Signals and Systems Laboratory (EC2P002)

EXPERIMENT-5

Shorya Sharma

19EE01017

Aim of the experiment:

- Find out the magnitude and phase spectrum by performing Fourier transform operation on familiar signals such as sine, square, triangular, unit impulse, decaying exponential, unit step and pulse signals.
- Find the magnitude and phase spectrum of the Fourier transform of the signals given in 4.1 and 4.2.
- Validate the properties of Fourier transform using problems given in 4.3.

Theory: -

The **Fourier transform** (**FT**) decomposes a function of time (a *signal*) into its constituent frequencies. The FT for an integrable continuous-time signal x (t) can be given by:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}\,dt$$

The FT for a summable discrete-time sequence x (n) is given as:

$$X(j\omega) = \sum_{n=-\infty}^{\infty} x\,(n)e^{-j\omega n}$$

2.2 Properties of Fourier Transform

2.2.1 Linearity

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

and $y(t) \xrightarrow{\mathcal{F}} Y(j\omega)$

then $ax(t) + b\,y(t) \xrightarrow{\mathcal{F}} aX(j\omega) + bY(j\omega)$

2.2.2 Time Shifting

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

then $x(t - t_0) \xrightarrow{\mathcal{F}} exp(-j\omega t_0)\,X(j\omega)$

2.2.3 Conjugation and Conjugate Symmetry

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

then $x^*(t) \xrightarrow{\mathcal{F}} X^*(-j\omega)$

2.2.4 Differentiation and Integration

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

then $\frac{dx(t)}{dt} \xrightarrow{\mathcal{F}} j\omega\,X(j\omega)$

and $\int_x^t x(\tau)d\tau \xrightarrow{\mathcal{F}} \frac{1}{j\omega}\,X(j\omega) + \pi X(0)\delta(\omega)$

2.2.5 Time and Frequency scaling

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

then $x(at) \xrightarrow{\mathcal{F}} \frac{1}{|a|}X(\frac{j\omega}{a})$

2.2.6 Duality

If $x(t) \xrightarrow{\mathcal{F}} X(j\omega)$

then $X(t) \xrightarrow{\mathcal{F}} 2\pi x(-j\omega)$

2.2.7 Convolution

If $\quad x(t) \xrightarrow{\mathcal{F}} X(j\omega)$ and $h(t) \xrightarrow{\mathcal{F}} H(j\omega)$ and $y(t) = x(t) * h(t)$

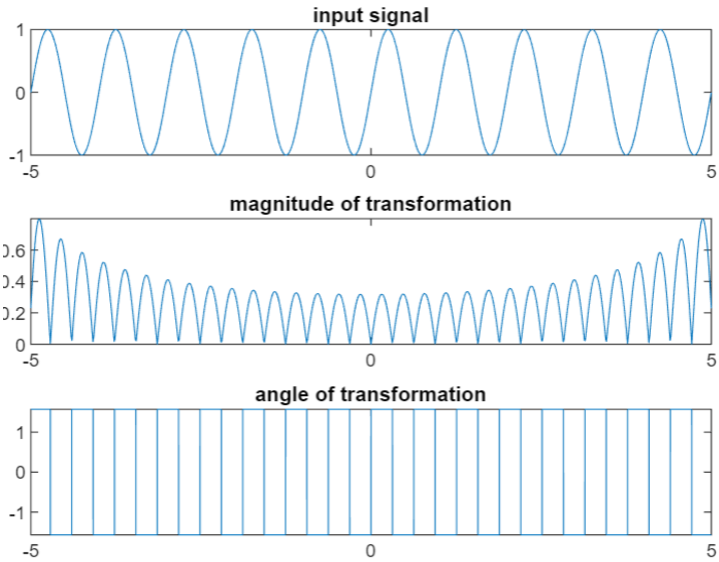
then $\quad Y(j\omega) = X(j\omega)H(j\omega)$

where, $\quad y(t) \xrightarrow{\mathcal{F}} Y(j\omega)$

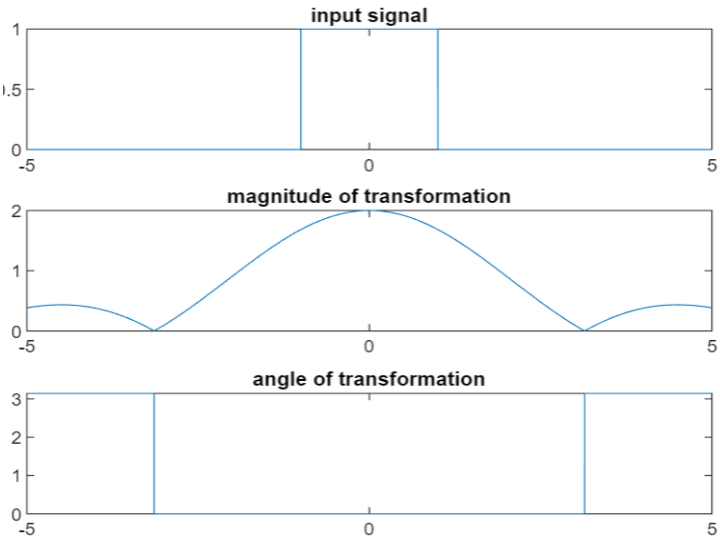
Results:

PART-1:

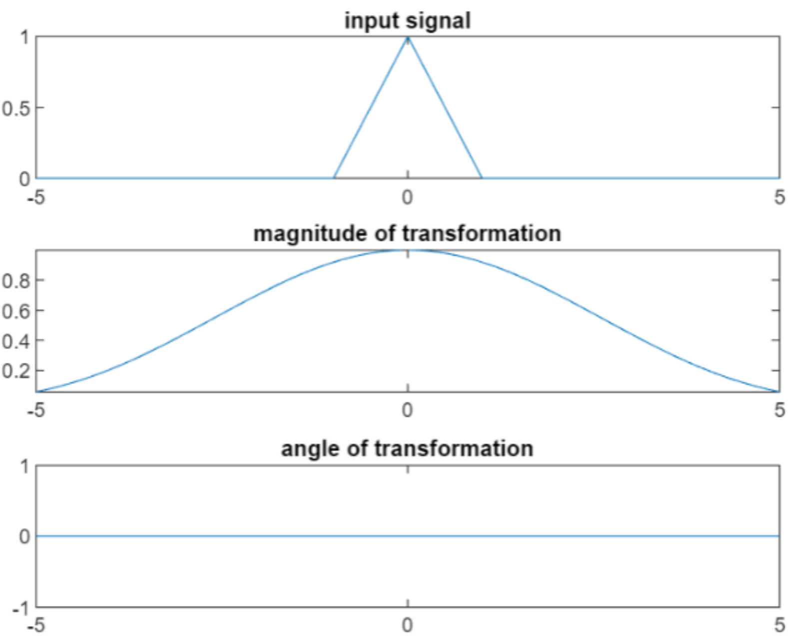
Sine signal:



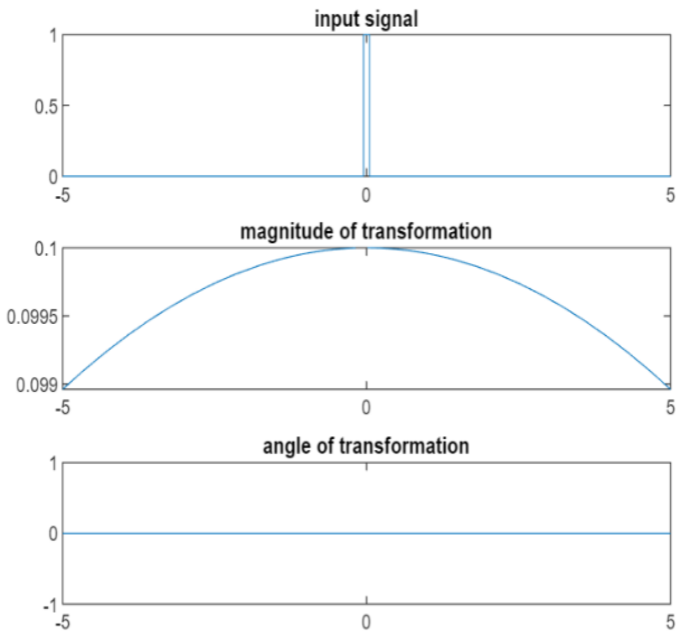
Square wave:



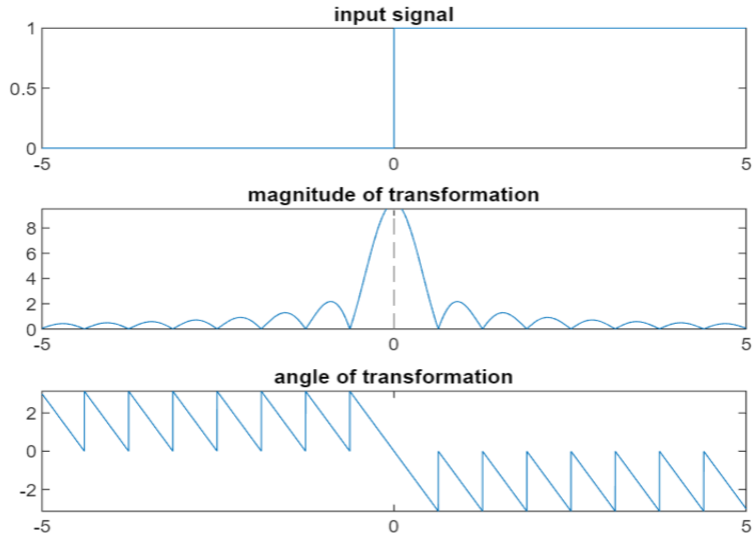
Triangular wave:



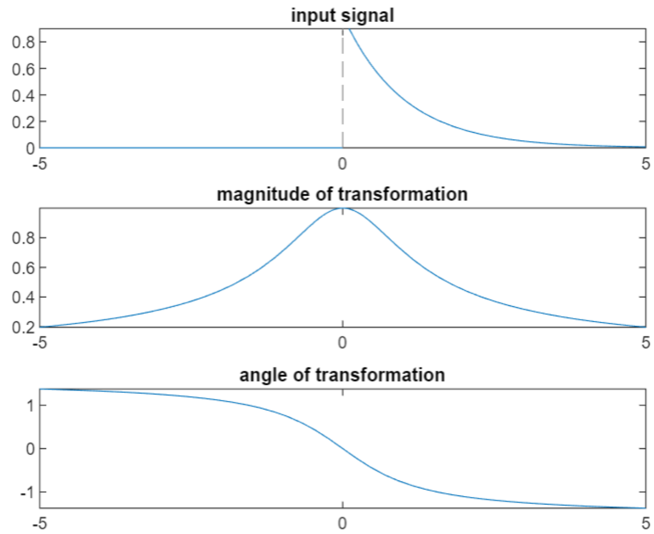
Unit impulse wave:



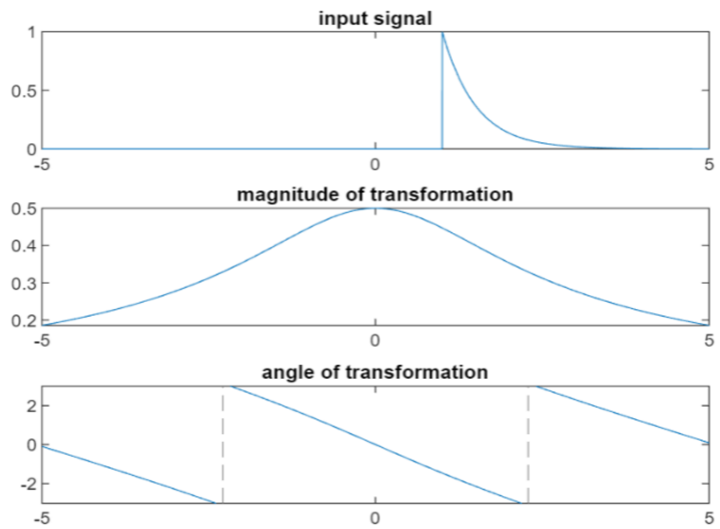
Unit step wave:



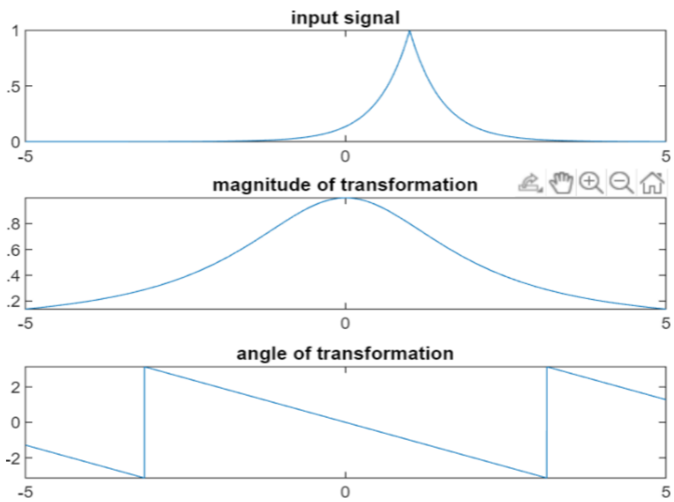
Decaying signal:



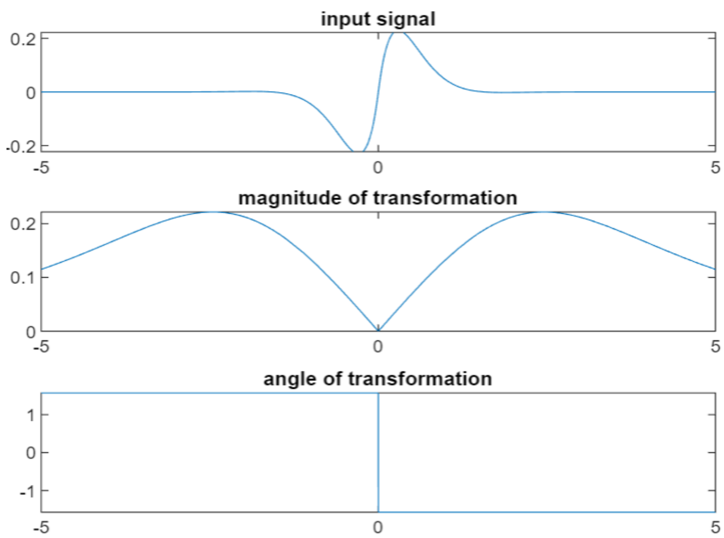
e^-2(t−1) *u(t− 1):



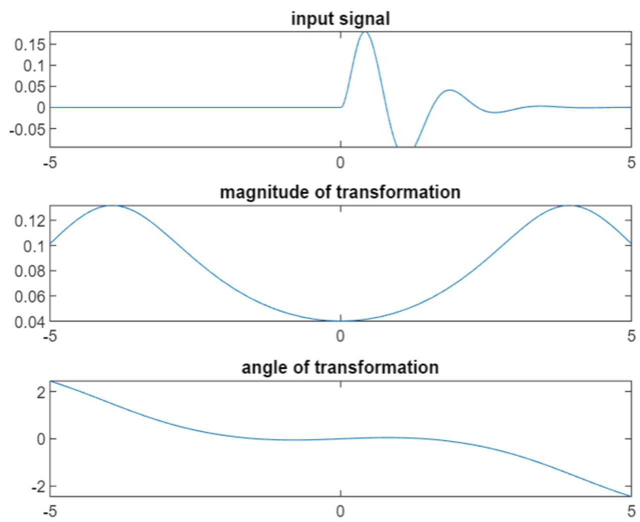
e^ −2|t−1|



e ^−3|t| sin(2t)



[te^ −2t sin(4t)]*u(t)



CODE For Above Graphs:

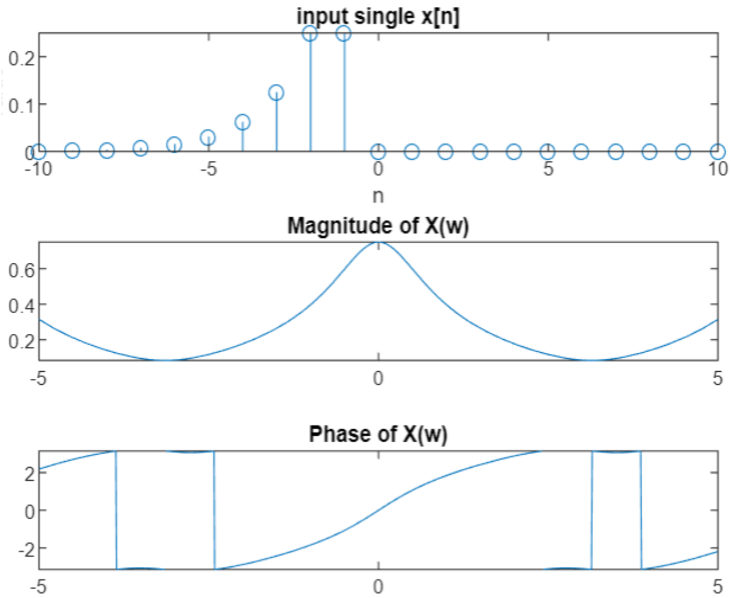
```
clc;
syms w t; % defining time and angular frequency symbolic variables
x(t)= piecewise (t<0,0, t>=0,exp(-t));
% formula of fourier transformation
X(w)=int(x(t)*exp(-1i*w*t),t,-100,100);
subplot(311)
fplot(x(t)),title("input signal");
subplot(312)
fplot(abs(X(w))),title("magnitude of transformation");
subplot(313)
fplot(angle(X(w))),title("angle of transformation");
```

For different signals we can replace x(t) to get respective transformation: -

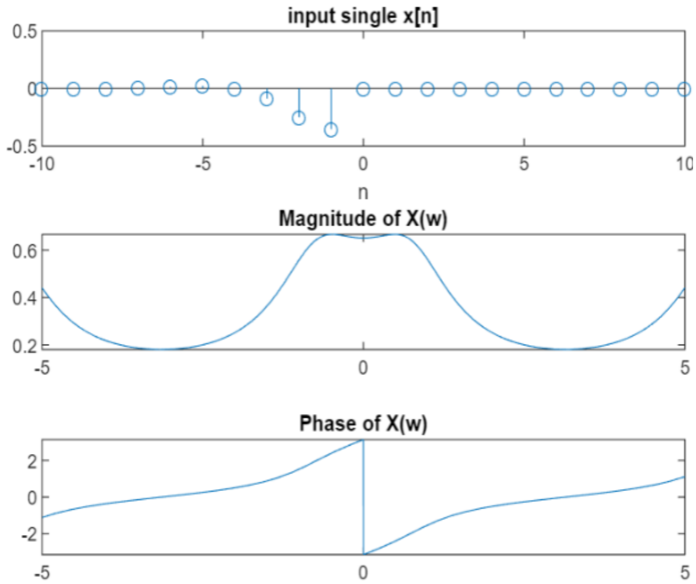
- **Sine wave**: x(t) = sint(2*pi*t)
- **Square wave**: x(t) = rectangularpulse(-1,1,t)
- **Triangular wave**: x(t) = triangularpulse(-1,1,t)
- **Unit impulse wave**: x(t) = piecewise(t<-0.05,0,t>=0.05&t<=0.05,1,t>0.05,0) (Here width given for pulse is 0.1)
- **Unit step wave**: x(t) = piecewise (t<0,0, t>=0,1)
- **$[te^{-2t}\sin(4t)]u(t)$:** x(t) = t*exp(-2*t)*sin(4*t)*heaviside(t)
- **e^−2(t−1)*u (t−1):** x(t) = exp(-2*(t-1))*heaviside(t-1)
- **e ^−2|t−1|** : x(t) = piecewise(t<0, exp(-2*(1-t)),t>=0,exp(-2*(t-1)))
- **$e^{-3|t|}\sin(2t)$** : x(t) = piecewise(t<0,exp(-3*(-t)),t>=0,exp(-3*(t)))*sin(2*t)

PART-2:

$(0.5) \wedge -n * u[-n-1]$

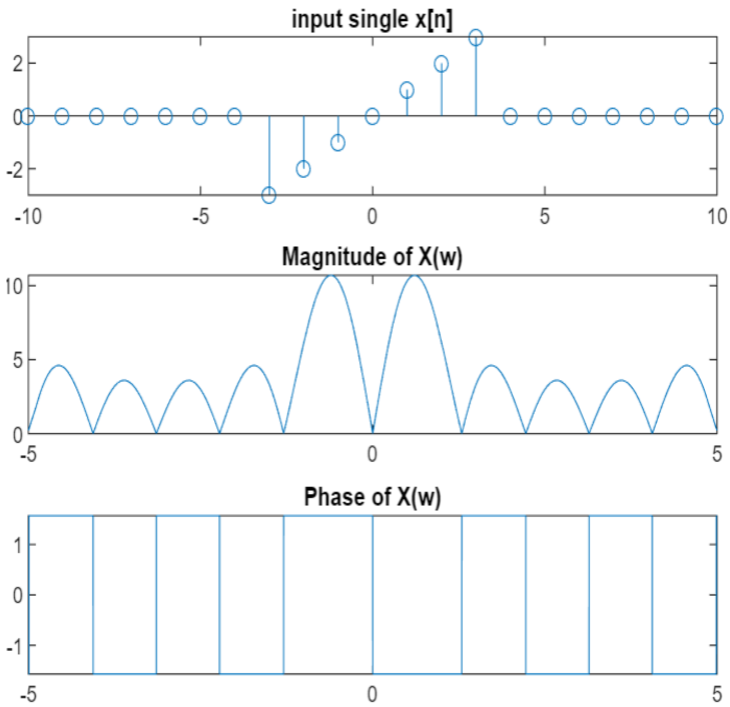


$2 \wedge n * \sin(\pi n/4) * u[-n]$

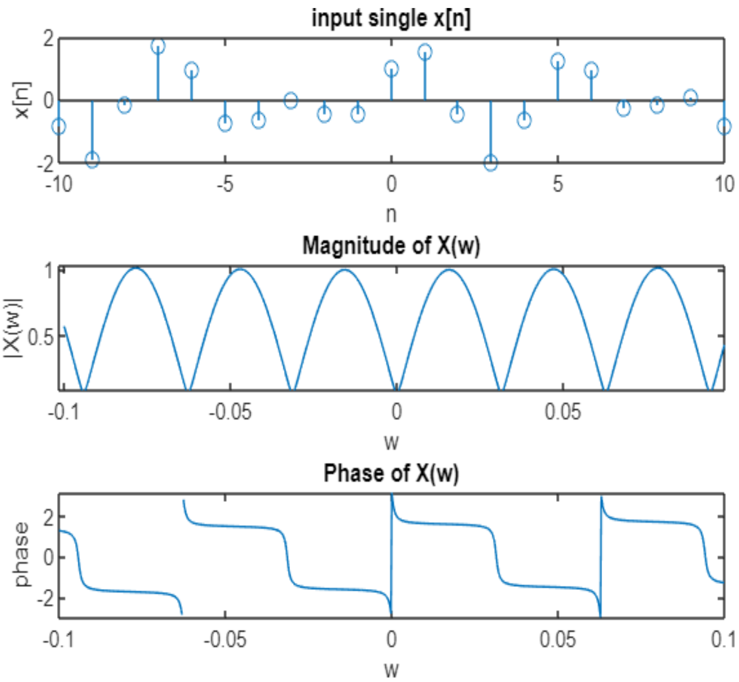


$x[n] = \{n,\ -3 \le n \le 3;$

$0,\ otherwise\}$



$\sin(\pi n/2) + \cos(n)$



CODE For Above Graphs:

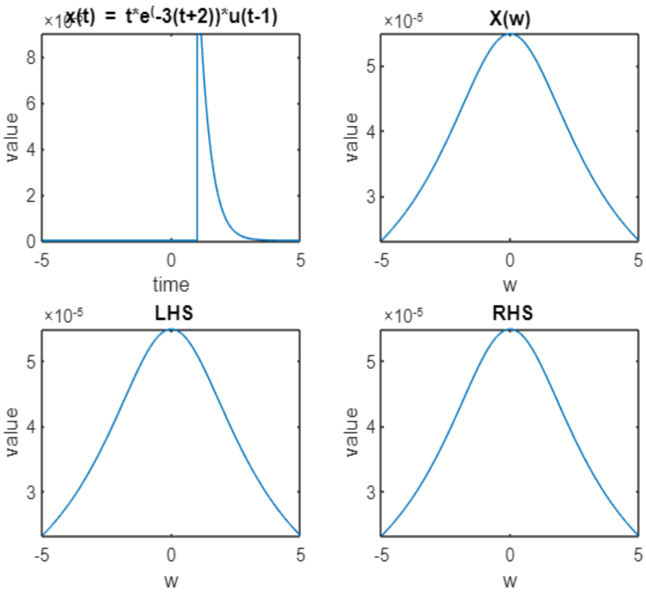
```
clc;
syms n w x(n) X(w);
 x(n)= (2^n).*sin(pi*n*0.25)*heaviside(-n);
m = -10:1:10;
subplot(3,1,1)
stem(m,x(m)),title('input single x[n]');
z = -100:1:100;
X(w)=sum(x(z).*exp(-1i*w*z));
subplot(3,1,2)
fplot(abs(X(w))),title('Magnitude of X(w)');
subplot(3,1,3)
fplot(angle(X(w))),title('Phase of X(w)');
```

For different signals we can replace x(t) to get respective transformation: -

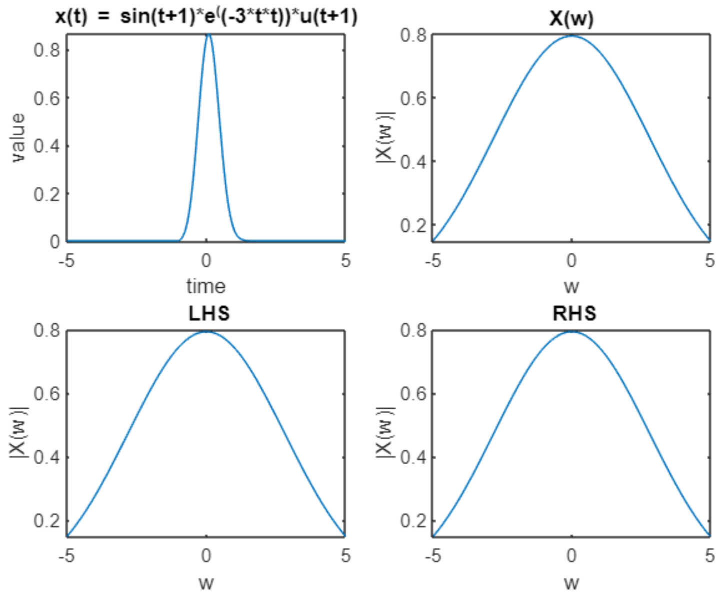
- **(0.5)^−n * u[−n − 1]** : x(n) = (0.5)^ (−n)*heaviside(-n-1)
- **sin (πn/2) + cos(n)** : x(n)=sin(pi*n*0.5)+cos(n)
- **$x[n]$ = { n, −3 ≤ n ≤ 3; 0, *otherwise*** : x(n) = piecewise(n<-3,0,n>=-3&n<=3,n,n>3,0)

PART-3:

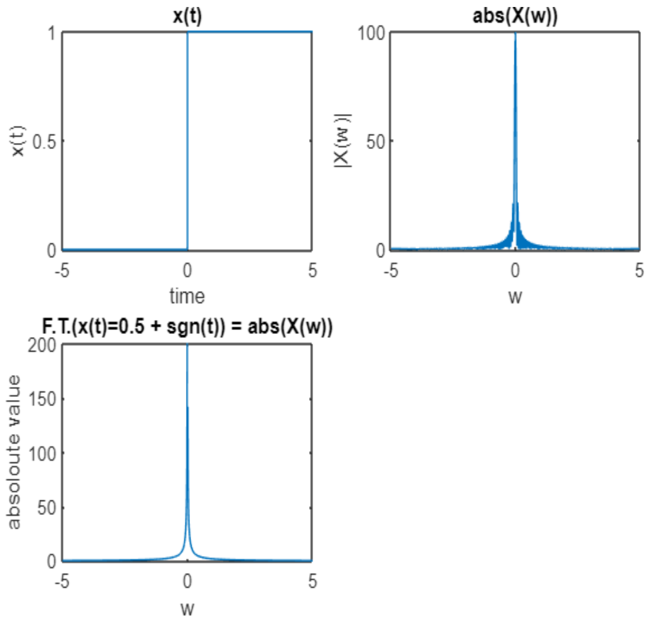
(a)


x(t) = t*e^(-3(t+2))*u(t-1)
X(w)
LHS
RHS
value
time
w
×10^-5

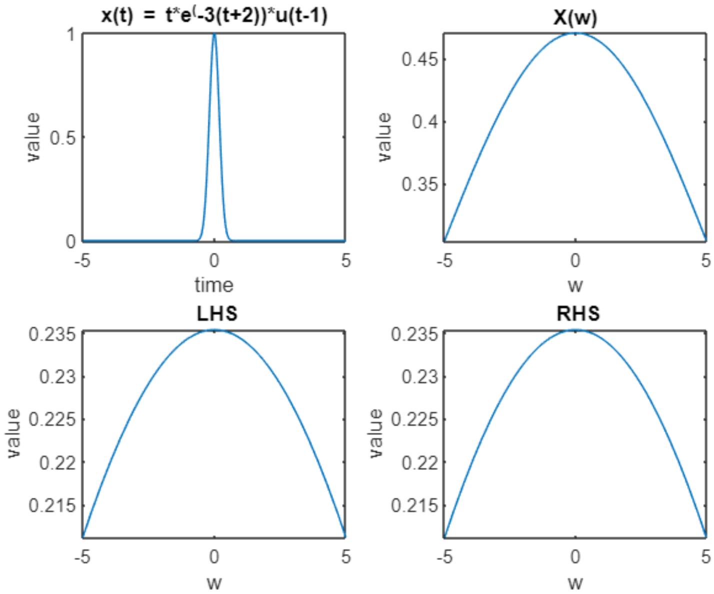
(b).


x(t) = sin(t+1)*e^((-3*t*t))*u(t+1)
X(w)
LHS
RHS
value
|X(w)|
time
w

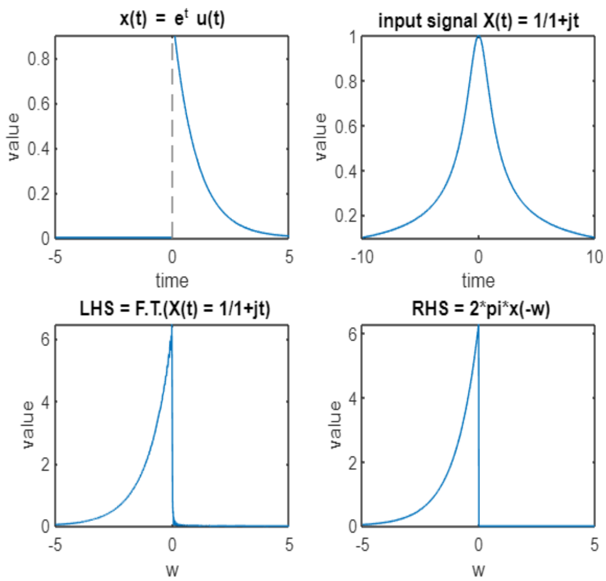
(b)


x(t)
abs(X(w))
F.T.(x(t)=0.5 + sgn(t)) = abs(X(w))
x(t)
|X(w)|
absoloute value
time
w

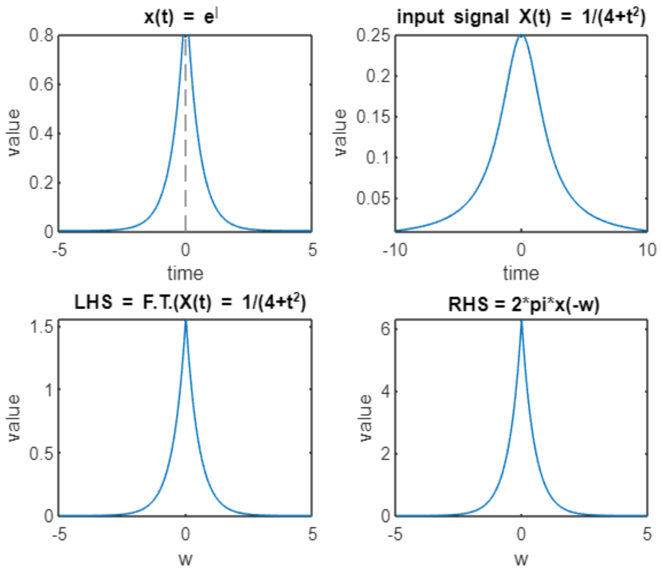
(d).


x(t) = t*e^(-3(t+2))*u(t-1)
X(w)
LHS
RHS
value
time
w

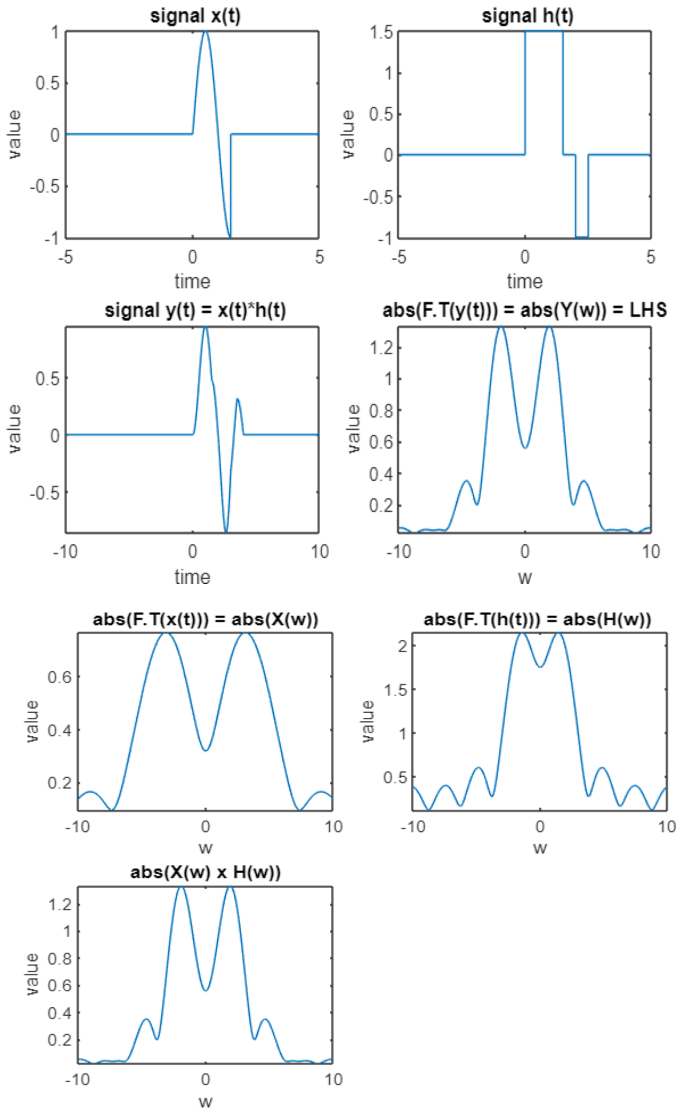
(e).


x(t) = e^t u(t)
input signal X(t) = 1/1+jt
LHS = F.T.(X(t) = 1/1+jt)
RHS = 2*pi*x(-w)
value
time
w

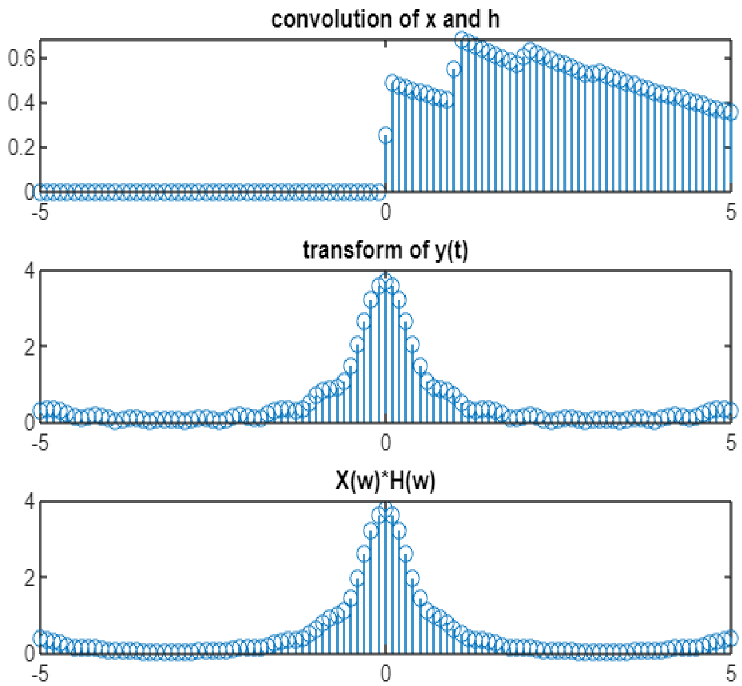
(f).


x(t) = e^|
input signal X(t) = 1/(4+t^2)
LHS = F.T.(X(t) = 1/(4+t^2)
RHS = 2*pi*x(-w)
value
time
w

(g).


signal x(t)
signal h(t)
signal y(t) = x(t)*h(t)
abs(F.T(y(t))) = abs(Y(w)) = LHS
abs(F.T(x(t))) = abs(X(w))
abs(F.T(h(t))) = abs(H(w))
abs(X(w) x H(w))
value
time
w

(h).


convolution of x and h
transform of y(t)
X(w)*H(w)

CODE For Above Graphs:

(a).

```
clc
clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x1 = @(t) t*exp(-3*(t+2))*u(t-1);
X1 = @(w) int(x1(t)*exp(-1i*w*t),t,-100,100);
figure()
subplot(2,2,1)
fplot(x1(t),[-5 5]);
title('x(t) = t*e^(-3(t+2))*u(t-1)');
xlabel('time');
ylabel('value');
subplot(2,2,2)
fplot(w,abs(X1(w)),[-5 5]);
xlabel('w');
ylabel('value');
title('X(w)');
%Time shifting poroperty
% F(x(t-to)) = e^(-jwto)X(w);
%First finding LHS
syms to
to=2;
x2(t) = x1(t-to);
X2 = @(w) int(x2(t)*exp(-1i*w*t),t,-100,100);
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5,5]);
xlabel('w');
ylabel('value');
title(' LHS');
%Finding RHS
X3= @(w) exp(-1i*w*to)*int(x1(t)*exp(-1i*w*t),t,-100,100);
subplot(2,2,4)
fplot(w,abs(X3(w)),[-5,5]);
xlabel('w');
ylabel('value');
title(' RHS');
```

(b).

```
clc

clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x1 = @(t) sin(t+1)*exp(-3*t*t)*u(t+1);
X1 = @(w) int(x1(t)*exp(-1i*w*t),t,-100,100);


figure() subplot(2,2,1)
fplot(x1(t),[-5 5]);
title('x(t) = sin(t+1)*e^((-3*t*t))*u(t+1)');
xlabel('time');
ylabel('value');


subplot(2,2,2)
fplot(w,abs(X1(w)),[-5 5]);
xlabel('w');
ylabel('|X(w)|');
title('X(w)');
```

```
%Time shifting poroperty
% F(x(t-to)) = e^(-jwto)X(w);
%First finding LHS
syms to
to=2;
x2(t) = x1(t-to);
X2 = @(w) int(x2(t)*exp(-1i*w*t),t,-100,100);
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5,5]);
xlabel('w');
ylabel('|X(w)|');
title(' LHS');

%Finding RHS

X3= @(w) exp(-1i*w*to)*int(x1(t)*exp(-1i*w*t),t,-100,100);
subplot(2,2,4)
fplot(w,abs(X3(w)),[-5,5]);
xlabel('w');
ylabel('|X(w)|');
title(' RHS');
```

(c).

```
clc
clear all
close all
syms t T w;
% u(t) = 0.5 + sgn(t)
x = @(t) piecewise(t<0,0,t>=0,1);
X = @(w) int(x(t)*exp(-1i*w*t),t,-100,100);
%FT of x(t)
deltaFunction = @(t) piecewise(t<0,0,t==0,50,t>0,0);
%F.T.(0.5) = pi*delta(w) , %F.T. sgn(t) = 2/jw;;
X2 = @(w) 2/(1i*w) + pi*deltaFunction(w);
%F.T. computed from sgn(t);
figure()
subplot(2,2,1)
fplot(x(t),[-5 5]);
title('x(t)');
xlabel('time');
ylabel('x(t)');
subplot(2,2,2)
fplot(w,abs(X(w)),[-5 5]);
title('abs(X(w))');
xlabel('w');
ylabel('|X(w)|');
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5 5]);
title('F.T.(x(t)=0.5 + sgn(t)) = abs(X(w))');
xlabel('w');
ylabel('absoloute value');
```

(d).

```
clc
clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x1 = @(t) cos(2*t)*exp(-(12*t^2))*u(2*(t+1));
X1 = @(w) int(x1(t)*exp(-1i*w*t),t,-100,100);
figure()
subplot(2,2,1)
fplot(x1(t),[-5 5]);
title('x(t) = t*e^(-3(t+2))*u(t-1)');
xlabel('time');
ylabel('value');
subplot(2,2,2)
fplot(w,abs(X1(w)),[-5 5]);
xlabel('w');
ylabel('value');
title('X(w)');
%Time scaing poroperty
% F(x(a*t)) = 1/|a| X(jw/a);
%First finding LHS
syms a
a=2;
x2(t) = subs(x1(t),t,a*t);
X2 = @(w) int(x2(t)*exp(-1i*w*t),t,-100,100);
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5,5]);
xlabel('w');
ylabel('value');
title(' LHS');
%Finding RHS
X3= @(w) (1/a)*subs(X1(w),w,w/a);
subplot(2,2,4) fplot(w,abs(X3(w)),[-
5,5]); xlabel('w');
ylabel('value');
title(' RHS');
```

(e).

```
clc
clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x1 = @(t) exp(-t)*u(t);
x2 = @(t) 1/(1 + 1i*t);
X2 = @(w) int(x2(t)*exp(-1i*w*t),t,-100,100);
figure()
subplot(2,2,1)
fplot(x1(t),[-5 5]);
title('x(t) = e^-^t u(t)');
xlabel('time');
ylabel('value');
subplot(2,2,2)
fplot(t,abs(x2(t)),[-10 10]);
xlabel('time');
ylabel('value');
title('input signal X(t) = 1/1+jt');
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5 5]);
xlabel('w');
ylabel('value');
title(' LHS = F.T.(X(t) = 1/1+jt)');
subplot(2,2,4)
fplot(w,abs(2*pi*x1(-w)));
xlabel('w');
ylabel('value');
title(' RHS = 2*pi*x(-w)');
```

(f).

```
clc
clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x1 = @(t) exp(-2*abs(t));
x2 = @(t) 1/(4 + t^2);
X2 = @(w) int(x2(t)*exp(-1i*w*t),t,-100,100);
figure()
subplot(2,2,1)
fplot(x1(t),[-5 5]);
title('x(t) = e^-^|^t^| ');
xlabel('time');
ylabel('value');
subplot(2,2,2)
fplot(t,abs(x2(t)),[-10 10]);
xlabel('time');
ylabel('value');
title('input signal X(t) = 1/(4+t^2)');
subplot(2,2,3)
fplot(w,abs(X2(w)),[-5 5]);
xlabel('w');
ylabel('value');
title(' LHS = F.T.(X(t) = 1/(4+t^2)');
subplot(2,2,4)
fplot(w,abs(2*pi*x1(-w)));
xlabel('w');
ylabel('value');
title(' RHS = 2*pi*x(-w)');
```

(g).

```
clc
clear all
close all
syms t T w;
u = @(t) piecewise(t<0,0,t>=0,1);
x = @(t) sin(pi*t)*(u(t)-u(t - 1.5));
h =@(t) 1.5*(u(t)- u(t- 1.5))- u(t - 2) + u(t - 2.5);
y = @(t)int(x(T)*h(t-T),T,-10,10);
Y = @(w) int(y(t)*exp(-1i*w*t),t,-10,10);
X = @(w) int(x(t)*exp(-1i*w*t),t,-10,10);
H = @(w) int(h(t)*exp(-1i*w*t),t,-10,10);
figure()
subplot(2,2,1)
fplot(x(t),[-5 5]);
title(' signal x(t)');
xlabel('time');
ylabel('value');
subplot(2,2,2)
fplot(h(t),[-5 5]);
title(' signal h(t)');
xlabel('time');
ylabel('value');
subplot(2,2,3)
fplot(y(t),[-10 10]);
title(' signal y(t) = x(t)*h(t)');
xlabel('time');
ylabel('value');
subplot(2,2,4)
fplot(w,abs(Y(w)),[-10 10]);
title('abs(F.T(y(t))) = abs(Y(w)) = LHS');
xlabel('w');
ylabel('value');
figure()
subplot(2,2,1)
```

```
fplot(w,abs(X(w)),[-10 10]);
title('abs(F.T(x(t))) = abs(X(w))');
xlabel('w');
ylabel('value');
subplot(2,2,2)
fplot(w,abs(H(w)),[-10 10]);
title('abs(F.T(h(t))) = abs(H(w)) ');
xlabel('w');
ylabel('value');
subplot(2,2,3)
fplot(w,abs(X(w)*H(w)),[-10 10]);
title('abs(X(w) x H(w))');
xlabel('w');
ylabel('value');
```

(h).

```
clc
clear all
close all
syms t w k;
x(t)=((0.8)^t)*heaviside(t);
h(t)= ((0.3)^t)*heaviside(t);
x1(t)=subs(x(t),t,t-k);
% convolution
y(t)=symsum(x1(t)*h(k),k,-10,10);
% fourier transform
X(w)=symsum(x(t)*exp(-1i*w*t),t,-10,10);
H(w)=symsum(h(t)*exp(-1i*w*t),t,-10,10);
Y(w)=symsum(y(t)*exp(-1i*w*t),t,-10,10);
Y1(w)= X(w)*H(w);
l=-5:.1:5;
subplot(311)
stem(l,y(l)),title("convolution of x and h");
subplot(312)
stem(l,abs(Y(l))),title("transform of y(t)");
subplot(313)
stem(l,abs(Y1(l))),title("X(w)*H(w)");
```

DISSCUSION:

Fourier Transformation has wide range of applications. If an application cannot be done in time domain, then it can be done in frequency domain and vice versa. And properties like convolution is easy in frequency domain than in time domain.

Few applications:

- Signal Processing
- Image processing
- Heat distribution mapping
- Light Implication (interference, Diffraction etc.)

CONCLUSION:

By this experiment, we have successfully written the algorithm for finding of Fourier transform input functions. And we also checked the properties of the transform. Here transformations may not be completely accurate as we are not integrating through -infinity to infinity but taking a certain interval.