Communication System Practicals

Sahil Bondre

U18CO021

BTech 2nd Year

Computer Engineering Department

Index

Sno.	**Practical Name**
1.	Introduction to MATLAB and plot basic functions and signals like sine, cosine, tangent, unit impulse, unit step, unit ramp, and periodic signals like impulse train, square wave and triangular wave.
2.	To perform Sampling and Reconstruction of signal (Hardware) and obtain its waveforms. Also Verify the Nyquist Criteria.
3.	Write a program to compute exponential fourier series coefficients and plot the magnitude and phase spectrum. Also, plot the periodic signal using fourier series.
4.	To perform Amplitude modulation and demodulation (Hardware) and obtain its waveforms. Also calculate the three different modulation indices.
5.	a) To perform Pulse Amplitude Modulation: (Hardware) a. To modulate signal by Pulse Amplitude Modulation Scheme using Natural & Flat top sampling. b. To demodulate signal by Pulse Amplitude Modulation Scheme using Sample & Hold, Flat Top. c. Verify the sampling theorem by changing modulating & carrier frequency b) To perform Pulse Position Modulation and Demodulation and obtain its waveforms. (Hardware) c) To perform Pulse Width Modulation and Demodulation and obtain its waveforms(Hardware)
6.	To study frequency modulation and demodulation and observe the waveforms. a) Observe the spectra of FM signal in labAlive virtual communication lab and Calculate the modulation index for FM b) To perform FM transmission via virtual lab labAlive for the audio signal c) To perform FM reception via virtual lab labAlive for the obtained recorded signal

7.	a) To Generate and demodulate an amplitude shift keying(ASK) signal in MATLAB. b) To study Frequency Shift Keying (FSK) Modulation in MATLAB Simulink. c) To study Binary Phase Shift Keying (BPSK) Modulation in MATLAB Simulink.
8.	Write a program for amplitude modulation and demodulation considering input as sinusoidal wave and plot the various signals in time domain and frequency domain (MATLAB)
9.	Write a MATLAB code to modulate and demodulate the given signal by Delta Modulation Technique.
10.	Write a program for frequency modulation and demodulation considering input as sinusoidal wave and plot the various signals in time domain and frequency domain in MATLAB
11.	To find the Numerical Aperture of given optical fiber in Virtual LAB.

27-01-2020

Experiment 1
Basic Functions and Signals in MATLAB

Sahil Bondre (u18co021)

Aim.

1. Plot basic functions: sin, cosine, exponent, tan in MATLAB.
2. Plot basic signals such as unit impulse, unit step and unit ramp.
3. Plot the periodic signals impulse train, square wave, sawtooth wave and triangular wave.

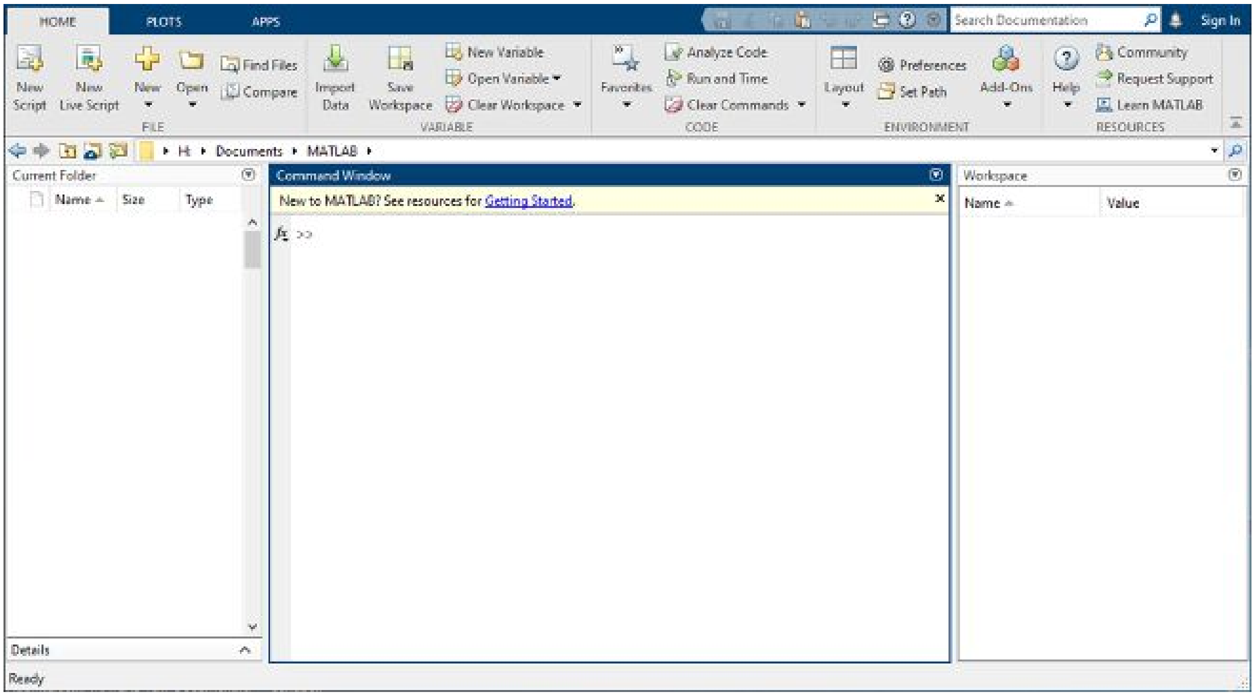
Theory.

MATLAB is a programming language developed by MathWorks. It started out as a matrix programming language where linear algebra programming was simple. It can be run both under interactive sessions and as a batch job.

Matlab Window

The desktop includes these panels:

- Current Folder — Access your files.
- Command Window — Enter commands at the command line, indicated by the prompt (>>).
- Workspace — Explore data that you create or import from files.



MATLAB Window

Features of MATLAB

- It is a high-level language for numerical computation, visualization and application development.
- It also provides an interactive environment for iterative exploration, design and problem solving.
- It provides a vast library of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and solving ordinary differential equations.
- It provides built-in graphics for visualizing data and tools for creating custom plots.
- MATLAB's programming interface gives development tools for improving code quality maintainability and maximizing performance.
- It provides tools for building applications with custom graphical interfaces.
- It provides functions for integrating MATLAB based algorithms with external applications and languages such as C, Java, .NET and Microsoft Excel.

Common Commands:

1. **clc**: Clears command window.

2. **clear:** Removes variables from memory.
3. **exist**: Checks for existence of file or variable.
4. **global**: Declares variables to be global.
5. **help**: Searches for a help topic.
6. **lookfor**: Searches help entries for a keyword.
7. **quit**: Stops MATLAB.
8. **who**: Lists current variables.
9. **whos**: Lists current variables (long display).
10. **cd**: Changes current directory.
11. **date**: Displays current date.
12. **delete:** Deletes a file.
13. **diary**: Switches on/off diary file recording.
14. **dir**: Lists all files in the current directory.
15. **load**: Loads workspace variables from a file.
16. **path**: Displays search path.
17. **pwd**: Displays current directory.
18. **save**: Saves workspace variables in a file.
19. **type**: Displays contents of a file.

Input/Output Commands:

1. **disp**: Displays contents of an array or string.
2. **fscanf**: Read formatted data from a file.
3. **format**: Controls screen-display format.
4. **fprintf** : Performs formatted writes to screen or file.
5. **input**: Displays prompts and waits for input.

Matrix Commands:

1. **cat**: Concatenates arrays.
2. **find**: Finds the indices of nonzero elements.

3. **length** : Computes number of elements.
4. **linspace:** Creates regularly spaced vector.
5. **max**: Returns the largest element.
6. **min**: Returns the smallest element.
7. **prod**: Product of each column.
8. **reshape**: Changes size.
9. **size**: Computes array size.
10. **sort**: Sorts each column.
11. **sum**: Sums each column.
12. **eye**: Creates an identity matrix.
13. **ones**: Creates an array of ones.
14. **zeros**: Creates an array of zeros.
15. **cross**: Computes matrix cross products.
16. **dot**: Computes matrix dot products.
17. **det**: Computes determinant of an array.
18. **inv**: Computes inverse of a matrix.
19. **rank**: Computes rank of a matrix.
20. **rref**: Computes reduced row echelon form.

Plot Commands

1. **axis**: Sets axis limits.
2. **grid**: Displays gridlines.
3. **plot**: Generates xy plot.
4. **print**: Prints plot or saves plot to a file.
5. **Title**: Puts text at top of plot.
6. **xlabel**: Adds text label to x-axis.
7. **ylabel**: Adds text label to y-axis.
8. **axes**: Creates axes objects.
9. **close**: Closes the current plot.
10. **close all:** Closes all plots.
11. **figure**: Opens a new figure window.
12. **hold**: Freezes current plot.

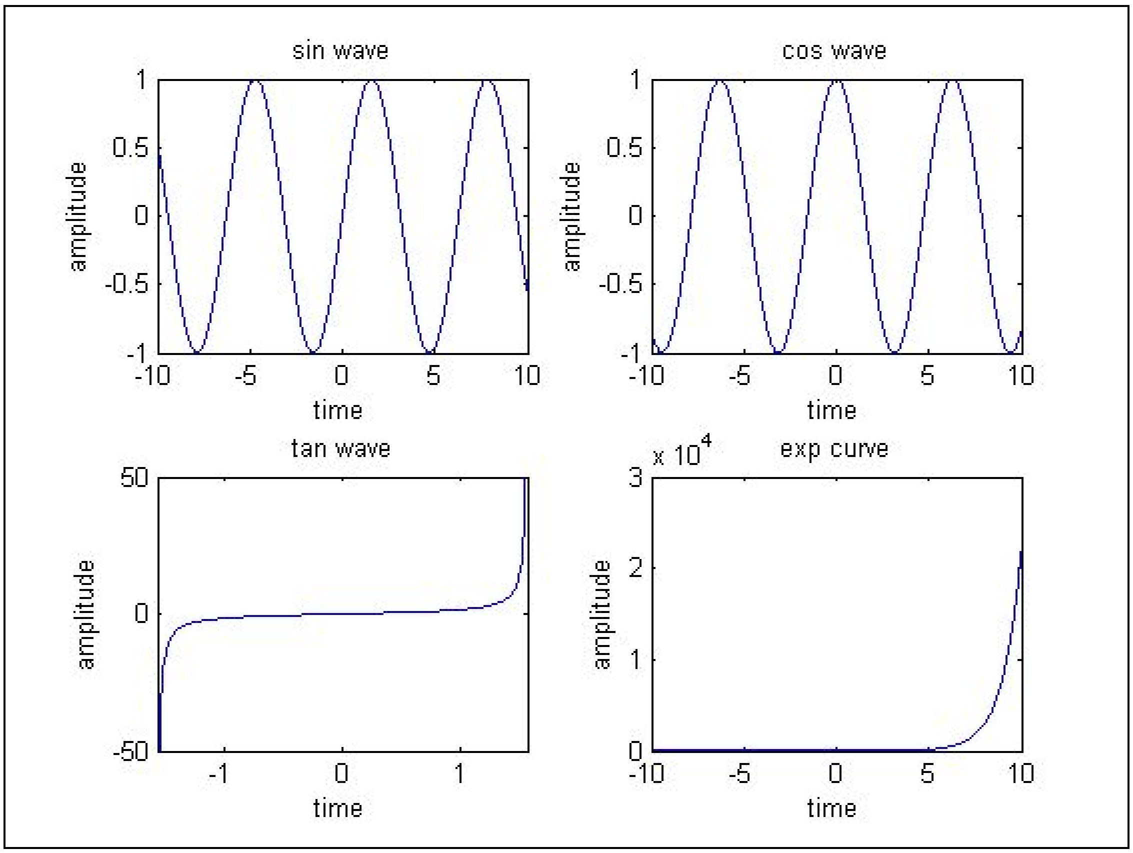
13. **legend**: Legend placement by mouse.
14. **refresh**: Redraws current figure window.
15. **set**: Specifies properties of objects such as axes.
16. **subplot**: Creates plots in subwindows.
17. **text**: Places string in figure.
18. **bar**: Creates bar chart.

Code.

1. Sin, Cosine, Tan and Exponential Functions

```
clc;

clear all;
close all;

t = -10:0.01:10;

x = sin(t);
y = cos(t);
z = tan(t);
w = exp(t);

% figure;
subplot(2, 2, 1);
plot(t, x);
xlabel('time');
ylabel('amplitude');
title('sin wave');

% figure;
subplot(2, 2, 2);
plot(t, y);
xlabel('time');
ylabel('amplitude');
title('cos wave');

subplot(2, 2, 3);
```

```
plot(t, z);
axis([-pi/2 pi/2 -50 50]);
xlabel('time');
ylabel('amplitude');
title('tan wave');

subplot(2, 2, 4);
plot(t, w);
xlabel('time');
ylabel('amplitude');
title('exp curve');
```

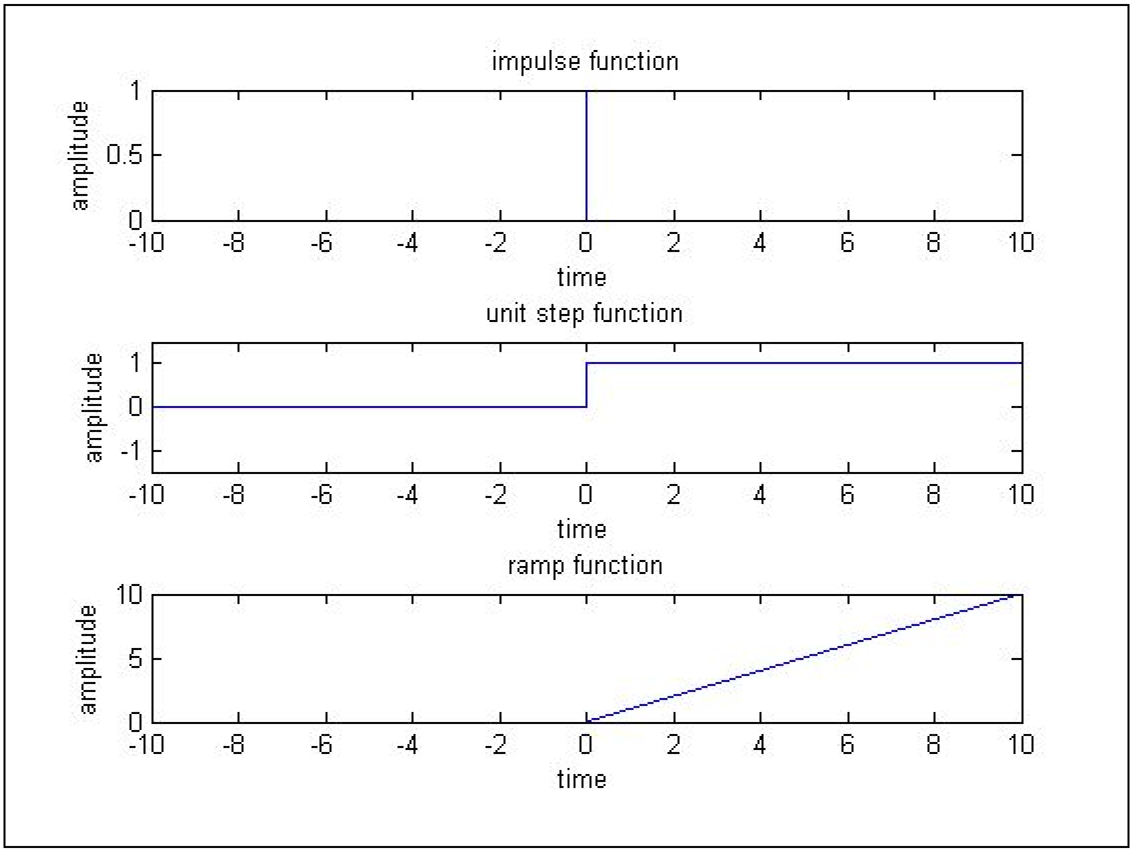


sin, cosine, tan and exponent curves plot

2. Unit Impulse, Unit Step and Unit Ramp Functions

```
clc;
clear all;
close all;
```

```
t = -10:0.01:10;
d = zeros(1, length(t));
e = zeros(1, length(t));
f = zeros(1, length(t));

d(t == 0) = 1;
e(t >= 0) = 1;
f(t >= 0) = t(t >= 0);

subplot(3, 1, 1);
plot(t, d);
xlabel('time');
ylabel('amplitude');
title('impulse function');

subplot(3, 1, 2);
plot(t, e);
axis([-10 10 -1.5 1.5]);
xlabel('time');
ylabel('amplitude');
title('unit step function');

subplot(3, 1, 3);
plot(t, f);
xlabel('time');
ylabel('amplitude');
title('ramp function');
```

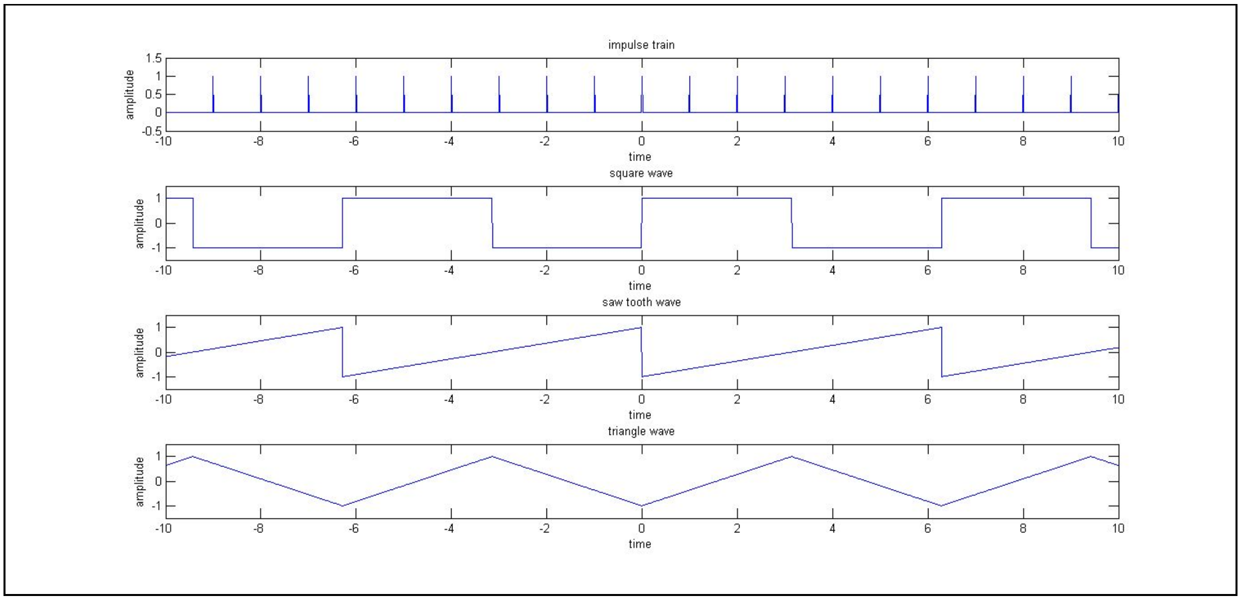


Impulse, unit step and ramp function

3. Periodic Signals: Impulse Train, Square Wave, Sawtooth Wave and Triangular Wave

```
clc;
clear all;
close all;

t = -10:0.01:10;
d = zeros(1, length(t));
period = 1;

d(mod(t, period) == 0) = 1;
e = square(t * period);
f = sawtooth(t * period);
g = sawtooth(t * period, 0.5);
```

```
subplot(4, 1, 1);
plot(t, d);
axis([-10 10 -0.5 1.5]);
xlabel('time');
ylabel('amplitude');
title('impulse train');

subplot(4, 1, 2);
plot(t, e);
axis([-10 10 -1.5 1.5]);
xlabel('time');
ylabel('amplitude');
title('square wave');

subplot(4, 1, 3);
plot(t, f);
axis([-10 10 -1.5 1.5]);
xlabel('time');
ylabel('amplitude');
title('saw tooth wave');

subplot(4, 1, 4);
plot(t, g);
axis([-10 10 -1.5 1.5]);
xlabel('time');
ylabel('amplitude');
title('triangle wave');
```



Impulse train, square wave, sawtooth and triangle wave

Conclusion.

In this experiment we generated basic functions like sine, cosine, tangent and exponential, basic signals like unit impulse, unit step and unit ramp and periodic signals like impulse train, square wave, sawtooth wave and triangular wave using MATLAB.

Remarks.

Signature.

05-02-2000

Experiment 2
Sampling and Reconstruction of Signals

Sahil Bondre (u18co021)

Aim.

1. To Study Sampling and Reconstruction of signal.
2. Verify Nyquist criteria.

Apparatus.

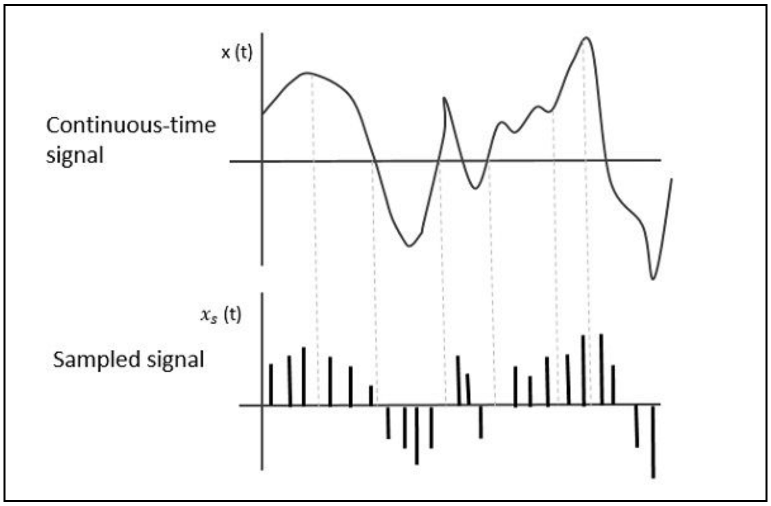
- Model ST 2151 W kit
- Connection wires
- CRO/DSO

Theory.

In signal processing, sampling is the reduction of a continuous-time signal to a discrete-time signal. A common example is the conversion of a sound wave (a continuous signal) to a sequence of samples (a discrete-time signal).

A sample is a value or set of values at a point in time and/or space.

The original signal is retrievable from a sequence of samples, up to the Nyquist limit, by passing the sequence of samples through a type of low pass filter called a reconstruction filter.



Signal Sampling Representation

To discretize the signals, the gap between the samples should be fixed. That gap can be termed as a sampling period Ts.

$$Sampling\ Frequency = 1/T_s = F_s$$

Where,

- Ts is the sampling time
- Fs is the sampling frequency or the sampling rate

Sampling frequency is the reciprocal of the sampling period. This sampling frequency can be simply called the Sampling rate. The sampling rate denotes the number of samples taken per second, or for a finite set of values.

For an analog signal to be reconstructed from the digitized signal, the sampling rate should be highly considered. The rate of sampling should be such that the data in the message signal should neither be lost nor it should get over-lapped. Hence, a rate was fixed for this, called the **Nyquist rate.**

Suppose that a signal is band-limited with no frequency components higher than W Hertz. That means, W is the highest frequency. For such a signal, for effective reproduction of the original signal, the sampling rate should be twice the highest frequency.

$$F_s = 2W$$

Where,

- Fs is the sampling rate
- W is the highest frequency

This rate of sampling is called the **Nyquist rate.**

Procedure.

A. Set up for Sampling and reconstruction of signal.

Initial set up of trainer:

Duty cycle selector switch position : Position 5

Sampling selector switch : Internal position

1. Connect the power cord to the trainer. Keep the power switch in ‘Off’ position.
2. Connect 1 KHz Sine wave to signal Input as shown in Fig 1.
3. Switch ‘On’ the trainer's power supply & Oscilloscope.
4. Connect BNC connector to the CRO and to the trainer’s output port.
5. You can observe the process of step-by-step generating sine wave signal from Square wave of 1 KHz at TP3, TP4, TP5 and TP6 respectively.

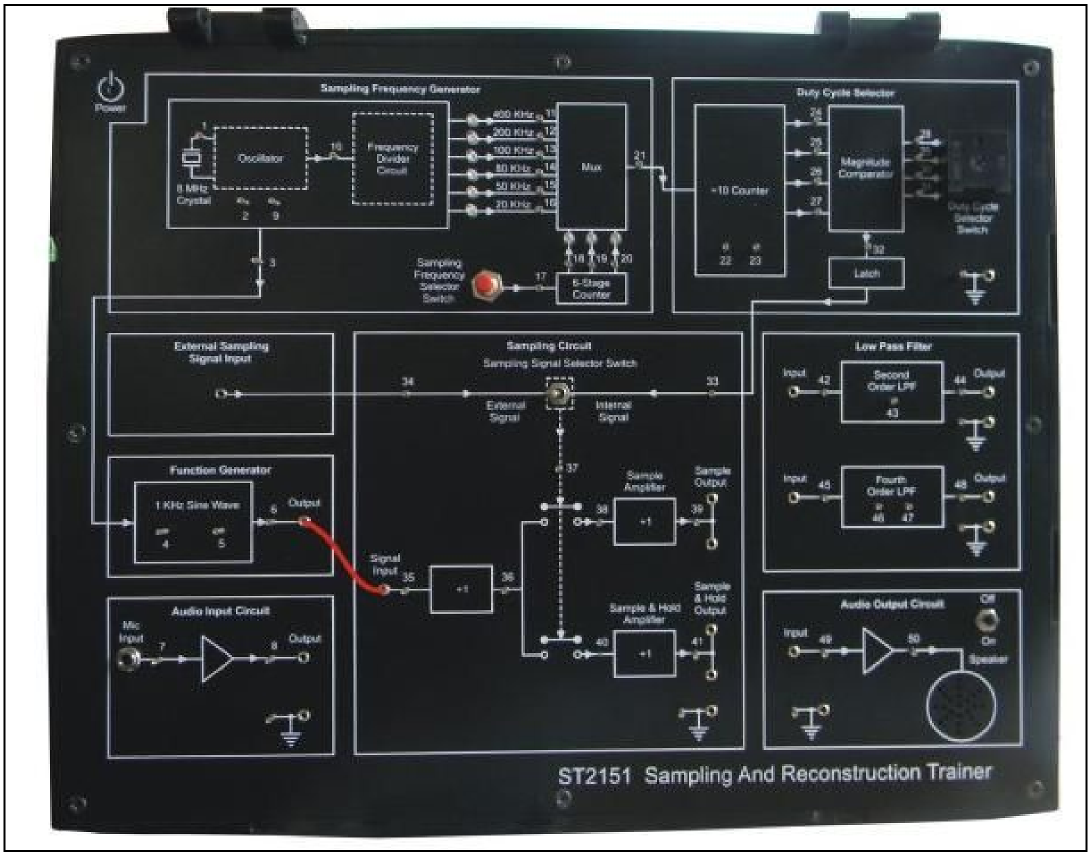


Fig 1. Connection diagram for sampling a signal

B. Set up for effect of Sample Amplifier and Sample and Hold Amplifier on reconstructed signal.

Set up for effect of II order and IV order Low Pass Filter on reconstructed signal.

Initial setup of trainer:

Duty cycle selector switch position : Position 5

Sampling selector switch: Internal position

1. Connect the power cord to the trainer. Keep the power switch in ‘Off’ position.
2. Connect 1 KHz Sine wave to signal Input.
3. Switch ‘On’ the trainer's power supply & Oscilloscope.
4. Connect BNC connector to the CRO and to the trainer’s output port.
5. Select sampling frequency of 5 KHz by Sampling Frequency Selector Switch pressed till 50 KHz signal LED glows.

6. Observe 1 KHz sine wave and Sample Output (TP39) on oscilloscope. The display shows

1 KHz sine wave being sampled at 5 KHz, so there are 5 samples for every cycle of the sine wave.

7. Connect Sample Output to Fourth Order low pass filter Input as shown in figure 1.2. Observe the filtered output (TP48) on the oscilloscope. The display shows the reconstructed 1 KHz sine wave.

8. Similarly observe the sampled 1 KHz sine wave at and Sample and Hold Output (TP41) on oscilloscope. The display shows 1 KHz sine wave being sampled and hold signal at 5 KHz. Connect Sample and Hold Output to Second Order low pass filter Input and observe the filtered output (TP44) on oscilloscope. The display shows the reconstructed1 KHz sine wave.

9. By pressing Sampling Frequency Selector Switch, change the sampling frequency from 2 KHz, 5 KHz, 10 KHz, 20 KHz up to 40 KHz (Sampling frequency is 1/10th of the frequency indicated by the illuminated LED). Observe how Sample output (TP39) and Sample and Hold Output (TP41) changes in each case.

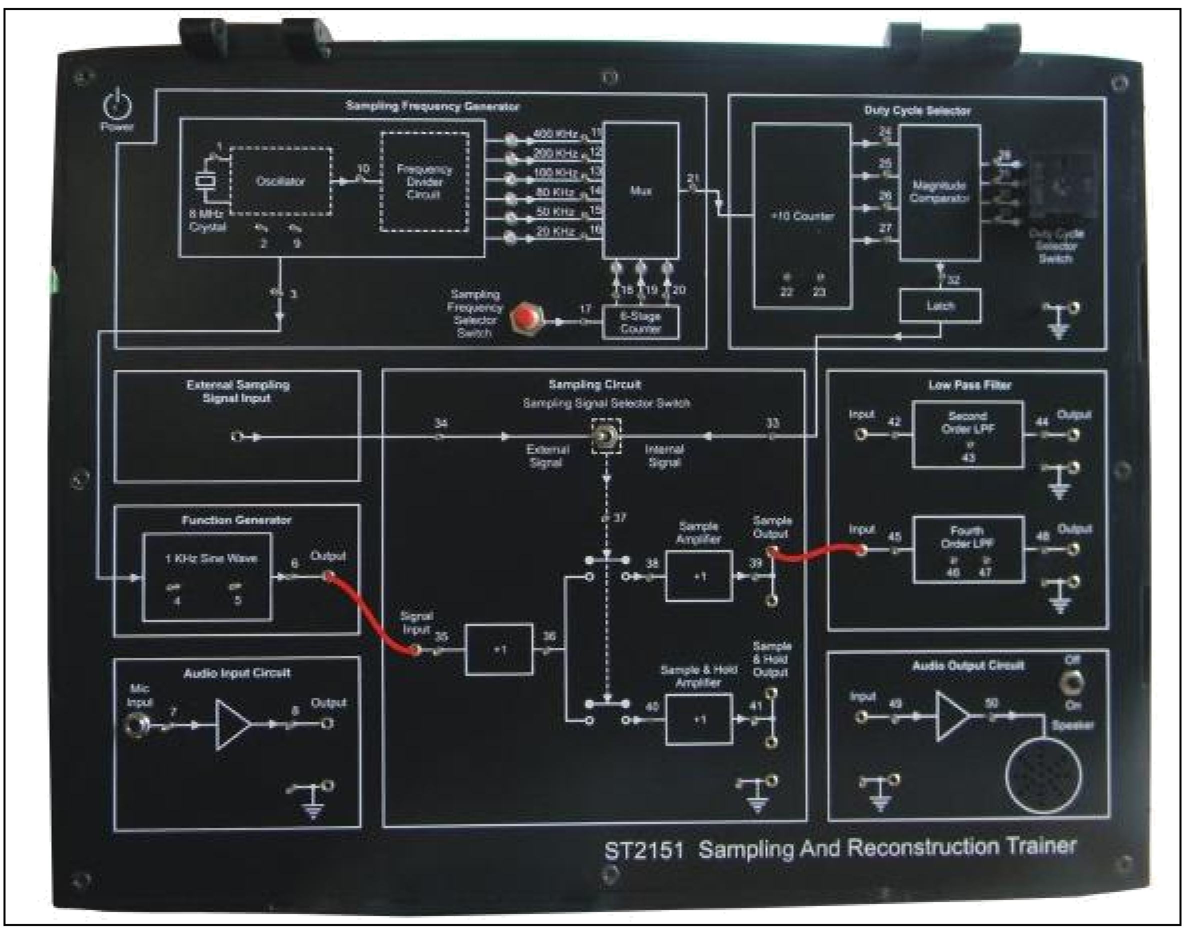
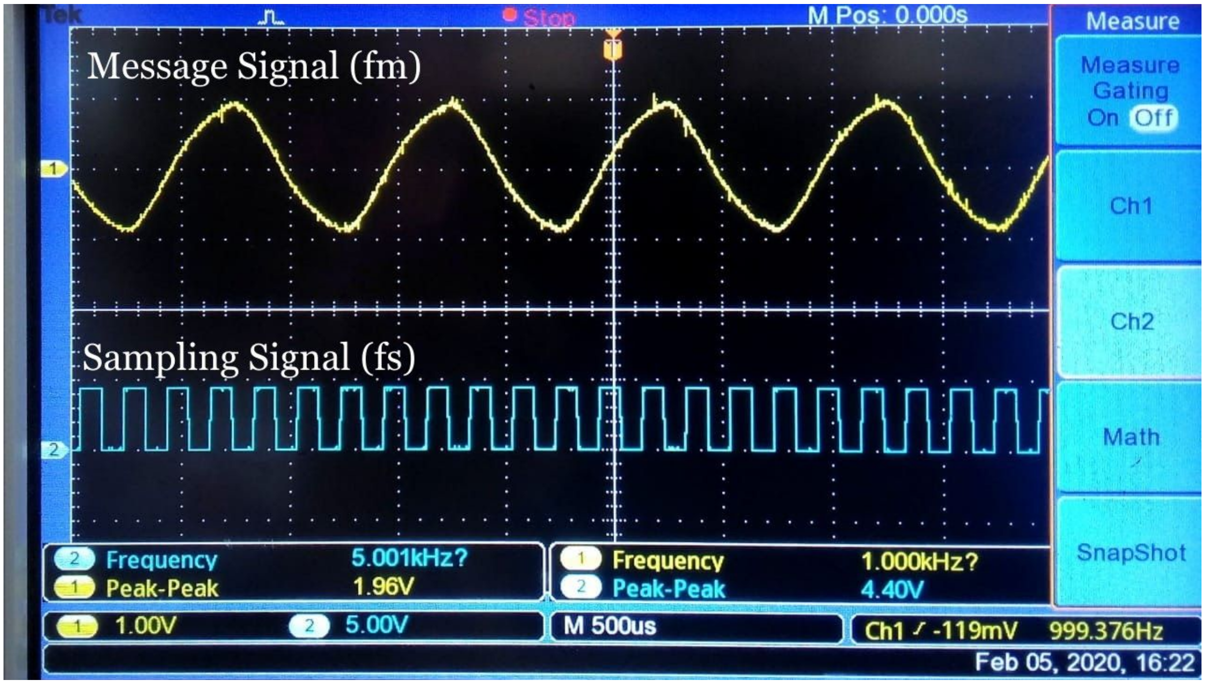


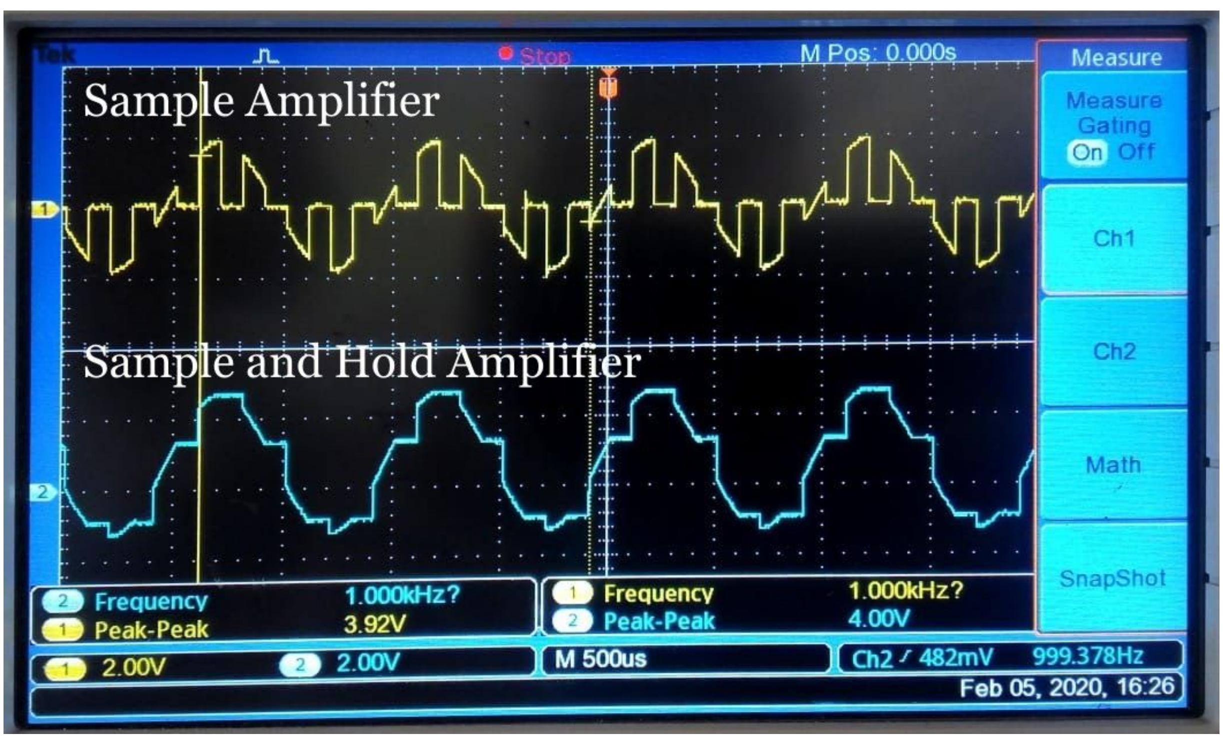
Fig 2. Connection diagram for reconstruction of a sampled signal

Observation.

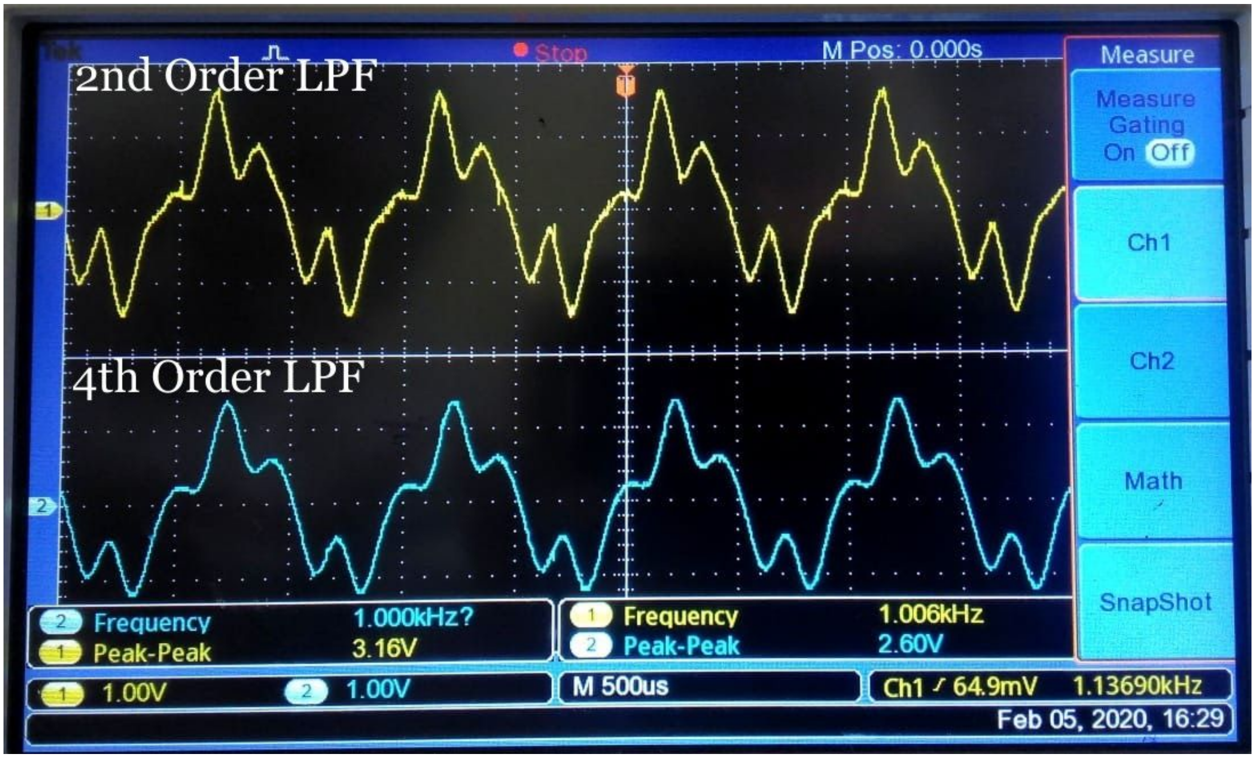
5 Khz Sample Frequency



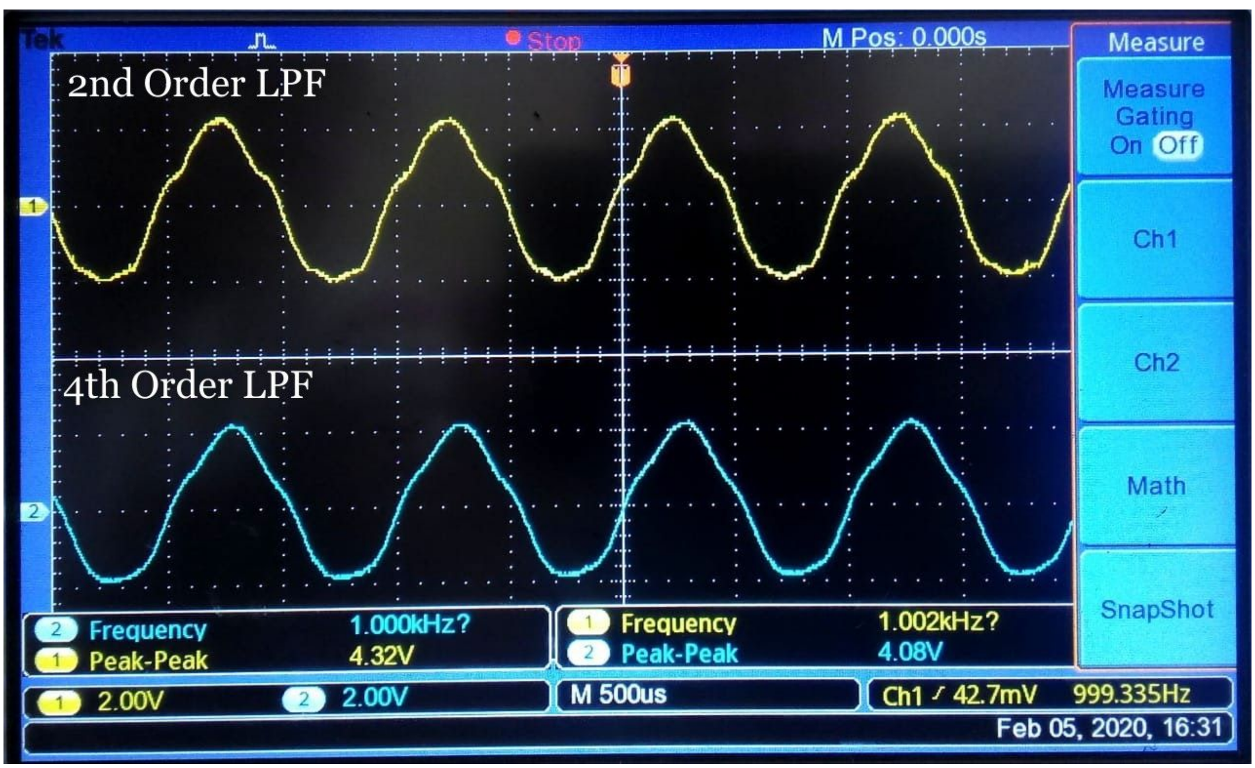
Message Sine Wave (fm) and Sample Signal (fs)



Sample Amplifier and Sample & Hold Amplifier Output

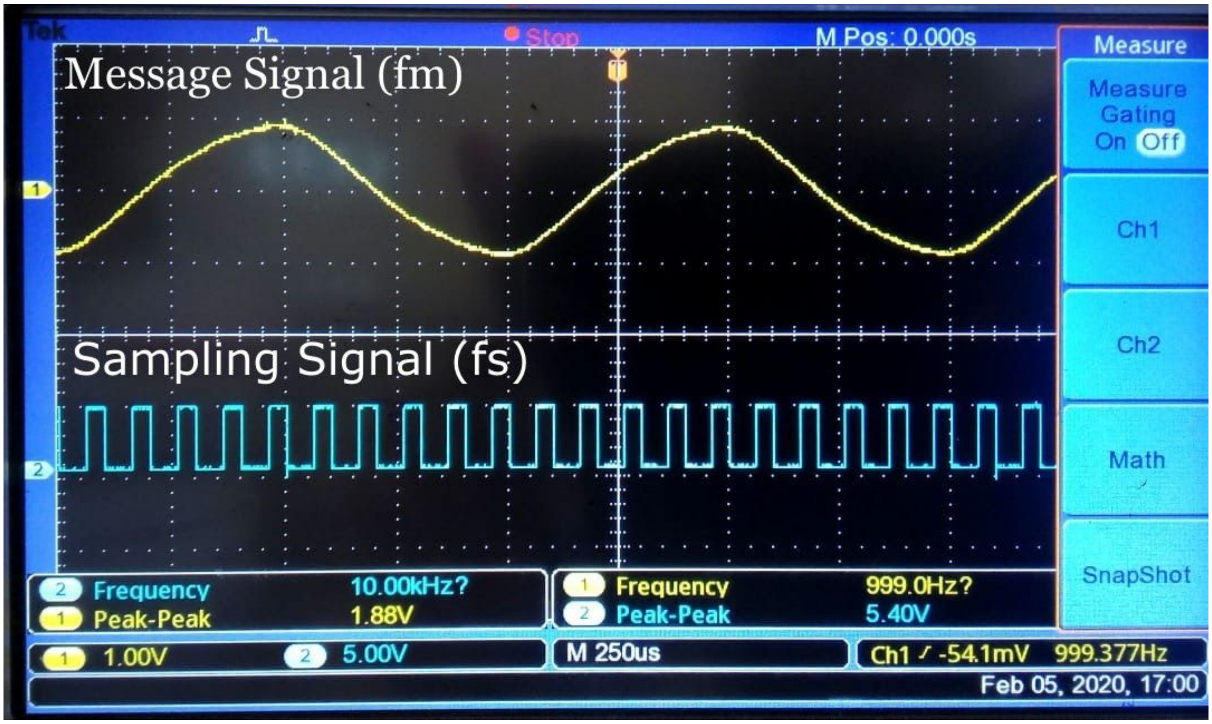


2nd and 4th Order Low Pass Filter Output for Sample Amplifier

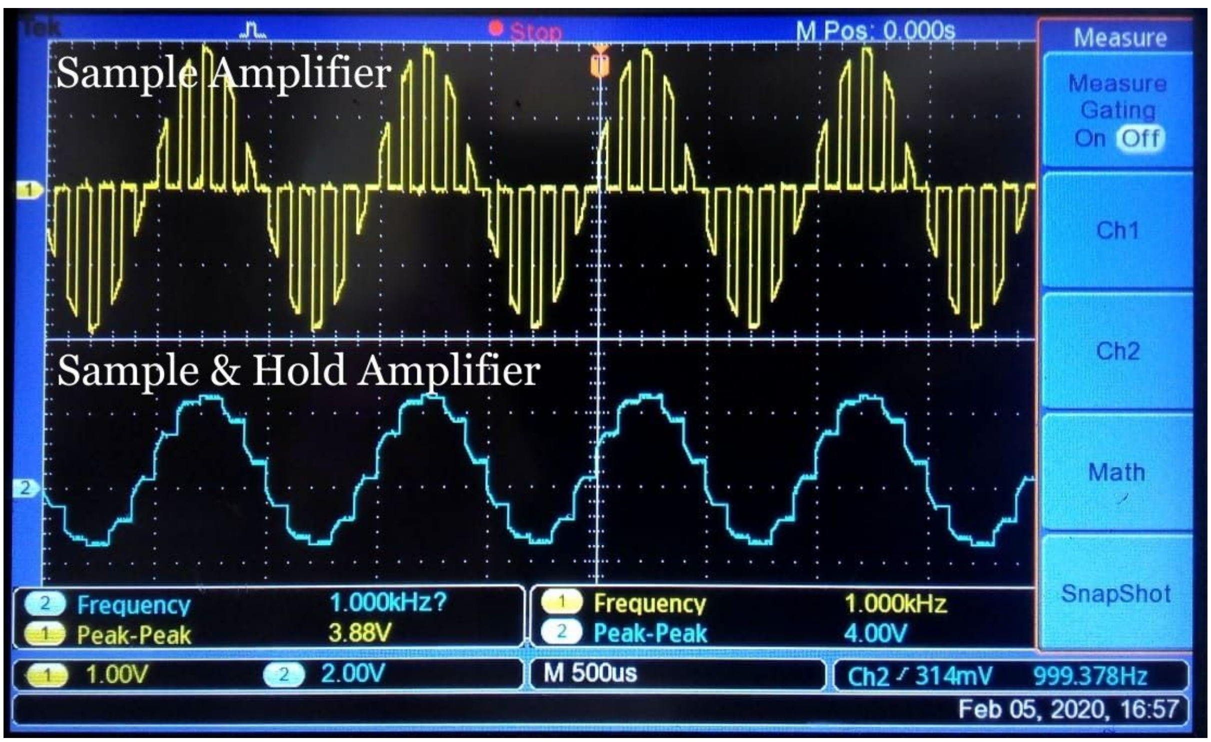


2nd and 4th Order Low Pass Filter Output for Sample & Hold Amplifier

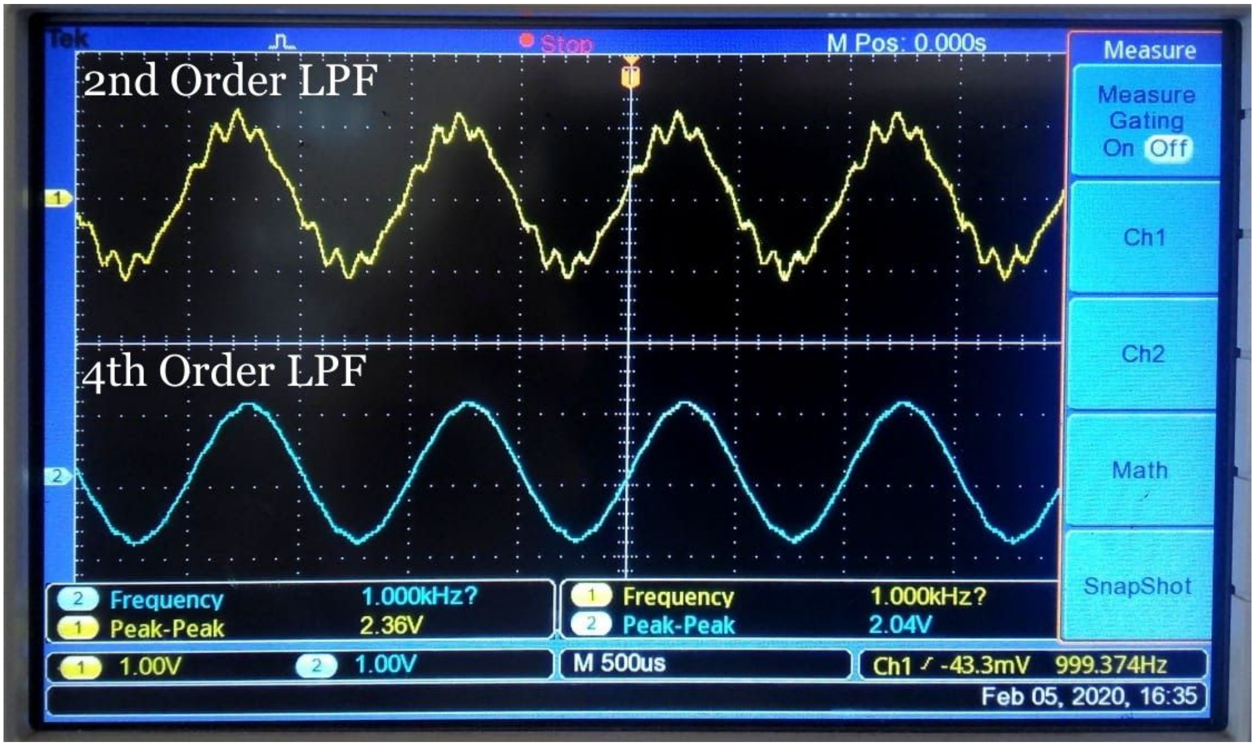
10 Khz Sample Frequency



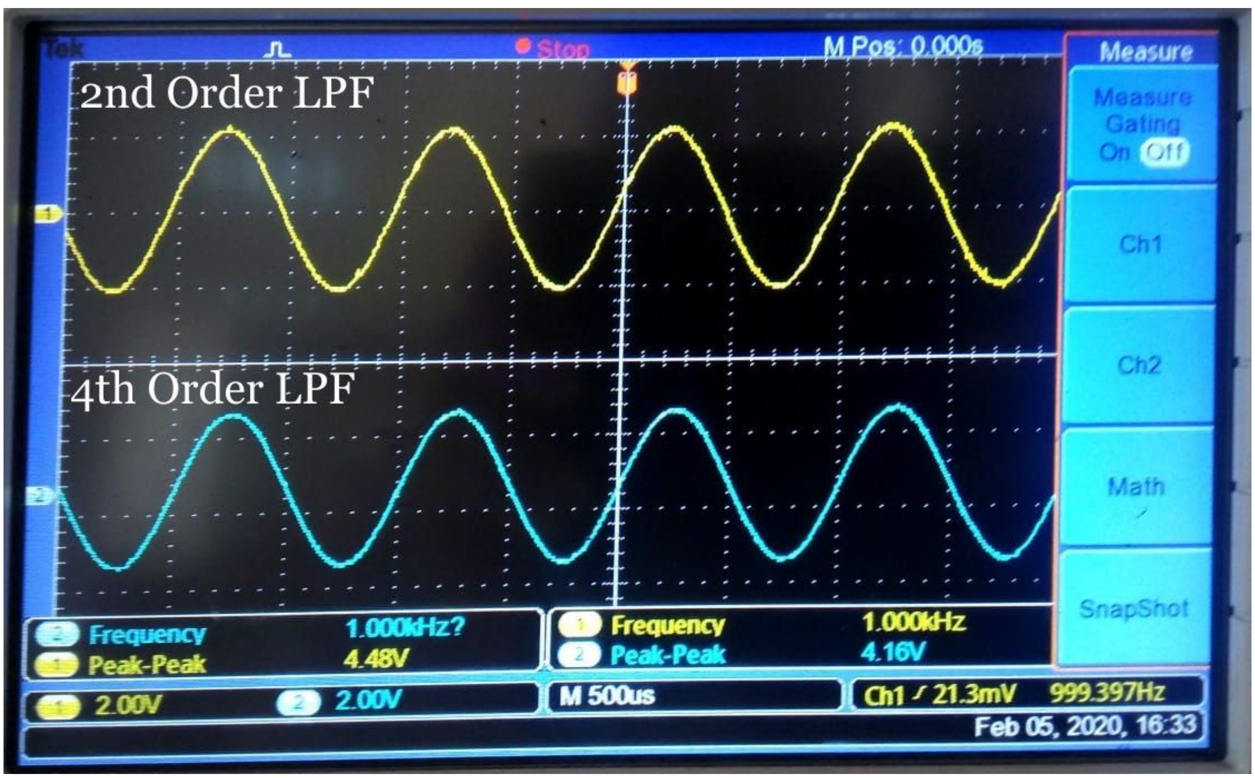
Message Sine Wave (fm) and Sample Signal (fs)



Sample Amplifier and Sample & Hold Amplifier Output



2nd and 4th Order Low Pass Filter Output for Sample Amplifier



2nd and 4th Order Low Pass Filter Output for Sample & Hold Amplifier

Conclusion.

In this experiment, we studied sampling and performed the reconstruction of 1 kHz sine signal using a sample frequency of 5 kHz and 10 kHz. We analysed the output of the sampled signal thru Sample Amplifier and Sample and Hold Amplifier. Finally, we reconstructed the original message signal by passing thru 2nd Order and 4th Order Low Pass Filters.

Remarks.

Signature.

12-02-2020

Experiment 3
Fourier Series in MATLAB

Sahil Bondre (u18co021)

Aim.

1. To compute the Fourier Coefficients of Exponential and Square Wave
2. Plot their Magnitude and Phase Spectrum

Theory.

Fourier Series

Fourier series is a periodic function composed of harmonically related sinusoids, combined by a weighted summation. With appropriate weights *(Fourier Coefficients)*, one cycle (or period) of the summation can be made to approximate an arbitrary function in that interval (or the entire function if it too is periodic). As such, the summation is a synthesis of another function. The discrete-time Fourier transform is an example of the Fourier series.

$$\begin{cases} a_0 = \dfrac{1}{2\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)dx, \\ a_k = \dfrac{1}{\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)\cos(kx)dx, \quad 1 \le k \le n \\ b_k = \dfrac{1}{\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)\sin(kx)dx, \quad 1 \le k \le n. \end{cases}$$

Integration in MATLAB

In contrast to differentiation, symbolic integration is a more complicated task. A number of difficulties can arise in computing the integral:

- The antiderivative, F, may not exist in closed form.

- The antiderivative may define an unfamiliar function.
- The antiderivative may exist, but the software can't find it.

Nevertheless, in many cases, MATLAB can perform symbolic integration successfully.

```
q = integral(fun,xmin,xmax)
```

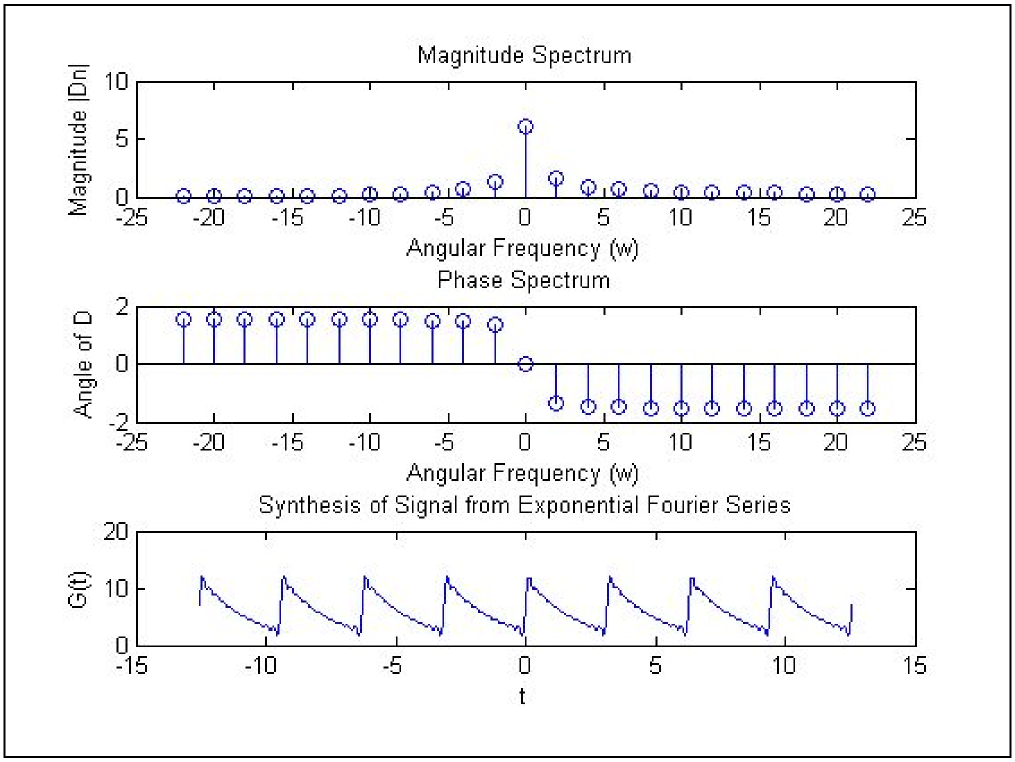
numerically integrates function `fun` from `xmin` to `xmax` using global adaptive quadrature and default error tolerances.

Code.

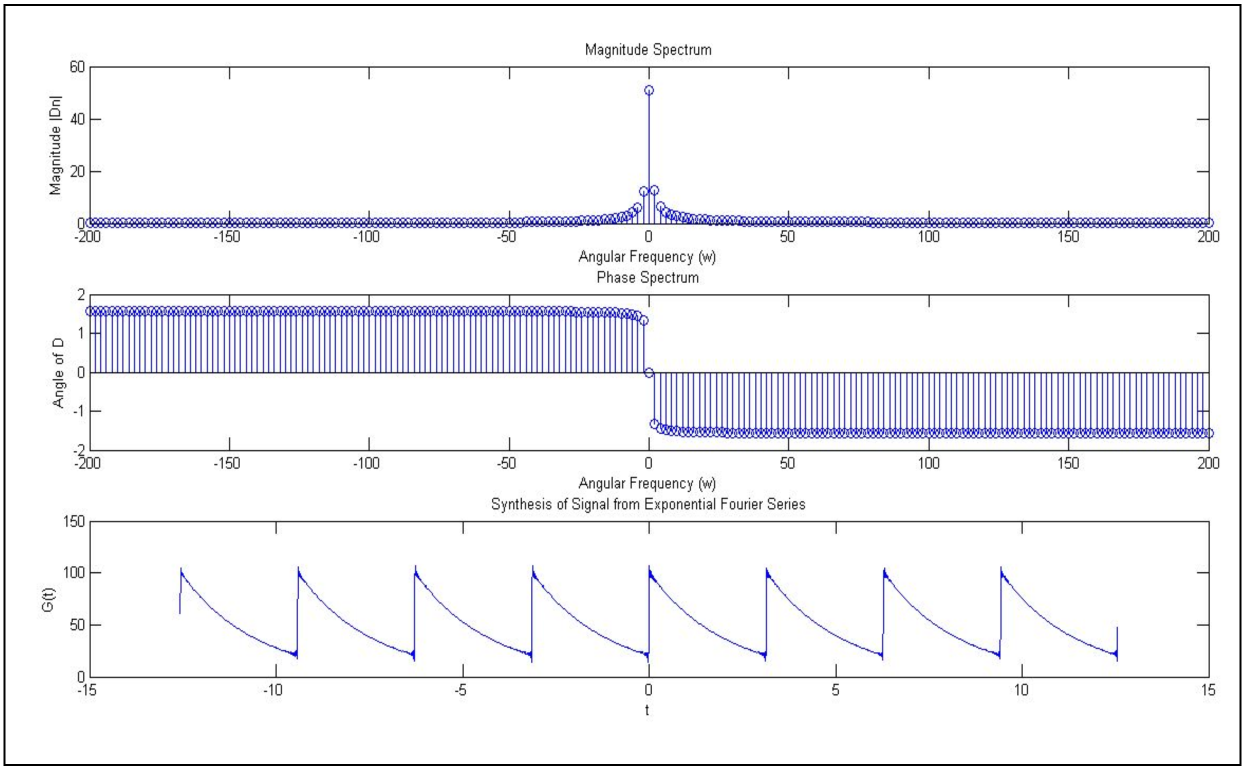
1. Exponent Function

```
clc;
clear all;
close all;

T0 = 2 * pi;
N = 100;
% N = 11;
w0 = 2*pi / T0;
t = -4 * pi : 0.01 : 4 * pi;
w = (-N: N) * (2 * pi / T0);

for i = 1:length(w)
      D(i) = i / T0 * integral(@(t)exp(-t/2).*exp(-1j*w(i)*t), 0, T0);
end

figure;
subplot(3, 1, 1);
stem(w, abs(D));
xlabel('Angular Frequency (w)');
ylabel('Magnitude |Dn|');
title('Magnitude Spectrum');

subplot(3, 1, 2);
stem(w, angle(D));
xlabel('Angular Frequency (w)');
ylabel('Angle of D');
title('Phase Spectrum');
```

```
sum = 0;
for i = 1:length(w)
      sum = sum + D(i)*exp(j*w(i)*t);
end

subplot(3, 1, 3);
plot(t, sum);
xlabel('t');
ylabel('G(t)');
title('Synthesis of Signal from Exponential Fourier Series');
```



Magnitude and Phase Spectrum for Exponent Curve N = 11

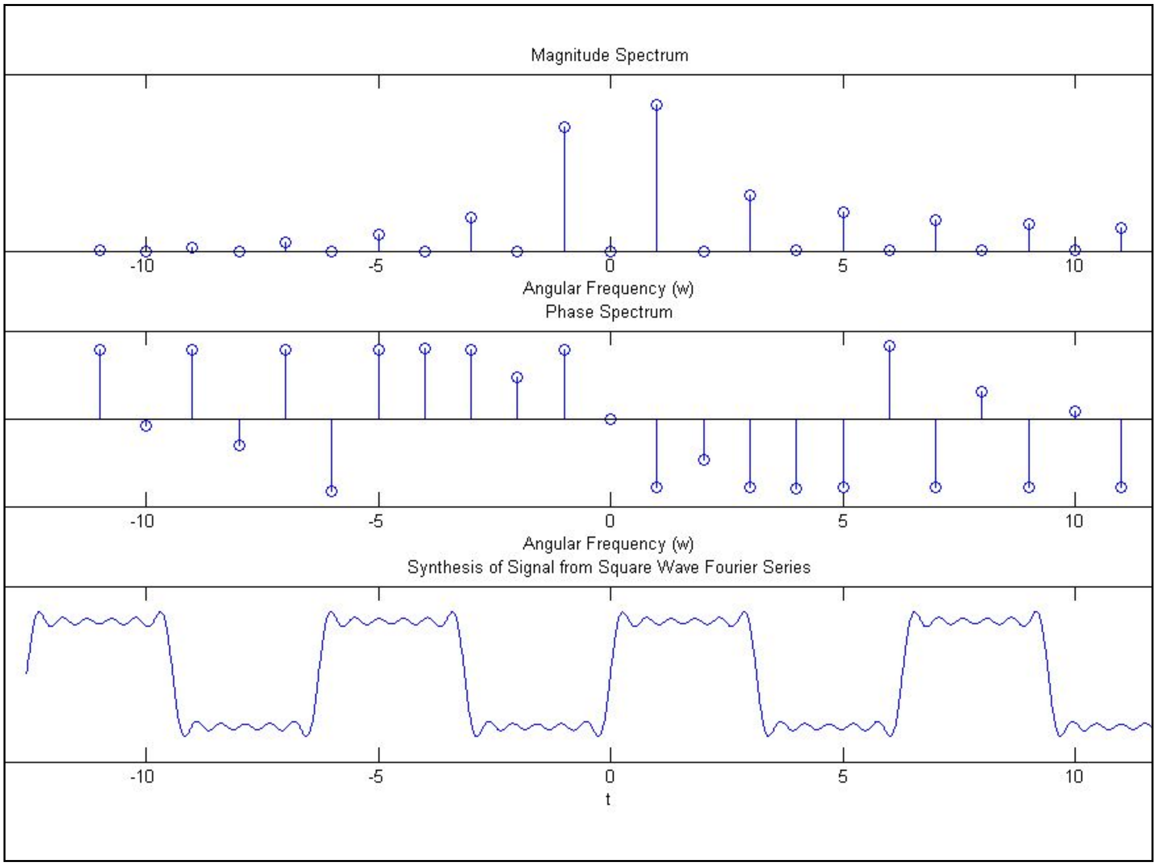


Magnitude and Phase Spectrum for Exponent Curve N = 100

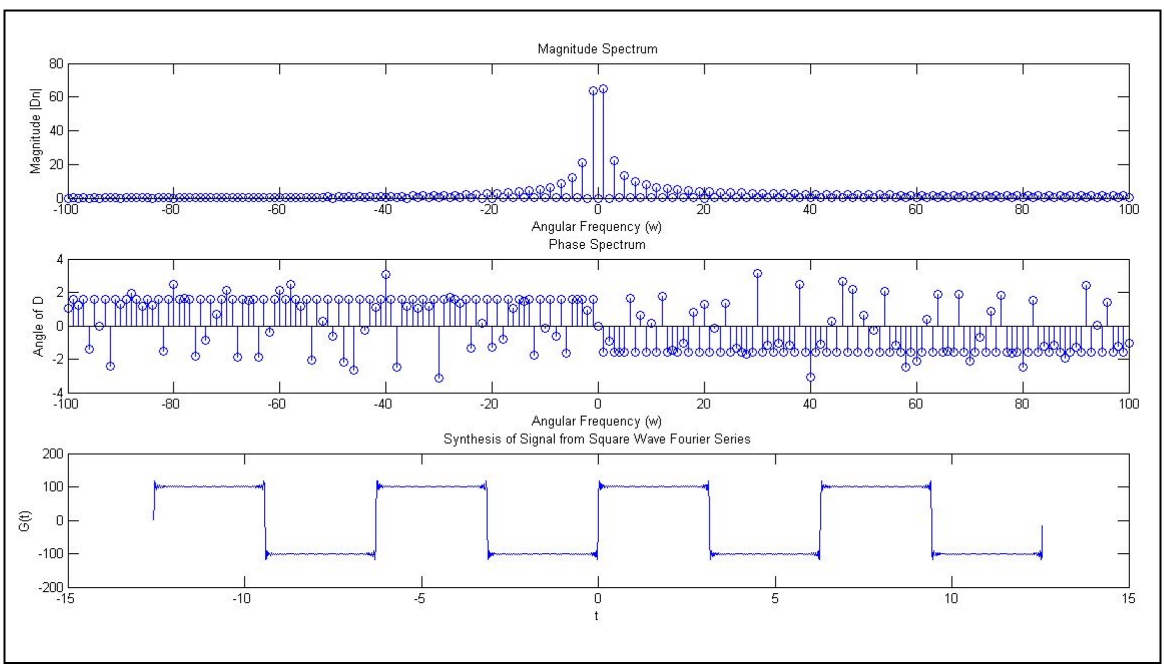
2. Square Wave

```
clc;
clear all;
close all;

T0 = 2 * pi;
N = 100;
% N = 11;
w0 = 2*pi / T0;
t = -4 * pi : 0.01 : 4 * pi;
w = (-N: N) * (2 * pi / T0);

for i = 1:length(w)
      D(i) = i / T0 * integral(@(t)square(t).*exp(-1j*w(i)*t), 0, T0);
end

figure;
subplot(3, 1, 1);
stem(w, abs(D));
xlabel('Angular Frequency (w)');
```

```
ylabel('Magnitude |Dn|');
title('Magnitude Spectrum');

subplot(3, 1, 2);
stem(w, angle(D));
xlabel('Angular Frequency (w)');
ylabel('Angle of D');
title('Phase Spectrum');

sum = 0;
for i = 1:length(w)
      sum = sum + D(i)*exp(j*w(i)*t);
end

subplot(3, 1, 3);
plot(t, sum);
xlabel('t');
ylabel('G(t)');
title('Synthesis of Signal from Square Wave Fourier Series');
```



Magnitude and Phase Spectrum for Square Wave N = 11



Magnitude and Phase Spectrum for Square Wave N = 100

Conclusion.

In this experiment we computed the Fourier Coefficients of Exponent and Square Wave Functions. Using these coefficients, we plotted the Phase Spectrum, Magnitude Spectrum and Synthesised the Original Signals.

Remarks. Signature.

19-02-2020

Experiment 4
Amplitude Modulation

Sahil Bondre (u18co021)

Aim.

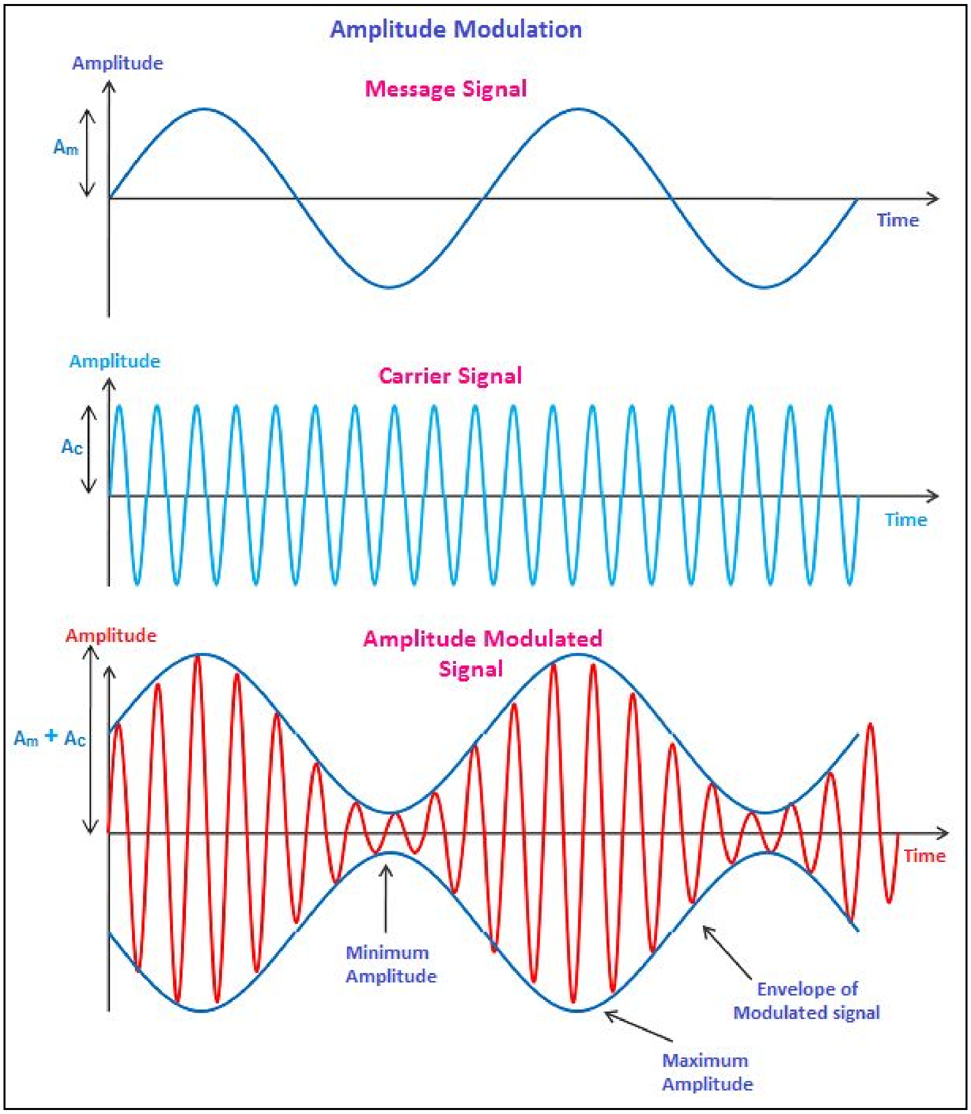
- To study amplitude modulation and observe the waveforms for three different modulation indices.

Apparatus.

- Trainer board ST 2201 & 2202
- Power supply
- Connecting Wires
- CRO
- Function Generator
- Carrier Generator.

Theory.

Amplitude modulation (AM) is a modulation technique used in electronic communication, most commonly for transmitting information via a radio carrier wave. In amplitude modulation, the amplitude (signal strength) of the carrier wave is varied in proportion to that of the message signal being transmitted. The message signal is, for example, a function of the sound to be reproduced by a loudspeaker, or the light intensity of pixels of a television screen. This technique contrasts with frequency modulation, in which the frequency of the carrier signal is varied, and phase modulation, in which its phase is varied.



Amplitude Modulation

Description of different blocks of Amplitude Modulation Circuit:

(1) Input audio amplifier section:

This section is used to amplify low level audio signals coming from Mic/Loudspeaker and give it to A.M. Modulator section for live A.M. modulation. It consists of a pre-amplifier stage and output amplifier stage. Transistor BC148B is used for pre-amplification. Input signal from the Mica is given to BC148B through coupling capacitor. Output of BC148B is given to pin 10 of IC 810 which contains audio amplifier, driver & output stage. The amplified output is obtained at pin 16 which can be used as modulating signal.

(2) Modulating audio signal generator section:

IC 8038 is used to generate sine wave signals. Pot P2 is used to vary its frequency .The range is 20 Hz to20KHz.Two 100k pots are used to adjust the peaks of the sine wave and 1K preset is used for duty cycle adjustment. The sine wave signal is available at pin 2 of IC8038. This signal is amplified by IC LM356.Pot P2 is used to vary the amplitude of sine wave signal.

(3) RF Carrier oscillator section:

Transistor BC107B is used to generate RF sine wave signals. Pot P1 is used to vary its frequency from 200 kHz to1MHz.Here transistor Q2, Q3, Q4 and Q5 is used to amplify the RF signal of Q1. Pot P2 is used to vary the amplitude of sine wave from 0 to 10 Vpp.

(4) Double balanced amplitude modulator section:

IC 1496 is used as a balanced modulator. The modulating audio signal is connected at pin 1 through buffer transistor Q1.This IC has two inputs as it works as a balanced modulator. The Second input can be connected at pin 4 through buffer transistor Q2.The RF carrier signal is connected at pin 8 through coupling capacitor from RF carrier oscillator section. The modulated outputs are available at pin 12 and 6 of this IC which are then balanced amplified by Q3, Q4, Q5 and Q6.The final balanced modulated output is available at output terminals. Bal-A preset is used to balance carrier signal while Bal-B preset is used to balance input audio signal. 1K preset is used to adjust output zero DC level.

(5) DC voltage generating section:

To observe the effect of dc voltages on AM modulating signal +1V dc and -8V to +8V dc voltage is required which is generated using IC741 and presets.

(6) Filter section:

Here notch filter of 455 KHz is designed using crystal. This filter is used to obtain suppressed carrier double sideband modulated signal from DSB signal.

(7) AM demodulators:

(a) Diode detector circuit:

This circuit consists detector diode OA79 and capacitor C1, C2,C3 and load resistor R1. It works as an envelope detector circuit.R1 and C forms a low pass filter meant to reduce the carrier frequency ripple in the output.

(b) Product detector:

This section is similar to AM balance modulator section. the difference is only that input pin 8 is given RF carrier oscillator signal from RF carrier oscillator and pin 1 is given

AM modulated signal from the balanced modulator section. The output is product of these two signals which contains basic audio modulating signal which can be filtered by low pass filter.

(8) Output audio amplifier section:

This section is same as input audio amplifier section except the pre-amplifier section.

(9) Power supply section:

The regulated power supply is used for different supply voltages. Using step down transformer, diode bridge and IC7805, 7815, 7915 we can obtain different DC supply voltages required for the operation of different blocks.

Procedure.

Modulation:

1. Fig. 1 shows the AM transmitter panel. Ensure that the following initial conditions exist on the board.
    1. Audio input select switch in INT position:
    2. Mode switch in DSB position.
    3. Output amplifier's gain pot in full clockwise position.
    4. Speakers switch in OFF position.

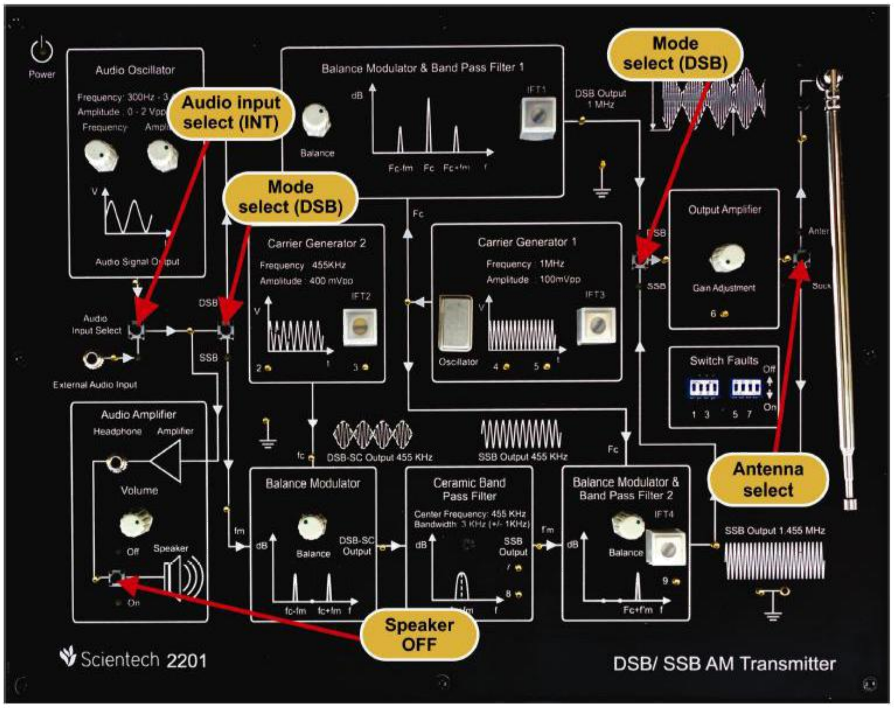


Fig. 1 AM Transmitter Panel

2. Turn on power to the **ST2201** board.
3. Turn the audio oscillator block's amplitude pot to its full clockwise (MAX) position, and examine the block's output (t.p.14) on an oscilloscope
4. Monitor, in turn, the two inputs to the balanced modulator & band pass filter circuits 1 block, at t.p.1 and t.p.9
5. Next, examine the output of the balanced modulator & band pass filter circuit 1 block (at t.p.3), together with the modulating signal at t.p.1 Trigger the oscilloscope on the t.p. 1 signal.
6. To determine the depth of modulation, measure the maximum amplitude (Vmax) and the minimum amplitude (V min) of the AM waveform at t.p.3, and use the following formula:

$$Percentage\ Modulation = \frac{V_{max} - V_{min}}{V_{max} + V_{min}}$$

Where Vmax and Vmin are the maximum and minimum amplitudes.

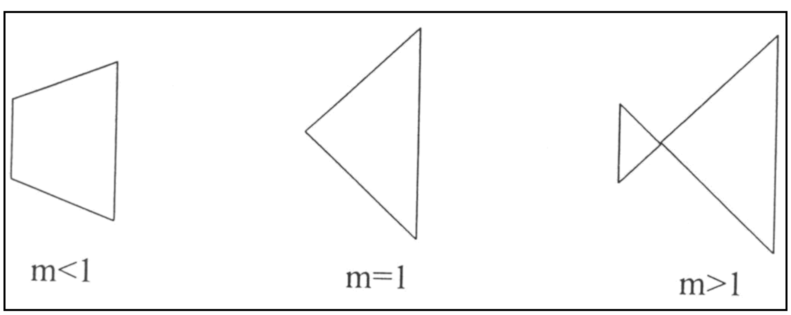
7. Now vary the amplitude and frequency of the audio-frequency sine wave, by adjusting the amplitude and frequency present in the audio oscillator block. Note the effect that varying each pot has on the amplitude modulated waveform. The amplitude and frequency amplitudes of the two sidebands can be reduced to zero by reducing the amplitude of the modulating audio signal to zero. Do this by turning the amplitude pot to its MIN position, and note that the signal at t.p. 3 becomes an un-modulated sine wave of frequency 1 MHz, indicating that only the carrier component now remains.

To calculate modulation index of DSB wave by trapezoidal pattern.

1. Repeat from step no. 1 to step no. 6
2. Now apply the modulated waveform to the Y input of the oscilloscope and the modulating signal to the X input.
3. Press the XY switch, you will observe the waveform similar to the one given below:
4. Calculate the modulation index by substituting in the formula

$$Percentage\ Modulation = \frac{V_{max} - V_{min}}{V_{max} + V_{min}}$$

4. Some common trapezoidal patterns for different modulation indices are as shown:



Demodulation:

1. Fig. 2 shows the AM receiver panel. Position the **ST2201** & **ST2202** modules, with the **ST2201** board on the left, and a gap of about three inches between them.

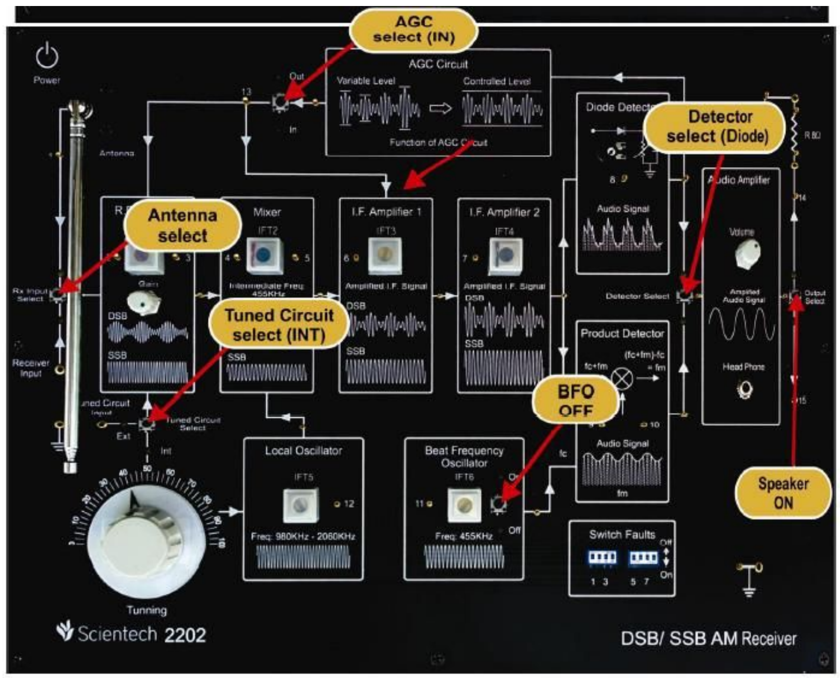


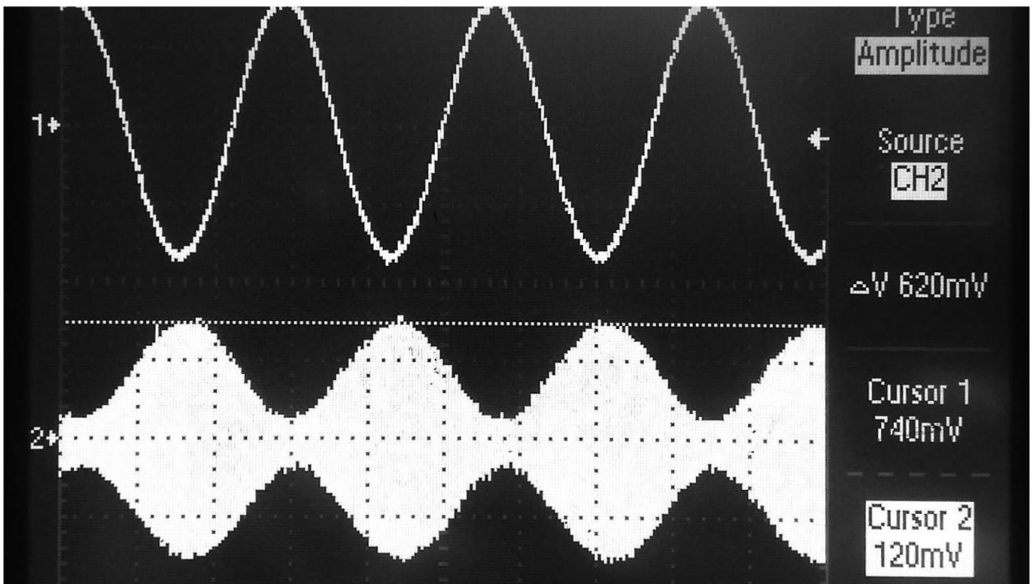
Fig. 2 AM RecieverPanel

2. Ensure that proper initial conditions exist on the **ST2201 & ST2202** board
3. Turn on power to the modules. We will now transmit the SSB waveform to the **ST2202** receiver. The mode of transmission can be selected by a selection switch (i.e. by an antenna or by a link).
4. On the **ST2202** module, monitor the output of the IF amplifier 2 block (t.p. 28) and turn the tuning dial until the amplitude of the monitored signal is at its greatest. Check that you have tuned into the SSB signal, by turning **ST2201**'s amplitude pot (in the audio oscillator block) to its MIN position, and checking that the monitored signal amplitude drops to zero. Return the amplitude pot to its MAX position.

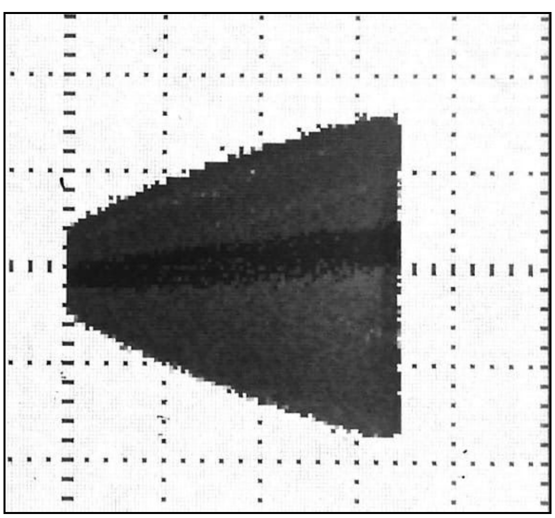
5. On the **ST2202** module, monitor the output of the product detector block (at t.p. 37), together with the output of the audio amplifier block (t.p. 39), triggering the scope with the later signal.

Observation.

m < 1



Signal and AM Wave



$V_{max} = 740\ mV$

$V_{min} = 120\ mV$

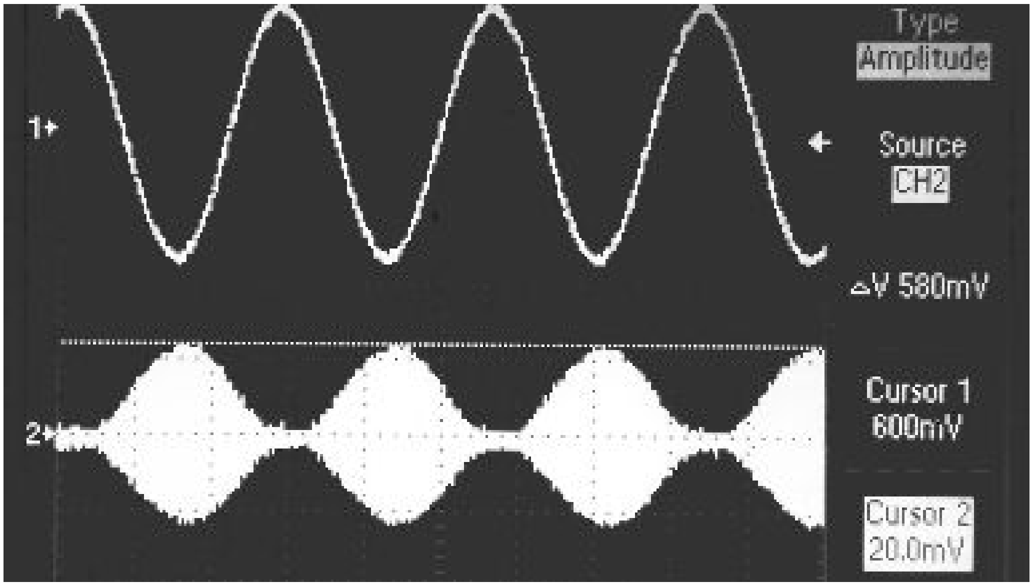
$m = 0.72$

$f_m = 3.6\ kHz$

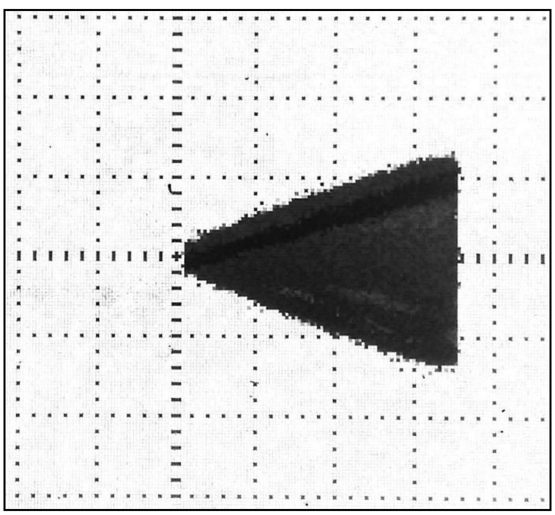
$f_c = 1\ MHz$

Trapezoid Pattern

m = 1



Signal and AM Wave



$$V_{max} = 600\ mV$$

$$V_{min} = 20\ mV$$

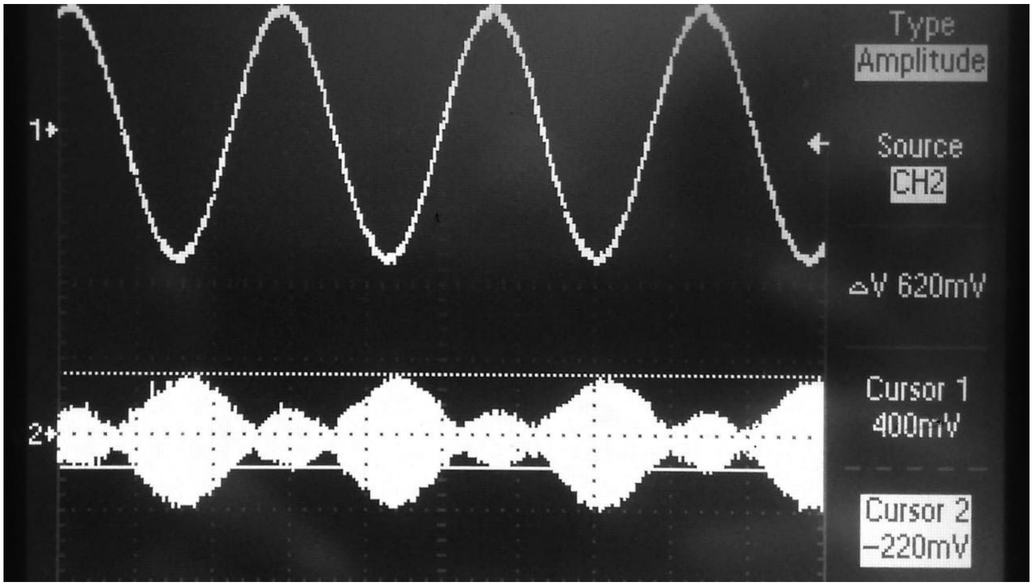
$$m = 0.93$$

$$f_m = 3.6\ kHz$$

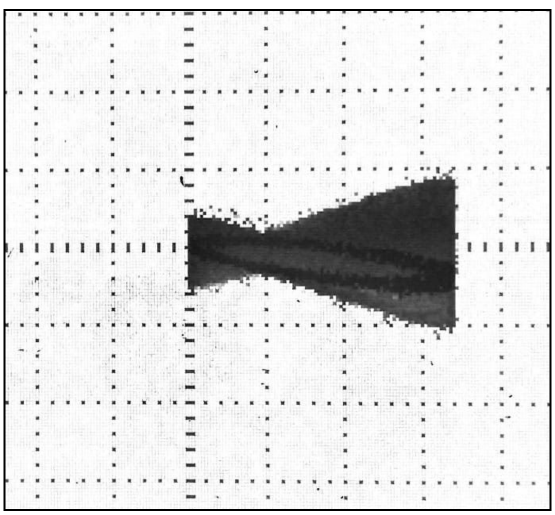
$$f_c = 1\ MHz$$

Trapezoid Pattern

m > 1



Signal and AM Wave



$$V_{max} = 400\ mV$$

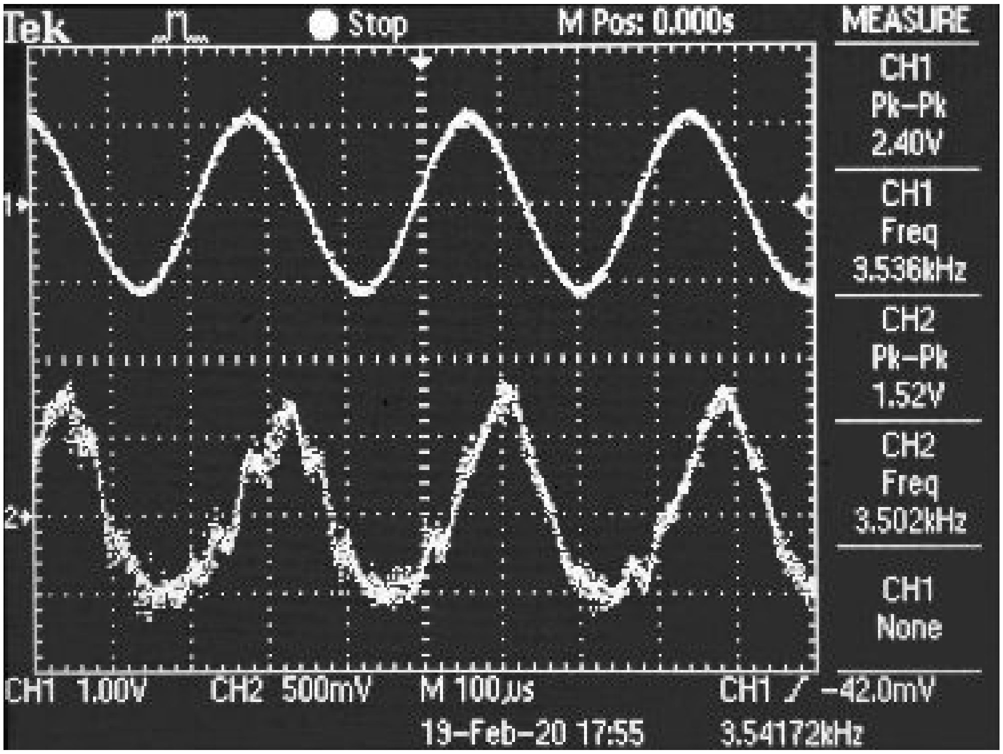
$$V_{min} = -220\ mV$$

$$m = 3.44$$

$$f_m = 3.6\ kHz$$

$$f_c = 1.2\ MHz$$

Trapezoid Pattern



Reconstructed Signal

Conclusion.

In this experiment we studied Amplitude Modulation by transmitting a sine wave as a message signal by modulating it. We also analysed the waveforms produced for three different modulation indices namely ($m < 1$, $m = 1$, $m > 1$). Finally, we demodulated and reconstructed the original message signal.

Remarks.

Signature.

19-02-2020

Experiment 5

PAM, PWM, PPM

Sahil Bondre (u18co021)

Aim.

- To study Pulse Amplitude Modulation and Demodulation (Sample, Sample & Hold and Flat Top)
- To study Pulse Position Modulation and Demodulation
- To study Pulse Width Modulation and Demodulation

Apparatus.

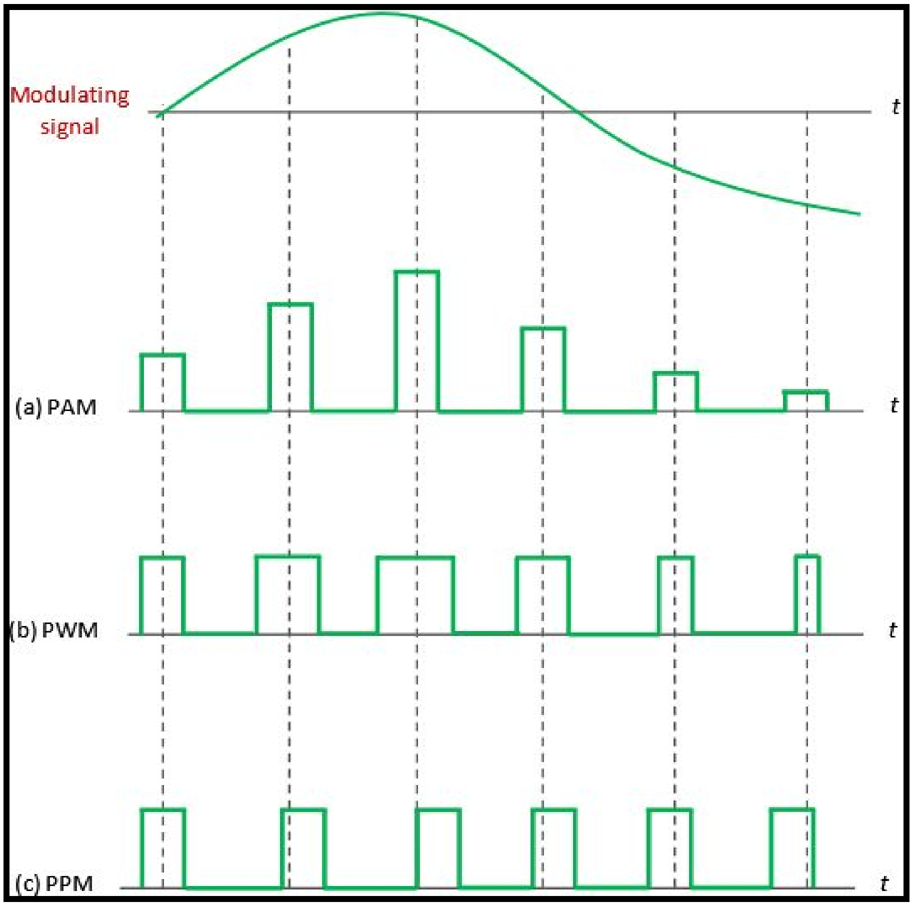
- Trainer board ST 2110
- Power supply
- Connecting Wires
- CRO
- Probes

Theory.

Pulse-amplitude modulation (PAM), is a form of signal modulation where the message information is encoded in the amplitude of a series of signal pulses. It is an analog pulse modulation scheme in which the amplitudes of a train of carrier pulses are varied according to the sample value of the message signal. Demodulation is performed by detecting the amplitude level of the carrier at every single period.

Pulse width modulation (PWM), or pulse-duration modulation (PDM), is a method of reducing the average power delivered by an electrical signal, by effectively chopping it up into discrete parts. The average value of voltage (and current) fed to the load is controlled by turning the switch between supply and load on and off at a fast rate. The longer the switch is on compared to the off periods, the higher the total power supplied to the load.

Pulse-position modulation (PPM) is a form of signal modulation in which M message bits are encoded by transmitting a single pulse in one of 2^M possible required time shifts.This is repeated every T seconds, such that the transmitted bit rate is M/T bits per second. It is primarily useful for optical communications systems, which tend to have little or no multipath interference.



PAM, PWM, PPM

Procedure.

PAM:

1. Connect the circuit as shown in Fig. 1
2. Output of Sine wave to Modulation Signal in PAM block keeping the switch in 1 kHz position
3. 8 kHz pulse output to Pulse IN
4. Switch On power supply.
5. Monitor the outputs at tp. 3, 4 and 5, these are natural, Sample and Hold flat top outputs respectively.

6. Observe the difference between the two outputs and try giving reasons behind them.
7. Try Varying the amplitude and frequency of sine wave by amplitude pot and frequency change over switch. Observe the effect on all the two outputs.
8. Also, try varying the frequency of pulse, by connecting the pulse input to the 4 frequencies available i.e. 8.16.32. 64 kHz in pulse output look.
9. For demodulation part Connect the sample output low pass filter input and Output of low pass filter to input of AC amplifier. Keep the gain pot in AC amplifier block in max position.
10. Follow the steps as of the modulation part.
11. Monitor the output of the AC amplifier. It should be a pure sine wave similar to input.
12. Similarly connect the sample and hold and flat top outputs to Low Pass Filter and see the demodulated waveform at the output of AC amplifier.

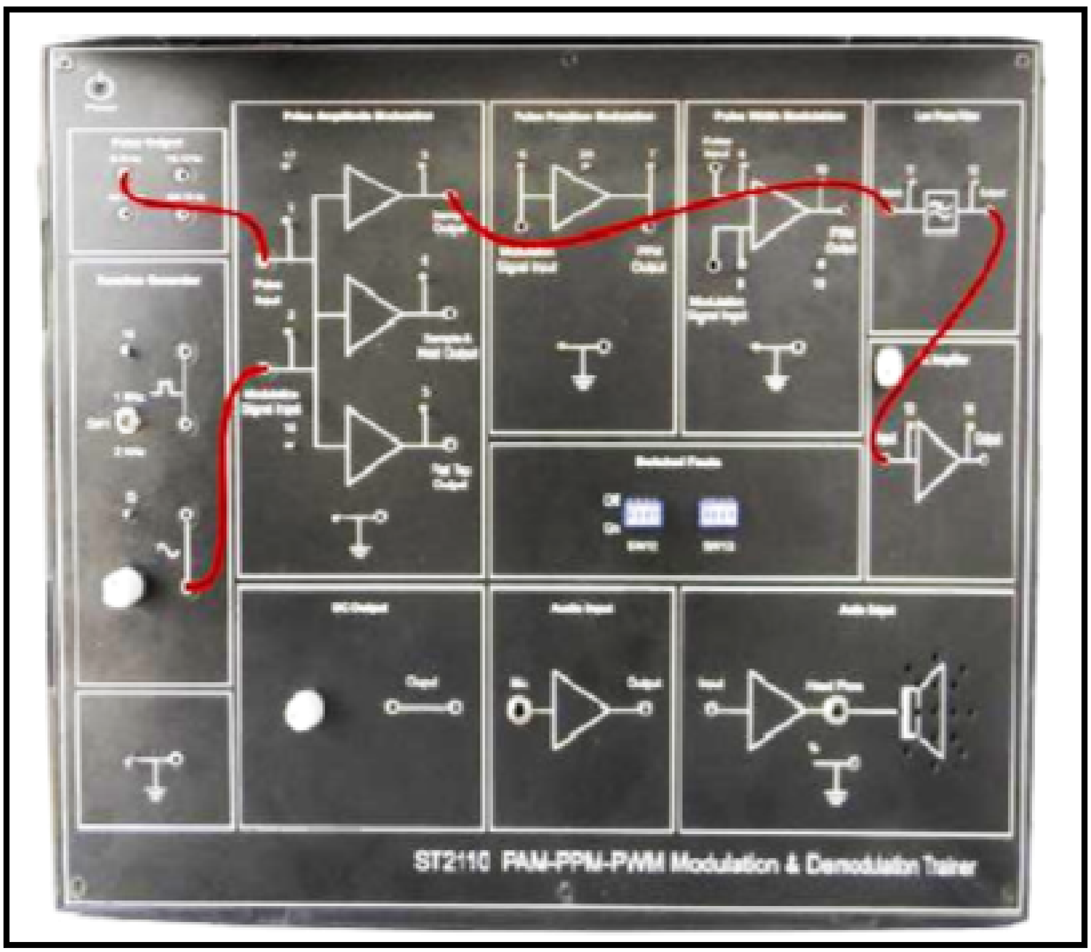


Fig. 1: PAM configuration

PPM:

1. Connect the circuit as shown in Fig. 2
2. Output of Sine wave to Modulation Signal in PPM block keeping the switch in 1 kHz position
3. 8 kHz pulse output to Pulse IN
4. Switch On power supply.
5. Observe the difference between the two outputs and try giving reasons behind them.
6. Try Varying the amplitude and frequency of sine wave by amplitude pot and frequency change over switch. Observe the effect on all the two outputs.
7. Also, try varying the frequency of pulse, by connecting the pulse input to the 4 frequencies available i.e. 8.16.32. 64 kHz in pulse output look.
8. For demodulation part Connect the sample output low pass filter input and Output of low pass filter to input of AC amplifier. Keep the gain pot in AC amplifier block in max position.
9. Follow the steps as of the modulation part.
10. Monitor the output of the AC amplifier. It should be a pure sine wave similar to input.

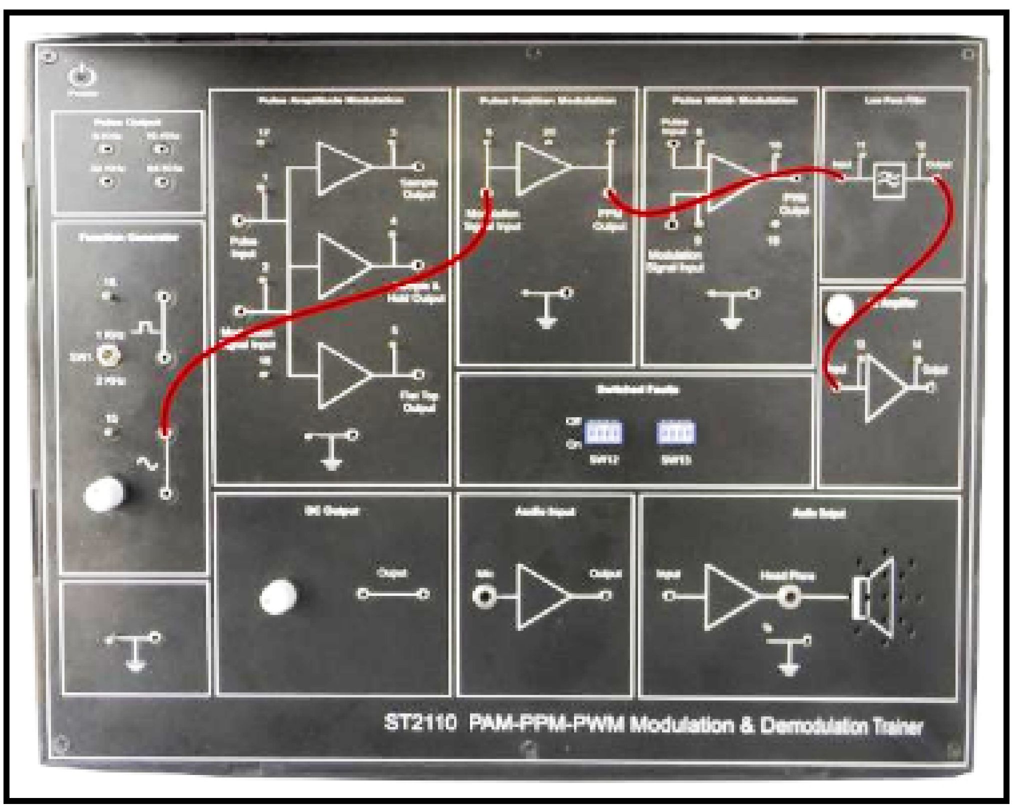


Fig. 2: PPM configuration

PWM:

1. Connect the circuit as shown in Fig. 3
2. Output of Sine wave to Modulation Signal in PWM block keeping the switch in 1 kHz position
3. 8 kHz pulse output to Pulse IN
4. Switch On power supply.
5. Observe the difference between the two outputs and try giving reasons behind them.
6. Try Varying the amplitude and frequency of sine wave by amplitude pot and frequency change over switch. Observe the effect on all the two outputs.
7. Also, try varying the frequency of pulse, by connecting the pulse input to the 4 frequencies available i.e. 8.16.32. 64 kHz in pulse output look.
8. For demodulation part Connect the sample output low pass filter input and Output of low pass filter to input of AC amplifier. Keep the gain pot in AC amplifier block in max position.
9. Follow the steps as of the modulation part.
10. Monitor the output of the AC amplifier. It should be a pure sine wave similar to input.

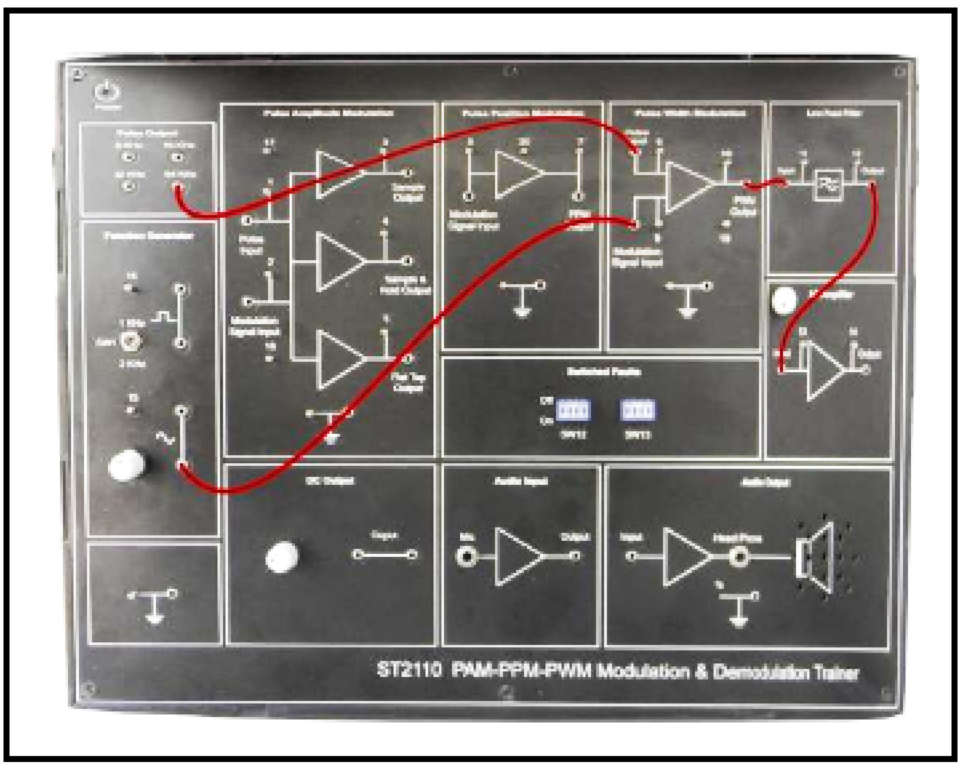
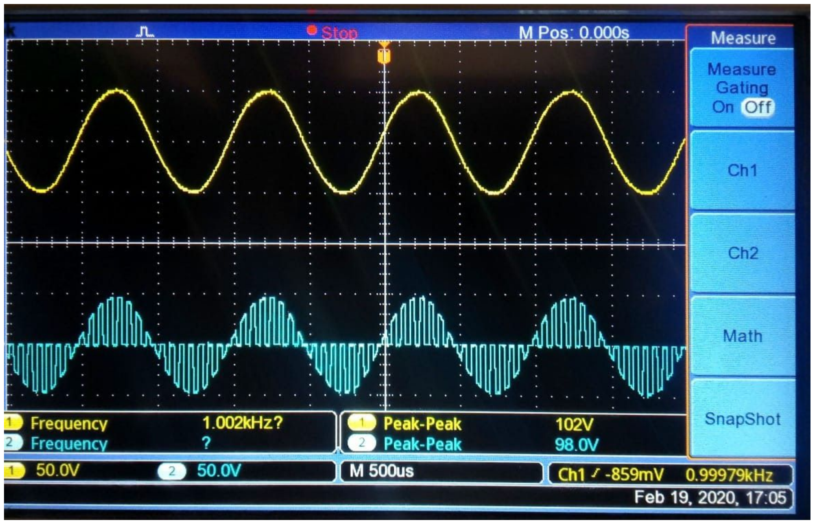


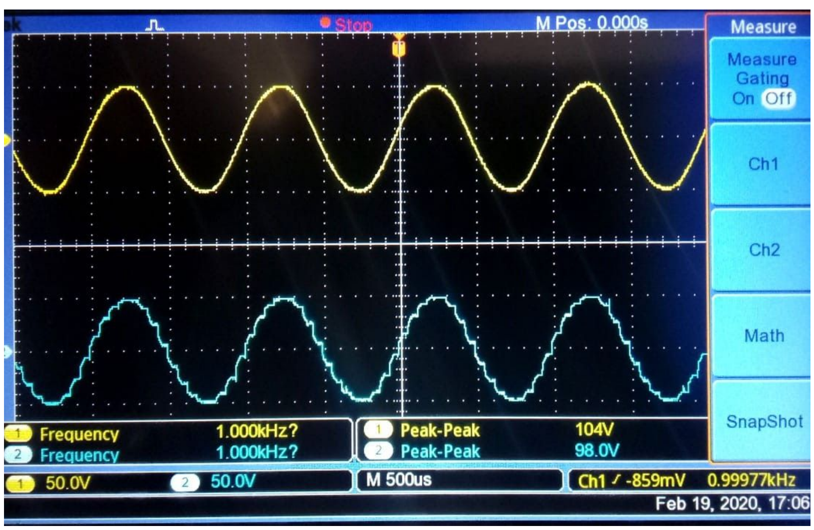
Fig. 3: PWM configuration

Observation.

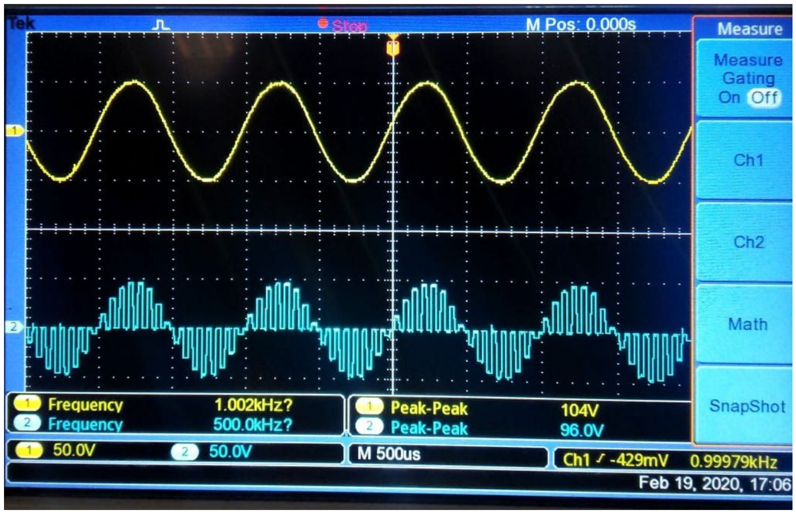
1. PAM



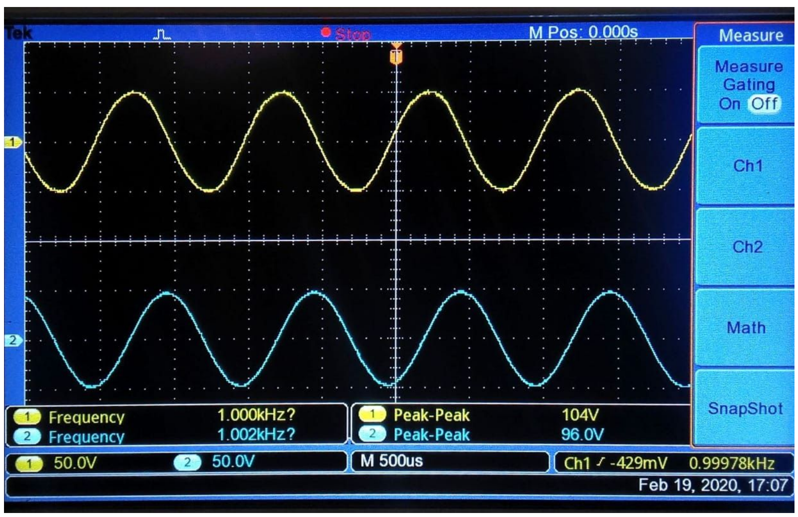
Message Signal and PAM Sample Signal



Message Signal and PAM Sample and Hold Signal

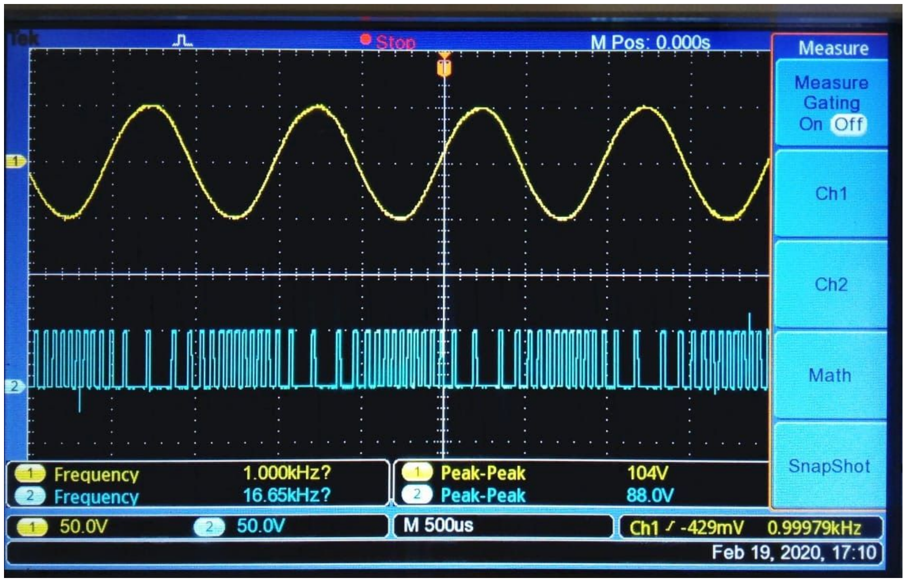


Message Signal and PAM Flat Top Signal

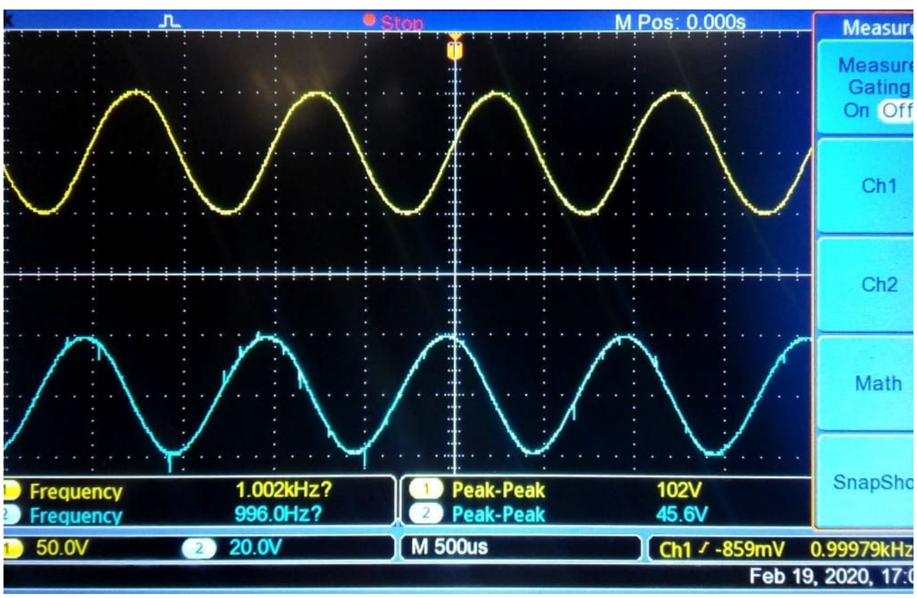


Message Signal with Reconstruction from PAM Flat Top Signal

2. PPM

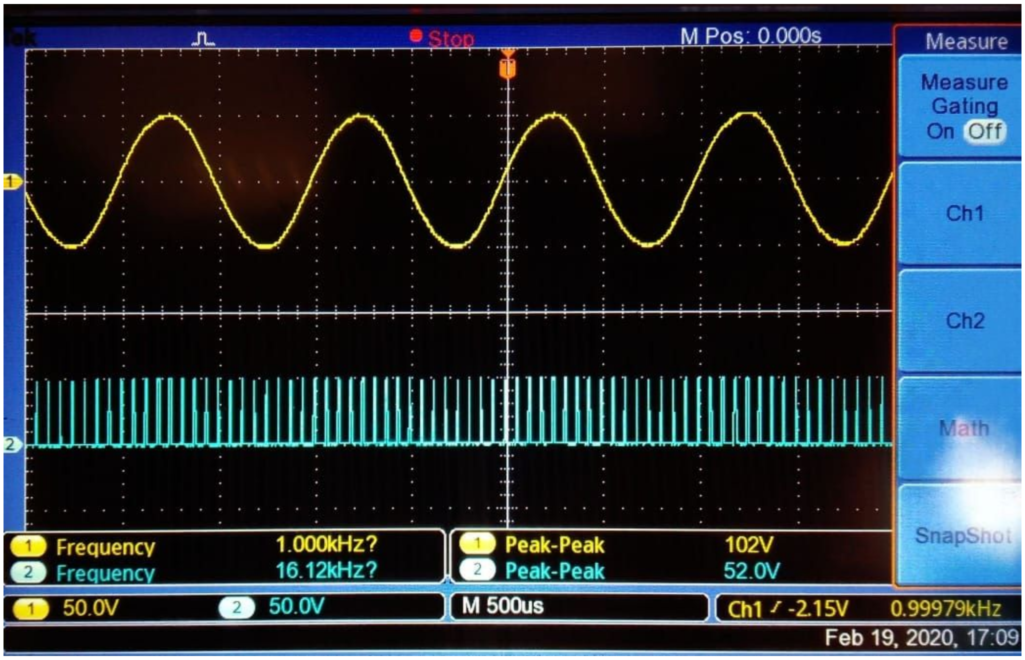


Message Signal and PPM Sample Signal

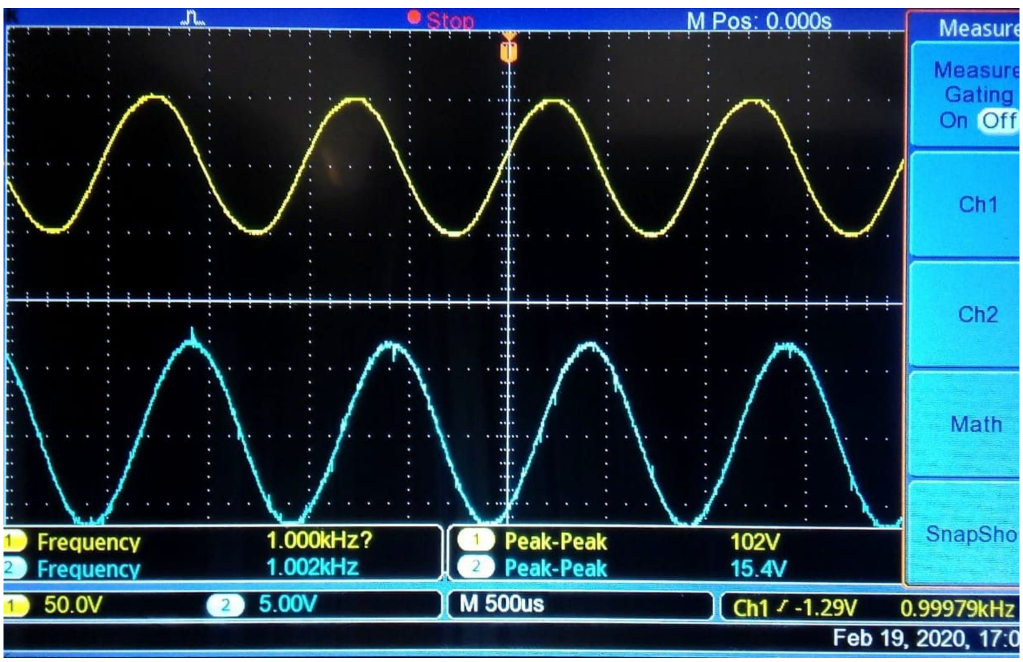


Message Signal with Reconstruction from PPM Signal

3. PWM

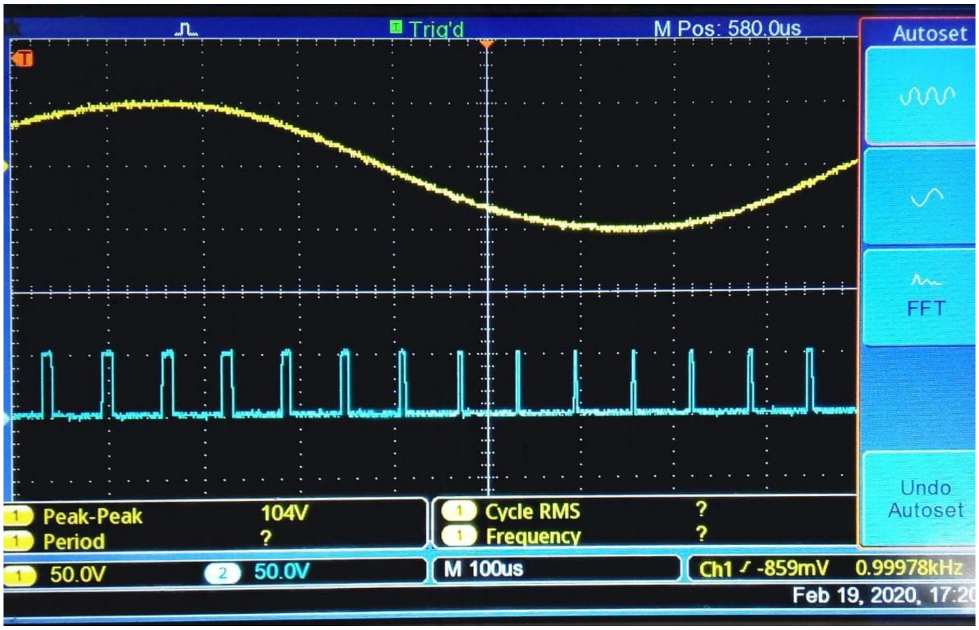


Message Signal and PWM Sample Signal

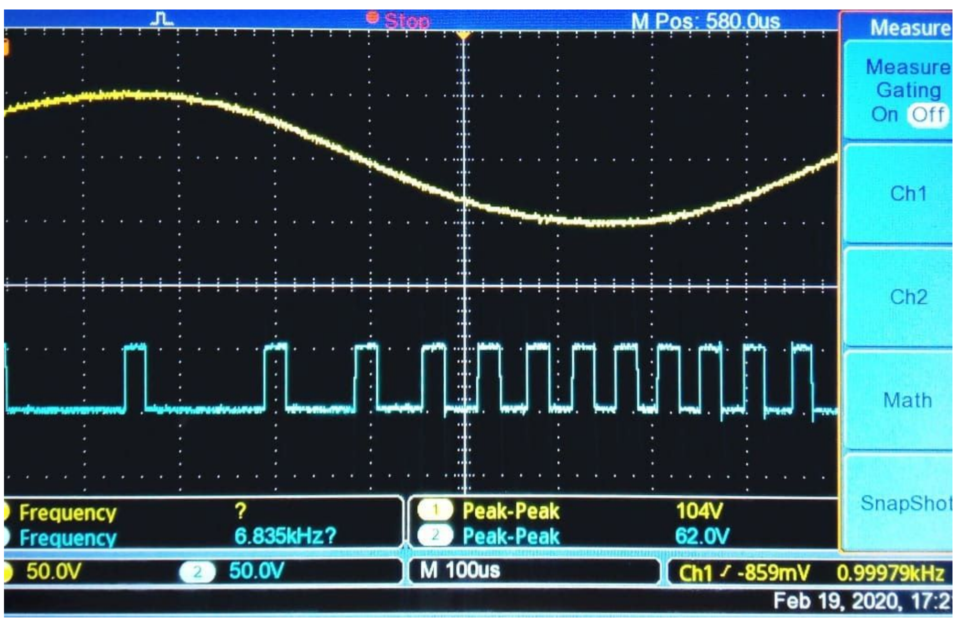


Message Signal with Reconstruction from PWM Signal

PWM vs PPM



Magnified View of PWM



Magnified View of PPM

Conclusion.

In this experiment we studied Pulse Amplitude Modulation, Pulse Width Modulation and Pulse Position Modulation by transmitting a sine wave as a message signal by modulating it. We also demodulated and reconstructed the original message signal and analysed the effectiveness and power efficiency of each modulation method.

Remarks.

Signature.

<u>FREQUENCY MODULATION (FM)</u>

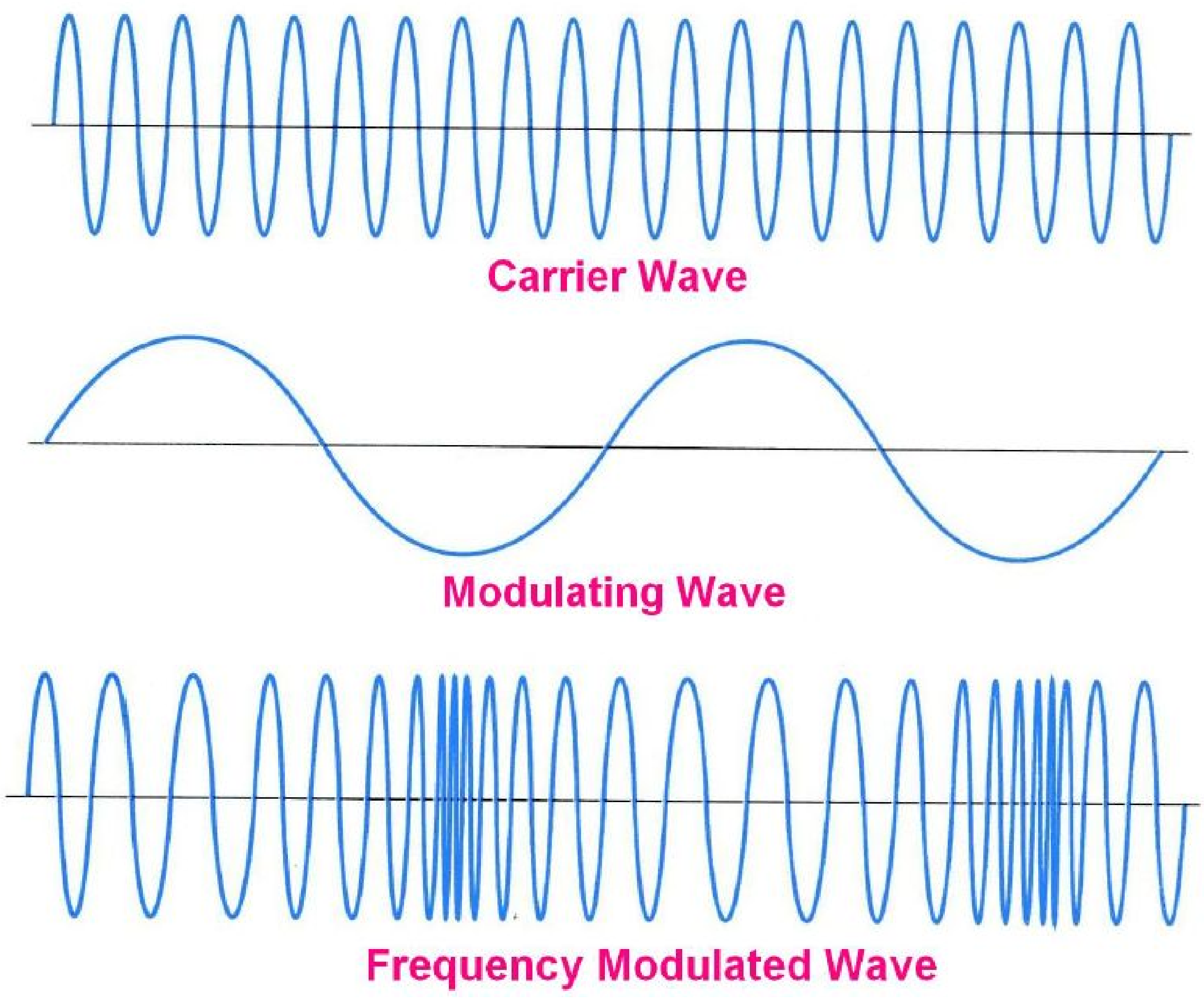
Experiment No: 6 **Date:**

Aim: To study frequency modulation and demodulation and observe the waveforms.

a) **Observe the spectra of FM signal in labAlive virtual communication lab and Calculate the modulation index for FM**
b) **To perform FM transmission via virtual lab labAlive for the audio signal**
c) **To perform FM reception via virtual lab labAlive for the obtained recorded signal**

Frequency Modulation (FM):

The frequency of the carrier waveform varies with the information signal



Frequency modulation is a system in which the amplitude of the modulated carrier is kept constant, while its frequency is varied by the modulating signal, the modulating signal is sinusoidal. This signal has two important parameters which must be represented by the modulation process without distortion: namely its amplitude and frequency.

If carrier signal, $e_c = E_c \sin sin\ \omega_c t$ and modulating signal, $e_m = E_m \sin sin\ \omega_m t$ then, the peak or maximum frequency deviation:
$\Delta f\ \alpha\ \ e_m$
$\Delta f = k_f e_m$

Where, k_f is proportionality constant[V/Hz], and e_m is the instantaneous value of the modulating signal amplitude. Thus the frequency of the FM signal is:

$$e_s(t) = e_c + \Delta f = e_c + k_f e_m(t)$$

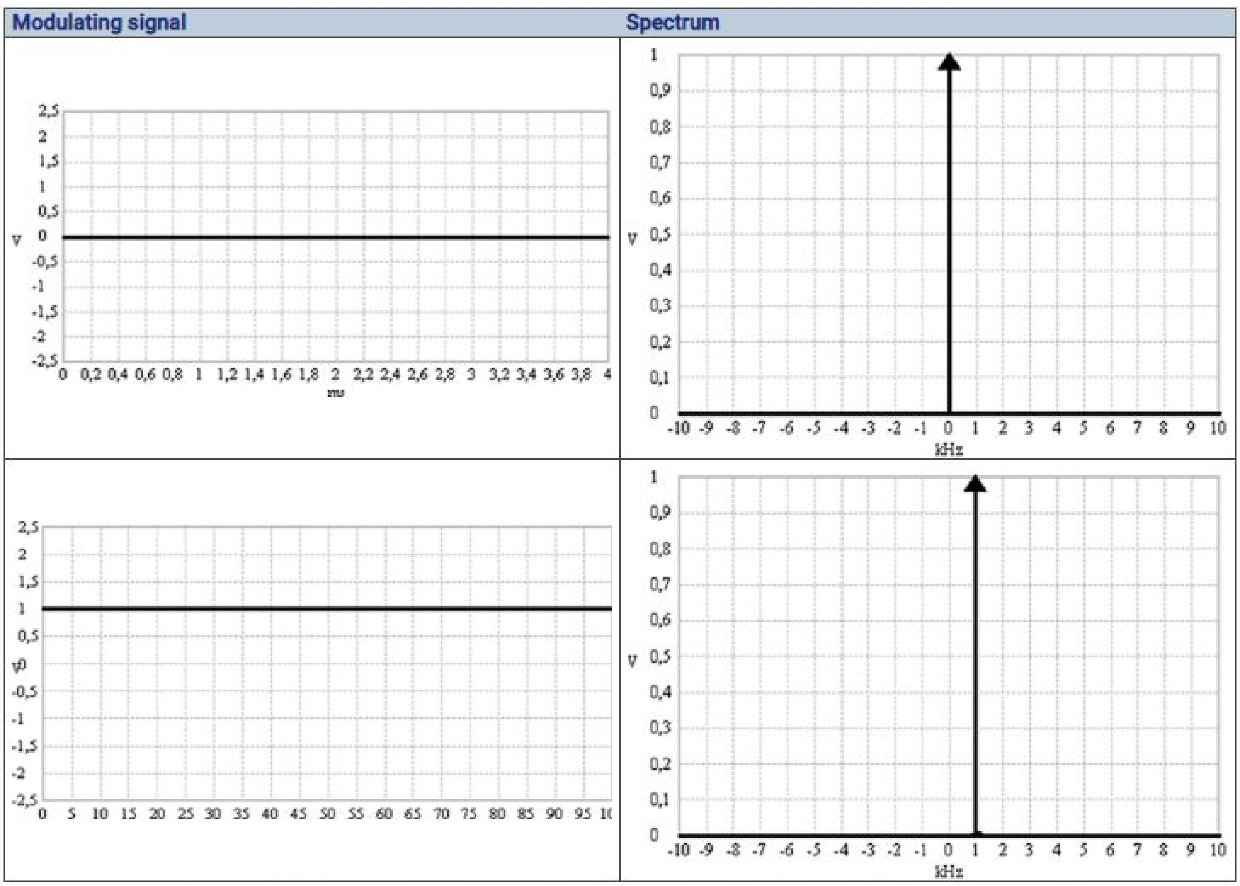
$$\text{Then,}\quad e_s(t) = e_c + k_f E_m \sin sin\ \omega_m t$$

Then the equation for the FM signal is:

$$e_s(t) = E_c \sin sin\ (\omega_c t + \beta\ \ \sin sin\ \omega_m t\ )$$

Where, β = modulation index, which can be greater than 1. It is measured in radians
β = Freq. Deviation / Modulating Freq.

$$\beta = \frac{\Delta f}{f_m}$$

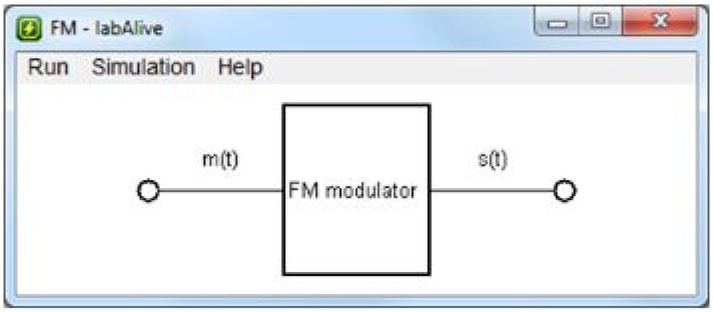


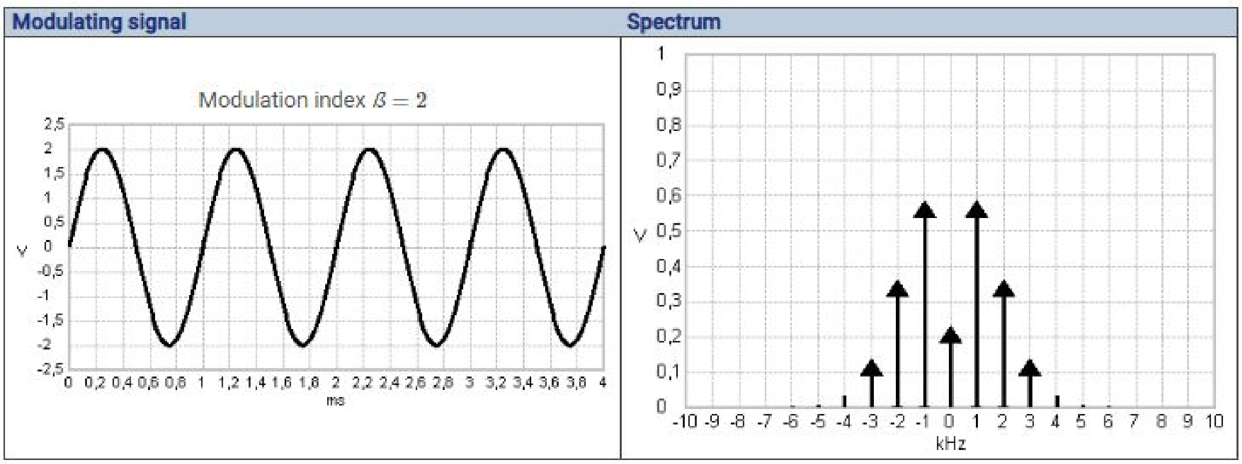
Frequency modulation example - frequency deviation is 1 kHz for a 1V-DC modulating signal

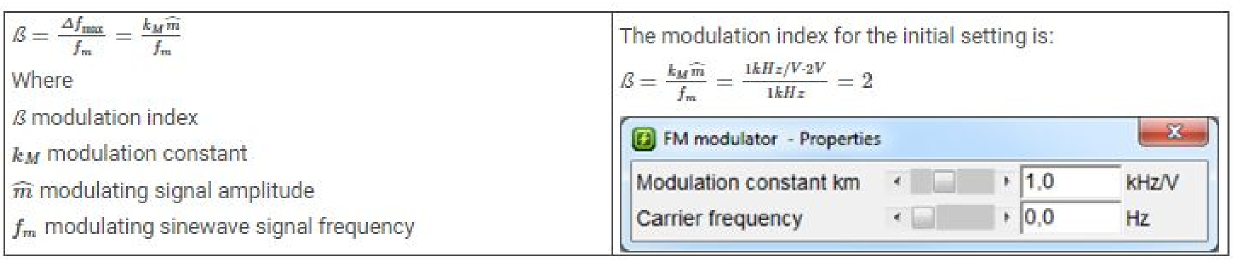
Part a): In this experiment a sinewave signal is frequency modulated. Modulating signal and modulator parameters determine the spectrum of the resulting FM transmission signal.

Procedure:

- On launching the experiment, you will see the following windows:



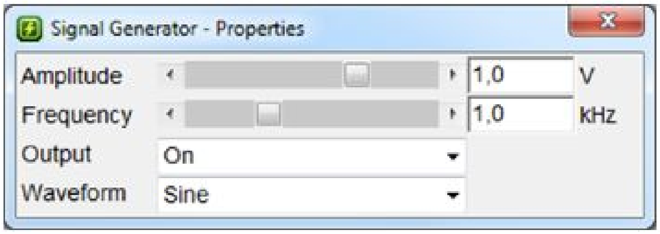




The modulation index ß is the ratio of the maximum frequency deviation of the carrier to the frequency of the sinewave modulating signal.

The Bessel function values at the resulting modulation index determine the spectrum of the FM signal.

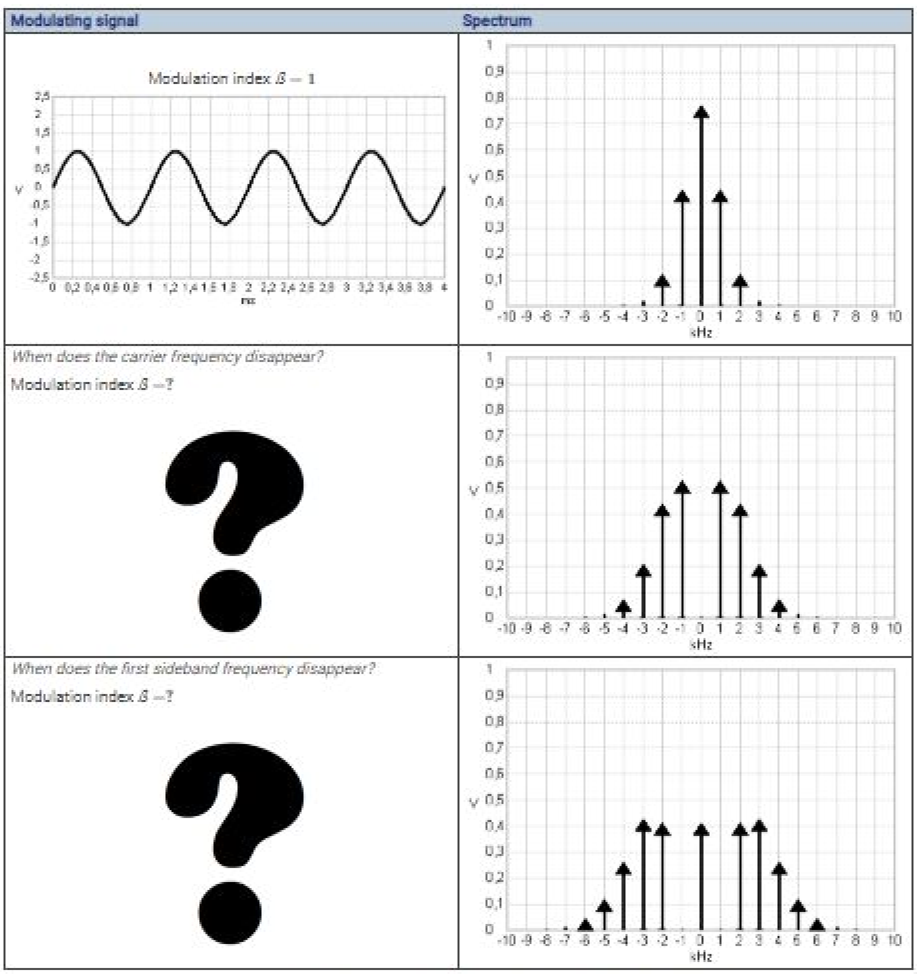
- Vary the modulating signal amplitude $\hat{m}$.



- The modulation index is proportional to the modulating signal amplitude. In this setting the amplitude in Volts is the modulation index:

$$\beta = \frac{\Delta f}{f_m} = \frac{k_f \hat{m}}{f_m} = \frac{\frac{1kHz}{V} \times \hat{m}}{1kHz} = \frac{\hat{m}}{V}$$

- The adjusted modulating signal amplitude determines the spectral amplitudes of the carrier and sideband frequencies. For some values the carrier or specific sideband frequencies disappear. This relates to zero crossings of the respective Bessel function at the corresponding modulation index.



FM signal spectra for sinewave modulation with different modulation indices.

The carrier frequency is 0 Hz in this setting. It might be changed via the modulator properties.

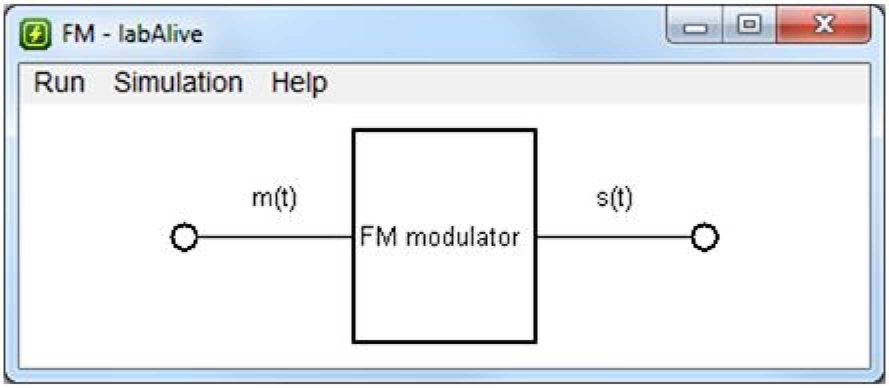
NEXT STEPS

- When do the 2nd and 3rd sideband frequencies disappear?

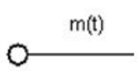
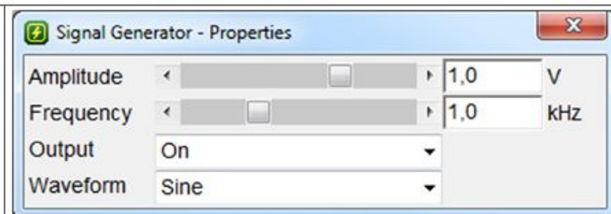
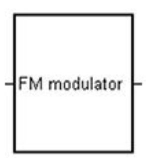
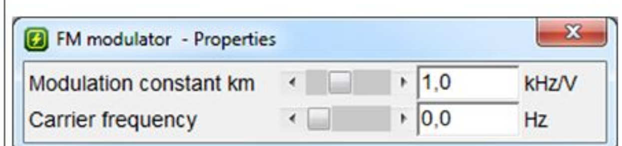
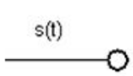
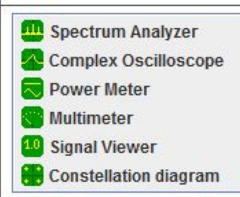
  ____

- Vary the modulating sinewave signal frequency.
- Select different waveforms (signal generator properties) and regard the FM spectrum.
- Use the Bessel functions to determine the spectrum of an FM signal with $\beta = 3$

This simulation implements frequency modulation. The FM signal is generated for the chosen modulating signal. Its spectrum is shown in a spectrum analyzer. All parameters of the modulating signal and modulator can be adjusted.



To change the different settings click on the corresponding wiring:

Adjust parameters of input signal 	
Adjust parameters of FM modulator Left click on FM modulator: 	
Open measure for transmission signal Right click on s(t): 	

Adjust parameters of FM

- Click on the launch tab on the link https://www.etti.unibw.de/labalive/experiment/fm/

- Change the amplitude and frequency of the modulating signal, observe the spectra (attach the output waveforms you observe) and complete the following observation table.
- Note down the frequency deviation from the spectra of FM signal as suggested in the figures below

Reference: Communication Systems by Simon Haykin, 4th edition. Refer Example 2.2 on page 116-117.

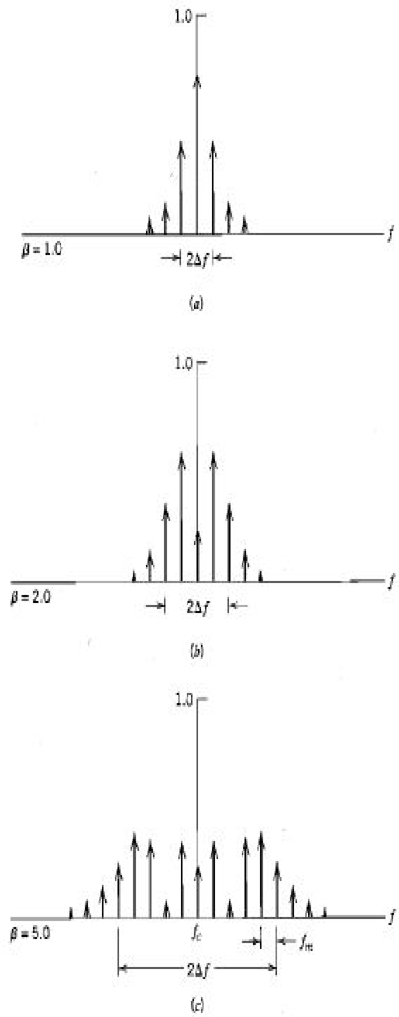


FIGURE 2.24 Discrete amplitude spectra of an FM signal, normalized with respect to the carrier amplitude, for the case of sinusoidal modulation of fixed frequency and varying amplitude. Only the spectra for positive frequencies are shown.

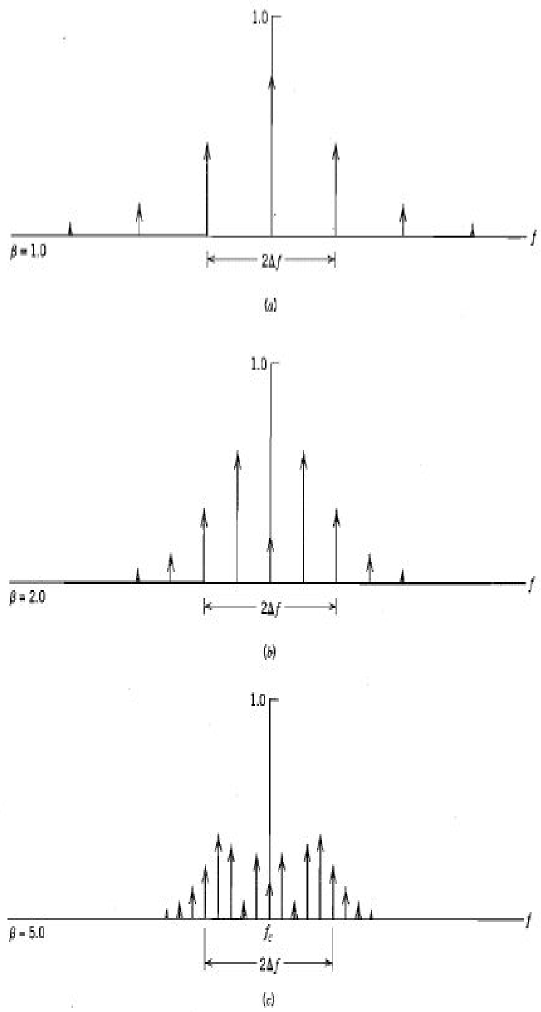


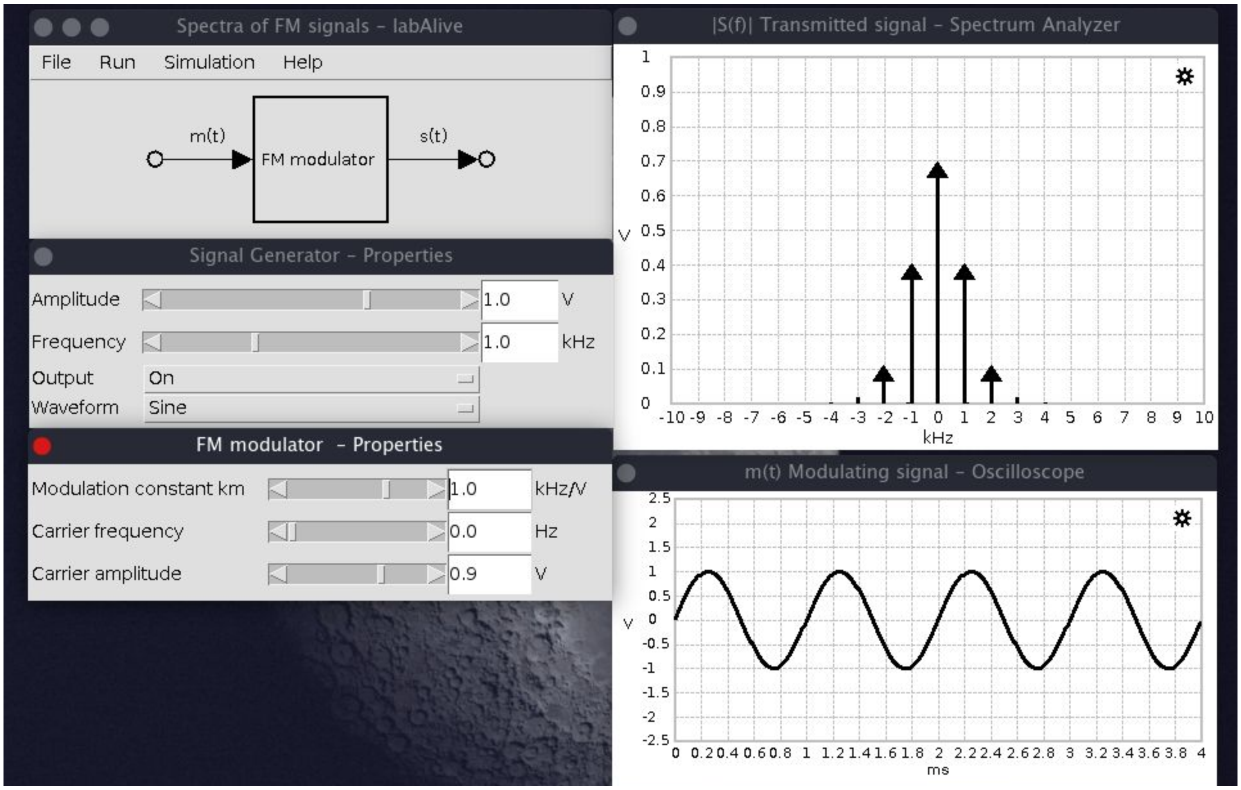
FIGURE 2.25 Discrete amplitude spectra of an FM signal, normalized with respect to the carrie amplitude, for the case of sinusoidal modulation of varying frequency and fixed amplitude. Only the spectra for positive frequencies are shown.

Sr. no	**Modulating Signal frequency, f_m**	**Modulating signal amplitude, $\hat{m}$**	**Frequency Deviation, Δf**	**Modulation Index, $\beta = \frac{\Delta f}{f_m}$**
1	1kHz	1.0V	1.0	1
		2.0V	2.0	2
2	2kHz	2.0V	3.0	1.5
		3.0V	4.5	2.25
3	3kHz	3.0V	9	3
		2.0	6	2

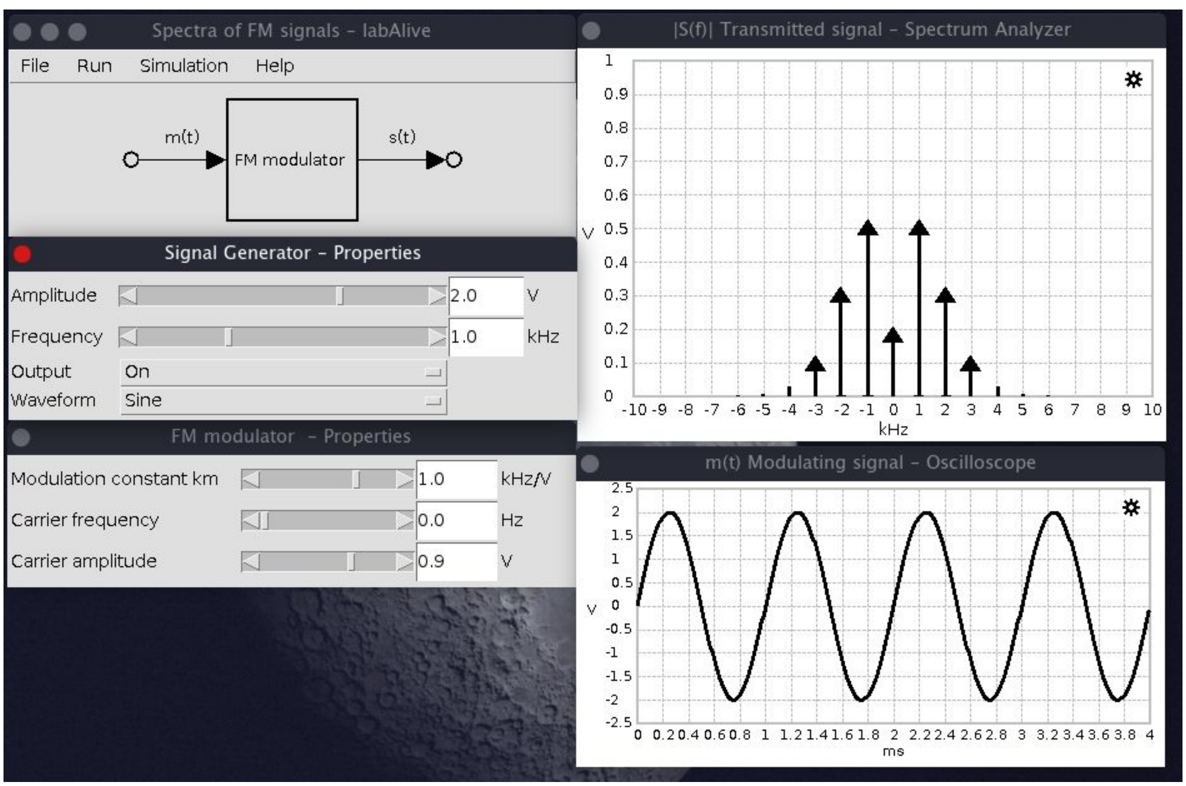
Follow this link for detailed procedure and setup

https://youtu.be/THzJ6bjf1HI, https://www.etti.unibw.de/labalive/experiment/fm/

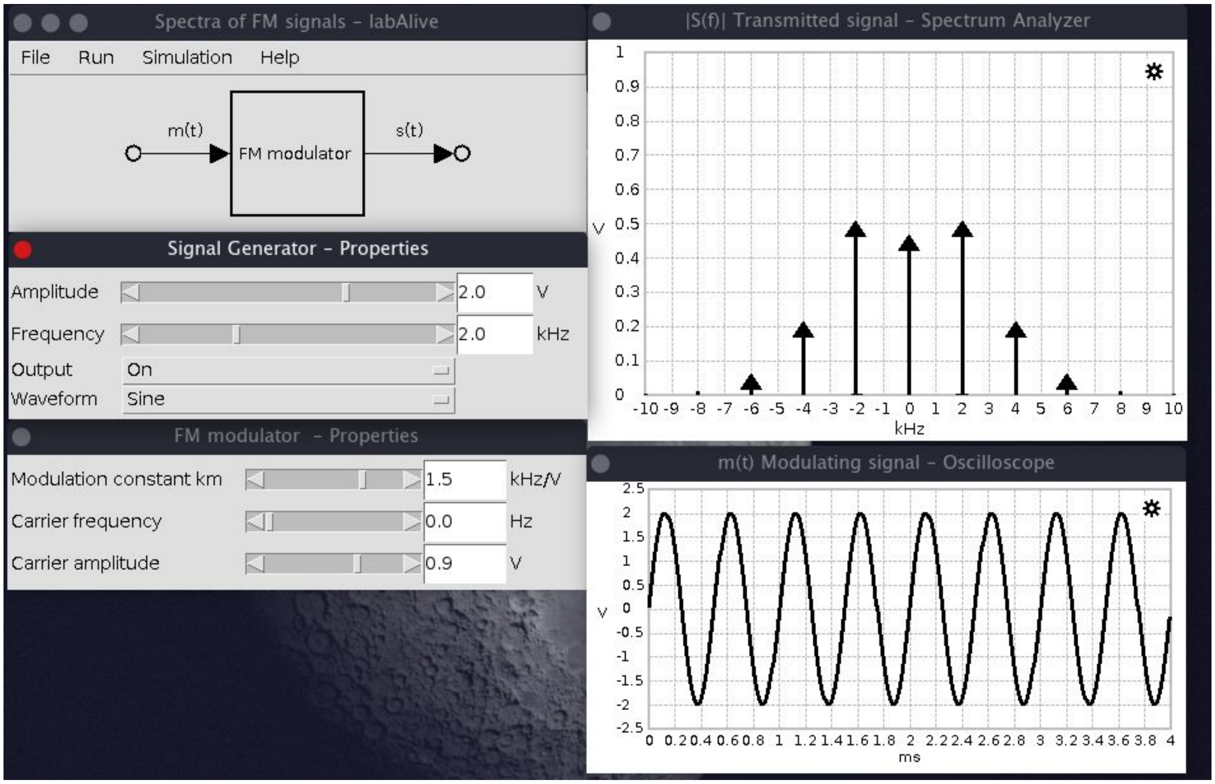
Output Waveforms:



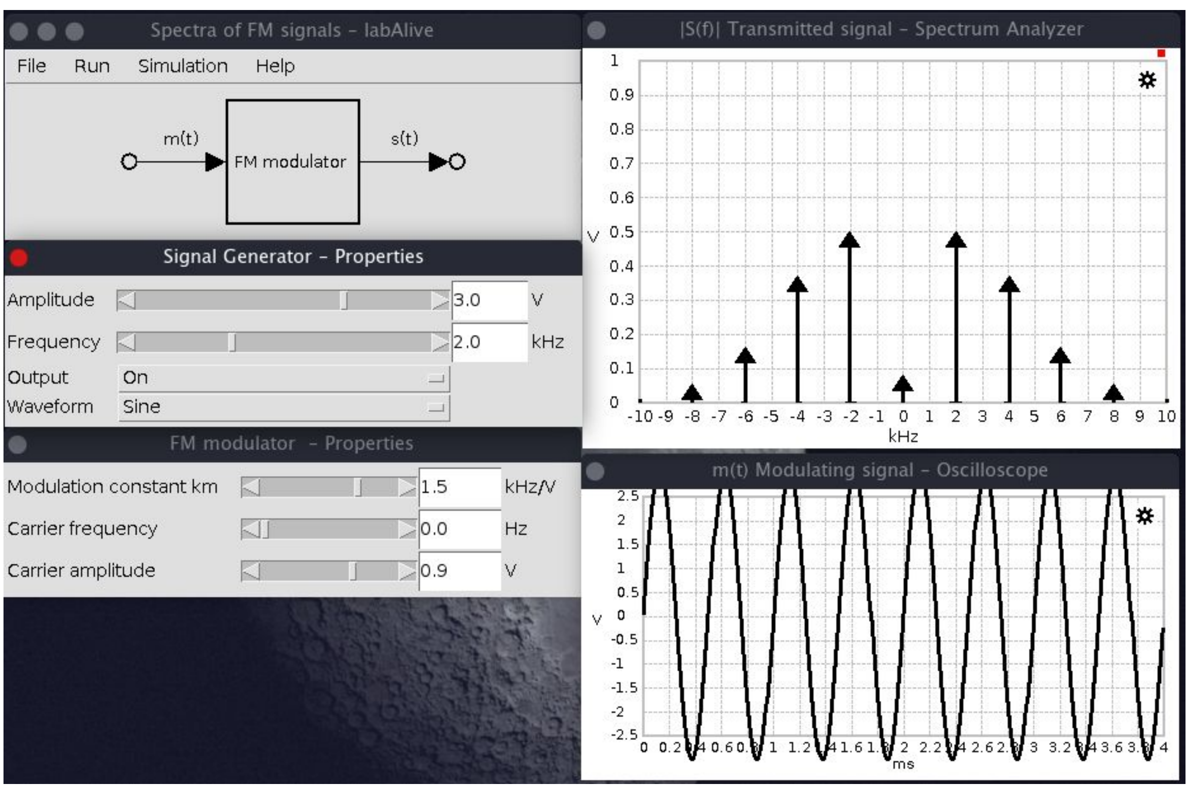
fm = 1kHz m=1.0V Beta=1



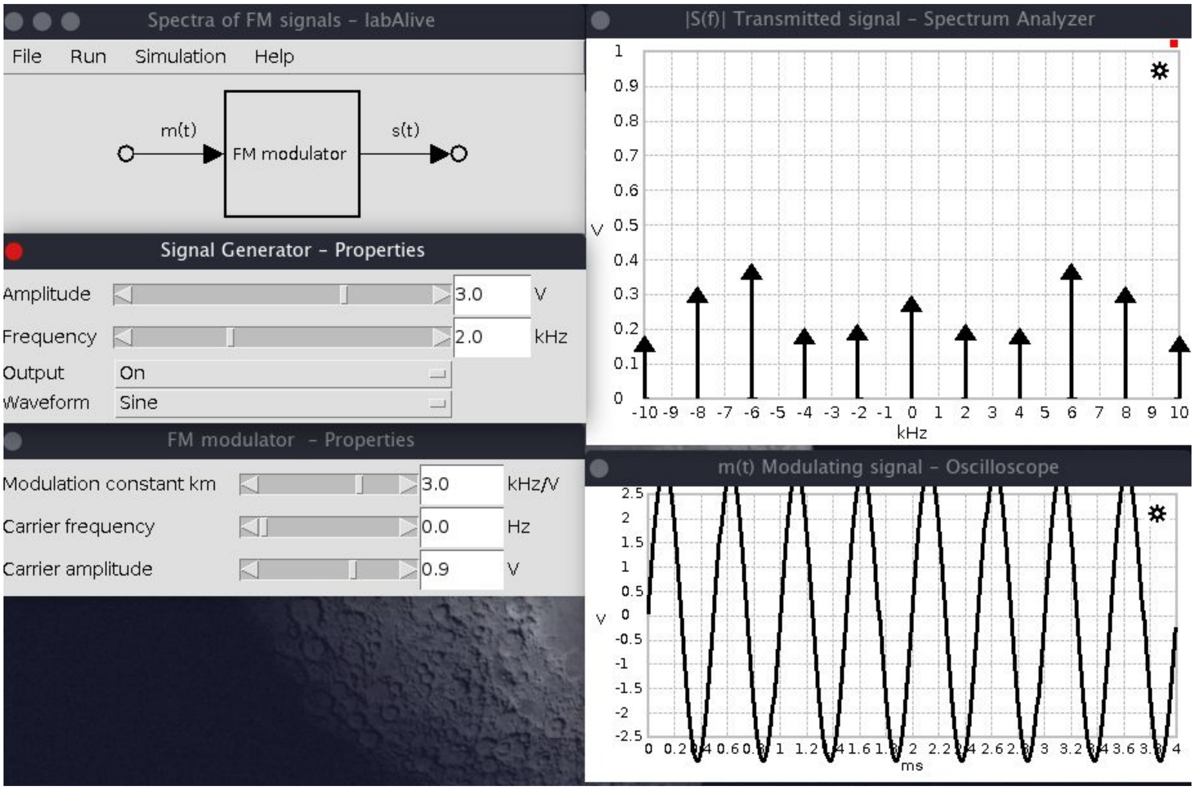
fm = 1kHz m=2.0V Beta=2



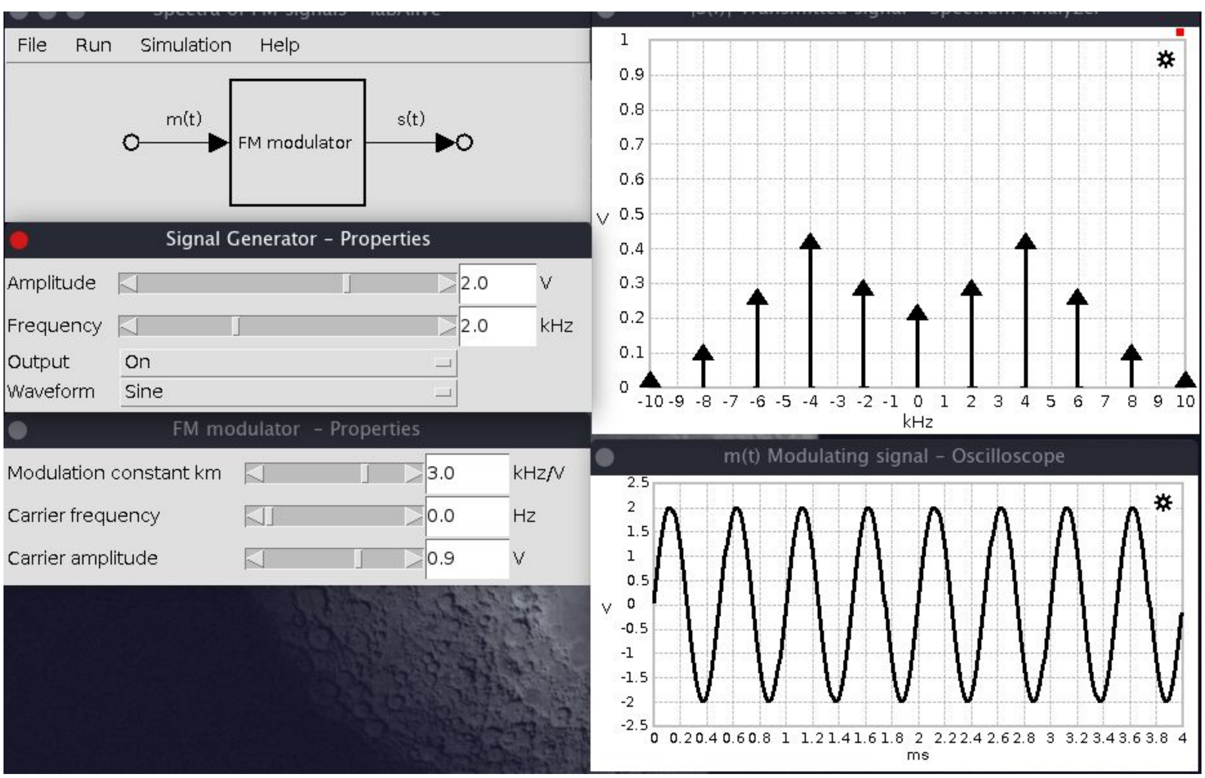
fm = 2kHz m=2.0V Beta=1.5



fm = 2kHz m=3.0V Beta=2.25



fm = 3kHz m=3.0V Beta=3

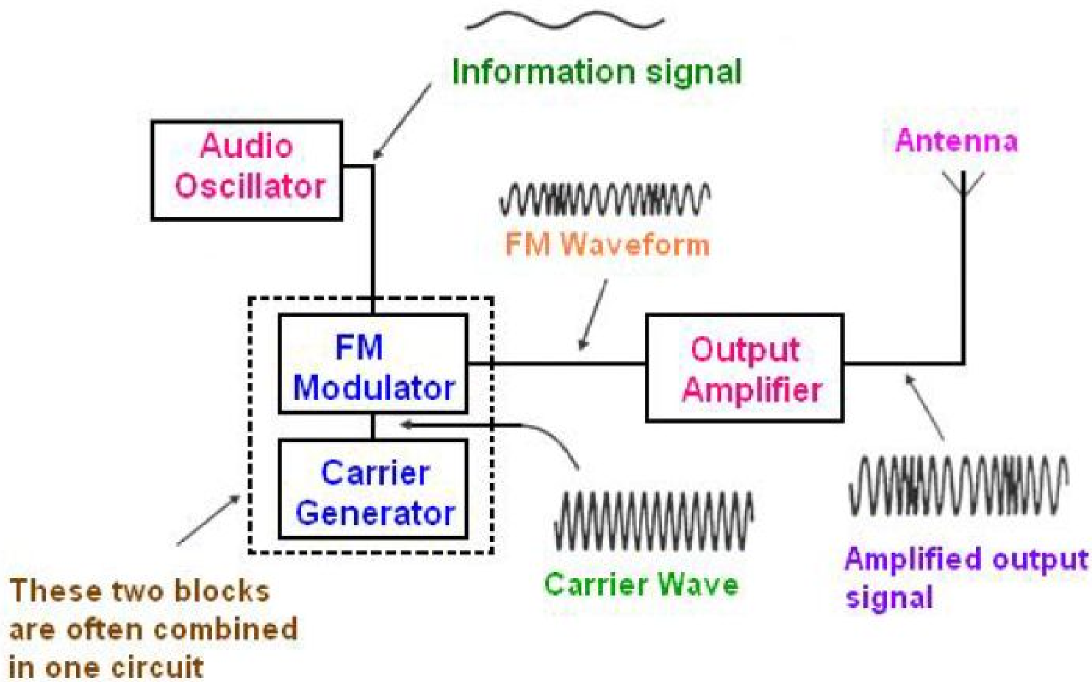


fm = 3kHz m=2.0V Beta=2

Part b) To send an audio file via FM transmission link

FM Transmitter:

The block diagram is shown in figure.



Procedure:

START

Initially a music signal provided by the server is frequency modulated. You might select your own audio file in the format 44.1kHz, 16 bit, stereo or the microphone *Line in*.

SET FREQUENCY DEVIATION

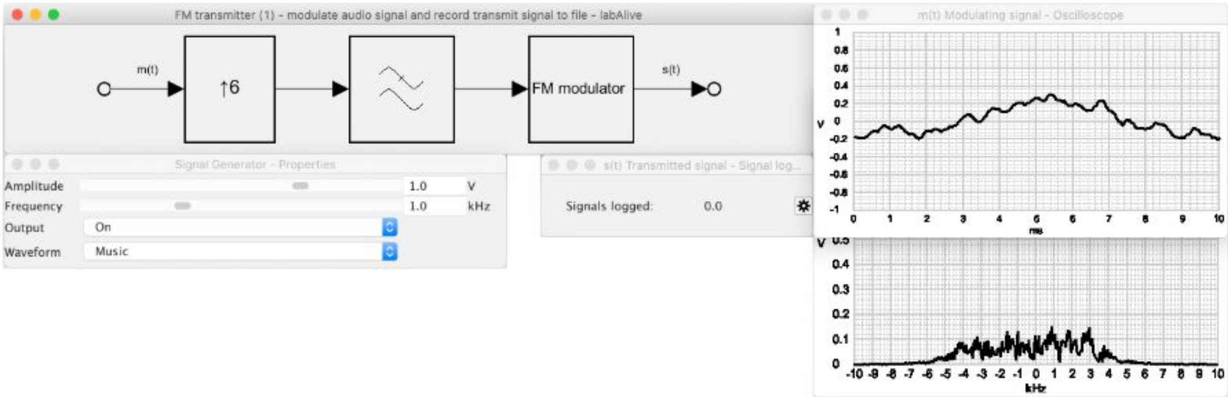
Increase the signal generator's amplitude so that the frequency deviation, i.e. the maximum shift away from the carrier frequency, is 75 kHz. In this base band simulation the carrier frequency is set to 0 Hz.

RECORD THE FM MODULATED SIGNAL TO A FILE

Open the settings of the signal logger - click the gear wheel settings icon. Click *Start save samples to file* to record the transmitted signal to a file.

DEMODULATE THE FM MODULATED SIGNAL

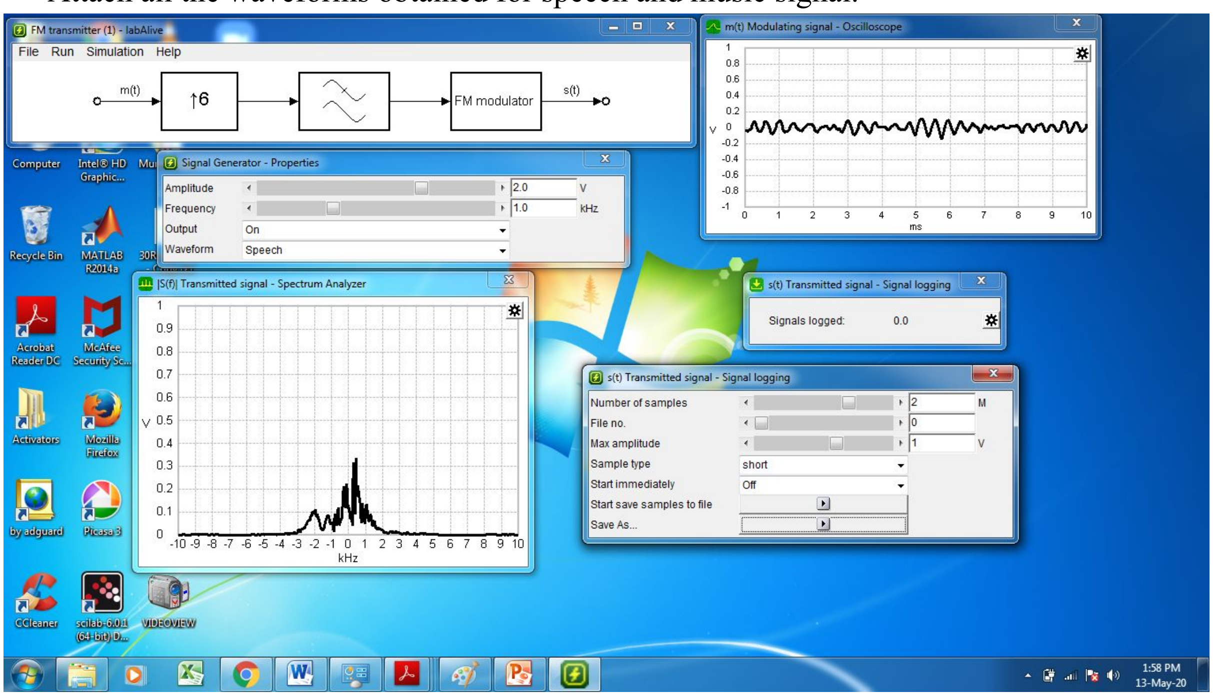
Go the the FM receiver and demodulate the FM transmitted signal.



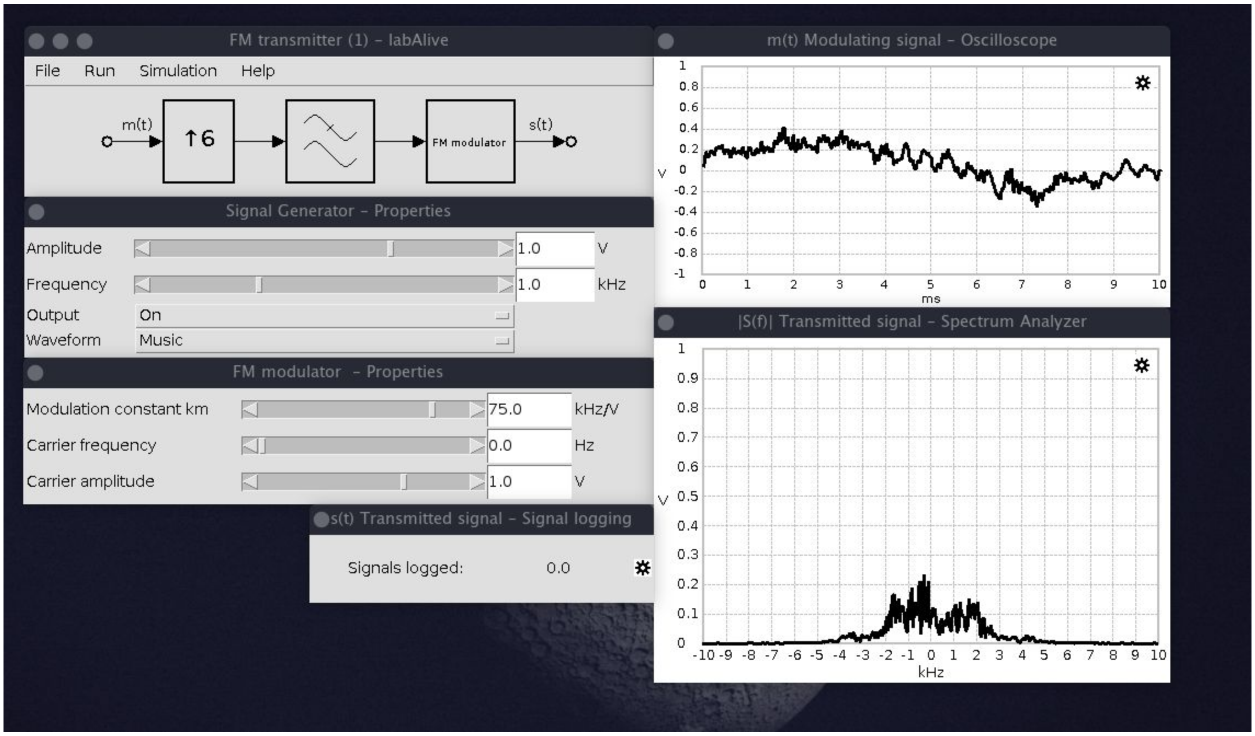
FM transmitter - modulate an audio signal and record the transmit signal to file.

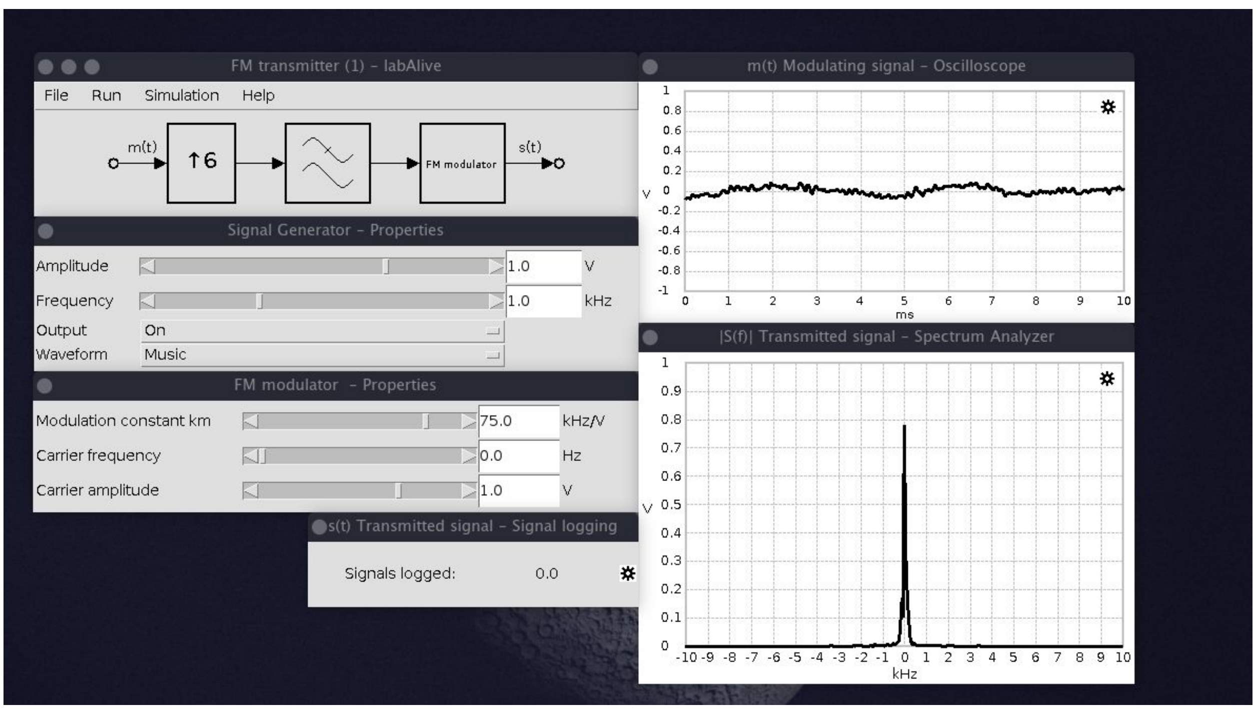
Notes:

- You may modify the input signal from the signal generator properties
- Varying the frequency and amplitude of input signal will change frequency deviation.
- To save the samples, click on the settings icon in signal logging window and save it. You will use this same file for reception.
- You can scale the graph by clicking on setting icon in modulating signal window and frequency spectrum analyser window.
- Attach all the waveforms obtained for speech and music signal.



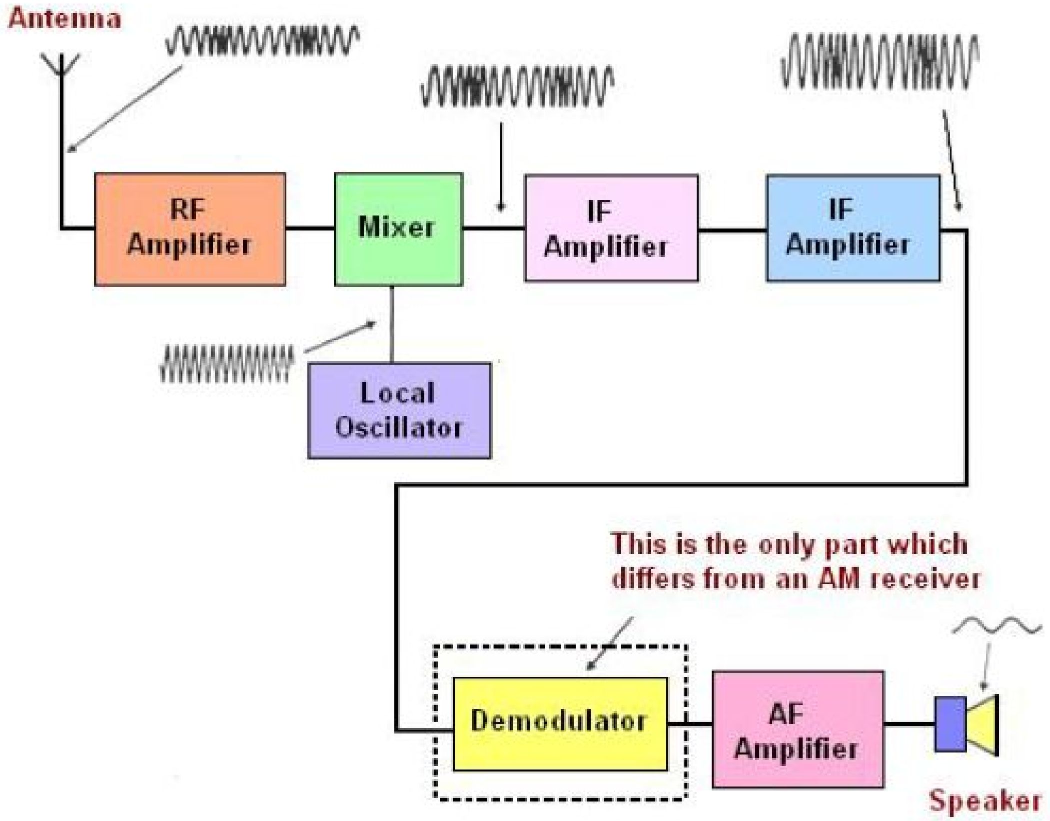
Output Waveforms:


FM transmitter (1) - labAlive
File
Run
Simulation
Help
m(t)
↑6
FM modulator
s(t)
Signal Generator - Properties
Amplitude
1.0
V
Frequency
1.0
kHz
Output
On
Waveform
Music
FM modulator - Properties
Modulation constant km
75.0
kHz/V
Carrier frequency
0.0
Hz
Carrier amplitude
1.0
V
s(t) Transmitted signal - Signal logging
Signals logged:
0.0
m(t) Modulating signal - Oscilloscope
V
ms
|S(f)| Transmitted signal - Spectrum Analyzer
V
kHz


FM transmitter (1) - labAlive
File
Run
Simulation
Help
m(t)
↑6
FM modulator
s(t)
Signal Generator - Properties
Amplitude
1.0
V
Frequency
1.0
kHz
Output
On
Waveform
Music
FM modulator - Properties
Modulation constant km
75.0
kHz/V
Carrier frequency
0.0
Hz
Carrier amplitude
1.0
V
s(t) Transmitted signal - Signal logging
Signals logged:
0.0
m(t) Modulating signal - Oscilloscope
V
ms
|S(f)| Transmitted signal - Spectrum Analyzer
V
kHz

Part C): To receive an audio file via FM receiver.

FM receiver:



FM Receiver

Procedure:

START

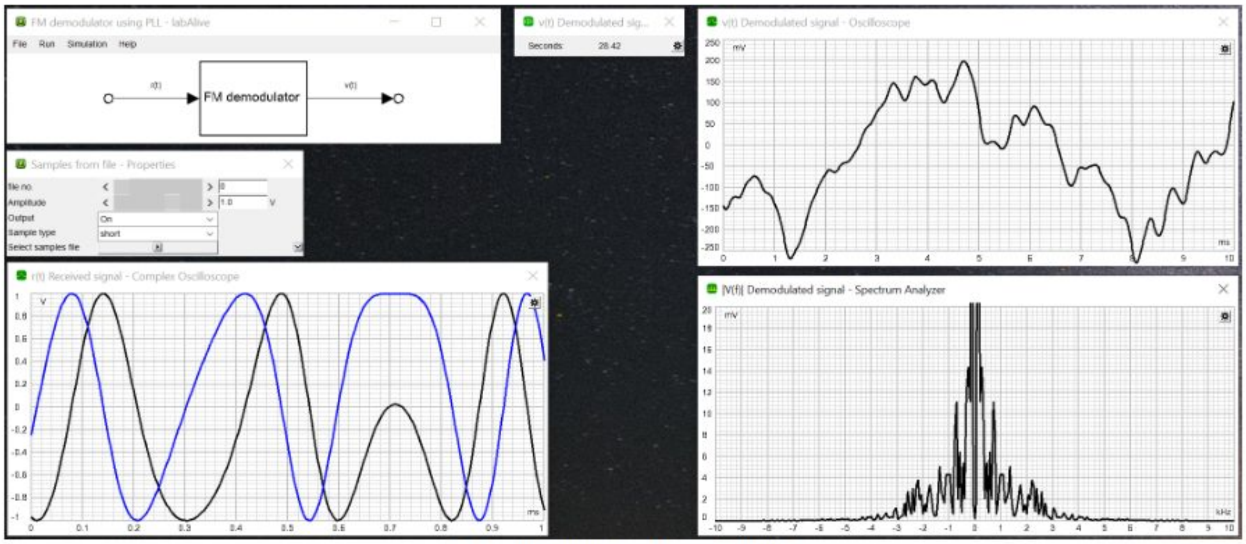
Select the file of the FM modulated signal you created with the FM transmitter.

LISTEN TO THE DEMODULATED SIGNAL

Enjoy the demodulated audio signal. It should be fine if you modulated the audio signal properly. If it's too quiet or distorted analyze if the frequency deviation is too large or too small.

VARY FREQUENCY DEVIATION AND CREATE DIFFERENT FM MODULATED SIGNALS

Vary the modulating signal's amplitude and thus also the frequency deviation using the FM transmitter.

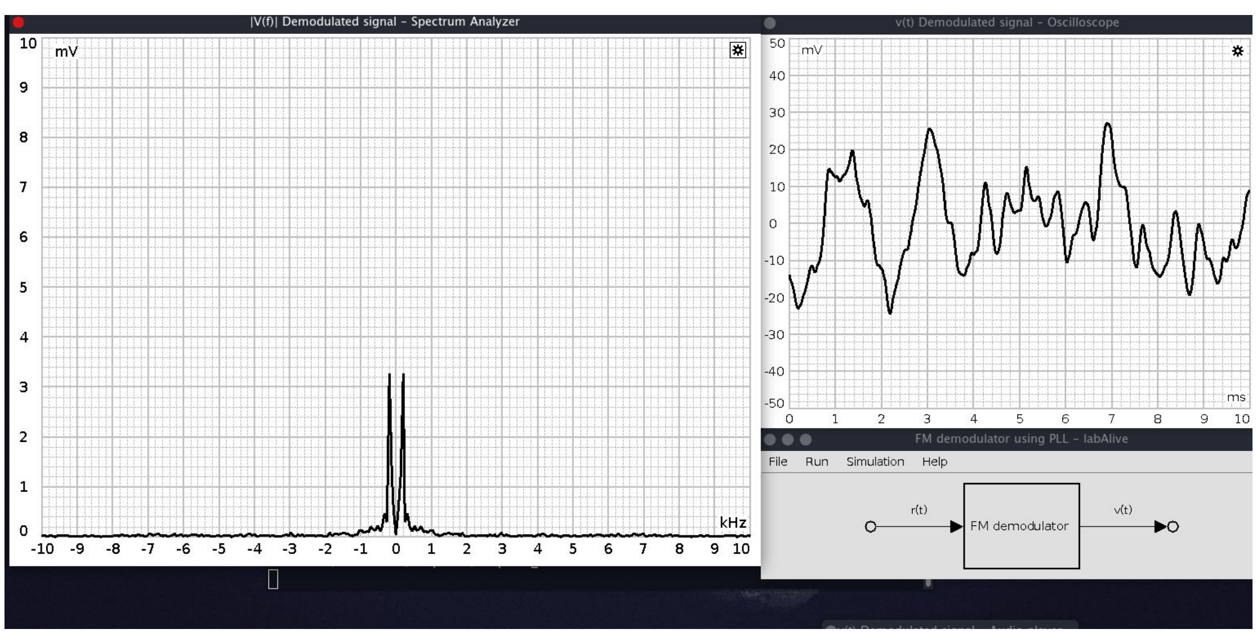


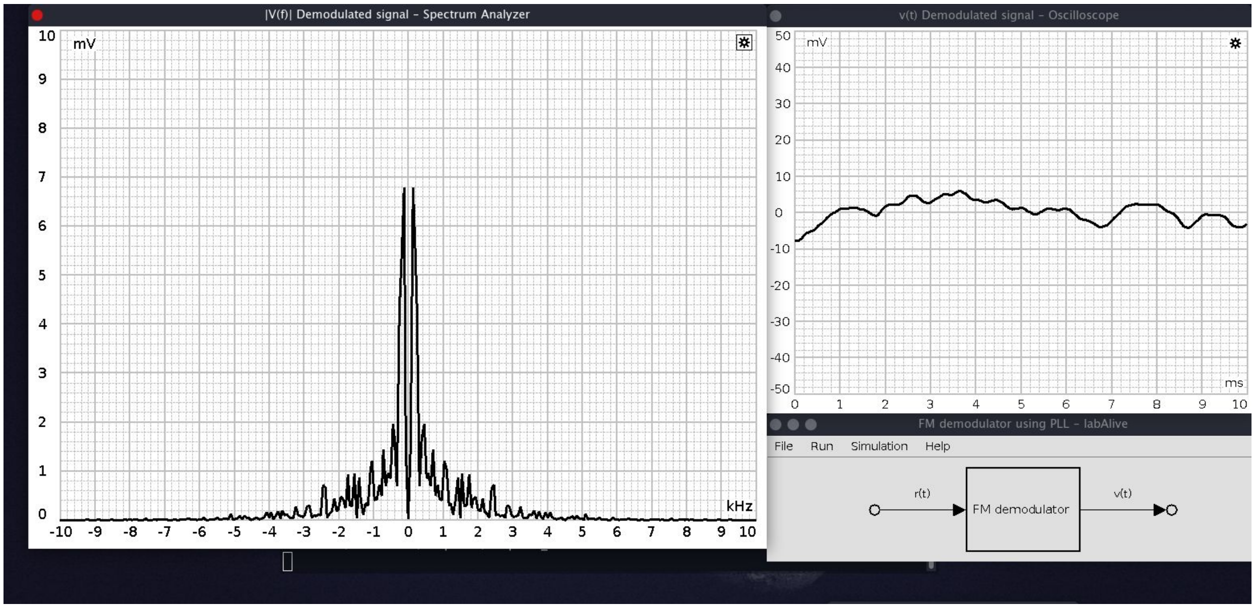
FM receiver using a PLL demodulator.

Notes:

- In the signal logging window, change the sample type to double and number of samples to 1G if you're not able to save the file.
- Listen to received audio signal and change the parameter values in modulator and demodulator block to listen the beat if hissing sound is coming.
- Take the screenshots of the graphs observed and paste it in the output waveforms

Output Waveforms:





Conclusion:

In this experiment, we have observed and studied the Spectrum of an FM signal using labAlive. We also performed FM Modulation and Demodulation of audio signals for various transmission parameters like Fm, Modulation Index, Message Signal Amplitude and observed their generated spectra.

Remark **Signature**

Experiment No.: 7 **Date:**

Aim:

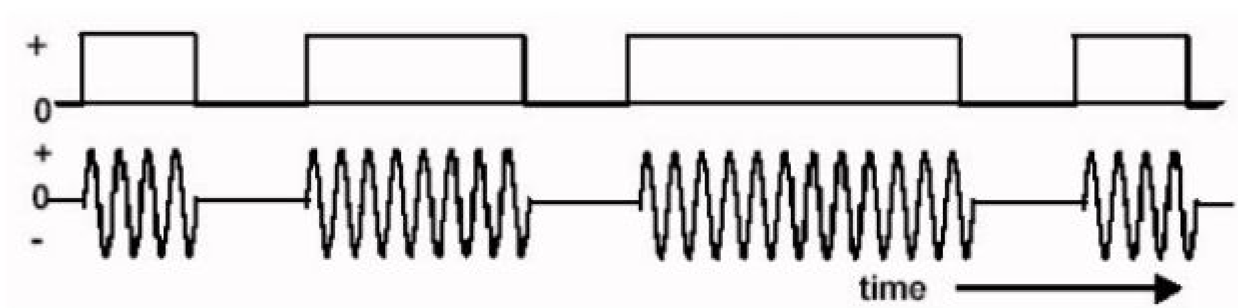
To Generate and demodulate an amplitude shift keying (ASK) signal.

Theory:

Amplitude Shift Keying (ASK) is the digital modulation technique. 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. This modulated signal can be expressed as:

$$x_c(t) = \begin{cases} 0 & \text{symbol "0"} \\ A\cos\omega_c t & \text{symbol "1"} \end{cases}$$

Amplitude shift keying (ASK) in the context of digital signal communications is a modulation process, which imparts to a sinusoid two or more discrete amplitude levels. These are 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. Figure 1 illustrates a binary ASK signal (lower), together with the binary sequence which initiated it (upper). Neither signal has been band limited.

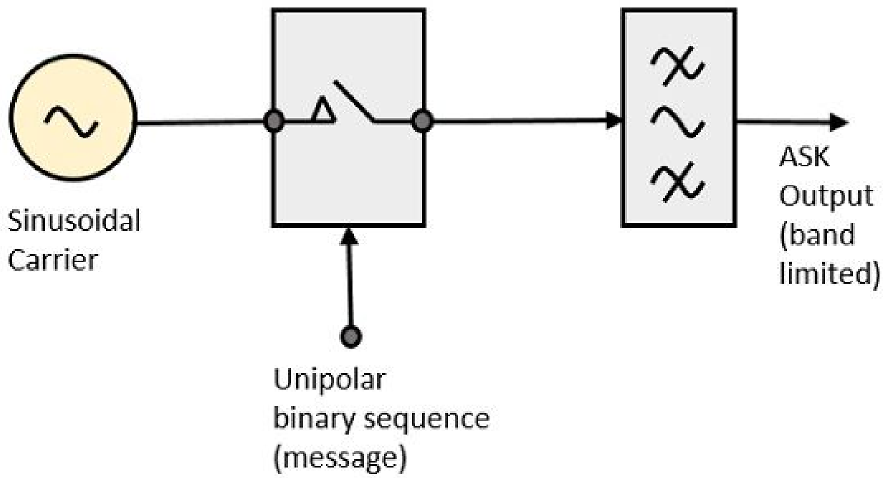


Message Signal & ASK Signal

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.

ASK Modulator:

The ASK modulator block diagram comprises of the carrier signal generator, the binary sequence from the message signal and the band-limited filter. Following is the block diagram of the ASK Modulator.



The carrier generator, sends a continuous high-frequency carrier. The binary sequence from the message signal makes the unipolar input to be either High or Low. The high signal closes the switch, allowing a carrier wave. Hence, the output will be the carrier signal at high input. When there is low input, the switch opens, allowing no voltage to appear. Hence, the output will be low.

The band-limiting filter shapes the pulse depending upon the amplitude and phase characteristics of the band-limiting filter or the pulse-shaping filter.

ASK Demodulator:

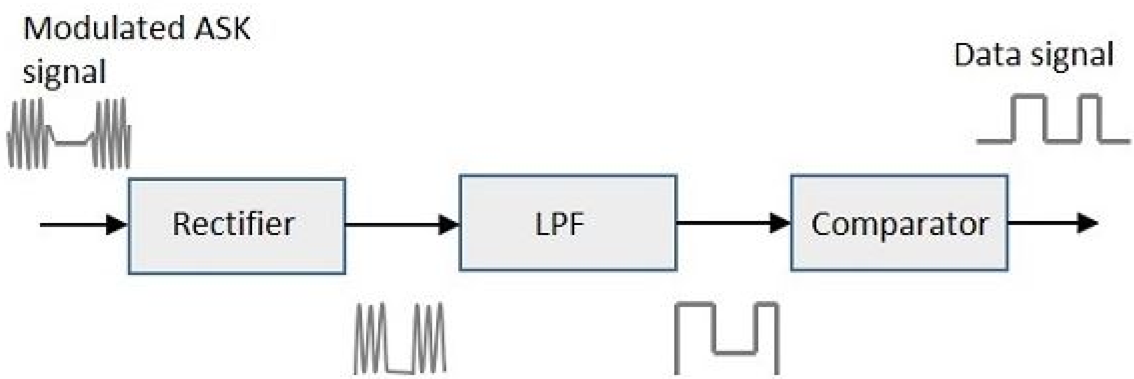
There are two types of ASK Demodulation techniques.

- Asynchronous ASK Demodulation/detection
- Synchronous ASK Demodulation/detection

The clock frequency at the transmitter, when matches with the clock frequency at the receiver, it is known as a Synchronous method, as the frequency gets synchronized. Otherwise, it is known as Asynchronous.

Asynchronous ASK Demodulator:

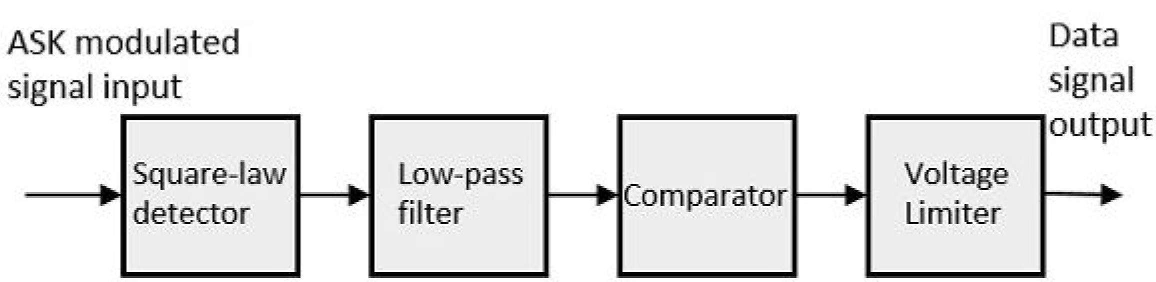
The Asynchronous ASK detector consists of a half-wave rectifier, a low pass filter, and a comparator. Following is the block diagram for the same.



The modulated ASK signal is given to the half-wave rectifier, which delivers a positive half output. The low pass filter suppresses the higher frequencies and gives an envelope detected output from which the comparator delivers a digital output.

Synchronous ASK Demodulator:

Synchronous ASK detector consists of a Square law detector, low pass filter, a comparator, and a voltage limiter. Following is the block diagram for the same.



The ASK modulated input signal is given to the Square law detector. A square law detector is one whose output voltage is proportional to the square of the amplitude modulated input voltage. The low pass filter minimizes the higher frequencies. The comparator and the voltage limiter help to get a clean digital output.

Algorithm to implement ASK modulation & demodulation on MATLAB:

ASK Modulation:

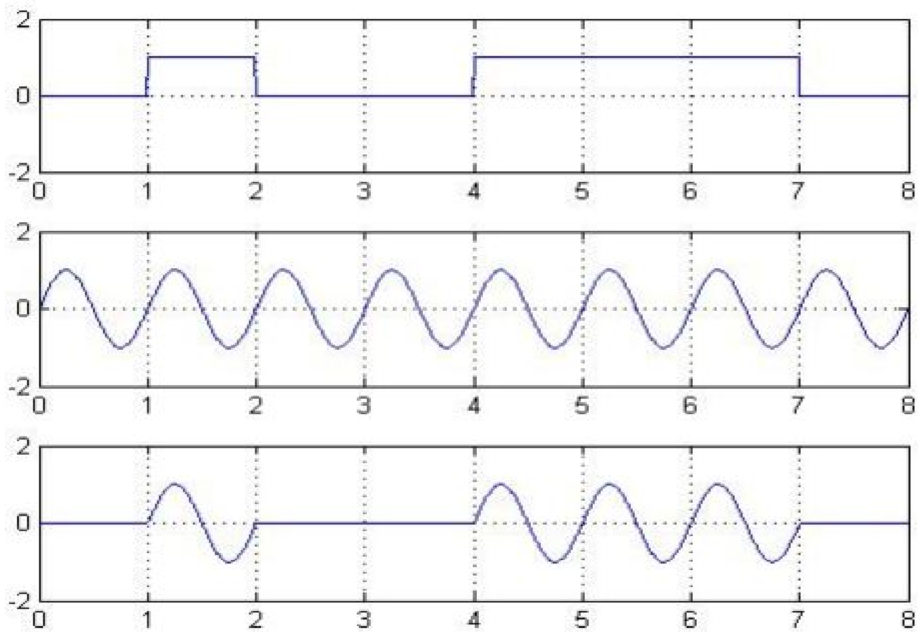
1. Generate a carrier signal of frequency fc.
2. Start a FOR Loop.
3. Generate a binary sequence, a message signal.
4. Generate ASK modulated signal, which will transmit carrier signal for logic 1 and zero signal for logic 0.
5. Plot message signal and ASK modulated signal.
6. End FOR Loop.
7. Plot binary data and carrier.

ASK Demodulation:

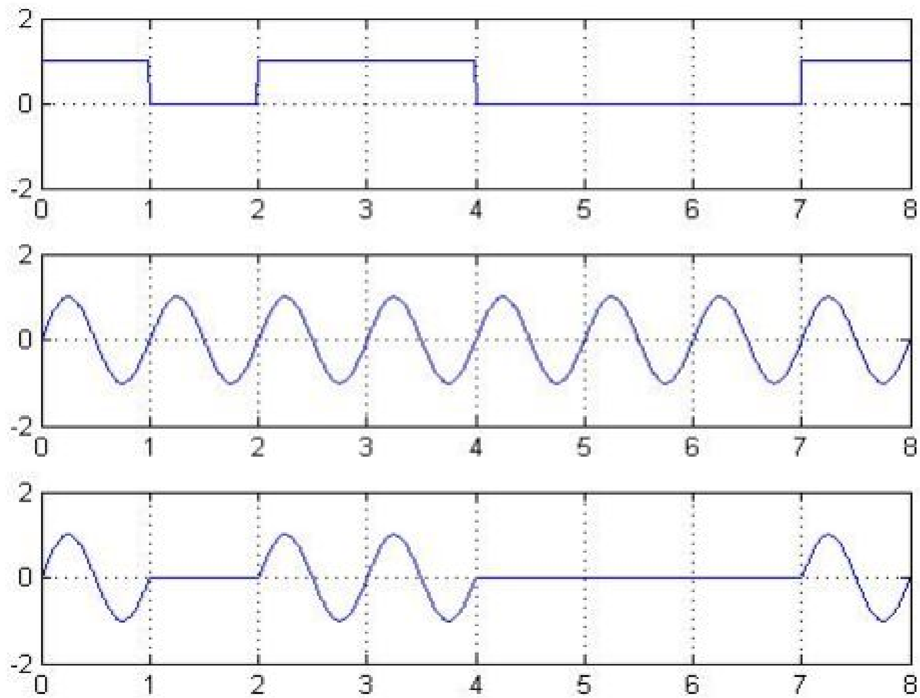
1. Start FOR Loop.
2. Perform correlation of ASK signal with carrier to get decision variable.
3. Make decision to get demodulated binary data. If $x > 0$, choose ‘1’ else choose ‘0’.
4. Plot the demodulated binary data.

Expected Result:[1. Message signal 2. Carrier signal 3. ASK signal]

- Observation waveform for the bit stream [0 1 0 0 1 1 1 0]



- Observation waveform for the bit stream [1 0 1 1 0 0 0 1]

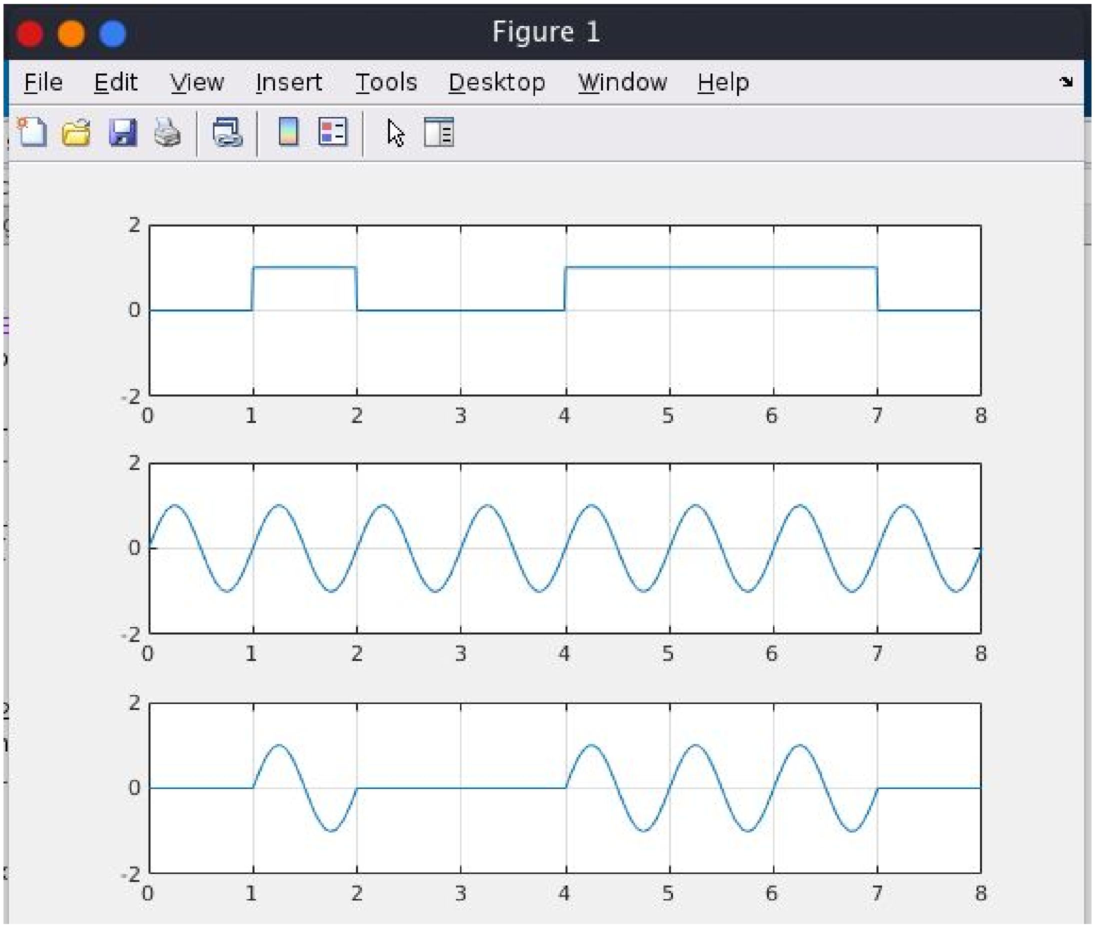


MATLAB Code:

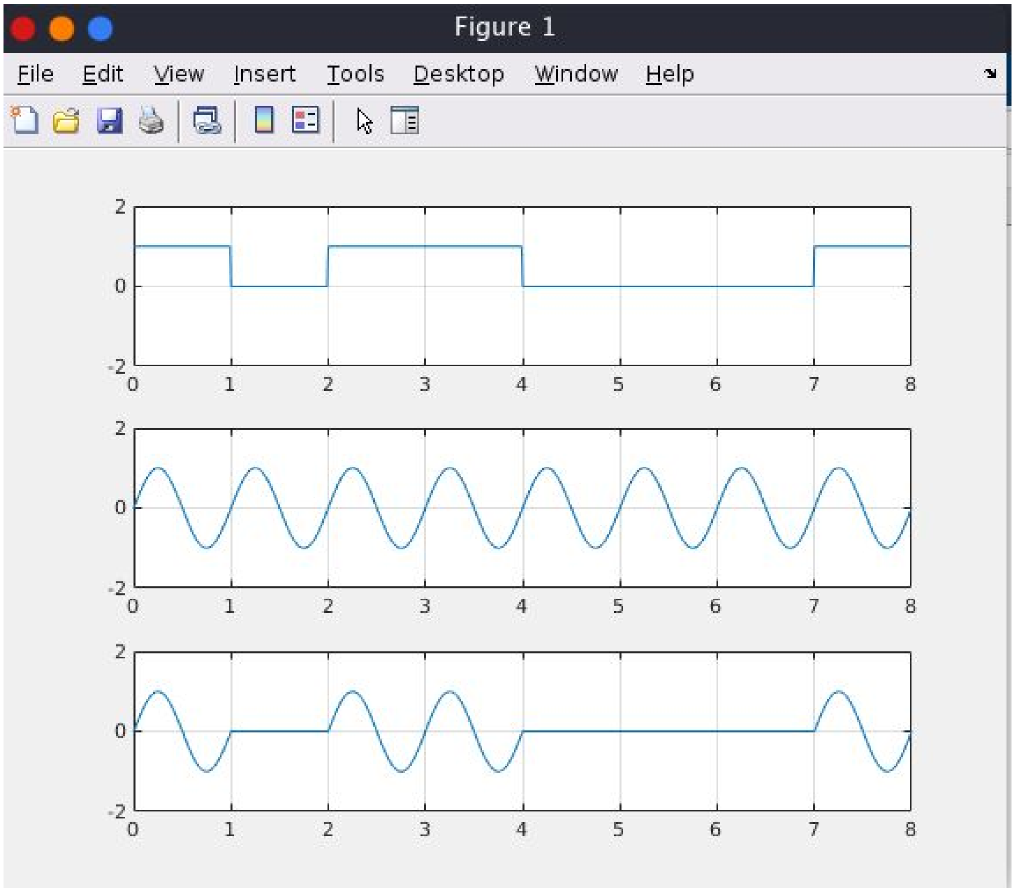
```
clc

clear all

b = input('Enter the Bit stream \n ');

n = length(b);

t = 0:.01:n;

x = 1:1:(n+1)*100;

for i = 1:n

      for j = i:.1:i+1

            bw(x(i*100:(i+1)*100)) = b(i);

      end

end
```

```
bw = bw(100:end);

sint = sin(2*pi*t);

st = bw.*sint;

%Plotting all using subplot

subplot(3,1,1)

plot(t,bw)

grid on ; axis([0 n -2 +2])

subplot(3,1,2)

plot(t,sint)

grid on ; axis([0 n -2 +2])

subplot(3,1,3)

plot(t,st)

grid on;

axis([0 n -2 +2])
```

Result:



01001110



10110001

Application of ASK:

- Low-frequency RF applications
- Home automation devices
- Industrial networks devices
- Wireless base stations
- Tire pressuring monitoring systems

References:

1. ASK Basics, waveforms, introduction to modulation & demodulation:

https://www.youtube.com/watch?v=ucrZlde8vtk

2. ASK Modulation:

https://www.youtube.com/watch?v=AmRjVzHFBAk

3. ASK Demodulation:

https://www.youtube.com/watch?v=_uf3j5Y2sNs

4. ASK on Kit:

https://www.youtube.com/watch?v=NDvRg-R8Mwo

5. ASK on MATLAB:

https://www.youtube.com/watch?v=x7mjVGFKnk8

Conclusion:

In this experiment we learnt how to generate and demodulate amplitude shift keying signals using ***MATLAB.***

Signature **Remarks**

EXPERIMENT 7b

AIM: To study Frequency Shift Keying (FSK) Modulation.

APPARATUS: MATLAB Simulink.

BLOCK DIAGRAM:

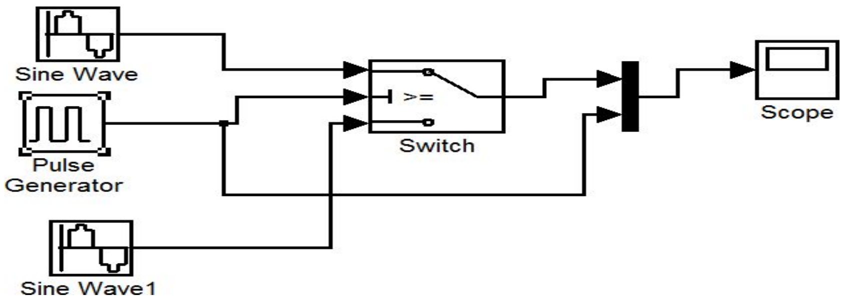


Fig.1: Block Diagram of FSK Modulator in Simulink MATLAB

THEORY:

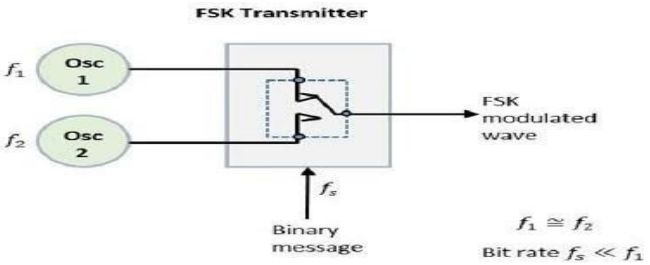


Fig.2: Basic principle of FSK modulator block

Frequency-shift keying (FSK) is a frequency modulation scheme in which digital information is transmitted through discrete frequency changes of a carrier wave. The simplest FSK is binary FSK (BFSK). BFSK uses a pair of discrete frequencies to transmit binary (0s and 1s) information. With this scheme, the "1" is called the mark frequency and the "0" is called the space frequency. If the incoming bit is 1, a signal with frequency f1 is sent for the duration of the bit. If the bit is 0, a signal with frequency f2 is sent for the duration of this bit. This is the basic principle behind FSK modulation.

WAVE FORM

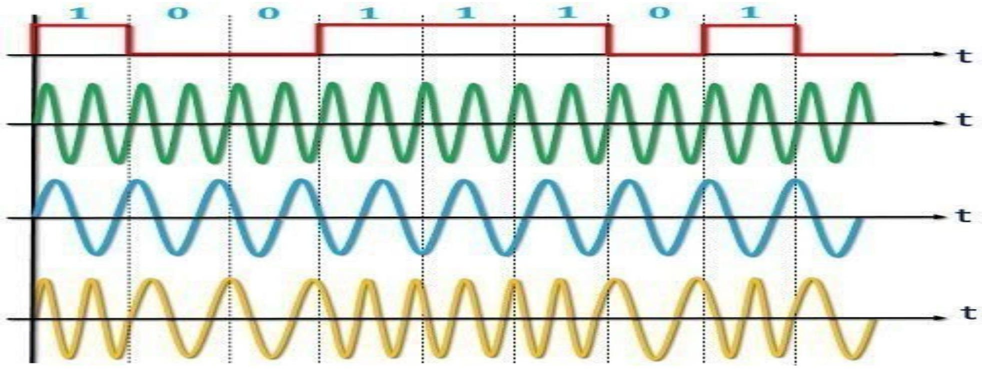


Fig.3: Waveform of FSK Modulator

First waveform is Digital bit stream according to it switching process will be proceed. Second and third waveforms are HIGH frequency carrier wave & LOW frequency carrier waveform respectively. Fourth is FSK modulation wave , here when input bit stream is 1 then get HIGH frequency and 0 then LOW frequency.

PROCEDURE:

Modulation:

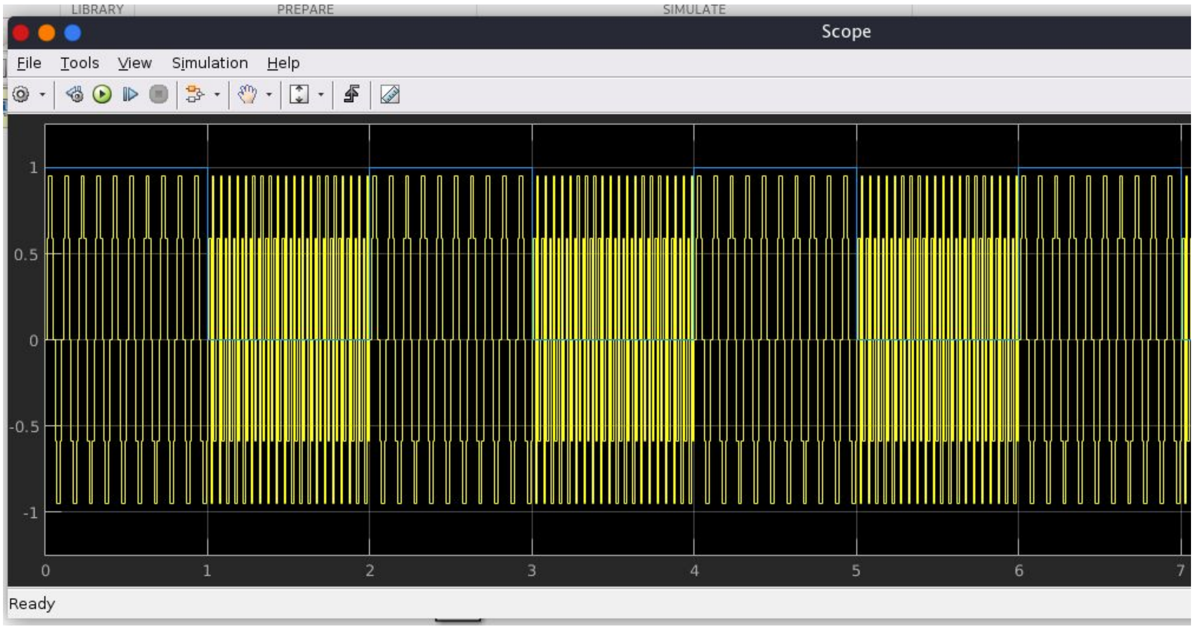
1. Connect all the blocks in Simulink according to given steps.(Which is given in FSK_designingStep document).
2. After designing entire diagram click on RUN.
3. Observe the waveforms at output of modulator using virtual scope.

OBSERVATION TABLE:

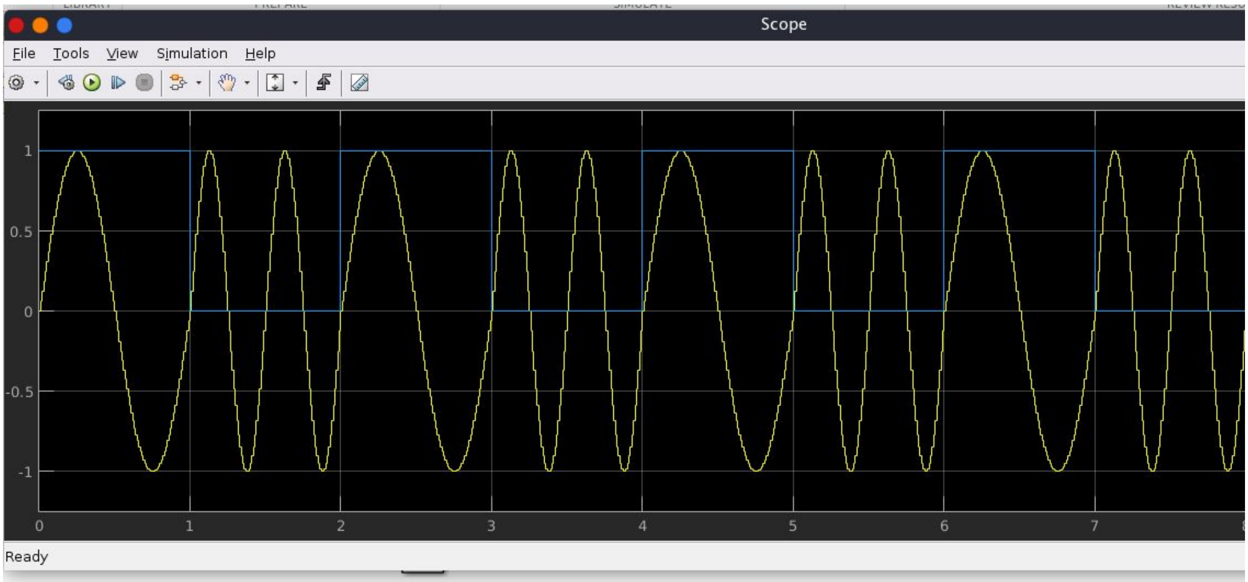
To observe waveform in Simulink by selecting different frequencies as per given Table.

Lower Frequency	Higher Frequency
10Hz	40Hz
1Hz	2Hz
2Hz	5Hz

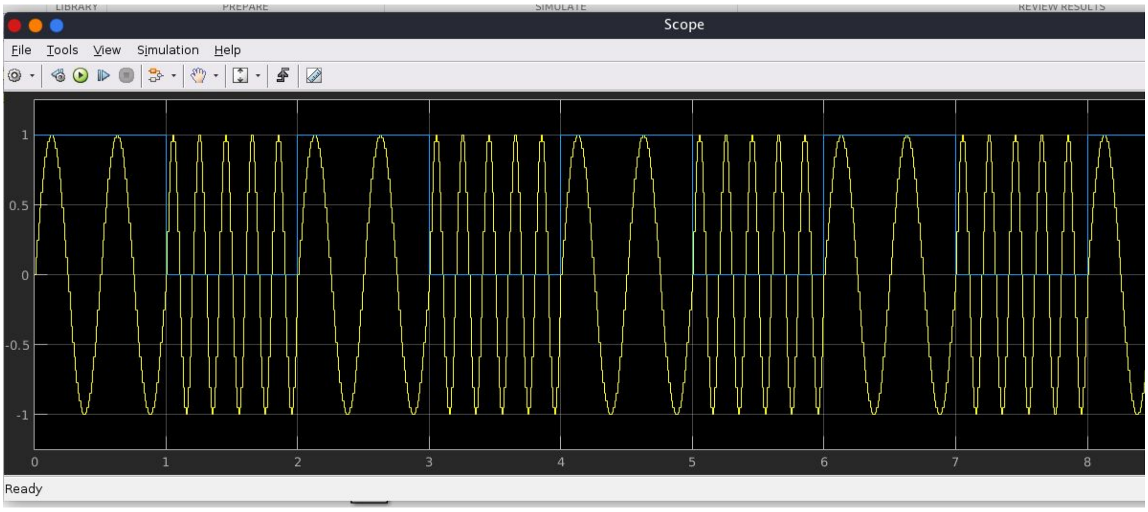
RESULT:



Simulink Waveform of FSK Modulator for 10Hz Low & 40Hz High frequencies



Simulink Waveform of FSK Modulator for 1Hz Low & 2Hz High frequencies



Simulink Waveform of FSK Modulator for 2Hz Low & 5Hz High frequencies

CONCLUSION:

In this experiment we performed FSK modulation using MATLAB simulink for various frequency pairs.

Remarks: **Signature:**

EXPERIMENT 7c

AIM: To study Binary Phase Shift Keying (BPSK) Modulation.

APPARATUS: MATLAB Simulink.

BLOCK DIAGRAM:

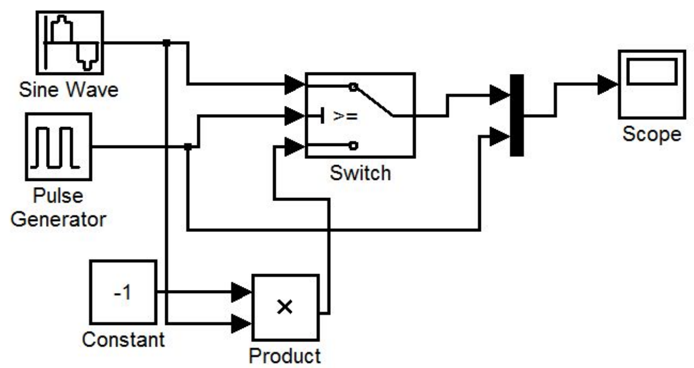


Fig.1: Block Diagram of BPSK Modulator in Simulink MATLAB

THEORY:

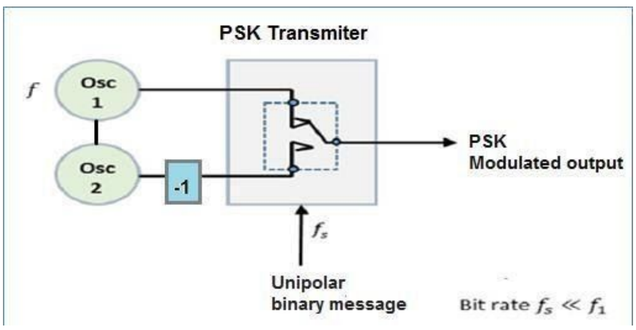


Fig.2: Basic principle of BPSK modulator block

BPSK is a digital modulation scheme which is analogues to phase modulation. Binary Phase Shift Keying (BPSK) is the simplest form of PSK. In binary phase shift keying two output phases are possible for a single carrier frequency one out of phase represent logic 1 and logic 0. As the input digital binary signal change state the phase of output carrier shift two angles that are 180^o out of phase.

Here input bit stream is unipolar, so instead of multiplication technique use discrete frequency changes technique same as FSK. But here both frequency carrier sources are same but one of it multiplied with -1. If the incoming bit is 1, a signal with frequency f is sent for the duration of the bit & so no phase shift or 0^o phase shift. If the bit is 0 then same frequency of carrier signal is sent but first it multiply with -1 so that get 180^o phase shift at output.

WAVE FORM

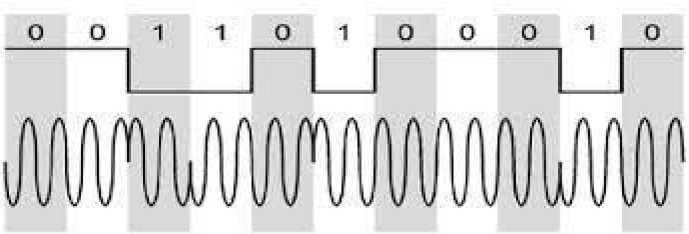


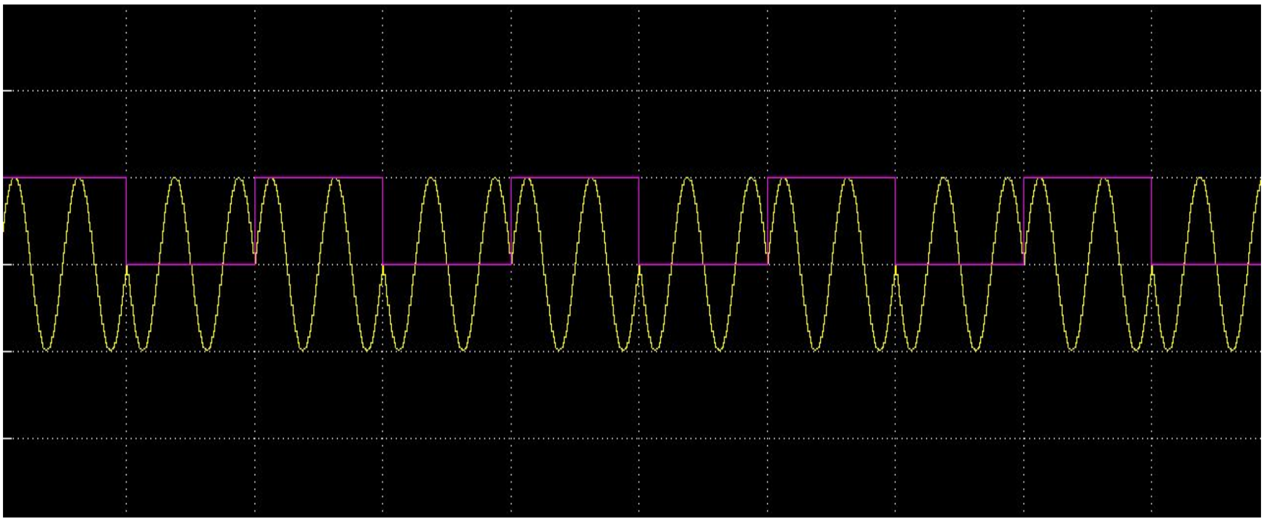
Fig.3: Waveform of BPSK Modulator

First waveform is Digital bit stream according to it switching process will be proceed. Second is BPSK modulation wave, here when input bit stream is 1 then get direct signal wave at output without shifting. But when 0 is come then carrier signal multiply with -1 so that whatever amplitude of carrier signal has that reverse its value; and output of such signal shows 180^o phase shifted. So when input 1 then output is 0^o shifted and 0 then 180^o shifted.

PROCEDURE:

Modulation:

1. Connect all the blocks in Simulink according to given steps.(Which is given in PSK_designingStep document).
2. After designing entire diagram click on RUN.
3. Observe the waveforms at output of modulator using virtual scope.

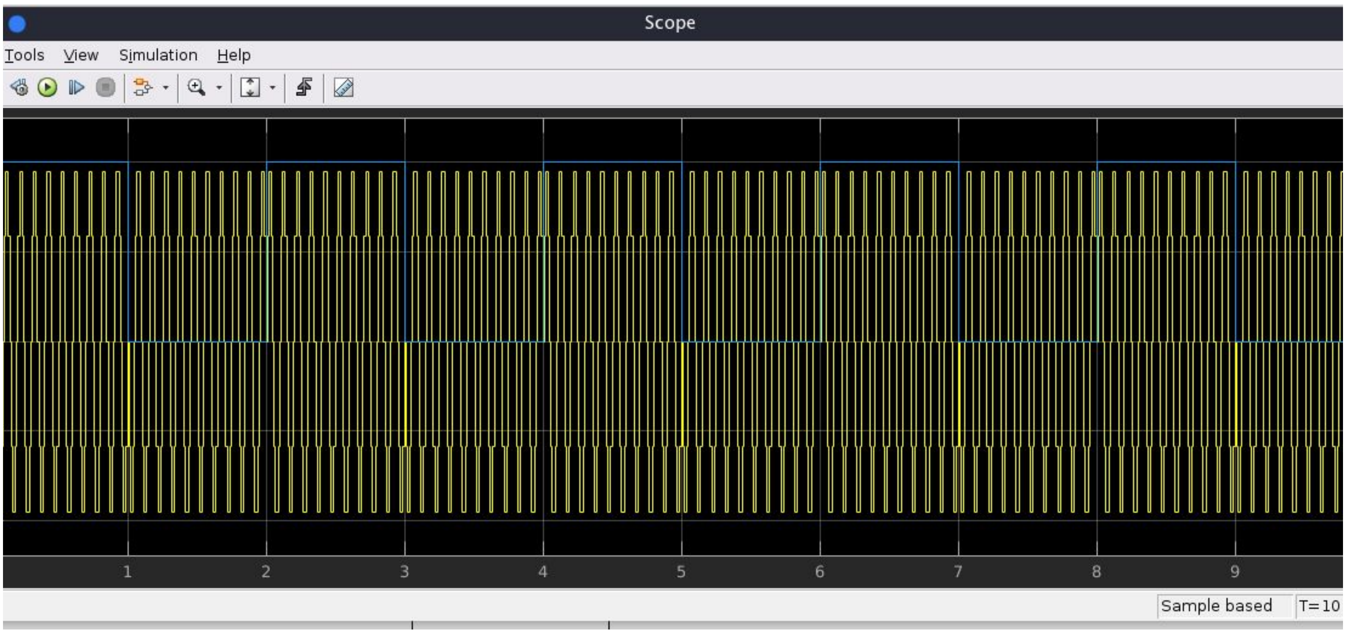


OBSERVATION TABLE:

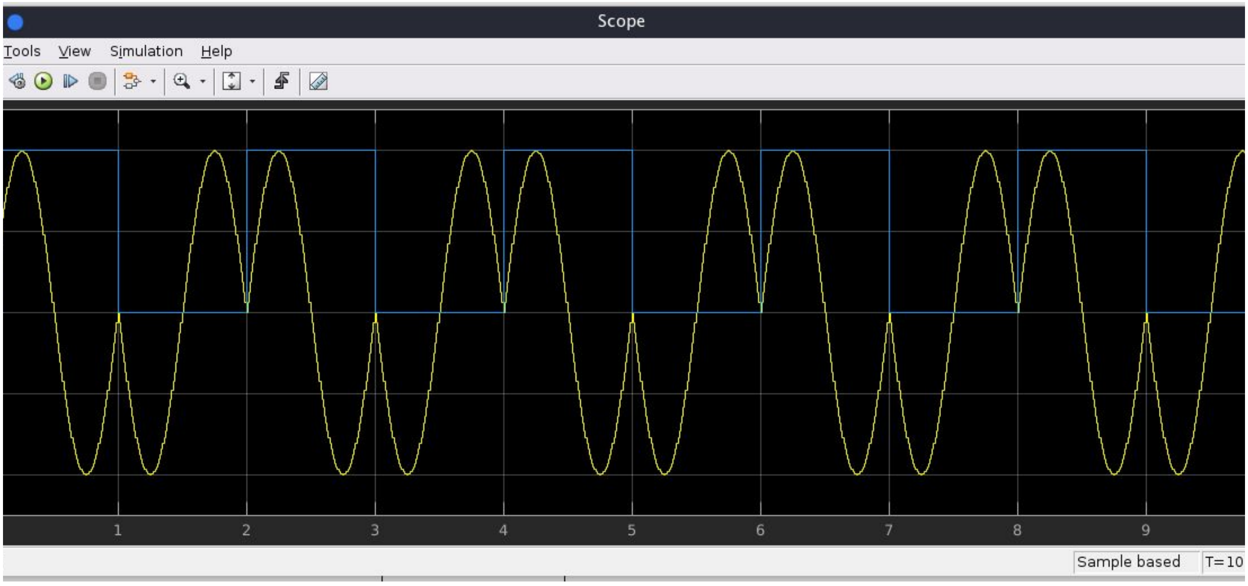
To observe waveform in Simulink by selecting different frequencies as per given Table.

Frequency
10Hz
1Hz
2Hz

RESULT:



Simulink Waveform of BPSK Modulator for 10Hz frequency



Simulink Waveform of BPSK Modulator for 1Hz frequency

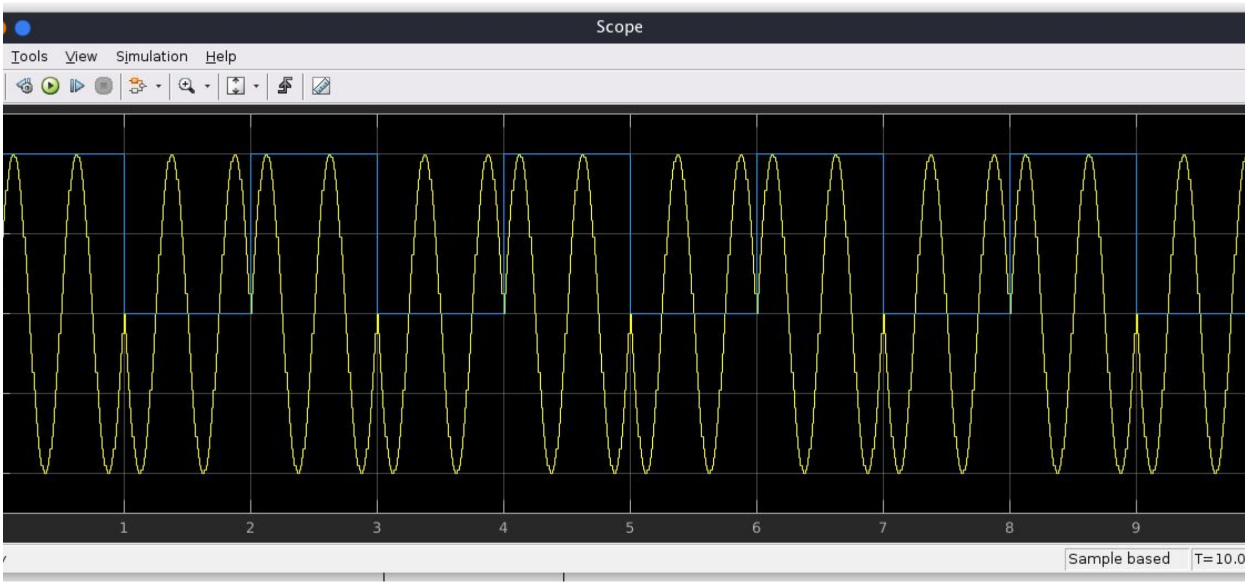


Fig.4: Simulink Waveform of BPSK Modulator for 2Hz frequency

CONCLUSION:

In this experiment we performed Binary phase shift keying(BPSK) modulation using MATLAB simulink for various frequencies.

Remarks: **Signature:**

AMPLITUDE MODULATION IN MATLAB

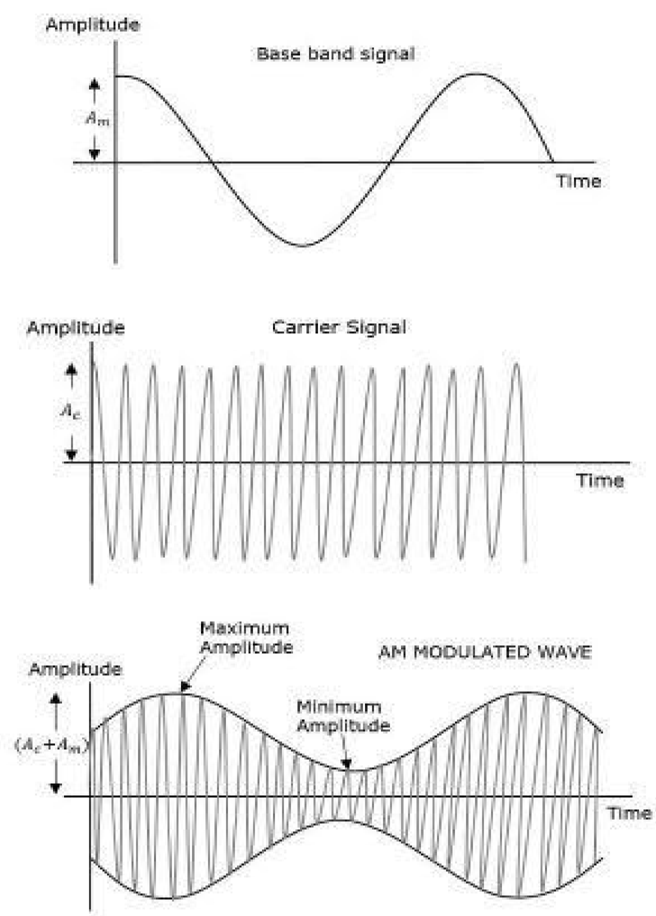
Experiment No: 8 **Date:**

Aim: To implement amplitude modulation and demodulation using MATLAB.

Brief Theory/Equations:

A continuous-wave goes on continuously without any intervals and it is the baseband message signal, which contains the information. This wave has to be modulated.

According to the standard definition, “The amplitude of the carrier signal varies in accordance with the instantaneous amplitude of the modulating signal.” Which means, the amplitude of the carrier signal containing no information varies as per the amplitude of the signal containing information, at each instant. This can be well explained by the following figures.



The first figure shows the modulating wave, which is the message signal. The next one is the carrier wave, which is a high frequency signal and contains no information. While, the last one is the resultant modulated wave.

It can be observed that the positive and negative peaks of the carrier wave, are interconnected with an imaginary line. This line helps recreating the exact shape of the modulating signal. This imaginary line on the carrier wave is called as Envelope. It is the same as that of the message signal.

Time-domain Representation of the Waves

Let the modulating signal be,

$$m(t) = A_m cos(2\pi f_m t)$$

and the carrier signal be,

$$c(t) = A_c cos(2\pi f_c t)$$

Where,

Am and *Ac* are the amplitude of the modulating signal and the carrier signal respectively. *fm* and *fc* are the frequency of the modulating signal and the carrier signal respectively. Then, the equation of Amplitude Modulated wave will be

$$s(t) = [Ac + Am\ cos(2\pi fmt)]cos(2\pi fct)$$

Modulation index:

A carrier wave, after being modulated, if the modulated level is calculated, then such an attempt is called as Modulation Index or Modulation Depth. It states the level of modulation that a carrier wave undergoes.

$$s\,(t) = Ac\ [1 + \left(\frac{Am}{Ac}\right)\ cos(2\pi fmt)]cos(2\pi fct)$$

$$s\,(t) = Ac\ [1 + \mu\ cos(2\pi fmt)]cos(2\pi fct)$$

Where, μ is Modulation index and it is equal to the ratio of Am and Ac. Hence, we can calculate the value of modulation index by using the above formula, when the amplitudes of the message and carrier signals are known.

Now, let us derive one more formula for Modulation index by considering Equation. We can use this formula for calculating modulation index value, when the maximum and minimum amplitudes of the modulated wave are known.

Let Amax and Amin be the maximum and minimum amplitudes of the modulated wave. We will get the maximum amplitude of the modulated wave, when $cos(2\pi fmt)$ is 1. $Amax=Ac+Am$.

We will get the minimum amplitude of the modulated wave, when $cos(2\pi fmt)$ is -1. $Amin=Ac-Am$

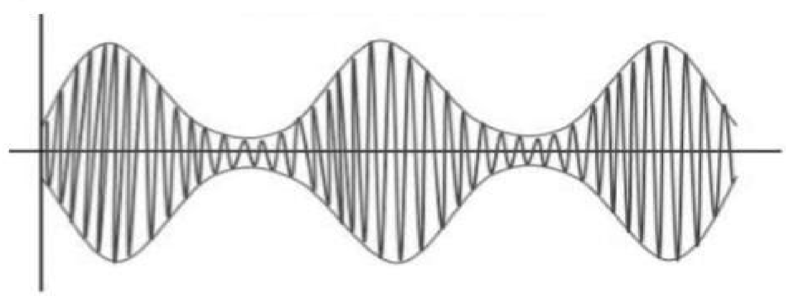
$$Amax+Amin = Ac+Am+Ac-Am=2\,Ac \;\Rightarrow\; Ac = \frac{Amax+Amin}{2}$$

$$Amax-Amin = Ac+Am-(Ac-Am)=2\,Am \;\Rightarrow Am= \frac{Amax-Amin}{2}$$

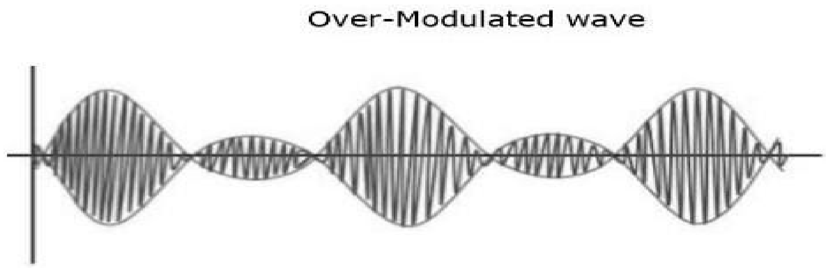
$$\frac{Am}{Ac} = \frac{(Amax-Amin)/2}{(Amax+Amin)/2} \Rightarrow \mu = \frac{Amax-Amin}{Amax+Amin}$$

Therefore, there are the two formulas for Modulation index. The modulation index or modulation depth is often denoted in percentage called as Percentage of Modulation. We will get the **percentage of modulation**, just by multiplying the modulation index value with 100. For a perfect modulation, the value of modulation index should be 1, which implies the percentage of modulation should be 100%. For instance, if this value is less than 1, i.e., the modulation index is 0.5, then the modulated output would look like the following figure. It is called as **Under-modulation**. Such a wave is called as an **under-modulated wave**.

Under-Modulated wave



If the value of the modulation index is greater than 1, i.e., 1.5 or so, then the wave will be an **over-modulated wave**. It would look like the following figure. As the value of the modulation index increases, the carrier experiences a 180^o phase reversal, which causes additional sidebands and hence, the wave gets distorted. Such an over-modulated wave causes interference, which cannot be eliminated.



Reference: Modern Digital and Analog Communication by B.P.Lathi

Bandwidth of AM:

Bandwidth (BW) is the difference between the highest and lowest frequencies of the signal. Mathematically, we can write it as $BW = f_{max} - f_{min}$ Consider the following equation of amplitude modulated wave.

$$s(t) = Ac[1 + \mu\, cos(2\pi fmt)]cos(2\pi fc\ t)$$

$$s\,(t) = Ac\ cos\,(2\pi fct) + Ac\ \mu\ cos\,(2\pi fct)\, cos(2\pi fmt)$$

$$s\,(t) = Ac\ cos(2\pi fct) + \frac{Ac\,\mu}{2} cos[2\pi(fc + fm)t] + \frac{Ac\,\mu}{2} cos[2\pi(fc - fm)t]$$

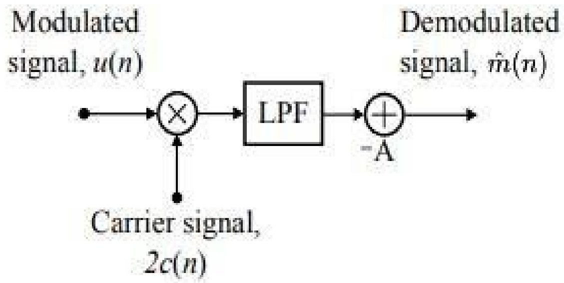
Hence, the amplitude modulated wave has three frequencies. Those are carrier frequency *fc,* upper sideband frequency *fc+fm* and lower sideband frequency *fc−fm* Here, *fmax=fc+fm* and *fmin=fc−fm.* Substitute, *fmax* and *fmin* values in bandwidth formula.

$$BW=(fc+fm) - (fc-fm) \Rightarrow BW=2fm$$

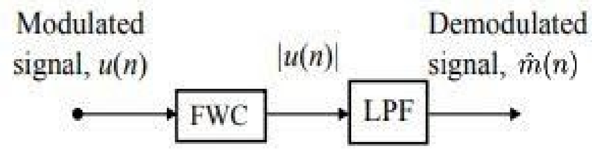
Thus, it can be said that the bandwidth required for amplitude modulated wave is twice the frequency of the modulating signal.

Amplitude Demodulation:

There are two types of demodulation: synchronous demodulation and asynchronous demodulation. They differ in the use of a carrier signal in demodulation processes. The receiver must generate a carrier signal synchronized in phase and frequency for synchronous demodulation, but the carrier signal is not needed in asynchronous demodulation.



(a) Synchronous demodulator



(b) Asynchronous demodulator

Synchronous Demodulation. This kind of demodulation can be referred to as a coherent or product detector. Figure (a) shows a flow chart of a synchronous demodulator. At the receiver, we multiply the incoming modulated signal by a local carrier of frequency and phase in synchronism with the carrier used at the transmitter.

$$e(t) = u(t)\,c(t) = [A + m(t)]\ cos^2(w_c t)$$

$$= \tfrac{1}{2}\ [A + m(t)] + \tfrac{1}{2}\ [A + m(t)]\ cos(2w_c t)$$

Let us denote $m_a(t) = A + m(t)$

$$e(t) = \tfrac{1}{2}\, m_a(t) + \tfrac{1}{2}\, m_a(t)\ cos(2w_c t)$$

The fourier transform of the signal e(t) is

$$E(w) = \tfrac{1}{2}\, M_a(w) + \tfrac{1}{2}[\, M_a(w + 2w_c) + M_a(w - 2w_c)$$

The spectrum $E(w)$ consists of three components as shown in figure. The first component is the message spectrum. The two other components, which are the modulated signal of m(t) with carrier frequency $2\,w_c$., are centered at $\pm 2\,w_c$. The signal *e(t)* is then filtered by a lowpass filter (LPF) with a cut-off frequency of fc to yield $\frac{m_a(t)}{2}$. We can fully get $m_a(t)$ by multiplying the output by two. We can also get rid of the inconvenient fraction 1 2 from the output by using the carrier 2 $cos(w_c)$. instead of $cos(w_c)$. Finally, the message signal m(t) can be recovered by ˆm(t) = $m_a(t)$ − A.

Asynchronous Demodulation. An asynchronous demodulator can be referred to as an envelope detector. For an envelope detector, the modulated signal *u(t)* must satisfy the requirement that $A + m(t) \geq 0,\ \forall t$. A block diagram of the asynchronous demodulator is shown in Figure (b). The incoming modulated signal, u(t), is passed through a full wave rectifier (FWR) which acts as an absolute function. The FWR output which is the absolute value of *u(t)*, $|u(t)|$, is then filtered by a low-pass filter resulting in the demodulated signal, *ˆm(t)*.

Algorithm:

- Define the sampling frequency $f_s \geq 2(f_m)$ say, $f_s = 100Hz$;
- Define the time range using the sampling frequency t= -10 : $1/f_s$: 10

- Consider, message signal, $m = A_m \cos cos\ \omega_m t$, where $f_m = 1Hz$ and carrier signal $c = A_c \cos cos\ \omega_c t$ where $f_c = 10Hz$. Keep the amplitude of message and carrier signal same.
- Assume $w_m = 2\pi f_m$ and $w_c = 2\pi f_c$
- For AM, $s(t) = [Ac + Am\ cos(2\pi fmt)]cos(2\pi fct)$
- Figure1: Plot input signal, carrier signal, AM signal using subplot;
- Figure 2: plot the frequency spectrum of AM signal using the commands fft and fftshift.
  - Define n=length(t) %gives no. of columns in t
  - Define the step size for frequency axis fp which should be of same size as that of t. (matrix dimensions must match for plotting).
    - df=fs/n; where fs is the sampling frequency
  - Define frequency axis fp = -fs/2:df:fs/2-df (± df for getting same size matrices)
  - Take the fourier transform of AM using fft command
  - Y = fft(x) returns the discrete Fourier transform (DFT) of vector x, computed with a fast Fourier transform (FFT) algorithm.
  - Y = fftshift(X) rearranges the outputs of fft by moving the zero-frequency component to the center of the array. It is useful for visualizing a Fourier transform with the zero-frequency component in the middle of the spectrum.
  - Can write in a single syntax as y=fftshift(fft(x));
  - Plot the frequency spectrum of AM using the command plot(fp,y).

- **Demodulate the AM signal**
  - Multiply the above signal with carrier signal.
    - Do .* element wise multiplication
- Do the low pass filtering of the above signal using butter and filter command.
  - [b,a] = butter(n,Wn,'ftype'). **This creates a filter**
  - [b,a] = butter(n,Wn) designs an order n lowpass digital Butterworth filter with normalized cutoff frequency Wn. It returns the filter coefficients in length n+1 row vectors b and a, with coefficients in descending powers of z.

$$H(z) = \frac{b(1) + b(2)z^{-1} + \ldots + b(n+1)z^{-n}}{1 + a(2)z^{-1} + \ldots + a(n+1)z^{-n}}$$

  - where the string 'ftype' is one of the following:
    - 'high' for a highpass digital filter with normalized cutoff frequency Wn
    - 'low' for a lowpass digital filter with normalized cutoff frequency Wn
    - 'stop' for an order 2*n bandstop digital filter if Wn is a two-element vector, Wn = [w1 w2]. The stopband is w1 < ω < w2.
    - 'bandpass' for an order 2*n bandpass filter if Wn is a two-element vector, Wn = [w1 w2]. The passband is w1 < ω < w2. Specifying a two-element vector, Wn, without an explicit 'ftype' defaults to a bandpass filter.

- Cutoff frequency, Wn is that frequency where the magnitude response of the filter is . For butter, the normalized cutoff frequency Wn must be a number between 0 and 1, where 1 corresponds to the Nyquist frequency, π radians per sample.
- Choose order, n=3 ; Wn=fm/fs; ftype='low'

- Use the filter command to apply the filter
  - y = filter(b,a,X), where b and a are the coefficients obtained in the previous step and X will the multiplied signal output. y is the filtered output. Adjust order so that you get approximate input signal.
- Figure 3: Plots for AM demodulation
  - Include three figures using subplot(211) to subplot(212)
    - 1^{st} - multiplied signal output (modulated AM signal*Carrier Signal)
    - 2^{nd} – filtered output y. Include message signal as well in this
      - Example: plot(t,m,t,y)

Expected Output waveforms:

Figure 1:

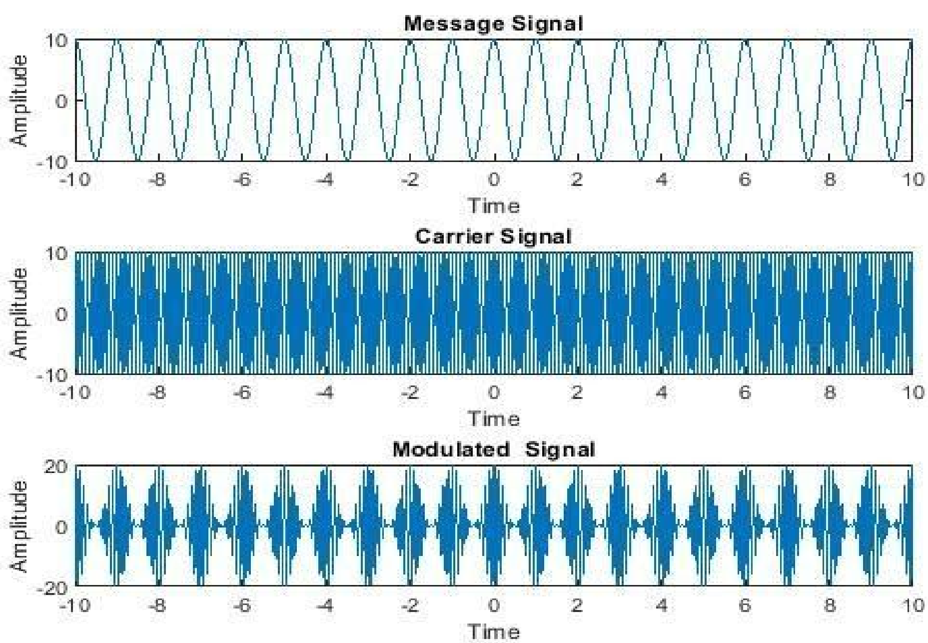


Figure 2:

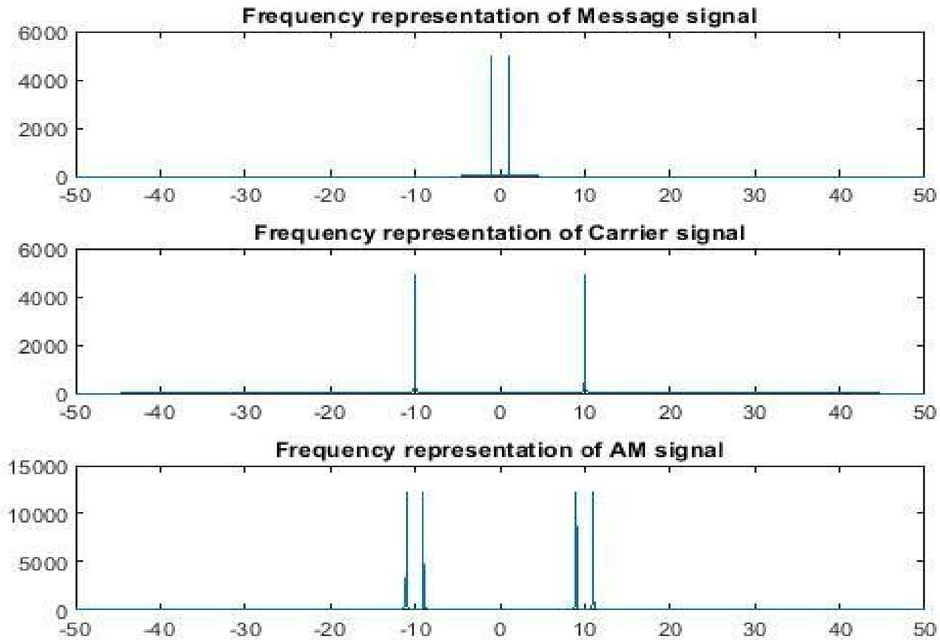
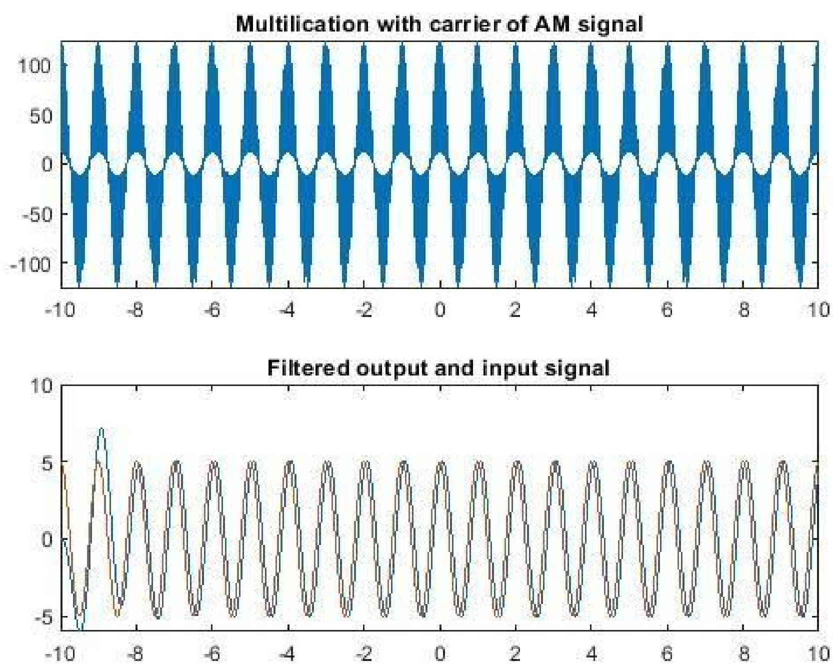

Frequency representation of Message signal
Frequency representation of Carrier signal
Frequency representation of AM signal

Figure 3:


Multilication with carrier of AM signal
Filtered output and input signal

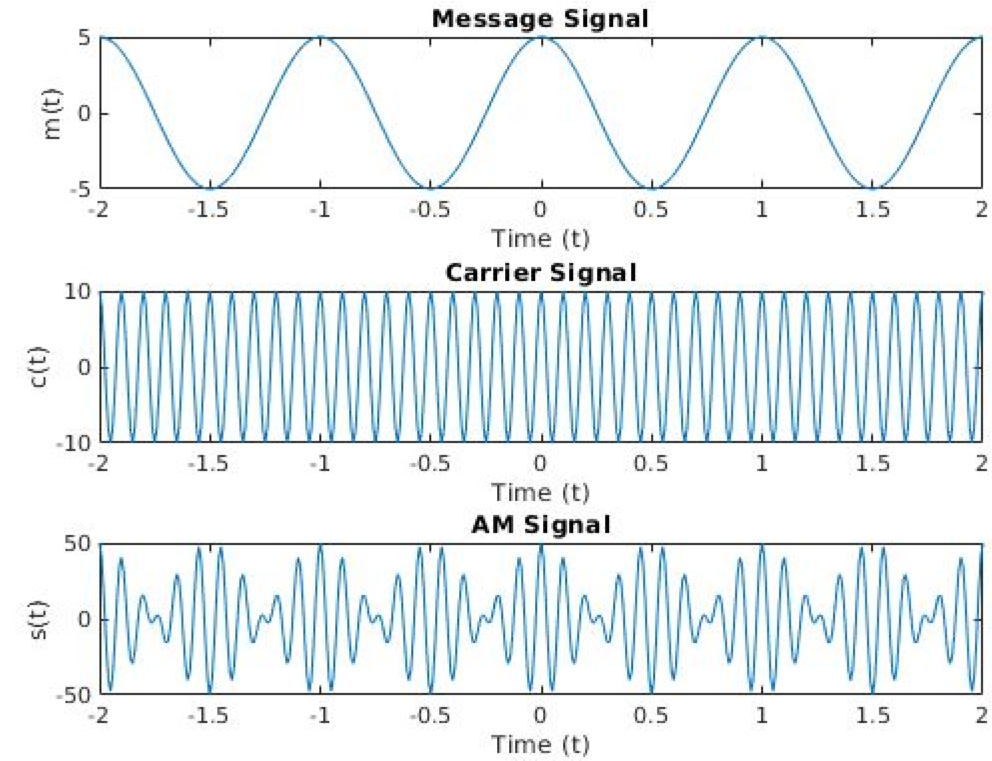
Code:

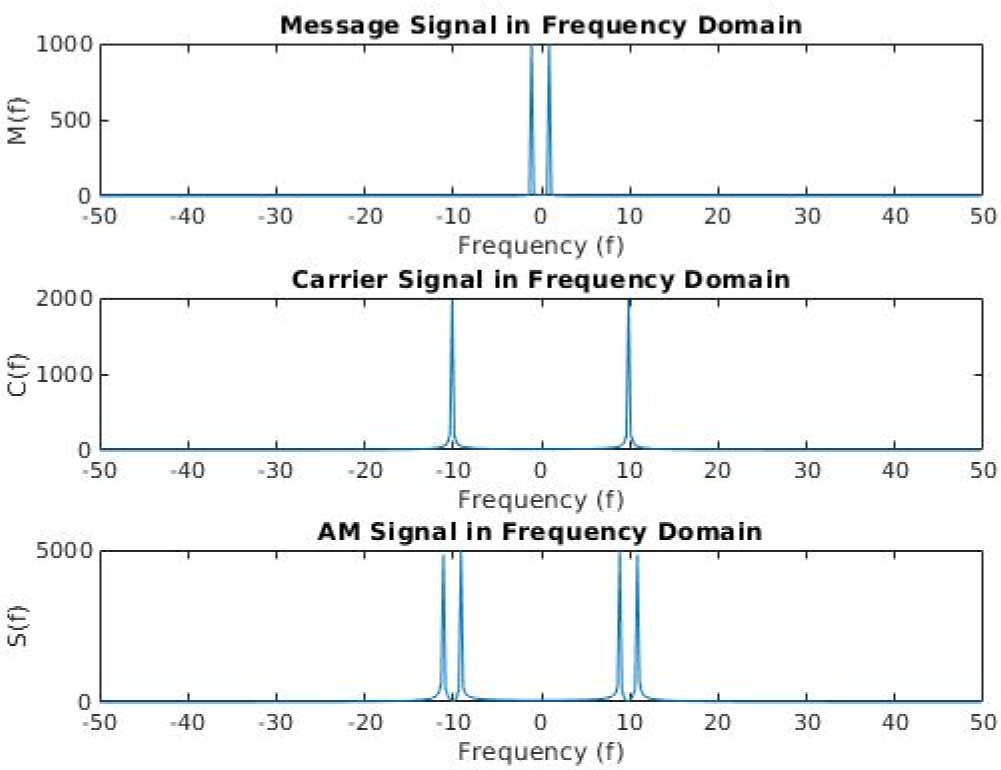
```
clear all;

clc;

vm = 5;

vc = 10;

fm = 1;

fc = 10;

fs = 100;

wm = 2*pi*fm;

wc = 2*pi*fc;

t = -2:1/fs:2;

n = length(t);

df = fs / n;

f = -fs/2:df:fs/2 - df;

% Modulation

m = vm*cos(wm*t);

c = vc*cos(wc*t);

s = m .* c;

ftm = fftshift(fft(m));

ftc = fftshift(fft(c));

fts = fftshift(fft(s));


subplot(3, 1, 1);

plot(t, m);

xlabel('Time (t)');

ylabel('m(t)');
```

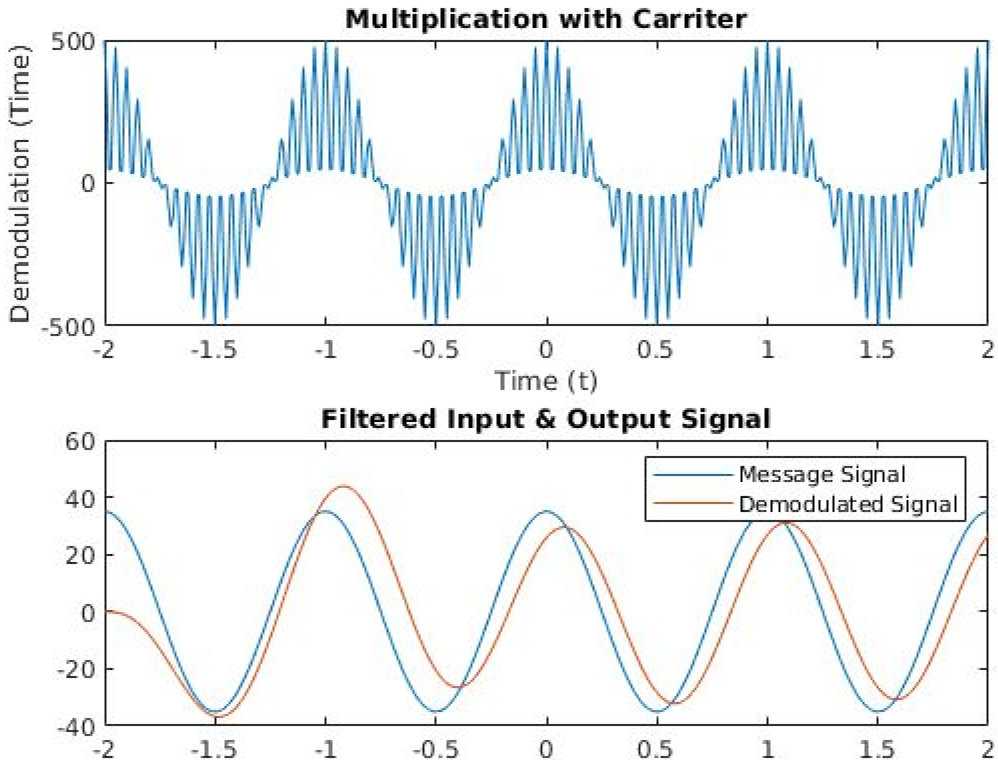
```
title('Message Signal');

subplot(3, 1, 2);
plot(t, c);
xlabel('Time (t)');
ylabel('c(t)');
title('Carrier Signal');

subplot(3, 1, 3);
plot(t, s);
xlabel('Time (t)');
ylabel('s(t)');
title('AM Signal');

figure;
subplot(3, 1, 1);
plot(f, abs(ftm));
xlabel('Frequency (f)');
ylabel('M(f)');
title('Message Signal in Frequency Domain');

subplot(3, 1, 2);
plot(f, abs(ftc));
xlabel('Frequency (f)');
ylabel('C(f)');
title('Carrier Signal in Frequency Domain');
```

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

plot(f, abs(fts));

xlabel('Frequency (f)');

ylabel('S(f)');

title('AM Signal in Frequency Domain');


% Demodulation

demt = s .* c;

demf = fftshift(fft(demt));


[B, A] = butter(3, fm/fs, 'low');

yt = filter(B, A, demt);

yf = fftshift(fft(yt));


figure;

subplot(2, 1, 1);

plot(t, demt);

xlabel('Time (t)');

ylabel('Demodulation (Time)');

title('Multiplication with Carrtier');



subplot(2, 1, 2);

plot(t, 7*m, t, -1*yt);

legend('Message Signal', 'Demodulated Signal');

title('Filtered Input & Output Signal');
```

Output Waveforms:







Conclusion:

In this experiment, we have implemented Amplitude Modulation and Demodulation in MATLAB. We also analyzed the spectrum of the modulated waves in frequency domain.

Remarks: **Signature:**

References:

- NPTEL communication systems lectures
  https://www.youtube.com/watch?v=S8Jod9AtpN4&list=PL7748E9BEC4ED83CA&index=8
  https://www.youtube.com/watch?v=NTcDup0_B4w&list=PL7748E9BEC4ED83CA&index=7
- Modern Analog and Digital Communication by B.P. Lathi (3rd or 4th edition)
- Communication Systems by Simon Haykin (4th edition)

DELTA MODULATION DEMODULATION

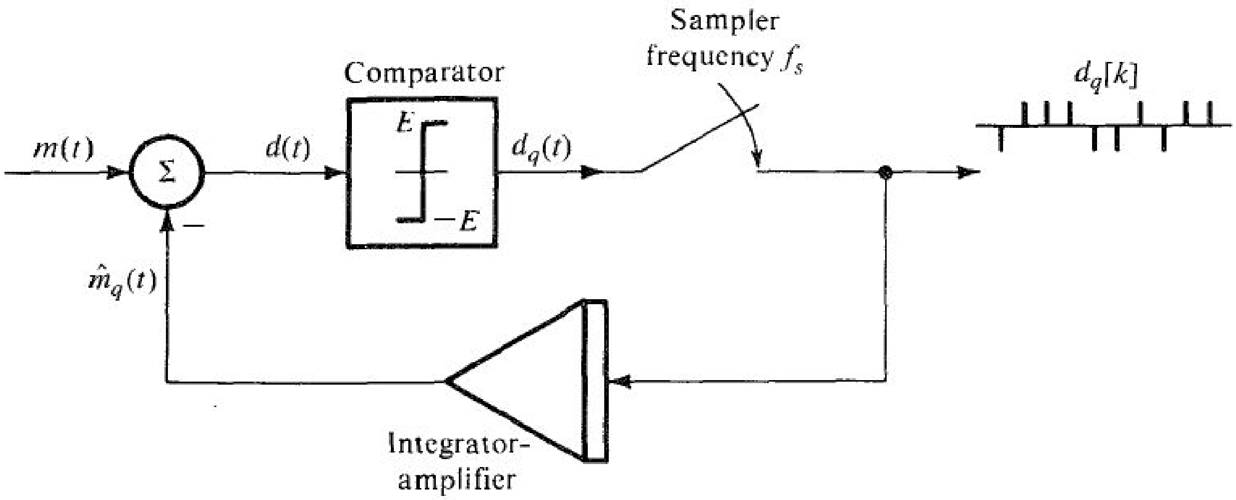
Experiment No: **9** **Date:**

Aim: **Write a MATLAB code to modulate and demodulate the given signal by Delta Modulation Technique.**

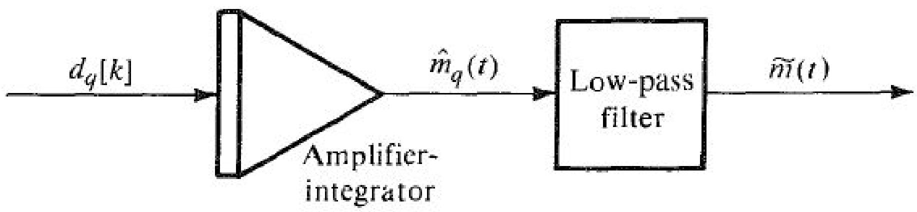
Brief Theory/Equations:

A delta modulation (DM or Δ-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:

- The analog signal is approximated with a series of segments.
- Each segment of the approximated signal is compared to the preceding bits and the successive bits are determined by this comparison.
- 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.
- To achieve high signal-to-noise ratio, delta modulation must use oversampling techniques, that is, the analog signal is sampled at a rate several times higher than the Nyquist rate.



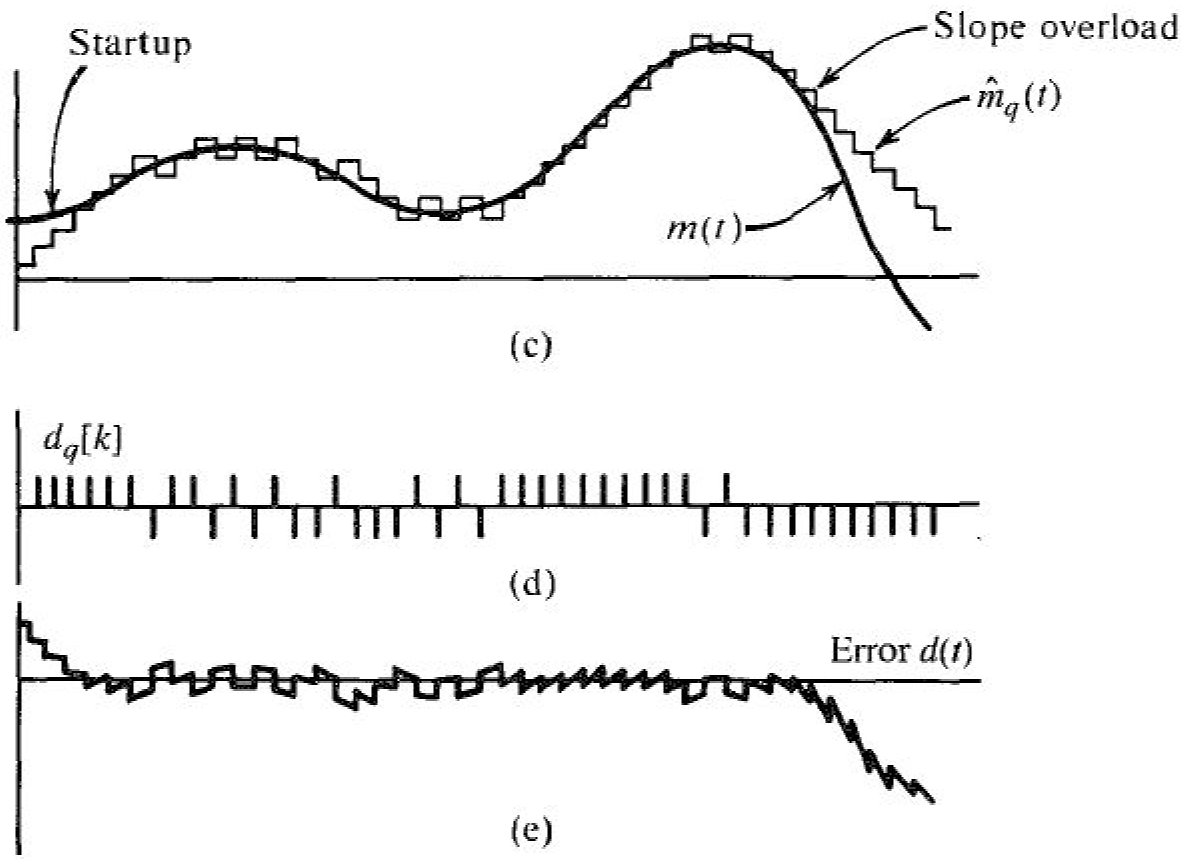
(a) Delta modulator



(b) Delta demodulator

The modulator consists of comparator and sampler in the direct path and an integrator amplifier in the feedback path. The analog signal $m(t)$ is compared with the feedback signal $\hat{m}_q(t)$. The error signal $d(t) = m(t) - \hat{m}_q(t)$ is applied to the comparator. If $d(t)$ is positive, the comparator output is of constant amplitude E and if $d(t)$ is negative, the comparator output is of constant value –E. Thus, the difference is the binary signal that is needed to generate 1 bit DPCM (Differential Pulse Code Modulation). The comparator output is sampled by a sampler at sampling frequency f_s, samples per second where f_s is much higher than the nyquist sampling rate. The sampler thus produces train of narrow pulses $d_q(k)$ (to simulate impulses) with a positive pulse when $m(t) > \hat{m}_q(t)$ and negative pulse when $m(t) < \hat{m}_q(t)$. Each sample is coded by a single binary pulse. The pulse train $d_q(k)$ is delta modulated pulse train. The modulated signal $d_q(k)$ is amplified and integrated in the feedback path to generate $\hat{m}_q(t)$, which tries to follow $m(t)$.

Each pulse in $d_q(k)$ at the input of the integrator gives rise to a step function (positive or negative depending on the polarity of pulse) in $\hat{m}_q(t)$. If for example, $m(t) > \hat{m}_q(t)$, a positive pulse is generated in $d_q(k)$, which gives rise to a positive step in $\hat{m}_q(t)$ trying to equalize $\hat{m}_q(t)$ to $m(t)$ in small steps at every sampling instant as shown in figure below. It can be seen that $\hat{m}_q(t)$ is kind of staircase approximation of $m(t)$. When $\hat{m}_q(t)$ is passed through a low pass filter, the coarseness of staircase in $\hat{m}_q(t)$ is eliminated and we get smoother and better approximation of $m(t)$. The demodulator at the receiver consists of an amplifier-integrator (identical to that of feedback path of modulator) followed by a low pass filter.



In DM, the modulated signal carries information not about the signal samples but about the difference between successive samples. If the difference is positive or negative, a positive or negative pulse is generated in the modulated signal $d_q(k)$. Basically, therefore, DM carries the information about the derivative of $m(t)$, hence the name delta modulation. This can also be seen from the fact that integration of delta modulated signal yields $\hat{m}_q(t)$, which is an approximation of $m(t)$. The information of difference between successive samples is transmitted by a 1 bit code word.

Threshold of Coding and Overloading

Threshold and overloading effects can be seen in the figure c. Variations in $m(t)$, smaller than the step value (threshold of coding) are lost in DM. Moreover, If $m(t)$ is too fast, derivative of it $\dot{m}(t)$ is too high, $\hat{m}_q(t)$ cannot follow $m(t)$ and overloading occurs. This is known as slope overloading, which gives rise to the slope overload noise. This noise is one of the basic limiting factors in the performance of DM. We should expect slope overload rather than amplitude overload in DM, because DM basically carries the information about $m(t)$. The granular nature of the output signal gives rise to the granular noise similar to the quantization noise. The slope overload noise can be reduced by increasing E (the step size). This unfortunately increases the granular noise. There is an optimum value of E, which yields the best compromise giving the minimum overall noise. This optimum value of E depends on the sampling frequency f_s and the nature of the signal.

The slope overload occurs when $\hat{m}_q(t)$ cannot follow $m(t)$. During the sampling interval T_s, $\hat{m}_q(t)$ is capable of changing by σ, where σ is the height of the step (amplitude). Hence, the maximum slope that $m(t)$ can follow is $\frac{\sigma}{T_s}$, or σf_s, where f_s, is the sampling frequency. Hence, no overload occurs if

$$|\dot{m}(t)| \geq \sigma f_s$$

Consider the case of tone modulation (meaning a sinusoidal message):

$$m(t) = A \cos cos\ \omega t$$

The condition for no overload is

$$|\dot{m}(t)|_{max} = \omega A < \sigma f_s$$

Hence, the maximum amplitude A_{max} of this signal that can be tolerated without overload is given by

$$A_{max} = \frac{\sigma f_s}{\omega}$$

The overload amplitude of the modulating signal is inversely proportional to the frequency ω. For higher modulating frequencies, the overload occurs for smaller amplitudes. For voice signals, which contain all frequency components up to (say) 4 kHz, calculating A_{max} by using $\omega = 2\pi \times 4000$ in above equation will give an overly conservative value. It has been shown that A_{max} for voice signals can be calculated by using $\omega_r \approx 2\pi \times 800$

$$[A_{max}]_{voice} \sim \frac{\sigma f_s}{\omega_r}$$

Thus, the maximum voice signal amplitude A_{max} that can be used without causing slope overload in DM is the same as the maximum amplitude of a sinusoidal signal of reference frequency f_r that can be used without causing slope overload in the same system

Fortunately, the voice spectrum (as well as the television video signal) also decays with frequency and closely follows the overload characteristics. For this reason, DM is well suited for voice (and television) signals. Actually, the voice signal spectrum (curve b) decreases as $\frac{1}{\omega}$ up to 2000 Hz, and beyond this frequency, it decreases as $\frac{1}{\omega^2}$. If we had used a double integration in the feedback circuit instead of a single integration, A_{max} would be proportional to $\frac{1}{\omega^2}$. Hence, a better match between the voice spectrum and the overload characteristics is achieved by using a single integration up to 2000 Hz and a double integration beyond 2000 Hz. Such a circuit (the double integration) responds fast but has a tendency to instability, which can be reduced by using some low-order prediction along with double integration. A double integrator can be built by placing in cascade two low-pass RC integrators with time constants R1 C1 = l /200pi and R2C2 = 1/4000pi, respectively. This results in single integration from 100 to 2000 Hz and double integration beyond 2000 Hz.

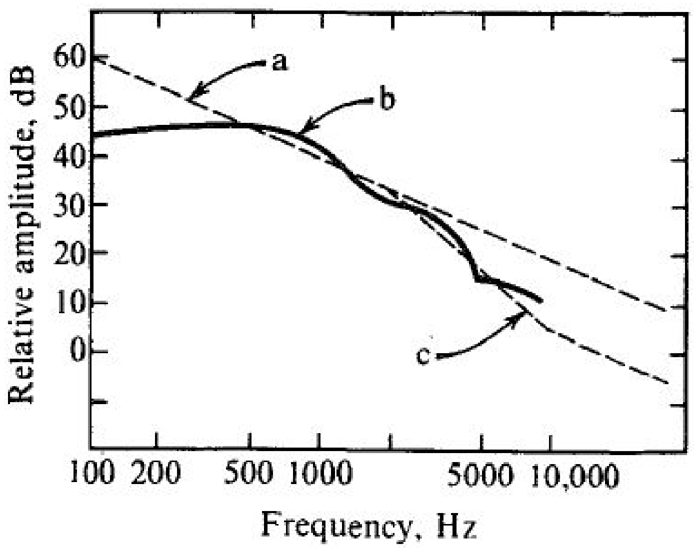


Figure 6.21 Voice signal spectrum.

Delta Modulation : A special case of DPCM

Sample correlation used in DPCM is further exploited in ***delta modulation (DM)*** by oversampling (typically four times the Nyquist rate) the baseband signal. This increases the correlation between adjacent samples, which results in a small prediction error that can be encoded using only one bit (L = 2). Thus, DM is basically a 1-bit DPCM, that is, a DPCM that uses only two levels (L = 2) for quantization of $m[k]-m_q[k]$. In comparison to PCM (and DPCM), it is a very simple and inexpensive method of *A/D* conversion. A 1-bit codeword in DM makes word framing unnecessary at the transmitter and the receiver. This strategy allows us to use fewer bits per sample for encoding a baseband signal.

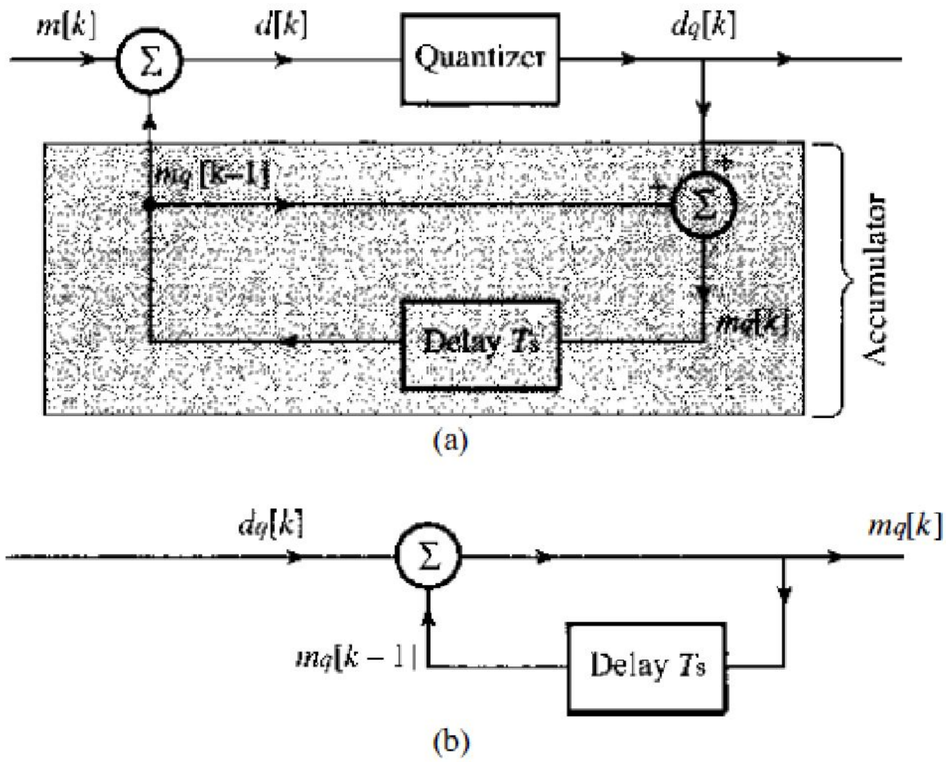
In DM, we use a first-order predictor, which, as seen earlier, is just a time delay of T_s, (the sampling interval). Thus, the DM transmitter (modulator) and receiver (demodulator) are identical to those of the DPCM in Fig. below, with a time delay for the predictor, as shown in Fig, from which we can write

$$m_q[k] = m_q[k-1] + d_q[k]$$

Hence,

$$m_q[k-1] = m_q[k-2] + d_q[k-1]$$

Figure 6.30
Delta modulation is a special case of DPCM.



Or we can write,

$$m_q[k] = m_q[k-2] + d_q[k-1] + d_q[k]$$

Proceeding in this manner, assuming zero initial condition, i.e. $m_q[0]=0$, we write

$$m_q[k] = \sum_{m=0}^{k} d_q[m]$$

This shows that the receiver (demodulator) is just an accumulator (adder). If the output $d_q[k]$ is represented by impulses, then the accumulator (receiver) may be realized by an integrator because its output is the sum of the strengths of the input impulses (sum of the areas underthe impulses). We may also replace with an integrator the feedback portion of the modulator (which is identical to the demodulator). The demodulator output is $m_q[k]$, which when passed through a low-pass filter yields the desired signal reconstructed from the quantized samples.

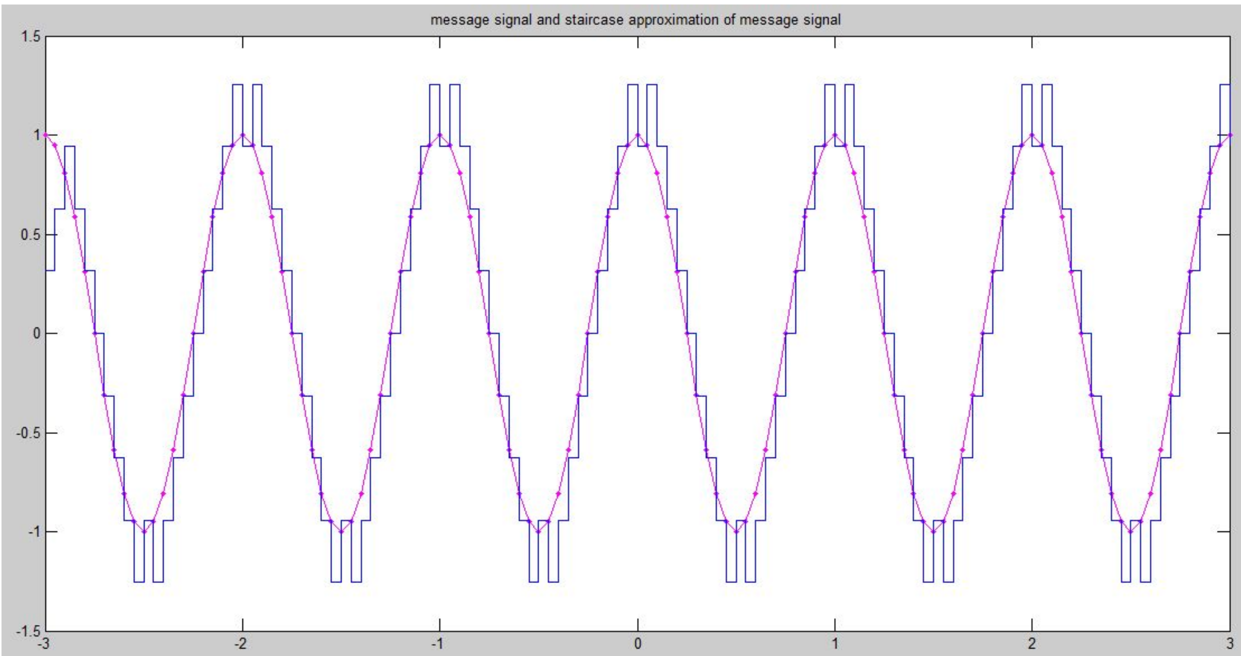
Algorithm:

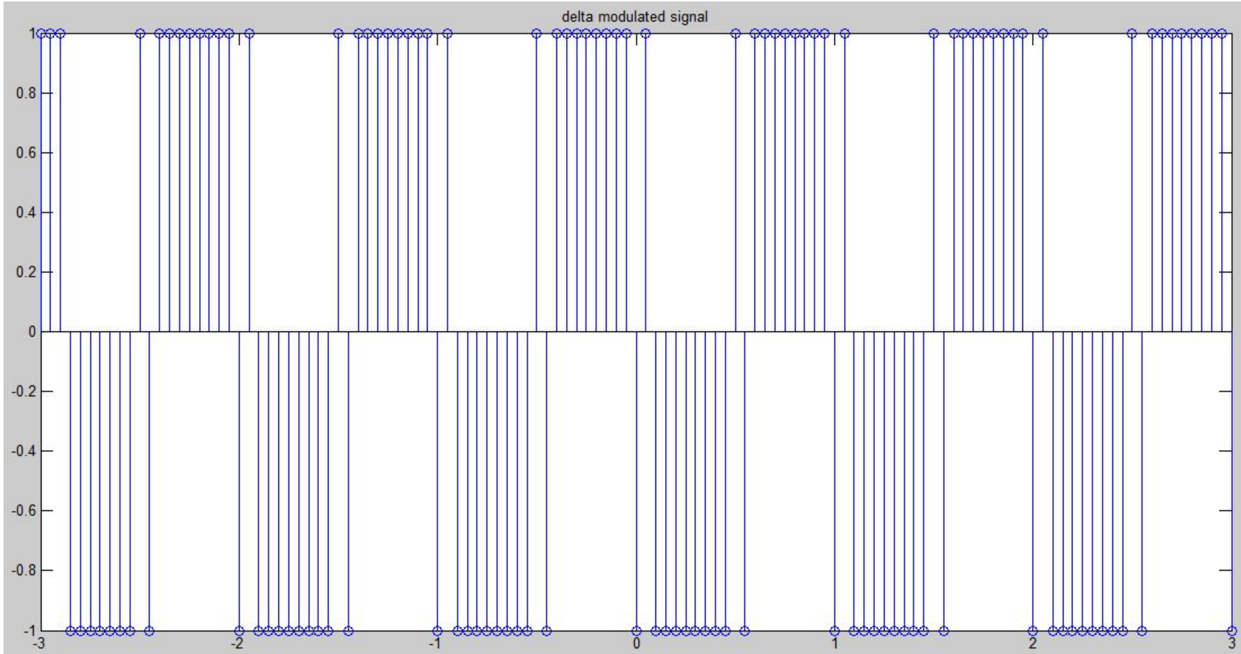
- Impement the block diagram of DM as a special case of DPCM (Figure 6.30, page 295, "Modern Analog and Digital Communication" by B.P. Lathi 4th edition)
- Consider the input/message signal as sinusoidal `m=Am*cos(2*pi*fm*t)`, with parameters, Am=1V, fm=1Hz.
- Define the time range with sampling frequency fs=20*fm (oversampling), hence, t can be defined as `t=-3:1/fs:3;`
- Define the step size del for the delta modulator which should satisfy the condition

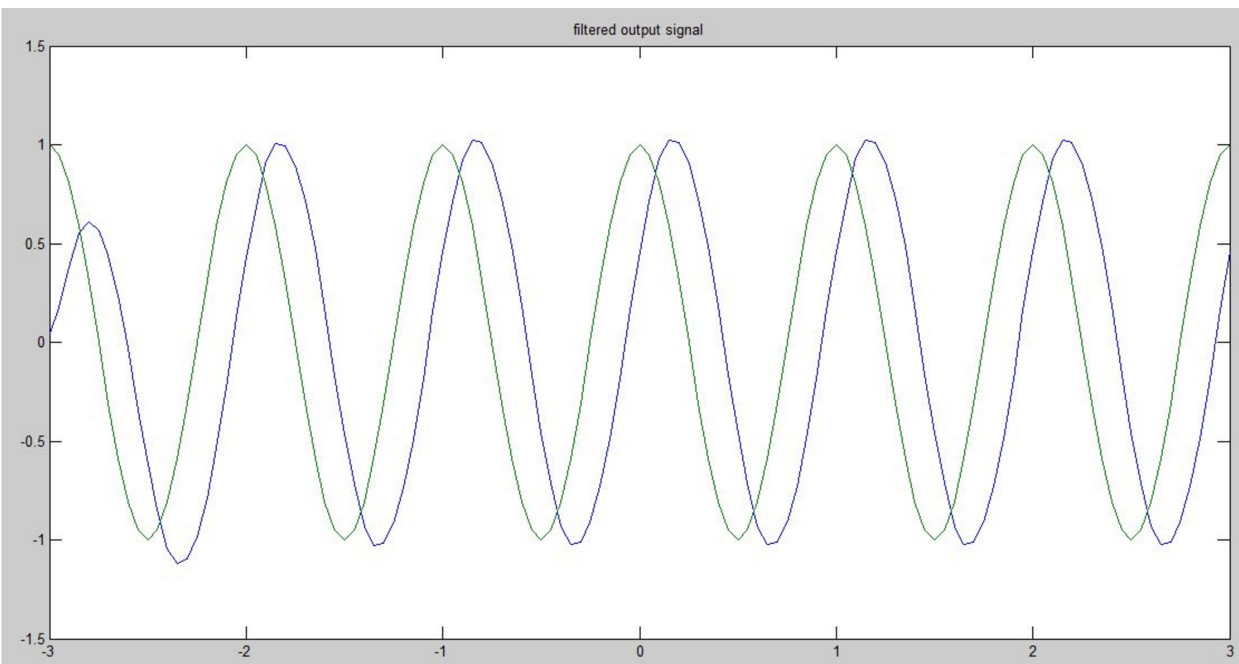
$$\Delta \leq \frac{2\pi A_m f_m}{f_s}$$

  Hence, `del=(2*pi*fm*Am)/fs;`
- Choose the index i=1:length(t) ; length(t)=max no. of columns in t.
    - If i=1, then mq=0.
        - So, the difference signal `d(i)=m(i);`
        - Use the sign function of MATLAB to determine whether d is +ve or –ve
        - Determine the approximate difference value dq by applying hard limiting operation i.e. by multiplying sign(d) with del.
        - Approximated message signal mq=dq, for i=1.
    - Else
        - The difference signal, `d(i)=m(i)-mq(i-1);`
        - The approximated difference operation will be same as in case for i=1.
        - Approximated message signal (staircase approximation), `mq(i)=dq(i)+mq(i-1);`
- Figure1: plot the message signal and staircase approximation signal the same window. Use the command hold on. And use command stairs(t,mq) for approximated message signal.
- Figure2: plot the delta modulated signal, consider the modulated output x to be +1 if dq>0 else x will be -1.
    - Use stem command for plotting as it is a discrete time signal
- For demodulation, pass the approximated message signal via low pass filter. (See FM demodulation algorithm for filtering logic).
- Figure 3: plot filtered output signal and original input signal in the same window.
    - `plot(t,2*y,t,m);`

Expected Ouput Waveforms:




delta modulated signal
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-3
-2
-1
0
1
2
3

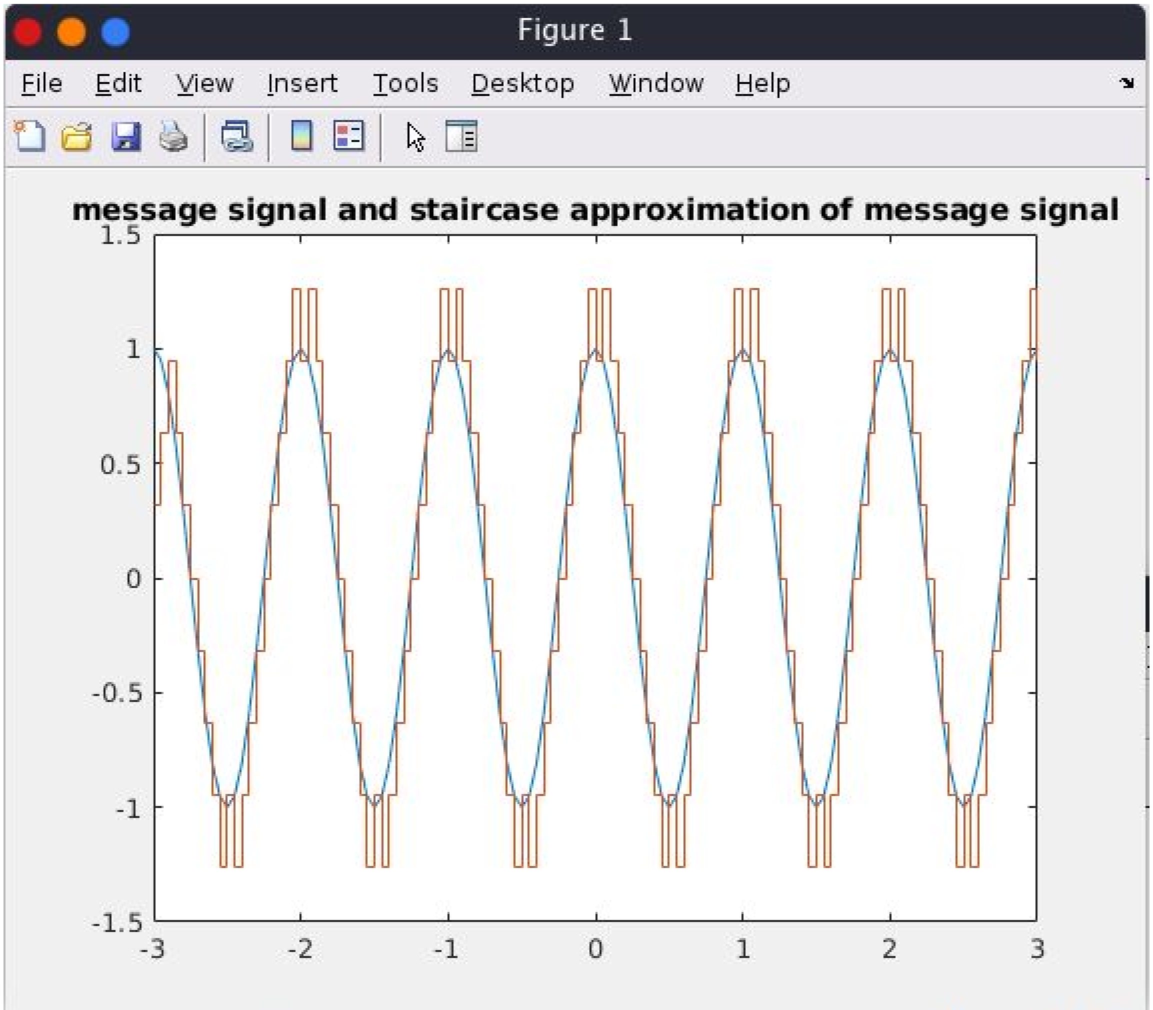

filtered output signal
1.5
1
0.5
0
-0.5
-1
-1.5
-3
-2
-1
0
1
2
3

Code:

```
clc

clear all
fm=1;%defining fm
fs=20*fm;%defining fs
t=-3:(1/fs):3;%defining t
am=1;%defining am
m=am*cos(2*pi*fm*t);%the message signal
del=(2*pi*fm*am)/fs;%defining del
%Running a for loop
for i=1:length(t)
    if(i==1)
        mq=0;
        d(i)=m(i);
        dq=(sign(d))*del;
        mq=dq;
    else
        d(i)=m(i)-mq(i-1);
        dq=(sign(d))*del;
        mq(i)=dq(i)+mq(i-1);
    end
end
%plotting
plot(t,m);
%holding
hold on;
stairs(t,mq);
title('message signal and staircase approximation of message signal');%title
of plot
%running for loop
for n=1:length(t)
if(d(n)>0)
    dm(n)=1;
else
    dm(n)=-1;
end
end
%another figure
figure
stem(t,dm);
title('Delta modulated signal');%title of plot
n=1;
wn=fm/fs;%defining wn
[b a]=butter(n,wn,'low');
y=filter(b,a,mq);%y is output of filter
%plotting again
```

```
figure
plot(t,2*y,t,m);
title('filtered output signal');%title of the plot
```

<u>Output Waveforms:</u>



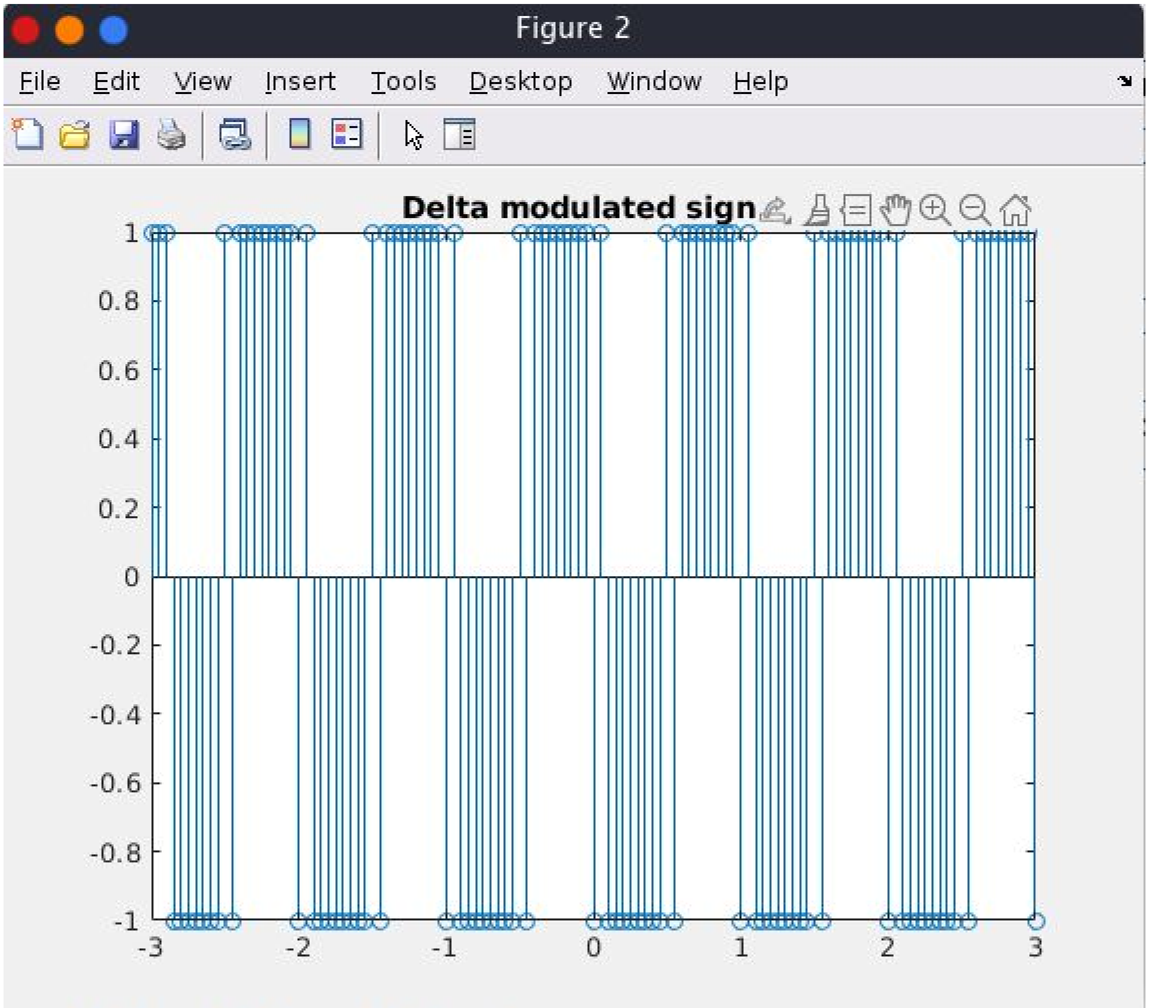
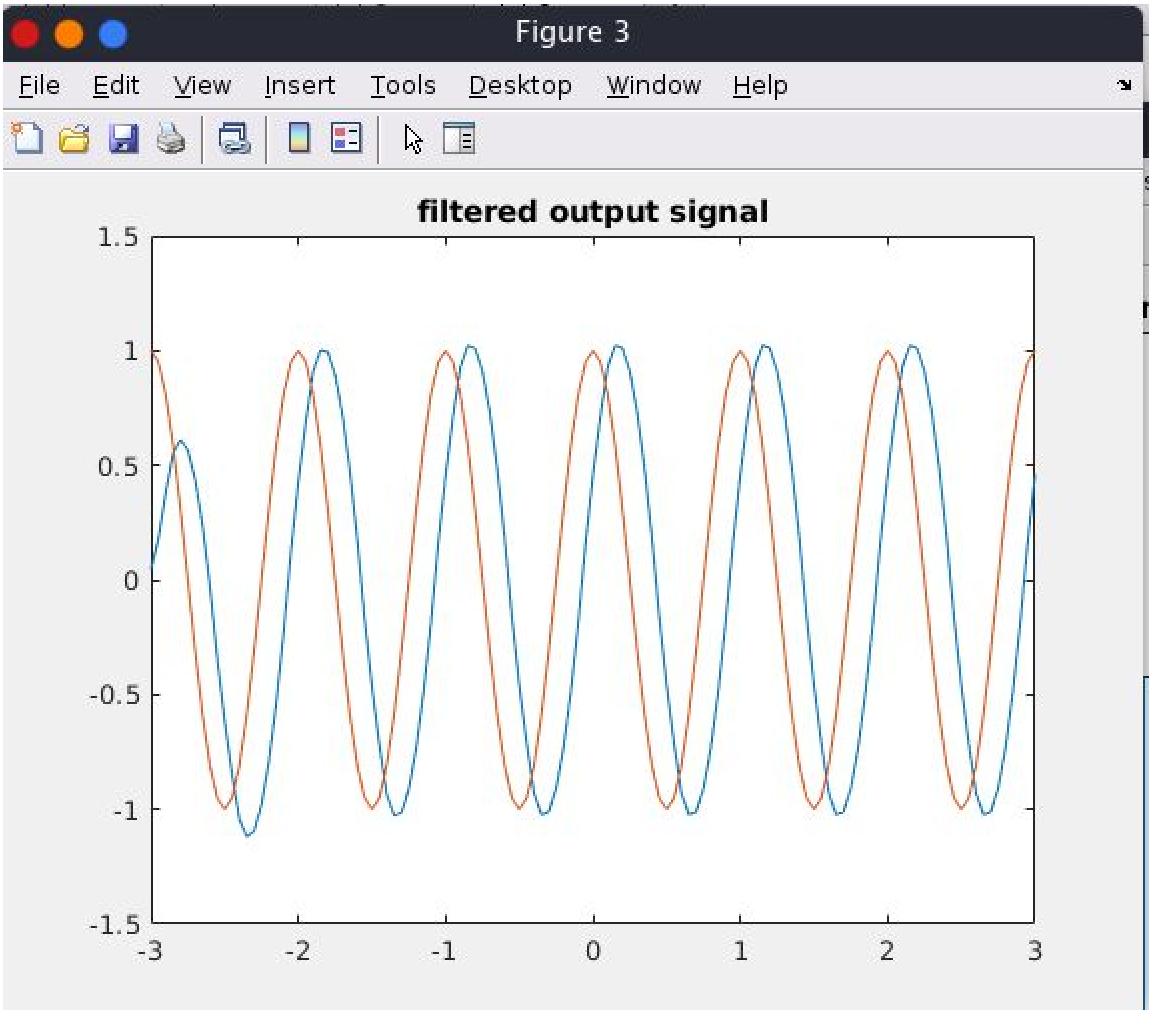


Figure 2
File
Edit
View
Insert
Tools
Desktop
Window
Help
Delta modulated sign
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-3
-2
-1
0
1
2
3



Conclusion:

In this experiment we performed Delta modulation and demodulation of message signal using ***MATLAB*** and plotted the modulated and filtered signals.

Remarks: **Signature**

References:

- NPTEL digital communication systems lectures
  http://www.digimat.in/nptel/courses/video/108102096/L02.html
- Modern Analog and Digital Communication by B.P. Lathi (3^{rd} or 4^{th} edition)
- Communication Systems by Simon Haykin (4^{th} edition).
- Delta modulation/demodulation in MATLAB
  https://www.youtube.com/watch?v=XHHrh-vyhcE

FREQUENCY MODULATION IN MATLAB

Experiment No: 10 **Date:**

Aim: To implement frequency modulation and demodulation using MATLAB.

Brief Theory/Equations:

When the angle of the carrier wave is varied in some manner with respect to modulating signal $m(t)$, the technique of modulation is known as angle modulation or exponential modulation.

General form, $s(t) = A \cos cos\,(\omega_c t + \varphi(t))$, where $s(t)$ is the modulated signal, ω_c is the carrier frequency in radians and $\varphi(t)$ is the phase function which is time varying and captures the information, which you want to convey.

So, modulating signal $m(t)$ somehow modifies this $\varphi(t)$

Let $\theta_i(t) = \omega_c t + \varphi(t)$ be the instantaneous phase of the carrier signal, then instantaneous frequency can be obtained by taking the derivative of instantaneous phase w.r.t. time, t.

$\omega_i(t) = \frac{d\theta_i(t)}{dt} = \omega_c + \frac{d\varphi(t)}{dt}$ is the instantaneous frequency of the carrier wave or modulated signal

Where, $\varphi(t)$ is the instantaneous phase deviation and $\frac{d\varphi(t)}{dt}$ is the instantaneous frequency deviation.

Phase Modulation: (PM)

The instantaneous phase deviation carries the information or $\varphi(t)$ is varied linearly with $m(t)$

$$\varphi(t) \quad \alpha \quad m(t)$$

$$\varphi(t) \quad = \quad k_p\, m(t)$$

Where, k_p is the phase modulation constant/ phase sensitivity of the modulator measured in radians/volt.

For PM, instantaneous phase $\theta_i(t) = \omega_c t + k_p\, m(t)$

And instantaneous frequency $\omega_i(t) = \omega_c + k_p \frac{dm(t)}{dt}$

Modulated signal for PM, $s(t) = A \cos cos\,(\omega_c t + k_p\, m(t))$

Frequency Modulation: (FM)

The instantaneous frequency carries the information or message signal.

$$\frac{d\varphi(t)}{dt} \; \alpha \; m(t)$$

$$\frac{d\varphi(t)}{dt} = k_f \; m(t)$$

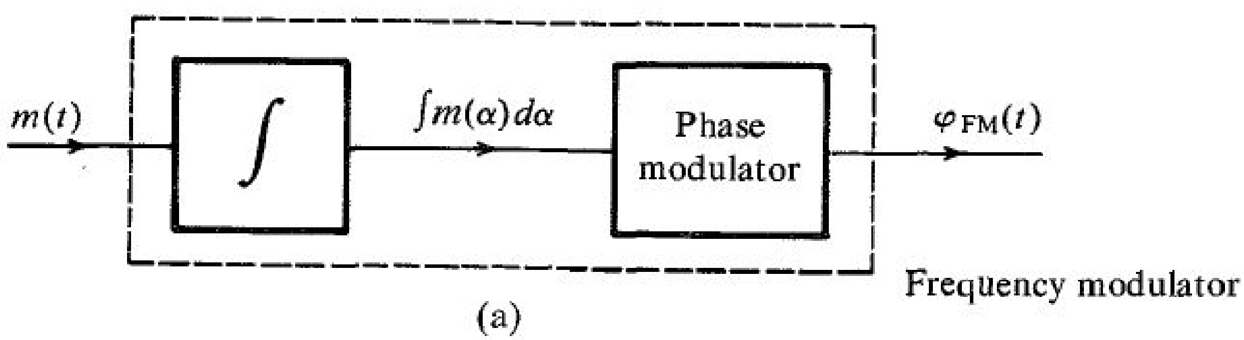
Where, k_f is the frequency modulation constant/ frequency sensitivity of the modulator measured in radians/(seconds*volt) or Hz/volt.

For FM, instantaneous phase deviation, $\varphi(t) = k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha$

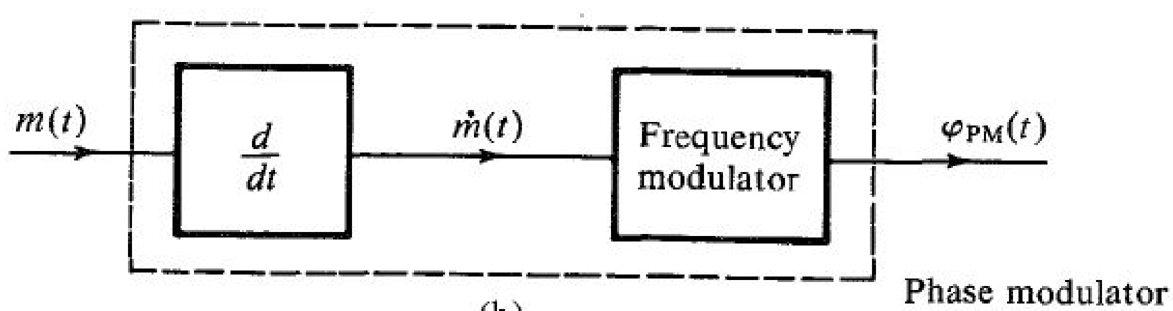
Hence, instantaneous phase, $\theta_i(t) = \omega_c t + k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha$

And instantaneous frequency, $\omega_i(t) = \omega_c + k_f \; m(t)$

Modulated Signal for FM, $s(t) = A \cos cos\, (\omega_c t + k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha)$



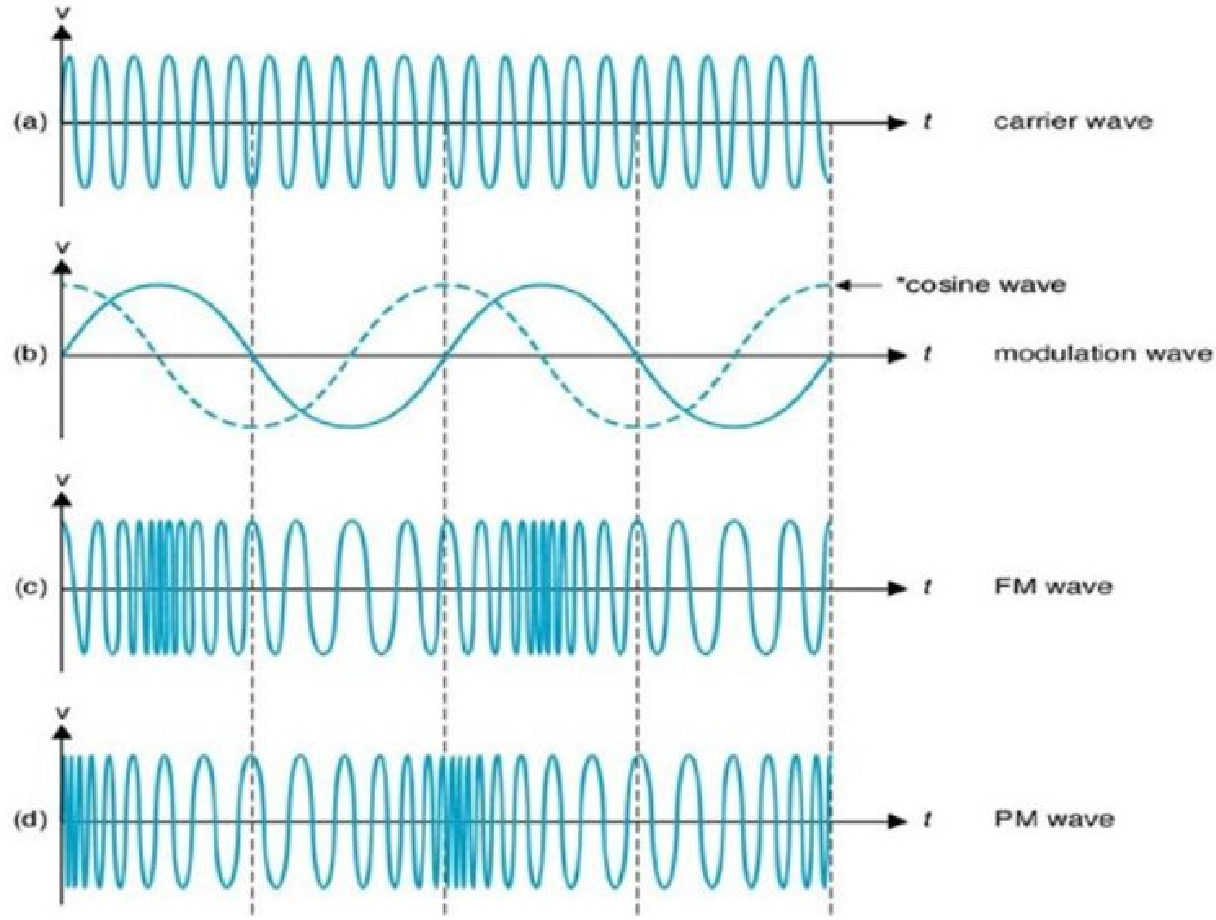
(a)



(b)

- Phase and frequency modulation are inseparable.

Considering, message signal as sine wave, the derivative of it will be cosine and respective FM and PM wave are shown in the next figure.



or else if we consider $m(t) = A_m \cos cos\ \omega_m t$, the instantaneous phase deviation for PM and FM will be;

$$\text{PM,} \quad \varphi(t) \quad = \quad k_p\, m(t) = k_p A_m \cos cos\ \omega_m t$$

FM,

$$\varphi(t) = k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha = k_f \int_{-\infty}^{t} A_m \cos cos\ \omega_m \alpha\, d\alpha = \frac{k_f A_m \ \sin sin\ \omega_m t}{\omega_m}$$

Hence, the modulated signal for PM and FM become

$$s(t) = A \cos cos\ (\omega_c t + k_p\, m(t)) = \ A \cos cos\ (\omega_c t + k_p\, A_m \cos cos\ \omega_m t\,) = \ A \cos cos\ \left(\omega_c t + \beta_{pm} \cos cos\ \omega_m t\,\right)$$

Where, $\beta_{pm} = \ k_p\, A_m$ is the modulation index for PM

$$s(t) = A \cos cos\, (\omega_c t + k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha) \ = A \cos cos\, \left(\omega_c t + \frac{k_f A_m \ \sin sin\, \omega_m t}{\omega_m}\right) = \ A \cos cos\, \left(\omega_c t + \beta_{fm} \sin sin\, \omega_m t\right)$$

Where, $\beta_{fm} = \frac{k_f A_m}{\omega_m} = \frac{\Delta f}{f_m}$ is the modulation index for FM

$\Delta f = k_f A_m$, is the peak frequency deviation

Frequency Deviation:

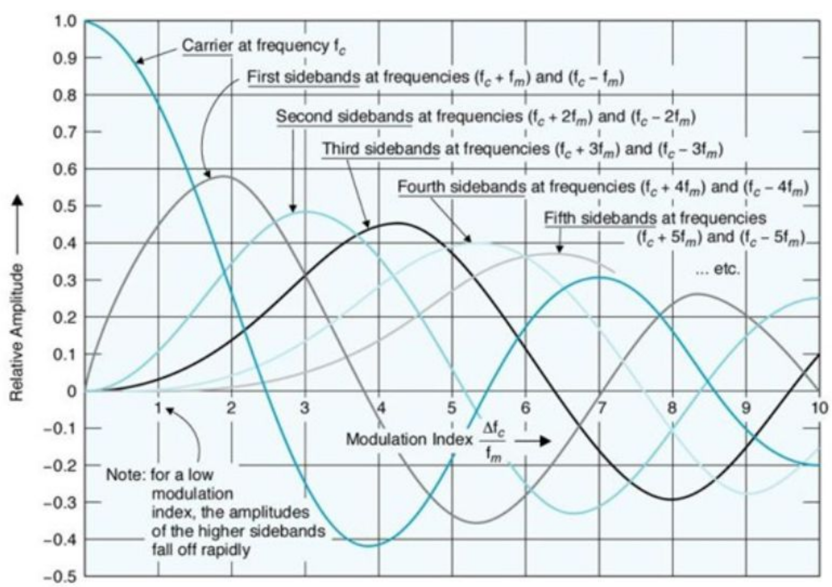
- The amount of change in the carrier frequency produced, by the amplitude of the input modulating signal, is called **frequency deviation**.
- The Carrier frequency swings between fmax and fmin as the input varies in its amplitude.
- The difference between fmax and fc is known as frequency deviation. Δf = fmax – fc
- Similarly, the difference between fc and fmin also is known as frequency deviation. Δf = fc –fmin

The frequency modulated signal can also be expressed in terms of Bessel function by taking the fourier series expansion.

Reference: Modern Digital and Analog Communication by B.P.Lathi

$$s(t)_{FM} \ = \ A \sum_{n=-\infty}^{\infty} J_n(\beta) \cos cos\, ((\omega_c + n\omega_m))\, t$$

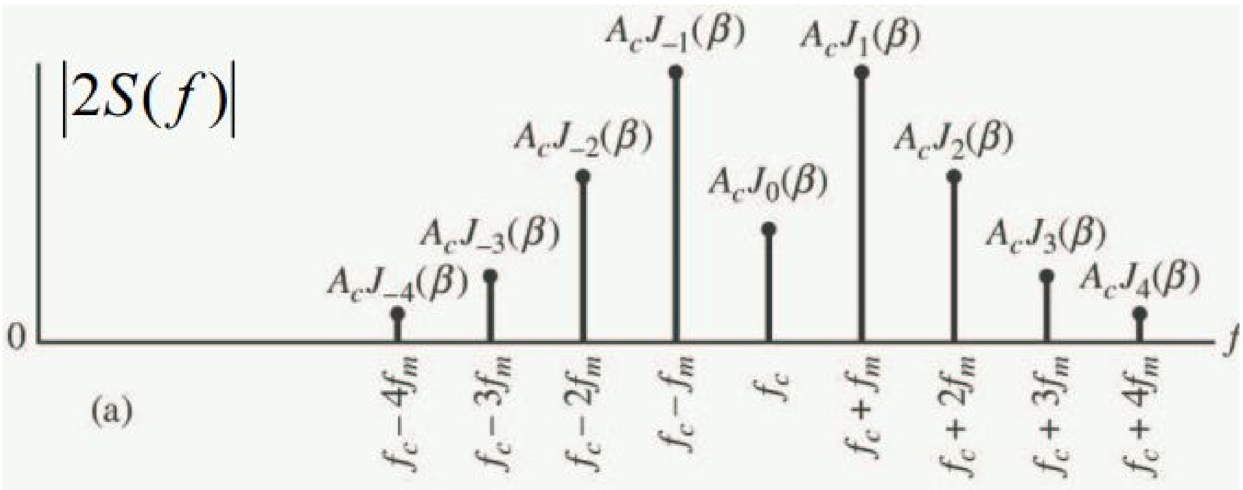
Where, $J_n(\beta)$ is Bessel function of n order with argument β.



By taking the fourier transform of the above equation, we get

$$S(f) = \frac{A}{2} \sum_{n=-\infty}^{\infty} J_n(\beta)\,[\delta(f-(f_c+nf_m)) + \delta(f+(f_c+nf_m))]$$

Theoretically, single tone FM has carrier component with infinite number of side-bands at $(f_c \pm nf_m)$ and its bandwidth is infinite.



Bandwidth of FM:

As the value of n increase, the significant power level in sideband component decreases

i.e. $lim_{n\to\infty} J_n(\beta) = 0$.

So we consider only k sidebands on either sides of f_c and look at these 2k+1 components i.e. from $(f_c + kf_m)$ to $(f_c - kf_m)$ and if we consider the power ratio,

$$\text{Power ratio} = \frac{\frac{1}{2}A_c^2 \sum_{k=-\infty}^{\infty} J_n(\beta)^2}{\frac{1}{2}A_c^2} = \frac{\text{sideband power}}{\text{carrier power}} = J_0(\beta)^2 + 2\sum_{n=1}^{k} J_n(\beta)^2$$

($since$, $J_0(\beta)$ is the carrier related component and $J_n(\beta)$ is an even function)

The bandwidth is defined as band of frequencies over which signal contains 98% of its power that means the power ratio is 0.98.

In general, $BW = 2kf_m$

In empirical sense, if we choose $k = \beta + 1$, then power ratio turns to 0.98. (depends on the Bessel function values)

Hence, $BW = 2(\beta+1)f_m = 2(\beta f_m + f_m) = 2(\Delta f + f_m)$ $[since\ \beta = \frac{\Delta f}{f_m}]$

Frequency Demodulation:

We know that the equation of FM wave is

$$s(t) = A_c \cos\left(2\pi f_c t + 2\pi k_f \int m(t)\,dt\right)$$

Differentiate the above equation with respect to '**t**'.

$$\frac{ds(t)}{dt} = -A_c\left(2\pi f_c + 2\pi k_f m(t)\right)\sin\left(2\pi f_c t + 2\pi k_f \int m(t)\,dt\right)$$

We can write,

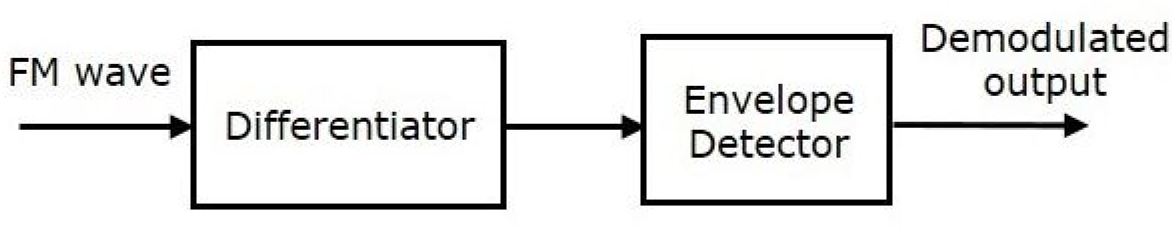
$-\sin\theta$ as $\sin(\theta - 180^0)$.

$$\Rightarrow \frac{ds(t)}{dt} = A_c\left(2\pi f_c + 2\pi k_f m(t)\right)\sin\left(2\pi f_c t + 2\pi k_f \int m(t)\,dt - 180^0\right)$$

$$\Rightarrow \frac{ds(t)}{dt} = A_c\left(2\pi f_c\right)\left[1 + \left(\frac{k_f}{k_c}\right)m(t)\right]\sin\left(2\pi f_c t + 2\pi k_f \int m(t)\,dt - 180^0\right)$$

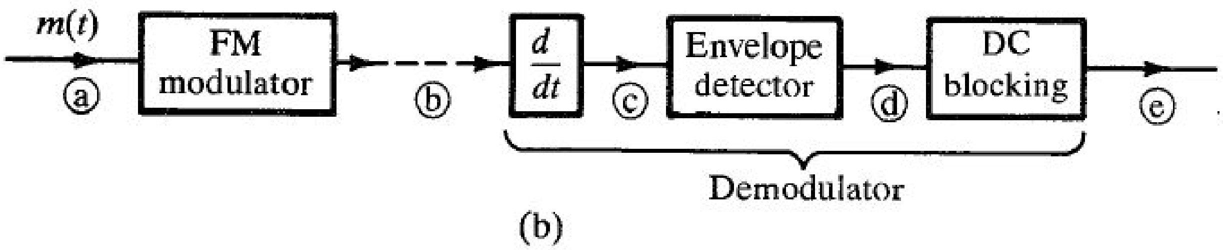
In the above equation, the amplitude term resembles the envelope of AM wave and the angle term resembles the angle of FM wave. Here, our requirement is the modulating signal m(t)m(t). Hence, we can recover it from the envelope of AM wave.

The following figure shows the block diagram of FM demodulator using frequency discrimination method.



This block diagram consists of the differentiator and the envelope detector. Differentiator is used to convert the FM wave into a combination of AM wave and FM wave. This means, it converts the frequency variations of FM wave into the corresponding voltage (amplitude) variations of AM wave. We know the operation of the envelope detector. It produces the demodulated output of AM wave, which is nothing but the modulating signal.

The FM communication link is shown below:



Algorithm:

- Define the sampling frequency $f_s > 2(f_c + k_{max}f_m)$ say, $f_s = 100Hz$;
- Define the time range using the sampling frequency t= -10 : $1/f_s$: 10
- Consider, message signal, $m = A_m \cos cos\ \omega_m t$, where $f_m = 1Hz$ and carrier signal $c = A_c \cos cos\ \omega_c t$ where $f_c = 10Hz$. Keep the amplitude of message and carrier signal same.
- Assume $k_p = 1$ and $k_f = 2\pi f_m$
- For PM, $s = A \cos cos\ (\omega_c t + k_p\ m(t)) = A \cos cos\ (\omega_c t + k_p\ A_m \cos cos\ \omega_m t\ )$
- For FM, $s(t) = A \cos cos\ (\omega_c t + k_f \int_{-\infty}^{t} m(\alpha)\, d\alpha) = A \cos cos\ \left(\omega_c t + \frac{k_f A_m\ \sin sin\ \omega_m t}{\omega_m}\right)$

  Use integral function in MATLAB in general so it can be done for any input signal.
- Figure1: Plot input signal, carrier signal, FM signal, derivative of input signal and PM signal using subplot(511) to subplot(515);
  Use diff function in matlab for the derivative of the signal
  For example, y=diff(x), pad the last value of the signal so as to ensure the same size because difference values will be one less than the given values.
  y=[diff(x) diff(end)]
- Figure 2: plot the frequency spectrum of FM and PM signals using the commands fft and fftshift.
  - Define n=length(t) %gives no. of columns in t
  - Define the step size for frequency axis fp which should be of same size as that of t. (matrix dimensions must match for plotting).
    - df=fs/n; where fs is the sampling frequency
  - Define frequency axis fp = -fs/2:df:fs/2-df ($\pm$df for getting same size matrices)
  - Take the fourier transform of FM and PM using fft command

- Y = fft(x) returns the discrete Fourier transform (DFT) of vector x, computed with a fast Fourier transform (FFT) algorithm.
- Y = fftshift(X) rearranges the outputs of fft by moving the zero-frequency component to the center of the array. It is useful for visualizing a Fourier transform with the zero-frequency component in the middle of the spectrum.
- Can write in a single syntax as y=fftshift(fft(x));
- Plot the frequency spectrum of FM and PM using the command plot(fp,y).Use subplot(211) to (212).

- **Demodulate the FM signal**
  - Take the derivative of FM signal using diff function of MATLAB as mentioned in one of the previous step.
  - Multiply the above signal with carrier signal, this will look like AM wave.
    - Do .* element wise multiplication
- Do the low pass filtering of the above signal using butter and filter command.
    - [b,a] = butter(n,Wn,'ftype'). **This creates a filter**
    - [b,a] = butter(n,Wn) designs an order n lowpass digital Butterworth filter with normalized cutoff frequency Wn. It returns the filter coefficients in length n+1 row vectors b and a, with coefficients in descending powers of z.

$$H(z) = \frac{b(1) + b(2)z^{-1} + \ldots + b(n+1)z^{-n}}{1 + a(2)z^{-1} + \ldots + a(n+1)z^{-n}}$$

    - where the string 'ftype' is one of the following:
      - 'high' for a highpass digital filter with normalized cutoff frequency Wn
      - 'low' for a lowpass digital filter with normalized cutoff frequency Wn
      - 'stop' for an order 2*n bandstop digital filter if Wn is a two-element vector, Wn = [w1 w2]. The stopband is w1 < ω < w2.
      - 'bandpass' for an order 2*n bandpass filter if Wn is a two-element vector, Wn = [w1 w2]. The passband is w1 < ω < w2. Specifying a two-element vector, Wn, without an explicit 'ftype' defaults to a bandpass filter.
      - Cutoff frequency, Wn is that frequency where the magnitude response of the filter is . For butter, the normalized cutoff frequency Wn must be a number between 0 and 1, where 1 corresponds to the Nyquist frequency, π radians per sample.
      - Choose order, n=4 ; Wn=fm/fs; ftype='low'
- Use the filter command to apply the filter
  - y = filter(b,a,X), where b and a are the coefficients obtained in the previous step and X will the multiplied signal output (derivative of FM signal*Carrier Signal). y is the filtered output. Adjust order so that you get approximate input signal.
- Figure 3: Plots for FM demodulation
  - Include three figures using subplot(311) to subplot(313)

- 1^{st} – derivative of FM signal
- 2^{nd} - multiplied signal output (derivative of FM signal*Carrier Signal)
- 3^{rd} – filtered output y. Include message signal as well in this
  - Example: plot(t,m,t,y)

Expected Output waveforms:

Figure 1:

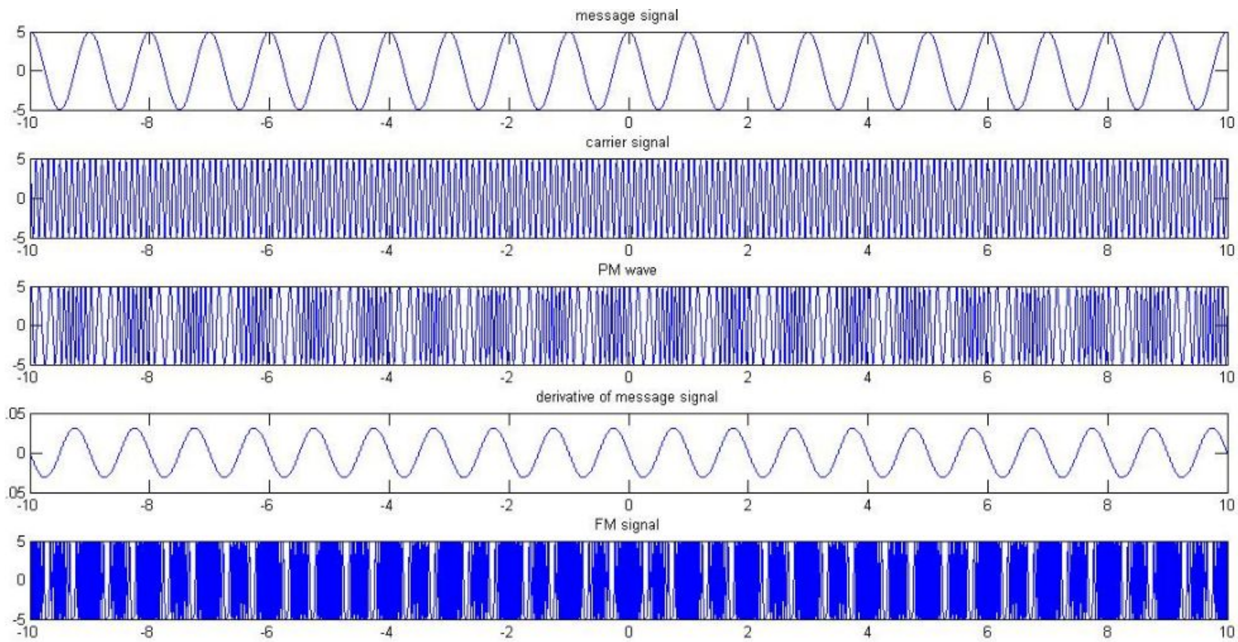


Figure 2:

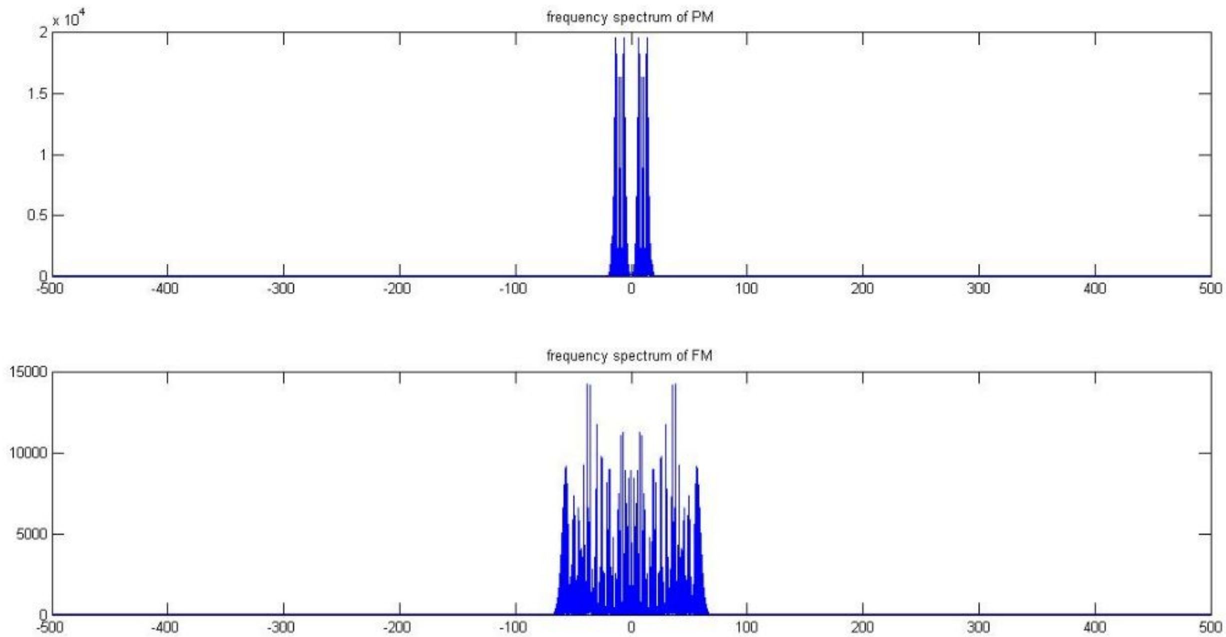
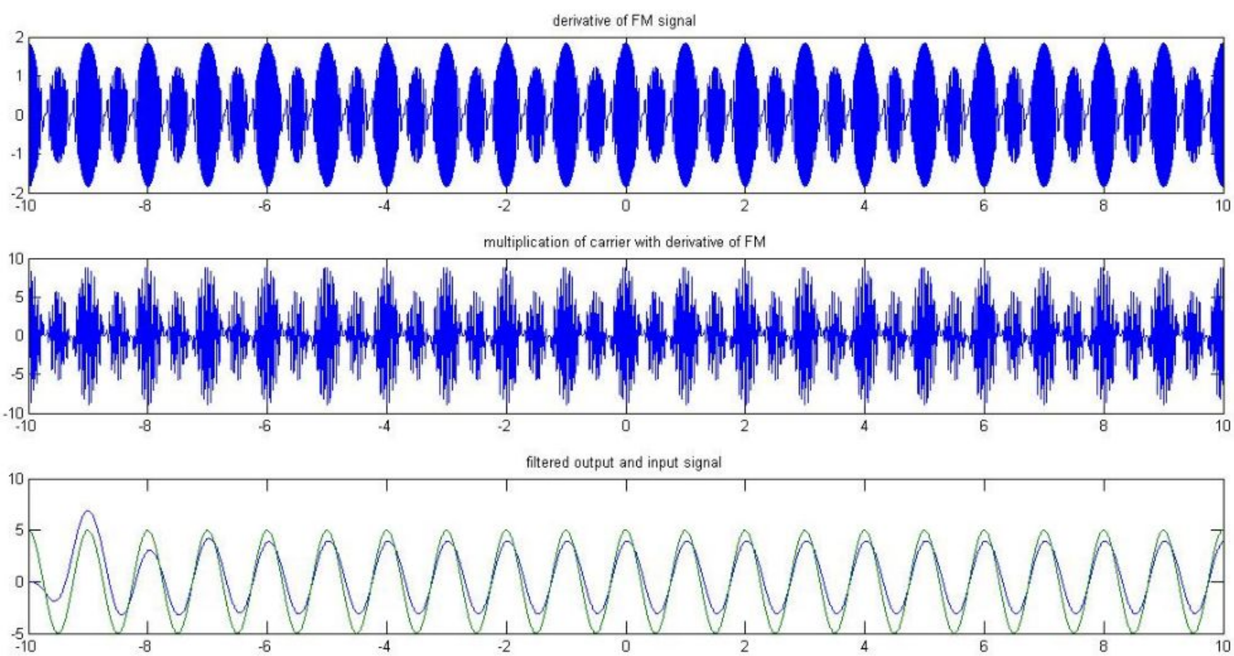

frequency spectrum of PM
x 10^4
2
1.5
1
0.5
0
-500
-400
-300
-200
-100
0
100
200
300
400
500
frequency spectrum of FM
15000
10000
5000
0

Figure 3:

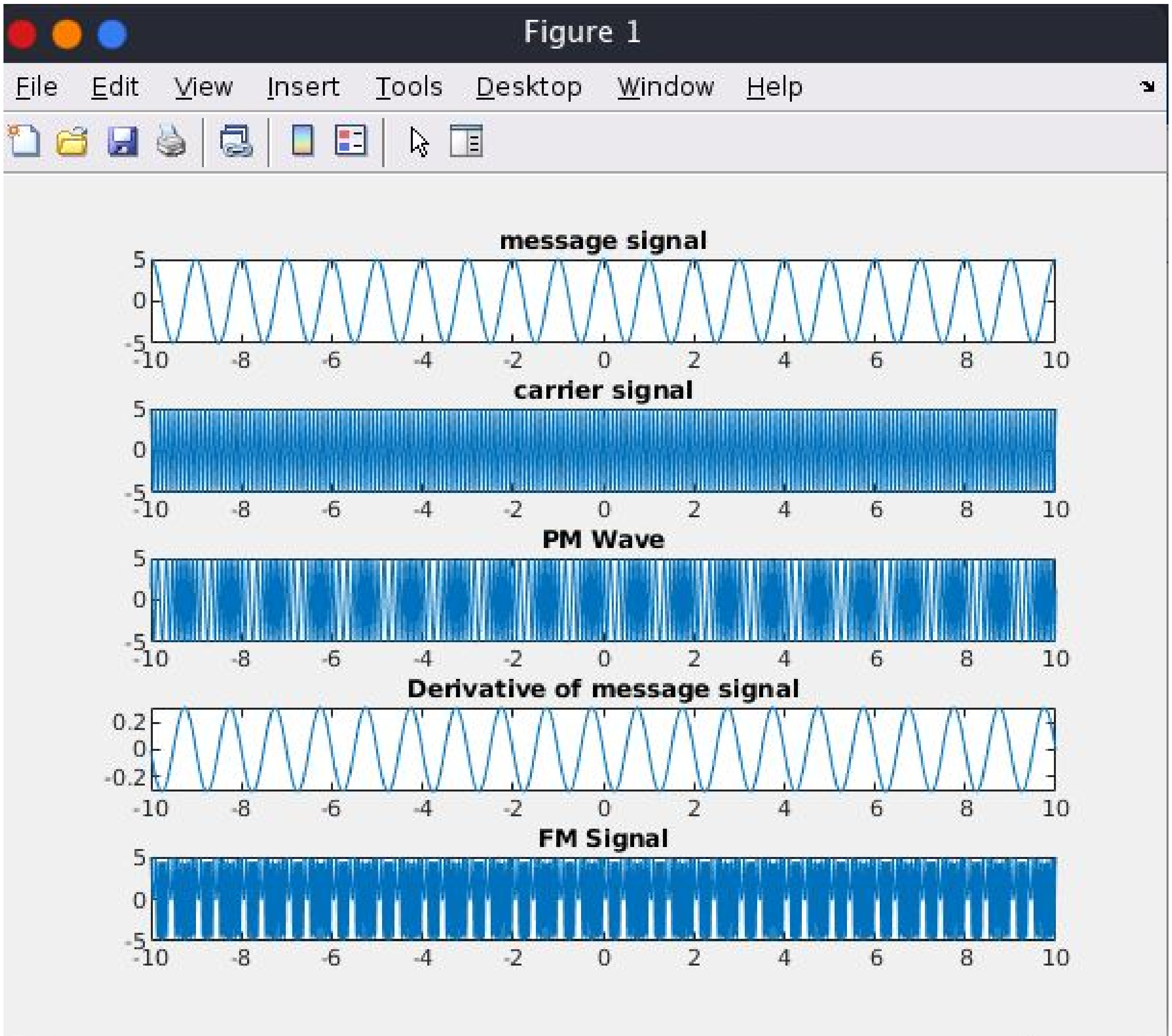

derivative of FM signal
2
1
0
-1
-2
-10
-8
-6
-4
-2
0
2
4
6
8
10
multiplication of carrier with derivative of FM
10
5
0
-5
-10
filtered output and input signal
10
5
0
-5

Code:

```
clc
clear all
fs=100;
t=-10:(1/fs):10;
fm=1;
am=5;
ac=5;
wm=2*pi*fm;
fc=10;
wc=2*pi*fc;
m=am*cos(wm*t);
c=ac*cos(wc*t);
kp=1;
kf=2*pi*fm;

F = griddedInterpolant(t,m);
fun = @(l) F(l);

s1=ac*cos(wc*t+(kp*m));
s2=ac*cos((wc*t)+(kf*fun(t)));
s3=[diff(m) 0];


subplot(5,1,1);
plot(t,m);
title('message signal');
subplot(5,1,2);
plot(t,c);
title('carrier signal');
subplot(5,1,3);
plot(t,s1);
title('PM Wave');
subplot(5,1,4);
plot(t,s3);
title('Derivative of message signal');
subplot(5,1,5);
plot(t,s2);
title('FM Signal');
```

```
n=length(t);
disp(n);
df=fs/n;
fp=(-fs/2):df:(fs/2)-df;
y1=fftshift(fft(s1));
y2=fftshift(fft(s2));
figure;
subplot(2,1,1);
plot(fp,y1);
title('Frequency Spectrum of PM');
subplot(2,1,2);
plot(fp,y2);
title('Frequency Spectrum of FM');

dfm=[diff(s2) 0];

s4=dfm.*c;
[b,a] = butter(4,fm/fs,'low');
y=filter(b, a, s4);

figure;
subplot(3,1,1);
plot(t,dfm);
title('Derivative of FM Signal');
subplot(3,1,2);
plot(t,s4);
title('Multiplication of carrier with derivative of FM Signal');
subplot(3,1,3);
plot(t,m, t, y);
title('Filtered output and input signal');
```

Output Waveforms:



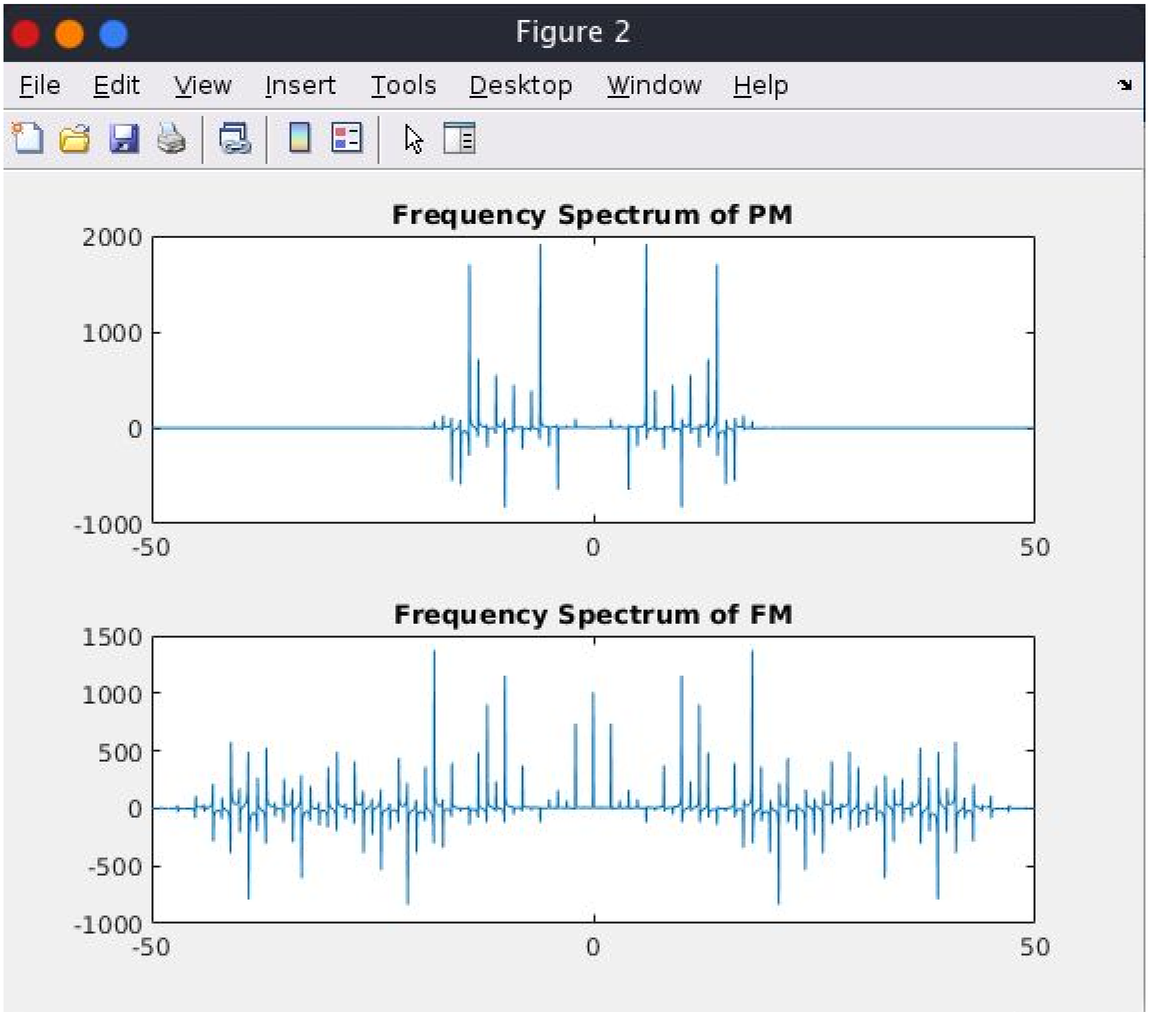


Figure 2
File
Edit
View
Insert
Tools
Desktop
Window
Help
Frequency Spectrum of PM
2000
1000
0
-1000
-50
0
50
Frequency Spectrum of FM
1500
1000
500
0
-500
-1000
-50
0
50

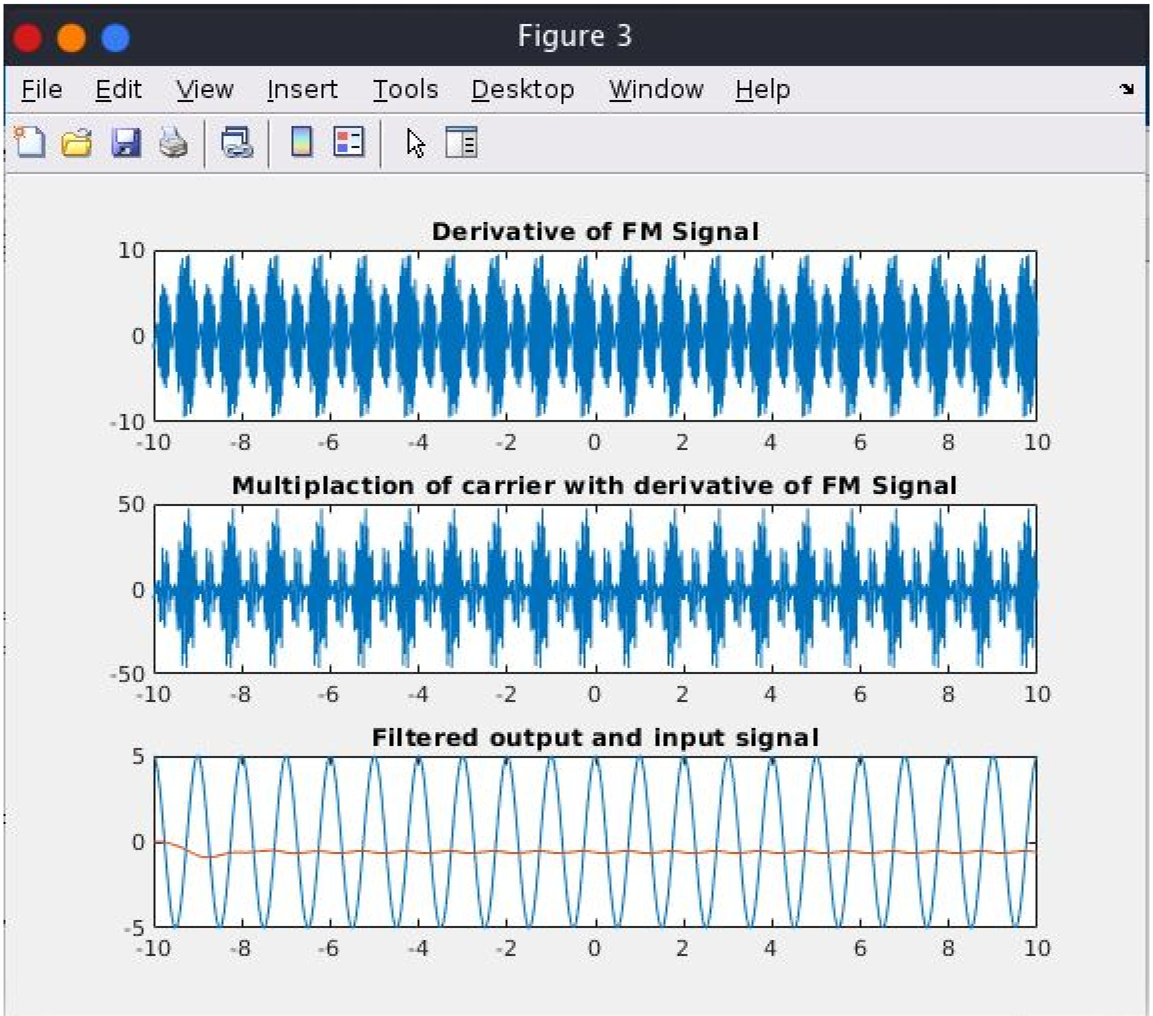


Figure 3
File Edit View Insert Tools Desktop Window Help
Derivative of FM Signal
Multiplaction of carrier with derivative of FM Signal
Filtered output and input signal

Conclusion:

In this experiment we have implemented frequency modulation and demodulation in *Matlab*. We also analysed the spectrum of modulated FM and PM signals.

To implement frequency modulation and demodulation using MATLAB.

Remarks: **Signature:**

References:

- NPTEL communication systems lectures
  https://www.youtube.com/watch?v=gsUaHawPy-w&list=PL7748E9BEC4ED83CA&index=15
- Modern Analog and Digital Communication by B.P. Lathi (3rd or 4th edition)
- Communication Systems by Simon Haykin (4th edition)

<u>Experiment no.:</u> 11 **<u>Date:</u>**

AIM: To find the Numerical Aperture of given optical fiber.

APPARATUS: Emitter, Fiber cable, Fiber stand, Detector

THEORY:

What is optic fiber?

Optical fibers are fine transparent glass or plastic fibers which can propagate light. They work under the principle of total internal reflection from diametrically opposite walls. In this way light can be taken anywhere because fibers have enough flexibility. This property makes them suitable for data communication, design of fine endoscopes, micro sized microscopes etc. An optic fiber consists of a core that is surrounded by a cladding which is normally made of silica glass or plastic. The core transmits an optical signal while the cladding guides the light within the core. Since light is guided through the fiber it is sometimes called an optical wave guide. The basic construction of an optic fiber is shown in figure (1).

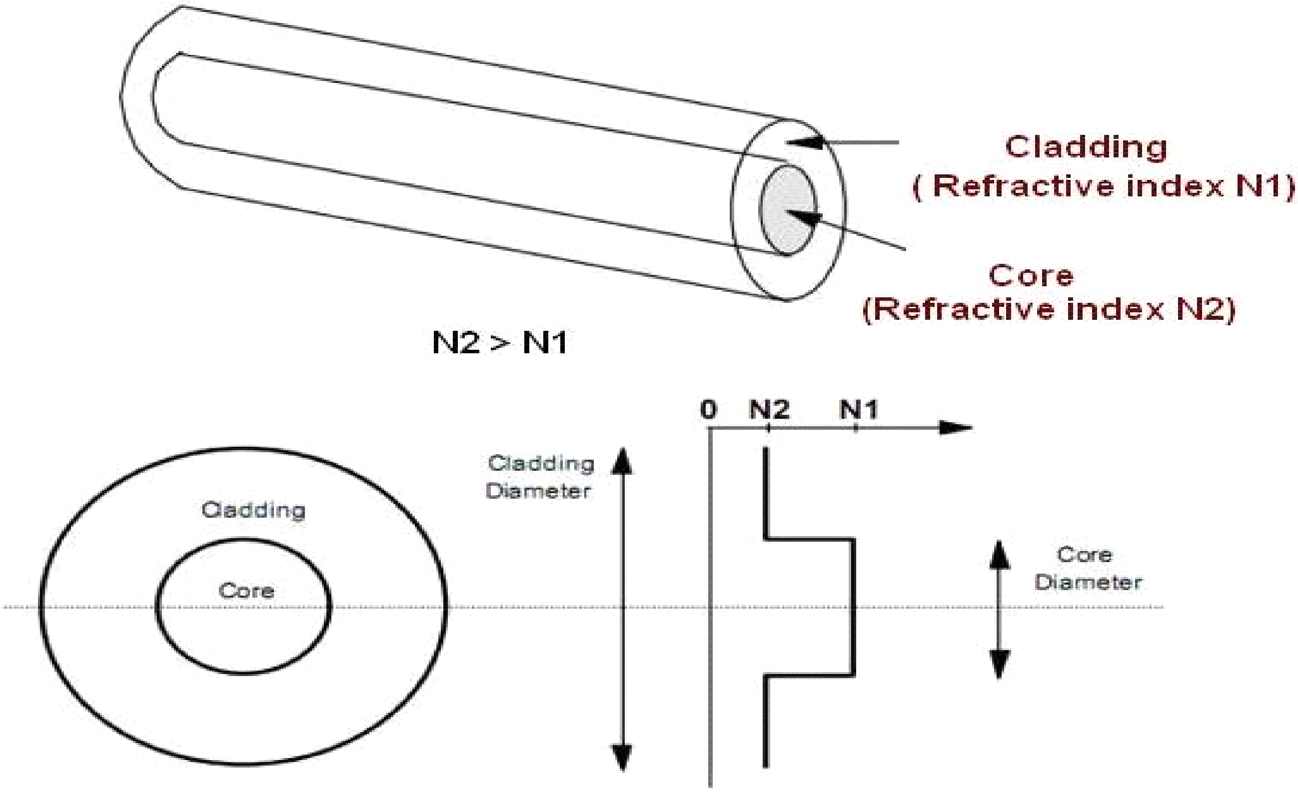


Fig (1)

In order to understand the propagation of light through an optical fiber, consider the figure (2). Consider a light ray (i) entering the core at a point A, travelling through the core until it reaches the core cladding boundary at point B. As long as the light ray intersects the core-cladding boundary at small angles, the ray will be reflected back in to the core to travel on to point C where the process of reflection is repeated .i.e., total internal reflection takes place. Total internal reflection occurs only when the angle of incidence is greater than the critical angle. If a ray enters an optic fiber at a steep angle (ii), when this ray intersects the core-cladding boundary, the angle of intersection is too large. So, reflection back in to the core does not take place and the light ray is lost in the cladding. This means that to be guided through an optic fiber, a light ray must enter the core with an angle less than a particular angle called the acceptance angle of the fiber. A ray which enters the fiber with an angle greater than the acceptance angle will be lost in the cladding.

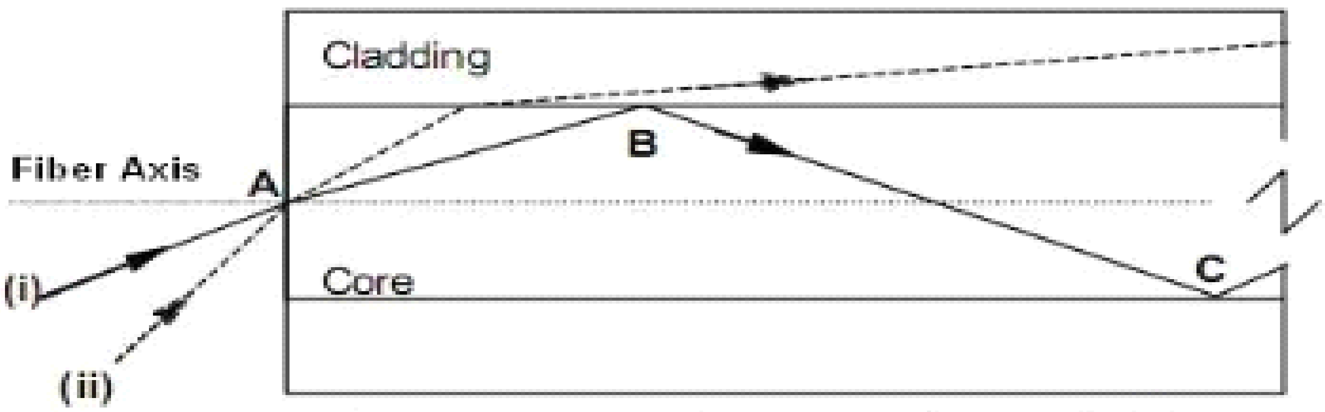


Figure 2 Propagation of light in an optical fibre

Consider an optical fibre having a core of refractive index n_1 and cladding of refractive index n_2. let the incident light makes an angle i with the core axis as shown in figure (3).

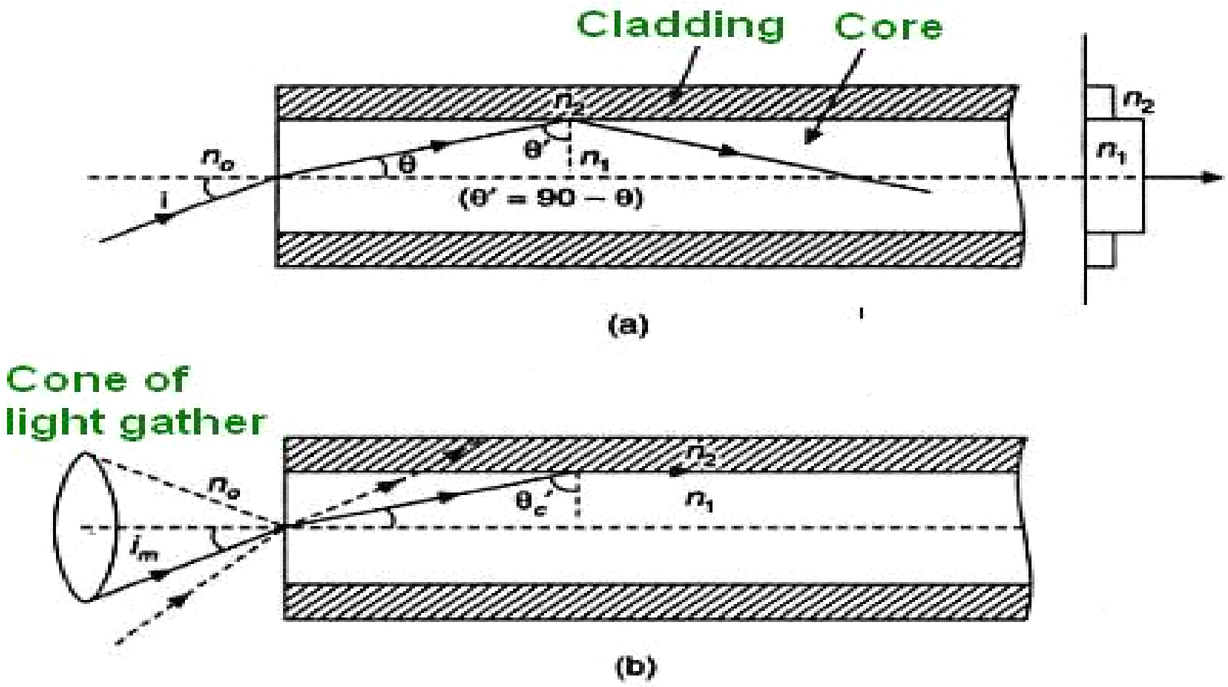


Figure 3.

Then the light gets refracted at an angle θ and fall on the core-cladding interface at an angle where,

$$\theta' = (90 - \theta) \quad \text{-----------------------} \ (1)$$

By Snell's law at the point of entrance of light in to the optical fiber we get,

$$n_0 \sin i = n_1 \sin\theta \quad \text{---------------------} \ (2)$$

Where n_0 is refractive index of medium outside the fiber. For air $n_0 = 1$.

When light travels from core to cladding it moves from denser to rarer medium and so it may be totally reflected back to the core medium if θ' exceeds the critical angle θ'_c. The critical angle is that angle of incidence in denser medium (n1) for which angle of refraction become 90°. Using Snell's laws at core cladding interface,

$$n_1 \sin\theta'_c = n_2 \sin 90$$

or

$$Sin\,\theta'_c = \frac{n_2}{n_1} \quad \text{------------------------ (3)}$$

Therefore, for light to be propagated within the core of optical fiber as guided wave, the angle of incidence at core-cladding interface should be greater than θ'_c. As i increases, θ increases and so θ' decreases. Therefore, there is maximum value of angle of incidence beyond which, it does not propagate rather it is refracted in to cladding medium (fig: 3(b)). This maximum value of i say i_m is called maximum angle of acceptance and $n_0 \sin i_m$ is termed as the numerical aperture (NA). From equation(2),

$$NA = n_0 \sin i_m = n_1 \sin\theta$$

$$= n_1 \sin(90 - \theta_c)$$

$$Or\ NA = n_1\ Cos\,\theta'_c$$

$$= n_1 \sqrt{1 - Sin^2\theta'_c}$$

$$\sin\theta'_c = \frac{n_2}{n_1}$$

From equation (2)

$$NA = n1\sqrt{1 - \frac{n_2^{\,2}}{n_1^2}}$$

Therefore,

$$NA = \sqrt{n_1^{\,2} - n_2^{\,2}}$$

The significance of NA is that light entering in the cone of semi vertical angle i_m only propagate through the fibre. The higher the value of i_m or NA more is the light collected for propagation in the fibre. Numerical aperture is thus considered as a light gathering capacity of an optical fibre.

Numerical Aperture is defined as the Sine of half of the angle of fibre's light acceptance cone. i.e. NA= Sin θ_a where θ_a, is called acceptance cone angle.
Let the spot size of the beam at a distance d (distance between the fiber end and detector) as the radius of the spot(r). Then,

$$\sin\theta = \frac{r}{\sqrt{r^2 + d^2}} \quad \text{-------------------------} \quad (4)$$

PROCEDURE:

- Select all tools i.e. emitter, fiber, fiber stand, output screen by clicking on that.
- Now press start button.
- Vary the distance of screen (L) by scrolling the button. Diameter (D) will also vary. Note down that value.
- Repeat that process and get the value of L and D.

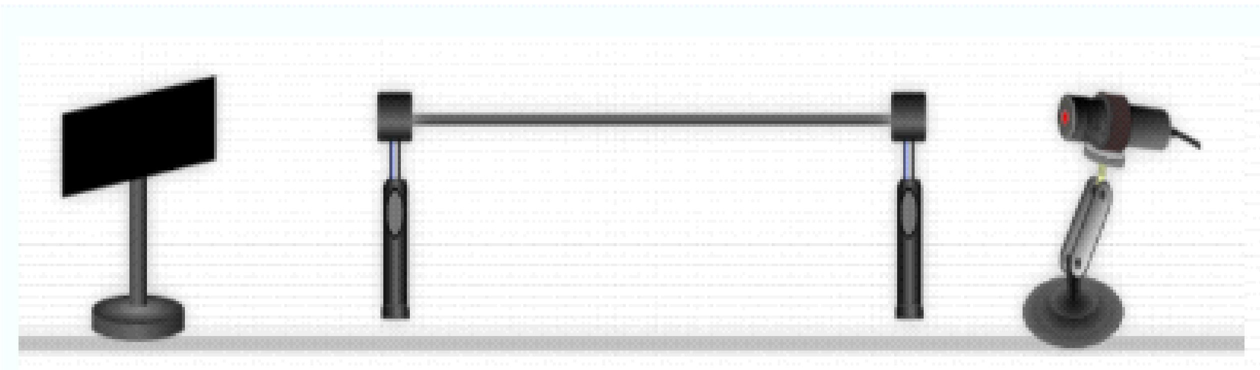
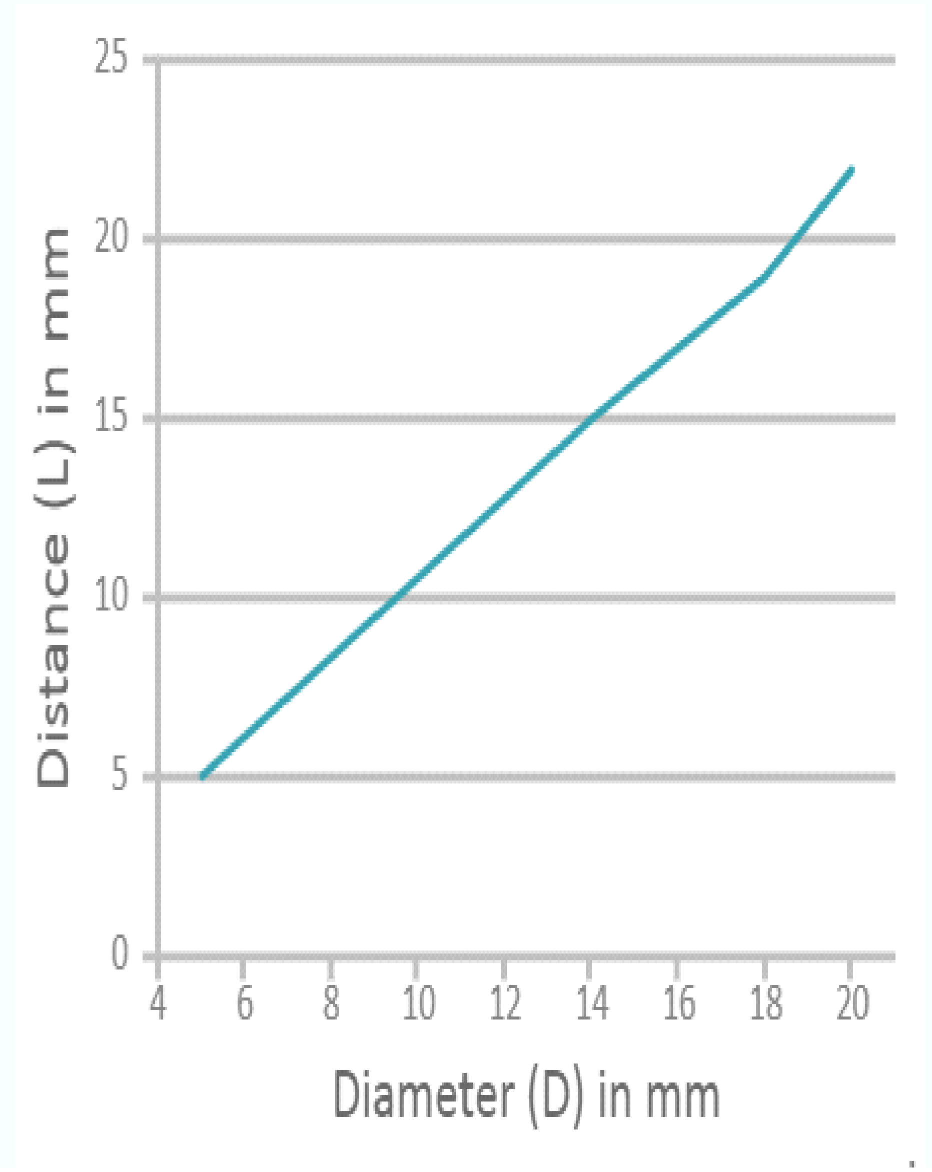


Fig.4 Tools arrangement

OBSERVAITION:

DATA TABLE		
SR. NO	Distance of screen (L) in mm	Diameter(D) in mm
1	5	5.554
2	14	15.554
3	18	19.998
4	20	22.22

DRAW GRAPH:



CONCLUSION:

In this experiment, we found the numerical aperture of the given optical fiber by using virtual lab.

Remarks: **Signature:**