The Filter Wizard
issue 11: Simulate Circuits in a Spreadsheet with some 'Ladderal Thinking'
Kendall Castor-Perry

So, filter fans, I hope you tried out my recent spreadsheet-modified simulation schematic method ("Excel tunes up your schematic files"). It's a nice way of creating small circuit chunks with calculated component values, especially when the sums can't be carried out using the math capabilities of your simulator's preprocessor.

But say you now need to adjust component values to achieve some system goal, such as frequency response or predefined time behaviour, and there's no closed method for working out those values. You have to spend quite a bit of money to get proper circuit optimization features in a SPICE-compatible simulator; none of the free or affordable packages support it (QUCS looks intriguing, but its SPICE compatibility is very poor). And in today's busy world, it's hard to find the time to write bespoke analysis and optimization programs. I count myself lucky that I had to do lots of that in the past because it's such a good way of learning about practical circuit calculations.

In "An Excelent Fit, Sir!" we saw that the Solver tool in a spreadsheet can mould the coefficients of a factorized transfer function to fit a frequency response specification. That's useful if you can define the transfer function first, then implement it with a suitable topology. Now, if you're comfortable turning transfer functions into component values and interconnections, then – congratulations, you're a filter designer!

Let's assume, though, that this time you've got to optimize the response of an existing circuit without altering the topology, and that it isn't a simple cascade of second-order biquad blocks. Maybe it's like figure 1 – the bandpass filter from "Fainting in Coils":

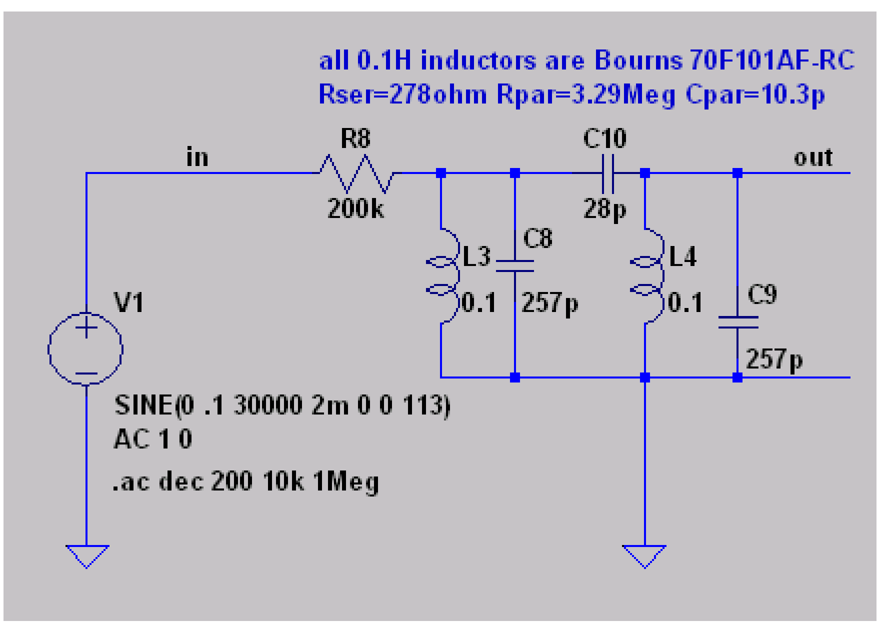


figure 1: the bandpass filter from "Fainting in Coils"

Now, you could analyze this by hand (homework! No, just kidding), or with a symbolic maths package. This gives you a bunch of fat expressions for the component values, involving the coefficients of the specific form of transfer function that the circuit realizes.

Then someone tells you that *this* component *here* has to be *that* value because it's got some approval, and, oh, of course *those* two inductors must stay equal, and at this point you might just throw up your hands in despair and wish that you could calculate the darned circuit's response right there on the spreadsheet.

Shazzam! Your wish is my command. Actually, the process of calculating the frequency response of a bunch of arbitrarily interconnected linear components is pretty straightforward. The bothersome part is the housekeeping – tracking what's connected to what, and how to map this efficiently into the 'nodal admittance matrix', a set of complex-coefficient equations that can be solved for any of the voltages and currents in the network. This is really all that SPICE is doing in an .AC analysis All the hard stuff lies in finding the circuit's operating point, or simulating the system in the time domain with all the ugly non-linearities.

We can make things *waaay* easier by scratching out that word 'arbitrarily', and restricting ourselves to the common 'ladder' form of circuit, shown in figure 2. This form doesn't require a full matrix solution, and can be solved with far simpler methods, which are described in circuit theory textbooks. We'll see shortly why we use impedance to quantify the series branches and admittance for the shunt branches (the ones that go to ground). Note that the left-most branch is always a series one and the right-most branch is always a shunt one:

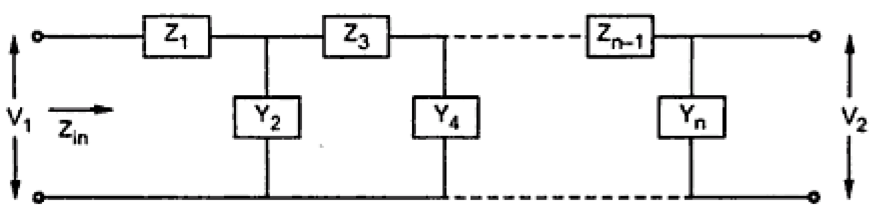


figure 2: the general form of a ladder network

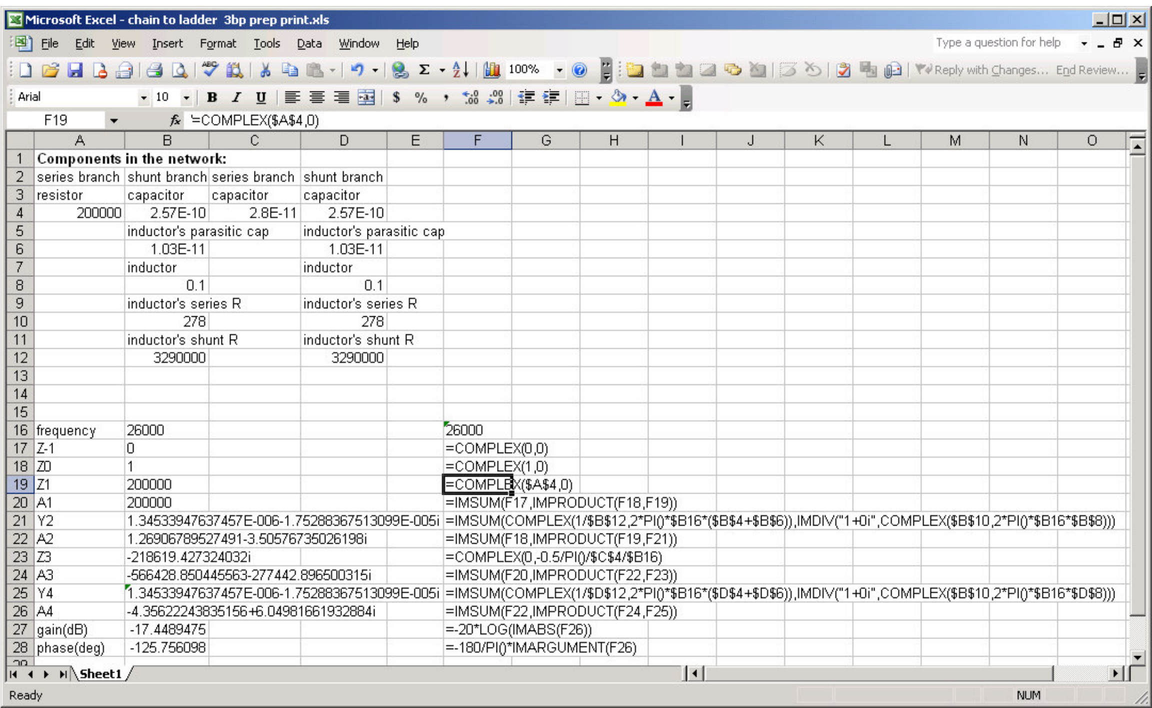
There are several ways of computing the frequency response of a ladder network. The most general involves breaking the ladder up into a cascade of interconnected 'two-ports'. Each two-port's voltage/current relationship is expressed as a 2x2 'chain matrix' (look it up), and the overall port properties of the entire network are determined by multiplying up all these matrices. This can be done in a spreadsheet – I've tried it – but it's more cumbersome than necessary for simple two-ports that are just passive components.

Much easier is this simple iterative procedure, which steps along the ladder and directly delivers the value of the attenuation A_n (the reciprocal of the gain) when you reach the end. The inputs to this algorithm are the complex *impedances* Z_i of the series branches and the *admittances* Y_i of the shunt branches. Starting with $\mathbf{A_{-1}=0}$ and $\mathbf{A_0=1}$, calculate $\mathbf{A_i=Z_iA_{i-1}+A_{i-2}}$ for i odd (series branch) or $\mathbf{A_i=Y_iA_{i-1}+A_{i-2}}$ for i even (shunt branch). After

the last iteration at Y_n, you can calculate the gain as **$-20\log_{10}(|A_n|)$** and the phase as **$-\arctan(im(A_n)/re(A_n))$** – and you're done!

The complex-number calculations can of course be done by manipulation of the real and imaginary parts using real arithmetic. But with a considerable saving in effort – though a definite reduction in readability – you can use the standard spreadsheet complex maths functions (not in Google Docs at present) such as IMSUM() to add complex numbers, and IMPRODUCT() to multiply. For pure passive components the impedance and admittance expressions are easy to write down. The impedance of an inductor is just COMPLEX(0,2*pi()*<frequency>*<value>), and so on. Writing formulae for impedances in series in the series arms, and admittances in parallel in the shunt arms, is also trivial.

Now in Figure 1, the inductor to ground has a series resistance (within the component, not visible on the schematic). So here, first write down an expression for the series impedance of L and R, and then take the reciprocal to get the admittance. Then add the admittances of the other parasitics and of the shunt capacitor. In figure 4 you can see the calculations, and the formulae that produced them, laid out at a single frequency (26kHz) for clarity (well, I *hope* it's clear!):


Microsoft Excel - chain to ladder 3bp prep print.xls

F19 =COMPLEX($A$4,0)

	A	B	C	D	E	F
1	**Components in the network:**					
2	series branch	shunt branch	series branch	shunt branch		
3	resistor	capacitor	capacitor	capacitor		
4	200000	2.57E-10	2.8E-11	2.57E-10		
5		inductor's parasitic cap		inductor's parasitic cap		
6		1.03E-11		1.03E-11		
7		inductor		inductor		
8		0.1		0.1		
9		inductor's series R		inductor's series R		
10		278		278		
11		inductor's shunt R		inductor's shunt R		
12		3290000		3290000		
13						
14						
15						
16	frequency	26000				26000
17	Z-1	0				=COMPLEX(0,0)
18	Z0	1				=COMPLEX(1,0)
19	Z1	200000				=COMPLEX($A$4,0)
20	A1	200000				=IMSUM(F17,IMPRODUCT(F18,F19))
21	Y2	1.34533947637457E-006-1.75288367513099E-005i				=IMSUM(COMPLEX(1/$B$12,2*PI()*$B16*($B$4+$B$6)),IMDIV("1+0i",COMPLEX($B$10,2*PI()*$B16*$B$8)))
22	A2	1.26906789527491-3.50576735026198i				=IMSUM(F18,IMPRODUCT(F19,F21))
23	Z3	-218619.427324032i				=COMPLEX(0,-0.5/PI()/$C$4/$B16)
24	A3	-566428.850445563-277442.896500315i				=IMSUM(F20,IMPRODUCT(F22,F23))
25	Y4	1.34533947637457E-006-1.75288367513099E-005i				=IMSUM(COMPLEX(1/$D$12,2*PI()*$B16*($D$4+$D$6)),IMDIV("1+0i",COMPLEX($B$10,2*PI()*$B16*$D$8)))
26	A4	-4.35622243835156+6.04981661932884i				=IMSUM(F22,IMPRODUCT(F24,F25))
27	gain(dB)	-17.4489475				=-20*LOG(IMABS(F26))
28	phase(deg)	-125.756098				=-180/PI()*IMARGUMENT(F26)

figure 4: all the calculations to give figure 1's response at 26kHz

Figure 5 shows the calculated amplitude response of the filter from 10kHz to 100kHz which (as it should do!) matches the response given in "Fainting in Coils":

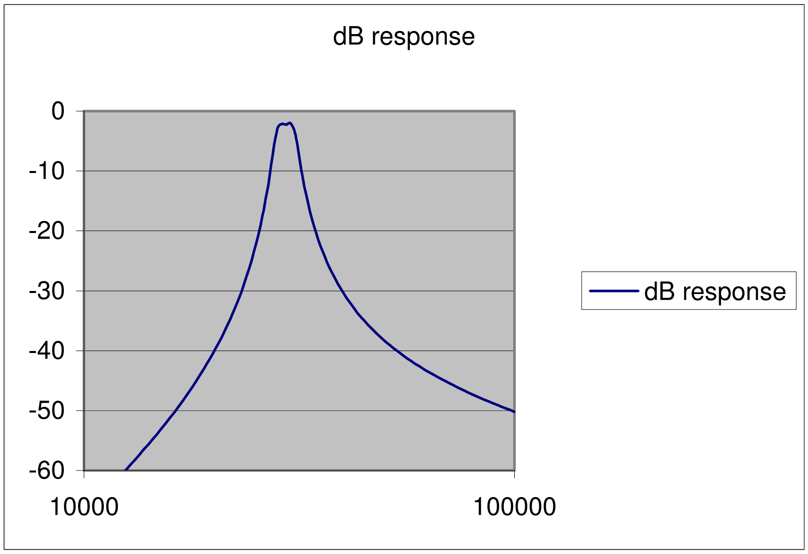


figure 5: amplitude response of figure 1's filter, calculated by Excel

So, there we have it: response computation of circuits using a spreadsheet. And you get to see the clockwork of the simulation process in front of you; it's much more hands-on, don't you think? And of course you already know how to create a complete LTSpice schematic from the modified circuit data.

In the next column, I'll show how combining this ladder analysis technique with the use of the Excel Solver gives a means for automatically tweaking circuit values to get a better fit to the response you want – in either the frequency or the time domain. It's even capable of creating complete filter designs from scratch, including unusual constraints. Take some time to look under the hood of *your* ladder filter circuits! best – Kendall