American International University – Bangladesh (AIUB)

OEL Lab Report

Course Title: Data Communication

Submitted To-

Abrar Fahim Liaf

Faculty

FACULTY OF ENGINEERING

Submitted by

Name: MD. Sumon

ID:20-42556-1

Section: I

Title: Demonstrate an experiment for shift keying (ASK) and multiplexing (FDM) to communicate binary bits as analog signals; Demultiplex and convert them back into binary bits at the receiver.

Purpose:

This experiment aims to develop an understanding of Digital Analog Conversion using MATLAB and to understand FDM (Frequency Division Multiplexing) concept and ASK (Amplitude Shift Keying) And how to implement this in MATLAB.

Procedure:

1. Choose three Digital Binary data according to choice and convert them into Analog Signals using ASK.
2. Then the Analog Signals are multiplexed using FDM.
3. The Analog composite signals are demultiplexed to their original individual signals.
4. Assuming some data as MSG1, MSG2, and MSG3 the individual Analog signals are converted back into the original digital data (as binary bits).

Results:

Source Code:

```
clc

close all;

f=5;

% input signal 1 ;

msg1=[1 0 0 1]

nx=size(msg1,2);

i=1;

while i<nx+1

t = i:0.001:i+1;

if msg1(i)==1

ask1=sin(2*pi*f*t);

else

ask1=0;

end

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

```
plot(t,ask1);
hold on;
grid on;
axis([1 10 -1 1]);
title('Amplitude Shift Key MSG1')
i=i+1;
end
% input signal 2;
msg2=[0 1 1 0 ]
nx=size(msg2,2);
i=1;
while i<nx+1
t = i:0.001:i+1;
if msg2(i)==1
ask2=sin(2*pi*f*t);
else
ask2=0;
end
subplot(3,1,2);
plot(t,ask2);
hold on;
grid on;
axis([1 10 -1 1]);
title('Amplitude Shift Key MSG2')
i=i+1;
end

% input signal 3;
msg3=[1 1 0 1 ]
nx=size(msg3,2);
i=1;
while i<nx+1
t = i:0.001:i+1;
if msg3(i)==1
ask3=sin(2*pi*f*t);
```

```
else
ask3=0;
end
subplot(3,1,3);
plot(t,ask3);
hold on;
grid on;
axis([1 10 -1 1]);
title('Amplitude Shift Key MSG3')
i=i+1;
end
%% Message Signal Generation
fs = 4001; %Sampling Frequency
t = 0:1/fs:1-1/fs; %Generating Time axis
Am1 = 9; %Amplitude of First Message Signal
fm1 = f; %Frequency of First Message Signal
m1 = Am1*cos(2*pi*fm1*t); % First Message Signal
Am2 = 6; %Amplitude of Second Message Signal
fm2 = f; %Frequency of Second Message Signal
m2 = Am2*cos(2*pi*fm2*t); % Second Message Signal
Am3 = 13; %Amplitude of Third Message Signal
fm3 = f; %Frequency of Third Message Signal
m3 = Am3*cos(2*pi*fm3*t); % Third Message Signal
%%
%% Carrier Signal Generation
Cm1 = 1; %Amplitude of First Carrier Signal
fc1 = 100; %Frequency of First Carrier Signal
c1 = Cm1*cos(2*pi*fc1*t); % First Carrier Signal
Cm2 = 1; %Amplitude of Second Carrier Signal
fc2 = 175; %Frequency of Second Carrier Signal
c2 = Cm2*cos(2*pi*fc2*t); % Second Carrier Signal
Cm3 = 1; %Amplitude of Third Carrier Signal
fc3 = 250; %Frequency of Third Carrier Signal
c3 = Cm3*cos(2*pi*fc3*t); % Third Carrier Signal
%%
```

```
%% Composite Signal Generation
x = (m1).*c1+(m2).*c2+(m3).*c3;
%%%% Plotting the Signals in Time-Domain and Frequency-Domain
figure
subplot(3,1,1)
plot(t,m1)
xlabel('time')
ylabel('amplitude')
title('Message Signal 1 in Time Domain')
ylim([-Am1 Am1])
subplot(3,1,2)
plot(t,m2)
xlabel('time')
ylabel('amplitude')
title('Message Signal 2 in Time Domain')
ylim([-Am2 Am2])
subplot(3,1,3)
plot(t,m3)
xlabel('time')
ylabel('amplitude')
title('Message Signal 3 in Time Domain')
ylim([-Am3 Am3])
M1 = abs(fftshift(fft(m1)))/(fs/2); %Fourier Transformation of m1
M2 = abs(fftshift(fft(m2)))/(fs/2); %Fourier Transformation of m2
M3 = abs(fftshift(fft(m3)))/(fs/2); %Fourier Transformation of m3
X = abs(fftshift(fft(x)))/(fs/2); %Fourier Transformation of x
f = fs/2*linspace(-1,1,fs);
figure
subplot(3,1,1)
stem(f,M1)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 1 in Frequency Domain')
axis([-10 10 0 12])
subplot(3,1,2)
```

```
stem(f,M2)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 2 in Frequency Domain')
axis([-10 10 0 10])
subplot(3,1,3)
stem(f,M3)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 3 in Frequency Domain')
axis([-10 10 0 15])
figure
subplot(2,1,1)
plot(t,x)
xlabel('time')
ylabel('amplitude')
title('Composite Signal in Time Domain')
subplot(2,1,2)
stem(f,X)
xlabel('frequency')
ylabel('amplitude')
title('Composite Signal in Frequency Domain')
axis([-270 270 0 10])
%%
%% Passing the Composite Signal Through Bandpass Filter
[num1, den1] = butter(5, [(fc1-fm1-6)/(fs/2),(fc1+fm1+6)/(fs/2)]);
%Butterworth Filter Window Determining for Bandpass Filter
bpf1 = filter(num1,den1,x); %Filtering is done here
[num2, den2] = butter(5, [(fc2-fm2-6)/(fs/2),(fc2+fm2+6)/(fs/2)]);
%Butterworth Filter Window Determining for Bandpass Filter
bpf2 = filter(num2,den2,x); %Filtering is done here
[num3, den3] = butter(5, [(fc3-fm3-6)/(fs/2),(fc3+fm3+6)/(fs/2)]);
%Butterworth Filter Window Determining for Bandpass Filter
bpf3 = filter(num3,den3,x); %Filtering is done here
%%
```

```
%% Mixing
z1 = 2*bpf1.*c1;
z2 = 2*bpf2.*c2;
z3 = 2*bpf3.*c3;
%%
%% Passing the Mixed Signals Through Lowpass Filter
[num4, den4] = butter(5, (fm1+3)/(fs/2)); %Low pass filter is made here
rec1 = filter(num4,den4,z1); %Filtering is done here
[num5, den5] = butter(5, (fm2+3)/(fs/2)); %Low pass filter is made here
rec2 = filter(num5,den5,z2); %Filtering is done here
[num6, den6] = butter(5, (fm3+3)/(fs/2)); %Low pass filter is made here
rec3 = filter(num6,den6,z3); %Filtering is done here
%%
%% Plotting the Received Signals in Time-Domain and Frequency Domain
figure
subplot(3,1,1)
plot(t,rec1)
xlabel('time')
ylabel('amplitude')
title('received signal 1 in time domain')
ylim([-Am1 Am1])
subplot(3,1,2)
plot(t,rec2)
xlabel('time')
ylabel('amplitude')
title('received signal 2 in time domain')
ylim([-Am2 Am2])
subplot(3,1,3)
plot(t,rec3)
xlabel('time')
ylabel('amplitude')
title('received signal 3 in time domain')
ylim([-Am3 Am3])
R1 = abs(fftshift(fft(rec1)))/(fs/2); %Fourier Transformation is done here
R2 = abs(fftshift(fft(rec2)))/(fs/2); %Fourier Transformation is done here
```

```
R3 = abs(fftshift(fft(rec3)))/(fs/2); %Fourier Transformation is done here
figure
subplot(3,1,1)
stem(f,R1)
xlabel('frequency')
ylabel('amplitude')
title('received signal 1 in frequency domain')
xlim([-10 10])
ylim([0,10])
subplot(3,1,2)
stem(f,R2)
xlabel('frequency')
ylabel('amplitude')
title('received signal 2 in frequency domain')
xlim([-10 10])
ylim([0,6])
subplot(3,1,3)
stem(f,R3)
xlabel('frequency')
ylabel('amplitude')
title('received signal 3 in frequency domain')
xlim([-10 10])
ylim([0,14])
%% End
time_duration = 0.2;
%% Analog-like signal's representation
% Analog signal generation is not possible in MATLAB
a = [9 6 13]; % amplitude array for composite signal
f = [5 5 5]; % frequency array for composite signal
analog_t = 0:0.0001:time_duration;
analog_sig = a(1)*sin(2*pi*f(1)*analog_t) + a(2)*cos(2*pi*f(2)*analog_t) + a(3)*sin(2*pi*f(3)*analog_t +pi/4);
figure
subplot(1,2,1)
plot(analog_t, analog_sig,'linewidth',1.5)
grid on
```

```
xlabel('time in seconds')
ylabel('amplitude in volts')
title('analog signal')
%% Sampling Frequency
fs = 250;
ts = 1/fs;
%% Sampling
samp_t = 0:1/fs:time_duration;
samp_sig = a(1)*sin(2*pi*f(1)*samp_t) + a(2)*cos(2*pi*f(2)*samp_t) + a(3)*sin(2*pi*f(3)*samp_t + pi/4);
subplot(1,2,2)
plot(samp_t, samp_sig,'o','linewidth',1.5)
grid on
xlabel('time in seconds')
ylabel('amplitude in volts')
title(['sampled signal for ',num2str(fs),' Hz sampling frequency'])
%% Levels for Quantization
L = 8;
%% Quantizing
delta = (max(samp_sig) - min(samp_sig))/(L-1); % step size
quant_sig = min(samp_sig) + round((samp_sig-min(samp_sig))/delta)*delta; % quantized signal
figure
subplot(1,2,1)
plot(samp_t, samp_sig,'o','linewidth',1.5)
grid on
xlabel('time in seconds')
ylabel('amplitude in volts')
title('sampled signal')
subplot(1,2,2)
plot(samp_t, quant_sig,'x','linewidth',1.5);
xlabel('time')
ylabel('amplitude')
title('quantized samples')
%% Number of Bits/Sample
nb = log2(L);
%% Encoding
```

```
i = round((samp_sig-min(samp_sig))/delta); % index for encoding

dig_data_matrix = de2bi(i,nb)% encoded binary bits are as amatrix here

dig_data = reshape(dig_data_matrix',1,[]); % encoded binary bitsare as an array here

disp(['The index values for encoding from quantization of thesampled signal are: ',num2str(i)])

disp(['The converted bits from the input analog signal are:',num2str(dig_data)])
```

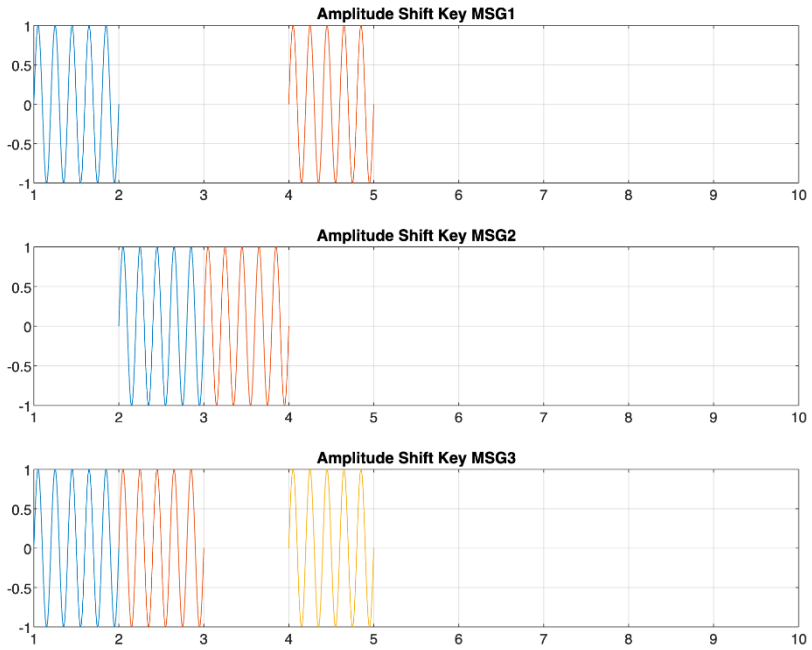

Fig 1: Amplitude Shift Keying

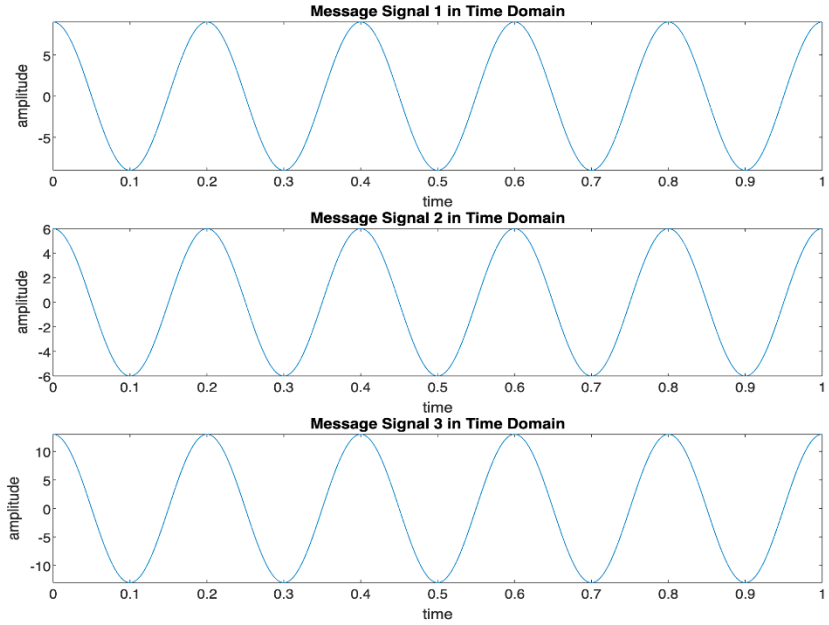

Fig 2: Message Signals in Time Domain

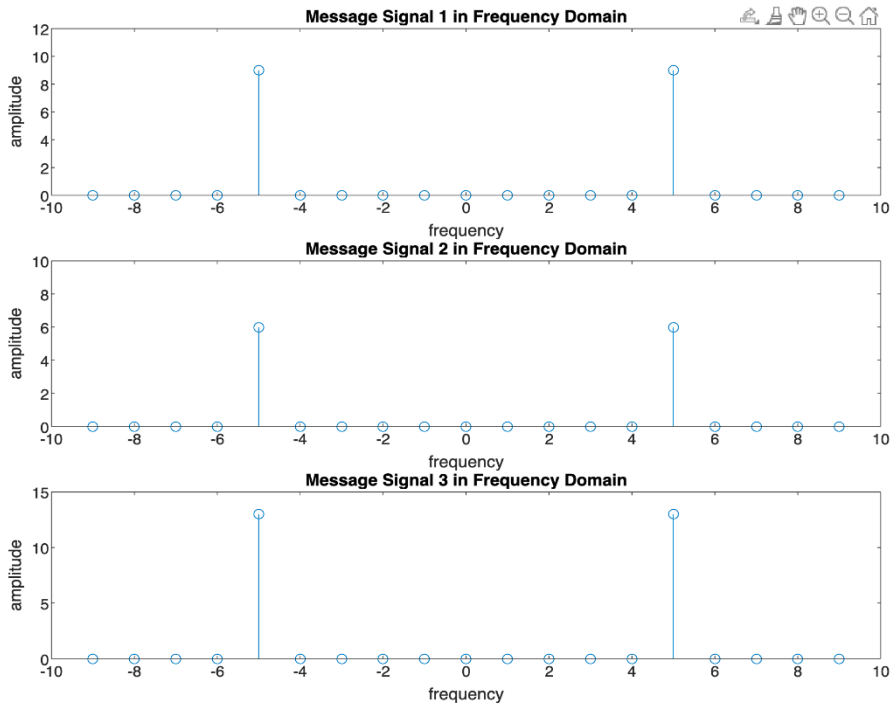

Fig 3: Message Signals in Frequency Domain

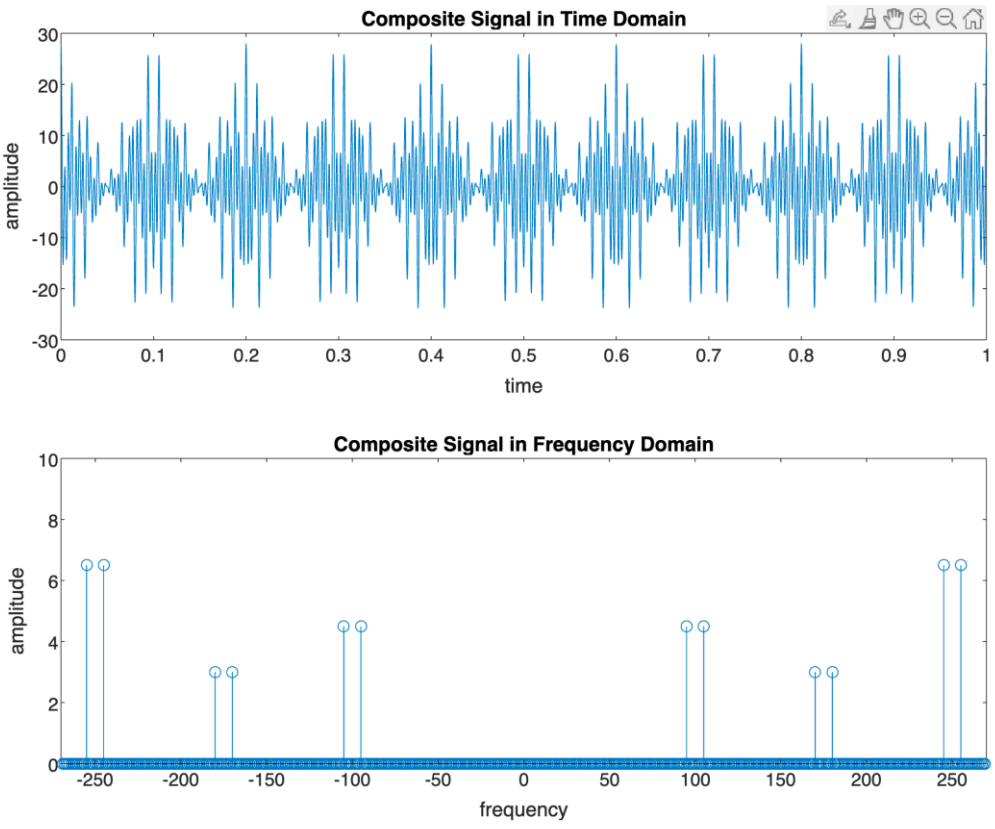

Fig 4: Composite Signal in Frequency Domain

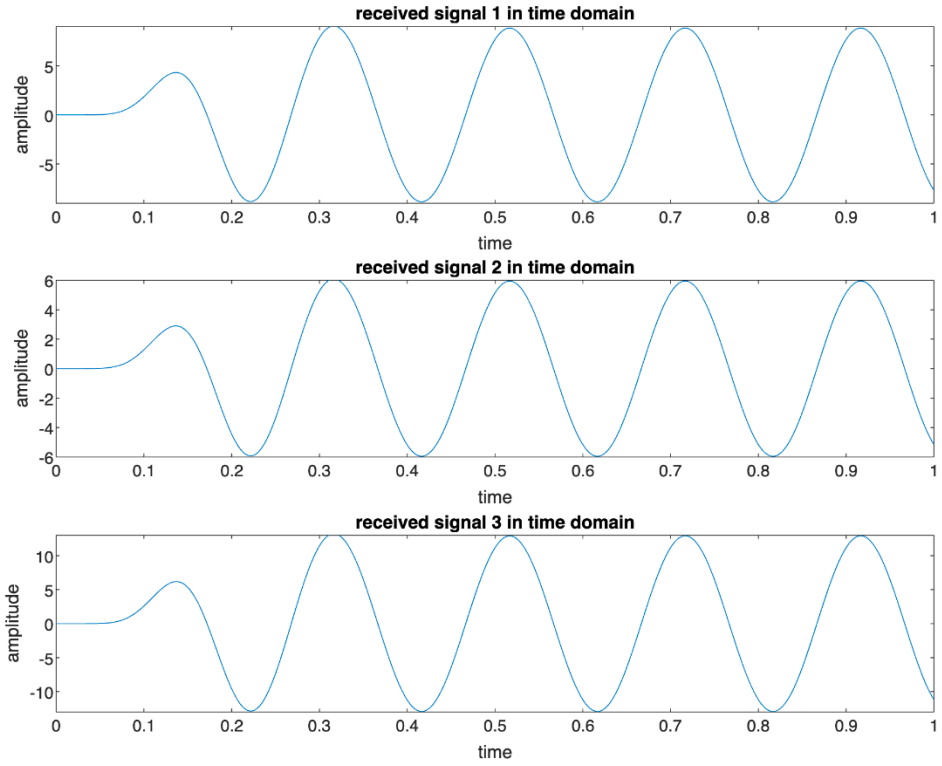

Fig 5: Received Signals in Time Domain

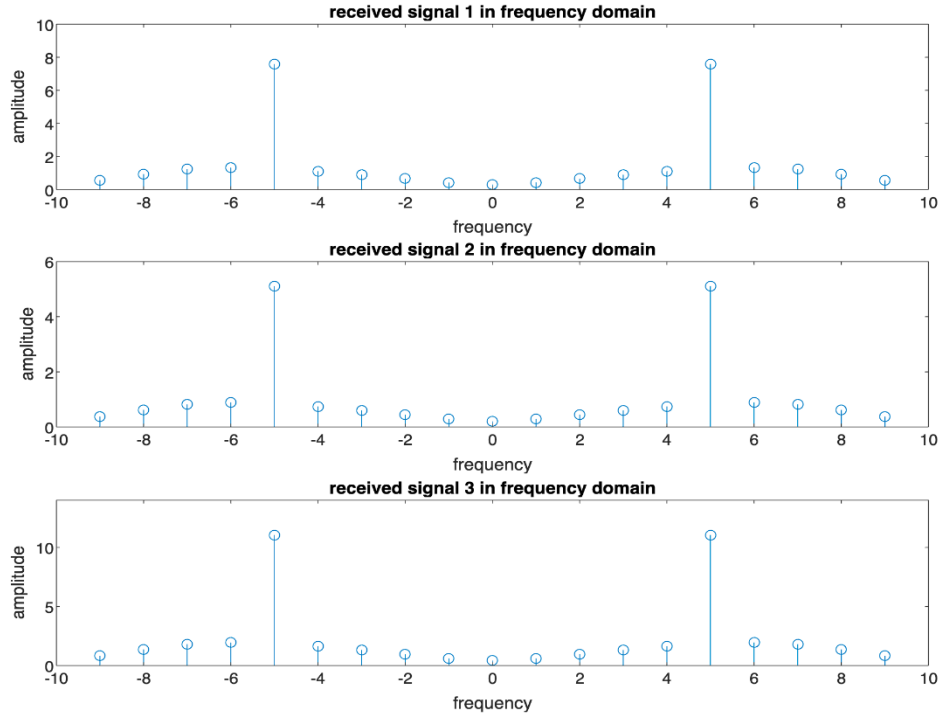

Fig 6: Received Signals in Frequency Domain

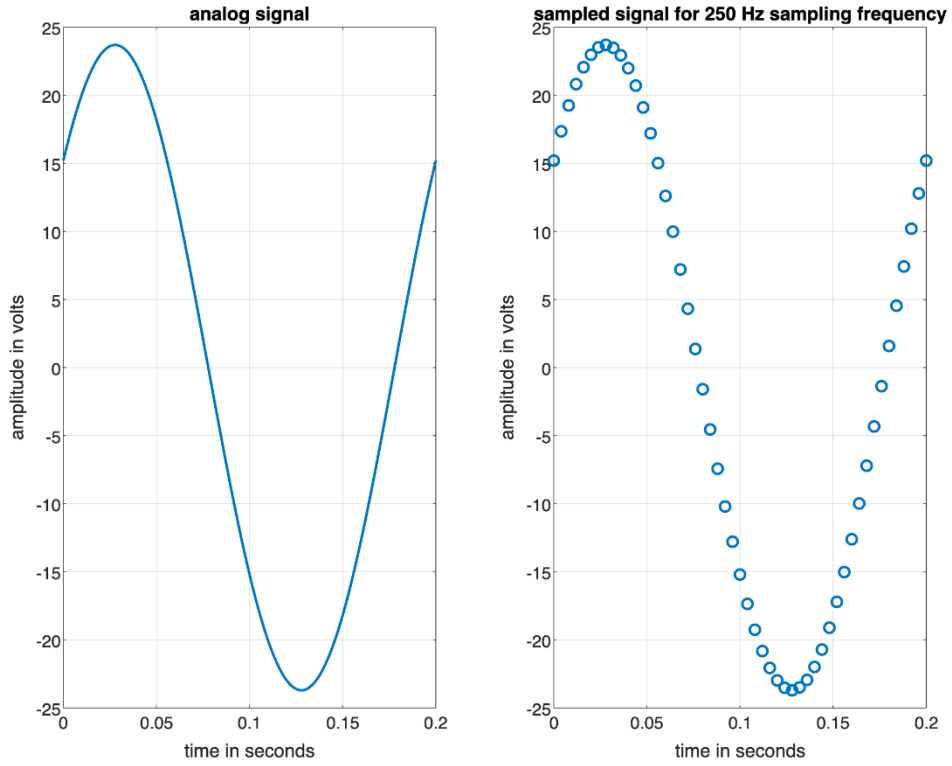

Fig 7: Sampled Signal

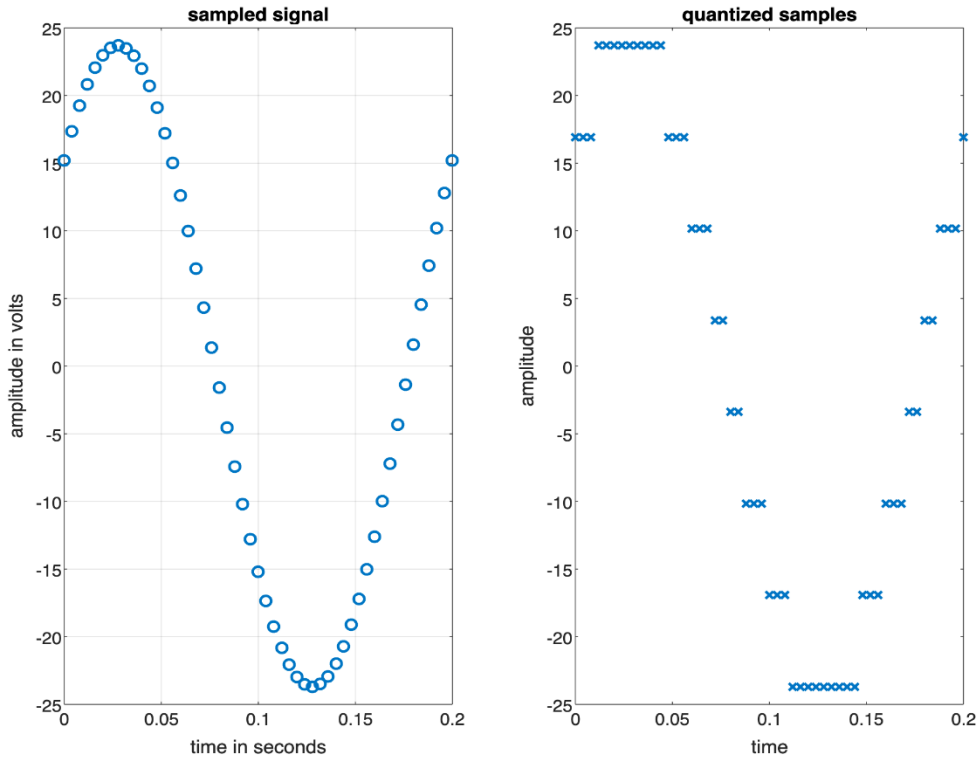

Fig 8: Quantized Samples

Impact on Society, health, and safety of:

ASK (Amplitude Shift Keying):

The modulated data is digital with an analog carrier signal when using the Amplitude Shift Keying technique. Processes for ASK modulation and demodulation are relatively inexpensive. Every digital communication channel uses ASK, including cable TV and cell phone. Most wireless links, including those used for satellite TV and high-definition TV broadcast channels, use two ASK links running in parallel. Using only one ASK channel and ignoring the phase is common in fiber optics. It is an effective technique to increase the input amplitude characteristics in communications. However, these ASK-modulated waveforms are easily affected by noise. Moreover, this leads to amplitude variations. Due to this, there will be voltage fluctuations in the output waveforms.

ASK technique is not suitable for high-bit-rate data transmission. It has poor bandwidth efficiency. ASK techniques are not suitable for high-bit-rate data transmission.

Applications for amplitude shift keying include:

- Low-frequency RF equipment;
- Home automation systems;
- Equipment for industrial networks.
- Wireless base stations
- Monitoring of tire pressure

FDM (Frequency Division Multiplexing):

Frequency division multiplexing (FDM) is a technique that divides the available bandwidth into multiple sub-bands, each of which can carry a signal. Using the FDM system, multimedia data can be transferred with high efficiency and low noise and distortion.

- FDM is commonly used in TV networks, telephone, broadband connections, stereo Frequency Modulation or FM transmission systems, and Radio broadcasting, also used in Amplitude Modulation or AM radio transmission systems.
- Frequency Division Multiplexing System is also used in the Satellite Communication system. It helps to transmit multiple channels of data in satellite communication.
- Frequency Division Multiplexing or FDM system is used for multimedia data such as video, audio, and image transmission.

Discussion and Conclusion:

Signal conversion is required for data transmission—sending data from one location to another. For this, it is necessary to convert the digital signal into an analog signal before converting it back into a digital signal.

In an experiment, shift keying (ASK) and multiplexing (FDM) were used to transmit binary bits as analog signals using MATLAB. Initially used Binary digital data to create analog signals. After that, signals were demultiplexed and converted back into binary bits at the receiver. The final result differed from the input result because choose the frequency arbitrarily.

Reference:

[1] Prakash C. Gupta, "Data communications," Prentice Hall India Pvt.
[2] William Stallings, "Data and Computer Communications," Pearson
[3] Forouzan, B. A. "Data Communication and Networking. Tata McGraw." (2005).