

Servo Motor Control with IR Remote and Arduino

1. Introduction

This project demonstrates controlling a servo motor using an IR remote and Arduino. By pressing different buttons on the remote, the servo motor rotates to specific angles.

2. Key Components

- Arduino Uno (or compatible board)
- Servo Motor (e.g., SG90)
- IR Remote Control
- IR Receiver Module (e.g., TSOP1738)
- Jumper Wires
- Breadboard

3. Working Principle

- The IR remote sends an infrared signal when a button is pressed.
- The IR receiver module detects this signal and sends it to the Arduino.
- Arduino decodes the received IR signal to identify which button was pressed.
- Based on the decoded signal, the Arduino controls the servo motor to move to a predefined angle (0°, 90°, or 180°).
- After executing the command, the Arduino prepares to receive the next IR signal.

4. Circuit Overview

- Connect the IR receiver's "OUT" pin to Arduino digital pin 5.
- Connect the servo motor's control (signal) wire to Arduino digital pin 3.
- Connect both the IR receiver and the servo motor's power (VCC) to 5V and ground (GND) on the Arduino.
- Use a breadboard and jumper wires to establish connections properly.

5. Code

/*Code written by-

Fuad Hasan

BME, KUET */

/*Servo motor with IR remote and Arduino*/

#include <Servo.h>

#include <IRremote.h>

#define IR_RECEIVE_PIN 5

#define SERVO_PIN 3

Servo myServo;

IRrecv irrecv(IR_RECEIVE_PIN);

decode_results results;

void setup() {

 Serial.begin(9600);

 myServo.attach(SERVO_PIN);

 irrecv.enableIRIn(); // Start the IR receiver

}

void loop() {

 if (irrecv.decode(&results)) {

 Serial.println(results.value, HEX); // Print received IR code

 // Example: Change servo angle based on IR remote button

```

if (results.value == 0xFFA25D) { // Button 1
    myServo.write(0); // Move to 0 degrees
}
else if (results.value == 0xFF629D) { // Button 2
    myServo.write(90); // Move to 90 degrees
}
else if (results.value == 0xFFE21D) { // Button 3
    myServo.write(180); // Move to 180 degrees
}

irrecv.resume(); // Receive next value
}
}

```

6. Code Explanation

- **Library Inclusions:**

- Servo.h for servo motor control.
- IRremote.h for decoding IR signals.

- **Pin Definitions:**

- IR_RECEIVE_PIN (Pin 5) is used for IR receiver input.
- SERVO_PIN (Pin 3) is used for servo motor signal output.

- **Object Initialization:**

- Servo myServo; creates a servo object.
- IRrecv irrecv(IR_RECEIVE_PIN); initializes the IR receiver.
- decode_results results; holds the decoded IR data.

- **Setup Function:**

- Begins Serial communication at 9600 baud.

- Attaches servo motor to pin 3.
- Enables IR receiver to start listening for signals.
- **Loop Function:**
 - Checks if an IR signal has been received.
 - Prints the received IR code to the Serial Monitor.
 - Compares the received code with predefined values:
 - 0xFFA25D: Moves servo to 0°.
 - 0xFF629D: Moves servo to 90°.
 - 0xFFE21D: Moves servo to 180°.
 - After action, resumes IR receiver for next input.

7. Conclusion

This project effectively demonstrates how to integrate an IR remote control with a servo motor using Arduino. By understanding IR signal decoding and servo positioning, users can build more complex remote-controlled systems for robotics, automation, and IoT applications.

