Alexandria University
Faculty of Engineering
Electrical Engineering Department

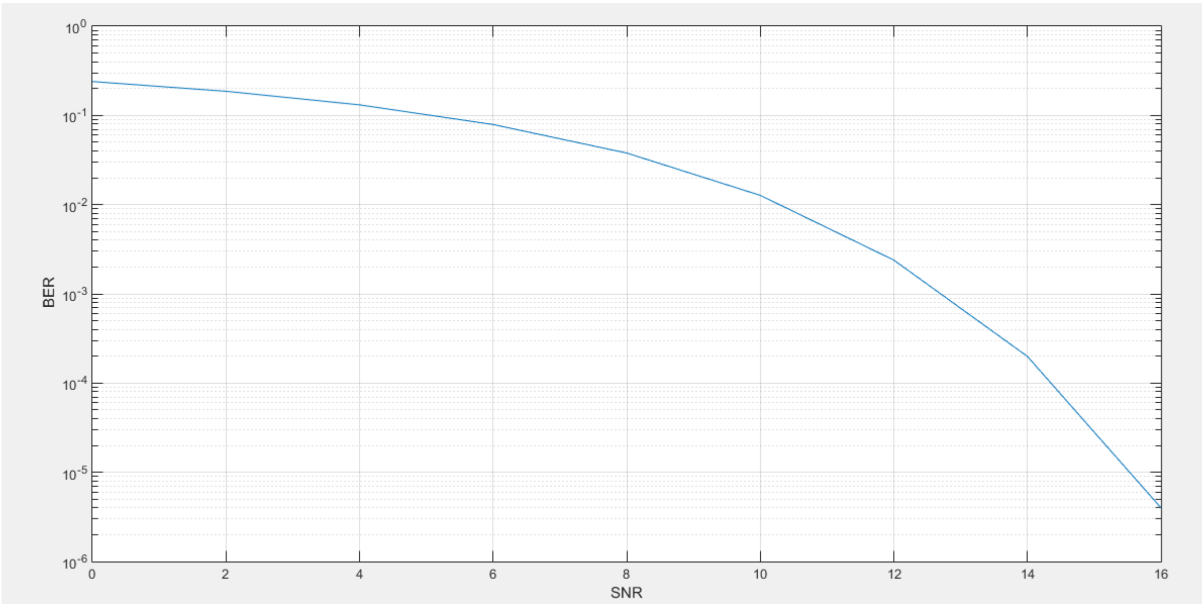
Digital Communications Project

Name	ID
Islam Ayman Mohamed	42
Safaa Mohamed Gamal	100
Abd El-Rahman Ibrahim Rezk	102
Mostafa Mahmoud Khalil	197
Hagar Mohamed Gamal	210
Yaman Abd El-Salam Kanaan	221

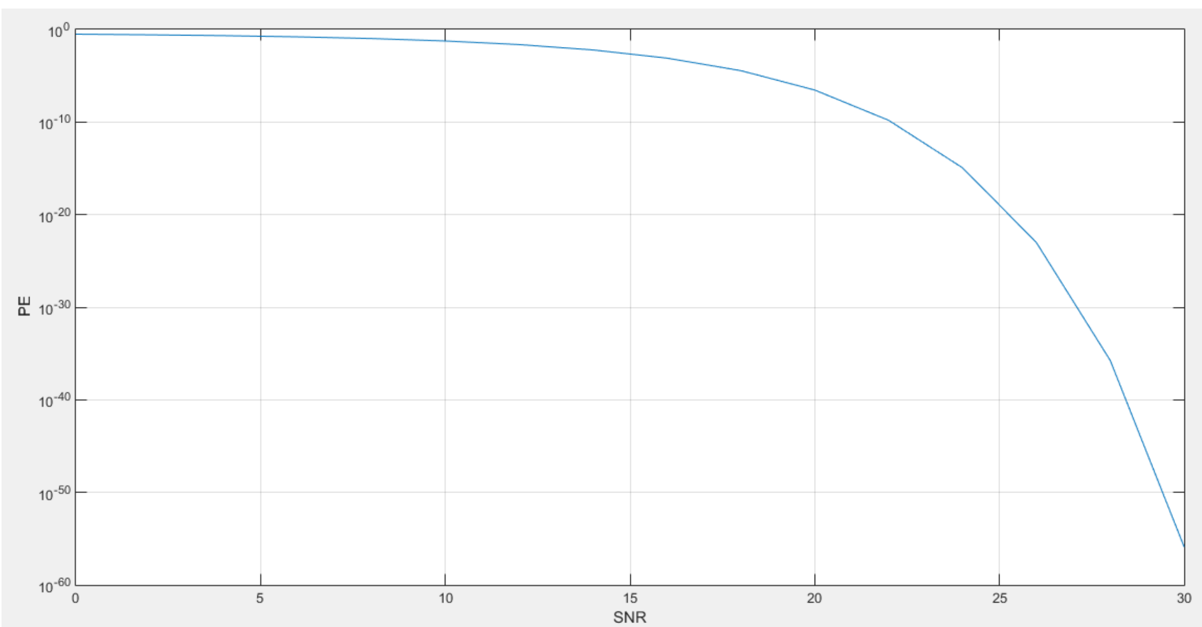
Submitted to: Eng. Hossam Hassan

Experiment 1

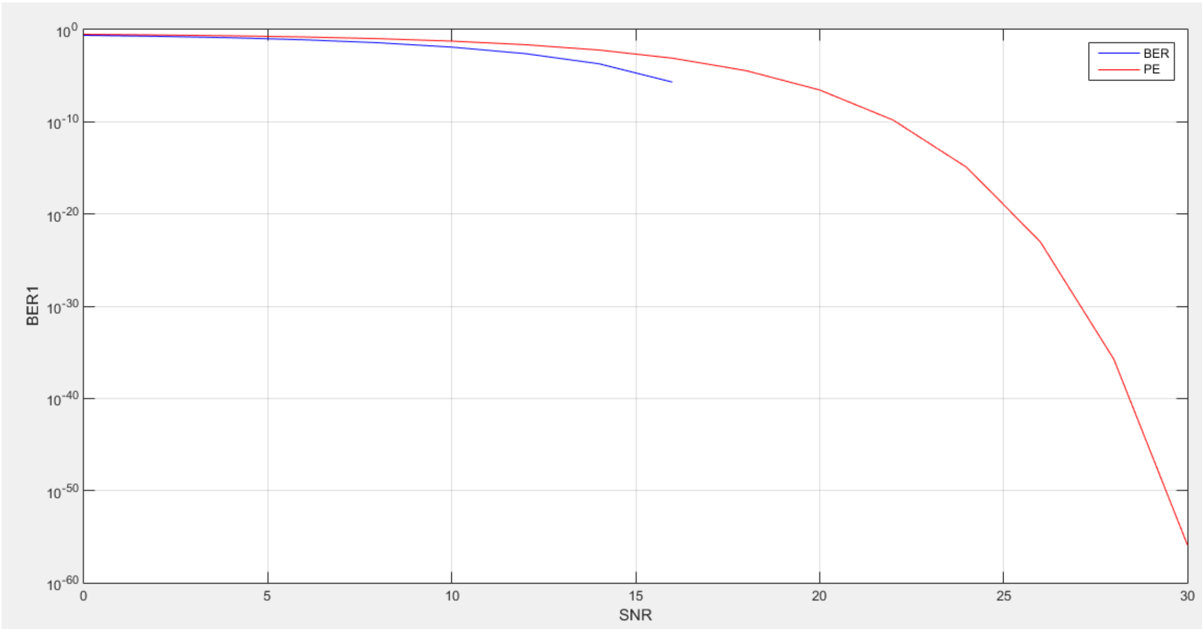
1) BER curve against SNR:



2) Pe curve against SNR:



3) Comparison Pe and BER with SNR



4) At which value of SNR the system is nearly without error?
The system started to be without error at SNR = 16

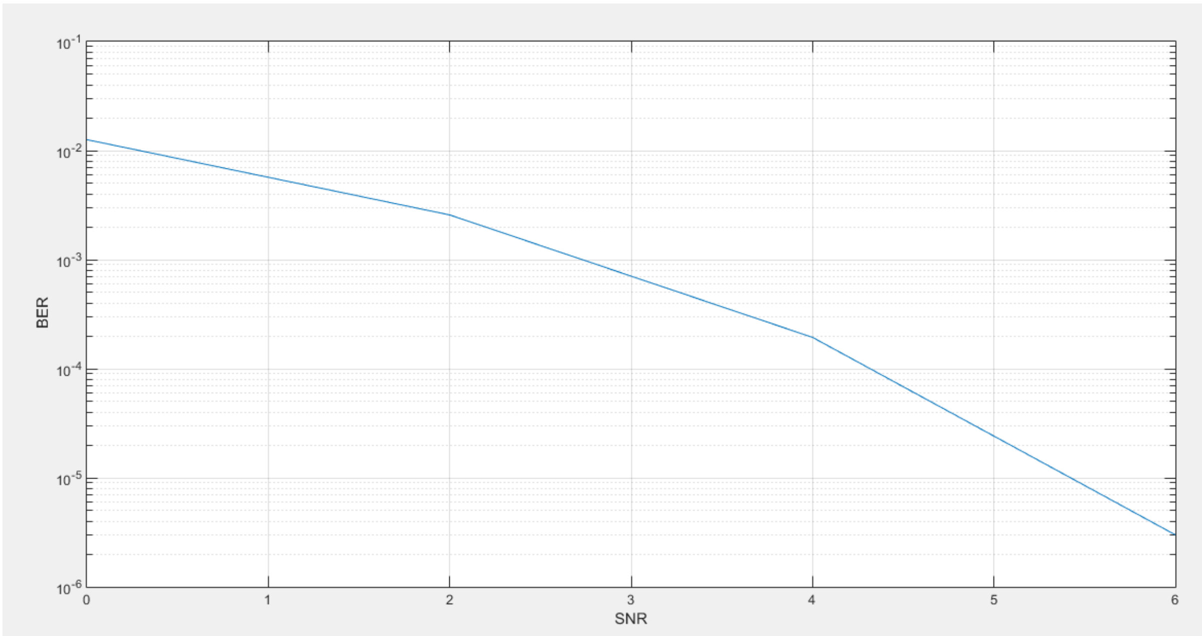
5) Identifying meaning of ‘measured’ field in (3)?
Awgn function measures the power of the signal before adding noise according to SNR

6) Calculation of The Transmitted power:

```
P_Tx = (1/NumberOfBits) * sum(BinaryBits.^2)
P_Tx =  0.5001
```

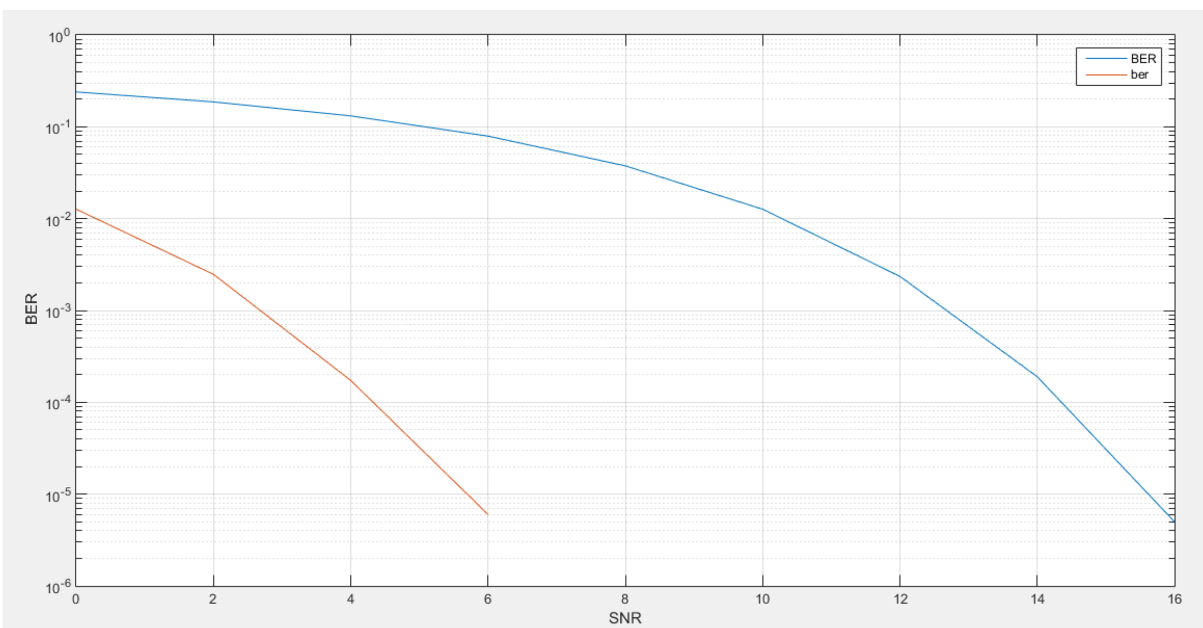
Experiment 2

1) BER curve against SNR of MF & correlator:



Comment: Matched Filter and Correlator both have the same output over all points of SNR.

Comparison of MF and Correlator with Simple Detector:



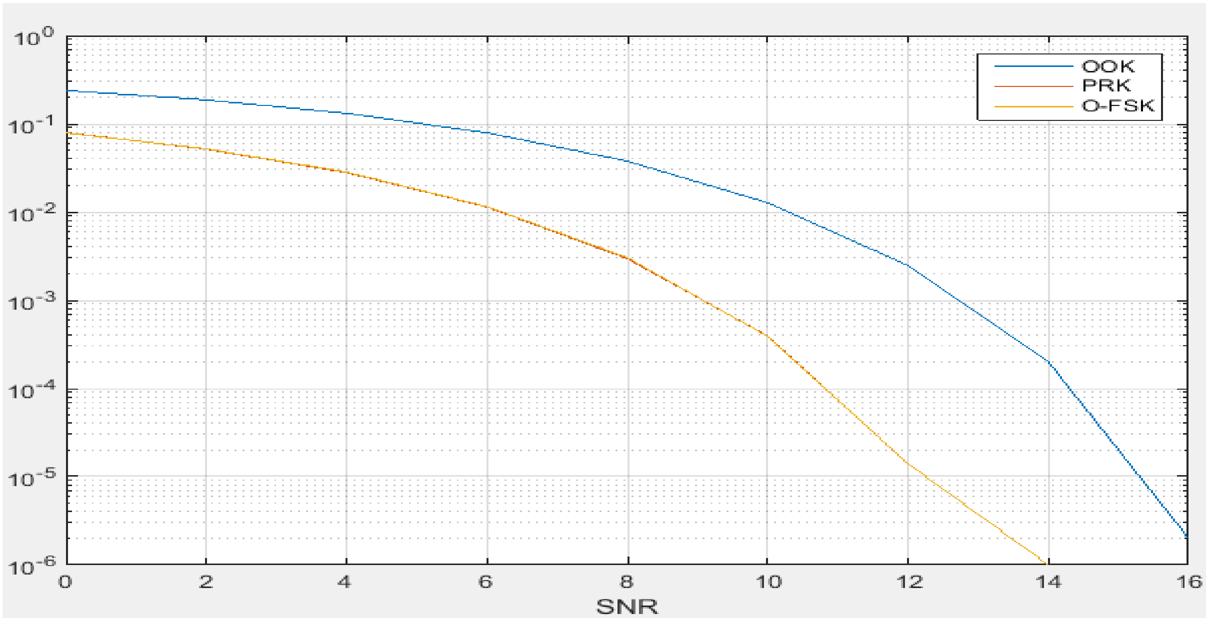
Comment: MFs & correlators are designed to provide Maximum SNR, then minimize the BER.
The Simple detector started to be without error at SNR = 16; but the matched filter & correlator started to be without error at SNR = 6

2) Calculation of The Transmitted power:
   tansmisttedPower = (1/dataLength) * sum(data.^2);
   tansmisttedPower = 0.4991

3) At which value of SNR the system is nearly without error (for the given frame)?
   The system started to be without error at SNR = 6

Experiment 3

1) BER of the three modulation schemes (ASK – PSK - FSK)



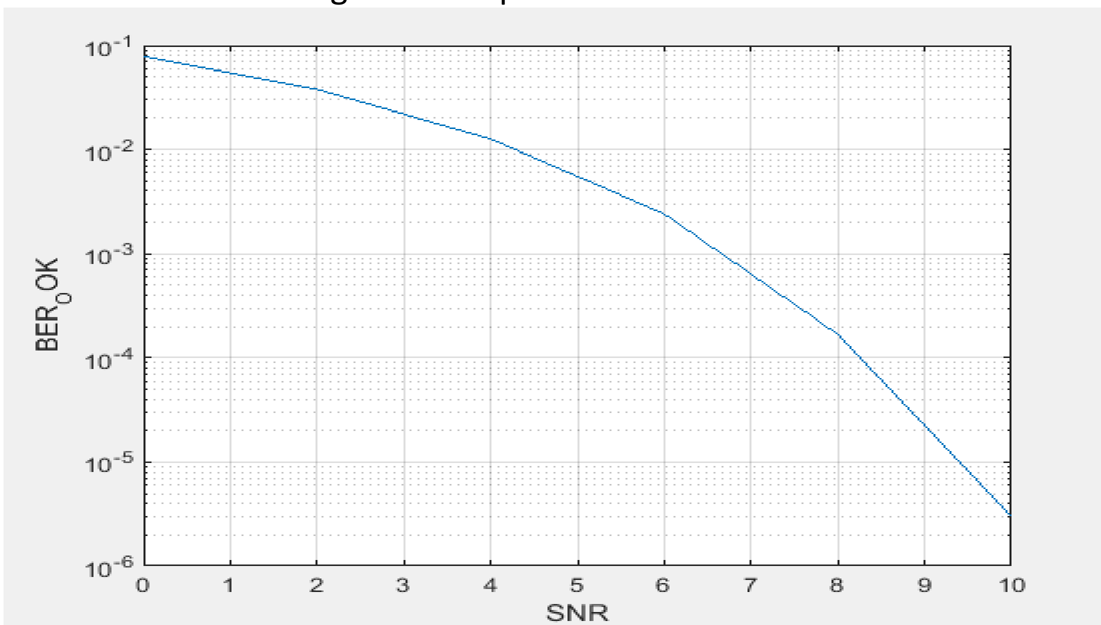
2) The PRK and orthogonal-FSK both has the best performance over OOK, as the BER of them is lower than the BER of OOK.

3) We can detect when the system began with nearly zero error by some line of codes that will make a breakpoint to tell when the system started to be without errors

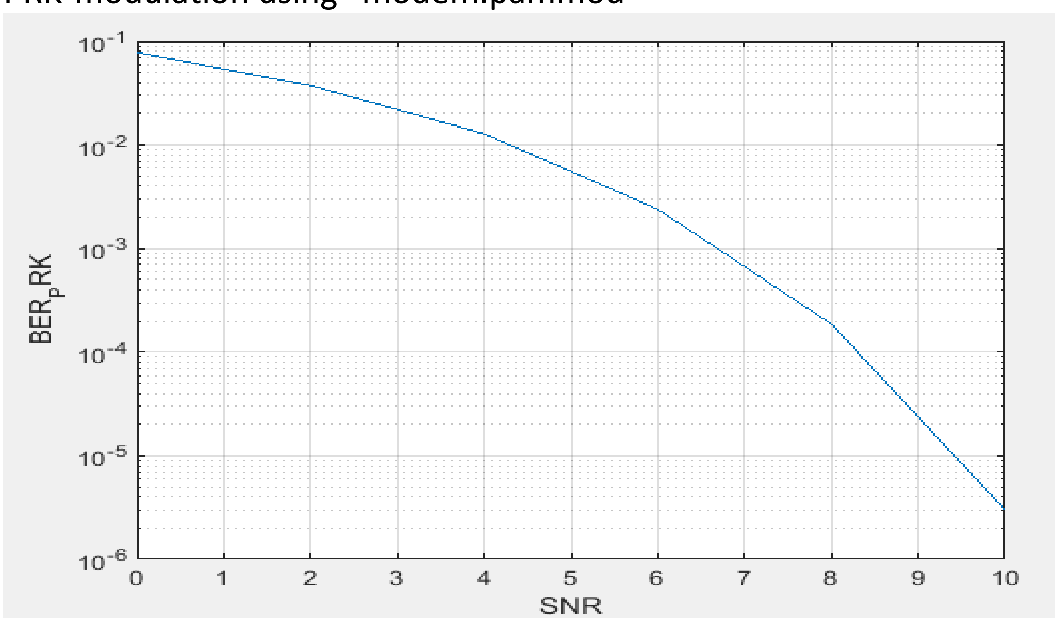
```
The OOK system started to be without error @ SNR = 18
The PRK system started to be without error @ SNR = 14
The o-FSK system started to be without error @ SNR = 14
>>
```

BONUS

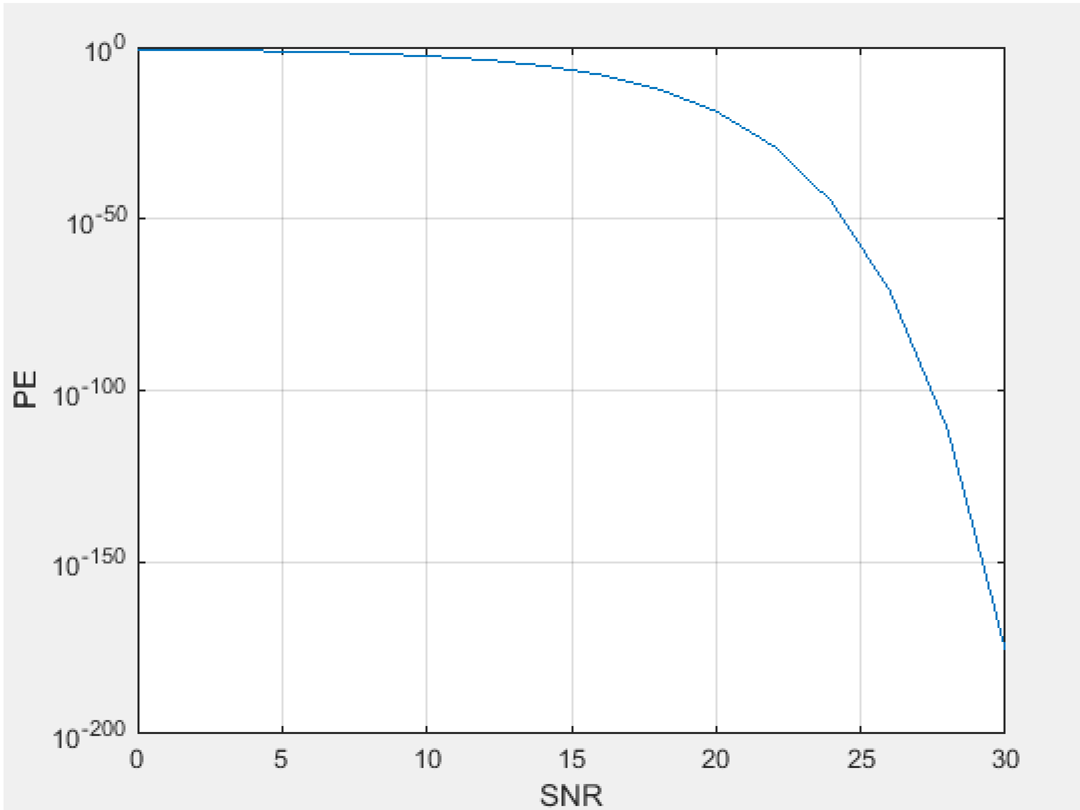
1) OOK-modulation using “modem.pammod”



2) PRK-modulation using “modem.pammod”



3) 16 QAM modulation

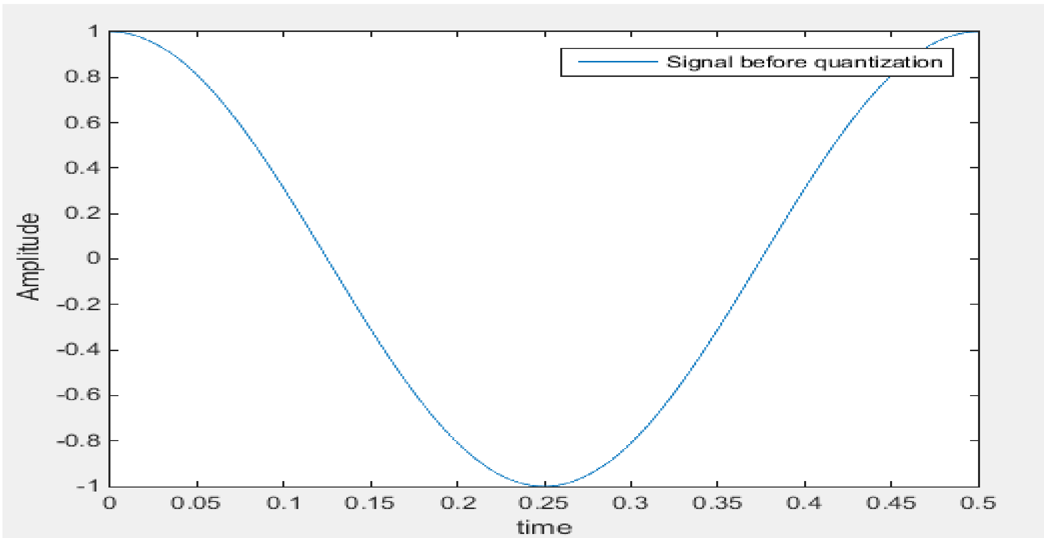


Has high probability of error.

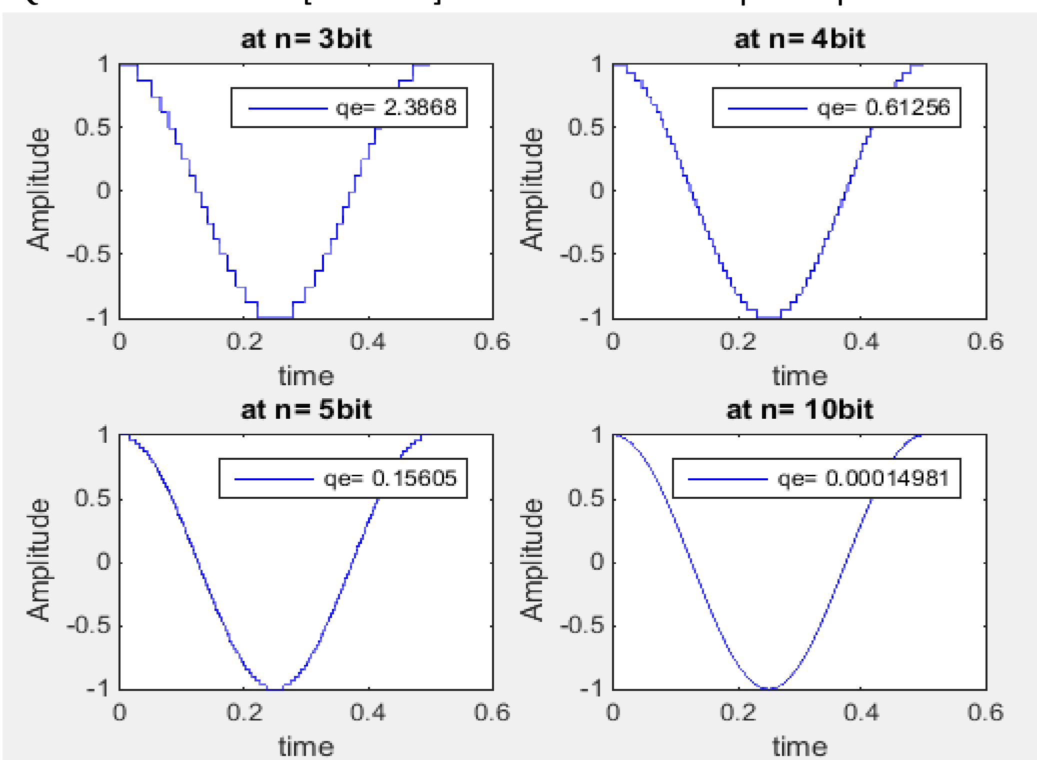
Experiment 4

Part1 (Quantization Error)

1) Original signal before quantization:



2) Quantization in n =[3 4 5 10] with there mean square quantization error:



As we can see, increasing the levels of quantization, decreases the quantization error.

3) Signal in binary form:

First to convert every level of the signal to binary code we must shrink the array of the quantized signal as it has repeated values.

```
index=30;
valuesOfQuantization(1:(length(y_quantized)-1)/2)=zeros(1,(length(y_quantized)-1)/2);
valuesOfQuantization(1:(length(y_quantized)-1)/2)=y_quantized(1:(length(y_quantized)-1)/2);
for l=1:3   %shrinking the array to be with single values, not repeted values
    for c=1:length(valuesOfQuantization)
        for j=1+c:length(valuesOfQuantization)
            if(valuesOfQuantization(c)==valuesOfQuantization(j))
                valuesOfQuantization(j:end-1)=valuesOfQuantization(j+1:end);
                j=j-1;
                valuesOfQuantization(end)=index;
                index=index+1;
            end
        end
    end
end
```

Then redistribute those binaries to the original signal so we get:

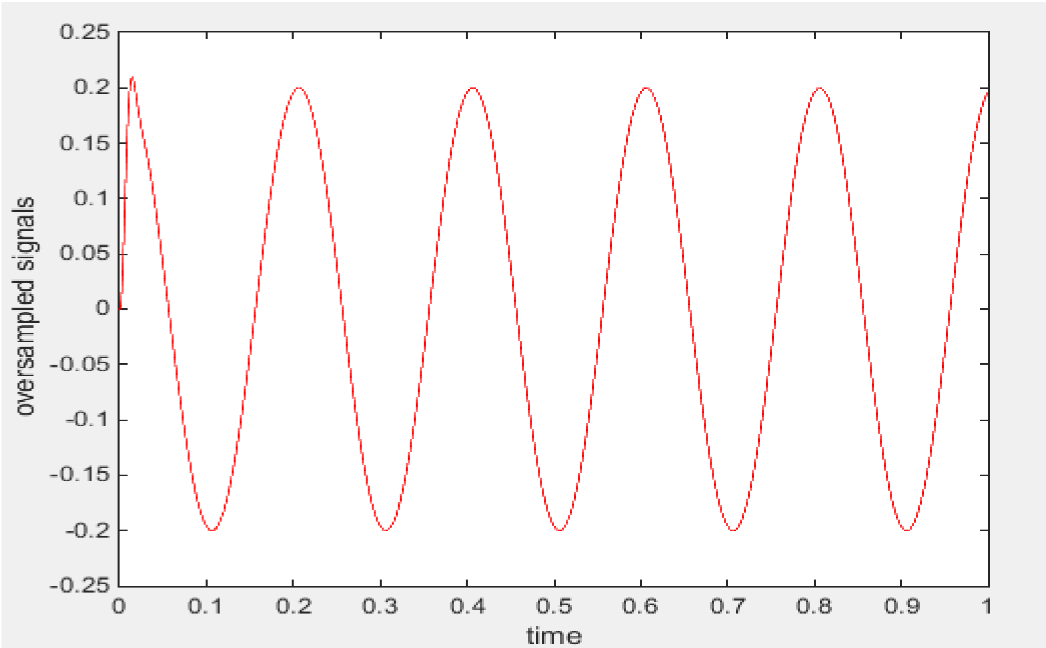
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0

the column in the left is the voltage of each level but repeated, and the rest is the binary.

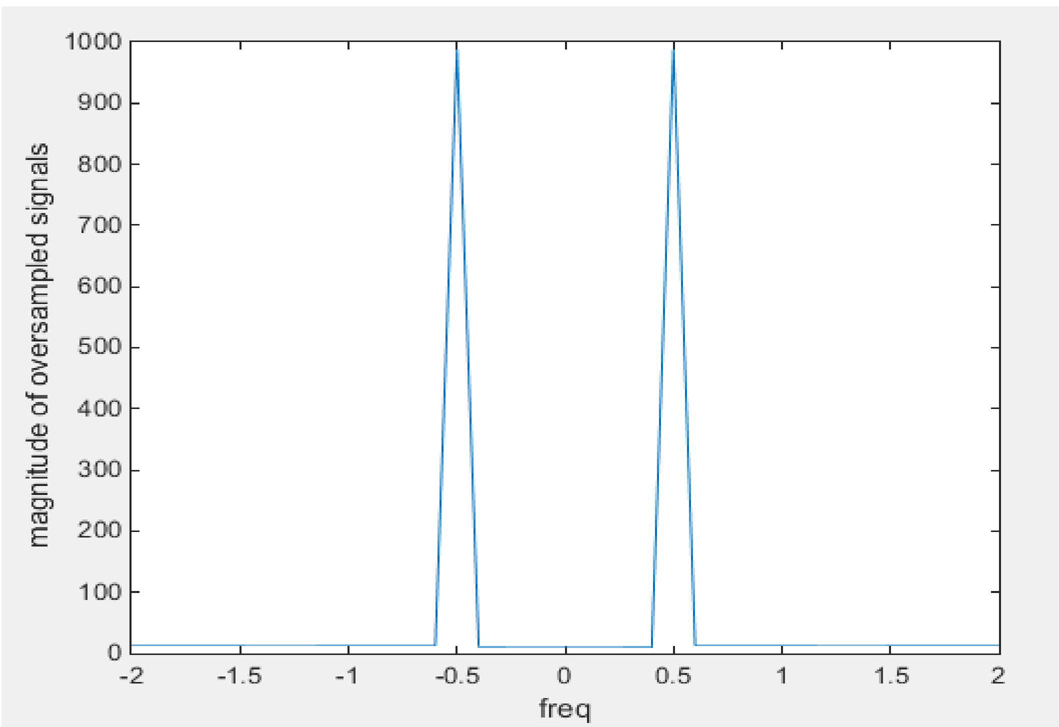
Part2(Sampling Distortion)

1) reconstruction from oversampling :

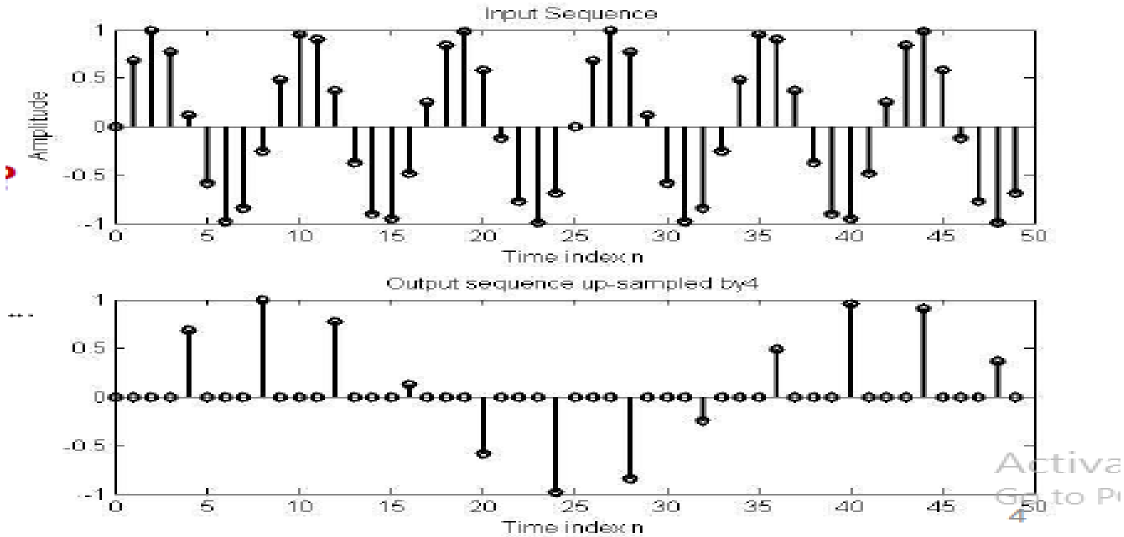
    −in time domain:



−in frequency domain:



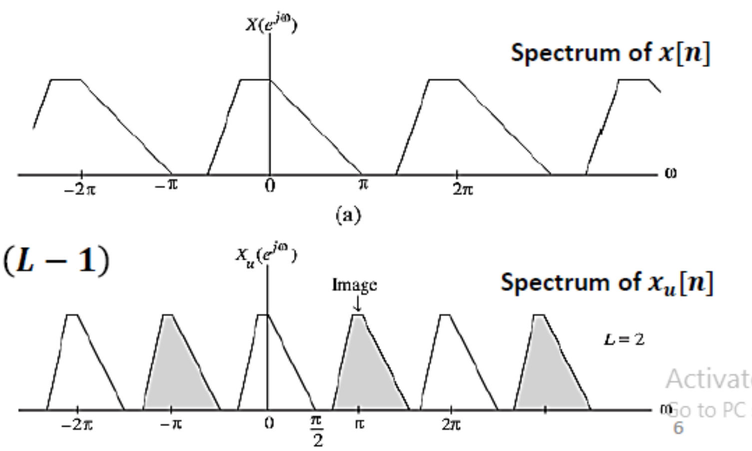
As we can see frequency of the signal is 5Hz, but due to the interpolation process (upsampling + LPF), the bandwidth and the frequency will shrink by number of oversampling.



On the unit circle

$$\boxed{X_u\left(e^{j\omega}\right) = X\left(e^{j\omega L}\right)}$$

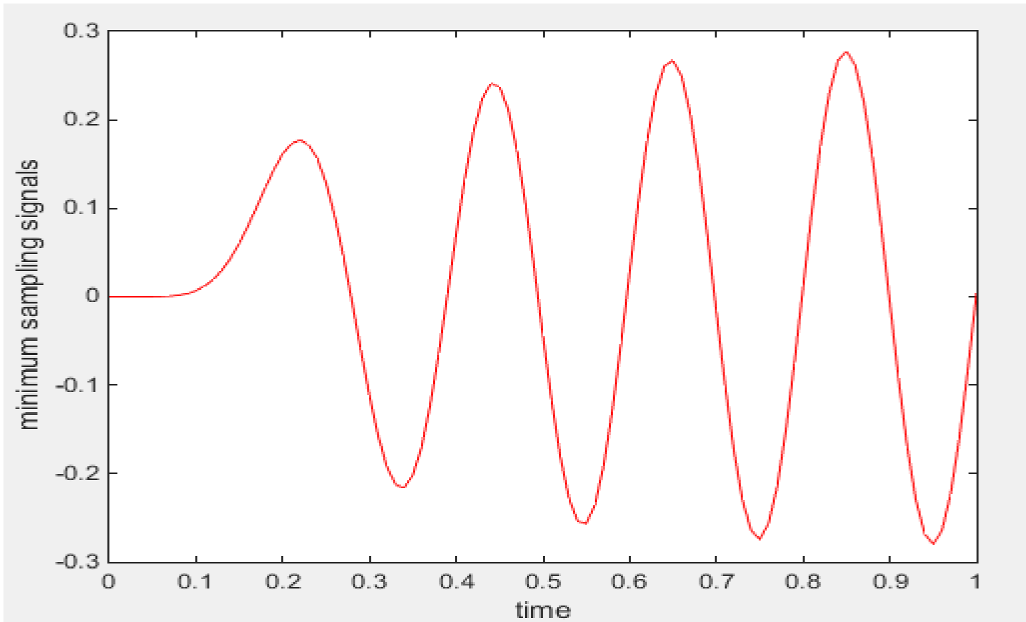
up-sampling by L introduces $(L-1)$
Images in the spectrum



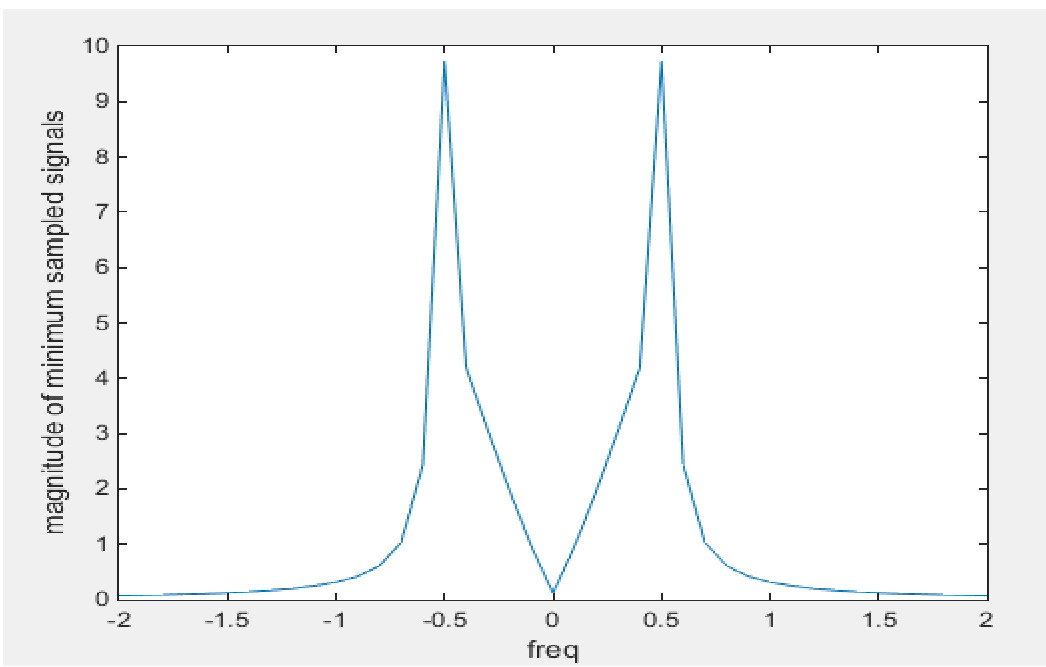
11/18/2017

2) reconstruction from minimum sampling :

   −in time domain:



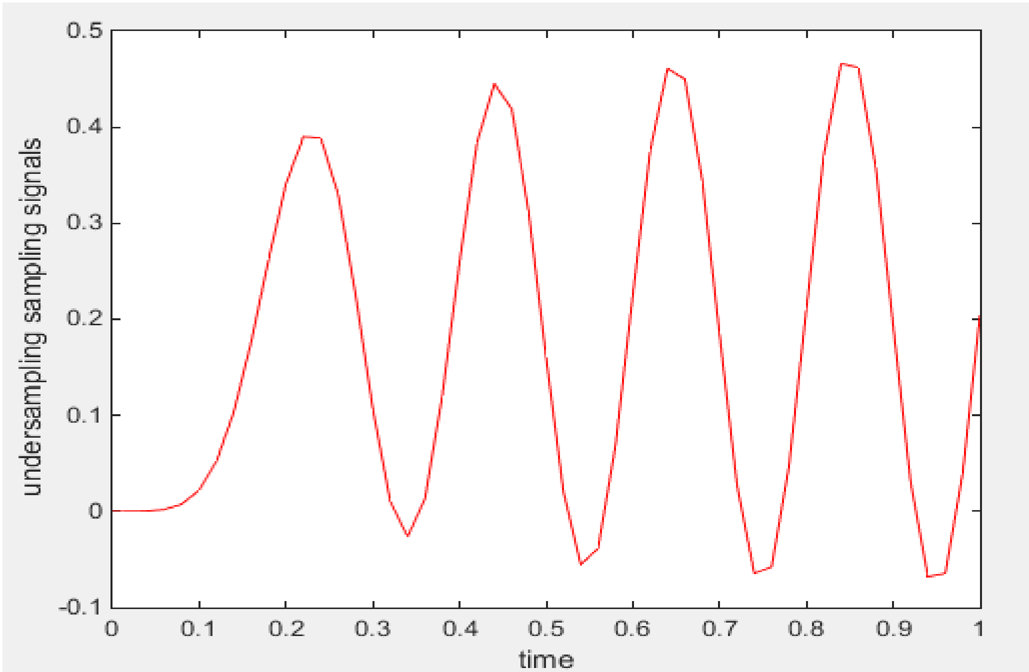
−in frequency domain:



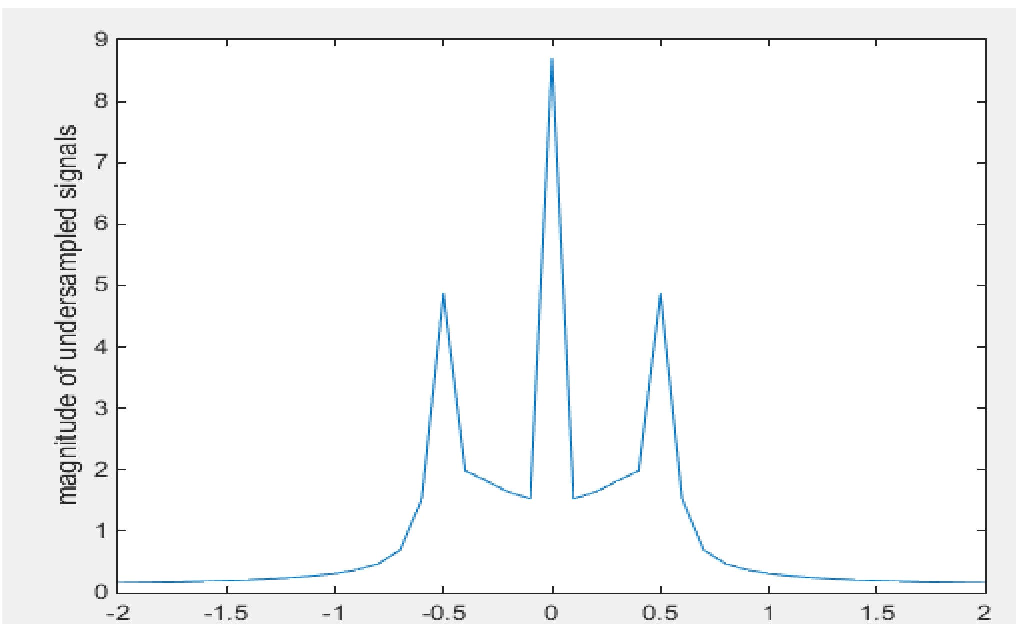

```
% minimum sampling Fn = 2*Fm
```

) reconstruction from under sampling :

−in time domain:



−in frequency domain:

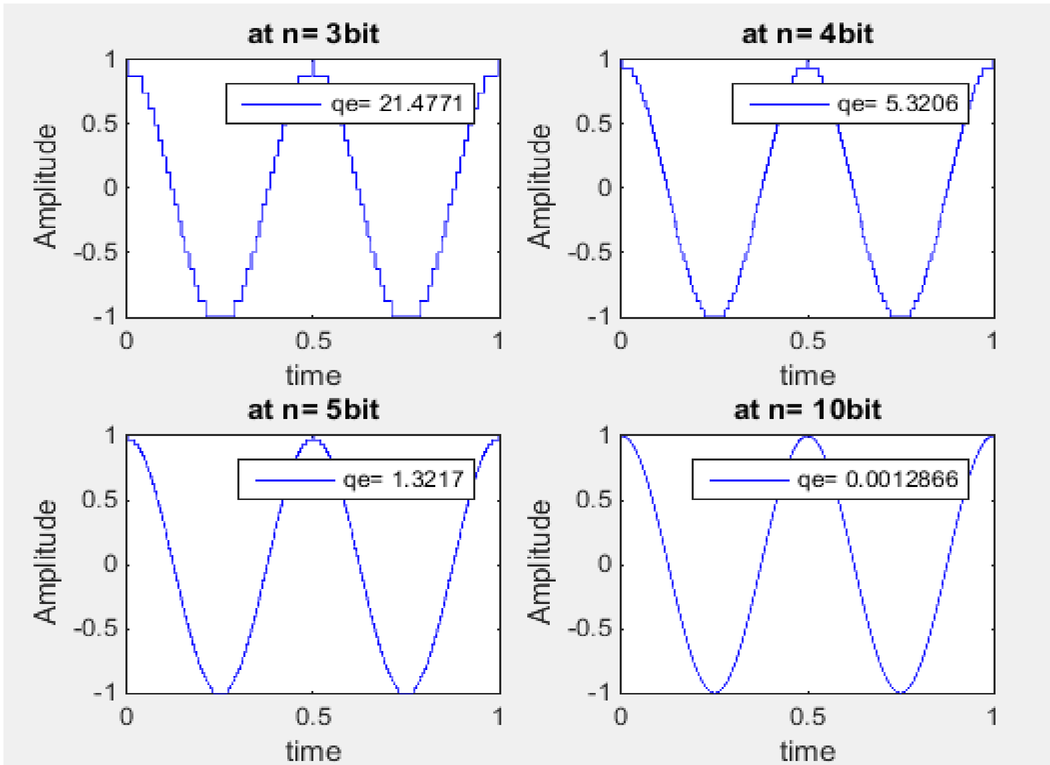



```
% undersampling sampling Fn < 2*Fm
```

***Bonus part**

i) Quantization with "quantize":

-Quantization in n =[3 4 5 10] with there mean square quantization error:



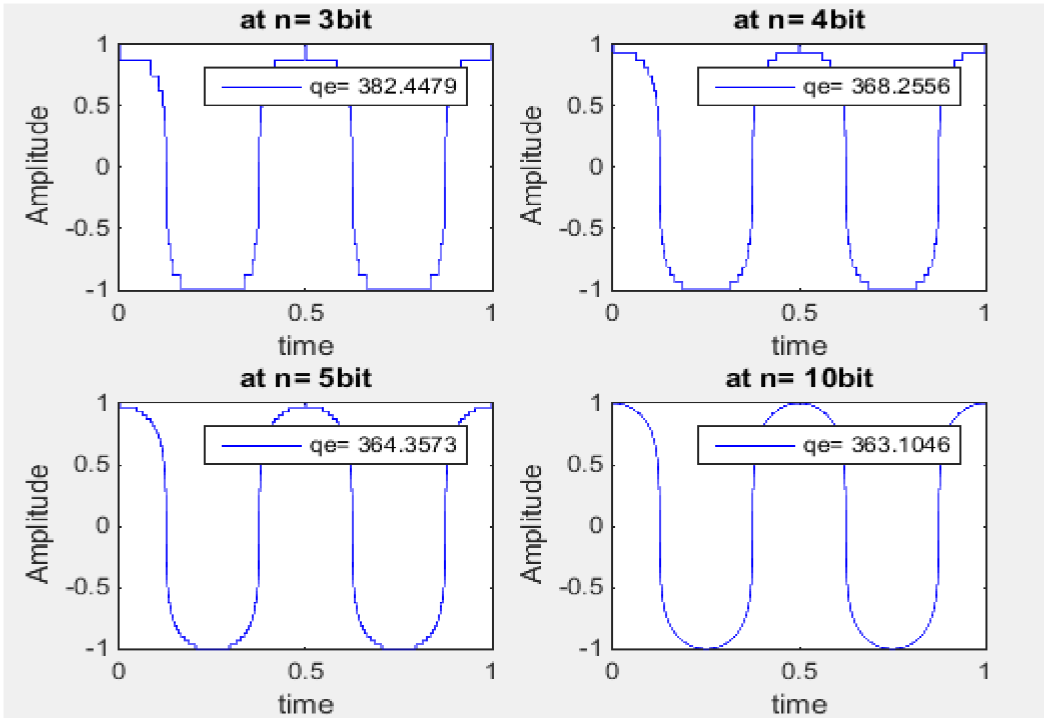
- Signal in binary form:

Is the same as above

0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0

ii) Non-uniform Quantization with "compand":

-Quantization in n = [3 4 5 10] with their mean square quantization error:



As obvious the quantization error is large.

-Signal in binary form:

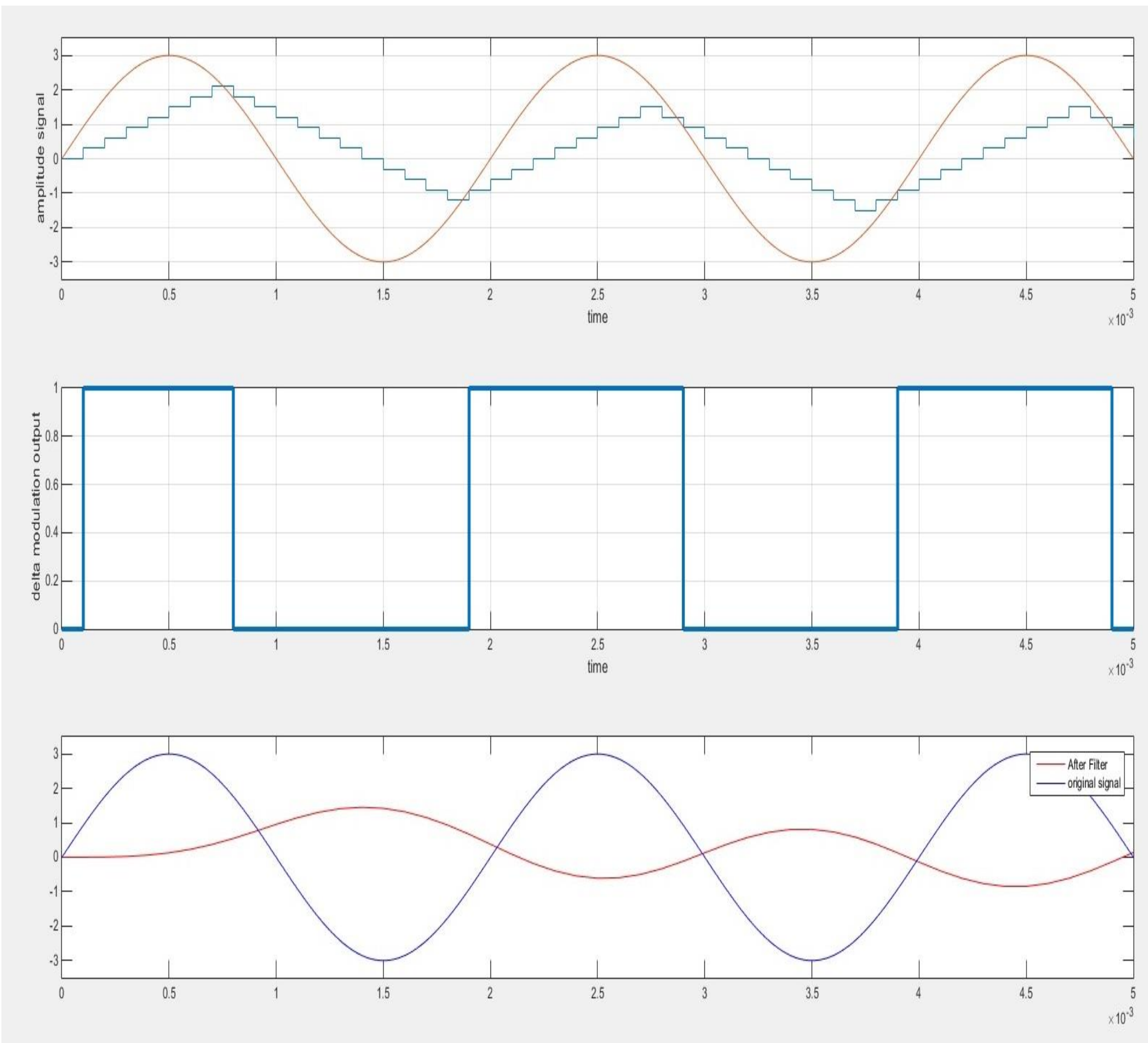
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8125	1.0000	0	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0
0.8750	0	1.0000	1.0000	1.0000	1.0000	0

Also as the binary bits above.

Experiment 5

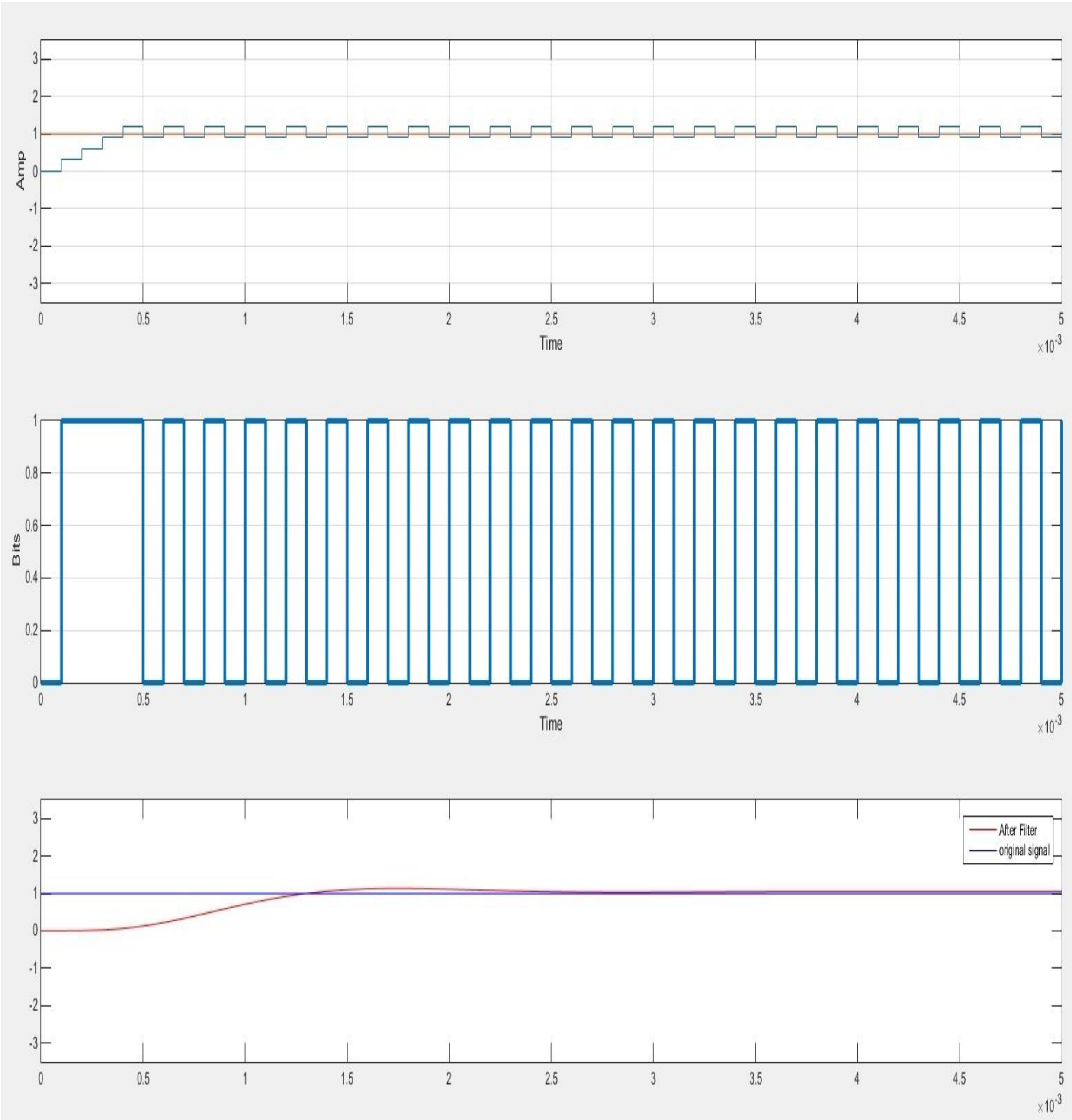
Simulation parameters: Fm=500Hz , Fs(signal)=200000Hz ,Delta=0.3, TsScale=Fs(signal)/Fs(delta modulated signal) =20

Sine wave:



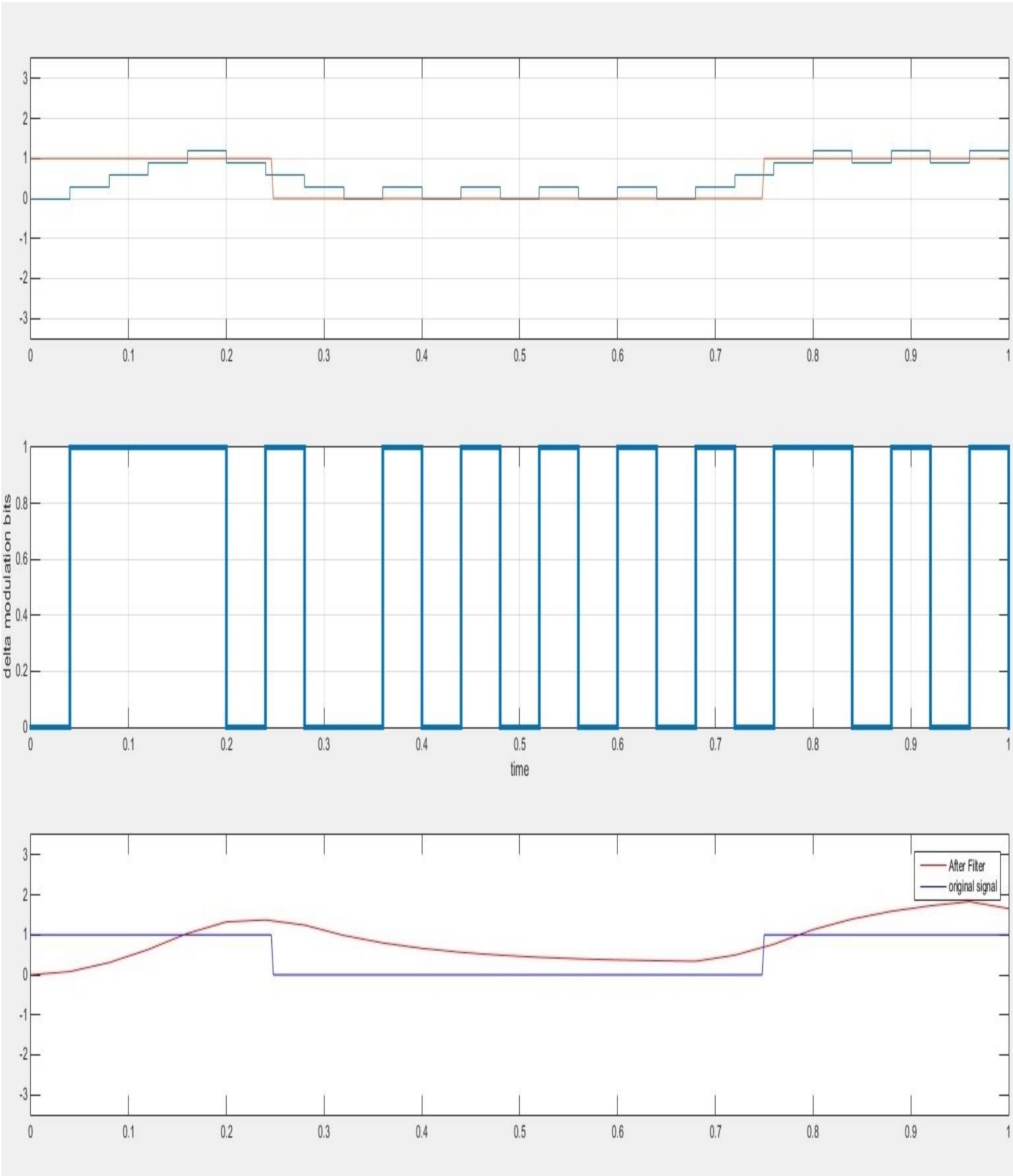
Square error=3.429336e-02

DC Signal:



Square error=2.5e-03

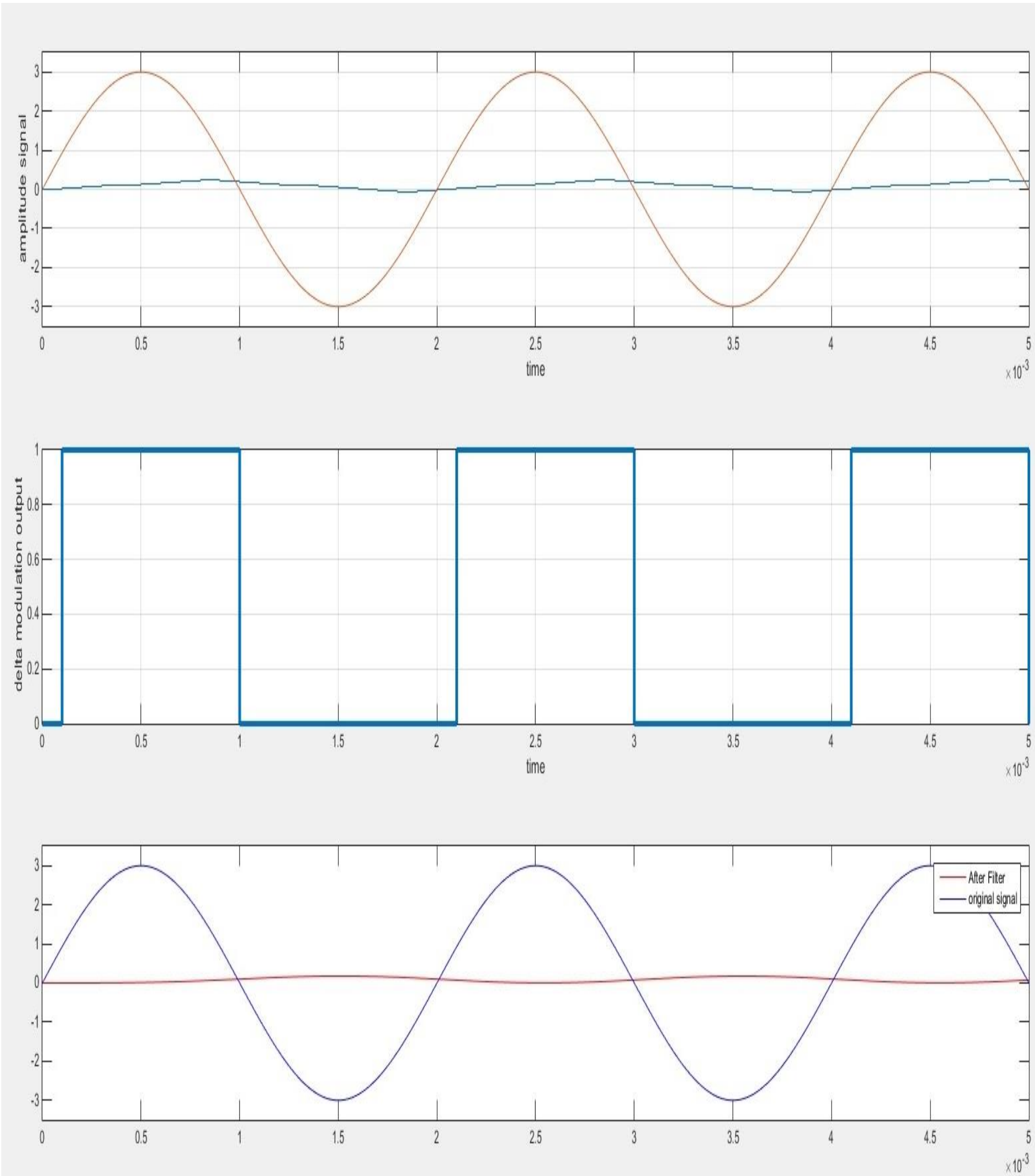
Square wave:



Square error=6.893950e-01

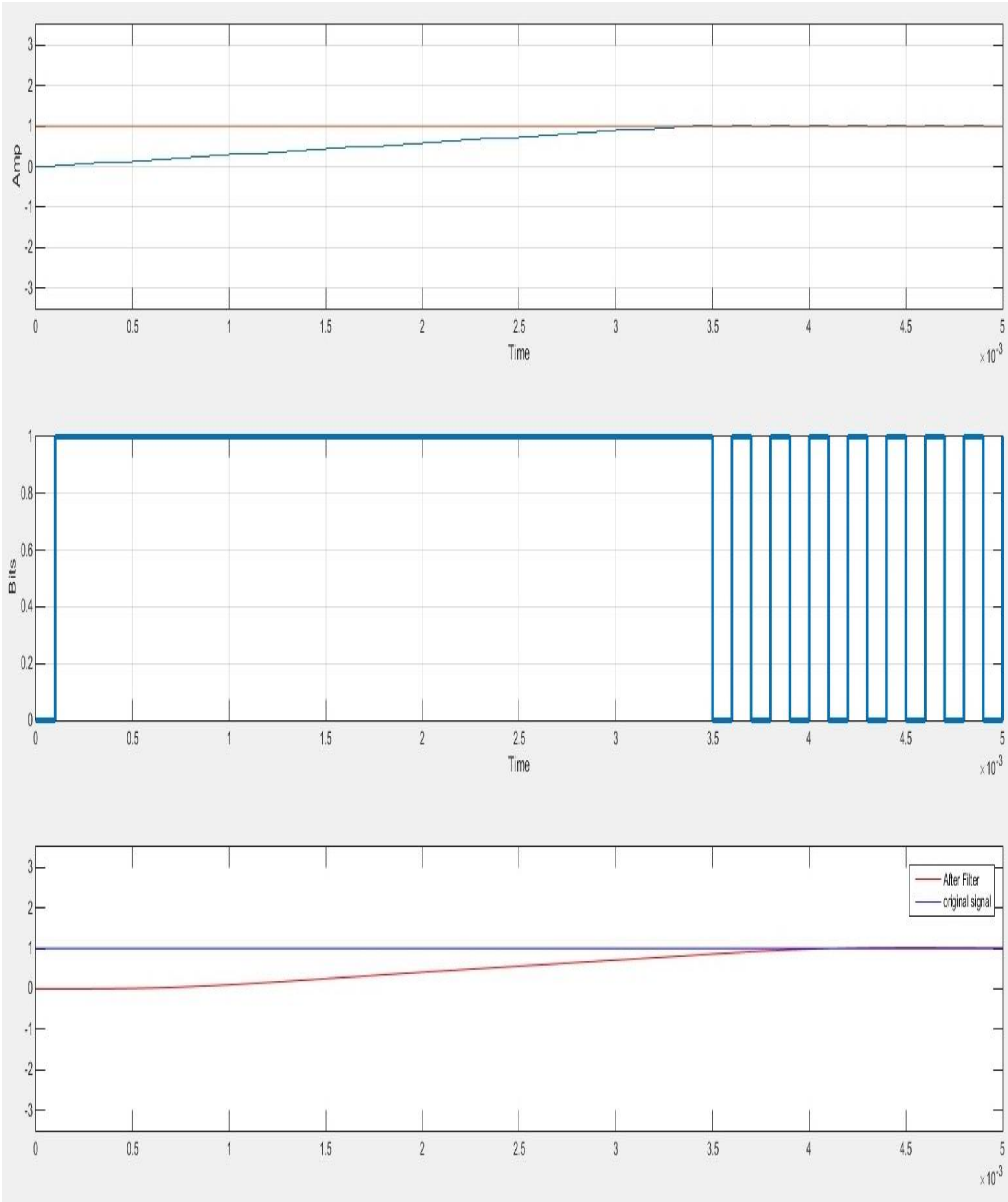
For deltanew=0.1deltaold:

Sine wave:



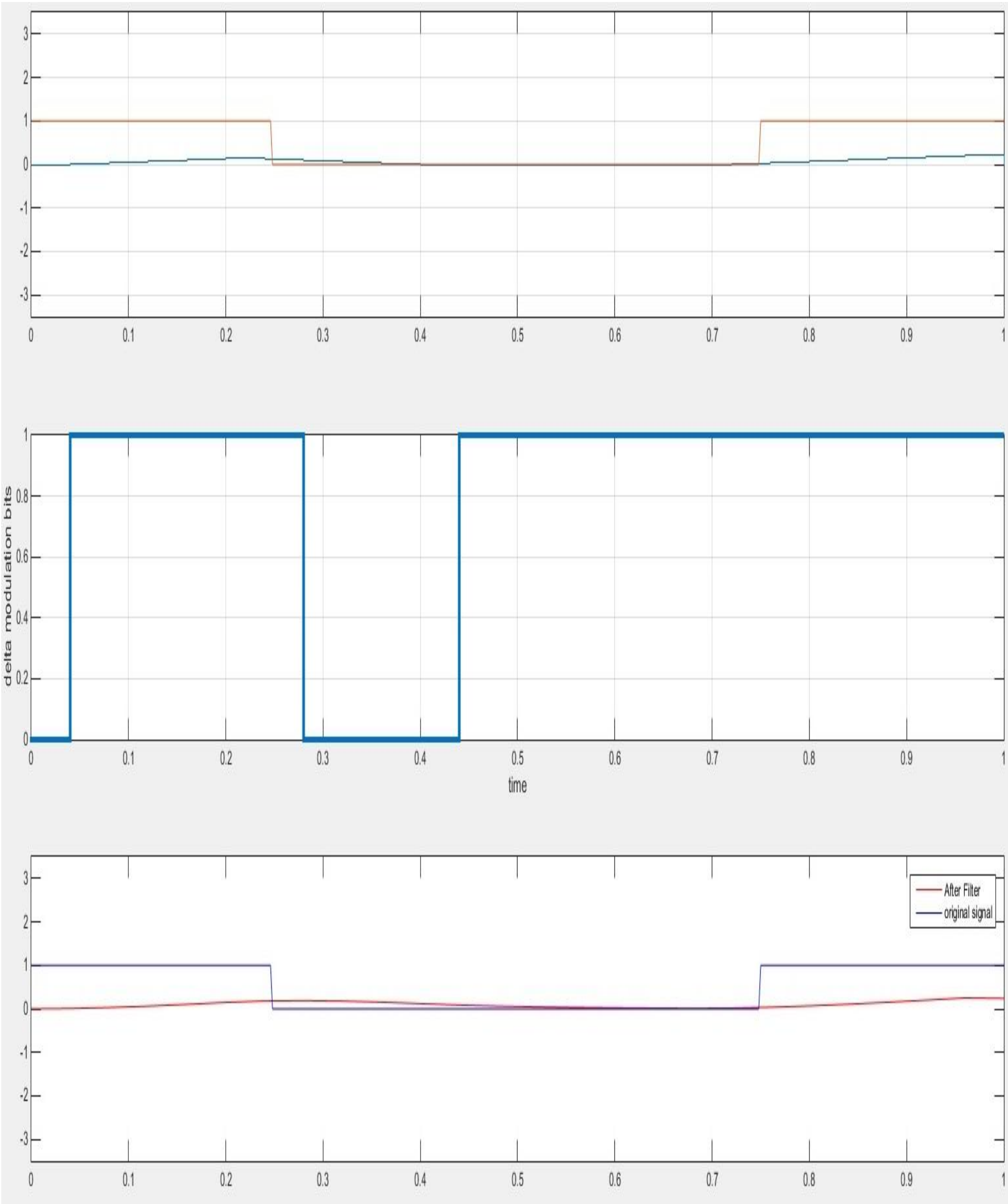
Square error=3.130668e-02

DC Signal:



Square error=2.5e-05

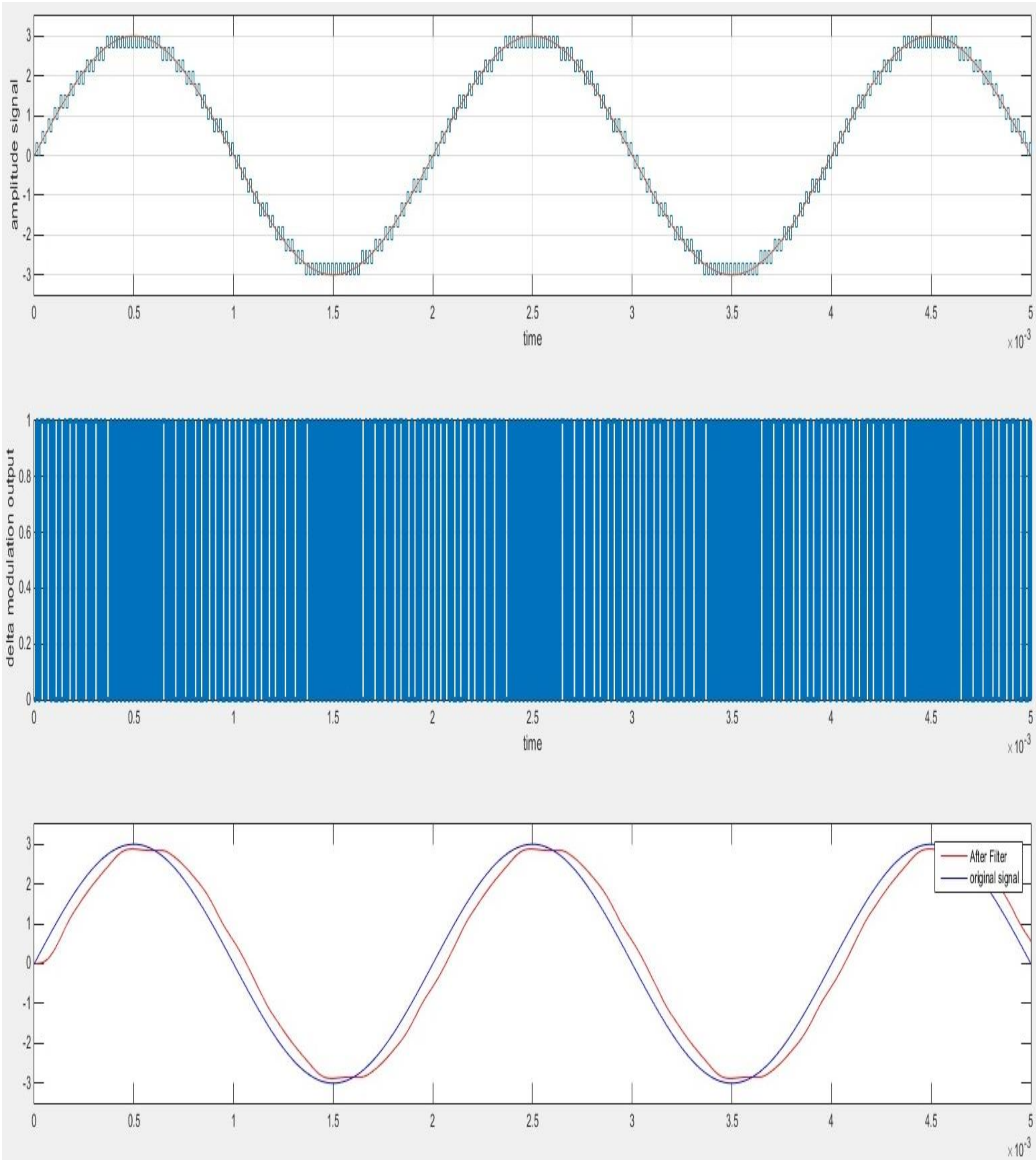
Square wave:



Square error=5.591981e-1

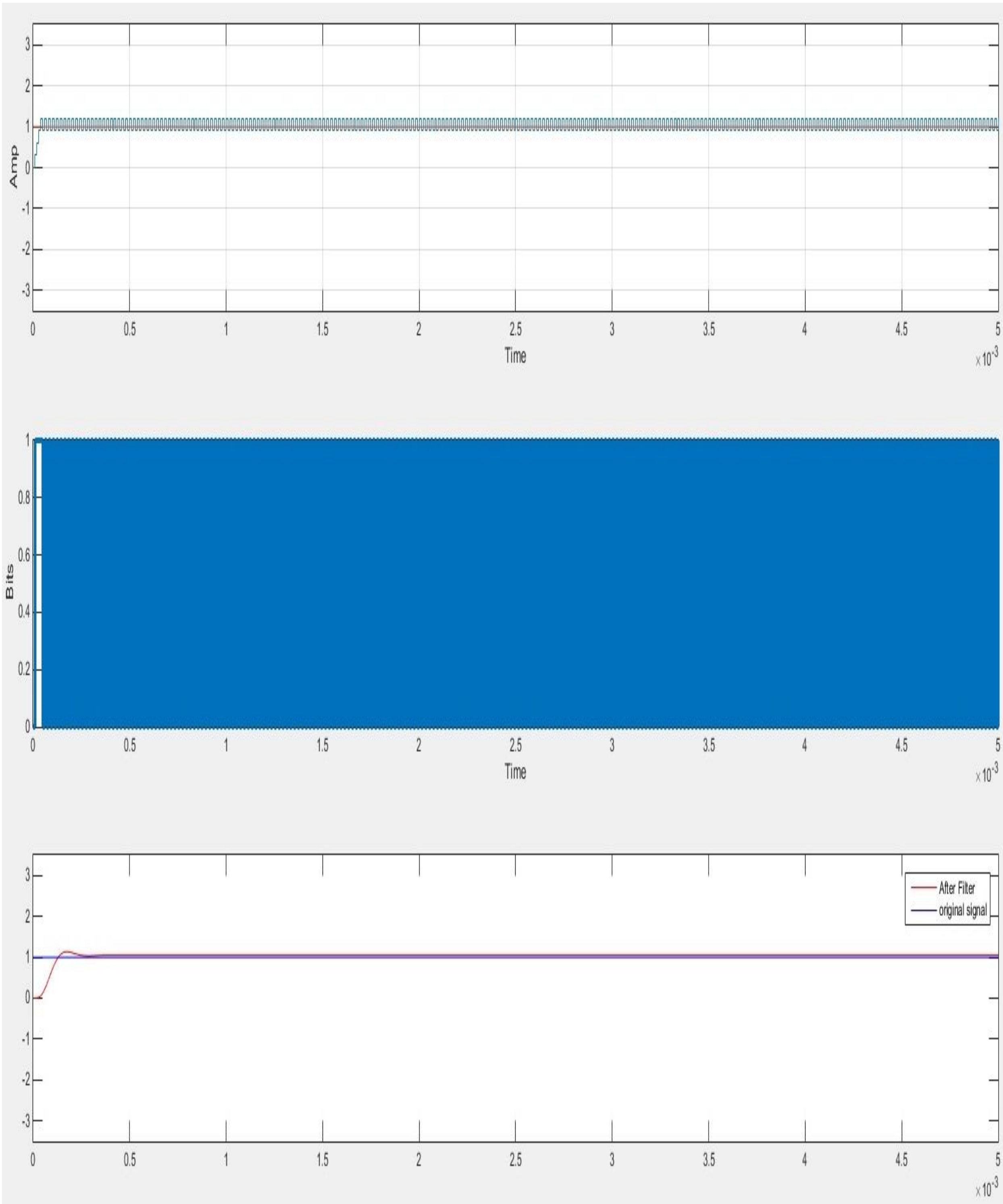
For Ts_{new}=0.1Ts_{old}:

Sine wave:



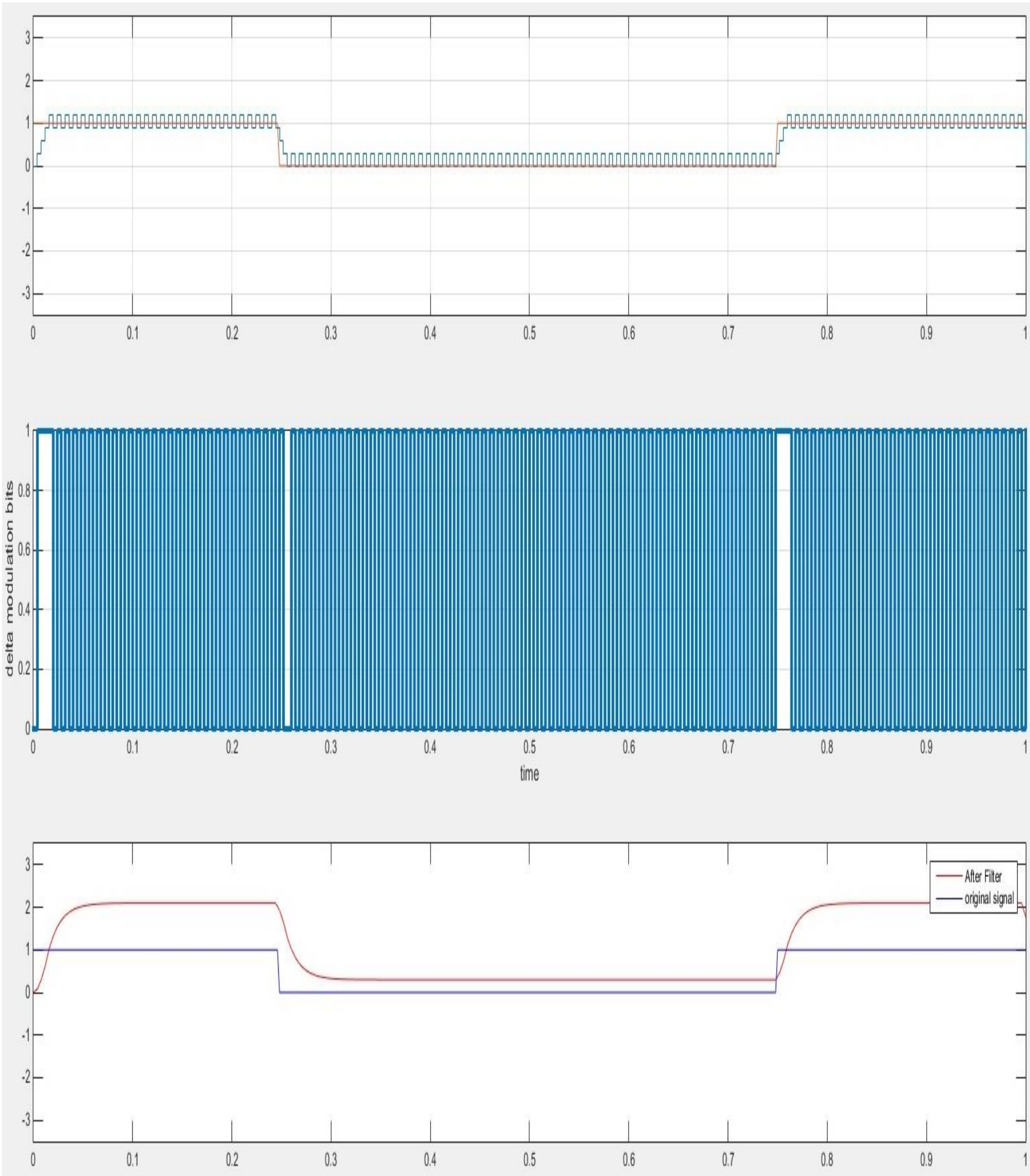
Square error=3.624149e-01

DC Signal:



Square error=2.5e0-3

Square wave:



Square error=1.21e+00

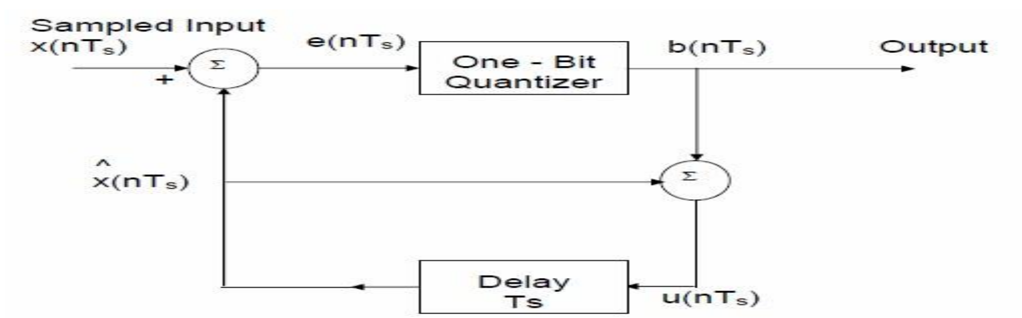
*The lowest type of signal which has the lowest error is the DC signal because signal-to-noise ratio is increased in the low frequency area of the spectrum

Modified delta modulation technique:

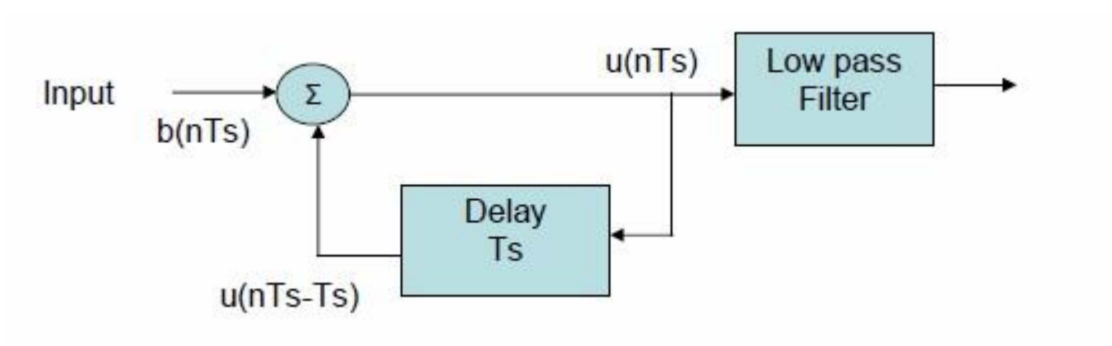
To reduce the Slope overload and Granular noise we use adaptive Delta Modulation (ADM).

To overcome the quantization error due to slope overload distortion and granular noise, the step size (Δ) is made adaptive to variations in input signal x (t). Particularly in the step segment of the x (t), the step size is increased. Also, if the input is varying slowly, the step size is reduced.

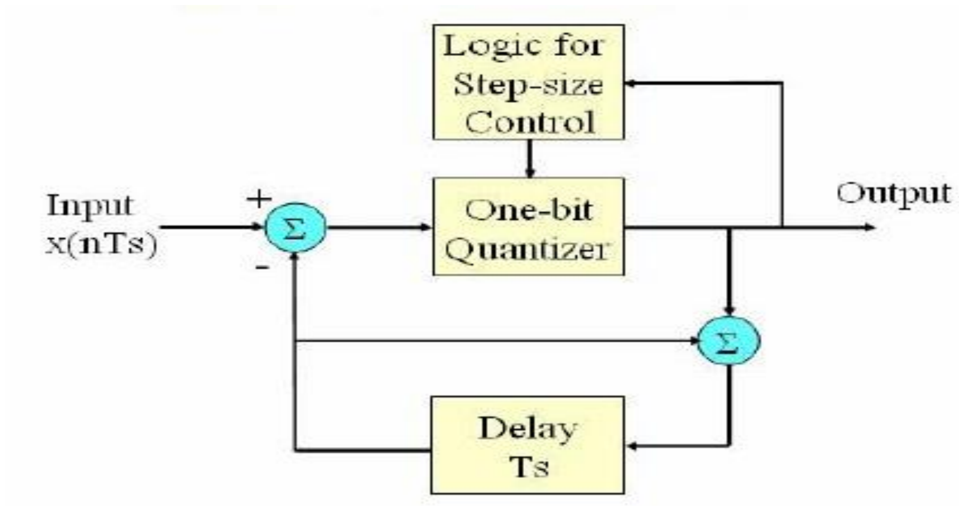
While Scheme for delta modulation transmitter:



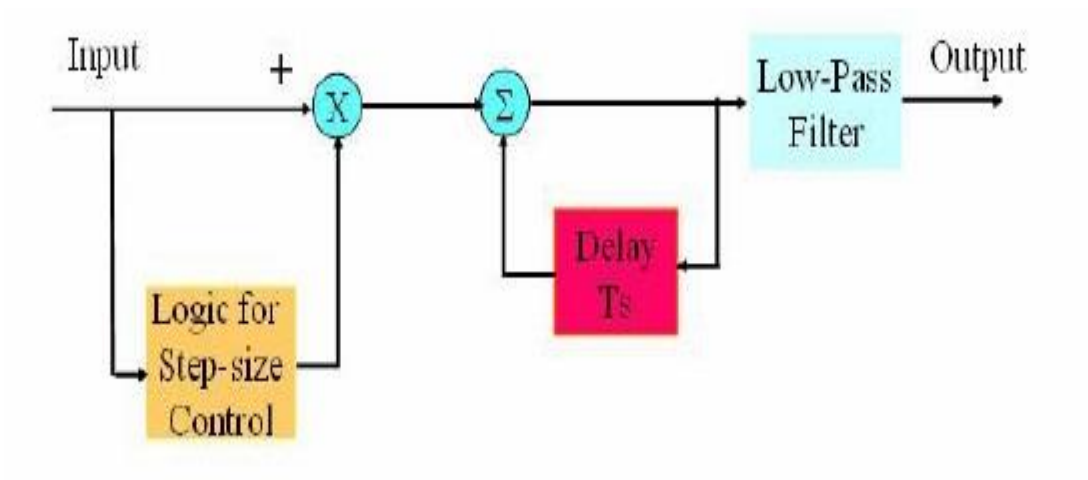
Scheme for delta modulation Receiver:



The Scheme for adaptive delta modulation transmitter:

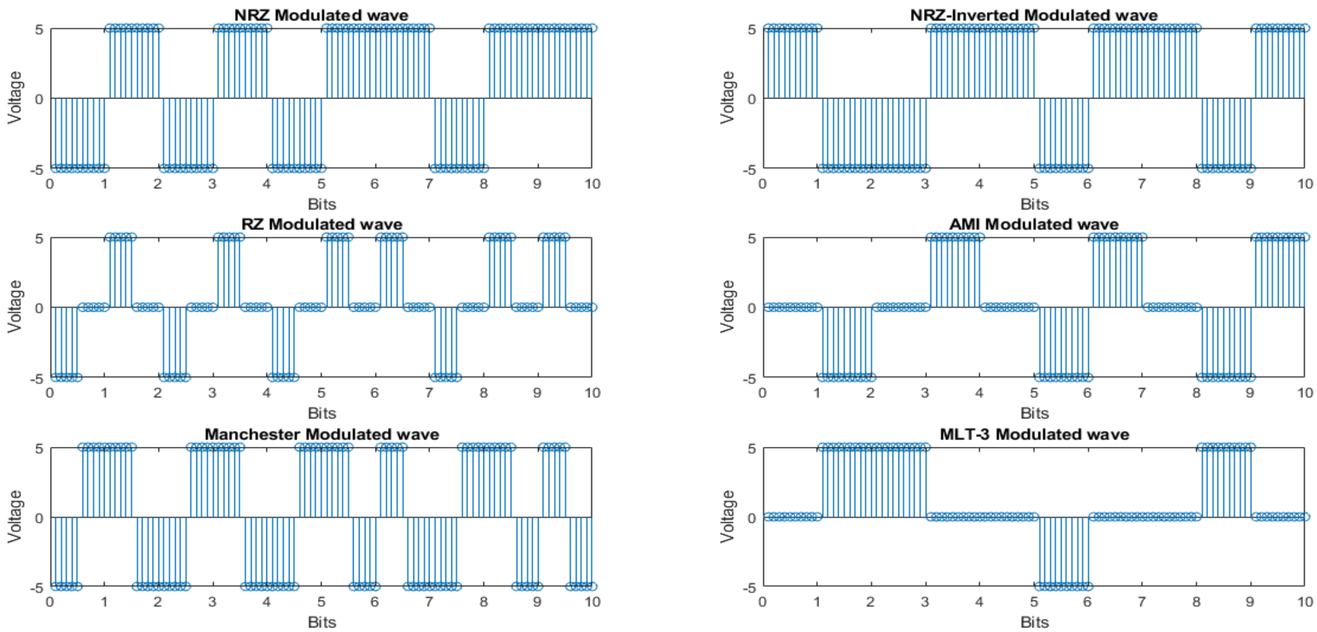


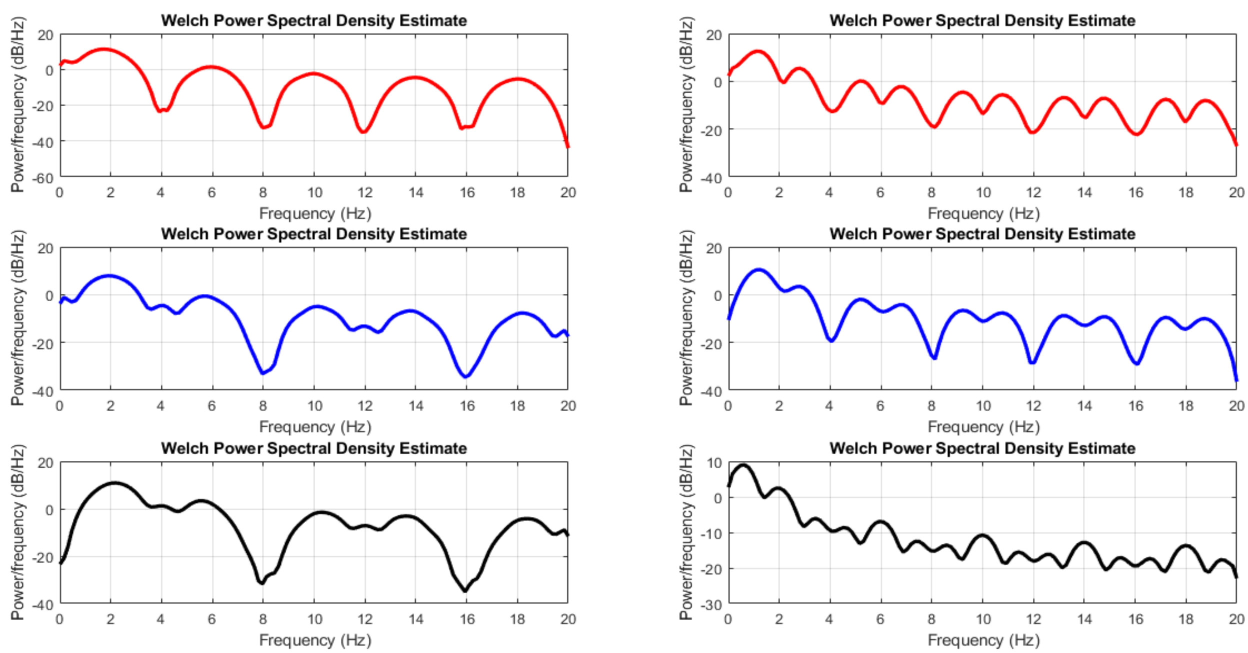
The Scheme for adaptive delta modulation Receiver:



Experiment ٦

2-





3- Manchester

4- **NRZ**:

Advantages

The advantages of Polar NRZ are −

It is simple.

No low-frequency components are present.

Disadvantages

The disadvantages of Polar NRZ are −

No error correction.

No clock is present.

The signal droop is caused at the places where the signal is non-zero at 0 Hz.

Manchester:

Advantages of Manchester Coding: There is always a zero dc level regardless of the data sequence .Good clock recovery - a string of 0s will not cause a loss of clock signal

Disadvantage of Manchester Coding: Null bandwidth is twice that of the polar NRZ (NRZ-L), unipolar NRZ, or bipolar RZ (RZ-AMI) signals.

RZ

Advantages

The advantages of Polar RZ are −

- It is simple.
- No low-frequency components are present.

Disadvantages

The disadvantages of Polar RZ are −

- No error correction.
- No clock is present.

• caused at places where the signal is non-zero at 0 Hz.

NRZI

NRZI marks a one by having a transition from +V to -V (or vice versa) and marks a zero by having NO transition. Now, a long string or ones will recover clock and will no problem to detect, but a long string of zeroes may still be an issue. However, by adding in a run length limited coding scheme like 4B/5B, you can force enough transitions take place for reliable clock recovery. The average signal problem also goes away since detecting the presence or absence of a transition is easier than trying to detect if the voltage is close enough to a threshold of +V or -V to correctly decode the bit.

AMI (Alternate Mark Inversion)

AMI (Alternate Mark Inversion) is a synchronous clock encoding technique which uses bipolar pulses to represent logical 1 values. It is therefore a three level system. A logical 0 is represented by no symbol, and a logical 1 by pulses of alternating polarity. The alternating coding prevents the build-up of a d.c. voltage level down the cable. This is considered an advantage

multi level transmission 3

ADVANTAGES: 1-The PSD acquires a zero value at all 2nTb, thus by using proper filters like Raised cosine filters the side lobes can be suppressed and the multiplexing of the signals can be easily achieved

2- There is a transition of the D.C. amplitude level in the Coded Waveform after every Tb /2 time period irrespective of the data transmitted (1 is transmitted or 0 is transmitted or even if two 1s are transmitted or a 1 and a 0, or vice-versa) and never is zero D.C. amplitude level. Thus irrespective of the number of Zeros and Ones transmitted in any possible combination the VCO is able to reproduce the waveform without being in the idle condition (which arises when a continuous stream of 0s is transmitted using Unipolar RZ and Unipolar NRZ, VCO being given Zero i/p goes in idle condition.)

3- Since there is a transition of the D.C. amplitude level in the Coded Waveform after every Tb /2 time period irrespective of the data transmitted the timing information of a bit can be easily extracted (noting time when D.C. amplitude level remains same and doubling it).

. DISADVANTAGES:

The disadvantage of the Multilevel NRZ is that many transitions take place due to four discrete D.C. levels, used to transmit the data.

5- **MFM Encoding Scheme:**

More data can be stored on the same surface or the data storage density can be increased, if the number of pulses required to store the data can be minimized.

- When minimizing the pulses, one should be careful that the number of no pulses together should not be very long; otherwise the disk controller may go out of synchronization with the data.

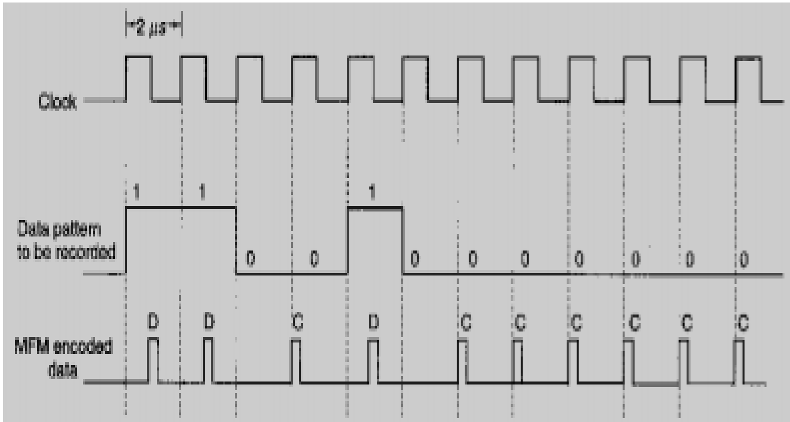
- The MFM (modified frequency modulation) method of data storage, by reducing the number of pulses, is able to store more data without any data and synchronization number of pulses, is able to store more data without any data and synchronization loss. In MFM recording the 0s and 1s are encoded as given below

- 1 is always stored as no pulse, and a pulse(NP)

- 0, when preceded by another 0, is stored as a pulse, and no pulse(PN)

- 0, when preceded by a 1, is stored as two no pulses(NN)

- If you store 1001 on the disk surface using the MFM storage method, it would be stored as NP NN PN NP.



RLL Encoding Scheme

• The RLL is encoding or the run length limited encoding is the most common encoding scheme used in the hard disk storage.

• This encoding scheme can be more accurately called as 2,7 RLL encoding because in this scheme in a series or in a running length the minimum number of 0s next to each other is two, and the maximum number of 0s together can not be more than seven.

• The RLL encoding scheme can store 50 percent more information than MFM encoding scheme on a given surface and it can store three times as much information as the FM encoding scheme. The Run length Limited name comes from the minimum number (run Length) and maximum number (run Limit) of “no pulse” values allowed between two pulses.

For the RLL encoding, an encoder/decoder (Endec) table is used to find the pulse signal to be used for different data bit groups. Endec table used by the IBM to convert bit information to the pulse signal is shown below

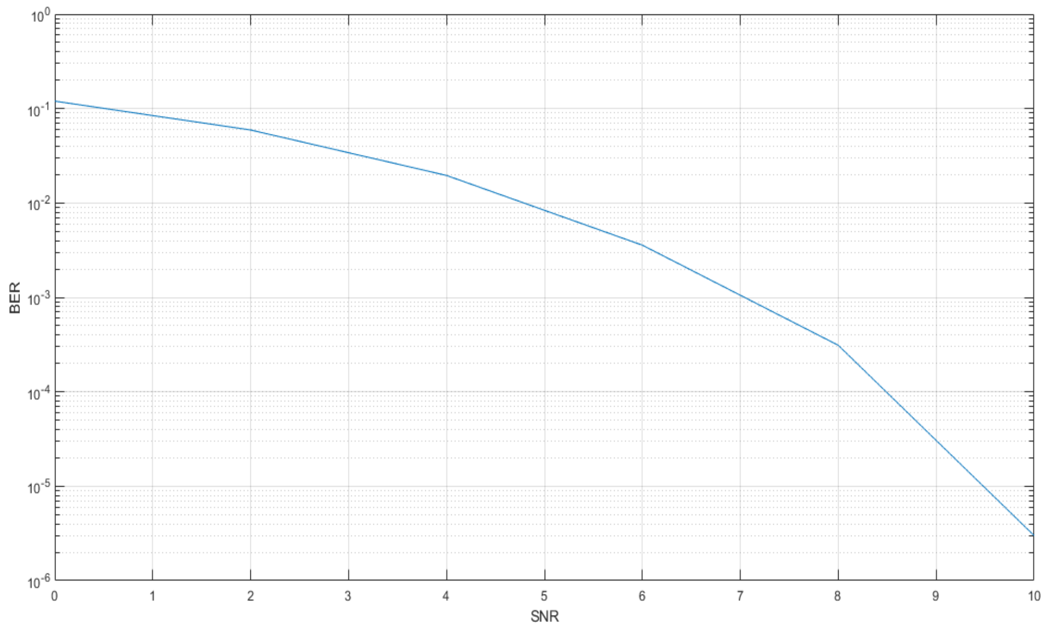
Data Bit	Pulse Encoding
10	NPNN
11	PNNN
000	NNNPNN
010	PNNPNN
011	NNPNNN
0010	NNPNNPNN
0011	NNNNPNNN

For example, if you want to encode a byte 100011 to proper RLL pulse signal then the

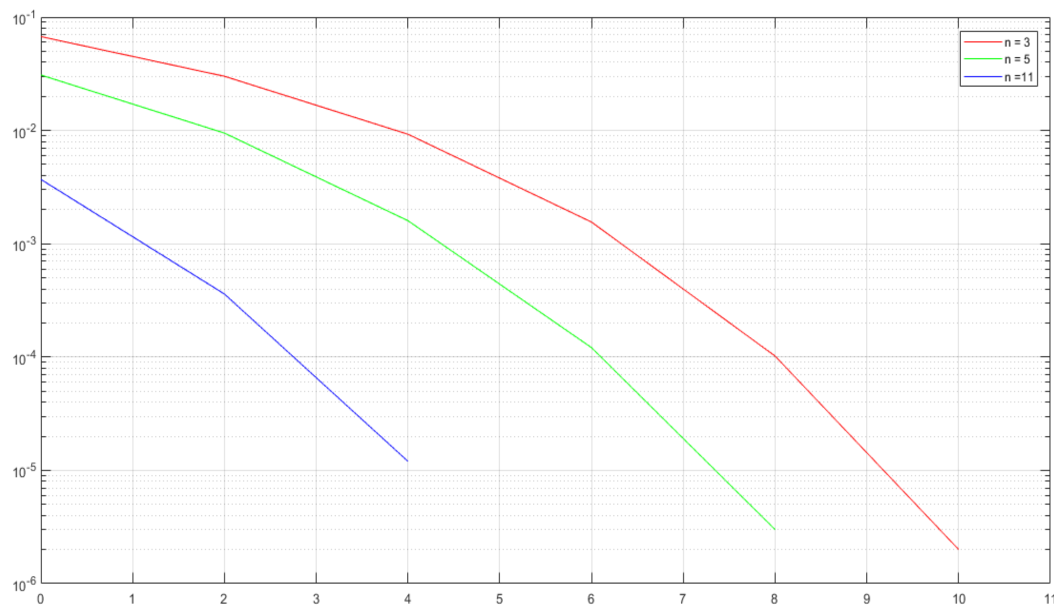
• Bit 10 can be encoded as NPNN • Bit 0011can be encoded as NNNNPNNN

Experiment 7

1- BER for systematic linear block code

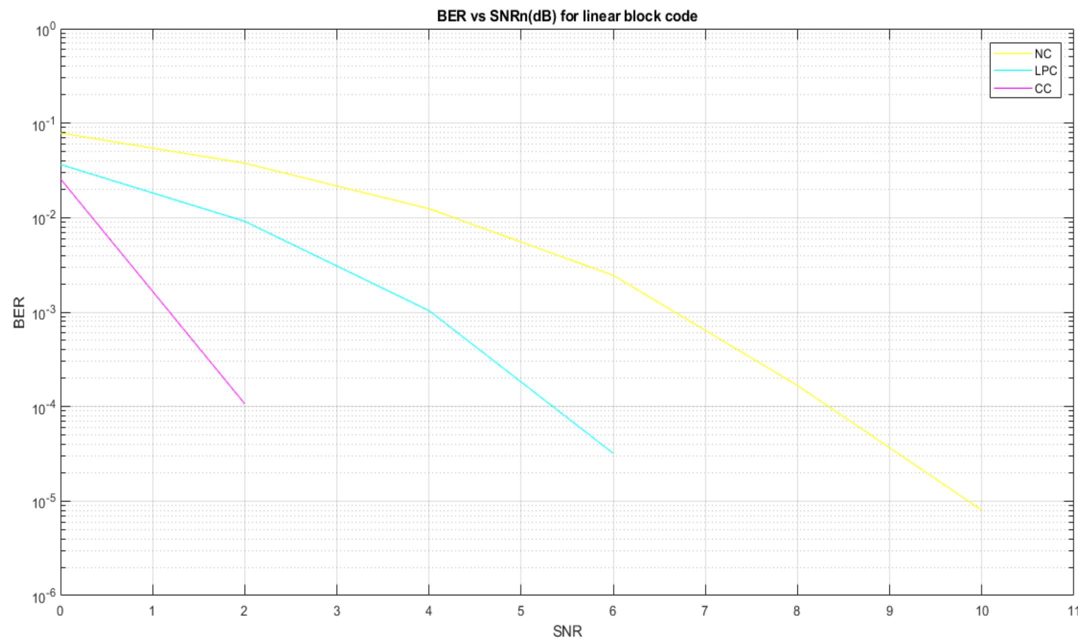


2- Probability of error for n = 3, 5, 11 repetition code



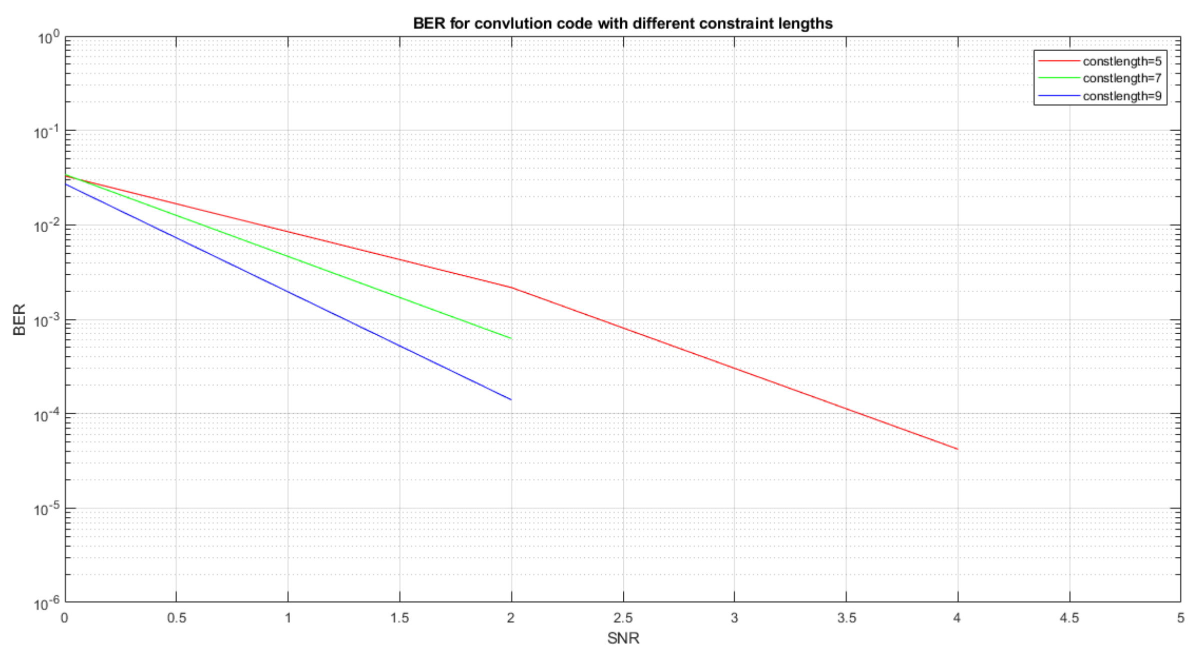
The higher the repetition, the lower the probability of error is present but bit time is highly increased.

3- Comparison between uuencoded, linear block code, convolution code



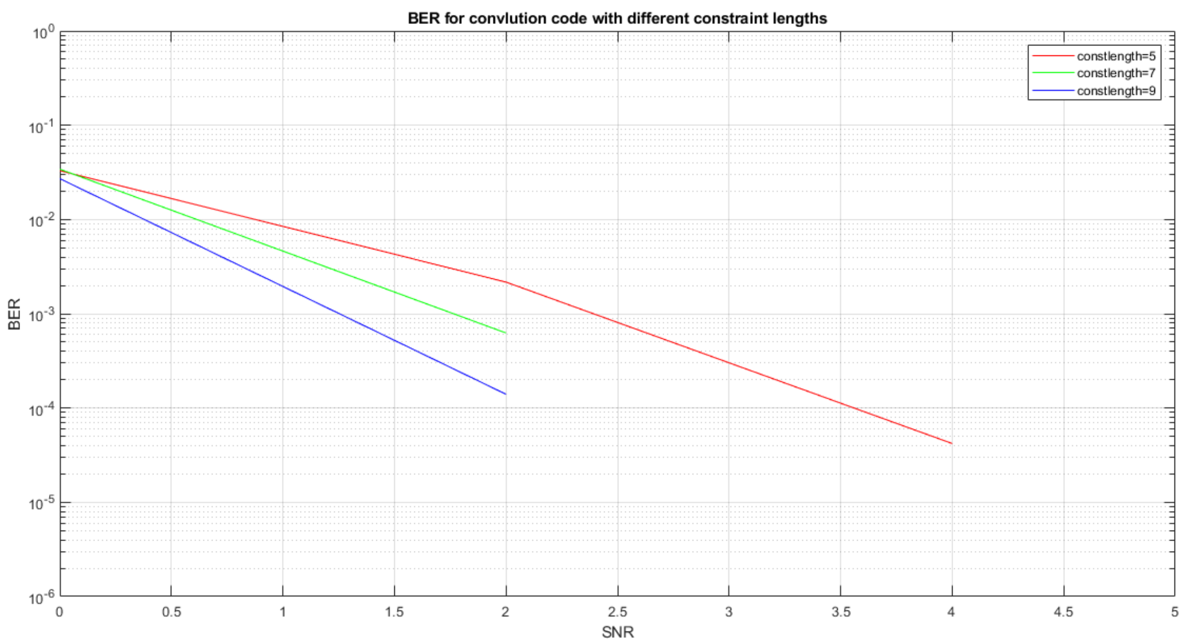
No coding is less immune against noise then linear block code then convolution code, however any coding system adds redundancy to the actual noise with lowers the bit rate somehow.

4- Changing the constraint length in convolution code case and see the difference



Clearly, the higher the constraint length, the lower BER we get, however increasing the constrain length, decreases the system time response and adds delay to it.

5- Changing the constraint length in convolution code case and see the difference



Clearly, the higher the constraint length, the lower BER we get, however increasing the constrain length, decreases the system time response and adds delay to it.