µSAFABOT: A Robotics Learning Platform for a Hands-On Laboratory Based Approach in an Introductory ECE Course

Abstract

This paper focuses on the design, implementation, and evaluation of an introductory electrical and computer engineering course that integrates theory, hands-on practice, and real-world application through the implementation of a fully customizable robotics system, µSAFABOT. This course is required for all electrical and computer engineering students, but specifically targets undeclared students at a liberal arts military academy. The previous version of the course was a traditional, lecture-based course that introduced circuits and software design. The new version of the course covers most of the same topics, but through a series of projects that culminate in a robotics maze competition. Redesign was motivated after two shortfalls were discovered: first, the number of electrical and computer engineering majors graduating each year was declining, and second, students graduating out of the major did not have necessary experience or knowledge in robotics.

Using best practices in engineering education, the course transitioned from a lecture model to a project-based learning model that includes three blocks over a forty-lesson semester: block 1, introductory topics; block 2, robotics design and implementation; and block 3, maze competition. Each laboratory includes a brief fifteen-minute introduction to a fundamental electrical and computer engineering concept and 3.5-hours of hands-on application. For example, after learning how the average power of a system can be controlled via pulse-width modulation, students integrate motors into the robot and connect each motor to a modern measurement tool to observe the digital pulse-width modulation signals sent to the robot and calculate the response time of the motors.

Evaluation measures include a pre/post survey that measure student excitement in the course, intent to major in electrical and computer engineering, and understanding of the field. Additionally, an exit survey upon graduation evaluates student intent to pursue a career in robotics. Lastly, registration data observes pre/post number of students in the major. Results show significant increases in interest in the field of electrical and computer engineering, number of majors, and student learning.

Introduction

Over the past ten years there has been a steady negative trend in the number of electrical and computer engineering (ECE) majors graduating from the United States Air Force Academy (USAFA). In 2010 there were 28 graduates. In 2020 there were 15 ECE graduates. This decline is not necessarily unique to USAFA. The United States saw only 10% of the global science, technology, engineering, and math (STEM) bachelor's degrees in 2018 [1]. About half of students who enter a STEM program do not graduate with a STEM degree; most of these students drop out during their first or second year of college due to the increasing difficulty and complexity of STEM programs [2], [3], [4]. To motivate and engage students, undergraduate educators must design a curriculum that integrates hands-on learning early in the program that is relevant and can provide students a sense of ownership of their educations [5].

Additionally, engineers with robotics skills are specifically increasing in demand, but the needs are outpacing the workforce and a shortfall is emerging that undergraduate programs are not satisfying [6], [7], [8]. To fill both shortfalls, the number of ECE majors and those with a knowledge base in robotics, the Introduction to Electrical and Computer Engineering course was redesigned with two primary goals: first, increase recruitment for the ECE major and, second, spark interest in robotics. To accomplish these goals, the course was transitioned from a traditional, lecture-based course to a predominantly project-based course focused on the design and implementation of a robotics system shown in Figure 1, µSAFABOT.

This paper introduces µSAFABOT, a testbed developed to spark and promote undergraduate robotics research. The testbed is a modular platform built using additive technologies and an Arduino Uno that provides control of the on-board motors and sensors. This paper also describes how the introductory course was redesigned to teach students fundamental ECE skills through the implementation of µSAFABOT. The course also introduces complex robotics topics that are expanded on in a series of classes in a new robotics track in the major.

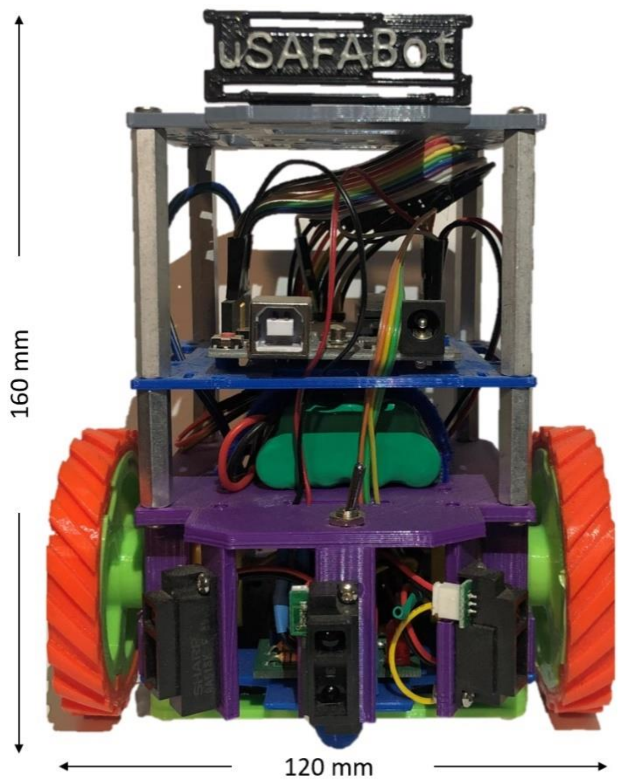


Figure 1: µSAFABOT.

Desired Objectives for Course

The objectives of the course were specifically aligned to the two goals of the course, increase recruitment for the ECE major and spark interest in robotics research. Robotics is a field that requires an understanding of topics covered throughout a typical ECE curriculum [9], therefore, this redesign focused on covering a broad spectrum of topics to provide a surface-level understanding of ECE and robotics skills that are expanded on in future courses. Most importantly, the course attempts to motivate students by focusing on hands-on experience and project-based learning, while also reaching the following educational objectives:

1. Prepare students for upper-level courses by teaching fundamental ECE tools and techniques.
2. Maximize student engagement.
3. Increase student understanding of the electrical and computer engineering field and major.
4. Increase student recruitment for the ECE major.
5. Facilitate student interest in robotics.

Theoretical Motivation

The theoretical motivation that drove course redesign was centered on evidence-based pedagogy pointing towards the benefits of project-based learning. Fundamentally, Kolb's Experiential Learning Model asserts that a cycle of concrete experience, reflective observation, abstract conceptualization, and active experimentation results in higher student learning [10]. Implementation of Kolb's method along with the integration of theory and hands-on practice in and introductory engineering course at Harvey Mudd College saw a statistically significant gain in student learning and an increased enthusiasm amongst female students [5]. To incorporate these theories, our course implemented cycling along with an increase in hands-on experience through the course structure. Students are introduced to fundamental ECE topics in block one through a series of hands-on laboratories. In block 2, students begin building their robots and must recall previous concrete experiences from block 1. They must reflect on how the topics they learned earlier in the class apply to the more complex example of a robot and experiment with these theories during robot development. Lastly, they must incorporate all previous lessons during block 3, maze competition.

The course redesign was also focused on developing professional traits that students could use in their careers as leaders in the United States Air Force. Findings show that active learning in the classroom provides longer-term benefits such as "career preparation, professional behaviors, and enhanced learning or learning processes" [11]. Team-based learning (TBL) also enhances professional skills and encourages collaboration, communication, and other interpersonal skills [12]. Part of the course redesign included an increase focus on engineering applications and collaboration. All labs allow students to work in small groups while building their individual robots.

Lastly, while there are no grades assigned in this course, frequent checkpoints allow for no-stakes testing which has been proven to increase student performance, especially in minorities [13]. Instead of taking tests, students work at their own pace to meet deliverable deadlines that they set individually. For example, a student may decide to have their motor driver and power distribution boards soldered by lesson 21. This form of contract learning enables students to work at their own pace to gain understanding [14] and empowers them to take responsibility for their education [15].

The µSAFABOT Testbed

Given the above motivation for hands-on project-based learning, we provide our µSAFABOT testbed. Built from the ground up by students, the testbed provides a platform for hands-on experimentation through ten laboratories.

Physical Platform

The platform for the µSAFABOT testbed includes the integration of three main components adapted from previous testbed designs [16]: a body, nerves, and brain.

The **body** of the testbed includes customizable 3D printed tiers that enable the mounting of motors, sensors, and other components. The µSAFABOT testbed's current configuration can carry an Arduino Uno, Raspberry Pi 4, two direct current (DC) motors, three Sharp infrared distance sensors, and a Pololu QTR-8RC Reflectance Sensor Array. Mounting holes and slots in the layers provide flexibility to mount components in a variety of orientations. The testbed is powered using a pack of seven rechargeable AAA batteries stored on one of the layers.

The **nerves** consist of a custom printed circuit board (PCB) that provides voltage regulation and motor control. The PCB takes an input voltage between 7 V and 20 V and provides a fixed 3.3 V and 5 V output at a peak current of 250 mA and 1.5 A, respectively. An integrated motor driver chip converts direction and PWM signals from the Arduino Uno to power and control two DC motors.

The **brain** is an Arduino Uno Rev3. On-board sensors are connected to the brain using digital and analog input and output pins. The brain makes decisions and passes direction and speed (PWM) signals to the nerves using digital pins. The brain is powered using the 5 V/1.5 A output on the PCB.

Software Architecture

The software architecture is presented in Figure 2 and creates a control loop between the brain, nerves, and body. Starting at the brain, a controller process continuously polls the body for environment information from on-board sensors to inform the robot's current state. The controller utilizes the current state information and sensor inputs to provide direction and PWM signals to the nerves. The nerves translate the control signal to drive the left and right motors accordingly, completing the loop.

Motivation

There are many off-the-shelf robotics kits available commercially [17], [18], [19]. However, from experience, many of these kits lose support over time, have difficult instructions, or have parts impossible to replace when a student inevitably breaks something. Building µSAFABOT from the ground up allows students to not only understand every interaction between each hardware and software component, but also allows instructors to easily replace parts or troubleshoot issues. The µSAFABOT system is less focused on fundamental understanding of each underlying system, but more on the integration, design, and motivation sides of student learning.

Future Work with the µSAFABOT Platform

One of the benefits of the µSAFABOT testbed is its modularity. Two areas currently under research to expand µSAFABOT's capabilities are network connectivity and simulation. As previously mentioned, the body of the testbed can hold a Raspberry Pi 4. Two students are researching how to expand the power capabilities of the testbed to enable a Raspberry Pi 4 to act as a networked brain and allow a supervisor to wirelessly control the μSAFABOT. This would allow for the testing of more advanced artificial intelligence and teaming algorithms with the testbed. Additionally, students are working to model the μSAFABOT body and physical constraints to enable students to test their maze algorithms in simulation prior to implementing them on the real robot.

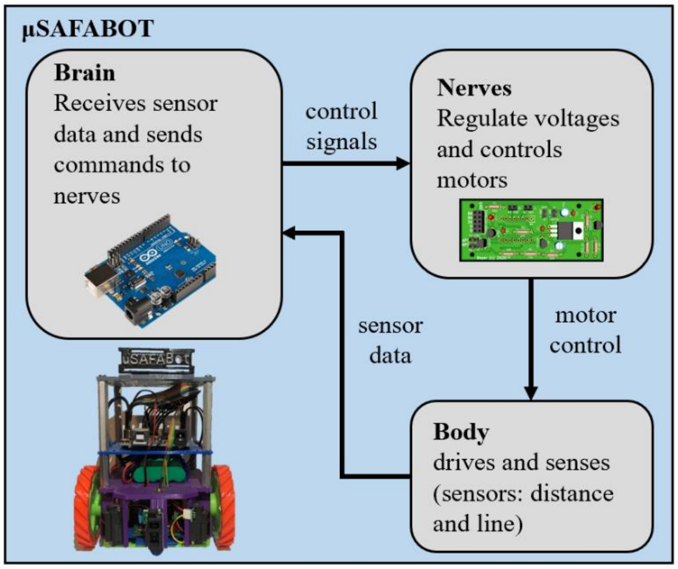


Figure 2: Architecture diagram for the μSAFABOT testbed.

μSAFABOT as an Introductory Educational Tool

The course redesign focused on effectively integrating the educational methods proposed by theory with the μSAFABOT testbed. Active learning techniques are designed into each lesson and, in keeping with TBL practices, students are encouraged to collaborate. As a result, students more advanced in a topic often assist those struggling around them, further forming their skills through helping their peers. The course consists of ten incremental laboratories which build on each other and culminate in a maze competition. At the beginning of each laboratory individual students agree to a deadline for completion promoting performance and removing anxiety regarding grades [14]. The course is broken into three blocks over a forty-lesson semester: block 1, introductory topics; block 2, robotics design and implementation; and block 3, maze competition.

Block 1: Introductory Topics

This course is the first engineering class for most students; therefore, four labs provide fundamental ECE skills and tools that are then used in building μSAFABOT in block 2. The course starts by introducing basic circuit troubleshooting techniques and tools. For example, students learn how to utilize a digital multimeter and other modern measurement tools to find circuit values. Unique to this course is Liquid Instrument’s Moku:Lab, a new device that integrates 12 tools into one system and enables cadets to learn how to use ECE tools with an iPad user interface as seen in Figure 3.

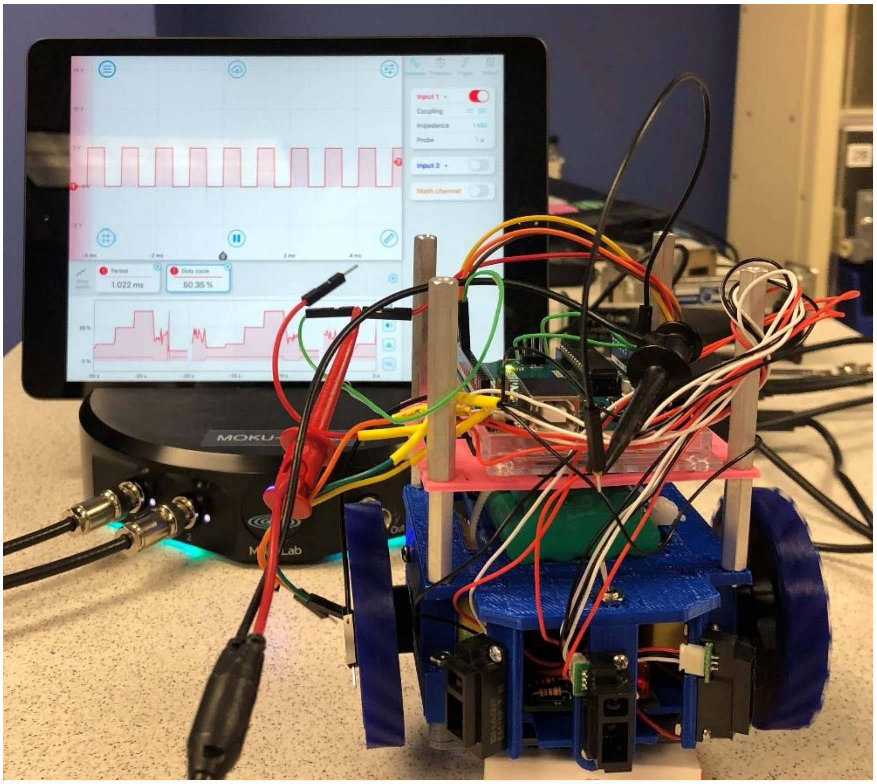


Figure 3: Liquid Instrument's Moku:Lab displaying PWM signal sent to one of the DC motors.

The next lab introduces additive manufacturing by training students to use LulzBot 3D printers and Fusion 360 modeling. Students pick an object to design and then print the model. These skills are called upon later to design and customize the μSAFABOT's body in block 2. The students then learn how to solder with a basic soldering kit such as a burglar alarm or bagpipes. This prepares students to build the motor control and power distribution board for the robot. Lastly, the students learn how to program using an Arduino Uno and peripheral devices such as light-emitting diodes, switches, sonar sensors, and photoresistors. These skills are applied when programming the Arduino controller on the robot.

Block 2: Robotics Design and Implementation

The second block is designed to reflect a large-scale engineering project and uses an incremental approach observed in the spiral model or agile software development model. Each laboratory is presented as a distinct requirement and product solution. Students must stick to a deadline and work through an entire development process to plan, design, develop, test, and evaluate each component before moving to the next subsystem. Each lab corresponds to a specific engineering concept shown in Figure 4.

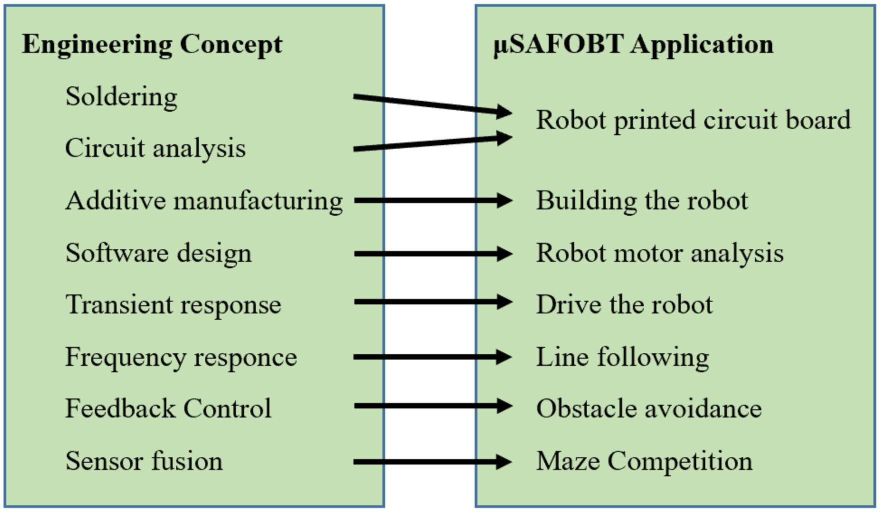


Figure 4: Correlation between engineering concept and corresponding µSAFABOT laboratory.

Students begin the robot build by developing the requirements for a motor controller and power distribution board. After students design their own solution, an operational circuit is provided to ensure success. The students must then solder the circuit, investigate specific test points, and troubleshoot if issues are found. The completed printed circuit board (PCB) is then integrated into a testbench that will demonstrate if all parts of the PCB are operational. Figure 5 provides a model of a completed printed circuit board without the off-the-shelf motor driver board (TB6612FNG Dual Motor Driver Carrier). The complete circuit diagram for the PCB is available in Appendix A.

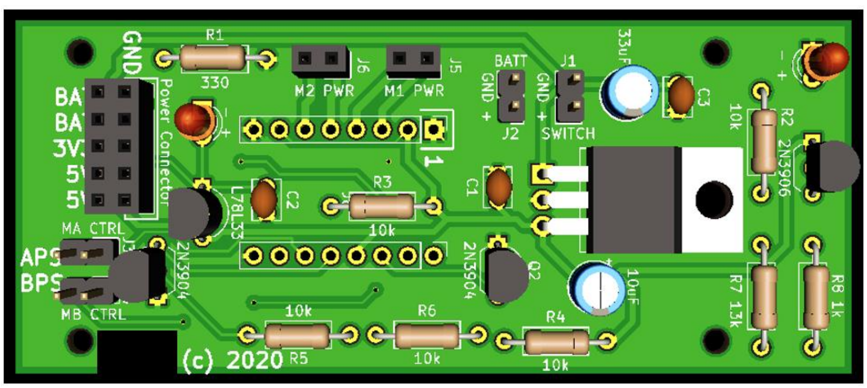


Figure 5: 3D Model of the motor controller and power distribution board.

Next, students use additive manufacturing and modeling to design the robot body and build the robot. The body is fully customizable, but templates are provided as a starting point. Several off-the-shelf components (motors, nuts, bolts, switches, etc.) are integrated to the body to create the robot. The students must wire their PCB to the different systems before completing the robot per the robot circuit diagram provided in Figure 6. A picture of a student's µSAFABOT mid-build is provided in Figure 7. Appendix B provides models for each of the 3D printed components of the robot, and Appendix C supplies the complete list of components used for the robot build.

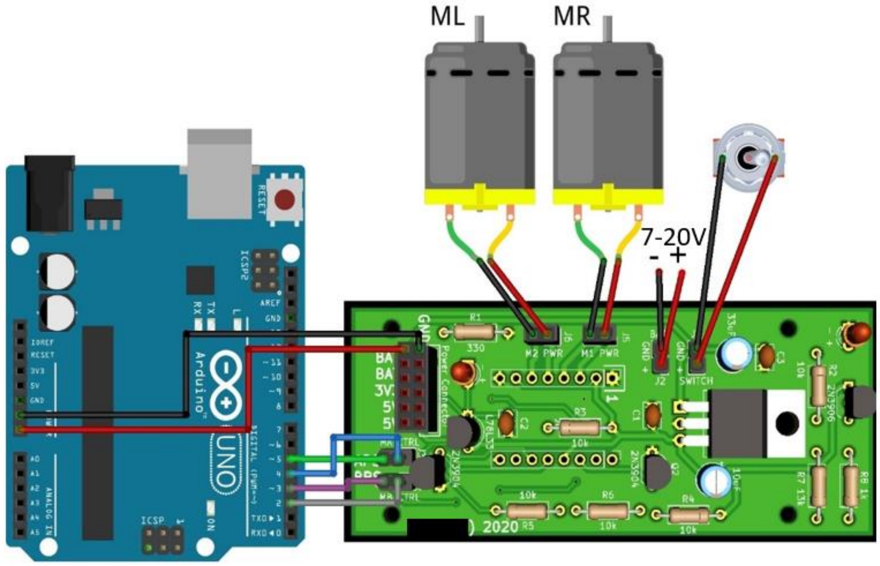


Figure 6: Circuit diagram for the printed circuit board

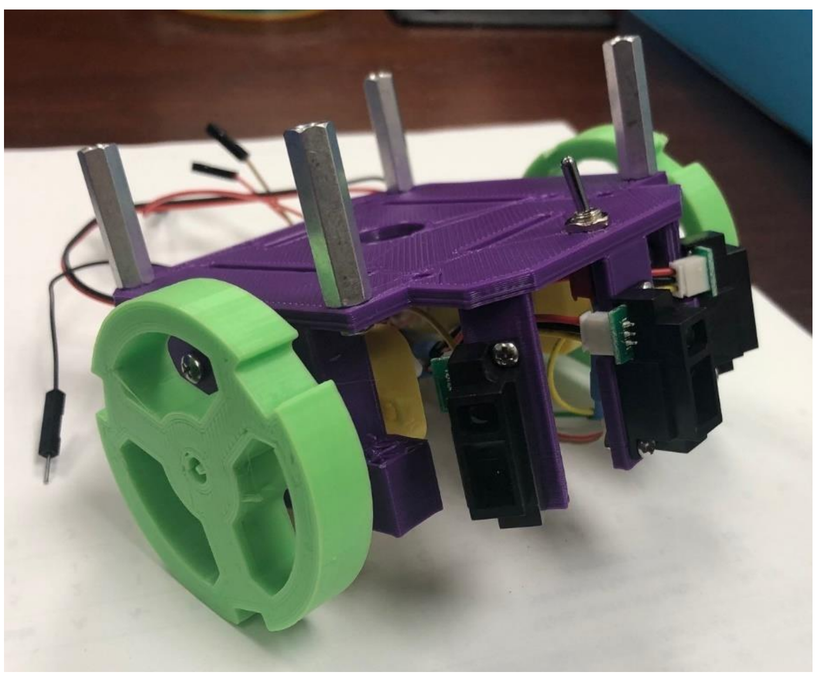


Figure 7: Partially built µSAFABOT.

After building the robot, students test and characterize the motors using the Moku:Lab tool. They also program the motors to drive the robot in several different patterns. Students observe that there is no feedback from the motors, and, consequently, it is extremely difficult to drive the two motors at the same speed.

The last two labs include integrating the line following sensors and distance sensors. Both are calibrated and tested individually then integrated with the motors. The robot must follow a line, one wall, and stay between two walls. Basic control systems are introduced, and students design a proportional and integral controller to improve line and wall following.

Block 3: Maze Competition

The last block of the course integrates all previous labs as students compete in a maze competition. The maze competition demonstrates that interfacing embedded robotics systems with the physical world is a very complex task. The maze competition consists of three different levels, each presenting unique problems. The first level is a line following maze which challenges students to balance motor response with complex turns. Students must experimentally establish proportional constants to ensure their robot responds fast enough to follow the line through turns. The second level of the competition is wall following through a complex maze. Students experiment with different methods to turn and solve the maze in the quickest and most reliable manner (they must complete three successful time trials). The third level combines the two and requires students to transition between line following and wall following, seamlessly. A diagram of the maze is provided in Figure 8.

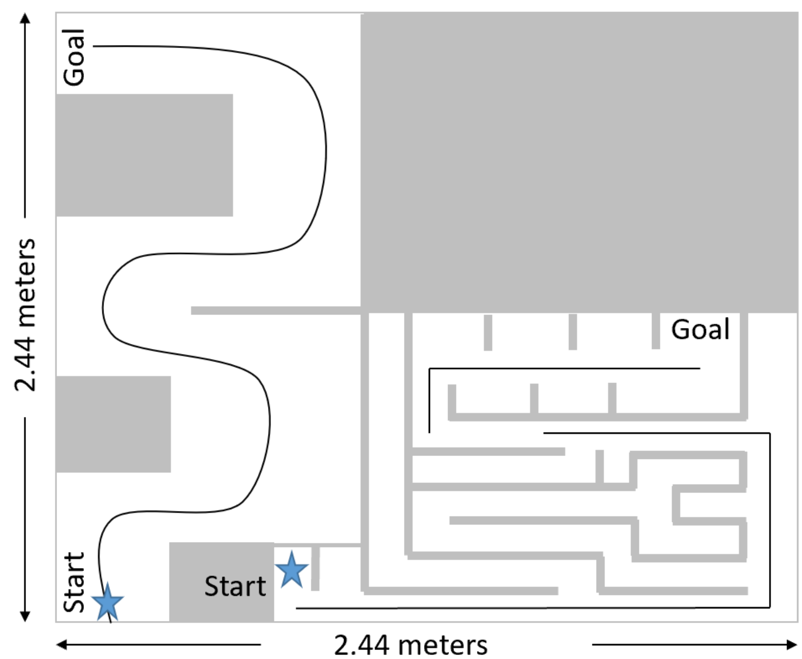


Figure 8: Maze with three levels (grey lines indicate walls, while black lines indicate line following tracks).

Evaluation Methods

The evaluation methods used to measure objectives are summarized in Table 1. Each method was used in the original course in Fall 2017 and repeated with the redesigned course in the Spring and Fall semesters of 2018, 2019, and 2020. A total of 70 students completed surveys over this period, 23 from the first course and 47 students from the revised course.

Table 1: Evaluation objectives and methods.

Evaluation Objectives	Method
Increase student preparedness for upper-level courses	Pre/post survey
Increase engagement	Pre/post survey and Student Evaluations of Teaching
Increase student understanding of the electrical and computer engineering field and major	Pre/post survey
Increase student recruitment for the ECE major	
Facilitate student interest in robotics	Exit survey

The survey results include both quantitative and qualitative responses. Quantitative responses are based on a 5-point scale with 0 referring to a negative response and 5 referring to a positive test. The first student class to take the redesigned course graduated in May 2021; an exit survey was provided to these students requesting qualitative responses regarding the impact of the redesigned course in their interest to complete robotics research.

Results

The evaluation methods described above were used to assess the five desired course objectives. When applicable, a two-sample t-test with an alpha level of 0.05 was accomplished to demonstrate the significance of the redesign.

1. **Increase student preparedness for upper-level courses.**
   Instructors were queried which key skills students should have experience with prior to upper-level courses. These skills included understanding of circuit analysis, soldering, electronic prototyping, software design, hardware troubleshooting, software troubleshooting, and use of MATLAB. The course is less focused on fundamental understanding of these key skills, and more on getting students familiarity with the topics. The topics are then expanded on in future course. Students were asked during pre- and post-course surveys what their experience level was for each of these skills and these responses were used to measure student preparedness for upper-level courses. The first column set in Figure 9 illustrates the improvement in student preparedness between courses. Students started both versions of the course with about the same level of knowledge, however, students in the new course felt more confident in their experience (3.57/5.0) compared to the old (3.15/5.0). Experience in ECE fundamentals increased significantly in the new course (1.92/5.0 to 3.57/5.0, $p = 1.48 \times 10^{-48} < \alpha$).

   Additionally, in an exit survey, seniors were asked, "Which topics in the introductory course were most helpful throughout your time in the ECE department?" One student responded, "The µSAFABOT was a fantastic introduction to all of the real applications of ECE that I built on in my other classes." Another student enjoyed the incremental approach of the course and reported that the course was "perfectly designed and implemented such that each student had to understand each concept taught throughout the semester in order to be successful."

2. **Increase engagement.**

Student engagement was measured during pre- and post-course surveys in which students were asked if they were excited about the course. The second column set in Figure 9 illustrates the increase in excitement between courses. Students began the course with an increased excitement compared to the previous version of the course (from 2.69/5.0 with the previous course to 3.26/5.0 for the new course; the increase is likely due to efforts to advertise the new version of the course and students hearing about the redesign and use of μSAFABOT from other students). The previous version of the course saw a slight increase in excitement levels (2.69/5.0 to 3.26/5.0, $p = 0.02 < \alpha$), while the redesigned course saw a larger increase in excitement (3.17/5.0 to 4.11/5.0, $p = 1.66 \times 10^{-6} < \alpha$).

In the optional comments section of the survey for the new version of the course, one student reported, "This course has probably been my favorite course I have ever taken, not just at USAFA, but also in all my education thus far." Another student who spent much more time than required working on μSAFABOT, said he did so "not because [he] *had* to, to earn a grade, but because [he] was engaged and passionate about the nature of the class and what [he] was learning."

3. **Increase student understanding of the electrical and computer engineering field and major.**
   Student perception of their understanding of the ECE field and major was measured using the third survey item in Figure 9. Students in the new version of the course believed they had a better understanding of what it meant to be an ECE professional (4.27/5.0 compared to 3.20/5.0 in the old version).

4. **Increase student recruitment for the ECE major.**
   Two forms of evaluation were used to determine if the new version of the course helped increase student recruitment: first, student intent to declare (fourth survey item results shown in Figure 9), and second, the number of students per class that declare. Both versions of the class saw an increase in the intent of students to declare (from 2.87/5.0 to 4.13/5.0 for the old version and from 3.09/5.0 to 4.36/5.0 for the new version). However, in the latest section of the revised course 79% of students in the course declare ECE, while the last section of the old version only saw 45% of students declare.

   While it cannot be directly correlated to the course redesign, after the first year of the redesign the ECE department had the best recruiting class in 20 years with 65% diversity of candidates and 14 total females (more than the last 6 classes combined). The five student classes prior to the course redesign had an average of 22 ECE majors per class, while there has been an average of 31 ECE majors per year since the redesign.

5. **Facilitate student interest in robotics.**
   Student interest in robotics was one of the more difficult goals to measure. Prior to the course redesign there was no focus on robotics in the course. The course's impact on robotics through the μSAFABOT testbed was primarily measured using student comments in the post-survey. One student commented that the course "motivated [him] to pursue graduate school for robotics and become an instructor at USAFA through the

graduate school program.” Another student saw the hands-on format of the class and focus on µSAFABOT “lead [him] to take more in-depth robotics courses…and an internship with the NSA focusing on Artificial Intelligence and Machine Learning.”

During the second iteration of the new course, instructors were advised to identify students with an advanced ability in robotics. A total of eight students were identified and targeted for robotics research in the department. Two years later, these eight cadets have accomplished over 24 semester hours of robotics research, developed an Introduction to Robotics Research course, and competed in robotics competitions. These students might not have been identified if the course was not centered around the hands-on, project-based model.

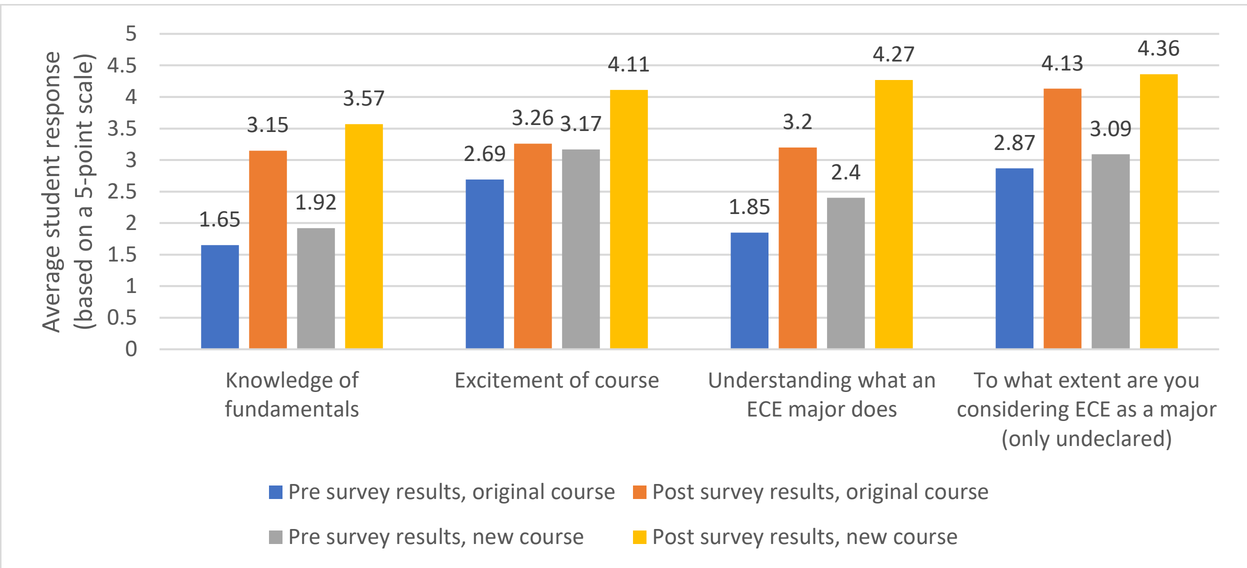


Figure 9: Pre/post survey results for original and new course (N=23 for the original course; N=47 for revised course).

Conclusion

Three years ago, USAFA’s Introduction to Electrical and Computer Engineering course was redesigned. Driven by a ten-year negative trend in the number of students declaring as ECE majors and a lack of robotics knowledge, the course was transitioned from a lecture-based class to primarily hands-on and project-based. Motivated by similar successes at Harvey Mudd College and Kolb’s Experiential Learning Theory, the course was designed to revolve around a robotics system designed by USAFA instructors, µSAFABOT. The µSAFABOT testbed is a modular robotics system that enables students to experiment with ECE fundamentals while building a robot from the ground up. Based on an incremental approach, lessons builds on each other and culminate in a maze competition. Students learn about the spiral dsign process as each lab requires them to stick to a schedule as they plan, design, develop, test, and evaluate each subsystem before moving to the next subsystem.

Results of pre/post surveys, a senior exit survey, registrar data, and student feedback show that there was a significant increase in student excitement and understanding regarding the field of ECE. Additionally, the course accomplished its goal of increasing recruitment in the major as the number of majors per year on average increased by 41%. Lastly, the course transition sparked an interest in robotics throughout the department and students. Overall, this course redesign has been very successful, and the department head even fights to teach it.

For more information on the technical implementation details and to implement the laboratories associated with μSAFABOT, please refer to https://github.com/AF-ROBOTICS/DFEC_RoboticsLearningPlatform.

References

[1] National Science Board, "Science & Engineering Indicators 2018," National Science Foundation, Alexandria, VA, USA, NSB-2018-1, 2018. Accessed: Mar 4, 2021. [Online]. Available: https://www.nsf.gov/statistics/indicators/.

[2] X. Chen, "STEM Attrition: College Students' Paths Into and Out of STEM Fields," National Center for Education Statistics, Institute of Education Sciences, U.S. Department of Education, Washington, DC, USA, NCES-2014-001, 2013. Accessed: Sep 15, 2020. [Online]. Available: https://nces.ed.gov/pubs2014/2014001rev.pdf.

[3] E. Seymour and N. M. Hewitt, *Talking about leaving: Why undergraduates leave the sciences*. Boulder, CO, USA: Westview Press, 2000.

[4] C. Drew. "Why Science Majors Change Their Minds (It's Just So Darn Hard)." The New York Times, November 4, 2011. Accessed: Mar 1, 2021. [Online]. Available: https://www.nytimes.com/2011/11/06/education/edlife/why-science-majors-change-their-mind-its-just-so-darn-hard.html.

[5] Lape, Nancy K., et al. "Integrating Theory and Hands-On Practice using Underwater Robotics in a Multidisciplinary Introductory Engineering Course." *2017 ASEE Annual Conference & Exposition*. ASEE Conferences, Columbus, Ohio. https://peer.asee.org/28561. 2017.

[6] U.S. Bureau of Labor Statistics, "Occupational Outlook Handbook: Mechanical Engineers," 2017. Accessed: Mar 1, 2021. [Online]. Available: https://www.bls.gov/ooh/architecture-and-engineering/mechanical-engineers.htm/.

[7] R. Thomas, R. Bhat, A. Khan, and P. Devan, "2018 Deloitte and the Manufacturing Institute skills gap and future of work study," Delloitte Development LLC, 2018. Accessed: Mar 1, 2021. [Online]. Available: https://www.themanufacturinginstitute.org/wp-content/uploads/2020/03/MI-Deloitte-skills-gap-Future-of-Workforce-study-2018.pdf.

[8] E. Demaitre. “Robots.Jobs addresses robotic skills shortage with a national approach.” The Robot Report, February 6, 2019. Accessed: Mar1, 2021. [Online] Available: https://www.therobotreport.com/robots-jobs-addresses-skills-shortage/.

[9] H. C. Powell and J. B. Dugan, “Embedded and cyber-physical computing courseware as the nexus of curriculum reform in electrical and computer engineering,” in *Proc of ASEE-SE Section*, Macon, GA, USA: ASEE, 2014.

[10] Kolb, David A. "Management and the learning process." *California management review* 18.3 (1976): 21-31.

[11] Clark, Renee M., Autar Kaw, and Mary Besterfield-Sacre. "Comparing the effectiveness of blended, semi-flipped, and flipped formats in an engineering numerical methods course." *Advances in Engineering Education* 5.3 (2016): n3.

[12] Michaelsen, Larry K., Arletta Bauman Knight, and L. Dee Fink, eds. *Team-based learning: A transformative use of small groups.* Greenwood publishing group, 2002.

[13] Haak, David C., et al. “Increased structure and active learning reduce the achievement gap in introductory biology.” *Science* 332, no. 6034 (2011): 1213-1216.

[14] H. Taylor, “Contract grading,” Nat. Inst. Educ., 1980.

[15] C. A. Tomlinson, *The Differentiated Classroom: Responding to the Needs of All Learners*. Upper Saddle River, NJ: Merrill-Prentice-Hall, 1999.

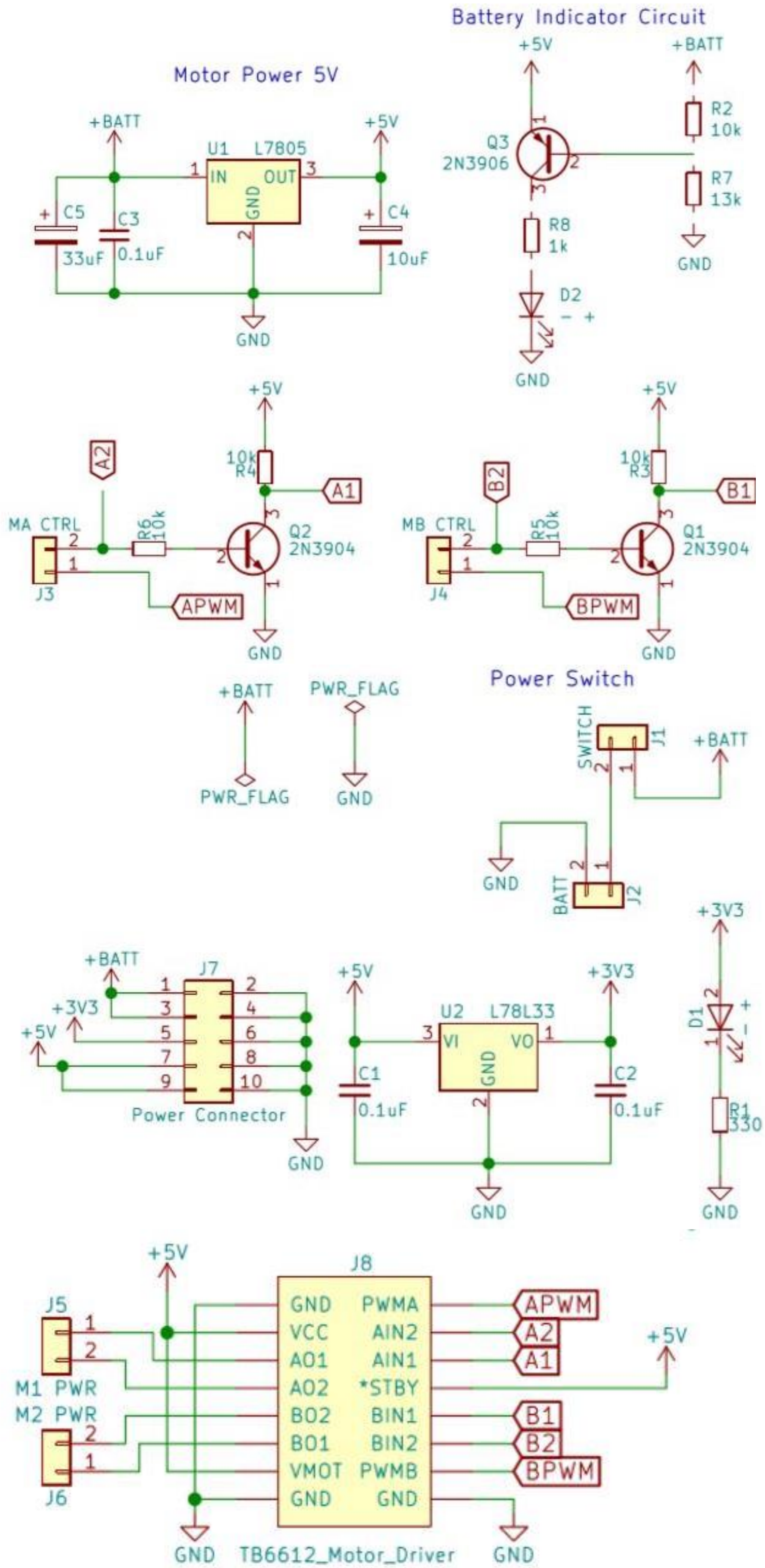
[16] T. L. Crenshaw and S. Beyer, “UPBOT: A testbed for cyber-physical systems,” in *Proc of the 3rd CSET*, Aug. 2010, Article no. 1-8.

[17] “Arduino Robot,” Arduino Store. Accessed: Mar 4, 2021. [Online]. Available: https://store.arduino.cc/usa/arduino-robot.

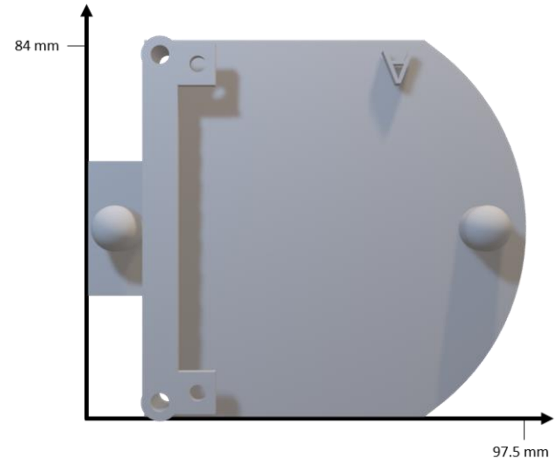
[18] “Pololu 3pi Robot,” Pololu Robotics & Electronics. Accessed: Mar 4, 2021. [Online]. Available: https://www.pololu.com/category/279/original-3pi-robot.

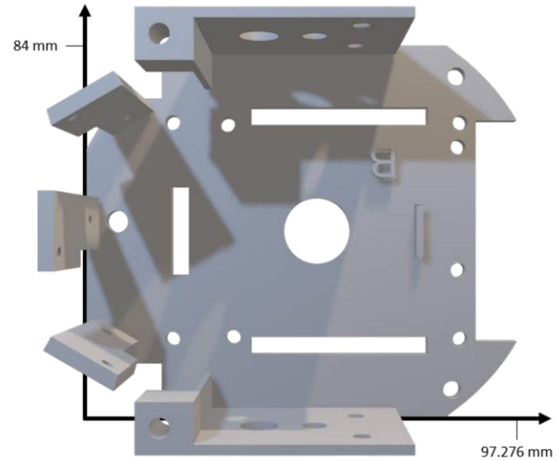
[19] “SparkFun micro:bit kit for micro:bit - v2.0,” sparkfun. Accessed: Mar 4, 2021. [Online]. Available: https://www.sparkfun.com/products/16275.

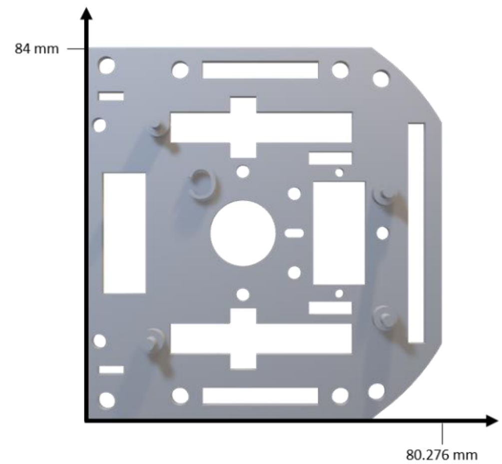
Appendix A. Motor Controller and Power Distribution Board Circuitry.

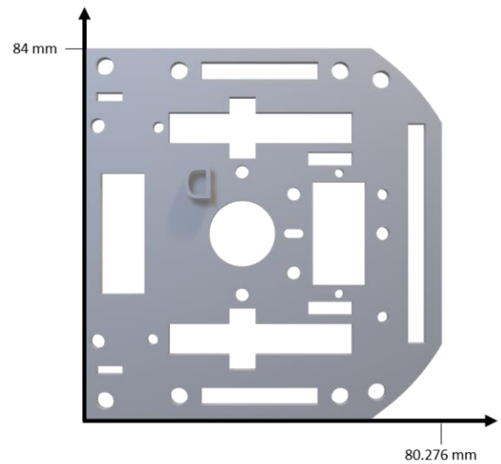


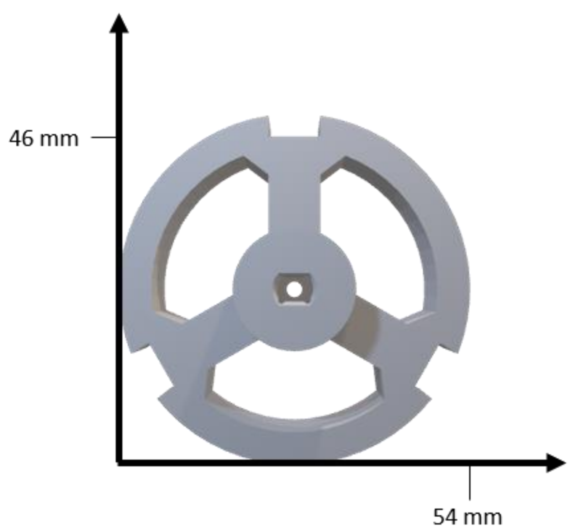
Appendix B. 3-Dimensional Models of µSAFABOT Testbed.

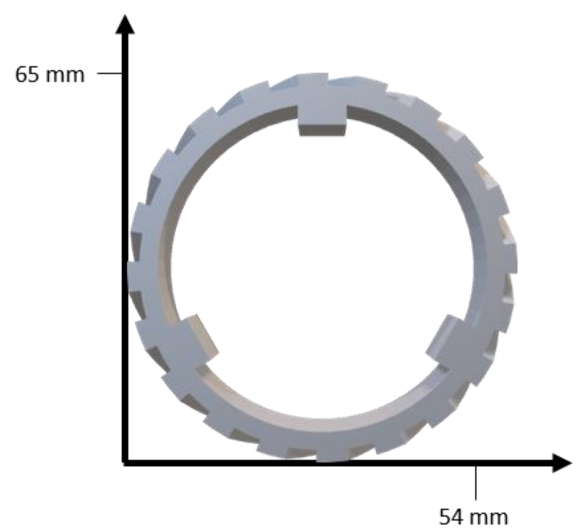












Appendix C. On-Board Sensors.

1. Soldered Printed Circuit Board (PCB)
2. Arduino Uno
3. 3D-printed structural pieces (A-D)
4. 2 - DC motors
5. 2 - wheels
6. 2 - tires (note left and right are different)
7. 1 - Rechargeable NiMH Battery pack (8.4 V)
8. Jumper wire pack
9. 1 - Switch
10. 3 - Sharp GP2Y0A51SK0F Analog Dist Sensor
11. 1 - QTR-8RC Reflectance Sensor Array
12. 1 - 1x11 90^o header
13. 8 - hex standoffs (4 short and 4 long)
14. Bolts
    (a) 6 - # 4-40 x 1"
        i. 4 for motors
        ii. 2 for connecting A and B
    (b) 8 - # 4-40 x 1/4"
        i. 8 for connecting standoffs to layers
    (c) 10 - # 2-56 x 1/4"
        i. 2 for line sensors
        ii. 6 for IR sensors
        iii. 2 for wheels
15. Nuts
    (a) 6 - # 4-40
    (b) 8 - # 2-56

Appendix D. Example Laboratory

Robot Motors

1 Objectives.

1. Understand the fundamentals of Pulse Width Modulation (PWM) including period, frequency, duty cycle, and pulse width.
2. Understand how PWM can be used to control the speed of motors.
3. Demonstrate understanding by manipulating the motors at different speeds.

2 Pre-lab - To be done prior of class.

2.1 Reading:

Adapted from Embedded Systems: Real-Time Interfacing to the MSP432 Microcontroller, Jonathan W. Valvano, Third Edition, 2019.

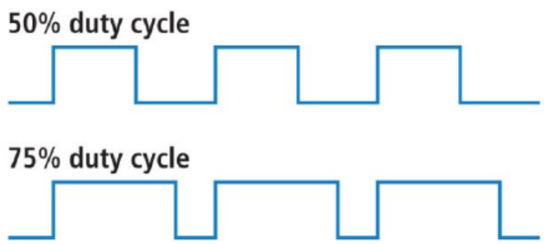
One effective way to deliver power to a DC motor in a variable manner is to use pulse width modulation (PWM). The basic idea of PWM is to create a **digital** output wave of fixed frequency, but allow the microctonroller to vary its duty cycle. The system is designed in such a way that **High+Low** is constant (meaning frequency/period is fixed). The **duty cycle** is defined as the fraction of time the signal is high:

$$duty\ cycle = \frac{High}{High + Low} = \frac{High}{Period}$$

Hence, duty cycle varies from 0 to 1 and will often be referred to as a percent. We interface this digital output wave to an external actuator (like a DC motor), such that power is applied to the motor when the signal is high, and no power is applied when the signal is low. We purposely select a frequency high enough so the DC motor does not start/stop with each individual pulse, but rather responds to the overall average value of the wave.

2.2 Key Terms:

- **Pulse Width Modulation (PWM)**: A technique to deliver a variable signal (power) using an on/off signal with a variable percentage of time the signal is on (duty cycle).
- **Period/Frequency**: Time required to complete one full cycle (on/off). Frequency is the inverse of period and is constant. The Arduino pins used by the robot operate at 980 Hz.
- **Duty Cycle**: Percentage of time a digital signal is **on** over a period of time



2.3 Examples:

1. Pulsing an LED on and off.

You have already applied PWM to pulse the red LED using the Arduino. You varied the duty cycle through the use of the *analogWrite()* function within the *for-loop* which varied the power applied to the LED and consequently, the brightness. The code you used is to the right.

```
// Global variable
static int LED_Pin = 11;

void setup() {
  // Set LED Pin as an output pin
  pinMode(LED_Pin, OUTPUT);
}

void loop() {
  // pulse the LED on (dim to bright)
  for(int i = 0; i< 255; i++){
    analogWrite(LED_Pin, i);
    delay(5);
  }

  // pulse the LED off (bright to dim)
  for(int i = 255; i >=0; i--){
    analogWrite(Led_Pin, i);
    delay(5)
  }
}
```

2. Answer the following given the example shown below in Figure 1.
   (a) What is the period of the signal?

   (b) What is the pulse width of the signal (how long is the signal **HIGH**)?

   (c) What is the duty cycle?

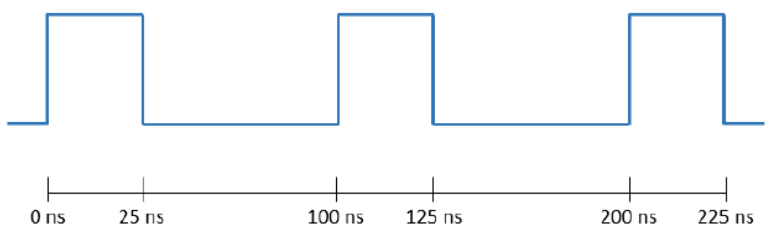


Figure 1

3. Given that the *analogWrite(pin, value)* function's *value* parameter is an integer that represents the duty cycle between 0 (always off) and 255 (always on), what would be the *value* passed to provide a 75% duty cycle to the motors?

2.4 Code:

1. Copy the Arduino sketch folder `robot_motors` from **Teams** (`Labs/robot_motor`)
2. Open the `robot_motors.ino` sketch.
3. In the *loop()* function, update the five calls to *Motor_Forward()* to drive the motor at the designated duty cycles.
   - The example provided, *Motor_Forward(50, 50);*, will drive both the left and right motors at (50/255)% of the duty cycle. This is equivalent to a duty cycle of about 20%.
4. Upload the sketch to the robot. Both motors of the robot should now operate at 6 different speeds (from 20% duty cycle up to 100% and back down to 15%).

3 Lab.

This lab will demonstrate how varying the PWM signal changes the average power delivered to the DC motors and, effectively, the speed of the motors.

3.1 Materials:

1. Assembled BeyerBot
2. Arduino programming cable
3. Laptop
4. Male-to-female/male Y-connector and male-to-male wire (provided for you in the Moku:Lab)

3.2 Connect the Moku:Lab probes:

1. Disconnect **ONLY** the wire from *Pin-5* on the Arduino.
2. Insert male end of the Y-connector into *Pin-5*.
3. Connect the male end of the wire coming from your Printed Circuit Board (the one disconnected from *Pin-5*) into the female end of the Y-connector.
4. Connect the positive (red) end of the Moku:Lab probe to *Pin-5* on the Arduino using the Y-connector.
5. Insert a male-to-male wire into the *GND-Pin* on the Arduino and connect the ground (black) end of the Moku:Lab probe to the other end of the wire.
6. Your setup should now look like Figure 2. All other wires should still be connected as they were previously.

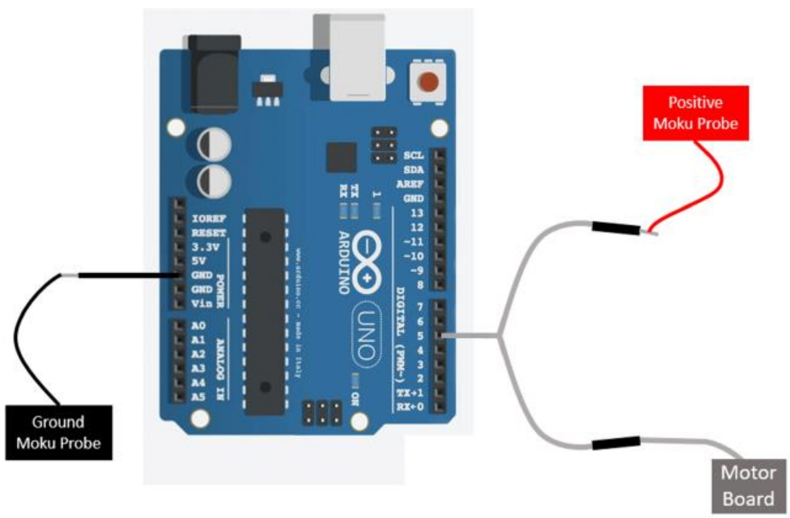


Figure 2

3.3 Measuring Pulse Width Modulation Signals:

1. Place your robot on the Digital Multi Meter (DMM) (to allow the program to run without your robot driving away)

2. Your Moku:Lab display should look like Figure 3. If it does not, raise your hand.

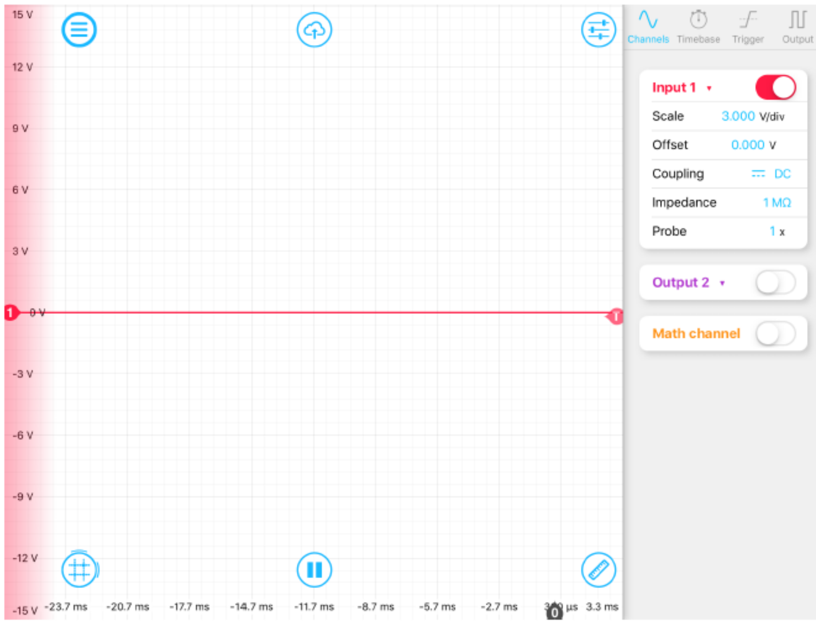
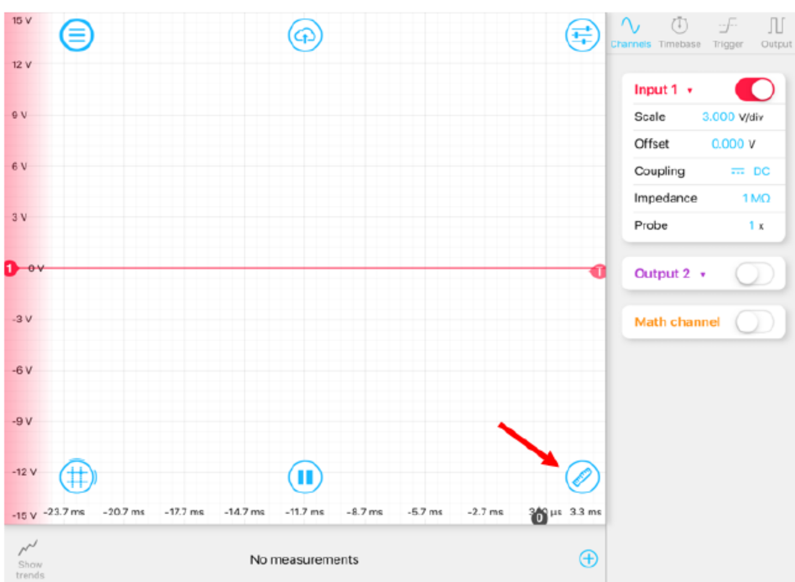
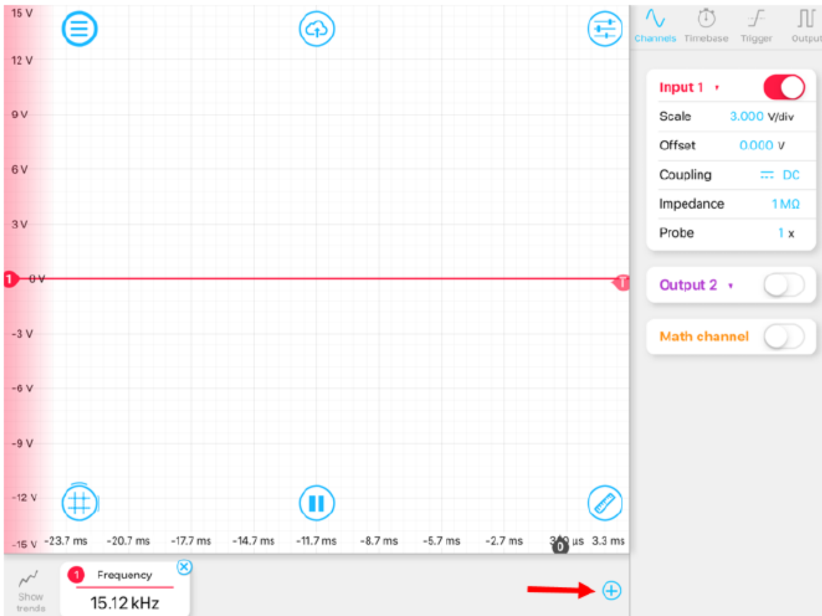


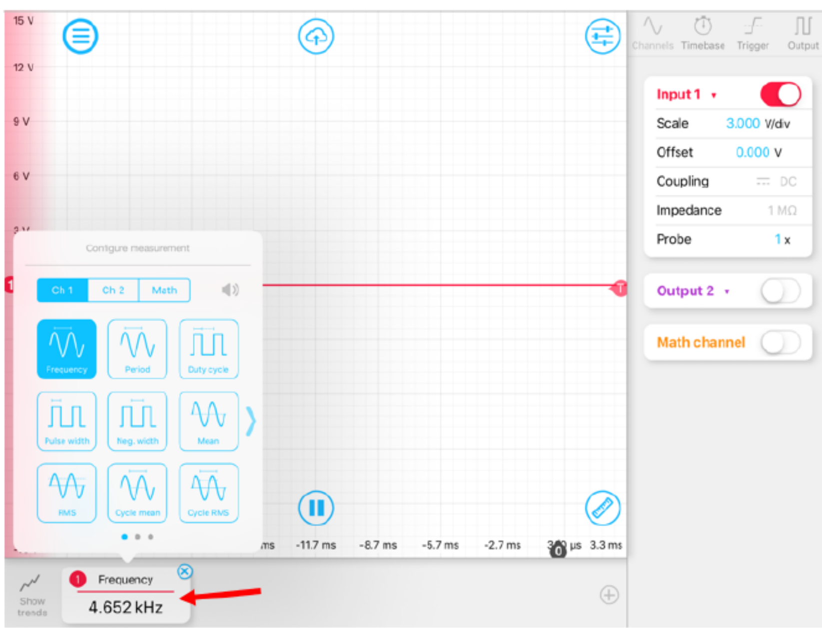
Figure 3

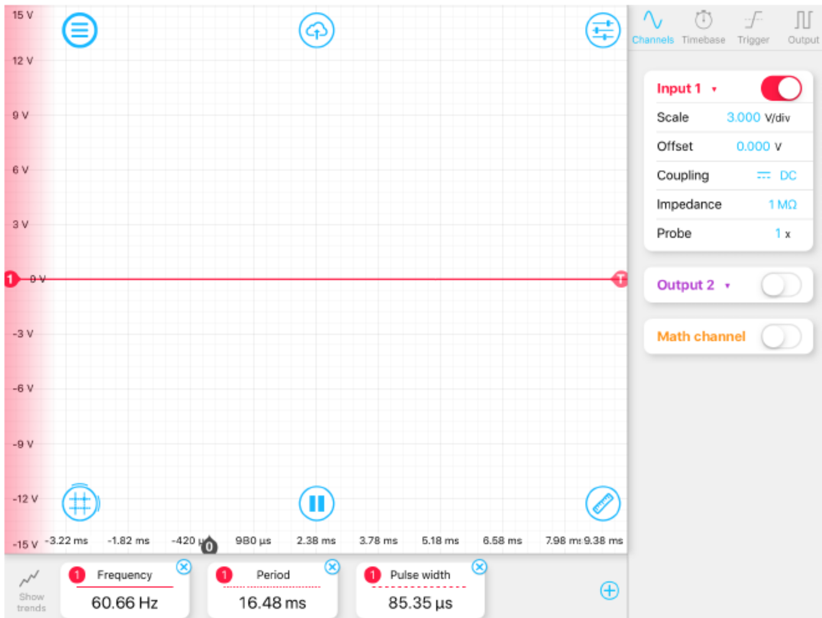
3. Open the measurements bar by pressing the ruler in the bottom right of your screen.



4. Add three measurement tools: Frequency, Period, and Pulse width. Press the "plus" sign within the measurements toolbox to add a new tool. Select the tool to change it. Repeat this for each of the four tools.







5. Turn on your robot and run the previously uploaded code.

6. What do you notice?

7. Comment out all of the code within ***loop()*** except for the portion that runs the motors at 75% duty cycle (*Motor_Forward(191,191)*; *delay(2000);*) and answer the following questions:

   (a) What is the frequency of the signal?

   (b) What is the period of the signal?

   (c) What is the pulse width of the signal (how long is the signal **HIGH**)?

   (d) Using these values, calculate the duty cycle.

   (e) Is this duty cycle what you expected?

   (f) Add a Duty cycle measurement tool on the display (reference step 4.3-4). Does the value match what you calculated?

   (g) How does the speed of the motors correlate to the duty cycle?

4 Feedback.

1. Was there any part of the lab that was confusing or unclear?

2. Is there any aspect of PWM that you still find confusing or would like more clarification on?

3. What would you change from this lab for next semester?