"How to Arduino" for greater good!

Luke Geeson

# What is an arduino?

- Arduino is a <u>microcontroller</u>
- microcontroller:
  - micro = <u>small</u> (no implication of reduced capability)
  - controller = used to <u>measure</u> and <u>interact</u> with the physical world (physical computing)
- Has a microprocessor (is a small computer)
- written in arduino C (a superset of the C programming language)

Sensors 'sense', actuators 'act'

- 2 general types of component: things that <u>measure</u> the world and things that <u>change</u> it!
  - e.g LEDs illuminate (they are actuators)
  - e.g temperature sensors measure (sensors)
- the Arduino uses these, and some programming to make amazing things.

Electronics 101

- Electricity concerns the charge of energy in a given place
  - positive charge moves towards negative charge as a 'current' or 'flow of electrons'.
- There must be a 'complete circuit' for this to work
  - have a charge go to either ground or a negative terminal
- The charge is an 'analog' value and will vary, although we typically use discrete values (e.g 3.3V, 5V)
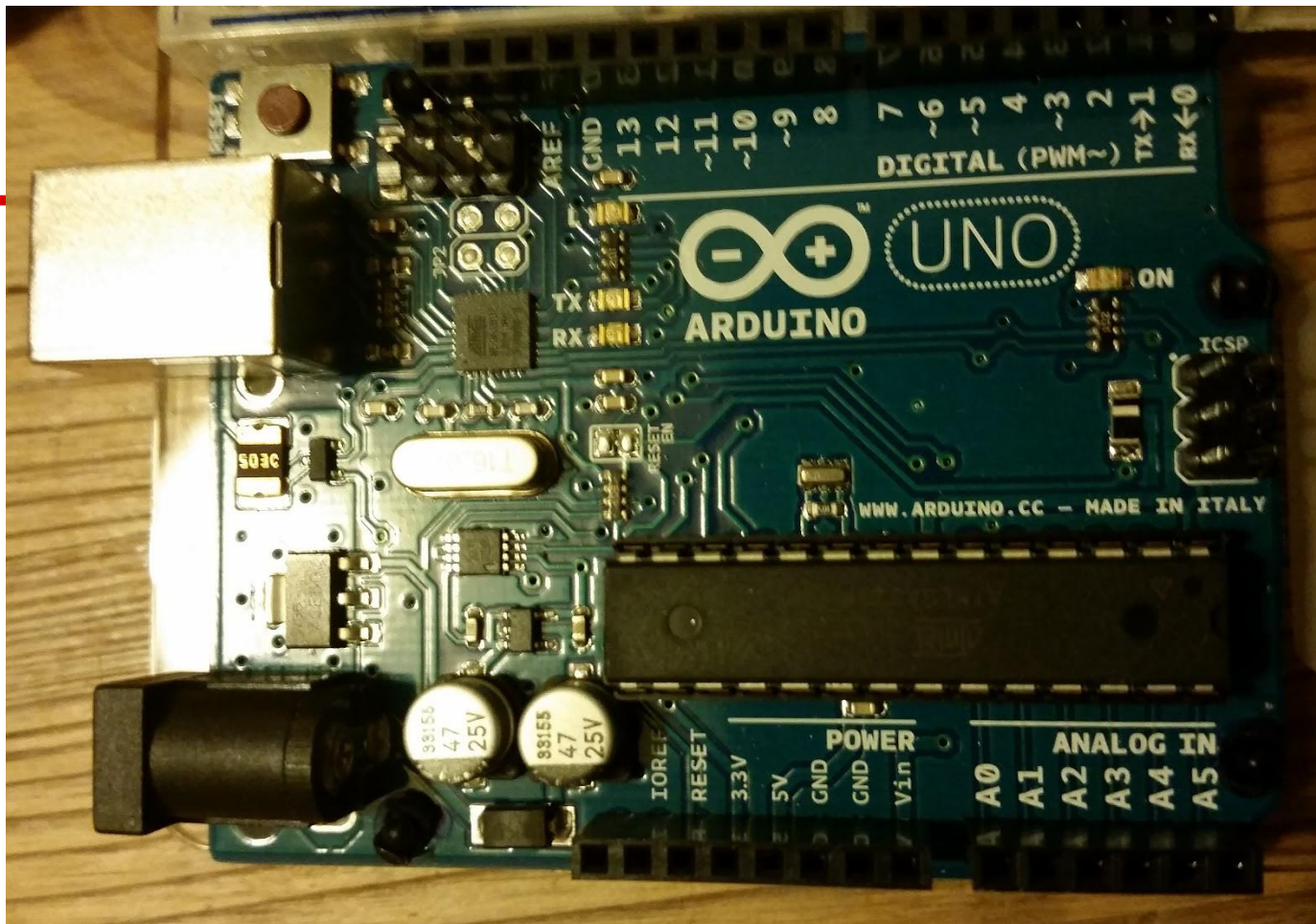
LEDS, Resistors, Jumpers and bread

- we will make a simple circuit and program to make an LED flash!
- we will use LEDS, and resistors in the circuit and some in-built functions (for the arduino) to create a program that will send current to arduino, delay for a period of time and then turn it off for a period of time.

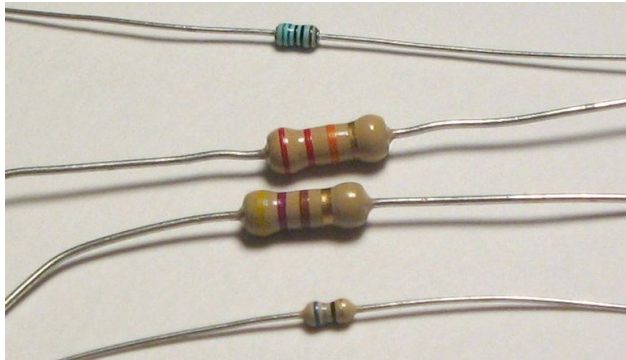# Light emitting diodes

- Diodes only conduct electricity in 1 direction.
- LEDs emit light when a current passes through them.
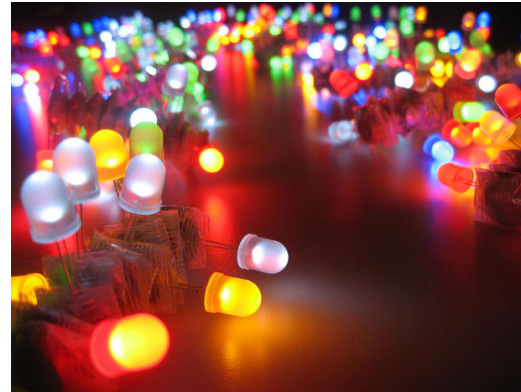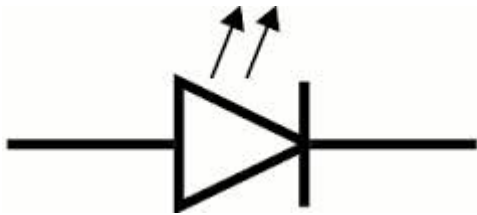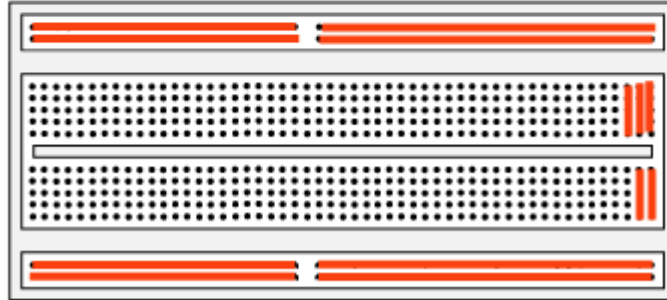
Resistor

- They will resist the flow of electrical current
- current flowing through a resistor will be proportional to the <u>value</u> of the resistor.
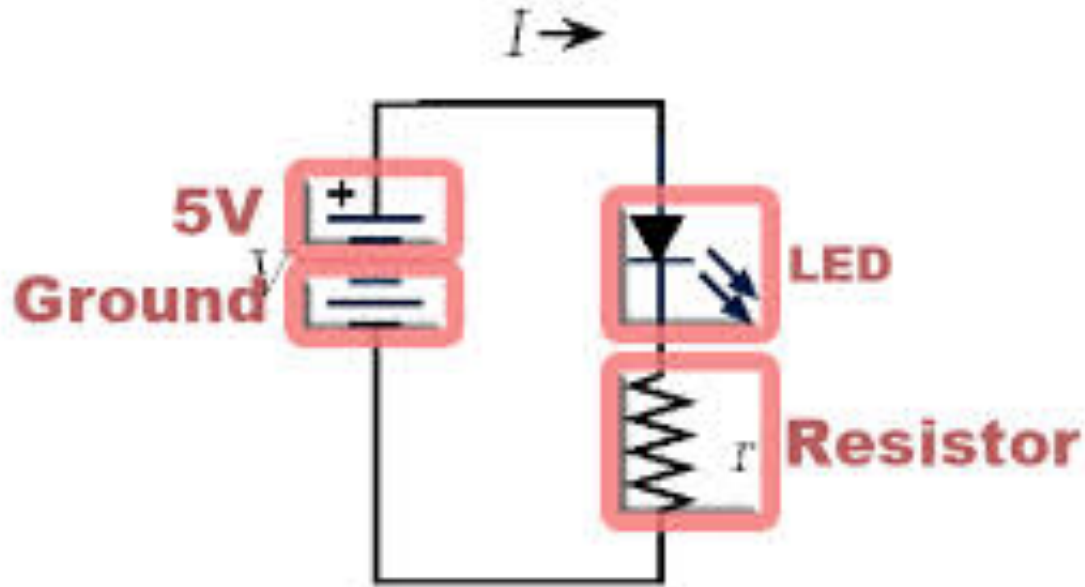
Breadboards!

Used to plug in components and test things.

-test it before you deploy!

Breadboard Connections
Shown in Orange

Let's make a thing!

let's do some programming

- The arduino is written in 'arduino C'
- we write programs called 'sketches' and load them onto the board
- every time the board is given power, it will try and run this program.

The Code

```
void setup(){
    pinMode(13, OUTPUT);
}
void loop(){
    digitalWrite(13, HIGH);
    delay(500);
    digitalWrite(13, LOW);
    delay(500);
}
```

>**Programming :: data types**

data types: int, char, String, boolean, byte, short, long, float, double

byte <= short <= int <= long <= float <= double

boolean = true OR false

String = "A string of characters such as this."

byte, short, int, long = Integers such as 4

float, double = floating point numbers 4.444….

>**Programming :: variables**

- A <u>variable</u> is simply a 'thing' used to <u>store</u> a <u>value</u>
- it has a type and a name (used to express what it is), just like in Maths!
- assign a value with the '=' operator

dataType name = value (must be same as type)

int x = 4;

>Programming :: functions

- A function is collective 'block' of code which does a certain thing (possibly given some input) and could possibly produce an output
- Maths function: f(x) = x + 1
- code:

```
int f(int x){
    x = x + 1;
    return x;
}
```

>**Programming :: functions 2**

- functions are used when a 'process' needs to be done over and over possibly with different given input
- functions may or may not have a 'return type' which may return a value (VOID if no data is returned)
- functions have a name and code is contained within {} brackets

returnType functionName([optional input]){...code}

void setup(){}   *void loop(){}*

int size(String s){}   *boolean isEmpty(char[] list){}*

Arduino's in built functions

void pinMode(int pinNum, int io)

void digitalWrite(int pinNum, int value)

int digitalRead(int pinNum)

void delay(int numMillis)

void analogWrite(int pinNum, int value)

int analogRead(int pinNum)

The previous example

```
void setup(){
    pinMode(13, OUTPUT);
}
void loop(){
    digitalWrite(13, HIGH);
    delay(500);
    digitalWrite(13, LOW);
    delay(500);
}
//add semi-colons after every statement!!!! it's the 'full-stop' of the coding world!
```

More useful functions from Arduino!

int map(int val, int minVal, int maxVal, int minRange, int maxRange)

//there's a terminal to see some outputs

void Serial.begin(int baudRate)

void Serial.print(value)

//There are many integer, String and maths functions already defined in Libraries

Libraries and the open source

- Arduino is 'open source'
- the blueprints for all the tech is <u>freely</u> and <u>publicly available</u> - you could make your own!
- many contributors <u>make</u> and <u>test</u> code so you don't have to! (external libraries)
- search the web for libraries that do what you want (it's likely to exist)
- Make cool stuff!

Arduino extensions (shields)

- Not only is there open-source code, but also hardware!
- add-ons are made by many companies that fit onto the connections of the arduino to both compliment and extend its functionality
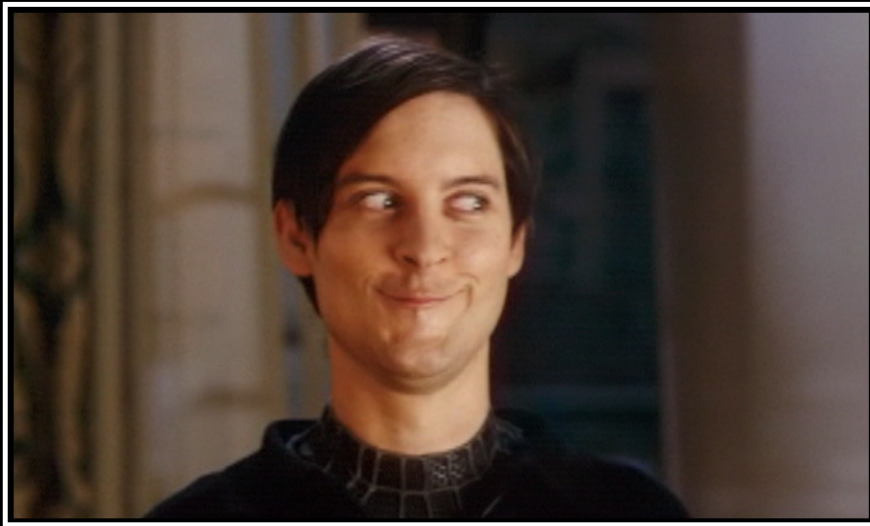 - The arduino platform is a very powerful platform as a result!

The End, and Questions of course :)

We have arduinos available to have a play with, come see me and we can try and do some <u>hacking!</u>

Thank you for your time

Additional exercise!

- get the Arduino IDE: http://www.arduino.cc/
- if you have a 'hacked' arduino: look at the data sheets for the pins: http://bit.ly/1xDrKNb
- try and implement the blinking light example! and then vary the frequency and pattern of its blinking

SPIDERMAN

He saw what you did.