
ARDUINO

EVOLUTION OF ARDUINO

2005 2006 2007 2008 2009 2010 2011 2012 2013

In 2005, a group at Italy's **Interaction Design Institute Ivrea** developed Arduino as a low-cost, easy-to-use electronics platform for students and artists. It borrows its name from nearby watering hole **Bar di Re Arduino**. Since exploding onto the maker scene, Arduino has cultivated a flourishing community of inventors, engineers, and hackers dedicated to sharing code and developing hardware under an open-source banner.

Old-style RS-232 serial port rather than USB.

Designed to be built on a home-etched PCB.

Adds female pin headers, data transfer LEDs.

First board to bear the Arduino name.

Most advanced etch-it-yourself PCB design.

Atmel's 8-bit megaAVR microcontroller family is an Arduino signature.

ATmega8

First to ship with ATmega168.

The ATmega168 doubles on-board memory to 16KB, but is otherwise nearly identical to the ATmega8.

ATmega168

Large 28-pin Plastic Dual In-line Package (PDIP-28) for through-hole soldering.

Thin Quad Flat Package (TQFP) designed for surface-mount soldering.

Very thin Quad Flat No-lead (VQFN) package replaces leads with underside pads.

See-through contact pads for connecting conductive thread.

Underside pins for breadboard connection.

Designed for battery-powered wireless projects.

First board to use surface-mount processor.

Auto-selects power supply. First to ship with ATmega328.

Designed for quicker/easier installation.

Mini form-factor compatible.

On-chip memory doubles again to 32KB.

ATmega328

Replaces see-through contacts with button snaps.

The Mega hook Arduino to a new level, quadrupling on-chip memory to 128KB and more than tripling the total number of I/O and input pins in a significantly larger form factor.

ATmega1280

Official reference model for Arduino platform.

Emulates mouse and keyboard over USB.

Integrates USB controller into processor chip.

ATmega32u4

With the Mega2560, memory doubled again to 256KB. Though larger, the new form factor remains pin-compatible with the standard Arduino shield footprint.

ATmega2560

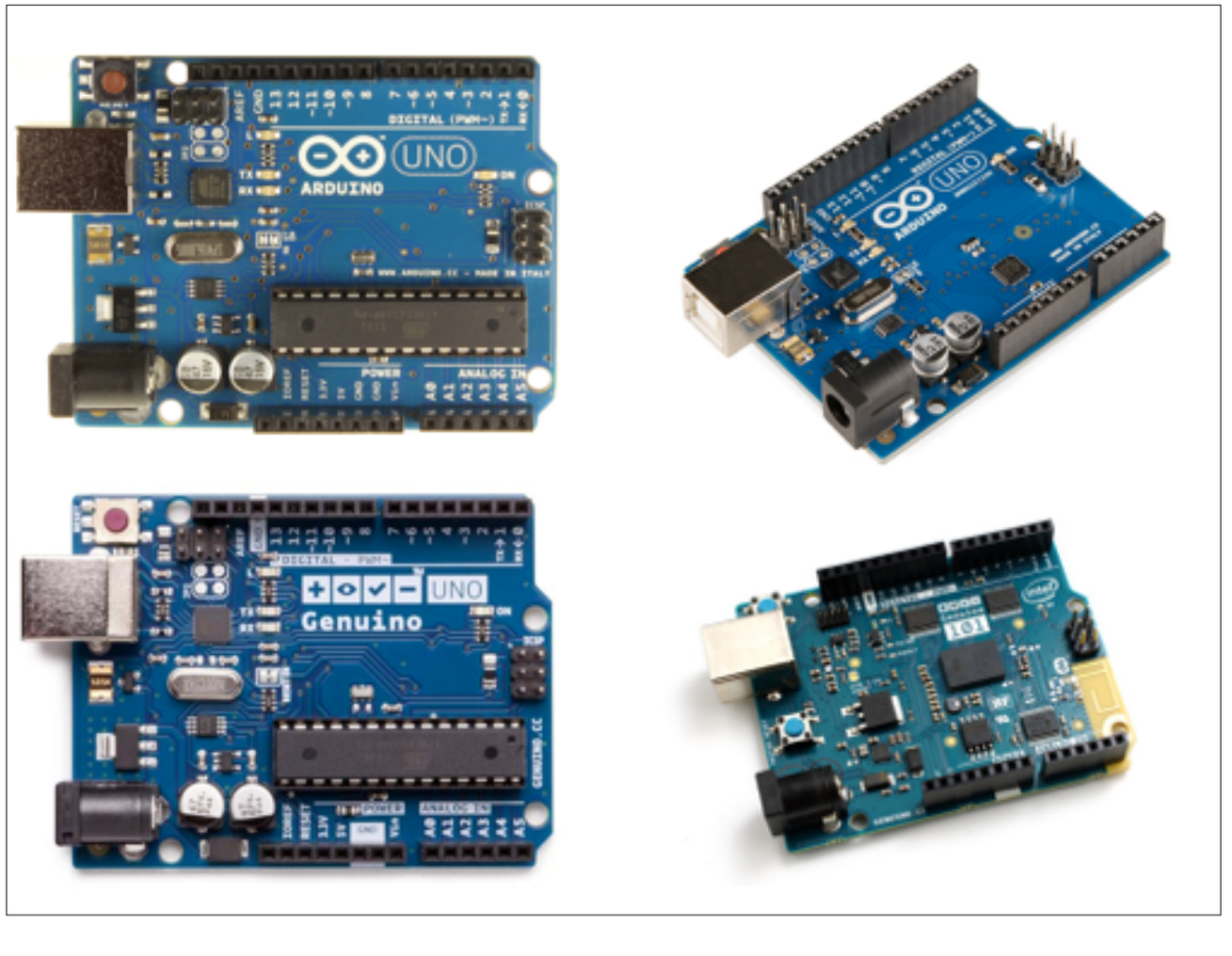
First dual-core model, combining ATmega32u4 with MIPS embedded Linux machine.

Mega update for use with Android Accessory Development Kit (ADK).

The Due marks Arduino's first departure from the AVR architecture. The ATSAM3X8E is an ARM Cortex-M3 processor with twice the memory and four times the clock speed of the ATmega2560.

ATSAM3X8E

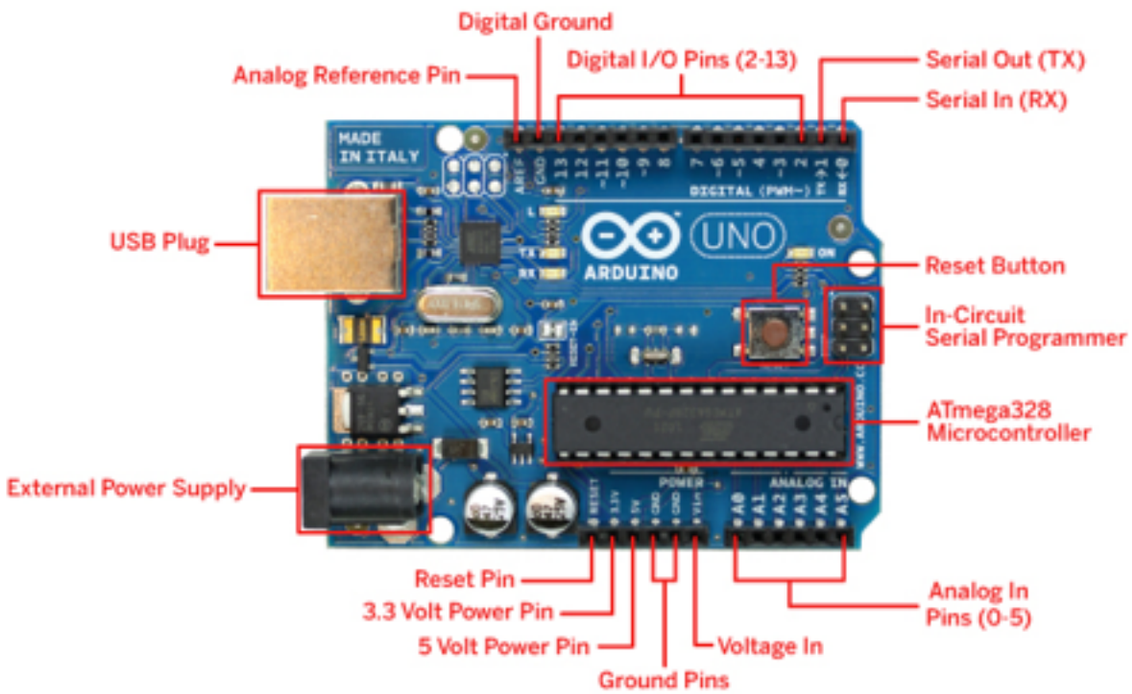
First Arduino to mount 32-bit processor. Runs at 3.3V rather than 5V.



Arduino & Genuino

Arduino open-source, label Arduino-compatible vs Arduino

Meet the arduino



https://www.element14.com/community/groups/arduino/blog/2014/03/28/getting-to-know-arduino-part-1-hello-world

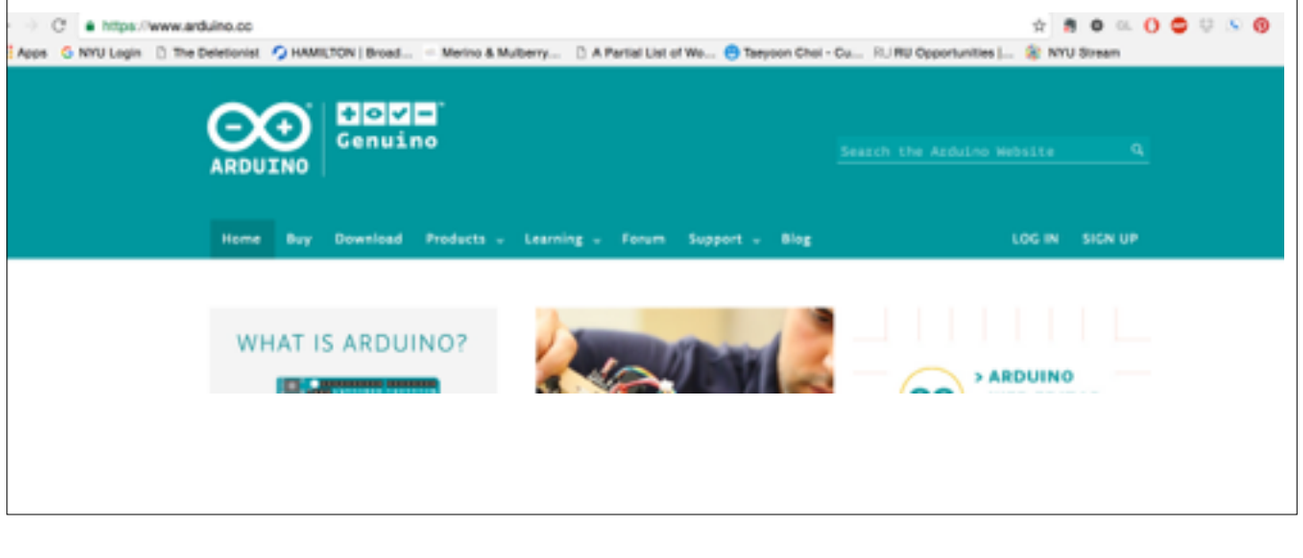
the size of a credit card, it has 14 digital IO pins and 5 analog, and runs off 5v. It can be powered either by the USB cable or up to 12v DC via a barrel jack.

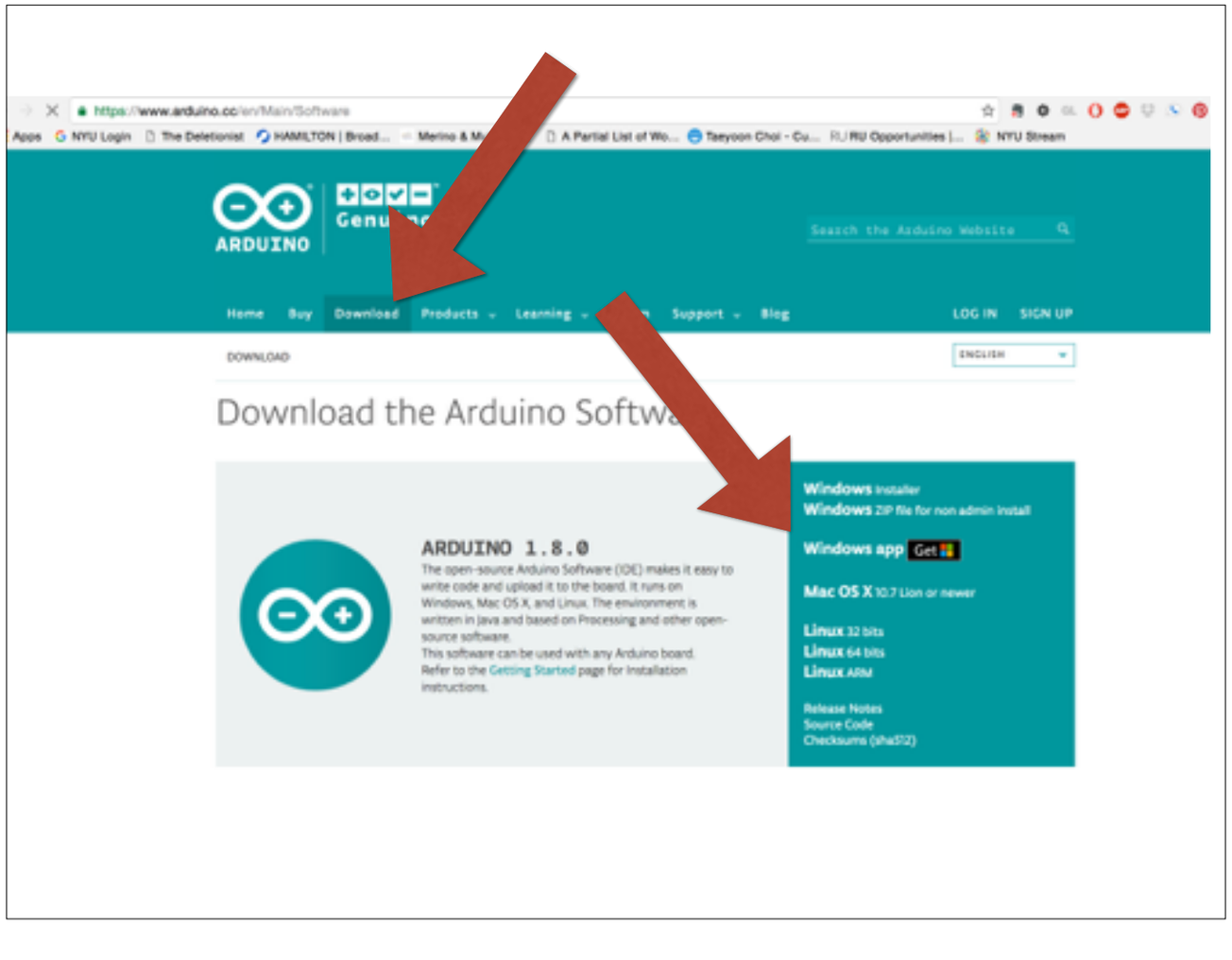
ATMEGA328P, with 32Kb of program memory, which is small by today's standards.

At some point you'll probably hit that limit, but as a starter device to learn from, the Uno is perfect.

https://www.element14.com/community/groups/arduino/blog/2014/03/28/getting-to-know-arduino-part-1-hello-world

Go to: www.arduino.cc



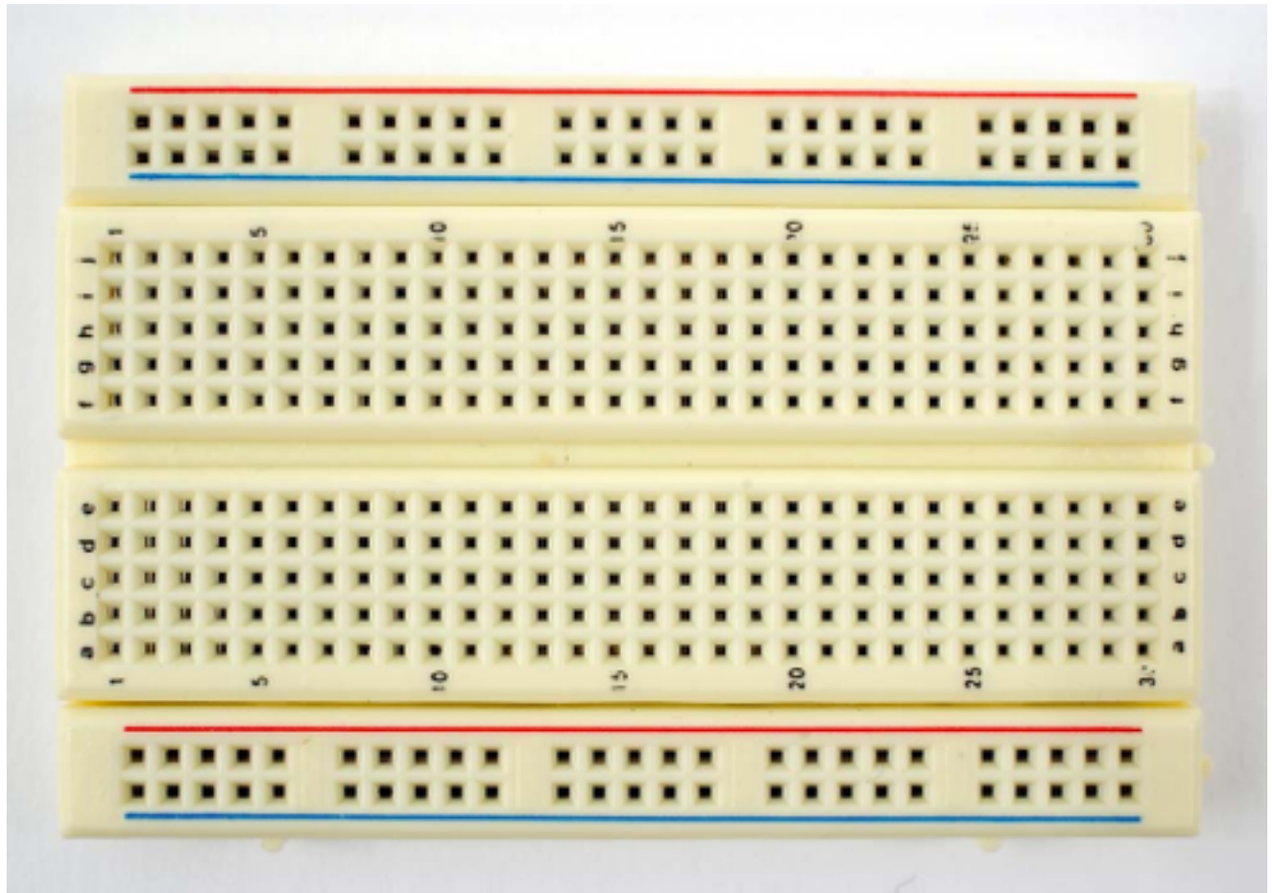


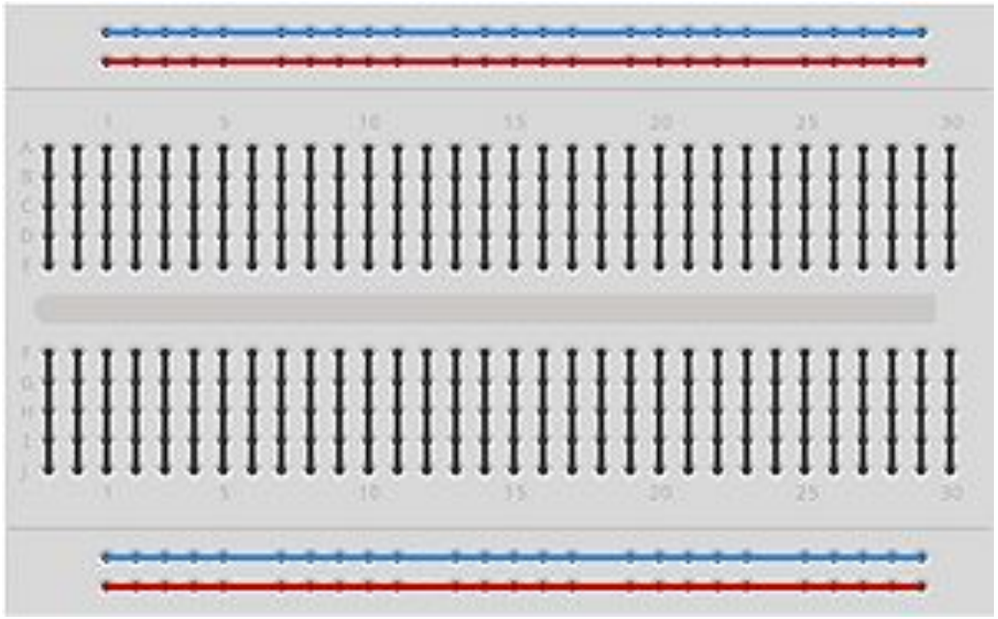
Install the software and open it

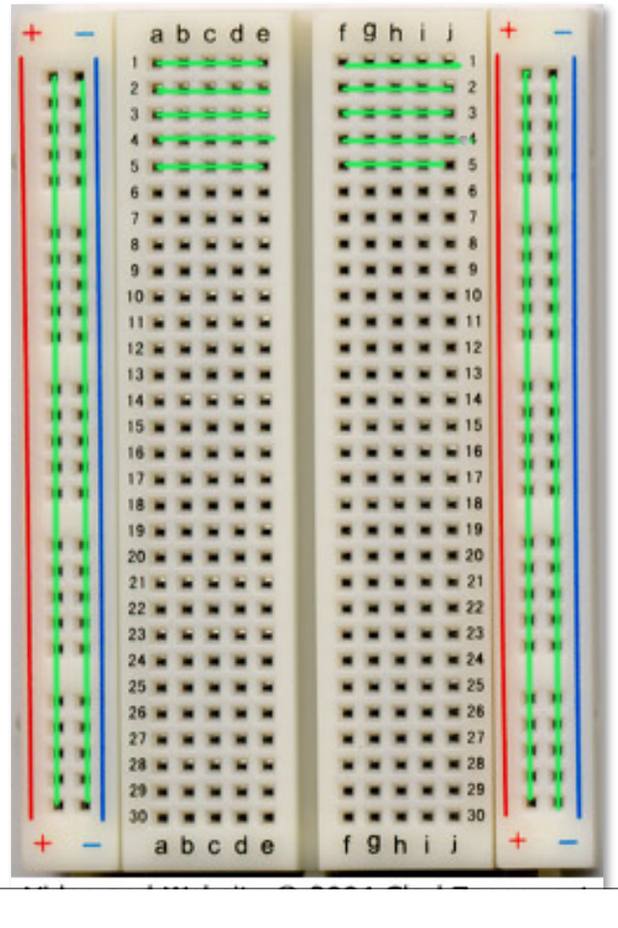
Download the p5 libraries

- https://github.com/vanevery/p5.serialport
- https://github.com/vanevery/p5.serialcontrol/releases

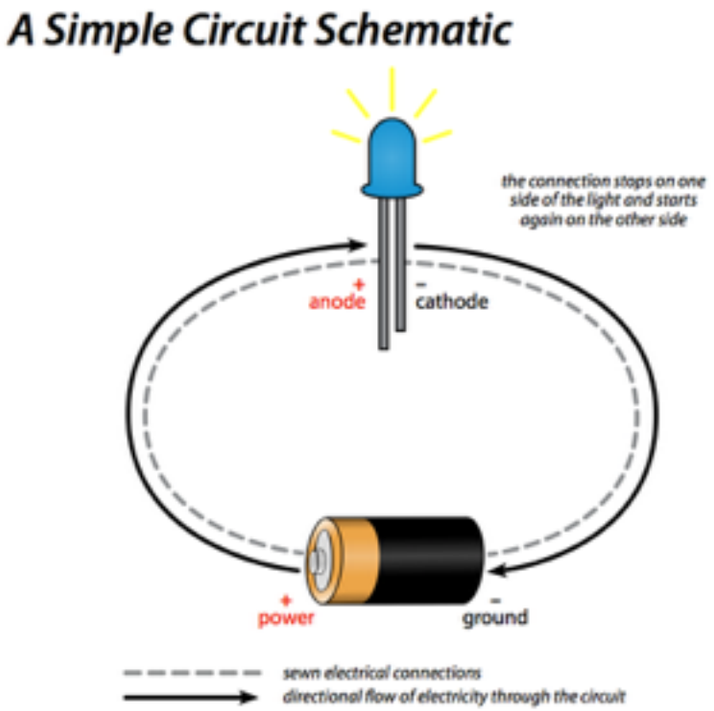
Need 2 things: the library in your p5 project folder + an internal server running (either with node.js installed + node startserver.js from the command line OR the p5.SerialControl (stand alone) application. We are going to use the p5.SerialControl app in conjunction with our p5 sketches

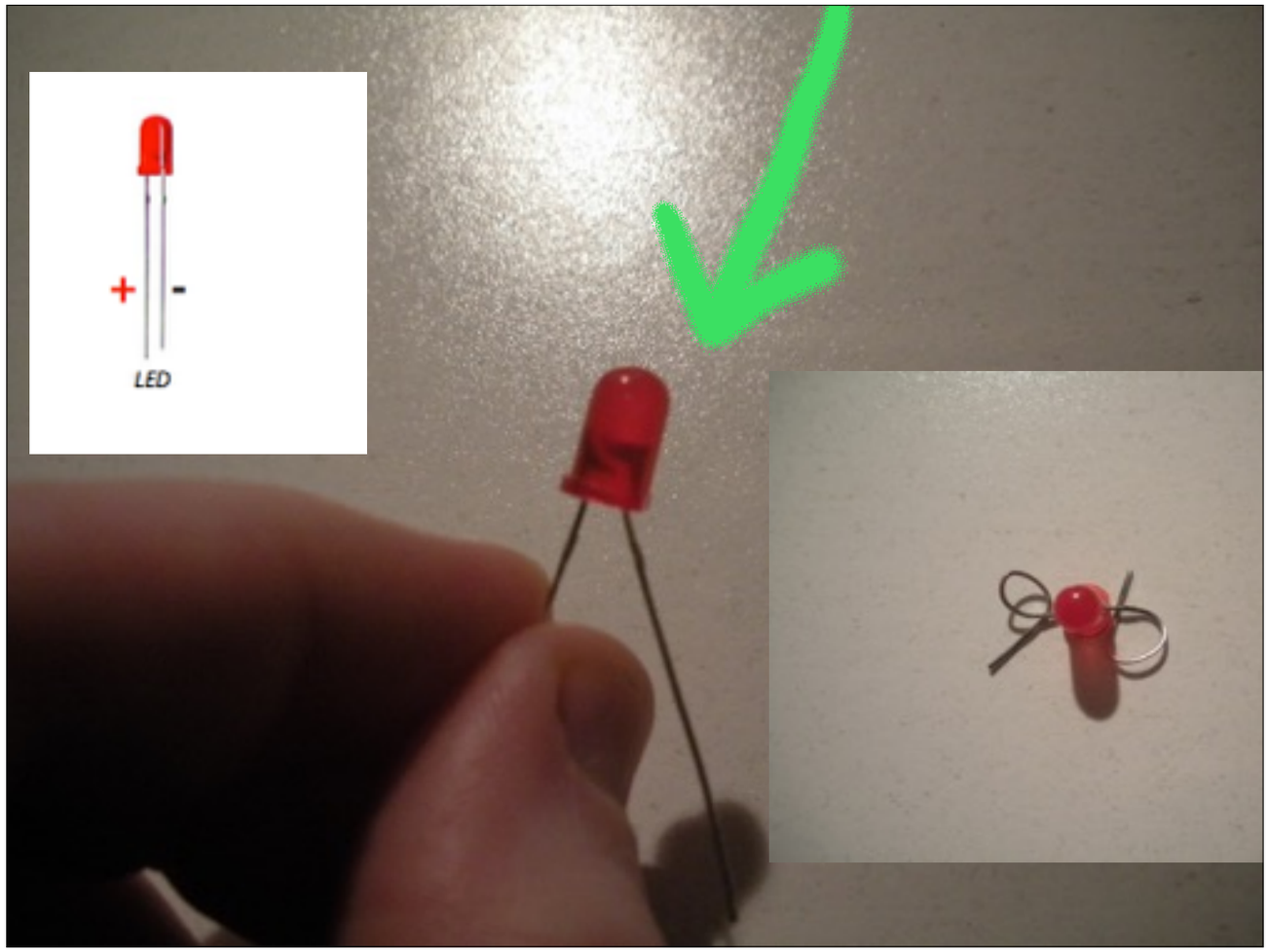

f g h i j
a b c d e
1
5
10
15
20
25




a b c d e
f g h i j
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

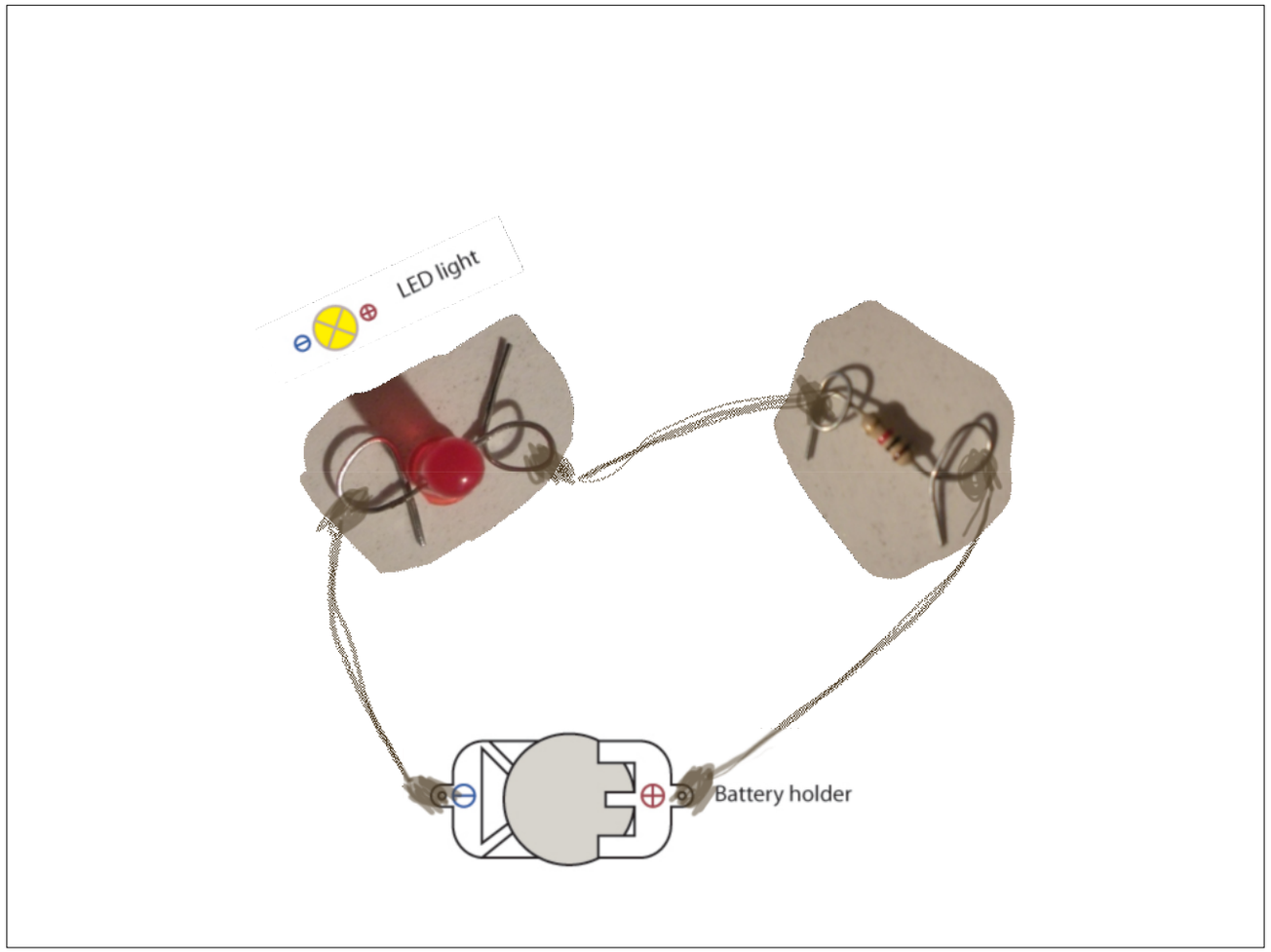
simple circuits



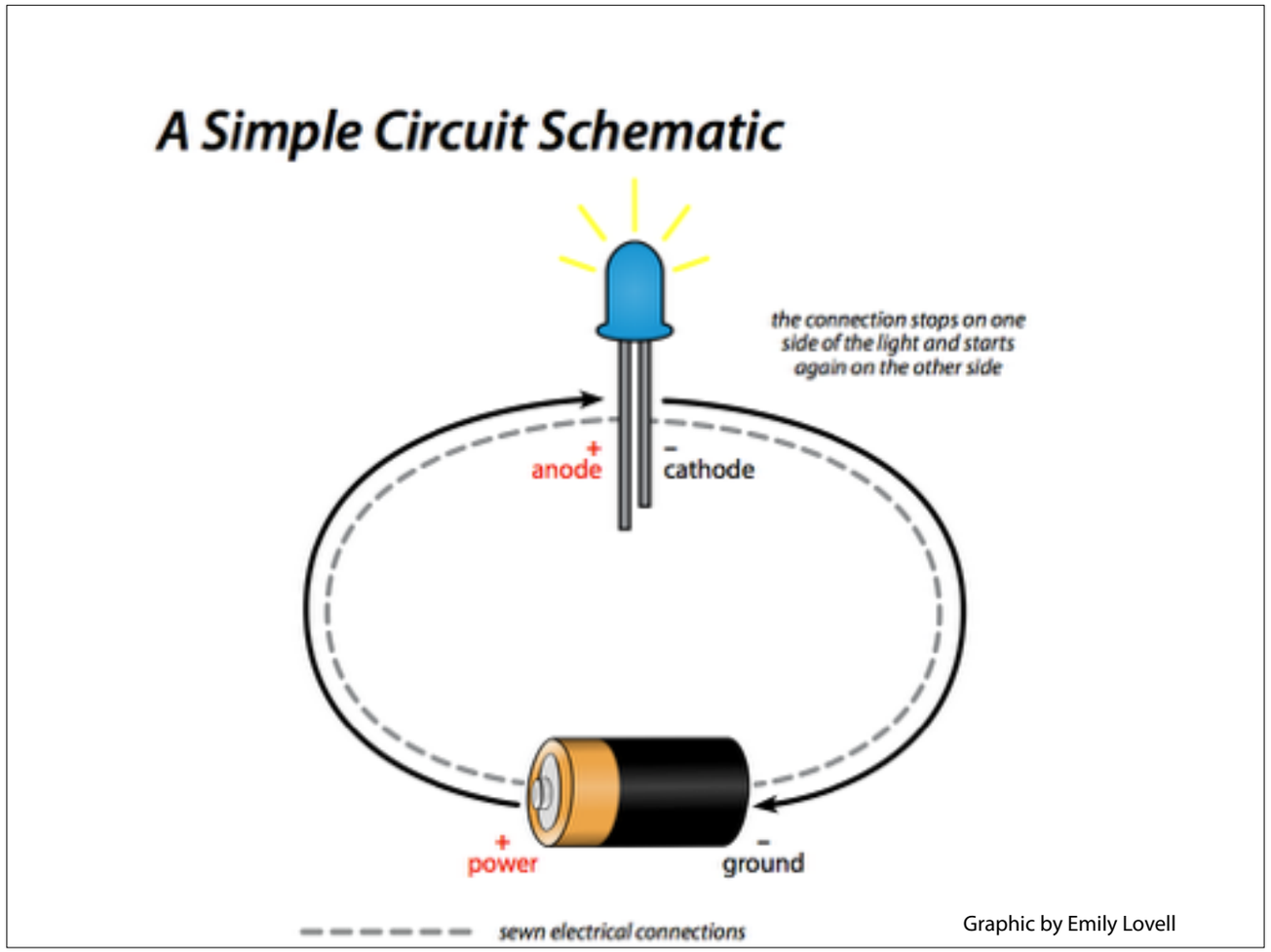


LED’s are polarized.

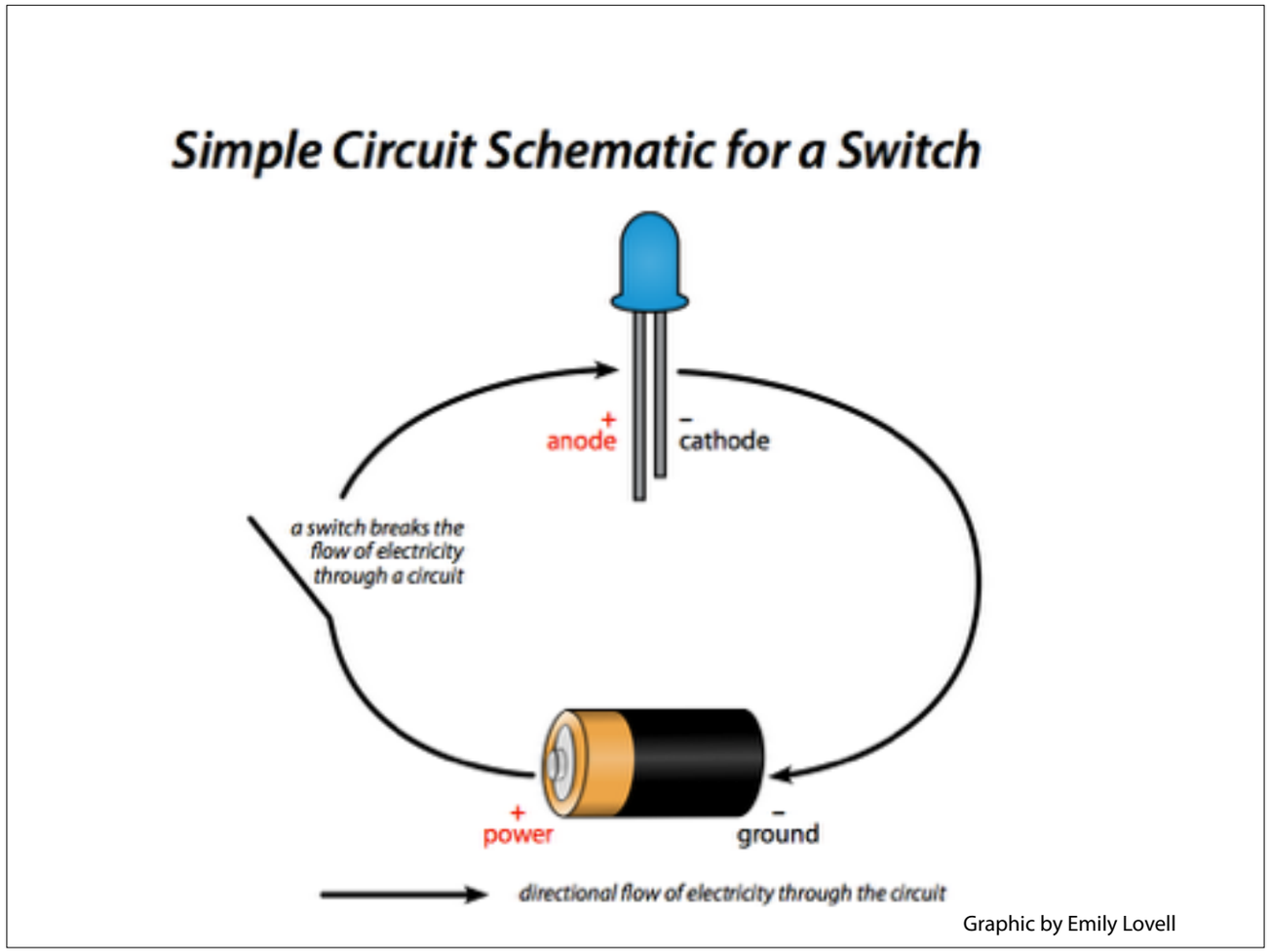
This means that current can only flow through in one direction. As a result, we need to pay attention to which end is which, when we are connecting them.



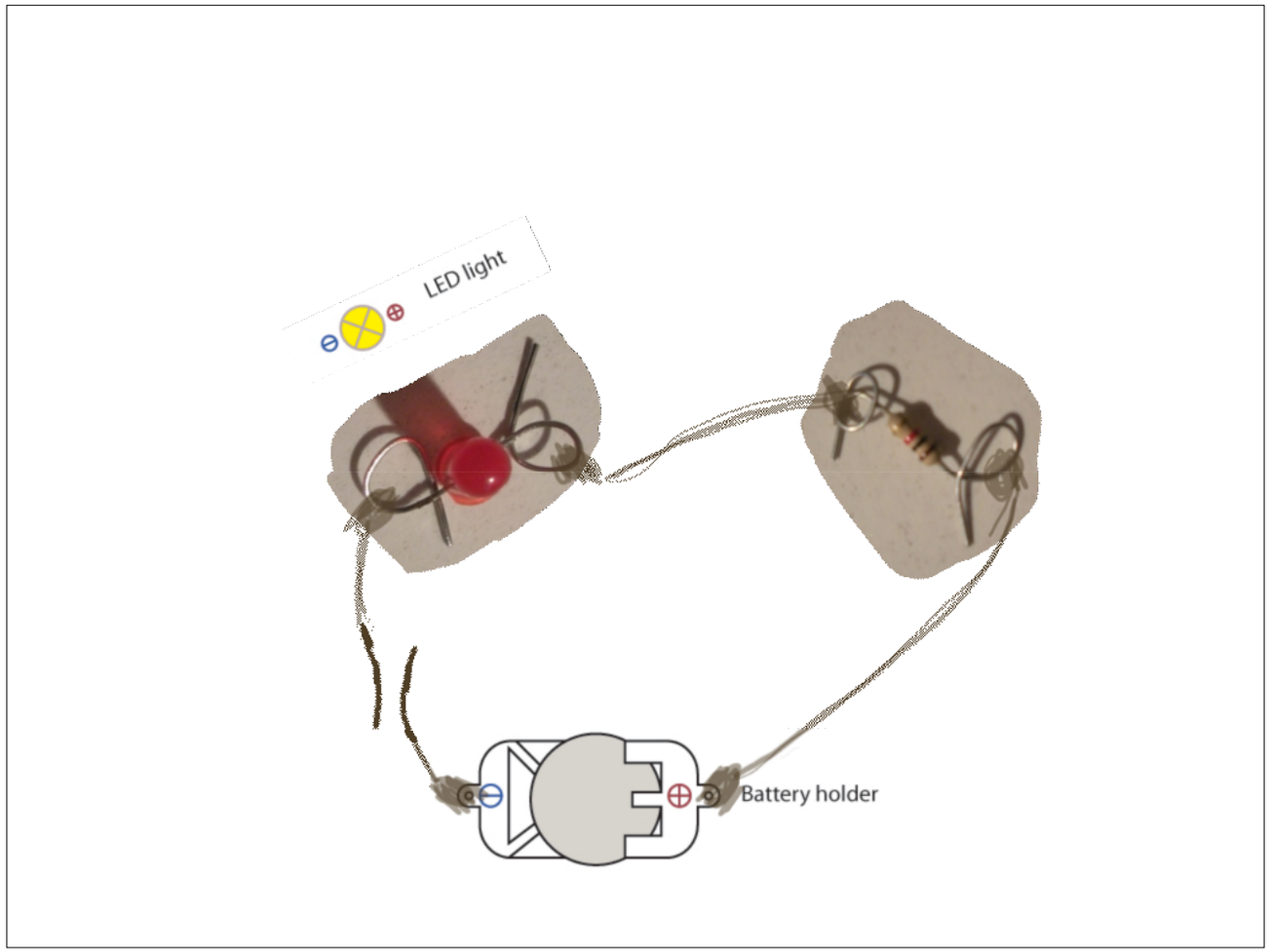
Notice that the positive side of the LED, is facing the direction of the positive side of the battery. The negative side of the LED is facing the direction of the negative side of the battery.

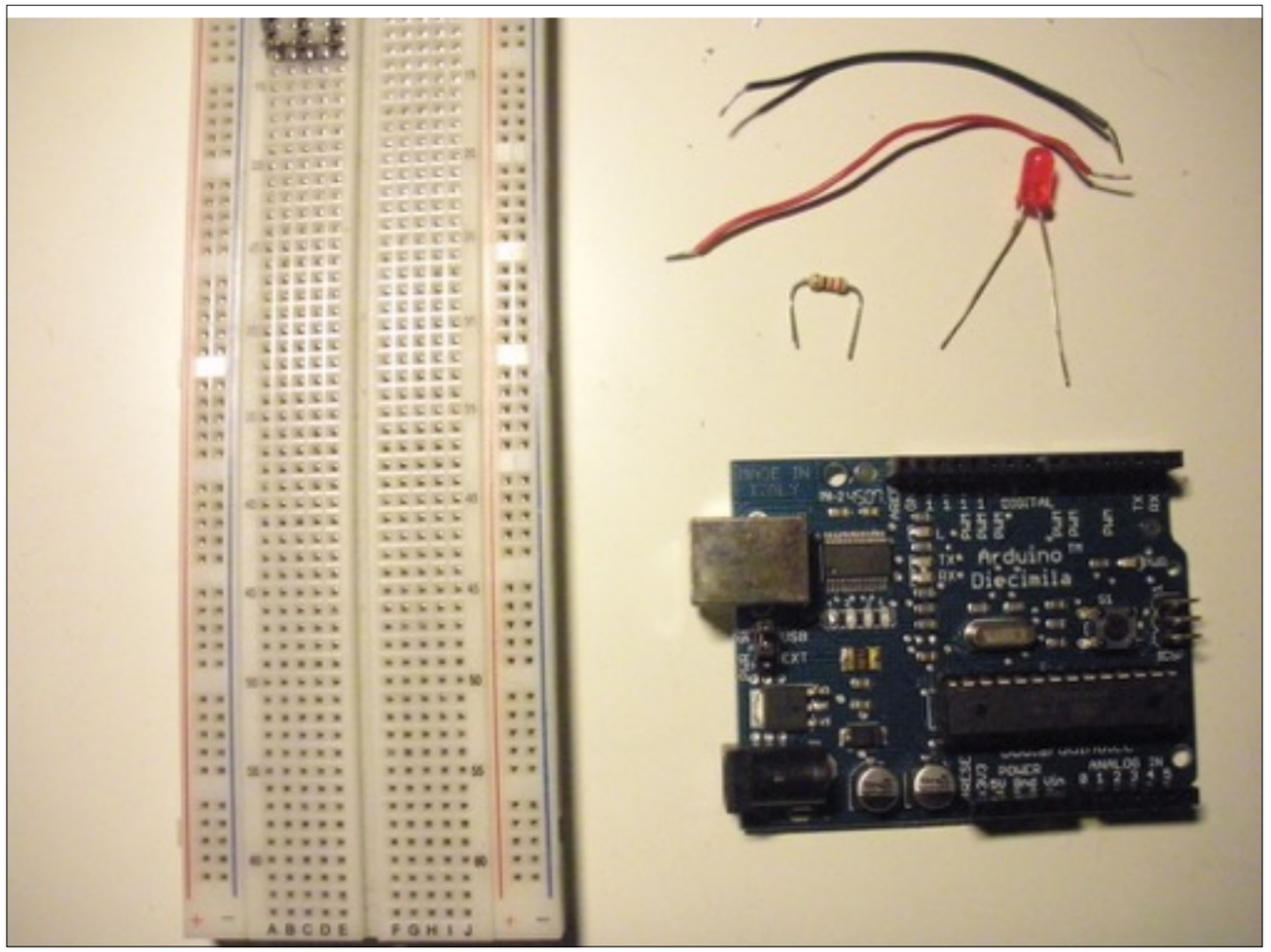


Graphic by Emily Lovell

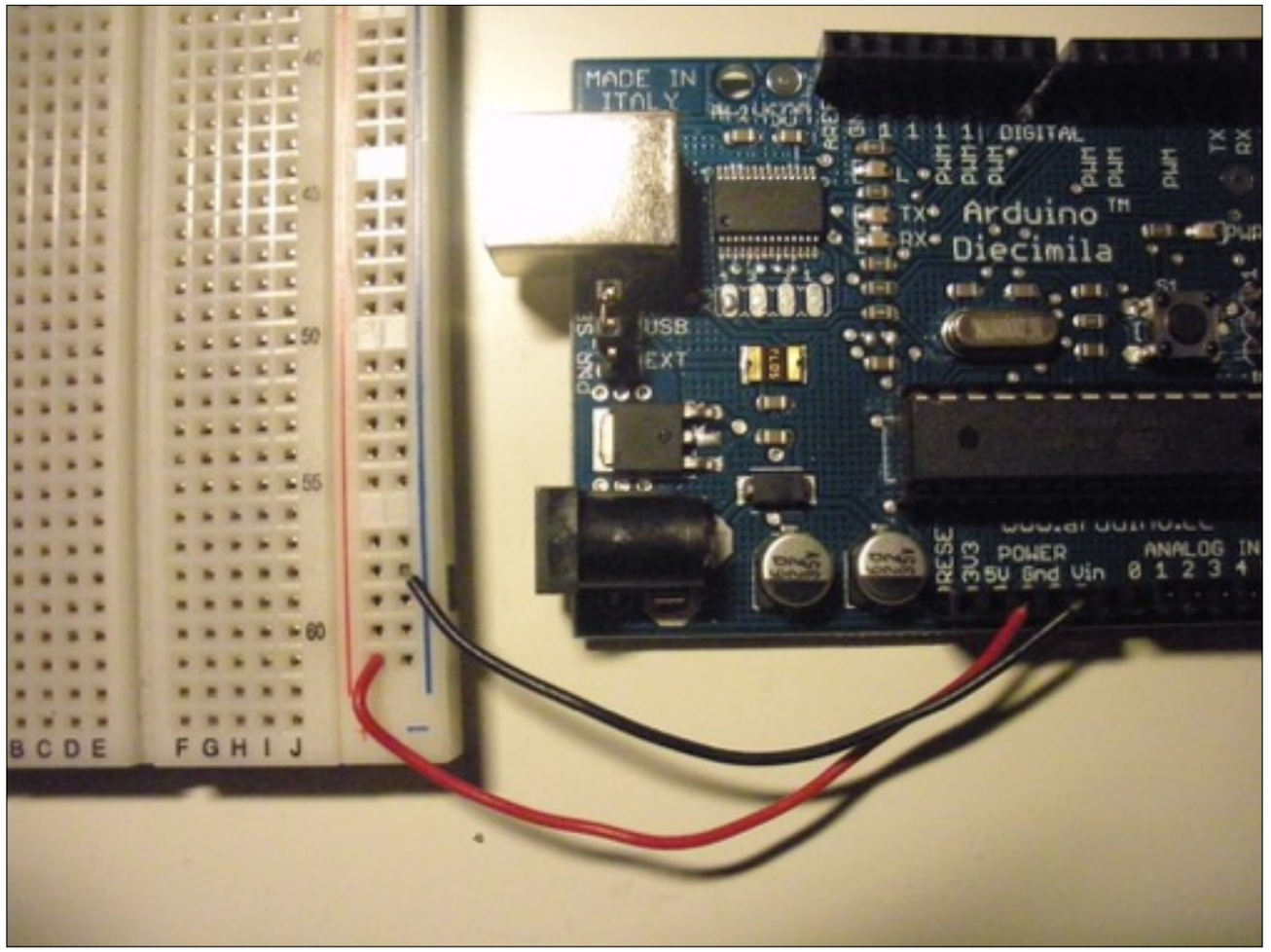


Graphic by Emily Lovell

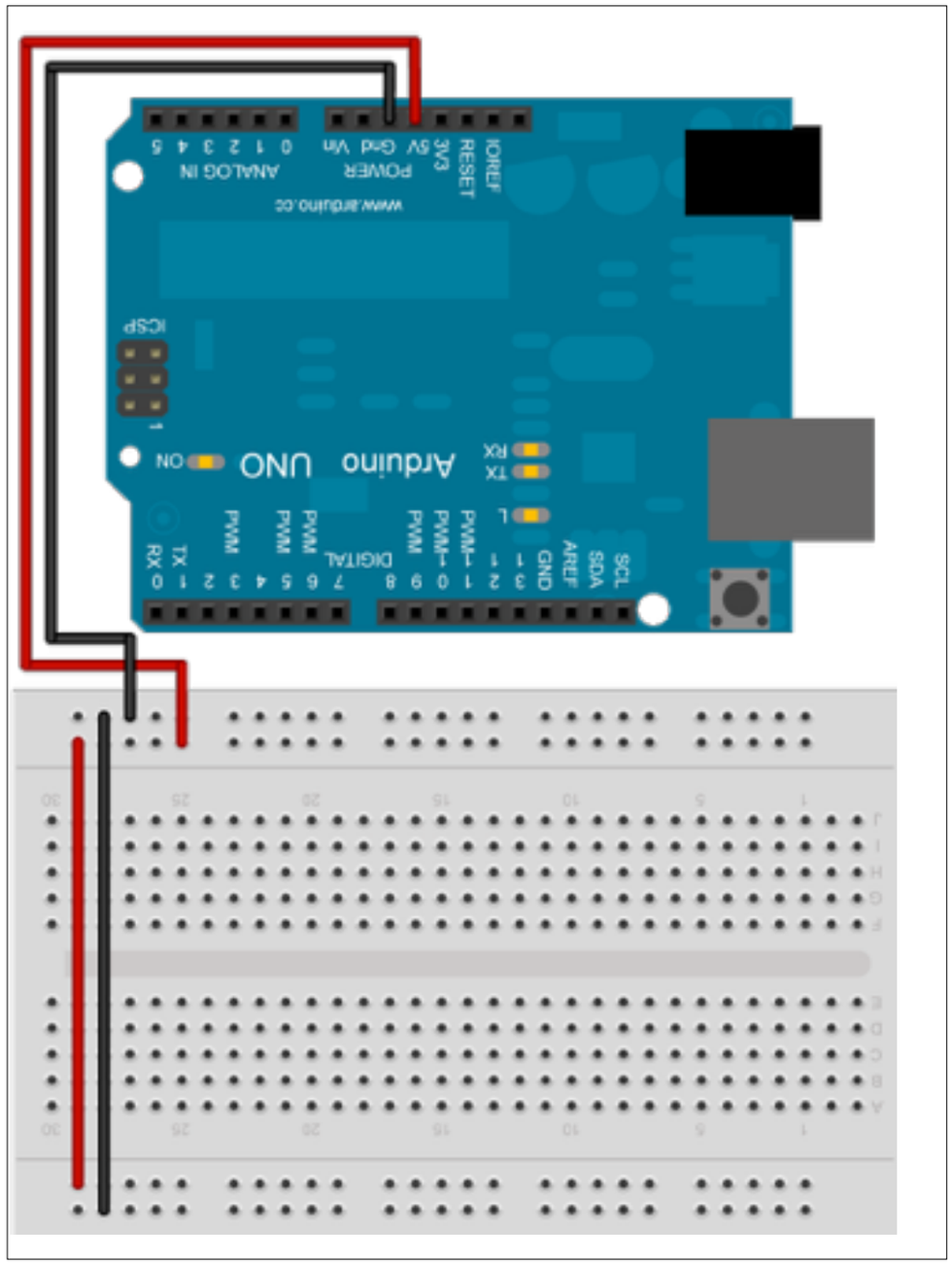

LED light
Battery holder

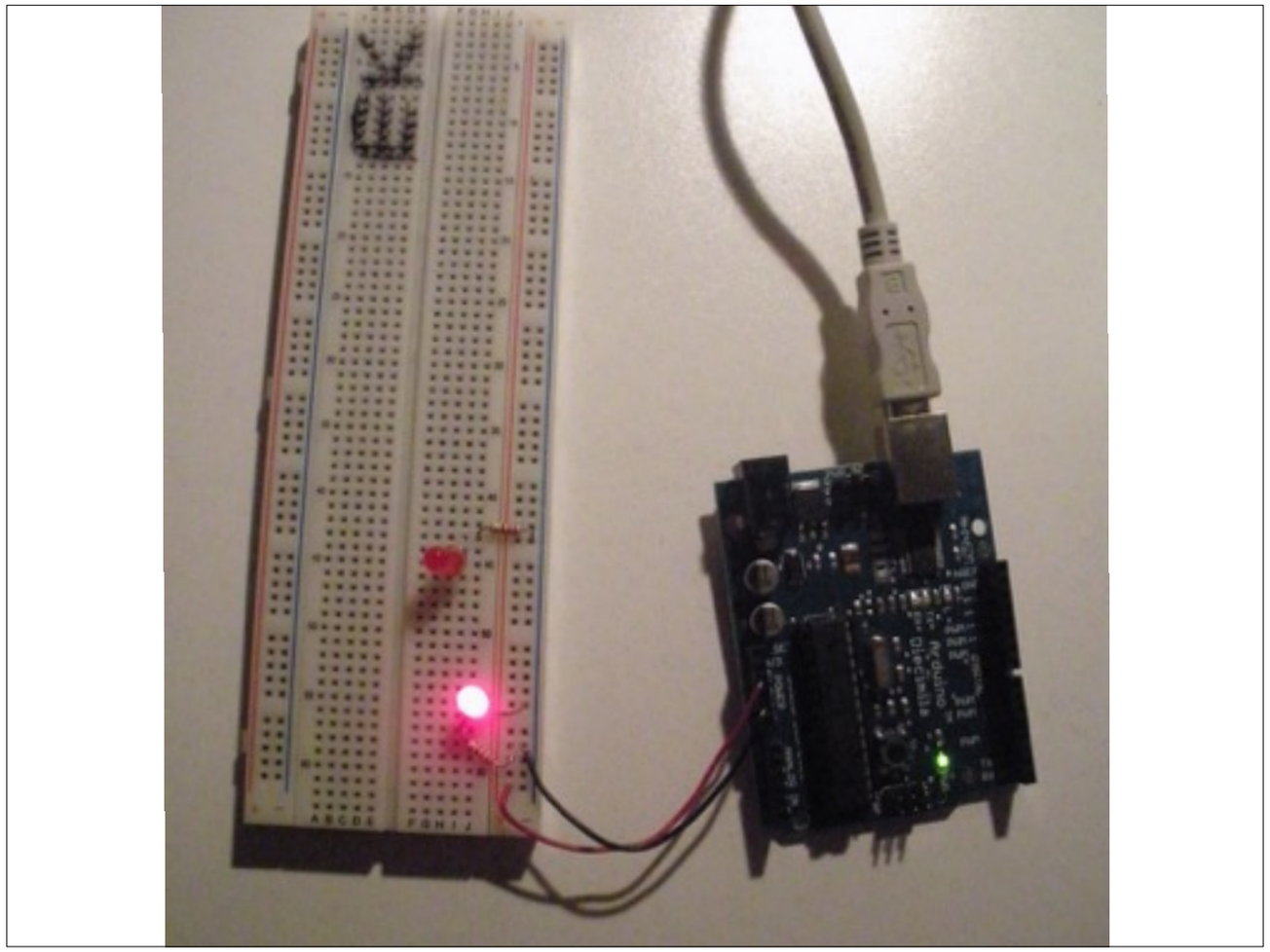


supplies

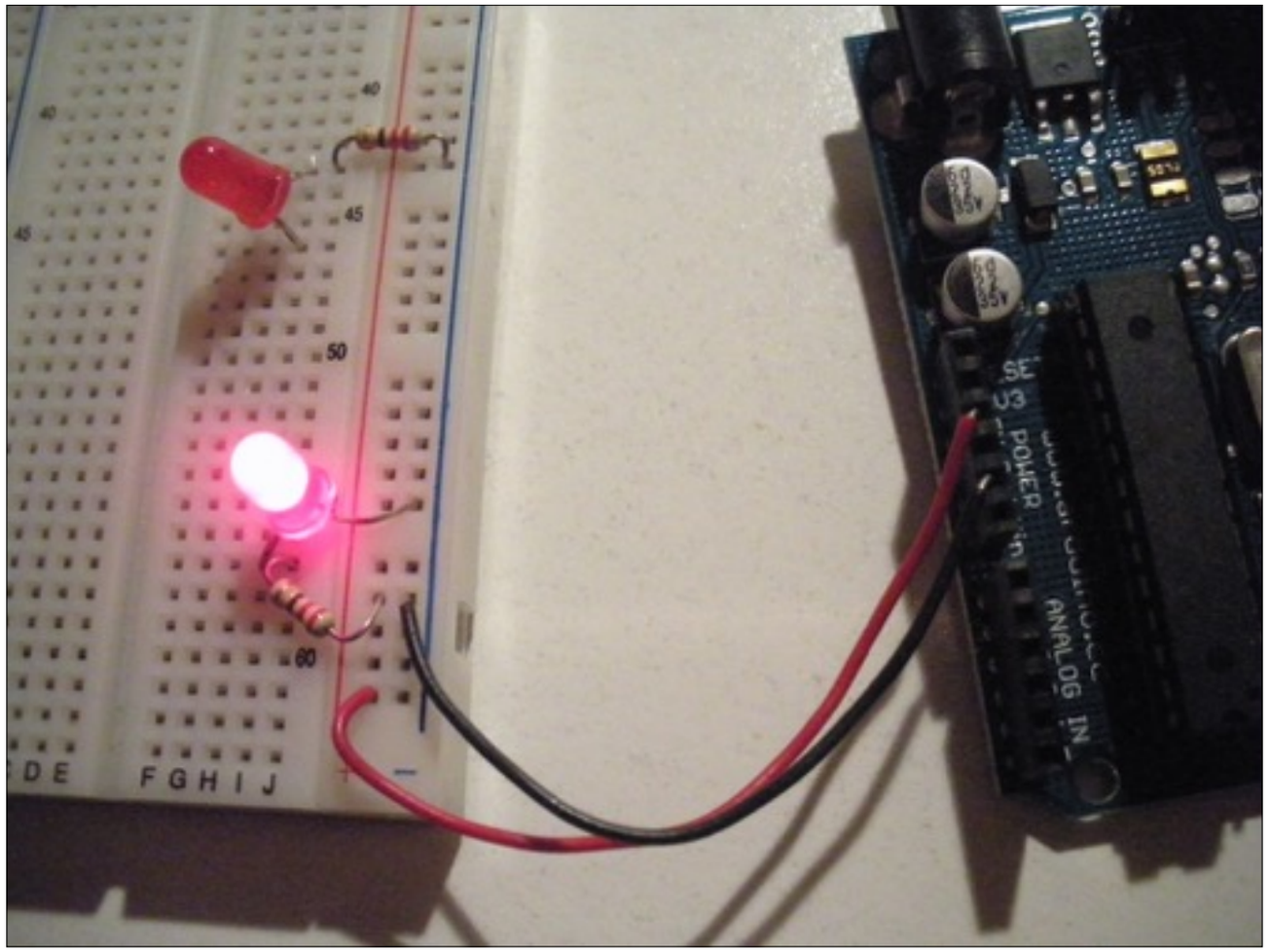


Jumper those lines over to your breadboard.
Place the black ground wire into the blue (-) side of your breadboard.
Place the red 5V wire into the red(+) side of your breadboard

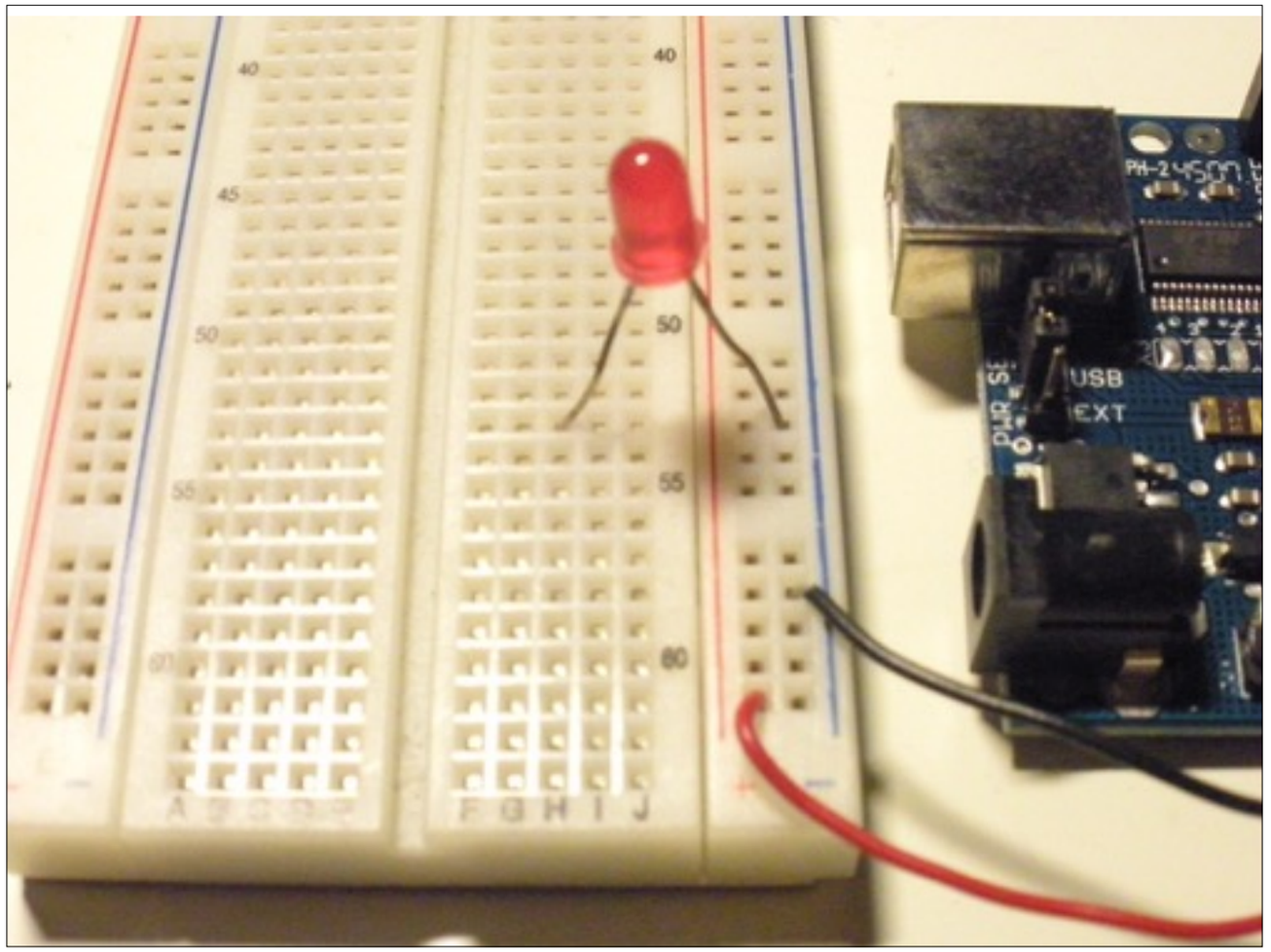

IOREF
RESET
3V3
5V
Gnd
Vin
POWER
0 1 2 3 4 5
ANALOG IN
www.arduino.cc
ICSP
ON
Arduino
UNO
TX
RX
L
DIGITAL
PWM
PWM
PWM
PWM
PWM
PWM
0 1 2 3 4 5 6 7 8 9 10 11 12 13
GND
AREF
SDA
SCL
RX
TX



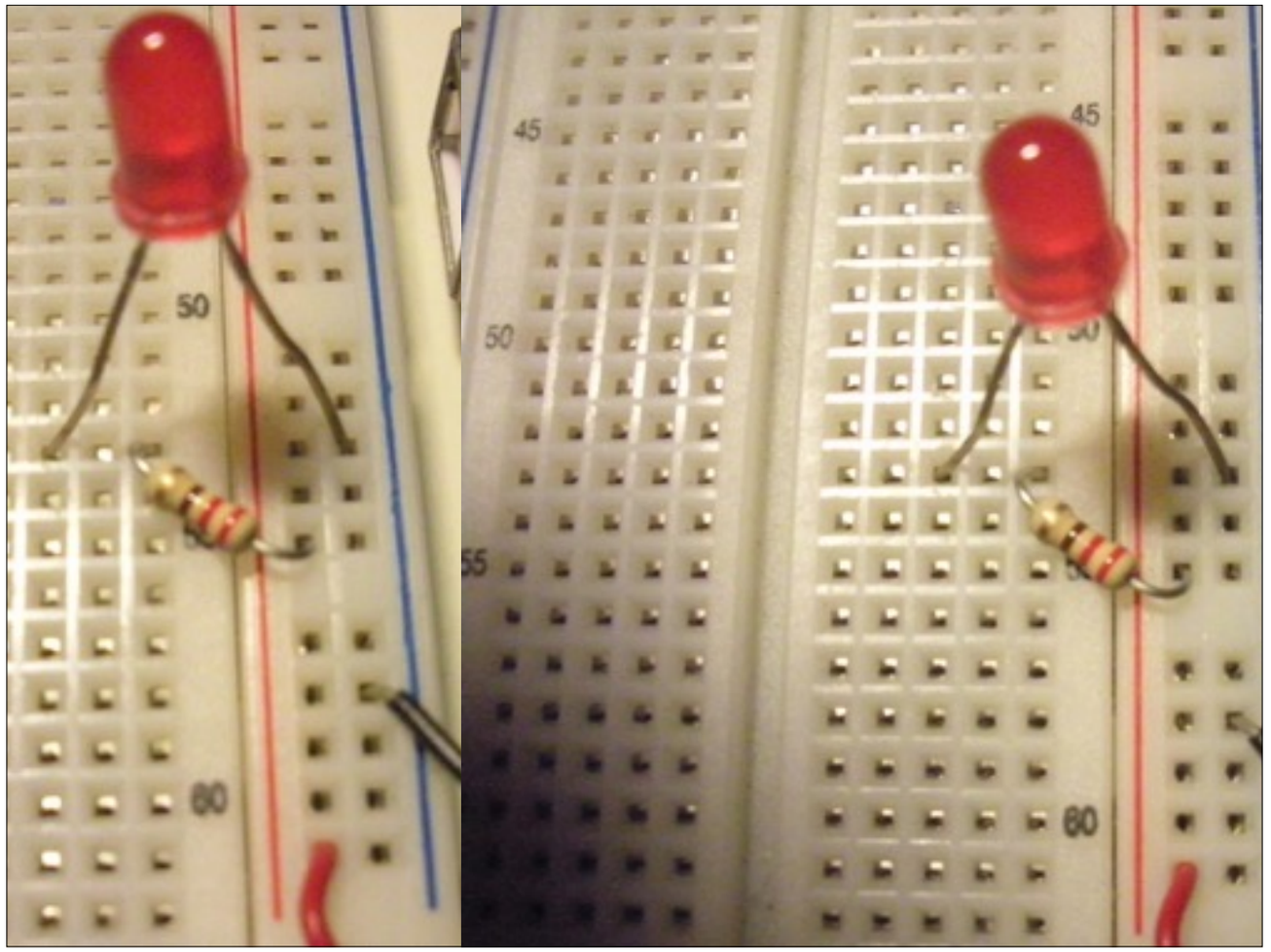
Plug the usb cable into the Arduino board and your laptop



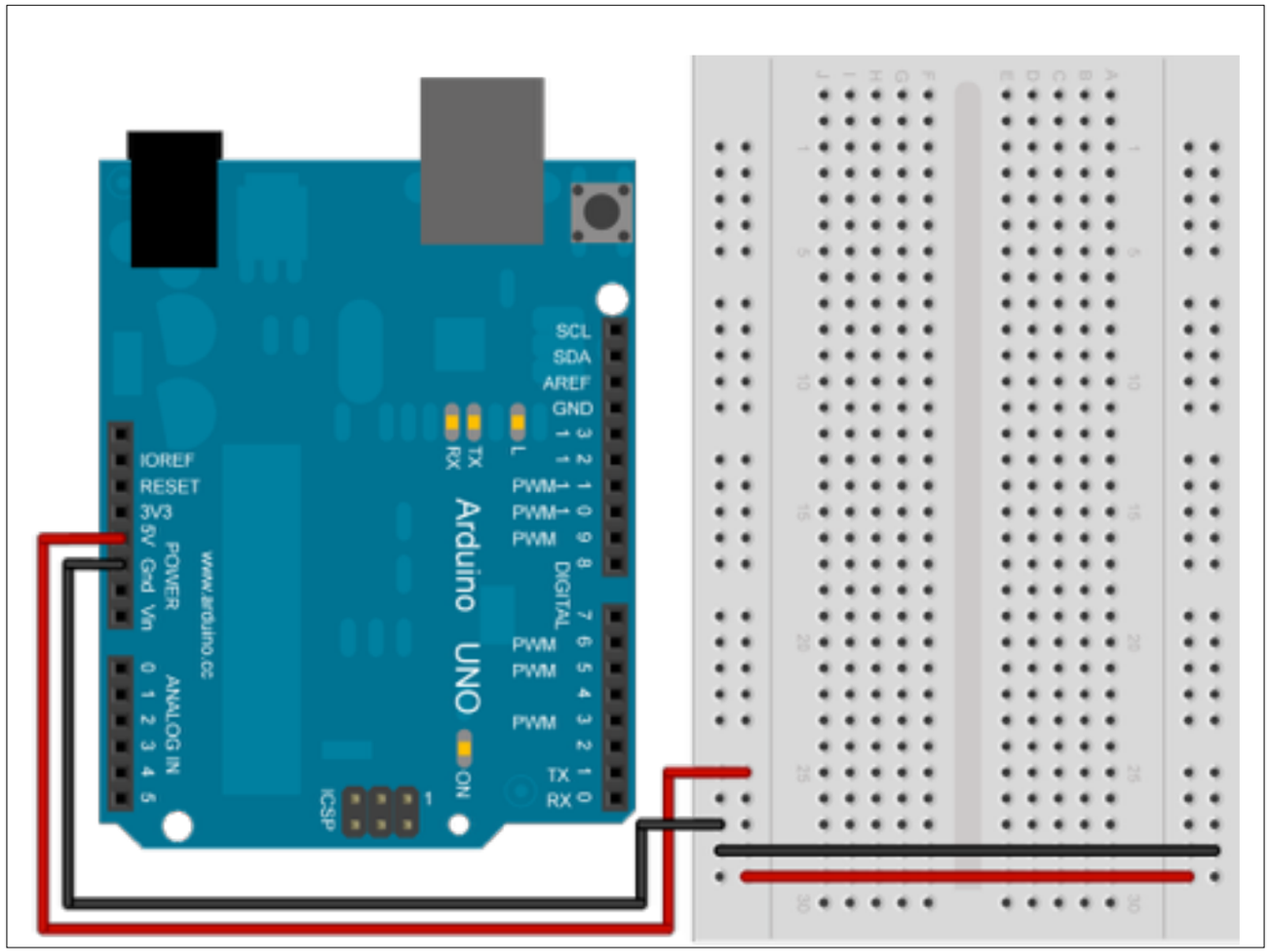
Status check



Place a test led on your breadboard. This will let us know that our breadboard is receiving power.
Make sure your led is in properly - ground must go to the blue (ground) side of your board.



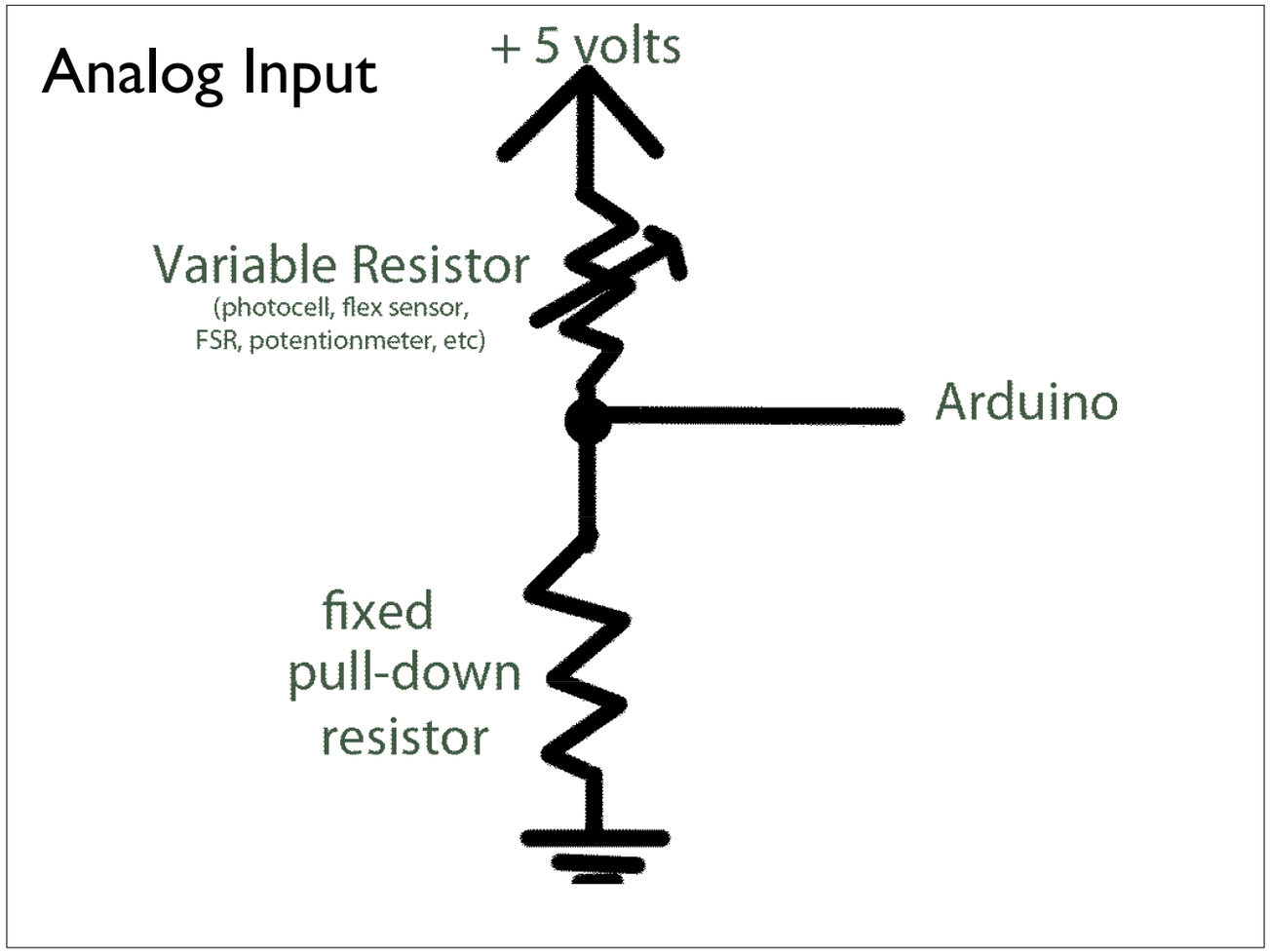
Insert 1 leg of your resistor into the same row of the led is in.
Plug the other leg of the resistor to the red (5V) side of your breadboard.



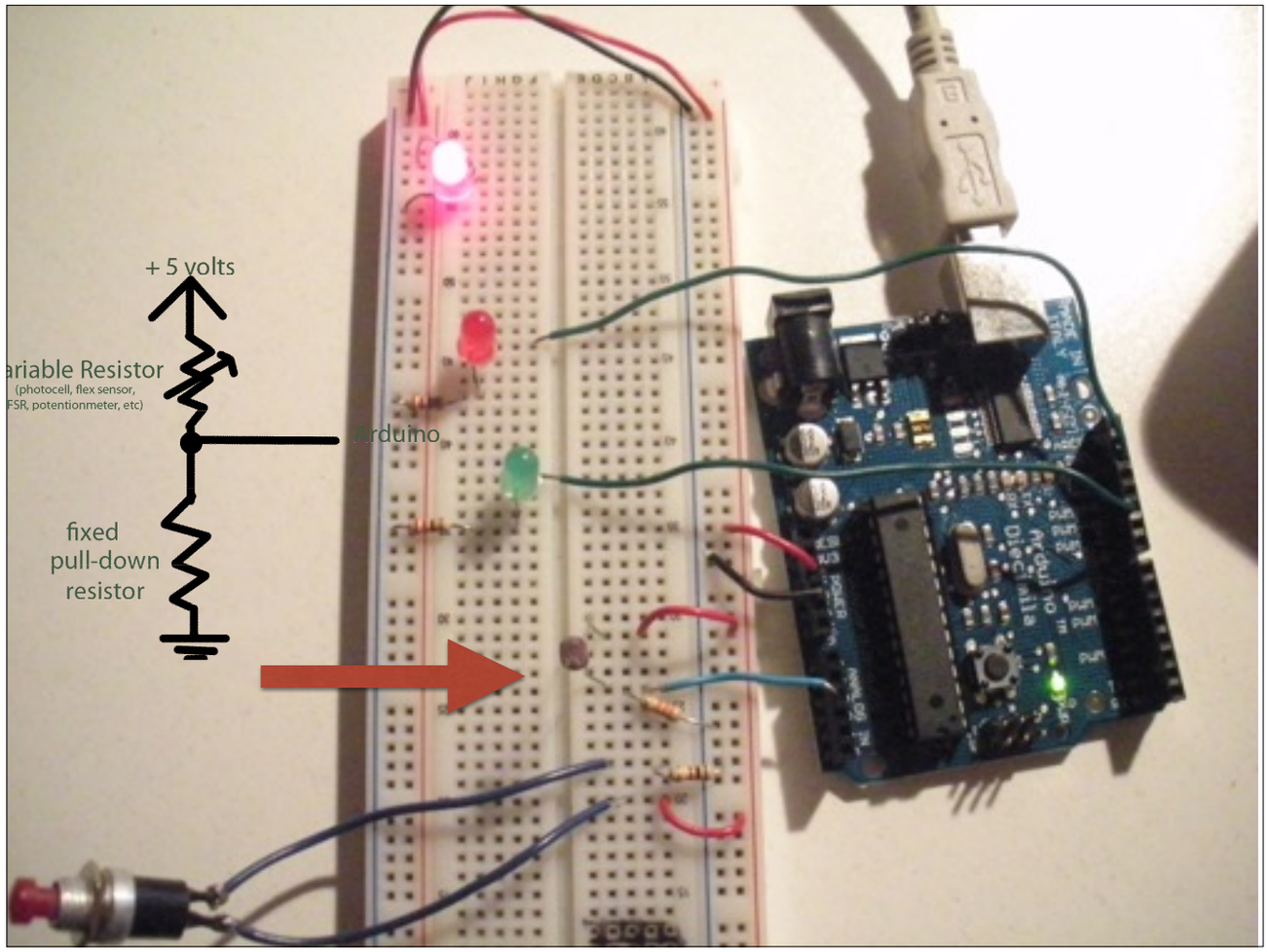
make sure your test LED is on your board as well

Input

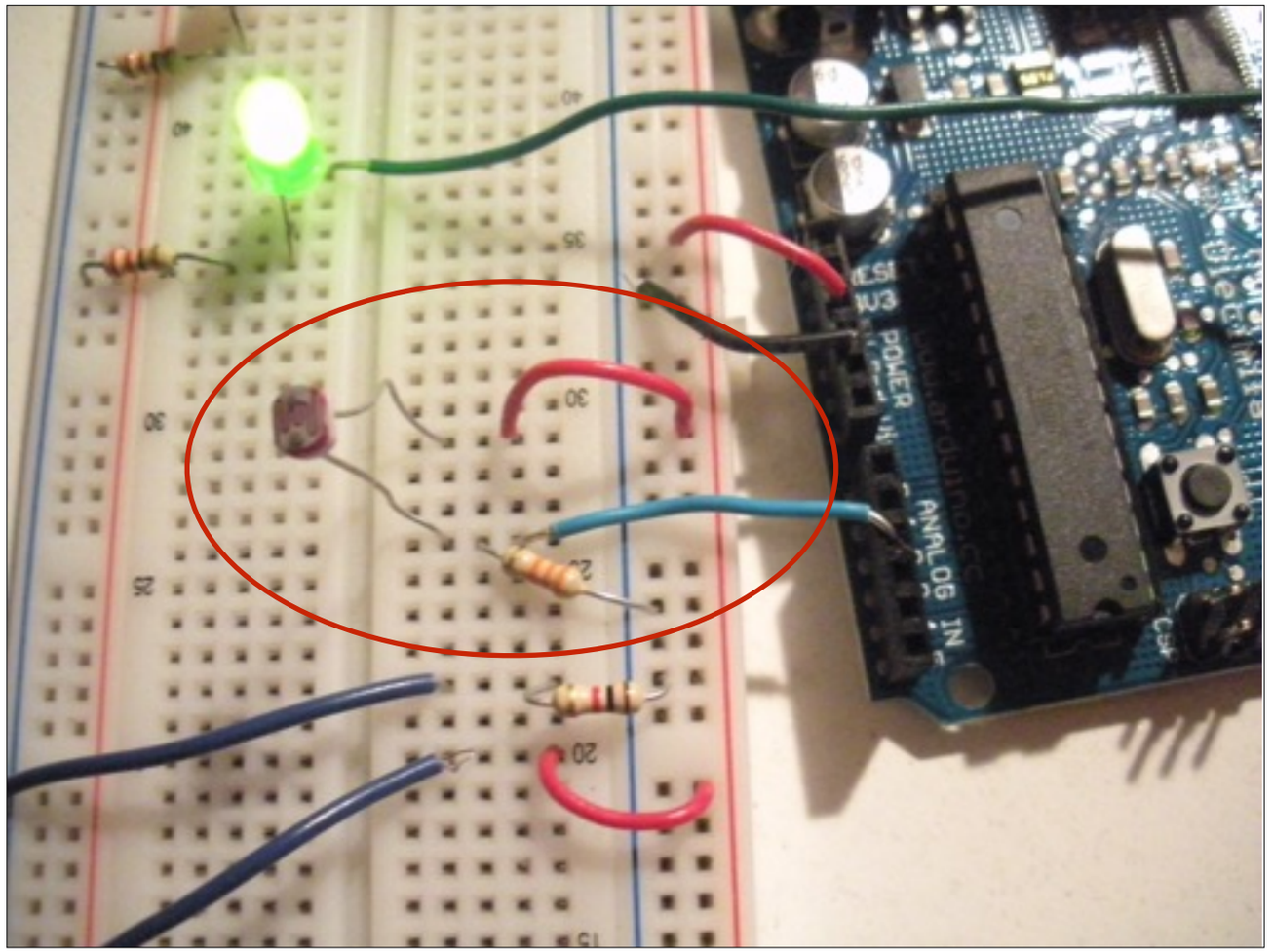
- Analog Input



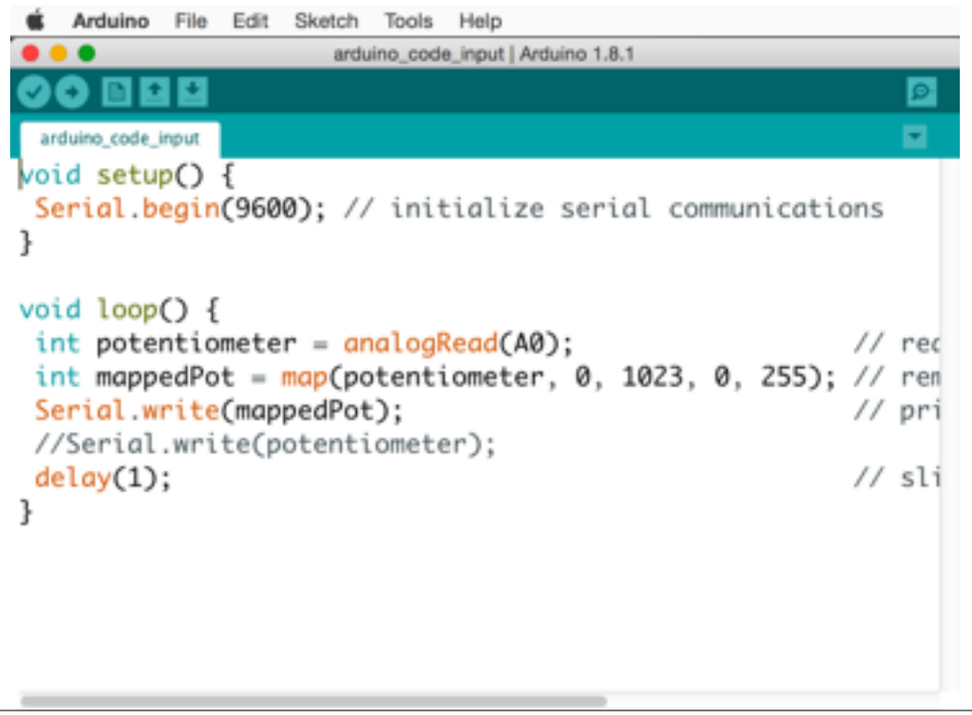
Voltage Divider


+ 5 volts
ariable Resistor
(photocell, flex sensor,
FSR, potentionmeter, etc)
Arduino
fixed
pull-down
resistor

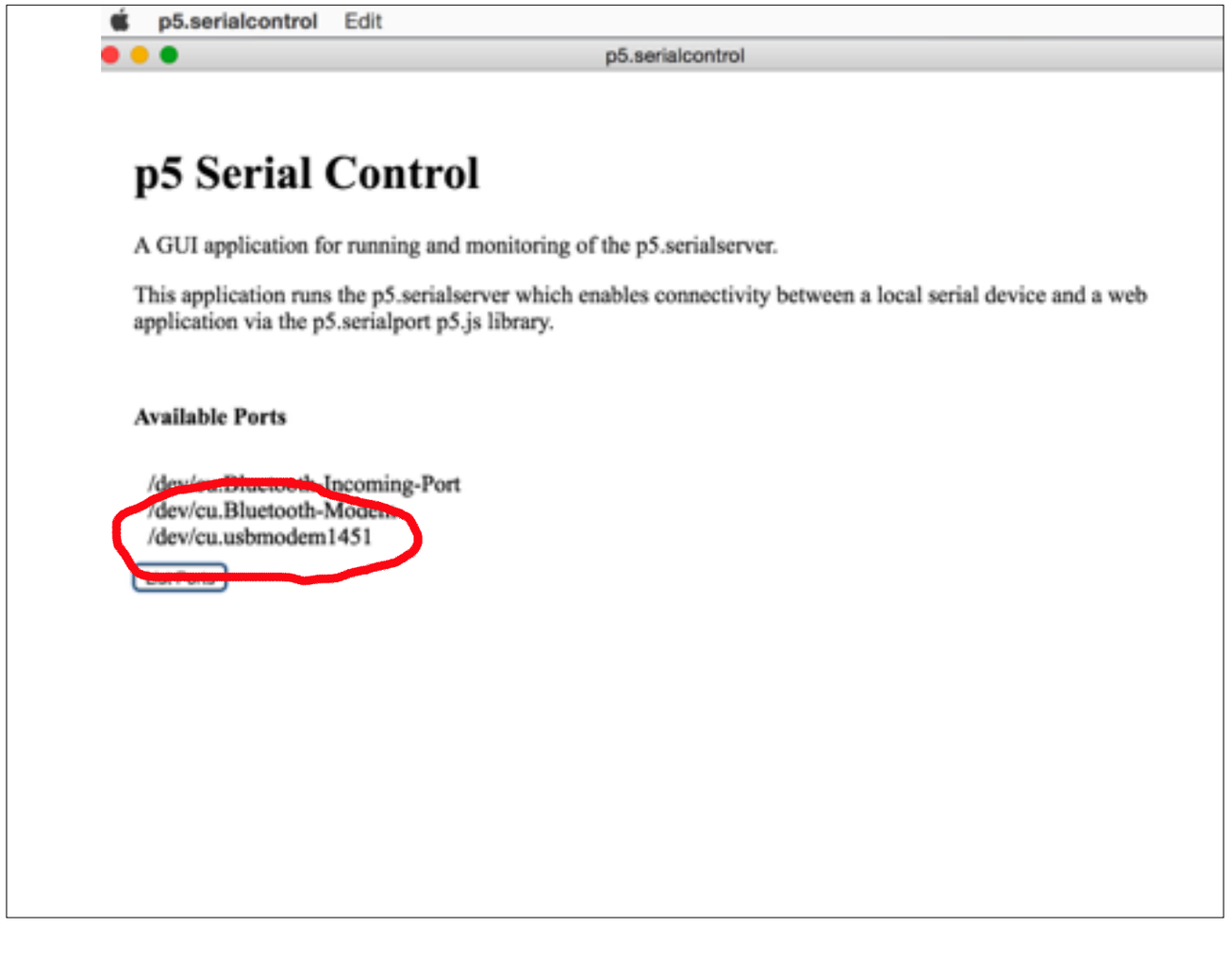
Analog In

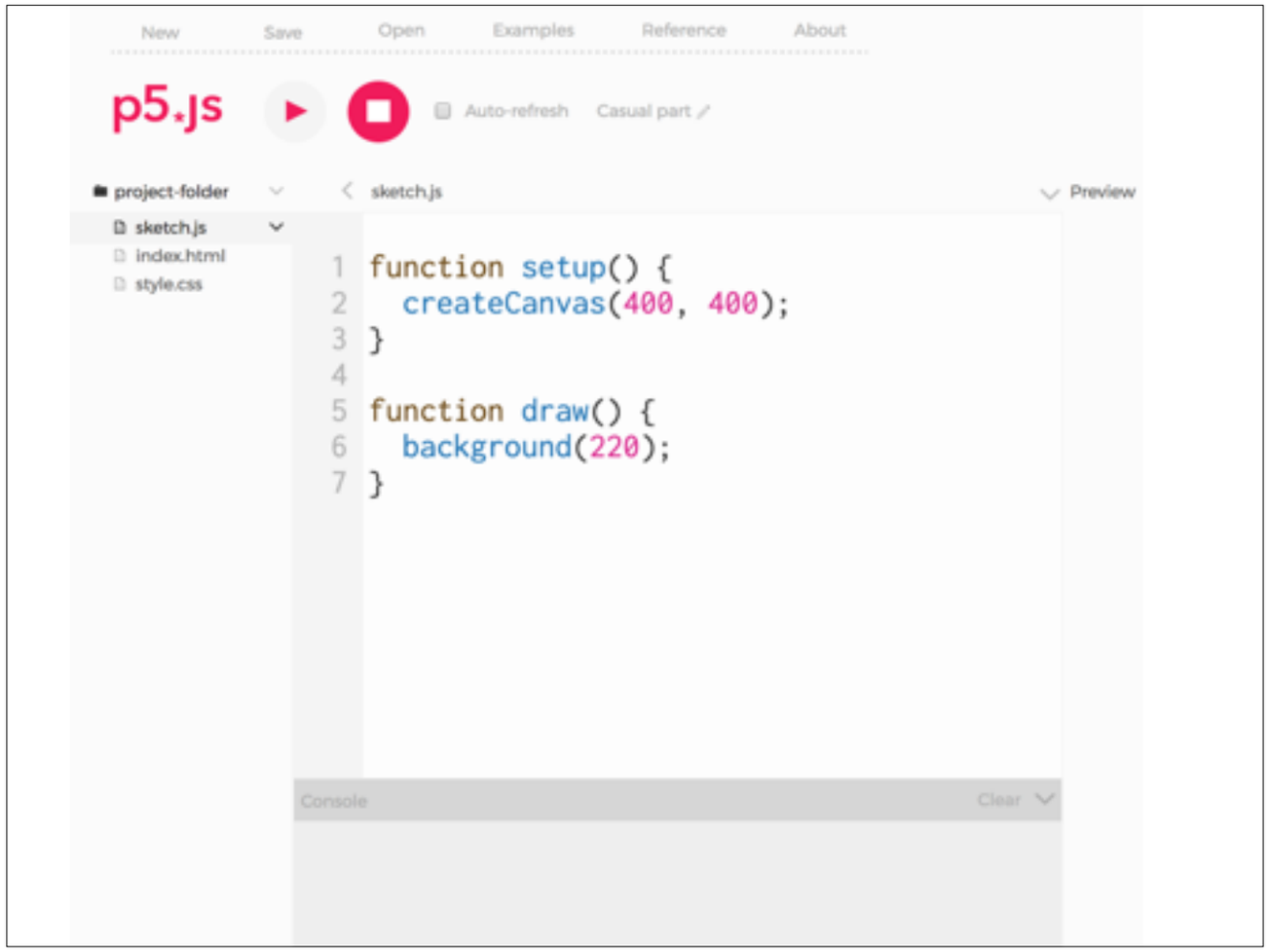

POWER
ANALOG IN

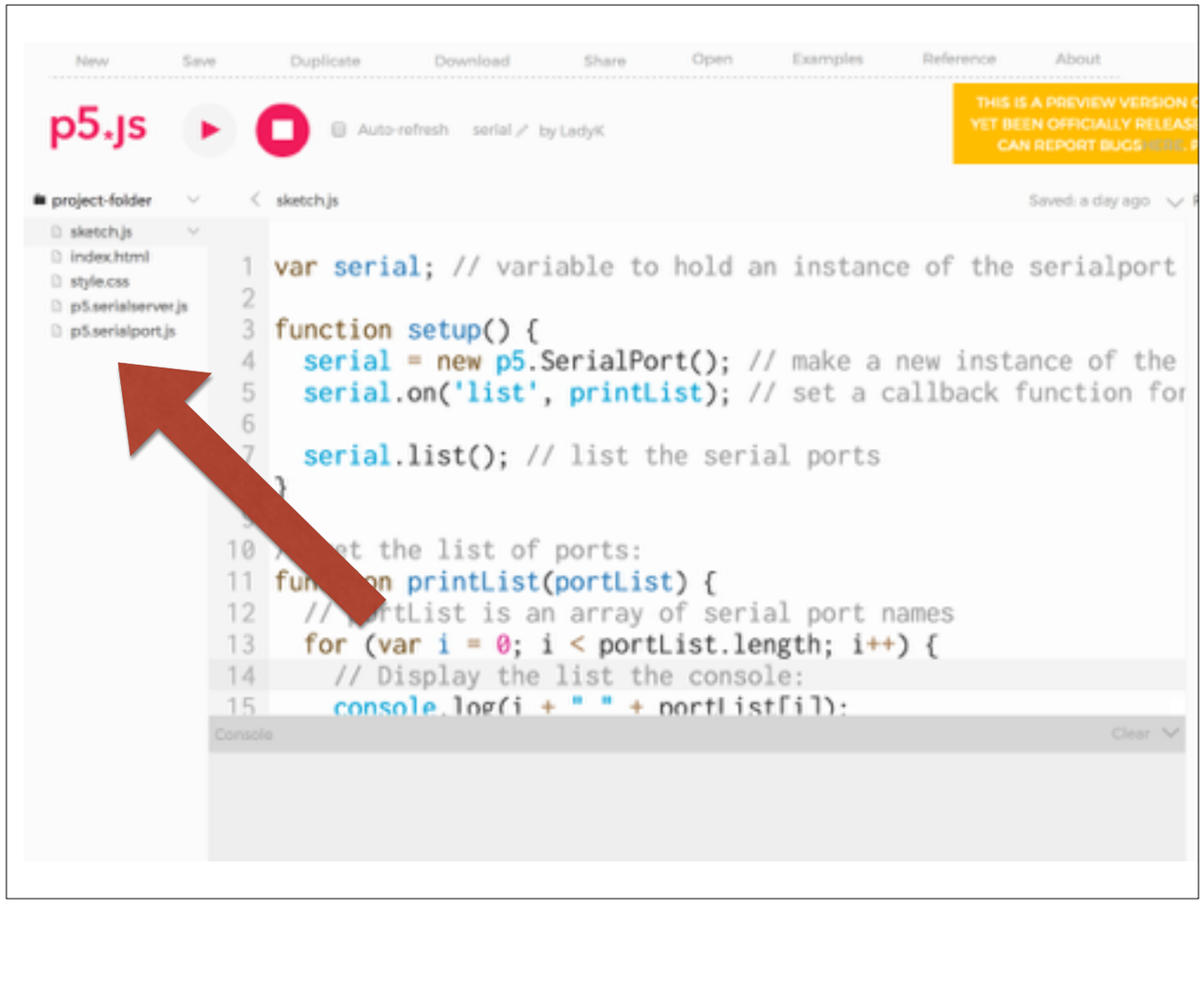
Burn the code onto the arduino



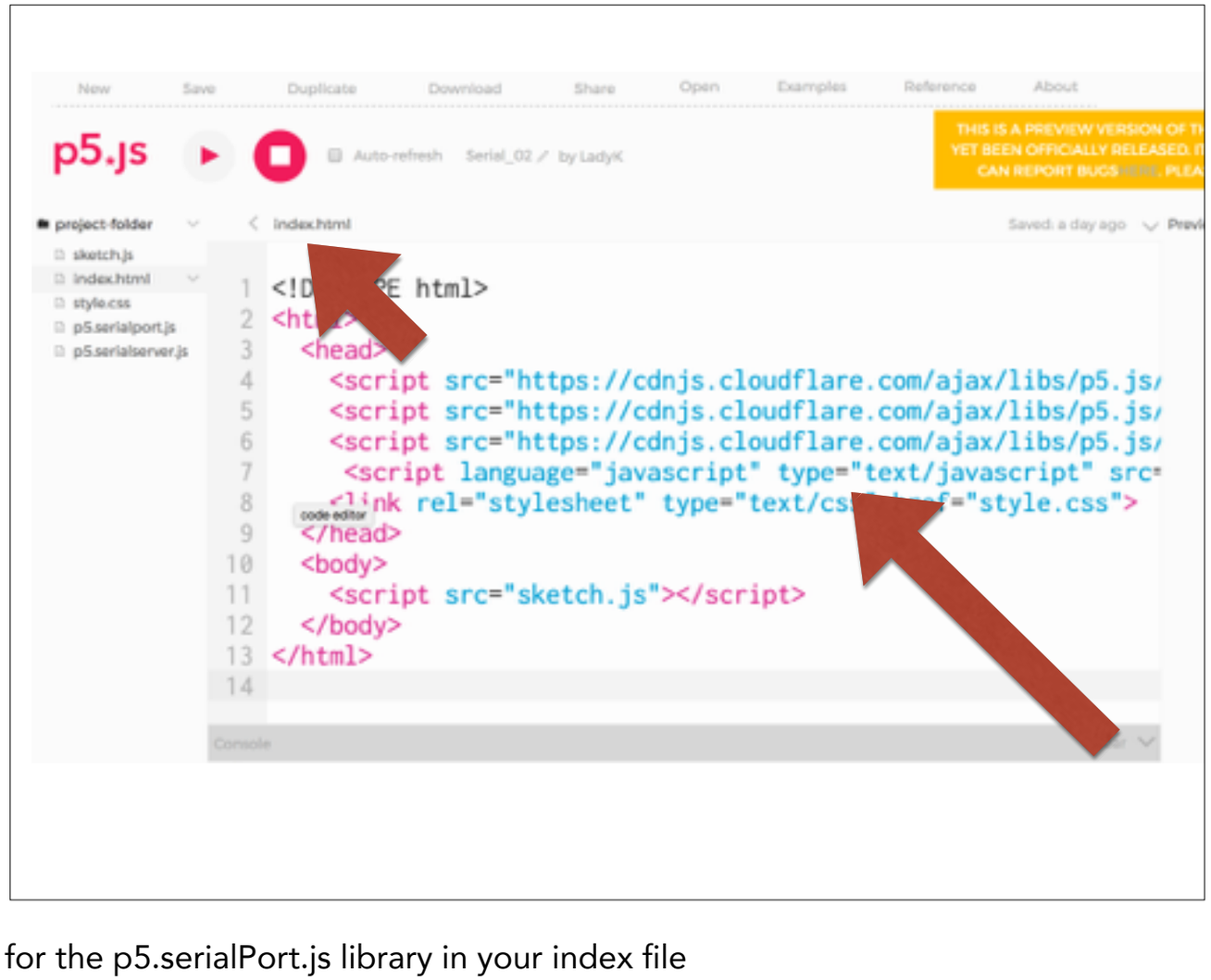
Start the p5 Serial Control App


p5.serialcontrol Edit
p5.serialcontrol
p5 Serial Control
A GUI application for running and monitoring of the p5.serialserver.
This application runs the p5.serialserver which enables connectivity between a local serial device and a web application via the p5.serialport p5.js library.
Available Ports
/dev/cu.Bluetooth-Incoming-Port
/dev/cu.Bluetooth-Modem
/dev/cu.usbmodem1451

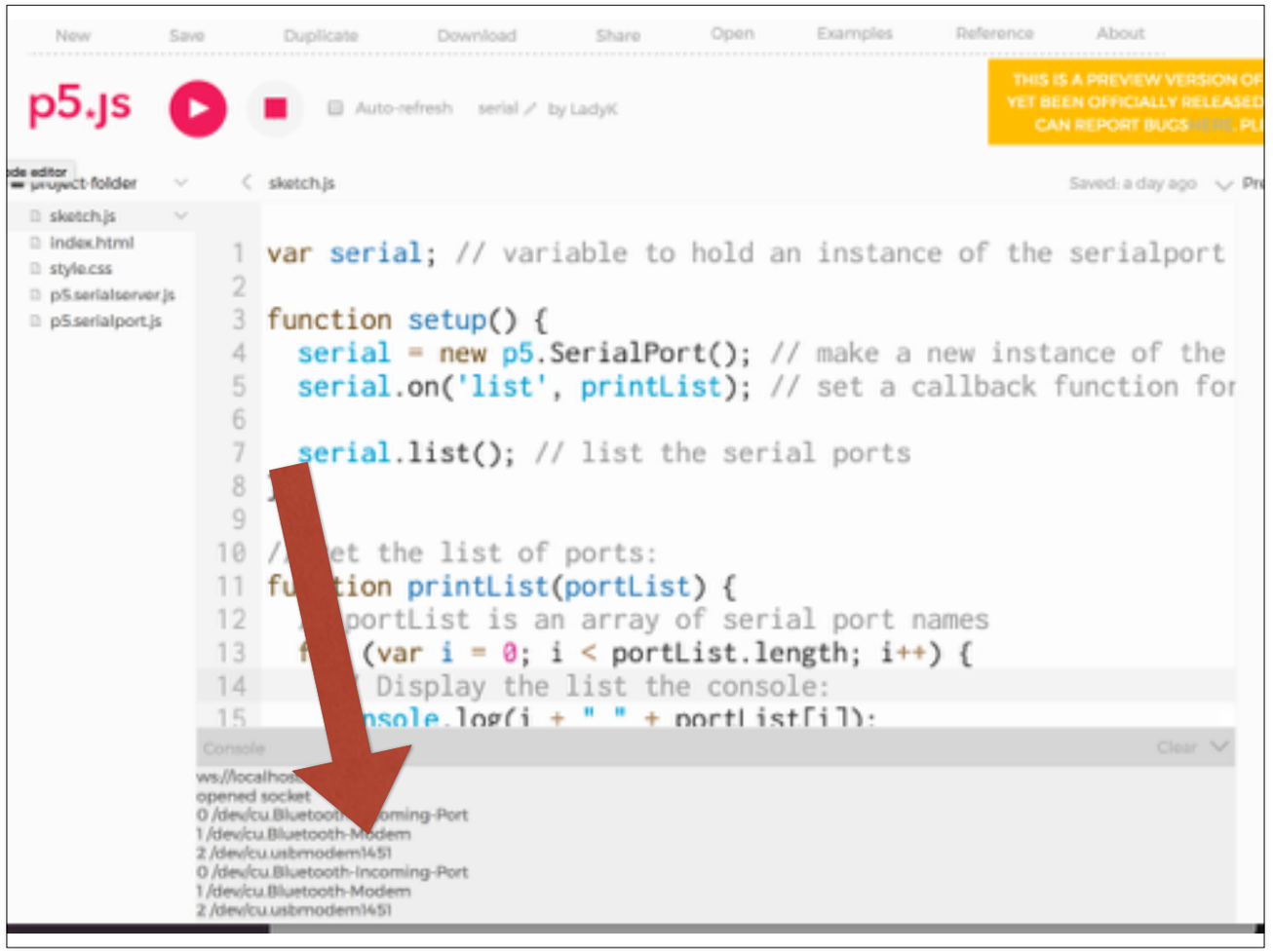

New
Save
Open
Examples
Reference
About
p5.js
Auto-refresh
Casual part
project-folder
sketch.js
index.html
style.css
sketch.js
Preview
1 function setup() {
2 createCanvas(400, 400);
3 }
4
5 function draw() {
6 background(220);
7 }
Console
Clear

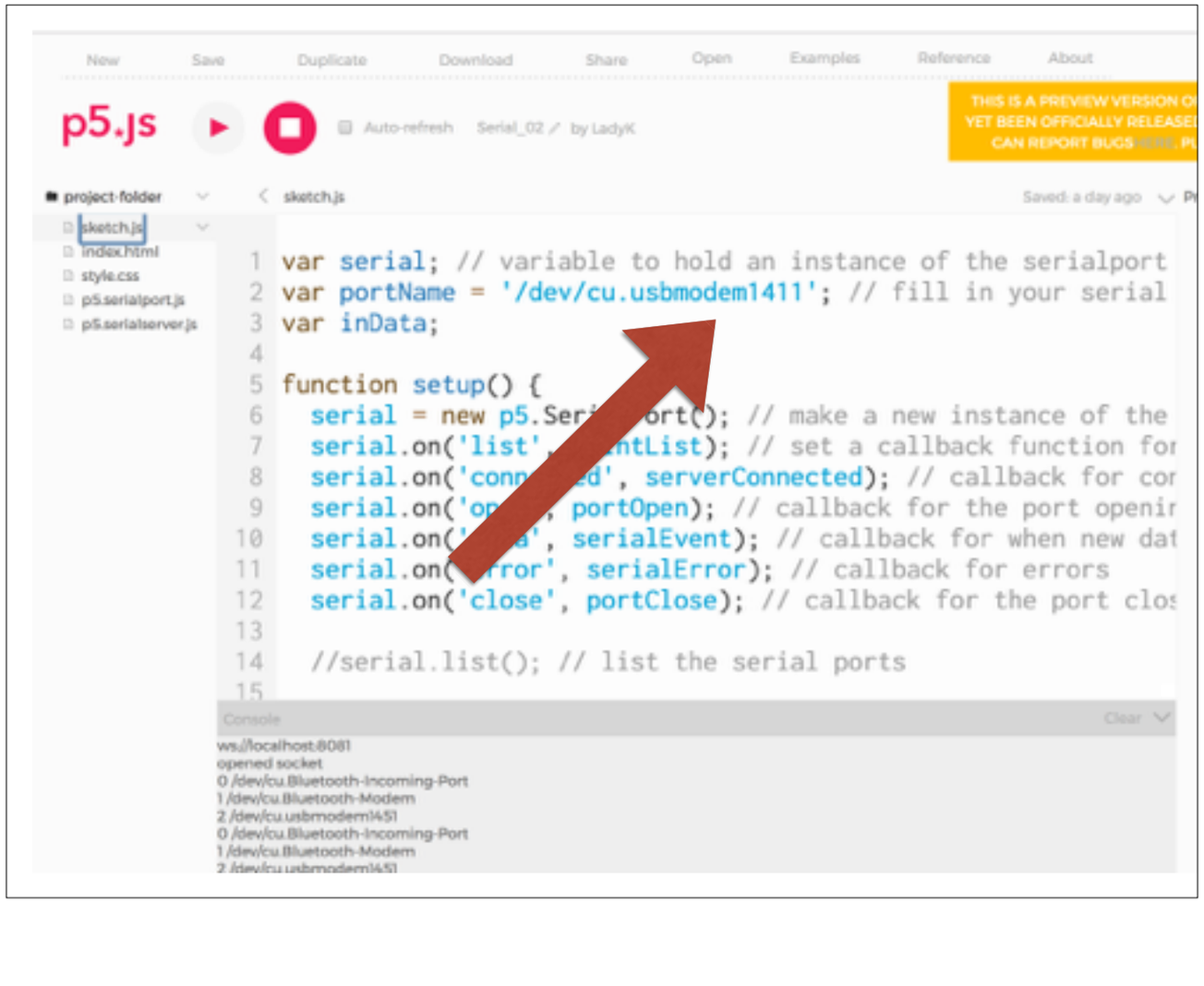


Serial example



Make a reference in your index.html for the p5.serialPort.js library in your index file


New
Save
Duplicate
Download
Share
Open
Examples
Reference
About
p5.js
Auto-refresh
serial / by LadyK
THIS IS A PREVIEW VERSION OF
YET BEEN OFFICIALLY RELEASED
CAN REPORT BUGS
sketch.js
Saved: a day ago
sketch.js
index.html
style.css
p5.serialserver.js
p5.serialport.js
1 var serial; // variable to hold an instance of the serialport
2
3 function setup() {
4 serial = new p5.SerialPort(); // make a new instance of the
5 serial.on('list', printList); // set a callback function for
6
7 serial.list(); // list the serial ports
8
9
10 // get the list of ports:
11 function printList(portList) {
12 // portList is an array of serial port names
13 for (var i = 0; i < portList.length; i++) {
14 // Display the list the console:
15 console.log(i + " " + portList[i]);
Console
Clear
ws://localhost
opened socket
0 /dev/cu.Bluetooth-Incoming-Port
1 /dev/cu.Bluetooth-Modem
2 /dev/cu.usbmodem1451
0 /dev/cu.Bluetooth-Incoming-Port
1 /dev/cu.Bluetooth-Modem
2 /dev/cu.usbmodem1451



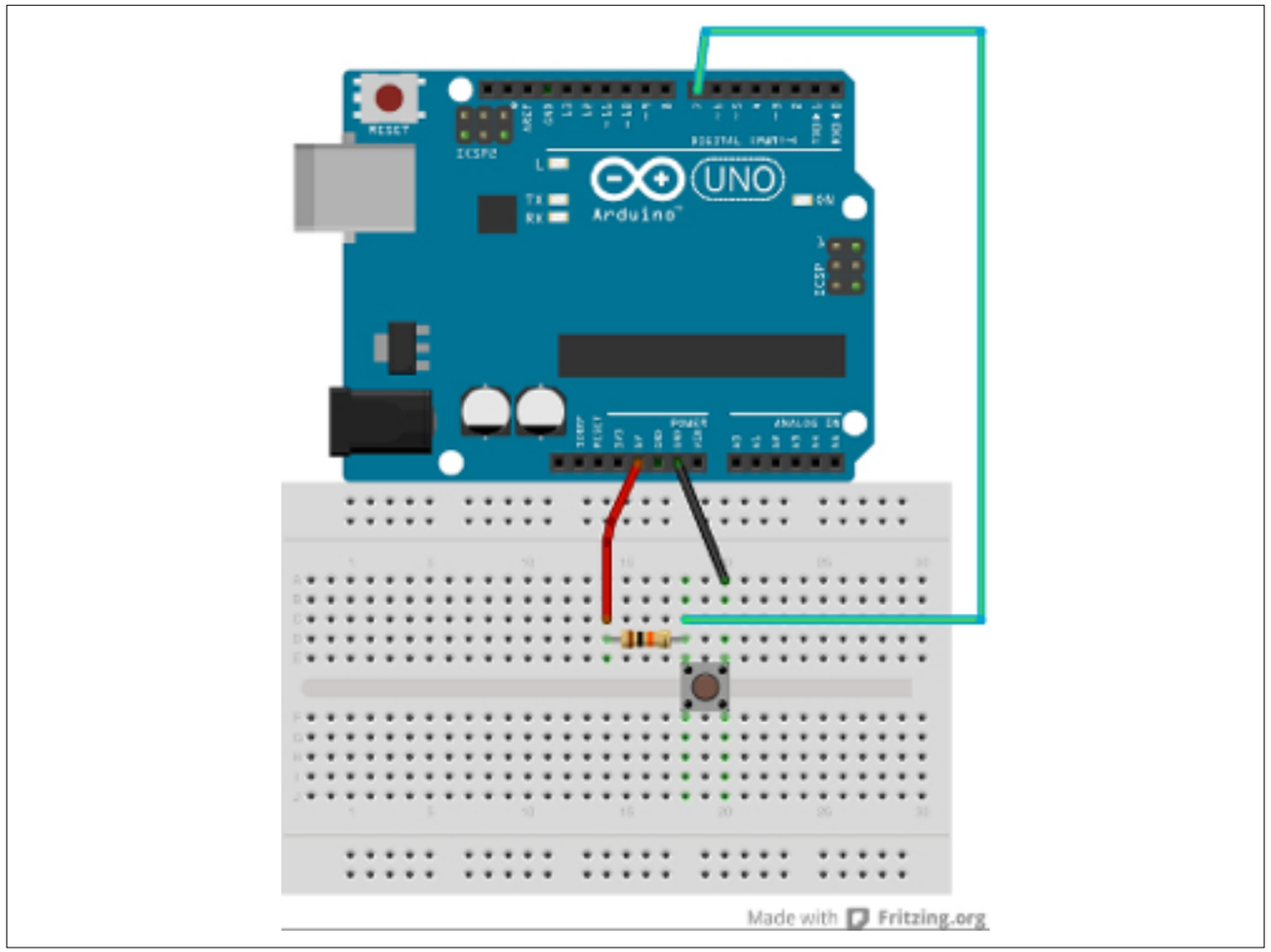
change your port name

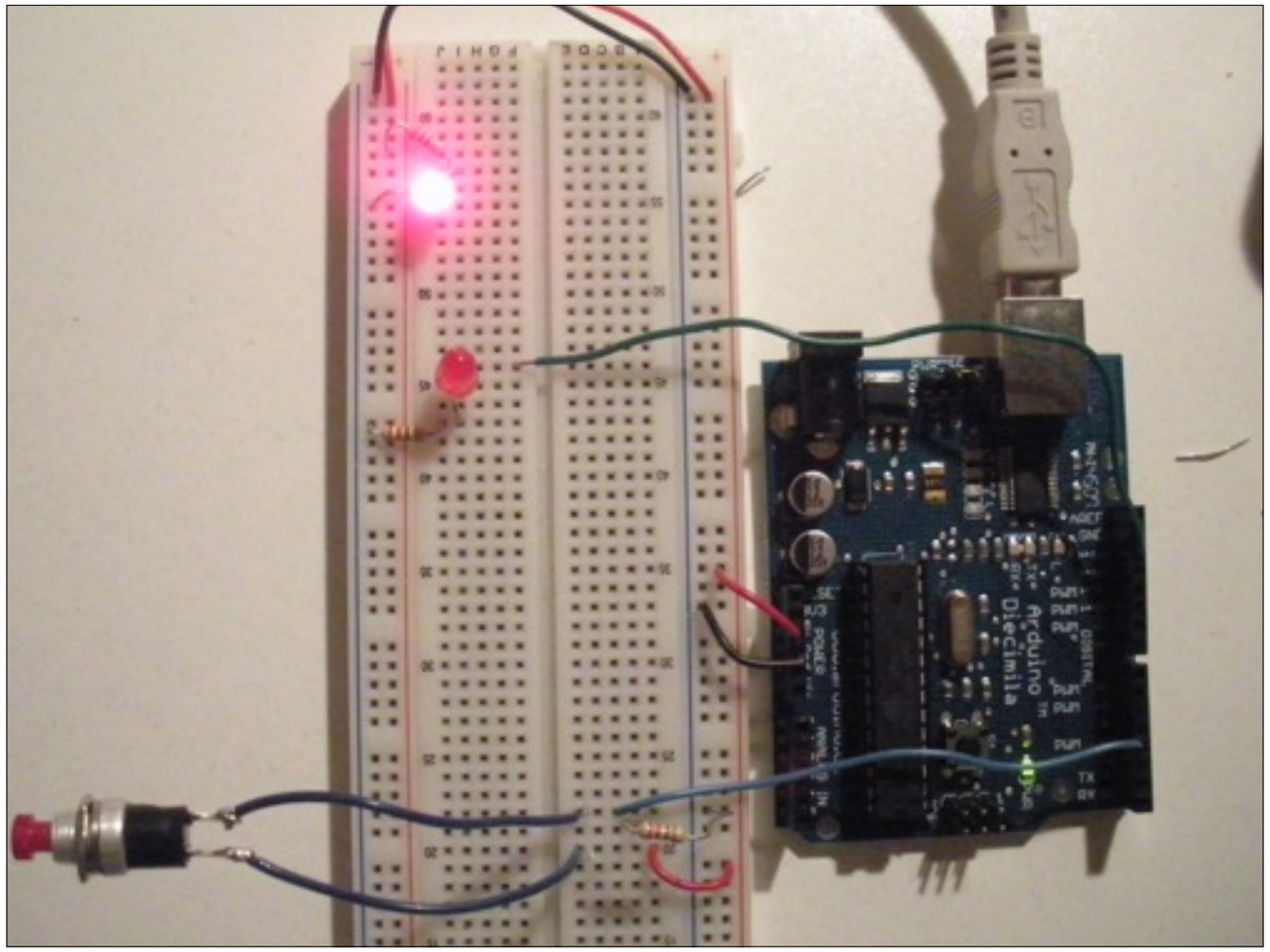
Input

- Digital switch

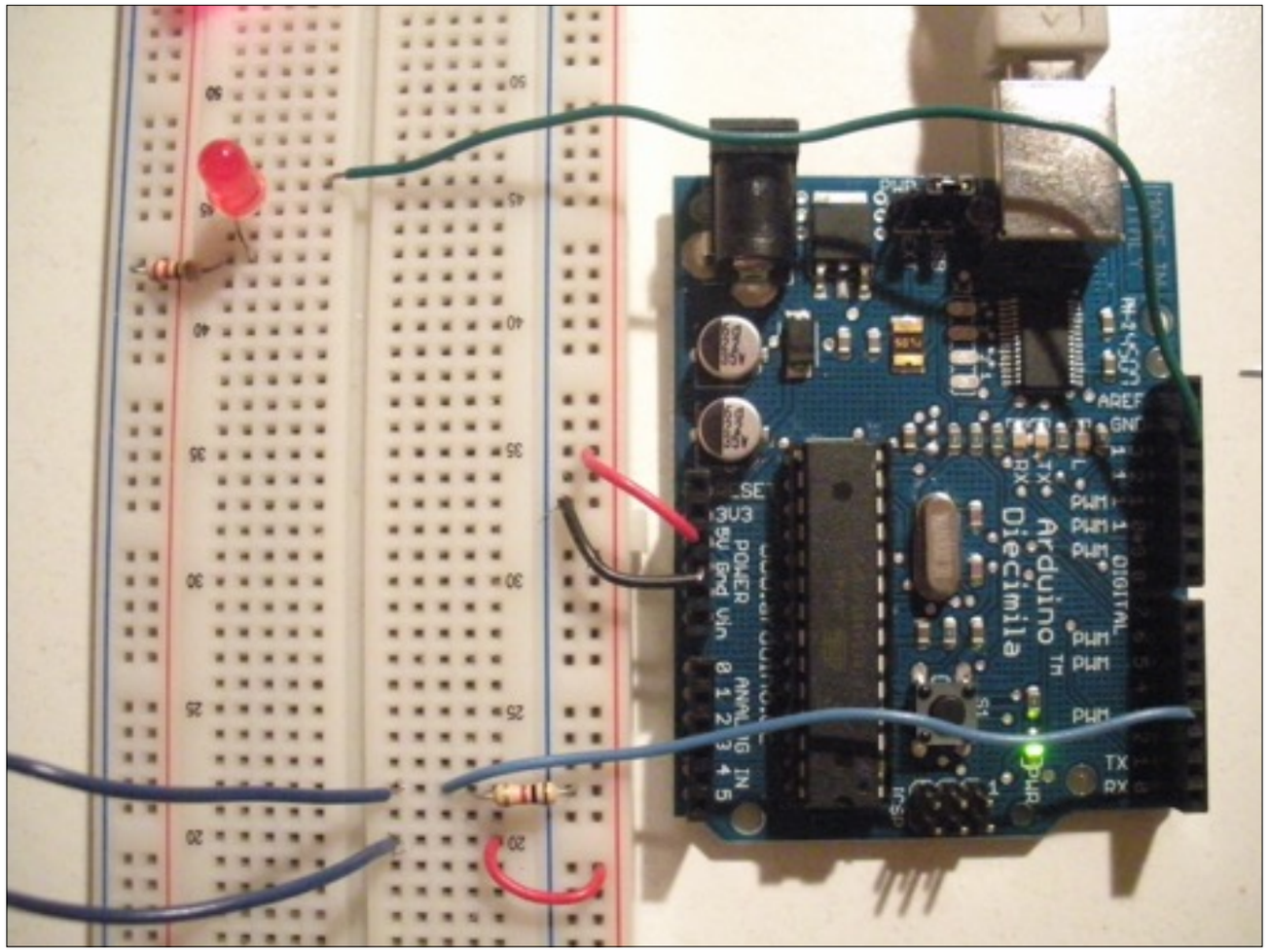
Burn the Digital Output code on the Arduino

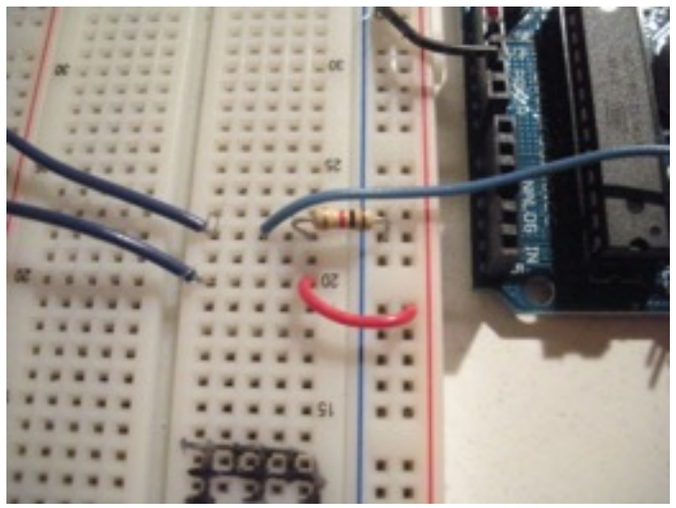
Start the p5 Serial Control App

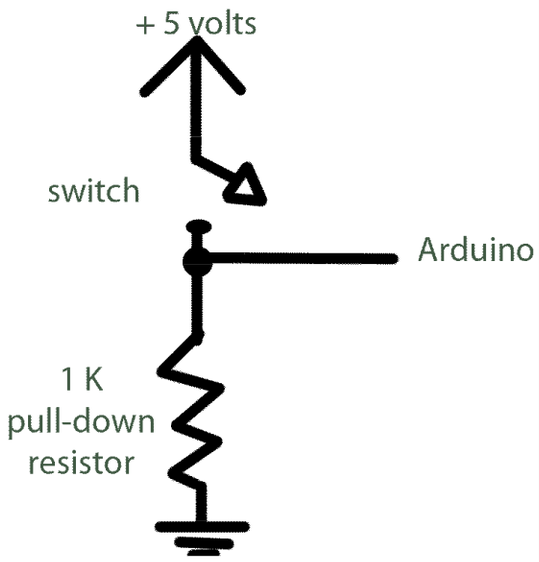

RESET
ICSP2
UNO
Arduino
ICSP
POWER
ANALOG IN
Made with Fritzing.org



Basic set-up for the switch: switch, 1k resistor, wires.

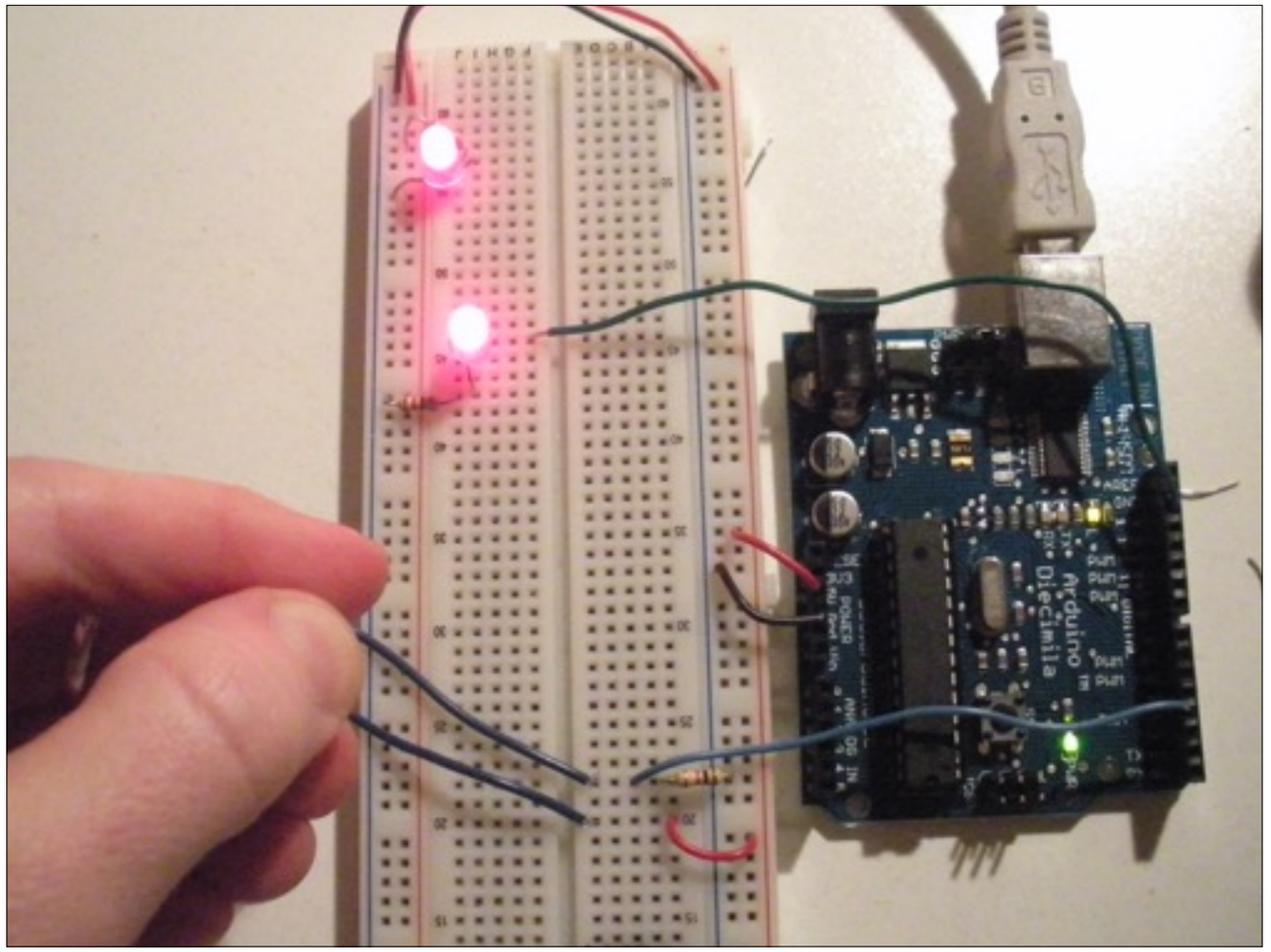

Arduino
Diecimila
POWER
ANALOG IN
DIGITAL
PWR
TX
RX



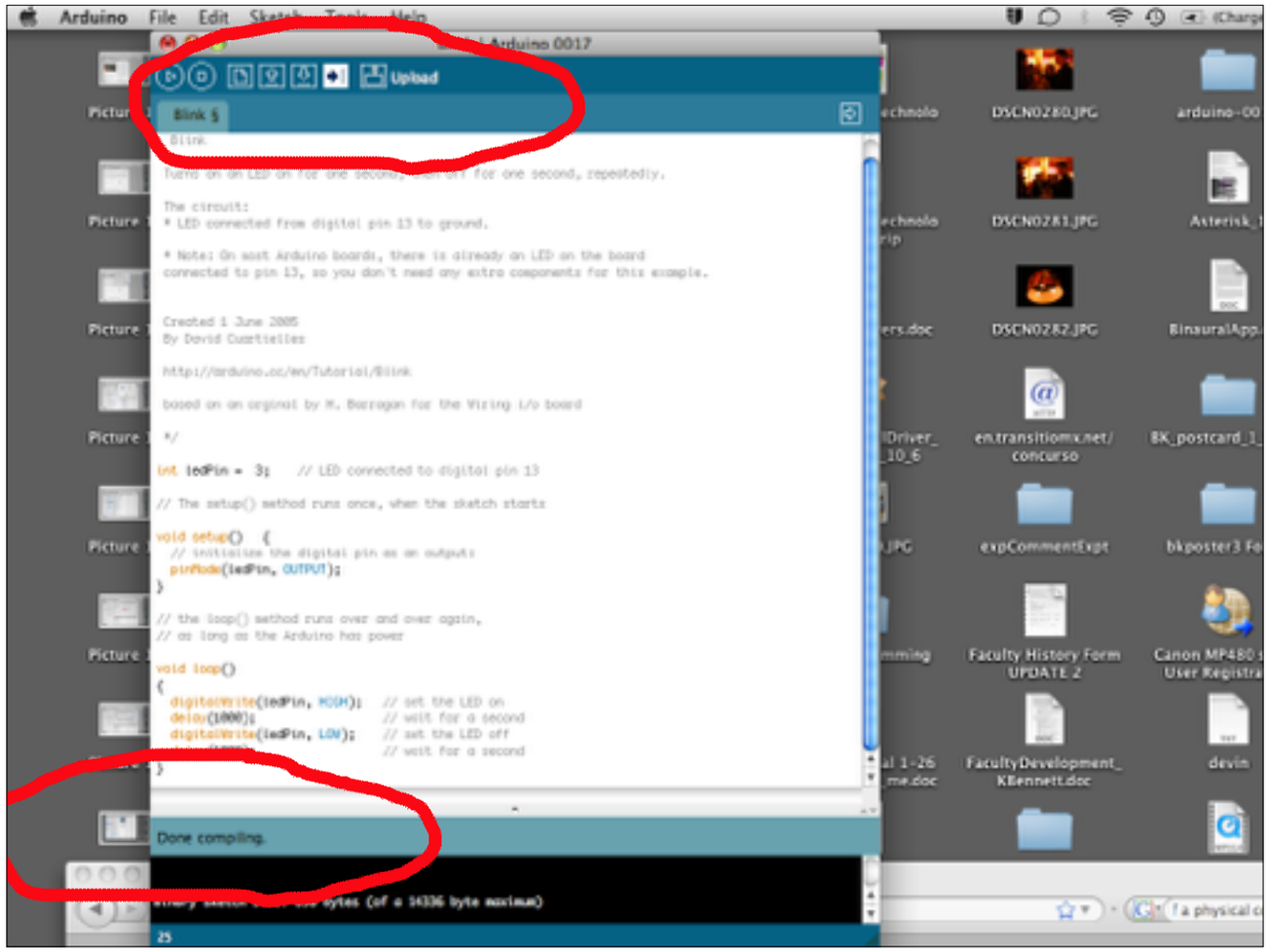


A Switch
Switches ALWAYS need a pull-down resistor. One side of this resistor goes between ground. The other side goes to the “node” where one end of the switch also connects to the wire going into our Arduino Board.
The other end of the switch is connected to the positive 5 volts.

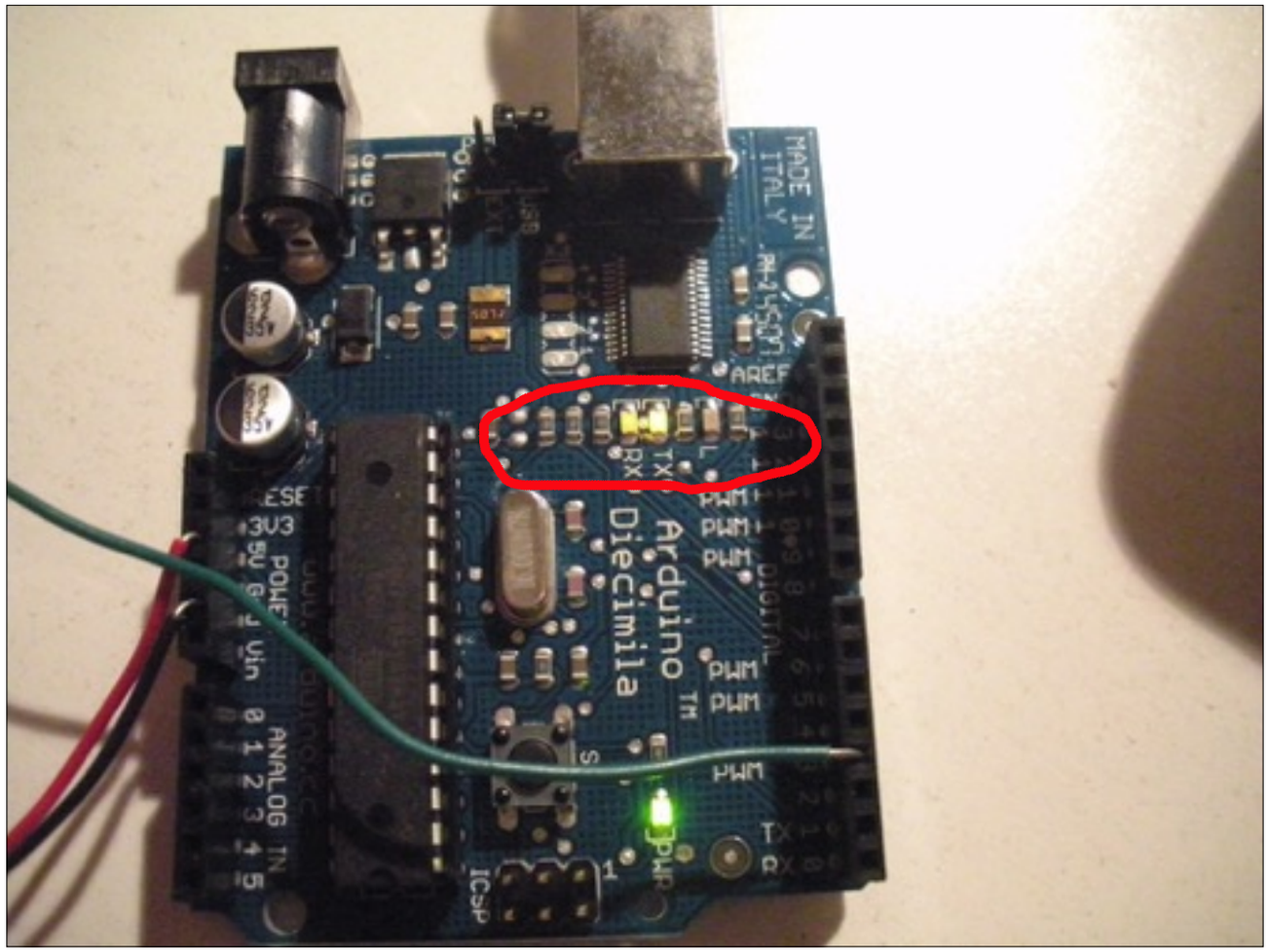
The resistor here prevents the Arduino from reading low voltages in the atmosphere.
Electrical current always flows from the path of least resistance to ground. If you just had a wire connecting the switch and the pin to ground, then, when you close the switch, the path of least resistance would be through the wire, and you’d have a short circuit. The easiest path for the current is through the closed switch. When the switch is open, the resistor offers the only path and the current goes through it.



Make sure you have the correct ports and chip selected in the Arduino Program.
Upload your button code. Make sure the code reflects the correct pins you've wired your button (input- listening) and led(output - talking) to.

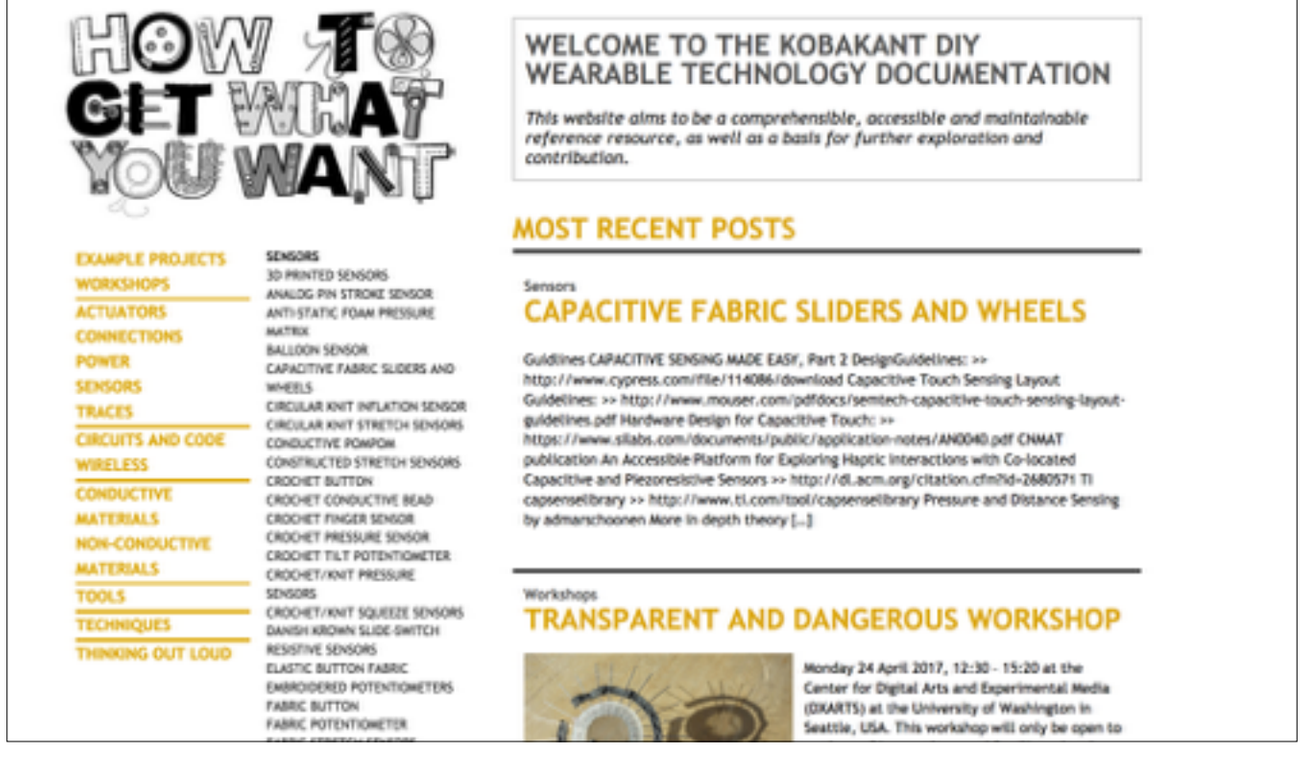


Let’s upload the code to the chip by pressing the upload button
at the top of your Arduino sketch window, on your computer.
This is now burning the code onto the chip.
Notice the flashing TX and RX lights on the Arduino board while this is happening.



Notice the flashing TX and RX lights on the Arduino board while this is happening.

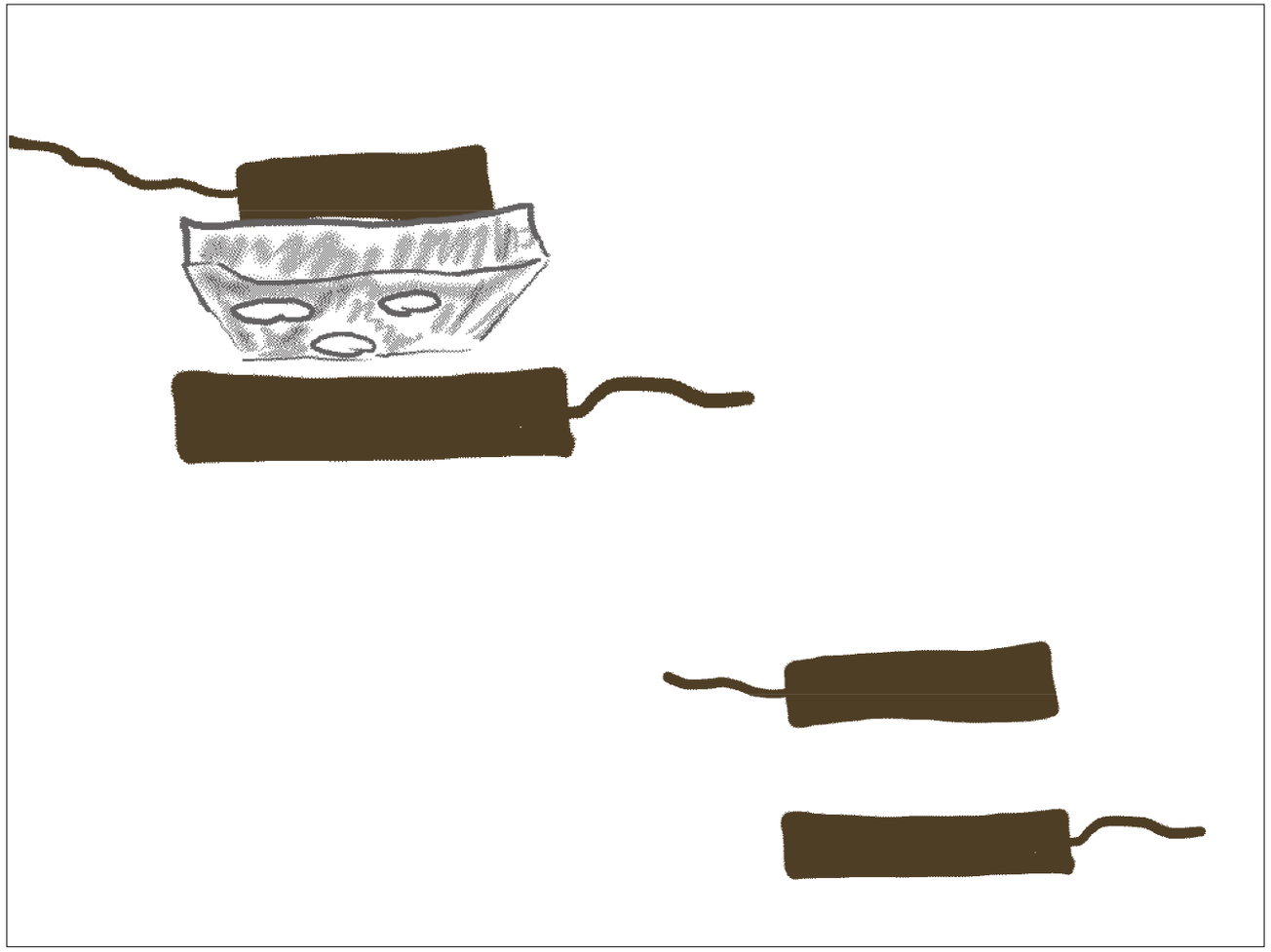
Custom DIY Switches

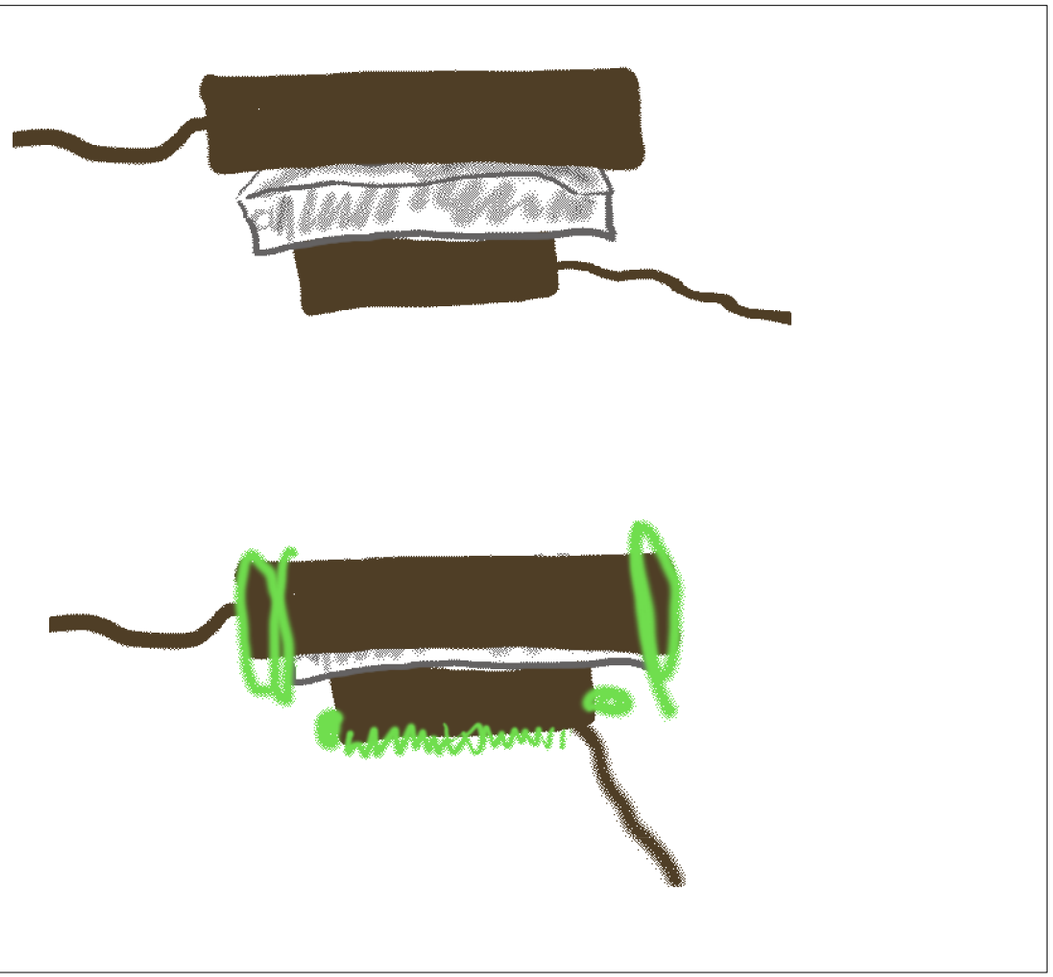


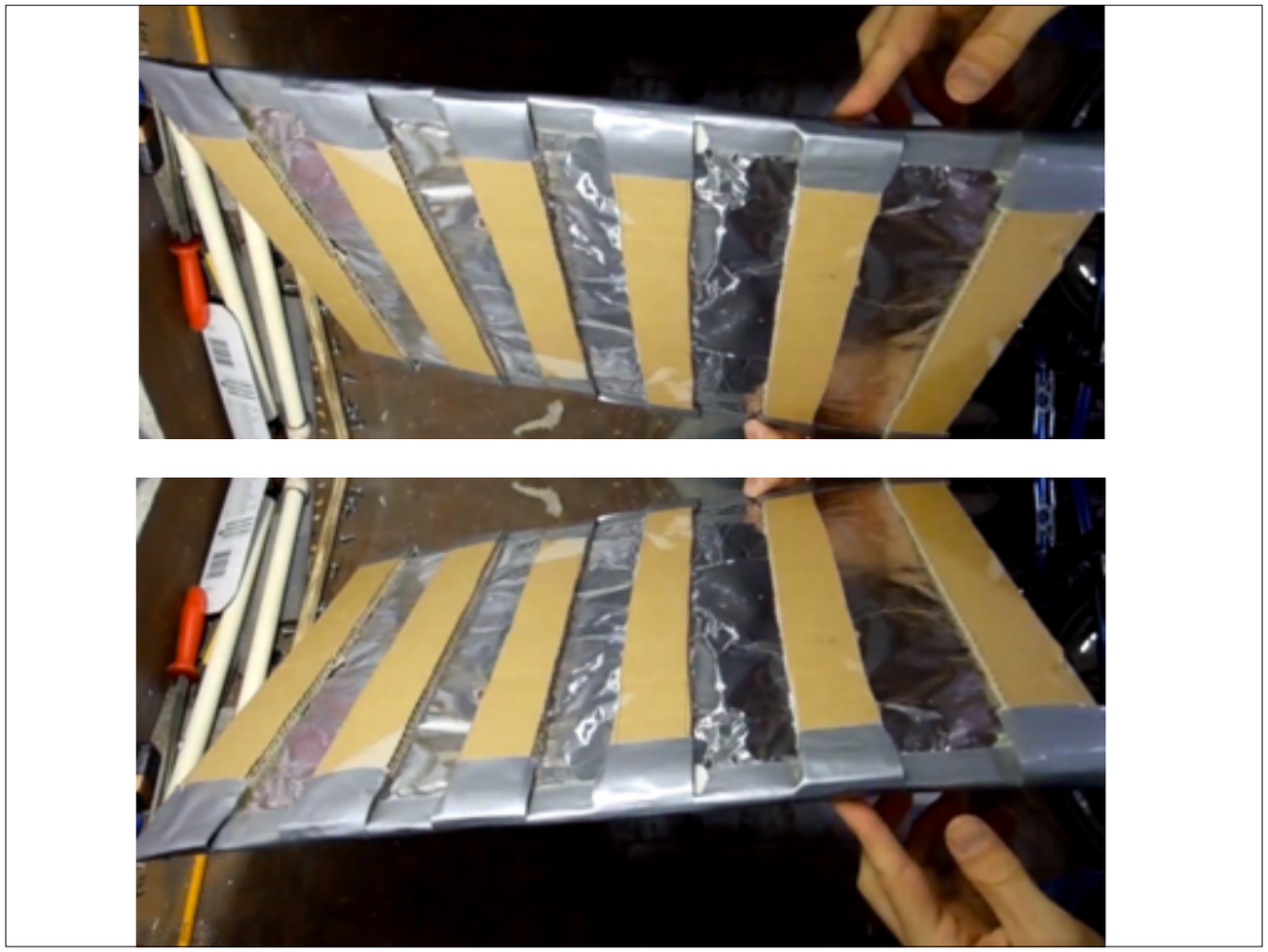
Custom DY Switches

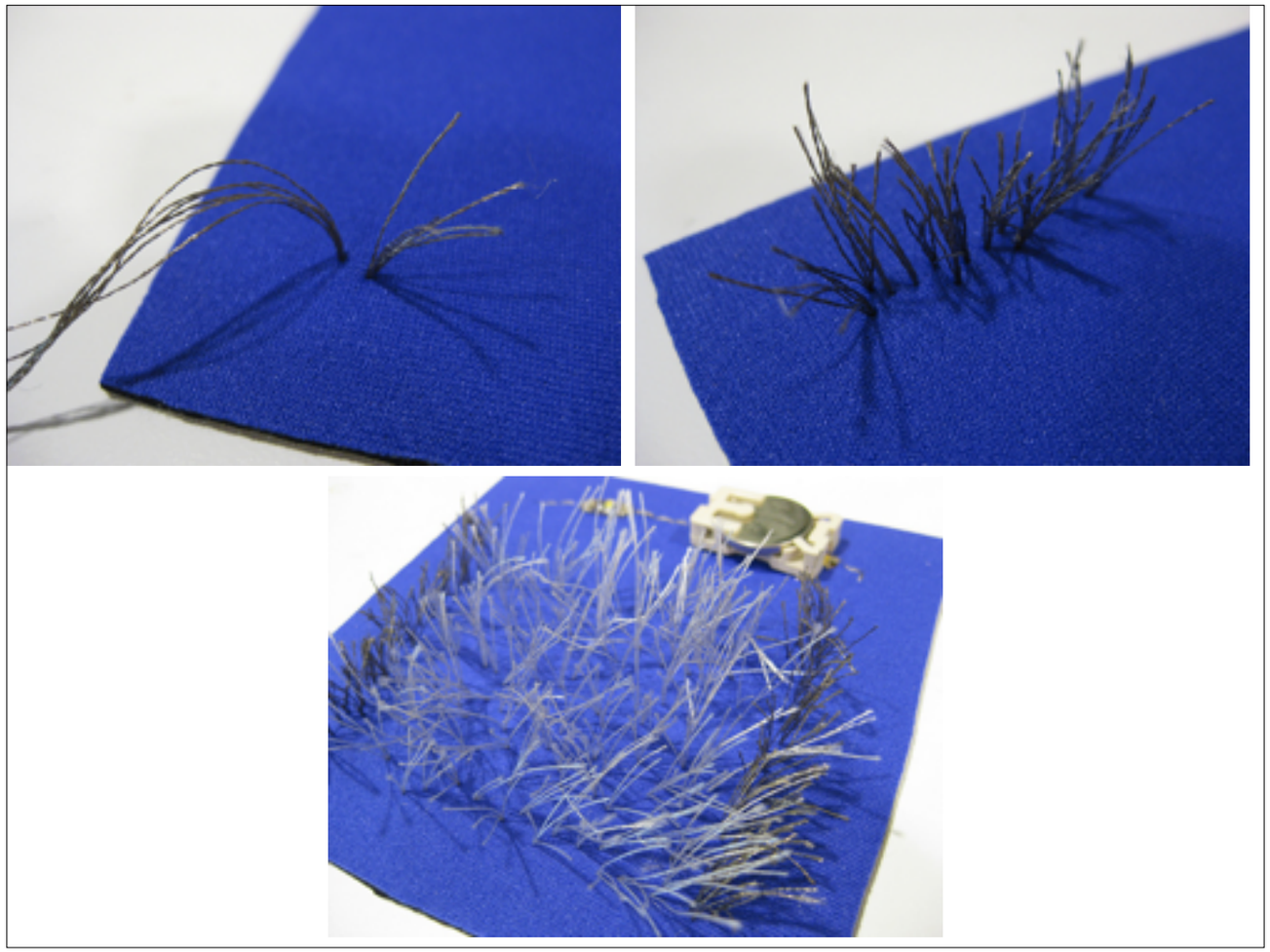
conductive materials

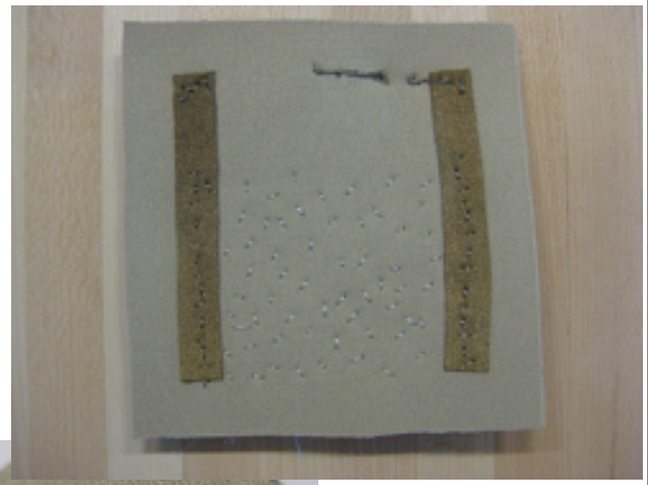
http://www.kobakant.at/DIY/

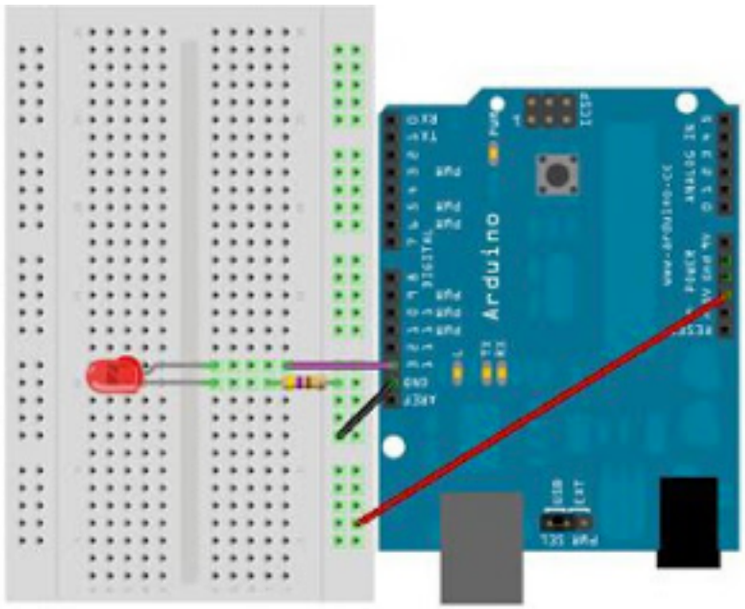




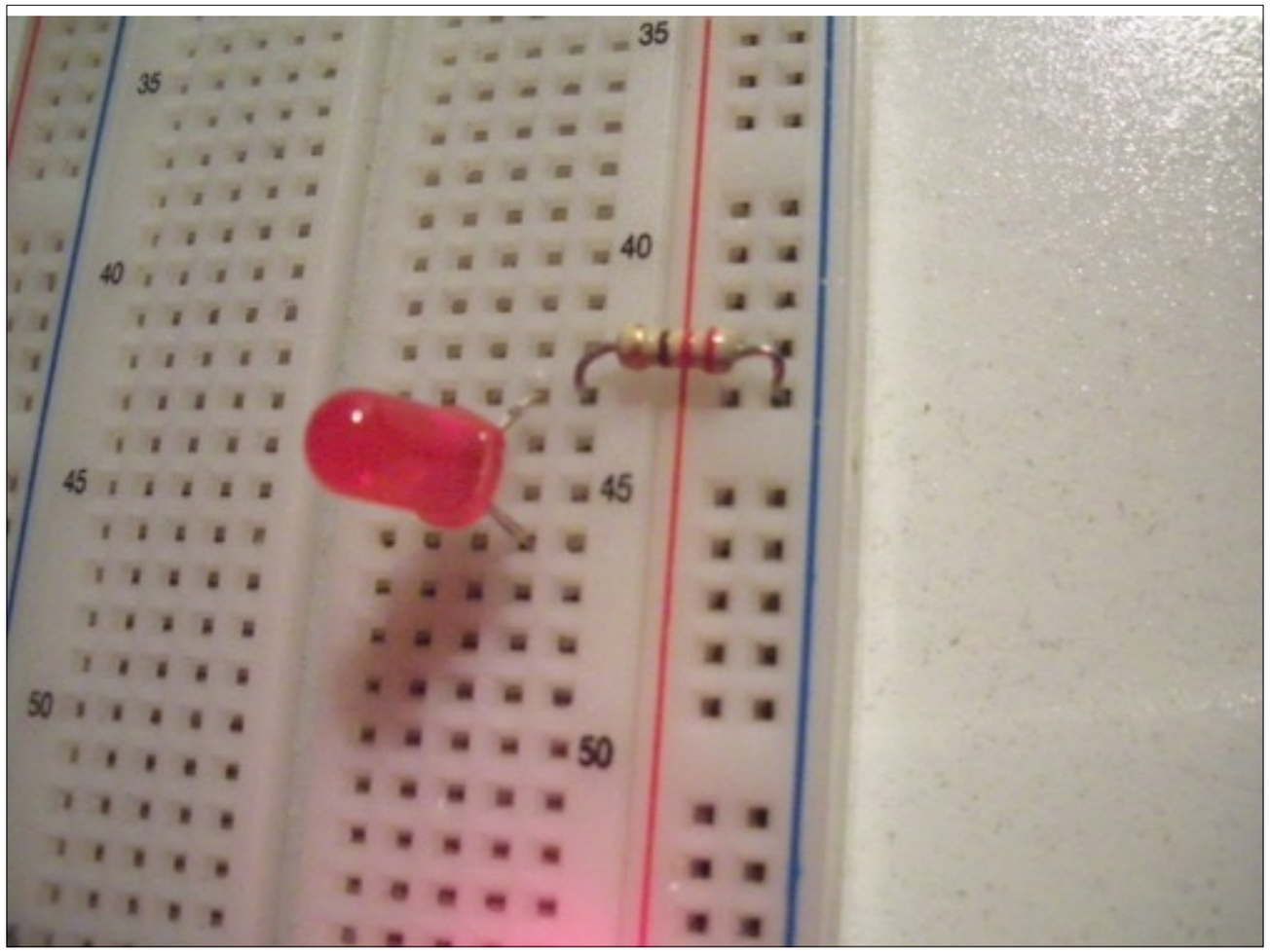




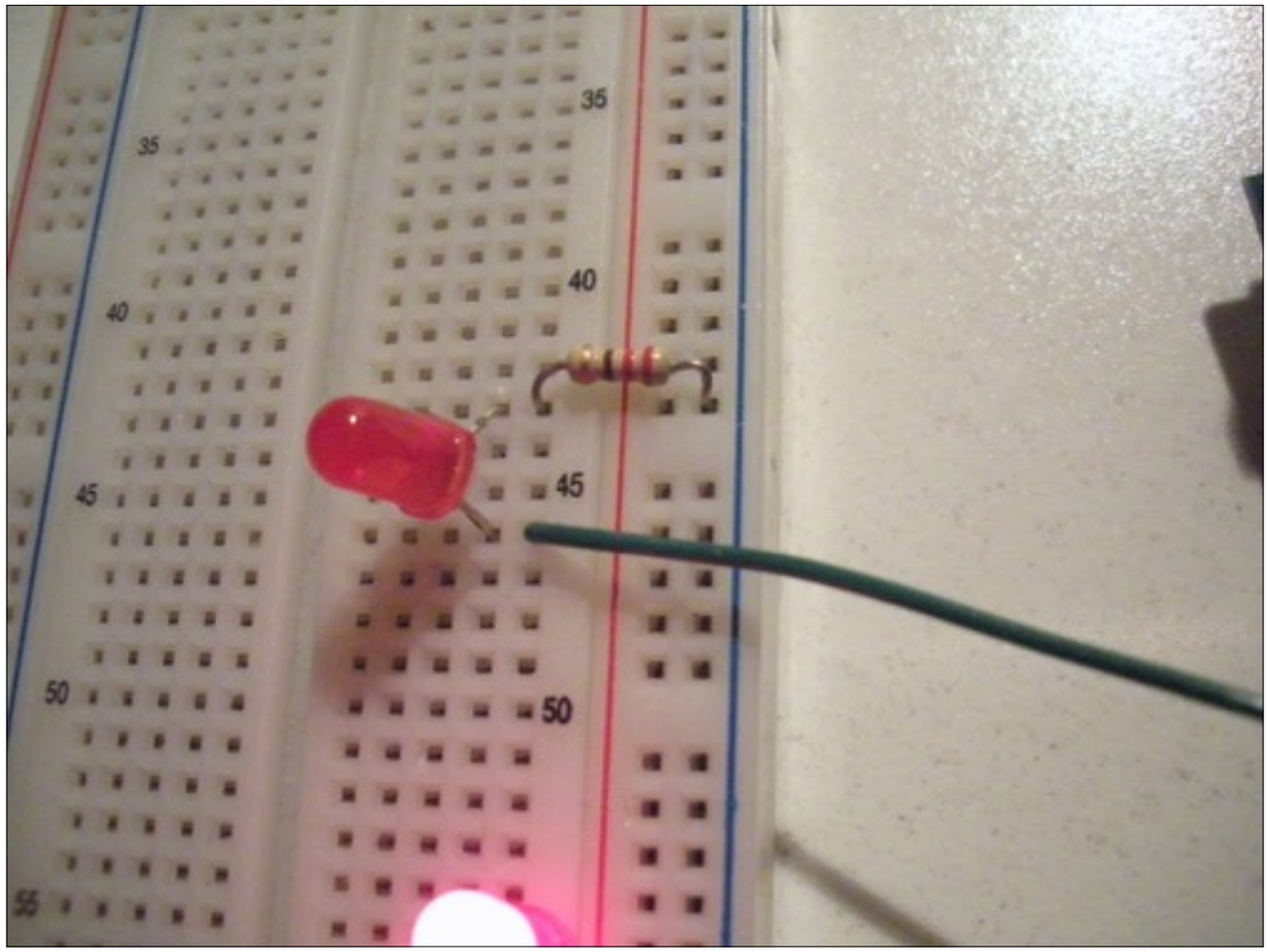




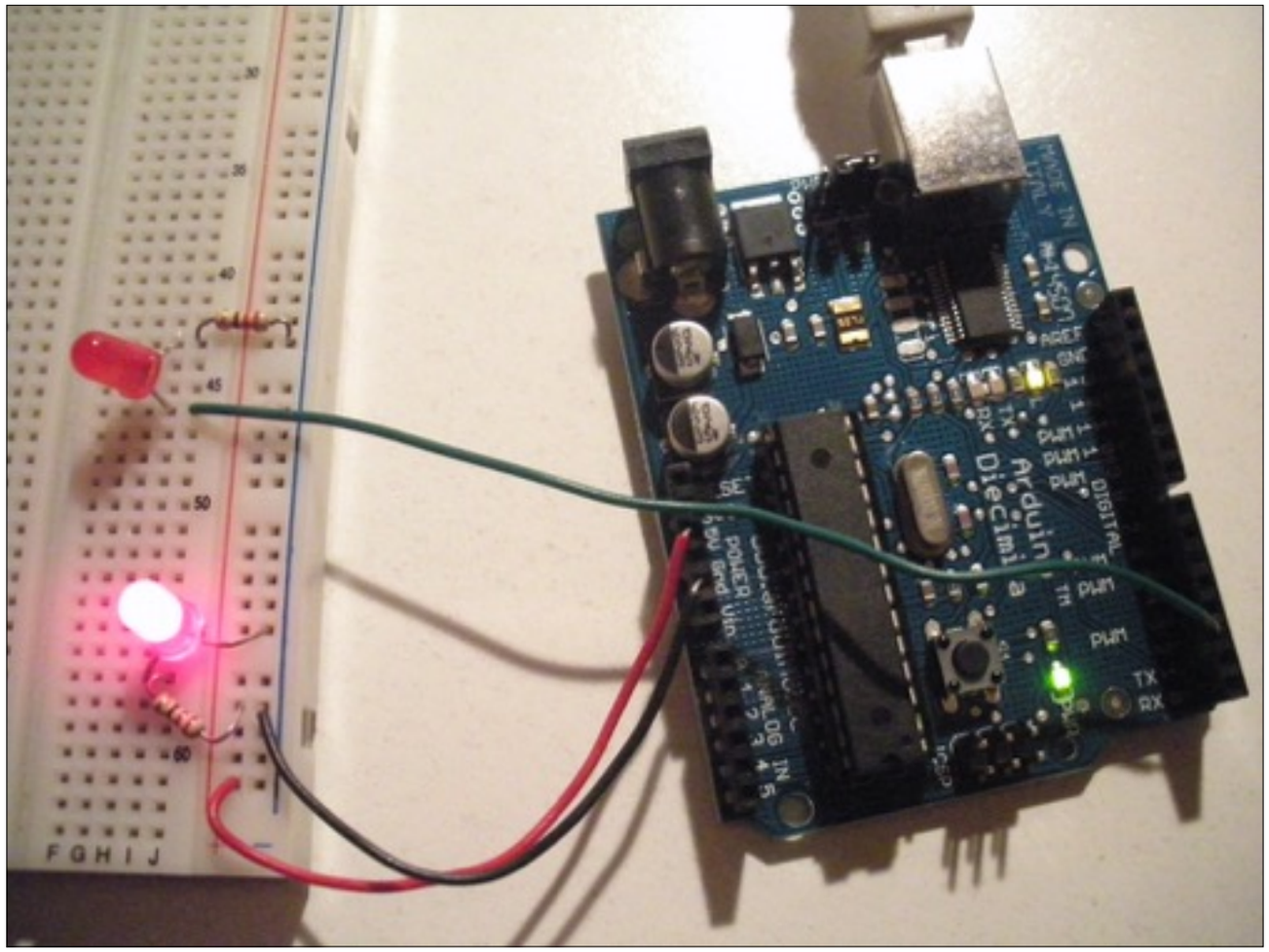
Wired for output



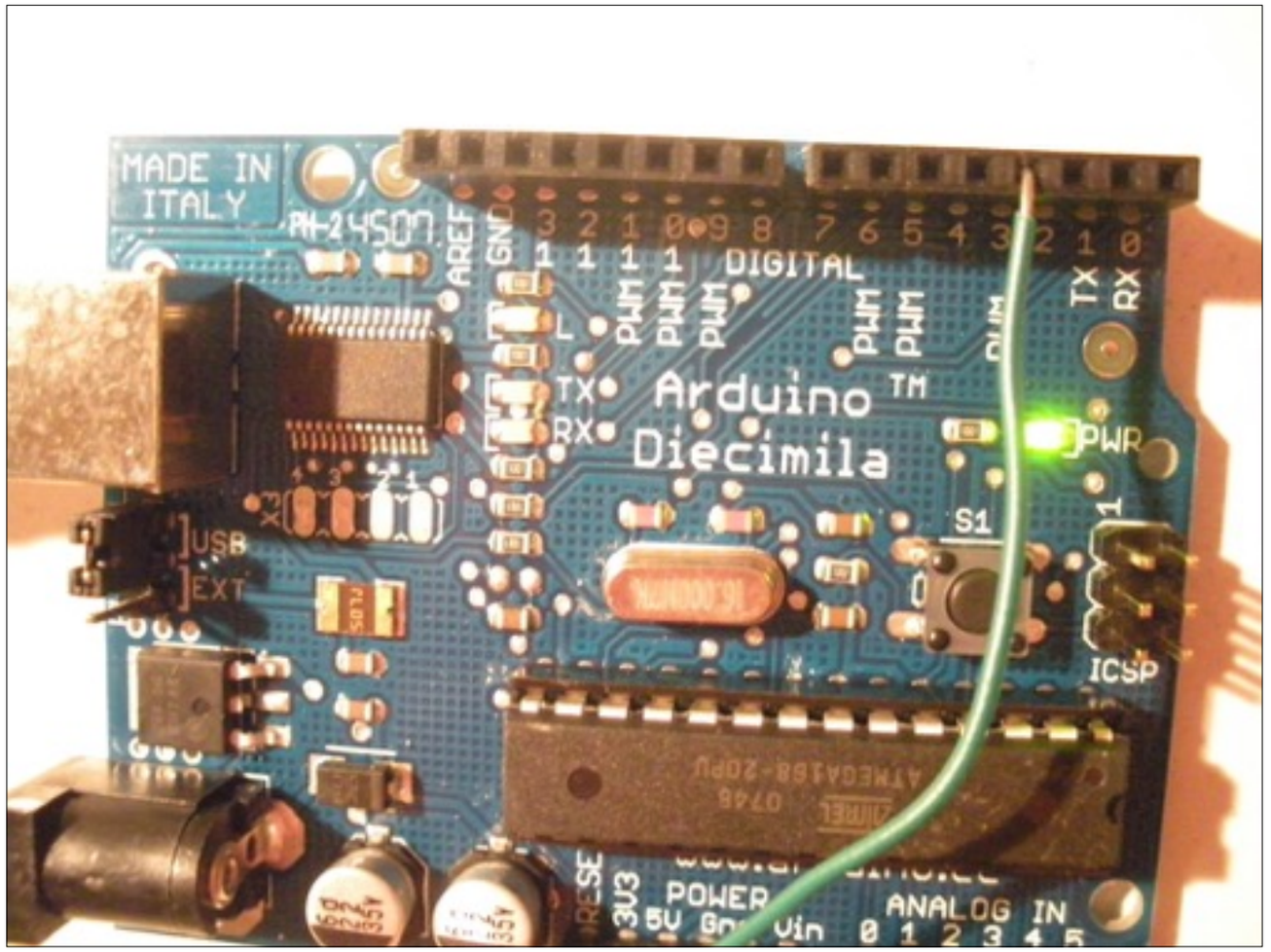
Straddle another led across a few rows of your breadboard. Doing so, will ensure that the two sides of the led aren't connected to each other.
Place 1 leg of a resistor into the same row as your led. Plug the other leg of that resistor into the ground side of your breadboard



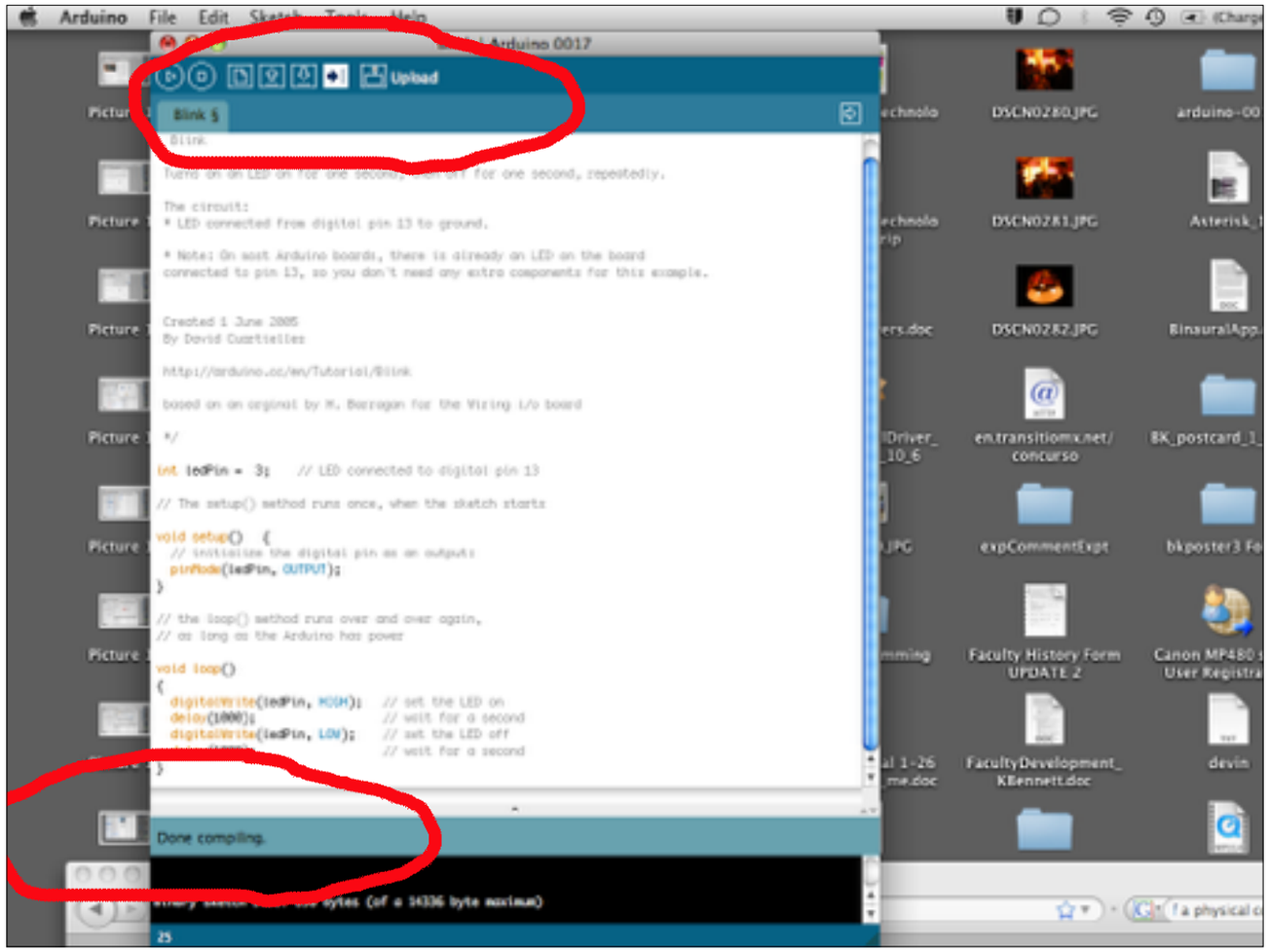
Plug a wire that is not red or black, into the side of the led that is unconnected.



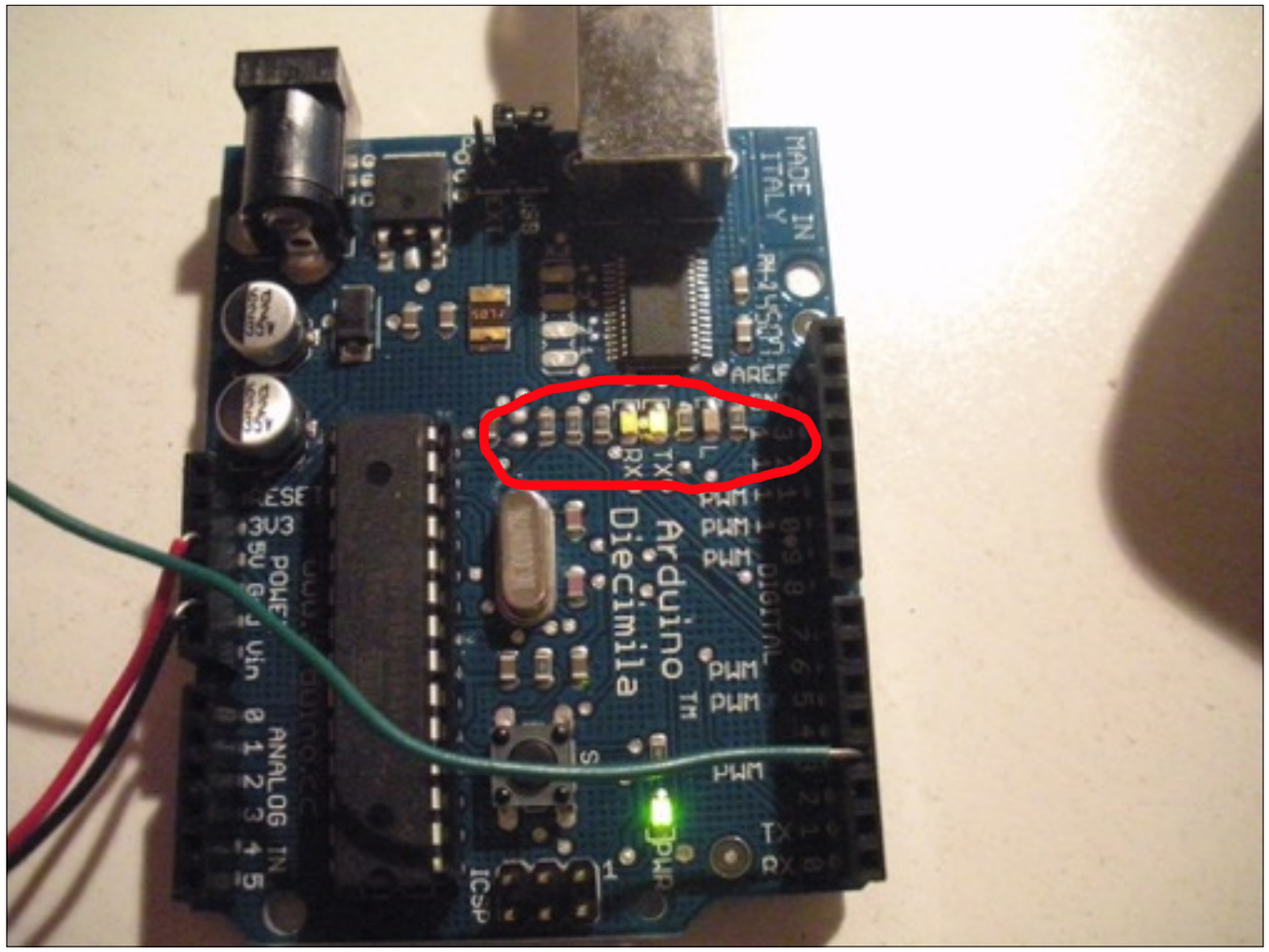
Plug the other side of that colored wire, into a DIGITAL pin on your Arduino board.
Do not choose digital pins 0 or 1. I chose digital pin 3.



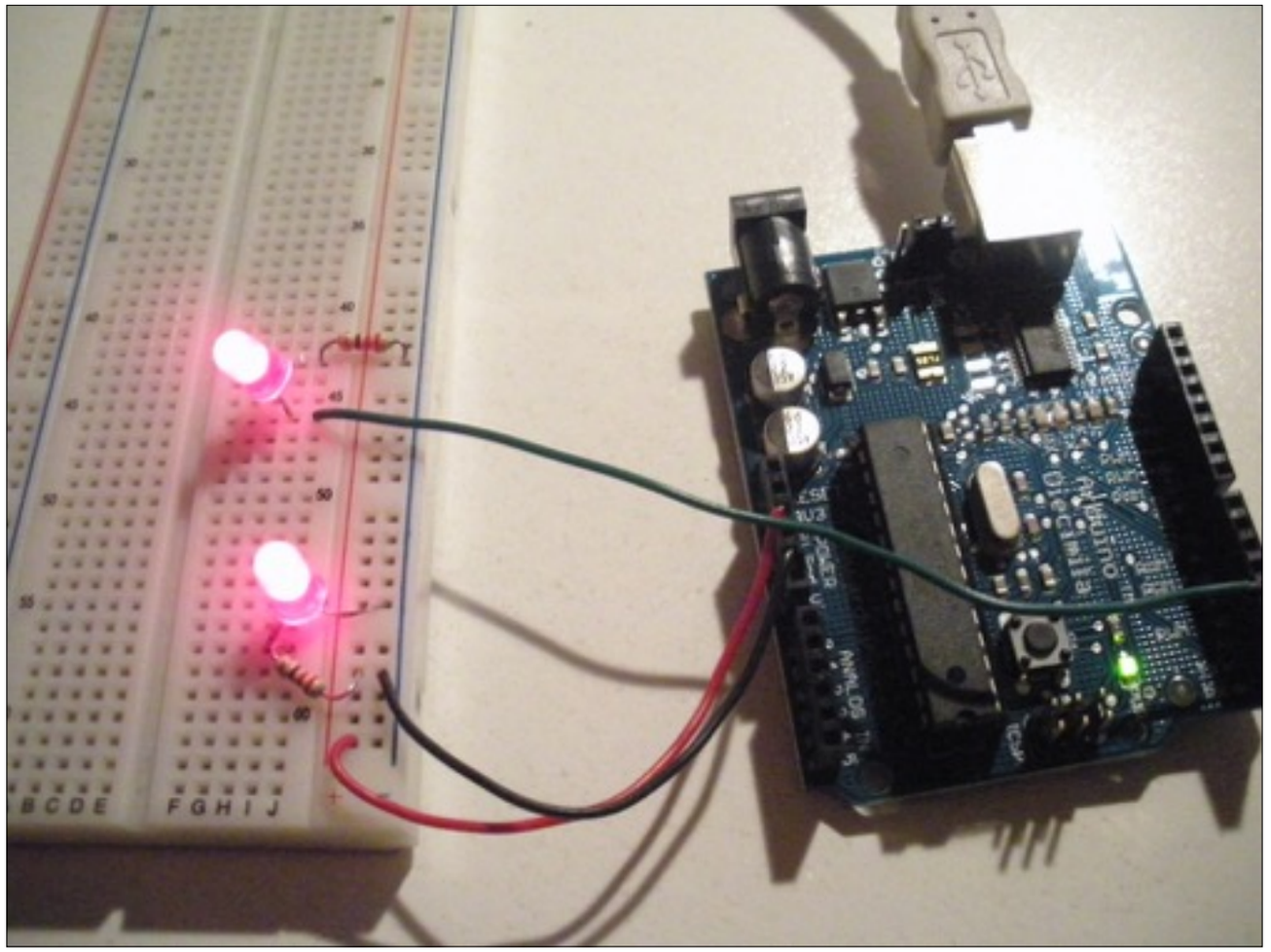
Plug the other side of that colored wire, into a DIGITAL pin on your Arduino board.
Do not choose digital pins 0 or 1. I chose digital pin 3.



Let’s upload the code to the chip by pressing the upload button
at the top of your Arduino sketch window, on your computer.
This is now burning the code onto the chip.
Notice the flashing TX and RX lights on the Arduino board while this is happening.



Notice the flashing TX and RX lights on the Arduino board while this is happening.



Now, your led, on your breadboard, should be flashing in the pattern that you programmed it.

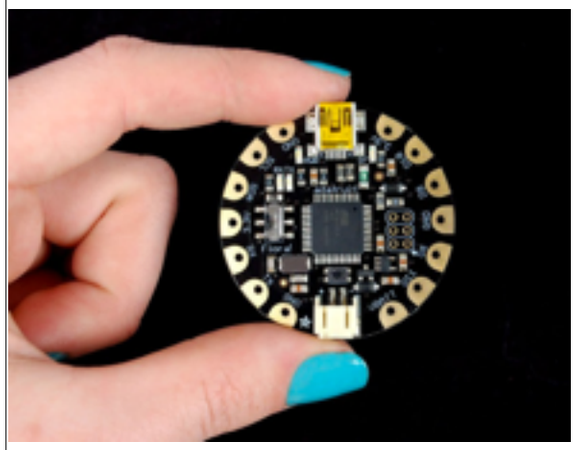
Review the code and try to have it flash in a sequence of 3 different patterns

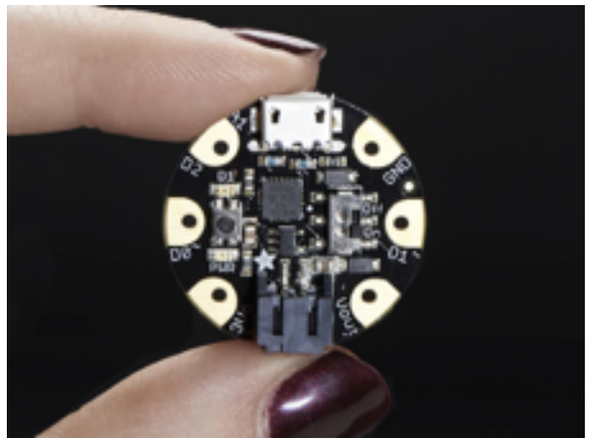
Next workshop, we'll add sensors to the arduino board, giving it ears.
And will give the circuit it's own power supply, so it doesn't have to use your computer's power.

More Info

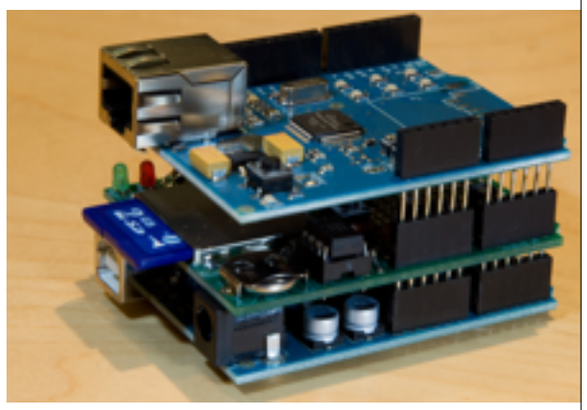
Arduino comes in many different flavors…

Such as the Gema

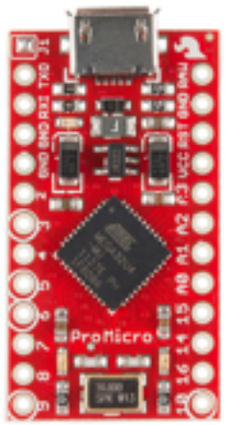


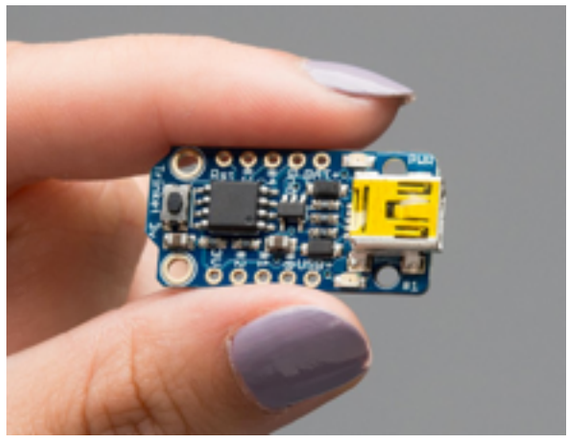


Gemma - The Attiny85 is a fun processor because despite being so small, it has 8K of flash, and 5 I/O pins, including analog inputs and PWM 'analog' outputs.
“Washable” & wearable



Shields
Extend functionality of the arduino
http://ece.jntuhceh.ac.in/openhardware/sites/default/files/shield%20arduino.jpg





Pro Micro
ATmega32U4 running at 5V/16MHz
ATMega 32U4 running at 3.3V/8MHz
4 x 10-bit ADC pins
12 x Digital I/Os (5 are PWM capable)
Rx and Tx Hardware Serial Connections

+ Trinket + ProTrinket
IO pins to play with; a total of 18, some of which can be analog, and only 32Kb of memory.
USB Device
5v and 3.3v versions,

ATmega328P onboad chip in QFN package
16MHz clock rate, 28K FLASH available
18 GPIO, 2 extra analog inputs, 28K of flash, and 2K of RAM.

Pro Mini + more

Resources

Web

Craft
http://www.craftzine.com

Fashioning Technology
http://www.fashioningtech.com

How To Get What You Want
http://www.kobakant.at/DIY/

Instructables
http://www.instructables.com

LilyPond
http://lilypond.media.mit.edu

Make
http://www.makezine.com

Soft Circuit Saturdays
http://softcircuitsaturdays.com/

talk2myShirt
http://www.talk2myshirt.com/

Graphic by Emily Lovell