American International University- Bangladesh

Course Name: Data Communication Laboratory

Experiment No: 08

Experiment title: Frequency Division Multiplexing using MATLAB

Section: D

Semester: Summer 23-24

Name: Tutul Majumder

ID:23-51364-1

Performance Task for Lab Report: (My ID = 23-51364-1)

You have four message signals:
a) mt1 = am1*cos(2*pi*fm1*t);
b) mt2 = am2*cos(2*pi*fm2*t);
c) mt3 = am3*cos(2*pi*fm3*t);
d) mt4 = am4*cos(2*pi*fm4*t);
where,
am1 = (F+2);
am2 = (F+5);
am3 = (F+8);
am4 = (F+11);
and
fm1 = (G+1);
fm2 = (G+2);
fm3 = (G+3);
fm4 = (G+4);
We want to simultaneously transmit these four signals through a single data link that can support a frequency range of 50 Hz to 250 Hz.
*** Write a code that can modulate and multiplex the four given message signals in transmitting side (use appropriate carrier signals for amplitude modulation as required) and de-multiplex (use appropriate cut-off frequencies in your bandpass filters) and de-modulate (use appropriate cut-off frequencies in your lowpass filters) to recover the four message signals in receiving side.

Code:

```
A=2;
B=3;
C=5;
D=1;
E=3;
F=6;
G=4;
H=1;
Am1 = (F+2);
Am2 = (F+5);
Am3 = (F+8);
Am4 = (F+11);
fm1 = (G+1);
fm2 = (G+2);
```

```
fm3 = (G+3);
fm4 = (G+4);
%% Message Signal Generation
fs = 4001; %Sampling Frequency
t = 0:1/fs:1-1/fs; %Generating Time axis
m1 = Am1*cos(2*pi*fm1*t); % First Message Signal
m2 = Am2*cos(2*pi*fm2*t); % Second Message Signal
m3 = Am3*cos(2*pi*fm3*t); % Third Message Signal
m4 = Am4*cos(2*pi*fm4*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 = 150; %Frequency of Second Carrier Signal
c2 = Cm2*cos(2*pi*fc2*t); % Second Carrier Signal
Cm3 = 1; %Amplitude of Third Carrier Signal
fc3 = 200; %Frequency of Third Carrier Signal
c3 = Cm3*cos(2*pi*fc3*t); % Third Carrier Signal
Cm4=1;
fc4 = 250;
c4 = Cm4*cos(2*pi*fc4*t); % Third Carrier Signal
%%
%% Composite Signal Generation
x = (m1).*c1+(m2).*c2+(m3).*c3+(m4).*c4;
%%%% Plotting the Signals in Time-Domain and Frequency-Domain
figure
subplot(4,1,1)
plot(t,m1)
xlabel('time')
ylabel('amplitude')
title('Message Signal 1 in Time Domain')
ylim([-Am1 Am1])
subplot(4,1,2)
plot(t,m2)
xlabel('time')
ylabel('amplitude')
title('Message Signal 2 in Time Domain')
ylim([-Am2 Am2])
subplot(4,1,3)
plot(t,m3)
xlabel('time')
ylabel('amplitude')
title('Message Signal 3 in Time Domain')
ylim([-Am3 Am3])
```

```
subplot(4,1,4)
plot(t,m4)
xlabel('time')
ylabel('amplitude')
title('Message Signal 4 in Time Domain')
ylim([-Am4 Am4])
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
M4 = abs(fftshift(fft(m4)))/(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(4,1,1)
stem(f,M1)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 1 in Frequency Domain')
axis([-10 10 0 2.5])
subplot(4,1,2)
stem(f,M2)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 2 in Frequency Domain')
axis([-10 10 0 3.5])
subplot(4,1,3)
stem(f,M3)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 3 in Frequency Domain')
axis([-10 10 0 4.5])
subplot(4,1,4)
stem(f,M4)
xlabel('frequency')
ylabel('amplitude')
title('Message Signal 4 in Frequency Domain')
axis([-10 10 0 5.5])
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 2.5])
%%
%% 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
%% Passing the Composite Signal Through Bandpass Filter
[num4, den4] = butter(5, [(fc4-fm4-6)/(fs/2),(fc4+fm4+6)/(fs/2)]);
%Butterworth Filter Window Determining for Bandpass Filter
bpf4 = filter(num4,den4,x); %Filtering is done here
%%
%% Mixing
z1 = 2*bpf1.*c1;
z2 = 2*bpf2.*c2;
z3 = 2*bpf3.*c3;
z4 = 2*bpf4.*c4;
%%
%% Passing the Mixed Signals Through Lowpass Filter
[num5, den5] = butter(5, (fm1+3)/(fs/2)); %Low pass filter is made here
rec1 = filter(num5,den5,z1); %Filtering is done here
[num6, den6] = butter(5, (fm2+3)/(fs/2)); %Low pass filter is made here
rec2 = filter(num5,den6,z2); %Filtering is done here
[num7, den7] = butter(5, (fm3+3)/(fs/2)); %Low pass filter is made here
rec3 = filter(num7,den7,z3); %Filtering is done here
[num8, den8] = butter(5, (fm4+3)/(fs/2)); %Low pass filter is made here
rec4 = filter(num8,den8,z4); %Filtering is done here
%%
%% Plotting the Received Signals in Time-Domain and Frequency Domain
figure
subplot(4,1,1)
plot(t,rec1)
xlabel('time')
ylabel('amplitude')
title('received signal 1 in time domain')
ylim([-Am1 Am1])
subplot(4,1,2)
plot(t,rec2)
xlabel('time')
```

```
ylabel('amplitude')
title('received signal 2 in time domain')
ylim([-Am2 Am2])
subplot(4,1,3)
plot(t,rec3)
xlabel('time')
ylabel('amplitude')
title('received signal 3 in time domain')
ylim([-Am3 Am3])
subplot(4,1,4)
plot(t,rec4)
xlabel('time')
ylabel('amplitude')
title('received signal 4 in time domain')
ylim([-Am4 Am4])
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
R4 = abs(fftshift(fft(rec4)))/(fs/2); %Fourier Transformation is done here
figure
subplot(4,1,1)
stem(f,R1)
xlabel('frequency')
ylabel('amplitude')
title('received signal 1 in frequency domain')
xlim([-10 10])
subplot(4,1,2)
stem(f,R2)
xlabel('frequency')
ylabel('amplitude')
title('received signal 2 in frequency domain')
xlim([-10 10])
subplot(4,1,3)
stem(f,R3)
xlabel('frequency')
ylabel('amplitude')
title('received signal 3 in frequency domain')
xlim([-10 10])
subplot(4,1,4)
stem(f,R4)
xlabel('frequency')
ylabel('amplitude')
title('received signal 4 in frequency domain')
xlim([-10 10])
```

Output:

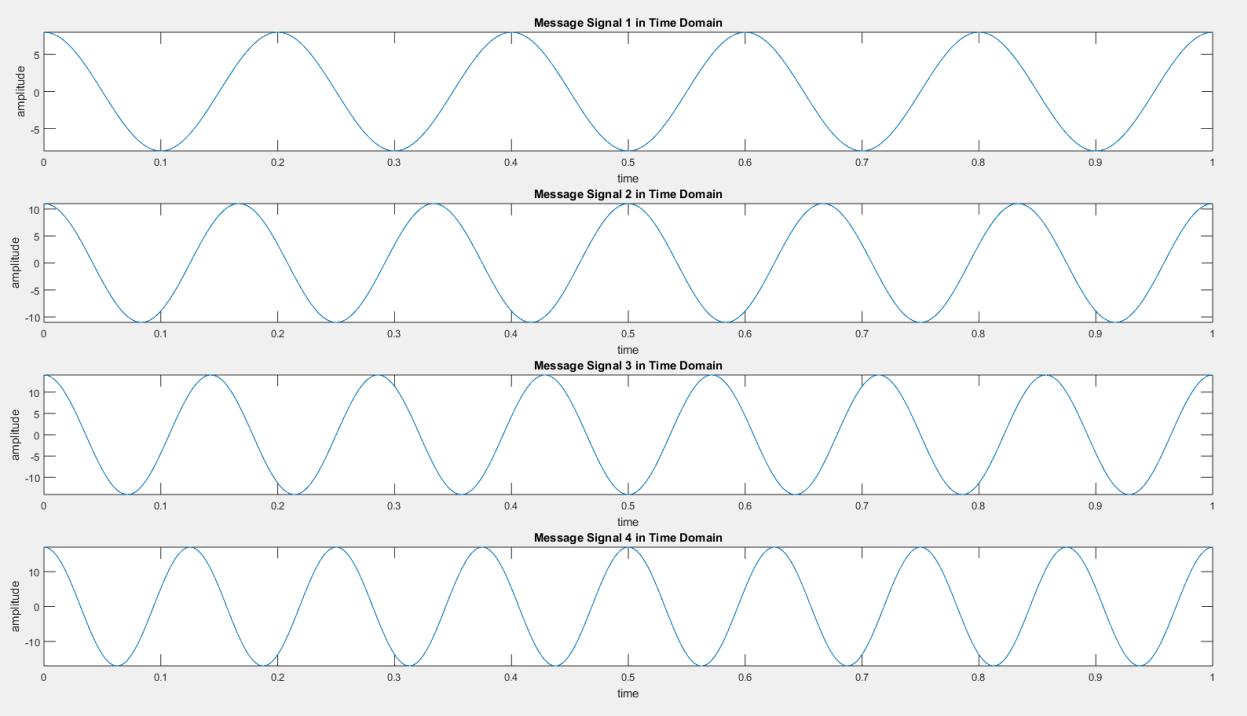

Fig 01

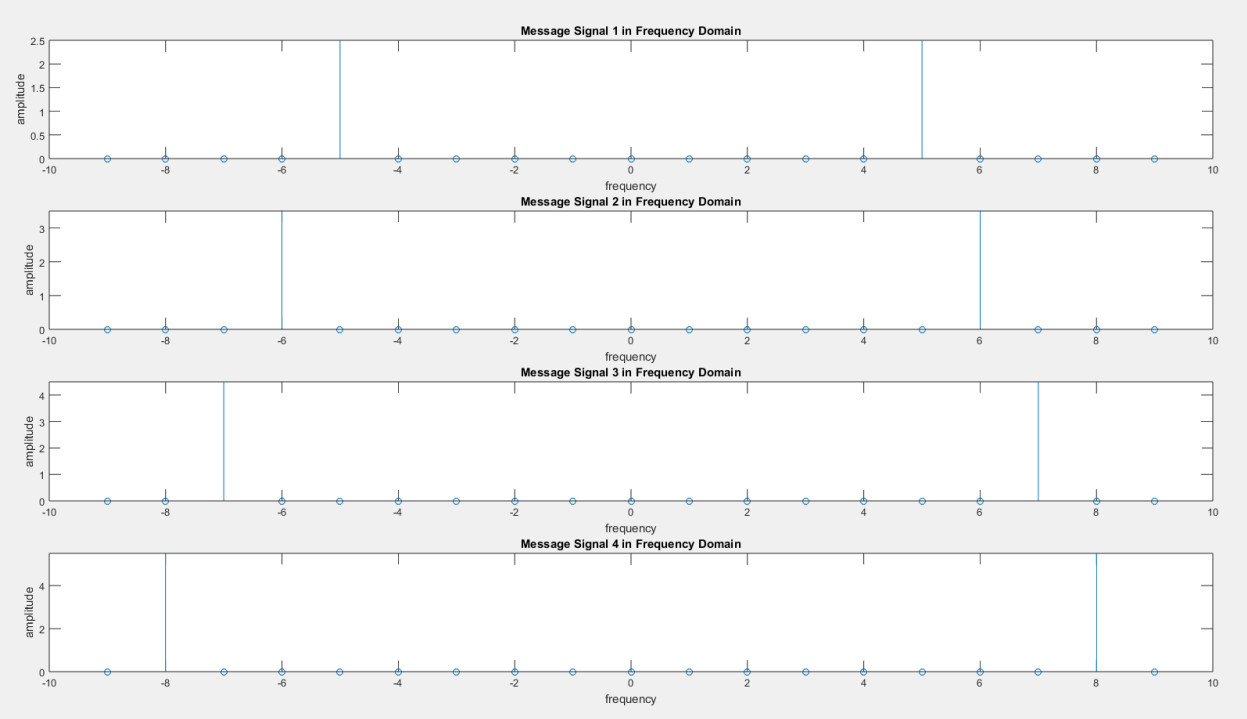

Fig 02

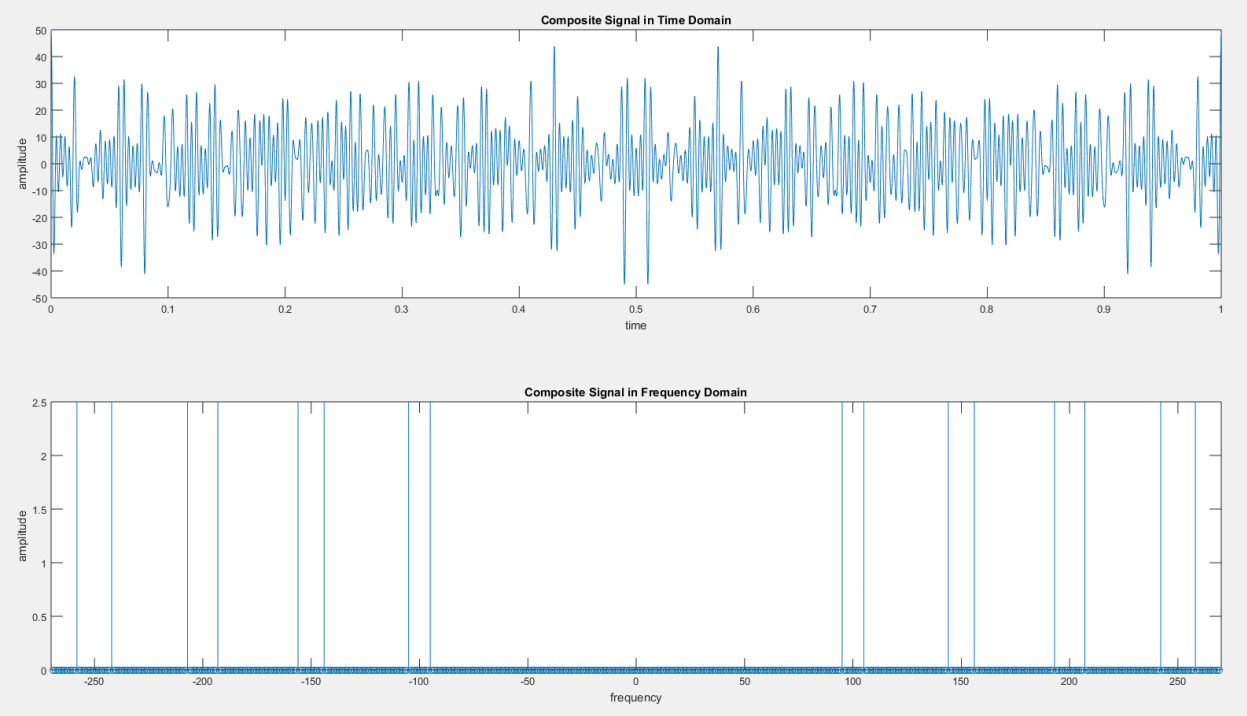
Composite Signal in Time Domain
amplitude
time
Composite Signal in Frequency Domain
amplitude
frequency

Fig 03

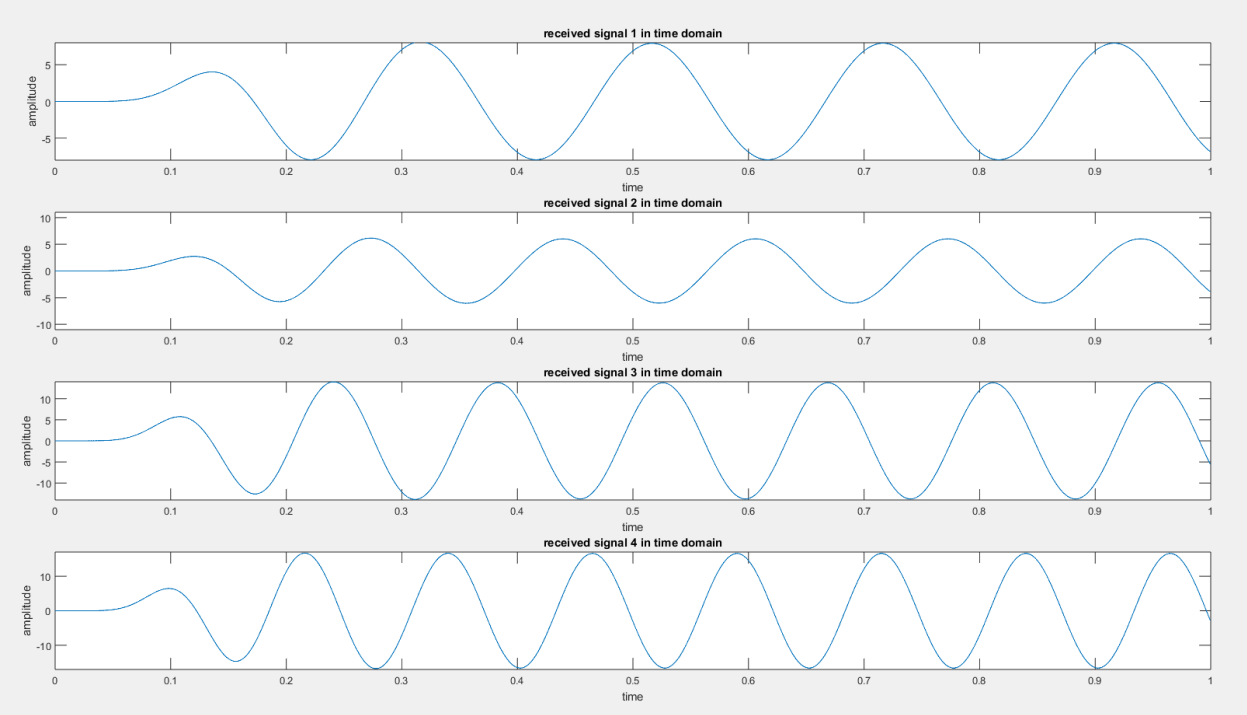
received signal 1 in time domain
amplitude
time
received signal 2 in time domain
amplitude
time
received signal 3 in time domain
amplitude
time
received signal 4 in time domain
amplitude
time

Fig 04

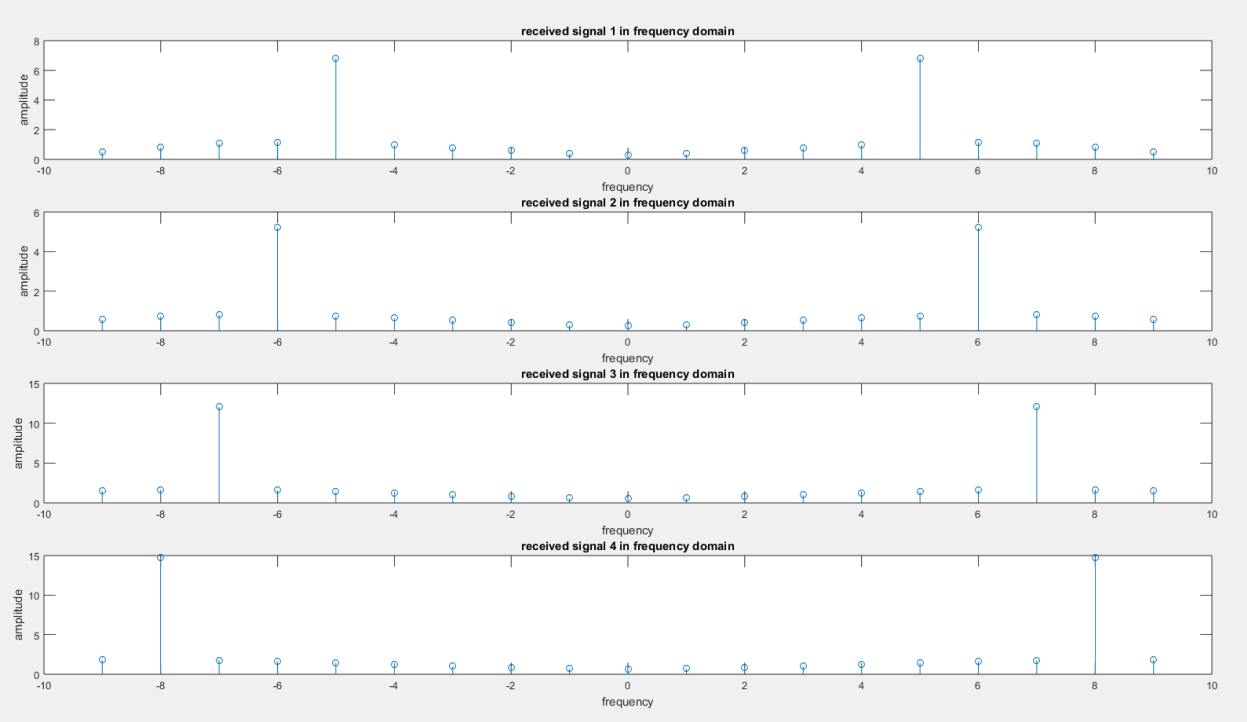
received signal 1 in frequency domain
amplitude
frequency
received signal 2 in frequency domain
amplitude
frequency
received signal 3 in frequency domain
amplitude
frequency
received signal 4 in frequency domain
amplitude
frequency

Fig 05