Calculate Entrophy(H)& Rate of information

```
clear;
px1=input("Enter the first value=")
px2=input("Enter the second value=")
px3=input("Enter the third value=")
px4=input("Enter the fourth value=")
px5=input("Enter the fifth value=")
Tb=10^-3;
HX=px1*log2(1/px1)+px2*log2(1/px2)+px3*log2(1/px3)+px4*log2(1/px4)+px5*log2(1/px5)
printf("i)source entropy\n\n tH(x) = %2f bits/symbol\n",HX);
r=1/Tb;
R=r*HX;
printf("\n\n ii) information rate \n\n t R = =%d bits /sec",R);
```

[1/2,1/4,1/8,1/16,1/16] – op

RZ at unipolar waveform

```
clear;
close;
clc;
x=[1 0 1 0 0 1 1 0]
z=0;
for i=1:length(x)
   t=[z:z+1]
      subplot(2,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Data')
      plot(t,x(i))
      t=[z:0.5:z+0.5]
      subplot(2,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Polar RZ')
      if(x(i)==0)
         plot(t,0)
      else
         plot(t,1)
      end
   t=[z+0.5:0.5:z+1]
   plot(t,0)
      z=z+1;
end
```

NRZ at unipolar waveform

```
clear;
close;
clc;
clf;
x=[1 0 1 0 0 1 1 0]
z=0;
for i=1:length(x)
   t=[z:1:z+1]
   subplot(2,1,1)
   a=gca();
   a.data_bounds=[0,1.5;length(x),-1.5]
   a.grid=[1,-1]
   title('Data')
   plot(t,x(i))

   subplot(2,1,2)
   a=gca();
   a.data_bounds=[0,1.5;length(x),-1.5]
   a.grid=[1,-1]
   title('NRZ')
   if(x(i)==0)
      plot(t,0)
   else
      plot(t,1)
   end
   z=z+1
   end
```

RZ at polar waveform

```
clear;
close;
clc;
x=[1 0 1 0 0 1 1 0]
z=0;
for i=1:length(x)
   t=[z:z+1]
      subplot(2,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Data')
      plot(t,x(i))
    t=[z:0.5:z+0.5]
      subplot(2,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
```

```
      title('Polar RZ')
      if(x(i)==0)
         plot(t,-1)
      else
         plot(t,1)
      end
  t=[z+0.5:0.5:z+1]
   plot(t,0)
   z=z+1;
end
```

NRZ at polar waveform

```
clear;
close;
clc;
x=[1 0 1 0 0 1 1 0]
z=0;
for i=1:length(x)
   t=[z:1:z+1]
      subplot(2,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Data')
      plot(t,x(i))
      subplot(2,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Polar NRZ')
      if(x(i)==0)
         plot(t,-1)
      else
         plot(t,1)
      end
   z=z+1
end
```

Bipolar RZ waveform

```
clear;
close;
clc;
x=[1 0 1 0 0 1 1 0]
z=0;
ob=-1;
for i=1:1:length(x)
    subplot(2,1,1)//Data Plot
   a=gca();
```

```
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Data')
       t=[z:1:z+1]
      plot(t,x(i))
  subplot(2,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Bipolar RZ')
      if(x(i)==0)
         t=[z:1:z+1]
         plot(t,0)//Plot zero
      else
         t=[z:0.5:z+0.5]
         ob=-ob
         plot(t,ob)
         t=[z+0.5:0.5:z+1]
         plot(t,0)
      end
    z=z+1
end
```

BIPOLAR NRZ waveform

```
clear;
close;
clc;
x=[1 0 1 0 0 1 1 0]
z=0;
ob=-1;
for i=1:1:length(x)
  subplot(2,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Data')

      t=[z:1:z+1]
      plot(t,x(i))
   subplot(2,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(x),1.5]
      a.grid=[1,-1]
      title('Bipolar NRZ')
      if(x(i)==0)
         t=[z:1:z+1]
         plot(t,0)
      else
```

```
        t=[z:1:z+1]
        ob=-ob
        plot(t,ob)
      end
   z=z+1
end
```

Amplitude Shift Key [ASK]

```
clear;
clc;
close;
t = 0:0.01:1;
f=2;
I=[0,0,1,1,0,1,0,1];
z=0;
for n=1:length(I)
   subplot(3,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('Carrier')
         plot((t+z),sin(2*%pi*f*t));
   subplot(3,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('Data')
         plot((t+z),I(n));
   subplot(3,1,3)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('ASK Waveform')
         plot((t+z),(sin(2*%pi*f*t))*(I(n)));
   z=z+1;
end
```

Frequency Shift Key [FSK]

```
clear;
clc;
close;
t = 0:0.01:1;
f1=2;
f2=4;
I=[0,0,1,1,0,1,0,1];
```

```
z=0;
for n=1:length(I)
   subplot(4,1,1)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('Carrier 1')
         plot((t+z),sin(2*%pi*f1*t));
   subplot(4,1,2)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('Carrier 2')
         plot((t+z),sin(2*%pi*f2*t));
   subplot(4,1,3)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('Data')
         plot((t+z),I(n));
    subplot(4,1,4)
      a=gca();
      a.data_bounds=[0,-1.5;length(I),1.5];
      a.x_location="bottom";
      a.grid=[1,-1];
      title('FSK Waveform')
         if (I(n)==0)
            plot((t+z),sin(2*%pi*f1*t));
         elseif (I(n)==1)
            plot((t+z),sin(2*%pi*f2*t));
         end
   z=z+1;
end
```

Phase Shift Keying [PSK]

```
clear;
clc;
close;
t = 0:0.01:1;
f=2;
I=[0,0,1,1,0,1,0,1];
z=0;
for n=1:length(I)
   subplot(3,1,1)
      a=gca();
```

```
    a.data_bounds=[0,-1.5;length(I),1.5];
    a.x_location="bottom";
    a.grid=[1,-1];
    title('Carrier')
      plot((t+z),sin(2*%pi*f*t));
 subplot(3,1,2)
    a=gca();
    a.data_bounds=[0,-1.5;length(I),1.5];
    a.x_location="bottom";
    a.grid=[1,-1];
    title('Data')
      plot((t+z),I(n));
 subplot(3,1,3)
    a=gca();
    a.data_bounds=[0,-1.5;length(I),1.5];
    a.x_location="bottom";
    a.grid=[1,-1];
    title('PSK Waveform')
      if (I(n)==1)
        plot((t+z),sin(2*%pi*f*t));
      elseif (I(n)==0)
        plot((t+z),sin((2*%pi*f*t)+%pi));
      end
  z=z+1;
end
```

Aim:- To study signal sampling and its reconstruction.

Equipments:- Experimenter kit DCL 01, Connecting chords, Power supply, 20 MHz, Dual Trace Oscilloscope.

Theory:-

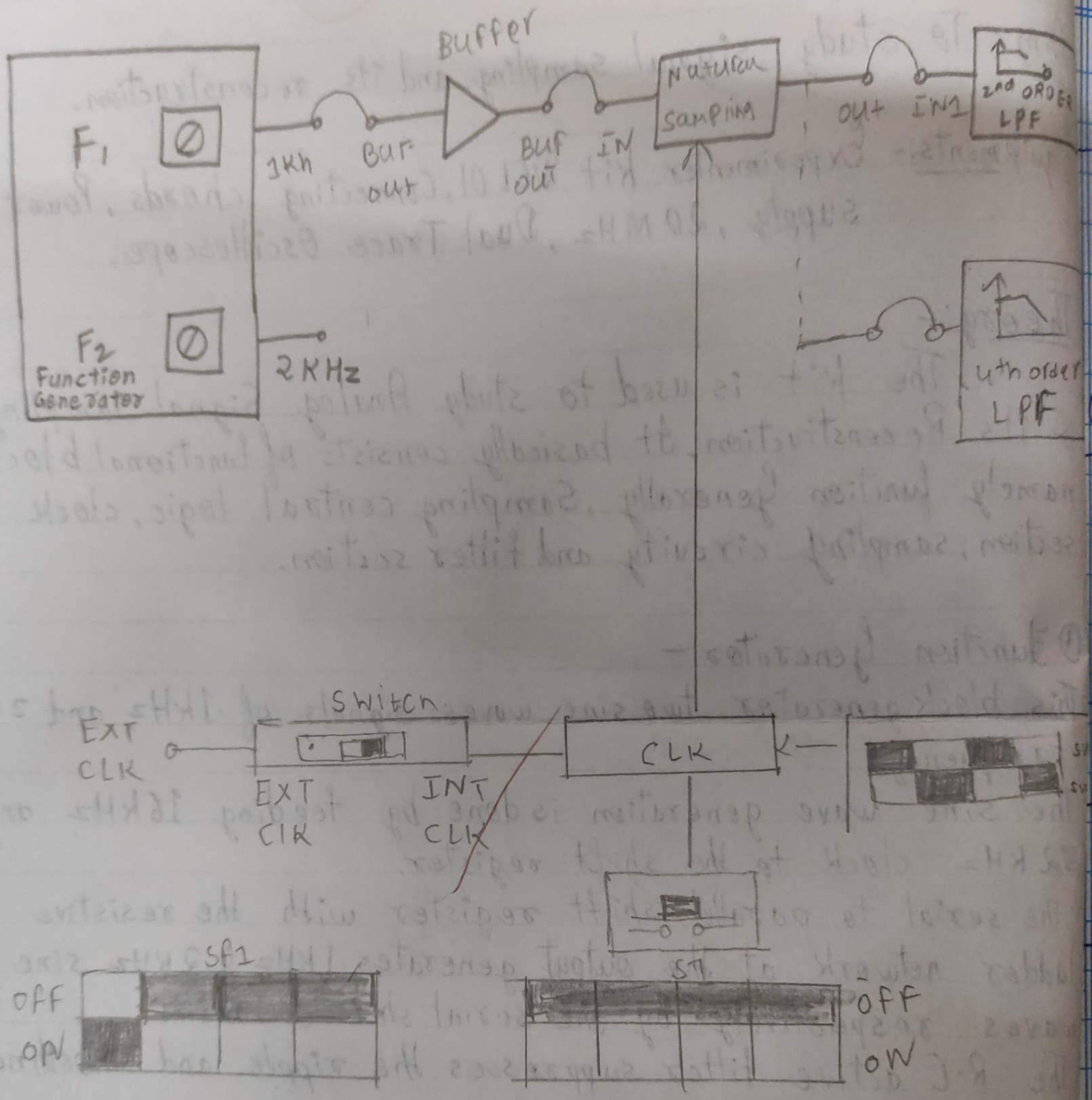
The kit is used to study Analog Signal Sampling & its Reconstruction. It basically consists of functional blocks, namely function Generally, Sampling central logic, clock section, sampling circuity and filter section.

① Function Generator -

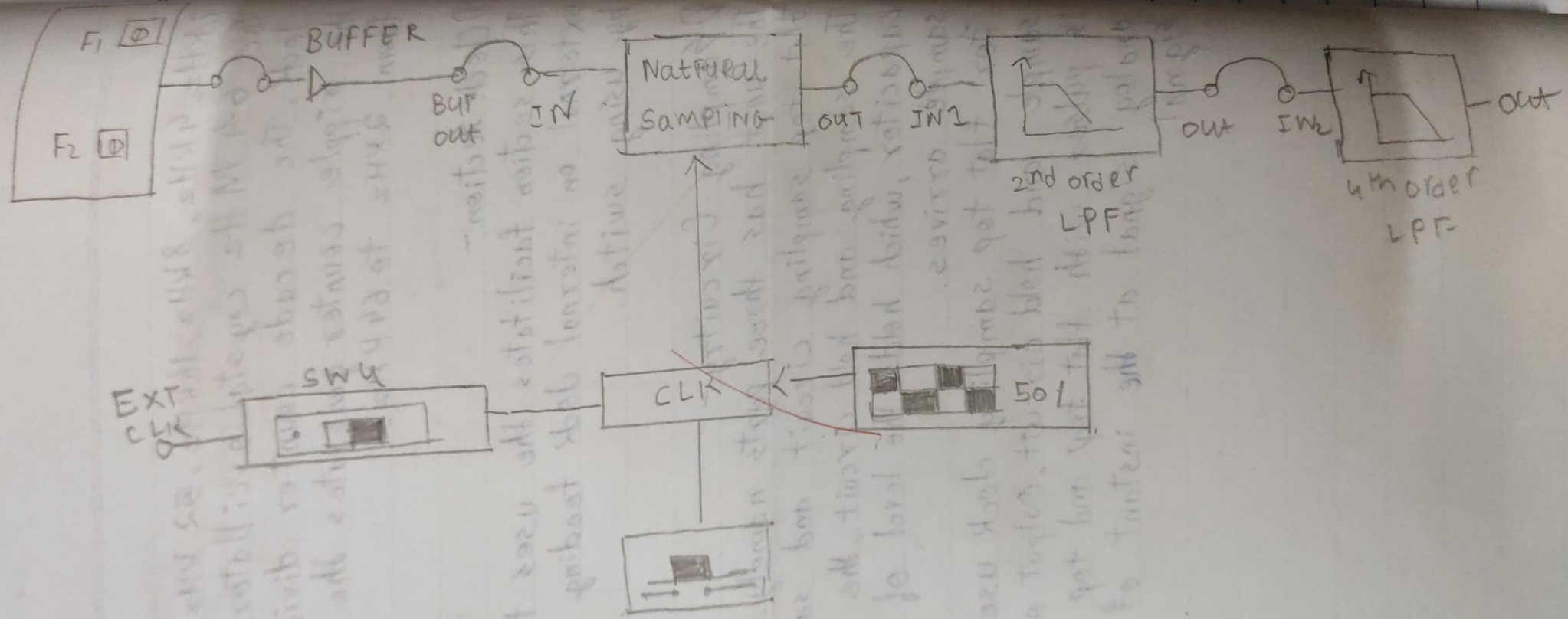
- This block generator two sine waves signals of 1kHz and 2kHz frequency.
- The sine wave generation is done by feeding 16kHz and 32kHz clock to the shift register.
- The serial to parallel shift register with the resistive ladder network at the output generates 1kHz & 2kHz sine waves respectively by the serial shift operation.
- The R-C active filter suppresses the ripple and smoothness the sine wave.
- The unity gain amplifer buffer takes care of the impedence matching between sinewave generation and sampling circuit.

② Sampling Control Logic:-

- This unit generates two main signals used in the study of sampling Theorem, namely the analog signals of frequency 1kHz & 2kHz and sampling signal of freq




SWITCH FAULTS selection switch


F1
F2
BUFFER
BUF OUT
IN
Natural Sampling
OUT
IN1
2nd order LPF
OUT
IN2
4th order LPF
OUT
EXT CLK
SW4
CLK
50%

Procedure:-

① Refer the block diagram and carry out the following connections and switch settings.

② Connect power supply in property to the DCL-01 and switch it on.

③ Connect the 1KHz, 5Vpp sine wave signal, generated on board, to the BUF IN past of the BUFFER and BUFOUT past of the BUFFER to the IN post of the Natural sampling block by means of the connecting chords provided.

④ Connect the sampling frequency clock in the internal viode INTCLK using switch.

~~⑤ Using clock selector frequency clock~~

⑤ Using clock selector switch (S1) select 8KHz sampling frequency.

⑥ Using switch SW2 select 50% duty cycle.

⑦ Connect the OUT post of the Natural sampling block to the input IN1 post of the 2nd order low Pass Butterworth filter and then connect OUTPOST of Natural sampling block to input of 4th order low Pass Butterworth filter and take necessary observations.

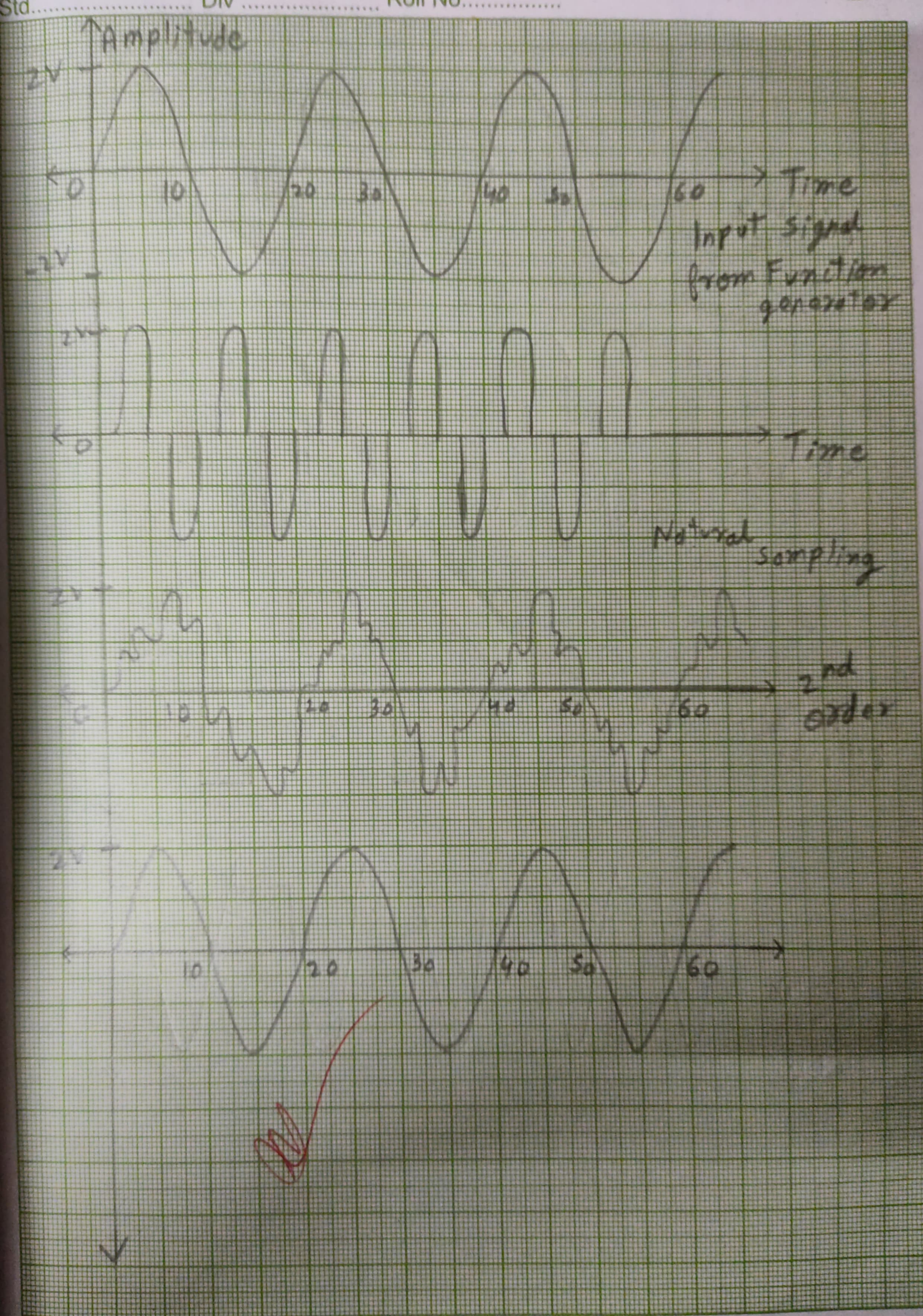
⑧ Connect OUTPOST of the Natural Sampling block to Input post of 2nd order Butterworth LPF and OUTPUT of 2nd Order Butterworth LPF to Input of 4th Order Butterworth LPF and take necessary observations:- the following waveform in order for every setting and plot it on the paper.

ⓐ Analog input waveform.

ⓑ Sampling frequency waveform.

ⓒ Natural sampling signal & it's corresponding output of 2nd and 4th Order low Pass Butterworth Filter.

Std..... Div ..... Roll No.....



Aim - To Calculate Entropy (H) & Rate of information using scilab.

Software used - Scilab.

Producedure :-

The five message given by probabilities 1/2, 1/4, 1/8, 1/16, 1/16.

$P(x_1) = 1/2$, $P(x_2) = 1/4$, $P(x_3) = 1/8$, $P(x_4) = 1/16$, $P(x_5) = 1/16$.

$r = 1$ messages/second

Source entropy (H).

$$H = \sum_{K=1}^{5} P_K \log_2 \frac{1}{P_K}$$

$$\therefore H = \left[P_1 \log_2 \frac{1}{P_1}\right] + \left[P_2 \log_2 \frac{1}{P_2}\right] + \left[P_3 \log_2 \frac{1}{P_3}\right] + \left[P_4 \log_2 \frac{1}{P_4}\right] + \left[P_5 \log_2 \frac{1}{P_5}\right]$$

$$= \frac{1}{2} \log_2 \frac{1}{1/2} + \frac{1}{4} \log_2 \frac{1}{1/4} + \frac{1}{8} \log_2 \frac{1}{1/8} + \frac{1}{16} \log_2 \frac{1}{1/16} + \frac{1}{16} \log_2 \frac{1}{1/16}$$

$$= \frac{1}{2} \log_2 2 + \frac{1}{4} \log_2 4 + \frac{1}{8} \log_2 8 + \left[\frac{1}{16} \log_2 16\right] x^2$$

$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{1}{2} = \underline{1.875 \text{ bits/second}}$$

# Information Rate

$R = r \cdot H$ bits/second
$R$ = Rate of information
$r$ = rate of which Positive
$H$ = Average information

$r = 1/T_b \quad \therefore (T_b = 10^{-3})$

$r = \frac{1}{10^{-3}}$

$r = 1000$

$\therefore R = 1000 \times 1.875 = 1875$ bit/symbol

Observed Entropy - $h(x) = \cancel{2.125}\ 1.875$ bit/symbol
Observed Information Rate = 1875 bit/symbol

Conculsion - Calculated value & observed values of Entropy & Information Rate are same.

8

**Aim:-** To study and observe NRZ & RZ at unipolar and polar waveforms.

**Software used:-** Scilab

**Theory:** $x = [1, 0, 1, 0, 0, 1, 1, 0]$

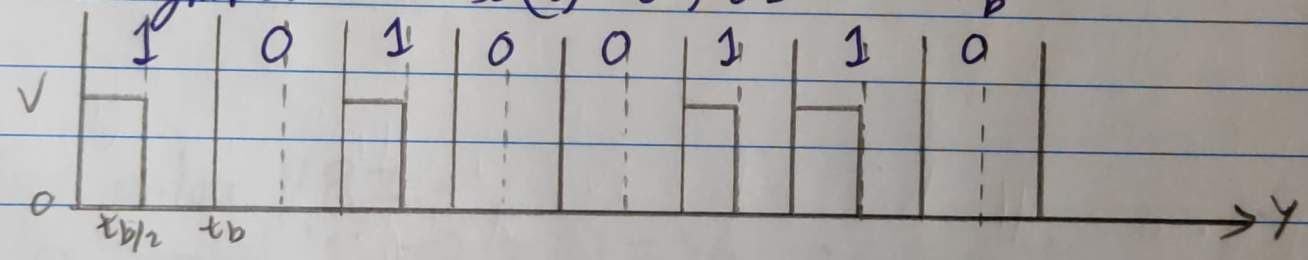
1) Unipolar:

ⓐ **For unipolar RZ:-**

→

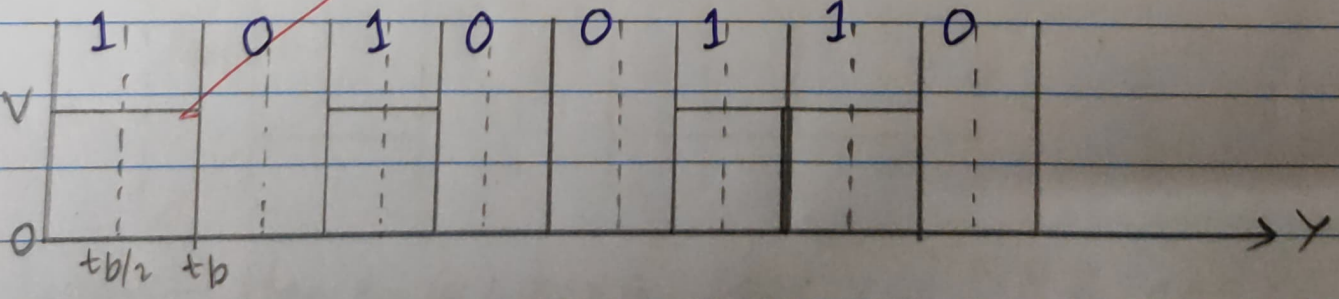
For symbol '1' ⇒ $x(t) = v;\ 0 < t < t_{b/2}$
$= 0;\ t_{b/2} < t < t_b$

→ for symbol '0' ⇒ $x(t) = 0;\ 0 \le t \le t_b$




ⓑ **For unipolar NRZ:-**

→ For symbol '1': $x(t) = v;\ 0 < t < t_b$

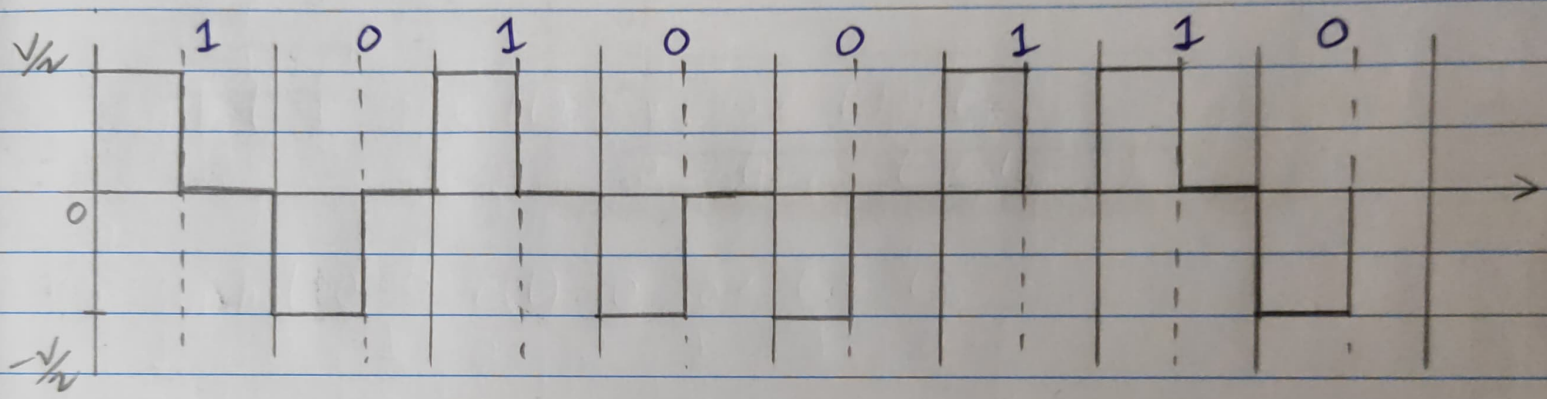


2) Polar :-

ⓐ For polar RZ :-

→ For Symbol '1' ⇒ $x(t) = V/2$ ; $0 \le t \le t_{b/2}$
$= 0$ ; $t_{b/2} \le t \le t_b$

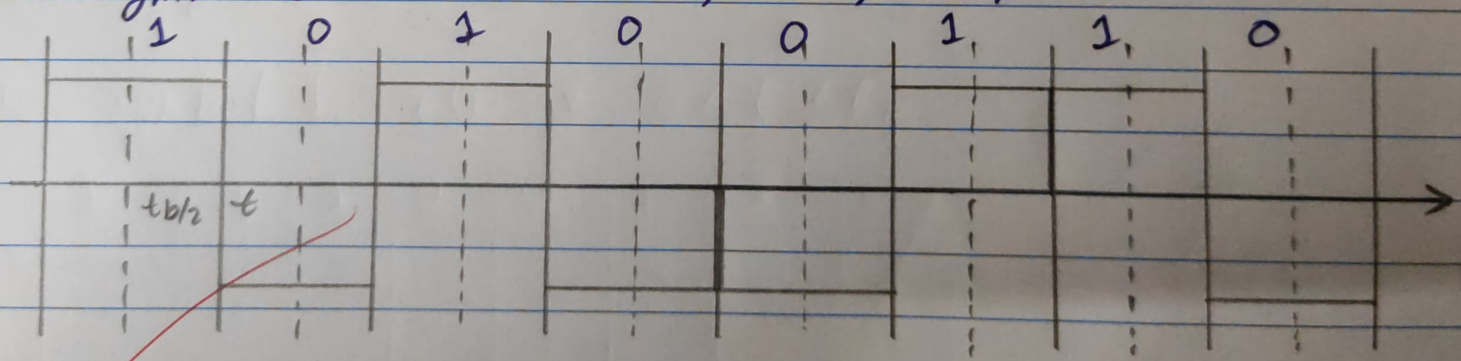
→ for symbol '0' ⇒ $x(t) = -V/2$ ; $0 \le t \le t_{b/2}$
$= 0$ ; $t_{b/2} \le t \le t_b$




ⓑ For polar NRZ :-

→ for symbol '1' ⇒ $x(t) = V/2$; $0 \le t \le t_b$

→ for symbol '0' ⇒ $x(t) = -V/2$ ; $0 \le t \le t_b$




Conclusion :- Hence, we have observed NRZ and RZ condition for unipolar and polar wave form.
DC value is zero for unipolar.

**Aim:-** To study and observe NRZ and RZ of bipolar waveform.

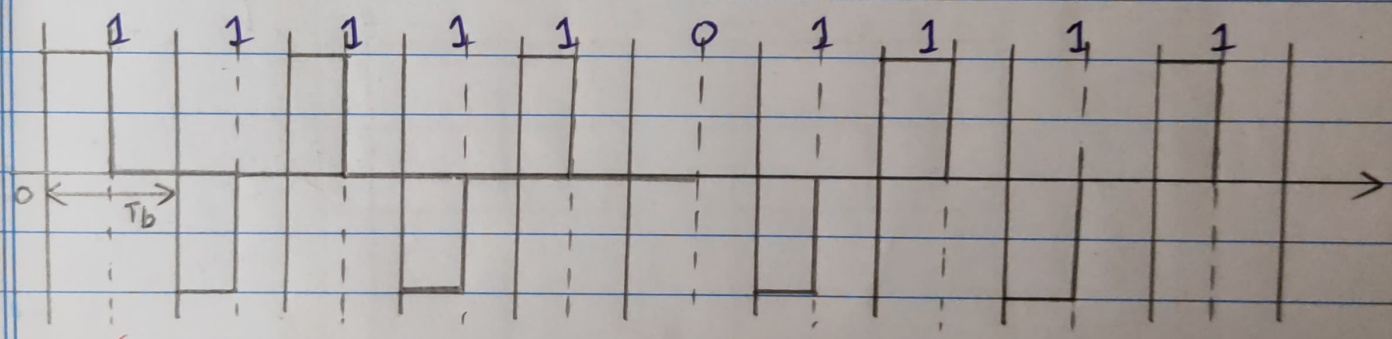
**Software used:-** Scilab

**Theory:-**

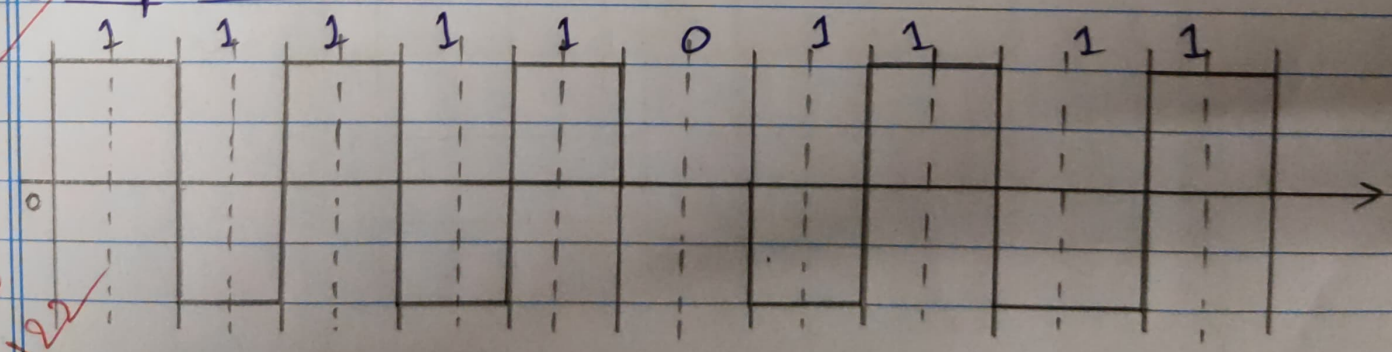
① **Bipolar RZ:-**

→ Successive '1' are represented by pulses but with opposite polarity and 0's are presented with no pulse.

X = [1 1 1 1 1 0 1 1 1 1]




② **Bipolar NRZ:-**




**Conclusion:-** Hence, we have observed RZ and NRZ conditions for bipolar waveforms. DC value is non-zero for Bipolar.

Aim:- Observation of the Constellation Diagram For event odd Data & study Bandwidth efficiency of QPSK.

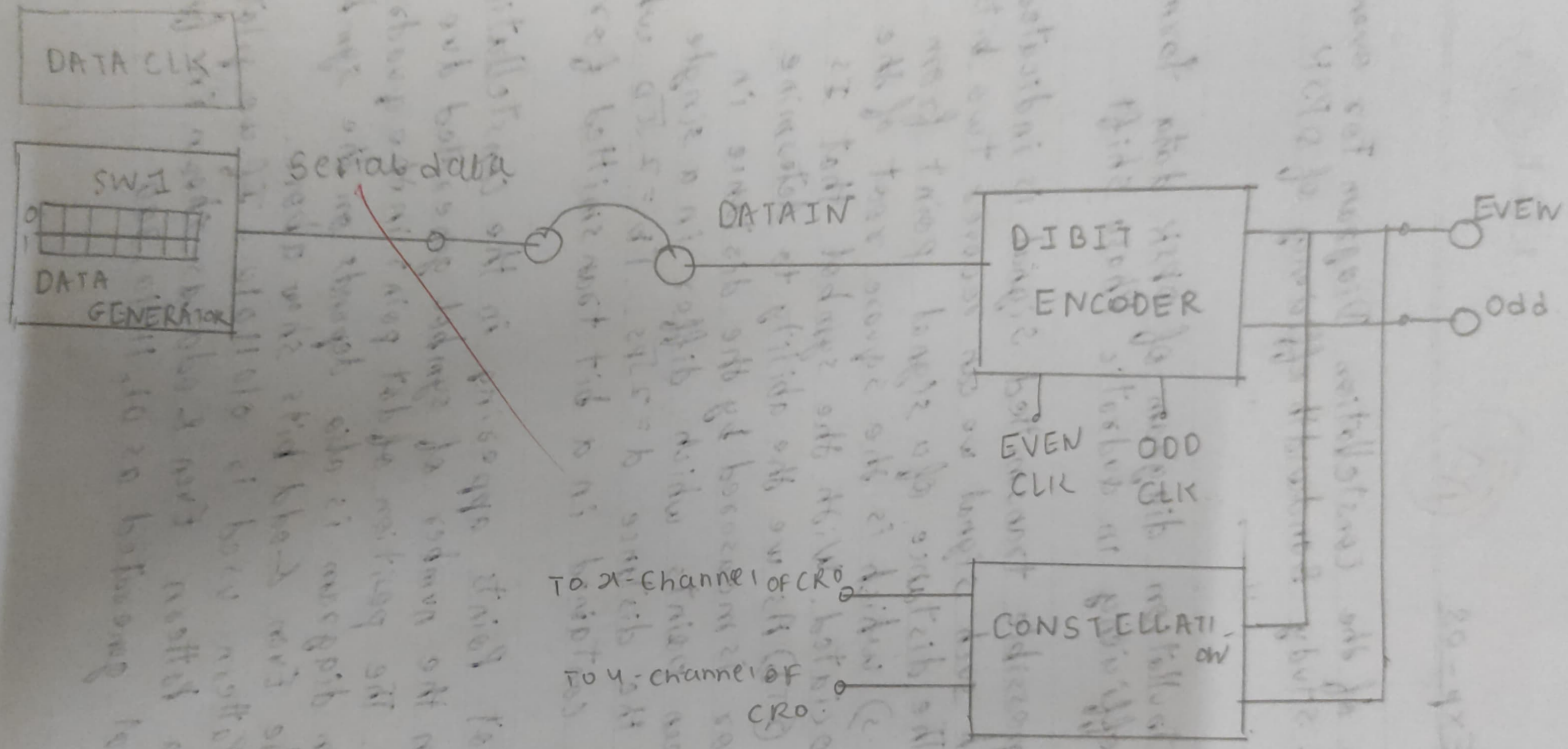
Objective:-

1) To study the Construllation diagram of QPSK data format
2) Study Bandwidth efficieny in Qudratic Phase shift.

Theory:- The four possible transmitted signal is indicated by dots. for each such signal we can recover two bits rater than one. The distance of a signal point from the origin is ($\sqrt{E_s}$) which is the square root of the signal energy associated. With the symbol that Is $E_S = P_S$ H $T_S = P_S$ ($2T_b$) As we the ability to determine a bit without error is measured by the distance in signal space between points which differ in a single bit are seperated by the distance $d = 2\sqrt{P_S}$, $T_b = 2\sqrt{E_D}$ where Eb is the energy contained in a bit transmitted for a Time Tb.

The no. of dot Points appering in the constellation diagram depends on the number of symbol generated due to Even & odd bit. The position of dot point in the quadrant of the constrellation diagram is also depends on the symbol generated due to the Even & odd bits show above.

The data Pattern used is 01011010. If we plot the expected data Pattern Even & odd as shown in fig there are 4 symbol generated as 01, 11, 10, 300


DATA CLK
SW-1
0
1
DATA GENERATOR
Serial data
DATA IN
DIBIT ENCODER
EVEN
Odd
EVEN CLK
ODD CLK
TO X-Channel OF CRO
TO Y-channel OF CRO
CONSTELLATION

Producer :-

1) Refer to block diagram & carryout the following connection & switch settings.

2) Conner Power supply in proper polarity to the Kit DCL - QPSK & switch it on.

3) Select Data Pattern of Simulated data using SW1

4) Connect serial Data generated to Data ~~using~~ In of the DIBIT coder.

5) Connect the coded data NRZ-L Data to the Data In of DIBIT conversion.

6) Observe the coded LP Signal Even data & odd data bit oscilloscope.

7) Connect Even data to X-channel of CRO & odd data bit to Y channel of CRO.

8) Put the CRO in X-Y mode observe the constellation diagram in oscilliscope shown in fig 3.2 & refer theory to observe different type of constellation diagram.

CLOCK
t
EVEN CLOCK
t
ODD CLOCK
t
DATA
t
ODD
t
EVEN
01 01 01 11 10 10 10 00 01 01 01 01 01 11 10 10
Generated symbol