https://www.halvorsen.blog

|Programming with Arduino|

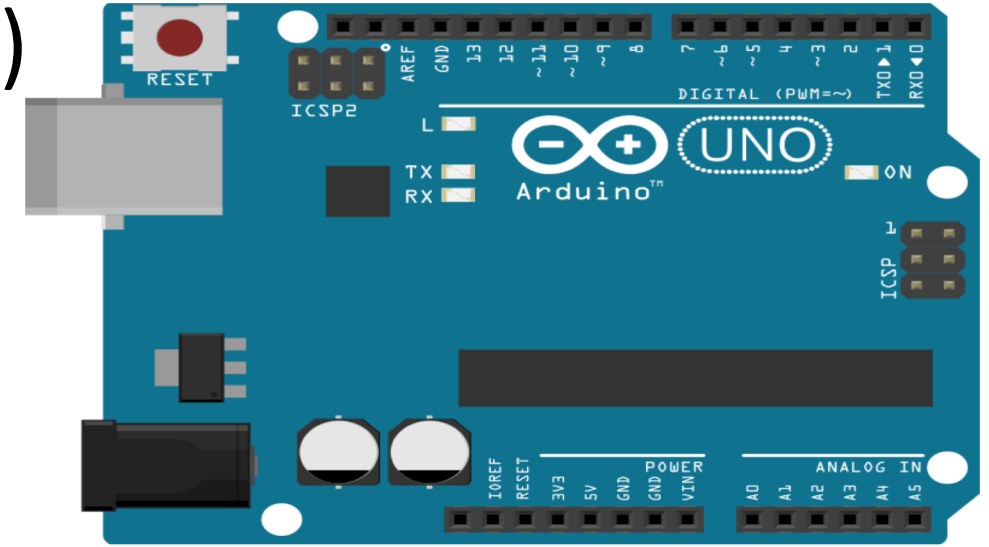
|Hans-Petter Halvorsen |

Table of Contents

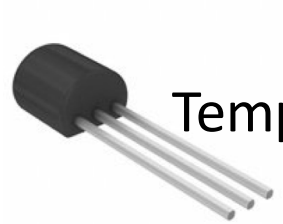
What do you need?

To get started you need the following:

- PC (Windows, Mac, Linux)
- **Arduino UNO** (~200 NOK)
  or a Starter Kit (~800 NOK)
- Software (free)
- Electrical components
  (wires, resistors, etc.)

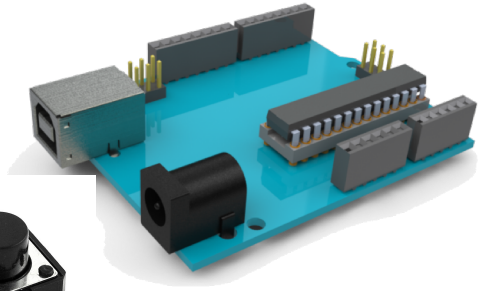


Equipment

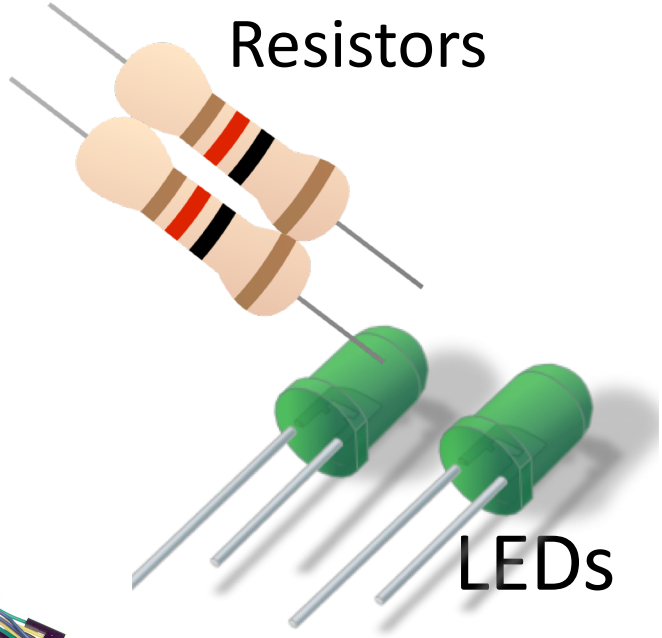


Temperature sensor

Switch



Arduino

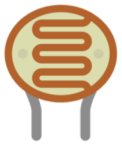


Resistors

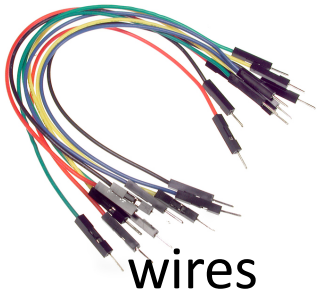
LEDs

Potentiometer

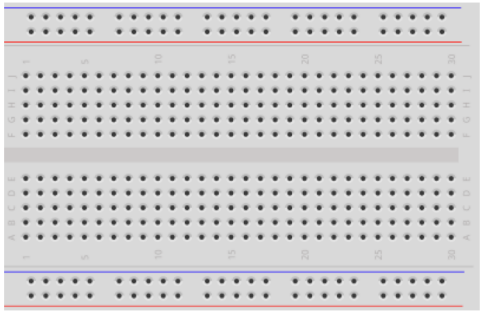
Light sensor



USB cable



wires



Breadboard

Thermistor

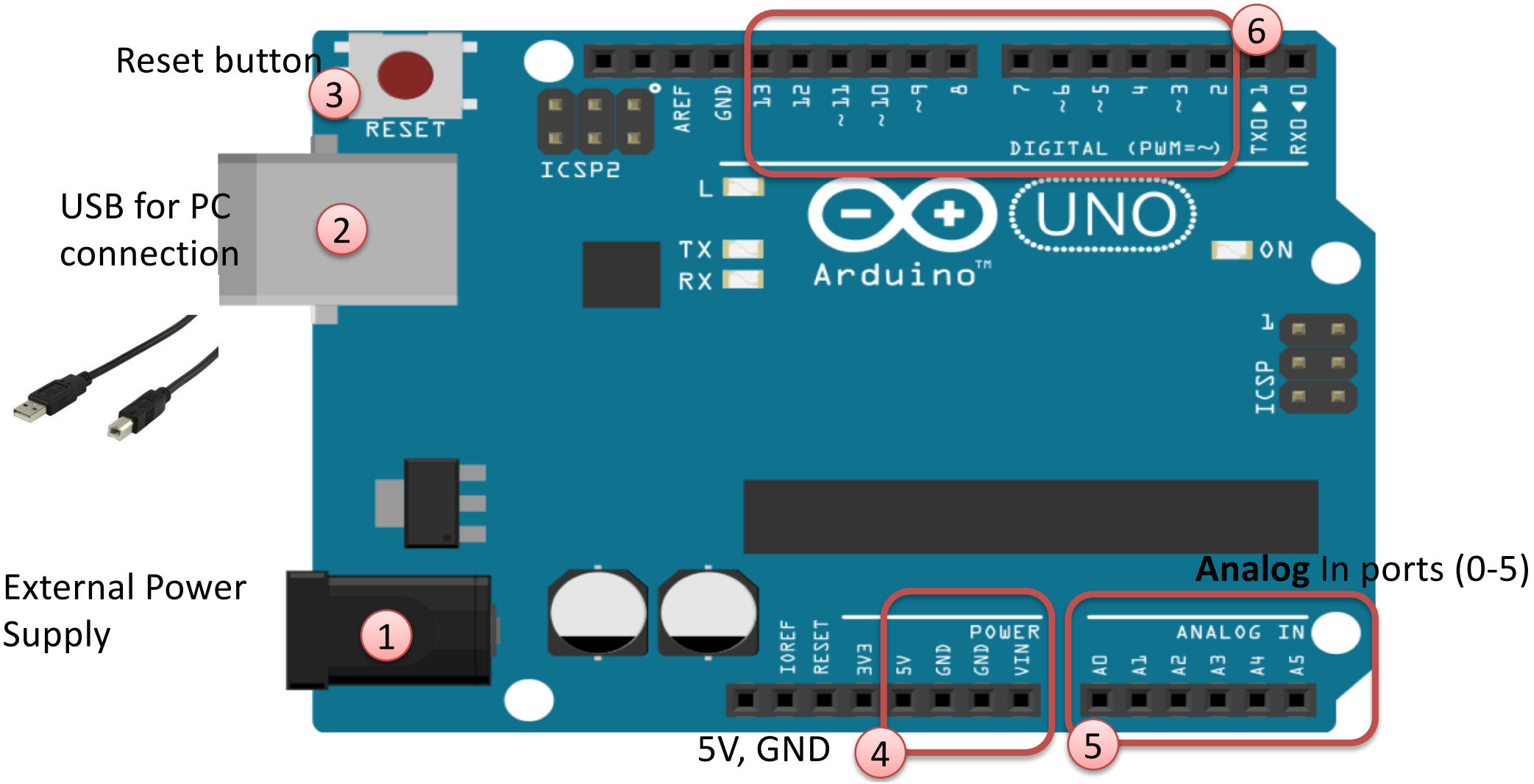
Multimeter

https://www.halvorsen.blog

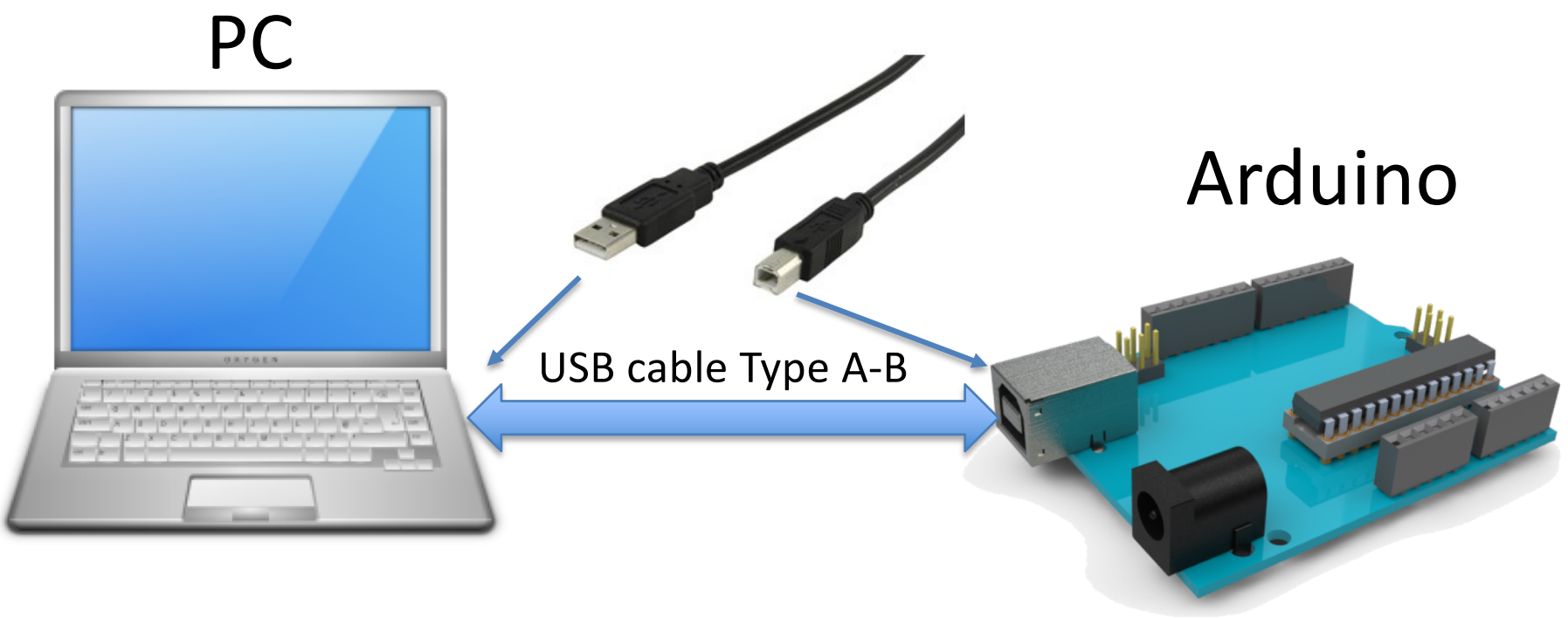
Introduction to Arduino

|Hans-Petter Halvorsen |

Arduino UNO Overview



Connect your Arduino to your PC



Play and Explore

|Programming with Arduino|

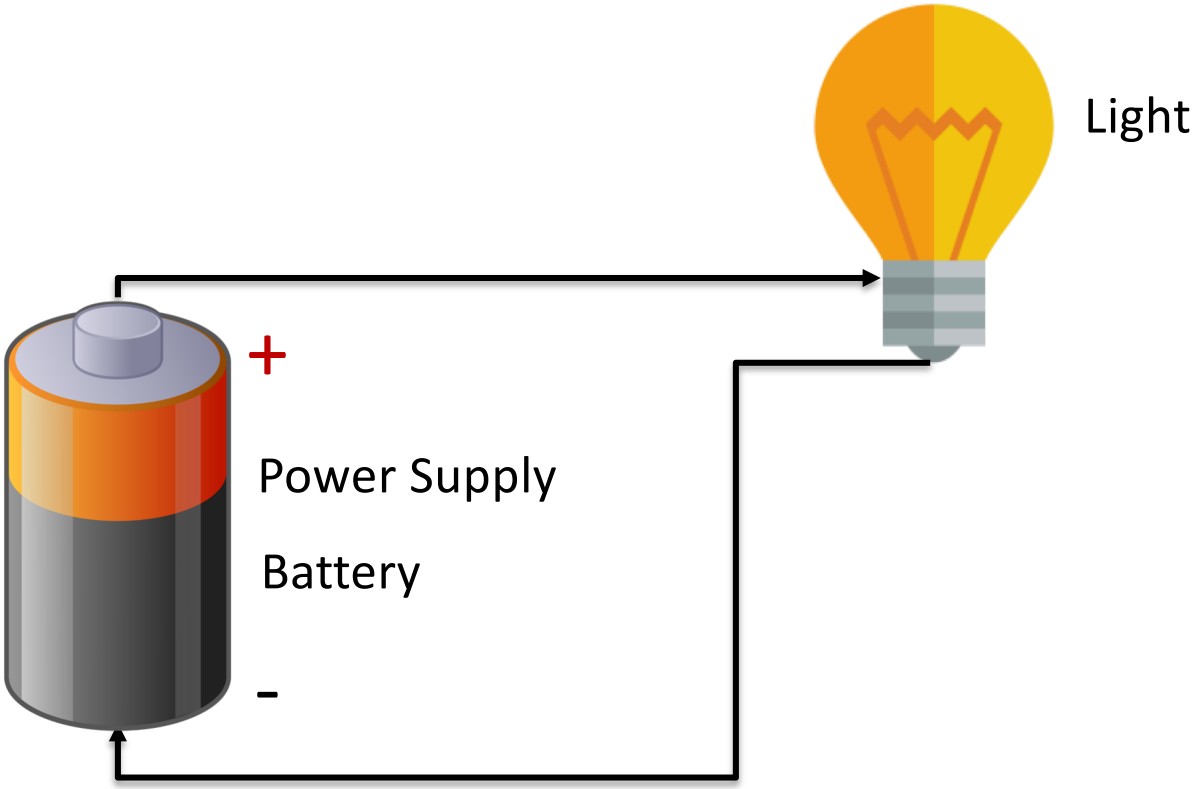
https://www.halvorsen.blog

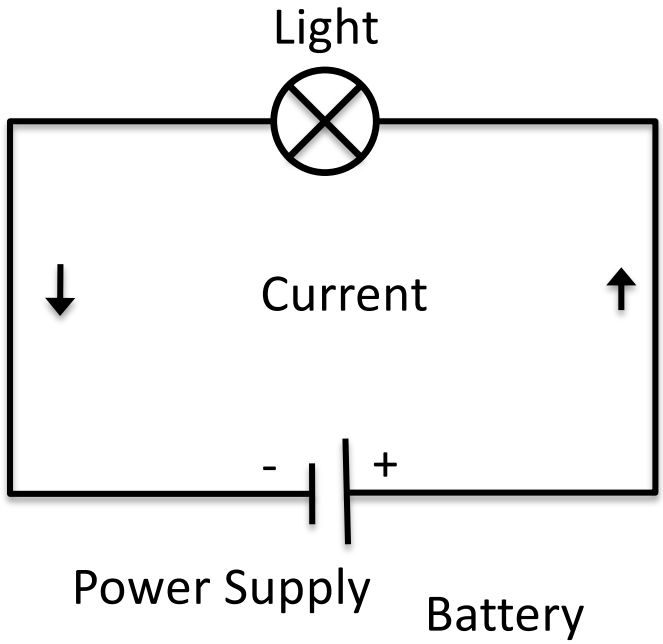
Electronics Foundation

|Hans-Petter Halvorsen |

Electrical Circuit

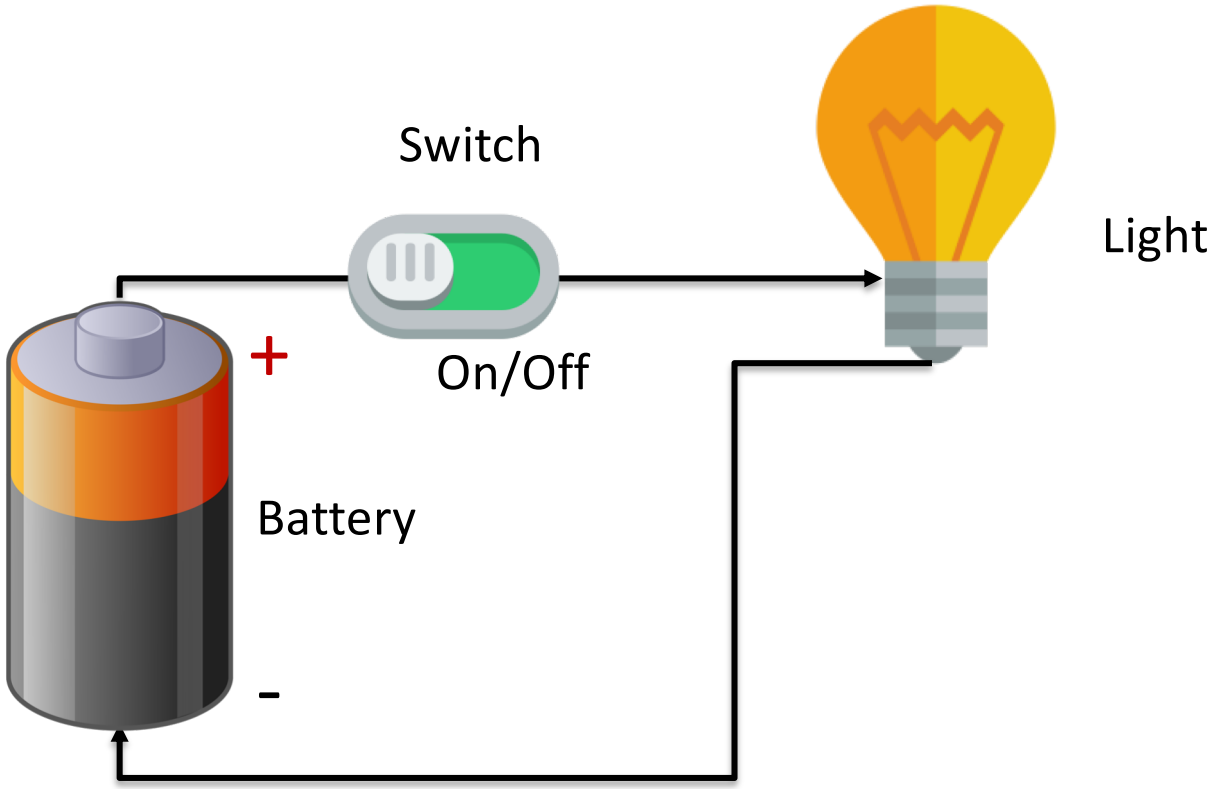
Here you see a basic Electrical Circuit:

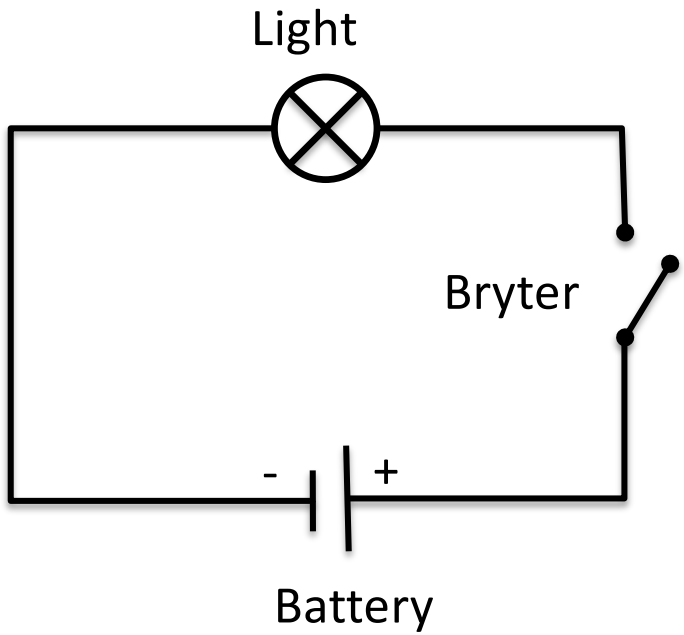




Electrical Circuit with a Switch

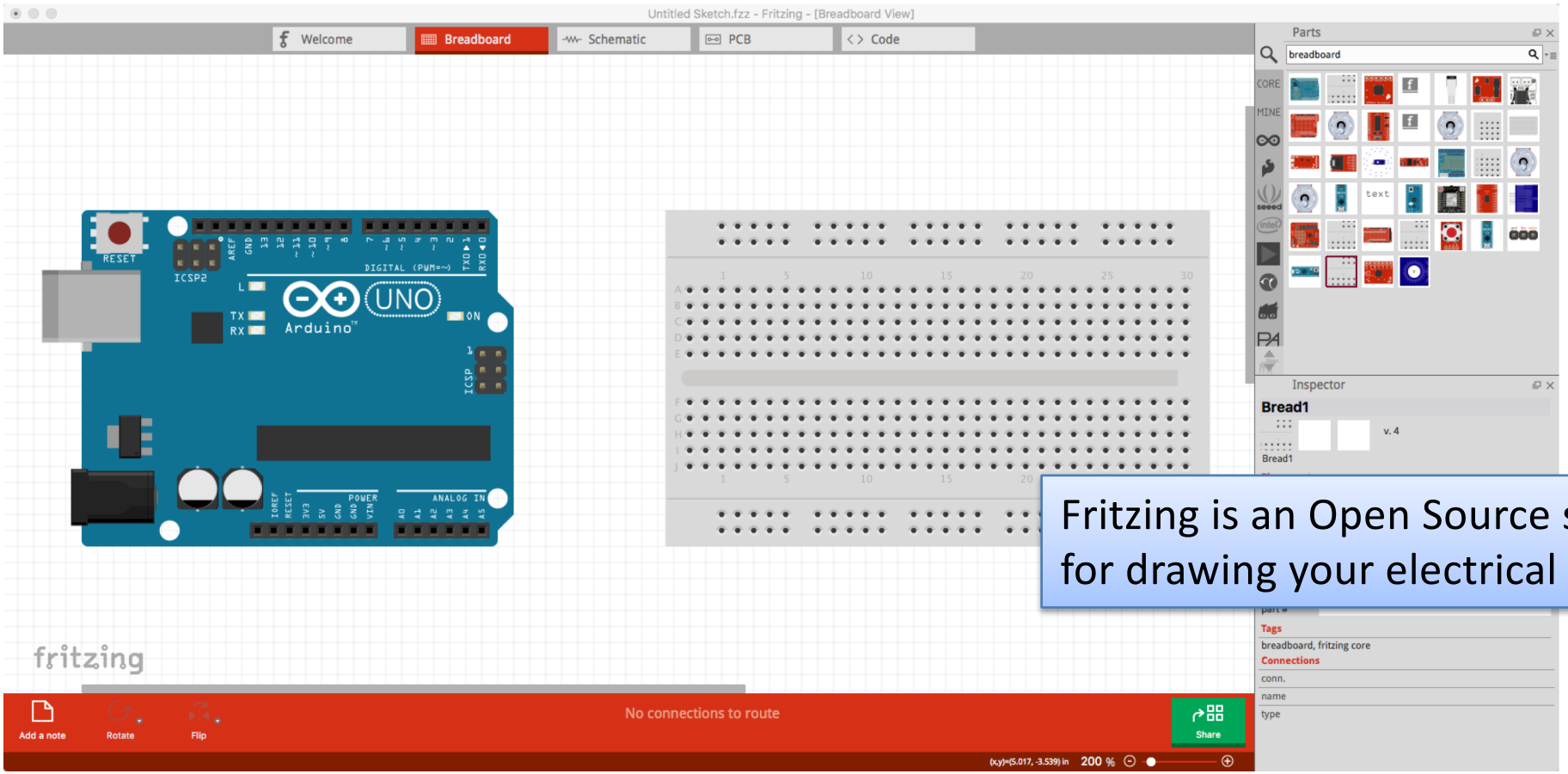
Here you see a basic Electrical Circuit with a Switch:





fritzing

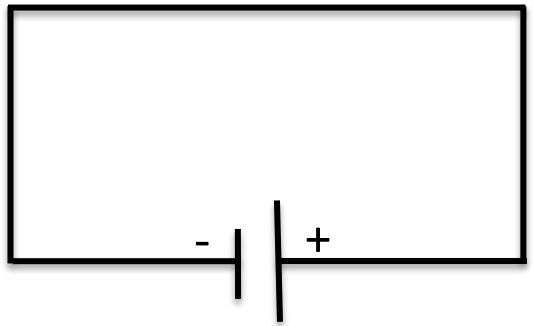
[http://www.fritzing.org]



Short Circuit

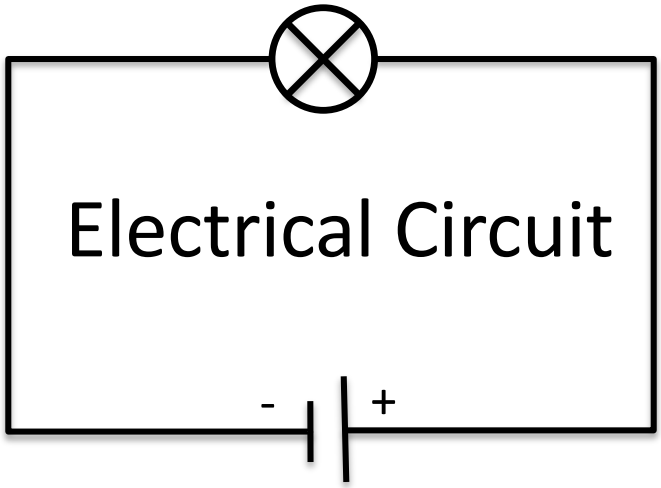
- We must never connect positive and negative side to a power source without having an electrical component in between.
- If you do, it is called a short circuit.
- For example, if you short circuit a battery, the battery will get very hot and the battery will run out very quickly.
- Some batteries may also start to burn.
- When it starts to smoke from electrical components, it happens because it has become too hot.
- In most cases, it means that the component is broken.

Short Circuit!!



Power Supply

Ohms Law



This is Ohms Law:

$$U = RI$$

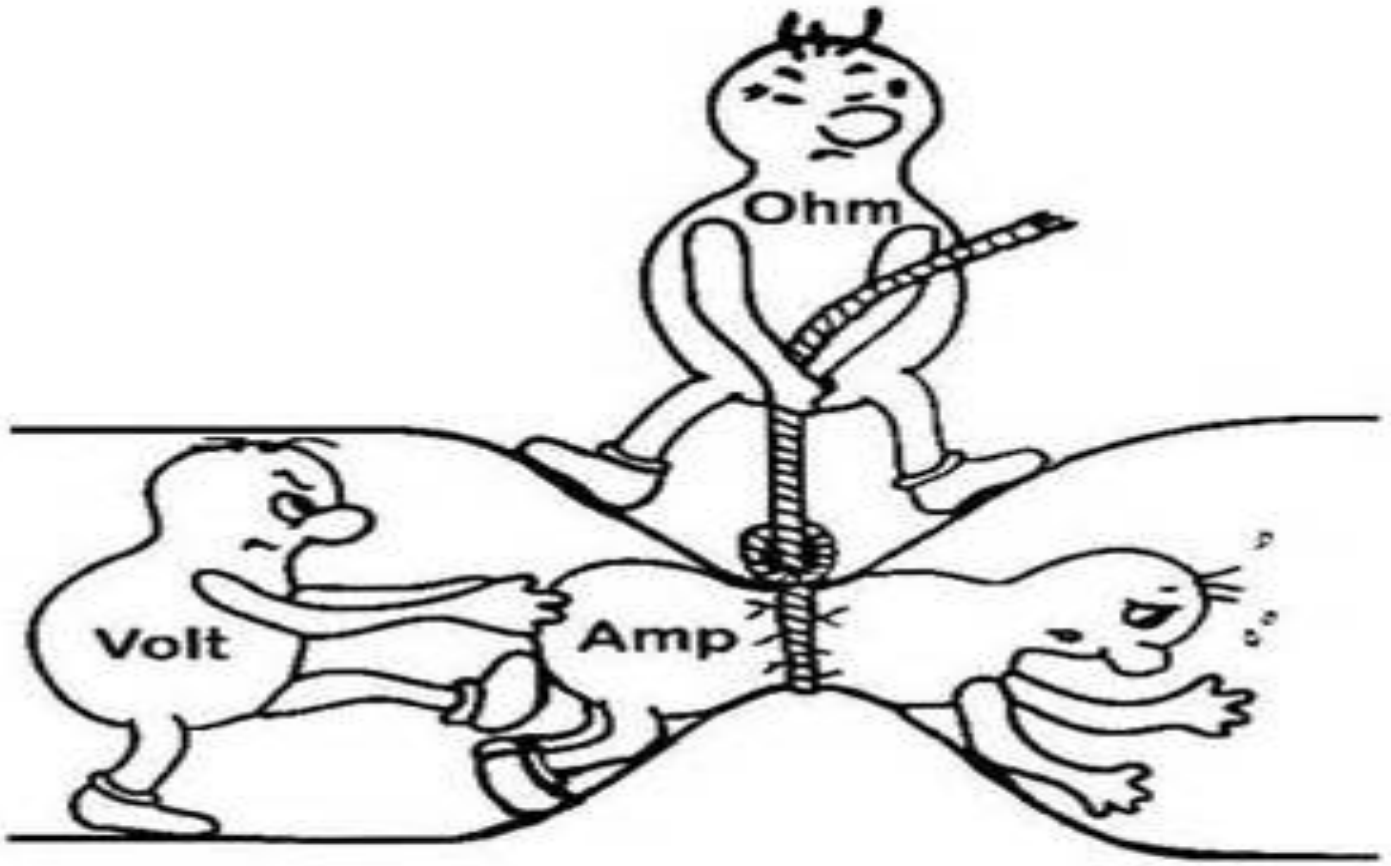
U – Voltage $[V]$
R – Resistance $[\Omega]$
I – Current [A]

$$R = \frac{U}{I}$$

$$I = \frac{U}{R}$$

Ohms Law

$$U = RI$$

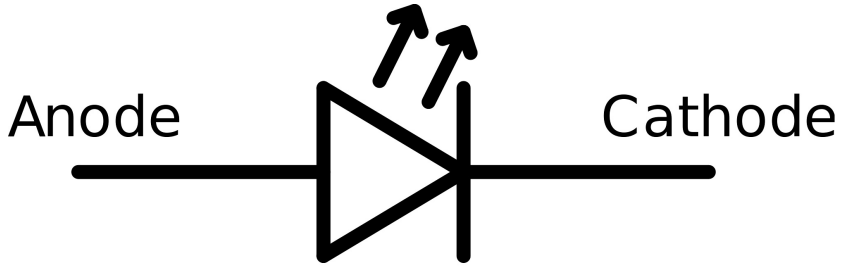


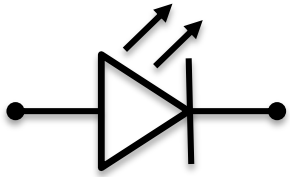
Multimeter

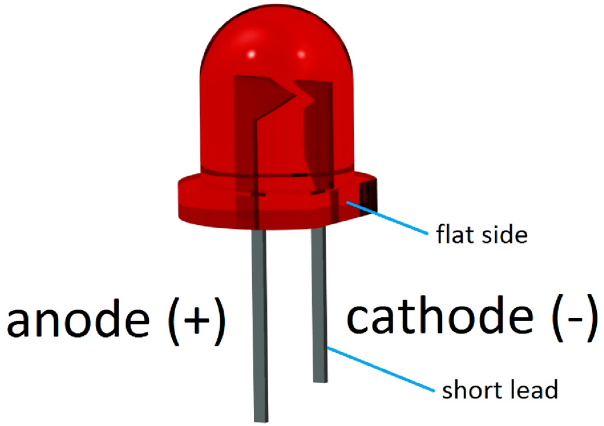
You can use a Multimeter to measure current, voltage, resistance, etc. in an electric circuit.

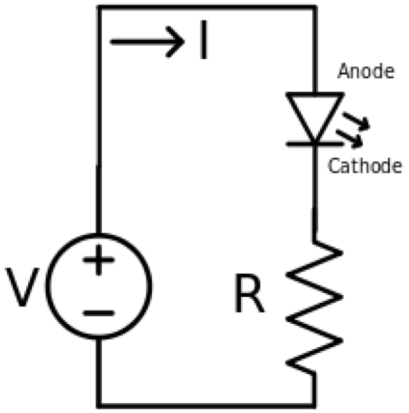
https://learn.sparkfun.com/tutorials/how-to-use-a-multimeter

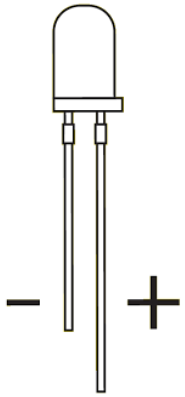
Light-Emitting Diode - LED







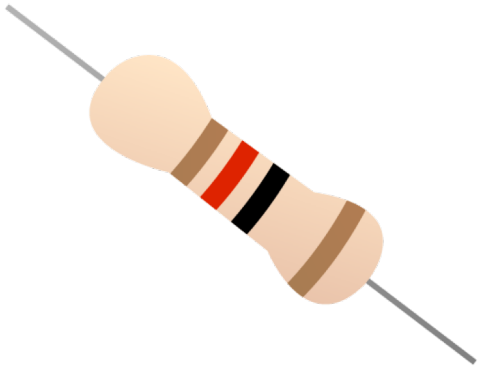




[Wikipedia]

Resistors

Resistance is measured in Ohm (Ω)



Resistors comes in many sizes, e.g., 220Ω , 270Ω, 330Ω, 1kΩm 10kΩ, ...

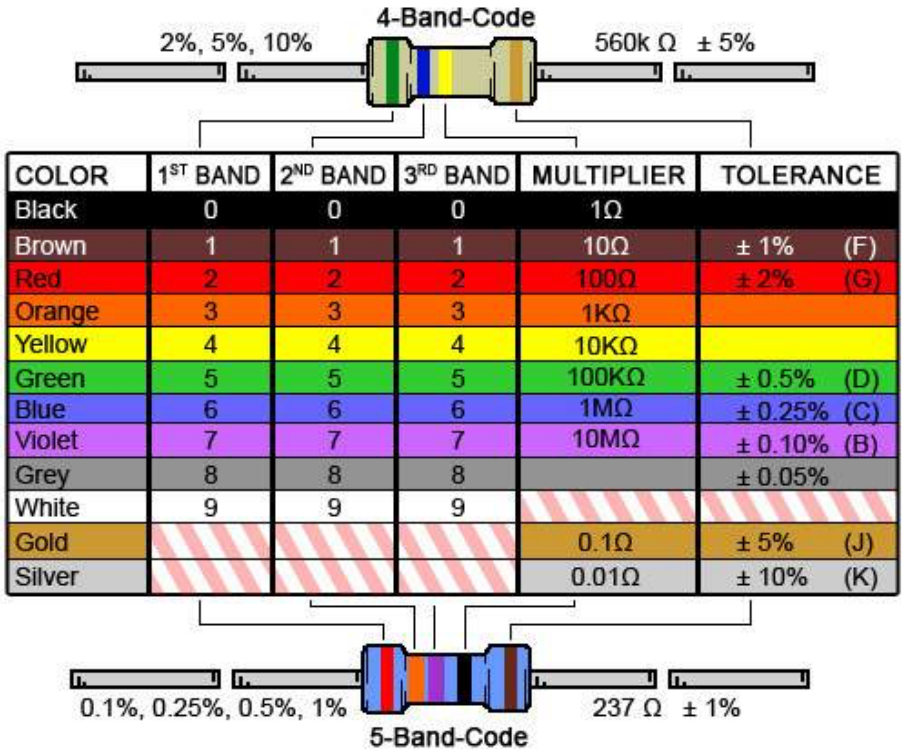
The resistance can be found using Ohms Law

$U = RI$

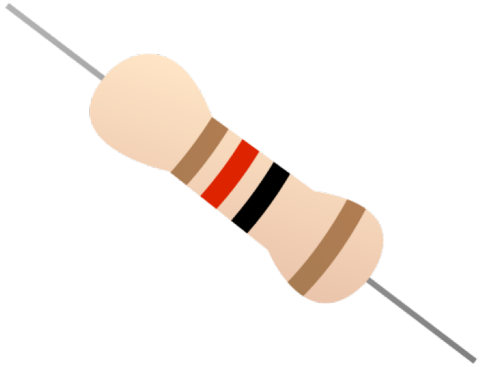
https://en.wikipedia.org/wiki/Resistor

Electrical symbol:

Resistor Color Codes



COLOR	1ST BAND	2ND BAND	3RD BAND	MULTIPLIER	TOLERANCE
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	± 1% (F)
Red	2	2	2	100Ω	± 2% (G)
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	± 0.5% (D)
Blue	6	6	6	1MΩ	± 0.25% (C)
Violet	7	7	7	10MΩ	± 0.10% (B)
Grey	8	8	8		± 0.05%
White	9	9	9		
Gold				0.1Ω	± 5% (J)
Silver				0.01Ω	± 10% (K)



Resistor Color Codes

TRY IT OUT!

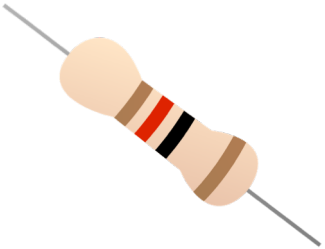
- What is the values for your resistors?
- Use the Color Codes to figure it out
- Use also a Multimeter to see if you get the same results.

Resistor Color Codes

TRY IT OUT!

What is the values for your resistors?

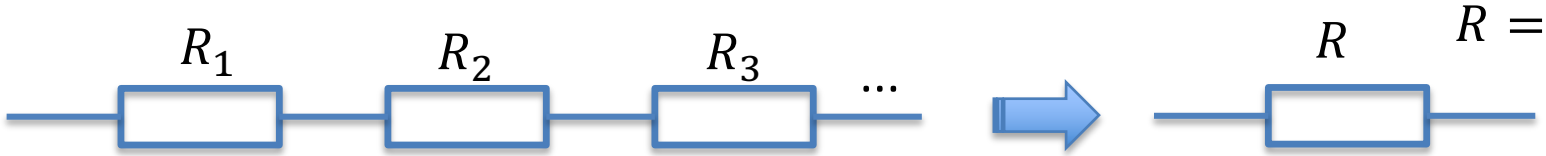
Use a «Resistor Color Code Calculator», which you can find on Internet



http://www.allaboutcircuits.com/tools/resistor-color-code-calculator/

Resistors in Series and Parallel

Resistors in Series:

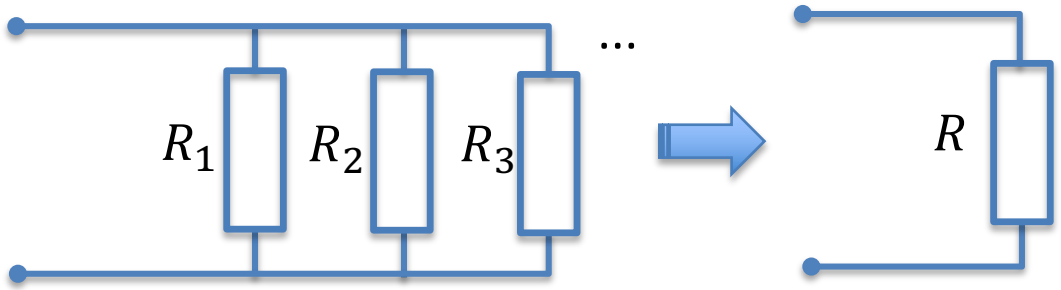


$$R = R_1 + R_2 + R_3 + \cdots$$

The total resistance of resistors connected in series is the sum of their individual resistance values.

When we have resistors in series, the sum of the sub-voltages is equal to the voltage of the voltage source

Resistors in Parallel :



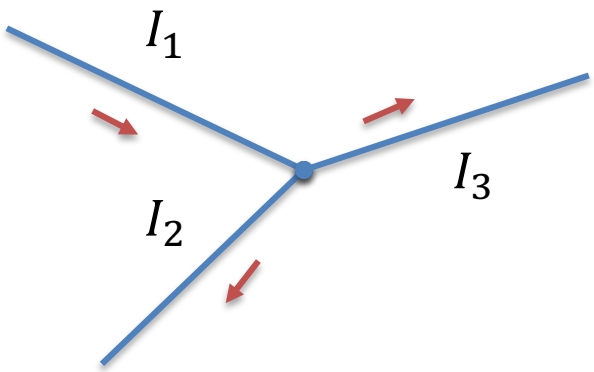
$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \cdots$$

When we have resistors in parallel, the total resistance is always less than the smallest resistors

Kirchhoff's Laws

Kirchhoff's Current Law:

$$\sum_{K=1}^{n} I_k = 0$$

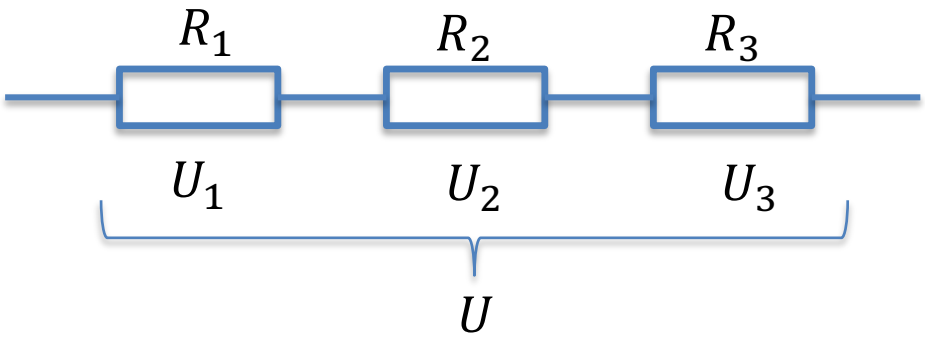


$$I_1 = I_2 + I_3$$

$$I_1 - I_2 - I_3 = 0$$

Kirchhoff's Voltage Law:

$$\sum_{K=1}^{n} U_k = 0$$



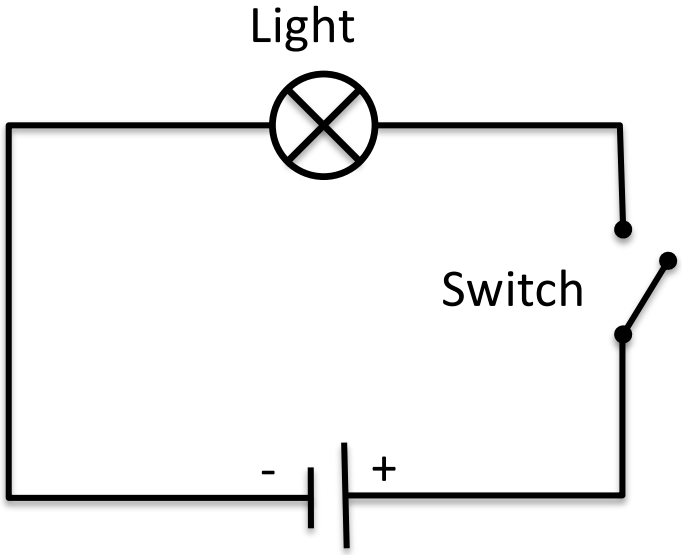
$$U = U_1 + U_2 + U_3 + \cdots$$

https://en.wikipedia.org/wiki/Kirchhoff%27s_circuit_laws

Switch

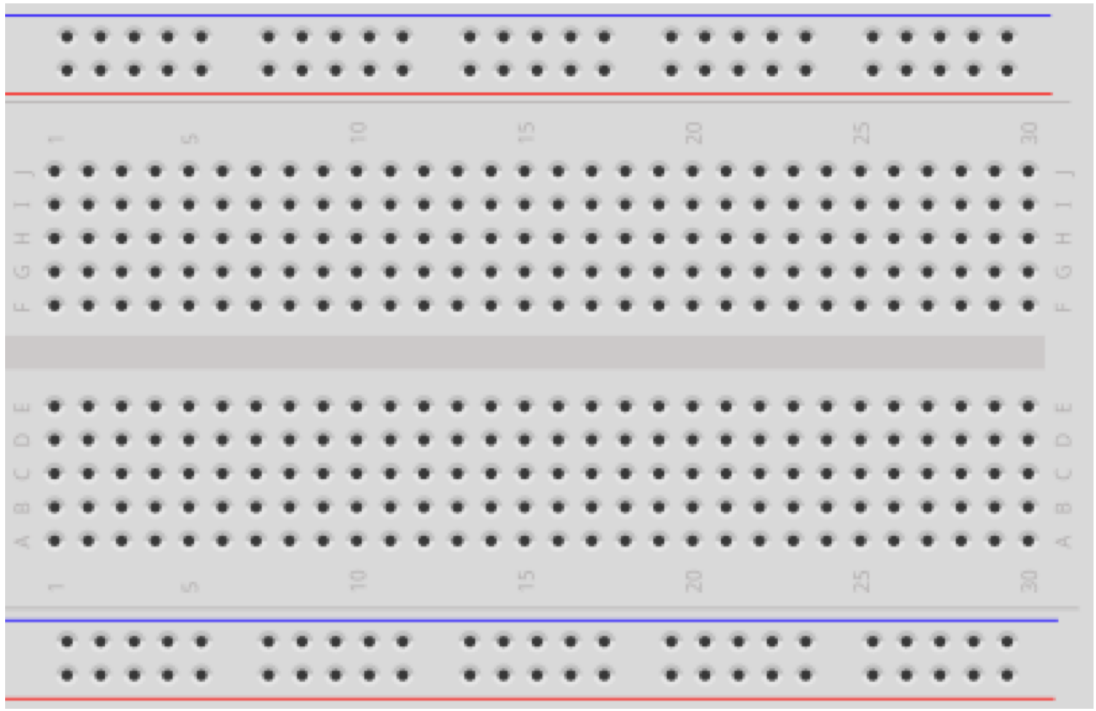
A switch breaks the flow of current through a circuit when open. When closed, the current will flow unobstructed through the circuit.

A switch comes in many flavors

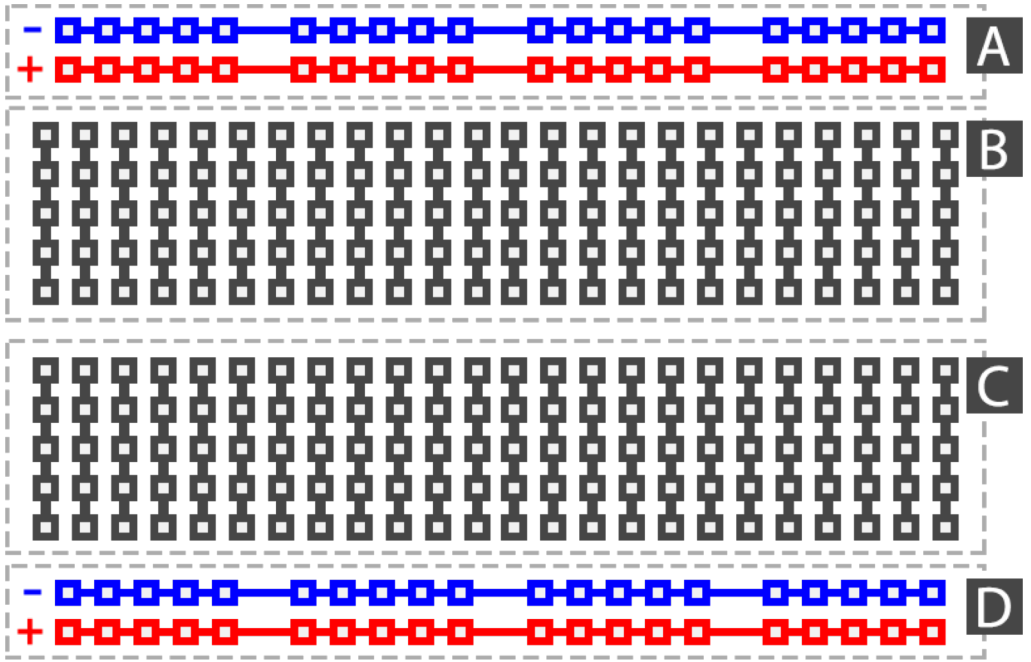


Battery

Breadboard

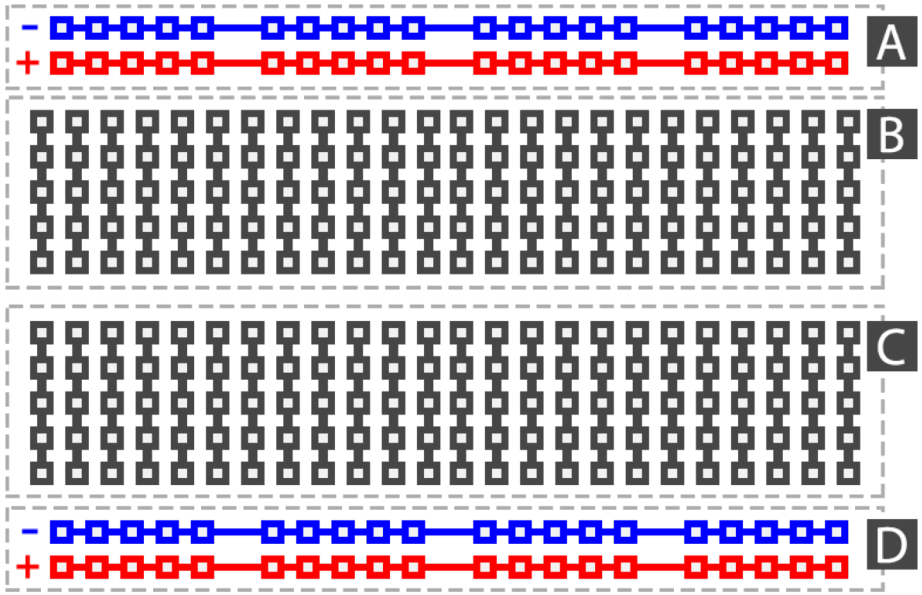


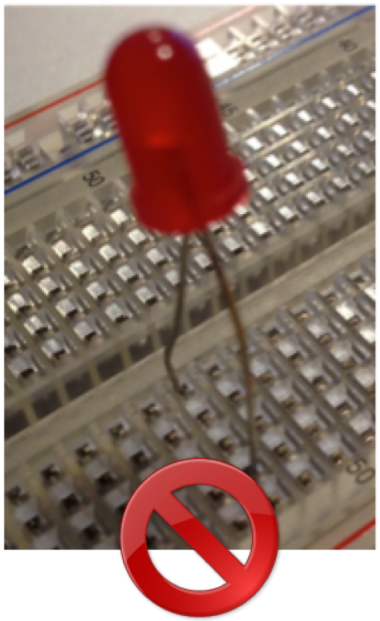
A breadboard is used to wire electric components together

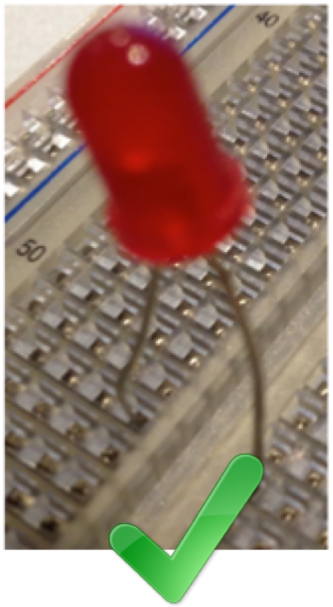


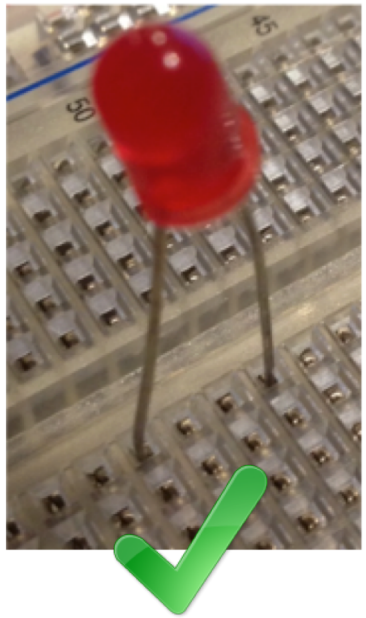
Breadboard – Correct Wiring

Make sure not to short-circuit the components that you wire on the breadboard









Play and Explore

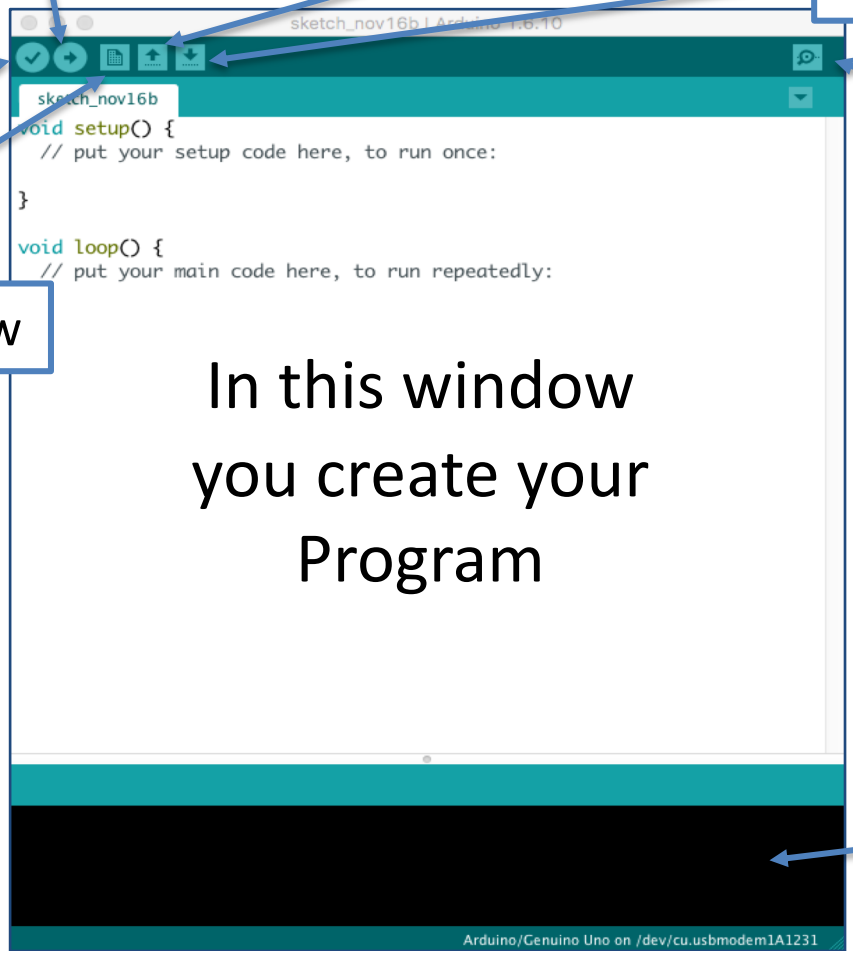
|Programming with Arduino|

https://www.halvorsen.blog

Arduino Development Environment

|Hans-Petter Halvorsen |

Arduino Software



Upload Code to Arduino Board

Open existing Code

Save

Compile and Check if Code is OK

Open Serial Monitor

Creates a New Code Window

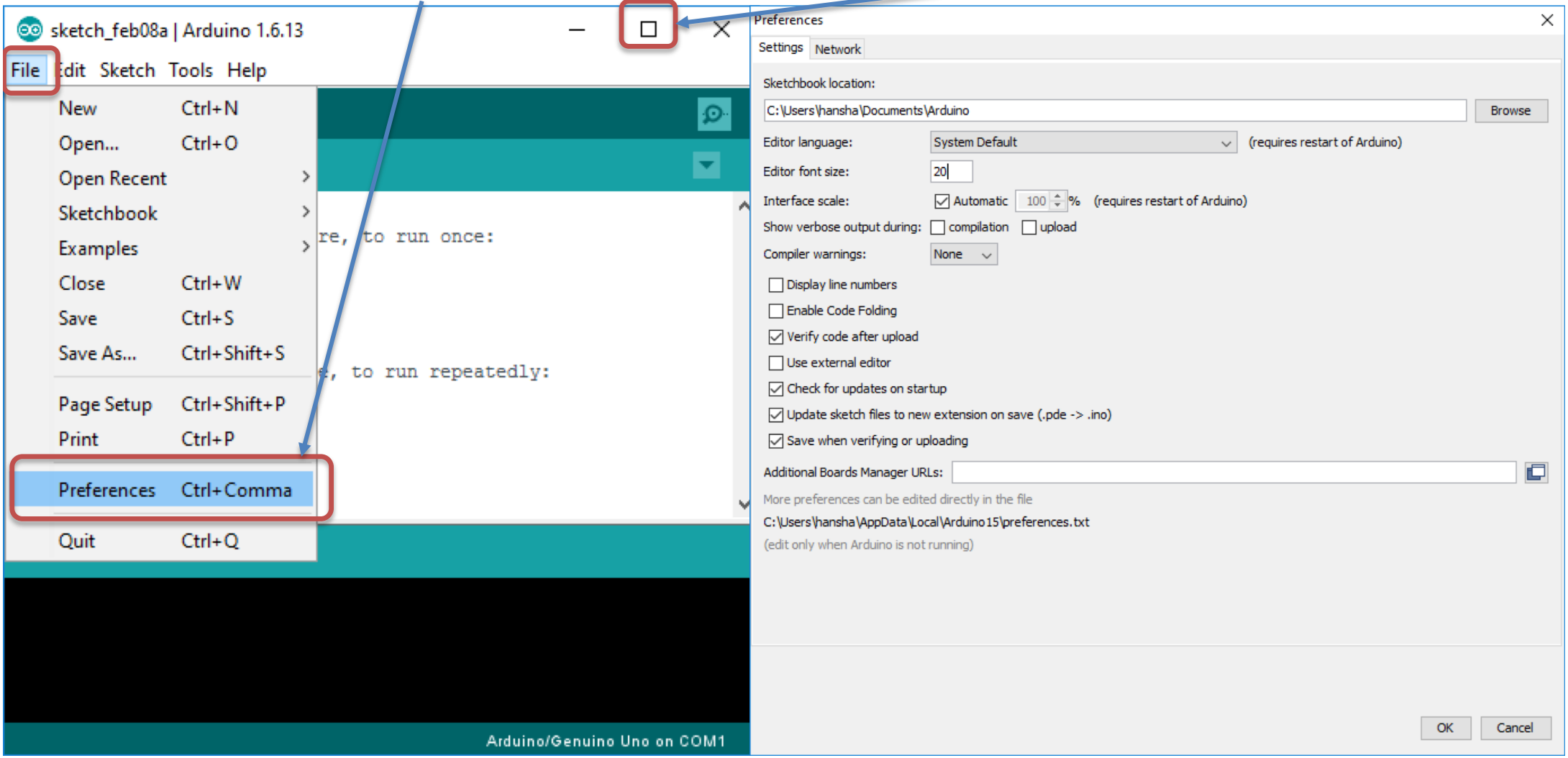
In this window you create your Program

An be downloaded for free:

www.arduino.cc

Error Messages can be seen here

Editor Preferences

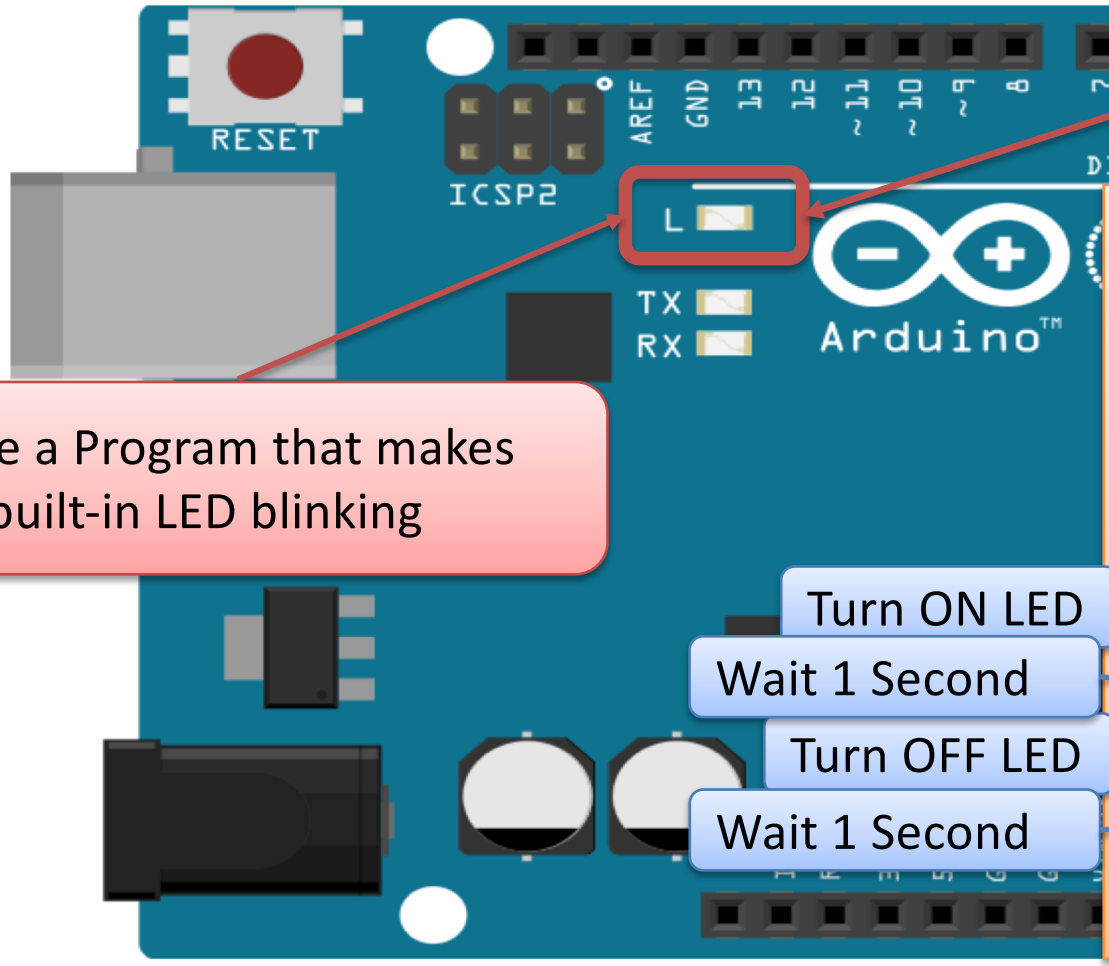


Blinking LED Example

TRY IT OUT!

Arduino UNO has a built-in LED that is connected to Port 13

Make a Program that makes the built-in LED blinking




```
void setup()
{
    pinMode(13, OUTPUT);
}

void loop()
{
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```

Turn ON LED

Wait 1 Second

Turn OFF LED

Wait 1 Second

Blinking LED Example

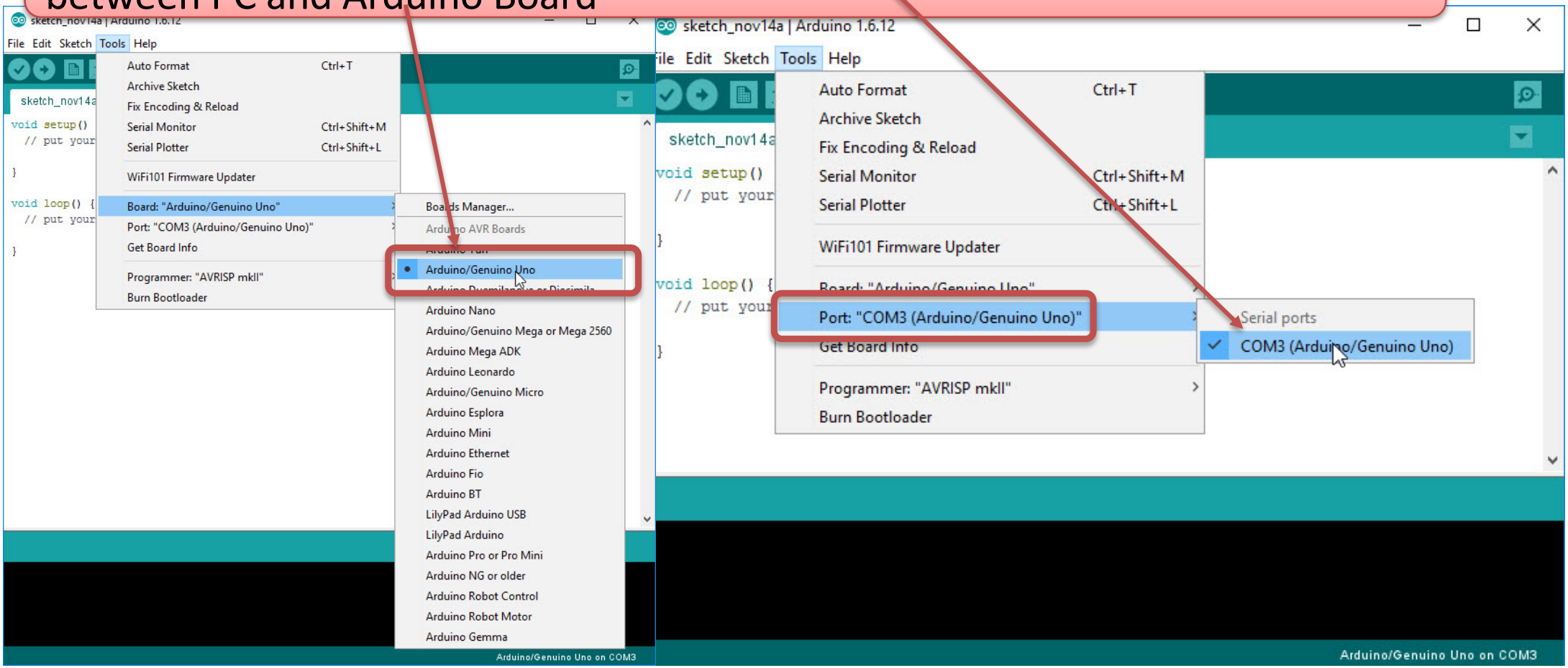
```
void setup()
{
   pinMode(13, OUTPUT);
}

void loop()
{
   digitalWrite(13, HIGH);
   delay(1000);
   digitalWrite(13, LOW);
   delay(1000);
}
```

This Program makes the built-in LED blinking

Try to change from 1000 to 100 – What happens then?

Do you get an Error Message?

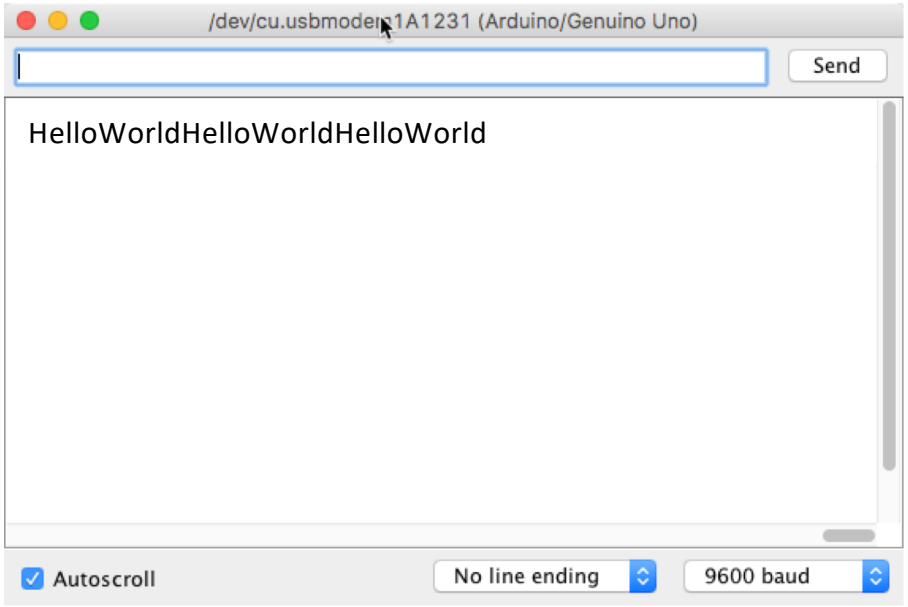
Choose correct Board (Arduino UNO) and the correct Port for Communication between PC and Arduino Board



Serial Monitor

TRY IT OUT!

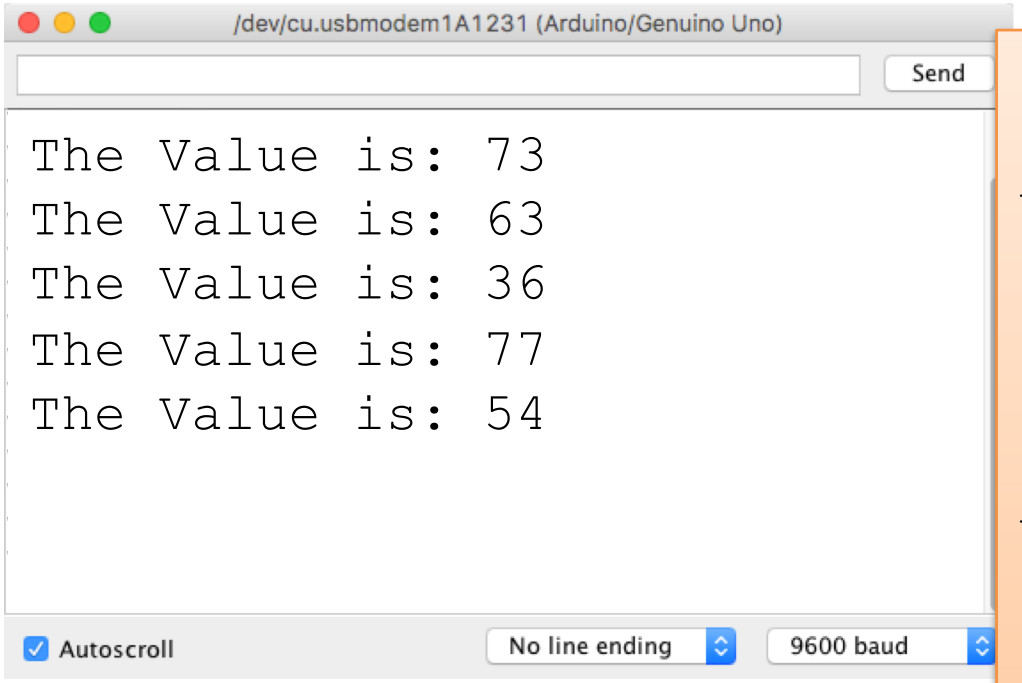
You use the Serial Monitor when Debugging Arduino programs or when you want to show data or values from your program.
You need to have Arduino connected to your PC in order to use the Serial Monitor.




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

void loop()
{
    Serial.print("Hello World");
    delay(1000);
}
```

Serial Monitor

TRY IT OUT!



Here you see how we can write a value to the Serial Monitor. This can be a value from a sensor, e.g., a temperature sensor.

```
int myValue = 0;

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

void loop()
{
    myValue = random(100);
    Serial.print("The Value is: ");
    Serial.println(myValue);
    delay(1000);
}
```

Play and Explore

|Programming with Arduino|

Arduino Programmering

Hans-Petter Halvorsen

Arduino Programs

All Arduino programs must follow the following main structure:

```
// Initialization, define variables, etc.

void setup()
{
   // Initialization
   ...
}

void loop()
{
   //Main Program
   ...
}
```

Arduino Program - Example

```
void setup()
{
   pinMode(11, OUTPUT);          //Set the Pin as an Output
}

void loop()
{
   digitalWrite(11, HIGH);    // Turn on the LED
   delay(1000);               // Wait for one second
   digitalWrite(11, LOW);     // Turn off the LED
   delay(1000);               // Wait for one second
}
```

Arduino Program – Using Comments

```
void setup()
{
    pinMode(11, OUTPUT);          //Set the Pin as an Output
}

void loop()
{
    digitalWrite(11, HIGH);    // Turn on the LED

    /*
    ... This will not be executed by the program because
    it is a comment...
    */
}
```

Creating and Using Functions

```
int z;

void setup()
{
    }

void loop()
{
  z = calculate(2,3);
}

float calculate(int x, int y)
{
    return (x + y);
}
```

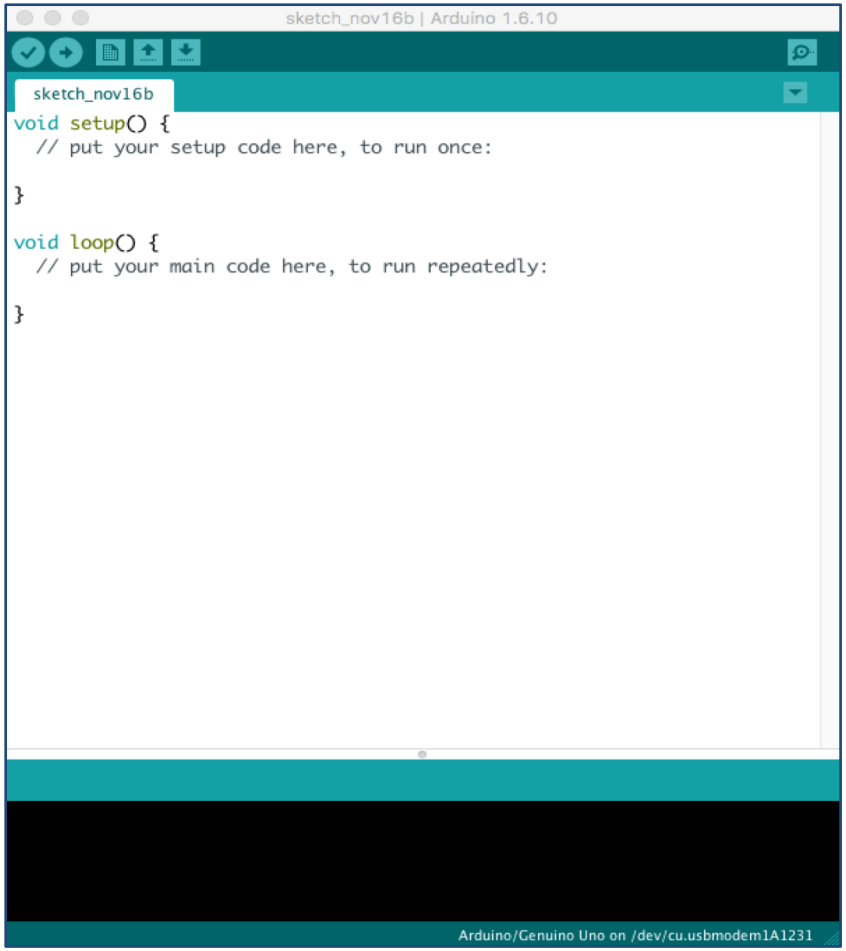
Using the Function

Creating the Function

TRY IT OUT!

Here are some Arduno Examples you should try.

Make sure your Arduino is connected to the PC and start the Code Editor



”Hello World” Example

TRY IT OUT!

Create the following program:

Open the ”Serial Monitor” in order to se the output

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

    Serial.println("Hello, world!");
}

void loop()
{

}
```

”Hello World” Example

TRY IT OUT!

Create the following program:

Open the ”Serial Monitor” in order to se the output

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

void loop()
{
    Serial.println("Hello, world!");
    delay(1000);
}
```

Example

Create the following program:

Open the ”Serial Monitor” in order to se the output

TRY IT OUT!

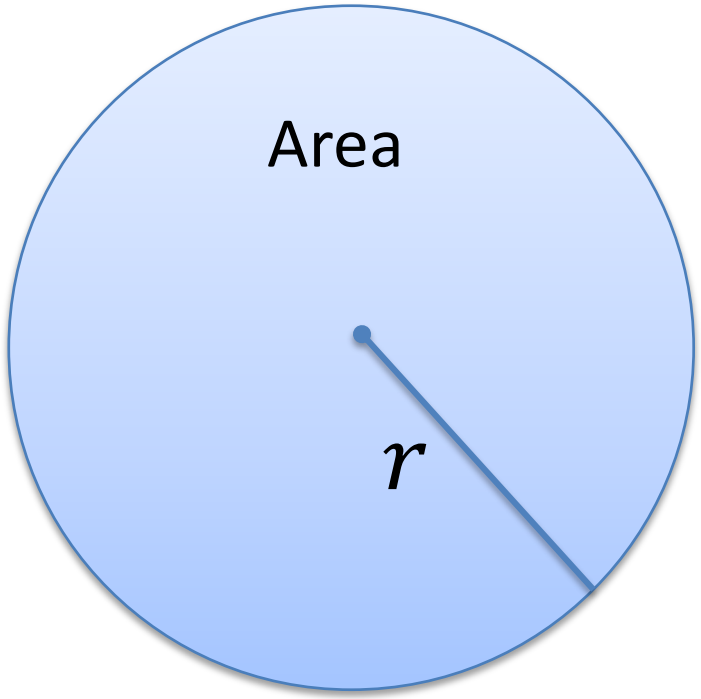
```
int z;int a;int b;
void setup()
{
    Serial.begin(9600);
}
void loop()
{
    a = random(100);
    b = random(100);
    z = calculate(a,b); //Adding 2 Numbers

    //Write Values to Serial Monitor
    Serial.print(a);
    Serial.print(" + ");
    Serial.print(b);
    Serial.print(" = ");
    Serial.println(z);

    delay(1000);
}
float calculate(int x, int y)
{
    return (x + y);
}
```

Creating Functions

TRY IT OUT!

Create a function that calculates the area of a circle with a given radius.

Write the result to the Serial Monitor.



Solution

TRY IT OUT!

```
void setup()
{
    float area;

    Serial.begin(9600);
    // calculate the area of a circle with radius of 9.2
    float r=9,2;
    area = CircleArea(r);
    Serial.print("Area of circle is: ");
    // print area to 4 decimal places
    Serial.println(area, 4);
}

void loop()
{

}

// calculate the area of a circle
float CircleArea(float radius)
{
    float result;
    const float pi = 3.14;

    result = pi * radius * radius;

    return result;
}
```

For Loop

In this program we use a For Loop to find the Sum of 100 Random Numbers.

Then we find the Average.

The Sum and Average should be written to the Serial Monitor.

```
int x; int sum = 0; float gjennomsnitt = 0;
void setup()
{
     Serial.begin(9600);
}

void loop()
{
     sum = 0;
     for (int i = 0; i<100; i++)
     {
          x = random(100);
          sum = sum + x;
     }

     average = sum / 100;
     Serial.print(" Sum = ");
     Serial.print(sum);
     Serial.print(" ,
     Average = ");
     Serial.println(average);
     delay(1000);
}
```

TRY IT OUT!

Arrays

Here we shall use arrays in the Arduino program

Create this program from scratch and open the Serial Monitor to see the result.

TRY IT OUT!

```
const int arraysize = 100;
int x;
int sum = 0;
float average = 0;
int myarray[arraysize];

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

void loop()
{
	sum = 0;
	for (int i = 0; i < arraysize; i++)
	{
		x = random(200);
		myarray[i] = x;
	}
	sum = calculateSum(myarray);
	average = sum / 100;
	Serial.print(" Sum = ");
	Serial.print(sum);
	Serial.print(" , Average = ");
	Serial.println(average);
	delay(1000);
}

int calculateSum (int sumarray[])
{
	for (int i = 0; i < arraysize; i++)
	{
		sum = sum + sumarray[i];
	}
	return sum;
}
```

Arduino Programming

Here you will find complete overview of the Arduino programming language:

https://www.arduino.cc/en/Reference/HomePage

Play and Explore

|Programming with Arduino|

https://www.halvorsen.blog

Arduino Examples

|Hans-Petter Halvorsen |

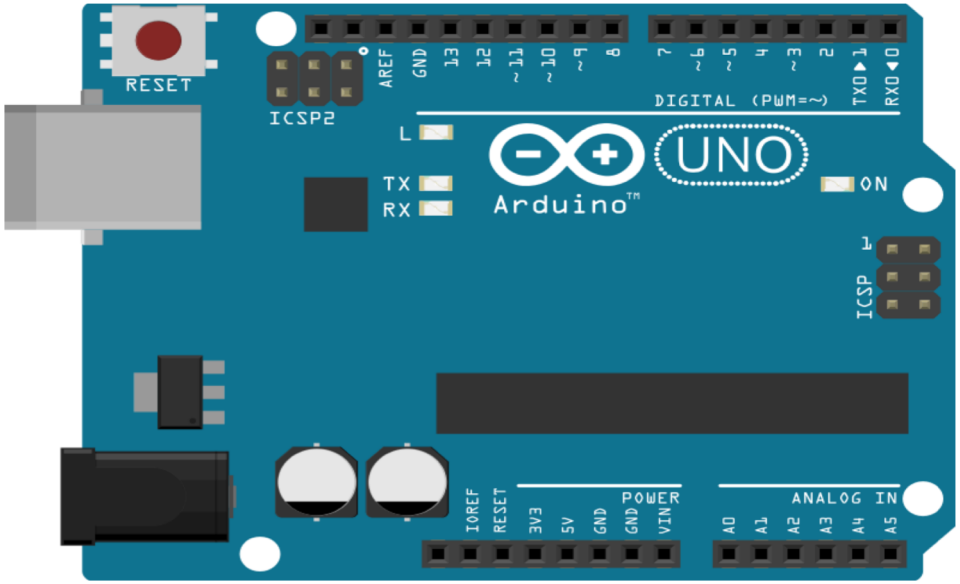
Arduino Examples

On the next pages you will find some Examples/Tasks that you should do step by step as stated in the text.

When you have finished a specific Example/Task, it is recommended that you “Play and Explore”, i.e., make small changes in the program, etc.

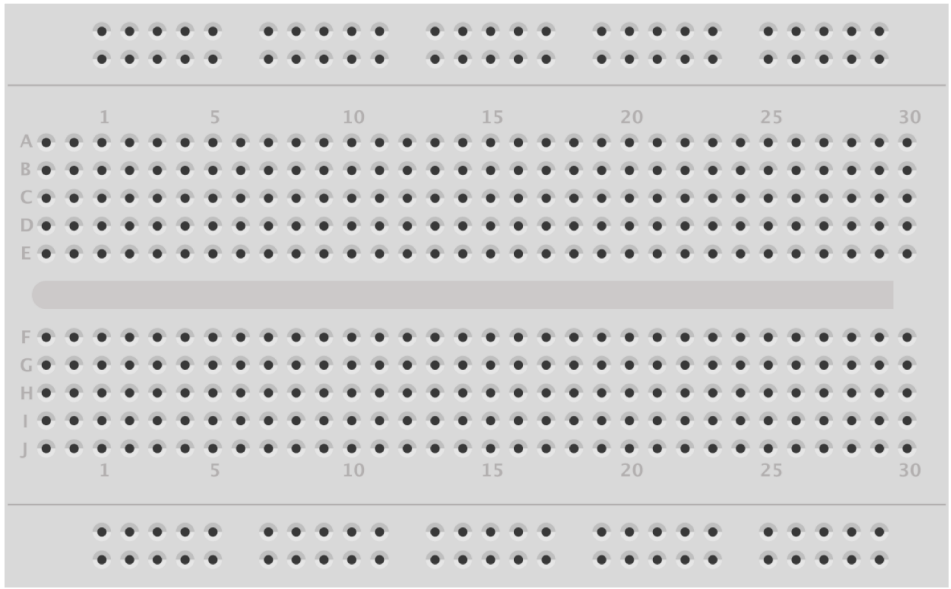
We need the following

Arduino UNO



+

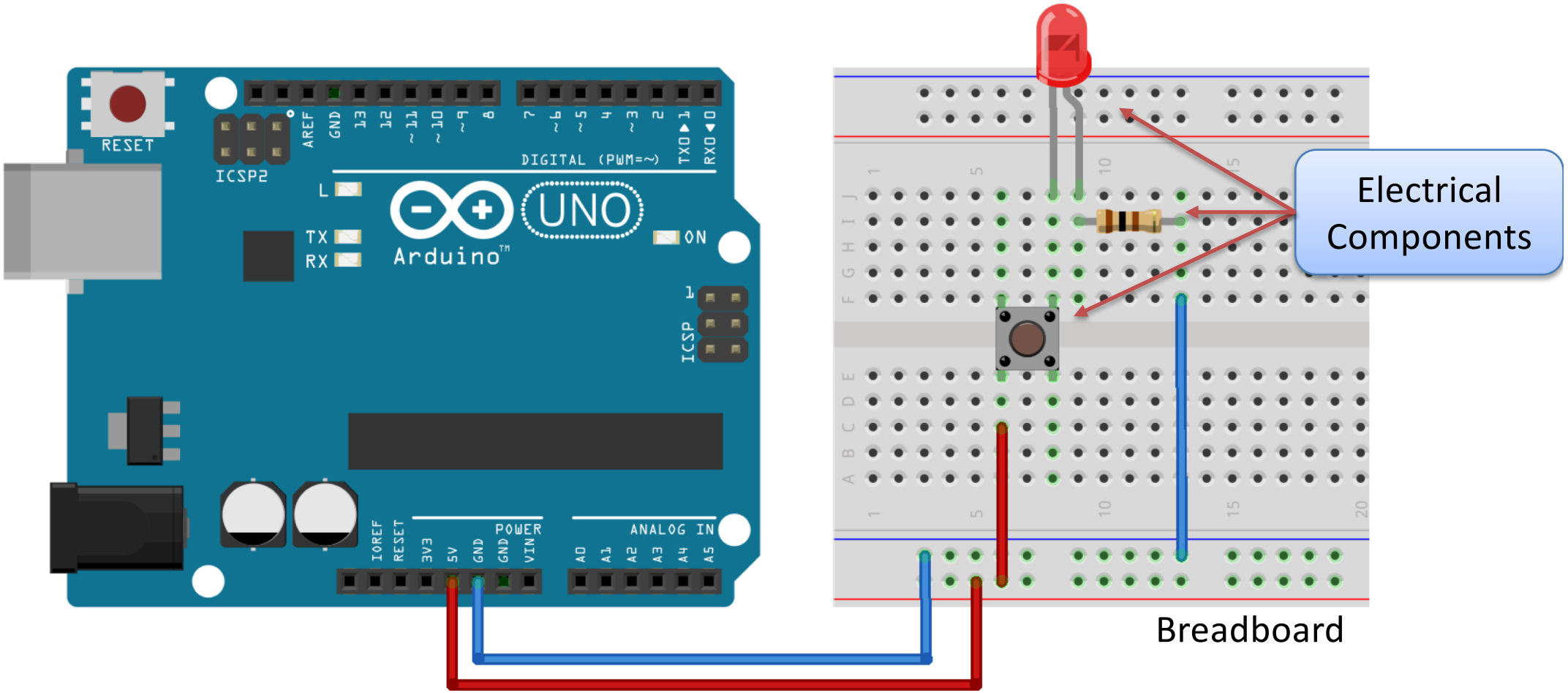
Breadboard



+ LEDs, Resistors, Wires, etc.

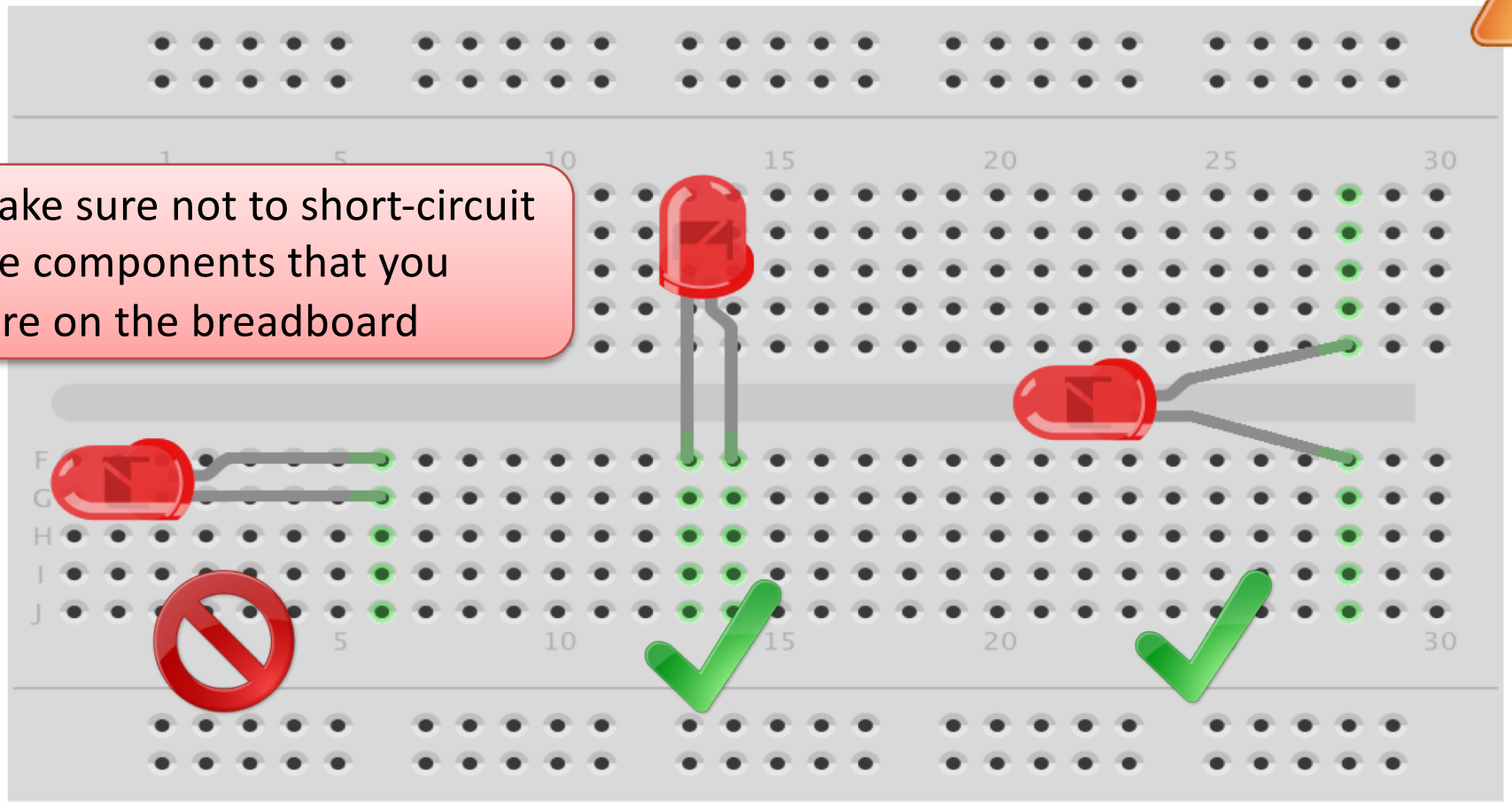
The Breadboard is used to connect components and electrical circuits

Breadboard Example



Wires used to connect the electrical components together to a closed circuit

Breadboard Wiring



The Breadboard is used to connect components and electrical circuits

fritzing

Resistors

TRY IT OUT!

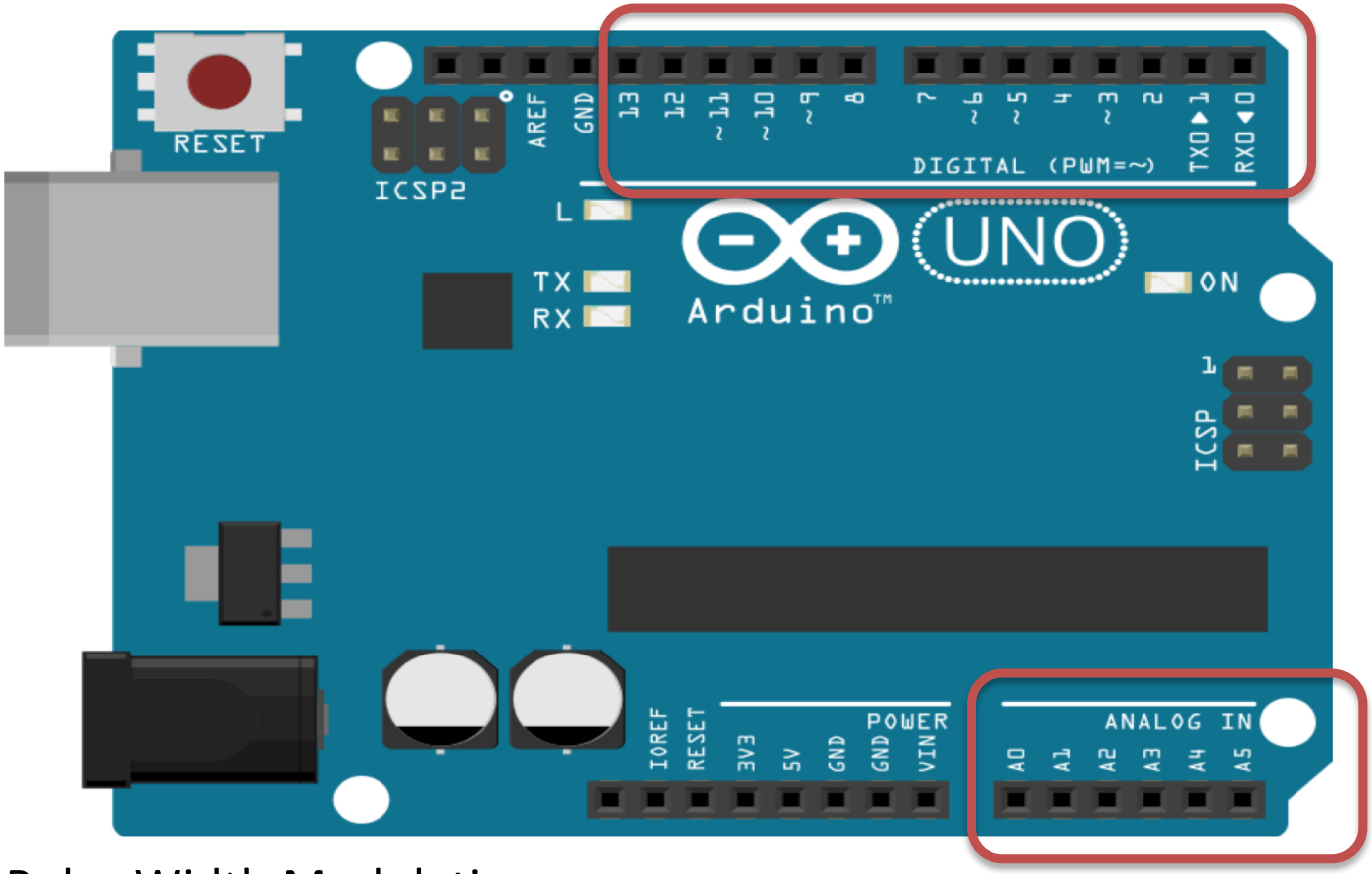
What is the values for your resistors?

- Use the Color Codes to figure it out
- Or use also a Multimeter to see if you get the same results.
- Or use a Resistor Calculator

http://www.allaboutcircuits.com/tools/resistor-color-code-calculator/

Inputs and Outputs (Analog and Digital)

Digital Inputs and Digital Outputs



You can choose from the code if they are to be inputs or outputs

Those marked with ~ can also be used as "Analog Outputs", so-called PWM outputs

Analog Inputs

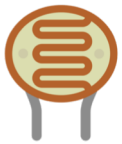
PWM - Pulse Width Modulation

Sensors and Actuators

Temperature Sensor

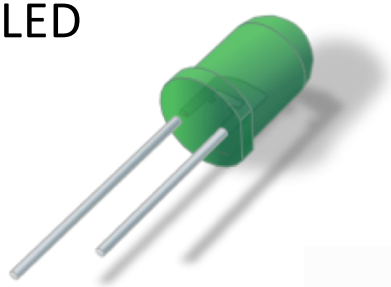
Push Button

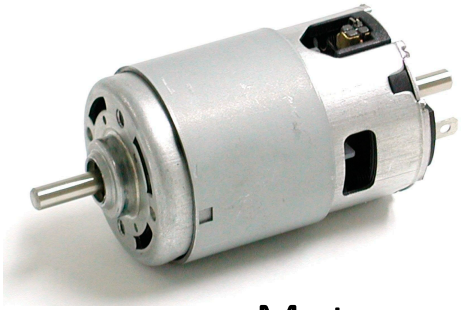
Light Sensor



Potentiometer

LED



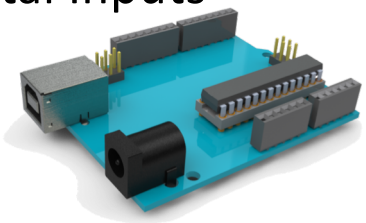


Motor

Buzzer

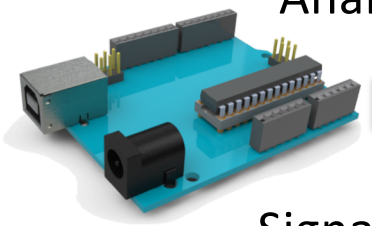
Analog/Digital Inputs

Sensors



Signals from the surroundings

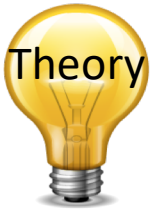
Analog/Digital Outputs



Actuators

Signals to the surroundings

Sensors and Actuators



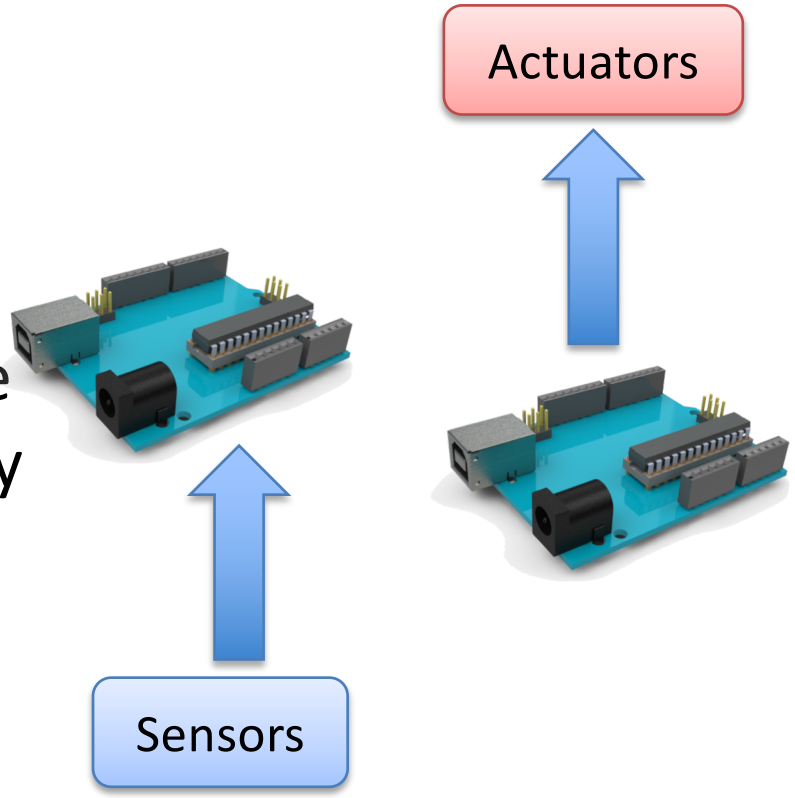
- A **Sensor** is a converter that measures a physical size and converts it to a signal that can be read by an instrument, data acquisition device, or an Arduino.
  Examples: temperature sensor, pressure sensor, etc.
- An **Actuator** is a kind of motor that moves or controls a mechanism or system. It is powered by an energy source, typical electric current, hydraulic fluid pressure, or air pressure, and converts this energy into motion.
  Examples: Engine, Pump, Valve, etc.

http://en.wikipedia.org/wiki/Sensor

http://en.wikipedia.org/wiki/Actuator

Sensors and Actuators

- The sensors and actuators can be either digital or analog.
- Some sensors and actuators have been made for Arduino, while others need to be connected in some circuit to work properly with Arduino.
- Many of these come with ready-made libraries for Arduino, so they are easy to use.



Examples

1. Electrical Circuit Example
2. Blinking LED Example
3. Switch Example
4. Potentiometer Example
5. Temperature Example
6. Light Sensor Example
7. Thermistor Example

Play and Explore

|Programming with Arduino|

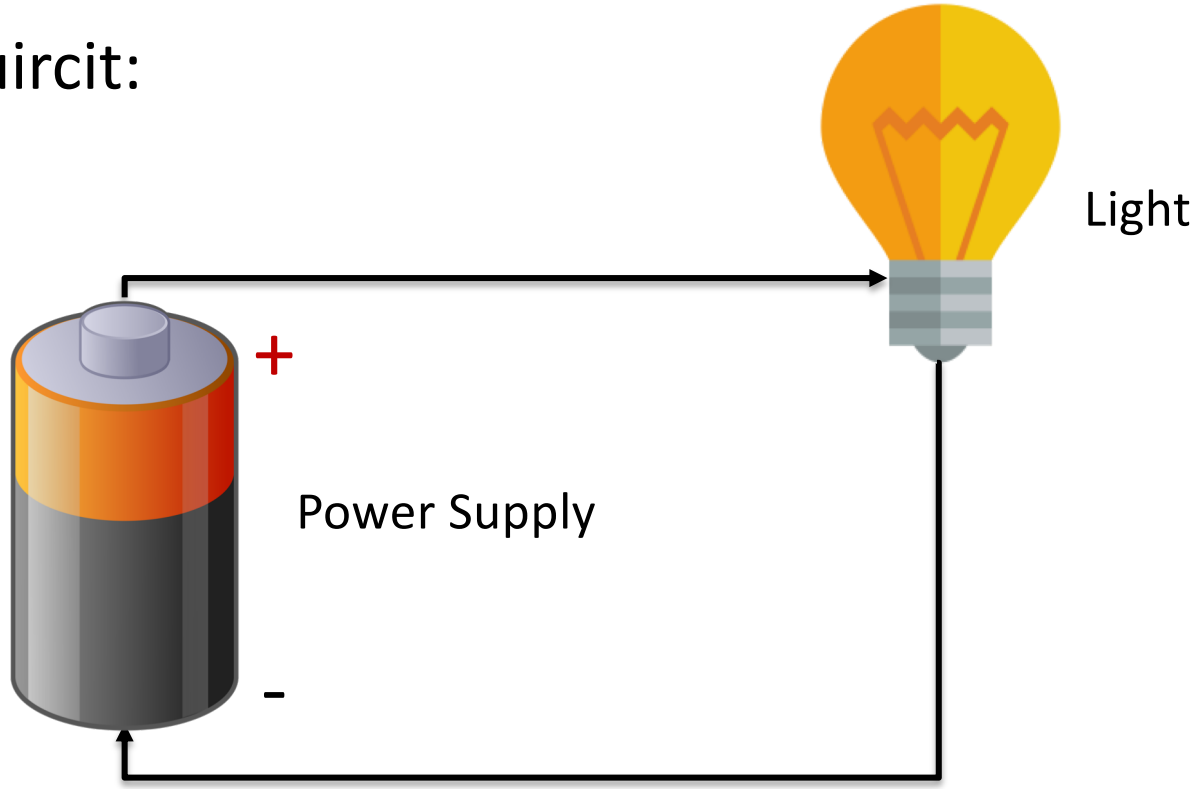
Example 1

TRY IT OUT!

Electrical Circuits

Electrical Cuicits

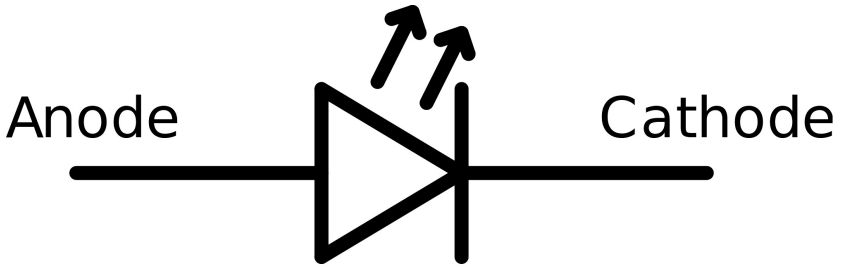
Lets create the following cuircit:

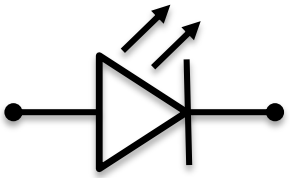


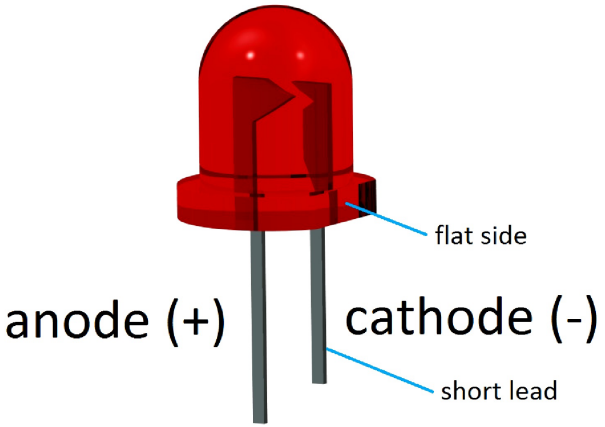
Instead of using a Battery we will use the Arduino board as a Power Supply (5V)

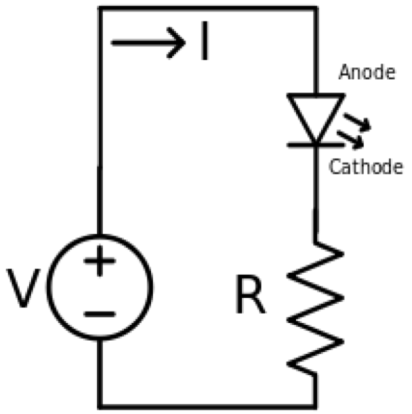
Example 1

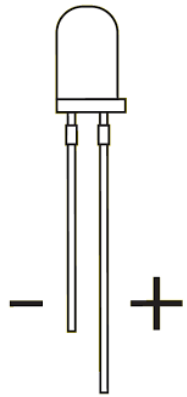
Light-Emitting Diode - LED





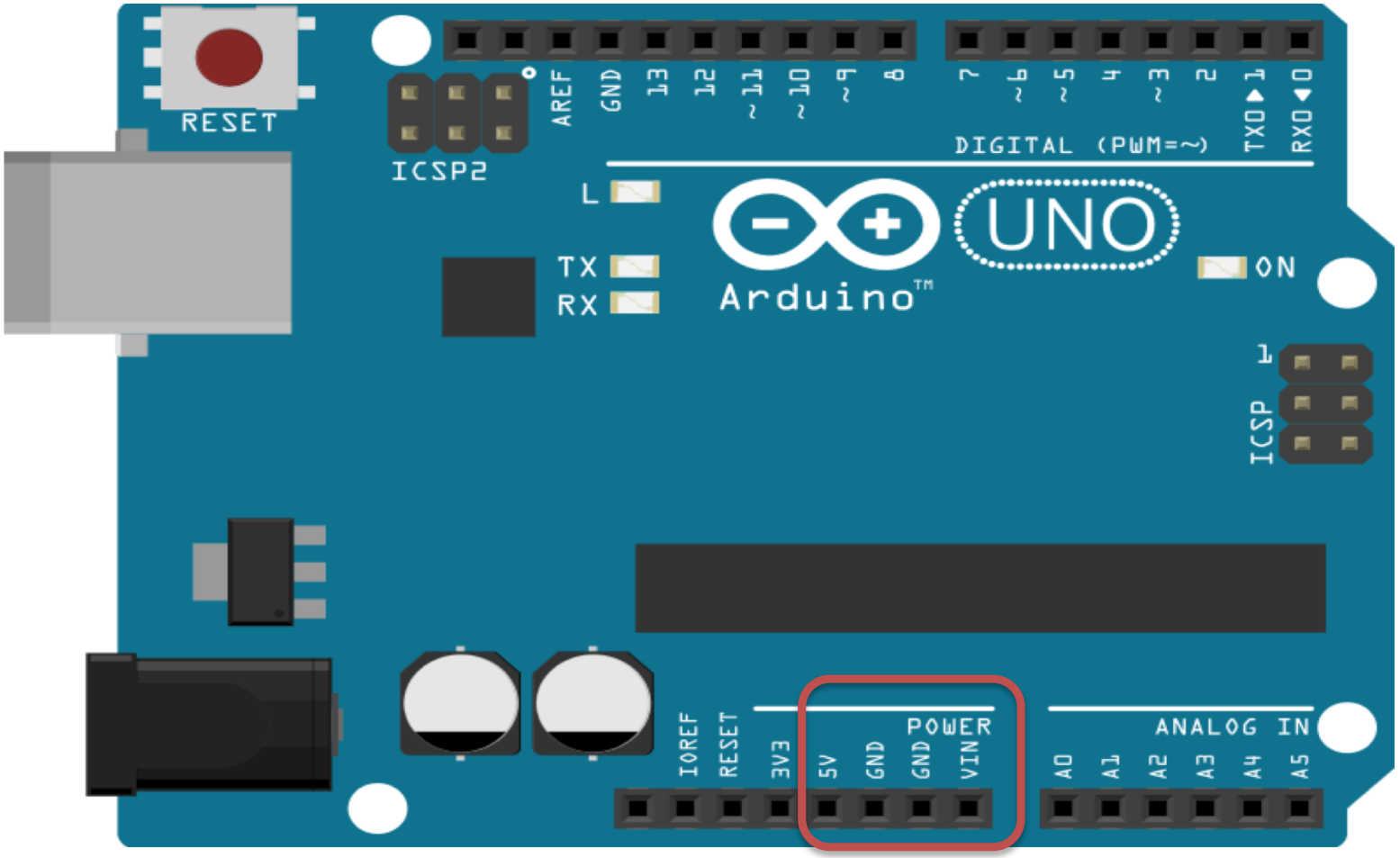






[Wikipedia]

Introduction



We will use the POWER ports ”5V” and ”GND” on the Arduino board

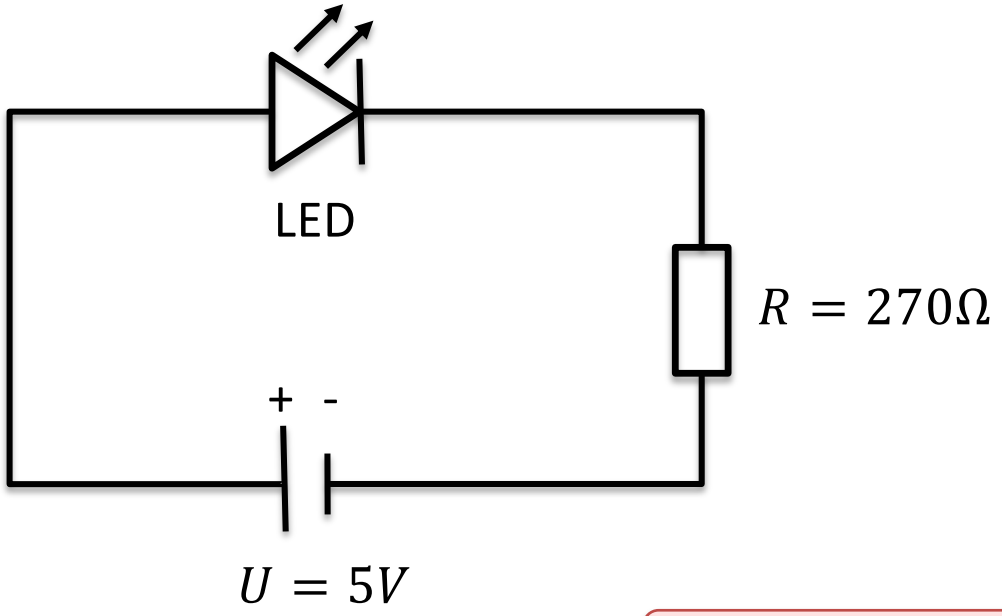
GND = Ground

Example 1

Electrical Cuircuit

TRY IT OUT!

Make the following circuit using the Arduino board and a Breadboard:



Equipment:

- Breadboard
- LED
- Resistor
- Wires
- Multi-meter

Note! No Arduino Program is needed in this example

Example 1

Why do you need a Resistor?

If the current becomes to large, the LED will be destroyed. To prevent this to happen, we will use a Resistor to limit the amount of current in the cuircuit.

What should be the size of the Resistor?

A LED typically need a current like 20mA (can be found in the LED Datasheet).
We use Ohm's Law:

$$U = RI$$

Arduino gives U=5V and I=20mA. We then get:

$$R = \frac{U}{I}$$

The Resistor needed will be $R = \frac{5V}{0.02A} = 250\Omega$. Resistors with R=$250\Omega$ is not so common, so we can use the closest Resistors we have, e.g., 270Ω

Example 1

Find the Resistor Size

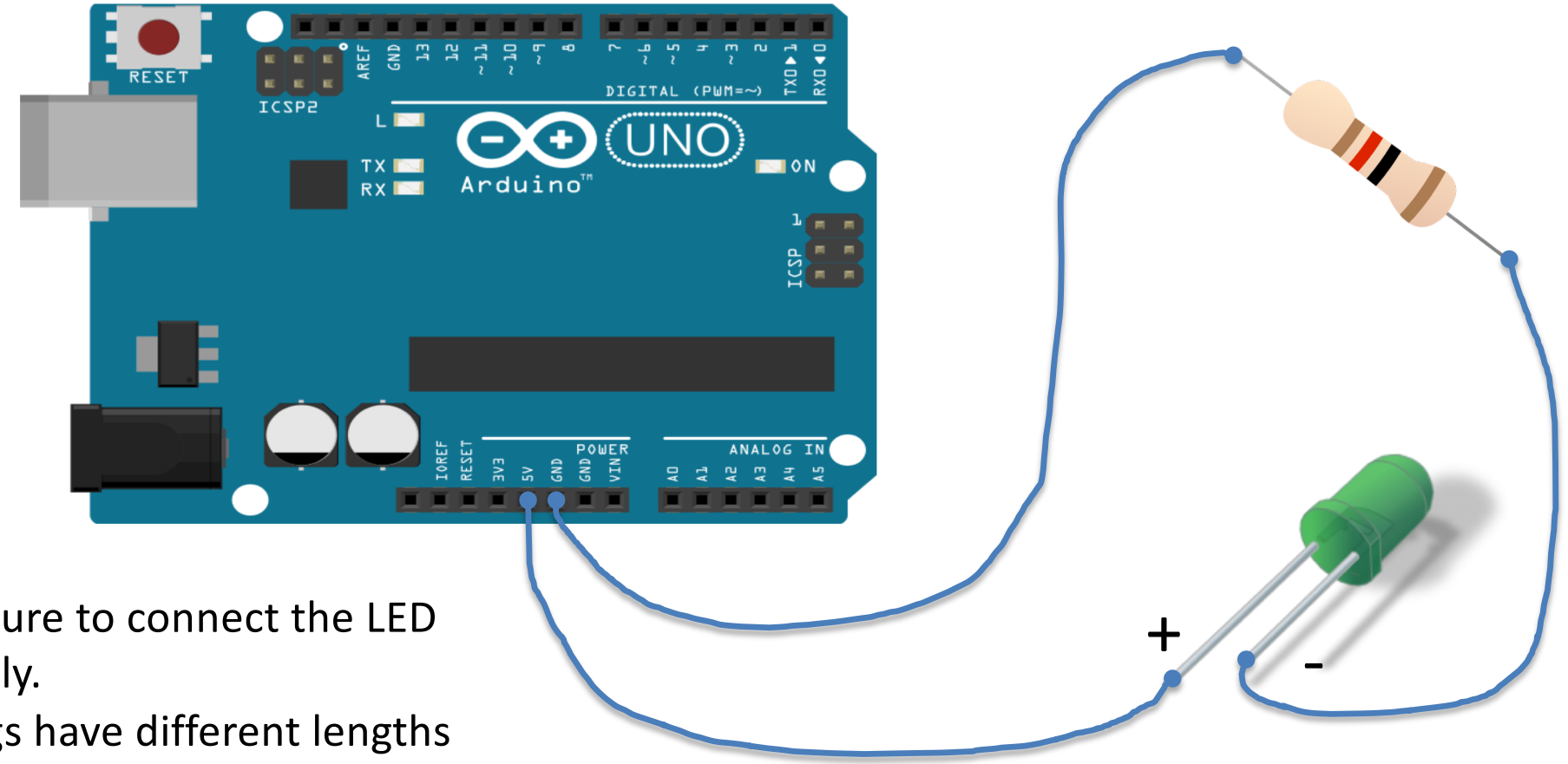
Find the correct Resistor by using the Color Codes or a Mult-meter:

You may also use a ”Resistor Calculator” which you find online:

http://www.allaboutcircuits.com/tools/resistor-color-code-calculator/

Example 1

Wiring

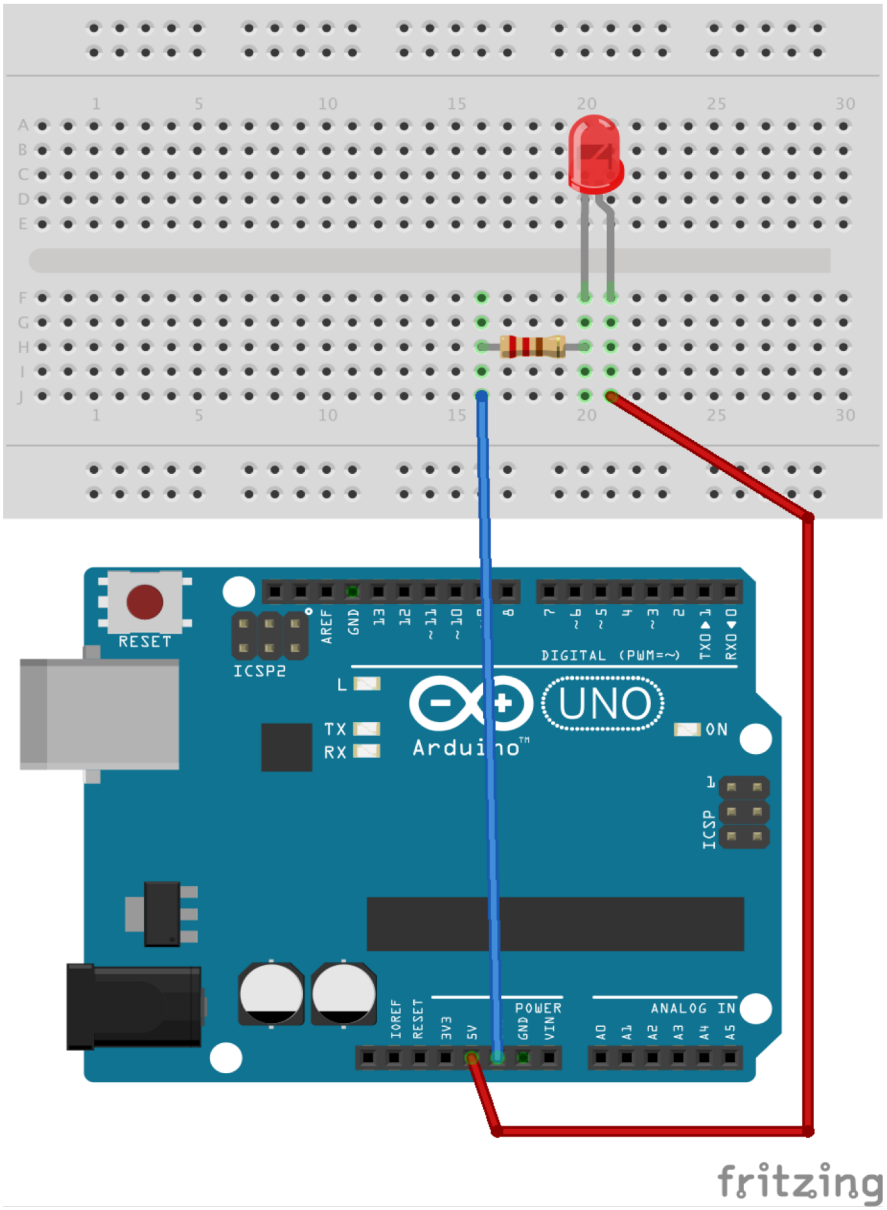


Make sure to connect the LED correctly.
The legs have different lengths

Example 1

Use a Breadboard – see next page

Wiring



TRY IT OUT!

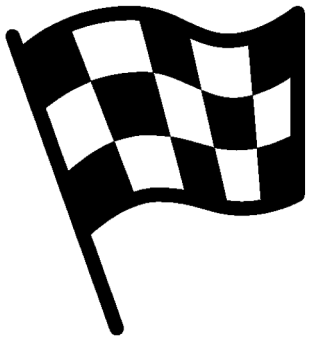
Make sure to connect the LED correctly.
The legs have different lengths

Note! Use a Resistor $R = 270\Omega$ to protect the LED

Example 1

Play and Explore

|Programming with Arduino|

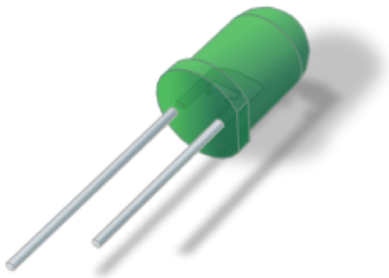


|Programming with Arduino|

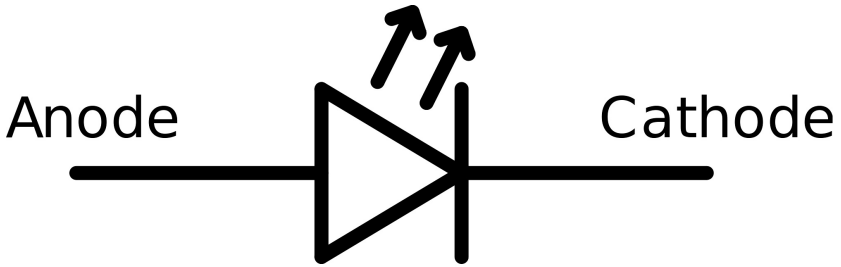
Example 2

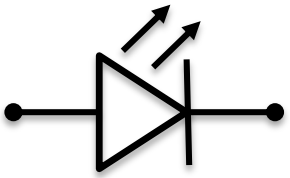
TRY IT OUT!

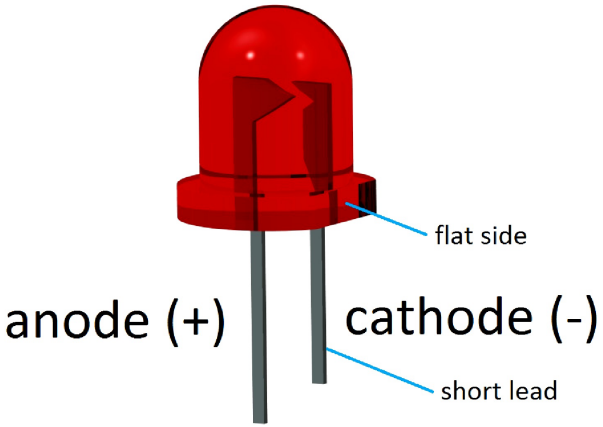
Blinking LED

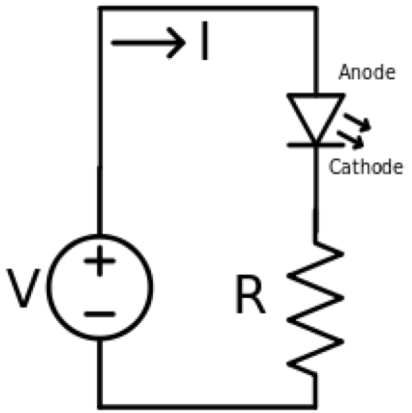


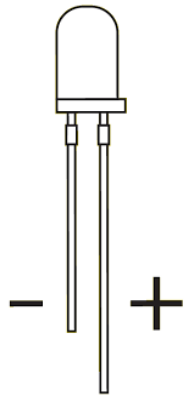
Light-emitting diode - LED











[Wikipedia]

Introduction

We will make a program that makes the LED start blinking.

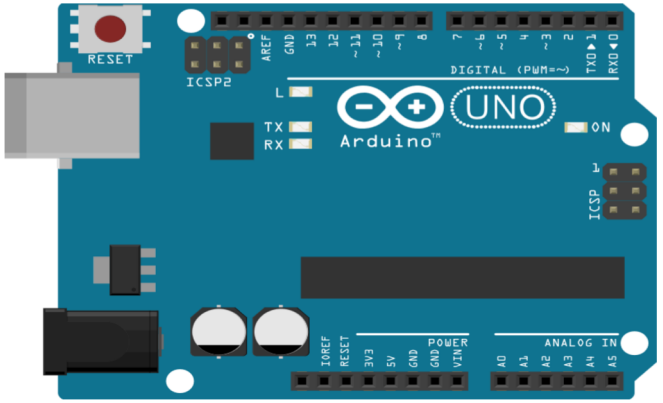
How-To Do it:

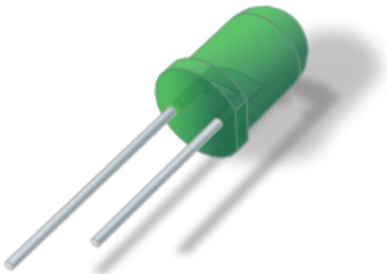
1. Wire the cuircuit and components
2. Make the Arduino program

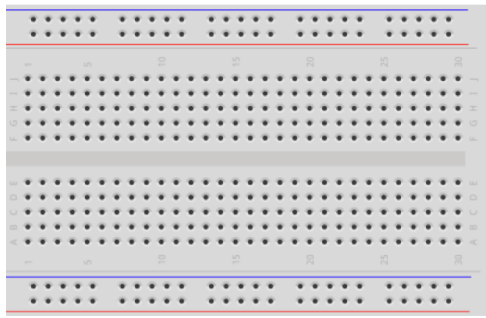
Example 2

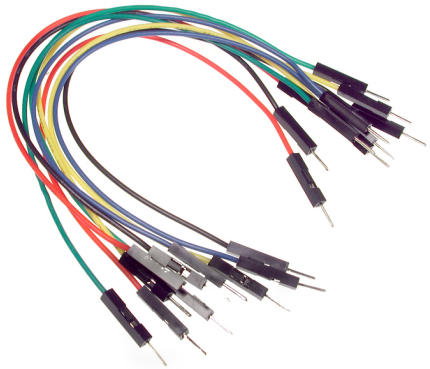
Equipment

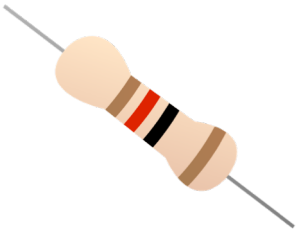
- Arduino UNO
- Breadboard
- LED
- Resistor, $R = 270\Omega$
- Wires (Jumper Wires)







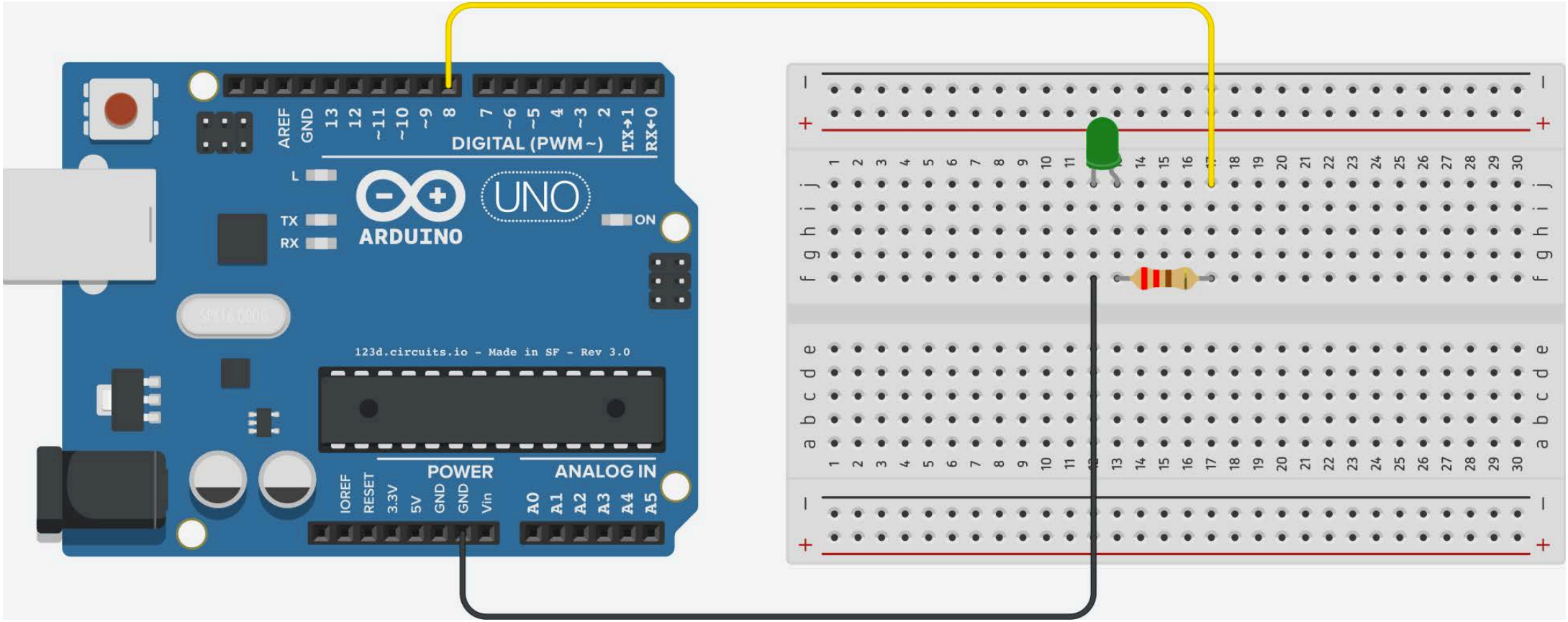




Example 2

Wiring

TRY IT OUT!



Example 2

Programming

Program Structure

```
//Globale variable
...

void setup()
{
  //Initialization
}

void loop()
{
  //Main Program
}
```

You need to use the following:

Which Pin (0, 1, 3, ...) are you using?

```
pinMode(pin, mode);
```

A Digital Pin can either be an INPUT or an OUTPUT. Since we shall use it to turn-on a LED, ww set it to OUTPUT.

```
digitalWrite(pin, value);
```

Turn-on LED

Turn-off LED

A Digital PIn can have 2 values, either HIGH or LOW

```
delay(ms);
```

The delay() fuction make a small pause in milliseconds (ms), e.g,. delay(1000) pause the program for 1 second

Example 2

Arduino Language Reference: https://www.arduino.cc/en/Reference

Arduino Program

TRY IT OUT!

```
void setup()
{
   pinMode(8, OUTPUT);
}

void loop()
{

   digitalWrite(8, HIGH);    // Turn on the LED
   delay(1000);              // Wait for one second
   digitalWrite(8, LOW);     // Turn off the LED
   delay(1000);              // Wait for one second
}
```

Example 2

Arduino Program

Example 2

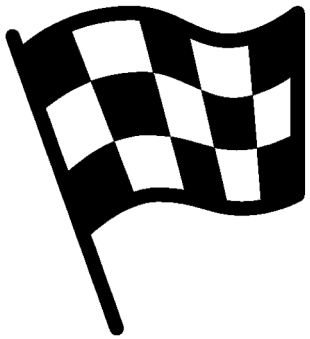
TRY IT OUT!

```
int ledPin = 8;

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

void loop()
{

    digitalWrite(ledPin, HIGH);   // Turn on the LED
    delay(1000);                  // Wait for one second
    digitalWrite(ledPin, LOW);    // Turn off the LED
    delay(1000);                  // Wait for one second

}
```

En ørliten forbedring. Vi bruker en variabel til å definere pinne-nummeret

Play and Explore

|Programming with Arduino|



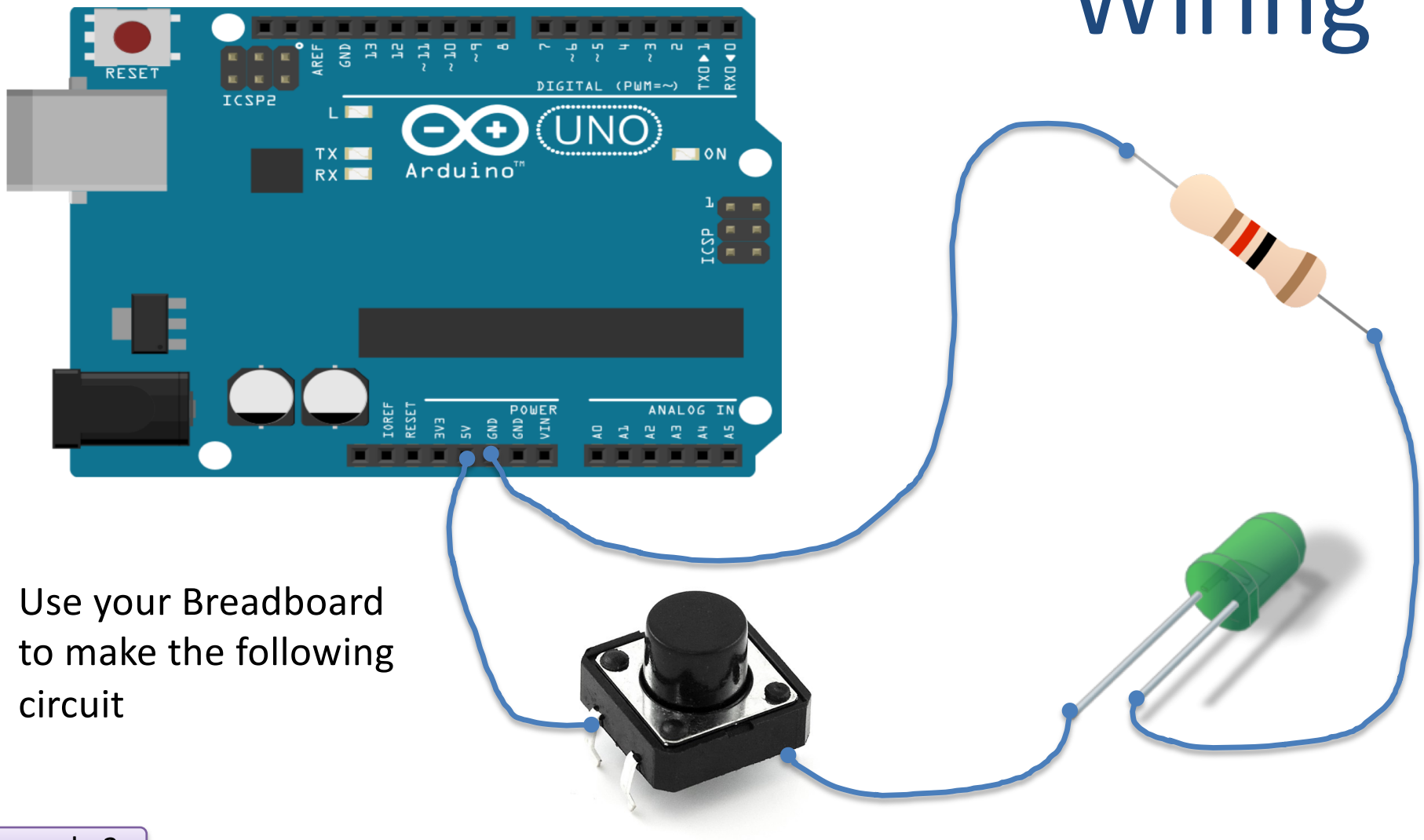
|Programming with Arduino|

Example 3

TRY IT OUT!

Switch

Wiring

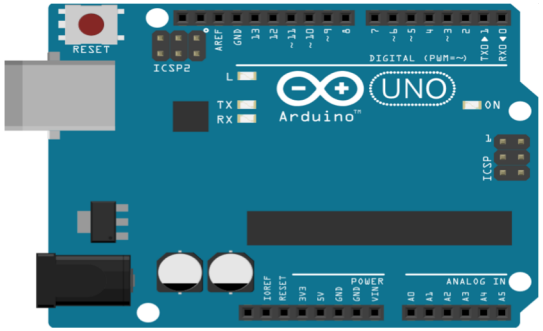


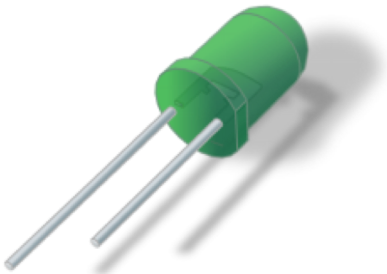
Use your Breadboard to make the following circuit

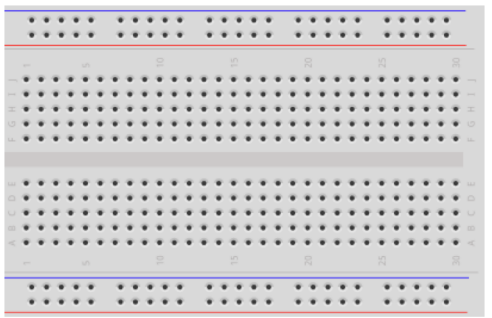
Example 3

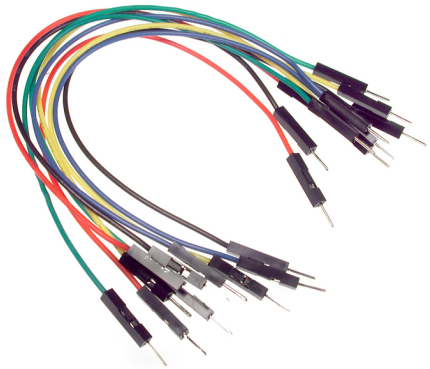
Equipment

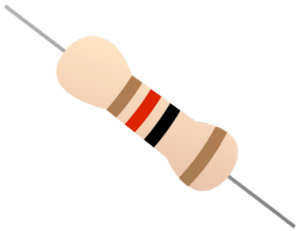
- Arduino
- Breadboard
- LED
- Switch
- Resistor, $R = 270\ \Omega$
- Some Wires





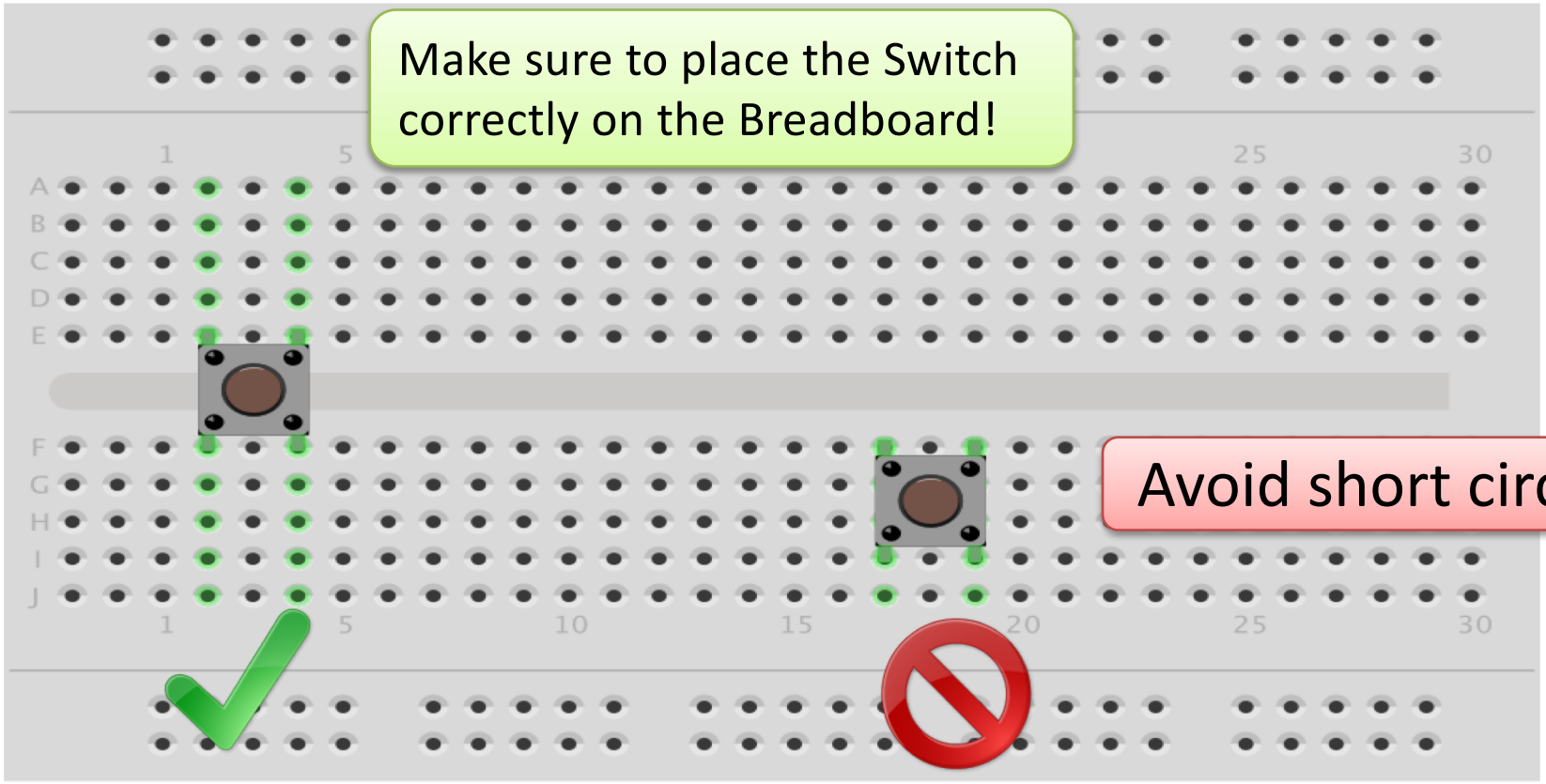






Example 3

Breadboard



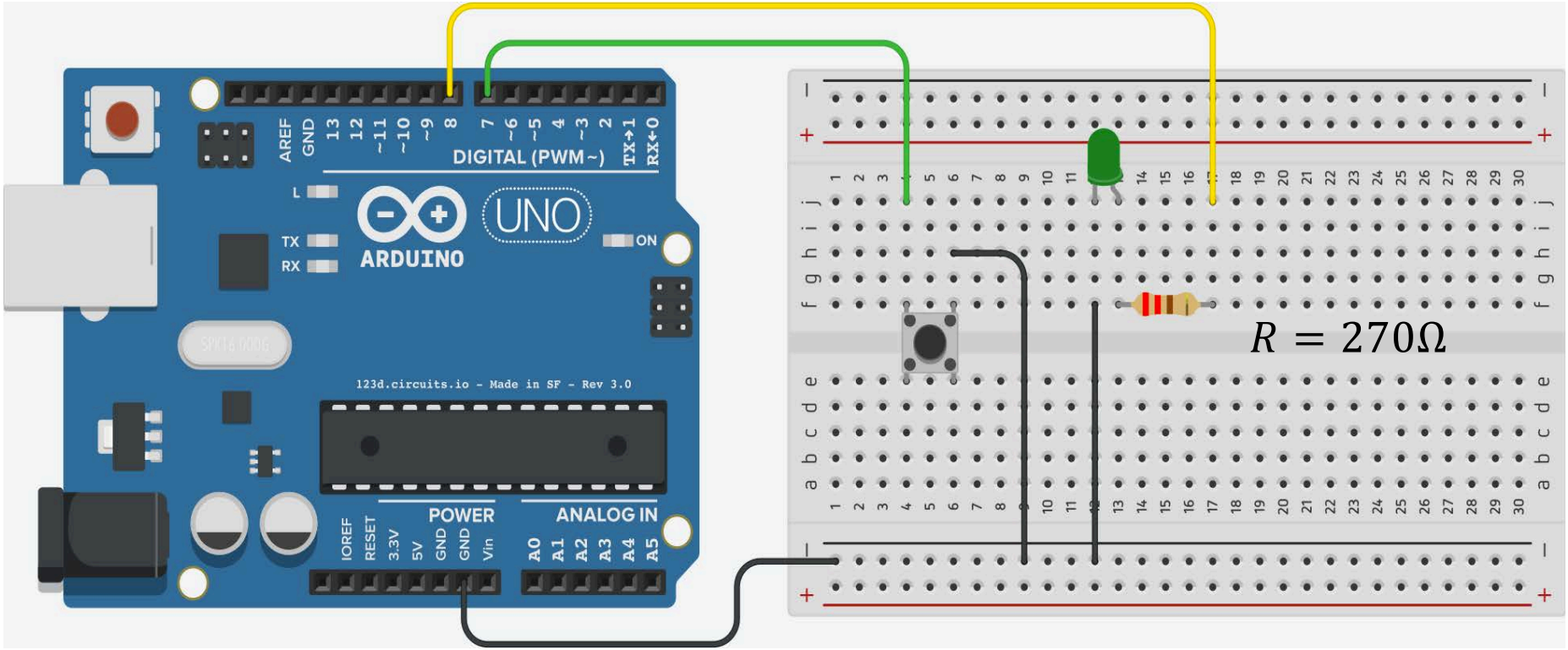
Avoid short circuit

fritzing

Example 3

Wiring

TRY IT OUT!



Note! In this configuration, we use an internal "pull-up" resistor to prevent "short-circuiting".

Example 3

Set pinMode like this: `pinMode(pin, INPUT_PULLUP);`

Arduino Program

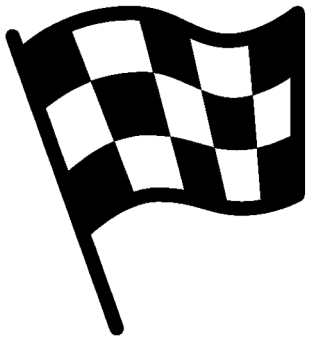
Example 3

```
const int buttonPin = 7;
const int ledPin =  8;

int buttonState = 0;

void setup()
{
    pinMode(ledPin, OUTPUT);

    pinMode(buttonPin, INPUT_PULLUP);
}

void loop()
{
  buttonState = digitalRead(buttonPin);


  if (buttonState == HIGH)
  {
        digitalWrite(ledPin, HIGH);
  }
  else
  {
        digitalWrite(ledPin, LOW);
  }

}
```

TRY IT OUT!

Play and Explore

|Programming with Arduino|



|Programming with Arduino|

Example 4

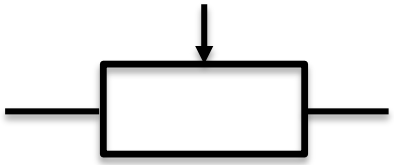
TRY IT OUT!

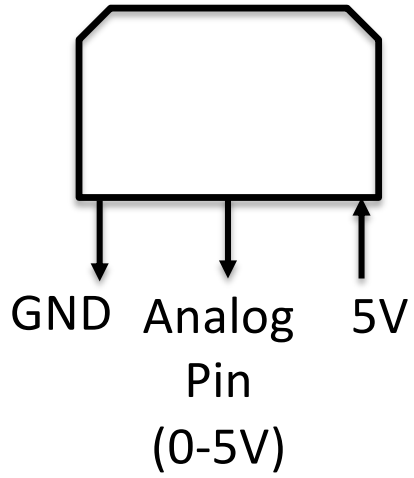
Potentiometer

Potentiometer

A potentiometer is a simple knob that provides a variable resistance, which we can read into the Arduino board as an analog value.

Electrical symbol:

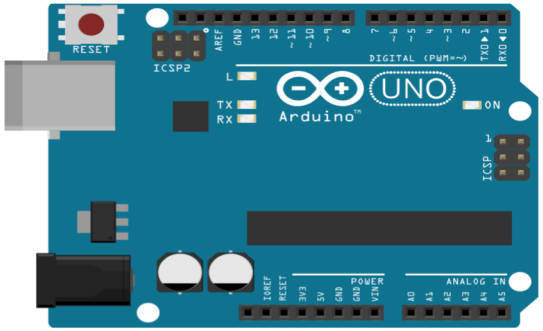


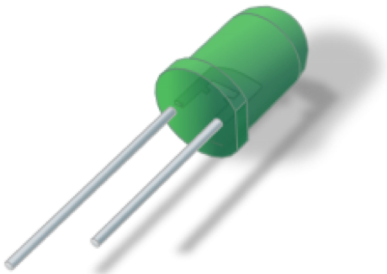


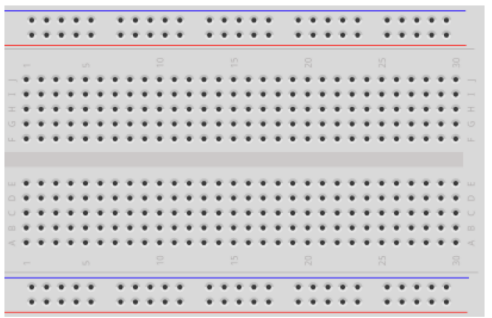
Example 4

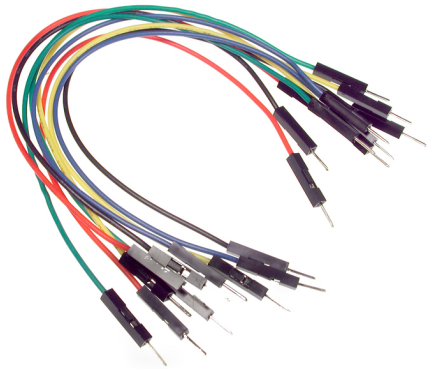
Equipment

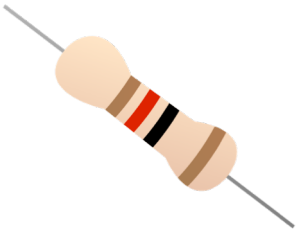
- Arduino
- Breadboard
- Potentiometer
- LED
- Resistor, $R = 330\Omega$
- Wires (Jumper Wires)





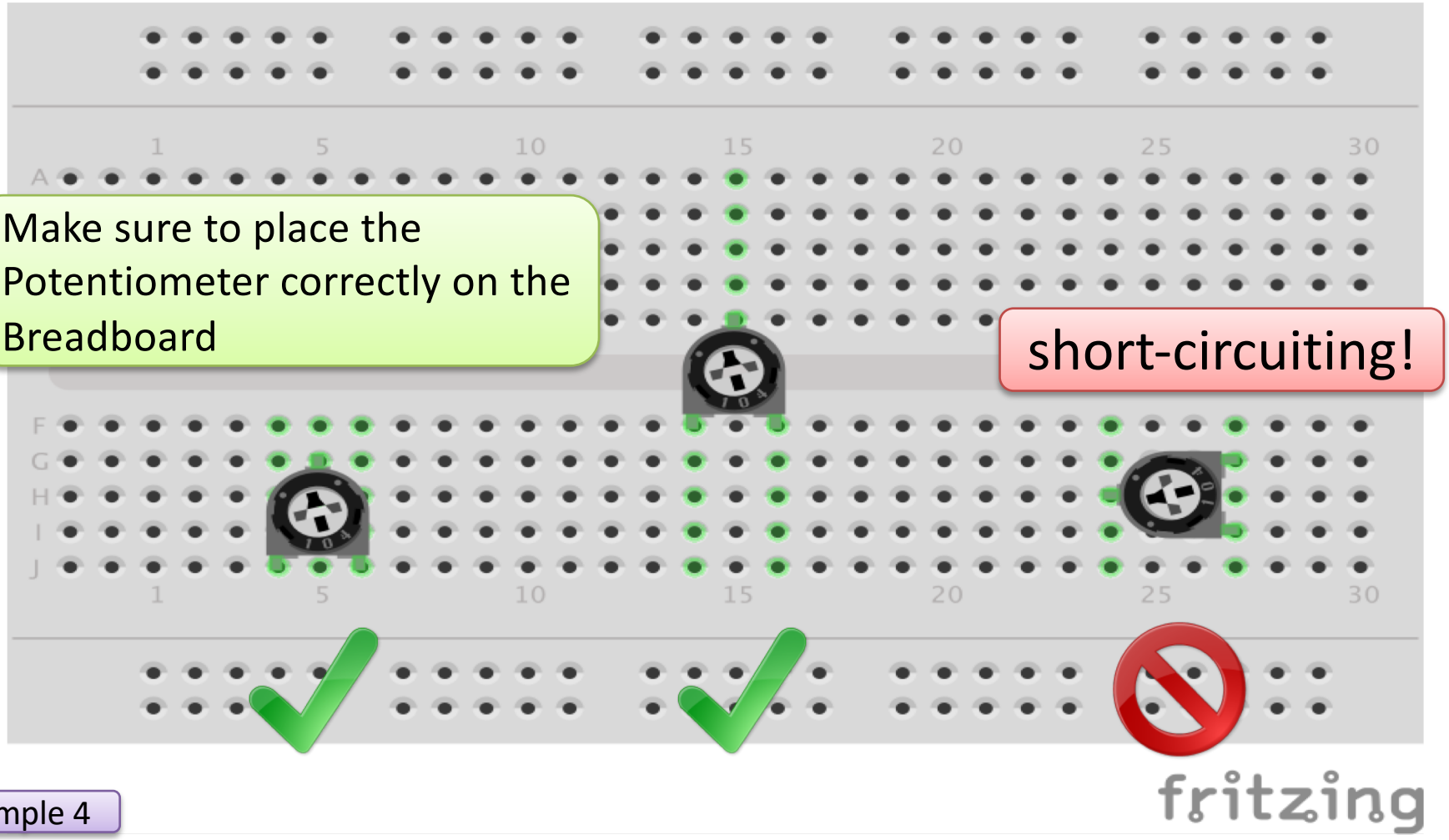






Example 4

Breadboard



Example 4

Dimmer

In this example we will make a simple dimmer using a potentiometer that control the intensity of the light.

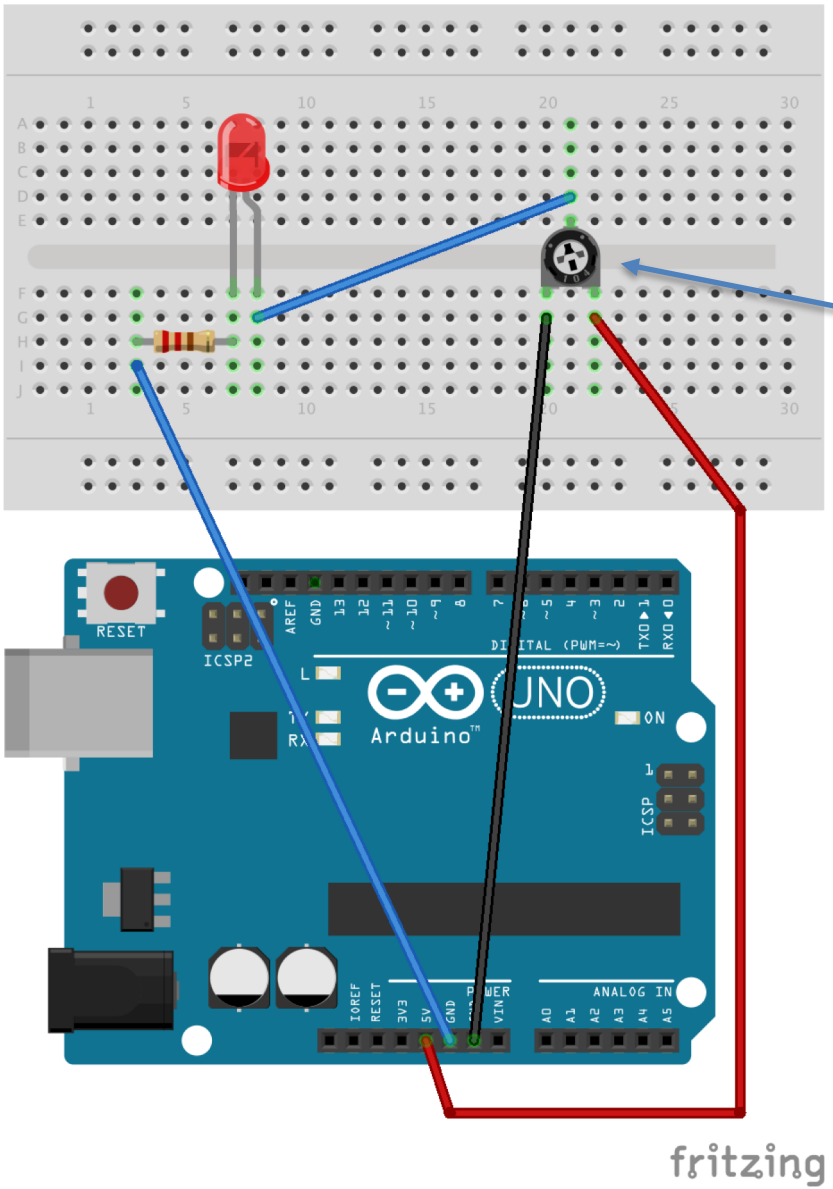
This is a typical example where a potentiometer is used. Everybody have a dimmer at home.

When the voltage in the circuit increases, the intensity of the LED will increase.

Example 4

Note! No Arduino Program is needed in this example

Wiring



TRY IT OUT!

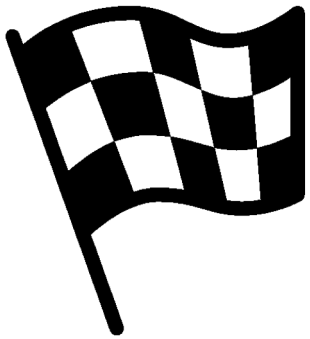
Adjust the the dimmer (potentiometer) in order to increase or decrease the intensity of the light (LED)

Note! No Arduino Program is needed in this example

Example 4

Play and Explore

|Programming with Arduino|



|Programming with Arduino|

Example 5

TRY IT OUT!

Temperature

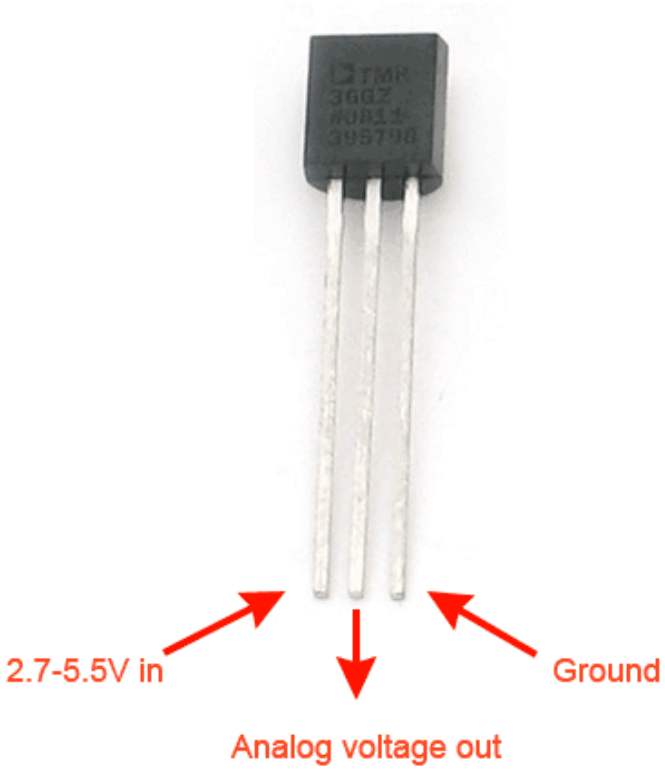
Introduction

In this example we will use a small temperature sensor to read the temperature in the room.

In this example we will use one of the "Analog In" ports on the Arduino board

Example 5

TMP36 Temperature Sensor



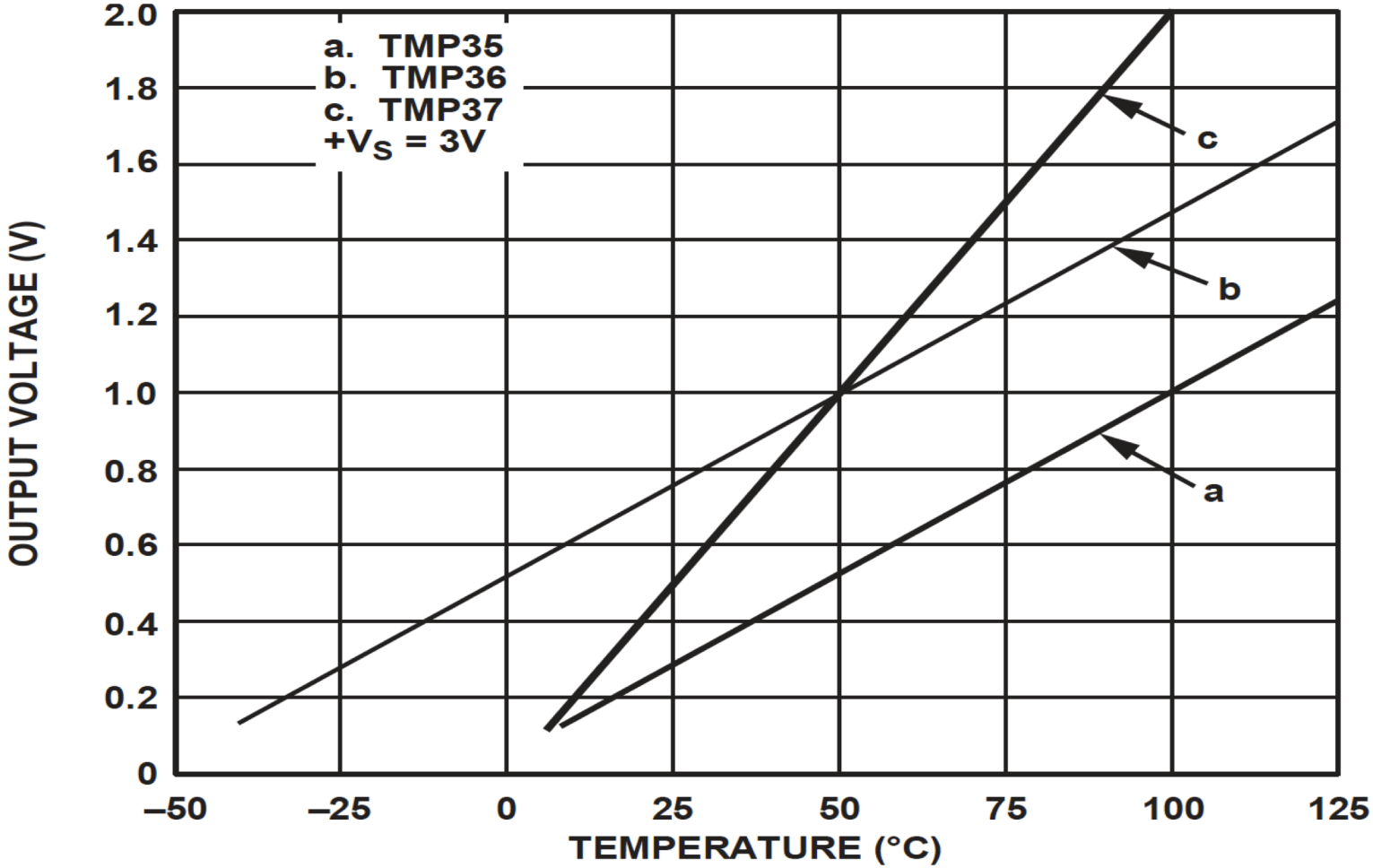
A Temperature sensor like TM36 use a solid-state technique to determine the temperature.

They use the fact as temperature increases, the voltage across a diode increases at a known rate.

Example 5

https://learn.adafruit.com/tmp36-temperature-sensor

Temperature Sensor Datasheet

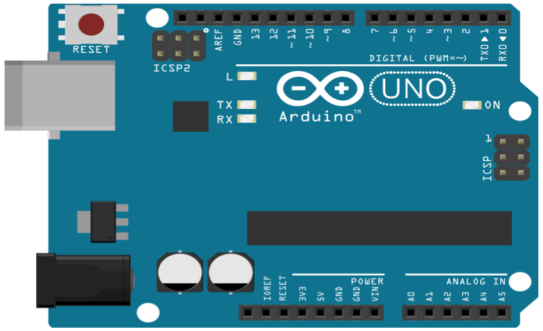


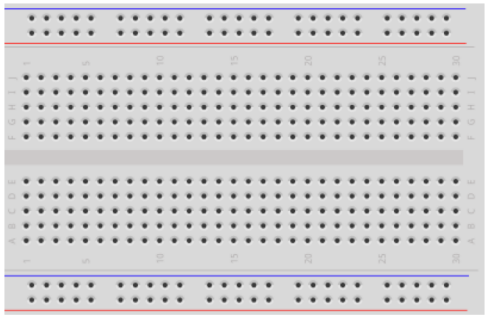
Example 5

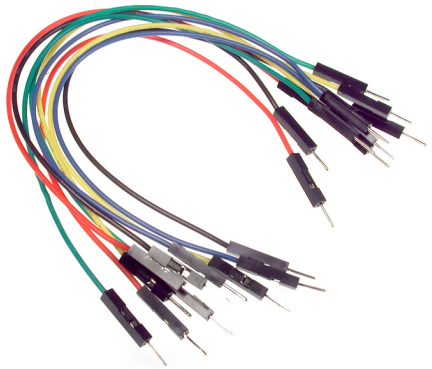
[http://no.rs-online.com/webdocs/14cd/0900766b814cd0a1.pdf]

Necessary Equipment

- Arduino
- Breadboard
- TMP36
- Wires (Jumper Wires)



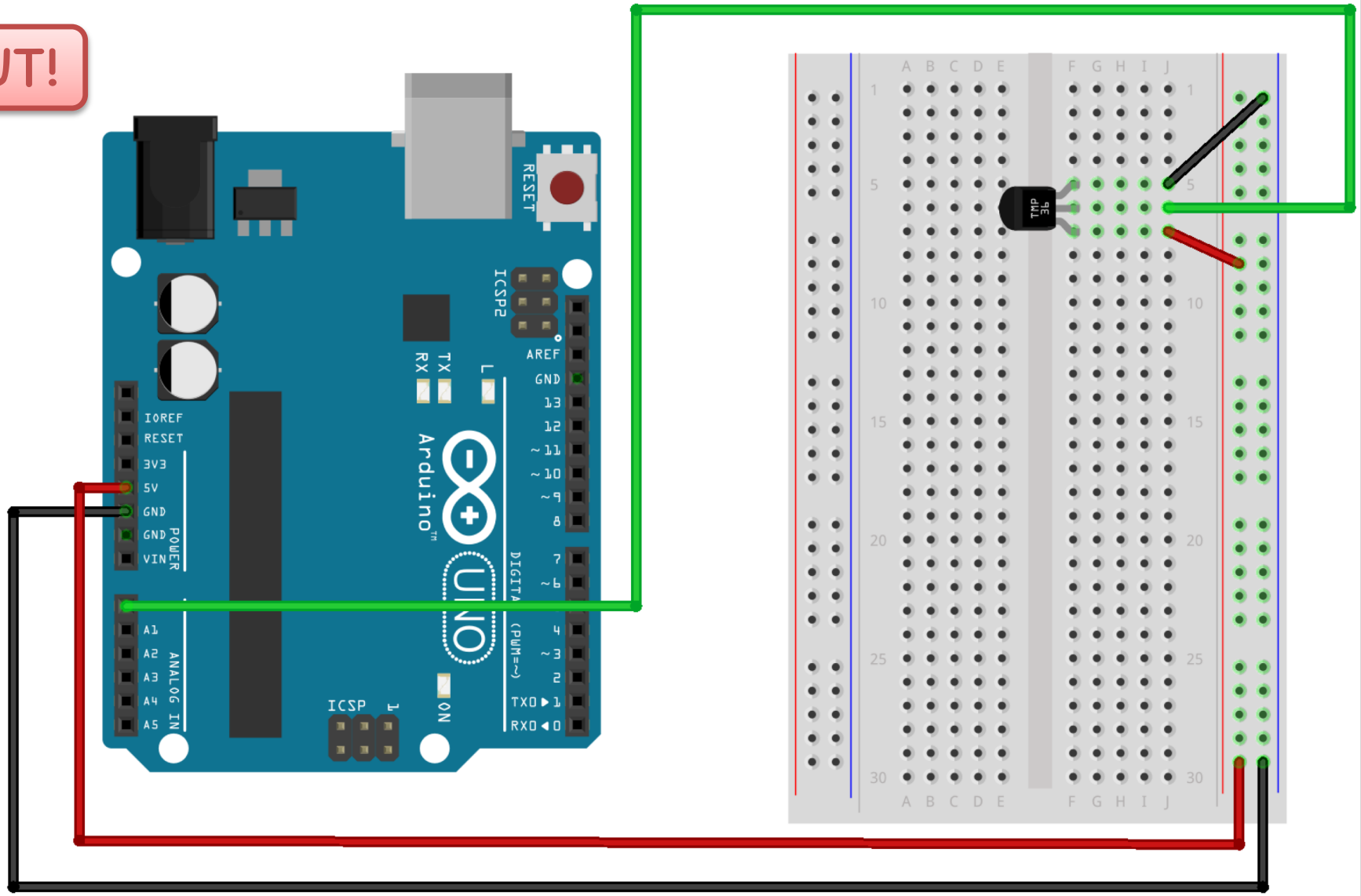




Example 5

TRY IT OUT!

Wiring



Example 5

fritzing

analogRead

analogRead reads the value from a specific analog pin.

The Arduino UNO board has 6 analog pins (channels), and uses a 10-bit analog to digital converter.

Syntax:

value = analogRead(analogPin);

value will then be between 0 and 1023

Example:

```
int sensorPin = 0;
int sensorValue;

void setup()
{

}

void loop()
{
  sensorValue = analogRead(sensorPin);
}
```

Example 5

https://www.arduino.cc/en/Reference/AnalogRead

Temperature coversion

We want to present the value from the sensor in degrees Celsius:

1. analogRead() gives a value between 0 and 1023
2. Then we convert this value to 0-5V
3. Finally, we convert to degrees Celsius using information from the Datasheet presented on the previous page

Example 5

Arduino Program

Example 5

```
const int temperaturePin = 0;

float adcValue;
float voltage;
float degreesC;

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

void loop()
{

  adcValue = analogRead(temperaturePin);

  voltage = (adcValue*5)/1023;

  degreesC = 100*voltage - 50;

  Serial.print("ADC Value: ");
  Serial.print(adcValue);

  Serial.print("  voltage: ");
  Serial.print(voltage);

  Serial.print("  deg C: ");
  Serial.println(degreesC);

  delay(1000);
}
```

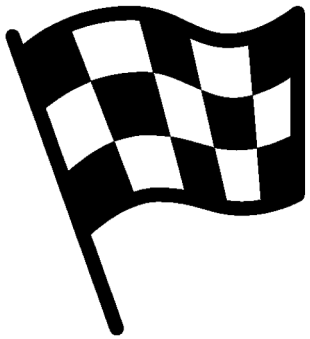
TRY IT OUT!

Convert from ADC-value (0-1023) to Voltage (0-5V)

Convert from Volate to degrees Celsius

Play and Explore

|Programming with Arduino|

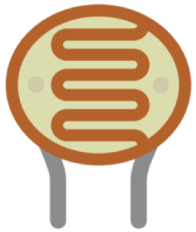


|Programming with Arduino|

Example 6

TRY IT OUT!

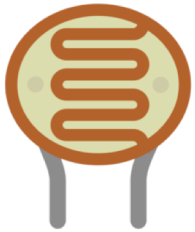
Light Sensor



Introduction

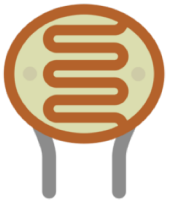
In this example we will use a light sensor to measure the light intensity of the room.

If it's dark, we will turn on the light (LED)
If it's bright, we'll turn off the light (LED)



Example 6

Light Sensor



or

Light sensor, Photocell (Photo resistor), LDR (light-dependent resistor)

A light sensor / photocell is a sensor used to detect light.

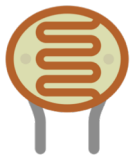
The resistance decreases with increasing light intensity (stronger light).

Example 6

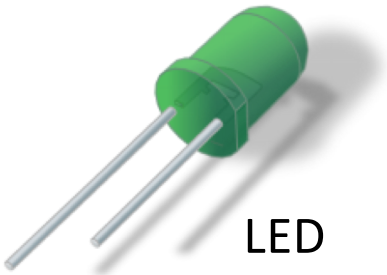
Necessary Equipment

- Arduino
- Breadboard
- Light Sensor
- LED
- Resistors, $R = 330\Omega$, $R = 10\ k\Omega$
- Wires (Jumper Wires)

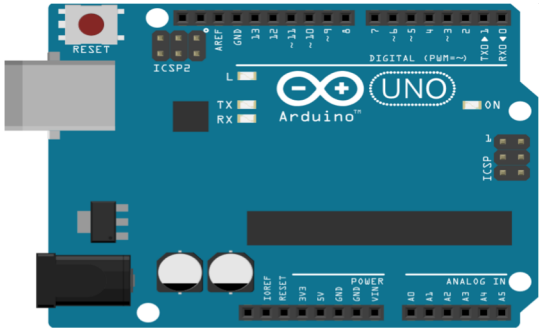
Light Sensor

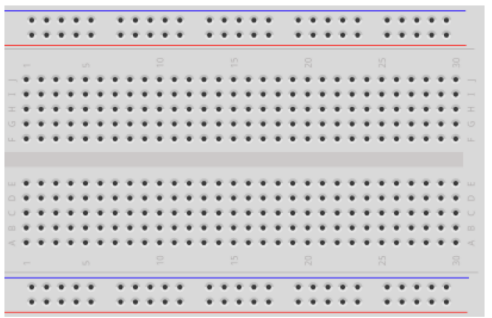


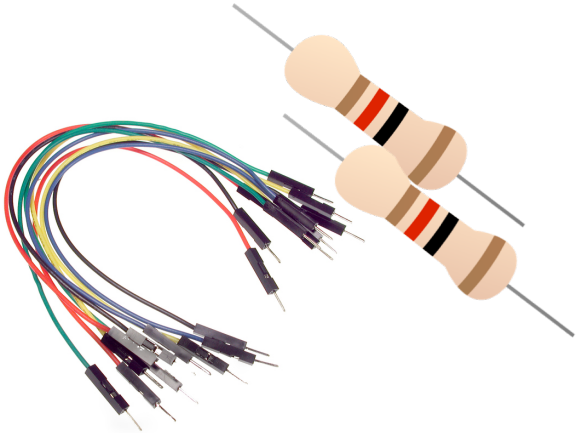
or



LED

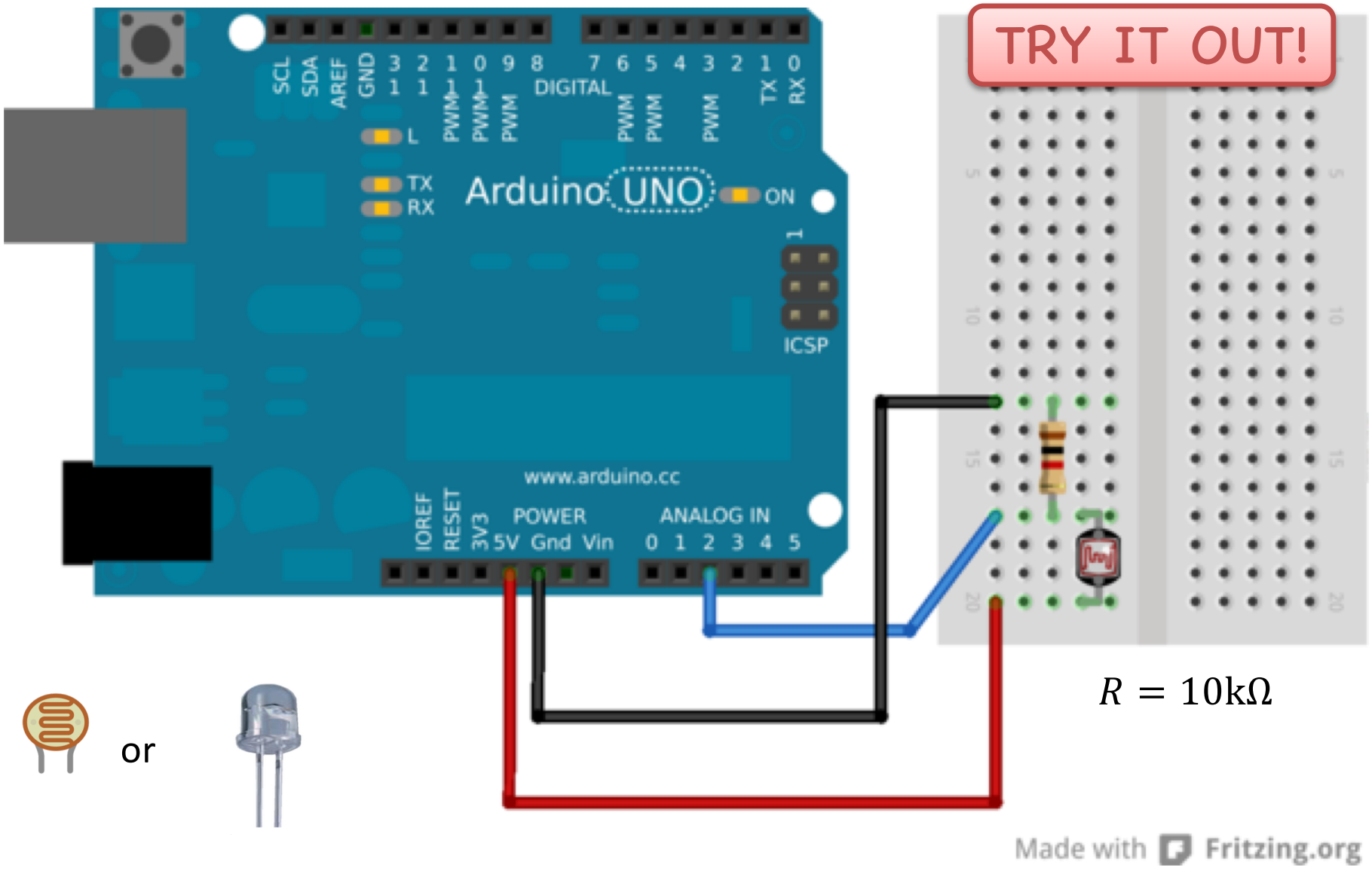






Example 6

Wiring



Example 6

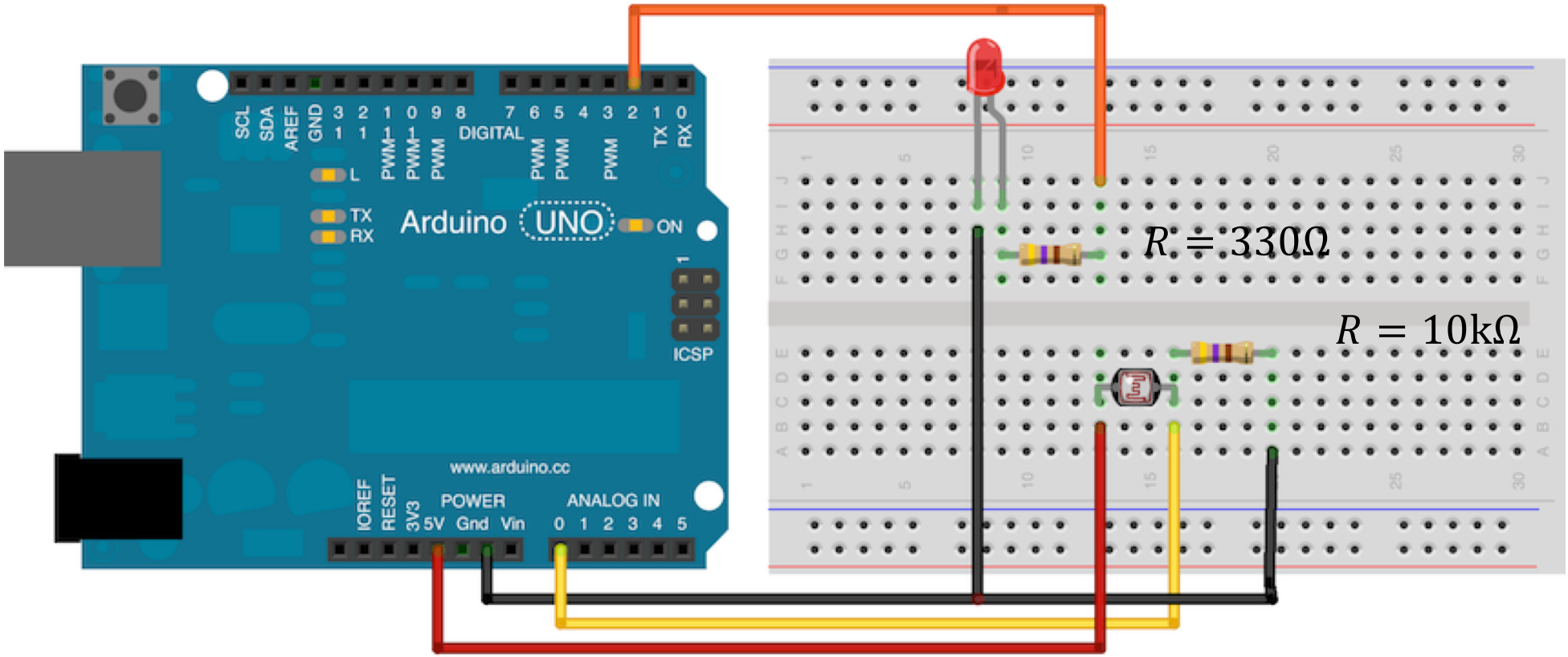
Arduino Program

```
int photocellPin = 2;
int photocellReading;

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

void loop(void)
{
    photocellReading = analogRead(photocellPin);

    Serial.print("Analog reading = ");
    Serial.println(photocellReading);

    delay(1000);
}
```

TRY IT OUT!

Example 6

Wiring



Made with Fritzing.org

or

Example 6

Arduino Program

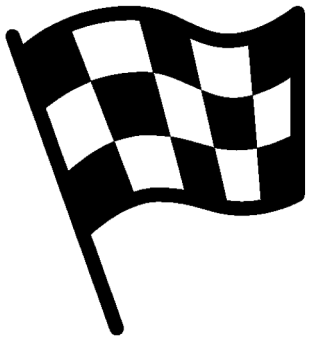
Example 6

```
int photocellPin = 0;
int ledPin = 2;
int photocellReading;
const float limit = 100;

void setup(void)
{
      Serial.begin(9600);
      pinMode(ledPin, OUTPUT);
}

void loop(void)
{
      photocellReading = analogRead(photocellPin);
      Serial.print("Analog reading = ");
      Serial.println(photocellReading);

      if (photocellReading < limit)
      {
            digitalWrite(ledPin, HIGH);
      }
      else
      {
            digitalWrite(ledPin, LOW);
      }

      delay(1000);

}
```

TRY IT OUT!

Play and Explore

|Programming with Arduino|



|Programming with Arduino|

Example 7

TRY IT OUT!

Thermistor

Introduction

In this example we will use a small thermistor to read the temperature in the room.

In this example we will use one of the "Analogue Inn" ports at Arduino.

Example 7

Thermistor

A thermistor is an electronic component that changes resistance to temperature - so-called Resistance Temperature Detectors (RTD). It is often used as a temperature sensor.

Our Thermistor is a so-called NTC (Negative Temperature Coefficient). In a NTC Thermistor, resistance decreases as the temperature rises.

There is an non-linear relationship between resistance and excitement. To find the temperature we can use the following equation (Steinhart-Hart equation):

$$\frac{1}{T} = A + B\ln(R) + C(\ln(R))^3$$

[Wikipedia]

where A, B, C are constants given below

$A = 0.001129148, B = 0.000234125\ and\ C = 8.76741E-08$

Example 7

NTC Thermistor - Datasheet

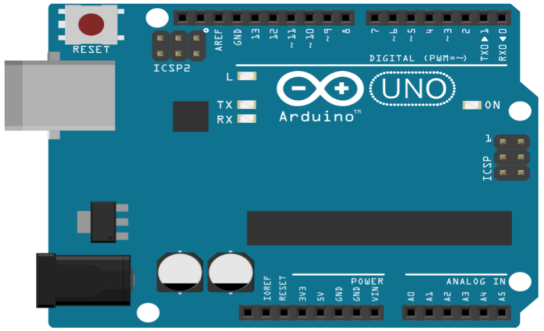
Technical data	
Resistance @ 25°C	10 kΩ
Temperature range	-40...+125 °C
Power max.	500 mW
Pitch	2.54 mm
Resistance tolerance	±5 %
$W_{25/100}$ value	3977 K
B value tolerance	±0.75 %
Thermal time constant	15 s

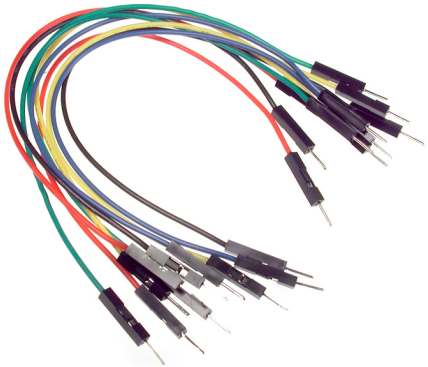
Datasheet: https://www.elfadistrelec.no/no/ntc-motstand-kablet-10-kohm-vishay-ntcle100e3103jb0/p/16026041?q=160-26-041&page=1&origPos=1&origPageSize=50&simi=98.0

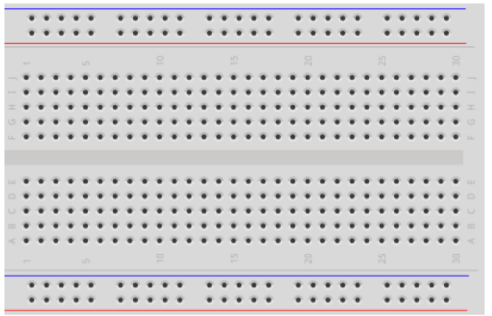
Example 7

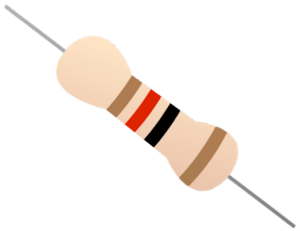
Equipment

- Arduino
- Breadboard
- Thermistor
- LED
- Resistor 10 kΩ
- Wires (Jumper Wires)





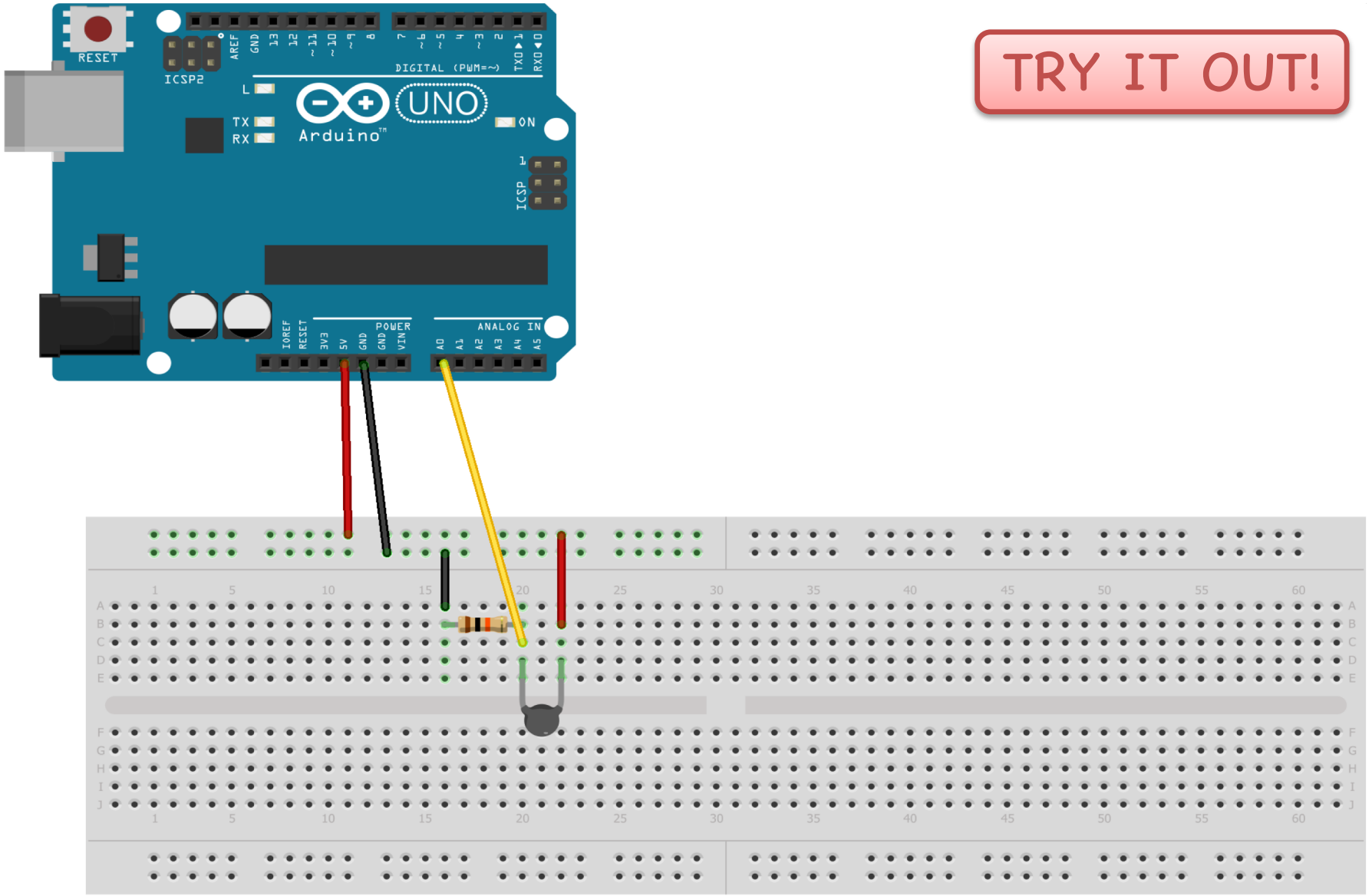




Example 7

Wiring

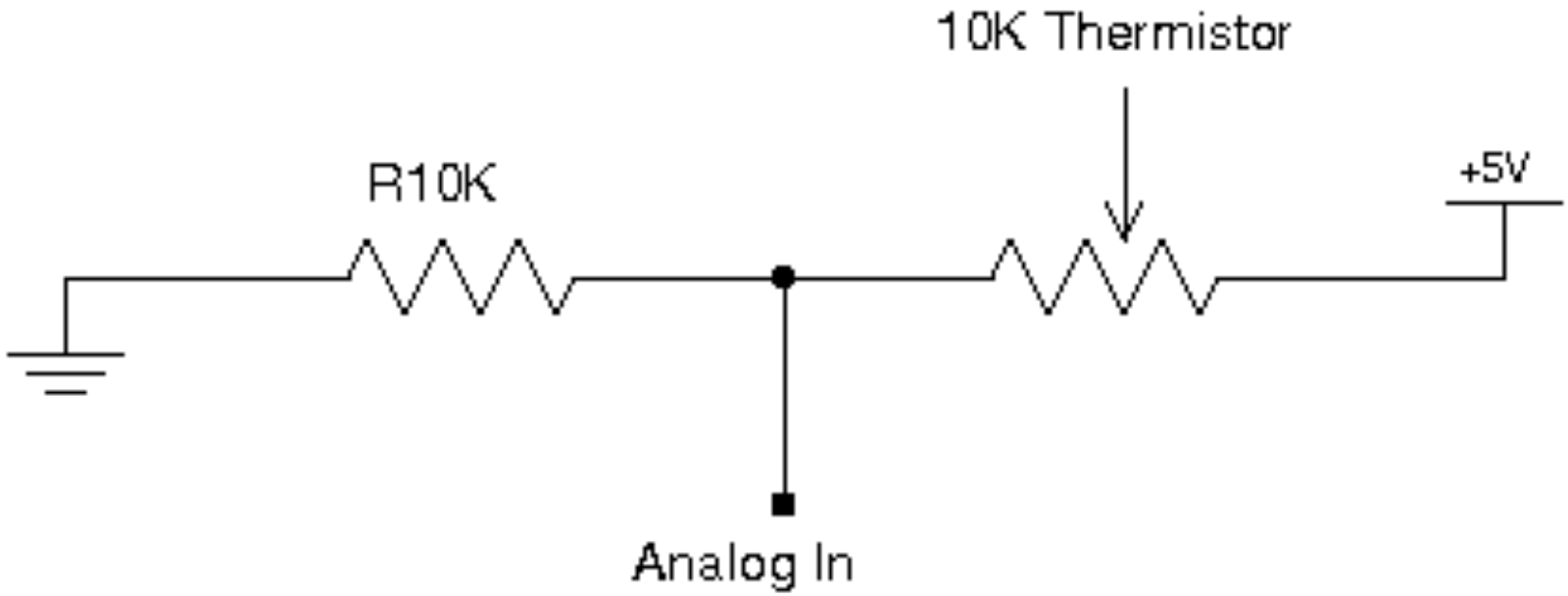
TRY IT OUT!



Example 7

Wiring

The wiring is called a Voltage divider:



[https://en.wikipedia.org/wiki/Voltage_divider]

Arduino Program

Example 7

TRY IT OUT!

```
const int temperaturePin = 0;

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

void loop()
{
  int temperature = getTemp();
  Serial.print("Temperature Value: ");
  Serial.print(temperature);
  Serial.println("*C");
  delay(1000);
}

double getTemp()
{
  // Inputs ADC Value from Thermistor and outputs Temperature in Celsius

  int RawADC = analogRead(temperaturePin);
  long Resistance;
  double Temp;

  // Assuming a 10k Thermistor.  Calculation is actually: Resistance = (1024/ADC)
  Resistance=((10240000/RawADC) - 10000);

  // Utilizes the Steinhart-Hart Thermistor Equation:
  // Temperature in Kelvin = 1 / {A + B[ln(R)] + C[ln(R)]^3}
  // where A = 0.001129148, B = 0.000234125 and C = 8.76741E-08

  Temp = log(Resistance);
  Temp = 1 / (0.001129148 + (0.000234125 * Temp) + (0.0000000876741 * Temp * Temp * Temp));
  Temp = Temp - 273.15;  // Convert Kelvin to Celsius
  return Temp;  // Return the Temperature
}
```

Additional Exercises

TRY IT OUT!

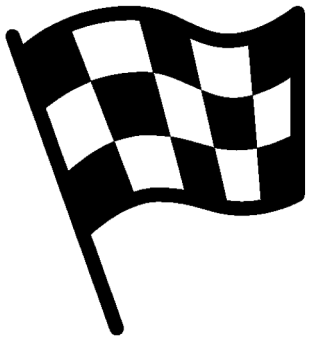
Try out the following:

- Use both the TMP36 Temperature Sensor we used earlier and the Thermistor.
- Compare the values from the Thermistor with the TMP36 Temperature Sensor we used earlier. Display both values in the Serial Monitor.
- Draw circuit and wiring using the Fritzing software

Example 7

Play and Explore

|Programming with Arduino|



|Slutt på eksemplet|

Play and Explore

|Programming with Arduino|

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: https://www.halvorsen.blog