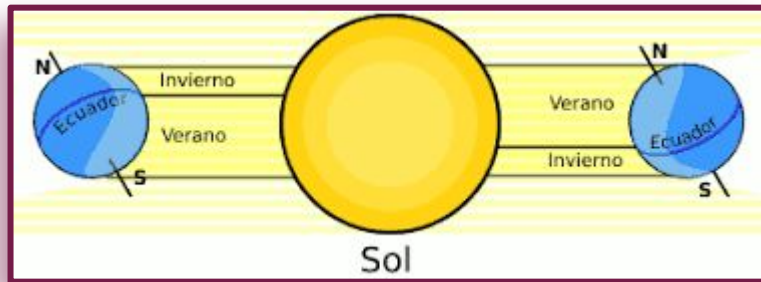


PRÁCTICA A6: MOTORES PASO A PASO (PAP)



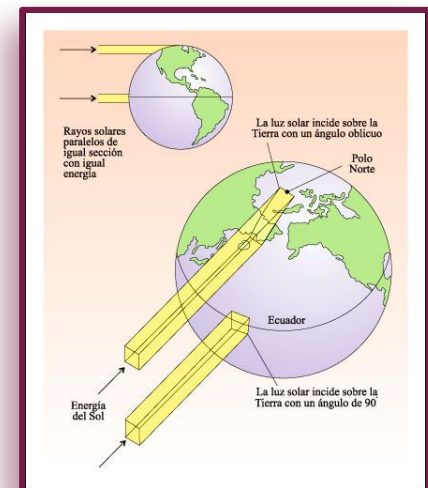
por [Aurelio Gallardo](#)



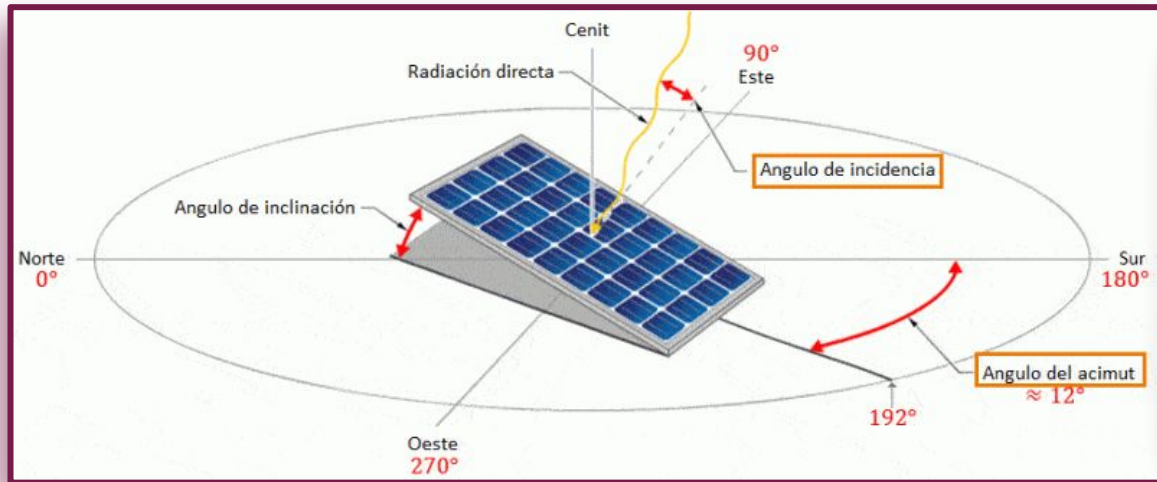
Debido a la lejanía del sol, los rayos solares puede considerarse que inciden paralelos a la línea imaginaria que une los dos astros. Somos nosotros, al encontrarnos en un planeta cuya superficie es esférica, los que estamos inclinados (23.5°).

El sol alcanza su punto más alto (mediodía hora solar), cada día, en un ángulo diferente, dependiendo de la latitud y del momento del año. En los equinoccios el sol en el punto más alto se desvía del cenit justo la latitud del lugar.

<http://www.jocecy.org/puzzle/ambiental/orienmas.htm>



PRÁCTICA A6: MOTORES PASO A PASO (PAP)



Sobre el año 1998 este tema fue el proyecto fin de carrera de mi hermano. Hicimos los cálculos en VB y programamos un PIC16F84A con los datos para la latitud de Sevilla

¿Y qué importancia tiene conocer ese ángulo? Si oriento cada día del año una placa solar fotovoltaica perpendicular a los rayos solares, puedo aumentar hasta en un 30% (en nuestra latitud) la energía recogida por la placa a lo largo del año.

Para mover una placa solar e inclinarla todos los días el ángulo preciso, necesitaré un motor en el que pueda controlar muy bien y con precisión el ángulo que gira. Ese motor puede ser perfectamente **un motor paso a paso**. Y si además la placa va siguiendo el movimiento del sol, cambiando su acimut, más eficacia.

PRÁCTICA A6: MOTORES PASO A PASO (PAP)

Necesitamos saber pues la altura solar (α). Podemos conocerla en función de la latitud (L) y de la declinación (δ) con la expresión: **$\text{Sen } \alpha = \text{sen } L \bullet \text{sen } \delta + \cos L \bullet \cos H \bullet \cos \delta$** .

- H es el ángulo diario. Se calcula sabiendo que a las 12 del mediodía hora solar es 0. Como el sol recorre un arco de 15° cada hora, a las 11:00 su valor es de -15° y a las 13:00 de $+15^\circ$. Si sólo necesito saber el ángulo α a las 12:00, $\cos H = 1$ ($H=0^\circ$).
- δ , la declinación, se puede calcular de forma muy precisa con la fórmula de Spencer (error máximo 0.0006 rad), o con las aproximaciones de Perrin o Cooper. Os dejo esta última, que calcula (en grados) la declinación en función del día juliano del año.
- <https://www.tesisenred.net/bitstream/handle/10803/6839/10Nvm10de17.pdf?sequence=11&isAllowed=y>
- <http://www.geografia.us.es/web/contenidos/profesores/materiales/archivos/PRACTICA2.pdf>

$$\delta = 23.45 \sin \left[\frac{360}{365} (d_n + 284) \right] \quad (1 \leq d_n \leq 365),$$

Motores y Arduino

Actuadores electromecánicos que giren, es decir, “motores eléctricos” susceptibles de ser usados o controlados por Arduino hay muchos. Desde mi modesto conocimiento os pongo aquí una pequeña relación:

- Motores de corriente continua (DC).
- [Servomotores](#): son motores DC que normalmente llevan una reductora para aumentar su par y que mediante un circuito de control y una retroalimentación encoder, **son capaces de determinar su posición con exactitud**. Giran de 0º a 180º. Son muy usados con Arduino.
- [Servomotor de rotación continua](#): es una modificación de los anteriores en los que se puede girar 360º y lo que controlamos con exactitud es la velocidad y sentido de giro. Se pueden usar como motores de un robot móvil.
- [Motores PAP](#): tienen varias bobinas en su interior lo que les permite avanzar una cantidad determinada de grados cada vez, al activarlas en secuencia. ***Es un motor con gran precisión y reproducibilidad de la posición***. Son los típicos de una impresora en 3D, de un lector DVD o un disco duro, etc.
- Motores Brushless: son motores sin escobillas. Necesitan de un circuito adicional para controlarlos, llamados ESC. Son capaces de generar altas velocidades de giro por lo que se emplean en la construcción de drones.

Si queréis saber más, os recomiendo el artículo de Luis Llamas:

<https://www.luisllamas.es/tipos-motores-rotativos-proyectos-arduino/>

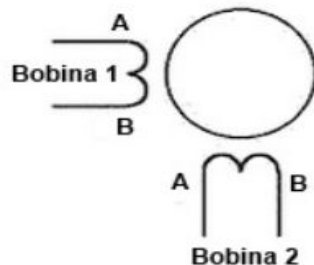
Tipos de motores PAP

Los motores paso a paso se fabrican de dos tipos:

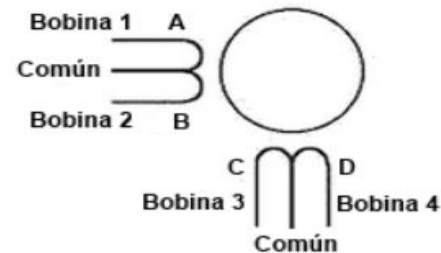
- **Bipolares:** cuentan con dos bobinas, y, por lo tanto para acceder a ellas necesitamos cuatro contactos.
- **Unipolares:** cuentan con dos bobinas pero además hay uno o dos terminales más para conectarles comunes.

El bipolar cuenta con más torque pero es más complejo de manejar. Sin embargo, es muy común usando un unipolar como bipolar anulando o no usando sus comunes. En lo que sigue, usaremos sólo configuraciones bipolares.

Motor paso a paso bipolar



Motor paso a paso unipolar



Drivers

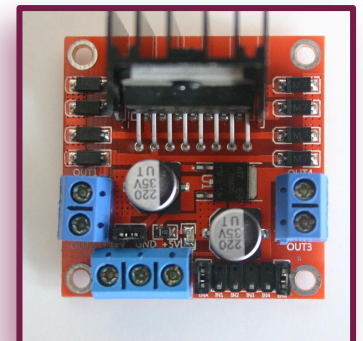
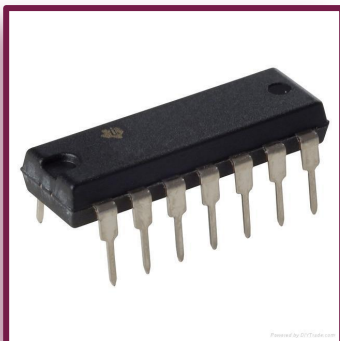
En general, para manejar un motor con Arduino nos encontramos con otro problema. Los motores suelen tener un régimen de funcionamiento cuya tensión nominal varía (posiblemente más elevada de a la que funciona nuestros Arduinos) y consumen una intensidad mayor de la que proporciona las salidas de Arduino (máx 40mA).

Claramente, Arduino es un dispositivo de control; no un elemento de potencia.

Así que necesitamos un circuito que haga de intermediario entre Arduino y nuestro motor. Este circuito se denomina genéricamente *driver*.

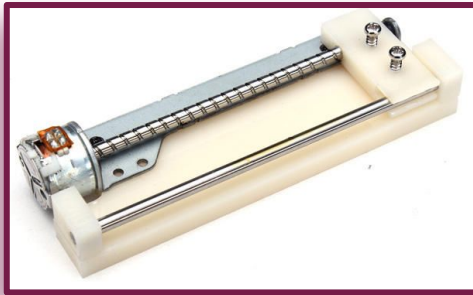
Drivers hay muchos. E incluso podemos usar un transistor BJT o un MOSFET. Tenemos los circuitos ULN2003, L293D (puente-H 4.5~36V 600mA por circuito), L298N (puente-H 3~35V 2A), [A4988](#), [DRV8825](#), [TB6612FNG](#), [TMC2100](#), [TMC2130](#), [TMC2208](#)...

Así que como veis este tema da para mucho. Según la aplicación que necesitemos, elegiremos motor y de qué tipo, y asimismo el driver adecuado.



Motor Paso a Paso DVD

Para nuestra primera práctica vamos a usar un motor PAP bipolar de pequeña potencia que podemos encontrar fácilmente desarmando un lector/grabador de CD o DVD antiguo.



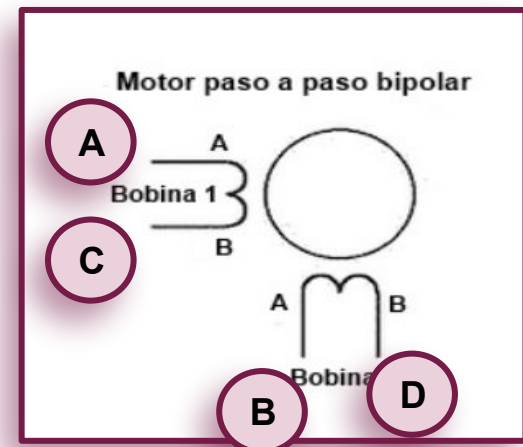
En Internet hay tutoriales para construirse pequeñas CNC (por ejemplo) con este tipo de motores.

Para probarlo, no hace falta saber sus pasos por vuelta. Habría que calcularlo para una aplicación seria. Por internet dicen que 200 ¿?



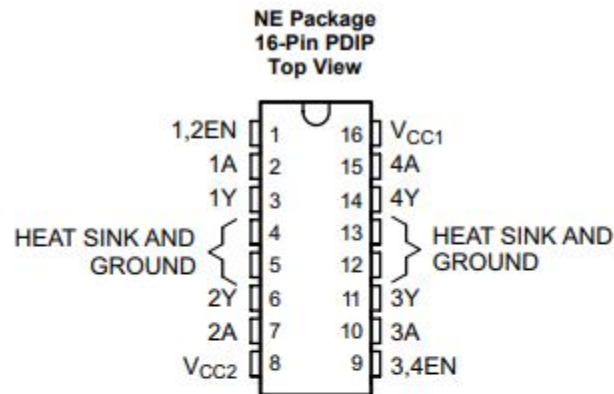
Lo primero es lo primero. Vamos a identificar de las cuatro conexiones las que forman cada bobina. ¿Cómo? Con un voltímetro (¡¡no las supongas!!)

- Escoge un contacto, y ve midiendo la resistencia con el resto.
- Si no arroja un valor es que ese contacto no es de la bobina.
- Si el valor es de unos 10 Ohm, es que es el otro borne de la bobina.
- Nombra a los terminales A y C, para una bobina, y B y D para la otra.



Motor Paso a Paso DVD: esquema de montaje

En segundo lugar, vamos a suponer que el puerto USB (de 500 -2.0- a 900 mA -3.0-) de nuestro ordenador es capaz de entregar, a 5V, la suficiente intensidad como para mover este pequeño motor. Y que éste funciona a 5V. ¿Es descabellado? No tanto; probablemente así sea, ya que no deja de ser un componente que funciona dentro de un lector de CDs que ya funciona dentro de un ordenador.

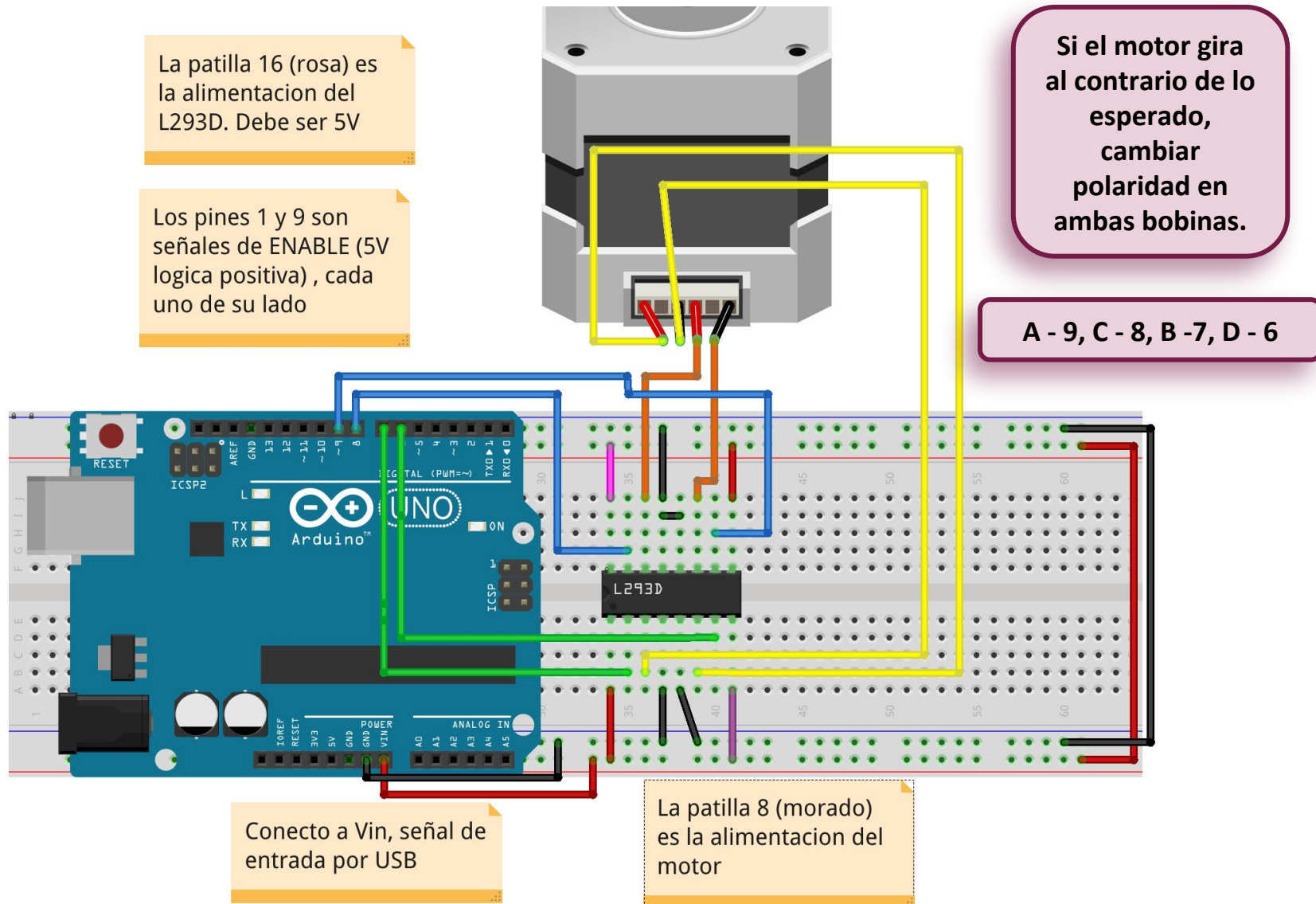


El L293D es capaz de entregar hasta 600mA en la salida. El L293, 1A

Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1:4>A	2, 7, 10, 15	I	Driver inputs, noninverting
<1:4>Y	3, 6, 11, 14	O	Driver outputs
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to printed-circuit-board ground plane with multiple solid vias
V _{CC1}	16	—	5-V supply for internal logic translation
V _{CC2}	8	—	Power VCC for drivers 4.5 V to 36 V

Motor Paso a Paso DVD: esquema de montaje



Motor Paso a Paso DVD: modos de funcionamiento

Hay varias formas de activar un motor bipolar. En la secuencia se muestra el modo **NORMAL GIRO ANTIHORARIO**. Tenemos que programar nuestro **ARDUINO** para que sea capaz de girar tanto a izquierdas como a derechas. Pero hay dos modos más: el modo **WAVE DRIVE** y **MEDIO PASO**. Empezaremos por el modo **NORMAL**...

PA-SO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	ON	OFF	OFF	
2	OFF	ON	ON	OFF	
3	OFF	OFF	ON	ON	
4	ON	OFF	OFF	ON	

La función
“secuencia” en el
programa,
reproduce la
activación del
gráfico

Motor Paso a Paso DVD: programa

```
// Control de un motor PASO A PASO.  
// Por Aurelio Gallardo Rodríguez.  
  
// Establece las constantes a utilizar  
int A=9,C=8; //Pines del motor paso a paso: primer bobinado  
int B=7,D=6; //Pines del motor paso a paso: segundo bobinado  
  
int microSegundosPaso=1500000; //Tiempo en microsegundos entre paso y paso  
  
// *****  
// Funciones  
// *****  
void secuencia(int cual) {  
    digitalWrite(A,((cual==1) or (cual==4))); // El A se activa en la secuencia 1 y 4  
    digitalWrite(B,((cual==1) or (cual==2))); // El B se activa en la secuencia 1 y 2  
    digitalWrite(C,((cual==2) or (cual==3))); // El C se activa en la secuencia 1 y 3  
    digitalWrite(D,((cual==3) or (cual==4))); // El D se activa en la secuencia 1 y 4  
    delayMicroseconds(microSegundosPaso);  
}  
  
// Función de parada motor  
void paro(){  
    secuencia(0); // Sería LOW en todos  
}
```

Motor Paso a Paso DVD: programa

```
// Función izquierdas
void izquierda(){
    secuencia(1); // ver esquema NORMAL
    secuencia(2);
    secuencia(3);
    secuencia(4);
}

// Función derechas
void derecha(){
    secuencia(4); // ver esquema NORMAL
    secuencia(3);
    secuencia(2);
    secuencia(1);
}

// Función que establece el sentido y los pasos del motor
void pasos(int pasos, char sentido){
    for (int i = 0; i < pasos; i++) {
        if (sentido=='d') {derecha();}
        if (sentido=='i') {izquierda();}
    }
}
```

Motor Paso a Paso DVD: programa

```
// *****  
// SETUP  
// *****  
void setup() {  
  pinMode(A, OUTPUT); //Configura A como salida  
  pinMode(B, OUTPUT); //Configura B como salida  
  pinMode(C, OUTPUT); //Configura C como salida  
  pinMode(D, OUTPUT); //Configura D como salida  
  paro();  
}  
  
// *****  
// LOOP  
// *****  
void loop() {  
  pasos(10,'d'); // 10 pasos a la derecha  
  delay (500);   // Una paradita  
  pasos(15,'i'); // 15 pasos a la izquierda  
  delay (500);   // Una paradita  
}
```

Motor Paso a Paso DVD: otros modos

Modo WAVE DRIVE

PA-SO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	OFF	OFF	OFF	
2	OFF	ON	OFF	OFF	
3	OFF	OFF	ON	OFF	
4	OFF	OFF	OFF	ON	

Modo MEDIO PASO

PA-SO	Bobina A	Bobina B	Bobina C	Bobina D	
1	ON	OFF	OFF	OFF	
2	ON	ON	OFF	OFF	
3	OFF	ON	OFF	OFF	
4	OFF	ON	ON	OFF	
5	OFF	OFF	ON	OFF	
6	OFF	OFF	ON	ON	
7	OFF	OFF	OFF	ON	
8	ON	OFF	OFF	ON	

Motor Paso a Paso DVD: otros modos

Modo MEDIO PASO

```
// *****
// Funciones
// *****
void secuencia(int cual) {
    digitalWrite(A,(cual==1) or (cual==2) or (cual==8)); // El A se activa en la secuencia 1,2 y 8
    digitalWrite(B,(cual==2) or (cual==3) or (cual==4)); // El B se activa en la secuencia 2,3 y 4
    digitalWrite(C,(cual==4) or (cual==5) or (cual==6)); // El C se activa en la secuencia 4,5 y 6
    digitalWrite(D,(cual==6) or (cual==7) or (cual==8)); // El D se activa en la secuencia 6,7 y 8
    delayMicroseconds(microSegundosPaso);
}

// Función de parada motor
void paro(){
    secuencia(0); // Sería LOW en todos
}

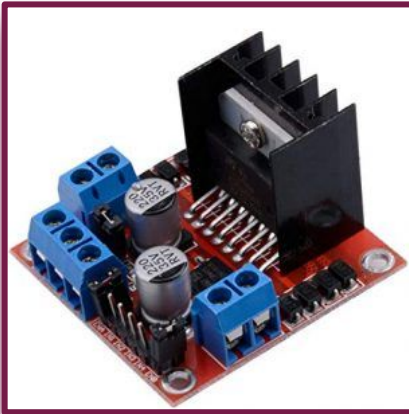
// Función izquierdas
void izquierda(){
    for (int i=0;i<8;i++) {
        secuencia(i+1);
    }
}

// Función derechas
void derecha(){
    for (int i=0;i<8;i++) {
        secuencia(8-i);
    }
}
```

Modo WAVE DRIVE

```
// *****
// Funciones
// *****
void secuencia(int cual) {
    digitalWrite(A,(cual==1)); // El A se activa en la secuencia 1
    digitalWrite(B,(cual==2)); // El B se activa en la secuencia 2
    digitalWrite(C,(cual==3)); // El C se activa en la secuencia 3
    digitalWrite(D,(cual==4)); // El D se activa en la secuencia 4
    delayMicroseconds(microSegundosPaso);
}
```


Motor Paso a Paso → driver L298N

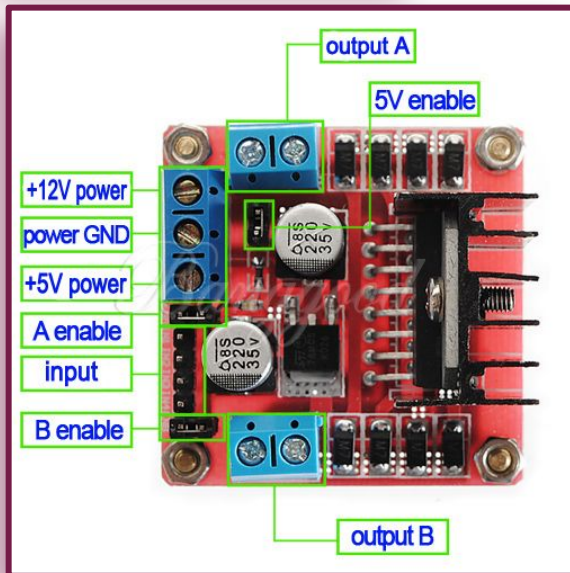


Usaremos el módulo del driver L298N. Este módulo puede trabajar entre los 6 y 35 V de voltaje y proporcionar hasta 2A. Además posee su propio rectificador de corriente a 5V para la parte lógica (que puede habilitarse o deshabilitarse) .

Podemos trabajar con dos motores DC o un motor PAP, ya que tiene salidas para controlar dos bobinados.

Las entradas de control son 3 por bobinado.

- Las salidas se habilitan si en los pines ENABLE pongo 5V. Se deshabilitan si tengo 0V.
- Las entradas IN1, IN2 puedo conectarlas como:
 - IN1=LOW, IN2=HIGH → Gira en un sentido
 - IN1=HIGH, IN2=LOW → Gira en el otro sentido
 - IN1=LOW, IN2=LOW → PARO.
- Para control de velocidad actúo con una salida PWM variable al pin ENABLE.



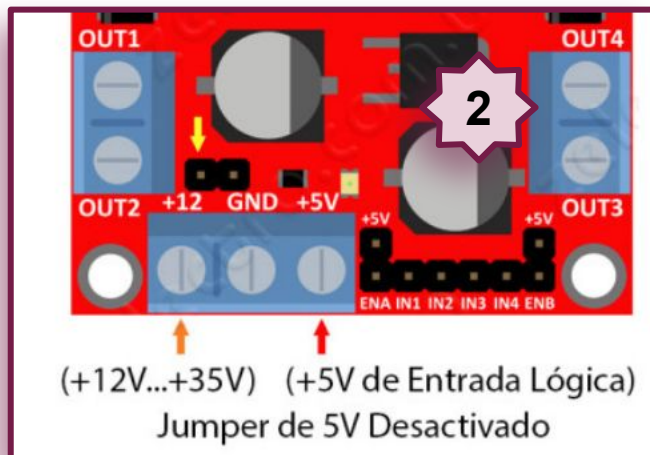
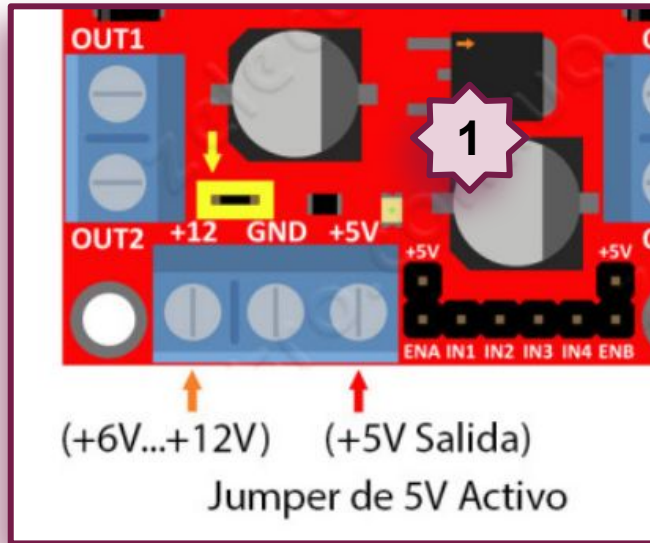
Salidas inabilitadas



Salidas habilitadas



Motor Paso a Paso: driver L298N → configuración



Configuración 1

- Jumper de 5V activo
- Alimentación de motor no puede sobrepasar 12V
- La salida de 5V proporciona tensión para alimentar un ARDUINO u otro elemento de control.

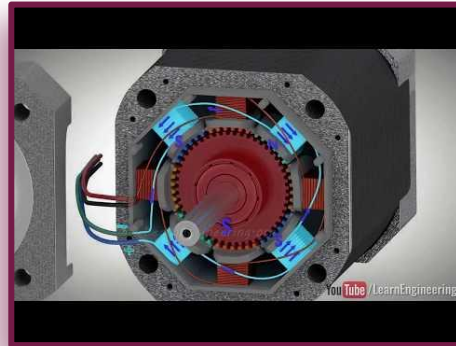
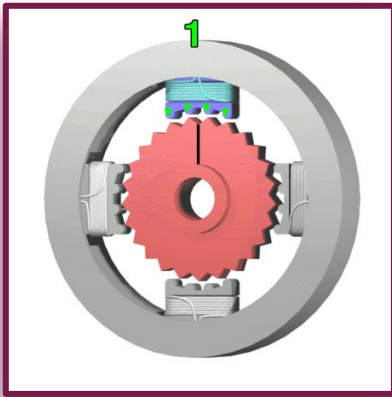
By

<https://electronilab.co/tutoriales/tutorial-de-uso-driver-dual-l298n-para-motores-dc-y-paso-a-paso-con-arduino/>

Configuración 2

- Jumper de 5V inactivo
- Alimentación de motor entre 12 y 35 V
- El pin de 5V tiene que ser alimentado desde un circuito de control.

Motor Paso a Paso: nº de pasos por vuelta



Para comprender mejor cómo controlar el motor hay que conocer cómo funciona, para saber con exactitud lo que es un paso.

<https://youtu.be/CmXVEBX78Rs>

Para averiguar el nº de pasos por vuelta puedes hacer:

1. Mirar la etiqueta del motor (si tiene) y buscar esa información. Si no lo pone directamente, es posible que venga el modelo y buscar por internet.
2. La “cuenta de la vieja”. Suelen tener unos 24,48,100,200 o incluso 400 pasos. Gira el rotor a mano e intenta estimar si tiene un paso pequeño o grande por la sensación que te da. Haz un programa que mueva ese número de pasos y coloca una referencia en el rotor (pega un papel, por ejemplo, o un trocito de cinta aislante). Si no da una vuelta tiene más y si más de una, tiene menos.
3. En lo que sigue, voy a usar un motor STP42D144. Haciendo una búsqueda en internet parece que tiene una tensión nominal de 7V y 200 pasos. El que se compra en solectroshop, el 28BYJ, tiene 48 pasos y es de 5V.

Motor Paso a Paso DVD: conexión

Un bobinado en cada salida

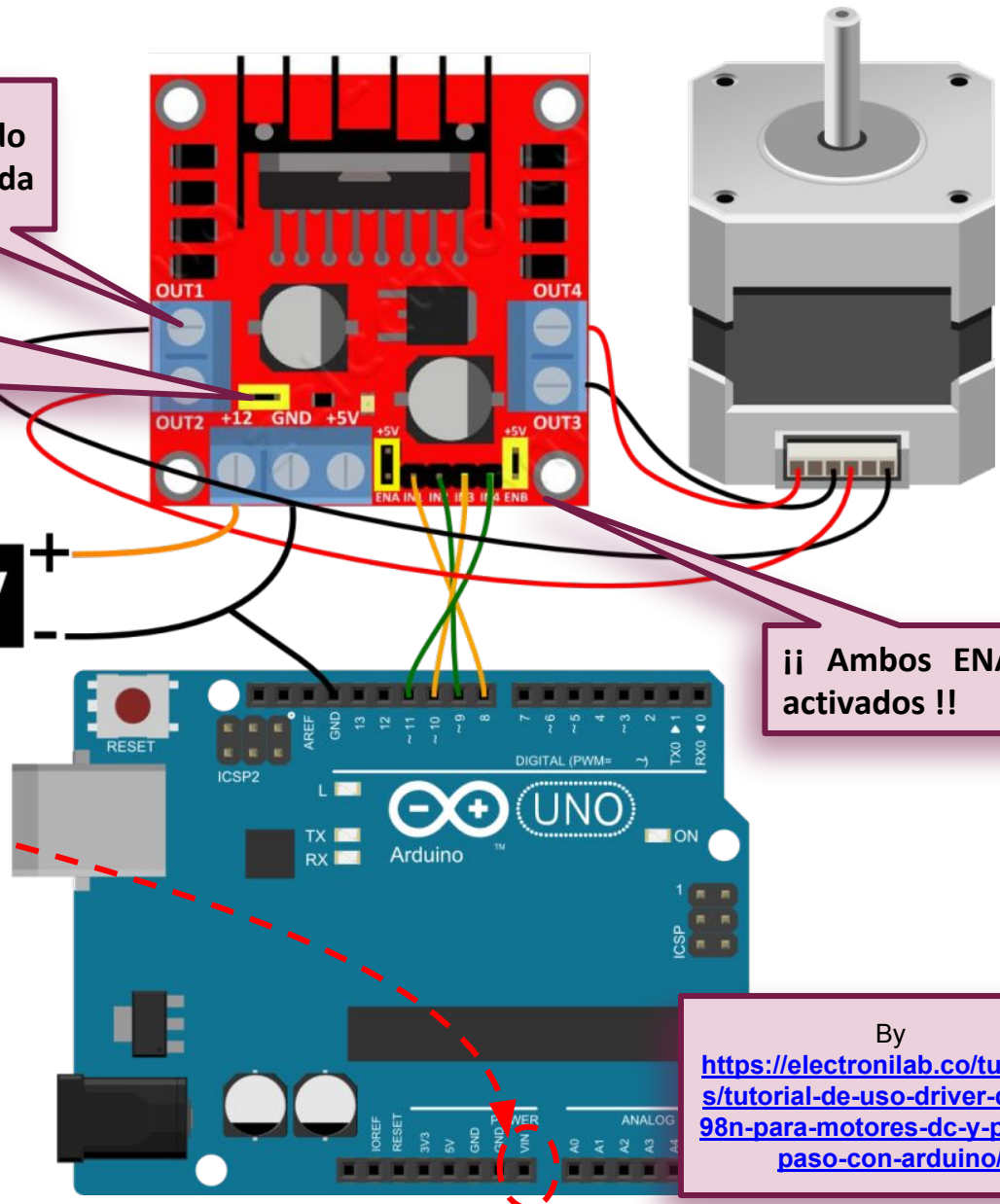
Configuración 1. Imprescindible para 7V. Aunque Arduino lo voy a conectar en principio al puerto USB, podría alimentarlo de forma independiente usando **sus 5V a la entrada Vin.**

6-12V

STP42D144 a 7V. Se puede usar el esquema de baterías TP4056 con Boost Step-up

ii Ambos ENABLE activados !!

By
<https://electronilab.co/tutoriales/tutorial-de-uso-driver-dual-l298n-para-motores-dc-y-paso-a-paso-con-arduino/>



Motor Paso a Paso DVD: programa usando librería STEPPER

```
/*
  Stepper Motor Control - one revolution
  Este programa impulsa un motor paso a paso unipolar o bipolar.
  El motor está conectado a los pines digitales 8 - 11 de la Arduino.
  Created 11 Mar. 2007 by Tom Igoe
  Modificado por Aurelio Gallardo, 16/2/2020
  */

#include <Stepper.h>

const int stepsPerRevolution = 200;
// cambie este valor por el numero de pasos de su motor

// inicializa la libreria 'stepper' en los pines 8 a 11
Stepper myStepper(stepsPerRevolution, 8,9,10,11);

void setup() {
  // establece la velocidad en 5 rpm
  myStepper.setSpeed(5);
  Serial.begin(9600);
}

void loop() {
  Serial.println("clockwise"); // gira una vuelta en una direccion
  myStepper.step(stepsPerRevolution);
  delay(5000); // Espera 5 segundos
  Serial.println("counterclockwise"); // gira otra vuelta en la otra direccion
  myStepper.step(-stepsPerRevolution);
  delay(5000);
}
```


Ejercicio A: ángulo de una placa fotovoltaica

Nuestro proyecto consistirá en el control de un motor PAP que haga girar una placa fotovoltaica y colocarse, cada día del año, perpendicular al ángulo de incidencia de los rayos solares cuando éste llega a su punto más alto en el cielo. Su implantación real exige coordinar y/o programar distintos elementos: motor y reloj. Así que, vamos a simplificarlo:

1. Usa un pulsador que simulará un pulso cuando sean las 4:00 de la mañana (que seguro que es de noche) que le envía un supuesto reloj. A esta hora la placa girará y se colocará en la posición adecuada para recibir los rayos solares de forma eficaz a las 12:00 hora solar.
2. Una variable (¿un contador? ¿un potenciómetro?) almacenará el valor del día juliano. Los cálculos se referirán a este valor siempre.
3. Todos los días la placa se coloca en su posición siguiendo la misma rutina:
 - a. Señal de pulso del “reloj”. Más información se puede encontrar en la web: <https://www.luisllamas.es/reloj-y-calendario-en-arduino-con-los-rtc-ds1307-y-ds3231/>
 - b. Busca la posición de ángulo 0º girando en sentido contrario (horizontal, paralelo al suelo). No lo calcula, sino que un final de carrera (simularlo) le da la información a Arduino de que ha llegado al cero (autocalibración).
 - c. Una vez calibrado, gira el ángulo calculado para ese día.