**MATLAB CODE:**

clc;

clear all;

close all;

p=1; %fundamental overtone

l=0.001; % length of crystal

E=7.9\*(10^10); % young’s modulus of crystal

dc=2650; %density of crystal

X=sqrt(E/dc);

f=(p/(2\*l))\*X; %frequency of the sound wave

disp('frequency of the sound wave');

disp(f); % displaying freq of us wave

fs=2\*f; %sampling frequency

x=0:1/fs:0.0002;

I=3.47\*(10^4);

d=997;

c=1480;

om=2\*pi\*f;

a=2\*I/(d\*c\*[(om)^2]);

A=sqrt(a);

disp('Amplitude of the sound wave');

disp(A);

y=A\*sin(2\*pi\*f\*x);

plot(x,y);

set(gca,'FontSize',7);

title('Ultrasonic wave');

xlabel('time','FontSize',7,"FontWeight","bold");

ylabel('Amplitude','FontSize',7,"FontWeight","bold");

%sampled signal plotting

figure(2);

hold on

stem(y);

title('Sampled Signal');

xlabel('time');

ylabel('amplitude');

hold off

% Quantization

m = input('Enter the number of bits per sample: '); ;%number of bits per sample

L = 2^m; %no of levels of quantization

xmax = 2;

xmin = -2;

del = (xmax - xmin)/L; %defining del

part = xmin:del:xmax; % definition of decision lines

codebook = xmin - del/2 : del : xmax + del/2; % definition of representation levels

[index, quants] = quantiz(y, part, codebook); % gives rounded off values of the samples %quantiz is inbuilt function which is used to quantize the signal

figure(3);

hold on

stem(quants); %plotting of quantized signal

title('Quantized Signal');

xlabel('time','FontSize',7,"FontWeight","bold");

ylabel('Amplitude','FontSize',7,"FontWeight","bold");

hold off;

% Normalization

% to convert 1 to n as 0 to n-1 indicies

l1 = length(index); % finding length of indicies

for i=1:l1

if index(i)~=0

index(i) = index(i)-1;

end

end

% to convert the end representation levels

l2 = length(quants); % finding length of quants

for i=1:l2

if quants(i)==xmin-del/2

quants(i) = xmin+del/2;

end

if quants(i)==xmax+del/2

quants(i) = xmax-del/2;

end

end

% Encoding

code = de2bi(index, 'left-msb'); %decimal to binary conversion of indices

k = 1;

for i=1:l1

for j=1:m

coded(k) = code(i, j); % to convert column vector to row vector

j = j+1;

k = k+1;

end

i = i+1;

end

% Plot Digital Signal

figure(4);

hold on

stairs(coded);

axis([0 200 -2 2])

title('Digital Signal');

xlabel('time','FontSize',7,"FontWeight","bold");

ylabel('Amplitude','FontSize',7,"FontWeight","bold");

hold off

% Demodulation

code1 = reshape(coded, m, (length(coded)/m)); % reshaping signal using reshape command

index1 = bi2de(code1, 'left-msb'); %converting from decimal to binary

resignal = del\*index+xmin+del/2;

figure(5);

hold on

plot(resignal); % plotting demodulated us signal

title('Demodulated Signal');

xlabel('time','FontSize',7,"FontWeight","bold");

ylabel('Amplitude','FontSize',7,"FontWeight","bold");

hold off