

SUMO O EXPLORADOR ARDUINO

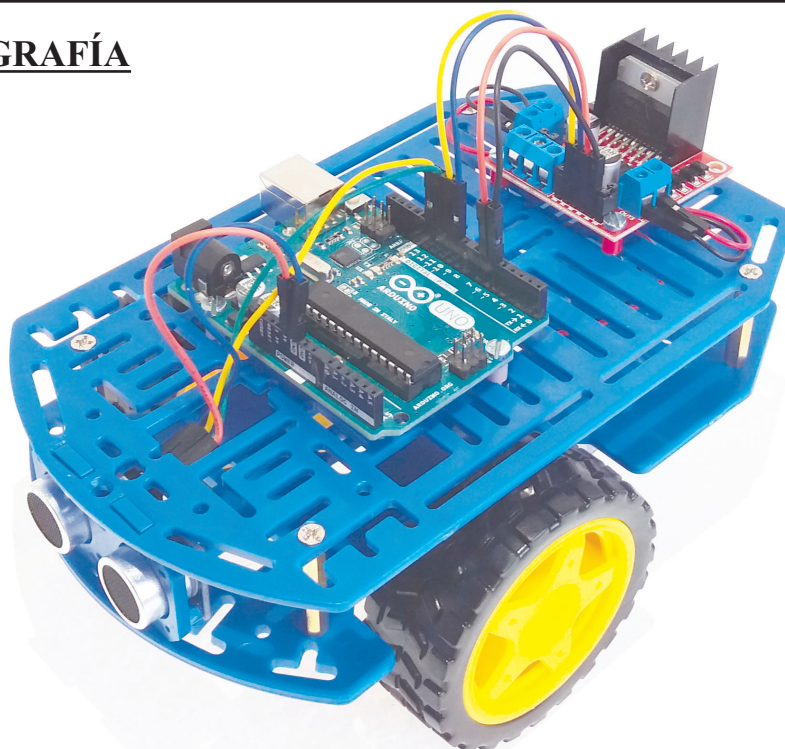
MICRO-LOG[®]
LOGKIT
4203A

1. OBJETIVOS

Con un mismo montaje el vehículo puede realizar dos funciones:

- Busca un oponente y lo empuja (lucha de sumo)
- Circula evitando obstáculos (laberinto/explorador)

2. FOTOGRAFÍA



3. FUNCIONAMIENTO

El coche dispone de un sensor ultrasónico que envía una señal a la tarjeta Arduino cuando detecta un objeto a una cierta distancia. Esta señal la utilizará la tarjeta para valorar si acudir hacia el objeto o cambiar su dirección según la función que estemos programando (sumo o explorador).

4. LISTA DE MATERIALES

- | | |
|---------------------------------|---|
| 1 Arduino UNO REV-3 LOG 4031 | 1 Controlador de motores L298N LOG 4044 |
| 1 Cable USB A-B LOG 4009 | 1 Conector 9V para Arduino LOG 7734 |
| 1 Plataforma móvil LOG 4079 | 4 Bulones de plástico S 220P |
| 4 Tornillos M3 x 16 mm LOG 464M | 8 Latiguillos macho-hembra S 9518 |
| 4 Tuercas M3 LOG 480 | 1 Latiguillo macho-macho S 9519 |
| 1 Sensor Ultrasónico LOG 4046 | 1 Hoja Técnica H4203A |

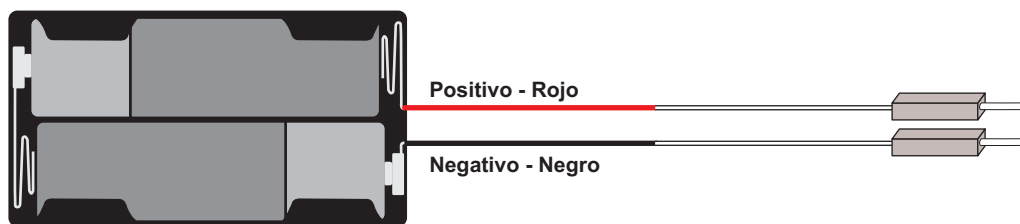
Leer todas las instrucciones y comprobar el listado de materiales antes de empezar el proyecto.

SUMO O EXPLORADOR ARDUINO

MICRO-LOG[®]
LOGKIT
4203A

5. CONSTRUCCIÓN

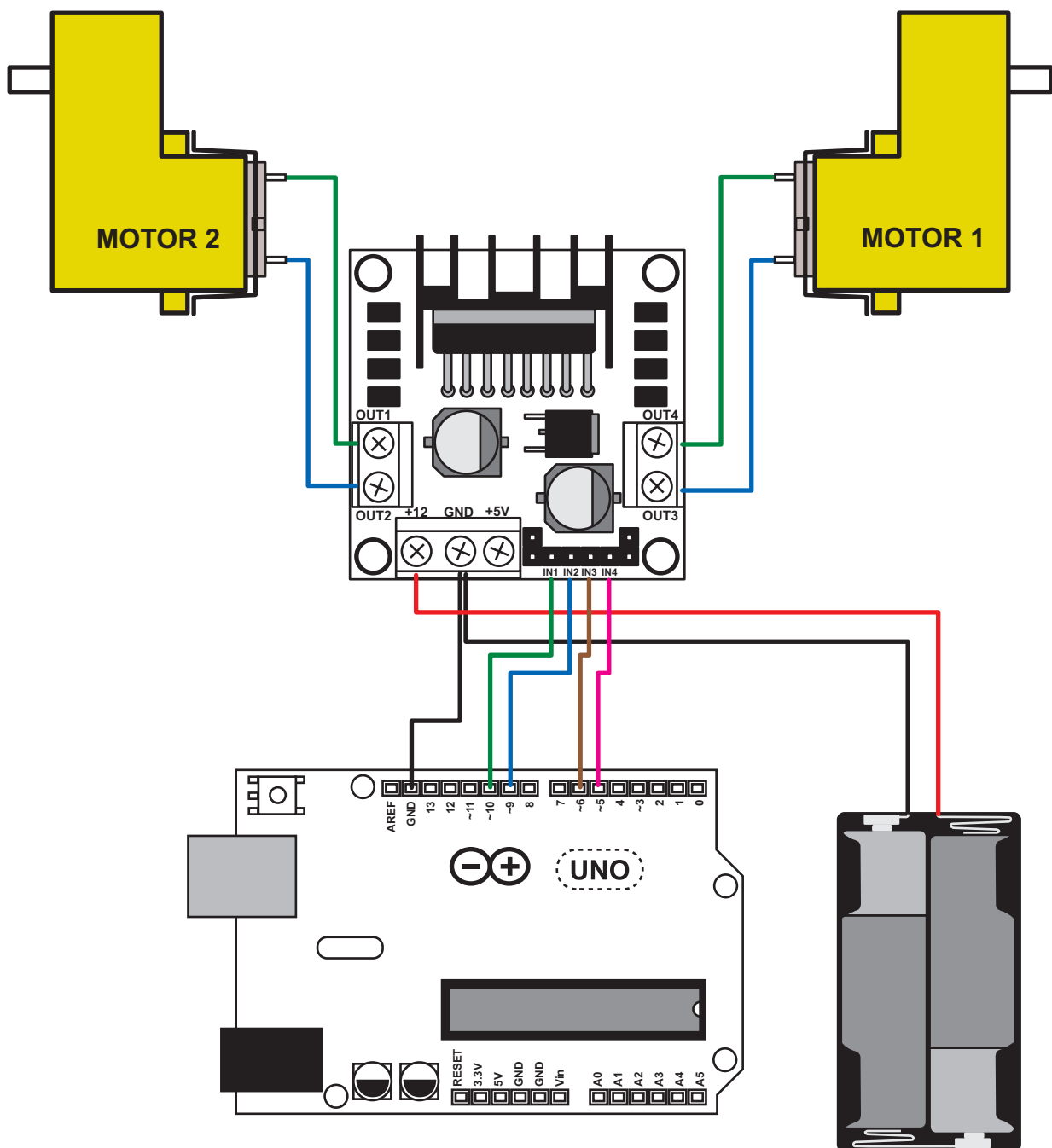
- Montar la plataforma móvil según sus instrucciones.
- Atornillar el Arduino UNO y el controlador de motores a la plataforma superior, utilizando 2 tornillos M3 LOG 464, 2 bulones de plástico y 2 tuercas M3 LOG 480, por componente.
- Cortar el latiguillo macho-macho por la mitad y soldar a los cables del portapilas.



- Aplicar una gota de pegamento termofusible en las esquinas del ultrasónico, para que se quede fijo en el soporte.

6. CONEXIÓN DEL CONTROL DE MOTORES

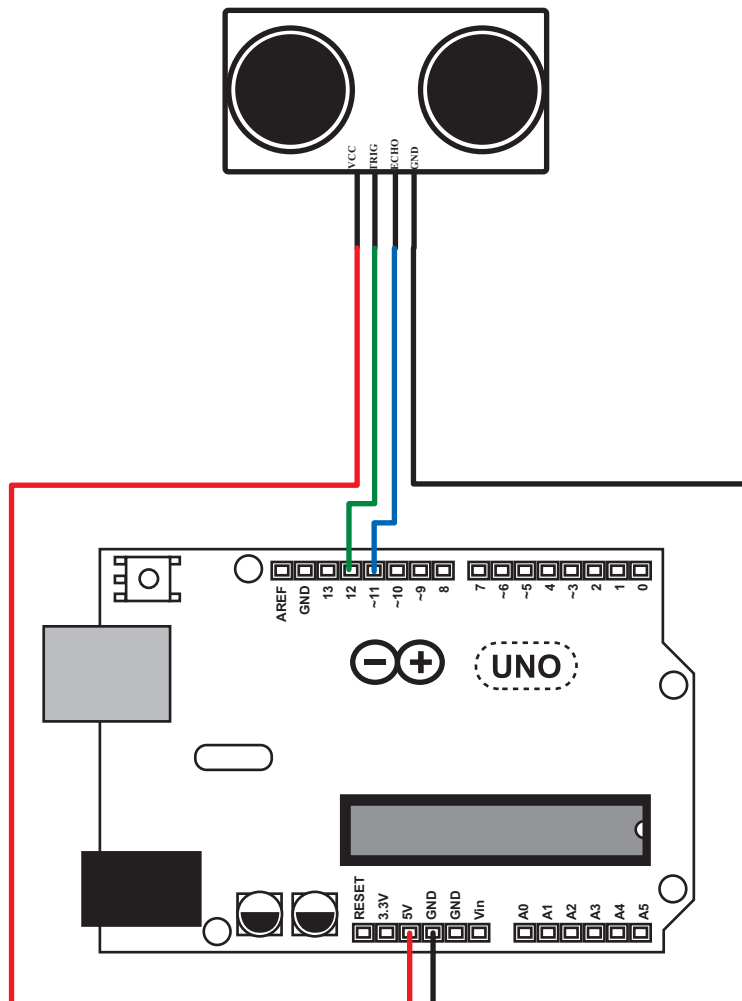
- El esquema incluye un controlador de motores L298N puente H. Un puente H consiste en 4 interruptores que nos permiten controlar la polaridad de la corriente que llega a los dos motores conectados al controlador de motores. Así en función de la polaridad podemos controlar el sentido de giro de los motores. 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 5, 6, 9 y 10 de Arduino que permiten la modularidad (PWM).



7. CONEXIÓN DEL SENSOR ULTRASONÍCO

El sensor ultrasonido dispone de 4 pines:

- VCC: se conecta a la salida de 5V de la tarjeta Arduino.
- GND: se conecta a una toma tierra (gnd) de la tarjeta Arduino.
- TRIG: se conecta a un pin digital de la tarjeta Arduino y su función será pedir al sensor que lance un pulso de sonido.
- ECHO: se conecta a un pin digital de la tarjeta Arduino y su función será medir el tiempo que tarda el pulso de sonido en llegar a un obstáculo.



8. PROGRAMACIÓN CON MBOT PARA EL SUMO

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 motor1avance a 6
fijar motor1atras a 5
fijar motor2avance a 10
fijar motor2atras a 9
por siempre
si lee el sensor ultrasónico trig pin 12 echo pin 11 < 10 entonces
  fijar salida pin digital motor1avance a ALTO
  fijar salida pin digital motor1atras a BAJO
  fijar salida pin digital motor2avance a ALTO
  fijar salida pin digital motor2atras a BAJO
si no
  fijar salida pin digital motor1avance a BAJO
  fijar salida pin digital motor1atras a ALTO
  fijar salida pin digital motor2avance a ALTO
  fijar salida pin digital motor2atras a BAJO
```

9. PROGRAMACIÓN CON MBOT PARA EL EXPLORADOR

```

Programade Arduino
fijar motor1avance a 6
fijar motor1atras a 5
fijar motor2atras a 9
fijar motor2avance a 10
fijar direccion a byte leído en el serial
por siempre
si lee el sensor ultrasónicotrig pin 12 echo pin 11 < 10 entonces
fijar pin PWM motor1avance a 0
fijar pin PWM motor1atras a 0
fijar pin PWM motor2avance a 0
fijar pin PWM motor2atras a 0
fijar direccion a número al azar entre 1 y 3
decir lee el sensor ultrasónicotrig pin 12 echo pin 11
si direccion = 1 entonces
fijar pin PWM motor1avance a 0
fijar pin PWM motor1atras a 150
fijar pin PWM motor2avance a 100
fijar pin PWM motor2atras a 0
si no
fijar pin PWM motor1avance a 100
fijar pin PWM motor1atras a 0
fijar pin PWM motor2avance a 0
fijar pin PWM motor2atras a 150
esperar 0,5 segundos
fijar pin PWM motor1avance a 0
fijar pin PWM motor1atras a 0
fijar pin PWM motor2avance a 0
fijar pin PWM motor2atras a 0
si no
fijar pin PWM motor1avance a 255
fijar pin PWM motor1atras a 0
fijar pin PWM motor2avance a 255
fijar pin PWM motor2atras a 0
    
```

10. PROGRAMACIÓN CON ARDUINO IDE PARA EL SUMO

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.

```

/*
  sumo

  Programa que guía al robot hacia un obstáculo y lo desplaza.
  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 pinSondeo = 12; // Pin para activar el sensor LOG 4046
int pinEcho = 11; // valor del sensor de ultrasonidos LOG 4046
int motor1Avance = 6; // pin para el sistema motriz LOG 48
int motor1Atras = 5; // pin para el sistema motriz LOG 48
int motor2Avance = 10; // pin para el sistema motriz LOG 48
int motor2Atras = 9; // pin para el sistema motriz LOG 48

void setup() {
  Serial.begin(9600);
  pinMode(pinSondeo, OUTPUT);
  pinMode(pinEcho, INPUT);
  pinMode(motor1Avance, OUTPUT); //pin de motores en modo salida
  pinMode(motor1Atras, OUTPUT);
  pinMode(motor2Avance, OUTPUT);
  pinMode(motor2Atras, OUTPUT);
}

void loop() {
  long duration, distance;
  digitalWrite(pinSondeo, LOW);
  delayMicroseconds(2);
  digitalWrite(pinSondeo, HIGH); //envío de pulso de luz
  delayMicroseconds(10);
  digitalWrite(pinSondeo, LOW);
  duration = pulseIn(pinEcho, HIGH); //lectura del tiempo que tardó
  Serial.println("Duration");
  Serial.println(duration);
  distance = (duration/2) / 29.1; //calcula la distancia
  if (distance < 100) { //si la distancia < 100
    digitalWrite(motor1Avance, HIGH); //avanza
    digitalWrite(motor1Atras, LOW);
    digitalWrite(motor2Avance, HIGH);
    digitalWrite(motor2Atras, LOW);
  }
  else{
    digitalWrite(motor1Avance, LOW); //busca
    digitalWrite(motor1Atras, HIGH);
    digitalWrite(motor2Avance, HIGH);
    digitalWrite(motor2Atras, LOW);
    Serial.println("Buscando");
  }
}

```

11. PROGRAMACIÓN PARA EL EXPLORADOR

```

/*
  explorador

  Programa que guía a un vehículo por una sala evitando obstáculos.
  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 pinSondeo = 12; // Pin para activar el sensor LOG 4046
// Pin para recibir el tiempo de desplazamiento del pulso del sensor LOG 4046
int pinEcho = 11;
int motor1Avance = 6; // Conexión del sistema motriz LOG 48
int motor1Atras = 5; // Conexión del sistema motriz LOG 48
int motor2Avance = 10; // Conexión del sistema motriz LOG 48
int motor2Atras = 9; // Conexión del sistema motriz LOG 48

void setup() {
  Serial.begin (9600);
  pinMode (pinSondeo, OUTPUT);
  pinMode (pinEcho, INPUT);
  pinMode (motor1Avance, OUTPUT); //Declara pin de Salida
  pinMode (motor1Atras, OUTPUT); //Declara pin de Salida
  pinMode (motor2Avance, OUTPUT); //Declara pin de Salida
  pinMode (motor2Atras, OUTPUT); //Declara pin de Salida
}

void loop() {
  long duration, distance;
  digitalWrite (pinSondeo, LOW);
  delayMicroseconds (2);
  digitalWrite (pinSondeo, HIGH); //envía pulso de luz
  delayMicroseconds (10);
  digitalWrite (pinSondeo, LOW);
  duration = pulseIn (pinEcho, HIGH); //recibe el tiempo de desplazamiento
  Serial.println ("Duration");
  Serial.println (duration);
  distance = (duration/2) / 29.1; //Constante para calcular distancia
  if (distance < 50) { //si estamos cerca del objeto
    analogWrite (motor1Avance, 0); // los motores LOG 48 se detienen
    analogWrite (motor1Atras, 0);
    analogWrite (motor2Avance, 0);
    analogWrite (motor2Atras, 0);
    int direccion = random (1, 3); //seleccionamos una dirección aleatoria
    Serial.println (direccion);
    if (direccion==1){
      Serial.println ("derecha"); // la plataforma LOG 4080 gira a la derecha
      analogWrite (motor1Avance, 0);
      analogWrite (motor1Atras, 150);
      analogWrite (motor2Avance, 100);
      analogWrite (motor2Atras, 0);
    }
    else{
      Serial.println ("izquierda"); // la plataforma LOG 4080 gira a la izquierda
      analogWrite (motor1Avance, 100);
      analogWrite (motor1Atras, 0);
      analogWrite (motor2Avance, 0);
      analogWrite (motor2Atras, 150);
    }
    delay (500);
    analogWrite (motor1Avance, 0); // la plataforma LOG 4080 se detiene
    analogWrite (motor1Atras, 0);
    analogWrite (motor2Avance, 0);
    analogWrite (motor2Atras, 0);
  }
  else{
    analogWrite (motor1Avance, 150); // la plataforma LOG 4080 avanza recto
    analogWrite (motor1Atras, 0);
    analogWrite (motor2Avance, 125);
    analogWrite (motor2Atras, 0);
    Serial.println ("avanzo, campo libre");
  }
}

```


SUMO O EXPLORADOR ARDUINO



LOGKIT
4203A

12. CALIBRACIÓN

- En los programas, el control de los motores se realiza a través del envío de señales analógicas a los pares de salidas 5-6 y 9-10. Cada motor está controlado por dos pines de la tarjeta controladora, y cada pin determinará el sentido de giro del motor, permitiendo que el robot circule de frente o marcha atrás. Estos pines serán tratados como señales analógicas, enviándoles datos de 0 a 255 para poder controlar la velocidad del motor.
- 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.
- En cuanto al sentido de giro de los motores, éste viene controlado por las conexiones al puente H. Si observamos que cada rueda gira en un sentido, tendremos que modificar el cableado. Por ejemplo si tenemos los motores conectados a las salidas 1 y 3 del puente H y cada motor gira en un sentido, tendremos que cambiar uno de los motores y o bien pasarle de la salida 1 a la 2 o sino de la salida 3 a la 4.

13. 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/4203.zip

14. PRUEBAS

- Aumentar o disminuir la velocidad de los motores.