Ph 77 - Advanced Physics Laboratory

Department of Physics, California Institute of Technology

- Electronics Track -

Test Equipment & Amplifiers

Overarching Objectives

The *Electronics Track* in Ph77 encompasses an eight-week set of laboratory and prelab exercises focusing on electronic devices and test equipment with applications in experimental physics. One of our primary goals is for you to gain familiarity with the kinds of equipment and techniques you are likely to encounter in university or professional research labs involved with making precision physical measurements. Of course, we cannot include every hardware specialty in this short time, but we try to cover the most common types of electronic equipment together with a foundational description of measurement techniques and applications.

The supplementary documents *General Intro.pdf* and *eNotebook Example.pdf* (available on Canvas) contain important information that is common to all the Ph77 Tracks. Please read both these short documents before proceeding. If you are serious about learning about electronics at a practical level, we recommend that you purchase the textbook *The Art of Electronics*, by Paul Horowitz and Winfield Hill, and look through it independently while working on this Track.

Part I. Electronics Test Equipment

We begin this track by assuming that you are generally familiar with using electronics test equipment from your experiences in Ph3, Ph6, and Ph7. But we also assume that you may need some additional practice in Ph77 to refresh your memory. This first lab session, therefore, asks you to complete the series of exercises described in this lab handout, which you will document in your e-notebook.

First locate a Siglent SDG2042X Waveform Generator (Figure 1) and a Keysight EDUX1052A Oscilloscope (Figure 2), and begin by sending a 1 kHz sine-wave signal, 1 volt pk-pk to the oscilloscope. Press the *AutoScale* button on the ‘scope to do a general reset and place the waveform on the screen. Adjust the settings to put about five wave periods on the ’scope, thus yielding a screen that looks roughly like that in Figure 3.

As with most electronics devices, the engineers tried to make the user interfaces as intuitive as possible with these instruments, but they do not always succeed. Ask for help if you get stuck, and

Figure 1. The Siglent SDG2042X Waveform Generator.

Figure 2. The Keysight EDUX1052A oscilloscope.

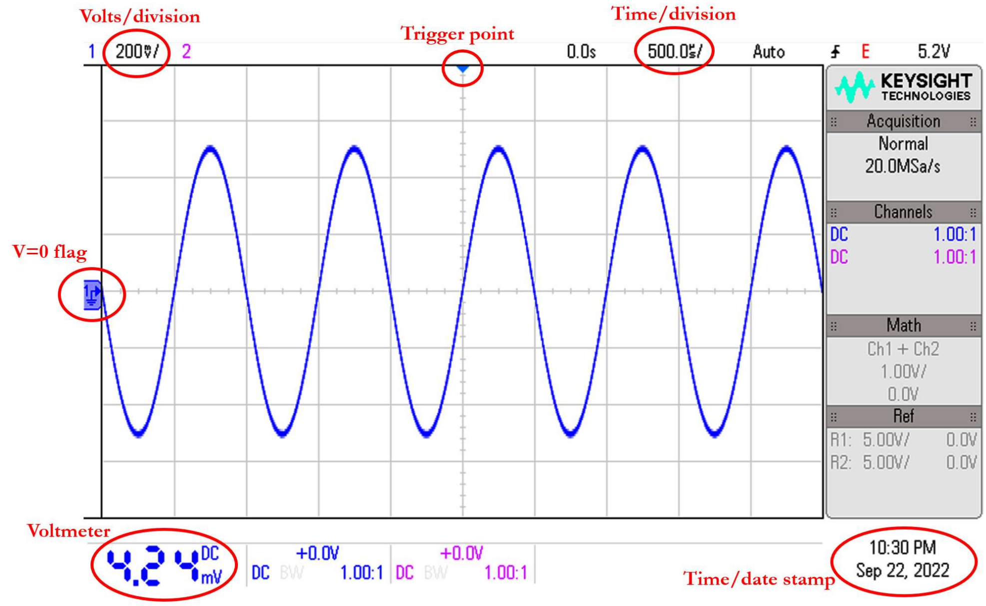

Figure 3. A (negative-image) screen shot of a 1kHz sine-wave signal displayed on the Keysight EDUX1052A oscilloscope. Note that the voltage scale (200mV/division) and time scale (500 μsec/division) are included in the image. The trigger point (small orange triangle) is also shown at the top center, and the ground (V=0) indicator is on the left. Check that the time/date stamp is correct on your 'scope.

you are also encouraged to find and consult the instrument user manuals (which you can find online) for guidance. If you continue working with electronics test equipment in the future, online manuals are often the best way to understand the myriad settings and features of each instrument, thus allowing you to use it most effectively.

Of course, sophisticated and powerful pieces of electronics test equipment tend to have thick manuals, so they can take time to master. But learning the basics in this Track will take you a long way, as the underlying physics and nomenclature remains relatively unchanged as you gain experience.

Note that the oscilloscope screen includes the observed waveform along with several useful pieces of information, as illustrated in Figure 3. These include:

1) The vertical scale in volts/division
2) The horizontal scale in time/division
3) The V = 0 flag (also called "ground")
4) The trigger point.

In case you have forgotten about triggering, the 'scope waits until the voltage passes through the trigger level (set to V=0 in Figure 3) with the proper slope (set to be positive in Figure 3). When this happens (a trigger event), the signal is plotted forward and backward in time, placing the trigger event at the trigger point on the screen. The 'scope then waits for the next trigger event, updates the signal, and this continues indefinitely. You can move the trigger point on the screen, and you can change the trigger level and slope. Try it.

Familiarize yourself with the Keysight oscilloscope by turning the various knobs, looking through the different menus, and generally exploring the settings. Because you have used oscilloscopes in the past, much of this should look somewhat familiar. Each oscilloscope is different, but the basic operations tend to look similar on most brands and models.

Exercise 1. When you have a stable signal on your 'scope (a 1 kHz sine-wave signal, 1 volt pk-pk), save a screenshot (a .png file) to the USB drive on the oscilloscope (use the Save/Recall button on the 'scope to set this up). Add this image to your e-notebook along with a brief description.

As promised, we are starting slow here. Think of this exercise as a simple first milestone. You do it just to make sure that you know how to do it, so then we can move on to more difficult tasks. And you put the screenshot in your e-notebook to document this (minor) milestone.

You are welcome to write something simple like "Result from Exercise 1" as a figure caption, and that is sufficient. But please do **not** paste the image in your notebook with no comment whatsoever. If you write "Exercise 1" in a caption, at least that ties the image to the handout, and others can make some sense of that. But a bare image floating on the page is hard to decipher.

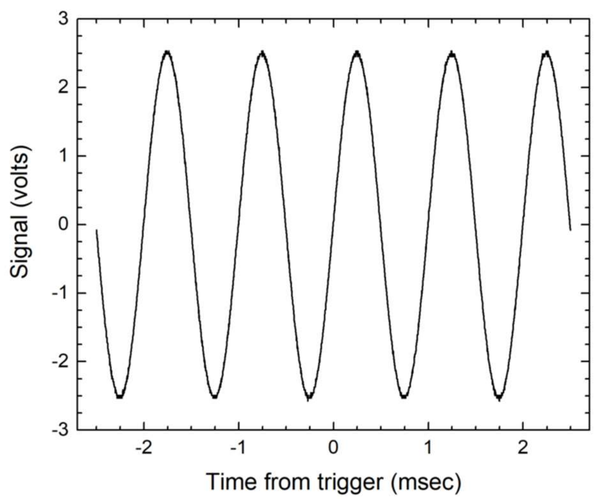

Figure 4. A plot of the numerical data corresponding to the screenshot in Figure 3.

Exercise 2. With a 5Vpp signal on the 'scope, save the numerical data to a .csv file (comma-separated-values in text format) on the USB drive and use your own computer to produce a graph like that in Figure 4, and add this to your e-notebook with a brief description. [Again, "Result from Exercise 2" is sufficient.] To extract, plot, and analyze the data in this file, you might use Mathematica, Python, or some other software of your choosing. Please see the supplementary document *MathematicaNotes.pdf* (available on Canvas) for tips on how to handle data files using Mathematica. Many of the plots in this document – including Figure 4 – were made using a package called *Microcal Origin* (for Windows), and a free 6-month trial version is available if you are interested. In experimental physics, it is often useful to have a broad variety of tools at your disposal, including software tools.

Of course, there are many software products to choose from, so you have to decide for yourself which you want to focus on, and which you want to leave behind. We recommend Word for your e-notebook, and we support Mathematica for doing data analysis. LaTeX is the norm when doing physics theory, as it handles equations especially well.

Keep in mind that the visual quality of your written documents and oral presentations will make a difference whatever your career choice, as you inevitably need to communicate your ideas and results to others. So, pick your software tools accordingly, and try to learn them well.

Exercise 3. Fit a sine function to the data in your .csv file and extract the wave frequency (in Hz) from the fit. Mathematica is a good tool for this task, as described in *MathematicaNotes.pdf*. If you know Python, that is a good option as well. Figure 5 was generated using Microcal Origin. Whatever

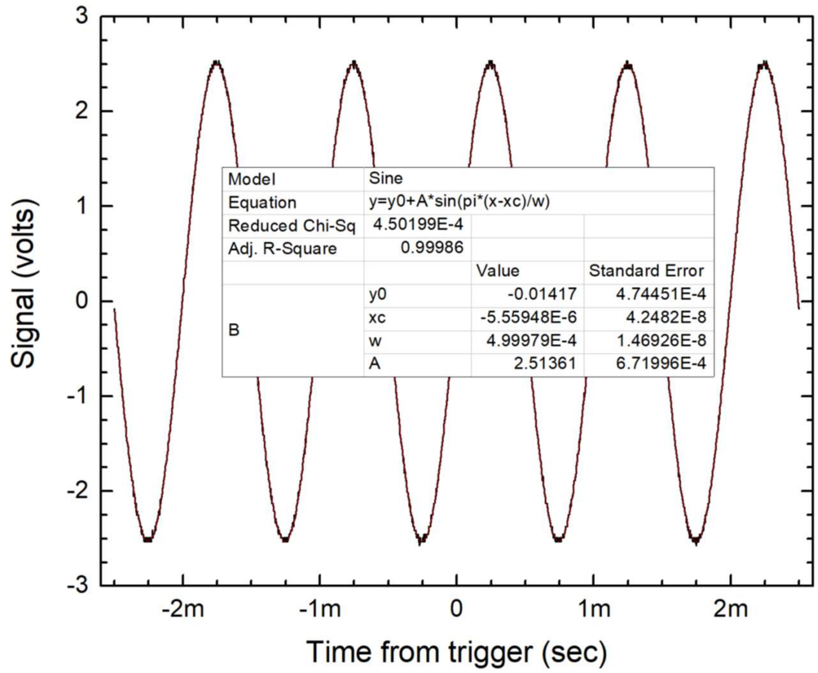

Figure 5. A sine-wave fit of the data shown in Figure 4. As shown in the parameter table, the fit yielded a frequency of 1000.04 ± 0.03 Hz.

software you use, show your work in your e-notebook, including a description of the software you used to make your fit. Insert screenshots showing your calculations as needed.

Include an uncertainty estimate in your frequency measurement, which should be generated by your fitting routine. You will find that the measured frequency is quite close to what you entered into the waveform generator. This is because clock technology is remarkably advanced, so the onboard frequency timer on the waveform generator will likely match the onboard timer on the oscilloscope to 100ppm or better.

For your next task, turn the Siglent waveform amplitude down to 10 mV pk-pk and use a BNC inline 10x attenuator (also called a 20 dB attenuator) to reduce the signal amplitude further, so now the 'scope sees a 1mV pk-pk input signal. (When using inline attenuators, the signal should go into the small BNC end and out the large BNC end. So connect the attenuator directly to the 'scope input.)

The 'scope triggering circuitry may not work on such a weak and noisy signal, so you will want to use *External Triggering* to obtain a stable signal on the 'scope. If you have forgotten how this works, see the setup shown in Figure 6. You will need to set Trigger/Trigger Type/Source=Ext on the 'scope.

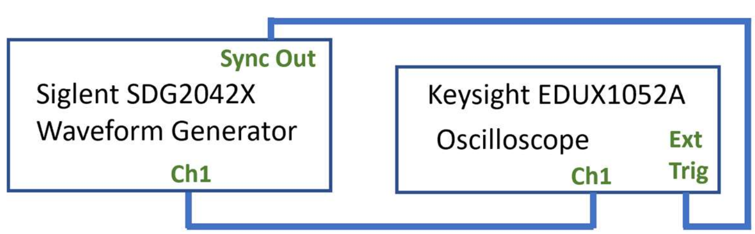

Figure 6. A wiring diagram for using external triggering.

Press the External button on the 'scope and you can view the Ext trigger signal along with Ch1. If any of this remains puzzling, talk it over with your lab partner and/or your TA.

Once you have a stably triggered signal, set the 'scope vertical gain to 1mV/division and verify that the 'scope indeed sees a signal that is 1mV pk-pk. (If not, figure out why, or ask your TA.)

Exercise 4. When you have a suitable low-voltage signal on the 'scope, add a screenshot to your e-notebook, along with a brief description. Note that the 'scope's on-screen settings embedded in the image (see Figure 3) allow you to verify that the signal has the expected frequency and amplitude. (The attenuation factor is probably not exactly 10x, as high-precision resistors cost more than low-precision resistors.)

Exercise 5. Use the *Average* feature on the 'scope (in the Acquire menu) to average 128 traces of this signal. Because the noise on each trace is independent, averaging N traces should reduce the noise by a factor of $\sqrt{N}$. Add this screenshot of this to your e-notebook (with a short description, as always).

Another option is to use the *High Resolution* acquisition mode, which also reduces the noise on your signal (try it). In this mode, the 'scope samples the signal very rapidly (up to 1Gsample/second) and does a running average to reduce the noise. [Pro tip: when you average traces on this 'scope, it automatically goes into high-res mode. This explains why averaging just two traces yields such a low-noise signal.]

The take-home message here is that averaging is an easy way to reduce the noise in a signal seen on the oscilloscope. Averaging requires a stable signal, and averaging traces takes time, so this technique may not be desirable in every situation. But, as you see here, a bit of averaging can yield a much cleaner oscilloscope signal, making it quite a useful tool.

Exercise 6. Use the oscilloscope *Measure* feature to measure the frequency, the peak-to-peak voltage, and the AC RMS (N cycles) voltage of the averaged waveform. Set it up so all three of these measurements appear on the screen simultaneously. Understand the relation between pk-pk and RMS (look it up) and show that the measurements agree with expectations. As usual, add a screenshot to your e-notebook.

Exercise 7. Turn averaging off and take another screenshot for your e-notebook. How does the additional noise affect the measurements? To see why this is, imagine a signal with zero amplitude but lots of noise; clearly the RMS will be less than the pk-pk measurement in that case. This tells you that pk-pk measurements on noisy data might not be so useful when you want to know the amplitude of the underlying signal.

Note also that the oscilloscope makes measurements based almost entirely on what you see on the screen – and nothing else. If you change the horizontal scale so fewer than one cycle appears on the screen, then the frequency measurement will not give a sensible answer. Without a clear view of several sine-wave periods on the screen, the measurement algorithm will not know what to do. Try it.

If you go in the opposite direction and put too many cycles on the screen, then the frequency measurement algorithm will again get confused. This time the problem comes from digital sampling and *aliasing*, as illustrated in Figure 7. In general, a sampling interval of Δt only gives unambiguous information when signal frequencies are below the Nyquist frequency $\nu_{Nyquist} = 1/2\Delta t$.

When using an oscilloscope, the solution to both these problems is simply to make sure you have a reasonable number of cycles visible on the screen whenever you are using the *Measure* feature.

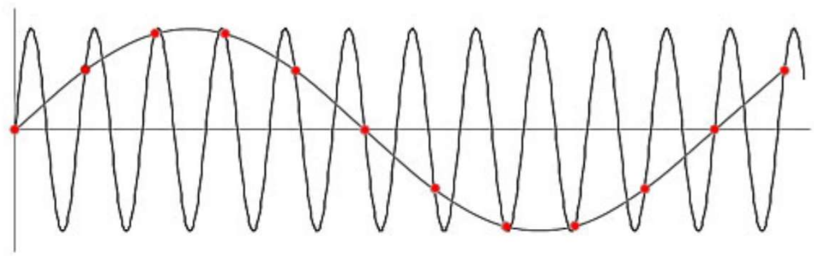

Figure 7. An example showing "aliasing" that can arise when a sine-wave signal (high-frequency sine wave) is under-sampled (red dots). If the sampling interval is greater than the period of the signal, then the dots can be interpreted incorrectly, as a low-frequency sine wave in this example.

Exercise 8. Use the oscilloscope *Cursors* feature to again measure both the amplitude and frequency of the averaged waveform. Add a screenshot to your e-notebook that shows the cursor information on the screen. [Pro tip: when you want to change which cursor you are moving, press the cursor knob.]

In practice, the *Measure* feature is great when it works, which it does when your signal is clean and strong. For weak and noisy signals, however, *Measure* can give erratic results. In these circumstances, the *Cursor* feature gives you an alternative, as your brain is pretty good at interpreting a noisy signal to know where to place the cursors.

The previous exercises illustrate the basic features of the Keysight oscilloscope, and other 'scopes typically have these features as well. We will explore more oscilloscope characteristics as the term progresses, and high-end 'scopes can have a vast array of useful analysis capabilities built in. It often takes a long time to become proficient with high-end models, while inexpensive 'scopes are easier to master. In all cases, of course, you can find online *User Guides* online that may help you understand what is going on.

Sampling, noise, and averaging

Pushing into new frontiers in science and engineering often involves making difficult, never-been-made-before measurements. In the electronics realm, this means observing weaker signals (microvolts, nanovolts, picovolts), working at ultra-high frequencies, making measurements with ever greater precision (more digits, fewer parts-per-million uncertainties), obtaining ever greater absolute measurement accuracies, and all manner of variations on these themes. In the Ph77 Electronics Track, therefore, we will talk a lot about noise in electronic measurements. Improving the signal-to-noise ratio (SNR) is the goal, and this means both creating larger signals and reducing the noise.

The Keysight EDUX1052A is an inexpensive oscilloscope designed mainly for observing voltages in the 10 mV to 10 V range, and these are generally considered large signals. But the 'scope also does a pretty good job getting down to the few μV range if you know what you are doing.

Inside the box

Figure 8 shows a schematic of what happens to the input signal inside an oscilloscope (greatly simplified, obviously!). First the signal is amplified, then it goes into a fixed 8-bit analog-to-digital (A/D) converter, and from there it goes to the internal computer for further processing and display.

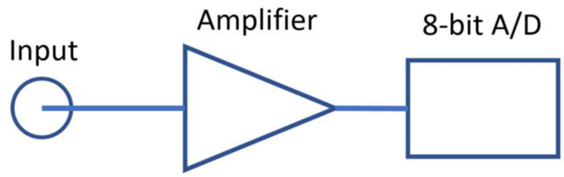

Figure 8. A greatly simplified view of the input signal processing in an oscilloscope. First a low-noise, variable-gain amplifier boosts the signal voltage, and from there it goes to an analog-to-digital (A/D) converter. Inexpensive 'scopes typically have 8-bit A/Ds, while expensive 'scopes might have 12-bit A/Ds. The digitized signal then goes to the computer for further processing and display.

The amplifier adds some noise to the input signal, and the A/D adds additional noise of about half a bit.

One normally talks about amplifier noise at its input because there the noise is independent of the amplifier gain. If you send a noise-free signal of amplitude Vin to the 'scope input, then the signal-to-noise ratio at the input is equal to SNR = Vampnoise/Vin, where Vampnoise is the amplifier noise at its input. After amplification, the SNR remains unchanged, because both Vin and Vampnoise are amplified by the same factor.

When you change the vertical gain on the 'scope, all you are doing is changing the gain of the input amplifier in Figure 8. When you change the horizontal gain, you are mainly changing the sampling rate of the A/D. Everything else is just computation and display.

If you are using the 'scope to look at a very weak signal, then you probably want to turn the gain up high. This makes your small signal visible on the screen, and the accompanying noise will likely be dominated by the amplifier noise. Put another way, if your small signal has no intrinsic noise of its own, then the 'scope will add some noise via the input amplifier.

Exercise 9. To see this for yourself, ground the 'scope input (using a BNC short-circuit GND, a 50Ω terminator, or 75Ω terminator … these all set the input signal to zero volts with negligible intrinsic signal noise) and then turn the 'scope gain up to 1 mV/division. Make sure that the Acquire mode is set to Normal (no averaging). What you see is essentially zero volts with some noise on the screen, but a .csv file will show you more.

Note that there are 8 vertical divisions on the 'scope screen (each about 1 cm high). Internally, the 'scope adds another division above and below the visible screen, making 10 divisions total. An 8-bit A/D yields 256 levels, so 10 mV (10 divisions @ 1mV/division) divided by 256 gives 40 μV per level … and that is what you see in Figure 9. Reproduce this and add it to your e-notebook. (If you see some voltage offset, this is normal. The 'scope calibration changes slightly over time.)

In this graph, the amplifier noise is the main source of the noise (vertical scatter) in the data points. The A/D noise is only about half a bit, roughly 20 μV here, so the A/D noise is essentially negligible in Figure 9.

Changing the vertical scale to lower gain settings has a large effect on the digitized signal. Reducing the vertical gain reduces the amplifier gain in Figure 8, and this gives larger voltage steps at the A/D and a different noise profile.

For example, Figure 10 shows the same grounded signal with the 'scope vertical scale set to 10V/division. At this scale, the amplifier gain is very low, so the amplifier noise is low as well. At this vertical scale, we expect the A/D to produce voltage levels of 100V/256 = 0.4 volts, and this is indeed what you see in Figure 10. Note that now the noise is confined to just a few levels, reflecting the

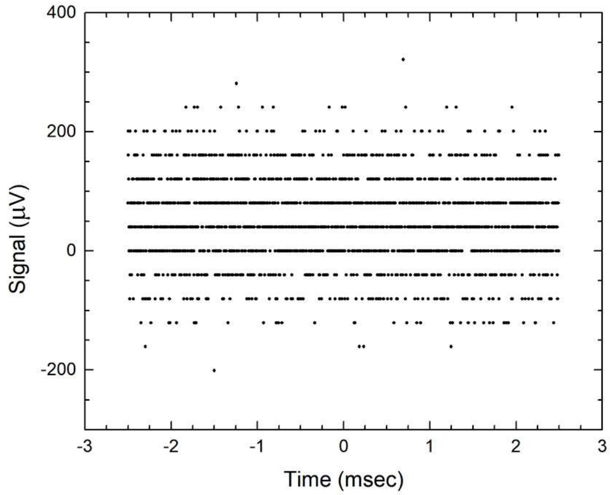

Figure 9. A digitized 'scope trace at 500μsec/div horizontal and 1mV/div vertical. The 8-bit A/D then produces voltage levels separated by 40μV. The noise level is about 70μV RMS, and the Measure feature will show this on the 'scope screen. Note that some voltage offset is visible at this high-gain setting.

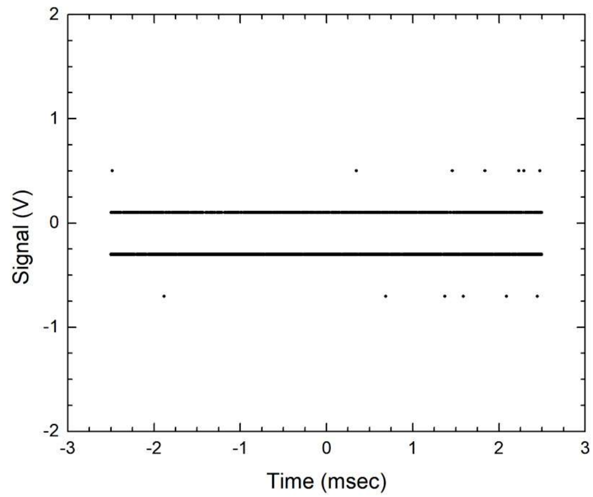

Figure 10. Similar to Figure 9, except with a vertical scale of 10V/div. Now the noise in the data is dominated by A/D noise, yielding about a half-bit of noise at all low gain settings.

internal A/D noise. In this extreme case, the amplifier noise is essentially negligible, and all the vertical scatter comes from the A/D.

Exercise 10. Again, reproduce Figure 10 and add it to your e-notebook, with suitable captioning.

Note that the A/D noise is there on purpose, and it plays an important role. With a half-bit of noise like you see in Figure 10, you can digitally average many traces to improve the voltage measurement (which is nonzero because of various voltage offsets in the 'scope). With no A/D noise, the data in Figure 10 might populate only one level, so digital averaging would not work effectively. With a half-bit of added A/D noise, however, the digital average can always yield resolutions below the 0.4-volt A/D levels, so an averaged voltage measurement can be much more accurate.

Think about this counterintuitive result – in the world of digital averaging, adding a half-bit of noise allows you to make more accurate measurements! This effect is sometimes called *stochastic resonance* (not a very descriptive name, unfortunately), and it can be found in most digital measurements. Essentially all A/D converters have a half-bit of noise hardwired in to facilitate better signal averaging via this phenomenon.

Improving resolution

One useful way to think about an oscilloscope is that it creates an image showing your electronic signal – ultimately voltage versus time. Given the discussion above, you can see that the Keysight 'scope has an image resolution of about 2000x200 pixels. This 'scope captures 2000 time points in a .csv file, independent of the horizontal scale (time per division). And the 8-bit A/D provides about 200 voltage levels (256 levels in 10 vertical divisions, but two divisions are off the screen).

Now a typical cell phone might have a 12-megapixel camera, giving an image resolution of about 4000x3000. On the 'scope, 2000 horizontal pixels is respectable, but 200 vertical levels gives a rather poor look at your signal. Expensive 'scopes do better with 12-bit A/Ds, thus giving 4096 levels, but amplifier noise will often reduce the effective resolution to far fewer than 4096 effective levels.

Averaging is the most effective tool to improve the vertical resolution beyond a measly 200 voltage levels. Averaging 128 traces reduces the noise ten-fold (root-N to be more precise), and thus the effective resolution increases to 2000x2000 pixels. You will not always see this improvement on the 'scope screen (because of its limited image resolution), but you will see it clearly in the data files.

The moral of this story is simple: whenever you are using an oscilloscope, you should consider using the average (or high-res) feature if you have a stably triggered signal. It costs very little to do a bit of averaging, and you immediately improve your vertical resolution (and thus the voltage signal-to-noise ratio).

The downside of averaging is that the signal can become quite sluggish. Try this yourself with a square-wave signal and 1024 averages. Change the signal from a sine wave to a square wave and you will see that it takes a while for the 'scope trace to respond. Even with just a few averages, the trace can be confusingly sluggish. A good strategy is to leave averaging off when you are setting up your signal, because then you want a fast response. Once you have the signal placed how you want it, maybe turn on some averaging for a better SNR.

Oscilloscopes can also average data by oversampling in time and averaging adjacent temporal bins. Your 50MHz Keysight 'scope can sample the input signal as fast as 1GSample/second, and this is much faster than needed when observing low-frequency signals. For example, the trace in Figure 9 has 2000 points covering 5 msec, so 1 μsec/point. If the 'scope's internal computer is fast enough, it could sample at full speed and average 50 samples for each of the data points in Figure 9, reducing the noise accordingly (Analog low-pass filters at the input could accomplish this also.)

The Keysight 'scope does some of this temporal averaging in high-res mode described above, but it does not have the computing power to fully realize the potential of this technique. Moreover, much of this kind of averaging is buried inside the software, hidden from the user, so we really do not know exactly what kind of averaging the 'scope is doing at its different settings. In a serious physics experiment, one builds a full data acquisition system that allows total control of these data-analysis tricks. Test equipment like oscilloscopes, on the other hand, are convenient instruments for general-purpose laboratory applications.

Observing Small Signals - I

Our next step is to look at how you can use these methods to observe quite small signals using just an inexpensive oscilloscope designed mainly for signals in the 10mV to 10V range.

Exercise 11. Set up a 5 kHz sine-wave signal with an amplitude of 5 mVpp on your signal generator. Using a BNC Tee, send this signal directly to ch1 on your 'scope and also to ch2 after first passing through a 1000x attenuator. (Do this using a pair of 30dB BNC attenuators at the 'scope input, or a 20dB plus a 40dB. Note 20dB means a factor of 100x in *power*, which means a factor of 10x in voltage. Thus 60dB gives 1000x voltage attenuation.) The attenuators should be attached to ch2 on the 'scope, and make sure the Sync output of the signal generator is off.

Use ch1 triggering on the 'scope and check that you see a stable signal on ch1. The 'scope may have some difficulty triggering on such a weak and noisy signal, but you can make it work. (If you have difficulty, try turning on HF Reject in the Trigger menu.)

With this setup, you have just 5 μVpp going to ch2, which is much too small to see on the screen. (Check this setup is correct by switching to a 5Vpp input signal and make sure you see a 5mVpp signal on ch2. If you also check ch1, you will see a 2.5Vpp signal… half what you might have expected. More on that 2x factor below. For now, focus on ch2, and go back to having a 5 μVpp signal there.)

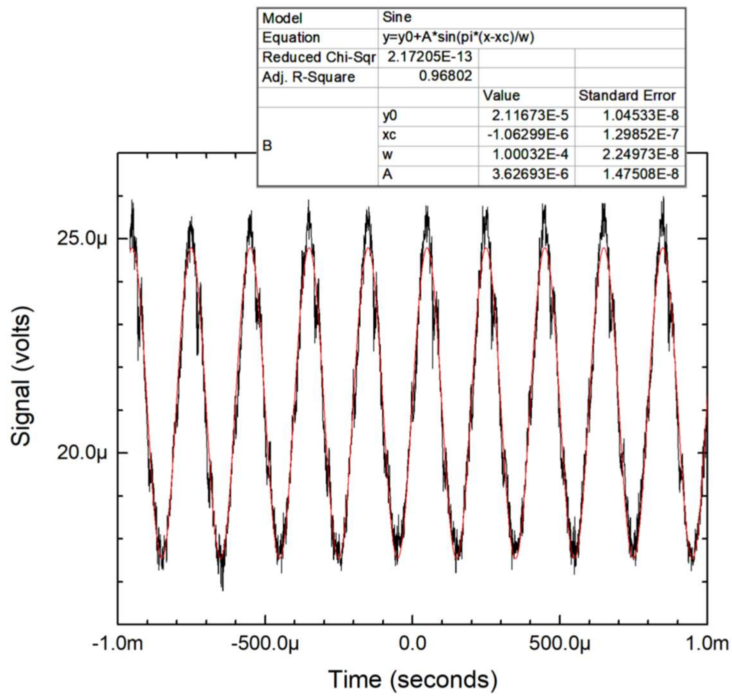

Figure 11. With lots of data averaging, the signal amplitude A can be measured to a 1σ accuracy of 15nV! This is more than 1000x smaller than the intrinsic data sampling level spacing (shown in Figure 9).

Average 1024 traces to reduce the noise, then a .csv file will give a ch2 signal that looks something like that in Figure 11, showing a reasonably nice sine wave. Here you see that a fit to the data yielded a signal amplitude of 7 µV pk-pk (a bit larger than expected) with a 1σ error of just 15 nV. Reproduce this result and add it to your e-notebook, with notes describing your analysis. If you used Mathematica, for example, include a screenshot of your notebook. Note that the uncertainty in the signal amplitude (15 nV) is more than 1000x smaller than the digitization spacing of 40µV (seen in Figure 9).

The moral of this small-signal exercise is that even an inexpensive oscilloscope can produce impressive measurement accuracy. The trick is to start with a clean, stable signal, trigger it well, and then average, average, average. Let root-N work for you.

Part II. Electronic Amplifiers

Amplifiers are ubiquitous elements in electronic circuits, often used to boost weak signals so they can be observed and analyzed using other devices. A simple voltage amplifier will produce an output voltage

$$V_{out}(t) = GV_{in}(t)$$

where G is the amplifier *gain*. As described in Part I above, the first input stage in any oscilloscope is a variable-gain voltage amplifier.

Another example is called a *transimpedance amplifier* (TIA) that converts an input current to an output voltage, giving

$$V_{out} = G_{trans}I_{in}$$

TIAs are often used to convert a photodiode signal input (a photocurrent) to a voltage output.

In this section we will examine electronic amplifiers in some detail, explore their characteristics, and demonstrate some of their capabilities in the lab.

Impedance

In the real world, there are no ideal amplifiers, and it soon becomes important to understand some of their non-ideal characteristics. To this end, Figure 12 shows the effective circuit of a basic voltage amplifier.

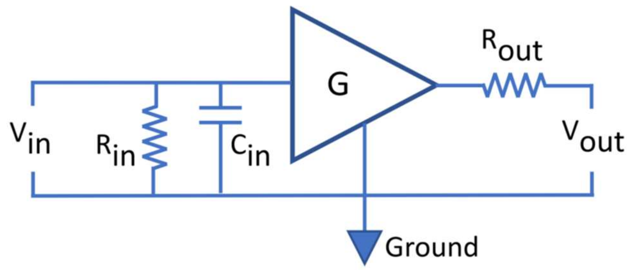

*Figure 12. Electronic amplifiers are often characterized by a **gain** G, an **Input Impedance** Rin, an **Input Capacitance** Cin, and an **Output Impedance** Rout. The triangle symbol represents an ideal amplifier with gain G.*

Input Impedance and Capacitance. When you connect an amplifier to some input voltage Vin, then the input impedance Rin will draw current from the input. For an ideal amplifier, Rin would be infinite, and the circuit being examined would not be perturbed by the added amplifier. But real amplifiers always have some finite input impedance, usually in the MΩ to GΩ range.

Similarly, the input capacitance Cin will also draw current from AC input signals, again perturbing the signal you are trying to measure. For an ideal amplifier, Cin would be zero, but typical values are in the pF range.

Note that the notation is a bit confusing here. In Ph1 you learned about complex impedances having these values for resistors and capacitors:

$$Z_R = R$$

$$Z_C = \frac{1}{i\omega C}$$

Given this, one could combine Rin and Cin in Figure 1 into a single complex input impedance

$$\frac{1}{Z_{tot}} = \frac{1}{Z_R} + \frac{1}{Z_C}$$

(because Rin and Cin are wired in parallel). However, in electronics discussions involving input and output impedances, the word "impedance" is often used to represent the resistive part only. The capacitor becomes a separate component, as shown in Figure 12. The context of the discussion matters here, but it starts to make sense after you have some experience working with electronic instruments.

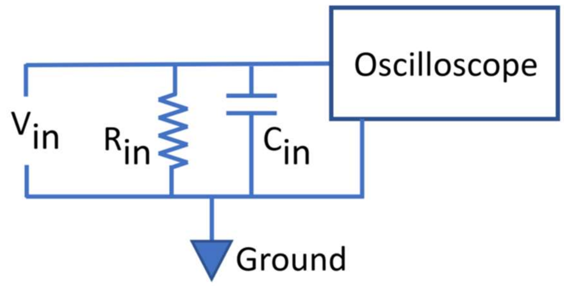

Figure 13. The first stage of an oscilloscope is its input amplifier, so an oscilloscope is also characterized by an Input Impedance Rin and an Input Capacitance Cin. The Keysight EDUX1052A has Rin = 1MΩ and Cin = 16 pF.

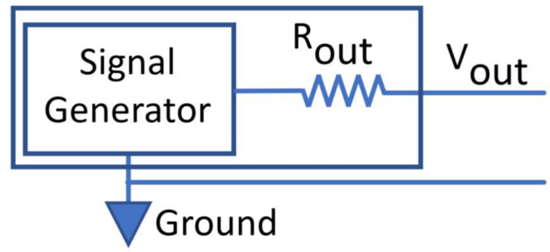

Figure 14. The Siglent SDG2042X Signal Generator has an output impedance of Rout = 50 Ω, which is an industry standard. Thus the output signal depends on the input impedance of whatever is connected to the Siglent output. Here the smaller rectangle represents an ideal signal generator with zero output impedance.

Output Impedance. The value of Rout in Figure 12 reflects the fact that a real amplifier cannot source an infinite current. As a result, Vout will depend on what you connect to the amplifier output. In an ideal amplifier, Rout would equal zero, but 50**Ω** is typical. Having a proper output impedance is also important when working with high frequencies.

Impedance examples. The concept of input and output impedances appear everywhere in electronics devices, and one example is an oscilloscope. As shown in Figure 13, the input impedance of your Keysight 'scope is 1M**Ω**. This value is high enough that usually you can connect the 'scope to view a signal without significantly altering the signal you are trying to observe.

If 1M**Ω** is not high enough for your purposes, then one solution is to use an oscilloscope "probe" that is basically a voltage divider. This increases the impedance seen by the circuit being observed, but at a cost. With a 10X probe, for example, the input impedance increases from 1 M**Ω** to 10 M**Ω**, which is good; it causes less load on the circuit. But the signal amplitude seen by the 'scope decreases by 10X, making it more difficult to see small signals. Probes are especially useful with digital circuits, when the signals are large while the circuit is easily perturbed by a 1M**Ω** impedance.

A better (and more expensive) solution is to use a "preamplifier" (often called a "preamp") that has a high input impedance (>1M**Ω**). The preamp is then said to "buffer" the signal, so it is no longer perturbed by the 1M**Ω** input impedance of the 'scope. A preamp is sometimes called an "active probe", as opposed to a "passive probe" (a resistor divider). These techniques come up when using oscilloscopes, but we will not explore them much in Ph77.

As another example, signal generators and power supplies all have output circuits that can be approximated by the circuit shown in Figure 14. The Siglent SDG2042 has an output impedance of 50**Ω**, which is fairly standard for electronic test equipment. However, the standard for some audio equipment is 600 Ohms, and the (analog) video standard is typically 75 Ohms.

In many applications, these details do not matter much. As long as all input impedances are sufficiently high, and all output impedances are sufficiently low, then everything will work as expected. For example, if you connect the signal generator (Rout = 50**Ω**) to the oscilloscope (Rin = 1M**Ω**), then the voltage you set on the signal generator will equal what you see on the 'scope to high accuracy. But input and output impedances matter a great deal once you start doing anything even remotely interesting in electronics.

Attenuator Impedances

One place you can run into immediate trouble is with the BNC attenuators we often use in this lab, because many (but not all) are designed to have 50**Ω** input and output impedances (Why? We will talk more about that later when we look at transmission lines).

To see why this can cause problems, send a 1kHz, 1Vrms sine-wave signal into ch1 of the oscilloscope, and then use a BNC Tee to send the same signal to a 20-dB attenuator attached to ch2. Measure the two signals on the 'scope and you will see factor-of-two discrepancies that can be somewhat confusing. (Try it.)

These 2x factors arise because of *impedance mismatches* in your simple set-up. The attenuator is meant to be used in a circuit with 50**Ω** input and output impedances everywhere, and that is not how you are using it. To really understand the signals you are seeing in this simple exercise, you can read the supplementary *Attenuators.pdf* document that is provided in Canvas. Without laboring over the details, remember these two overarching lessons:

1) Impedances matter in electronic circuits.
2) You cannot always trust your signal settings. If your signal generator says you have a 1Vrms signal, then put it on the 'scope and make sure. Impedance mismatch issues can easily cause factor-of-two errors if you are not paying attention.
3) Verify that your signals are what you think they are.

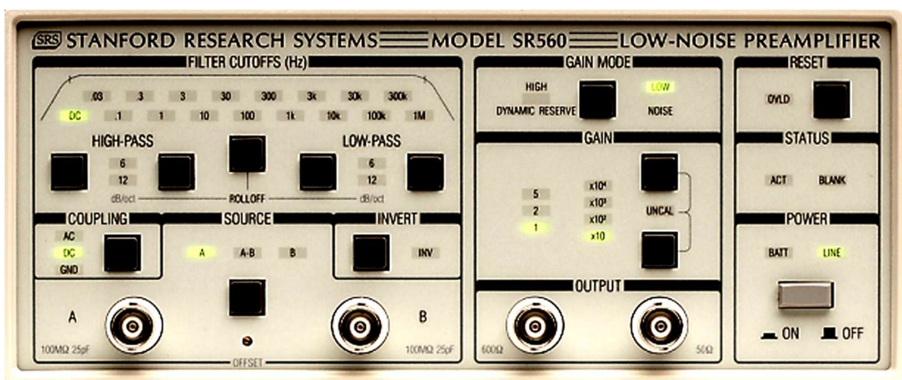

Figure 15. The SR560 amplifier.

The SR560 Amplifier

The SR560 (Figure 15) is our workhorse amplifier in Ph77, as it is a good general-purpose amplifier with lots of adjustments. It has an input impedance of 100 M**Ω** with Cin = 25 pF, making it especially well-suited for amplifying low-frequency signals, providing an adjustable gain from 1 to 50,000. So, let's fire it up and see what it can do.

Before you get started, a few words about safety (not for you, in this case, but for the instrument). A sensitive amplifier like the SR560 is easily damaged, so please treat it with care. It is best to ground the inputs (lower left, Coupling → GND) whenever you change any of the cabling. This helps avoid electrostatic discharge problems. Note also the over-voltage OVLD light (upper right); if this light goes on, then lower the amplifier gain or take some other action. Also, you should not send large input signals into the SR560; if you have a large signal, then you do not need to amplify it.

To see the SR560 in action, turn it on, ground the inputs (using the Coupling button), and set the amplifier gain to 1. Turn both the low-pass and high-pass filters off by cycling the central filter button. The SR560 controls are generally self-explanatory but ask your TA if you have any questions.

Next generate a 100mVpp sine-wave signal using your function generator and send this to the oscilloscope ch1 for monitoring. Use a BNC Tee to send this same signal to the SR560 (input A) and send the SR560 output to ch2 of the 'scope (either the 50Ω or 600Ω output-impedance channels will work fine, because the 'scope has a high input impedance). Try the various buttons and verify that you get the signals you expect.

Figure 16 illustrates what the transfer function looks like when you include a 6dB/octave ($1/f$) low-pass filter with a rolloff frequency of f_0 = 1kHz. The data in this plot were generated using the "1-3-10" rule to give points that are roughly equally spaced on a log scale. (For a higher point density, a 1-2-5-10 rule is also popular.) The line in Figure 16 is not a fit to the data but shows the theoretical expectation.

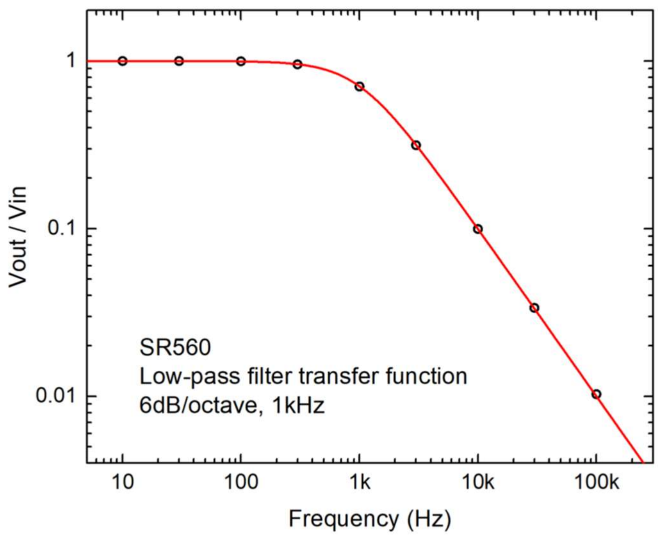

Figure 16. The transfer function, A=Vout/Vin, for a sine-wave signal passing through a 1kHz, 6dB/octave low-pass filter in the SR560. Points are measurements, the line is theory.

Exercise 12. Derive this curve, which results from a basic RC low-pass filter with the rolloff frequency

$$f_0 = \frac{1}{2\pi RC}$$

In addition, derive the transfer function for a 12dB/octave ($1/f^2$) low-pass filter. [Hint: a 12dB filter is essentially just two 6dB filters in series.]

Exercise 13. Reproduce measurements like those shown in Figure 16, except using a high-pass filter with $f_0 = 1$ kHz. Add a second theory line on the same plot (not shown in Figure 16) showing the expected transfer function for a 12dB/octave filter.

As you are making measurements, remember that AC coupling on the 'scope can distort low-frequency signals. If Vout becomes small and difficult to measure, try the usual oscilloscope tricks: average traces and measure Vrms instead of Vpp (as the latter measurement is less susceptible to noise). Also, remember that the SR560 is an amplifier, so you can adjust the gain.

The plot in Figure 16 was generated using Microcal Origin, but the Mathematica notes show you how you can handle the data in Mathematica. Python is popular also. No matter how you do your analysis, add a description in your e-notebook, including your code if appropriate. This is important so others (or future you, which is far more likely) can easily understand and reproduce what you did today. As you develop your own repertoire of software tools, you may be shocked at how difficult it is to remember how to use them all effectively. Keeping good notes with working examples helps.

And put some labels on your plot; a plot of something versus something is not adequate when doing physics. Descriptive text labels on your plot can be helpful also.

Looking at the phase information using a low-pass filter, you should find that Vin and Vout are in phase at low frequencies, but the phase difference goes to 90 degrees at high frequencies. This is typical for simple low-pass filters – when the amplitude drops off like $1/f$, it is accompanied by a 90-degree phase shift. When the amplitude drops as $1/f^2$, it is accompanied by a 180-degree phase shift. Any change in a transfer function amplitude as a function of frequency must be accompanied by a corresponding change in the phase. This general result comes from the Kramers-Kronig relation, which you can look up for further information. (It's complicated, but this is a universal mathematical result.)

If you look at very high frequencies using a 6dB/octave filter, you will see the phase shift increases beyond the theoretical expectation of 90 degrees. The reason is simply that the SR560 cannot respond instantaneously, and small delay times in the instrument translate into large phase shifts at high frequencies.

Exercise 14. With the SR560 set to DC (no input filtering), at what signal frequency do you see a 90-degree phase shift? What does this tell you about the internal time-delay in the SR560 circuitry? [Pro tip: put the 'scope in XY mode (in the Acquire menu) to better observe a 90-degree phase shift.]

Filtered Square-wave Signals

It is also useful to look at filter responses in the time domain, and one way to get a feel for this is by examining a filtered square-wave signal. To this end, Figure 17 shows a 1 kHz square wave before and after a 3 kHz low-pass filter.

In a square wave, the sharp transitions require a fast response time, which means many high-frequency Fourier components. These high-frequency terms are suppressed by the low-pass filter, and this means that abrupt transitions are no longer possible. As seen in Figure 17, the square-wave transitions are indeed slower in the filtered signal compared to the unfiltered signal. Try this for yourself in the lab with different filter settings (because this is so easy to set up), but you do not have to record anything in your e-notebook.

Figure 18 shows the same square wave upon passing through a 3 kHz high-pass filter. Now the high-frequency components pass straight through the filter, yielding abrupt square-wave transitions.

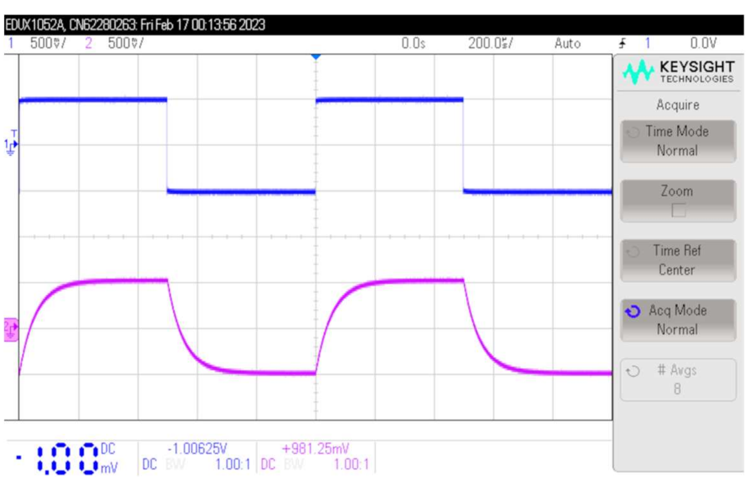

Figure 17. A 1 kHz square-wave signal before (top) and after (bottom) going through a 6dB/octave low-pass filter with f_0 = 3 kHz.

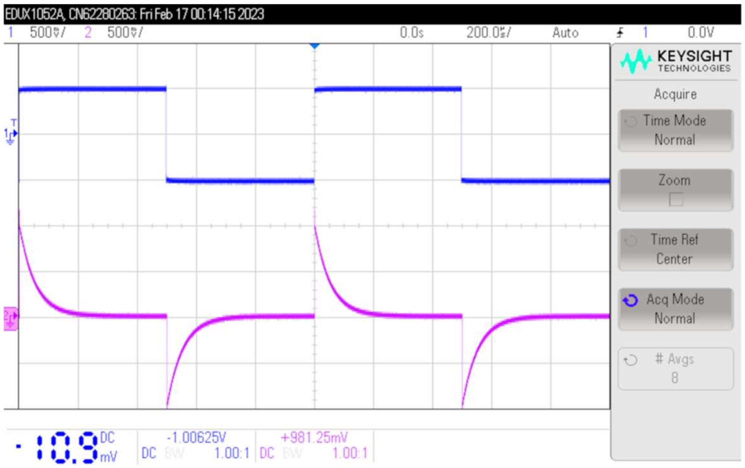

Figure 18. A 1 kHz square-wave signal before (top) and after (bottom) going through a 3 kHz 6dB/octave high-pass filter.

When the signal becomes constant (at the tops and bottoms of the square wave), this low-frequency content is filtered out, yielding zero volts in these regions.

Note that the original 1Vpp square wave signal has a min/max of ±0.5 volts, but the high-pass filtered signal has a min/max of ±1 volts. (Try it.) Something to ponder….

Figure 19 zooms in on a square-wave transition, and here you can see a couple interesting features. First, the transition is delayed by about 1 μsec, which results from intrinsic delays in the SR560. No amplifier can act instantaneously, so these delays are always present. If you operate the SR560 with no filters at all, the square-wave transition looks about the same as in Figure 19.

Comparing Figure 19 and Figure 17, you can conclude that the SR560 acts like it contains an internal low-pass filter with $f_0 \approx 500$ kHz. This is typical for low-noise electronics hardware, as it is relatively easy to get good performance below 1 MHz, but more challenging to get better high-frequency response without sacrificing some noise performance on other areas.

Another feature you see in Figure 19 is a bit of overshoot after the transition. This is another common feature in electronic devices, but we will not discuss it further here.

If you want to model how a low-pass filter behaves in the time domain, one way is to simply create a running average of the input voltage at times before the current time t. The length of the running average is roughly $1/f_0$, and how you weight the different times in the average determines the character of the filter performance. A high-pass filter can be modeled the same way, except this time the output voltage is equal to $V_{in}(t)$ minus the running average.

Low-pass filters are often used to reduce the noise in electronic signals, essentially smoothing out the input with a running average. Likewise, high-pass filters are often used to get rid of low-frequency elements, especially the DC signal. Your oscilloscope inserts a 10-Hz high-pass filter when you choose

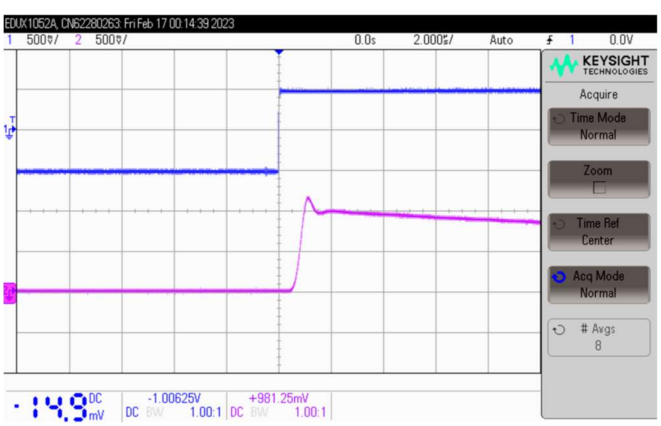

Figure 19. A zoomed-in view of the center transition in Figure 18.

AC coupling, for example. But electronic filters are ubiquitous elements in nearly all electronic circuits, serving a broad range of purposes that you will learn more about if your career takes you deeper into the electronics realm.

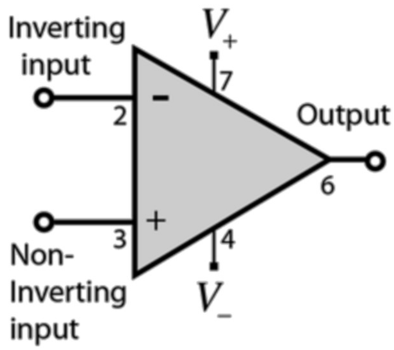

Figure 20. The output of an op-amp is essentially infinity times (Vnon-inverting - Vinverting). When used correctly, this means Vnon-inverting = Vinverting, giving a finite output voltage.

Operational Amplifiers

It is useful to look inside the SR560 box (figuratively, in this case) to see what is happening at the chip level. In the amplifier world, that means having a discussion about what are called *operational amplifiers*, or often simply *op amps*.

Figure 20 shows the basic schematic of a typical op amp, including two inputs, an output, and V+/V- power-supply inputs. The output is Vout = Gopamp*(V3 – V2), where Gopamp is so high as to be essentially infinite. Now you might think that an infinite-gain amplifier would always drive its output to infinity (or, more accurately, the output would hit the rails at V+ or V-). And this would not be a useful device. The answer to this dilemma is that op amps *always* use negative feedback to keep the output at some finite value. To see how this works, let’s look at some practical circuits.

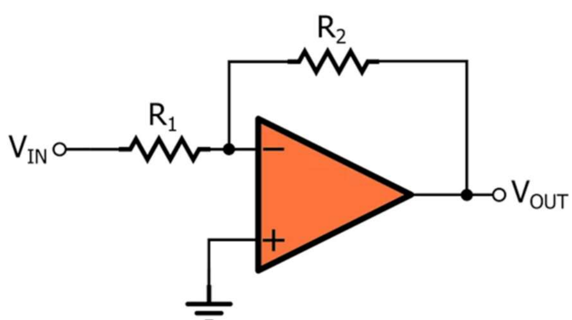

Figure 21. An op-amp is used here to make an ***inverting amplifier*** *circuit. The gain is quite stable because it is determined by the resistor values.*

Exercise 15. Show that the op-amp circuit in Figure 21 yields an *inverting amplifier* with

$$V_{out} = -\frac{R_2}{R_1} V_{in}$$

To obtain this result, you will need to know the rules of op-amp operation:

1) Neither op-amp input can source or sink any current. (Therefore, in Figure 21, any current coming into the inverting input through R1 will have to equal the current going out of the input through R2.)
2) If the op-amp is being used correctly, then Vnon-inverting = Vinverting to high accuracy. If this were not the case, then the near-infinite gain would make the output go to infinity.

Feedback control makes the world go 'round, and this concept is essential for making electronic circuits of just about any kind.

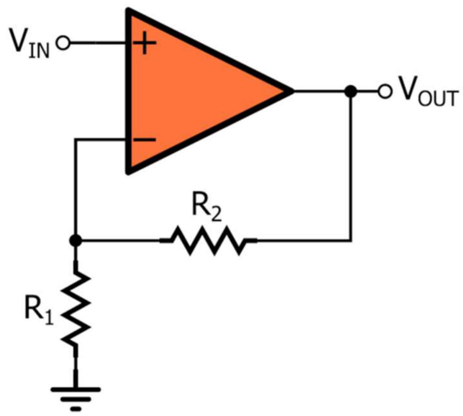

Figure 22. An op-amp here makes a ***non-inverting amplifier*** *circuit.*

Exercise 16. Use these same rules to show that the op-amp circuit in Figure 22 yields a *non-inverting amplifier* with

$$V_{out} = \left(1 + \frac{R_2}{R_1}\right) V_{in}$$

This circuit has the additional advantage that (essentially) no current is drawn from Vin, so the amplifier does not much perturb the voltage it is trying to amplify.

Many fascinating and useful op-amp circuits have been created over the years, and you can find a good selection documented online. Another example is the *transimpedance amplifier* shown in Figure 23, which converts an input current into an output voltage. The TIA is often used in conjunction with photodiode, converting the light-generated photocurrent (essentially one electron per detected photon, including some efficiency factor <1) to an output voltage.

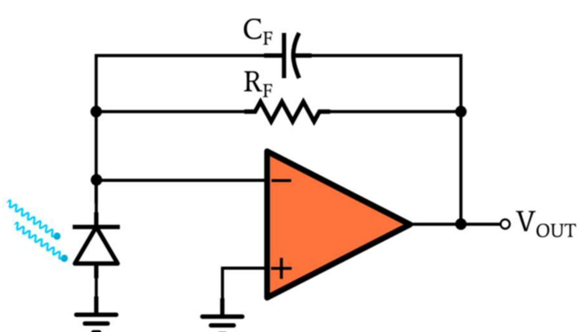

Figure 23. This op-amp makes a ***transimpedance amplifier*** *circuit that converts a photocurrent to an output voltage. Here the capacitor adds an RC low-pass filter to the output signal. A similar capacitor could be added in Figure 21 or Figure 22 with the same effect.*

Observing Small Signals - II

One of the most common uses of electronic amplifiers is to boost small signals, and you can see this in the lab simply by looking at a small signal on your oscilloscope before and after amplification. And one quick way to roughly characterize the improvement is with something we like to call the "Minimum Obviously Observable Signal", or MOOS. (This is a Ph77 invention, pronounced "moose").

As the name suggests, the MOOS represents a signal that is clearly visible but would be hard to see if the amplitude were substantially smaller. Clearly, this is a qualitative assessment, and we will get more quantitative later when we look at the noise power-spectral-density. That analysis can be a bit tedious, however, while the MOOS is straightforward and it gets the general pedagogical points across.

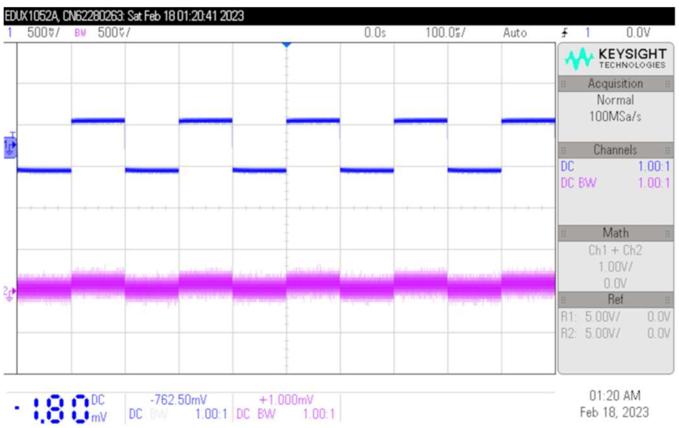

Figure 24. A 5kHz square wave signal (top) together with the attenuated MOOS signal (bottom). Note the top scale is 500 mV/division, while the bottom is 500μV/division.

Exercise 17. Start by identifying the MOOS on your oscilloscope directly, with no supplementary signal amplification or averaging. Use your function generator to produce a 5 kHz square wave signal and send this to ch1 of your oscilloscope, setting the 'scope to trigger off ch1.

Use a BNC Tee to send this same signal to ch2 of your 'scope, but this time attenuate the signal amplitude by a factor of 10^4. (We have special 10^4 attenuators to avoid impedance-mismatch issues, so it's best to use one of those…Box48) Make sure there is no averaging on the oscilloscope (Acquire = normal). Note that you do not need to measure the ch2 signal directly, because you already know the signal amplitude from the function generator and the attenuation. (Of course, you will measure it anyway because you need to verify everything you do in the lab. In this case, verify the expected conversion factor by using a large input signal.)

Set everything up on the 'scope to give a MOOS screen that looks something like that in Figure 24. Here you can see that the MOOS is quite small, but large enough so a glance at the screen indicates that a square-wave signal is present. Again, this exercise is not meant to be especially precise, so all that matters is that you use a reasonably consistent MOOS definition throughout.

Add your MOOS screenshot to your e-notebook along with your measured MOOS voltage. What this exercise says is that sending V_{MOOS} to your oscilloscope input gives you a signal that you can see reasonably well above the noise without any averaging.

Exercise 18. Next measure the MOOS using a signal amplified by the SR560. Attenuate the signal by the same factor of 10^4 (using a 10,000x attenuator plugged into the SR560 input) and view the SR560 output on the 'scope. In this case, V_{MOOS} is the signal going to the SR560 input, because that is where we begin observing the signal.

Once again, adjust the amplitude of the signal until the SNR looks about like the MOOS you defined previously. Calculate V_{MOOS} from the various settings. As you adjust the SR560 settings, you should find that the new MOOS is substantially smaller than you found with the 'scope alone.

This makes sense, of course – the SR560 was designed to amplify small signals without adding much noise; if it cannot beat an inexpensive 'scope, there is little point in having an expensive preamplifier.

If you change the SR560 gain, you will see that the MOOS (as seen at the SR560 input) does not change dramatically with gain, although higher gains generally yield a smaller MOOSs. As discussed above, this is why amplifier noise is usually referred to the input. Changing the gain amplifies the noise with the signal, so the SNR (and therefore the MOOS) is roughly independent of gain. You should also try adjusting the SR560 filters to reduce the MOOS as much as possible.

Create another screenshot that looks like that in Figure 24 for your e-notebook and record your best MOOS value. This demonstrates that the SR560 is a useful instrument to have around the lab; it amplifies small signals so they are more easily seen on the oscilloscope.

Exercise 19. As a final exercise in this series, measure the MOOS for a signal amplified by an LT1028 op-amp mounted on a printed-circuit board (PCB) shown in Figure 25. This PCB was fabricated for this lab, and the schematic is shown in Figure 26. Note the included factor-of-10^6 attenuation built

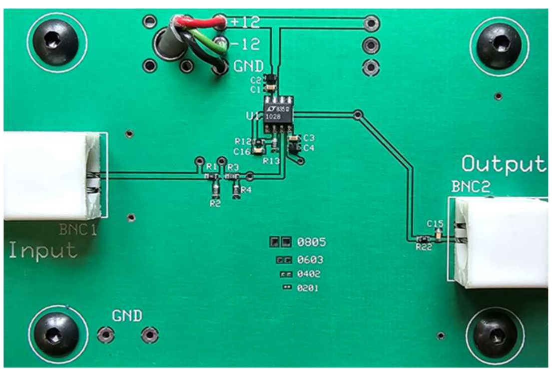

Figure 25. An LT1028 op amp mounted on a printed circuit board (PCB17).

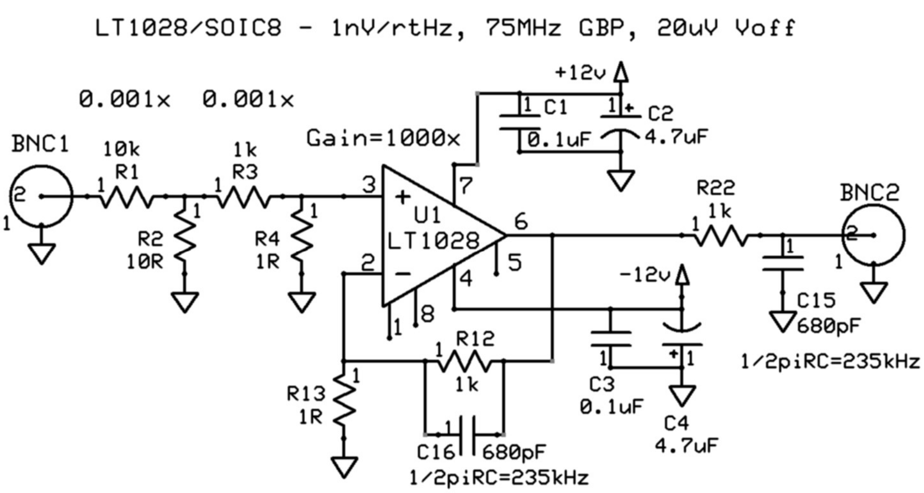

Figure 26. The schematic for the PCB in Figure 25. The input signal (left) is attenuated by a factor of 1,000,000 and then amplified by 1000x. A pair of low-pass filters reduce the signal (and noise) above 235 kHz.

into the circuit (so no additional attenuators are needed). Thus, the voltage applied to the LT1028 input is 10^6 times smaller than the signal at the BNC input to the PCB, and this tiny input signal becomes the MOOS. As seen in the circuit diagram (compare with Figure 22), the input signal is then amplified 1000x to produce the signal at the output BNC. Once again, pin 3 in Figure 26 is where you have the MOOS, as this is where the signal is first being observed (the attenuators are just a convenience to give you a suitably small signal to analyze).

Once again, measure the MOOS and record it in your e-notebook, together with another screenshot. You should find that the MOOS for the LT1028 is somewhat better than the SR560, allowing you to see input signals down to the μV level on the oscilloscope, without any averaging.

A Progression of Amplifiers

Some take-home lessons from these exercises include:

1) The Keysight oscilloscope is mainly intended for looking at fairly large signals, so its input amplifier is not especially sensitive. You can easily observe input signals in the 1mV range without averaging, but it takes a lot of averaging to do better (as you saw above).

2) The SR560 is a good multi-purpose lab amplifier that will let you see signals down to the 10-100 μV level (roughly) without averaging, making it much better than the 'scope alone for looking at small signals. Of course, you can still average to reduce the noise, just as you did without any preamplification.

Moreover, the SR560 gain can be varied over a large range, and you can insert low-pass and high-pass filtering easily. Thus, this is a handy laboratory tool whenever electronic signals are being observed.

3) Going down to the chip level opens up another world of voltage sensitivity and flexibility, allowing one to build powerful analog signal processing capabilities right into an electronics system. The downside is that you cannot use off-the-shelf test equipment (like the SR560) to obtain this extra boost of sensitivity – you have to build these amplifier chips into your experiment from the outset. In the most extreme cases, you can even forgo ICs and use discrete transistors for the first amplifier stage.

You might be wondering why manufacturers don't use low-noise chips (like the LT1028) everywhere, so all instruments have the lowest possible input noise. The reasons come down to cost and a host of practical considerations. The Keysight EDUX1052A is a low-end oscilloscope, for example, and there is a lot of competition for that market. Not many users need low-noise performance, so compromises are made. That being said, this 'scope is about 10x better than 'scopes from ten years ago, so progress in the low-noise direction is being made.

The SR560 is a fairly expensive instrument made for low-noise performance, so here chip costs are not so important. However, the hardware needs to be robust, and that means protecting the inputs from electrical shocks. And that protection invariably reduces the noise performance somewhat.

The PCB uses one of the best low-noise op-amps available, and it is protected from the user by the built-in 10^6 attenuation. If we connected the BNC directly to the input, the chip would be toast as soon as people started using it. This chip is not suitable for use in a multipurpose instrument where robustness is important.

Some words about chip-level electronics

People are often afraid to think about chip-level electronics, as it seems (to the uninitiated) like something only for seasoned professionals. This is not the case, however, and one can make a good analogy with programming.

Most of you are familiar with programming in some capacity, using Python, Matlab, Mathematica, or other platforms. And, as you know, there is an initial learning barrier to get over before you can do even rudimentary programming. But soon you find that writing a bit of rudimentary code is not difficult. Moreover, being able to program opens up a powerful new world in which one can create custom computational tools.

The reason basic programming is relatively easy, of course, is because large armies of talented hardware and software engineers have already built the necessary computers, interfaces, and compilers that translate and run your code. You need not know anything about how a compiler or code interpreter works to start writing useful code.

Likewise, with chip-level electronics, more armies of people have laid all the groundwork that allows you to put together simple (yet quite powerful and useful) circuits with relative ease. Op amps are especially easy to work with, although you would be hard pressed to find anyone on the Caltech campus who knows how those chips work at the transistor level. (Some Caltech grads have probably gone down that road, however, if they ended up with jobs at Analog Devices, Texas Instruments, or other companies that make analog chips.)

The PCB you just used was designed using free software available from ExpressPCB, where they take your design and fabricate boards at low cost. The chips are available from various vendors, and the *surface-mount* components were added to the board by hand using solder paste and a glorified hair dryer that goes up to 300C. The next step up is to design your circuit, order the parts, and send everything out for the robots to assemble.

Like programming, making custom PCBs is quite simple and easy once you get over the initial learning barrier. If you find yourself working in a laboratory setting, this can be a useful skill to have, especially because not so many of your peers will possess this experimental-physics superpower. And, as you have just seen, custom circuits can easily yield order-of-magnitude improvements over commercial instruments.