Appendix C: Supplementary Problems on Modulation Systems

C.1. Examples on Amplitude Modulation Systems

Example C.1.1: MATLAB Demonstration of Double Sideband Suppressed Carrier (Ordinary) AM. Let us assume that the message has the following three values in specified intervals and the carrier is a sinusoid:

$$x(t) = \begin{cases} 1 & if \;\; 0 \le t \le t_0/3 \\ -2 & if \;\; t_0/3 \le t \le 2t_0/3 \\ 0 & Otherwise \end{cases} \qquad \text{and} \qquad c(t) = Cos(2p_c ft)$$

with $f_c = 250\ Hz$, $t_0 = 0.15\ Seconds$, and the modulation index: $mhu = m = 0.85$. Furthermore, suppose that the signal has been corrupted by a Gaussian noise with an SNR of 10.0 dB.

Let us note that that $|\max(x(t)| = 2.0$ and re-write the signal in terms of rectangular windows in the time-domain, we have:

$$x(t) = [1 + 0.85 * \frac{x(t)}{2}]Cos(2\pi_c ft)$$
$$= [1 + 0.425 * rect(\frac{t - 0.025}{0.05}) - 0.85 * rect(\frac{t - 0.075}{0.05})].Cos(500\pi t)$$

Term-by-term FT and re-organization of various terms results in:

$$X(w) = 0.05e^{-j0.05\pi f} Sinc(0.05 f)(1 - 2e^{-j0.1\pi f})$$

The corresponding modulated signal (convolution in frequency-domain) turns out to be:

$$U(f) = 0.010625e^{-j0.05\pi(f-250)} Sinc(0.05(f - 250))(1 - 2e^{-j0.1\pi(f-250)})$$
$$= 0.010625e^{-j0.05\pi(f+250)} Sinc(0.05(f + 250))(1 - 2e^{-j0.1\pi(f+250)})$$

Power budget of the message and the normalized message:

$$P_x = \frac{1}{0.15}[\int_0^{0.05} dt + 4 \int_{0.05}^{0.1} dt] = 1.66$$

$$P_x^{normalized} = \frac{1}{4}.P_x = \frac{1.66}{4} = 0.4167$$

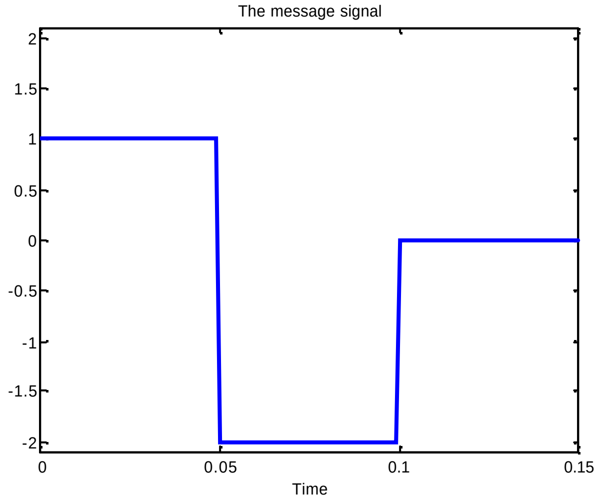
The modulation efficiency for the specific values given and computer is:

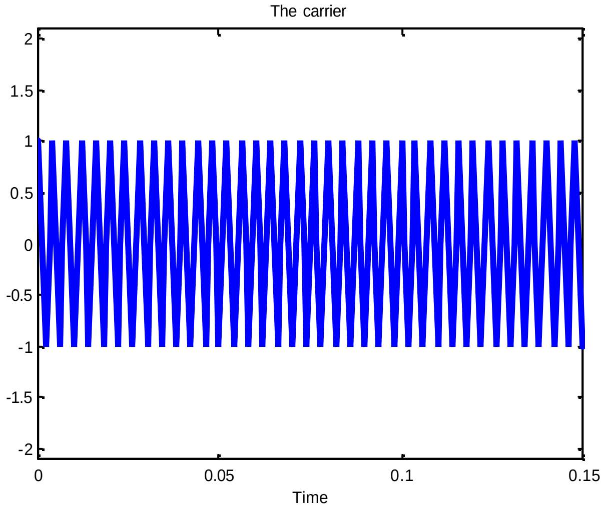
$$eta = h = \frac{m^2.P_x^{normalized}}{1 + m^2.P_x^{normalized}} = \frac{0.85^2 * 0.4167}{1 + 0.85^2 * 0.4167} = 0.2314 = 23.14\%$$

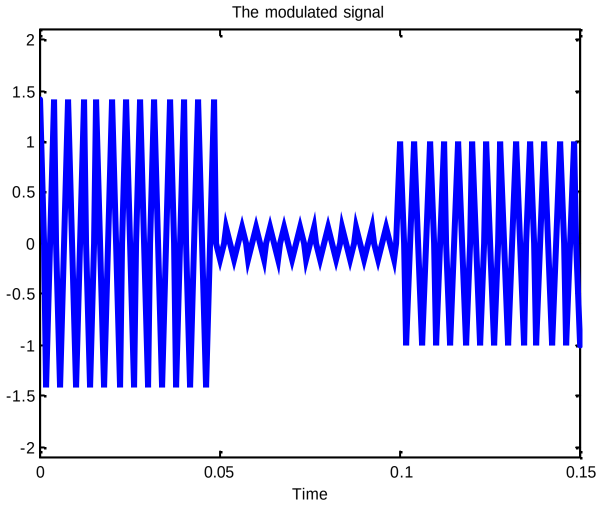
Finally, the power in the modulated signal is computed as:

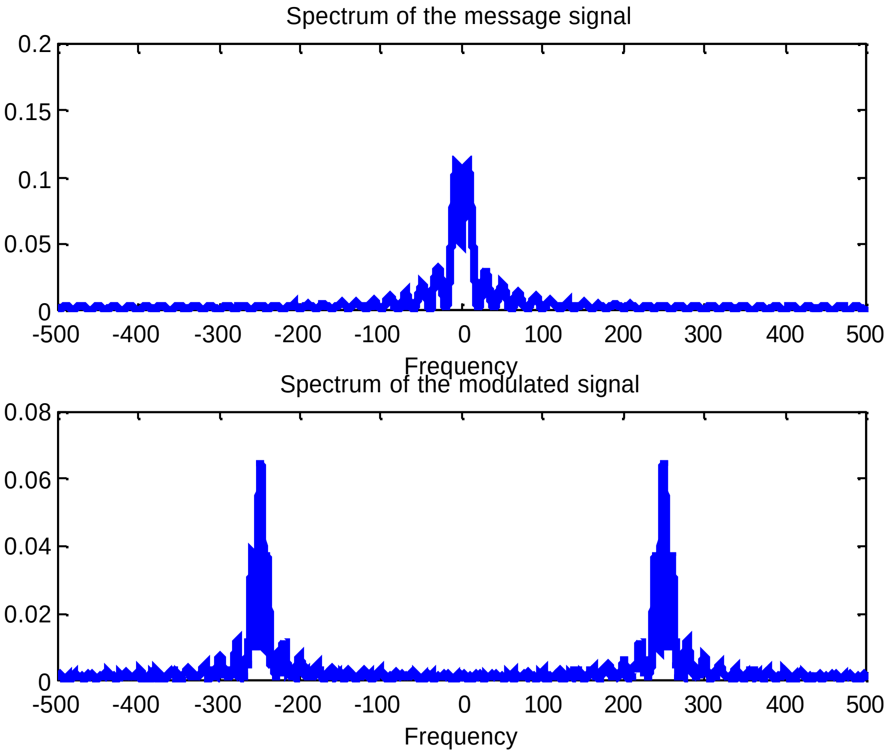
$$P_u = \frac{A_c^2}{2} AverageValue\{[1 + h * x_{normalized}(t)]^2\}$$
$$= \frac{1}{2}(1 + 0.3010 - 1.7 * \frac{0.025}{0.15}) = 0.5088$$

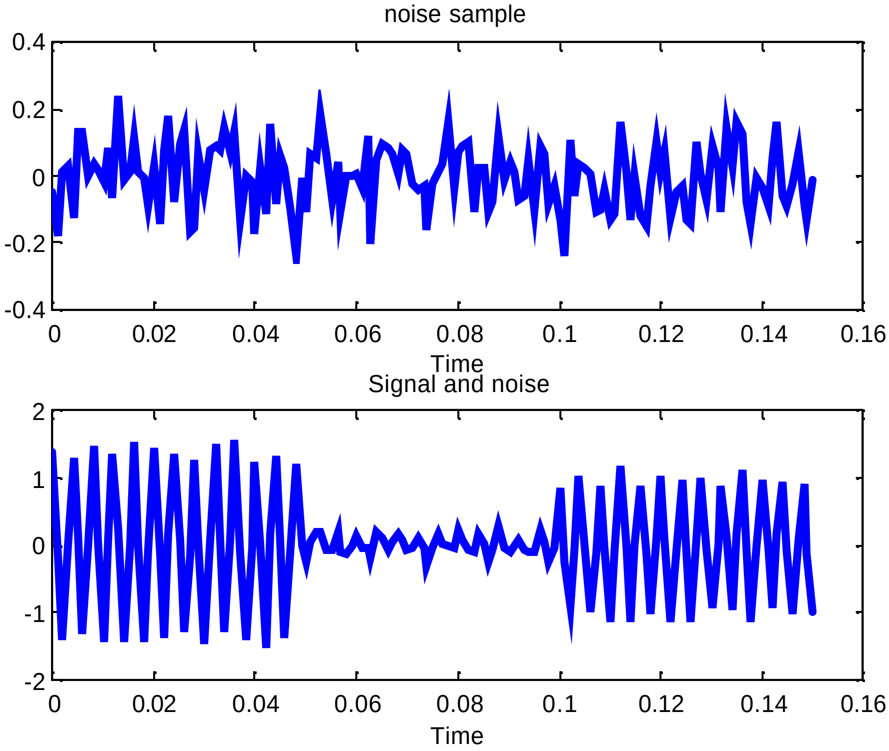
```
%  Example on AM Modulation Process
% am_mod.m
% Message Signal: +1 for 0 < t < t0/3; -2 for t0/3 < t < 2t0/3 and zero: otherwise.
t0=.15; ts=0.001;                % signal duration and sampling interval
fc=250; fs=1/ts;                 % carrier frequency and sampling frequency
snr=10;                          % SNR in dB (logarithmic)
a=0.85;                          % Modulation index
t=[0:ts:t0];                     % time vector
df=0.2;                          % required frequency resolution
snr_lin=10^(snr/10);             % SNR

m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];      % message signal
c=cos(2*pi*fc.*t);                         % carrier signal
m_n=m/max(abs(m));                         % normalized message signal
[M,m,df1]=fftseq(m,ts,df);                 % Fourier transform
M=M/fs;                                    % scaling
f=[0:df1:df1*(length(m)-1)]-fs/2;          % frequency vector
u=(1+a*m_n).*c;                            % modulated signal
[U,u,df1]=fftseq(u,ts,df);                 % Fourier transform
U=U/fs;                                    % scaling
signal_power=spower(u(1:length(t)));       % power in modulated signal
pmn=spower(m(1:length(t)))/(max(abs(m)))^2;          % power in normalized message
eta=(a^2*pmn)/(1+a^2*pmn);                           % modulation efficiency
noise_power=eta*signal_power/snr_lin;                % noise power
noise_std=sqrt(noise_power);               % noise standard deviation
noise=noise_std*randn(1,length(u));        % generate noise
r=u+noise;                                 % noise plus modulated signal
[R,r,df1]=fftseq(r,ts,df);                 % Fourier transform
R=R/fs;                                    % scaling

signal_power                               % Modulated signal power value
eta                                        % Modulation efficiency value

%Plots
plot(t,m(1:length(t))); axis([0 0.15 -2.1 2.1]); xlabel('Time'), title('The message signal');
figure; plot(t,c(1:length(t)));axis([0 0.15 -2.1 2.1]); xlabel('Time'); title('The carrier')
figure; plot(t,u(1:length(t))); axis([0 0.15 -2.1 2.1]); xlabel('Time'), title('The modulated signal')
figure; subplot(2, 1,1), plot(f,abs(fftshift(M))) ; xlabel('Frequency'),title('Spectrum of the message signal')
subplot(2,1,2), plot(f,abs(fftshift(U))) ; title('Spectrum of the modulated signal'), xlabel('Frequency')
figure; subplot(2,1, 1), plot(t,noise(1:length(t))) ; title('noise sample'), xlabel('Time')
subplot(2,1,2), plot(t,r(1:length(t))) ; title('Signal and noise'), xlabel('Time')
figure; subplot(2, 1,1), plot(f,abs(fftshift(U))) ; title('Signal spectrum'), xlabel('Frequency')
subplot(2,1,2), plot(f,abs(fftshift(R))) ; title('Signal and noise spectrum'), xlabel('Frequency')
end;

function p=spower(x)

%SPOWER      Returns the power in signal x
p=(norm(x)^2)/length(x);
```

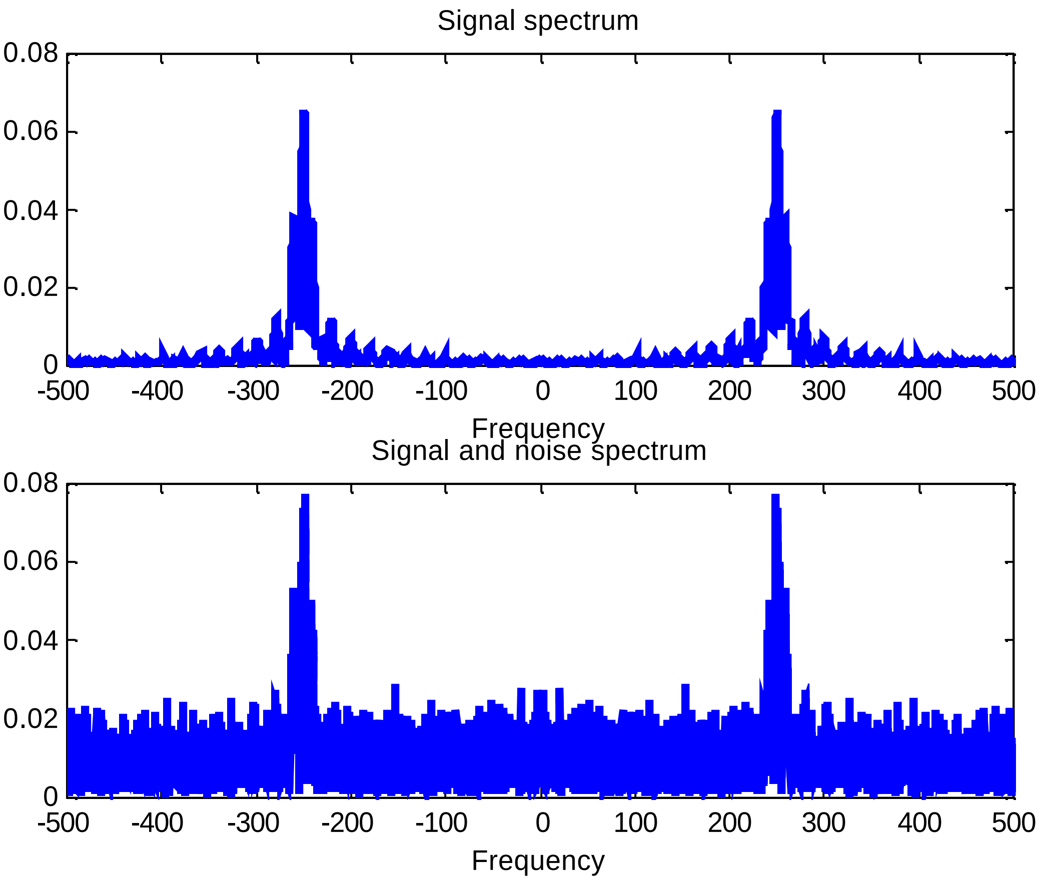

The message signal
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.05
0.1
0.15
Time


The carrier
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.05
0.1
0.15
Time


The modulated signal
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.05
0.1
0.15
Time


Spectrum of the message signal
0.2
0.15
0.1
0.05
0
-500
-400
-300
-200
-100
0
100
200
300
400
500
Frequency
Spectrum of the modulated signal
0.08
0.06
0.04
0.02
0
-500
-400
-300
-200
-100
0
100
200
300
400
500
Frequency


noise sample
0.4
0.2
0
-0.2
-0.4
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
Time
Signal and noise
2
1
0
-1
-2
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
Time



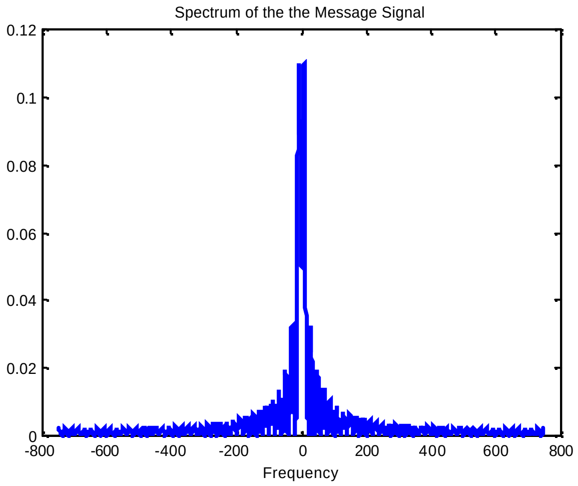
Example C.1.2: MATLAB demonstration of AM demodulation. Let us recall the Matlab implementation of AM case in Example 4.A.1 and use the multiplier (mixer) demodulator discussed in the text followed by a low-pass filter to recover the original signal. Again the message signal and the carrier are as before:

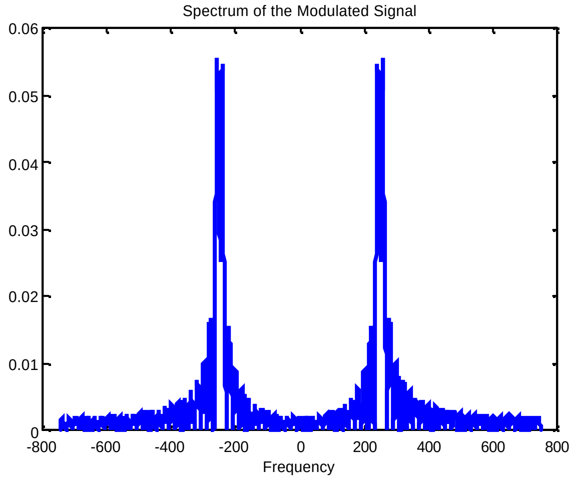
$$x(t)=\begin{cases} 1 & if\ \ 0\le t\le t_0/3 \\ -2 & if\ \ t_0/3\le t\le 2t_0/3 \\ 0 & Otherwise \end{cases} \qquad \text{and} \qquad c(t)=Cos(2p_{c}ft)$$

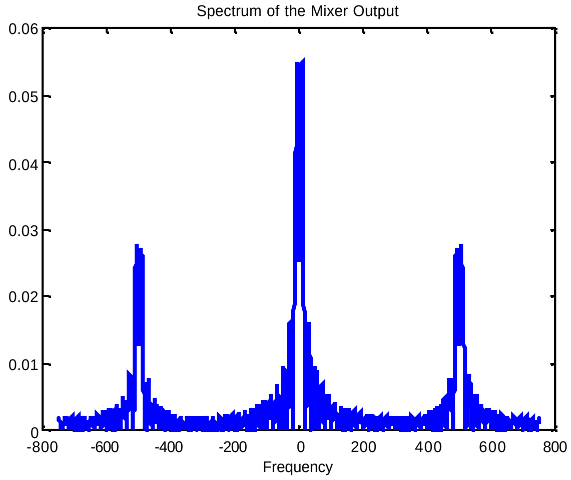
with $f_c = 250\ Hz$, $t_0 = 0.15\ Seconds$, and the modulation index: $mhu = m = 0.85$. This signal is corrupted by a Gaussian noise with an SNR of 10.0 dB. Note that $|\max(x(t)|= 2.0$.

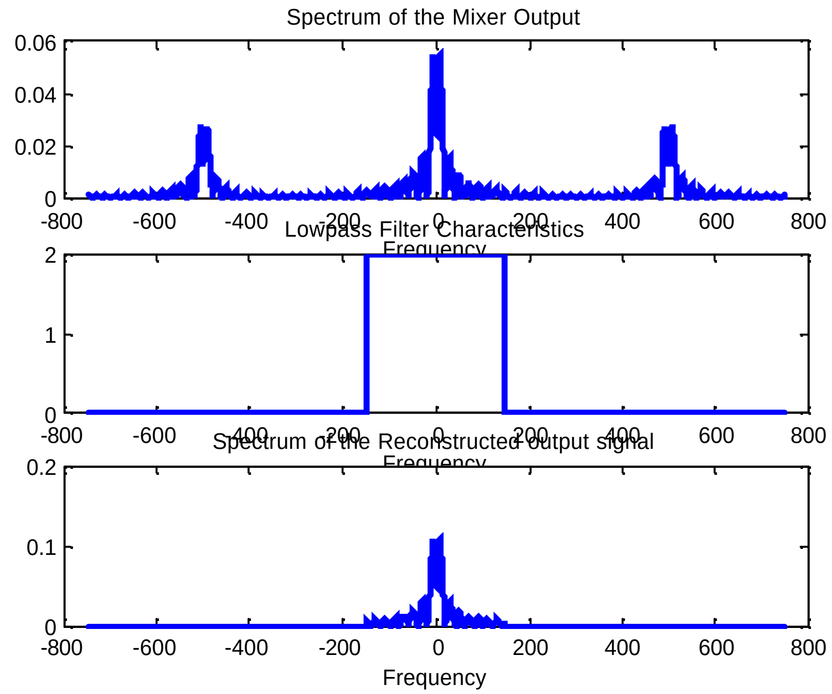
As we clearly see from the plots both in the frequency-domain and the time-domain, the envelope detector can closely track and reconstruct the original signal.

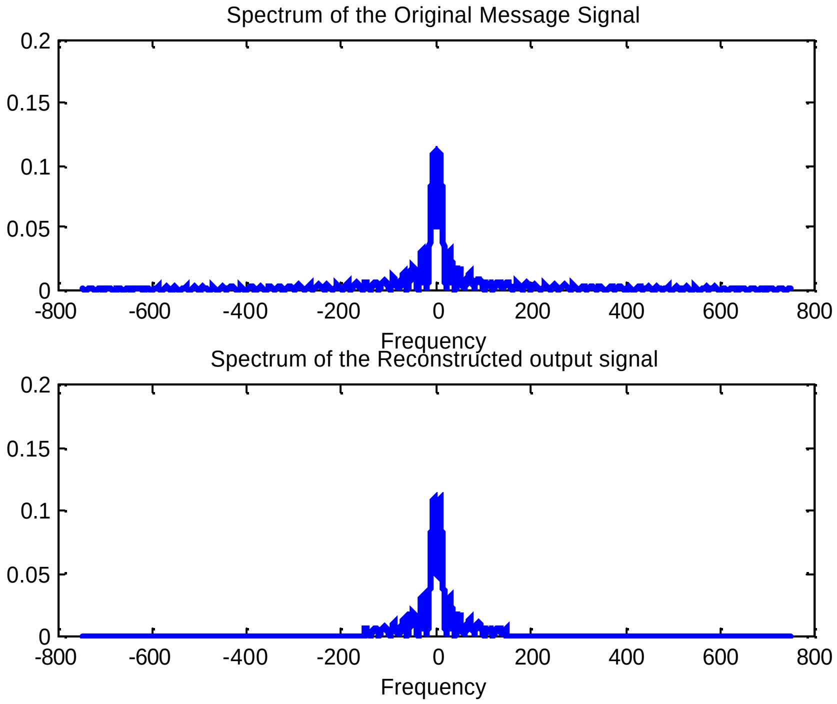
```
% Example on DSB-AM demodulation.

% System Parameters:
t0=.15; ts=1/1500;                      % signal duration & sampling interval
fc=250; fs=1/ts;                        % carrier and sampling frequency
t=[0:ts:t0];                            % time vector
df=0.3;                                 % desired frequency resolution
m=[ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];     % Signal Generation
c=cos(2*pi*fc.*t);                                                   % carrier signal
u=m.*c;                                                              % the modulated signal
y=u.*c;                                                              % mixing at the receiver
[M,m,df1]=fftseq(m,ts,df);  [U,u,df1]=fftseq(u,ts,df);           % Fourier transforms
[Y,y,df1]=fftseq(y,ts,df);                                           % Fourier transform
M=M/fs; U=U/fs;   Y=Y/fs;                                             % scaling

% Output Low-Pass Filter and filtering the output signal
f_cutoff=150;                           % cutoff freq. of the filter
n_cutoff=floor(150/df1);                % design the low-pass filter for recovery.
f=[0:df1:df1*(length(y)-1)]-fs/2;
H=zeros(size(f)); H(1:n_cutoff)=2*ones(1,n_cutoff);
H(length(f)-n_cutoff+1:length(f))=2*ones(1,n_cutoff);
DEM=H.*Y;                               % spectrum of the filter output
dem=real(ifft(DEM))*fs;                 % filter output

%Plotting Routines
clf; figure; plot(f,fftshift(abs(M))); title('Spectrum of the the Message Signal'); xlabel('Frequency')
figure; plot(f,fftshift(abs(U))); title('Spectrum of the Modulated Signal'); xlabel('Frequency')
figure; plot(f,fftshift(abs(Y))); title('Spectrum of the Mixer Output'); xlabel('Frequency')
figure; subplot(3,1,1), plot(f,fftshift(abs(Y))) ; title('Spectrum of the Mixer Output'); xlabel('Frequency')
subplot(3,1,2), plot(f,fftshift(abs(H))); title('Lowpass Filter Characteristics'); xlabel('Frequency')
subplot(3,1,3), plot(f,fftshift(abs(DEM))); title('Spectrum of the Reconstructed output signal');
xlabel('Frequency')
figure; subplot(2,1,1), plot(f,fftshift(abs(M))); title('Spectrum of the Original Message Signal');
xlabel('Frequency')
subplot(2,1,2), plot(f,fftshift(abs(DEM)))
title('Spectrum of the Reconstructed output signal'); xlabel('Frequency')
figure; subplot(2,1,1), plot(t,m(1:length(t))); title('The Original Message Signal'); xlabel('Time')
subplot(2,1,2), plot(t,dem(1:length(t))); title('The Reconstructed output signal'); xlabel('Time');

function [M,m,df]=fftseq(m,ts,df)

%FFTSEQ     Generates M, the FFT of the sequence m.
%       The sequence is zero padded to meet the required frequency resolution df.
%       ts is the sampling interval. The output df is the final frequency resolution.
%       Output m is the zero padded version of input m. M is the FFT.

fs=1/ts;
if nargin == 2
    n1=0;
else
    n1=fs/df;
end

n2=length(m); n=2^(max(nextpow2(n1),nextpow2(n2)));
M=fft(m,n); m=[m,zeros(1,n-n2)]; df=fs/n;
```

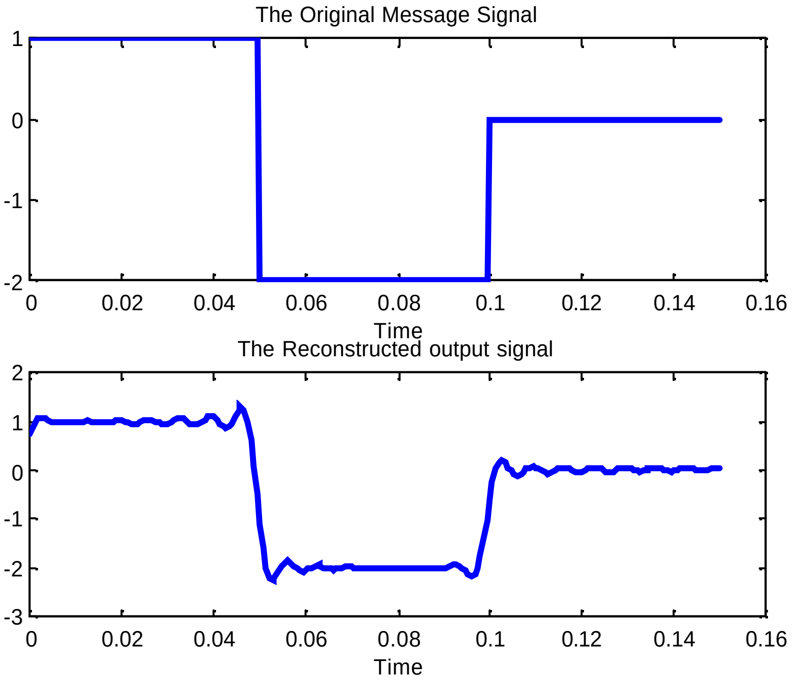

Spectrum of the the Message Signal
0.12
0.1
0.08
0.06
0.04
0.02
0
-800
-600
-400
-200
0
200
400
600
800
Frequency


Spectrum of the Modulated Signal
0.06
0.05
0.04
0.03
0.02
0.01
0
-800
-600
-400
-200
0
200
400
600
800
Frequency


Spectrum of the Mixer Output
0.06
0.05
0.04
0.03
0.02
0.01
0
-800
-600
-400
-200
0
200
400
600
800
Frequency


Spectrum of the Mixer Output
0.06
0.04
0.02
0
-800
-600
-400
-200
0
200
400
600
800
Frequency
Lowpass Filter Characteristics
2
1
0
-800
-600
-400
-200
0
200
400
600
800
Frequency
Spectrum of the Reconstructed output signal
0.2
0.1
0
-800
-600
-400
-200
0
200
400
600
800
Frequency


Spectrum of the Original Message Signal
0.2
0.15
0.1
0.05
0
-800
-600
-400
-200
0
200
400
600
800
Frequency
Spectrum of the Reconstructed output signal
0.2
0.15
0.1
0.05
0
-800
-600
-400
-200
0
200
400
600
800
Frequency



Example C.1.3: MATLAB demonstration on Amplitude Shift keying (ASK), Phase Shift Keying (PSK) and Frequency Shift Keying (FSK) to demonstrate modulating different parts of a sinusoidal carrier signal.

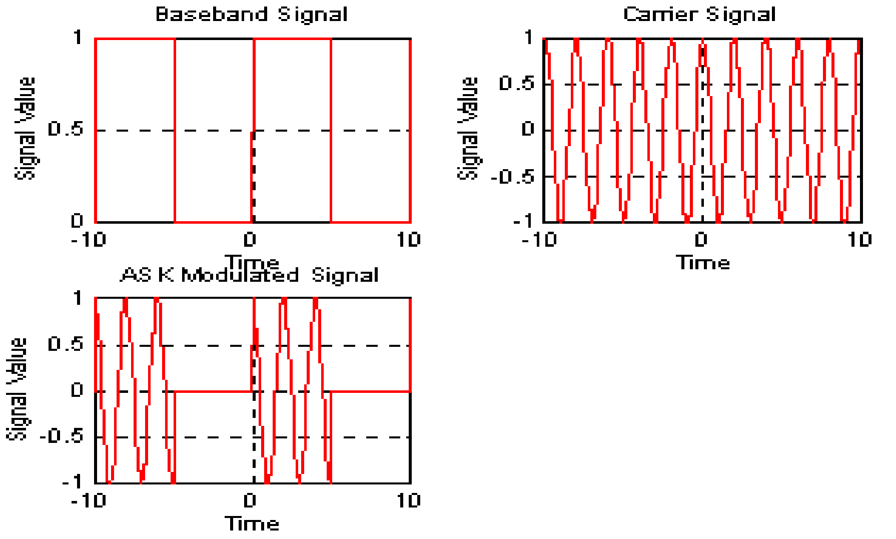
It is worth noting that these techniques together with many of their derivatives are normally discussed together with the digital modulation concepts.

Plots below are generated using the Matlab Communications Toolbox.

- Here the baseband signal used in each case is an 0.1Hz Non-Return-to-Zero pulse signal which alternates between a logic “0” and a logic “1”.
- The carrier signal is a 0.5Hz Cosine waveform.

Case 1: ASK: In Amplitude Shift Keying (ASK), the signal alternates between a “0” and a cosine signal:

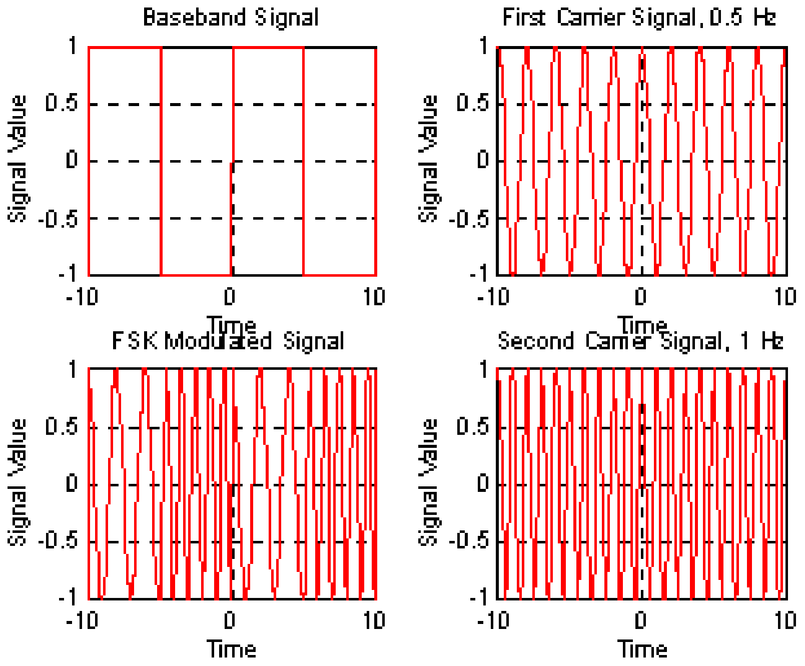
$$s(t) = \begin{cases} A\cos(2\pi fcTb) \rightarrow Symbol \quad 1 \\ 0 \rightarrow Symbol \quad 0 \end{cases}$$



Case 2: FSK: In Frequency Shift Keying (FSK), two carrier signals are used instead of one. In our example, we chose the second carrier frequency to be twice the first carrier frequency. The second carrier frequency will occur when a symbol 0 occurs. The definition of FSK is given below, where n is a positive integer :

$$s(t) = \begin{cases} A\cos(2\pi fcTb) \rightarrow Symbol \quad 1 \\ A\cos(2n\pi fcTb) \rightarrow Symbol \quad 0 \end{cases}$$

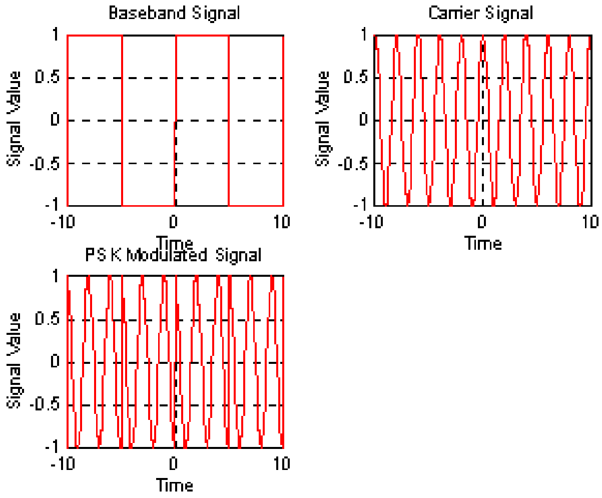
From the equation, we can see that in order to detect the logic level, we would need to be able to detect the two different frequencies.



Case 3: PSK: In Phase Shift Keying (PSK), the signals for logic "0" and logic "1" are 180 degrees out of phase with one another. By detecting the phase shift, we can determine if the transmitted signal is a logic 0 or logic 1. The definition of FSK is :

$$s(t) = \begin{cases} A\cos(2\pi fcTb) \rightarrow Symbol \quad 1 \\ A\cos(2\pi fcTb + \pi) \rightarrow Symbol \quad 0 \end{cases}$$

As can be seen from the definition, the signal is shifted by 180 degrees when a symbol 0 is transmitted. PSK has extensively used in both military and commercial communications.



C.2. Examples on Angle Modulation Systems

Example C.2.1: MATLAB Demonstration of Double Sideband Suppressed Carrier (Ordinary) AM. Let us assume that the message has the following three values in specified intervals and the carrier is a sinusoid:

$$x(t) = \begin{cases} 1 \quad if\ 0 \le t \le t_0/3 \\ -2 \quad if\ t_0/3 \le t \le 2t_0/3 \\ 0 \qquad Otherwise \end{cases} \tag{C.2.1}$$

and

$$c(t) = Cos(2\pi_c ft) \tag{C.2.2}$$

with $f_c = 250\ Hz$, $t_0 = 0.15\ Seconds$, and the modulation index: $mhu = \mu = 0.85$. Furthermore, suppose that the signal has been corrupted by a Gaussian noise with an SNR of 10.0 dB.

Let us note that that $|\max(x(t)| = 2.0$ and re-write the signal in terms of rectangular windows in the time-domain, we have:

$$x(t) = [1 + 0.85 * \frac{x(t)}{2}]Cos(2\pi_c ft)$$

$$= [1 + 0.425 * rect(\frac{t - 0.025}{0.05}) - 0.85 * rect(\frac{t - 0.075}{0.05})].Cos(500\pi t)$$

Term-by-term FT and re-organization of various terms results in:

$$X(w) = 0.05e^{-j0.05\pi f} Sinc(0.05f)(1 - 2e^{-j0.1\pi f})$$

The corresponding modulated signal (convolution in frequency-domain) turns out to be:

$$U(f) = 0.010625e^{-j0.05\pi(f-250)}Sinc(0.05(f-250))(1-2e^{-j0.1\pi(f-250)})$$
$$= 0.010625e^{-j0.05\pi(f+250)}Sinc(0.05(f+250))(1-2e^{-j0.1\pi(f+250)})$$

Power budget of the message and the normalized message:

$$P_x = \frac{1}{0.15}.[\int_0^{0.05} dt + 4.\int_{0.05}^{0.10} dt] = 1.66$$
$$P_x^{normalized} = \frac{1}{4}.P_x = \frac{1.66}{4} = 0.4167$$

The modulation efficiency for the specific values given and computer is:

$$eta = \eta = \frac{\mu^2.P_x^{normalized}}{1+\mu^2.P_x^{normalized}} = \frac{0.85^2 * 0.4167}{1+0.85^2 * 0.4167} = 0.2314 = 23.14\%$$

Finally, the power in the modulated signal is computed as:

$$P_u = \frac{A_c^2}{2} AverageValue\{[1+h * x_{normalized}(t)]^2\}$$
$$= \frac{1}{2}(1+0.3010-1.7*\frac{0.025}{0.15}) = 0.5088$$

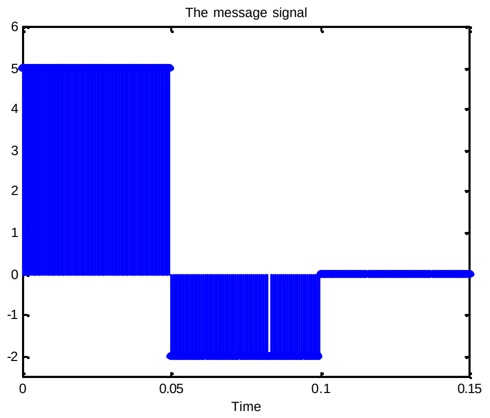
Example C.2.2: Consider the following message signal:

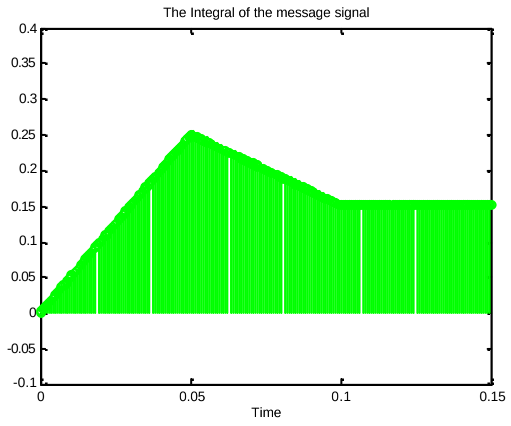
$$m(t) = \begin{cases} 0.5 & 0 < t < T_0/3 \\ -2.0 & T_0/3 < t < 2T_0/3 \\ 0 & Otherwise \end{cases}$$

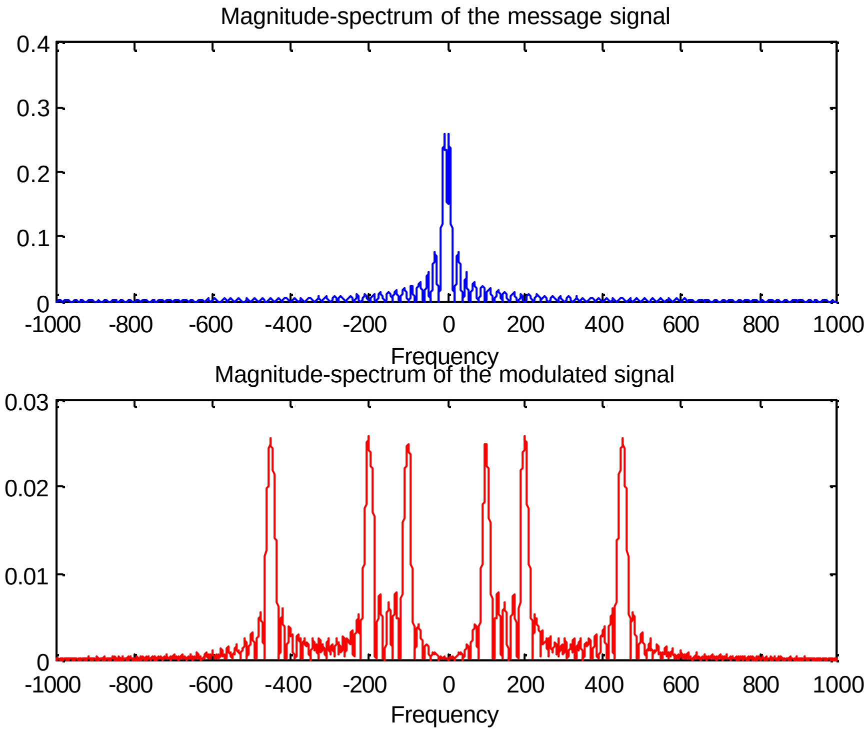
Using Matlab tools demonstrate the FM modulation for this signal with $f_c = 200$, $k_f = 50$, a sampling frequency of $f_S = 2.0\,kHz$ and a frequency resolution of $d_f = 0.25$.

```
% M-File for  Frequency Modulation (FM)
% The message signal is +5.0 for 0 < t < t0/3, -2.0 for t0/3 < t < 2t0/3
% and zero otherwise.
% System Parameters
    t0=.15; ts=0.0005;              % signal duration and sampling interval
    fc=200;   fs=1/ts;              % carrier frequency and sampling frequency
    kf=50;                          % Modulation index
    t=[0:ts:t0];                    % time vector
    df=0.25;                        % required frequency resolution

% Message signal and its Fourier transform
    m=[5*ones(1,t0/(3*ts)),-2*ones(1,t0/(3*ts)),zeros(1,t0/(3*ts)+1)];
    int_m(1)=0;
    for i=1:length(t)-1             % Integral of m
      int_m(i+1)=int_m(i)+m(i)*ts;
    end
    [M,m,df1]=fftseq(m,ts,df);      % Fourier transform
```

```
% Modulation Process
    M=M/fs;                                  % scaling
    f=[0:df1:df1*(length(m)-1)]-fs/2;        % frequency vector
    u=cos(2*pi*fc*t+2*pi*kf*int_m);          % modulated signal
    [U,u,df1]=fftseq(u,ts,df);               % Fourier transform
    U=U/fs;                                  % scaling

% Plots
    pause          % Press any key to see a plot of the message and the modulated signal
    stem(t,m(1:length(t)));axis([0 0.15 -2.5 6.0]);
    xlabel('Time'); title('The message signal'); figure;
    stem(t,int_m(1:length(t)),'g'); axis([0 0.15 -0.1 0.4]),
    xlabel('Time'), title('The Integral of the message signal'); figure;
    plot(t,u(1:length(t)),'r');axis([0 0.15 -2.1 2.1]),
    xlabel('Time'),title('The modulated signal')
    pause          % Press any key to see plots of the magnitude of the message and the
                   % modulated signal in the frequency domain.
    figure;
    subplot(2,1,1), plot(f,abs(fftshift(M)))
    xlabel('Frequency'),
    title('Magnitude-spectrum of the message signal')
    subplot(2,1,2), plot(f,abs(fftshift(U)),'r')
    title('Magnitude-spectrum of the modulated signal')
    xlabel('Frequency')
```







A few important observations from the results associated with this example:

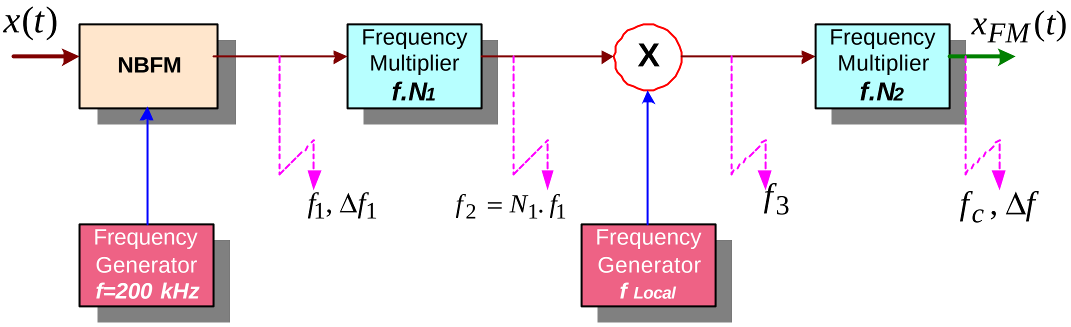
- The spectrum of an FM signal contains a strong carrier component and an infinite number of side frequencies located symmetrically on either side of the carrier at frequency separations of $f_m, 2f_m, 3f_m, \cdots$. Therefore, the result is drastically different from AM, where there were spectra located at two locations only.

- For small b, Bessel coefficients of order $J_0(b)$ and $J_1(b)$ are significant and the rest is almost zero. The spectrum is composed of a carrier and two side frequencies. This is the special case of NBFM (NBPM) as presented earlier.

- $J_n(b)$ oscillates for increasing b with decreasing amplitude though.

- $J_n(b)$ is zero at several values of b. For instance, $J_0(b)$ is zero at 2.4048, 5.5201, and 8.6537, then the spectrum of the output will not contain a component at the carrier frequency for these values of b and hence, they are known as the carrier nulls in the business. Similarly, $J_1(b)$ has nulls at 0, 3.8317, and 7.0156 resulting at zero components at the first sideband locations: $f = f_c \pm f_m$.

Example C.2.3: Let us compute the multiplier and mixer frequencies, $n_1, n_2,$ and f_{LO} for an Armstrong transmitter shown below. In this case, the NBFM crystal oscillator is operating at *200* kHz, with a maximum deviation ratio, $\beta = 0.2$. The frequency range of the input audio signal is expected to be in the range: 50 £ f_m £ 15 kHz, the carrier frequency is $f_c = 108$ MHz, and the expected frequency deviation to be $\Delta f = 75{,}000\ Hz$.



$x(t)$
NBFM
Frequency Multiplier f.N1
X
Frequency Multiplier f.N2
$x_{FM}(t)$
$f_1, \Delta f_1$
$f_2 = N_1 \cdot f_1$
f_3
$f_c, \Delta f$
Frequency Generator f=200 kHz
Frequency Generator f Local

Figure C.2.1 Functional diagram of the FM generator in Example C.2.3.

$$Df_1 = b f_m = (0.2)(50) = 10 \quad \text{Hz}$$
$$Df / Df_1 = 75 \ x\ 10^3 / 10 = 7500 = n_1 * n_2$$
$$f_2 = n_1 f_1 = n_1(200{,}000) \quad \text{Hz}$$

In the down-conversion stage, we need:

$$f_2 - f_{LOCAL} = f_c / n_2$$

Therefore, the relationship among various parameters becomes

$$f_{LOCAL} = n_1 f_1 - \frac{f_c}{n_2} = \frac{7500 \ x\ 200{,}000 - 108 \ x\ 10^6}{n_2} = \frac{1392}{n_2}(10)^6 \quad Hz.$$

If we choose: $n_2 = 150$, then we need a mutliplication factor of: $n_1 = 50$ and the corresponding local oscillator frequency of $f_{LOCAL} = 9.28$ MHz can do the job.

Example C.2.4: Pair of simple RC-circuits are used as pre-emphasis and de-emphasis circuits. In this case the critical frequency values are: $w_1 \approx 2\pi.2100 \ radians / s$; $w_2 \geq 2\pi.30{,}000 \ radians / s.$

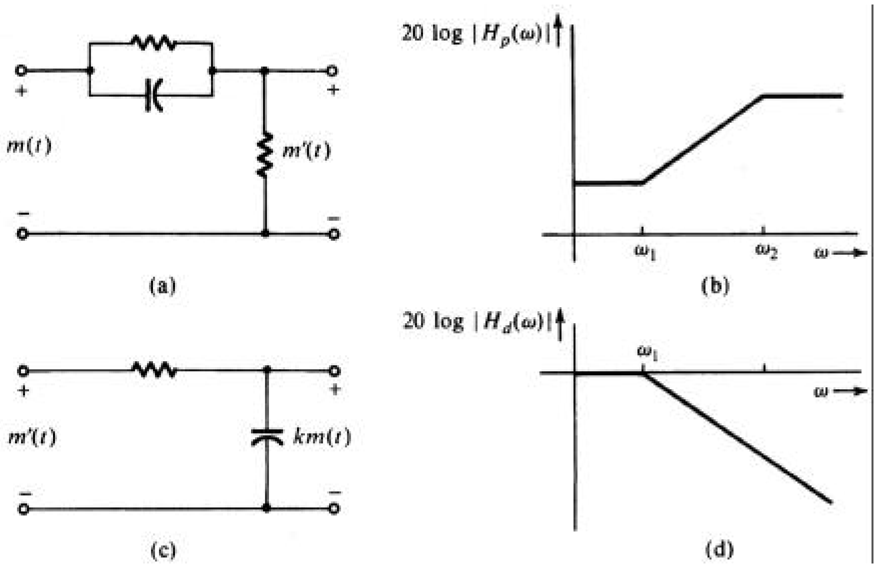


Figure C.2.2. RC-networks for pre-emphasis and de-emphasis in Example C.2.4. a.) Pre-emphasis RC-filter, b.) Log frequency response of (a); c.) De-emphasis RC-filter, and d.) its log frequency response.

The transfer function of the pre-emphasis filter:

$$H_p(\omega) = K\frac{j\omega+\omega_1}{j\omega+\omega_2} \qquad \text{(C.2.3)}$$

Let us set the filter gain to: $K = w_2 / w_1$ and substitute above to obtain:

$$H_p(\omega) = \left(\frac{\omega_2}{\omega_1}\right)\frac{j\omega+\omega_1}{j\omega+\omega_2} \qquad \text{(C.2.4)}$$

Notes:

1. For small values of w, $\omega << \omega_1$, we have $H_P(w) \approx 1$, which implies that low-frequency (l.f.) components of the input signal is unaffected by this operation.
2. However, for the mid-frequency range: $\omega_1 << \omega << \omega_2$, we can approximate the filter frequency response by: $H_p(\omega) \cong j\omega/\omega_1$. Let us recall the Fourier transform property that multiplication by jw in the frequency-domain is equivalent to a differentiation in the time-domain, the filter now acts as a differentiator for 2.1 - 15 kHz.
3. The transfer function of the de-emphasis filter is given by:

$$H_D(\omega) = \frac{\omega_1}{j\omega+\omega_1} \qquad \text{(C.2.5)}$$

4. For low frequencies, that is: $\omega << \omega_2$, we have the property:

$$H_P(w).H_D(w) \approx 1 \quad \textit{in the frequency range}: 0 - 15\,kHz \qquad \text{(C.2.6)}$$

5. Dolby Labs has developed a similar but more elaborate scheme for noise reduction in tapes and other audio applications.