

GEORGIA INSTITUTE OF TECHNOLOGY  
SCHOOL of ELECTRICAL and COMPUTER ENGINEERING

**ECE 2025 Fall 1999**  
**Lab #11: AM Communication System**

Date: 10–13 April 2000

---

**Lab Quiz #4 will be held during lab next week: 17–20 April;** with emphasis on the last few labs.

This is *the official* Lab #11 description.

The lab report for this lab will be **INFORMAL**, but it will probably be **LONG**. Include derivations as well as plots of the spectrogram to illustrate the spectral content of the different AM channels in part 4. Staple the **Instructor Verification** sheet to the end of your lab report.

The report will **due during the week of 24 April at the start of your lab**.

---

## 1 Introduction & Objective

The goal of this laboratory project is to illustrate the inner workings of a communication system based on AM (Amplitude Modulation). Once a working AM system is programmed, it is possible to use the spectrogram to see how the signal information is being placed into different “channels” in the frequency domain. This also provides a link to the Fourier transform and a mathematical description of the frequency domain behavior of the AM system. Two versions of AM will be investigated: DSBAM-SC and QAM. These acronyms stand for double sideband AM, suppressed carrier and for quadrature amplitude modulation. The lab project will involve the simulation of a receiver for DSB-AM, and the extension of that system to quadrature modulation where two signals can be broadcast simultaneously on one channel by exploiting the real and imaginary parts of the complex exponential. In communication systems, the real and imaginary parts are usually called the *in-phase* and *quadrature* channels, or simply the I and Q channels.

## 2 Background: Fourier Transform and Modulation

Since AM Radio is an analog communication system, we cannot implement it on a computer (i.e., in MATLAB). However, we can carry out a *simulation* of an actual system. This is done by using a very high sampling rate for all the signals so that absolutely no aliasing will occur. Then we can be assured that the digital signals are being processed exactly the same way that the analog signals would be filtered and modulated. If we denote the sampling interval for the simulation as  $T_{\text{sim}}$ , then we are saying that as  $T_{\text{sim}} \rightarrow 0$  the digital simulation will behave exactly like the analog hardware.

When writing MATLAB programs, we would prefer to talk about the sampling rate of the system, which in this case will be  $f_s = 1/T_{\text{sim}}$ . If  $T_{\text{sim}} \rightarrow 0$ , then  $f_s \rightarrow \infty$ , so the higher the sampling rate, the more memory we need for the simulation. To keep the MATLAB vectors down to a reasonable size, we will use 44.1 kHz for the sampling rate and use relatively low frequencies for the carrier frequencies. At 44.1 kHz we will also be able to “listen to the simulation,” because 44.1 kHz is sampling rate for audio CDs.

## 2.1 Double Side-band Amplitude Modulation

Recall the *modulation property* of the Fourier transform for the specific case where one of the signals is a sinusoid: If we multiply  $x(t)$  by the sinusoid  $\cos(\omega_c t + \phi)$ ,

$$\text{DSBAM Definition: } y(t) = x(t) \cos(\omega_c t + \phi) \quad (1)$$

then the Fourier transform is shifted in frequency.

$$Y(j\omega) = \frac{1}{2}e^{j\phi}X(j(\omega - \omega_c)) + \frac{1}{2}e^{-j\phi}X(j(\omega + \omega_c)) \quad (2)$$

The notation  $\omega_c$  uses the subscript “c” to mean “carrier” frequency. Notice that we end up with two complete copies of the input spectrum—one shifted up by  $\omega_c$  and one shifted down by  $\omega_c$ . If the original spectrum  $X(j\omega)$  was centered around zero, then one copy will be shifted up to be centered around  $\omega = \omega_c$  while the other copy will be shifted into negative frequency and be centered around  $\omega = -\omega_c$ . If the signal  $x(t)$  is bandlimited, meaning  $X(j\omega) = 0$  for  $|\omega| > \omega_b$ , then we can choose  $\omega_c$  to be large so that the spectral component  $X(j(\omega - \omega_c))$  will be entirely in positive frequency.

The system depicted in Fig. 1 is an amplitude modulation (AM) system. For reasons that will soon

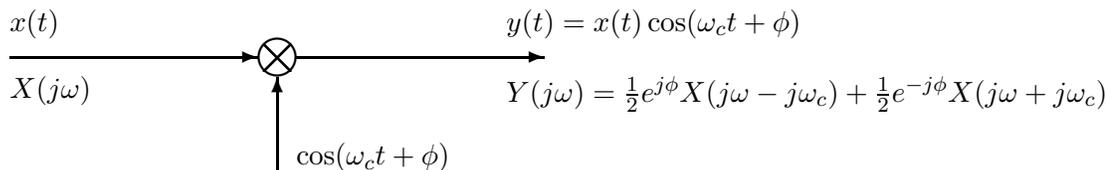


Figure 1: Block diagram representation of a sinusoidal amplitude modulator (also called a *mixer*).

be clear, the system of Fig. 1 is called *double-sideband amplitude modulation* (DSBAM). In this context,  $\cos(\omega_c t + \phi)$  is called the *carrier*, and  $x(t)$  is called the *modulating signal*. When the carrier is very high frequency and  $x(t)$  is slowly varying compared to  $\cos(\omega_c t + \phi)$ , a plot of  $y(t)$  reveals that  $x(t)$  modulates (varies) the amplitude of the carrier  $\cos(\omega_c t + \phi)$ .

The frequency-domain representation of the amplitude modulation system is the Fourier transform of the output of the sinusoidal modulator which is composed of two frequency-shifted copies of the Fourier transform of the input. This is illustrated in Fig. 2.<sup>1</sup> Figure 2(a) depicts a “typical” bandlimited Fourier transform  $X(j\omega)$ . It is typical in the sense that  $X(j\omega) = 0$  for  $|\omega| > \omega_b$ . For now, we are not particularly concerned with the details of the spectrum within the band  $|\omega| < \omega_b$ , so we have simply shown a triangle since it is easy to draw. The main point is that scaled copies of  $X(j\omega)$  are shifted to  $\pm\omega_c$ . If  $X(j\omega)$  is bandlimited as in Fig. 2(a), then the copies  $X(j(\omega \pm \omega_c))$  do not overlap when they are added, and, therefore, they retain all the information in the original signal in a clearly recognizable form in Fig. 2(b). The condition that guarantees that the two copies do not overlap is  $\omega_c - \omega_b > 0$  or  $\omega_c > \omega_b$ . Note that we have shaded the negative frequency part of  $X(j\omega)$  so that we can follow where each part of the spectrum has moved. Figure 2(b) illustrates why this type of modulation is called double side-band amplitude modulation; the amplitude modulation process creates two “side-bands” of frequency around  $\omega_c$ . The lower side-band corresponds to the negative frequencies of  $X(j\omega)$  and upper side-band corresponds to the positive frequencies of  $X(j\omega)$ .

Amplitude modulation moves the frequency spectrum of the signal to a new band of frequencies, and since the original Fourier transform representation appears to be preserved in  $Y(j\omega)$ , it should be possible to recover the original signal from  $y(t)$ . This allows us, for example, to take a lowpass filtered voice signal, whose spectrum is concentrated at frequencies below 5 kHz, and create a signal whose spectrum is concentrated in a 10 kHz band around a much higher frequency such as 750 kHz. The new signal can be transmitted

<sup>1</sup>Figure 2 represents  $X(j\omega)$  with a “typical” bandlimited function.

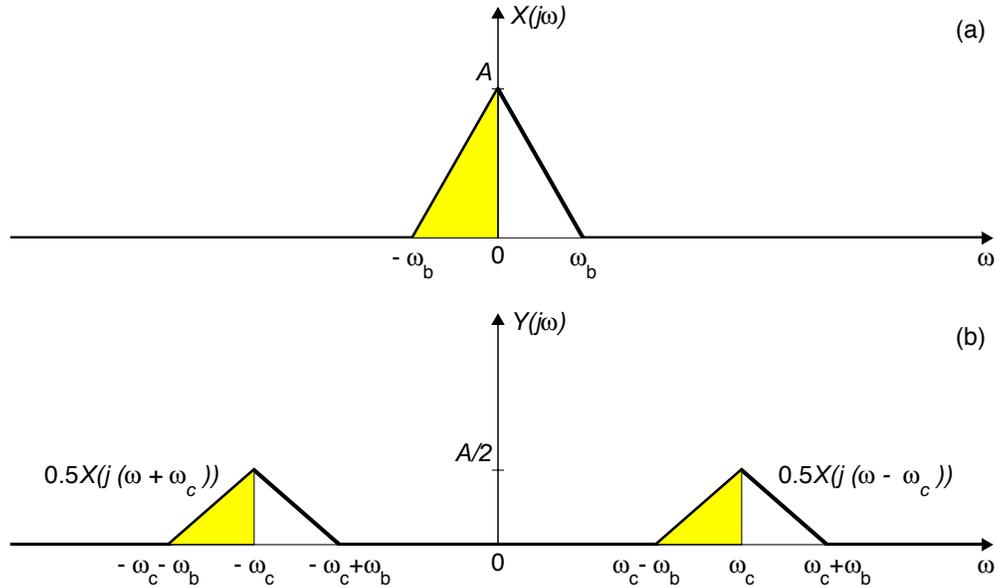


Figure 2: Illustration of sinusoidal amplitude modulation: (a) “Typical” Fourier transform of input; (b) Corresponding Fourier transform of output.

to a remote location using electromagnetic (radio) waves. In addition, a single propagation channel can be shared by a multiplicity of bandlimited signals.

The process of recovering the original signal  $x(t)$  from the amplitude modulation signal  $y(t)$  is called *demodulation*. What we need to do is to re-center the Fourier transform copies on zero frequency. This can be done in two steps: First, we multiply by a sinusoid at the carrier frequency—a system that does this multiplication is often called a *mixer*. Second, we follow the mixer with a lowpass filter as shown in the system inside the dashed box in Fig. 3. The lowpass filter is needed to remove the extraneous copies of the spectrum

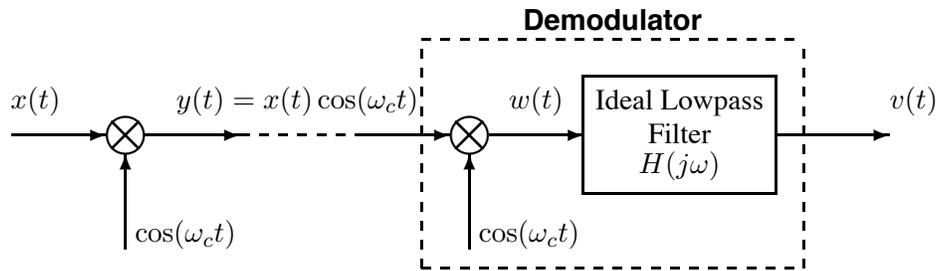


Figure 3: Block diagram of AM modulation followed by demodulation shown for the case where the transmitter and receiver are *synchronized*—both have a phase of zero in the sinusoidal multipliers.

that are created at  $2\omega_c$  by the cosine multiplier. When specifying the ideal communication system, we can use an ideal lowpass filter of the form

$$H(j\omega) = \begin{cases} G & -\omega_{co} < \omega < \omega_{co} \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

but the MATLAB implementation will require a “realizable” lowpass filter  $H_r(j\omega)$  whose passband edge

( $\omega_{cp}$ ) is lower than its stopband edge ( $\omega_{cs}$ ):

$$H_r(j\omega) = \begin{cases} G & -\omega_{cp} < \omega < \omega_{cp} \\ 0 & \omega_{cs} < |\omega| \\ \text{“transition”} & \omega_{cp} < |\omega| < \omega_{cs} \end{cases} \quad (4)$$

In the “transition region” the filter’s magnitude response should go smoothly from  $G$  to 0.

## 2.2 Practical Considerations

Using non-ideal filters is one practical consideration, but there are others. Most important is the fact that it is extremely difficult to synchronize the frequencies and phases of two sinusoidal oscillators that are not in the same physical location. This is illustrated by the modulation/demodulation system in Fig. 4, which shows the general case where the carrier frequencies at the modulator and demodulator are different and there is a phase shift of  $\psi$  between the two carriers.

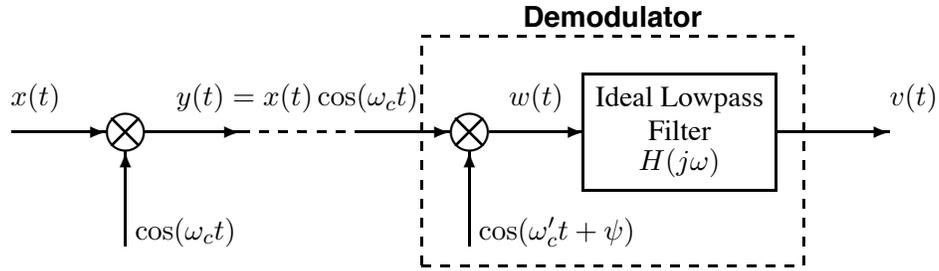


Figure 4: Block diagram of sinusoidal modulation followed by demodulation with carriers differing in frequency and phase.

If the carriers in Fig. 4 have the same frequency, but are out of phase, the output will be proportional to  $x(t)$ , but with an amplitude scale factor that depends on  $\psi$ . When  $\psi = 0$  the carriers are “in-phase”; when  $\psi = \pm\pi/2$  they are said to be “90° out of phase.”

On the other hand, if the carriers are not exactly synchronized in frequency, the Fourier transform of  $x(t)$  is not shifted all the way back to where it originated. In the time-domain, this can cause the signal  $x(t)$  to fade in and out at a rate determined by  $\omega_c - \omega'_c$ . The effect is the same as the “beat note” phenomenon observed when two sinusoids of nearly the same frequency are added.

## 2.3 DSBAM with Transmitted Carrier (DSBAM-TC)

Problems with carrier frequency/phase synchronization was originally solved by using a DSBAM system in which carrier information was transmitted along with the modulating signal. This led to very inexpensive receivers and the proliferation of “transistor radios” in the 1960s. The input/output relation for a DSBAM system with transmitted carrier (DSBAM-TC) is

$$\begin{array}{l} \text{DSBAM-TC Definition.} \\ y(t) = [C + x(t)] \cos(\omega_c t) = e(t) \cos(\omega_c t), \end{array} \quad (5)$$

where  $e(t) = C + x(t)$ . The constant  $C$  is chosen so that  $C \geq \max\{|x(t)|\}$  so that  $e(t) \geq 0$  for all  $t$ . Since  $e(t)$  is always positive, it is called the envelope of the peaks of the modulated cosine wave.

If we expand the expression for  $y(t)$ , we obtain

$$y(t) = C \cos(\omega_c t) + x(t) \cos(\omega_c t), \quad (6)$$

which shows that the DSBAM-TC signal contains a cosine of amplitude  $C$  and also a component that is identical to the DSBAM signal. This is why this type of modulation is called *double side-band amplitude modulation with transmitted carrier*. By transmitting the carrier signal and ensuring that the envelope  $e(t)$  is always positive, it becomes possible to demodulate with a much simpler system, called an “envelope detector.” We can extract the envelope of the DSBAM-TC signal by using a half-wave rectifier followed by a lowpass filter (similar to the power supply circuit). In a practical setting, the combination of half-wave rectifier and lowpass filter is generally approximated very well by an electronic circuit consisting only of a diode, resistor, and capacitor. For this reason, AM receivers can be made very economically.

## 2.4 Frequency Division Multiplexing

The Fourier spectrum of a bandlimited signal can be shifted by amplitude modulation to a higher frequency band as is required for radio transmission of voice, music, or data. In addition, we can choose multiple carrier frequencies such that the shifted spectra of two or more bandlimited signals do not overlap, so we can simultaneously transmit many signals over the same channel (e.g., the atmosphere or outer space). This stacking of signals in the frequency domain is called *frequency-division multiplexing* or *FDM*. The essence of frequency-division multiplexing is depicted in Fig. 5, which shows two bandlimited signals that modulate different sinusoidal carriers of frequencies  $\omega_{c1}$  and  $\omega_{c2}$ . The two DSBAM signals  $y_1(t) = x_1(t) \cos \omega_{c1}t$  and  $y_2(t) = x_2(t) \cos \omega_{c2}t$  are superimposed (added) and so their Fourier transforms are added too.<sup>2</sup> The plot in the middle of Fig. 5 depicts the resulting Fourier transform  $Y(j\omega) = Y_1(j\omega) + Y_2(j\omega)$ . Note that the copies of the original bandlimited Fourier transforms must not overlap, which requires that  $\omega_{c2} > \omega_{c1} + 2\omega_b$ .

The process of sorting out the individual signals from the sum is called *demultiplexing*. If the shifted frequency spectra of the input signals do not overlap, then we can recover each of the individual signals  $y_1(t)$  and  $y_2(t)$  from the sum with bandpass filters. For example, to recover the signal  $x_1(t)$ , we need the bandpass filter in Fig. 6(a), which is centered at  $\omega_1$  and selects the channel containing the spectral information for  $y_1(t)$ . Once  $y_1(t)$  has been extracted from  $y(t)$  by bandpass filtering, we can extract the information signal  $x_1(t)$  by removing the carrier signal—this process is called demodulation. Figure 5 shows a *synchronous* demodulator for DSBAM which consists of a mixer and a lowpass filter.<sup>3</sup> The system of Fig. 5 can be generalized to more than two channels simply by choosing more carrier frequencies that are spaced by  $2\omega_b$  plus an additional amount to allow for non-ideal bandpass filters.

Government regulation has set aside certain portions of the frequency domain are set aside for specific applications and modulation techniques.<sup>4</sup> For example, the range of carrier frequencies for the AM broadcast system is 530 kHz to 1620 kHz with the channels (stations) separated by 10 kHz. Since both sidebands must fit into the 10 kHz, the input voice and music signals must be bandlimited to nominally 5 kHz before modulation. In the case of the FM broadcast system, the carrier frequencies range from 88.1 MHz to 107.9 MHz with a spacing of 200 kHz. In this case, the input signals to an FM modulator are bandlimited to 15 KHz, but the FM modulation process expands the bandwidth in addition to shifting the resulting spectrum to be centered on the carrier frequency. In broadcast applications such as these, the receiver “tunes in” to one channel at a time. Therefore, the demultiplexer generally consists of a single bandpass filter whose center frequency can be adjusted to the desired carrier frequency.

<sup>2</sup>In radio broadcasting, the signals are naturally superimposed because each station radiates its signal into the atmosphere.

<sup>3</sup>If DSBAM-TC were used, then the demodulator would be a simple envelope detector.

<sup>4</sup>Here’s one link that discusses spectrum allocation: <http://www.rentcom.com/wpapers/telex/telex3.html>, or get a PDF version of the entire frequency spectrum from <http://www.ntia.doc.gov/osmhome/allochrt.html>.

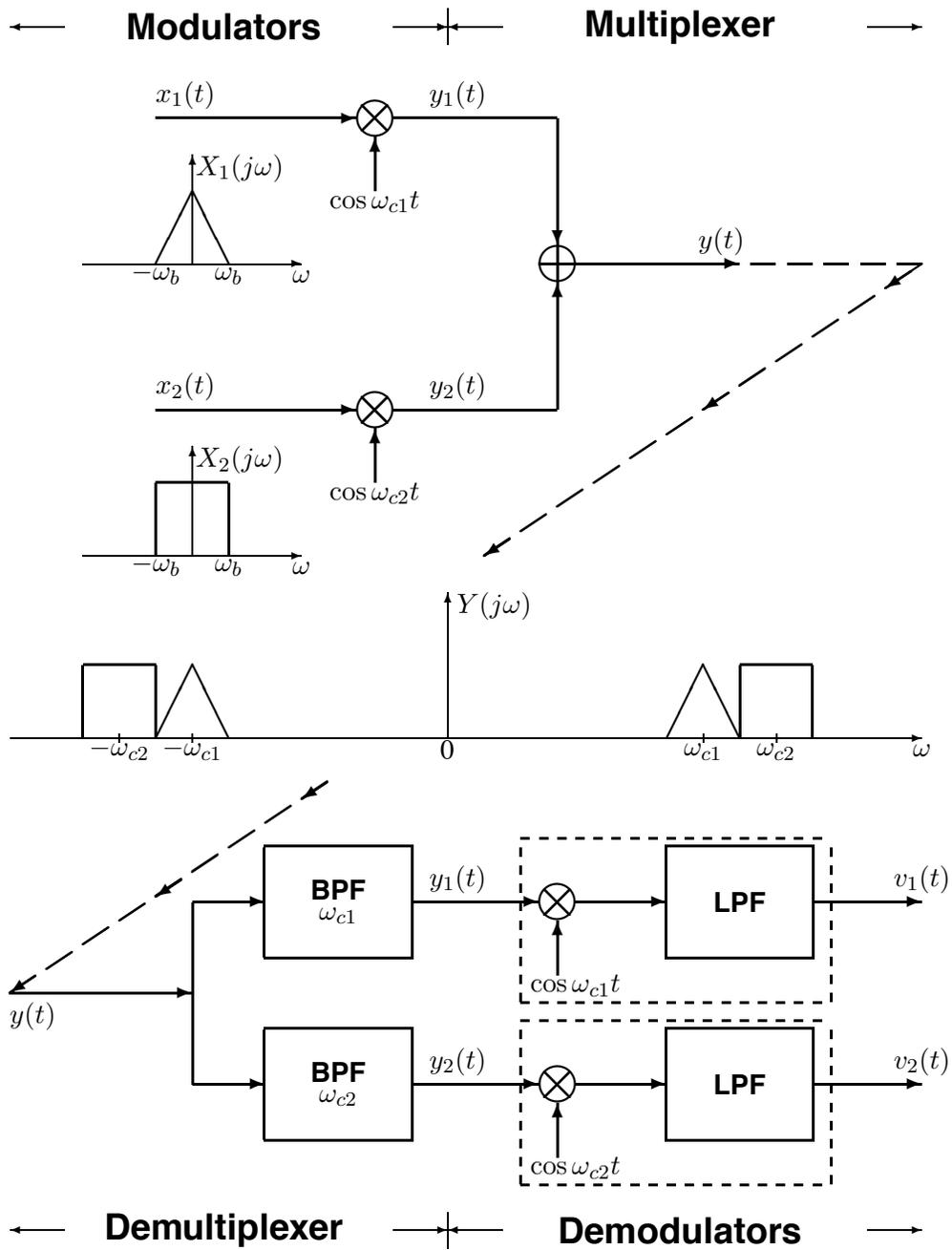


Figure 5: Frequency-division multiplexing for the two-channel case.

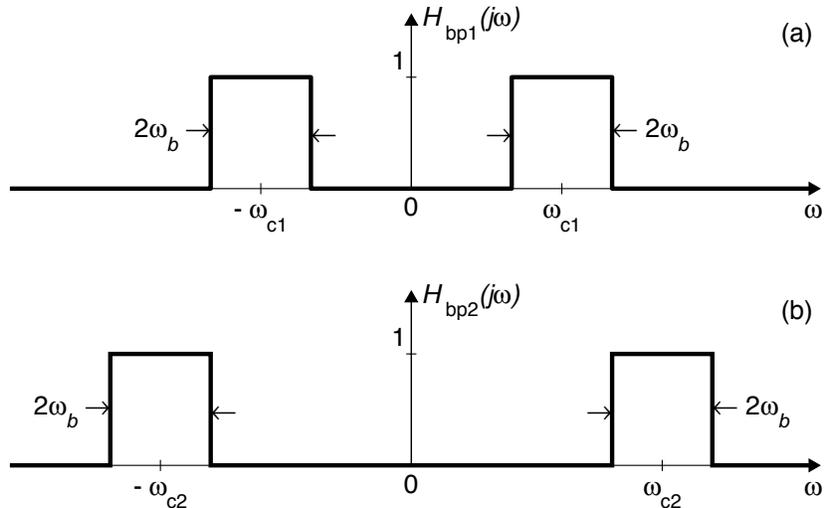


Figure 6: Bandpass filters for demultiplexing: (a) Channel centered on  $\omega_{c1}$ , (b) Channel centered on  $\omega_{c2}$ .

### 3 Warmup

In this warm-up you must test out the pieces of a simple AM system that can operate on one channel. Since we are trying to study an *analog system*, we can only do a *simulation*. We know that any simulation requires a sampling rate for the simulation—in this case, we will use  $f_s = 44,100$  Hz. This makes it possible to do the simulation in MATLAB, but keep in mind that real AM transmission systems operate with carrier frequencies near 1 MHz ( $10^6$  Hz). Simulating a system operating at those rates would require a “simulation sampling rate” in the range of 5 to 10 MHz.

#### 3.1 Modulation

Demonstrate that you can create a modulated signal. For the test signal use a sinusoidally modulated chirp:

$$x_1(t) = \cos(2\pi(800)t + 30 \cos(2\pi(20)t))$$

- Determine the instantaneous frequency of  $x_1(t)$  by doing a simple mathematical derivation. What is the maximum (and minimum) frequency in  $x_1(t)$ ?
- Write some MATLAB code to create  $x_1(t)$  with a duration of 0.6 seconds, then modulate it with a carrier frequency of  $\omega_c = 2\pi(3600)$  rad/sec.
- Make a plot of the spectrogram of the signal to verify that the modulated signal has the desired frequency location. **Since the sampling rate for the simulation is so high, you should change the parameters `NFFT` and `window` in `specgram.m` to be longer than their default values.** Good choices would be 1024 and 512, so use `specgram(yy, 1024, fsim, 512)`. Another consequence of the high simulation sampling rate is that you should zoom to see the (relatively small) frequency region of interest near 3600 Hz. Explain to your TA where you see the frequency content of  $x_1(t) \cos(\omega_c t)$  and why it is correct.

**Instructor Verification** (separate page)

NOTE: the MATLAB `specgram.m` command will make a plot of only the *positive frequency region*, but you can visualize the negative frequencies by recognizing that symmetry applies (the Fourier transform is conjugate symmetric with respect to  $\omega = 0$ ).

### 3.2 Analog Filter Simulation

The demodulator will contain an analog bandpass filter to separate out the channel of interest.<sup>5</sup> A MATLAB function called `anfilt.m` is available for “simulating” the behavior of an “almost-ideal” analog filter.<sup>6</sup> This M-file can be downloaded from Web-CT. The function `hh = anfilt( fc, BW, fsim )` takes three input arguments: the center frequency `fc` in Hz for the passband of the bandpass filter, the total width `BW` in Hz for the passband of the bandpass filter, and the simulation sampling rate `fsim` in Hz for the analog system. The output from `hh = anfilt( fc, BW, fsim )` is the impulse response of the analog filter, but, of course, the impulse response is being sampled at the simulation rate, `fsim`.

Normally, the frequency response of the analog filter should be obtained by “taking the Fourier transform” of the impulse response. But, this is where things get a little confusing, because the simulation of the analog filter is actually a digital filter. To get the frequency response you should follow a three step procedure:

- i. Recognize that the impulse response is a discrete-time signal,  $h[n]$ .
  - ii. Use `freqz()` to get the frequency response of the digital filter that is doing the simulation. If you made this plot, you would use a horizontal axis whose variable is  $\hat{\omega}$  from  $-\pi$  to  $\pi$ .
  - iii. Convert the frequency axis from  $\hat{\omega}$  to  $\omega$ . This requires that you use the simulation sampling frequency to get the correct units for the frequency axis.<sup>7</sup> A convenient way to figure out this rescaling is to remember that  $\hat{\omega} = \pi$  corresponds to half the (simulation) sampling frequency.
- (a) The task in this part is to design an analog filter whose passband extends from 3000 Hz to 5000 Hz.
  - (b) Then make a plot of the frequency response versus  $\omega$  in rad/sec to verify that the passband is in the correct place.
  - (c) The parameter `BW` approximates the width of the passband, but it measures that width between the 0.9 points. Thus, you should examine the location of the passband and stopband of the simulated filter using the plot from part (b). If you measure the location of the passband as the region where the filter gain is above 0.9, what is the width of the passband? Likewise, if you measure the stopband edges as the points where the filter gain is below 0.1, how far are the stopband edges from the passband edges? This distance between passband and stopband edges is called the “transition zone” of the BPF. It must be taken into account when trying to pack AM channels close together, because we cannot build ideal BPFs to demultiplex the AM channels.

**Instructor Verification** (separate page)

### 3.3 Demodulation: the Mixer

The demodulation section consists of two parts: a cosine multiplier, usually called a *mixer*, followed by a lowpass filter. In this part, you should examine the spectrum of the signal at the output of the mixer.

- (a) Take the signal from Section 3.1(b) and multiply by a cosine of the correct frequency to do the mixing.
- (b) Plot the spectrogram of the signal at the output of the mixer. Explain all the spectral components that you see in the spectrogram. Zoom in to see details if necessary.

**Instructor Verification** (separate page)

<sup>5</sup>This bandpass filter is not essential, but we’ll use it in this lab.

<sup>6</sup>Since it is a simulation, the filter is actually an FIR digital filter.

<sup>7</sup>you will be creating what is called the “effective analog filter.”

As a final comment, the lowpass filter needed when implementing the AM demodulator (in the next section) can also be designed with `anfilt.m`.

### 3.4 MATLAB Demo

In MATLAB's signal processing toolbox, there is a demo called `moddemo` which illustrates the AM system, as well as some other modulation schemes. The demo can show the time waveforms and also the spectrograms in a DSBAM-TC system. It works fine for the speech input, but is not as good for the sine input.

## 4 AM Communication System

In this part of the lab project you must do the entire implementation of a two-channel AM communication system. The transmitter and receiver sections should be written as separate MATLAB M-files. The assigned channel frequencies are 3600 Hz and 8700 Hz. One design question that you will have to answer is how wide can the channels be (in Hz). This is dependent on the filters that are used, as well as the channel frequencies.

### 4.1 Transmitter: `myxmit.m`

Write a MATLAB function, called `myxmit.m`, that will take two baseband signals<sup>8</sup>  $x_1(t)$  and  $x_2(t)$  and produce the modulated signal which is called  $y(t)$  in Fig. 5.

- Provide the MATLAB code for `myxmit.m` in your report.
- For testing, use the signal from the warm-up Section 3.1(b) for  $x_1(t)$  but change its duration to 1.25 secs. For the second signal,  $x_2(t)$ , use a constant-frequency sinusoid at  $\omega = 2\pi(1000)$  rad/sec. with the same duration. Plot a spectrogram of the modulated signal  $y(t)$  to show that the signals are in the correct place in the frequency domain.
- Determine the *maximum usable bandwidth* of the two-channel AM system. This is the width of the AM channels in Fig. 5; assuming both channels have the same width. Remember that the maximum bandwidth must accommodate the baseband signals are going to be modulated and transmitted. There can be no frequency overlap of the two transmitted channels, so the highest frequency of the baseband signal ( $\omega_b$ ) is limited such that  $2\omega_b$  is less than the maximum usable bandwidth.

Note: this “maximum usable bandwidth” will be recalculated in the section dealing with the receiver.

### 4.2 Receiver: `mytuner.m`

In this section, you need to implement the synchronous AM receiver, which was shown in Fig. 5.

- Write a MATLAB function that will “tune” to any AM channel, like the dial on a radio. Call the function

$$vv = \text{mytuner}(yy, f\text{station}, \text{mixerphase})$$

where the arguments are `yy`, the signal into the demodulator, `fstation`, the frequency of the station you want to hear, and `mixerphase`, the phase of the multiplier sinusoid in the mixer. The output `vv` is the demodulated signal. Having these two parameters will facilitate the experiment in Section 4.4. Write the function `mytuner(yy, fstation, mixerphase)` with the following capabilities:

---

<sup>8</sup>The term *baseband* refers to the fact that a signal such as speech or music has a Fourier transform that is centered around  $\omega = 0$ .

- Pick the bandwidth of the BPF to be as wide as possible for the two given stations (3600 Hz and 8700 Hz), but obey the following constraint: the gain at the neighboring channel that could interfere should be less than  $1/100 = 0.01$ . This is a tighter spec than in the warm-up, so you will have to recalculate the stopband edges and the passband edges.
  - When the receiver tunes to a new station, the bandpass filter must move in frequency. Thus, your M-file should design the BPF filter “on the fly” using the frequency of the station. The bandwidth of the BPF should be the same for both channels and should not depend on `fstation`.
  - Design the lowpass filter (after the mixer) with `anfilt.m` so that any other interfering spectral components are removed. State what frequency you used for the passband cutoff of the LPF.
- (b) Plot the frequency response of one of the BPFs for both magnitude and phase. Since the passband is the most interesting part, plot only the region near the passband to show that your filter design meets the stopband and passband specs.
- (c) Since the filter doing the simulation is a digital filter, the implementation of the filter for processing signals must be done in the “index domain,”  $n$ . Therefore, use the appropriate MATLAB function for the digital filtering. For your lab report, include a listing of your MATLAB program that simulates the receiver.
- (d) Based on your BPF design in part (a), recalculate the “maximum usable bandwidth” of the overall AM system. Explain how the non-ideal characteristic of the analog BPF limits the “maximum usable bandwidth” to a smaller value than the one found previously in Section 4.1.

### 4.3 Testing and Listening

First of all, verify that your system works on the synthetic signals from Section 4.1(b). This can be done with a “listening test” because the simulation rate is 44.1 kHz. Listen to the input signal and then the demodulated output signal to verify that they sound identical. Both the chirp and the sinusoid should be well within the bandwidth of the DSBAM system, so there should be virtually no loss of signal quality. Make plots of the input and output signals, and then measure the amplitudes of the input and output signals to verify that they are the same.

Now test the entire AM mod-demod system with two “real” signals from the file `harp2k.mat`. The sampling rate for both is 44.1 kHz. The signals are called `xx1` and `xx2` when loaded into MATLAB with the load command. Run `myxmit()` followed by `mytuner()` to verify that the system works correctly and that you can identify the two signals at the output. Note: you do not have to turn in any plots for this part; just write a brief description of the two “real” signals and what you heard.

### 4.4 Testing and Degradations

Now it is time to run tests on your system to show how it degrades when the phase and frequency of the receiver’s mixer are not perfectly synchronized with the transmitter.

- (a) Try the following values for the phase of the mixing sinusoid:  $\psi = \pi$  and  $\pi/2$ ; remember that you’ve already been using  $\psi = 0$ . Write down your observations about the amplitude of the output signal from the demodulator—this is the signal called  $v_i(t)$  in Fig. 5. Try to make an accurate measurement of the output amplitude.
- (b) Explain the observations from part (a) by drawing sketches (by hand) of the Fourier transform at the mixer output for the two cases. Do this for the 8700 Hz channel only. If you make a spectrogram, it will

help in drawing the sketches of the Fourier transform, but note the scale on the spectrogram because functions like `specgram()` will automatically scale to the maximum value.

- (c) Now set the phase back to zero, but change the tuning frequency. Listen to the output when the tuning frequency is slightly off from the correct value. For example, the channel at 8700 Hz should be observed for tuning frequencies such as 8703 Hz, and 8707 Hz. Describe the effect that you hear.
- (d) Do some mathematical analysis to explain what you heard in the previous case; and then draw a sketch of the spectrum for the following case: One situation that *can be analyzed in great detail* is the case where  $x_2(t) = \sin(2\pi(1000)t)$ , and the mixing frequency is 8707 Hz. In this case you should be able to derive an exact expression for the final output of the AM system and make a detailed sketch of the Fourier transform at the output. Finally, you should be able to relate the sketch (or formula) to what you hear.

#### 4.5 QAM: Quadrature AM System

In a quadrature AM system, the transmitter can use either  $\cos(\omega_c t)$  or  $\sin(\omega_c t)$  when doing the modulation, or both. As illustrated in the previous part, the receiver works correctly when its mixer is “in-phase” with the transmitter. Putting these two ideas together, it is possible to *double the capacity* of the AM transmission system by *exploiting the phase dependence* of the receiver.

For this part, you should use your function `mytuner()` in a “trial-and-error” mode to find all the signals buried in the “mystery transmission” called `lab11s00.mat` which can be downloaded from Web-CT. The mystery signal is about 2 secs. in duration and is sampled at the simulation rate of 44.1 kHz. The transmit channels are still 3600 Hz and 8700 Hz, but the transmitter uses a *quadrature* modulation scheme.

- (a) Process the “mystery transmission” multiple times to see how many signals you can extract from it. Use your function `mytuner()`, but vary the phase as you did in Section 4.4(a). Listen to the outputs each time. Describe in detail what you heard and how you extracted the different signals. How many did you find? Include a list of the descriptions of the extracted signals in your lab report, and give the settings that you used for `mytuner()` in each case.
- (b) Explain using Fourier transforms, mathematics and sketches how the system is able to send more than two signals over the two AM channels. One aspect of this is to explain why “phase matters” in the receiver’s mixer, but it is more important to figure out how the transmitter is forming the signals. Give a formula for the time signal that is the output of the transmitter. Write the formula as a summation in terms of all the input signals,  $x_i(t)$  and the sines and cosines at the two carrier frequencies.

**Lab #11**  
**ECE-2025 Spring-2000**  
**Instructor Verification**

Name: \_\_\_\_\_

Date of Lab: \_\_\_\_\_

Part 3.1: Show the spectrogram of an amplitude modulated chirp signal. Explain to your TA the expected range for the frequency content of  $x_1(t) \cos(\omega_c t)$ .

Verified: \_\_\_\_\_

Date/Time: \_\_\_\_\_

Part 3.2: Design an analog filter with a specified passband, and make a plot of its frequency response (magnitude) versus  $\omega$  in rad/sec. Determine the stopband region from the plot.

Verified: \_\_\_\_\_

Date/Time: \_\_\_\_\_

Part 3.3: Determine the output of the mixer in an AM demodulator. Show the spectrogram. Explain all the spectral components that you see in the spectrogram.

Verified: \_\_\_\_\_

Date/Time: \_\_\_\_\_