American International University- Bangladesh

COE 3103: DATA COMMUNICATION

Final Lab Assignment

Spring 2021-2022

Section: **I**
Date: **21/04/2022**

Submitted by,

Student Name	Student Id
Rahman, Sheikh Talha Jubayer	19-41468-3

Questions:

Assume your **ID** is **AB-CDEFG-H**.
Form a string of four characters, by taking any two letters and any two numbers of your choice. For example, your string can be '**6Lm7**' or '**4pT9**' or '**U8q7**'. Do not use any string from these examples. This is your text message. In this assignment you must show how we can transmit a text message and how we can recover the text message again at receiver.

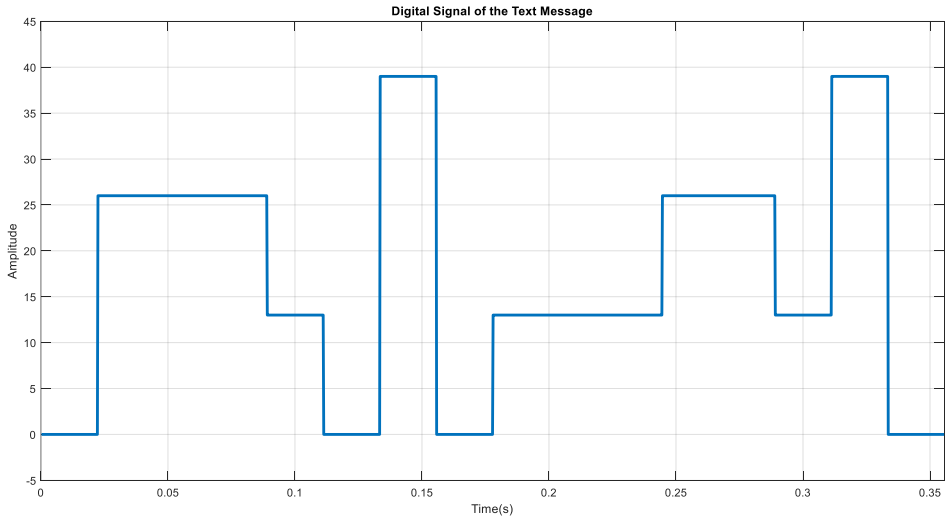
a) Convert your text message into binary bit sequence.
b) Display the bit sequence from **(a)** as four level unipolar digital signal. Use **0** volt for binary '**0 0**', use **(G+5)** volt for binary '**0 1**', use **2*(G+5)** volt for binary '**1 0**', use **3*(G+5)** volt for binary '**1 1**'. Bit rate of your digital signal must be **(G+1)*10** bps.
c) Apply QASK on digital signal from (b). Use a carrier frequency of **(G+1)*40** Hz. Assume we are transmitting this analog signal.
d) Add noise to your modulated signal and assume that the noisy signal is your received signal.
e) Recover the bit sequence from the received noisy signal.
f) Display the recovered bit sequence from **(e)** as four level unipolar digital signal. Use **0** volt for binary '**0 0**', use **(G+5)** volt for binary '**0 1**', use **2*(G+5)** volt for binary '**1 0**', use **3*(G+5)** volt for binary '**1 1**'. Bit rate of your digital signal must be **(G+1)*10** bps.
g) Regenerate your text message from recovered bit sequence of **(e)**.
h) Increase and decrease noise power at step **(d)** to analyze its impact on communication quality. What is your observation about impact of noise?

***** All codes must be inserted as text in the report.**

Answer of the Questions

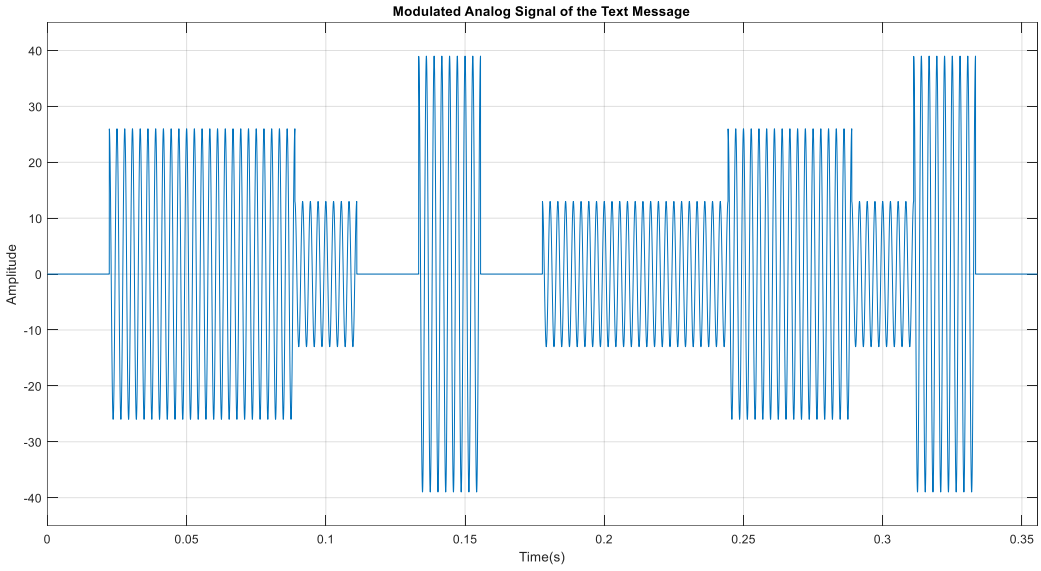
```
%ID: 19-41468-3
A = 1;
b = 9;
C = 4;
D = 1;
E = 4;
F = 6;
G = 8;
H = 3;

am00 = 0;
am01 = (G+5); %(8+5) = 13
am10 = 2*(G+5); %2*(8+5) = 26
am11 = 3*(G+5); %3*(8+5) = 39

am = [am00 am01 am10 am11]; %amplitude for 4 level signal

cf = (G+1)*40; %carrier frequency = (8+1)*40 = 360 Hz

br = (G+1)*10; %bit rate = (8+1)*10 = 90
```

```
mbr = br/2; %modulated bit rate
mbp = 1/mbr; %modulated two bits period

txt = 'T2j9'; %message to transmit

%ans of (a) - convertion of text message into binary bit sequence

dec1 = double(txt); %text to ASCII (decimal)
p1 = 2.^(0:-1:-7); %2^0,2^-1,.......,2^-7
b1 = mod(floor(p1'*dec1),2); %decimal to binary conversion
bs = reshape(b1,1,numel(b1)); %bytes to serial conbversion

%end of ans of (a)

%ans of (b) - display of the bit sequence from (a)

ds = []; %digital signal of the bit sequence
for n=1:2:length(bs) %convertion of each two bits from the bit sequence to digital
signal elements
    if bs(n)==0 && bs(n+1)==0
        rse = am(1)*ones(1,100);
    elseif bs(n)==0 && bs(n+1)==1
        rse = am(2)*ones(1,100);
    elseif bs(n)==1 && bs(n+1)==0
        rse = am(3)*ones(1,100);
    else
        rse = am(4)*ones(1,100);
    end
    ds = [ds rse];
end
t1 = mbp/100:mbp/100:100*(length(bs)/2)*(mbp/100); %time period for the digital signal
figure;
plot(t1,ds,'lineWidth',2.5);
ylabel('Amplitude');
xlabel('Time(s)');
title('Digital Signal of the Text Message');
grid on;
axis([ 0 mbp*(length(bs)/2) -5 45]);
```



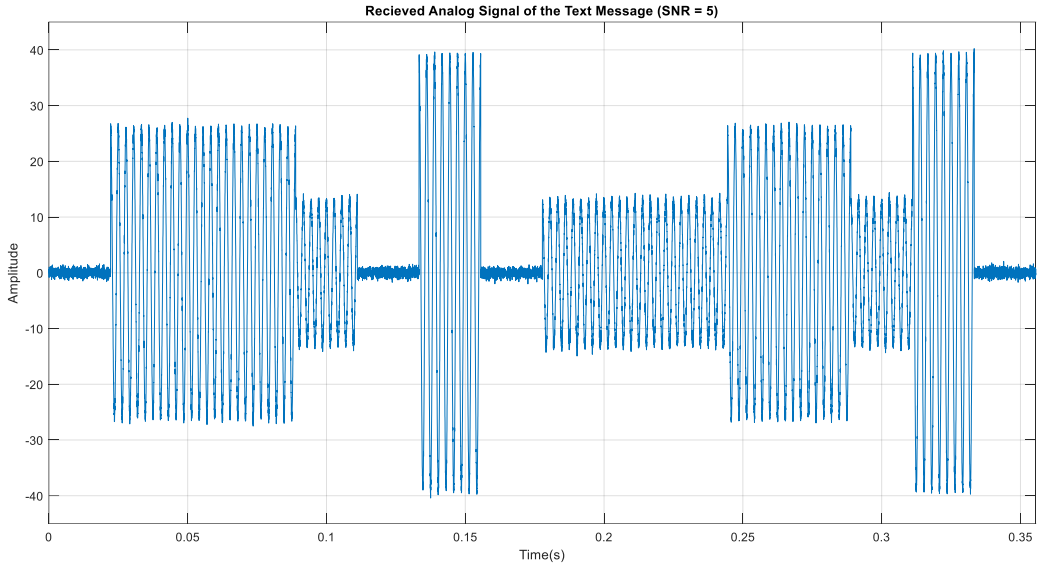

```
%end of ans of (b)
```

```
%ans of (c) - apply QASK on digital signal from (b)

t2 = mbp/1000:mbp/1000:mbp; %time period for each signal elements
as = []; %modulated analog signal of the digital signal
for n=1:2:length(bs) %convertion of each two bits from the bit sequence to analog
signal elements
    if bs(n)==0 && bs(n+1)==0
        y = am(1)*cos(2*pi*cf*t2);
    elseif bs(n)==0 && bs(n+1)==1
        y = am(2)*cos(2*pi*cf*t2);
    elseif bs(n)==1 && bs(n+1)==0
        y = am(3)*cos(2*pi*cf*t2);
    else
        y = am(4)*cos(2*pi*cf*t2);
    end
    as = [as y];
end
t3 = mbp/1000:mbp/1000:mbp*(length(bs)/2); %time period for the analog signal
figure;
plot(t3,as);
ylabel('Amplitude');
xlabel('Time(s)');
title('Modulated Analog Signal of the Text Message');
grid on;
axis([ 0 mbp*(length(bs)/2) -45 45]);
```



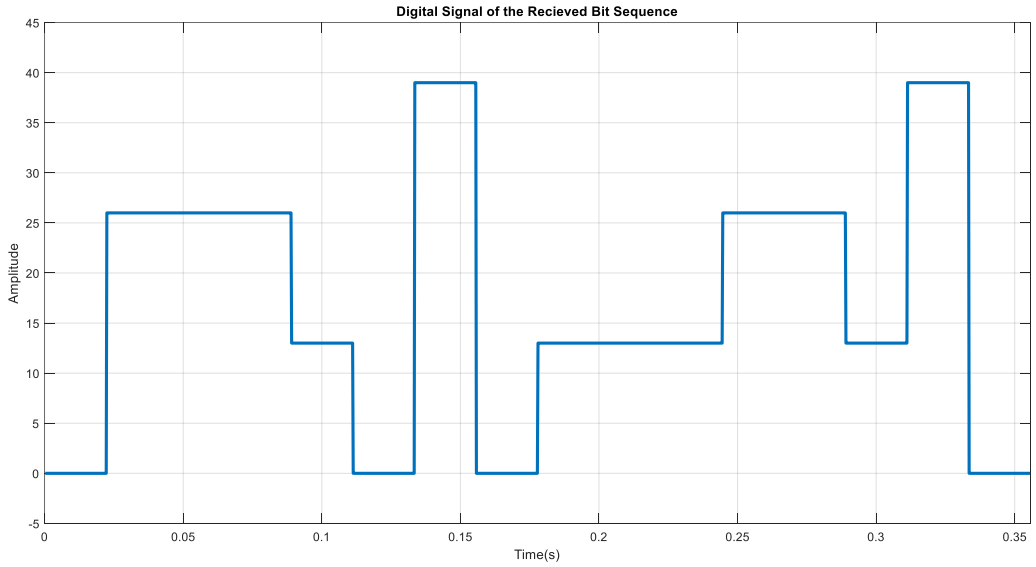

```
%end of ans (c)

%ans of (d) - add noice to the modulated signal from (c)

ras = awgn(as,5); %recieved noisy signal - add additive white gaussian noise

figure;
plot(t3,ras);
ylabel('Amplitude');
xlabel('Time(s)');
title('Recieved Analog Signal of the Text Message (SNR = 5)');
```

```
grid on;
axis([ 0 mbp*(length(bs)/2) -45 45]);
```




```
%end of ans (d)

%ans of (e) - recover bit sequence from the received noisy signal from (d)

rbs=[]; %recieved bit sequence of the noisy signal
for n=length(t2):length(t2):length(ras) %convertion of noisy signal elements to bit
sequence
    y = cos(2*pi*cf*t2); %carrier siignal
    mm = y.*ras((n-(length(t2)-1)):n);
    z = trapz(t2,mm);
    zz = round((2*z/mbp));
    if(zz<(am(1)+am(2))/2) %logic level, if the value is less than the middle point
between the next amplitude
        a = [0 0];
    elseif(zz<(am(2)+am(3))/2)
        a = [0 1];
    elseif(zz<(am(3)+am(4))/2)
        a = [1 0];
    else
        a = [1 1];
    end
    rbs=[rbs a];
end

%end of ans (e)

%ans of (f) - display of the recieved bit sequence from (e)

rds = []; %recieved digital signal of the bit sequence
for n=1:2:length(rbs) %convertion of each two bits from the recieved bit sequence to
digital signal elements
    if rbs(n)==0 && rbs(n+1)==0
        rse = am(1)*ones(1,100);
    elseif rbs(n)==0 && rbs(n+1)==1
        rse = am(2)*ones(1,100);
```

```
    elseif rbs(n)==1 && rbs(n+1)==0
        rse = am(3)*ones(1,100);
    else
        rse = am(4)*ones(1,100);
    end
    rds = [rds rse];
end
figure;
plot(t1,rds,'lineWidth',2.5);
ylabel('Amplitude');
xlabel('Time(s)');
title('Digital Signal of the Recieved Bit Sequence');
grid on;
axis([ 0 mbp*(length(bs)/2) -5 45]);
```




```
%end of ans of (f)

%ans of (g) - convertion of recieved binary bit sequence into text message

l = length(rbs); %length of input string
l8 = 8*floor(l/8); %multiple of 8 length
b2 = reshape(rbs(1:l8),8,l8/8); %cols of b are bits of chars
p2 = 2.^(0:7); %power of 2
dec2 = p2*b2; %binary to decimal conversion
rtxt = char(dec2); %ASCII (decimal) to txt

Sent_Message = txt %output of sent message
Recieved_Message = rtxt %output of recieved message
```

```
Sent_Message =

'T2j9'


Recieved_Message =

'T2j9'
```

```
%end of ans of (g)
```

```
%ans of (h) - impact of noise change in (d)

% Increasing the noise of the channel or decreasing the SNR results on
% unusable received signal at the receiver point. Changing the SNR value of
% the code 5 to -45, change the sent message to a different message
% at the reciever point.
```

```
Sent_Message =

    'T2j9'


Recieved_Message =

    'vli2'
```

```
%end of ans (h)
```

Full Code

```
%ID: 19-41468-3
A = 1;
b = 9;
C = 4;
D = 1;
E = 4;
F = 6;
G = 8;
H = 3;

am00 = 0;
am01 = (G+5); %(8+5) = 13
am10 = 2*(G+5); %2*(8+5) = 26
am11 = 3*(G+5); %3*(8+5) = 39

am = [am00 am01 am10 am11]; %amplitude for 4 level signal

cf = (G+1)*40; %carrier frequency = (8+1)*40 = 360 Hz

br = (G+1)*10; %bit rate = (8+1)*10 = 90
mbr = br/2; %modulated bit rate
mbp = 1/mbr; %modulated two bits period

txt = 'T2j9'; %message to transmit

dec1 = double(txt); %text to ASCII (decimal)
p1 = 2.^(0:-1:-7); %2^0,2^-1,.......,2^-7
b1 = mod(floor(p1'*dec1),2); %decimal to binary conversion
bs = reshape(b1,1,numel(b1)); %bytes to serial conbversion
```

```
ds = []; %digital signal of the bit sequence
for n=1:2:length(bs) %convertion of each two bits from the bit sequence to digital
signal elements
    if bs(n)==0 && bs(n+1)==0
        rse = am(1)*ones(1,100);
    elseif bs(n)==0 && bs(n+1)==1
        rse = am(2)*ones(1,100);
    elseif bs(n)==1 && bs(n+1)==0
        rse = am(3)*ones(1,100);
    else
        rse = am(4)*ones(1,100);
    end
    ds = [ds rse];
end
t1 = mbp/100:mbp/100:100*(length(bs)/2)*(mbp/100); %time period for the digital signal
figure;
plot(t1,ds,'lineWidth',2.5);
ylabel('Amplitude');
xlabel('Time(s)');
title('Digital Signal of the Text Message');
grid on;
axis([ 0 mbp*(length(bs)/2) -5 45]);

t2 = mbp/1000:mbp/1000:mbp; %time period for each signal elements
as = []; %modulated analog signal of the digital signal
for n=1:2:length(bs) %convertion of each two bits from the bit sequence to analog
signal elements
    if bs(n)==0 && bs(n+1)==0
        y = am(1)*cos(2*pi*cf*t2);
    elseif bs(n)==0 && bs(n+1)==1
        y = am(2)*cos(2*pi*cf*t2);
    elseif bs(n)==1 && bs(n+1)==0
        y = am(3)*cos(2*pi*cf*t2);
    else
        y = am(4)*cos(2*pi*cf*t2);
    end
    as = [as y];
end
t3 = mbp/1000:mbp/1000:mbp*(length(bs)/2); %time period for the analog signal
figure;
plot(t3,as);
ylabel('Amplitude');
xlabel('Time(s)');
title('Modulated Analog Signal of the Text Message');
grid on;
axis([ 0 mbp*(length(bs)/2) -45 45]);

ras = awgn(as,5); %recieved noisy signal - add additive white gaussian noise

figure;
plot(t3,ras);
ylabel('Amplitude');
xlabel('Time(s)');
title('Recieved Analog Signal of the Text Message (SNR = 5)');
grid on;
axis([ 0 mbp*(length(bs)/2) -45 45]);

rbs=[]; %recieved bit sequence of the noisy signal
for n=length(t2):length(t2):length(ras) %convertion of noisy signal elements to bit
sequence
```

```
    y = cos(2*pi*cf*t2); %carrier siignal
    mm = y.*ras((n-(length(t2)-1)):n);
    z = trapz(t2,mm);
    zz = round((2*z/mbp));
    if(zz<(am(1)+am(2))/2) %logic level, if the value is less than the middle point
between the next amplitude
        a = [0 0];
    elseif(zz<(am(2)+am(3))/2)
        a = [0 1];
    elseif(zz<(am(3)+am(4))/2)
        a = [1 0];
    else
        a = [1 1];
    end
    rbs=[rbs a];
end

rds = []; %recieved digital signal of the bit sequence
for n=1:2:length(rbs) %convertion of each two bits from the recieved bit sequence to
digital signal elements
    if rbs(n)==0 && rbs(n+1)==0
        rse = am(1)*ones(1,100);
    elseif rbs(n)==0 && rbs(n+1)==1
        rse = am(2)*ones(1,100);
    elseif rbs(n)==1 && rbs(n+1)==0
        rse = am(3)*ones(1,100);
    else
        rse = am(4)*ones(1,100);
    end
    rds = [rds rse];
end
figure;
plot(t1,rds,'lineWidth',2.5);
ylabel('Amplitude');
xlabel('Time(s)');
title('Digital Signal of the Recieved Bit Sequence');
grid on;
axis([ 0 mbp*(length(bs)/2) -5 45]);

l = length(rbs); %length of input string
l8 = 8*floor(l/8); %multiple of 8 length
b2 = reshape(rbs(1:l8),8,l8/8); %cols of d are bits of chars
p2 = 2.^(0:7); %power of 2
dec2 = p2*b2; %binary to decimal conversion
rtxt = char(dec2); %ASCII (decimal) to txt

Sent_Message = txt %output of sent message
Recieved_Message = rtxt %output of recieved message
```