Dhirubhai Ambani
Institute of Information and Communication Technology

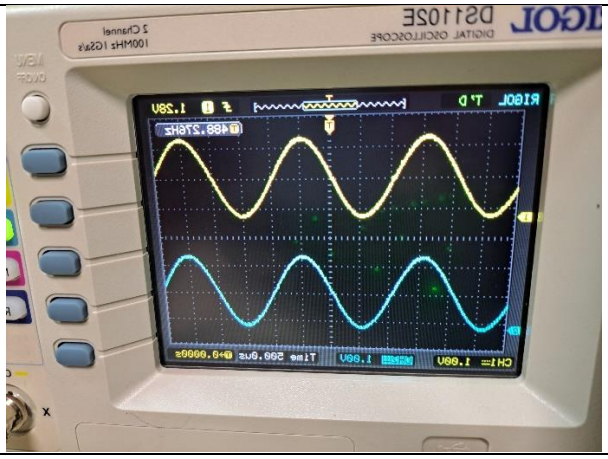
Digital Communication Lab3

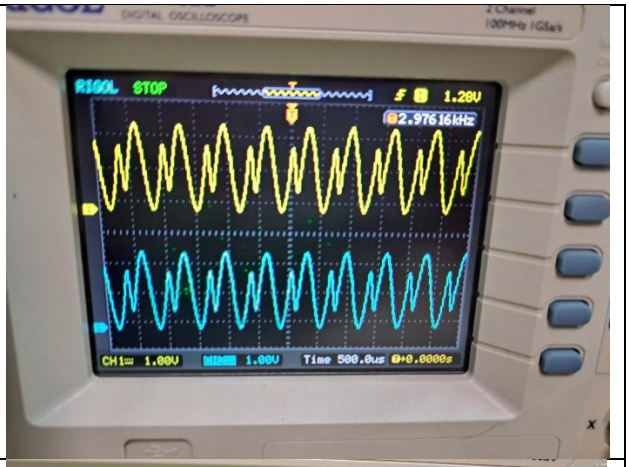
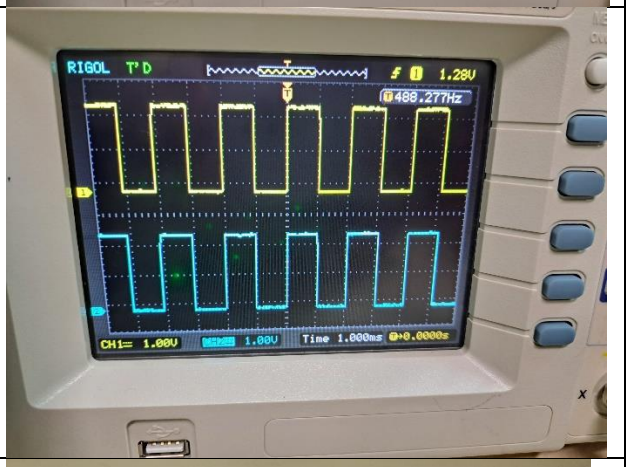
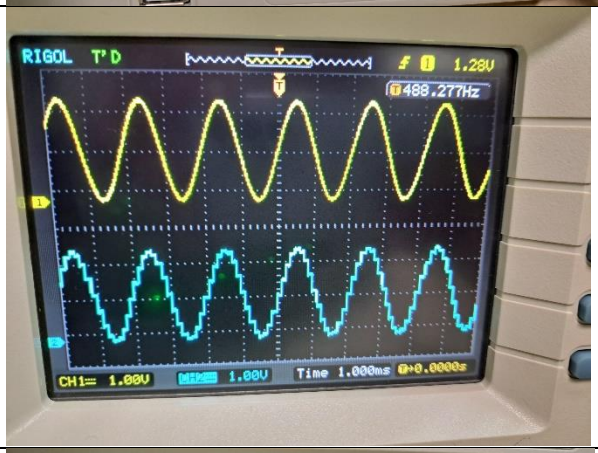
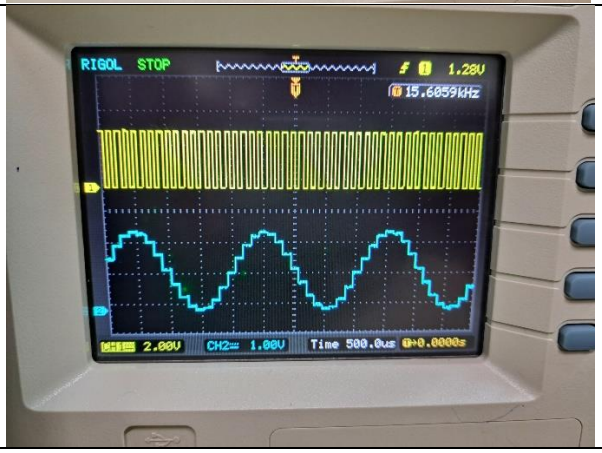
Name : Heman Jashvantbhai Chauhan

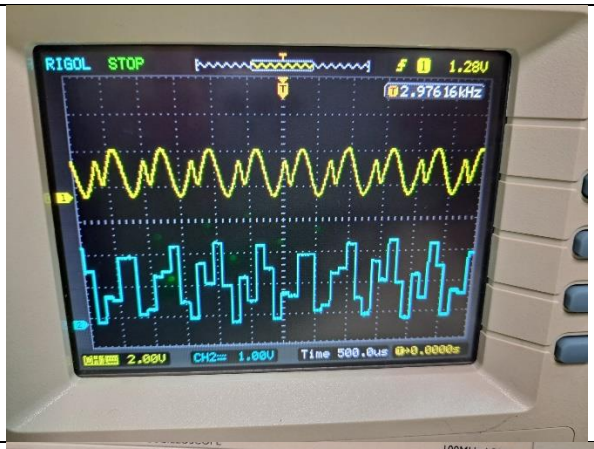
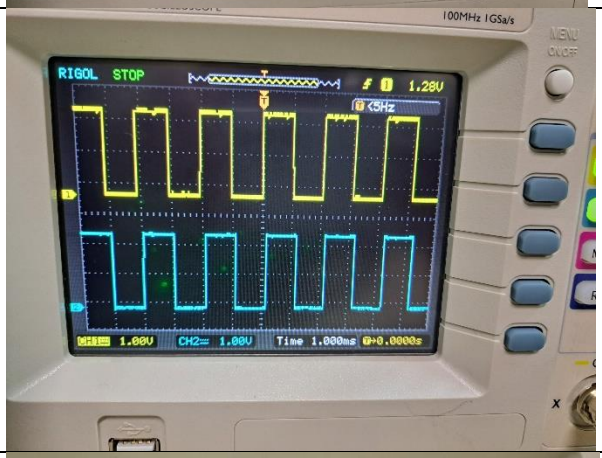
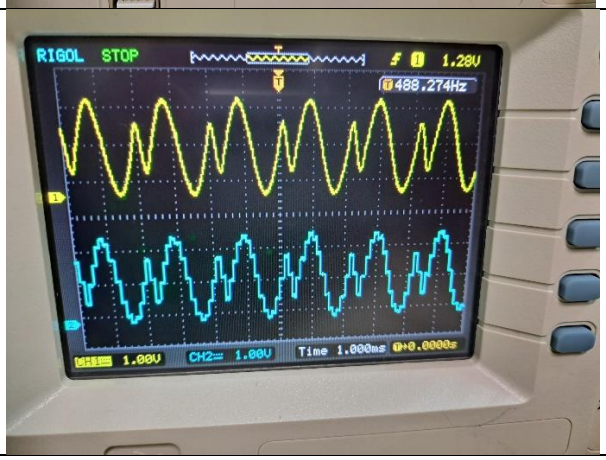
Student ID : 202201267

Task 1 :: **From [1], perform experiments 8 to 14, which pertain to TDM. The page numbers for the same are given in the table of contents. You need to follow the procedures given in the manual and observe the output as suggested in the observations. (Note: Time-division multiplexing (TDM) is one very useful application of PCM which allows multiple users to use a common channel. For section A, it will be soon covered in the lectures.)**

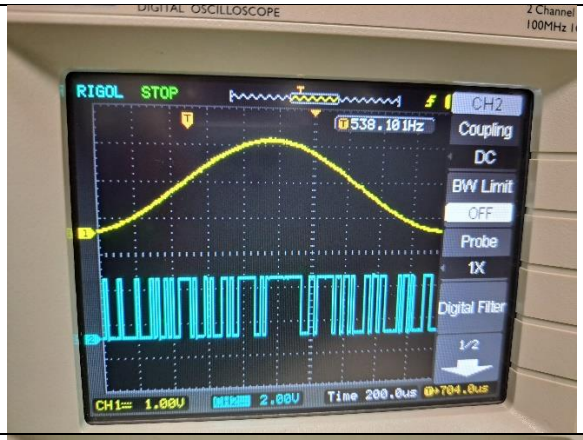
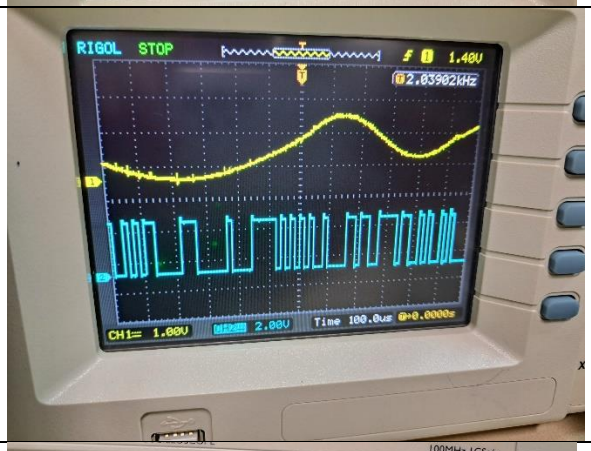
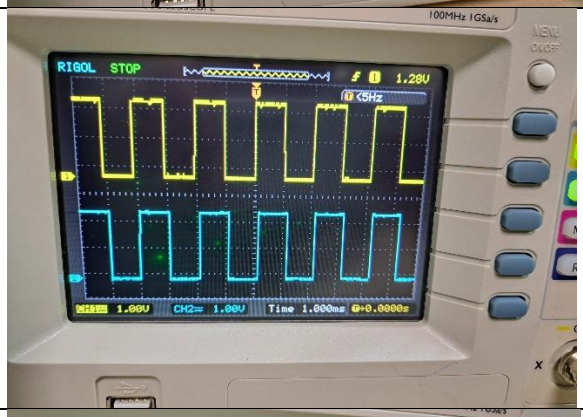
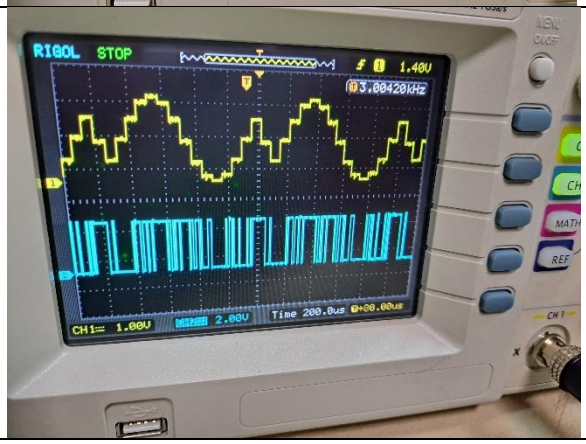
Experiment 8 ::

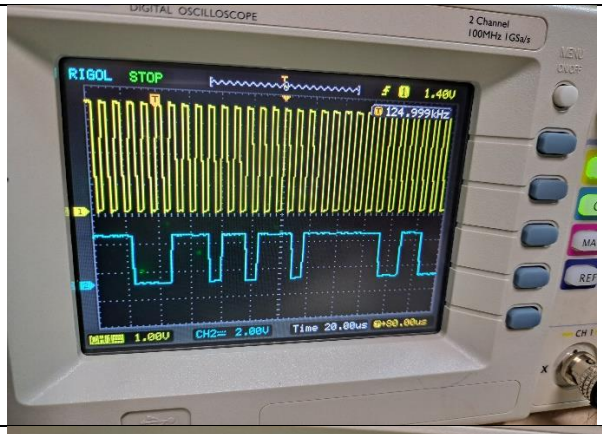
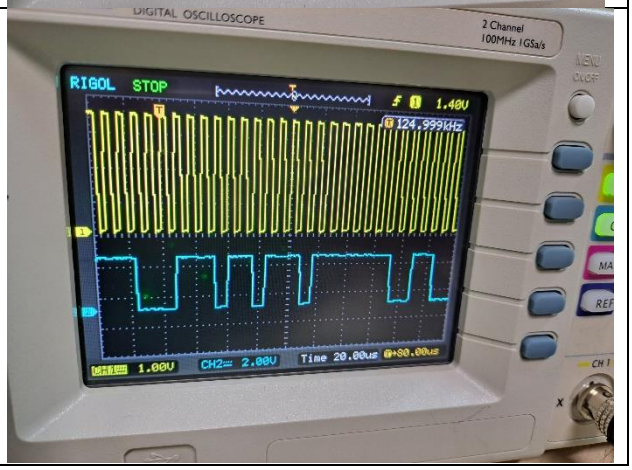
Sine/500 Hz	**Channel1**		

Arbitrary /1.5 KHz	**Channel2**		
Square /500 Hz	**Channel3**		
Sine/500 Hz	**Channel1**	**8 KHz**	
Sine/500 Hz	**Channel1**	**8 KHz**	

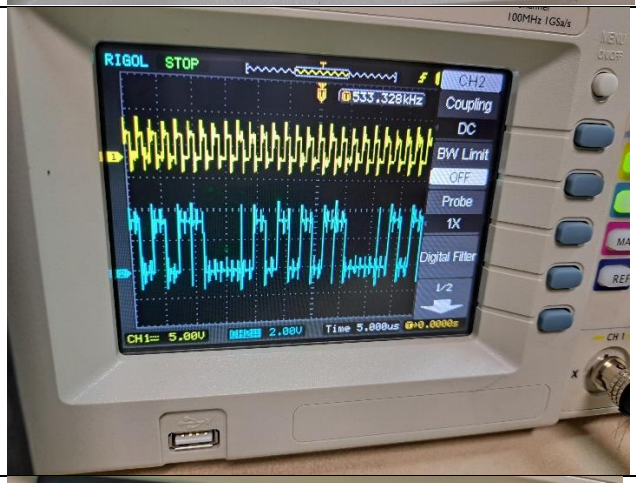
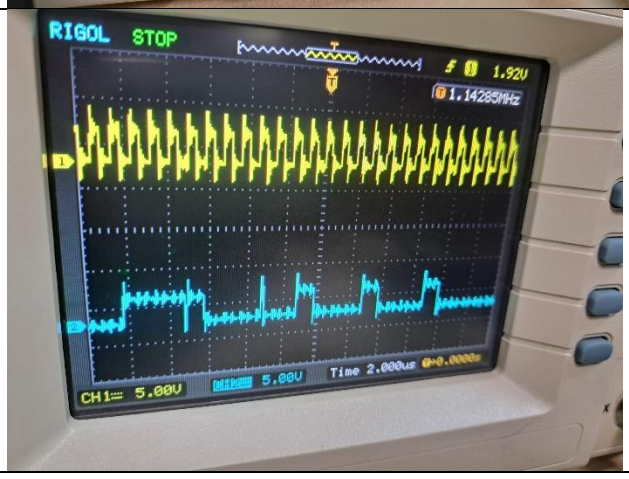
Arbitrary/1.5 KHz	**Channel2**	**16 KHz**	
Square/500 Hz	**Channel3**	**8 KHz**	
Arbitrary/1.5 KHz	**Channel4**	**8 KHz**	

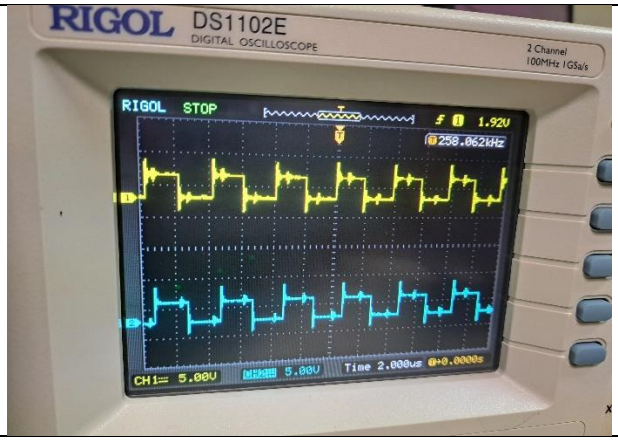
Experiment 9 ::

Sine/500 HZ	**Channel1**	**8 KHz**	
Arbitrary/ 500Hz	**Channel1**	**8 KHz**	
Square/500 Hz	**Channel1**	**8 KHz**	
Arbitrary /1.0KHz	**Channel1**	**16 KHz**	

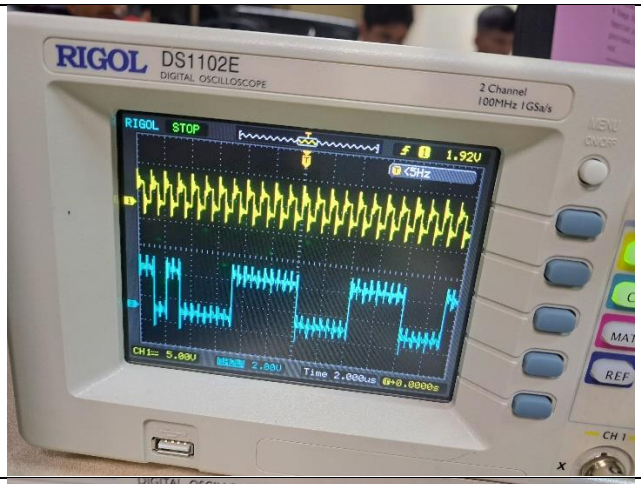
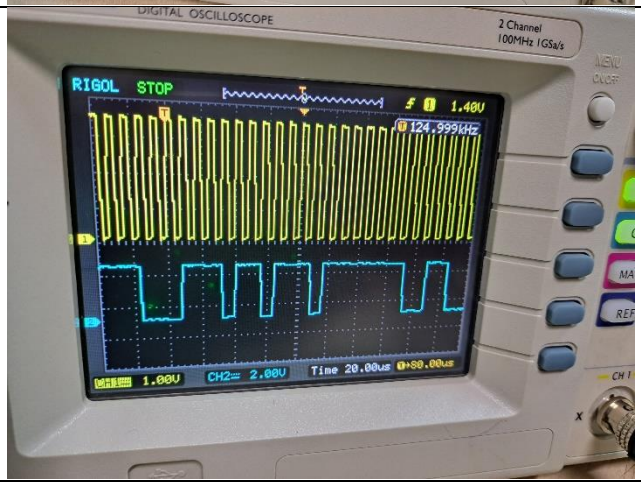
Sine/500 Hz	**Channel1**	**8 KHz**	
Sine/1.5 KHz	**Channel1**	**8 KHz**	
Sine/500 Hz	**Channel1**	**8 KHz**	

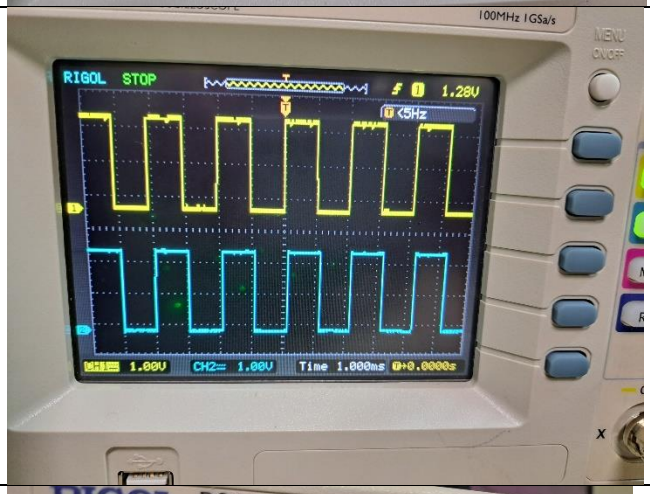
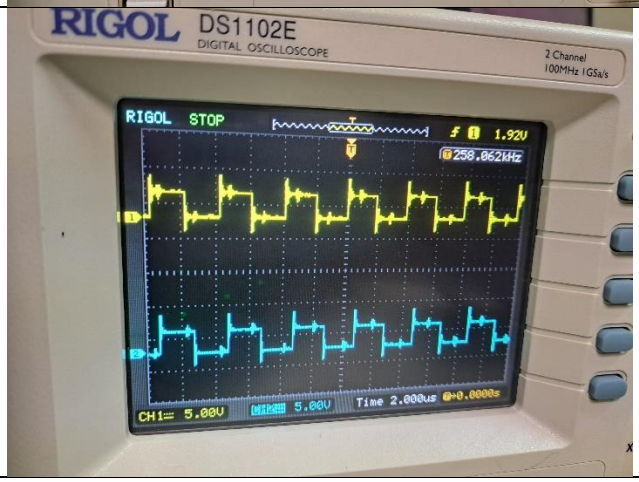
Experiment 10 ::

Sine/500 Hz		**8 KHz**	
Sine/500 Hz		**16 KHz**	
Sine/500 Hz		**32 KHz**	

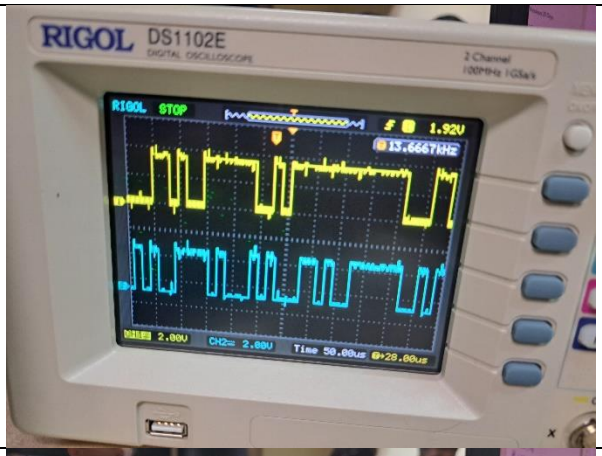
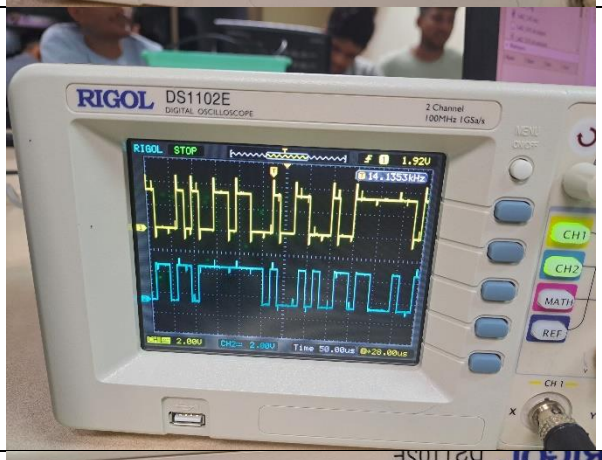
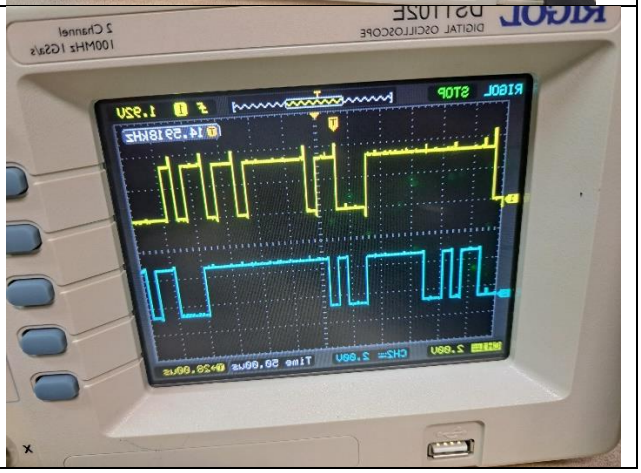
Sine/500 Hz		**8 KHz**	

Experiment 11 ::

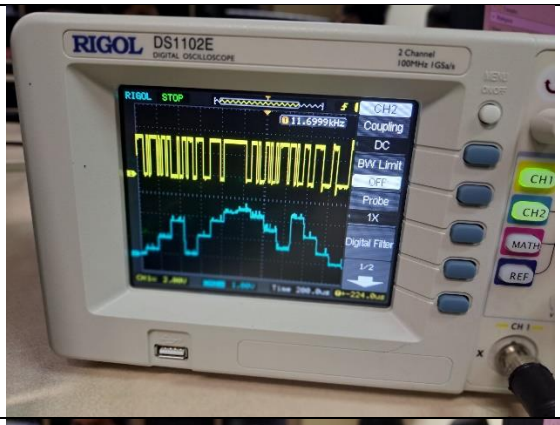
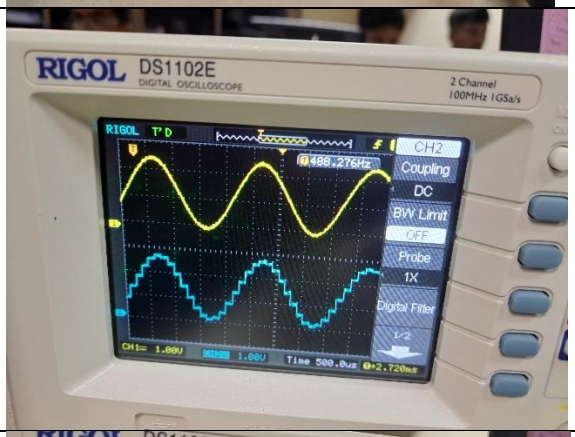
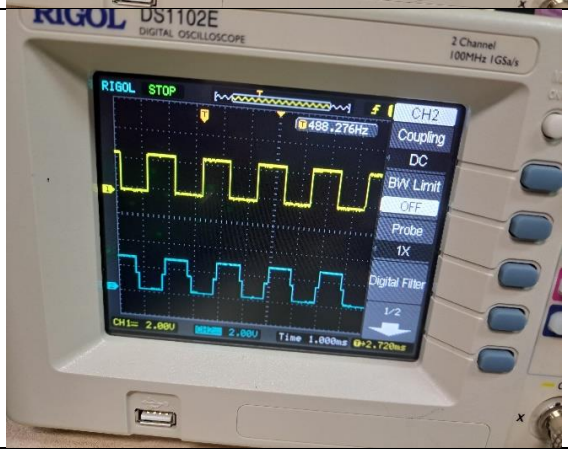
Sine/500 Hz		**8 KHz**	
Sine/500 Hz		**16 KHz**	

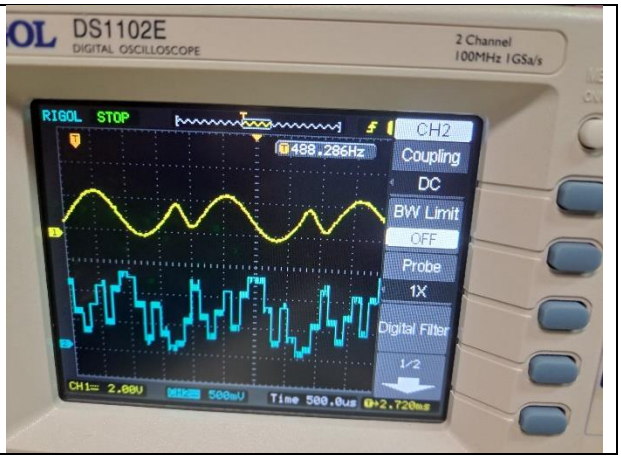
Sine/500 Hz		**32 KHz**	
Sine/500 Hz		**8 KHz**	
Sine/500 Hz		**8 KHz**	

Experiment 12 ::

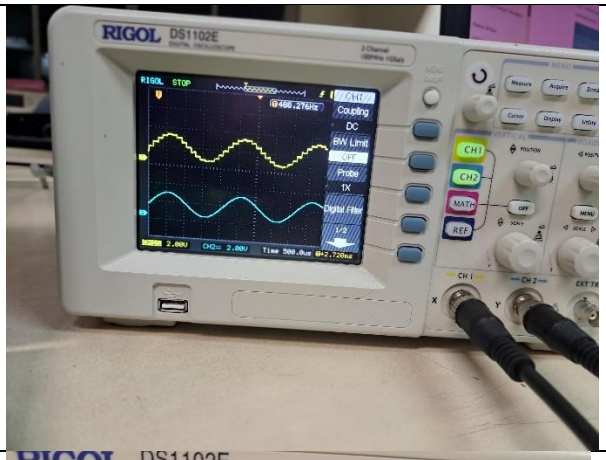
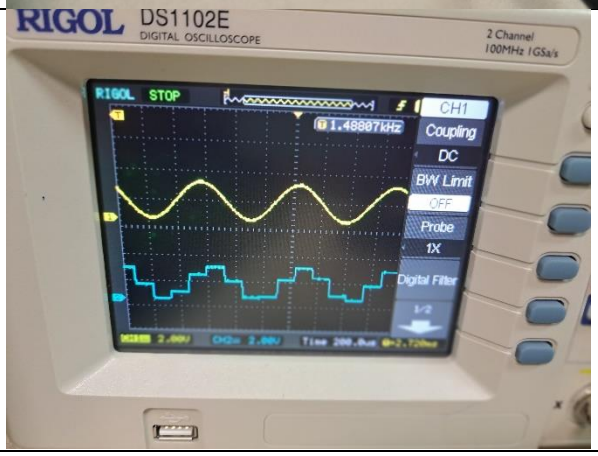
Sine/500 Hz	**Channel1**	**8 KHz**	
Sine/500 Hz	**Channel2**	**8 KHz**	
Sine/500 Hz	**Channel3**	**8 KHZ**	

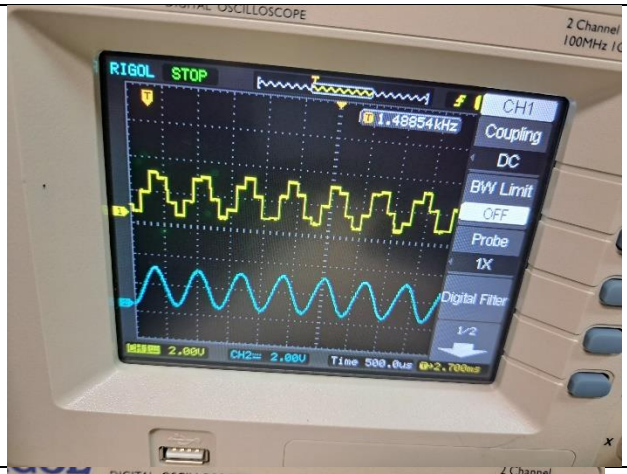
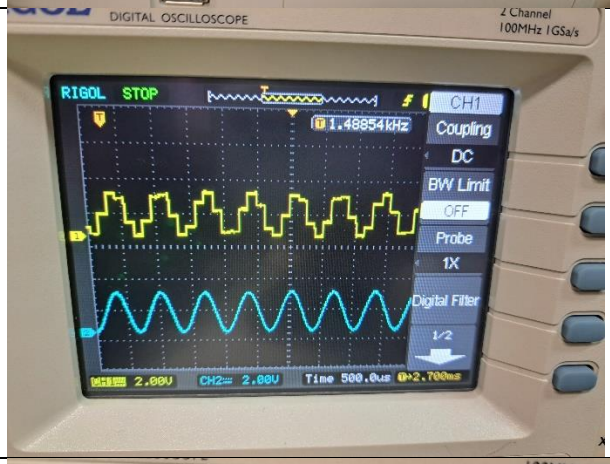
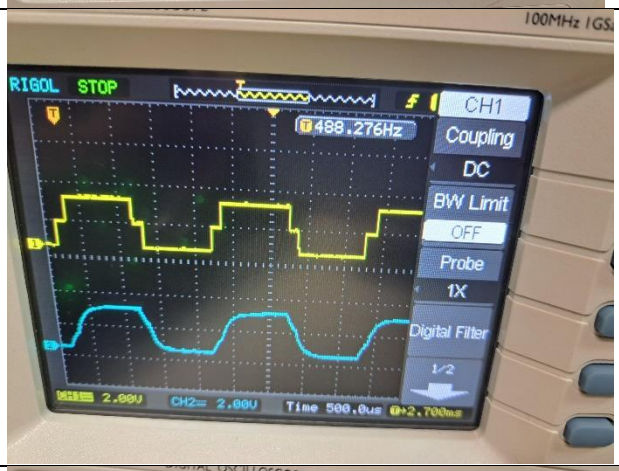
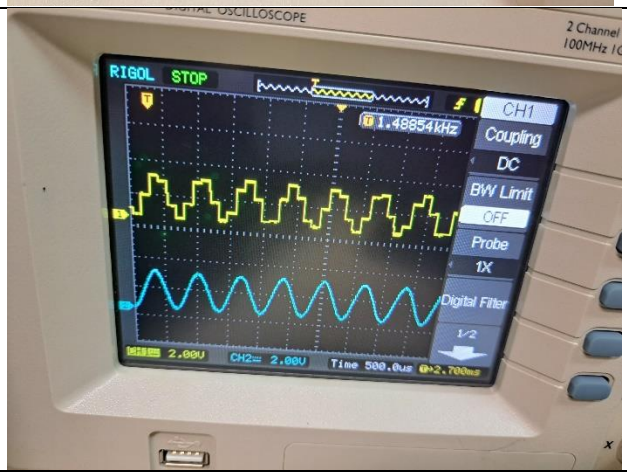
Experiment 13 ::

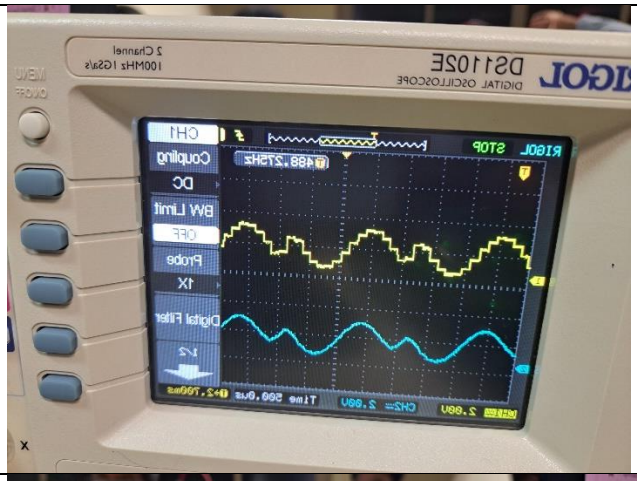
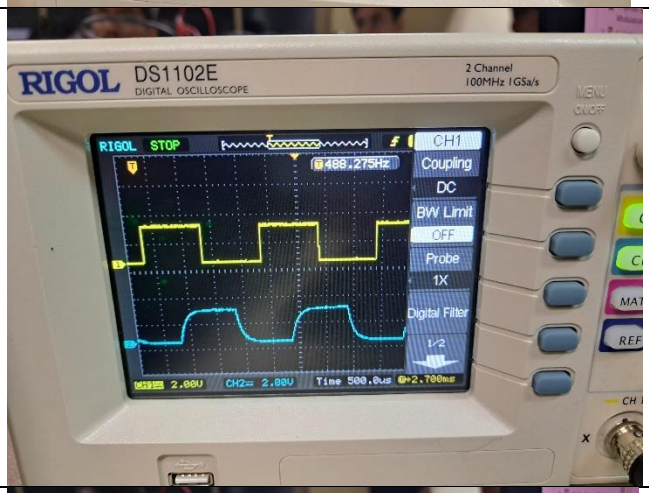
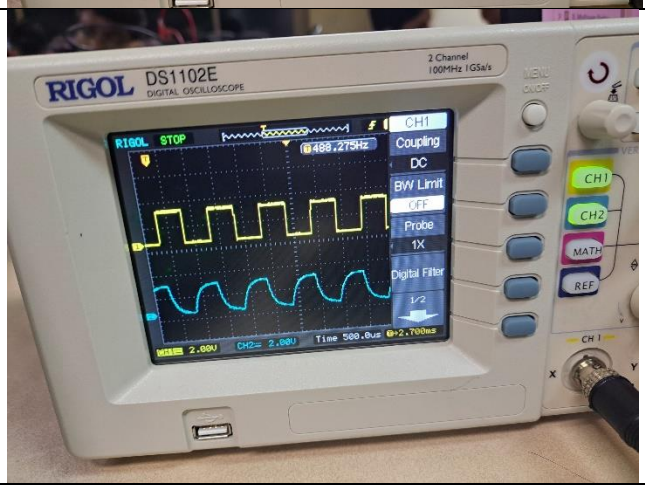
Sine/500 Hz	**Channel1**	**8 KHz**	
Sine/500 Hz	**Channel1**	**8 KHz**	
Arbitrary/500 Hz	**Channel2**	**8 KHz**	
Square/500 Hz	**Channel3**	**8 KHz**	

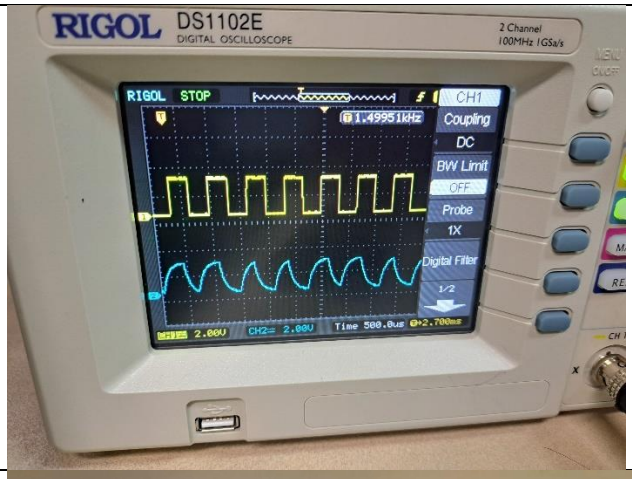
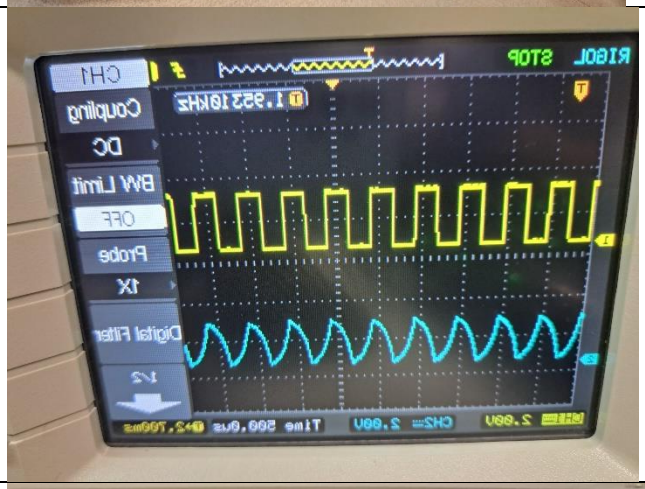
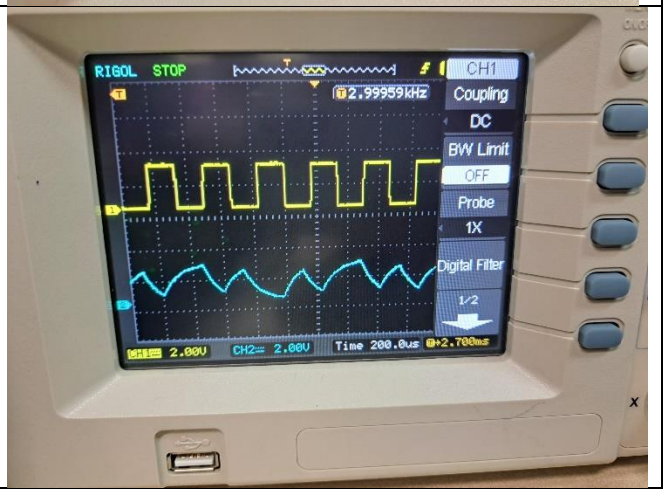
Arbitrary/500 Hz	**Channel4**	**8 KHz**	

Experiment 14 ::

Sine/500 Hz	**Channel1**	**8 KHz**	
Sine/1.5 KHz	**Channel1**	**8 KHZ**	

Sine/1.5 KHz	**Channel2**	**8 KHz**	
Sine/1.5 KHz	**Channel2**	**16 KHz**	
Square/500 Hz	**Channel3**	**8 KHz**	
Sine/1.5 KHz	**Channel3**	**8 Khz**	

Arbitrary/500 Hz	**Channel4**	**8 KHz**	
Square /500 Hz	**Channel1**	**8 KHz**	
Square/1 KHz	**Channel1**	**8 KHz**	

Square/1.5 KHz	**Channel1**	**8 KHz**	
Square/2 KHz	**Channel1**	**8 KHz**	
Square/3 KHz	**Channel1**	**8 Khz**	

➔ **Conclusion ::**

➔ **In this part we performed different sampling of signals and reconstruction of them like sine , square etc.**

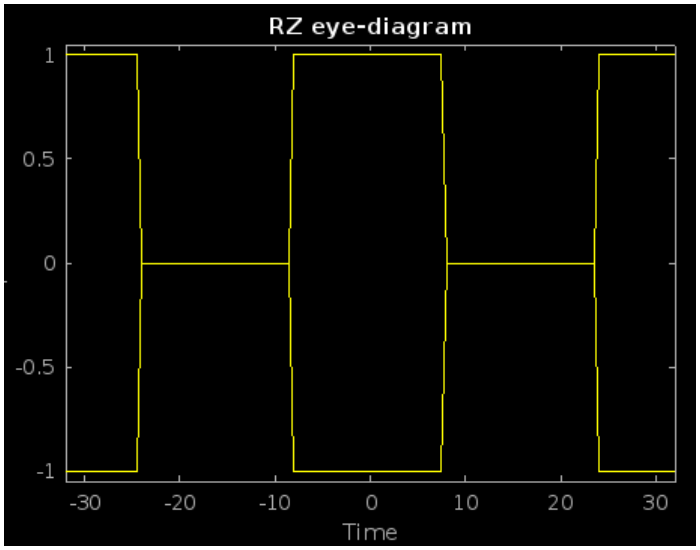
<u>Task 2 ::</u> Go through section 7.6 and subsequently, section 7.10 from [2]. Thereafter, implement the _les \pnrz.m", \prz.m", \psine.m", \prcos.m",

\binary eye.m", and \Mary eye.m". Subsequently, with the understanding developed, solve problem 4.38 from [3]. The function \eyediagram" is a built-in M-function in MATLAB. (Note: The concept of eye-diagram is well explained in section 7.6. However, it will be covered in the lectures on \communication over band-limited channels, scheduled for the second week after In-sem 1.)

Code ::

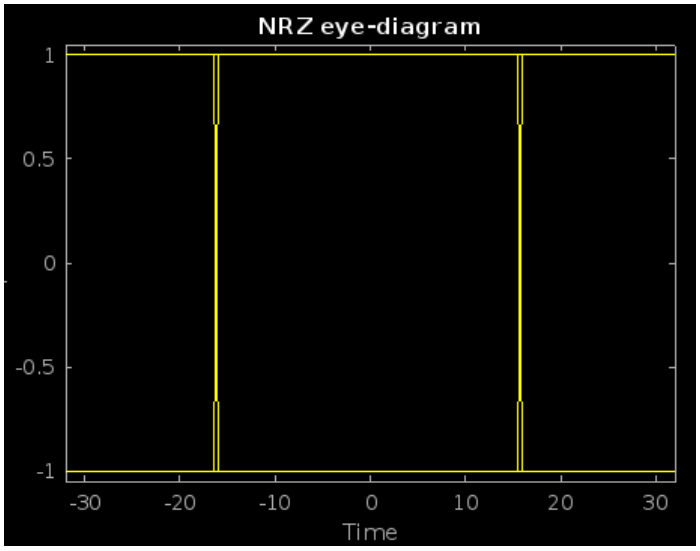
```
% (pnrz.m)
% Generating a rectangular pulse of width T
% Usage: function pout = pnrz(T)
function pout = pnrz(T)
    % Generate a rectangular pulse of width T
    pout = ones(1, T);
end

% (prz.m)
% Generating a rectangular pulse of width T/2
% Usage: function pout = prz(T)
function pout = prz(T)
    % Generate a rectangular pulse of width T/2
    l = 1; % Set the number of rows (for a single-row vector)
    pout = [zeros(l, T/4), ones(l, T/2), zeros(l, T/4)];
end

% (psine.m)
% Generating a sinusoidal pulse of width T
function pout = psine(T)
    % Generate a sinusoidal pulse of width T
    pout = sin(pi * (0:T-1) / T);
end

% (prcos.m)
function  y  = prcos(rollfac,length,T)
y = rcosdesign(rollfac,length,T);
end

clear; clc;

data= sign(randn(1, 400)); % Generate 400 random bits

Tau=64 ; % Define the symbol period
```

```
dataup= upsample(data, Tau) ; % Generate impulse train

yrz= conv(dataup, prz(Tau)); % Return to zero polar signal

yrz= yrz (1: end-Tau+1);

ynrz= conv(dataup, pnrz(Tau)); % Non-return to zero polar

ynrz= ynrz(1:end-Tau+1) ;

ysine=conv(dataup, psine (Tau)); % half sinusoid polar

ysine=ysine (1: end-Tau+1);

Td=4; % truncating raised cosine to 4 periods

yrcos= conv( dataup , prcos (0.5, Td, Tau)); % rolloff factor = 0.5

yrcos= yrcos(2*Td*Tau : end- 2*Td*Tau + 1) ; % generating RC pulse train

eyel= eyediagram(yrz, 2*Tau, Tau, Tau/2);

title('RZ eye-diagram');
```



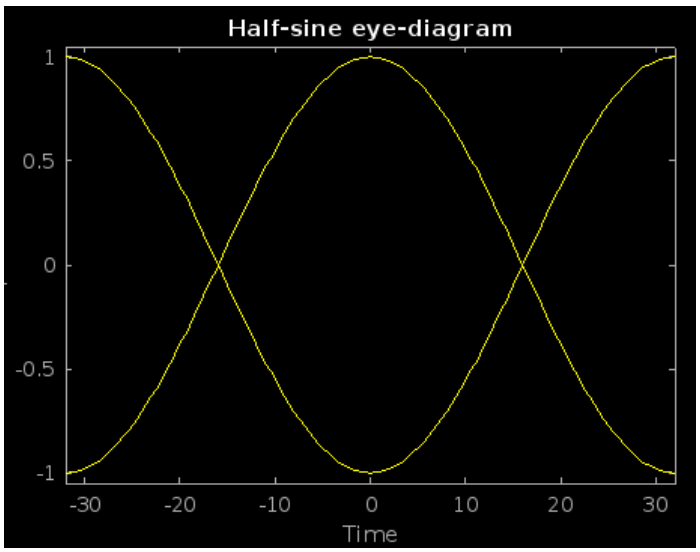

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
eye2= eyediagram(ynrz , 2*Tau, Tau, Tau/2);

title('NRZ eye-diagram');
```



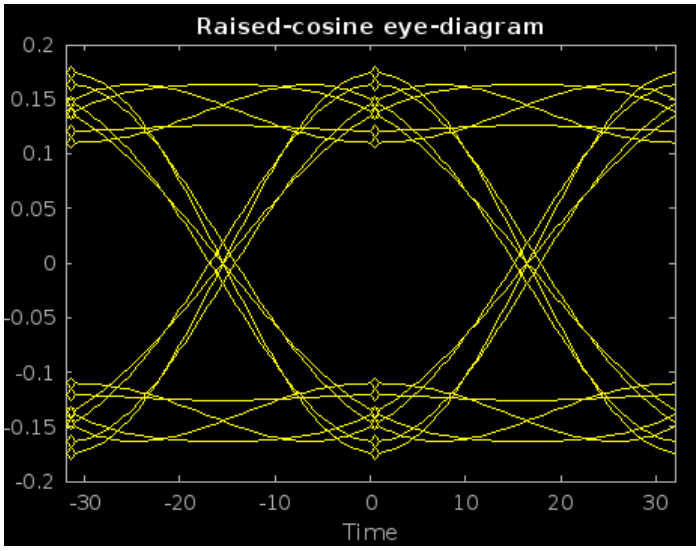

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
eye3= eyediagram(ysine, 2*Tau, Tau, Tau/2);

title('Half-sine eye-diagram');
```



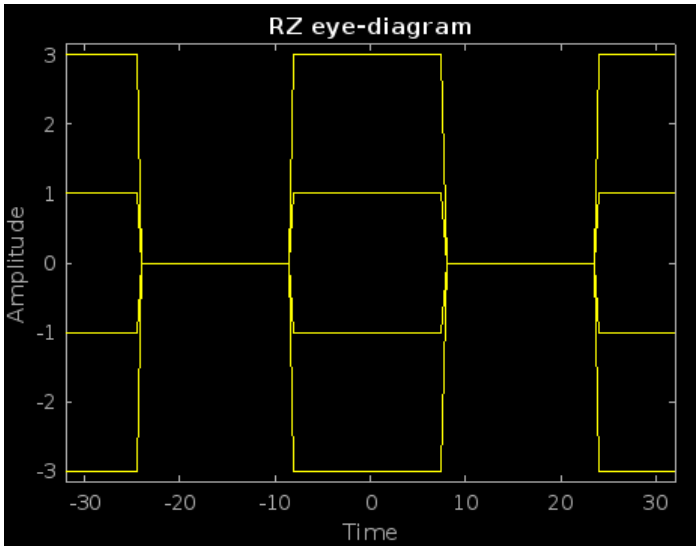

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
eye4= eyediagram( yrcos, 2*Tau, Tau) ;

title('Raised-cosine eye-diagram');
```



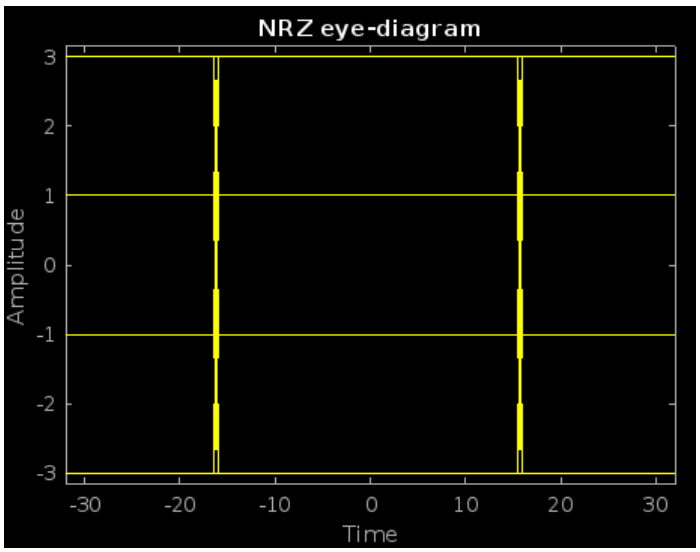

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
clear; clc;

data= sign(randn(1, 400)) + 2*sign(randn(1, 400) ); % 400 PAM symbols

Tau=64; % Define the symbol period

dataup= upsample(data , Tau) ; % Generate impulse train

yrz =conv(dataup, prz(Tau)); % Return to zero polar signal

yrz =yrz (1: end-Tau+1) ;

ynrz =conv(dataup, pnrz (Tau)); % Non-return to zero polar

ynrz =ynrz(1:end-Tau+1) ;

ysine= conv(dataup, psine (Tau)); % half sinusoid polar

ysine= ysine (1 : end-Tau+1) ;

Td=4 ; % truncating raised cosine to 4 periods

yrcos= conv( dataup, prcos (0.5, Td, Tau)); % rolloff factor = 0.5

yrcos= yrcos(2*Td*Tau : end-2*Td*Tau+1); % generating RC pulse train

eye1=eyediagram (yrz , 2*Tau, Tau, Tau/2);

title ('RZ eye-diagram');
```



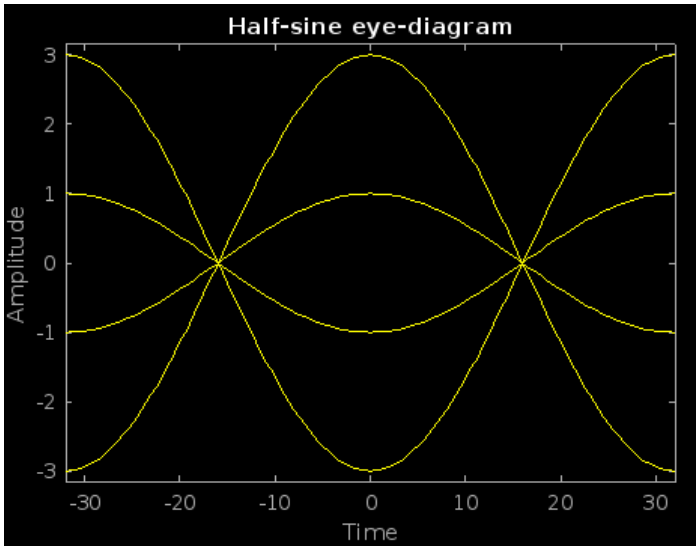

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
eye2=eyediagram ( ynrz , 2*Tau, Tau, Tau/2);

title ('NRZ eye-diagram');
```



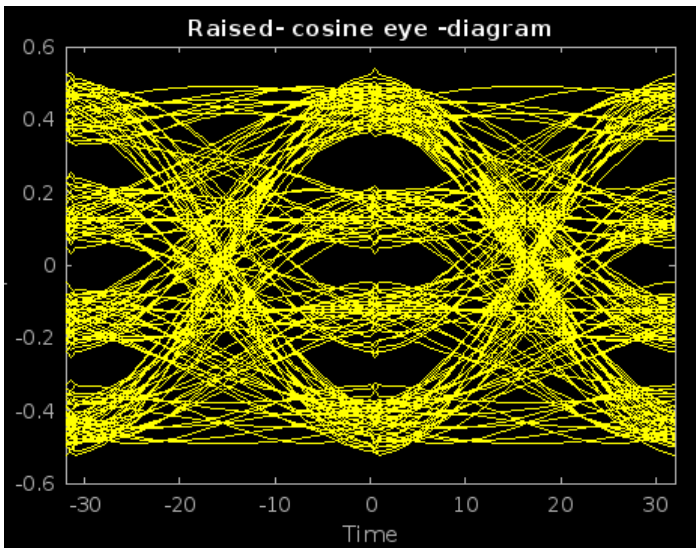

```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
eye3=eyediagram (ysine , 2*Tau, Tau, Tau/2);

title('Half-sine eye-diagram');
```




```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

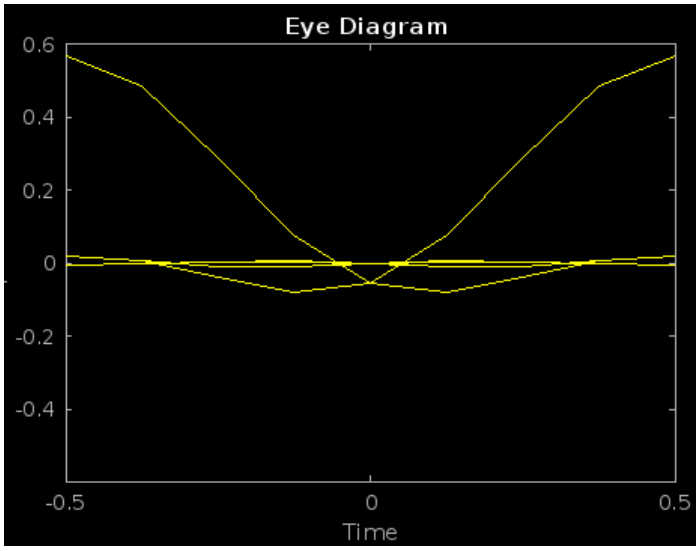
```
eye4=eyediagram ( yrcos , 2 * Tau, Tau);

title('Raised- cosine eye -diagram');
```




```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
clear all

%Define the M-ary number, calculation sample frequency

M=4; Fs=20;
```

```
%Define the number of points in the calculation

pd=500;


%Generate an integer message in range [0, m-1]

msg_d = exp_randint(pd,1,M);


%Use square constellation PAM method for modulation

msg_a = exp_modmap(msg_d,Fs,M);


%nonlinear channel

alpha=0.0;

msg_a=msg_a +alpha*msg_a.^2;


%raised cosine filtering

span = 10;

rolloff = 0.5;

sps = 4; % samples per symbol

rcv_a = rcosdesign(rolloff, span, sps, 'sqrt');


% rcv_a=rcosdesign(msg_a, Fs);


%eye pattern

eyediagram(rcv_a, 2 * sps);
```




```
Warning: Error updating Legend.

Execution of script eye as a function is not supported:
/MATLAB Drive/eye.m
```

```
function y= exp_modmap(x,Fs,M)

%PAM modulation

x=x-(M-1)/2;

x=2*x/(M-1);

y=zeros(length(x)*Fs,1);

p=0;

for k=1:Fs:length(y)

p=p+1;

y(k:(k+Fs-1))=x(p)*ones(Fs,1);

end

end


function out = exp_randint(p,q,r)

%random integer generator
```

```
r = [0, r-1];

r = sort(r);

r(1) = ceil(r(1));

r(2) = floor(r(2));

if r(1) == r(2)

out = ones(p,q) * r(1);

return;

end

d = r(2) - r(1);

r1 = rand(p,q);

out = ones(p,q)*r(1);

for i = 1:d

index = find(r1 >= i/(d+1));

out(index) = (r(1) +1) * index./index;

end

end
```

➔ **Conclusion ::**

We plotted different eye diagrams of the different sampled signals like pulse , sinusoidal , cosine and etc .