Building a Heart Rate Monitor: A Hands-On Project to Teach Programming Concepts to Undergraduate Students

Trevor Ham
Department of Biomedical Engineering
University of Akron
Akron, OH 44325
Email: trh56@zips.uakron.edu

S. Cyrus Rezvanifar
Department of Biomedical Engineering
University of Akron
Akron, OH 44325
Email: sr106@zips.uakron.edu

Rouzbeh Amini
Department of Biomedical Engineering
University of Akron
Akron, OH 44325
Email: ramini@uakron.edu

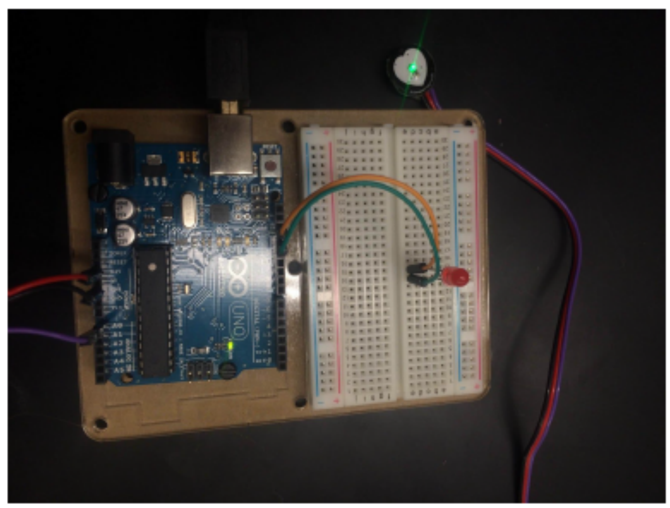

Figure 1: Arduino-based heart rate monitor, consisting of a photoplethysmograph connected to an Arduino UNO and an LED light which blinks on each beat connected via breadboard.

Introduction: Due to the pervasive nature of computer-driven devices (networked sensors, "smart" devices, etc.) and computationally-driven engineering tools (finite element analysis, machine learning, 3D modeling, etc.), undergraduate engineering students should graduate with a fundamental understanding of computer programming. For example, in the biomedical engineering field, devices such as a smart pill bottle which tracks doses and encourages patient compliance (Pillsy™), or an insulin pump which replicates some of the closed-loop functionality of the pancreas (MiniMed™ 670G) showcase the power of new software-based technologies to revolutionize the medical devices field. Whereas in the past, graduates may have listed "Microsoft® Office" on their resumes, today they are better off listing proficiencies in specific programming languages (e.g. MATLAB or Python) and showcasing examples of "smart" devices that they've worked on for student projects. Finally, for those students who wish to pursue graduate training in biomedical engineering, computer programing is an essential tool in many fields, including image analysis[1, 2], control of testing equipment[3, 4], and finding numerical solutions for initial and/or boundary value problems[5, 6].

At the University of Akron, undergraduate biomedical engineering students are required to take a 200-level undergraduate programming course (Biomedical Computing), where they are taught to use the MATLAB programming language and environment. Acknowledging the educational values of active learning[7-10], we designed a hands-on project for this class, giving the students an opportunity to actively participate in the design, development, and testing of a "smart" device. For this hands-on project, we chose the Arduino microprocessor, which can be purchased inexpensively and for which there are many commercially available sensors. Originally, we allowed the students to design a product to meet a need that they chose themselves (e.g., measuring joint angles using a potentiometer), but over time we refined the project and gave them specific design criteria that must be met. We settled on a heart rate monitor (HRM), consisting of a photoplethysmograph connected to an Arduino UNO microprocessor (Figure 1). Despite the robust Arduino interface built in to MATLAB, we also wanted to explore whether having to program the device in a different programming language (Arduino Integrated Development Environment, AIDE, which closely resembles C syntax) would reinforce basic programming concepts. To that end, we split a section of the course into two groups for the project (one using MATLAB, one using AIDE) and then compared final exam scores between the groups. The students were also given a survey to assess their perceptions of how the project influenced comprehension of the course material.

Materials and Methods: The students were sorted into teams of 3 based on average exam 1 grade, and then those teams were sorted into either group to keep prior exam performance consistent between groups. Both groups (MATLAB and AIDE) were instructed to build the HRM from an Arduino UNO and a commercially-available photoplethysmograph. They were given separate instruction on how to use the Arduino (1.5h each) and a detailed walkthrough that was tailored to each language. We gave the students 5 weeks to complete the project, and then asked them to complete detailed surveys regarding their perceptions of the project. Finally, we compared final exam scores (all students took the same final exam, which was in MATLAB) between the groups using Student's t-test ($\alpha = 0.05$).

Results and Discussion: On the final exam, the MATLAB students achieved an average score of 79.50 ± 19.02 (mean ± standard deviation), while the AIDE students' average was 82.64 ± 13.97. A comparison of final exam scores showed that there was no significant difference between the two groups ($p = 0.622$), indicating that there was no positive or negative effect on final exam performance when students were asked to design a "smart" device using a second, new programming language. This means that students gained proficiency in another language while not sacrificing overall performance on the final exam. Additionally, it is possible that completing the project in another language did reinforce fundamental concepts in a manner which the final exam did not measure. The results from the surveys shed some light on the nuances between groups: more AIDE students (13/14) than MATLAB students (11/14) indicated that they enjoyed the project. Interestingly, the majority of MATLAB students (10/14) thought that the project helped them to prepare for the final, while none of the AIDE students did (0/14). This result suggests that by asking the students to use AIDE for the project, we required them to learn another language while still putting in extra time (i.e., time spent on the project was not time spent preparing for the final) to retain their understanding of MATLAB for the final exam. The ability to juggle multiple programming languages at once and conceptualize programming as a broad set of tools (rather than just one single interface) will undoubtedly be helpful for many students going forward.

However, the effectiveness of this project-based component goes beyond learning two languages. Learning about how "smart" devices are constructed and programmed helps the students to actively participate in a project that clearly has real-world applications. Indeed, both groups scored very well on the project itself: 94.79 ± 7.50 and 96.43 ± 10.11 for MATLAB and AIDE, respectively. Overall comments for the project indicated that many students strongly felt that the project helped to prepare them for a career in biomedical engineering.

Based on the results from this study, we have continued to refine the hands-on project and experiment with different outcomes to measure project efficacy. We have dropped the AIDE component from the project, as we felt that it was ancillary to the main impact (active participation in a project that has clear parallels to real-world product design). We have also implemented new outcome measures, and are currently working to understand how students' self-efficacy changes after completing a hands-on project.

Conclusion: Students completed a hands-on project to design, build, program, and test a HRM using an Arduino UNO and commercially-available photoplethysmograph. They completed the project in one of two different languages: AIDE or MATLAB. Learning a new language specifically for the project did not affect average performance on the final exam. This result indicates that learning two languages in one semester does not adversely affect sophomore biomedical engineering students' overall academic achievement. In the future, we are working to measure changes in self-efficacy following our hands-on project.

References

. [1] Powell, T. A., Amini, R., Oltean, A., Barnett, V. A., Dorfman, K. D., Segal, Y., and Barocas, V. H., 2010. "Elasticity of the porcine lens capsule as measured by osmotic swelling". *Journal of Biomechanical Engineering,* 132(9), 08, pp. 091008–091008–9.

. [2] Yair, M., DiMarzio, C. A., McLean, J., Ruberti, J. W., Robitaille, M., and Zareian, R., 2014. Image processing methods and systems for fiber orientation, Dec. 22. US Patent App. 14/578,869.

. [3] Amini Khoiy, K., Biswas, D., Decker, T. N., Asgarian, K. T., Loth, F., and Amini, R., 2016. "Surface strains of porcine tricuspid valve septal leaflets measured in ex vivo beating hearts". *Journal of Biomechanical Engineering,* 138(11), 10, pp. 111006–111006–9.

. [4] Kloefkorn, H. E., Pettengill, T. R., Turner, S. M. F., Streeter, K. A., Gonzalez-Rothi, E. J., Fuller, D. D., and Allen, K. D., 2017. "Automated gait analysis through hues and areas (agatha): A method to characterize the spatiotemporal pattern of rat gait". *Annals of Biomedical Engineering,* 45(3), Mar, pp. 711–725.

. [5] Amini, R., and Barocas, V. H., 2010. "Reverse pupillary block slows iris contour recovery from corneoscleral indenta- tion". *Journal of Biomechanical Engineering,* 132(7), p. 071010.

. [6] Gsellman, L., and Amini, R., 2016. "Patients with intravitreal gas bubbles at risk of high intraocular pressure without exceeding elevation of surgery: Theoretical analysisvitrectomy, high iop, and no high altitude travel". *Investigative Ophthalmology & Visual Science,* 57(7), pp. 3340–3347.

. [7] Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., and Wenderoth, M. P., 2014. "Active learning increases student performance in science, engineering, and mathematics". *Proceedings of the National Academy of Sciences,* 111(23), pp. 8410–8415.

. [8] Hall, S. R., Waitz, I., Brodeur, D. R., Soderholm, D. H., and Nasr, R., 2002. "Adoption of active learning in a lecture- based engineering class". In Frontiers in Education, 2002. FIE 2002. 32nd Annual, Vol. 1, IEEE, pp. T2A–T2A.

. [9] Prince, M., 2004. "Does active learning work? a review of the research". *Journal of Engineering Education,* 93(3), pp. 223–231.

. [10] Clyne, A. M., and Billiar, K. L., 2016. "Problem-based learning in biomechanics: Advantages, challenges, and imple- mentation strategies". *Journal of Biomechanical Engineering,* 138(7), p. 070804.