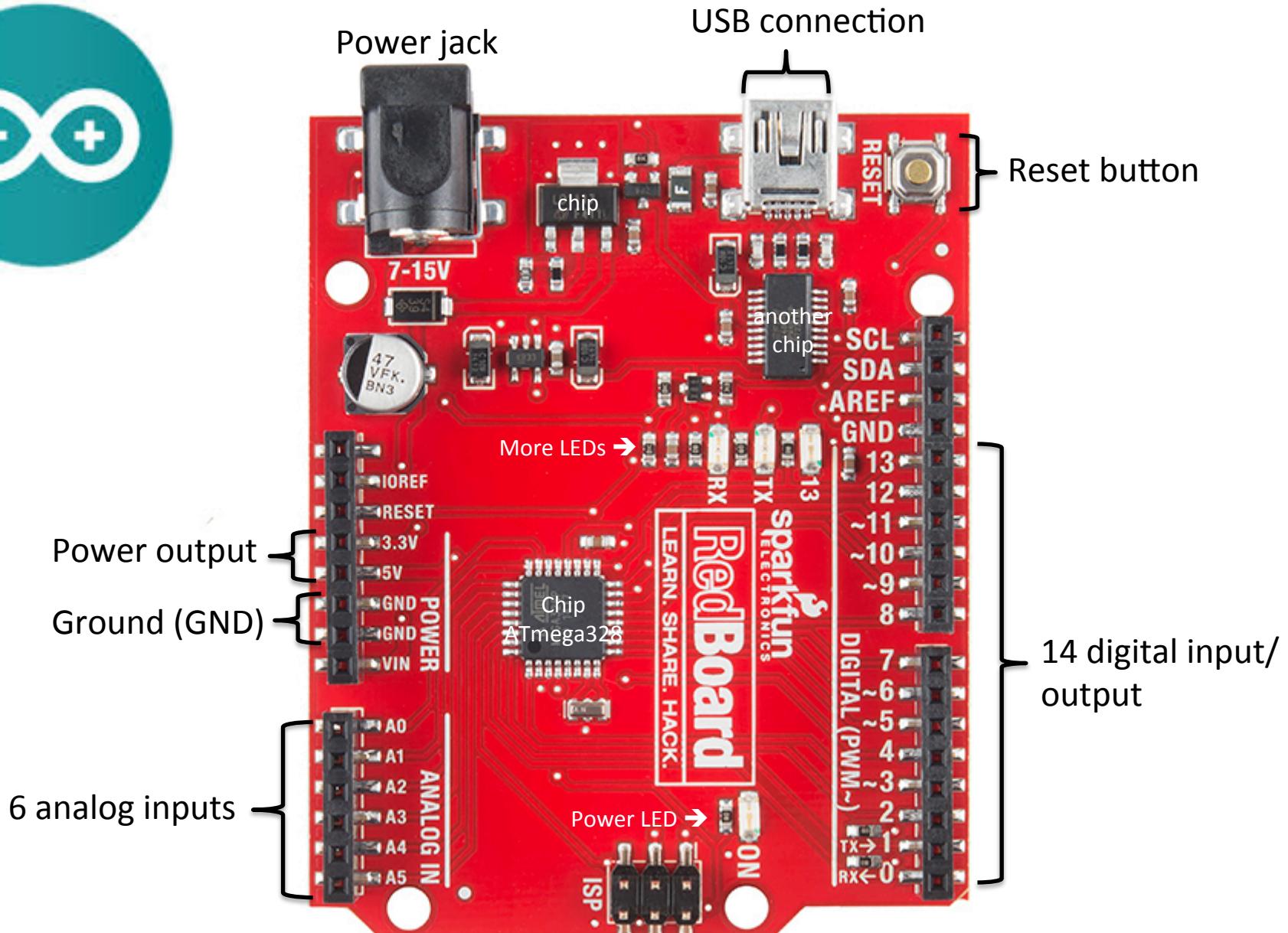
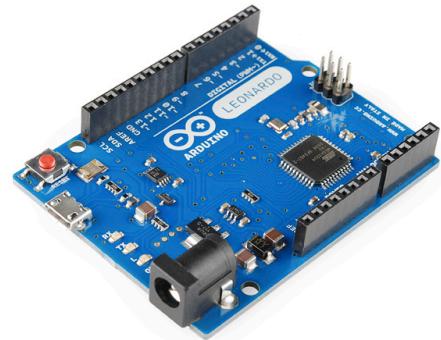
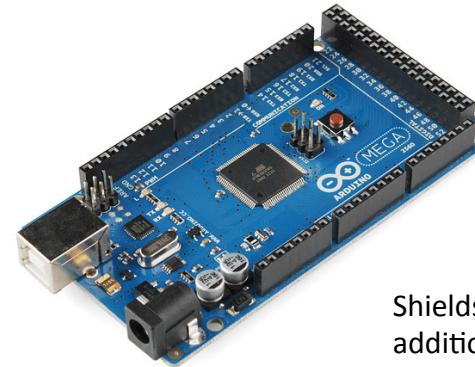
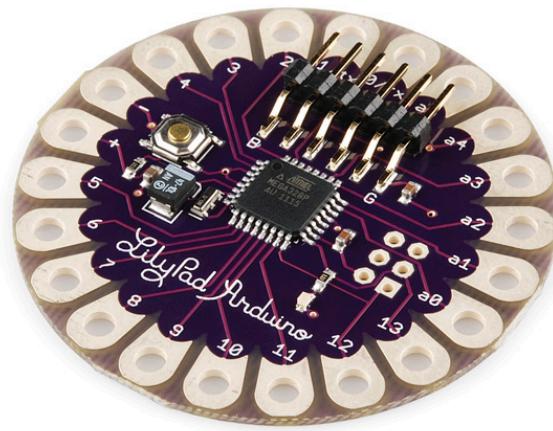
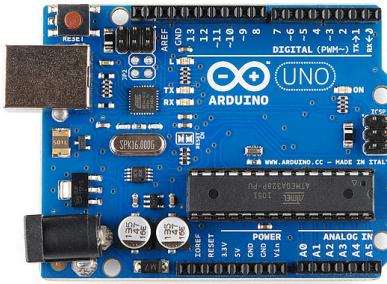


ARDUINO

A physical programmable circuit board, and a software to upload code to the board.



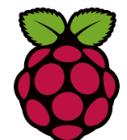
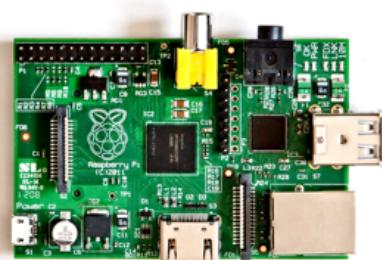
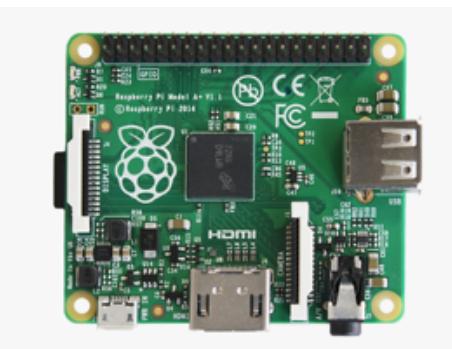
Arduino Family



Shields that provide additional capabilities

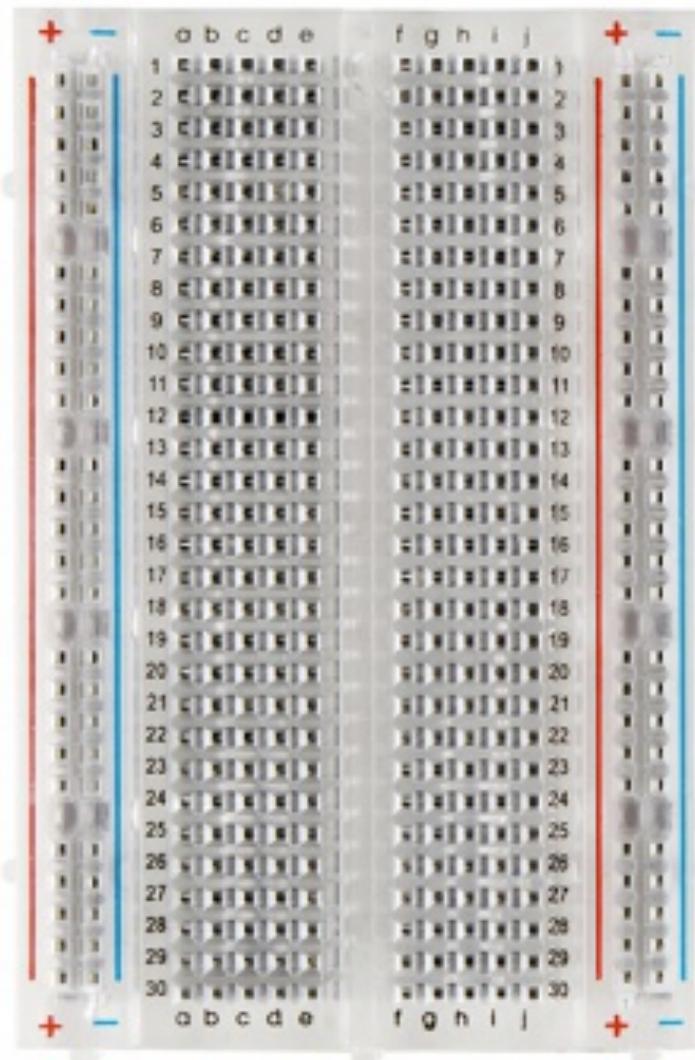


Raspberry Pi Family



Credit: wikipedia

How to use a breadboard



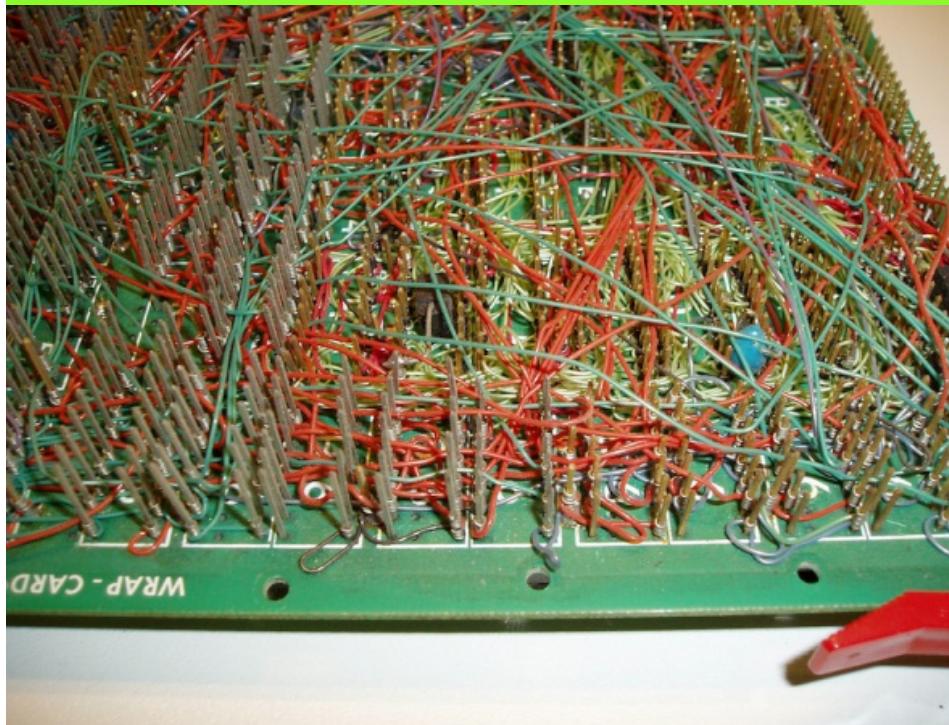
Power rails



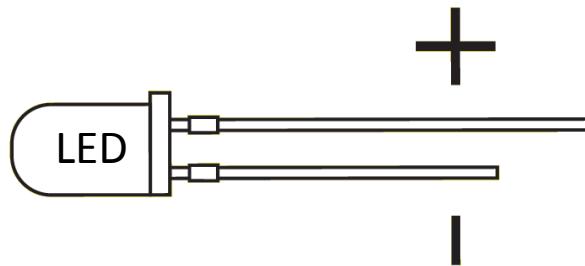
Power rails



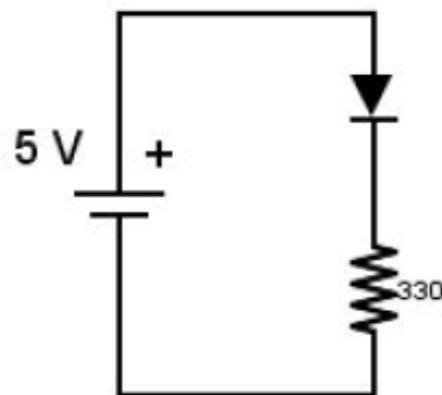
We use breadboards to avoid making a mess...



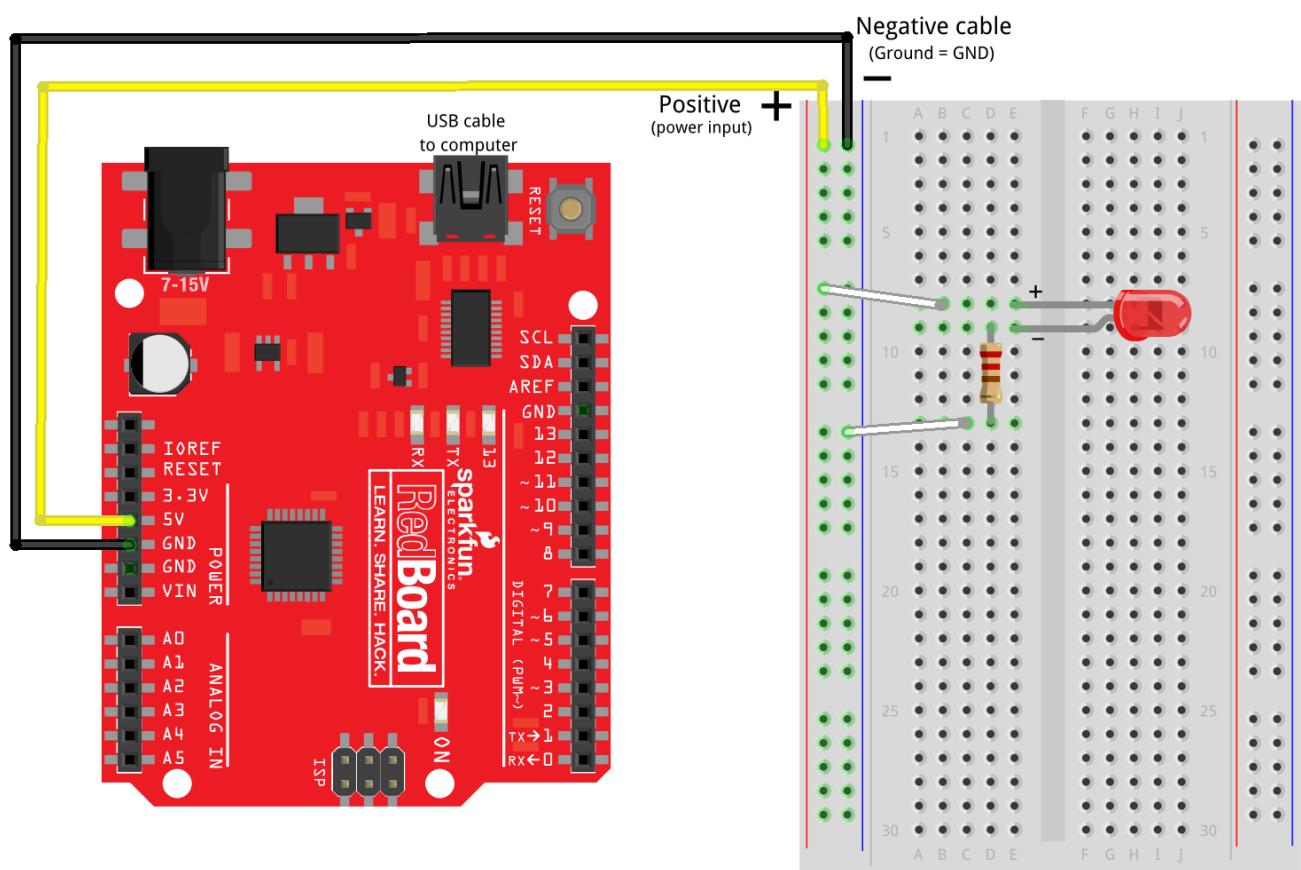
LED circuit: blink program



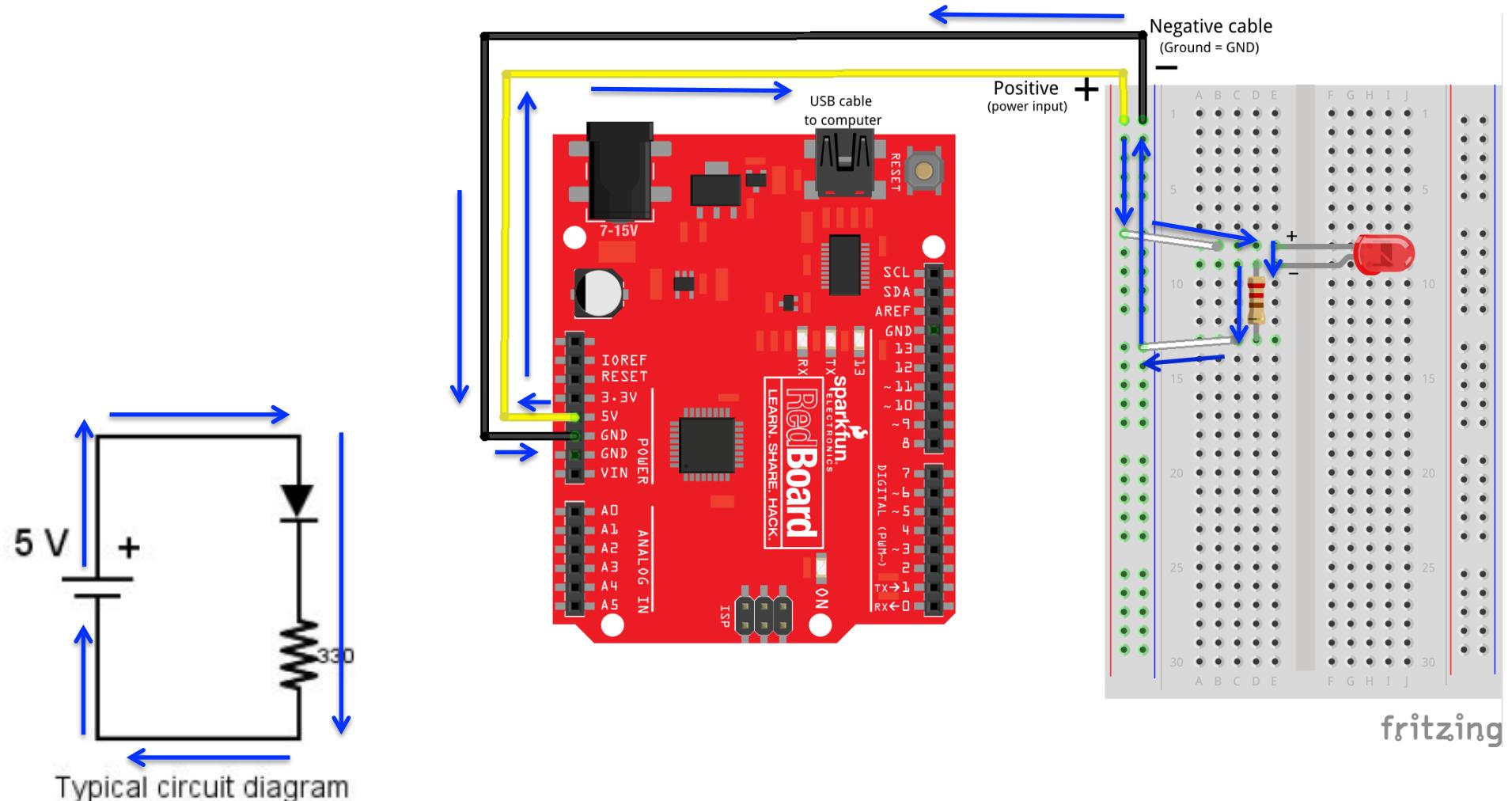
Resistor = 330Ω



Typical circuit diagram

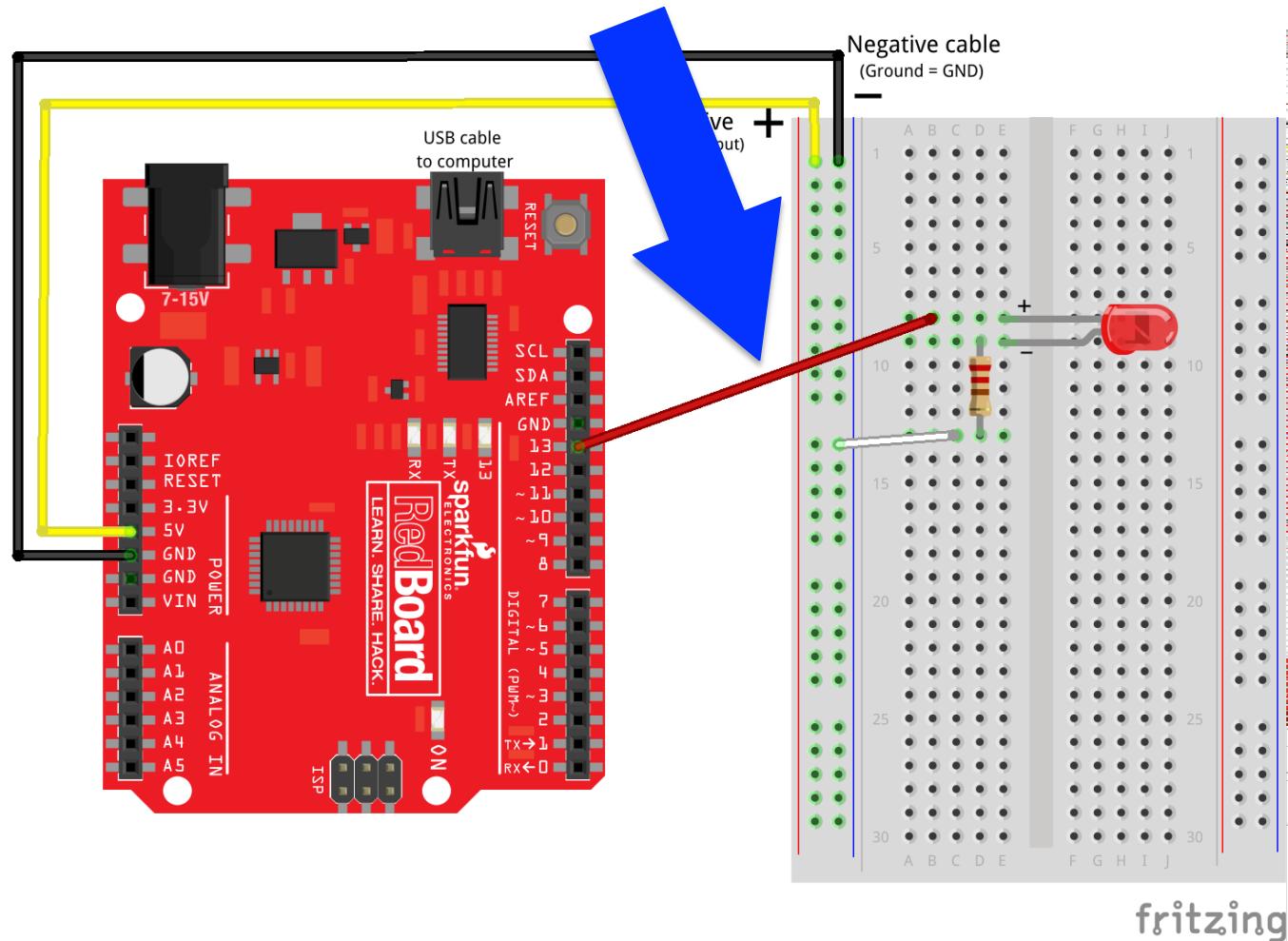


Direction of current = Direction of charge flow



When you do this correctly,
the LED will light up!

Now, move the connecting wire that you used to connect the LED to the power. Move this wire over to Digital I/O Pin 13.



Pin 13 behaves like a switched power source that
the Arduino can control!

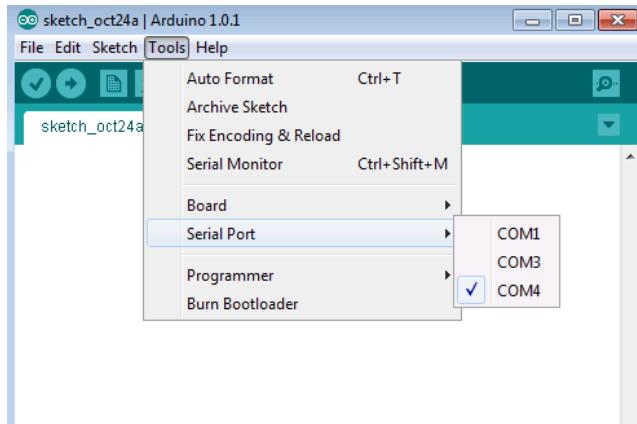
Arduino software: SKETCH of blink program



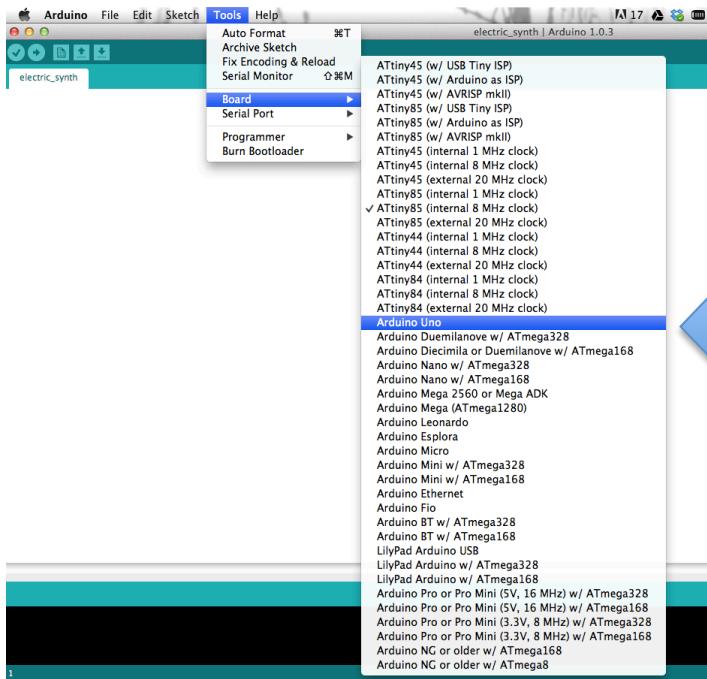
Open up the example sketch:
File → Examples → 01.Basic → Blink.

Check the **COM port settings:**
Tools → Port.

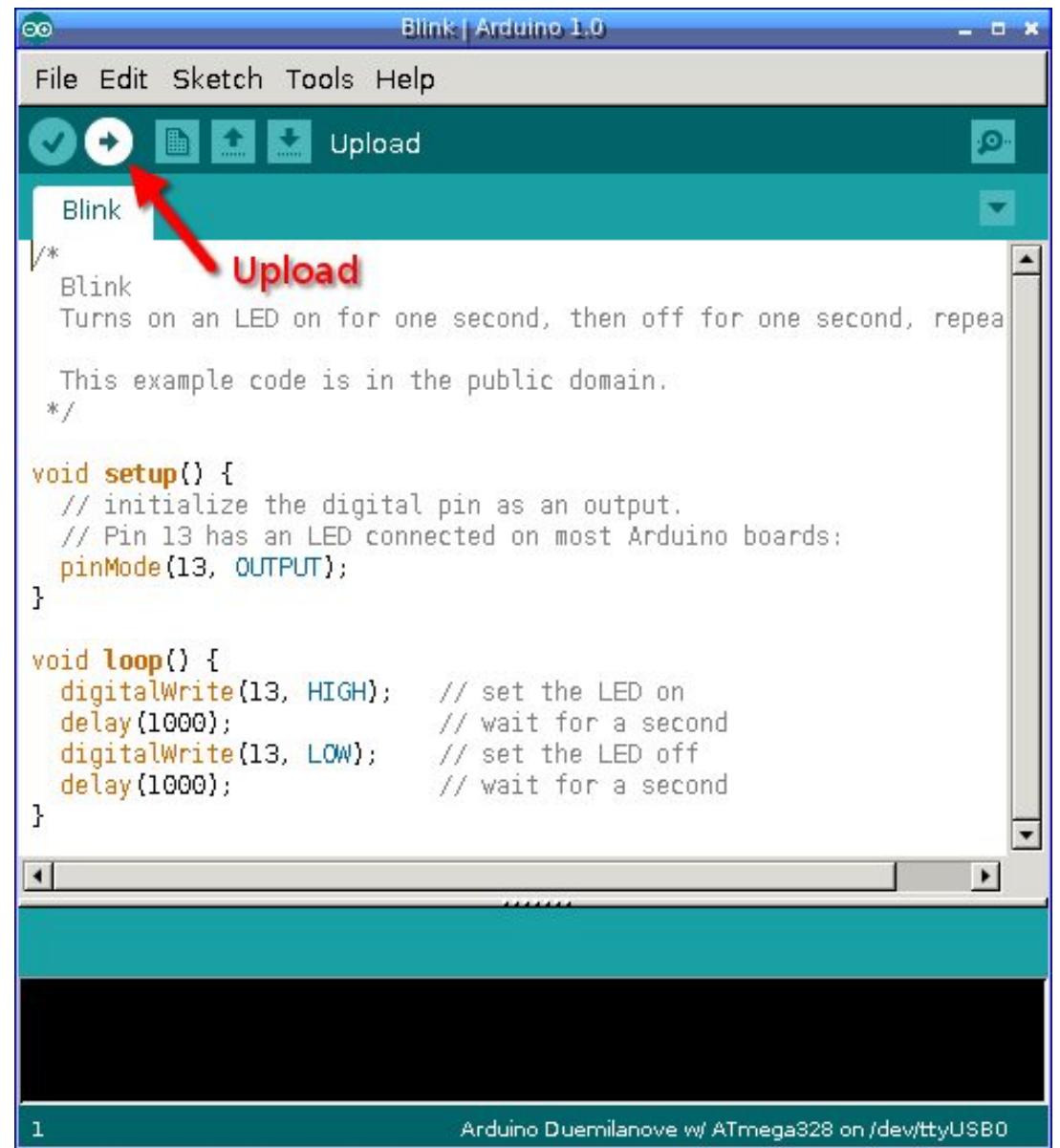
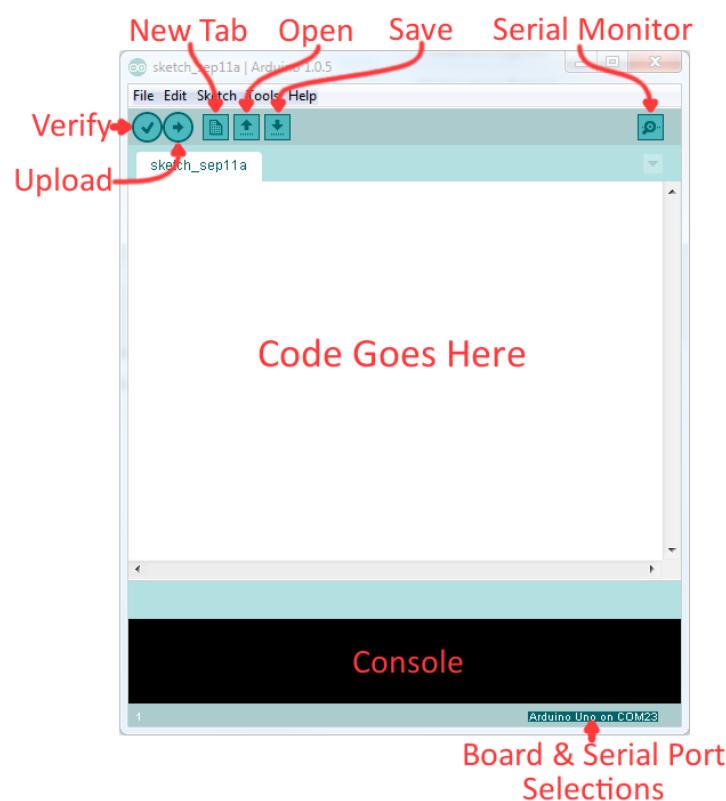
COM Port should be the highest COM# available.



Check the **Board type setting:**
Tools → Board.
Select Arduino Uno.



Arduino software: SKETCH of blink program

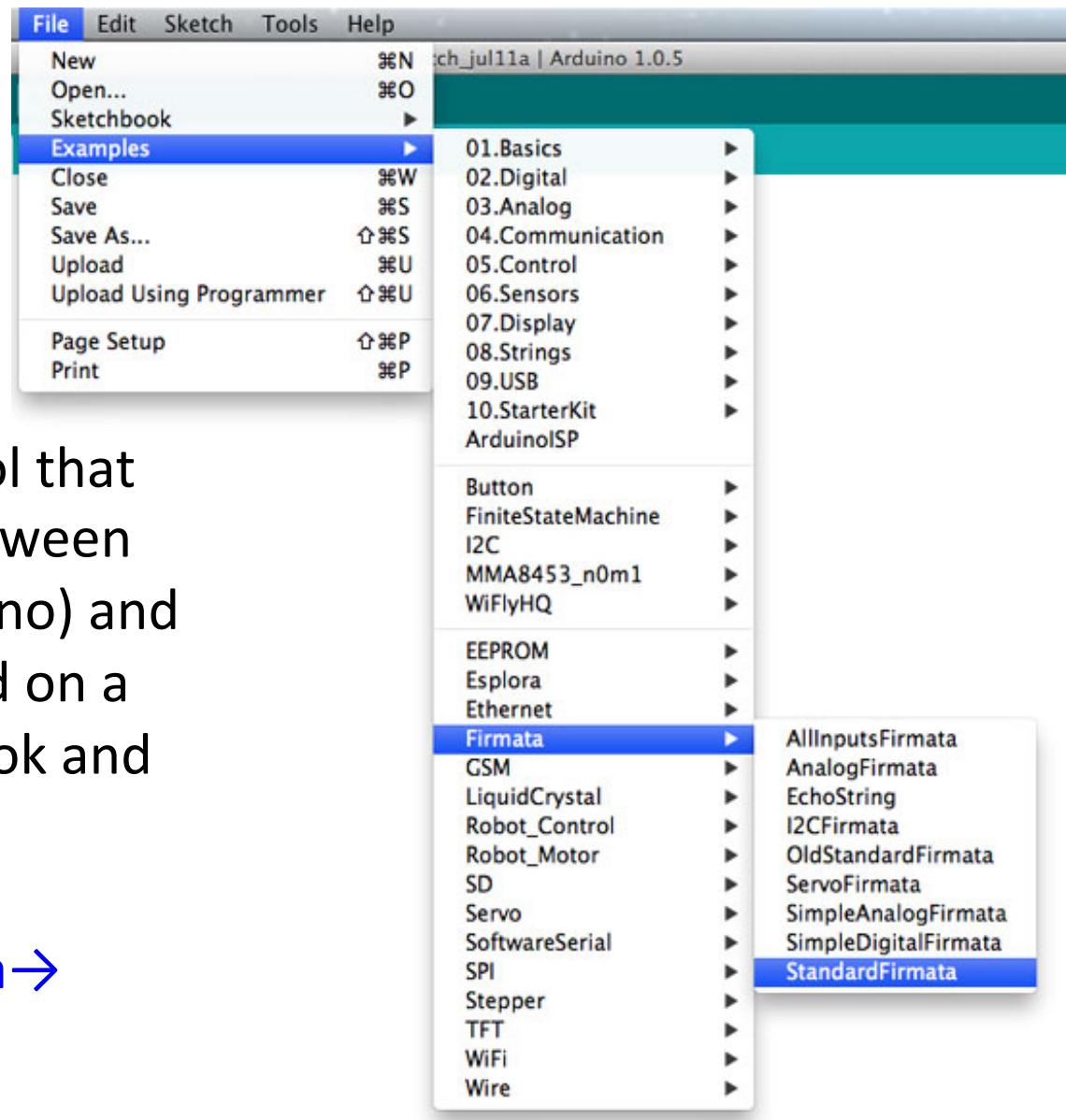


Doing the same with Python: blink program

First we need to bridge Arduino with Python.

We do this using a protocol that allows communication between the microcontroller (Arduino) and the software that is hosted on a computer (Python notebook and coding).

File → Examples → Firmata → StandardFirmata.



Doing the same with Python: blink program



The screenshot shows the Arduino IDE interface. The menu bar includes Arduino, File, Edit, Sketch, Tools, and Help. The title bar says "StandardFirmata |". The toolbar has several icons, with the "Upload" icon (a blue arrow pointing right) circled in red. Below the toolbar, the text "Upload Using Programmer" is visible. The code editor displays the "StandardFirmata" sketch:

```
1  /*
2   * Firmata is a generic protocol for communicating with microcontro
3   * from software on a host computer. It is intended to work with
4   * any host computer software package.
5   *
6   * To download a host software package, please click on the followi
7   * to open the download page in your default browser.
8   *
9   * http://firmata.org/wiki/Download
10  */
11
12 /*
13  Copyright (C) 2006-2008 Hans-Christoph Steiner. All rights reserv
14  Copyright (C) 2010-2011 Paul Stoffregen. All rights reserved.
15  Copyright (C) 2009 Shigeru Kobayashi. All rights reserved.
16  Copyright (C) 2009-2011 Jeff Hoefs. All rights reserved.
17 */
```

Look for the “Done uploading” message at the bottom.

Doing the same with Python: blink program

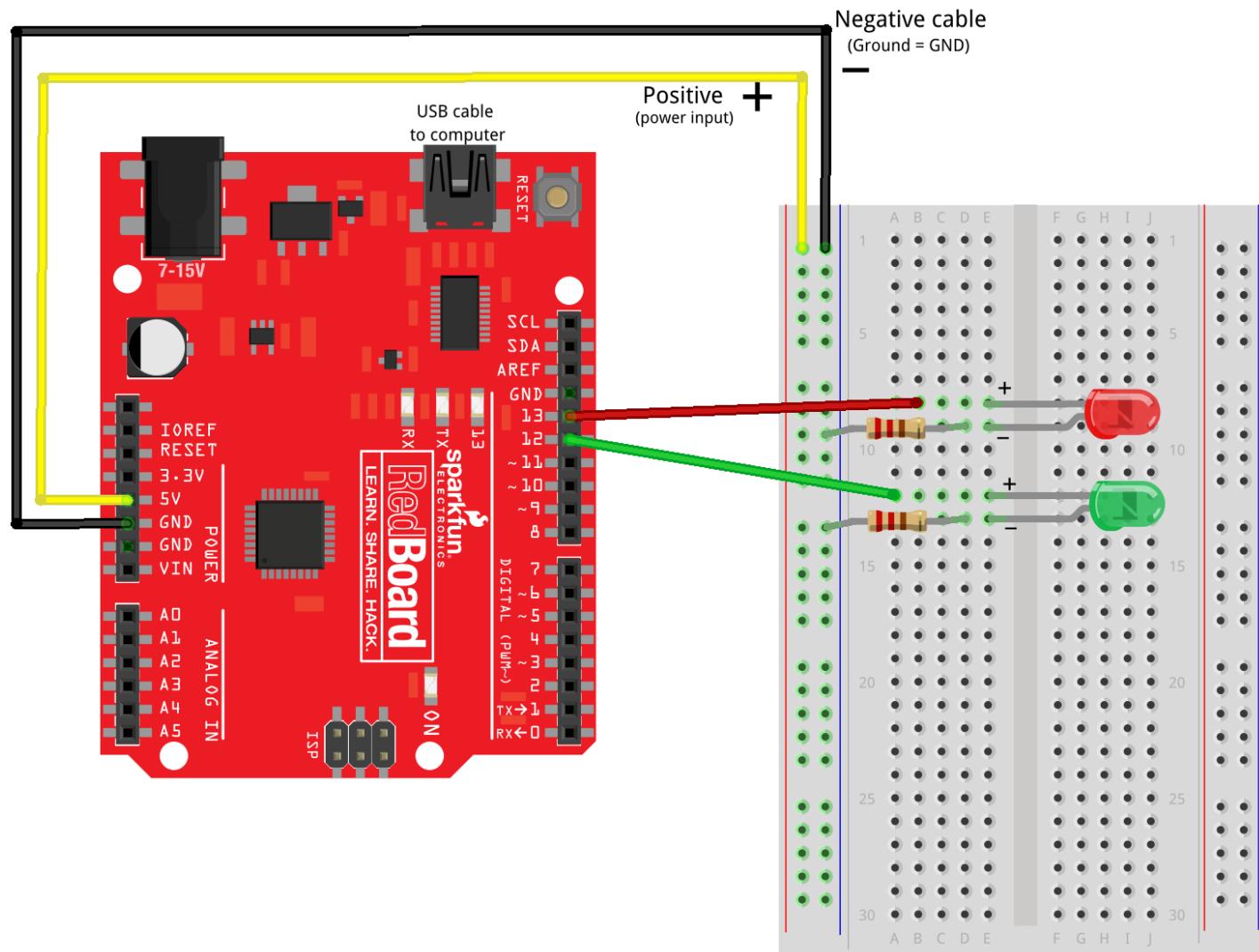
Open the program **oneLED.ipynb** available at
<https://github.com/sccc-python-workshop/python-examples>

Execute the python code (shift+enter).

Observe the LED in your circuit:

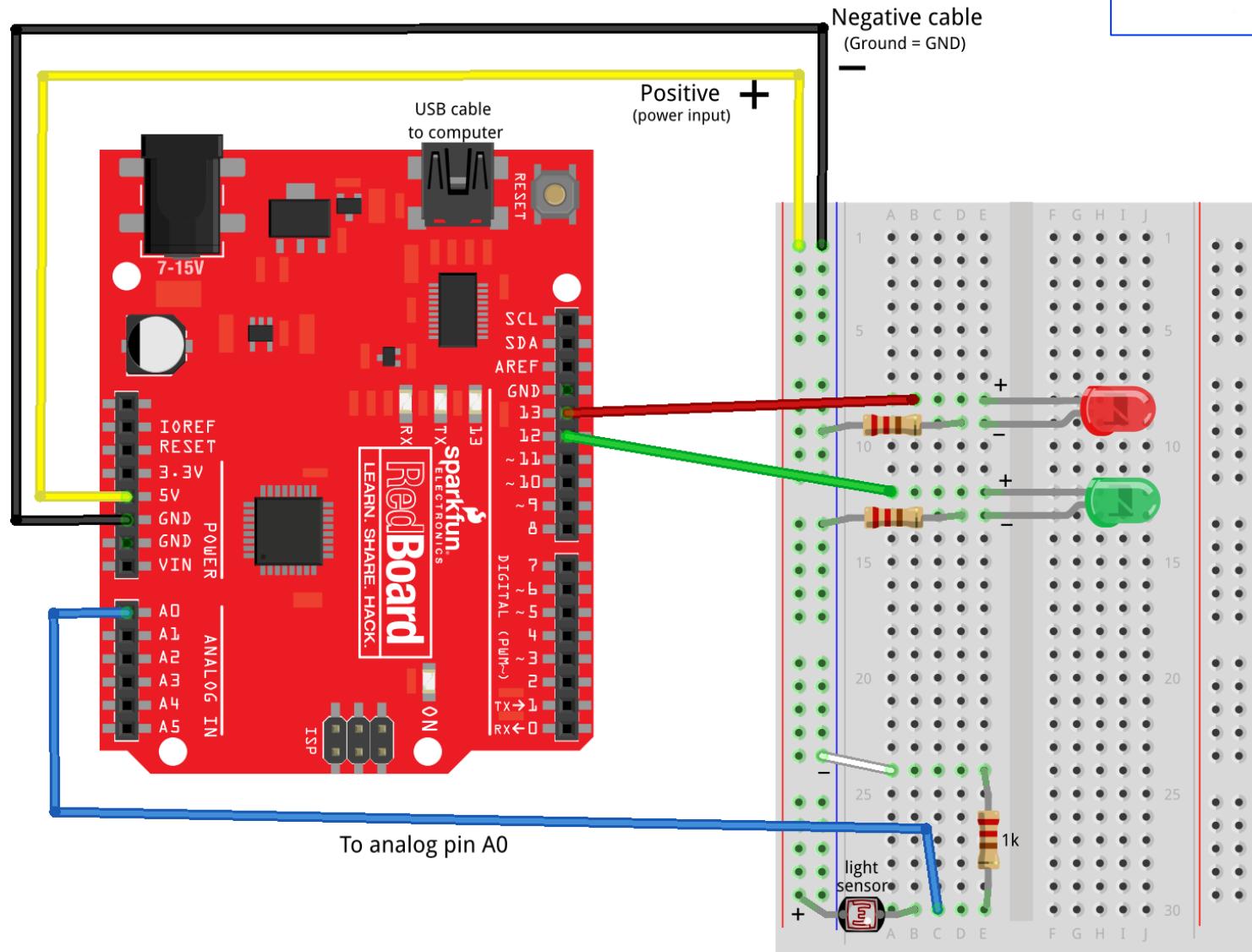
- Is it blinking every 3 seconds?
- Can you make it blink once every five seconds?

Build the circuit below with two LEDs, and run: **twoLEDs.ipynb**



fritzing

LED + light sensor circuit: Controlled blink program



fritzing

Python code for LED + light sensor: Controlled blink program

Run program **LEDsLightSensor.ipynb**

available at

<https://github.com/sccc-python-workshop/python-examples>

Execute the python code (shift+enter).

Cover the sensor with a finger:

- Is the red LED blinking?

Move your finger away from the sensor:

- Did the red LED stop blinking?
- Is the green LED on?