جامعة المنارة
قسم هندسة الميكاترونكس

الآلات الكهربائية
Electrical Machines

8

الدكتور المهندس
علاء الدين أحمد حسام الدين

مفردات المقرر

- مفاهيم عامة في الآلات الكهربائية.
- المحولات الكهربائية.
- آلات التيار المستمر.
- مبادئ عامة في آلات التيار المتناوب.
- الآلات التحريضية ثلاثية الأطوار.
- محركات الخطوة.
- محركات السيرفو.

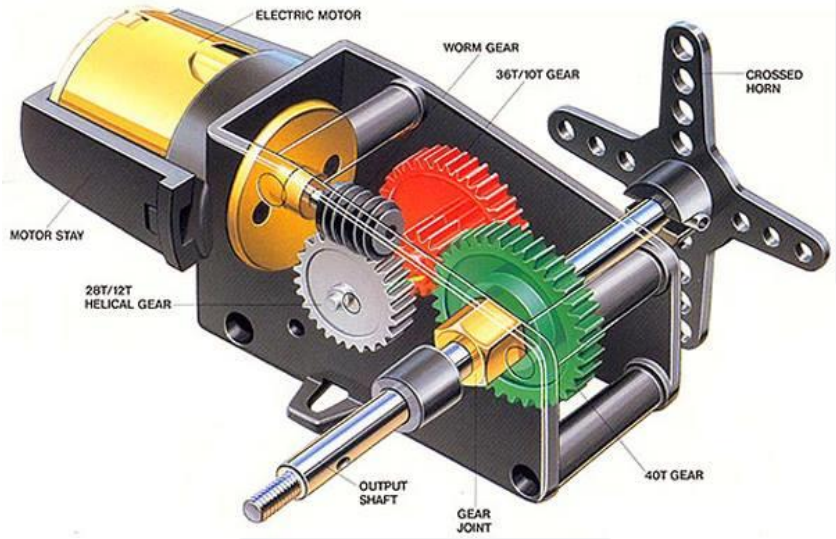
محركات السيرفو

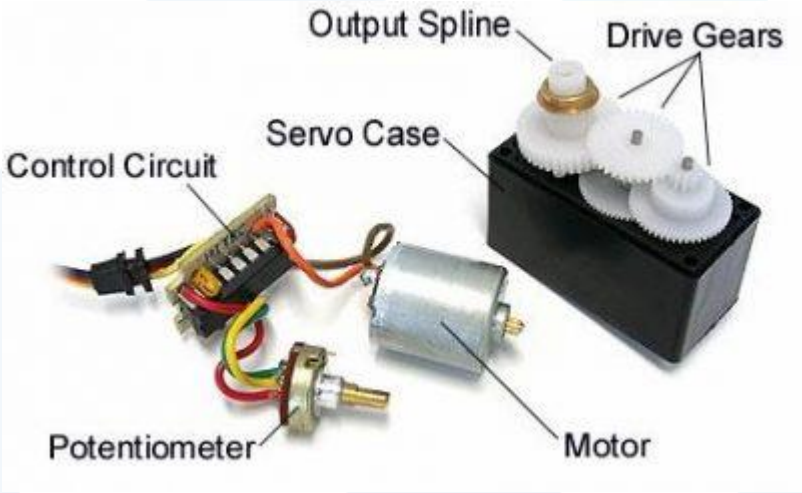
Servo Motors

محركات السيرفو (محركات الزاوية) Servo Motors

هي إحدى أنواع المحركات الخاصة، ولها نوعان: محركات تعمل على تيار مستمر DC، ومحركات تعمل على التيار المتناوب AC.
وغالبية هذه الأنواع تستخدم أسلوب التحكم بالمتحرض Armature Control لتحقيق التحكم بالموضع Position Control عن طريق تغيير قيمة الجهد.

تزود محركات السيرفو بعلبة سرعة ميكانيكية مدمجة تؤمن العزم المطلوب للتحريك وتحتوي على تشكيلة كهربائية تؤمن للمحرك تغذية راجعة أو عكسية Feedback.





أنواع محركات السيرفو:

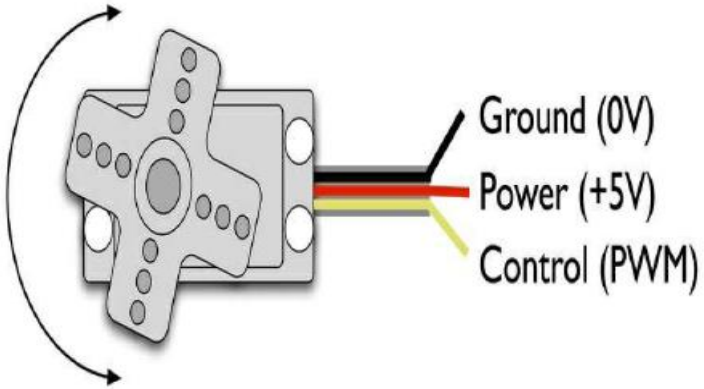
يوجد نوعين من محركات السيرفو:

المحرك القياسي Standard Servo Motor: القابل للدوران بزاوية من 0^{o} حتى 180^{o}.

المحرك المتواصل Continuous Servo Motor: القابل للدوران بزاوية من 0^{o} حتى 360^{o}.

آلية عمل هذه المحركات:

لمحرك السيرفو ثلاثة أقطاب الأول (لون أحمر) يغذى بجهد 5 V أو 6 V والثاني (لون أسود) هو الجهد المرجعي GND، أما القطب الأخير (لون أبيض أو أصفر) فهو قطب التحكم. يمرر إلى قطب التحكم نبضات متغيرة العرض Pulse Width Modulation PWM ذات عرض يتراوح بين 1-2 ms التي توافق دوران للمحرك من 0-180°, (وبعضها حتى 360°). هذه النبضات تطبق بشكل مستمر على المحرك وبتردد 40-50 Hz (لا يعمل المحرك بدون نبضات التحكم), وتجدر الإشارة إلى أهمية حدود عرض النبضة، وترددها، لأن أي خروج عن المجالات المذكورة يؤدي إلى تلف المحرك.

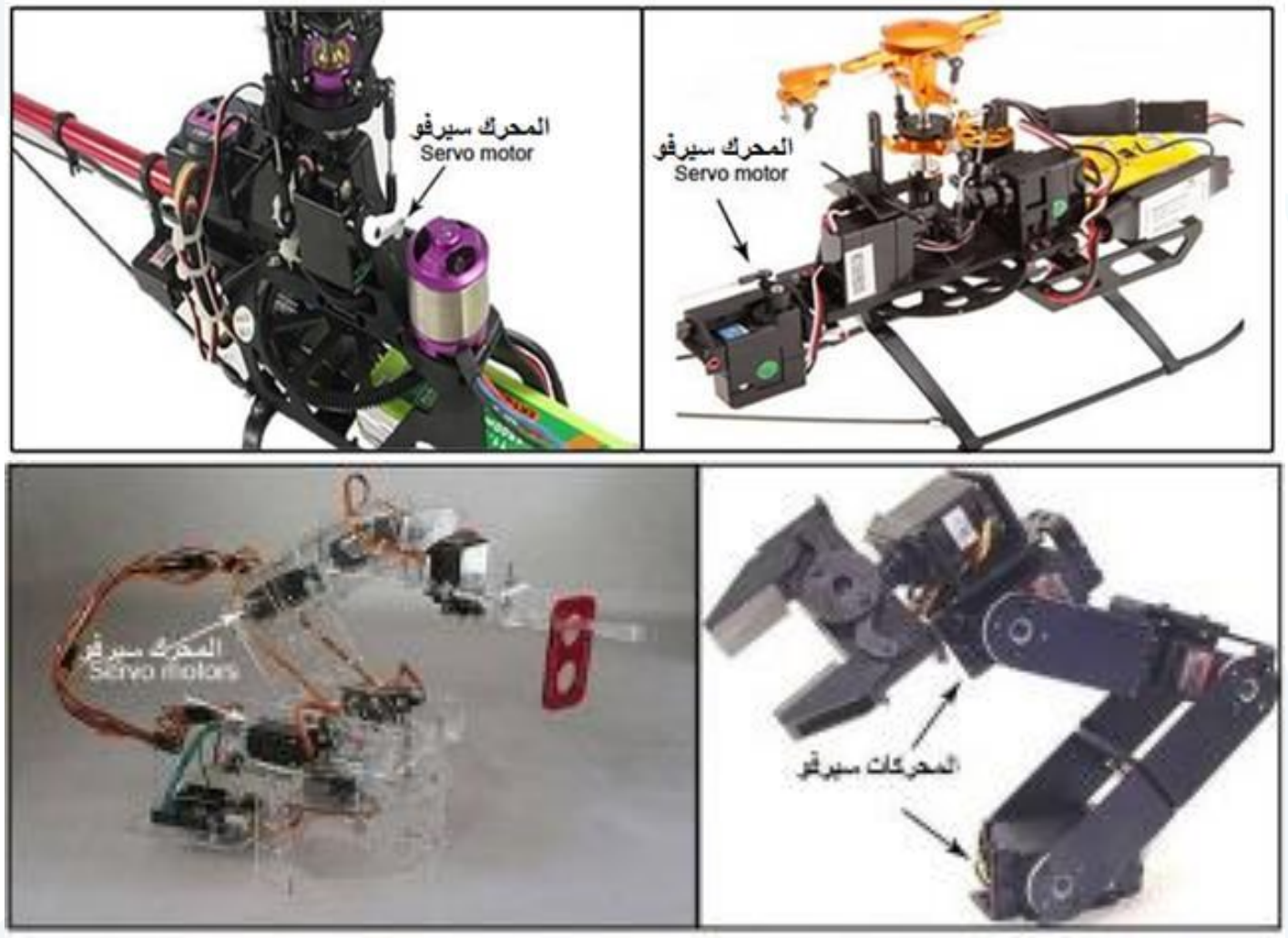


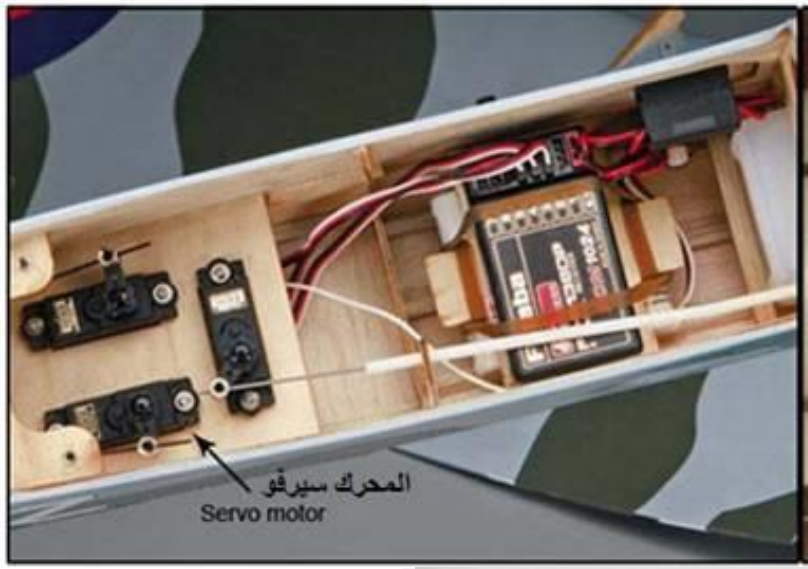
مزايا محركات سيرفو:

1. استجرار ضعيف نسبياً للتيار مقارنةً مع محركات الخطوة.
2. عزوم قوية يمكن أن تصل إلى 40 [kg.cm] وذلك لوجود علبة السرعة الميكانيكية المدمجة.
3. التوافر بأحجام مختلفة لتلائم جميع التطبيقات (الروبوتات الحشرية والروبوتات البشرية).
4. امكانية العمل لفترات طويلة دون ارتفاع في الحرارة.
5. حساسية كبيرة لتغيرات عرض النبضة المطبقة الأمر الذي يتطلب دقة هائلة في التحكم.
6. صغر دارة القيادة مقارنةً مع غيرها من المحركات.

تستعمل محركات السيرفو في تطبيقات الروبوتات الصغيرة والميكانيزمات التي تتطلب دقة في التحريك و ذلك لما تمتازبه هذه المحركات من عزوم قوية، وسهولة، ودقة في التحكم.

من أمثلة تطبيقات هذا المحرك تحريك أجهزة الرادار، وأطباق استقبال الأقمار الصناعية، ويستخدم أيضاً في تحريك أجنحة الطائرات وبعض أنواع أجهزة الطباعة, ومن الملاحظ أن كل هذه التطبيقات تعمل على سرعه بطيئة جداً, لذلك فإن محركات السيرفو تتميزبسرعات بطيئة جداً.


المحرك سيرفو
Servo motor
المحرك سيرفو
Servo motor
المحرك سيرفو
Servo motors
المحركات سيرفو


المحرك سيرفو
Servo motor

المحرك سيرفو
Servo motor

I-SOBOT

و نظراً لمتطلبات تشغيل هذه الأنواع من الأحمال، يجب أن تتو افر في محرك السيرفو الخصائص التالية:

1. الاستجابة الفائقة للسرعة fast response ، بمعنى أن تصل سرعة المحرك للقيمة الاسمية فورتوصيل المحرك بالمنبع الكهربائي, كما يجب أن يتوقف فورفصل المنبع عن المحرك.
2. أن تكون العلاقة بين الجهد والسرعة علاقه خطيه وذلك من أجل تبسيط منظومة التحكم ومكوناتها و تحسين كفاءته.
3. أن يقبل المحرك تكرارعمليات الفصل والتوصيل مهما تعددت.

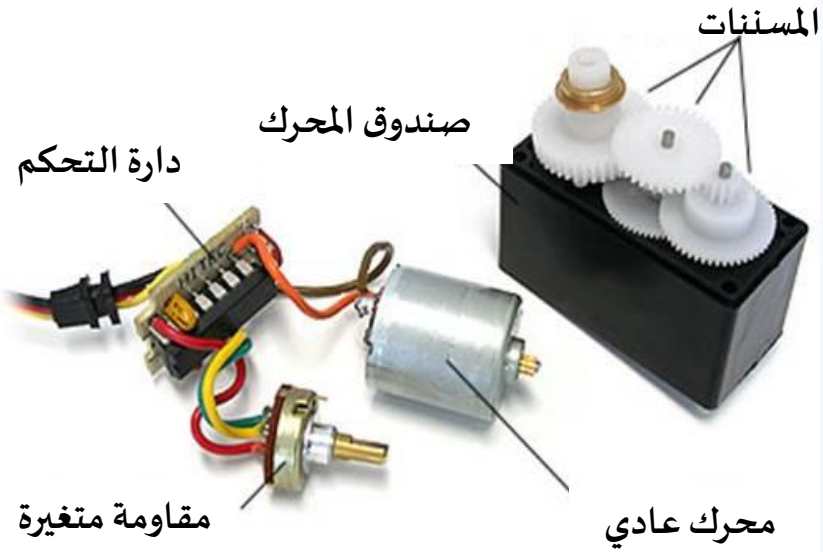
يتكون محرك السيرفو من العناصر الرئيسة التالية:

محرك عادي: هو محرك DC وظيفته القيام بالحركة.

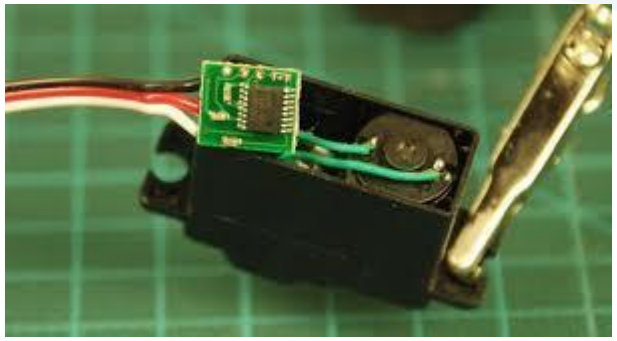
دارة التحكم: وظيفتها استلام إشارة التحكم من المتحكم (المايكروكونترولر) وتشغيل المحرك، ومن خلالها يتم التحكم بزاوية الدوران.

مجموعة المسننات: وظيفتها مضاعفة السرعة وزيادة العزم، وتساعد على الدوران في زاوية محددة.

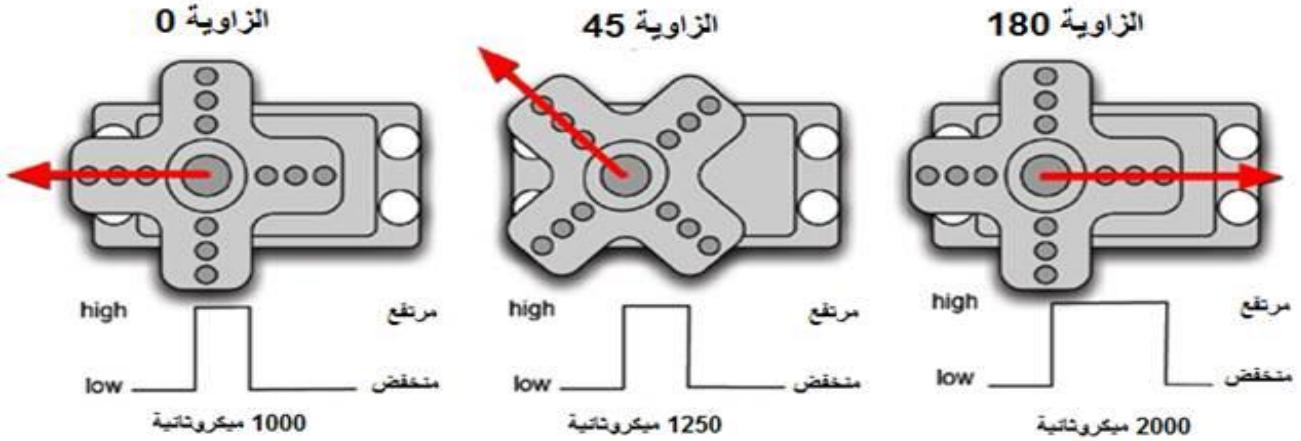
مقاومة متغيرة: وظيفتها إعطاء قيمة جهد تناظر وضع محور دوران المحرك بناء على قيمة مقاومتها، وهي تتحرك مع محور دوران المحرك.



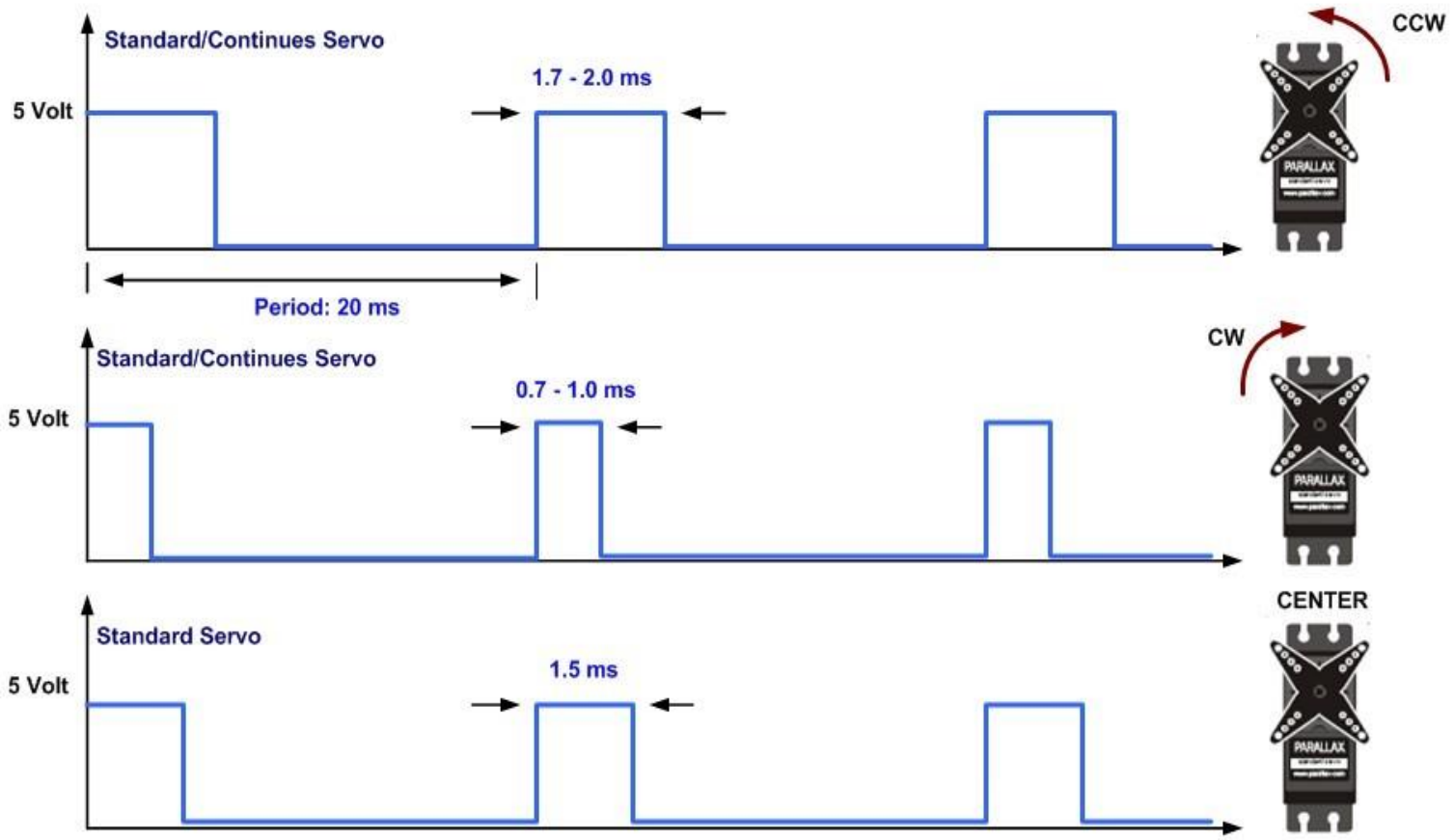
طريقة التحكم بمحركات السيرفو:



يتكون محرك السيرفو داخلياً من دارة تحكم تكون في الغالب مايكروكنترولر وعندما نعطي المحرك نبضات **Pulses** بثابت زمني معين يدور المحرك للزاوية حسب هذا الثابت الزمني.



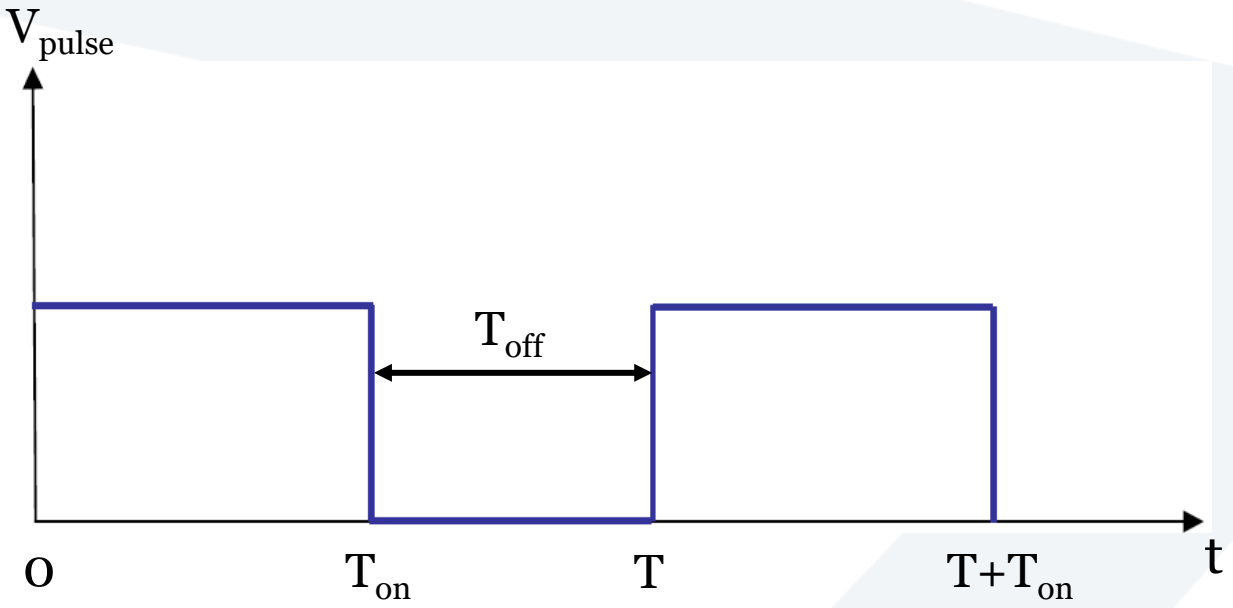
في كل نوع يختلف الثابت الزمني من محرك إلى الآخر حسب جهة التصنيع و النشرة الفنية التي تأتي مع محرك السيرفو.



Servo Motor PWM Timing Diagram

دور التشغيل Duty Cycle:

يمكن الحصول على القيمة المطلوبة لجهد الخرج المستمر عن طريق التحكم بزمن الفصل T_{off} وزمن الإغلاق T_{on} للمفتاح الإلكتروني (الترانزستور مثلاً)، حيث يتم التحكم في إغلاق وفتح المفتاح عن طريق إعطاء نبضة كهربائية لدارة التحكم الخاصة بهذا المفتاح، كما في الشكل، الذي يُظهر نموذج لنبضة التحكم بالمفتاح الإلكتروني.



دور التشغيل D هو النسبة بين زمن إغلاق المفتاح الإلكتروني T_{on} (زمن التشغيل)، وزمن دور نبضة التحكم T (الزمن الكلي)، وتكتب الصيغة الرياضية لدور التشغيل بالعلاقة:

$$D = \frac{T_{on}}{T_{on} + T_{off}} = \frac{T_{on}}{T}$$

D – دور التشغيل.

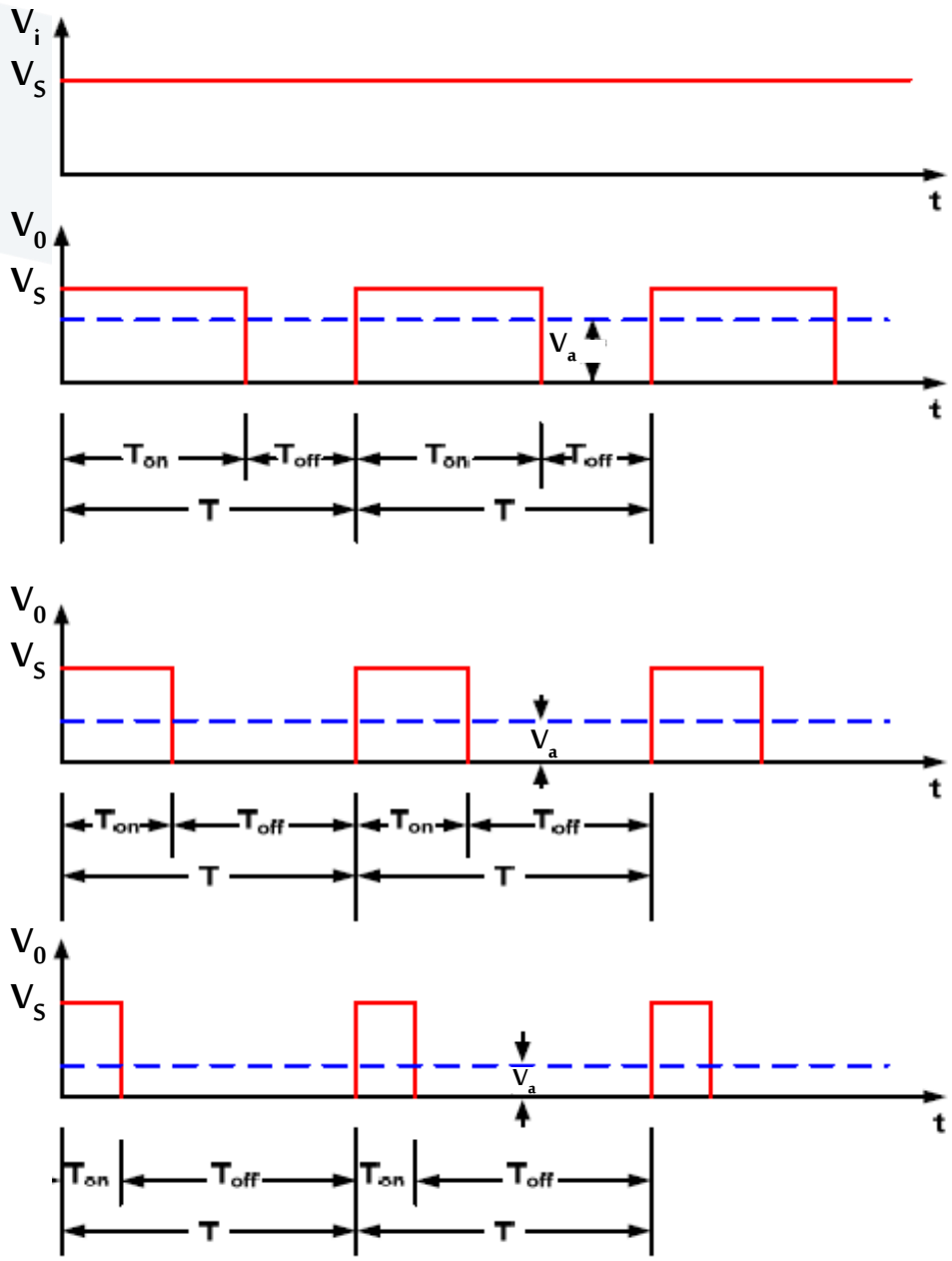
T_{on} – زمن إغلاق المفتاح الإلكتروني (زمن تشغيل الدارة، أو زمن التوصيل).

T_{off} – زمن فصل المفتاح الإلكتروني (زمن فصل الدارة).

T – زمن الدور (الزمن الكلي).

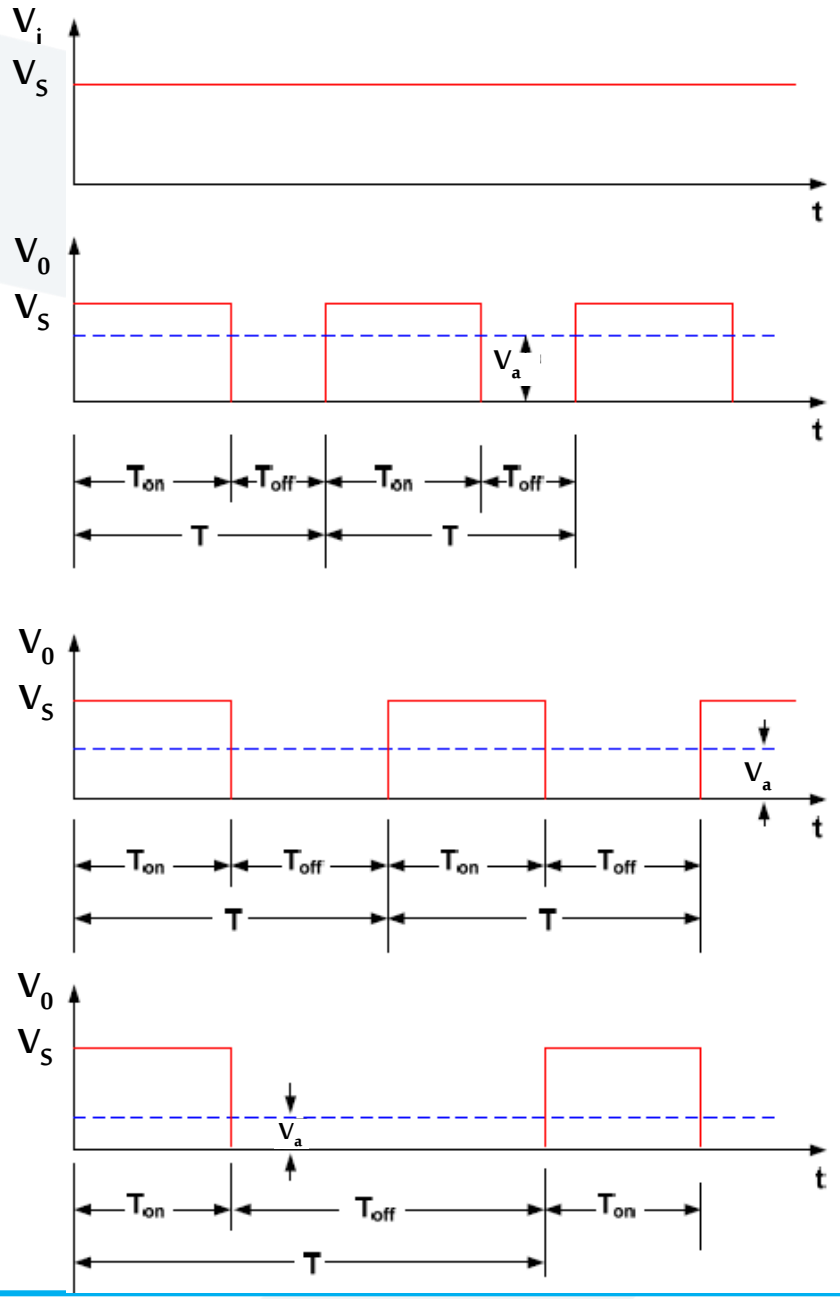
يتم التحكم بالجهد بطريقتين:

1. بتعديل عرض النبضة (Pulse-Width Modulation - PWM).
2. بتعديل تردد النبضة (Pulse-Frequency Modulation –PFM).



1. بتعديل عرض النبضة (Pulse-Width Modulation - PWM).

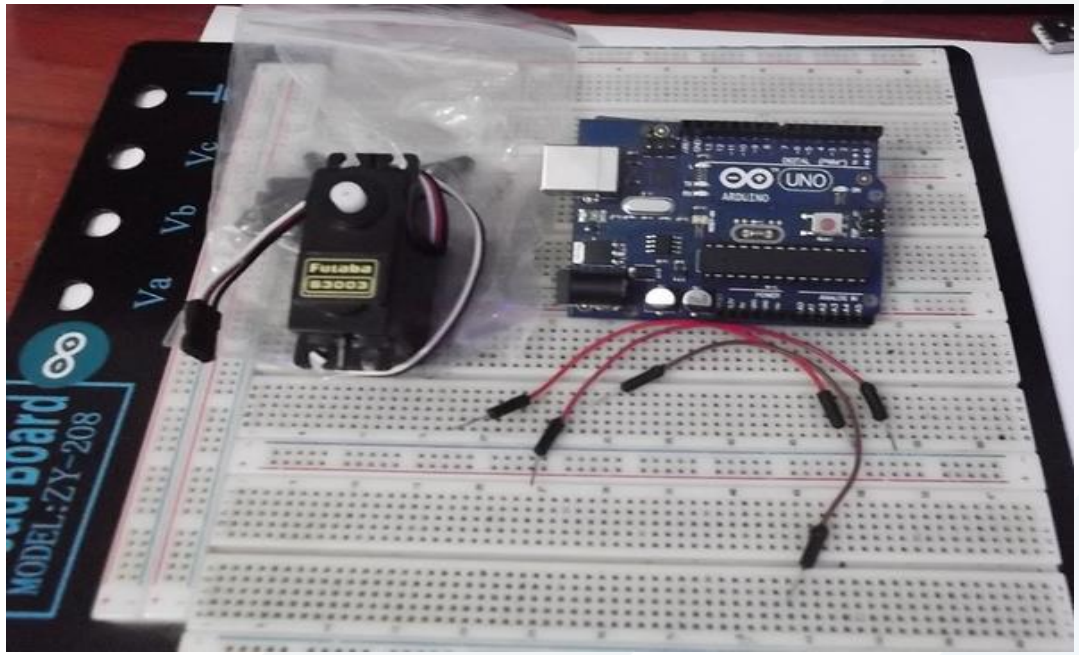
حيث يتم التحكم في جهد الخرج بتغيير فترة التوصيل T_{on} مع المحافظة على زمن الدور T ثابتاً، أي ثبات تردد فتح وإغلاق المقطّع.



2. بتعديل تردد النبضة (Pulse-Frequency Modulation –PFM).

حيث يتم التحكم في جهد الخرج بتغيير زمن الدور T مع المحافظة على زمن التوصيل T_{on} ثابتاً، أي بتغيير تردد فتح وإغلاق المقطّع.

في بيئة أردوينو البرمجية هناك مكتبة تسمى Servo Library مثبتة في البرنامج، هذه المكتبة تعطينا الإمكانية للتحكم في أغلب محركات السيرفو ذو الـ 180º، ومع نهاية هذه التدوينة ستكون لديك القدرة على استخدام أوامر هذه المكتبة عن طريق الأمثلة العملية .

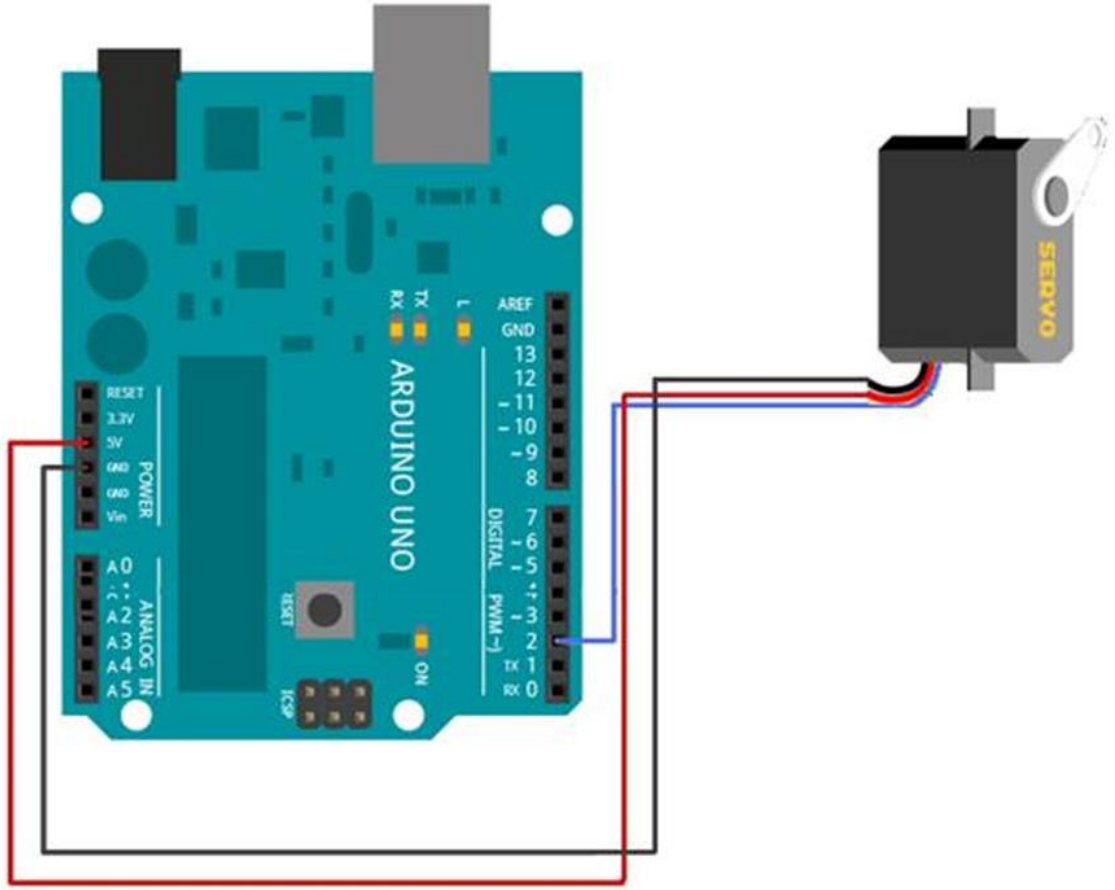


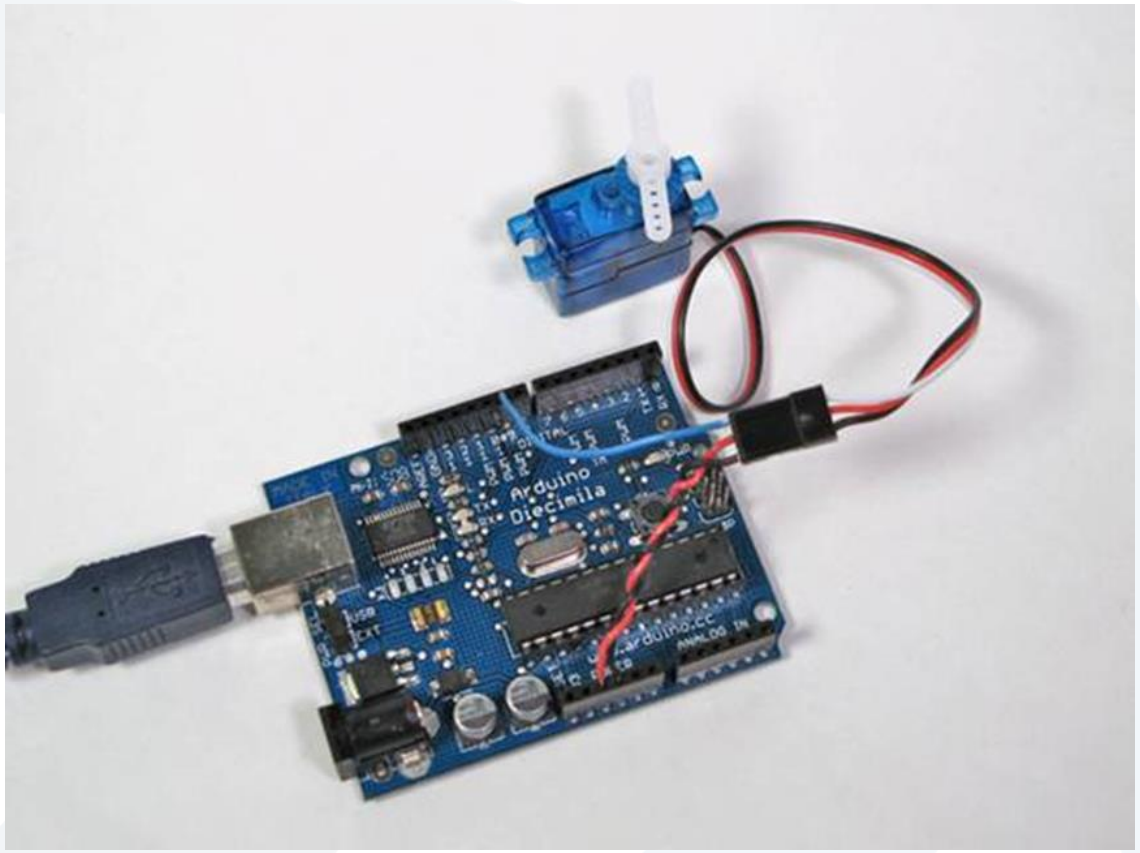
ما سنحتاجه لتنفيذ هذه التجربة:

1. بطاقة أردوينو.
2. محرك سيرفو.
3. اسلاك توصيل.
4. مصدر تغذية خارجي

Power supply , battery

طريقة التوصيل :

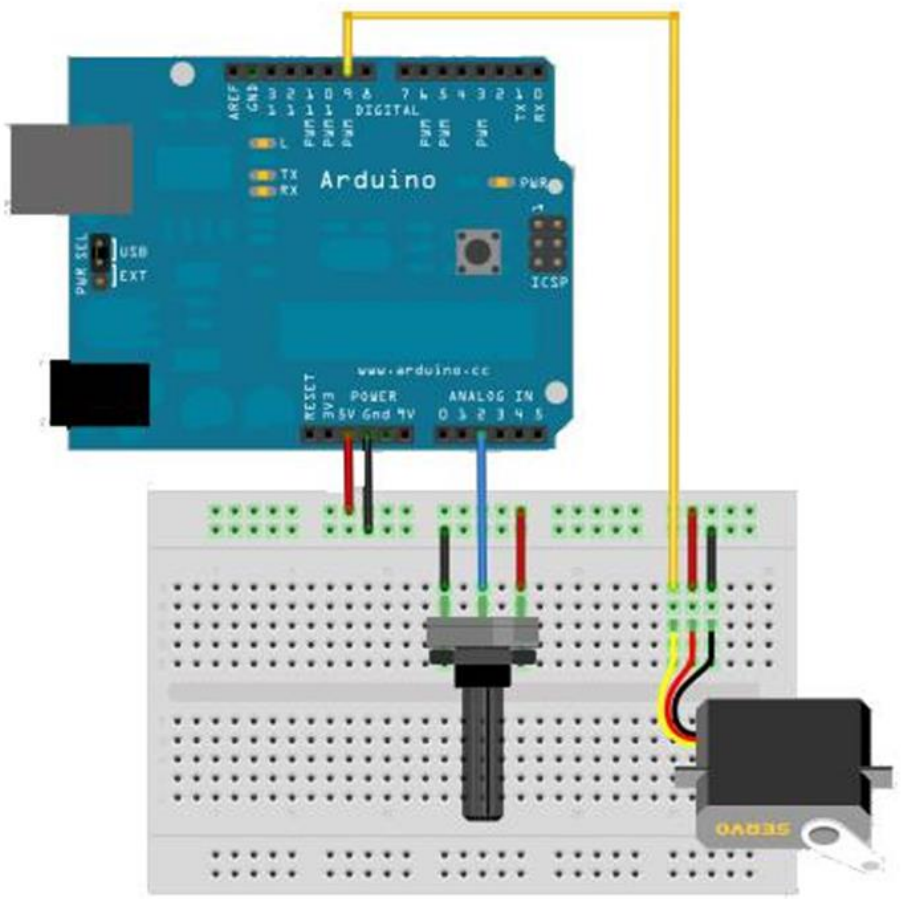



Arduino
Diecimila
Arduino.cc
ANALOG IN

```
#define motoPin 2  // المربط المنطقي المتصل بمربط الاشارة للمحرك
void setup()
{
 pinMode(motoPin,OUTPUT);
}
 void loop()
{
 for(int i = 0; i < 100; i++)
  {
    digitalWrite(motoPin,HIGH);
    delayMicroseconds(1850);
    digitalWrite(motoPin,LOW);
    delayMicroseconds(1850);
  }
 for(int j = 0; j < 100; j++)
 {
    digitalWrite(motoPin,HIGH);
    delayMicroseconds(1250);
    digitalWrite(motoPin,LOW);
    delayMicroseconds(1250);
 }
}
```

هذه شيفرة بسيطة من أجل التحكم بمحرك السيرفو برمجياً:

إضافة مقاومة متغيرة:



يمكن أن نتحكم بالمحرك من خلال مقاومة متغيرة (potentiometer). يتم وصل قطبي التغذية للمحرك والمقاومة المتغيرة بمنبع التغذية ثم نقوم بوصل مربط الإشارة للمحرك بالقطب المنطقي 9 مثلاً للبطاقة. ثم نوصل مربط الاشارة للمقاومة المتغيرة بقطب تناظري للبطاقة.

```
#include <Servo.h>
Servo myservo; // انشأ متغيرا يمثل المحرك سيرفو
int potpin = 2;  // المربط التناظري للبطاقة المتصل بمربط الإشارة للمقاومة المتغيرة
int val;         //  متغير مساعد

void setup()
{
  myservo.attach(9);      // نصل المربط المنطقي 9 بمربط الاشارة للمحرك
}

void loop()
{
  val = analogRead(potpin);          // قراءة قيمة المقاومة المتغيرة (ستكون محصورة بين 0 و 1023)
  val = map(val, 0, 1023, 0, 179);  //تحويل القيمة إلى قيمة محصورة بين 0 و179
  myservo.write(val);               // نطبق القيمة المحولة على المحرك
  delay(15);                        //ننتظر لبعض الوقت
}
```

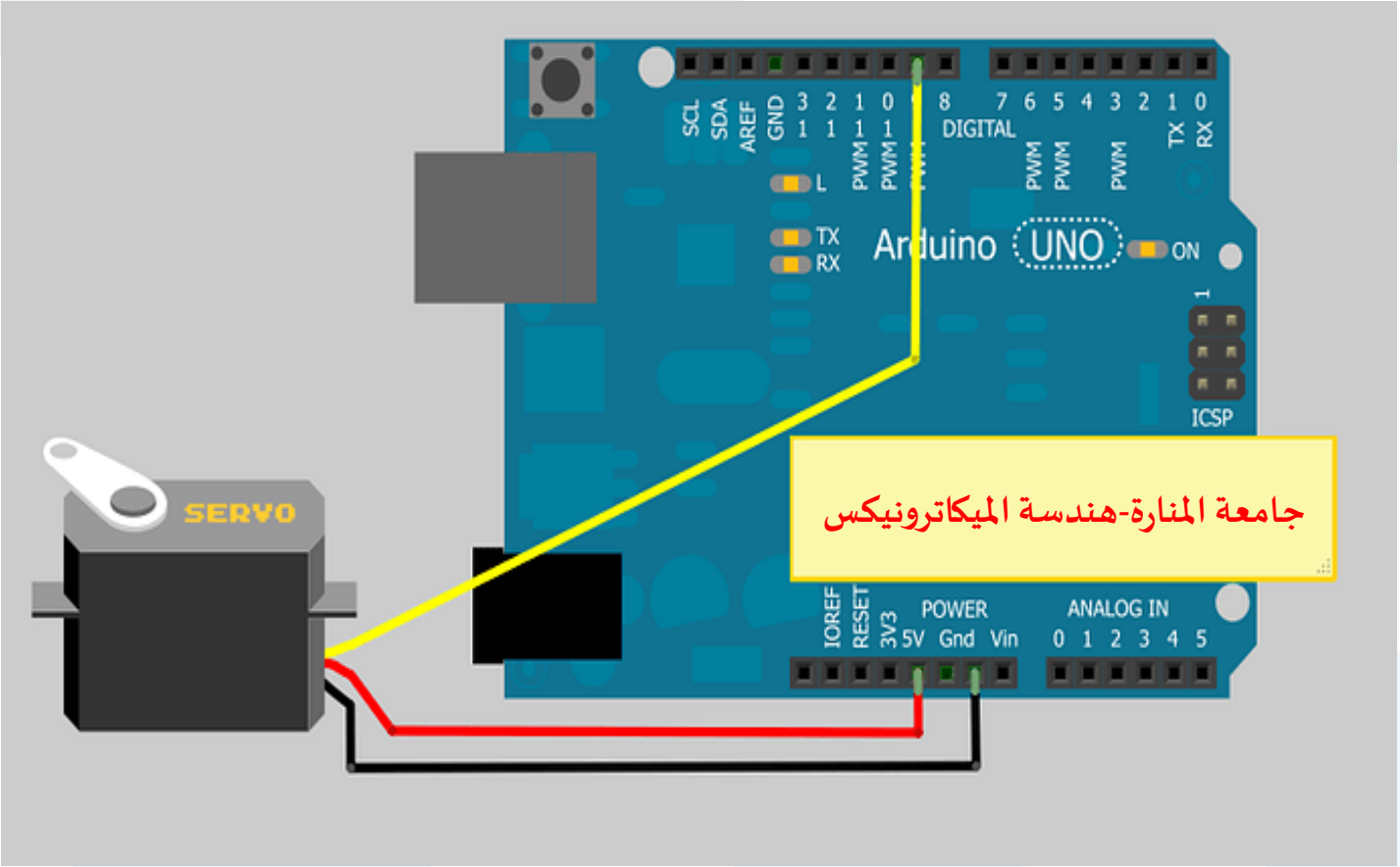
```
*/
بواسطة جهاز خارجي بالمحرك سيرفو
*/
// المربط المنطقي الموصل بالمحرك
int servoPin = 2;

int pulseWidth = 0;    //    وسع الذبذبة المخصصة لزاوية المحرك
long lastPulse = 0;    //    آخر ذبذبة تم حسابها بالميلي ثانية
int refreshTime = 20;  //   الوقت المخصص بين كل ذبذبتين
int val;               //       متغير مساعدة لقراءة الأمر من المنفذ المتتالي

int minPulse = 500;    //    أصغر وسع للذبذبة
int maxPulse = 2250;  // أكبر وسع للذبذبة

void setup()
{
 pinMode(servoPin, OUTPUT); //   مربط المحرك يعرف كمخرج
 pulseWidth = minPulse;          //       تهيئة المتغير
 Serial.begin(9600);             //       تهيئة سرعة التحويل
 Serial.println("Servo control program ready");
}

void loop()
{
```

التحكم من خلال لوحة المفاتيح أو ما شابه

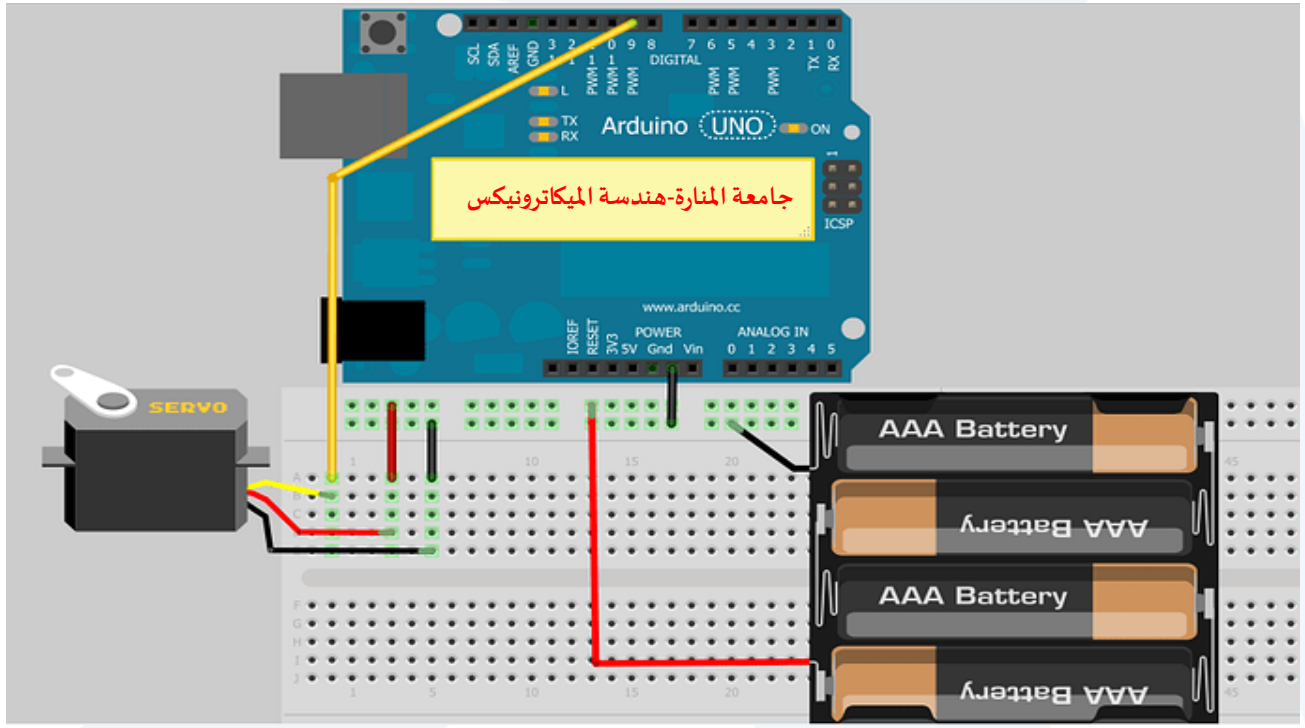
وهذه الشيفرة متوسطة التطور تمكنك من التحكم بالمحرك من خلال لوحة المفاتيح، مثلاً عن طريق المنفذ التفرعي:

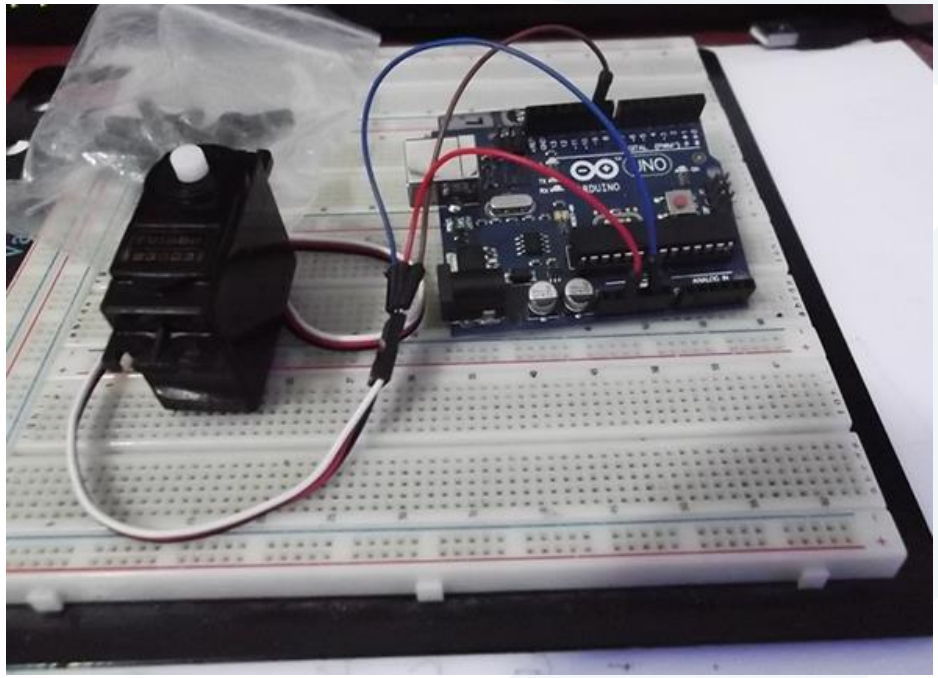
```
val = Serial.read();      //قراءة قيمة القادمة من المنفذ المتتالي
  if (val >= '1' && val <= '9' )
  {
    val = val - '0';   //          تحويل القيمة إلى قيمة رقمية
     val = val - 1;    //          حصر القيمة بين 0 و 8
     pulseWidth = (val * (maxPulse-minPulse) / 8) + minPulse;  // تحويل إلى الميكرو ثانية
     Serial.print("Moving servo to position ");
    Serial.println(pulseWidth,DEC);
  }
  updateServo();   // تحديث موضع المحرك
{

 //servoPi, pulsewidth, lastPulse, refreshTime  تستعمل المتغيرات
void updateServo()
{
  // تحريك رأس المحرك في كل 20 ميلي ثانية
  if (millis() - lastPulse >= refreshTime)
 {
    digitalWrite(servoPin, HIGH);         //     تشغيل المحرك
     delayMicroseconds(pulseWidth);  // طول الذبذبة
     digitalWrite(servoPin, LOW);         //       إيقاف المحرك
     lastPulse = millis();                        //          احتفاظ بقيمة الذبذبة
{
{
```

طريقة توصيل ثانية:



تذكر إذا كان محرك السيرفو الذي تملكه كبير أو يحتاج لتيار عالي, يجب توصيله مع مصدر جهد خارجي كالبطارية وهو أفضل و أكثر أماناً.





لمحرك السيرفو الذي نستخدمه 3 اسلاك وهي :

1.مدخل تغذية Vcc ويكون بين 4.8-6 V.

2.مدخل الأرضي Gnd.

3.مدخل التحكم Data والذي نرسل الأوامر عبره لمحرك السيرفو.

```
//for Genotronex blog
//connect Data pin of servo with D9
#include<Servo.h>
Servo S3003; //Define the name of Servo object
int t=200;
intServo_angle=0; // set the angle of servo
voidsetup(){
S3003.attach(9);//connect servo to Pin D9
Serial.begin (9600);
}
voidloop(){
for(Servo_angle=0;Servo_angle<180;Servo_angle+=9) {
S3003.write(Servo_angle);
delay(t);
}
for(Servo_angle=180;Servo_angle>0;Servo_angle-=9){
S3003.write(Servo_angle);
delay(t);
  }
}
```

بعد هذا نقوم بإجراء البرمجة:

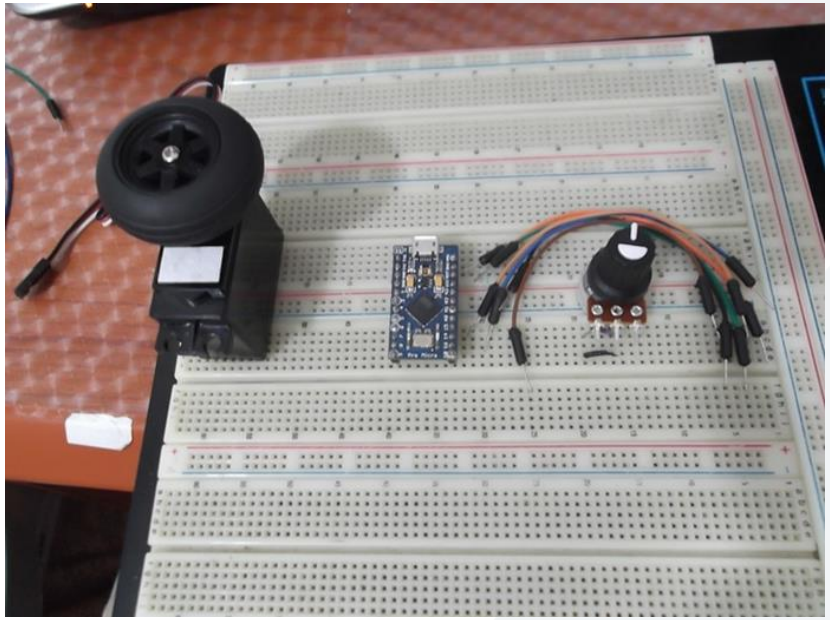
هذا البرنامج البسيط يقوم بتدوير المحرك بمقدار 9^{o} كل مرة بثابت زمني 200 ms مع وعكس عقارب الساعة.

لنتذكر أمراً مهماً جداً هو أن السيرفو يحتاج لنبضات دقيقة للتحكم به، وليست كل مداخل المتحكم مؤهلة للتعامل مع محرك السيرفو. يمكنك استخدام أي مدخل من المداخل التناظرية Analoge (A0-A5) واستخدام المداخل الرقمية التي تدعم التعديل النبضي PWM وهي (D3,D5,D6,D9,D10,D11) في بطاقات الأردوينو التي تستخدم المتحكم ATMEGA328 مثل الـ UNO

محركات السيرفو ذات الدوران المتواصل 360º

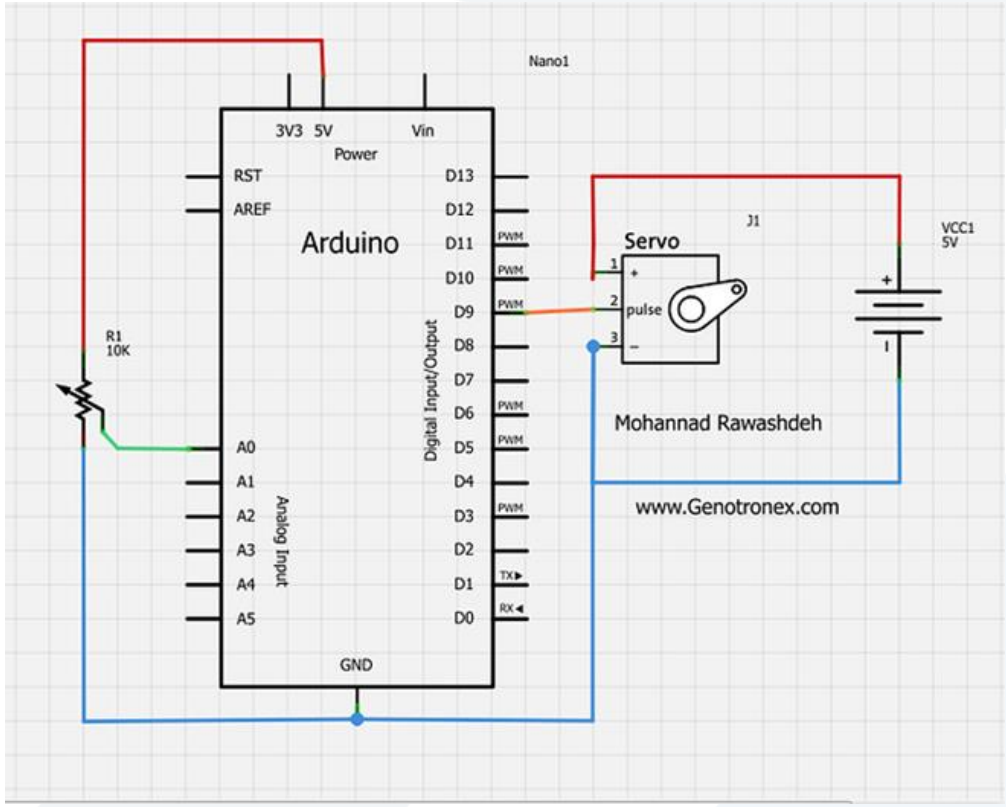
هذه المحركات تستطيع بناء روبوت بسهولة عن طريقها، هناك طريقتين لبرمجتها، باستخدام المؤقتات Timer وتفعيل اوامر الـ Microseconds delay، وطريقة أخرى سهلة من نفس مكتبة الأردوينو الرسمية للسيرفو، وهي ما سنتطرق به هنا لسهوله التحكم به من خلالها .

ما سنحتاج إليه لتنفيذ هذه التجربة



1. بطاقة أردوينو (استعملت هنا بطاقة Arduino Leonardo Micro Pro الرائعة).
2. محرك سيرفو 360º
3. مقاومة متغيرة.
4. اسلاك توصيل.
5. مصدر جهد خارجي لتغذية السيرفو .

تحذير: يجب الحذر عند تغذية محركات السيرفو مباشرة من مخرج الـ 5V الخاص بالأردوينو لان تيار السيرفو العالي قد يتسبب في إتلاف بطاقة الأردوينو، لذلك يفضل أن نستخدم مصدر جهد خارجي ذو تيار كافي (5V , 1 A).



طريقة التوصيل :

البرمجة:

```
/*
this program taken from arduino Example .
http://www.genotronex.com
   This code used to control Servo Motor 360 Degree using potebtiometer connected on A0
*/
#include<Servo.h>
int t=10;
Servo SR04;
// Full rotational
intPinReading=0;
int potentiometer=0;
voidsetup(){
// put your setup code here, to run once:
SR04.attach(9);
// servo connected to D9
pinMode(A0,INPUT);
}
voidloop(){
// put your main code here, to run repeatedly: PinReading=analogRead(A0);
delay(t);
potentiometer=map(PinReading,0,1023,0,180);
potentiometer=constrain(potentiometer,0,180);
SR04.write(potentiometer);
delay(50);
}
```