Department Of electrical and computer Engineering

ENEE 3309 Communication Systems

Course project

Instructor: Dr. Wael Hashlamon

Made By: Islam Jihad

ID: 1191375

Date:8/12/2021

As we have the signal

$m(t) = \cos(2\pi f_m t)$

we want to send and modulate it using the carrier

$c(t) = \cos(2\pi f_c t)$

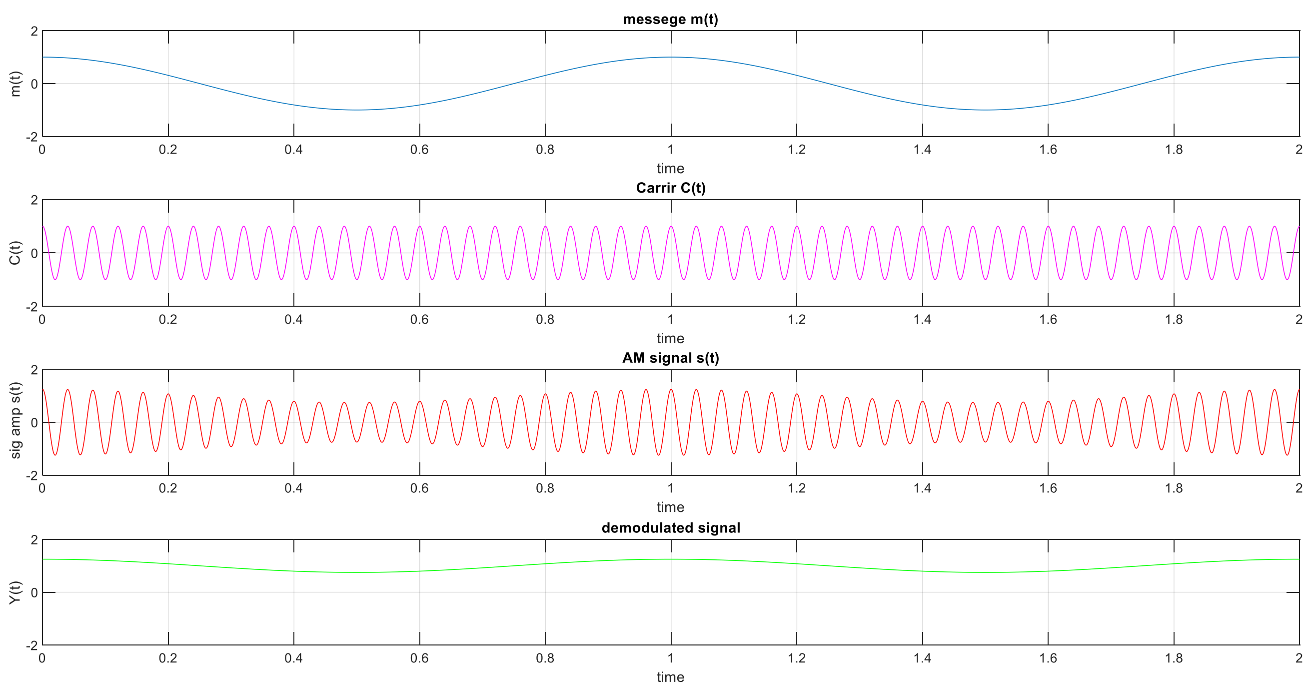
using the AM modulation method so the general formula will be as following:

$s(t) = A_c [1 + \mu \cos(2\pi f_m t)] \cos(2\pi f_c t)$

And giving that $A_c=1$ $\mu=0.25$ $f_m = 1Hz$ $f_c = 25Hz$

We have to plot s(t) over 2 cycles of the message that means we have to know the time for the message for 1 cycle $T=1/fm =1/1 =1$ and multiply it with 2 to get for 2 cycles will be 2 seconds.

The message will be carried by carrier c(t) which need to be 10 times bigger at least.



As seen the blue graph show the message signal over 2 cycles and the pink graph is the carrier over 2 cycles so when we modulate it by AM method, we get the 3rd figure in red for s(t) which give us the carrier signal under message signal effect over 2 seconds.

The green graph shows the demodulated signal from s(t) and this was by taking the absolute value of $A_c [1 + \mu \cos(2\pi f_m t)]$ without the carrier signal (Ac was ignored in the code because it equal to 1), this work as an **ideal envelope detector**

In the 3rd part of the project we were asked to plot D vs taw and to do this we need to find D (mean squared error between $s(t)$ and $y(t)$)

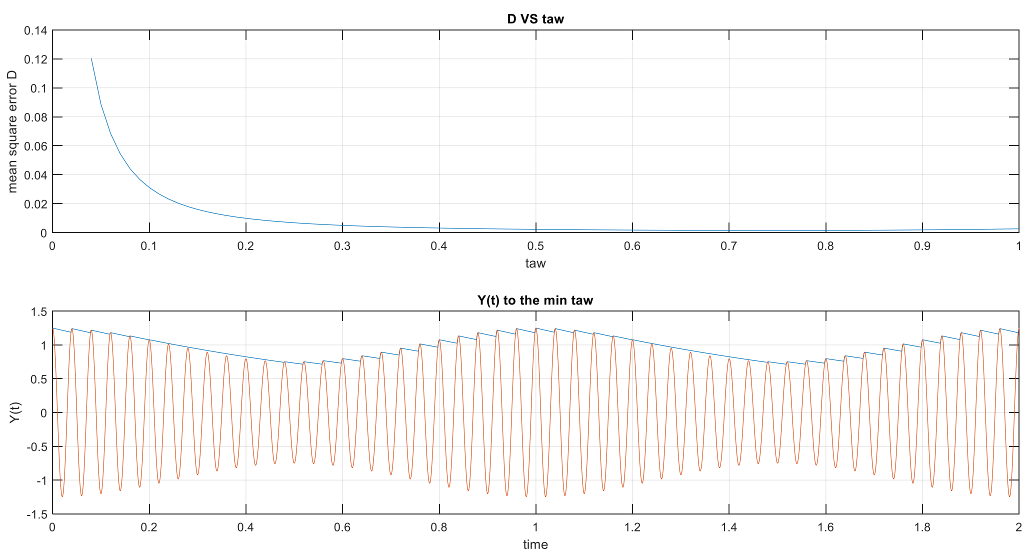
And taw ($1/fc \leq \tau \leq 1/f_m$)

To calculate taw, it equal R*C in physical way but to calculate it from the message it's the less mean squared error between $s(t)$ and $y(t)$, so I had to make a for loop that loops over taw values and every value I calculated the error with the real time for Y(t) and add the result to a summation variable instead of integration as it's easier to implement

$$D = \frac{1}{T_m} \int_0^{T_m} (y(t) - m(t))^2 dt$$

Then after finding the less mean squared error which means taw, I implemented the Y(t) function with constant taw value which was 0.75 with me as it's mean squared error was 0.0014 (the lowest)

Attach this screenshot



The blue graph shoes D vs taw as seen the lowest value of taw is between 0.7-0.8

Which actual was 0.75

In the second graph I plot Y(t) with constant taw over the s(t) function with time to see where the top points are as seen the capacitor is drains it's voltage tell it reach s(t) so the diode start to work again and charge the capacitor to the maximum value of s(t) in that range, the capacitor or in coding the Y(t) is generated as following:

If s(t) >= $s(0)e^{-t/taw}$ then Y(t) at that point of time equal to s(t)

If s(t) < $s(0)e^{-t/taw}$ then Y(t) at that point of time equal to $s(0)e^{-t/taw}$

Until reach the next peak of s(t) then the exponential factor changes to the next peak and goes on like that till it finish the 2 cycles.

Attached the 2 codes of the project here:

1st code:

```
% Ac[1 + m cos(2 pi fmt)] cos(2pi fc t)
t= 0:0.001:2;
fc=25;
fm=1;
m=0.25;
Ac=1;
mt= cos(2 * pi * fm * t);
ct= cos( 2 * pi * fc * t);
st= Ac* ct .* (1 + m.*mt);

subplot(4,1,1);
plot(t,mt);
axis([0 2 -2 2]);
title('messege m(t)');
xlabel('time');
ylabel('m(t)');
grid on;
%***********

subplot(4,1,2);
plot(t,ct,'m');
axis([0 2 -2 2]);
title('Carrir C(t)');
xlabel('time');
ylabel('C(t)');
grid on;
%***********

subplot(4,1,3);
plot(t,st,'r');
axis([0 2 -2 2]);
title('AM signal s(t)');
xlabel('time');
ylabel(' sig amp s(t)');
grid on;
%***********
absm=(1 + m.*mt);
env= abs(absm);
subplot(4,1,4);
plot(t,env,'g');
axis([0 2 -2 2]);
title('demodulated signal');
xlabel('time');
ylabel('Y(t)');
grid on;
```

the 2nd code:

```
t= 0:0.001:2;
st=1*(1 + 0.25 * cos(2 * pi * 1 * t)).* cos(2 * pi * 25 * t);
%plot(t,st);

absf=(1 + 0.25 * cos(2 * pi * 1 * t));
env= abs(absf);
%plot(t,absf);


taw= 0.04:0.01:1;

Y=zeros(1,2001);
arr_sum=zeros(1,97);
To=0;
b=0;
for i = 1:1:97

    for j=1:1:2001
        if st(j)< (b*exp(-1*((t(j)-To)/taw(i))))
            Y(j)= (b*exp(-1*((t(j)-To)/taw(i))));
        else
            Y(j)=st(j);

            if st(j)== env(j)
            b=st(j);
            To=t(j);
            end

        end

    end

    sum=0;
    for x= 1:1:2001
        sum= sum + (env(x) - Y(x))^2;
    end
    arr_sum(i)=sum/2001;
    To=0;
    b=0;

end

m=min(arr_sum);

for i= 1:1:97
    if m==arr_sum(i)
        index=i;
    end
end

disp(taw(index));
```

```
subplot(2,1,1);
plot(taw, arr_sum);
title('D VS taw');
xlabel('taw');
ylabel('mean square error D');
grid on;
```

```
arr=zeros(1,97);
Yn=zeros(1,2001);
To=0;
b=0;
for i = 1:1:97

    for j=1:1:2001
        if st(j)< (b*exp(-1*((t(j)-To)/taw(index))))
            Yn(j)= (b*exp(-1*((t(j)-To)/taw(index))));
        else
            Yn(j)=st(j);

            if st(j)== env(j)
            b=st(j);
            To=t(j);
            end
        end
    end

    To=0;
    b=0;
end

subplot(2,1,2);
plot(t,Yn,t,st);
%plot(t,Yn,t,st,t,Y);
title('Y(t) to the min taw');
xlabel('time');
ylabel('Y(t)');
grid on;
```