

New Program Directly Synthesizes Filters With Arbitrary Transmission Zero Placement

By Glenn Parker
Eagleware Corporation

Eagleware Corporation announces the release of its newest synthesis program, S/FILTER. This new program accepts design specifications from the user and directly synthesizes arbitrary schematic solutions.

'Look-up' design approach

Many electronic circuit synthesis programs today use a "look-up" approach to design. This technique involves fitting a preexisting schematic to the user's design criteria.

Over the years, hundreds of designs have been calculated, and these designs can be frequency- and impedance-scaled to fit many different scenarios. This technique usually works well — unless an engineer needs a slightly different topology than the existing ones, or when a finite transmission zero needs to be added to an existing schematic to notch-out an unwanted frequency. In the past, engineers would often take a preexisting design, add some customizing elements to it and launch a linear optimizer to recover the desired network response. This approach requires more of the engineer's time, and the engineer must often settle for a non-optimal response.

Direct synthesis

S/FILTER uses a technique called direct synthesis that does not suffer from these weaknesses. Contrary to the "look-up" approach described above, direct synthesis takes the user's design criteria, forms a purely mathematical representation — a transfer function — and extracts a schematic.

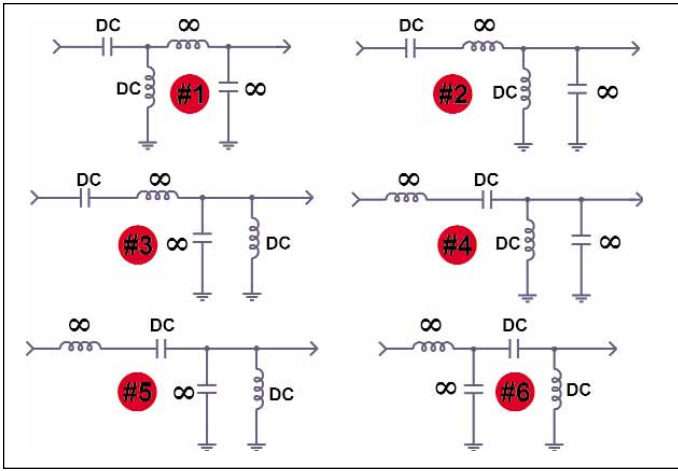
A filter's transfer function can be represented as a ratio of sums of products, or as a ratio of products of sums:



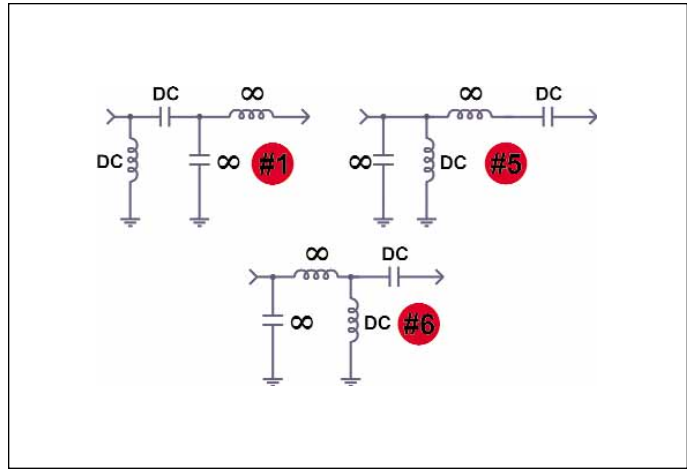
▲ Eagleware's new S/FILTER synthesis program directly synthesizes schematic solutions based on user input.

$$T(s) = \frac{\sum_{i=0}^m \alpha_i s^i}{\sum_{k=0}^n \beta_k s^k} = \frac{\prod_{i=1}^m (s - z_i)}{\prod_{k=1}^n (s - p_k)} = \frac{(s - z_1)(s - z_2)(s - z_3) \dots (s - z_m)}{(s - p_1)(s - p_2)(s - p_3) \dots (s - p_n)}$$

Using the products-of-sums representation, the equivalent filter network can be obtained by extracting one product at a time, where each



▲ Figure 1. Solutions for the Chebyshev bandpass example with series elements first.



▲ Figure 2. Solutions for the Chebyshev bandpass example with shunt elements first.

successive product represents some combination of circuit elements. S/FILTER adds functionality over existing direct synthesis programs by automating the element extraction process and by providing high level schematic customization tools and transform capabilities.

The transfer function can generally be realized by extracting pieces in any order, with each extraction sequence generating a different schematic. For example, consider a second order (4th degree) Chebyshev bandpass filter. There are two zeros at DC, two zeros at infinity, and no finite transmission zeros, giving a total of four zeros. The number of permutations for this filter is shown by the equation shown above right.

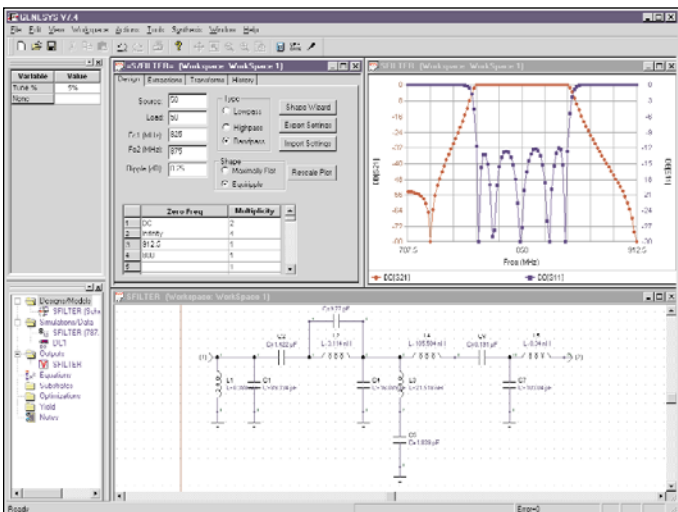
Thus, there are six permutations of the four transmission zeros:

$$\begin{aligned} \# \text{Permutations} &= \frac{(\text{Total } \# \text{ Zeros})!}{(\# \text{DC Zeros})!(\# \text{Infinite Zeros})!(\# \text{Finite Zeros})!} \\ &= \frac{4!}{2!2!0!} = \frac{24}{4} = 6 \end{aligned}$$

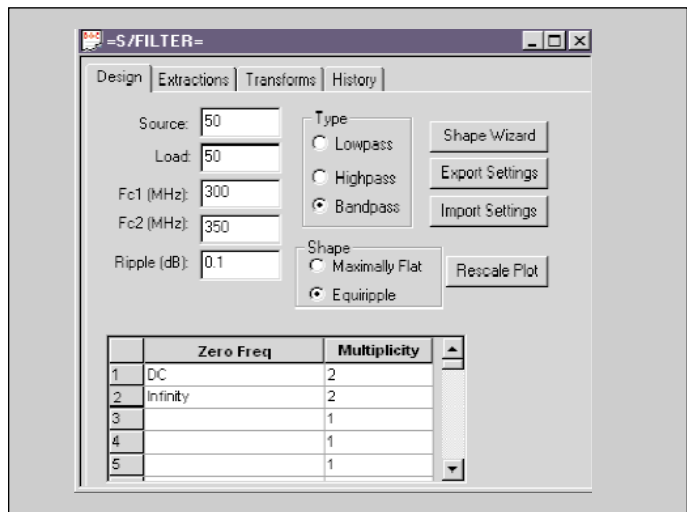
▲ Number of permutations for the example filter.

1. DC DC ∞ ∞
2. DC ∞ DC ∞
3. DC ∞ ∞ DC
4. ∞ DC DC ∞
5. ∞ DC ∞ DC
6. ∞ ∞ DC DC

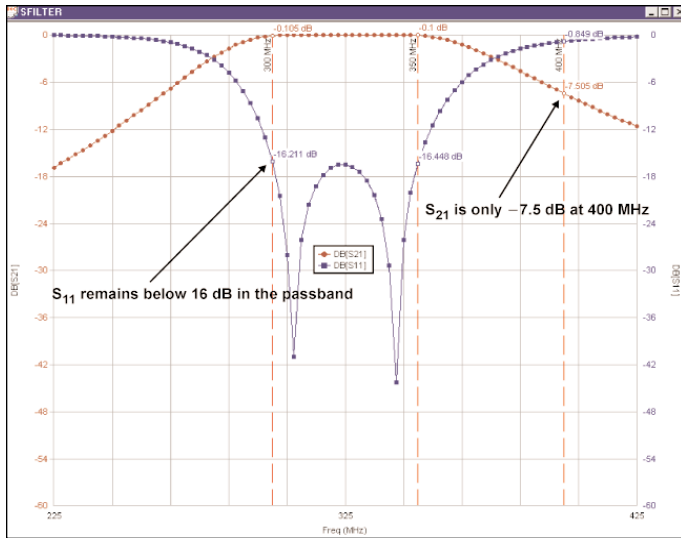
S/FILTER extracts a permutation in order from left to right (from the source to the load). The six permuta-



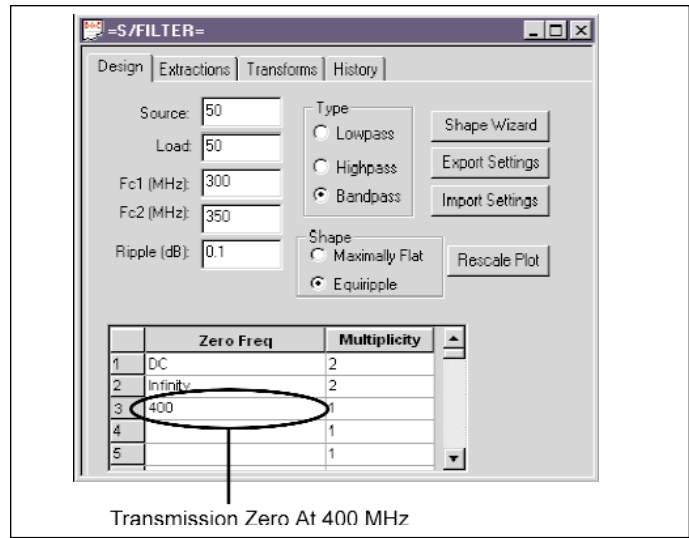
▲ Figure 3. S/FILTER design screen.



▲ Figure 4. S/FILTER design tab shown with initial design parameters.



▲ Figure 5. Response for the initial filter design.



▲ Figure 6. S/FILTER design tab showing the new 400 MHz transmission zero.

tions for the Chebyshev bandpass example correspond to the six schematics shown in Figure 1, when using the “series element first” option.

The schematics shown in Figures 1 and 2 have been labeled to indicate what type of transmission zero each element represents. For example, a series capacitor blocks DC and therefore represents a DC transmission zero. Notice that several of the schematics in Figure 1 are identical except for the resonator element ordering (#3 and #5). S/FILTER automatically detects this redundancy, and for this example reports that there are only three “unique” permutations. If the “Series Element First” option in S/FILTER is unchecked, another array of filters is generated. The unique solutions with a shunt element first are shown in Figure 2.

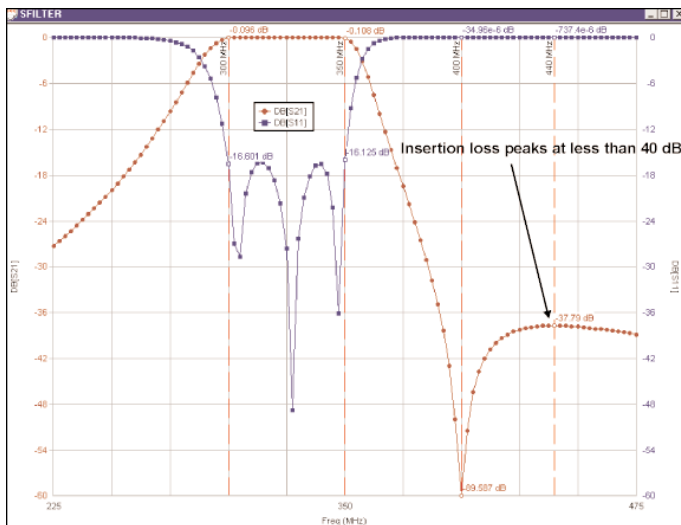
For many filters, hundreds, and in some cases, thou-

sands of different filter schematics can be generated for a single set of design criteria, with each schematic having the exact desired response. This gives the design engineer an unprecedented array of choices for schematic realization of a particular filter. Figure 3 shows S/FILTER within the GENESYS design environment.

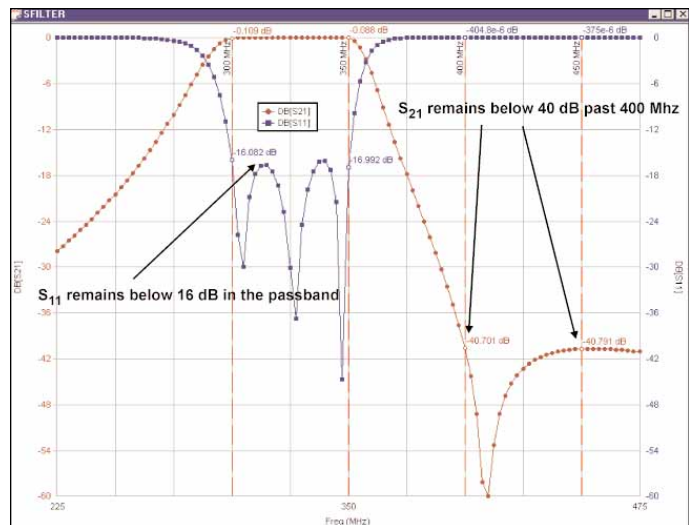
Design example

The filter to be designed has the following requirements:

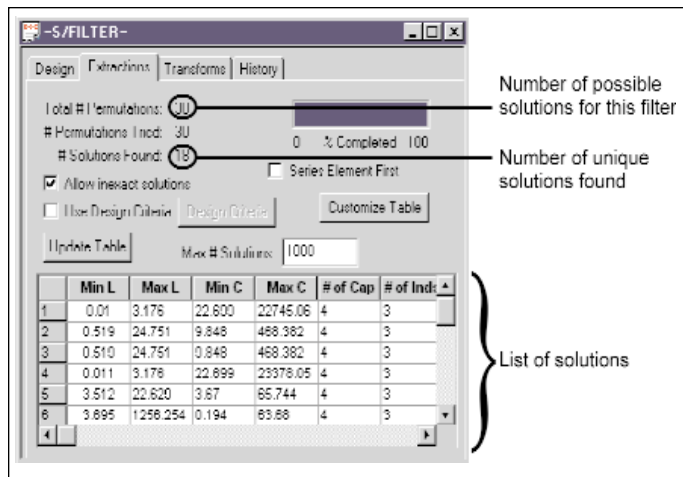
- 50 ohm terminations
- Passband from 300 to 350 MHz
- Return loss > 15 dB within the passband
- Insertion loss > 40 dB above 400 MHz
- No transformer



▲ Figure 7. Filter response after adding the 400 MHz transmission zero.



▲ Figure 8. Filter response after tuning the transmission zero to 409 MHz.



▲ Figure 9. S/FILTER extractions tab.

Figure 4 shows the initial S/FILTER screen with the design specifications entered.

In a lossless LC filter, the passband return loss can be calculated from the ripple value

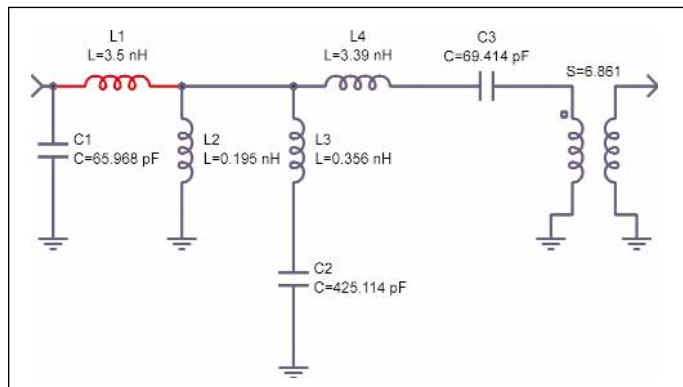
$$R.L.(dB) = -10 \times \log_{10}(\epsilon^2)$$

where

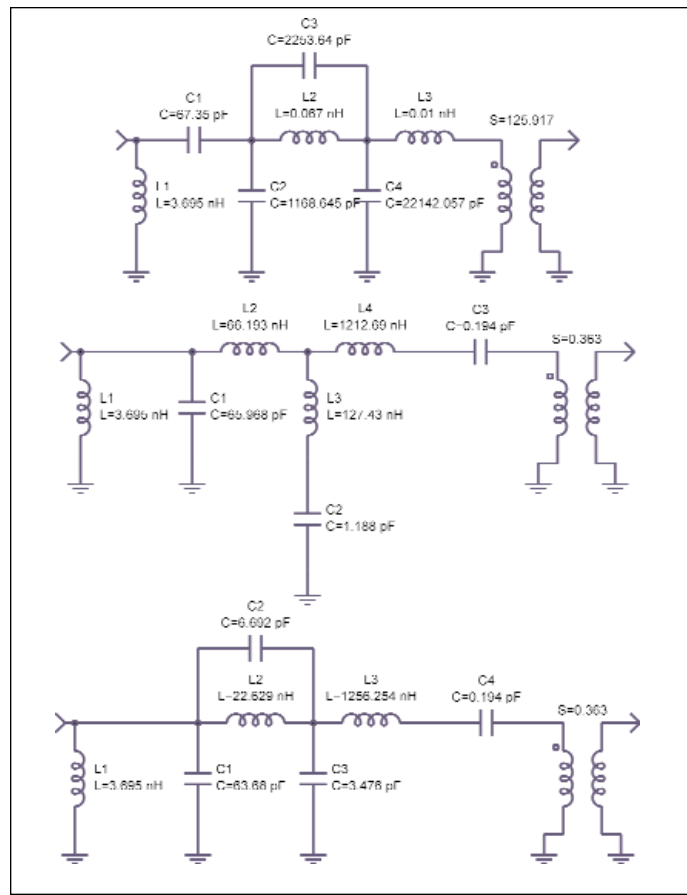
$$\epsilon = \sqrt{10^{\frac{R_{dB}}{10}} - 1}$$

and R_{dB} is the passband ripple in dB. The passband ripple for the example filter is set to 0.1 dB, which would correspond to just over 16 dB return loss in a lossless filter. The number of DC and infinite frequency zeros have each been initially set at 2. This corresponds to a traditional second order (4th degree) bandpass filter. Figure 5 shows the filter response based on the parameter set in Figure 4.

The passband return loss (S_{11}) is greater than 15 dB, per the design requirement. However, the marker in



▲ Figure 11. Initial schematic for the example filter.

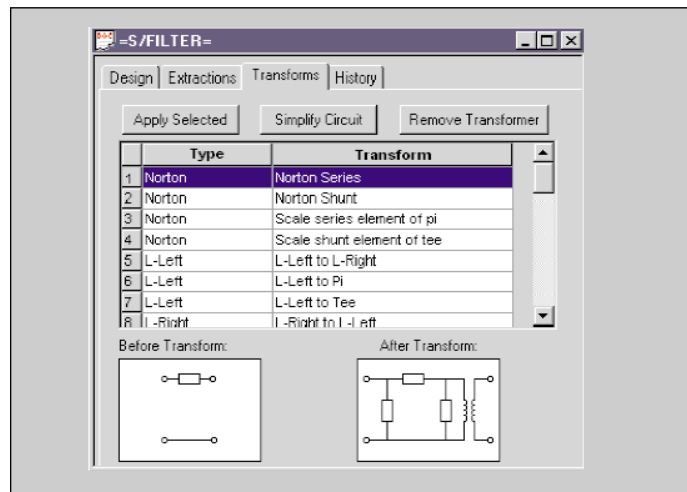


▲ Figure 10. Some of the solutions for the example filter.

Figure 5 at 400 MHz shows that the insertion loss (S_{21}) is only 7.5 dB. To improve this condition, a transmission zero is added at 400 MHz. Figure 6 shows the S/FILTER design tab with the new zero indicated.

Figure 7 shows the filter response after adding the transmission zero at 400 MHz. The frequency scale has been adjusted to show rejection at higher frequencies.

The insertion loss at 400 MHz is well below the



▲ Figure 12. S/FILTER transforms tab.

requirement of 40 dB. However, the marker at 440 MHz in Figure 7 shows that the insertion loss falls below 38 dB. To improve this, the transmission zero at 400 MHz can be tuned to a higher frequency. Figure 8 shows the filter response after the transmission zero has been tuned to 409 MHz.

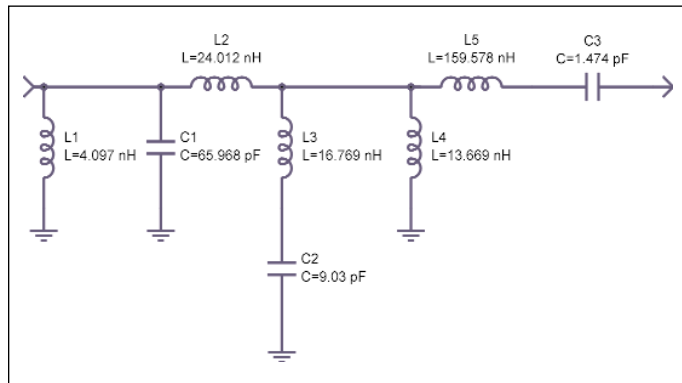
While tuning the frequency of the transmission zero higher improves the filter's stopband rejection, it also decreases the insertion loss at 400 MHz. With the zero tuned to 409 MHz, the insertion loss reaches 40 dB at 400 MHz and remains below 40 dB at higher frequencies. Next, the filter schematic is extracted. Figure 9 shows the S/FILTER extractions tab.

According to Figure 9, there are 30 possible ways to design the filter. A total of 18 unique solutions exist. Some of these solutions are shown in Figure 10. Notice that all of the solutions contain transformers. The transformer is removed later by performing circuit transformations on the schematic.

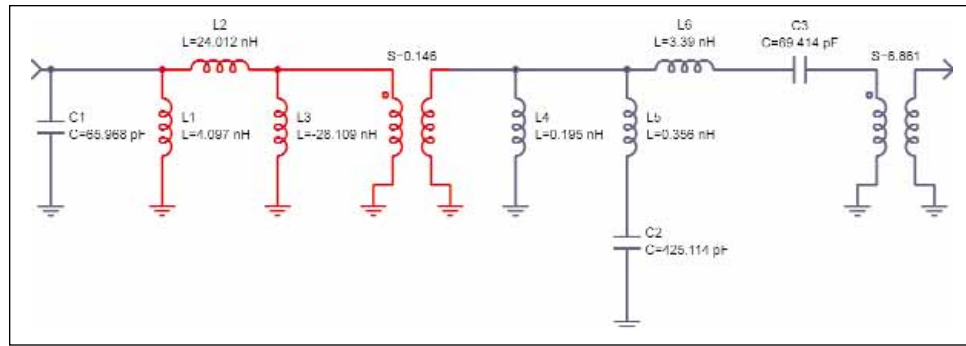
Notice that the filters in Figure 10 have very different topologies. Usually, one topology will be more favorable than another because of board layout or component value restrictions. For example, if coaxial resonators will be used in construction, it may be desirable to have all resonators in shunt. The schematic chosen for this filter is shown in Figure 11.

S/FILTER has more than 100 built-in circuit transformations. The transforms tab is shown in Figure 12.

To remove the transformer in the schematic in Figure 11, a Norton Series transform is applied to the inductor drawn in red in Figure 11. This transform replaces a series element with a three element pi network and transformer. S/FILTER has an option to calculate the new transformer ratio as the inverse of the existing transformer. Then, all transformers are combined to yield a transform ratio of 1 (no net transformer). The schematic after applying the Norton Series transform is



▲ Figure 14. Final filter schematic.



▲ Figure 13. Filter schematic after applying a Norton Series transform.

shown in Figure 13.

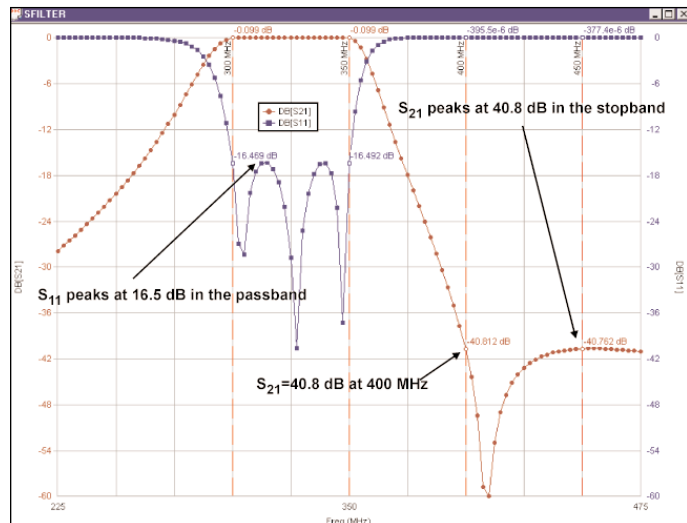
The elements drawn in red in Figure 13 are the result of the Norton Series transform. Notice that one of the red inductors (L3) is negative. Norton transforms always produce a negative element. In this case, the positive shunt inductor on the same node (L4) can absorb the negative element. Figure 14 shows the simplified schematic obtained by clicking the "Simplify Circuit" button shown in Figure 12.

S/FILTER offers an option to automatically search through the filter schematic for ways to apply Norton transforms to achieve an overall positive network. This option automates the procedure described above to remove any transformers from the schematic. Figure 15 shows the response for the final filter design shown in Figure 14.

S/FILTER features

With the new S/FILTER, engineers are able to

- Design lowpass, highpass, and bandpass filters;
- Directly synthesize multiple solutions for each design;
- Tune zero locations to optimize a filter design;



▲ Figure 15. Insertion and return loss for the schematic shown in Figure 14.

- Design classic transfer function types using the Shape Wizard (Butterworth, Chebyshev and Elliptic);
 - Asymmetrically place DC and infinite zeros to increase high- and/or low-side filter selectivity for overall response symmetry;
 - Enter criteria for design solutions based on component such as values and ratios;
 - Completely customize the solution table — display solutions and sort them by any parameter;
 - Control extraction sequence with series or shunt first element;
 - Use more than 100 available circuit transformations, including Norton, Pi and Tee;
 - Apply transforms to create component symmetry;
 - Apply transforms to create equal-inductor filters;
 - Design exact solutions for narrow- and wide-band filters;
 - Create arbitrary stopbands by placing and tuning finite transmission zeros;
 - Remove transformers automatically by applying the “Remove Transformer” macro;
 - Design “non Orchard” filters.
- In the case of these “non Orchard” filters, due to precision concerns, many filter extraction sequences are not allowed in the original research that S/FILTER is based on [1]. However, S/FILTER has an option to force these extractions, which may exhibit some non-exact response behavior. These filters often have fewer parts or more favorable topologies than the exact solutions have, and can be optimized or tuned to exhibit more exact behavior. ■

References

1. H.J. Orchard and G.C.

Temes, “Filter Design Using Transformed Variables,” *IEEE Transactions on Circuit Theory*, Volume CT-15, December 1965, pp. 385-408.

Author information

Glenn Parker is a research and development engineer with Eagleware. He may be reached at the address given below.

For more information, contact:

Eagleware Corporation
635 Pinnacle Court
Norcross, GA 30071
Tel: 678-291-0995
Fax: 678-291-0971
E-mail: eagleware@eagleware.com
Internet: www.eagleware.com

Or circle Reader Service #200