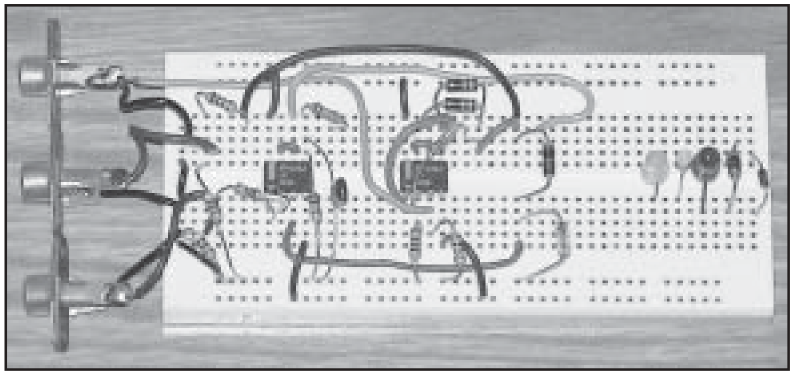
I-V Curve Tracing With A PC

Design of an Inexpensive PC Sound Card Curve Tracer

Dr George R. Steber, WB9LVI



Curve tracing is an interesting way to learn more about electronic components. By curve tracing, we mean making a graph of current versus voltage (an I-V curve) to display the basic characteristic of the electronic device. The I-V curves of diodes, LEDs, transistors and other devices are very helpful in understanding their nonlinear operation. Such curves are useful in establishing circuit models as well as determining bias points, load lines and limitations of the devices. For many, curve tracing seems to be a lost art and many experimenters rely only on curves supplied by the device manufacturer.

Unfortunately, the manufacturers' I-V curves are not always handy or the devices may not be documented at all. Of course, there are commercial curve tracers that will do the job but they are outside the realm of most experimenters. In this article I will attempt to remedy that situation and describe for you an economical way of obtaining the I-V curves of two and three terminal devices using a PC, a simple circuit, a *Windows*-compatible sound card and a bit of software.

To my knowledge, no one has previously used the sound card inputs of the PC for curve tracing without resorting to fancy modulation schemes. There is good reason for this as most sound card inputs are ac coupled. That is, there is a capacitor in series with each input that effectively removes the dc value of the signal. When tracing a nonlinear device, like a diode, the actual dc value must be used or accuracy will suffer. While the PC sound card has a nice pair of 16 bit analog-to-digital (A/D) converters of relatively good accuracy they would be of little value unless this coupling problem was solved. A novel and completely new method that retains the dc value when using a sound card was developed by the author and disclosed in a recent article.[1] One outstanding feature of the procedure is that it can be implemented without modifying the sound card or even opening up the PC case.

For this project I have designed a new circuit that is much simpler than the one described previously (see Note 1) and have also written new software to make it easier to use. An abbreviated article describing construction and operation of this new curve tracer appears in the July 2006 issue of *QST*.[2] Presented here are the technical details behind this unusual system, dubbed Curve Tracer II. In case you don't have the *QST* article handy I will also present some material on installation and operation of the curve tracer as well as theoretical details and additional application material for tracing transistors and integrated circuits. It's a project that should appeal to almost anyone involved in electronics.

This project requires a PC running the *Windows* operating system. You can use one of the newer 3 GHz PCs or dust off your old 200 MHz PC. Of course, there are no guarantees that this project will work with your system. But it has been tested with a 200 MHz Pentium Pro, a 500 MHz Pentium III, and a 1.1 GHz AMD Athlon processor running *Windows 98SE* and *Windows XP* with *Sound Blaster (SB) Live!, Value Edition*.

So, if you have a Pentium or AMD PC with a *Windows*-compatible sound card, you probably have the basis for a very good *Windows*-driven curve tracer. All you need to do is build the simple circuit described, connect it to your computer sound card and run the program. You will be able to accurately plot the I-V characteristics of many two and three terminal devices such as resistors, diodes, Zener diodes, LEDs, transistors, integrated circuits and others. Data is captured via the sound card stereo input, processed and plotted on the screen. The circuitry is very low cost (less than $2.00) and uses readily available parts. You can build the circuit on a solderless breadboard like I did, or design a circuit board for it. In any case you will need a digital voltmeter (DVM) for calibration purposes. To get started it will be helpful to review some concepts of curve tracing.

DC and AC Curve Tracing

Curve tracing with dc signal levels is simple in principle, and requires an adjustable dc power supply, ammeter, voltmeter and current limiting resistor, R, as shown in Figure 1A. In this case only a positive voltage is shown, but both polarities are often used for tracing. One adjusts the supply voltage while monitoring the meters and writes down the meter readings as a table of I and V values. From the table, data points can be plotted to obtain a graph, like the one shown at Figure 1B. While this works well, it has several drawbacks. One has to manually adjust the voltage at reasonable intervals, read and write down the data, and finally plot it. You also have to switch power leads if you want to plot negative voltages. If more than a few devices need characterization, an automatic method is preferable. Clearly, if you have a bag of diodes, LEDs or Zeners that you just bought at a hamfest and want to test them, this is not the way to do it.

[1]Notes appear on page 9.

9957 N River Road
Mequon, WI 53092
steber@execpc.com

Classic ac curve tracers are faster, but slightly more complicated. You apply an ac voltage to the device under test (DUT) and plot its current versus voltage on an oscilloscope. You need an ac voltage source, transformer, a few resistors and, of course, the oscilloscope. A straightforward curve tracer circuit is shown in Figure 2A.

Transformer T1 is used to provide isolation and voltage reduction from the ac source, usually 120 V ac, 60 Hz. Resistor R_1 limits the current while R_2 is used to measure current. An oscilloscope is used in dc-coupled x-y mode for display. The scope x input measures the DUT voltage and the scope y input measures the voltage across R_2, which is proportional to current. For example, if R_2 equals 100 Ω and there is 1 V across it, it follows that there is 10 mA through the DUT. This means the y channel is calibrated for 10 mA per V. Notice that since the ground point for the scope is at the connection of R_2 and the DUT, the current signal measured by R_2 will be inverted. This can usually be corrected at the oscilloscope by reversing the polarity of the channel. Notice that the ac tracer sweeps both positive and negative voltages so a diode is sometimes placed across the DUT to provide reverse voltage protection. A typical I-V curve is shown at Figure 2B.

While both curve tracing approaches are straightforward, there are some things to consider. The dc curve tracer is painfully slow and prone to errors. The ac curve tracer is faster but requires a transformer and an oscilloscope, which requires specialized knowledge to use and calibrate properly. Since we are often saving, comparing and printing I-V curves, a PC can be of considerable help. So, now, let's consider how we can use a PC and a sound card for curve tracing.

Sound Card Software Model

Modern PC sound cards are marvelous examples of technology. They are low cost, low noise and feature high sampling rates, operating well into the high audio range. Most impressive is that they have dual 16 bit A/D converters, dual 16 bit D/A converters and can operate at full duplex. While there are a few cards that are dc coupled, the vast majority of them are ac coupled. This means they can't be used directly in many data acquisition tasks requiring measuring dc voltages. Hence, most people have given up using them for this type of application or have conjured up fancy modulation schemes that require ac carriers. This article describes a method that measures dc voltages with your ac sound card, without modifying it. This method uses a software model of the ac input circuit to predict and reverse the effects of the ac coupling. Here is how it works.

AC coupling is accomplished by means of large capacitors, which allow the frequency response to go down to very low frequencies, typically 20 Hz. A generic model of one channel is shown in Figure 3A. Figure 3B shows the step voltage response. The signal rises quickly to the value of the step, then decays exponentially to zero. If you apply a step voltage to your sound card input, record it, and view it, that is what you will see. The voltage transfer function of this circuit, using the Laplace variable, *s*, is given by:

$$\frac{V_2}{V_1} = G_1(s) = \frac{R_1 C_1 s}{R_1 C_1 s + 1} \qquad \text{[Eq 1]}$$

Now consider the addition of another transfer function, $G_2(s)$ in cascade with $G_1(s)$ as shown in Figure 3C. If we could make $G_2(s)$ equal to $1/G_1(s)$, the overall transfer function would be unity and we would see a flat, non-decaying step response at point V_3. Notice from Equation 1 that there is a zero at the origin. This forces $G_2(s)$ to have a pole at the origin, making it marginally stable. If we limit our time interval, however, the potential instability can be managed quite well.

Since we don't want to open the PC and add the circuit for $G_2(s)$ to the sound card, what can we do? We can build a software model of $G_2(s)$ in the computer. Basically we

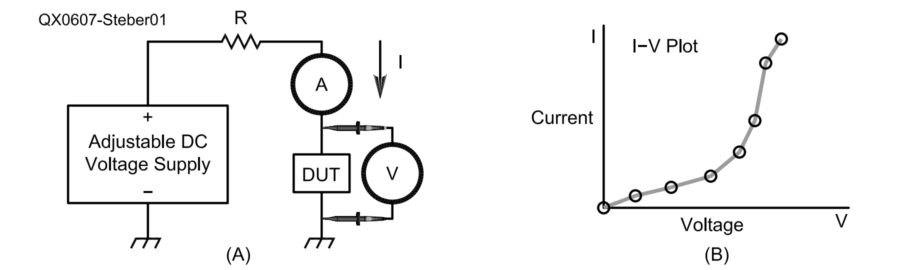


Figure 1 — Simple dc curve tracer setup and I-V plot example.

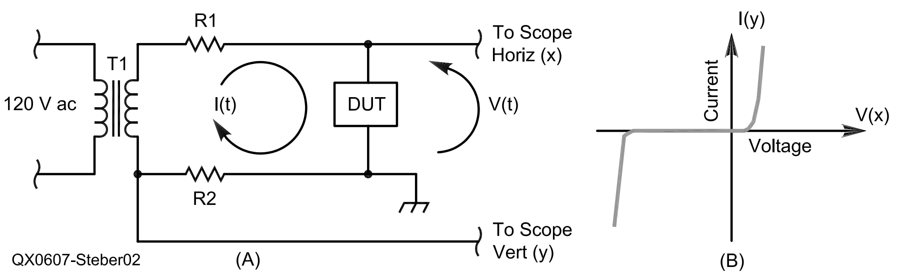


Figure 2 — This schematic shows a basic ac curve tracer setup.

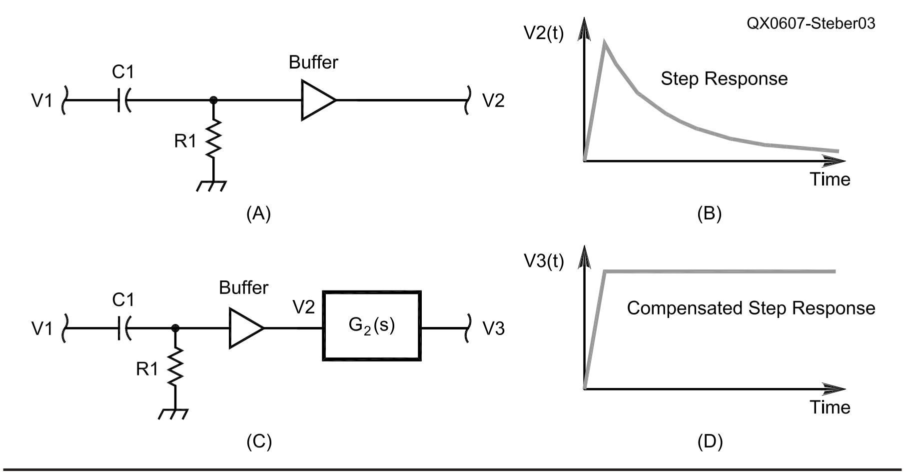


Figure 3 — The circuit at A shows a typical sound card input and B shows the step voltage response. The circuit at C shows the addition of a hypothetical compensation circuit and D shows the corrected step response.

use a discrete-time digital filter in software that recursively solves a difference equation given the initial conditions and input data sequence from the A/D. In effect, we use a sampled version of $G_2(s)$, which will now be called $G_2(z)$. How can we design a digital filter that satisfies the specifications of the analog filter $G_2(s)$? Mathematically we use the following bilinear transformation:

$$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}} \quad \text{[Eq 2]}$$

where T is the sampling period and z is the z transform of s. In our case, the resulting $G_2(z)$ is given by the following:

$$G_2(z) = 1 + \frac{T}{2R_1C_1}\frac{1+z^{-1}}{1-z^{-1}} = \frac{V_3(z)}{V_2(z)} \quad \text{[Eq 3]}$$

To prevent aliasing, we must limit the input frequencies to less than ½ of the sampling frequency. Implementing the digital filter given by Equation 3 on the sound card data will correct it and compensate for the effect of the coupling capacitor. One slight problem remains; we don't know the value of the product R_1C_1. We can determine this parameter of the filter experimentally, however. If we apply a step input to the sound card, the resulting output can be observed on the screen. We can then adjust the parameter of the filter until we get a flat step response. Knowing the exact value of the step amplitude, as measured with our DVM, also allows us to accurately calibrate the sound card dc voltage. This procedure should work well if we limit the time duration of the input. In practice it works well for a curve tracer.

Sound Card Interface Circuit

The simple ac curve tracing circuit mentioned earlier is not appropriate for this approach since it doesn't provide a step output for calibration and it can't be started and stopped. So a special interface circuit was designed for this project. That circuit is shown in Figure 4.

A few general comments are in order. Sound cards typically have a 3.5 mm stereo jack for input and output so you will need a cable, preferably shielded, to connect between the card and the circuit. Actually, you will need two stereo cables (one for input signals and one for output signals) with the 3.5 mm stereo plug on one end and the appropriate plugs (like RCA plugs) on the other end. Such cables are commonly found at your local electronics or AV store.

Regarding the circuit, make sure to use LM358 op amps. Substitutes like the TL082 or TL272 will not work well here. Also, use a good, well-regulated dual power supply of ± 12 V with at least 100 mA capability. The DUT may require a pulsed current of 60 mA or more, and the supply voltage should not droop during this time.

The interface uses one sound card output (either left or right channel) and two sound card inputs (the left and right channels). Although the schematic shows the inputs separately for clarity, they are in fact combined into a single stereo jack. The line output will provide a special sine wave as described later. Resistor R_1 is there to provide a ground reference for the sound card line output. U1A and U1B form a dc power amplifier to boost the line output and provide more current capability. This power amp does not provide equal current source and sink capability since the LM358 is unbalanced in this respect. Typically an LM358 op amp can source 40 mA and sink 20 mA. Since we are using two op amps in our power amplifier, we should be able to double those figures. My breadboard circuit actually produces less, sourcing only up to 65 mA and sinking up to 32 mA. Nonetheless, this is a stable, overload-protected circuit with little dc offset, which is why I used it. Observe that in the case of diodes or LEDs, one can always orient the DUT so that current is sourced to the DUT, if desired. Note that resistor R_2 controls the gain of the power amp. If you need to increase the gain, decrease the value of R_2.

U2B is a simple step generator, providing a slow step voltage, at about ½ the positive supply voltage, at test point TP1 when S1 closes. Use a single-action *momentary* switch for S1 and avoid using a slide switch. Jumper block J1 is used to select the step or sine wave signal for the sound card. The step voltage is only used once to calibrate the system. Note that there is also a test point, TP2, near pin 1 of J1. With no signal applied to the power amp, the voltage at TP2 should be less than 5 mV dc. This will avoid having a dc voltage applied to the sound card inputs

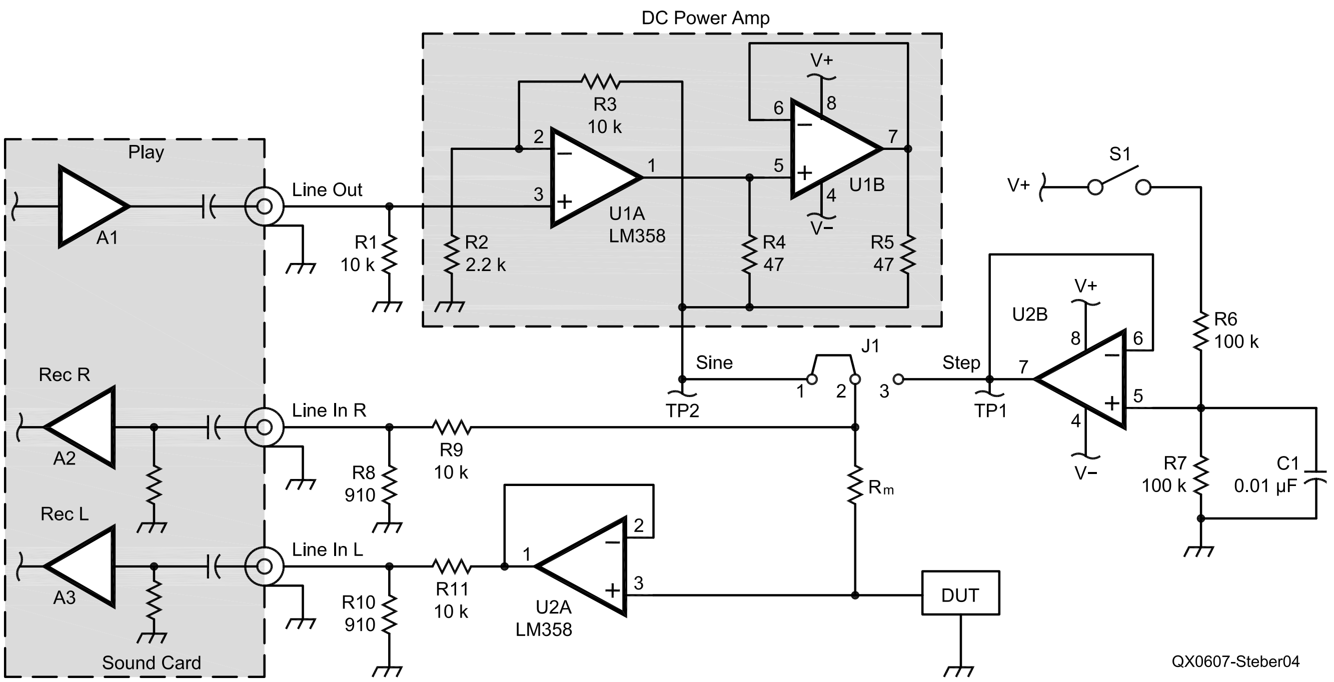


Figure 4 — Curve tracer circuit and the interface to the PC sound card.

and causing an error. In my circuit, the voltage at TP2 is around 3 mV.

A current measuring resistor, Rm, is connected in series with the DUT. This resistor is typically 100 to 1000 Ω. The exact value can be chosen to suit requirements. R_8 and R_9 form an attenuator to reduce the voltage to something the sound card line input *right* channel can accept without overloading, typically around 1 V. (My *SB Live* begins clipping at 0.82 V.) The voltage across the DUT is buffered by U2A, similarly attenuated and fed to the *left* channel. The *current* through the DUT is the difference between the right and left channel voltages, divided by Rm. So we need to know the value of Rm accurately, say within 1%. Given the value of Rm, the software calculates the current in the DUT.

The circuit is potentially harmful to some devices, since in normal operation the sine wave sweeps through both positive and negative voltages up to 10 V. To restrict the voltage sweep to only positive or negative excursions, a diode can be appropriately placed across the DUT. The sine wave may flattop as more current is drawn from the power amplifier by the DUT. This is normally not a problem. Increasing Rm may help reduce this effect. This circuit can easily be modified to handle higher voltages and currents. It is only a starting point. As it stands, a large number of devices can be curve traced with this circuit.

PC Curve Tracer Operation

Basically the curve tracer works by applying a test signal, a few cycles of a 240 Hz sine wave, to the DUT and measuring the current through the device and the voltage across it. Referring to Figure 4, the shortened sine wave is generated in the computer and is output to the line output of the sound card amplifier A1 when the Start button on the computer screen is clicked. The sine wave is boosted by the dc power amplifier and presented to the top of Rm via pin 2 of J1. This sine wave produces a current in the DUT and a voltage across it, just as in the simple dc curve tracer discussed earlier. The voltage at the top of Rm is sent to the right channel of the sound card and the voltage at the bottom of Rm is sent to the left channel. Knowing the voltage across Rm enables the computer to calculate the current using Ohm's Law. Notice that the voltage at the bottom of Rm is also the voltage across the DUT, which is sent to the left sound card channel.

As mentioned earlier, we need to calibrate the software model so that it exactly reverses the effects of the ac coupling. For this purpose we use a specific test input, namely a step voltage and the circuitry associated with U2B. With the DUT removed and the jumper on J1 set between pins 2 and 3, we start a software capture. Shortly after clicking Start we initiate a step voltage by pressing S1. Holding S1 closed, the step voltage amplitude is also measured with a digital voltmeter at TP1. The program captures the step voltage and is now ready for calibration of the software model, given some input from you. More specific details on how to do this are given later. The calibration is then saved and used for all further curve tracer operations. The calibration values have been found to hold up well over many months without drift, but can always be redone when needed.

Sound Card Considerations

A low-distortion, low-noise, full-duplex sound card is desirable. The economical *Sound Blaster Live, Value Edition* fills the bill nicely and probably many others will too. Since I cannot test them all, I will restrict my attention to this one. Referring to Figure 4, we see that A1 is the line output amplifier, and A2 and A3 are the right and left channel line inputs of the sound card. The mixer control in your *Windows* software controls the gain of these amps. The level of the line output signal is controlled in the Play section of the mixer via the Wave and Spkr sliders. So to control the amplitude of the sine wave output, just adjust the sliders.

The gain of the line input amplifiers A2 and A3 are controlled in the Record section of the mixer by the Line slider. They will saturate if the input voltage is too high, regardless of the Record setting in the mixer. On the SB, this occurs at 820 mV. Fortunately, this is easy to detect, because the captured signal will flattop. Since the linear gain of the amplifier is lost at saturation, we may also see a fictitious current bump in the DUT current trace at the extreme voltage limits. A real time oscilloscope function is provided in the software to help monitor for this condition. So, if you see a current bump in the I-V trace when there is no DUT in place, it is likely caused by an amplifier saturating. Just back off a little with the Play output signal or just ignore it.

A note is in order on earlier Sound Blaster sound cards such as the SB16 and AWE 32 since there are so many of these still around. Unfortunately, they do not provide true full duplex operation and are noisy. The same may be said of SB *compatible* cards, so be wary. For example (with the latest drivers) the SB AWE32 can only play unsigned 8 bits and record signed 16 bits at once. It also has a built-in amplifier that may overdrive the interface circuit. My advice is not to use any of these cards.

Curve Tracer Software Installation and Operation

The Curve Tracer software is available on the *QEX* Web site and is zipped for fast downloading.[3] Unzip it to a new folder and you are ready to go. Just run the executable (exe) program. It was tested with *Windows 98* and *XP*. When you run the software you may get a message like "Required DLL file MSVBVM60.DLL was not found." This is a Visual Basic run time file and is on many systems. If not found, you will need to obtain it and install it on your system. It is freely available from Microsoft and other sites on the Web. It is usually available as Visual Basic 6.0 SP5: Run-Time Redistribution Pack (VBRun60sp5 .exe) and is a self-extracting file. Download takes about 6 minutes at 28.8 kbps.

If you get the message "Component 'COMDLG32.OCX' or one of its dependencies is not correctly registered: a file is missing or invalid." when you try to run the program, you will need to register it on your system. This file is also available free from Microsoft and other sites on the Web. More

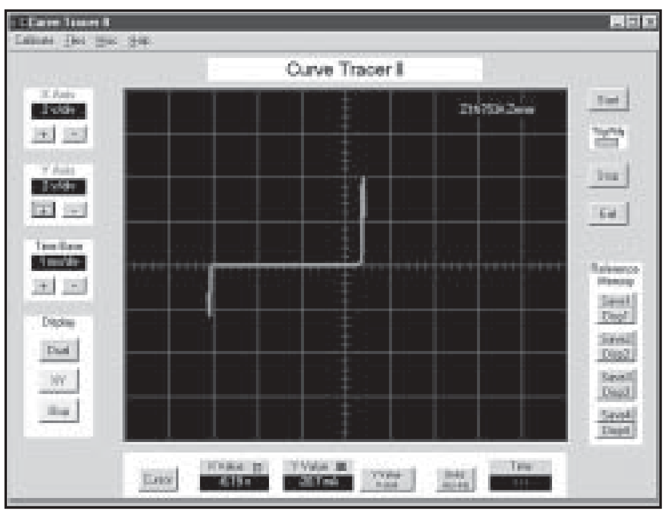


Figure 5 — The main window of the curve tracer program showing Zener diode plot.

details are included with the software.

If you just want to experiment with the Curve Tracer program, go ahead, as it does not modify the registry or install any other material on your computer. You can remove it by just deleting the entire folder where you copied the software.

Figure 5 shows a screen shot of the Curve Tracer II. The oscilloscope display is in the middle of the screen. It functions like a real oscilloscope and has controls for dual trace and X-Y display as well as sensitivity settings on the left side. The X-Y mode is essentially the I-V mode with current on the vertical axis and voltage on the horizontal. The *origin* is at the center of the screen. This screen shot shows the I-V display of a 1N753A Zener diode. It is rated at 6.2 V at 20 mA. by the manufacturer. I measured 6.19 V at 20.7 mA with my Zener.

Below the scope screen is the cursor readout, which functions with the on-screen cursor and is controlled by your left mouse button. It works in both dual and X-Y mode. On the right side are the Start button, Stop button and TrigRdy LED. The Start button initiates a capture, which automatically stops after 1 to 2 seconds while processing the data and displaying it on the screen. In case something is wrong, like you forgot to turn on the circuit power supply, the Stop button may be used to stop the capture. Below the Stop button are four temporary memories for saving and comparing data. Additional features include a second cursor (right mouse button) for time measurement, provisions for printer output, screen capture and file saving for use by *Excel* or other programs.

System Setup and Calibration

First make sure the circuit is powered and connected to the sound card line inputs, as shown in Figure 4. Check the mixer program that came with your sound card or the one that came with *Windows* because you may have changed its settings. When you start the program, a little notice comes on the screen to remind you to do this. Basically you want to set the output level, input gain and stereo balance. The details on how to do this will obviously vary from system to system. Here is how it's done with the SB.

In the mixer Play section, enable Wave and Spkr, set the sliders to maximum and mute all others (including Line to avoid audio feedback). In the Record section, enable Line, set it to maximum and mute all others. Set the stereo balance to center for all controls. These settings are *important*. For example, if you forget to mute Line in the Play section, there will be errors.

Calibration needs to be performed only once since parameters are saved in a file CTinit upon exit. Follow these steps closely so good accuracy can be achieved. First, remove the DUT. Next, set the jumper J1 on the circuit so that the step voltage is applied to Rm. Run the program. Click Start and wait for the TrigRdy LED to change from green to red. Now create the step voltage by closing switch S1 on the circuit. Upon capture of the step voltage data, the TrigRdy LED will revert to green. With S1 still closed, measure the step voltage at TP1 with a good voltmeter, preferably a digital type, and write it down. View the data captured by the program using the left cursor in *dual* mode. You should see a large step voltage on the left (L) channel (green) and a small voltage on the right minus left (R – L) channel (blue). See Figure 6 for an example. [The blue line on the computer screen is shown as a heavy horizontal line on Figure 6. — *Ed.*] The data should start out clean without much noise. If there is a lot of noise at the start of capture (probably due to S1 bounce), repeat the capture until it is clean.

The captured data is used to calibrate the program. Click on the Calibrate button to begin. Change the values shown in the Calibrate box as needed. All calibration is done on the originally captured step voltage. It does not need to be recaptured each time. Use the left cursor (left mouse button) to measure the data on the L and (R – L) channels as needed.

Adjust the Channel Balance value until the (R – L) channel is close to zero. This is essentially the voltage across Rm, which should be zero since the current in it is zero because the DUT is not present.

Adjust the Compensation value until there is no droop or rise of the step voltage of the L channel. After its initial rise, the step should flatten out to be a straight horizontal line across the window of the scope. If it is over corrected, the line will rise while if under corrected, it will decline. See Figure 6 for an example of over, under and normal compensation. Make corrections in *very* small amounts. Use the cursor to verify it is straight by moving it across the waveform while looking at the readout.

The volts-per-bit calibration is done on the L channel of the sound card. Use the left cursor and measure the voltage of the L channel. Adjust this value until the cursor value is close to the value you measured with the voltmeter earlier. Each time you click the Apply button in the Calibration box, these values are applied to the captured step data. You will immediately see the results on the screen. Repeat these steps until the best calibration is achieved.

Other Calibration Box Values

The Trigger Level controls the voltage level at which the beginning of the input signal is captured. If you get false captures, you can increase this value. If the value is too high, you will miss the leading edge of the captured waveform. If it is too low, you will get false triggers due to noise.

The reason for the Invert Input box is that some sound cards invert the signal. The SB Live inverts the data, so this box needs to be checked to correct the input polarity. To determine if your sound card inverts the data, look at the step waveform of the L channel. It should start out positive for a positive input such as provided by the hardware circuit. If not, you need to check the box to invert the data.

Enter the value of the current measuring resistor in ohms in the Rm box. The cursor readout will display device current based on this value using Ohm's Law.

Here are values for my SB Live card. Channel Bal: 0.9551; Compensation: +10; Volts Per Bit: 0.00033982; Trigger Level: 200; Rm: 98.8 and Invert: Box checked.

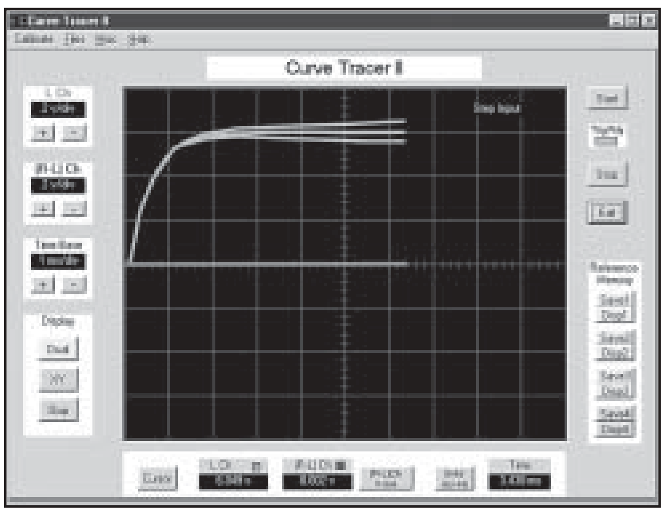


Figure 6 — The main window of the curve tracer program showing step voltage input.

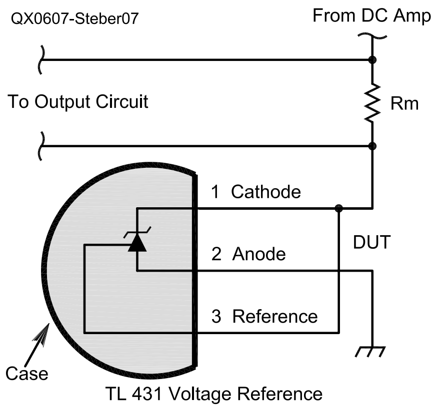


Figure 7 — Voltage reference diode circuit.

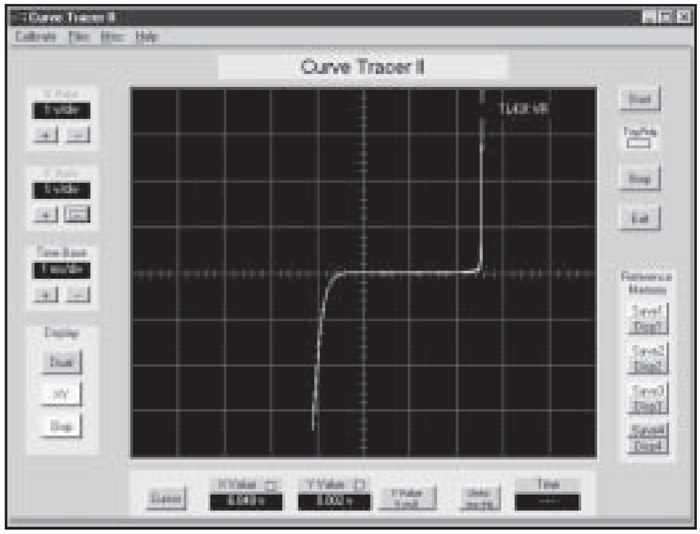


Figure 8 — Voltage reference diode I-V curve.

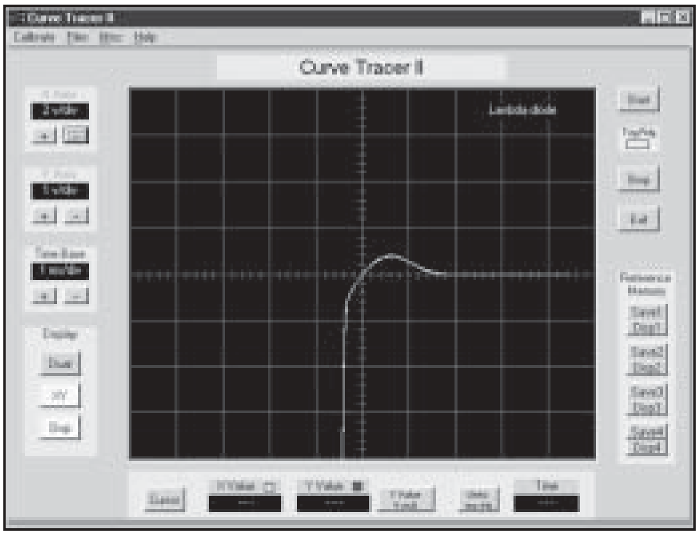


Figure 9 — Lambda diode I-V trace.

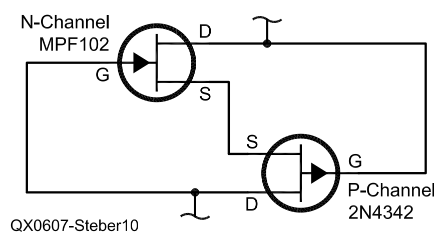


Figure 10 — Lambda diode circuit.

Obtaining I-V Traces

It is easy to curve trace a component. Make sure the hardware circuit is connected to the sound card, J1 connected to *sine*, the program calibrated, and the DUT connected. So you know what to expect, use a known resistor, diode or Zener as your first DUT. To curve trace it, follow these steps.

Click the Start button. The green TrigRdy LED will *not* turn red as the sine wave generated is only a few cycles long. After 1 to 2 seconds the data will be captured, processed and will appear on screen. You have now curve traced your first component! If for some reason, the trigger did not occur, the TrigRdy LED will turn red and can be manually reset using the Stop button. Normally this is not required unless there is something wrong with your setup. You cannot exit the program unless the TrigRdy LED is green, a precaution to make sure all processes are shut down.

Curve tracing various components is interesting and instructive. Resistors, of course, plot as straight lines. Their value can be checked by choosing a point on the line with the cursor (yielding the voltage and current at a given point) and using Ohm's Law. Diodes, Zeners, and LEDs show plots similar to Figure 5. Capacitors and inductors do not plot as their characteristic elliptical shape because there are not enough cycles in the test sine wave to reach steady state.

An interesting practical application is checking of voltage reference devices like the TL431, LM336Z and others. Figure 7 shows a circuit for testing a TL431 programmable reference device. The typical reference voltage given by manufacturer is 2.495 V at 10 mA. My device measured 2.52 V at 11 mA, which is within tolerance. Such reference diodes are also useful for checking the veracity of the curve tracer. The I-V curve of the TL431 is shown in Figure 8.

Speaking of diodes, a strange device called a lambda diode makes an interesting I-V curve as shown in Figure 9. Notice the large negative resistance region to the right of the origin. This device has been found to be useful in oscillator circuits. You can make your own device, like I did, by combining back-to-back N-channel and P-channel FETs as shown in Figure 10.

Curve Tracing of Transistors and Integrated Circuits

Figure 11shows the collector curves of a 2N3904 transistor. Multiple curves are displayed using the reference memory, with each curve saved separately. You need to provide a simple base current source for transistors. Such a base current circuit is shown in Figure 12. R1 is chosen to be very large to essentially form a current source to the base. Adjustment of R2 varies the base current. Normally you would not trace the negative voltage region as done here as it might damage the device. In this case it is interesting to see the negative resistance region of a reverse biased transistor (on the left side of the origin) and fortunately the device was not damaged. FETs can be traced in a similar manner.

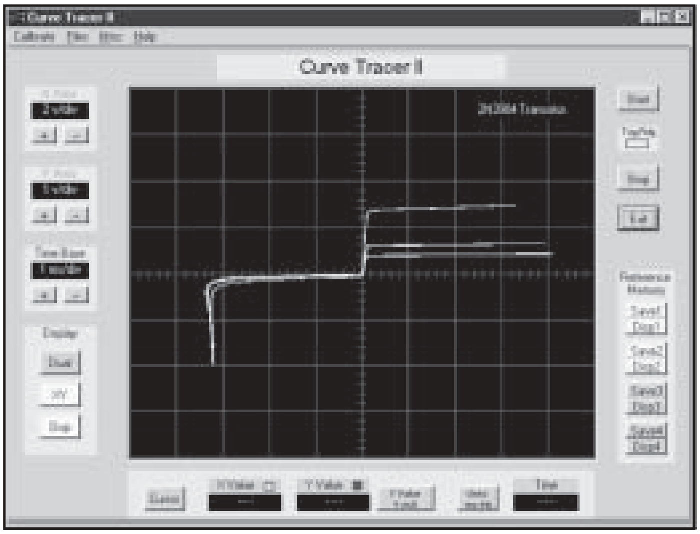


Figure 11 — Collector I-V curves of 2N3904 transistor.

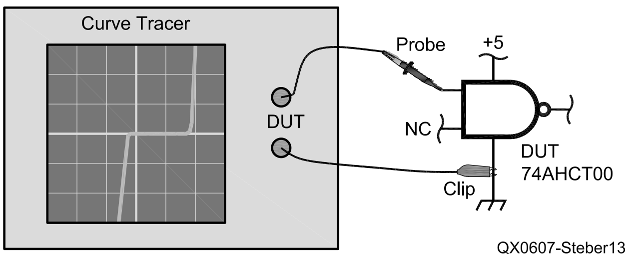


Figure 13 — Integrated circuit test setup. [Normally any unused inputs would be tied to Vcc. In this example, the device was tested with no connection to the second input or to the output —*Ed.*]

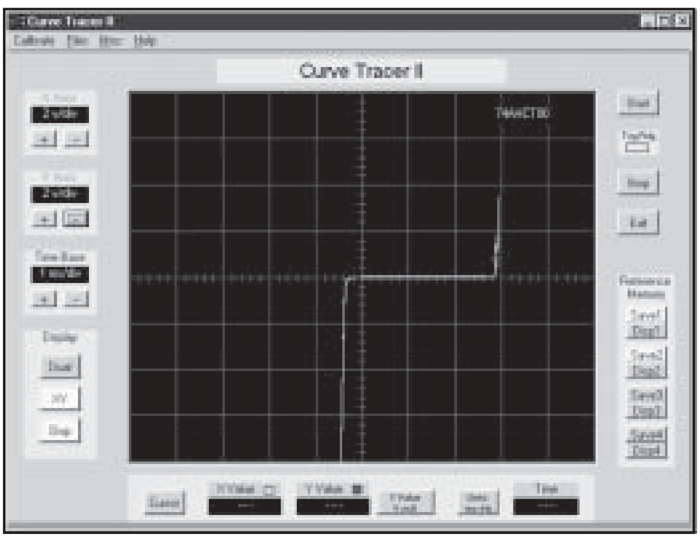


Figure 14 — Sample I-V curve of 74AHCT00 integrated circuit.

Curve tracing of integrated circuits under various conditions has been described in a recent article for purposes of rework.[4] Tracing the inputs and outputs of logic ICs — under power on and off conditions — produced some unusual curves. So I decided to try it myself. Consider the 74AHCT00 IC shown in Figure 13. Curve tracing between one of the inputs and ground produced the curve shown in Figure 14. Curve tracing of integrated circuits is a new technique, at least to me. So if you endeavor to try this, be prepared for unusual curves and possible destroyed ICs. You may learn how to test and repair some circuits during the rework process, though, as described in the article of Note 4.

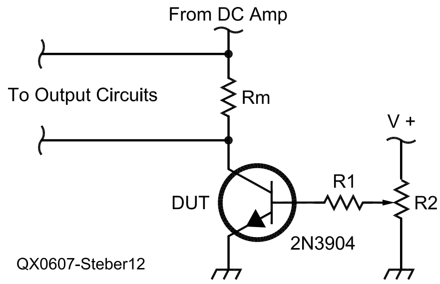


Figure 12 — Simple base current circuit for transistor.

Final Thoughts

Files of several components that I have traced and other curves are included in the zipped file package mentioned earlier. You can use the program to read them and see what a trace looks like without having to build the circuit. There are probably many more applications for this curve tracer, perhaps like analog signature analysis. Let me know if you come up with an interesting application.

The Curve Tracer II was compared to an ac curve tracer built using my Tektronix TDS360 Digital Real Time Oscilloscope and the results were in good agreement. With any luck, you may achieve similar results. In any case, I hope I haven't thrown too many *curves* your way!

Notes

[1]George Steber, "Tracing Voltage And Current," *Circuit Cellar*, January 2004

[2]George Steber, "A Low Cost Automatic Curve Tracer," *QST*, July 2006, pp 32-36.

[3]You can download the software associated with this article from the ARRL Web at **www.arrl.org/qexfiles/**. Look for 7x06Steber.zip.

[4]Tom Mathews and Timothy Toroni, *Design Feature*, "Rework Within Your Reach," September 2, 2004, *EDN*, pp 75-81.

George R. Steber, PhD, is Emeritus Professor of Electrical Engineering and Computer Science at the University of Wisconsin-Milwaukee. George, WB9LVI, has an Advanced class license and is a Life Member of ARRL and the IEEE. Dr Steber has written several articles for QST, *the most recent one being "A Low Cost Automatic Impedance Bridge" in the October 2005 issue and a companion article in the Sep/Oct issue of* QEX, *"An LMS Impedance Bridge." He has industrial experience as a company officer, consultant, and product designer, with 18 patents issued. In his spare time he enjoys racquetball, reading, editing video, astronomy and playing classical, Broadway and big band favorites with his trumpet in a local band. You may reach him at* **steber@execpc.com** *with "Curve Tracer" in the subject line. Send your message in text mode.*