Year 2020-21

Digital Communication
E-Laboratory and practicals
B.Tech. II (CSE), semester –III

Department of Electronics Engineering
Sardar Vallabhbhai National Institute of Technology, Ichchhanath, Surat-395 007, Gujarat

Experiment-9

Objective- Study of pulse code modulation (PCM) and demodulation technique.

Pulse Code Modulation (PCM)

- PCM is a technique, which is used to convert an analog signal into digital signal.

- PCM is a preferred method of communication within the public switched telephone network (PSTN)

- A PCM stream is determined by two following steps:

Sampling rate- which is the number of times per second that samples are taken.
Bit depth- which determines the number of possible digital values that can be used to represent each sample.

Hence, the output of a PCM resembles a binary sequence.

Why digital transmission??

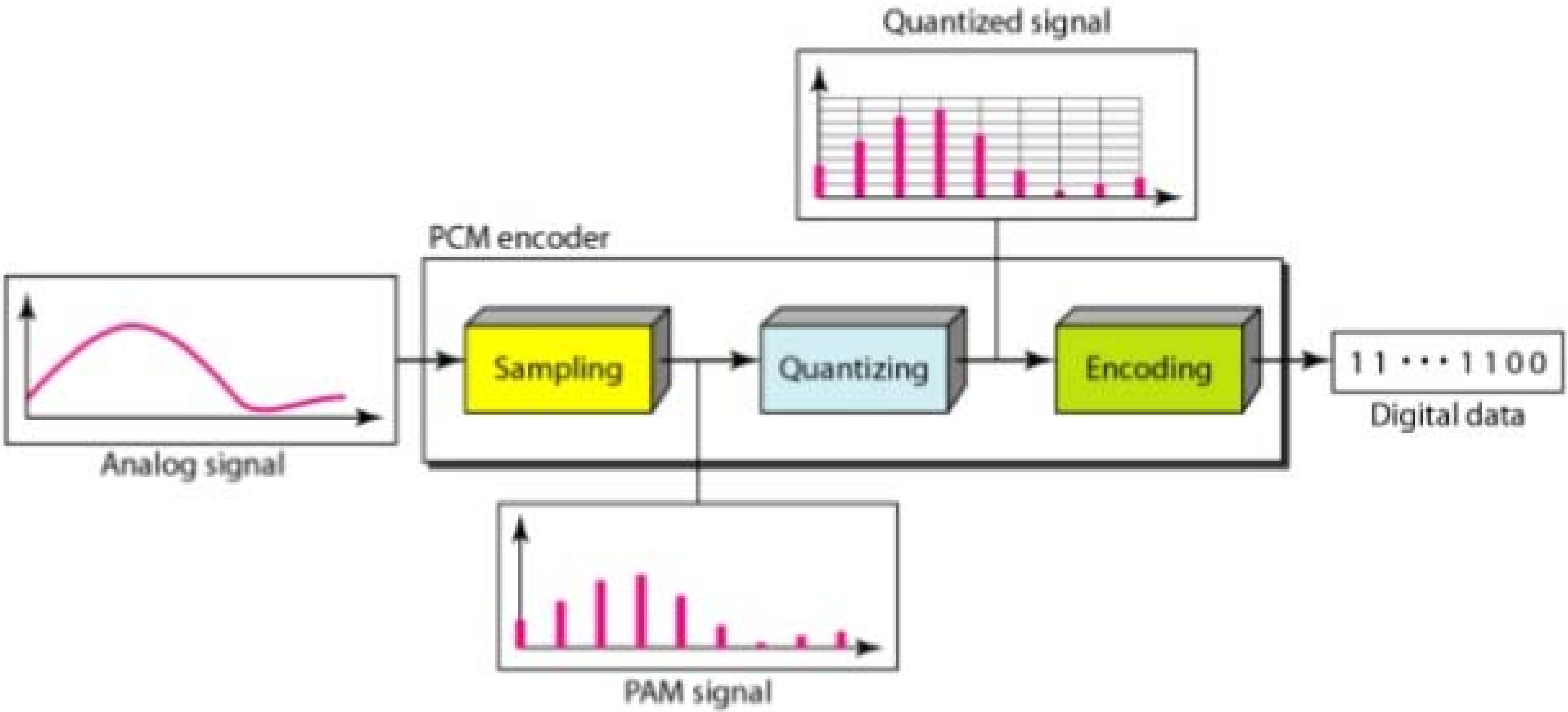
- Less susceptible to interference cause by noise due to discrete level.

- Easy to detect errors due to discrete level.

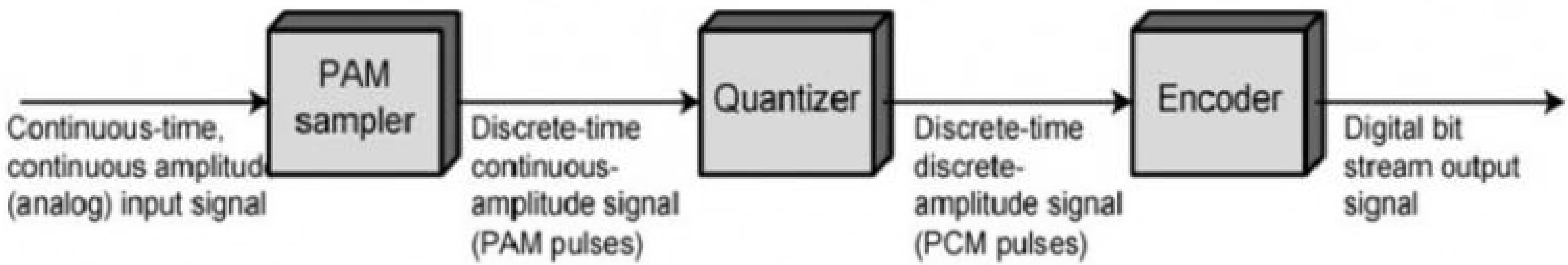
- Easy to encrypt(Higher security)

- Simpler to store digital data

BLOCK DIAGRAM OF PCM

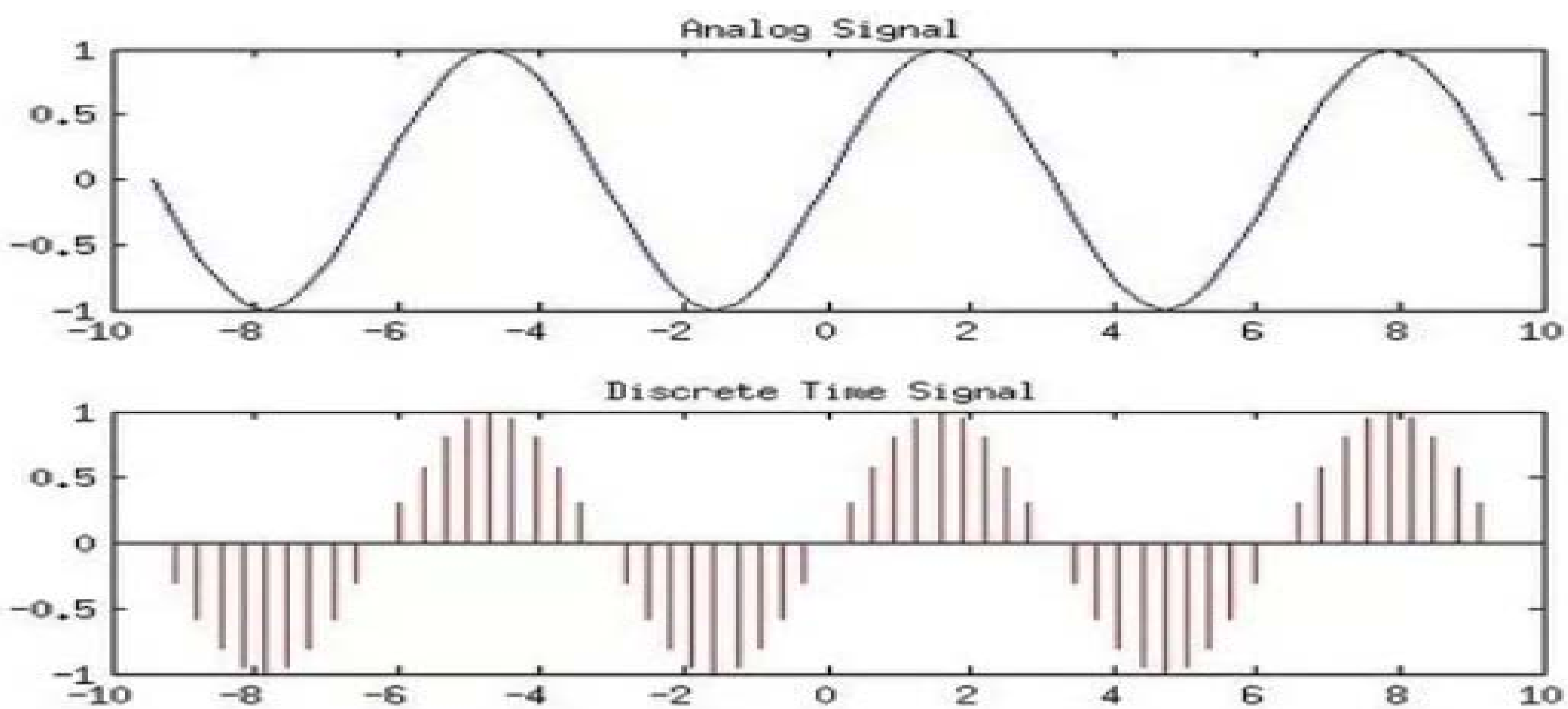


Block diagram of PCM



Sampling

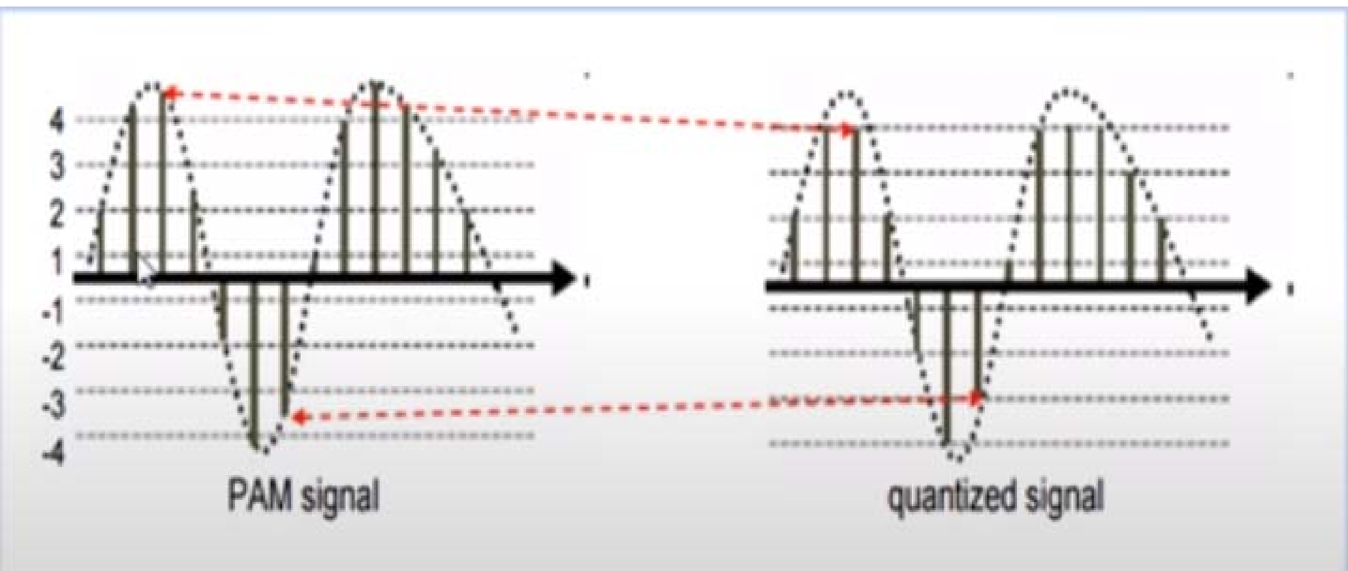
- Sampler extract samples of a continuous signal.
- Sampler produces samples that are equivalent to the instantaneous value of the continuous signal at the specified various points.
- The Sampling process generates flat- top Pulse Amplitude Modulated (PAM) signal.



Quantization

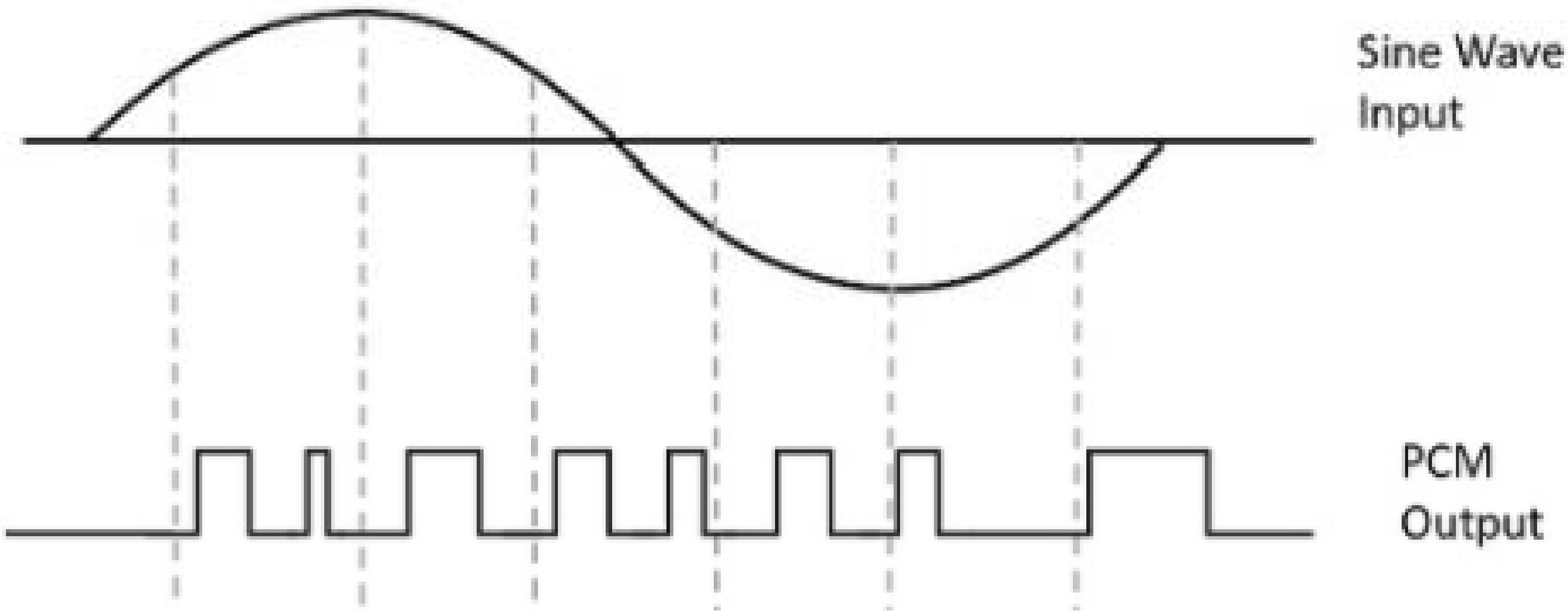
- Quantization is done by dividing the range of possible values of the analog samples into some different levels and assigning the center value of each level to any sample in the quantization interval.

- Quantization approximates the analog sample values with the nearest quantization values.



Level	Code word
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

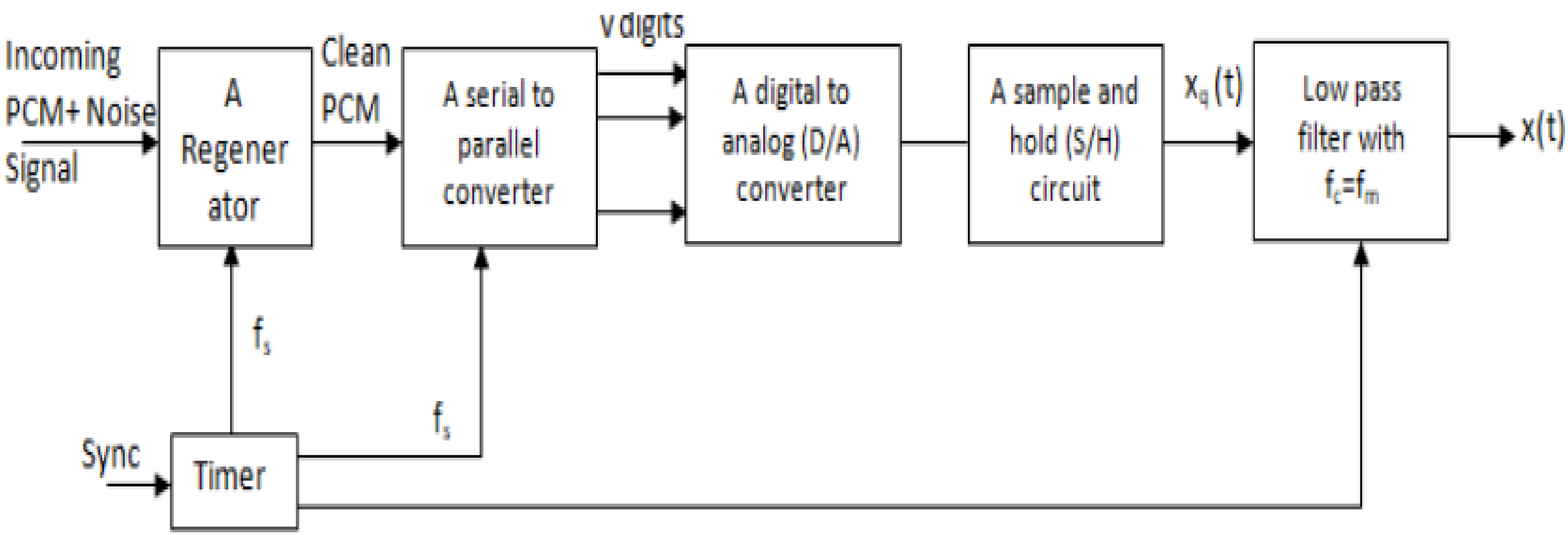
A/D output = n bits per sample (quantization level $M=2^n$)



- PCM produces a series of numbers or digits instead of a pulse train.

- Each one of these digits, in binary code, represent the approximate amplitude of the signal sample at that instant.

PCM Receiver



Conclusions

- In PCM transmitter , the signal x(t) is first passed through the low-pass filter of cut-off frequency f_m Hz .
- This low-pass filter blocks all the frequency components above f_m Hz. This means that now the signal x(t) is band-limited to f_m Hz .
- The sample and hold circuit then samples this signal at the rate of f_s.
- Sampling frequency f_s is selected sufficiently above Nyquist rate to avoid aliasing
- The output from sample and hold circuit is denoted by $x(nT_s)$.
- This signal $x(nT_s)$ is discrete in time and continuous in amplitude.
- A q-level quantizer compares input $x(nT_s)$ with its fixed digital levels.
- Quantized signal is then encoded in PCM output using encoder.

APPLICATION

- In compact disk
- Digital telephony
- Digital audio applications

MATLAB Code and Simulation Results

1. Sampling

```
n=input('Enter n value for n-bit PCM system :  ');
n1=input('Enter number of samples in a period : ');
L=2^n;
% % Signal Generation
% x=0:1/100:4*pi;
% y=8*sin(x);                          % Amplitude Of signal is 8v
% subplot(2,2,1);
% plot(x,y);grid on;
% Sampling Operation
x=0:2*pi/n1:4*pi;               % n1 nuber of samples have tobe selected
s=8*sin(x);
subplot(3,1,1);
plot(s);
title('Analog Signal');
ylabel('Amplitude--->');
xlabel('Time--->');
subplot(3,1,2);
stem(s);grid on;  title('Sampled Sinal');  ylabel('Amplitude--->');  xlabel('Time--->');
```

2. Quantization

```
% Quantization Process
vmax=8;
vmin=-vmax;
del=(vmax-vmin)/L;                      % level are between vmin and vmax with difference of del
part=vmin:del:vmax;
code=vmin-(del/2):del:vmax+(del/2);     % Contain Quantized values
[ind,q]=quantiz(s,part,code);           % Quantization process

l1=length(ind);
l2=length(q);

for i=1:l1                              % To make index as binary decimal so started from 0 to N
   if(ind(i)~=0)
      ind(i)=ind(i)-1;
   end
   i=i+1;
end
 for i=1:l2                             % To make quantize value in between the levels
    if(q(i)==vmin-(del/2))
        q(i)=vmin+(del/2);
    end
end
subplot(3,1,3);                         % Display the Quantize values
stem(q);grid on;
title('Quantized Signal');
ylabel('Amplitude--->');
xlabel('Time--->');
```

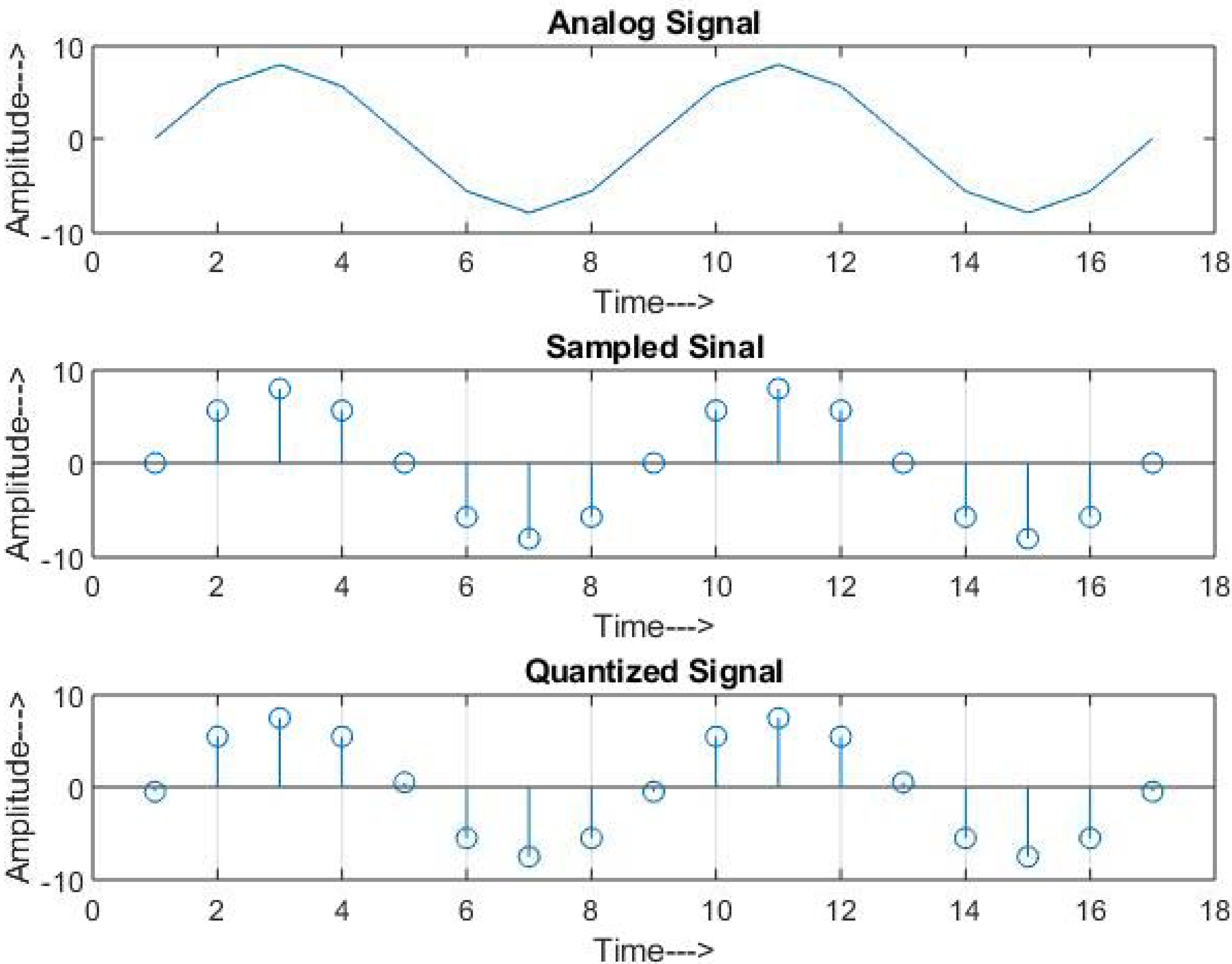
3. Encoding

```
% Encoding Process
figure
code=de2bi(ind,'left-msb');                    % Cnvert the decimal to binary
k=1;
for i=1:l1
    for j=1:n
        coded(k)=code(i,j);                    % convert code matrix to a coded row vector
        j=j+1;
        k=k+1;
    end
    i=i+1;
end
subplot(2,1,1); grid on;
stairs(coded);                                 % Display the encoded signal
axis([0 100 -2 3]);  title('Encoded Signal');
ylabel('Amplitude--->');
xlabel('Time--->');
```

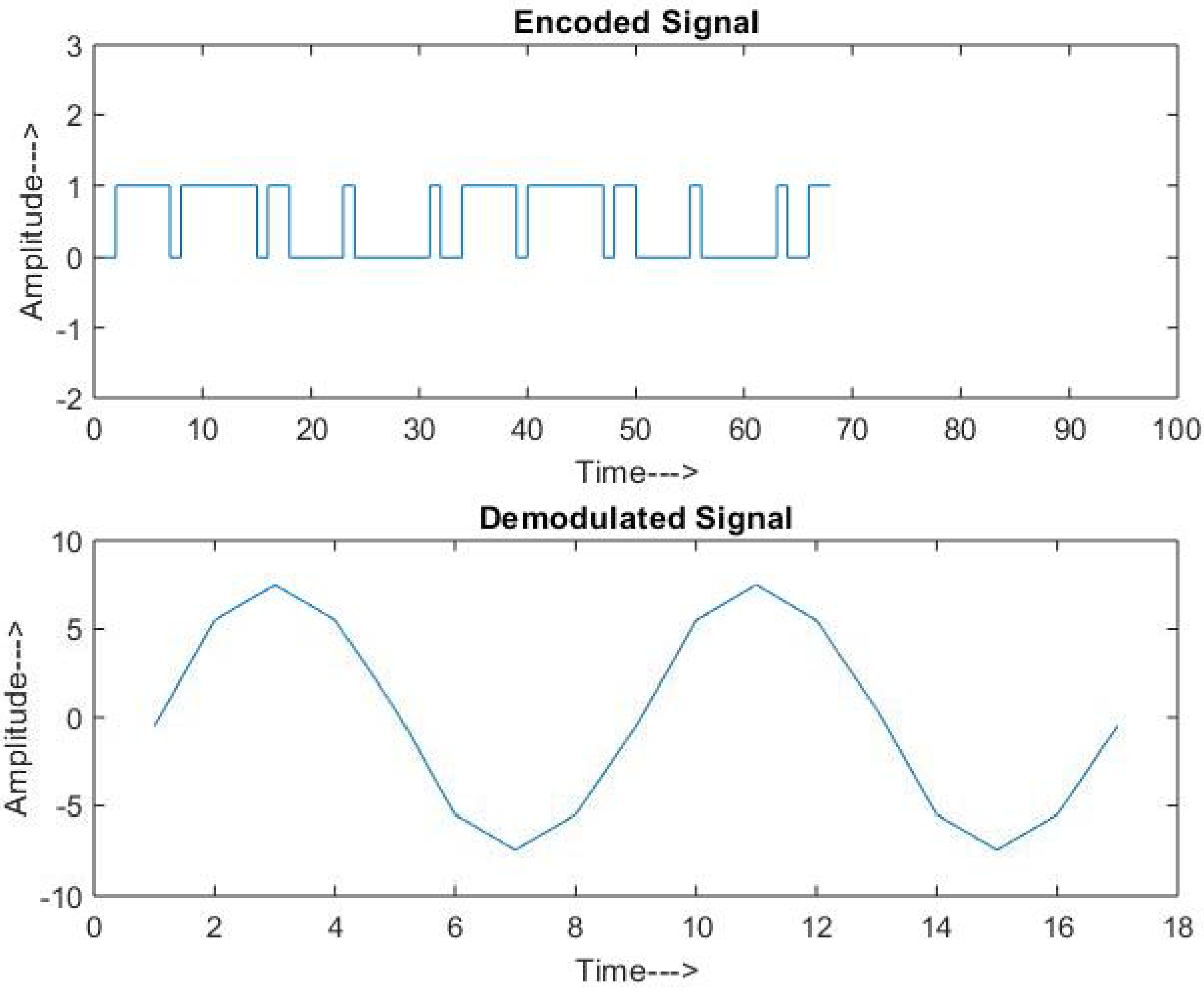
4. Demodulation

```
%   Demodulation Of PCM signal

qunt=reshape(coded,n,length(coded)/n);
index=bi2de(qunt','left-msb');                    % Getback the index in decimal form
q=del*index+vmin+(del/2);                        % getback Quantized values
subplot(2,1,2); grid on;
plot(q);                                                      % Plot Demodulated signal
title('Demodulated Signal');
ylabel('Amplitude--->');
xlabel('Time--->');
```



Analog Signal
Amplitude--->
10
0
-10
0 2 4 6 8 10 12 14 16 18
Time--->
Sampled Sinal
Amplitude--->
10
0
-10
0 2 4 6 8 10 12 14 16 18
Time--->
Quantized Signal
Amplitude--->
10
0
-10
0 2 4 6 8 10 12 14 16 18
Time--->



Encoded Signal
Amplitude--->
Time--->
3
2
1
0
-1
-2
0
10
20
30
40
50
60
70
80
90
100
Demodulated Signal
Amplitude--->
Time--->
10
5
0
-5
-10
0
2
4
6
8
10
12
14
16
18

Any
questions?