

Electronics for Inventors

MicroPython edition

Thomashudson.org

Portland Community College
Microcontroller and Soldering

General Instructions:

<https://docs.micropython.org/en/latest/esp8266/esp8266/tutorial/index.html>

<https://learn.adafruit.com/micropython-basics-loading-modules/overview>

My PCC Website:

<http://www.pcc.edu/staff/index.cfm/1514.html>

Electronics for Inventors

Arduino is a microcontroller, computer, and prototyping platform developed in 2005 in Italy.

A prototyping platform allows you to test out your ideas. You can test multiple ideas see which ones work.

The **code** is a list of instructions that are written in a programming language. Arduino is C++ and compiled into machine code while MicroPython is interpreted. The computer executes Python similar to how we read instructions.



```
Blink | Arduino 1.0.5
File Edit Sketch Tools Help

Blink$

int led = 13;

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

void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Example [\[edit\]](#)

```
:10010000214601360121470136007EFE09D2190140
:100110002146017EB7C20001FF5F16002148011988
:10012000194E79234623965778239EDA3F01B2CAA7
:100130003F0156702B5E712B722B732146013421C7
:00000001FF
```

-  Start code
-  Byte count
-  Address
-  Record type
-  Data
-  Checksum

Sample Projects

Cat Door:

http://www.youtube.com/watch?feature=player_detailpage&v=jiS18IxrSM4



Sample Projects

Book Alarm

http://www.youtube.com/watch?feature=player_embedded&v=OJA5uA5gmDo



Sample Projects

Turn Signals

http://www.youtube.com/watch?feature=player_embedded&v=ZtNEPkwCfxA



Sample Projects

R/C Lawn Mower

http://www.youtube.com/watch?feature=player_embedded&v=wbOZa9SQPpA



Thomas Hudson.org



http://www.youtube.com/watch?feature=player_embedded&v=LRsdcSe9cuM

Thomas Hudson.org



OMSI Exhibits

The light based reactive component we have accomplished through similar installations of bucket walls (pickathon 2016).



NODE MCU (ESP8266) \$5



ONE ON EACH
PELOUM MONITORS
~~ACCELEROMETER~~ AND
~~DRIVES~~ LEDS
RECEIVES/SENDS WSPS DATA
FROM ROUTER

SERIAL

SSDN \$5
APRIST ACCELEROM.

MINI \$3
LEONARDO 32V4

MONITORS ACCELEROMETER
FOR HITS AND
DRIVES LEDS

I2C

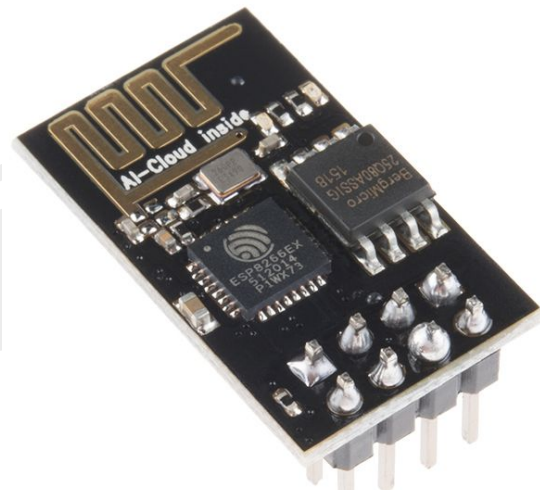
WS2812 LEDS
RGB ADDRESSABLE
~\$10/METRE

TEENSY 3.6
& AUDIO SHIELD

RECEIVES PELOUM HIT DATA
AND PLAYS AUDIO STORED IN
MEMORY VIA 1/8 AUDIO



[Pickathon 2017](#)

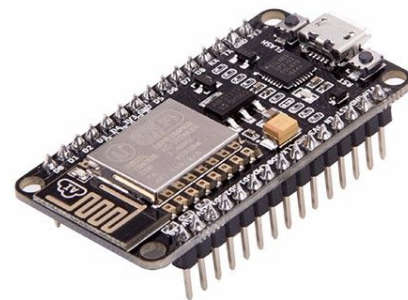
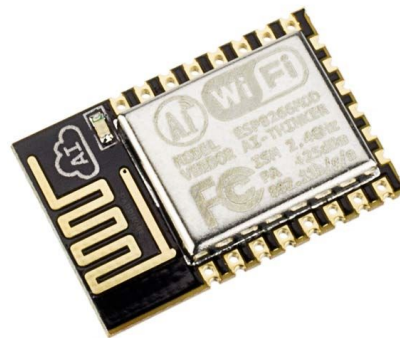
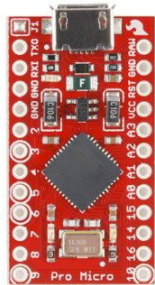


Pro Mini

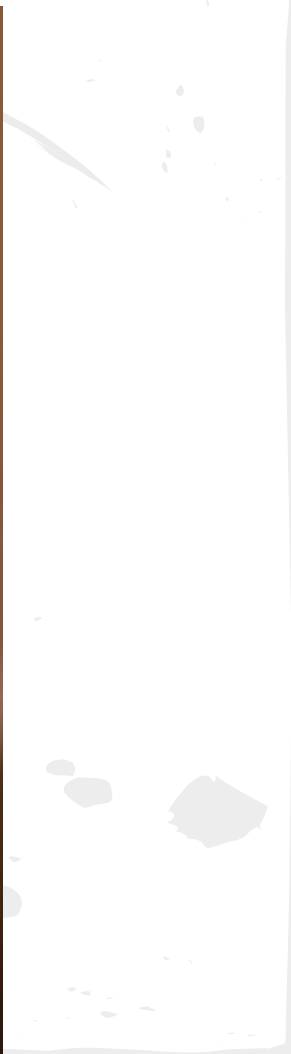
Pro Micro

Nano

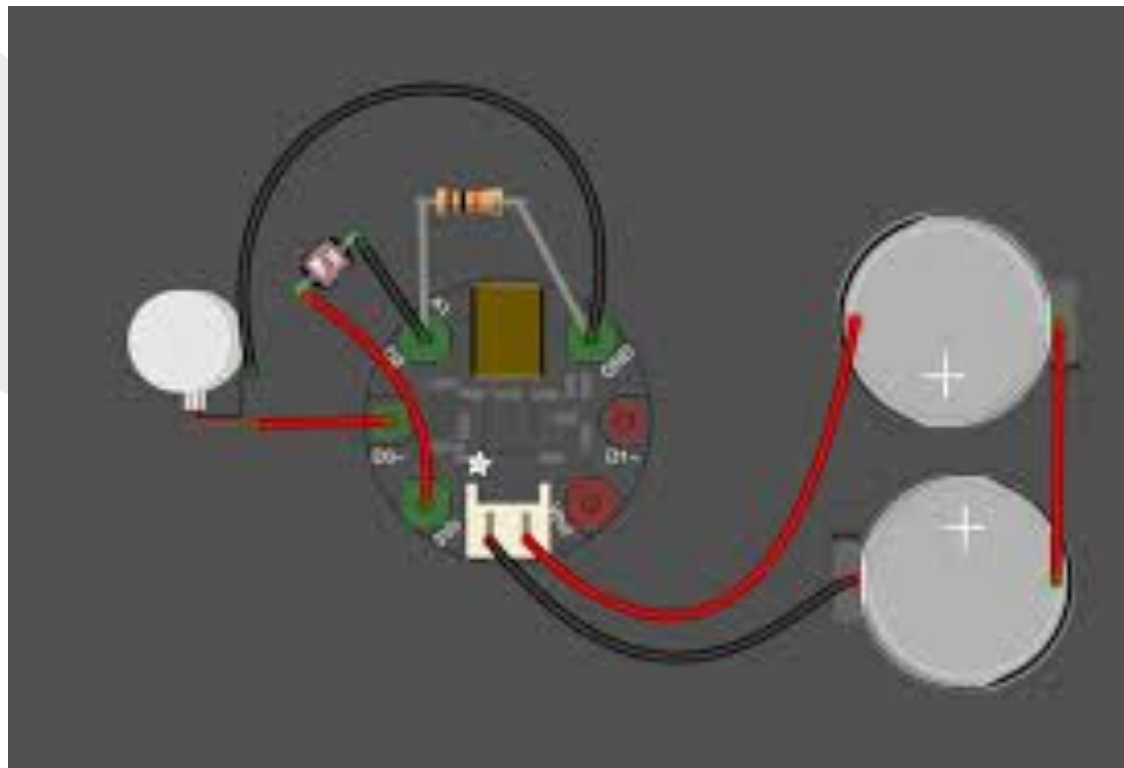
Micro













adafruit
sparkfun

Community and Beyond

[Dorkbot](#) has meetings every other week where people bring and discuss their projects without a structured format. An amazing array of people available to help you get your projects done. Meets at PDX Hackerspace... control H, see below

Digital Classes

- [Adafruit](#) has some classes and tutorials online
- [Instructables](#). Learn and share from other peoples blunders here!
- [Sparkfun](#) has self taught classes and tutorials here developed in part by our Portlander Derek Runberg

Brick and Mortar Classes in Portland mostly

- [Control H](#) - Portland's electronics Maker Space aka PDX Hackerspace
- PNCA offers an [Electronics for Artists class](#) by famous Portlander Mark Keppinger
- PNCA offers an advanced [Arduino Class](#) by famous Portlander Donald Delmar Davis
- [ADX](#) sometimes offers classes at their maker space in SE Portland
- Dorkbot sometimes offers free or almost free classes at Control H
- Seattle's [Metrix](#) (maker space) has circuit church every sunday from 6pm-midnight. I highly recommend going to this if you are in Seattle. They also offer other intro to electronics courses that are probably pretty good too.

Electronic Parts and Tools Resources

- [Sparkfun](#). Get your proven prototyping supplies here
- [Instructables](#). Learn and share your inventions here
- [OSHPark](#). Get your locally manufactured Printed Circuit Boards here.
- [Adafruit](#). This is like Sparkfun but a bit more raw and generously priced gadgets
- [Dorkbot](#) has its own list here.

Portland Mini
Maker Faire

Build. Break. Make.

OMSI mini-maker faire
Seattle mini-maker faire
[Instructables.com](https://www.instructables.com)

Introductions

Why are you here?

What are you interested in learning?

Do you have a specific project you are working on?

Peripheral Devices

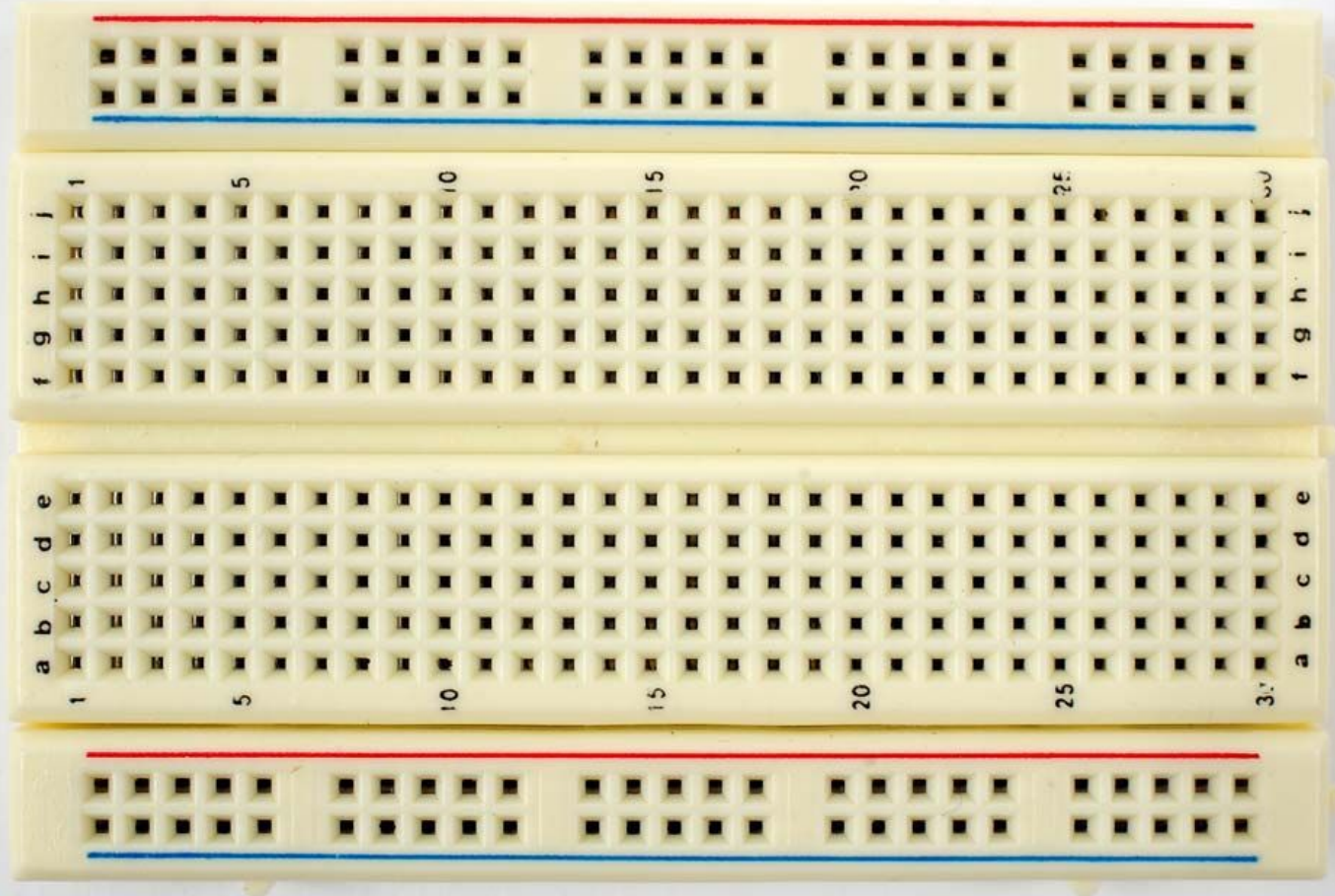
Sensors, Actuators, I/O Devices

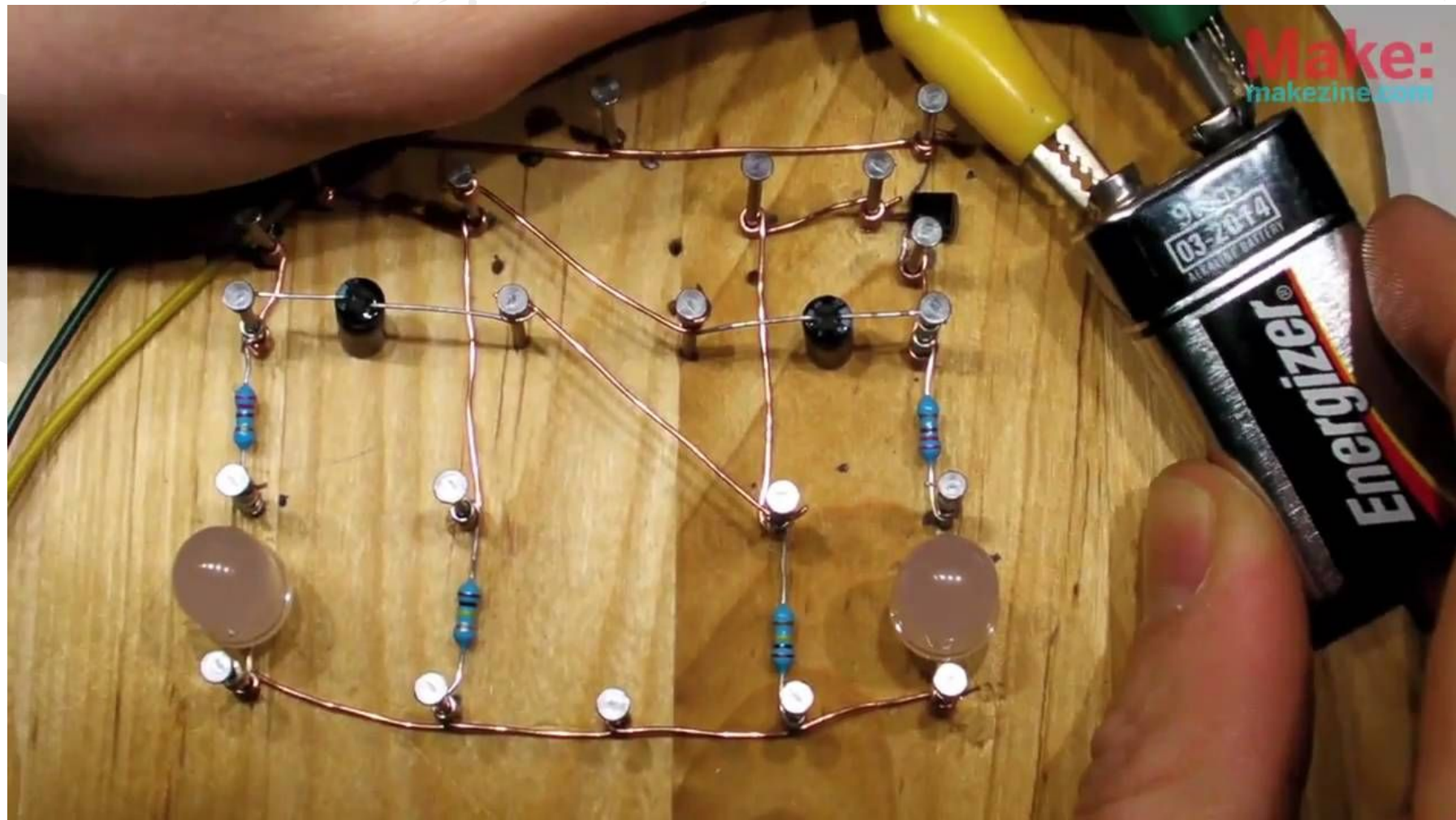
Digital Input: buttons, water meter

Digital Output: LEDs, lock?

Analog Input: Light sensor, flex sensors

Analog Output: LEDs, Sounds





New Ctrl+N

Open... Ctrl+O

Sketchbook

Examples

Close Ctrl+W

Save Ctrl+S

Save As... Ctrl+Shift+S

Upload Ctrl+U

Upload Using Programmer Ctrl+Shift+U

Page Setup Ctrl+Shift+P

Print Ctrl+P

Preferences Ctrl+Comma

Quit Ctrl+Q

01.Basics

02.Digital

03.Analog

04.Communication

05.Control

06.Sensors

07.Display

08.Strings

09.USB

10.StarterKit

ArduinoISP

Teensy

Adafruit_NeoPixel

AnalogReadSerial

BareMinimum

Blink

DigitalReadSerial

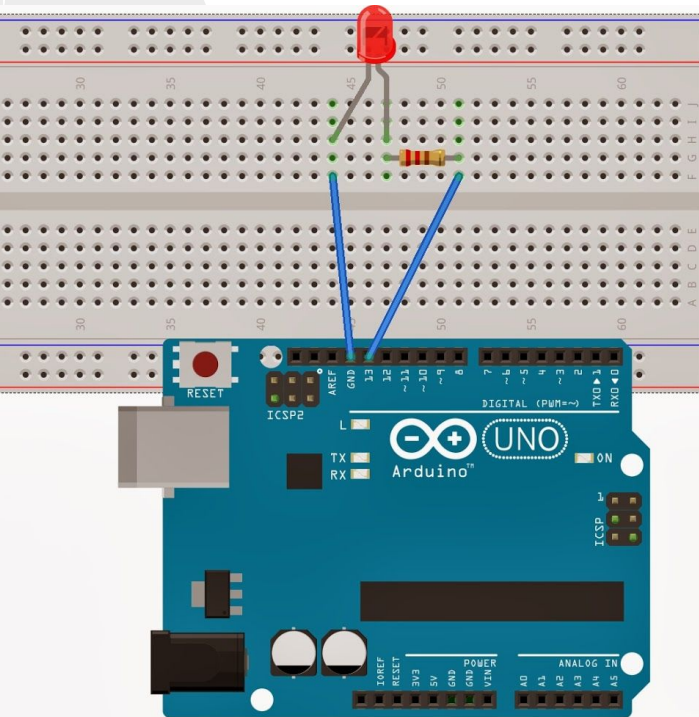
Fade

ReadAnalogVoltage

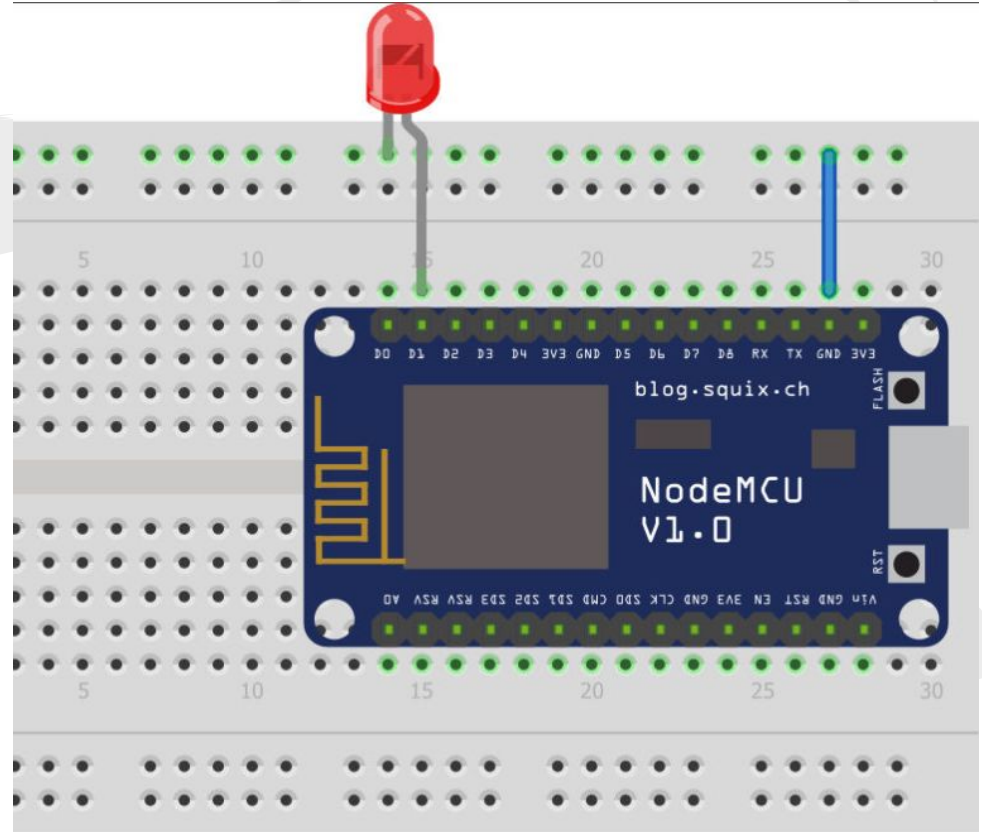


Arduino Example Sketch

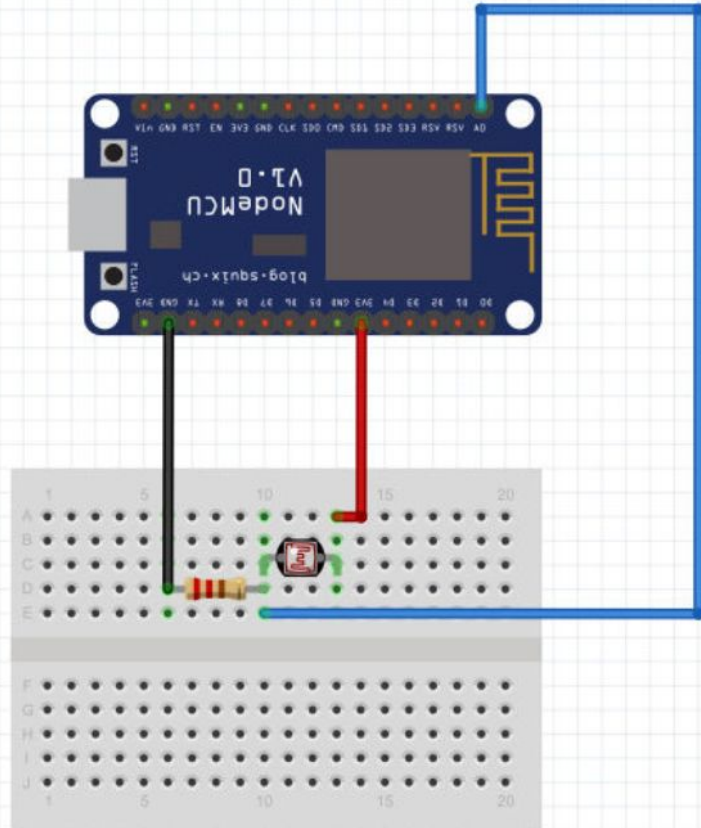
BlinkYourLED.ino



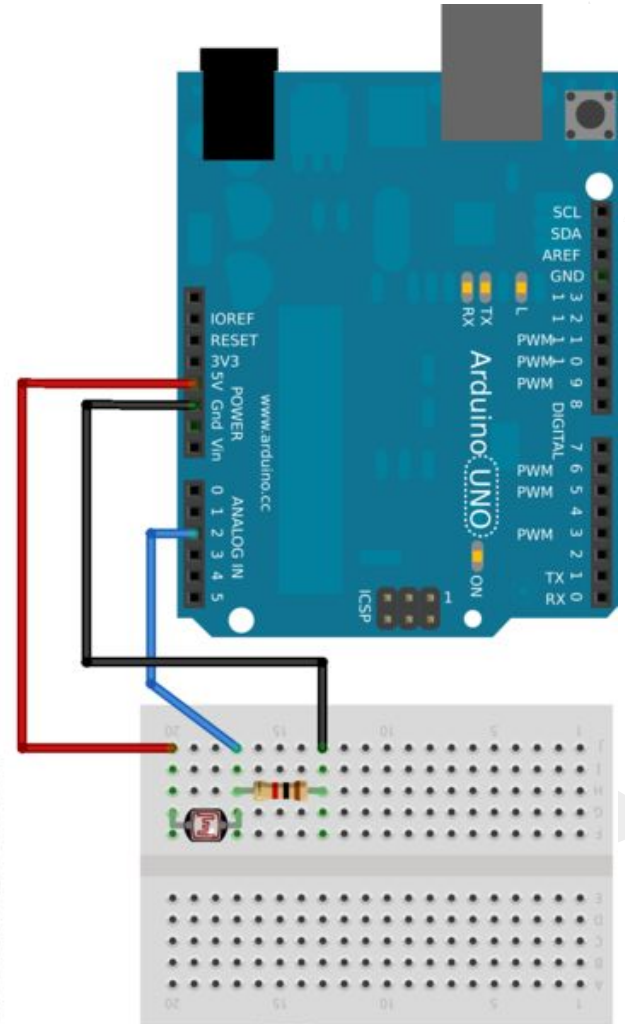
Made with  Fritzing.org



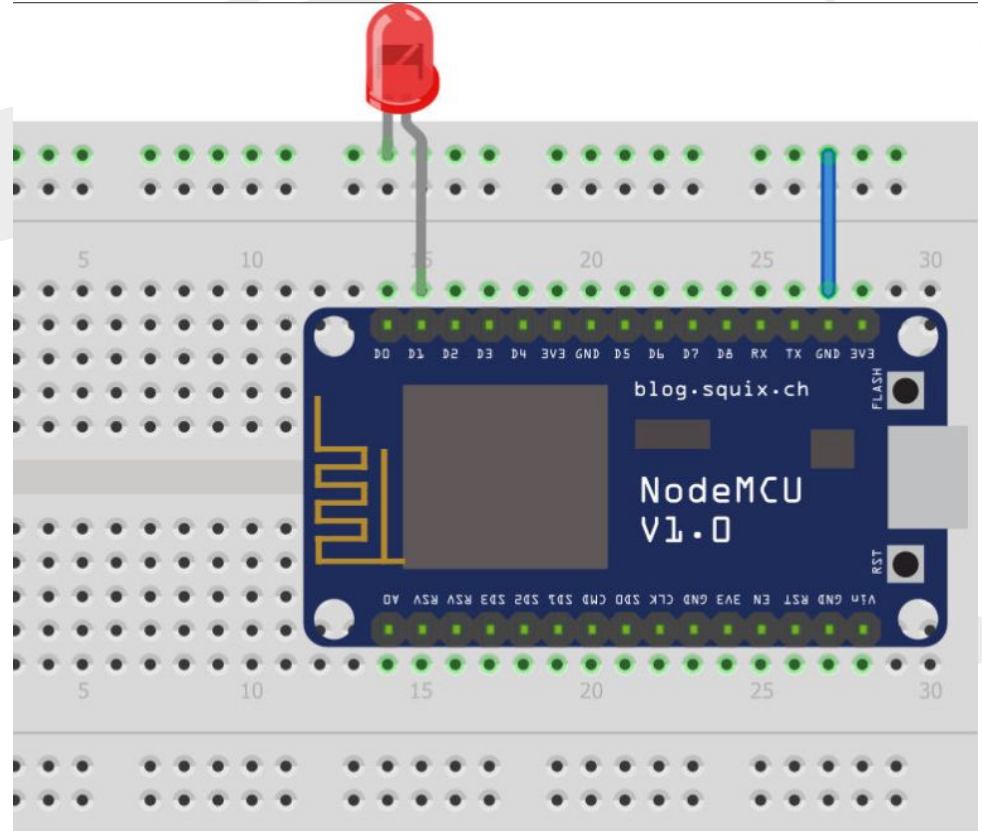
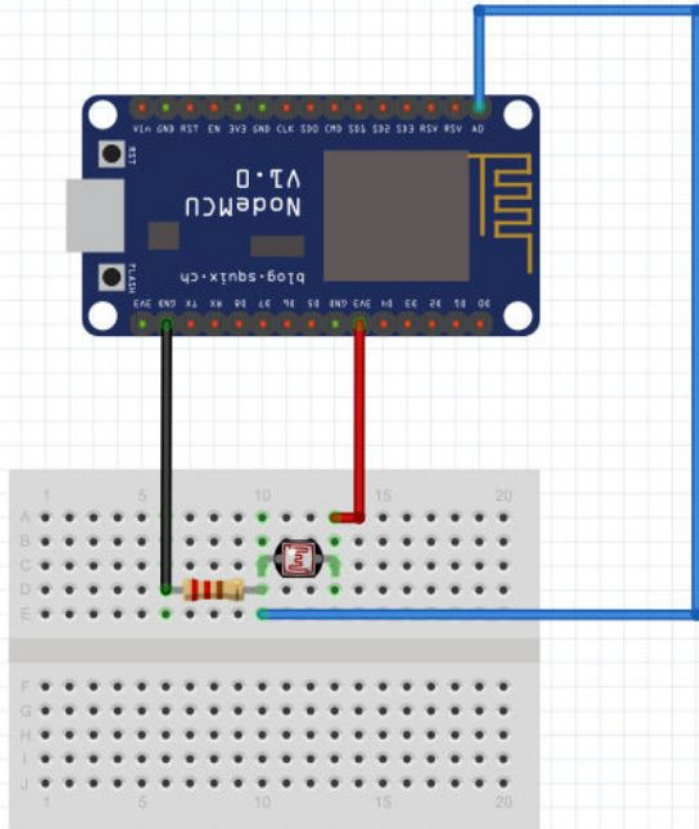
LightSensor.ino



Made with  Fritzing.org

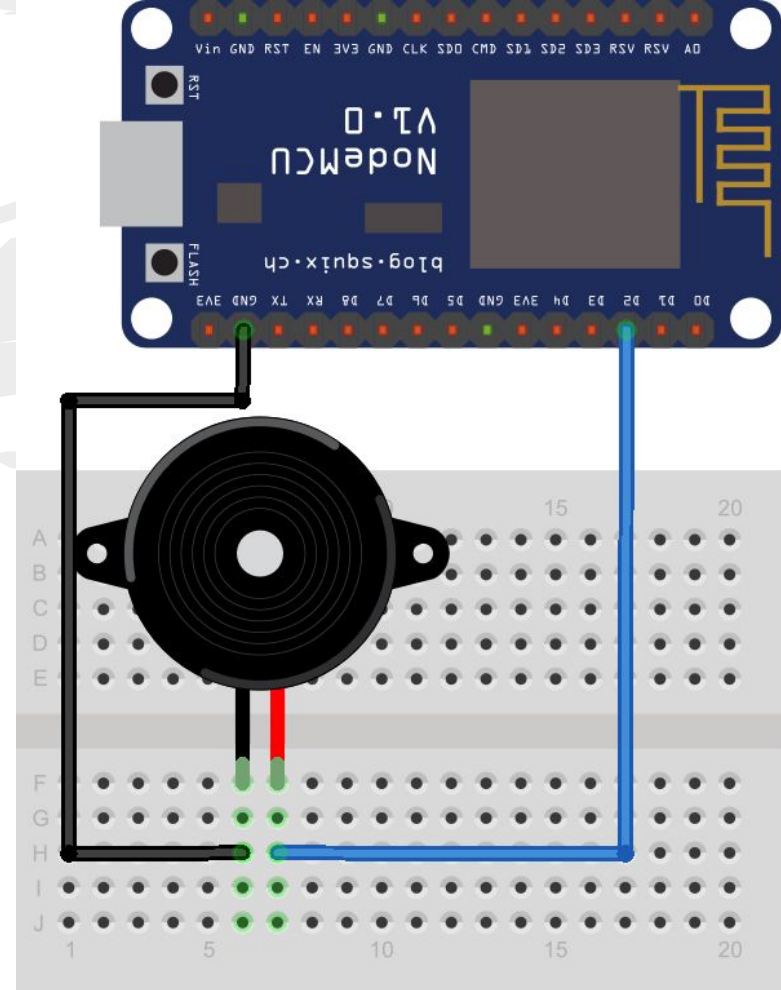


LightSensor_and_LED



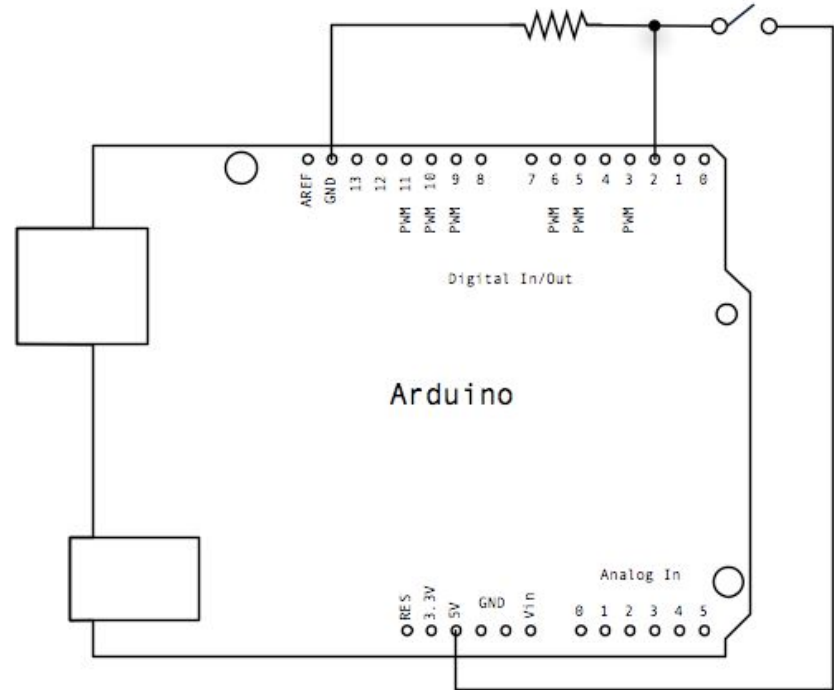
Play Tone

```
int audioPin = 4; //D2 on ESP8266
```

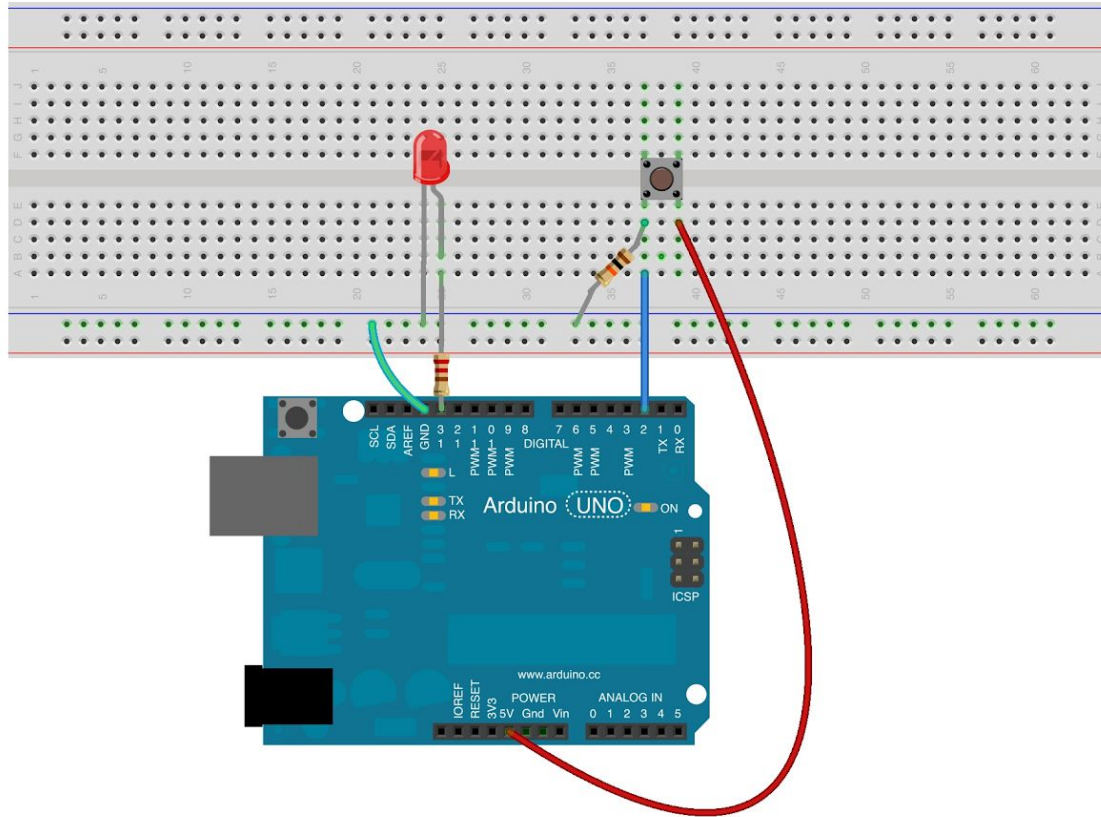


Arduino Example Sketch

File=>Examples=>02Digital=>
Button



File=>Examples=>02Digital=> Button



GEMMA

To upload to your Gemma

- 1) Select the proper board from the Tools->Board Menu
- 2) Select USBtinyISP from the Tools->Programmer
- 3) Plug in the Gemma/Trinket, make sure you see the green LED lit
- 4) For windows, install the USBtiny drivers
- 5) Press the button on the Gemma/Trinket - verify you see the red LED pulse. This means it is ready to receive data
- 6) Click the upload button above within 10 seconds

