

UNIVERSIDAD PÚBLICA DE EL ALTO

INGENIERIA DE SISTEMAS



ROBOTICA INDUSTRIAL

LABORATORIO 3

CONTROL DE MOTORES CON L298N

Universitario: WILLY MARCOS CHANA TITO

Carrera: INGENIERIA DE SISTEMAS

Docente: ING. ELIAS ALI ALVAREZ

Materia: SISTEMAS DIGITALES

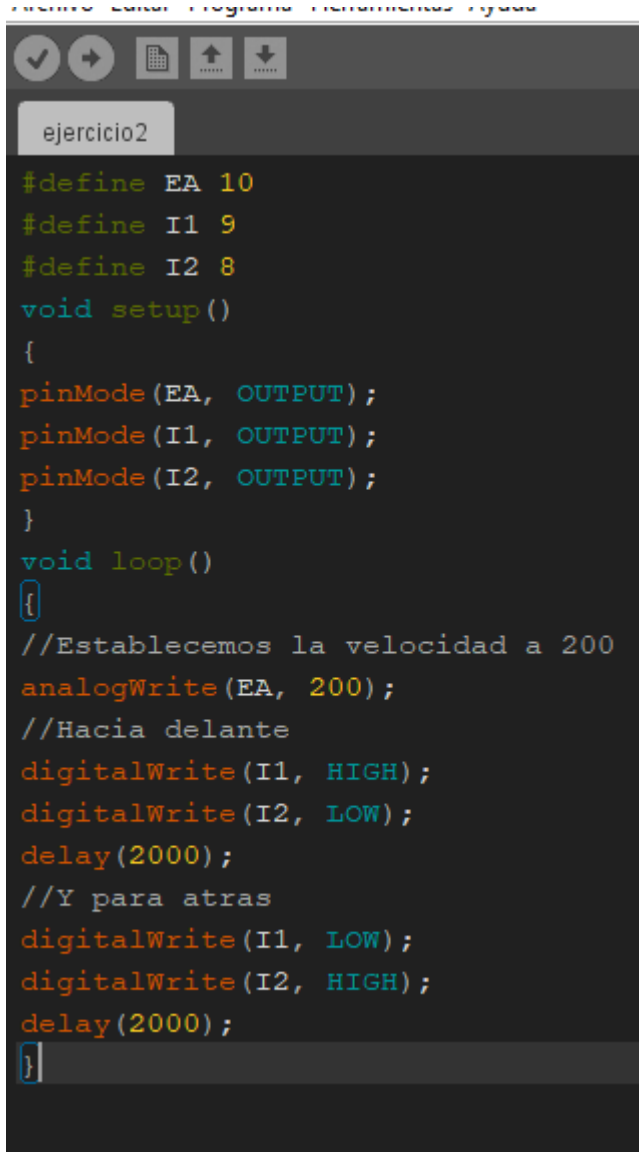
Paralelo: 9° C

Fecha: 31/03/2021

INFORME

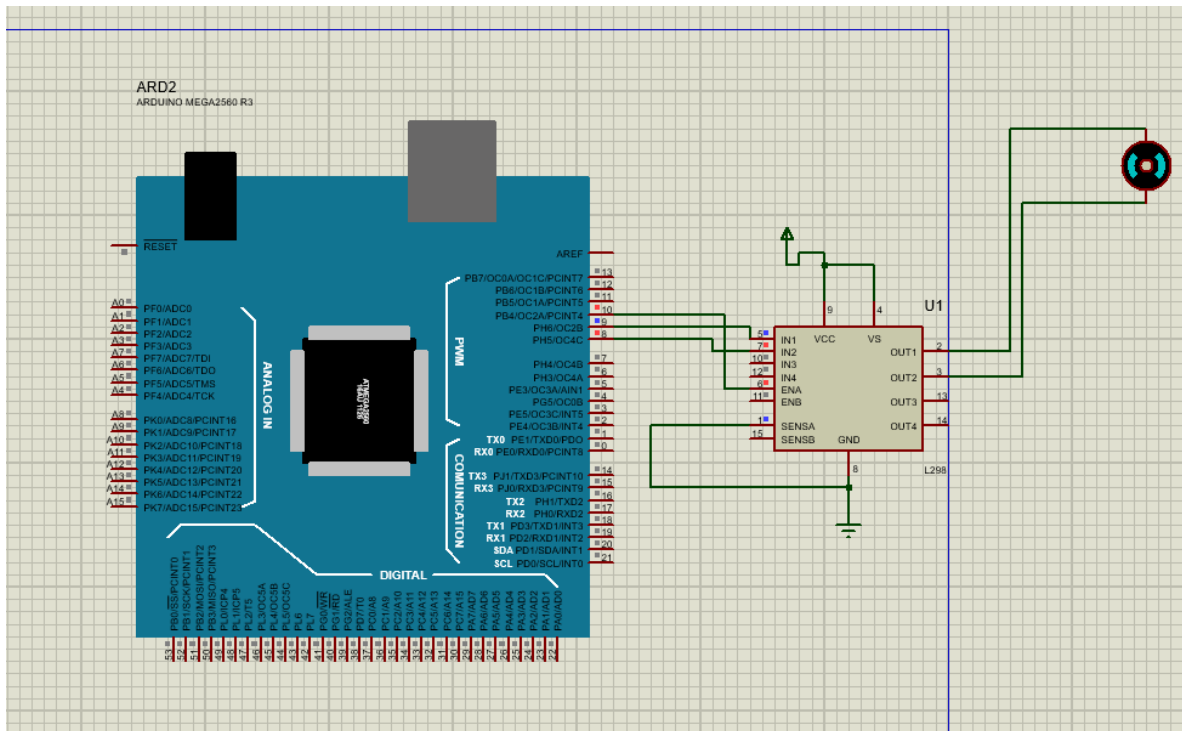
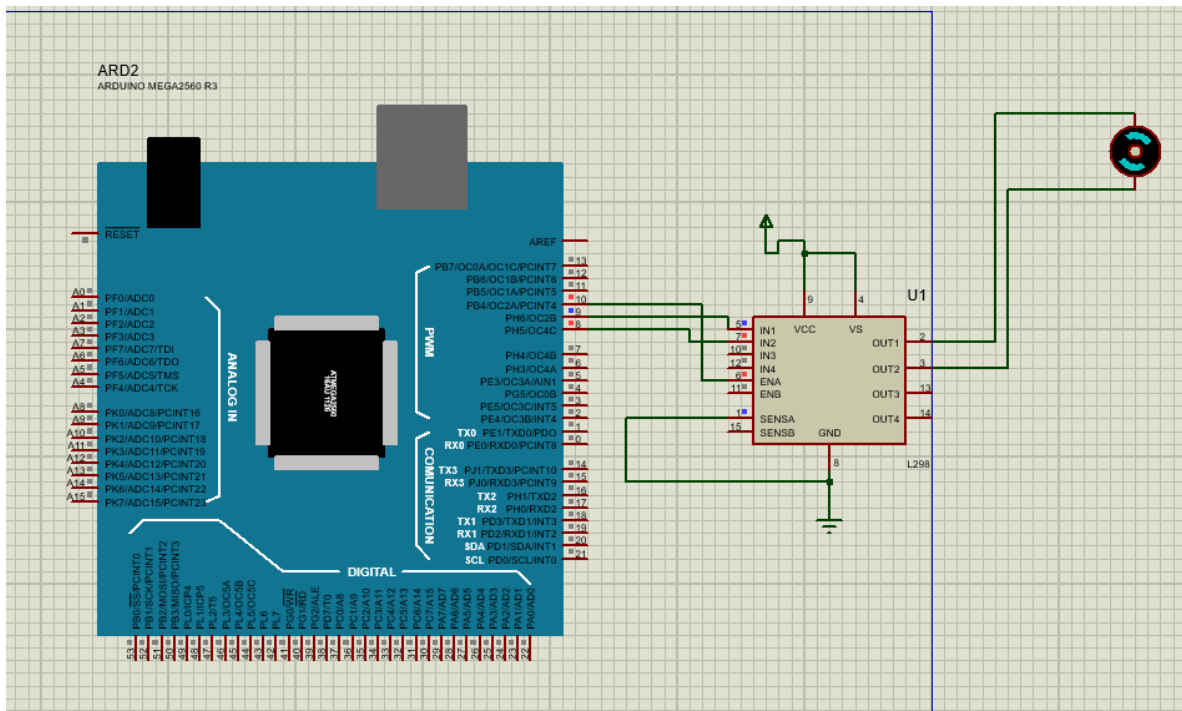
1. Anexe a su informe los resultados obtenidos del punto VII.

Ejercicio 1.

A screenshot of the Arduino IDE interface. The top toolbar shows icons for checking, running, saving, and uploading. Below the toolbar, a tab labeled 'ejercicio2' is active. The code editor displays the following C++ code:

```
#define EA 10
#define I1 9
#define I2 8
void setup()
{
  pinMode(EA, OUTPUT);
  pinMode(I1, OUTPUT);
  pinMode(I2, OUTPUT);
}
void loop()
{
  //Establecemos la velocidad a 200
  analogWrite(EA, 200);
  //Hacia delante
  digitalWrite(I1, HIGH);
  digitalWrite(I2, LOW);
  delay(2000);
  //Y para atras
  digitalWrite(I1, LOW);
  digitalWrite(I2, HIGH);
  delay(2000);
}
```

```
#define EA 10
#define I1 9
#define I2 8
void setup()
{
  pinMode(EA, OUTPUT);
  pinMode(I1, OUTPUT);
  pinMode(I2, OUTPUT);
}
void loop()
{
  //Establecemos la velocidad a
  200
  analogWrite(EA, 200);
  //Hacia delante
  digitalWrite(I1, HIGH);
  digitalWrite(I2, LOW);
  delay(2000);
  //Y para atras
  digitalWrite(I1, LOW);
  digitalWrite(I2, HIGH);
  delay(2000);
}
```



Estudiante: Willy Marcos Chana Tito

Ejercicio 2.

```
#define EA 10
#define EB 11
#define I1 9
#define I2 8
#define I3 6
#define I4 7

void adelante(int motor, int velocidad){
    if(motor == 1){
        analogWrite(EA, velocidad);
        digitalWrite(I1, HIGH);
        digitalWrite(I2, LOW);
    }else{
        analogWrite(EB, velocidad);
        digitalWrite(I3, HIGH);
        digitalWrite(I4, LOW);
    }
}

void atras(int motor, int velocidad){
    if(motor == 1){
        analogWrite(EA, velocidad);
        digitalWrite(I1, LOW);
        digitalWrite(I2, HIGH);
    }else{
        analogWrite(EB, velocidad);
        digitalWrite(I3, LOW);
        digitalWrite(I4, HIGH);
    }
}

void parar(int motor){
    if(motor == 1){
        digitalWrite(I1, LOW);
        digitalWrite(I2, LOW);
    }else{
        digitalWrite(I3, LOW);
        digitalWrite(I4, LOW);
    }
}

void setup(){
    pinMode(EA, OUTPUT);
    pinMode(I1, OUTPUT);
    pinMode(I2, OUTPUT);
    pinMode(EB, OUTPUT);
    pinMode(I3, OUTPUT);
    pinMode(I4, OUTPUT);
}
```

```
void loop(){
    adelante(1, 100);
    delay(1000);
    atras(1, 100);
    delay(1000);
    parar(1);
    delay(1000);

    adelante(2, 100);
    delay(1000);
    atras(2, 100);
    delay(1000);
    parar(2);
    delay(1000);
    adelante(1, 100);
    adelante(2, 100);
    delay(1000);
    parar(1);
    parar(2);
    delay(1000);
    atras(1, 100);
    atras(2, 100);
    delay(1000);
    parar(1);
    parar(2);
    delay(1000);
}
```

```

#define EA 10
#define EB 11
#define I1 9
#define I2 8
#define I3 6
#define I4 7
void adelante(int motor, int
velocidad){
    if(motor == 1){
        analogWrite(EA, velocidad);
        digitalWrite(I1, HIGH);
        digitalWrite(I2, LOW);
    }else{
        analogWrite(EB, velocidad);
        digitalWrite(I3, HIGH);
        digitalWrite(I4, LOW);
    }
}
void atras(int motor, int
velocidad){
    if(motor == 1){
        analogWrite(EA, velocidad);
        digitalWrite(I1, LOW);
        digitalWrite(I2, HIGH);
    }else{
        analogWrite(EB, velocidad);
        digitalWrite(I3, LOW);
        digitalWrite(I4, HIGH);
    }
}
void parar(int motor){
    if(motor == 1){
        digitalWrite(I1, LOW);
        digitalWrite(I2, LOW);
    }else{
        digitalWrite(I3, LOW);
        digitalWrite(I4, LOW);
    }
}
}

```

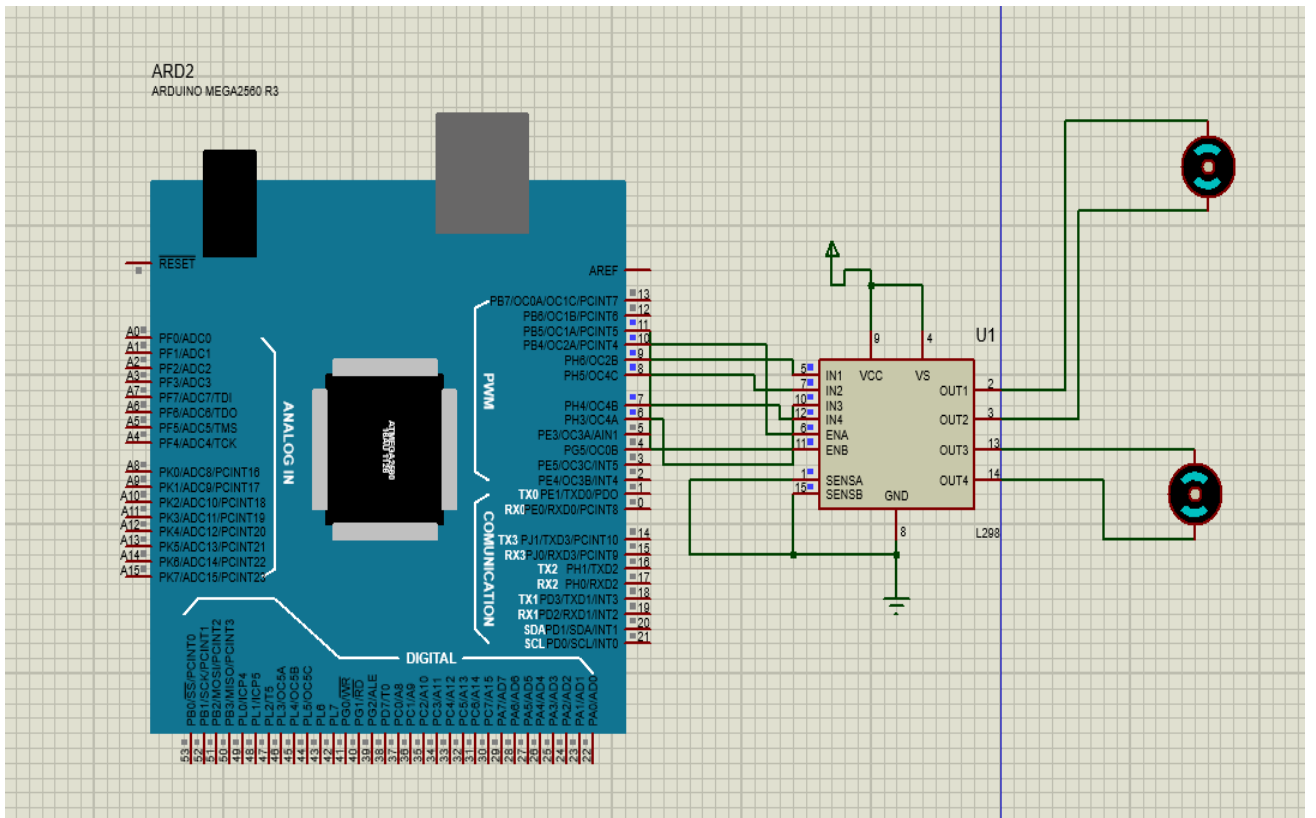
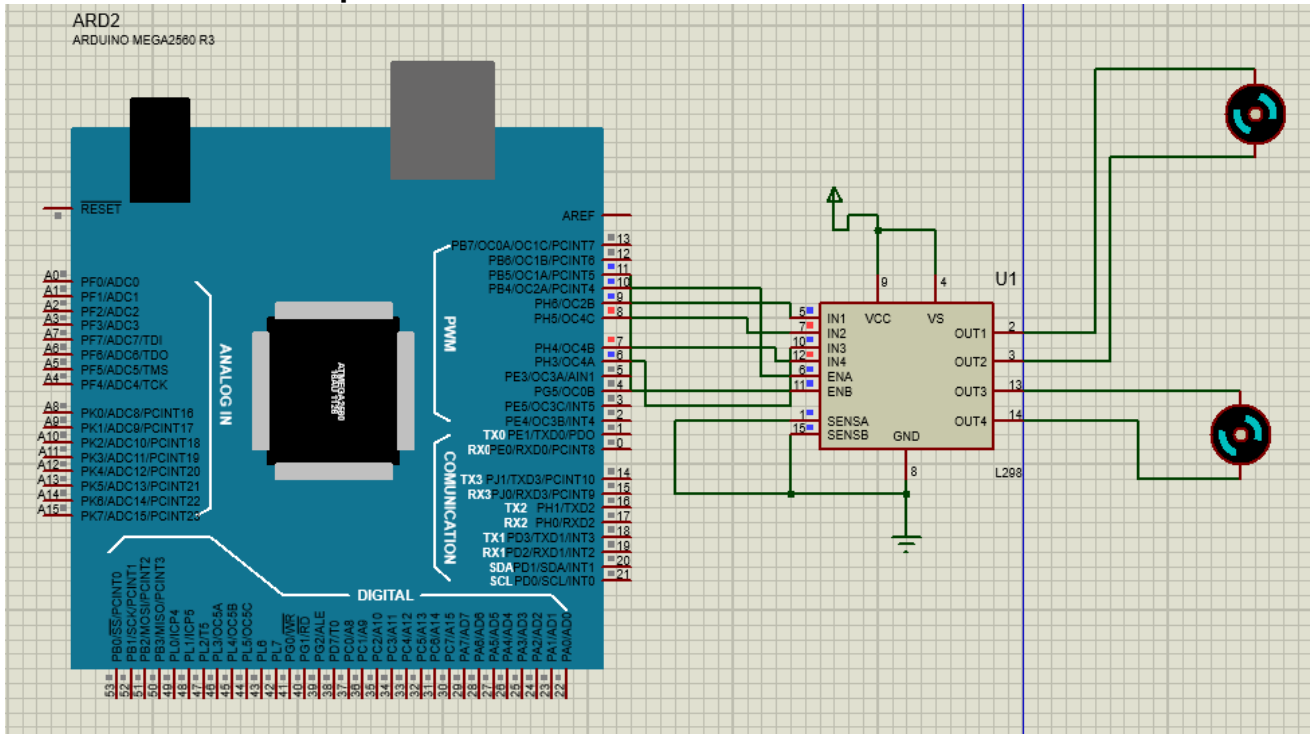
```

void setup(){
    pinMode(EA, OUTPUT);
    pinMode(I1, OUTPUT);
    pinMode(I2, OUTPUT);
    pinMode(EB, OUTPUT);
    pinMode(I3, OUTPUT);
    pinMode(I4, OUTPUT);
}
void loop(){
    adelante(1, 100);
    delay(1000);
    atras(1, 100);
    delay(1000);
    parar(1);
    delay(1000);

    adelante(2, 100);
    delay(1000);
    atras(2, 100);
    delay(1000);
    parar(2);
    delay(1000);
    adelante(1, 100);
    adelante(2, 100);
    delay(1000);
    parar(1);
    parar(2);
    delay(1000);
    atras(1, 100);
    atras(2, 100);
    delay(1000);
    parar(1);
    parar(2);
    delay(1000);
}

```

Simulación en protheus



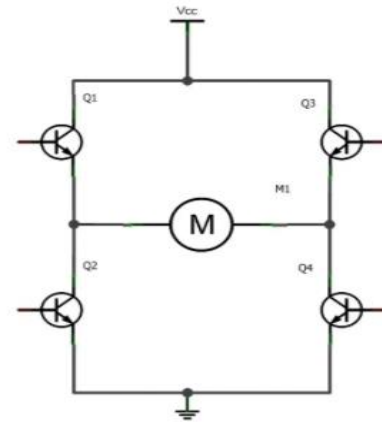
Estudiante: Willy Marcos Chana Tito

2. Investigue si existen otros Drivers controladores de motores

Driver L293D

El L293D es un integrado para controlar motores DC que usa doble puente en H. Es un sistema para controlar el sentido de giro de un motor DC usando cuatro transistores y también para variar la velocidad del motor. En la imagen vemos que los transistores se comportan como interruptores y dependiendo que transistores conducen y cuáles no cambia la polarización del motor y con esto el sentido de giro.

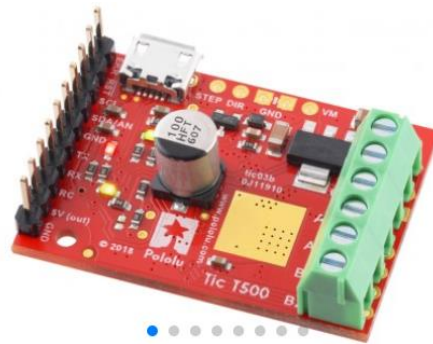
- **modelo DCM-0010** Para el control de motor Dc 9 Amperios



- **DCM-0045** Controlador de motores inteligente G2 18v15 / 15A



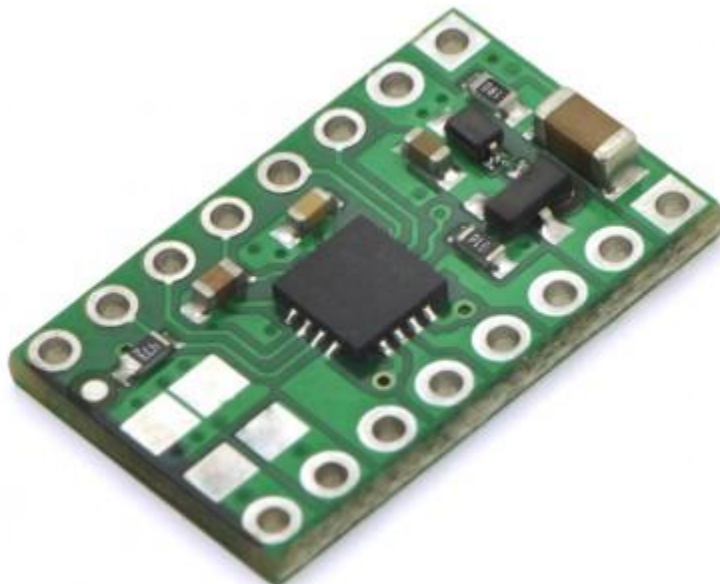
- **MOT-0027** Controlador de motores PaP Tic T500 USB



- **Driver motor paso a paso STSPIN820**, Ref: MOT-0026, Driver de motores paso a paso silenciosos de Pololu STSPIN820 con 256 micropasos.



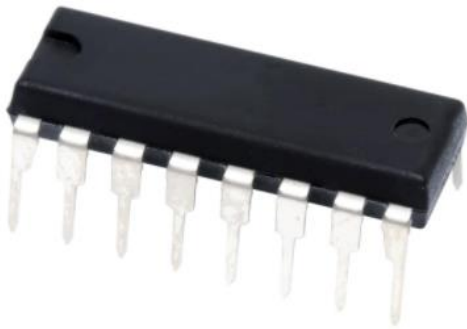
- **Controlador motor DC dual DRV8833**



t

Estudiante: Willy Marcos Chana Tito

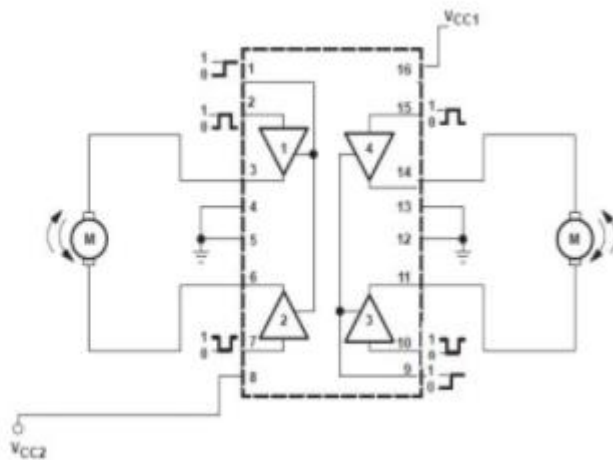
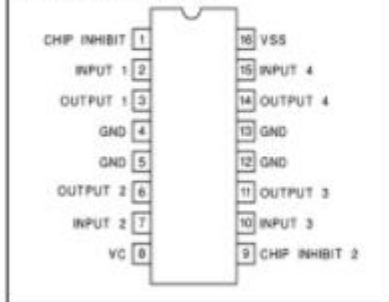
3. Describa el circuito integrado del L293N



CONNECTION DIAGRAMS

DIL-16 (TOP VIEW)

N Package, SP Package

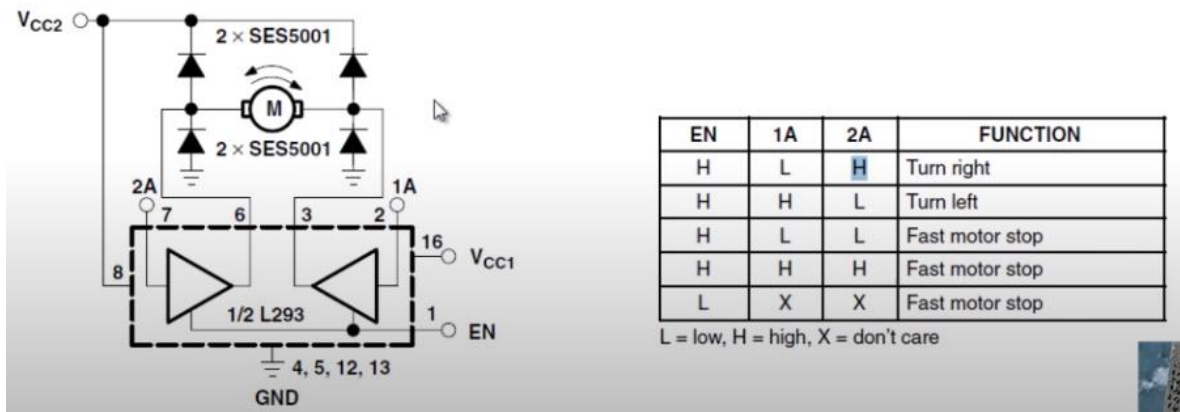


El circuito integrado L293 permite controlar motores DC de pequeña potencia. Para utilizarlo hay que hacer un montaje externo a Arduino, en una placa de pruebas, y alimentar a los motores a través de este circuito integrado.

El CI L293 tiene las siguientes características:

- Se pueden controlar hasta 2 motores.
- Proporciona 1A a los motores (en total) y permite cambiar el sentido de giro.
- Utiliza un puente en H que funciona según se observa en las figuras (internamente utiliza transistores para conmutar*)

4.Describa como controlar un motor paso paso con el Driver L293N



Este es el circuito del driver L293N el cual sirve para controlar motores paso paso

Según el diagrama y la tabla nos dice que si en 1A es Low y 2A es High entonces el motor jira a la derecha, luego si 1A High y 2A Low entonces el motor jira a la izquierda, luego si ambos son Low o High entonces el motor se detiene.

5.Mencione algunas aplicaciones de este laboratorio

Este laboratorio puede ser aplicada en la robótica, podemos controlar el movimiento de las ruedas y extremidades.

También los podemos utilizar este laboratorio para maquinarias y ejes eléctricos, como el control de un ascensor o grúas.

6.Conclusiones y observaciones

Conclusión

En este laboratorio se pudo comprobar que Arduino con la ayuda de algunos drivers puede controlar los motores con mucha facilidad lo que facilita su uso en diferentes proyectos.

Observación

Al momento de compilar el sketch se debe seleccionar el Arduino de trabajo, en mi caso seleccione Arduino mega 2560 pero lo compile en Arduino uno, esto hace que el programa compilado no funcione adecuadamente.