Expt. No. 5 Adm No : U19CS048 Date 16/09/2020

Page No.s: 66-84

Experiment No: 05
PULSE MODULATION

Objective: To examine the pulse amplitude modulation (PAM), pulse position modulation (PPM) and pulse width modulation (PWM) and verify and draw the resultant waveforms. Illustrate the circuit diagrams for PAM and PWM. Show and draw the output waveforms using Matlab code / Simulink using virtual mode.

Software: Matlab

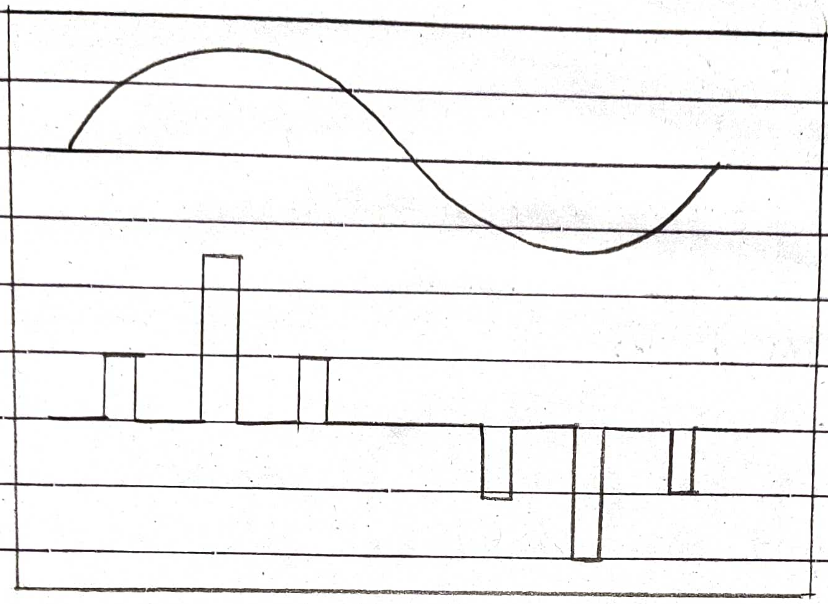
Theory:

Pulse modulation is a type of modulation in which the signal is transmitted in the form of pulses. In pulse modulation, continuous signals are sampled at regular intervals.

Pulse modulation is further divided into Analog and Digital modulation and further analog and digital modulation is subdivided in PAM, PWM, PPM (analog) and PCM, DM (digital)

PAM:-

In Pulse Amplitude Modulation (PAM) a pulse signal is used to sample an analog signal. The result is a train of constant-width pulses. The amplitude of each pulse is proportional to the amplitude of the message signal at the time of sampling. The PAM signal follows the amplitude of the original signal, as the signal traces out the path of the whole wave.



Properties →

- Amplitude of the pulse is proportional to amplitude of modulating signal
- Bandwidth of the transmission channel depends on the pulse width
- Instantaneous power of the transmitter varies
- Noise interference is high
- Similar to Amplitude Modulation
- System is complex to implement.

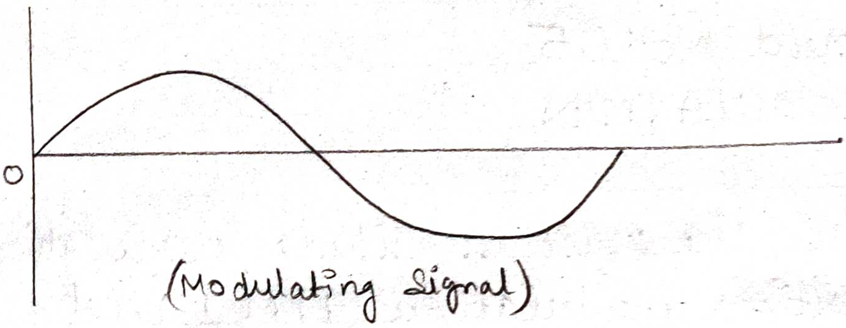
PAM Signal Generation →

We can generate PAM signal by two types of sampling possible:

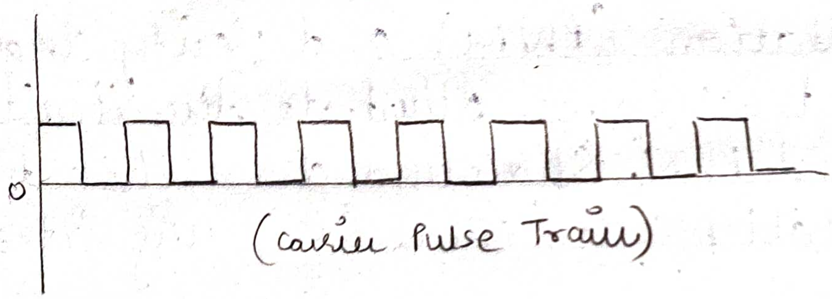
a) Natural Sampling

b) Flat - top Sampling

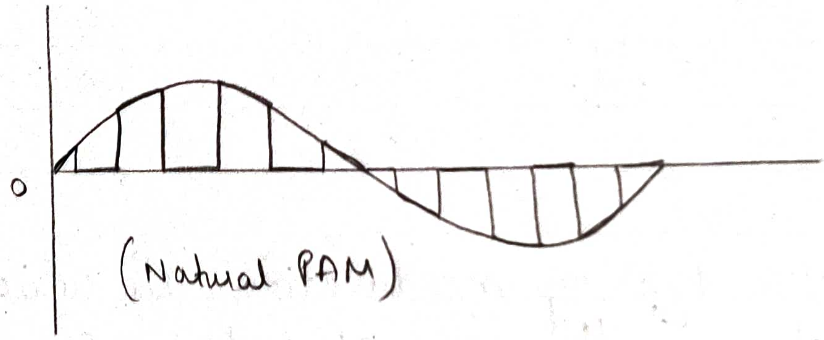
Natural Sampling :-



(Modulating Signal)

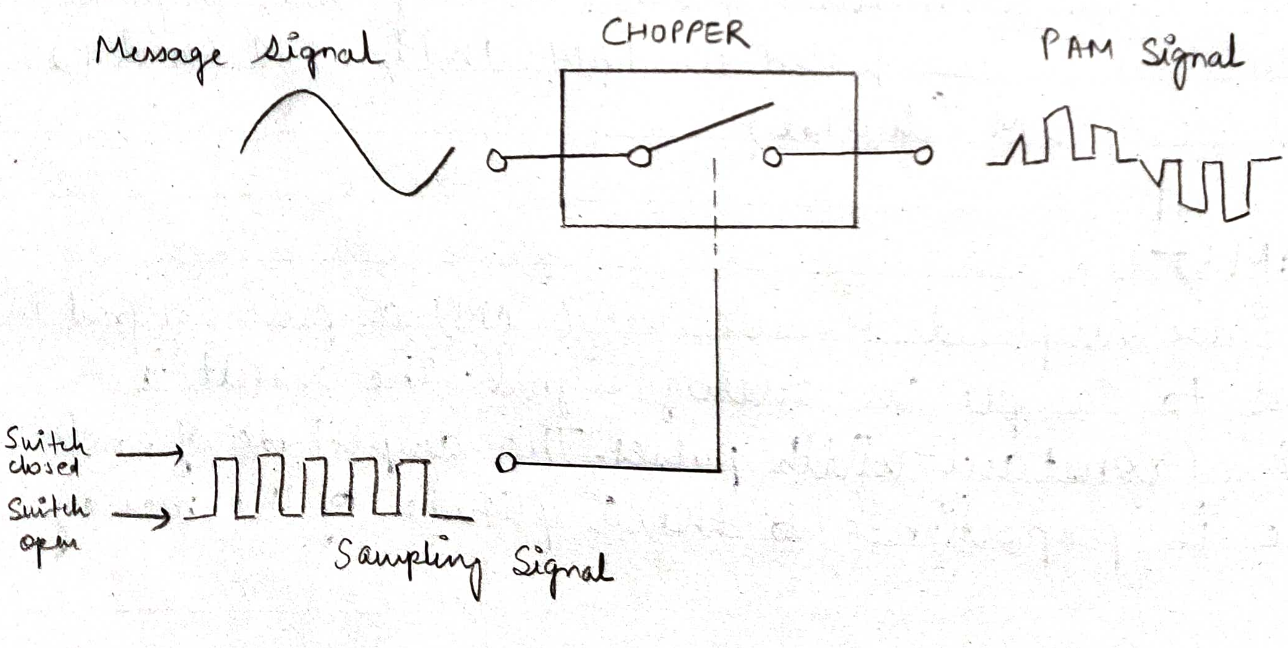


(carrier Pulse Train)

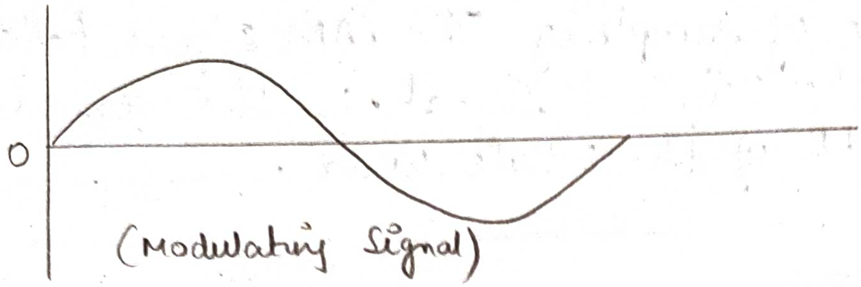


(Natural PAM)

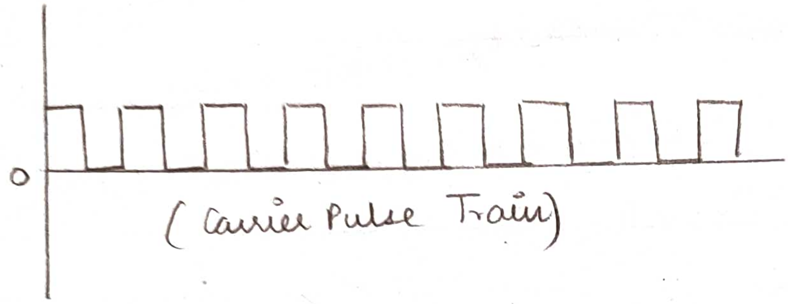
Generation of PAM by Natural Sampling :



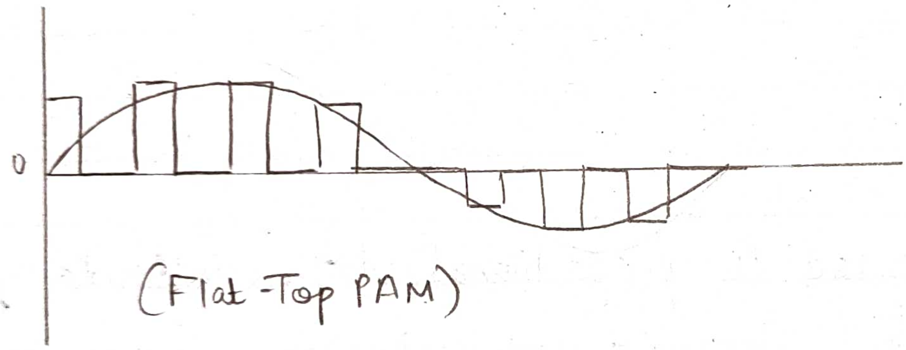
Flat Top Sampling :-



(Modulating Signal)

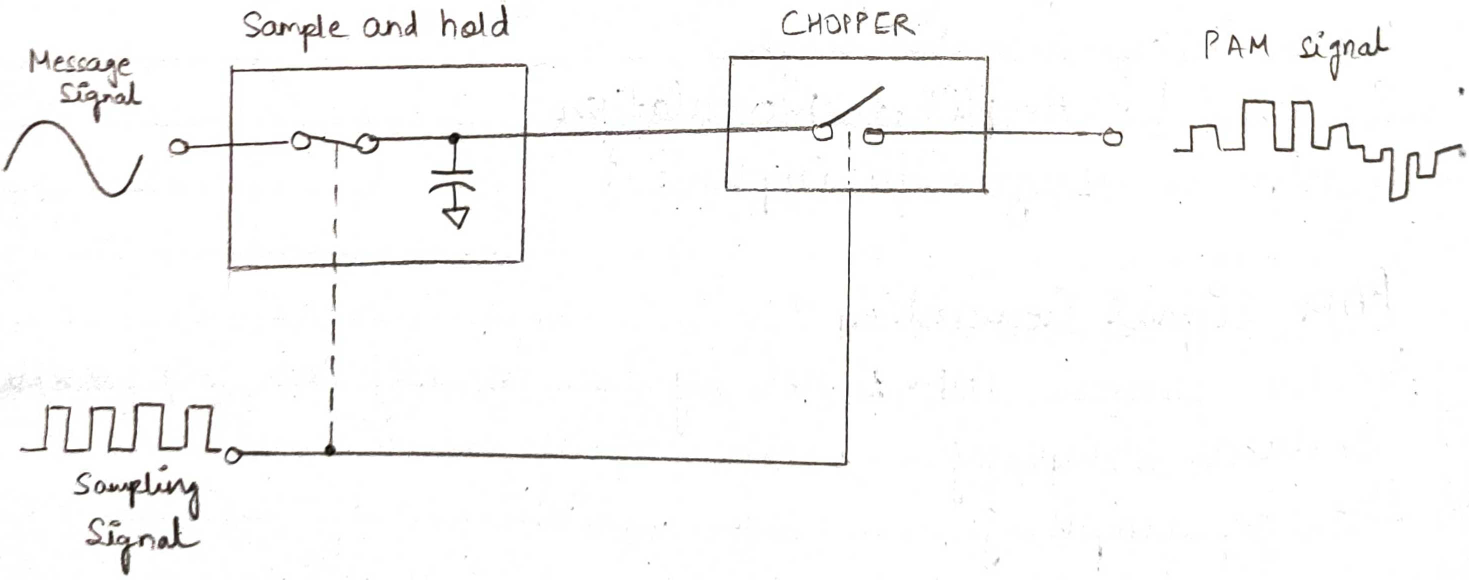


(Carrier Pulse Train)



(Flat-Top PAM)

Generation of PAM by Flat-Top Sampling :

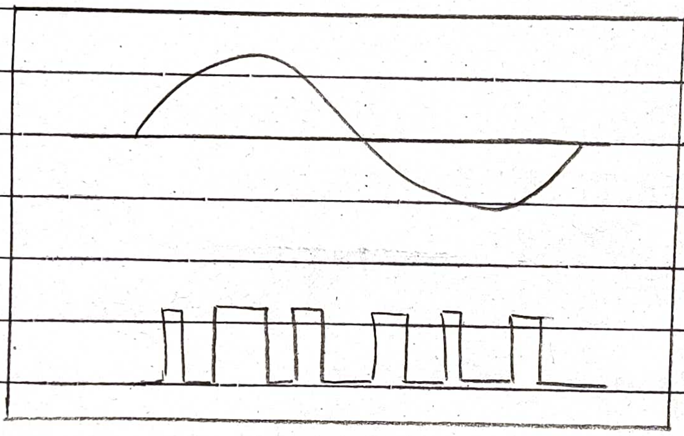


Natural Sampling → For a PAM signal produced with natural sampling, the sampled signal follows the waveform of the input signal during the time that each sample is taken.

Flat-top Sampling → In this sampled signal can be represen-ted in pulses for which the amplitude of signal cannot be changed with respect to analog signal to be sampled.

PWM :→

In this type, the amplitude is maintained constant but duration or length or width of each pulse is varied in accordance with instantaneous value of analog signal.



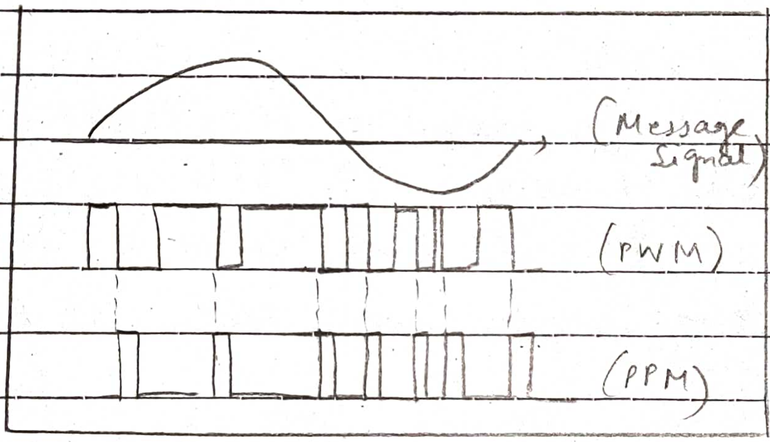
Properties →

- Width of the pulse is proportional to amplitude of modulating signal
- Bandwidth of the transmission channel depends on the rise time of the pulse
- Instantaneous power of the transmitter varies.
- Noise interference is minimum

- System is simple to implement
- Similar to frequency modulation.

PPM :→

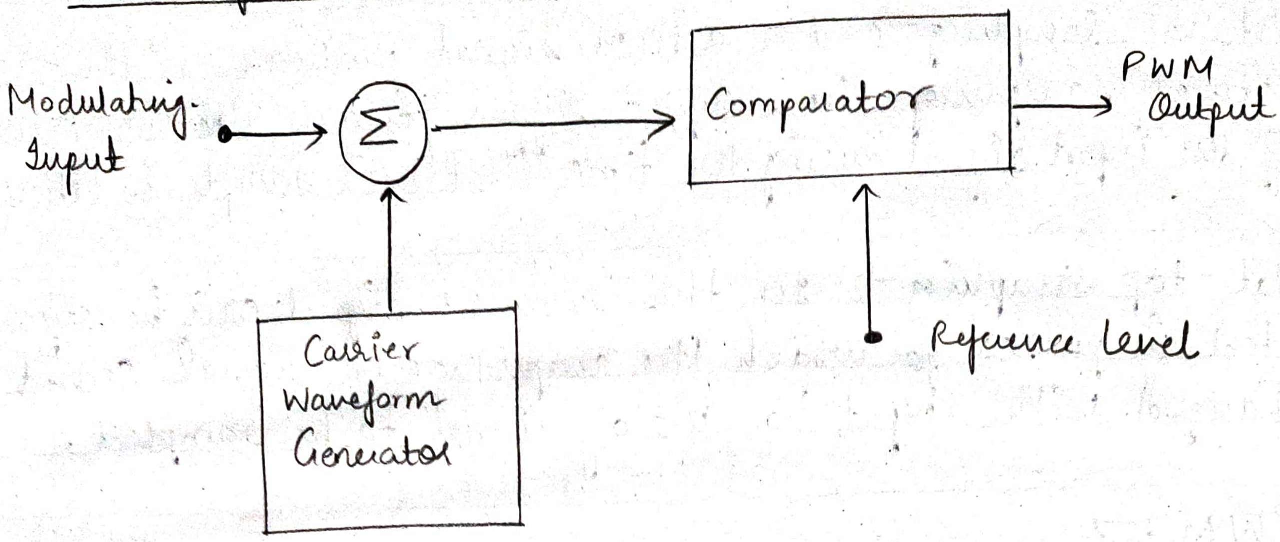
In this type of modulation, both the amplitude and width of pulse are kept constant. We vary the position of each pulse according to the instantaneous sampled value of message signal. PPM is further modification of PWM.



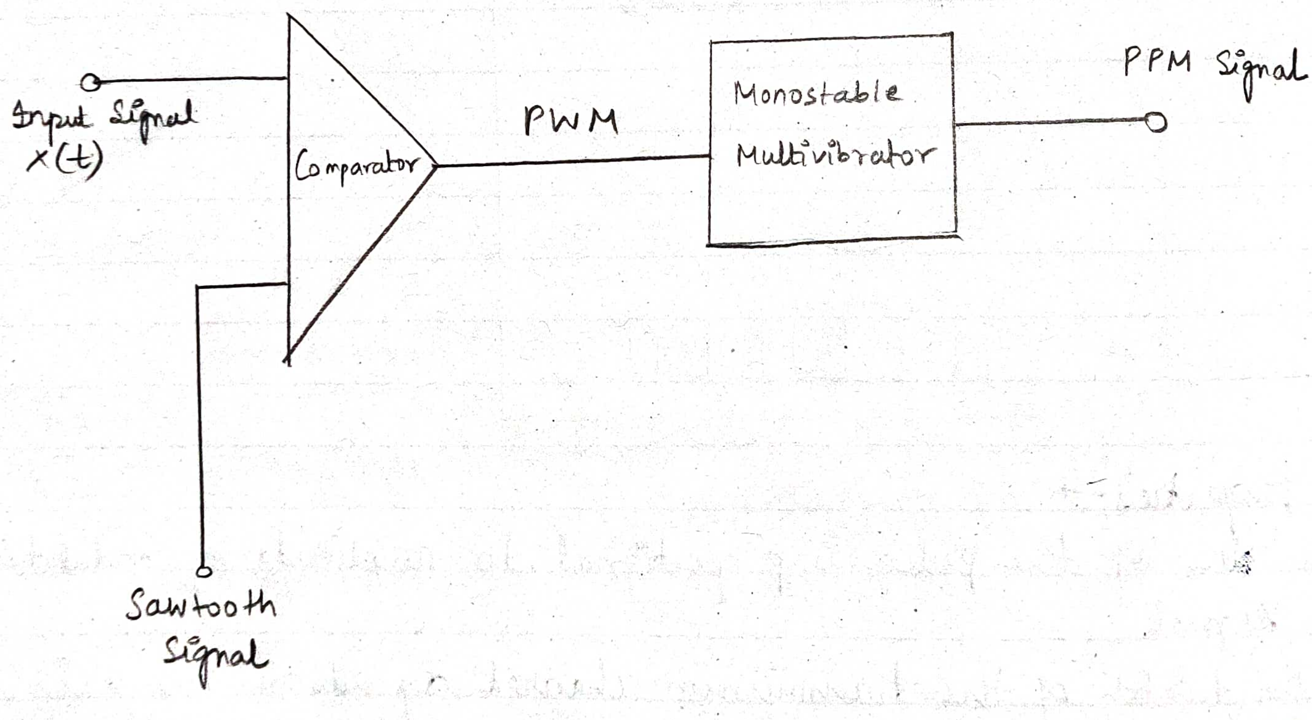
Properties →

- The relative position of the pulse is proportional to amplitude of modulating signal.
- Bandwidth of the transmission channel depends on the rising time of pulse.
- Instantaneous power of the transmitter remains constant
- Noise interference is minimum
- System is simple to implement
- Similar to phase modulation.

PWM Signal Generation:



PPM Signal Generation:



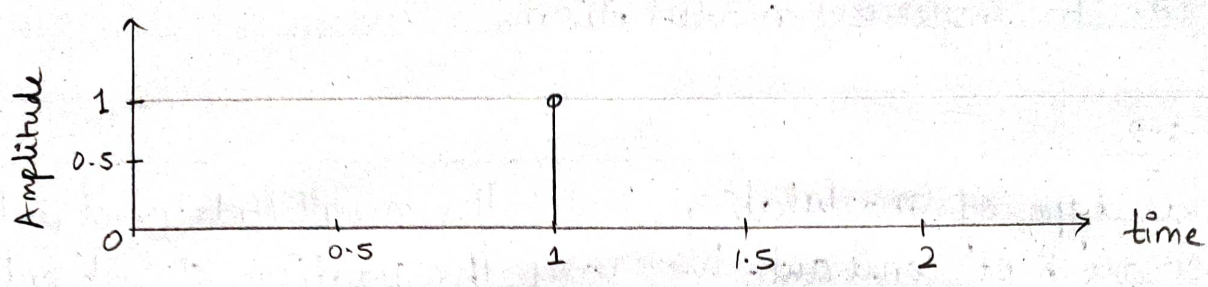
Matlab Code:

% PAM using ideal Sampling

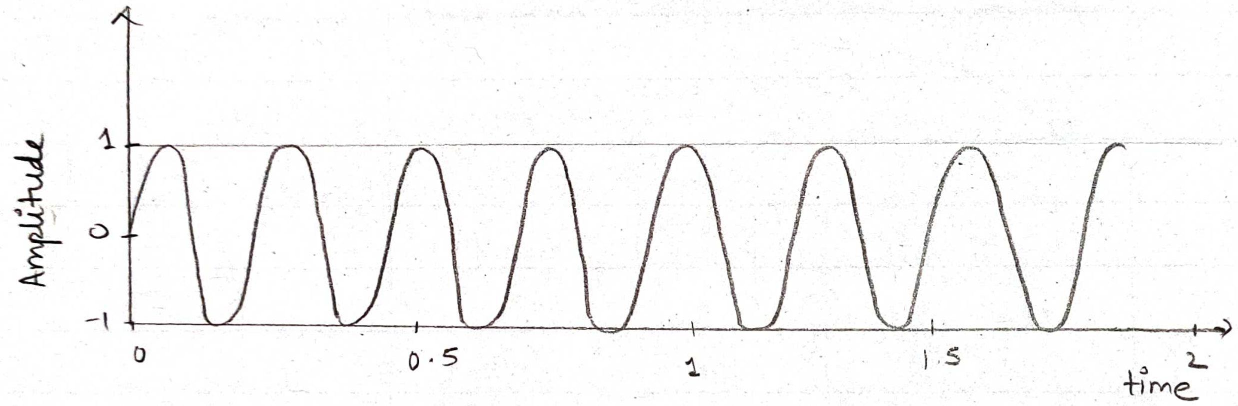
```
clc;
close all;
clear all;
a = input('Enter the amplitude = ');
f = input('Enter the frequency = ');
t = 0:0.02:2;
x1 = 1; % generation of an impulse signal
x2 = a*sin(2*pi*f*t); % generation of sine wave
y = x1.*x2; % modulation step
subplot(3,1,1); % for impulse signal plot
stem(x1)
title('Impulse Signal');
xlabel('Time');
ylabel('Amplitude');
subplot(3,1,2) % for sine wave plot
plot(t,x2)
title('Sine Wave');
xlabel('Time');
ylabel('Amplitude');
subplot(3,1,3) % for PAM wave plot
stem(t,y)
title('PAM Wave');
xlabel('Time');
ylabel('Amplitude');
```

(Amplitude = 1, freq = 5)

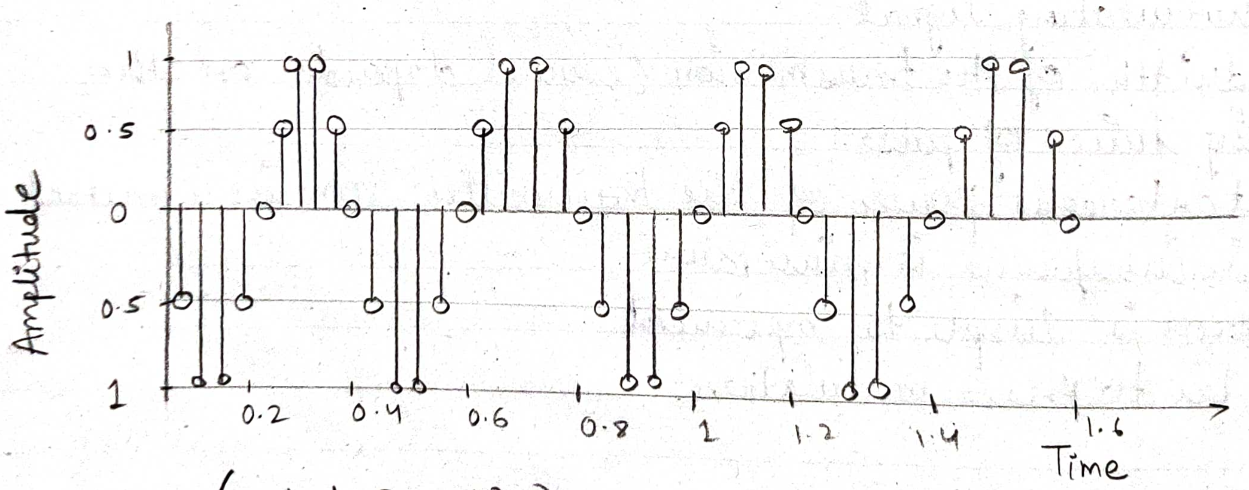
Impulse Signal:



Sine Wave:



PAM Wave:

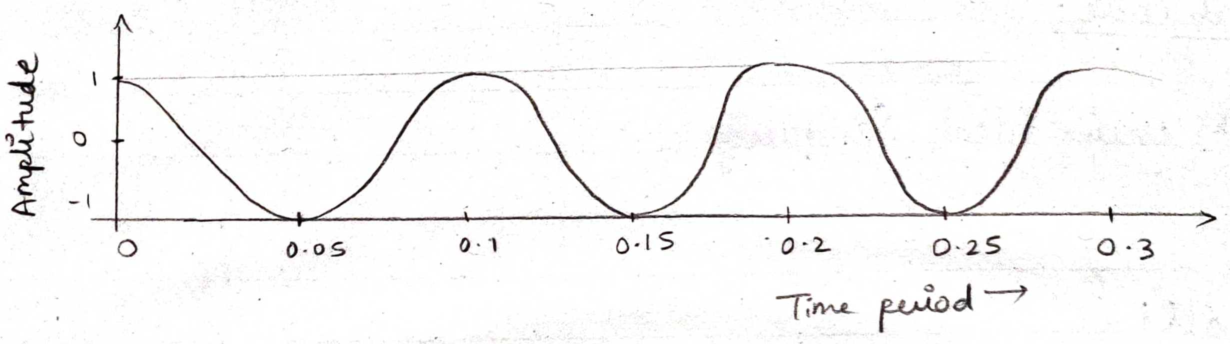


(Ideal Sampling)

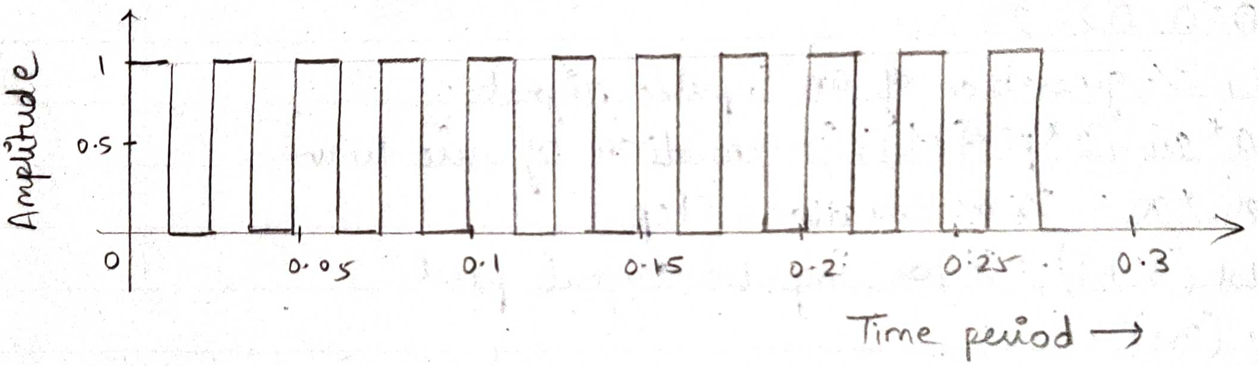
```
% PAM using natural Sampling
clc; clear all; close all;
fc=100
fm = fc/10
fs = 100* fc
t=0:1/fs:4/fm;
Msg_sgl = Cos(2*pi*fm*t);
Carr_sgl = 0.5*Square(2*pi*fc*t)+0.5
Mod_sgl = Msg_sgl.*Carr_sgl;
tt=[];
for i=1:length(Mod_sgl);
    if Mod_sgl(i)==0;
        tt=[tt, Mod_sgl(i)];
    else
        tt=[tt, Mod_sgl(i)+2];
    end
end
figure(1)
Subplot(4,1,1);
plot(t, Msg_sgl);
title('Message Signal');
xlabel('Time period');
ylabel('Amplitude');
```

```
subplot (4,1,2);
plot ( t, Carr_sgl);
title('Carrier Signal');
xlabel ('Time period ');
ylabel ('Amplitude');

subplot (4,1,3);
plot ( t, Mod_sgl );
title('PAM Modulated signal')
xlabel ('Time Period');
ylabel ('Amplitude')
```

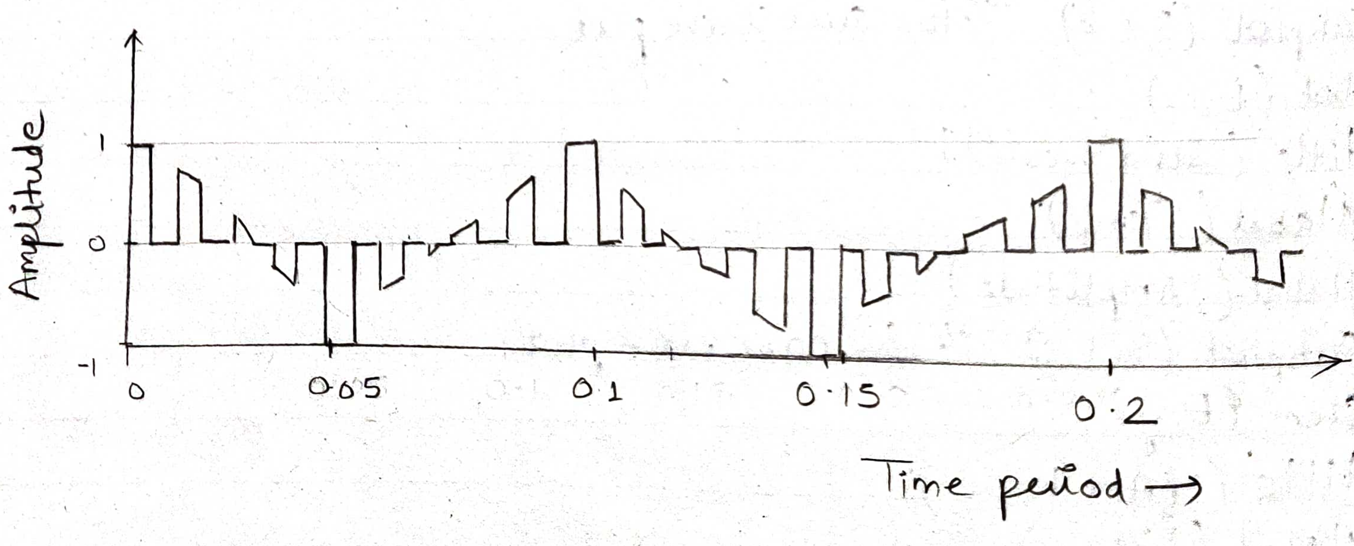
Message Signal :



Carrier Signal :



PAM Modulated Signal :

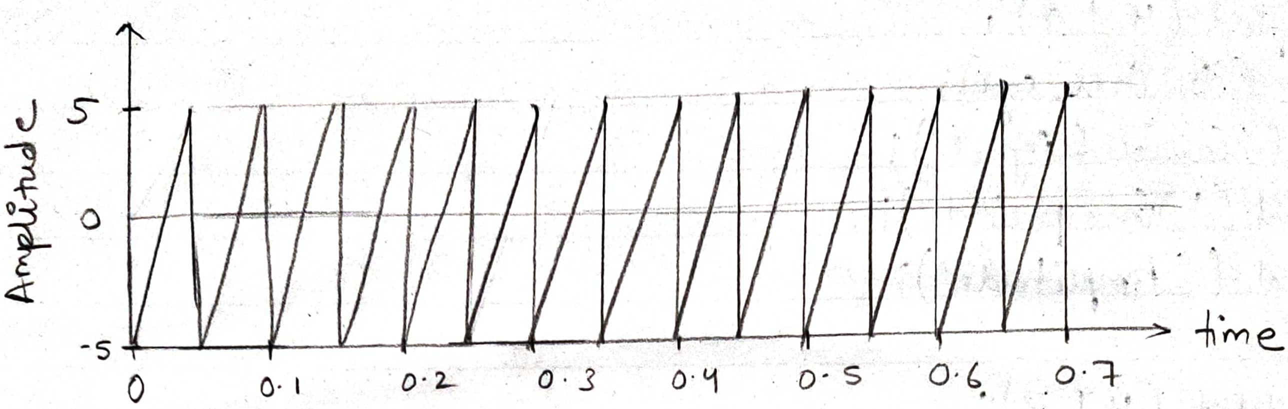


% PWM Signal

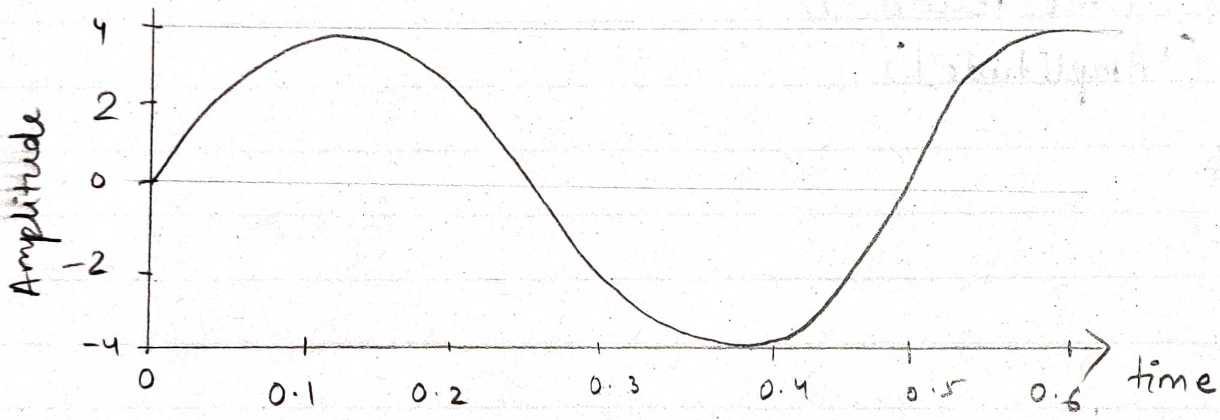
```
clc; clear all; close all;
F2 = input('Message frequency =');
F1 = input('Carrier Sawtooth frequency=');
A=5;
t=0:0.001:1
c = A* sawtooth(2*pi*F1*t);
subplot(3,1,1);
plot(t,c);
xlabel('time'); ylabel('Amplitude');
title('Carrier Sawtooth Wave');
grid on;
m = 0.75*A.*sin(2*pi*F2*t);
subplot(3,1,2);
plot(t,m);
xlabel('Time'); ylabel('Amplitude');
title('Message signal'); grid on;
n = length(c);
for i=1:n
    if (m(i)>=c(i))
        pwm(i)=1
    else
        pwm(i)=0
    end
end
```

```
subplot (3,1,3);
plot (t, pwm);
xlabel ('Time');
ylabel ('Amplitude');
title ('Plot of PWM');
axis ([0 1 0 2]);
grid on;
```

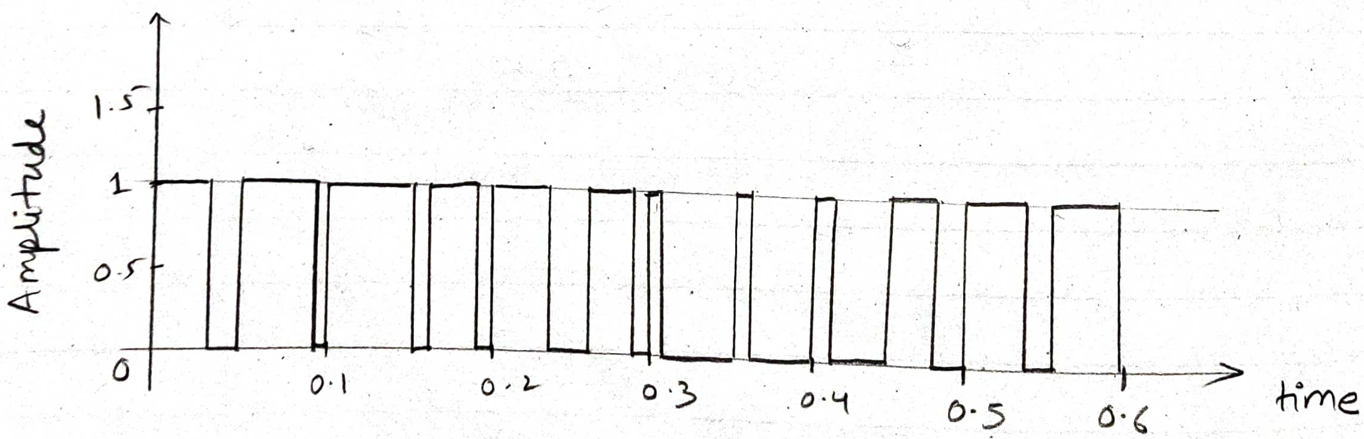
Carrier Sawtooth wave:



Message signal:



Plot of PWM:

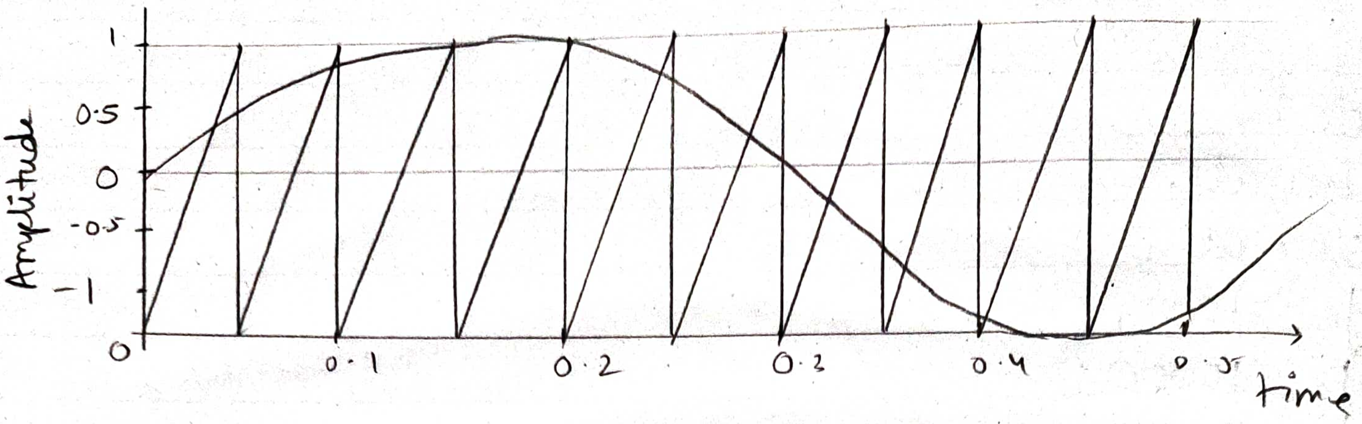


% PPM Signal

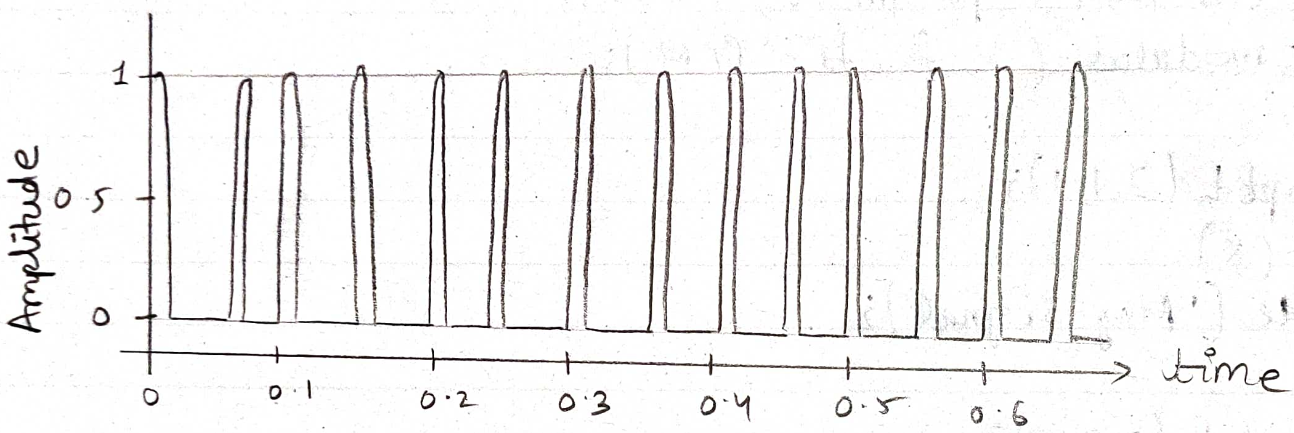
```
clc; clear all; close all;
fc = 20;
fs = 1000;
t = 1;
fm = 2;
n = [0 : 1/fs : t]
n = n ( 1: end -1) ]
duty = 10;
per = fs / fc;
on_t = per / duty;
s = square (2* pi *fc * n, duty);
s ( find (s<0) ) = 0
m = sin( 2* pi * fm *n);

% Transfer wave
A = 1.25;
c = A.* sawtooth (2* pi * fc*n);

PPM = zeros( 1, length (s));
id = find (c> m);
idd = diff (id);
iddd = find (id~= 1)
temp (1) = id (1)
temp (2: length (iddd)+1) = id (iddd +1);
for i=1 : length (temp)
```

```
    ppm (temp(i) : temp (i) + on_t -1) = 1;
end

% plot
Subplot (3,1,1); plot (n, m, 'linewidth', 2);
title ('message signal'); hold on;
plot (n, c, 'r', 'linewidth', 2); grid on;

subplot (3,1,2); plot (n, s, 'linewidth', 2);
title ('Pulse train');
grid on;
ylim ([-0.2 1.2])
Subplot (3,1,3); plot (n, ppm, 'linewidth', 2);
title ('pulse train'); grid on;
ylim ([-0.2 1.2])
Subplot (3,1,3); plot (n, ppm, 'linewidth', 2);
title ('PPM signal'); grid on;
ylim([-0.2 1.2]);
```

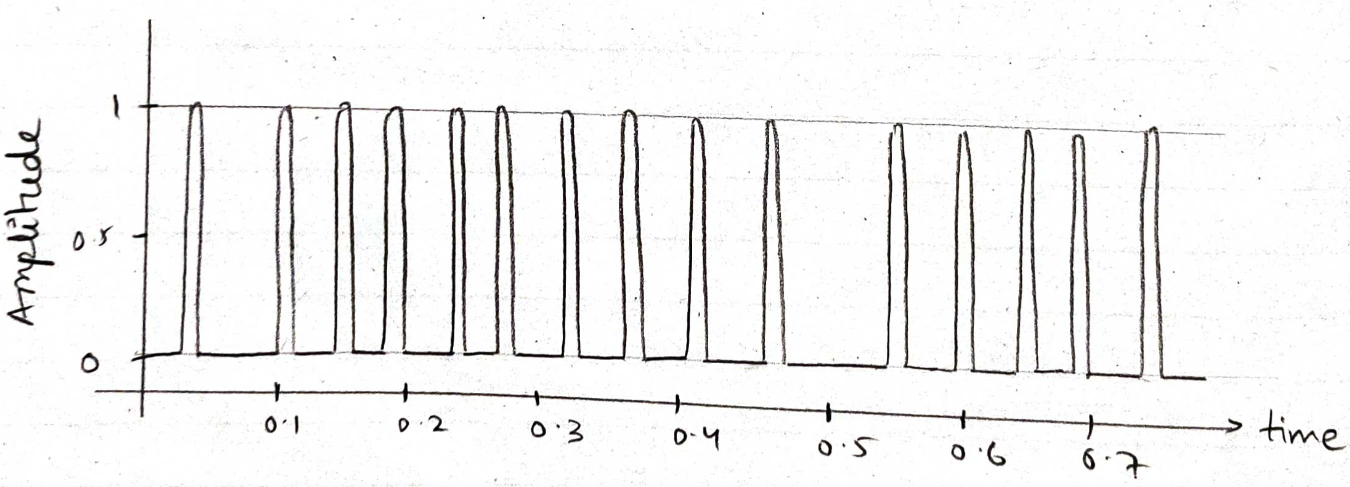
Message Signal:



Pulse Train:



PPM Signal:



Advantages of Pulse Modulation:

1) The pulses are quite short as compared to the time in between, so a pulse modulated wave remains off most of the time.
2) The time interval between pulses may be filled with sample values from other messages, so we can send many messages at a time on a pulse communication system.
3) One of the chief advantages of PM is that if we combine pulse modulation with continuous modulation (AM, FM, PM) we can obtain "multi channel" communication system, a desirable feature for data transmission.

Conclusion:

Successfully performed PAM, PWM, PPM and analysed and verified its input and output waveforms using MATLAB software.