Intro to Electronics

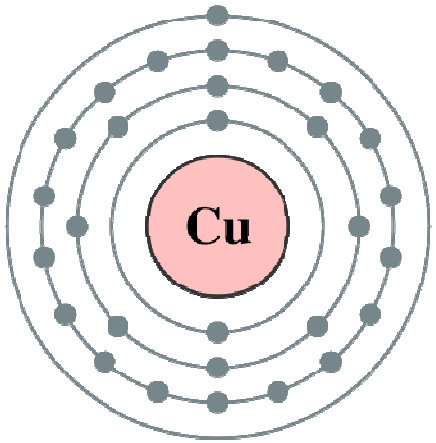
Electricity, Electronics, Transduction, Circuits, Ohm's Law
Definition of Components
Using a Multimeter
Reading schematics
Soldering

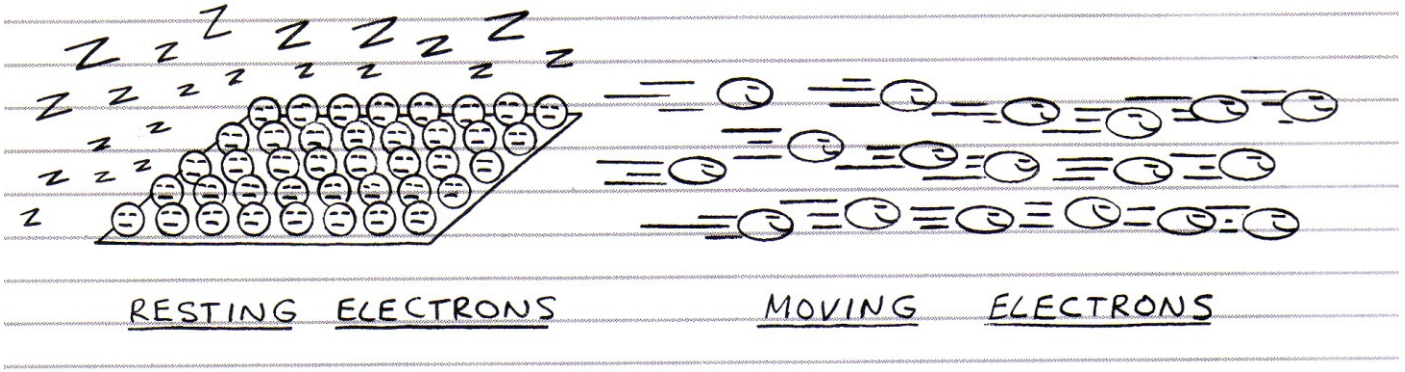
Electricity

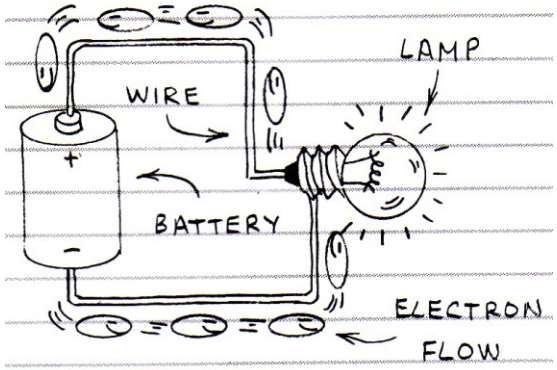
- Form of energy resulting from the existence of charged particles (such as electrons or protons), either statically as an accumulation of charge or dynamically as a current.
- Is the flow of electrical power or charge.

Electricity

29: Copper **2,8,18,1**

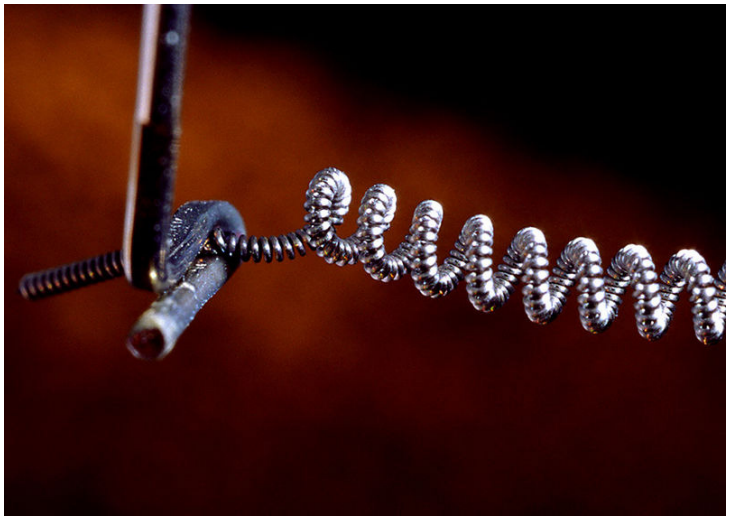


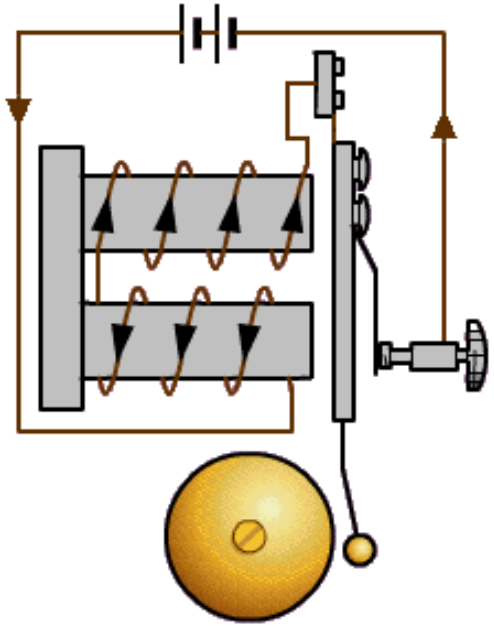




Electronics

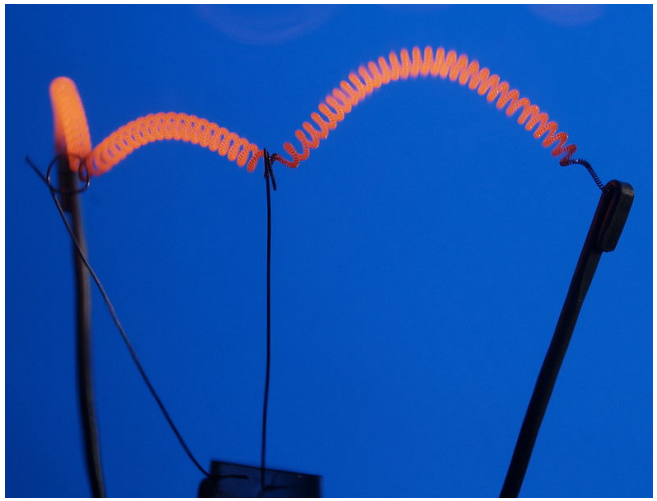
- Devices that work by controlling the flow of electrons



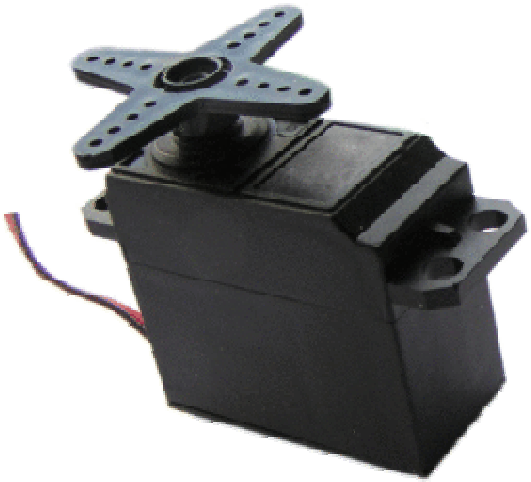


Transduction

- The conversion of one type of energy into another.



Electric to Light

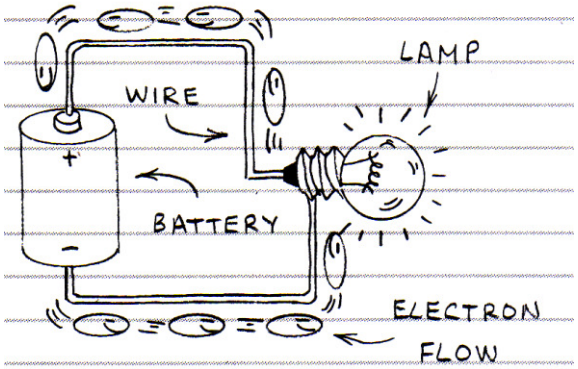


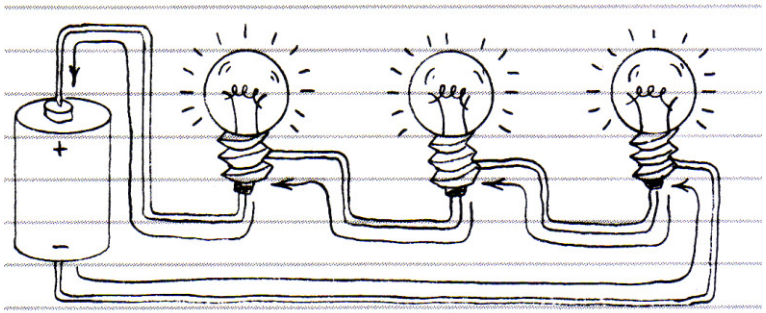
Electric to Rotational

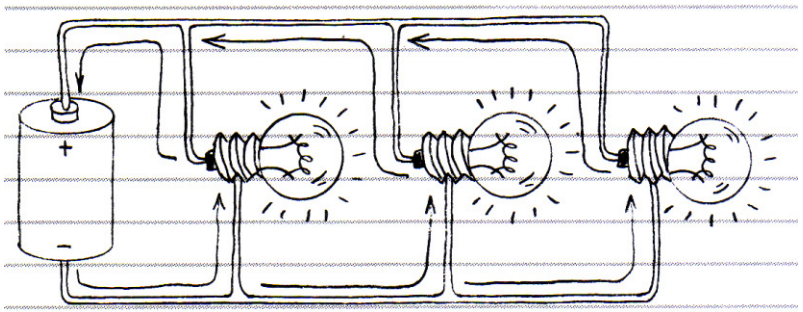
Electric to Heat

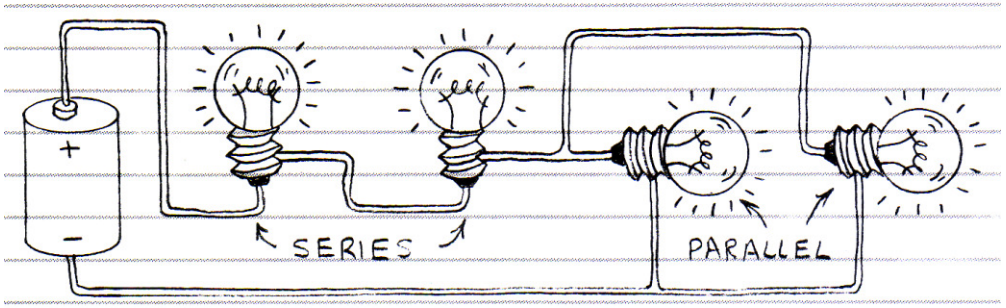
Circuits

A closed path formed by interaction of electrical components through which the electric current can flow.

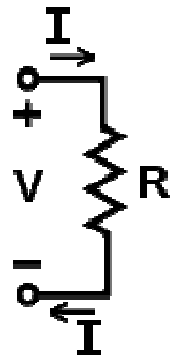








Key Terms



Voltage (V):

Potential difference in circuit

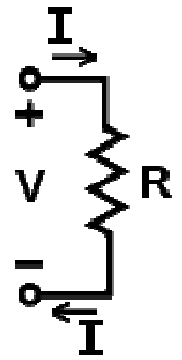
Unit: Volt (V)

Resistance (R):

a measure of the degree to which an object opposes an electric current through it.

Unit: OHM (Ω)

Key Terms



Current (I):

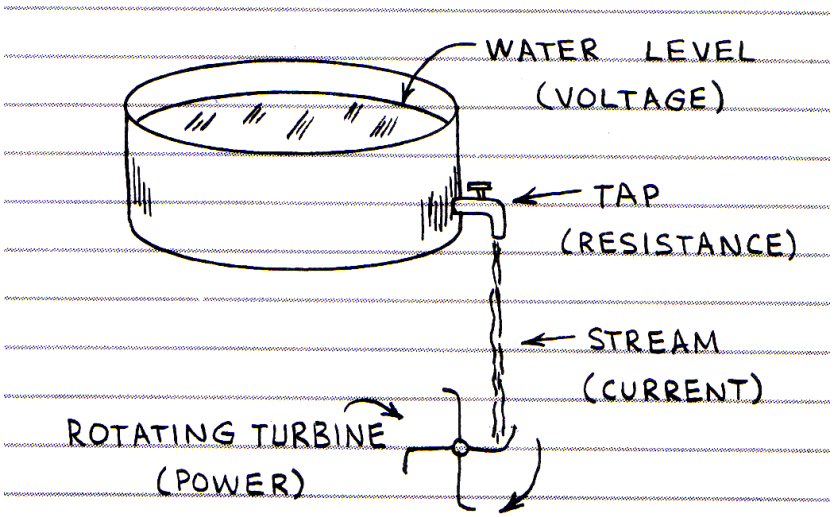
Quantity of electrons passing a given point.

Unit : Ampere (A)

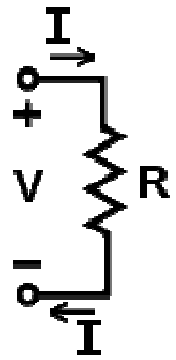
Power (P):

Work done by electrical current.

Unit : Watt (W)



Ohm's Law



- For a given resistance, voltage across the two points is directly proportional to the current between them.

$$V = IR$$

$$I = V/R$$

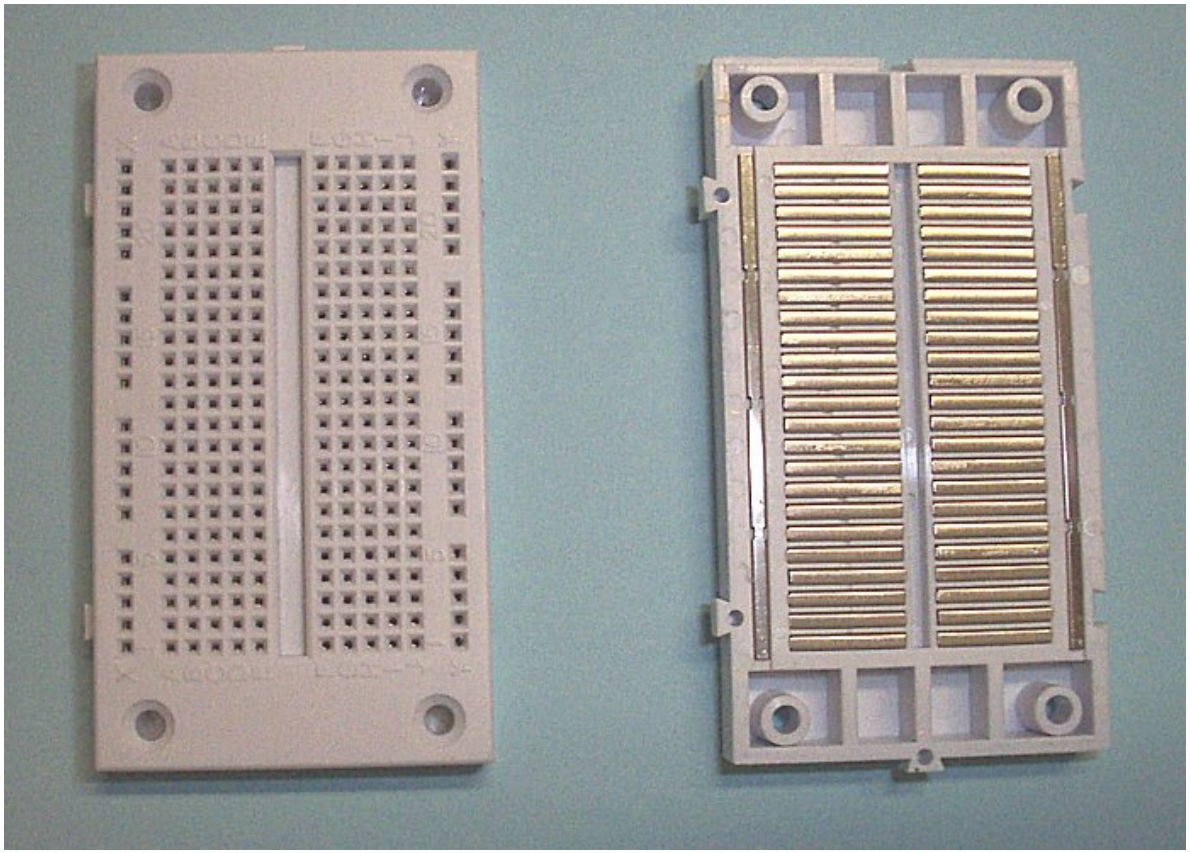
$$R = V/I$$

Definition of Components

Connectors:

Hook-up Wires

Alligator Clips



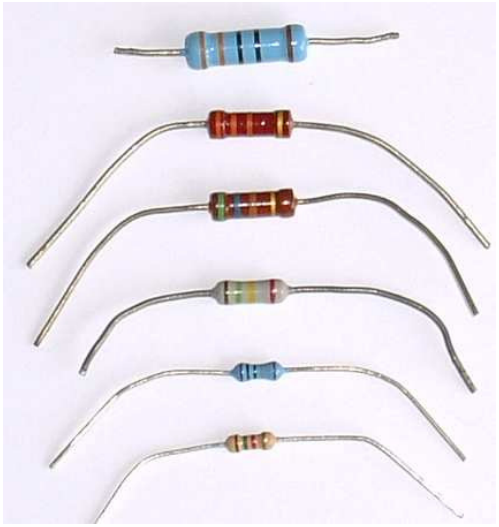
Breadboard

Definition of Components

Resistors:

Color code:

Black	0
Brown	1
Red	2
Orange	3
Yellow	4
Green	5
Blue	6
Voilet	7
Gray	8
White	9
Gold	5%
Silver	10%



BB ROY of **G**reat **B**ritain had a **V**ery **G**ood **W**ife

http://www.dannyg.com/examples/res2/resistor.htm

http://itp.nyu.edu/physcomp/Labs/Components

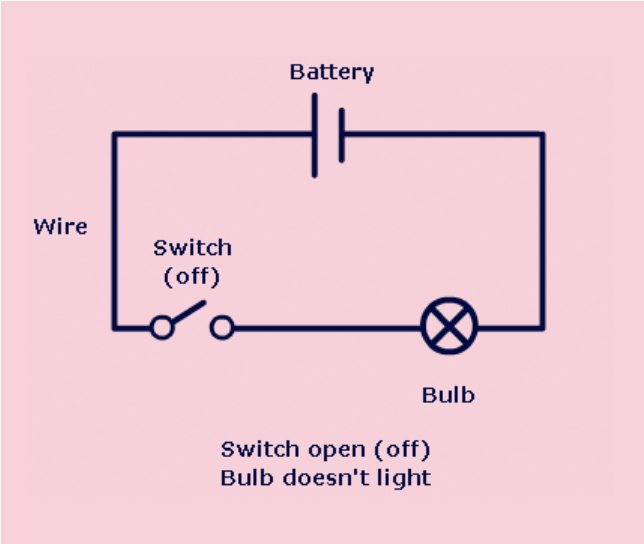
Multimeter

- Also called Volt/OHM Meter (VOM)

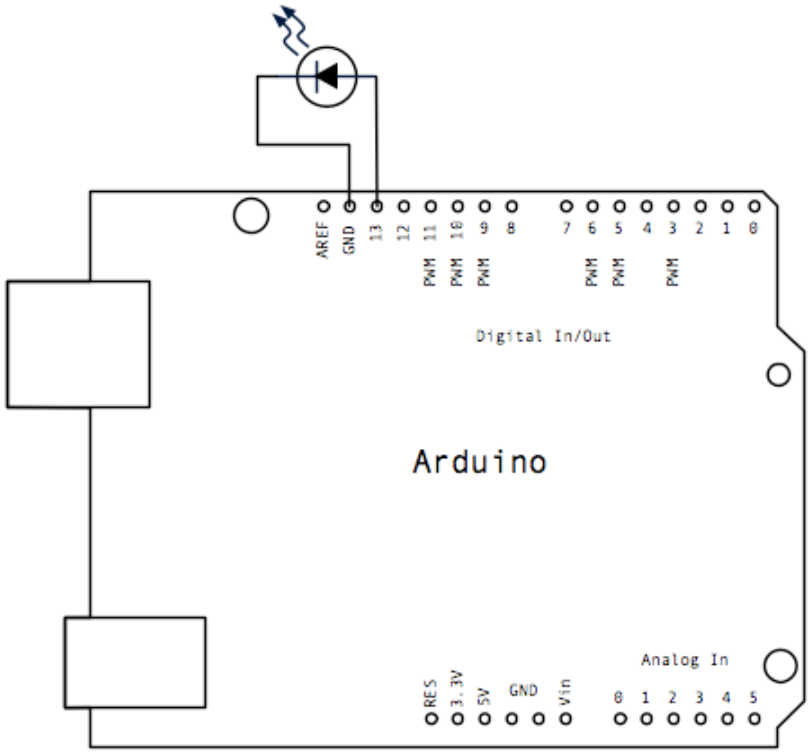
- Ability to measure Voltage, Current and Resistance

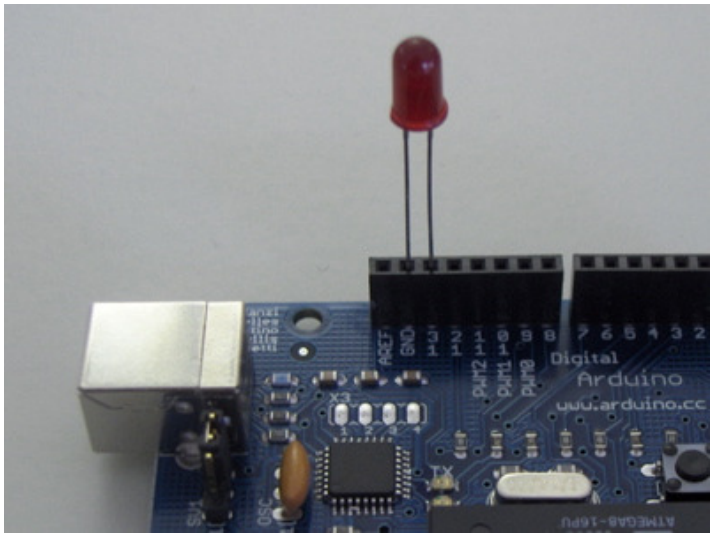
Reading Schematics

- A schematic is a diagram representing a circuit in which:
  Symbols represent components,
  Lines represent wires.



Schematics





http://www.rapidtables.com/electric/electrical_symbols.htm

Soldering

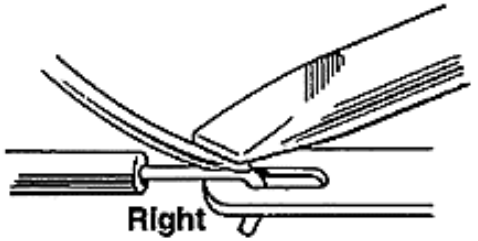
TIPS:

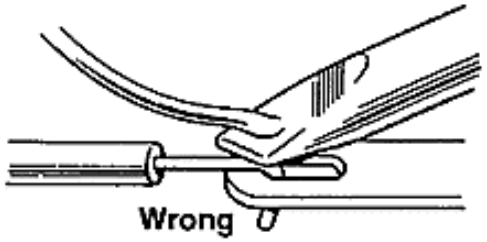
- Never touch the tip of solder iron.
- Always return the iron to the stand once soldering is done.
- Well-ventilated area.
- Wash your hands after using solder.

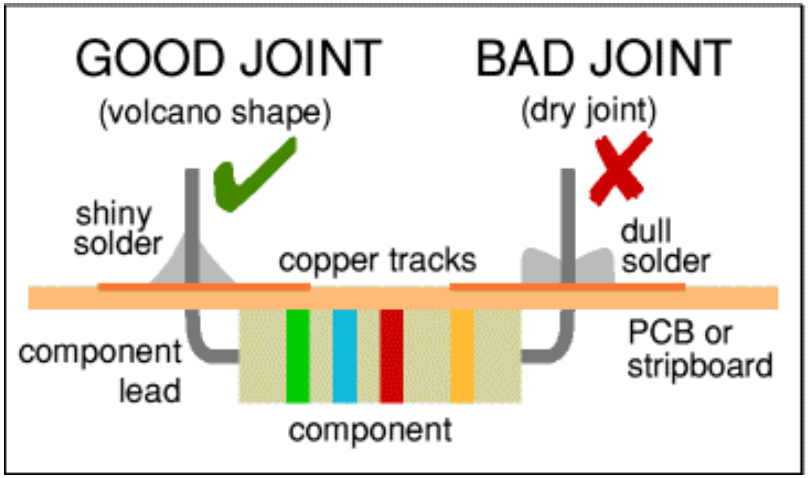
Soldering

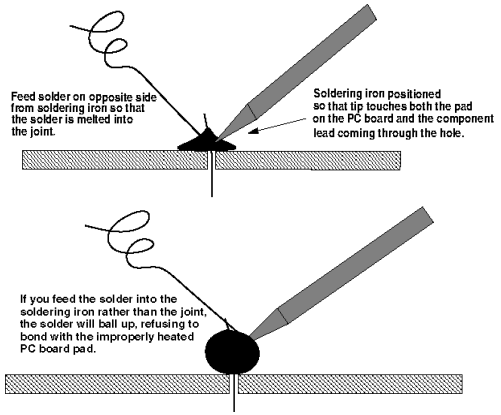
Always feed solder to joint and Solder iron to solder.

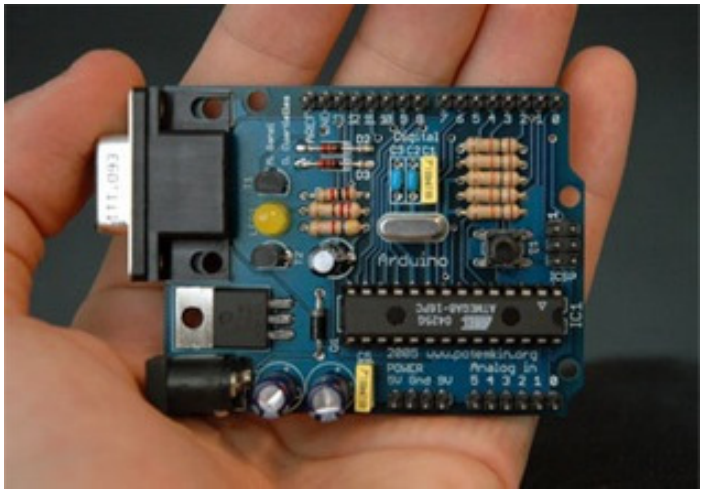
Good solders are quick joints.











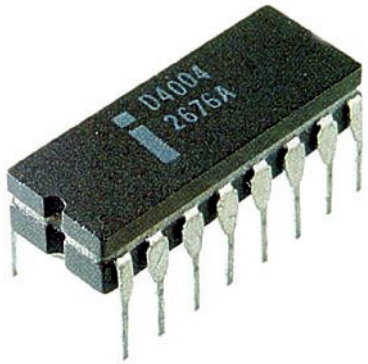
Digital In/Out

Microcontroller

Arduino

Lab

Micro-processor

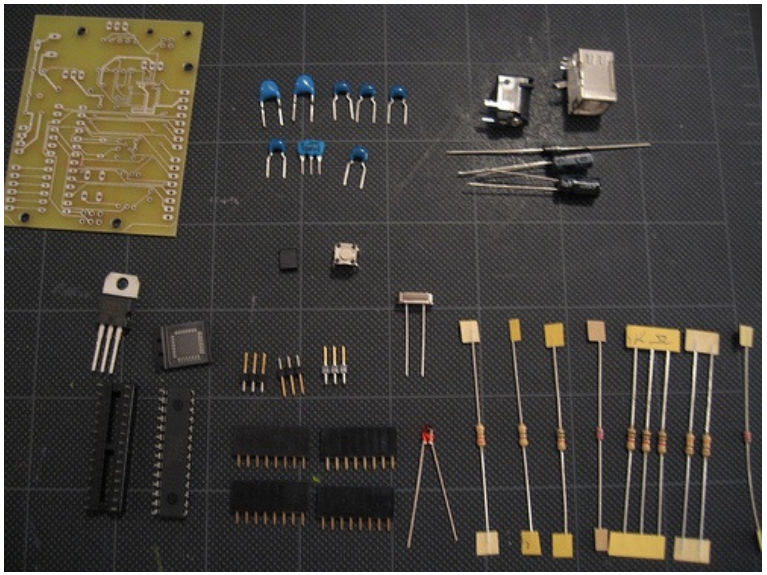


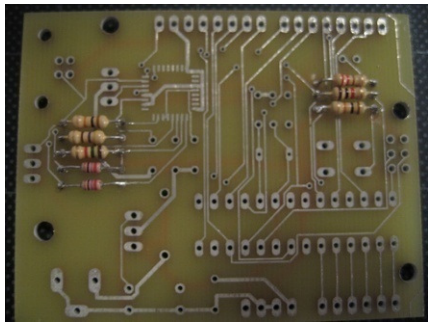
- incorporates most or all of the functions of a central processing unit (CPU) on a single integrated circuit (IC).

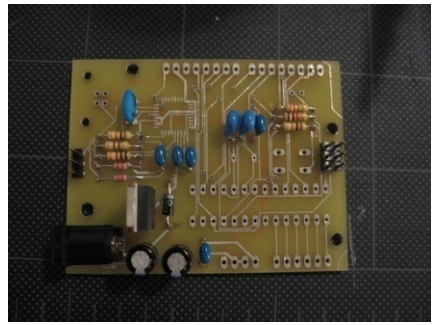
Micro-controller

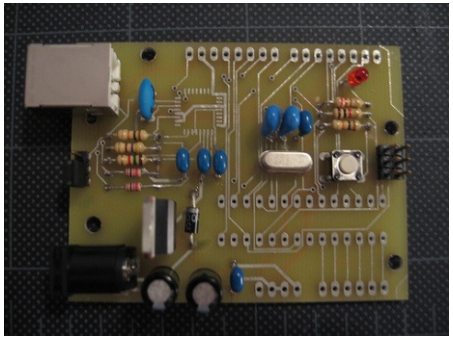
- Consists of microprocessor combined with support functions such as a crystal oscillator, timers, watchdog, small amount of RAM, serial and analog I/O etc.
- Microcontrollers are designed for small or dedicated applications.

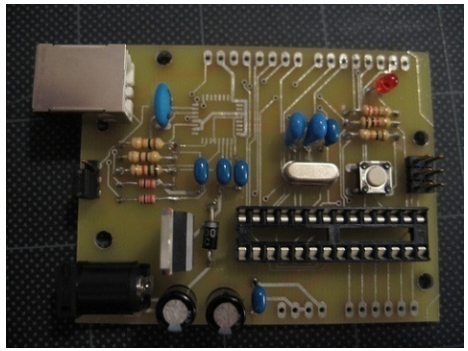
Microcontroller

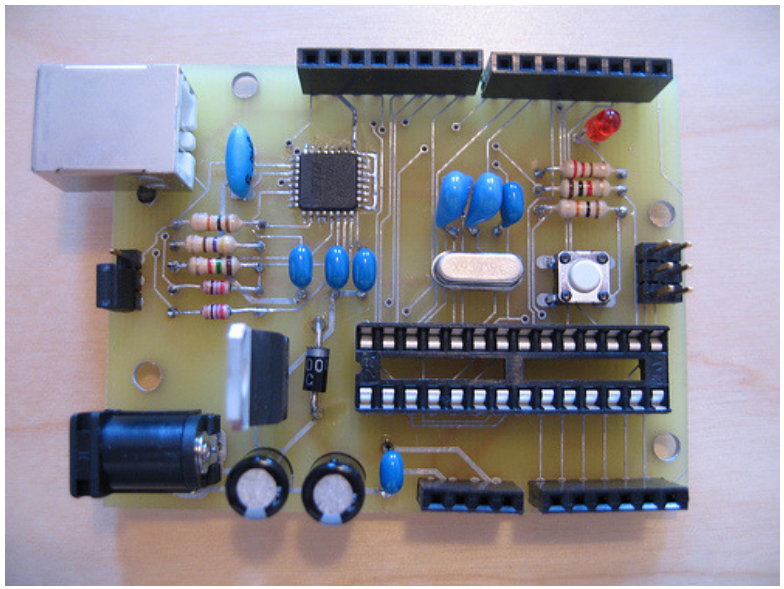










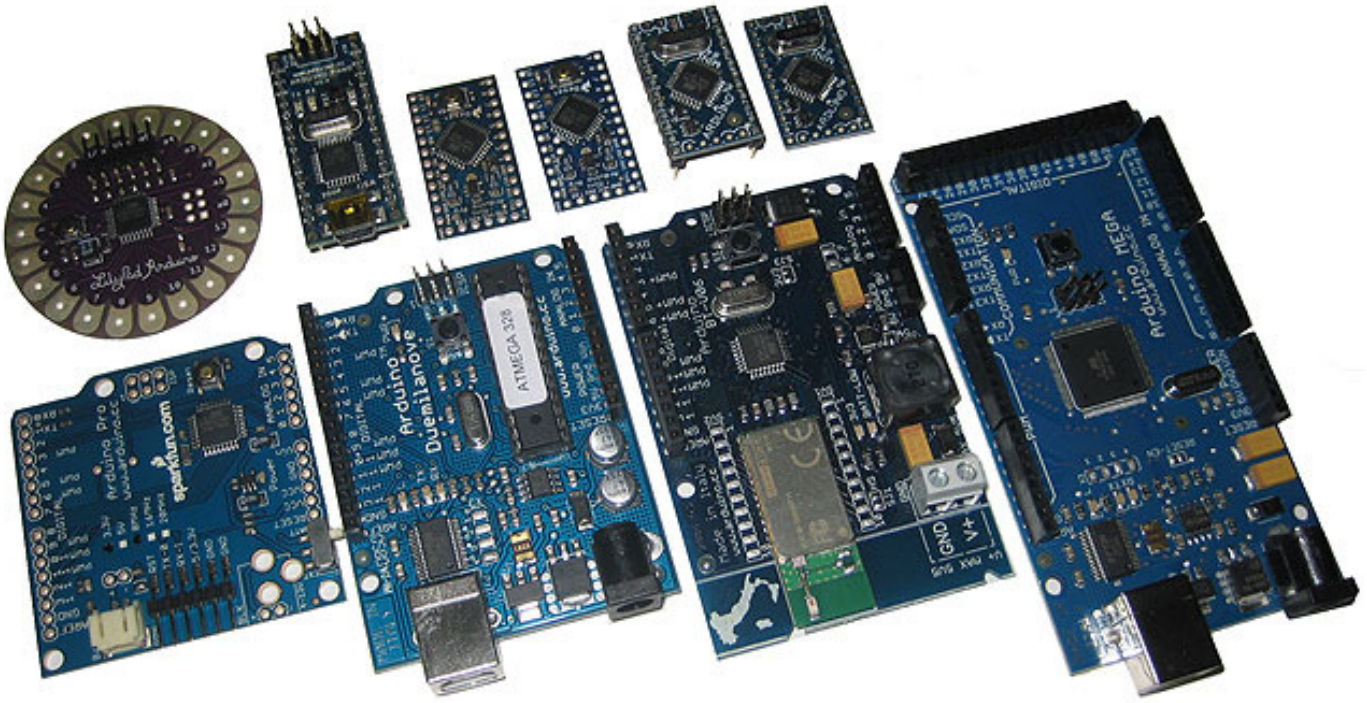


Arduino

- Is a tool for making computers that can sense and control more of the physical world than your desktop computer.
- Can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs.

Arduino Types

- Manufactured differently which serve different chip size / needs.



Mini light

Pro Mini

Nano

Lylipad

Pro

Diecimila,

Duemilanove

Bluetooth

Mega

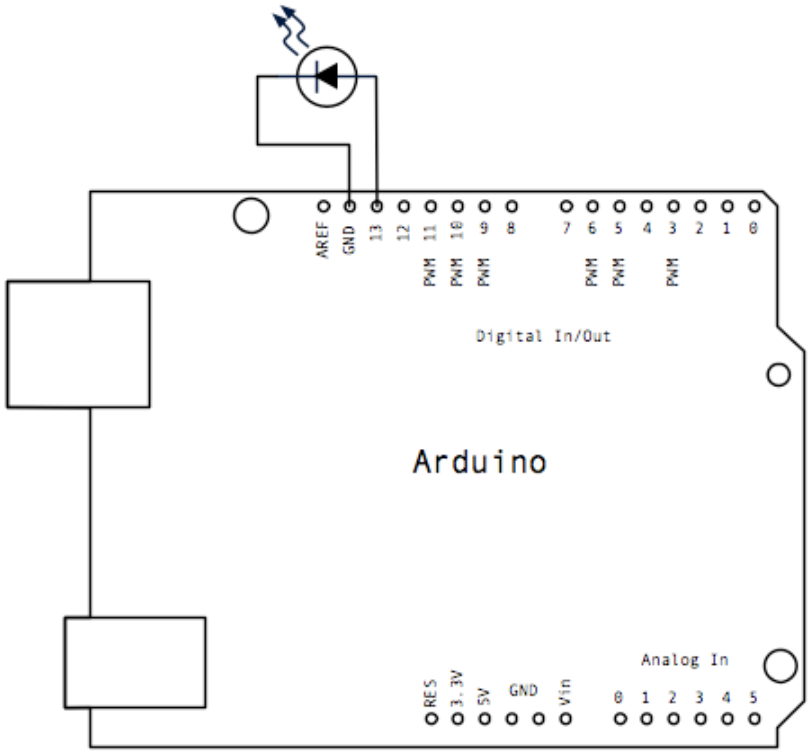
Processor Programming

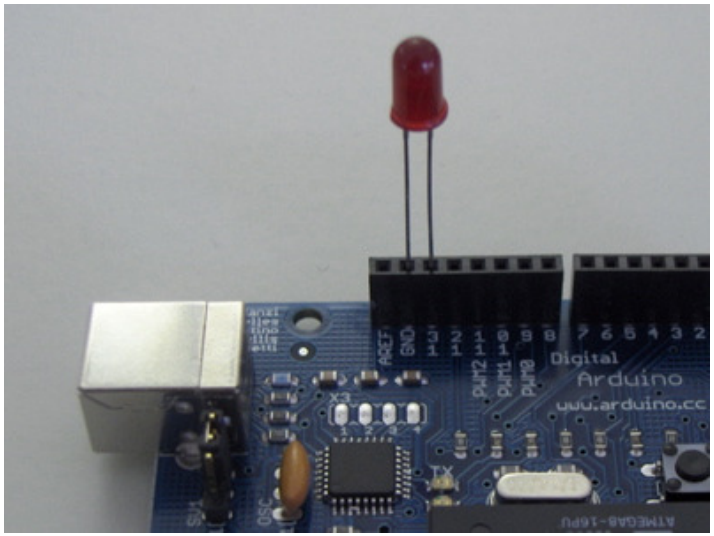
Programming a processor is basically to assign new functions to the microprocessor.

Arduino uses C coding to program its microprocessor.

Arduino developed its own platform (software) to code its processor. www.arduino.cc

Blink : Hardware

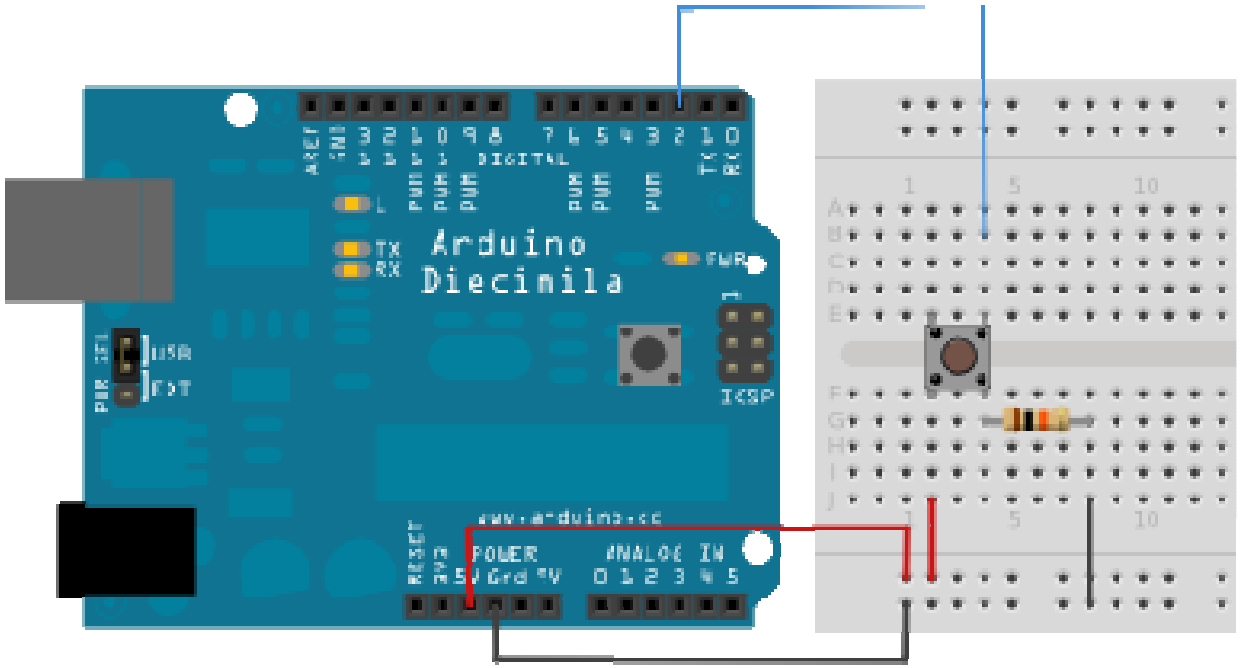


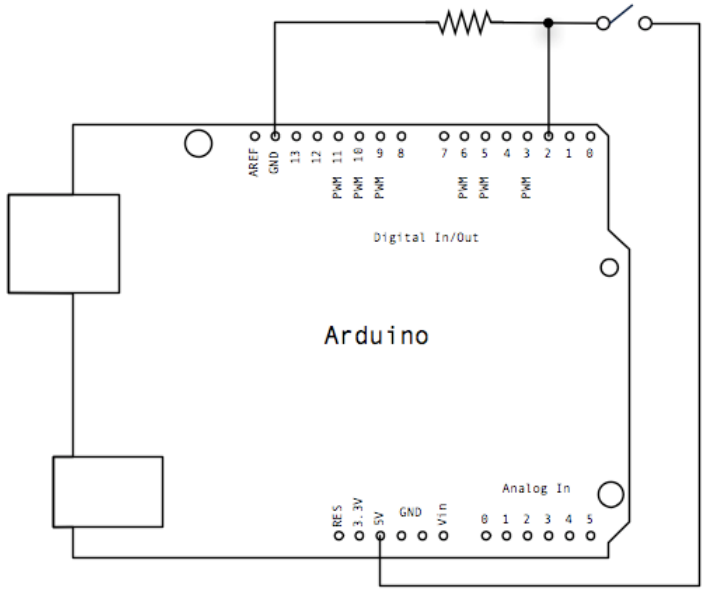


Blink : Code

```
int ledPin =  13;    // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup()   {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

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

Arduino : Button





Code

```
// constants won't change. They're used here to
// set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

void loop(){
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed.
  // if it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin, HIGH);
  }
  else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}
```