

ESP32 Servo Motor Control using ESP32Servo Library

1. Objective

The goal of this project is to control a servo motor using an ESP32 and make it sweep smoothly from 0° to 180° and back, repeating indefinitely.

2. Required Components

- ESP32 Dev Board
- Servo Motor (e.g., SG90)
- External Power Source (optional)
- Jumper Wires
- Breadboard

3. Pin Configuration

- Servo Signal -> GPIO 13 (PWM-capable)
- Servo VCC -> 5V (external recommended)
- Servo GND -> GND (shared with ESP32)

Important: Use external power for the servo for stable operation.

4. Wiring

Connect servo signal wire to GPIO13 of ESP32.

Connect servo power to external 5V and ground to both servo and ESP32.

5. Full Code

```
#include <Arduino.h>
```

ESP32 Servo Motor Control using ESP32Servo Library

```
#include <ESP32Servo.h>

#define SERVO_PIN 13

Servo myServo;

void setup() {

  Serial.begin(115200);

  myServo.setPeriodHertz(50);

  myServo.attach(SERVO_PIN, 500, 2400);

  Serial.println("ESP32 Servo Ready");

}

void loop() {

  for (int pos = 0; pos <= 180; pos++) {

    myServo.write(pos);

    delay(15);

  }

  delay(500);

  for (int pos = 180; pos >= 0; pos--) {

    myServo.write(pos);

    delay(15);

  }

  delay(500);

}
```

ESP32 Servo Motor Control using ESP32Servo Library

6. Line-by-Line Explanation

- Includes Arduino and ESP32Servo libraries.
- Defines GPIO pin 13 for servo.
- Sets up PWM with 50Hz.
- Attaches the servo with pulse width range.
- Loop moves the servo from 0° to 180°, then back to 0° with delays for smooth motion.

7. Technical Concepts

- PWM (Pulse Width Modulation): Controls servo angle with pulse widths (500µs for 0°, 2400µs for 180°).
- Servo Frequency: 50Hz (standard for hobby servos).
- ESP32Servo: Uses ESP32 hardware timers for accurate control.

8. Summary

This project demonstrates basic servo control using an ESP32.

It teaches PWM, motion control, and hardware interfacing essential for robotics and embedded systems.