LAB#08

***NAME** : ABDULLAH ZUNORAIN*

***REG NO** : 19JZELE0338*

***SUBJECT** : DATA COMMUNICATION(LAB)*

***SUBMITTED TO** : DR. UZAIR GILLANI*

***SECTION** : A*

***DEPT** : ELECTRICAL COMM*

***CAMPUS** : JALOZAI*

Analog Signal Generation using Digital Data in Matlab

OBJECTIVES OF THE LAB

In this lab, we will cover the following topics:

- Understand matlab program for ASK & BFSK
- and test it for different user inputs

7.1 AMPLITUDE SHIFT KEYING (ASK)

In ASK, the two binary values are represented by two different amplitudes of the carrier frequency. Commonly, one of the amplitudes is zero, i.e. one binary digit is represented by the presence, at constant amplitude, of the carrier; while the other is represented by the absence of the carrier. The resulting transmitted signal for one bit time is

$$t(\ ) = \begin{cases} A\cos(2\pi f_c t)\ s & ,binary\ 1 \\ 0 & ,binary\ 0 \end{cases}$$

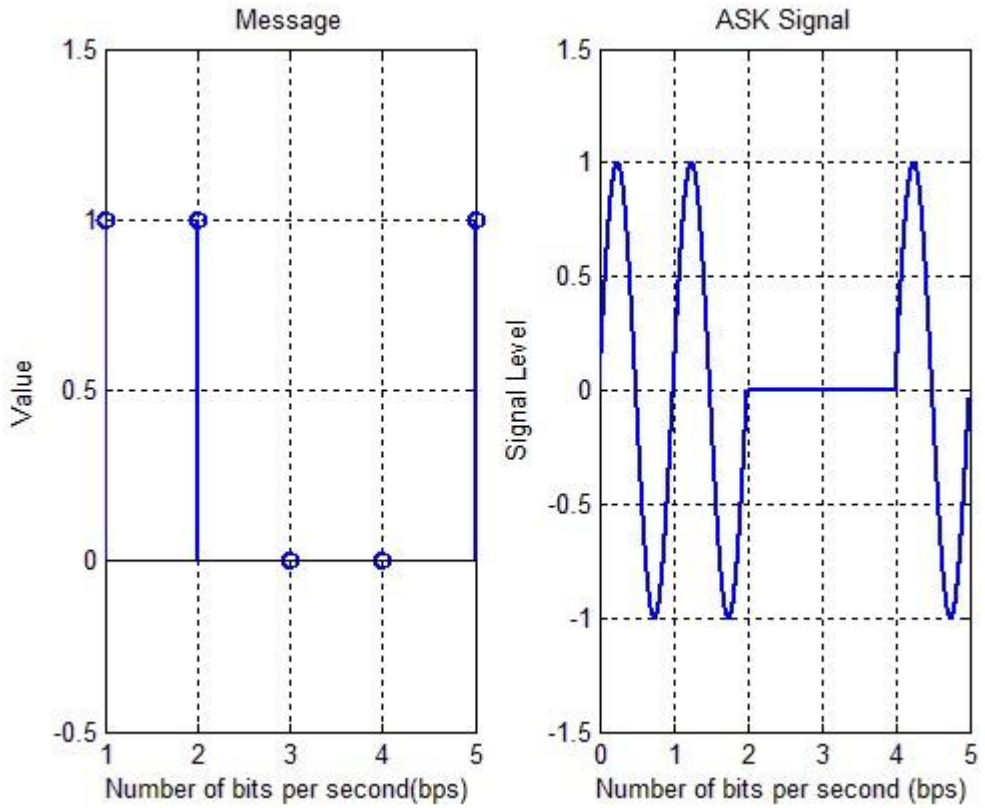
where carrier signal is $A\cos(2\pi f_c t)$.

Matlab Code for Amplitude Shift Keying (ASK):

```
clc clear
all

% taking input message from user
message = input('Enter binary data with spaces eg: [1 1 0 0 1] : ');
f_one = 1;                          % frequency of sinusoid for binary-1
fs = 1000;                          % sampling frequency
t = [0:1/fs:1];                     % time duration for 1-pulse
x = sin(2*pi*f_one*t);              % sinusoid for binary-1
y = zeros(1,length(t));             % Zero level signal for binary-0
% generating timing vector tim
= t;
for(i=1:length(message)-1)
tim = [tim i+t]; end

% creating ASK signal
ask_sig = []; for
i=1:length(message)
  if(message(i) == 0)
    ask_sig = [ask_sig y];
  else  ask_sig = [ask_sig
    x];
  end end

% plotting ASK signal w.r.t timing vector
figure; subplot(121);
stem(message, 'linewidth', 2);
title('Message');
xlabel('Number of bits per second (bps)');
ylabel('Value');
```

```
axis([1 length(message) -0.5 1.5]);
grid on; subplot(122);
plot(tim(1,1:(length(message)*fs)),ask_sig(1,1:(length(message)*fs)),'linew
idth', 1.5); title('ASK Signal');
xlabel(' Number of bits per second (bps)'); ylabel('Signal
level');
axis([0 length(message) -1.5 1.5]); grid
on;
```

Output Signal:



----------TASK_01----------

Write matlab code for Amplitude Shift Keying (ASK) using the following message:

[0 0 1 1 1 0 0 1 0 1]

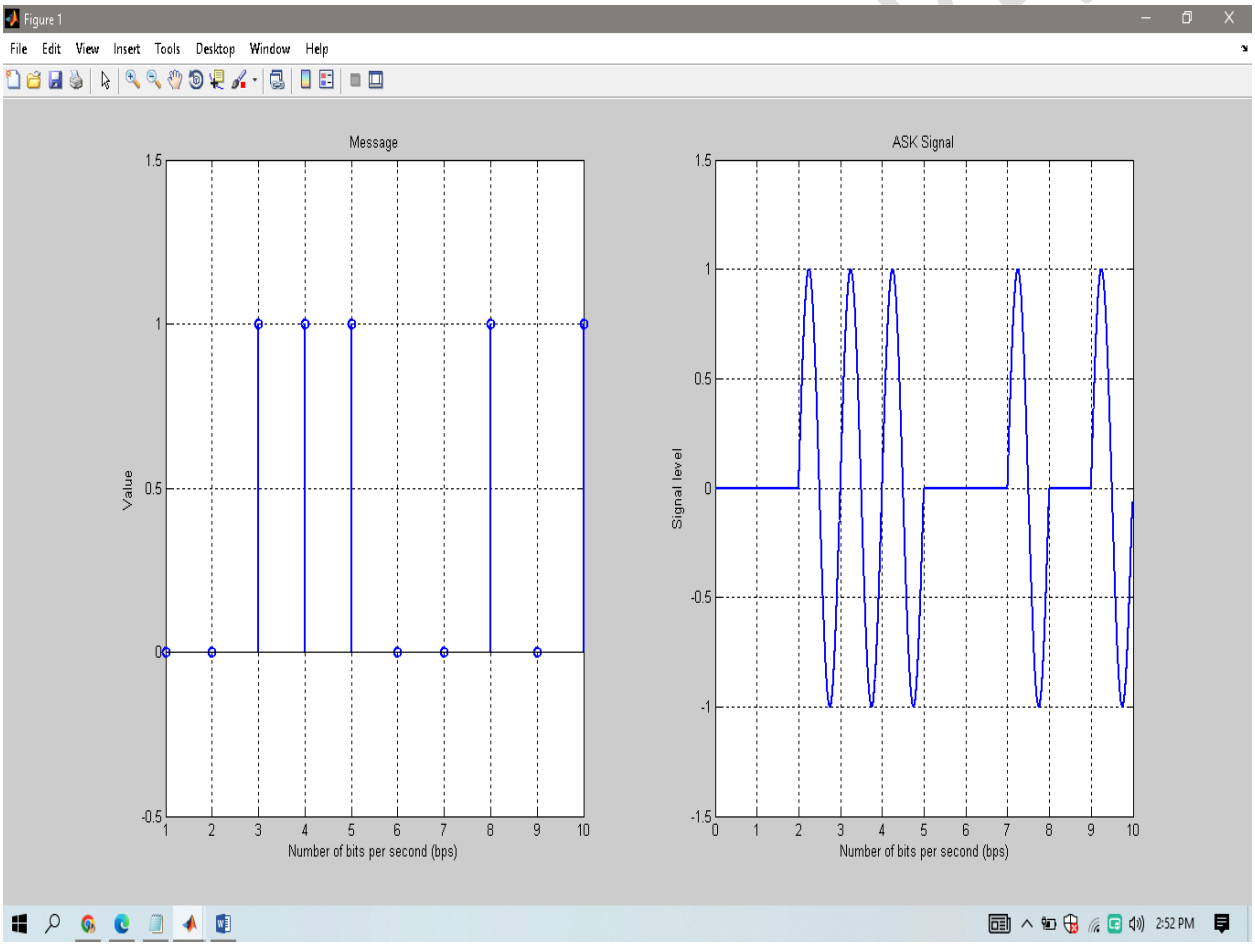
ANS:

CODING OF THE ABOVE TASK#01 FOR AMPLITUDE SHIFT KEYING:

```
% taking input message from user
message = [0 0 1 1 1 0 0 1 0 1];
f_one = 1;      % frequency of sinusoid for binary-1
fs = 1000;      % sampling frequency
t = [0:1/fs:1];           % time duration for 1-pulse
x = sin(2*pi*f_one*t);          % sinusoid for binary-1
y = zeros(1,length(t));         % Zero level signal for binary-0
% generating timing vector
tim = t;
for(i=1:length(message)-1)
tim = [tim i+t];
end
% creating ASK signal
ask_sig = [];
for i=1:length(message)
 if(message(i) == 0)
ask_sig = [ask_sig y];
 else
ask_sig = [ask_sig x];
    end
 end
% plotting ASK signal w.r.t timing vector
figure;
subplot(121);
stem(message, 'linewidth', 2);
title('Message');
xlabel('Number of bits per second (bps)');
ylabel('Value');
axis([1 length(message) -0.5 1.5]);
```

```
grid on;
subplot(122);
plot(tim(1,1:(length(message)*fs)),ask_sig(1,1:(length(message)*fs)),'linewidth', 1.5);
title('ASK Signal');
xlabel(' Number of bits per second (bps)');
ylabel('Signal level');
axis([0 length(message) -1.5 1.5]);
grid on;
```

RESULT OF THE CODING (OUTPUT SIGNAL):



7.2 BINARY FREQUENCY SHIFT KEYING (BFSK)

In BFSK, the two binary values are represented by two different frequencies near the carrier frequency. The resulting transmitted signal for one bit time is

$$t( ) = \begin{cases} A\cos(2\pi f\ t)\ s & ,binary\ \ 1 \\ 1 & ,binary\ \ 0 \\ A\cos(2\pi f\ t_2) & \end{cases}$$

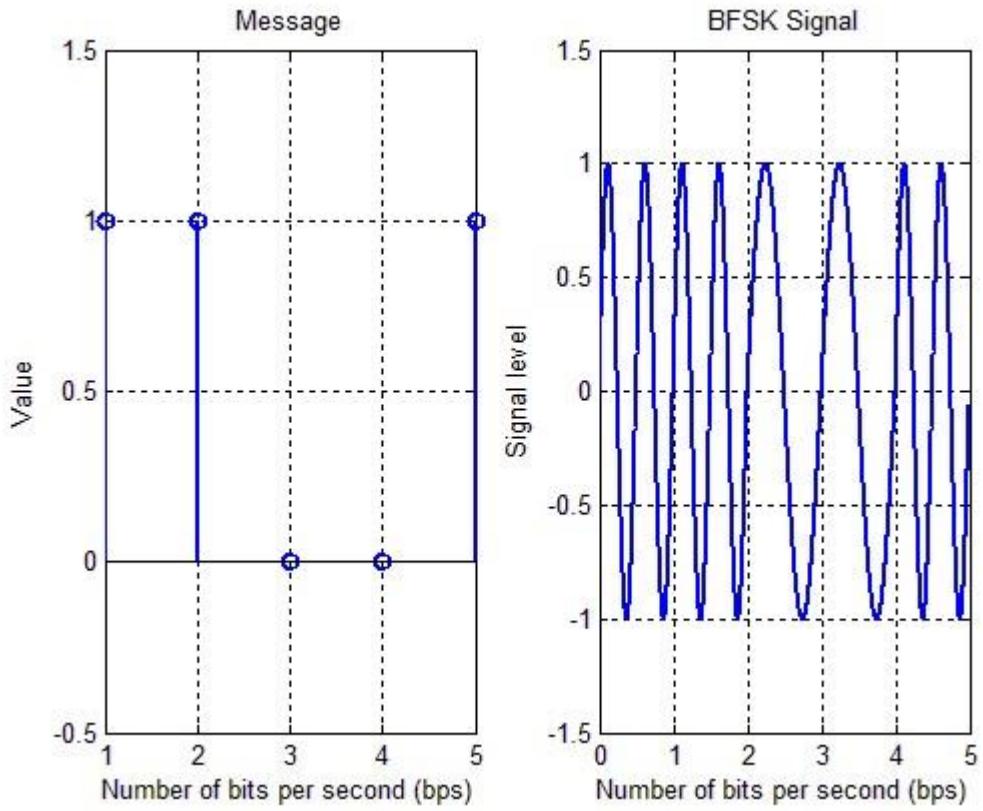
where f_1 and f_2 are typically offset from the carrier frequency f_c by equal but opposite amounts.

Matlab Code for Binary Frequency Shift Keying (BFSK):

```
clc clear
all

% taking input message from user
message = input('Enter binary data with spaces eg: [1 1 0 0 1] : ');
f1 = 1;                    % frequency for binary-0
f2 = 2;                    % frequency for binary-1
fs = 1000;                 % sampling frequency
t = [0:1/fs:1];            % time duration for 1-pulse
x = sin(2*pi*f2*t);        % sinusoid representing binary-1
y = sin(2*pi*f1*t);        % sinusoid representing binary-0
% generating timing vector tim
= t;
for(i=1:length(message)-1)
tim = [tim i+t]; end

% creating BFSK signal
fsk_sig = []; for
i=1:length(message)
  if(message(i) == 0)
    fsk_sig = [fsk_sig y];
  else  fsk_sig = [fsk_sig
    x];
  end end

% plotting BFSK signal w.r.t timing vector
figure; subplot(121);
stem(message, 'linewidth', 2); title('Message');
```

```
xlabel('Number of bits (bps)'); ylabel('Value');
axis([1 length(message) -0.5 1.5]);
grid on; subplot(122);
plot(tim(1,1:(length(message)*1000)),fsk_sig(1,1:(length(message)*1000))
, 'linewidth', 1.5); title('BFSK
Signal'); xlabel('Number of
bits (bps)'); ylabel('Signal
level');
axis([0 length(message) -1.5 1.5]); grid
on;
```

Output Signal:



----------TASK_02----------

Write matlab code that generates Binary Frequency Shift Keying (BFSK) signal using the following message:

[1 0 0 1 0 1 0 0 1]

ANS:

CODING FOR BINARY FREQUENCY SHIFT KEYING :

```
% taking input message from user
message = [1 0 0 1 0 1 0 0 1];
f1 = 1;          % frequency for binary-0
f2 = 2;          % frequency for binary-1
fs = 1000;       % sampling frequency
t = [0:1/fs:1];          % time duration for 1-pulse
x = sin(2*pi*f2*t);      % sinusoid representing binary-1
y = sin(2*pi*f1*t);      % sinusoid representing binary-0
% generating timing vector
tim = t;
for(i=1:length(message)-1)
tim = [tim i+t];
end
% creating BFSK signal
fsk_sig = [ ];
for i=1:length(message)
if(message(i) == 0)
fsk_sig = [fsk_sig y];
else
fsk_sig = [fsk_sig x];
end
end
% plotting BFSK signal w.r.t timing vector
```

```
figure;
subplot(121);
stem(message, 'linewidth', 2);
title('Message');
xlabel('Number of bits (bps)');
ylabel('Value');
axis([1 length(message) -0.5 1.5]);
grid on;
subplot(122);
plot(tim(1,1:(length(message)*1000)),fsk_sig(1,1:(length(message)*1000)),'linewidth',1.5
);
title('BFSK Signal');
xlabel('Number of bits (bps)');
ylabel('Signal level');
axis([0 length(message) -1.5 1.5]);
grid on;
```

RESULT OF THE CODING (OUTPUT SIGNAL):