ID	الاسم
10	إبراهيم صبحى إبراهيم
14	أحمد حسن عبد القادر
19	أحمد سمير محمد حسانين
129	عمر عادل عمر فتوح ابو سيد احمد
131	عمر عبد العزيز فهمى راجح على
259	يحيى حسام على محمد سليمان

Digital Communication Project

1/6/22

```
%%%%%%%%%%%part 1 %%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fs = 5e6;                    % Sampling rate (samples per sec)
Ts = 1/fs;                   % Sampling time
N = 102400 - 1;              % Total number of samples
t_axis = (0:N-1)*Ts;         % Time axis (the same during the entire experiment)
t_axis_sh = ((-(N-1)/2):((N-1)/2))*Ts;
f_axis = -fs/2:fs/N:fs/2-1/N;   % Frequency axis (the same during the entire experiment)

% Generate one square pulse with the following parameters
Energy_per_bit = 50.5;          % The total energy of all samples constituting the square pulse
B = 100*10^3;
T_sq = 2/B;
N_sq = round(T_sq/Ts);         %N_sq = 100

%generating one pulse
x_bits = 1;
x_square = GenerateSquarePulses(t_axis,T_sq,Energy_per_bit,fs,x_bits,1);

figure
subplot(2,1,1)

plot(t_axis,x_square,'linewidth',2)
grid on
xlim([0 T_sq*1.2])
xlabel('Time','linewidth',2)
ylabel('Square pulse','linewidth',2)

X_squareF = (1/fs)*abs(fftshift(fft(x_square)));


subplot(2,1,2)
plot(f_axis,abs(X_squareF),'linewidth',2)
title('Square in freq','linewidth',10)
grid on
xlim([-1/T_sq 1/T_sq]*5)
xlabel('Frequency','linewidth',2)
ylabel('Frequency ressponse magnitude','linewidth',2)
subplot(2,1,1)
title('A square pulse in time and frequency domains','linewidth',10)

%creating the channel

channelf = rectpuls(f_axis , 2*B);
channel = ifft(ifftshift(channelf));
figure
subplot(2,1,1)
```

```
plot(f_axis,abs(channelf),'linewidth',2)
title('Channel in Frequency','linewidth',10)
xlim([-1/T_sq 1/T_sq]*5)
subplot(2,1,2)
plot(t_axis,channel,'linewidth',2)
title('Channel in Time','linewidth',10)
xlim([0 T_sq*1.2])

%passing the pulse to the channel

temp = conv(x_square , channel);
y = temp(:,1:length(t_axis));
figure
grid on
plot(t_axis,y,'linewidth',2)
title('Recieved pulse in Time','linewidth',10)
xlim([0 T_sq*2])
yf = (1/fs)*abs(fftshift(fft(y)));
figure
plot(f_axis,yf,'linewidth',2)
title('Recieved pulse in Frequency','linewidth',10)
xlim([-1/T_sq 1/T_sq]*5)

%creating two separate pulses

x_bits = 1;
x_square1 = GenerateSquarePulses(t_axis,T_sq,Energy_per_bit,fs,x_bits,1);

x_bits = [0 1];
x_square2 = GenerateSquarePulses(t_axis,T_sq,Energy_per_bit,fs,x_bits,1);
figure
plot(t_axis,x_square1,t_axis,x_square2 , 'linewidth',2)
xlim([0 T_sq*2.4])

%passing the two pulses to the channel

temp = conv(x_square1 , channel);
y1 = temp(:,1:length(t_axis));

temp = conv(x_square2 , channel);
y2 = temp(:,1:length(t_axis));


figure
grid on
plot(t_axis,y1,t_axis,y2,'linewidth',2)
title('Recieved pulse in Time','linewidth',10)
xlim([0 T_sq*3])
```

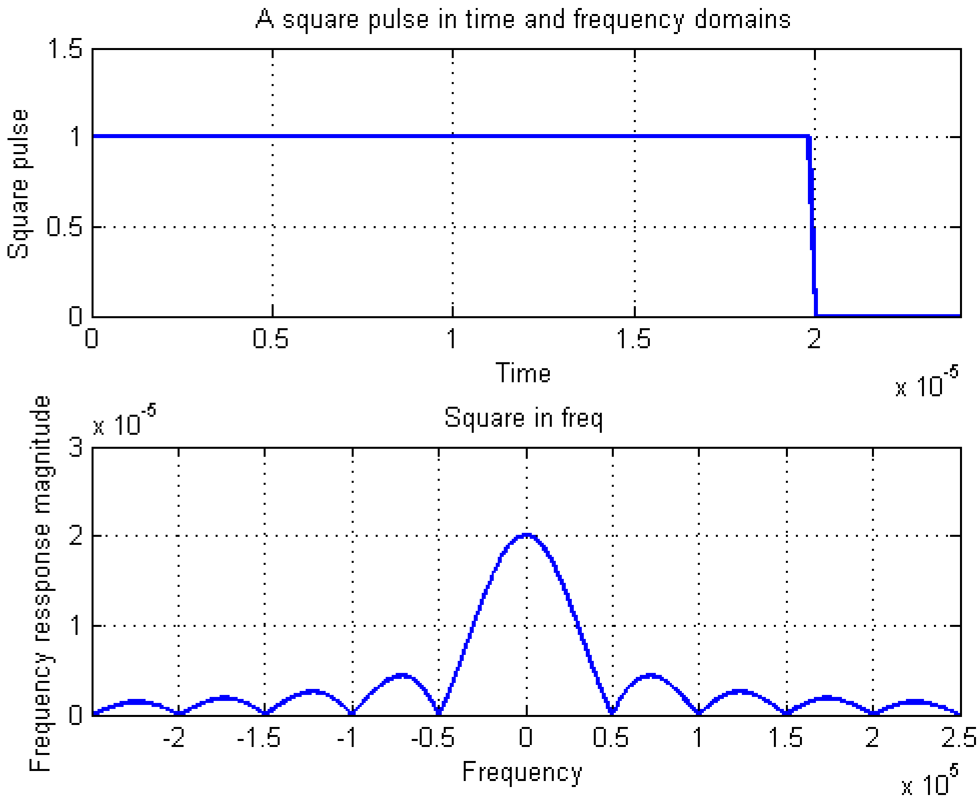
```
%raised cosine

x_bits = [1 1 0 1 0 1];
x = GenerateSquarePulses(t_axis_sh,T_sq,Energy_per_bit,fs,x_bits,2);
figure
plot(t_axis_sh,x , 'linewidth',2)
xlim([-T_sq*2 T_sq*6])
title('Transmitted signal in Time with Rasied Cosine','linewidth',10)
grid on

%passing through channel
temp = conv(x , channel);
y1 = temp(:,1:length(t_axis));
figure
plot(t_axis_sh, y1 , 'linewidth',2)
xlim([-T_sq*2 T_sq*6])
title('Recievied signal in Time with Rasied Cosine','linewidth',10)
grid on
```

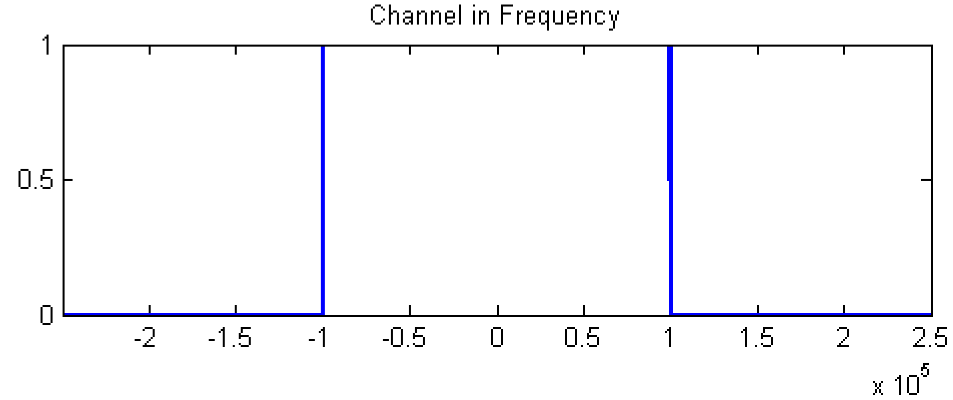
```
function x_square = GenerateSquarePulses(t_axis,T_sq,E_bit,fs,x_bits,type)

Ts = 1/fs;
N = length(t_axis);


N_sq = round(T_sq/Ts);
one_square = zeros(1,N_sq);
if type == 1
    A = sqrt(2*E_bit / N_sq);
    one_square = A + one_square;
else
    alpha = 0.5;
    W = 1/T_sq;
    n = 102400 - 1;
    t_axis_sh = ((-(n-1)/2):((n-1)/2))*Ts;
    one_raised_cos = sinc(2*W*t_axis_sh).*cos(2*pi*alpha*W*t_axis_sh)./(1-(4*alpha*W*t_axis).^2);
end
x_square = zeros(1, N);

for i = 1:length(x_bits)
    if type == 1
        if x_bits(i) == 1
            x_square((i-1)*N_sq +1:i*N_sq) = x_square((i-1)*N_sq +1:i*N_sq) + one_square;
        end
    else
        if x_bits(i) == 1
            x_square = x_square + circshift(one_raised_cos' , N_sq*(i-1))';
        else
```

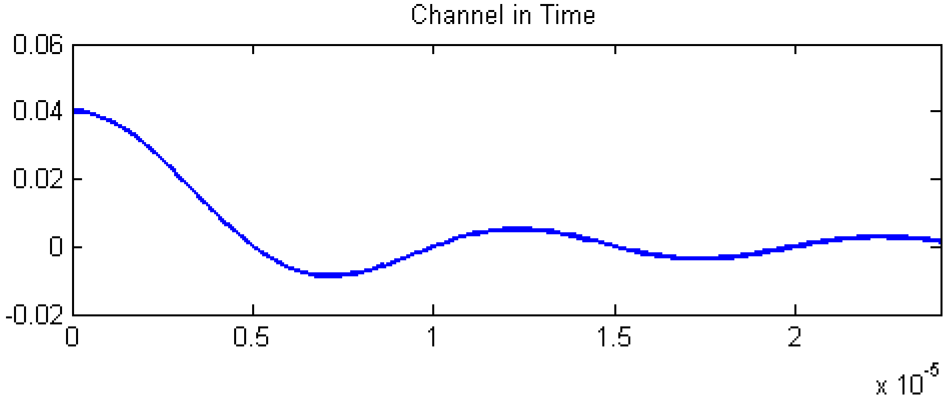
```
            x_square = x_square - circshift(one_raised_cos' , N_sq*(i-1))';
        end
    end
end

end
```

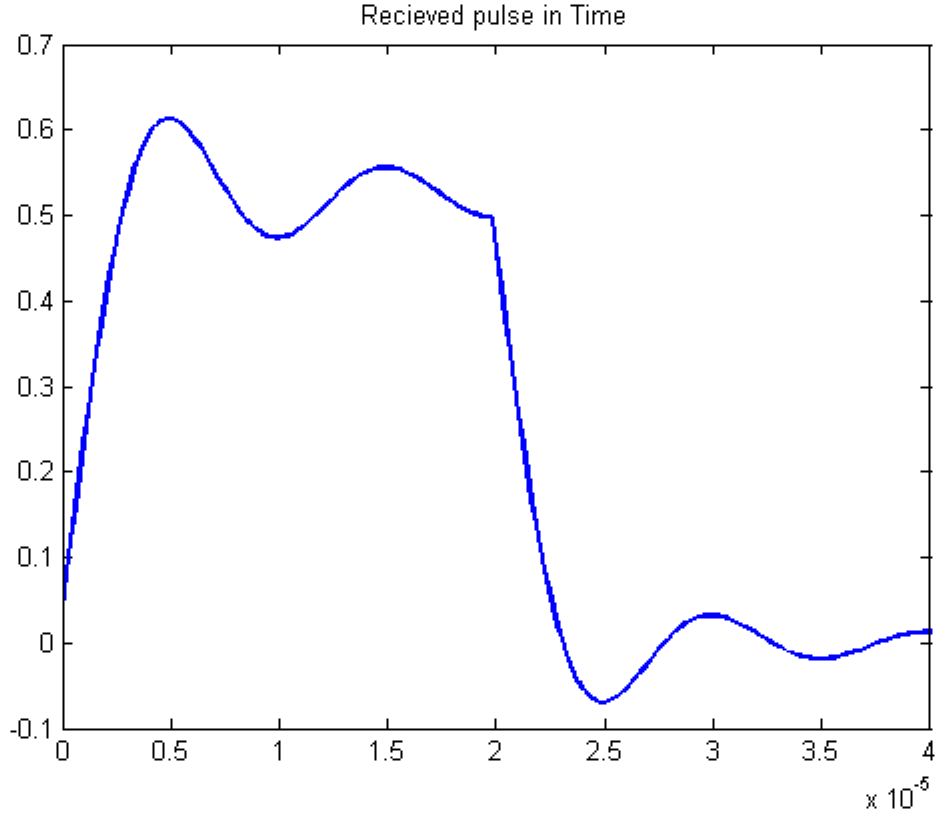
```
L=1000;
N=1;
h = randn(L,N) + 1i*randn(L,N);
power_profile = exp(-0.5*[0:L-1])';
power_profile = repmat(power_profile,1,N);
h = abs(h).*power_profile;

H = zeros(L,L);
last_ind = 1;
z = 1;
for n = 1:L
    for k = last_ind:-1:1
        H(n,z) = h(k);
        z = z +1;
    end
    z = 1;
    last_ind = last_ind + 1;
end
H_inv = inv(H);
BER_temp = [];
Eb = 1;
No = [1 0.5 0.1 0.05 0.01 0.005 0.001];
BER = [];
for nn = No
    N = sqrt(nn/2)*randn(L,1);
    for i = 1:10        %we will calculate the BER 10 times for each noise
        valid_vals = setdiff(-1:1, 0);
        x = valid_vals( randi(length(valid_vals), L, 1) ); %stream of bits transmitted

        y = H*x' + N;        %recieved

        x_rec = H_inv*y;     %recieved bits plus noise
        D = zeros(size(x_rec));
        for k = 1:L
            if x_rec(k) <= 0
                D(k)= -1 ;
            else
                D(k) = 1;
            end
        end

        n = 0;
        for k = 1:L
            if D(k) ~= x(k)
                n = n + 1;
            end
        end


        BER_temp = [BER_temp n/L];
    end
    BER = [BER mean(BER_temp)];
    BER_temp = [];
end

plot(Eb./No , BER)
xlim([0 100])
```

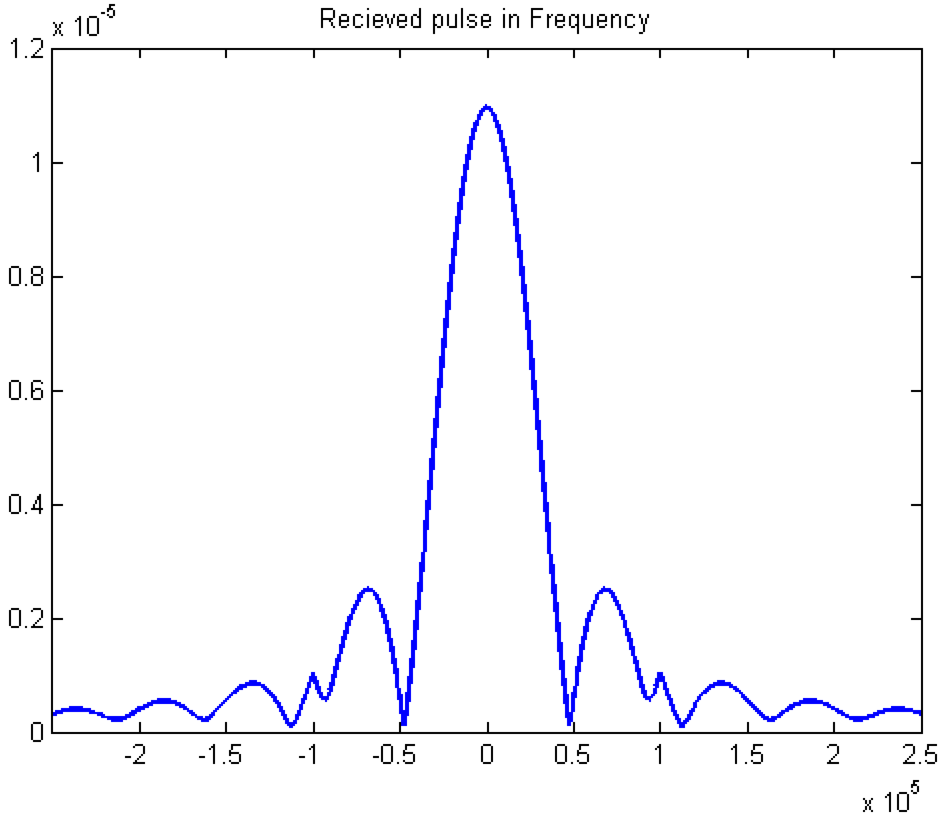
Outputs:

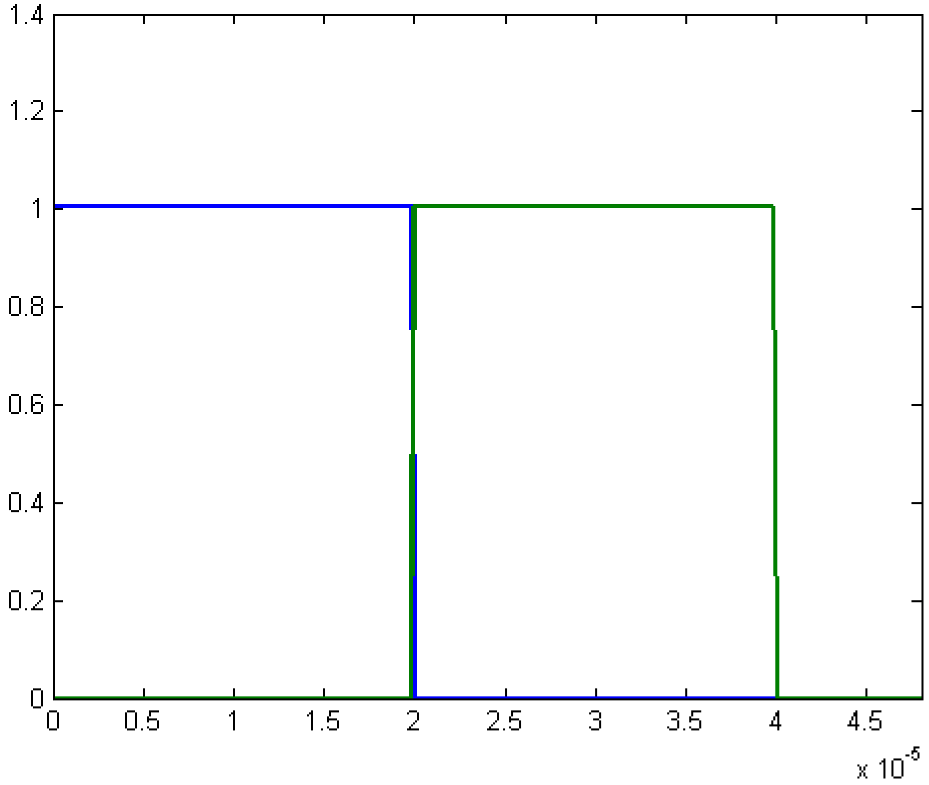


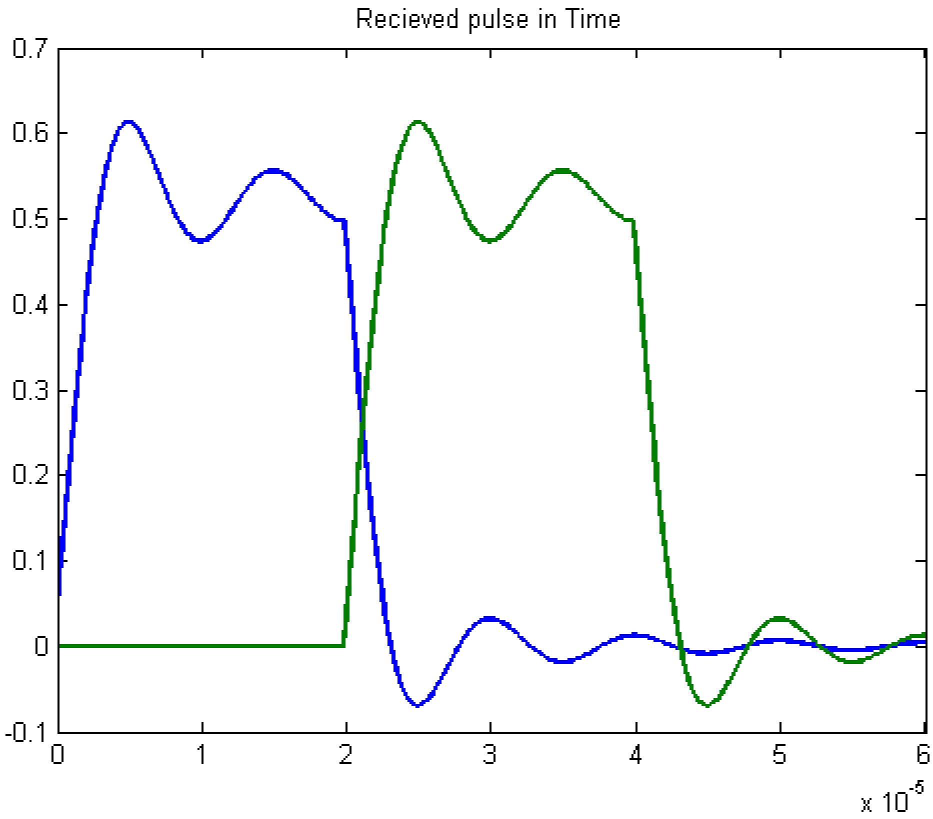

Channel in Frequency
1
0.5
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
x 10^5


Channel in Time
0.06
0.04
0.02
0
-0.02
0
0.5
1
1.5
2
x 10^-5

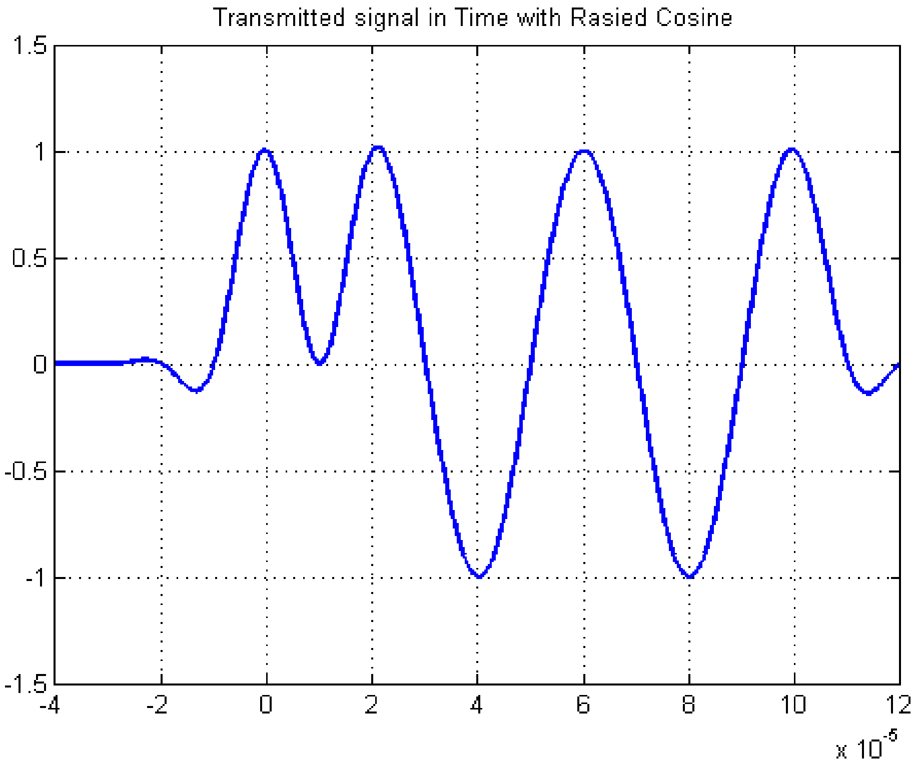

Recieved pulse in Time
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
0
0.5
1
1.5
2
2.5
3
3.5
4
x 10^-5

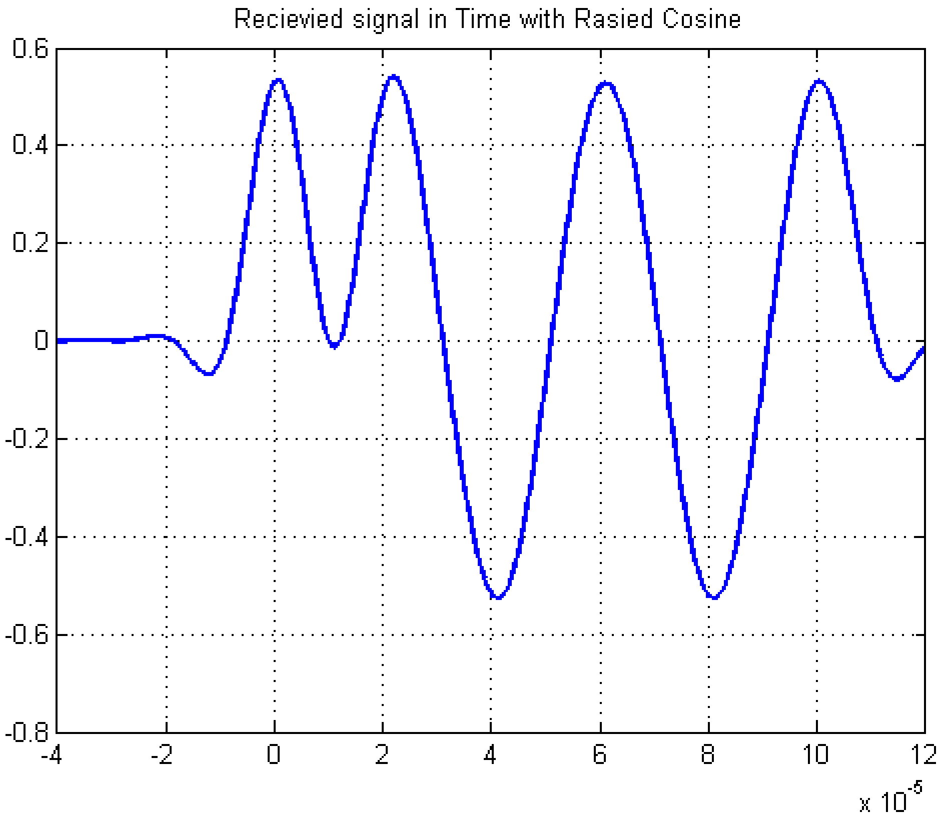

Recieved pulse in Frequency
x 10^-5
1.2
1
0.8
0.6
0.4
0.2
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
x 10^5


1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
x 10^-5

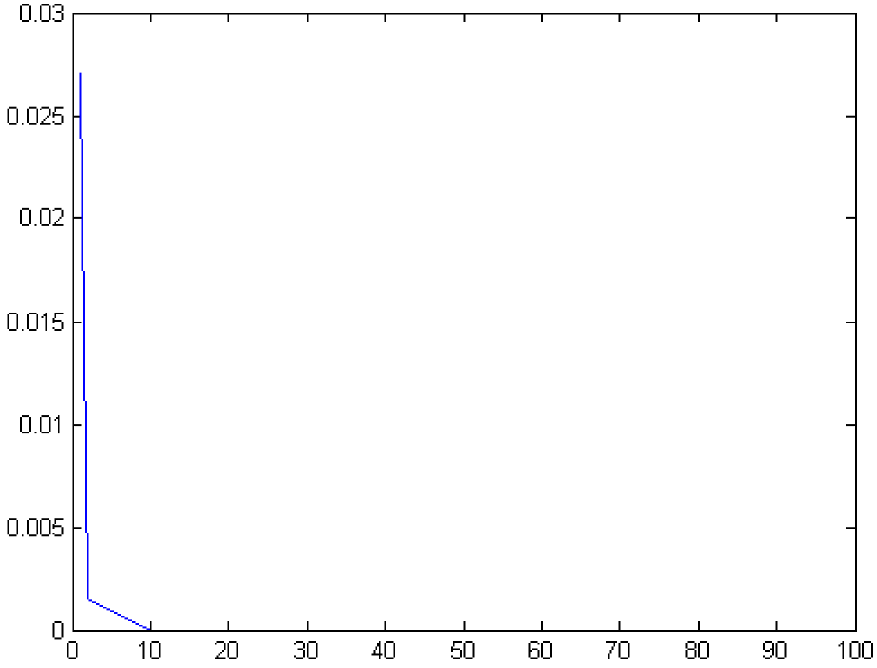


Using Raised Cosine





Recievied signal in Time with Rasied Cosine



0.03
0.025
0.02
0.015
0.01
0.005
0
0
10
20
30
40
50
60
70
80
90
100