

COMPUTER INTERFACES FOR TEACHING THE NINTENDO® GENERATION

Thad B. Welch, Brian Jenkins

Department of Electrical Engineering
U.S. Naval Academy, MD

Cameron H. G. Wright

Department of Electrical Engineering
U.S. Air Force Academy, CO

1. Introduction

The utilization of the computer in the classroom is well documented and continues to grow in both availability and capability. The number of papers, e.g. (1-3) discussing the use of software such as MATLAB, as well as the number of textbooks, e.g. (4-8) that now offer companion disks or CD-ROMs clearly validates this statement. Where it is not at all uncommon for a Professor to require the use of commercially available software in a course or lab, the introduction of educational software designed to facilitate the understanding of a particular topic at the junior/senior level of the Electrical Engineering curriculum has not been as pervasive. For example, the use of MATLAB and its toolboxes has become commonplace at a large number of universities. MATLAB has allowed students to explore various aspects of the course material in greater depth, and Professors and textbook authors regularly provide additional MATLAB m-files to demonstrate a particular phenomenon or effect. The majority of these m-files are command line based and require more sophistication to glean the important lesson. In contrast, a well designed graphical user interface (GUI) driving this same m-file can largely reduce the need for user sophistication by *guiding* the student through what may have otherwise been a nearly infinite number of input parameter possibilities.

2. Discussion

This *guiding* of the students sounds appealing, but are we discouraging intellectual curiosity? Or even worse, are we entertaining instead of educating our students? To begin to address these questions, we turn to ASEE's PRISM magazine and rely on its longstanding commitment to articles on teaching.

Recent examples include:

From *What Is Good Teaching*⁹, "Good teaching involves five basic components",
"... instructional methods that facilitate student involvement, ..."

From *10 Habits of Highly Effective Teachers*¹⁰, “Motivate your student. Learning is far more efficient if the student is motivated and energized to put forth the effort necessary to learn a task. Incorporate as many motivating forces into your courses as you can.”

From *Grab student’s attention with Multimedia*¹¹, “By combining learning technologies such as visual simulation, web-based environments, and interactive materials, educators can deliver information in shorter and more intense cycles - the best method to grab and hold the attention of today’s demanding student.”

Or from *The Seven Laws of Teaching*¹², “Stimulate the pupil’s own mind to action. Keep his thought as much as possible ahead of your expression, placing him in the attitude of a discoverer, an anticipator.”

All of these references challenge us as educators to involve our students in the learning process. We strongly believe that a well designed GUI contributes significantly to the learning process.

Three examples which use GUI’s to engage the student in the learning process are described in this paper. The first (The Pole-Zero Plot) is used in both a *signals and systems* course and a *digital signal processing* course. The second (Passive Filters) is used in an *elementary circuit analysis* course. The third (The Effects of Filter Coefficient Quantization on Filter Performance) is used in a *digital signal processing* course.

2.1. The Pole-Zero Plot

Digital signal processing (DSP) using MATLAB is being taught at the undergraduate level all around the world. Even with the tremendous power of MATLAB, the significance of the pole/zero plot remains a mystery to many of our students. The ability to predict the effect of pole/zero location on either the magnitude and/or phase plot can be significantly enhanced using computer software. Software programs already exist that allow a student to calculate the magnitude and phase response associated with the arbitrary placement or movement of real and complex poles/zeros upon the complex plane. Unfortunately, these programs all have significant limitations. Our program adds several features that are either totally unavailable, or only partially available from these other programs. Specifically, we use the familiar MATLAB environment to add a GUI which allows for easy interactive pole/zero placement, relocation, and/or deletion. This includes both real and complex conjugate pairs of both poles and zeros. After a satisfactory pole/zero plot is constructed (e.g., the notched filter shown in Fig. 1), clicking on the *Plot mag/phase* button causes the magnitude and phase plots to be calculated and displayed in a separate figure (not shown). Rapid updates are possible and there is no need for a command line interface. Additionally, clicking the *Load/run DSK* button with the mouse downloads the calculated filter coefficients to an attached TMS320C31 DSK and executes this filter. With this program, the student now has a free, powerful, easy to learn and use software tool that allows for graphical filter design, hardware implementation, and exploration of the “what happens if ...” that we all desire for our students! This program¹³ has been used for both classroom demonstrations and filter design projects in both a Junior level *signals and systems* class as well as in a Senior level *digital signal processing* class.

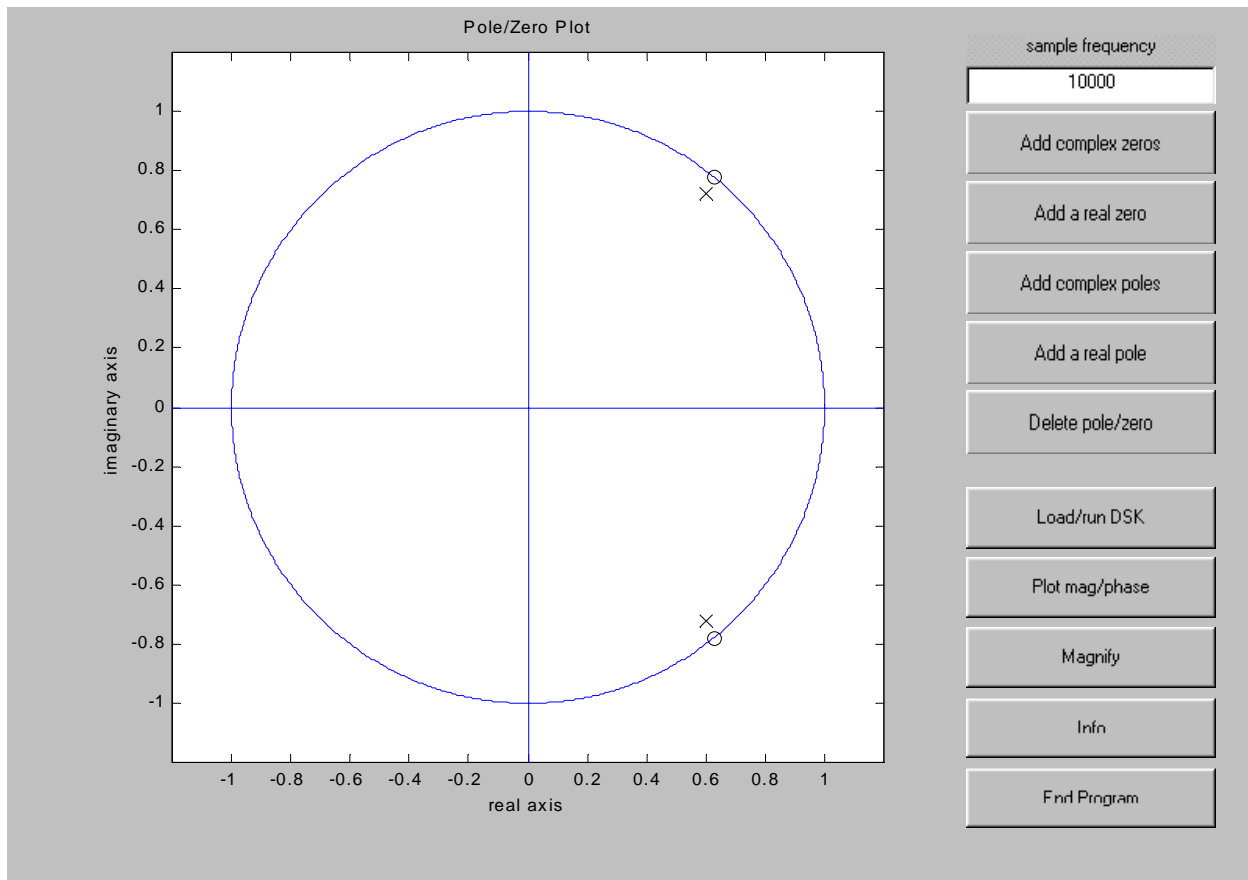


Fig. 1. Pole-Zero Design GUI.

2.2. Passive Filters

A GUI for use in an *elementary circuit analysis* course is illustrated in Fig. 2. The user interface implements a callback to an m-file in MATLAB that animates the magnitude and phase relationships between the voltages on components in a series resistor-capacitor (RC) network. The interface requires the student to enter the magnitude of the source voltage and the values of resistance R and capacitance C. During the animation, which is initiated by clicking the mouse on RUN, the time domain representation of each component voltage evolves as a function of the frequency f , where f varies logarithmically between $0.1f_c$ and $10f_c$ and the critical frequency $f_c = 1/(2\pi RC)$. This is similar to what a student would see in a laboratory by varying frequency on a function generator if the time base of an oscilloscope could be varied automatically, without human interaction. The three plots on the right illustrate the transfer function (magnitude and phase) in the frequency domain. The student may pause the simulation at any time by clicking on the left mouse button; animation is then resumed by pressing any key on the keyboard. The user interface provides a way for the student to learn a variety of concepts which often cause confusion at the beginning of the electrical engineering curriculum:

- 1) There is a direct correlation between the transfer function displayed in the plots on the right and the time domain representation displayed on the left, since the meaning of magnitude and phase at a given frequency is immediately observable as the animation progresses.
- 2) The filtering effects of the circuit, either as a low-pass or a high-pass filter, are easy to discern.
- 3) The magnitude plots enable a student to more easily understand the use of decibel (dB) notation.
- 4) The relationship between the component voltages at the critical frequency may be explored.
- 5) Kirchoff's Voltage Law is validated in the time domain.

Finally, the animation can be adapted for series resistor-inductor (RL) circuits, for various RLC networks, or for any systems which can be described easily using a transfer function.

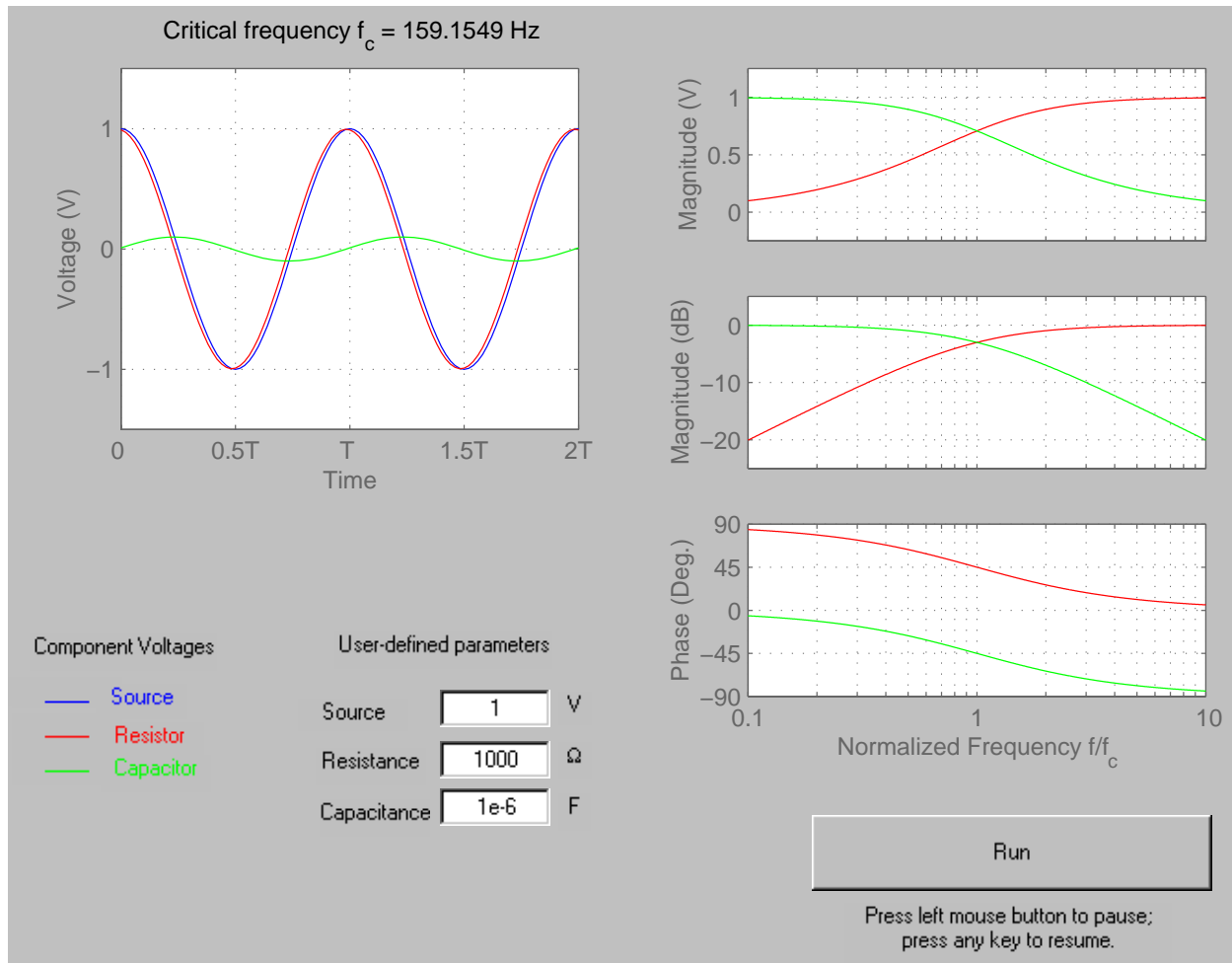


Fig. 2. Passive Filter GUI.

2.3. The Effects of Filter Coefficient Quantization on Filter Performance

The final GUI takes up where the MATLAB program `sptool` leaves off, adjusting the filter coefficients to simulate fixed-point hardware, allowing interactive analysis of the design effects, and seamlessly downloading the filter code to a TMS320C31 DSK when the user is ready. This allows the floating-point DSK to simulate a fixed-point device as desired (except for register and ALU effects, which are less of an issue at the introductory level), eliminating the need for buying fixed-point hardware just for this purpose. The program allows the student to interactively compare the theoretical filter performance with the real-world performance that would be encountered using any fixed-point DSP microprocessor, while still making full use of `sptool`. The actual performance of the student's filter design can be observed in real-time with the click of a mouse button, which loads and runs the filter on the TMS320C31 DSK. The program eliminates the need for the student to learn another software interface, manually program the DSK, and is perfectly suited to educational use. While it runs outside of `sptool`, the program^{14,15} easily exchanges information in both directions by using the same data structure format defined by `sptool`. The student can immediately see that with quantization effects, the filter performance can be altered radically.

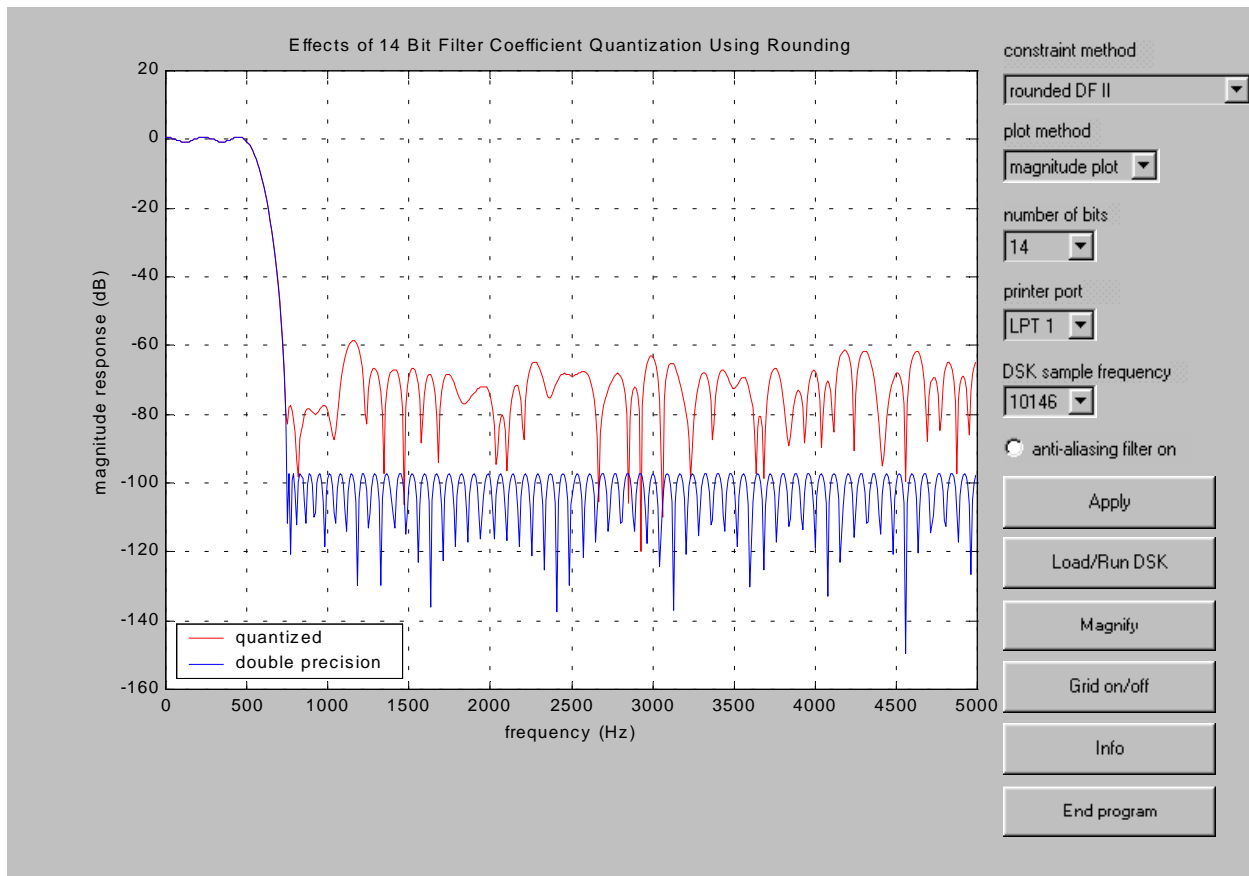


Fig. 3. Filter Coefficient Quantization GUI.

There are significant changes to the originally calculated magnitude (Fig. 3) response of the filter, which were predicated on the assumption of floating-point processing. When the filter design is satisfactory, the user can simply click the “Load/Run DSK” button on the GUI to download the software to the TMS320C31 DSK and run the filter algorithm for a real-time demonstration.

3. Conclusions

This *ease of use* can attract students to computer based educational experiences. The result can be profound as some students spend several hours exploring a phenomenon or effect that may receive only cursory interest using traditional techniques. We are exploring the use of GUI's in other areas of Electrical Engineering. This includes electromagnetics, fiber optics, and analog and digital communications. All of this software is freely available for downloading via the www site http://wseweb.ew.usna.edu/ee/LINKS/EE_Links.htm (should the URL be changed, from the Naval Academy home page, select Academics, Academic Divisions and Departments, Electrical Engineering, Links).

References

- [1] Kubichek, R. F., “Using MATLAB in a Speech and Signal Processing Class,” *Proceedings of the 1994 ASEE Annual Conference*, pp. 1207–1210, June 1994.
- [2] Burrus, C. S., “Teaching Filter Design Using MATLAB,” *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 20–30, April 1993.
- [3] Jacquot, R. G., Hamann, J. C., Pierre, J. W., and Kubichek, R. F., “Teaching Digital Filter Design Using Symbolic and Numeric Features of MATLAB,” *ASEE Computers in Education Journal*, vol. VII, no. 1, pp. 8–11, January-March 1997.
- [4] Porat, B., *A Course in Digital Signal Processing*, John Wiley & Sons, Inc., 1997.
- [5] Ingle, V. K., and Proakis, J. G., *Digital Signal Processing Using MATLAB V.4*, PWS Publishing, 1997.
- [6] Mitra, S. K., *Digital Signal Processing: A Computer-Based Approach*, McGraw-Hill, 1998.
- [7] Ambardar, A., and Borghesani, C., *Mastering DSP Concepts Using MATLAB*, Prentice-Hall, 1998.
- [8] McClellan, J. H., Burrus, C. S., Oppenheim, A. V., Parks, T. W., Schafer, R. W., and Schuessler, H. W., *Computer-Based Exercises for Signal Processing Using MATLAB 5*, Prentice-Hall, 1998.
- [9] Wankat, P., and Oreovicz, F., *What Is Good Teaching?*, ASEE PRISM Mag., pp. 16, September 1998.
- [10] Stice, J.E., *Habits Of Highly Effective Teaching*, ASEE PRISM Mag., pp. 28-31, November 1998.
- [11] Millard, D.L., *Grab Students' Attention With Multimedia*, ASEE PRISM Mag., pp. 26-29, December 1998.
- [12] Gregory, J.M., *The Seven Laws of Teaching*, Baker Books, 1997, (originally published in 1884).
- [13] Welch, T. B., Wright, C. H. G., and Morrow, M. G., “Poles, and Zeros, and MATLAB, Oh My!,” *Proceedings of the 1999 ASEE Annual Conference*, session 1320-02 [CD-ROM], June 1999.
- [14] Wright, C. H. G., Welch, T. B., Morrow, M. G., and Gomes III, W. J., “Teaching Real-World DSP Using MATLAB and the TMS320C31 DSK,” *Proceedings of the 1999 ASEE Annual Conference*, session 1320-06 [CD-ROM], June 1999.
- [15] Wright, C. H. G., Welch, T. B., Gomes III, W. J., and Morrow, M. G., “Teaching DSP Concepts Using MATLAB and the TMS320C31 DSK,” *Proceedings of the 1999 ICASSP Annual Conference*, session DSP E-1.3, vol. 6, pp. 3573-3576, March 1999.

Commander THAD B. WELCH, PhD, is an Assistant Professor in the Department of Electrical Engineering at the U.S. Naval Academy (from 1994–1997 he was an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy). His research interests include multicarrier communication systems analysis and signal processing. He is a member of ASEE, IEEE, and Eta Kappa Nu. Email: t.b.welch@ieee.org

BRIAN JENKINS received the B.S. and M.S. degrees from The Ohio State University in 1983 and 1991 and the PhD degree from the University of Colorado in 1995. From 1983-1989 he worked as a development engineer for IBM Corporation in Endicott, NY. He is an Assistant Professor in the Department of Electrical Engineering at the U.S. Naval Academy where he teaches and does research in fiber optic communications. Email: bjenkins@nadm.navy.mil

Major CAMERON H. G. WRIGHT, PhD, PE, is an Associate Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include signal and image processing, biomedical instrumentation, communications systems, and laser/electro-optics applications. He is a member of ASEE, IEEE, SPIE, NSPE, Tau Beta Pi, and Eta Kappa Nu. Email: c.h.g.wright@ieee.org