Indian Institute of
Information Technology,
Nagpur

ECL-320
Final Report

Submitted By :
Sarthak Babra (BT19ECE028)

Semester 6
Electronics and Communication Engineering Dept.

Submitted To :
Dr. Rashmi Pandhare
Course Instructor

INDEX

Indian Institute of
Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 1

Author:
Sarthak Babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

To study sampling Theorem and analyze the process of Reconstruction of the input signal.

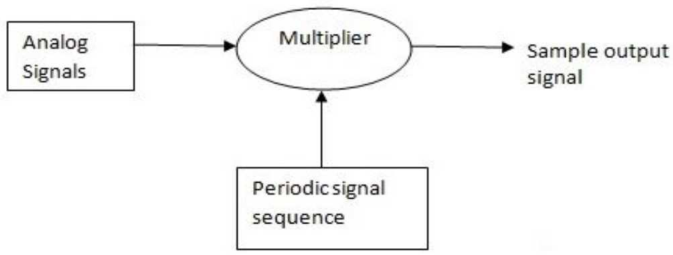
Apparatus or Equipment's Required

1. PCM trainer kit
2. DSO (0-20)MHZ
3. CRO Probes
4. Connecting Wires

Theory

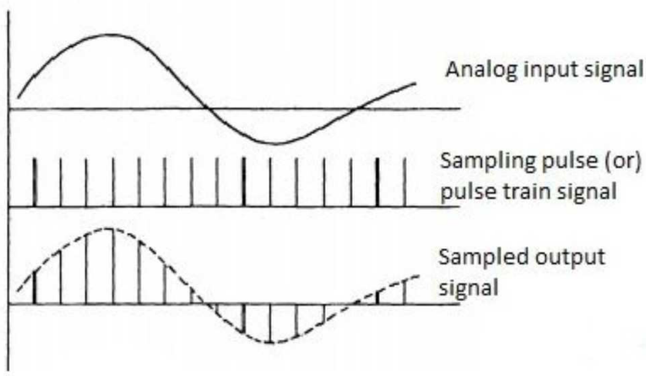
Sampling Theorem

In sampling theorem, the input signal is in an analog form of signal and the second input signal is a sampling signal, which is a pulse train signal and each pulse is equidistance with a period of "Ts". This sampling signal frequency should be more than twice of the input analog signal frequency. If this condition satisfies, analog signal perfectly represented in discrete form else analog signal may be losing its amplitude values for certain time intervals.



Sampling theorem states that "continues form of a time-variant signal can be represented in the discrete form of a signal with help of samples and the sampled signal can be recovered to original form when the sampling signal frequency Fs having the greater frequency value than or equal to the input signal frequency Fm.

$$\mathbf{Fs \geq 2Fm}$$

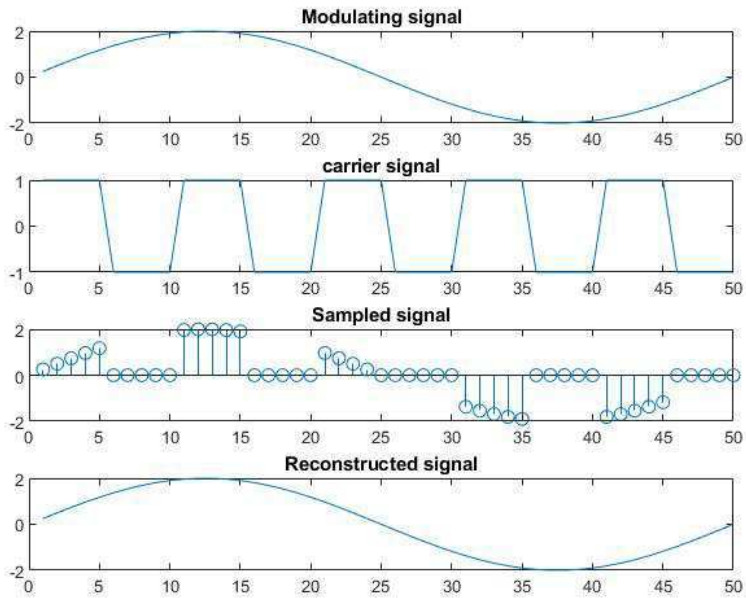


Code:

```
t=1:50;

%creating modulation signal
sig=2*sin(2*pi*0.02*t);
subplot(4,1,1);
plot(t,sig);
title("Modulating signal")

%creating carrier signal
x=sin(2*pi*0.1*t);
for i=1:50
    if x(i)<0
        x(i)=-1;
    else if x(i)==0
        x(i)=0;
        else
            x(i)=1;
        end
    end
end
subplot(4,1,2);
plot(t,x);
title("carrier signal");

%creating sample signal
for i=1:50
    if x(i)==-1
        sig(i)=0;
    end
end
subplot(4,1,3)
stem(t,sig);
title("Sampled signal")

%reconstruction the signal
for i=1:50
```

```
    if sig(i)==0

        sig(i)=2*sin(2*pi*0.02*i);

    end

end

subplot(4,1,4);

plot(t,sig);

title("Reconstructed signal")
```

Output:



Conclusion

We have successfully sampled an analog signal using sampling theorem and reconstructed the input signal from the sample one using Matlab.

Indian Institute of
Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 2

Author:
Sarthak Babra (BT19ECE028)

Submitted to:
Dr. Rashmi Pandhare

AIM:

Pulse Code Modulation.

Apparatus or Equipment's Required

1. PCM trainer kit
2. DSO (0-20)MHZ
3. CRO Probes
4. Connecting Wires

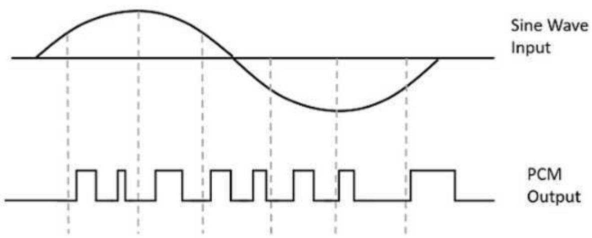
Theory

Pulse Code Modulation

Modulation is the process of varying one or more parameters of a carrier signal in accordance with the instantaneous values of the message signal.

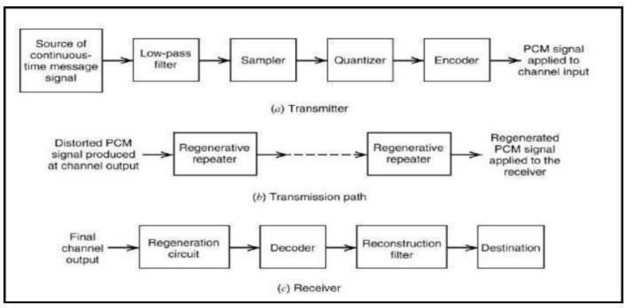
The message signal is the signal which is being transmitted for communication and the carrier signal is a high frequency signal which has no data, but is used for long distance transmission.

There are many modulation techniques, which are classified according to the type of modulation employed. Of them all, the digital modulation technique used is Pulse Code Modulation A signal is pulse code modulated to convert its analog information into a binary sequence, i.e., 1s and 0s. The output of a PCM will resemble a binary sequence. The following figure shows an example of PCM output with respect to instantaneous values of a given sine wave.



Instead of a pulse train, PCM produces a series of numbers or digits, and hence this process is called as digital. Each one of these digits, though in binary code, represent the approximate amplitude of the signal sample at that instant.

In Pulse Code Modulation, the message signal is represented by a sequence of coded pulses. This message signal is achieved by representing the signal in discrete form in both time and amplitude.

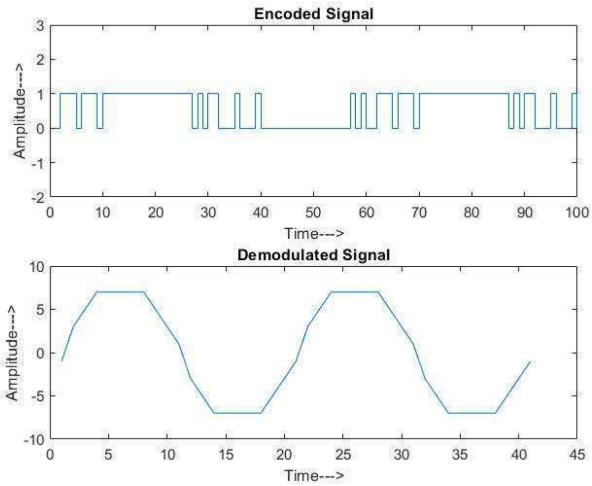


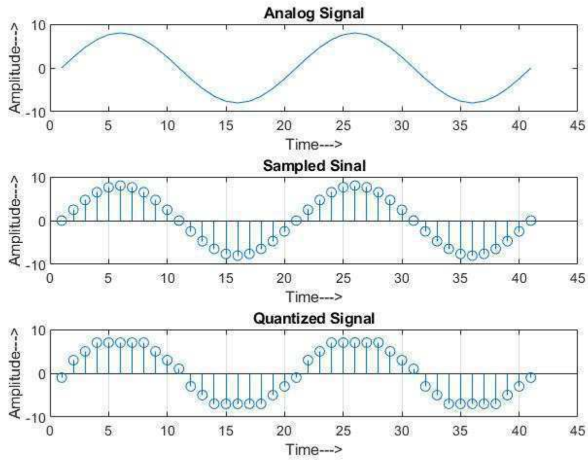
Code:

```
clc;
close all;
clear all;
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--->');
 %  Quantization Process
 vmax=8;
 vmin=-vmax;
 del=(vmax-vmin)/L;
 part=vmin:del:vmax;                                  % level are between vmin and vmax with difference of del
 code=vmin-(del/2):del:vmax+(del/2);         % Contaion Quantized valuses
 [ind,q]=quantiz(s,part,code);                     % Quantization process
                                                                      % ind contain index number and q contain quantized  values
 l1=length(ind);
 l2=length(q);

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

 %  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--->');

%   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--->');
```

Output:





Conclusion

We have successfully sampled an analog signal, its Pulse Code Modulation and the Reconstruction of Signal.

Indian Institute of
Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 3

Author:
Sarthak babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

To study sampling oanalog signal, its Delta Modulation and the Reconstruction of Signal.

Apparatus or Equipment's Required

1. PCM trainer kit
2. DSO (0-20)MHZ
3. CRO Probes
4. Connecting Wires

Theory

Delta modulation

A delta modulation (DM or -modulation) is an analog-to-digital and digital-to analog signal conversion technique used for transmission of voice information where quality is not of primary importance. DM is the simplest form of differential pulse code modulation (DPCM) where the difference between successive samples is encoded into n-bit data streams. In delta modulation, the transmitted data are reduced to a 1-bit data stream. Its main features are:

1. The analog signal is approximated with a series of segments.
2. Each segment of the approximated signal is compared to the preceding bits and the successive bits are determined by this comparison.
3. Only the change of information is sent, that is, only an increase or decrease of the signal amplitude from the previous sample is sent whereas a no-change condition causes the modulated signal to remain at the same 0 or 1 state of the previous sample.

Derived forms of delta modulation are continuously variable slope delta modulation, delta-sigma modulation, and differential modulation. Differential pulse-code modulation is the superset of DM.

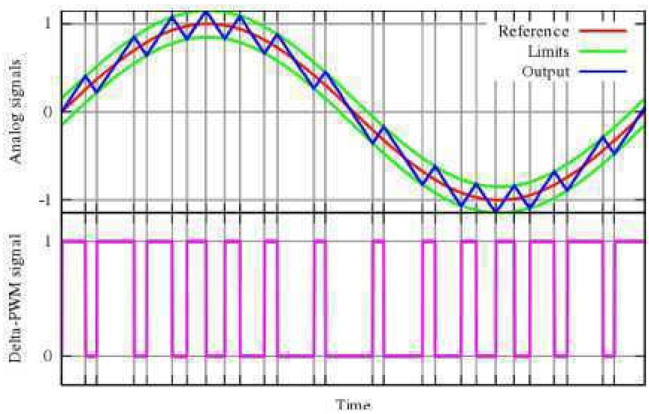
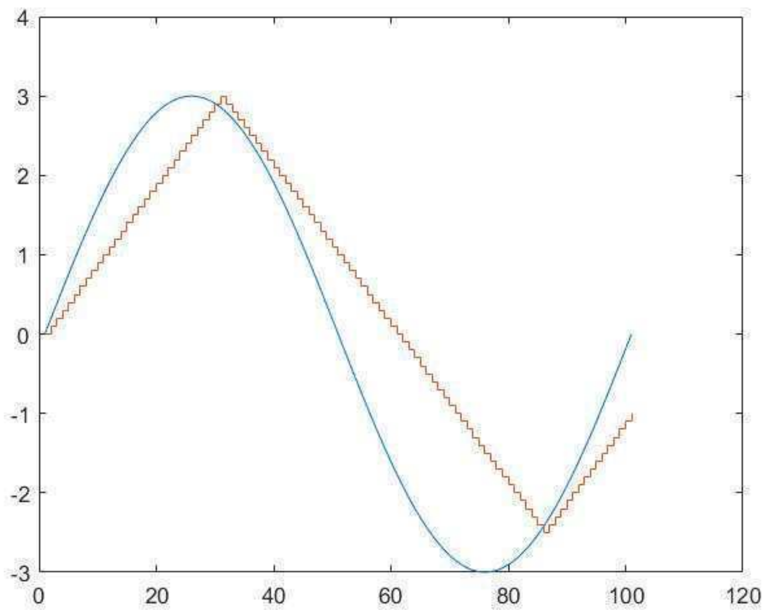


Figure 1: Principle of the delta PWM. The output signal (blue) is compared with the limits (green). The limits (green) correspond to the reference signal (red), offset by a given value. Every time the output signal reaches one of the limits, the PWM signal changes state

Code:

```
function [y MSE]=Delta_Modulation(del, A)
%del=step size
%A=amplitude of signal
%y=output binary sequence
%Vary del value and check when MSE is least

t=0:2*pi/100:2*pi;
x=A*sin(t);
plot(x)
hold on
y=[0];
xr=0;
for i=1:length(x)-1
    if xr(i)<=x(i)
        d=1;
        xr(i+1)=xr(i)+del;
    else
        d=0;
        xr(i+1)=xr(i)-del;
    end
    y=[y d];

end
stairs(xr)
hold off
MSE=sum((x-xr).^2)/length(x);
end
```

Output:



Conclusion

We have successfully sampled an analog signal, its Delta Modulation and the Reconstruction of Signal.

INDIAN INSTITUTE OF
INFORMATION TECHNOLOGY

DIGITAL COMMUNICATION

ECE-320

ELECTRONICS AND COMMUNICATION DEPARTMENT

Experiment 4

Author:
Sarthak Babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

Perform a Amplitude Shift Keying Modulation Using Matlab.

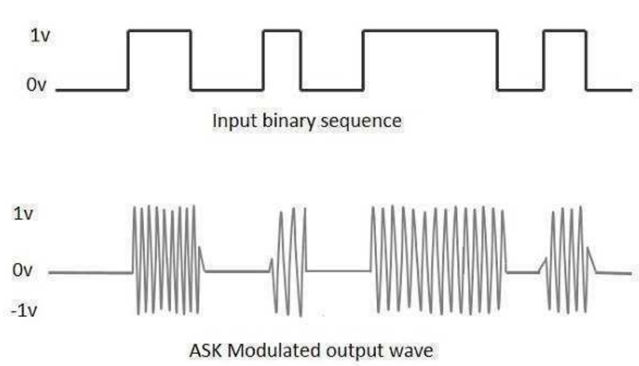
Theory

Amplitude Shift Keying

Amplitude Shift Keying (ASK) is a type of Amplitude Modulation which represents the binary data in the form of variations in the amplitude of a signal.

Any modulated signal has a high frequency carrier. The binary signal when ASK modulated, gives a **zero** value for **Low** input while it gives the carrier output for **High** input.

The following figure represents ASK modulated waveform along with its input.

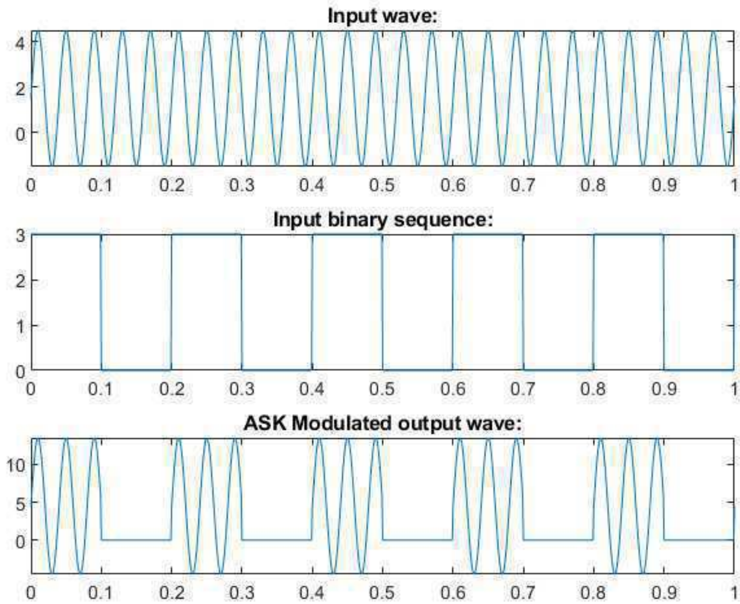


To find the process of obtaining this ASK modulated wave, let us learn about the working of the ASK modulator.

Code:

```
clear all
close all
clc

f1 = 25;
f2 = 5;
a = 3;
t = 0:0.001:1;

x = a * sin(2*pi*f1*t) + (a/2);
u = (a/2) * square(2*pi*f2*t) + (a/2);
v = x.*u;

subplot(3, 1, 1);
plot(t,x);
title('Input wave:');

subplot(3, 1, 2);
plot(t,u);
title('Input binary sequence:');

subplot(3, 1, 3);
plot(t,v);
title('ASK Modulated output wave:');
```

Output:



Conclusion

We have successfully performed Amplitude Shift Keying modulation on a analog signal and plotted the outputs.

Indian Institute of
Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 5

Author:
Sarthak Babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

Perform a perform PSK and FSK Modulation using Matlab.

Theory:

Phase Shift Keying (FSK):

Phase Shift Keying (PSK) is the digital modulation technique in which the phase of the carrier signal is changed by varying the sine and cosine inputs at a particular time. PSK technique is widely used for wireless LANs, bio-metric, contactless operations, along with RFID and Bluetooth communications.PSK is of two types, depending upon the phases the signal gets shifted..

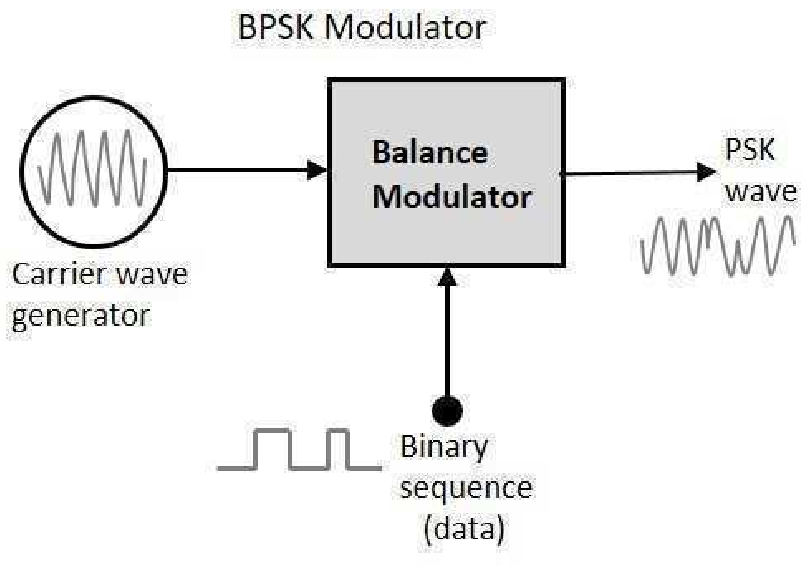
1)Binary Phase Shift Keying (BPSK) This is also called as 2-phase PSK or Phase Reversal Keying. In this technique, the sine wave carrier takes two phase reversals such as 0° and 180°.BPSK is basically a Double Side Band Suppressed Carrier (DSBSC) modulation scheme, for message being the digital information.

2)Quadrature Phase Shift Keying (QPSK) This is the phase shift keying technique, in which the sine wave carrier takes four phase reversals such as 0°, 90°, 180°, and 270°.

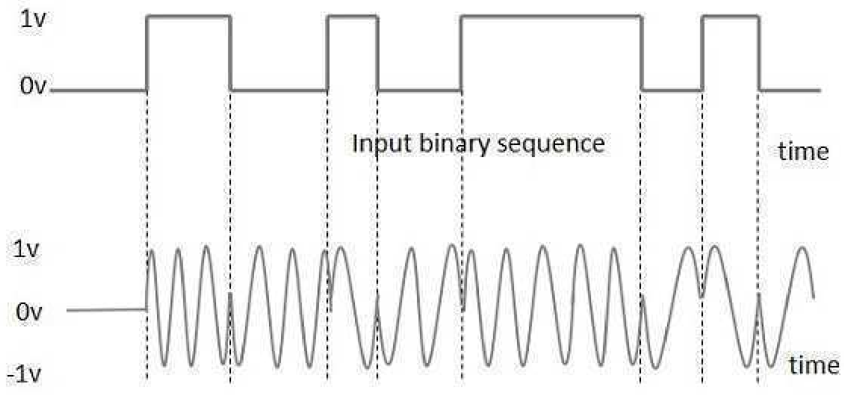
If this kind of techniques are further extended, PSK can be done by eight or sixteen values also, depending upon the requirement.

BPSK Modulator:

The block diagram of Binary Phase Shift Keying consists of the balance modulator which has the carrier sine wave as one input and the binary sequence as the other input. Following is the diagrammatic representation



The modulation of BPSK is done using a balance modulator, which multiplies the two signals applied at the input. For a zero binary input, the phase will be 0° and for a high input,the phase reversal is of 180°.
The following figure represents PSK modulated waveform along with its input.



BPSK Modulated output wave

The output sine wave of the modulator will be the direct input carrier or the inverted 180° phaseshifted input carrier, which is a function of the data signal.

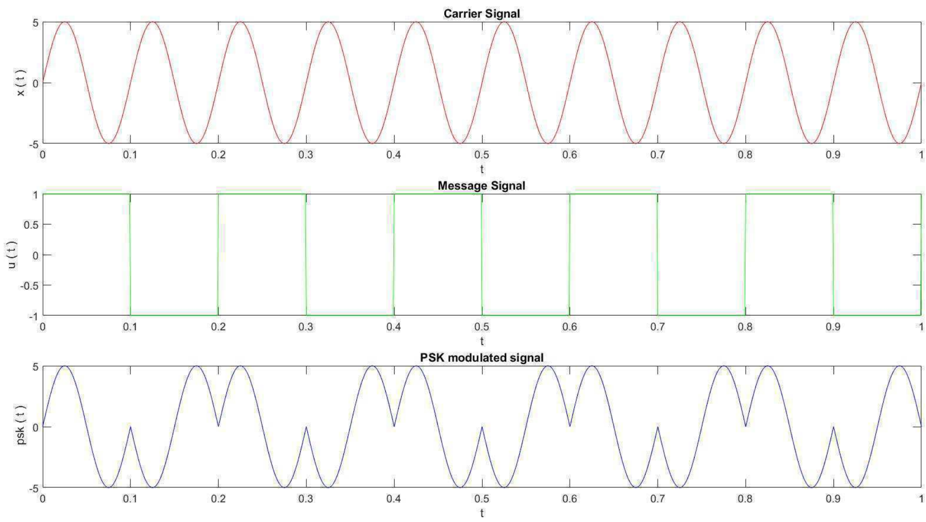
Procedure

•Firstly, we have to initialize f1 and f2 i.e.- the frequencies of carrier signal and modulating signal.
• We then define the amplitude variable, A and the time variable, t.
• Then we represent the carrier signal, x.
• After that we define the square modulating signal, u.
• Finally we have calculated the PSK modulated signal, y.

Matlab Code:

```
clc ;
clear all ;
close all ;
A = 5;
t = 0:0.001:1;
f1 = 10;
f2 = 5;
x = A .* sin (2* pi * f1 * t ) ; % carrier
subplot (3 ,1 ,1) ;
plot (t ,x , 'r') ;
title (" Carrier Signal ") ;
xlabel (" t ") ;
ylabel (" x ( t ) ") ;
u = square (2* pi * f2 * t ) ; % message signal
subplot (3 ,1 ,2) ;
plot (t ,u , 'g') ;
title (" Message Signal ") ;
xlabel (" t ") ;
ylabel (" u ( t ) ") ;
y = x .* u ;
subplot (3 ,1 ,3) ;
plot (t ,y , 'b') ;
title (" PSK modulated signal ") ;
xlabel (" t ") ;
ylabel (" psk ( t )");
```

Output:



Frequency Shift Keying (FSK):

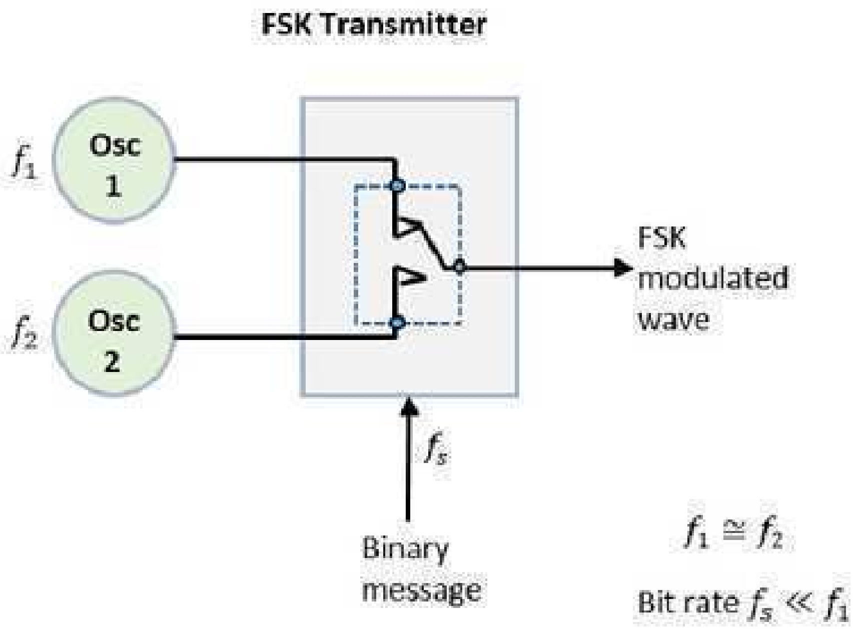
Frequency Shift Keying (FSK) is the digital modulation technique in which the frequency of the carrier signal varies according to the digital signal changes. FSK is a scheme of frequency modulation.

The output of a FSK modulated wave is high in frequency for a binary High input and is low in frequency for a binary Low input. The binary **1s** and **0s** are called Mark and Space frequencies.

The following image is the diagrammatic representation of FSK modulated waveform along with its input.

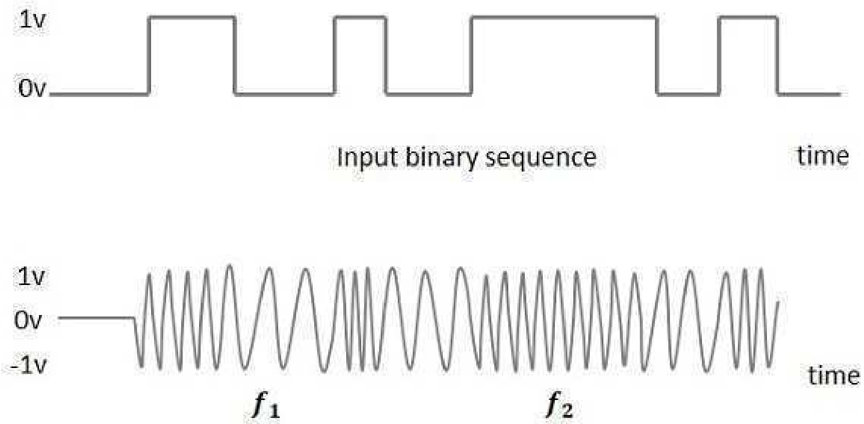
FSK Modulator:

The FSK modulator block diagram comprises of two oscillators with a clock and the input binary sequence. Following is its block diagram.



The two oscillators, producing a higher and a lower frequency signals, are connected to a switch along with an internal clock. To avoid the abrupt phase discontinuities of the output waveform during the transmission of the message, a clock is applied to both the oscillators, internally. The binary input sequence is applied to the transmitter so as to choose the frequencies according to the binary input.

The following figure represents FSK modulated waveform along with its input.



FSK Modulated output wave

Procedure

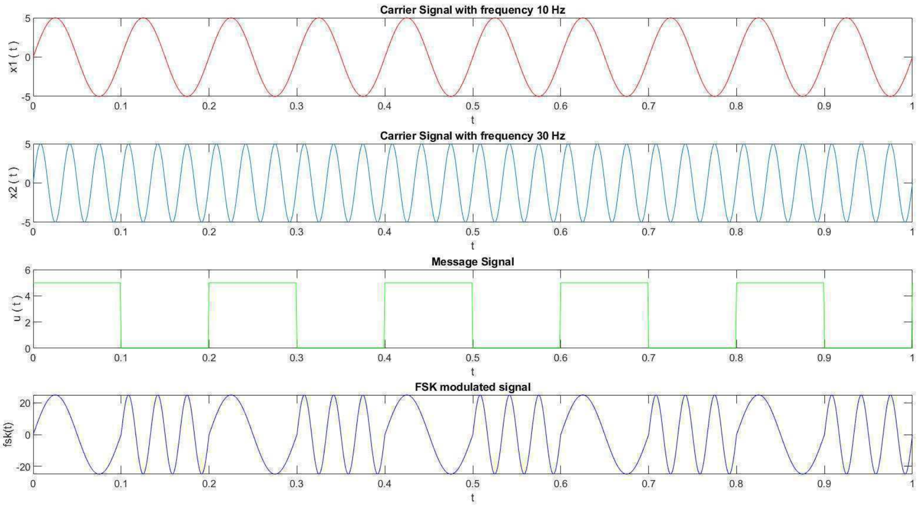
- Firstly, we have to initialize the frequencies f1, f2 and f3.
- We then define the amplitude variable, A and the time variable, t.
- Then we represent the carrier signal, x1 and x2 with frequencies f1 and f3 respectively.
- After that we define the square modulating signal, u.
- Finally we have calculated the FSK modulated signal

Matlab Code:

```
clc ;
clear all ;
close all ;
A = 5;
t = 0:0.001:1;
f1 = 10;
f2 = 5;
f3 = 30;
x1 = A .* sin (2* pi * f1 * t ) ;
x2 = A .* sin (2* pi * f3 * t ) ;
subplot (4 ,1 ,1) ;
plot (t , x1 , 'r') ;
title (" Carrier Signal with frequency 10 Hz ") ;
xlabel (" t ") ;
ylabel (" x1 ( t ) ") ;
subplot (4 ,1 ,2) ;
plot (t , x2 ) ;
title (" Carrier Signal with frequency 30 Hz ") ;
xlabel (" t ") ;
ylabel (" x2 ( t ) ") ;
u =( A /2) * square (2.* pi * f2 * t ) +( A /2) ; % message signal
subplot (4 ,1 ,3) ;
plot (t ,u , 'g') ;
title (" Message Signal ") ;
xlabel (" t ") ;
ylabel (" u ( t ) ") ;
```

```
u1 =( A /2) * square (2.* pi * f2 * t ) -( A /2) ;
y = x1 .* u + x2 .*( - u1 ) ;
subplot (4 ,1 ,4) ;
plot (t ,y , 'b') ;
title (" FSK modulated signal ") ;
xlabel (" t ") ;
ylabel (" fsk(t) ") ;
```

Output:



Conclusion

We have understood the concepts of frequency shift keying modulation and phase shift keying modulation. And practically performed the modulation techniques using MATLAB.

Indian Institute of
Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 6

Author:
Sarthak Babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

To perform Quadrature Phase Shift Keying(QPSK) modulation and demodulation using mat- lab.

Theory:

Phase Shift Keying (FSK):

Phase Shift Keying (PSK) is the digital modulation technique in which the phase of the carrier signal is changed by varying the sine and cosine inputs at a particular time. PSK technique is widely used for wireless LANs, bio-metric, contactless operations, along with RFID and Bluetooth communications.PSK is of two types, depending upon the phases the signal gets shifted..

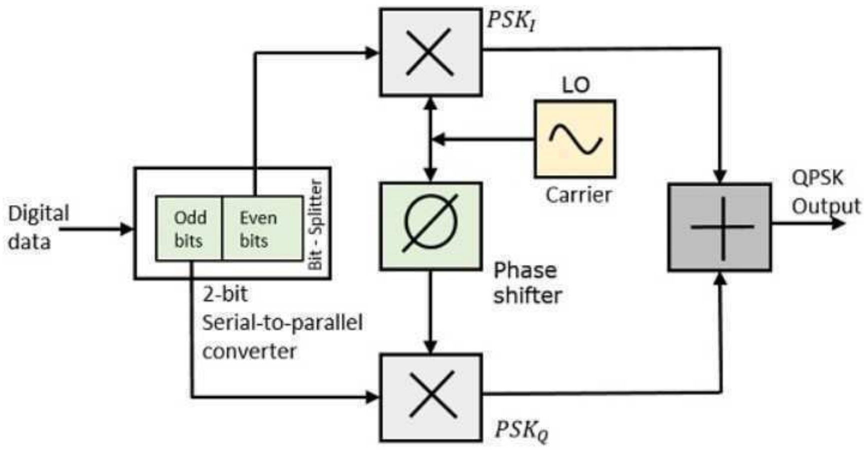
1)Binary Phase Shift Keying (BPSK) This is also called as 2-phase PSK or Phase Reversal Keying. In this technique, the sine wave carrier takes two phase reversals such as 0° and 180°.BPSK is basically a Double Side Band Suppressed Carrier (DSBSC) modulation scheme, for message being the digital information.

2)Quadrature Phase Shift Keying (QPSK) This is the phase shift keying technique, in which the sine wave carrier takes four phase reversals such as 0°, 90°, 180°, and 270°.

If this kind of techniques are further extended, PSK can be done by eight or sixteen values also, depending upon the requirement.
We have performed QPSK in this lab.

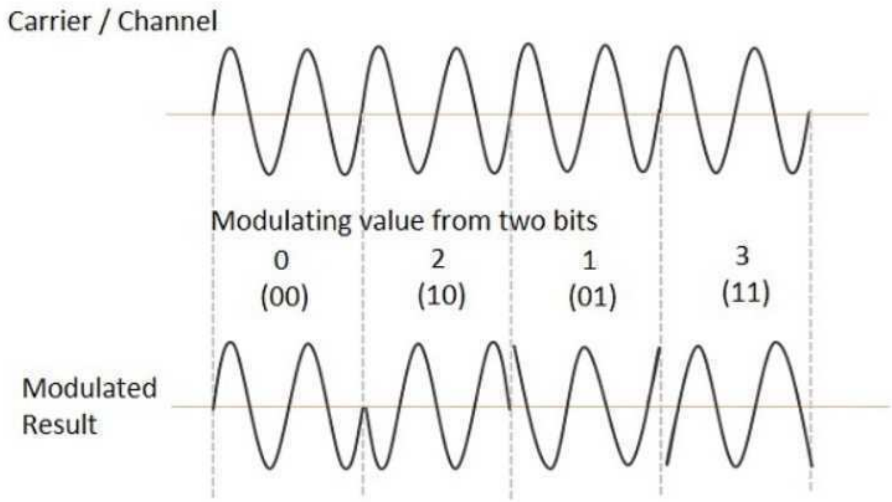
QPSK Modulator:

The QPSK Modulator uses a bit-splitter, two multipliers with local oscillator, a 2-bit serial to parallel converter, and a summer circuit. Following is the block diagram for the same.



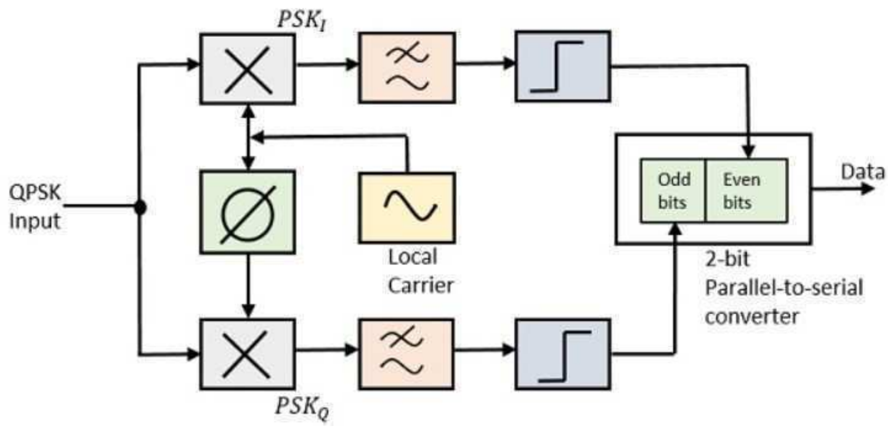
At the modulator's input, the message signal's even bits (i.e., 2 nd bit, 4 th bit, 6 th bit, etc.) and odd bits (i.e., 1 st bit, 3 rd bit, 5 th bit, etc.) are separated by the bits splitter and are mul- tiplied with the same carrier to generate odd BPSK (called as PSKI) and even BPSK (called as PSKQ). The PSKQ signal is anyhow phase shifted by 90° before being modulated.

The QPSK waveform for two-bits input is as follows, which shows the modulated result for different instances of binary inputs.



QPSK Demodulator:

The QPSK Demodulator uses two product demodulator circuits with local oscillator, two band pass filters, two integrator circuits, and a 2-bit parallel to serial converter. Following is the diagram for



the same.
The two product detectors at the input of demodulator simultaneously demodulate the two BPSK signals. The pair of bits are recovered here from the original data. These signals after processing, are passed to the parallel to serial converter.

Procedure

- Firstly we generate the quadrature carrier signal.
- We then generate message signal.
- We separately modulate even and odd bits signals.
- Then we plot the QPSK modulated signal.
- Then for demodulation, we define corelators and decision device.

Matlab Code:

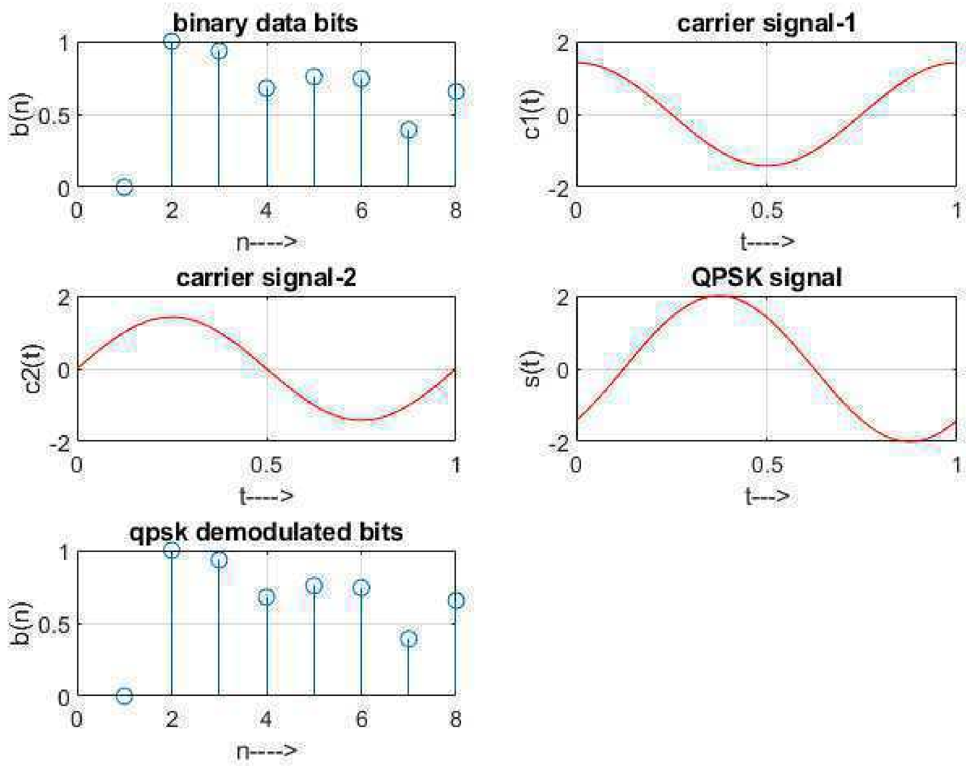
```
% QPSK Modulation

clc;

clear all;

close all;
```

```
%GENERATE QUADRATURE CARRIER SIGNAL

Tb=1;t=0:(Tb/100):Tb;fc=1;

c1=sqrt(2/Tb)*cos(2*pi*fc*t);

c2=sqrt(2/Tb)*sin(2*pi*fc*t);

%generate message signal

N=8;m=rand(1,N);

t1=0;t2=Tb

for i=1:2:(N-1)

t=[t1:(Tb/100):t2]

if m(i)>0.5

m(i)=1;

m_s=ones(1,length(t));

else

m(i)=0;

m_s=-1*ones(1,length(t));

%odd bits modulated signal

odd_sig(i,:)=c1.*m_s;

if m(i+1)>0.5

m(i+1)=1;

m_s=ones(1,length(t));

else

m(i+1)=0;

m_s=-1*ones(1,length(t));

end

%even bits modulated signal

even_sig(i,:)=c2.*m_s;

%qpsk signal

qpsk=odd_sig + even_sig;

%Plot the QPSK modulated signal

subplot(3,2,4);

plot(t,qpsk(i,:),'r');

title('QPSK signal');
xlabel('t--->');
ylabel('s(t)');

grid on;
hold on;

t1=t1+(Tb+.01);

t2=t2+(Tb+.01);
```

```
end
hold off
subplot(3,2,1);
stem(m);
title('binary data bits');
xlabel('n---->');
ylabel('b(n)');
grid on;
subplot(3,2,2);
plot(t,c1,'r');
title('carrier signal-1');
xlabel('t---->');
ylabel('c1(t)');
grid on;
subplot(3,2,3);
plot(t,c2,'r');
title('carrier signal-2');
xlabel('t---->');
ylabel('c2(t)');
grid on;
t1=0;t2=Tb
for i=1:N-1
t=[t1:(Tb/100):t2]
%correlator
subplot(3,2,5);
stem(m);
title('qpsk demodulated bits');
xlabel('n---->');
ylabel('b(n)');
grid on;
x1=sum(c1.*qpsk(i,:));
x2=sum(c2.*qpsk(i,:));
%decision device
if (x1>0&&x2>0)
demod(i)=1;
demod(i+1)=1;
```

```
elseif (x1>0&&x2<0)

demod(i)=1;

demod(i+1)=0;

elseif (x1<0&&x2<0)

demod(i)=0;
demod(i+1)=0;

elseif (x1<0&&x2>0)

demod(i)=0;

demod(i+1)=1;

end

t1=t1+(Tb+.01);

t2=t2+(Tb+.01);

end

subplot(3,2,5);

stem(demod);

title('qpsk demodulated bits');

xlabel('n---->');

ylabel('b(n)');

grid on;

end
```

Output:



The first red signal is the binary bit stream and the final red signal is the QPSK demodulated bit stream. We also have two carrier signals of green and blue colours. The QPSK modulated signal is also shown in the plots above.

Conclusion

We have understood the concepts of Quadrature phase shift keying modulation and demodulation techniques. And practically performed the modulation techniques using MATLAB.

Indian Institute of Information Technology

Digital Communication

ECE-320

Electronics And Communication Department

Experiment 7

Author:
Sarthak Babra
BT19ECE028

Submitted to:
Dr. Rashmi Pandhare
Course Instructor

AIM:

To perform Visualize channel capacity varition with bandwidth and signal power using matlab.

Theory:

Shannon capacity theorem :

Shannon capacity theorem defines the maximum amount of informa- tion, or data capacity, which can be sent over any channel or medium (wireless, coax, twister pair, fiber etc.).

$$C = B * log2(1 + S/N) \tag{1}$$

Where,
C is the channel capacity in bits per second (or maximum rate of data).
B is the bandwidth in Hz available for data transmission.
Sis the received signal power.
N is the total channel noise power across bandwidth B.
This says is that higher the signal-to-noise (SNR) ratio and more the channel bandwidth, the higher the possible data rate. This equation sets the theoretical upper limit on data rate, which of course is not fully achieved in practice.
It does not make any limitation on how low the achievable error rate will be. That is dependent on the coding method used.

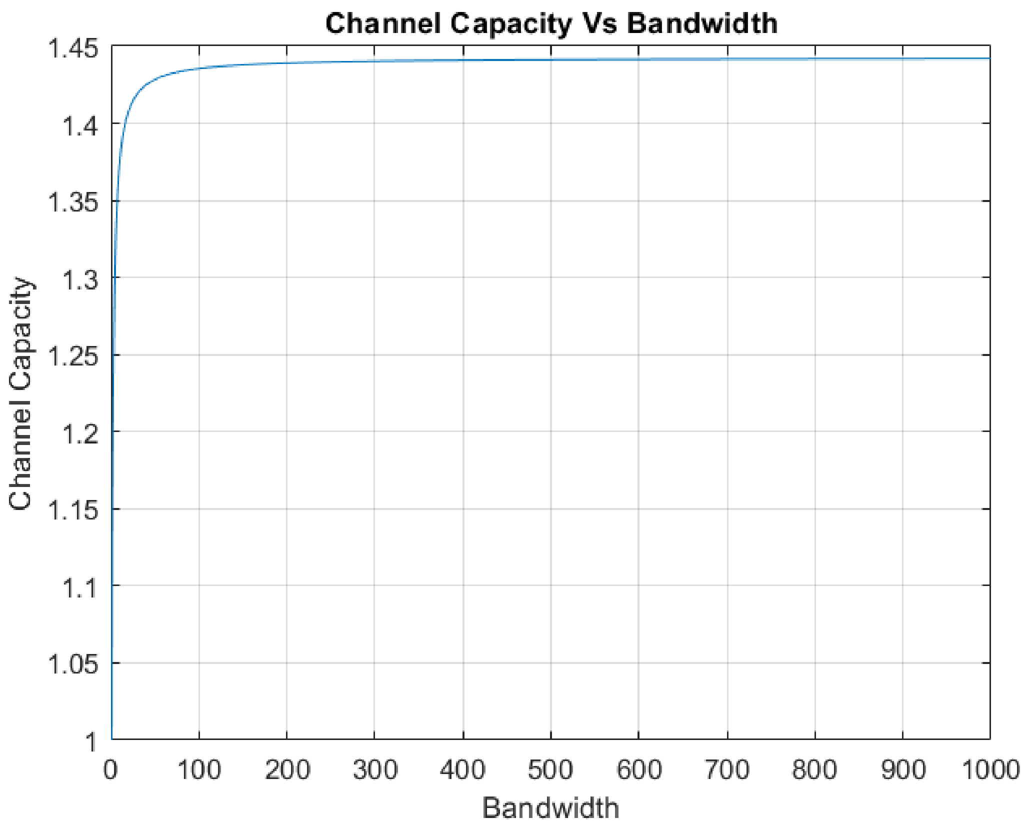
Procedure

- Firstly we generate the carrier signal.
- We then generate message signal.
- We apply C = B ∗ log2(1 + S/N) .
- Then we plot the Channel Capacity Vs Bandwidth.

Matlab Code:

```
clc
clear all
close all
B = 1:0.01:1000;
S = 2 ;
N0 = 2;
N = N0 * B;
C=B.*log2(1+ S./N);
plot(B,C);
xlabel('Bandwidth');
ylabel('Channel Capacity');
title('Channel Capacity Vs Bandwidth');
grid on;
```

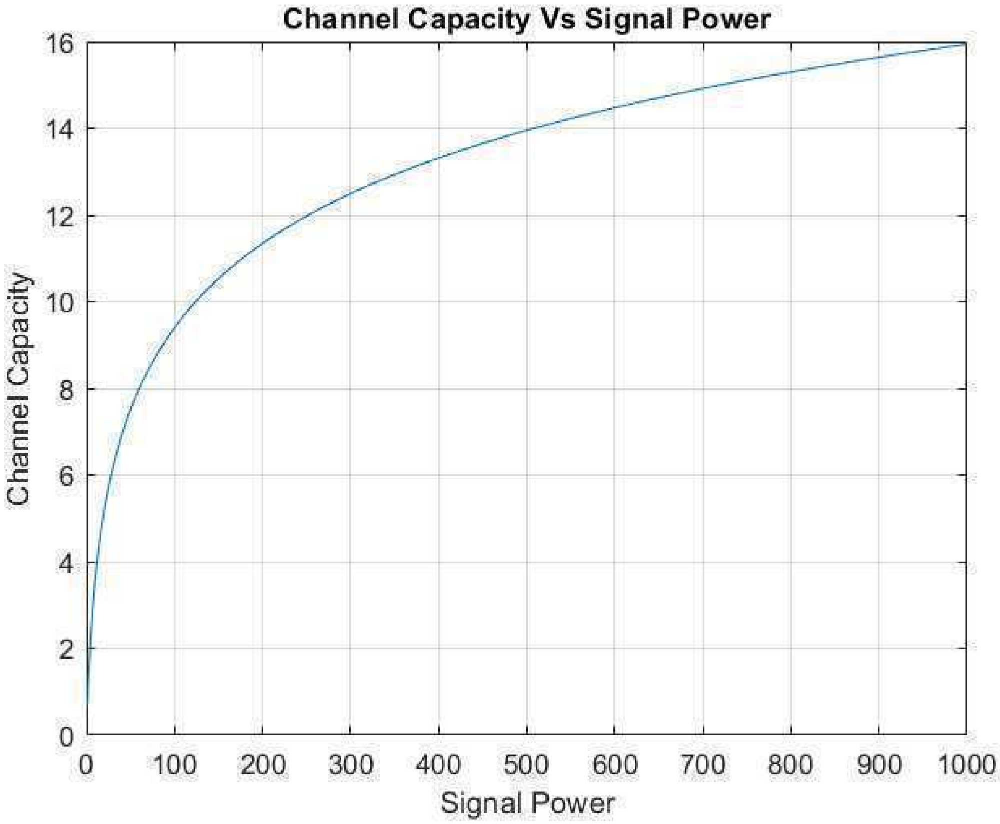
Output:



Matlab Code:

```
clc
clear all
close all
S = 1:0.01:1000;
B = 2 ;
N0 = 2;
N = N0 * B;
C=B.*log2(1+ S./N);
plot(S,C);
xlabel('Signal Power');
ylabel('Channel Capacity');
title('Channel Capacity Vs Signal Power');
grid on;
```

Output:



Conclusion

In this report we have successfully performed the Shannon Theorem using matlab.

Indian Institute of
Information Technology,
Nagpur

Digital Communication (ECL-320)

Lab Report 8

Submitted By :
Sarthak Babra (BT19ECE028)
Semester 6
Electronics and Communication Engineering Dept.

Submitted To :
Dr. Rashmi Pandhare
Course Instructor

Experiment-8: Encoding and Decoding of Linear Block Code using MATLAB

Theory:

Linear block code is a type of error-correcting code in which the actual information bits are linearly combined with the parity check bits so as to generate a linear codeword that is transmitted through the channel. Another major type of error-correcting code is convolution code.

In the linear block code technique, the complete message is divided into blocks and these blocks are combined with redundant bits so as to deal with error detection and correction.

Code:

Listing 1: "Linear Block Code"

```
clc;
clear all;
close all;

% Codeword length
n = input('Enter Codeword length: ');
% Message length
k = input('Enter Message length: ');

P = input('Enter Parity Length: ');

I = eye(k);

M = input('Enter the message Signal: ');

G = [I P]

encData = encode(M,n,k,'linear/binary',G)

decData = decode(encData,n,k,'linear/binary',G)
```

Figure 1: Output

```
clc;
clear all;
close all;

% Codeword length
n = input('Enter Codeword length: ');
Enter Codeword length: 6
% Message length
k = input('Enter Message length: ');
Enter Message length: 3

P = input('Enter Parity Length: ');
Enter Parity Length: [1 0 0;1 1 1;0 1 0]

I = eye(k);

M = input('ENter the message Signal: ');
ENter the message Signal: [1 0 0]

G = [I P]

G =

     1     0     0     1     0     0
     0     1     0     1     1     1
     0     0     1     0     1     0


encData = encode(M,n,k,'linear/binary',G)

encData =

     1     0     0     1     0     0


decData = decode(encData,n,k,'linear/binary',G)
Single-error patterns loaded in decoding table.  3 rows remaining.
2-error patterns loaded.  0 rows remaining.

decData =

     1     0     0

fx >>
```

Conclusion: In this report we have successfully encoded and decoded message signal using matlab.

Lab Report - 9

Indian Institute of Information Technology, Nagpur

Name:	Sarthak Babra
Enrollment Number:	BT19ECE028
Course Title:	Digital Communication
Course Code:	ECL 320
Course Instructor:	Dr Rashmi Pandhare
Date of Submission:	26 April 2022

Aim

To write a matlab code to encode and decode of cyclic block code.

Apparatus or Equipment's Required

A PC with the software MATLAB.

Theory

The cyclic property of code words is that any cyclic-shift of a code word is also a code word. Cyclic codes follow this cyclic property.

For a linear code C, if every code word i.e., $C = C_1, C_2, ...C_n$

from C has a cyclic right shift of components, it becomes a code word. This shift of right is equal to n-1 cyclic left shifts. Hence, it is invariant under any shift. So, the linear code C, as it is invariant under any shift, can be called as a Cyclic code.

Cyclic codes are used for error correction. They are mainly used to correct double errors and burst errors.

Hence, these are a few error correcting codes, which are to be detected at the receiver. These codes prevent the errors from getting introduced and disturb the communication. They also prevent the signal from getting tapped by unwanted receivers.

Procedure

- Firstly, take the message length as input.
- Then take the code word length as input.
- Take the message signal input form the user.

- Calculate the generator matrix.
- Generate the equation of message with the help of generator matrix.
- Encode using build in encode funtion.
- Decode the encoded data using the function decode.

MATLAB Code

```
% Encode and Decode Message with Cyclic Block Code
clc ;
clear all;
close all;
k = input('Enter the length of Msg word : ');
n = input('Enter the length of Codeword : ');
m = input('Enter the Msg. Word : ');
G = cyclpoly(n,k,'max');
qx = poly2sym(G);
disp('Equation = ')
disp(qx)
C = encode(m,n,k,'cyclic',G);
disp('C = ')
disp(C);
D = decode(C,n,k,'cyclic',G);
disp('D = ')
disp(D);
```

Results and Discussion

```
Enter the length of Msg word : 4
Enter the length of Codeword : 7
Enter the Msg. Word : [1 0 0 1]
Equation =
x^3 + x^2 + 1

C =
     0     1     1     1     0     0     1

D =
     1     0     0     1
```

Here we can see that the data is encoded using the built-in function encode and decoded using the built-in function using decode and the decode message signal is same as that of the input message signal.

Conclusion

We have done the encoding and decoding of cyclic block code using matlab.

INDIAN INSTITUTE OF
INFORMATION TECHNOLOGY,
NAGPUR

ECL-320
Digital Communication System
Project Report

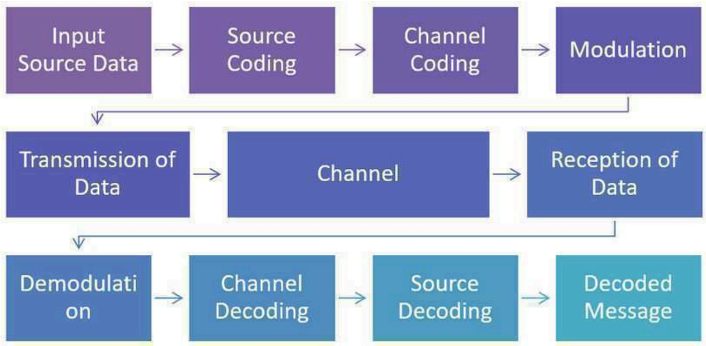
Submitted By :
Sarthak Babra (BT19ECE028)
Avish fakirde (BT19ECE037)
Ganesh (BT19ECE031)

Semester 6
Electronics and Communication Engineering Dept.

Submitted To :
Dr. Rashmi Pandhare
Course Instructor

About our project :-

This project is the simulation of a complete digital communication system. A digital communication system consists of multiple blocks and each block is implemented here as a MATLAB function and the maincode.m file combines them all to build the complete system. The system reads text and process the text data accordingly and writes the received text. The block diagram below shows all the blocks of the system in a sequential manner.



Each block is built using the following techniques.

- Source Coding: Huffman encoding
- Channel Coding: Convolutional encoding
- Modulation: BPSK modulation
- Channel: Additive White Gaussian Noise Channel
- Demodulation: BPSK demodulation
- Channel Decoding: Viterbi decoding
- Source Decoding: Huffman decoding

Source Coding :-

The source encoder compresses the data into minimum number of bits. This process helps in effective utilization of the bandwidth. It removes the redundant bits unnecessaryexcessbits,i.e.,zeroes

- Huffman encoding :- Huffman Coding is a technique of compressing data to reduce its size without losing any of the details. It was first developed by David Huffman. Huffman Coding is generally useful to compress the data in which there are frequently occurring characters

Channel Coding :-

The channel encoder, does the coding for error correction. During the transmission of the signal, due to the noise in the channel, the signal may get altered and hence to avoid this, the channel encoder adds some redundant bits to the transmitted data. These are the error correcting bits.

- Convolutional encoding :- Convolutional code is another type of error-correcting code where the output bits are obtained by performing a desired logical operation on a present bitstream along with considering some bits of the previous stream. This coding technique rather than depending on the block of bits shows dependency on bitstream.

Modulation :-

it is the process of encoding a digital information signal into the amplitude, phase, or frequency of the transmitted signal. The encoding process affects the bandwidth of the transmitted signal and its robustness to channel impairments.

- Binary Phase-shift keying (BPSK) :- it is a digital modulation scheme that conveys data by changing, or modulating, two different phase s of a reference signal (the carrier wave). The constellation points chosen are usually positioned with uniform angular spacing around a circle. This gives maximum phase-separation between adjacent points and thus the best immunity to corruption

Channel :-

The channel or a medium, allows the analog signal to transmit from the transmitter end to the receiver end.

- Additive White Gaussian Noise Channel :- AWGN is a basic noise model used in information theory to mimic the effect of many random processes that occur in nature.

Demodulation :-

This is the first step at the receiver end. The received signal is demodulated as well as converted again from analog to digital. The signal gets reconstructed here.

- Binary Phase-shift keying (BPSK) :- The BPSK Demodulator Baseband block demodulates a signal that was modulated using the binary phase shift keying method. The input is a baseband representation of the modulated signal. This block accepts a scalar or column vector input signal. The input signal must be a discrete-time complex signal.

Channel Decoding :-

The channel decoder, after detecting the sequence, does some error corrections. The distortions which might occur during the transmission, are corrected by adding some redundant bits. This addition of bits helps in the complete recovery of the original signal.

- The Viterbi algorithm is based on the Maximum-Likelihood decoding technique. The main purpose of the decoder is to select the code word with the minimum distance between the received signal and the code word.Viterbi algorithm is utilized to decode the convolutional codes.

Source Decoding :-

The resultant signal is once again digitized by sampling and quantizing so that the pure digital output is obtained without the loss of information. The source decoder recreates the source output.

- Huffman decoding :- In Huffman decoding we use Huffman Encoded data to obtain the initial, uncompressed data again. Having our Binary Huffman Tree obtained during encode phase, decoding is a very simple process to perform.

Project Link For Code Files :-

`https://github.com/sarthu07/digital_communication.git`

Output :-

```
Command Window
Reading data: heyy i am sarthak babra my enrollment no. is bt19ece028, heyy i am avish my enrollment no. is bt19ece037,
Source statistics: Elapsed time is 0.319694 seconds.
Huffman encoding: Elapsed time is 0.218978 seconds.
Stream generator: Elapsed time is 0.046207 seconds.
Channel coding: Elapsed time is 0.812274 seconds.
Modulation: Elapsed time is 0.169558 seconds.
Channel: Elapsed time is 0.402365 seconds.
Demodulation: Elapsed time is 0.254308 seconds.
Channel decoding: Elapsed time is 1.296515 seconds.
Huffman decoding: Elapsed time is 0.065582 seconds.
Writing data: heyy i am sarthak babra my enrollment no. is bt19ece028, heyy i am avish my enrollment no. is bt19ece037,
Total execution time: Elapsed time is 3.648025 seconds.
Total Bit Error: 0
fx >>
```

Conclusion

In this project we have successfully make multiple block of Digital communication system using matlab for transmitting and receiving messages.