

Introduction to Arduinos and Programmable Micro-Controllers

1. Obtain an Arduino Uno and a USB cable. Then, go to the Arduino Homepage:
<http://arduino.cc/en/Guide/HomePage>
2. Click on the Learning Menu and select the “Getting Started” option. Click on the word “[Introduction](#)” on this page to get a brief summary of what an Arduino is.
3. Step back to the “Getting Started” page in your browser. Under “Install the Arduino Software”, click on the “[Windows](#)” option. Complete Steps #5-#9 to load your first program (programs are called sketches) to your Arduino. Steps #1 – Steps #4 should not be needed if using school computers in the lab as the computers already have the Arduino Integrated Development Environment (IDE) loaded on them.
 - a) Show your instructor your blinking light on Pin 13, the outcome of your last activity.
4. Again, step-back to the “Getting Started” page and click on “[visit the Foundations](#)”. This page is an excellent resource to be used in the future when you get stuck trying to do something with the Arduino. For now, read through the following things on this page:
 - a) Basics – Sketch
 - b) Programming – Variables
 - c) Programming -- Functions
5. Go to <http://www.ladyada.net/learn/arduino/index.html>. When doing each of the lessons on this website, be sure to attempt to answer all questions BEFORE looking at the answers. Do not race through these activities. Instead, go slow enough so that you can master the content.
6. Complete Lesson 2.
 - a) Show your instructor your Arduino running the program you create for Exercise 3.
7. Complete Lesson 3. You will need a breadboard, a red, blue & green LED, three 1K Ω resistors, and 22 Gauge solid-core jumper wire. Do all of the activities.
 - a) To turn in: a printed screen shot of your code for the last activity – the activity in which you turn each color on and then off in order. For full credit, included descriptive text for each program line in a fashion similar to the author of the online tutorial.
 - b) Show your pattern of LED lighting to your instructor after the final exercise.
8. Complete Lesson 4:
 - a) Be sure to look through the first linked tutorial called [A C++ tutorial](#) when you get to it. You can ignore the other two links.
 - b) Turn in a screen shot of your Ohm’s Law Program as well as the readout of your program on your Serial Monitor (both can be shown on the same screen shot).
 - c) At the point where it requests you to experiment with variable size, type and signedness, create a program that uses all three variable types (bit, int, and long) but that shows a rollover error when performing a calculation. Then, correct the program by showing that changing the sign type of the three variables corrects the calculations. Provide screen shots of the Serial Monitor both before and after the correction.
 - d) Finally, turn in a screen shot showing both your code as well as the Serial Monitor output for the final Drive Size project at the end of the lesson.
9. Complete Lesson 5: You will need a breadboard, five red LED’s, one 6mm tact switch (pushbutton), one 100 Ω resistor, five 1K Ω (any values from 300 to 2K Ω are probably okay), and one 10K Ω (any value from 5K Ω to 100K Ω is probably okay) resistor.

- a) Show your instructor your working circuit for the sketch associated with Figure 5.9 in which your single switch controls two LED's in pins 11 and 12, one being on and the other off at all times.
- b) Show your instructor your working circuit for Design Challenge Part II – the blinking bike light that toggles on and off with each press of the switch.
- c) Show your instructor your working circuit for Design Challenge Part II – the multi-mode bike light. Also, print out your sketch for this design. It shouldn't be an exact duplicate of the one provided in the lesson.