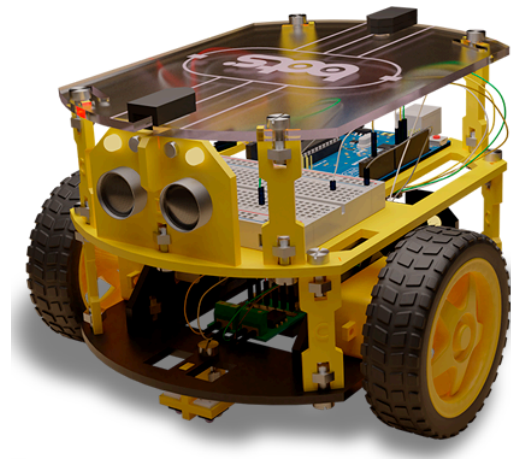


Proyecto Bots

Introducción

¡Bienvenido al emocionante mundo de la robótica! En este instructivo, aprenderás a construir tu propio **vehículo a control remoto** utilizando un Arduino. Pero eso no es todo: este robot será muy especial porque también tendrá luces para iluminar su camino y un **sensor ultrasónico** que le permitirá esquivar obstáculos automáticamente.



Este proyecto es perfecto para empezar a explorar cómo funcionan los robots y la programación, además de ser una forma divertida de aprender ciencia y tecnología.

A lo largo de este instructivo, seguiremos pasos simples para armar el robot, programarlo y ponerlo en acción. Al final, tendrás un vehículo que podrás controlar desde la distancia y que además será capaz de moverse de manera inteligente por su cuenta.

¡Prepárate para convertirte en un inventor y darle vida a este increíble robot! 🚗🤖

Materiales necesarios

- Piezas constructivas
- Placa Arduino UNO
- Protoboard
- Cable UBS
- Cable jack
- S. Ultrasónico
- S. Infrarrojo x2
- R. Infrarrojo + Control remoto
- Fotorresistencia x2
- Motor DC + Rueda x2
- Puente H
- Leds Blancos x2
- Leds Rojos x2
- Leds RGB x2
- Cables M-M 10cm
- Cables M-M 20cm
- Cables M-H 10cm
- Cables M-H 20cm
- Resistencias 10k Ω
- Resistencias 180 Ω
- Resistencias 220 Ω
- Resistencias 270 Ω
- Resistencias 330 Ω

Paso 1: Armar el chasis y montar los motores

En este paso, ensamblaremos la base del robot, instalaremos los motores y conectaremos los componentes principales. Sigue estos pasos:

1.

2. Fijar el puente H

Toma la **placa B** (la base del robot) y coloca el **puente H** en el lugar indicado.

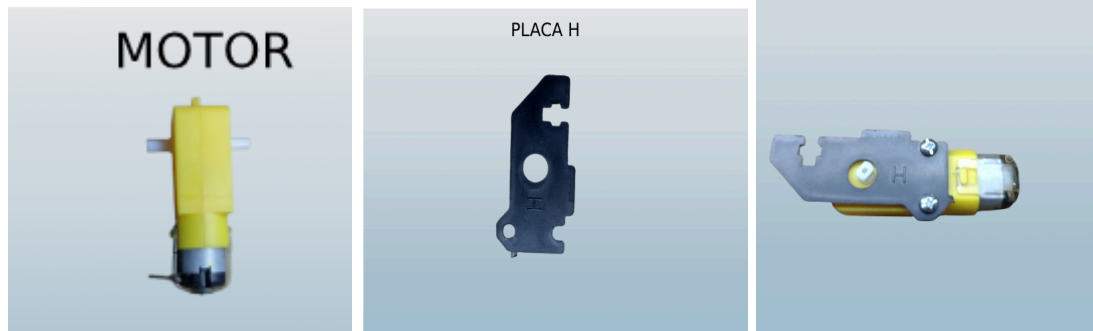
Utiliza los **tornillos A2** para fijarlo firmemente a la placa.



3. Montar los motores en las bases

Coloca los motores en las **bases H** (las piezas diseñadas para sostener los motores).

Asegúralos utilizando los **tornillos A3** y las tuercas correspondientes.



4. Instalar los motores en la placa B

Monta las bases con los motores ya instalados sobre la **placa B**.

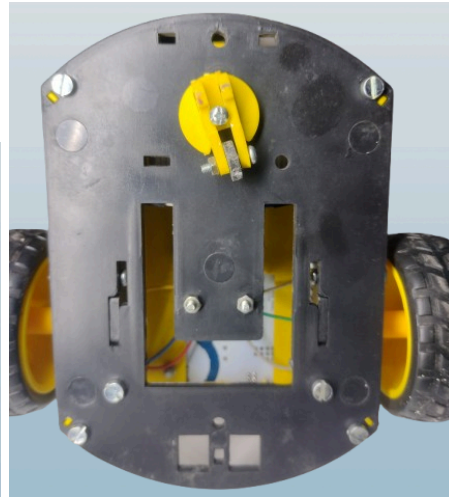
Fíjalos con los **tornillos A1** para que queden firmes.



5. Colocar la rueda de apoyo

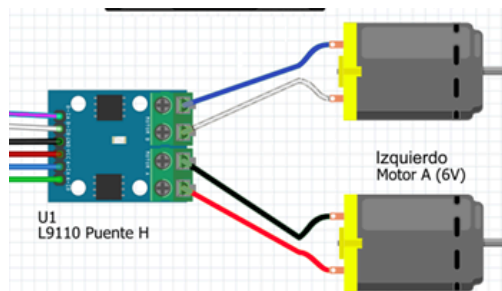
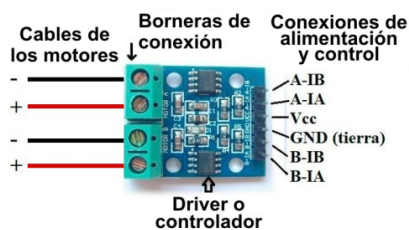
Instala la **rueda de apoyo M** (la rueda delantera).

Asegúrala con el tornillo A4 en el lugar indicado(asegurate de no ajustar del todo para que tenga movimiento).



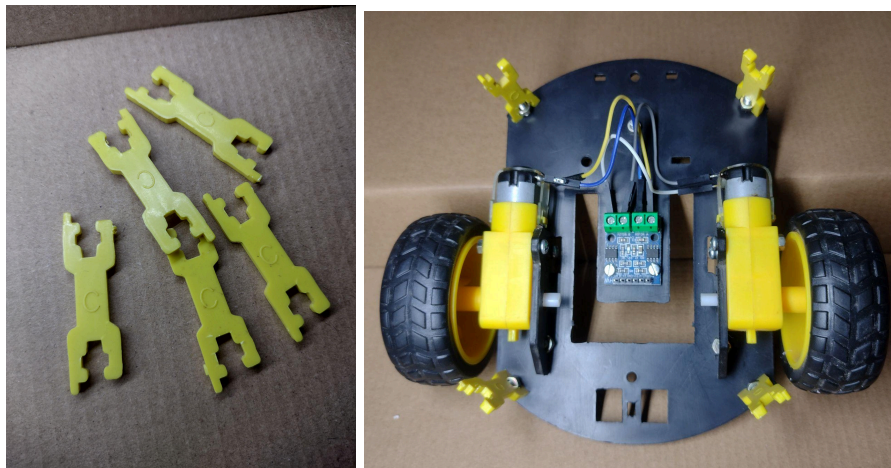
6. Conectar los motores al puente H

- Utiliza los cables M-H de 10 cm para conectar los motores al puente H.



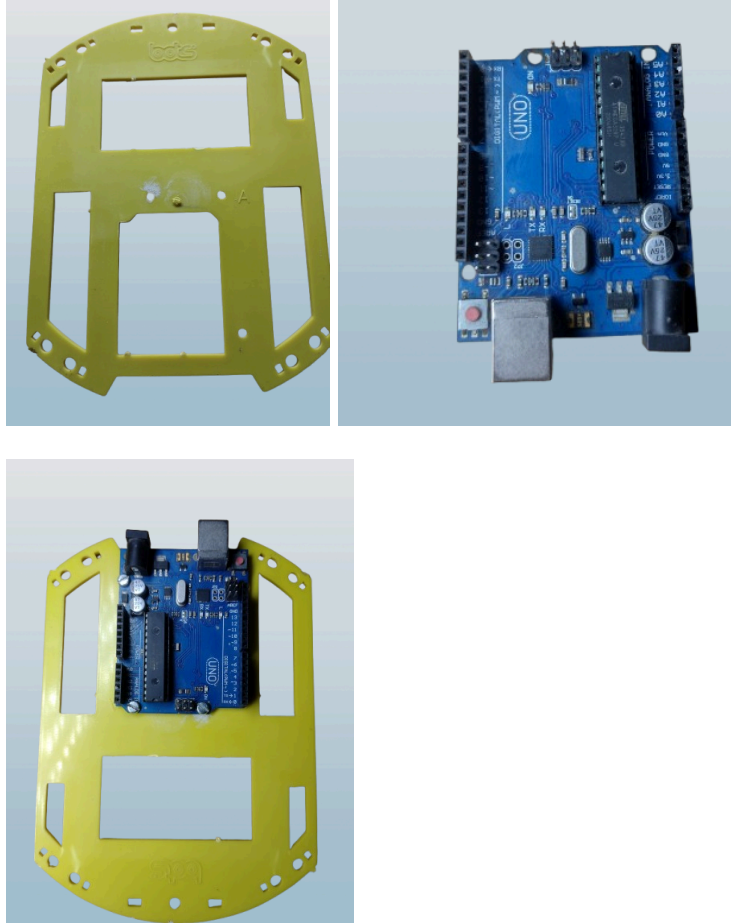
7. Instalar los pilares C

- atornilla los pilares C en la placa B utilizando los tornillos A1.
- Estos pilares servirán como soporte para la placa superior.



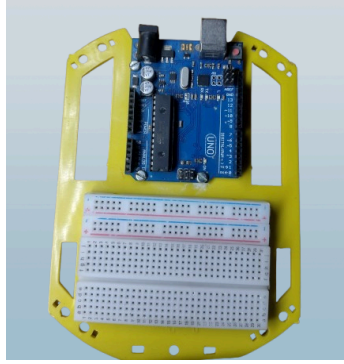
8. Montar el Arduino Uno

- Toma la placa A (la superior) y coloca el Arduino Uno en el lugar indicado.
- Fíjalo con los tornillos A2 para que quede seguro.



9. Pegar la protoboard

- a. Usa cinta doble faz para fijar la protoboard en la placa A junto al Arduino.



b.

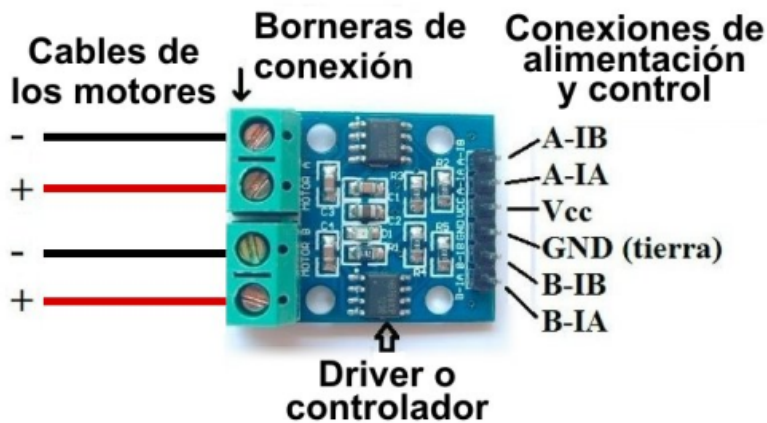
10. Acoplar la placa A sobre los pilares C

- a. Coloca la placa A encima de los pilares instalados en la placa B.
- b. Atornilla la placa con los tornillos A1 para completar la estructura.

Paso 2: Conectar el arduino y los motores

1. Identificar las terminales del puente H

- a. Encuentra las terminales etiquetadas como:
 - i. **VCC:** Entrada de energía positiva.
 - ii. **GND:** Conexión a tierra.
 - iii. **A-IB, I-IA, B-IB, B-IA:** Entradas de control desde el Arduino.



2. Conectar el puente H al Arduino

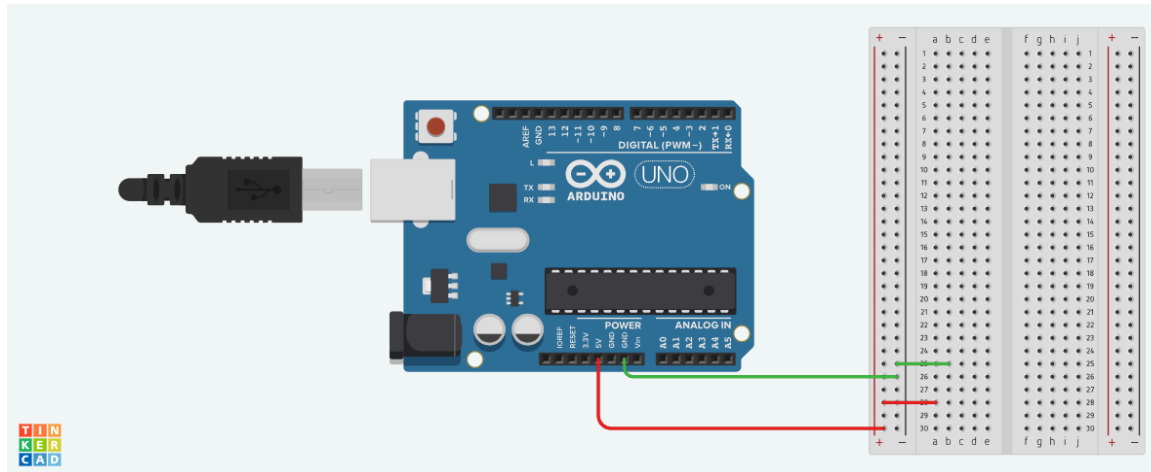
- a. Conecta las terminales A-IB, I-IA del puente H a los pines digitales del Arduino (por ejemplo, pin 6 y pin 7).
- b. Conecta las terminales B-IB, B-IA del puente H a otros pines digitales del Arduino (por ejemplo, pin 11 y pin 10).



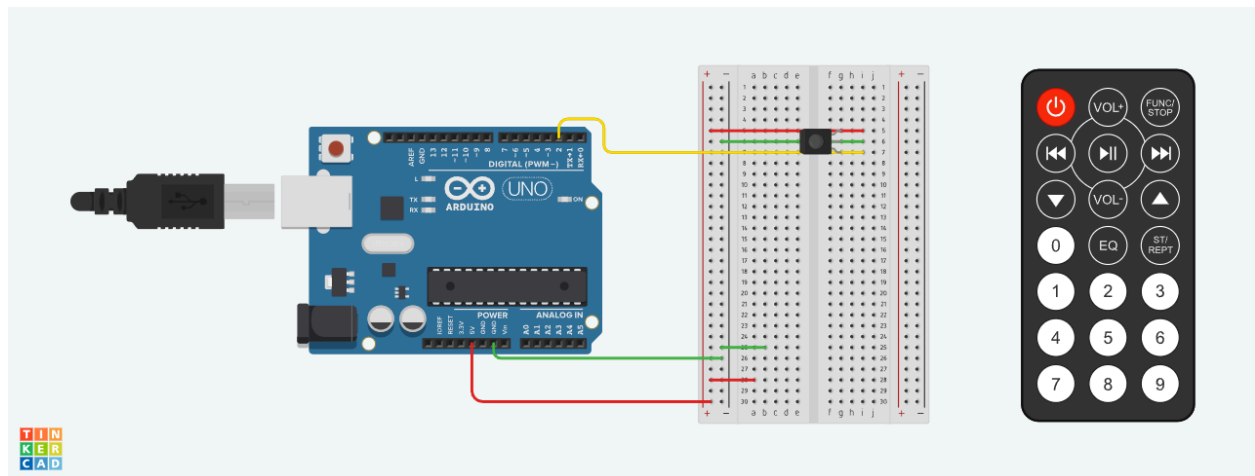
c.

3. Conectar la alimentación del puente H

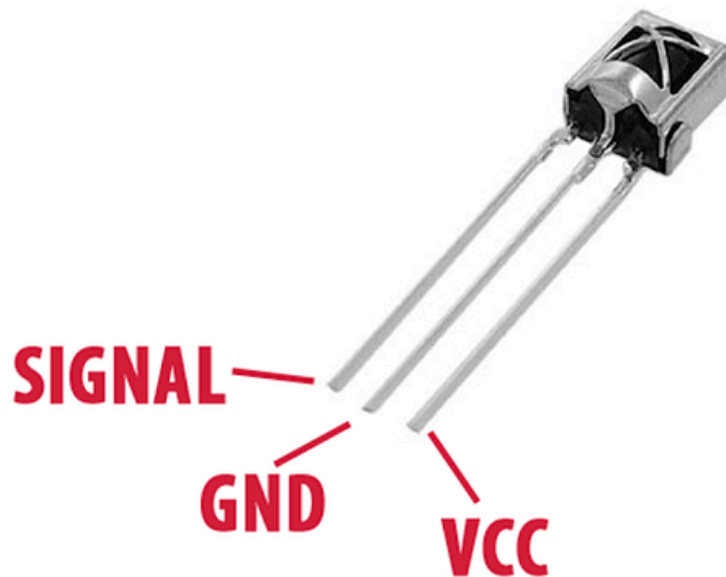
- Conecta la terminal VCC del puente H a la misma fila de la protoboard que conectaste pin 5V del Arduino para alimentarlo (ej fila 28 lado a-e).
- Conecta la terminal GND del puente H a la misma fila de la protoboard que conectaste pin Gnd del Arduino para alimentarlo (ej fila 25 lado a-e).



Paso 2: Conectar el control remoto



- c. Conecta la terminal VCC del recepto IR ala misma fila de la protoboard que lleva 5V para hacerlo toma un cable y conecta un extremo a la fila + de la protoboard y llevalo hasta la fila que deseas conectar (ej fila 5 lado f-j).
- d. Conecta la terminal GND del recepto IR ala misma fila de la protoboard que lleva GND para hacerlo toma un cable y conecta un extremo a la fila - de la protoboard y llevalo hasta la fila que deseas conectar (ej fila 6 lado f-j).
- e. Conecta la terminal Signal a un pin digital del Arduino uno(ej pin 2)



Paso 3: Programación

Instalación del IDE de Arduino

El IDE de Arduino es el programa que usaremos para escribir, cargar y probar el código del robot. Sigue estos pasos para instalarlo:

1. Descargar el IDE de Arduino
 - a. Abre tu navegador de internet y ve al sitio oficial de Arduino: <https://www.arduino.cc/>.
 - b. En la página principal, busca la sección Software y selecciona Downloads.
 - c. Elige la versión del IDE que corresponda a tu sistema operativo:
 - i. Windows

ii. macOS

iii. Linux

2. Instalar el IDE

- a. Una vez descargado el archivo, ábrelo para iniciar la instalación.
- b. Sigue las instrucciones del instalador:
- c. Acepta los términos y condiciones.
- d. Selecciona la ubicación de instalación (puedes dejar la opción predeterminada).
- e. Permite que se instalen los controladores necesarios para que el Arduino funcione correctamente.
- f. Cuando la instalación haya terminado, abre el programa para asegurarte de que funciona.

Código de ejemplo:

copia el siguiente código en un nuevo sketch del IDE arduino:

```
#include <IRremote.h> // Biblioteca para manejar el receptor infrarrojo
```

```
// Configuración del receptor IR
```

```
const int receptorIR = 2; // Pin conectado al receptor IR
```

```
// Definición de los códigos de los botones del control remoto
```

```
const unsigned long botonAvanzar = 0xE718FF00;
```

```
const unsigned long botonReversa = 0xAD52FF00;
```

```
const unsigned long botonIzquierda = 0xF708FF00;
```

```
const unsigned long botonDerecha = 0xA55AFF00;
```

```
const unsigned long botonStop = 0xE31CFF00;
```

```
// Configuración del receptor IR
```

```
IRrecv irrecv(receptorIR);
```

```
decode_results resultados;
```

```
// Configuración de los pines de los motores
```

```
const int motorIzq1 = 6; // Motor izquierdo, terminal A-IB
```

```
const int motorIzq2 = 7; // Motor izquierdo, terminal A-IA  
const int motorDer1 = 10; // Motor derecho, terminal B-IA  
const int motorDer2 = 11; // Motor derecho, terminal B-IB
```

```
void setup() {  
  // Inicializar comunicación serial para depuración  
  Serial.begin(9600);  
  
  // Configurar pines de motores como salida  
  pinMode(motorIzq1, OUTPUT);  
  pinMode(motorIzq2, OUTPUT);  
  pinMode(motorDer1, OUTPUT);  
  pinMode(motorDer2, OUTPUT);  
  
  // Configurar el pin LED de estado como salida  
  pinMode(13, OUTPUT);  
  
  // Iniciar el receptor IR  
  IrReceiver.begin(receptorIR, ENABLE_LED_FEEDBACK);  
  Serial.println("Esperando señales del control remoto...");  
}
```

```
void loop() {
```

```
// Verificar si se ha recibido una señal IR
if (IrReceiver.decode()) {
    unsigned long codigo = IrReceiver.decodedIRData.decodedRawData;

    // Filtrar señales inválidas o repetitivas
    if (codigo != 0x0 && codigo != 0xFFFFFFFF) {
        // Identificar el botón presionado
        if (codigo == botonAvanzar) {
            Serial.println("Avanzar");
            avanzar();
        } else if (codigo == botonReversa) {
            Serial.println("Reversa");
            retroceder();
        } else if (codigo == botonIzquierda) {
            Serial.println("Izquierda");
            girarIzquierda();
        } else if (codigo == botonDerecha) {
            Serial.println("Derecha");
            girarDerecha();
        } else if (codigo == botonStop) {
            Serial.println("Stop");
            stop();
        } else {
```



```
        Serial.print("Código no reconocido: ");
        Serial.println(codigo, HEX);
    }
} else {
    Serial.println("Código repetitivo ignorado");
}

// Preparar para recibir la siguiente señal
IrReceiver.resume();
}
}

// Funciones de control de los motores
void avanzar() {
    digitalWrite(motorIzq1, HIGH);
    digitalWrite(motorIzq2, LOW);
    digitalWrite(motorDer1, HIGH);
    digitalWrite(motorDer2, LOW);
}

void retroceder() {
    digitalWrite(motorIzq1, LOW);
    digitalWrite(motorIzq2, HIGH);
```

```
digitalWrite(motorDer1, LOW);  
digitalWrite(motorDer2, HIGH);  
}
```

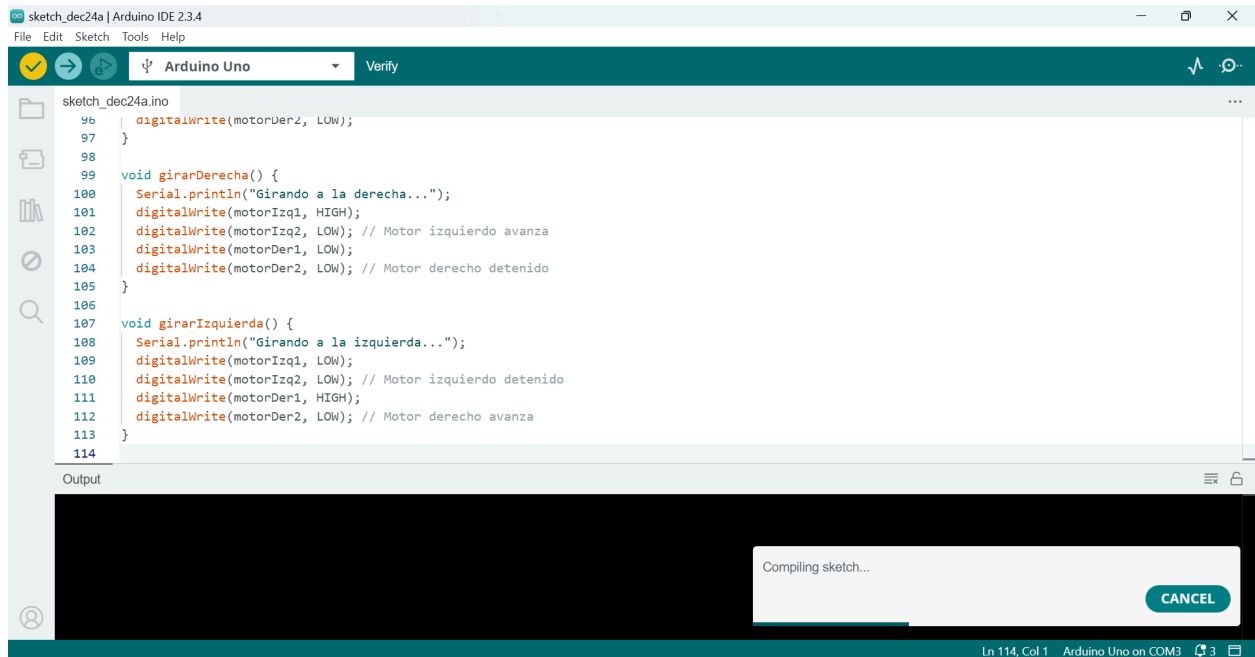
```
void stop() {  
    digitalWrite(motorIzq1, LOW);  
    digitalWrite(motorIzq2, LOW);  
    digitalWrite(motorDer1, LOW);  
    digitalWrite(motorDer2, LOW);  
}
```

```
void girarDerecha() {  
    Serial.println("Girando a la derecha...");  
    digitalWrite(motorIzq1, HIGH);  
    digitalWrite(motorIzq2, LOW); // Motor izquierdo avanza  
    digitalWrite(motorDer1, LOW);  
    digitalWrite(motorDer2, LOW); // Motor derecho detenido  
}
```

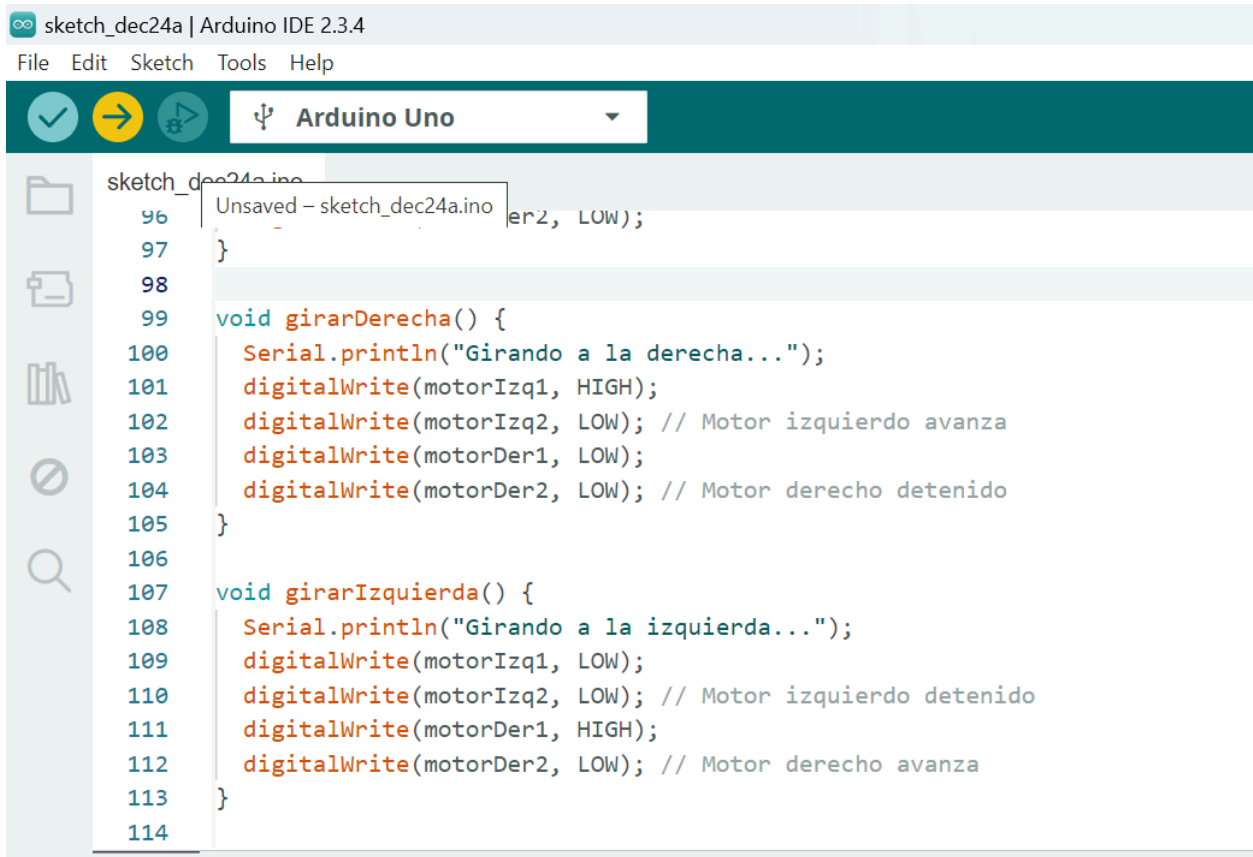
```
void girarIzquierda() {  
    Serial.println("Girando a la izquierda...");  
    digitalWrite(motorIzq1, LOW);  
    digitalWrite(motorIzq2, LOW); // Motor izquierdo detenido
```

```
digitalWrite(motorDer1, HIGH);  
  
digitalWrite(motorDer2, LOW); // Motor derecho avanza  
  
}
```

Verifica que no contenga errores:



haz click en el botón señalado en amarillo para ejecutar la verificación; luego con el Bot conectado a nuestro ordenador mediante USB cargamos el código a nuestro arduino:



```
sketch_dec24a | Arduino IDE 2.3.4
File Edit Sketch Tools Help

[Checkmark] [Yellow Arrow] [Gear] [USB] Arduino Uno

sketch_dec24a.ino
96   digitalWrite(motorDer2, LOW);
97   }
98
99   void girarDerecha() {
100     Serial.println("Girando a la derecha...");
101     digitalWrite(motorIzq1, HIGH);
102     digitalWrite(motorIzq2, LOW); // Motor izquierdo avanza
103     digitalWrite(motorDer1, LOW);
104     digitalWrite(motorDer2, LOW); // Motor derecho detenido
105   }
106
107   void girarIzquierda() {
108     Serial.println("Girando a la izquierda...");
109     digitalWrite(motorIzq1, LOW);
110     digitalWrite(motorIzq2, LOW); // Motor izquierdo detenido
111     digitalWrite(motorDer1, HIGH);
112     digitalWrite(motorDer2, LOW); // Motor derecho avanza
113   }
114
```

haz click en el botón señalado en amarillo y espera a que complete la carga.

Paso 3: Prueba de funcionamiento

Desconecta el Bot del USB

Conecta la batería mediante el cable plug

Prueba el con el control remoto:

- Botón 2: Avanza;
- Botón 4: Gira a la Izquierda;
- Botón 6: Gira a la Derecha;
- Botón 8: Retrocede;
- Botón 5: Stop;