

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.

- Plot the message signal $m(t) = A_m \cos(2\pi f_m t)$ for complete 4 cycles.
- 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 $m(t)$.
- Plot the SSB-SC signal $\phi_{SSB}(t)$ for both USB and LSB transmissions.
- Plot the frequency spectrum of the signals in part (a), (b) and (c).
- 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 $\phi = 90^\circ$.

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.

- Plot the signal $A_1 \cos(2\pi f_1 t)$ for complete two cycles.
- 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 $A_1 \cos(2\pi f_1 t)$.
- 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.
- 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 $\phi = 90^\circ$.

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:

```

1      Am = 2;          Ac = 2;          %Parameters
2      fm = 100;        fc = 2000;
3      fs = 50000;      df = 1;
4      t = [0:fs-1]*1/fs;          %Time vector
5      f = [-fs/2:fs/2-1]*df;      %Frequency vector
6
7      m_t = Am*cos(2*pi*fm*t);      %Message signal
8      c_t = Ac*cos(2*pi*fc*t);      %Carrier signal
9      m_ht = imag(hilbert(m_t));
10
11     ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC signal for LSB transmission
12     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.

```

1      fftM = fftshift(fft(m_t))/length(m_t);          %Message signal
2      fftC = fftshift(fft(c_t))/length(c_t);          %Carrier signal
3      fftSSBL = fftshift(fft(ssbl))/length(ssbl); %SSB-SC signal for LSB transmission
4      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:

```

1      ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-L signal
2      r2 = Ac*cos(2*pi*fc*t + pi/2);          %signal from receiver
3      m2ssbl = r2.*ssbl;          %Multiply by Accos(2*pi*fc*t + phi)
4      m2ssbl = lowpass(m2ssbl, 150, fs);          %Pass through low pass filter
5      m2ssbl = m2ssbl/(Ac^2/2);          %Divide by Ac/2
6      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:

```
1 Am = 2;          Ac = 2;          %Parameters
2 fm = 100;        fc = 2000;        fs = 50000;    df = 1;
3 t = [0:fs-1]*1/fs;          %Time vector
4 f = [-fs/2:fs/2-1]*df;      %Frequency vector
5 m_t = Am*cos(2*pi*fm*t);     %Message signal
6 c_t = Ac*cos(2*pi*fc*t);     %Carrier signal
7 m_ht = imag(hilbert(m_t));   %hilbert transform of message signal
8 ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-SC - LSB
9 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 %<===== 1(a) =====>
2 subplot(2, 1, 1);          %Plot the message signal
3 plot(t, m_t, 'LineWidth', 1.5);
4 title('1(a) Message Signal: A_m cos(2\pi f_m t)');
5 xlabel('----> time(s)');    ylabel('----> m(t)');
6 xlim([0 0.04]);            ylim([-3 3]);
7 grid on;
8 %<===== 1(b) =====>
9 subplot(2, 1, 2);          %Plot the carrier signal
10 plot(t, c_t, 'LineWidth', 1.5);
11 title('1(b) Carrier Signal: A_c cos(2\pi f_c t)');
12 xlabel('----> time(s)');    ylabel('----> c(t)');
13 xlim([0 0.04]);            ylim([-3 3]);
14 grid on;
```

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

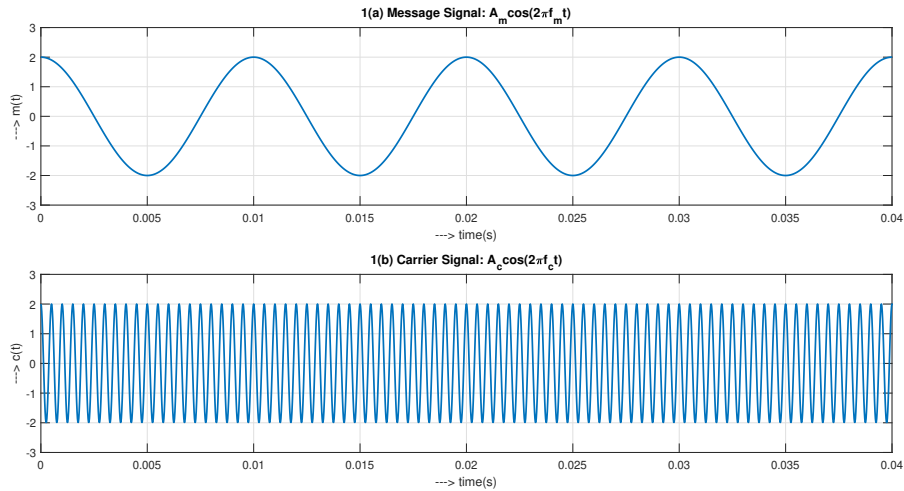


Figure 1: Message signal and carrier signal

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

```

1  %<===== 1(c) =====>
2  subplot(2,1,1);          %Plot the signal
3  plot(t, ssbl, 'LineWidth', 1.5);
4  title('SSB-SC signal - LSB');
5  xlabel('---> time(s)');   ylabel('---> \phi(t) - LSB');
6  xlim([0 0.04]);          ylim([-5 5]);
7  yticks(-5:1:5);          grid on;
8
9  subplot(2,1,2);          %Plot the signal
10 plot(t, ssbu, 'LineWidth', 1.5);
11 title('SSB-SC signal - USB');
12 xlabel('---> time(s)');   ylabel('---> \phi(t) - USB');
13 xlim([0 0.04]);          ylim([-5 5]);
14 yticks(-5:1:5);          grid on;
15 sgtitle('Plots for 1(c)');

```

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

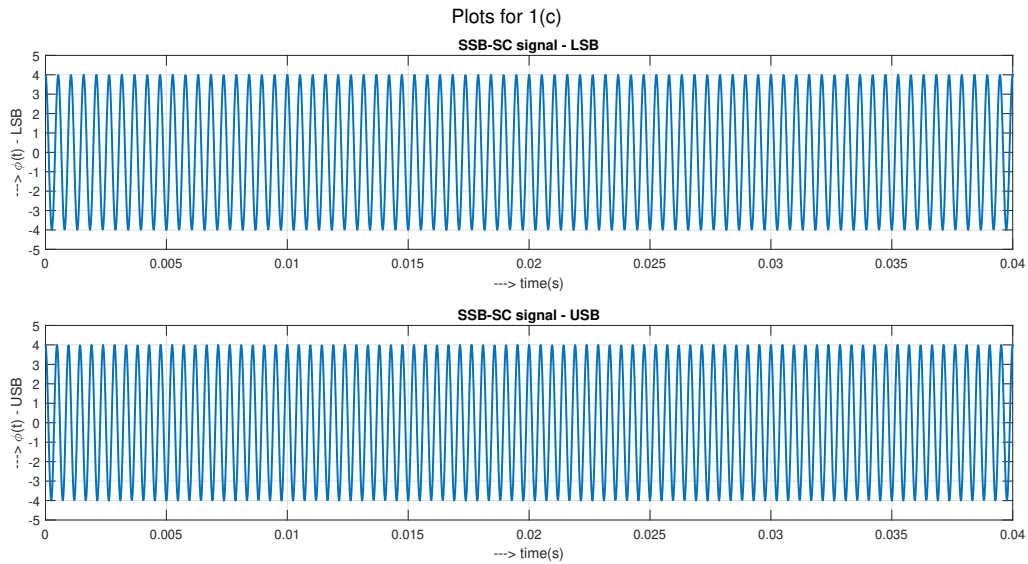


Figure 2: SSB-SC signal ($\phi_{SSB}(t)$)

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

```

1  %<===== 1(d) =====>
2  f = [-fs/2:fs/2-1]*df;          %Frequency vector
3  fftM = fftshift(fft(m_t))/length(m_t); %Find FFT of the signals
4  fftC = fftshift(fft(c_t))/length(c_t);
5  fftSSBL = fftshift(fft(ssbl))/length(ssbl);
6  fftSSBU = fftshift(fft(ssbu))/length(ssbu);
7
8  subplot(2,2,1);          %Plot FFT of message signal
9  plot(f, abs(fftM), 'LineWidth', 1.7);
10 title('Frequency spectrum of m(t)');

```

```

11 xlabel('---> Frequency(Hz)'); ylabel('---> |M(f)|');
12 xlim([-200 200]); ylim([0 1.2]);
13 grid on;
14
15 subplot(2,2,2); %Plot FFT of carrier signal
16 plot(f, abs(fftC), 'LineWidth', 1.7);
17 title('Frequency spectrum of c(t)');
18 xlabel('---> Frequency(Hz)'); ylabel('---> |C(f)|');
19 xlim([-2500 2500]); ylim([0 1.2]);
20 grid on;
21
22 subplot(2,2,3); %Plot FFT of SSB-SC - LSB signal
23 plot(f, abs(fftSSBL), 'LineWidth', 1.7);
24 title('Frequency spectrum of \phi_{SSB}(t) - LSB');
25 xlabel('---> Frequency(Hz)'); ylabel('---> |\phi_{SSB}(f)|');
26 xlim([-2500 2500]); ylim([0 2.5]);
27 grid on;
28
29 subplot(2,2,4); %Plot FFT of SSB-SC - USB signal
30 plot(f, abs(fftSSBU), 'LineWidth', 1.7);
31 title('Frequency spectrum of \phi_{SSB}(t) - USB');
32 xlabel('---> Frequency(Hz)'); ylabel('---> |\phi_{SSB}(f)|');
33 xlim([-2500 2500]); ylim([0 2.5]);
34 grid on; sgtitle('Plots for 1(d)');

```

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

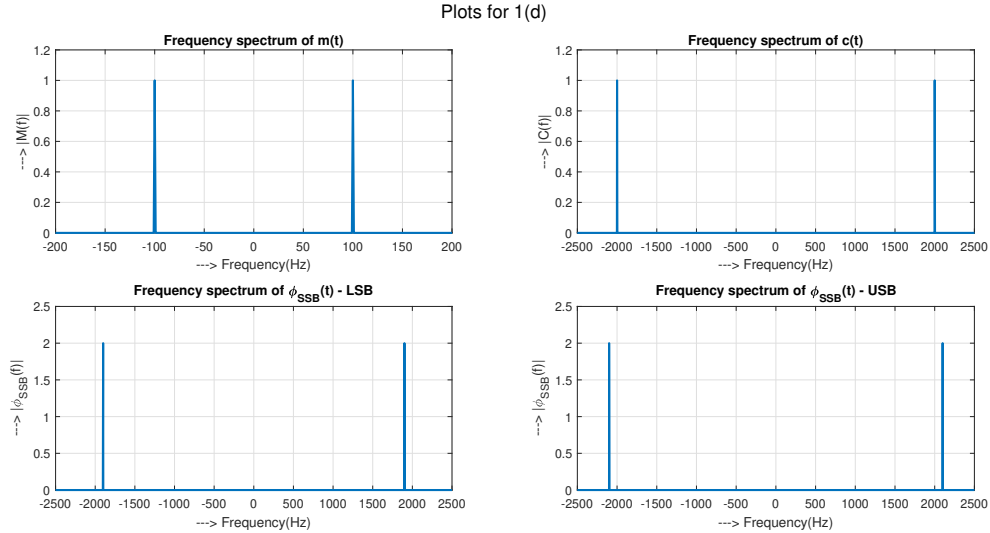


Figure 3: Frequency spectrum of $m(t)$, $c(t)$, $\phi_{SSB-L}(t)$ and $\phi_{SSB-U}(t)$

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      %<=====1(e1)=====>
2      ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t); %SSB-L signal
3      ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t); %SSB-U signal
4      r1 = Ac*cos(2*pi*fc*t);      %Receiver generated signal
5
6      mlssbl = r1.*ssbl;      %Multiply with receiver generated signal
7      mlssbu = r1.*ssbu;
8
9      mlssbl = lowpass(mlssbl, 150, fs); %Pass through low pass filter
10     mlssbu = lowpass(mlssbu, 150, fs);
11
12     mlssbl = mlssbl/(Ac/2);      %Divide by Ac/2
13     mlssbu = mlssbu/(Ac/2);
14
15     fftMlssbl = fftshift(fft(mlssbl))/length(mlssbl);      %Find FFT of ...
16     demodulated signals
17     fftMlssbu = fftshift(fft(mlssbu))/length(mlssbu);
18     f = [-fs/2:fs/2-1]*df;      %Frequency vector
19
20     sgtitle('Plots for 1(e) for receiver signal phase 0^o');
21     subplot(2,2,1);
22     plot(t, mlssbl, 'LineWidth', 1.7);
23     title('Demodulated signal: m(t) - LSB');
24     xlabel('---> time(s)');      ylabel('---> m(t)');
25     xlim([0 0.04]);      ylim([-3 3]);
26     yticks(-5:1:5);
27     grid on;
28
29     subplot(2,2,2);
30     plot(f, abs(fftMlssbl), 'LineWidth', 1.7);
31     title('Frequency spectrum of m(t) - LSB');
32     xlabel('---> Frequency(Hz)');      ylabel('---> |M(f)|');
33     xlim([-200 200]);      ylim([0 1.5]);
34     grid on;
35
36     subplot(2,2,3);
37     plot(t, mlssbu, 'LineWidth', 1.7);
38     title('Demodulated signal: m(t) - USB');
39     xlabel('---> time(s)');      ylabel('---> m(t)');
40     xlim([0 0.04]);      ylim([-3 3]);
41     yticks(-5:1:5);
42     grid on;
43
44     subplot(2,2,4);
45     plot(f, abs(fftMlssbu), 'LineWidth', 1.7);
46     title('Frequency spectrum of m(t) - USB');
47     xlabel('---> Frequency(Hz)');      ylabel('---> |M(f)|');
48     xlim([-200 200]);      ylim([0 1.5]);
49     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:

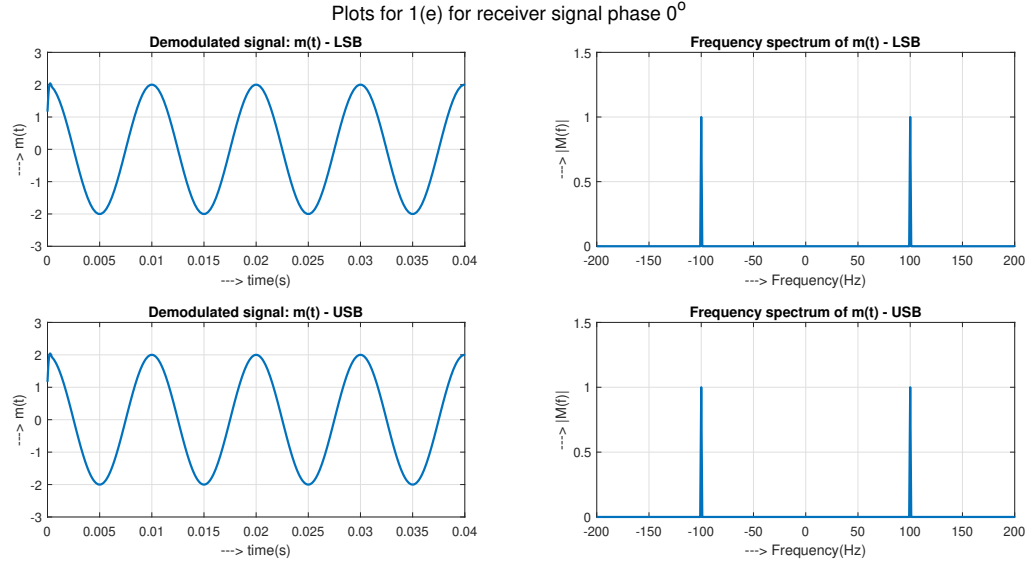


Figure 4: Demodulated signal and its frequency spectrum for receiver signal $A_c \cos(2\pi f_c t)$

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

```

1      %<=====1(e2)=====>
2      ssbl = m_t.*c_t + m_ht.*Ac.*sin(2*pi*fc*t);      %SSB-L signal
3      ssbu = m_t.*c_t - m_ht.*Ac.*sin(2*pi*fc*t);      %SSB-U signal
4      r2 = Ac*cos(2*pi*fc*t + pi/2);      %Receiver generated signal
5      m2ssbl = r2.*ssbl;      %Multiply with receiver generated signal
6      m2ssbu = r2.*ssbu;
7      m2ssbl = lowpass(m2ssbl, 150, fs);      %Pass through low pass filter
8      m2ssbu = lowpass(m2ssbu, 150, fs);
9      m2ssbl = m2ssbl/(Ac^2/2);      %Divide by Ac/2
10     m2ssbu = m2ssbu/(Ac^2/2);
11     m2ssbl = imag(hilbert(m2ssbl));      %shift phase by pi/2
12     m2ssbu = -imag(hilbert(m2ssbu));
13     fftM2ssbl = fftshift(fft(m2ssbl))/length(m2ssbl);      %Find FFT of the ...
14     demodulated signal
15     fftM2ssbu = fftshift(fft(m2ssbu))/length(m2ssbu);
16
17     sgtitle('Plots for 1(e) for receiver signal phase 90^o');
18     subplot(2,2,1);
19     plot(t, m2ssbl, 'LineWidth', 1.7);
20     title('Demodulated signal: m(t) - LSB');
21     xlabel('---> time(s)');      ylabel('---> m(t)');
22     xlim([0 0.04]);      ylim([-3 3]);
23     yticks(-5:1:5);      grid on;
24
25     subplot(2,2,2);
26     plot(f, abs(fftM2ssbl), 'LineWidth', 1.7);
27     title('Frequency spectrum of m(t) - LSB');
28     xlabel('---> Frequency(Hz)');      ylabel('---> |M(f)|');
29     xlim([-200 200]);      ylim([0 1.5]);
30     grid on;

```



```

30
31 subplot(2,2,3);
32 plot(t, m2ssbu, 'LineWidth', 1.7);
33 title('Demodulated signal: m(t) - USB');
34 xlabel('---> time(s)');      ylabel('---> m(t)');
35 xlim([0 0.04]);             ylim([-3 3]);
36 yticks(-5:1:5);             grid on;
37
38 subplot(2,2,4);
39 plot(f, abs(fftM2ssbu), 'LineWidth', 1.7);
40 title('Frequency spectrum of m(t) - USB');
41 xlabel('---> Frequency(Hz)'); ylabel('---> |M(f)|');
42 xlim([-200 200]);           ylim([0 1.5]);
43 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:

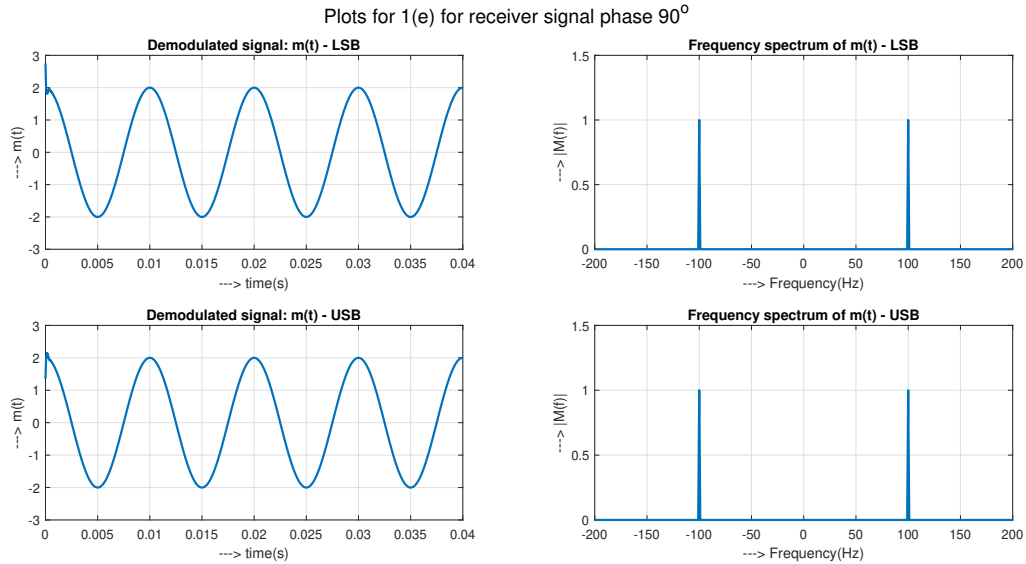


Figure 5: Demodulated signal and its frequency spectrum for receiver signal $A_c \cos(2\pi f_c t + \pi/2)$

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:

```

1  Ac = 1; A1 = 1; A2 = 1;
2  f1 = 100;   f2 = 200;   fc = 2000;   df = 1;
3  fs = 50000;                               %Sampling Frequency
4  t = [0:fs-1]*1/fs;                         %Time vector
5  f = [-fs/2:fs/2-1]*df;                     %Frequency vector
6
7  m1 = A1*cos(2*pi*f1*t);                     m2 = A2*cos(2*pi*f2*t);
8  m_t = m1 + m2;                             %Message signal vector
9  c_t = Ac*cos(2*pi*fc*t);                   %Carrier signal vector

```

```

10     m_ht = imag(hilbert(m_t));
11
12     ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);    %SSB-U signal vector
13     ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);    %SSB-L signal vector

```

The MATLAB code to plot $A_1\cos(2\pi f_1t)$, $A_2\cos(2\pi f_2t)$ and carrier signal $A_c\cos(2\pi f_ct)$ (part (a) and (b)) is as follows:

```

1     %<===== 2(a) =====>
2     subplot(3, 1, 1);           %Plot the signals
3     plot(t, m1, 'LineWidth', 1.7);
4     title('2(a) A_1cos(2\pif_1t) signal');
5     xlabel('---> time(s)');     ylabel('---> Signal');
6     xlim([0 0.02]);            grid on;
7
8     %<===== 2(b) =====>
9     subplot(3, 1, 2);
10    plot(t, m2, 'LineWidth', 1.7);
11    title('2(b) A_2cos(2\pif_2t) signal');
12    xlabel('---> time(s)');     ylabel('---> Signal');
13    xlim([0 0.02]);            grid on;
14
15    subplot(3, 1, 3);
16    plot(t, c_t, 'LineWidth', 1.7);
17    title('2(b) Carrier signal: A_ccos(2\pif_fc t)');
18    xlabel('---> time(s)');     ylabel('---> Signal');
19    xlim([0 0.02]);            grid on;

```

The plots of $A_1\cos(2\pi f_1t)$, $A_2\cos(2\pi f_2t)$ and $A_c\cos(2\pi f_ct)$ is as follows:

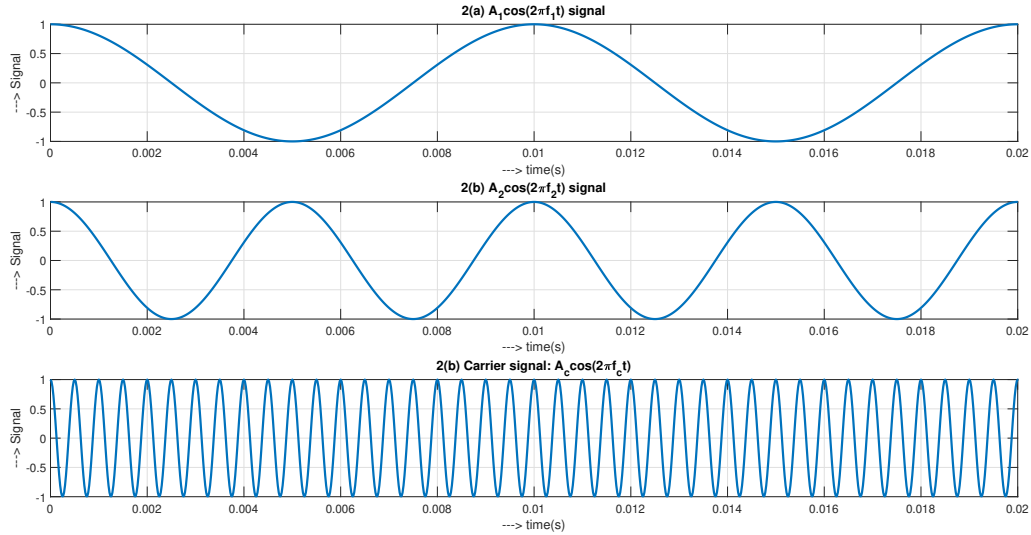


Figure 6: $A_1\cos(2\pi f_1t)$, $A_2\cos(2\pi f_2t)$ and $A_c\cos(2\pi f_ct)$ signals.

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

```

1 subplot(2,1,1);
2 plot(t, ssbl, 'LineWidth', 1.5);
3 title('SSB-SC signal - LSB');
4 xlabel('---> time(s)'); ylabel('---> \phi(t) - LSB');
5 xlim([0 0.02]); ylim([-3 3]);
6 yticks(-5:1:5); grid on;
7
8 subplot(2,1,2);
9 plot(t, ssbu, 'LineWidth', 1.5);
10 title('SSB-SC signal - USB');
11 xlabel('---> time(s)'); ylabel('---> \phi(t) - USB');
12 xlim([0 0.02]); ylim([-3 3]);
13 yticks(-5:1:5); grid on;
14 sgtitle('Plots for 2(c)');

```

The plot of SSB-SC signal is as follows:

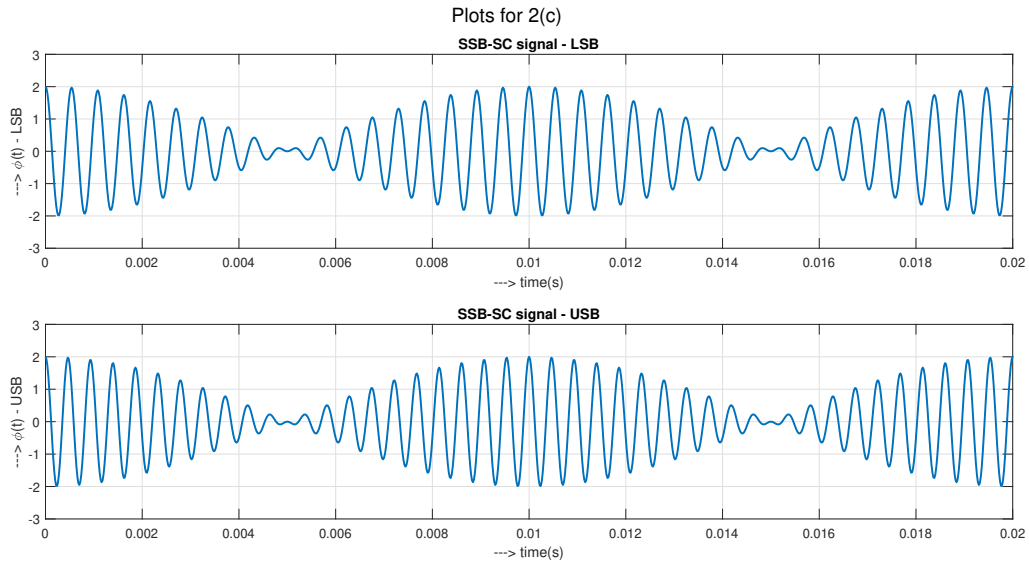


Figure 7: SSB-SC signal($\phi_{SSB}(t)$)

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

```

1 fftM = fftshift(fft(m_t))/length(m_t); %FFT of message signal
2 fftC = fftshift(fft(c_t))/length(c_t); %FFT of carrier signal
3 fftSSBL = fftshift(fft(ssbl))/length(ssbl); %FFT of SSB-L signal
4 fftSSBU = fftshift(fft(ssbu))/length(ssbu); %FFT of SSB-U signal
5 f = [-fs/2:fs/2-1]*df; %Frequency vector
6
7 sgtitle('Plots for 2(d)');
8 subplot(2,2,1);
9 plot(f, abs(fftM), 'LineWidth', 1.7);
10 title('Frequency spectrum of m(t)');
11 xlabel('---> Frequency(Hz)'); ylabel('---> |M(f)|');
12 xlim([-300 300]); ylim([0 0.7]);
13 grid on;
14

```

```

15 subplot(2,2,2);
16 plot(f, abs(fftC), 'LineWidth', 1.7);
17 title('Frequency spectrum of c(t)');
18 xlabel('---> Frequency(Hz)'); ylabel('---> |C(f)|');
19 xlim([-2500 2500]); ylim([0 0.7]);
20 grid on;
21
22 subplot(2,2,3);
23 plot(f, abs(fftSSBL), 'LineWidth', 1.7);
24 title('Frequency spectrum of \phi_{SSB}(t) - LSB');
25 xlabel('---> Frequency(Hz)'); ylabel('---> |\phi_{SSB}(f)|');
26 xlim([-2500 2500]); ylim([0 0.7]);
27 grid on
28
29 subplot(2,2,4);
30 plot(f, abs(fftSSBU), 'LineWidth', 1.7);
31 title('Frequency spectrum of \phi_{SSB}(t) - USB');
32 xlabel('---> Frequency(Hz)'); ylabel('---> |\phi_{SSB}(f)|');
33 xlim([-2500 2500]); ylim([0 0.7]);
34 grid on

```

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

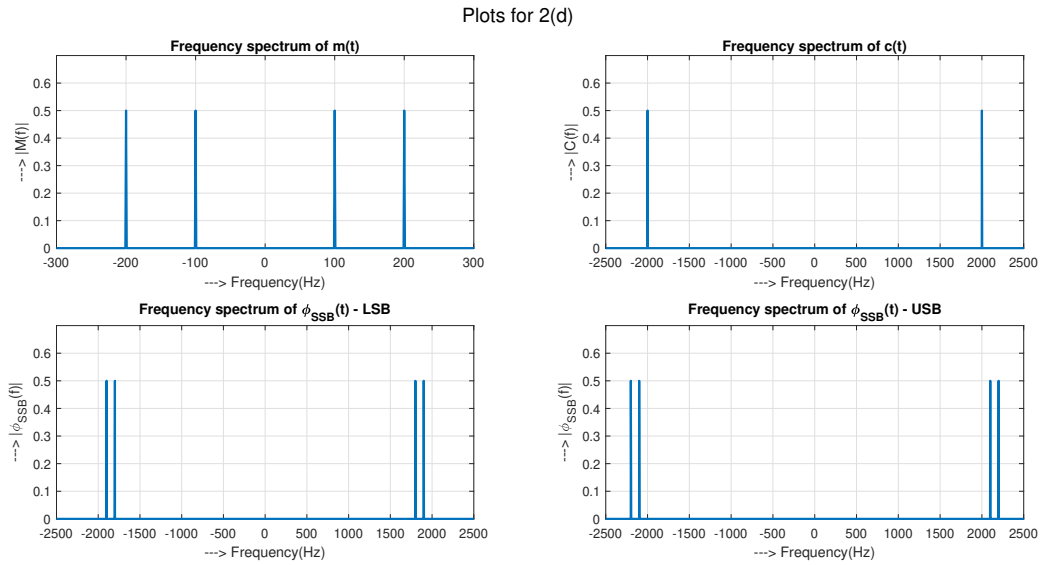


Figure 8: Frequency spectrum of $m(t)$, $c(t)$, $\phi_{SSB-L}(t)$ and $\phi_{SSB-U}(t)$ signals.

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:

```

1      %<=====2(e1)=====>
2      ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);
3      ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);
4
5      r1 = Ac*cos(2*pi*fc*t);          %Signal generated by receiver
6
7      mlssbl = r1.*ssbl;                %Multiply with receiver generated signal
8      mlssbu = r1.*ssbu;
9
10     mlssbl = lowpass(mlssbl, 150, fs); %Pass through low pass filter
11     mlssbu = lowpass(mlssbu, 150, fs);
12
13     mlssbl = mlssbl/(Ac^2/2);          %Divide by Ac^2/2
14     mlssbu = mlssbu/(Ac^2/2);
15
16     fftMlssbl = fftshift(fft(mlssbl))/length(mlssbl); %Find FFT
17     fftMlssbu = fftshift(fft(mlssbu))/length(mlssbu);
18     f = [-fs/2:fs/2-1]*df;            %Frequency vector
19
20     sgtitle('Plots for 2(e) for receiver signal phase 0^o');
21
22     subplot(2,2,1);
23     plot(t, mlssbl, 'LineWidth', 1.7);
24     title('Demodulated signal: m(t) - LSB');
25     xlabel('---> time(s)');           ylabel('---> m(t)');
26     xlim([0 0.02]);                   ylim([-2 2]);
27     yticks(-5:1:5);
28     grid on;
29
30     subplot(2,2,2);
31     plot(f, abs(fftMlssbl), 'LineWidth', 1.7);
32     title('Frequency spectrum of m(t) - LSB');
33     xlabel('---> Frequency(Hz)');      ylabel('---> |M(f)|');
34     xlim([-400 400]);                  ylim([0 0.6]);
35     grid on;
36
37     subplot(2,2,3);
38     plot(t, mlssbu, 'LineWidth', 1.7);
39     title('Demodulated signal: m(t) - USB');
40     xlabel('---> time(s)');           ylabel('---> m(t)');
41     xlim([0 0.02]);                   ylim([-2 2]);
42     yticks(-5:1:5);
43     grid on;
44
45     subplot(2,2,4);
46     plot(f, abs(fftMlssbu), 'LineWidth', 1.7);
47     title('Frequency spectrum of m(t) - USB');
48     xlabel('---> Frequency(Hz)');      ylabel('---> |M(f)|');
49     xlim([-400 400]);                  ylim([0 0.6]);
50     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:

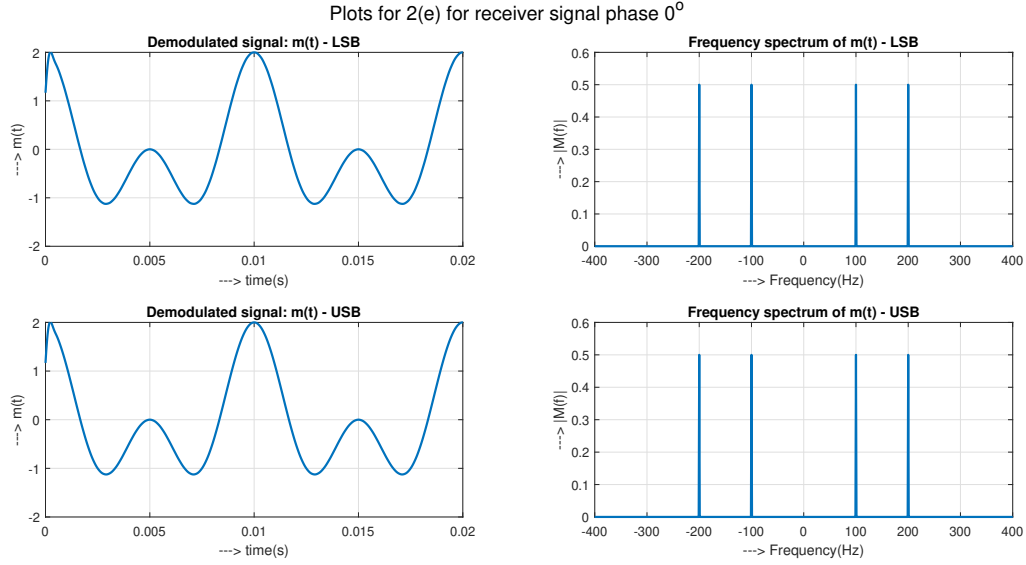


Figure 9: Demodulated signal and its frequency spectrum for receiver signal $A_c \cos(2\pi f_c t)$

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

```

1      %=====2 (e2)=====>
2      ssbu = m_t.*c_t - Ac.*m_ht.*sin(2*pi*fc*t);      %SSB-U signal
3      ssbl = m_t.*c_t + Ac.*m_ht.*sin(2*pi*fc*t);      %SSB-L signal
4      r2 = Ac*cos(2*pi*fc*t + pi/2);                  %Receiver signal
5      m2ssbl = r2.*ssbl;                               %Multiply with receiver generated signal
6      m2ssbu = r2.*ssbu;
7      m2ssbl = lowpass(m2ssbl, 250, fs);               %Pass through low pass filter
8      m2ssbu = lowpass(m2ssbu, 250, fs);
9      m2ssbl = m2ssbl/(Ac^2/2);                         %Divide by Ac^2/s
10     m2ssbu = m2ssbu/(Ac^2/2);
11     m2ssbl = imag(hilbert(m2ssbl));                   %Phase shift by pi/2
12     m2ssbu = -imag(hilbert(m2ssbu));
13     fftM2ssbl = fftshift(fft(m2ssbl))/length(m2ssbl); %Find FFT
14     fftM2ssbu = fftshift(fft(m2ssbu))/length(m2ssbu);
15
16     sgtitle('Plots for 2(e) for receiver signal phase 90^o');
17     subplot(2,2,1);
18     plot(t, m2ssbl, 'LineWidth', 1.7);
19     title('Demodulated signal: m(t) - LSB');
20     xlabel('---> time(s)');      ylabel('---> m(t)');
21     xlim([0 0.02]);              ylim([-2 2]);
22     yticks(-5:1:5);              grid on;
23
24     subplot(2,2,2);
25     plot(f, abs(fftM2ssbl), 'LineWidth', 1.7);
26     title('Frequency spectrum of m(t) - LSB');
27     xlabel('---> Frequency(Hz)'); ylabel('---> |M(f)|');
28     xlim([-400 400]);            ylim([0 0.6]);
29     grid on;
30

```

```

31 subplot(2,2,3);
32 plot(t, m2ssbu, 'LineWidth', 1.7);
33 title('Demodulated signal: m(t) - USB');
34 xlabel('---> time(s)');    ylabel('---> m(t)');
35 xlim([0 0.02]);          ylim([-2 2]);
36 yticks(-5:1:5);          grid on;
37
38 subplot(2,2,4);
39 plot(f, abs(fftM2ssbu), 'LineWidth', 1.7);
40 title('Frequency spectrum of m(t) - USB');
41 xlabel('---> Frequency(Hz)');    ylabel('---> |M(f)|');
42 xlim([-400 400]);          ylim([0 0.6]);
43 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:

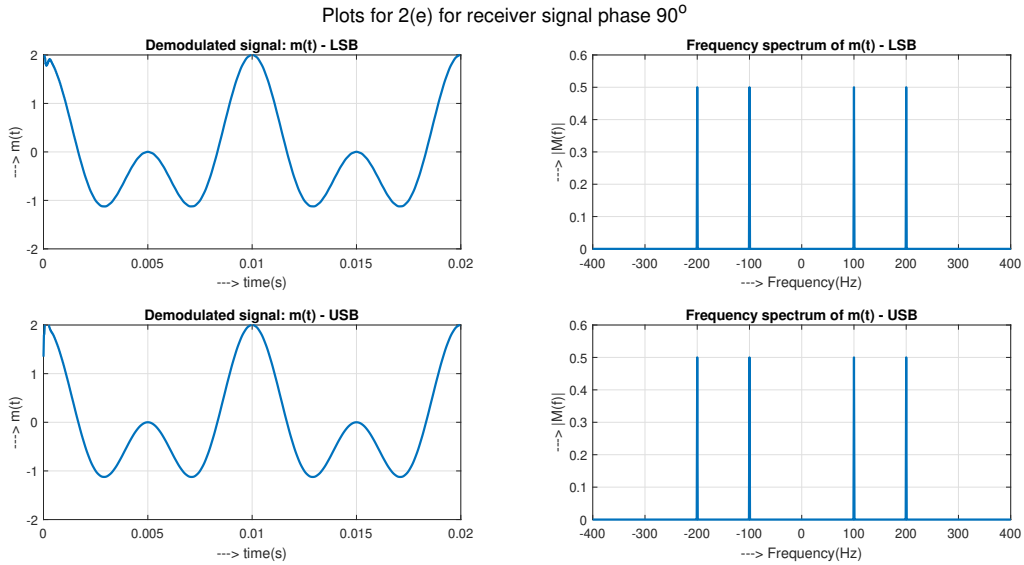


Figure 10: Demodulated signal and its frequency spectrum for receiver signal $A_c \cos(2\pi f_c t + \pi/2)$

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.