

Lab 2-Blink an LED

Learning Outcomes:

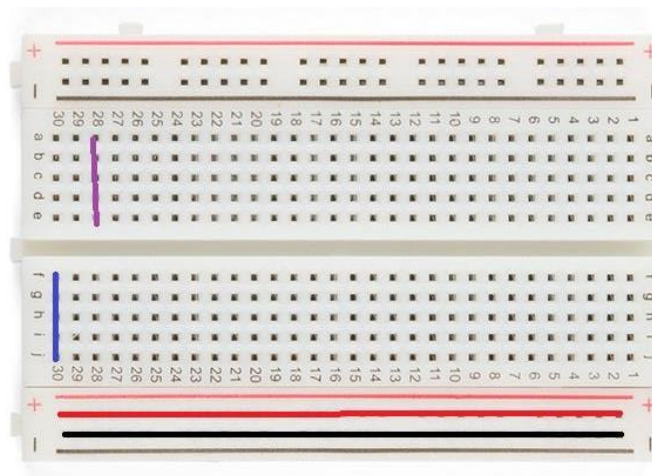
- Learn how to code Arduino Nano
- Learn prototyping with breadboards
- Learn how LED's work

Background:

When working with electronics we usually use a breadboard to prototype our circuit before permanently creating it on a printed circuit board. To finish this lab, you need to show that you can blink an LED with your Arduino Nano.

Before we start creating our circuit please pay attention to a few things:

First, notice how the breadboard is organized: the (-) and (+) rails are connected horizontally, but all other rails are connected vertically in groups of five. This is important to remember and take advantage of when prototyping.



Breadboard. Image courtesy of Newark.

Second, notice that in the Tinkercad diagram below one side of the LED is bent and the other is straight. This is because an LED is a polarized component (meaning that current only flows one way; if you put it in backwards, the LED won't turn on). The positive side (bent side) is called the anode, and the negative side (straight side) is called the cathode. Make sure the anode is connected to power, and the cathode is connected to ground.

Third, notice that the LED is in series with the 330-ohm resistor. The LED has almost no resistance, so if we put it directly between power and ground too much current would flow and

the LED would burn out. The resistor constricts the current to a much smaller value that the LED can handle.

Fourth, notice that there is a connection between ground on the breadboard and the ground (gnd) pin on the Arduino Nano. Without this connection there would not be a common ground between the board and the Arduino, and the circuit would not work.

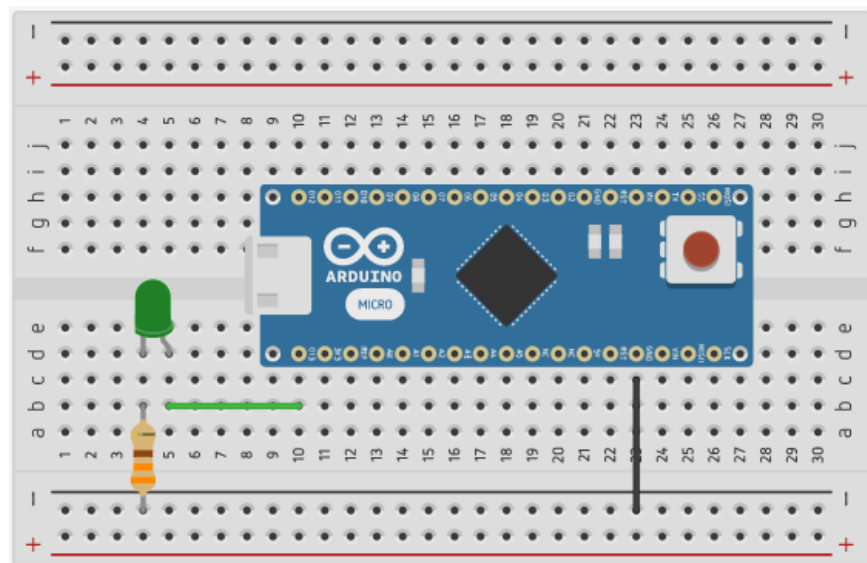
The objectives for today's lab are to:

- 1) build a circuit on a breadboard to blink an LED
- 2) install Arduino IDE
- 3) code our Arduino Nano to blink our LED.

Step 1 Blink LED Circuit Diagram:

Create the circuit shown in the picture below. (Make sure to push your Arduino Nano far enough into the board to make a connection, but be careful not to break any pins).

1. Place the Arduino in the middle of the breadboard bridging both sides
2. Place a wire from the ground pin (GND) on the Arduino Nano to the (-) rail on the breadboard
3. Place the LED somewhere to the side of the Arduino (don't place it on any of the same rails that Arduino Nano is on)
4. Place a wire from Pin13 (D13, look for the small white print by the pin) on the Arduino to the anode of the LED
5. Place a 330-ohm resistor (look up resistor band chart) between the - rail and the cathode of the LED



Circuit diagram from Tinkercad

Step 2 Downloading Arduino Software:

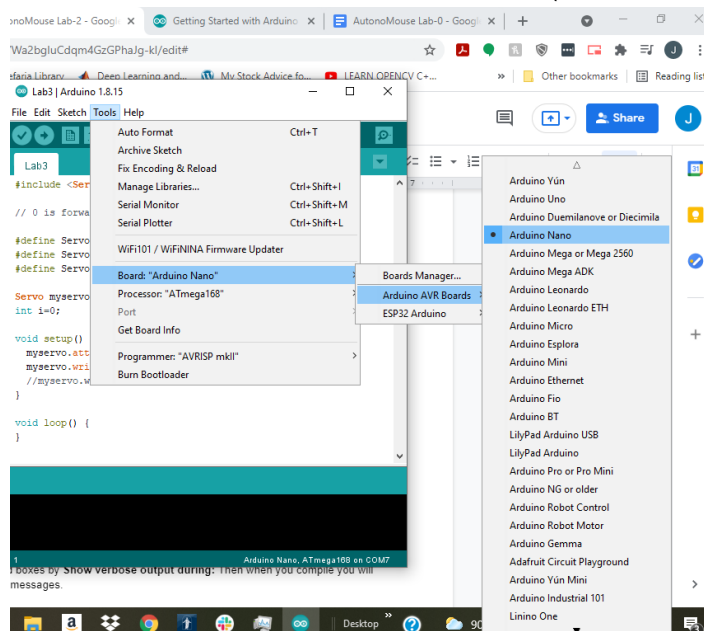
[How to Install Arduino Software \(IDE\) on Windows 10](#)

You are not required to learn much about coding in this course, however, you will still need to learn to download and use the Arduino integrated development environment (Arduino IDE). An IDE is software that helps you program the specific microcontroller/microprocessor you're using. The Arduino IDE is small and simple to use. Choose your operating system and follow the instructions to download Arduino IDE at [Getting Started with Arduino products](#). Scroll down and follow the instructions under [Install the Arduino Desktop IDE](#) header.

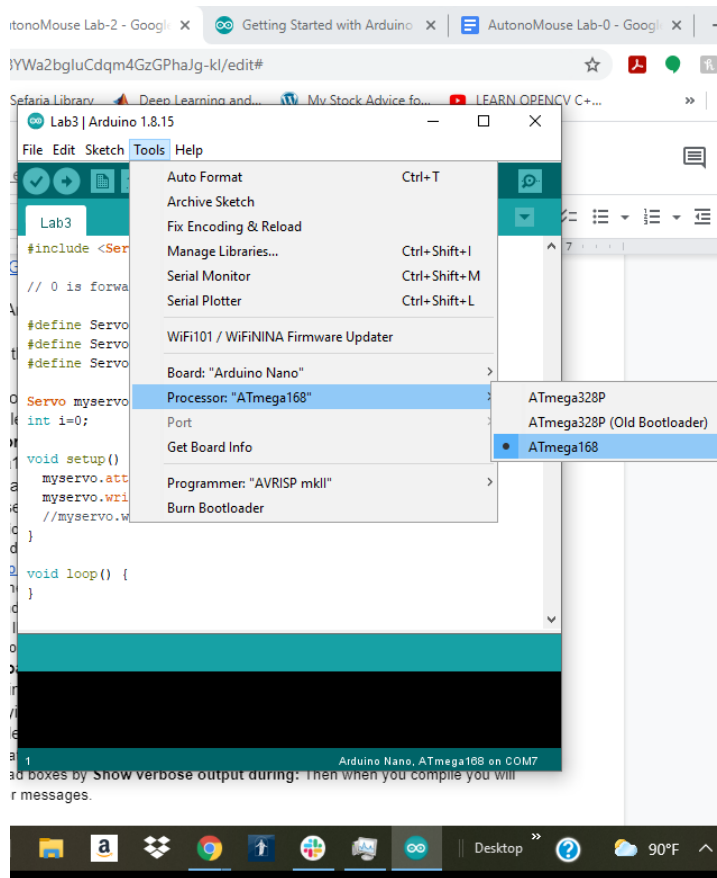
Once you've installed the Arduino IDE, copy and paste the code provided from GitHub into the IDE. [Code for blinking light](#).

Once you're done copying the code we will now try to get it to interface with your Arduino Nano.

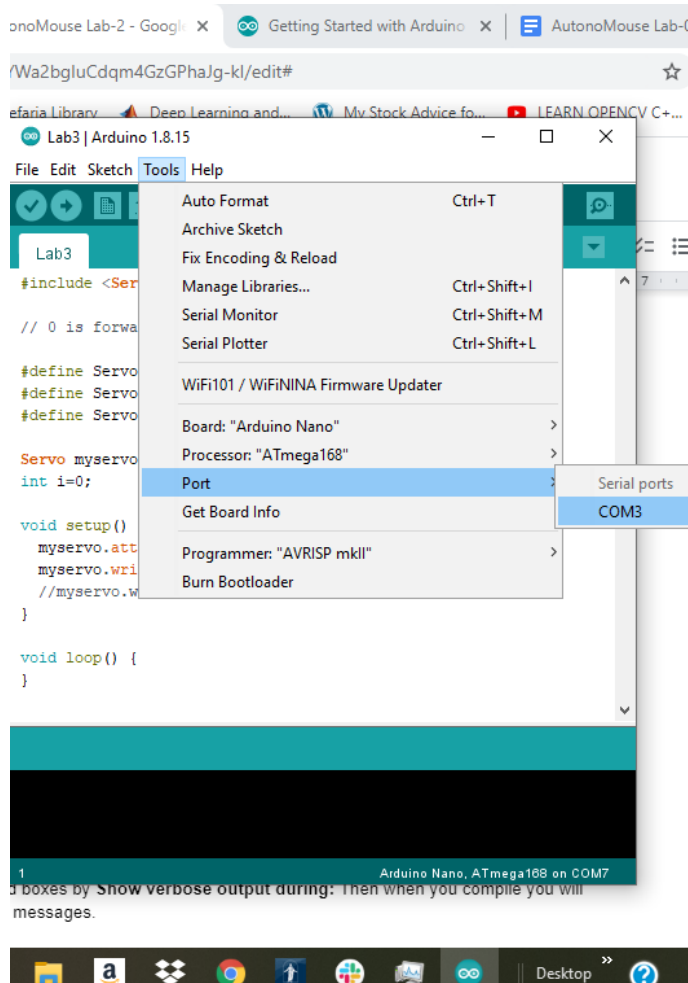
1. Open the IDE
2. Plug in your Arduino Nano via USB
3. In **Tools/Board:** select **"Arduino Nano"** (this tells the IDE what controller to expect)



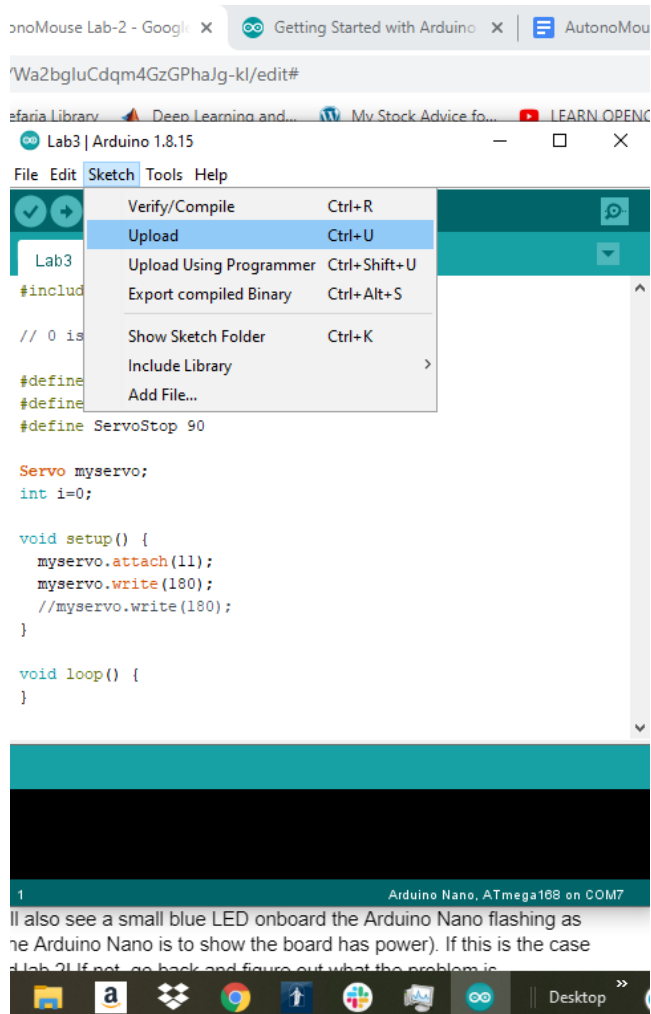
4. In **Tools/Processor:** select the processor that your Arduino Nano uses. I'm using a board with ATmega168. (if you get an error `"avrdude: stk500_recv(): programmer is not responding"` then change which processor you're using)



5. Under **Tools/Port** select which port on your computer the Arduino Nano is plugged into. Note: this might be tricky. My computer's USB driver wasn't compatible with Arduino and it wouldn't install the Arduino drivers which came with Arduino IDE so I found a CH340 Driver at <https://sparks.gogo.co.nz/ch340.html> that worked. First go to device manager (on windows) and look under ports for the port your Arduino is connected to. (An easy way to do this is to plug in your Arduino nano and see if a new port will pop up) If it does then you should be able to select which port in Arduino IDE. If not, check if an unidentified USB device came up instead. If that's the case then you need to install a compatible driver.



6. Select **Sketch/Upload** to upload the software onto the Arduino Nano. If it worked, your Arduino Nano should start doing what you told it to do. If not, you will see error messages saying that your code compiled incorrectly. You will need to fix whatever is causing the problem. Note: it's helpful sometimes to get the whole error message when you're debugging what went wrong. To change these settings, go to **File/Preferences** and select compilation and upload boxes by **Show verbose output during**: Then when you compile you will get the complete error messages.



Step 3 Blinking the LED:

If you did steps 1 and 2 correctly then you should see the LED turn on for a second and then turn off for a second. You will also see a small blue LED onboard the Arduino Nano flashing as well (the red LED onboard the Arduino Nano is to show the board has power). If this is the case congratulations! You finished lab 2! If not, go back and figure out what the problem is.

Bonus Questions:

How could you get the LED to go on for 2 seconds followed by 1 second off?