INDEX

10	Write a program in Matlab to calculate Co-channel and Adjacent Channel Interferences.	20		
11	Write a program in Matlab to calculate the median path loss for Hata model for outdoor propagation.	22		

Practical: 1

Aim: Write a Program in Matlab to Calculate the free-space path loss (in dB) of a 10 GHz radar signal over a distance of 10 km.

Input:

```
fc = 10.0e9;
lambda = physconst('LightSpeed')/fc;
R = 10e3;
L = fspl(R,lambda)
```

Output:

```
L =

  132.4478
```

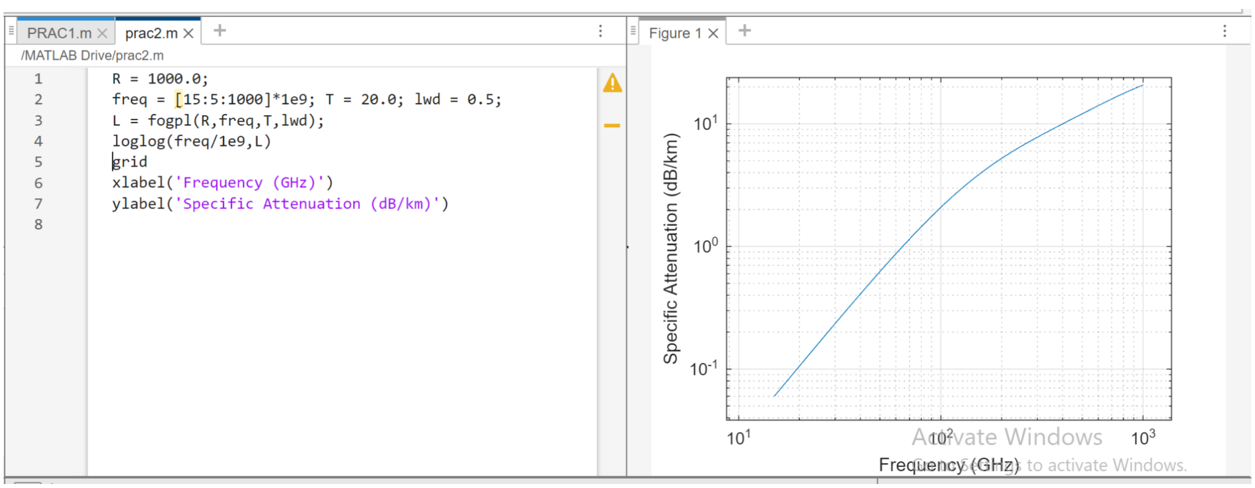
Practical: 2

Aim: Write a Program in Matlab to Compute the attenuation of signals propagating through a cloud that is 1 km long at 1000 meters altitude. Compute the attenuation for frequencies from 15 to 1000 GHz. A typical value for the cloud liquid water density is 0.5 g/m^3. Assume the atmospheric temperature at 1000 meters is 20^{0}C.

Input:

```
R = 1000.0;
freq =[15:5:1000]*1e9;
T = 20.0; lwd = 0.5;
L = fogpl(R,freq,T,lwd);

loglog(freq/1e9,L)
grid
xlabel('Frequency (GHz)')
ylabel('Specific Attenuation (dB/km)')
```

Output:



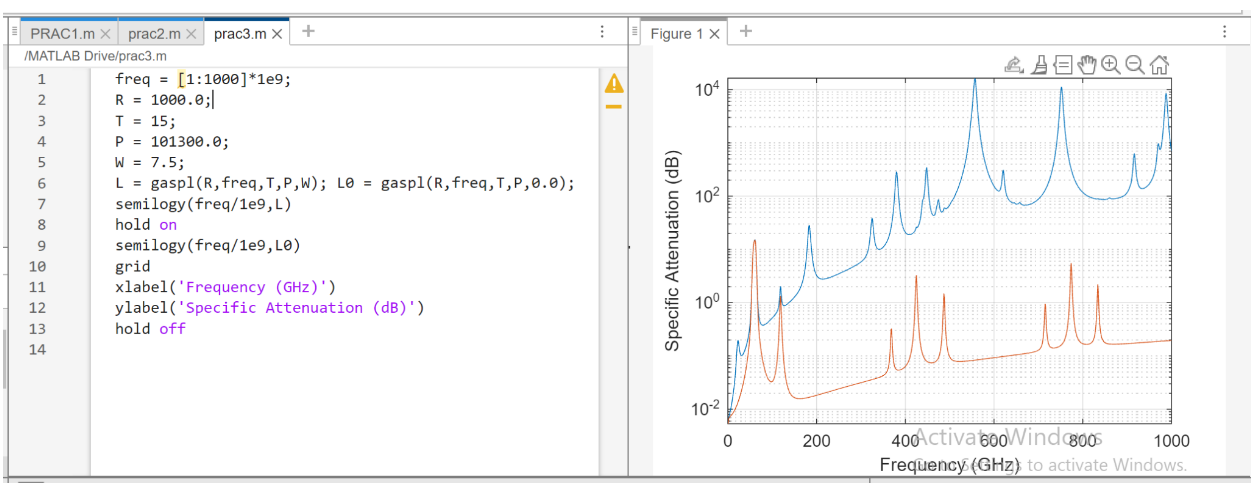
Practical: 3

Aim: Write a Program in Matlab to Compute the attenuation spectrum from 1 to 1000 GHz for an atmospheric pressure of 101.300 kPa and a temperature of 15C. Plot the spectrum for a water vapor density of 7.5 g/m^3 and then plot the spectrum for dry air (zero water vapor density).

Input:

```
freq = [1:1000]*1e9;
R = 1000.0;
T = 15;
P = 101300.0;
W = 7.5;
L =
gaspl(R,freq,T,P,W); L0
= gaspl(R,freq,T,P,0.0);
semilogy(freq/1e9,L)
hold on
semilogy(freq/1e9,L0)
grid
xlabel('Frequency (GHz)')
ylabel('Specific Attenuation (dB)')
hold off
```

Output:



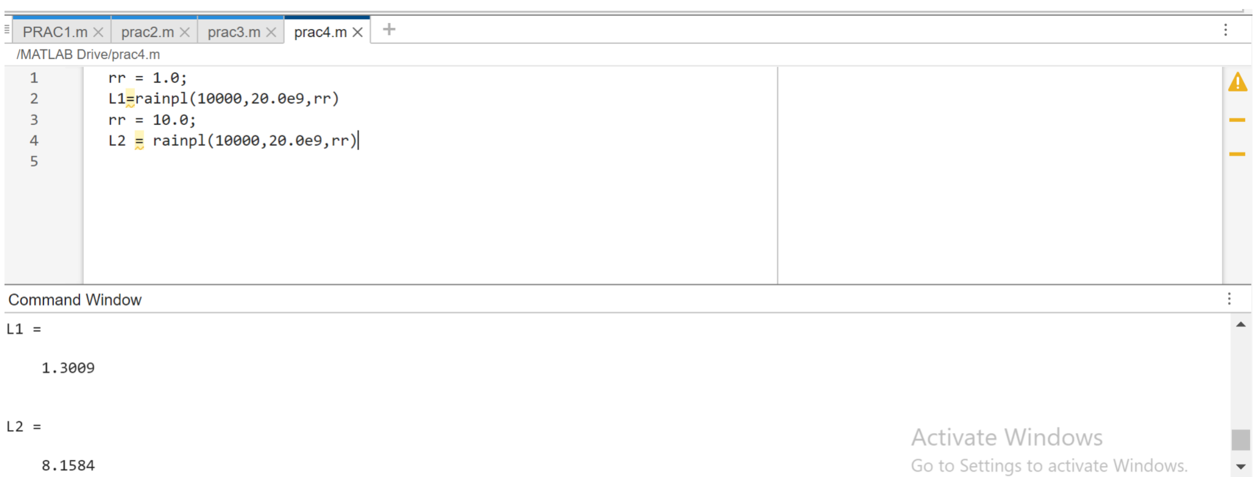
Practical: 4

Aim: Write a program in Matlab to Compute the signal attenuation due to rainfall for a 20 GHz signal over a distance of 10 km in light and heavy rain. Propagate the signal in a light rainfall of 1 mm/hr.

Input:

```
rr = 1.0;
L=rainpl(10000,20.0e9,rr)
rr = 10.0;
L = rainpl(10000,20.0e9,rr)
```

Output:



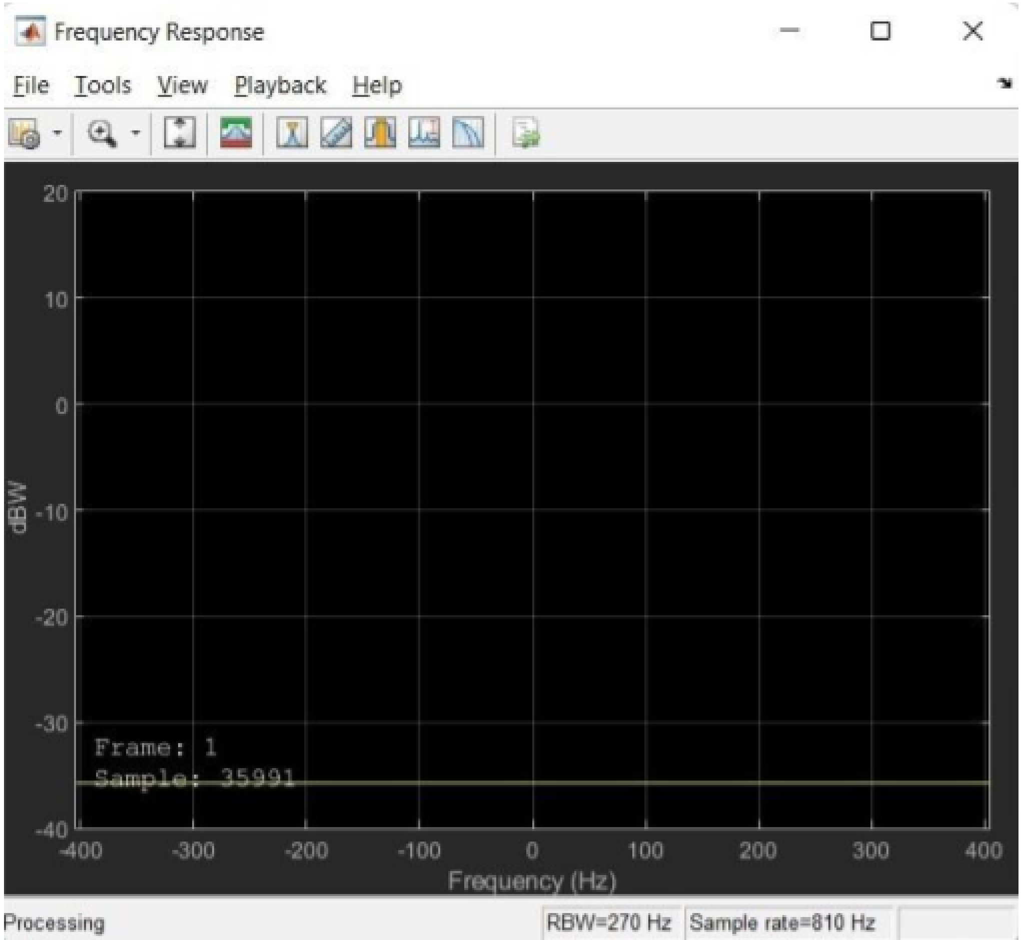
Practical: 5

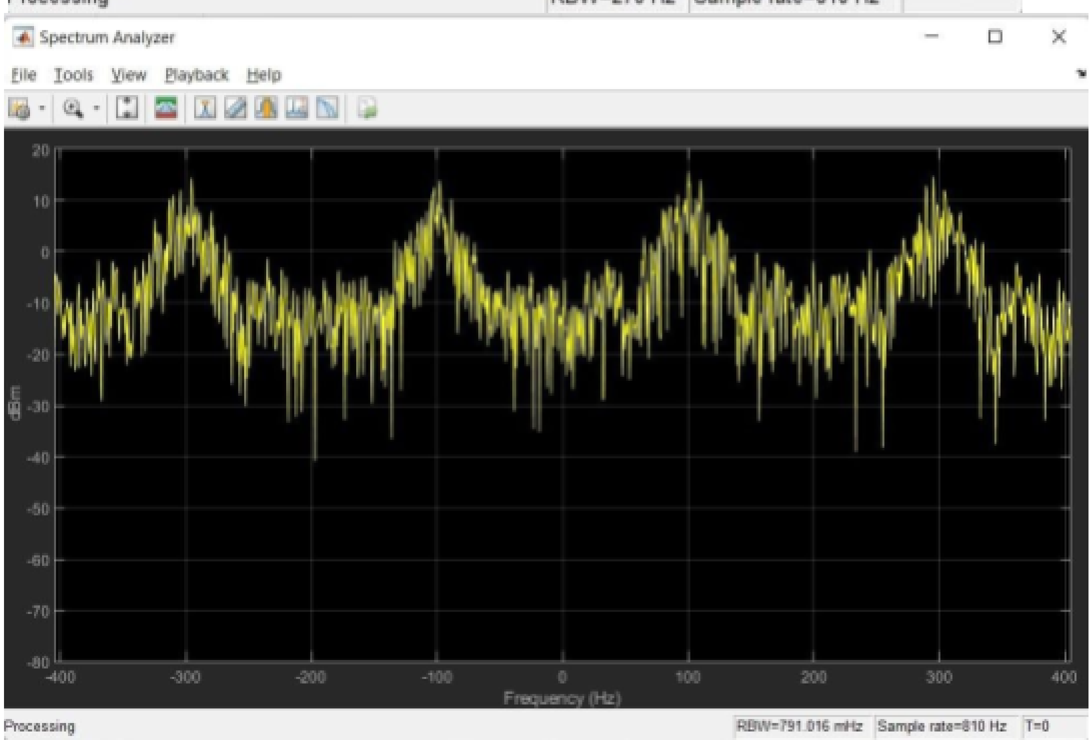
Aim: Write a program in Matlab to Pass an FSK signal through a Rayleigh multipath fading channel. Change the signal bandwidth to observe the impact of the fading channel on the FSK spectrum.

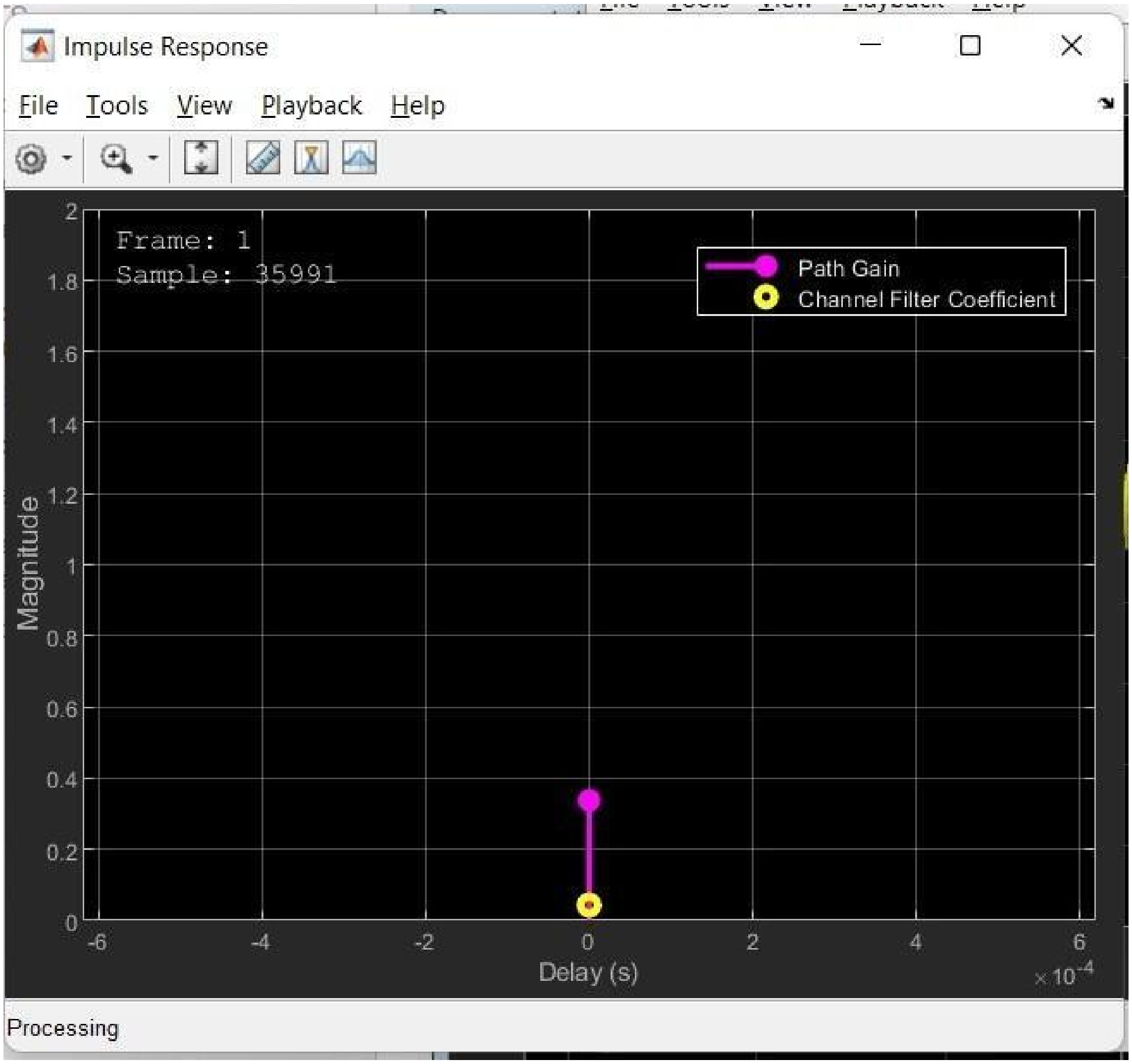
Input:

```
M = 4; % Modulation order

symbolRate = 45; % Symbol rate (bps)
freqSep = 200;  % Frequency separation (Hz)

sampPerSym = ceil(M*freqSep/symbolRate);
fsamp = sampPerSym*symbolRate;

fskMod = comm.FSKModulator(M, ...
'FrequencySeparation',freqSep, ...
'SamplesPerSymbol',sampPerSym, ...
'SymbolRate',symbolRate);

pathDelays = [0 3 10]*1e-6; % Discrete delays of three-path channel (s)
avgPathGains = [0 -3 -6]; % Average path gains (dB)

channel = comm.RayleighChannel(...
'SampleRate',fsamp, ...
'PathDelays',pathDelays, ...
'AveragePathGains',avgPathGains, ...
'MaximumDopplerShift',0.01, ...
'Visualization','Impulse and frequency responses', ...
'SamplesToDisplay','10%');

data = randi([0 3],2000,1);
modSig = fskMod(data);

spectrum = dsp.SpectrumAnalyzer('SampleRate',fsamp);
spectrum(modSig)

snrdB = 25;
rxSig = awgn(channel(modSig),snrdB);
spectrum(rxSig)
```

Output:


Frequency Response
File Tools View Playback Help
20
10
0
-10
-20
-30
-40
dBW
Frame: 1
Sample: 35991
-400 -300 -200 -100 0 100 200 300 400
Frequency (Hz)
Processing
RBW=270 Hz
Sample rate=810 Hz


Spectrum Analyzer
File Tools View Playback Help
20
10
0
-10
-20
-30
-40
-50
-60
-70
-80
dBm
-400 -300 -200 -100 0 100 200 300 400
Frequency (Hz)
Processing
RBW=791.016 mHz
Sample rate=810 Hz
T=0


Impulse Response
File Tools View Playback Help
Frame: 1
Sample: 35991
Path Gain
Channel Filter Coefficient
Magnitude
Delay (s)
× 10^-4
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-6
-4
-2
0
2
4
6
Processing

Practical: 6

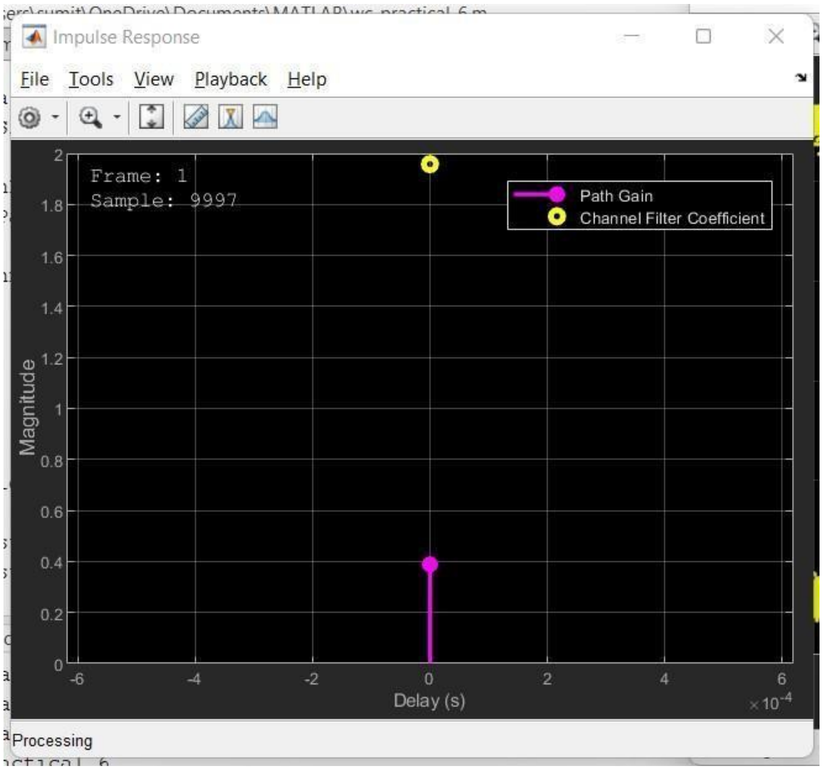
Aim: Write a program in Matlab to Pass a QPSK signal through a Rayleigh multipath fading channel. Change the signal bandwidth to observe the impact of the fading channel on the QPSK constellation.

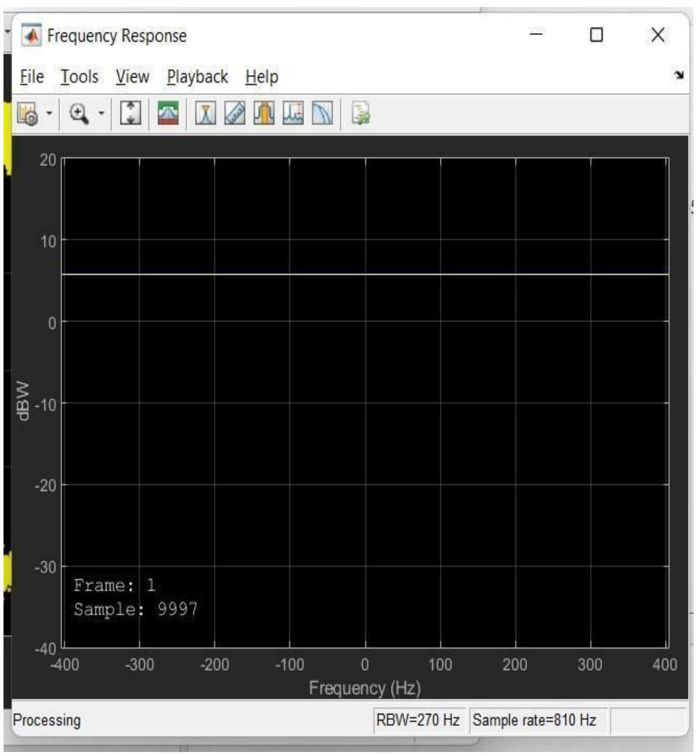
Input:

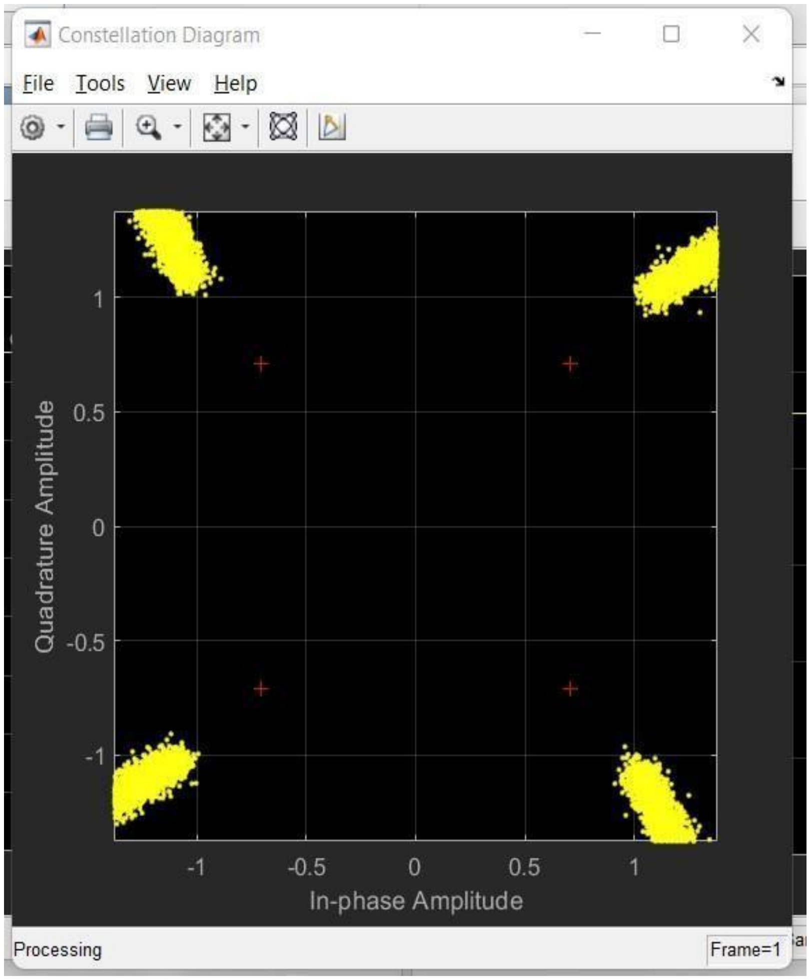
QPSK Modulation in Fading Channel

```
symbolRate = 500;

data = randi([0 3],10000,1);
modSig = pskmod(data,4,pi/4,'gray');

pathDelays = [0 3 10]*1e-6;        % Discrete delays of three-path channel (s)
avgPathGains = [0 -3 -6]; % Average path gains (dB)fsamp = symbolRate;

channel = comm.RayleighChannel(...
'SampleRate',fsamp, ...
'PathDelays',pathDelays, ...
'AveragePathGains',avgPathGains, ...
'MaximumDopplerShift',0.01, ...
'Visualization','Impulse and frequency responses');

rxSig = awgn(channel(modSig),25);

constDiagram = comm.ConstellationDiagram;
constDiagram(rxSig)
```

Output:

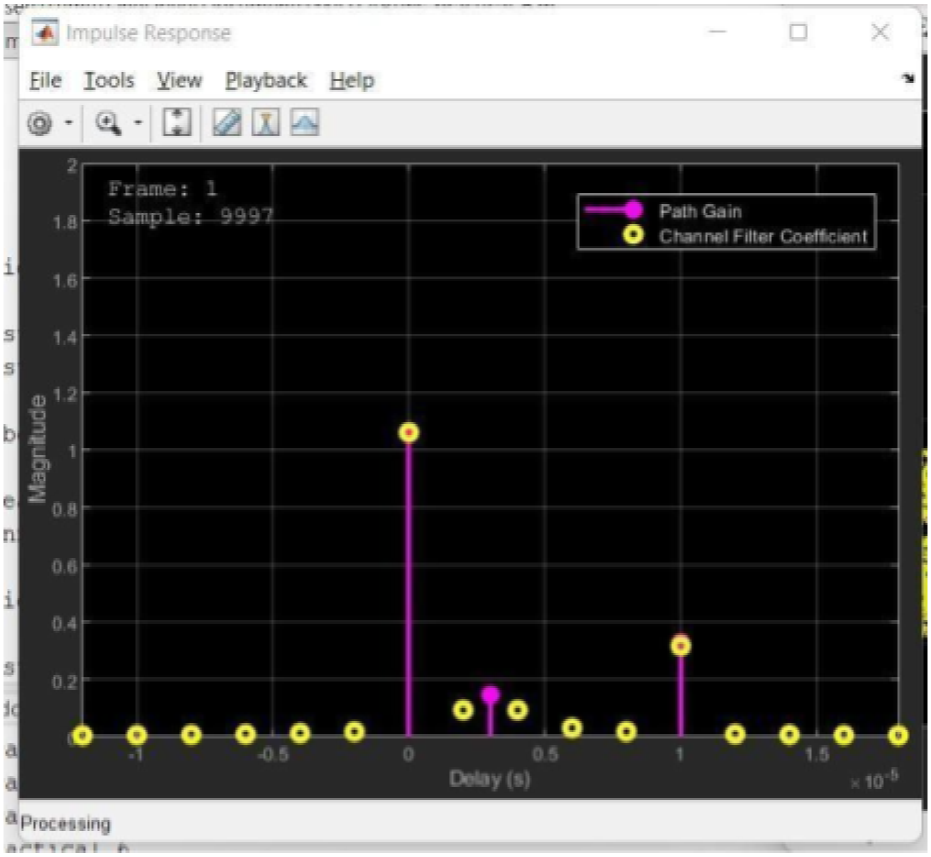

Impulse Response
File Tools View Playback Help
Frame: 1
Sample: 9997
Path Gain
Channel Filter Coefficient
Magnitude
Delay (s)
× 10^-4
Processing

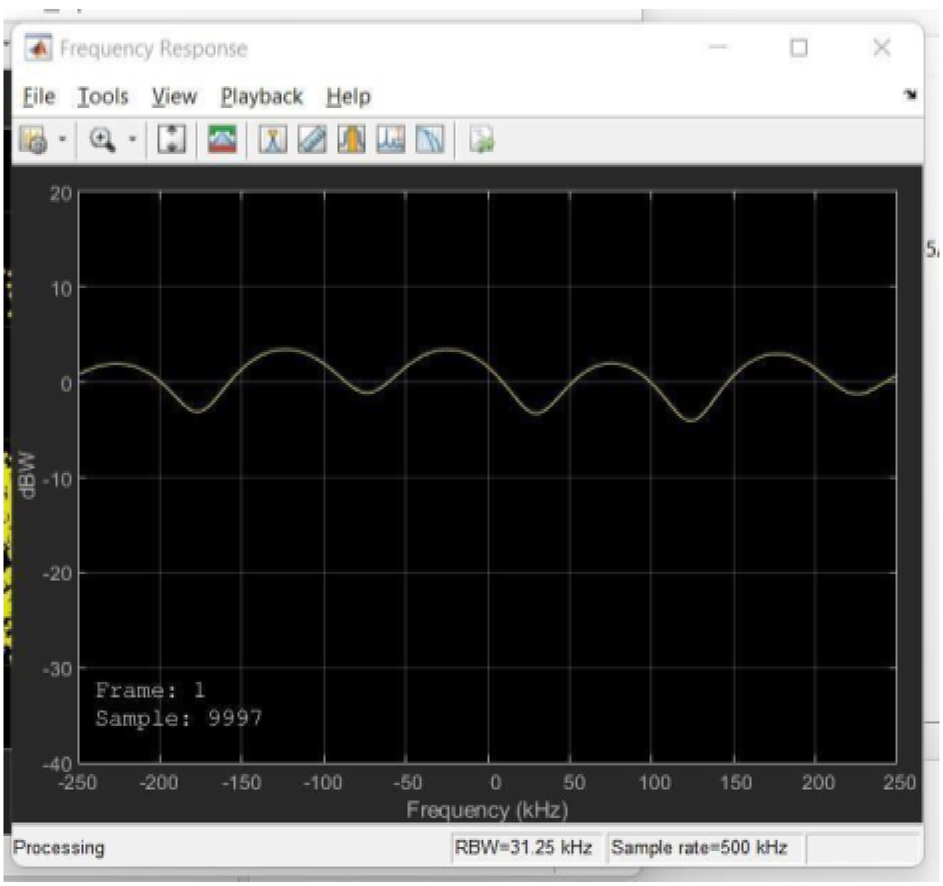

Frequency Response
File Tools View Playback Help
dBW
Frame: 1
Sample: 9997
Frequency (Hz)
Processing
RBW=270 Hz
Sample rate=810 Hz

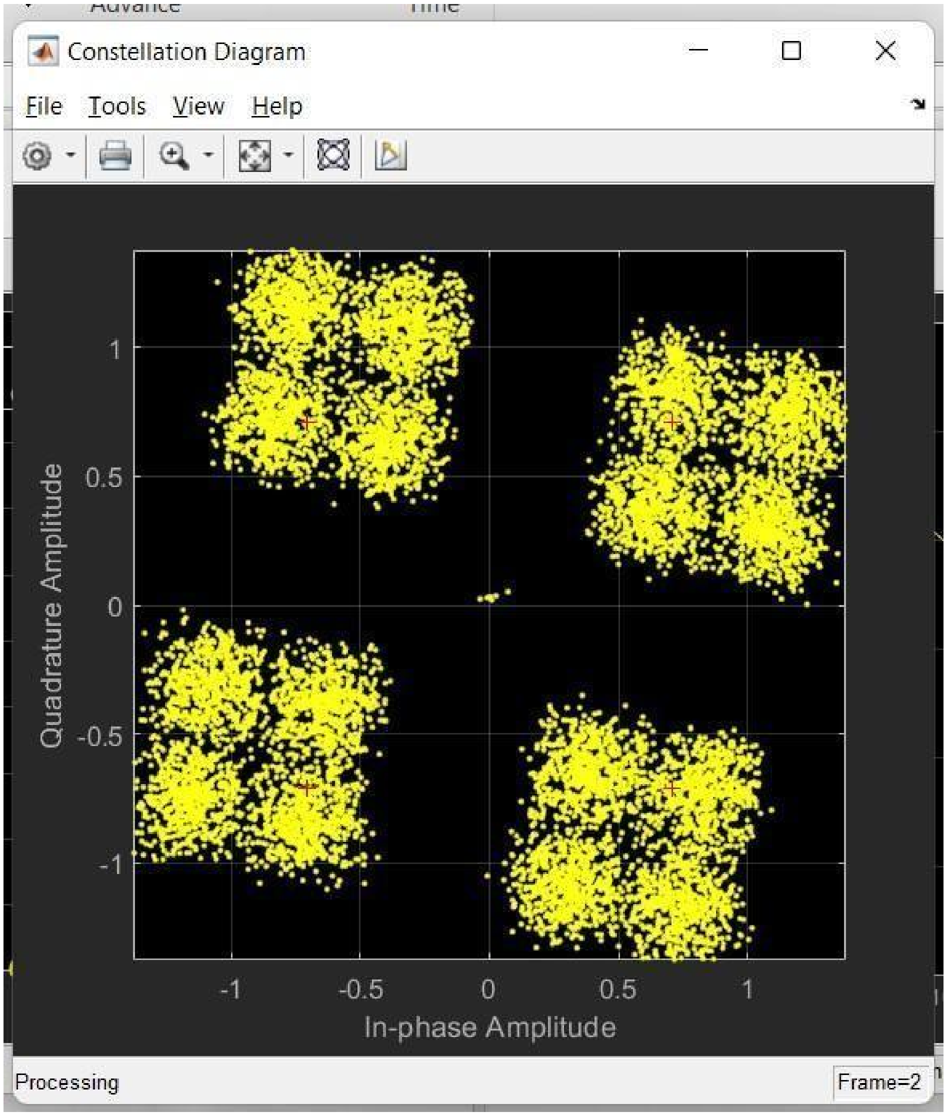


QPSK Modulation in Frequency-Selective Fading:

```
symbolRate = 500e3;
release(channel) channel.SampleRate
= symbolRate; rxSig =
awgn(channel(modSig),25);
constDiagram(rxSig)
```


Impulse Response
File Tools View Playback Help
Frame: 1
Sample: 9997
Path Gain
Channel Filter Coefficient
Magnitude
Delay (s)
$\times 10^{-5}$
Processing


Frequency Response
File Tools View Playback Help
dBW
Frame: 1
Sample: 9997
Frequency (kHz)
Processing
RBW=31.25 kHz
Sample rate=500 kHz


Constellation Diagram
File Tools View Help
1
0.5
0
-0.5
-1
Quadrature Amplitude
-1
-0.5
0
0.5
1
In-phase Amplitude
Processing
Frame=2

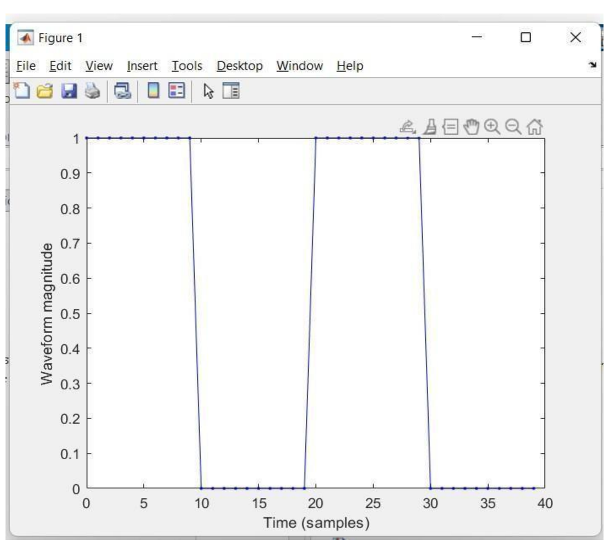
Practical: 7

Aim: Write a program in in Matlab to Create a nonpolarized electromagnetic field consisting of two rectangular waveform pulses at a carrier frequency of 100 MHz. Assume the pulse width is 10 ms and the sampling rate is 1 MHz. The bandwidth of the pulse is 0.1 MHz. Assume a 50% duty cycle in so that the pulse width is one-half the pulse repetition interval. Create a two-pulse wave train. Set the GroundReflectionCoefficient to 0.9 to model strong ground reflectivity. Propagate the field from a stationary source to a stationary receiver. The vertical separation of the source and receiver is approximately 10 km.

Input:

```
c = physconst('LightSpeed');
fs = 1e6;
pw = 10e-6;
pri = 2*pw;
PRF = 1/pri;
fc = 100e6;
lambda = c/fc;
waveform =
phased.RectangularWaveform('SampleRate',fs,'PulseWidth',pw,...'PRF',PRF,'Output
Format','Pulses','NumPulses',2;
wav = waveform();
n = size(wav,1);
figure;
plot((0:(n-1)),real(wav),'b.-');
xlabel('Time (samples)') ylabel('Waveform magnitude')
```

Output:



Practical: 8

Aim: Write a program in Matlab to approximate Calculation of Diffraction Loss.

Input:

```
mue=-5:5 % value of mue
inde=0
for vmuer=-5:5
inde=inde+1
intFe=quad('exp((-j*pi*x.^2)/2)',vmuer,20) %Integration of the function used in integral part of
Complex Fresnel Integral fe=abs((0.5+0.5*j)*intFe ) %Complex Fresnel Integral.
Gdb_e(inde)=20*log10(fe) %attenuation factor introduced by Knife-edge diffraction
End
```

Output:

```
mue =
  -5   -4   -3   -2   -1    0    1    2    3    4    5
inde =
   0
inde =
   1
intFe =
  1.0636 - 0.9833i
fe =
   1.0242
Gdb_e =
  0.2080
inde =
   2
intFe =
  0.9984 - 0.9046i
fe =
  0.9527
Gdb_e =
  0.2080   -0.4212
inde =
   3
```

```
intFe =  1.1057 - 0.9804i
fe =
   1.0449
Gdb_e =
   0.2080   -0.4212    0.3818
inde =
    4
intFe =
   0.9882 - 0.8275i

Gdb_e =

  Columns 1 through 9

 0.2080   -0.4212    0.3818   -0.8056    0.9496   -6.1589  -13.9552  -20.0162  -22.4683

  Columns 10 through 11

-26.9428  -26.6971
```

Practical: 9

Aim: Write a program in Matlab for processing a signal using a multipath fading channel.

Input:

```
sampleRate500kHz = 500e3; % Sample rate of 500K Hz
sampleRate20kHz  = 20e3; % Sample rate of 20K Hz
maxDopplerShift  = 200; % Maximum Doppler shift of diffuse components (Hz)
delayVector = (0:5:15)*1e-6; % Discrete delays of four-path channel (s)

gainVector  = [0 -3 -6 -9]; % Average path gains (dB)

KFactor = 10; % Linear ratio of specular power to diffuse power
specDopplerShift = 100; % Doppler shift of specular component (Hz)

% Configure a Rayleigh channel object
rayChan = comm.RayleighChannel( ...
        'SampleRate',sampleRate500kHz, ... 'PathDelays',delayVector, ...
        'AveragePathGains',gainVector, ...
        'MaximumDopplerShift',maxDopplerShift, ...
        'RandomStream','mt19937ar with seed', ...
        'Seed',10, ...
        'PathGainsOutputPort',true);

% Configure a Rician channel object ricChan =
comm.RicianChannel( ...
        'SampleRate',sampleRate500kHz, ... 'PathDelays',delayVector, ...
        'AveragePathGains',gainVector, ...
        'KFactor',KFactor, ...
        'DirectPathDopplerShift',specDopplerShift, ...
        'MaximumDopplerShift',maxDopplerShift, ...
        'RandomStream','mt19937ar with seed', ...
        'Seed',100, ...
        'PathGainsOutputPort',true);

qpskMod = comm.QPSKModulator('BitInput',true,'PhaseOffset',pi/4); % Number of bits
transmitted per frame is set to be 1000. For QPSK % modulation, this corresponds to 500
symbols per frame.
```

bitsPerFrame = 1000;
msg = randi([0 1],bitsPerFrame,1); % Modulate data for transmission over channel modSignal = qpskMod(msg); % Apply Rayleigh or Rician channel object on the modulated data

rayChan(modSignal); ricChan(modSignal);

release(rayChan); release(ricChan);

rayChan.Visualization = 'Impulse and frequency responses';

rayChan.SamplesToDisplay = '100%'; % Display impulse and frequency responses for 2 frames

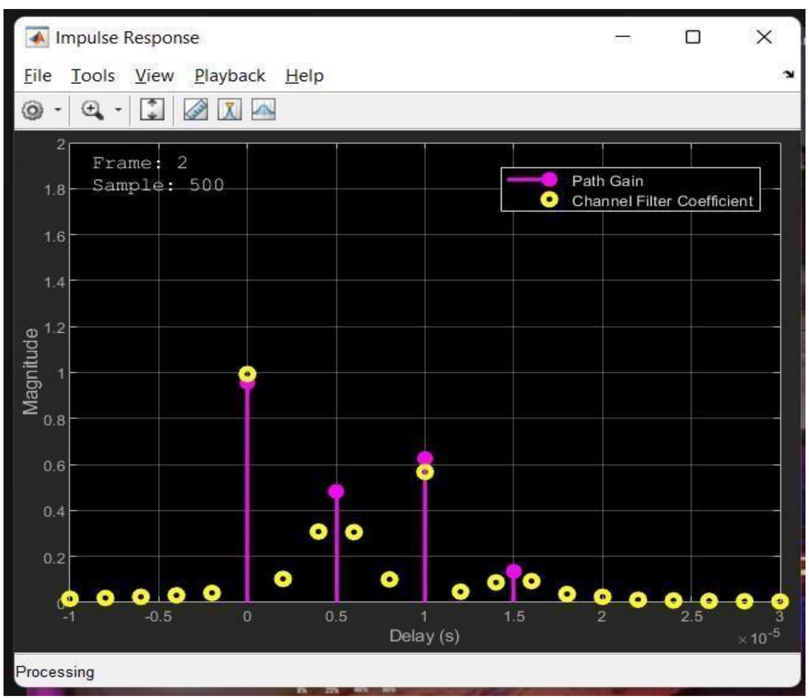
numFrames = 2;

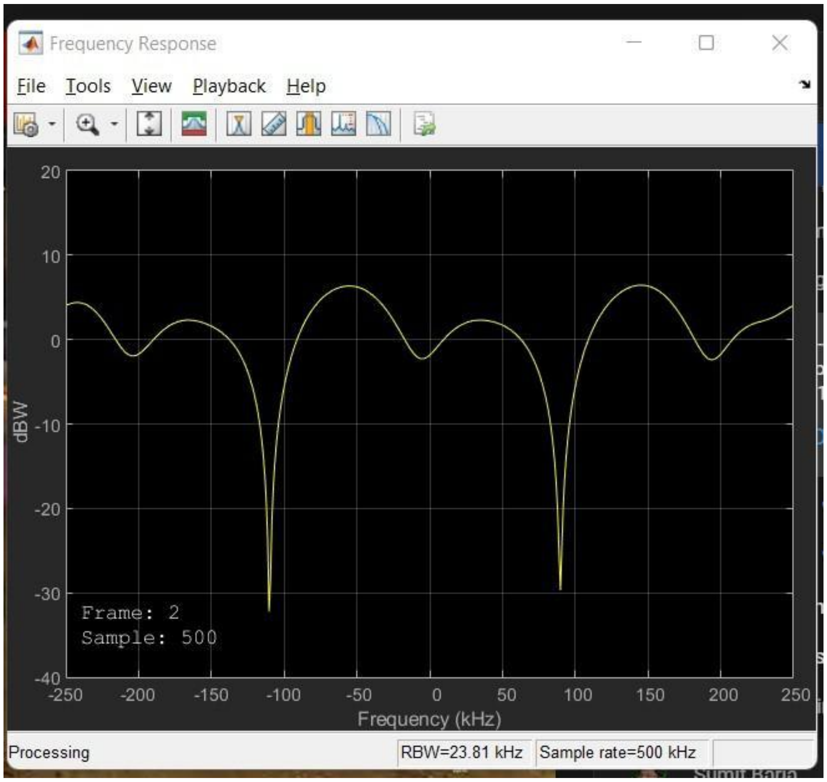
for i = 1:numFrames % Create random data

msg = randi([0 1],bitsPerFrame,1);% Modulate data

modSignal = qpskMod(msg); % Filter data through channel and show channel responses
rayChan(modSignal);
end

Output:




Frequency Response
File Tools View Playback Help
20
10
0
-10
-20
-30
-40
dBW
Frame: 2
Sample: 500
-250 -200 -150 -100 -50 0 50 100 150 200 250
Frequency (kHz)
Processing
RBW=23.81 kHz
Sample rate=500 kHz

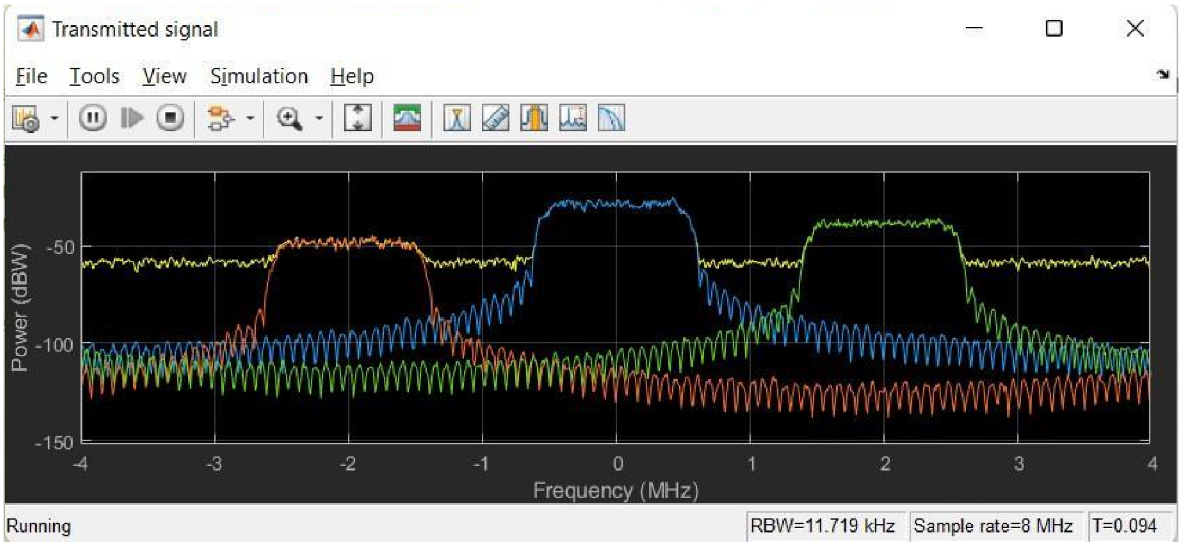
Practical: 10

Aim: Write a program in Matlab to calculate Co-channel and Adjacent Channel Interferences.

Input:

```
modelname = 'commadjcochanint';
open_system(modelname);% Define Simulink(R) objects as
variables trans = [modelname '/Transmitted signal'];
rec   = [modelname '/Received signal'];
const = [modelname '/Received constellation'];
% Set scope visibility
rec_sc = get_param(rec,'ScopeConfiguration');
rec_sc.OpenAtSimulationStart = false;
set_param( const, 'openScopeAtSimStart', 'off' );
trans_sc = get_param(trans,'ScopeConfiguration');
trans_sc.OpenAtSimulationStart = false;
sim(modelname);
close_system({trans, rec, const});
trans_sc.OpenAtSimulationStart = true; % Set scope visibility parameter
sim(modelname);% Set the simulation parameters for the next display:
trans_sc.OpenAtSimulationStart = false; rec_sc.OpenAtSimulationStart = true;
close_system(trans); % Close scope from previous simulation
sim(modelname);% Set the simulation parameters for the next
display: rec_sc.OpenAtSimulationStart = false;
set_param( const, 'openScopeAtSimStart', 'on' );
close_system(rec); % Close scope from previous simulation
sim(modelname);
close_system(modelname, 0);
```

Output:


Transmitted signal
File Tools View Simulation Help
-50
-100
-150
Power (dBW)
-4
-3
-2
-1
0
1
2
3
4
Frequency (MHz)
Running
RBW=11.719 kHz
Sample rate=8 MHz
T=0.094

Practical: 11

Aim: Write a program in Matlab to calculate the median path loss for Hata model for outdoor propagation.

Input:

```
clc;
clear all;
close
all; f=1;
t=input ('Enter type of city (1 - small or medium city, 2 - large
city)--:'); u=input ('Enter type of city (1 - urban , 2 - suburban,
3-rural)--:'); ht=input('Enter height of transmitting antenna(30 to
200m)--:'); hr=input('Enter height of receiving antenna(1 to 10m)--:');
f=input('Enter freuency in MHZ from 150 to 1500--:');
display('The median path loss for your given data is L50 in dB is');

if t==1
cf=(1.1*log(f)-0.7)*hr-(1.566*log(f)-0.8);
elseif f <= 300
cf=8.29*(log(1.54*hr))*(log(1.54*hr))-1.
1; else
cf=3.2*(log(11.75*hr))*(log(11.75*hr))-4.97
; end
        L50=69.55+26.16*log(f)-13.82*log(ht)-c
f; if u==2
   L50=L50-(2*log(f/28)*log(f/28))-5.
4; elseif u==3
      L50=L50-(4.78*log(f)*log(f))+18.33*log(f)-40.94;
end

display(L50)
; figure
f=1;
% plot for entire range of frequencies
display('The plot for entire range of frequencies from 150 MHz to 1500 MHz is shown in the
plot');
h = waitbar(0,'plotting the path loss for the entire range of frequencies please wait        ');
for f=150:2:1500
if t==1
cf=(1.1*log(f)-0.7)*hr-(1.566*log(f)-0.8);
elseif f <= 300
```

```
cf=8.29*(log(1.54*hr))*(log(1.54*hr))-1.
1; else
cf=3.2*(log(11.75*hr))*(log(11.75*hr))-4.97
; end
        L50=69.55+26.16*log(f)-13.82*log(ht)-c
f; if u==2
   L50=L50-(2*log(f/28)*log(f/28))-5.
4; elseif u==3
      L50=L50-(4.78*log(f)*log(f))+18.33*log(f)-40.94;
end
subplot(1,1,1);
plot(f,L50);hold on;

title('The plot for entire range of frequencies from 150 MHz to 1500 MHz');
xlabel('frequency(in MHz)');ylabel('Median path loss in dB');

waitbar(f / 1500)
end

close(h);
```

Output:

Command Window

```
Enter type of city (1 - small or medium city, 2 - large city)--:1
Enter type of city (1 - urban , 2 - suburban, 3-rural)--:1
Enter height of transmitting antenna(30 to 200m)--:40
Enter height of receiving antenna(1 to 10m)--:5
Enter freuency in MHZ from 150 to 1500--:200
The median path loss for your given data is L50 in dB is

L50 =

  139.0301

The plot for entire range of frequencies from 150 MHz to 1500 MHz is shown in the plot
fx >>
```