# Informe Final de Proyecto de Laboratorio

Autor: Wadhy Zahir Callan

## Título del Proyecto:

Vehículo de Exploración y Recolección de Datos en Terrenos de Difícil Acceso

## Propósito y Funcionalidades del Prototipo

El objetivo principal de este proyecto es desarrollar un vehículo autónomo y controlado de forma remota que pueda recolectar datos ambientales en terrenos y ambientes de difícil acceso, donde el ser humano no puede llegar fácilmente. Este vehículo está diseñado específicamente para tareas de exploración y reconocimiento.

### Funcionalidades principales del prototipo:

1. Recolección de datos ambientales:  
 - Medición de temperatura y humedad mediante un sensor DHT11.  
 - Detección de niveles de monóxido de carbono (CO) con un sensor MQ9.  
2. Almacenamiento de datos:  
 - Los datos recopilados se registran y almacenan en una tarjeta SD utilizando un módulo lector de tarjeta SD.  
3. Sistema móvil autónomo:  
 - Implementación de un sistema de control remoto vía Bluetooth (HC-05).  
 - Capacidad para moverse de manera estable en terrenos complejos, utilizando un sistema de motores y sensores para interactuar con su entorno.

## Selección de Componentes

### Placas de Control:

1. Arduino UNO R3 (Primera placa):  
 - Controla los motores a través de un shield L293D.  
 - Gestiona la comunicación Bluetooth con el módulo HC-05.  
2. Arduino UNO R3 (Segunda placa):  
 - Gestiona la adquisición de datos mediante los sensores DHT11 y MQ9.  
 - Almacena los datos en una tarjeta SD utilizando el módulo lector correspondiente.

### Sensores y Actuadores:

1. Sensor DHT11: Mide la temperatura y la humedad.  
2. Sensor MQ9: Detecta niveles de monóxido de carbono (CO).  
3. Motores:  
 - Cuatro motores DC de 3 a 6 V con caja reductora.  
4. Módulo Bluetooth HC-05: Permite el control remoto del vehículo.

### Otros Componentes:

1. Módulo lector de tarjeta SD para el almacenamiento de datos.  
2. Shield Drive L293D para el control de los motores.  
3. Cables jumper para realizar las conexiones eléctricas.

## Código Fuente: Control Remoto del Vehículo

#include <AFMotor.h>  
#include <SoftwareSerial.h>  
  
int bluetoothTx = 2;  
int bluetoothRx = 3;  
SoftwareSerial bluetooth(bluetoothTx, bluetoothRx);  
  
AF\_DCMotor motor1(1, MOTOR12\_1KHZ);  
AF\_DCMotor motor2(2, MOTOR12\_1KHZ);  
AF\_DCMotor motor3(3, MOTOR34\_1KHZ);  
AF\_DCMotor motor4(4, MOTOR34\_1KHZ);  
  
void setup() {  
 Serial.begin(9600);  
 bluetooth.begin(9600);  
 motor1.setSpeed(200);  
 motor2.setSpeed(200);  
 motor3.setSpeed(200);  
 motor4.setSpeed(200);  
 detener();  
 Serial.println("Esperando comandos Bluetooth...");  
}  
  
void loop() {  
 if (bluetooth.available()) {  
 char command = bluetooth.read();  
 Serial.print("Comando recibido: ");  
 Serial.println(command);  
  
 switch (command) {  
 case 'F': adelante(); break;  
 case 'B': atras(); break;  
 case 'L': izquierda(); break;  
 case 'R': derecha(); break;  
 case 'S': detener(); break;  
 default: detener(); break;  
 }  
 }  
}  
  
void adelante() {  
 motor1.run(FORWARD);  
 motor2.run(FORWARD);  
 motor3.run(FORWARD);  
 motor4.run(FORWARD);  
}  
  
void atras() {  
 motor1.run(BACKWARD);  
 motor2.run(BACKWARD);  
 motor3.run(BACKWARD);  
 motor4.run(BACKWARD);  
}  
  
void izquierda() {  
 motor1.run(FORWARD);  
 motor2.run(FORWARD);  
 motor3.run(BACKWARD);  
 motor4.run(FORWARD);  
}  
  
void derecha() {  
 motor1.run(FORWARD);  
 motor2.run(FORWARD);  
 motor3.run(FORWARD);  
 motor4.run(BACKWARD);  
}  
  
void detener() {  
 motor1.run(RELEASE);  
 motor2.run(RELEASE);  
 motor3.run(RELEASE);  
 motor4.run(RELEASE);  
}

## Código Fuente: Recolector de Datos Ambientales

#include <Adafruit\_Sensor.h>  
#include <DHT.h>  
#include <DHT\_U.h>  
#include <SD.h>  
  
#define DHTPIN A0  
#define DHTTYPE DHT11  
  
DHT dht(DHTPIN, DHTTYPE);  
#define MQ9PIN A1  
  
const int chipSelect = 10;  
  
void setup() {  
 Serial.begin(9600);  
 dht.begin();  
 if (!SD.begin(chipSelect)) {  
 Serial.println("Error al inicializar la tarjeta SD.");  
 while (true);  
 }  
 Serial.println("Tarjeta SD inicializada correctamente.");  
}  
  
void loop() {  
 float hum = dht.readHumidity();  
 float temp = dht.readTemperature();  
 if (isnan(hum) || isnan(temp)) {  
 Serial.println("Error al leer el sensor DHT.");  
 return;  
 }  
 int gasValue = analogRead(MQ9PIN);  
 Serial.print("Temperatura: ");  
 Serial.print(temp);  
 Serial.print(" °C, Humedad: ");  
 Serial.print(hum);  
 Serial.print(" %, Gas: ");  
 Serial.println(gasValue);  
  
 File dataFile = SD.open("datos.txt", FILE\_WRITE);  
 if (dataFile) {  
 dataFile.print("Temperatura: ");  
 dataFile.print(temp);  
 dataFile.print(" °C, Humedad: ");  
 dataFile.print(hum);  
 dataFile.print(" %, Gas: ");  
 dataFile.println(gasValue);  
 dataFile.close();  
 Serial.println("Datos guardados correctamente en la tarjeta SD.");  
 } else {  
 Serial.println("Error al abrir el archivo en la tarjeta SD.");  
 }  
 delay(1000);  
}