3TR4: Communication Systems
Lab 3

Instructor: Dr. Chen

Tamer Rafidi – L09 – 400333527
Samer Rafidi – L09 – 400333524

Part 1: Simulation

Part 1i: Plotting DSB-SC Signal both in time and frequency domains.

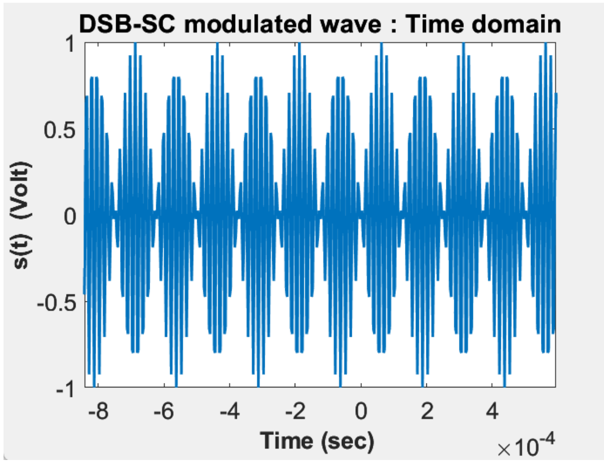


Figure 2. DSB-SC Signal in Time Domain

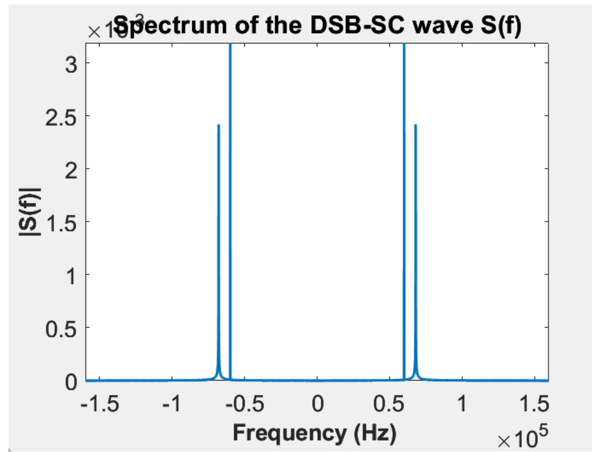


Figure 1. DSB-SC Signal in Frequency Domain

Part 1ii: Plotting DSB-SC Signal both in time and frequency domains.

Part i: Carrier is power is half of the total power.

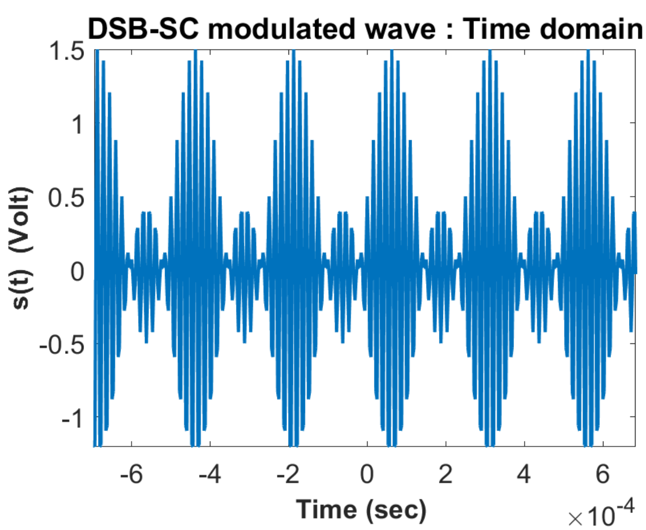


Figure 3. 50% - DSB-SC Signal in Time Domain

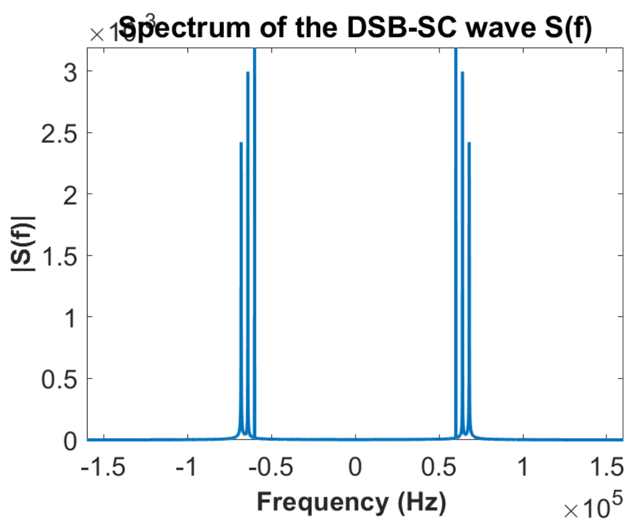


Figure 4. 50% - DSB-SC Signal in Frequency Domain

Part ii: The carrier power is three times the total power.

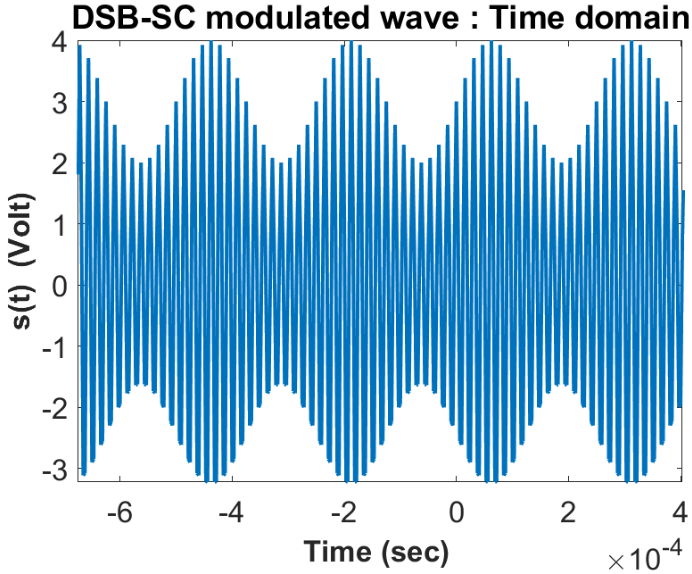


Figure 6. 3X - DSB-SC Signal in Time Domain

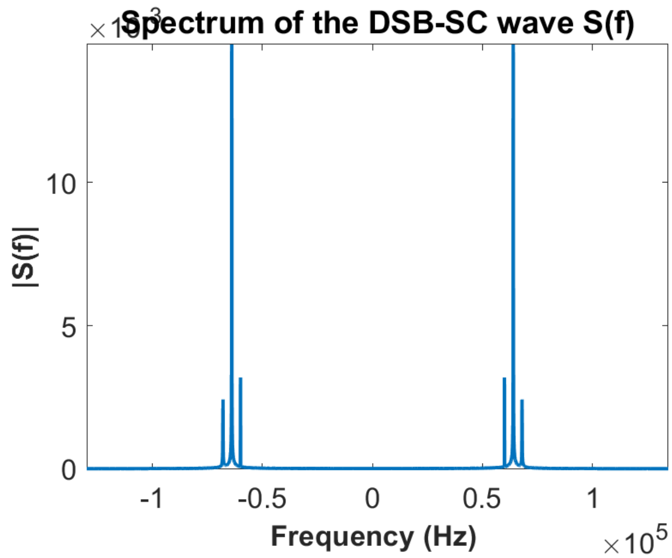


Figure 5. 3X - DSB-SC Signal in Frequency Domain

Part iii: The carrier power is more than three times the total power.

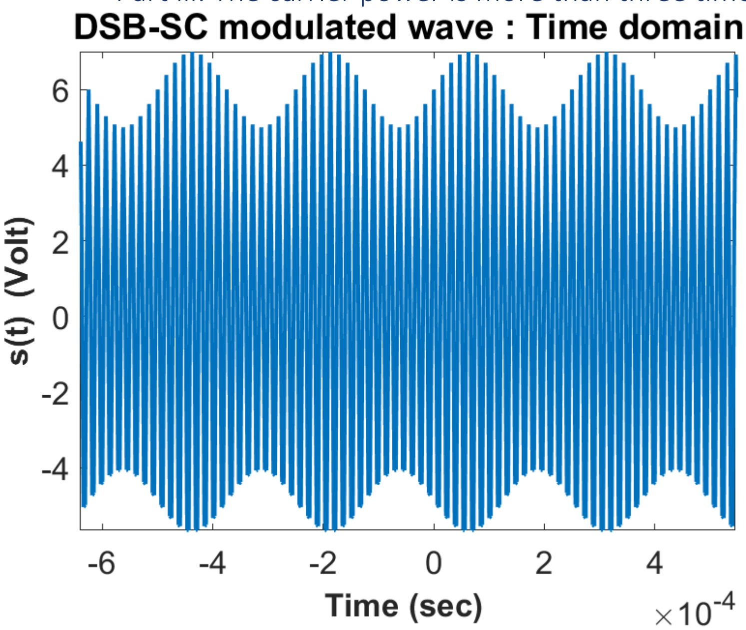


Figure 8. 6X - DSB-SC Signal in Time Domain

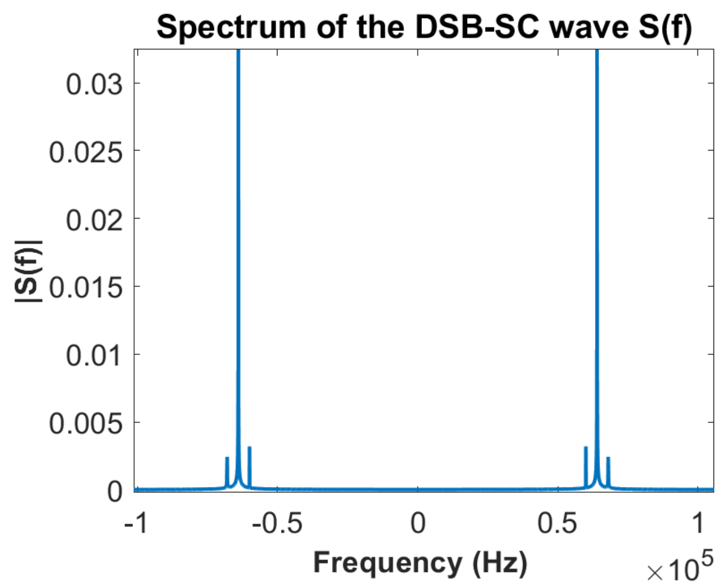


Figure 7. 6X - DSB-SC Signal in Frequency Domain

Part 1iii: Plotting DSB-SC Signal both in time and frequency domains.

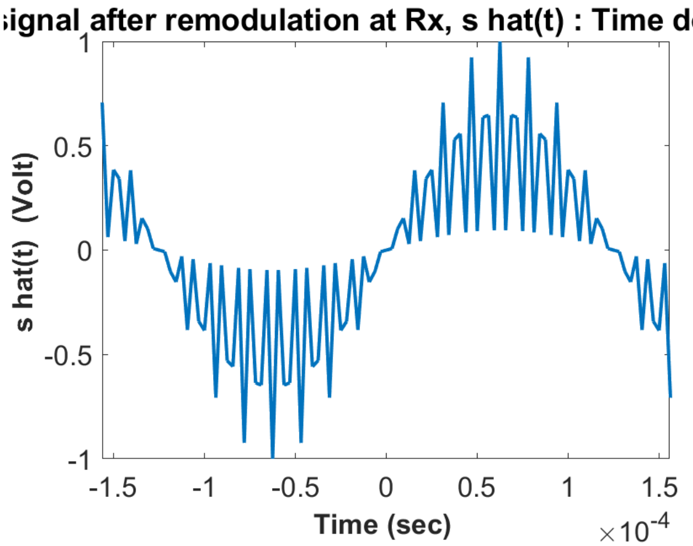


Figure 10. DSB-SC Signal in Time Domain

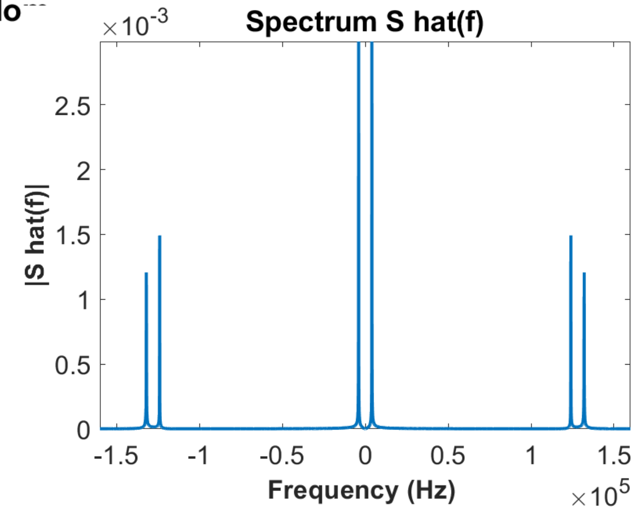


Figure 9. DSB-SC Signal in Frequency Domain

Part 1iv: Plotting DSB-SC Signal both in time and frequency domains using a low pass filter of 6 kHz.

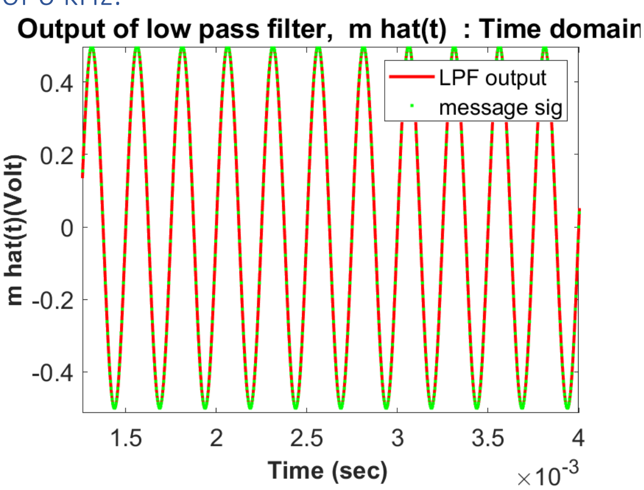


Figure 12. DSB-SC Signal in Time Domain

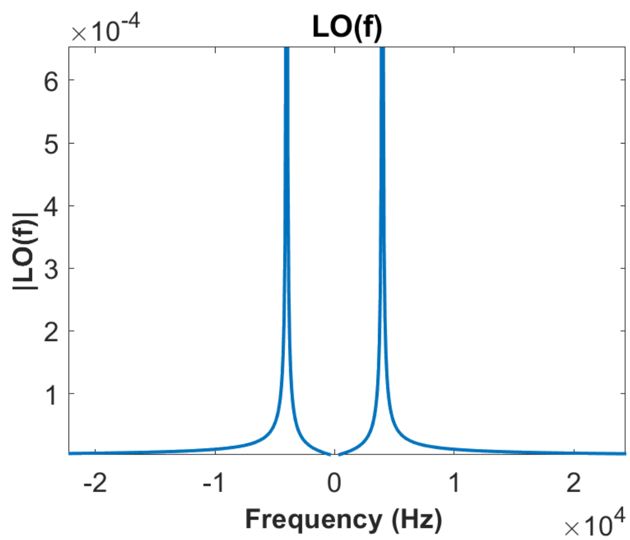


Figure 11. DSB-SC Signal in Frequency Domain

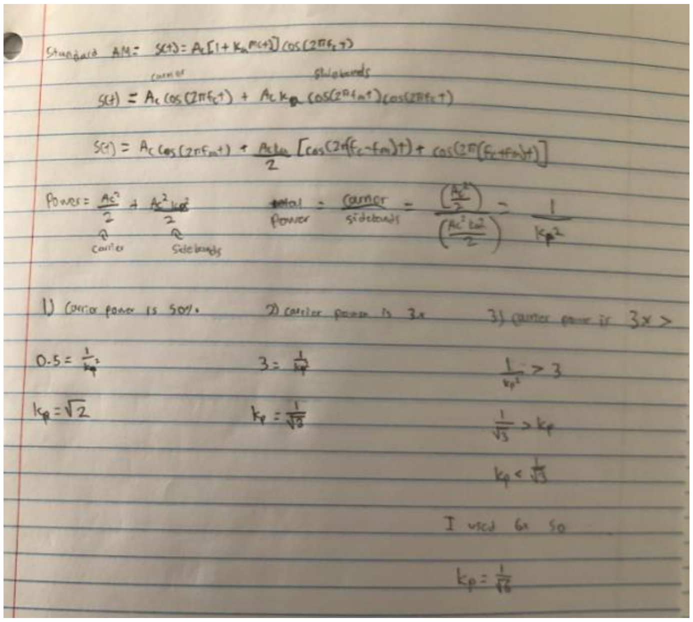
Part 1v: Underlying Theoretical Development

Part 1v: Section 2i

In the first part of the MATLAB, we are asked to plot the DSBSC waveform with a 64 kHz carrier and a 4 kHz sinusoidal message in time and frequency domains. The time domain is produced when a low frequency sinusoid (Message signal in this case) is multiplied with a higher frequency sinusoid (carrier signal in this case). For the frequency domain, we get two peaks at +- Fc.

Part 1v: Section 2ii

In this section, we are getting plots for our AM modulation using different values for our carrier. Theoretical calculations provided below.



Standard AM: $s(t) = A_c[1 + k_a m(t)]\cos(2\pi f_c t)$

carrier sidebands

$s(t) = A_c\cos(2\pi f_c t) + A_c k_a \cos(2\pi f_m t)\cos(2\pi f_c t)$

$s(t) = A_c\cos(2\pi f_c t) + \frac{A_c k_a}{2}\left[\cos(2\pi(f_c - f_m)t) + \cos(2\pi(f_c + f_m)t)\right]$

Power $= \frac{A_c^2}{2} + \frac{A_c^2 k_a^2}{2}$

carrier sidebands

$\frac{\text{~~total~~ Power}}{} = \frac{\text{carrier}}{\text{sidebands}} = \frac{\left(\frac{A_c^2}{2}\right)}{\left(\frac{A_c^2 k_a^2}{2}\right)} = \frac{1}{k_p^2}$

1) carrier power is 50%.

$0.5 = \frac{1}{k_p^2}$

$k_p = \sqrt{2}$

2) carrier power is 3x

$3 = \frac{1}{k_p^2}$

$k_p = \frac{1}{\sqrt{3}}$

3) carrier power is 3x >

$\frac{1}{k_p^2} > 3$

$\frac{1}{\sqrt{3}} > k_p$

$k_p < \frac{1}{\sqrt{3}}$

I used 6x so

$k_p = \frac{1}{\sqrt{6}}$

Part 1v: Section 2iii

In this section, we are multiplying a local oscillator (cos(2*pi*fc*t)) with s(t). We do this so we can then use a low-pass filter in the next section to retrieve the original message.

Part 1v: Section 2iv

In this section, we are retrieving the original signal using a cut-off frequency of 6 kHz. We see that the output we get is very similar to what we expected from the 4 kHz.

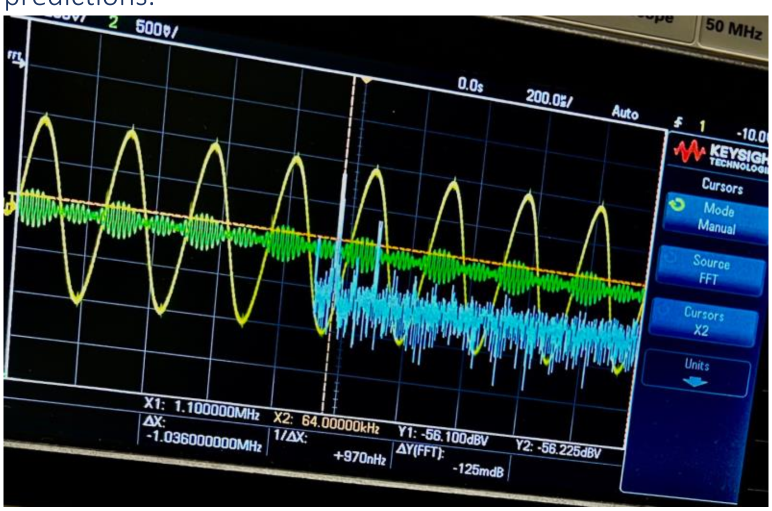
Part 2: Experimental Details

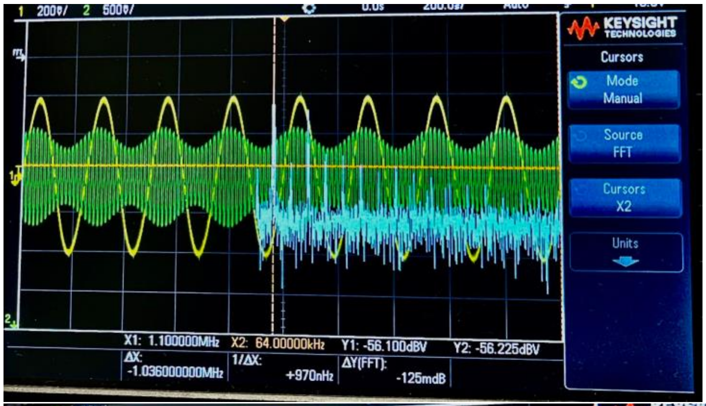
Part 2a: Transmit Section

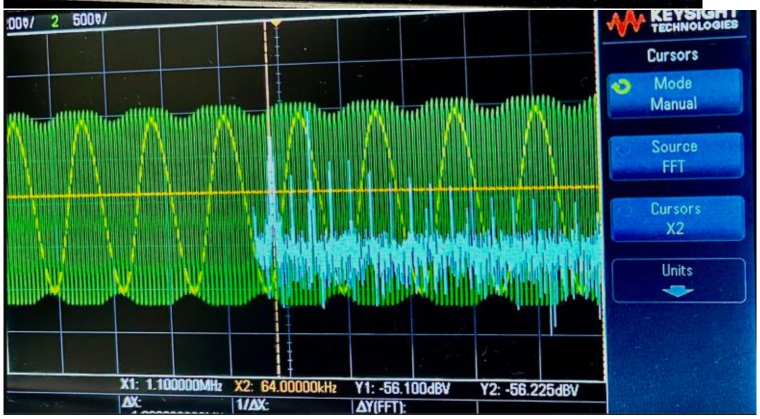
Part 2a i: Plot the observed DSBSC waveform and the simulated DSB-SC waveform in the same figure (in both time and frequency domains). Introduce a suitable amplitude scaling factor, if needed. Do you see any discrepancies between the two? Explain.

When comparing the oscilloscope waveform from the experiment to the simulated version, I can see many similar trends in the time domain. I can see the amplitude alter every time. When comparing the frequency domains, I see a set of double peaks which are shown in both the experiment and the simulated version. Looking at the theory, we expect a sinusoid at +- 4 kHz in frequency domain. And centered at +- 64kHz. This means we have peaks at +- 60 kHz and +- 68 kHz. However, the experiment disagrees with that as well as the MATLAB simulation as we see that the peaks are not at the same amplitudes.

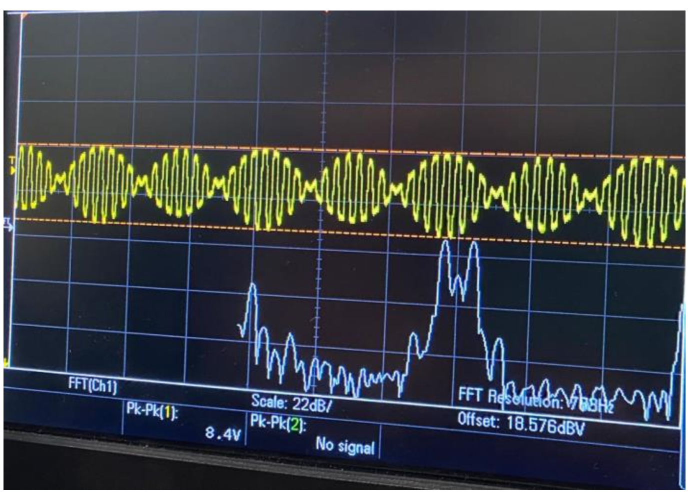
Part 2a ii: By varying PT1, test the DSBSC signal for linearity. Compare with your theoretical predictions.

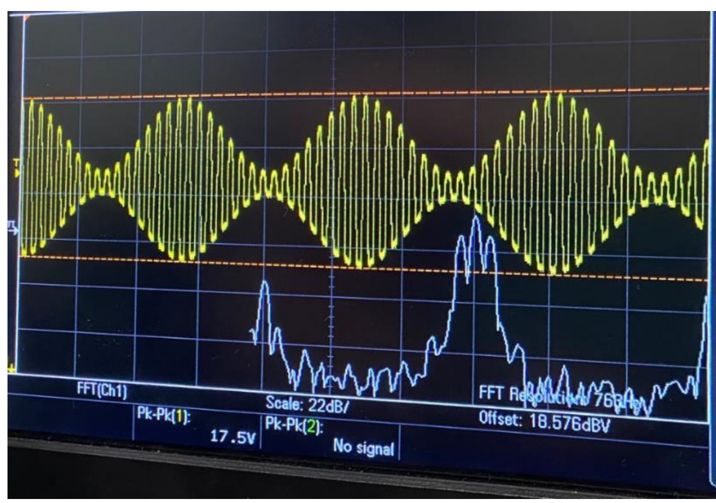


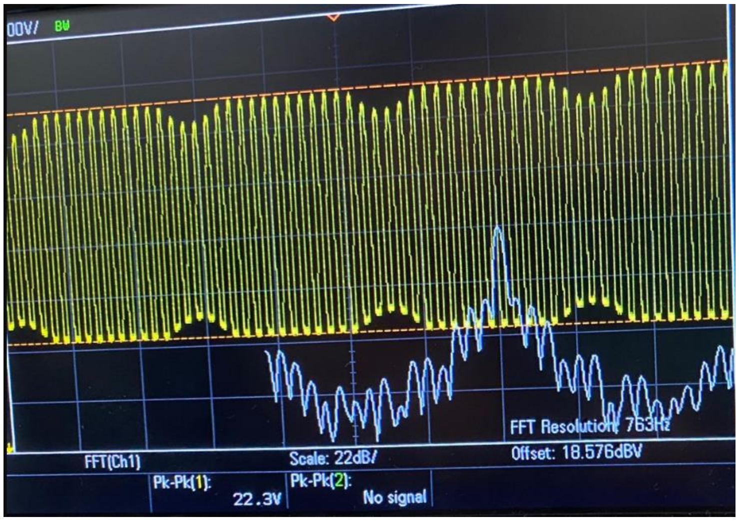




Part 2a iii: Vary the potentiometer PT2 (which adds the carrier) and observe the effect on the signal in both the time and frequency domain signals with your MATLAB plots for the standard AM.







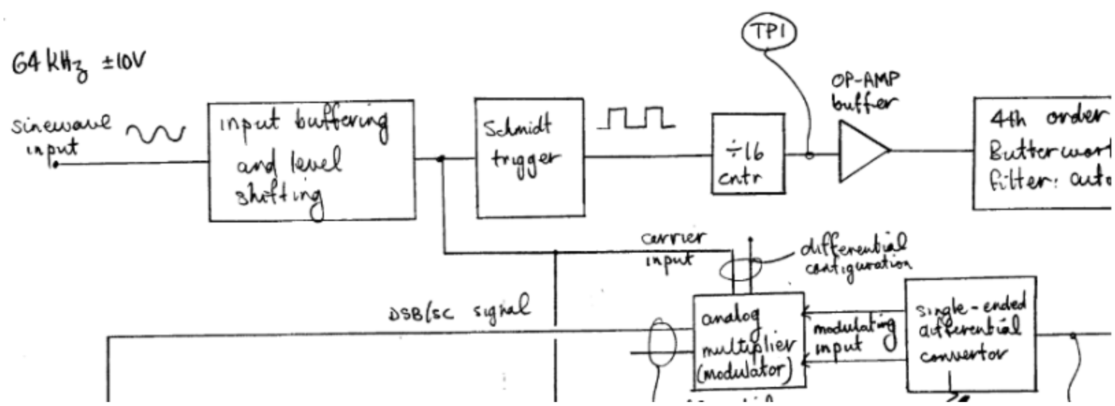
What we observed in our oscilloscope is very similar to what we observed on MATLAB. As we adjust the TP2, the limit of the yellow signal reaches a large amplitude making it very hard to demodulate the signal.

Part 2a iv: From the perspective gleaned above, what is the difference between a sinusoidally modulated DSBSC wave, and a 100% sinusoidally modulated AM wave.

The difference between a sinusoidally modulated DSBSC wave and a sinusoidal modulated AM wave is that the AM wave has a larger carrier which the DSBSC doesn't. Both waves have a carrier, but the DSBSC one is much smaller.

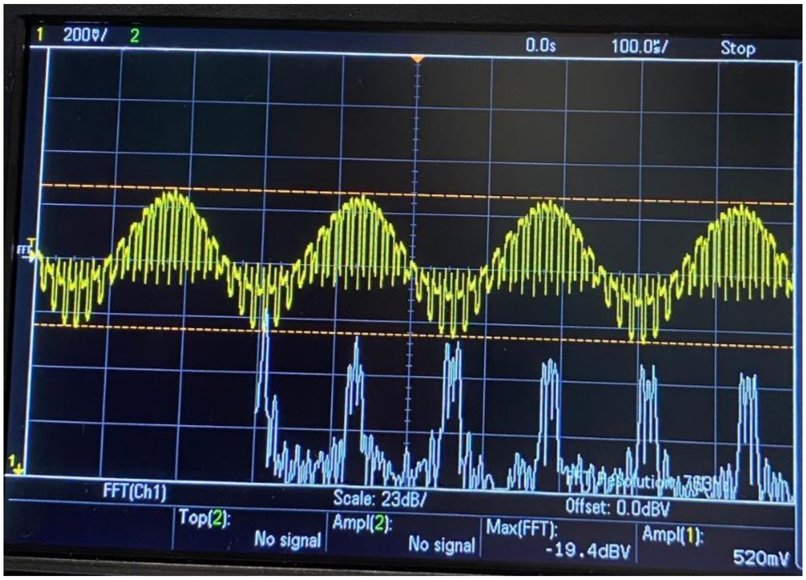
Part 2a v: Explain how the 4 kHz message signal is generated by the board. What is the advantage of generating it this way?

The 4 kHz message signal is generated because we input a 64 kHz signal which is sent into two different paths. It goes to the carrier signal from 1 path and the second path goes to the Schmidt trigger which makes the sinusoidal wave into a square wave. Then, the signal frequency is divided by 16. This signal then goes through a buffer and into a 4th order lowpass Butterworth filter. This results in a 4 kHz filter. The advantage of generating it this way is to ensures that the message and carrier signals will be in phase which will save us work to correct it.



Part 2b: Receive Section

Part 2b i: Write a brief description comparing the signal at TP4 with that obtained from the MATLAB simulations in both the time and frequency domains.

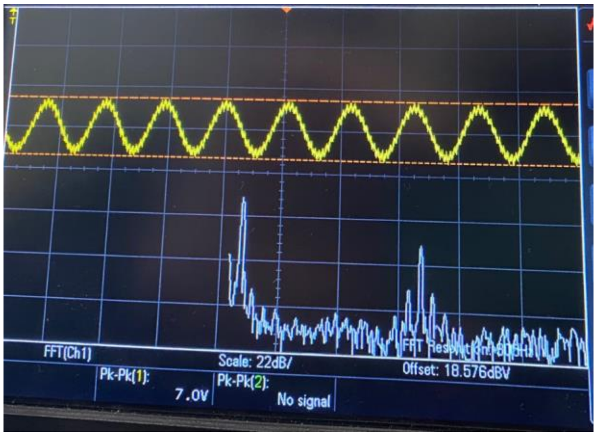


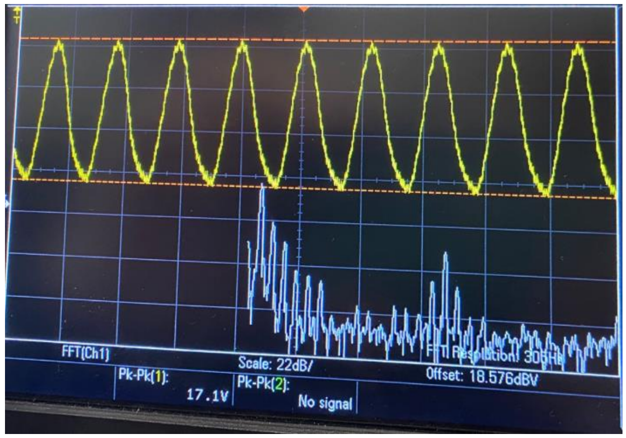
The signal received from the oscilloscope very clearly resembles the MATLAB output including the spikes in time domain. However, the negative portion does not have any of the spikes which were shown in the MATLAB simulation. This can potentially relate to the sampling rate. I think increasing the sampling rate in MATLAB will not produce any of the spikes.

Part 2b ii: What operation is performed on the second (receiver) multiplier output to extract the final demodulated signal? Explain how this operation is performed on our circuit boards.

As explained earlier, to extract the final demodulated signal, we pass the signal through a buffer and a low pass filter. This ensures that the higher frequencies are removed so we are left with our 4 kHz message signal. To do this on our circuit boards, we create an RC circuit.

Part 2b iii: Compare the demodulated output at TP5 with that obtained from the MATLAB simulations. Comment on the similarity of the demodulated output to the original message m(t). What can be done to improve the similarity?





A big difference that I can see when comparing the original message to the demodulated output is that the amplitude is affected by a large amount. This can happen due to weak or broken equipment which can cause an error in our data such as power loss.

Part 2b iv: Describe the observation you have made by varying PT1.

By varying PT1, an observation I made was the amplitude increases or decreases. Two pictures above show that pattern. The amplitude increases by a large scale when varying PT1.

Matlab

```
clear
hold off
format long e
N = 4096; %No. of FFT samples
sampling_rate = 320.0e3; %unit Hz
tstep = 1/sampling_rate;
tmax = N*tstep/2;

tmin = -tmax;
tt = tmin:tstep:tmax-tstep;
fmax = sampling_rate/2;
fmin = -fmax;
fstep = (fmax-fmin)/N;
freq = fmin:fstep:fmax-fstep;

fc=64e3;
fm = 4e3;
Ac = 1;
Am = 1;
ct=Ac*cos(2*pi*fc*tt);
mt = Am*sin(2*pi*fm*tt);
% Add 0.5*ct, 3*ct, and 6*ct for part ii.
st = mt.*ct;

figure(1)
Hp1 = plot(tt,ct);
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Time (sec) ');
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('Carrier c(t)  (Volt)');
set(Hx,'FontWeight','bold','Fontsize',16)
title('Carrier : Time domain');
axis([-1e-3 1e-3 -1.1 1.1])
pause(1)
```

```

figure(2)
Hp1 = plot(tt,mt);
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Time (sec) ');
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('message  m(t)  (Volt)');
set(Hx,'FontWeight','bold','Fontsize',16)
title('message signal : Time domain');
axis([-0.01 0.01 0 1.1])
pause(1)
figure(3)
Hp1 = plot(tt,st);
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Time (sec) ');
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('s(t)  (Volt)');
set(Hx,'FontWeight','bold','Fontsize',16)
title('DSB-SC modulated wave : Time domain');
axis([-10/fc 10/fc min(st) max(st)])
pause(1)
Mf = fftshift(fft(fftshift(mt)))/(2*fmax);
figure(4)
%The amplitude of the spectrum is different from the Fourier transform
%amplitude due to discretization of discrete Fourier transform
Hp1=plot(freq,abs(Mf));
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Frequency (Hz) ');
```

```
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('|M(f)|');
set(Hx,'FontWeight','bold','Fontsize',16)
title('Spectrum of the message signal');
axis ([-15e3 15e3 0 max(abs(Mf))])
pause(1)


Sf = fftshift(fft(fftshift(st)))/(2*fmax);
figure(5)
Hp1=plot(freq,abs(Sf));
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Frequency (Hz) ');
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('|S(f)|');
set(Hx,'FontWeight','bold','Fontsize',16)
title('Spectrum of the DSB-SC wave S(f)');
axis ([-30e3 30e3 0 max(abs(Sf))])
pause(1)

%DSB-SC demodulation

%Local oscillator at the receiver perfectly synchronized
thet=0;
lo = cos(2*pi*fc*tt + thet);
st1 = st .* lo;
figure(6)
Hp1=plot(tt,st1);
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Time (sec) ');
set(Hx,'FontWeight','bold','Fontsize',16)
```

```
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('  s hat(t)  (Volt)');
set(Hx,'FontWeight','bold','Fontsize',16)
title('signal after remodulation at Rx, s hat(t) : Time domain');
axis([-10/fc 10/fc min(st1) max(st1)])
pause(1)
Sf1 = fftshift(fft(fftshift(st1)))/(2*fmax);
figure(7)
Hp1=plot(freq,abs(Sf1));
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Frequency (Hz) ');
set(Hx,'FontWeight','bold','Fontsize',16)
% Change to |LO(f)|for part iv.
Hx=ylabel('|S hat(f)|');
set(Hx,'FontWeight','bold','Fontsize',16)
% Change to LO(f) for part iv.
title('Spectrum S hat(f)');
axis ([-50e3 50e3 0 max(abs(Sf1))])
pause(1)
%Low pass filtering
%6e3 for part iv, 30e3 for other parts
f_cutoff = 6e3;
%ideal low pass filter
n=1;
for f = freq
    if abs(f) < f_cutoff
        Hf(n) = 1;
    else
        Hf(n) = 0;
    end
n=n+1;
end
Mf1 = Sf1 .* Hf;
mt1 = 2*fmax*fftshift(ifft(fftshift(Mf1)));
figure(8)
```

```
Hp1=plot(tt,mt1,'r',tt,mt*0.5,'g.');
set(Hp1,'LineWidth',2)
Ha = gca;
set(Ha,'Fontsize',16)
Hx=xlabel('Time (sec) ');
set(Hx,'FontWeight','bold','Fontsize',16)
Hx=ylabel('m hat(t)(Volt)');
set(Hx,'FontWeight','bold','Fontsize',16)
title('Output of low pass filter,  m hat(t)  : Time domain');
axis([-0.01 0.01 min(mt*0.5) max(mt*0.5)])
legend('LPF output', 'message sig');
```