

Joystick Controlled Servo Motor Using Arduino

Project Overview

This project allows users to control a **servo motor** using a **joystick module**. By moving the joystick, the servo motor rotates accordingly, making it ideal for **robotic arms, camera gimbals, and pan-tilt mechanisms**.

Objectives

- ✓ **Control servo motor movement** using a joystick.
- ✓ **Convert joystick input** into servo angles.
- ✓ **Smooth and precise rotation** for various applications.
- ✓ **Use an LCD display (optional)** to show the servo angle.

Components Required

1. **Arduino Uno** – Main microcontroller.
2. **Joystick Module (XY Axis)** – Controls servo movement.
3. **Servo Motor (SG90/MG995)** – Moves based on joystick input.
4. **Breadboard & Jumper Wires** – For connections.
5. **5V Power Supply** – Powers the system.

How the System Works

1. The **joystick module** has two potentiometers (X and Y axes) that provide voltage values.
2. The **Arduino reads** these values and maps them to **servo angles (0°–180°)**.
3. The **servo rotates** according to the joystick movement.
4. (Optional) An **LCD display** shows the current servo position.

Circuit Diagram

Joystick Module to Arduino

Joystick Pin	Arduino Pin
--------------	-------------

Joystick Pin Arduino Pin

GND GND
VCC 5V
VRX (X-axis) A0
VRY (Y-axis) A1
SW (Button) D2 (Optional)

Servo Motor to Arduino

Servo Pin Arduino Pin

VCC 5V
GND GND
Signal D9

Arduino Code for Joystick-Controlled Servo

```
#include <Servo.h>

// Define joystick pins
#define VRX A0    // Joystick X-axis
#define VRY A1    // Joystick Y-axis

// Define servo
Servo servoX;    // Servo for X-axis

void setup() {
  Serial.begin(9600);
  servoX.attach(9);    // Connect servo signal wire to pin 9
}

void loop() {
  // Read joystick values (0-1023)
  int xValue = analogRead(VRX);

  // Map joystick input (0-1023) to servo angles (0-180)
  int angleX = map(xValue, 0, 1023, 0, 180);

  // Move the servo
  servoX.write(angleX);

  // Print values to Serial Monitor
  Serial.print("X-Axis: ");
  Serial.print(xValue);
  Serial.print(" | Servo Angle: ");
  Serial.println(angleX);

  delay(10);    // Small delay for smooth movement
}
```

How to Use the Joystick Controlled Servo

1. **Connect the components** as per the circuit diagram.
2. **Upload the code** to the Arduino using the Arduino IDE.
3. **Move the joystick** left or right to control the servo.
4. **Check the Serial Monitor** for joystick readings and servo angles.
5. (Optional) Add a **second servo** for Y-axis control.

Features & Benefits

- ✓ **Real-time servo control** using a joystick.
- ✓ **Precise movement** with mapped angles.
- ✓ **Simple and effective** for robotic applications.
- ✓ **Expandable** – Add more servos for dual-axis control.

Future Enhancements

Add a second servo to control another axis (pan-tilt).

Wireless joystick control using Bluetooth or RF.

LCD display to show live joystick positions.

Multi-servo control for robotic arms.

