

فهرست:

توضیحات

نحوه عملکرد Servo motor

اجزا داخلی Servo motor

انواع سروو موتورها

اطلاعات دیتاشیت Servo Motor Micro SG90

اتصال Servo motor به Arduino

اولین کد برای Servo

Servo Driver – PCA9685

پایه های درایور

اتصال درایور به آردوینو

Driver's Power

Driver's output port

Adding Capacitor

Code for One servo by PCA9685

Chaining Drivers

Addressing the boards

Code for Two PCA9685s

منابع

### توضیحات

سروو موتورها، موتورهای الکتریکی هستند که امکان کنترل دقیق حرکت دورانی و خطی را می دهند. این موتورها به دلیل اینرسی (لختی) پایینی که دارند می توانند به سرعت تغییر جهت داده و به موقعیت نهایی با زاویه خواسته شده برسند.



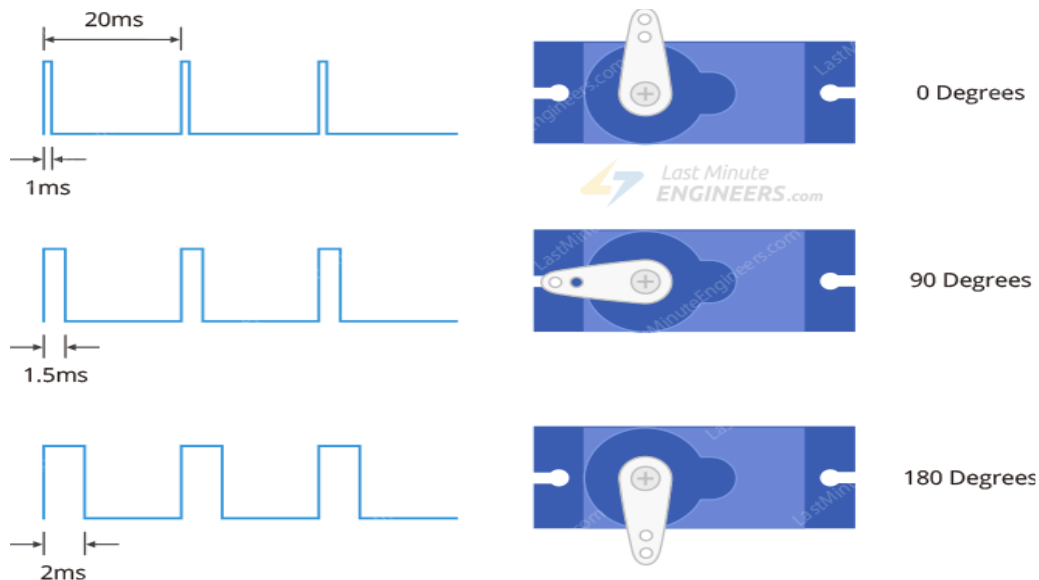
### نحوه عملکرد Servo motor

عملکرد یک سروو موتور به سیگنال های دریافتی که از طریق سیستم کنترلی تعبیه شده در آن به سروو اعمال میشود، بستگی دارد.

هر سروو یک شفت خروجی دارد که این محور به وسیله ی دریافت مجموعه ای از سیگنال های رمزی (Code Signals) طی یک حرکت زاویه ای متناسب با سیگنال ورودی قرار بگیرد.

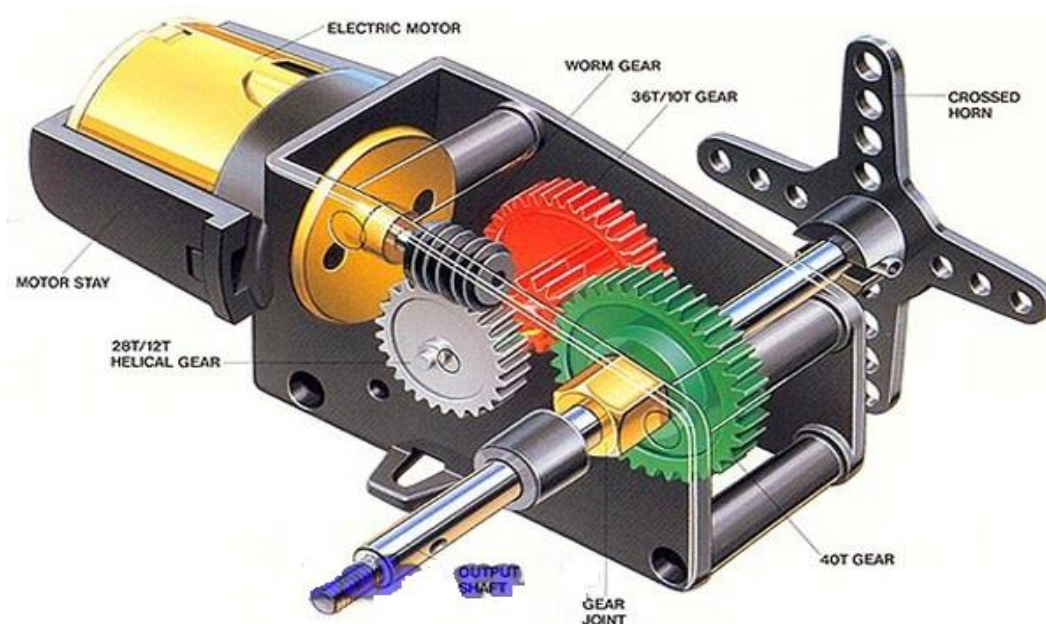
به عبارت دیگر، سروو زمانی که پالس ورودی فرستاده شده به آن فعال باشد (1 منطقی) میتواند در مکان زاویه ای منحصر به فرد آن قرار بگیرد.

موقعیت شفت موتور با توجه به پالس ورودی:

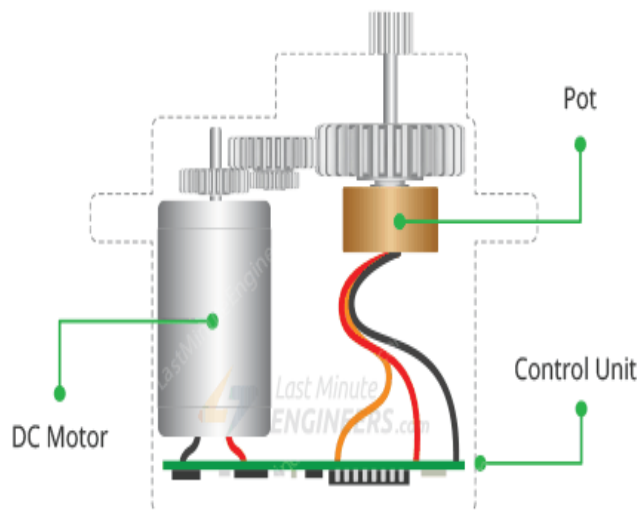


اجزا داخلی Servo motor:

سرو موتورها از موتورهای DC، گیربکس و بردهای الکترونیکی (میکروکنترلر) تشکیل شده.

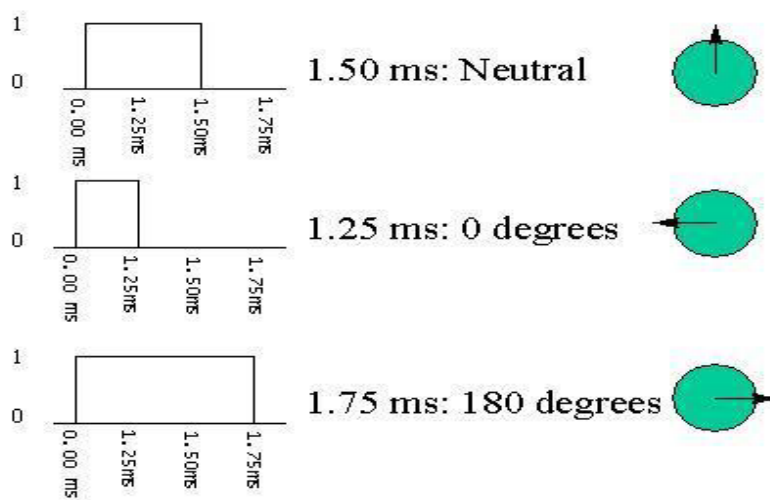


وظیفه اصلی برد الکترونیکی (پتانسیومتر) کنترل زاویه چرخش شفت سروو موتور هست. پتانسیومتر با دریافت سیگنال ورودی، خروجی چرخنده رو کنترل میکنه تا مطابق با زاویه خواسته شده حرکت کند.



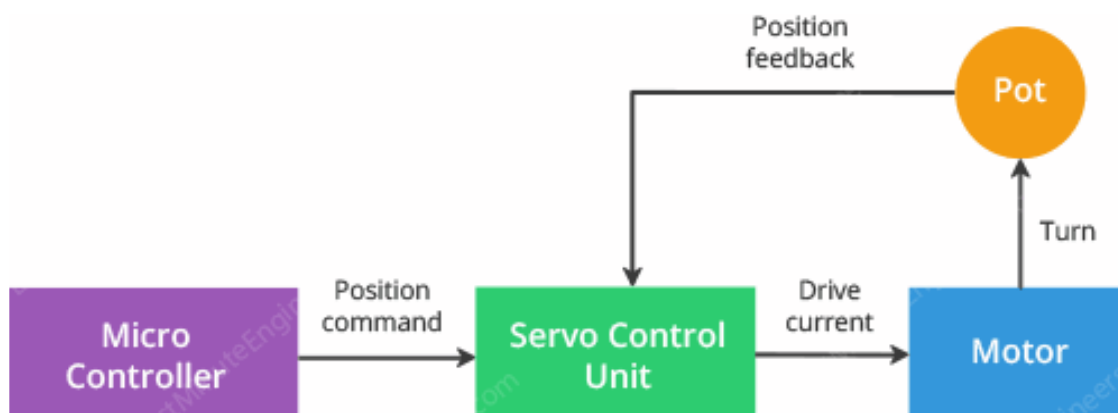
میکروکنترلر با دریافت پالس و بررسی عرض پالس ارسال شده، موقعیت جدید شفت رو تعیین میکنه. بنابراین عرض پالس ورودی، زاویه نهایی شفت رو تعیین میکنه.

برای مثال پالس  $0.5 \text{ ms}$ <sup>۱</sup> زاویه صفر و پالس  $25 \text{ ms}$  حداکثر زاویه شفت خواهد بود. (این مقادیر ممکن است برای سروو موتورها مختلف متفاوت باشد).



<sup>1</sup> ms : milli second

یک سیستم کنترل فیدبک<sup>۲</sup>، موقعیت دستگاه و همچنین قدرت و سرعت سروو موتورها را تغییر می دهند.



### انواع سروو موتورها:

سروو موتورها در دو نوع خطی و دورانی ساخته میشوند، که در نوع خطی، حرکت دورانی توسط گیربکس به حرکت خطی تبدیل میشود.

انکودرها جز جدانشدنی از سروو ها هستند چراکه این موتورها با فیربکی که از سیستم در حال کار میگیرند و به کمک انکودر حرکت خود را تصحیح میکنند. از این رو سیستم کنترل بسته و تصحیح حرکت از مزیت های سروو موتورها نسبت به استپ موتور<sup>۳</sup>هاست.

<sup>۲</sup>: سیستم کنترل فیدبک سیستمی است که به منظور کنترل، با مقایسه و بررسی توابع کمیت های کنترل شده و کمیت های مرجع، براساس اختلاف این دو کمیت رابطه ای ایجاد میکند.

<sup>۳</sup> Step motor

### اطلاعات دیتاشیت :Servo Motor Micro SG90

#### Specifications

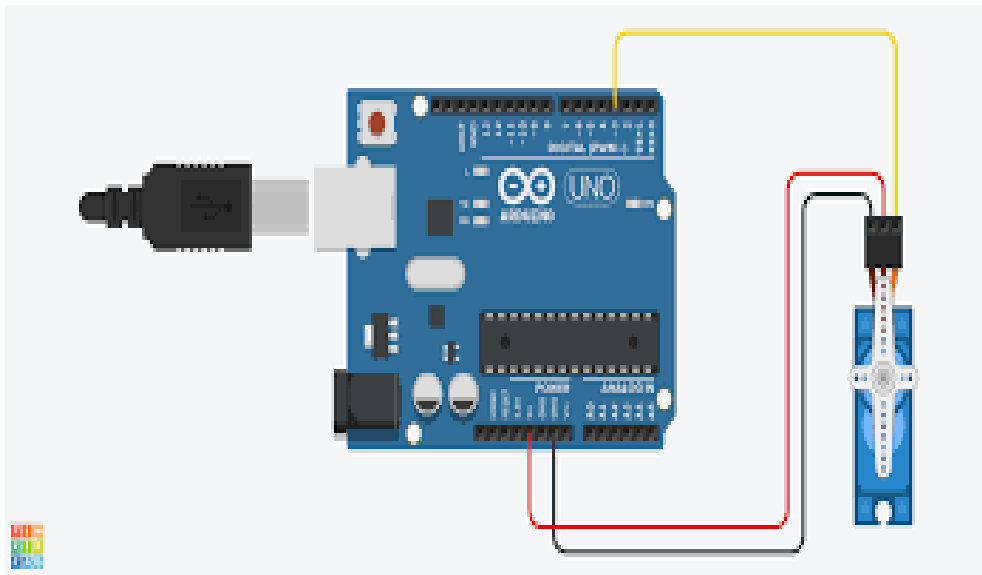
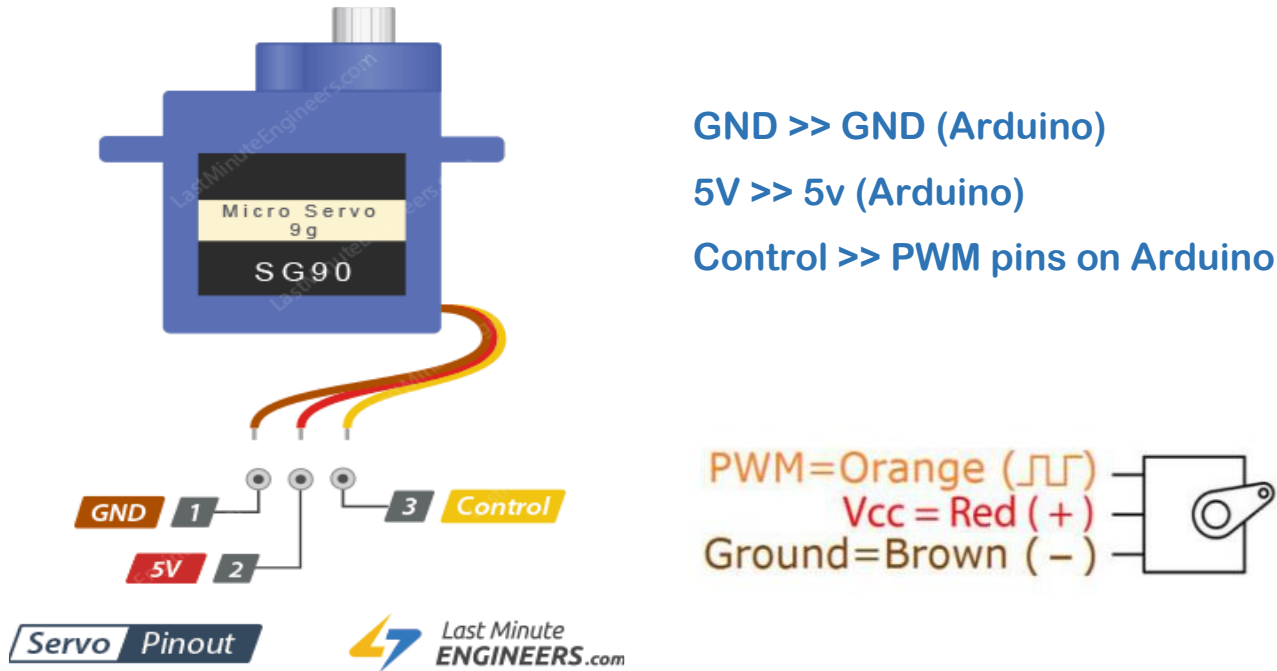
- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10  $\mu$ s
- Temperature range: 0 °C – 55 °C

#### TECHNICAL SPECIFICATIONS

Motor Model	Generic SG90 (China)
Drive Type	Analog
Degree Rotation	180° ( $\pm 15^\circ$ )
Operating Ratings	
Voltage	4.8-6VDC (5V Typical)
Current (idle)	10mA (typical)
Current (typical during movement)	100-250mA
Current (stall)	360mA (measured)
Stall Torque	1.7 kg·cm (measured)
Speed	0.12s / 60 degree (varies with VDC)

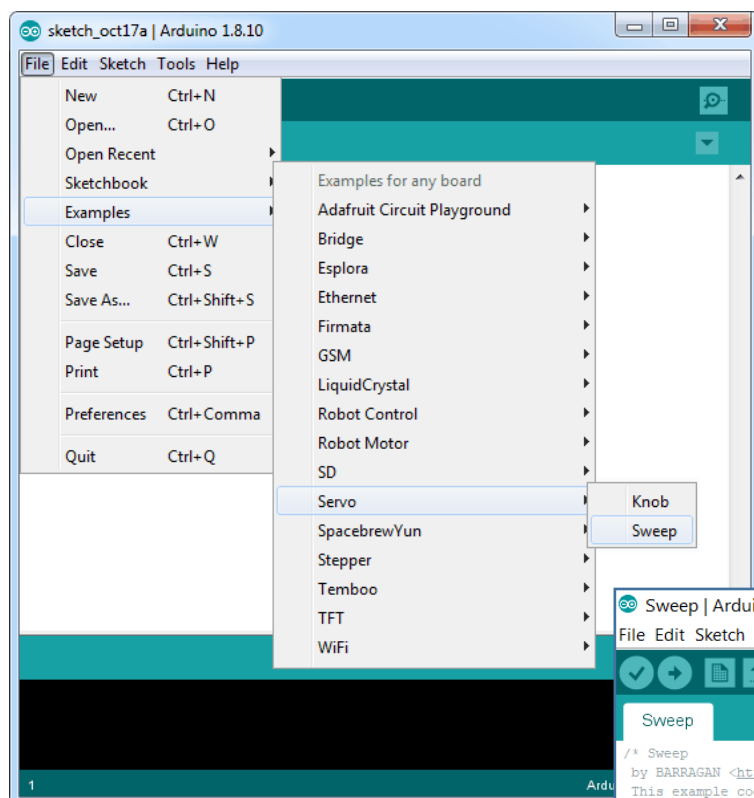
Stall torque is **the torque load that causes the output rotational speed of a device to become zero.**

اتصال Servo motor به Arduino:



### اولین کد برای Servo:

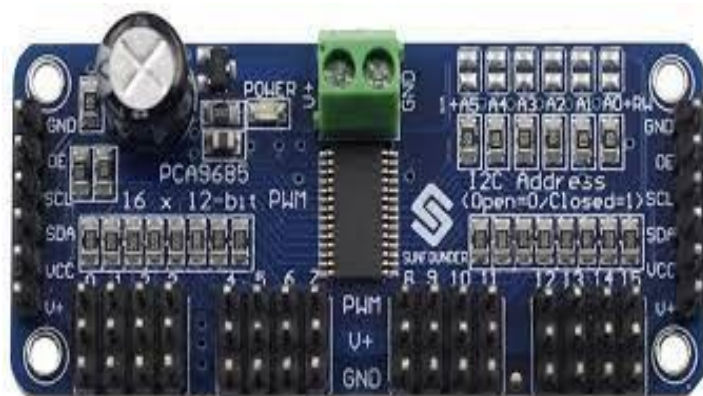
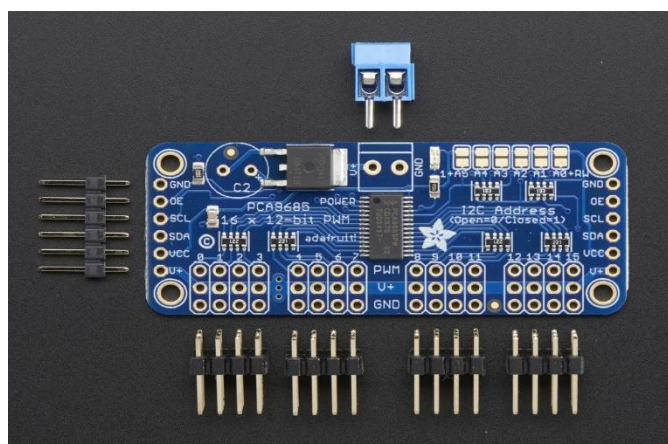
برای شروع کار با سرو موتور  
میتوانیم از مثالی که در IDE هست  
آرduino استفاده کنیم.





### Servo Driver – PCA9685

به کمک این درایور میتونیم تا 16 سروو موتور رو بطور همزمان و تنها از طریق 2 پایه ی SDA و SCL کنترل کنیم. علاوه بر اون میتونیم ماکسیمم 62 درایور رو بهم متصل و تا 992 سروو موتور رو بطور مستقل بکار ببریم.



### پایه های درایور

حتما چک کنین که پایه های درایور، مخصوصا V+ و VCC برد بهم لحیم (متصل) نشده باشن.

**GND >> GND (Arduino)**

**OE : output enable (بصورت پیشفرض Pulled down)**

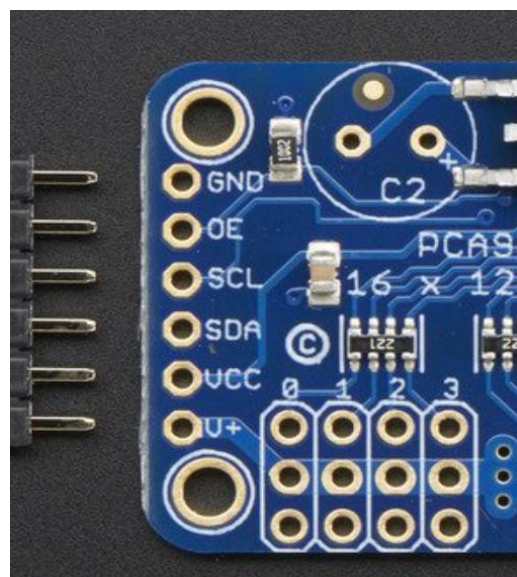
اگر پین OE، low باشه یعنی همه ی پین های سرووها enable هستن و اگر High باشه تمام پین ها disable میشن.

**SCL >> Analog 5 (I2C clock pin)**

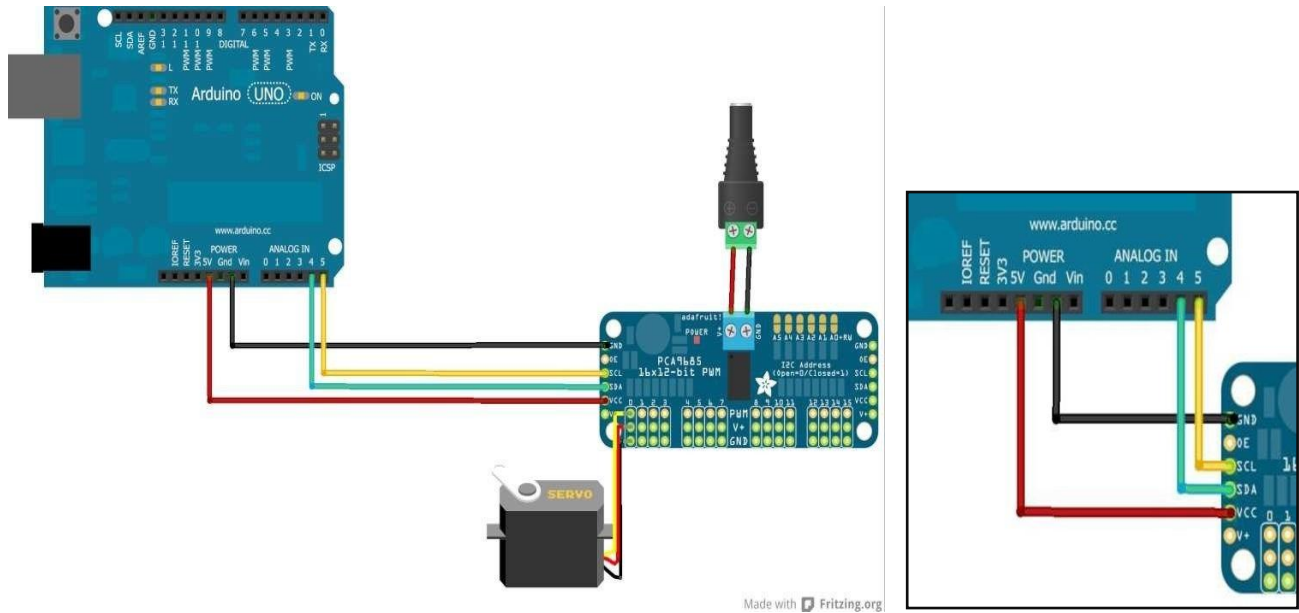
**SDA >> Analog 4 (I2C Data pin)**

**VCC >> 5v / 3.3v**

**V+ : optional power pin**



### اتصال درایور به آردوینو

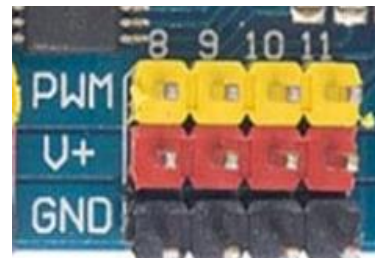
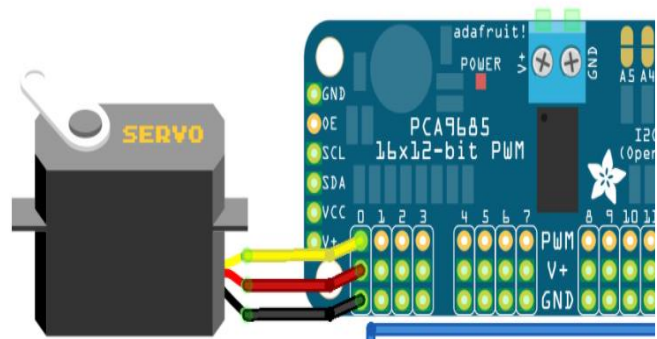


### اتصال سرووموتور به درایور PCA9685

سیم نارنجی سرووموتور << پایه PWM درایور

سیم قرمز سرووموتور << پایه V+ درایور

سیم مشکی سرووموتور << پایه GND درایور



### Driver's Power

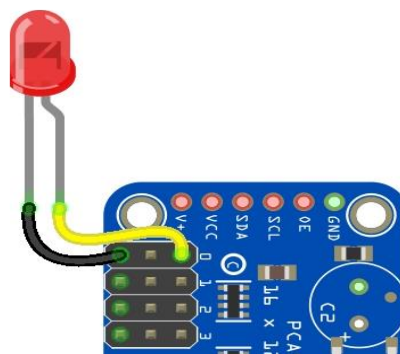
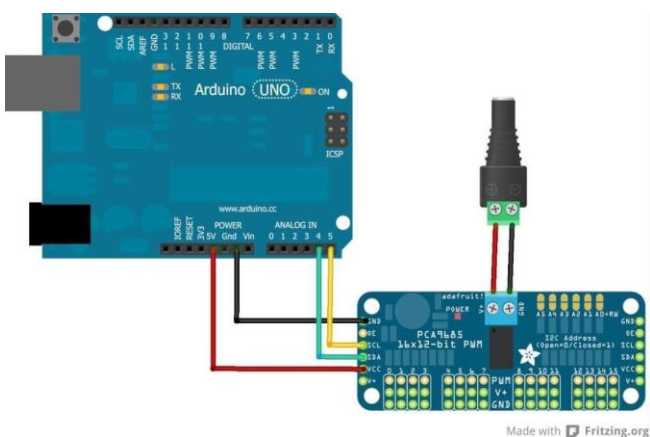
پایه VCC برای روشن شدن خور برده و اگر بخواهیم سروو موتور یا LED کنترل کنیم باید پایه ی V+ رو به 6v متصل کنیم. اما بهتره که تغذیه برد رو از طریق ترمینال بالای برد تامین کنیم.

حواستون باشه تغذیه برد رو به بیشتر از 6v وصل نکنین. بالاینکه که Adafruit میگه میتونیم منبع 12v وصل کنیم اما اگر پایه هارو اشتباه متصل کنیم به برد آسیب میرسه.

ترمینال تغذیه برد یه دیود داره که از مدار در برابر جریان معکوس محافظت میکنه اما پایه V+ این قابلیت رو نداره.

بیشتر سرووها طوری طراحی شدند که با 5 یا 6 ولت کار میکنند، اما زمانیکه چند سروو بطور همزمان کار کنند جریان زیادی لازمه.

حتی میکروسروو ها گاهی چند صد میلی آمپر جریان میکشند و همینطور سرووهای گشتاور بالا (High torque servo motor) تحت بار، ممکنه حتی بیشتر از 1A جریان لازم داشته باشند.



## Driver's output port

هر سرووموتور سه پایه داره که پایه ی PWM اونها همزمان و بطور کاملا مستقل کار میکنند اما تمام موتور ها باید فرکانس PWM یکسانی داشته باشند.

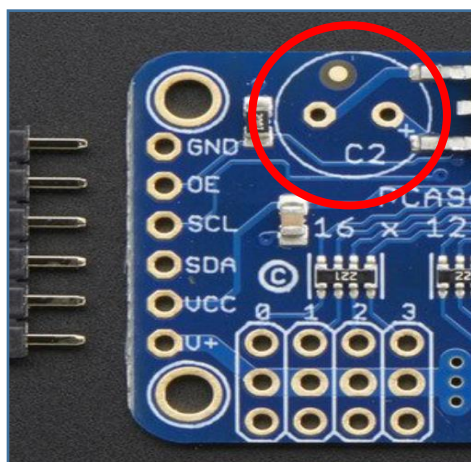
این درایور برای کنترل سرووموتور طراحی شده ولی میتونیم LED هم کنترل کنیم.  
**ماکسیم جریان هر پایه 25mA هست و کل جریانی که میتونیم از درایور بکشیم 8A هست.**

## Adding Capacitor

زمانی که سرووموتورهای زیادی رو متصل کرده و کنترل میکنیم، افت ولتاژ و جریان خواهیم داشت، بنابراین روی برد درایور PCA9685 که یک جا برای خازن هست، یک خازن با ظرفیت  $n * 100 \mu F$  ( n تعداد سروو های متصل شده به درایور) اضافه میکنیم.

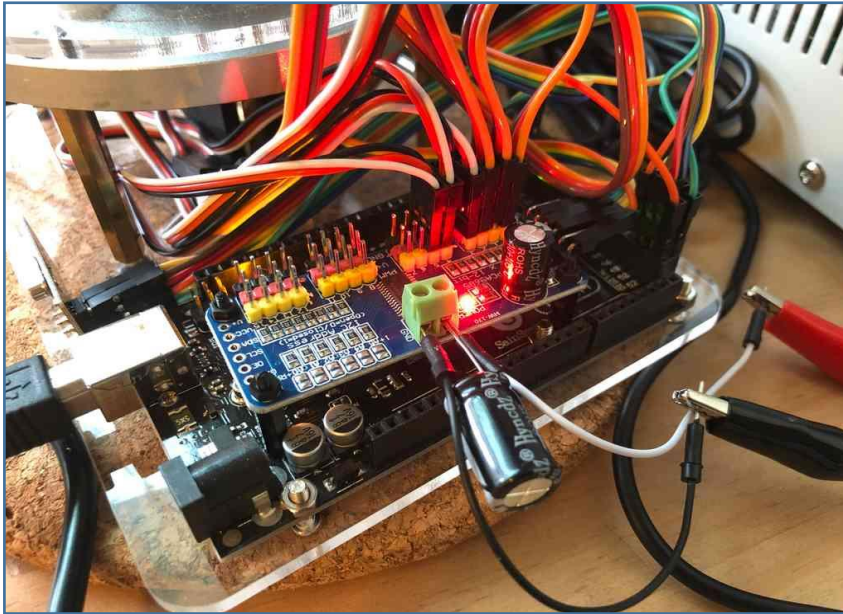
برای مثال، خازن  $470\mu F$  یا بیشتر برای کنترل 5 سرووموتور مناسبه.

اما از اونجایی که ظرفیت این خازن کاملاً به تعداد سرووها، گشتاور هر موتور و منبع تغذیه بستگی داره، نمیتونیم مقدار دقیقی برای آن پیشنهاد بدیم.



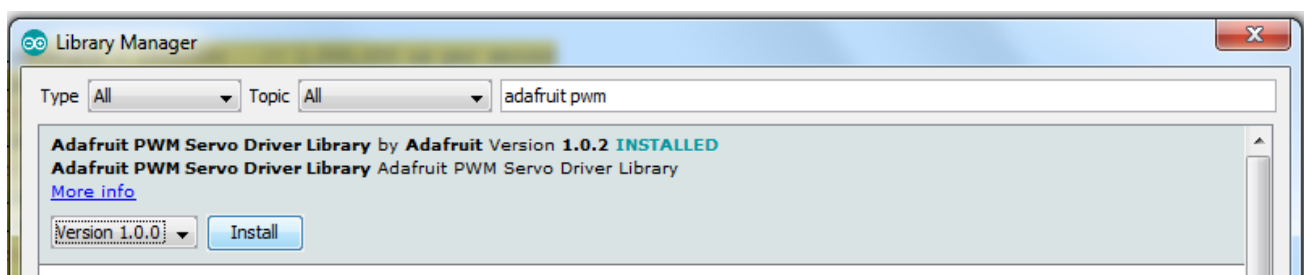


برای زمانی که سرعت سرووموتورها خیلی سریع تغییر میکنه، میتوانیم یک خازن رو به ترمینال تغذیه متصل کنیم تا جریان رو هموارتر و منظم تر کنه.



### Code for One servo by PCA9685

باید اول کتابخانه ی [Adafruit\\_PWMServoDriver](#)<sup>4</sup> رو اضافه کنیم.



<sup>4</sup> راهنمای اضافه کردن کتابخانه

```

    /// one Driver with a servo connect to 7th pin of Driver board  \\\

#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>

#define Servo_pin 7

Adafruit_PWMServoDriver servo = Adafruit_PWMServoDriver();

int Servo_Max = 600;
void setup() {
    Serial.begin(9600);
    Serial.println("16 channel PWM test!");

    servo.begin();
    servo.setPWMPFreq(1600); // This is the maximum PWM frequency

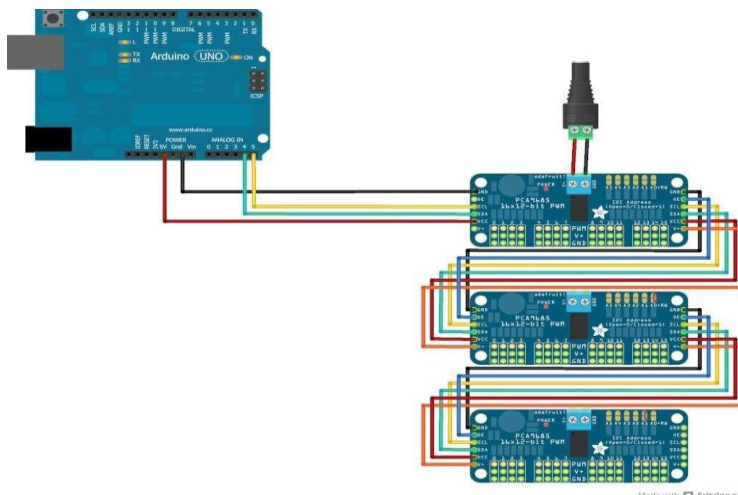
}

void loop()
{
    servo.setPWM(Servo_pin , 0 , Servo_Max); //setPWM(channel, on, off)
    delay(300);
}

```

## Chaining Drivers

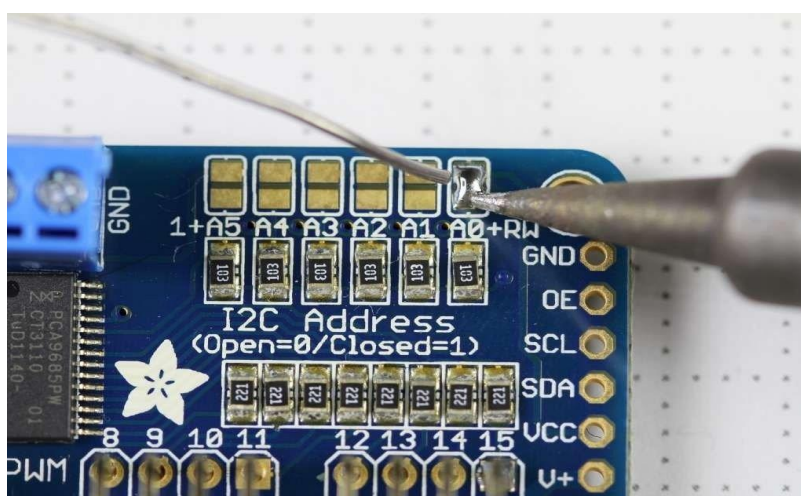
اگر به کنترل بیشتر از 16 سرووموتور نیاز داشتیم، میتونیم چند درایور رو از طریق پین هدرهای (pin header) دو طرف برد، بصورت موازی بهم متصل کنیم.



برای کد زدن برای درایورهای موازی، باید آدرس بردها رو عوض کنیم.

### Addressing the boards

در زنجیره ای از درایورها هر برد باید آدرس منحصر بفردی داشته باشه. برای تغییر دادن آدرس آنها میتونیم جامپرهای پشت برد رو مثل شکل زیر بهم متصل کنیم.



آدرس اولیه ای که هر برد داره معمولاً 0x40 هست. بصورت باینری اگر بخوایم بگیم، درحالتی که جامپرهای در حالت صفر قرار دارند مقدار Offset میشه 0000.

Address : 0x40 >> Offset : 0000 (no jumpers required)

Address : 0x41 >> Offset : 0001(bridge A0 as in the photo above)

Address : 0x42 >> Offset : 0010 (bridge A1)

Address : 0x43 >> Offset : 0011(bridge A0 & A1)

Address : 0x44 >> Offset : 0100(bridge A2)

.....

### Code for Two PCA9685s

```
//// Two Servo Drivers in parallel connection \\\n\n#include <Wire.h>\n#include <Adafruit_PWMServoDriver.h>\n\nAdafruit_PWMServoDriver pwm1 = Adafruit_PWMServoDriver(0x40);\nAdafruit_PWMServoDriver pwm2 = Adafruit_PWMServoDriver(0x41);\n\nvoid setup() {\n  Serial.begin(9600);\n  Serial.println("16 channel PWM test!");\n\n  pwm1.begin();\n  pwm1.setPWMPFreq(1600); // This is the maximum PWM frequency\n\n  pwm2.begin();\n  pwm2.setPWMPFreq(1600); // This is the maximum PWM frequency\n}
```



1. [https://www.tutorialspoint.com/arduino/arduino\\_servo\\_motor.htm](https://www.tutorialspoint.com/arduino/arduino_servo_motor.htm)
2. <https://lastminuteengineers.com/servo-motor-arduino-tutorial>
3. [https://www.tutorialspoint.com/arduino/arduino\\_servo\\_motor.htm](https://www.tutorialspoint.com/arduino/arduino_servo_motor.htm)
4. <https://barghnews.com/fa/news/40902/%D9%87%D9%85%D9%87-%DA%86%DB%8C%D8%B2-%D8%AF%D8%B1%D8%A8%D8%A7%D8%B1%D9%87-%D8%B3%D8%B1%D9%88%D9%88-%D9%85%D9%88%D8%AA%D9%88%D8%B1>
5. <https://learn.adafruit.com/16-channel-pwm-servo-driver?view=all>
6. <https://www.youtube.com/watch?v=vvqUnWweFDs>
7. <https://forum.arduino.cc/t/pca9685-driver-and-simultaneous-servo-motors-issues/623755>
8. [joshschertz.com](http://joshschertz.com)