
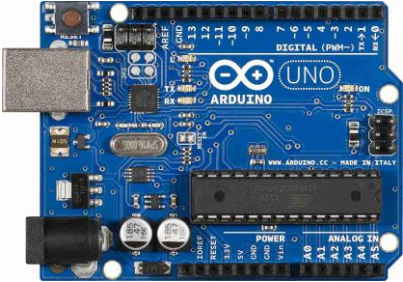


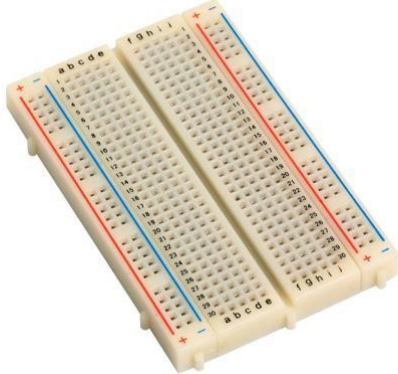
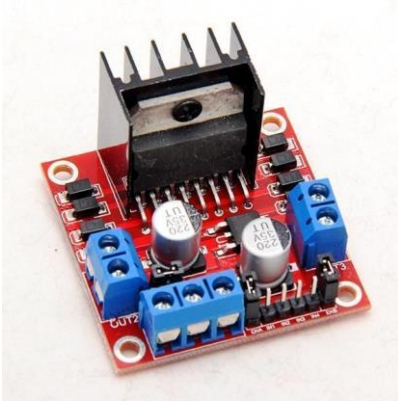

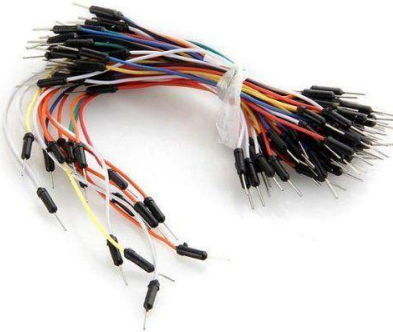
**MANUAL TÉCNICO
DE
“ROBOT SIGUE LINEAS”**



Este manual técnico proporcionará instrucciones detalladas para la construcción de un robot seguidor de líneas destinado a participar en una competición de carreras. El robot seguirá líneas utilizando sensores infrarrojos para detectar el camino y ajustará su dirección utilizando dos motores de corriente continua. A continuación, se detallan los pasos para construir este robot.

Materiales y Componentes.

Nombre	Imagen	Descripción
Plataforma universal robótica		<p>La plataforma que nos permitirá conseguir el movimiento de nuestro robot está compuesta por los siguientes elementos:</p> <ul style="list-style-type: none"> • 2 motores de corriente continua (motor CC) acoplados a ruedas neumáticas se encargan de dar tracción • Una rueda loca central permitirá que pueda girar. • El portapilas de 4 pilas AA nos permitirá alimentar nuestro dispositivo. • Una base de contrachapado hará de chasis
Placa Arduino Uno		<p>Placa Arduino Uno (o cualquier otra versión)</p>

<p>Protoboard, breadboard o placa de conexiones</p>		<p>Placa de conexiones o protoboard que usaremos en el montaje de circuitos eléctrico-electrónico para ahorrar y simplificar el cableado.</p>
<p>1 driver L298n</p>		<p>El driver L298n nos permitirá gobernar los motores de corriente continua con las ruedas acopladas y, por tanto, gobernar el movimiento de nuestro vehículo.</p>
<p>2 sensores de infrarrojos TCRT5000</p>		<p>El sensor de infrarrojos nos permitirá detectar la presencia de una línea negra y de esta manera enrutar nuestra trayectoria.</p>
<p>Cables de conexiones entre los componentes, la placa de pruebas y Arduino</p>		<p>Estos cables nos permiten conectar de manera sencilla todos los elementos del circuito.</p>

Pasos para la Construcción del Robot.

Paso 1: Ensamblaje de la Plataforma.

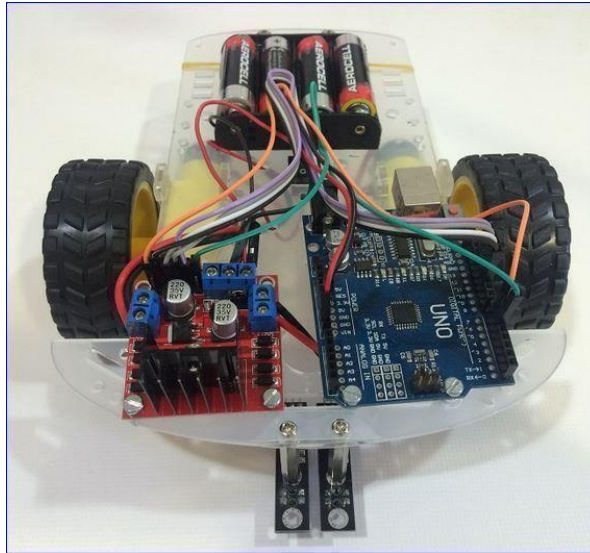
1. Coloca la base de contrachapado en la plataforma robótica y fíjala de manera segura. Esto servirá como el chasis del robot.
2. Instala los dos motores de corriente continua en la plataforma utilizando soportes o abrazaderas, asegurándote de que estén alineados y paralelos entre sí.
3. Conecta las ruedas neumáticas a los ejes de los motores.
4. Instala la rueda loca central en la parte frontal o trasera del robot para permitir giros suaves.



Paso 2: Montaje de Componentes Electrónicos

1. Coloca la Placa Arduino Uno en la plataforma, asegurándote de que esté bien sujeta.
2. Conecta el driver L298n a la Placa Arduino Uno utilizando cables jumper. Asegúrate de seguir el esquema de conexiones adecuado para el driver L298n.
3. Conecta los motores a las salidas del driver L298n, asegurándote de conectar los cables de manera que los motores giren en la dirección correcta.
4. Conecta los sensores de infrarrojos TCRT5000 a la protoboard y luego conecta la protoboard a la Placa Arduino Uno utilizando cables jumper.

Asegúrate de conectar los sensores según corresponda (izquierda y derecha).



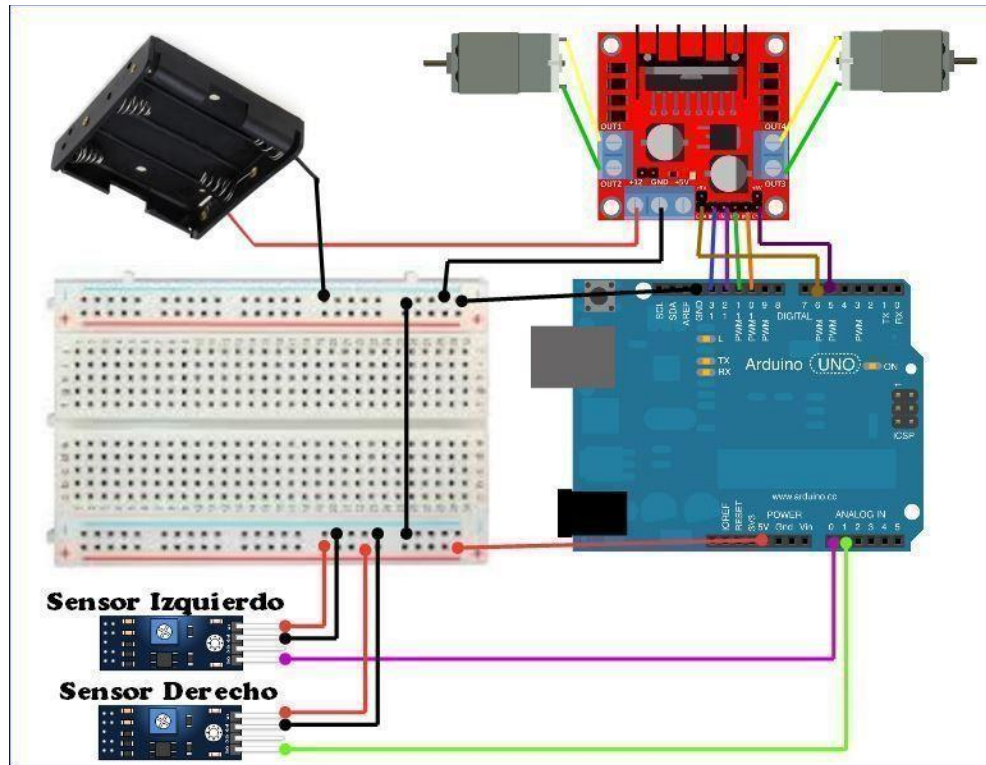
- **Lógica del circuito.**

- Si los dos sensores están sobre la línea: sigue recto.
- Si se ha salido hacia la izquierda: giramos a la derecha hasta tener los dos sensores sobre la línea.
- Si se ha salido hacia la derecha: giramos hacia la izquierda hasta tener los dos sensores sobre la línea.
- Si por alguna razón se salieran los dos, haremos que se pare.



- **Diagrama de conexiones.**

Dado que en otras sesiones ya trabajamos con el driver L298n para controlar 2 motores de corriente continua y con el sensor de infrarrojos TCRT5000 para detectar la presencia de la línea negra, combinaremos la circuitería de ambos circuitos para crear nuestro proyecto conjunto.



Paso 3: Programación del Robot.

1. Descarga e instala el entorno de desarrollo Arduino en tu computadora si aún no lo has hecho.

<https://www.arduino.cc/en/software>

2. Escribe el código necesario para que el robot siga líneas utilizando los datos de los sensores infrarrojos. Asegúrate de calibrar los sensores adecuadamente para la pista de la competición.

Arduino Code

```
1. // Definición de variables y constantes relacionadas con el m
   otor izquierdo
2. const int IN1 = 13; // Pin digital 13 para controlar sentido
   giro motor izquierdo
```

```

3. const int IN2 = 12; // Pin digital 12 para controlar sentido
   giro motor izquierdo
4. const int ENA = 6;
5.
6. // Definición de variables y constantes relacionadas con el m
   otor derecho
7. const int IN3 = 11; // Pin digital 11 para controlar sentido
   giro motor izquierdo
8. const int IN4 = 10; // Pin digital 10 para controlar sentido
   giro motor izquierdo
9. const int ENB = 5;
10.
11. const int vel = 175;
12.
13. // Definición de variables y constantes relacionadas con l
   os sensores IR
14. int lecturaSensorIzq;
15. int lecturaSensorDer;
16. const int sensorIzqPin = A0;
17. const int sensorDerPin = A1;
18.
19. void setup()
20. {
21.     // Se declaran todos los pines como salidas
22.     // Pines asociados a los motores
23.     pinMode (IN1, OUTPUT);
24.     pinMode (IN2, OUTPUT);
25.     pinMode (IN3, OUTPUT);
26.     pinMode (IN4, OUTPUT);
27.     pinMode (ENA, OUTPUT);
28.     pinMode (ENB, OUTPUT);
29.     pinMode( sensorIzqPin , INPUT) ;
30.     pinMode( sensorDerPin , INPUT) ;
31.     Serial.begin(9600); // Se inicia el puerto de comunicaci
   ones en serie
32. }
33.
34. void loop()
35. {
36.
37.     lecturaSensorIR(); // Se lee el valor de los sensores IR
38.     if(lecturaSensorIzq == 0 && lecturaSensorDer == 0
   )
39.     {
40.         robotParar(); // El robot para

```

```
41.
42.     }
43.     // Si el izquierdo retorna 0 (zona blanca) y el derecho 1
    (negra) el robot gira derecha
44.     if (lecturaSensorIzq == 0 && lecturaSensorDer ==
    1)
45.     {
46.         robotDerecha();
47.         // El robot gira a la derecha
48.
49.     }
50.     // Si el izquierdo retorna 1 (zona negra) y el derecho 0
    (blanca) el robot gira izquierda
51.     if (lecturaSensorIzq == 1 && lecturaSensorDer ==
    0)
52.     {
53.         robotIzquierda();
54.
55.     }
56.     // Si ambos sensores retornan 0 (zona negra) el robot si
    gue recto
57.     if (lecturaSensorIzq == 1 && lecturaSensorDer ==
    1)
58.     {
59.         robotAvance(); // El robot avanza
60.         Serial.println("robot avanza");
61.     }
62.
63. }
64. void lecturaSensorIR()
65. {
66.     lecturaSensorIzq = digitalRead(sensorIzqPin); // Almacen
    a la lectura del sensor izquierdo
67.     lecturaSensorDer = digitalRead(sensorDerPin); // Almacen
    a la lectura del sensor derecho
68.
69.     Serial.println("El valor del sensor izquierdo es ");
70.     Serial.println(lecturaSensorIzq);
71.
72.     Serial.println("El valor del sensor derecho es ");
73.     Serial.println(lecturaSensorDer);
74.
75. }
76. void robotAvance()
77. {
```



```

78.     digitalWrite (IN1, HIGH);
79.     digitalWrite (IN2, LOW);
80.     analogWrite (ENA, vel); //Velocidad motor A
81.
82.     // Motor derecho
83.     // Al mantener un pin HIGH y el otro LOW el motor gira e
n un sentido
84.     digitalWrite (IN3, HIGH);
85.     digitalWrite (IN4, LOW);
86.     analogWrite (ENB, vel); //Velocidad motor B
87. }
88. void robotRetroceso()
89. {
90.     digitalWrite (IN1, LOW);
91.     digitalWrite (IN2, HIGH);
92.     analogWrite (ENA, vel); //Velocidad motor A
93.     digitalWrite (IN3, LOW);
94.     digitalWrite (IN4, HIGH);
95.     analogWrite (ENB, vel); //Velocidad motor B
96. }
97. void robotDerecha()
98. {
99.     // Motor izquierdo
100.    // Se activa el motor izquierdo
101.    digitalWrite (IN1, HIGH);
102.    digitalWrite (IN2, LOW);
103.    analogWrite (ENA, vel); //Velocidad motor A
104.
105.    // Motor derecho
106.    // Se para el motor derecho
107.    digitalWrite (IN3, LOW);
108.    digitalWrite (IN4, HIGH);
109.    analogWrite (ENB, vel); //Velocidad motor A
110.
111. }
112. )
113. void robotIzquierda ()
114. {
115.     // Motor izquierdo
116.     // Se para el motor izquierdo
117.     digitalWrite (IN1, LOW);
118.     digitalWrite (IN2, HIGH);
119.     analogWrite (ENA, vel); //Velocidad motor A
120.
121.     // Motor derecho

```

```

122.    // Se activa el motor derecho
123.    digitalWrite (IN3, HIGH);
124.    digitalWrite (IN4, LOW);
125.    analogWrite (ENB, vel); //Velocidad motor A
126. }
127. void robotParar()
128. {
129.    // Motor izquierdo
130.    // Se para el motor izquierdo
131.    digitalWrite (IN1, LOW);
132.    digitalWrite (IN2, LOW);
133.
134.    // Motor derecho
135.    // Se para el motor derecho
136.    digitalWrite (IN3, LOW);
137.    digitalWrite (IN4, LOW);
138.
139. }

```

En la segunda no controlaremos la velocidad del motor y no estarán activados los jumpers (ENA, ENB) del driver L298n.

Arduino Code

```

1. // Definición de variables y constantes relacionadas con el motor i
   izquierdo
2. const int IN1 = 13; // Pin digital 13 para controlar sentido giro m
   otor izquierdo
3. const int IN2 = 12; // Pin digital 12 para controlar sentido giro m
   otor izquierdo
4. // Definición de variables y constantes relacionadas con el motor d
   erecho
5. const int IN3 = 11; // Pin digital 11 para controlar sentido giro m
   otor izquierdo
6. const int IN4 = 10; // Pin digital 10 para controlar sentido giro m
   otor izquierdo
7. // Definición de variables y constantes relacionadas con los sensor
   es IR
8. int lecturaSensorIzq; // Almacena el valor de la lectura del sensor
   izquierdo
9. int lecturaSensorDer; // Almacena el valor de la lectura del sensor
   derecho

```

```

10. const int sensorIzqPin = A1; // El sensor izq irá conectado al pin
    analógico A0
11. const int sensorDerPin = A0          ; // El sensor derecho irá c
    onectado al pin analógico A1
12.
13. void setup()
14. {
15.     pinMode (IN1, OUTPUT);
16.     pinMode (IN2, OUTPUT);
17.     pinMode (IN3, OUTPUT);
18.     pinMode (IN4, OUTPUT);
19.     pinMode( sensorIzqPin , INPUT) ;
20.     pinMode( sensorDerPin , INPUT) ;
21.     Serial.begin(9600); // Se inicia el puerto de comunicaciones en
        serie
22. }
23.
24. void loop()
25. {
26.
27.     lecturaSensorIR(); // Se lee el valor de los sensores IR
28.     if(lecturaSensorIzq == 0 && lecturaSensorDer == 0)
29.     {
30.         robotParar(); // El robot para
31.     }
32.     // Si el izquierdo retorna 0 (zona blanca) y el derecho 1 (negra
        ) el robot gira derecha
33.     if (lecturaSensorIzq == 0 && lecturaSensorDer == 1)
34.     {
35.         robotDerecha();
36.         // El robot gira a la derecha
37.     }
38.     // Si el izquierdo retorna 1 (zona negra) y el derecho 0 (blanca
        ) el robot gira izquierda
39.     if (lecturaSensorIzq == 1 && lecturaSensorDer == 0)
40.     {
41.         robotIzquierda();
42.     }
43.     }
44.     if (lecturaSensorIzq == 1 && lecturaSensorDer == 1)
45.     {
46.         robotAvance(); // El robot avanza
47.         Serial.println("robot avanza");
48.     }
49.

```

```
50. }
51. void lecturaSensorIR()
52. {
53.     lecturaSensorIzq = digitalRead(sensorIzqPin); // Almacena la lectura del sensor izquierdo
54.     lecturaSensorDer = digitalRead(sensorDerPin); // Almacena la lectura del sensor derecho
55.
56.     Serial.println("El valor del sensor izquierdo es ");
57.     Serial.println(lecturaSensorIzq);
58.
59.     Serial.println("El valor del sensor derecho es ");
60.     Serial.println(lecturaSensorDer);
61.
62. }
63. void robotAvance()
64. {
65.     // Motor izquierdo
66.     digitalWrite (IN1, HIGH);
67.     digitalWrite (IN2, LOW);
68.
69.     // Motor derecho
70.     digitalWrite (IN3, HIGH);
71.     digitalWrite (IN4, LOW);
72. }
73. void robotRetroceso()
74. {
75.     // Motor izquierdo
76.     digitalWrite (IN1, LOW);
77.     digitalWrite (IN2, HIGH);
78.
79.     // Motor derecho
80.     digitalWrite (IN3, LOW);
81.     digitalWrite (IN4, HIGH);
82. }
83. void robotDerecha()
84. {
85.     // Motor izquierdo
86.     // Se activa el motor izquierdo
87.     digitalWrite (IN1, HIGH);
88.     digitalWrite (IN2, LOW);
89.
90.     // Motor derecho
91.     // Se para el motor derecho
92.     digitalWrite (IN3, LOW);
```

```

93.   digitalWrite (IN4, HIGH);
94.
95. }
96. void robotIzquierda ()
97. {
98.     // Motor izquierdo
99.     digitalWrite (IN1, LOW);
100.    digitalWrite (IN2, HIGH);
101.
102.    // Motor derecho
103.    // Se activa el motor derecho
104.    digitalWrite (IN3, HIGH);
105.    digitalWrite (IN4, LOW);
106. }
107. void robotParar()
108. {
109.    // Motor izquierdo
110.    digitalWrite (IN1, LOW);
111.    digitalWrite (IN2, LOW);
112.
113.    // Motor derecho
114.    // Se para el motor derecho
115.    digitalWrite (IN3, LOW);
116.    digitalWrite (IN4, LOW);
117.
118. }

```

3. Carga el código en la Placa Arduino Uno utilizando un cable USB.

Paso 4: Pruebas y Ajustes.

1. Coloca el robot en una pista de prueba o en la pista de competición y asegúrate de que los sensores detecten la línea correctamente.
2. Realiza pruebas de seguimiento de líneas y ajusta el código según sea necesario para que el robot siga el camino de manera precisa.

Paso 5: Preparación para la Competición.

1. Asegúrate de que el robot esté en óptimas condiciones antes de la competición, y realiza pruebas en la pista de la competición si es posible.
2. Realiza ajustes finales en el hardware y el software según sea necesario.