Department of Instrumentation and Control Engineering,
Faculty of Technology,
Dharmsinh Desai University

Course: Bachelor of Technology
Branch: Instrumentation and Control Engineering
Semester: V
Subject: Communication Systems (Laboratory Manual)

Prepared by:
Tigmanshu Patel,
Asst. Prof, IC dept

Index

LAB 1

To study Amplitude Modulation and Demodulation

Aim: To study amplitude modulation and demodulation, and to simulate the various techniques for same in MATLAB/SCILAB.

Theory:

Modulation is the process of influencing data and information on the carrier. It is to regulate some parameter of high-frequency carrier wave with a low-frequency information/ message signal.

Need of modulation?

- Size of Antenna:
    - It was found that antenna dimensions had to be of the same order of magnitude as the wavelength of the signal transmitted.
    - Since frequency and wavelength(λ) are related by the following relation:

$$\mathbf{f\lambda = c}$$

    - Hence, from the relation we can see that frequency is inversely proportional to the wavelength(λ), and since λ is directly proportional to the size of the antenna, for low frequency signals we can say that the λ will be high and hence the size of the antenna too.
- Interference from other signals for low frequency signals.

Hence, for practical implementation of modulation, the carrier frequency has to be much higher than the information/message signal.

Types of Modulation:

1. Amplitude Modulation
2. Angle Modulation
    a. Frequency Modulation

b. Phase Modulation

In this experiment, we will be focusing on amplitude modulation technique. The topic angle modulation study is covered in next experiment.

Amplitude Modulation:

Amplitude of carrier signal is changed in proportion to the information/ message signal, while keeping the frequency and the phase components of the signal constant. The voltage/ amplitude is directly proportional to the modulating signal which is added to the carrier amplitude. Mathematically it is represented by,

$$e_c(t) = [E_{cmax} + e_m(t)]\cos(2\pi f_c t + \phi_c) \text{ ………………………. eq. 1.1.}$$

where, $e_c(t)$ is the carrier signal, $e_m(t)$ modulating signal.

The term $E_{cmax} + e_m(t)$ describes the envelope of the modulated wave.

Concept of Modulation Index:

It is the measure of extent of modulation done on a carrier signal. In AM, it is defined as the ratio of the amplitude of the modulating signal ($e_m(t)$) to the amplitude of the carrier signal ($e_c(t)$).

$$m = \frac{A_m}{A_c} = \frac{E_{max} - E_{min}}{E_{max} + E_{min}} \text{ ………………………. eq. 1.2.}$$

The physical significance of this index is that how much the signal will increase by the factor (m) and decrease to (1-m) of it original level. According to the different values of m, it is classified into three categories:

1. Under-Modulation ($0 < m < 1$)
2. Critical-Modulation ($m=1$)
3. Over-Modulation ($m>1$)

Ideally critical modulation is favoured, but practically implemented is the under-modulation type since, if over-modulation is encountered, then the problems of distortion of signal and difficulty in demodulating the modulated signal is encountered. A phenomenon called as sideband splatter is also encountered due on over-modulation.

Sinusoidal AM:

Let the modulating/ information signal be,

$$e_m(t) = E_{mmax}\cos(2\pi f_m t + \phi_m) \text{ ……………………… eq. 1.3.}$$

The carrier signal is,

$$e_c(t) = E_{cmax}\cos(2\pi f_c t + \phi_c) \text{ ……………………… eq. 1.4.}$$

Hence, the modulated signal is given as from equation 1.1.,

$$e(t) = (E_{cmax} + E_{mmax}\cos(2\pi f_m t))\cos(2\pi f_m t) \text{ ……………………… eq. 1.5.}$$

From equation 1.2.,

$$\boxed{e(t) = E_{cmax}(1 + m\cos(2\pi f_m t))\cos(2\pi f_m t)} \text{ ……………………… eq. 1.6.}$$

where, m = modulation index

DIY: *The students are required to explore DSBSB, SSBSC. Compare the equations for the same. Also, try to get the power associated with each of the case, which one is the best AM?*

Modulator Circuits for AM:

1. Square Law Modulator
2. Switching Modulator – Diodes, Transistors and Vacuum tubes.

Demodulation of AM:

1. Square law demodulation (m <= 1)
2. Envelope Detector (m <= 1)

3. Synchronous Detector (for any m)

Simulation:

The following is the reference code for the simulation of amplitude modulated signal. The student is required to go through the code, understand the same by utilizing the above theory part and relating the code. Run the code in SCILAB. Observe the various waveforms obtained.

Code:

```
clf ()
clc
Ec =10 , ma =1.5 , wm =2* %pi *100 , wc =2* %pi *10000 , fs =100000 , f =100;
x =0:1/ fs :((2/ f) -(1/ fs));

deff("[m]= f(x)" ,"m=sin(wm*x)");
subplot(3 ,1 ,1);
fplot2d(x,f);
xlabel("Time" , "fontsize" , 3);
ylabel("Amplitude" , "fontsize" , 3, "color" , "red");
title("Message Signal");
deff("[c]= f(x)" ,"c=Ec*sin(wc*x)");
subplot(3 ,1 ,2);
fplot2d(x,f);
xlabel("Time" , "fontsize" , 3);
ylabel("Amplitude" , "fontsize" , 3, "color" , "red");
title("Carrier Signal");
deff("[y]= f(x)" ,"y=Ec*(1+ma*(sin(wm*x)))*sin(wc*x)");
subplot(3 ,1 ,3);
fplot2d(x,f);
xlabel("Time" , "fontsize" , 3);
ylabel("Amplitude" , "fontsize" , 3, "color" , "red");
title("Amplitude Modulated Signal");
```

Test different values for modulation index in above code, and find out the differences being observed.

Assignment:

Instructions:

- Assume suitable data wherever necessary and clearly mention the same.
- The final submission from student side shall have a pdf that has answers/results in it. It should include codes and Simulink block diagrams if any.
- Each problem should have a final answer highlighted in case of calculative problems or conclusion clearly written in case of program.

Q.1. Design a MATLAB/SCILAB script-based Amplitude modulator. Plot modulating, carrier and modulated signals on the same plot using subplots. Generate outputs with different modulation index and verify for yourself the effect of modulation in the output. Use various modulation index of 0.25, 0.5, 0.75, 1, 1.25, 1.50.

Q.2. Design a SIMULINK/XCOS based Amplitude modulator. Plot modulating, carrier and modulated signals on the same plot using subplots.

Q.3. A sinusoidal carrier is amplitude modulated by a square wave that has zero dc component and a peak-to-peak value of 2 V. The periodic time of the square wave is 0.5ms. The carrier amplitude is 2.5 V, and its frequency is 10 kHz. Plot the modulating signal, carrier and the modulated wave in MATLAB/SCILAB.

Q.4. For an amplitude modulated signal design a demodulator circuit using MATLAB/SCILAB script such that it detects the envelope. Plot modulated signal, information signal, and demodulated signal on single graph.

Q.5. For an amplitude modulated signal design a diode demodulator circuit using SIMULINK/XCOS such that it detects the envelope.

LAB 2

To study Angle Modulation and Demodulation

Aim: To study frequency modulation – demodulation and phase modulation – demodulation technique, and to simulate the various techniques for same in MATLAB/SCILAB.

Theory:

A carrier modulation can be achieved by modulating the amplitude, frequency, and phase of a **sinusoidal carrier** of frequency f_c. When the frequency of the carrier sinusoidal signal is varied in proportion to the message signal, it is called as the **frequency modulation**. Whereas when the phase of the carrier sinusoidal signal is varied in proportion to the message signal, it is called as the **phase modulation**.

Frequency Modulation:

The idea of frequency modulation (FM), where the carrier frequency would be varied in proportion to the message m(t), was quite intriguing. The frequency of the carrier signal changes, whereas the amplitude and the phase of the signal are kept constant. In a well-designed modulator the change in carrier frequency will be proportional to the modulating voltage and thus can be represented as $ke_m(t)$, where k is a constant known as the frequency deviation constant. The instantaneous frequency is given as:

$$\mathbf{f_i(t) = f_c + ke_m(t)} \text{ .... eq. 2.1.}$$

where $f_i(t)$ is the instantaneous frequency, f_c is the carrier frequency, k is the frequency deviation constant (Hz/V) and the $e_m(t)$ is the modulating signal.

To the above-mentioned instantaneous frequency, the corresponding angular velocity will be,

$$\omega_i(t) = 2\pi f_i(t) \text{ .... eq. 2.2}$$

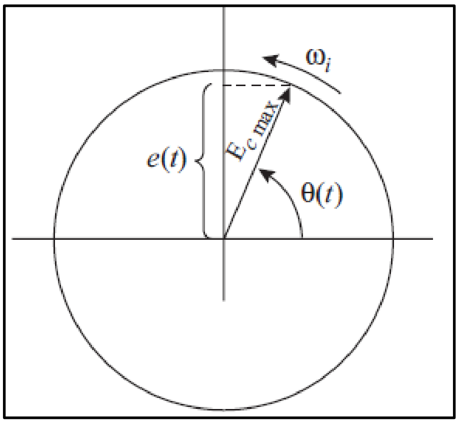

Fig. 2.1. – Phasor representation of carrier signal with amplitude E_{cmax} rotating at instantaneous angular velocity $\omega_i(t)$

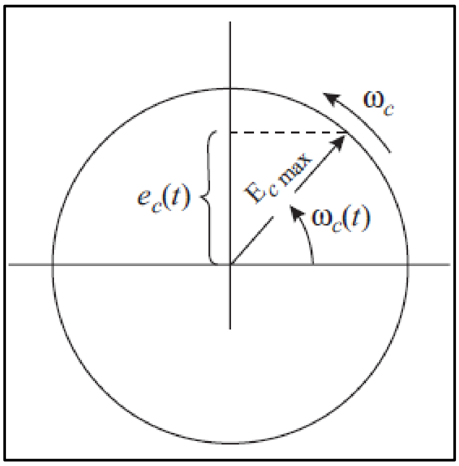

Fig. 2.2. – Phasor representation of carrier signal with amplitude E_{cmax} rotating at constant angular velocity $\omega_C(t)$

The angle turned through in time t is shown as θ(t), The angle θ(t) is found by noting that the angular velocity is the time rate of change of angle,

$$\theta(t) = 2\pi f_c t + 2\pi k\int_0^t e_m(t)dt \text{........................... eq. 2.3}$$

Sinusoidal FM:

Let modulating signal $e_m(t)$ be,

$$e_m(t) = E_{mmax} \cos 2\pi f_m t$$

Hence from equation 2.1., the instantaneous frequency will be,

$$f_i(t) = f_c + kE_{mmax} \cos 2\pi f_m t \text{ ……………………… eq. 2.4}$$

$$f_i(t) = f_c + \Delta f \cos 2\pi f_m t \text{……………………… eq. 2.5}$$

where $\Delta f = kE_{mmax}$ which is the frequency deviation directly proportional to the peak modulating signal.

The expression for the modulated signal will be,

$$e(t) = E_{cmax} \cos\theta(t) \text{ ……………………… eq. 2.6}$$

From equation 2.3, the above equation will be,

$$e(t) = E_{cmax} \cos[2\pi f_c t + \Delta f 2\pi \int_0^t \cos 2\pi f_m t \, dt]$$

$$\boxed{\mathbf{e(t) = E_{cmax} \cos[2\pi f_c t + \frac{\Delta f}{f_m} \sin 2\pi f_m t]}}$$

where $\beta = \frac{\Delta f}{f_m}$ modulation index for frequency modulation (FM).

Hence,

$$e(t) = E_{cmax} \cos[2\pi f_c t + \beta \sin 2\pi f_m t] \text{ ……………………… eq. 2.7}$$

<u>Phase Modulation:</u>

PM type of modulation is intended for transmitting signals. It changes the message signal in accordance with the carrier signal due to the differences in immediate phases. This modulation is combination of two principal forms such as frequency and angle modulation.

The unmodulated carrier is given as,

$$e_c(t) = Ecmax\cos(\omega_c t + \phi_c)$$

where ϕ_c is a=arbitrary phase angle.

Mathematically, the phase modulation can be represented as,

$$\Phi(t) = \phi_c + Ke_m(t) \text{ ……………………… eq. 2.8}$$

where ϕ_c is the unmodulated signal, K is the phase deviation constant(rads/V) (which is equivalent to frequency deviation constant k in FM), and $e_m(t)$ is the message signal.

Hence, the phase modulated wave can be represent in general way as:

$$e(t) = E_{cmax}\cos(\omega_c t + Ke_m(t)) \quad \text{………………………} \quad \text{eq. 2.9}$$

Note: in the above equation 2.9 the term ϕ_c is neglected since it is constant term.

From above, the important points that can be drawn are:

- If the phase deviation (K) is more, implies the input signal amplitude increases, hence +ve slope, and therefore the carrier will undergo phase lead.
- If the phase deviation (K) is less, implies the input signal amplitude decreases, hence -ve slope, and therefore the carrier will undergo phase lag.

Sinusoidal PM:

Let modulating signal $e_m(t)$ be,

$$e_m(t) = E_{mmax} \cos 2\pi f_m t$$

Hence, the modulated signal will be given as from equation 2.9. as,

$$e_m(t) = E_{mmax} \cos[2\pi f_c t + KE_{mmax}\cos 2\pi f_m t]$$

$$\boxed{\mathbf{e_m(t) = E_{mmax} \cos[2\pi f_c t + \Delta\phi \cos 2\pi f_m t]}} \quad \text{………………………} \quad \text{eq. 2.10}$$

where $\Delta\phi$ is the peak phase deviation.

Relation between FM and PM:

From equation 2.7 and 2.10, one can observe and draw the relation between FM and PM. It is apparent that PM and FM not only are very similar but are inseparable. Replacing m(t) in equation 2.9 with $\int_0^t e_m(t)dt$ changes PM into FM. Thus, a signal that is an FM wave corresponding to m(t) is also the PM wave corresponding to $\int_0^t e_m(t)dt$. Similarly, a PM wave corresponding to m(t) is the FM wave corresponding to m'(t). Therefore, by looking only at an angle-modulated signal, there is no way of telling whether it is FM or PM. In fact, it is meaningless to ask an angle-modulated wave whether it is FM or PM.

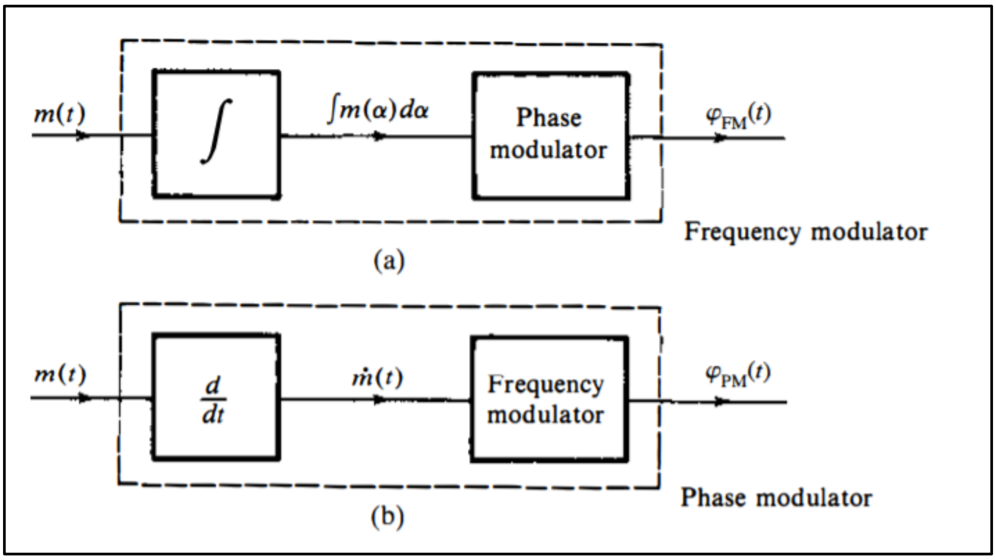

Fig. 2.3. – Phase and Frequency modulation are equivalent and interchangeable

---

DIY: *Explore NBFM and WBFM, which one is used and why?*

<u>Angle Modulation Detectors/ Demodulation:</u>

It is divided into 2 categories:

1. Frequency discrimination method
   a. Slope Detector
   b. Balanced Slope Detector / Round-Travis detector
2. Phase discrimination method
   a. Foster Seley Detector
   b. Ratio Detector

c. PLL method
d. Quadrature detector

Simulation:

The following is the reference code for the simulation of frequency modulated signal. The student is required to go through the code, understand the same by utilizing the above theory part and relating the code. Run the code in SCILAB, test various values of frequency deviation constant and amplitude of the message signal. Observe the various waveforms obtained.

Code:

```
clear ;
clc ;
close ;
Ec =10 , wm =2* %pi *100 , wc =2* %pi *10000 , f =100;

fs =100000;

kf= input ( ' Enter the Frequency deviation constant (k): ')
Em= input ( ' Enter the amplitude of the message signal (Em): ' )
de1 = kf*Em ;

x =0:1/ fs :((2/ f) -(1/ fs)); // time axis setting

mf=de1/f; // modulation index of FM
disp (mf , ' Modulation Index of FM' )

//Message singal
previousprot = funcprot (1)
deff (" [m]= f ( x ) " ,"m=cos(wm*x )")
subplot (3 ,1 ,1)
fplot2d(x,f)
xlabel(" t ime " , " fontsize" , 3);
ylabel("Amplitude " , " fontsize" , 3, "color" , "red");
title("Message signal")
```

```
//Carrier Signal
funcprot (0)
deff (" [ c ]= f ( x ) " ," c=Ec*cos(wc*x ) ")
subplot (3 ,1 ,2)
fplot2d (x,f)
xlabel (" t ime " , " fontsize" , 3);
ylabel ("Ampl i tude " , " fontsize" , 3, "color" , "red");
title ("Carrier Signal")

// Modulated Signal
funcprot (0)
deff (" [ y]= f ( x ) " ,"y=Ec*cos( ( wc*x )+mf*sin(wm*x))")
subplot (3 ,1 ,3)
fplot2d (x,f)
xlabel (" time " , "fontsize" , 3);
ylabel ("Amplitude " , " fontsize" , 3, "color" , "red");
title (" Frequency Modulated Signal ")
```

Test Cases:

Test Case	K_f (Hz/V)	E_m(V)
#1	5	10
#2	100	10
#3	1000	10
#4	1000	5
#5	1000	100

Write down your comments, observations, and conclusion from the above test cases.

Assignment:

Instructions:

- Assume suitable data wherever necessary and clearly mention the same.
- The final submission from student side shall have a pdf that has answers/results in it. It should include codes and Simulink block diagrams if any.
- Each problem should have a final answer highlighted in case of calculative problems or conclusion clearly written in case of program.

Q.1. Design a MATLAB/SCILAB script-based Frequency modulator to demonstrate frequency modulation conceptually.

Q.2. In the above example, generate outputs with different modulation index and verify for yourself the effect of modulation in the output. Use various modulation index of 0.25, 0.5, 0.75, 1, 1.25, 1.50.

Note: *Use the above program. Further generate multiple output signals by writing down multiple equations where each equation will have a different modulation index. Then plot all signals in subplots. The subplots should be in form of 3x3 matrix which can be initiated by following commands: figure(), subplot(3,3,1), subplot(3,3,2) and so on. Use legends to specify the modulation index on each plot. For example, It can be specified by: legend(„\mu= 0.5‟). Some subplots might remain empty*

Q.3. For the above program now modify the information signal to preferably a recorded file from your microphone (Spell "DDU" and record from microphone) and demonstrate Frequency modulation for the entire recorded signal by displaying modulating signal, carrier signal and modulated signal on the same graph.

Q.4. Design a MATLAB/SCILAB script-based Frequency demodulator to demonstrate frequency demodulation conceptually. Use a sinusoidal audio signal and plot the demodulated signal on top of original information signal.

Q.5. Demonstrate PLL method in SIMULINK/XCOS. The answer must contain the block diagram, the output, and the necessary assumptions if any made, you must mention the same.

LAB 3

To study Sampling Theorem and Reconstruction

Aim: To demonstrate the time domain sampling of bandlimited signals (Nyquist theorem), of non-bandlimited signals and antialiasing filter, and signal reconstruction using filters.

Theory:

Sampling:

Quantizing the amplitude of continuous signal to digital data of discrete times is referred to as sampling. Quantization is the process of constraining an input from a continuous or otherwise large set of values (such as real numbers) to discrete set (such as integers). A continuous time signal (analog signal) can be processed by analysing its samples through a discrete time system. Now to reconstruct the signal to the continuous time domain signal from its discrete time samples without any error, the signal must be sampled at a sufficient rate. This sufficient rate is determined with the help of the sampling theorem.

Need of Sampling:

The real-world signals are in the form of analog in nature i.e., continuous time domain signals. Using digital communication to transmit the message from one place to another is efficient when compared to analog communication. Hence, the continuous time domain signals are required to be converted into digital signals so that the computers and other digital devices can process the data, the information signal and can transmit the same. Hence, the need of converting analog signals to digital signal is accomplished by ADCs, and in this process, there is a need of sampling phenomenon.

Nyquist Sampling Theorem:

If a signal is band limited and its samples are taken at sufficient rate than those samples uniquely specify the signal and the signal can be reconstructed from those samples. The condition in which this is possible is known as Nyquist sampling theorem and is stated below.

'*The sampling theorem indicates that a continuous signal can be properly sampled, only if it does not contain frequency components above one-half of the sampling rate.*'

For instance, a sampling rate of 2,000 samples/second requires the analog signal to be composed of frequencies below 1000 cycles/second. If frequencies above this limit are present in the signal, they will be aliased to frequencies between 0 and 1000 cycles/second, combining with whatever information that was legitimately there.

The theorem can also be reframed as:

'*To be able to completely reconstruct a 1D signal from a set of signals, the sampling frequency f_s/ the sampling rate must be at least twice the highest frequency (f_h/f_m) present in the signal.*'

Mathematically,

$$\mathbf{f_s >= 2f_m}$$

Conditions for recovering continuous time signal from the discrete samples are:

1. Signal must be band-limited.
2. The sampling frequency (f_s) must be greater than the twice of the highest frequency(f_m) of the message signal.

From these, three main observations/ cases are identified:

Case#1: $f_s = 2f_m \rightarrow$ Critical sampling

- This is the ideal situation and most favourable for all the sampling circuits, but practically hard to achieve.

Case#2: $f_s < 2f_m$ → Under sampling

- Sampling is done at frequency being less than the twice of high frequency, which results in under-sampling. This phenomenon leads to the effect of **aliasing**.
- This aliasing effect causes problem in the reconstruction of the message signal.
- This leads to loss of information, and phenomena such as **ISI** and **cross-talks** (Sometimes when you call your parents or friends, you have dialled the correct contact number, but on the other side you are hearing someone else's voice, and you might have referred this as it happened a cross. This is nothing but cross-talk).

Case#3: $f_s > 2f_m$ → Over-sampling

- This is what we want, but sometimes the sampling frequency being much greater than twice of the message signal highest frequency will increase the bandwidth and hence the cost increases. This is the only disadvantage.

Aliasing:

In reconstructing a signal from its samples, there is another practical difficulty. The sampling theorem was proved on the assumption that the signal x(t) is bandlimited. All practical signals are time limited, i.e., they are of finite duration. As a signal cannot be time-limited and bandlimited simultaneously. Thus, if a signal is time-limited, it cannot be bandlimited and vice versa (but it can be simultaneously non time-limited and non-bandlimited).

Clearly it can be said that all practical signals which are necessaily time limited, are non-bandlimited, they have infinite bandwidth and the spectrum consists of overlapping cycles of repeating every f_s Hz (sampling frequency). Because of infinite bandwidth, the spectral overlap will always be present regardless of what ever may be the sampling rate chosen. Because of the overlapping tails, has not complete information about the signal and it is not possible, even theoretically to recover it from the sampled signal. If the sampled signal is passed through an ideal lowpass filter, the output is distorted as a result of some causes.

Anti-Aliasing Filter:

It is for the non-bandlimited signals. Anti-aliasing filter is a filter which is used before a signal sampler, to restrict the bandwidth of a signal to approximately satisfy the sampling theorem. The potential defectors are all the frequency components beyond fs/2 Hz. We should have to eliminate these components from original signal before sampling it. As a result of this we lose only the components beyond the folding frequency $f_s/2$ Hz. These frequency components cannot reappear to corrupt the components with frequencies below the folding frequency.

This suppression of higher frequencies can be accomplished by an ideal filter of bandwidth $f_s/2$ Hz. This filter is called the **anti-aliasing filter**. The anti-aliasing operation must be performed before the signal is sampled. The anti-aliasing filter, being an ideal filter is unrealizable. In practice, we use a steep cut-off filter/ LPF, which leaves a sharply attenuated residual spectrum beyond the folding frequency $f_s/2$.

Verifying the sampling theorem:

Code:

```
clc
clear
xdel(winsid())
t = -100:1:100;
fm = 0.02;
x = cos(((2*%pi)*t)*fm);
subplot(2,2,1);
plot(t,x);
xlabel("time in sec");
```

```
ylabel("x(t)");
title("continuous time signal");
fs1 = 0.02;
n = -2:2;
x1 = cos((((2*%pi)*fm)*n)/fs1);
subplot(2,2,2);
plot2d3('gnn',n,x1);
set(gca(),"auto_clear","off")
subplot(2,2,2);
plot(n,x1,":");
title("discrete time signal x(n) with fs<2fm");
xlabel("n");
ylabel("x(n)");
fs2 = 0.04;
n1 = -4:4;
x2 = cos((((2*%pi)*fm)*n1)/fs2);
subplot(2,2,3);
plot2d3('gnn',n1,x2);
set(gca(),"auto_clear","off")
subplot(2,2,3);
plot(n1,x2,":");
title("discrete time signal x(n) with fs>2fm");
xlabel("n");
ylabel("x(n)");
n2 = -50:50;
fs3 = 0.5;
x3 = cos((((2*%pi)*fm)*n2)/fs3);
subplot(2,2,4);
plot2d3('gnn',n2,x3);
set(gca(),"auto_clear","off")
subplot(2,2,4);
plot(n2,x3,":");
xlabel("n");
ylabel("x(n)");
title("discrete time signal x(n) with fs=2fm");
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Assignment:

Consider an analog signal x(t) consisting of three sinusoids of frequencies of 1 kHz, 4 kHz, and 6 kHz:

$$x(t)= \cos(2\pi t)+ \cos(8\pi t)+ \cos(12\pi t)$$

where t is in milliseconds. Show that if this signal is sampled at a rate of f_s = 5 kHz, it will be aliased with the following signal, in the sense that their sample values will be the same:

$$x_a(t)= 3\cos(2\pi t)$$

On the same figure, plot the two signals x(t) and $x_a(t)$ versus t in the range $0 \le t \le 2$ msec. To this plot, add the time samples x(tn) and verify that x(t) and $x_a(t)$ intersect precisely at these samples. These samples can be evaluated and plotted as follows:

```
fs = 5;
T = 1/fs;
tn = 0:T:2;
xn = x(tn);
plot(tn, xn, '.');
```

You can refer the above sample code given for simulation.

LAB 4

Pulse Amplitude Modulation (PAM)

Aim: To study the waveform of Pulse Amplitude Modulation (PAM) technique.

Theory:

Introduction:

Pulse amplitude modulation (PAM) is used extensively in telecommunications as an intermediate step of other techniques such as phase shift keying (PSK), quadrature amplitude modulation (QAM) and pulse code modulation (PCM). PAM however is an amplitude modulated (AM) form of a pulse carrier, and hence has all the advantages and disadvantages of the purely analog AM, a major disadvantage being noise. PAM can be time-division multiplexed (TDM), as can pulse code modulation (PCM) which is a digital signal.

TDM is one of the multiplexing techniques used in telephony (the other is space division multiplexing). PAM is used as a first step in converting voice signal to PCM in the public switched telephone network (PSTN), and is also used to produce high-level modulation schemes for data modems and digital radio. High-level modulation is done in the output circuit of the radio frequency (RF) power amplifier stage, and is more efficient than low-level modulation. PCM is used for long-distance telecommunications, making PAM an important pulse modulation technique in communications systems.

PAM:

If amplitude of the carrier signal is varied according to the amplitude of the message signal is called as the pulse amplitude modulated (PAM) signal. As mentioned above that the effect of noise is high and it is one of the disadvantages of this technique, this is because the PAM message signal is transmitted through wired channel and amplitude is being continuous in nature, hence the effect of noise is high.

The PAM technique can be multiplexed by TDM and is extensively used for communication. The technology that employs TDM is less expensive than that which employs FDM. PAM is used as an intermediate form of modulation with pulse modulation techniques such as pulse shift keying (PSK), quadrature amplitude modulation (QAM), and pulse code modulation (PCM). It is an analog technique in which the amplitude of each pulse is proportional to the amplitude of the signal when it was sampled.

Plotting the waveform of PAM signal in SCILAB:

Code:

```
clc;
clear all;

t = 0:1:100;
fm = input( 'Enter the message signal frequency = ' );
x = cos(2* %pi *fm*t);
subplot(2 ,1 ,1);
plot(t,x);
xlabel("Time" , "fontsize" , 3);
ylabel("Amplitude" , "fontsize" , 3, "color" , "red");
title('Message Signal');

fs3 = input('Enter the sampling  frequency = ' );
x3 =cos(2* %pi *fm*t/ fs3 );
subplot(2 ,1 ,2);
plot2d3(t,x3)
xlabel("Time" , "fontsize" , 3);
ylabel("Amplitude" , "fontsize" , 3, "color" , "red");
title('PAM Signal');
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. $f_m = 2$, $f_s = 2$
2. $f_m = 2$, $f_s = 1$
3. $f_m = 2$, $f_s = 0.5$
4. $f_m = 5$, $f_s = 10$
5. $f_m = 5$, $f_s = 7$

Paste the output plots, and give your comments on same. Put down the conclusion for same.

LAB 5

Pulse Width Modulation (PWM)

Aim: To study the waveform of Pulse Width Modulation (PWM) technique.

Theory:

Pulse width modulation (PWM) also called as pulse duration modulation (PDM). The width of the carrier signal is varied according to the message signal amplitude. The amplitude of the signal is kept constant. The width of the pulse and the position of the pulse is made proportional to amplitude of the signal.

This is achieved in three ways:

1. By keeping the leading-edge constant and varying the pulse width with respect to the leading-edge.
2. By keeping the tailing-edge constant.
3. By keeping the centre of the pulse constant.

The baseband signal information appears in the signal and is not distorted by any modulation effects. It is recovered by using simple LPF which will remove the carrier and its harmonics, and a HPF which will remove the DC component if any present.

Applications of PWM:

1. Telecommunication systems
2. Power delivering systems
3. Embedded applications
4. Robotics
5. Audio effects and amplifications purposes

Advantages of PWM:

1. Noise interference is less due to amplitude being kept constant.
2. Signals can be separated very easily at demodulation time, and if any noise present it can also be separated easily.
3. No need of synchronization between transmitter and receiver.

Disadvantages of PWM:

1. Due to varying width of the pulse, power is variable.
2. Bandwidth is large when compared to PAM.

Plotting the waveform of PWM signal in SCILAB:

Code:

```
clc;
clear all;

t = 0:0.001:1;
fc = input("Enter Carrier Signal Frequency = ");
c = asin(sin(2*%pi*fc*t));
fm = input ("Enter Message Signal Frequency = ");
m = (2/%pi)*sin(2*%pi*fm*t);
n = length(c);
for i = 1:n
   if (m(i) >= c(i))
      pwm(i) = 1;
   else (m(i) <= c(i))
      pwm(i) = 0;
   end
end

figure;
subplot(3 ,1 ,1);
```

```
plot(t, m);
xlabel("Time");
ylabel("Amplitiude");
title("Message Signal");
subplot(3 ,1 ,2);
plot(t,c);
xlabel("Time");
ylabel("Amplitude");
title ("Carrier Signal");
subplot(3, 1, 3);
plot(t,pwm');
xlabel("Time");
ylabel("Ampl i tude ");
replot([0 -1 1 2]);
xlabel("Time");
ylabel("Amplitude");
title ("Pulse width modulation");
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. $f_c = 10$, $f_m = 2$
2. $f_c = 50$, $f_m = 2$

Paste the output plots, and give your comments on same. Put down the conclusion for same.

LAB 6

Pulse Position Modulation (PPM)

Aim: To study the waveform of Pulse Position Modulation (PPM) technique.

Theory:

The position of the carrier signal is varied according to the message signal amplitude. The amplitude and the width of the signal is kept constant. It uses the pulses of same breadth and height but it is displaced in time from some base position according to the amplitude of the signal at time of sampling. Position of pulse is 1:1 which is proportional to the width of the pulse and also directly proportional to the instantaneous amplitude of the sampled modulating signal.

It requires a pulse width generator and a monostable multivibrator. The spectrum obtain contains fixed DC component. A set of carrier and phase modulated sidebands at each harmonic of carrier signal.

Applications of PWM:

6. RF communications, like contactless smart cards, RFID, etc.
7. Non-coherent detection where a receiver does not need any PLL for tracking the phase.

Advantages of PWM:

4. Low noise interference when compared to PAM.
5. Noise removal and separation is very easy in PPM.
6. Power usage is very low.

Disadvantages of PWM:

3. Synchronization between transmitter and receiver is required.
4. Large bandwidth required.
5. Special Equipments are needed.

Plotting the waveform of PPM signal in SCILAB:

Code:

```
clear ;
t = 0:0.001:1;
fc = input ("Enter Carrier Signal Frequency = ");
c = (2)*asin(sin(2*%pi*fc*t));
fm = input("Enter Message Signal Frequency = ");
m = sin(2*%pi*fm*t);
n = length(c);
for i = 1 : n
   if (m(i) >= c(i))
      ppm(i) = 0;
   else (m(i) <= c(i))
      ppm(i)=1;
   end
end
figure;
subplot(3 ,1 ,1);
plot(t,m);
xlabel("Time");
ylabel("Amplitude");
title("Message Signal");
subplot(3, 1 ,2);
plot(t,c);
xlabel("Time");
ylabel("Amplitude");
title("Carrier Singal");
```

```
for i = 1 : n
   if (ppm(i) == 1 && ppm(i+1) == 0)
      ppm(i) = 1;
   else
      ppm(i) = 0;
   end
end
subplot(3, 1, 3);
plot(t,ppm');
xlabel("Time");
ylabel("Amplitude");
replot([0 -1 1 3]);
xlabel("Time");
ylabel("Amplitude");
title ("Pulse Position Modulation Signal");
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. $f_c = 10$, $f_m = 2$
2. $f_c = 50$, $f_m = 2$

Paste the output plots, and give your comments on same. Put down the conclusion for same.

Assignment:

Compare PAM, PWM, and PPM. Write down your observations, conclusion, by comparing the same. Also, the comparison must include fields of comparison like SNR, Immune towards noise, bandwidth dependency, etc.

LAB 7

Amplitude Shift Keying (ASK)

Aim: To study amplitude shift keying (ASK) technique by plotting the waveforms for same.

Theory:

Amplitude shift keying (ASK) with respect to the context of digital communications is a modulation process which imparts to a sinusoid two or more discrete amplitude levels. It is also called as on-off keying (OOK). These are related to the number levels adopted by the digital message. In amplitude shift keying, the amplitude of the carrier signal is varied to create signal elements. Both frequency and phase remain constant while the amplitude changes. In ASK, the amplitude of the carrier assumes one of the two amplitudes dependent on the logic states of the input bit stream. The modulated signal can be expressed as:

$$x_o(t) = 0 \rightarrow \text{symbol "0"}$$

$$x_o(t) = A_c \cos w_c t \rightarrow \text{symbol "1"}$$

ASK is related to the number of levels adopted by the digital message. For a binary message sequence there are two levels, one of which is typically zero. Thus, the modulated waveform consists of bursts of a sinusoid.

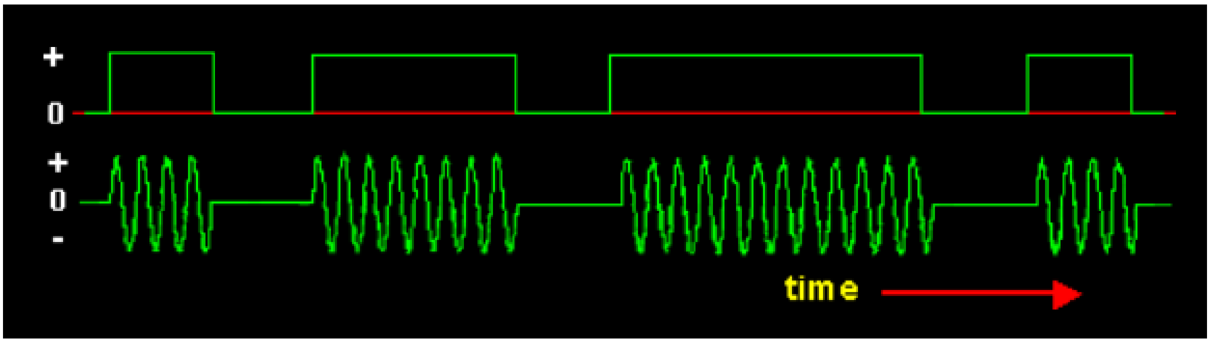

Fig. 7.1. – ASK signal (below) and the message signal (above)

There are sharp discontinuities shown at the transition points. These result in the signal having an unnecessarily wide bandwidth. Band limiting is generally introduced before transmission, in which case these discontinuities would be 'rounded off'. The band limiting may be applied to the digital message, or the modulated signal itself. The data rate is often made a sub-multiple of the carrier frequency.

Demodulation Methods:

It is apparent from the fig. 8.1. that ASK signal has a envelope, and thus it is amenable to demodulate the ASK modulated signal using an envelope detector. A synchronous demodulator can be used. Hence, the demodulation is a two stage-process:

1. Recovery of the bandlimited bit stream.
2. Regeneration of the binary bit stream.

Plotting the waveform of ASK modulated signal in SCILAB:

Code:

```
clear;
clc;

b = input("Enter the Bit stream = ");
n = max(size(mtlb_double(b)));
t = 0:0.01:n;
x = 1:1:(n+1)*100;

for i = 1:n
 for j = i:0.1:i+1
  bw(1,x(i*100:(i+1)*100)) = matrix(mtlb_e(b,i),1,-1);
 end;
end;

bw = bw(100:$);
sint = sin((2*%pi)*t);
```

```
st = mtlb_double(bw) .*sint;

subplot(3,1,1)
plot(t,bw)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("Message Signal (Bit Stream)")

subplot(3,1,2)
plot(t,sint)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("Carrier Signal")

subplot(3,1,3)
plot(t,st)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("ASK modulated signal")
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

6. [0 1 0 1 1 1 0]
7. [1 0 1 0 0 1 1]
8. [1 1 1 1 0 1 1]

Paste the output plots, and give your comments on same. Put down the conclusion for same.

Assignment:

Develop a script which demodulates the above ASK modulated signal. You can take the help of the above reference code for the modulation of the signal. One can also use SIMULINK or XCOS for same.

LAB 8

Binary Phase Shift Keying (BPSK)

Aim: To study binary phase shift keying (BPSK) technique by plotting the waveforms for same.

Theory:

Binary phase shift keying (BPSK) shifts the phase angle of the carrier frequency to one of two discrete phases during the bit time T_b for the representation of binary logic signals for the transmission of information. The information that is transmitted over a communication channel is impressed on the phase of the carrier. Since the range of the carrier phase is $0 \leq \theta \leq 2\Pi$, the carrier phases used to transmit digital information via digital-phase modulation are $\theta_m = 2\Pi m/M$, for m = 0, 1, 2,…, M-1.Thus for binary phase modulation (M=2), the two carrier phase are $\theta_0 = 0$ and $\theta_1 = \Pi$ radians. For M-array phase modulation=2k, where k is the number of information bits per transmitted symbol.

The general representation of a set of M carrier-phase-modulated signal waveforms is:

$$u_m(t) = AgT(t)\cos(2\Pi f_c t + 2\Pi m/M)$$

where m = 0,1,…,M-1; gT(t) is the transmitting filter pulse shape; A is the signal amplitude.

The transmitting filter pulse shape determines the spectral characteristics of the transmitted signal. This type of the modulation is called as binary phase-shift keying.

Plotting the waveform of BPSK modulated signal in SCILAB:

Code:

```
clear;

clc;

b = input("Enter the Bit stream = ");
n = max(size(mtlb_double(b)));
t = 0:0.01:n;
x = 1:1:(n+1)*100;

for i = 1:n
  if mtlb_logic(mtlb_double(mtlb_e(b,i)),"==",0) then
    b_p(1,i) = -1;
  else
    b_p(1,i) = 1;
  end;

  for j = i:0.1:i+1
    bw(1,x(i*100:(i+1)*100)) = matrix(b_p(i),1,-1);
  end;
end;

bw = bw(100:$);
sint = sin((2*%pi)*t);
st = mtlb_double(bw) .*sint;

subplot(3,1,1)
plot(t,bw)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("Message signal");

subplot(3,1,2)
plot(t,sint)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("Carrier signal");
```

```
subplot(3,1,3)
plot(t,st)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
title("BPSK modulated signal")
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. [0 1 0 1 1 1 0]
2. [1 0 1 0 0 1 1]
3. [1 1 1 1 0 1 1]

Paste the output plots, and give your comments on same. Put down the conclusion for same.

Assignment:

Develop a script which demodulates the above BPSK modulated signal. You can take the help of the above reference code for the modulation of the signal. One can also use SIMULINK or XCOS for same.

LAB 9

Quadrature Phase Shift Keying (QPSK)

Aim: To study quadrature phase shift keying (QPSK) technique by plotting the waveforms for same.

Theory:

Quadrature Phase Shift Keying (QPSK) is the digital modulation technique. Quadrature Phase Shift Keying (QPSK) is a form of Phase Shift Keying in which two bits are modulated at once, selecting one of four possible carrier phase shifts (0, Π/2, Π, and 3Π/2). QPSK perform by changing the phase of the In-phase (I) carrier from 0° to 180° and the Quadrature-phase (Q) carrier between 90° and 270°. This is used to indicate the four states of a 2-bit binary code. Each state of these carriers is referred to as a Symbol.

QPSK perform by changing the phase of the In-phase (I) carrier from 0° to 180° and the Quadrature-phase (Q) carrier between 90° and 270°. This is used to indicate the four states of a 2-bit binary code. Each state of these carriers is referred to as a Symbol. Quadrature Phase-shift Keying (QPSK) is a widely used method of transferring digital data by changing or modulating the phase of a carrier signal. In QPSK digital data is represented by 4 points around a circle which correspond to 4 phases of the carrier signal. These points are called symbols.

Plotting the waveform of QPSK modulated signal in SCILAB:

Code:

```
clear;
clc;

b = input("Enter the Bit stream = ");
n = max(size(double(b)));
t = 0:0.01:n;
x = 1:1:(n+2)*100;

for i = 1:n
 if mtlb_logic(mtlb_double(mtlb_e(b,i)),"==",0) then
  b_p(1,i) = -1;
 else
  b_p(1,i) = 1;
 end;
 for j = i:0.1:i+1
  bw(1,x(i*100:(i+1)*100)) = matrix(b_p(i),1,-1);
  if pmodulo(i,2)==0 then
   bow(1,x(i*100:(i+1)*100)) = matrix(b_p(i),1,-1);
   bow = mtlb_i(bow,x((i+1)*100:(i+2)*100),b_p(i));
  else
   bew(1,x(i*100:(i+1)*100)) = matrix(b_p(i),1,-1);
   bew = mtlb_i(bew,x((i+1)*100:(i+2)*100),b_p(i));
  end;
  if pmodulo(n,2)<>0 then
   bow = mtlb_i(bow,x(n*100:(n+1)*100),-1);
   bow = mtlb_i(bow,x((n+1)*100:(n+2)*100),-1);
  end;
 end;
end;
//be = b_p(1:2:$);
//bo = b_p(2:2:$);
bw = bw(100:$);
bew = bew(100:(n+1)*100);
bow = bow(200:(n+2)*100);
cost = cos((2*%pi)*t);
```

```
sint = sin((2*%pi)*t);
st = mtlb_a(mtlb_double(bew) .*cost,mtlb_double(bow) .*sint);
subplot(4,1,1)
plot(t,bw)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,2)
plot(t,bow)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,3)
plot(t,bew)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,4)
plot(t,st)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. [0 1 0 0 1 1 1 0]
2. [1 0 1 1 0 0 0 1]
3. [1 1 1 1 0 0 1 1]

Paste the output plots, and give your comments on same. Put down the conclusion for same.

Assignment:

Develop a script which demodulates the above QPSK modulated signal. You can take the help of the above reference code for the modulation of the signal. One can also use SIMULINK or XCOS for same.

LAB 10

Frequency Shift Keying (FSK)

Aim: To study frequency shift keying (FSK) technique by plotting the waveforms for same.

Theory:

Binary frequency shift keying (FSK) shifts the carrier frequency to one of two discrete frequencies during the bit time T_b for the representation of binary logic signals for the transmission of information. The modulated sinusoidal carrier signal has an amplitude of A voltage, a frequency of f_c Hz, and a 0° reference phase angle, as given by the analytical expression

$$m_i(t) = A\sin(2\pi(f_c + k_f m_i(t))t)$$

In frequency-shift keying, the signals transmitted for marks (binary ones) and spaces (binary zeros) are respectively.

$$m_1(t) = A\cos(\omega_1 t + \theta_c);\ 0 < t <= T$$

$$m_2(t) = A\cos(\omega_2 t + \theta_c);\ 0 < t <= T$$

This is called a discontinuous phase FSK system, because the phase of the signal is discontinuous at the switching times. A signal of this form can be generated by the following system. If the bit intervals and the phases of the signals can be determined (usually by the use of a phase-lock loop), then the signal can be decoded by two separate matched filters.

Plotting the waveform of FSK modulated signal in SCILAB:

Code:

```
clear;
clc;
b = input("Enter the Bit stream = ");
n = max(size(double(b)));
t = 0:0.01:n;
x = 1:1:(n+1)*100;
for i = 1:n
 if mtlb_logic(double(mtlb_e(b,i)),"==",0) then
  b_p(1,i) = -1;
 else
  b_p(1,i) = 1;
 end;
 for j = i:0.1:i+1
  bw(1,x(i*100:(i+1)*100)) = matrix(b_p(i),1,-1);
 end;
end;
bw = bw(100:$);
wo = 2*((2*%pi)*t);
W = 1*((2*%pi)*t);
sinHt = sin(wo+W);
sinLt = sin(wo-W);
st = sin(mtlb_a(wo,double(bw) .*W));
subplot(4,1,1)
plot(t,bw)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,2)
plot(t,sinHt)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,3)
plot(t,sinLt)
set(gca(),"grid",[1,1]);
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
subplot(4,1,4)
plot(t,st)
set(gca(),"grid",[1,1]);
```

```
set(gca(),"data_bounds",matrix([0,n,-2,2],2,-1))
```

Run the above code in SCILAB, observe the output and write down your observation and conclusion.

Test cases:

1. [0 1 0 0 1 1 1 0]
2. [1 0 1 1 0 0 0 1]
3. [1 1 1 1 0 0 1 1]

Paste the output plots, and give your comments on same. Put down the conclusion for same.

Assignment:

Develop a script which demodulates the above FSK modulated signal. You can take the help of the above reference code for the modulation of the signal. One can also use SIMULINK or XCOS for same.