

Work in Progress: A New Approach to Student Learning of Real-time Biomedical Digital Signal Processing

Dr. Willis J. Tompkins, University of Wisconsin-Madison

Willis J. Tompkins received the B.S. and M.S. degrees in electrical engineering from the University of Maine at Orono in 1963 and 1965, respectively, and the Ph.D. degree in biomedical electronic engineering from the University of Pennsylvania in 1973. He is currently Professor Emeritus of Biomedical Engineering at the University of Wisconsin-Madison, where he was on the faculty from 1974 until he retired in 2014. He previously served for five years as Chair of the Department of Electrical and Computer Engineering. His teaching specialty is on the topic of computers in medicine, an area in which he has developed two courses. One of these two courses, he has evolved and taught for 40 consecutive years. He has received a number teaching awards including the University of Wisconsin Chancellor's Award for Excellence in Teaching and the Theo C. Pilkington Outstanding Educator Award from the Biomedical Engineering Division of the ASEE. His research interests include development of microprocessor-based medical instrumentation, on-line biomedical computing, and real-time computer processing of electrocardiograms. Dr. Tompkins is a Life Fellow of the IEEE (Institute of Electrical and Electronics Engineers), a Founding Fellow of the AIMBE (American Institute for Medical and Biological Engineering), and an Inaugural Fellow of BMES (Biomedical Engineering Society). He is a past President of the IEEE EMBS (Engineering in Medicine and Biology Society) and a past Chair of the ASEE Biomedical Engineering Division.

Dr. Amit Janardhan Nimunkar, University of Wisconsin, Madison

Amit J Nimunkar received his B.E. in Electronics Engineering from the University of Mumbai, India in 1999, M.S. in Bioengineering from the University of Toledo, Ohio in 2000 and Ph.D. in Biomedical Engineering from the University of Wisconsin-Madison, Wisconsin in 2009. He is currently the Associate Faculty Associate in Biomedical Engineering at the University of Wisconsin-Madison. His teaching specialty is on the topic of Biomedical Engineering Design and Bioinstrumentation and has taken initiative to develop hands-on blended learning based courses on the same topics. His research interest is on global health and engineering and currently working on projects in Honduras, Ethiopia, India and Vietnam. He has received the Recognition Award for Achievement in Global Engaged Scholarship in 2013 through the Wisconsin Without Borders at the University of Wisconsin-Madison, the Professor of the Year Award in 2012, through the Biomedical Engineering Society at the University of Wisconsin-Madison, and a number of teaching awards.

A new approach to student learning of real-time biomedical digital signal processing

Abstract

We have revised the labs in our biomedical digital signal processing course to include real-time digital filter implementation using a single board microcontroller. In the past, students only experienced the filter design process in lab using a unique MATLAB program we developed that provides a tool for designing digital filters and visualizing the filter results on a database of pre-sampled electrocardiogram (ECG) signals. After designing the filters and obtaining their coefficients with this program, in the new labs, students then write C/C++ programs to implement the filters on a powerful single-board microcontroller and process ECG signals in real-time. They then analyze the results of their filter implementations using a real-time ECG simulator as well as their own live ECG signals as inputs.

Introduction

In a typical biomedical engineering digital signal processing course, students have limited real-world design opportunity. Typically, they implement algorithms and analyze signals with powerful software like MATLAB, but have limited chance to design and run signal processing algorithms in real time on a microprocessor. This approach limits student understanding of constraints in designing and implementing digital filters and evaluating their actual performance characteristics.

For a number of years, we have been using a program called UW DigiScope that we developed for the lab experiments in our course on biomedical digital signal processing. This is a MATLAB digital oscilloscope program that gives the user a range of basic functions typical of a digital oscilloscope such as display of waveforms and adjustment of sensitivity controls. More importantly, the graphical user interface (GUI) of this program also provides a digital signal processing package with a comprehensive set of libraries that facilitate interactively implementing finite impulse response (FIR) and infinite impulse response (IIR) filters and quickly visualizing their results on a signal. The filter design library includes impulse response, pole-zero, amplitude, and phase plots to assist the student in the design process. Included with DigiScope are a library of sampled ECG signals from the PhysioNet ECG database [1]. Once a filter is designed in DigiScope the processing results can be visualized on the included database of electrocardiograms.

We included concepts of real-time digital signal processing in a book developed for this course some time ago [2]. In the Spring semester of 2018, we added lab experiments with real-time microcontroller programming of the signal processing algorithms in addition to analysis of pre-recorded signals with the DigiScope program. After an acceptable filter is designed, DigiScope provides the coefficients of the designed filter that the students use to implement the filter as a real-time digital filter on the microcontroller platform by programming it in C/C++.

Our approach is different from the off-line approaches taught in most digital signal processing courses. We expect that the use of real-time signal processing with microcontrollers will greatly

enhance the students' hands-on design experience. Our new set of 10 labs provides a mechanism for anyone interested in biomedical signal processing to study the field without requiring any other resources except a PC, MATLAB, an oscilloscope, an inexpensive, high-performance microcontroller board, and a free on-line C/C++ compiler.

Nucleo-32 Microcontroller Board

The platform that we chose for students to program real-time signal processing algorithms is the Nucleo-32 (STM32F303K8) microcontroller board [3]. This board is based on a 32-bit ARM-Cortex-M microprocessor and is one of a family of commercial products from multiple companies that are called *mbed* or *mbed-enabled* and use the same set of software development tools. This platform also includes online software development tools including a C/C++ compiler, cloud services, and a developer ecosystem [4]. In addition, the *mbed* platform is supported by an excellent reference textbook [5].

We also chose the Nucleo-32 development board because it has a powerful general-purpose RISC processor with a floating-point unit (FPU) and up to nine analog I/O channels (ADC and DAC) suitable for implementing real-time digital signal processing algorithms for biomedical signals like the ECG, electroencephalogram (EEG) and electromyogram (EMG). Each student is required to purchase this microcontroller board (about \$11) for the lab experiments as part of a kit of parts (about \$50) including a USB cable to connect to a computer, a breadboard, hook-up wires, and components to build an ECG amplifier circuit [6]. There are no additional costs to the students since the textbook and references are available on-line for free.

Laboratory Experiments

The ECG is the model biomedical signal that we chose for experimenting with digital signal processing techniques. Following introduction of each signal processing algorithm throughout the course is a DigiScope-based prelab that the students complete before they complete the in-lab microcontroller-based programming that gives insight into the performance and potential pitfalls of each algorithm. By the end of the course, students implement a number of FIR and IIR filters as well as a variety of other signal processing techniques and use them to analyze ECG signals.

At the beginning of the course, an ECG laboratory teaches about analog signal acquisition and preprocessing by having each student build circuitry on a breadboard for amplifying his/her own ECG. Students use this ECG amplifier circuit throughout the lab course to provide live ECG signals to the input of the microcontroller as one test of their filter designs.

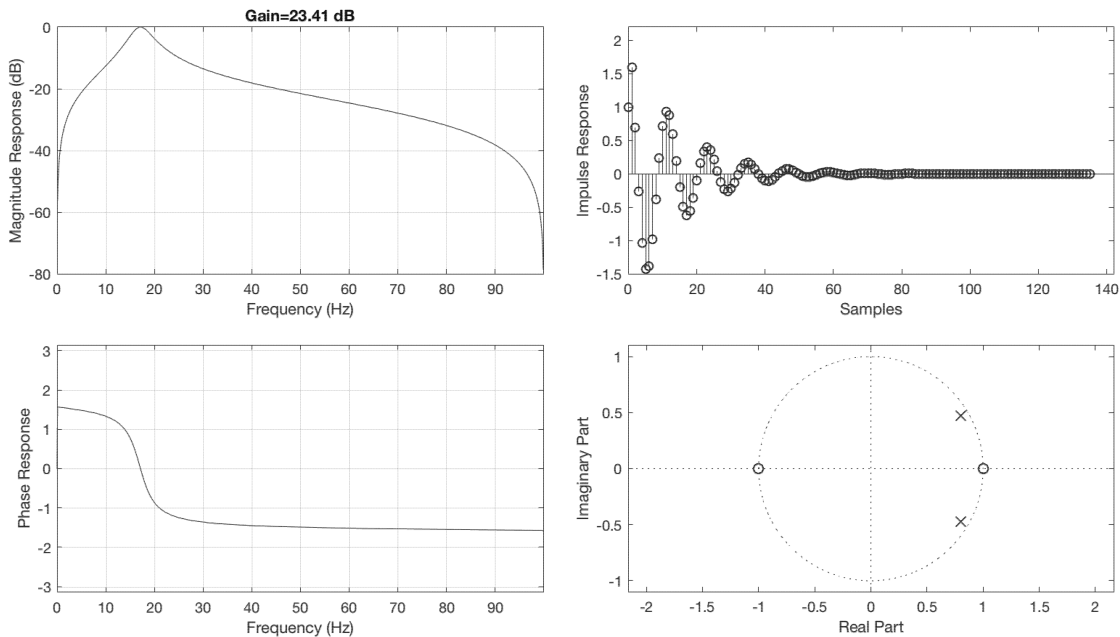
Students individually complete all the course prelabs and work individually on the first five in-lab experiences. For the final five labs, they organize into two-student teams. For developing and testing the filter algorithms, one student's Nucleo runs an ECG simulator program we developed that provides a real-time ECG signal to the other team member's Nucleo that the team programs with their digital filter algorithm.

The ECG simulator program provides realistic signals based on waveform templates sampled from real ECGs. Through keyboard input, students can specify the heart rate, type of abnormality

(e.g., periodic PVC's), type and amplitude of noise added to the signal including 60-Hz, EMG (random noise), and baseline drift. As the labs proceed, the students learn the basic programming strategies for implementing different types of FIR and IIR filters for ECG processing. Ultimately, the students implement a cascade of filters and other signal processing strategies on the Nucleo microcontroller to achieve reliable real-time QRS detection based on the Pan-Tompkins algorithm [7].

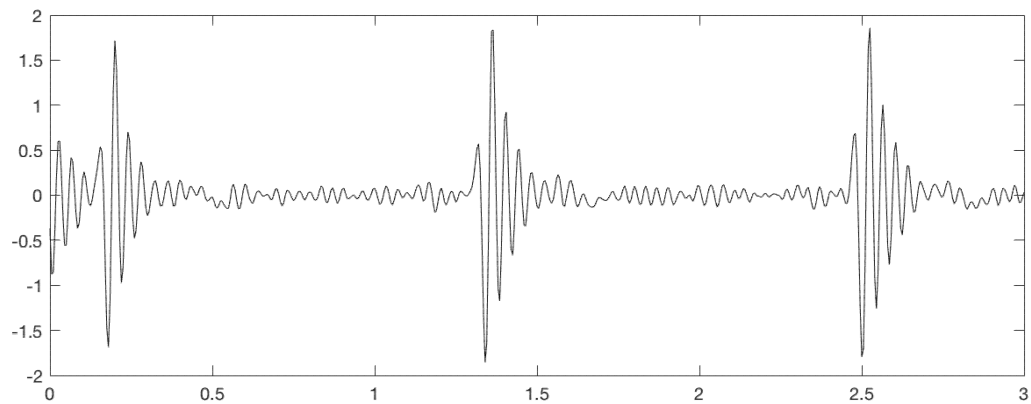
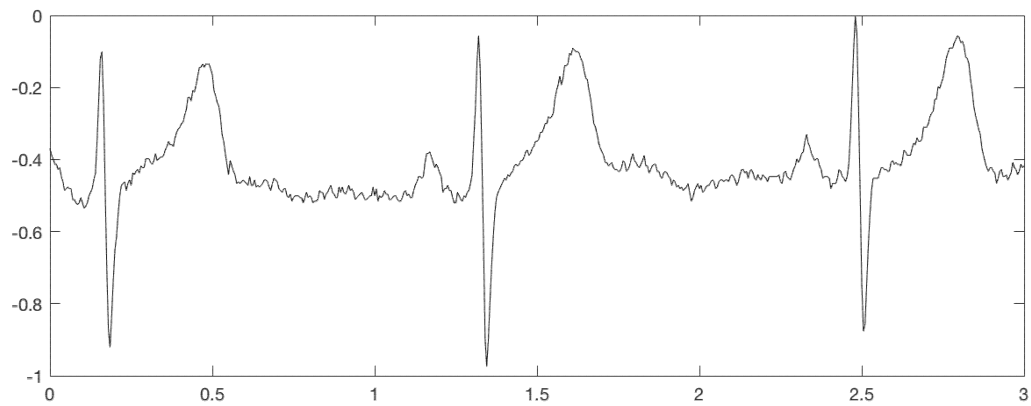
For each lab, student teams further test their algorithms on their own live ECG's using the ECG amplifier that they built on a breadboard in the first lab. In this way, they are able to see the real-world problems like motion artifacts, 60-Hz noise, and EMG noise that challenge the best algorithms when analyzing real-time ECG's.

For a final laboratory experience in the course, students choose a project from a list provided by the instructors. They work individually or in teams of up to three students to complete a real-time signal processing program using the microcontroller platform and present their results to the rest of the class.



$$H(z) = \frac{1 - z^{-2}}{1 - 1.6z^{-1} + 0.865z^{-2}}$$

(a)



(b)



(c)

Figure. Design and real-time implementation of an IIR filter used as a preprocessor for a cardiometer. (a) DigiScope design panel showing the amplitude and phase responses, impulse response, pole-zero plot, and transfer function for a student-designed filter. (b) DigiScope signals panel showing an ECG input signal in the top trace and the output of the IIR filter in the bottom trace. (c) Oscilloscope screenshot showing real-time implementation of filter using the Nucleo microcontroller. Top trace is a normal ECG generated by the Nucleo ECG simulator program. Bottom trace is the IIR filter output.

Assessment

The following are the course learning outcomes. Students will be able to:

1. Design and analyze a variety of FIR and IIR digital filters using a MATLAB program called UW DigiScope specially designed for this purpose.
2. Implement these filters on a microcontroller platform for real-time processing of biomedical signals including electrocardiograms using the C/C++ language.
3. Write real-time software to find significant features in biomedical signals including their own live ECGs from an ECG amplifier that they breadboarded in the course.
4. Write software to implement specialized signal processing techniques such as time epoch signal averaging.

In this new version of the course, students learn in a blended way by first doing off-line DigiScope design and analysis prior to coming to the lab followed by the in-lab programming of the algorithms for real-time signal processing using a microcontroller. We plan to assess whether this new approach for learning has helped the students to better understand the material through a concept inventory and their ability to do independent design projects.

At the beginning of the course, we will assess the knowledge and skills of the students by administering a concept inventory. This inventory consists of multiple choice questions that focus on fundamental concepts taught in previous versions of the course. Since the student population is a mixture of ECE and BME majors that have some differences in prerequisite backgrounds, the inventory is likely to discover some knowledge weaknesses in the student body as well as document the lack of knowledge of some of the concepts that we plan to teach. If the majority of students exhibit weakness in a particular concept that we view as a critical prerequisite, we will cover it in the class and/or laboratory. If a minority of the class shows weak understanding of a concept, we will either provide supplemental information or direct them to a source where they can improve their knowledge of the concept.

At mid-semester, we will administer the same concept inventory and compare the results with those that we obtained at the beginning of the course. We will also evaluate student weaknesses from the results of a mid-semester exam. If most students have still not grasped a concept, we will focus more on student learning of that concept. At the end of the course, we will administer an expanded concept inventory designed to assess all aspects of the new course learning outcomes. We will use this expanded concept inventory to assess how well the students have learned the new real-time signal processing concepts that have been added to this course compared to past offerings. We will use this new concept inventory to assess the background of the students the next time this course is offered. These evaluations will provide us with a measure of the learning in the course and will guide us in future offerings of the course.

Summary

This course includes labs that emphasize real-time digital signal processing techniques that are different from the off-line approaches to digital signal processing taught in most biomedical digital signal processing courses. Students experience hands-on experience of programming real-

time microcontroller-based signal processing algorithms. We believe that the lab experience including real-time analysis of their own live ECGs will greatly enhance the students' design capabilities and appreciation of the challenges of working with real-time biomedical signals.

Resources Available

The UW DigiScope MATLAB program and a pdf copy of the textbook may be obtained directly from the authors.

References

- [1] PhysioNet, <https://www.physionet.org>.
- [2] W. J. Tompkins (ed.) *Biomedical Digital Signal Processing*, Englewood Cliffs, NJ: Prentice Hall, 1993.
- [3] Nucleo-32 (STM32F303K8) microcontroller board, <https://os.mbed.com/platforms/ST-Nucleo-F303K8/>
- [4] Arm Mbed OS Developer Site, <https://os.mbed.com>.
- [5] R. Toulson and T. Wilmshurst, *Fast and Effective Embedded System Design: Applying the ARM mbed*, 2nd Ed. Oxford UK: Newnes, 2017.
- [6] A. J. Nimunkar, "Microcontrollers in Medical Instrumentation," in J. G. Webster and A. J. Nimunkar (ed.), *Medical Instrumentation: Application and Design*. 5th ed. New York: Wiley, 2018.
- [7] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, **BME-32**(3):230-236, 1985.