A Low Cost Automatic Impedance Bridge

George R. Steber, WB9LVI

A PC, a sound card and some software team up to make a powerful LCR meter.
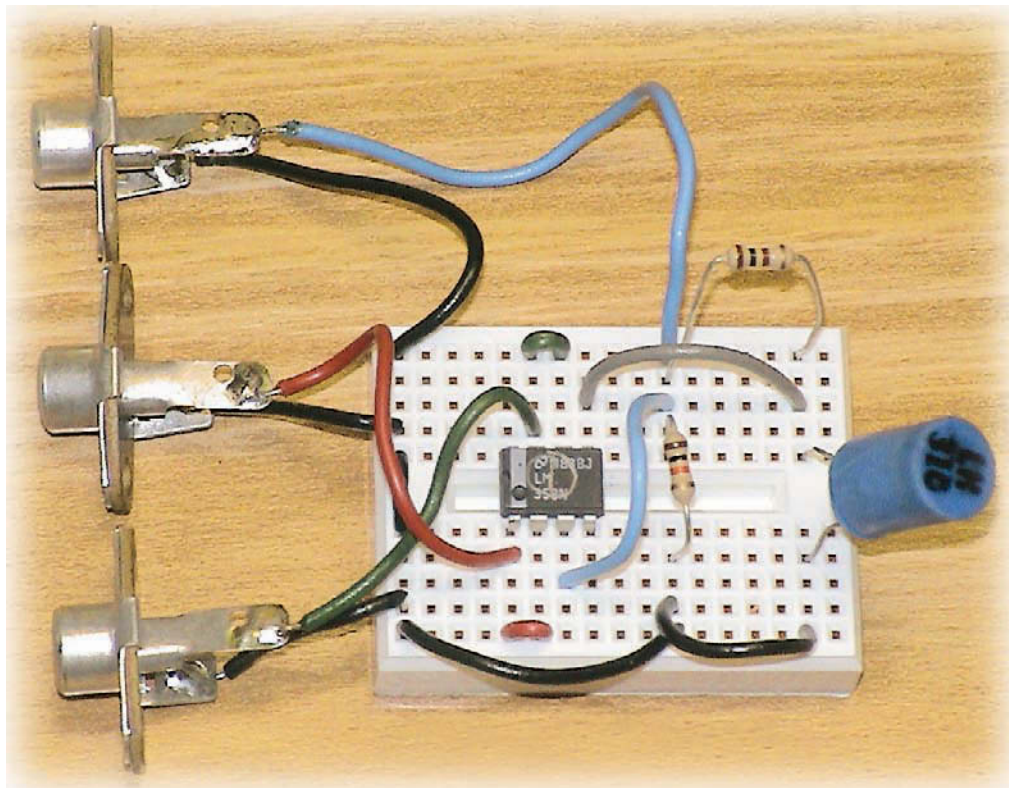
The concept of impedance is fundamental to understanding and utilizing electrical components and circuits. There are numerous instruments available to measure impedance, including the ubiquitous ohmmeter for resistors; resistance bridges; ac bridges for capacitors and inductors; automatic LCR bridges, and vector impedance meters. Many of the more sophisticated instruments are beyond the realm of the average amateur radio experimenter. Hopefully, this project will remedy that situation as well as remove excuses for not measuring those accumulated parts stored away in the old junk box.

While working on another project (curve tracing with a PC), I hit upon the idea of using a PC with a sound card to measure impedance automatically.[1] After reviewing some theory, writing a bit of software, and playing around with vari-

ous circuits, this project was born. It turned out to be much more than I had originally hoped for, yielding a wide range, low cost, impedance measuring system. Since it's based on LMS (least mean square) theory, it's called an LMS impedance bridge or meter. More in-depth technical details about this project are covered in a separate article in *QEX*.[2] In this article I will mainly cover how to build your own version and how to get it

Table 1
Bridge Range Versus R_m Values for Bridge Frequency of 1225 Hz

R_m (Ω)	Inductance (L)	Capacitance (C)
10	12.99 µH - 12.99 mH	0.1299 µF - 1299 µF
100	129.9 µH - 129.9 mH	0.01299 µF - 129.9 µF
1 k	1.299 mH - 1299 mH	1.29 nF - 12.9 µF
10 k	12.99 mH - 12.99 H	1299 pF - 1.29 µF
100 k	129.9 mH - 129.9 H	12.99 pF - 0.129 µF

up and running. It's a project that should appeal to almost anyone involved in electronics. All you need to do is build the simple circuit described, connect it to your computer sound card, and run the program. This impedance bridge allows you to automatically measure inductors, capacitors, resistors, input impedances, audio transformers, negative resistances and more, at a wide range of audio frequencies. It has outstanding capabilities and accuracy.

The cost of the project is less than $1 (yes…one dollar!), not counting the PC and power supplies. The circuit uses only two resistors and a dual op-amp. It can be built on a solderless breadboard like I did or you can design a printed circuit board for it. In any case, a digital voltmeter (DVM) is desirable for calibration purposes, although even this is not a definite requirement.

Nearly any PC can be used, as it doesn't have to be modified in any way. You can use one of the newer 3 GHz PCs or dust off that old 200 MHz machine that's sitting on the shelf. No need to open the cabinet either, since access to the sound card stereo line jacks is all that is required, and that can usually be accomplished from a panel on the rear of the computer. While I won't *guarantee* that this project will work with your system, I will say that I have tested it with a 200 MHz Pentium Pro, a 500 MHz Pentium III and a 1.1 GHz AMD Athlon processor running *Windows 98* or *XP* with a Sound Blaster (SB) Live! sound card.

So, if you have a Pentium or AMD PC with a *Windows*-compatible full duplex sound card, you may have the basis for a very good *Windows*-driven impedance measuring system.

Impedance 101

Since we will be measuring impedance it might be a good idea to review some basic concepts. Impedance is a more general form of the resistance parameter. Impedance can have a resistive and a reactive component. For resistors, the reactive part is very small, except for wire wound types. As shown in Figure 1, inductors and capacitors have both a resistive and reactive component. The resistive part is shown modeled in series with the reactance. Although the series model is used here, other models are sometimes used, with parallel resistors. At a given frequency, impedance can be written in either polar (vector) or rectangular form as in Eq 1.

$$Z = |Z| \angle \theta = R \pm jX \qquad [1]$$

where Z is impedance in ohms, |Z| is the

magnitude of Z, θ is the angle of Z, R is the resistive part of Z and X is the reactive part of Z. The two forms in Eq 1 are related by:

$$|Z| = \sqrt{R^2 + X^2} \quad \text{and} \quad \theta = \tan^{-1}(X/R) \quad [2]$$

Our impedance meter will measure impedance and display the results in both of these forms—rectangular and vector—on the computer screen. In addition, the impedance will be automatically converted to capacitance, inductance or resistance, as the case may be, and simultaneously displayed. Other parameters such as the bridge frequency (f), Q (quality factor) or D (dissipation factor), associated with passive components, are also displayed.

The Sound Card Interface Circuit

Figure 2 shows my breadboard setup for interfacing to the PC sound card. Only a few components are required. In this instance, I am using a solderless breadboard, as it is an easy way to get started. Designing a PCB may be a better solution for those interested in a long-term, less fragile system. The blue object on the right is a 10 mH inductor that is being measured. RCA type connectors are on my breadboard for the sound card input and outputs, but this was only for convenience; any other kinds of jacks could be used. Sound cards typically have a 3.5 mm stereo jack for input and output so you will need a cable, preferably shielded, to convert to RCA. 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 RCA plugs on the other end. Such cables are commonly found at your local electronics or audio-video store.

The interface circuit schematic is shown in Figure 3. It uses one sound card output (either left or right channel) and two sound card inputs, the left and right channels. Resistor R1 is there to provide a ground reference for the sound card output. R_m is the reference resistor and Z_x is the unknown, as in Eq 1. Note that the unknown impedance is grounded. Two op-amps, U1A and U1B, provide isolation and buffering of the sound card voltages. They are connected as unity gain, high input impedance/low output impedance drivers. V_r is the sinusoidal voltage applied to the circuit via a line output of the sound card. It is fed back to the sound card input *right* channel via U1A. The voltage across the unknown, Z_x, is buffered by U1B and fed back to the sound card input *left* channel via U1B.

U1 is a low cost LM358 dual op-amp or equivalent, which can be powered from bipolar power supplies of 3 V dc to 15 V
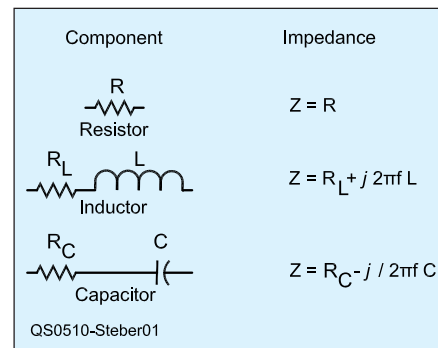

Figure 1—Series equivalent circuits for R, L and C components. Frequency (f) is in hertz, resistance (R) in ohms, inductance (L) in henrys and capacitance (C) in farads.

dc. It is best to keep the power supply voltages low in order to protect the sound card line inputs in case of problems. My circuit runs at about ±3 V dc. A bipolar battery supply could also be made using four AAA batteries, with a center tap to ground.

R_m is a reference value for the software, so it is critical that we know its precise resistance. To enable different instrument ranges we may also wish to suitably change the value of R_m. We will talk more about this later.

This project uses the least mean square (LMS) method to process the signals and calculate the impedance. Other methods of doing this are discussed in the *QEX* article. It has been found that the LMS method produces the best results.

Briefly, here's how it works: A sinusoidal ac voltage, V_r, is generated at a specific frequency in the computer and is sent to the sound card line output jack. Data, corresponding to V_r and V_x, is received in 0.25 second chunks via the sound card line inputs. At 44100 samples/ second, this corresponds to a capture length of 11025 samples. The LMS method is applied to each of these data chunks and the impedance is calculated. This is a mathematically intensive calculation, but it can be easily handled by a modern computer. All of the parameters are calculated and displayed on the PC screen when finished. Four measurements per second can be handled in this manner, even by a slow computer.

Sound Card Considerations

A low distortion, low noise, full duplex, sound card is desirable. A Sound Blaster (SB) Live! card fills the bill nicely, and probably many others will, too. But since I cannot test them all, I will restrict my attention to this one. Refer-
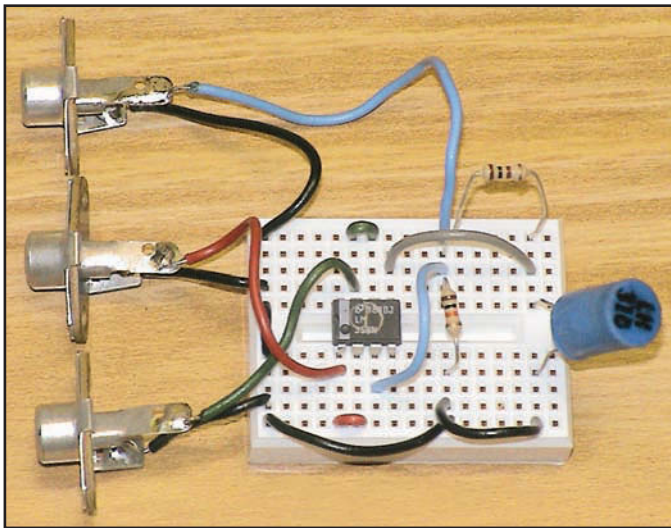
Figure 2—The impedance bridge breadboard setup.

Figure 3—Impedance measuring circuit and the interface to the PC sound card.

scope function in the bridge display so that sine waves can be monitored for possible "flat-topping" or distortion. If a digital voltmeter is handy, the voltage V_x can be measured and used to calibrate the 'scope. Provision for this is provided in the program, but it is not required, and it does not affect operation of the bridge.

One other consideration is balancing of the two input channels. We need to calculate V_r–V_x precisely in the program, so the two channels must be balanced. This is provided for in the calibration section of the program.

A note is in order for earlier SB 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 they are noisy. The same may be said of SB *compatible* cards, so be careful. For example (with 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. Only after some extensive tweaking did I manage to get one working with this program, but the results were not good. My advice is not to use any of these cards.

Impedance Bridge Software Installation and Operation

The LMS bridge software is available on the ARRL 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 *Windows98* and *XP*. When you run the software you may get a message similar to "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.[4]

If you just want to experiment with the LMS bridge program, don't worry, 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 (directory) it is located in.

A screen shot of the impedance bridge is shown in Figure 4. There is a lot of information on the screen. The scope and controls are on the left side. The most important part is in the lower right corner labeled UNKNOWN where all of the relevant data on the measured impedance is displayed.

The bridge is easy to use but some considerations are in order. The value of R_m must be known *exactly* as all results
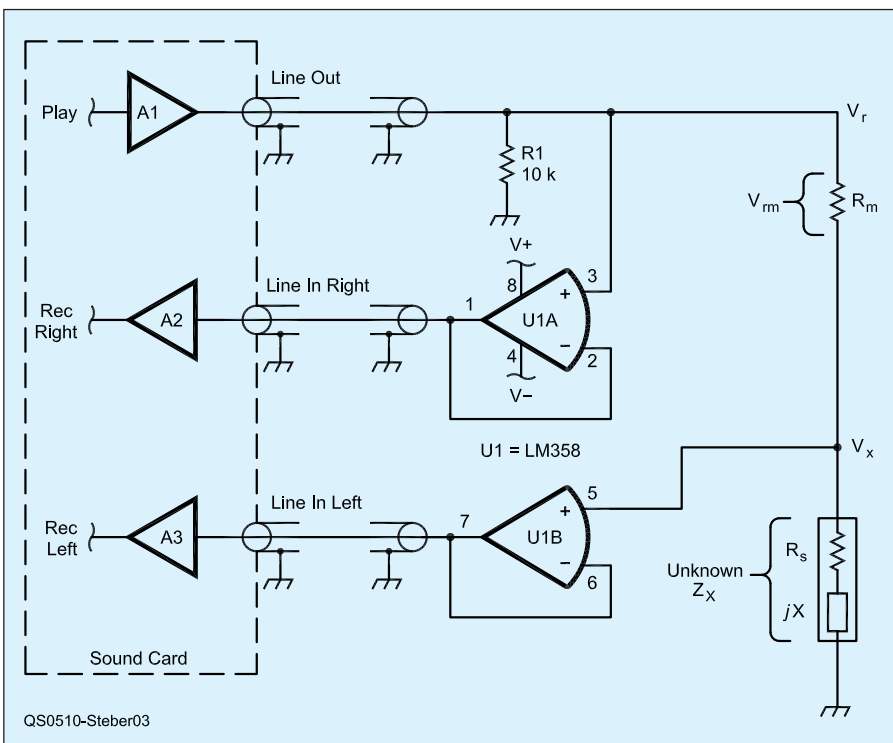
ring to Figure 3, we see that A1 is the line output amplifier, and A2 and A3 are the right and left channel line inputs of the sound card. These amplifiers can be a source of distortion if proper levels are not maintained. There are two culprits here. One is overdriving, and the other is saturation. If A1 sources too much current it will distort. I viewed V_r on my Tektronix TDS360 Digital Real Time Oscilloscope with FFT and saw a lot of second and third harmonic distortion when V_r exceeded 820 mV with R_m = 10 Ω and Z = 0. Since the full output level is 1.62 V, it needs to be attenuated. This can be done either with the sound card

mixer or with the signal generator level control in my program. I choose to set the PLAY level to maximum in the mixer and set the level to 0.5 in the program, as it is easier to remember.

The other consideration is that the line input amps A1 and A2 will saturate if the input voltage is too high. This is regardless of the RECORD setting in the mixer. On the SB card this occurs at 820 mV. Since we are using gains of unity in the circuit, we can prevent this by adjusting V_r as noted above. So, for my SB card, I just set the PLAY (output) level to 0.82 V and both conditions are satisfied. Just in case, I have provided a real time oscillo-
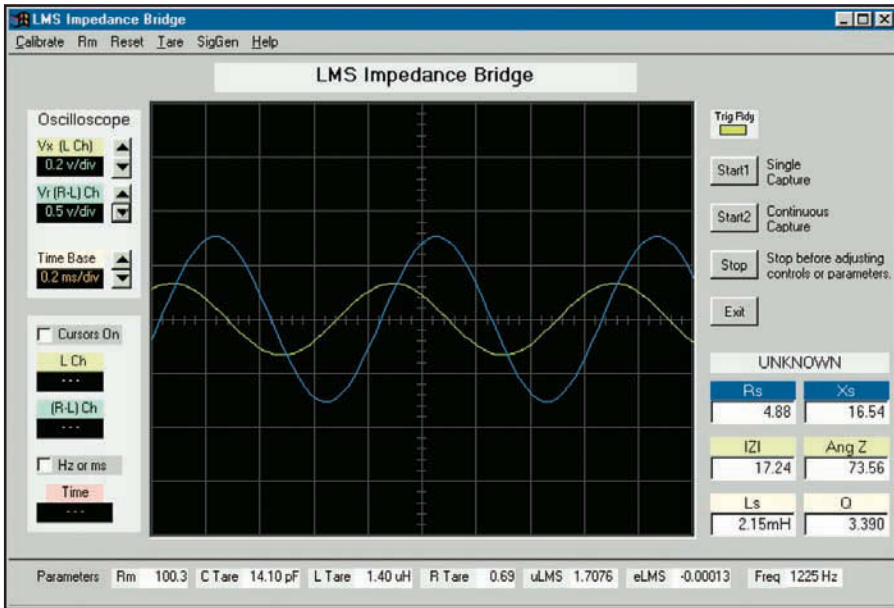
Figure 4—The main window of the LMS impedance bridge program.

depend on it. Strive for at least 1 percent here and do not use an inductive resistor—carbon or film types are fine. Although the bridge has a wide range, it is best to keep the levels of V_x and V_{rm} reasonable. The software 'scope helps monitor these levels. To reap the maximum benefit from the bridge, R_m should be selected for the approximate range of impedance. For example, if $R_m = 10\ \Omega$, that is the approximate impedance to measure. This can be done by trial and error or by making educated guesses, just as with most bridges. In general, you should be able to measure over a 0.01-100 range, based on R_m.

Use Figure 1 to estimate the expected impedance. Table 1 illustrates the range you can expect for a given R_m at 1225 Hz (the bridge frequency will affect this range). For example, from the table, if $R_m = 10\ \Omega$, you can measure L between 12.9 µH and 12.9 mH and C between 1299 µF and 0.129 µF, at 1225 Hz. Provisions are made in the software to store several values of R_m for the program to use. Just make sure that is the value that is actually in the circuit.

At the high and low ends of the bridge, stray capacitance and inductance start to play a role. These values can be compensated out by using the bridge to measure them—the unknown impedance being an open-circuit and short-circuit, respectively. These values, called "tare," are then automatically used to compensate the result. For example, in my circuit, there is about 14.1 pF of stray capacitance with Z open and $R_m = 100$ kΩ. So, that value is what I enter for C TARE in the program. Obviously, this only makes a

difference when measuring small capacitors. Similar compensation may be made for the inductive wiring component when Z is near zero.

Once you have balanced the stereo channels and (optionally) calibrated the scope, you are ready to start measuring. All calibrations are saved so that you only need to do it once. The mixer that came with your sound card or the one that came with *Windows* needs to be checked, as you may have changed its settings. When you start the program, a notice appears 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 sound card.

In the mixer PLAY section, enable WAVE and SPKR; set the sliders to max, and mute all others (including LINE to avoid audio feedback). In the RECORD section, enable LINE; set it to max, and mute all others. Set the stereo balance to center for all controls. These settings are *important*. When I tested the program with *WindowsXP* I forgot to mute LINE in the PLAY section and got meaningless results (due to audio echoes) until I realized my error.

Using the Impedance Bridge

Measuring impedances is very easy with this bridge but it always helps to think about what you are doing so you don't misinterpret results. Don't try to measure a 100 pF capacitor with an R_m of 10 Ω. Assuming you have a "ballpark" R_m, just connect the unknown to the Z points of the circuit (shown in Figure 3) and click the

START button on the screen. The bridge will automatically determine if the reactance is capacitive or inductive at the measuring frequency. A number of items are then calculated and displayed, including the real and reactive parts of Z; the magnitude and angle of Z; the L or C value of the component; and its Q or D factor. The 'scope is handy for looking at the relative magnitude and phase of V_x and V_{rm} and to see if you have reasonable levels.

If you aren't too familiar with measuring impedance (and even if you are) you may run into some situations that are unusual or seem to give inconsistent values. As a guide, remember that impedance is usually defined for a given frequency only and may be different at other frequencies or signal levels. If you are measuring a resistor, you will find it has some reactance and it will show up in the box on the screen as either a capacitor or inductor. The *QEX* article gives some examples for measuring input impedance, transformer impedance, electrolytic capacitors, iron core inductors and even *negative resistance*.[5]

Final Thoughts

Numerous components were measured on a commercial LCR bridge and compared to the LMS bridge. The commercial bridge had an accuracy specification of between 1 and 5 percent—depending upon range, component type and frequency. Good agreement was achieved between the two—better than 1 percent in most cases. With any luck, you will achieve similar results. In any case, I hope this is a project you can't *resist*!

Notes
[1]G. Steber, "Tracing Current And Voltage," *Circuit Cellar*, Jan. 2004, pp 56-61.
[2]G. Steber, "LMS Impedance Bridge," *QEX*, Sep/Oct 2005. This version is available at **www.arrl.org/qex/2005/Steber.pdf**.
[3]**www.arrl.org/files/qst-binaries/steber1005.zip**.
[4]**www.microsoft.com/downloads/**.
[5]See Note 2.

Photos by the author.

George R. Steber, WB9LVI, is emeritus Professor of Electrical Engineering and Computer Science at the University of Wisconsin-Milwaukee. Dr Steber has been a corporate officer, a consultant, and a product designer, and has received 17 patents. George has an Advanced class license and is a life member of ARRL. He enjoys racquetball, reading, playing his Bach trumpet, astronomy and editing video; he recently restored a previously lost and badly damaged NBC "Tonight Show" program for the Kate Smith Commemorative Society. Dr Steber was awarded the QST *cover plaque in May 1975. You can reach him at* **steber@execpc.com** *with "LMS" in the subject line.*