Using Fourier Analysis in Introductory Physics

The MacScope II Instruction Manual and five articles on using MacScope and Fourier analysis in an introductory physics course.
The MacScope II program is an audio software oscilloscope for both Mac and Windows which uses the computer’s sound input to acquire data. The main features are Fourier analysis, signal averaging and working with triggered curves.
Using Fourier Analysis in Introductory Physics

This book is designed to show you how to conveniently and effectively use Fourier Analysis in an introductory physics course.

For convenience we introduce the MacScope II computer program that turns any USB equipped computer (both Macintosh and Windows) into a laboratory oscilloscope that can both acquire and analyze experimental data. The second half of this book is a detailed instruction manual for MacScope II. Part of the convenience is that we have made MacScope II a shareware program where the latest version can be freely downloaded from the Physics2000.com website and used anywhere. (Eg., put it on your course website).

For effectiveness, we present five articles showing how to use MacScope both as a laboratory instrument and as a theoretical tool. In the first article we show how to use MacScope and inexpensive equipment (the $40 iMic™ and an $8 Wal-Mart microphone) to record the sound of a note played on a grand piano. We looked at the piano and discovered that all the hammers struck the piano strings at a node of the seventh harmonic of the vibration of the strings. A Fourier analysis of the sound produced by a string showed that the seventh harmonic was missing. This is our first example of the use of Fourier analysis as a mathematical tool for studying physical phenomena. (In an appendix, this article provides a non calculus derivation of the formulas for calculating Fourier coefficients.)

In the second article, we introduce what we call the pulse Fourier transform to provide an explanation of the time-energy form of the uncertainty principle. The pulse Fourier transform allows us to model a short pulse like those produced by femto-second lasers. Experimentally a short laser pulse contains a broad spectrum of frequencies. We can see why when we take the Fourier transform of a short pulse. When you include Max Born’s probability interpretation of particle waves, the uncertainty principle becomes an immediate consequence of the analysis.

In the third paper we use MacScope to introduce Fourier optics by showing that the Fourier transform of a slit structure has the same shape as the intensity the diffraction pattern produced by a laser beam passing through the slits. We used MacScope’s math waves feature to create the slit structures. Then we used a Huygens analysis of the diffraction pattern to see why the diffraction pattern and the Fourier transform were similar. Our main conclusion is that the Huygens construction is essentially a graphical way to do Fourier analysis.

The last two papers demonstrate how MacScope can be used as a dual beam storage oscilloscope. In the fourth paper, we tape inexpensive microphones to both ends of a 10 ft. long steel pipe, hit one end of the pipe with a hammer, and measure how long it takes the sound pulse to travel down the pipe. The answer is 0.60 milliseconds for a speed of 5,080 meters per second.

The final paper is a study of the speed of both transverse and compressional waves in a Hook’s law media. Our two examples are a Slinky™ where the wave speed is of the order of one meter per second, and a steel guitar string where the compressional wave speed is 4,960 meters per second. We used a television camera to measure the Slinky wave speed, and MacScope to measure the speeds of transverse and compressional waves in the guitar string. The paper introduces a non calculus derivation of the speed of compressional waves in Hooke’s law media.

E. R. Huggins
Dartmouth College
I) Teaching Fourier Analysis in Introductory Physics

It is difficult to imagine an introductory physics course where light and color are studied without the aid of a prism or diffraction grating—a course where we do not observe the spectrum of light. Yet we often discuss sound without introducing a similar device to analyze the spectrum of complex sound waves. That device is Fourier analysis which, for example, allows us to understand why different musical instruments sound differently even when playing the same note.

Two reasons are usually given for not introducing Fourier analysis. One is that the equipment required to do the job is much harder to obtain and deal with than a prism and a diffraction grating. The second is that the mathematics behind the Fourier analysis is thought to be beyond the mathematical capabilities of students in introductory physics courses.

In this paper we introduce a solution to both problems. We are introducing an inexpensive software program that analyzes experimental data brought into the computer (Mac or Windows) through the computer’s sound input. Then the mathematics behind Fourier analysis is presented in a non-calculus approach that was developed for high school students.

Middle C Project

We will illustrate the use of the computer program with a project in which we acquire and analyze the sound of the middle C note played on a piano. We will see that the vibrational modes of an oscillating piano wire provide a graphic introduction to Fourier analysis. The sounds produced by the modes can be detected individually as the Fourier components in the analysis of the sound.

The Hardware

The equipment we used in the project is shown in Fig. 1. We have an under $10 microphone obtainable at stores like Wal-Mart, the $40 Griffin iMic which converts the microphone voltage output to a USB output, and a laptop computer. Some computers have a built-in microphone input and thus do not need an iMic. The computer shown has a built-in microphone, and thus does not require any external hardware. We chose the hardware shown because it works with any USB enabled computer we have encountered, and because it allows us to cut down on the background noise by placing the microphone just above the piano wires being studied, as shown in Fig. 2.
MacScope II

The software for this project is MacScope® II, a software audio oscilloscope written to run on both Mac and Windows\(^1\). The original MacScope, which we designed in the 1980s, worked only on Macs, and required a $2,000 external interface box. With MacScope II, we have considerably reduced external hardware costs, down to as low as $0.

Because MacScope II uses the computer’s audio input to grab experimental data, the frequency range is limited to the audio frequency range. Explicitly, computers and the iMic have an AC coupled input that has a low frequency cutoff around 10 Hz, and both digitize the voltage input at a rate of 44,000 points per second. Since the maximum frequency that can be detected requires at least 2 points per cycle, the upper frequency for these audio inputs is 22,000 Hz, which is called the Nyquist frequency. If you try to look for frequencies higher than that in the data, you get spurious results called aliasing.

We designed MacScope II to be as easy to use as possible. Once you have selected which audio input you intend to use, you press the Record button and you get curves like those shown in Fig. 3. The curves are effectively triggered, meaning that repetitive signals like vowel sounds produce stable curves. We get two curves from the standard computer stereo input. If we send the same signal into both sides of the stereo input, we can simultaneously observe the same signal on two different time scales, a feature not available in previous generation oscilloscopes.

MacScope II has a number of advanced features that are not needed for the middle C project. For example, the scope can do live signal averaging of triggered curves, a feature now being used in a neurobiology course to study action potentials. For our project we will leave the advanced controls hidden.

Once you have recorded several curves, you can display curve A and curve B data from different files for direct comparison. In Fig. 3, the curve B data was taken just after we played the note, whereas we waited a while to record the sound for curve A. The fact that the two curves are different in shape tells us that there is no such thing as a unique shape for a piano middle C sound curve.

**Frequency Measurement**

What we can tell from the shape of the middle C curves is that they represent repetitive waves whose period can be directly measured. In curve A we are selecting one cycle of the wave. As we do the selection, a window (on the right) pops up telling us the period and frequency of our selection. In this case the period \( t \) is 3.9 milliseconds, and the corresponding frequency \( 1/t \) is 256.7 Hz. Because middle C should have a frequency of 262 Hz, this tells us that the piano needs tuning.
Modes of oscillation

While we can use the shape of the middle C curves to determine their frequency, we do not learn much else by simply looking at the curves. Further analysis requires studying the spectrum of frequencies contained in these waves. To do this we use Fourier analysis.

To introduce Fourier analysis, it is instructive to look at the ways a guitar string or a piano wire can oscillate. In Fig. 4 we have sketched the allowed modes of oscillation, which are the sinusoidal modes that have nodes at the endpoints. The first harmonic has a wavelength $\lambda_1 = 2L$ equal to twice the length $L$ of the string or wire. The other modes have wavelengths $\lambda_n = 2L/n$ as shown.

In the piano, the sound is produced by a felt hammer striking the piano wires. For a number of notes across the keyboard, we measured the distance from the striking point of the hammer to the end of the wires. To our surprise, on our piano anyway, the striking point was always 14% of the way from one end of the wire. As seen in the diagram, that means that the hammer is striking at a node for the seventh harmonic.

When you strike a stretched string or wire, you tend to excite various modes into oscillation, and the sound you hear is the sum of the sound waves produced by each mode. For example, if you strum a guitar at the center of the strings, you will excite mostly the first harmonic and the sound will be smooth and plain. Strumming down closer to the bridge excites more of the shorter wavelength, higher frequency harmonics, and you get a richer, sharper sound. What you do not excite are harmonics that have a node at the point you are strumming. Looking at Fig. 4, we suspect that the hammer striking 14% from the end will not excite much of the seventh harmonic.

Fig. 4. Modes of oscillation of a guitar string or piano wire. On our piano, we found that the hammer was always 14% away from one end.

Fig. 5. Fourier analysis window for Curve A. When we click on the button labeled Fourier we get an analysis of the selected section of the curve. Here we see that the seventh harmonic is missing.
Fourier Analysis

How do we tell what harmonic frequencies are contained in the sound wave we hear? How can we tell whether the seventh harmonic mode was excited or not? With a beam of light, we can tell if a particular frequency is present by using a prism or diffraction grating. With sound, we use Fourier analysis.

A Fourier analysis capability is built into MacScope; in fact, that is the main reason why we wrote the program. In Fig. 3, we have already selected one cycle of the sound wave seen in curve A. When we press the button labeled Fourier just above the curve, the selected section of curve A expands to fill its window, and below we see the curve A Fourier analysis window as shown in Fig. 5.

The vertical bars indicate the relative amplitude of the harmonic frequencies contained in the curve A sound wave. The fourth harmonic has the largest amplitude, and harmonics one, two and five are relatively large. An important feature is that the seventh harmonic is missing, as we predicted from an analysis of the way the piano wires were struck.

In Fig. 6, we look at each of the four large harmonics individually. When we click on one of the vertical bars, that harmonic is drawn over the selected curve. In Fig. 7 we are reconstructing the curve from selected harmonics. You can see that the more harmonics we select, the closer we come to reproducing the original curve. If you add up all the harmonics, you get the original curve.

Fig. 6. The big harmonics in Curve A, middle C. The way we selected the cycle, the harmonics look like cosine waves.

Fig. 7. Reconstructing the curve from the big harmonics. The more harmonics we select, the closer we get.
The Mathematics of Fourier Analysis

How does MacScope’s Fourier analysis program figure out how much of each harmonic is present in the sound wave? How to do this is easy to understand, but tedious to carry out, unless you have a computer.

The mathematical statement of the problem can be expressed as follows. We have one cycle of a wave that has some arbitrary shape we will call \( F(t) \), a function \( F \) of time \( t \). What we want to do is to build up this function \( F(t) \) out of sine and cosine waves. We can do this with the formula

\[
F(t) = A_1 \cos(1t) + A_2 \cos(2t) + A_3 \cos(3t) + \ldots + B_1 \sin(1t) + B_2 \sin(2t) + B_3 \sin(3t) + \ldots
\]  

(1)

In Eq. 1, the coefficients \( A_1, A_2, B_1, B_2, \ldots \), represent how much of each sine or cosine wave is contained in our function \( F(t) \). From the pictures in Fig. 6 we saw that for the big harmonics we had almost pure cosine waves, meaning that the corresponding sine wave coefficients \( B_1, B_2, B_4, \) and \( B_5 \) are nearly zero. These \( A \)'s and \( B \)'s are known as Fourier coefficients.

How do we calculate the Fourier coefficients? We will illustrate how by calculating the coefficient \( B_3 \). We begin by making a mess of Eq. 1, by multiplying both sides by \( \sin(3t) \), to get

\[
F(t) \sin(3t) = A_1 \cos(1t) \sin(3t) + A_2 \cos(2t) \sin(3t) + A_3 \cos(3t) \sin(3t) + \ldots + B_1 \sin(1t) \sin(3t) + B_2 \sin(2t) \sin(3t) + B_3 \sin(3t) \sin(3t) + \ldots
\]  

(2)

Equation 2 is not as bad as it first looks, if we realize that most of the terms can be expressed as pictures. For example, the product \( \sin(2t) \sin(3t) \) in the range \( t = 0 \) to \( 2\pi \) is

\[
\sin(2t) \sin(3t) = \begin{array}{c}
\text{area above zero line}
\end{array}
\]  

\[
\text{area below zero line}
\]

\[
\text{zero net area}
\]

(3)

Our next step is to replace all the products of sines and cosines in Eq. 2 by their corresponding pictures. (The pictures were obtained from a simple TrueBASIC® program.) We get

\[
F(t) \sin(3t) = A_1 \times + A_2 \times + A_3 \times + \ldots + B_1 \times + B_2 \times + B_3 \times + \ldots
\]  

(4)

Looking through the pictures in Eq. 4, we see that all but one term has as much area above as below the zero line, i.e., has zero net area. The exception is that \( \sin(3t) \sin(3t) \), being a square, has to have a positive area.

If we take the area under both sides of the equation all terms on the right except the \( \sin(3t) \sin(3t) \) term will vanish. This is the trick that allows us to calculate the Fourier coefficients.
As we mentioned, on the right side of Eq. 4, all but one term has zero net area. If we take the area under both sides of the equation, most of the terms vanish, and we are left with

\[
\text{Area Under}[F(t) \sin(3t)] = B_3 \times \text{Area Under}
\]

\[
= B_3 \times \int_0^{2\pi} [F(t) \sin(3t)]dt
\]

\[
\text{(5)}
\]

It turns out to be quite easy to determine the area under the picture in Eq. 5. Draw a line that cuts the curve at a height .5, as shown in Fig. 8.

![Fig. 8. Crests above the height .5 just fill in the troughs below .5 to give us a rectangle of area \(\pi\).](image)

We see that the bumps above the .5 line would just fit into the gaps below the .5 line, giving us a solid rectangle .5 high and \(2\pi\) wide. The area of this rectangle is thus \(\pi\) and we get

\[
\text{Area Under}[\sin(3t) \sin(3t)] = \pi
\]

\[
\text{(6)}
\]

Using Eq. (6) in Eq. (5) and solving for \(B_3\) gives

\[
B_3 = \frac{1}{\pi} \times \text{Area Under}[F(t) \sin(3t)]
\]

\[
\text{(7)}
\]

For those with some calculus background, this equation can be written

\[
B_3 = \frac{1}{\pi} \times \int_0^{2\pi} [F(t) \sin(3t)]dt
\]

\[
\text{(7a)}
\]

**Amplitude, Phase, and Intensity**

In Eq. 1, we are attempting to reproduce our curve \(F(t)\) as the sum of sine and cosine waves. An alternate formula for \(F(t)\) is

\[
F(t) = C_1 \cos(1t - \phi_1) + C_2 \cos(2t - \phi_2) + \ldots
\]

\[
\text{(8)}
\]

where the \(\phi\)'s are phase angles that tell us how far down the \(t\) axis the cosine wave starts, as shown in Fig. 9. The coefficients \(C_n\) are related to the coefficients \(A_n\) and \(B_n\) of Eq. 1 by the trigonometric identity

\[
A \cos(t) + B \sin(t) = C \cos(t - \phi)
\]

\[
\text{(9)}
\]

where

\[
C^2 = A^2 + B^2
\]

\[
\tan(\phi) = B/A.
\]

The MacScope program gives you the option of plotting the amplitudes, which are the \(C_n\), the phases, which are the \(\phi_n\), or the intensities, which are \(C_n^2\). When you use a prism or diffraction grating to separate light beams into the individual colors or wavelengths, what you see in the relative brightness of the individual colors is the intensity of that component of the wave. The intensity is related to the amount of energy in that wavelength component which is proportional to the square of the amplitude of the wave. Thus the plot of intensities \(C_n^2\) gives you a picture of the relative amounts of energy contained in each Fourier component.
Why We Selected One Cycle

To get meaningful results in our Fourier analysis of the middle C sound wave, we were careful to select one cycle of a repeating wave. The reason is that, according to Eq. 1, we are attempting to reconstruct the wave from terms like \( A_1 \cos(1t) \) and \( B_3 \sin(3t) \), all of which are exactly repetitive in the range 0 to \( 2\pi \). In our analysis, we are in effect assuming that the selected section of wave exactly repeats, time after time.

Figure 10 shows us what happens if we do not select a repeating section of a wave. In 10a we selected half a cycle of a sine wave and got the mess of harmonics seen in 10b. By reconstructing the curve from the first ten harmonics in 10c, we see that the reconstructed curve is forced to be repetitive by forcing the ends to have the same height. When we select a repetitive section of the curve, the ends are already at the same height, and no spurious harmonics are introduced.

When you select a section of curve for Fourier analysis, think of the selected section as repeating indefinitely. For example, when we view the selected half cycle repeating, as shown in Fig. 11, we see that we are trying to reconstruct a discontinuous curve. The extra harmonics in Fig. 10 are trying to approximate the discontinuity.

It is traditional to introduce Fourier analysis by discussing the analysis of a square wave. The problem is that a square wave is, by definition, a discontinuous function. As you add harmonics to reconstruct a square wave, you see ripples at the edges of the square, ripples that get finer but do not change in overall height as you add in more harmonics. These ripples are known as the Gibbs effect. By our not selecting a repetitive section of the curve in Figure 10, we have introduced a discontinuity, and in Fig. 10c are beginning to see the Gibbs effect.

Notes

1. MacScope II will be released in late spring of 2005 on the $10 CD Physics2000 at Physics2000.com

2. Derivation of Equation 9. Start with
   \[
   \cos(x-y) = \cos(x) \cos(y) + \sin(x) \sin(y)
   \]
   Let \( x = t \), \( y = \phi \) and multiply through by \( C \) to get
   \[
   C\cos(t-\phi) = [C\cos(\phi)]\cos(t) + [C\sin(\phi)]\sin(t)
   \]
   This is Eq. 8 if we set
   \[
   A = [C\cos(\phi)] ; \quad B = [C\sin(\phi)]
   \]
   Thus \( \tan(\phi) = \sin(\phi)/\cos(\phi) = B/A. \)
Teaching Fourier Analysis
II) Teaching the Uncertainty Principle In Introductory Physics

Eliminating the artificial divide between classical and modern physics in introductory physics courses has long been our goal. To do this, we have been looking at the modern physics curriculum to see where topics could appropriately appear earlier in the course. For over 30 years, we have known that special relativity is best taught at the very beginning of the course where the focus can be on physical concepts rather than mathematical formalism.¹ For mathematics, only the Pythagorean theorem is needed to teach time dilation, the Lorentz contraction and the lack of simultaneity. The physics background needed is a ride in a jet plane to have the experience that you do not feel uniform motion.

In contrast, the uncertainty principle appears to require an extensive background in both physics and mathematics. In most introductory physics texts, the uncertainty principle, if discussed at all, appears in the last few pages of the text. The energy time form \( \Delta E \Delta t > h \) gets the worst treatment. If it is not simply stated as a fact, it is derived only for non-relativistic particles, from the momentum-position form \( \Delta p \Delta x > h \).

The purpose of this paper is to demonstrate that the energy time form of the uncertainty principle can be taught in a comprehensive manner with little mathematics and a limited physics background. What mathematics is needed is handled by the Fourier pulse transform capabilities of the MacScope II² computer program. The physics background needed is the particle-wave nature of matter and the probability interpretation of the particle wave.

If you began the course with special relativity, you have no difficulty going directly from a discussion of light waves to the photoelectric effect and a discussion of light particles. The probability interpretation of particle waves is beautifully illustrated by the 1989 experiment in which electrons are sent, one at a time, through two slits and we see the buildup of the 2-slit pattern. Clearly it does not take a year and a half of college level physics to develop this background. It could easily be done in one semester of a non calculus high school course.

**Femtosecond Laser Pulses**

Our discussion of the uncertainty principle will focus on the very short infra-red laser pulses first created in the 1990s. These pulses are so short that they contain only a few wavelengths of light. An example is shown in Fig. 1 which shows the intensity of the electric field in the pulse as a function of time³. The pulse is roughly 20 femtoseconds (fs) wide (1 fs = \( 10^{-15} \) sec), and contains six wavelengths in the range \(-10fs\) to \(+10fs\). You see twelve maxima in this region because the intensity and energy content of a wave is proportional to the square of the amplitude. (When you square a sine wave, you get two maxima per cycle.)
Fig. 1. Intensity of the fields in a 20 femtosecond (fs) laser pulse. The center wavelength is 800 nanometers (nm), which corresponds to a period of 2.67 fs. Thus the range from –10 fs to +10 fs should contain about 6 periods. We see 12 maxima in this region because, when you square a sine wave to get intensity, you get two maxima per cycle.

Figure 2 shows the spectrum of frequencies contained in the laser pulse of Fig. 1. The central frequency has a wavelength $\lambda$ of 800 nanometers (nm), (1 nm = $10^{-9}$ meter), but the spectrum ranges from $\lambda = 750$ nm up to 850 nm. One usually thinks of a laser beam as being a very pure beam of light with a single frequency or wavelength. Why does this short pulse contain a spectrum of frequencies?

There are two ways of explaining why. One is to apply the uncertainty principle to the photons in the pulse. The other is to use Fourier analysis to study the harmonics that make up the pulse. Seeing the spectrum of frequencies in the pulse explained in these two ways provides an insight into the origin of the uncertainty principle. We will see that, for the laser pulse, the uncertainty principle is a direct consequence of the particle-wave nature of the waves.

**Using the Uncertainty Principle**

Because of the photoelectric effect $E = hf$, measuring the spectrum of frequencies $f$ in the pulse corresponds to a measurement of the energy $E$ of the photons. But the time $\Delta t$ we have to make this energy measurement is limited to the time it takes the pulse to pass by us, which is about 20 fs for the pulse in Fig. 1. According to the uncertainty principle, if we only have a time $\Delta t$ to make an energy measurement, then the results must be uncertain by at least an amount $\Delta E = \hbar/\Delta t$. We can interpret the spread in frequencies seen in Fig. 2 as resulting from the uncertainty in the energies of the photons.

To apply the uncertainty principle to the laser pulse of Fig. 1, it is easier to turn the analysis around and use the spectrum to calculate how long the pulse was.

We start with the spread in wavelengths that goes from $\lambda = 750$ nm, up to $\lambda = 850$ nm. The corresponding frequencies $f_+$ and $f_-$ and the spread in frequency $\Delta f$ are

$$f_+ = \frac{c}{\lambda_-} = \frac{3 \times 10^8 \text{ m/s}}{750 \times 10^{-9} \text{ m}} = 400 \times 10^{15} \text{ sec}^{-1} \quad (1)$$

$$f_- = \frac{c}{\lambda_+} = \frac{3 \times 10^8 \text{ m/s}}{850 \times 10^{-9} \text{ m}} = 352 \times 10^{15} \text{ sec}^{-1} \quad (2)$$

$$\Delta f = f_+ - f_- = 0.048 \times 10^{15} \text{ sec}^{-1} \quad (3)$$

Now apply the uncertainty principle in the form

$$\Delta t = \frac{\hbar}{\Delta E} = \frac{\hbar}{h \Delta f} = \frac{1}{\Delta f} \quad (4)$$

where Planck’s constant cancelled. We get, using Eq. 3 in Eq. 4

$$\Delta t = \frac{1}{0.048 \times 10^{15} \text{ sec}^{-1}} = 20.8 \times 10^{-15} \text{ sec} = 20.8 \text{ fs}$$

The answer comes out about 21 femtoseconds which is in excellent agreement with what we see in Fig. 1.
Teaching the Uncertainty Principle

The second way to explain the spectrum of wavelengths is to do a Fourier analysis of the harmonics contained in a short pulse. Mathematically a sine wave is infinitely long. If you chop off the ends to create a finite length wave, you have to introduce harmonics to cancel the wave beyond the ends. The shorter the piece of wave you keep, the more harmonics that are needed. We will see that the spectrum of harmonics in Fig. 2 is the spectrum needed to create the pulse seen in Fig. 1.

Applying Fourier analysis to experimental data is analogous to using a prism or diffraction grating to analyze the spectrum of a beam of light. The computer program MacScope II, which is a software oscilloscope, was explicitly designed to capture and then Fourier analyze experimental data. A description of this capability is in the paper “Teaching Fourier Analysis in Introductory Physics.”

In addition to the ability to analyze experimental data, we added to the MacScope II program the capability to create and analyze pulses. To see how this works, we start off by selecting one cycle of a sine wave that was recorded from a sine wave generator. The selection process is shown in Fig. 3. Next we press the button labeled Fourier and get the results shown in Fig. 4. At the top of the figure, the selected section of the wave has expanded to fill the data window. Below that we see one vertical bar in the Fourier analysis window indicating that only one harmonic, the first, is present in the selected section of curve.

As we mentioned, a pure sine wave is an infinitely long wave. MacScope makes the explicit assumption that when it does a Fourier analysis, the selected section of curve repeats indefinitely in both directions. Repeating the one cycle selected in Fig. 3 does give us the single pure sine wave.

To study the harmonics contained in a pulse, we have introduced in MacScope the Pulse Fourier Transform. What this does is instead of expanding the selected section of curve as in Fig. 4 it zeros out all the curve except the selected section as seen in Fig. 5.
Teaching the Uncertainty Principle

As before, MacScope assumes that what is seen in the data window is repeated indefinitely. Thus the curve we are actually analyzing is the set of repeated pulses shown in Fig. 6. This is a reasonable model for pulsed infrared lasers, because these lasers emit a steady stream of pulses. The main difference is that the laser pulses are much farther apart than MacScope’s simulated pulses.

In Figs. 7, the Fourier analysis window shows what harmonics are required to create the pulse shown in the data window. By clicking on the intensity button, the height of the vertical bars now represents the relative intensities of the various harmonics.

When you click on the vertical bar representing a particular harmonic, a picture of that harmonic is superimposed on the curve in the data window. Clicking on the biggest harmonic as shown in Fig. 7a produces a tiny sinusoidal wave that looks nothing like the pulse. This illustrates the great difference between a short pulse and a continuous wave.

To create the pulse, we have to add together a number of harmonics. To see how this works, we have in Fig. 7b selected the five biggest harmonics. MacScope adds together the sinusoidal waves of these 5 harmonics, and superimposes the sum on our pulse curve. Now we see that the sum of the five harmonics is beginning to add up in the region of the pulse and cancel outside the pulse. Altogether there are about 32 harmonics that contribute to the pulse. Selecting 16 of them gives a fair representation of the pulse as seen in Fig. 7c. Selecting all 32 gives a fairly accurate representation seen in Fig. 7d.

The spectrum of harmonics seen in Figs. 7 is analogous to the laser pulse spectrum seen in Fig. 2. We now have an answer to why the pulsed laser must contain a mixture or spectrum of wavelengths. The individual waves in the spectrum have to add together in just the right way to build the pulse and to cancel the waves between pulses.
As an exercise, let us imagine that the stream of pulses in Fig. 6 represents the actual output of a pulsed laser. We will see that in this case the spectrum of harmonics in Figs. 7 turns out to be proportional to the energy spectrum of the photons in each pulse.

We can show this by converting the harmonic scale which has values \( n = 1, 2, 3, \ldots \) to an energy scale. First note that the period \( T_0 \) between the pulses in Fig. 6 is just the period of the first harmonic, as indicated in Fig. 8. The second harmonic has a period half as long, \( T_0/2 \). In general the \( n \)th harmonic has a period \( T_0/n \).

If each harmonic in the pulse corresponds to a light wave moving at a speed \( c \), then a wave of period \( T_n \) will have a frequency \( f_n = c/T_n = n(c/T_0) \). The photons in this harmonic will have an energy \( E_n = hf_n = n(hc/T_0) \). In other words the energy of the photons in a given harmonic is strictly proportional to the harmonic number \( n \). The harmonic scale \( n = 1, 2, \ldots \) in our Fourier analysis plot can be interpreted as an energy scale where we are using a system of units in which \( (hc/T_0) \) is a unit energy.

Because we are plotting intensities of the harmonics, which is proportional to the energy in the wave, we can view our harmonic spectrum as an energy spectrum where the horizontal axis is the individual photon energy and the height of each bar represents the relative amount of energy that photons of that frequency contribute to the pulse.

**Probability Interpretation**

Here is an interesting question. What if, on the average, there were only one photon per pulse? How do you get a spectrum of wavelengths or energies with just one photon?

To answer that question, we look to another experiment that paradoxically involves just one particle at a time. The experiment, which we proposed in 1968\(^5\), was finally carried out in 1989\(^6\). It involves sending electrons, one at a time, through two slits and observing where they strike a distant screen. The results are seen in Fig. 9. Initially, when only 10 electrons have struck, the pattern appears random. The authors say that the next 10 electrons produce a different, apparently random pattern.

But the pattern cannot be random, because, when many thousands of electrons have struck the screen, we see the two-slit interference pattern with its dark bands. The dark bands are where a wave from one of the slits cancels the wave from the other slit. There must be no chance that an individual electron lands on one of these dark bands if the band is to remain dark after thousands of electrons have landed. Even though the electrons were sent through the slits one at a time, some kind of a wave had to go through both slits to produce the cancellation.

The results shown in Fig. 9 follow directly if we interpret the electron wave as a probability wave. The two-slit interference pattern tells us the probability of the electron landing on that particular location.

![Fig. 8. The first few harmonics in the pulses. (Amplitude and phase of the harmonics not to scale.)](image)

![Fig. 9. Experiment in which the electron interference pattern is built up one electron at a time.](image)
landing at some point on the screen. The electron has essentially an equal probability of landing in one of the future bright bands, and zero probability of landing where the waves from the two slits cancel. When only a few electrons have landed, the pattern looks random. But when many have landed, most land where the probability is high, and we see the two-slit pattern emerge.

Returning to the question of how a single photon in a pulse could have a spectrum of wavelengths or energies, the answer lies in the probability interpretation of the photon’s wave. We can interpret the intensity of each harmonic in the Fourier analysis spectrum as being proportional to the probability that the photon has an energy equal to the energy represented by that harmonic. The spectrum represents a probability distribution for the photon’s energy. Looking at Fig. 7 (or Fig. 10a), we see that if this represented an actual laser pulse, a photon in that pulse would have a small probability of having an energy as low as \( \frac{hc}{T_0} \) or as high as \( 32 \frac{hc}{T_0} \). Most likely its energy would be in the range of 8 to 24 times the unit energy \( \frac{hc}{T_0} \).

The important point for this discussion is that the probability interpretation of the photon’s wave requires that the photon’s energy is uncertain. Since the photon’s energy has some probability of being anywhere from 1 to 32 \( \frac{hc}{T_0} \), we can say that, roughly speaking, the uncertainty \( \Delta E \) of the photon’s energy is

\[
\Delta E = 32 \left( \frac{hc}{T_0} \right)
\]

This uncertainty is caused by the fact that the photon is in a short pulse, and to make a short pulse, many harmonics are required.

**Testing the Uncertainty Principle**

If our argument is right, if the uncertainty \( \Delta E \) of the photon’s energy is caused by the shortness \( \Delta t \) in the length of the pulse, and the relationship is given by the simple equation \( \Delta E > h/\Delta t \), then we can make a simple prediction. If we double the length \( \Delta t \) of the pulse, we should cut the minimum uncertainty in energy \( \Delta E \) in half.

To test this prediction, we have in Fig. 10b done a pulse Fourier transform on two cycles of our sine wave. This doubles the length \( \Delta t \) of the pulse, and we see that the range of large harmonics has been cut in half, from about 32 down to 16. Doubling \( \Delta t \) again by selecting 4 cycles in Fig. 10c cuts the range down to 8 harmonics; and doubling the length of the pulse again to 8 cycles, reduces the range of large harmonics to 4.

This brings out the key feature of the uncertainty principle, which we see going up from Fig. 10d to 10a. The less time you have to make an energy measurement, the more uncertain that measurement has to be.

Fig. 10. When we double the length \( \Delta t \) of the pulse, we cut the spread \( \Delta E \) of the harmonics in half. The product \( \Delta E \Delta t \) remains constant.
A More Realistic Pulse

The pulses we have studied so far, where we simply chopped off the wave after one or a few cycles, look a bit unrealistic. To make pulses that look more like the experimental pulse of Fig. 1, we introduced the Gaussian Pulse Fourier Transform which is obtained by selecting the Gaussian instead of Centered menu item seen in Figs. 5a and 11.

To mimic the pulse in Fig. 1, we selected four cycles of our sine wave, chose a Gaussian Pulse, and got the results shown in Fig. 11. The top curve shows the amplitude of the wave we are analyzing. If you squared that wave to get an intensity, you would double the number of maxima and get a result looking much like Fig. 1.

Comparing the harmonics in our Gaussian Pulse with the harmonics in Fig. 10c where we simply chopped the curve off at 4 cycles, we see that the results are quite similar. This begins to show that the spread in harmonics for our laser pulse, the energy uncertainty $\Delta E$, depends on the length $\Delta t$ of the pulse, but not so much on the shape of the pulse within $\Delta t$.

References

   The authors call this pulse a 13 fs pulse because they measure the width where the intensity is greater than half maximum.
Teaching the Uncertainty Principle
III) Introduction to Fourier Optics

Much like a physical prism which displays the frequency components of a light wave, Fourier analysis can be thought of as a mathematical prism that can tell us what harmonics or frequency components are contained in a recording of a sound wave. We wrote the MacScope II program so that the user could not only see a plot of the harmonic amplitudes, but also see the harmonic waves and how these waves add together to recreate the original sound curve.

In this paper, we discuss another powerful feature of Fourier analysis. We see that if we take the Fourier transform of a slit structure, a plot of the harmonic intensities accurately matches the diffraction pattern that results when a laser beam is sent through the slits. In Fig. (1) we show this explicitly for 1, 2, 3, and 4 slit diffraction patterns.

To explain why this happens, we compare the wavefronts produced by a collection of Huygens wavelets emerging from a relatively wide single slit with the harmonic waves in the Fourier transform of a single slit. The match between the two approaches demonstrates that the Huygens construction we use so often in introductory physics courses is essentially a graphical technique for doing Fourier analysis.

The Diffraction Patterns

The diffraction patterns and the slits we used to create them are shown in Fig. (2). We created the slits by drawing them in Adobe Illustrator™, scaling them to the desired size, and then using a high resolution printer to print them in the negative on a transparent sheet. We then sent a laser beam through each slit structure and photographed the results on a distant wall. In the white box we show the right half of the diffraction patterns which we compared to the Fourier transform of the slits.

**Fourier Transform of the Slits**

The essential feature of the slits we constructed was that the slit width was $1/3$ of the spacing between the slits. In Fig. (1) we used the math wave and pulse Fourier transform features of MacScope to construct slit patterns and plot the Fourier coefficients. We then placed the corresponding diffraction pattern underneath each plot.

When we photograph a diffraction pattern, we are observing the intensity or energy density in the wave, which is proportional to the square of the amplitude of the diffracted light wave. Thus, in the Fourier analysis window we clicked on the intensity button to display the intensity or square of the Fourier coefficients.

The only manipulation we did to get the match between the diffraction patterns and the Fourier coefficient patterns was to scale the diffraction pattern photographs so the first minimum in each pattern lined up.

![Fig. 2. Diffraction patterns of the slit structures](image)

**Fig. 1.** Comparison of the Fourier transforms of the slit structures with the diffraction patterns produced by a laser beam passing through the slits.
Why are the Patterns Similar?

To see why the patterns are similar, we will first look at the harmonic waves required to reconstruct a single slit. We will then compare these waves with the amplitudes of the wavelets in a wavefront of a Huygens construction for the slit.

THE FOURIER TRANSFORM

In Fig. (3), we have used MacScope to construct a slit that is 1/10th as wide as the plotting window. When we take the Fourier transform of this slit, the first minimum of the harmonic patterns is at the 10th harmonic. We will see shortly that this is not an accident.

To view the individual harmonics, we have in Fig. (3a) clicked on the first harmonic and see a sine wave where one wavelength just fills the plotting window. In general, if we click on the nth harmonic, we get a wave with n wavelengths within the plotting window.

We would like to have clicked on the 10th harmonic bar to see the 10th harmonic wave, but could not do that because this wave has zero amplitude. About the best we could do is click on the 8th harmonic, and see that we are approaching the case where a full wavelength fits within the slit.

The reason why the tenth harmonic wave has zero amplitude is because the harmonic amplitudes, which we represent by the height of the bars in the Fourier analysis plot, are proportional to the amount of area the wave has within the slit\(^2\). When one full wavelength just fits in the slit, there is just as much positive as negative area, giving it zero net area and a zero harmonic amplitude. In contrast, the first harmonic wave has a maximum area under the slit, with the result that it is the biggest harmonic amplitude.

Note that MacScope shifts the sinusoidal wave left or right so that there is a maximum under the slit. This makes sense because MacScope calculates the harmonic waves that will add up to recreate the slit. To see this recreation, in Fig. (4) we have selected various harmonics, and see that they are adding up to produce the biggest bump at the slit. The more harmonics we add, the closer we get to the slit’s actual shape.

Summary of the Harmonics

In Fig. (5) we have collected the MacScope plots of the harmonics seen in Figs. (3). In doing so, we made one modification. In the MacScope plots, the amplitude of the harmonic waves decrease as we head toward the 10th harmonic. In Fig. (5), we have drawn the harmonic waves all with the same amplitude. This allows us to see the 10th harmonic that has one wavelength and zero net area within the slit.

We have also shaded in the area, within the slit, of these harmonic waves. It is the magnitude of these shaded areas that is plotted by vertical bars in the MacScope Fourier transform plots.

The reason why the tenth harmonic wave has zero amplitude is because the harmonic amplitudes, which we represent by the height of the bars in the Fourier analysis plot, are proportional to the amount of area the wave has within the slit\(^2\). When one full wavelength just fits in the slit, there is just as much positive as negative area, giving it zero net area and a zero harmonic amplitude. In contrast, the first harmonic wave has a maximum area under the slit, with the result that it is the biggest harmonic amplitude.

Note that MacScope shifts the sinusoidal wave left or right so that there is a maximum under the slit. This makes sense because MacScope calculates the harmonic waves that will add up to recreate the slit. To see this recreation, in Fig. (4) we have selected various harmonics, and see that they are adding up to produce the biggest bump at the slit. The more harmonics we add, the closer we get to the slit’s actual shape.

Summary of the Harmonics

In Fig. (5) we have collected the MacScope plots of the harmonics seen in Figs. (3). In doing so, we made one modification. In the MacScope plots, the amplitude of the harmonic waves decrease as we head toward the 10th harmonic. In Fig. (5), we have drawn the harmonic waves all with the same amplitude. This allows us to see the 10th harmonic that has one wavelength and zero net area within the slit.

We have also shaded in the area, within the slit, of these harmonic waves. It is the magnitude of these shaded areas that is plotted by vertical bars in the MacScope Fourier transform plots.
THE HUYGENS CONSTRUCTION

To explain why the Fourier transform plot matches the diffraction pattern, we will now analyze the slit using the Huygens construction. In this construction, one breaks the wavefront emerging from the slit into a number of short sections, and treats each section as a source of a spherical wavelet. Far from the slit, say on a distant wall, these wavelets will add together or cancel each other to produce a diffraction pattern.

In Fig. (6) we have drawn a slit of width W that is fairly wide compared to the wavelength $\lambda$ of the wave pressing through it. We have broken the emerging wave into 10 sections each with its own wavelet. In Fig. (6a) we show the wavelets heading in the forward direction. In the remaining figures we see the wavelets heading up at increasing angles. When the wavelets reach the distant wall, they will have converged to a point, and all the wavelets will add together.

In Fig. (6d), at the angle where the path length difference to the screen is one wavelength $\lambda$ as shown, all the wavelets cancel in pairs, wave (#1) cancelling (#6), (#2) cancelling (#7), etc. Thus this is the angle to the first minimum in the diffraction pattern. Figures (6a), (6b), and (6c) show how we approach this minimum.

In Fig. (6a) we have drawn a wavefront, indicated by the dashed line, that passes through the maxima of all the wavelets. When this wavefront reaches the screen, all these wavelets will add together to produce a maxima at the center of the diffraction pattern. On the right side of Fig. (6a) we have drawn vectors representing the amplitudes of the wavelets in our wavefront. At the distant screen we calculate the brightness at the center by adding these vectors together.

In Fig. (6b), we are looking at the wavelets as they head upward at a slight angle. We moved our wavefront line so that it passes through the maxima of the middle wavelets. The amplitude of the end wavelets (#1) and (#10) are down a bit as shown in the plot of the amplitude vectors in the right hand plot.

In Fig. (6c) we are looking up at an angle where the path length difference between the top and bottom wavelet is $\lambda/2$. When we look at a wavefront where the center wavelets have a maximum, the edge wavelets have zero amplitude as we see in the drawing of the amplitude vectors. The wavefronts we selected in Figs. (6b) and (6c) give the maximum amplitude we can get when we add the corresponding wavelet amplitudes together.

---

**Fig. 5.** The Fourier coefficient $c_n$ is proportional to the area, within the slit, under the $n$'th harmonic wave.

**Fig. 6.** Wavefront amplitudes at different angles. The sinusoidal envelope curves correspond to the harmonics in the Fourier transform of the slit.
**Adding Wavelet Amplitudes**

Exactly how do we add the wavelet amplitudes? That appears to depend on how we did the Huygens construction. In Figs. (6) we broke the slit into 10 sections. Thus we can think of each wavelet as carrying 1/10 of the light passing through the slit. If, instead, we had broken the slit down into 20 sections, we would have 20 wavelets. We would get twice as many wavelets, but each wavelet would be half as strong. At the screen we would have 20 vectors rather than 10 to add up.

Another way of looking at the problem is to draw the envelope curve that passes through the arrow tips. This envelope curve is a sinusoidal curve whose wavelength gets shorter and shorter as we increase the angle. At the angle of the first minimum, precisely one wavelength of the envelope curve fits within the slit width W.

The process of adding up the wavelet vectors is equivalent to finding the (shaded) area of the envelope curve within the slit. Since you get the same shape envelope curve with 10 vectors as you do with 20 vectors, calculating the area under the envelope curve gives you an answer that is independent of the number of wavelets (provided you use enough of them).

In Fig. (7) we compare the Huygens’ envelope curves we have just calculated, with the harmonic waves calculated by MacScope. We see that they are the same curves. The diffraction pattern amplitude for each angle is the area under the envelope curve. The Fourier coefficient amplitude is the area under the harmonic wave.

![Fig. 7. Comparison of the areas under the wavelet amplitude envelope curves with the areas under the Fourier harmonics. The similarity demonstrates that Fourier analysis is the mathematics behind the Huygens constructions.](image)

The reason for the first minimum is when these curves have zero net area within the slit. If we scale the diffraction so that the first minima line up as in Figs. (2), the pattern of Fourier coefficients will be similar to the diffraction pattern amplitude.

Perhaps the lesson we should take from this exercise is that the Huygens construction is a graphical method of doing Fourier analysis. Or, that Cristian Huygens was doing Fourier analysis nearly 150 years before Fourier developed the related mathematics.

Our examples so far represent one dimensional optics, where our patterns vary is only one dimension (across the slits). To give a feeling for how our discussion extends to two dimensional objects and images, we would like to quote from Theo Lasser’s discussion of the Rayleigh model. The “Rayleigh model for image formation considers an image as a combination of point source diffraction patterns (e.g. Airy profiles) which an optical system would produce from light leaving individual points of the object.”

**Notes**

1. MacScope II is now a shareware program that can be downloaded from www.physics2000.com and used anywhere. The program has been tested to run on Windows 98 & XP, and on Mac OS8,9 & OSX. The version on the web is the latest version, and may be considered an upgrade for those who own the $10 Physics2000 CD. Any questions concerning the use of MacScope should be addressed to lish.huggins@dartmouth.edu.

2. We developed the non calculus derivation of the Fourier coefficients for our daughter’s high school physics class. The result is that the formula for the n’th harmonic $c_n$ is

$$\pi c_n = \text{area under} [f(t) \cos (nt - \varphi_n)]$$

where $f(t)$ is the function being analyzed and $\varphi_n$ is a phase factor. For the single slit we are discussing, $f(t)$ is equal to 1 inside the slit and zero outside. The phase factor $\varphi_n$ moves the cosine wave left or right so that it has a maximum within the slit.

The derivation can be found in the Teaching Fourier Analysis article, on pages 5-6 of this book.

3. Theo Lasser Optical Design II

http://lob.epfl.ch/webdav/site/lob/shared/Teaching/Optical%20Microscopy%20and%20Metrology/Fourier%20optics
IV) Speed of Sound in Metal Pipes

We describe a direct measurement of the speed of sound in a steel and in a copper pipe using apparatus that costs less than $100, plus a USB equipped computer.

WAVE MOTION

One of the best ways to introduce wave motion in a lecture demonstration is to send a wave pulse down a horizontally suspended stretched Slinky™. Both transverse and compressional pulses travel at nearly the same speed, given by the formula \( v = \sqrt{\frac{T}{\mu}} \), where \( T \) is the tension in the Slinky and \( \mu \) its mass per unit length. (For a compressional wave, the formula is \( v = \sqrt{\frac{KL}{\mu}} \) where \( K \) is the spring constant and \( L \) is the length of the spring. A non calculus derivation of this formula is given in the next article.) When you set up this demonstration you can watch a pulse move slowly back and forth across the Slinky and quite easily check the formula for the speed of the wave.

In contrast, the speed of sound in a rigid material like steel is far higher even than the speed of sound in air. Introductory physics texts¹ ² give values for the speed of sound in steel that range from 5000 meters per second¹ to 5941 m/s², with Wikipedia³ coming in at 5100 m/s. We wondered about the variation in values, and have long wondered if there were a way for students to measure these high speeds.

Watching the Slinky wave bounce back and forth suggested the following experiment. Tape a microphone to one end of a steel pipe, for example the 10 ft. (3 meter) electrical conduit pipes found in hardware stores. Hit the other end of the pipe with a hammer and record the sound pulse as it bounces back and forth down the pipe. A return trip of 20 ft. (6 meters) should take about 1.2 milliseconds if the pulse traveled at the predicted speed of 5000 m/s or 5 meters per millisecond. A time of 1.2 milliseconds is easily measured by an audio oscilloscope.

![Fig. 1. Experimental setup. The Griffin Technology iMic5 takes the input from two microphones, does a stereo A/D conversion and sends the result to the computer’s USB port. In the computer, the shareware program MacScopeII has the computer act as a dual](image1)

![Fig. 2. Hammer striking pipe. Battery powered mono lapel microphones are taped to each end of the pipe.](image2)

![Fig. 3. The signals from the 2 microphones enter the Y connector supplied by Griffin. We had to convert from mono 1/8” microphone plugs to RCA inputs on the Y connector.](image3)

![Fig. 4. Setting both oscilloscope screens to trigger mode and using a low trigger voltage level.](image4)
THE EXPERIMENTS

We tried this experiment taping a $9 Wal-Mart microphone to the end of a conduit pipe and recorded the microphone output using the MacScope II shareware audio oscilloscope program. We did see a pattern of spikes about a millisecond apart, but there were a number of other bumps in the signal. It was not clear whether we were observing a pulse bouncing back and forth, or observing some complex pattern of modes of oscillation of the pipe.

Two Microphones

To be sure that we were measuring the speed of a pulse, we decided to tape a Wal-Mart microphone to each end of the pipe and use MacScope’s dual beam capability to record the delay in the first arrival of the pulse to the far microphone. This did not work because the inexpensive Wal-Mart electric microphones are powered by the $40 iMic USB sound interface we were using. In order for the iMic to power both microphones, the output signals had to be connected together and we could not see the time delay.

This problem was cured by using two $25 battery powered lapel microphones from Radio Shack. We also needed two $4.00 connectors to allow the microphones to plug into the Y connector supplied with the iMic. This brought our total cost to $98, including the cost of the iMic but not the conduit.

The setup is shown in Fig.(1). In Fig.(2) we see a lapel microphone taped to the conduit. Figure (3) shows the required connectors, and the MacScope trigger settings are seen in Fig.(4).

In Fig.(5) we see a clear delay of 0.60 milliseconds indicating that the pulse traveled at a speed of 5080 meters per second down the 3.048 meter long pipe. This is closest to the 5100 m/s value given by Wikipedia.

One concern remained. Are the two MacScope windows for curve A and curve B really using the same time scale? Is the 0.60 ms time delay real? To find out, we moved the far microphone to the middle of the pipe and repeated the measurement. If the curve B time delay were real, the delay should be cut in half, which it is, as seen in Fig.(6).

\[ v_{\text{steel}} = \frac{10 \text{ ft}}{0.06 \text{ ms}} = \frac{3.048 \text{ meters}}{0.0006 \text{ sec}} = 5080 \text{ meters/sec} \]

Fig. 5. Curve A shows the signal from the microphone at the hammer end of the pipe. The microphone at the other end, whose output is seen in curve B, shows no response for 0.60 milliseconds. The spikes in curve A are 1.20 milliseconds apart, representing round trips of the sound pulse.

Fig. 6. As a check that both curve A and Curve B time scales start at the same time, we moved the B microphone to the middle of the pipe, and got half the delay.
Copper pipe

We happened to have a 6 ft. section of 1/2 inch copper pipe left over from a construction project. Tapping the microphones to that pipe, we got a delay of 0.47 ms, corresponding to a sound speed in copper of 3900 m/s. Introductory physics texts give the formula \( v = \sqrt{\frac{Y}{\rho}} \) for the speed of sound in a metal rod, where \( Y \) represents the stiffness (Young Modulus) of the material and \( \rho \) its density. Using the density values \( \rho = 8.0 \text{ gm/cm}^3 \) for steel and 8.9 gm/cm3 for copper, we get as the ratio of the stiffness of the two metals as

\[
\frac{Y_{\text{steel}}}{Y_{\text{copper}}} = \frac{(\rho v^2)_{\text{steel}}}{(\rho v^2)_{\text{copper}}} = 1.53
\]

indicating that steel is about one and a half times as stiff as copper.

Notes:

4. The shareware program MacScope II, which turns any USB Mac or Windows computer into an audio oscilloscope, can be downloaded from [www.physics2000.com](http://www.physics2000.com)
5. The iMic is discussed at [www.griffintechnology.com/products/imic](http://www.griffintechnology.com/products/imic)
V) Speed of Wave Pulses in Hooke’s Law Media

As students watched a compressional pulse bounce back and forth on the horizontally suspended Slinky™, shown in Fig.(1), we wrote down the formula for the speed of the pulse, and promised that later in the course we would derive the formula. As part of our introduction to Einstein’s special theory of relativity, we emphasized that the formula was for the pulse’s speed relative to the Slinky medium. It would predict the pulse’s speed past us only if we were at rest relative to that medium.

The problem is that we did not keep our promise. Following the standard practice in introductory physics texts, we derived the formula \( v_T = \sqrt{\frac{K(L - L_0)}{\mu}} \) for the speed \( v_T \) of a transverse wave in a rope or spring of mass per unit length \( \mu \) subject to a tension \( T \). We never found a satisfactory derivation for the speed \( v_C \) of a compressional pulse on a spring. This past summer we set out to create a non calculus derivation for \( v_C \) that had the same conceptual simplicity as the standard derivation of \( v_T \).

The derivation we came up with applies to a compressional pulse in any media obeying Hooke’s law \( F = K(L - L_0) \) where \( F \) is the restoring force when an object of unstretched length \( L_0 \) and spring constant \( K \), is stretched by an amount \((L - L_0)\). For such a Hooke’s law medium, the formulas for \( v_T \) and \( v_C \) become

\[
v_T = \sqrt{\frac{K(L - L_0)}{\mu}}; \quad v_C = \sqrt{\frac{KL}{\mu}} \tag{1}
\]

In this paper we outline the derivation of the formula for \( v_C \), and put the full derivation in an appendix.

Most of the paper is devoted to an experimental study of these formulas in two cases with widely different parameters. One is the horizontally stretched Slinky that has essentially no unstretched length \( L_0 \), a weak spring constant of just over a quarter of a newton per meter, and wave speeds of the order of one meter per second. Because \( L_0 \) is so small, \( v_T \) and \( v_C \) are essentially the same.

The other example is a stretched steel guitar string wire that consists almost entirely of unstretched length \( L_0 \) and has a spring constant 20,000 times larger than that of the Slinky. Because the length \( L \) is so much bigger than the distance \((L - L_0)\) the wire is stretched, \( v_T \) and \( v_C \) are very different, with \( v_C \) being the expected speed of sound in steel, close to 5,000 meters/ sec. The experiments confirmed both formulas with a greater degree of accuracy than we expected.

Fig. 1. Stretched horizontal Slinky™ for the demonstration of both transverse and compressional (longitudinal) wave pulses.
DERIVATION OF THE SPEED FORMULAS

In the standard derivation of the speed of a pulse on a stretched rope, you walk along with the pulse so that you see the rope moving through a stationary pulse. You notice that the top of the pulse is essentially circular, allowing you to use the circular motion formula \( a = \frac{v^2}{r} \) to calculate the rope’s acceleration. Noting that this acceleration is caused by the tension forces \( T \) in the rope, you easily end up with the formula \( v_T = \sqrt{\frac{TM}{\mu}} \).

For the derivation of the speed \( v_c \) of a compressional pulse in a Slinky, we also walk along with the pulse so that we see a stationary pulse with the Slinky coils moving through it as shown in Fig.(2). Inside the pulse the Slinky coils are closer together. To see that the coils must also be moving slower inside the pulse than outside it, we thought of an analogy to the flow of cars in a highway construction bottleneck.

As you approach such a construction site, you have to slow down, and endure bumper to bumper traffic. When you get past the bottleneck, you can speed up (accelerate), and then resume driving at your normal speed. The same thing happens to the Slinky coils. As they approach the pulse, they have to slow down and come closer together. Once through the pulse they speed up (accelerate) and resume their normal spacing and speed. As shown in Fig.(2), the acceleration out of the pulse is caused by the difference in the Hooke’s law force on the closely spaced coils inside the pulse and the more stretched apart coils outside the pulse. In the appendix we relate the acceleration of the coils out of the pulse to this net Hooke’s law force to obtain the formula \( v_C = \sqrt{\frac{KL}{\mu}} \). (The dependence on \( L_0 \) cancels when we calculate the difference in the Hooke’s law forces.)

A Special Feature

When we first derived the formula for \( v_c \) and tried applying it, we found that it made an unexpected prediction. If you set \( \mu = \frac{M}{L} \) where \( M \) is the mass of the Slinky, and use the formula for \( v_c \) to calculate the time \( t \) it takes to go down and back, a distance \( 2L \), you get

\[
v_C = \sqrt{\frac{KL}{\mu}} = \sqrt{\frac{KL}{M/L}} = L \sqrt{\frac{K}{M}}
\]

\[
t = \frac{2L}{v_C} = 2 \sqrt{\frac{M}{K}}
\]

(2)

Since the Slinky’s stretched length \( L \) cancels, Eq.2 predicts that the time it takes the pulse to go down and back does not depend on how far you stretch the Slinky. Not believing this prediction, we brought a video camera to the lecture hall, adjusted the Slinky to the four different lengths shown in Fig.(3), and taped the motion of a compressional pulse down and back. The resulting video can be seen online at [video]. In all four cases, for lengths ranging from 2.60 meters down to 0.86 meters, the times were within \( 1/30 \) of a second of each other. We felt that this was an impressive verification of the formula for \( v_c \).

We were not the only ones surprised by this result. In a classroom demonstration for elementary school teachers, David Keeports\(^{1,2} \) of Mills College accidentally found that the travel time of the compressional pulse on a stretched spring was independent of the length of the spring. That led him to search for the formula that would explain this interesting experimental result. He finally found the formula \( v_C = \sqrt{KL/\mu} \) in two relatively advanced textbooks on wave motion.

---

**Fig. 2.** The point of view that the Slinky is moving through a stationary pulse resembles the situation of cars moving through a construction bottleneck. Cars slow down as they enter the bottleneck and speed up as they leave. Coils slow down as they enter the pulse, and accelerate as they leave. This acceleration is caused by the difference \( (F_1 - F_2) \) in the Hooke’s law force. \( F_1 \) is weaker where the coil spacing is less.)
MEASURING SPRING CONSTANTS

In order to test the speed formulas in Eq.1, we first needed to measure the spring constants of the Slinky and the guitar string. The results are shown in Fig.(4). For the Slinky, we had to be careful that the support threads were straight up and down each time we measured the tension in the Slinky. For the guitar string we used the apparatus sketched in Fig.(5). For that experiment we used a sturdy pivot nail and the lightest steel guitar string available.

EXPERIMENTAL RESULTS

Our first result was the observation that \( v_T \) and \( v_C \) were essentially the same on the Slinky. We could not do as accurate a measurement of the speed of our longer transverse pulses, and thus could not detect any effect of the small negative \( L_0 \).

In the video, we found that the average value of the time down and back was 2.85 seconds. Our prediction for this time is

\[
t = \frac{2L}{v_C} = 2 \sqrt{\frac{M}{K}} = 2 \sqrt{\frac{0.510 \text{ kg}}{0.266 \text{ n/m}}} = 2.77 \text{ sec}
\]

where we measured the Slinky mass at .510 kg and used our value of 0.266 newtons/meter for \( K \). The result, 2.77 sec, is only 3% less than the experimental result.

To study wave speeds on the guitar string, we measured the oscillation periods \( t_T \) and \( t_C \) for transverse and compressional, resonant standing wave vibrations, of the guitar string. Using the idea that a standing wave can result from two traveling waves moving through each other, we can relate the traveling wave speed to the periods of oscillation. We then note that the period of oscillation of the first harmonic is the time the traveling wave takes to travel one wavelength, which is twice the length of the wire. Thus the period of oscillation of the first harmonic is the same as the time it would take a pulse to travel down and back.
At this point, a paper by Karl Mamola\(^2\) becomes crucial. How do you set up a compressional wave in a wire? In a steel pipe we set up a compressional pulse by hitting the end of the pipe with a hammer, but that does not work for a slender guitar string. Mamola’s paper told us to stroke the stretched wire with a wet sponge and the wire will sing with a compressional wave. The effect resembles rubbing a damp finger around the rim of a wine goblet.

To measure the resonant frequencies, we set the tension in the wire to 32.15 newtons, gently strummed the wire at the center to create a transverse oscillation, and stroked it with a wet sponge to create the compressional oscillation. These sounds were recorded by MacScope\(^3\) as shown in Fig.(6). The resulting periods of oscillation were \(t_T = 4.75 \text{ ms}\) for the transverse wave and \(t_C = 0.382 \text{ ms}\) for the compressional wave. In order to record the fundamental note of the transverse wave we had to place the microphone very close to the wire.

The corresponding velocities \(v = 2L/t\) are

\[
\begin{align*}
  v_T &= \frac{2 \times .953 \text{ m}}{4.75 \times 10^{-3} \text{ s}} = 401 \text{ m/s} \\
  v_C &= \frac{2 \times .953 \text{ m}}{0.382 \times 10^{-3} \text{ s}} = 4990 \text{ m/s}
\end{align*}
\]

where \(L\) was .953 meters. Unlike the example of the Slinky where \(v_T\) and \(v_C\) were essentially the same, for the guitar string the compressional wave was over 12 times faster than the transverse wave.

The one part of the experiment that required specialized equipment was accurately weighing the fine (.18 mm) steel guitar string. To get three place accuracy we used a Mettler balance and got a value of 0.196 gm for a 99.06 cm length of wire, giving us \(\mu = 1.98 \times 10^{-4} \text{ kg/m}\). Using this value of \(\mu\), and with \(K = 5120 \text{ n/m}\) for the wire, \(L = .953 \text{ m}\), and \(T = 32.15 \text{ n}\), we get for the predicted speeds

\[
\begin{align*}
  v_T &= \sqrt{\frac{T}{\mu}} = \sqrt{\frac{32.15}{1.98 \times 10^{-4}}} = 403 \text{ meters/see} \\
  v_C &= \sqrt{\frac{KL}{\mu}} = \sqrt{\frac{5120 \times .953}{1.98 \times 10^{-4}}} = 4960 \text{ meters/see}
\end{align*}
\]

Both of these theoretical values are within 1\% of our experimental results. Also the speed \(v_C\) of the compressional pulse is within 2.5\% of the value \(v_C = 5080 \text{ m/s}\) we got for the speed of a sound pulse produced in a steel pipe hit by a hammer\(^4\).

References
4. See Article 4) Speed of Sound In Metal Pipes.

Fig. 5a. Our actual experimental setup for studying waves on a guitar string. Due to the craftsmanship and fine materials used in this setup, we may get a call from the Smithsonian.

Fig. 6. Fundamental frequencies of the transverse and compressional waves. We selected 10 cycles of both waves.
APPENDIX: DERIVATION OF THE SPEED OF THE COMPRESSIONAL WAVE IN A SLINKY

In the Slinky movie, we create a compressional pulse by pulling back on the end of the Slinky and letting go. The striking feature of the movie is that the pulse took the same length of time to go down and back, independent of the length L of the stretched Slinky. Here we derive the formula that predicted this result, namely

\[ v_{\text{pulse}} = \sqrt{\frac{KL}{\mu}} \]

Figure (A1) is a sketch of a compressional Slinky pulse. The coils drawn as bold lines are part of the compressional pulse, where the lighter lines represent the uncompressed part of the Slinky. We are assuming that the pulse is moving to the left at a speed \( v_{\text{pulse}} \).

The analysis is simplified if we take the point of view that we are walking along with the pulse, so that we see a stationary pulse with the Slinky moving to the right, through the pulse, as shown in Fig.(A2). The uncompressed part of the Slinky is moving to the right at a speed \( v_{\text{Slinky}} \) which is opposite to \( v_{\text{me}} \) of Fig.(A1).

What may seem surprising is that the more closely spaced coils inside the pulse are traveling a bit slower than the coils outside the pulse. But this is actually a familiar situation. If you are driving down a highway and come to a construction site, the cars slow down and come closer together. After you get through the construction site, the cars speed up and their spacing increases. In an aerial view of the highway, the compression of the car spacing at the construction site would look much like our compressional pulse on the Slinky, as shown in Fig.(A3).

As the cars approach the construction site they have to slow down, i.e., decelerate. Once through the site, they have to accelerate to get back up to speed. By analogy, the Slinky coils entering the pulse have to slow down, and then accelerate again once through the pulse. It is the Hooke’s law spring forces that cause the coils to decelerate and then accelerate.

Our analysis will focus on the coils leaving the pulse, the coils that have to speed up from the slower speed \( v_1 \) at the center of the pulse to the higher speed \( v_2 \) of the coils outside. We will calculate the acceleration of these coils, the spring forces causing this acceleration, and then use Newton’s second law to find the formula for the speed of the pulse.

Figure A1
Compressional pulse in a Slinky moving to the left at a speed \( v_{\text{pulse}} \). I am walking along with the pulse.

Figure A2
I see a stationary pulse with the Slinky coils moving through it.

Figure A3
Slinky coils passing through the compressional pulse behave like cars driving through a construction bottleneck. They slow down and get closer together as they enter the pulse or bottleneck, and speed up when they leave.
Conservation of Coils

We can relate the spacing \( d_1 \) and \( d_2 \) between coils to their respective speeds \( v_1 \) and \( v_2 \) by using the idea that coils do not get lost as they pass through the pulse. If we calculate the number of coils per second passing a point along the Slinky, that number must be the same all the way along. For example, if there were fewer coils per second leaving the pulse than the number per second passing by the center of the pulse, that would imply that not all the coils made it out.

The use of dimensions tells us how to calculate the number of coils per second. The coil spacings \( d_1 \) and \( d_2 \) have dimensions

\[
(d_1,d_2)\text{meters/coil}
\]

while the velocities \( v_1 \) and \( v_2 \) have the usual dimensions meters/second. The ratios \( v_1/d_1 \) and \( v_2/d_2 \) have dimensions

\[
\frac{v_1}{d_1}\text{meters/second} \quad \frac{v_2}{d_2}\text{meters/second}
\]

The meters cancel and we are left with

\[
\frac{v_1}{d_1}\text{ coils/second}
\]

as our formula for the number of coils per second passing by us. Conservation of coils requires

\[
\frac{v_1}{d_1} = \frac{v_2}{d_2}
\]

In Fig.(A4) you can see that both \( d \) and \( v \) increase as the coil comes out of the pulse, with the result the ratio \( d/v \) remains constant.

Later on we will use Eq.(A4) in a slightly different form which we get as follows

\[
v_1 = d_1 \frac{v_2}{d_2}
\]

\[
\frac{v_1}{v_2} = \frac{d_1}{d_2}
\]

Equation (A4a) will allow us to express changes in velocity \( v \) in terms of changes in coil spacings \( d \).

Acceleration of the Coils

As we mentioned, coils leaving the pulse have to accelerate from the lower speed \( v_1 \) to the higher speed \( v_2 \). They have to do this in the time \( t \) it takes a coil to move from the center to the edge of the pulse. If the acceleration \( A \) is constant during this time, the constant acceleration formulas give

\[
v_2 = v_1 + At
\]

\[
A = \frac{v_2 - v_1}{t}
\]

Let the half width of the pulse be \( W \) as shown in Fig.(A5) and let \( v_{av} = (v_1 + v_2)/2 \) be the average velocity of a coil leaving the pulse. Then the time \( t \) to leave the pulse is

\[
t = \frac{W}{v_{av}}\text{meters/sec} = \frac{W}{v_{av}}\text{sec}
\]

Our formula for the coil’s acceleration while leaving becomes

\[
A = \frac{1}{t}(v_2 - v_1) = \frac{v_{av}}{W}(v_2 - v_1)
\]

Figure A5

Calculating the acceleration of the coils leaving the pulse.
**Weak Pulse**

Equation (A6) is an exact result if the coil exits with constant acceleration. However we get a simpler formula if we assume that the pulse is weak, meaning that the velocity $v_1$ at the center of the pulse is not much less than the outside velocity $v_2$. That means that the average velocity $v_{av}$ is not much different from $v_2$, and we can replace $v_{av}$ by $v_2$ in Eq.(A6), to get

$$A = \frac{v_{av}(v_2 - v_1)}{W} \approx \frac{v_2(v_2 - v_1)}{W} = \frac{v_2^2}{W}(1 - \frac{v_1}{v_2})$$

(A-7)

We will find it convenient to use our formula for the conservation of coils to express the ratio $v_1/v_2$ in terms of the spacings $d_1$ and $d_2$,

$$\frac{v_1}{v_2} = \frac{d_1}{d_2}$$

(A-4 repeated)

Thus

$$A = \frac{v_2^2}{W} \left(1 - \frac{d_1}{d_2}\right)$$

(A-8)

**Mass of Exiting Coils**

We will assume that the acceleration of the exiting coils is caused by the Hooke’s law spring force acting on them. We are going to calculate the net spring force on the entire section of the exiting coils (all the coils within the width $W$), and then equate this net force to the total mass $M$ of the exiting coils times their acceleration $A$.

To calculate the mass $M$ of the exiting coils, we will again assume that the pulse is weak, and that the mass per unit length $\mu$ inside the pulse is not much greater than it is in the rest of the Slinky. With this assumption, the mass $M$ of a length $W$ of the Slinky is simply

$$M \text{ kg} = \mu \frac{\text{kg}}{\text{meter}} \times W \text{ meters} = \mu W \text{ kg}$$

(A9)

**MA forExiting Coils**

The product $MA$ for the exiting coils, using Eqs.(A8) and (A9), becomes

$$MA = \mu W \times \frac{v_2^2}{W} \left(1 - \frac{d_1}{d_2}\right)$$

The half width $W$ of the pulse cancels and we are left with

$$MA = \mu v_2^2 \left(1 - \frac{d_1}{d_2}\right)$$

(A-10)

We will need to remember that $v_2^2$ is simply the square of the speed of the pulse ($v_2^{\text{pulse}}$) as seen from the point of view of Fig.(1) where the Slinky is at rest and the pulse is moving.

**Figure A6**

*Measuring the force required to stretch the Slinky.*
**Hooke’s Law for the Suspended Slinky**

Before we used Hooke’s law, we wanted to make sure that it actually applied to the suspended Slinky. After all, there are a lot of threads attached to the Slinky and the threads might have some effect on the motion of the pulse. (The threads do affect your measurements if you are not careful to make sure that the threads are straight up and down.)

In Fig.(A6) we show the results of the measurements of the forces $F$ required to stretch the Slinky four different lengths $L$. The result is that all the points lie along a straight line that intersects the zero force axis at a distance

$$L_0 = -6.2\, \text{cm} = -0.062\, \text{m}$$

(A-11)

The formula for this line is

$$F = K(L - L_0) \quad \text{Hooke’s law} \quad (A-12)$$

which is Hooke’s law. The spring constant $K$ for this particular Slinky is

$$K = 0.266 \, \text{newtons/meter}$$

(A-13)

which we obtain by solving Eq.(A-12) for $K$ and using numerical values for one of the points on the line. The summary of these results is shown in Fig.(A7).

The minus sign for the unstretched length $L_0$ simply means that the Slinky was wound in such a way that some force is required even to begin to stretch it.

**Figure A7**

Demonstration that our suspended Slinky obeys Hooke’s law. The mass of the Slinky is 0.510 kg.

**Net Spring Force**

We are now in a position to use Hooke’s law to calculate the net spring force acting on the section of coils emerging from the pulse. These are the coils in the section of length $W$, subject to the left directed force $F_1$ acting in the middle of the pulse, and the right directed force $F_2$ at the edge of the pulse, as shown in Fig.(A8).

The spring force $F_1$ is not as strong as the spring force $F_2$ because the coils at the center are not stretched as far apart as the coils at the edge. It is the difference in the strengths of these forces, $(F_2 - F_1)$ that gives rise to the net right directed force that causes the acceleration $A$ that we just calculated.

We can use Hooke’s law to express the spring forces $F_1$ and $F_2$ in terms of the spring lengths $L_1$ and $L_2$. We have

$$F_1 = K(L_1 - L_0)$$
$$F_2 = K(L_2 - L_0)$$

When we calculate the difference $(F_2 - F_1)$ in the spring forces, we get

$$(F_2 - F_1) = (KL_2 - KL_0) - (KL_1 - KL_0)$$
$$= KL_2 - KL_1 + KL_0 - KL_0$$
$$= KL_2 - KL_1$$

(A-14)

**Figure A8**

Because the coils are more stretched apart at the edge of the pulse than at the center, the spring tension $F_2$ is greater at the edge than the tension $F_1$ at the center. The net force on this section of Slinky is $(F_2 - F_1)$. 

The important feature of this calculation is that the $KL_0$ terms cancel, and we see that the net force $(F_2 - F_1)$ acting on our section of the Slinky does not depend on the spring’s unstretched length $L_0$.
In order to relate our net force \( F_{\text{net}} = (F_2 - F_1) \) to our formula for \( MA \) in Eq.(A-10) we will rewrite Eq.(A14) in the form

\[
F_{\text{net}} = (F_2 - F_1) = KL_2(1 - \frac{L_1}{L_2}) \quad (A-14a)
\]

Our next step will be to express this net force in terms of the coil spacings \( d_1 \) and \( d_2 \) rather than the total lengths \( L_1 \) and \( L_2 \).

In Fig.(A9) we see that if there are \( N \) coils in the Slinky and we simply stretch the Slinky to a length \( L_1 \), the distance \( L_1 \) and \( d_1 \) will be related by

\[
L_1 = Nd_1; \quad d_1 = L_1/N \quad (A-15a)
\]

Likewise if we pull the Slinky out to a greater distance \( L_2 \) we have

\[
L_2 = Nd_2; \quad d_2 = L_2/N \quad (A-15b)
\]

Since our formula for \( MA \) involved the ratio \( d_1/d_2 \), we can use Eqs.(A-15) to express this ratio in terms of the spring lengths \( L \)

\[
\frac{d_1}{d_2} = \frac{L_1/N}{L_2/N} = \frac{L_1}{L_2} \quad (A-16)
\]

Using Eq.(A-16) in Eq.(A-14a) gives

\[
F_{\text{net}} = KL_2(1 - \frac{d_1}{d_2}) \quad (A-17)
\]

Next, note that \( L_2 = Nd_2 \) is essentially the length \( L \) of the Slinky, since all the coils in the Slinky except the few in the pulse have a separation \( d_2 \). Thus our formula for \( F_{\text{net}} \) becomes

\[
F_{\text{net}} = KL(1 - \frac{d_1}{d_2}) \quad (A-18)
\]

**Applying Newton’s Second Law**

To apply Newton’s second law, we equate \( F_{\text{net}} \) from Eq.(A-18) to \( MA \) from Eq.(A-10) to get

\[
F_{\text{net}} = MA
\]

\[
KL(1 - \frac{d_1}{d_2}) = \mu v_2^2(1 - \frac{d_1}{d_2})
\]

The factors \((1 - d_1/d_2)\) cancel and we are left with

\[
KL = \mu v_2^2 \quad (A-19)
\]

Remembering that \( v_2 \), the speed of the Slinky through the pulse, has the same magnitude as the pulse speed \( v_{\text{pulse}} \), we get

\[
KL = \mu v_{\text{pulse}}^2 \quad (A20)
\]

Solving for \( v_{\text{pulse}} \) gives us our final result

\[
v_{\text{pulse}} = \sqrt{\frac{KL}{\mu}} \quad (A-21)
\]

**Figure A9**

*Relationship between the spring length and the coil spacing.*
The MacScope II program is an audio software oscilloscope for both Mac and Windows which uses the computer’s sound input to acquire data. The main features are Fourier analysis, signal averaging and working with triggered curves.
AUTHOR’S INTRODUCTION

In my third year of teaching physics at Dartmouth College, back in the 1960s, I was assigned to teach a large introductory course. The lab for that course had a collection of Heathkit oscilloscopes which I, as a theoretical physicist, found impossible to operate. If you changed any scale, the curve disappeared somewhere for a few seconds, then floated by a few times before settling down somewhere. I refused to teach the course unless we scrounged up some Tektronix scopes that were stable enough for even me to operate. Shortly after that I got a grant to replace all the Heathkits with Hewlett Packard scopes which, remarkably, are still running today.

When I first saw the Macintosh computer in 1984, with its scroll bars for controls, I realized that the computer would make a great oscilloscope. The advantage would be that the data would end up in the computer where it could be immediately analyzed. No more digitizing of Polaroid photographs of HP or Tektronix screens.

Within a couple of years we developed MacScope I, which had an external box that acquired the data and shipped it over to the Macintosh. The external box had its own microprocessor which the Mac controlled. This system was very good at acquiring data down to the microvolt range, on time scales from weeks to milliseconds. Later we used the MacScope software with the Mac connected to a Hewlett Packard computer and could begin to grab data in the microsecond range. This is how we captured the microsecond scale data seen in the Physics2000 text.

There were two problems with the first MacScope. One was that our external hardware box sold for nearly $2000, and the equipment and interface became obsolete when the Mac went to USB.

The main change that led to MacScope II was the great improvement in the sound input of computers. Eight bit sound input appeared in the 1980s, but using that as an oscilloscope input gave very poor results. It is the 16 bit sound input that gives us the excellent results we can get with MacScope II.

Using the computer’s sound input capability eliminates the need for expensive external equipment. The cost of external equipment now ranges from $0 to $50. The disadvantage is that we are limited to the audio frequency range of 10 Hz to 2200 Hz.

The MacScope program is the collaborative effort of myself and Chris Sweeney. Chris has stayed up on the internal workings of Mac and Windows computers, and writes sample basic C++ code to handle the fast operations. I expand on Chris’s C++ code and then ask him what I did wrong. I then construct the user interface, all the controls and plotting using the program REALbasic. REALbasic immediately compiles for both Mac and Windows, and the people at REALbasic have been very helpful. Several years ago I wrote the field plotting program Charges2000 in order to learn REALBasic before applying that language to MacScope II.

E. R. Huggins
Dartmouth College
MACSCOPE II INSTRUCTION MANUAL

Cover............................................................ M:1
Author's Introduction ...................................... M:3
Contents........................................................ M:4
Instruction Manual ........................................... M:5
  1) Hardware................................................. M:5
  2) The MacScope Program......................... M:6
  3) Sound Input ........................................ M:7
  4) Observing Data ..................................... M:8
  5) Changing time scales............................. M:8
  6) Volt Scales ........................................... M:9
  7) Voltage Scroll Bar ................................. M:9
  8) THE (Record**) BUTTON....................... M:10
  9a) Time Offsets ...................................... M:10
  9b) Voltage Offsets .................................. M:11
 10) Stop and Store .................................... M:12
 11) Saving Stored Files.................... M:13
 12) Naming Stored files .......................... M:13
 13) Creating composite files.................. M:14
 14) Selecting Data ................................... M:15
 15) Fourier Analysis .............................. M:16
      Pulse Fourier Transform ........ M:17

Laboratory Work ........................................... M:18
  16) Voltage Range and Scales .... M:18
  17) Signal Averaging ...................... M:21
      Synapse Junction Potential ........ M:22
  18) Show Extended ............................... M:23
  19) Automatic Triggering ................. M:25
      Change + to – ................................. M:25
 20) Manual Triggering ......................... M:26
 21) Setting Trigger Level ................. M:26
 22) Stop on Trigger Mode .................. M:27
 23) The Voltage Columns ................. M:28
 24) Average Triggered Curves .......... M:29
      When to Use................................. M:29

New Features ................................................. M:30
  25) Math Waves & Signal Generator .... M:30
      Fourier Analysis of a Square Wave M:31
      Frequency Response ................. M:32
  26) Pulse Fourier Transform .............. M:33
  27) Details, Pulse Fourier Transform .... M:35
      Repeated Pulses ......................... M:35
      Creating the Pulse ................... M:35
      Creating Slits ......................... M:36
      Gaussian Pulse ....................... M:37
  28) Play Sound & Signal Generator .... M:38

Index ............................................................ I:1
  A - C......................................................... i:1
  D - F......................................................... i:2
  G - H......................................................... i:3
  I - M......................................................... i:4
  N - P......................................................... i:5
  Q - S......................................................... i:6
  T.......................................................... i:7
  U.......................................................... i:8
  V - Z......................................................... i:9
1) HARDWARE

MacScope® II is a software audio oscilloscope written to run on both Macintosh and Windows. The original MacScope, which we designed in the 1980s, worked only on Macs, and required a $2,000 external interface box. With MacScope II, we have considerably reduced external hardware costs, down to as low as $0.

Here we show different hardware inputs. The simplest to use is a built-in microphone input, if your computer has one. In that case no external hardware is required. For computers without a microphone input, you can use the Griffin iMic, shown in Figures 1.2 and 1.3, which converts a microphone signal to a USB output. In Figure 1.4 we show how to go from the BNC cable often found in labs, to the stereo input used by most computers and by the iMic.

Because MacScope II uses the computer’s audio input to grab experimental data, the frequency range is limited to the audio frequency range. Explicitly, computers and the iMic have an AC coupled input that has a low frequency cutoff around 10 Hz, and both digitize the voltage input at a rate of 44,000 points per second. Since the maximum frequency that can be detected requires at least 2 points per cycle, the upper frequency for these audio inputs is 22,000 Hz, which is called the Nyquist frequency. If you try to look for frequencies higher than that in the data, you get spurious results called aliasing.

![Figure 1.1](image1.png)

**Figure 1.1**
Some computers have a built-in microphone input. If you use that to record sounds, no external hardware is required.

![Figure 1.2](image2.png)

**Figure 1.2**
If your computer does not have an adequate sound input, you can use the $40 Griffin iMic that converts an audio voltage input to a USB output.

![Figure 1.3](image3.png)

**Figure 1.3**
The Griffin iMic has an unlabeled switch that changes the input from Line Voltage (1 to 2 volt range) to Mic Voltage (millivolt range). Keep the input voltage under 2 volts to prevent damage to the iMic. Set the switch to Mic Voltage, as shown, for the greatest sensitivity. (iMic available at www.griffintechnology.com.)

![Figure 1.4](image4.png)

**Figure 1.4**
Inexpensive connectors that allow us to get a signal from a BNC cable into both sides of the stereo input. (Connectors from www.audiogear.com.)
2) THE MACSCOPE PROGRAM

The two icons associated with the MacScope II are shown in Figure 2.1. The first represents the application itself. The other represents Sound Data Files that are created when you save curves that MacScope has grabbed. You can open MacScope by clicking on either icon, or dropping a Sound Data file icon onto the application icon.

To obtain the Macscope display seen in Figure 2.2, we double clicked on the file named Sound Data - Piano Notes.sdf that is included with the Acrobat version of these notes. (The .sdf extension stands for “sound data file”.) This file contains recordings of six different notes on the piano. The first, which we see in Figure 2.2, is middle c, recorded a few seconds after the piano key was struck. This is the note that was analyzed in detail in the accompanying article Teaching Fourier Analysis in Introductory Physics. It is the note that is lacking a seventh harmonic.

![Figure 2.1](image1.png)

MacScope icons. If you click on the application icon, the program opens ready to take data. Clicking on a Sound Data File opens MacScope with all the saved files ready for analysis. It is also ready to take more data.

![Figure 2.2](image2.png)

MacScope Data window and Tools window. The data, which had been saved in the Sound Data File, is a recording of middle c on a piano. Five other notes are also recorded in the same file.
3) SOUND INPUT

Before we start recording new data, we have to select one of the sound input devices discussed earlier in the hardware section. To make the selection, we go to the Input item in the MacScope menu, and select Sound Input Control as shown in Figure 3.1. What you get depends on the computer, and the computer system you are using.

We see the Sound Input Control panels for Mac OSX, and Mac OS(8 or 9) in Figure 3.2, and Windows in Figure 3.3. In all three panels we are given a choice of the available sound inputs. We chose the USB input, which is from the iMic shown in Figures 1.2 and 1.3.

Recording data with MacScope II is much like using a tape recorder. You should set the input gain or volume so that the input signal level is about 2/3s of the way up the level indicator bar as shown in the two Mac input control panels. If you set the gain too high, the tops and bottoms of your data will be clipped off. Set it too low and you lose sensitivity.

The arbitrary gain control built into computers affects the voltage scale in our plots. MacScope assumes that the maximum voltage range is ± 2 volts. But the actual range depends upon your input volume setting. (If you are using the iMic, the results depend greatly on where the line/mic voltage switch is set. Setting the switch to mic voltage amplifies the voltage by a factor of about 20.)

If the actual voltage values are not critical, and they are usually not for microphone data, then just consider the voltage scales as giving you the relative voltages. If the actual voltages are important, you can use the calibration method we will describe later.

![Figure 3.1](image1.png)

**Figure 3.1**
Selecting the Sound Input. This brings up the computer’s sound input control panel.

![Figure 3.2](image2.png)

**Figure 3.2**
Macintosh sound input panels, OSX above, OS(8 or 9) below. We set the input volume or gain so that the level indicator was about 2/3s the way up.

![Figure 3.3](image3.png)

**Figure 3.3**
Windows GX sound input panel. Clicking on “Volume” in the “Sound Recording” section gave us the gain scroll bar seen in Figure 16.5a.
4) OBSERVING DATA

Once you have selected a sound input, simply go to the Tools window and press the Record button. If you have a repetitive input signal, you automatically get a stable picture of that signal. The reason for this steadiness is that the computer looks through a broad section of data to find the highest point (maximum voltage), and plots that point at \( t = 0 \) on the plotting window. As a result, the computer plots the highest point of a repetitive curve at the same place each time and the curve looks stable.

Figure 4.1 shows the results of saying the vowel sound “oh” into a microphone attached to an iMic as shown in Figure 1.2. Because the microphone is sending almost the same signal to the two stereo inputs, the two plots, Curve A and Curve B are almost the same. On the right side, where a voltage scale will appear when we press Stop, we now see the value of the maximum voltage which was plotted at \( t = 0 \). The voltage scale will be in millivolts, and the time scale is in milliseconds.

5) CHANGING TIME SCALES

In Figure 4.1, it is not clear whether the vowel sound “oh” is repetitive or not. We can find out by changing the time scale of the plots. We do this using the Time Scale scroll bars in the Tools Window.

In Figure 5.1, we have changed the Curve A time display to 50 milliseconds (ms) from its original setting of 10 ms. We changed the Curve B setting to 20 ms, and got the results seen in Figure 5.2. Now it is quite clear that the signal is repetitive.

(Note that MacScope has the ability to display the same signal simultaneously on two different time scales.)

![Figure 5.1](image)

**Figure 5.1**

We used the time scale scroll bar to change the time scale on Curve A from 10 to 50 milliseconds.

![Figure 4.1](image)

**Figure 4.1**

Live plot of the vowel sound “Oh”. We get nearly the same plot in Curve A and Curve B because the microphone is sending nearly the same signal on both sides of the stereo input.

![Figure 5.2](image)

**Figure 5.2**

Live plot of the vowel sound “Oh” seen simultaneously at two different time scales.
6) VOLT SCALES

When MacScope is running, there are three modes for displaying the voltage scale. They are Auto Scale, Volt Scale-Tracking, and Volt Scale-Fixed, selected by the volt scale menu displayed in Figure 6.1.

In Auto Scale, the \( t = 0 \) voltage value, which is usually the maximum recent voltage value, is plotted one square down from the top, and the corresponding voltage value is printed on the right side of the plot. This has the advantage of starting each successive curve at the same place on the screen which makes the curves look steady. What can continually change is the printed voltage value.

If you want to see a voltage scale, you can select Voltage Scale-Tracking. In this mode, MacScope looks at the \( t = 0 \) voltage value and prints the curve using an appropriate voltage scale. For example if the \( t = 0 \) voltage were 88 millivolts, a 100 mV voltage scale would be used. If the \( t = 0 \) voltage rose to 122 millivolts, MacScope would switch to a 250 mV voltage scale. The advantage of the Volt Scale-Tracking mode is that you always have an appropriate voltage scale. The disadvantage is that the voltage scale can keep changing, which may be annoying.

If MacScope is running and you select Volt Scale-Fixed, MacScope selects a volt scale appropriate for the current \( t = 0 \) voltage value and stays there. The volt scale no longer hops around, but the data can go off scale.

When you stop, MacScope always switches to a fixed volt scale that is appropriate for the data being displayed.

7) VOLTAGE SCROLL BAR

You can manually change voltage scales by using the voltage scroll bar shown in Figure 7.1. If MacScope is running in either Auto Scale or Volt Scale-Tracking mode and you make any adjustment to the voltage scroll bar, MacScope automatically switches to the Volt Scale-Fixed mode. This allows you to adjust the voltage scale while MacScope is running.

In Auto Scale, the \( t = 0 \) voltage value, which is usually the maximum recent voltage value, is plotted one square down from the top, and the corresponding voltage value is printed on the right side of the plot. This has the advantage of starting each successive curve at the same place on the screen which makes the curves look steady. What can continually change is the printed voltage value.

If you want to see a voltage scale, you can select Voltage Scale-Tracking. In this mode, MacScope looks at the \( t = 0 \) voltage value and prints the curve using an appropriate voltage scale. For example if the \( t = 0 \) voltage were 88 millivolts, a 100 mV voltage scale would be used. If the \( t = 0 \) voltage rose to 122 millivolts, MacScope would switch to a 250 mV voltage scale. The advantage of the Volt Scale-Tracking mode is that you always have an appropriate voltage scale. The disadvantage is that the voltage scale can keep changing, which may be annoying.

If MacScope is running and you select Volt Scale-Fixed, MacScope selects a volt scale appropriate for the current \( t = 0 \) voltage value and stays there. The volt scale no longer hops around, but the data can go off scale.

When you stop, MacScope always switches to a fixed volt scale that is appropriate for the data being displayed.
8) THE (RECORD**) BUTTON
If you adjust the voltage scroll bar or make any other significant adjustment like changing the time scale, then when you stop recording, a new button labeled Record ** appears near the Record button. If you want to preserve these adjustments the next time you record, press the Record ** button instead of the Record button. The Record ** button also preserves any adjustments you make when MacScope is stopped, or any adjustments contained in a saved curve you have brought up.

The reason that there is both a Record and a Record** button is that it is easy to make adjustments that you do not want to see when recording again. When MacScope is stopped you may have amplified the voltage scale to look at a piece of low voltage data. If you preserve these settings when you record again, the voltages can be so far off scale that you cannot tell what has gone wrong.

When you press the regular Record button, you essentially do a factory reset to normal settings. The volt scale reverts to Auto Scale, the time scale reverts to 10 milliseconds, the starting time reverts to $t = 0$, and the voltage value $V = 0$ moves back to the center of the window. There are times when you want to preserve settings and times you do not. We give you the choice with these two buttons.

9A) TIME OFFSETS
Along the top of both Curve A and Curve B are horizontal scroll bars with a button labeled [T] on the left. These are Time Offset scroll bars that adjust where $t = 0$ is located. You adjust this scroll bar to look at earlier or later data, as shown in Figure 9-1.

![Figure 8.1](image1.png)
Figure 8.1
When you start recording using the Record** button, you preserve your current settings.

![Figure 9.1](image2.png)
Figure 9.1
Two clicks in the Time Offset scroll bar moves $t = 0$ two squares over so that we can see what happened there.

If you want to get back to having $t = 0$ at the beginning of the curve, press the button labeled [S]. We call this a reset button. We put reset buttons on almost all our scroll bars in order to quickly get back to some preferred value.

![Figure 9.1c](image3.png)
Figure 9.1c
Pressing the reset button moves $t = 0$ back to its normal position.
9B) VOLTAGE OFFSETS

On the left side of each curve is a vertical scroll bar with an reset button labeled [O]. This is the Voltage Offset scroll bar that moves the curve up and down. You can move the curve up until just the bottom tip of the curve is showing as seen in Figure 9-2, or down until just the top tips are visible. The reset button puts the zero of the voltage scale back in the center.

If you make an adjustment in either of these scroll bars for either window, the Record button will allow you to preserve the adjustments, while Record will perform resets.

Figure 9.2
How to study curves in detail. Suppose that we want to look very closely at the tops of the middle c curve seen in A). The first thing we do is to use the offset scroll bar to move the curve down as shown in B), so that the tops of the curve are at the center of the plotting window. Then when we use the voltage scroll bar to amplify the curve, the tops will remain at the center of the window, and we can get an extremely detailed picture of the curve tops.
10) STOP AND STORE

When you press the Stop button, the current data (up to 177,000 points) is stored in RAM along with previously stored data. As seen in Figure 10.1, the name of the window has changed from Scope to Curve #7. It is Curve #7 because we got six data files when we opened MacScope by double clicking on the “Sound Data - Piano Notes.sdf” file shown in Figure 2.1. That sound data file (.sdf) already contained six files, so that our “Oh” vowel sound file becomes a seventh.

There are several ways to see what data files are stored and ready for viewing. One way, that gives you a complete summary of available files, is to go to Return To in the Edit Menu as shown in Figure 10.2.

To go rapidly between stored files, you can use the Store Select scroll bar underneath the Record button. In Figure 10.3, we started off with Curve #7. Clicking once in the left arrow brings up Curve #6, Octave above a. Moving the thumb all the way to the left brings up the first data file Curve #1 Mid c later.

Figure 10.1
Stored plot of the vowel sound “Oh”. When you press Stop, the most convenient near by volt scale appears, and the curve is adjusted to that scale. Each time you record and then press Stop a new curve is stored in RAM (but it is not yet saved to your hard drive).

Figure 10.2
The Return To menu shows you all the stored files. Up to 25 files can be stored in RAM. If you store a 26th file, it replaces Curve #1 etc. If you press Stop and do not like the results, you can get rid of that file by selecting Unstore Current. (Restore Next puts it back.) Undo takes you to the previous file, and Redo to the next file.

Figure 10.3
Using the Store Select scroll bar to rapidly go between stored files.
11) SAVING STORED FILES
The stored files are stored in RAM, the Random Access Memory that disappears when you shut your computer off. Worse yet, these files disappear when you Quit MacScope. You lose them unless you Save them on your hard drive, a CD, or some other storage media.

MacScope gives you the two options for saving data files seen in Figure 11.1. The option Save All Data As... in the File menu, saves all currently stored data files as a single file on your hard drive. The file Sound Data - Piano Notes.sdf that we double clicked back in section 2, was created this way. When you open such a file, the curves are added to any you may have currently stored. (For example, if we already had the seven files shown in Figure 6.2, and we used Open Data File(s) to open the piano notes file again, we would end up with 7 + 6 = 13 stored files.)

With Save Current Data in the File menu, you also have the option of saving just the currently displayed data file. This is useful if you have taken a lot of data, but only one or two of the files are worth saving.

The option Save Plot in the file menu merely saves a picture of your current data window. It is essentially a screen dump of that window. It is not nearly as useful as saving the data, because you cannot manipulate the data later on. If you are working in the lab, we recommend that you save data files, and create plots only when needed.

12) NAMING STORED FILES
The name “Data #7” is not the ideal name for our vowel sound “Oh” data file. You can see in Figure 10.2 that the other data files have names like “Mid c later” and “Mid c trig” which tell us something about what is in the data file. To add a name to a data file, we use the Add Window Label command in the File menu.

When we make that choice, we get the window shown in Figure 12.1 that allows us to type in a name. After selecting Done, the new window name appears in three places, as the name of the data window (Figure 12.2a), under the store select scroll bar (Figure 12.2b), and in the Return to menu (Figure 12.2c).

![Figure 11.1](image1.png)
Using the File Menu to save data files.

![Figure 12.1](image2.png)
Entering a name for Data #7.

![Figure 12.2](image3.png)
The new name appears in three places.
13) CREATING COMPOSITE FILES

At a workshop, one of the participants suggested that he should be able to compare files, by putting one file in Curve A, and a different file in Curve B. We have implemented this feature with the Create Composite File command in the File menu.

We will illustrate this capability by creating a composite file with “Data #1 Mid c later” in Curve A and “Data #2 Mid c trig” in Curve B. The result will become Data #8.

We begin constructing the composite file by first making “Data #1 Mid c later” the current file being displayed in the data window. Then we go to the File menu and select Create Composite File and then choose Replacing Curve B as seen in Figure 13.1.

That brings up the “Replace Curve B” window shown in Figure 13.2. We have moved the scroll bar so that we will be replacing Curve B with Data #2 of Curve A. The result is the composite curve shown in Figure 13.3. If we do not like the long name the computer gave the file, we can change it using the Add Window Label command in the File Menu.

![Figure 13.1](image1.png)
After getting the Curve A file we wanted, we choose to replace Curve B.

![Figure 13.2](image2.png)
Choosing to replace Curve B with the file “A Data #2”.

![Figure 13.3](image3.png)
The resulting composite file, consisting of “Mid c later” in Curve A and “Mid c trig” in Curve B. This is the composite file we used in the accompanying article “Teaching Fourier Analysis in Introductory Physics”.

We begin constructing the composite file by first making “Data #1 Mid c later” the current file being displayed in the data window. Then we go to the File menu and select Create Composite File and then choose Replacing Curve B as seen in Figure 13.1.

That brings up the “Replace Curve B” window shown in Figure 13.2. We have moved the scroll bar so that we will be replacing Curve B with Data #2 of Curve A. The result is the composite curve shown in Figure 13.3. If we do not like the long name the computer gave the file, we can change it using the Add Window Label command in the File Menu.
14) SELECTING DATA

The main advantage of using the computer as an oscilloscope is that the data is immediately available for analysis. For example, the curve “Mid c later” which is Curve A of our composite file, is the middle c piano note recorded a short while after the key was pressed. Noting that the curve is repetitive, we can use the selection feature of MacScope to measure the period and frequency of the note.

In Figure 14.1, we are selecting one cycle of the curve by dragging the cursor over the curve. As we make the selection, a new window appears, a window labeled “Curve A values” seen in Figure 14.2. This window tells us that the period \( t \) of the selected data is \( t = 3.9 \) milliseconds (ms), and the corresponding frequency \( f = \frac{1}{t} \) is \( f = 256.7 \) cycles per second (Hz).

Checking on the web, we found that the middle c frequency should be 262 Hz, which means that our piano is slightly out of tune.

To help position the selection rectangle, we have the feature that if you hold down the shift key while you move the cursor, the selection rectangle moves as a unit with the cursor, as shown in Figure 14.3. This helps you to precisely position the starting position of the selection rectangle. You can then release the shift key and adjust the ending point of the selection rectangle.

Once you release the mouse button, the selected section of the curve is highlighted as shown in Figure 14.4. If you press the button labeled Expand, as we are doing in Figure 14.4, the selected section of curve expands to fill the entire window, as seen in Figure 14.5. This gives us a detailed view of the selected section of data.

Figure 14.1
Selecting a section of the curve.

Figure 14.2
Data window for Curve A selection, showing Curve A values.

Figure 14.3
Hold down the shift key while dragging and the entire selection rectangle moves with the cursor.

Figure 14.4
The selected section of data is highlighted. Here we are pressing the Expand button for a closer look.

Figure 14.5
Selected data expands to fill window.
15) FOURIER ANALYSIS

One of our main aims in writing the MacScope program was to make it easy to do Fourier analysis of experimental data. The accompanying article “Teaching Fourier Analysis in Introductory Physics” contains a description of not only how to use MacScope to do Fourier analysis, it also shows the mathematics used by MacScope to do the analysis. Here we will briefly summarize some of the results of that article.

To get Figure 15.1, we pressed the Fourier button instead of the Expand button. Curve A data expanded as it did in Figure 14.5, but now the Curve B window is covered by a graph of the harmonics contained in the expanded data.

In Figure 15.2, we clicked on the bar representing the first harmonic, and we see that harmonic superimposed on our expanded data. In Figure 15.3, we selected both the first and second harmonics and see the sum of these two harmonics superimposed on the expanded data. This sum has a few of the general features of the data. When we select harmonics 1, 2, and 4, the sum is quite close to the data.

To choose Fourier analysis, you must first select a section of the experimental data. We made this requirement because the mathematics makes the assumption that the selected section of data repeats indefinitely. If by accident you select data that does not repeat, spurious harmonics appear as shown in Figure 10 of the Teaching Fourier Analysis article.

Figure 15.1
Harmonics contained in the selected data.

Figure 15.2
First harmonic superimposed on the data.

Figure 15.3
Sum of the first and second harmonics.

Figure 15.4
We get close adding the three biggest harmonics.
**Pulse Fourier Transform**

A unique feature we included in MacScope is the Pulse Fourier Transform that allows us to study the harmonics contained in a pulse. Here we show the harmonics in a short one cycle pulse. The reason for studying the harmonic structure of a pulse is that the analysis leads directly to an understanding of the energy-time form of the uncertainty principle, as described in the accompanying article “Teaching the Uncertainty Principle in Introductory Physics.”

**Figure 15.5 Creating a pulse**

We can create a one cycle pulse by selecting one cycle of a sine wave as shown in (a), and then choosing the Pulse Fourier Transform from the file menu as in (b). In (c), we see that a pulse contains many harmonics.

**Figs. 15.6. Fourier analysis of a pulse.**

Here we see how a short pulse is constructed from long sinusoidal waves. In (a) we selected the largest harmonic and all it represents is a small sine wave. When we add together the five biggest harmonics in (b), a pulse begins to form. When we add up the 32 biggest harmonics, we get a close representation of the pulse in (d). We need a lot of harmonics to cancel the wave outside the pulse.
16) VOLTAGE RANGE AND VOLTAGE SCALES

When we are using MacScope for studying sounds, using either a Wal-Mart type microphone or a computer built-in microphone, we are not likely to damage equipment and we do not really care about the voltage scales on the MacScope windows. This changes, however, when we start plugging other laboratory equipment into MacScope. Then we have to be careful about the voltages we are using, and to get accurate voltage scales we have to calibrate MacScope.

Voltage Range

In the past, a standard calibration signal, produced either by an oscilloscope or signal generator, was a 5 volt amplitude square wave. If you put that signal into an iMic, you could destroy it (we have done that). Perhaps this could also destroy the sound input of a computer (we have not done that). The reason is that the iMic and computer inputs are designed to work in the 0 to 2 volt range, and larger voltages can be damaging. If you wish to study a signal whose voltage range is over a volt, you should use something like the voltage divider circuit shown in Fig. 15.1 to bring the voltage down to about the 100 millivolt (.1 volt) range. Using that circuit, the potentially damaging 5 volt signal is reduced to .05 volts, or 50 millivolts, which is in the ideal range for the computer inputs.

Voltage Scale

The voltage scale you have been seeing on the MacScope windows is based on the assumption that the full scale input into the computer is in the zero to two volt amplitude range. That is what the computer manuals say, but it is not true in practice because various stages of amplification are inserted before the data gets to the computer.

To see what the different input devices did to our voltage signals, we took an inexpensive sine wave generator, attached it to a standard calibrated oscilloscope, and adjusted the signal to an amplitude of 100 millivolts. This gave us a calibrated 100 mV sine wave signal that we plugged into six different kinds of MacScope inputs. We got four different results which demonstrates the necessity of using MacScope’s calibration feature if the voltage scale is important.

Computer sound inputs can be divided into two general categories, line input and microphone input. The iMic, shown in Fig. 1.3 has a switch on the front that switches between these two inputs. The Macintosh G4 computer we are using has an external microphone input, while our PowerBook has an external line input. All these inputs give different results. The only thing consistent was that the iMic values were the same for both Mac and Windows when the Mac amplification was turned all the way down. (More about Mac amplification later.)

Figure 1.3
Switch on the iMic.

From Figure 16.1
Voltage divider circuit. If your input voltage exceeds about one volt in amplitude, you should use a voltage divider to protect your equipment. With the above circuit, the output voltage is reduced by a factor of one hundred. If you replace the $10^6$ ohm resistor by a $10^5$ ohm resistor, the voltage is reduced by a factor of one thousand.
The simplest way to see what these different inputs are doing is to look at what they do to our 100 mV signal.

<table>
<thead>
<tr>
<th>Input</th>
<th>Uncalibrated Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>iMic microphone input</td>
<td>510 mV</td>
</tr>
<tr>
<td>iMic line input</td>
<td>27 mV</td>
</tr>
<tr>
<td>G4 external microphone</td>
<td>128 mV</td>
</tr>
<tr>
<td>PowerBook line in</td>
<td>15 mV</td>
</tr>
</tbody>
</table>

From this you can see that there is a general trend that microphone inputs amplify the input signal while line inputs reduce the signal. But there is clearly no standardization.

What we have done in the MacScope software is add a calibration routine that corrects the input signal so that the voltage scale reads 100 mV when we put in a 100 mV signal. It can reduce the 520 mV signal or amplify the 15 mV signal to 100 mV.

We get the calibration window by going to the input menu and selecting Calibrate as shown in Fig. 16.2. If you want the same calibration for both curve A and curve B, then select Calibrate as shown. If we want separate calibration factors for the two windows, use Calibrate A and B.

![Figure 16.2](image)
Selecting Calibrate.

![Figure 16.3](image)
Uncalibrated result of a 100 millivolt signal into the iMic microphone input using a Windows computer.
The calibration routine works while MacScope is running. You change the scroll bar in the calibration window until the voltage scale in the scope window has the expected value. For example, in Fig. 16.3 we have put our 100 millivolt sine wave into an iMic attached to a Windows computer and set the line/microphone switch on the iMic to the microphone setting. We see that the voltage scale reads .5 volts (510 millivolts), a factor of 5.1 times too high. In Fig. 16.4, we moved the calibrate scroll bar to the left to get an amplification factor of .199 which reduces the voltage reading to the desired 100 millivolts.

When we calibrate on the Macintosh, there is an extra amplification to consider. As seen in Fig. 3.2, the Mac sound input window itself has an amplification scroll bar that is called Level on Mac OS9 and Input Volume on OSX. If the thumbs of these scroll bars are set all the way to the left, and we send our 100 millivolt signal in through an iMic, we get the same values on the Mac as we got on Windows.

Our Windows 98 machine has no Input Volume scroll bar but the Windows GX machine does. To get at the GX scroll bar, shown in Figure 16.5a, click on “Volume” in the sound recording section of the Audio Input Control Panel.

![Figure 16.5](image)
The **Input Volume** scroll bar on the Mac OSX sound input window.

![Figure 16.5a](image)
The **Input Volume** scroll bar obtained by clicking on “volume” in the Windows XP sound input window.
When using the Mac, it would be convenient if the range of the Mac amplifier were great enough so that we did not have to use the MacScope Calibrate control. In one case we did this by using the iMic set to the line input and using the Mac amplifier to boost the voltage from 27 mV up to the desired 100 mV. The result is seen in Figure 16.6. However, the MacScope calibrate scroll bar is more sensitive than the Mac one, and you may want to use MacScope’s calibration anyway.

When we sent our 100 mV signal directly into our G4 External Microphone Input in order to avoid the iMic, this left us with a 128 mV reading at the lowest Mac amplifier setting. To get the voltage down to the desired 100 mV, we would then have to have used the MacScope Calibrate feature.

17) SIGNAL AVERAGING

In real world laboratory situations, the data we deal with can be inherently noisy, either due to the nature of the object being studied, or to ambient electrical noise in the lab. This is particularly true of biological experiments where the phenomenon being studied continually varies from measurement to measurement, and it is impossible to completely shield the apparatus from outside noise like the 60 cycle radiation generated by the electrical wiring in the building. The cure for both of these problems is signal averaging.

![Figure 16.6](image)

Using the Macintosh Input Volume amplifier to calibrate MacScope. Because this amplifier only amplifies, we could not use it to reduce the 128 millivolts we got when our 100 mV signal was plugged into our G4 Sound In port. Here we amplified the 27 mV signal from the iMic Line input.
Synapse Junction Potential

We will illustrate the effectiveness of signal averaging with the measurement of the synaptic potential produced in a muscle fiber when an action potential pulse arrives from an attached neuron. (See Figure 17.1.) The results are shown in Fig. 17.2. The spike on the left is the action potential pulse. Then there is about a ten millisecond delay as chemicals cross the synaptic gap between the neuron and the muscle fiber. After crossing the gap, the chemicals cause the depolarizing potential which is seen as the fairly rapid voltage rise followed by a relatively slow decay.

Curve A in Figure 17.2 is a signal averaged curve. Before signal averaging, the data looked like that shown in Curve B. In this figure, Curve A is the average of 152 sections of raw data.

The way the experiment is done is that an electronic device called a stimulator is attached to the neuron and for this experiment sends out a voltage pulse every tenth of a second (every 100 milliseconds). This stimulator pulse causes the neuron to send an action potential pulse to the neuron-muscle fiber synapse. As we can see in Fig. 17.2, the neuron action potential pulse arrives every 100 milliseconds.

Figure 17.1
Neuron attached to a muscle fiber. (Figure adapted from Essentials of Neural Science Behavior by Kandel, Schwartz and Jessell.)

Figure 17.2
MacScope can simultaneously display the signal averaged data (Curve A) and the raw data that is being averaged (Curve B). In the Tools Window averaging section, for Curve A we selected Average Until Stop, while for Curve B averaging we selected None.
A special feature of MacScope is that if you send the same raw data into Curve A and Curve B, you can simultaneously observe signal averaged data and raw data as we did in Fig. 17.2. In the Tools Window, remove the covers over the advanced controls (by clicking on them) and you see the signal averaging control underneath. For curve A we selected Averaging Until Stop. That means that every cycle is included in the average until we press the Stop Button. For curve B we selected None for signal averaging. With that setup we can watch the signal averaged curve smooth out while simultaneously checking that the instantaneous data is still valid.

Instead of averaging Until Stop, you can select to average a given number of curves, as shown in Figure 17.3. For example, if you select 10 curves, then the display will show the average of essentially the last 10 curves recorded. This allows you to observe smoothed data that is slowly changing. The more curves you average, the slower the changes will be.

**18) SHOW EXTENDED**

Once you remove the cover over the advanced controls, three new features appear. One is Signal Averaging which we have just discussed. Another is the Trigger Control that we will discuss in the next section. Here we will describe the Show Extended feature.

When recording, MacScope maintains two separate banks of data. One bank is the raw data which is 177,000 points long and is constantly refilled as sound data comes into the computer. The second data bank, 2,200 points long, holds the current signal averaged data. During recording, MacScope displays from the shorter signal averaged bank of data. When you stop, you can look at the much larger raw data bank by clicking on Show Extended. This allows you to scroll through all 177,000 points, which is just over 400 windows of data. (If you do not select signal averaging, then the signal average data bank simply holds the last 2200 points of raw data.)

When you scroll through the extended data, the time scale is often not useful. To get Fig. 18.2, we started with the smoothed curve A of Fig. 17.2 and clicked on Show Extended to see the same raw data as being displayed in curve B. Then we scrolled back 32 seconds to see the data 160 windows earlier.

If we are interested in the structure of this earlier data, having the time scale start at –32.2164 seconds and stops just beyond –32.0564 seconds, is not particularly convenient. You can zero the time scale and have it go back to milliseconds, by pressing the little button labeled “Z” for zero, as shown in Fig. 18.3. Now you can easily see that the cycles repeat at 100 millisecond intervals.
Figure 18.2
Showing extended data in Curve A. When we click on the Show Extended button, we have access to 400 windows of data, representing the latest unaveraged data. Here we have scrolled back to 32 seconds earlier than the data being displayed in Curve B.

Figure 18.3
Zeroing the time scale for Curve A. When you press the Z button, you temporarily set the time scale to start at zero. That makes it much easier to see that the spikes are still 100 microseconds apart.
19) AUTOMATIC TRIGGERING

The most convenient feature of MacScope is that in most circumstances it is automatically triggered. The data from the synapse potential experiment is an excellent example of data that is well handled by automatic triggering.

In Fig. 19.1 we are taking a close look at the raw data from one cycle of that experiment. From this picture we clearly see that the sharp neuron action potential spike on the left has the highest positive voltage in the cycle. To draw the curve on the screen, the computer looks for the highest voltage in the next block of data and displays the highest voltage at the t = 0 position on the screen. As a result for our synapse data the action potential spike will be located at t = 0.

Sometimes you run across a nerve where the action potential spike is not as high as the resulting muscle voltage. If this happens then the noisy bump at point (a) in Fig. 19.1 would be the highest voltage, and the computer would move the curve over so that point (a) was located at t = 0. If sometimes the action potential spike was higher than point (a), and sometimes lower, then the curve would flip back and forth and we would say that the curve was not well triggered. This can be remedied using the manual triggering to be discussed shortly.

**Change + to –**

Another way you can get improper triggering for this experiment is to reverse signal and ground wires so that the action potential spike is down, rather than up. Since MacScope only looks for maximum positive values, it would completely miss the action potential spike. Then the curve would move all over the place, triggered by some noise at the end of the cycle. This problem is quickly remedied by going to the Input Menu and selecting Change + to –. This flips the curve over which is equivalent to reversing the connections to the + and – terminals.

![Input Menu](image)

**Figure 19.2**

Selecting [Change + to –] turns the curve over. It has the same effect as reversing the positive and ground terminals.

![Figure 19.1](image)

**Figure 19.1**

With **Automatic Triggering**, the computer looks for the highest point in the next block of data, and plots that at t = 0. Here the high action potential spike is plotted at t = 0.
20) MANUAL TRIGGERING
When automatic triggering fails or does not do what you want it to do, you can use manual triggering where the curve is triggered if the voltage exceeds a trigger voltage level that you set.

In MacScope II, there are two separate trigger modes. One is Stop on Trigger and the other Average Triggered Curves. In Stop on Trigger, the computer stops recording shortly after the first time it detects that the voltage rises above the trigger voltage. It displays the triggered data with \( t = 0 \) at the trigger point. The Stop on Trigger mode is most useful for capturing one-time events like the ring of a bell.

The Average Triggered Curves mode allows you to signal average those curves that reach the trigger voltage level, while ignoring curves that do not reach the trigger level. This is convenient when you want to signal average data that is triggered only once in a while and you want to ignore the noise between triggers.

We will discuss more about these two modes after the next section where we describe how to set the trigger voltage level.

21) SETTING TRIGGER LEVEL
Each curve, Curve A and Curve B has its own trigger level setting area. The area for Curve A is shown in Figure (21-1). To set the trigger, the first step is to click on the Trigger A button to enable manual triggering for Curve A. The next step is to use the vertical scroll bar to do a gross adjustment of the trigger voltage. The allowed values are 0 to 2 volts in steps of 20 millivolts. The two sets of small arrows allow fine scale adjustments, the first set in steps of 1 millivolt and the second set in steps of .1 millivolts (100 microvolts).

![Figure 21.1](image)
The trigger control panel for Curve A.

![Figure 21.2](image)
Changing the trigger voltage. One click in the down arrow of the main scroll bar (a) drops the trigger 20 millivolts, from 240 mV to 220 mV. Each click in the first small arrows (b) changes the trigger voltage by one millivolt, and each click in the second small arrows (c) changes the trigger voltage by a tenth of a millivolt.

With these three controls, we can set the trigger level anywhere in the range of 0 to 2 volts, to an accuracy of .1 millivolts.
22) STOP ON TRIGGER MODE

In the Stop on Trigger mode, MacScope stops recording shortly after it first encounters a trigger, i.e., the first time the incoming voltage exceeds the trigger voltage. MacScope then discards any old data that may be in the averaging buffer and displays the triggered data. The average data display now contains one window of pretrigger data (before \( t = 0 \)) and four windows of data after the trigger (after \( t = 0 \)). If you want to see still earlier data, choose Show Extended to view the extended data. In this mode, no signal averaged data is saved once we get a trigger.

As we mentioned, the Stop on Trigger mode is most useful for recording one time event like the ring of a bell or the call of a frog. The pretrigger window shows you how the event got started.

![Sound of a wooden frog.](image)

**Figure 22.1**
To get this frog to talk, you hold the frog by its front legs, heading away from you, and stroke its back with its mallet. We captured the sound it makes using Stop on Trigger. (Frog purchased from Educational Innovations Inc, www.teachersource.com.)

![Sound of a wooden frog.](image)

**Figure 22.2**
Sound of a wooden frog.
23) THE VOLTAGE COLUMNS

When you click on the Trigger A button you will notice a narrow vertical column appears at the left side of Curve A window, as shown in Figure 23.1. The top of the column is blue and the bottom is red. The dividing line indicates the value of the trigger voltage, in the range 0 to 2 volts. For example, if you set the trigger voltage to 1 volt, the dividing line will be half way up, as in Figure 23.1b. As you change the trigger voltage, you will see the dividing line move up and down.

When MacScope is running, there is another vertical column colored red whose height changes. This voltage column, seen in Figures 23.2 indicates, on a two volt scale, the maximum voltage in the current window being displayed.

When you are using manual triggering and MacScope is recording, you can tell how close the current data is coming to the trigger level by comparing the two voltage columns. In Figure 23.2a, the trigger level column is set at .400 volts and the voltage column is slightly shorter because the sine wave has an amplitude slightly under .4 volts. In Figure 23.2b, the sine wave amplitude exceeds .4 volts, the voltage column is higher than the trigger level column, and we get a triggered curve, which is colored blue.

![Figure 23.1](image1)

The trigger voltage column on the left shows the selected trigger voltage on a scale of 0 to 2 volts. At 240 mV, the column is 1/8 the way up, and half way up at 1 volt.

![Figure 23.2](image2)

Comparing the trigger level and voltage level columns.
24) AVERAGE TRIGGERED CURVES

As we mentioned, in the Average Triggered Curves mode MacScope signal averages only curves which reach the trigger level, ignoring those that do not reach this level. All triggered curves are displayed with $t = 0$ at the trigger point which, most of the time, is the point where that curve first reaches the trigger level.

If you are running in Average Triggered Curves mode and have not yet gotten a trigger, the scope runs normally, looking like the automatic trigger mode. Once you get a trigger, the triggered curve is plotted in blue and the averaged data from non-triggered curves is discarded. From that point on, the display changes only when a new triggered curve is averaged in. If you select None for signal averaging, then the individual triggered curves will be displayed one after the other.

When to Use Average Triggered Curves

There are two main uses of the Average Triggered Curves mode. One is where the trigger occurs infrequently and you want to avoid averaging in the noise between windows.

If your curves are triggered at a regular time interval, and you can set the time scale so that a possible trigger occurs regularly within one or two windows of data, then the automatic trigger will not add in untriggered curves and you do not need to use a triggering mode.

There are cases, even when the triggering occurs regularly, that you may need to use the trigger mode. Consider Figure 24-1, a repeat of Figure 18.2 where the junction potential curve is triggered by the action potential spike. In this case, the action potential spikes at $t = 0$ is higher than the top of the next junction potential curve, but the spike at $t = 100$ is lower than the following junction potential.

If this change happens fairly often, which it can for some nerves, and we trigger on the highest point as we do with automatic triggering, the trigger point will move back and forth between the spike and the curve. On the screen you will see an unaveraged curve move back and forth and the signal averaged curve will not be accurate. To see if this is happening, it is good to look at both a signal averaged curve and a non-averaged one at the same time. If the non-averaged one is hopping around you should fix the problem before trusting the averaged one. You can fix this problem by using the Average Triggered Curves mode and making a good choice of the trigger level.

To set the trigger level, first stop the recording so that you have a couple of good examples of curves you want to trigger, as we did in Figure 24.1. You may want to look through the extended data to find good examples. Then choose a trigger level that is high enough to avoid most of the noise, but low enough to always catch the action potential spike. Because the action potential spike always comes just before the junction potential, the curve will reach the trigger level at the action potential first, and that will almost always be the trigger point.

Once you have adjusted the trigger level to the height you want (I would start with 1 millivolt for Figure 24.1), check that you have signal averaging, choose Average Triggered Curves, and then press Record to preserve these settings. If you make a mistake and press Record, you will have to select Averaging again. You can change the trigger level while the program is running to search for the best trigger level. If you do that, stop and start again to get rid of unwanted data.

Figure 24.1

Automatic Triggering would not work here because the action potential spike is not always higher than the following synapse potential. Here Manual Triggering is needed.
25) MATH WAVES AND SIGNAL GENERATOR

It is easy to get a sine wave into MacScope, just whistle into a microphone. This is how we made the sine waves we used in our discussion of the uncertainty principle.

Because of MacScope’s Fourier analysis capability, it is worthwhile getting other kinds of wave structures into MacScope so that one can study their harmonic structure. In a traditional electronics lab this would be done by recording the output of a device called a signal generator. A typical signal generator outputs sine waves, square waves and triangle waves at frequencies and amplitudes which you can dial.

What we have done with MacScope is to include math waves, so that MacScope can create its own waveforms. Then the fact that MacScope can play selected harmonics or repeated sections of a curve, allows MacScope to act as a signal generator, as well as an oscilloscope. And the math waves are immediately available for analysis.

The math waves we have included are the sine wave, square wave, rectangle wave, triangle wave, and two ramp or sawtooth waves, all shown in Fig.(25.1). The square wave alternates between two voltage levels, spending equal lengths of time at each level. The rectangle wave can spend different lengths of time at the upper and lower voltage levels. We introduce the rectangle wave so that we could have a wave pattern that matched the slit structures used in our study of Fourier optics.

To get the waveforms shown in Fig.(25.1), first go to the File menu and select Create Math Wave as shown in Fig.(25.2). This brings up the Create Math Wave window shown in Fig.(25.3). The first thing you do is select the frequency you want for the wave, and then use the drop down menu to select the kind of wave that you want. In Fig.(25.3) we have selected a frequency of 440 Hz and will be choosing a square wave. (If we had chosen a sine wave and played the result, we would have A above middle C.) The resulting square wave is shown in Fig.(25.4).

For the rectangle wave there is an additional step. After selecting the rectangle wave, you get the additional window shown in Fig.(25.5) asking you to choose the ratio of the width of the tops, to the widths of the bottoms. In the background of that figure is a 440 Hz rectangle wave whose tops are 33% of one cycle.
**Fourier Analysis of a Square Wave**

In textbooks that introduce Fourier analysis, it is traditional to use a square wave as the first example to show how a complex waveform can be constructed from a harmonic series of sine waves. This is a bit ironic, because a square wave is discontinuous at each jump, while Fourier’s theorem tells us that any continuous wave shape can be constructed out of harmonic sine waves. The result is that when you try to construct an ideal square wave from sine waves, you end up with a small blip at the discontinuity, a blip that will not go away as you add more and more harmonics. This blip is called the **Gibbs effect**.

Despite the blip, looking at the harmonic structure of a square wave can be quite instructive. In Fig.(25.6) we are selecting precisely one cycle of a square wave. Pressing the **Fourier** button we get the analysis shown in Fig.(25.7). The first thing we notice is that all the even harmonics are missing. The square wave is composed entirely of odd harmonics 1, 3, 5, etc.

There is also a smooth progression in the way the amplitudes of the harmonics decreases. If you look closely, you will see that the 3rd harmonic has 1/3 the amplitude of the first; the 5th harmonic has 1/5 the amplitude of the first, and so on. You can see this result more clearly if you click on the button labeled **FT Data** and look at the numerical values of the harmonics.

In Fig.(25.8) we look at how the square wave can be reconstructed from its harmonics. The first harmonic is clearly the sine wave that most closely fits into the square wave. The second harmonic, when added in, lowers the peaks in the sine wave. As we go up through harmonic 13, it is clear that each harmonic added in brings the composite wave closer to a square wave.

---

**Fig. 25.6.** Selecting one cycle of a square wave.

**Fig. 25.7.** The square wave has only odd harmonics.

**Fig. 25.8.** Building up a square wave from its harmonics.
Frequency Response

In Fig.(25.9), we added all harmonics up to #50. We chose to add the first 50 harmonics for the following reason. With our basic frequency at 440 Hz, the 50th harmonic has a frequency of 50*440 = 22,000 Hz. But 22,000 Hz is the maximum frequency that can be recorded by the computer’s audio input. (The computer records 44,000 points per second, and you need at least 2 points per cycle to recognize a frequency.)

As a result, if you tried to record a perfect square wave with MacScope, or any audio computer input, no frequency component above 22,000 Hz would be recorded, and the result would look like Fig.(25.9).

When we were first testing MacScope, we had our electronics technician build a 440 Hz square wave generator. When we looked at the output of the generator on his high frequency lab scope, we saw a smooth square wave. We then took his square wave generator home and looked at its output on MacScope, and saw bumps at every transition. At first we wondered what had gone wrong. Why couldn’t MacScope give as good a plot of the square wave as the technician’s lab scope?

We took the square wave generator back to the technician to see if it had been damaged on the way home. The answer was NO—the generator still produced a clean square wave on his lab oscilloscope. It slowly began to dawn that MacScope’s limited frequency range was the problem. In effect, MacScope was showing only the first 50 harmonics of the square wave being put out by the square wave generator.

Standard computer audio inputs cut off at both the high and low frequency ends. By sampling the voltage 44,000 times a second, no frequencies higher than 22,000 Hz can be recorded. (This maximum frequency for sampled data is called the Niquist frequency.) There is also an electronic filter that prevents frequencies below about 20 Hz from being recorded. (The result is that MacScope is called an AC oscilloscope, i.e., one that records only “alternating current” signals and ignores low frequency voltage changes.)

We had our electronics technician drop the frequency of our square wave generator from 440 Hz to 60 Hz, and got the result shown in Fig.(25.10). The upper window shows three full cycles of the wave, while the lower window shows one cycle in more detail. (1/60 sec = 16 2/3 milliseconds.) In these diagrams we see the effects of both the high and low frequency cutoffs. The little bumps due to the high frequency cutoff are still there, while the sags in voltage at the top and bottom of the square wave are due to the low frequency cutoff.

The lesson we learned was that we were not going to get a good square wave into MacScope through the frequency cutoffs used by computer sound inputs. That is when we decided to create square waves inside MacScope with a Math Wave option.

![Fig. 25.9](image1) Selecting harmonics up to a frequency of 22,000 Hz, the computer’s Niquist frequency.

![Fig. 25.10](image2) Recording of a 60 Hz square wave. The low frequency cutoff causes the curve to sag, the high frequency cutoff causes the blips.
26) PULSE FOURIER TRANSFORM

To do a Fourier transform with MacScope, you first select a repetitive section of a curve and press the Fourier button. MacScope expands that section of the curve to fill the display window and then calculates what harmonics are needed to reconstruct the curve out of sine and cosine waves. The plot displayed in the Fourier transform window shows the amplitudes of the required sinusoidal waves.

An explicit mathematical assumption is that the expanded wave pattern you see in the display window repeats forever. This is a mathematical requirement because the sine and cosine waves out of which you are reconstructing your curve are, themselves, infinitely long.

When we say that a sine wave is infinitely long, we mean that if you chop off a sine wave at a finite length, it is no longer a pure sine wave. To see what a chopped off sine wave actually is, we have introduced the pulsed Fourier transform into MacScope.

As we mentioned, when you press the Fourier button, the selected section of curve expands tofill the display window. If, instead, you press the Pulse FT button, the selected section of curve stays where it is, and the rest of the curve is zeroed.

As an example, in Fig.(26.1a) we used Math Wave to create a sine wave. (We could get nearly the same result whistling into a microphone.) We then selected one cycle of the sine wave near the center of the window.

To get Fig.(26.1b), we pressed the Fourier button. That one cycle of the sine wave expanded to fill the display window. If that one cycle were repeated indefinitely, we would get a pure sine wave, which is indicated by the fact that we see there is only one harmonic present.

To get Fig.(26.2c) we pressed the Pulse FT button instead. The non selected part of the curve was zeroed, and the selected cycle remained as a pulse. You may think of this one cycle as a sine wave, but that is not what the Fourier analysis window is telling us. We see that the chopped off sine wave is a complex mixture of many waves. It takes a lot of sine waves to make up a short pulse.
To see how we build up a pulse out of sine waves, we have in Fig.(26.3a) selected one of the center harmonics and see a feeble sine wave in the display window. This wave has about the same frequency as our original sine wave, but a much smaller amplitude.

In Fig.(26.3b) we selected 5 of the center harmonics and see that we are beginning to build a pulse at the center and reduce the waves at the edges. In Fig.(26.3c) we selected about 2/3 of the center harmonics and see that the pulse has been mostly reconstructed and the waves out from the pulse have been mostly cancelled out. By Fig.(26.3d) where we have selected the center bulge of harmonics, we see that we have an almost complete reconstruction of the chopped off section of a sine wave. The lesson is that a pure sine wave is mathematically infinitely long in both directions. Whenever you cut off a sine wave, the result is no longer a pure sine wave but, instead, a mixture of sine waves.

Our analysis here is not just some abstract mathematical theory. There are important physical consequences which can be directly seen in the behavior of laser beams.

A laser beam is famous for the purity of the light in the beam. The typical laser beam is essentially a single wave with a unique frequency (color) and amplitude. A laser beam comes about as close as we can to a physical representation of a mathematical sine wave. For example, a red laser beam one kilometer long contains nearly two billion similar wavelengths.

The continuous lasers you are familiar with are not the only kind used in research labs. Chemists, for example, need lasers that emit a very short pulse in order to study the behavior of atoms during a chemical reaction. There are now infrared pulsed lasers, where the pulses are only a few wavelengths long. When you create a short laser pulse, you are essentially chopping the light wave off in much the same way we chopped off the sine wave in Fig.(26.2c). Instead of getting a beam of a pure color (a single frequency or harmonic), you get a beam with a spread of colors (or frequencies).

Figure (26.4a) shows the electric field waves in an infrared pulsed laser beam about 12 cycles long. Figure (26.4b) shows the spread in wavelength (spread in colors) of the light in the pulse. Notice that the spread in harmonics contained in our chopped off sine wave closely resembles the spread in wavelengths in the laser pulse. (1 nanometer[nm] = 10⁻⁹ meters).
27) TECHNICAL DETAILS, PULSE FOURIER TRANSFORM

With the pulse Fourier transform, we can study different shaped pulses by selecting different parts of any wave we can record, or get using math waves.

Here we wish to discuss some different ways we have to create a pulse, and to look more carefully at how we interpret the results. First the interpretation.

Repeated Pulses

When we do a pulse Fourier transform, the only difference between that and a regular Fourier transform is that we zero out the non selected part of the curve rather than expanding the selected section to fit the display window. After that everything else is the same.

Explicitly what you see in the display window is assumed, mathematically, to repeat forever. If we display one complete cycle of a sine wave, and that repeats indefinitely, the result becomes a pure sine wave that shows up as a single harmonic in Fig.(26.1b). If we created a pulse as in Fig.(26.2c), MacScope still assumes that the window is repeated indefinitely. As a result the Fourier analysis window shows the harmonic structure of the repeated series of pulses as indicated in Fig.(27.1b).

Our analogy to a pulsed laser still holds fairly well. A pulsed laser sends out not a single pulse, but a steady series of pulses like that in Fig.(27.1b). The main difference between the pulsed laser and our Fig.(27.1b) is that the repeated laser pulses are much farther apart, i.e., separated by many more wavelengths. This wider separation, however, does not appear to make much of a difference in the resulting spectra of harmonics. Our pulse Fourier analysis spectra has essentially the same shape as the spectra of the laser pulse.

Creating the Pulse

For convenience, MacScope provides three different ways of constructing the pulse. When you press the Pulse FT you get the window shown in Fig.(27.2) allowing you to choose between zeroing the curve 1) at the center of the pulse, 2) at the left edge, and 3) using a Gaussian envelope. In Fig.(26.2c) we zeroed the curve at the center of the pulse, leaving as much curve above as below the zero line. This gave us the symmetric pulse we wanted to study.

In a Pulse Fourier Transform, you select a section of a curve, and we zero the rest of the curve to create a pulse. Then we do the Fourier transform.

There are three ways we can zero out the rest of the curve. We can make the height of the zero line at the center of the pulse. We use this for sine wave like pulses in the study of the uncertainty principle.

We can draw the zero at the left edge of the pulse. We use this along with the rectangle math waves to create slit structures to study diffraction patterns.

The third way is to use a gaussian envelope. This gives the smoothest pulse.

Fig. 27.2 Where to put the zero line.

Creating the Pulse

For convenience, MacScope provides three different ways of constructing the pulse. When you press the Pulse FT you get the window shown in Fig.(27.2) allowing you to choose between zeroing the curve 1) at the center of the pulse, 2) at the left edge, and 3) using a Gaussian envelope. In Fig.(26.2c) we zeroed the curve at the center of the pulse, leaving as much curve above as below the zero line. This gave us the symmetric pulse we wanted to study.

In a Pulse Fourier Transform, you select a section of a curve, and we zero the rest of the curve to create a pulse. Then we do the Fourier transform.

There are three ways we can zero out the rest of the curve. We can make the height of the zero line at the center of the pulse. We use this for sine wave like pulses in the study of the uncertainty principle.

We can draw the zero at the left edge of the pulse. We use this along with the rectangle math waves to create slit structures to study diffraction patterns.

The third way is to use a gaussian envelope. This gives the smoothest pulse.

Zero at Center

Zero at Left

Gaussian

Cancel

Fig. 27.2 Where to put the zero line.
Creating Slits

In our paper on Fourier Optics, we formed a pulse using the left edge option for zeroing the curve. The point was to create what looked like the set of slits that we used to study the diffraction of a laser beam. That allowed us to demonstrate that the Fourier transform of a slit structure was similar to the diffraction pattern produced by a laser beam passing through the slits.

This study began when we helped with a student project by creating the slits using Adobe Illustrator™. First we drew parallel lines spaced so that the distance between line centers was three times the thickness of the lines. We then used Illustrator’s Scale command to reduce the size of the plot until the thickness of the lines was 50 microns (fifty millionths of a meter). The students took the program over to a high resolution printer at the medical school and printed the lines in reverse on a transparent medium. The result was a diffraction grating with slits 50 microns wide spaced 150 microns apart. These slits and the resulting diffraction patterns are shown in Fig.(27.3).

To create corresponding slits with MacScope, we used the Rectangle Math Wave, with the top of the waves 33% of one cycle as shown in Fig.(25.5). To create three slits, we selected three cycles, pushed the Pulse FT button, selected to zero the curve at the left edge and got the result shown in Fig.(27.4).

In Fig.(27.4) we have plotted the laser diffraction pattern below the graph of the harmonics contained in the slit pattern. The only adjustment we made was to scale the diffraction pattern so that the bumps in the diffraction pattern match the bumps on the Fourier harmonic plot.

When you photograph a diffraction pattern, you are observing the intensity or energy density of the light pattern. The intensity is proportional to the square of the amplitude of the light wave.

In MacScope there are three ways to look at the Fourier transform harmonics. You can look at the amplitudes by selecting the Amplitude button, the phases by selecting the Phase button, or the intensities by selecting the Intensity button. With the intensities we are simply plotting the square of the amplitudes. Figure (27.4) shows the intensities of the harmonics contained in the three slits, intensities that closely match the diffraction pattern.

---

Fig. 27.3 The slits we created using Adobe Illustrator, and the corresponding diffraction patterns.

Fig. 27.4 Comparison of the right half of a 3 slit diffraction pattern with the Fourier transform of the slit structure.
**Gaussian Pulse**

Here we started with the sine wave of Fig.(26.2a) and chose the *Gaussian* envelope to more closely represent the experimental data. In Figs.(27.5) we see the result of making a Gaussian pulse from one, two, four, and eight sine wave cycles.

(Mathematicians like the Gaussian pulse because of the theorem that the Fourier transform of a Gaussian pulse is a Gaussian distribution of harmonics.)

![Fig. 27.5a Gaussian pulse from 1 cycle](image)

![Fig. 27.5b Gaussian pulse from 2 cycles](image)

![Fig. 27.5c Gaussian pulse from 4 cycles](image)

![Fig. 27.5d Gaussian pulse from 8 cycles](image)

*Fig. 26.4 (repeated)*

The experimental pulse is more like a Gaussian.
28) PLAY SOUND & SIGNAL GENERATOR

MacScope has the ability to play selected harmonics. There are two reasons for doing this. One is to hear how a note is constructed by listening as you add harmonics. The other is to use MacScope as a signal generator.

To demonstrate what we get when MacScope acts as a signal generator, we used Math Wave to create a 256 Hz Ramp Up wave seen in Fig.(28.1a). Selecting one cycle, we pressed the Fourier button and got the transform seen in Fig.(28.1b). After selecting the first six harmonics, we pressed the Play button and got the Start Playing Sound window. If you have set your computer to play sounds, you will hear a note composed of the first six harmonics of the ramp wave.

What we did instead, is to take the voltage signal that would have gone to the loudspeaker, and use it as the input to MacScope running on an older OS9 Macintosh. Fourier analyzing that signal on the old Mac gave us the results seen in Fig.(28.3). We see that the six harmonics came through quite well.

For Fig.(28.4), we created a 256 Hz Ramp Up wave on a Dell Windows computer, played six harmonics, and recorded them a Mac. We still see the six harmonics, but the relative amplitudes appear to be a bit more distorted by the Dell’s output amplifier.
INDEX

Symbols
S0, external hardware 2
$10 microphone 1
$40 Griffin iMic M:5
+ to -, change M:25
.sdif M:6
[S] reset button M:10
[V] reset button M:11
10 Hz frequency cutoff 2, M:5
20 femtoseconds Laser pulse 9
22,000 Hz
Cutoff frequency 2, M:5
Nyquist frequency M:5
2 volts M:5
44,000 points per second 2, M:5

A
AC coupled input M:5
Action potential pulse M:22
Add Window Label M:13, M:14
Adobe Illustrator M:36
Creating slits 17
Advanced controls M:23
Hidden 2
Aliasing
Nyquist frequency M:5
Spurious results 2
Amplification
Calibrate M:19
Mac amplifier M:21
Mac input M:18
Amplification scroll bar on Mac M:20
Amplitude
Of Harmonic M:36
Amplitude, phase, intensity
Section on 6
Fourier coefficients 22
Amplitudes, wavefront 19
Application icon M:6
Area under both sides of equation
Derivation of Fourier component 5
Area under curve
Fourier analysis 22
Area under picture
Derivation of Fourier component 6
Audio frequency range M:5
Audiogear.com M:5
Audio oscilloscope M:5
Audio voltage input M:5
Automatic triggering
Section on M:25
Stable picture M:8, M:25

Auto Scale M:9, M:10
Average Triggered Curves
Introduction to M:26
Section on M:29
Using Record** M:29
When to Use M:29
Averaging
Given number of curves M:23
Introduction to signal averaging M:21
None M:23
Until Stop M:23

B
Banks of data M:23
Bell, recording using trigger M:27
Big harmonics in Curve A
Reconstructing a curve 4
BNC cable M:5
Both sides of the stereo input M:5
Built in microphone M:5
Button
Fourier M:16
Pulse FT M:35
Record** M:10

C
C, middle M:6
Cable, BNC M:5
Calibrate M:19
Calibrated 100 mV sine wave signal M:18
Calibration feature M:18
Calibration method M:7
Calibration routine M:19
While MacScope is running M:20
Change + to – M:25
Changing time scales
Section on M:8
Composite files, creating M:14
Compressional wave
Derivation of speed 31
In Slinky 29
Waves in wires 30
Computer input
Frequency response M:32
Connectors
BNC M:5
RCA M:5
Stereo MiniPlug M:5
Continuous sine wave M:35
Copper pipe
Sound in 25
Create Math Wave  M:30
Create Text File  M:13
Creating a Pulse  M:35
Creating composite files
  Section on  M:14
Creating slits  M:36
Cursor, dragging  M:15
Curve A and Curve B  M:8
Curve A time display  M:8
Curve A values  M:15
Curves
  Effectively triggered  2
  Repeated indefinitely  7
Cycle, selecting  M:15

D

Data
  Averaged  M:22
  Creating & saving text file  M:14
  Expand selected data  M:15
  Fourier transform data
    See Fourier menu  M:16
  MacScope data banks  M:23
  Noisy data  M:22
  Observing data  M:8
  Raw data, 177,000 points  M:23
  Saving text file
    See File menu  M:14
  Selecting data  M:15
  Show Extended data  M:23
  Show Normal data  M:23
  Signal averaged, 2200 points  M:23
Data Window
  Frequency  M:15
  Period  M:15
  Depolarizing potential  M:22
  Diffraction pattern
    Intensity  17
  Different time scales  2, M:8
  Diffraction pattern  M:36
  Diffraction patterns
    Same as Fourier Transform  17
  Digitize voltage  M:5
  Dragging cursor to select data  M:15
  Dragging holding shift key  M:15
  Dual beam scope
    Y connector  23

E

Edit menu
  Redo  M:12
  Restore Next Curve  M:12
  Return To  M:12
  Undo  M:12
  Unstore Current Curve  M:12
Educational Innovations Inc  M:27
Electrical conduit pipe
  Speed of sound in  23
Electrical noise  M:21
Electric field
  In laser pulse  9
Electric Field of pulsed laser  M:34
Electron
  Interference pattern, probability interpretation  13
  Sent one at a time  9
  Two-slit interference pattern  13
  Wave  13
Energy
  In wave  13
  Photon probability distribution  14
  Energy scale to harmonic scale  13
  Energy spectrum
    Section on  13
  Energy-time form of uncertainty principle  9
  Expand button  M:15
  External hardware  M:5
  External Microphone Input  M:21

F

F(t), a function F of time t  5
Factory reset  M:10
Felt hammer
  Piano  3
Femtosecond Laser Pulses
  Section on  9
File menu
  Add Window Label  M:13
  Create Composite File  M:14
  Create Text File  M:13
  Open Data file(s)  M:13
  Save All Data As...  M:13
  Save Current Data as...  M:13
  Save Plot  M:13
  Save Text File  M:13
Files
  Composite files  M:14
  Creating stored files
    When you press the stop button  M:12
  Naming Stored files  M:13
  Saving Stored Files  M:13
First harmonic
  Guitar string  3
  Superimposed  M:16
**I**

Icon
  MacScope II M:6
  Sound Data File M:6
Illustrator, Adobe
  Creating slits 17
iMic 1, M:5, M:18
  Line/microphone switch M:20
  Voltage range M:18
iMic
  www.griffintechnology.com 25
Infra-red laser pulses 9
Infrared pulsed laser M:34
Input
  AC coupled M:5
  External microphone M:5, M:21
Griffin iMic M:5
  Hardware M:5
  Input volume M:20
  Input volume amplifier M:21
  Input volume or gain M:7
Microphone M:5
  Sound input panels M:7
Input menu
  Calibrate M:19
  Calibrate A and B M:19
  Change + to - M:25
  Sound Input Control M:7
Intensity
  Fourier transform of slits 17
  Diffraction pattern 17
  Of harmonic M:36
Intensity, Amplitude, Phase
  Section on 6
Intensity, phase, amplitude
  Fourier coefficients 22
Intensity of the fields
  Laser pulse 10

**J**

Junction Potential, Synapse M:22

**K**

Karl Mamola
  Compressional waves in wires 30
Keyboard, piano
  Striking point 3
Mathematics of Fourier analysis

Section on 5

Math waves M:30
Fourier optics M:30
Ramp waves M:30
Rectangle for slits M:36
Rectangle wave M:30
Sawtooth wave M:30
Signal generator M:30
Sine wave M:30
Square wave M:30
Triangle wave M:30
Maximum voltage at (t = 0) M:8

Metal pipes
Speed of sound in 23

Microphone
Built in microphone M:5
For piano project 1
G4 external input M:21
iMic line input M:18
iMic microphone input M:18
Line voltage & Mic voltage M:5
Two for dual beam 24
Various inputs M:5
Voltage range M:18

Middle c frequency M:15
Should be 262 Hz 2

Middle C Project
Introduction to 1

Modes of Oscillation
Section on 3
Muscle fiber M:22
Musical instruments
Fourier analysis of 1

N

Naming Stored files
Section on M:13
Nanometer M:34
Neuron-muscle fiber synapse M:22
Nodes
Node at point being strumming 3
Nodes at endpoints 3
Noise
Electrical M:21
Signal Averaging M:21
Noisy data M:21
Nyquist frequency 2, M:5

O

“Oh” sound, recording M:8

Observing Data
Section on M:8
Offset scroll bar
Amplifying curves M:11
One cycle
Of a repeating wave 7
Why selected 7
One cycle, selecting M:15
One cycle pulse M:17
Open Data file(s) M:13

P

Particle, electron
One at a time 13
Particle waves 9

Period and frequency, Fourier analysis M:15
Period t M:15
Frequency measurement 2
Phase
Of Fourier coefficients 18
Of Harmonic M:36
phase, amplitude, intensity
Fourier coefficients 22
Phase, Intensity, Amplitude
Section on 6

Photoelectric effect
Particle/wave nature 9
Photon
Energy of 13
Energy probability distribution 14

Piano
Felt hammer 3
Piano wire 3
Piano notes
Sound data file of M:6
Play sound M:38
Plotting window M:8
Points per second M:5

Potential, Synapse Junction M:22
PowerBook M:18
Preserve adjustments M:10
Pretrigger window M:27
Prism or diffraction grating
Analogy to Fourier analysis 1, 4
Probability distribution
Of photon energies in a pulse 1
Probability interpretation
Section on 13
Program, MacScope M:6
Pulse
  Creating M:35
  Double the length of 14
  Fourier analysis of 12
  Gaussian M:37
  Gaussian Pulse Fourier Transform 15
  Harmonics required for 12
  Laser beam M:34
  Repeated M:35
  Repeats indefinitely 12
  Sine waves in M:34
  Spectrum of harmonics in 12
Pulse, Gaussian
  Fourier transform of 15
  Harmonics in 15
Pulse, laser
  Intensity of the fields 10
  Spectrum of the radiation 10
  Using Fourier analysis 11
Pulse, one cycle M:17
Pulsed laser M:35
Pulse Fourier transform M:33
  Fourier Transform 11
  Section on M:17
  Uncertainty Principle M:17
Pulse FT
  Button M:35
  Creating a pulse M:35
  Options for zerowing curve M:35
  Where to zero M:35
  Zero curve outside pulse M:35
Pulse FT button M:33
Pulses, Femtosecond Laser
  Section on 9

Q
Quit MacScope
  Loose stored files M:13

R
Ramp Up wave M:38
  Fourier transform of M:38
  Measuring signal distortion M:38
Ramp waves M:30
Random Access Memory
  Stored files M:13
Raw data M:23
Rayleigh model for image formation 20
Realistic Pulse
  Section on 15
Reconstructing the curve
  Using Fourier Analysis 4

Record** button
  Preserve adjustments M:10
  Section on M:10
  Voltage Scroll bar M:11
Record button
  Auto Scale M:10
  Compared to Record** button M:10
  Factory reset M:10
  Starting to record M:8
Recording new data M:7
Rectangle Math Wave M:36
Rectangle wave M:30
Relative amplitude
  Of harmonic frequencies 4
Repeated set of pulses M:35
Repetitive signals M:8
Replace Curve B M:14
Reset buttons M:10
Restore Next Curve M:12
Return To M:12

S
Save
  All Data As... M:13
  Fourier Transform Data M:16
  Save Current Data As... M:13
  Save plot M:13
  Saving Stored Files M:13
  Saving text file M:13
  Saving Stored Files
    Section on M:13
Sawtooth wave M:30
Scroll bars
  Mac Amplification M:20
  Time Offsets M:10
  Time scale M:8
  Voltage M:9
  Voltage Offsets M:11
.sdf extension M:6
Selecting data
  Section on M:15
  Selected data expands M:15
  Selected data highlighted M:15
  Selected section repeats indefinitely 11
  Selecting one cycle 2, M:15
  Selecting section of curve M:15
Selecting the Sound Input M:7
Selection rectangle M:15
  Moving starting position M:7
Setting Trigger Level
  Section on M:26
Seventh harmonic
  In piano c note 3
Shareware program
  MacScope 20
  Shift key while dragging M:15
Show Extended

Section on M:23
Signal averaging
  Section on M:21
  Signal averaged data bank M:23
  Synapse Junction Potential M:22
Signal generator M:30, M:38
Simultaneous display of different signals M:8, M:22
Simultaneous display on different time scales M:8

sin(3t)sin(3t)

Calculating Fourier coefficients 5
Sine wave M:30
  Continuous M:35
  In a pulse M:34
Sine wave is infinitely long
  Fourier analysis 11
Single slit
  Fourier transform of 18
Sinusoidal modes
  Modes of oscillation 3
Slinky
  Acceleration of the coils 32
  Coils moving through pulse 31
  Conservation of Coils 32
  Measuring spring constants 29
  Newton’s Second Law 35
  Time down and back 28
  Video of compressional wave 29
  Waves on & in 27
Slinky waves
  And metal pipes 23
Slits
  Creating M:36
  Fourier transform of 18
Slit structure, Fourier transform of 17
Sound
  Fourier analysis of 1
  Playing M:38
  Signal generator M:38
Sound Data files M:6
  Opening sound data files M:12
  Piano Notes.sdf M:12
  Sound “Oh” M:8
  Sound Data file icon M:6
Sound in pipes
  Copper pipe 25
  Measuring time delay 24
Sound input M:5
  External line input M:18
  iMic Switch M:18
  Line input M:18
  Microphone input M:18
  Piano project 1
  Section on M:7
  Selecting sound input M:7
  Sound input control panels M:7
  Sound of a wooden frog M:27
Special relativity
  Teaching 9
Spectrum
  Of energy, section on 13
  Of frequencies 3, 10
  Of harmonics in a pulse 12
  Of light 1
  Of radiation in a laser pulse 10
  Spectrum of light, pulsed laser M:34
Speed of sound
  In metal pipes 23
  Young’s modulus 25
Spring constants
  Measuring 29
Square wave M:30
  Fourier Analysis of M:31
  Frequency response M:32
  Harmonics in M:32
  Square wave generator M:32
Stable picture
  Automatic triggering M:8
Stereo inputs 2, M:8
  Curve A M:8
  Curve B M:8
Stimulator pulse M:22
Stop and Store
  Section on M:12
  Stop button M:8
  Data stored in RAM M:12
  Stop on Trigger M:26
  Stop on Trigger Mode
    Section on M:27
  Stored files M:13
    Naming M:13
    Saving M:13
  Storing files M:12
    Return To M:12
Synapse Junction Potential
  Section on M:22

T

T = 0, plotting at M:8
Teaching
  Fourier analysis 1
  Special relativity 9
  Uncertainty principle 9
Teaching Fourier Analysis in Introductory Physics M:6, M:14, M:16
  Paper on 1
Teaching the Uncertainty Principle in Introductory Physics
  Paper on 9, M:17
Testing the Uncertainty Principle
  Section on 14
Text File, create M:13
Index

Time Offsets
Scroll bar M:10
Section on M:10

Time scales
Different time scales while recording M:8
In milliseconds M:8
Scroll bar M:8
Section on M:8
Tools window M:6
Record** button M:10
Record button M:8
Recording M:8
Stop button M:8
Time scale scroll bars M:8
Voltage scroll bars M:9

Topics
(Record **) button M:10
Automatic Triggering M:25
Average Triggered Curves M:29
Calibrate M:19
Changing time scales M:8
Creating composite files M:14
Fourier Analysis M:16
Hardware M:5
MacScope Program M:6
Manual Triggering M:26
Naming Stored files M:13
Observing Data M:8
Pulse Fourier Transform M:17
Saving Stored Files M:13
Selecting Data M:15
Setting Trigger Level M:26
Show Extended M:23
Signal Averaging M:21
Sound Input M:7
Stop and Store M:12
Stop on Trigger Mode M:27
Synapse Junction Potential M:22
The Voltage Columns M:28
Time Offsets M:10
Voltage Offsets M:11
Voltage Range M:18
Voltage Scale M:18
Voltage Scroll Bar M:9
Volt Scales M:9
Triangle wave M:30

Triggered Curves
Averaging M:29
Using Record** M:29
When to Use M:29

Triggering
Automatic M:25
Average Triggered Curves M:26
Button for Curve A M:26
Change + to - M:25
Manual M:26
Pretrigger window M:27
Setting trigger voltage level M:26
Stop on Trigger M:26

Trigger Mode
Average triggered curves M:29
Stop on trigger M:27
Trigger voltage, changing M:26
Trigger voltage column M:28
Trigger voltage level M:26
Two-slit interference pattern
Electron 9, 13

U
Uncalibrated Voltage Amplitude M:19
Uncertainty principle
Energy time form 9
Momentum-position form 9
Paper on 9
Teaching in Introductory Physics M:17
Time-energy form M:17
Uncertainty Principle, testing it
Section on 14
Unstore Current Curve M:12
USB 1, M:5
IMic USB output M:5
Sound Input Control panel M:7
Index  i:9

V

Voltage
Amplitudes for 100 mV input  M:19
Digitize at 44,000 points/sec  M:5
Hardware inputs  M:5
Line voltage  M:5
Maximum range is ± 2 volts volts  M:7
Mic voltage  M:5
Voltage divider, need for  M:18
Voltage divider, circuit  M:18
Voltage reduction  M:18
Voltage scale & sound input  M:7, M:18
Voltage Scroll Bar
Section on  M:9
Voltage columns  M:28
Data level  M:28
Section on  M:28
Trigger level  M:28
Voltage Offsets
Section on  M:11
Voltage offset scroll bar  M:11
Voltage range
iMic  M:18
Range is + - 2 volts  M:7, M:18
Voltage Range and Voltage Scales
Section on  M:18
Voltage scale  M:7, M:18
Calibrate  M:18
Calibrated 100 mV signal  M:18
Scale in millivolts  M:8
Voltage Scroll Bar
Amplifying curves  M:11
Section on  M:9
Volt Scales
Auto Scale  M:9
Section on  M:9
Volt Scale-Fixed  M:9
Volt Scale-Tracking  M:9
Vowel sound "oh" different time scales  M:8

W

Wal-Mart type microphone  M:18
Wave
Energy in  13
Highway construction bottleneck  28
Video of slinky  29
Wavefront amplitudes  19
Wavelength
Of pulse  10
wavelength  3
Wavelengths, spread in  M:34
Wavelengths of light
In Femtosecond laser pulse  9
Wave motion
Sound in metal pipes  23
Waves
In Hooke’s Law Media  27
On & in Slinky  27
On & in steel guitar string  27
Why We Selected One Cycle
Section on  7
Window Label, Adding one  M:14
Windows  M:5
Sound Input Control panel  M:7
www.audiogear.com  M:5
www.griffintechnology.com  M:5
www.griffintechnology.com
iMic  25

Y

Y connector
For dual beam scope  23
Young’s modulus  25

Z

Z button  M:23, M:24
Zero curve outside pulse
Pulse FT  M:35
Zero the time scale  M:23, M:24
Physics2000 Textbook

In the special theory of relativity, Einstein combined two apparently simple ideas. One is that you cannot detect uniform motion. That means that when you are riding in a jet aircraft, traveling at three quarters the speed of sound, things behave about the same way they do when you are sitting in your living room. The second idea is that the speed of light is the same to all observers. We are not as familiar with this idea because light travels too fast to easily measure its speed.

We begin the Physics 2000 text by looking at the consequences of these two ideas. Using the Pythagorean theorem, we find that moving clocks run slow, moving lengths contract, and perhaps what is most interesting, even the order of two events can be different depending upon the point of view of the observer. To see that this is the way nature actually behaves, we have included on the CD the 36 minute movie showing the experiment on the lifetime of muons raining down on Mt. Washington. We see that muons traveling at .995 the speed of light live longer than muons at rest – nine times longer. From the point of view of the muons, however, Mt. Washington has contracted to 1/9 of its published height.

By introducing the concepts of special relativity in the first chapter of the text, we can use these ideas throughout the text to present physics from a twentieth century, not nineteenth century, point of view. Early on we discuss the relationship between mass and energy. Magnetism is introduced as a relativistic effect of Coulomb’s electric force law. After discussing Maxwell’s theory of light waves, we can go directly to the particle nature of light and the simple mechanics of a zero rest mass particle. With this background, we introduce the general particle-wave nature of matter and the basic ideas of quantum mechanics. This approach avoids the great divide that usually occurs between classical and modern physics.

Teaching Relativity in Week One

Working with thought experiments rather than mathematical formalism

Aside from the fact that it is an exciting way to begin the course, the advantage of teaching Special Relativity at the beginning of an introductory physics course is that you can fit twentieth century concepts into the course as you go along.

The Teaching Relativity pamphlet provides a detailed outline of how to teach special relativity in week one using Chapter 1 of the Physics2000 text and the Muon Lifetime Movie that is on the Physics2000 CD.

Even if you do not use the remainder of the Physics2000 text, you will see the advantage of starting with special relativity no matter what text you use or what mathematical background your students have. All they need is the Pythagorean theorem.

Teaching Relativity Pamphlet

The first 23 pages of this 64 page pamphlet are a detailed outline of how to teach the basic concepts of Special Relativity in five 45 minute lectures. The remainder is a printed version of Chapter 1 of the Physics2000 text. Add to this the 36 minute Muon Lifetime Movie on the Physics2000 CD and you have all you need to start your course with Einstein’s Special Theory of Relativity.

“The Physics2000 textbook introduces relativity early in the curriculum. This major difference from the usual introduction to physics provides an alternative to the “frozen curriculum” that hasn’t changed despite all the changes in physics over the past century.”

From the announcement of the Spring Meeting of the Michigan Section of the AAPT.
Teaching Relativity in Week One
Working with thought experiments rather than mathematical formalism

Based on the Physics2000 text
Free download at Physics2000.com
**Physics2000 CD**

**for only $10**

Contains the entire introductory Physics course, including laboratory demonstrations and movies, plus Calculus2000, a physics based calculus text, and Charges2000, a field and potential plotting program. Now has MacScope II.

**Physics2000 Printed Books**

Physics2000 Part 1 & 2 plus
Physics2000 CD
$25 U.S. Postage included
(Hardcover version $40)

**Movies**

The full 36 minute movie on Time Dilation provides the experimental basis for the discussion of special relativity in Chapter 1. The newsreel footage of the Tacoma Narrows bridge has a clip from the point of view of the dog in the car. More than 15 other short movies of various demonstrations are also on the CD.

**Charges2000**

A Field and Potential plotting program for both the scientist and artist.
Free download at Physics2000.com
First two harmonics in a piano c note.