International Institute of Information Technology
Bangalore

Principles of Communication Systems Lab

ECE 303P

Lab 5: SSB-SC modulation and demodulation

Mohd. Rizwan Shaikh
(IMT2019513)

September 8, 2021

Lab 5: SSB-SC modulation and demodulation

Mohd. Rizwan Shaikh

September 8, 2021

Abstract

In this Lab report, we study SSB-SC modulation and demodulation. We plot the message signal, carrier signal and their frequency spectrum. We study SSB-SC modulation by plotting modulated signal and frequency spectrum. We also demodulate the transmission signal and study its various properties.

1 Introduction

1.1 Question 1

Consider a SSB-SC modulated signal $\phi_{SSB}(t) = A_c m(t) cos(2\pi f_c t) \pm j A_c m_h(t) sin(2\pi f_c t)$ with message signal $m(t) = A_m cos(2\pi f_m t)$. Assume $A_m = A_c = 2$, $f_m = 100$ Hz and $f_c = 2$ KHz.

(a) Plot the message signal $m(t) = A_m cos(2\pi f_m t)$ for complete 4 cycles.

(b) Plot the carrier signal $c(t) = A_c cos(2\pi f_c t)$ for a duration equal to the duration of the message signal $m(t)$.

(c) Plot the SSB-SC signal $\phi_{SSB}(t)$ for both USB and LSB tansmissions.

(d) Plot the frequency spectrum of the signals in part (a), (b) and (c).

(e) Demodulate the above SSB-SC signal using the synchronous detector discussed in the class. Plot the demodulated signal and its spectrum. Assume that the receiver can generate a carrier signal i) $c(t) = A_c cos(2\pi f_c t)$ and ii) $c(t) = A_c cos(2\pi f_c t + \phi)$ with $\phi = 90^o$.

1.2 Question 2

Consider a multi-tone signal $m(t) = A_1 cos(2\pi f_1 t) + A_2 cos(2\pi f_2 t)$ and a carrier signal $A_c cos(2\pi f_c t)$ with $A_c = A_1 = A_2 = 1$, $f_1 = 100$ Hz, $f_2 = 200$ Hz and $f_c = 2$ KHz.

(a) Plot the signal $A_1 cos(2\pi f_1 t)$ for complete two cycles.

(b) Plot the signal $A_2 cos(2\pi f_2 t)$ and the carrier signal $A_c cos(2\pi f_c t)$ over the duration of signal $A_1 cos(2\pi f_1 t)$.

(c) Plot the SSB-SC signal $\phi_{SSB}(t) = A_c m(t) cos(2\pi f_c t) \pm j A_c m_h(t) sin(2\pi f_c t)$ for both USB and LSB transmissions.

(d) Plot the frequency spectrum of the message signal $m(t)$, the carrier signal $c(t) = A_c cos(2\pi f_c t)$ and the signal $\phi_{SSB}(t)$ for both USB and LSB transmissions.

(e) Demodulate the above SSB-SC signal using the synchronous detector discussed in the class. Plot the demodulated signal and its spectrum. Assume that the receiver can generate a carrier signal i) $c(t) = A_c cos(2\pi f_c t)$ and ii) $c(t) = A_c cos(2\pi f_c t + \phi)$ with $\phi = 90^o$.

2 Method

2.1 Question 1

To begin with, we first create a time vector and frequency vector followed by message signal vector, carrier signal vector and ssbsc vector as shown below:

```
Am = 2;          Ac = 2;      %Parameters
fm = 100;        fc = 2000;
fs = 50000;      df = 1;
t = [0:fs-1]*1/fs;            %Time vector
f = [-fs/2:fs/2-1]*df;        %Frequency vector

m_t  = Am*cos(2*pi*fm*t);     %Message signal
c_t  = Ac*cos(2*pi*fc*t);     %Carrier signal
m_ht = imag(hilbert(m_t));

ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC signal for LSB transmission
ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC signal for USB transmission
```

Here, we take the sampling frequency as 50 KHz. We plot the message signal, carrier signal and SSB-SC signals using `plot(t, x)` command.

Using the following piece of code, we find the Fourier transform of the message signal, carrier signal and SSB-SC signals.

```
fftM = fftshift(fft(m_t))/length(m_t);       %Message signal
fftC = fftshift(fft(c_t))/length(c_t);       %Carrier signal
fftSSBL = fftshift(fft(ssbl))/length(ssbl); %SSB-SC signal for LSB transmission
fftSSBU = fftshift(fft(ssbu))/length(ssbu); %SSB-SC signal for USB transmission
```

To demodulate the transmission signal, we use the concepts of synchronous detector. First we multiply the transmission signal by $A_c cos(2\pi f_c t + \phi)$. Then we pass it through a low pass filter. Finally, we divide the signal by $A_c^2/2$ to get the phase shifted version of the original signal. Finally, we use hilbert transform to get back the original signal. The code for this process is shown below:

```
ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-L signal
r2 = Ac*cos(2*pi*fc*t + pi/2);              %signal from receiver
m2ssbl = r2.*ssbl;      %Multiply by Accos(2*pi*fc*t + phi)
m2ssbl = lowpass(m2ssbl, 150, fs);      %Pass through low pass filter
m2ssbl = m2ssbl/(Ac^2/2);             %Divide by Ac/2
m2ssbl = imag(hilbert(m2ssbl));     %Take Hilbert transform
```

2.2 Question 2

We follow the same steps as question 1 to get the desired results.

3 Results and Analysis

3.1 Question 1

To solve the question, we create time vector, frequency vector, message signal vector, carrier signal vector and SSB-SC signal vectors as shown below:

```
Am = 2;         Ac = 2;     %Parameters
fm = 100;       fc = 2000;      fs = 50000;     df = 1;
t = [0:fs-1]*1/fs;          %Time vector
f = [-fs/2:fs/2-1]*df;      %Frequency vector
m_t = Am*cos(2*pi*fm*t);    %Message signal
c_t = Ac*cos(2*pi*fc*t);    %Carrier signal
m_ht = imag(hilbert(m_t));  %hilbert transform of message signal
ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC - LSB
ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC - USB
```

The MATLAB code to plot the message signal and carrier signal (part(a) and part(b)) is:

```
%≤===== 1(a) ======>
subplot(2, 1, 1);           %Plot the message signal
plot(t, m_t, 'LineWidth', 1.5);
title('1(a) Message Signal: A_mcos(2\pif_mt)');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.04]);             ylim([-3 3]);
grid on;
%≤===== 1(b) ======>
subplot(2, 1, 2);           %Plot the carrier signal
plot(t, c_t, 'LineWidth', 1.5);
title('1(b) Carrier Signal: A_ccos(2\pif_ct)');
xlabel('---> time(s)');     ylabel('---> c(t)');
xlim([0 0.04]);             ylim([-3 3]);
grid on;
```

The plot of message signal and carrier signal (part (a) and part (b)) is as follows:

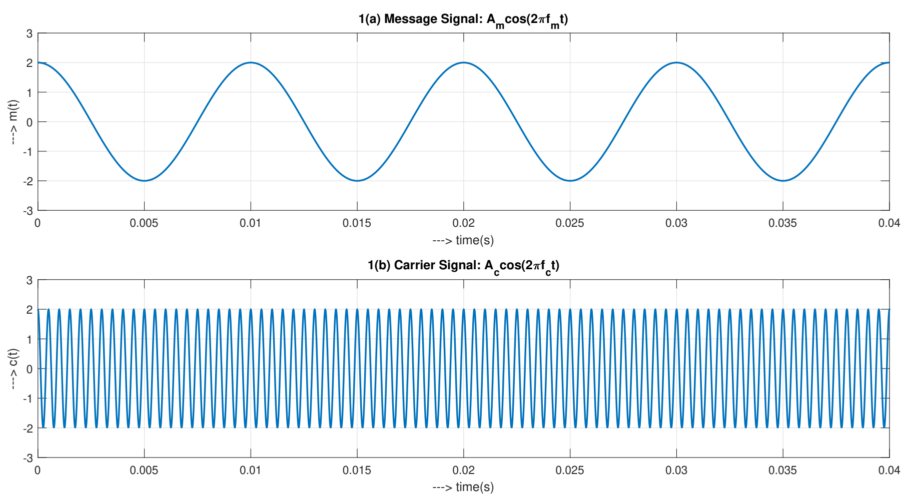

Figure 1: Message signal and carrier signal

The MATLAB code to plot SSB-SC signal for both USB and LSB transmissions (part (c)) is shown below:

```
%<===== 1(c) ======>
subplot(2,1,1);                  %Plot the signal
plot(t, ssbl, 'LineWidth', 1.5);
title('SSB-SC signal - LSB');
xlabel('---> time(s)');     ylabel('---> \phi(t) - LSB');
xlim([0 0.04]);             ylim([-5 5]);
yticks(-5:1:5);             grid on;

subplot(2,1,2);                  %Plot the signal
plot(t, ssbu,  'LineWidth', 1.5);
title('SSB-SC signal - USB');
xlabel('---> time(s)');     ylabel('---> \phi(t) - USB');
xlim([0 0.04]);             ylim([-5 5]);
yticks(-5:1:5);             grid on;
sgtitle('Plots for 1(c)');
```

The plot of SSB-SC signal (part(c)) is as follows:

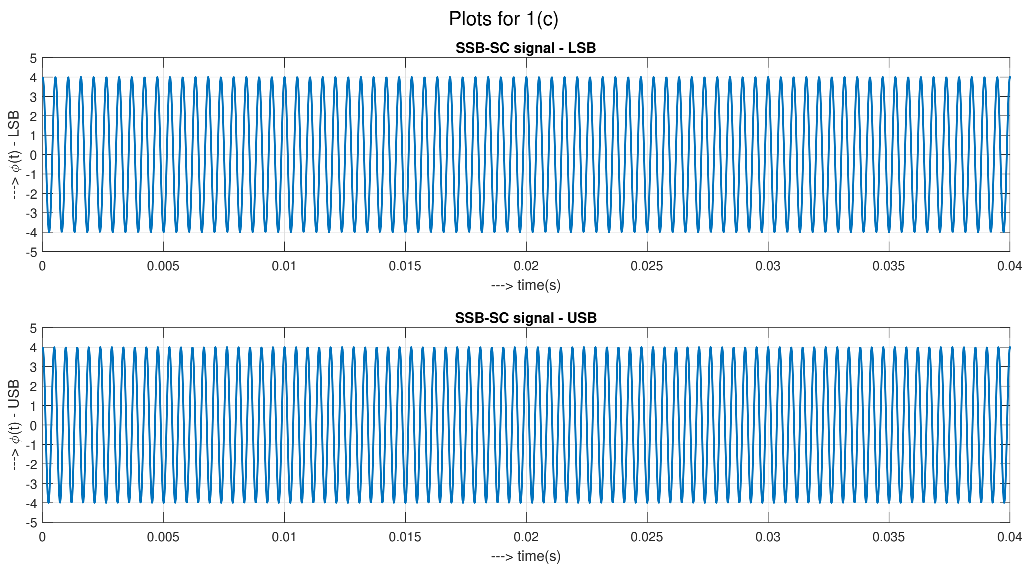

Figure 2: SSB-SC signal ($\phi_{SSB}(t)$)

The MATLAB code to plot the frequency spectrum of the signals (part(d)) is as follows:

```
%<===== 1(d) ======>
f = [-fs/2:fs/2-1]*df;                   %Frequency vector
fftM = fftshift(fft(m_t))/length(m_t);   %Find FFT of the signals
fftC = fftshift(fft(c_t))/length(c_t);
fftSSBL = fftshift(fft(ssbl))/length(ssbl);
fftSSBU = fftshift(fft(ssbu))/length(ssbu);

subplot(2,2,1);                          %Plot FFT of message signal
plot(f, abs(fftM), 'LineWidth', 1.7);
title('Frequency spectrum of m(t)');
```

```
    xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
    xlim([-200 200]);               ylim([0 1.2]);
    grid on;

    subplot(2,2,2);                         %Plot FFT of carrier signal
    plot(f, abs(fftC), 'LineWidth', 1.7);
    title('Frequency spectrum of c(t)');
    xlabel('---> Frequency(Hz)');   ylabel('---> |C(f)|');
    xlim([-2500 2500]);             ylim([0 1.2]);
    grid on;

    subplot(2,2,3);                         %Plot FFT of SSB-SC - LSB signal
    plot(f, abs(fftSSBL), 'LineWidth', 1.7);
    title('Frequency spectrum of \phi_{SSB}(t) - LSB');
    xlabel('---> Frequency(Hz)');   ylabel('---> |\phi_{SSB}(f)|');
    xlim([-2500 2500]);             ylim([0 2.5]);
    grid on

    subplot(2,2,4);                         %Plot FFT of SSB-SC - USB signal
    plot(f, abs(fftSSBU), 'LineWidth', 1.7);
    title('Frequency spectrum of \phi_{SSB}(t) - USB');
    xlabel('---> Frequency(Hz)');   ylabel('---> |\phi_{SSB}(f)|');
    xlim([-2500 2500]);             ylim([0 2.5]);
    grid on;        sgtitle('Plots for 1(d)');
```

The plot of the frequency spectrum of the signals (part (d)) is shown below:

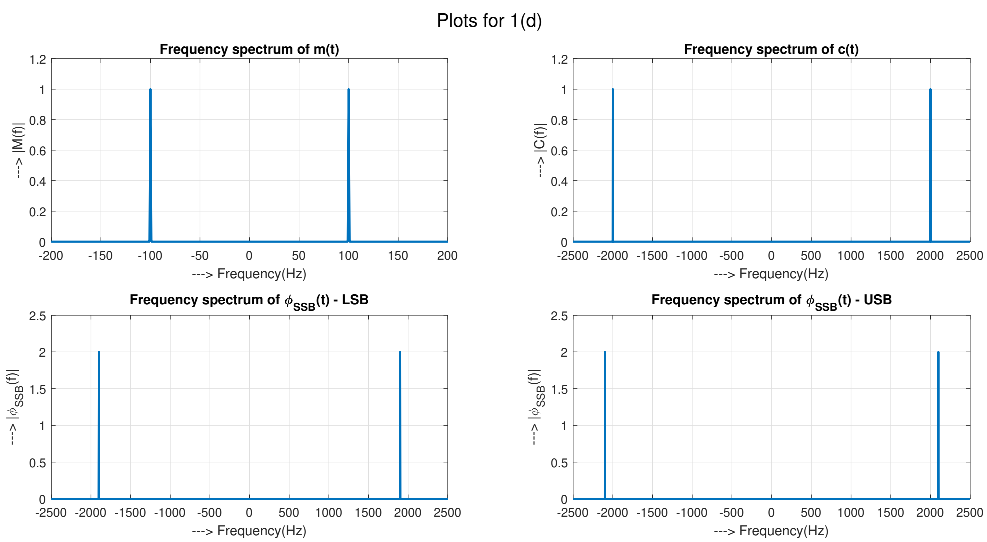

Figure 3: Frequency spectrum of $m(t)$, $c(t)$, $\phi_{SSB-L}(t)$ and $\phi_{SSB-U}(t)$

Here,

- The tone of message signal $m(t)$ is at 100 Hz.
- The tone of carrier signal $c(t)$ is at 2 KHz.
- The tone of $\phi_{SSB-L}(t)$ is at 1.9 KHz.
- The tone of $\phi_{SSB-U}(t)$ is at 2.1 KHz.

The MATLAB code to demodulate the transmission signal when the receiver generates $A_c cos(2\pi f_c t)$ (part 1(e1)) is shown below:

```
%<======1(e1)=======>
ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-L signal
ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t); %SSB-U signal
r1 = Ac*cos(2*pi*fc*t);     %Receiver generated signal

m1ssbl = r1.*ssbl;          %Multiply with receiver generated signal
m1ssbu = r1.*ssbu;

m1ssbl = lowpass(m1ssbl, 150, fs); %Pass through low pass filter
m1ssbu = lowpass(m1ssbu, 150, fs);

m1ssbl = m1ssbl/(Ac/2);         %Divide by Ac/2
m1ssbu = m1ssbu/(Ac/2);

fftM1ssbl = fftshift(fft(m1ssbl))/length(m1ssbl);   %Find FFT of ... demodulated signals
fftM1ssbu = fftshift(fft(m1ssbu))/length(m1ssbu);
f = [-fs/2:fs/2-1]*df;                       %Frequency vector

sgtitle('Plots for 1(e) for receiver signal phase 0^o');
subplot(2,2,1);
plot(t, m1ssbl, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - LSB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.04]);             ylim([-3 3]);
yticks(-5:1:5);
grid on;

subplot(2,2,2);
plot(f, abs(fftM1ssbl), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - LSB');
xlabel('---> Frequency(Hz)');       ylabel('---> |M(f)|');
xlim([-200 200]);                   ylim([0 1.5]);
grid on;

subplot(2,2,3);
plot(t, m1ssbu, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - USB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.04]);             ylim([-3 3]);
yticks(-5:1:5);
grid on;

subplot(2,2,4);
plot(f, abs(fftM1ssbu), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - USB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-200 200]);               ylim([0 1.5]);
grid on;
```

The Plot of the demodulated signal along with its frequency spectrum for SSB-SC signal for both LSB and USB transmission is shown below:

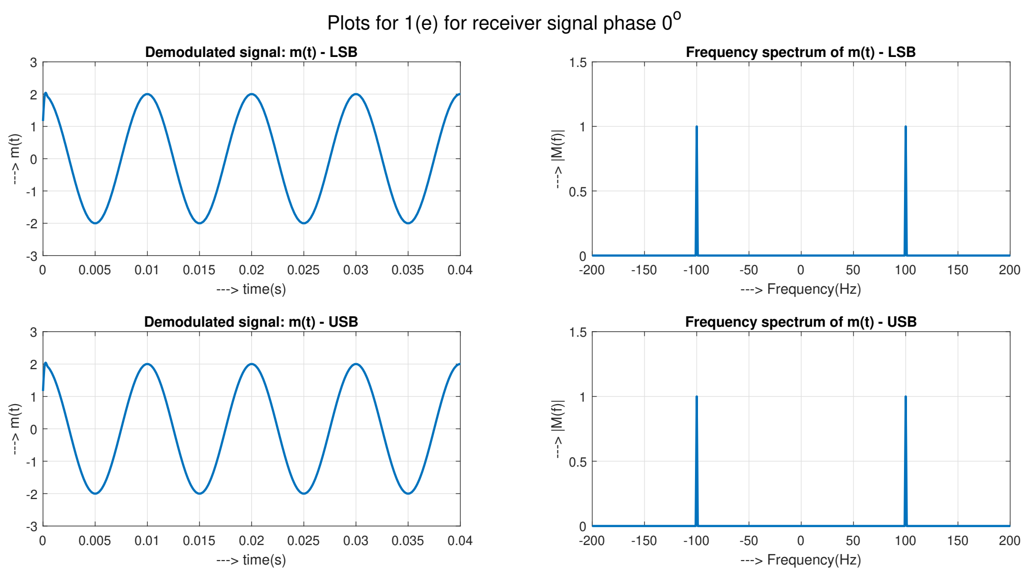

Figure 4: Demodulated signal and its frequency spectrum for receiver signal $A_c cos(2\pi f_c t)$

The MATLAB code to demodulate the transmission signal when the receiver generates $A_c cos(2\pi f_c t + \phi)$ $(\phi = 90^o)$ (part 1(e2)) is as follows:

```
%<======1(e2)=======>
ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t);     %SSB-L signal
ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t);     %SSB-U signal
r2 = Ac*cos(2*pi*fc*t + pi/2);      %Receiver generated signal
m2ssbl = r2.*ssbl;       %Multiply with receiver generated signal
m2ssbu = r2.*ssbu;
m2ssbl = lowpass(m2ssbl, 150, fs);  %Pass through low pass filter
m2ssbu = lowpass(m2ssbu, 150, fs);
m2ssbl = m2ssbl/(Ac^2/2);          %Divide by Ac/2
m2ssbu = m2ssbu/(Ac^2/2);
m2ssbl = imag(hilbert(m2ssbl));     %shift phase by pi/2
m2ssbu = -imag(hilbert(m2ssbu));
fftM2ssbl = fftshift(fft(m2ssbl))/length(m2ssbl);   %Find FFT of the ...
    demodulated signal
fftM2ssbu = fftshift(fft(m2ssbu))/length(m2ssbu);

sgtitle('Plots for 1(e) for receiver signal phase 90^o');
subplot(2,2,1);
plot(t, m2ssbl, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - LSB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.04]);             ylim([-3 3]);
yticks(-5:1:5);             grid on;

subplot(2,2,2);
plot(f, abs(fftM2ssbl), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - LSB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-200 200]);               ylim([0 1.5]);
grid on;
```

```
subplot(2,2,3);
plot(t, m2ssbu, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - USB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.04]);             ylim([-3 3]);
yticks(-5:1:5);             grid on;

subplot(2,2,4);
plot(f, abs(fftM2ssbu), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - USB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-200 200]);               ylim([0 1.5]);
grid on;
```

The plot of the demodulated signal along with its frequency spectrum for SSB-SC signal for both LSB and USB transmission is shown below:

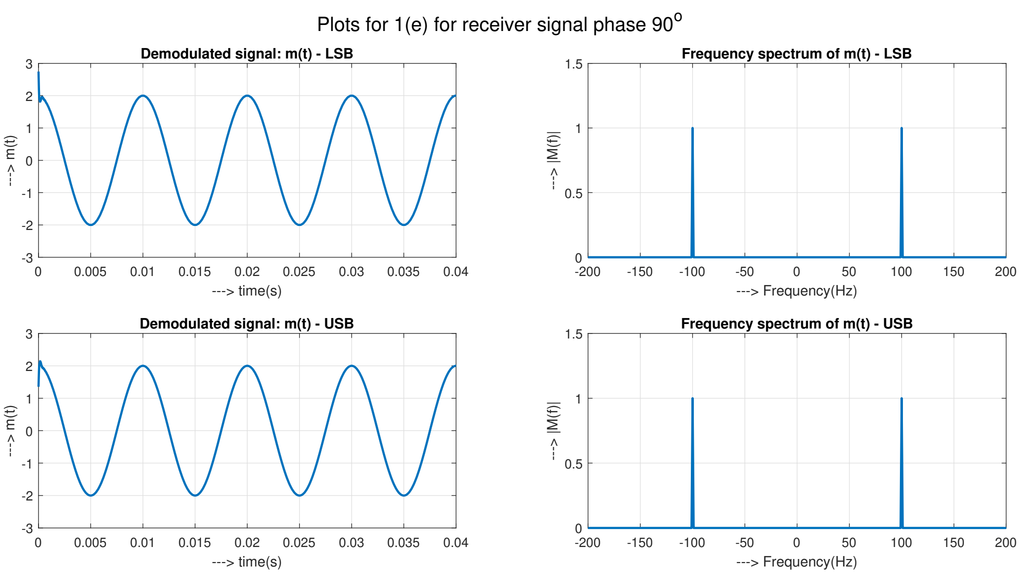

Figure 5: Demodulated signal and its frequency spectrum for receiver signal $A_c cos(2\pi f_c t + \pi/2)$

3.2 Question 2

To solve this question, we create time vector, frequency vector, message signal vector, carrier signal vector and SSB-SC signal vectors as shown below:

```
Ac = 1; A1 = 1; A2 = 1;
f1 = 100;   f2 = 200;   fc = 2000;  df = 1;
fs = 50000;                 %Sampling Frequency
t = [0:fs-1]*1/fs;          %Time vector
f = [-fs/2:fs/2-1]*df;      %Frequency vector

m1 = A1*cos(2*pi*f1*t);     m2 = A2*cos(2*pi*f2*t);
m_t = m1 + m2;              %Message signal vector
c_t = Ac*cos(2*pi*fc*t);    %Carrier signal vector
```

```
m_ht = imag(hilbert(m_t));

ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);      %SSB-U signal vector
ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);      %SSB-L signal vector
```

The MATLAB code to plot $A_1 cos(2\pi f_1 t)$, $A_2 cos(2\pi f_2 t)$ and carrier signal $A_c cos(2\pi f_c t)$ (part (a) and (b)) is as follows:

```
%<===== 2(a) ======>
subplot(3, 1, 1);             %Plot the signals
plot(t, m1, 'LineWidth', 1.7);
title('2(a) A_1cos(2\pif_1t) signal');
xlabel('---> time(s)');      ylabel('---> Signal');
xlim([0 0.02]);               grid on;

%<===== 2(b) ======>
subplot(3, 1, 2);
plot(t, m2,'LineWidth', 1.7);
title('2(b) A_2cos(2\pif_2t) signal');
xlabel('---> time(s)');      ylabel('---> Signal');
xlim([0 0.02]);               grid on;

subplot(3, 1, 3);
plot(t, c_t,'LineWidth', 1.7);
title('2(b) Carrier signal: A_ccos(2\pif_ct)');
xlabel('---> time(s)');      ylabel('---> Signal');
xlim([0 0.02]);               grid on;
```

The plots of $A_1 cos(2\pi f_1 t)$, $A_2 cos(2\pi f_2 t)$ and $A_c cos(2\pi f_c t)$ is as follows:

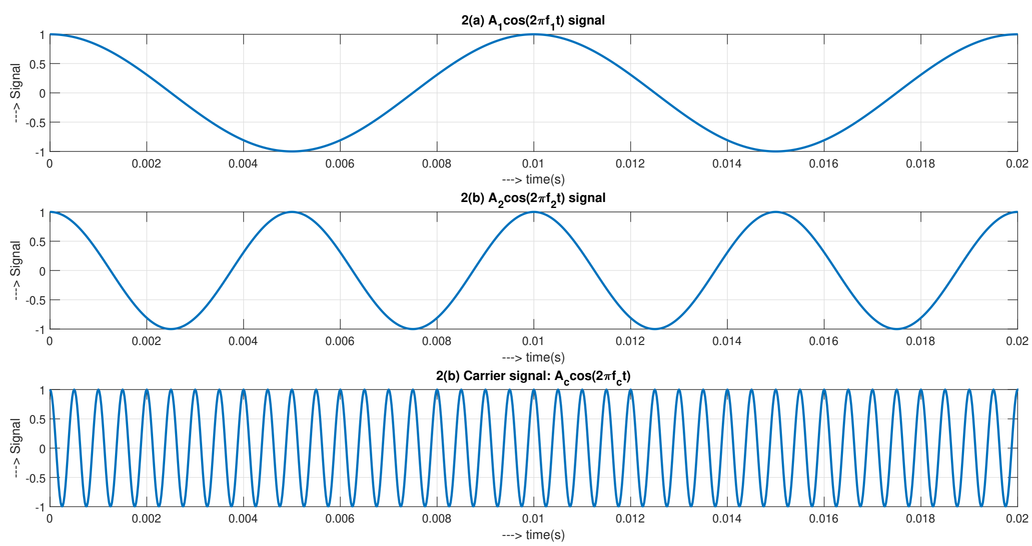

Figure 6: $A_1 cos(2\pi f_1 t)$, $A_2 cos(2\pi f_2 t)$ and $A_c cos(2\pi f_c t)$ signals.

The MATLAB code to plot SSB-SC signal for both USB and LSB transmission (part(c)) is as follows:

```
subplot(2,1,1);
plot(t, ssbl, 'LineWidth', 1.5);
title('SSB-SC signal - LSB');
xlabel('---> time(s)');     ylabel('---> \phi(t) - LSB');
xlim([0 0.02]);             ylim([-3 3]);
yticks(-5:1:5);             grid on;

subplot(2,1,2);
plot(t, ssbu,  'LineWidth', 1.5);
title('SSB-SC signal - USB');
xlabel('---> time(s)');     ylabel('---> \phi(t) - USB');
xlim([0 0.02]);             ylim([-3 3]);
yticks(-5:1:5);             grid on;
sgtitle('Plots for 2(c)');
```

The plot of SSB-SC signal is as follows:

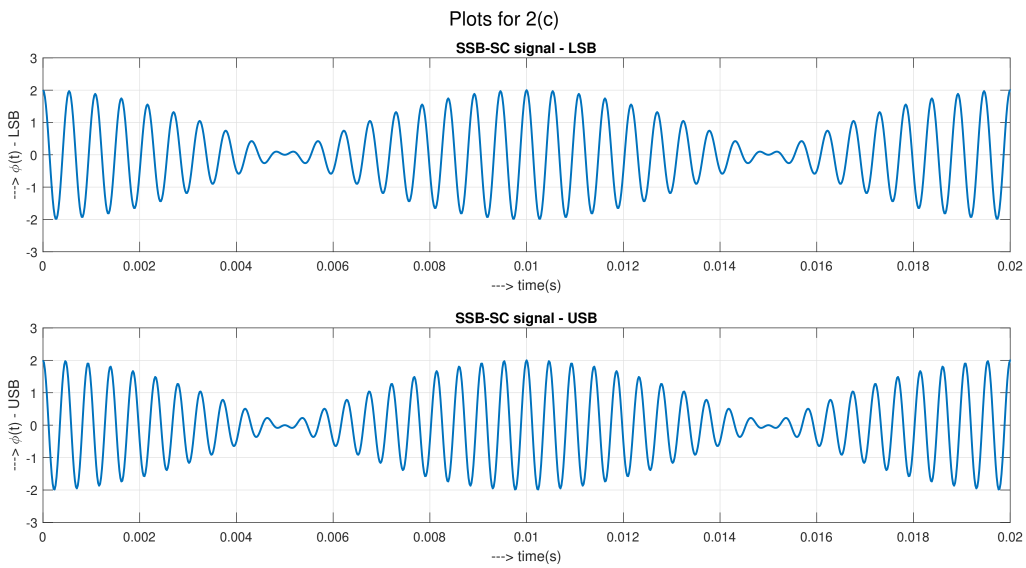

Figure 7: SSB-SC signal($\phi_{SSB}(t)$)

The MATLAB code to plot frequency spectrum of the signals (part(d)) is as follows:

```
fftM = fftshift(fft(m_t))/length(m_t);       %FFT of message signal
fftC = fftshift(fft(c_t))/length(c_t);       %FFT of carrier signal
fftSSBL = fftshift(fft(ssbl))/length(ssbl); %FFT of SSB-L signal
fftSSBU = fftshift(fft(ssbu))/length(ssbu); %FFT of SSB-U signal
f = [-fs/2:fs/2-1]*df;      %Frequency vector

sgtitle('Plots for 2(d)');
subplot(2,2,1);
plot(f, abs(fftM), 'LineWidth', 1.7);
title('Frequency spectrum of m(t)');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-300 300]);               ylim([0 0.7]);
grid on;

```

```
    subplot(2,2,2);
    plot(f, abs(fftC), 'LineWidth', 1.7);
    title('Frequency spectrum of c(t)');
    xlabel('---> Frequency(Hz)');   ylabel('---> |C(f)|');
    xlim([-2500 2500]);             ylim([0 0.7]);
    grid on;

    subplot(2,2,3);
    plot(f, abs(fftSSBL), 'LineWidth', 1.7);
    title('Frequency spectrum of \phi_{SSB}(t) - LSB');
    xlabel('---> Frequency(Hz)');   ylabel('---> |\phi_{SSB}(f)|');
    xlim([-2500 2500]);             ylim([0 0.7]);
    grid on

    subplot(2,2,4);
    plot(f, abs(fftSSBU), 'LineWidth', 1.7);
    title('Frequency spectrum of \phi_{SSB}(t) - USB');
    xlabel('---> Frequency(Hz)');   ylabel('---> |\phi_{SSB}(f)|');
    xlim([-2500 2500]);             ylim([0 0.7]);
    grid on
```

The plot of the frequency spectrum of the signals is as follows:

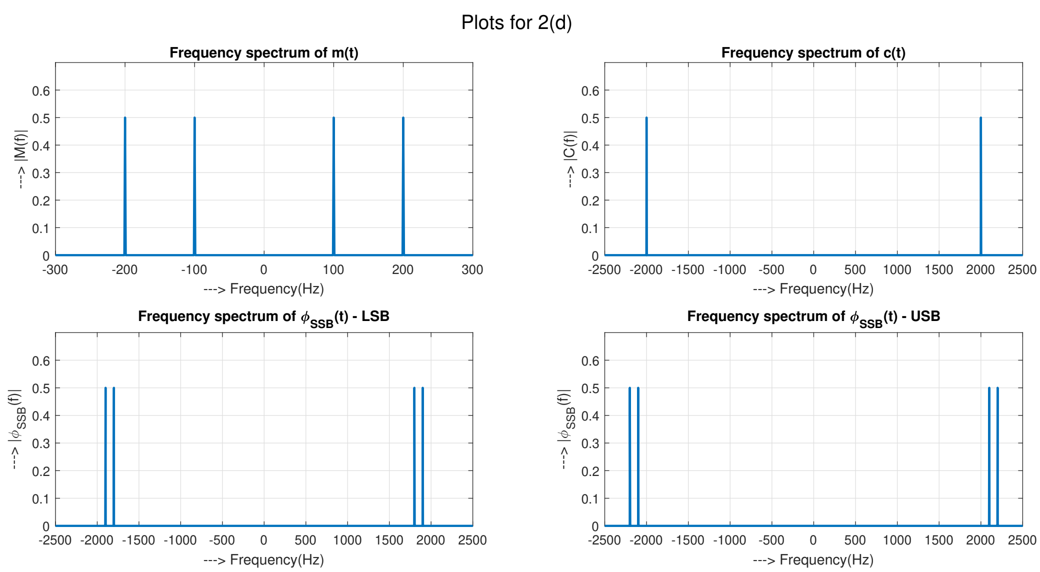

Figure 8: Frequency spectrum of $m(t)$, $c(t)$, $\phi_{SSB-L}(t)$ and $\phi_{SSB-U}(t)$ signals.

Here,

- The tone of message signal is at 100 and 200 Hz.
- The tone of carrier signal is at 2 KHz.
- The tone of $\phi_{SSB-L}(t)$ is at 1.8 and 1.9 KHz.
- The tone of $\phi_{SSB-U}(t)$ is at 2.1 and 2.2 KHz.

The MATLAB code to demodulate the transmission signal when the receiver generates $A_c cos(2\pi f_c t)$ (part 2(e1)) is shown below:

```
%<=======2(e1)========>
ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);
ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);

r1 = Ac*cos(2*pi*fc*t);     %Signal generated by receiver

m1ssbl = r1.*ssbl;          %Multiply with receiver generated signal
m1ssbu = r1.*ssbu;

m1ssbl = lowpass(m1ssbl, 150, fs);  %Pass through low pass filter
m1ssbu = lowpass(m1ssbu, 150, fs);

m1ssbl = m1ssbl/(Ac^2/2);       %Divide by Ac^2/2
m1ssbu = m1ssbu/(Ac^2/2);

fftM1ssbl = fftshift(fft(m1ssbl))/length(m1ssbl);   %Find FFT
fftM1ssbu = fftshift(fft(m1ssbu))/length(m1ssbu);
f = [-fs/2:fs/2-1]*df;              %Frequency vector

sgtitle('Plots for 2(e) for receiver signal phase 0^o');

subplot(2,2,1);
plot(t, m1ssbl, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - LSB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.02]);             ylim([-2 2]);
yticks(-5:1:5);
grid on;

subplot(2,2,2);
plot(f, abs(fftM1ssbl), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - LSB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-400 400]);               ylim([0 0.6]);
grid on;

subplot(2,2,3);
plot(t, m1ssbu, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - USB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.02]);             ylim([-2 2]);
yticks(-5:1:5);
grid on;

subplot(2,2,4);
plot(f, abs(fftM1ssbu), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - USB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-400 400]);               ylim([0 0.6]);
grid on;
```

The Plot of the demodulated signal along with its frequency spectrum for SSB-SC signal for both LSB and USB transmission is shown below:

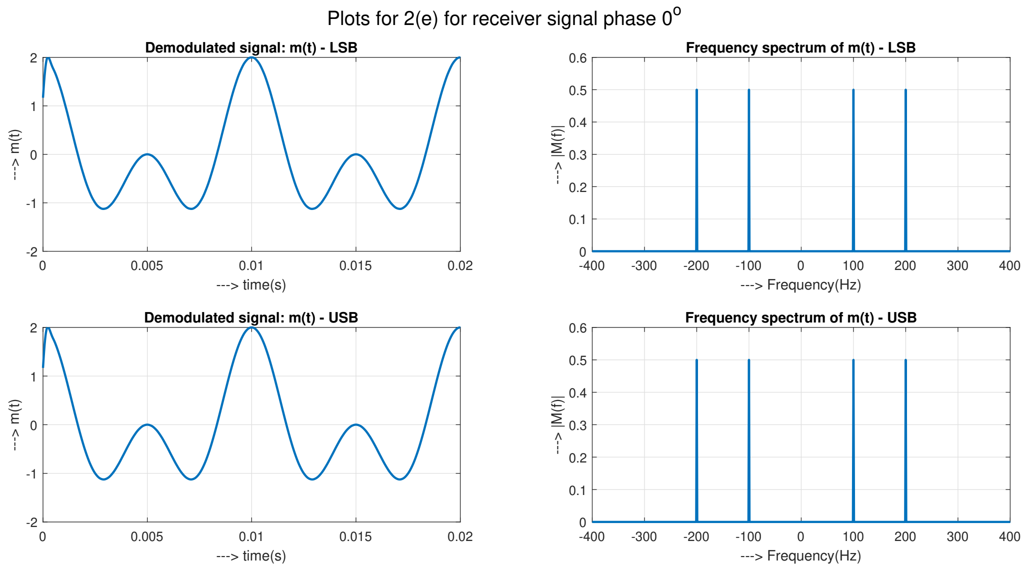

Figure 9: Demodulated signal and its frequency spectrum for receiver signal $A_c cos(2\pi f_c t)$

The MATLAB code to demodulate the transmission signal when the receiver generates $A_c cos(2\pi f_c t + \phi)$ $(\phi = 90^o)$ (part 2(e2)) is as follows:

```
%<=======2(e2)========>
ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);     %SSB-U signal
ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);     %SSB-L signal
r2 = Ac*cos(2*pi*fc*t + pi/2);                  %Receiver signal
m2ssbl = r2.*ssbl;                  %Multiply with receiver generated signal
m2ssbu = r2.*ssbu;
m2ssbl = lowpass(m2ssbl, 250, fs);          %Pass through low pass filter
m2ssbu = lowpass(m2ssbu, 250, fs);
m2ssbl = m2ssbl/(Ac^2/2);                   %Divide by Ac^2/s
m2ssbu = m2ssbu/(Ac^2/2);
m2ssbl = imag(hilbert(m2ssbl));             %Phase shift by pi/2
m2ssbu = -imag(hilbert(m2ssbu));
fftM2ssbl = fftshift(fft(m2ssbl))/length(m2ssbl);    %Find FFT
fftM2ssbu = fftshift(fft(m2ssbu))/length(m2ssbu);

sgtitle('Plots for 2(e) for receiver signal phase 90^o');
subplot(2,2,1);
plot(t, m2ssbl, 'LineWidth', 1.7);
title('Demodulated signal: m(t) - LSB');
xlabel('---> time(s)');     ylabel('---> m(t)');
xlim([0 0.02]);             ylim([-2 2]);
yticks(-5:1:5);             grid on;

subplot(2,2,2);
plot(f, abs(fftM2ssbl), 'LineWidth', 1.7);
title('Frequency spectrum of m(t) - LSB');
xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
xlim([-400 400]);               ylim([0 0.6]);
grid on;

```

```
    subplot(2,2,3);
    plot(t, m2ssbu, 'LineWidth', 1.7);
    title('Demodulated signal: m(t) - USB');
    xlabel('---> time(s)');      ylabel('---> m(t)');
    xlim([0 0.02]);              ylim([-2 2]);
    yticks(-5:1:5);              grid on;

    subplot(2,2,4);
    plot(f, abs(fftM2ssbu), 'LineWidth', 1.7);
    title('Frequency spectrum of m(t) - USB');
    xlabel('---> Frequency(Hz)');   ylabel('---> |M(f)|');
    xlim([-400 400]);               ylim([0 0.6]);
    grid on;
```

The plot of the demodulated signal along with its frequency spectrum for SSB-SC signal for both LSB and USB transmission is shown below:

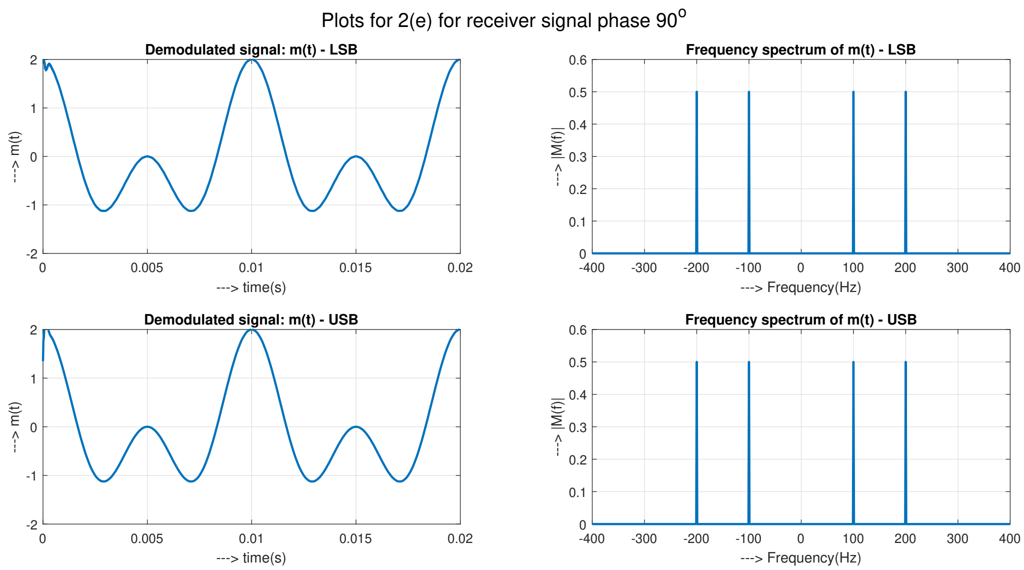

Figure 10: Demodulated signal and its frequency spectrum for receiver signal $A_c cos(2\pi f_c t + \pi/2)$

4 Discussion and Conclusion

4.1 Question 1

We plotted the message and carrier signal along with its frequency spectrum. We also plotted the SSB-SC modulated signal (both USB and LSB transmission) with its frequency spectrum. Finally, we replicated the function of synchronous detector to demodulate the transmission signal and plotted the demodulated signal with its frequency spectrum.

Some of the noteworthy observations are as follows:

- The given message signal is a single tone signal.
- The tone of the message signal is at 100 Hz and that of the carrier signal is at 2 KHz.
- In SSB-L signal, the tone of signal is at 1.9 KHz.

- In SSB-U signal, the tone of signal is at 2.1 KHz.
- Using synchronous detector, we were able to get back the original signal from the transmission signal for both USB and LSB transmissions.

4.2 Question 2

In this question too, we plotted the message signal, the carrier signal and the SSB-SC modulated signal along with its frequency spectrum. We also demodulated the transmission signal and plotted its frequency spectrum.

Some of the noteworthy observations are as follows:

- The given message signal is a multi tone signal.
- In the frequency spectrum of message signal, the tones are at 100 and 200 Hz. Similarly, for carrier signal, the tone is at 2 KHz.
- In SSB-L signal, the tone of the signal is at 1.8 and 1.9 KHz and in SSB-U signal, the tone of the signal is at 2.1 and 2.2 KHz.
- Using synchronous detector, we were able to get back the original signal from the transmission signal for both USB and LSB transmissions.