

UNIVERSIDAD NACIONAL AUTONOMA DE  
MEXICO

Proyecto:

Auto esquivando obstáculos con Arduino Uno

Asignatura: Estructura de Datos y algoritmos I

Alumna: Tierrablanca Oviedo Evelyn

## **INTRODUCCION:**

El proyecto está basada en la construcción de un carro esquivo obstáculos programado en arduino uno del canal Mert Arduino.

## **MATERIALES DEL PROYECTO:**

- Arduino car kit 4x4
- Arduino uno
- Servo motor SG90
- Arduino sensor Shield V5.0
- Sensor ultrasónico
- L298N
- 4 Motores reductores
- Switch de palanca deslizante
- Jumpers macho-hembra, hembra-hembra
- Cable UTP
- Batería de 9V
- 4 Baterías de 3.5V

## PROGRAMA DEL ARDUINO PARA EL PROYECTO:

```
#include <Servo.h>
#include <NewPing.h>

//programa para nuestro L298N control pins
const int LeftMotorForward = 5;
const int LeftMotorBackward = 4;
const int RightMotorForward = 3;
const int RightMotorBackward = 2;

// programacion para los pins del sensor
#define trig_pin A1
#define echo_pin A2

#define maximum_distance 200
boolean goesForward = false;
int distance = 100;

NewPing sonar(trig_pin, echo_pin, maximum_distance); //funtion del sensor
Servo servo_motor; //el nombre de nuestro servo

void setup(){

  pinMode(RightMotorForward, OUTPUT);
  pinMode(LeftMotorForward, OUTPUT);
  pinMode(LeftMotorBackward, OUTPUT);
  pinMode(RightMotorBackward, OUTPUT);

  servo_motor.attach(11); //nuestro pin del servo
}

servo_motor.write(90);
delay(2000);
distance = readPing();
delay(100);
distance = readPing();
delay(100);
distance = readPing();
delay(100);
distance = readPing();
delay(100);
distance = readPing();
delay(100);

}

void loop(){

  int distanceRight = 0;
  int distanceLeft = 0;
  delay(50);

  if (distance <= 20){
    moveStop();
    delay(300);
    moveBackward();
    delay(400);
    moveStop();
    delay(300);
    distanceRight = lookRight();
    delay(300);
    distanceLeft = lookLeft();
    delay(300);
```

---

```
    if (distance >= distanceLeft){
        turnRight();
        moveStop();
    }
    else{
        turnLeft();
        moveStop();
    }
}
else{
    moveForward();
}
distance = readPing();
}
```

```
int lookRight(){
    servo_motor.write(10);
    delay(500);
    int distance = readPing();
    delay(100);
    servo_motor.write(90);
    return distance;
}
```

```
int lookLeft(){
    servo_motor.write(170);
    delay(500);
    int distance = readPing();
    delay(100);
}
```

---

```
    }

int readPing() {
    delay(70);
    int cm = sonar.ping_cm();
    if (cm==0) {
        cm=250;
    }
    return cm;
}

void moveStop() {

    digitalWrite(RightMotorForward, LOW);
    digitalWrite(LeftMotorForward, LOW);
    digitalWrite(RightMotorBackward, LOW);
    digitalWrite(LeftMotorBackward, LOW);
}

void moveForward() {

    if (!goesForward) {

        goesForward=true;

        digitalWrite(LeftMotorForward, HIGH);
        digitalWrite(RightMotorForward, HIGH);

        digitalWrite(LeftMotorBackward, LOW);
```

```
        digitalWrite(RightMotorBackward, LOW);
    }
}
```

```
void moveBackward() {
```

```
    goesForward=false;
```

```
    digitalWrite(LeftMotorBackward, HIGH);
    digitalWrite(RightMotorBackward, HIGH);
```

```
    digitalWrite(LeftMotorForward, LOW);
    digitalWrite(RightMotorForward, LOW);
```

```
}
```

```
void turnRight() {
```

```
    digitalWrite(LeftMotorForward, HIGH);
    digitalWrite(RightMotorBackward, HIGH);
```

```
    digitalWrite(LeftMotorBackward, LOW);
    digitalWrite(RightMotorForward, LOW);
```

```
    delay(500);
```

```
    digitalWrite(LeftMotorForward, HIGH);
    digitalWrite(RightMotorForward, HIGH);
```

---

```
    digitalWrite(LeftMotorBackward, LOW);
    digitalWrite(RightMotorBackward, LOW);
```

```
}
```

```
void turnLeft() {
```

```
    digitalWrite(LeftMotorBackward, HIGH);
    digitalWrite(RightMotorForward, HIGH);
```

```
    digitalWrite(LeftMotorForward, LOW);
    digitalWrite(RightMotorBackward, LOW);
```

```
    delay(500);
```

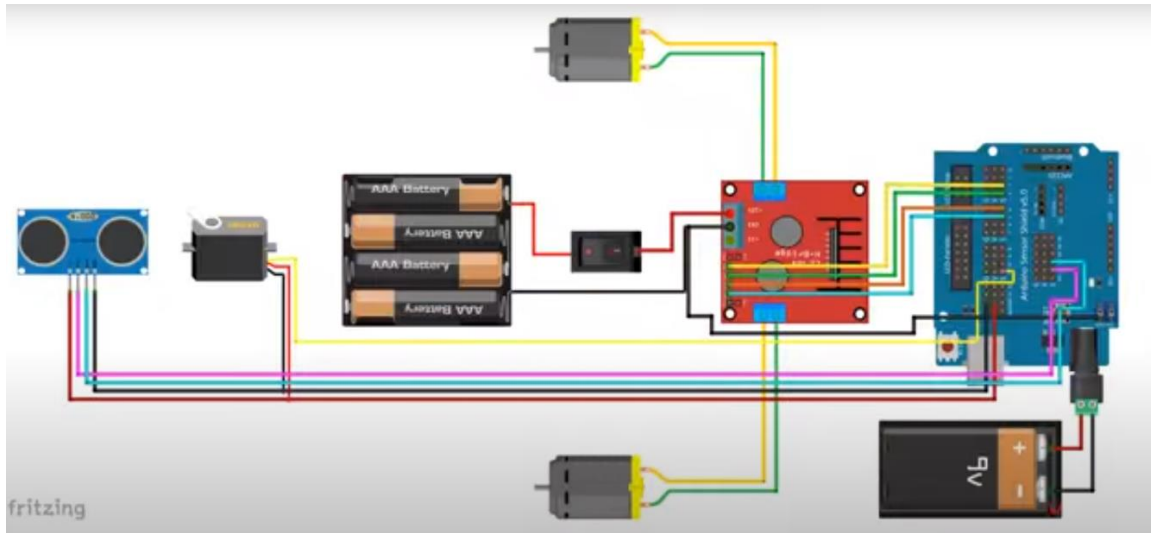
```
    digitalWrite(LeftMotorForward, HIGH);
    digitalWrite(RightMotorForward, HIGH);
```

```
    digitalWrite(LeftMotorBackward, LOW);
    digitalWrite(RightMotorBackward, LOW);
```

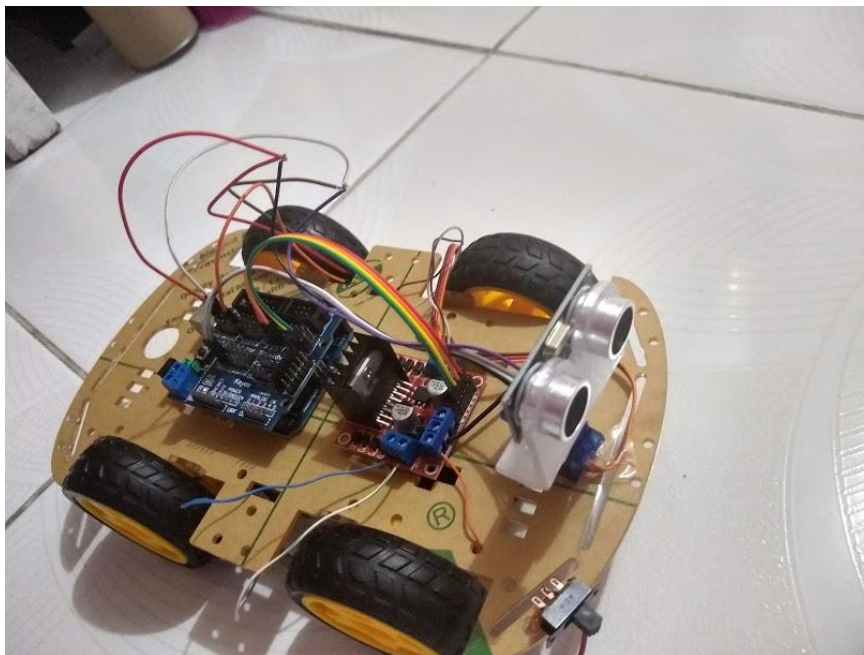
```
}
```

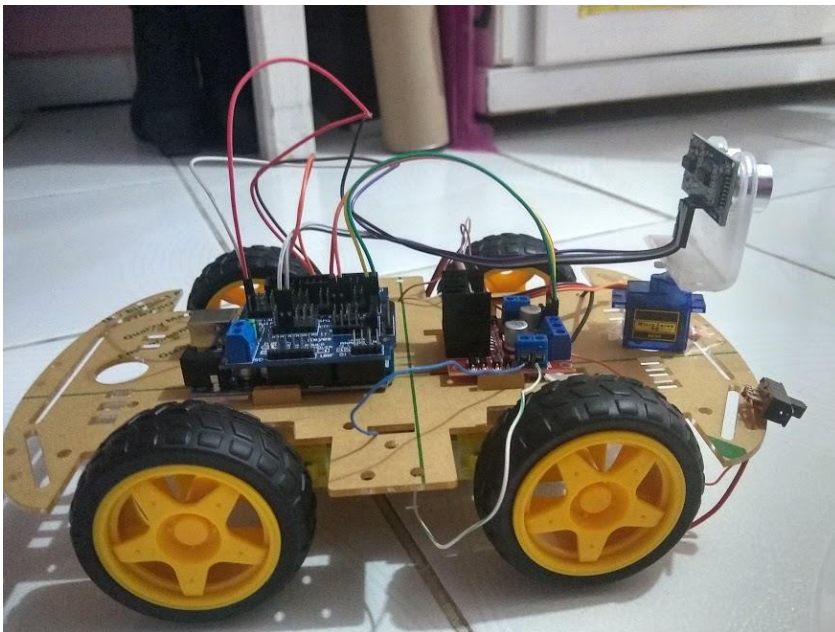
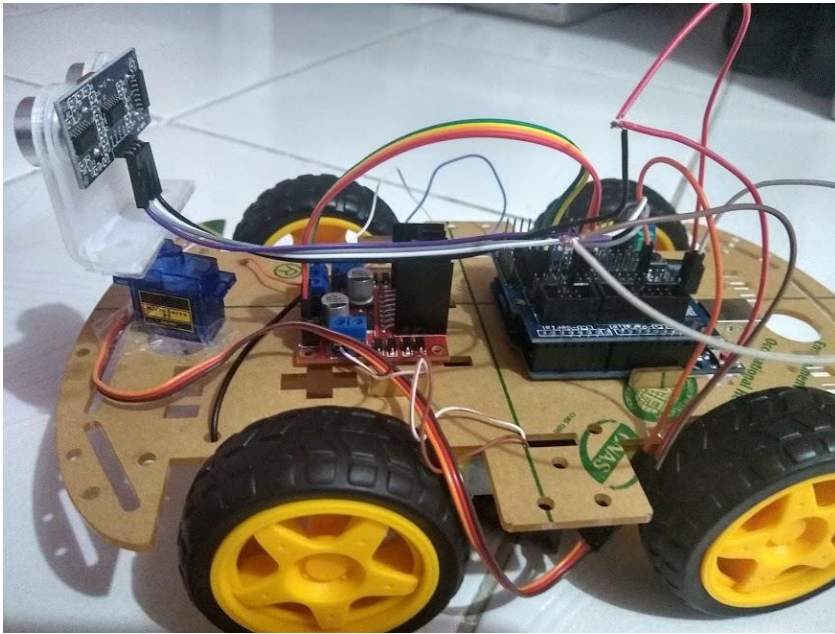
---

## DIAGRAMA ELECTRICO PARA EL PROYECTO:



## PROYECTO ARMADO:





## CONCLUSIONES

El proyecto salió armable, pero no con los resultados esperados, debido a que no giraba de dirección al momento de que el sensor encontraba libre el paso, esto se debió a las dos ruedas traseras del carro.



