PCM AND DPCM

EXP:11

AIM:

To implement PCM (Pulse Code Modulation) in Matlab and to study the working of DPCM

CODE:

```
clc;
clear all;
close all;
n=5;%no of sine waves
t=0:0.0001:(n*(0.5));
fm=2; %message frequency
am=8;%message amplitude
m=am*sin(2*pi*fm*t);
subplot(321)
plot(t,m);
title('message signal');
xlabel('time(s)');
ylabel('amplitude');
%sampling
r=20; %no of samples
fs=2*fm %sampling frequency
ts=(1/fs);
tsr=0:(1/(fs*r)):(n*ts);
sm=am*sin(2*pi*fs*tsr);
subplot(322);
stem(tsr,sm,'r');
title('sampled signal');
xlabel('time(s)');
ylabel('amplitude');

%quantisation
l=4;
stepsize=8;
del=(2*8)/l;
q=[];f=1;
for tsr=0:(1/(fs*r)):(n*ts)
    if(sm(f)<-6)
        q(f)=-6;
    elseif(sm(f)<-2&& sm(f)>=-6)
        q(f)=-2;
    elseif(sm(f)<2&& sm(f)>=-2)
        q(f)=-2;
    elseif (sm(f)<6&& sm(f)>=2)
```

```matlab
        q(f)=2;
    else
         q(f)=6;
    end
    f=f+1;
end
    tsr=0:(1/(fs*r)):(n*ts)
    subplot(323);
    plot(tsr,q);
    title('quantised signal');
xlabel('time(s)');
ylabel('amplitude');

  %reconstruction of the signal at the receiver
  [num,den]=butter(1,1.48*fm/fs);
 recon=filter(num,den,sm);
 subplot(325);
 plot(tsr,recon);
 title('Reconstructed signal in the receiver');
xlabel('time(s)');
ylabel('amplitude');

 %pcm wave
 f=1;c=1;
 e=[];
 for tsr=0:(1/(fs*r)):(n*ts)
    if(q(c)==-6 | q(c)==6)
        e(f)=1; e(f+1)=1;e(f+2)=0;
    elseif(q(c)==-2 | q(c)==2)
        e(f)=0; e(f+1)=1;e(f+2)=1;
    end
    f=f+3;
    c=c+1;
 end

 subplot(324);
 plot(e);
 title('pcm');
xlabel('time(s)');
ylabel('amplitude');
```
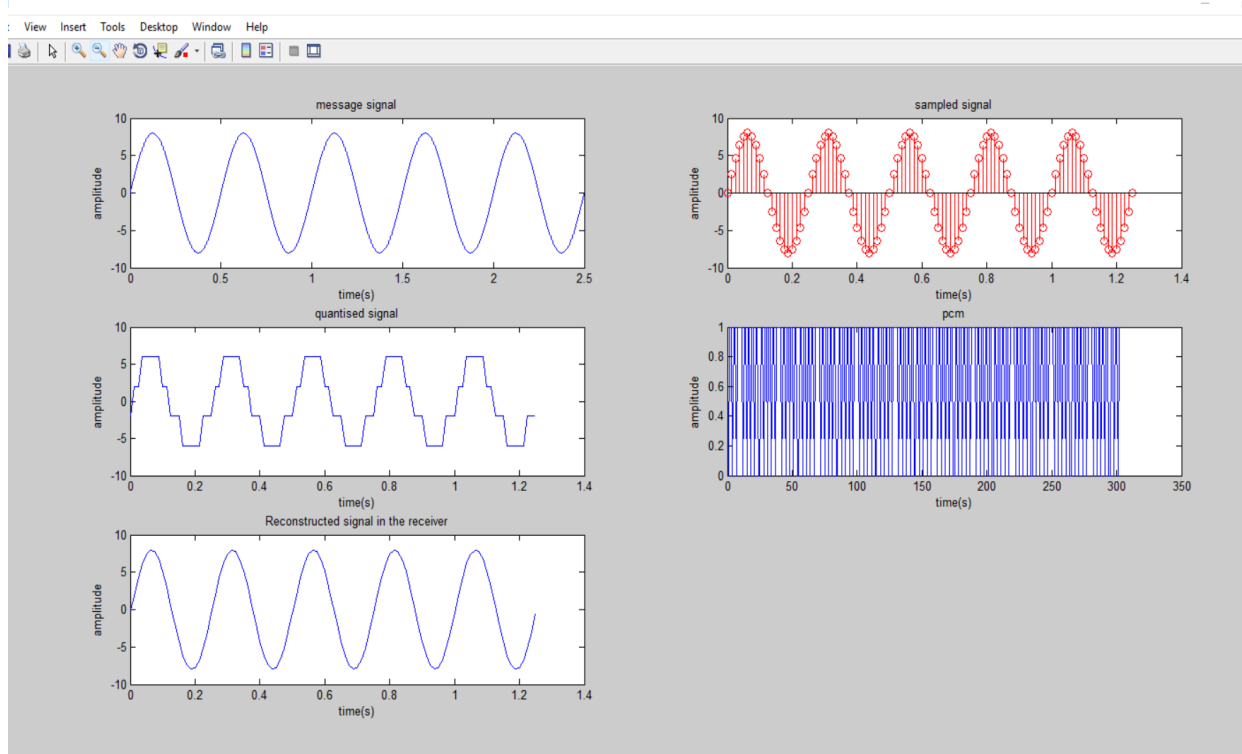
OUTPUT:

RESULT:

Thus the PCM(Pulse Code Modulation) is implemented in MATLAB and DPCM is been studied.