

## Experiment – 10

**Aim: Create an Arduino circuit to interface a servo motor with a 0-180 degree derelict.**

### Components

- Arduino Uno R3
- Small Breadboard
- Jumper cable
- Micro Servo

### Theory

#### 1. Arduino Uno R3

The Arduino Uno R3 is the main microcontroller in this circuit. It acts as the brain of the setup, executing the programmed code to control the LED. It provides both power and signal to the circuit, making it an essential component for automation and control projects.

#### 2. Small Breadboard

The small breadboard is used for easy and temporary circuit connections without soldering. It allows the LED and other components to be connected securely while maintaining flexibility for modifications.

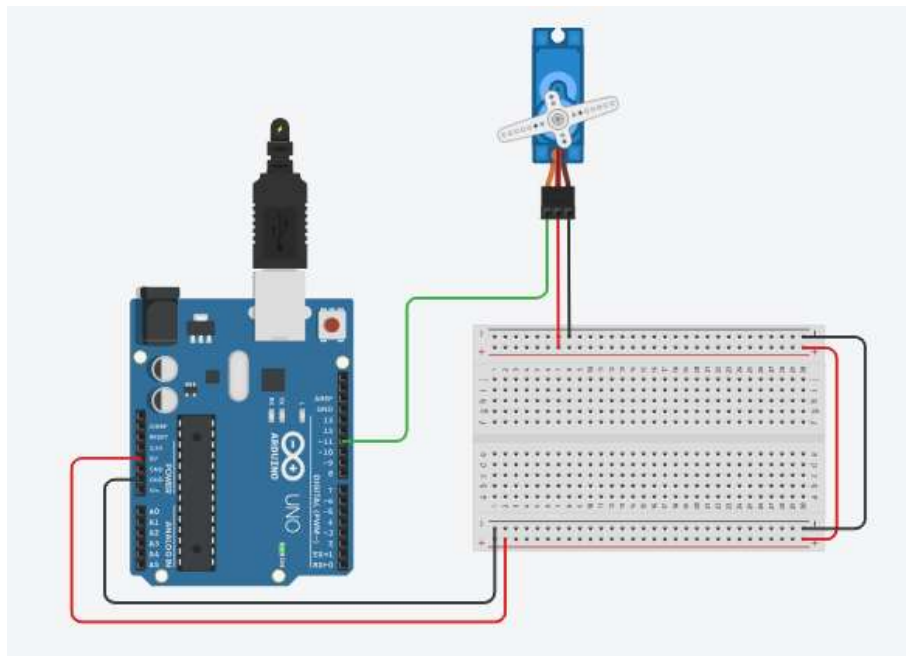
#### 3. Jumper Cable:

Used to make electrical connections between the Arduino, breadboard, and components.

#### 4. Micro Servo:

A micro servo is a compact motor that provides precise control of angular movement, making it ideal for applications in robotics and automation, while an oscilloscope visualizes electrical signals over time, enabling analysis of circuit behavior and performance...

### Circuit Design



### Procedure to Design a Servo Motor Control Circuit using Arduino Uno R3

#### Step 1: Add Components to the Workspace

1. Search for Arduino Uno R3 and add it to the workspace.
2. Search for a small breadboard and place it next to the Arduino.
3. Search for a micro servo motor and position it on the breadboard.
4. Use jumper cables to connect the components properly.

#### Step 2: Build the Circuit Connections

1. Servo Motor Connections:
2. Connect the control wire (usually orange or yellow) of the servo motor to a digital pin (e.g., pin 11) on the Arduino.
3. Connect the power wire (usually red) of the servo motor to the 5V pin on the Arduino.
4. Connect the ground wire (usually brown or black) of the servo motor to GND on the Arduino.

#### Step 3: Code for Servo Motor Control

```
#include <Servo.h>
int motor_pin = 11;
Servo s1;
void setup() {
    s1.attach(motor_pin);
}
void loop(){
    for(int i = 0; i <= 180; i++) {
        s1.write(i);
        delay(10);
    }
}
```

### Conclusion

In this project, we successfully implemented a circuit that utilizes the Arduino Uno R3 to control a micro servo motor across a range of 0 to 180 degrees. By sending position commands to the servo, we demonstrated how to create smooth movements, showcasing the capabilities of the Arduino in robotics and automation projects. The use of a small breadboard and jumper cables facilitated easy connections, allowing for flexibility in circuit design.