Use of the Arduino Platform in Fundamentals of Engineering

Dr. Djedjiga Belfadel, Fairfield University

Djedjiga Belfadel is an Assistant Professor in the Electrical and Computer Engineering department at Fairfield University. She obtained her Ph.D. degree from University of Connecticut in 2015, in electrical engineering. Her interests include embedded systems, target tracking, data association, sensor fusion, machine vision, engineering service, and education.

Ms. Marcia Arambulo Rodriguez, Fairfield University

Marcia Arambulo Rodriguez is the Assistant Dean of the School of Engineering at Fairfield University. She received her M.S. in Management of Technology from Fairfield University in 2015 and a B.S. in Electrical Engineering from Universidad Mayor de San Andres in La Paz, Bolivia.

Dr. Michael Zabinski, Fairfield University
Dr. Ryan Munden, Fairfield University

Dr. Ryan Munden teaches engineering at Fairfield University. He received his PhD in Applied Physics from Yale University and a BS in Physics from Stetson University. His areas of interest include semiconductor nanowires, nanotechnology education, first-year engineering initiatives, and engineering service, outreach, and education.

Use of the Arduino Platform in Fundamentals of Engineering Course

Djedjiga Belfadel, Marcia Arambulo Rodriguez, Michael Zabinski, Ryan Munden, James Cavallo

Abstract:

This Complete Evidence-based Practice paper outlines the benefits of incorporating Arduino activities into a first-year engineering class for students majoring in electrical engineering, bioengineering, undeclared engineering and mechanical engineering at Fairfield University. Arduino is an open-source hardware platform that has recently gained a wide following among students for its ease of utilization to build with it exciting interactive projects.

Teaching first-year engineering classes is a challenge given the importance of these classes in providing a solid foundation in engineering education. The traditional highly demanding first-year engineering curriculum which focuses primarily on engineering thinking and problem solving with a limited content of basic skills is not inspiring to incoming students and can become overwhelming. For this reason, we will use the Arduino platform in this first-year engineering class to add more hands-on activities to the curriculum and bring the joy of achievement for students.

Our goal is to add excitement to the Fundamentals of Engineering course by providing a hands-on and fun-to-learn environment designed to expose students to tools that will lead them to critical thinking, innovation, energy awareness, and problem-solving skills which will enable them to become part of a very competitive workforce in the future. Project-based courses attempt to resolve the tension between providing training in the fundamentals of engineering and problem solving by motivating and engaging students. Active learning in first year engineering courses is believed to improve retention and appeal to a diverse population of students [1].

We report on a two-year study using the Arduino platform to build a series of projects. We have used Arduino platform projects in two-course instances and present survey results describing student responses to learning with Arduino.

1. Introduction

Multiple active learning projects and hands-on activities are incorporated in the Fundamentals of Engineering course. This course is a required course for first-semester engineering students in this university. A key aspect of this course is the heterogeneous nature of the students taking the course, namely, electrical, bioengineering, undeclared and mechanical engineering students. This Evidence-based Practice paper describes the development and implementation of multiple Arduino activities and the results of the survey administered at the completion of these activities.

Arduino is used in many projects and applications from everyday objects to complex scientific instruments. The Arduino software is easy to use by first-year students, yet flexible enough for advanced classes. Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards read inputs, which are turned into the output.

Arduino simplifies the process of working with microcontrollers, and offers advantages over other systems:

- Inexpensive - Arduino boards are relatively inexpensive compared to other microcontroller platforms.
- Cross-platform - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems.
- The Arduino Software (IDE) is easy-to-use for beginners, yet good enough for advanced users.
- Open source and extensible software - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries.

- Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license.

2. Arduino Uno Platform

There are a wide variety of Arduino boards [2], in this class we used the Arduino Uno which is a high-performance 8-bit RISC-based microcontroller. The Arduino Uno has 14 digital input/output pins, 6 analog inputs, a resonator, a USB connector, a power, and a reset button. It has everything needed to support the microcontroller.

a) **Power:** The Arduino Uno can be powered via the USB connection or with an external power supply (AC-to-DC adapter or battery). Leads from a battery can be inserted in the Gnd and Vin pin headers of the power connector. The board can operate on an external supply of 6 to 20 volts.
b) **Input/Output**: Each of the 14 digital pins on the Uno can be used as an input or output. This can be done using the pinMode, digitalWrite and digitalRead functions [9]. They operate at 5 volts. In addition, some pins have specialized functions.
c) **Communication:** The Arduino Uno can communicate with a computer, or other microcontrollers. A Software Serial library allows for serial communication on any of the Uno's digital pins.
d) **Software:** Arduino environment is open-source and easy to use [1]. It runs on various platforms such as Windows, Mac OS X, and Linux. It is designed to introduce programming to people unfamiliar with software development, which makes it convenient for first year students. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and is also capable of compiling and uploading programs to the board with a single click. The Arduino integrated development environment (IDE) comes with a software library, which makes many common input/outputs operations much easier therefor the use or sensors.

3. Course Overview and Arduino Activities:

As the course has developed, it requires more resources to accommodate a growing number of incoming first-year engineering students while maintaining and improving the quality of delivery. To retain a consistent format, style, and outcomes the course is team taught to allow for continuous training of new faculty to teach the course, as well as to incorporate regular faculty feedback into the continuous improvement process. We can measure the effectiveness of the course improvements due to consistency in delivery achieved through close peer collaboration as team teachers. The course has scaled up through team teaching. The course also builds in year-over-year consistency by incorporating undergraduate student mentors and course assistants, who are trained in the assessment methods, after having taken the course, so that they can serve as effective Teaching Assistants (TA) in the class to the new first-year students. This has allowed us to offer four identical sections with four different instructors.

The course goals are: (I) Create a passion for engineering. (II) Develop an engineering mindset, problem-solving skills, and critical thinking. (III) Develop engineering professionalism. Each course outcome links to one of those goals. Those outcomes, which are also explicitly linked to various ABET defined student outcomes [10], are used to define and measure the success of each activity and learning module in the course. A process of continuous improvement of active learning techniques to achieve each course goal and demonstrate each outcome has resulted in the effective development of our first-year engineering students.

While the Arduino activities are intended to equip students with skills that can be used in the classroom setting, it is also intended to give students a glimpse into the outside world. The activities were designed to meet the following learning objectives:

Learning Objective 1: Students will explore the basics of using an Arduino microcontroller platform through hands-on simple activities.

Learning Objective 2: Students will gain a sense of curiosity and be motivated to explore more engineering projects.

4. **Arduino lecture organization:**

The Arduino activities were conducted during lecture meeting time, taught by the instructor and the teaching assistant, who are 2nd to 3th-year students, both males, and females, from School of Engineering.

The lecture consisted of a short introductory theoretical part, in which the students were taught the basics of Arduino and its programming language, and a hands-on section, in which 2 mini-projects were first covered by all the students, followed by a choice of one or two extra mini-projects (in the case of advanced students). The students were introduced to the open-source hardware philosophy, and general aspects about Arduino's make of, models and characteristics were discussed.

The fundamentals of Arduino programming language were then explained briefly, covering only those variables, functions and structures that were needed to understand the code of the mini projects planned.

All the needed parts for the Arduino examples were supplied in bags (one bag for two students) (Arduino Uno, breadboard, USB cable, electrical wires, LEDs, resistors, buzzer, capacitors, thermistors, and photoresistors) were supplied, without providing an entire kit.

Providing a starter kit for each student which include all of the basics (Arduino, a breadboard, cables, LEDs, resistors and pushbutton switches, motor) and other things that may help the student build some really fun things (temperature, flame, infrared sensors; light-dependent resistors; a stepper motor; seven-segment display; an LCD display) would also be a desirable option.

Arduino integrated development environment (IDE) open-source software, which runs on Windows, Mac OS X and Linux, was used.

Prior to starting with the mini-projects, the students were guided through the installation of Arduino software (IDE) in their laptops (students were required to bring their own laptops devices to the classroom) and helped, by the teaching assistants, with the installation of the board drivers when necessary, and the library installation. A brief explanation of the toolbar buttons was provided, but the students managed to pick up the important concepts and how to use the IDE properly as they worked their way through the mini-projects.

Hands-on section: it is very interesting to see the results of a project worked on. Be it a blinking light, a sound, or a moving device. Students like to receive immediate feedback. Arduino activities bring joy of achievement, students appreciate something that blinks or moves in response to their efforts. This interest is what motivates students to design something of their own. Even though it is very difficult to keep students engaged in an activity full of new concepts that could become overwhelming.

Mini-project #1: Blinking LED:

The project will use an LED to produce a blinking light. this example showed the students the simplest thing you can do with an Arduino to see the physical output: it blinks an LED, turning it on and off for definite periods of time. This allowed them to learn the use of digitalWrite() and delay() commands. This project allows students to receive a basic understanding of Arduino. After installing the Arduino IDE software onto the device used, the program allowed students to write, view, and upload code to the Arduino Uno board. This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink

Testing: Try adjusting the speed of the blink by changing the delay values!

Output: Will produce a blinking light.

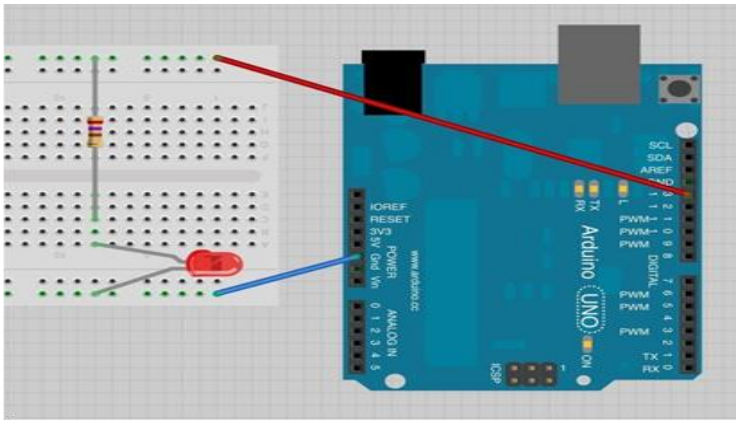


Figure 1

Mini-project # 2: Traffic Light:

Introduce students to basic computer programming, introduce students to basic circuit building, show how software and hardware can overlap and use one another.

The student will be able to further apply using LEDs with Arduino and create a traffic light simulation.

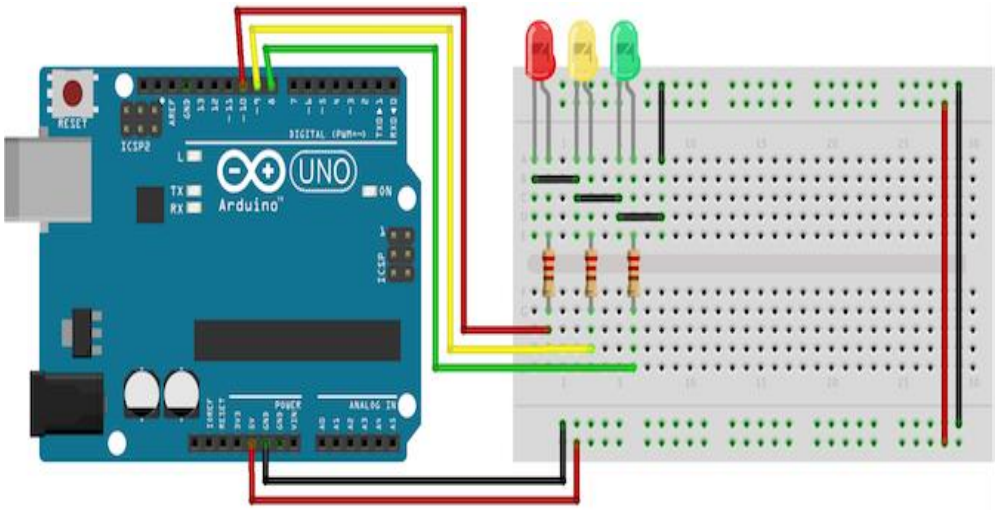


Figure 2

Project found at https://www.makeuseof.com/tag/arduino-traffic-light-controller/

Testing: Try adjusting the order of the lights by changing the delay times!

Output: The LED lights will form a traffic light like a simulation with a green, yellow, and then red light.

Mini-project #3: Tone Melody:

The project will use a piezo buzzer/speaker to play a little melody.

This project allows students to receive a basic understanding of Arduino. After installing the Arduino IDE software onto the device used, the program allowed students to write, view, and upload code to the Arduino Uno board. By using the code provided, students were able to create a small melody.

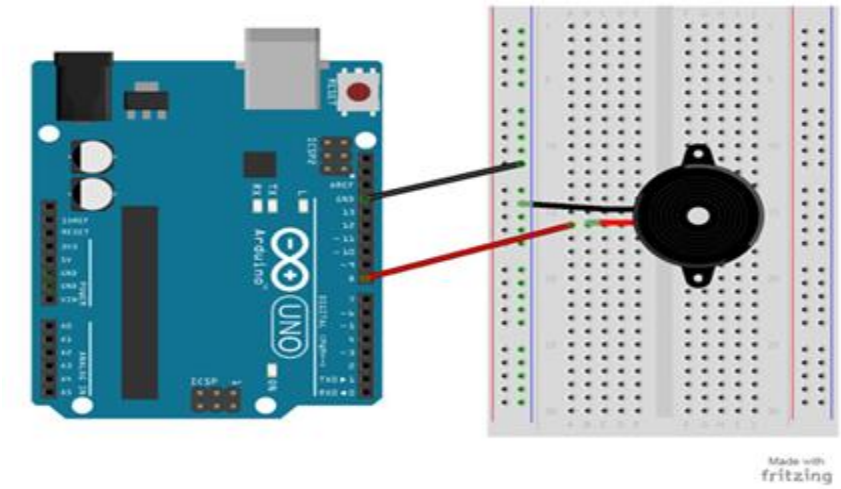


Figure 3

Project found at https://www.makerspaces.com/wp-content/uploads/2017/05/12-Tone-Melody_SMALL.jpg

Output: makes a short seven note music tone (da-da-da-da-da-da-da).

Testing: For different melodies go to https://circuitdigest.com/microcontroller-projects/playing-melodies-on-piezo-buzzer-using-arduino-tone-function to find more code options!

Mini-project #4: Distance Measurer:

The project will use an Ultrasonic Sensor to detect how far away an object is. Using the Arduino and Ultrasonic Sensor, the distance of an object will be measured and recorded. Students can test to see how accurate the Ultrasonic Sensor is by comparing the recorded measurements to their own measurements with a ruler.

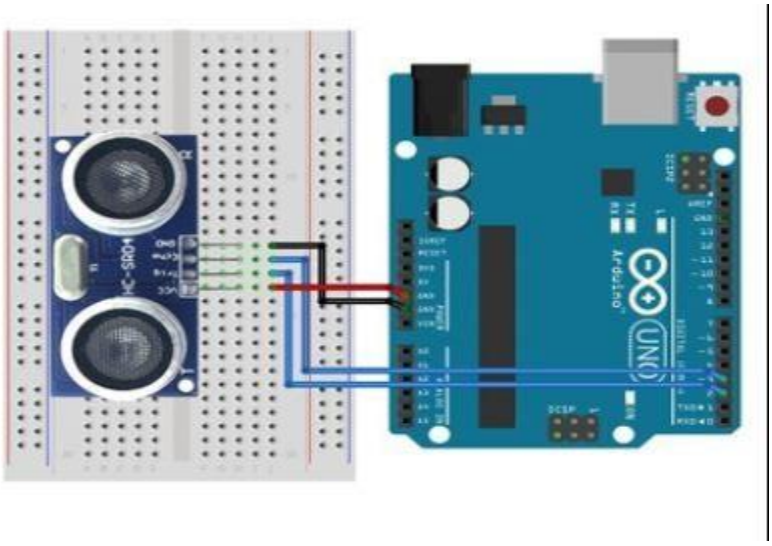


Figure 4

Project found at http://www.instructables.com/id/Distance-measurement-using-Ultrasonic-sensor-and-A/

Testing: Try testing out your project by laying a ruler down in front of the ultrasonic sensor and comparing your own measurements with the ones recorded in the serial monitor!

Output: The Serial Monitor will display the measurement detected by the Ultrasonic Sensor.

Mini-project #5: Temperature Measurement

The Temperature and Humidity Sensor will be able to detect changes in temperature around the sensor and the Serial Monitor will record these changes.

This project allows students to use a DHT11 Temperature and Humidity Sensor to keep track of humidity and temperature readings. After installing the Arduino IDE software onto the device used, the program allowed students to write, view, and upload code to the Arduino Uno board. By using the code provided, students were able to create a small temperature sensor.

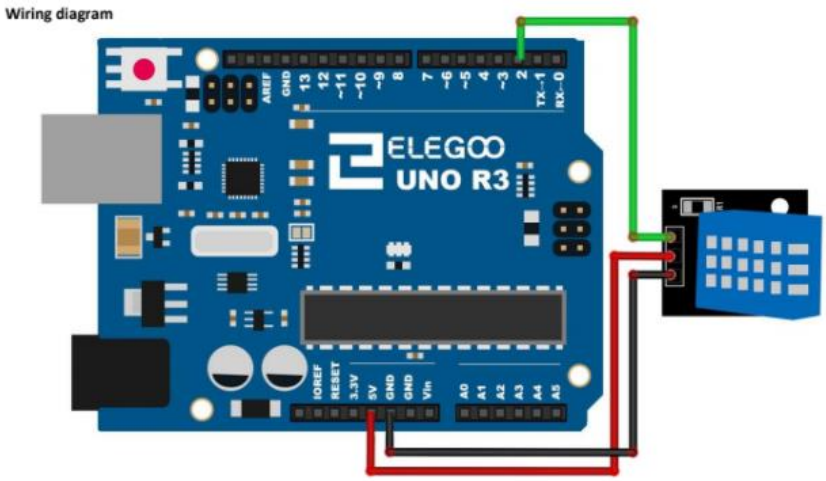


Figure 5

Project found at [Eligo Super Starter Kit for UNO V1.0.2017.7.9]

Testing: There are several different ways to change the temperature around the temperature sensor. Try the ones listed below and then see if you can come up with different ways on your own!

Cover the sensor with two hands

Below a hair dryer at the temperature sensor (be careful not to add to much heat to avoid melting the temperature sensor!)

Blow at the sensor with your mouth

Place a cold-water bottle in front of the sensor

Output: Changes in temperature will be recorded through the Serial Monitor.

Student Feedback:

Data was collected in Fall 2019, via Blackboard, for all the four sections of students taking Fundamental of engineering course.

Achievement of Arduino activities outcomes has been tracked through quantitative assessment for continuous improvement. The results are summarized as follow:

- 72% of students agreed that the hands-on experience has caused their interest in engineering to increase.
- 60 % indicated that the hands-on experience has caused their confidence in their ability to succeed in Engineering to increase.
- 68% Indicated that this class has caused their understanding of engineering design to increase.
- 64% of students agreed that working with the Arduino microcontroller has caused their interest in programming to increase.
- 41% agreed that the in-class exercises, such as programming, working with breadboard circuits have caused their motivation to study math, physics, and chemistry to increase.

As a conclusion outcome attainment can be said satisfactory, with more than 60% of the students agreed that after taking this class they are more likely to continue to pursue a degree in engineering.

In addition to the quantitative feedback from the student survey, we also collated anecdotal comments from the team feedback sessions. These sessions served as miniature focus groups with students to garner input for future course improvements. Students were explicitly asked to discuss the Arduino activities and other course assignments with the instructor in the feedback session, to provide any suggestions they liked. The instructors will consider these feedback items for the next iteration of the course. As a response to one of the survey questions: what is the best aspect of the Arduino activity – the one thing that should not be changed?
A few comments stood out as themes or particularly impactful,

``Great new thing to get a brief introduction to. Activities were great and gave a small insight. Good experience``
`` It was very informative and helpful looking from the word documents posted on Blackboard to follow the instructions in order to do the activity right.''
`` Was a good learning experience because it is something that I have struggled to learn in the past and something I have disliked. ``
``This activity has been anticipated by myself, and I am very happy to have learned more about the Arduino Uno. ``

``The best aspect is that we were given materials and experienced trial and error with the wiring. We learned to learn from our mistakes and avoid repeating them. ''

Student feedback regarding the Arduino activities, in summary, was that the hands-on activities were very interesting and increased the student's motivation and curiosity to explore more projects and brought some joy of achievement.
Combined with the fact that 72% of participants agreed that the hands-on experience has caused their interest in engineering to increase. these comments imply that these activities met the learning objectives of students exploring the basics of using an Arduino microcontroller platform through hands-on simple activities and gaining a sense of curiosity to explore more engineering projects.
A handful of students found the assignments to be too easy and wanted more challenging activities.
Also, students request to incorporate more Arduino projects, cover the Arduino early on the semester, and provide more detailed instructions about programming the board on advanced activities.
A few suggestions stood out.

``Incorporate more advanced exercises.'

``Increase the time we spend working with the Arduino. Work with it in more classes.''

``it would be helpful to get a brief understanding of how to do it before getting started. ''

``If we could do it individually after do it as a group.''

We will work to provide additional instruction on how to program and debug the code, as well as a more explicit description of those expectations. As this is the students' first exposure, it is not surprising that there was some confusion.
All student feedback, in addition to faculty observations, will be considered for the improvement of next year's course.

Conclusion
Arduino has many advantages: it provides all basic functionalities of a microcontroller at affordable price; it easily connects to a wide variety of sensors; literature on interesting projects is abundant. We believe that exposing students early on to the use of Arduino will motivate them to explore more engineering projects.

The results from the initial use of the Arduino activities were encouraging and highlighted numerous areas for additional consideration related to the activities of the Fundamentals of Engineering course. Students' feedback was positive and shows that student opinions greatly support the new Arduino activities incorporated in the course and engineering design practice. Results of this study show continued success in achieving course and Arduino activities outcomes.
We believe that the experiments reported here would help our colleagues who work on course development at different universities. Further details may be provided per request.

References

[1] R. Munden, M. Arambulo, D. Belfadel, and M. Zabinksi. Assessment of Critical Thinking and Design practices in Fundamentals of Engineering. ASEE Annual Conference and Exposition, Salt Lake City, Utah. June 2018.
[2] Nathaniel Bird. Use of The Arduino Platform for a Junior-Level Undergraduate Microprocessors Course. ASEE Annual Conference & Exposition, Number Ac 2011-437, June 2011.
[3] ABET student outcomes. http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2017-2018/#outcomes
[4] M. Eodice, A.E. Geller, and N. Lerner, The Meaningful Writing Project. Logan, UT: Utah State Univ. Press, 2017.
[5] S. Pearlman and D. Carrillo, The Critical Thinking Initiative. www.thecriticalthinkinginitiative.org (2017).
[6] B. Wood and A. Ganago, Using Arduino in Engineering Education: Motivating Students to Grow from a Hobbyist to a Professional. ASEE Annual Conference and Exposition, Salt Lake City, Utah. June 2018.
[7] Binary check sheet rubric, ABET Symposium, Baltimore, MD, 2017.
[8] Wiggins, G. and J. McTighe, Understanding by Design (Expanded Second Edition). Alexandria, VA, USA, Association for Supervision & Curriculum Development (ASCD).
[9] Siegel, C. Putting the Pieces Together: Linking Learning Outcomes, Assessment, and Curriculum. Center for Academic Excellence: Summer Institute on Integrative Learning. Fairfield University.
[10] Engineering Accreditation Commission. (2011). Criteria for Accrediting Engineering Programs. Baltimore, MD: ABET
[11] Handelsman, J., Miller, S. & Pfund, C. Scientific Teaching. New York, NY: W.H. Freeman.

Appendix A – Arduino Activities Student Survey

Arduino Rubric Fall 2018	I understand that my choice to complete this survey will not affect my grade in EG 31.	
True/False		
	Answers	*Percent Answered*
	TRUE	97.26%
	FALSE	1.37%
	Unanswered	1.37%
True/False	**I am voluntarily participating in this survey.**	
	Answers	*Percent Answered*
	TRUE	95.89%
	FALSE	2.74%
	Unanswered	1.37%
Multiple Choice	**Overall, this activity has increased my interest in engineering.**	
		Percent Answered
	Strongly agree	16.44%
	Agree	53.42%
	Neither agree nor disagree	20.55%
	Disagree	1.37%
	Strongly disagree	2.74%
	Unanswered	5.48%
	Sum of Agree and Strongly Agree	**69.86%**
Multiple Choice	**I am now more likely to continue to pursue a degree in engineering.**	
		Percent Answered
	Strongly agree	21.92%
	Agree	41.10%
	Neither agree nor disagree	27.40%
	Disagree	4.11%
	Strongly disagree	0.00%
	Unanswered	5.48%
	Sum of Agree and Strongly Agree	**63.02%**
Multiple Choice	**The hands-on experience has caused my interest in engineering to**	
		Percent Answered
	Increase a lot	24.66%
	Increase	47.95%
	No change	20.55%
	Decrease	1.37%
	Decrease a lot	0.00%
	Unanswered	5.48%
	Sum of Agree and Strongly Agree	**72.61%**
Multiple Choice	**The hands-on experience has caused my confidence in my ability to succeed in Engineering to**	
		Percent Answered
	Increase a lot	16.44%
	Increase	43.84%

	No change	32.88%
	Decrease	1.37%
	Decrease a lot	0.00%
	Unanswered	5.48%
	Sum of Agree and Strongly Agree	**60.28%**
Multiple Choice	**This activity has caused my understanding of engineering design to**	
		Percent Answered
	Increase a lot	12.33%
	Increase	56.16%
	No change	26.03%
	Decrease	0.00%
	Decrease a lot	0.00%
	Unanswered	5.48%
	Sum of Increase and Increase a lot	**68.49%**
Multiple Choice	**Working with the Arduino microcontroller has caused my interest in programming to**	
		Percent Answered
	Increase a lot	16.44%
	Increase	47.95%
	No change	26.03%
	Decrease	2.74%
	Decrease a lot	1.37%
	Unanswered	5.48%
	Sum of Increase and Increase a lot	**64.39%**
Multiple Choice	**The in-class exercises, such as, working with breadboard circuits has caused my motivation to study math, physics and chemistry to**	
		Percent Answered
	Increase a lot	13.70%
	Increase	27.40%
	No Change	50.68%
	Decrease	0.00%
	Decrease a lot	0.00%
	Unanswered	8.22%
	Sum of Increase and Increase a lot	**41.10%**

Appendix B: End-of-term survey

This survey designed to measure your confidence, career interest, and attitude toward the ARDUINO activities covered in EG 31. Your answers to these questions will not affect your grade and will not be associated with you personally in any way. This information will be used to help us improve the class and to share what we have learned from this class with our colleagues at Fairfield University and other universities.

☐ I understand that my choice to complete this survey will not affect my grade in EG31.

☐ I am voluntarily participating in this survey.

Please circle the word or phrase that best completes the opening statement, or that expresses your agreement with the opening statement.

1. **Overall, this class has increased my interest in engineering.**

Strongly disagree disagree neither agree nor disagree agree strongly agree

2. **Because of my experience in this class, I am more likely to continue taking courses toward a degree in engineering.**

Strongly disagree disagree neither agree nor disagree agree strongly agree

3. **The hands-on experience in this class has caused my interest in engineering to**

Decreased a lot decreased not changed increased increased a lot

4. **The hands-on experience in this class has caused my confidence in my ability to succeed in Engineering**

Decreased a lot decreased not changed increased increased a lot

5. **This class has caused my understanding of engineering design to**

Decreased a lot decreased not changed increased increased a lot

6. **Working with the Arduino microcontroller has caused my interest in programming to**

Decreased a lot decreased not changed increased increased a lot

7. **The in-class exercises, such as programming, working with breadboard circuits have caused my motivation to study math, physics and chemistry to**

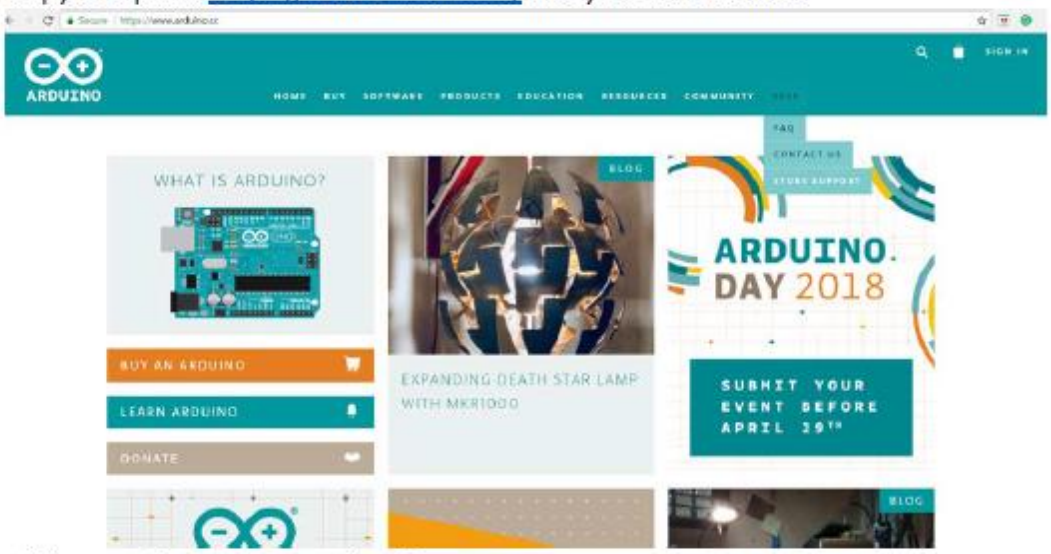
Decreased a lot decreased not changed increased increased a lot

8. **If the instructor were going to change one thing to improve the course, what would that be?**
9. **What is the best aspect of the course – the one thing that should not be changed?**
10. **Other comments**

Appendix C: How to download Arduino

ARDUINO WORKSHEET 0.5 : HOW TO DOWNLOAD ARDUINO

1. Copy and paste https://www.arduino.cc/ into your web browser

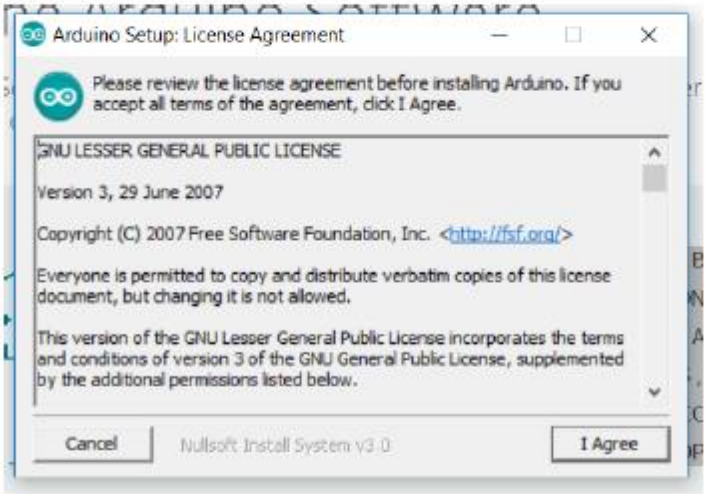


2. Click on **Software -> Downloads**
3. Scroll down to the section titled "Download the Arduino IDE"

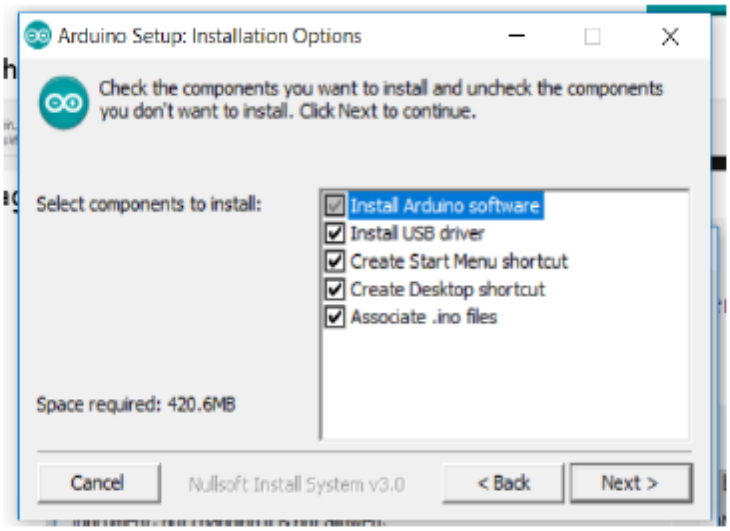
4. Select the operating system that aligns with your computer
5. Click on **Just Download**

6. Open the file after it finishes downloading.

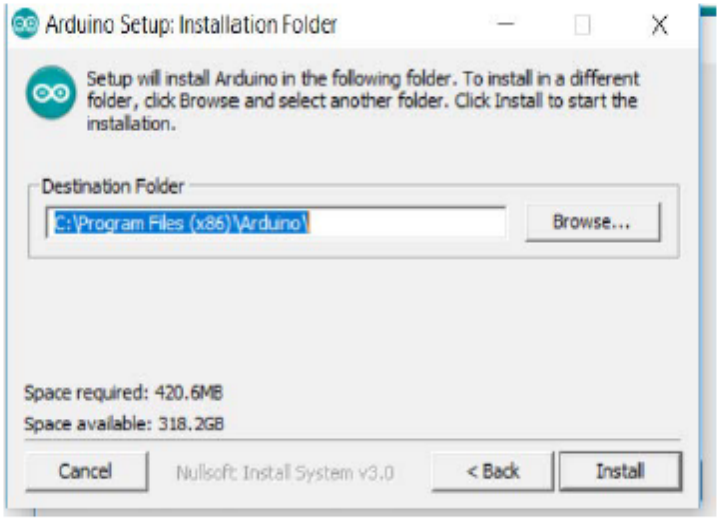
7. The software will ask "Do you want to allow this app to make changes to your device", click **Yes**
8. Click **I Agree**



9. Click **Next**



10. Assign a location for the Arduino files to be saved in and then click **Install.**



11. Finalize the downloading process by clicking **Close.**
12. The software is now downloaded to your device and you can begin working on projects!

Appendix D: How to Add New Libraries

Arduino Worksheet 1: Adding New Libraries to your IDE

What is an IDE?

An IDE (or Integrated Development Environment) is a program that runs on your computer, used to write and upload computer code to the physical board.

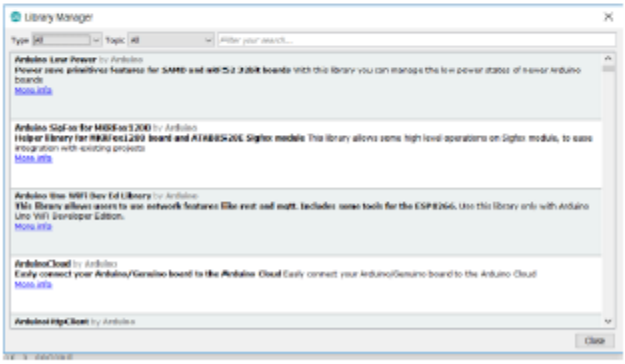
What is a library?

Libraries are files written in C or C++ (.c, .cpp) which provide your sketches with extra functionality (e.g. the ability to control an LED matrix, or read an encoder, etc.).

How do I add a library to my IDE?

1. Once in the Arduino program, click on **Sketch**

2. Click on **Include Library**
3. Click on **Manage Library**
4. Type in the library necessary for your project



5. Click **More Info**
6. Click **Install**