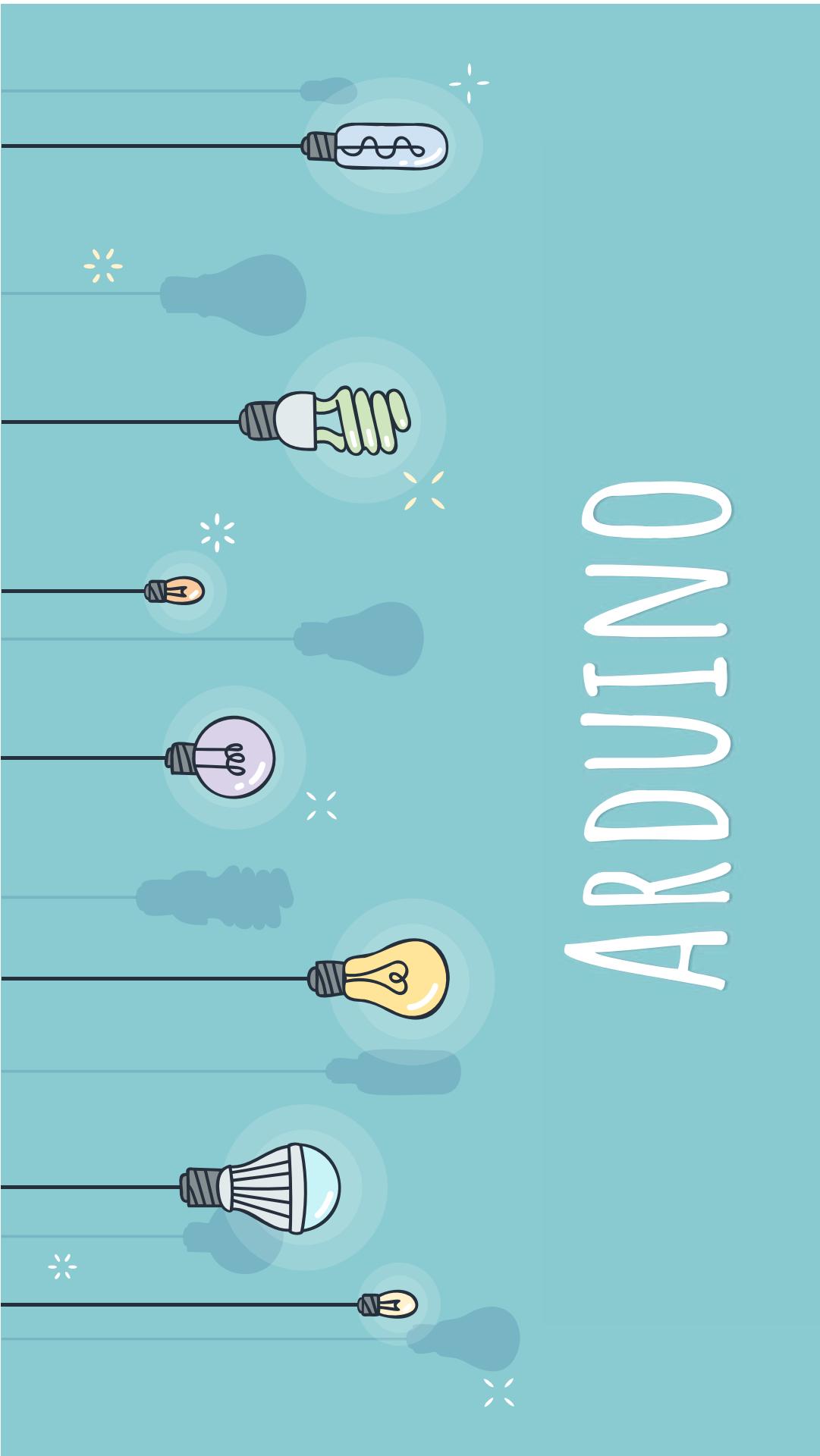


ARDUINO



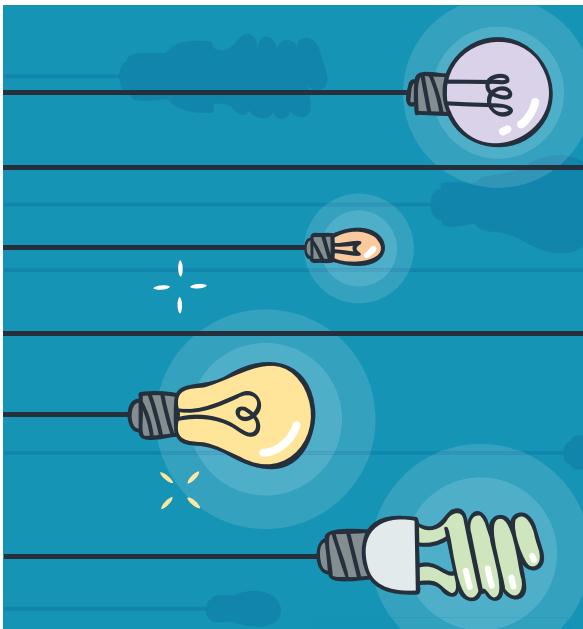
ARDUINO IS...

1

OPEN SOURCE HARDWARE AND SOFTWARE

OPEN SOURCE : anyone can USE it or
IMPLEMENT it.

(Source code and circuit diagram is publicly
available)

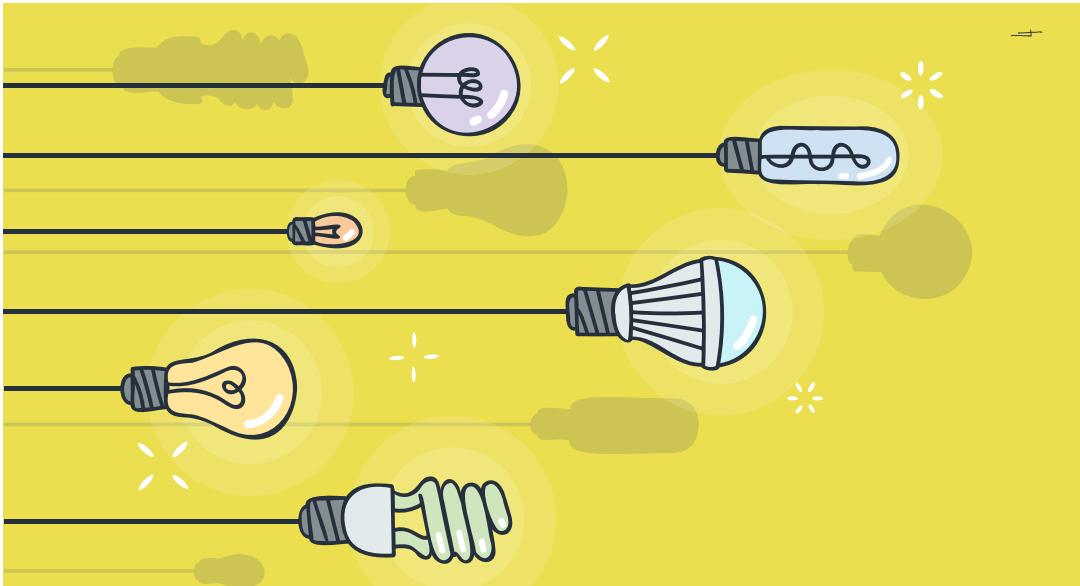
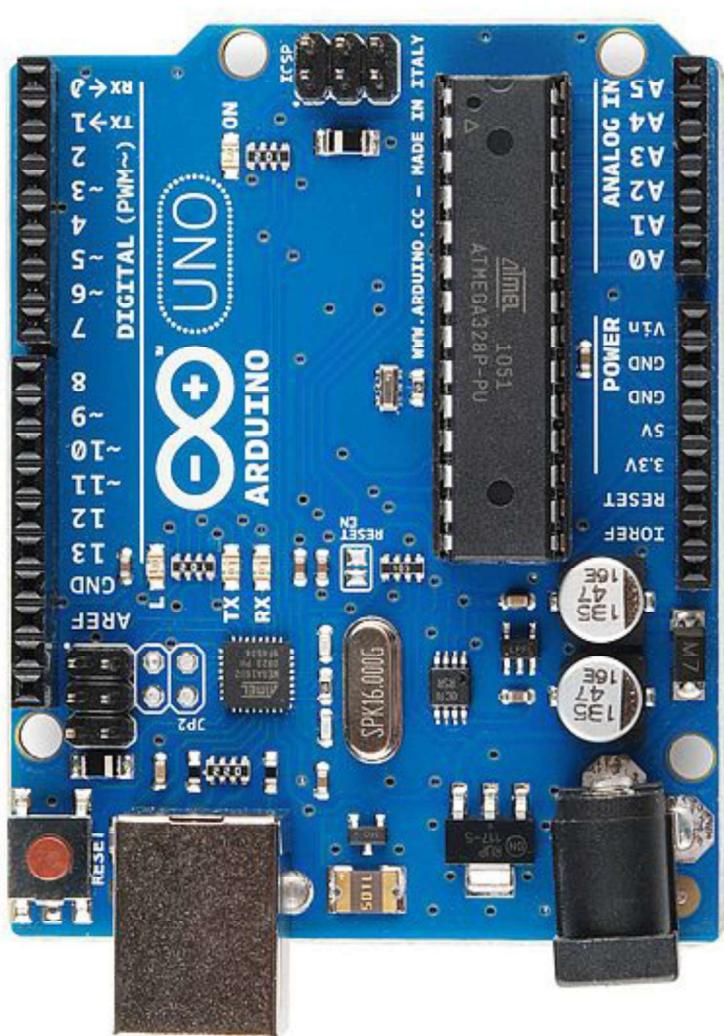


TYPES OF ARDUINO



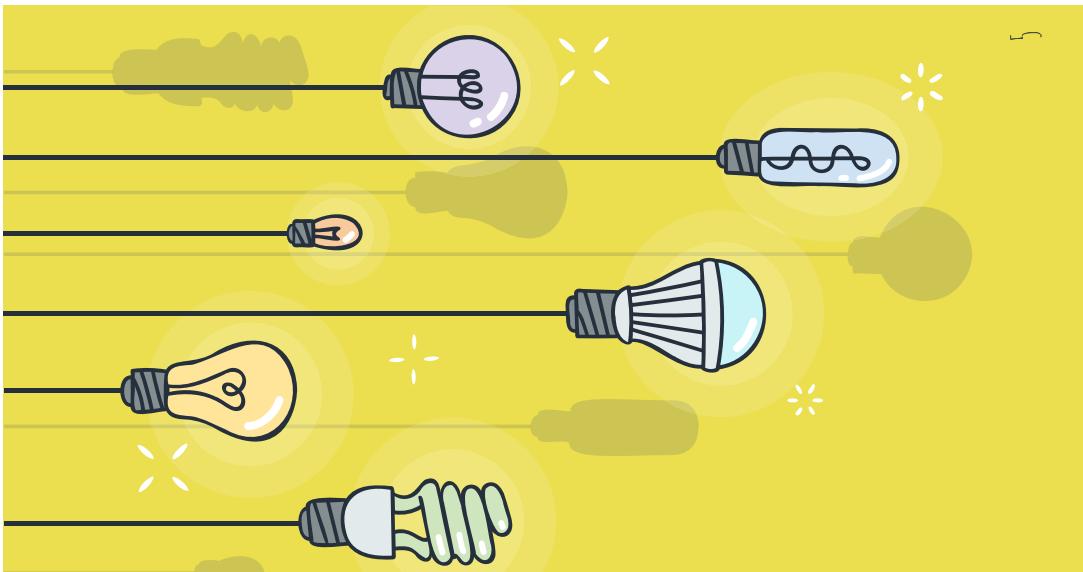
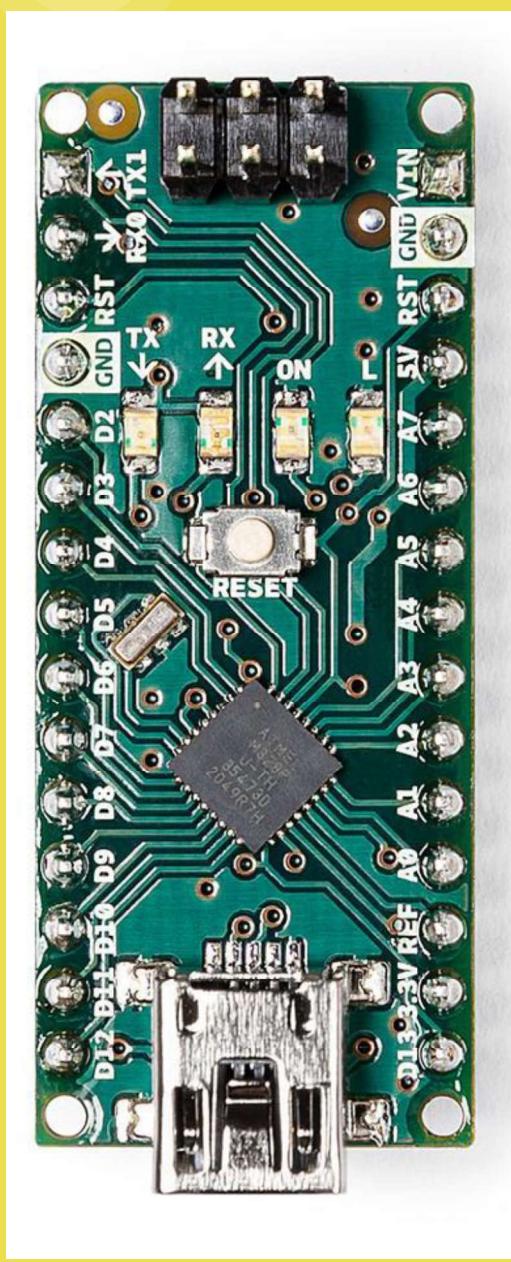
Arduino Uno

“



Arduino Nano

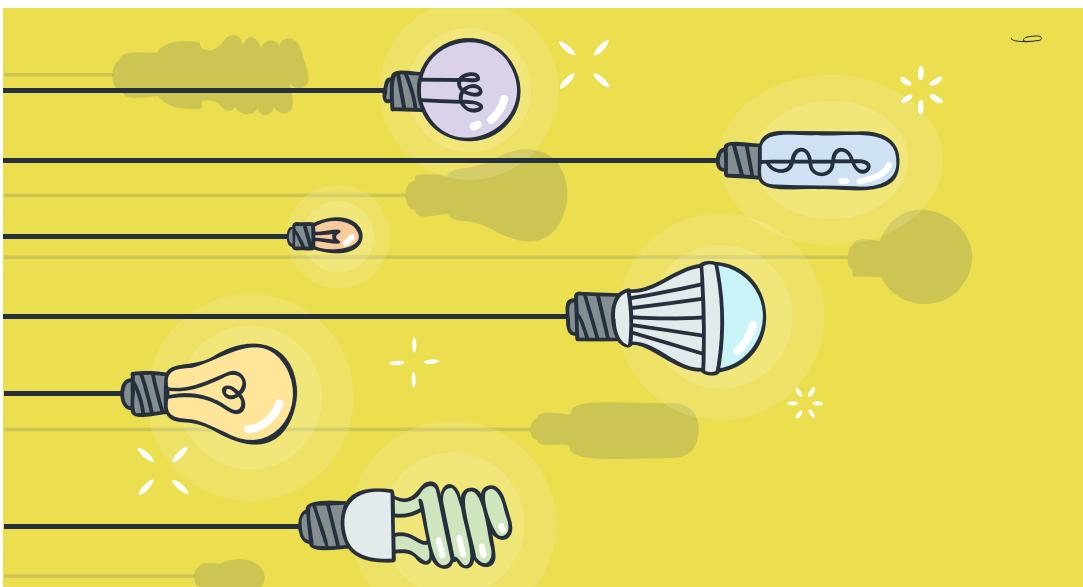
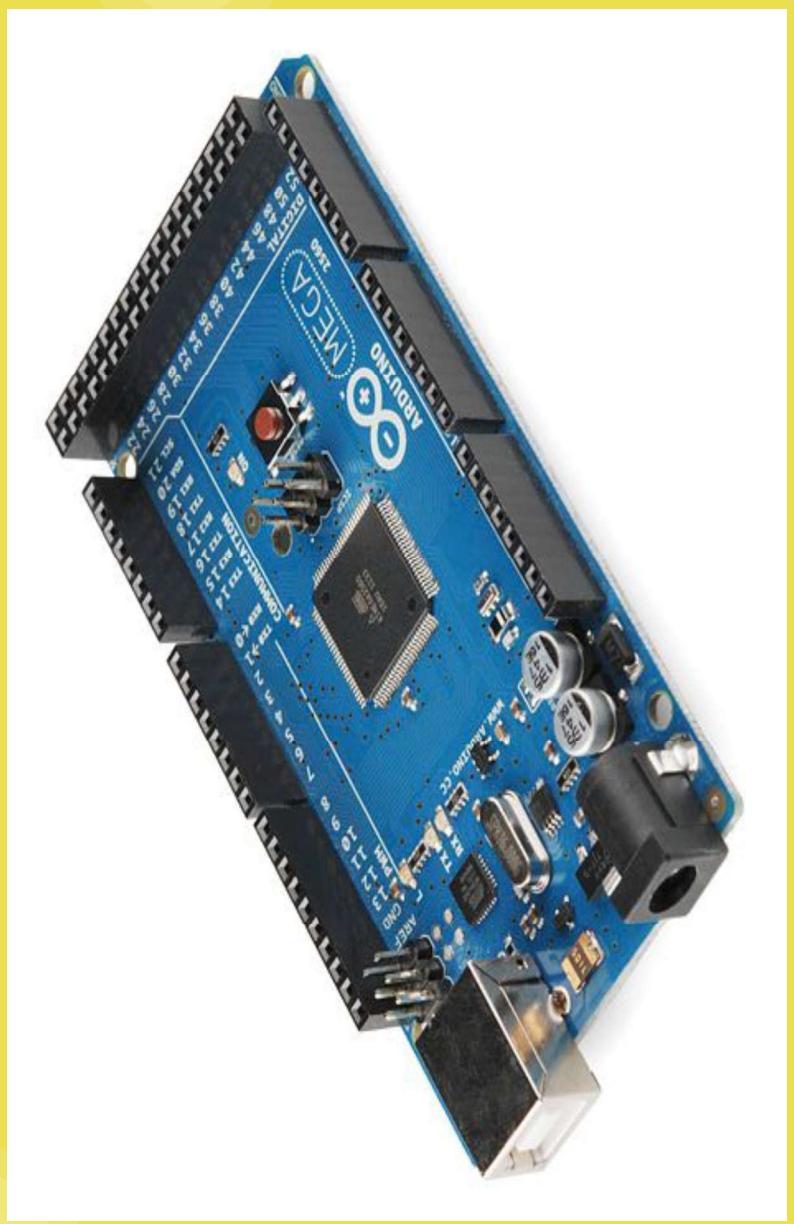
“



5

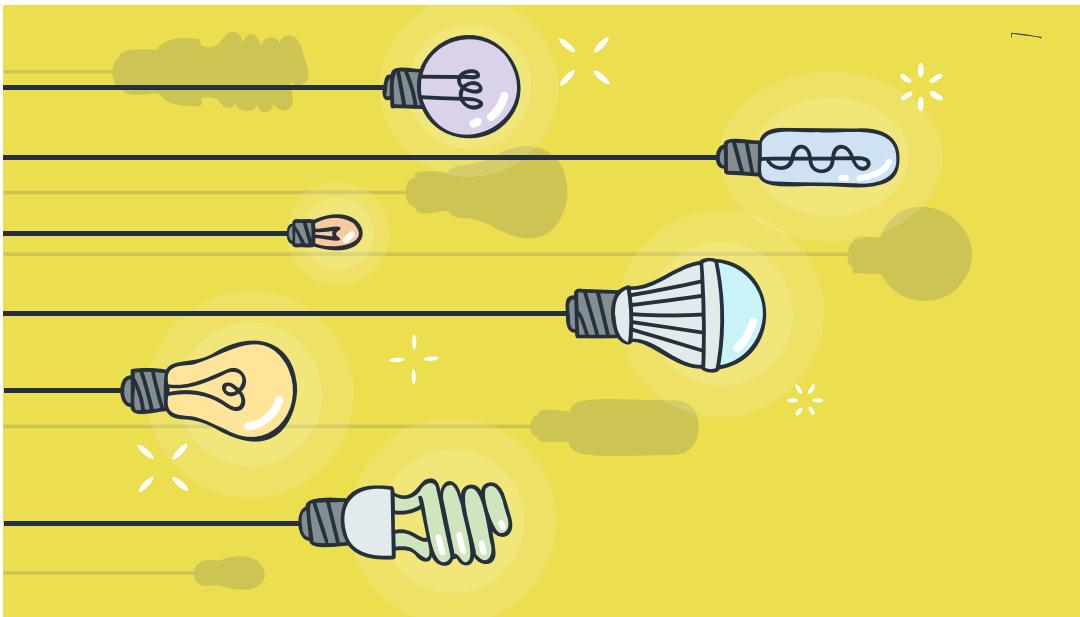
Arduino Mega

“

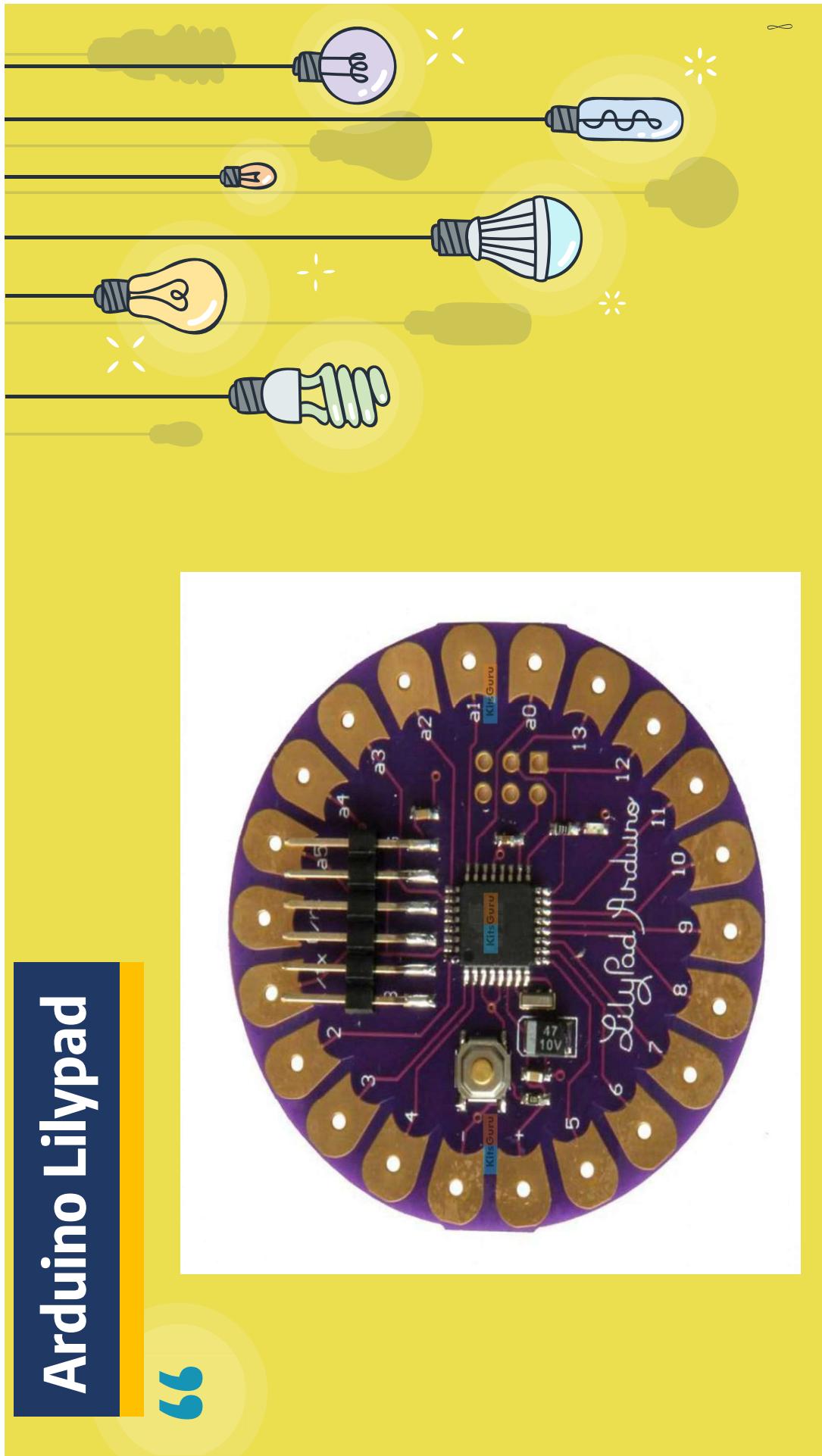


Arduino Due

“



1



**WHY SO MANY
ARDUINOS?**

REASON FOR THIS IS

1. Size and Weight
2. Number of pins
3. RAM and ROM
4. Communication protocols

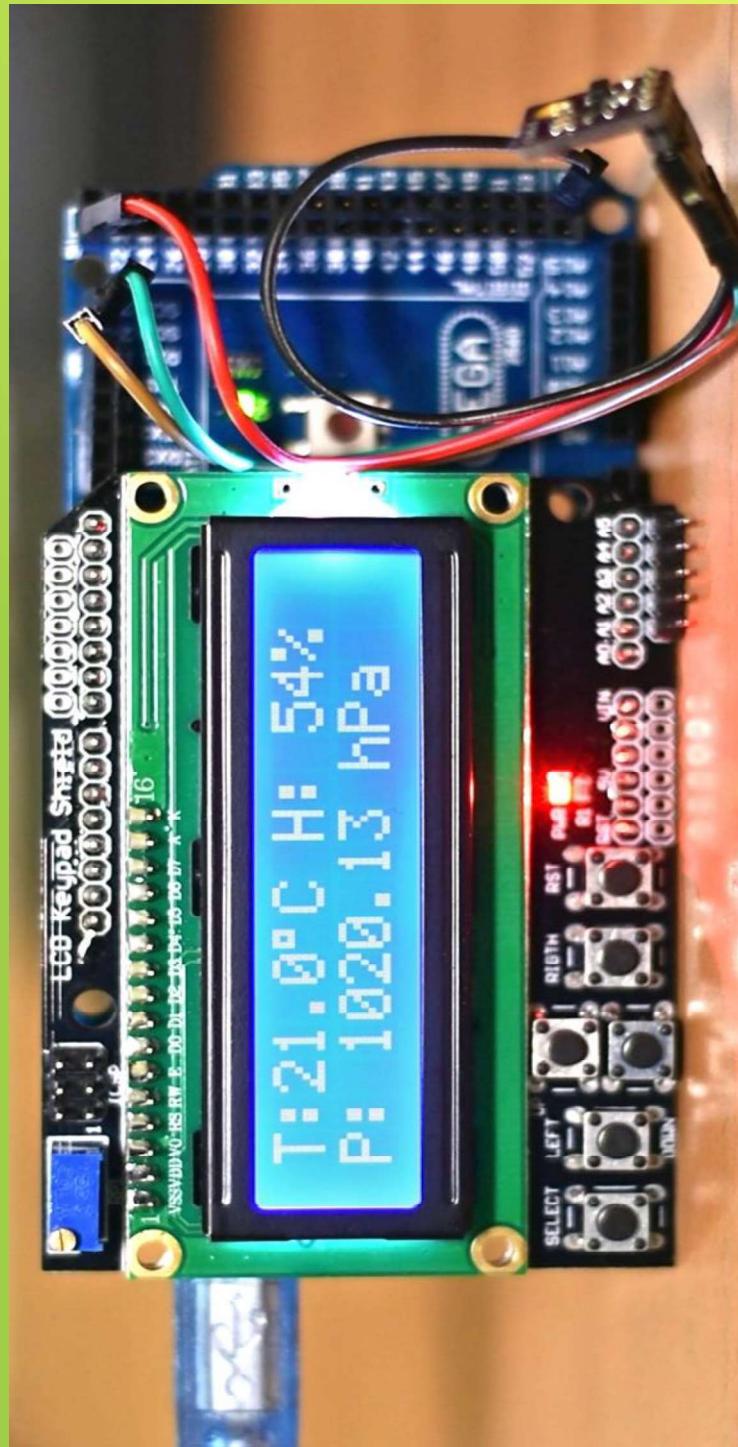
HOW THE ARDUINO ARE USED...



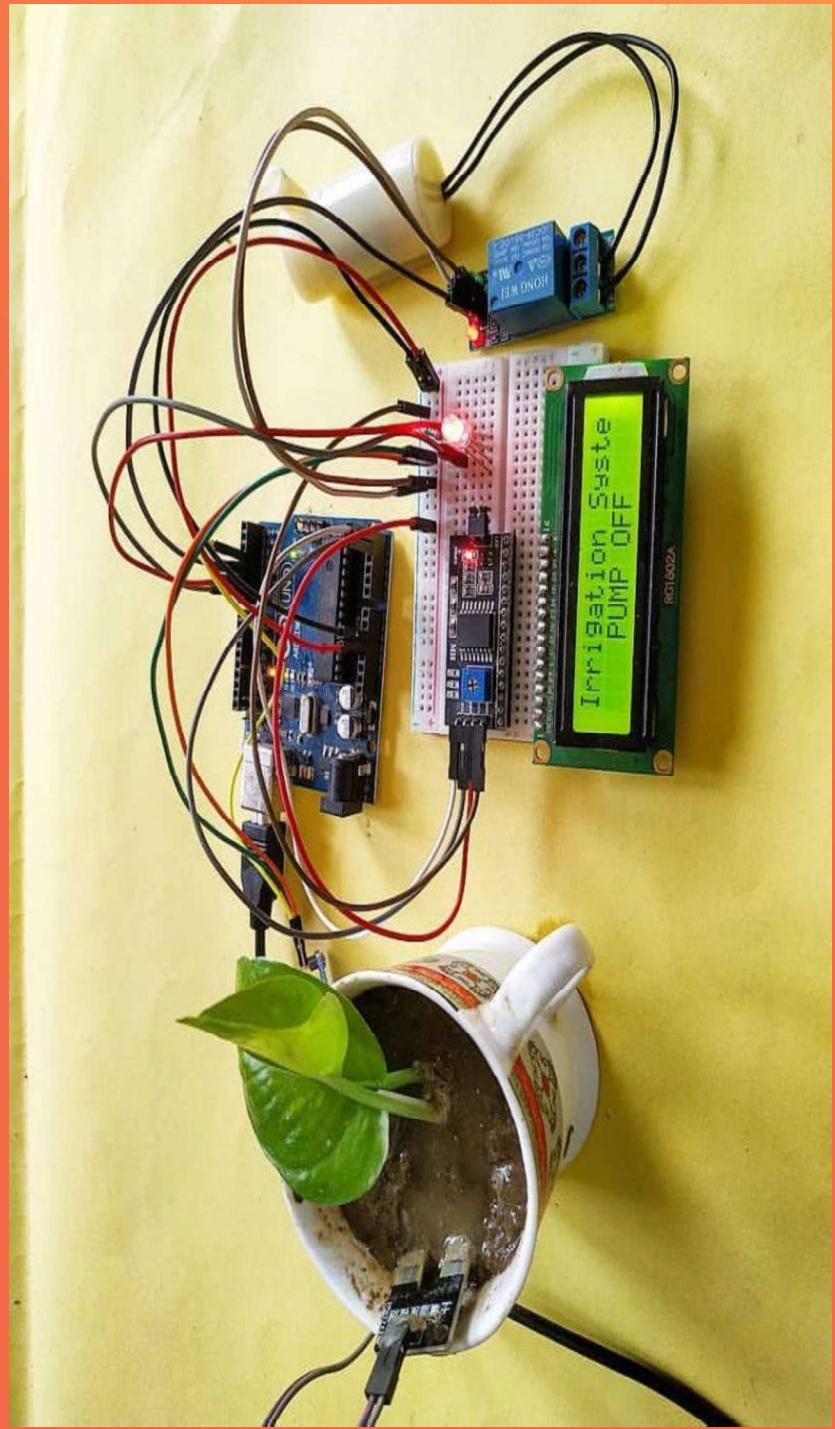
Bluetooth controlled car



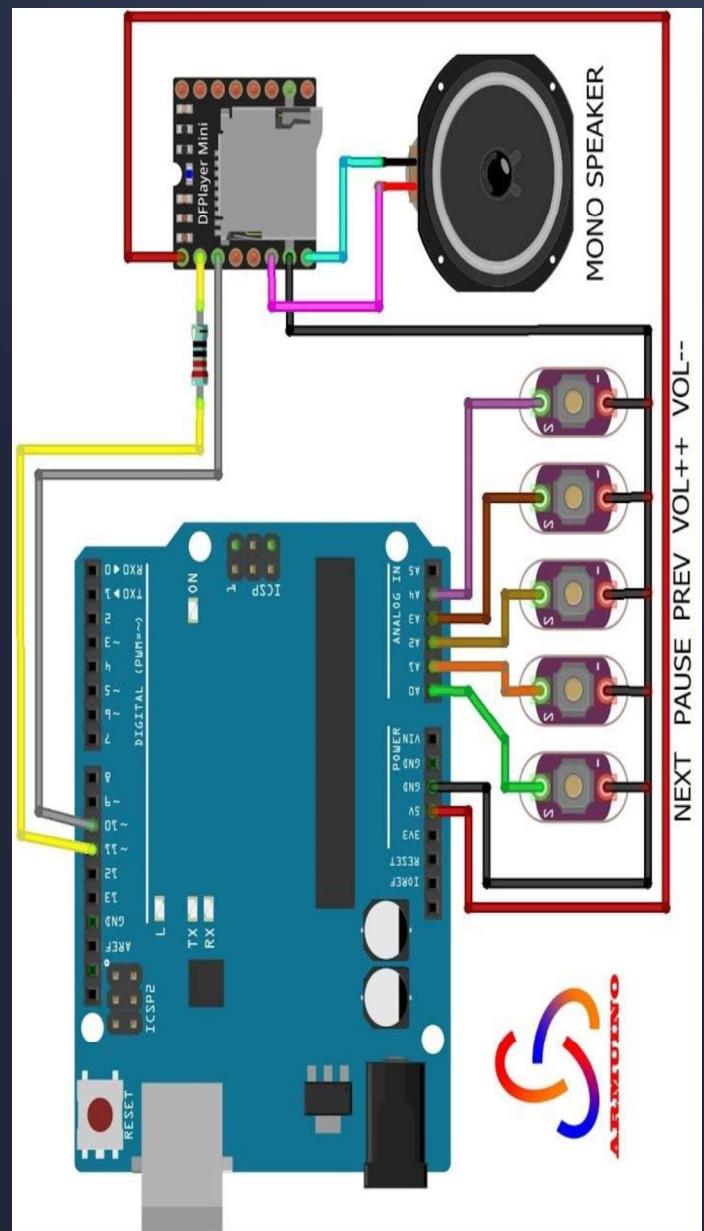
WEATHER MONITORING STATION



SMART AGRICULTURE SYSTEM

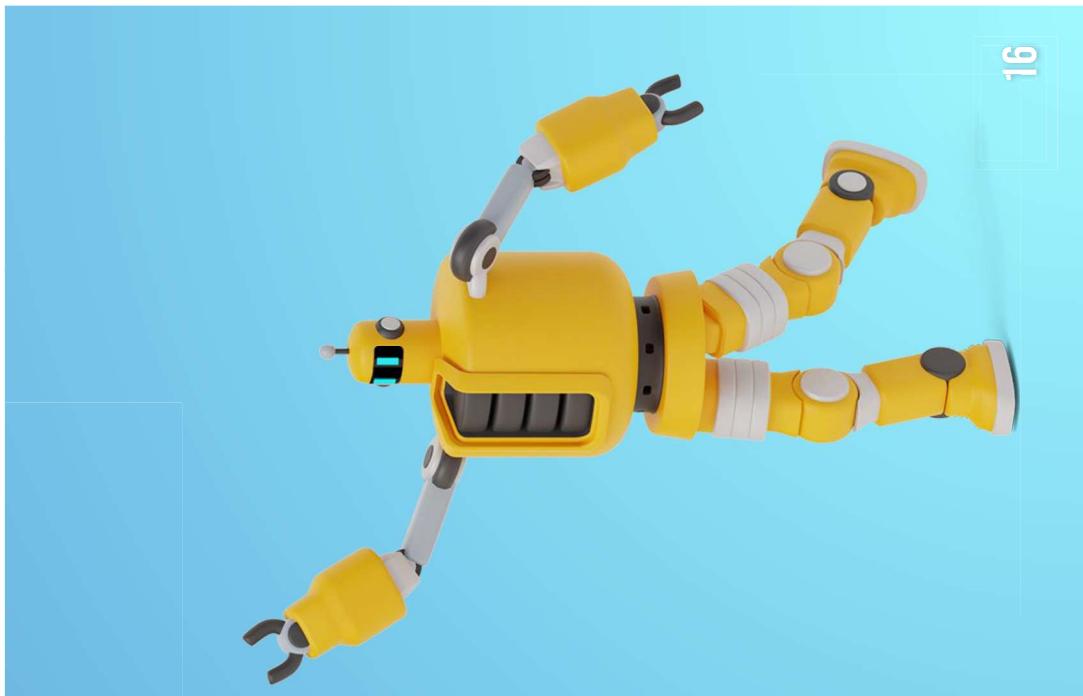


MP3 PLAYER



5

HAND GESTURE ROBOT



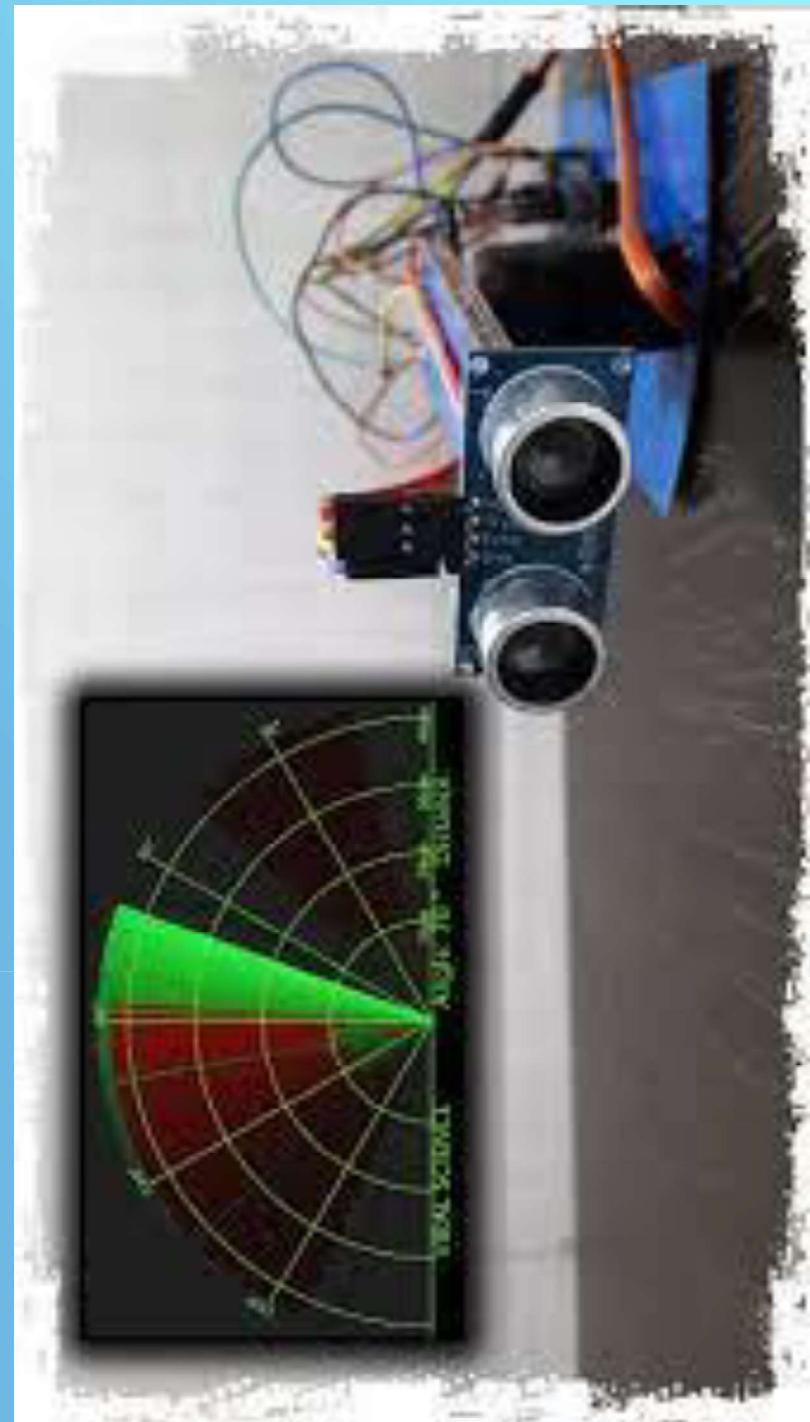
16

SMART DUSTBIN

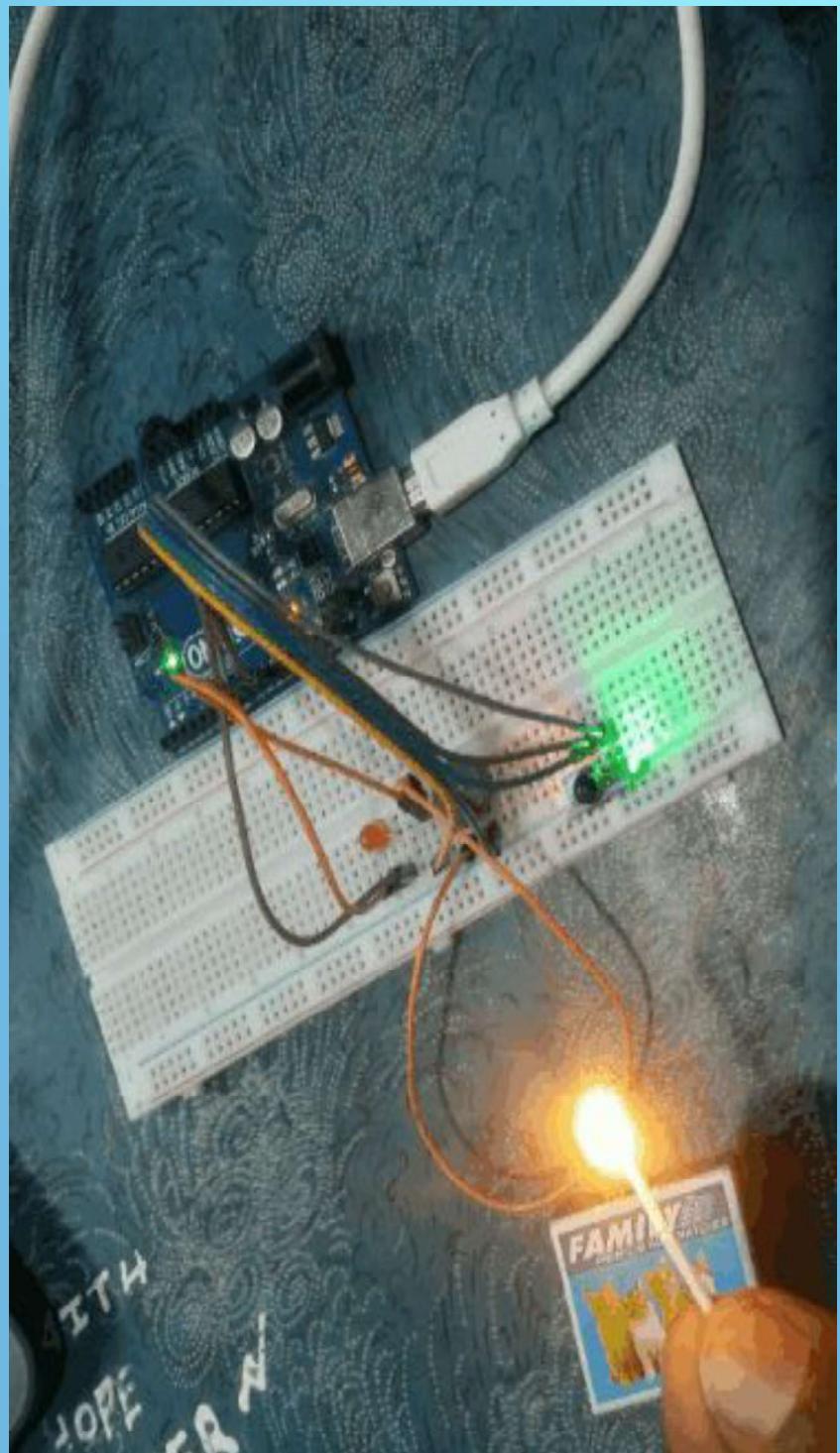
Arduino Project

Smart Dustbin

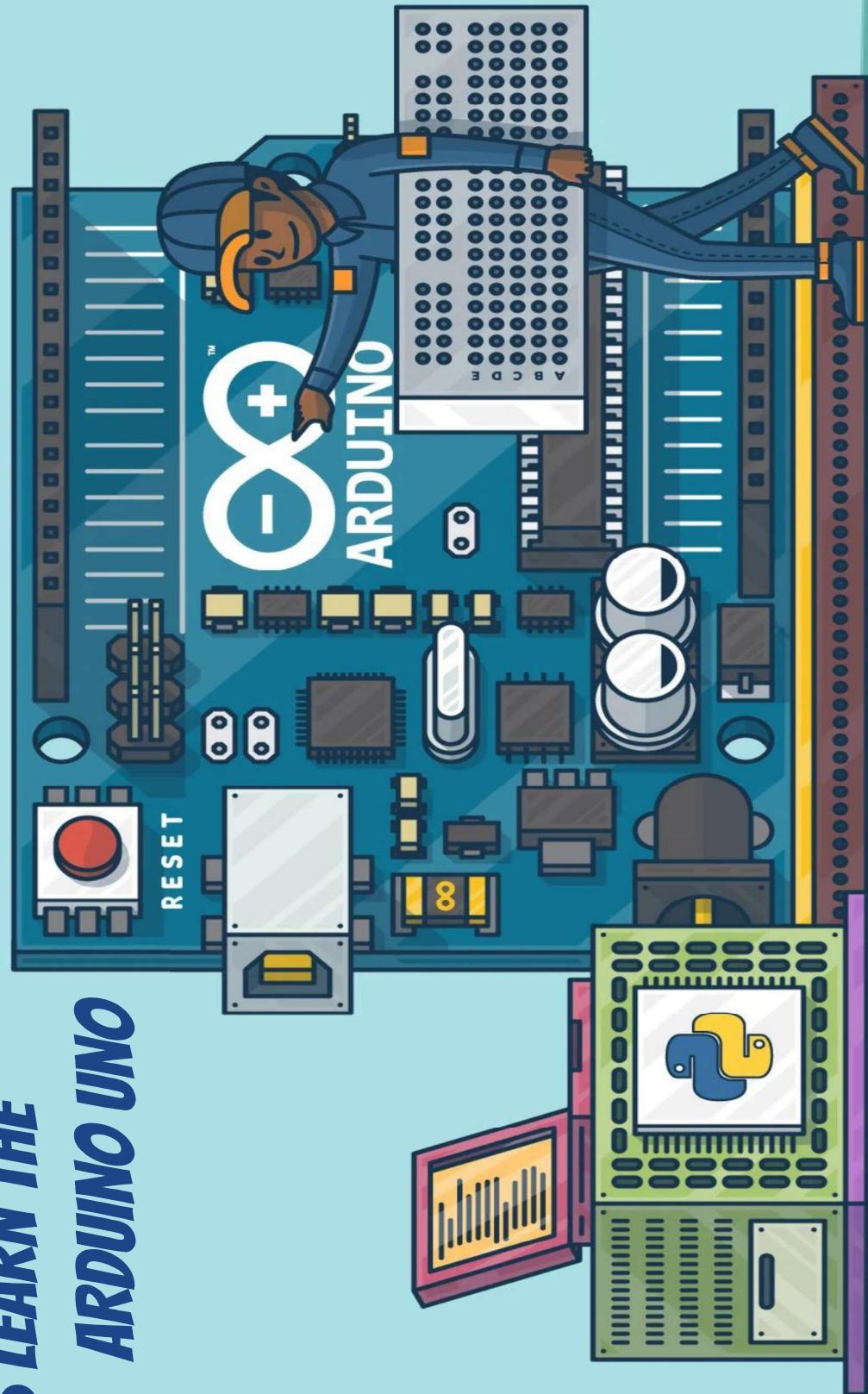
RADAR

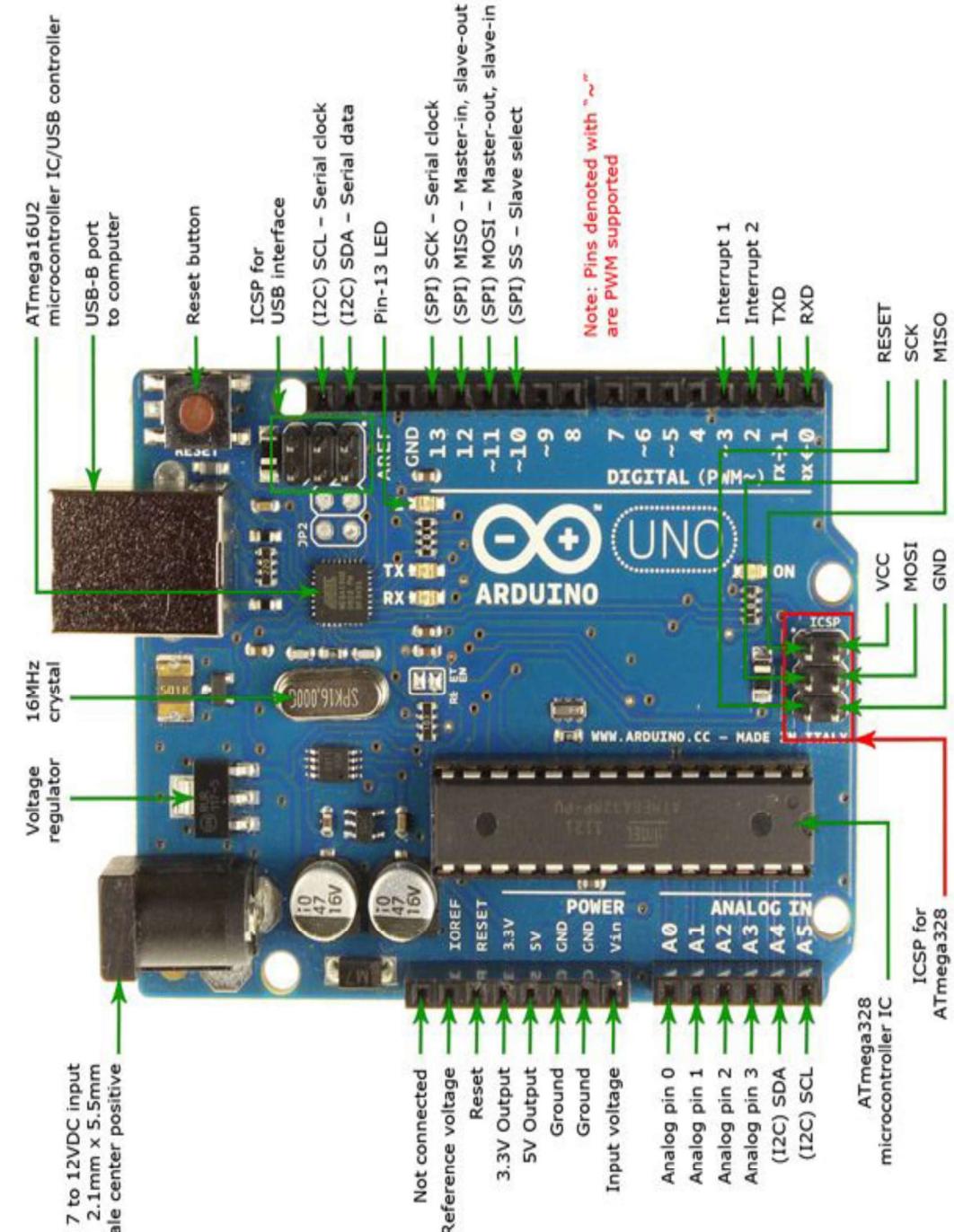


FIRE DETECTION AND ALERT SYSTEM



**LET'S LEARN THE
ARDUINO UNO**





ARDUINO UNO



Vin : 7V to 12V DC

Digital I/O Pins : 14 pins (0 to 13)

Analog I/O Pins : 6 pins (A0 to A5)

Output Voltage : 5v and 3.3v

RAM : 2KB and ROM: 32KB

Serial communication : digital 0(Rx) and 1(Tx) pin

Microcontroller : ATmega328P (8 bit)

Operating frequency : 16 MHz



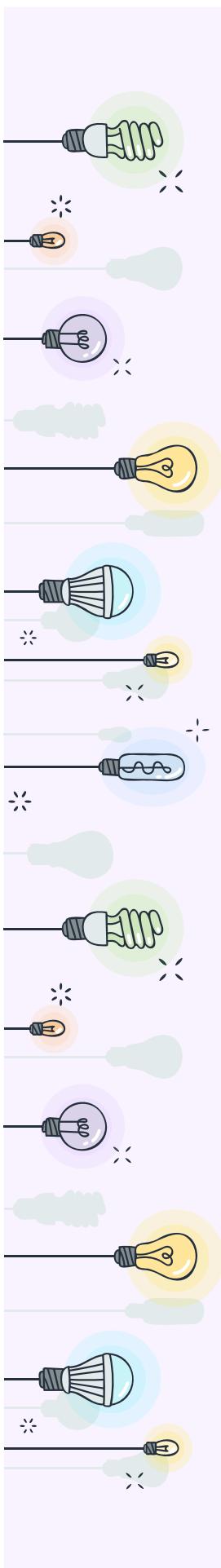
Reset button: To run the program from
start

Inbuilt LED : Power on
13 digital pin
Tx, Rx

Communication protocol: UART
SPI
I²C

Programming Cable: USB A to USB B





To program arduino we require the
Arduino ide :

<https://www.arduino.cc/en/software>





HOW TO USE ARDUINO... ...



IT'S TIME TO
HANDS-ON

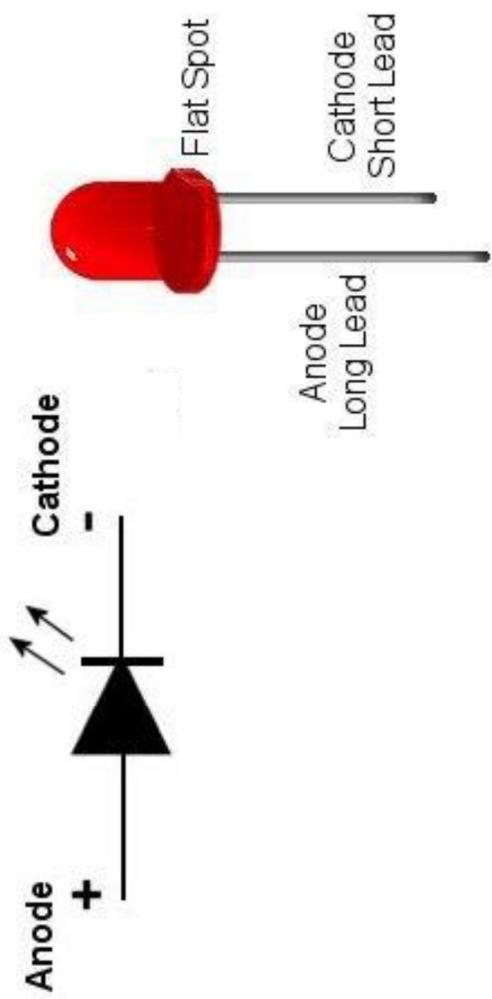
OUR PROCESS IS EASY

Code

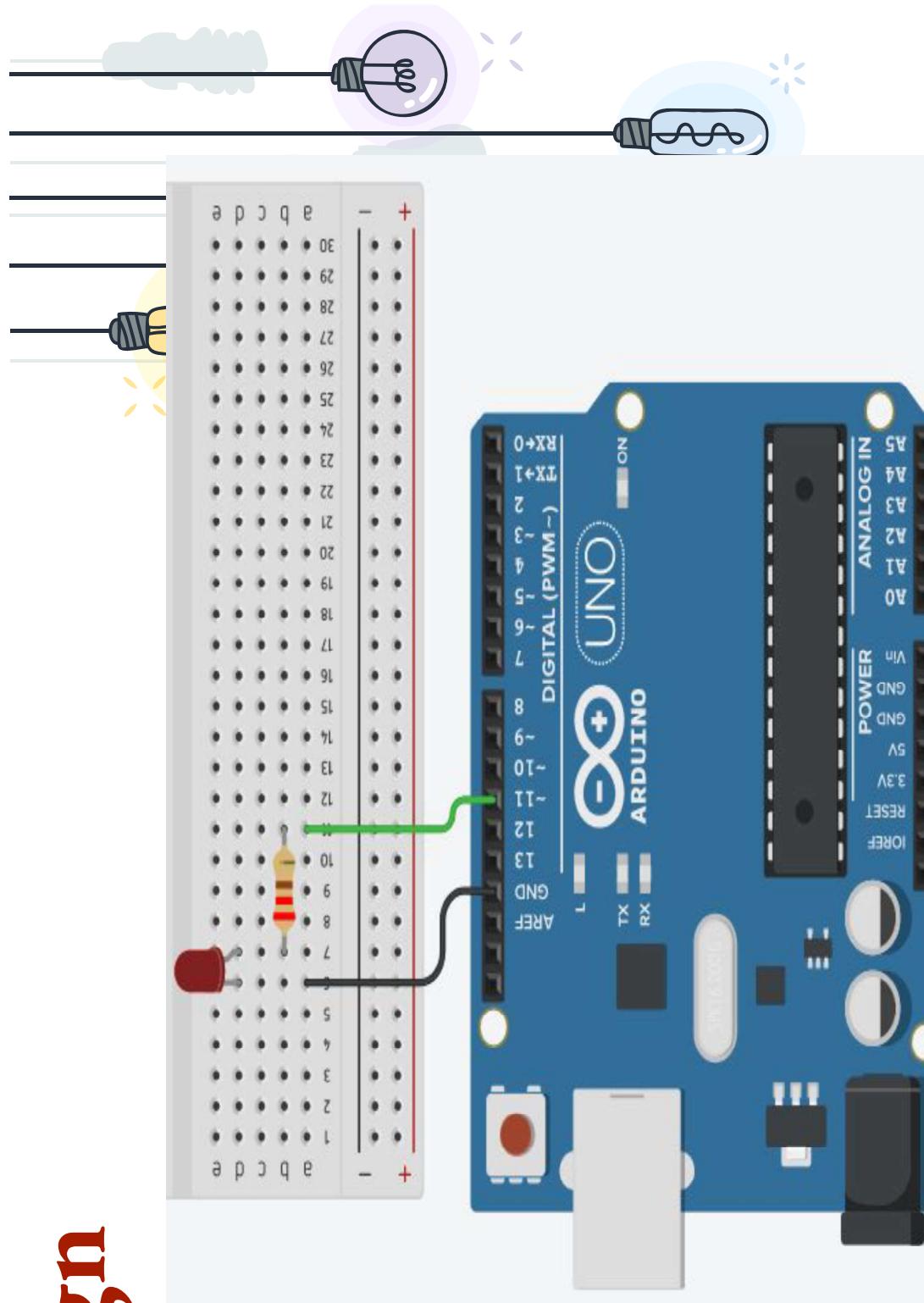
Connect

Design

LED



Design





DON'T HAVE ARDUINO
DON'T WORRY



<https://www.tinkercad.com/>

Basics of Programming

```
void setup()
{
    pinMode(pin_number, INPUT/OUTPUT);
    Serial.begin(Baud_rate);
}

void loop()
{
    digitalWrite(pin_number,HIGH/LOW);
    digitalRead(pin_number);
    analogWrite(pin_number, value);
    analogRead(pin_number);
    delay(ms);
}
```

pinMode()

It is used to **select** the pin and set the **direction** of pin.

```
pinMode(pin_number, INPUT/OUTPUT);
```

pin_number : 0,1,2,3,4,5,6,7,8,9,10,11,12,13, A0,A1,A2,A3,A4,A5

digitalWrite()

It is used produce an **output** to the **digital I/O** pin.

```
digitalWrite(pin_number, HIGH/LOW);
```

pin_number : 0,1,2,3,4,5,6,7,8,9,10,11,12,13

HIGH : 5V LOW : 0V

Turn on the LED

```
void setup()
{
    pinMode(pin_number, OUTPUT);

}

void loop()
{
    digitalWrite(pin_number,HIGH);
}
```

Can you calculate Resistance ?

Step 1:
Find out the **Forward voltage drop (VF)** and **Maximum forward current (IV)**.
Given in datasheet.

Step 2:
Know the Power supply (V_S)

Step 3:
 $R = (V_S - VF) / IV$

Step 4:
Choose nearest standard value.

CALCULATION

Input Voltage=5v

Required current= 20mA

Voltage drop across led = 2.5v

$$R = (\text{Input V} - \text{LED V}) / \text{Required current} \quad \dots\dots(V=I*R)$$

$$R = (5-2.5)/20mA$$

$$R=125\text{ ohm}$$

Turn off the LED

```
void setup()
{
    pinMode(pin_number, OUTPUT);

}

void loop()
{
    digitalWrite(pin_number,LOW);
}
```

Task 1 . Blink the LED

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

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

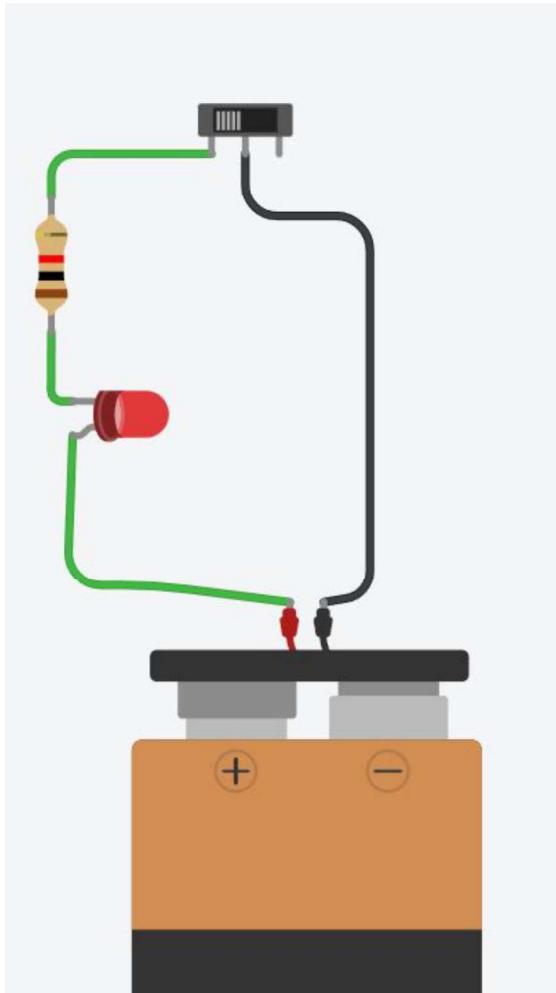
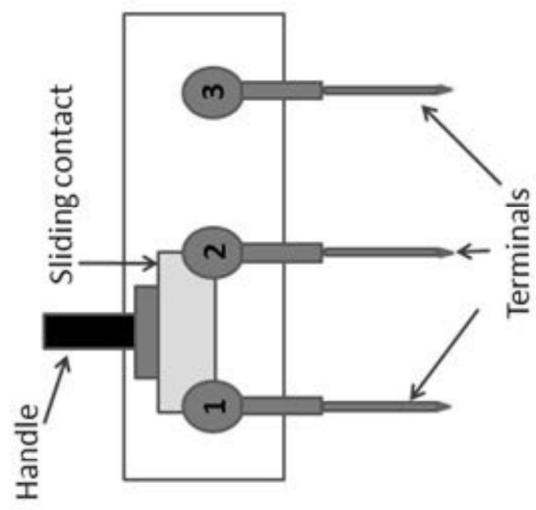
Task 2. Build Traffic light using 3 LED

```
int Red = pin_number //3
int Yellow= pin_number //4
int Green = pin_number //5

void setup()
{
    pinMode(Red, OUTPUT);
    pinMode(Yellow, OUTPUT);
    pinMode(Green, OUTPUT);
}

void loop()
{
    digitalWrite(red,HIGH);
    digitalWrite(yellow,LOW);
    digitalWrite(green,LOW);
    delay(5000);
    digitalWrite(red,LOW);
    digitalWrite(yellow,HIGH);
    digitalWrite(green,LOW);
    delay(2000);
    digitalWrite(red,LOW);
    digitalWrite(yellow,LOW);
    digitalWrite(green,HIGH);
    delay(5000);
}
```


Sliding Switch



We will connect the wire with two adjacent terminals.

If-Else

```
if (condition) {  
    ...execute if condition is true  
}  
  
else {  
    ...execute if condition is not true  
}
```

digitalRead()

It is used to receive the input at digital I/O pin.

digitalRead(pin_number);

Pin_number : 0,1,2,3,4,5,6,7,8,9,10,11,12,13

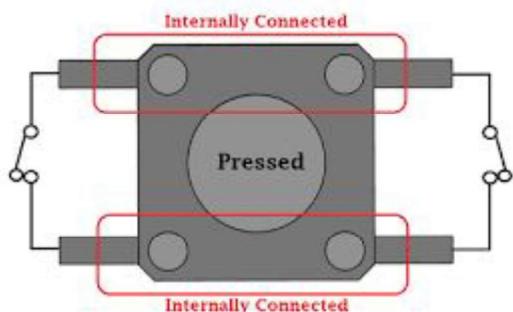
Task 3. Interface the sliding switch and control the LED with Arduino

```
void setup()
{
    pinMode(pin_number, INPUT);
    pinMode(pin_number, OUTPUT);
}

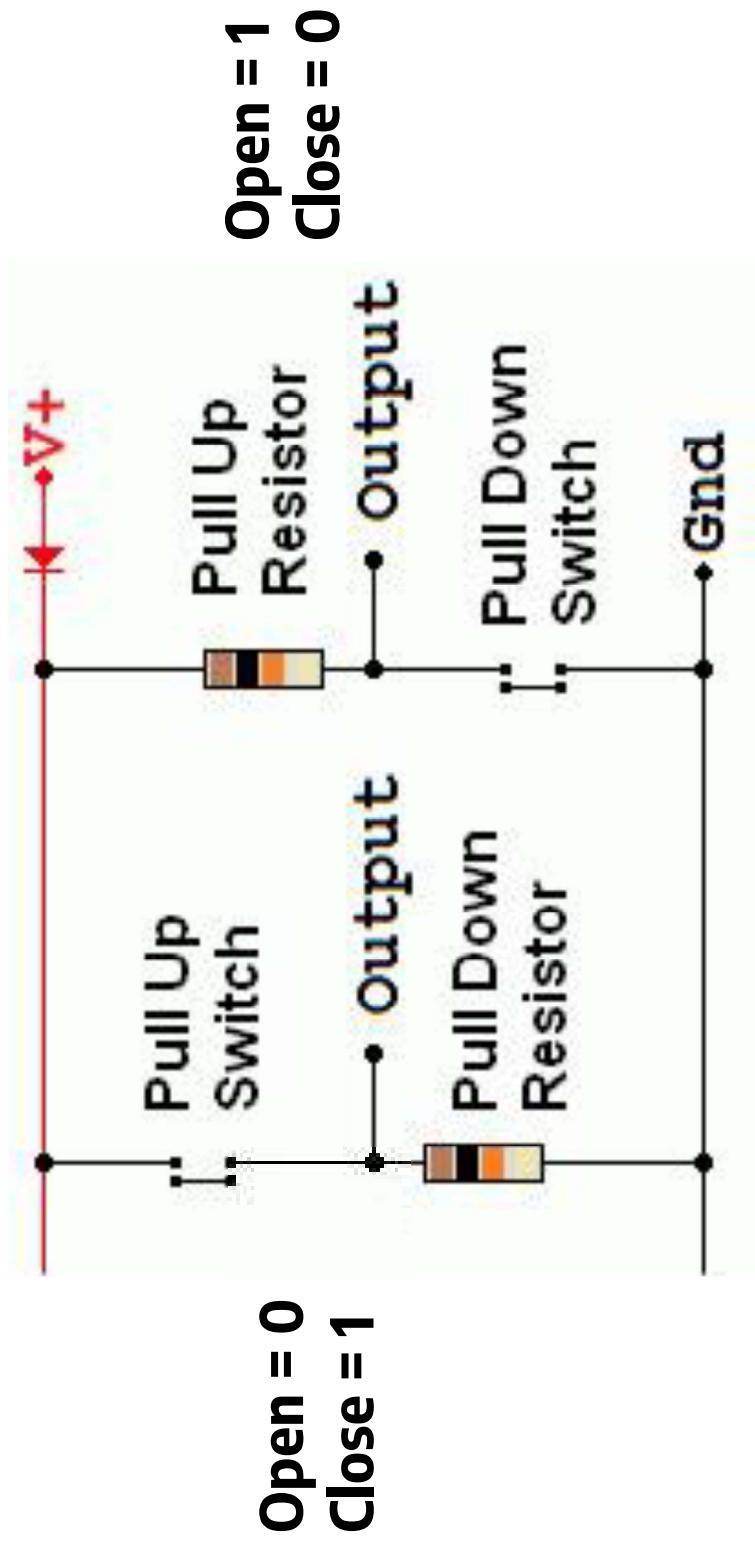
void loop()
{
    if (digitalRead(pin_number)==HIGH){
        digitalWrite(pin_number,HIGH);
    }
    else {
        digitalWrite(pin_number,LOW);
    }
}
```

Switches

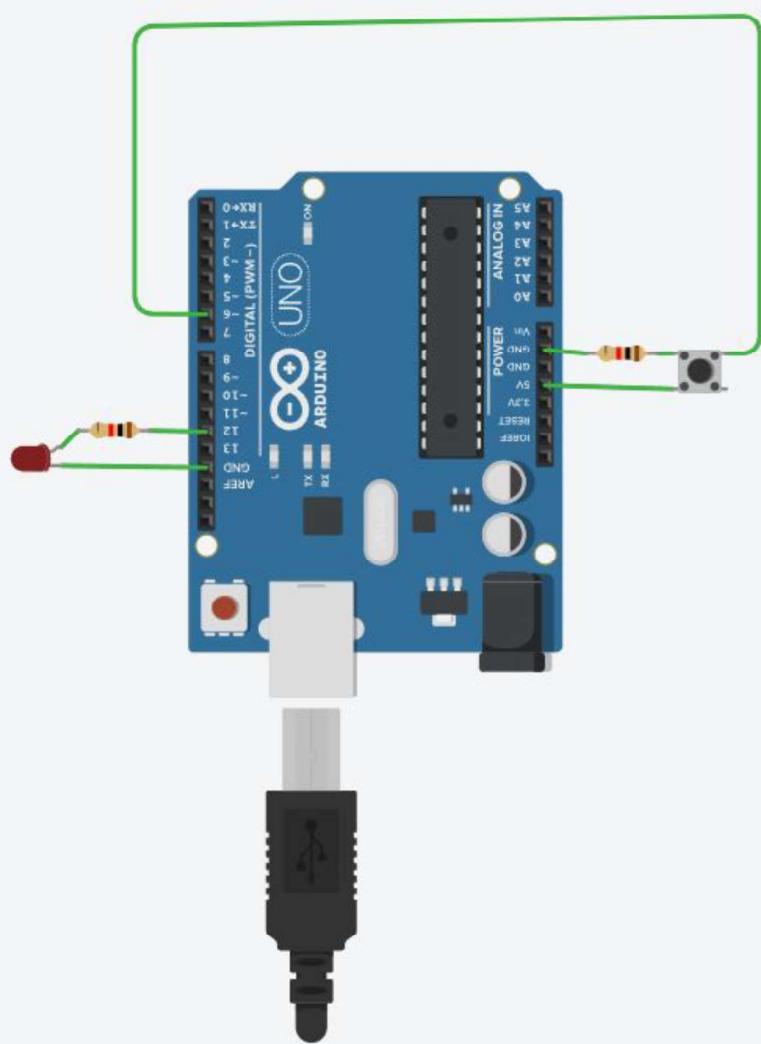
Curved legs (facing each other) are internally connected.



Pull Up and Pull Down Resistor



PULL DOWN RESISTOR



Task 4. Turn on and off LED using 4 leg switch.

```
const int buttonPin = 4;
const int ledPin = 13;

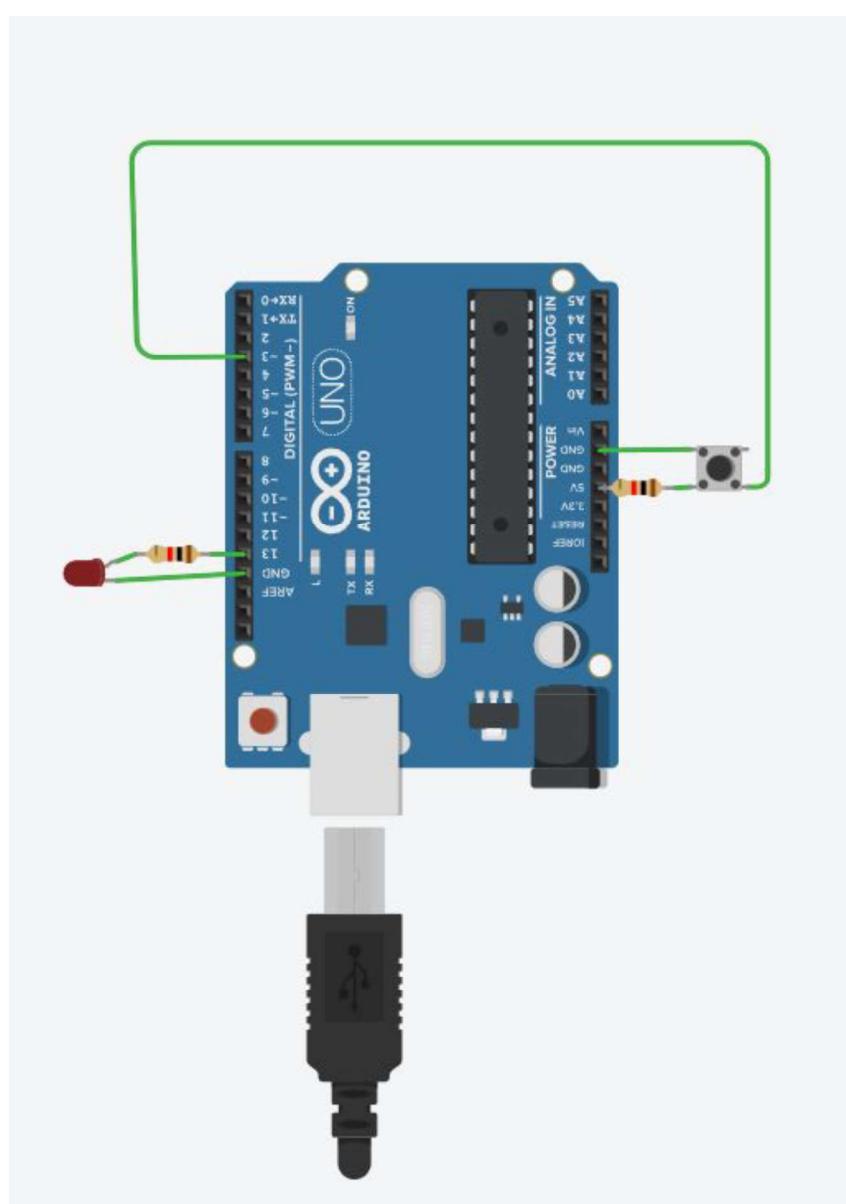
int buttonState = 0;

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

void loop() {

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

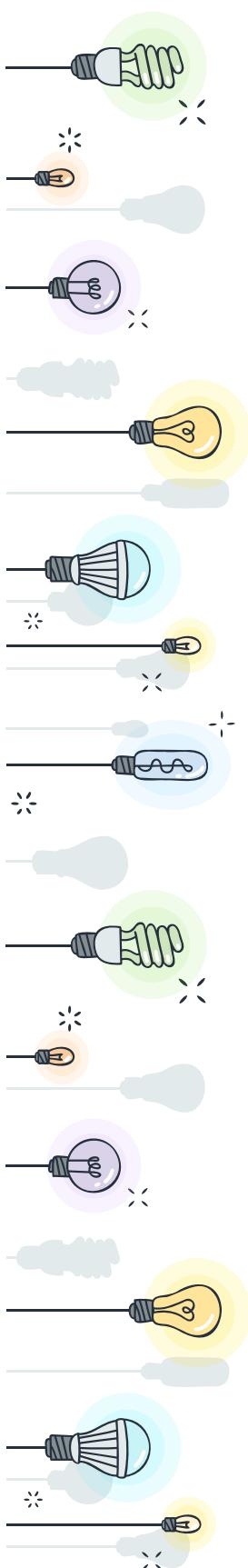
PULL UP RESISTOR



Task 5. Turn on and off LED using 4 leg switch.

```
const int buttonPin = 4;  
const int ledPin = 13;  
  
int buttonState = 0;  
  
void setup() {  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT);  
}  
void loop() {  
  
    buttonState = digitalRead(buttonPin);  
    if (buttonState == HIGH) {  
        digitalWrite(ledPin, LOW);  
    } else {  
        digitalWrite(ledPin, HIGH);  
    }  
}
```

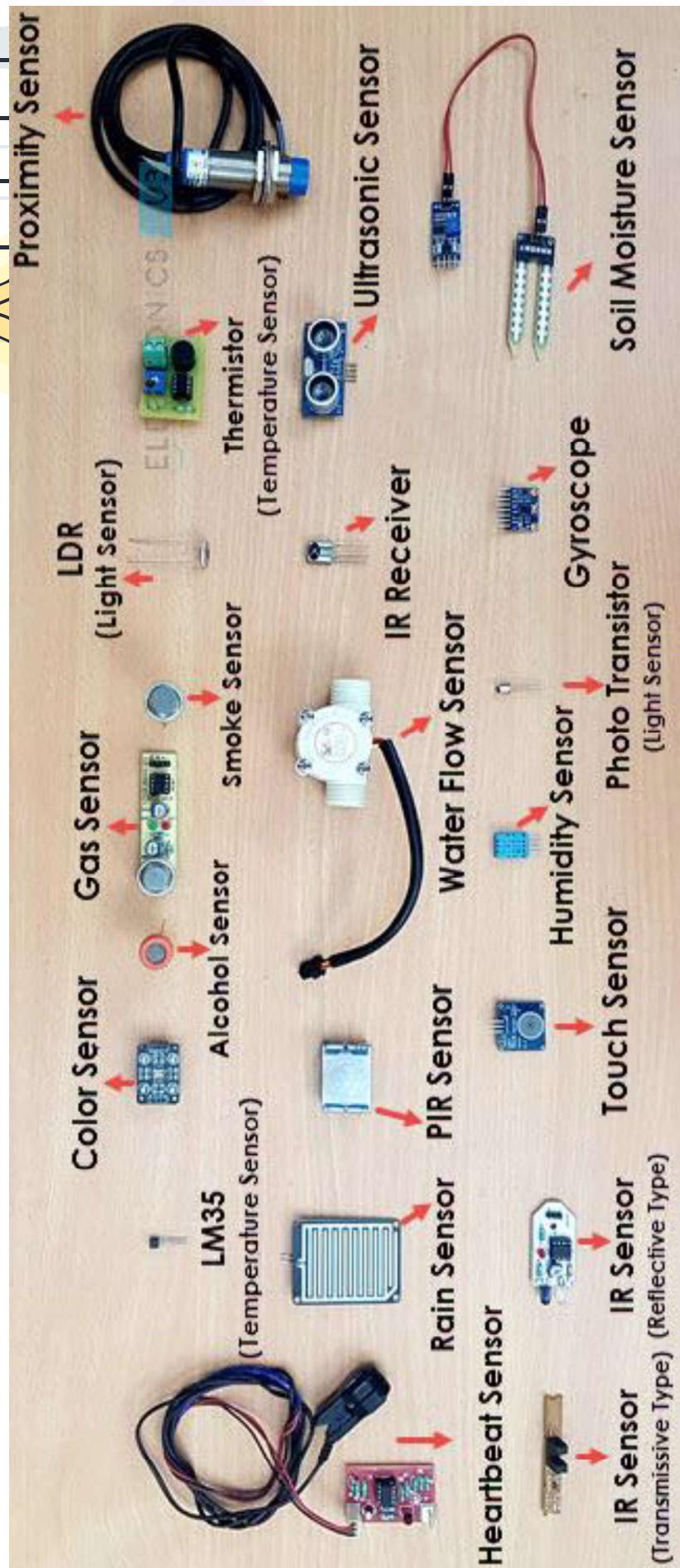
**NOW IT'S TIME
FOR SENSOR**



SENSORS ARE THOSE DEVICES WHICH TAKES PHYSICAL QUANTITIES AS INPUT AND CONVERT THEM INTO ELECTRICAL SIGNAL AS IN THE FORM OF OUTPUT.

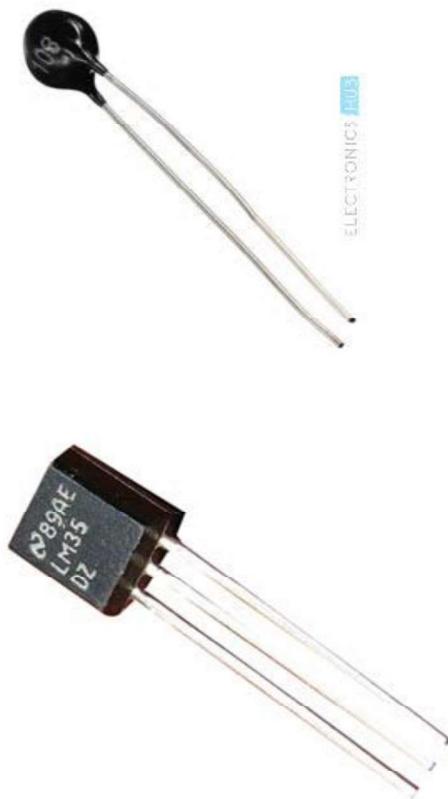
By measuring the physical quantities and using in the application we can make our life more easy.

Types of SENSOR



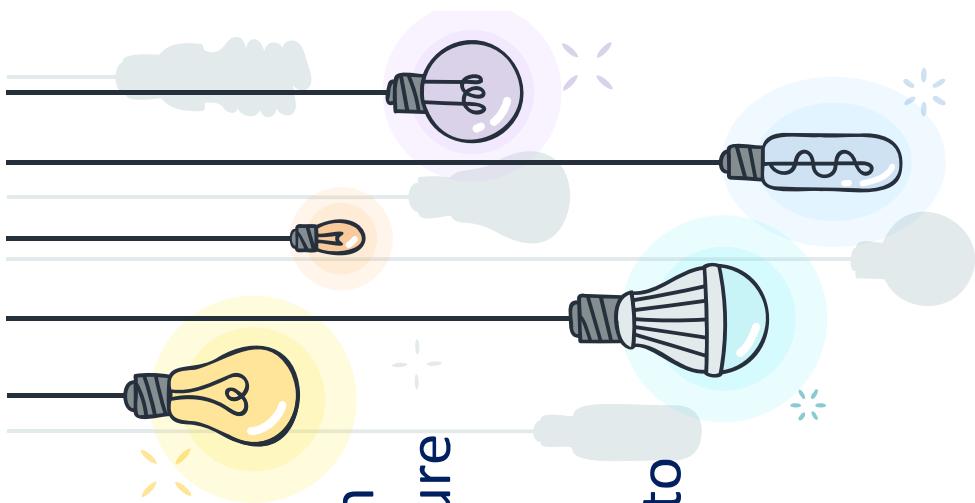
Temperature Sensor

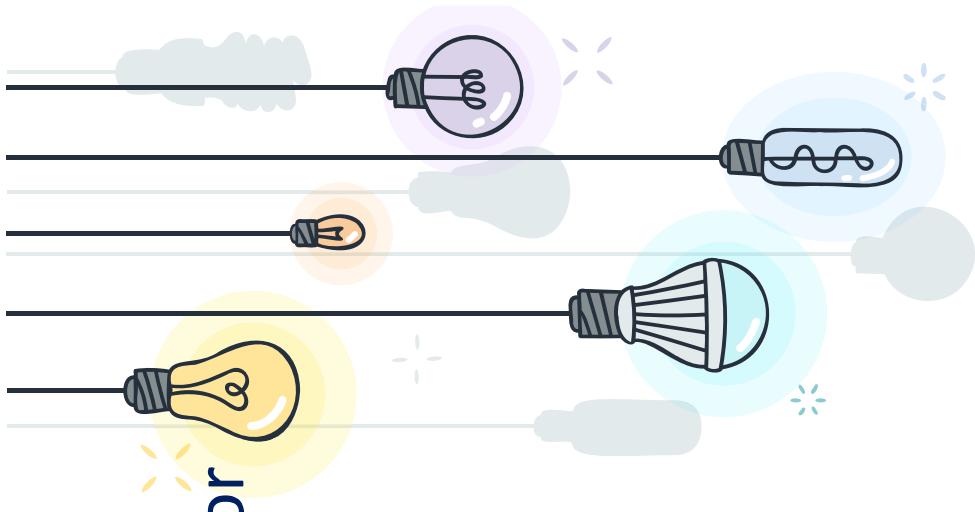
The LM35 is an inexpensive, precision Centigrade temperature sensor. It provides an output voltage that is linearly proportional to the Centigrade temperature .



LM35 - Temperature Sensor IC

10kΩ NTC Thermistor



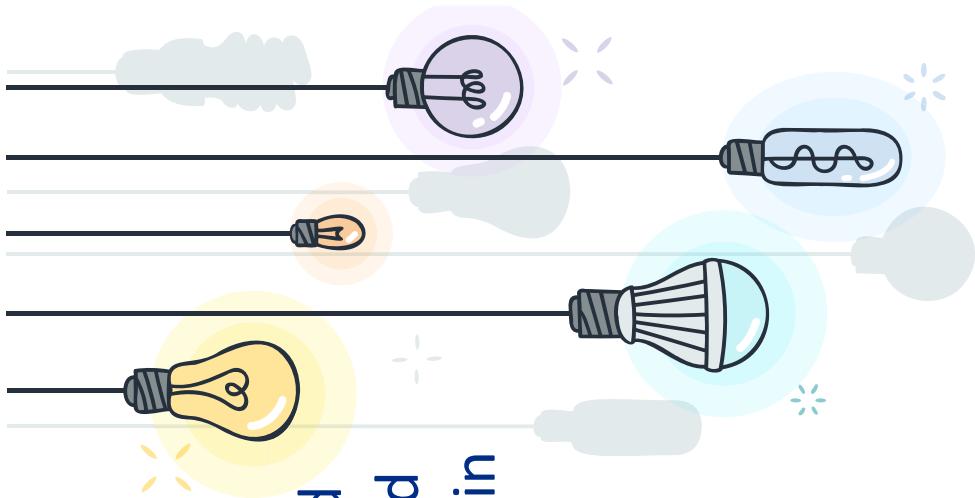


A Proximity Sensor
is a non-contact
type sensor that
detects the
presence of an
object

Proximity Sensor

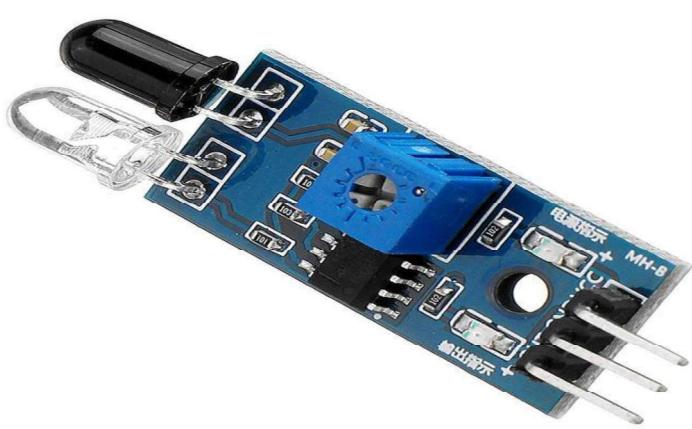
Inductive Proximity Sensor





IR Sensors or Infrared Sensor are light based sensor that are used in various applications like Proximity and Object Detection.

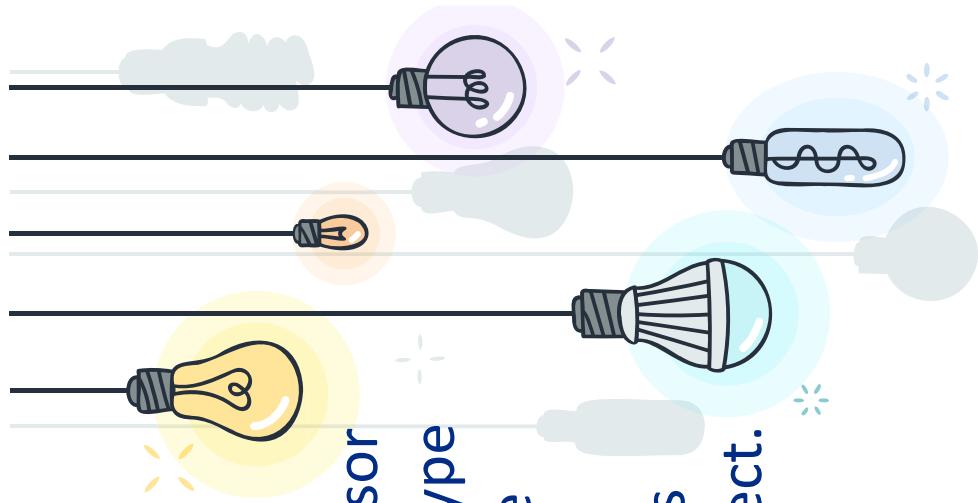
Infrared Sensor



Ultrasonic Sensor

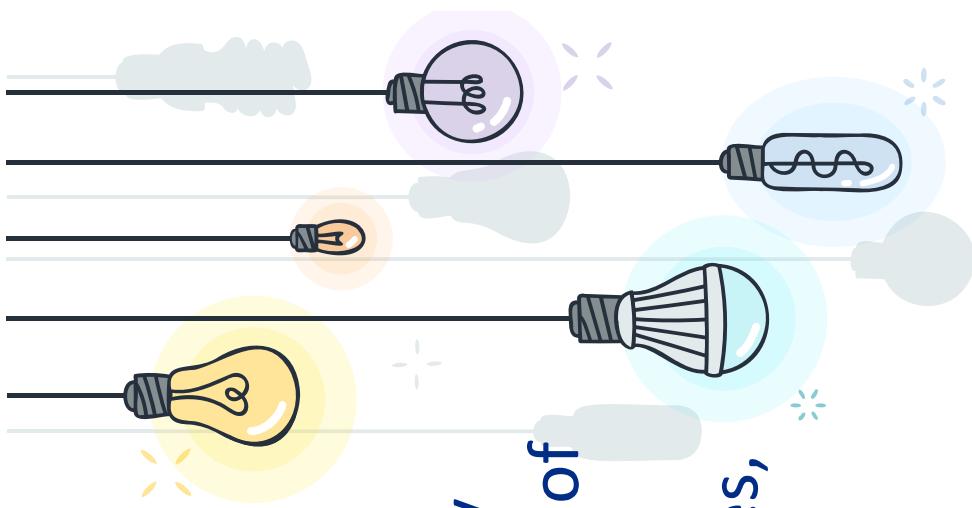
An Ultrasonic Sensor is a non-contact type device that can be used to measure distance as well as Ultrasonic Sensor velocity of an object.

ELECTRONICS HUB



LDR Sensor

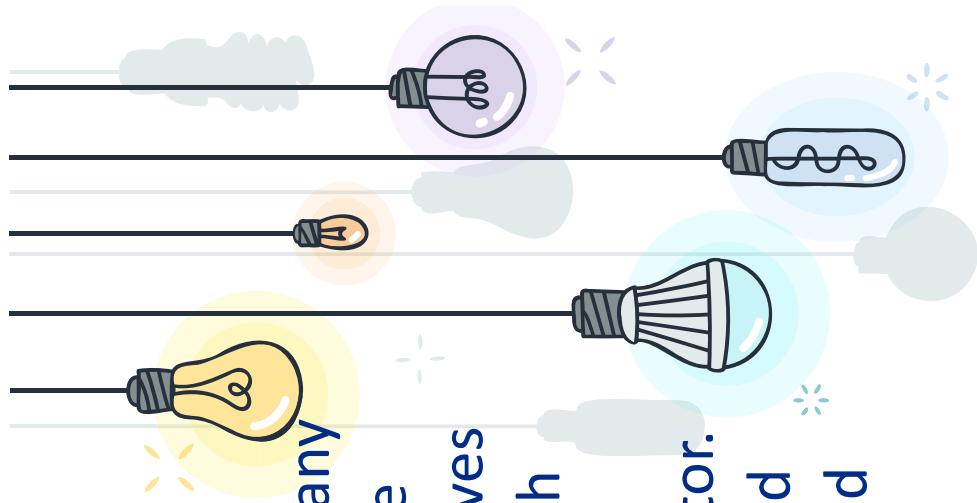
Light Dependent Resistor or LDR. The property of LDR is that its resistance is inversely proportional to the intensity of the ambient light i.e., when the intensity of light increases, its resistance decreases and vice-versa.



Smoke and Gas Sensor

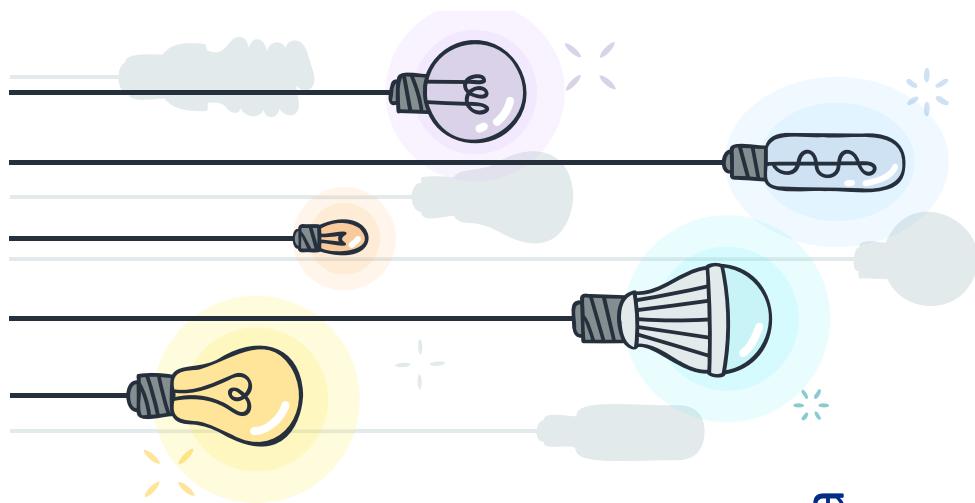


Detect Smoke and any particular gas in the atmosphere and gives the indication which set the alarm and other safety indicator. Usually industry and offices are equipped with this sensor



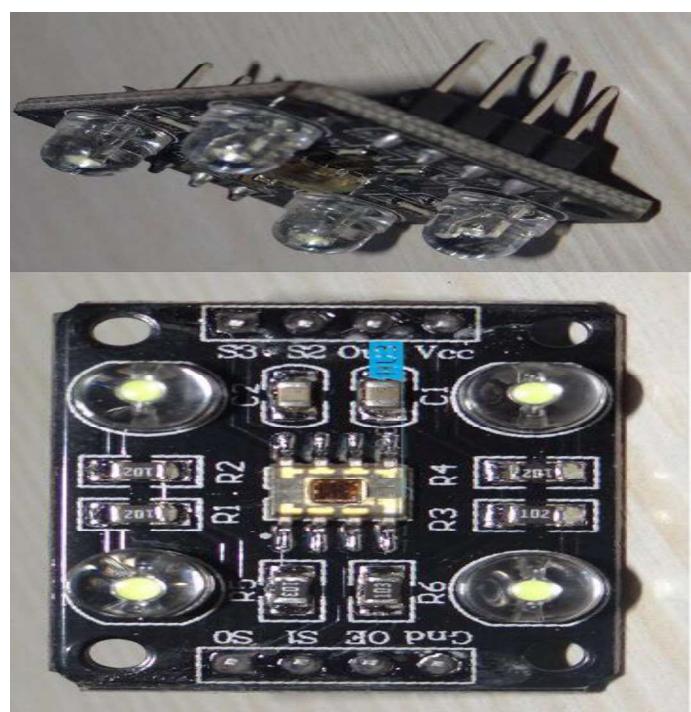
Alcohol Sensor

As the name suggests, an Alcohol Sensor detects alcohol. Usually, alcohol sensors are used in breathalyzer devices, which determine whether a person is drunk or not.



Color Sensor

A Color Sensor is an useful device in building color sensing applications in the field of image processing, color identification, industrial object tracking etc. The TCS3200 is a simple Color Sensor

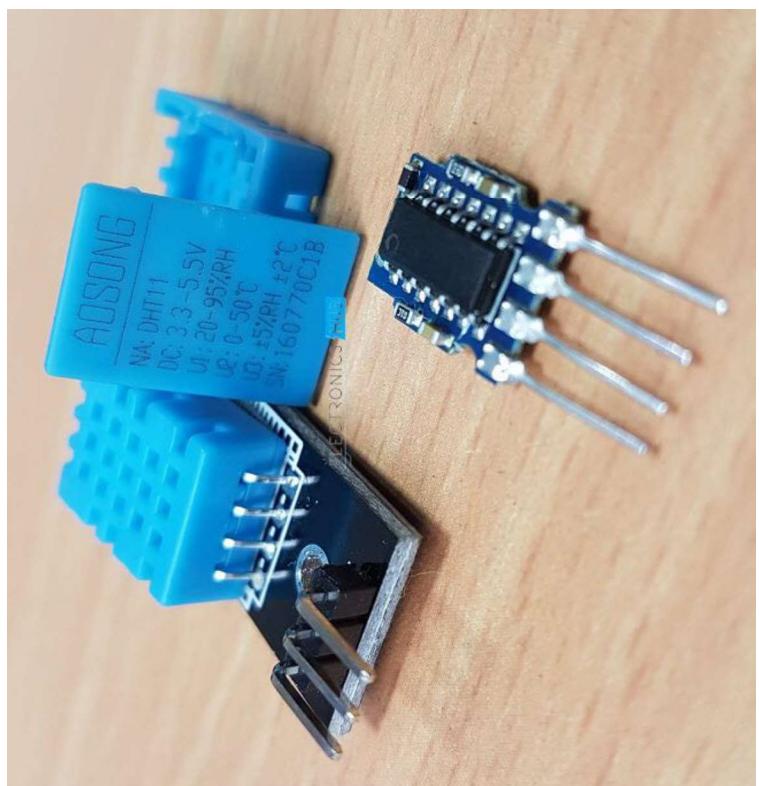


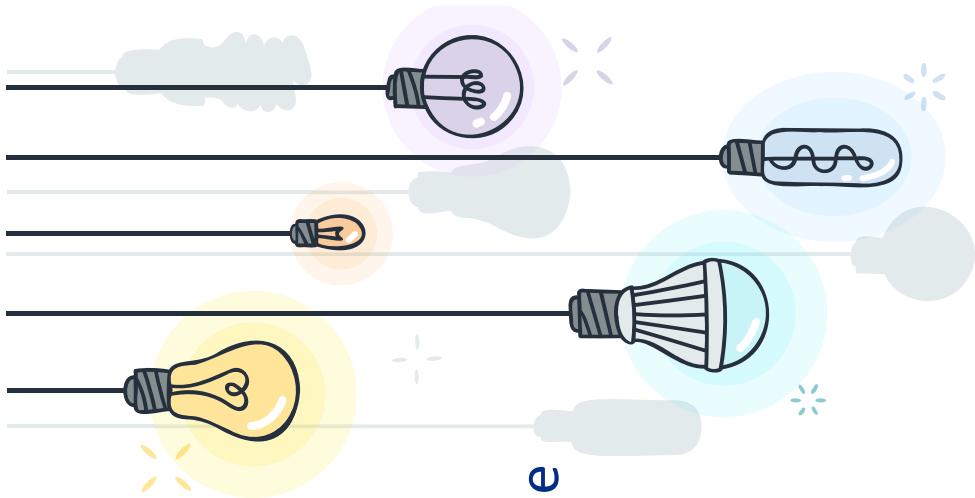
Humidity Sensor

Humidity sensor
measure the
temperature and
relative

Humidity. useful
for the
determining the
heat index.

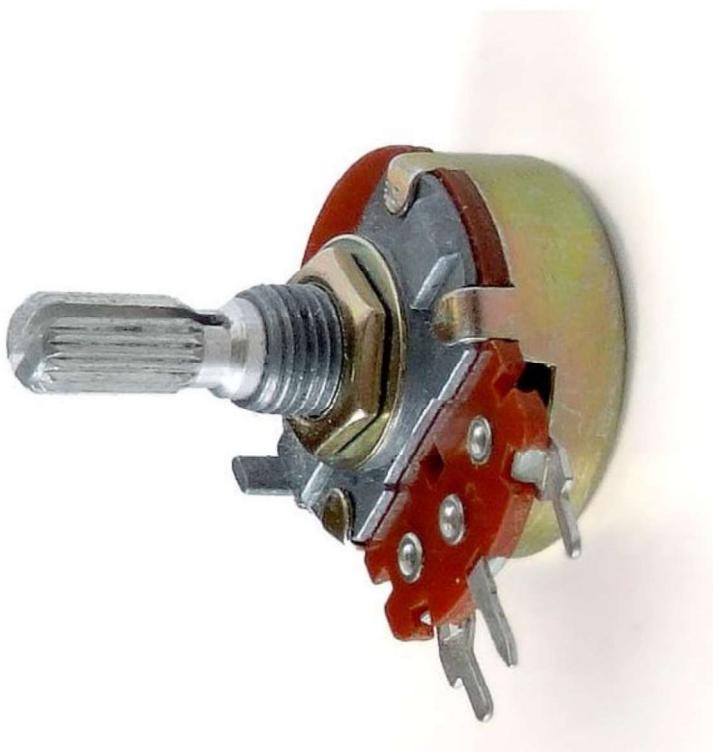
Ex. DHT11





A **potentiometer** is a three-terminal resistor with a rotating contact that forms an adjustable voltage divider.

POTENTIOMETER

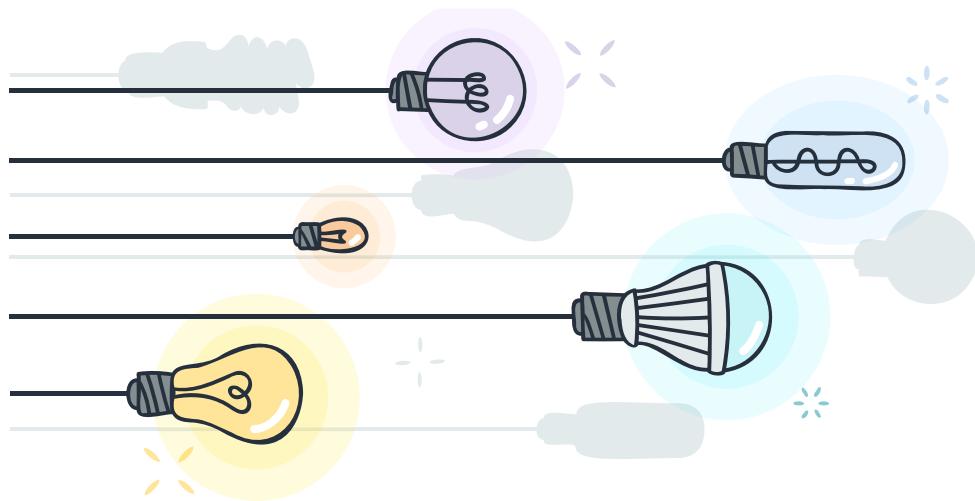


analogRead()

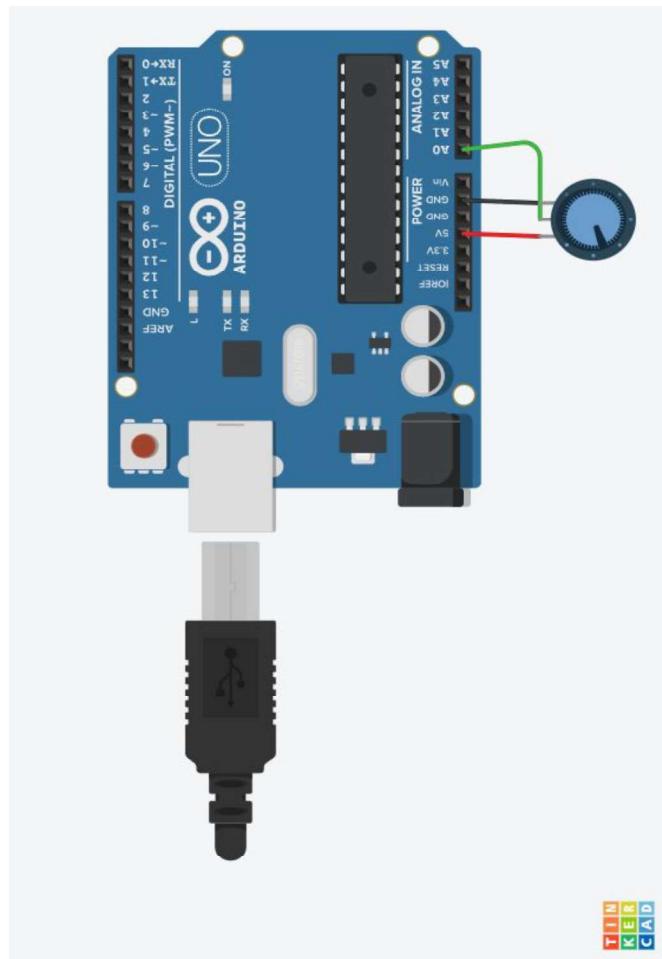
It is used to receive the input at analog I/O pin.

analogRead(pin_number);

Pin_number : A0, A1, A2, A3, A4, A5



💡 POTENTIOMETER WITH ARDUINO



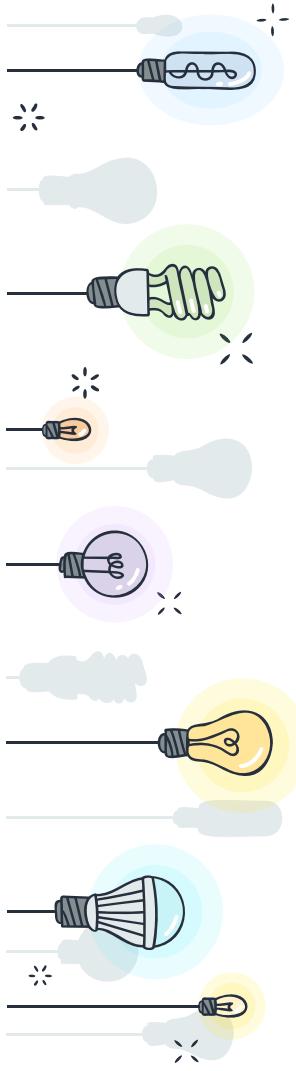


**WHERE WE CAN
SEE OUTPUT?**

SERIAL MONITOR AND SERIAL PLOTTER

This are tools used to print or display the output which helps in debugging and also used to communicate with arduino.





HOW TO USE SERIAL MONITOR AND PLOTTER ?

Serial.begin(baud rate)

....used to setup communication

Baud rate is rate at which data is transferred i.e. Bits per second

Serial.print()

....used to display on monitor

Serial.println()

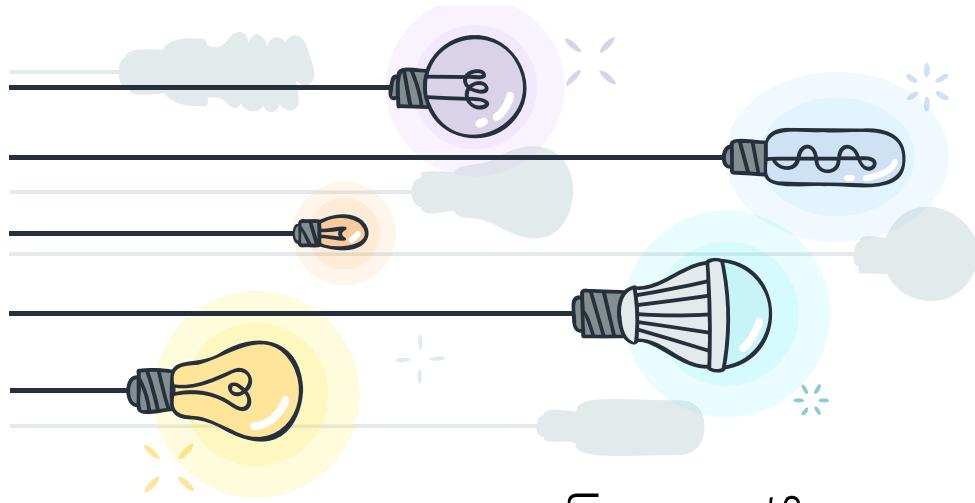
.... Used to display on each new line

Task 6. Interface potentiometer with Arduino and Print the values.

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int sensorValue = analogRead(A0); // read the input on analog pin 0:  
  
    // print out the value you read:  
    Serial.println(sensorValue);  
    delay(1); // delay in between reads for stability  
}
```



**WHY OUTPUT IS
FROM 0 TO
1023?**



To process the data microcontroller needs the data in digital format



To convert analog data into digital format ADC is used
Analog to Digital converter

In Arduino Six 10-bit ADC are used . That means ADC can generate value from 0 to 1023 (2^{10})

And whatever input range we provide will map with this range in our case
 $(0 - 5) \rightarrow (0 - 1023)$.



**HOW TO GET
ACTUAL PROVIDED
OUTPUT?**

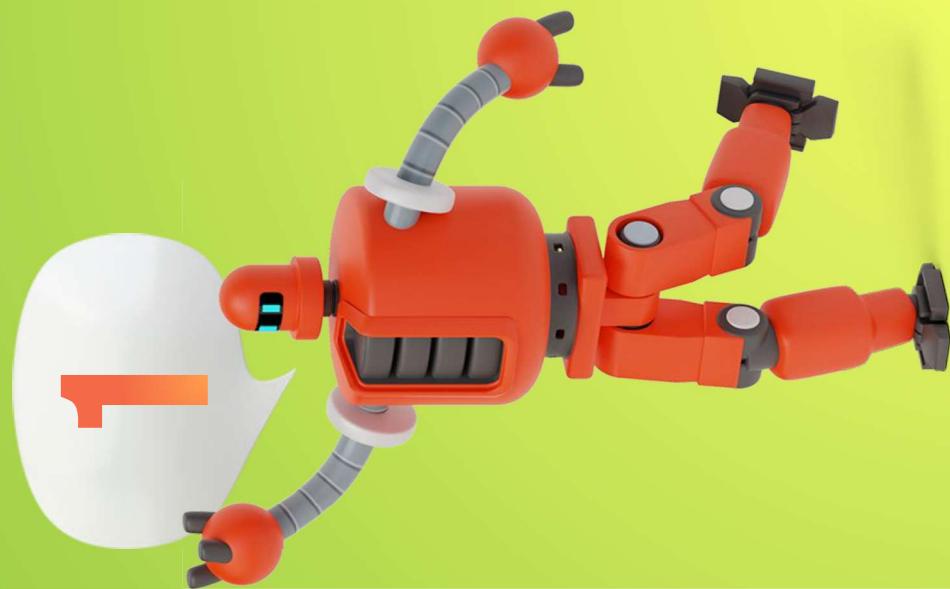
**TO GET THE ACTUAL VOLTAGE WE USE
FORMULA**

Sensor_Value * 1P_Voltage / 1024

Sensor_Value * 5.0 / 1024

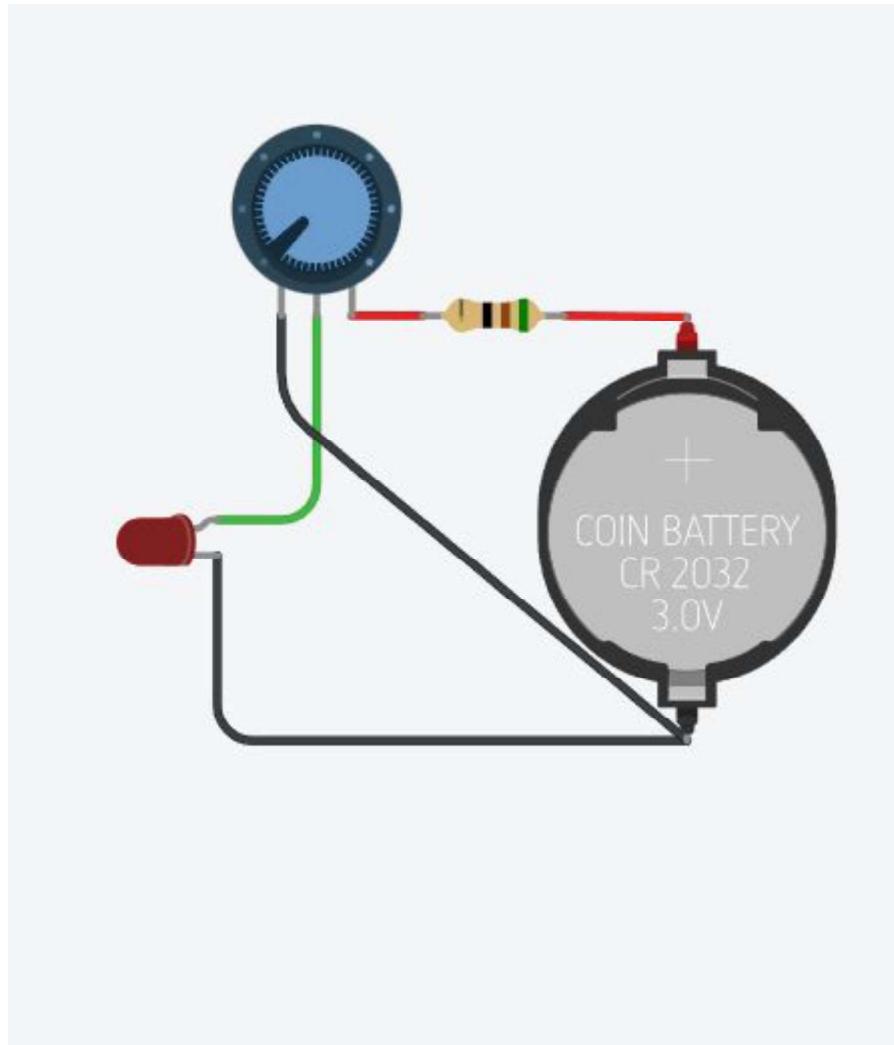
Task 7. Interface potentiometer with Arduino and Print the Actual voltage.

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int sensorValue = analogRead(A0); // read the input on analog pin 0:  
  
    // Convert the analog reading (which goes from 0 - 1023) to a voltage (0 - 5V):  
    float voltage = sensorValue * (5.0 / 1023.0);  
  
    // print out the value you read:  
    Serial.println(voltage);  
    delay(1);  
} // delay in between reads for stability
```



CAN WE CONTROL
BRIGHTNESS OF LED
USING
POTENTIOMETER?

Task 8. Interface potentiometer with LED and control brightness.

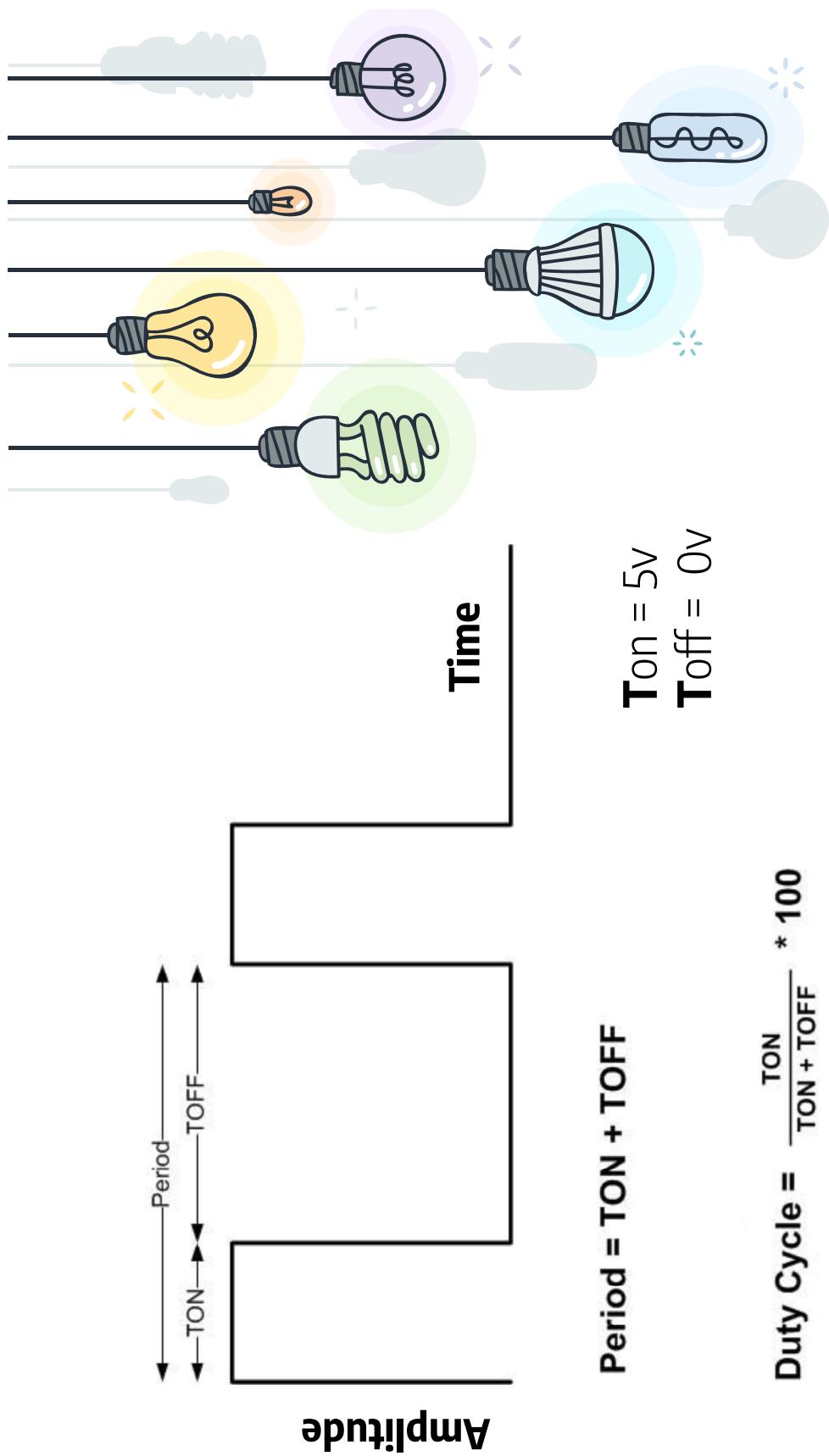


PWM

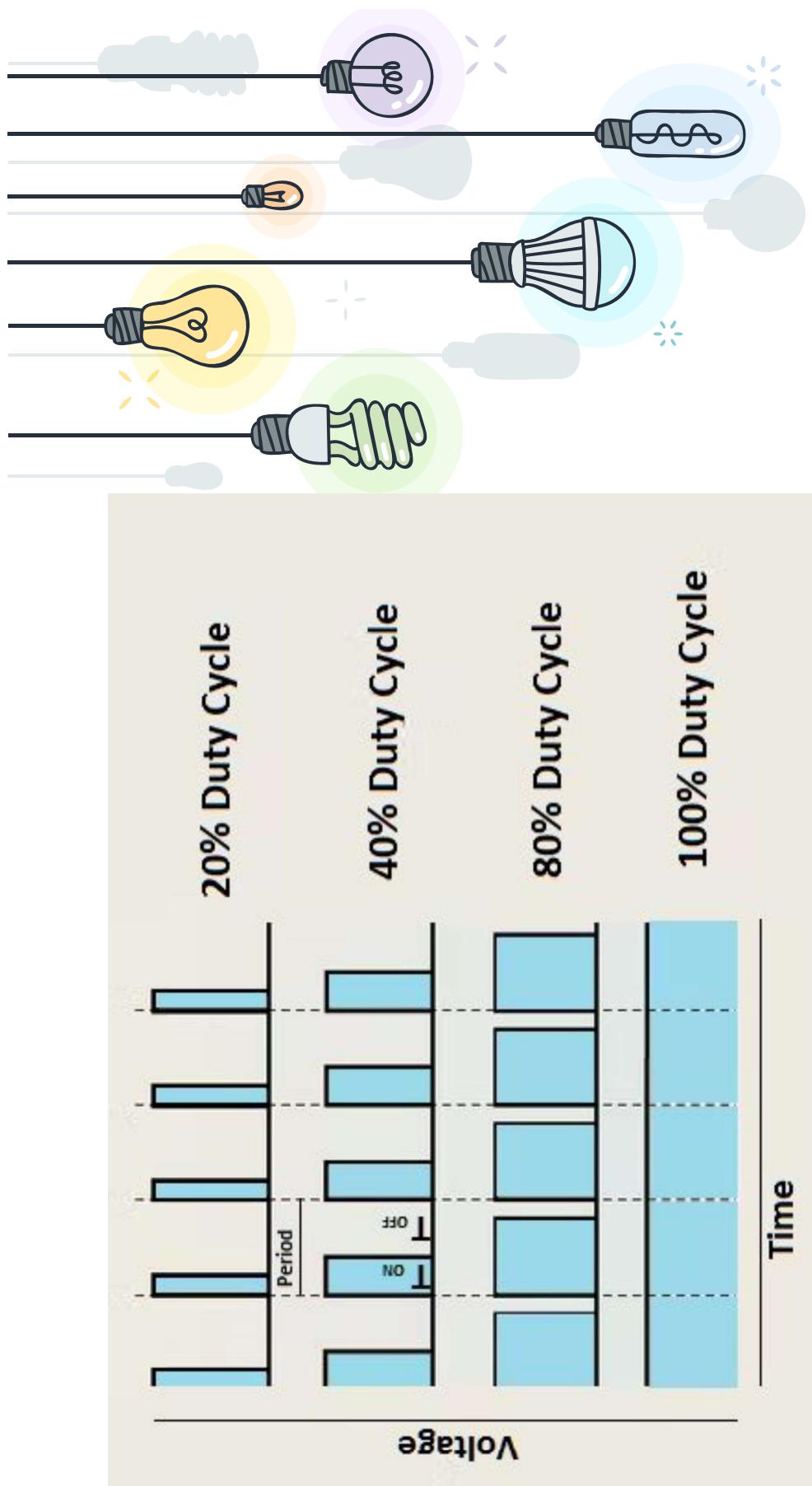
Pulse Width Modulation

Best and efficient way of getting
variable DC voltage.

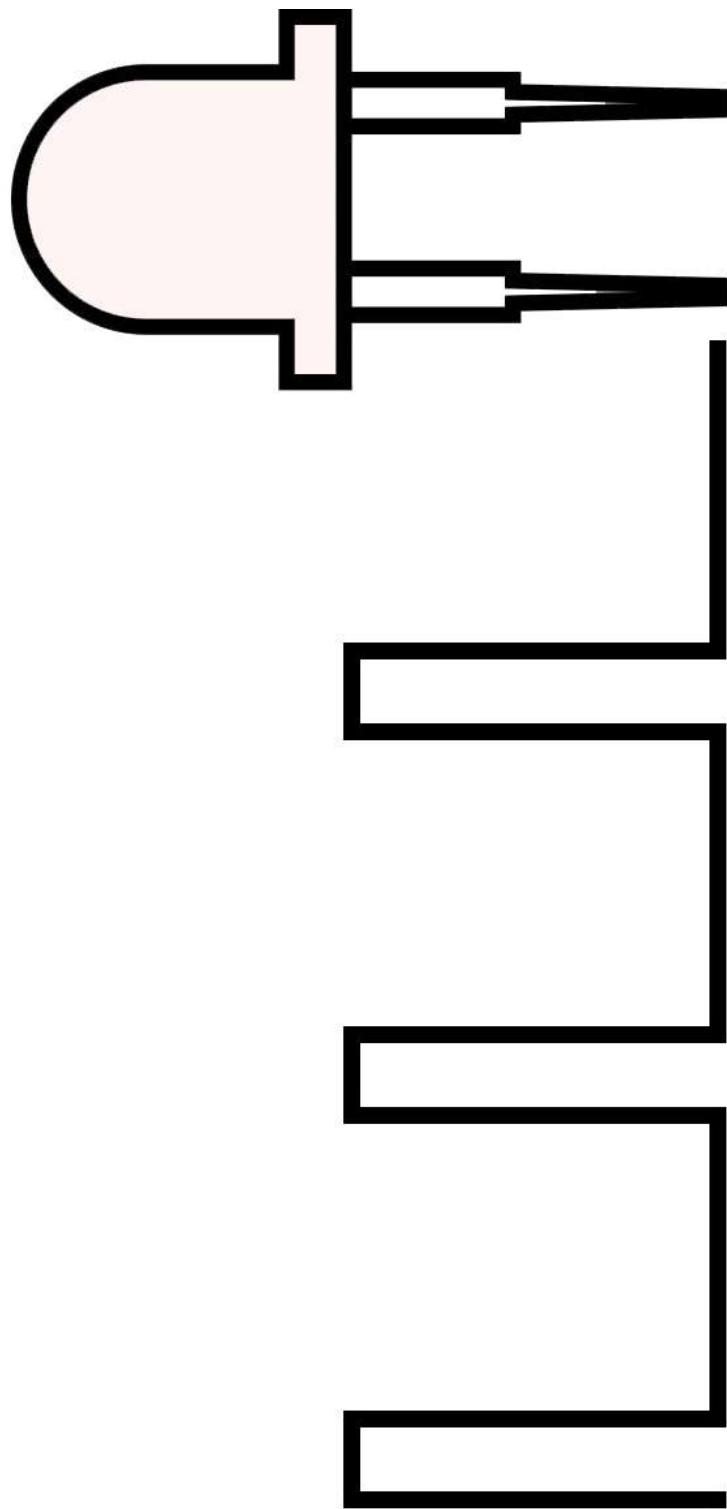




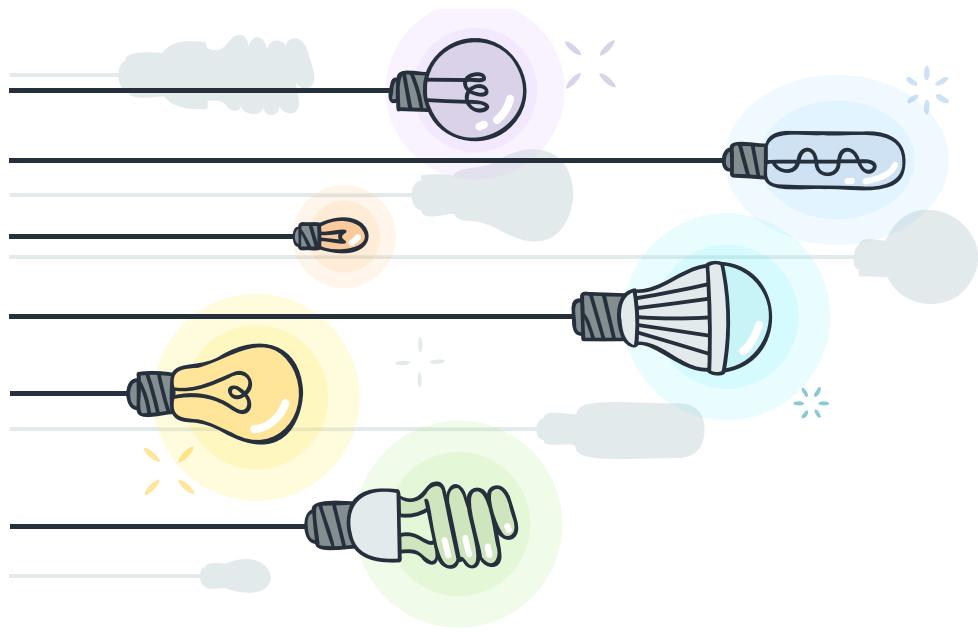
11



Controlling the brightness of LED using PWM.



WHAT OUTPUT WE GET ?



If signal of duty cycle '**D**' is given to system
then system we receive D% of Amplitude
Ex. 20% of 5v is 1v
40% of 5v is 2v

HOW TO GET SUCH SIGNAL ?

analogWrite(pin, 0-255)

- 0 -> 0% Duty cycle
- 127 -> 50% Duty cycle
- 255 -> 100% Duty cycle

analogWrite()

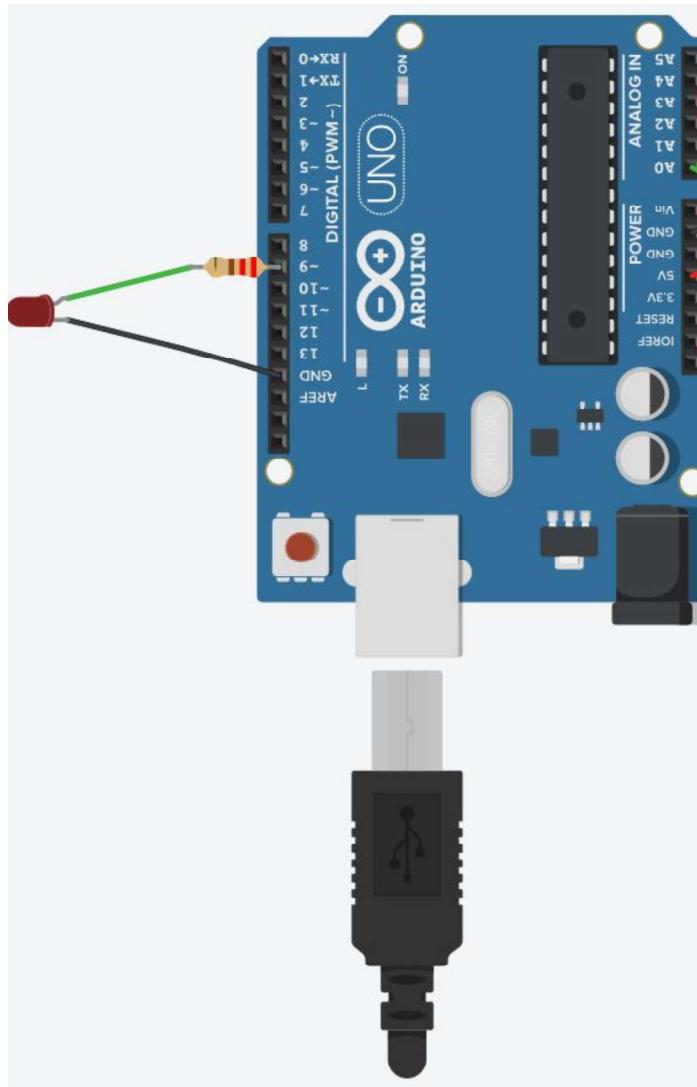
It is used produce an PWM signal at the GPIO pin.

analogWrite(pin_number, value);

Pin_number : 11,10,9,6,5,3

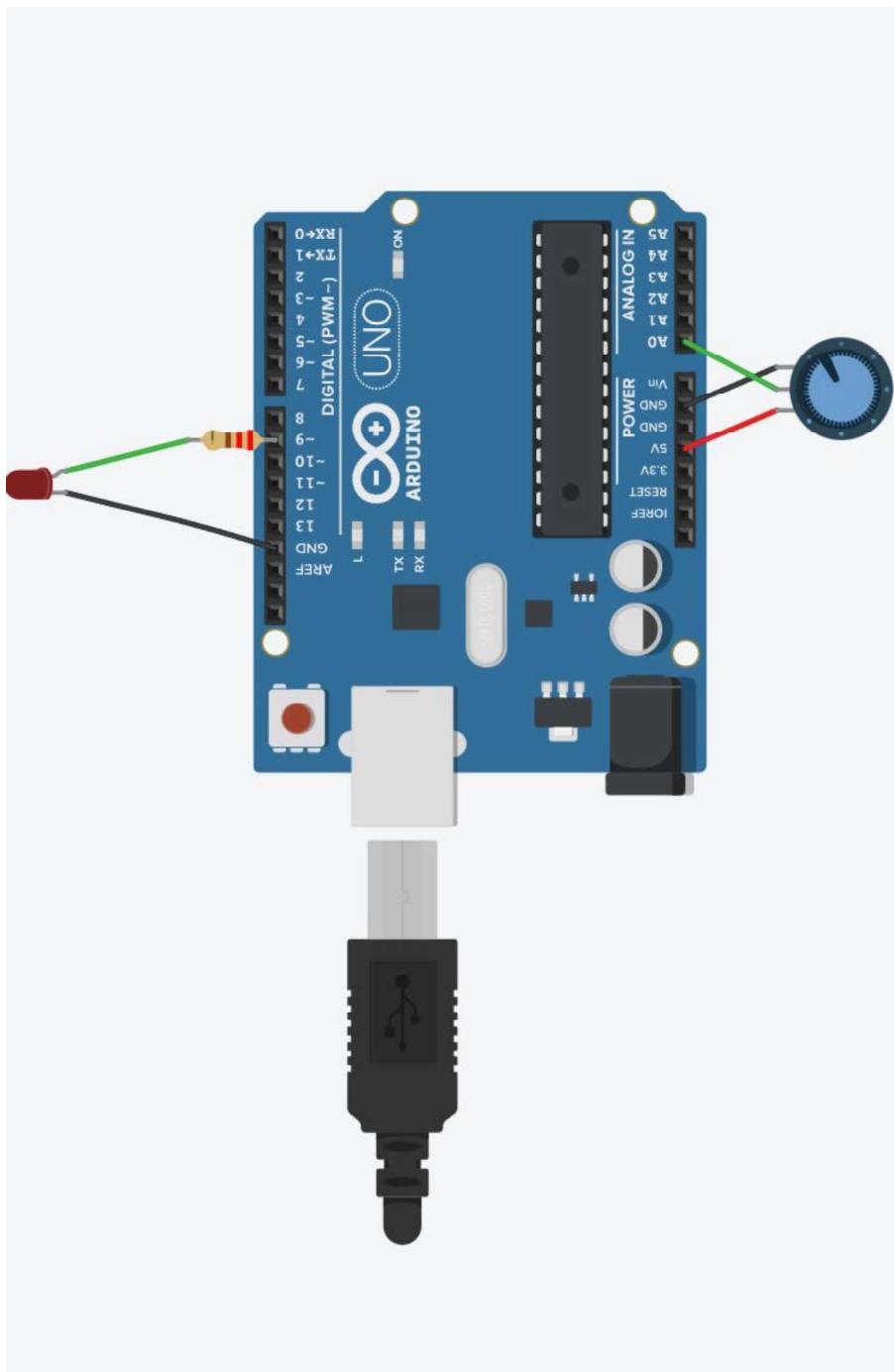
Value is range between **(0 to 255)**

Task 8. Interface LED with arduino and control brightness.



```
void setup() {  
  pinMode(9,OUTPUT);  
  Serial.begin(9600);  
}  
  
void loop() {  
  for(int i=0;i<255;i++){  
    analogWrite(9,i);  
    Serial.println(i);  
    delay(30);  
  }  
  for(int i=255;i>0;i--){  
    analogWrite(9,i);  
    Serial.println(i);  
  }  
}
```

Task 9. Interface potentiometer and LED with arduino and control brightness.



STOP

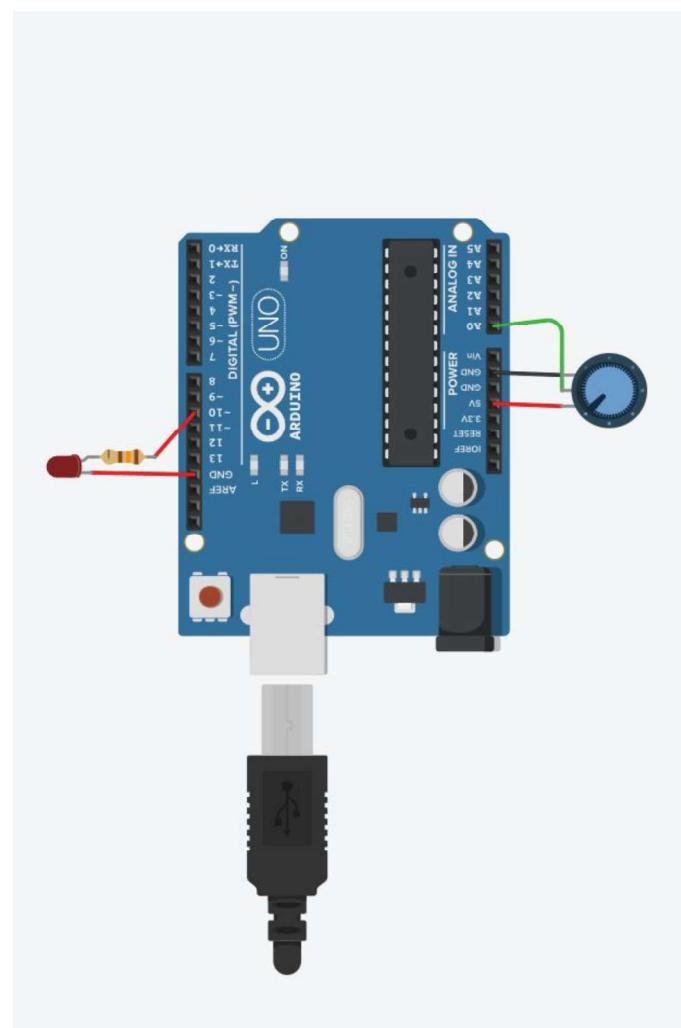
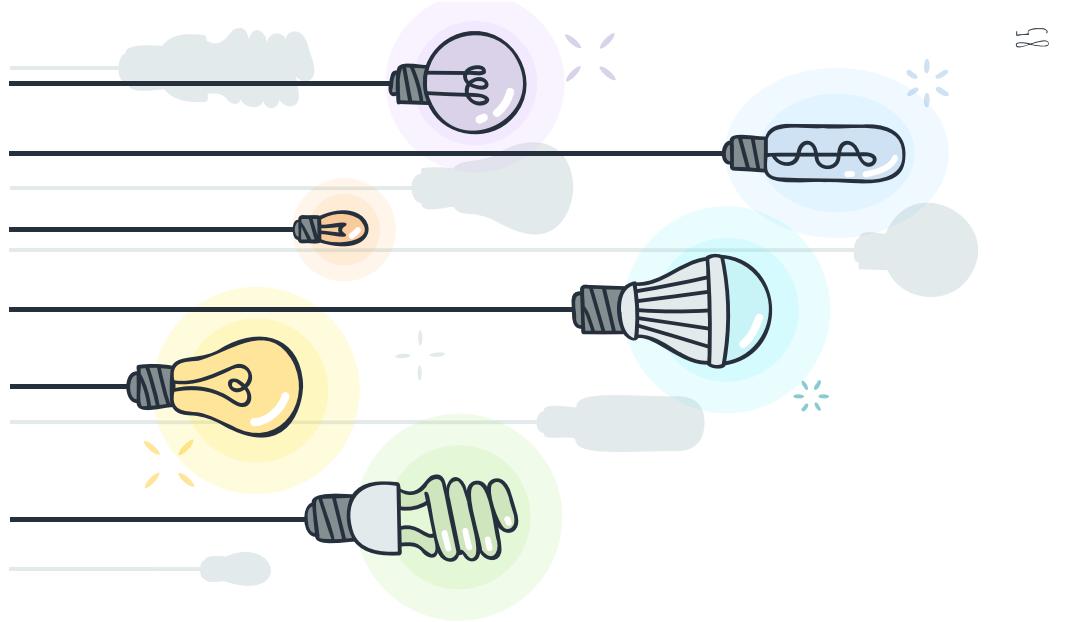


Input from potentiometer is 0 to 1023
And output for LED is 0 to 255

WHAT WE CAN DO ?

map()

LED_value = map(Pot_value, 0 ,1023 , 0 ,255)
This function will map or convert value from 0 to 1023
into 0 to 255



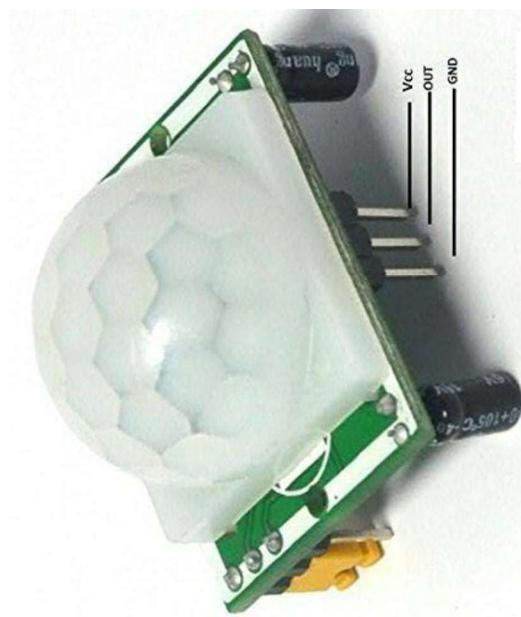
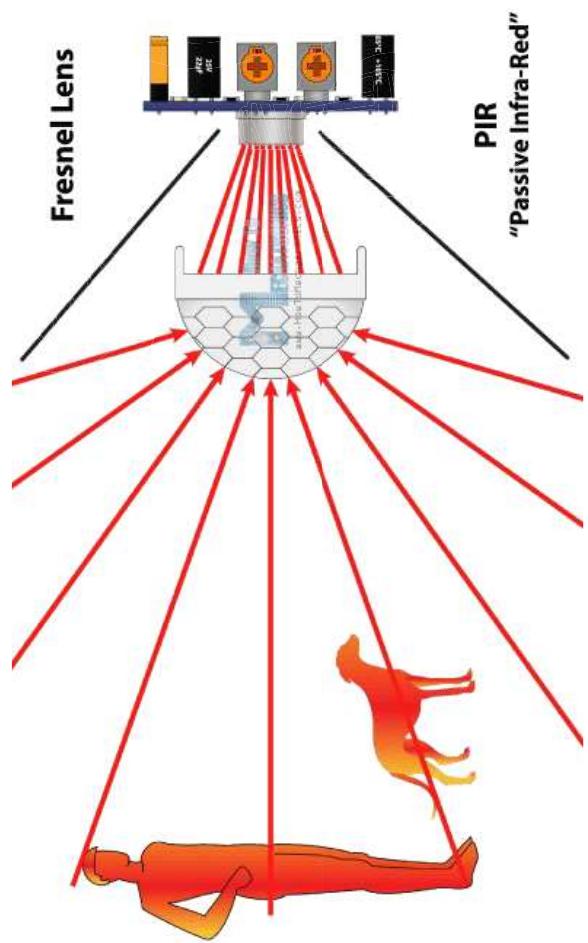
```
void setup() {  
  pinMode(A0,INPUT);  
  pinMode(9,OUTPUT);  
  Serial.begin(9600);  
}  
  
void loop() {  
  int pot_value = analogRead(A0);  
  int led_value=map(pot_value,0,1023,0,255);  
  analogWrite(9,led_value);  
  Serial.println(led_value);  
}
```

**CAN WE DO
SOME
MEANINGFUL ?**

Task 10. Interface PIR sensor with arduino and show the conditions using LEDs

PIR stand for Passive Infrared
PIR sensor sense the heat emitted by body which is in form of Infrared.
PIR sensor detect the change in infrared emission.

It is used to detect the of Motion of Humans and Animals.



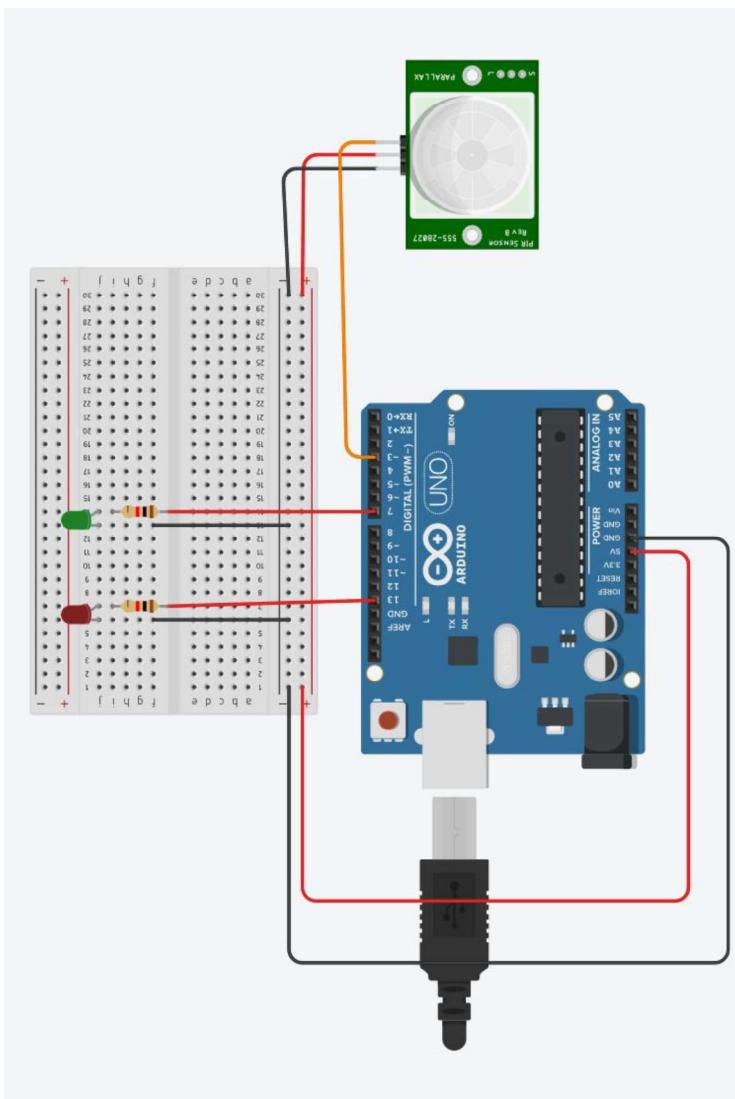
```

int red=13;
int green=7;
int pir=3;

void setup()
{
    pinMode(red, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(pir, INPUT);
}

void loop()
{
    int input=digitalRead(pir);
    if(input==1){
        digitalWrite(red,HIGH);
        digitalWrite(green,LOW);
    }
    else{
        digitalWrite(red,LOW);
        digitalWrite(green,HIGH);
    }
}

```



Task 11. Interface LED with arduino and control it using serial monitor.

Arduino has Serial buffer which can store data upto 64 Bytes.

Serial.available() => It return the number of character in serial buffer

Serial.read() => It reads the content of serial buffer.

```
int red=10;
int green=12;
char input;
void setup() {
  pinMode(red,OUTPUT);
  pinMode(green,OUTPUT);
  Serial.begin(9600);
}
void loop() {
  if(Serial.available()>0){
    input=Serial.read();
  }
  if(input=='R'){
    digitalWrite(red,HIGH);
    digitalWrite(green,LOW);
  }else if(input=='G'){
    digitalWrite(red,LOW);
    digitalWrite(green,HIGH);
  }else{
    digitalWrite(red,LOW);
    digitalWrite(green,LOW);
  }
}
```

Task 12. Interface Servo motor with arduino and control its rotation.

```
#include <Servo.h>

Servo myservo;
int pos = 0;
void setup() {
  myservo.attach(9);
}
void loop() {
  for (pos = 0; pos <= 180; pos += 1) {
    myservo.write(pos);
    delay(15);
  }
  for (pos = 180; pos >= 0; pos -= 1) {
    myservo.write(pos);
    delay(15);
  }
}
```



Brown Wire (GND)

Red Wire (VCC)

Orange Wire (PWM)

Output Shaft

Task 13. Interface Ultrasonic sensor with arduino and measure the distance of object



```
int trigPin = 2;
int echoPin = 3;
long duration;
int distance_cm;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance_cm = duration * 0.0343 / 2;

  Serial.print("Distance: ");
  Serial.print(distance_cm);
  Serial.println(" cm");

  delay(100);
}
```