Motivating Engineering Students: Simulation versus Real-Time

Cameron H. G. Wright
Department of Electrical and Computer Engineering
University of Wyoming, WY

Thad B. Welch
Department of Electrical and Computer Engineering
Boise State University, ID

Michael G. Morrow
Department of Electrical and Computer Engineering
University of Wisconsin-Madison, WI

ABSTRACT

Digital signal processing has evolved over the last few decades from being primarily a specialized graduate course topic in Electrical and Computer Engineering to being a mainstream (often required) undergraduate course topic, at least in terms of the fundamentals. With digital signal processing becoming an integral part of an ever-wider scope of engineering solutions, other technical majors are also seeking out such courses. For many companies who hire Electrical and Computer Engineering graduates, competence in basic signal processing has become a "must have" skill for new hires.

The authors have spent many years practicing, reading, and writing about signal processing education. In that time, we have observed several common themes. This paper discusses the trade-off between using "canned" simulations versus the use of real-time processing on real-world signals as a motivator that can improve the educational outcome for signal processing students.

1 INTRODUCTION

Digital signal processing (DSP) has evolved over the last few decades from being a specialized topic to what is now considered one of the "must know" topics by most employers of new electrical and computer engineering (ECE) graduates. While DSP may be taught in various ways, it is generally agreed by engineering educators in the field that a solid understanding of many fundamental DSP topics is more likely when students are required to implement selected DSP algorithms in real-time, typically in a language such as C [1]. While non-real-time (i.e., off-line simulation of) algorithm implementations with tools such as MATLAB or LabVIEW are easier to include in courses, and require only a comparatively modest investment in hardware and software, our experience has shown there is significant benefit for students to include real-time DSP in the curriculum.

The authors have been deeply involved in signal processing education for well over 20 years. Over that time, we have written quite a few papers (at ICASSP, ASEE, and other venues) that have described our various lessons learned, discussed projects that worked well, and presented many DSP educational tools that we made freely available to educators [1–32]. In all of those years, our underlying and primary goal of maximizing student engagement and understanding has remained constant.

We believe that if students are truly interested in a topic, not only will they pursue the "required" course knowledge, but they will also regularly exceed our educational expectations. This belief has been confirmed over and over with the many students we've taught at multiple institutions (that span two military service academies and four civilian universities).

Most professors would agree that interactive learning, exercises, and demonstrations are invaluable for helping students understand a given concept [28,33–42]. We've come to believe that even more effective than demonstrations where students just sit passively and watch are activities with actual hands-on exercises and projects [1, 3, 4, 12, 19, 22, 29]. We even provide a widely-used text book (now in its third edition) and a website that support hands-on projects for real-time DSP [43, 44].

In our opinion, the *hook* that (metaphorically speaking) *reels* the students in is working with real-world signals, preferably on real-world hardware. This is what industry does, and a significant number of our students will go on to work for industry.

2 WHAT HARDWARE PLATFORM TO USE?

A question immediately come to mind: what hardware platform should be used? The hardware target of choice can vary and will naturally evolve as the associated technology continues to advance. With respect to computing for signal processing, we've seen the transition from mainframe offline computers, to specialized processor architectures that led to the first DSP chips, to floating-point processors, to field programmable gate arrays (FPGAs), to high performance computing (HPC) or multicore signal processors. And we've seen the reinvigoration of the hobbyist (or "maker space" movement) with low-cost boards such as Arduino and Raspberry Pi and even implementing signal processing algorithms on smartphones on one end, to the other extreme of power hungry, massively parallel, arrays of graphical processing units (GPUs). This list is by no means complete, and even if we cataloged all of the major advances in signal processing computing, the next year or the next decade will show us something new and perhaps revolutionary. This constant advancement is what keeps industry going, and it also keeps a great number of engineers employed and keeps students applying for admission to our Colleges of Engineering.

Years ago, with the widespread adoption of personal computers into our classrooms and teaching labs, simulation took over and most students back then were amazed by the presentations that could be provided. We believe that those days are now gone; they are probably at least a decade in the past and today's students are not at all impressed or enthralled with just simulation. The only possible exception to this is if the simulation involves a video, but today's students are at best simply tolerant of these "canned" simulation presentations. However, hands-on demonstrations where they get to work with real-world signals still excite most students.

Working with real-world signals in an interactive, hands-on manner implies real-time DSP. While real-time DSP can be performed to some degree using low-cost boards such as Arduino [45–47] or Raspberry Pi [48–50], we also find that more specialized real-world hardware adds to the power and efficacy of the demonstration. By real-world hardware, we mean hardware (and the associated development tools) that would most likely be used to develop a commercial product in industry when the student becomes a working engineer. This tends to make Arduino, Raspberry Pi, and similar low-cost solutions less attractive.

The authors have used many platforms over the years, with the majority being based on one of the commercial-level DSP chips from Texas Instruments (known as the TMS320x family of chips). We have used the C31, C6201, C6211, C6711, C6713, and most recently, we have been using the OMAP-L138, which incorporates both a C6748 DSP core and an ARM-9 core. The Texas Instruments hardware is programmed by using the professional-grade integrated development environment (IDE) known as Code Composer Studio™, which the students seem to appreciate. There are certainly other hardware and software options, but this is what we have been using.

3 WHAT HANDS-ON DEMONSTRATIONS TO SHOW?

There are many possibilities for hands-on demonstrations using real-world signals and industry-grade real-time DSP hardware. For example, we have engaged our students in hands-on real-time demonstrations that include

- sampling and quantization;
- digital filter design, implementation, and testing;
- unstable system implementation;
- phase response and group delay;
- modulation and demodulation;
- beam forming, using inexpensive microphones;
- biomedical signal conditioning;

- encoding and decoding Caller-ID signals;
- control of high-amperage external devices;
- sampling rate conversion;
- signal processing of financial data;
- software defined radio;
- adaptive filters;
- video signal manipulation;
- multispectral imaging;
- and the basics of optical engineering.

Many more topics await; the possibilities are nearly endless.

No one realistically expects every ECE or EE professor to become an expert in the hardware and software tools for real-time DSP, but that expertise isn't necessary to be able to incorporate such demonstrations into their courses. The authors have regularly provided tools, freely available to educators, that make it relatively simple to get students involved with real-time hands-on demonstrations. For example, the original winDSK tool, and it's follow-ons winDSK6 and now winDSK8, allow any professor to perform real-time demonstrations with little effort [5,17,24,31].

The latest version, winDSK8, works with the OMAP-L138 multicore system-on-a-chip and the single core C6713. The winDSK8 main user interface is shown in Fig. 1. This tool provides a wide array of capabilities that allow an instructor to demonstrate DSP concepts to a class using real-time signals, or allow the students to use it on their own computer for either directed or self-directed hands-on exercises. The main demonstration areas of winDSK8 are Audio Demo Apps and two sections of Filters/Communications. See Fig. 2 for the Audio Effects user interface.

Hands-on demonstrations using real-time DSP for digital communications systems and software defined radio are somewhat complex; to better handle such concepts winDSK8 includes separate sub-interfaces called CommDSK (for modulation schemes that change the phase of the carrier such as BPSK, QPSK, QAM) and CommFSK (for modulation schemes that change the frequency of the carrier such as FSK) to support them. For example, see Fig. 3 for CommDSK and Fig. 4 for CommFSK.

4 RESULTS OF USING REAL-TIME DEMONSTRATIONS

A brief review of the results using some of the more recent demonstration topics listed above may be helpful to professors seeking to incorporate real-time hands-on demonstrations into their courses. The reference list of this paper provides more detailed resources.

At ICASSP 2013, we discussed the use of adaptive filters as the demonstration vehicle for our students' real-time hands-on exercises [27]. The majority of our students agreed with our belief that the adaptive filter demonstration helped them understand the underlying signal processing

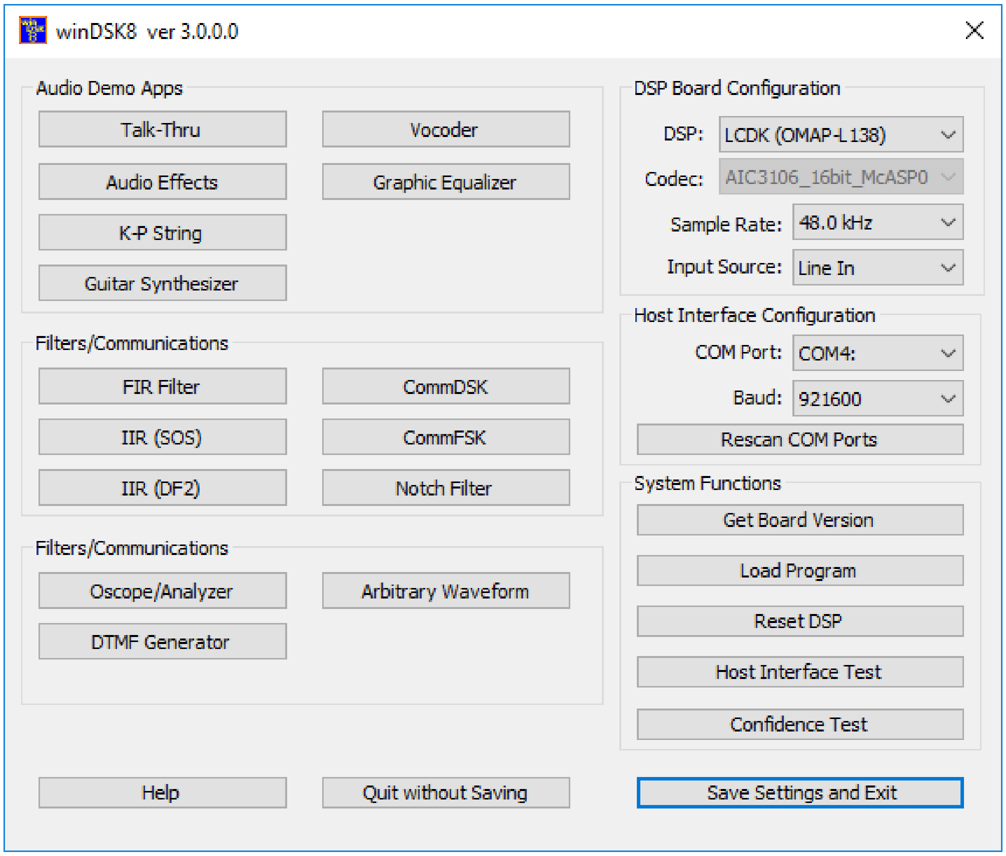

Figure 1: The main winDSK8 user interface.

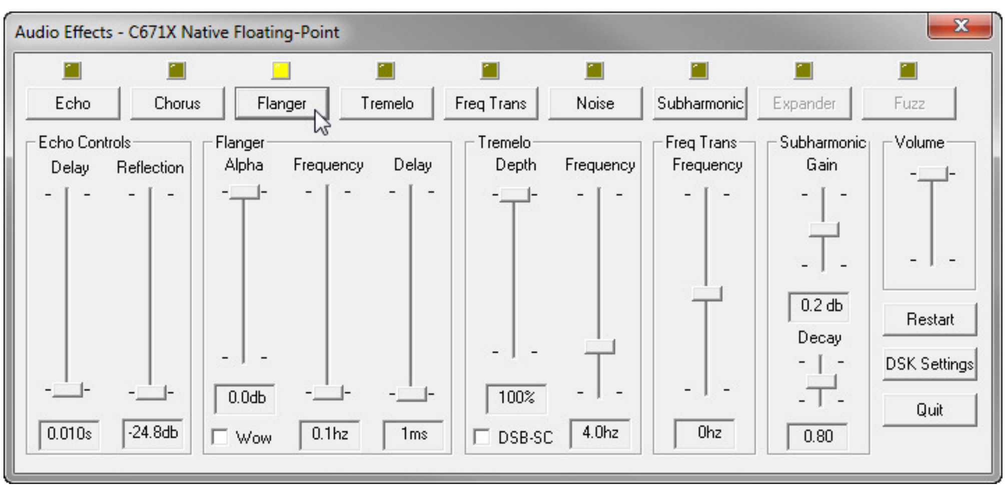

Figure 2: The user interface for Audio Effects.

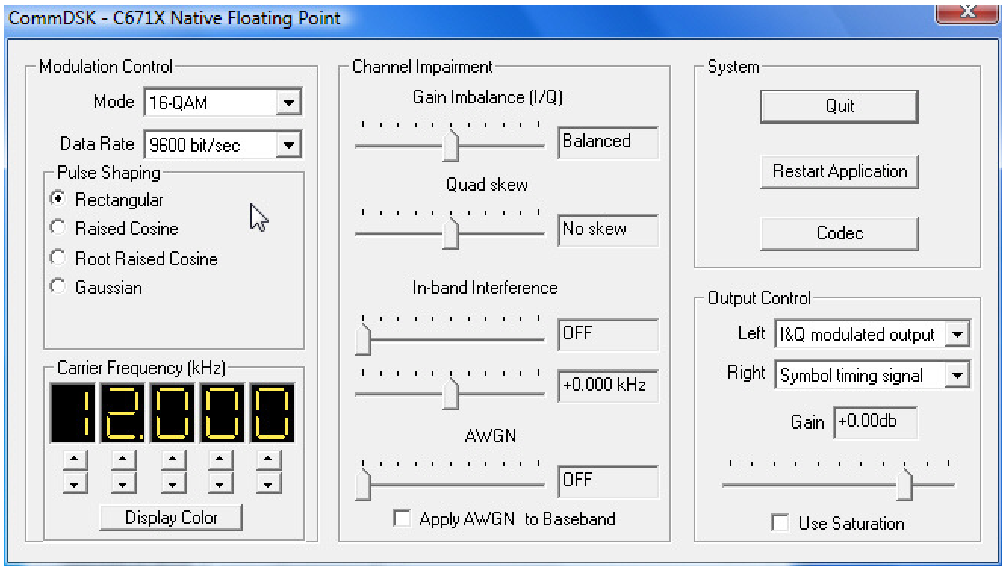

Figure 3: The user interface for CommDSK.

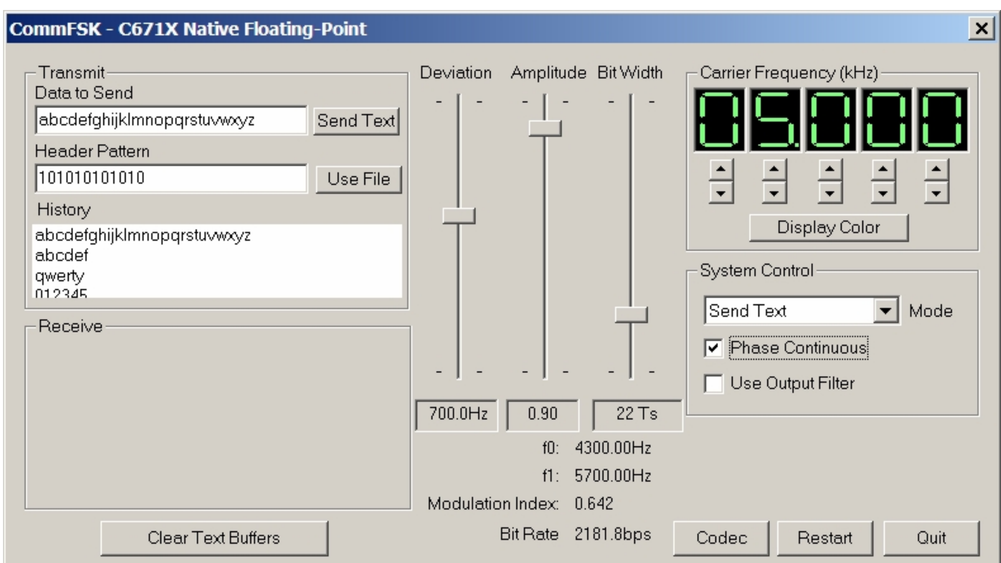

Figure 4: The user interface for CommFSK.

concepts, despite the fact that only a single class period was spent on this somewhat hard-to-approach topic. A survey question designed to assess the efficacy of DSP demonstrations in general averaged a score of 4.29 on a 5-point Likert scale, where 5 is most effective. This certainly is in line with our anecdotal experiences regarding the teaching effectiveness of demonstrations in the classroom.

At ICASSP 2014, we described using a video see-through application to help students understand not only signal processing concepts, but also the basics of video signal formatting and timing [28]. Using real-time digital signal processing rather than the less challenging technique of using off-line processing (such as with MATLAB) greatly increased the effectiveness of the exercise. While "see-through" is a simple process on first glance, we find that students must establish confidence in the hardware and software platform before any significant learning can begin. In the audio realm, a "talk-through" project accomplishes this. For moving on to the more complicated video signal, we used a see-through project.

This point really deserves more discussion, as many professors consider such a project a waste of time. Using a bare-bones first project, such as talk-through or see-through, to build student confidence in the real-time platform is a valuable pedagogical approach. We have found that by skipping this step, the instructor greatly reduces student motivation to pursue the real cause of incorrect results from a real-time DSP exercise. That is, students can often be quick to "blame" the platform rather than their own mistakes. Establishing a working baseline in each student's mind, using talk-through or see-through, is definitely worth the time.

At ICASSP 2016, we discussed how signal processing can be used to segue into related topics, such as optical engineering [30]. The daunting challenge of teaching practical optical engineering in a single course can be greatly reduced by taking advantage of existing student knowledge of signal processing concepts. In general, for students learning a new topic area, a particularly efficient and effective method takes best advantage of the students' prior knowledge. By making links to their existing cognitive frameworks, our students were far less intimidated by the new subject material, and more quickly mastered an acceptable level of expertise, compared to a "starting from scratch" approach to a new topic area.

At ICASSP 2018, we discussed how the latest generation of affordable infrared (IR) cameras can be used for an interesting exercise in DSP for the students. In particular, we use such tools for introducing the concept of multispectral signal processing [32]. These IR cameras provide both IR and visible images, hence the multispectral aspect. We described the use of a FLIR E60 IR camera in a graduate digital image processing course, in which both IR and visible images were used as the basis for an open-ended final project in the course that required them to design and implement an image fusion algorithm. By using both pre- and post-project questionnaires, we were able to confirm that the project was a positive experience for the students, and helped motivate them to master the course material.

5 CONCLUSIONS

Modern industry has a global need for engineers who are literate, confident, and competent in DSP methods and techniques. An obvious solution for supplying this need is for universities to continue to educate new engineers in the fundamental concepts, with an emphasis on practical skills. To help educate the next generation of DSP engineers, several powerful and highly versatile options are available. Real-time or quasi real-time systems can be implemented using, for example, microprocessors, field programmable gate arrays, dedicated DSP hardware, general-purpose processors, graphical processing units, or even smart phones. In this paper, we discussed our efforts in using a number of different dedicated DSP hardware devices to enhance our students' understanding of signal processing concepts and to develop a hardware and software skill-set that is routinely sought by future employers. We also laid out the case that real-time, hands-on exercises or demonstrations, using real-world signals, is still one of the most motivating methods to use with today's sometimes jaded students.

We strongly encourage any faculty member who teaches DSP to incorporate demonstrations and hands-on experience with real-time hardware for their students. We have made a wide variety of resources easily available to help in this endeavor, and they are freely available to educators.

REFERENCES

[1] C. H. G. Wright, T. B. Welch, D. M. Etter, and M. G. Morrow, "Teaching DSP: Bridging the gap from theory to real-time hardware," *ASEE Comput. Educ. J.*, pp. 14–26, July–September 2003.

[2] C. H. G. Wright and T. B. Welch, "Teaching DSP concepts using MATLAB and the TMS320C31 DSK," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 3573–3576, Mar. 1999.

[3] C. H. G. Wright and T. B. Welch, "Teaching real-world DSP using MATLAB," *ASEE Comput. Educ. J.*, pp. 1–5, January–March 1999.

[4] T. B. Welch, M. G. Morrow, and C. H. G. Wright, "Teaching practical hands-on DSP with MATLAB and the C31 DSK," in *Proceedings of the 2000 ASEE Annual Conference*, June 2000. Paper 1320-03.

[5] M. G. Morrow and T. B. Welch, "winDSK: A windows-based DSP demonstration and debugging program," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 3510–3513, June 2000. (invited).

[6] T. B. Welch, C. H. G. Wright, and M. G. Morrow, "Poles and zeroes and MATLAB, oh my!," *ASEE Comput. Educ. J.*, pp. 70–72, April–June 2000.

[7] M. G. Morrow, T. B. Welch, C. H. G. Wright, and G. W. P. York, "Demonstration platform for real-time beamforming," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 2693–2696, May 2001.

[8] T. B. Welch, M. G. Morrow, and C. H. G. Wright, "Teaching practical hands-on DSP with MATLAB and the C31 DSK," *ASEE Comput. Educ. J.*, pp. 13–20, April–June 2001.

[9] C. H. G. Wright, T. B. Welch, D. M. Etter, and M. G. Morrow, "A systematic model for teaching DSP," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. IV, pp. 4140–4143, May 2002. Paper 3243.

[10] C. H. G. Wright, T. B. Welch, D. M. Etter, and M. G. Morrow, "Teaching hardware-based DSP: Theory to practice," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 4148–4151, May 2002.

[11] G. W. P. York, C. H. G. Wright, M. G. Morrow, and T. B. Welch, "Teaching real-time sonar with the C6711 DSK and MATLAB," *ASEE Comput. Educ. J.*, pp. 79–87, July–September 2002.

[12] T. B. Welch, R. W. Ives, M. G. Morrow, and C. H. G. Wright, "Using DSP hardware to teach modem design and analysis techniques," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. III, pp. 769–772, Apr. 2003.

[13] T. B. Welch, M. G. Morrow, and C. H. G. Wright, "Using DSP hardware to control your world," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. V, pp. 1041–1044, May 2004. Paper 1146.

[14] T. B. Welch, M. G. Morrow, C. H. G. Wright, and R. W. Ives, "commDSK: a tool for teaching modem design and analysis," *ASEE Comput. Educ. J.*, pp. 82–89, April–June 2004.

[15] K. E. Wage, J. R. Buck, C. H. G. Wright, and T. B. Welch, "The signals and systems concept inventory," *IEEE Trans. Educ.*, vol. 48, pp. 448–461, Aug. 2005.

[16] T. B. Welch, C. H. G. Wright, and M. G. Morrow, "Caller ID: An opportunity to teach DSP-based demodulation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. V, pp. 569–572, Mar. 2005. Paper 2887.

[17] M. G. Morrow, T. B. Welch, and C. H. G. Wright, "A host port interface board to enhance the TMS320C6713 DSK," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. II, pp. 969–972, May 2006.

[18] T. B. Welch, C. H. G. Wright, and M. G. Morrow, "Teaching rate conversion using hardware-based DSP," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. III, pp. 717–720, Apr. 2007.

[19] C. H. G. Wright, M. G. Morrow, M. C. Allie, and T. B. Welch, "Using real-time DSP to enhance student retention and engineering outreach efforts," *ASEE Comput. Educ. J.*, pp. 64–73, October–December 2008.

[20] C. H. G. Wright, M. G. Morrow, M. C. Allie, and T. B. Welch, "Enhancing engineering education and outreach using real-time DSP," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. III, Apr. 2008.

[21] T. B. Welch, T. Kent, C. H. G. Wright, and M. G. Morrow, "Teaching with software defined radios," in *Proceedings of the 2009 ASEE Annual Conference*, June 2009.

[22] T. B. Welch, C. H. G. Wright, and M. G. Morrow, "The DSP of money," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2309–2312, Apr. 2009.

[23] T. B. Welch, C. H. G. Wright, and M. G. Morrow, "Software defined radio: inexpensive hardware and software tools," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2934–2937, Mar. 2010.

[24] M. G. Morrow, C. H. G. Wright, and T. B. Welch, "winDSK8: A user interface for the OMAP-L138 DSP board," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2884–2887, May 2011.

[25] M. G. Morrow, C. H. G. Wright, and T. B. Welch, "Old tricks for a new dog: An innovative software tool for teaching real-time DSP on a new hardware platform," *ASEE Comput. Educ. J.*, pp. 64–69, October–December 2011.

[26] T. B. Welch, M. G. Morrow, and C. H. G. Wright, "Connexions and the SPEN fellows program," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2785–2788, Mar. 2012. paper 3664.

[27] M. G. Morrow, C. H. G. Wright, and T. B. Welch, "Real-time DSP for adaptive filters: A teaching opportunity," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2013.

[28] A. Rothenbuhler, C. H. G. Wright, T. B. Welch, and M. G. Morrow, "DSP see-through: Going beyond talk-through," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2247–2251, May 2014.

[29] C. H. G. Wright, T. B. Welch, and M. G. Morrow, "Leveraging student knowledge of DSP for optical engineering," in *Proceedings of the 2015 IEEE Signal Processing and Signal Processing Education Workshop*, pp. 148–153, Aug. 2015.

[30] C. H. G. Wright, T. B. Welch, and M. G. Morrow, "Signal processing concepts help teach optical engineering," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 6275–6279, Mar. 2016.

[31] C. H. G. Wright, T. B. Welch, and M. G. Morrow, "Reinforcing signal processing theory using real-time hardware," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 6329–6333, Mar. 2017.

[32] C. H. G. Wright and T. B. Welch, "Image fusion: an introduction to multispectral signal processing," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 7001–7005, Mar. 2018. Paper: SS-L16.3.

[33] C. S. Burrus, "Teaching filter design using MATLAB," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 20–30, Apr. 1993.

[34] R. F. Kubichek, “Using MATLAB in a speech and signal processing class,” in *Proceedings of the 1994 ASEE Annual Conference*, pp. 1207–1210, June 1994.

[35] R. G. Jacquot, J. C. Hamann, J. W. Pierre, and R. F. Kubichek, “Teaching digital filter design using symbolic and numeric features of MATLAB,” *ASEE Comput. Educ. J.*, pp. 8–11, January–March 1997.

[36] J. H. McClellan, C. S. Burrus, A. V. Oppenheim, T. W. Parks, R. W. Schafer, and S. W. Schuessler, *Computer-Based Exercises for Signal Processing Using* MATLAB 5. MATLAB Curriculum Series, Upper Saddle River, NJ (USA): Prentice Hall, 1998.

[37] J. W. Pierre, R. F. Kubichek, and J. C. Hamann, “Reinforcing the understanding of signal processing concepts using audio exercises,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 3577–3580, Mar. 1999.

[38] W.-S. Gan, Y.-K. Chong, W. Gong, and W.-T. Tan, “Rapid prototyping system for teaching real-time digital signal processing,” *IEEE Trans. Educ.*, vol. 43, no. 1, pp. 19–24, 2000.

[39] T. S. Hall and D. V. Anderson, “A framework for teaching real-time digital signal processing with field-programmable gate arrays,” *IEEE Trans. Educ.*, vol. 48, no. 3, pp. pp. 551–558, 2005.

[40] A. Spanias, K. Huang, R. Ferzli, H. Kwon, V. Atti, V. Berisha, L. Iasemides, H. Krishnamoorthi, M. B. S. Misra, K. Tsakalis, and S. Haag, “Interfacing Java-DSP with a TI DSK board,” *ASEE Comput. Educ. J.*, vol. XVII, Jul.–Sep. 2007.

[41] H. Kwon, V. Berisha, A. Spanias, and V. Atti, “Experiments with sensor motes and Java-DSP,” *IEEE Trans. Educ.*, vol. 52, no. 2, pp. 257–262, 2009.

[42] C. H. G. Wright, T. B. Welch, and M. G. Morrow, “Using student knowledge of linear systems theory to facilitate the learning of optical engineering,” in *Proceedings of the 2015 ASEE Annual Conference*, pp. 26.1683.1–26.1683.11, June 2015. DOI: 10.18260/p.25019.

[43] T. B. Welch, C. H. G. Wright, and M. G. Morrow, *Real-Time Digital Signal Processing: From MATLAB to C with C6x DSPs*. Boca Raton, FL (USA): CRC Press, 3rd ed., 2017.

[44] “RT-DSP website.” `http://www.rt-dsp.com`.

[45] W. J. Esposito, F. A. Mujica, D. G. Garcia, and G. T. Kovacs, “The Lab-In-A-Box project: An Arduino compatible signals and electronics teaching system,” in *Proceedings of the 2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE)*, pp. 301–306, Aug. 2015. doi: 10.1109/DSP-SPE.2015.7369570.

[46] M. A. Wickert, “Real-time DSP basics using Arduino and the Analog Shield SAR codec board,” in *Proceedings of the 2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE)*, pp. 59–64, Aug. 2015. doi: 10.1109/DSP-SPE.2015.7369528.

[47] J. Jaldén, X. C. Moreno, and I. Skog, “Using the Arduino Due for teaching digital signal processing,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 6468–6472, 2018. doi: 10.1109/ICASSP.2018.8461781.

[48] M. S. Pattichis, S. Celedon-Pattichis, and C. Lopez-Leiva, "Teaching image and video processing using middle-school mathematics and the Raspberry Pi," in *Proceesings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6349–6353, Mar. 2017. doi: 10.1109/ICASSP.2017.7953378.

[49] J. Marot and S. Bourennane, "Raspberry Pi for image processing education," in *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, Aug. 2017. doi: 10.23919/EUSIPCO.2017.8081633.

[50] M. Azarpour, J. Siska, and G. Enzner, "Realtime binaural speech enhancement demo on Raspberry Pi," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6572–6573, Mar. 2017. doi: 10.1109/ICASSP.2017.8005296.