
```

clear;
clc;
% Set the sample rate and Nyquist rate
Fs = 1e5;
Rs = 10e3;
% Define the alpha
alpha = [0, 0.5, 1];
% Generate the roll-off system with alpha=0,0.5,1
span = 6; % Filter span in symbols
sps = Fs/Rs; % sps refers to sample per symbol
h1 = rcosdesign(alpha(1), span, sps);
h2 = rcosdesign(alpha(2), span, sps);
h3 = rcosdesign(alpha(3), span, sps);
% Plot the h1, h2, h3
t = (0:length(h1)-1)/Fs;
figure(1);
plot(t, h1, t, h2, t, h3);
title('Roll-off Filter with \alpha = 0,0.5,1 in time domain
respectively');
xlabel('Time (s)');
ylabel('Amplitude');
legend('\alpha = 0', '\alpha = 0.5', '\alpha = 1');
grid on;
xlim([0, max(t)]);

% Generate the roll-off system with alpha=0,0.5,1 by fomulation
Tb = 1; % Determine the Tb
dt = Tb/Fs;
df = Tb/Rs;
t = -3*dt:3;
f = -2/Tb:df:2/Tb;
alpha = [0, 0.5, 1];
% Create the null matrix to record the calculation for each dt and df
xt = zeros(length(alpha), length(t));
Xf = zeros(length(alpha), length(f));
for n=1:length(alpha)
    % Calculate the roll-off system in frequency domain
    for k=1:length(f)
        if abs(f(k))>0.5*(1+alpha(n))/Tb
            Xf(n,k)=0;
        elseif abs(f(k))<0.5*(1-alpha(n))/Tb
            Xf(n,k)=Tb;
        else
            Xf(n,k)=0.5*Tb*(1+cos(pi*Tb/
(alpha(n)+eps)*(abs(f(k))-0.5*(1-alpha(n))/Tb)));
        end
    end
    % Calculate the roll-off system in time domain
    xt(n,:) = sinc(pi*t/Tb).*(cos(alpha(n)*pi*t/Tb))./
(1-4*alpha(n)^2*t.^2/Tb^2);
end
% Plot the h1, h2, h3 in time domain

```

```

figure(3)
plot(t, xt(1,:), t, xt(2,:), t, xt(3,:))
legend('\alpha = 0', '\alpha = 0.5', '\alpha = 1');
xlabel('Time (s)');
ylabel('Amplitude');
title('Roll-off Filter with \alpha = 0,0.5,1 in time domain
    respectively');
% xlim([-0.5, 0.5])
ylim([-0.4, 1.2])
grid on
% Plot the h1, h2, h3 in frequency domain
figure(4)
plot(f, Xf(1,:), f, Xf(2,:), f, Xf(3,:))
legend('\alpha = 0', '\alpha = 0.5', '\alpha = 1');
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title('Roll-off Filter with \alpha = 0,0.5,1 in frequency domain
    respectively');
ylim([0, 1.2])
grid on

```

Published with MATLAB® R2021a