Computing Resources for Filter Design: Selecting a Properly Tuned Toolkit for the Classroom

Jerry C. Hamann
University of Wyoming

Abstract

A wide variety of software tools are available for use in the instruction of analog electronic filter design. In this paper, the selection of appropriate tools for both introductory and advanced filter design courses is examined. The main thesis of the paper is that "less may actually be more)' in the interest of our students.

1 Introduction

A wealth of software is available for various aspects of the electronic filter design task (refer to the concise overviews in [1]–[4]). Examples run the gamut from simple spreadsheets for completing fundamental design calculations to complex GUI-based systems [5] which integrate the steps of specification, response form selection, circuit design, and ultimately design simulation. Selecting the appropriate tools for use in the classroom can be a daunting task due to the large number of choices available. However, choosing the biggest, most comprehensive, and best integrated software suite may not be the best choice.

The focus of this paper is a proposed "well-tuned" selection of software tools for the classroom. The driving theme in the discussion is the goal to provide tools which complement a novice yet advancing level of filter synthesis maturity, That is, we contend that the student is best served by tools which do not "automate" the design process but rather "enable" it via reduction of mundane calculations. To support this thesis, an inventory of filter design "tasks" faced by the student is provided. The algorithmic nature of the design process quite naturally lends itself to software automation, however by providing too much automated guidance, the software may actually usurp most if not all intellectual responsibility for the design. In this scenario, the students' facility with filter design is an artificial one, critically depending upon access to a particular, familiar design automation tool.

Examples of representative software tools for classical analog filter design, both "homebrewed" and commercial, are discussed from the pedagogical viewpoint of "enabling" versus "automating." A cohesive set of well-tuned tools is then demonstrated for a typical filter design. The author's experiences with applying these tools in the classroom setting are summarized as appropriate to indicate the varying levels of student maturity and computational requirement.

2 The Design Setting

The focus of the filter design setting examined in this paper can be summarized as "high-order active filter synthesis for audio frequency bandwidths." The classical approach to this task, as undertaken in texts such as Van Valkenburg [6], involves the following tasks:

(1) expressing the desired filter specifications in lowpass attenuation form, referring to the "brick walls" of Figure 1,

$$\alpha(\omega) = -20\log_{10}|T(j\omega)|,$$

utilizing a frequency transformation for converting bandpass, highpass, and bind-elimination responses to a lowpass prototype if applicable,

(2) determination of the lowpass filter order and character (pole and zero positions), typically selecting one of several response forms (Butterworth, Chebyshev, Inverse Chebyshev, Cauer/Elliptic, Bessel/Thomson, etc.),

(3) transformation of the lowpass filter character back to the desired passband response type, if applicable,

(4) selection and component assignment of cascade active filter circuits to provide the required pole and zero combinations,

(5) simulation and tuning of the filter circuit.

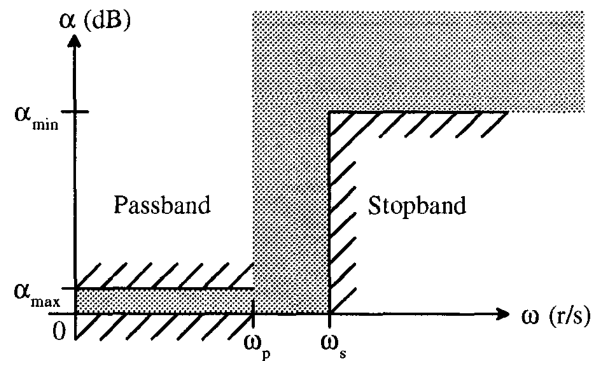

Figure 1: ***Lowpass Filter Attenuation Specifications***

3 The Instructional Goals

The instructional goals should not be to simply provide the student with a concise set of instructions for carrying out a filter design, but rather the student should be challenged to describe both how the design process steps provide a desirable path to the final design and how this process can be amended to obtain desirable effects. A critical and demonstrable example is knowledge of how tolerance can be included within the design process to result in a final design which is robust to electronic component variation. The "automated tools" might either ignore these possibilities or include them in a manner which is less than obvious to the "user. " And, as the verbiage implies, the ultimate objective is to develop filter knowledge and design skills, versus simply skill in using a particular automation tool.

4 The Essential Computations

While the "slick" automation features of the software described in [1]–[5] are difficult to resist, we argue that the novice filter designer is better served by use of basic computational tools. The application of basic tools will most likely require some implementation effort on the part of the student, and, indeed, this effort is essential to building the requisite knowledge for filter design. The tools need not be "pretty." In fact, limiting "bells-and-whistles" helps improve the "substance-to-noise" ratio. The essential calculation tasks are summarized in Figure 2.

5 A "Well-Tuned" Selection of Tools

In the spirit of providing continuity within the students' coursework, leveraging upon familiar tools is a natural goal. The difficulty of the calculations involved in classical filter design are readily accomplished with conventional spreadsheets, not to mention hand-held calculators and numerical tools such as Matlab. Some

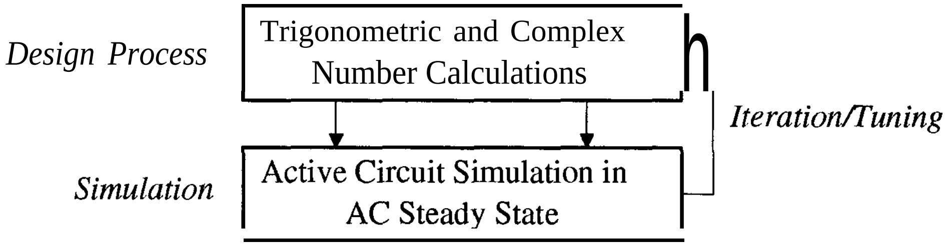

Figure 2: *The Essential Computational Tasks*

sophistication is required to carry out a few of the advanced procedures, for example the elliptic function evaluations associated with Cauer filters and the frequency transformation techniques for high order bandpass and band-elimination filters. However, with careful development by the instructor, these numerical tasks are not beyond the scope of interest or ability of students. Most importantly, critical design choices can be implemented by students within simple "what-if" scenarios, where their own calculations provide realistic bounds on filter parameters and thus define the "flexibility" inherent in knowing the design process well.

The author has successfully utilized spreadsheets for summarizing the calculations and design choices from filter specifications to poles and zeros of the required response. These "design templates" can be of the students' own work, with the construction time equivalent to a single homework assignment for a typical design process, e.g. lowpass specifications to Chebyshev response in biquad pole form. The basic Matlab core [7], without any additional toolbox functionality, provides enough numerical horsepower to complete the most difficult tasks as noted in the preceding paragraph. Simulation of the active circuit designs follows naturally with standard simulators such as SPICE2, SPICE3, PSpice or B^2Spice.

6 An Example Design

An example "no-frills" spreadsheet template for completing the principle calculations for a Chebyshev lowpass filter design is demonstrated in Figures 3 and 4. The bold and shaded cells in Figure 3 represent both the desired filter specification data and the choices left up to the designer. The remaining cell calculations follow directly from formulae developed within the course lecture. For the example given, the student is left with the design choices for filter order, n, and ripple factor, ϵ. The supporting calculations provide the pertinent performance data to help in the "what-if" scenario. The pole position calculations in Figure 4 complete the mundane, rote manipulations required. The computations demonstrate ed here are representative of the skills developed in a freshman level computing introduction to spreadsheet applications in engineering.

The selection of circuit components to complete the physical filter design is also aided by the computational simplicit y of a spreadsheet. The template in Figure 5 demonstrates the design of one of the two second order active lowpass stages required for the Chebyshev implementation. The steps of frequency and magnitude scaling are naturally carried out, allowing the designer to select near optimal nominal components.

Finally, the circuit designed to this point can be simulated with the use of another familiar tool, the SPICE software employed in preceding circuit analysis coursework. The attenuation response in Figure 6 provides confirmation of the design skills.

7 Conclusion

In this paper the author proposes that perhaps the best computational tools for use in the instruction of active analog filter design are those which are already in the students' "bag of tricks": namely, spreadsheets and similar numerical tools as well as basic circuit simulators. While it is agreed that the filter design paradigm lends itself quite naturally to automation, we argue that the student may not be best served by the thoroughly

Chebyshev Lowpass Filter Design Parameter Calculator (Complete Design in Bold Item Choices)				
	ω p	ω s	α_max	α_min
Step 1	1000	2000	1	36
	n_calc	n		
Step 2	4.1863647	5		
	ε_max	ε_min	ε_avg	ε
Step 3	0.5088471	0.1742757	0.3415614	0.34
	ω hp	α(ω_p)	α(ω_s)	
Final	1061.2875	0.4750851	41.804036	

Figure 3: *Chebyshev Lowpass Design Input Spreadsheet*

Chebyshev Lowpass Filter Pole Locations					
a	cosh(a)	sinh(a)			
0.35994	1.0654792	0.367758			
k	(2k+1)/(2n)	Real σ_k	Imag β_k	Frequency ω_0_k	Quality Q_k
0	0.1	-113.6435	1013.331	1019.684	4.486326
1	0.3	-297.5226	626.273	693.3524	1.16521
2	0.5	-367.7581	6.53E-14	367.7581	0.5
3	0.7	-297.5226	-626.273	693.3524	1.16521
4	0.9	-113.6435	-1013.33	1019.684	4.486326

Figure 4: *Calculated Chebyshev Lowpass Pole Positions*

automated tools. Indeed, if we wish to provide our students with more than just "black-box" knowledge of filter design, the use of basic computational tools will provide a well-tuned arsenal for the classroom.

REFERENCES

[1] L. P. Huelsman, cd., "PC Programs for Engineers," ***IEEE*** *Circuits and Devices Magazine,* Vol. 10, No. 3, pp. 3–9, May 1994.

[2] L. P. Huelsman, cd., "PC Programs for Engineers," ***IEEE*** *Circuits and Devices Magazine,* Vol. 11, No. 1, pp. 6–7, January 1995,

[3] L. P. Huelsman, cd., "PC Programs for Engineers," ***IEEE*** *Circuits and Devices Magazine,* Vol. 11, No. 2, pp. 4-6, March 1995.

[4] L. P. Huelsman, cd., "PC Programs for Engineers," *IEEE Circuits and Devices Magazine,* Vol. 11, No. 3, pp. 4-6, May 1995.

[5] C. Ouslis, W. M. Snelgrove and A. S. Sedra, "A Filter Designer's Filter Design Aid: filtorx," *Proc. of* ***IEEE*** *Int. Symp. on Circ. and Sys.,* pp. 376-379, June 1991.

[6] M. E. Van Valkenburg, *Analog Filter Design,* Saunders College Publishing, Holt, Rinehart and Winston, 1982.

[7] *The Matlab User's Manual,* The MathWorks, Inc., 1994.

Biquad Lowpass Circuit Design		ω_0	k_m	Q
Sallen-Key Option K=1		1019.7	300000	4.49
Comp.	@w_0=1	After k_f	After k_m	Nominal
R_1=1	1	1	300000	300k
R_2=1	1	1	300000	300k
C_1=2Q	8.98	0.008807	2.94E-08	0.030u
C_2=1/2Q	0.111359	0.000109	3.64E-10	360p
R_A=OC	infty	infty	infty	
R_B=SC	0	0	0	

Figure 5: *Example Lowpass Filter Stage Design*

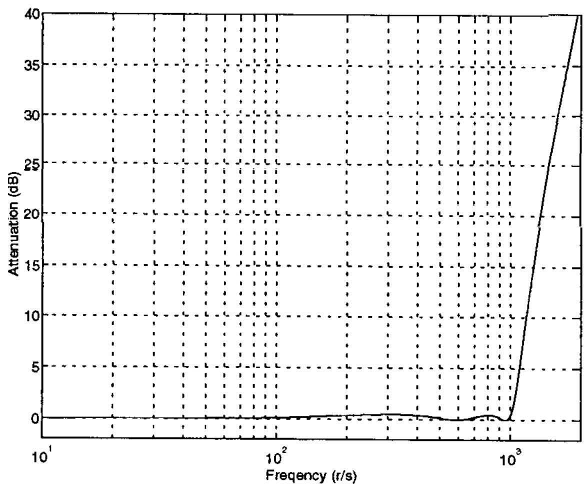

Figure 6: *SPICE Lowpass Filter Attenuation Simulation*

JERRY C. HAMANN

Jerry Hamann received the BS degree in Electrical Engineering/Bioengineering option from the University of Wyoming in 1984. After working as a product support engineer with Hewlett-Packard he returned to the University of Wyoming as a National Science Foundation Graduate Fellow to complete an MS in Electrical Engineering in 1988. In 1993 he completed the Ph.D. in Electrical Engineering at the University of Wisconsin-Madison where his research was focused in automatic control systems. He is currently an Assistant Professor in the Department of Electrical Engineering at the University of Wyoming. His professional interests include the modeling and analysis of uncertain dynamic systems, algorithms and software for simulation of dynamic systems, analysis and design of electronic filters, and applications of digital signal processing in real-time. He is a member of ASEE, IEEE, ISHM, Tau Beta Pi, Phi Kappa Phi and the Golden Key Honorary.