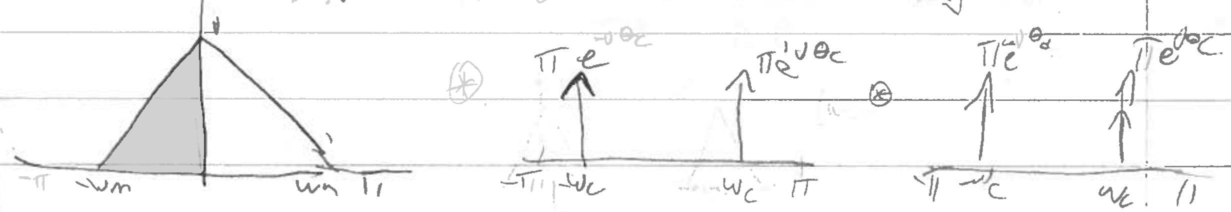
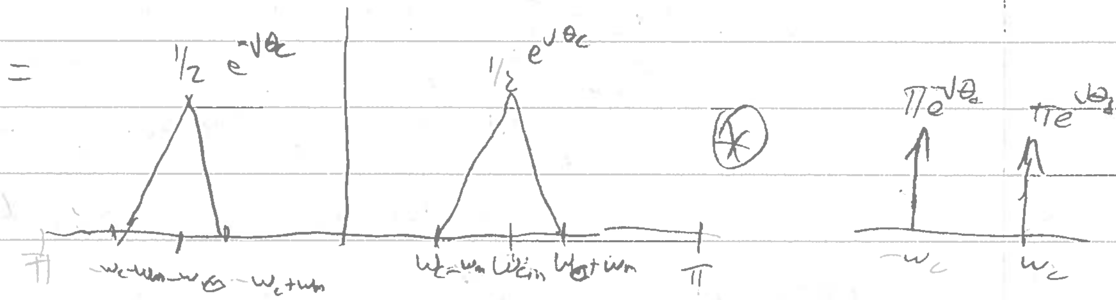
3/3

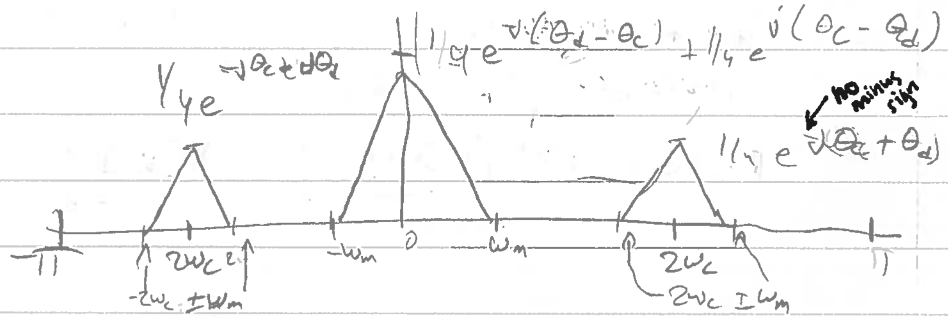
Dan Portz
6.003 PS8

1. 8.47

a). $w[n] = x[n]\cos[\omega_c n + \theta_c]\cos[\omega_c n + \theta_d]$







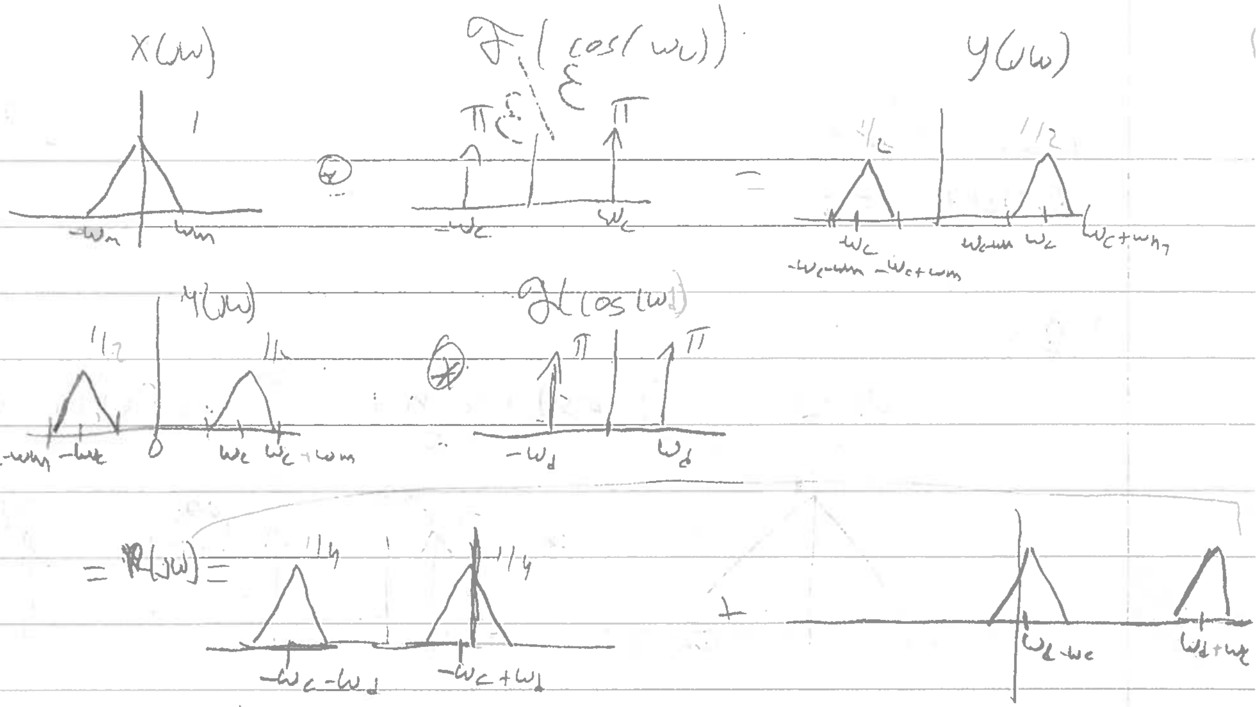
b. We choose $\omega = \omega_m$. Then the spectrum of $r[n]$ is the same as that of $x[n]$ except scaled by $2($

$$2\left(\tfrac14 e^{-j(\theta_d-\theta_d)} + \tfrac14 e^{j(\theta_c-\theta_d)}\right)$$

$$= \tfrac12 e^{j\Delta\theta} + \tfrac12 e^{-j\theta_c} = \cos\Delta\theta$$

That is, $r[n] = x[n]\cos\Delta\theta$

c.



$w[n] = x[n]\cos[w_c n]\cos(w_d n)$

$= x[n]\frac{1}{4}\left(e^{jw_c n} + e^{-jw_c n}\right)\left(e^{jw_d n} + e^{-jw_d n}\right)$

$= x[n]\frac{1}{4}\left(e^{j(w_c+w_d)n} + e^{-j(w_c+w_d)n} + e^{j\Delta w n} + e^{-j\Delta w n}\right)$

$= x[n]\frac{1}{2}\left[\cos((w_c+w_d)n) + \cos(\Delta w n)\right]$

Applying a LPF with cutoff $w_m + \Delta w$ and gain 2, we cancel the $\frac{1}{2}$ factor and remove the first (high-frequency) term

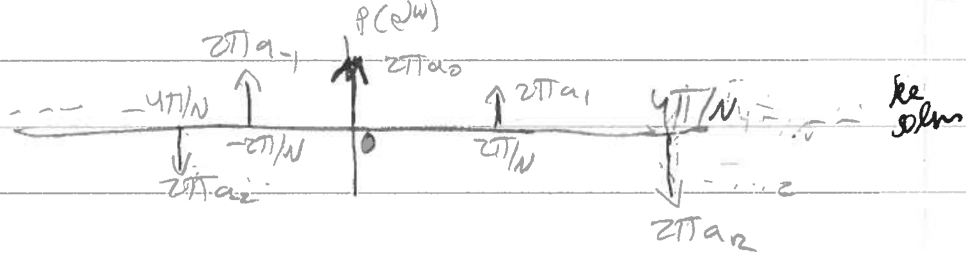
$r[n] = LPF(w[n]) = x[n]\cos(\Delta w n)$

2.8.48 a. Let $\omega_0 = 2\pi/N$.

$p[n]$ is periodic, so it has a Fourier series. Taking the transform of the discrete square wave and applying a phase shift, we find

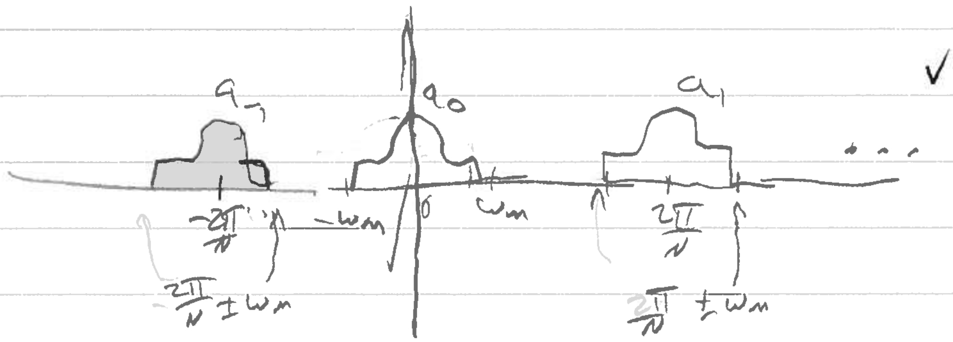
$$a_k = \begin{cases} e^{-jk\omega_0 M/2} \dfrac{\sin\left(k\omega_0 \frac{M+1}{2}\right)}{\sin(k\omega_0/2)}, & k \neq 0 \\ \dfrac{M+1}{N}, & k = 0, \pm N, \pm 2N \ldots \end{cases}$$

Converting to the Fourier transform, we have $P(e^{j\omega}) = \sum_{k=-\infty}^{\infty} 2\pi a_k \delta(\omega - k\omega_0)$



with a_k as given above, periodic w/ $T = 2\pi$

b. Assume the given spectrum of $X(e^{j\omega})$ has magnitude 1. Then, convolving it with the impulse train above (and dividing by 2π), we get:



✓

a_k as given above, periodic w/ $T = 2\pi$

c. To prevent aliasing, we need to ensure no overlap. That is, $2w_m < 2\pi/N$

$\Rightarrow$ max $w_m = \pi/N$. This does not depend on M.

d. Apply a LPF with cutoff frequency w_m and scale $1/a_0 = N/M+1$ to recover $x[n]$.

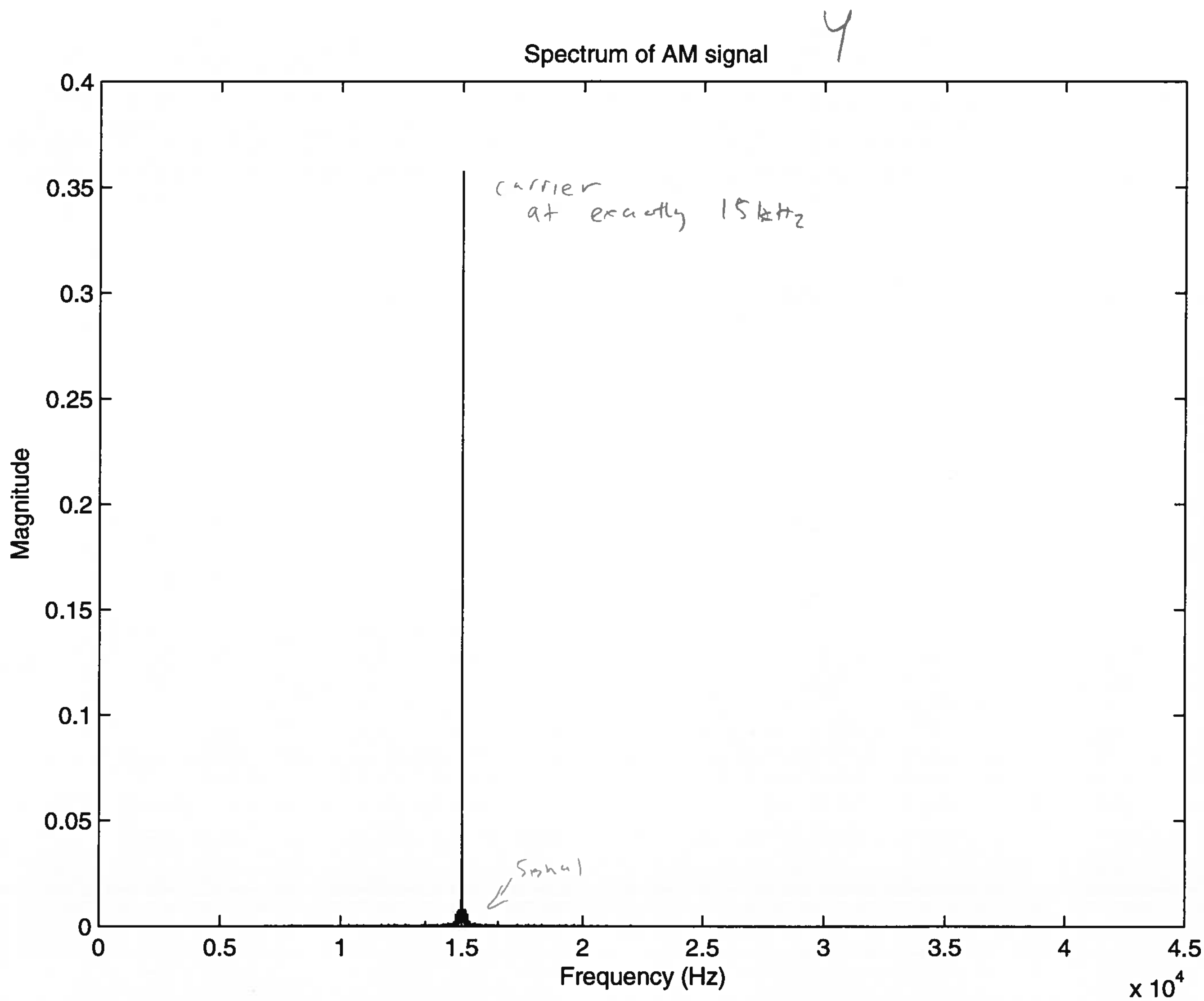
show graphically, or explicitly define the nature of the LPF... the cut-off freq. etc.

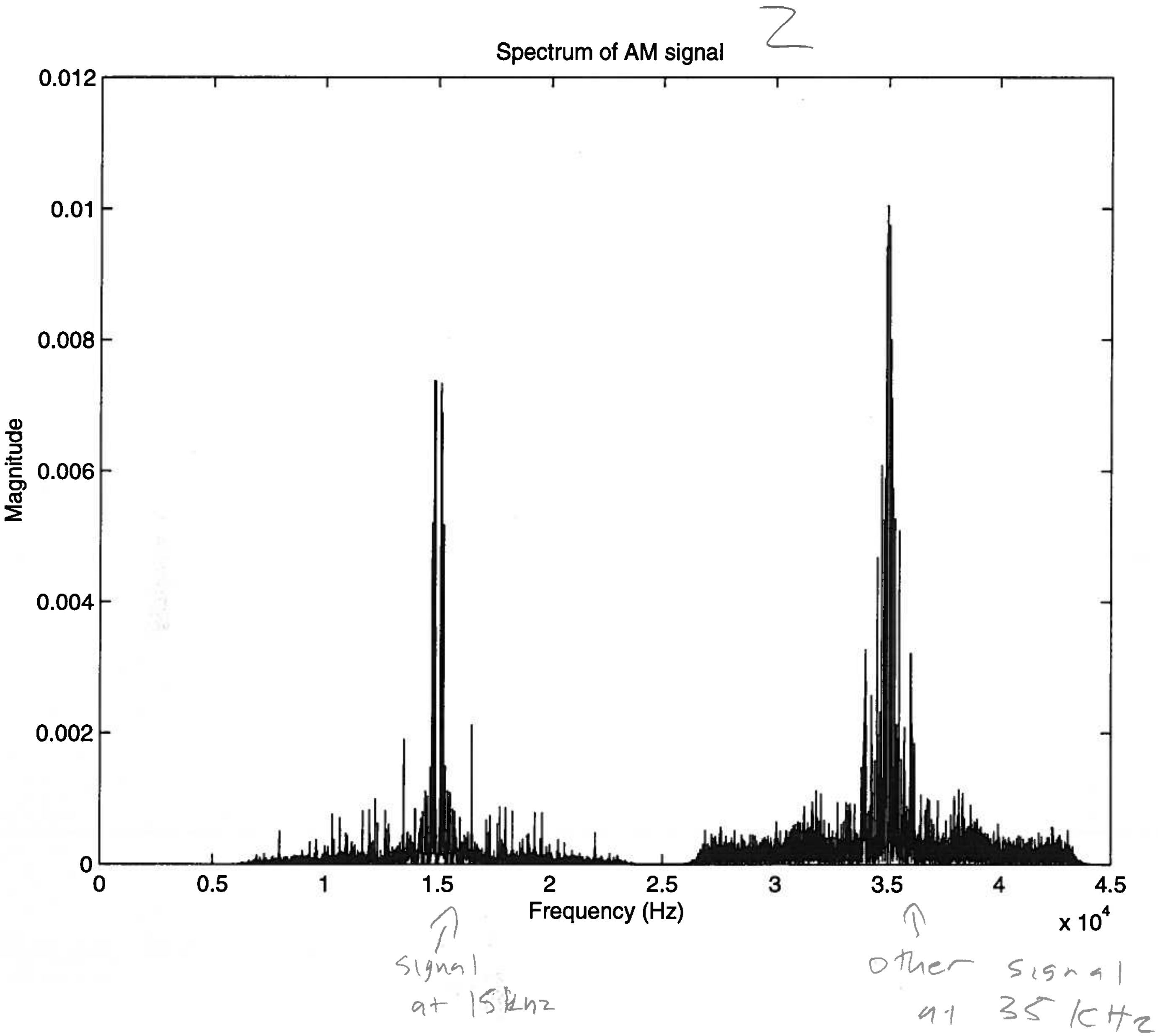
3. a. Printouts of graphs attached.
The sound is clear in Z but nearly inaudible in Y. It fades in and out.

b. Z has the second signal at 35 kHz, as seen on the graph. To modify synchdemod to demodulate it, we just change 15000 to 35000 on ✓ line 24 (printout attached).
The two songs are Bob Dylan's "Shelter from the Storm" and Dido's "Here with Me" ✓

c. The signal fades in and out, with full amplitude when the modulator and demodulator are in phase and zero when they are completely out of phase. This is what is explained in problem 1, parts a and b. ✓

d. We demodulate the signal by using a system essentially like the one in O&W problem 8.26. A printout of the script with explanatory comments is attached. It successfully demodulates the signal.


Spectrum of AM signal
4
carrier
at exactly 15kHz
Signal
Magnitude
Frequency (Hz)
x 10^4
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5



2
Spectrum of AM signal
Magnitude
0.012
0.01
0.008
0.006
0.004
0.002
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
Frequency (Hz)
x 10^4
signal
at 15kHz
other signal
at 35 kHz

Synch demod @ 35 kHz

```
function demod = synchdemod(r)
%Takes a signal r and demodulates using synchronous modulation.
%Assumptions:
% The input signal is sampled at 88.2kHz.
% The input signal is 10 seconds long.
% The carrier frequency is 35kHz.
%Returns:
% The demodulated signal downsampled to 44.1kHz.

%Overall sampling rate, 88.2kHz.
rate = 44100*2;

%Filter order
order = 50;

%A lowpass filter, operates at a sampling rate of 88.2kHz, with cutoff 8kHz.
b = fir1(order, 8e3/(rate/2)); a=1;

%Time axis, our signals are exactly 10 seconds long
t = 0:1/rate:10;
t = t(1:length(t)-1)';

%Generate carrier required for sync demod
mod_35k = cos(2*pi*35000*t);

%Plot spectrum of r
%We need a high fft_len to get good resolution
fft_len = 2^16;
R = fft(r,fft_len);
mag_R = abs(R)/fft_len;

%Our frequency scale, in Hz, positive frequencies only
f = rate*(0:fft_len/2)/fft_len;

figure(1);
plot(f,mag_R(1:fft_len/2+1))
title('Spectrum of AM signal');
xlabel('Frequency (Hz)');
ylabel('Magnitude');

%Synchronous demodulation
%LPF
demod = filter(b,a, mod_35k.*r);
%Rescale
demod = 2*demod;
%Downsample to the audio range
demod = decimate(demod,2);
```

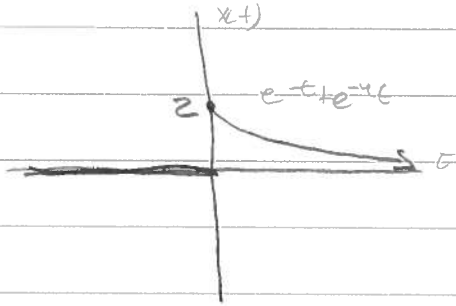
```
%demod is now sampled at 44.1kHz
%Play sound
soundsc2(demod,44100);
```

Asynch demod

```
function demod = asynchdemod(r)
%Takes a signal r and demodulates using asynchronous modulation.
%Assumptions:
% The input signal is sampled at 88.2kHz.
% The input signal is 10 seconds long.
% The carrier frequency is 15kHz.
%Returns:
% The demodulated signal downsampled to 44.1kHz.

%Overall sampling rate, 88.2kHz.
rate = 44100*2;

%Filter order
order = 50;

%A lowpass filter, operates at a sampling rate of 88.2kHz, with cutoff 8kHz.
b = fir1(order, 8e3/(rate/2)); a=1;

%Time axis, our signals are exactly 10 seconds long
t = 0:1/rate:10;
t = t(1:length(t)-1)';

%Plot spectrum of r
%We need a high fft_len to get good resolution
fft_len = 2^16;
R = fft(r,fft_len);
mag_R = abs(R)/fft_len;

%Our frequency scale, in Hz, positive frequencies only
f = rate*(0:fft_len/2)/fft_len;

figure(1);
plot(f,mag_R(1:fft_len/2+1))
title('Spectrum of AM signal');
xlabel('Frequency (Hz)');
ylabel('Magnitude');

%Asynchronous demodulation
%Multiply by sin/cos
c_15k = cos(2*pi*15000*t);
s_15k = sin(2*pi*15000*t);

xc = c_15k.*r;
xs = s_15k.*r;

% LPF at carrier frequency
lpfc = fir1(order, 15000/(rate/2));
```

```
xc = filter(lpfc,a,xc);
xs = filter(lpfc,a,xs);

% Throw out first (order) samples of xc and yc because the filter
% is not effective for those points

xc = xc(order:end);
xs = xs(order:end);

% Square, sum, and square root

s= ((xc.^2)+(xs.^2)).^(.5);

% Subtract average to remove constant added by injection of carrier
s = s-mean(s);

% LPF
demod = filter(b,a, s);
%Rescale
demod = 2*demod;
%Downsample to the audio range
demod = decimate(demod,2);

%demod is now sampled at 44.1kHz
%Play sound
soundsc2(demod,44100);
```

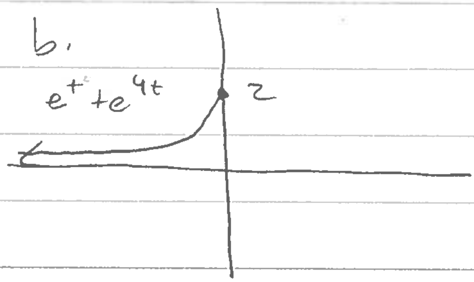
4. a. $x(t) = e^{-t}u(t) + e^{-4t}u(t)$



$X(s) = \frac{1}{s+1} + \frac{1}{s+4}$

ROC: $Re(s) > -1$ ✓

b.

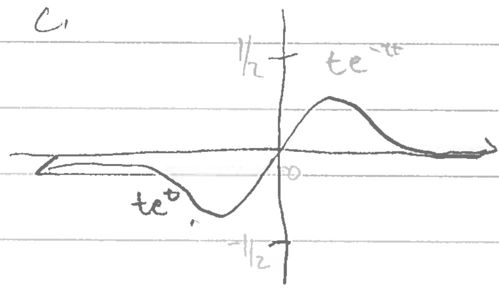


$x(t) = e^{t}u(t) + e^{4t}u(-t)$

$X(s) = \frac{-1}{s-1} + \frac{-1}{s-4}$ ✓

ROC: $Re(s) < 1$

c.

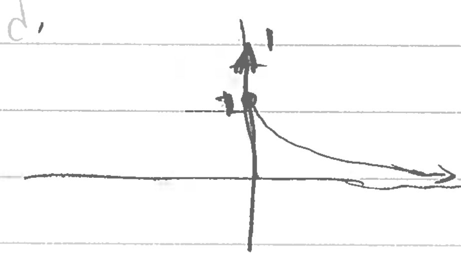


$x(t) = te^{-|t|} = te^{-t}u(t) + te^{t}u(-t)$

$X(s) = \frac{1}{(s+1)^2} - \frac{1}{(s-1)^2}$ ✓

ROC: $-1 < Re(s) < 1$

d.



$x(t) = \delta(t) + e^{-t}u(t)$

$X(s) = 1 + \frac{1}{s+1}$ ✓

ROC: $Re(s) > -1$

ROC

5. a. $X(s) = 1/s^2+16 = \frac{1}{4}\frac{4}{s^2+4^2}$, $Re(s) > 0$

$\Rightarrow x(t) = 1/4 \sin(4t)$ $u(t)$

b. $X(s) = s/s^2+16$, $Re(s) < 0$

This is the same as $\frac{s}{s^2+16}$, $Re(s)>0$ except for a ~~time~~-reversal. Thus,

$x(t) = \cos(-4t)$

$-\cos(4t)u(-t)$

c. $X(s) = \frac{-1}{s^2+5s+6} = \frac{-1}{(s+2)(s+3)} = \frac{A}{s+2} + B/s+3$

$\Rightarrow -1 = A(s+3) + B(s+2) \Rightarrow A = -1$
$B = 1$

$X(s) = \frac{-1}{s+2} + \frac{1}{s+3}$. The poles are at $s = -2$ and $s = -3$, so the ROC is between them and we expect a 2-sided signal.

$x(t) = e^{-2t}u(-t) + e^{-3t}u(t)$ ✓

d. $X(s) = \dfrac{-s^2}{s^2+5s+6} = -1 + \dfrac{5s+6}{(s+2)(s+3)}$

$= -1 + \dfrac{A}{(s+2)} + \dfrac{B}{(s+3)} \Rightarrow A = -4$

$B = 9$

$X(s) = -1 + \dfrac{-4}{s+2} + \dfrac{9}{s+3}$ ROC $Re(s) < 3$.

Poles at -2 and -3. The ROC indicates a left-sided signal.

✓

$x(t) = -\delta(t) + 4e^{-2t}u(-t) - 9e^{-3t}u(-t)$