



Ministry of Higher Education

Higher Technological Institute

10th of Ramadan City

6th of October Branch

Electrical and Computer Engineering Department

Communication Systems

EET 139 (G 92)

MATLAB Task

Under Supervision: Prof. Dr. Ali Elmoghazy

Eng. Mahmoud hamed/ Eng. Kareem Badawi

Name / Muhammad Magdy Srour

ID/ 32015251

Table of Contents

1 - AM (Amplitude modulation)	5
1.1 – introduction	5
1.2 – time domain modulation equation	6
1.3 – modulation method	6
1.4 – de-modulation method	6
1.5 – MATLAB code	7
Set the code parameters	7
Generate message and carrier	7
Plotting the message and carrier	7
message and carrier	8
Modulation	8
Demodulation	8
Plotting modulated and demodulated signals	9
modulated and demodulated signals	9
2 - DSB-SC (Double-sideband suppressed-carrier transmission)	10
2.1 – introduction	10
2.2 – time domain modulation equation	10
2.3 – modulation method	11
2.4 – de-modulation method	11
2.5 – MATLAB code	11
Set the code parameters	11
Generate message and carrier	12
Plotting the message and carrier	12
message and carrier	12
Modulation	13
Demodulation	13
Plotting modulated and demodulated signals	13
modulated and demodulated signals	14
3 – SSB (Single-sideband modulation)	15
3.1 – introduction	15

3.2 – time domain modulation equation	16
3.3 – modulation method	16
3.4 – de-modulation method	17
3.5 – MATLAB code.....	17
Set the code parameters.....	17
Generate message and carrier	17
Plotting the message and carrier.....	18
message and carrier.....	18
Modulation.....	19
Demodulation.....	19
Plotting modulated and demodulated signals	19
modulated and demodulated signals	20
4 – MATLAB used functions	21
4.1 amdemod	21
4.2 butter	21

Figure 1 AM signal	5
Figure 2 AM modulation method	6
Figure 3 AM de-modulation method	6
Figure 4 AM Set the code parameters	7
Figure 5 AM Generate message and carrier	7
Figure 6 AM Plotting the message and carrier	7
Figure 7 AM message and carrier	8
Figure 8 AM Modulation equation	8
Figure 9 AM Demodulation equation	8
Figure 10 AM Plotting modulated and demodulated signals	9
Figure 11 AM modulated and demodulated signals	9
Figure 12 DSB-SC modulation method	11
Figure 13 DSB-SC demodulation method	11
Figure 14 DCB-SC Set the code parameters	11
Figure 15 DCB-SC Generate message and carrier	12
Figure 16 DSB-SC Plotting the message and carrier	12
Figure 17 DCB-SC message and carrier	12
Figure 18 DSB-SC Modulation equation	13
Figure 19 DSB-SC Demodulation equation	13
Figure 20 DSB-SC Plotting modulated and demodulated signals	13
Figure 21 DSB-SC modulated and demodulated signals	14
Figure 22 SSB signal	15
Figure 23 SSB modulation method 1	16
Figure 24 SSB modulation method 2	16
Figure 25 SSB de-modulation method	17
Figure 26 SSB Set the code parameters	17
Figure 27 SSB Generate message and carrier	17
Figure 28 SSB Plotting the message and carrier	18
Figure 29 SSB message and carrier	18
Figure 30 SSB(LSB) Modulation equation	19
Figure 31 SSB(LSB) Demodulation equation	19
Figure 32 SSB(LSB) Plotting modulated and demodulated signals	19
Figure 33 SSB(LSB) modulated and demodulated signals	20
Figure 34 amdemod function	21
Figure 35 butter function	21

1 - AM (Amplitude modulation)

1.1 – introduction

amplitude modulation (AM) is a modulation technique used in electronic communication, most commonly for transmitting information via a radio carrier wave. In amplitude modulation, the amplitude (signal strength) of the carrier wave is varied in proportion to that of the message signal being transmitted. The message signal is, for example, a function of the sound to be reproduced by a loudspeaker, or the light intensity of pixels of a television screen. This technique contrasts with frequency modulation, in which the frequency of the carrier signal is varied, and phase modulation, in which its phase is varied.

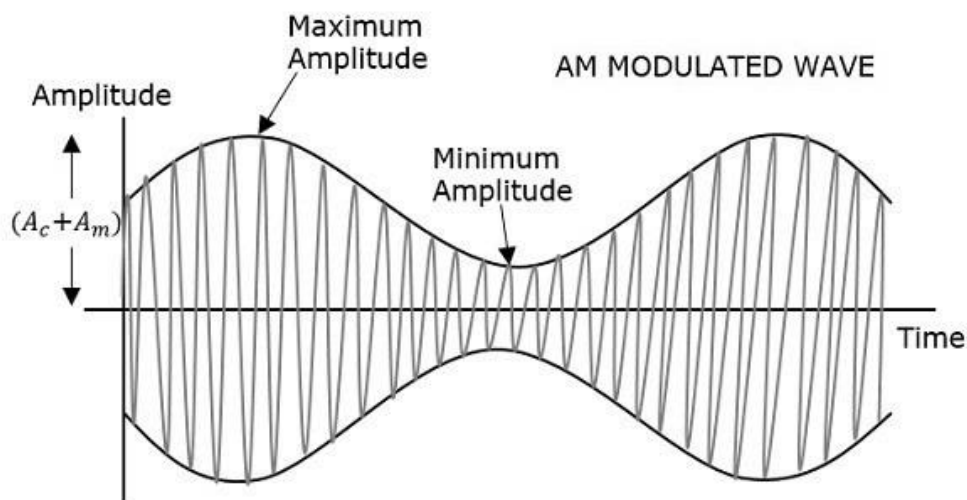


Figure 1 AM signal

1.2 – time domain modulation equation

$$S(t) = (A_c + m(t)) * \cos(W_c * t)$$

Where

$S(t)$ = AM modulated signal .

A_c = carrier amplitude.

$M(t)$ = message signal .

1.3 – modulation method

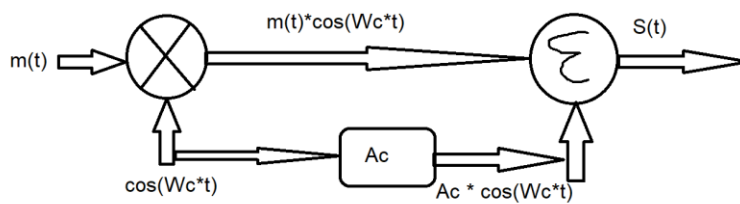


Figure 2 AM modulation method

1.4 – de-modulation method

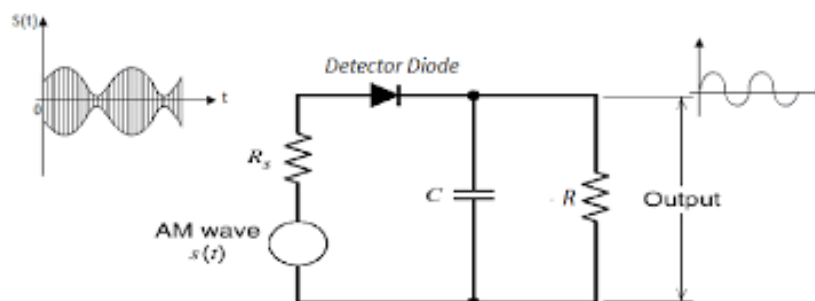


Figure 3 AM de-modulation method

1.5 – MATLAB code

Set the code parameters

```
5      %time setting
6 -    Fs = 800000;           %sample freq
7 -    tmin = 0;              %initial time
8 -    tmax = 0.001;          %max time
9 -    step = 1/Fs;           %sample time
10 -   t = tmin:step:tmax;     %time periode of signal
11      %amplitude setting
12 -   Am = 1;                 %msg amplitude
13 -   Ac = 2;                 %carrier amplitude
14 -   Fm = 2000;
15 -   Fc = 100000;
16
```

Figure 4 AM Set the code parameters

Generate message and carrier

```
18      %generate msg & carrier & modulated signals
19 -   m = Am*cos(2*pi*Fm*t);   %msg
20 -   c = Ac*cos(2*pi*Fc*t);   %carrier
```

Figure 5 AM Generate message and carrier

Plotting the message and carrier

```
28 -   figure(1);
29 -   %msg
30 -   subplot(2,1,1);
31 -   plot(t,m);
32 -   xlabel('time');
33 -   ylabel('amplitude');
34 -   title('msg signal');
35
36
37 -   %carrier
38 -   subplot(2,1,2);
39 -   plot(t,c);
40 -   xlabel('time');
41 -   ylabel('amplitude');
42 -   title('carrier signal');
43
```

Figure 6 AM Plotting the message and carrier

message and carrier

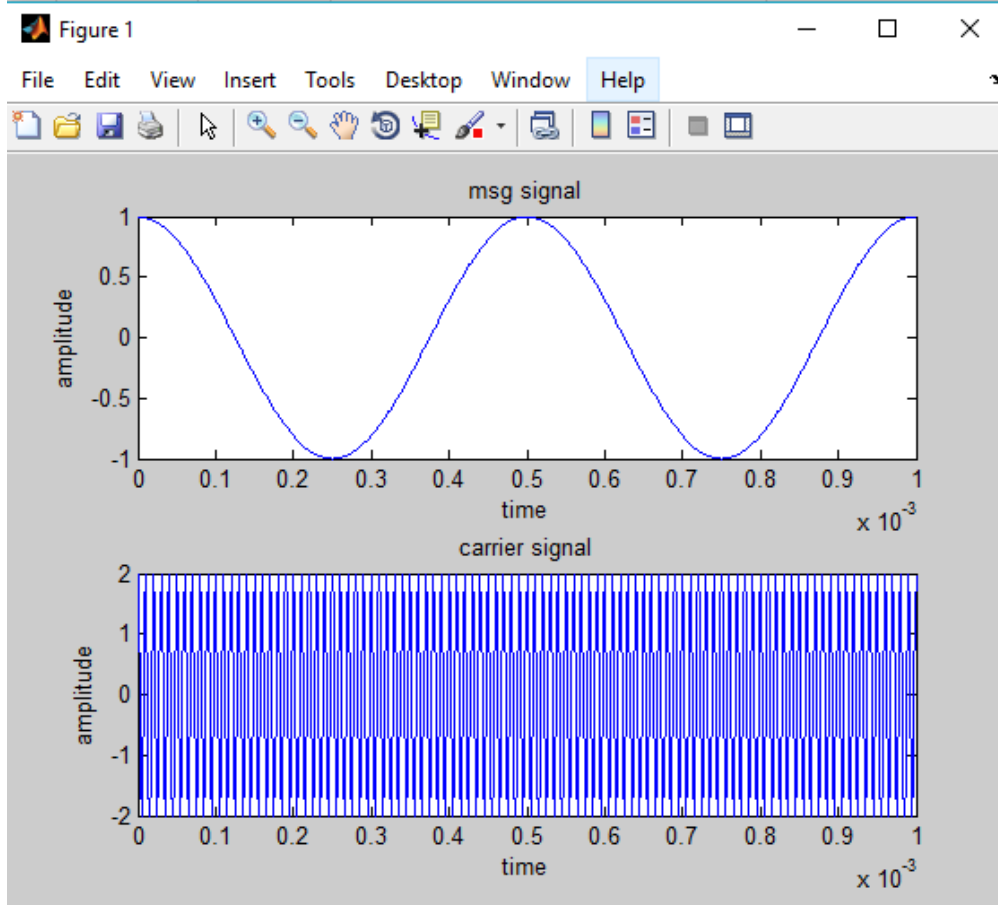


Figure 7 AM message and carrier

Modulation

```
21 - s = (1+m/Ac).*c; %AM modulated signal
```

Figure 8 AM Modulation equation

Demodulation

```
23 %demodulation of AM
24 - y = amdemod(s,Fc,Fs); %built in function for AM-DEmodulation
```

Figure 9 AM Demodulation equation

Plotting modulated and demodulated signals

```
44 - figure(2);
45 - %AM modulated signal
46 - subplot(2,1,1);
47 - plot(t,s); hold on;
48 - plot(t,Ac*(1+m/Ac),'r:'); hold on;
49 - plot(t,-Ac*(1+m/Ac),'r:'); hold on;
50 - xlabel('time');
51 - ylabel('amplitude');
52 - title('AM-modulated signal');
53 -
54 -
55 - %AM demodulated signal
56 - subplot(2,1,2);
57 - plot(t,y);
58 - xlabel('time');
59 - ylabel('amplitude');
60 - title('AM-demodulated signal');
61 -
```

Figure 10 AM Plotting modulated and demodulated signals

modulated and demodulated signals

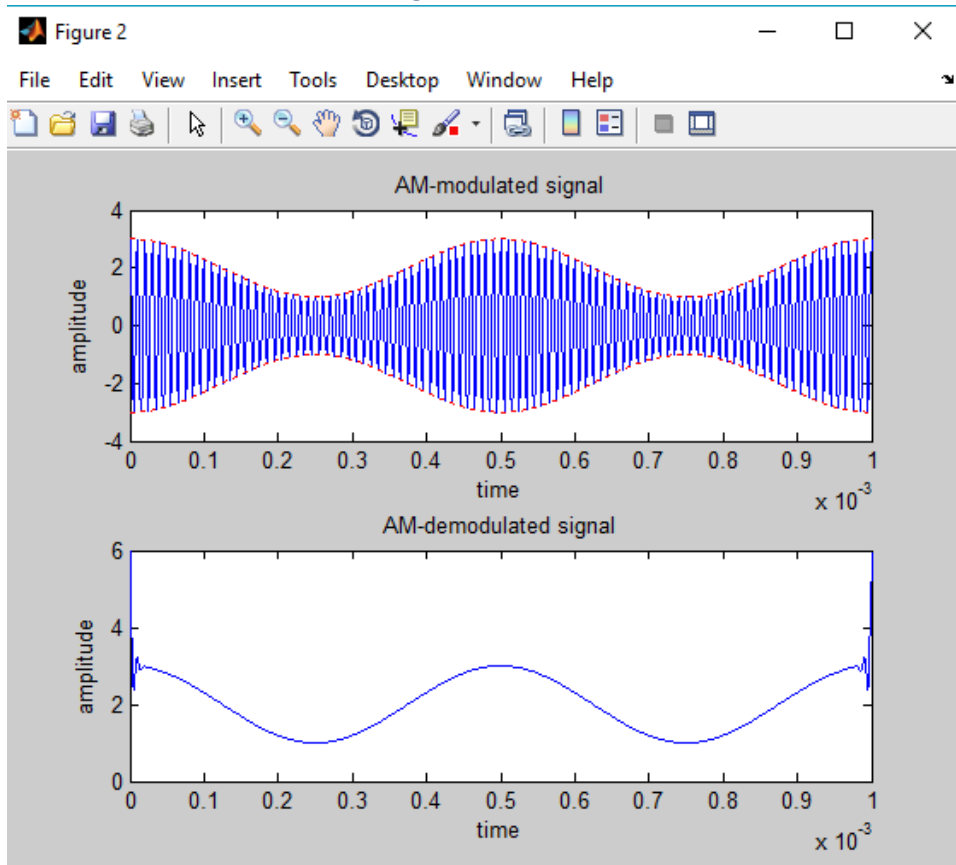


Figure 11 AM modulated and demodulated signals

2 - DSB-SC (Double-sideband suppressed-carrier transmission)

2.1 – introduction

is transmission in which frequencies produced by amplitude modulation (AM) are symmetrically spaced above and below the carrier frequency and the carrier level is reduced to the lowest practical level, ideally being completely suppressed.

In the DSB-SC modulation, unlike in AM, the wave carrier is not transmitted; thus, much of the power is distributed between the side bands, which implies an increase of the cover in DSB-SC, compared to AM, for the same power used.

DSB-SC transmission is a special case of double-sideband reduced carrier transmission. It is used for radio data systems.

2.2 – time domain modulation equation

$$S(t) = m(t) * A_c * \cos(W_c * t)$$

Where

$S(t)$ = DSB-SC modulated signal .

A_c = carrier amplitude.

$M(t)$ = message signal .

2.3 – modulation method

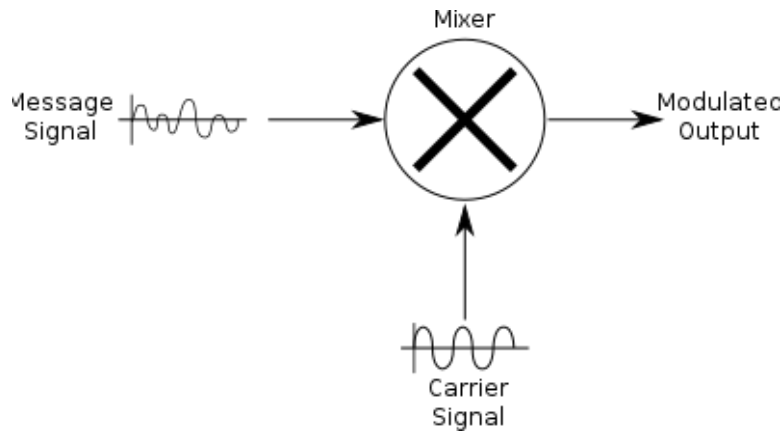


Figure 12 DSB-SC modulation method

2.4 – de-modulation method

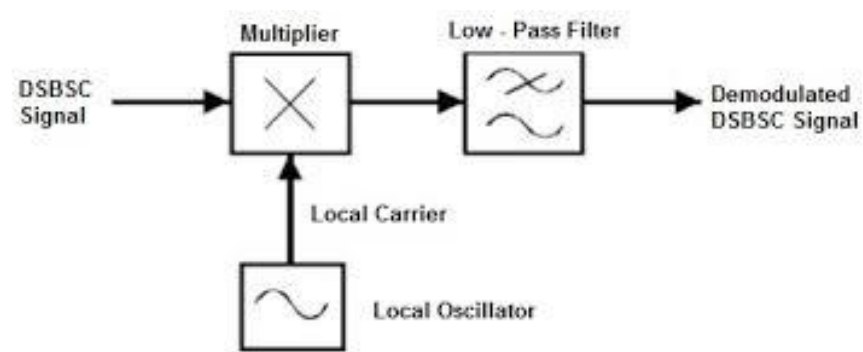


Figure 13 DSB-SC demodulation method

2.5 – MATLAB code

Set the code parameters

```
5 %time setting
6 - Fs = 800000;           %sample freq
7 - tmin = 0;              %initial time
8 - tmax = 0.001;         %max time
9 - step = 1/Fs;          %sample time
10 - t = tmin:step:tmax;   %time periode of signal
11 %amplitude setting
12 - Am = 1;               %msg amplitude
13 - Ac = 2;               %carrier amplitude
14 - Fm = 2000;
15 - Fc = 100000;
16
```

Figure 14 DSB-SC Set the code parameters

Generate message and carrier

```
18 %generate msg & carrier & modulated signals
19 - m = Am*cos(2*pi*Fm*t); %msg
20 - c = Ac*cos(2*pi*Fc*t); %carrier
```

Figure 15 DCB-SC Generate message and carrier

Plotting the message and carrier

```
28 - figure(1);
29 %msg
30 - subplot(2,1,1);
31 - plot(t,m);
32 - xlabel('time');
33 - ylabel('amplitude');
34 - title('msg signal');
35
36
37 %carrier
38 - subplot(2,1,2);
39 - plot(t,c);
40 - xlabel('time');
41 - ylabel('amplitude');
42 - title('carrier signal');
43
```

Figure 16 DSB-SC Plotting the message and carrier

message and carrier

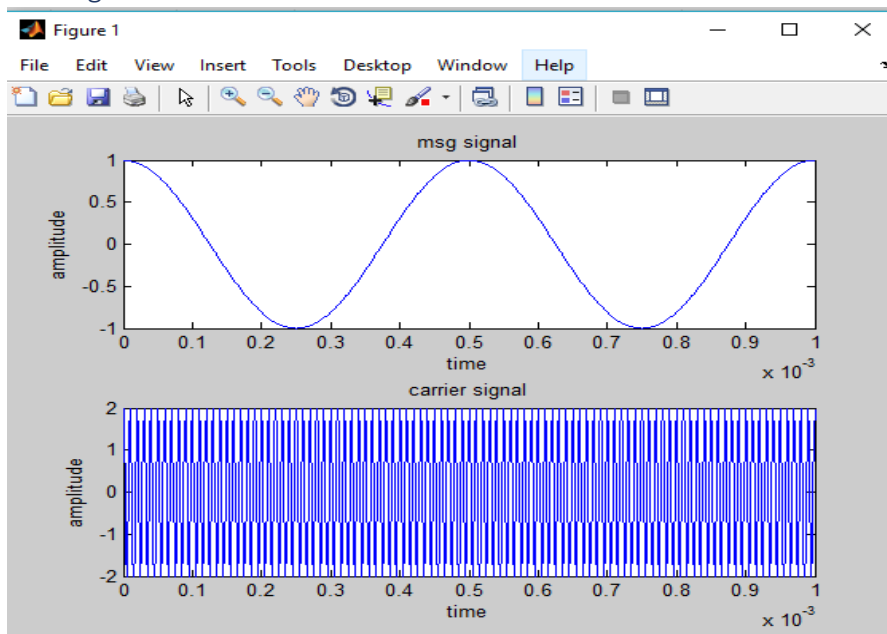


Figure 17 DCB-SC message and carrier

Modulation

```
21 -      s = m.*c;                                %DSB-SC modulated signal
```

Figure 18 DSB-SC Modulation equation

Demodulation

```
23      %demodulation of DSB-SC
24 -      x = s.*c;
25 -      Wn = Fc/(Fs/2);                          % Normalized cutoff frequency
26 -      [b a] = butter(5,Wn,'low'); %5th order low-pass filter
27 -      y = filter(b,a,x);
--      -
```

Figure 19 DSB-SC Demodulation equation

Plotting modulated and demodulated signals

```
47 -      figure(2);
48      %DSB-SC modulated signal
49 -      subplot(2,1,1);
50 -      plot(t,s); hold on;
51 -      plot (t,Ac*m,'r:'); hold on;
52 -      plot (t,-Ac*m,'r:'); hold on;
53 -      xlabel ('time');
54 -      ylabel ('amplitude');
55 -      title('DSB-SC modulated signal');
56
57
58      %DSB-SC demodulated signal
59 -      subplot(2,1,2);
60 -      plot(t,y);
61 -      xlabel ('time');
62 -      ylabel ('amplitude');
63 -      title('DSB-SC demodulated signal');
64
```

Figure 20 DSB-SC Plotting modulated and demodulated signals

modulated and demodulated signals

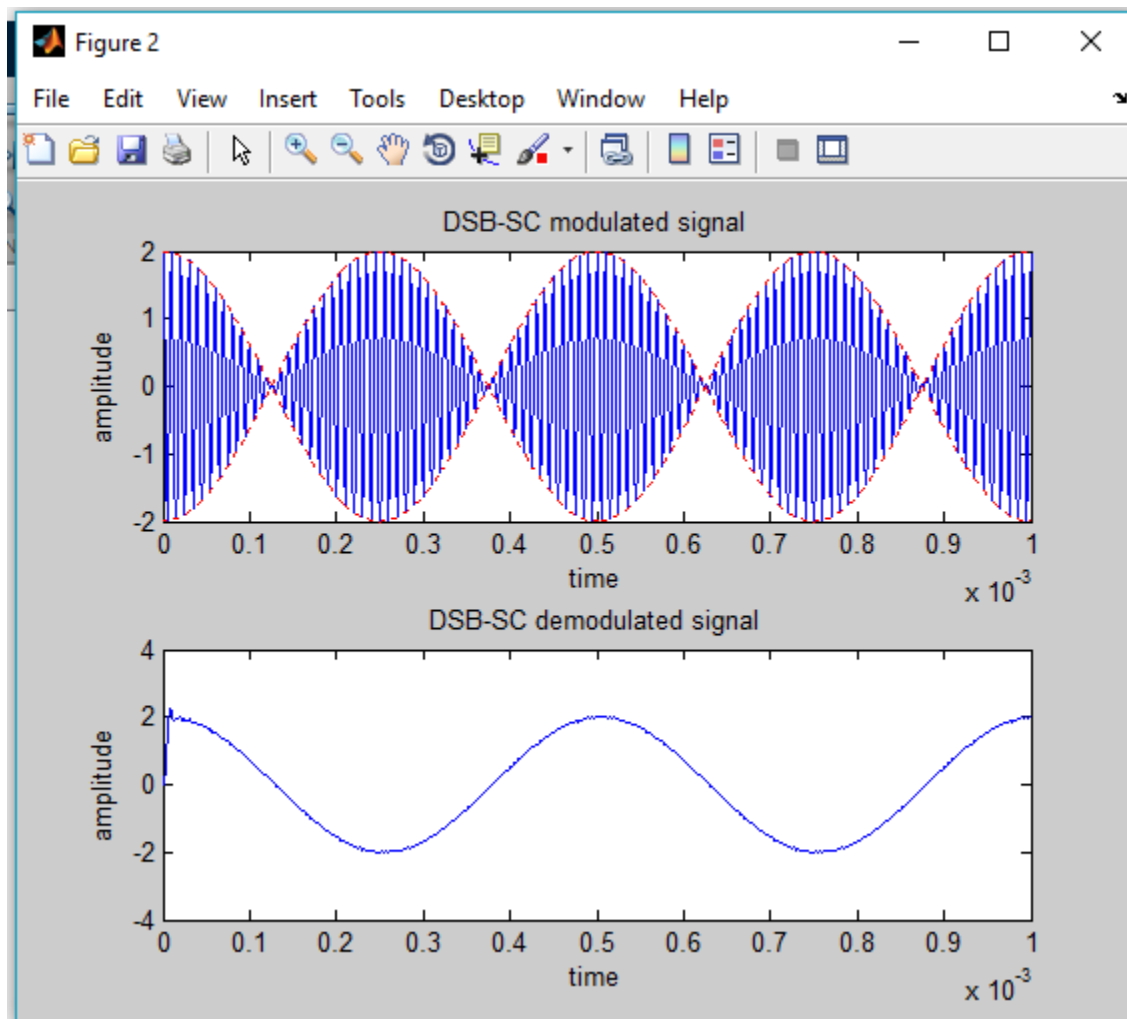


Figure 21 DSB-SC modulated and demodulated signals

3 – SSB (Single-sideband modulation)

3.1 – introduction

is a type of modulation, used to transmit information, such as an audio signal, by radio waves. A refinement of amplitude modulation, it uses transmitter power and bandwidth more efficiently. Amplitude modulation produces an output signal that has twice the bandwidth of the original baseband signal. Single-sideband modulation avoids this bandwidth doubling, and the power wasted on a carrier, at the cost of increased device complexity and more difficult tuning at the receiver.

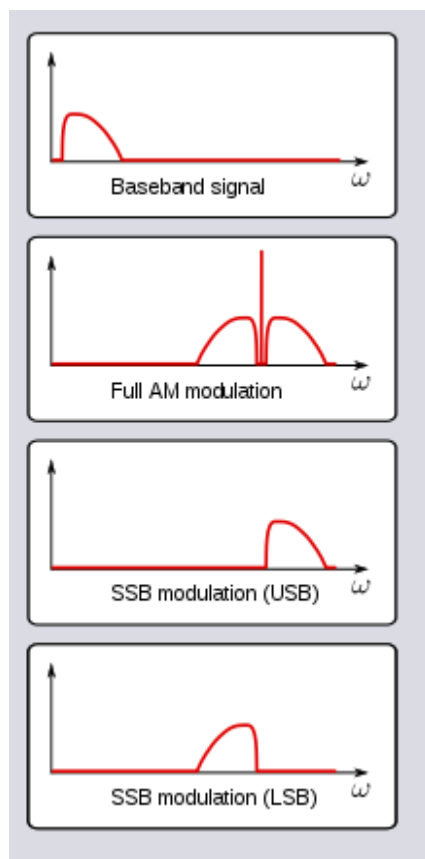


Figure 22 SSB signal

3.2 – time domain modulation equation

$$S(t) = m(t) \cdot \cos(W_c \cdot t) - M_h(t) \cdot \sin(W_c \cdot t) \quad \text{for USB (upper side band)}$$

$$S(t) = m(t) \cdot \cos(W_c \cdot t) + M_h(t) \cdot \sin(W_c \cdot t) \quad \text{for LSB (lower side band)}$$

Where

$S(t)$ = DSB-SC modulated signal .

$M(t)$ = message signal .

$M_h(t)$ = hilbert transform of $m(t)$.

3.3 – modulation method

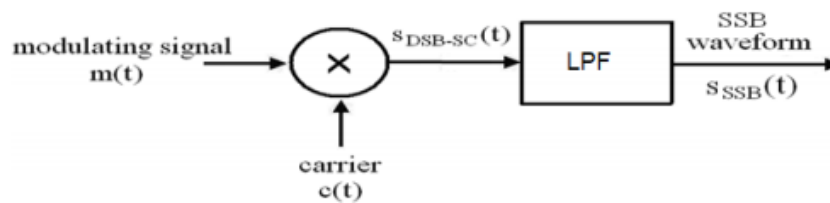


Figure 23 SSB modulation method 1

Phase-Shift Method

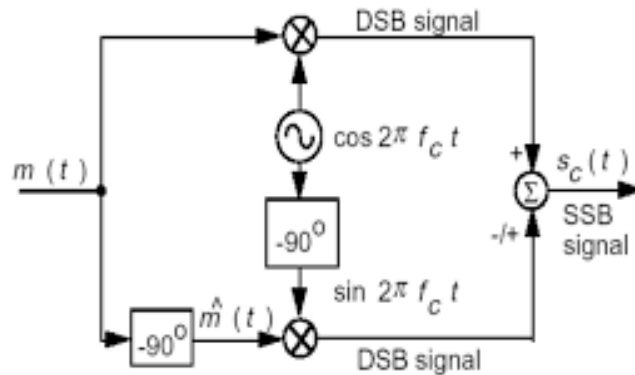


Figure 24 SSB modulation method 2

3.4 – de-modulation method

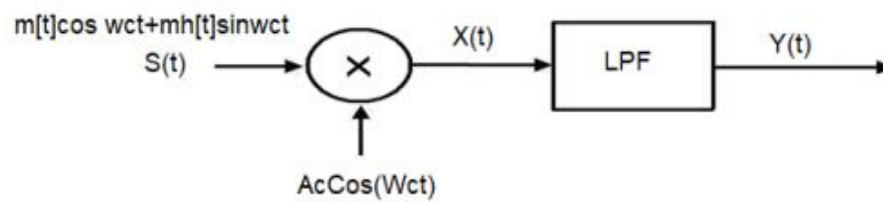


Figure 25 SSB de-modulation method

3.5 – MATLAB code

Set the code parameters

```
5 %time setting
6 - Fs = 800000; %sample freq
7 - tmin = 0; %initial time
8 - tmax = 0.001; %max time
9 - step = 1/Fs; %sample time
10 - t = tmin:step:tmax; %time periode of signal
11 %amplitude setting
12 - Am = 1; %msg amplitude
13 - Ac = 2; %carrier amplitude
14 - Fm = 2000;
15 - Fc = 100000;
16
```

Figure 26 SSB Set the code parameters

Generate message and carrier

```
19 - m = Am*cos(2*pi*Fm*t); %msg
20 - mh = Am*cos(2*pi*Fm*t+90); %helbert transform of msg
21 - c1 = Ac*cos(2*pi*Fc*t); %carrier (cos)
22 - c2 = Ac*sin(2*pi*Fc*t); %carrier (sin)
```

Figure 27 SSB Generate message and carrier

Plotting the message and carrier

```

35
36 - figure(1);
37 - %msg
38 - subplot(2,1,1);
39 - plot(t,m);
40 - xlabel ('time');
41 - ylabel ('amplitude');
42 - title('msg signal');
43
44 - %helbert transform of msg
45 - subplot(2,1,2);
46 - plot(t,mh);
47 - xlabel ('time');
48 - ylabel ('amplitude');
49 - title('helbert transform of msg');
50
51 - figure(2);
52 - %carrier
53 - subplot(2,1,1);
54 - plot(t,c1);
55 - xlabel ('time');
56 - ylabel ('amplitude');
57 - title('carrier signal');
58
59 - %carrier
60 - subplot(2,1,2);
61 - plot(t,c2);
62 - xlabel ('time');
63 - ylabel ('amplitude');
64 - title('carrier of helbert signal');
65

```

Figure 28 SSB Plotting the message and carrier

message and carrier

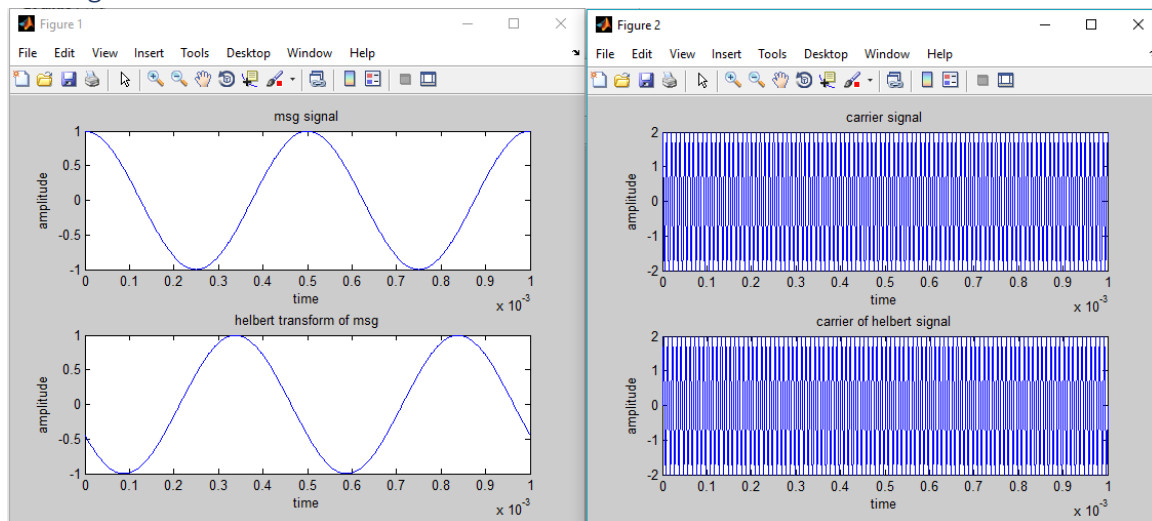


Figure 29 SSB message and carrier

Modulation

```
23 %SSB(LSB) modulated signal
24 - s1 = m.*c1;
25 - s2 = mh.*c2;
26 - s = s1+s2;
```

Figure 30 SSB(LSB) Modulation equation

Demodulation

```
28 %demodulation of SSB(LSB)
29 - x = c1.*s;
30 - Wn = Fc/(Fs/2); % Normalized cutoff frequency
31 - [b a] = butter(5,Wn,'low'); %5th order low-pass filter
32 - y = filter(b,a,x);
```

Figure 31 SSB(LSB) Demodulation equation

Plotting modulated and demodulated signals

```
66 - figure(3);
67 %SSB(LSB) modulated signal
68 - subplot(2,1,1);
69 - plot(t,s);
70 - xlabel('time');
71 - ylabel('amplitude');
72 - title('SSB(LSB)-modulated signal');
73
74
75 %SSB(LSB) demodulated signal
76 - subplot(2,1,2);
77 - plot(t,y);
78 - xlabel('time');
79 - ylabel('amplitude');
80 - title('SSB(LSB)-demodulated signal');
```

Figure 32 SSB(LSB) Plotting modulated and demodulated signals

modulated and demodulated signals

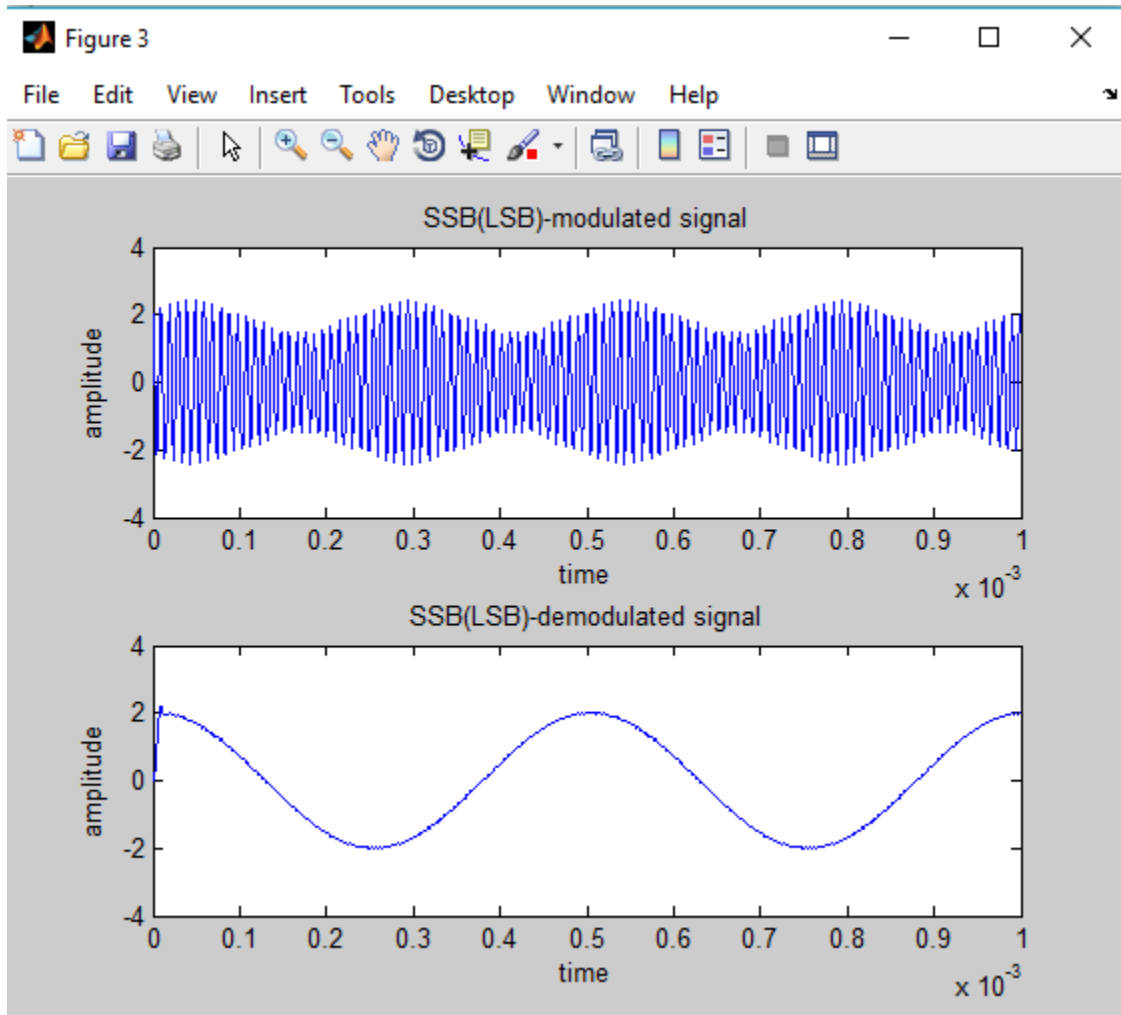


Figure 33 SSB(LSB) modulated and demodulated signals

4 – MATLAB used functions

4.1 amdemod

```
>> help amdemod
amdemod Amplitude demodulation.

    Z = amdemod(Y,Fc,Fs) demodulates the amplitude modulated signal Y from
    the carrier frequency Fc (Hz). Y and Fc have sample frequency Fs (Hz).
    The modulated signal Y has zero initial phase, and zero carrier
    amplitude, for suppressed carrier modulation. A lowpass filter is used
    in the demodulation. The default filter is: [NUM,DEN] =
    butter(5,Fc*2/Fs).

    Z = amdemod(Y,Fc,Fs,INI_PHASE) specifies the initial phase (rad) of the
    modulated signal.

    Z = amdemod(Y,Fc,Fs,INI_PHASE,CARRAMP) specifies the carrier amplitude
    of the modulated signal for transmitted carrier modulation.

    Z = amdemod(Y,Fc,Fs,INI_PHASE,CARRAMP,NUM,DEN) specifies the filter to
    be used in the demodulation.
```

Figure 34 amdemod function

4.2 butter

```
>> help butter
butter Butterworth digital and analog filter design.

    [B,A] = butter(N,Wn) designs an Nth order lowpass digital
    Butterworth filter and returns the filter coefficients in length
    N+1 vectors B (numerator) and A (denominator). The coefficients
    are listed in descending powers of z. The cutoff frequency
    Wn must be 0.0 < Wn < 1.0, with 1.0 corresponding to
    half the sample rate.

    If Wn is a two-element vector, Wn = [W1 W2], butter returns an
    order 2N bandpass filter with passband W1 < W < W2.
    [B,A] = butter(N,Wn,'high') designs a highpass filter.
    [B,A] = butter(N,Wn,'low') designs a lowpass filter.
    [B,A] = butter(N,Wn,'stop') is a bandstop filter if Wn = [W1 W2].

    When used with three left-hand arguments, as in
    [Z,P,K] = butter(...), the zeros and poles are returned in
    length N column vectors Z and P, and the gain in scalar K.

    When used with four left-hand arguments, as in
    [A,B,C,D] = butter(...), state-space matrices are returned.

    butter(N,Wn,'s'), butter(N,Wn,'high','s') and butter(N,Wn,'stop','s')
    design analog Butterworth filters. In this case, Wn is in [rad/s]
    and it can be greater than 1.0.
```

Figure 35 butter function