



Tecnológico Nacional de México  
Instituto Tecnológico Iztapalapa 1

Carrera: Ingeniería Mecatrónica  
Materia: Arquitectura de los Sistemas  
Embebidos

Grupo: IME-7AM  
Robot Sumo

Nombre: Emanuel Hernández Montes  
N.Control: 211080046  
Profesor: Soria Frias Sigfrido Oscar

Fecha de entrega: 14/05/24

# Introducción

En el ámbito de la ingeniería y la robótica, la competencia de Sumo Robot ha ido ganando terreno como una emocionante prueba que combina habilidades técnicas y creatividad en el diseño. Inspirados en el ancestral deporte japonés de sumo, estos robots son concebidos para enfrentarse en un ring, utilizando la fuerza para expulsar a su contrincante fuera del área de combate. La creación de un Sumo Robot demanda no solo conocimientos en ingeniería mecánica y eléctrica, sino también destrezas en programación y estrategia.

En el presente informe, vamos a explorar exhaustivamente el proceso completo de fabricación de un Sumo Robot, desde la concepción de la idea hasta la fase final de construcción y programación.

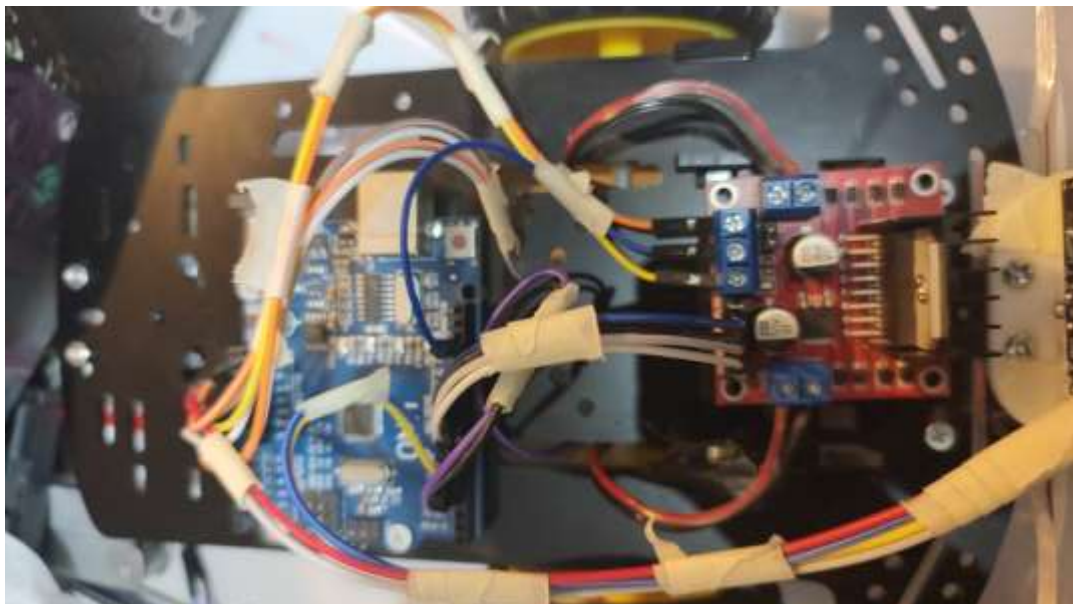
# Desarrollo

## Materiales para utilizar:

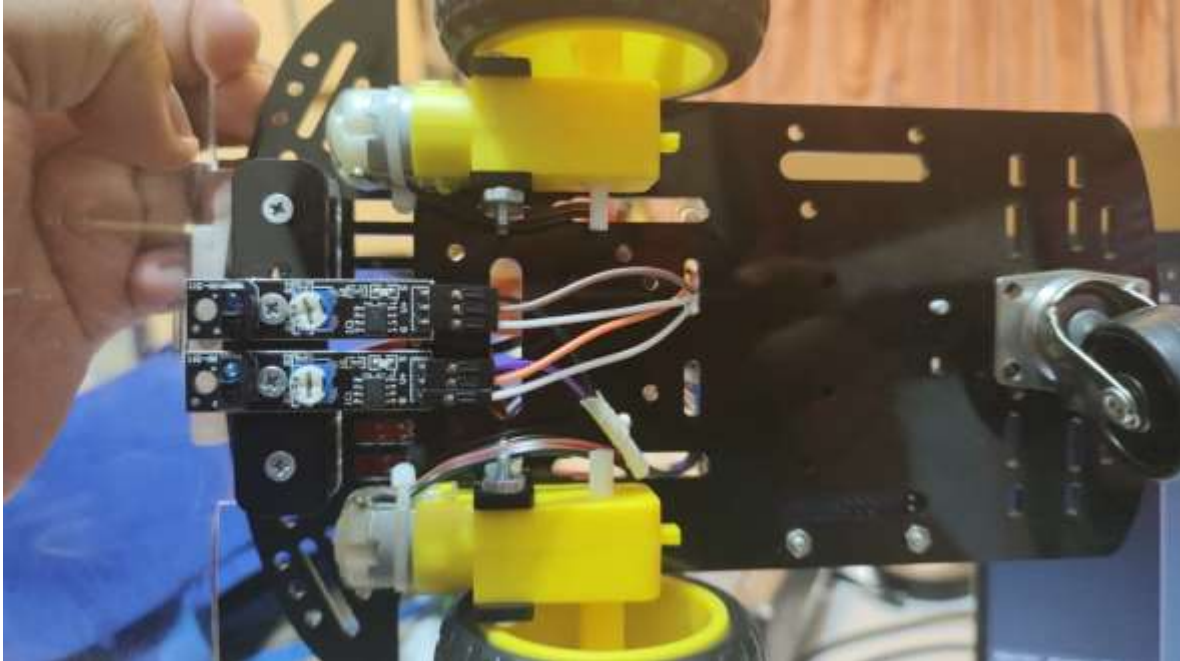
- Driver L298N
- Arduino Uno
- Jumpers macho, macho
- Jumpers macho, hembra
- 2 Motores 12v DC con juego de engranajes
- Tecneu Chasis de carro
- 2 Llantas para motorreductor 67x25 mm
- Sensor ultrasónico
- Tornillos con tuerca 3mm
- Hoja de navaja de plástico 15x5 cm

## Armado

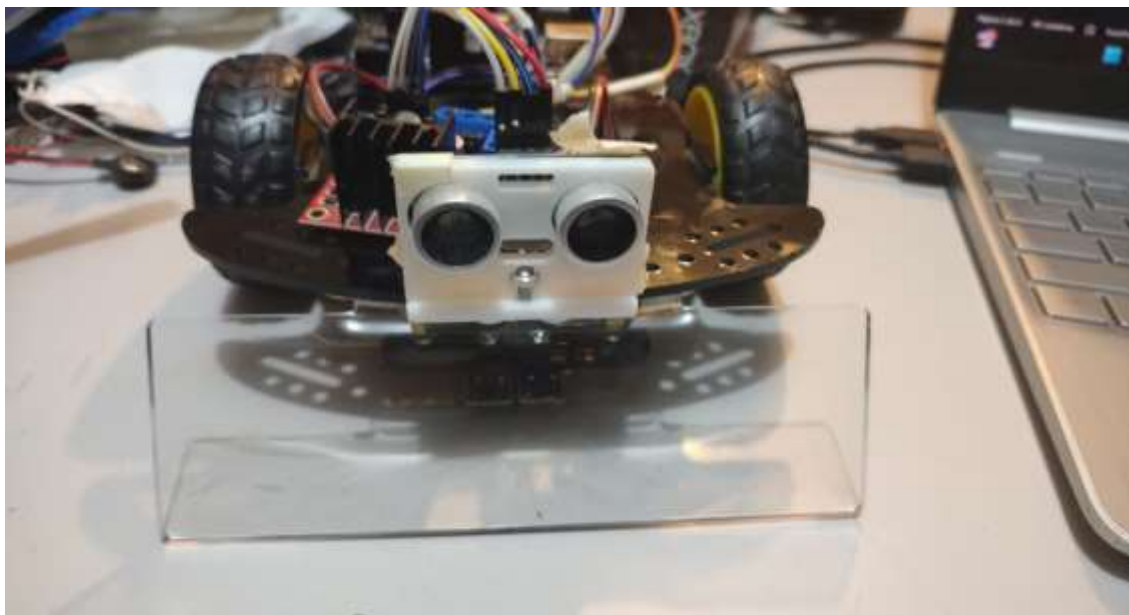
Se colocará en la parte de arriba del chasis lo que es el Arduino UNO, con el driver L298n y el sensor ultrasónico en su base todo sujetado respectivamente con los tornillos de 3mm.



Por la parte de abajo se pondrá los dos motores DC cada sujeto con su rueda para motorreductor e igualmente sujeto con tornillo y tuerca todo va sujeto con eso al igual.

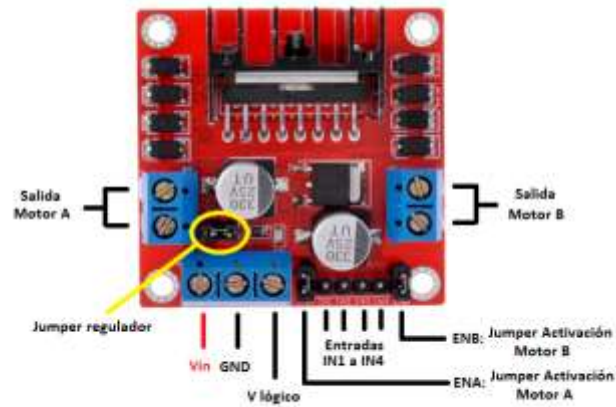


Se pondrá en la parte de enfrente abajo del ultrasónico la hoja de navaja de plástico para poder empujar al contrincante



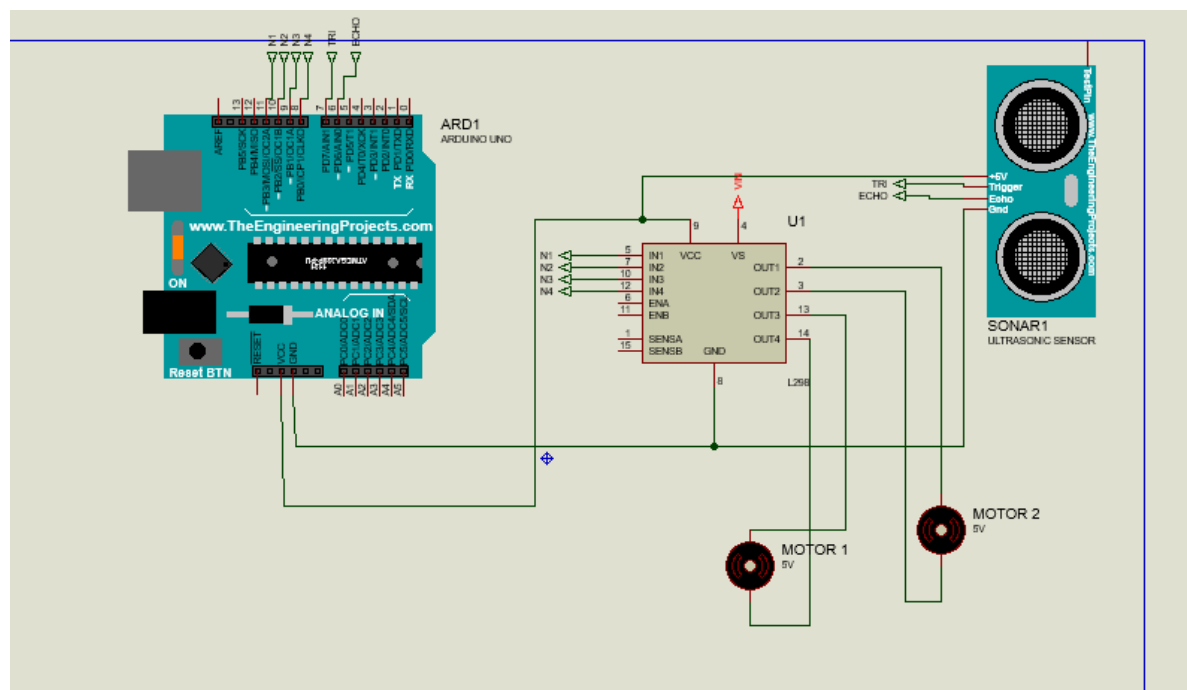
## Conexiones

Para conectar todo antes se investigo previamente los diagramas de las cosas a utilizar lo que es el diagrama de el driver L298n que es el siguiente



Y sucesivamente el diagrama del sensor ultrasónico para poder identificar los pines y donde colocar cada uno y su posición







# Código

En el código se empezó con estar practicando primeramente como funcionaba los motores con la librería que se utilizo llamada <AFMotor.h> que la encontré en un repositorio de GitHub el funcionamiento con los motores, pero en realidad la librería a descargar desde la IDE de Arduino se llama Adafruit Motor shield library y es para utilizar conjuntamente con el driver L298n.



```
1 #include <AFMotor.h>
2
3 //pines para los motores (IN1, IN2, IN3, IN4)
4 #define IN1 11
5 #define IN2 10
6 #define IN3 9
7 #define IN4 8
8
9 // pines sensor ultrasónico
10 #define trigPin 7
11 #define echoPin 6
12
13 // distancia max o limite (en centimetros)
14 #define distanciaLimite 20
15
16 // motores nombra con el driver L298N
17 AF_Motor motor1(11); // Motor izquierdo

Error descargando https://resource.heltec.cn/download/package_heltec_esp32_index.json
```

E igualmente se colocaron los pines elegidos para el driver y las señales del sensor ultrasónico , agarrando como pines para el sensor el trig en el pin 7 y el echo en el pin 6 lo demás que quedaron fue el pin 11 el N1 pin 10 el N2 pin 9 el N3 y por ultimo el pin 8 el N4 quedando el código de la siguiente manera.

```
#include <AFMotor.h>
```

```
//pines para los motores (IN1, IN2, IN3, IN4)
```

```
#define IN1 11
```

```
#define IN2 10
```

```
#define IN3 9
```

```
#define IN4 8

// pines sensor ultrasónico
#define trigPin 7
#define echoPin 6

// distancia max o limite (en centímetros)
#define distanciaLimite 20

// motores nombra con el driver L298N
AF_DCMotor motor1(1); // Motor izquierdo
AF_DCMotor motor2(2); // Motor derecho

void setup() {
  // pines de control como salidas
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);

  //pines del sensor ultrasónico
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  // motores
  motor1.setSpeed(255); // Velocidad máxima
  motor2.setSpeed(255); // Velocidad máxima
```



```

// Inicializa la comunicación serial
Serial.begin(9600);
}

void loop() {
    // Genera un pulso corto en el pin de trigger del sensor ultrasónico
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    // duración del pulso de eco del sensor ultrasónico
    long duracion = pulseIn(echoPin, HIGH);

    // calcula la distancia en centímetros
    int distancia = duracion * 0.034 / 2;

    // muestra la distancia en el monitor serial
    Serial.print("Distancia: ");
    Serial.println(distancia);

    // comprobar si hay un objeto a distancia
    if (distancia < distanciaLimite) {
        // Imprimir atacando
        Serial.println("Atacando");

        //avanzar

```

```

digitalWrite(IN1, HIGH);
digitalWrite(IN2, LOW);
digitalWrite(IN3, HIGH);
digitalWrite(IN4, LOW);
} else {
    // indicar si no hay objeto gira
    Serial.println("Girando a la derecha");

    // si no hay objetivos (enemigos), gira a la derecha
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
}

// tiempo de lectura del monitor serial
delay(100);
}

```

Como se logra apreciar utilice el serial y algunos print donde en el monitor nos mostrara el momento donde el sumo busca objetivo y girara en caso de que detecte un objetivo saldrá atacando y el sumo se dirigirá a ese objeto encontrado por el sensor ultrasónico previamente modificado la distancia en el mismo código que esta a 20 cm.

También se agrego en los print que te de en cm la distancia que marca el sensor ultrasónico y a que distancia detecta su objetivo

Eso sería todo igualmente ahí están los parámetros en el mismo código para personalizar la distancia de ataque y configurar la velocidad de los motores.

# Conclusión

En síntesis, desarrollar un Sumo Robot implica dominar diversas disciplinas como ingeniería, programación y táctica. A lo largo de este informe, hemos examinado los pilares esenciales de este proceso, desde la concepción inicial hasta la fase final de construcción y programación. Hemos resaltado la relevancia de comprender los desafíos técnicos y aplicar estrategias de diseño para mejorar el rendimiento del robot en el combate. Con el progreso tecnológico continuo y la inventiva de los diseñadores, el ámbito de los Sumo Robots seguirá siendo un campo dinámico y emocionante dentro de la robótica competitiva.

# Referencias

1.- APA = Johnmcombs. (s. f.). arduino-libraries/AFMotor/AFMotor.h at master · johnmcombs/arduino-libraries. GitHub. <https://github.com/johnmcombs/arduino-libraries/blob/master/AFMotor/AFMotor.h>

2.- APA = DANIEL PUERTA TABORDA. (2021, 15 abril). robot sumo [Video]. YouTube. <https://www.youtube.com/watch?v=EMPQvHfgQhQ>

3.- [https://naylampmechatronics.com/blog/11\\_tutorial-de-uso-del-modulo-l298n.html](https://naylampmechatronics.com/blog/11_tutorial-de-uso-del-modulo-l298n.html)