Project 4: Optical Communications Link

In this project you will build a transmitter and a receiver circuit. The transmitter circuit uses pulse frequency modulation to create a series of light pulses that encode an audio signal. The receiver takes in the light pulses, demodulates them, amplifies the signal back to the level the audio amplifier needs to see, and sends it through an audio amplifier to a speaker where it can be heard. Your initial design will reconstruct the input to create an audible signal of poor quality. In the final design, you will modify the circuit to create a much better quality demodulated audio signal.
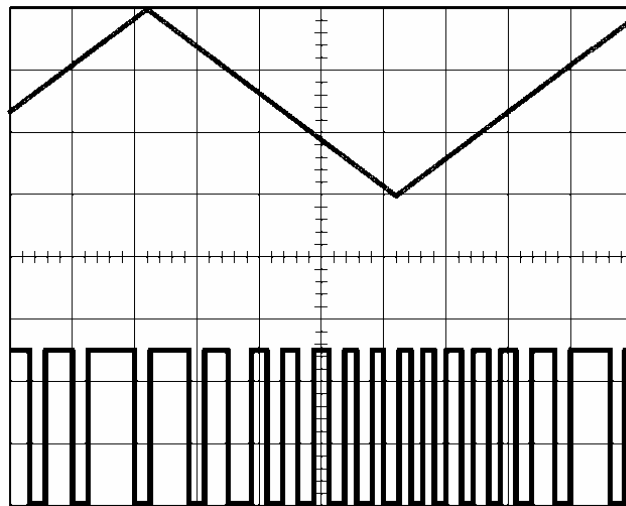

Figure 1

Figure 1 shows an input signal (top) and its pulse frequency modulated equivalent. Simply averaging this signal will demodulate it to some extent. The initial design for the receiver circuit provides this functionality to a small degree more-or-less by accident. If you enhance the time-averaging ability of the circuit with the addition of an integrator, you will be able to improve the reconstruction of the audio signal. The integration, however, will attenuate the signal level somewhat. To make up for this, you will have to adjust the gain of the circuit after the integrator is added. You will also be asked to add a smoothing capacitor to the circuit to further improve your output signal. The modified circuit is to be tested to be sure that it is an improvement over the initial design.
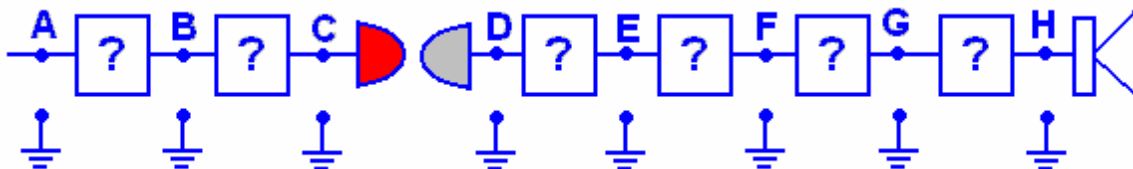

Figure 2

Another part of this project is to determine what each block of the circuit you are building does. Figure 2 shows a block diagram for the circuit. The piece of the circuit between each of the marked points is a sub-circuit that performs some function. You have encountered most of these smaller circuits sometime during the semester. (Information about the audio amplifier is contained in this handout.) You should be able to identify the function of each block shown in figure 2 and show that the block is functioning as expected by comparing the signal before the block to the signal after it.

Initial Design

In the initial design of this circuit you will build a transmitter and a receiver. You will also look at the function of a PSpice model of the circuit.

Transmitter Circuit
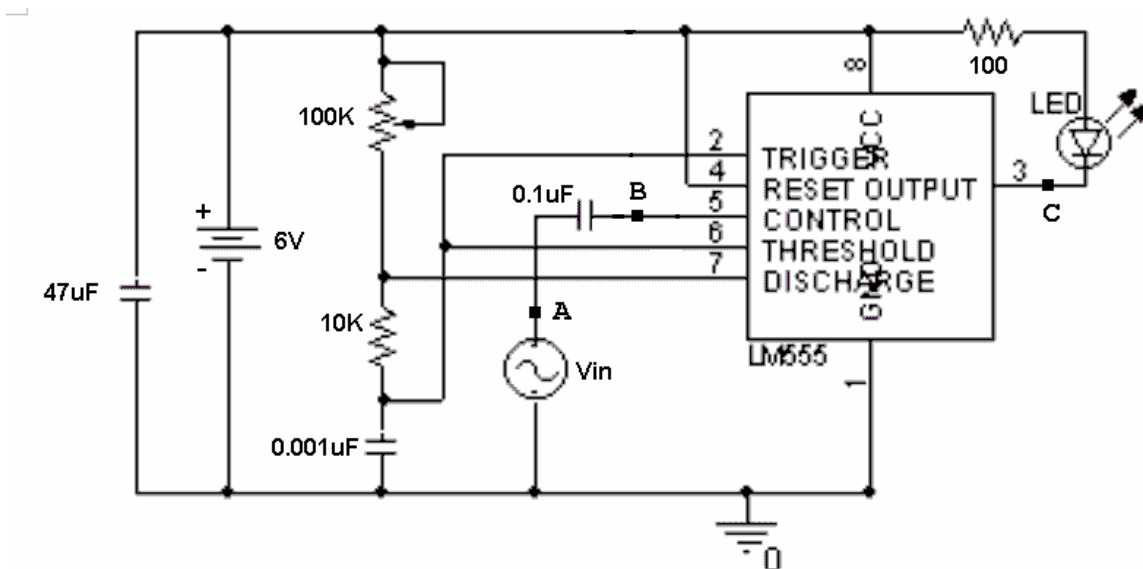The initial transmitter circuit is pictured in figure 3.

Figure 3

In this circuit, Vin is your input signal. In this project, you will examine the behavior of your circuit for two different types of input waves: a sine wave from the function generator and an audio source. Note the location of the three points: A, B and C. These points define the input and output to the blocks in the circuit. For example, the block between A and B is a DC blocking capacitor. It keeps the DC offset introduced by the 555 timer from interfering with the input signal (which has no DC offset). Block B-C is the 555-timer circuit that samples the signal. All of the measurements in this project should be taken with two channels of the scope and they should be measured with respect to ground. The 100K pot is marked 104.

Receiver Circuit
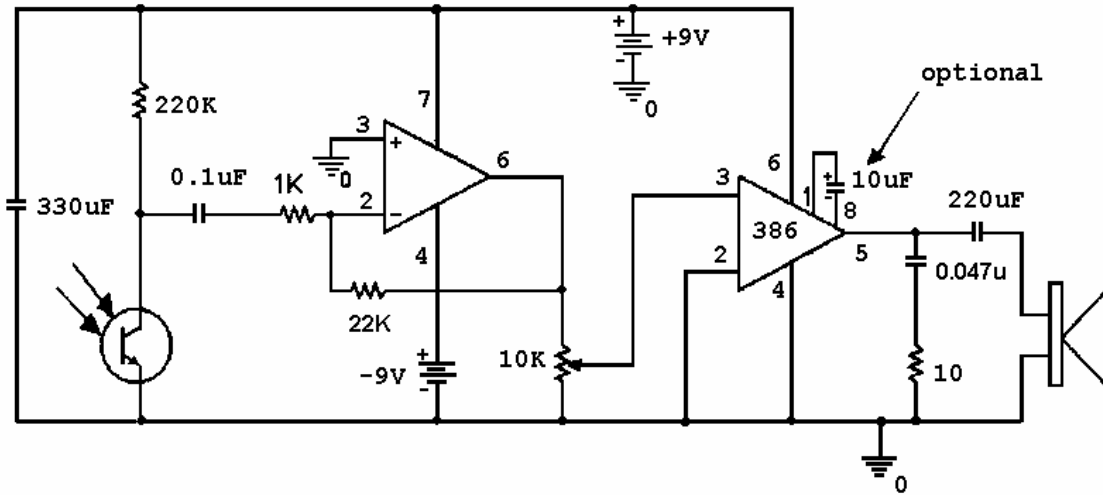The initial receiver circuit is shown in figure 4.

Figure 4

In this circuit, your input is the light pulses being detected by the phototransistor. The output is a demodulated and amplified signal played on a speaker. Again, note the locations of the points D, E, F, G and H. You will be measuring the voltage signals at these points and defining the circuit blocks between them. The 10K pot is marked 103.

The audio amplifier
Your circuit contains an audio amplifier. Details on this block of the circuit are contained in the spec sheet for the 386 amplifier. The image in figure 5 is taken from that source.
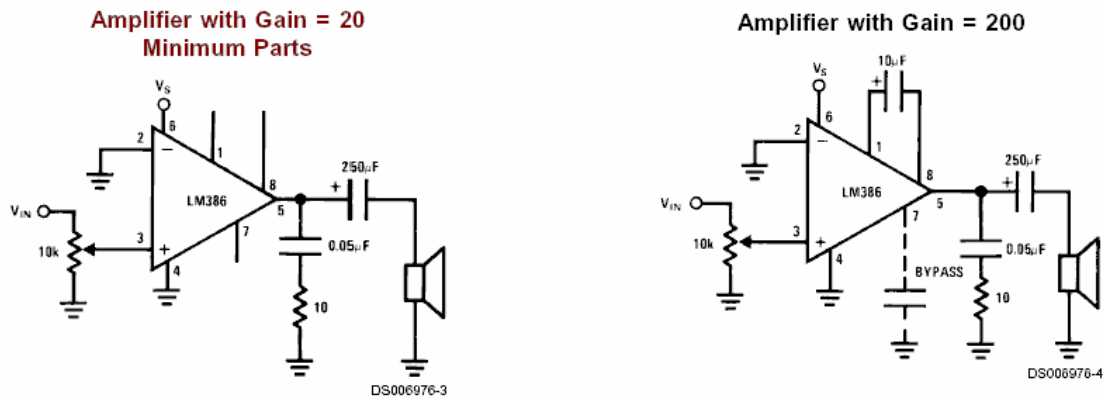

Figure 5

Note the addition of the capacitor to create a gain 10 times larger.

PSpice Model

We will give you a PSpice circuit that models the function of the initial transmitter and receiver circuit combination. It is available on the links page under project 4. Do not try to enter it by hand. You should generate PSpice output showing the function of each circuit block and also the function of the circuit as a whole.
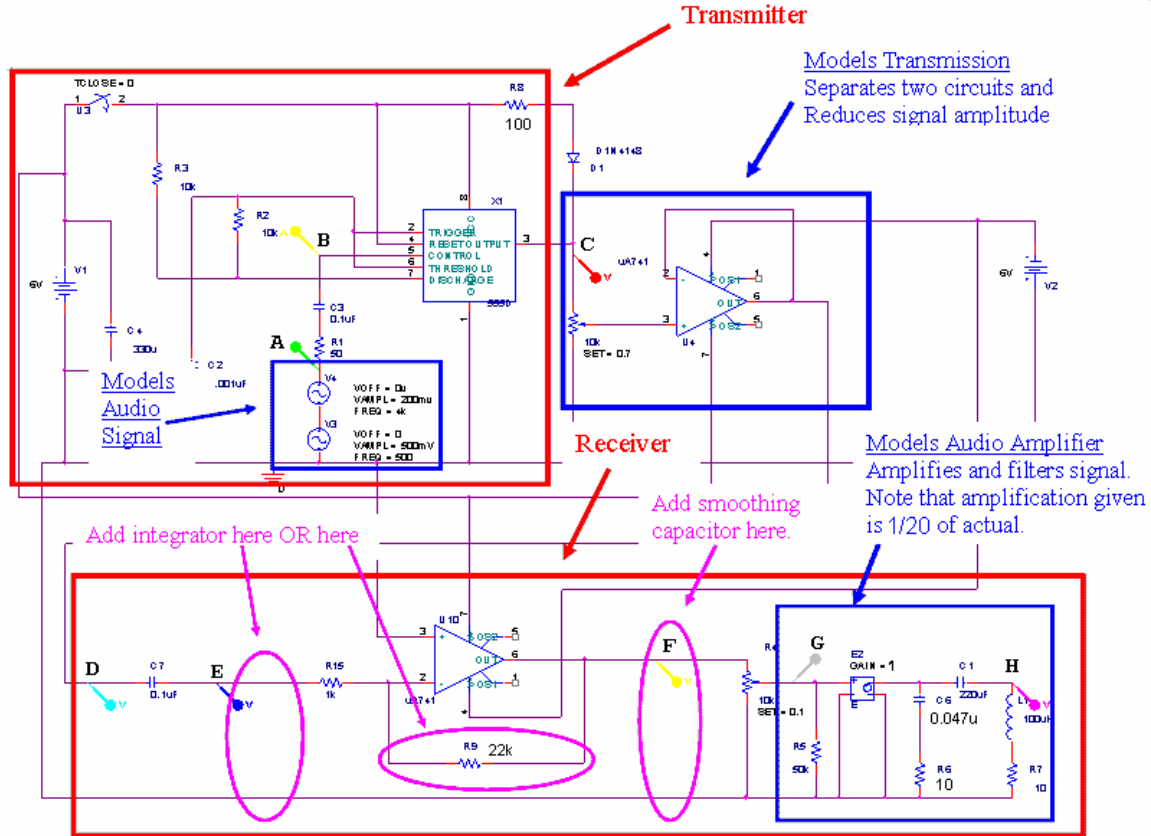

Figure 6

The model: Figure 6 shows the different pieces of the model. An audio source is approximated using two sine wave sources. The transmitter circuit is separated from the receiver circuit with a buffer. (In the real circuit that you build, this separation will be the air through which the light pulses are transmitted. In the simulation, we don't have light or air, but we can at least isolate the two circuits electronically with a buffer.) There is also a pot that creates the difference in amplitude between the pulses generated by the transmitter and the pulses picked up by the phototransistor. The audio amplifier in the receiver is modeled with a different amplifier (E) than the 386 used in the real circuit. (PSpice does not have a 386). In order to keep the amplitude of the output at the same order of magnitude as the input, the gain on this amplifier (1) is 1/20 of the gain on the 386 without the optional capacitor. The picture also shows the locations you will use to add the circuit elements you need for your final design.

Run the simulation: Start by running the simulation provided. Voltage markers have been added at each of the eight points (A-H). Remember that the initial design should give you a poor quality reproduction of your input signal. Therefore, you should expect the model of the initial

design to give you a poor quality reconstruction of the signal at the output. The input (point A) and output (point H) of the simulation should look something like figure 7. The input is one cycle of the lower frequency sine wave (at 500 Hz) with eight cycles of the higher frequency wave (at 4k Hz) superimposed on it. As you can see, the output shows some of the features of the input, but it consists mainly of the high frequency samples taken by the 555-timer oscillating between about minus 5V and plus 5V.
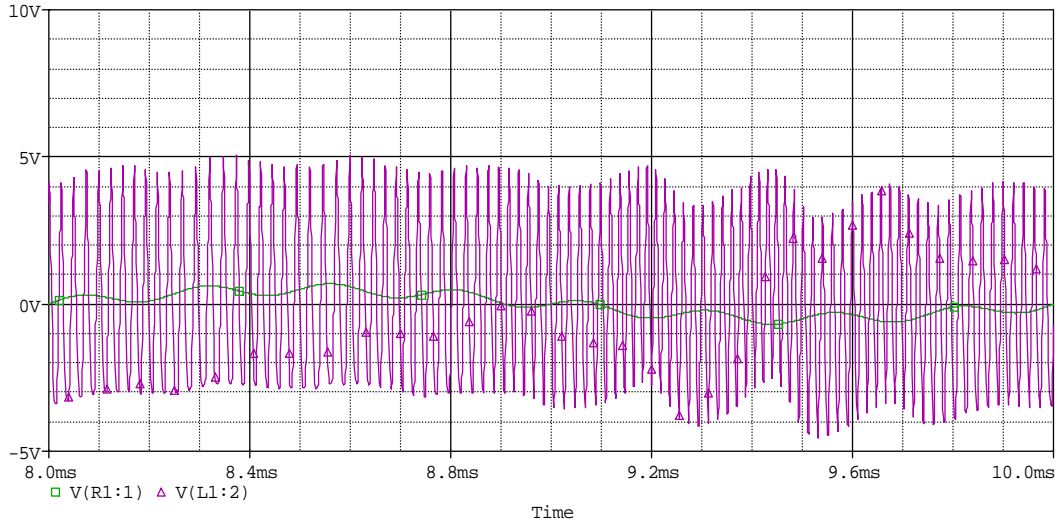

Figure 7 – "wide-angle" view

Figure 8 shows a closer view of the same signal. It was created by dividing the output wave by 10 and using the zoom-in feature of PSpice to get a close view. Note that the "wide-angle" view gives us the overall behavior of the wave and the "close-up" view lets us see the sampling frequency and the shape of the output pulses.

Figure 8 – "close-up" view [Note V(L1:2) has been divided by 10]

Determine your sampling frequency: The sampling frequency in the PSpice model is representative of a good sampling frequency for your circuit. It is hard to determine exactly what this frequency is by looking at the pulses because the frequency of the pulses changes with

the input signal. You can get a reasonable average by averaging over several samples in figure 8 or by calculating the frequency of the 555 timer circuit using the astable mode equations and the values of the components in the R-R-C combination (10k-10k-0.001u). Be sure that the PSpice frequency and the actual sampling frequency of your transmitter circuit match reasonably well when you take your data and design your integrator. You can match the frequencies either by turning the 100K pot in the actual circuit, replacing the 100K pot in the circuit with a 10k resistor, or by changing resistor R3 in the PSpice model.

Examine the different blocks: Once you have the simulation running, try looking at the input and output of each block separately. Also use the magnifying glass and look more closely at the wave shapes. (For example, look at the signals at point A and point B only. Which has the DC offset? Why?) This will give you an idea of what the block is doing and also what to expect from your circuit.

Building and Testing the Circuits
Build the circuit as shown in the circuit diagram. Leave out the optional 10μF capacitor to start. If the signal sounds very soft, try adding the 10μF capacitor. Turn up the volume. If the audio output sounds distorted, you should remove the capacitor. Test your circuit using a signal from the function generator representative of an audio wave. When the circuit works, have a staff member sign off on it and take your data.

test input signal: The transmitter is a 555-timer circuit very similar to an astable multivibrator. The only difference is that instead of generating a regular string of pulses, it generates a string of pulses which vary in frequency in response to the input signal at pin 5. It is best to use a sine wave to debug this circuit. A typical audio signal is pictured in figure 9. Note that the signal below has several frequencies. There are about 10 cycles of a typical wave in a 5 ms division. This corresponds to a frequency of about 2K hertz. Also note that the amplitude reaches a maximum of about 400mV. If we assume that this audio signal is typical of what we will get from our audio source, we should set the function generator to a frequency of 2K and an amplitude of 400mV.
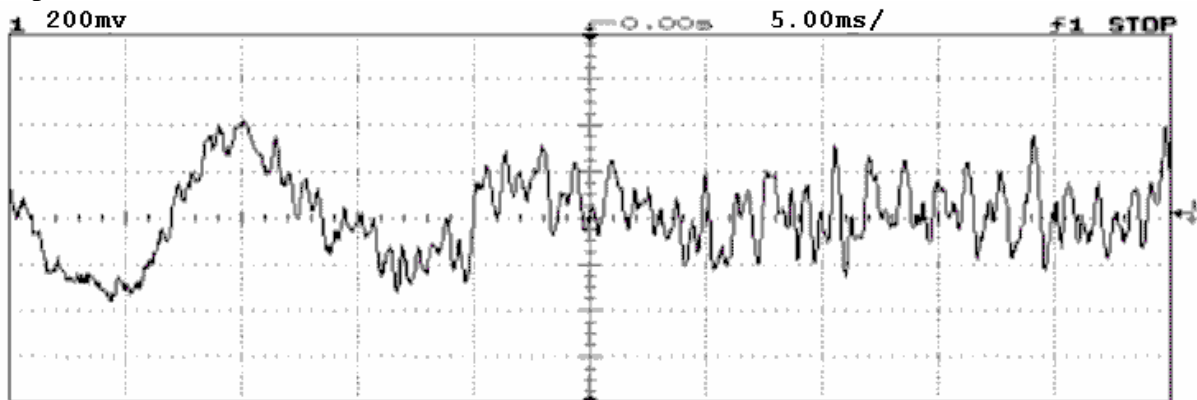

Figure 9

no common ground: The two circuits should NOT have a common ground. They should be completely separate and the transmission of the light pulses should be the only interaction.

orientation of the phototransistor: The phototransistor works in both orientations. In our experience with this circuit, it is best to use it in the orientation that generates signals of maximum
amplitude. Since we have different brands of phototransistors in circulation, we cannot tell you directly what orientation this is. The best thing for you to do is to test your phototransistor in both orientations and choose the one that gives you the strongest transmission signal.

verify that the transmitter is working: You should make sure that the pulses from your circuit are modulated by observing the voltage at the input (point A) and the voltage at the output (point C). The output should look like a pulse modulated signal. (If the input is low, the pulses should get closer together and if the input is high, the pulses should get farther apart.) As you turn the pot, the frequency of your input signal should change.

Set your sampling frequency: CD quality sound uses 44.1k samples per second to recreate an audio signal. Each pulse in your modulated signal is like a digital sample. A high sampling rate is desirable if you want to get high quality sound from your circuit. To vary your sampling rate, turn the 100K pot. To determine your sampling frequency, find the frequency of the pulses using the scope. You will not be able to get an exact measure of the sampling frequency of your circuit because it changes. However, you can get an estimate by averaging over several cycles. Since you will be comparing your data to the PSpice model (and using it to design your integrator), it is important that the sampling frequencies for your actual circuit and your PSpice circuit match fairly well. Either try to match your circuit to the PSpice (by using the pot or replacing it with a 10k resistor) or change the sampling rate of the PSpice circuit by altering the value of R3.

getting the receiver to work: The receiver circuit is more complicated than the transmitter circuit. If yours does not work, try debugging it in pieces. Check to see that the phototransistor is generating a set of pulses that corresponds to the pulses from the LED. Check to see that the inverting amplifier is making the pulses bigger. Make sure the volume pot alters the amplitude of the signal. Finally, check to see if the audio amplifier amplifies the signal again. If you identify a block that does not work, debug it before you continue on to the next one.

your audio signal: You will need to demonstrate that your circuit works using the function generator and a real audio signal. The computers in the studio have output jacks in the front. All the computers also have a copy of Winamp installed. You can bring your own music or find some on the internet. If you prefer, you can use your laptop or a portable audio device. Make sure that you use the scope to check the amplitude of your input signal. Set the volume of your signal so that it corresponds roughly to the 400mV amplitude of the test input. If you cannot turn the volume up that high, you may need to use another input device. *Do not let your input amplitude exceed 1V.* This will interfere with the 555 timer's ability to sample the input effectively.

signature: When your circuit works, have a staff member listen to the circuit with the sine wave input and with the audio input and sign the cover/signature page at the end of this handout.

<u>Taking your data</u>
Take data showing the input to and output from each block of the circuit and also the overall
function of the circuit. *All signals should be taken relative to ground.* The effect of the circuit on
the simple sine wave from the function generator is much easier to understand. Therefore, we
ask you to take most of your data using this signal. YOU MAY TAKE MORE THAN ONE
PICTURE OF EACH BLOCK. In some cases, it helps to take a "close-up" of the individual
pulses and a "wide-angle" of the overall signal shape. Whatever data you take, make sure that
the PSpice picture and the Agilent picture are at about the same time scale so that they can be
compared. *No signatures are required.*

take PSpice data: Before you take plots of the PSpice simulation, verify that the sampling
frequency is about the same as your circuit. If not, you can alter the sampling frequency by
changing R3 or turning the pot in your circuit. Remember that you want the sampling frequency
to be high. If you want in System (A-H), you can invert the output at H (multiply it by -1) to get
a better visual comparison with the input at A.

> **Block (A-B) : _____**   **Block(B-C): _____**   **Block(C-D):_____**
> **Block (D-E) : _____**   **Block(E-F): _____**
> **Block(F-G):_____**   **Block (G-H) : _____**   **System(A-H): _____**

take function generator data: The following is a list of the Agilent plots you should generate
using the function generator as input. For each block put one signal on channel 1 and the other
on channel 2. Be careful to keep track of which signal is which. If you want in System (A-H),
you can invert the output at H with the scope to get a better visual comparison with the input at
A. *Some of the speakers we have (with low impedance) distort the output at H. If you have this
problem, you can remove the speaker when you take the data at point H.*

> **Block (A-B) : _____**   **Block(B-C): _____**   **Block(C-D):_____**
> **Block (D-E) : _____**   **Block(E-F): _____**
> **Block(F-G):_____**   **Block (G-H) : _____**   **System(A-H): _____**

take audio data: Take the following additional data using your audio signal as input. If you want
in System(A-H), you can invert the output at H with the scope to get a better visual comparison
with the input at A. *Some of the speakers we have (with low impedance) distort the output at H.
If you have this problem, you can remove the speaker when you take the data at point H.*

> **Block(E-F): _____**   **System(A-H): _____**

<u>Comparison</u>
The final step in the initial design is to compare the output of the PSpice model to the
output of the actual circuit. Are they similar? Also examine the function of each circuit
block. Does each block work as expected?

Final Design

The initial design for the receiver of this project reconstructs the signal well enough to be audible, but it does a very poor job. You can improve the output of your circuit by adding an integrator and a smoothing capacitor. Your final signal at (point H) should look and sound much more like the original signal coming from your audio device at (point A).

Adding an integrator
The first change you will make to your circuit is to add an integrator.

What the integrator should do: When you integrate the modulated signal, you take advantage of the fact that the pulses vary in frequency. Your pulses (at point E) are square waves centered around zero. When you are on the positive part of the pulse, the signal (the integration of a positive constant) ramps up. When you are on the negative part of the pulse, the signal (the integration of a negative constant) ramps down. Since the pulses vary in frequency (and width) with the signal, adding this integration will bring out the variation in the amplitude of the original signal. Figure 10 shows the input (point A) and the output (point H) of the PSpice model with an appropriate integrator added. Note that you can still see the sampling pulses, but the output (which is inverted) captures the overall shape of the wave much better. You can see both the shape of the lower frequency and the higher frequency of the input.
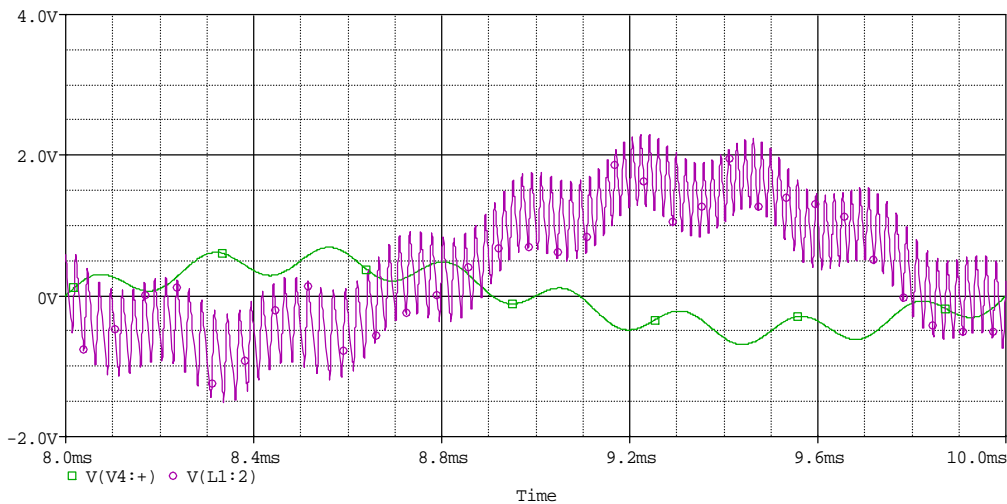

Figure 10

There are a few things you should note in this output. The first is that the signal is inverted because there is an inverting amplifier in the receiver circuit. Another is that there is a time shift between the input and the output. It takes a finite amount of time for the signal to travel through the circuit. The DC offset also wavers about a bit in the output. This will not affect the sound quality at the speaker.

Types of integrators: You can either add a passive or an active integrator to your circuit.
Both work well, so it is a matter of preference. Both are discussed in this section:

- Passive Integration
  Figure 11 shows a passive (approximate) integrator. It only works at higher
  frequencies. When $\omega \gg \omega_c = 1/RC$, then $V_{out} = \frac{1}{RC} \int V_{in} dt$. If you add this
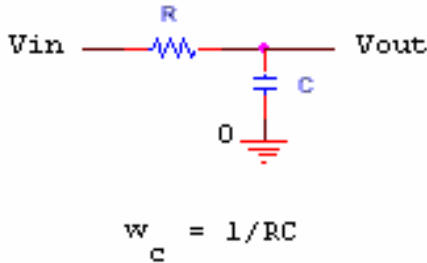 integrator, you will add it at point E.

Figure 11

  When you add this integrator, you may notice a marked change in the amplitude
  of your signal. You can change the amplitude by a factor of 10 by adding or
  removing the 10µF capacitor in the audio amplifier. You can make finer
  adjustments by changing the gain of the inverting amplifier.

- Active Integration
  Figure 12 shows an active integrator. To add this type of integrator, you can
  modify the inverting amplifier in your circuit by adding a capacitor to the
  feedback path. This integrator also does not work well at very low frequencies.
  When $\omega \gg \omega_c = 1/(R_f C)$, then $V_{out} = \frac{-1}{R_i C} \int V_{in} dt$.
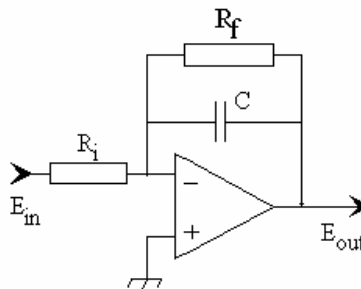

Figure 12

  When you add this integrator, you may notice a marked change in the amplitude
  of your signal. You can change the amplitude by a factor of 10 by adding or
  removing the 10µF capacitor in the audio amplifier. You can make finer
  adjustments by changing the gain of the integrator. The gain of the integrator can

be adjusted without changing the frequencies that are being integrated by changing the value of R_i.

Designing your integrator: You want your integrator to work on an audio signal. In the PSpice model, the sample input frequency varies between 500 Hz and 4k Hz. This range should be adequate. If you want to, you can look closely at the audio signal you are using and determine its lowest and highest frequencies. If they fall outside this range, change the frequencies of the two sine wave sources. You need to choose a frequency for your integrator much lower than 500 Hz that will still integrate 4k Hz AND preserve enough of the amplitude so that you still have a signal. With either integrator, you need to choose R and C such that your chosen frequency is much less than the frequencies you want to integrate, $f \ll 1/(2\pi RC)$. In the passive integrator, you can choose R arbitrarily, in the active integrator, R is the value of your feedback resistor.

Testing your choice with PSpice: Once you have chosen an integrator and your target component values, try adding the integrator to the PSpice model. Run the simulation and see if it works. Note that you will probably have to increase the of E2 by a factor of 10 (to 10). If you have to do this in PSpice, do not forget to also add the 10µF capacitor between pins 1 and 8 of the 386 amplifier in your protoboard circuit. Try adjusting the value of the capacitor until you get a good integration with adequate final amplitude. Remember, you don't need to worry about the pulses from the 555-timer sampling. You just want to get a good reconstruction of the input with the sampling pulses superimposed on it as shown in figure 10. [If you need to, you can use the inverting amplifier to change the gain as well.]

Adding the integrator to your circuit: After you have used PSpice to design an integrator, add the integrator to your circuit. Don't forget to also add any other changes you made in the PSpice model (like amplifying the signal). Does your integrator work? Many times the best design predicted by PSpice will not be the one that works optimally in your circuit. If you find a design that works better in the actual circuit, go back and adjust the PSpice.

signature: When your circuit works as expected with the integrator, have a staff member listen to the circuit with the sine wave input and sign the cover/signature page at the end of this handout

Take your data: When you have a working integrator, take a picture of the integrator functioning and the overall performance of the circuit and the model. If you want in System(A-H), you can invert the output at H with the scope and with PSpice to get a better visual comparison to the input at A. *Some of the speakers we have (with low impedance) distort the output at H. If you have this problem, you can remove the speaker and when you take the data at point H.*

PSpice output: **Block(E-F):** _____   **System(A-H):** _____
Circuit with sine wave input: **Block(E-F):** _____   **System(A-H):** _____

Smoothing out the sampling frequency
The final addition you will make to your circuit is to add a smoothing capacitor to
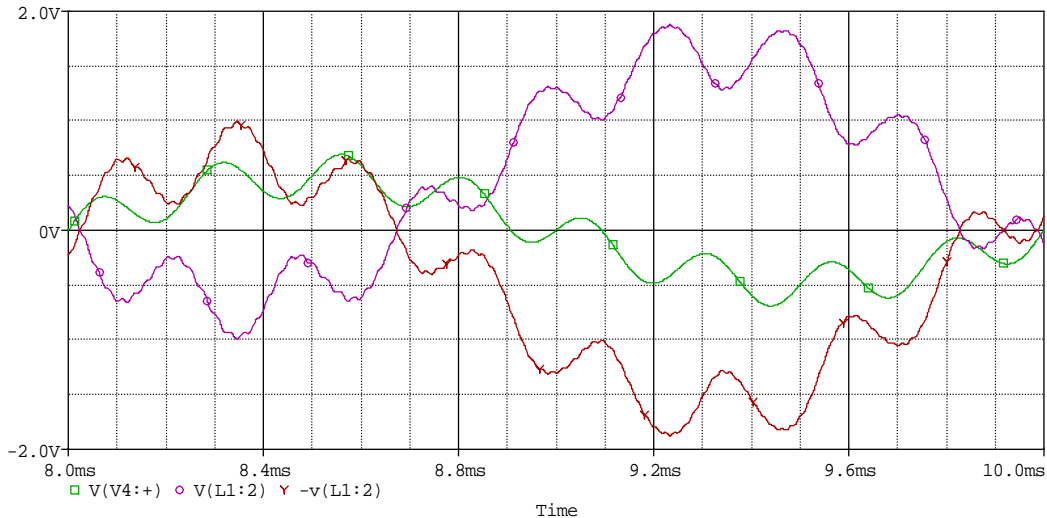eliminate the pulses created by the sampling.

Figure 13

What the smoothing capacitor should do: Recall that when you added a smoothing
capacitor to the output of a rectifier, the bumps from the rectified signal were smoothed
over and the output was much more even. You can apply this concept to the signal in this
circuit as well. If we add a smoothing capacitor of the correct size between point F and
ground, we should be able to smooth out the pulses caused by the sampling and come up
with an even closer reconstruction of the original audio wave. Figure 13 shows the result
of adding a smoothing capacitor to the integrated signal. It shows the original input (at
point A), the output of the circuit (at point H), and the inversion of the output (at point
H). Note that the original input frequencies have been preserved, but the pulses (that are
shown in figure 10) have been largely eliminated. Aside from a change in amplitude, a
slight time delay, and some inconsistencies in DC offset (caused by the integration), the
input and the output are now very much the same. You can see that this is a substantial
improvement over the initial data shown in figure 7.

What capacitance should I use? The exact value of your smoothing capacitor will
depend on your integrator and your sampling frequency. A capacitor that is too large,
will smooth over the high frequencies of your input signal. A capacitor that is too small
will not smooth over the sampling pulses. Try experimenting with the PSpice model by
adding a 1μF capacitor between point F and ground. Look at the output. Should the
capacitance be increased or decreased? Increase or decrease the capacitance in
increments of an order of magnitude (0.001μ, 0.01μ, 0.1μ, 1μ, 10μ, 100μ, 1000μ) until
you find one that gives you the best reconstruction. You can then fine tune the
capacitance using the values you have in your kit. If you find you need a capacitance that
you do not have, consult the staff to find out what we have available.

Check the smoothing capacitor in your circuit: We have found that the smoothing capacitors do not always work well in the actual circuit. Try adding the smoothing capacitor to your circuit and see if it works. You may need to try a slightly different size capacitor than the one predicted by PSpice. If you can find a design that both integrates and smoothes the signal, take a picture of the output for a sine wave input signal and have it signed by a staff member. Include it for extra credit. Make sure you indicate what components you used to get the circuit working.

Take your PSpice data: Once you have decided on a smoothing capacitor that works well in PSpice, take your data. If you want in System(A-H), you can invert the output at H with the scope and with PSpice to get a better visual comparison to the input at A. *Some of the speakers we have (with low impedance) distort the output at H. If you have this problem, you can remove the speaker and when you take the data at point H.*

PSpice output: **Block(E-F): _____   System(A-H): _____**

Take your circuit data:
If you were able to get the smoothing capacitor to work, take this last picture with both the integrator and the smoothing capacitor. If you did not, take it with only the integrator.

Final circuit with audio source input: **Block(E-F): _____   System(A-H): _____**

signature: When your circuit works as expected with the integrator (and possibly the smoothing capacitor), have a staff member listen to the circuit with the audio input and sign the cover/signature page at the end of this handout

The Report

Instead of a formal report this semester, we are asking you to provide the following packet. Basically, we want you to show a caparison between the PSpice output and the output from the circuit. You will be graded out of 70 points for the packet plus an additional 10 point general assessment. If you want you can include more than one pair of plots for any block. Also Include the cover/signature page attached to the end of this handout. ALL PLOTS SHOULD INDICATE WHICH TRACE CORRESPONDS TO THE SIGNAL AT WHICH POINT.

A. Plots for Initial Design with Function Generator Input (40 points)
  Part A1: Block A-B
        Output from PSpice (1)
        Output from circuit (1)
        Block description and brief comparison (3)
  Part A2: Block B-C
        Output from PSpice (1)
        Output from circuit (1)

Block description and brief comparison (3)
  Part A3: Block C-D
      Output from PSpice (1)
      Output from circuit (1)
      Block description and brief comparison (3)
  Part A4: Block D-E
      Output from PSpice (1)
      Output from circuit (1)
      Block description and brief comparison (3)
  Part A5: Block E-F
      Output from PSpice (1)
      Output from Circuit (1)
      Block description and brief comparison (3)
  Part A6: Block F-G
      Output from PSpice (1)
      Output from Circuit (1)
      Block description and brief comparison (3)
  Part A7: Block G-H
      Output from PSpice (1)
      Output from Circuit (1)
      Block description and brief comparison (3)
  Part A8: Circuit Function A-H
      Output from PSpice (1)
      Output from Circuit (1)
      Block description and brief comparison (3)

B. Plots for Initial Design with Audio Input (4 points)
  Part B1: Block E-F
      Output from Circuit (2)
  Part B2: Circuit Function A-H
      Output from Circuit (2)

C. Final Design with Integrator only (12 points)
  Part C1: Block E-F – Function Generator Input
      Output from PSpice (2)
      Output from Circuit (2)
      Brief comparison (2)
  Part C2:   Circuit Function A-H – Function Generator Input
      Output from PSpice (2)
      Output from Circuit (2)
      Brief comparison (2)

D. Final Design with Integrator and Smoothing Capacitor (14 points)
  Part D1: Final Design Schematic
      Print out of final design schematic from PSpice (3)
      Calculation of corner frequency of integrator (2)

State value of soothing capacitor (1)
Part D2: Block E-F – Function Generator Input
Output from PSpice (2)
Compare to output without smoothing (1)
Part D3: Circuit Function A-H – Function Generator Input
Output from PSpice (2)
Compare to output without smoothing (1)
Part D4: Circuit Function A-H –Audio Input
Output from Circuit (2)

Total: 70 points for project packet
+ 10 points for general assessment of report
+ 20 points for attendance
= 100 points

Attendance (20 possible points)
3 classes (20 points)   2 classes (10 points)   1 class (0 points)
Minus 5 points for each late.

No attendance at all = No grade for project.

You should expect the simulation and the actual data to be different because the input
waves are different. The simulation input is two superimposed sinusoids, whereas, the
circuit input is a single sinusoid. So the shape of the output will vary somewhat, but the
way the block acts on the output should be the same. The amplitudes of both the
simulated input and the actual input should be in the same approximate ballpark (unless
we deliberately altered the gain to make the signal visible). The sampling frequency
should be close. If there are significant differences, please note it with a reasonable
explanation. Here is an example of an appropriate "block description" and "brief
comparison"

*Block Description Block F-G is a volume control. As you can see, in both the
PSpice simulation and the actual circuit output, the output signal (the trace
marked G) is smaller than the input signal (the trace marked F). The amplitude
of the output at G is some fraction (between 0 and 1) of the input at F. Turning
the pot changes the relative amplitude of G. The output and the input are the
same basic shape and the frequencies are preserved exactly.*

*Brief Comparison The PSpice output and the output from the circuit have
comparable sampling frequency (around 48kHz). The amplitude of the PSpice
output is larger, primarily due to the fact that the output signal from the real
photodiode is smaller than the one from the simulation. The overall shape of both
(triangular samples superimposed over a signal which marginally represents the
input) are comparable given the difference in the input.*

Project 4
Electronic Instrumentation

Group _____

Report Grade: _____

_____ **Attendance:**

_____ **Attendance:**

_____ **Attendance:**

_____ **Attendance:**

<u>Signatures</u>

1. Initial Design

Function Generator Input _____

Audio Input _____

2. Final Design with Integrator Only

Function Generator Input _____

3. Final Design with Integrator (and Smoothing Capacitor)

Audio Input _____