Automating Street Lights for Sustainability

Objective

- Energy-saving by dimming when unnecessary.
- Cost-e ective solution for urban areas.
- Automatic lights adjust to brightness.
- Sensors detect light changes.
- Enhances safety for pedestrians, drivers.

Resources:

Physical

- Arduino Uno
- IR sensors
- 10mm white LEDs
- Connecting wires and breadboard
- Enclosure or casing

Financial

- Budget for purchasing the physical resources
- Funds for any additional components or tools required for installation or maintenance
- Budget for any unexpected costs or revisions to the project plan

Human

- Project Manager
- Electronics Engineer
- Software Developer
- Electrician Maintenance Technician
- Volunteers or community members

Budget

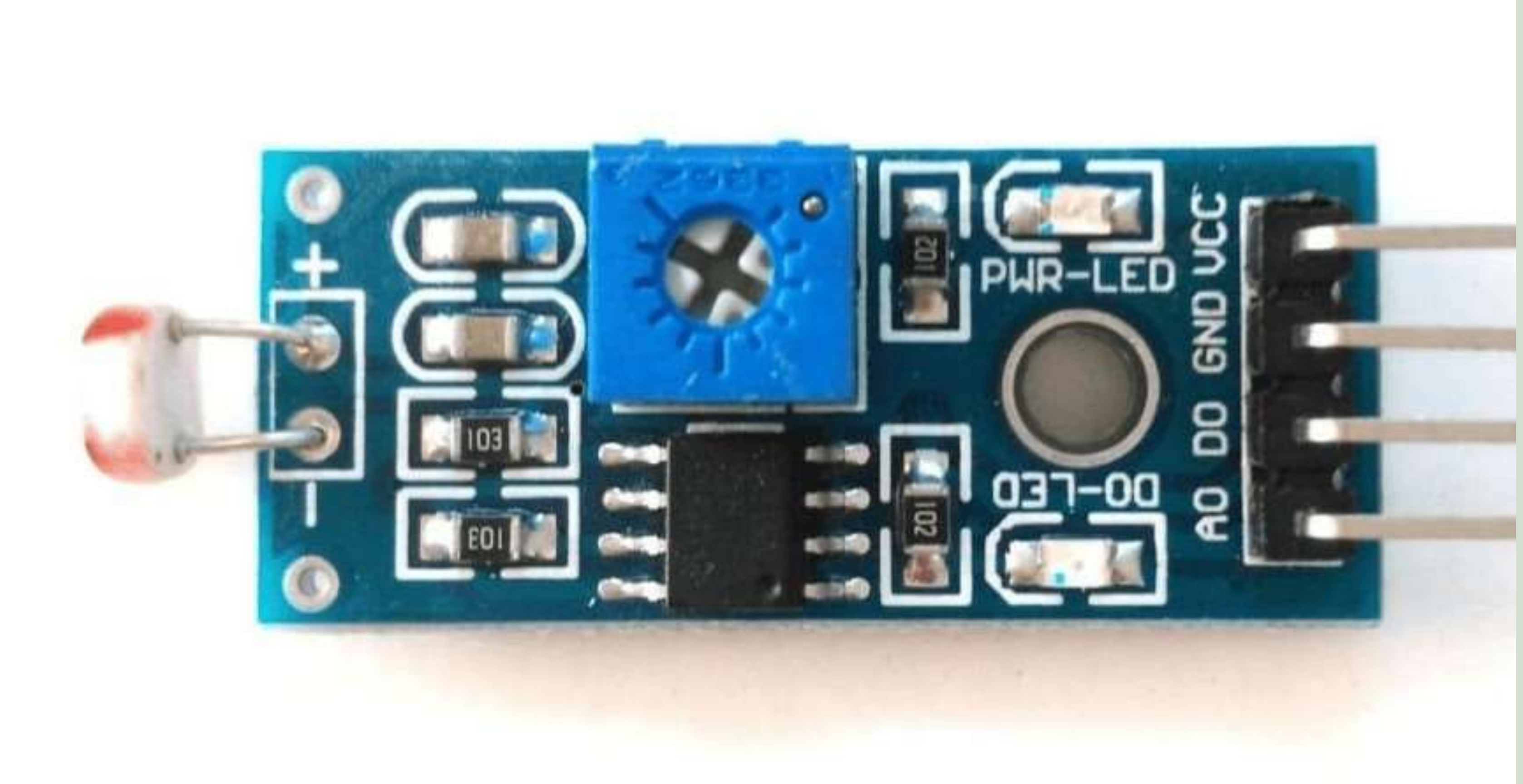
Component	Cost per Unit (₹)	Quantity	Total Cost Range (₹)
Hardware Costs			
Microcontroller: Arduino Uno	1,000 - 1,500	1	1,000 - 1,500
Sensors: IR Sensors	100 - 200	6	600 - 1,200
Sensors: LDR Modules	50 - 100	1	50 - 100
LEDs: 10mm White LEDs	10 - 20	5	50 - 100
Miscellaneous: Connecting Wires	100 - 200	10	1,000 - 2,000
Additional Components			
Breadboard	100 - 200	1	100 - 200
Resistors	20 - 50	1	20 - 50
Power Supply/Batteries	200 - 500	1	200 - 500
Software Development			
Arduino IDE	Free	-	0
Software and Coding Assistance	1,000 - 2,000	-	1,000 - 2,000

Summary of Total Costs:

Category	Total Cost Range (₹)
Hardware Costs	2,700 - 4,900
Additional Components	320 - 750
Software Development	1,000 - 2,000
Grand Total	**4,020 - 7,650**

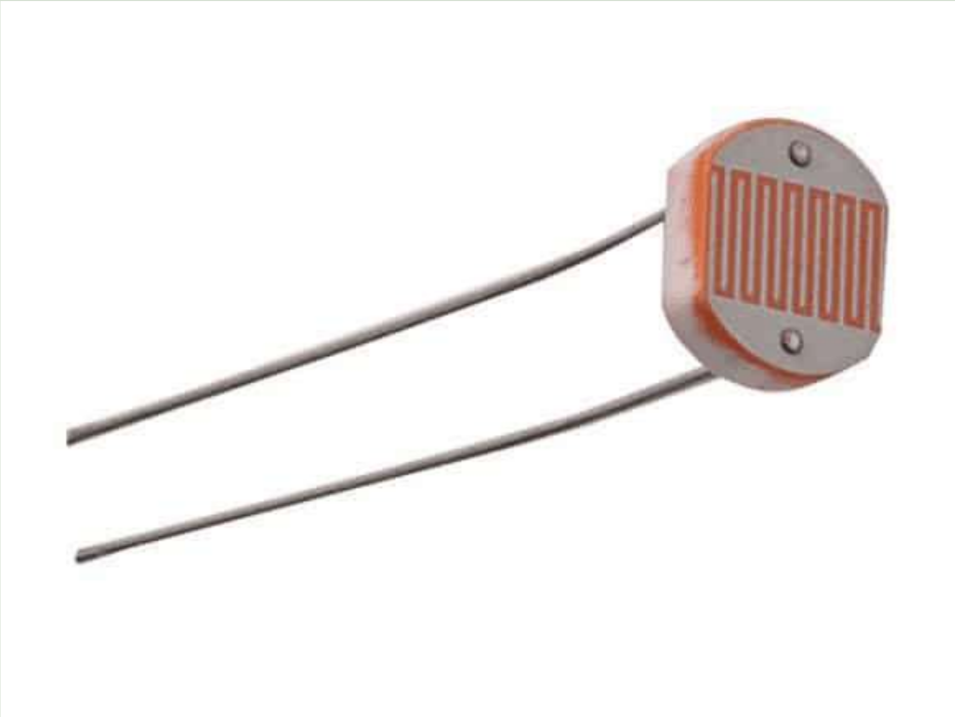
LDR MODULE

- LDR stands for Light Dependent Resistor.
- It functions based on photoconductivity
- Resistance diminishes with light intensity rise.
- This unique characteristic makes it adaptable.
- Essential in light-sensing applications.

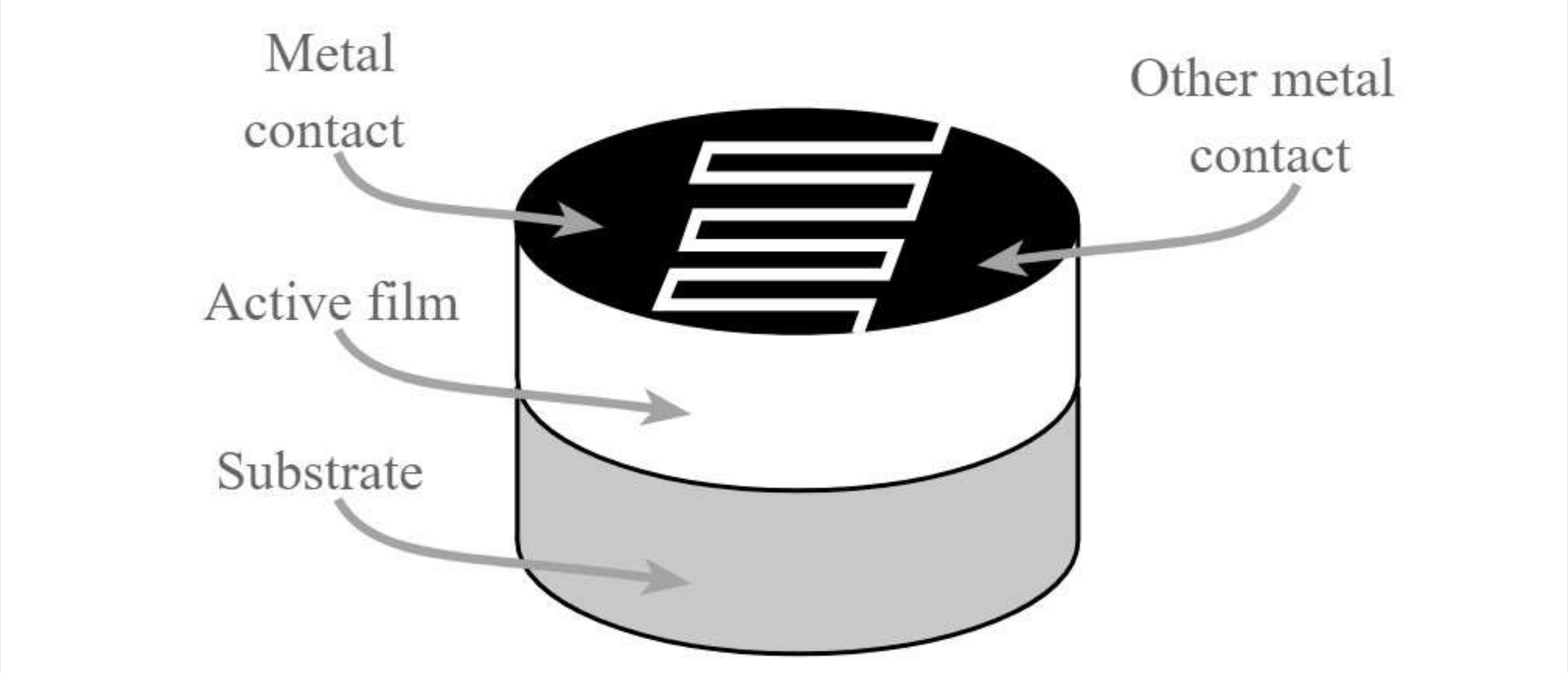


LDR

LDR is an acronym forLight Dependent Resistor. LDRs are tiny light-sensing devices also known asphotoresistors.



LDR structure

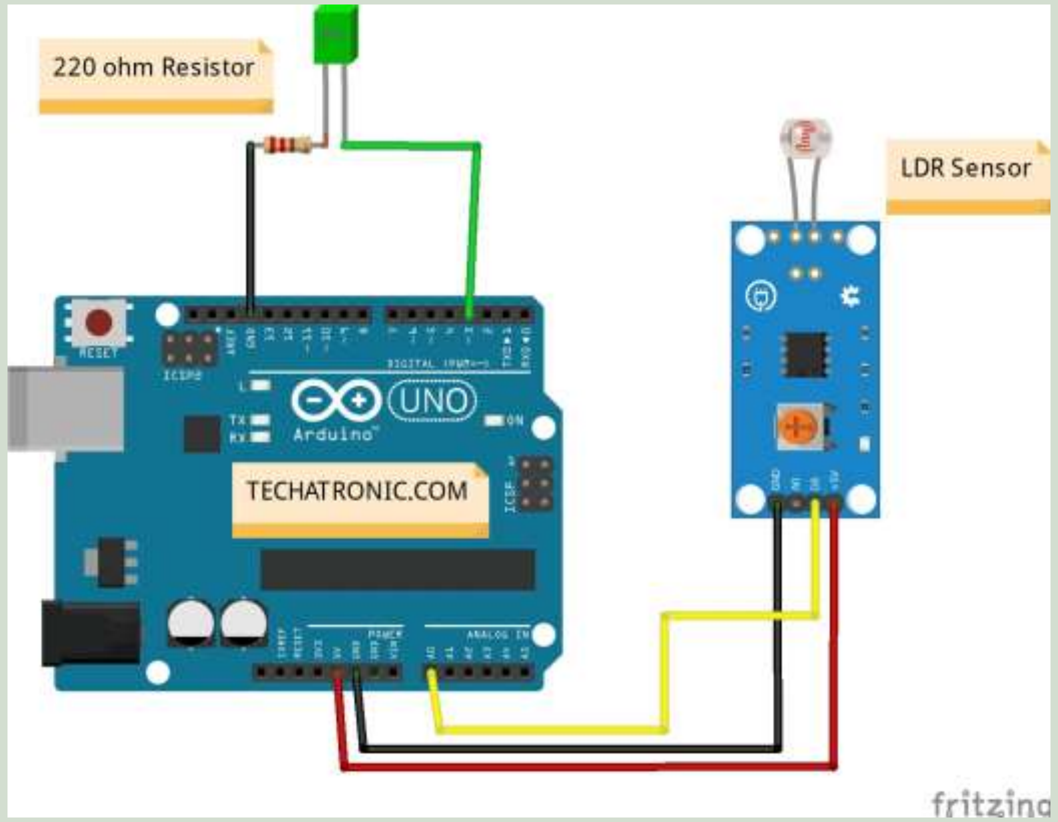


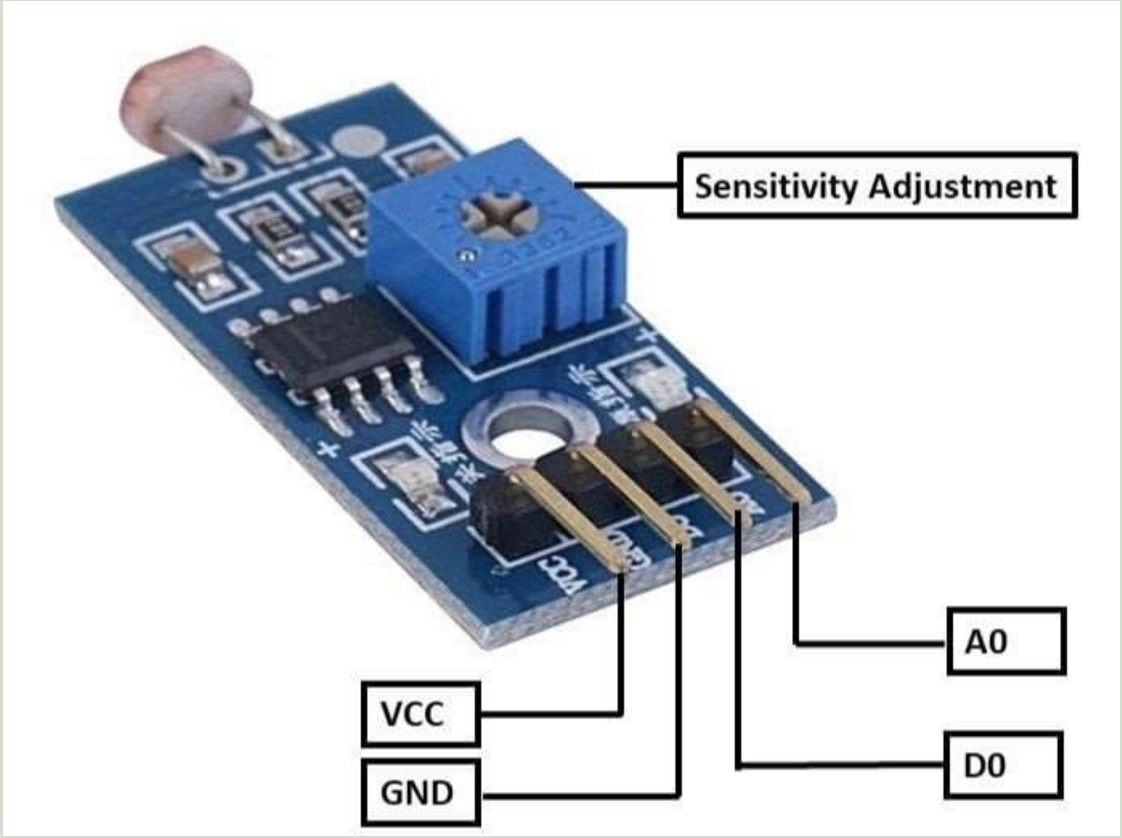
Working Model of LDR Module

- The LDR module functions as the eyes of automatic street lights, sensing changes in ambient light levels. Its Light Dependent Resistor adjusts its resistance based on light intensity. Connected to a control circuit, it triggers street lights to turn on when darkness falls below a set threshold.

- Conversely, when light levels rise above the threshold, the street lights switch off, optimizing energy usage. This real-time feedback loop ensures prompt response to environmental changes, enhancing reliability and energy efficiency.

- Overall, the LDR module plays a pivotal role in creating a smart and sustainable urban lighting infrastructure.



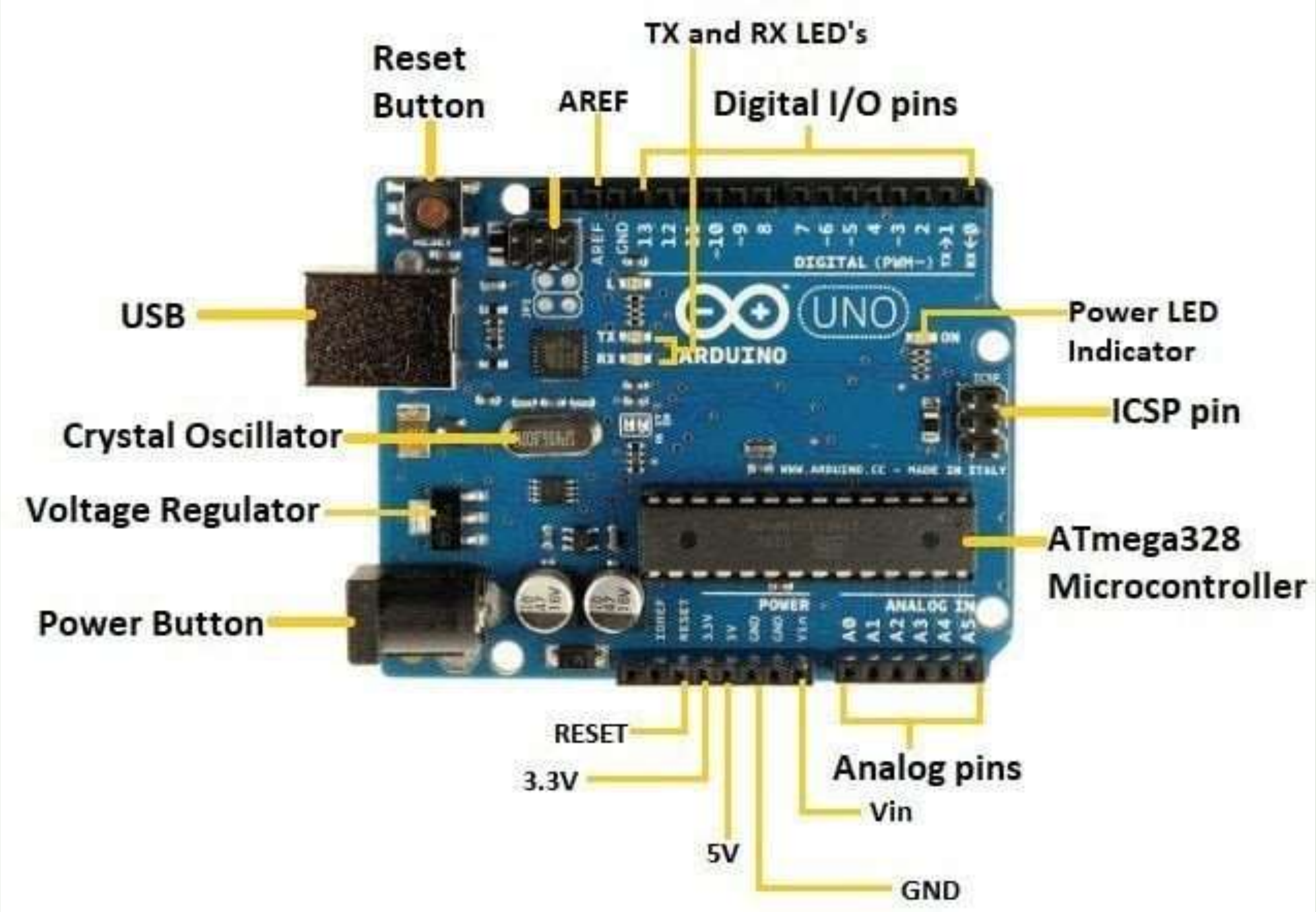


ARDUINO UNO

- Arduino UNO is based on an ATmega328Pmicrocontroller. It is easy to use compared to other boards, such as the Arduino Mega board, etc.
- The board consists of digital and analog Input/Output pins (I/O), shields, and other circuits.

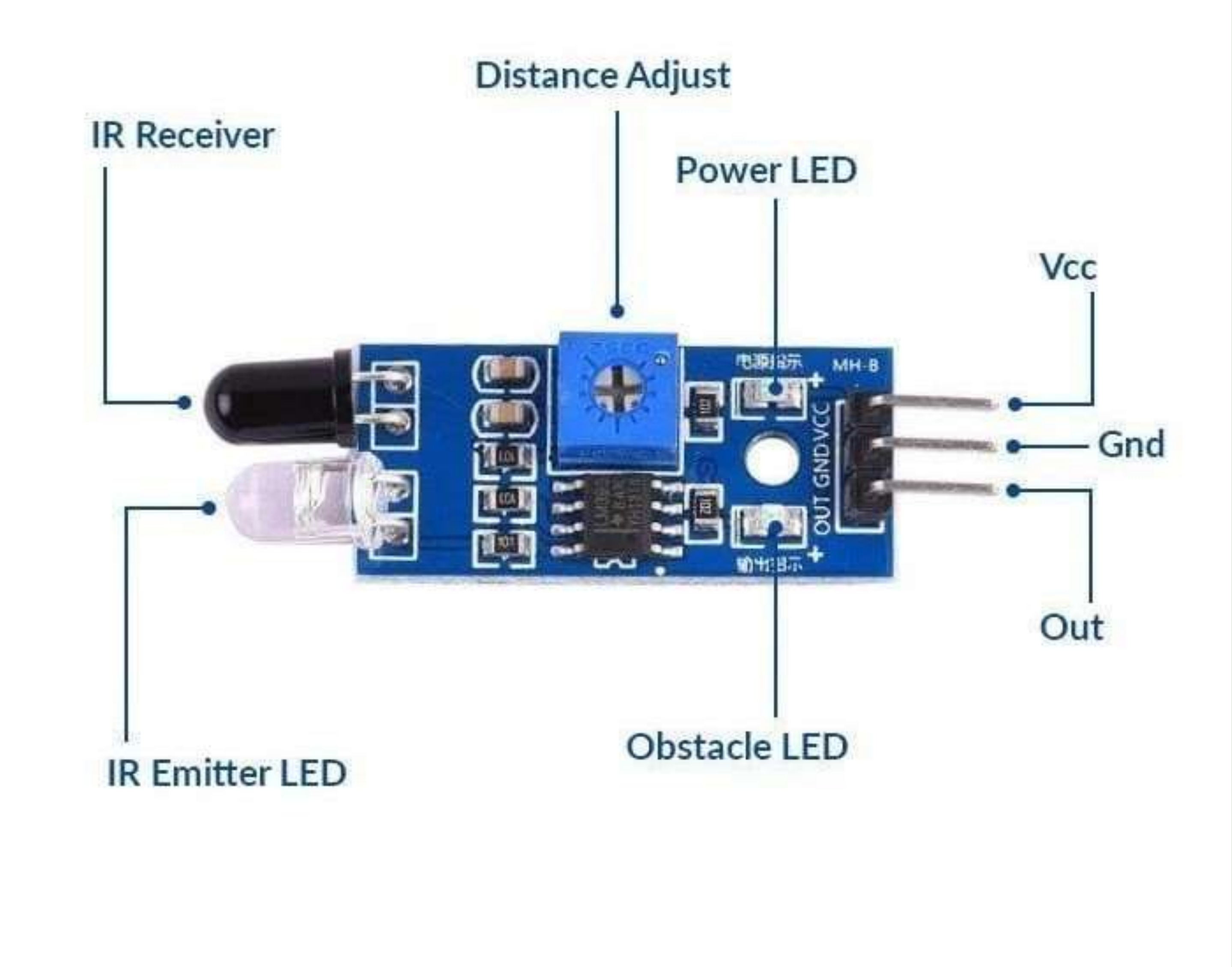
Working Model of Arduino Uno

- The Arduino Uno operates by running code written in the Arduino IDE on a connected computer
- which is then uploaded to the board via USB.
- The microcontroller executes the code, interacting with sensors, actuators, and other components connected to its input/output pins.



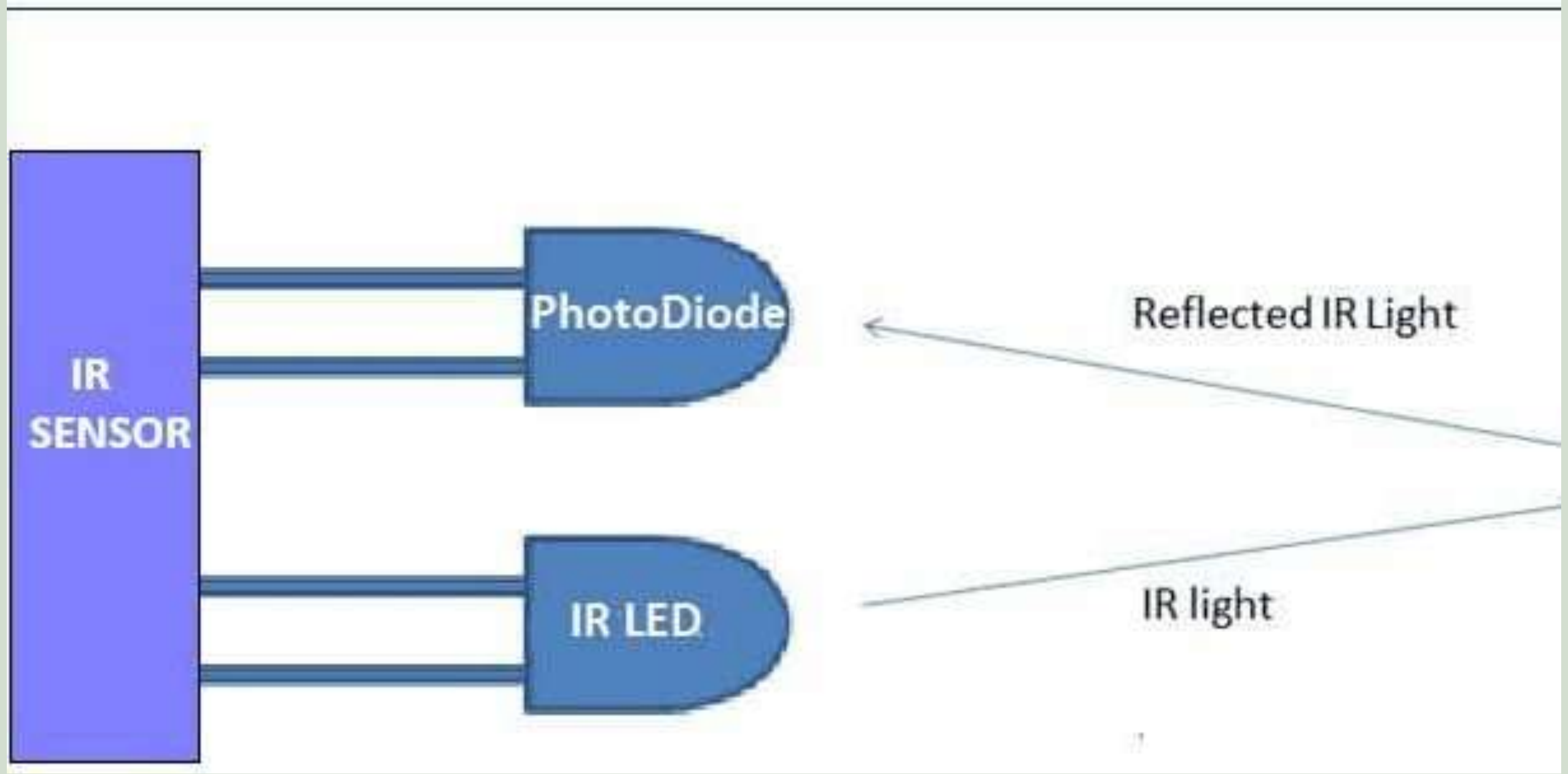
IR - SENSORS

IR sensors, short for Infrared sensors, are devices designed to detect and measure infrared radiation in the form of heat emitted by objects in their vicinity. These sensors typically consist of an emitter, which emits infrared light, and a receiver, which detects the reflected or emitted infrared radiation.



Working Model of IR Sensor

- IR sensors emit and detect infrared radiation from objects nearby, converting it into electrical signals.
- These signals are processed to determine object presence.
- They offer non-contact detection for various applications like motion sensing and temperature measurement.

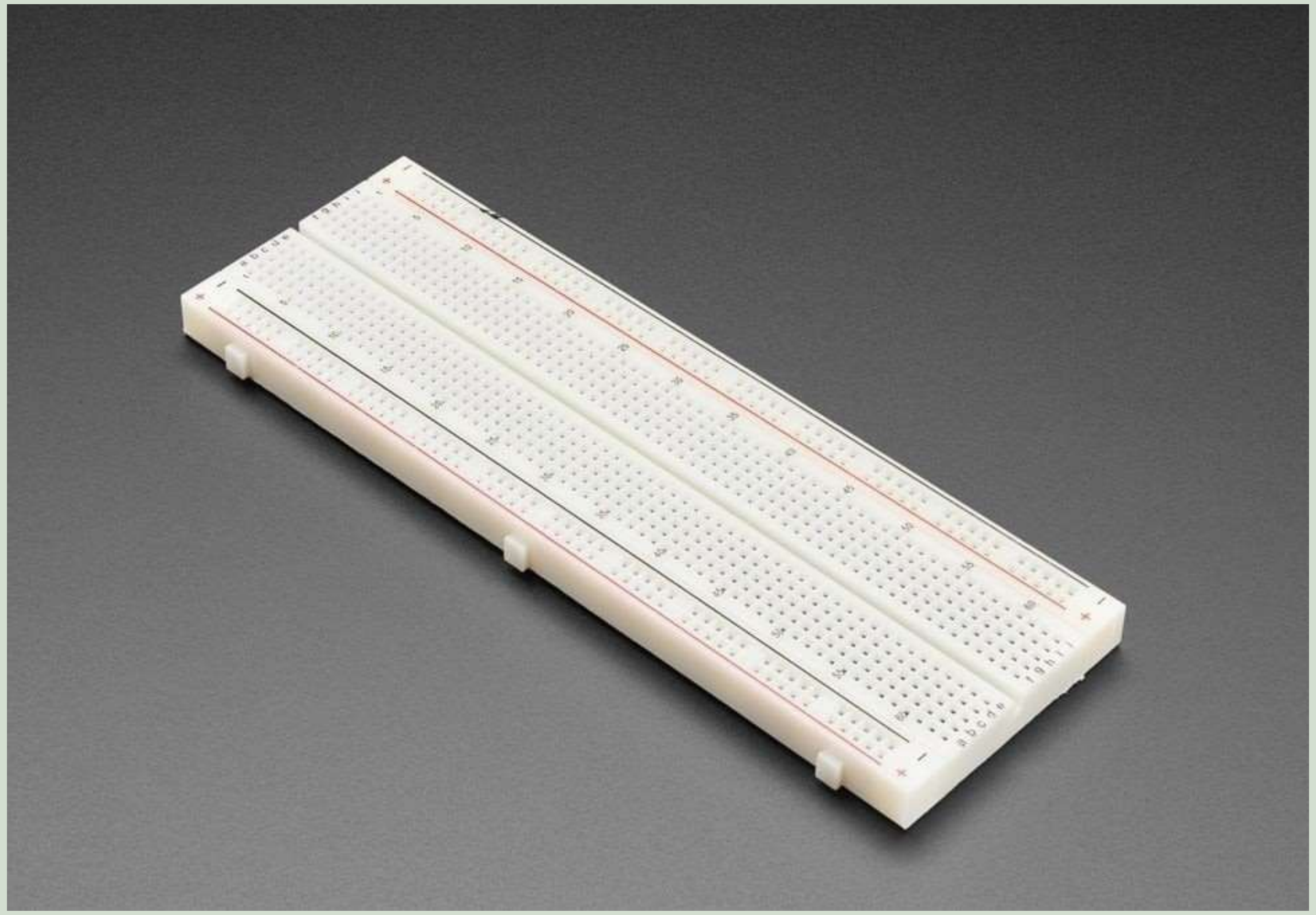


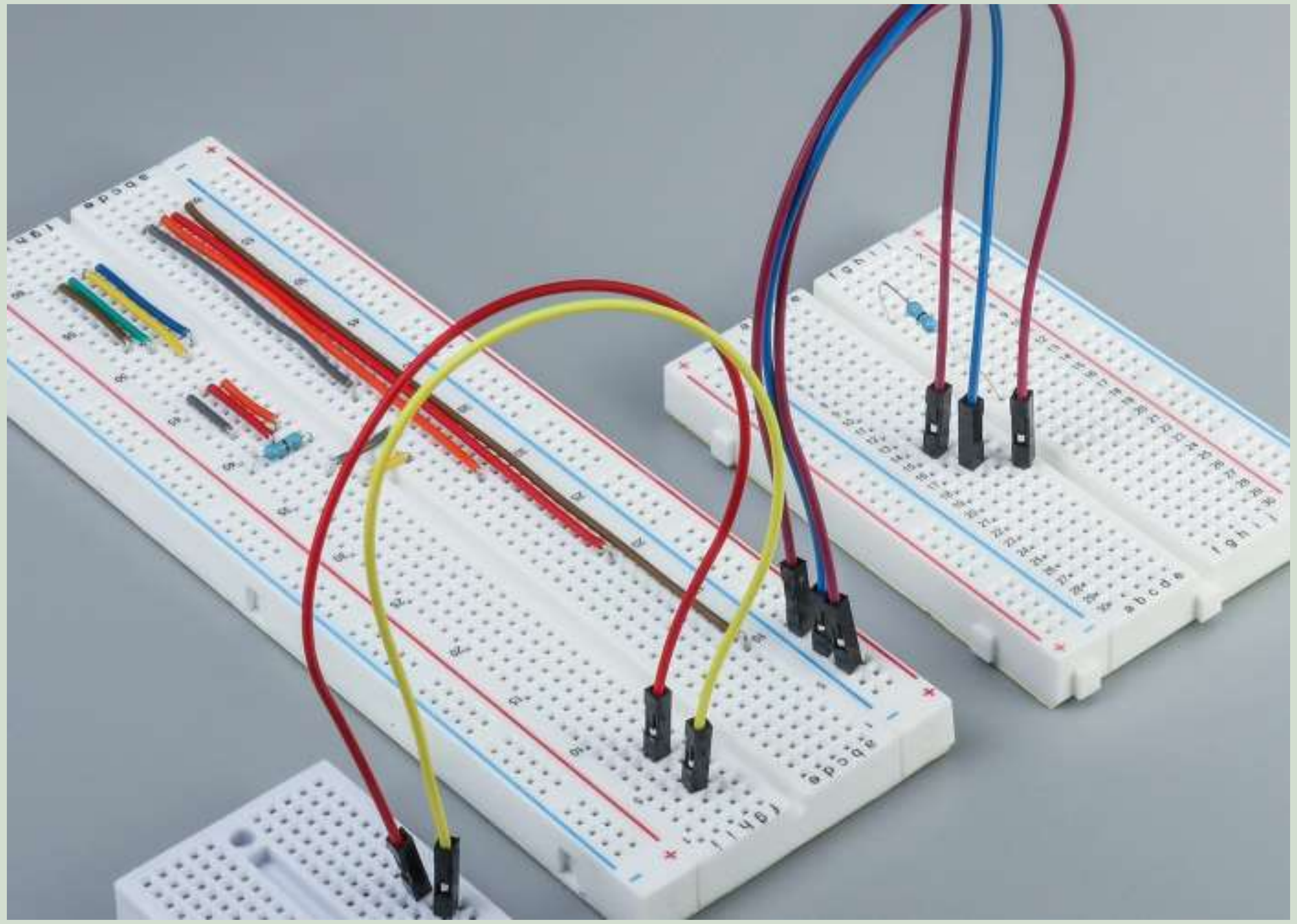
Breadbord

A breadboard (sometimes called protoboard) is essentiallythe foundation to construct and prototype electronics. A breadboard allows for easy and quick creation of temporary electronic circuits or to carry out experiments with circuit design.

Breadboards are usually divided into four sections,

- Two outer sections
- Two inner sections.

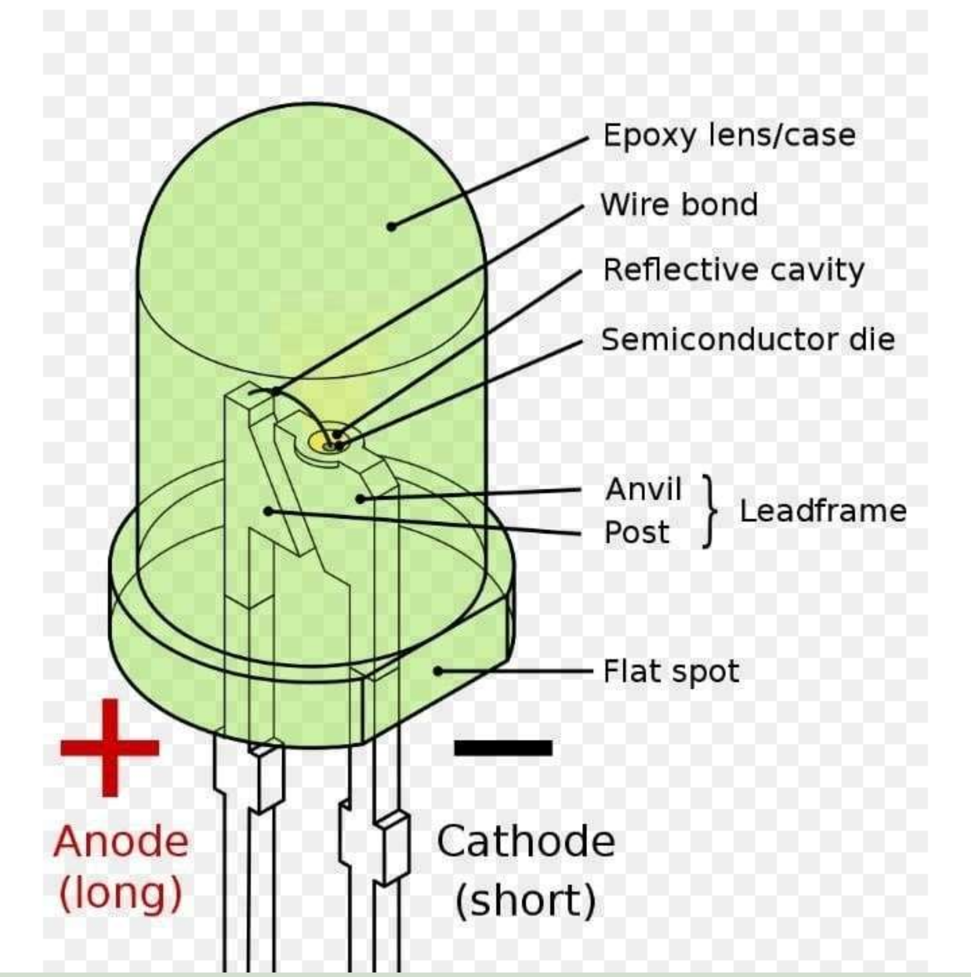




LED Lights

A light-emitting diode (LED) is a semiconductor device that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons.

The color of the light is determined by the energy required for electrons to cross the band gap of the semiconductor .White light is obtained by using multiple semiconductors or a layer of light-emitting phosphor on the semiconductor device.



Connecting Wires

Connecting wires are used for making connections between items on your breadboard and your Arduino’s header pins. Use them to wire up all your circuits

Arduino uno cable

The Arduino Uno cable is a type B USB cable, which is used to connect the Arduino Uno board to a computer for programming and power supply. It's a standard USB cable that can be used to transfer data and power between the board and the computer. The cable is also used to upload sketches to the board and to communicate with the board's serial monitor.

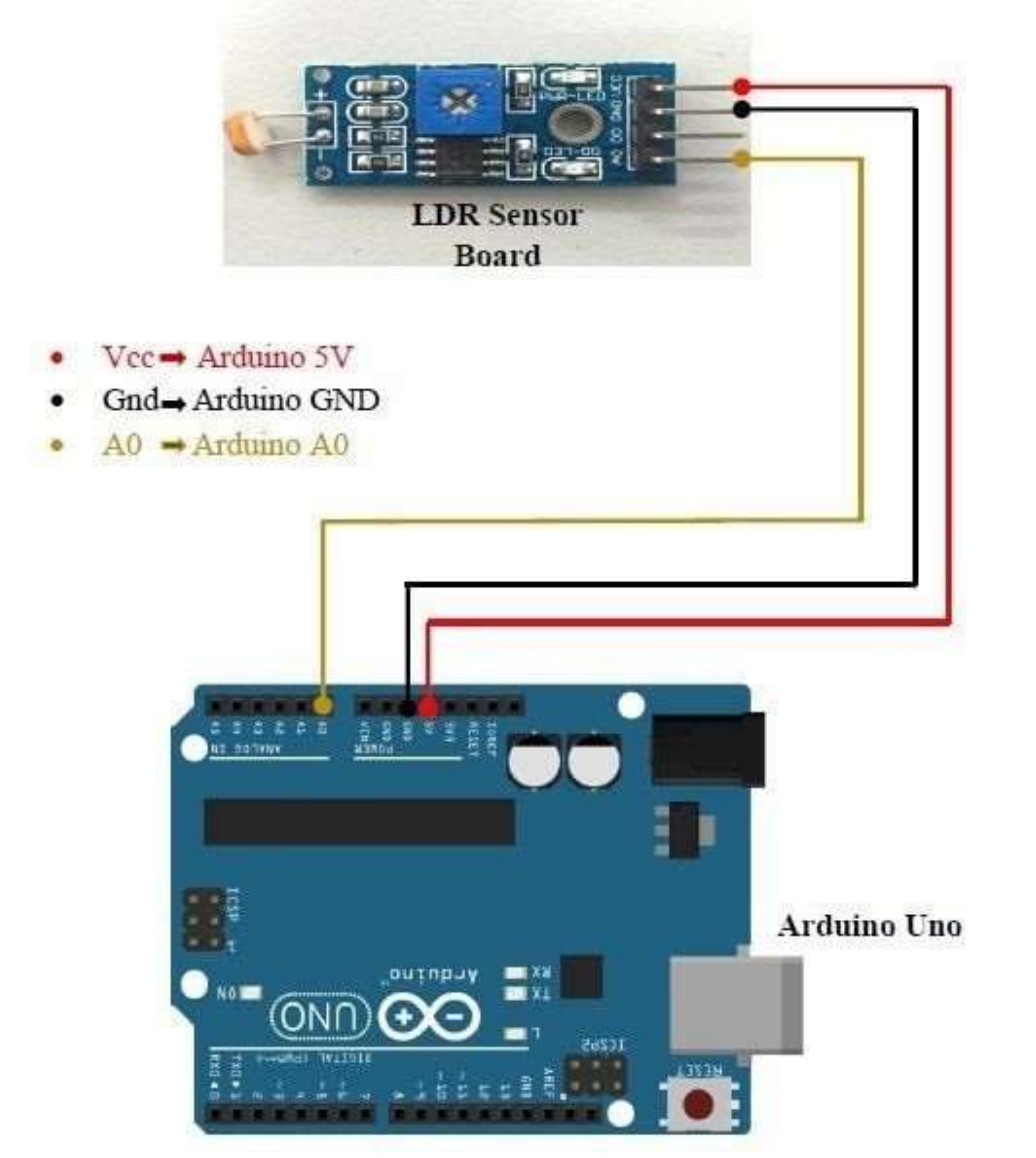
Step by Step process:

1. Setting up the LDR Module
2. Setting up the IR Sensor
3. Setting up the LEDs
4. Connecting the Breadboard

Setting up the LDR Module

- It changes its resistance based on the light intensity. More light decreases its resistance and less light increases it.

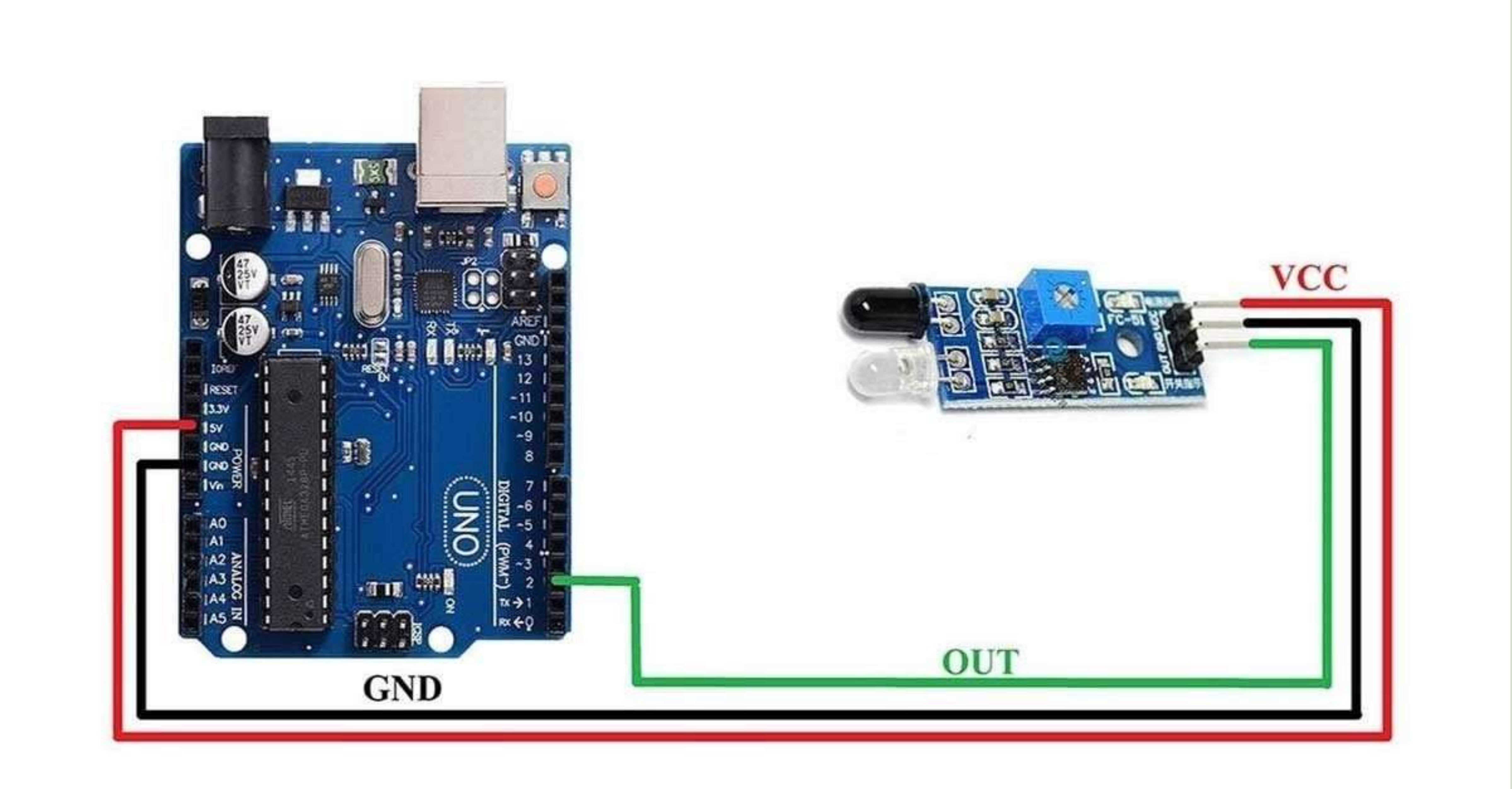
- Connections: One end of the LDR is connected to the 5V pin on the Arduino.
  The other end is connected to an analog input pin (A0) on the Arduino and also connected to a 10kΩ resistor, which is then connected to the GND pin.



Setting up the IR Sensor

- It detects the presence of an object.

- Connections: VCC pin of the IR sensor to 5V pin on the Arduino. GND pin of the IR sensor to the GND pin on the Arduino. Output pin of the IR sensor to a digital pin (D2) on the Arduino.

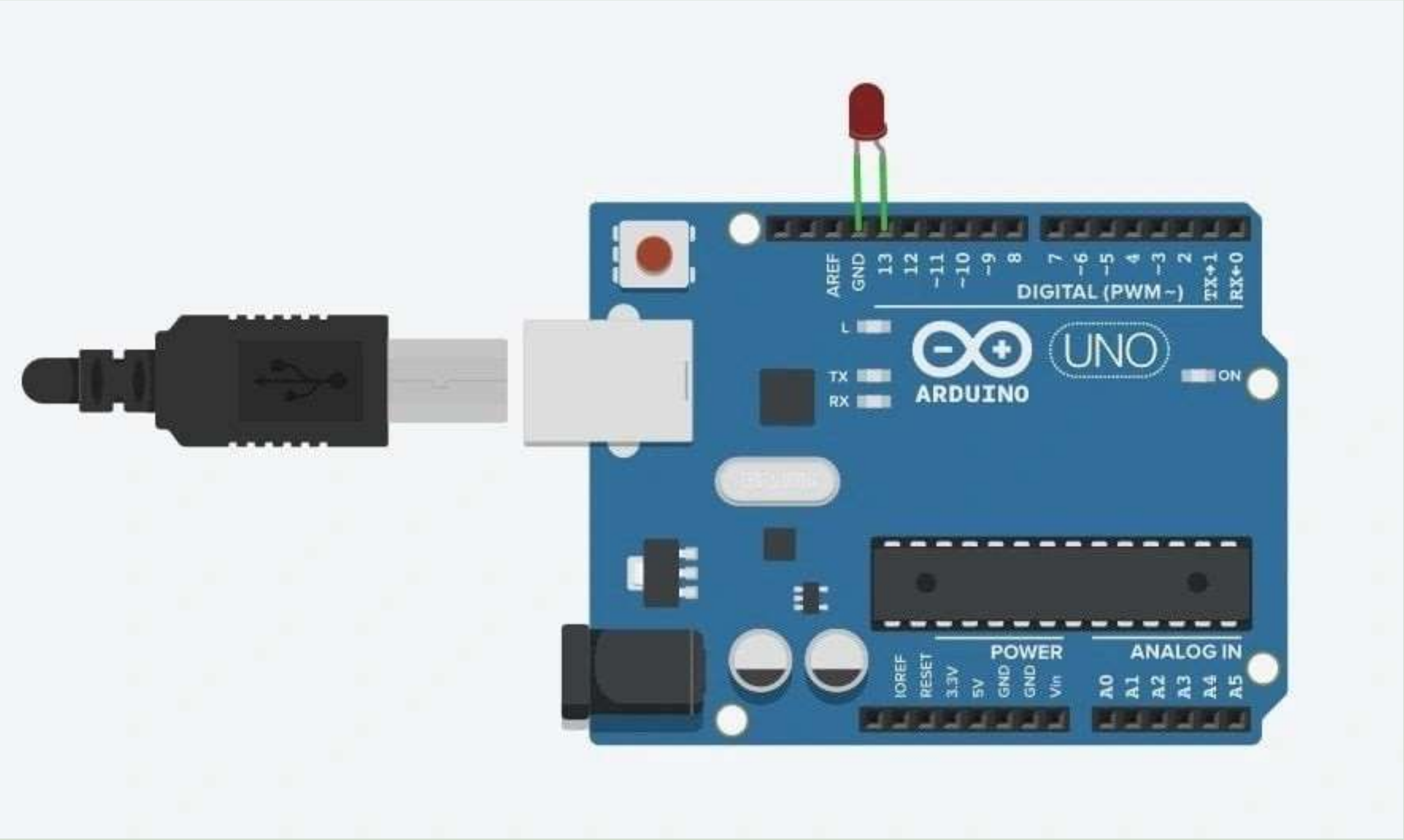


Setting up the LEDs

- White LEDs: Used as the street lights.

- Connections:
  Connect the positive leg (anode) of the LED to a digital pin (D9) on the Arduino through a 220Ω resistor.

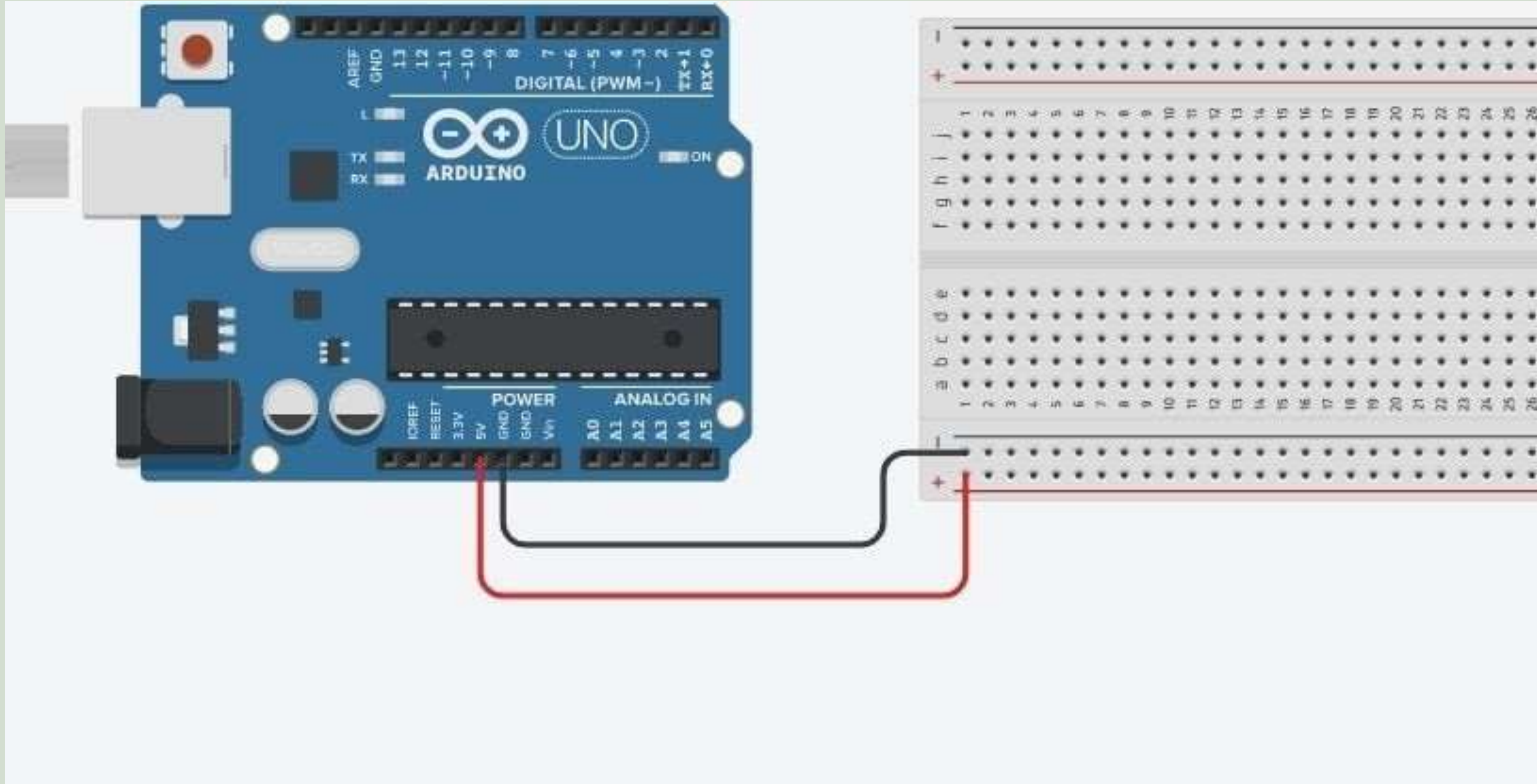
- Connect the negative leg (cathode) of the LED to the GND pin on the Arduino.



Connecting the Breadboard:

- Use the breadboard to organize the connections neatly.

- Connect the power and ground rails on the breadboard to the 5V and GND pins on the Arduino, respectively.



Code & descriptions

Pin Assignments:

Description:

- IR1, IR2, IR3 are the pins connected to the IR sensors.
- LDR is the pin connected to the LDR.
- led1, led2, led3 are the pins connected to the LEDs.
- val1, val2, val3, val4 are variables to store the readings from the sensors.

Code

```
int IR1 =8;
int IR2 =12;
int IR3 =13;
int LDR =7;
int led1 =3;
int led2 =5;
int led3 =6;
int val1;i
nt val2;
int val3;i
nt val4;
```

Setup Function:

Description:

- Sets the IR sensor pins (IR1, IR2, IR3) and the LDR pin (LDR) as inputs.

- Sets the LED pins (led1, led2, led3) as outputs.

Code

```
void setup()
{
pinMode(IR1,INPUT);
pinMode(IR2,INPUT);
pinMode(IR3,INPUT);
pinMode(LDR,INPUT);
pinMode(led1,OUTPUT);
pinMode(led2,OUTPUT);
pinMode(led3,OUTPUT);
Serial.begin(9600);
}
```

Loop Function

Description

- Reads the values from the IR sensors and LDR.

- These values are stored in val1, val2, val3, and val4.digitalRead reads the state of the pins (HIGH or LOW).

Code

```
void loop() {
val1 =digitalRead(IR1);
val2 =digitalRead(IR2);
val3 =digitalRead(IR3);
val4 =digitalRead(LDR);
Serial.print("val1:" );
Serial.println(val1)
;delay(50);
Serial.print("val4:" );
Serial.println(val4);
Serial.print("val2:" );
Serial.println(val2);
delay(50);
Serial.print("val3:" );
Serial.println(val3);
```

Conditional Logic

* Condition 1

```
if(val1==1&&val4==0&&val2==1&&val3==1)
{
    digitalWrite(3,LOW);
    digitalWrite(5,LOW);
    digitalWrite(6,LOW);
}
```

Description

- If IR1 detects an object (val1 ==1), LDR indicates darkness (val4 ==0), and both IR2 and IR3 detect objects (val2 ==1 and val3 ==1).
- Turn off all LEDs.

*Condition 2

```
elseif(val1==1&&val4==1&&val2==1&&val3==1)
{
    analogWrite(3,20);
    analogWrite(5,20);
    analogWrite(6,20);
}
```

Description

- If IR1 detects an object (val1 ==1), LDR indicates light (val4 ==1), and both IR2 and IR3 detect objects (val2 ==1 and val3 ==1).
- Set all LEDs to a low brightness (analogWrite with value 20).

*Condition 3

```
elseif(val1==0&&val4==1&&val2==1&&val3==1)
{
     analogWrite(3,500);
     analogWrite(5,20);
     analogWrite(6,20);
}
```

Description

- If IR1 does not detect an object (val1 ==0), LDR indicates light (val4 ==1), and both IR2 and IR3 detect objects (val2 ==1 and val3 ==1).

- Set led1 to a higher brightness (500), and led2 and led3 to a low brightness (20).

*Condition 4

```
else if(val1==1&&val4==1&&val2==0&&val3==1)
{
    analogWrite(3,20);
    analogWrite(5,500);
    analogWrite(6,20);
}
```

Description:

- If IR1 detects an object (val1 == 1), LDR indicates light (val4 == 1), IR2 does not detect an object (val2 ==0), and IR3 detects an object (val3 ==1).
- Set led2 to a higher brightness (500), and led1 and led3 to a low brightness (20).

* Condition 5

```
else if(val1==1&&val4==1&&val2==1&&val3==0)
{
    analogWrite(3,20);
    analogWrite(5,20);
    analogWrite(6,500);
}
```

Description:

- If IR1 detects an object (val1 ==1), LDR indicates light (val4 ==1), IR2 detects an object (val2 ==1), and IR3 does not detect an object (val3 ==0).
- Set led3 to a higher brightness (500), and led1 and led2 to a low brightness (20).

CODE

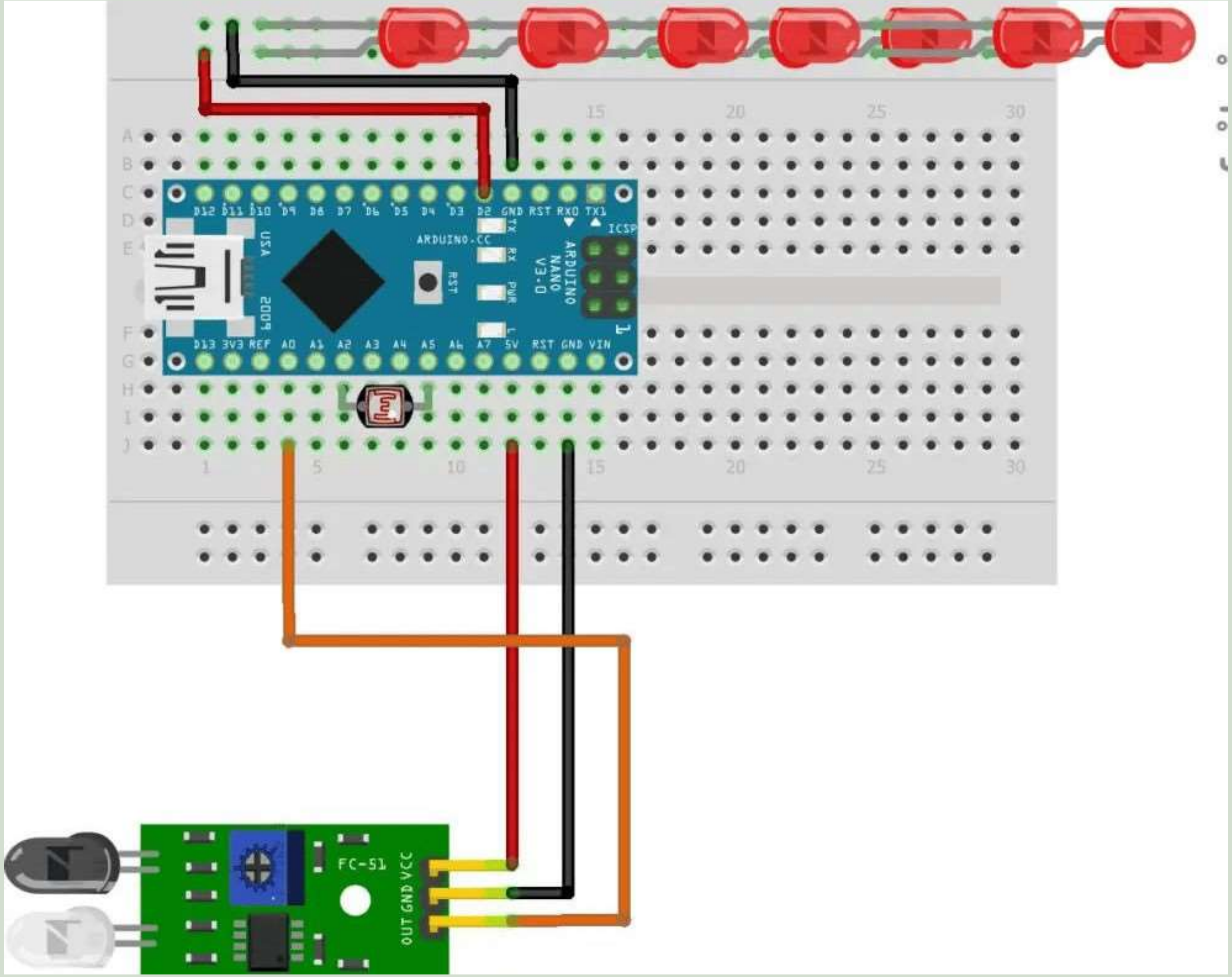
sketch_may25a.ino

```
int IR1 = 8;
int IR2 = 12;
int IR3 = 13;
int LDR = 7;
int led1 = 3;
int led2 = 5;
int led3 = 6;
int val1;
int val2;
int val3;
int val4;

void setup()
{
pinMode(IR1,INPUT);
pinMode(IR2,INPUT);
pinMode(IR3,INPUT);
pinMode(LDR,INPUT);
pinMode(led1,OUTPUT);
pinMode(led2,OUTPUT);
pinMode(led3,OUTPUT);
Serial.begin(9600);

}

void loop() {
val1 = digitalRead(IR1);
val2 = digitalRead(IR2);
val3 = digitalRead(IR3);
val4 = digitalRead(LDR);
Serial.print("val1:" );
Serial.println(val1);
delay(50);
```

sketch_may25a.ino

```
Serial.println(val4);

Serial.print("val2:" );
Serial.println(val2);
delay(50);
Serial.print("val3:" );
Serial.println(val3);

if(val1==1&&val4==0&&val2==1&&val3==1)
{
digitalWrite(3,LOW);
digitalWrite(5,LOW);
digitalWrite(6,LOW);

}
else if(val1==1&&val4==1&&val2==1&&val3==1)
{
analogWrite(3,20);
analogWrite(5,20);
analogWrite(6,20);
}

else if(val1==0&&val4==1&&val2==1&&val3==1)
{
analogWrite(3,500);
analogWrite(5,20);
analogWrite(6,20);
}
else if(val1==1&&val4==1&&val2==0&&val3==1)
{
analogWrite(3,20);
analogWrite(5,500);
analogWrite(6,20);
}
else if(val1==1&&val4==1&&val2==1&&val3==0)
{
analogWrite(3,20);
analogWrite(5,20);
analogWrite(6,500);
}
}
```

CIRCUIT DIAGRAM



Solution

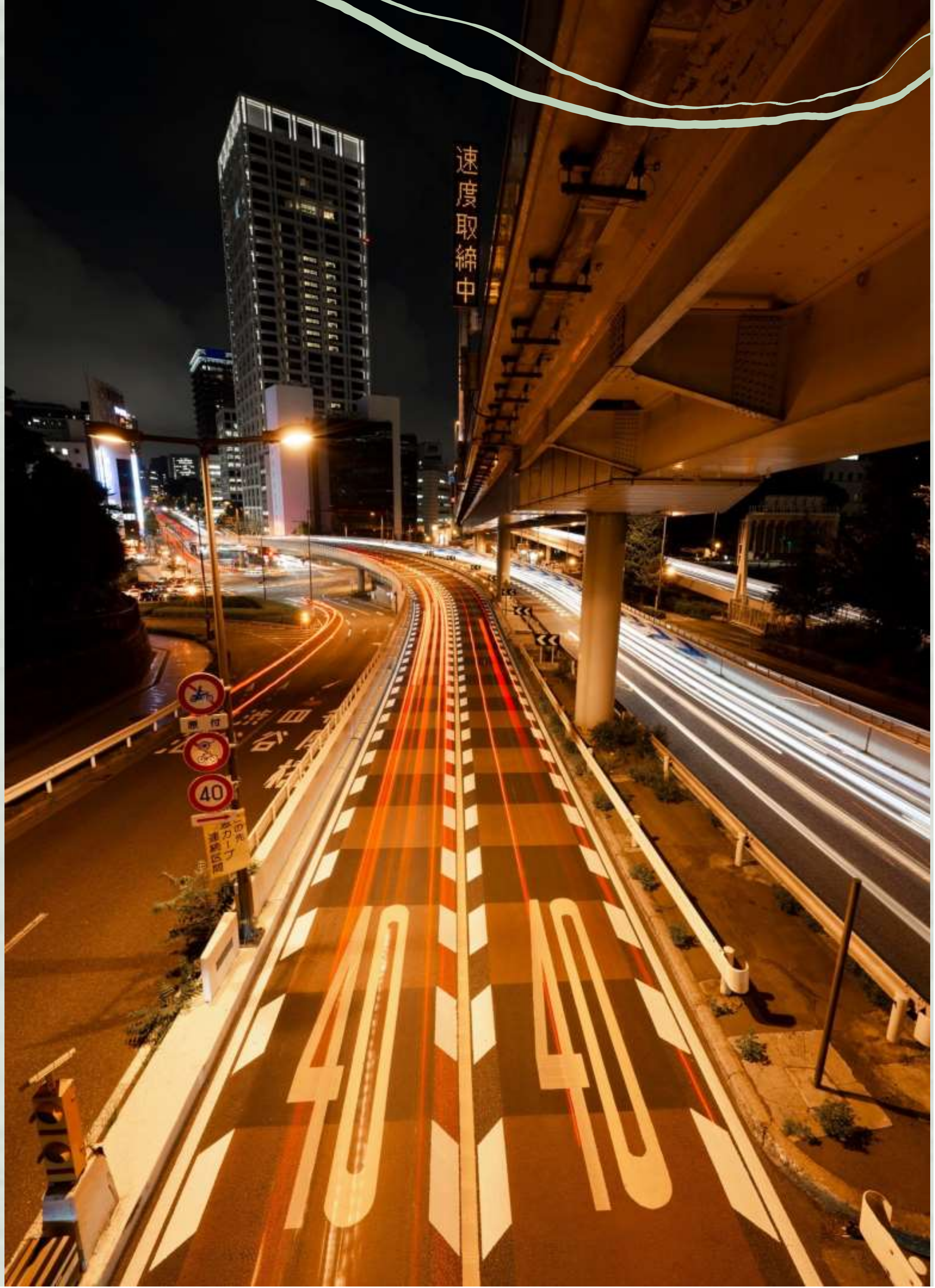
- Install light sensors and relays.
- Program microcontroller for light control.
- Connect LEDs to relays output.
- Test system under various conditions.
- Deploy and monitor for efficiency.

Execution plan:

- Define project scope and budget.
- Select appropriate sensors and microcontrollers.
- Select connectivity communication protocols.
- Design power supply solution accordingly.
- Test, deploy, monitor, and maintain.

Methodology

- Analyze requirements, constraints, and conditions
- Design system, select components, protocols.
- Develop, test prototype for validation.
- Deploy system, ensure integration.
- Monitor, maintain, gather feedback.



uses:

Automatic street lights improve safety.They save energy efficiently.Integration with smart city systems.Enhances urban infrastructure development.Contributes to livable urban environments.

Conclusion:

Implementing automatic street lights o ers a straightforward solution to enhance safety, conserve energy, and modernize urban infrastructure. By seamlessly integrating with smart city initiatives, this project not only improves the quality of urban life but also contributes to the sustainability and development of vibrant urban environments.

Thanks!