LADDER—A Microcomputer Tool for Passive Filter Design and Simulation

Robert D. Koller and Bogdan Wilamowski, *Senior Member, IEEE*

Abstract—Modern analog filter designs fit into two categories: cascade prototypes and ladder prototypes. Although software programs for cascade filters are common, the same cannot be said for ladder filters. In order to teach advanced topics such as element frequency transformations, simulated passive elements, leapfrog implementations, and switched capacitor implementations, students must be comfortable with passive filter design. The use of tables for ladder filter design is very complex and only offers a black box approach to the problem. Although many algorithms exist, most are too computationally complex to use in a classroom. A software package for advanced analog filter design and simulation has been developed. This new package has many features, including polynomial manipulation, transfer function design, and post design simulation. This paper includes a description of LADDER, along with some example filter designs, which help demonstrate LADDER's usefulness in education.

I. INTRODUCTION

ADVANCED courses in analog filters deal with the complex subjects of passive ladder filters and modern implementations such as Girling–Good, switched capacitor, and switched current filters. Many software packages exist for the design of cascade type filters [1], [2], and some also offer features that aid in the introductory material for the design of ladder filters [1]. Unfortunately, cascade-type filters are very sensitive to element variations. Lower-sensitivity filter implementations, such as leapfrog forms, require ladder prototypes as a model. A good software package for teaching advanced filter synthesis would not only include many of the features of current cascade synthesis programs, such as an easy menu driven format, various design approximations, graphics, and polynomial manipulation, but also ladder filter synthesis with insight to the design procedure, circuit diagrams, and post-design simulation. The new software package LADDER fits these guidelines and provides a good tool for teaching of advanced analog filters. While employing the same user interfaces, the new program LADDER does for passive filter designs what FILTER [1] did for cascade filter designs.

Traditionally, analog filter designs require students to become familiar with the use of complex passive filter tables [3]. Instruction on the use of these tables takes as much lecture time as some other more desirable topics. Also, shortcomings due to incomplete filter listings can limit the scope of the course. Finally, filter tables give only a final solution, without a clear description of the actual ladder design procedure. The program

LADDER is characterized by easy to use pop-up menus, so little instruction is needed. The general approach [4] taken toward all design approximations allows LADDER to simulate any filter transfer function with a ladder circuit. Additionally, by combining the polynomial manipulation routines with the step-by-step design option, both included in LADDER, a student gets the opportunity to work a particular problem "by hand," thus gaining insight into the design process.

Highly interactive menu driven screens allow the users of this program to design and simulate ladder prototypes quicker and easier than procedures involving traditional tables [3], making LADDER a much superior choice for classroom uses. Past programs for the synthesis of analog ladder filters [5], [6] needed several highly specialized algorithms to synthesize all filter types. FILSYN and ELLIP use Cauer's algorithm [7] for the synthesis of Cauer-Elliptic filters. To reduce numerical accuracy problems, FILSYN uses an auxiliary function approach for filters without zeros and a special modified version of Cauer's approach for Inverse Chebyshev filters. All of these algorithms generate only one circuit for each transfer function. LADDER uses a general approach [4] for ladder synthesis. The generality of the algorithm allows users to synthesize any transfer function into a ladder implementation. In addition, for transfer functions with zeros, for example Cauer and Inverse Chebyshev filters, this algorithm will generate several possible prototypes where tables and other programs produce only one. Although some realizations may not give positive, well-scaled elements, others may have better properties than the one-circuit realization given in tables or generated by other ladder synthesis programs. This algorithm can generate up to $N_z!$ possible circuits for one transfer function with zeros, where N_z is the number of conjugate zeros. For example, a sixth-order Cauer filter with three conjugate zeros could be realized by up to six different circuits.

In addition to the ladder synthesis algorithm, an algorithm for circuit simulation, Monte Carlo, is included. This algorithm takes advantage of some of the known properties of ladder circuits for faster results than do general circuit simulation programs such as SPICE; however, if desired, LADDER also generates SPICE files. Although a design is completed using ideal elements, nonideal elements can be used in the simulation to see how the circuit losses affect the frequency response.

Section II describes the important features of the program LADDER. LADDER includes built-in simulation of synthesized filters; Section III discusses the educational advantages of this feature. Sections IV–VI present worked examples demonstrating some of the features of LADDER. Section IV

```
─────────── LADDER RESISTANCES ───────────

Input resistance in ohms                    1.000000
Output resistance in ohms                   1.000000
1/Q for inductor Q=L/R                       0.000000
1/Q for capacitor Q=1/RC                     0.000000
 Minimum inductor-1 or capacitor-0 circuits  1

      Press Esc when done
```

Fig. 1. Ladder circuit specification menu.

synthesizes a low-pass filter without zeros and demonstrates the "step-by-step" feature. A low-pass filter with zeros is demonstrated in Section V. This example is chosen to demonstrate how to synthesize different circuits by rearranging the zero order. Section VI illustrates the use of element transformations to build a bandpass filter. The benefits of using LADDER as a tool for active filter implementations are discussed in Section VII. Finally concluding remarks along with how to obtain the program LADDER are in Section VIII.

II. DESCRIPTION OF THE LADDER PROGRAM

The software package LADDER offers many pop-up menus to guide a user through the design process. Upon selection of "Filter Synthesis" from the first menu, the user is taken step by step through the entire specification process. LADDER has all of the features of FILTER [1] for transfer function development and testing. Transfer function approximation algorithms include Butterworth, Chebyshev, Inverse Chebyshev, Cauer, and Bessel–Thompson. Furthermore, graphics routines allow users to see pole–zero locations and how they affect magnitude, phase, delay, and transient responses.

LADDER performs synthesis by first transforming the filter specifications to a low-pass prototype, then synthesizing the low-pass prototype, and finally doing element transformations to get the proper filter form. Because low-pass transfer functions that have equal numerator and denominator orders are not implementable in a ladder form, special frequency transformations [4] are added to the transfer function development. In order for a filter to be realizable, the order of the denominator must be greater than the order of the numerator. A special frequency transformation transforms the largest conjugate zero pair to infinity, thus reducing the order of the numerator by two. In addition, another transformation is introduced for filters without a maximum at zero, for example even order Cauer and Chebyshev filters. For these filters, the first maximum, on the frequency scale is transformed to zero. The second transformation allows the synthesis of even-order Cauer and Chebyshev filters with equal terminating resistances and 0-dB gain.

After designing the desired transfer function, the routines for ladder synthesis are invoked. Ladder synthesis starts with a menu, shown in Fig. 1, for the specification of the terminating resistances, inductor, and capacitor Q's (for simulation only), and the specification of a circuit with a minimum number of inductor or capacitor elements. Capitalized letters indicate the

```
─────────── LADDER ───────────

1 - Display the data
2 - draw the Circuit
3 - PSPICE parameter Setting
4 - Generate the pspice file
5 - Manually arrange zeros
6 - Re-synthesize circuit
7 - run Monte carlo simulation
8 - make New monte carlo array
9 - step By step synthesis
0 - standard Values
  - Quit
```

Fig. 2. Menu of LADDER's passive filter synthesis features.

"hot key" for that particular menu selection. After making the desired specifications, the synthesis is performed and a Ladder menu appears, as shown in Fig. 2. Included in this menu are the following choices.

1) "Display Data." This option displays the terminating resistances along with a list of the calculated capacitor and inductor values, displayed in a logical order.

2) "Draw the Circuit." Selection of this option will draw a picture of the designed circuit labeled with the calculated element values.

3) "PSPICE Parameter Settings" allows the specification of some of the SPICE file parameters. Some of the choices include element tolerances for Monte Carlo simulation, number of data points, number of Monte Carlo simulations, ideal or nonideal opamps, and magnitude and frequency scaling.

4) "Generate the PSPICE File." This option will generate, display, and save to disk a file usable by SPICE simulation programs.

5) "Manual/Automatic." Selection of option 5) will toggle back and forth between manual and automatic ordering of zeros. Selection of option 6) (below) with "Automatic" zero ordering will generate a filter that matches designs found in tables [3]. Selection of option 5) with "Manual" zero ordering will allow the creation of many possible circuits for one transfer function.

6) "Resynthesize Circuit" will begin the ladder synthesis process again, taking into account what has been selected in option 5).

```
LOWPASS FILTER - Chebyshev   7 order
ap = 5.000000dB   as = 70.00000dB      asr = 116.8739dB      POLYNOMIALS:
fp=1.000000  fs=2.000000                                     power denominator numerator
normalized wp=1.000000  ws=2.000000                            7  1.00000000
NORMALIZED POLE AND ZERO LOCATIONS:                            6  0.40903194
   1 SINGLE POLES:          6 CONJUGATE POLES:                 5  1.83365356
  -0.091018      +j0            -0.0820045  ±j0.43567725       4  0.58805499
                               -0.0567489  ±j0.78506327        3  0.96564058
                               -0.0202534  ±j0.97895788        2  0.20812079
  (s² + 0.164009s + 0.196539)  Q=2.703068 wo=0.443328          1  0.12800710
  (s² + 0.113498s + 0.619545)  Q=6.935039 wo=0.787112          0  0.01062587  0.01062587
  (s² + 0.040507s + 0.958769)  Q=24.17286 wo=0.979167
  (s + 0.091018)
  magnitude scaling factor   0.0106258650
  frequency scaling factor   1.0000000000
```

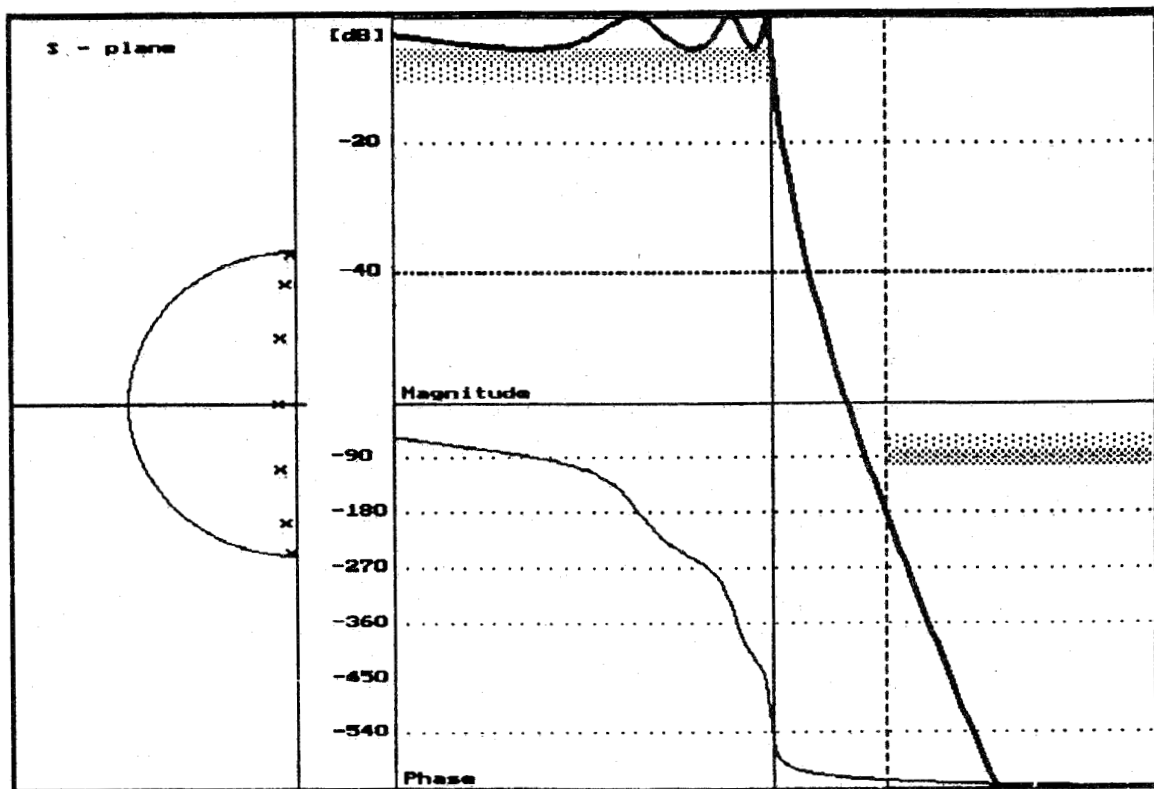Fig. 3. Transfer function data for the example of Section IV.

Fig. 4. Frequency response of the transfer function shown in Fig. 3.

7) "Monte Carlo." This option will graph the Monte Carlo simulation of the currently designed circuit.

8) "Create Monte Carlo Array." Choose this option if the current Monte Carlo array needs to be changed for some reason, for example, element tolerances were re-specified.

9) "Step by Step Synthesis." By selecting Step by Step synthesis, a student can follow along with the design process, with many intermediate steps being displayed.

10) "Standard Values." An input menu displaying the designed element values appears, allowing the user to replace the designed values with standardized element values.

11) "Quit" returns to the transfer function design menu.

III. LADDER SIMULATION TOOLS

Monte Carlo analysis was found to be a very educational tool. It gives a graphical illustration of how characteristics can change when elements are not exactly the calculated value; also, Monte Carlo simulations show the sensitivity of a circuit when all of the elements are varied simultaneously. For these reasons, Monte Carlo analysis was directly incorporated into the program.

In Monte Carlo simulation, using a uniform distribution with the desired tolerances, an array of randomly selected element values is generated and a transfer function using these new values is found. The frequency response of the new transfer function is plotted. Because of the unique opportunity

Rin= 1

Ca1= 0.778236

La2= 0.09191

Ca3= 0.999555

La4= 0.095255

Ca5= 0.999564

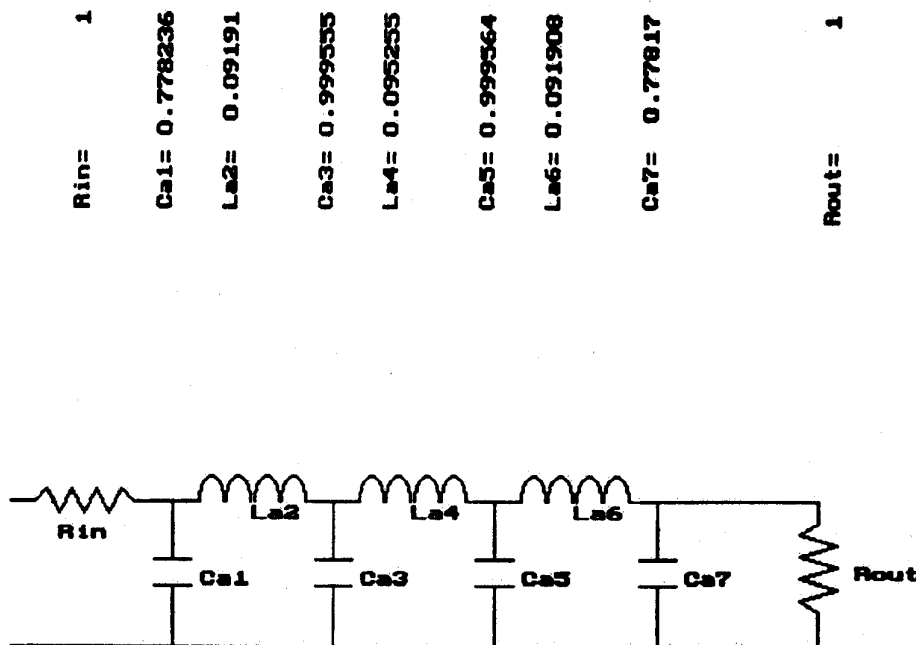La6= 0.091908

Ca7= 0.77817

Rout= 1

Fig. 5. Circuit diagram, drawn by LADDER, of the synthesized Chebyshev filter.

offered by LADDER, namely the possibility of many circuits for one transfer function, circuit simulation is important for finding the best circuit from a list of possible circuits. This simulation gives the students an opportunity to immediately see the advantages of one particular circuit over another. Using this Monte Carlo simulation, students visualize various aspects of sensitivity not easily seen in equations. Experiments demonstrating the sensitivity of ladder circuits to the terminating resistance ratios can be performed quickly and easily.

The software package LADDER also introduces nonideal element parameters into the simulation. With this feature, students can witness first-hand not only the effects of randomly varied element values, but also the effects of element losses on the frequency response.

The program can also generate a SPICE netlist. Although less convenient, by using this feature users can perform simulations on other circuit simulation programs. This is especially important when working with active filter implementations.

IV. EXAMPLE OF A LOW-PASS PROTOTYPE WITHOUT ZEROS

To illustrate the synthesis of a transfer function without zeros, a seventh-order low-pass Chebyshev filter is shown. First, the desired transfer function must be designed. LADDER allows brickwall specification, so selecting "Filter Synthesis" and then "Magnitude Specification" allows the input of the brickwall specifications for the desired filter. Next, selecting "Approximation" gives the user a chance to select from the many available approximation methods. In this example, a low-pass Chebyshev filter with attenuations: $\alpha_p = 5$ dB, $\alpha_s = 70$ dB, and corner frequencies: $\omega_p = 1.0$, $\omega_s = 2.0$, is designed.

After the specifications are entered, "Results" is selected. From the "Results" menu, relevant transfer function data is seen by selecting "Display Data," and graphs of the frequency

```
————————— MANUAL SYNTHESIS —————————
1 - find Td(s)Td(-s) and Tn(s)Tn(-s)
2 - find the Auxiliary function A(s)A(-s)
3 - Find A(s) from A(s)A(-s)
4 - find Z(s)
5 - find the Element values
6 - transform elements
7 - Quit
```
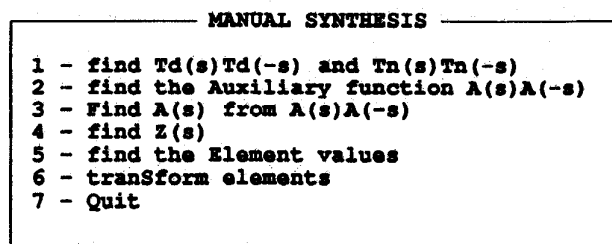
Fig. 6. Menu of the steps shown by LADDER in a "step-by-step" synthesis.

characteristics are seen by selecting "Graphics." Fig. 3 shows the screen with transfer function data for the designed filter. Fig. 4 is the screen with a graph of the pole and zero locations as well as the magnitude and phase responses.

Once the desired filter has been designed, the ladder synthesis is started with the selection of "Ladder." At this point, an input menu requesting the input and output resistances appears (Fig. 1). For this example, $R_{in} = R_{out} = 1.0$ is chosen. The ladder synthesis is performed and the "Ladder" menu appears (Fig. 2). Fig. 5 is the designed circuit diagram created by selecting "draw the Circuit."

In this example, the "Step By Step Synthesis" will be demonstrated. This choice is particularly helpful because is shows the important intermediate steps in the synthesis process. Because of its general approach to all filter types, the ladder synthesis algorithm used by LADDER calculates the element values by first finding an auxiliary function from the transfer function, then calculating the impedance function, and finally finding the elements using one of two continued fraction expansion forms. For transfer functions without zeros, the standard continued fraction expansion is applied. Transfer functions with zeros require a more complex continued fraction form [4]. Fig. 6 shows the "Manual Synthesis" menu,
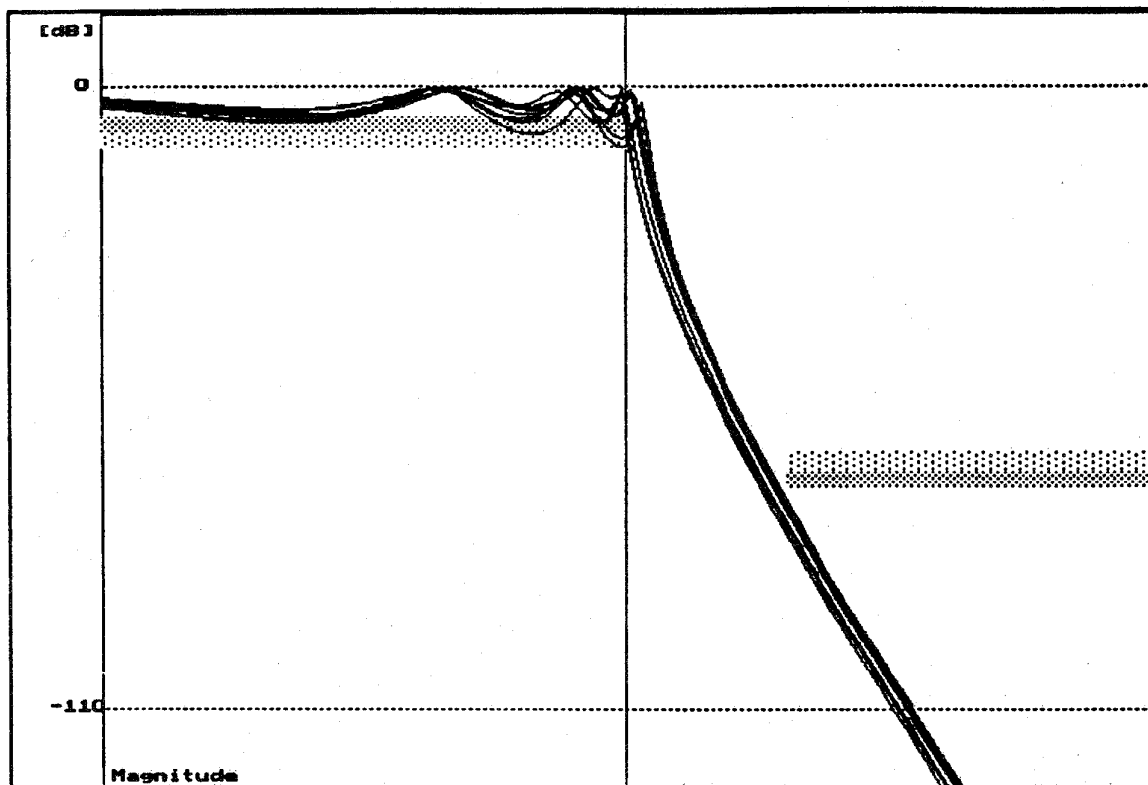
Fig. 7. Monte Carlo simulation of the circuit in Fig. 5, using the built-in simulation routine and 20% element tolerances.

Rin= 1

La1= 1.101179

Lb2= 0.115507
Cb2= 0.052463
La3= 1.354747

Lb4= 0.071147
Cb4= 0.057408
La5= 1.43085
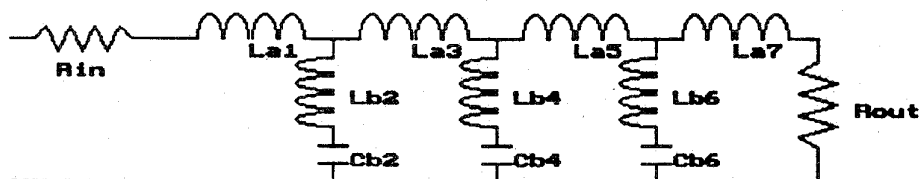
Lb6= 0.023966
Cb6= 0.055742
La7= 1.172695

Rout= 2

Fig. 8. Synthesized circuit for the example of Section V, using $R_{\mathrm{in}} = 1.0$ and $R_{\mathrm{out}} = 2.0$.

indicating the intermediate steps performed and shown by LADDER. Calculation of the auxiliary function, $A(s)$ [8], starts with the calculation of $A(s)A(-s)$, shown in (1) at the bottom of the next page. Next $A(s)$, derived as in (2), shown at the bottom of the next page, is derived from $A(s)A(-s)$. Then, the impedance function, derived as in (3), shown at the bottom of the next page, is displayed using the "Find $Z(s)$" menu option. Finally, the circuit elements are found

and displayed by selecting "find the element values." If a form other than low-pass is desired, the element values are transformed to the desired filter form (highpass, bandpass, bandstop) by individual element transformations and the final circuit is listed

After returning to the "Ladder" menu, selection of "Run Monte Carlo Simulation" initiates the built-in Monte Carlo simulation routine. The Monte Carlo simulation, with 20% tolerances, is shown in Fig. 7.

This highly interactive design procedure allows a student to see each step in the design process, instead of just the final result. The built-in Monte Carlo simulation also allows the student to see very quickly how well the design meets the specifications when random elements are chosen, within tolerances.

V. EXAMPLE OF A LOW-PASS PROTOTYPE WITH ZEROS

In the second example, a seventh-order Cauer-Elliptic filter is synthesized into several ladder forms. Using the same procedure described above, a Cauer-Elliptic transfer function with brickwall specifications, $\alpha_p = 5$ dB, $\alpha_s = 100$ dB, $\omega_p = 1.0$, $\omega_s = 2.0$, is designed. In this design, terminating resistances of $R_{in} = 1.0$ and $R_{out} = 2.0$ are used. Fig. 8 shows a screen with the circuit diagram of one possible set of element values.

Because this algorithm allows the design of more than one circuit for a single transfer function with zeros, LADDER offers user interaction in zero ordering. From the "Ladder" menu, first select option 5), "Manually Arrange Zeros," and then chose "Re-Synthesize Circuit." At this point, menus with the possible zero choices and their corresponding shunt capacitor are shown. Circuits found in tables [3] are generated by always selecting the smallest capacitor choice. Fig. 9 displays the successive menus for re-ordering synthesized zeros. By selecting a different order, a new circuit is created. Element values for this new circuit are given in Fig. 10. Table I shows all six possible circuits which can be generated from a seventh-order filter with zeros. A LADDER generated Monte Carlo simulation of set 6 from Table I, using 20% element tolerances, is shown in Fig. 11.

VI. EXAMPLE OF A BANDPASS CIRCUIT

Using LADDER, students may synthesize other filter types such as high-pass, bandpass, and bandstop. The desired filter response may be entered using brick wall specifications.
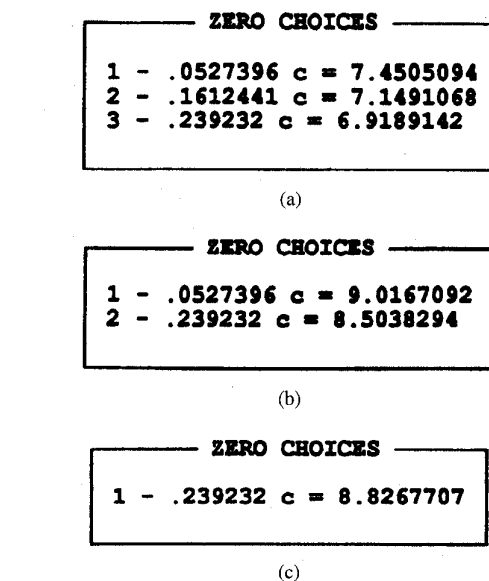

(a)

(b)

(c)

Fig. 9. Menus for selecting zero order: (a) first menu with three choices, (b) remaining choices after synthesizing the zero at 0.161, and (c) final zero choice.

LADDER transforms these characteristics to an equivalent low-pass filter using the appropriate frequency transformation (Table II). After synthesizing the low-pass circuit, individual elements are transformed, using Foster reactance functions [8], to the desired filter form.

In the final example, the seventh-order low-ass filter with zeroes of the second example is transformed into a bandpass circuit. The zero order is the same as Set 6 of Table II. Fig. 12 shows the newly synthesized bandpass circuit. Fig. 13 show a Monte Carlo simulation of the designed filter, using ideal elements and 10% tolerances. Typically, element losses due to capacitive elements are negligible, while inductive losses can be significant $(Q_L \approx 100)$. To illustrate the effects of lossy elements on the frequency response, the element Q's have been exaggerated slightly. Fig. 14 shows the Monte Carlo simulation of the bandpass circuit of Fig. 12 using an inductor Q equal to 30 and a capacitor Q equal to 1000.

VII. ACTIVE IMPLEMENTATIONS OF LADDER CIRCUITS

Modern practical realizations of passive circuits need ladder prototypes to finish the design. Because of its easy to use menu driven format, LADDER is superior to tables [3] for proto-

$$A(s)A(-s) = \frac{-s^{14} - 3.5s^{12} - 4.81s^{10} - 3.28s^8 - 1.15s^6 - 0.91s^4 - 0.0120s^2}{s^{14} - 3.5s^{12} + 4.81s^{10} - 3.28s^8 + 1.15s^6 - 0.191s^4 + 0.0120s^2 +} \tag{1}$$

$$A(s) = \frac{s^7 - 1.73e - 5s^6 + 1.75s^5 - 1.72e - 5s^4 + 0.875s^3 - 2.84e - 6s^2 + 0.109s}{s^7 + 0.409s^6 + 1.83s^5 + 0.588s^4 + 0.966s^3 + 0.208s^2 + 0.128s + 0.0106} \tag{2}$$

$$Z(s) = \frac{0.409s^6 + 0.0837s^5 + 0.588s^4 + 0.0906s^3 + 0.208s^2 + 0.0186s + 0.0106}{2s^7 + 0.409s^6 + 3.58s^5 + 0.588s^4 + 1.84s^3 + 0.208s^2 + 0.237s + 0.0106} \tag{3}$$

The input resistance is   1.000

```
L1[1]= 1.137816
C2[2]=   0.05538  L2[2]= 0.073751  Zero = 1.6124E-1
L1[3]= 1.435054
C2[4]= 0.061266  L2[4]= 0.021805  Zero = 5.2739E-2
L1[5]= 1.404824
C2[6]= 0.049349  L2[6]= 0.122795  Zero = 2.3923E-1
L1[7]=    1.025
```
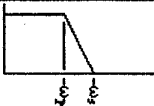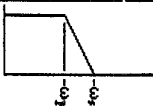
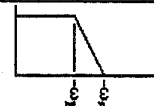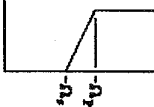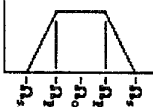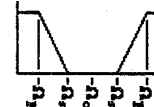**The output resistance is   2.000**

Fig. 10. Element values for a new circuit, obtained by changing the zero order.

TABLE I
ALL SIX POSIBLE CIRCUITS FOR THE EXAMPLE FILTER OF SECTION V

	Set 1		Set 2		Set 3		Set 4		Set 5		Set 6	
	L	C	L	C	L	C	L	C	L	C	L	C
1	1.101	0.000	1.101	0.000	1.138	0.000	1.138	0.000	1.186	0.000	1.186	0.000
2	0.116	0.052	0.116	0.052	0.074	0.055	0.074	0.055	0.023	0.059	0.023	0.059
3	1.355	0.000	1.401	0.000	1.353	0.000	1.435	0.000	1.398	0.000	1.433	0.000
4	0.071	0.057	0.022	0.061	0.111	0.054	0.022	0.061	0.111	0.054	0.071	0.057
5	1.431	0.000	1.466	0.000	1.394	0.000	1.405	0.000	1.349	0.000	1.351	0.000
6	0.024	0.056	0.078	0.052	0.024	0.055	0.123	0.049	0.078	0.052	0.121	0.050
7	1.173	0.000	1.065	0.000	1.188	0.000	1.025	0.000	1.100	0.000	1.045	0.000

TABLE II
FREQUENCY TRANSFORMATIONS TO CONVERT FROM TYPICAL FILTER FORMS TO A LOW-PASS PROTOTYPE

	HIGHPASS	BANDPASS	BANDSTOP
LOWPASS RESPONSE			
DESIRED RESPONSE			
PROTOTYPE STOPBAND CORNER FREQUENCY	$\omega_s = \dfrac{\Omega_p}{\Omega_s}$	$\omega_s = \dfrac{BW\,\Omega_{sr} + \sqrt{\left(\beta W \Omega_{sl}\right)^2 + \omega_0^2}}{BW\,\Omega_{pr} + \sqrt{\left(\beta W \Omega_{pl}\right)^2 + \omega_0^2}}$	$\omega_s = \dfrac{\dfrac{BW}{\Omega_{pr}} + \sqrt{\left(\dfrac{BW}{\Omega_{pr}}\right)^2 + \omega_0^2}}{\dfrac{BW}{\Omega_{sr}} + \sqrt{\left(\dfrac{BW}{\Omega_{sr}}\right)^2 + \omega_0^2}}$

type design. LADDER performs all frequency and magnitude scaling required before an active realization can be performed. Unlike tables, LADDER performs the element transformations necessary to achieve a desired filter form which eliminates messy calculations and reduces the opportunity for human error. LADDER easily converts between minimum inductor and minimum capacitor forms, displaying the desired circuit diagram. These conveniences allow students to concentrate on the topics of active realizations without the annoying restrictions of tables and the tedious calculations of various transformations.

Girling–Good form is the most common active implementation of passive filter prototypes. Other commonly used implementations are switched capacitor and GIC filters. Although these implementations are not performed by LADDER, examples illustrating these active designs using ladder prototypes are well presented in [8]. Additional information on active circuits used to simulate various Girling–Good stages is found in [9].

VIII. CONCLUSION

The program LADDER is a powerful tool for the teaching of advanced analog filter design and simulation. LADDER not only completes the ladder synthesis, but also aids in the simulation of designed ladder circuits. The ease of design offered by LADDER allows professors to increase examples without
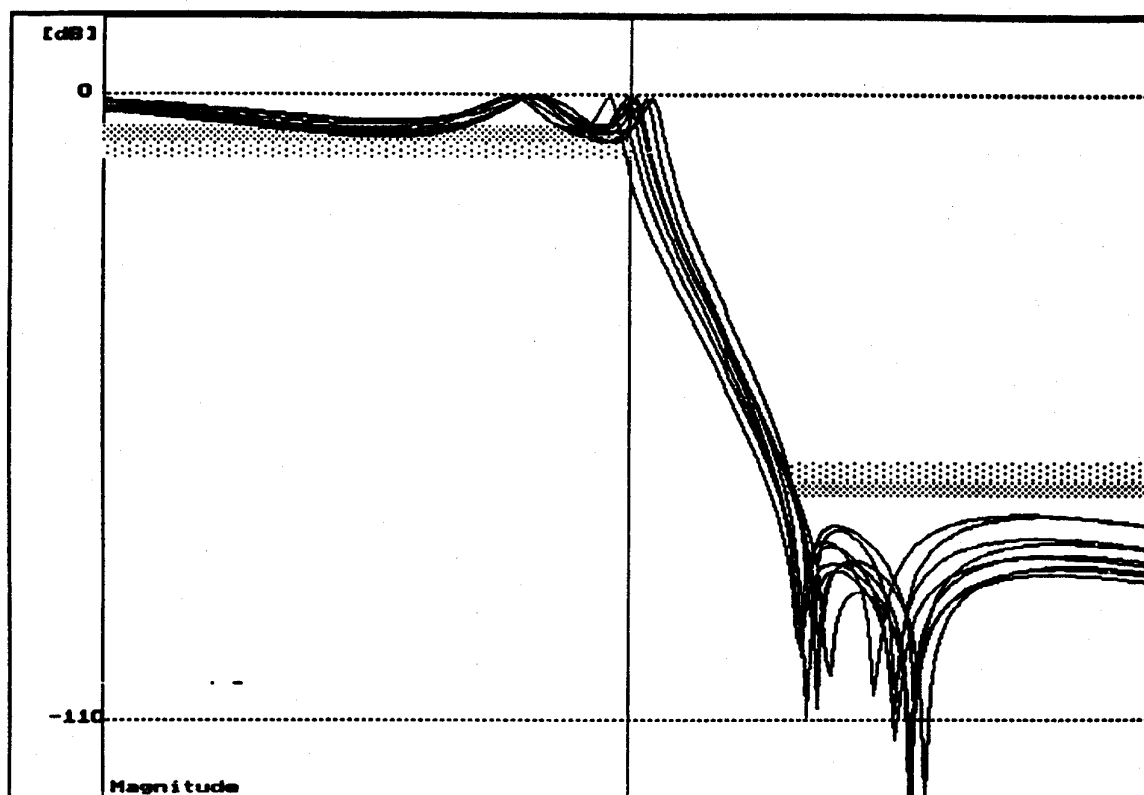
Fig. 11. Monte Carlo simulation of set 6 from Table I, with 20% element tolerances.
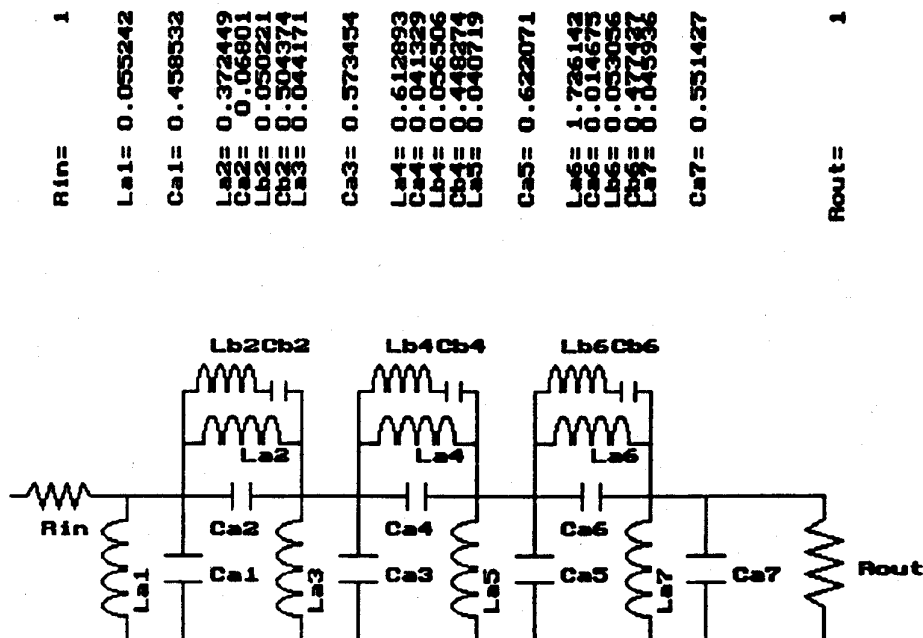

Fig. 12. Circuit diagram of the example bandpass filter of Section VI.

increasing student burden, thus giving students more experience in practical ladder filter problems. Practical filter designs that need ladder prototypes, such as Leap Frog forms, GIC circuits, and switched capacitor circuits, can use LADDER as a better alternative to tables for ladder prototype design. By including both design and analysis in one program, LADDER allows students to change various design specifications and instantly see the effects.

LADDER has been presented in [10]. This description includes some additional examples demonstrating LADDER.
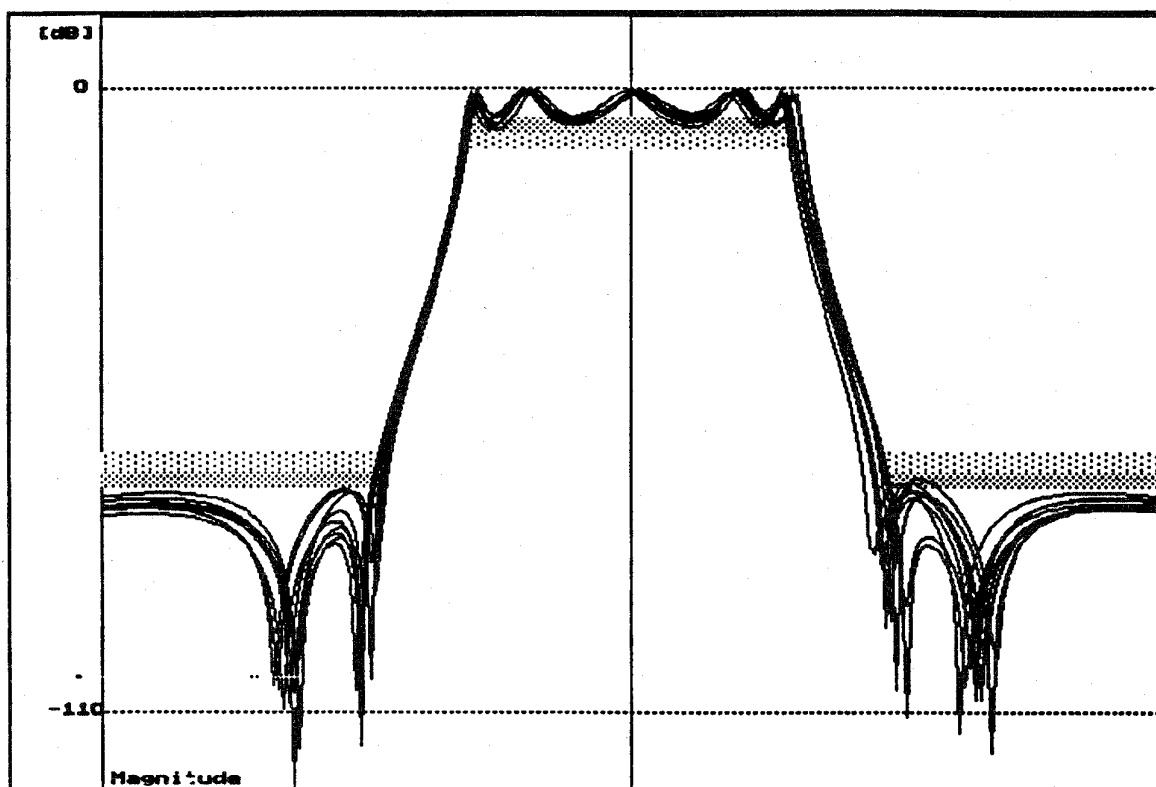
Fig. 13.   Monte Carlo simulation of the bandpass filter using ideal passive elements and 10% element tolerances.
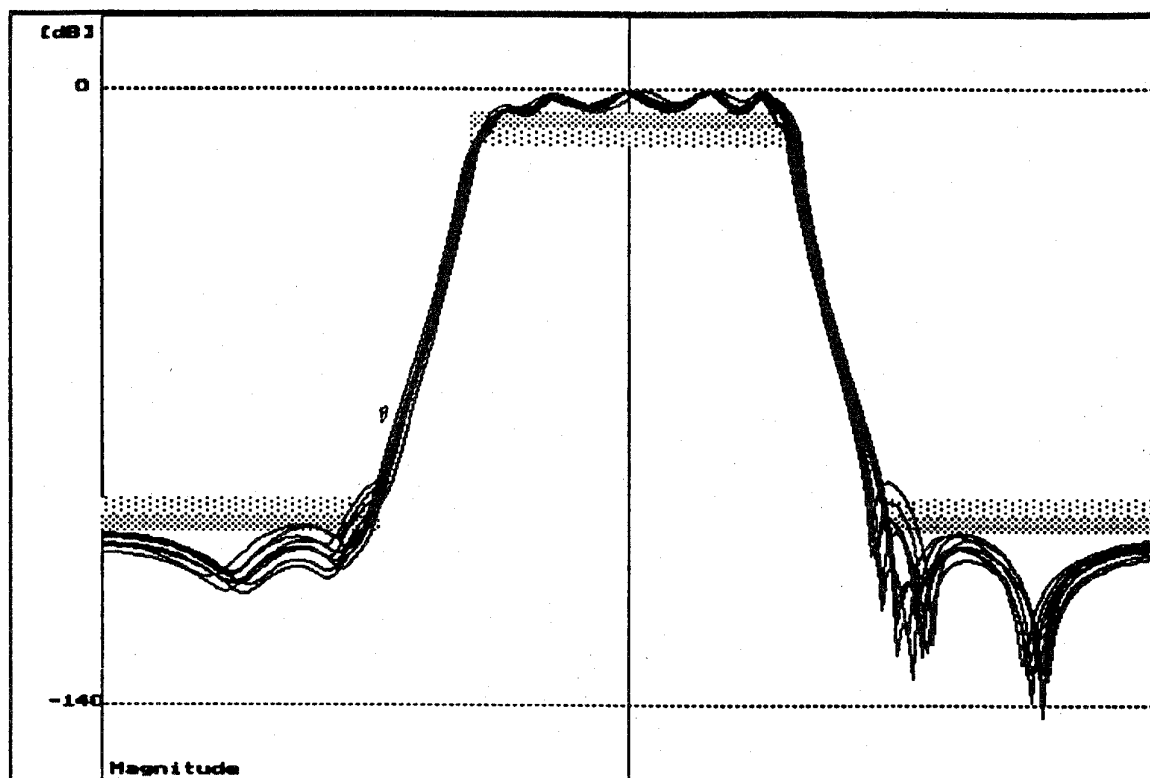

Fig. 14.   Same Monte Carlo simulation as in Fig. 13, except with lossy passive elements.

A copy of the programs LADDER and FILTER can be obtained for shipping and handling costs from Dr. Huelsman [10] or by anonymous FTP to *plains.uwyo.edu* in the *electrical directory*. This program works with IBM PC compatibles and was successfully tested on various graphical adapters such as Hercules, CGA, EGA, and VGA.

REFERENCES

[1] B. M. Wilamowski, S. Legowski, and J. W. Steadman, "Personal computer support for teaching analog filter analysis and design courses," *IEEE Trans. Educ.*, vol. 35, no. 4, pp. 144–152, 1992.

[2] M. R. Kobe, J. Ramirez-Angulo, and E. Sanchez-Sinencio, "FIESTA—A filter educational synthesis teaching aid," *IEEE Trans. Educ.*, vol. 12, no. 3, pp. 280–286, Aug. 1989.

[3] R. Saal, *Handbook of Filter Design*. AEG-Telefunken, 1979.

[4] R. D. Koller and B. M. Wilamowski, "A general algorithm for ladder prototype design," in *35th Midwest Symp. Circuits and Systems*, Washington, DC, USA, Aug. 9–12, 1992.

[5] G. Szentirmai, "FILSYN—A general purpose filter synthesis program," *Proc. IEEE*, Oct. 1977, vol. 65, pp. 1443–1458.

[6] D. J. M. B. Lopez, "Synthesis and sensitivity analysis of elliptic networks," Ph.D. Dissertation, University of Arizona, 1979.

[7] W. Cauer, *Synthesis of Linear Communication Networks*. New York: McGraw-Hill, 1958.

[8] M. E. Van Valkenburg, *Analog Filter Design*. New York: Holt, Rinehart, and Winston, 1982.

[9] R. C. Dorf, *The Electrical Engineering Handbook*. Boca Raton, FL: CRC, 1993, pp. 683–691.

[10] L. P. Huelsman, "PC programs for engineers," *IEEE Circuits and Devices Mag.*, Sept. 1993, pp. 3–4.

Robert D. Koller, biography not available at the time of publication.

Bogdan Wilamowski (SM'83) received the M.S. degree in computer engineering in 1966, the Ph.D. degree in neural computing in 1970 and the D.Sc. degree in integrated circuit design in 1977, all from Technical University of Gdansk, Gdansk, Poland.

He has been with the Technical University of Gdansk since 1966. He become Associate Professor in 1978 and Professor in 1987. During 1979–1981, he was the Director of the Institute of Electronic Technology and during 1988-1989, he was the Head of the Solid State Electronic Chair at this university. From 1968 to 1970 he was with the Nishizawa Laboratory at Tohoku University, Japan. In 1975-1976, he spent one year at the Semiconductor Research Institute, Sendai, Japan, as JSPS fellow. During 1981-1982, he was a Faculty Member at Auburn University, Auburn, AL. During 1982–1984 he was a Visiting Professor at the University of Arizona, Tucson. Since 1989 he has been an Associate Professor (Professor since 1991) in the Electrical Engineering Department of University of Wyoming, Laramie. In 1995–1996 academic year he was with the Alabama Microelectronics Science and Technology Center. He is the author of three books, more than 180 refereed publications, and 27 patents in seven countries. His main areas of interest are: neural and fuzzy systems, semiconductor devices, electronic and integrated circuits, analog and passive filters, computer simulation and modeling.