

L0: Hello, Arduino!

Due by 11:59 PM on your lab day, week of 2/18

Work in a group of FOUR people for this lab. Everyone in your group must all be enrolled for your same lab day, though not necessarily in the same section number on that day. You do not need to form a permanent lab group at this time; any four people will do.

Background: What is Arduino?

Arduino is one of the most popular open-source, single-board microcontrollers. But what does that really mean? Well, it means that it consists of a single board that provides all the circuitry you need to control electronic devices that you hook up to the board. And it means that information about the hardware and software that controls it is readily available. With it, you can build and connect electronic devices and control them with code that you write. Sweet!

As you work with your Arduino, remember that Arduino has a huge user community, so there are lots of resources available. In particular, the Arduino website¹ provides documentation, tutorials, and a nice FAQ page. Please also check out the Resources page on the COS 436 website.²

Below is a labeled diagram of an Arduino Uno SMD microcontroller. Use it to familiarize yourself with the basic components of the Arduino, and return to it throughout the assignment as needed. You don't need to memorize every component, but it is useful to have a general sense of what's going on on the microcontroller.

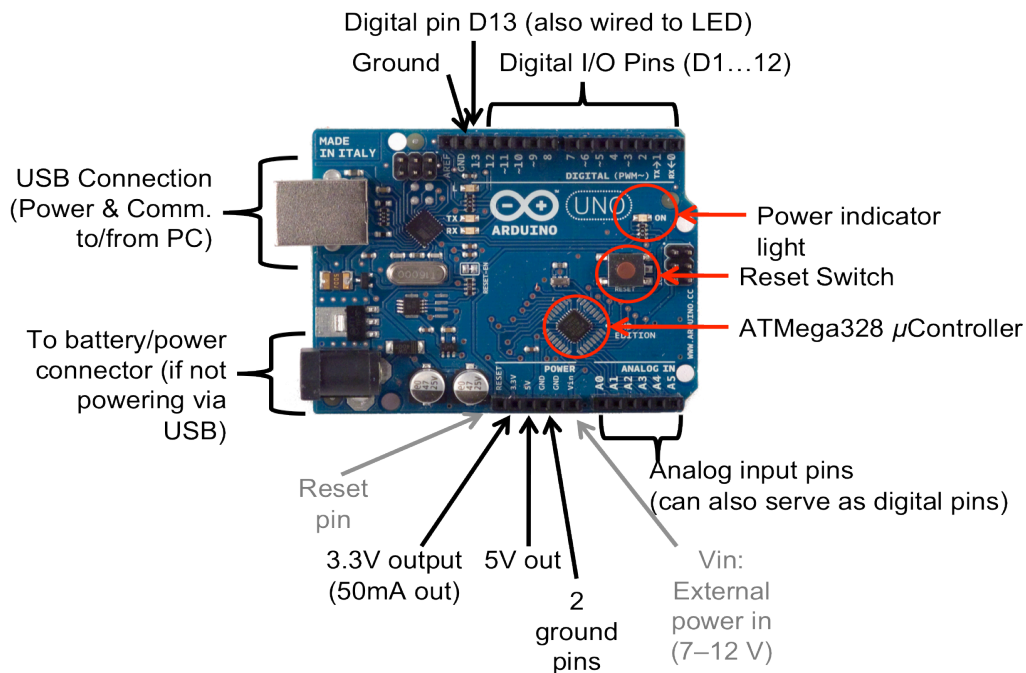


Figure 1: The parts of the Arduino that are most important for you to know about right now. You shouldn't have to use the parts labeled in grey.



¹ <http://www.arduino.cc/>

² <http://www.cs.princeton.edu/courses/archive/spring13/cos436/resources.html>

The purpose of this assignment is to familiarize yourself with the Arduino and have some creative fun with it. In the first part of the assignment, you will install the necessary software and set up your Arduino to control a blinking light. In the second part, you'll get fancy with the lights, setting up more than one and controlling the brightness of each. And in the third part you'll design and build your own diffuser for a system of lights.

How to Submit

All lab assignments will be “handed in” for grading using a combination of the following 3 mechanisms:


- 1) Demos to your peers. When you see the Demo Check icon , you should ask a classmate (someone enrolled in 436 but not in your lab group) to certify that the given component of your project works as required. Your classmate will sign your Demo Certification form, which you should print out from http://www.cs.princeton.edu/courses/archive/spring13/cos436/handouts/peer_demo_sheet.pdf
You will hand in these forms to the instructor at the beginning of the next lecture.
- 2) An online portfolio. The more creative and design-y parts of the assignments will be marked with the WordPress icon , and you will submit your work by creating an online portfolio at our course blog, <https://blogs.princeton.edu/humancomputerinterface/>.

Your submission should include images and—if possible—video. Phone- and webcam-captured videos and images are fine.

At the beginning of each lab, one member of your group will create a new post at <https://blogs.princeton.edu/humancomputerinterface>. To ensure your post can be easily found and graded, **please choose the appropriate category** (e.g., “Lab0”) from the right-hand menu when you create your post.

A note about privacy: Part of the motivation of using WordPress for your labs and projects is to make you famous, if you want to be! If you come up with a really awesome project, sharing it with the world through pictures, video, and code is a wonderful thing. By default, posts on WordPress are public and indexable by search engines, to help the world find your work.

On the other hand, we respect your need for privacy. You can change the visibility of any new post to “private” to make it viewable only by the instructors. If you have any concerns about privacy that are not addressed by this solution, come talk to us and we will make it work.

- 3) A Web form. When you see the Form icon , you should use the provided URL to submit answers to any lab questions. The form should be submitted by the lab due date. (Your submissions are time-stamped!) You will use this form to tell us the URL of your blog post, as well.

Academic Integrity

All code written in the labs must be your own work, except when explicitly stated otherwise in the assignment. If the lab tells you to use skeleton code, or encourages you to use a third-party library, **always acknowledge the use of other code in your blog writeup**. Provide a link to the sources of any third-party code, and use comments in your code to make it clear what is and is not your own original work. Even if you use someone else's online tutorial without copying their code, please acknowledge them in your writeup. Do not steal from other people's lab assignments, and do acknowledge any help or inspiration you received from your classmates.


In summary: **always err on the side of over-acknowledging your sources, and never open yourself up to even the perception of academic dishonesty!** And everyone will be happy.

And now... the lab!

Part 1: Blink (20 points)

1. Install the software necessary to control your Arduino, and set up a single blinking LED. If you have a Mac, visit <http://arduino.cc/en/Guide/MacOSX>, and if you're on a PC mosy on over to <http://arduino.cc/en/Guide/Windows>
2. Follow all the instructions on the page, which will guide you through setting up your Arduino and making the on-board light blink.
3. Now, follow the instructions at <http://arduino.cc/en/Tutorial/Blink> to make another LED blink (not just the one attached to your Arduino). Make this circuit with a resistor and LED, but **use a breadboard for your components** instead of connecting your board, resistor, and LED with magical invisible glue like in the picture. (Aren't sure how to use a breadboard? Check out <http://www.youtube.com/watch?v=zCX3Hr4ZsDg> and/or <http://eecs.vanderbilt.edu/courses/ee213/Breadboard.htm>)

(Clarification added 2/8/13: Go ahead and use a 330-ohm resistor, which will be found in your lab kits.)

4. Step back and admire. Then make a modification of your choice to Blink.ino to change the light behavior. For example, you might change the frequency of the blinking light. Feel free to get creative with it.
5.  Find a classmate and demo for them. **Your classmate should certify that your light blinks, and you have made some simple change to the light behavior from Blink.ino.** Fill out the peer demo certification form (Lab #0, Part #1, leave group number blank, and fill in your NetIDs).

Part 2: Burn (20 points)


1. Gather 3 LEDs: one red, one green, and one blue. Connect all three to your Arduino, using a separate digital pin for each. Do this on a breadboard. You'll want to use only digital pins with the ~ by their names (3, 5, 6, 9, 10, 11) so that we can control brightness using PWM (<http://arduino.cc/en/Tutorial/PWM>)
2. Write a script that controls the brightness of each LED. Feel free to experiment with different ways of controlling the lights. You might want to make them all blink randomly; or you might want to blink out some morse code; or perhaps you want to oscillate the brightness in some pattern. How you control the lights is up to you, as long as your code controls the brightness of each light in some capacity.

But wait, how do we control brightness, you ask? Especially using a digital output pin? Check out documentation at <http://arduino.cc/en/Reference/AnalogWrite> to learn about pulse width modulation and the analogWrite command.

For additional guidance, feel free to check out this tutorial from the Arduino site:

<http://arduino.cc/en/Tutorial/Dimmer>


That tutorial requires you also download and install Processing from processing.org, which is a nice idea anyway.

3.  Find a classmate and demo for them. **Your classmate should certify that you are controlling the brightness of the three lights individually.** Fill out the peer demo certification form (Lab #0, Part #2, leave group number blank, and fill in your NetIDs).


Part 3: Radiate (50 points)

For this part of the assignment, you will design a diffuser for an LED system. A diffuser is simply anything that alters the distribution of the light emanating from the light source. Any materials are fair game. Some

materials you might consider: foods, tinted glass, wax paper, etc. Your system may involve as many or as few LEDs as you like, and should make use of blinking or dimming in some way. This could be interactive (e.g., controlled using Processing like in the dimmer tutorial in Part 2), or not. But make it interesting, and have fun with it! (Clarification added 2/8/13: Please limit your use of electronics for this part to only those parts found in your lab kit.)

1. Before you start building or coding, complete at least three design sketches, on paper. Plan out the materials you will use, and how you will assemble the system. Take pictures (or scans) of these sketches. Cell phone photos are fine as long as the sketches are clearly visible in the photos.
2. Build your system!
3.  Document your system in a WordPress blog. Create a new blog post, then:
 - a. Add it to the “Lab0” category.
 - b. Include the following in your post:
 - i. The names of everyone in your group
 - ii. A short description of what you built, why, and your assessment of its success. You should mention what you liked about the final result, and you should indicate your opinion on what didn’t work, or what you might like to do differently. All this text should be **no more than one paragraph**.
 - iii. Photos of your sketches, and a short caption explaining what you did in each one
 - iv. Photos and/or video showing your final system in action, with short caption(s)
 - v. A list of parts used in your final system
 - vi. Instructions that would be sufficient for someone else to recreate your design exactly. (This should be no more than one paragraph.)
 - vii. Your entire source code for Part 3. If the code isn’t too long, copy-paste directly into your writeup, and format the code using either the <pre> tag (if you’re in the HTML editing interface) or using the “Preformatted” style, which you can access using the “kitchen sink” button, shown below. Or, if you have lots of code, you can upload it somewhere (e.g., Princeton WebSpace) and include a link to it.



4. Hit “Publish” to publish your blog post.
5.  Fill out the web form at <http://tinyurl.com/cos436lab0> and provide your group member NetIDs and the URL to your blog.
6. RETURN YOUR LAB EQUIPMENT TO THE CABINET WHEN YOU ARE DONE.

Grading Rubric:

Part 1 and 2: Pass/fail (Turn in 2 peer demo forms)

Part 3: Graded on:

- a) Completeness and quality of documentation: 5 points
 - 0 points: missing/incomplete
 - 3 points: documentation is there but missing key information, missing photos/video, or unclear, or an inappropriate length
 - 5 points: thorough enough for a novice to follow your instructions and re-create easily
- b) Documented design process and critical self-reflection: 5 points
 - 0 points: missing/incomplete
 - 3 points: documentation is there but missing some information, or not presented clearly, or an inappropriate length
 - 5 points: includes at least 2 sketches; includes thoughtful reflection on what worked and what didn’t; describes motivation or inspiration for your system
- c) Creativity & coolness: 5 points

COS 436 / ELE 469 Spring 2013
Last updated 2/8/13 10:40 AM

- 0 points: bare minimum, looks just like existing tutorials or previous parts of the lab
- 3 points: result differs substantially from existing tutorials, shows independent thought
- 5 points: Mind-blowingly awesome; better than anything the TAs would have come up with