AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

DATA COMMUNICATION LABORATORY

Fall 2023-2024

Section: I

Group: 4

EXPERIMENT NO: 6

Study of Digital to Analog Conversion using MATLAB

Submitted By:

Name	ID
1. MD. SHAHRIAR PARVEZ SHAMIM	21-44998-2
2. MD. AL FAIAZ RAHMAN FAHIM	21-45080-2
3. MD. OMAR FARUK SAKIB	21-45077-2
4. MD. ABU HOJIFA	21-45081-2
5. ASHFAT AHMAD MEDUL	21-44854-2

Date of Submission: December 20, 2023

Abstract:

In this experiment, we focused on utilizing MATLAB for solving communication engineering problems and developing an understanding of digital-to-analog conversion using MATLAB. The experiment involved designing a digital-to-analog converter using MATLAB and analyzing its performance. This provided us with a deeper understanding of the importance of digital-to-analog conversion in communication systems and how to implement it using MATLAB.

Theory:

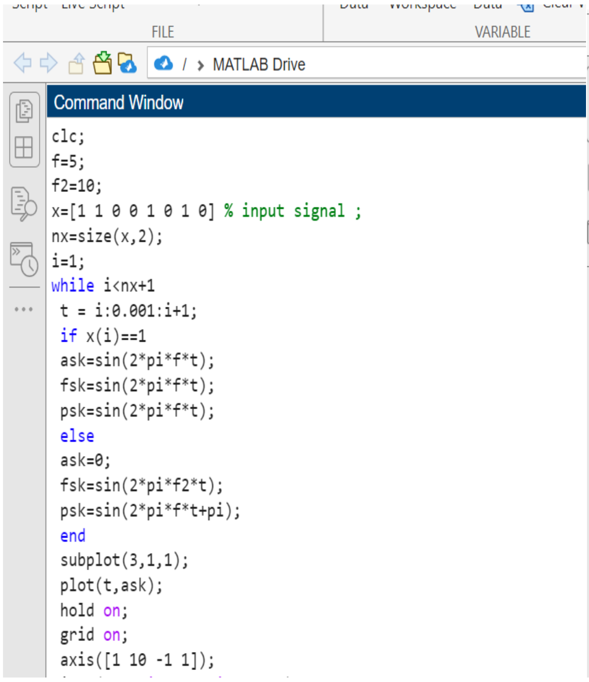
DIGITAL TO ANALOG CONVERSION: Digital to analog conversion is the process of changing one of the characteristics of an analog signal based on the information in digital data Figure bellow shows the relationship between the digital information, the digital to analog modulating process, and the resultant analog signal.

ASK: 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.

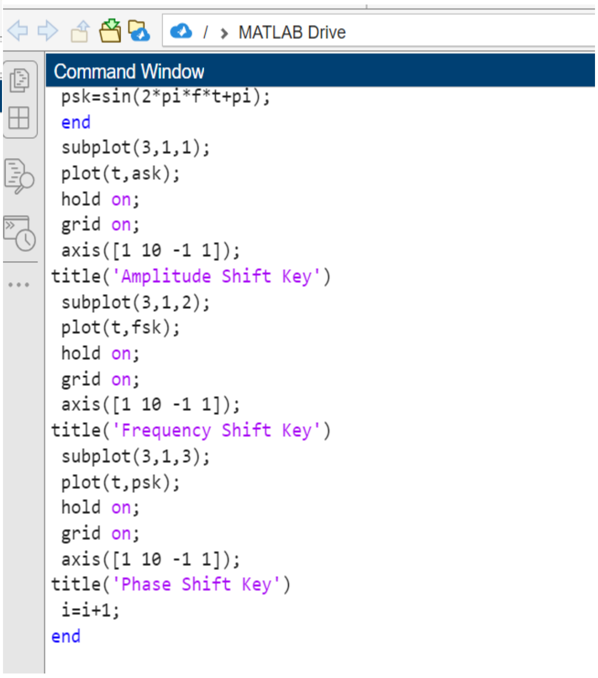
FSK: In frequency shift keying, the frequency of the carrier signal is varied to represent data The frequency of the modulated signal is constant for the duration of one signal element, but changes for the next signal element if the data element changes Both peak amplitude and phase remain constant for all signal elements.

PSK: In phase shift keying, the phase of the carrier is varied to represent two or more different signal elements Both peak amplitude and frequency remain constant as the phase changes Today, PSK is more common than ASK or FSK. However, we will see shortly that QAM, which combines ASK and PSK, is the dominant method of digital to analog modulation.

Results:




```
clc;
f=5;
f2=10;
x=[1 1 0 0 1 0 1 0] % input signal ;
nx=size(x,2);
i=1;
while i<nx+1
 t = i:0.001:i+1;
 if x(i)==1
 ask=sin(2*pi*f*t);
 fsk=sin(2*pi*f*t);
 psk=sin(2*pi*f*t);
 else
 ask=0;
 fsk=sin(2*pi*f2*t);
 psk=sin(2*pi*f*t+pi);
 end
 subplot(3,1,1);
 plot(t,ask);
 hold on;
 grid on;
 axis([1 10 -1 1]);
```




```
 psk=sin(2*pi*f*t+pi);
 end
 subplot(3,1,1);
 plot(t,ask);
 hold on;
 grid on;
 axis([1 10 -1 1]);
title('Amplitude Shift Key')
 subplot(3,1,2);
 plot(t,fsk);
 hold on;
 grid on;
 axis([1 10 -1 1]);
title('Frequency Shift Key')
 subplot(3,1,3);
 plot(t,psk);
 hold on;
 grid on;
 axis([1 10 -1 1]);
title('Phase Shift Key')
 i=i+1;
end
```

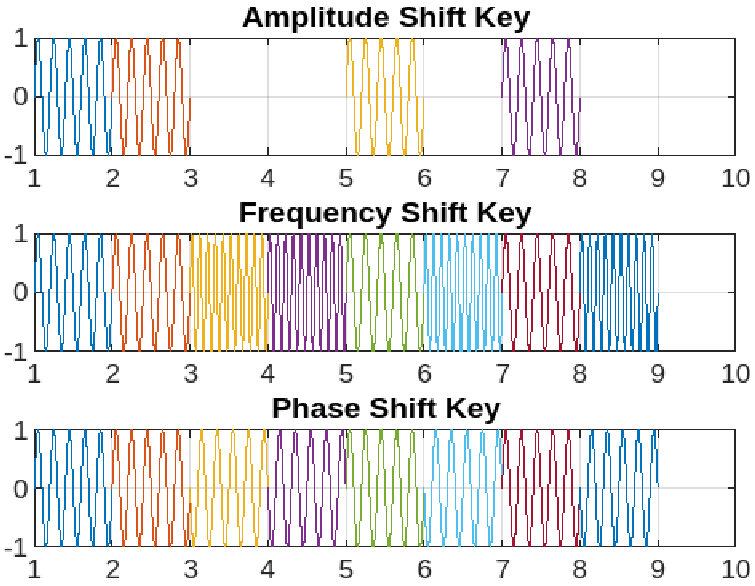


Figure 1: ASK, FSK and PSK

```
clc;
f=10;
x=[00 10 01 11] % input signal ;
x1=[0 1 0 1];
x2=[0 0 1 1];
nx=size(x1,2);
i=1;
while i<nx+1
 t = i:0.001:i+1;
 if x1(i)==1
 psk1=sin(2*pi*f*t);
 else
 psk1=sin(2*pi*f*t+pi);
 end

 if x2(i)==1
 psk2=sin(2*pi*f*t+pi/2);
 else
 psk2=sin(2*pi*f*t+pi+pi/2);
 end

 QPSK = psk1+psk2;
```

```
 subplot(3,1,1);
 plot(t,psk1);
 hold on;
 grid on;
 axis([1 4 -1 1]);
 title('PSK1')

 subplot(3,1,2);
 plot(t,psk2);
 hold on;
 grid on;
axis([1 4 -1 1]);
 title('PSK2')
 subplot(3,1,3);
 plot(t,QPSK);
 hold on;
 grid on;
 axis([1 4 -2 2]);
 title('QPSK')
 i=i+1;
end
```

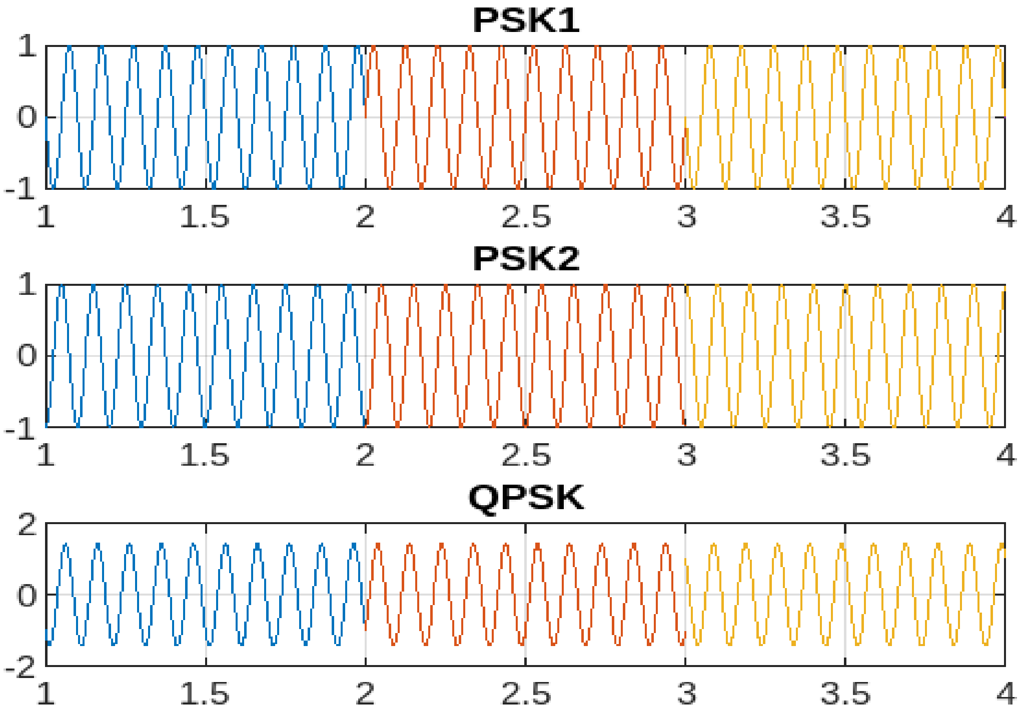


Figure 2: QPSK

Performance Task:

ID = AB-CDEFG-H **ID** = 21-45077-2	E=0 F=7 G=7	Bit Stream = 000001110000100100000011

1.

Command Window

```
>> %21-45077-2
%E = 0 1 0 0 0 1 0 1
%F = 0 1 0 0 0 1 1 0
%G = 0 1 0 0 0 1 1 1
%Bit Stream = 010001010100011001000111
f=5;
f2=10;
x = [ 010 001 010 100 011 001 000 111]; % input signal ;
nx=size(x,2);
i=1;
while i<nx+1
t = i:0.001:i+1;
if x(i)==1
ask=sin(2*pi*f*t);
elseif x(i)==010
ask=2*sin(2*pi*f*t);
elseif x(i)==011
ask=3*sin(2*pi*f*t);
elseif x(i)==100
ask=4*sin(2*pi*f*t);
elseif x(i)==101
ask=5*sin(2*pi*f*t);
```

Command Window

```
ask=2*sin(2*pi*f*t);
elseif x(i)==011
ask=3*sin(2*pi*f*t);
elseif x(i)==100
ask=4*sin(2*pi*f*t);
elseif x(i)==101
ask=5*sin(2*pi*f*t);
elseif x(i)==110
ask=6*sin(2*pi*f*t);
elseif x(i)==111
ask=7*sin(2*pi*f*t);
else
ask=0;
end
subplot(1,1,1);
plot(t,ask);
hold on;
grid on;
axis([1 10 -10 10]);
title('Amplitude Shift')
i=i+1;
end
>>
```

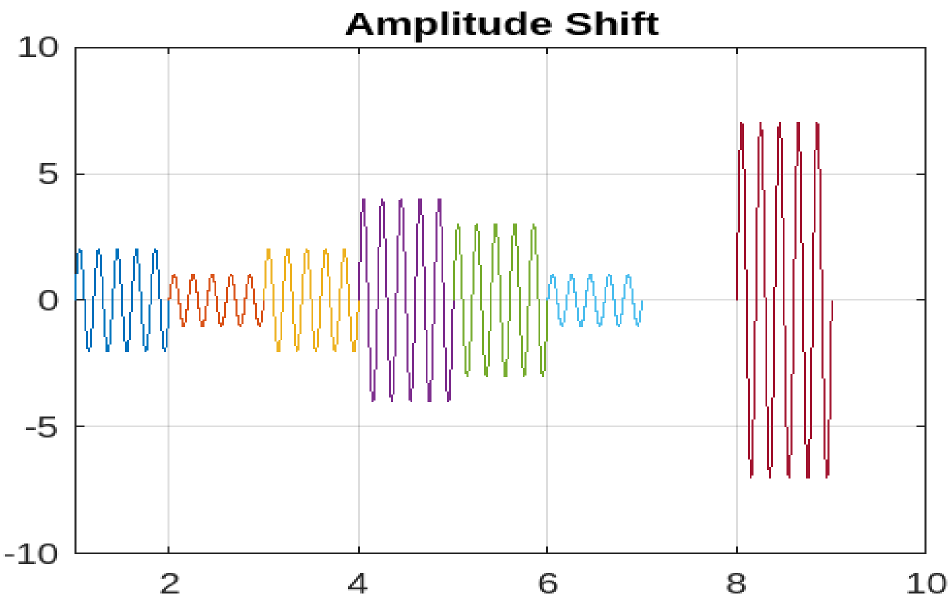


Figure 1: Performance Task 01(Amplitude Shift)

Command Window

```
>> f=5;
f2=10;
x = [010 001 010 100 011 001 000 111]; % input signal ;
nx=size(x,2);
i=1;
while i<nx+1
t = i:0.001:i+1;
if x(i)==1
fsk=sin(2*pi*2*f*t);
elseif x(i)==010
fsk=sin(2*pi*3*f*t);
elseif x(i)==011
fsk=sin(2*pi*4*f*t);
elseif x(i)==100
fsk=sin(2*pi*5*f*t);
elseif x(i)==101
fsk=sin(2*pi*6*f*t);
elseif x(i)==110
fsk=6*sin(2*pi*7*f*t);
elseif x(i)==111
fsk=sin(2*pi*8*f*t);
else
fsk=sin(2*pi*f2*t);
```

Command Window

```
elseif x(i)==010
fsk=sin(2*pi*3*f*t);
elseif x(i)==011
fsk=sin(2*pi*4*f*t);
elseif x(i)==100
fsk=sin(2*pi*5*f*t);
elseif x(i)==101
fsk=sin(2*pi*6*f*t);
elseif x(i)==110
fsk=6*sin(2*pi*7*f*t);
elseif x(i)==111
fsk=sin(2*pi*8*f*t);
else
fsk=sin(2*pi*f2*t);
end
subplot(1,1,1);
plot(t,fsk);
hold on;
grid on;
axis([1 10 -1 1]);
title('Frequency Shift ')
i=i+1;
end
```

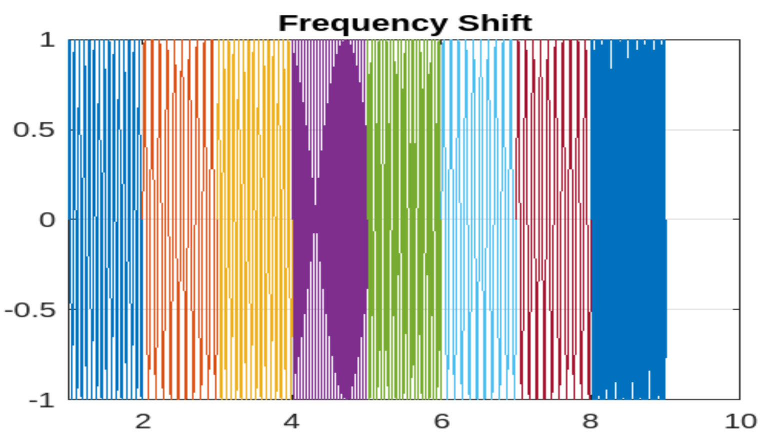


Figure 2: Performance Task 02 (Frequency Shift)

Command Window

```
>> f=5;
f2=10;
x = [010 001 010 100 011 001 000 111]; % input signal ;
nx=size(x,2);
i=1;
while i<nx+1
t = i:0.001:i+1;
if x(i)==1
psk=sin(2*pi*f*t);
elseif x(i)==010
psk=sin(2*pi*f*t+pi/4);
elseif x(i)==011
psk=sin(2*pi*f*t+3*pi/4);
elseif x(i)==100
psk=sin(2*pi*f*t+pi/2);
elseif x(i)==101
psk=sin(2*pi*6*f*t-pi/4);
elseif x(i)==110
psk=6*sin(2*pi*f*t-pi/2);
elseif x(i)==111
psk=sin(2*pi*f*t+pi);
else
psk=sin(2*pi*f*t-3*pi/4);
```

Command Window

```
psk=sin(2*pi*f*t+pi/4);
elseif x(i)==011
psk=sin(2*pi*f*t+3*pi/4);
elseif x(i)==100
psk=sin(2*pi*f*t+pi/2);
elseif x(i)==101
psk=sin(2*pi*6*f*t-pi/4);
elseif x(i)==110
psk=6*sin(2*pi*f*t-pi/2);
elseif x(i)==111
psk=sin(2*pi*f*t+pi);
else
psk=sin(2*pi*f*t-3*pi/4);
end
subplot(1,1,1);
plot(t,psk);
hold on;
grid on;
axis([0 10 -1 1]);
title('Phase Shift')
i=i+1;
end
>>
```

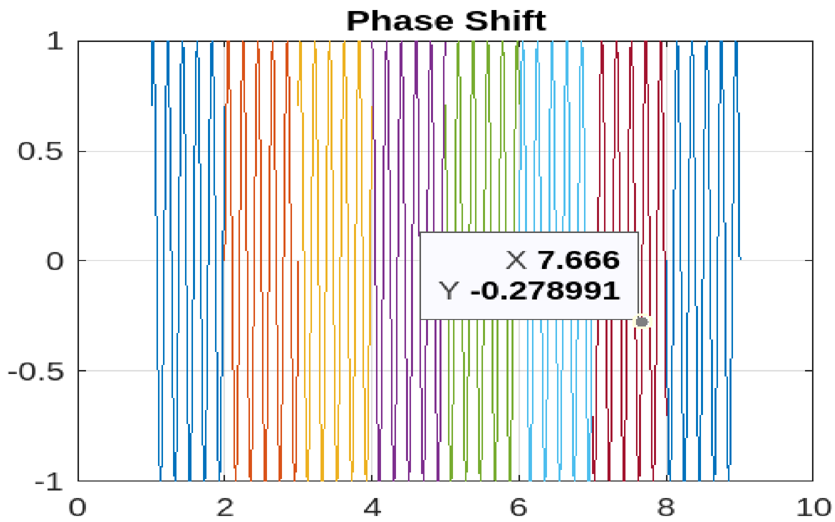


Figure 3: Performance Task 03(Phase Shift)

<u>Discussion & Conclusion:</u> In this lab experiment, we explored the process of digital-to-analog conversion using MATLAB. We performed digital to analog modulation using different modulation techniques such as Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), and Phase Shift Keying (PSK). In ASK modulation, we modulated a digital signal onto an analog carrier signal by varying the amplitude of the carrier signal. In FSK modulation, we modulated a digital signal onto an analog carrier signal by varying the frequency of the carrier signal. In PSK modulation, we modulated a digital signal onto an analog carrier signal by varying the phase of the carrier signal. We also explored Quadrature Phase Shift Keying (QPSK), which is a more advanced modulation technique used in modern communication systems. We wrote MATLAB code to generate digital signals, modulate them using different modulation techniques, and then convert them back to analog signals. We then plotted the results to visualize the modulation and demodulation processes.

References:

[1] W. Stallings, Data and computer communications. 2000., Accessed: Nov.11 , 2023. [Online]. Available:https://www.portcity.edu.bd/files/636444710465881602_Dataandcomputercommunications.pdf [Online Copy]

[2] B. A. Forouzan, C. A. Coombs, and S. C. Fegan, Introduction to data communications and networking. McGraw-Hill Science, Engineering & Mathematics, 1998., Accessed: Nov.11, 2023. [Online]. Available: https://archive.mu.ac.in/myweb_test/syllFybscit/dcn.pdf [Online Copy]