

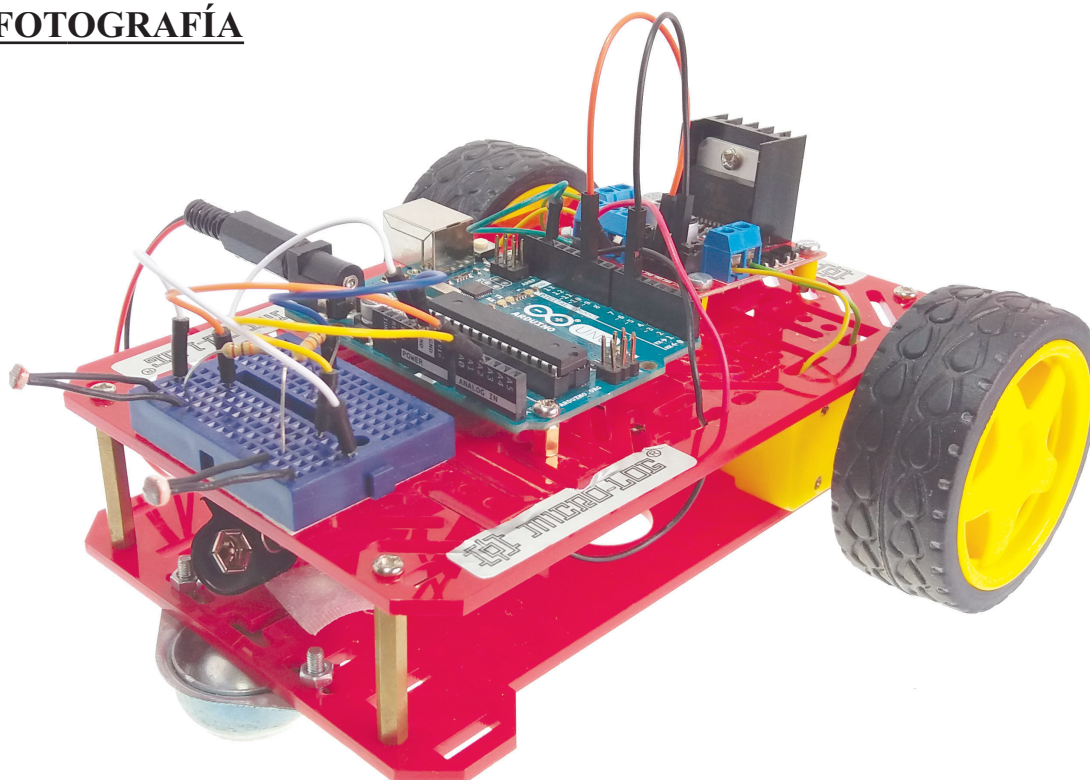
SEGUIDOR DE LUZ ARDUINO

MICRO-LOG[®]
LOGKIT
4204A

1. OBJETIVOS

Construir un robot autónomo seguidor de la luz.

2. FOTOGRAFÍA



3. FUNCIONAMIENTO

Las resistencias LDR se conectan a dos entradas analógicas de la tarjeta Arduino. Si las LDR detectan una luz, la tarjeta Arduino procesará la señal y dirigirá al robot hacia ella.

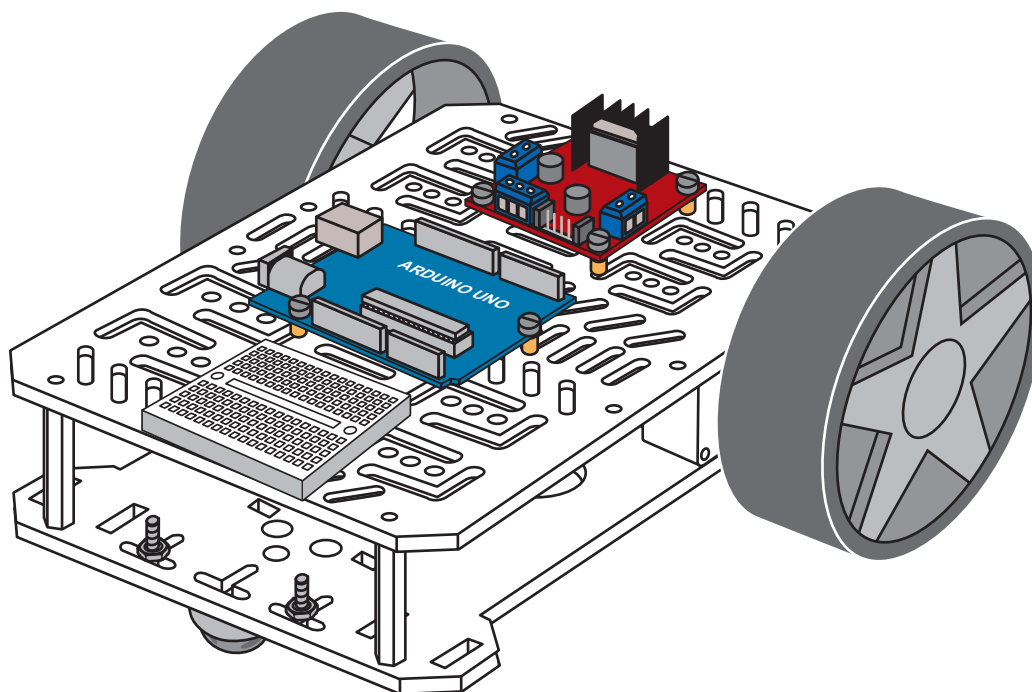
4. LISTA DE MATERIALES

- | | |
|-----------------------------------------|-----------------------------------------|
| 1 Arduino UNO REV - 3 LOG 4031 | 1 Mini placa board 4,5x3,5 cm LOG 885 |
| 1 Cable USB A-B LOG 4009 | 1 Controlador de motores L298N LOG 4044 |
| 1 Plataforma móvil LOG 4080 | 1 Conector 9V para Arduino LOG 7734 |
| 6 Tornillos M3 x 16 mm LOG 464M | 1 Conjuntos de latiguillos LOG 9519 |
| 6 Tuercas M3 LOG 480 | 2 Latiguillos macho-hembra S 9518 |
| 2 LDR LOG 731 | 6 Bulones de plástico S 220P |
| 2 Resistencias 10K Ω LOG 748 10K | 1 Hoja Técnica H4204A |

Leer todas las instrucciones y comprobar el listado de materiales antes de empezar el proyecto.

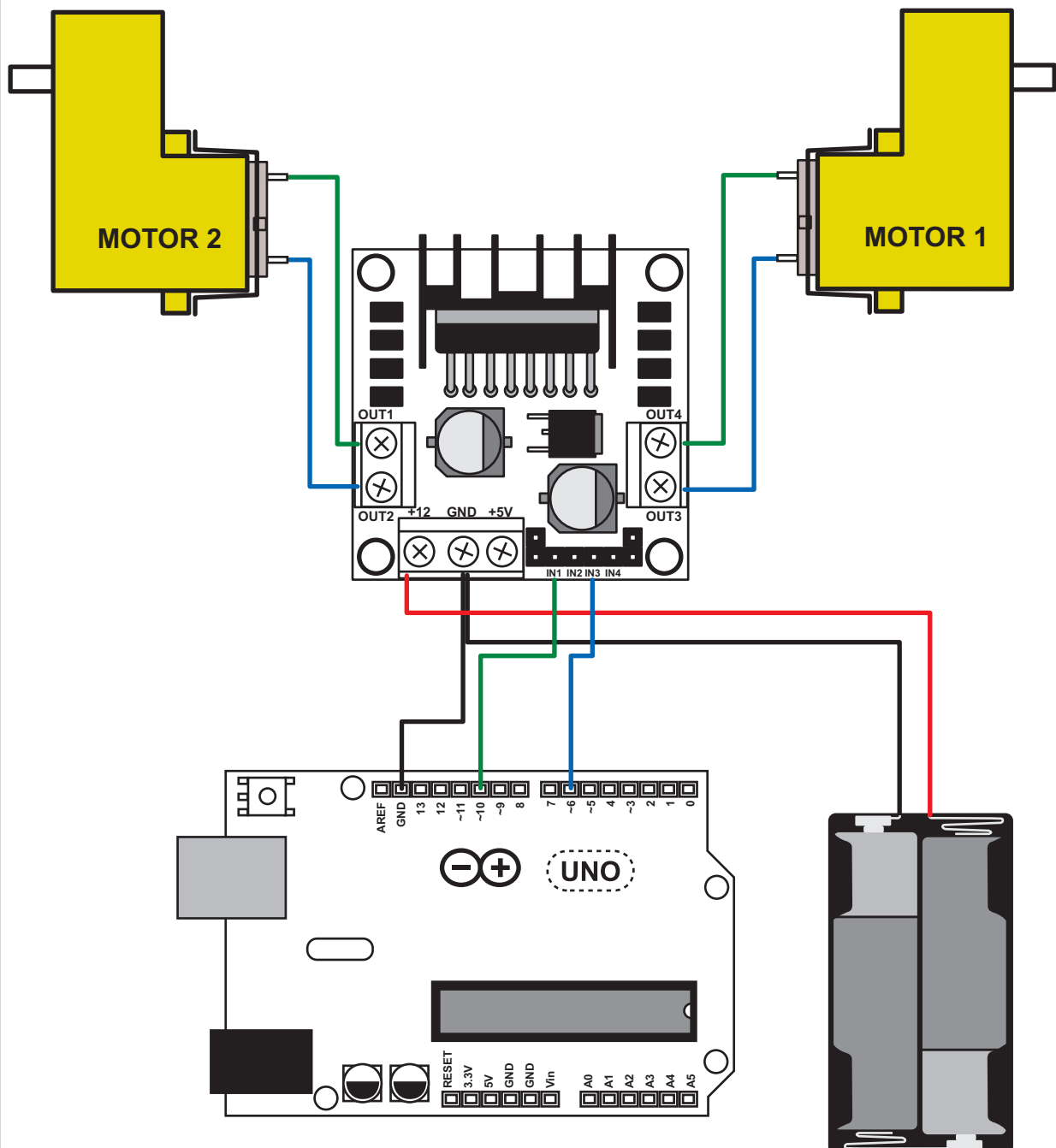
5. CONSTRUCCIÓN

- a) Montar la plataforma móvil según sus instrucciones.
- b) Soldar un latiguillo macho-macho a cada uno de los terminales de las reductoras.
- c) Pegar la placa board LOG 885 en la parte superior de la plataforma, como se indica en el dibujo.
- d) Atornillar el Arduino UNO en la plataforma superior, utilizando 2 tornillos M3 LOG 464, 2 bulones de plástico y 2 tuercas M3 LOG 480.
- e) Atornillar el controlador de motores con 4 tornillos M3 LOG 464, 4 bulones de plástico y 4 tuercas M3 LOG 480.



6. CONEXIÓN DEL CONTROL DE MOTORES

- El esquema incluye un controlador de motores L298N puente H. Un puente H consiste en 4 interruptores que nos permiten controlar la polaridad de la corriente que llega a los dos motores conectados al controlador de motores. Así en función de la polaridad podemos controlar el sentido de giro de los motores. Además funcionan como regulador de corriente, pudiendo determinar una velocidad de giro de 0 a 255.
- Para poder controlar la velocidad de giro de los motores, se han conectado a las salidas 6 y 10 de Arduino que permiten la modularidad (PWM).

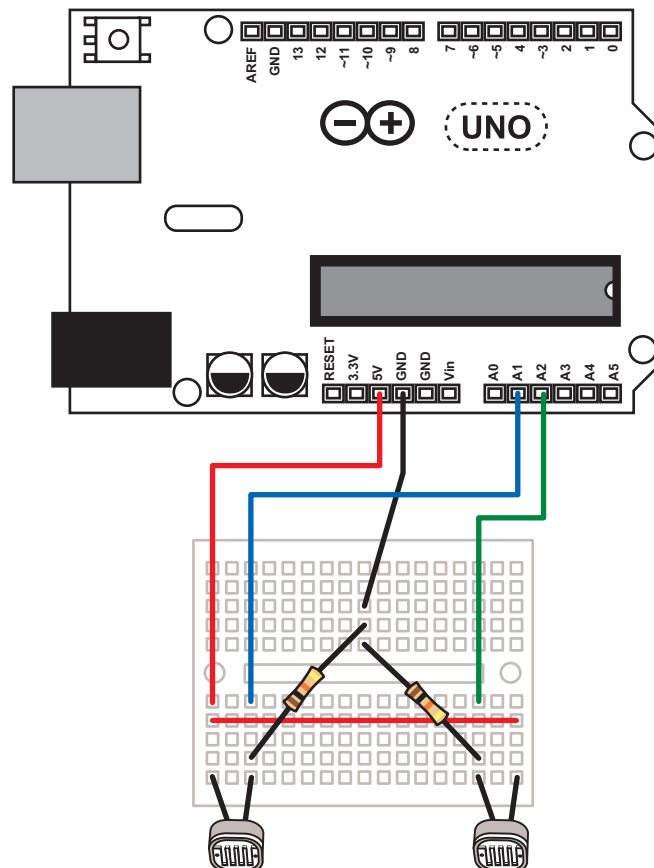


7. CONEXIÓN DE LOS SENSORES LDR

- El cometido de este vehículo es avanzar hacia una fuente de luz. Para detectar esa luz se han colocado en su frontal dos sensores basados en resistencias LDR. Estas resistencias están conectadas a dos entradas analógicas de la tarjeta Arduino (A1 y A2) y son alimentadas por la salida de 5V de la propia tarjeta.
- El programa que se ha planteado lee la señal generada por las LDR y se dirige en la dirección de la LDR que mayor luz detecte.

RESISTENCIAS

10K Ω : Marrón - Negro - Naranja



8. PROGRAMACIÓN CON MBOT

Makeblock es una herramienta de programación por bloques basado en Scratch.

Para instalar el software de Makeblock, entramos en makeblock.micro-log.com y desde el menú de "descargas" descargar el software mBlock.

Una vez instalado, abrir "mBlock":

- Pinchar en la pestaña "Placas" y pinchar donde pone "Arduino UNO".
- Pinchar en la pestaña "Conectar", seleccionar donde pone "Puerto serie" y pinchar donde pone "COM*Numero*".

```
Programa de Arduino
fijar motor2avance a 10
fijar motor1avance a 6
por siempre
  fijar sensorLDR1 a leer pin analógico (A) 1
  fijar sensorLDR2 a leer pin analógico (A) 2
  fijar pin PWM motor1avance a 0
  fijar pin PWM motor2avance a 0
  si sensorLDR1 > sensorLDR2 + 50 entonces
    fijar pin PWM motor1avance a 0
    fijar pin PWM motor2avance a 150
  si sensorLDR2 > sensorLDR1 + 50 entonces
    fijar pin PWM motor1avance a 150
    fijar pin PWM motor2avance a 0
```

9. PROGRAMACIÓN CON ARDUINO IDE

Arduino IDE es una herramienta de programación por códigos basado en C++.

Para instalar el software de Arduino, entramos en www.arduino.cc/en/Main/Software y descargamos el software Arduino IDE.

Una vez instalado, abrir "Arduino IDE":

- Pinchar en la pestaña "Herramientas", seleccionar donde pone "Placa" y pinchar en "Arduino/Genuino UNO".

- Pinchar en la pestaña "Herramientas", seleccionar donde pone "Puerto" y pinchar en "COM*Numero* Arduino/Genuino UNO".

Una vez realizado la conexión con la tarjeta ya podemos programar.

```

/*
  sigue_luz

  Programa guia a un vehículo hacia una fuente de luz

  Este programa ha sido creado por (c) Microlog Tecnología y Sistemas S.L.

  No está permitido el uso comercial del mismo ni su libre distribución

  Si está permitido su uso, modificación y distribución a nivel educativo
  dentro del mismo centro escolar, siempre respetando y manteniendo estas
  líneas y dejando clara la procedencia del mismo.

  Igualmente si se utiliza como material en algún curso, jornada o demostración
  etc debe ser citada la procedencia del material así como avisar previamente a
  microlog para obtener la autorización

  Para cualquier consulta www.microlog.net 917595910 pedidos@microlog.es
*/

int motor1Avance = 6; // pin para el sistema motriz LOG 48
int motor2Avance = 10; // pin para el sistema motriz LOG 48

void setup()
{
  pinMode(motor1Avance, OUTPUT); // pines de Salida
  pinMode(motor2Avance, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  int sensorLDR1 = analogRead(1); //Leemos el valor de las LDRs LOG 731
  int sensorLDR2 = analogRead(2);
  Serial.print(sensorLDR1);
  Serial.print(' ');
  Serial.println(sensorLDR2);
  if (sensorLDR1 > sensorLDR2+50) // Si LDR1 detecta mayor luz que LDR2
  {
    analogWrite(motor1Avance, 200);
    analogWrite(motor2Avance, 0); // Gira hacia la izquierda
    Serial.println("Mas luz en la izquierda");
  }
  if (sensorLDR2 > sensorLDR1+50) // Si LDR2 detecta mayor luz que LDR1
  {
    analogWrite(motor1Avance, 0); // Gira hacia la derecha
    analogWrite(motor2Avance, 200);
    Serial.println("Mas luz en la derecha");
  }
  delay(200);
  Serial.println("Parado"); // la plataforma LOG 4080 para
  analogWrite(motor1Avance, 0);
  analogWrite(motor2Avance, 0);
}

```

10. CALIBRACIÓN

- En los programas, el control de los motores se realiza a través del envío de señales analógicas a las salidas 6 y 10 de la tarjeta Arduino. La señal analógica a enviar tendrá un valor desde 0 hasta 255. Cada salida controla un motor del coche.
- Si en ambas salidas enviamos un 0, el coche se parará.
- La lógica nos dice que para que el coche circule en línea recta habrá que enviar a ambas salidas el mismo dato (mayor que 0). Si hacemos la prueba observamos que el coche tiende a girar en un determinado sentido. Esto es debido a la imprecisión de este tipo de motores. La mejor solución es regular la velocidad por programación enviando a un motor una señal ligeramente inferior con respecto al otro motor.
- Si lo que queremos es que el coche gire en un sentido, disminuirémos notablemente la velocidad de uno de los motores (enviando un dato de valor inferior a su salida de Arduino) provocando el giro gracias a la diferencia de velocidad entre ambos motores.

11. DETALLES DE TIPO PRÁCTICO

- Desconectar la alimentación del controlador de motores L298N cuando no se esté utilizando para no gastar las pilas.
- Necesita 4 pilas R6 de 1,5 V y 1 pila 6F22 de 9 V.
- Tiempo de construcción: 6 H.
- Nivel: Difícil
- Documentación en color y formato PDF: www.microlog.es/4204.zip

12. PRUEBAS

- Aumentar o disminuir la velocidad de los motores.