Introduction:

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

Performance Task:

ID is AB-CDEFG-H (18-39263-3), and then convert ‘E (2)’, ‘F (6)’ and ‘G (3)’ to 8 bit ASCII characters and together you have a bit stream of 24 bits. Convert this bit stream to Analog signal using the following

a) 8-ASK, different amplitudes in the modulated signal can be 1:0.5:4.5 for 000 to 111 in that order.

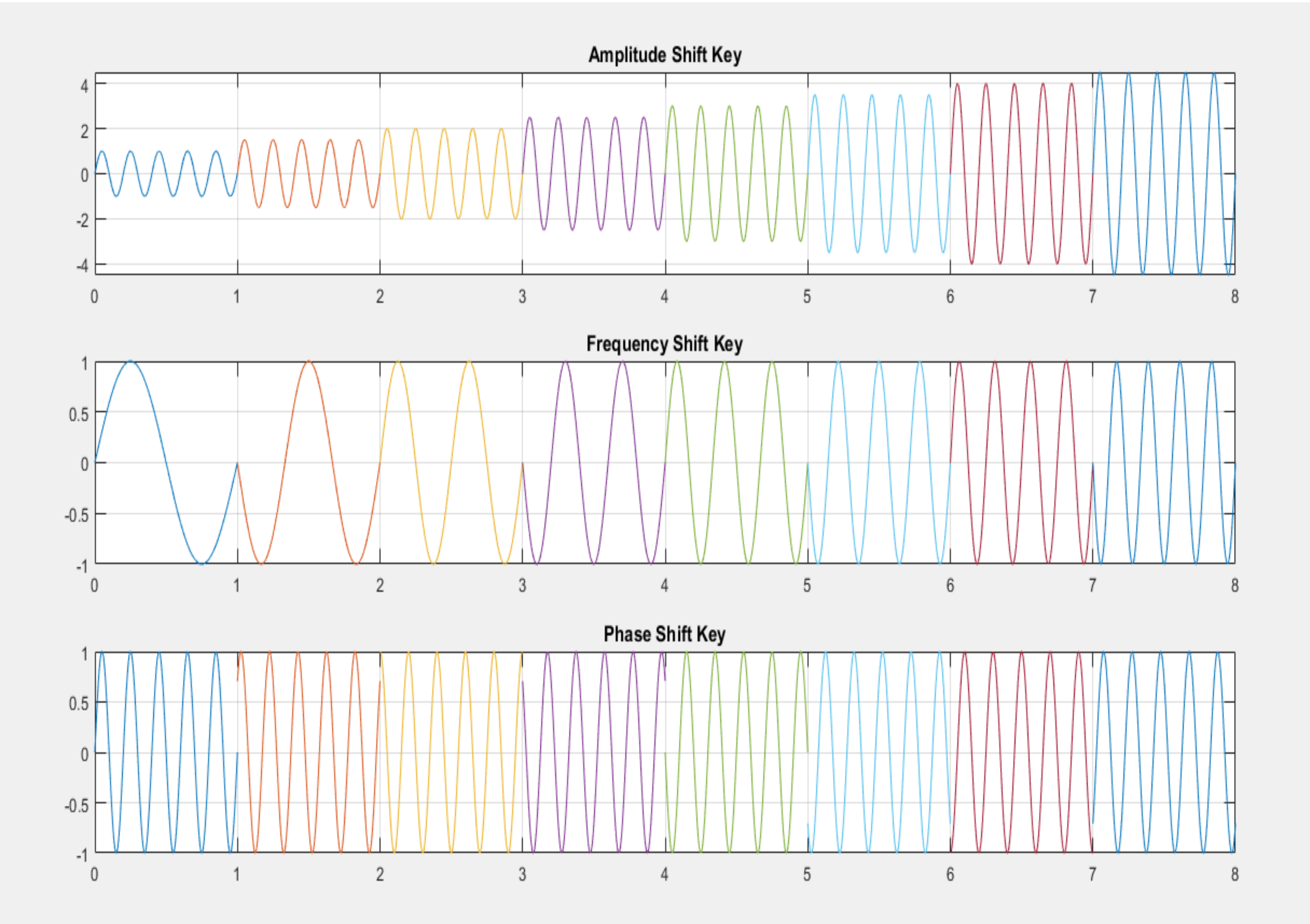
b) 8-FSK, different frequencies in the modulated signal can be 1:0.5:4.5 for 000 to 111 in that order.

c) 8-PSK, different phases in the modulated signal can be 0◦:45◦:330◦ for 000 to 111 in that order.

Keep in mind the width of any pulse or portion of Analog signal representing group of 3 bits should be 0.1 second.

```
close all;
clc;
f=5;
f1 = 1;
f2=1.5;
f3=2;
f4=2.5;
f5=3;
f6=3.5;
f7=4;
f8=4.5;
% X(EFG-263)=[0 0 1 1 0 0 1 0  0 0 1 1 0 1 1 0  0 0 1 1 0 0 1 1]
x=[0 0 0 0 0 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 ] % input signal ;
nx=size(x,2);
i=1;
a=0
while i<nx+1
    t = a:0.001:a+1;
    if (x(i)==0 && x(i+1)==0 && x(i+2)==0)
        ask=sin(2*pi*f*t);
        fsk=sin(2*pi*f1*t);
        psk=sin(2*pi*f*t);
    elseif (x(i)==0 && x(i+1)==0 && x(i+2)==1)
         ask=1.5*sin(2*pi*f*t);
         fsk=sin(2*pi*f2*t);
         psk=sin(2*pi*f*t+pi/4);
    elseif (x(i)==0 && x(i+1)==1 && x(i+2)==0)
         ask=2*sin(2*pi*f*t);
         fsk=sin(2*pi*f3*t);
         psk=sin(2*pi*f*t+2*(pi/4));
    elseif (x(i)==0 && x(i+1)==1 && x(i+2)==1)
        ask=2.5*sin(2*pi*f*t);
        fsk=sin(2*pi*f4*t);
        psk=sin(2*pi*f*t+3*(pi/4));
    elseif (x(i)==1 && x(i+1)==0 && x(i+2)==0)
        ask=3*sin(2*pi*f*t);
        fsk=sin(2*pi*f5*t);
        psk=sin(2*pi*f*t+4*(pi/4));
    elseif (x(i)==1 && x(i+1)==0 && x(i+2)==1)
        ask=3.5*sin(2*pi*f*t);
        fsk=sin(2*pi*f6*t);
        psk=sin(2*pi*f*t+5*(pi/4));
    elseif (x(i)==1 && x(i+1)==1 && x(i+2)==0)
        ask=4*sin(2*pi*f*t);
        fsk=sin(2*pi*f7*t);
        psk=sin(2*pi*f*t+6*(pi/4));
    elseif (x(i)==1 && x(i+1)==1 && x(i+2)==1)
        ask=4.5*sin(2*pi*f*t);
        fsk=sin(2*pi*f8*t);
        psk=sin(2*pi*f*t+7*(pi/4));
```

```
        end
        subplot(3,1,1);
        plot(t,ask);
        hold on;
        grid on;
        axis([0  8 -4.5 4.5]);
        title('Amplitude Shift Key')
        subplot(3,1,2);
        plot(t,fsk);
        hold on;
        grid on;
        axis([0 8 -1 1]);
        title('Frequency Shift Key')
        subplot(3,1,3);
        plot(t,psk);
        hold on;
        grid on;
        axis([0 8 -1 1]);
        title('Phase Shift Key')
    i=i+3;
    a=a+1;
    end
```



Conclusion:

The Simulink Analog signal representing group of 3 bits should be 0.1 second.