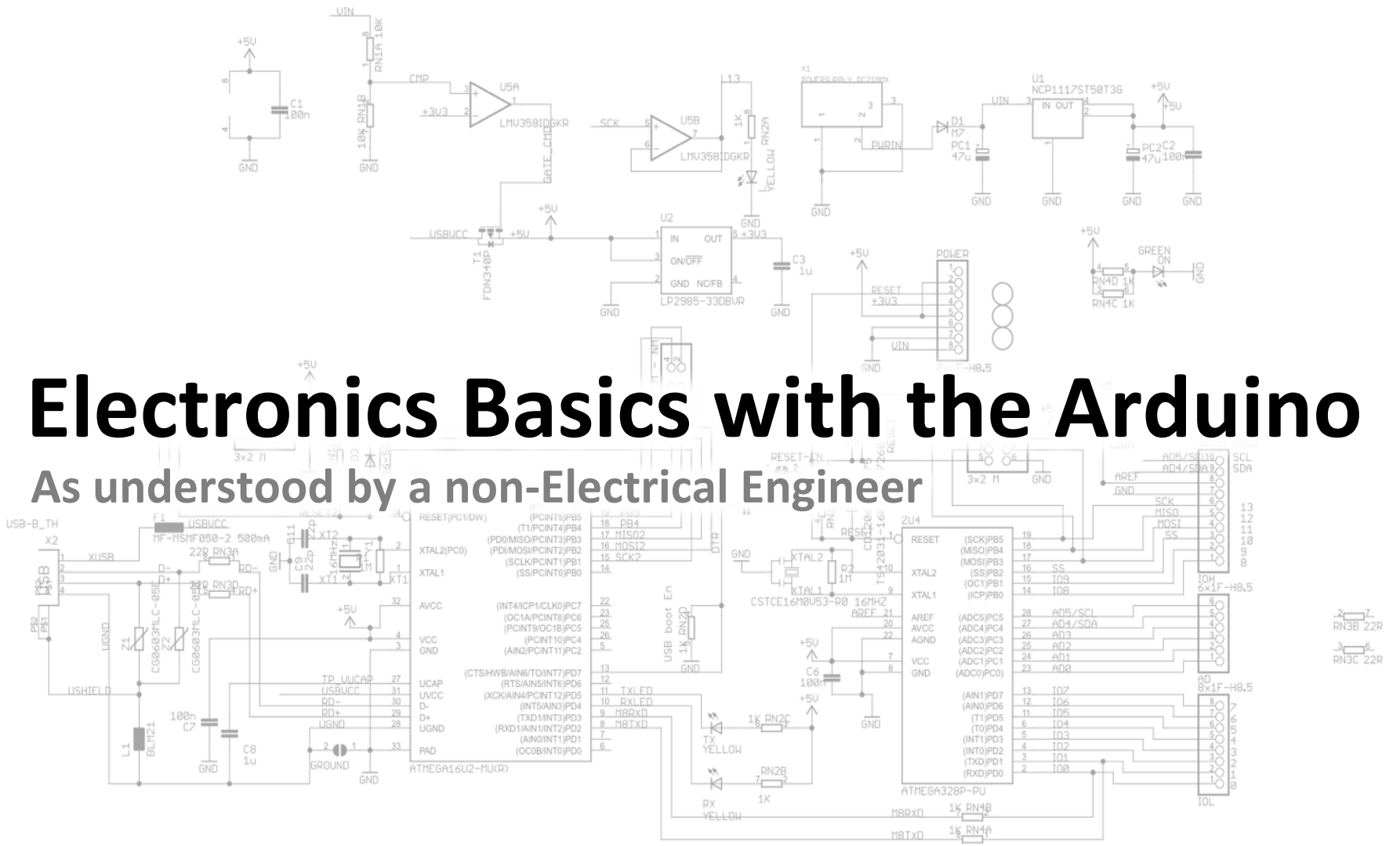
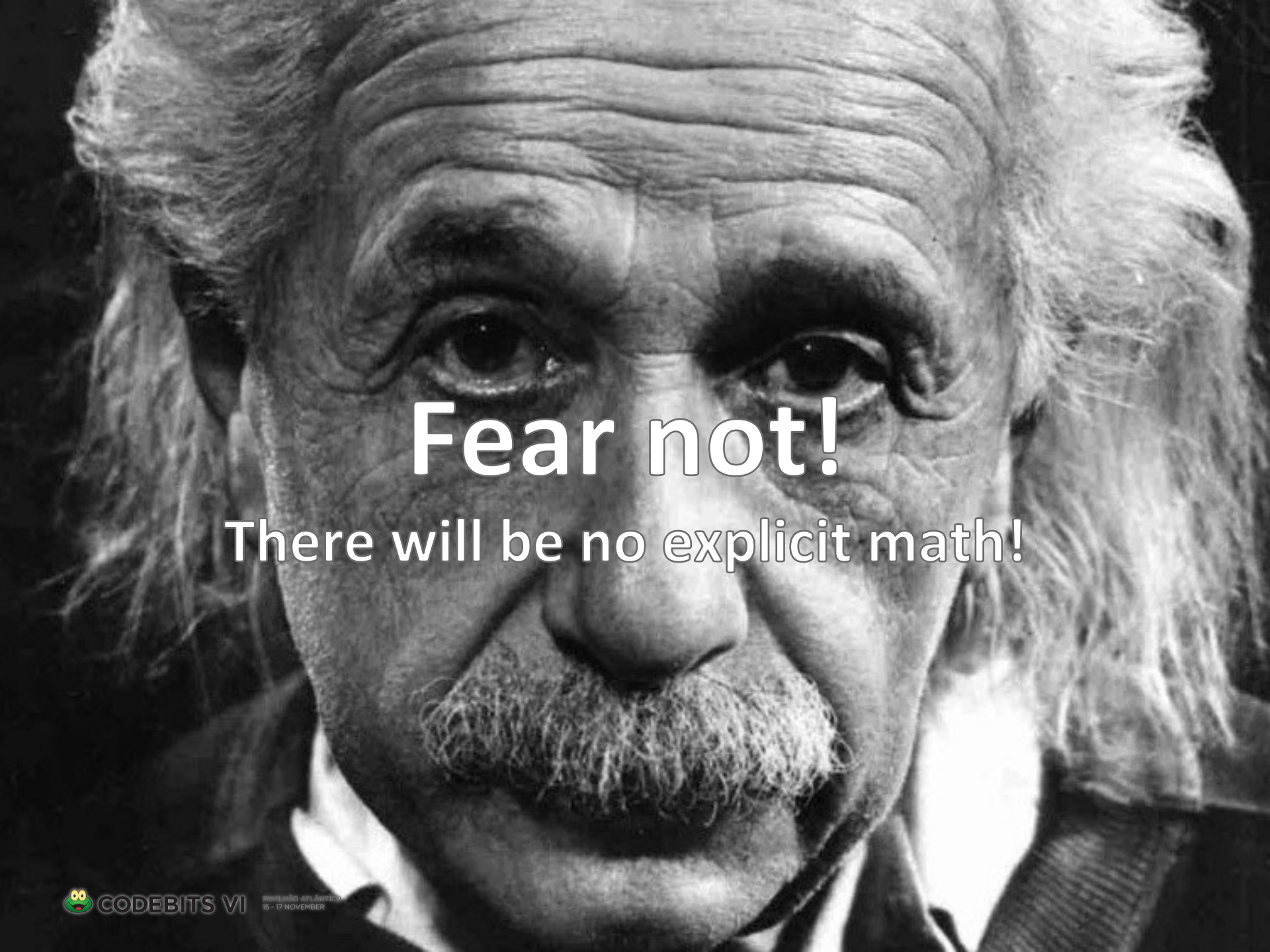
Spoken in Portuguese



Electronics Basics with the Arduino

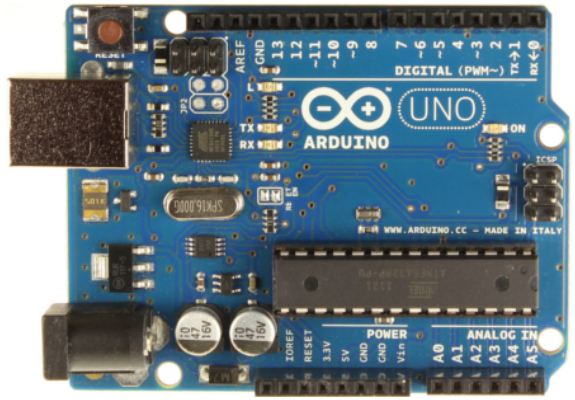
As understood by a non-Electrical Engineer

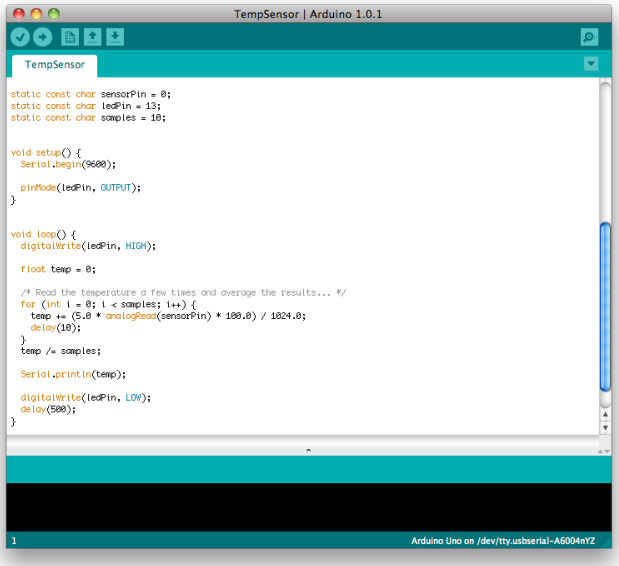
Carlos Rodrigues – 2012


Fear not!
There will be no explicit math!

The Arduino Platform

- A library abstracting low-level MCU* stuff
- A simplified IDE wrapping the GNU toolchain

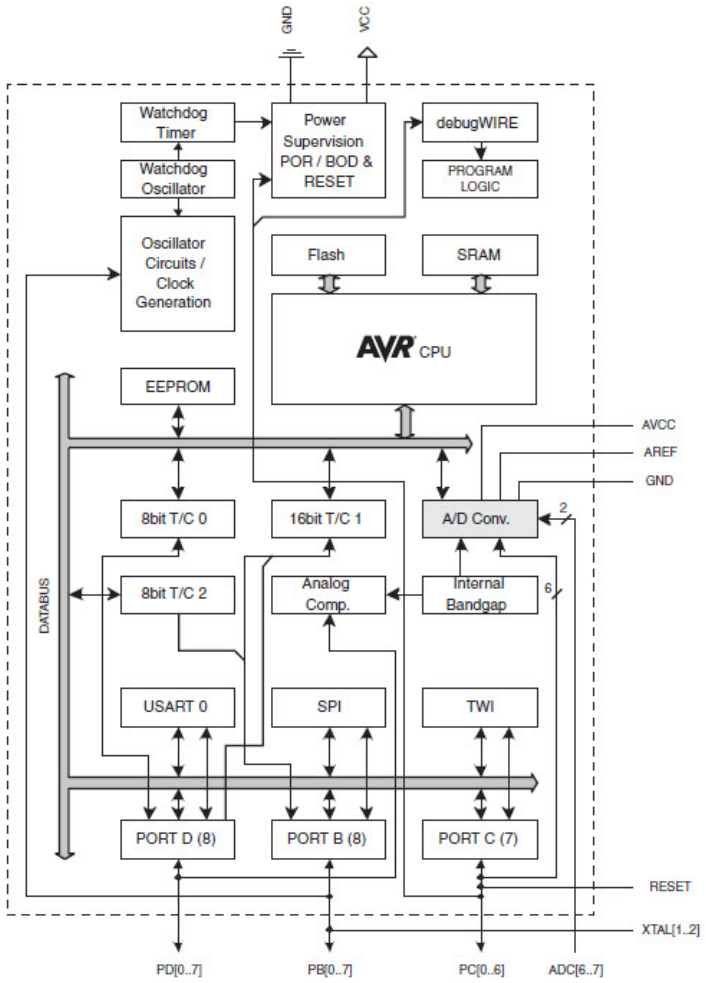




- Support circuitry around the ATmega 8-bit MCU*
- A bootloader to upload and run user programs

*Micro-Controller Unit

Hardware Specs (Arduino Uno)



- ATmega328P 8-bit MCU @ 16 MHz
- 2 KB of RAM
- 32 KB of Flash memory (programs)
  …plus 1 KB of EEPROM (persistent state)
- 14 digital I/O pins
  …6 with PWM capability (8-bit)
  …2 supporting interrupts
- 6 analog input pins (10-bit)

Electrical Specs (Arduino Uno)

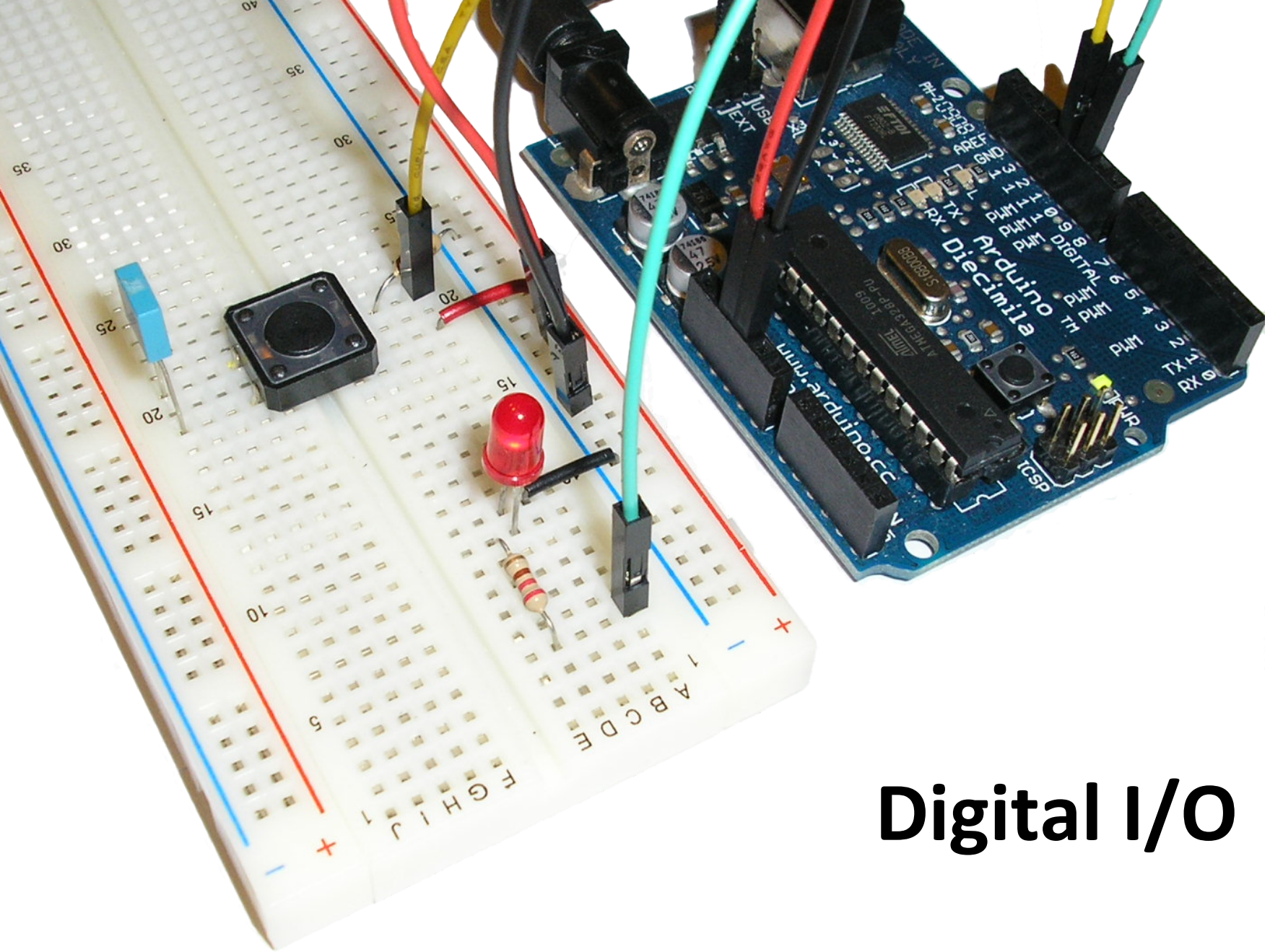
- Board components run at **5 V**
- Power supply can be **7** to **12 V**
  …can go up to 20 V, but try to avoid it
- Each I/O pin can handle **20 mA** (safely)
  …but **avoid** more than **100 mA** for the sum of all I/O pins
  …and **never** give them more than 5 V or negative voltages

- The “5V” pin can source about **100 mA** (without much heat)
  …but the unregulated V_{in} pin (power supply voltage) can handle up to **1 A**

So, get one of these…

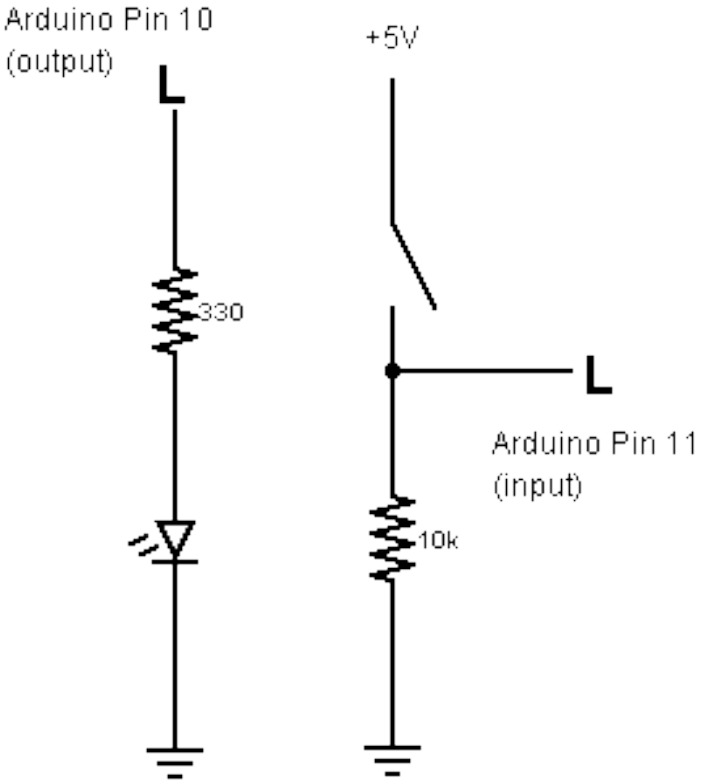
- Voltage (V_{dc} and V_{ac})
- Current (for both DC and AC)
- Resistance (Ω)
- Capacitance (F)
- Transistor gain (hFE)
- Continuity check
- Temperature (°C)

Any brand will do, but don't be cheap…


Arduino
DieCimila
www.arduino.cc
ATMEGA328P-PU
FTDI
AREF
GND
TX
RX
DIGITAL
PWM
TCSP

Digital I/O

Our First Circuit (Digital I/O)

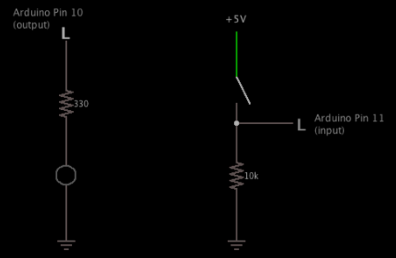



```
// Toggle an LED by pressing a button...

static const char led_pin = 10;
static const char button_pin = 11;

boolean led_state = LOW;
boolean last_button_state = LOW;

void setup() {
  pinMode(led_pin, OUTPUT);
  pinMode(button_pin, INPUT);
}

void loop() {
  boolean button_state = digitalRead(button_pin);

  if (button_state == HIGH && last_button_state == LOW) {
    led_state = !led_state;

    digitalWrite(led_pin, led_state);
  }

  last_button_state = button_state;
}
```

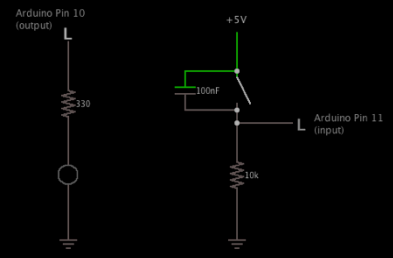
Binary size: 1082 bytes

Simulation

Simple Digital Input and Output



Simple Digital Input and Output (debounced)



This works because Arduino digital pins have input hysteresis (Schmitt-Trigger).

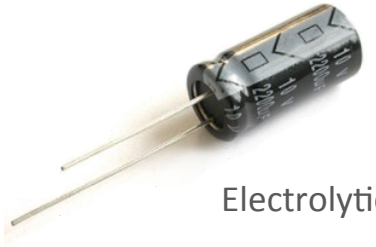
Switch Debounce with Capacitors

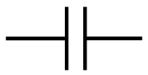
Polyester

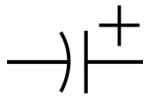
Tantalum

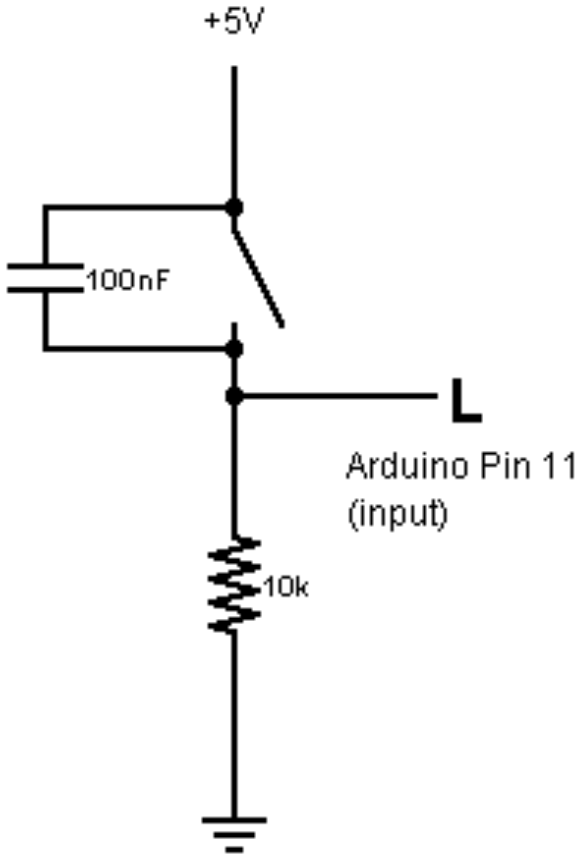
Ceramic

Electrolytic

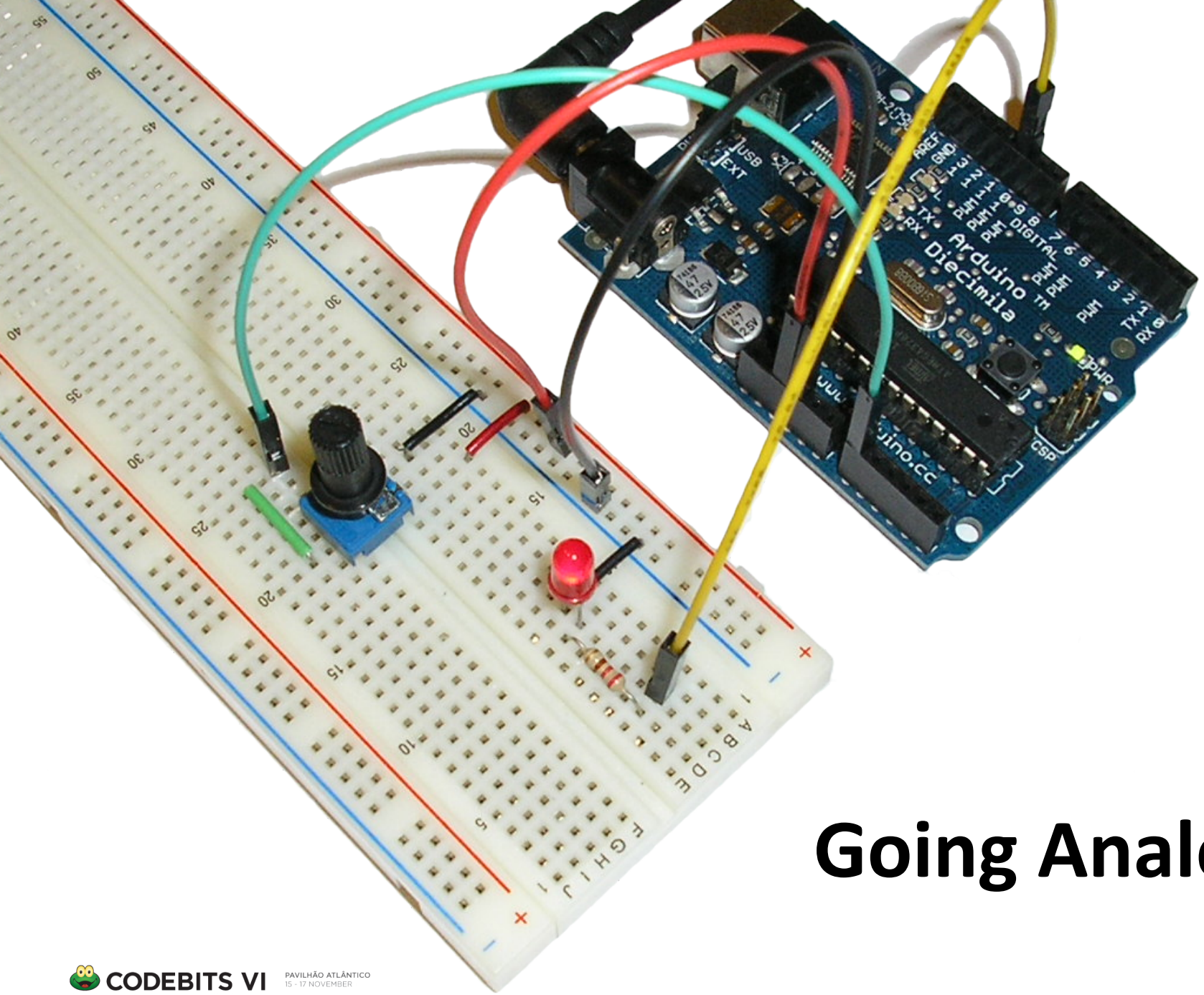






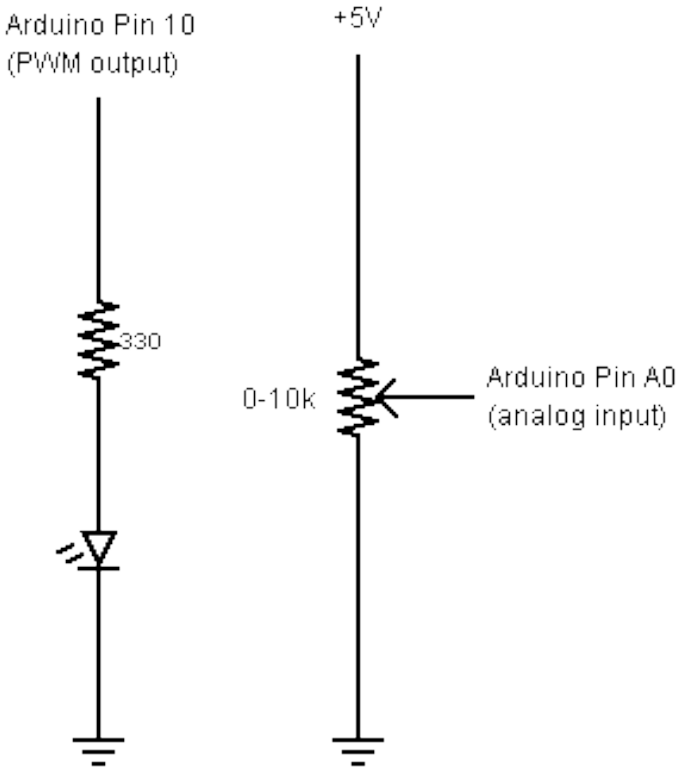


This works because Arduino digital pins have input hysteresis (Schmitt-Trigger).



Going Analog

Analog Input and PWM* Output



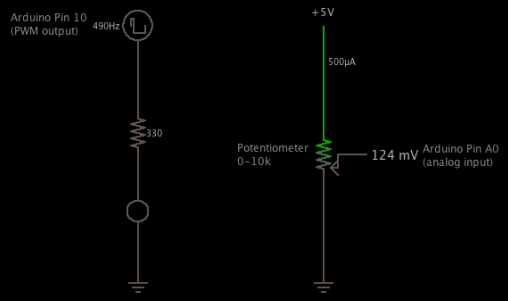

```
// Control an LED brightness using a potentiometer...


static const char led_pin = 10;
static const char pot_pin = A0;


void setup() {
  pinMode(led_pin, OUTPUT);
}


void loop() {
  short pot_value = analogRead(pot_pin);
  short brightness = map(pot_value, 0, 1023, 0, 255);

  analogWrite(led_pin, brightness);

  delay(10);
}
```

*Pulse-Width Modulation

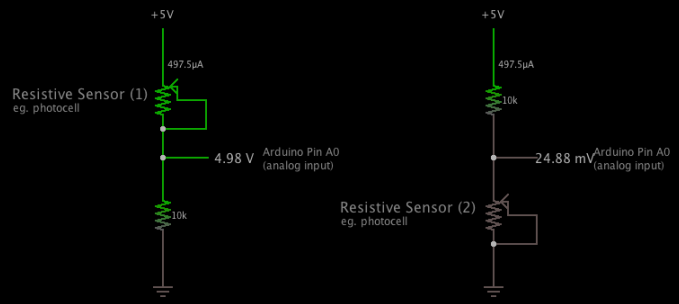
Binary size: 1764 bytes

Simulation

Simple Analog Input and Output



Resistive Analog Input (Voltage Divider)

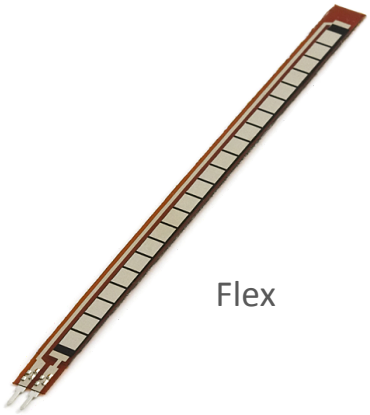


Analog Input: Resistive Sensors

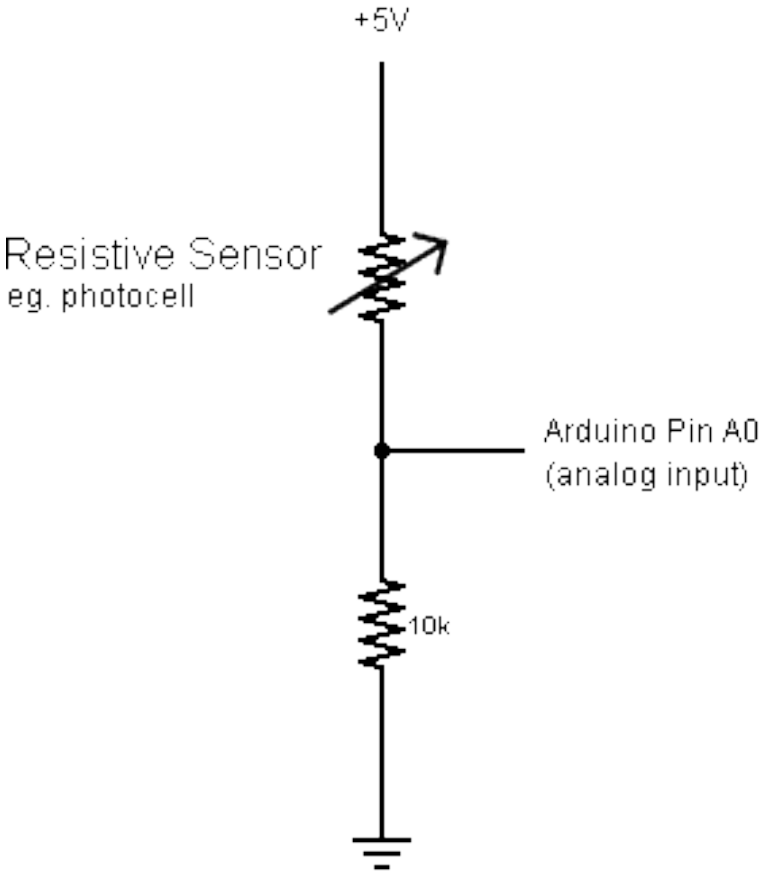
Light

Temperature

Force

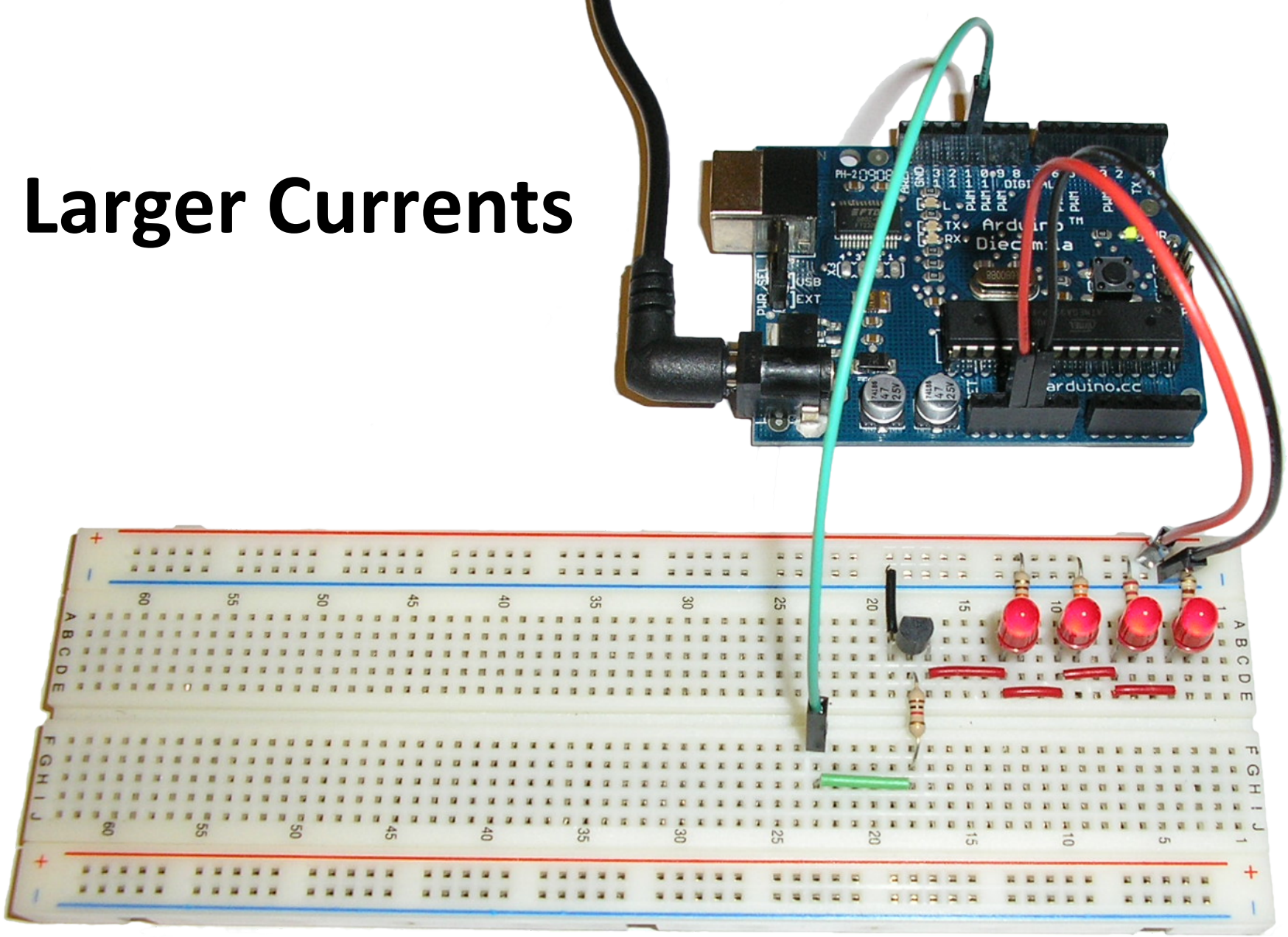


Flex

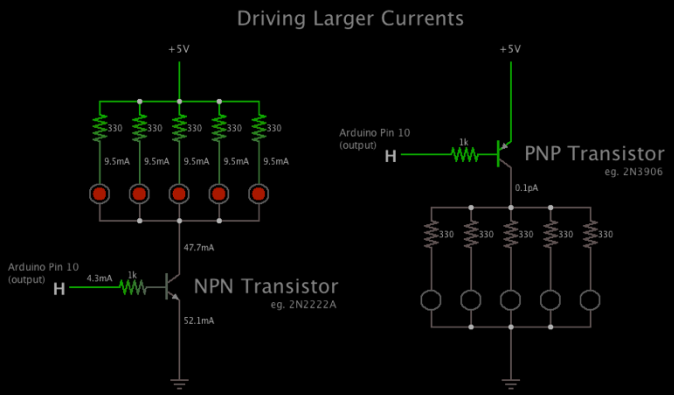


Many sensors are just variable resistors...

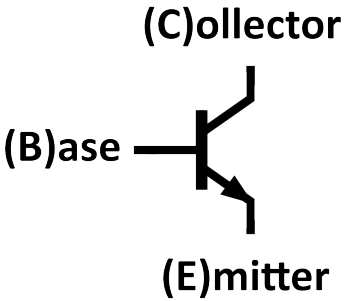
Larger Currents

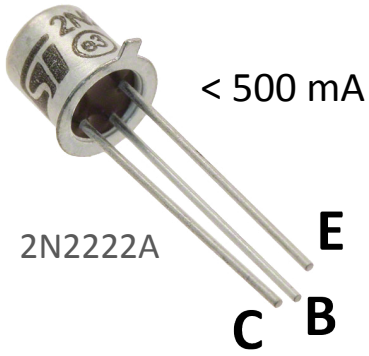


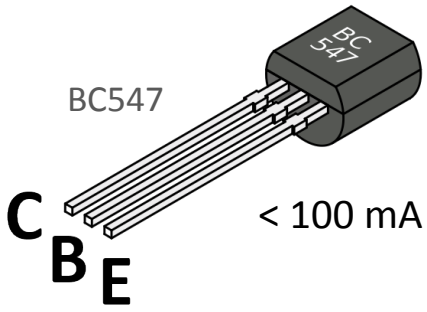
Simulation

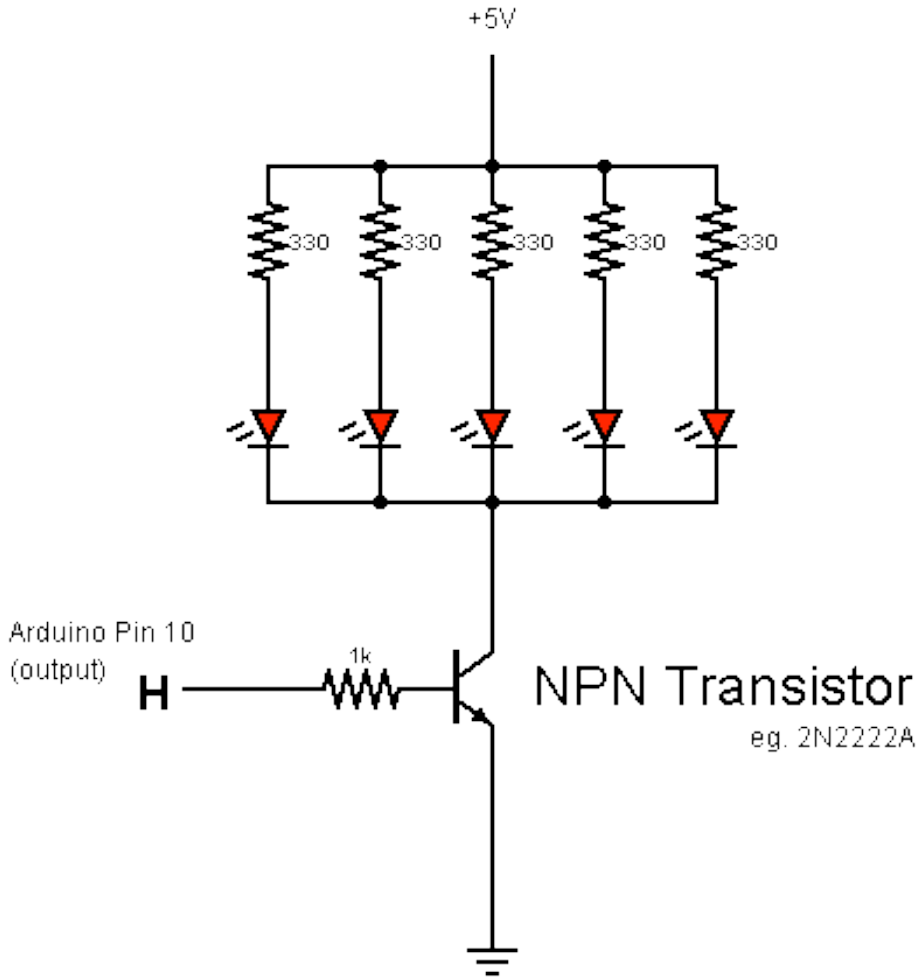

Driving Larger Currents
+5V
330
330
330
330
330
9.5mA
9.5mA
9.5mA
9.5mA
9.5mA
47.7mA
Arduino Pin 10
(output)
H
4.3mA
1k
NPN Transistor
eg. 2N2222A
52.1mA
+5V
Arduino Pin 10
(output)
H
1k
PNP Transistor
eg. 2N3906
0.1pA
330
330
330
330
330

Driving Larger Currents: BJTs*

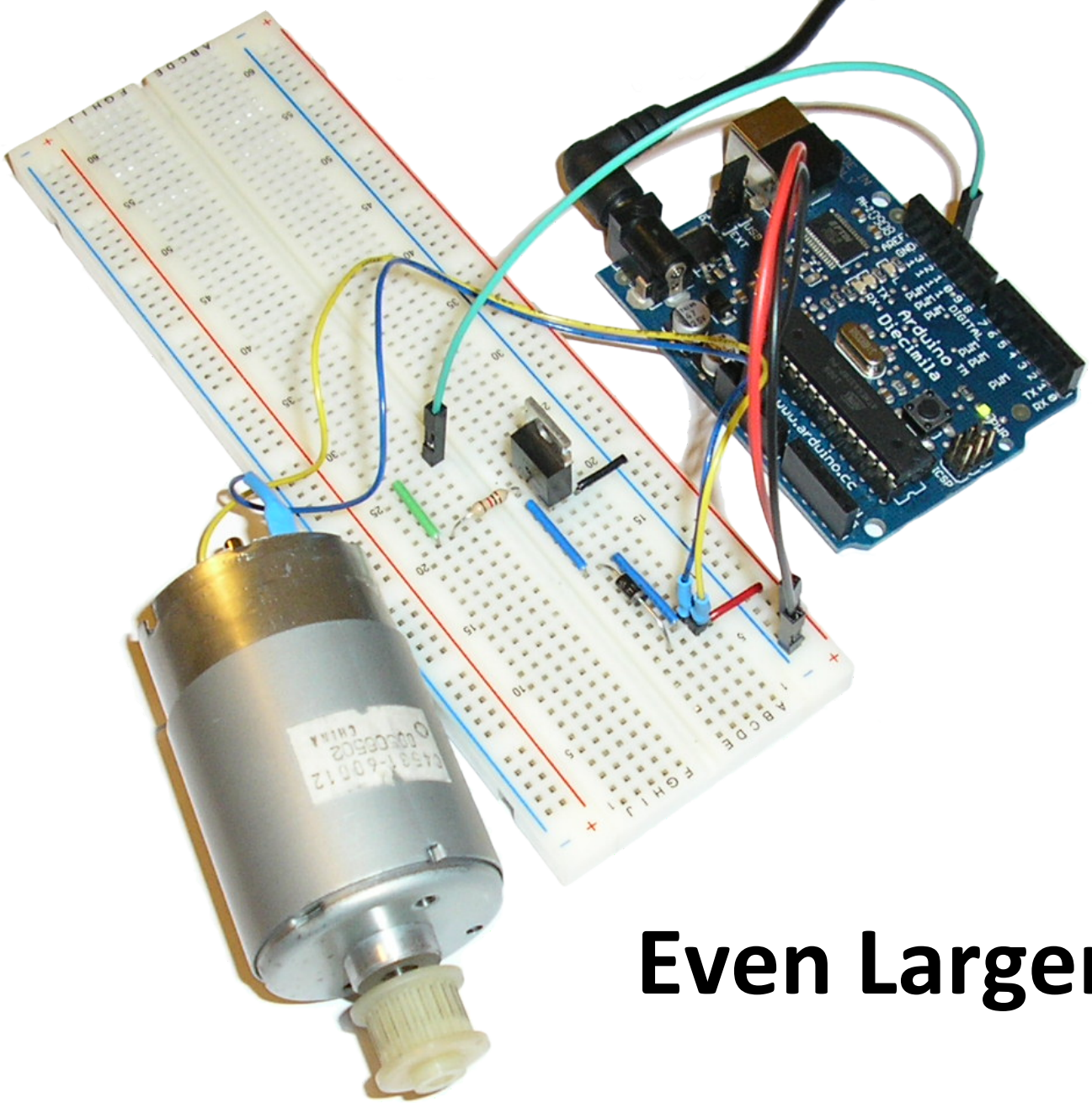








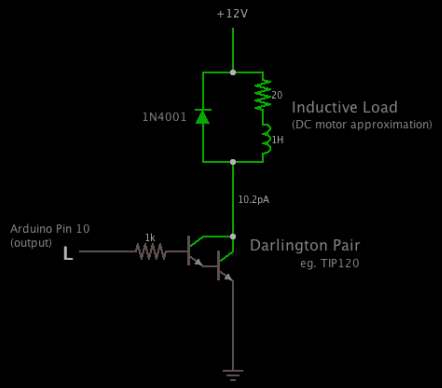
*Bipolar Junction Transistors



Even Larger Currents

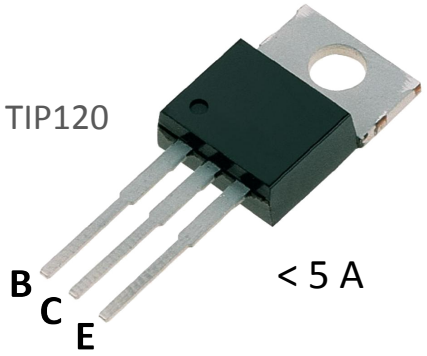
Simulation

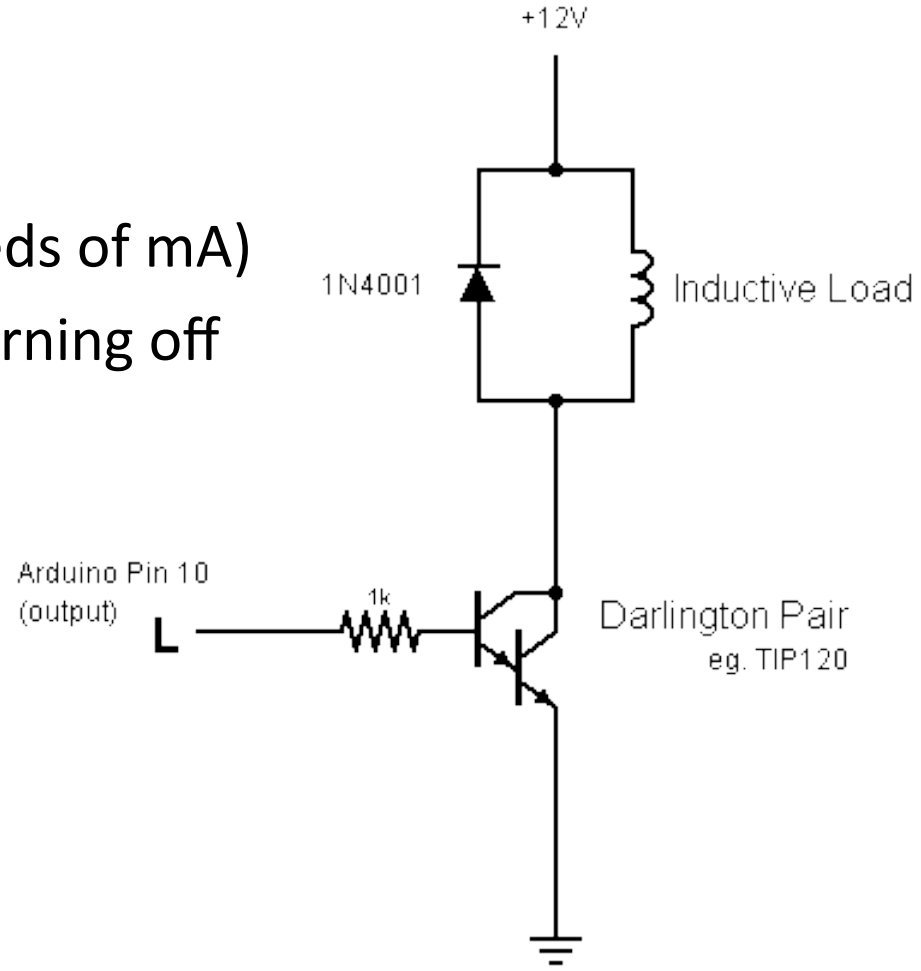
Driving Even Larger Currents (Darlington)

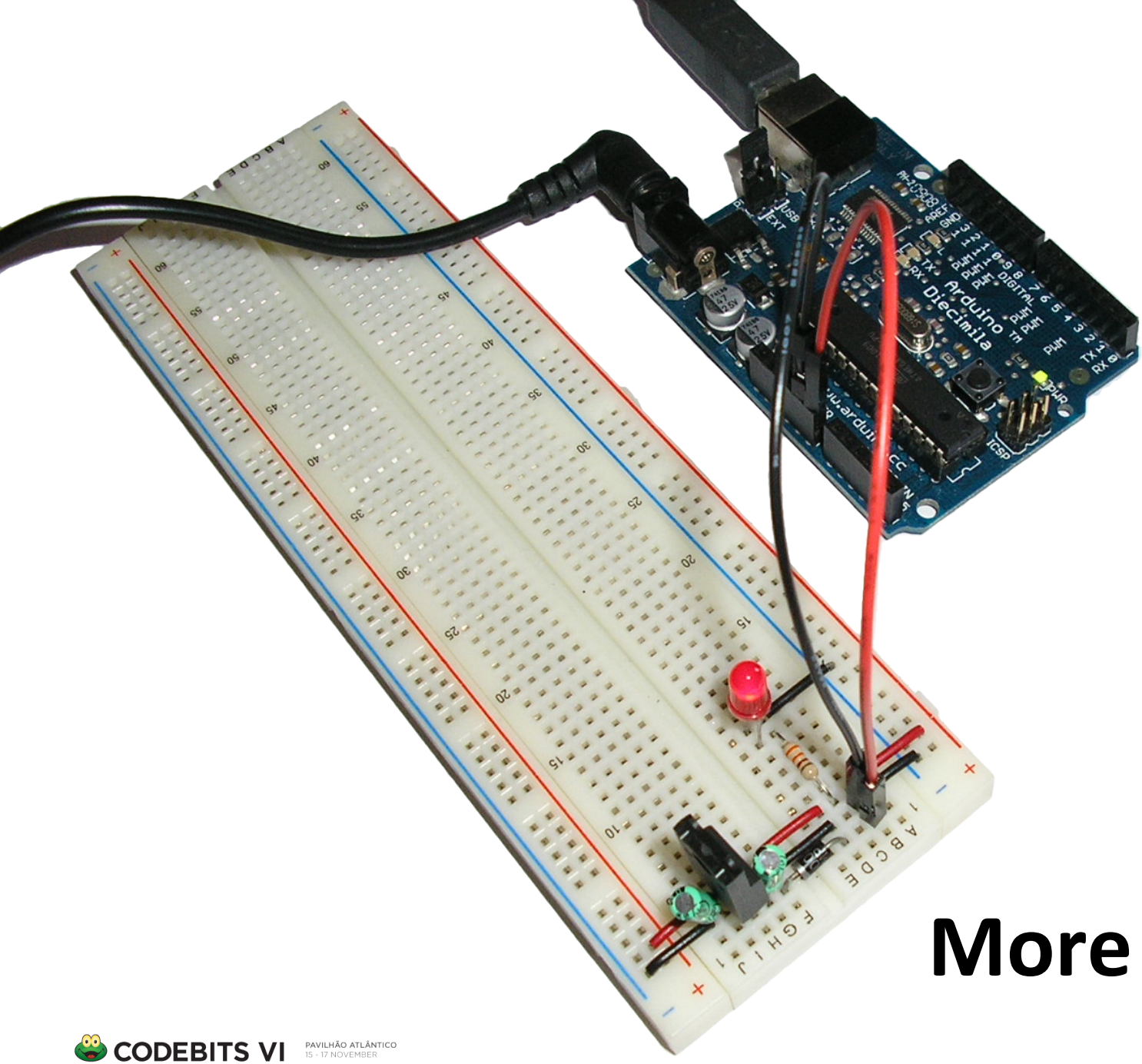


Inductive Loads

- Motors, solenoids, relays…
- Draw large currents (hundreds of mA)
- Produce “kickback” when turning off

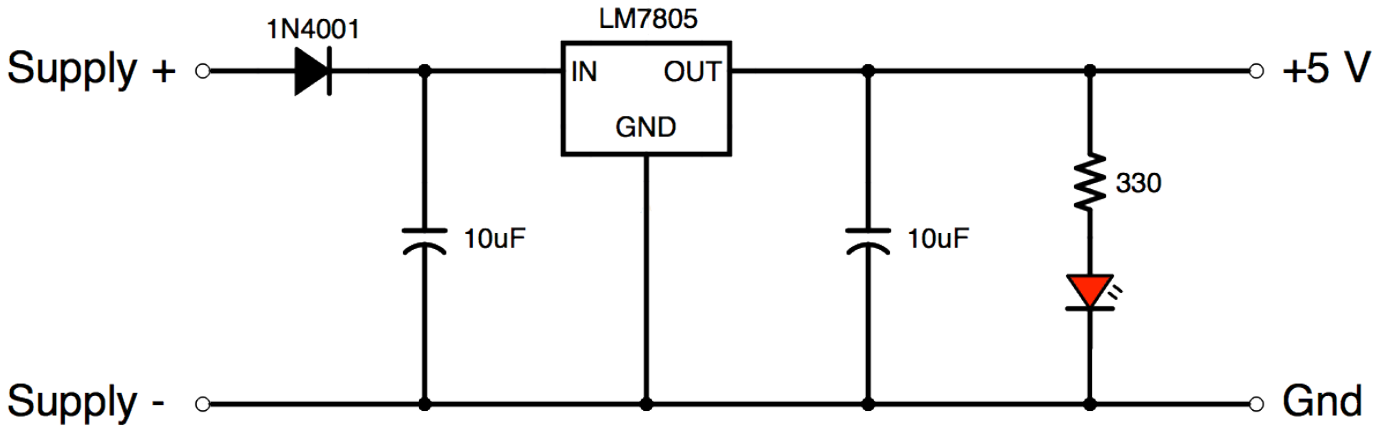




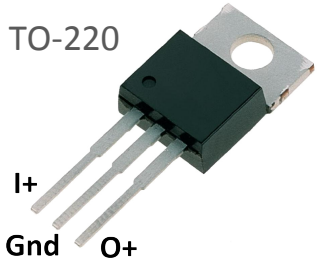


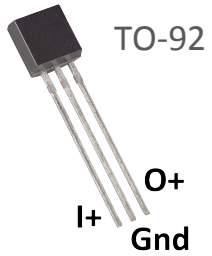
More Power

Voltage Regulation

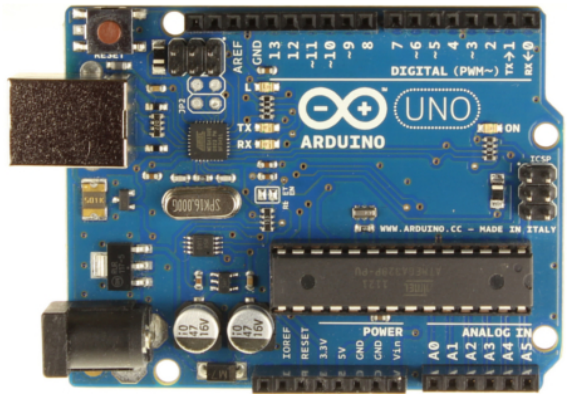


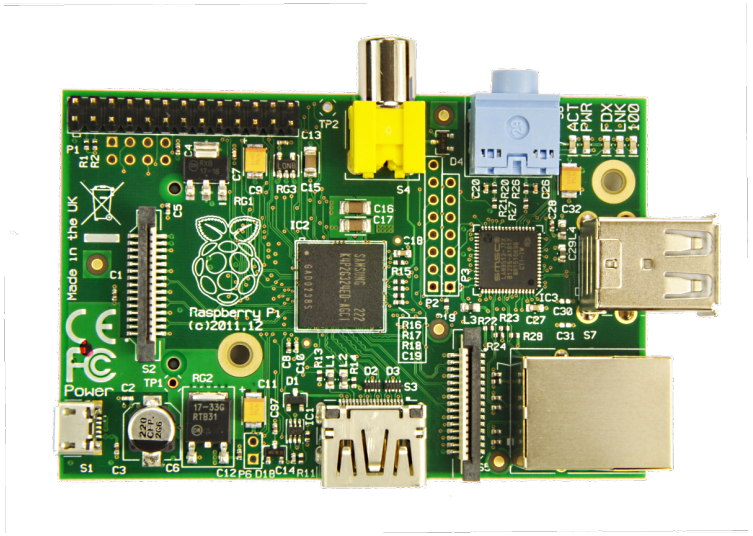
- 5 V (LM78<u>05</u>), 12 V (LM78<u>12</u>) and more
- Input voltage up to 35 V
- Max. 100 mA (TO-92) or **1 A** (TO-220)
- Capacitors **<u>required</u>** for proper regulation





What about the Raspberry Pi?





- 20€
- Robust GPIOs
- Analog inputs and several PWMs
- Low power consumption
- Replaceable microcontroller (< 5€)

- 35€
- Far greater computing power
- Network connectivity out-of-the-box
- Multiple programming languages
- Runs a “standard” Linux OS

The Arduino makes for a great addition to the RPi...

Thanks for Listening!

Any Questions?

This presentation and simulations can be found at:
www.carlos-rodrigues.com/files/codebits-2012

Carlos Rodrigues

cefrodrigues@gmail.com
twitter.com/carlosefr