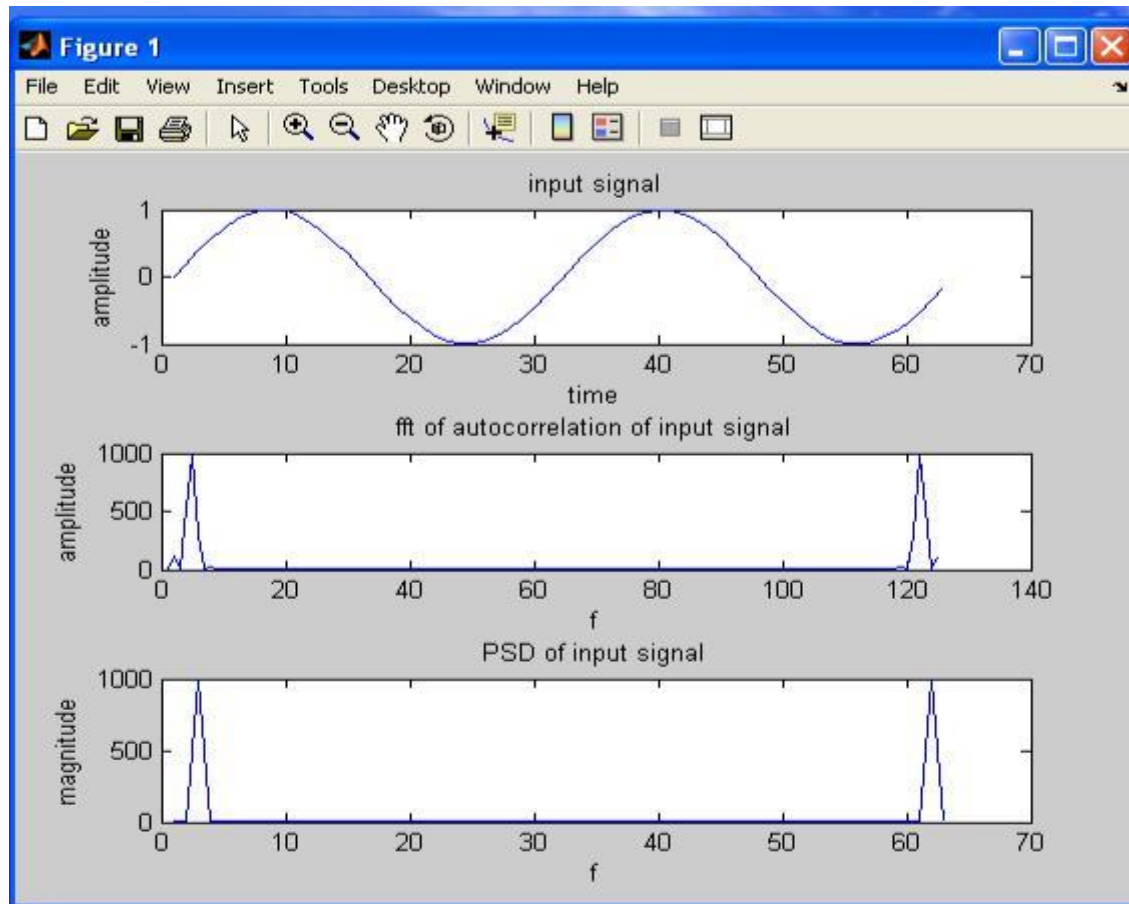
clc;
clear all;
close all;
t=0:0.1:2*pi;
%input signal
x=sin(2*t);
subplot(3,1,1);
plot(x);
xlabel('time');
ylabel('amplitude');
title('input signal');
%autocorrelation of input signal
xu=xcorr(x,x);
%fft of autocorrelation signal
y=fft(xu);
subplot(3,1,2);
plot(abs(y));
xlabel('f');
ylabel('amplitude');
title('fft of autocorrelation of input signal');
%fourier transform of input signal
y1=fft(x);
%finding the power spectral density
y2=(abs(y1)).^2;
subplot(3,1,3);
plot(y2);
xlabel('f');

```

```
ylabel('magnitude');  
title('PSD of input signal');
```

Result: wiener-khinchin relation is verified.

Output:



VIVA QUESTIONS:-

1. What is mean wiener khinchine relation?
2. Define fourier transform and its inverse?
3. What is the difference b/w convolution and correlation?
4. What is the importance of power spectrum?
5. What is the importance of correlation?

EXP.No:18

18. Generate a discrete time sequence of $N=1000$ i.i.d uniformly distributed random numbers in the interval $(-0.5,-0.5)$ and compute the autocorrelation of the sequence.

```
function [x rx]=randseq(n)
x=rand(1,n)-0.5;
rx=cor(x);
```

EXP.No:19

19. Obtain and plot the power spectrum of the output process when a white random process is passed through a filter with specific impulse response .

```
function sy=white_rand(w,H)
sy=abs(H).^2;
plot(w,sy)
```

