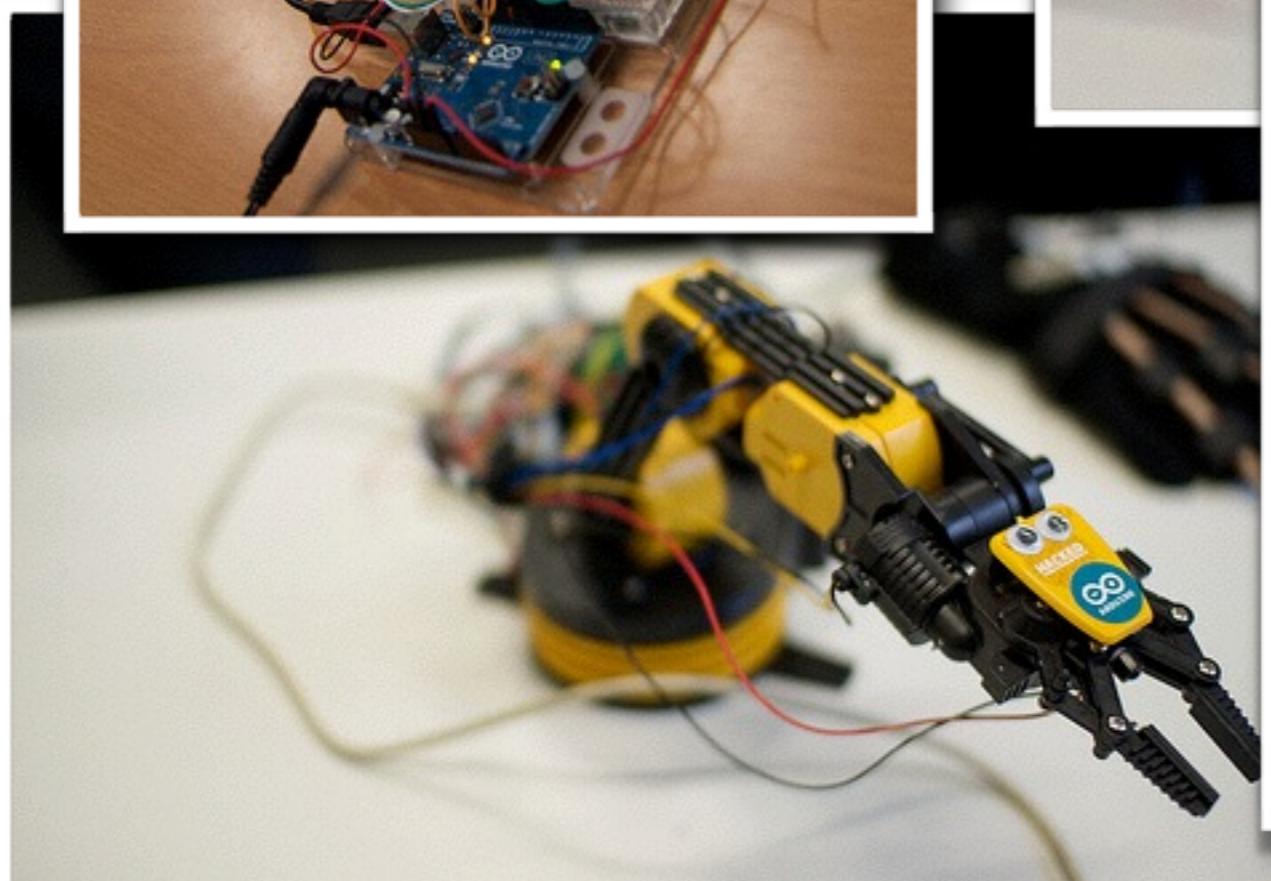
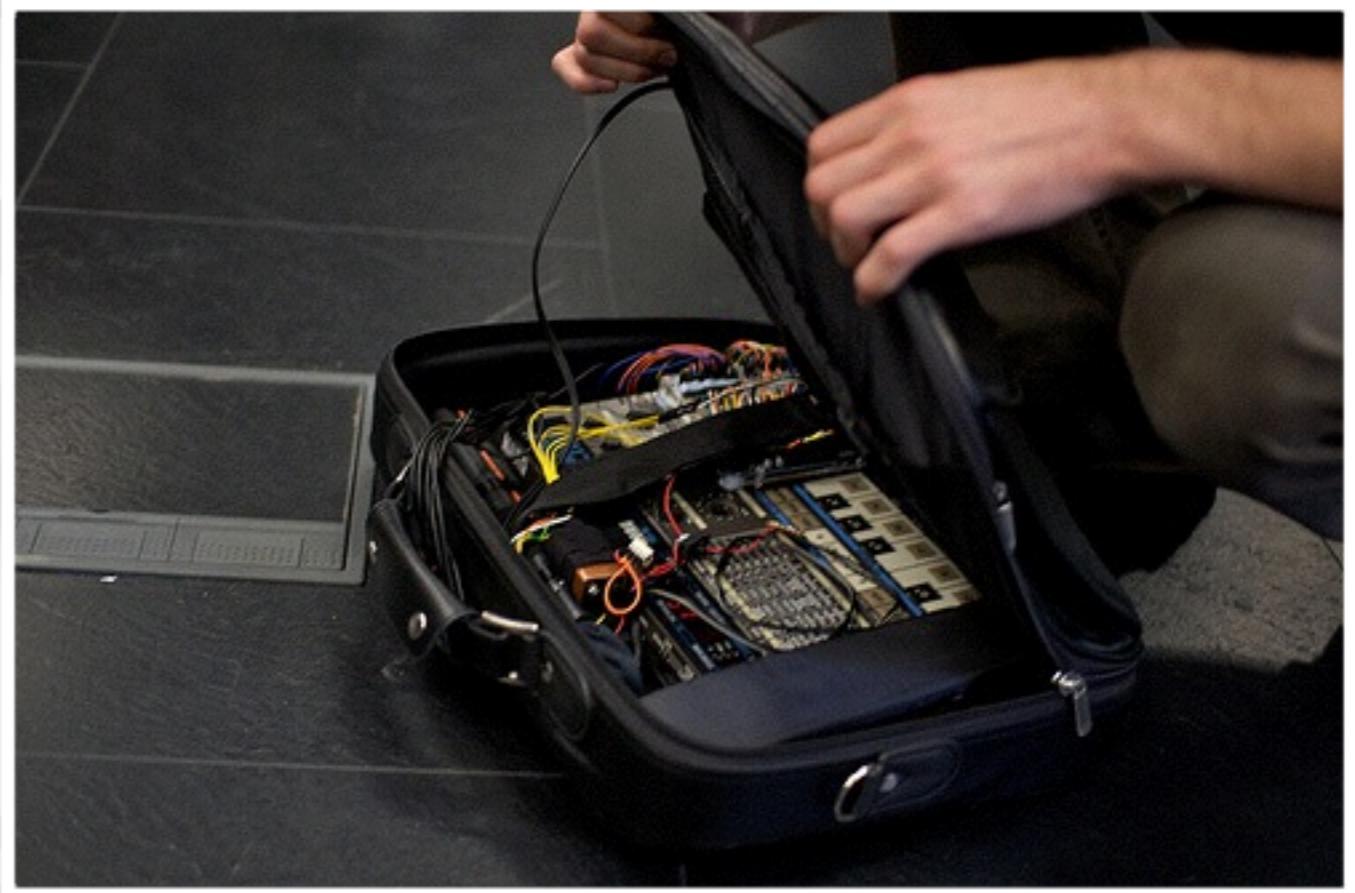
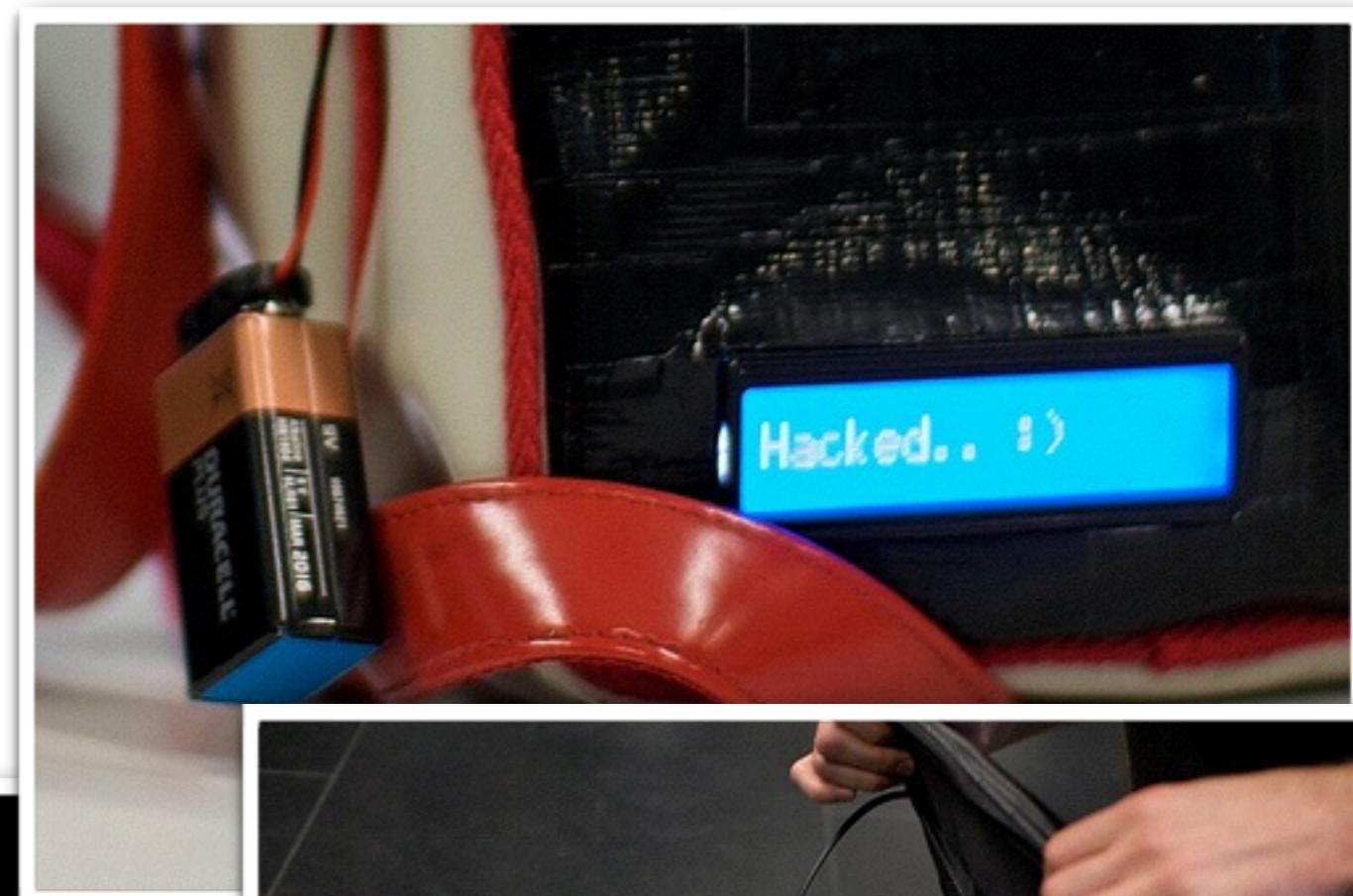
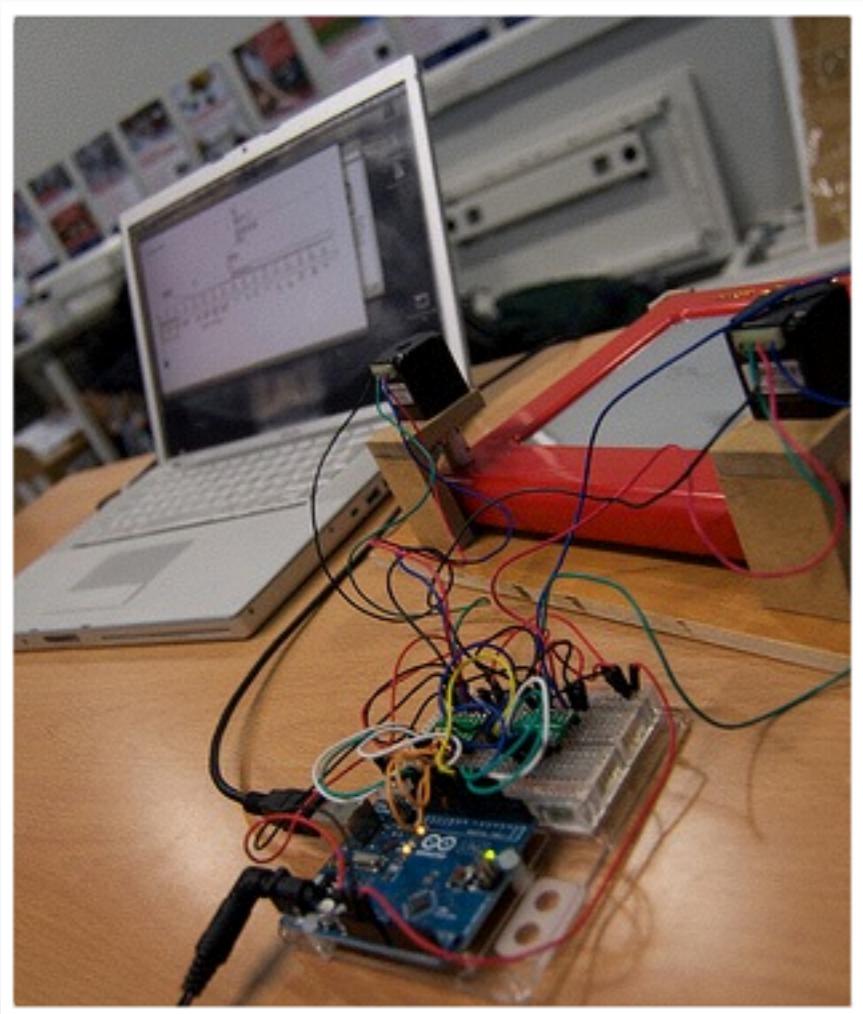


Interactive Digital Multimedia Techniques



Three things you need to know...

1. Two design projects (= 90% of module mark)

Project 1: due 7 November (week 7)

Project 2: due 12 December (week 12)

2. Class = mix of tutorials & labs

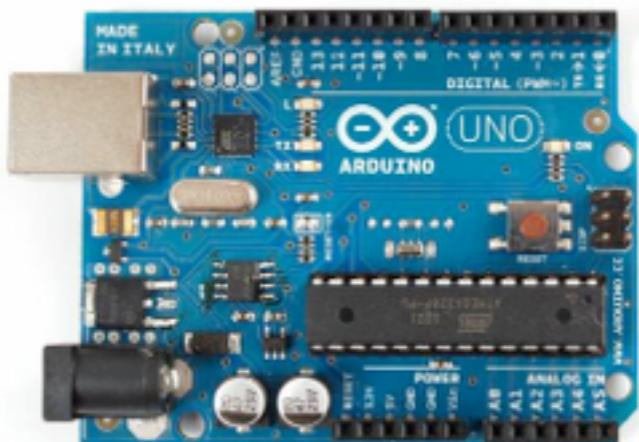
Come to class! (10-12 T, 2-4 W)

But also, the TAs and I are here to help. Get in touch.

3. Be creative and have fun!

Interactive Digital Multimedia Techniques

Human interaction /
Interfacing with the physical world



Raspberry Pi

Graphics



Processing



OpenFrameworks

Audio & Video

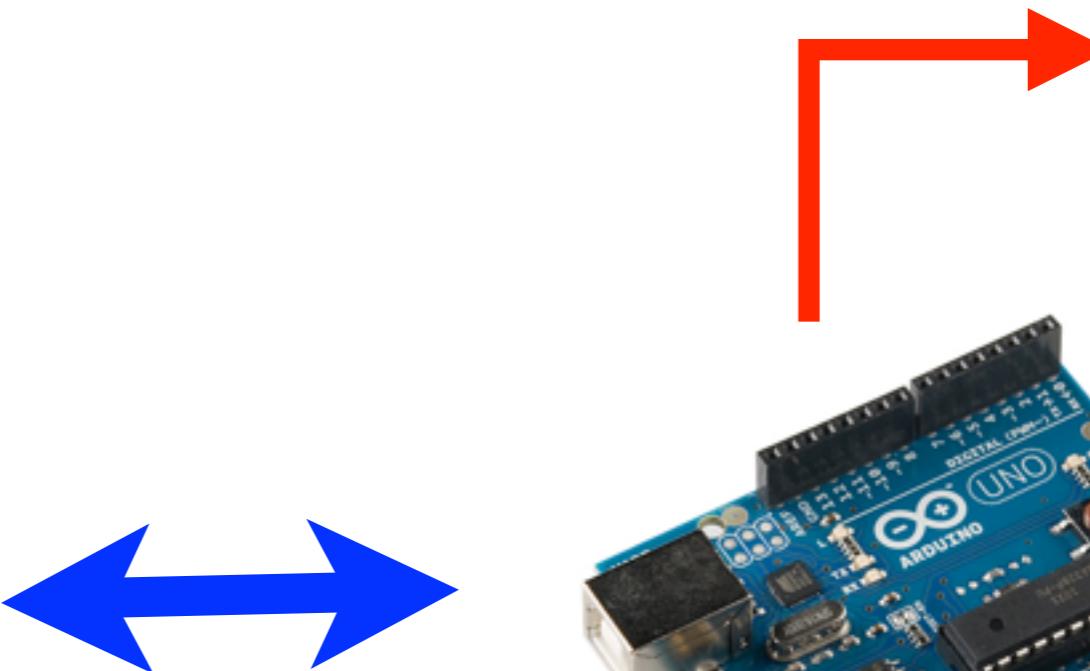


Max/MSP & Jitter



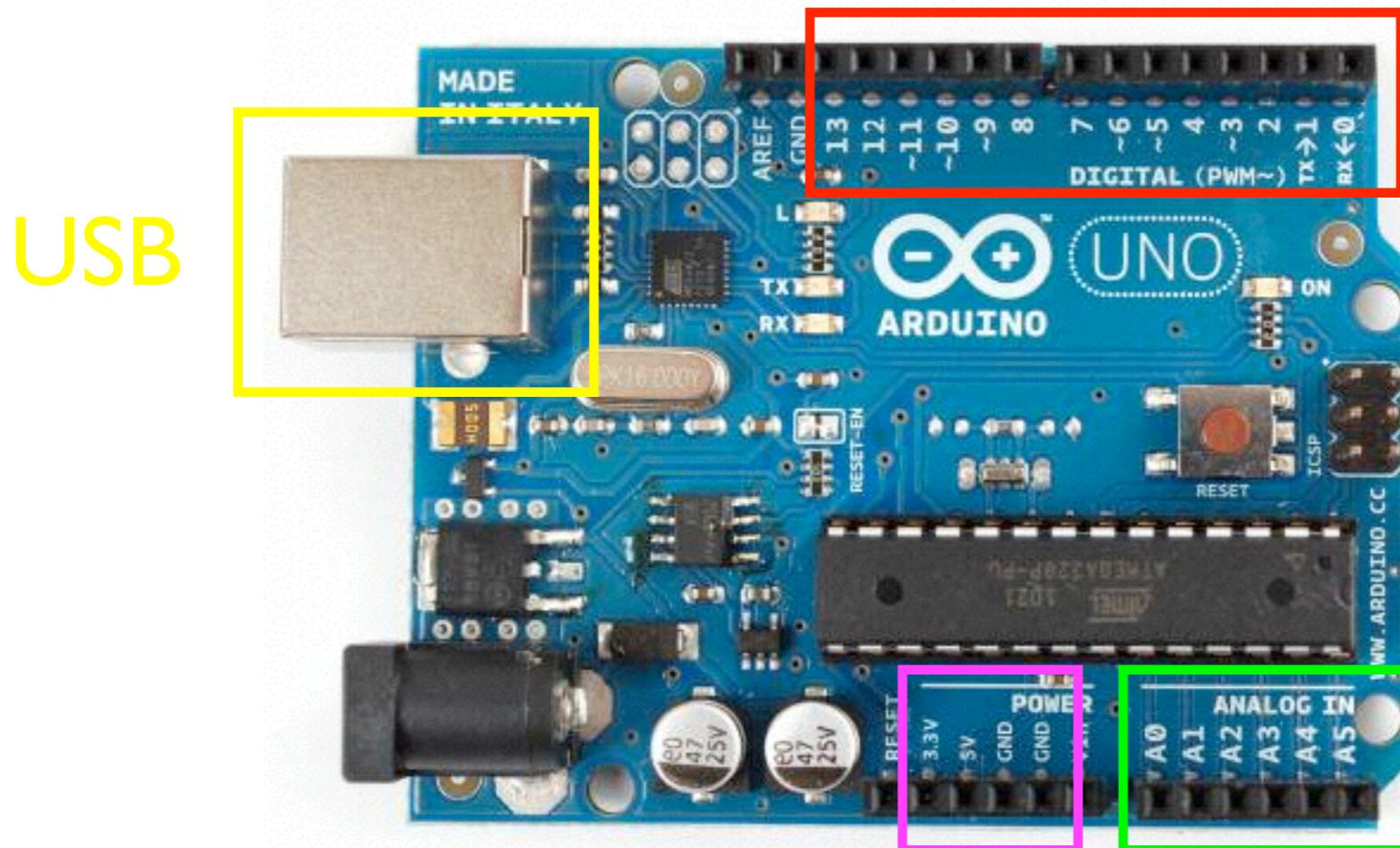
SuperCollider

Hacking the Physical World



Arduino

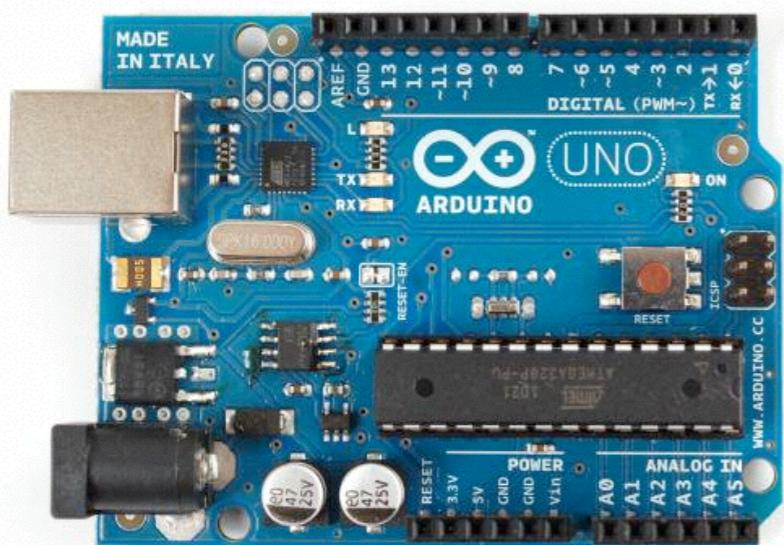
Digital inputs/outputs



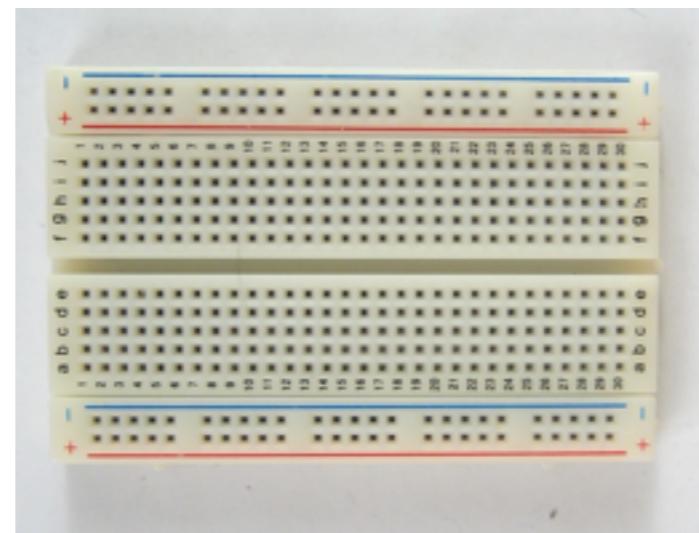
Power / Analog inputs
Ground

What's in your kit

Arduino



Breadboard



USB cable



Battery Snap



LEDs



Wires



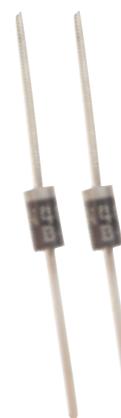
Light sensor



Motor



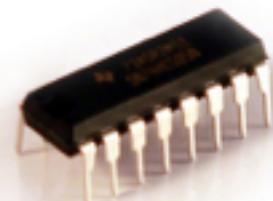
Diodes



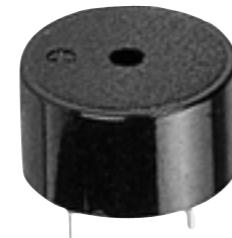
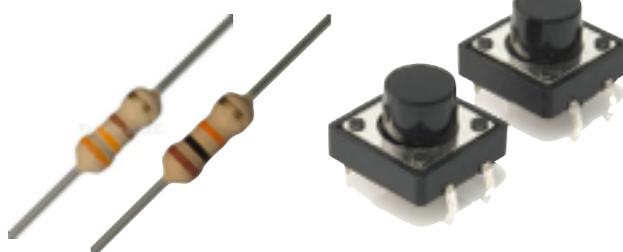
Transistor



Shift Register



Resistors



Buttons



Piezo Speaker

Potentiometer

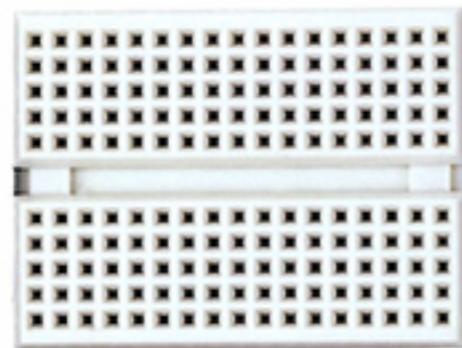
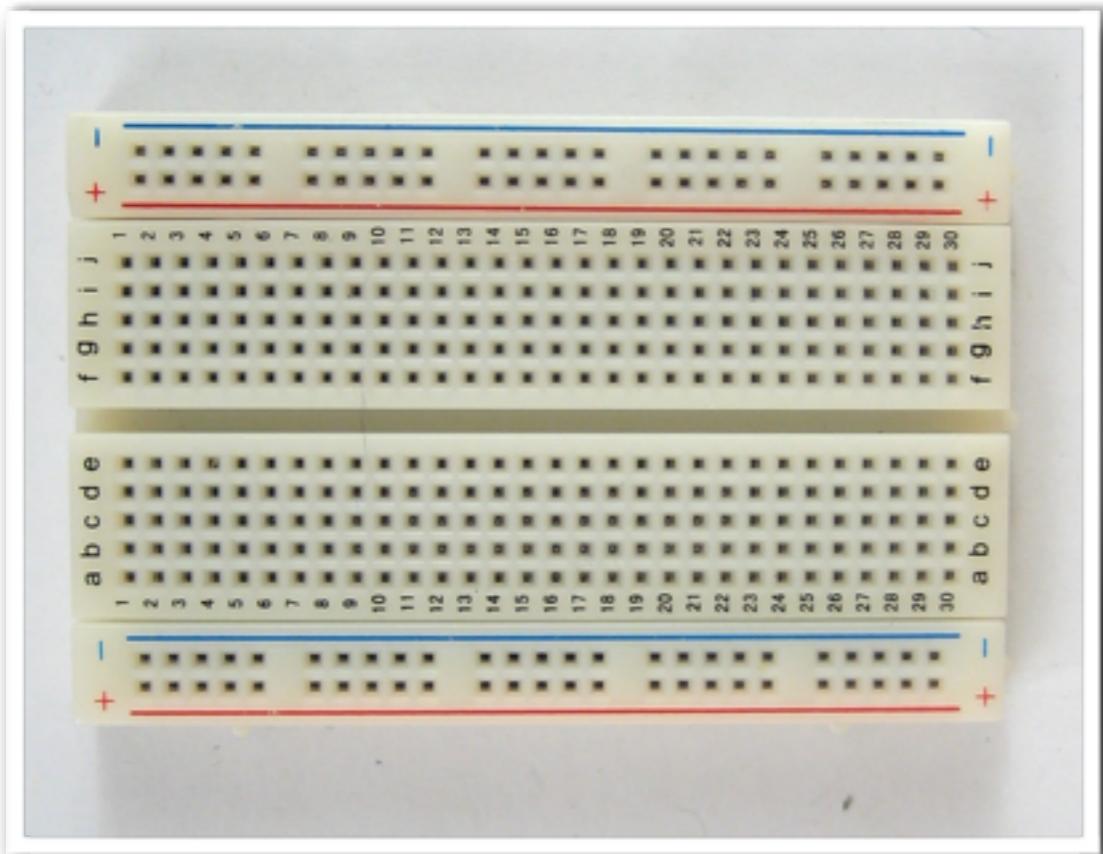


Servo Motor

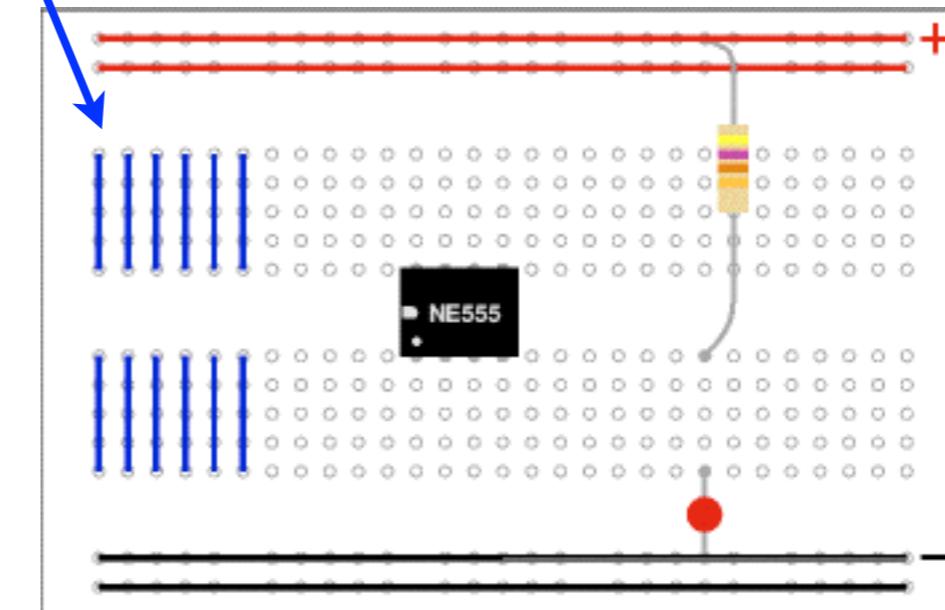


Relay

Breadboards



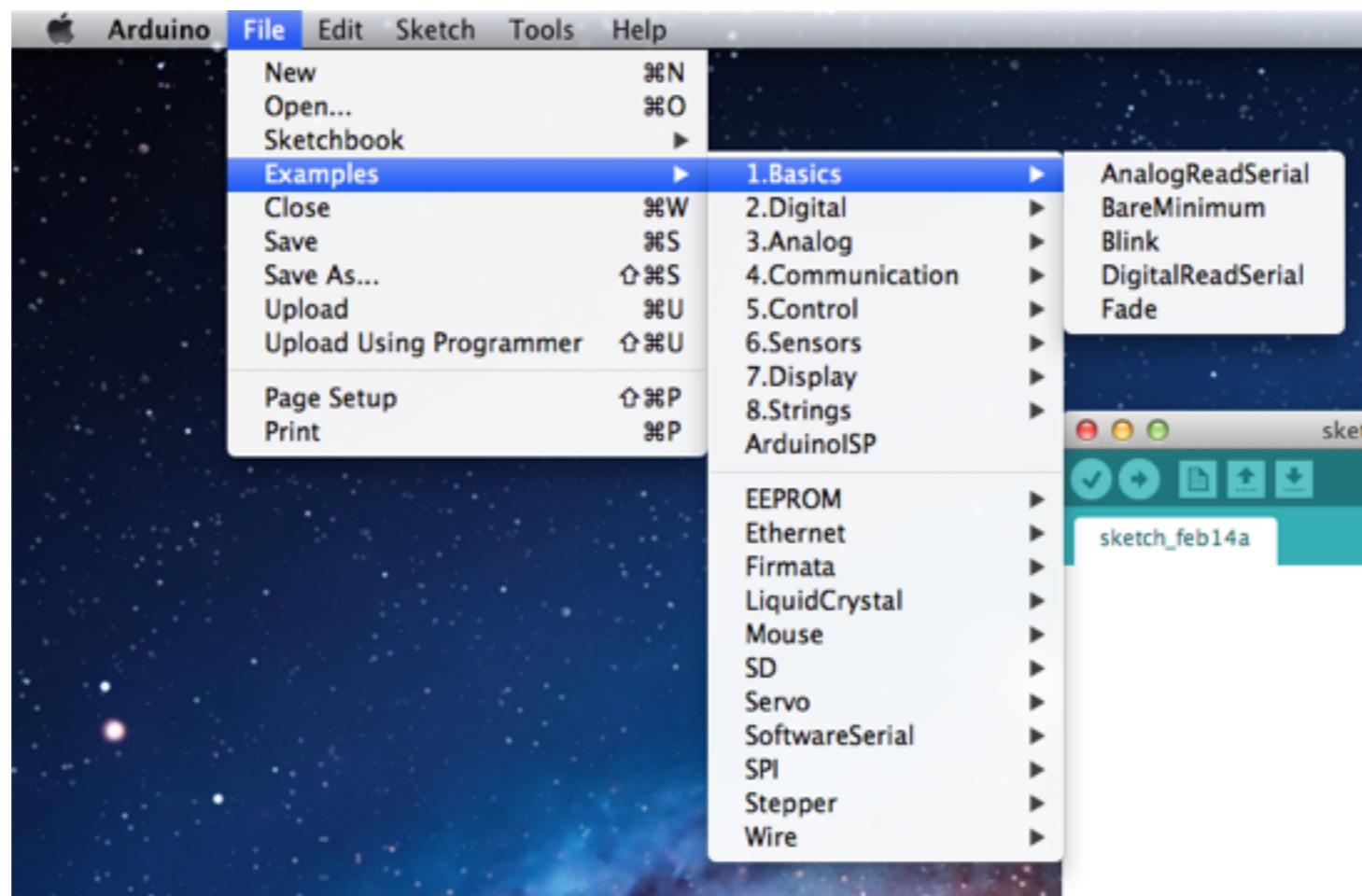
Each row of 5 is connected
(but not across the break in the middle!)



The long strips on the ends are connected
We usually use them for power and ground

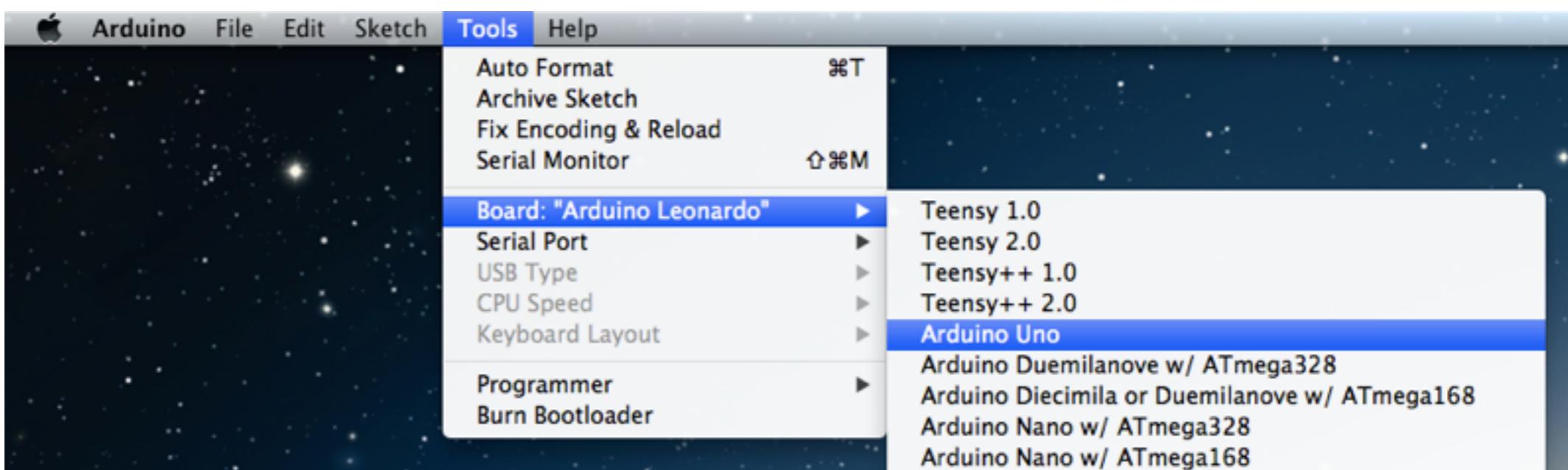
First project

1. Plug in Arduino to computer (USB)
2. Launch Arduino software
3. Open **Blink** example

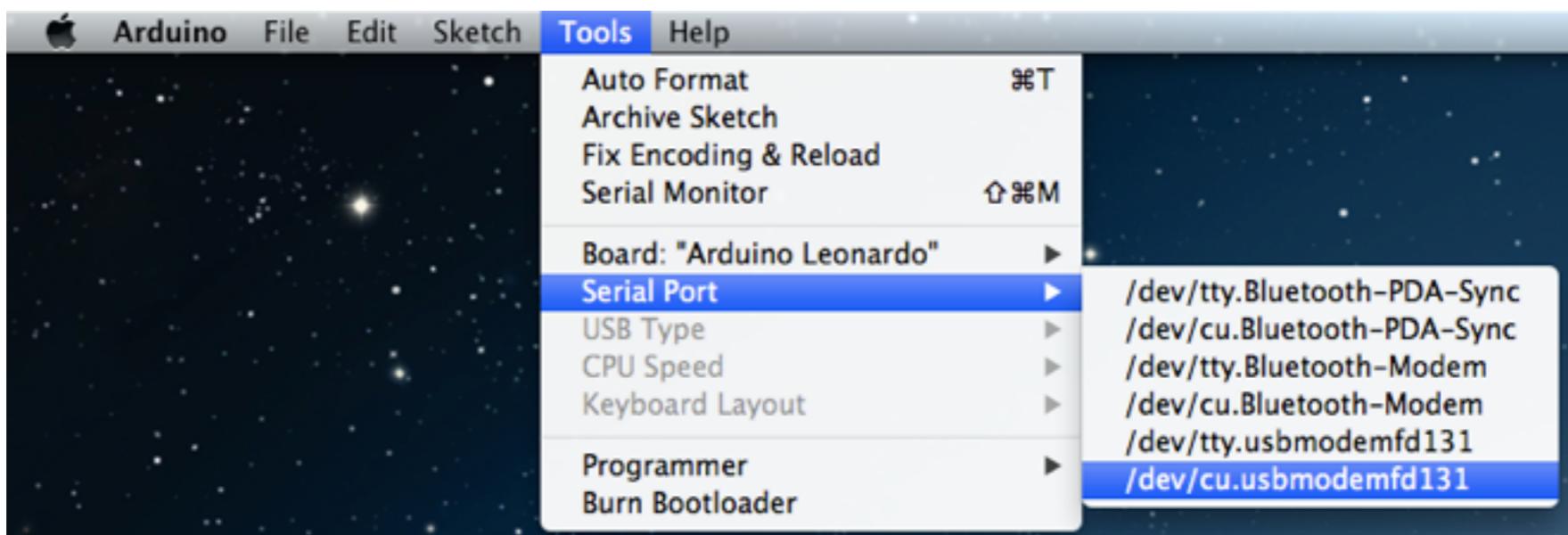


First project

4. Select **Arduino Uno** board *(not “Leonardo” unless your board says so)*

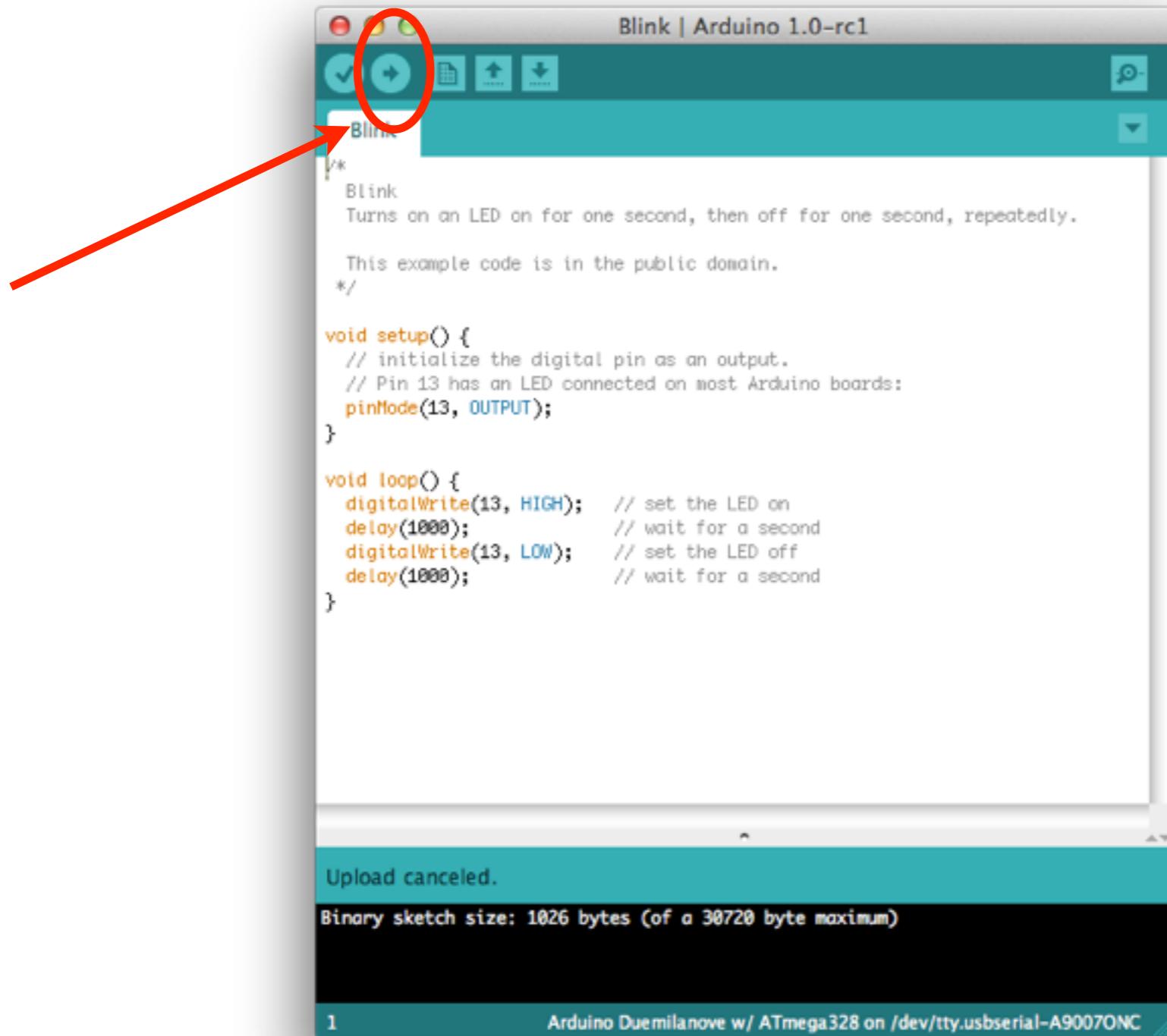


5. Select correct serial port: /dev/cu.usbmodem...



First project

6. Download sketch onto your Arduino



Blink | Arduino 1.0-rc1

Blink

Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.

```
void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);    // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);     // set the LED off
  delay(1000);              // wait for a second
}
```

Upload canceled.
Binary sketch size: 1026 bytes (of a 30720 byte maximum)

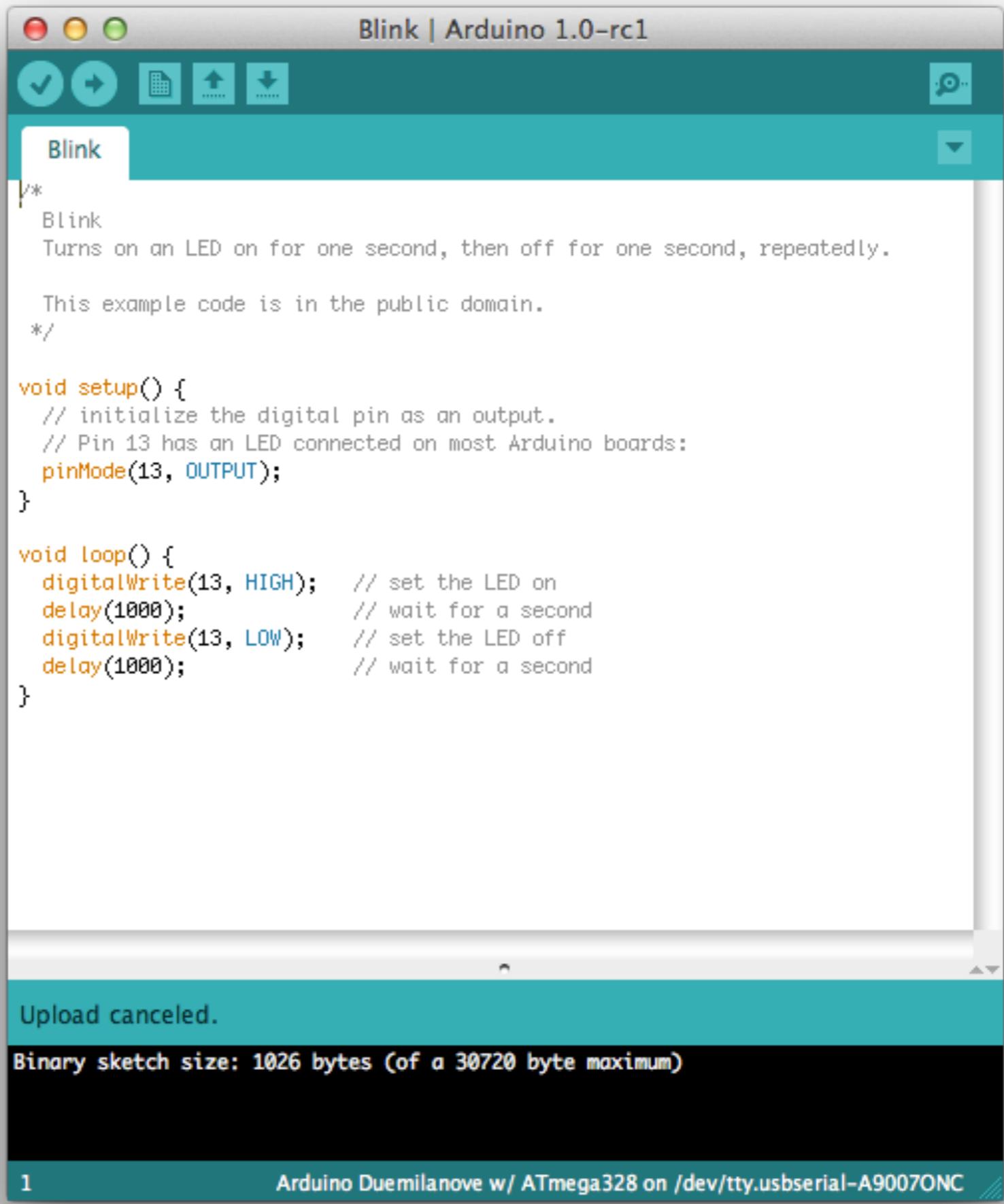
1 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbserial-A9007ONC



Watch here!

This runs **once**,
at the beginning:

This runs **in a loop**,
forever:



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1.0-rc1". The code editor contains the "Blink" sketch, which blinks an LED connected to pin 13. The code is divided into two sections: setup() and loop(). The setup() section initializes pin 13 as an output and sets it to HIGH. The loop() section alternates between setting the pin to HIGH and LOW with a one-second delay between each state change. A red vertical line highlights the setup() section, and a pink vertical line highlights the loop() section. The status bar at the bottom indicates "Upload canceled." and "Binary sketch size: 1026 bytes (of a 30720 byte maximum)".

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

void setup() {
    // initialize the digital pin as an output.
    // Pin 13 has an LED connected on most Arduino boards:
    pinMode(13, OUTPUT);
}

void loop() {
    digitalWrite(13, HIGH);      // set the LED on
    delay(1000);                // wait for a second
    digitalWrite(13, LOW);       // set the LED off
    delay(1000);                // wait for a second
}
```

Upload canceled.

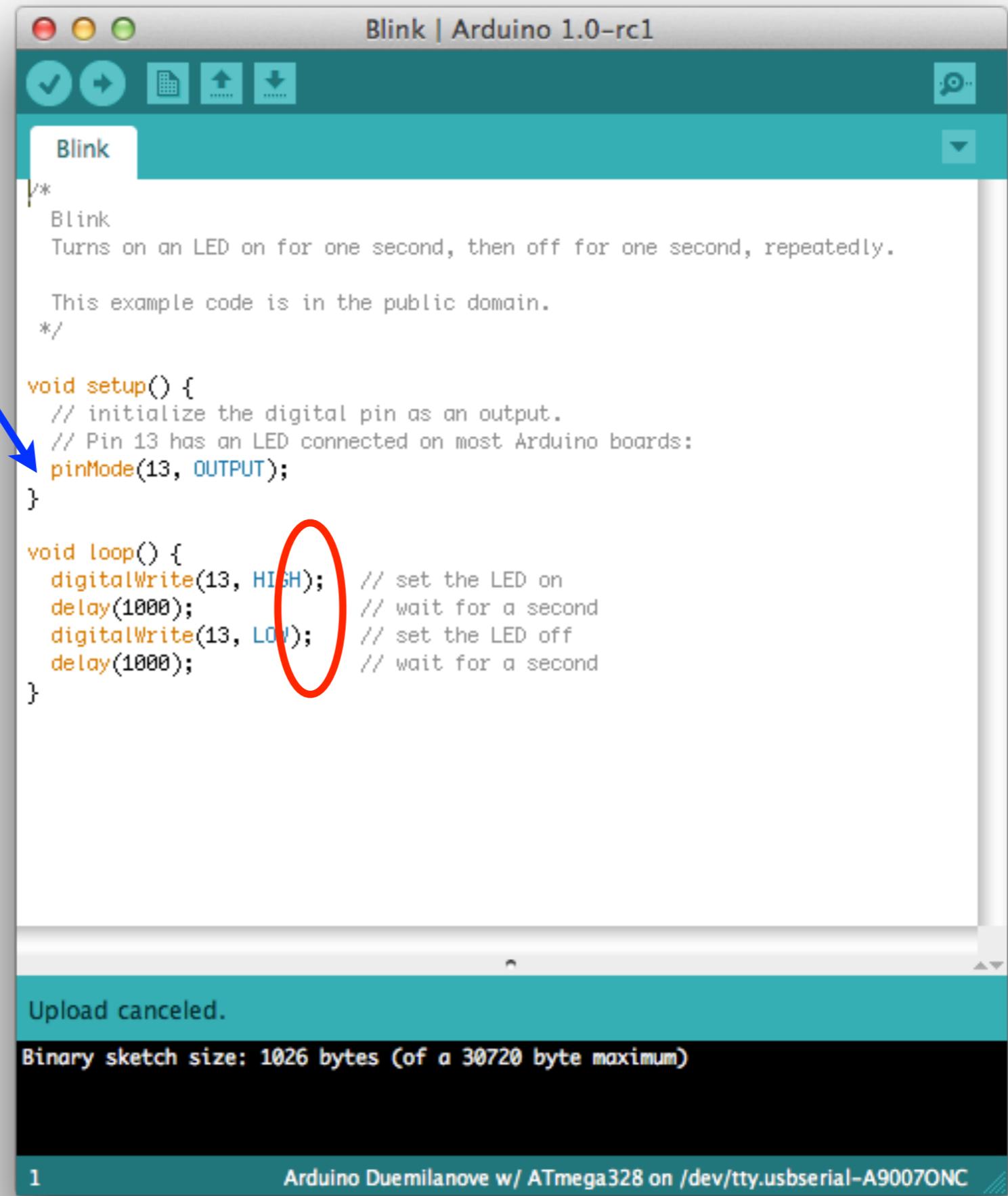
Binary sketch size: 1026 bytes (of a 30720 byte maximum)

1 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbserial-A9007ONC

This is a **function call**
What goes inside the ()
are the **arguments**

Lines starting with //
are comments

Don't forget the
semicolons!!!



```
/*  
 *  
 * Blink  
 * Turns on an LED on for one second, then off for one second, repeatedly.  
 *  
 * This example code is in the public domain.  
 */  
  
void setup() {  
    // initialize the digital pin as an output.  
    // Pin 13 has an LED connected on most Arduino boards:  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(13, HIGH); // set the LED on  
    delay(1000); // wait for a second  
    digitalWrite(13, LOW); // set the LED off  
    delay(1000); // wait for a second  
}
```

Upload canceled.
Binary sketch size: 1026 bytes (of a 30720 byte maximum)

1 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbserial-A9007ONC

pinMode() sets the pin
as an output, rather
than an input

digitalWrite() sets the
value of the pin (HIGH
= 5V, LOW = 0V)

delay(), well, delays...

How do you make the
LED blink faster?

Upload canceled.

Binary sketch size: 1026 bytes (of a 30720 byte maximum)

From code to circuit



Blink | Arduino 1.0-rc1

Blink

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);      // set the LED on
  delay(1000);                // wait for a second
  digitalWrite(13, LOW);       // set the LED off
  delay(1000);                // wait for a second
}
```

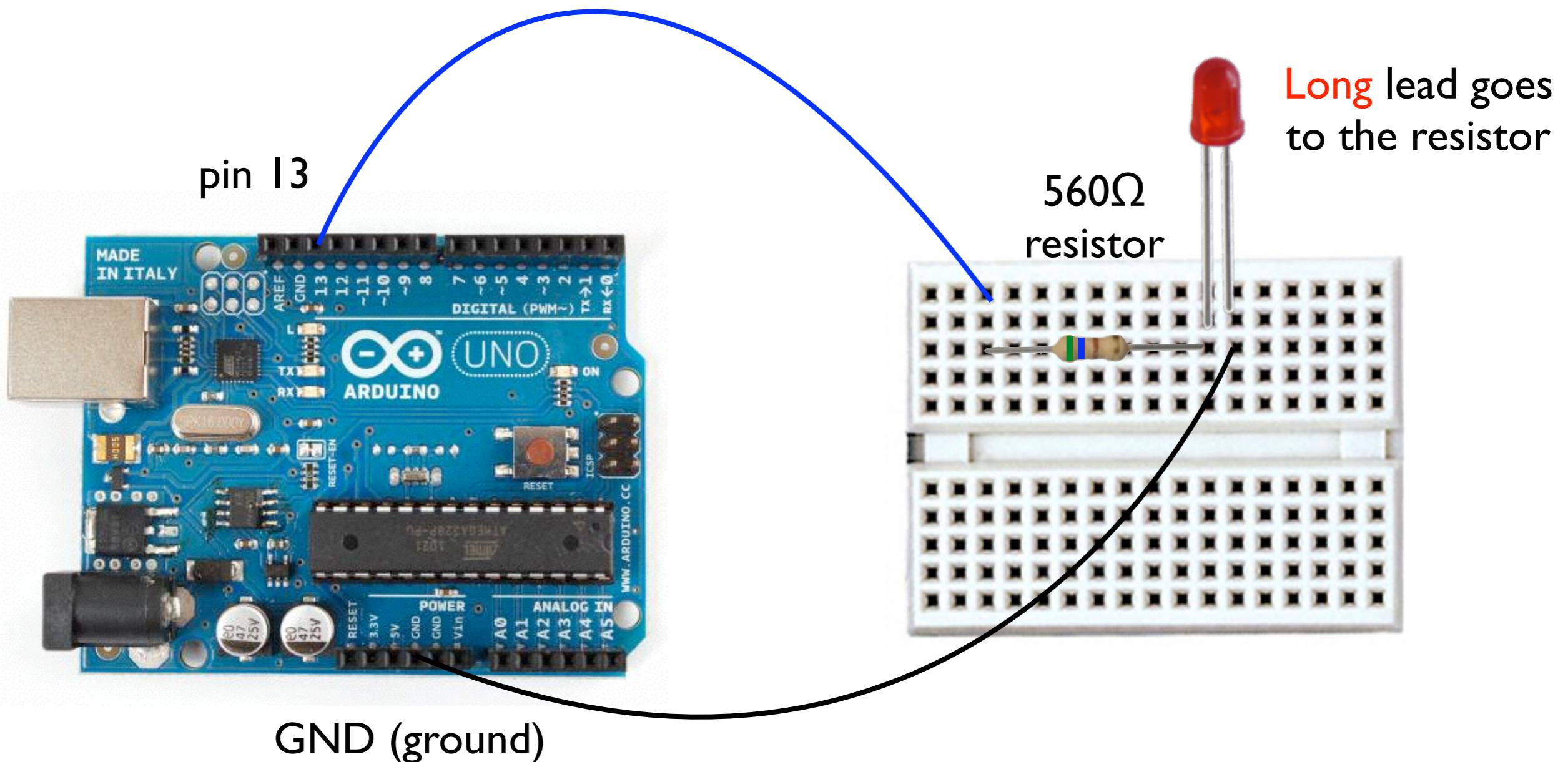
Upload canceled.

Binary sketch size: 1026 bytes (of a 30720 byte maximum)

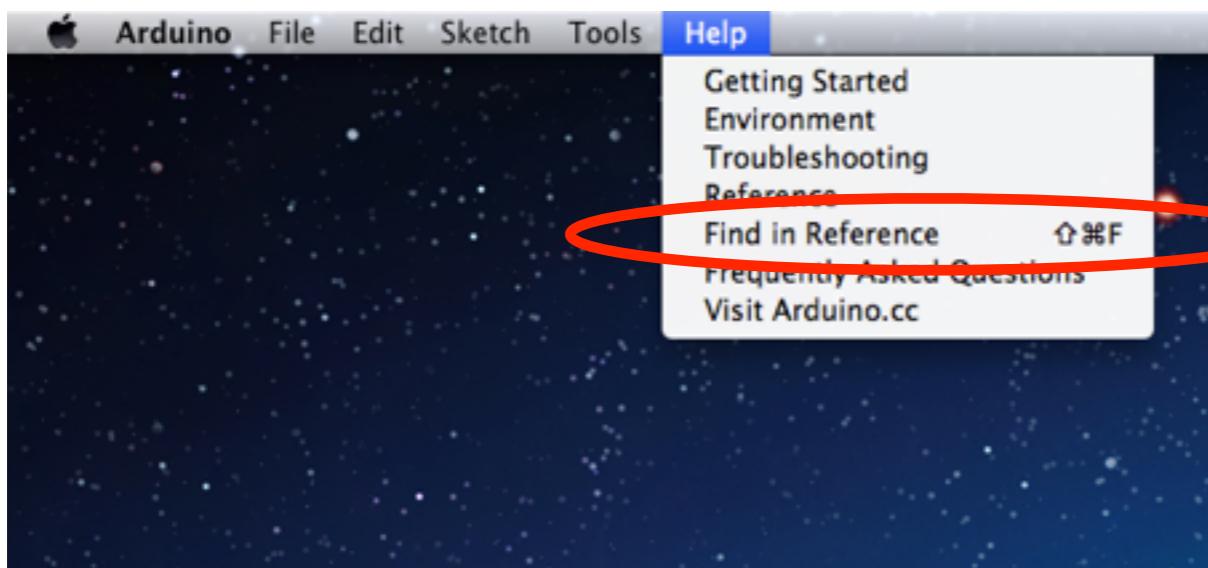
1 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbserial-A9007ONC

Connect an LED*

* Light-Emitting Diode



Where to get help



The screenshot shows the Arduino IDE interface. The top menu bar includes Arduino, File, Edit, Sketch, Tools, and Help. The Help menu is open, displaying options: Getting Started, Environment, Troubleshooting, Reference, Find in Reference (which is highlighted with a red oval), Frequently Asked Questions, and Visit Arduino.cc.

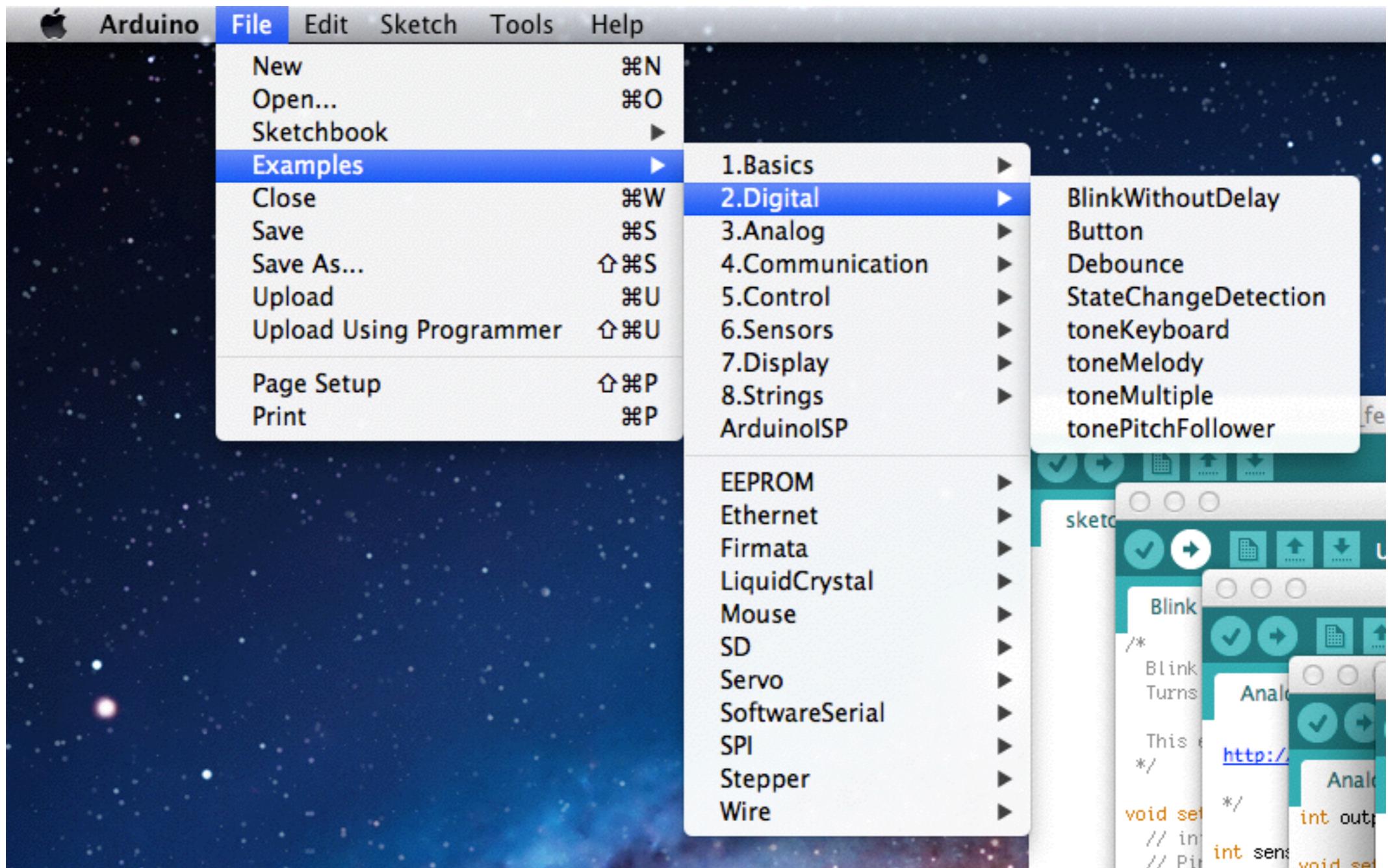
The main window title is "Arduino - Reference" and the URL is "Arduino.app/Contents/Resources/Java/reference/index.html". Below the title, there are links for Library, Jobs, Amazon, Penn music, eBay, WebCT, and Wikipedia. A search bar with a "search" button is also present.

The page content starts with a navigation bar: Reference | Language | Libraries | Comparison | Changes. The main section is titled "Language Reference". It states: "Arduino programs can be divided in three main parts: *structure*, *values* (variables and constants), and *functions*".

The content is organized into three columns:

- Structure**
 - * setup()
 - * loop()
- Variables**
 - Constants**
 - * HIGH | LOW
 - * INPUT | OUTPUT
 - * true | false
 - * integer constants
 - * floating point constants
 - Data Types**
 - * void
 - * boolean
 - * char
- Functions**
 - Digital I/O
 - * pinMode()
 - * digitalWrite()
 - * digitalRead()
 - Analog I/O
 - * analogReference()
 - * analogRead()
 - * analogWrite() - PWM
 - Advanced I/O
 - * tone()

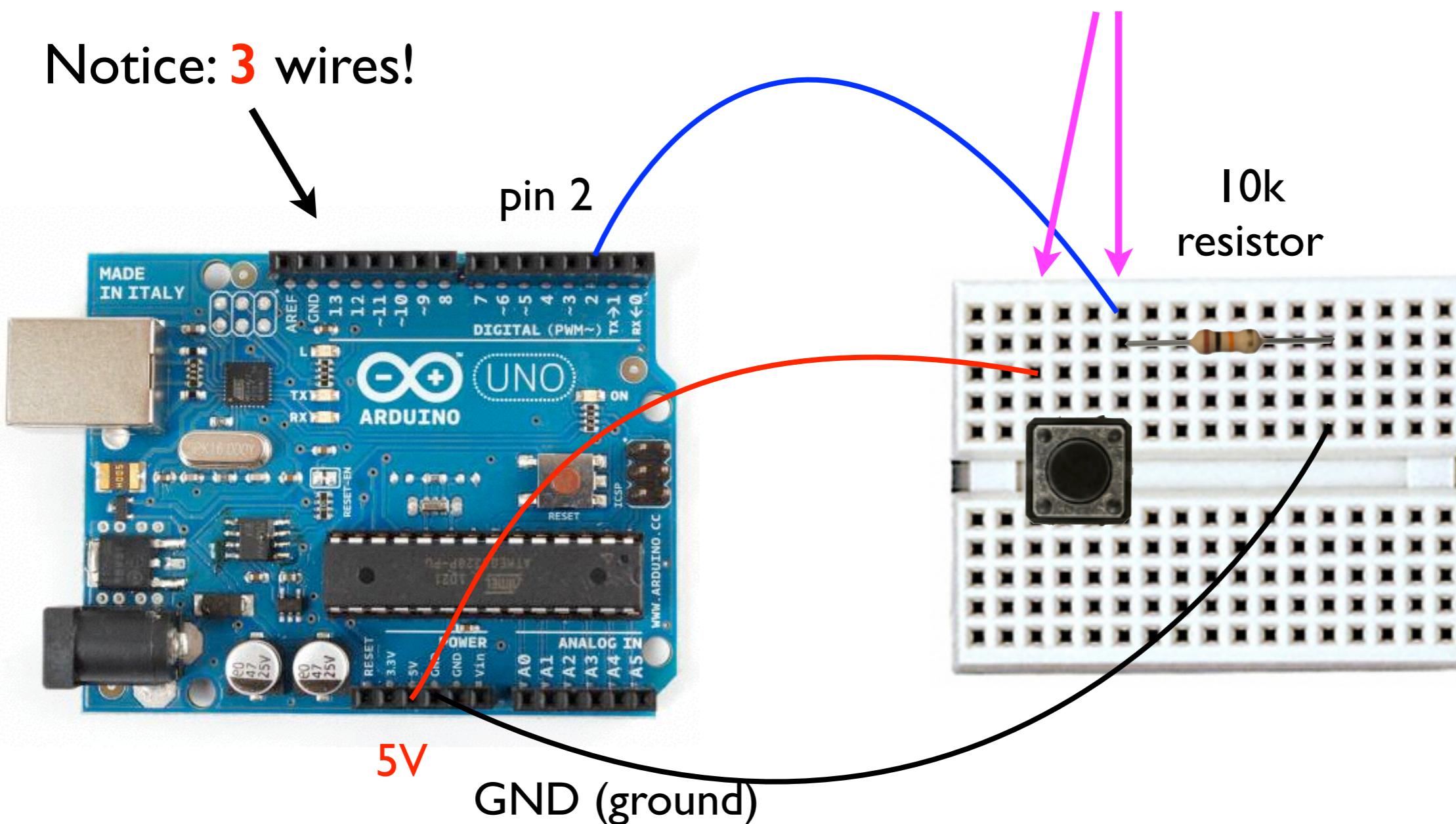
Using sensors



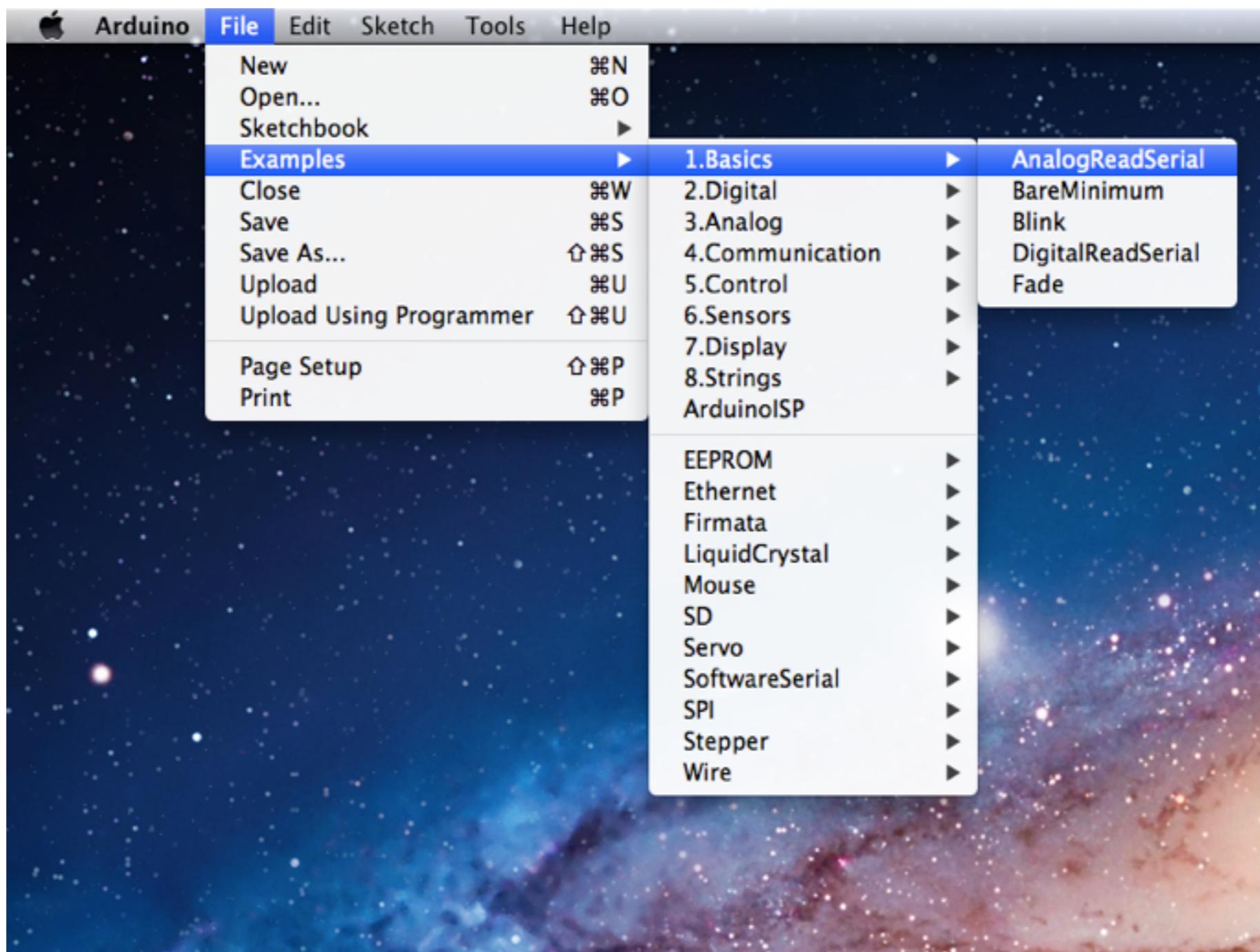
Open **Button** sketch

Connect a Button

Notice: **3** wires!



Using analog sensors

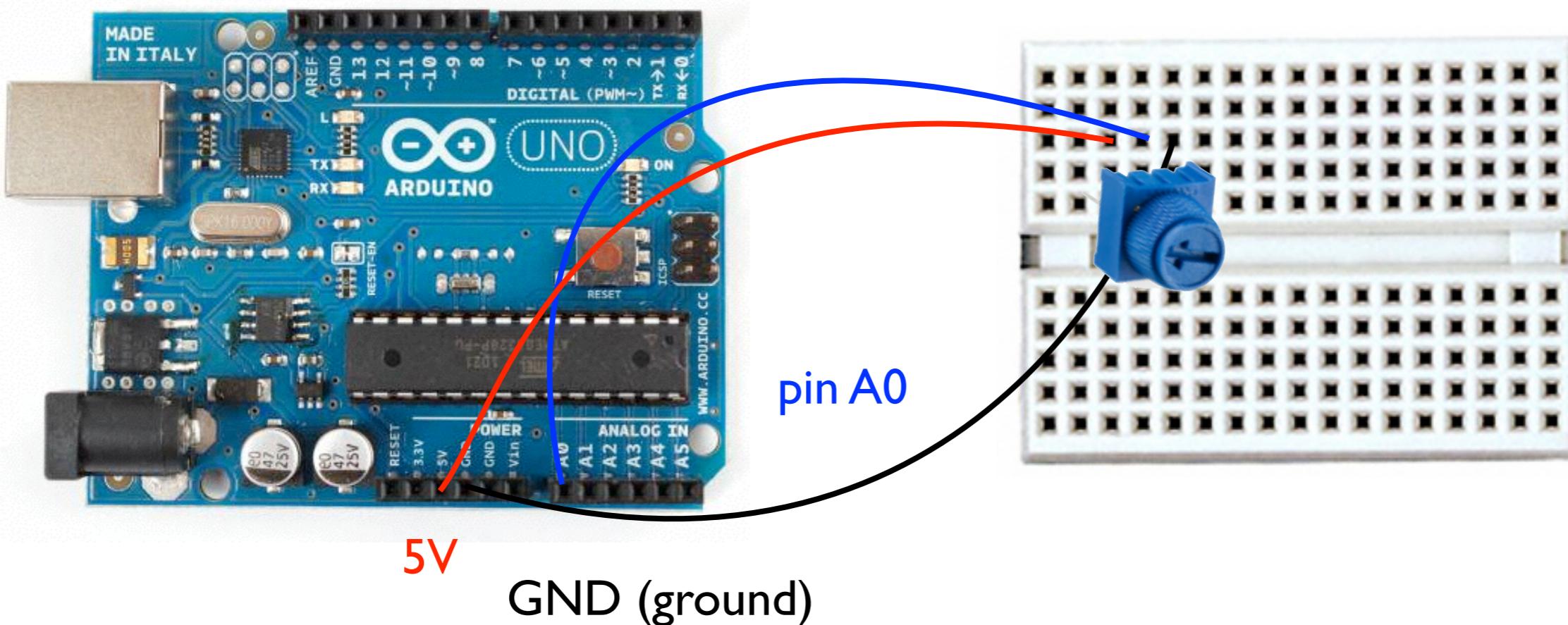


Open [AnalogReadSerial](#) sketch

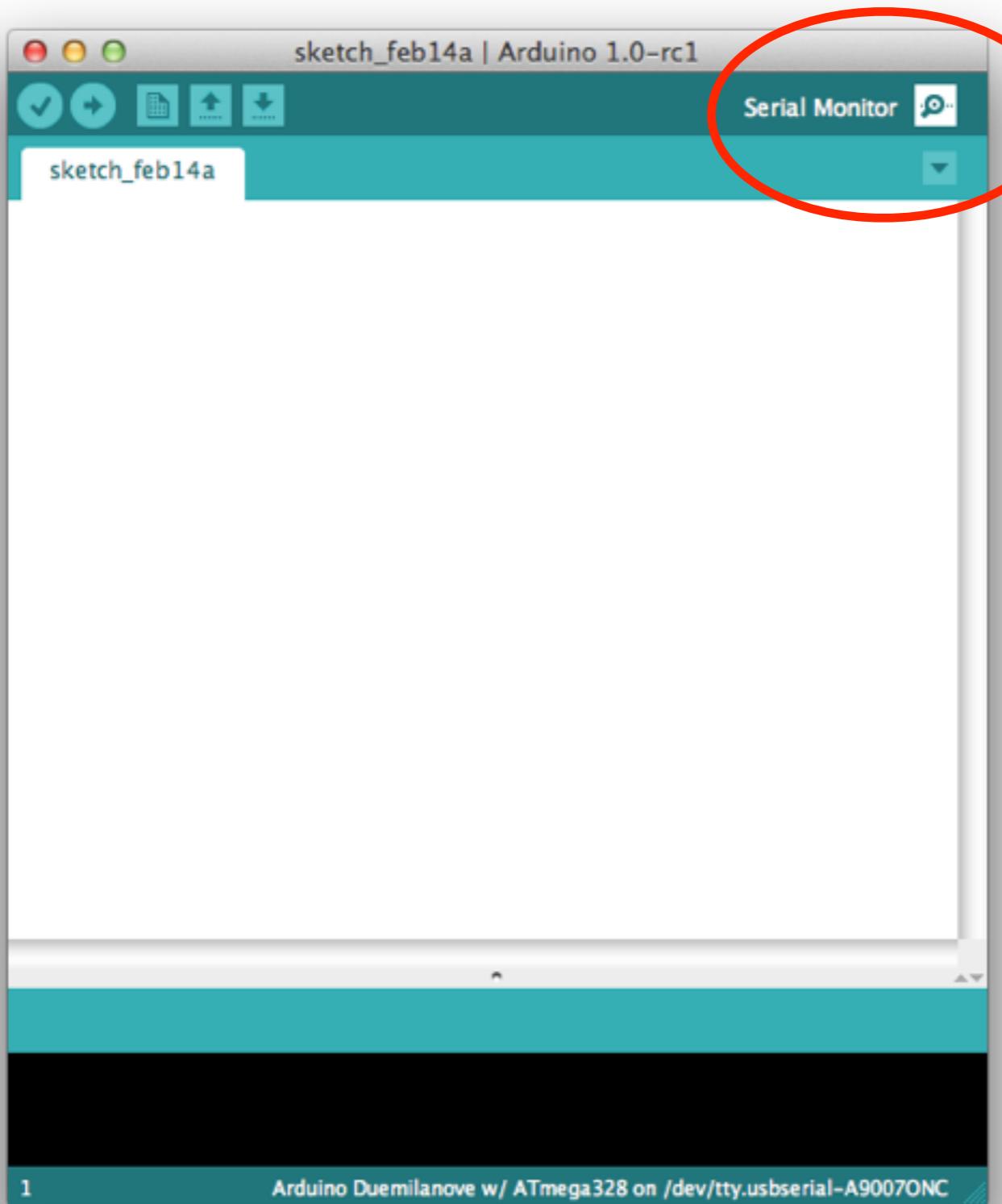
Connect a Potentiometer

a.k.a. a “pot” or knob

The pot has 3 pins
5V and GND on the outside
Arduino **A0** in the middle

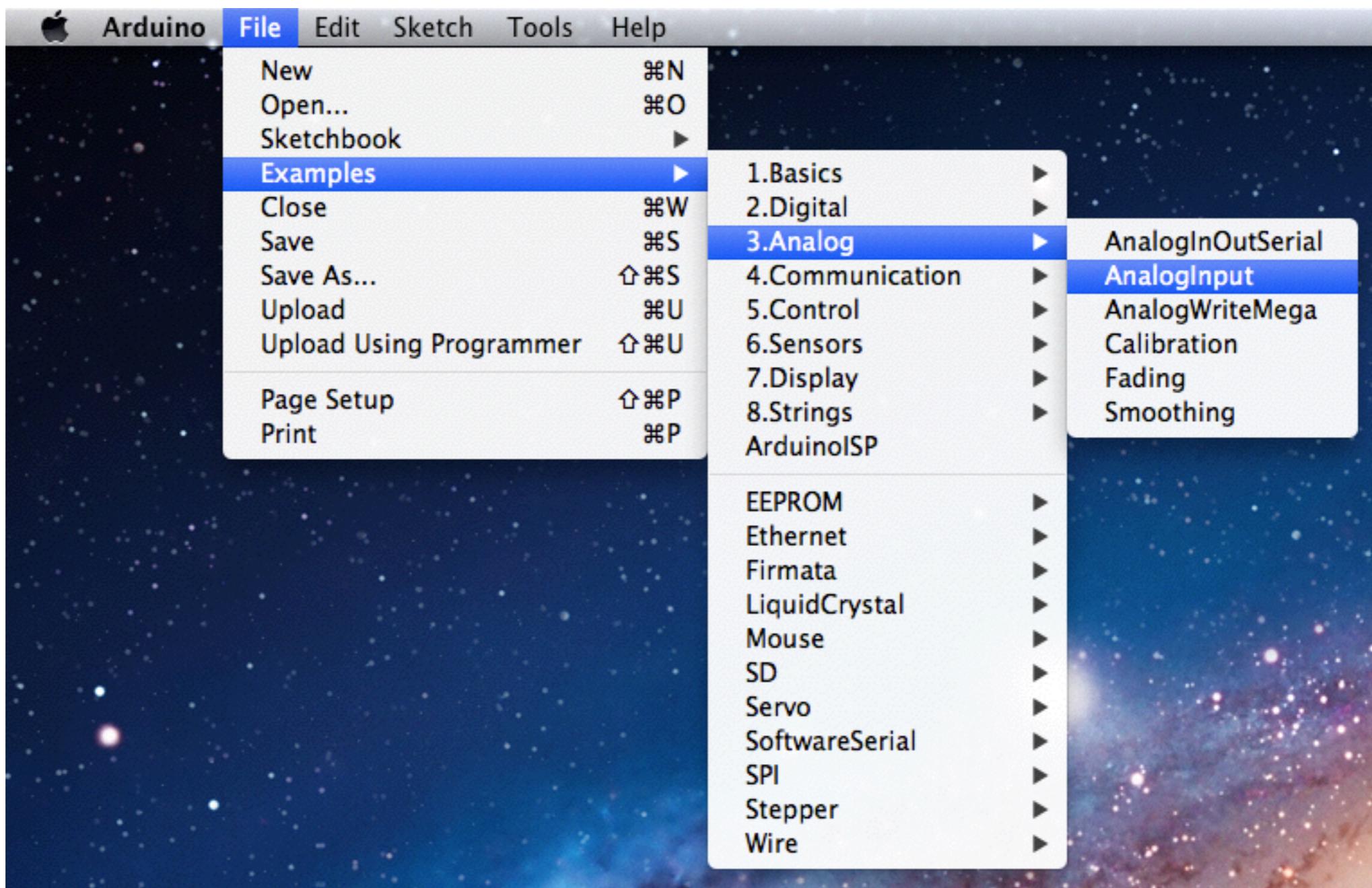


Reading values



Serial monitor
(after you've uploaded
your sketch!)

Sensors and LEDs

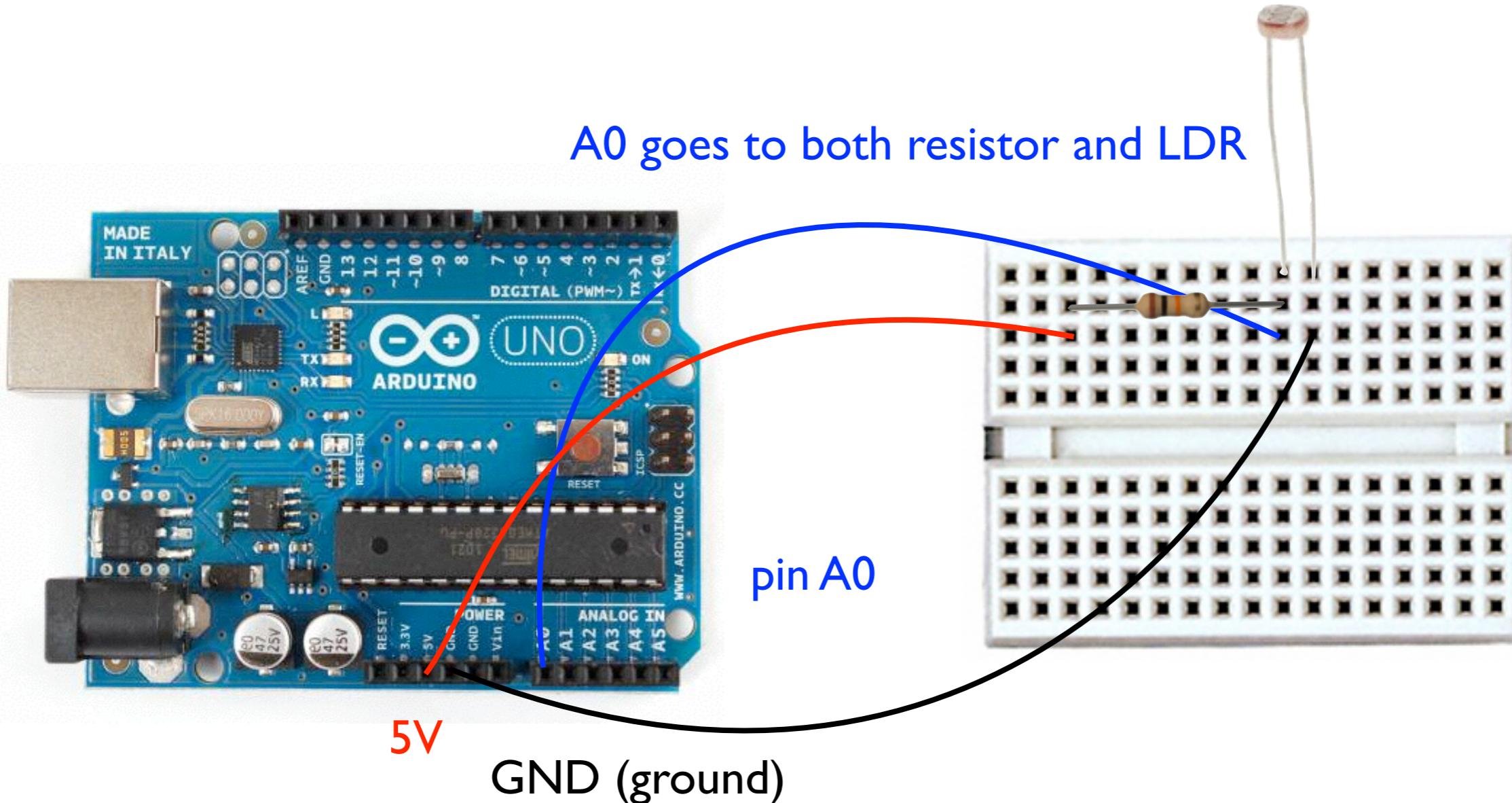


Open [AnalogInput](#) sketch

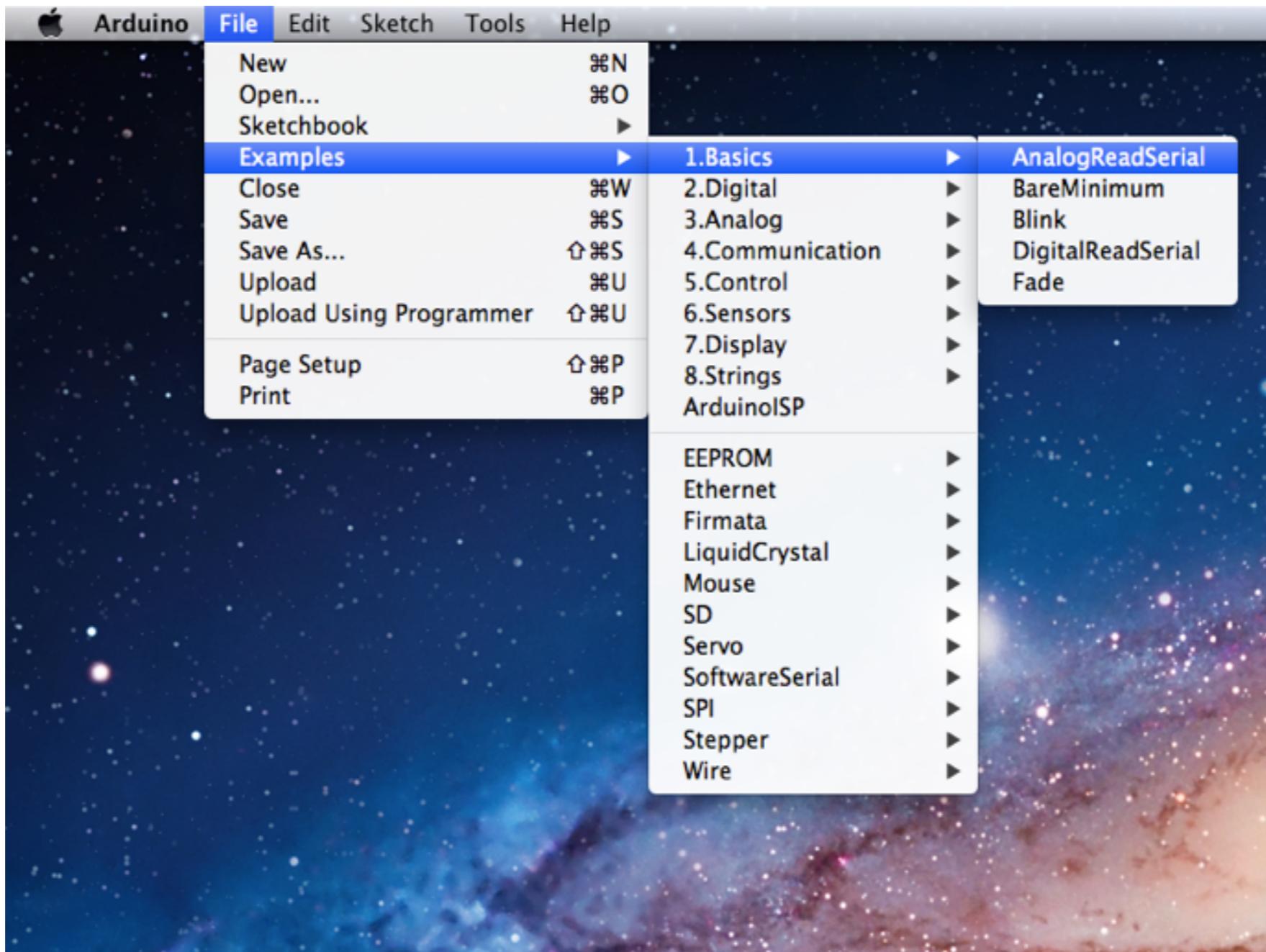
What happens to the LED when you turn the knob?

Connect a LDR*

* Light-Dependent Resistor



Back to this sketch...



Open [AnalogReadSerial](#) sketch
What is the range of possible values?

map() and constrain()



The screenshot shows the Arduino IDE interface with the title bar "AnalogReadSerial | Arduino 1.0-rc1". The code editor contains the "AnalogReadSerial" example. The code reads an analog input from pin A0 and prints it to the serial monitor. A red arrow points to the line "Serial.println(sensorValue);".

```
/*
 * AnalogReadSerial
 * Reads an analog input on pin 0, prints the result to the serial monitor
 *
 * This example code is in the public domain.
 */

void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(A0);
  Serial.println(sensorValue);
}
```

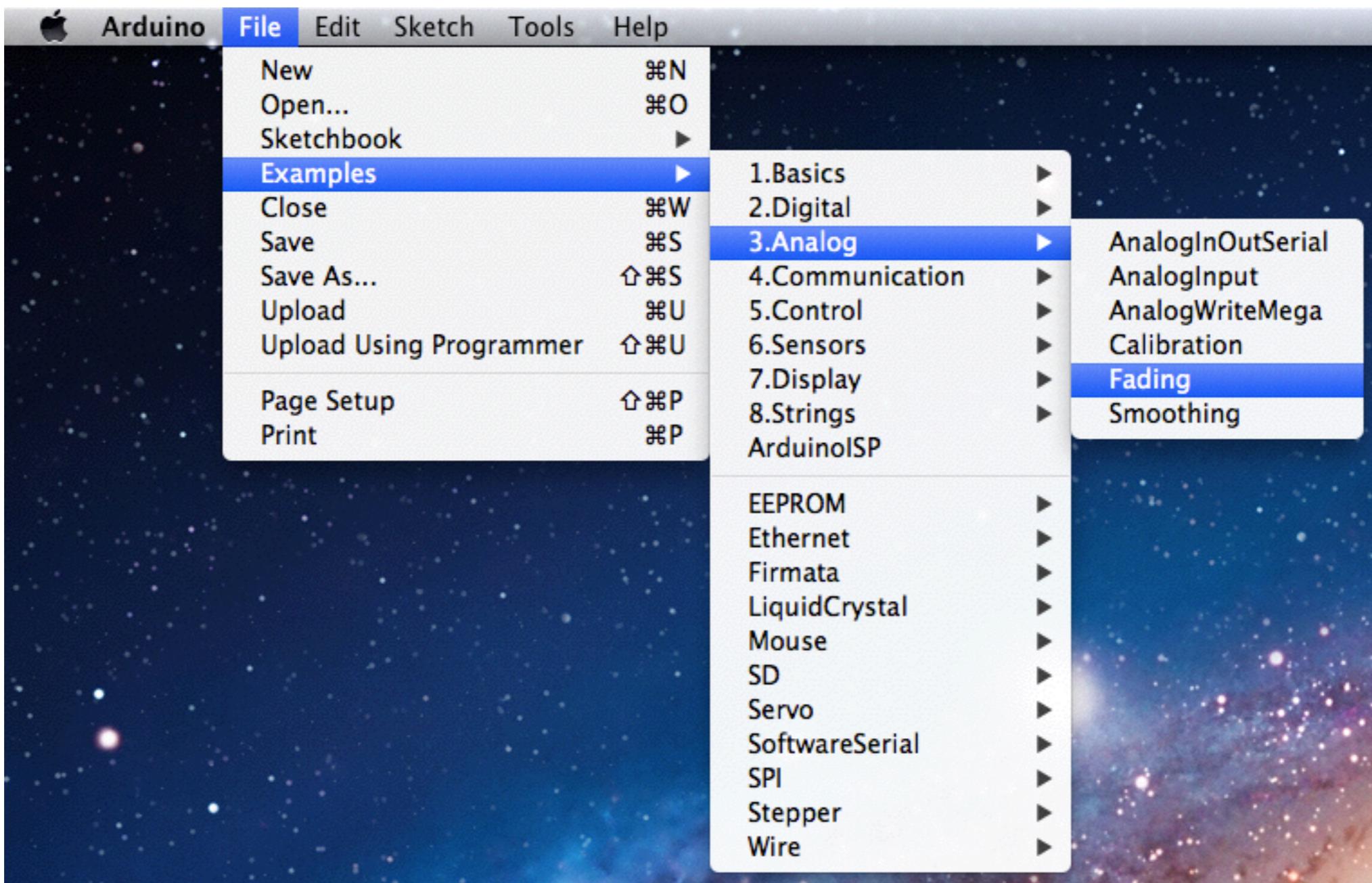
In here add:

```
int newValue = map(sensorValue,  
[...], [...], 0, 1023);
```

and change:

```
Serial.println(newValue);
```

Fading LEDs



Open **Fading** sketch
Hook up the LED to pin 9

Fading sketch

analogWrite()

What is the range of values?

```
for(start; finish; increment) {  
    // stuff in the loop  
}
```

```
Fading | Arduino 1.0-rc1  
Fading  
int ledPin = 9; // LED connected to digital pin 9  
  
void setup() {  
    // nothing happens in setup  
}  
  
void loop() {  
    // fade in from min to max in increments of 5 points:  
    for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5) {  
        // sets the value (range from 0 to 255):  
        analogWrite(ledPin, fadeValue);  
        // wait for 30 milliseconds to see the dimming effect  
        delay(30);  
    }  
  
    // fade out from max to min in increments of 5 points:  
    for(int fadeValue = 255 ; fadeValue >= 0; fadeValue -=5) {  
        // sets the value (range from 0 to 255):  
        analogWrite(ledPin, fadeValue);  
        // wait for 30 milliseconds to see the dimming effect  
        delay(30);  
    }  
}
```

8 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbserial-A9007ONC