



**Estudiantes:**

Joan Steven Gómez Contreras-----USIS014921  
Julio Alexander Maravilla Umaña-----USIS018420  
Katerin Estela Orellana Hernández----- USIS035821  
Dina Esperanza Gonzales Reyes----- USIS055521  
Roberto Francisco Reyes Palomo-----USIS032421

**Docente:**

Ing. Marvin Parada

**Asignatura:**

Robotica

## Análisis de requerimientos de hardware

### **Materiales:**

#### **Arduino UNO:**

Para insertare la programación dentro del carrito.

#### **Controlador L293D:**

Controlar los motores, servos y el sensor ultrasónico.

#### **Sensor Ultrasónico:**

Identificar obstáculos más cercanos.

#### **Servomotor:**

Utilizado para el movimiento del ultrasónico.

#### **Rueda Loca:**

Direccionamiento del carrito.

#### **Chasis para ensamblado:**

Para poder colocar los materiales necesarios del carrito.

#### **Cables:**

Conexión de los componentes entre sí.

#### **Protoboard:**

Para prototipado y conexión temporal de componentes electrónicos.

#### **Ruedas amarillas:**

Movimiento del carrito.

#### **Alimentación:**

Por medio de una batería de 9V

## Análisis de requerimiento de software

**Integración con Tinkercad:** Herramienta enfocada en la creación de animaciones virtuales, la cual es compatible con Arduino para poder llevar a cabo los proyecto probados dentro de ella.

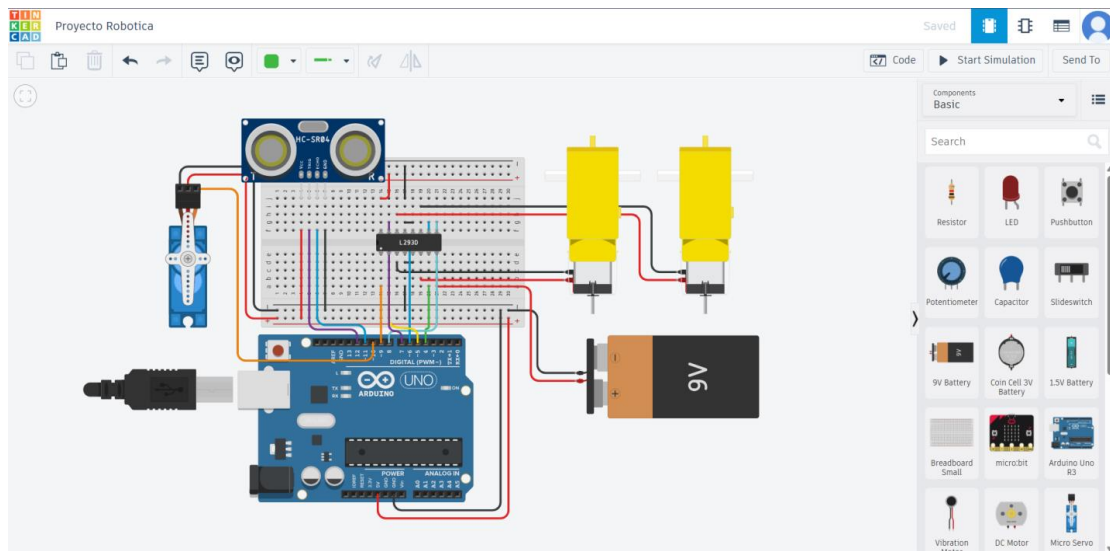
**Programa Arduino IDE:** Utilizado para poder integrar dentro de la placa Arduino todo el código necesario para el funcionamiento de nuestro carrito.

**Lectura de sensor ultrasónico:** Codificar la lectura del sensor ultrasónico para detectar obstáculos.

**Algoritmo de navegación:** Implementar un algoritmo para evitar obstáculos y navegar de manera autónoma.

## Diseño Físico

## Diseño lógico



Proyecto en Tinkercad:

<https://www.tinkercad.com/things/fELRUgDrHyB-proyecto-robotica/editel?returnTo=%2Fdashboard%3Fcollection%3Ddesigns%26type%3Dcircuits&sharecode=rZmeoEclzPZccQuLfvaCzjA0Bh1buQMTdgHtPPi-yNU>

## Herramientas utilizadas en la manufacturación del robot

**Base:** Utilizada para poder colocar los elementos de carrito.

**Pernos:** para poder anclar los materiales a la base.

**Desarmador de cruz:** Utilizado para atornillar la rueda loca, los motores y las placas en la base.

## Herramientas y entorno de programación

- **Tinkercad:** Para hacer el prototipo de nuestro carrito.
- **Arduino:** Necesario para la programación del carrito.

## Programacion

```
/*  
PROYECTO ROBOTICA  
Joan Steven Gomez Contreras-----USIS014921  
Julio Alexander Maravilla Umaña-----USIS018420  
Katerin Estela Orellana Hernandez----USIS035821  
Dina Esperanza Gonzales Reyes-----USIS055521  
Roberto Fransisco Reyes Palomo-----USIS032421  
*/
```

```

#include "AFMotor.h"
#include <Servo.h>

#define echopin A4 // echo pin
#define trigpin A5 // Trigger pin

Servo myservo;

const int MOTOR_1 = 1;
//const int MOTOR_2 = 2;
//const int MOTOR_3 = 3;
const int MOTOR_4 = 4;

AF_DCMotor motor1(MOTOR_1, MOTOR12_64KHZ); // create motor object,
64KHz pwm
//AF_DCMotor motor2(MOTOR_2, MOTOR12_64KHZ); // create motor object,
64KHz pwm
//AF_DCMotor motor3(MOTOR_3, MOTOR12_64KHZ); // create motor object,
64KHz pwm
AF_DCMotor motor4(MOTOR_4, MOTOR12_64KHZ); // create motor object,
64KHz pwm
//=====
=====
// Initialization
//=====
=====

int distance_L, distance_F, distance_R;
long distance;

int set = 20;

void setup() {
  Serial.begin(9600);           // Initialize serial port
  Serial.println("Start");

  myservo.attach(10);
  myservo.write(90);

  pinMode (trigpin, OUTPUT);
  pinMode (echopin, INPUT );

  motor1.setSpeed(180);
  motor4.setSpeed(180);       // set the motor speed to 0-255
  //motor2.setSpeed(180);
  // motor3.setSpeed(180);
}

```

```

//=====
=====
// Main
//=====
=====
void loop() {
  distance_F = data();
  Serial.print("S=");
  Serial.println(distance_F);
  if (distance_F > set){
    Serial.println("Forward");
    motor1.run(FORWARD);      // turn it on going forward
    // motor2.run(FORWARD);
    // motor3.run(FORWARD);
    motor4.run(FORWARD);
  }
  else{hc_sr4();}
}

long data(){
  digitalWrite(trigpin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigpin, HIGH);
  delayMicroseconds(10);
  distance = pulseIn (echopin, HIGH);
  return distance / 29 / 2;
}

void compareDistance(){
  if (distance_L > distance_R){
    motor1.run(BACKWARD);    // turn it on going left
    //motor2.run(BACKWARD);
    // motor3.run(FORWARD);
    motor4.run(FORWARD);
    delay(350);
  }
  else if (distance_R > distance_L){
    motor1.run(FORWARD);    // the other right
    // motor2.run(FORWARD);
    // motor3.run(BACKWARD);
    motor4.run(BACKWARD);
    delay(350);
  }
  else{
    motor1.run(BACKWARD);   // the other way
    //motor2.run(BACKWARD);
    // motor3.run(BACKWARD);
  }
}

```

```
motor4.run(BACKWARD);
  delay(300);
motor1.run(BACKWARD); // turn it on going left
// motor2.run(BACKWARD);
// motor3.run(FORWARD);
motor4.run(FORWARD);
  delay(500);
}
}

void hc_sr4(){
  Serial.println("Stop");
  motor1.run(RELEASE); // stopped
  // motor2.run(RELEASE);
  // motor3.run(RELEASE);
  motor4.run(RELEASE);

  myservo.write(0);
  delay(300);
  distance_R = data();
  delay(100);
  myservo.write(170);
  delay(500);
  distance_L = data();
  delay(100);
  myservo.write(90);
  delay(300);
  compareDistance();
}
```

