

# How to use a Servo Motor

## What is a Servo Motor?

A servo motor is a rotary actuator that allows for precise control of angular position, velocity, and acceleration. It is commonly used in robotics, CNC machinery, conveyor belts, and various automation systems. A servo motor works by receiving a control signal that represents a desired output position, and it uses an internal feedback system to adjust the motor's movement accordingly. We often use RC (Radio-Controlled) Servo Motors in physical computing.



## Types of Common RC Servo Motors

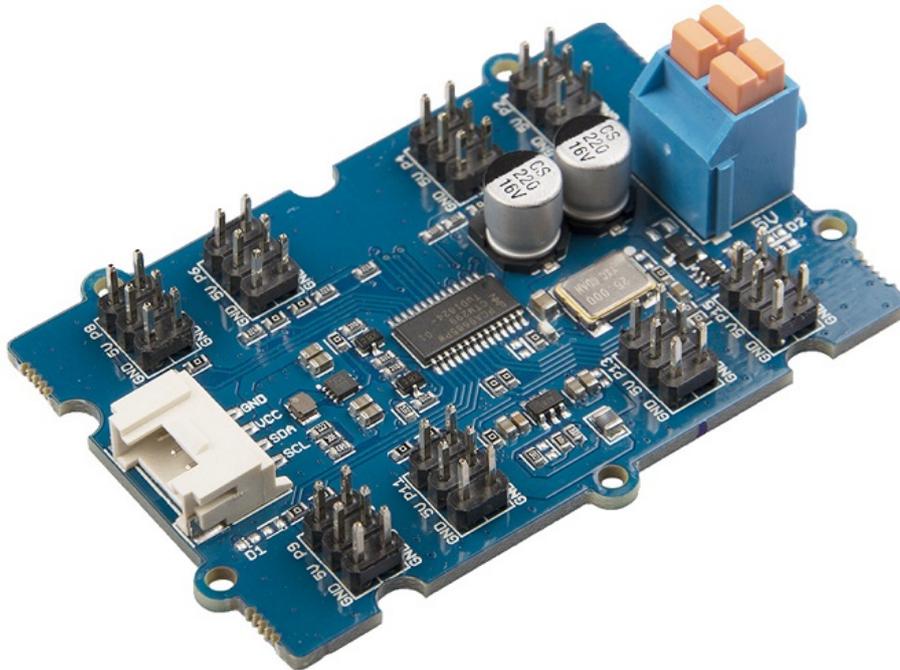
Servo Type	Rotation Range	Control Type	Common Use
Regular (Standard) Servo	0°-180° (sometimes 270°)	Angular position (PWM)	Arms, steering, gimbals
Continuous Rotation Servo	360° (infinite)	Speed & direction (PWM)	Wheels, drive systems
Mini / Micro Servo	Depends on type (limited or continuous)	Same as regular or continuous	Small robots, drones, model planes

## Using a PWM Driver

You may come across many online tutorials that show how to control servo motors directly without a PWM driver. While this method is possible, it's not the most reliable—especially when working

with multiple servos. Sooner or later, you're likely to run into issues, often related to power management and signal stability.

For this reason, in this tutorial, we'll be using the Seed Studio 16-channel PWM driver for more stable and scalable control. You can also use other similar PWM driver boards depending on your needs.



## Wiring

### 1. 5V Power Supply on the Left:

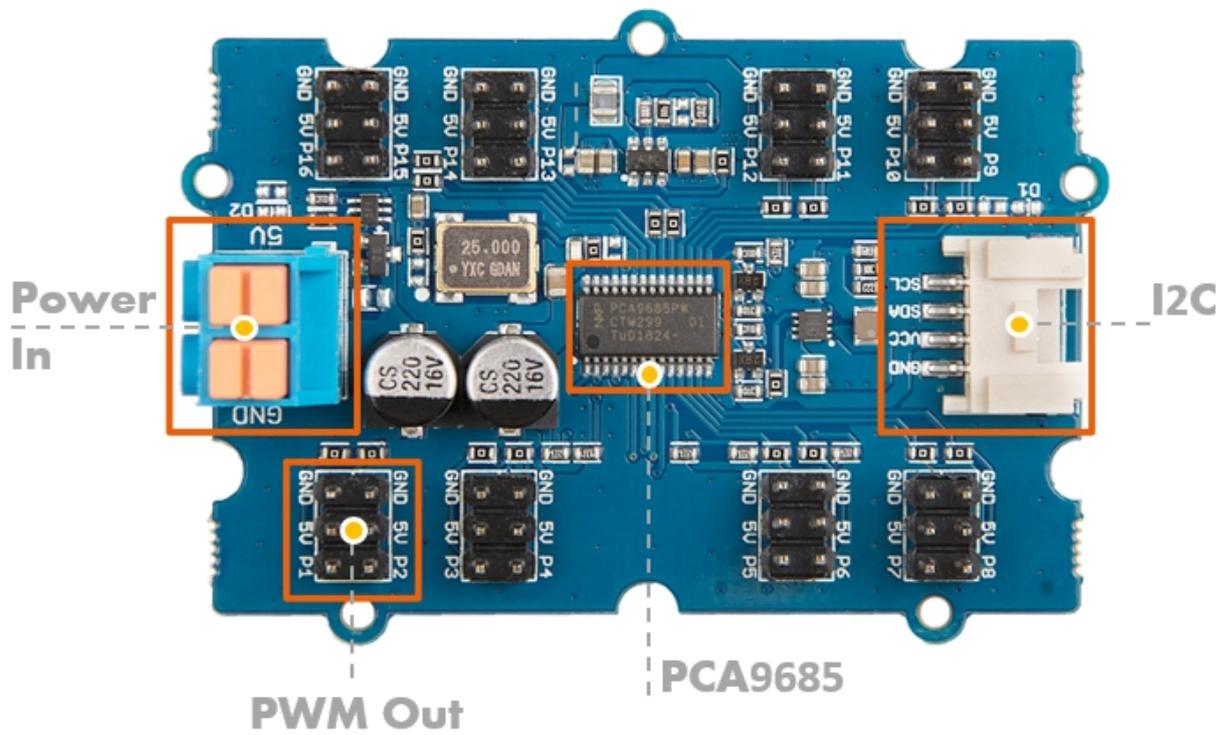
- to 5V
- to GND

### 2. i2C to Arduino on the Right:

- GND to GND
- 5V to 5V
- SCL to SCL
- SDA to SDA

### 3. Servo to PWM Out:

- Plug in the servo, where the black wire goes to the GND



## Library

**Grove-16-Channel PWM Driver-PCA9685** library will be used.

We have a tutorial on [how to install a library](#) here.

## Getting started - Regular Servo

This code is getting the eight servo motors to rotate to 0 degrees and then 90 degrees, one by one.

```
#include "PCA9685.h"
#include <Wire.h>

ServoDriver servo;

void setup()
{
  // join I2C bus (I2Cdev library doesn't do this automatically)
  Wire.begin();
  Serial.begin(9600);
  servo.init(0x7f);
}
```

```
void loop()
{
  // Drive 8 servos in turns
  for (int i = 1; i < 9; i++)
  {
    servo.setAngle(i, 0);
    delay(1000);
    servo.setAngle(i, 90);
    delay(1000);
  }
}
```

## Getting started - Continuous Servo

Since a continuous servo is controlled by a pulse, the 'degree' values work differently. Values like 0, 90, and 180 don't represent angles; instead, they typically control rotation direction and speed — for example, 0 means full speed clockwise, 90 means stop, and 180 means full speed counterclockwise.

The below code is getting the servo motor 1 rotates clockwise for 1 second and stop for 1 second.

```
#include "PCA9685.h"
#include <Wire.h>

ServoDriver servo;

void setup()
{
  // join I2C bus (I2Cdev library doesn't do this automatically)
  Wire.begin();
  Serial.begin(9600);
  servo.init(0x7f);

  //Most continuous servo motors' pulse width is 900-2100
  servo.setServoPulseRange(900,2100,180);
}

void loop()
{
  // Drive 8 servos in turns
```

```
servo.setAngle(1, 0);  
delay(1000);  
servo.setAngle(1, 90);  
delay(1000);  
  
}
```

---

Revision #2

Created 24 June 2025 13:44:06 by Joanne Leung

Updated 24 June 2025 14:17:18 by Joanne Leung