An Appendix to Filter Wizard #14 ("Match Point")

By Jorge Garcia, guest Filter Wizard columnist

Hello! Filter Fans, I'm sure you are all familiar with the classical approach to filter design. First you get your specifications, and then you go through a few simple calculations to determine the required order and response type for the filter (Butterworth, Chebyshev, Cauer, etc.). From there you choose a suitable prototype circuit (Sallen-Key, LC ladder, etc) with a 1 radian/second center frequency. Then you open a volume of charts and select the normalized component values. With the end in sight you scale the prototype circuit to the desired center frequency. It's very methodical and very tedious to design this way (though if you can afford fancy software, that can make things easier).

This was the only method I knew to design filters, so imagine my surprise when I read Kendall's article "An Excellent Fit, Sir" where he describes designing a filter using a spreadsheet. My first reaction was amazement. No sophisticated charts, no pole zero relationships, determination of a response type, none of those overly complex equations I learned in school.

His method seemed far more intuitive and practical. The ability to analyze circuits on a spreadsheet, to use Excel Solver to optimize circuit values, this was a technique that in my mind was essential for not only filter design, but as a powerful tool in my toolbox. So if you've been following along with his articles, besides answering many questions about filter design, I'm sure a few questions have popped up that need some small clarification and that's what I intend to do.

Throughout his various articles Kendall has posted some small "homework assignments". Being the inquisitive engineer that I am, I decided to explore those interesting assignments. This column is the result of those efforts. Specifically I'll be showing you the answer to the question he posed in the "Million Monkeys" article: "how could I modify the calculation to make use of all that gap between the upper and lower passband tolerance limits?" Specifically, how to allow the optimization to use gain values **greater than** the DC value, as well as **less than** it. Once that one is answered then the next question naturally follows: "Can the design still be optimized for minimum sensitivity?" I'll be walking you through the answers to those questions, so let's go!

It's straightforward to adjust the calculations so that the Solver can use the total of 0.2 dB ripple allotted in "Million Monkeys". The first method is to include a 0.1 dB "fudge factor" in such a way that the allowed ripple is between 0 dB and -0.2 dB referenced to the DC gain (instead of +0.1 dB and -0.1 dB) while keeping RL=RS. We still meet the condition of maximum power transfer at DC; there is no solution that will allow for a gain **above** the DC gain of -6.02 dB gain, so the solver ends up converging on a solution that pulls the pass band gain between -6.02 dB and -6.22 dB. So we still don't really use the allowed tolerance of 0.1 dB either side of the DC gain.

A smarter method is to change the load impedance to shift the maximum power point to the **top** of the desired ripple band, making it **different** from the DC gain. This is the method we're going to focus on here. Those who are used to designing filters using charts will recognize that when designing some responses, an adjustment factor is applied to the load impedance. In this column you'll find out why. The key to understanding all of this is in the diagrams in figure 1 below; I want you to take a good hard look at them, and engrave them in your memory. Got it? Good.

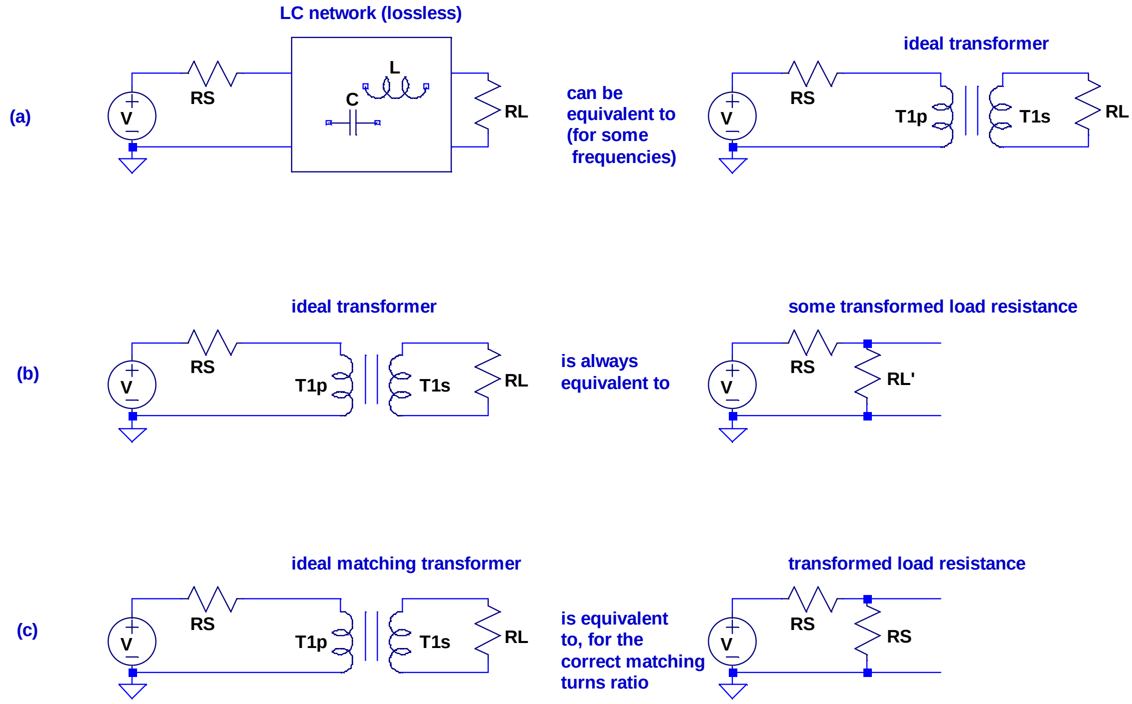

Figure 1 (a) to (c): An LC network can act as a transformer

The key to this whole discussion is to recognize that in some applications the LC network can behave as a transformer (at least at some frequencies). So we can use the transformer relationship to determine the effective turns ratio of the equivalent transformer that gives maximum power into RL, given RL and RS. We then set this turns ratio to provide just that extra gain we need to push the passband ripple to our upper limit, making this the 'maximum power' gain.

In Figure 1(b) I've replaced the LC Network with a transformer in our filter circuit. If we specify the transformer turns ratio correctly then we can obtain the maximum power transfer case (c) and everything to the right of the first RS can be replaced with a single resistor whose value is also RS.

I'll begin the derivation by first reviewing the basic transformer relations which will then be used to evaluate the maximum power voltage gain. A simple transformer is shown in figure 2.

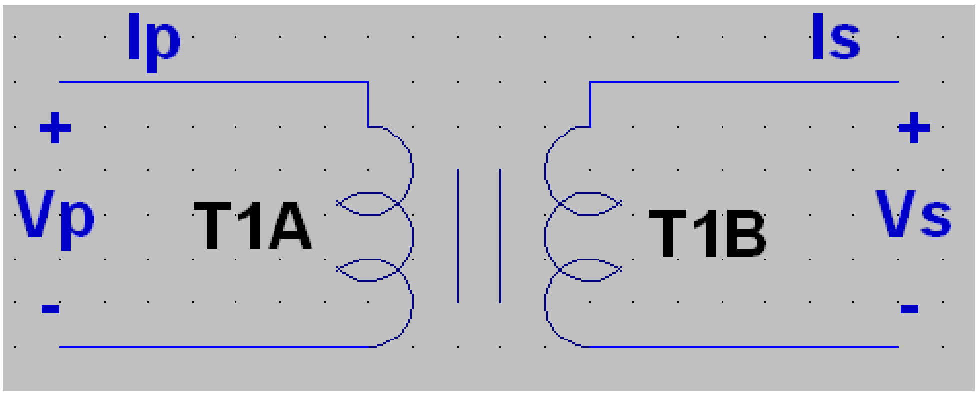

Figure 2. Transformer Parameters

We have, from the standard transformer relationships:

$$\frac{Vs}{Vp} = \frac{Ip}{Is} = n$$

and

$$RL = \frac{Vs}{Is}$$

$$RL' = \frac{Vp}{Ip}$$

so

$$\frac{RL}{RL'} = \frac{\frac{Vs}{Is}}{\frac{Vp}{Ip}} = \frac{Vs \cdot Ip}{Vp \cdot Is} = n^2$$

$$\underline{so\ n = \frac{Vs}{Vp} = \sqrt{\frac{RL}{RL'}}}$$

For maximum power transfer in figure 1(c) we have RL'=RS and therefore

$$\underline{n_{mp} = \sqrt{\frac{RL}{RS}}}$$

From the transformer derivation we can see that the voltage gain from the matching transformer is given by the square root of the ratio between the load resistance and the source resistance. With this information in tow we'll compare it with the DC gain. That's trivial since we know that at DC, inductors behave like shorts and capacitors like open circuits. The DC gain therefore is the voltage divider formed by RL and RS. By setting the 'maximum power' voltage gain equal to the DC gain times the ripple gain k (in V/V, not dB), we will be able to solve for the value(s) of

the ratio of load to source resistor, a=RL/RS, that will yield a minimum sensitivity design. The resulting expression is derived below.

Using a little algebra (yes, I said it, mwah hah hah) we arrive at a quadratic equation, as shown below:

$$G_{mp} = n_{mp} \cdot \frac{RS}{RS+RS} = 0.5 \cdot \sqrt{\frac{RL}{RS}} = 0.5 \cdot \sqrt{a}$$

$$G_{DC} = \frac{RL}{RL+RS}$$

$$so\ 0.5 \cdot \sqrt{a} = k \cdot \frac{RL}{RL+RS}$$

$$so\ a = 4 \cdot k^2 \cdot \frac{RL^2}{(RL+RS)^2} = 4 \cdot k^2 \cdot \frac{1}{\left(1+\frac{1}{a}\right)^2}$$

and after some manipulation, a quadratic:

$$\underline{a^2 + (2 - 4 \cdot k^2) \cdot a + 1 = 0}$$

Now let's play with this formula a little. Let's say we're designing a Butterworth (maximally-flat) response. What ratio of RL/RS will yield a minimum sensitivity design? Since we have no ripple, k=1, and using the quadratic formula we find that a=1 is the only solution.

So, if we are designing a maximally flat filter the impedance ratio for minimum sensitivity is 1 to 1 and this is perfectly inline with the conclusions stated in the "Match Point" article, just proven more rigorously. Now let's repeat this calculation for a maximum upper ripple of 0.1 dB, which translates to k=1.011579. The two values of a are now distinct: a=1.3566 and a=0.7371.

So, we have **two** impedance ratios that should be able to generate a minimum sensitivity design using the Excel Solver method (and look how different from unity they are even for this small ripple). At first glance we might be tempted to drop the second solution since we don't want to lose any more DC gain then necessary. However if you need to use the dual form of the filter then the lower value will be preferred since it will generate a larger output impedance hence preserving the dc gain of the dual form. See the "Dualling Master" and "Bruton Charisma" Filter Wizard articles for why this is useful.

Next we run the spreadsheet to get the values that will potentially yield minimum sensitivity designs with these load ratios. The passband response of a converged a=1.3566 solution is shown in figure 3. Note that it ripples both ways from the DC gain now.

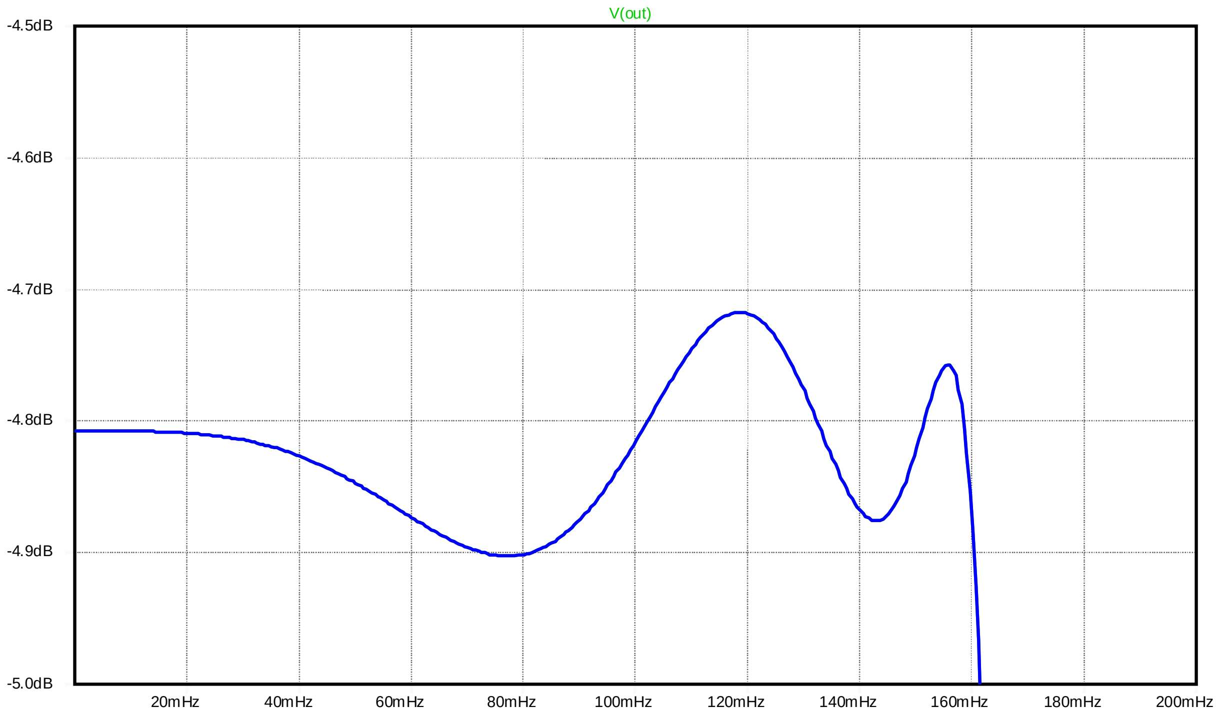

Figure 3: passband of a solution for a=1.3566

Let's look at some Monte Carlo simulations. I have centered the graphs for easy comparison and I assumed a 15% tolerance on the caps and inductors and 5% for the resistors since I run into these tolerances quite often. The basic schematic for the simulation is shown in figure 4; I have also included a response obtained from a non-minimum sensitivity design for comparison.

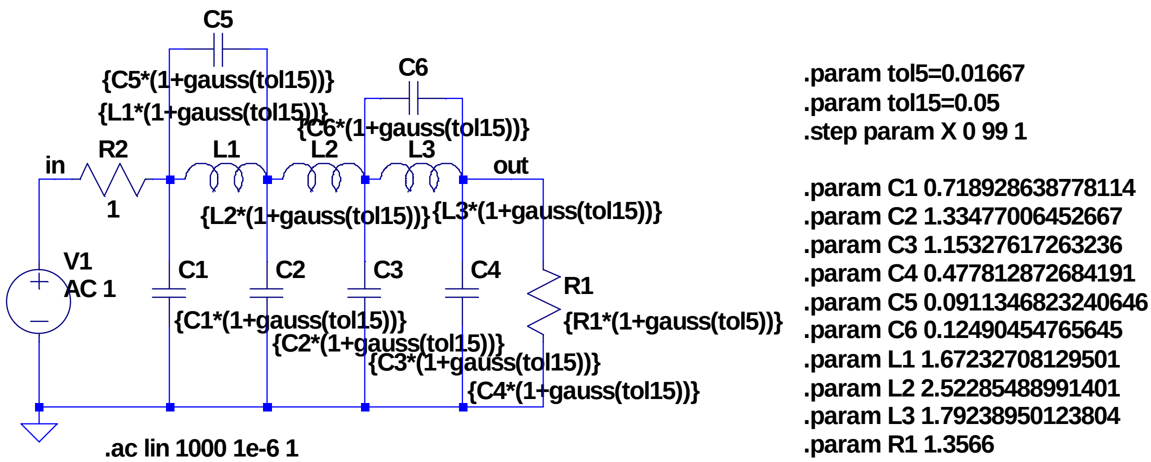

Figure 4. Monte Carlo Schematic

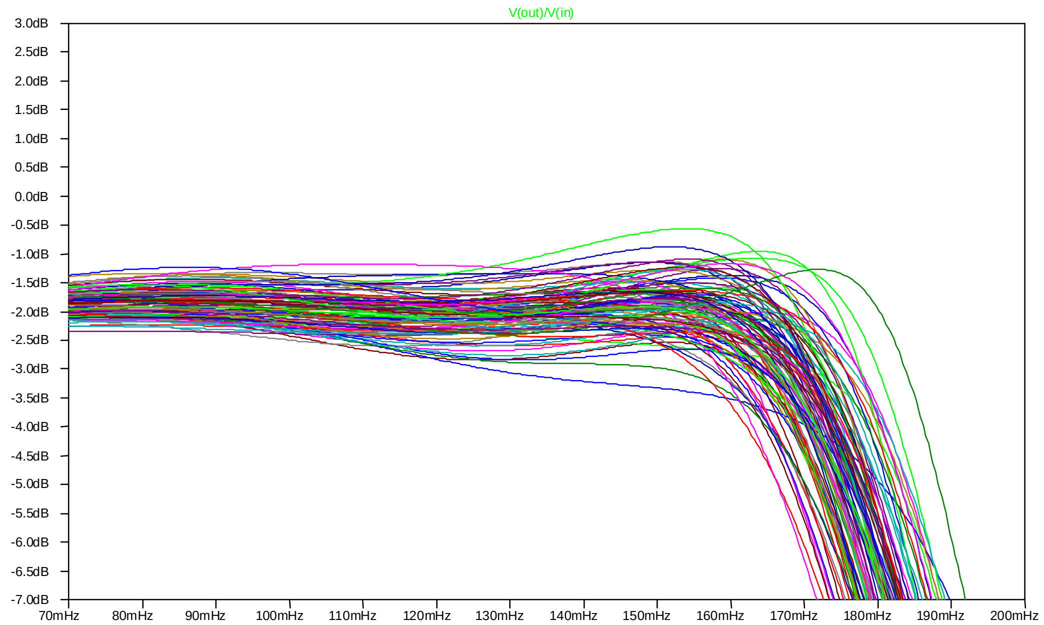

Figure 5. A Monte Carlo simulation for a sub optimal design

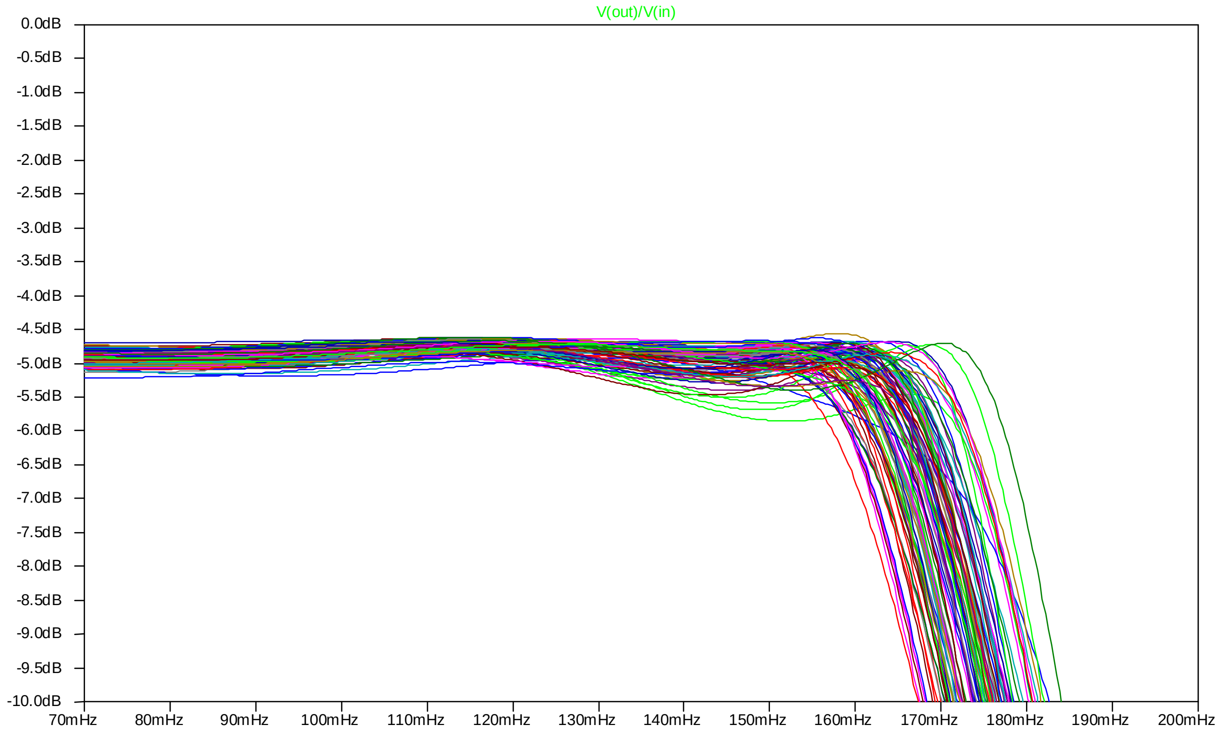

Figure 6. Monte Carlo for the a=1.3566 solution

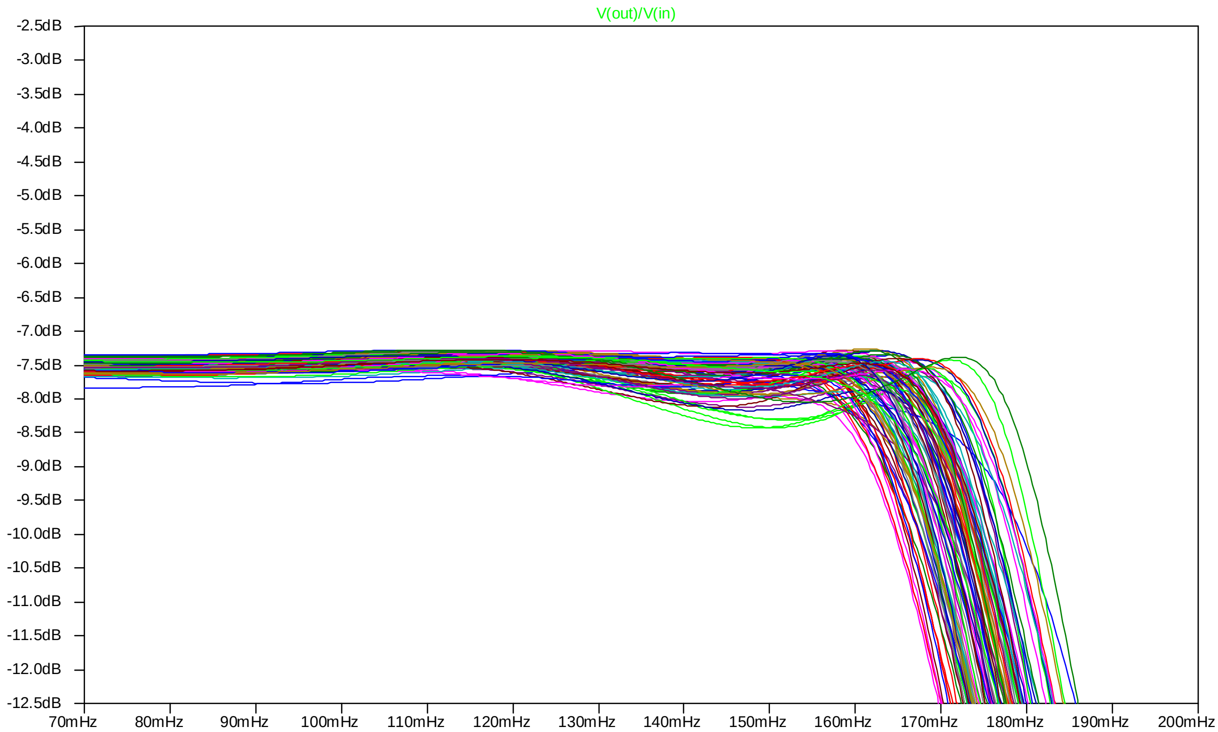

Figure 7. Monte Carlo for the a=0.7371 solution

As can be seen from figures 6 and 7 above, both of the impedance ratios we calculated yield low sensitivity solutions whose response doesn't exceed the top of the ripple band.

The Monte Carlo simulations assume that the tolerance distribution is Gaussian or Normal. Many real-world processes follow a Gaussian distribution; for an easy experiment grab a bag of resistors (at least 30, the more, the better) and measure the values. You will notice that many of them have a value that is much closer than the specified tolerance would lead you to believe. The Gaussian distribution has the following interesting properties:

-approx. 68% of the components will have a value within one standard deviation of the mean

-approx. 95% of the components will have a value within two standard deviations of the mean

-approx. 99.7% of the components will have a value within three standard deviations of the mean

What does all this mumbo-jumbo mean? Manufacturers specify tolerances to make sure none of their components exceed the specified tolerance, and they base the specification on 3 standard deviations from the mean, which is at the specified component value. So if a component has a specified tolerance of 15% then 68% of the components will have a tolerance of 5% or less. A similar trend is observed in the Monte Carlo sims where most of the runs seem to stay within a certain window while only a few yield extreme responses.

So what have we learned? We can use a simple quadratic equation to define the impedance ratios that will allow for minimum sensitivity designs that use the entire ripple around the DC gain, regardless of the specified ripple. I hope to have also shed some light on the subject of component tolerances and how they are specified.

Thank you for your time, and special thanks to Kendall for the opportunity and insight. If anyone has any comments or suggestions feel free to e-mail me at jorgegarcia08@comcast.net. Till next time (that is, if I'm ever allowed to do this again)!

About the author:

Jorge Garcia is a young engineer who enjoys all aspects of electrical engineering and design. He has a BSEE from Florida International University. His main interests include power electronics and embedded systems design. He currently resides in Miami, Florida with his wife Orlinda.