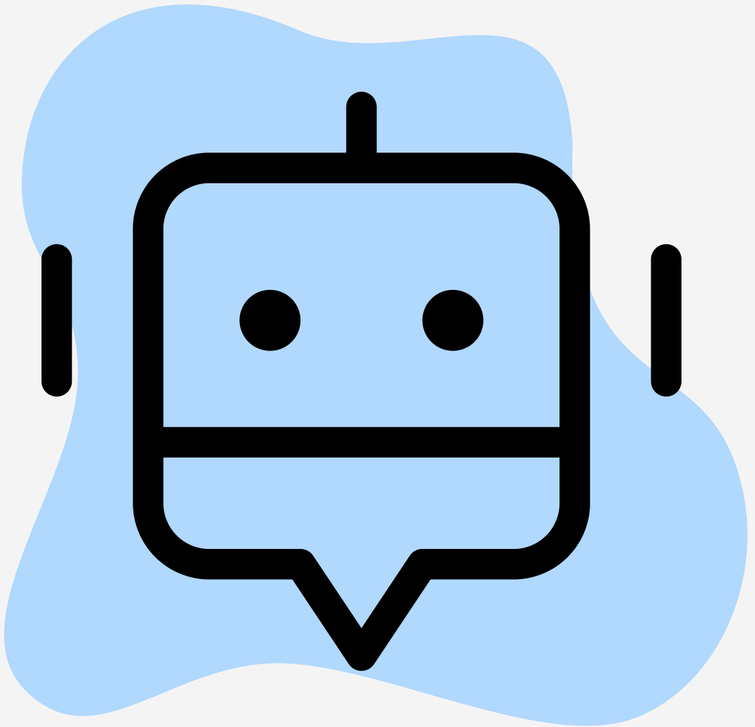
PassionBots

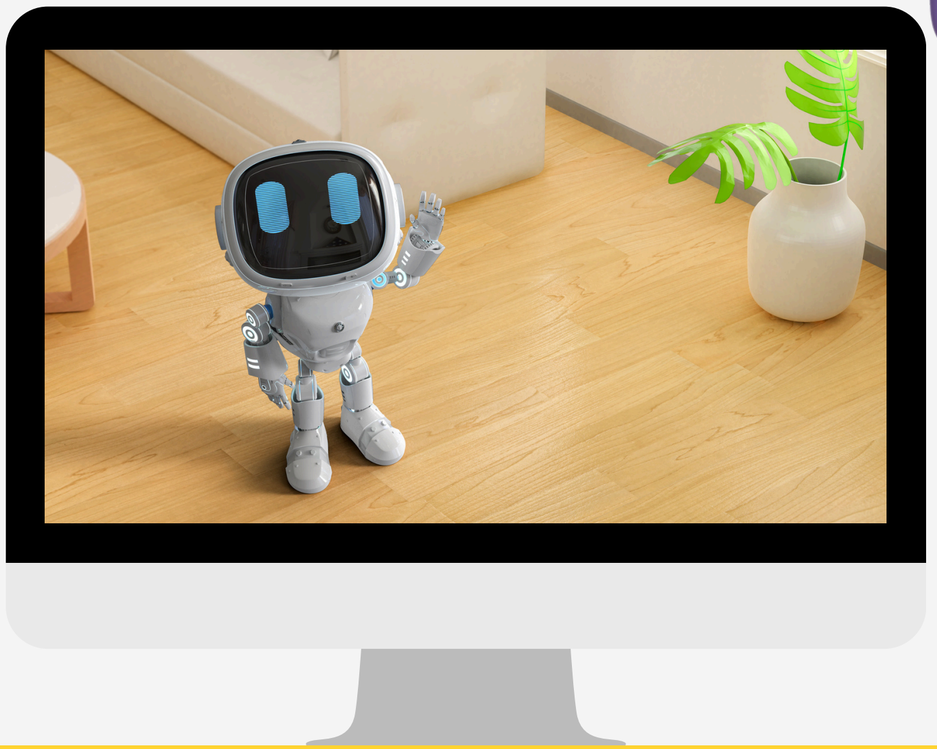
STUDENT'S GUIDE



Hello Passionbots Bot →

WWW.PASSIONBOTS.IN

Hello Pingal Bot



About The Class

AIM

Aim: Create a simple robotic project, "Hello Pingal Bot," using Arduino Uno.

Topics Covered

Introduction to robotics and basic robot designProgramming fundamentals using Arduino Uno

Time

120 minutes

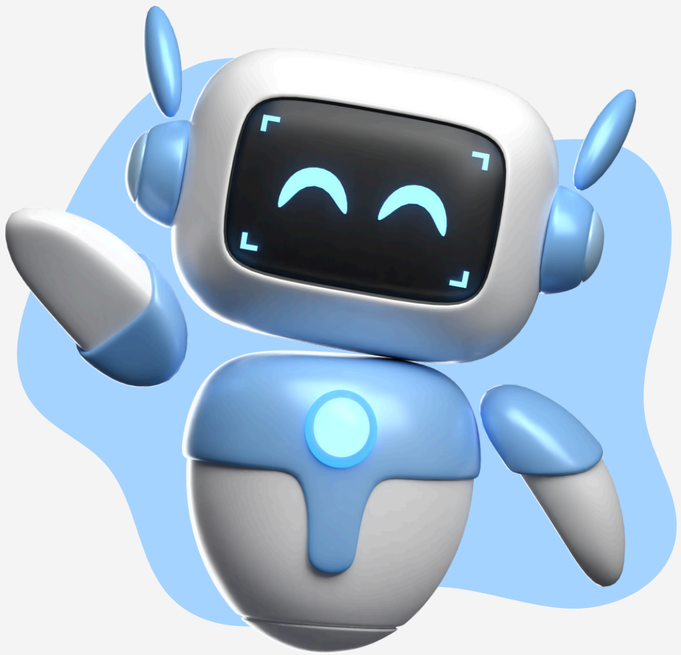
Tools Used

Arduino IDE

Benefits Of The Project

Robotics concepts and hardware integration

Hands-on experience with programming Arduino Uno



Understanding of sensor integration

Hands-On Learning

Introduction

What is Hello Thim Bot?

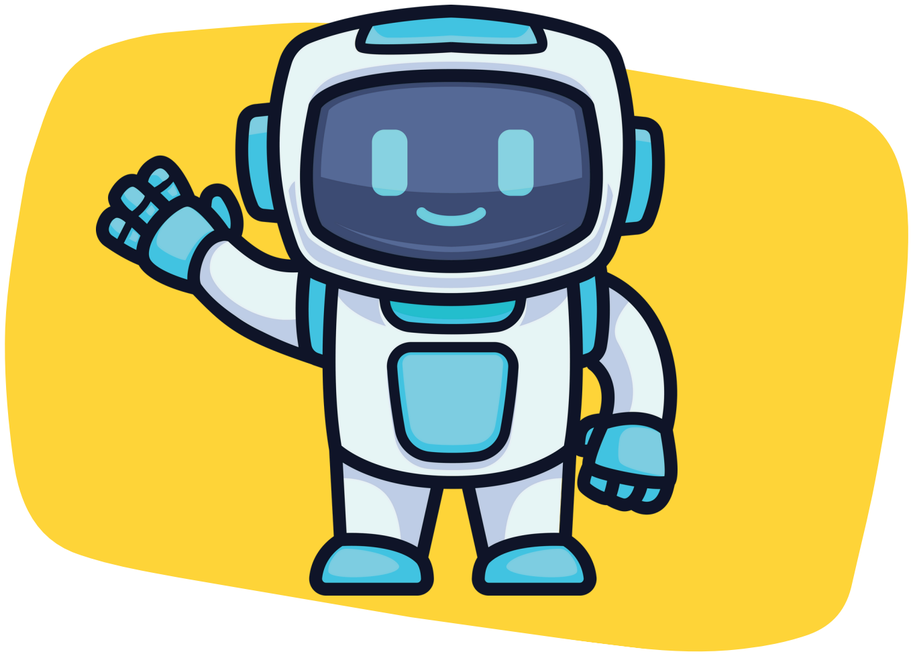
- **"Hello Pingal bot"** is an Arduino project where a robot comes to life using an Arduino Uno, a servo motor, and a **light sensor**. Its main purpose is to respond to by waving at us, infusing our technological creation with personality.

- It serves as an educational tool for learning about **robotics and programming.**

- It demonstrates the integration of **sensors and actuators** in an interactive way, offering insights into potential applications in automation, entertainment, and human-computer interaction.

Components

COMPONENT LIST	QUANTITY
Arduino Uno	1
Usb cable	1
Light sensor	1
Servo arm (small)	1
Servo	1
Base sheild	1

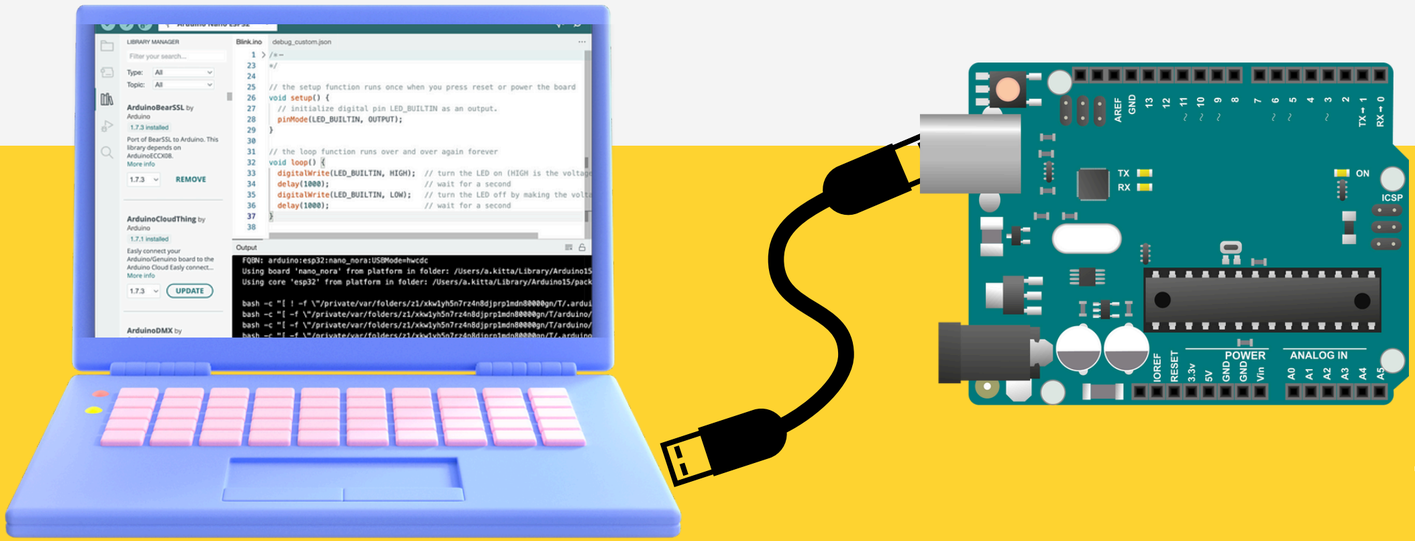


Arduino UNO

- Think of the Arduino Uno as a smart brain for your projects, helping you easily connect and control different electronic parts.

- It's like being part of a big, friendly club where everyone shares ideas and helps each other - that's the open-source magic of Arduino Uno.

- With its superpowers to understand and communicate with various gadgets, the Arduino Uno turns your ideas into reality, whether it's lighting up LEDs or making robots move.

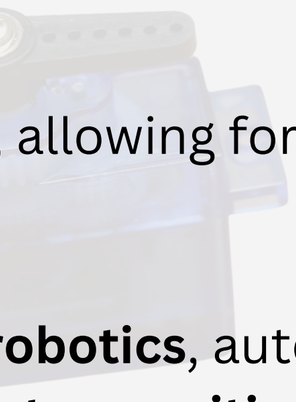


USB cable

- **Description:** A standard USB cable used to connect the Arduino Uno to a computer for programming and power.
- **Function:** Provides power to the Arduino Uno and allows data transfer between the board and the computer for programming.
- **Usage**: Connects the Arduino Uno board to a computer's USB port for programming and power supply.

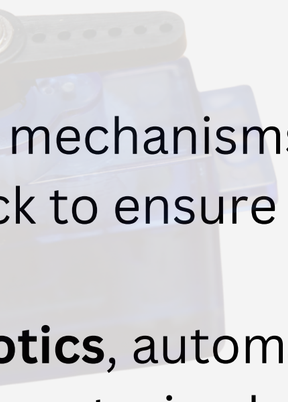
Servo arm

- The servo arm can **rotate to different angles** within a specified range, controlled by signals from the Arduino Uno.
- It offers **precise positioning control**, allowing for accurate alignment or manipulation of objects in projects.
- The servo arm finds applications in **robotics**, automation, and various DIY projects, where controlled **movement or positioning is required,** such as robotic arms, camera gimbals, or remote-controlled vehicles.



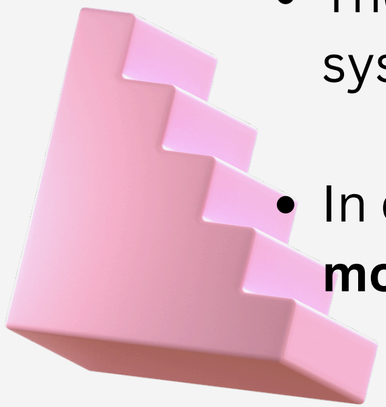
Servo

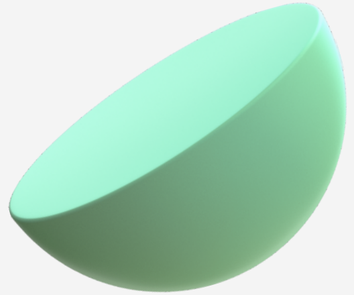
- **Servo motors** rotate to specific angles, providing precise control over rotational movement within a defined range.
- They often include **built-in** feedback mechanisms such as **potentiometers,** enabling accurate positional feedback to ensure precise control.
- Servo motors are widely used in **robotics**, automation, and RC (remote control) applications, including **robotic arms**, motorized vehicles, and aircraft control surfaces.



Light sensor

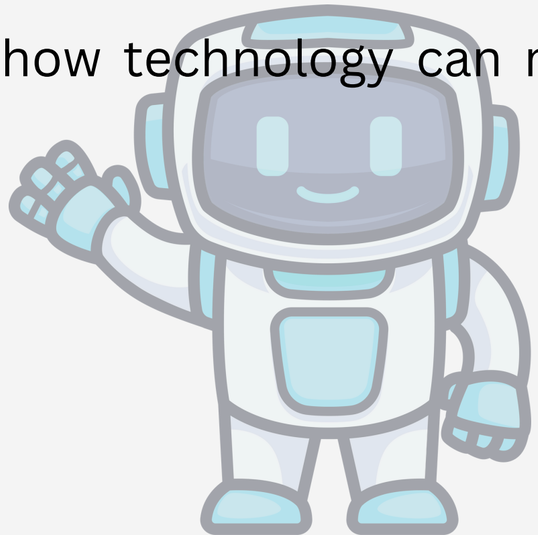
- The **Light sensor** can detect changes in ambient light intensity, converting these changes into electrical signals that the Arduino Uno can interpret.
- Allows for **continuous measurement of light levels** rather than discrete on/off detection.
- The light sensor can be used in various projects such as automatic lighting systems, **ambient light detection** for display brightness adjustment.
- In our “Hello Thimbot” project we will use the light sensor to **detect any hand movement** soo that our robot can wave back.





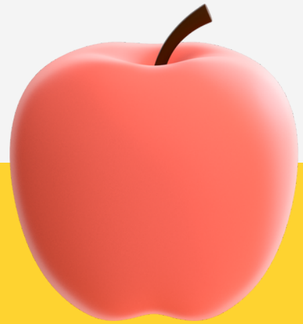
How it Works?

- In this project, we will first employ an **servo motor** accompanied by a servo arm and program them to **resemble hand movements.**
- We will then use a light sensor to **detect changes in light** intensity.
- When light is detected, the Arduino Uno **interprets the signal and activates a servo** motor, causing the robot's arm to wave in response.
- This simple yet engaging mechanism showcases how technology can mimic life like behavior through programming.

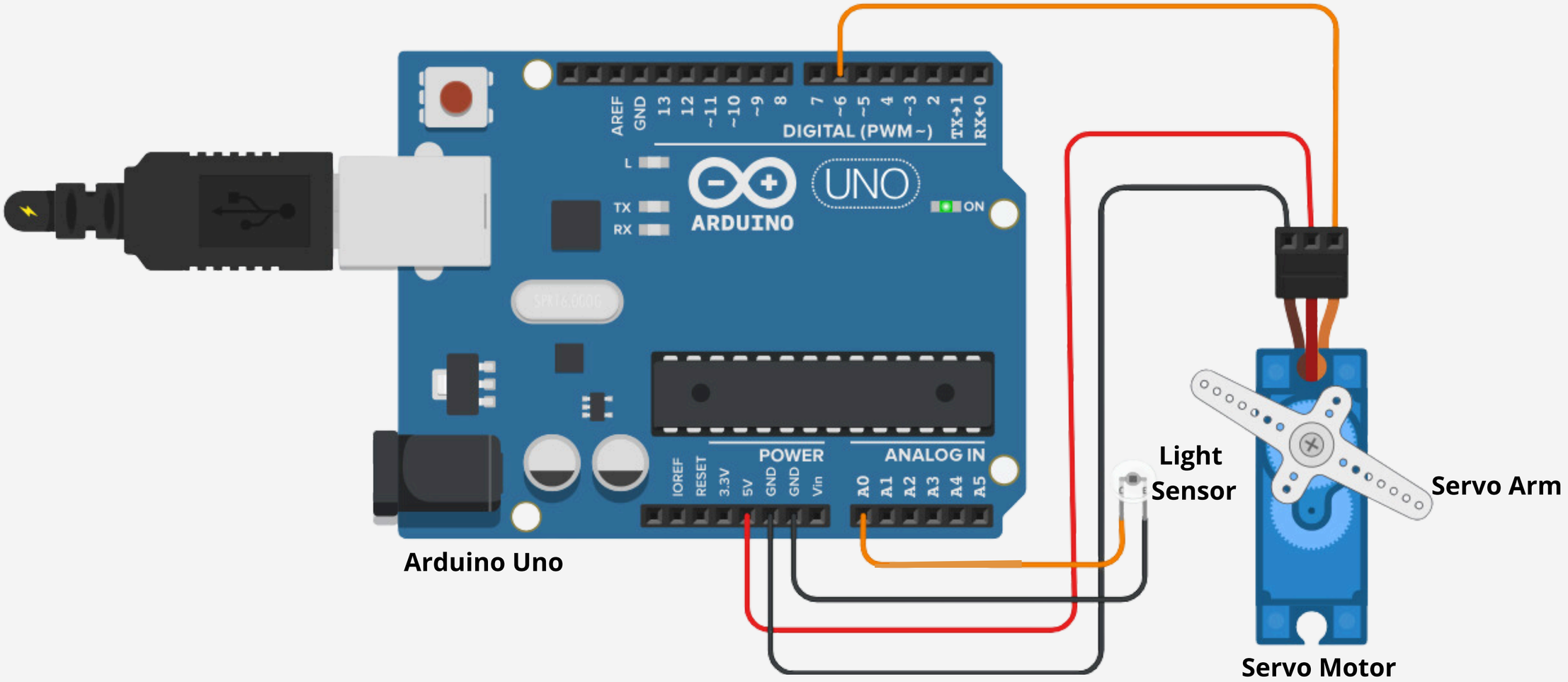


Student Activity Link

Activity reference video	**https://shorturl.at/cwFGV**

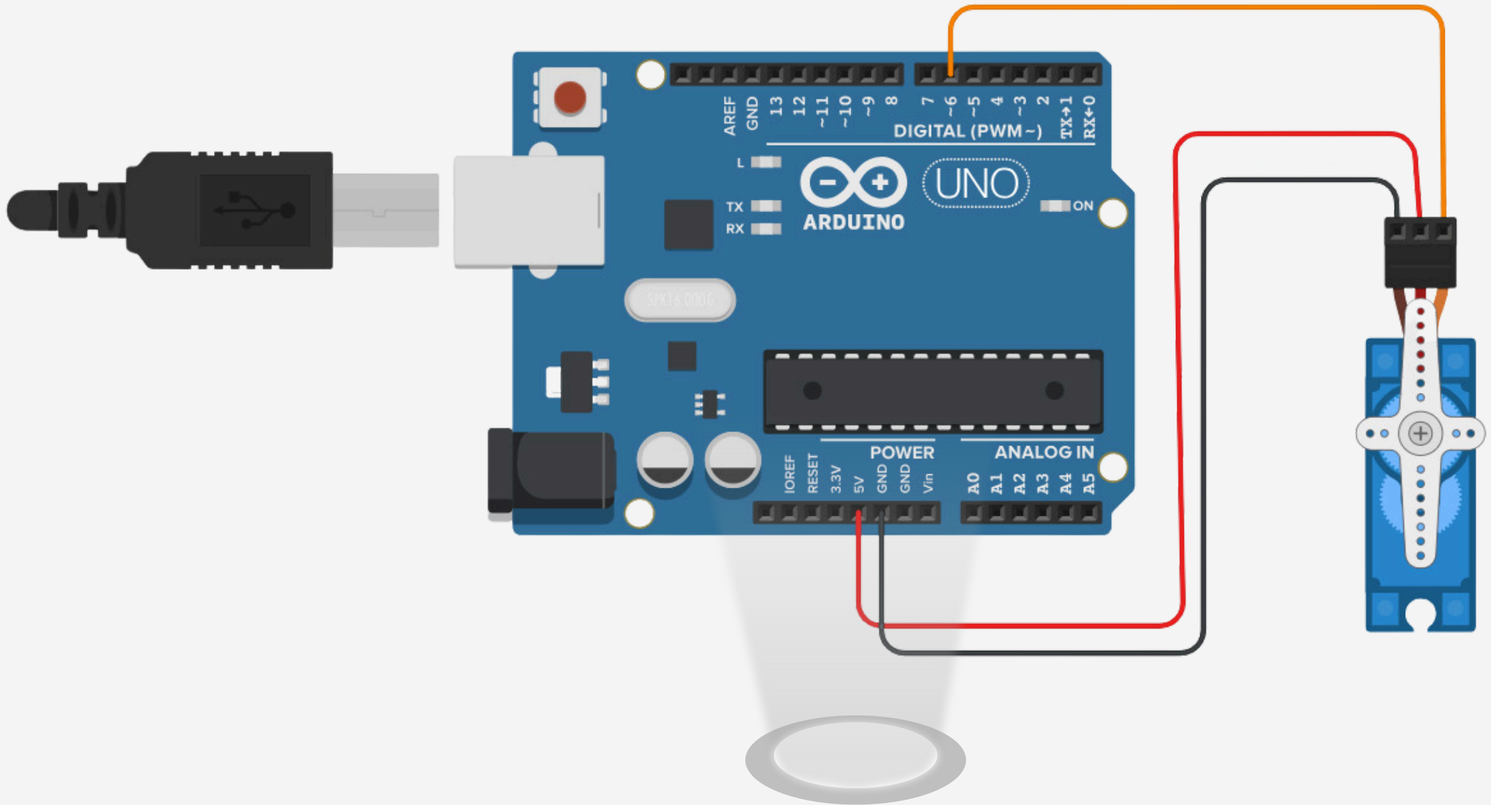


Circuit Connection



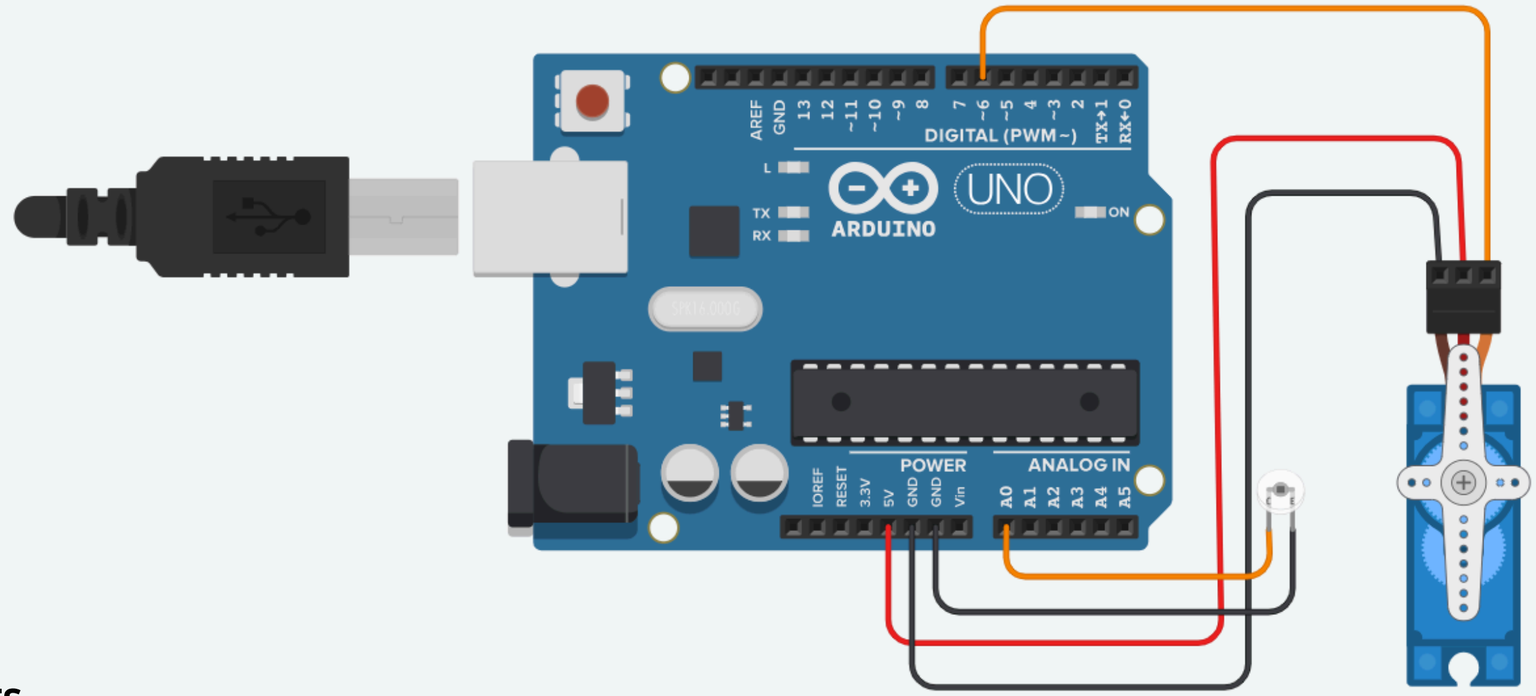
Step 1

- Connect the Signal pin of the Servo to the digital pin 6 of the Arduino Uno.
- Connect the Power and GND pin to those of the Arduino Uno



Step 2

- Connect the one terminal of the Light sensor to the Analog pin A0 of the Arduino and the other Terminal to the GND.



Step 3

- Let’s write the Code for our System.
- First of all, we need to include the Servo library.
- Then we define the servo and sensor sockets at pins A0 an 6 respectively.
- We then create an object of servo and initialize variables i.e. pos, val, trigger and waveSpeed in this case.

```
#define servoSocket 6
#define lightSensorSocket A0

Servo robotArm;

int pos;
int val;
int trigger = 100; //<- Change to YOUR measured value
int waveSpeed = 5;
```

Step 4

- The variables leftWave and rightWave define the positions of the servo motor (robot arm).
- LeftWave is set to 90, representing the position where the arm is waving to the left, and rightWave is set to 0, representing the position where the arm is waving to the right.
- In the setup() function, the code attaches the servo motor robotArm to the pin specified by servoSocket.

```
int leftWave = 90;
int rightWave = 0;

void setup() {
  robotArm.attach(servoSocket);
}
```

Step 5

- The loop keeps checking the light sensor to see if something is nearby. .
- When the sensor detects movement (`val` is less than the `trigger` value), the loop triggers the robot arm to start waving.
- First, the arm waves to the left, then back to the right, and repeats this motion twice.
- The `delay(waveSpeed)` controls how fast the arm moves during the waving motion.

```
void loop() {
  val = analogRead(lightSensorSocket);
  if (val < trigger) {
    for (pos = rightWave; pos <= leftWave; pos += 1) {
      robotArm.write(pos);
      delay(waveSpeed);
    }
    for (pos = leftWave; pos >= rightWave; pos -= 1) {
      robotArm.write(pos);
      delay(waveSpeed);
    }
    for (pos = rightWave; pos <= leftWave; pos += 1) {
      robotArm.write(pos);
      delay(waveSpeed);
    }
  }
}
```

Final OutPut

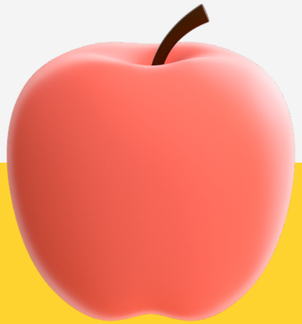
Link	**https://shorturl.at/cwFGV**

Congratulations

Hurray!

We have completed the

Hello Pingal bot Project



Learning outcomes

- Students explore how sensors (e.g., light sensors) and actuators (e.g., servo motors) work together to create responsive behavior in robots.
- They gain practical knowledge by working with servo motors and assembling the robot arm.
- Students can customize the project by adjusting parameters such as waving speed, angle, and trigger threshold, encouraging creativity and experimentation.

Notes

- Verify all connections with respected teachers.
- Safely use the electronics components.
- Check the Project is working properly.

Thank you!

Do you have any questions for me?