

An LMS Impedance Bridge

*Come learn about LMS impedance measurement and
build a unique PC sound card impedance bridge.*

By Dr George R. Steber, WB9LVI

Quite some years ago, I built an impedance bridge using a relatively new (at that time) DSP microprocessor, the TMS32010. It was based on a technical paper¹ that used the LMS (least mean square) algorithm. Results of that project were mixed. It was helpful because it verified that the LMS algorithm could be used for real-time impedance measurements. But it was disheartening because it was not very accurate and could only operate at a maximum bridge frequency of 50 Hz. So I filed it away for future reference.

Over the years I've maintained an interest in impedance measurement, digital signal processing (DSP) and

programming. Recently I did a novel project² using a PC with a sound card and some DSP techniques to implement a low-cost curve tracer (I versus V) for devices like Zeners, LEDs and transistors. During that task I sometimes thought of the old LMS bridge. Could it be implemented on a PC? After a lot of study and some serious modifications, the answer is a resounding "yes." In fact, it turned out to be much more than I had hoped for, yielding a wide range, low-cost impedance measuring system. An abbreviated article describing construction and operation of the LMS bridge has been written for *QST*.³ Presented here are the technical details behind this unusual system, its operation and some additional practical applications. In case you don't have the *QST* article handy, I will also present some material on installation and operation of the bridge.

Nearly any PC can be used in this project, as there is no need to modify

in any way. You can use one of the newer 3 GHz PCs or dust off that old 200 MHz PC 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. Of course I am not giving out guarantees that this project will work with your system, but 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* based impedance measuring system. 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, capaci-

¹Notes appear on page 47.

tors, 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 you will need a digital voltmeter (DVM) for calibration purposes, although even this is not absolutely required.

As usual, I am getting ahead of the story. As a professor, now retired, I am obliged to present more background and theory on this subject. I think you will find it interesting, so please try to resist the urge to skip to the end of the article.

Impedance

Impedance is basically the opposition to current flow. It is a more general form than resistance alone. Impedance can have a resistive part and a reactive part. For resistors, the reactive part is very small, unless they're wire wound. Inductors and capacitors have both resistive and reactive parts. The resistive part is often 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).

$$Z = |Z| \angle \theta = R + jX \quad (\text{Eq 1})$$

where Z is impedance in ohms, $|Z|$ is the magnitude of Z , θ is the angle of Z , R is the real (or resistive) part of Z ,

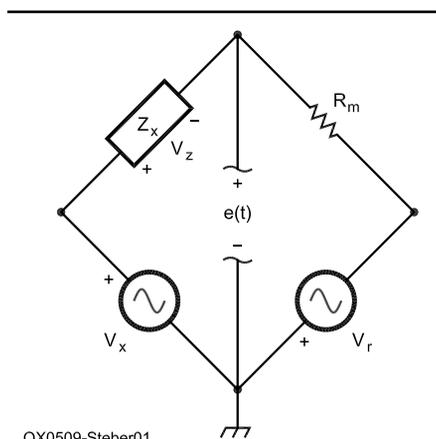


Fig 1—Old LMS bridge. Z_x is the unknown and R_m is reference resistor.

and jX is the imaginary (or reactive) part of Z . The two forms in Eq 1 are related by:

$$|Z| = \sqrt{R^2 + X^2} \quad \text{and} \quad \theta = \tan^{-1} \left(\frac{X}{R} \right) \quad (\text{Eq 2})$$

As noted above, impedance can also be written in other forms but in this article we will always model the unknown impedance as a series combination of resistance R and reactance X .

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.

Learning From The Old LMS Bridge

Since many of the ideas for the current project were derived after looking at the problems of the original LMS bridge project, we will look at it first. It is shown in Fig 1. The signals are all sampled signals, but we will not denote that at this time for sake of clarity. In this bridge, V_r and V_x are two sinusoidal voltage sources with the same radian frequency ω_0 , but with different amplitudes and phase shifts. Reference voltage source V_r is of con-

stant amplitude A and zero phase shift. However, V_x has a variable amplitude and phase shift. They can be written as follows:

$$\begin{aligned} V_r &= A \sin(\omega_0 t) \\ V_x &= B \sin(\omega_0 t + \phi) \end{aligned} \quad (\text{Eq 3})$$

The parameters B and ϕ are controlled to balance the bridge. V_r and V_x are generated via D/A (digital to analog) converters from the microprocessor. Voltage $e(t)$ is read into the microprocessor with an A/D (analog to digital) converter. Other elements of the bridge are the unknown impedance Z_x and the reference resistance R_m . When the bridge is balanced (voltage $e = 0$) the unknown impedance is given by

$$Z_x = R_m \frac{B}{A} \angle \phi \quad \text{at the frequency } \omega_0 \quad (\text{Eq 4})$$

Expressing V_x in terms of in-phase and quadrature components yields:

$$\begin{aligned} V_x &= B \cos \phi \sin(\omega_0 t) + B \sin \phi \cos(\omega_0 t) \\ (\omega_0 t) &= W_1 A \sin(\omega_0 t) + W_2 A \cos(\omega_0 t) \end{aligned} \quad (\text{Eq 5})$$

where $W_1 = (B/A) \cos \phi$ and $W_2 = (B/A) \sin \phi$ are the weights of the in-phase and quadrature components, respectively. With B and ϕ expressed in terms

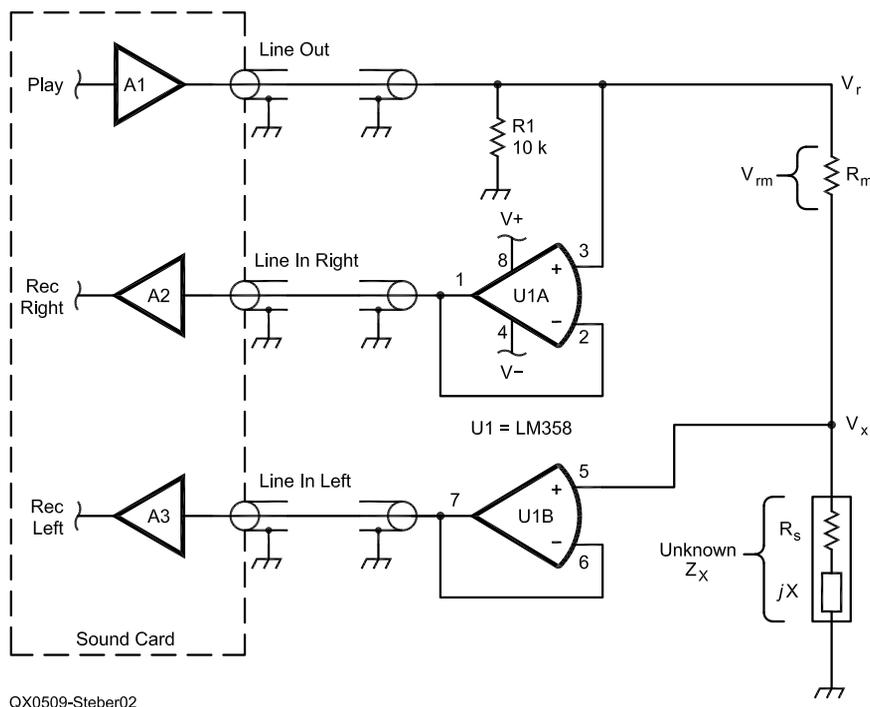


Fig 2—Impedance measuring circuit and interface to sound card. See text for more information.

of W_1 and W_2 , Eq 4 can be written as:

$$Z_x = RmW_1 + jRmW_2 \text{ at balance} \quad (\text{Eq 6})$$

The terms RmW_1 and RmW_2 are the real and imaginary parts of Z_x . To balance the bridge, one starts with initial values of W_1 and W_2 and iteratively modifies these to force $e(t)$ to zero. One method of doing this, the LMS algorithm, requires that the error be found at each new sample and updated values of W_1 and W_2 be calculated that hopefully force $e(t)$ to zero over time. The LMS algorithm does that well but we won't go into how it does it right now.

First we observe a few things about this bridge. On the *plus* side we see from Equation 6 that 'at balance,' the real and imaginary parts of Z_x depend only on the weights and Rm (reference resistor) and there is no requirement to know the amplitude A . That is a nice result. On the *minus* side, since the LMS algorithm requires that calculations occur at each new sample of $e(t)$, we have only one sample period to find new values of W_1 and W_2 . In addition, the sampling of $e(t)$ and the outputs V_r and V_x must be synchronized. Note too that since the sources V_r and V_x are generated digitally they must be calculated for each new point on the sine waves. Finally, we see that the unknown, Z_x is floating above ground, which is not desirable.

Consider now a typical sound card in a PC. It has very good 16 bit A/D and D/A converters. We can easily generate the signals V_r and V_x and output them to the bridge via the sound card line outputs. Similarly, the error signal $e(t)$ can be read with a line input. But how do we keep the inputs and outputs synchronized within one sample period? If you are familiar with *Windows* you probably know that all sound card I/O is done via buffers. And *Windows* decides when to empty and fill them. Because the input and output buffers may be long and not synchronized this argues against using the LMS algorithm, which needs to make sample to sample decisions. I wrestled with this problem, but could make no headway until I started looking at different circuit topologies.

Impedance Measuring Circuit

Consider the circuit in Fig 2. Instead of using two sound card outputs and one input, as in the old LMS bridge, it uses one output and two inputs. By intentional design, the two inputs are synchronized with each other but not necessarily to the output from the sound card. Resistor $R1$

is there to provide a ground reference for the sound card output. Rm is the reference resistor and Z_x is the unknown, as in Equation 1. We see that the unknown impedance is now grounded. Two op-amps U1A and U1B provide isolation and buffering of the bridge 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. Shielded audio cables are suggested for connecting to the sound card.

U1 is a cheap LM358 dual op-amp or equivalent which can be powered from bipolar power supplies of 3 volts to 15 V. It is best to keep the power supply voltages low to protect the sound card line inputs in case of problems. My circuit runs at about plus and minus 3 V. A bipolar battery supply could also be made using four AAA batteries with a center tap connected to ground.

Since all measurements depend on Rm , it is critical that we know its resistance precisely. To effectuate different ranges of the instrument, we may also wish to suitably change the value of Rm . We will discuss more about this later on.

There are many ways to measure impedance with the circuit of Fig 2. All of them require that a sinusoidal signal V_r be applied to the circuit and a number of sequential samples of V_r and V_x be captured in buffers. Discussed below are several methods of doing this.

Three Measurement Method

An old method of calculating impedance, sometimes called the three-voltmeter method, can be used. Here is how it works. Measure and record the three voltages V_r , V_x and V_{rm} as shown on Fig 2. We see that

$$Z = \frac{V_x}{V_{rm}} Rm = Z \angle \theta \quad (\text{Eq 7})$$

To find the magnitude of Z , simply take the magnitude of V_x divided by magnitude of V_{rm} and multiply by Rm as shown below.

$$|Z| = \frac{|V_x|}{|V_{rm}|} Rm \quad (\text{Eq 8})$$

It is a little more complicated to show, but considering the three voltages as vectors in a triangle and ap-

plying the law of cosines results in the equation for the phase angle θ as shown below.

$$\theta = 180 - \cos^{-1} \left[\frac{V_r^2 - V_{rm}^2 - V_x^2}{2|V_{rm}||V_x|} \right] \left[\frac{180}{\pi} \right] \quad (\text{Eq 9})$$

in degrees

There are other ways to calculate θ that involve multiplying the voltages and filtering, but they don't provide more benefits. A drawback to this way of finding θ is that it cannot distinguish between positive and negative angles of reactance. However, the correct θ can be found by looking at the zero crossings of $V_{rm} = A \sin(\omega t)$ and $V_x = B \sin(\omega t + \theta)$ since at $t=0$ (and multiples of the period), $V_x = B \sin \theta$ which is > 0 for inductances and < 0 for capacitances.

This method was implemented on a PC with good results over much of the measuring range. If you looked no further, this would be an acceptable method of finding impedance. When either V_x or V_{rm} is small, however, there is a substantial error in θ . This was attributed to noise and other errors from the measurement of these voltages. More sophisticated methods can reduce these errors.

Least Squares Method

The literature is full of different error-minimization criteria but the most widely used one is the least square approximation originated by Gauss. Simply stated, the least squares principle involves selecting the function that minimizes the sum of the squared errors. A more complete discussion can be found in Reference 4. It is applicable to both continuous and discrete systems. Since we will be dealing with discrete variables for the rest of this discussion, it is appropriate to introduce them now. Fig 3 shows a adaptive linear combiner we will use to illustrate the least squares method. It consists of unit-time delays, weights and summation blocks. There are two inputs: the data samples x_k and the desired response d_k . Sample delays are represented by z^{-1} with the samples taken at points $k, k-1, \dots, k-L$, going back in time through the data samples.

The L-input samples, $x_k, x_{k-1}, \dots, x_{k-L}$ may be represented as a vector.

$$X_k = [x_k \ x_{k-1} \ \dots \ x_{k-L}]^T \quad (\text{Eq 10})$$

where T stands for transpose. So X_k is actually a column vector and the subscript k is used as a time index. Similarly, we define a weight vector

$$W_k = [w_{0k} \ w_{1k} \ \dots \ w_{Lk}]^T \quad (\text{Eq 11})$$

From Fig 3, we see that the output, y_k , is a linear combination of the input samples and the weights. The error signal with time index k is the difference between the desired response d_k and the output y_k and is given by:

$$e_k = d_k - y_k = d_k - W^T X_k \text{ (dropping the subscript of } W \text{ for clarity)} \quad (\text{Eq 12})$$

and

$$e_k^2 = d_k^2 + W^T X_k X_k^T W - 2d_k X_k^T W \quad (\text{Eq 13})$$

This is the instantaneous squared error and is the function we wish to minimize by adjustment of the weights, W . This rather imposing task can be attacked via two main methods, the *non-recursive* approach called the Wiener-Hopf method or the *recursive* approach of the Widrow-Hopf (LMS) method. Solutions presented in Note 4 will be used.

A closed-form solution for the Wiener-Hopf method can be written as:

$$W^* = R^{-1}P \quad (\text{Eq 14})$$

where $R = E[X_k X_k^T]$, $P = E[d_k X_k]$, E denotes taking the expected value, and W^* is the optimal weight vector. The procedure is straightforward. We capture L samples, calculate the expected values for R and P and simply calculate W^* . Surprisingly, in practice, it works quite well.

The LMS method proceeds by first calculating the error as in Eq 12 and then applying a steepest-descent algorithm to find W_{k+1} as:

$$W_{k+1} = W_k + 2\mu \epsilon_k X_k \quad (\text{Eq 15})$$

where μ is a constant that controls the speed and stability of adaptation. This deceptively simple recursive equation obtains the same solution as Wiener-Hopf, when converged.

Adaptive Impedance Bridge

Refer again to Fig 2 and notice that

$$V_x = \left(\frac{V_r - V_x}{R_m} \right) Z = \frac{V_{rm}}{R_m} Z \quad (\text{Eq 16})$$

where $V_{rm} = V_r - V_x$. Now, let $V_{rm} = A \sin(\omega_0 t)$ and $V_x = B \sin(\omega_0 t + \theta)$. Following the same procedure as for the old LMS bridge, V_x can be written as:

$$V_x = W_1 A \sin(\omega_0 t) + W_2 A \cos(\omega_0 t) \quad (\text{Eq 17})$$

To find the weights, we apply the methods described in the previous section. Fig 4 illustrates the adaptive lin-

ear combiner for this case where V_{rm} is the input, V_x is the desired signal and e_k is the error. The unknown impedance is given as:

$$Z = R_m W_1 + jR_m W_2 \quad (\text{Eq 18})$$

Both the Wiener-Hopf and LMS methods were implemented and simulated for comparison of accuracy and convergence using real data captured from a sound card. Many different cases of unknown impedances, whose values were precisely measured on a commercial LCR bridge, were tested. A sound card sampling frequency of 44,100 samples per second and a sinusoidal frequency of 1225 Hz was used for the test signal. Each channel was set to a capture length of 11,025 samples, which provided 0.25 seconds of data. As expected, both methods tended toward the same, and correct, solution for Z .

The Wiener-Hopf method provided the best (lowest) error for this number of samples in most cases. The LMS method varied in error depending on the amplitude of the signals and the value of μ . If nothing further were con-

sidered, Wiener-Hopf would be the method of choice. However, further experimentation showed that if the sampled data in the capture buffer was re-iterated several times, the LMS method improved dramatically. This, in effect, increases the number of samples. Since the LMS algorithm is so efficient, only a few milliseconds is added to the calculation. A similar re-iteration for the Wiener-Hopf method does not provide additional improvement and would require a larger capture buffer. Finally, the LMS algorithm was normalized to improve the speed of convergence over a wide range of signal levels.

Both methods are very good and greatly surpass the three-measurement method described earlier. Since I wanted to keep the capture length small (in order to have about 4 measurements per second) I chose to implement the LMS method.

Sound Card Considerations

A low-distortion, low-noise full-duplex sound card is desirable. The Sound Blaster Live fills the bill nicely

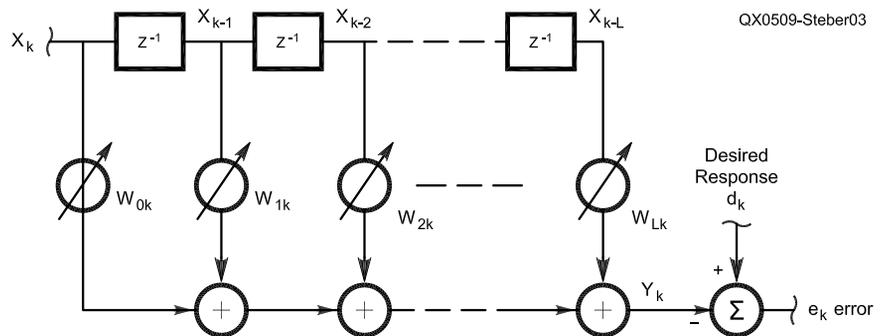


Fig 3—An adaptive linear combiner.

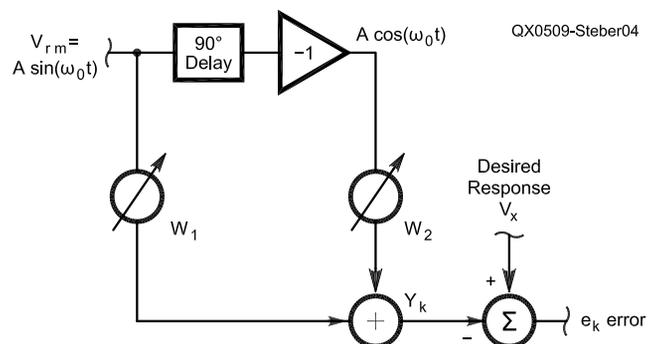


Fig 4—An adaptive combiner for the LMS bridge.

and probably many others will too. Since I cannot test them all, I will restrict my attention to this one. Referring to Fig 2, 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 the proper levels are not maintained.

There are two culprits here: One is excessive drive and the other is saturation. If A1 sources too much current, it will distort. I viewed V_r on my Tektronix TDS 360 (Digital Real Time Oscilloscope with FFT) and saw a lot of second- and third-harmonic distortion when V_r exceeded 820 mV with $R_m = 10 \Omega$ and $Z = 0$ (a short circuit). 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 chose 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. One of the nice things about the LMS bridge is that you don't need to know the amplitude of V_r .

The other consideration is that the line input amps A1 and A2 will saturate if the input voltage is too high. This is so regardless of the *Record* setting in the mixer. On the SB this occurs at 820 mV. Since we are using gains of "one" in the circuit, we can prevent this by adjusting V_r as noted above. So, for my SB Live, I just set the output level to 0.82 V and both conditions are satisfied. Just in case, I have provided a real-time oscilloscope function in the bridge display, so the sine waves can be monitored for possible flat topping. If a digital voltmeter is handy, V_x can be measured and used to calibrate the scope. This is provided in the program, but it is not required and does not affect operation of the bridge.

One other consideration is the balance of the two input channels. Since we need to calculate $V_r - V_x$ precisely, the two channels must be balanced. This function is also provided in the calibration section of the program.

A note is in order about earlier Sound Blaster sound cards such as the SB16 and AWE 32, since there are so many of these still in service. Unfortunately, they do not provide true, full-duplex operation. The same may be said of SB compatible cards, so be wary. For example, (with the latest drivers) the SB AWE32 can simultaneously play only unsigned 8 bits and record signed 16 bits. It also has a built-in amplifier that may over-

drive the LMS circuit. After some extensive tweaking, I managed to get one working with this program, but the results were not as good. I advise you not to use any of these cards.

Impedance Bridge Installation and Operation

The LMS bridge software is available on the ARRLWeb site and is zipped for fast downloading. Unzip it to a new folder and you are ready to go. Just run the EXE program. It was tested with *Win98* and *XP*. Fig 5 shows a screen shot of the bridge; there is a lot of information on the screen. The monitoring scope with its controls is on the left side. The most important part is in the lower-right corner labeled UNKNOWN, where all of the relevant data about 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 depend on it. Strive for 1% accuracy here (or better) 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 of the bridge, R_m should be selected for the approximate range of impedance. For example, if R_m is 10Ω , that is the approximate impedance to measure. This value can be chosen 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 to 100 range based on R_m . The chart in Table 1 illustrates the range you can expect for a given R_m at 1225 Hz. (The bridge frequency will affect this range.) From the chart, if $R_m = 10 \Omega$, you can measure L between 12 μH and 129 mH, and C between 1299 μF and 0.129 μF at 1225 Hz. The software has provisions to store several values of R_m . Just make sure that is the value *actually* in the circuit.

At the high and low ends of the bridge's range, stray capacitance and inductance start to play a role. These values can be compensated out by us-

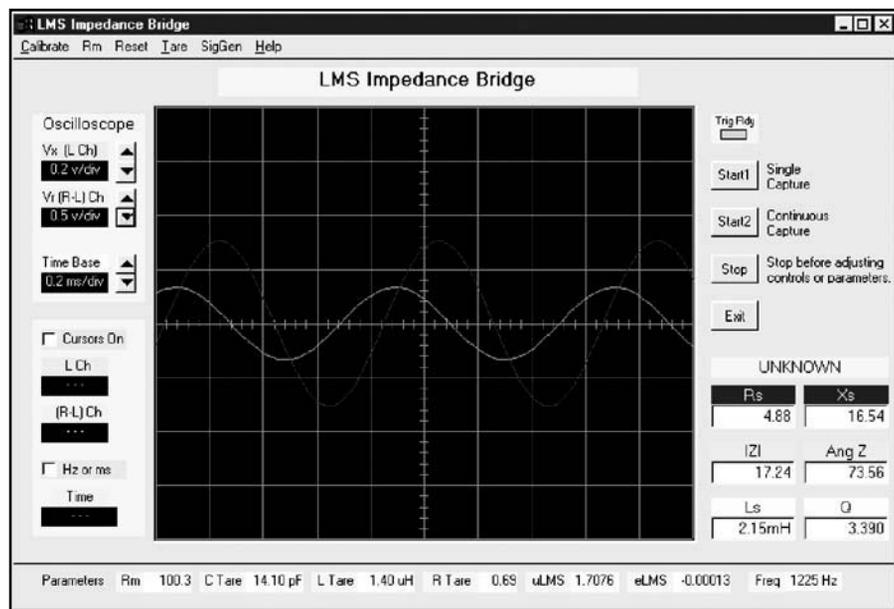


Fig 5—Main window of the LMS impedance bridge program.

Table 1—Range of Bridge for various R_m with a Bridge Frequency of 1225 Hz. (See text)

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

ing the bridge, itself, to measure them with 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 $R_m = 100\text{ k}\Omega$. 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, when Z is near zero.

Once you have balanced the stereo channels and (optionally) calibrated the scope, you are ready to start measuring. By the way, all calibrations are saved so that you only need to do it once. Next, the mixer that came with your software or the one that came with *Windows* needs to be checked, as you may have changed its settings. When you start the program, a little notice comes on the screen to remind you about this.

Basically, you want to set the output level, input gain and stereo balance. The details of how to do this 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 their maximums 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.

Using The Impedance Bridge

Measuring impedances is very easy with this bridge, but it's helpful 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 that you have a “ballpark” R_m , connect the unknown to the Z points of the circuit shown on Fig 2 and click the “Start” button on the screen. The bridge will automatically determine whether the reactance is capacitive or inductive at the measuring frequency. Several items are 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 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 the value of an impedance is usually defined only for a given frequency 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. Since you know it's a resistor, just look at the magnitude of Z or real part of Z .

You are not limited to just L and C components; the bridge can measure all kinds of impedances including input impedance, audio transformer impedance, speaker impedance, solenoid impedance and even negative resistance. Some of these topics are covered in the following sections.

Measuring Input Impedance

The input impedance of an amplifier or other circuit may be found by connecting it to the unknown Z terminals as shown in Fig 6. Take care not to overdrive the amplifier input being measured. This can be controlled by adjusting the output level in the mixer and choosing a suitable value for R_m . Note, too, that impedance may vary greatly with frequency, so try several bridge frequencies.

Here's an interesting special case: If you want to measure the input impedance of the left channel of your sound card line input, do the following. Bypass U1B and connect the lower end of R_m directly to that channel input. This makes the input impedance the unknown Z . On my SB, the input impedance Z_{in} measured 28.2 k $\angle -6.41^\circ$ at 1225 Hz (with $R_m = 1\text{ k}\Omega$).

Measuring Transformer or Speaker Impedance

The impedances of audio transformers or speakers can be found by simply connecting them to the bridge as shown in Fig 7. For speakers, use $R_m = 10\ \Omega$ to get started.

As an example, I measured the

reflected impedance of a small audio transformer (Mouser 42MC003, 1.2 k: 8 Ω). I connected the primary to the unknown Z terminals with an 8.2 Ω resistor connected to the secondary, as shown in Fig 7. The bridge read 1.24 k $\angle 13.78^\circ$ at 1225 Hz (with $R_m = 1\text{ k}\Omega$). The reflected impedance depends on the load connected to the secondary, so you can experiment with different loads to see the effect.

Measuring Large Electrolytic Capacitors

While I was working on this project, the power supply capacitor in my old oscilloscope went out. The replacement, a 1000 μF unit, was too great for my C meter. So, I put $R_m=10$ in the bridge and measured it easily. Be careful with these kinds of capacitors and make sure they are discharged before measuring them.

Large capacitors can often have leakage and internal resistance. It's interesting to see how the capacitance of an electrolytic capacitor changes with frequency. A junk box capacitor measured 15.62 μF at 120 Hz, and it read 14.35 μF at 1225 Hz. By the way, it was marked 22 μF and thus outside its minus 20% tolerance.

Measuring Iron-Core Inductors

Iron-core inductors can give some strange results. First measure a small *air* core inductor (150 μH , or so) and vary the output level of the line by varying the volume out in the *Windows* mixer. Note that the reading for inductance barely changes; only due to noise and harmonic distortion errors. Now measure a small *iron*-core inductor and do the same thing. You will notice that the inductance decreases with signal level. This unexpected result (at least for me) required some research. It turns out that

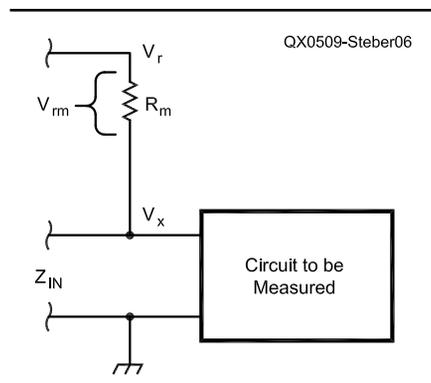


Fig 6—Setup for measuring circuit input impedance.

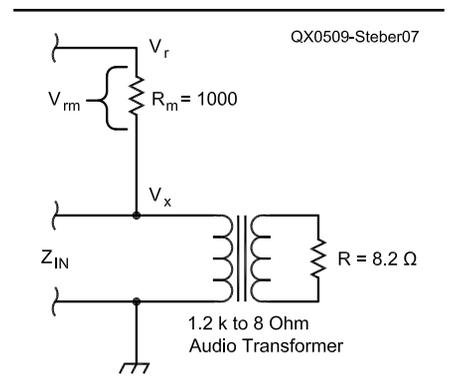


Fig 7— Setup for measuring transformer or speaker impedance.

the permeability of the iron starts out low for small currents and becomes higher and more constant (linear range) with increasing current. Since L is proportional to the permeability, there is a reduction in L at low currents. If you continue to increase the current beyond the linear range, the iron will saturate (I knew that!) and L will decrease. This last part is of concern to those who design switching regulators. Reaching such large currents is not within the capabilities of the bridge as it stands.

Another interesting thing about inductors is that they can have a resistive component that may vary with frequency. Using the transformer primary as shown above, with the secondary open, illustrates this. As the bridge frequency was varied from 525 Hz to 2205 Hz, the resistive part varied from 1.14 k Ω to 3.93 k Ω . Properly terminated with 8.2 Ω , it only varied from 1.06 k Ω to 1.26 k Ω over the same frequency range. Solenoids exhibit similar behavior and their impedance varies with plunger position.

Measuring Negative Impedance

It's possible to build a negative-resistance circuit. An article in *EDN Magazine* (Reference 5) about using negative resistance prompted me to do this experiment. Connect the negative impedance converter shown in Fig 8 as the unknown impedance Z . The op amp can be an LM358. Make sure R is less than R_m or else you will cancel it and cause oscillations. I used $R_m = 1000 \Omega$ and $R = 470 \Omega$. The bridge indicated minus 470 Ω . Be cautious with circuits of this type as they are likely to oscillate.

Final Thoughts

This program runs fine on Win98 and Win XP. It was written in Visual Basic 6.0. 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 already on many systems. If it is 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. The download takes about six minutes at 28.8 kbps.

If you just want to experiment with the 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

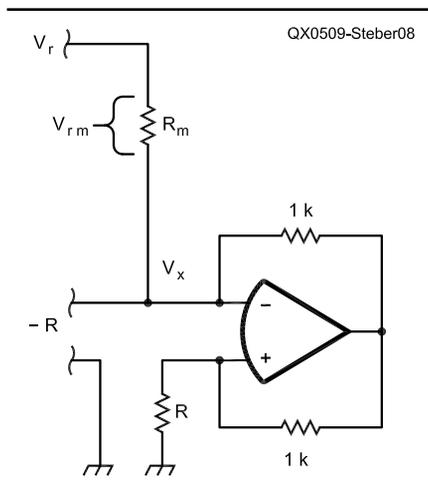


Fig 8—A negative resistance converter. Choose $R < R_m$ to avoid oscillations.

folder where it is located.

Numerous components were measured on a commercial LCR bridge and compared to the LMS bridge. Said LCR bridge was rated between 1% and 5% depending on range, component type and frequency. Good agreement was achieved between the two, better than 1% in many cases, a notable exception being small iron-core inductors (see comments above), which varied more. With any luck you will find similar results. In any case, don't be impeded in your quest for a fine LMS bridge. Perhaps this is one you can't resist.

George R. Steber PhD, is emeritus Professor of Electrical Engineering and Computer Science at the University of Wisconsin-Milwaukee. George, WB9LVI, is a life member of ARRL and was awarded the QST cover plaque in May 1975 for a ground breaking article on digital slow scan TV. Dr Steber has considerable industrial experience as a corporate officer, consultant and product designer, with 18 patents issued. In his spare time he enjoys racquetball, reading, playing his Bach trumpet, editing video and astronomy. He recently restored a previously lost, badly damaged NBC Tonight Show program for the Kate Smith Commemorative Society. You may reach him at steber@excepc.com with "LMS" in subject line.

Notes

- ¹M. Dutta, et al, "An Application Of The LMS Adaptive Algorithm For A Digital AC Bridge," *IEEE Transactions on Instruments and Measurements*, Vol IM-36, pp 894-897, (Dec 1987).
- ²G. Steber, "Tracing Current and Voltage," *Circuit Cellar Magazine*, Vol #162, pp 56-61, Jan 2004.
- ³G. Steber, "Low Cost Automatic Impedance Bridge," *QST* (Oct 2005).
- ⁴B. Widrow et al, *Adaptive Signal Processing* (Prentice-Hall ISBN 0-13-004029-0, 1985).
- ⁵E. Simons, "Negative Resistance Load Canceller Helps Drive Heavy Loads," *Electronic Design Magazine*, March 19, 2001. □□

New Science Book

Hard cover

Physics - Astronomy - Sciences

New Theories with Interpretations. Read about "The Death of Modern Gravity Theory", "Electricity, Flow of Electrons or Magnetism?", "Electromagnetic Pulses or Waves?", "Will an Object Launched into Space Ever Stop?", "Distance and Time - Are They the Same?", "Electromagnetic Pulse Speeds", and much more.

To order: Fax 972.874-0687 or send order to:

Walter H. Volkman W5OMJ
P.O. Box 271797
Flower Mound, TX 75027-1797
\$16.00 Postpaid USA
\$24.00 Postpaid Foreign Airmail

30 Day Money Back Guarantee

You must be satisfied with book or return postpaid for full refund of purchase price. No questions asked.