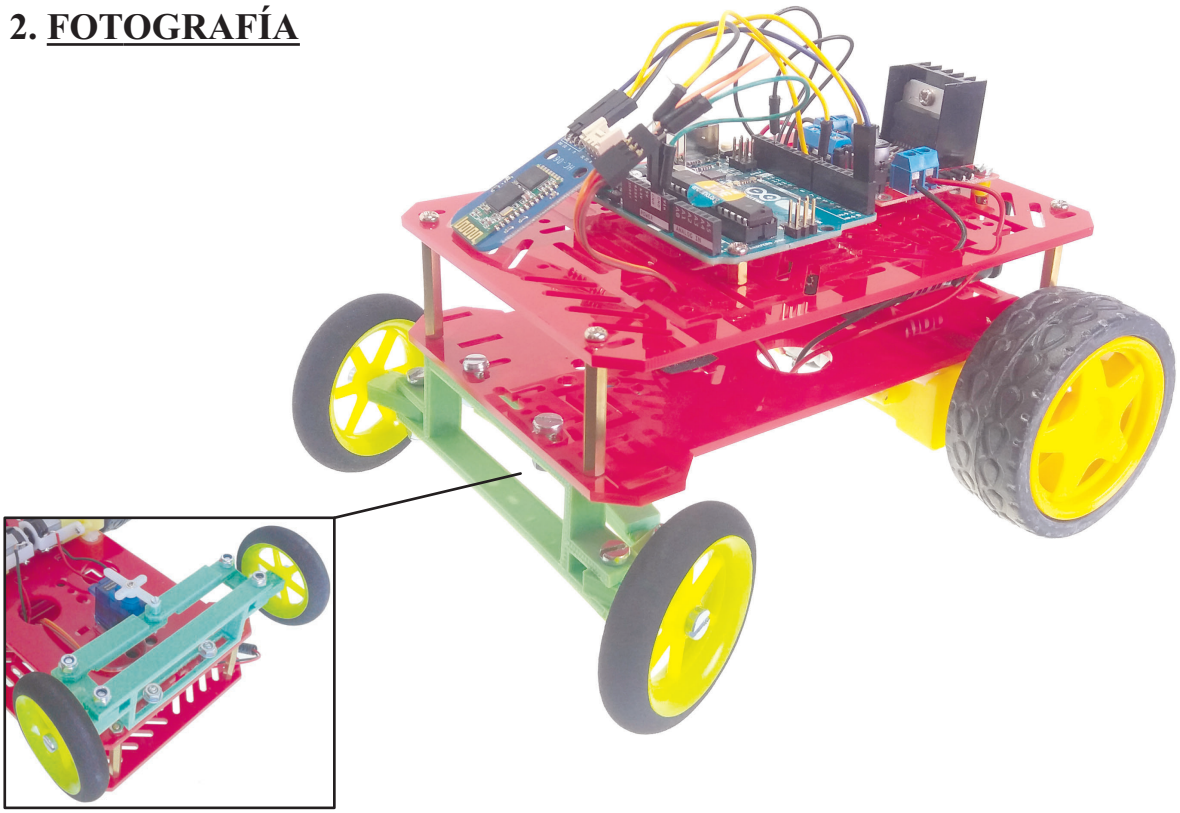
COCHE BLUETOOTH CON DIRECCIÓN

MICRO-LOG
LOGKIT
4202A

1. OBJETIVOS

Construir un coche cuyos movimientos se controlan con un dispositivo móvil con función bluetooth.

2. FOTOGRAFÍA



3. FUNCIONAMIENTO

El circuito del coche contiene un dispositivo bluetooth que permite su comunicación con cualquier móvil. Instalando la app "coche bluetooth microlog" podrás dirigir el coche con un móvil android. La app envía una serie de códigos al Arduino que éste interpreta generando el movimiento deseado en los motores.

4. LISTA DE MATERIALES

1 Arduino UNO REV - 3 LOG 4031
1 Cable USB A-B LOG 4009
1 Plataforma móvil LOG 4080
1 Servo motor LOG 06
2 Ruedas de goma espuma LOG 44
6 Tornillos M3 x 16 mm LOG 464M
8 Tornillos M4 x 16 LOG 467
6 Tuercas M3 LOG 480
6 Tuercas autoblocantes M4 LOG 484
1 Tornillo M2 x 20 mm LOG 474
1 Tuerca Autoblocante M2 LOG 482P
1 Controlador de motores L298N LOG 4044
1 Shield Bluetooth HC-06 LOG 4058
1 Conector 9V para Arduino LOG 7734
8 Latiguillos hembra-macho S 9518
8 Latiguillos macho-macho S 9519
6 Bulones de plástico S 220P
1 Hoja Técnica H4202A

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

5. CONSTRUCCIÓN CON BOLA DE RODADURA

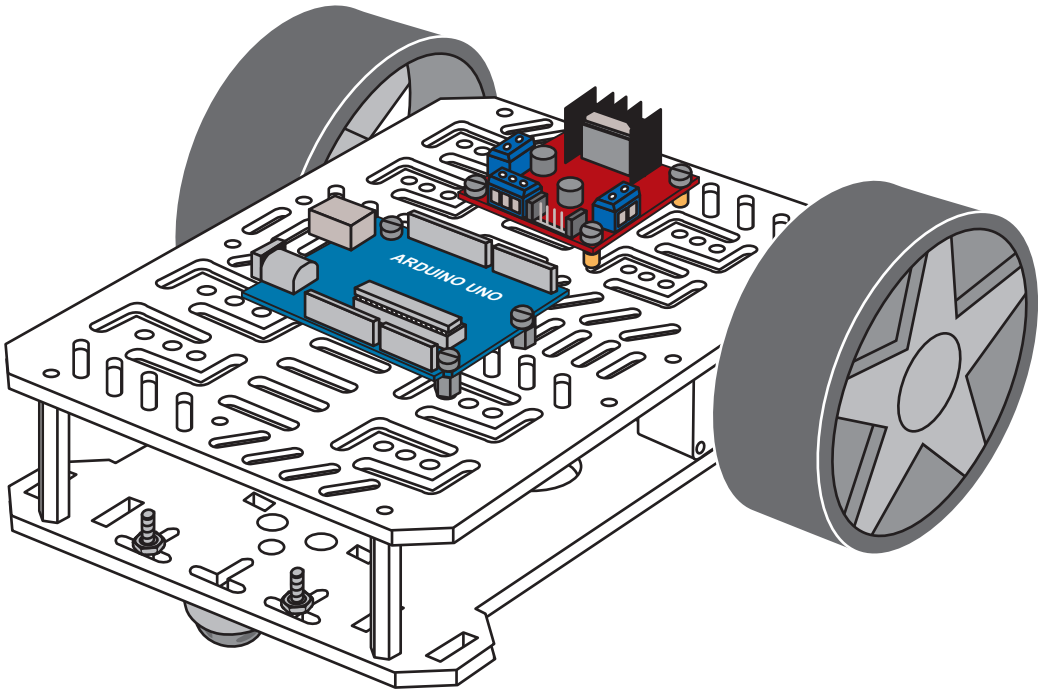
a) Montar la plataforma móvil según sus instrucciones.

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.

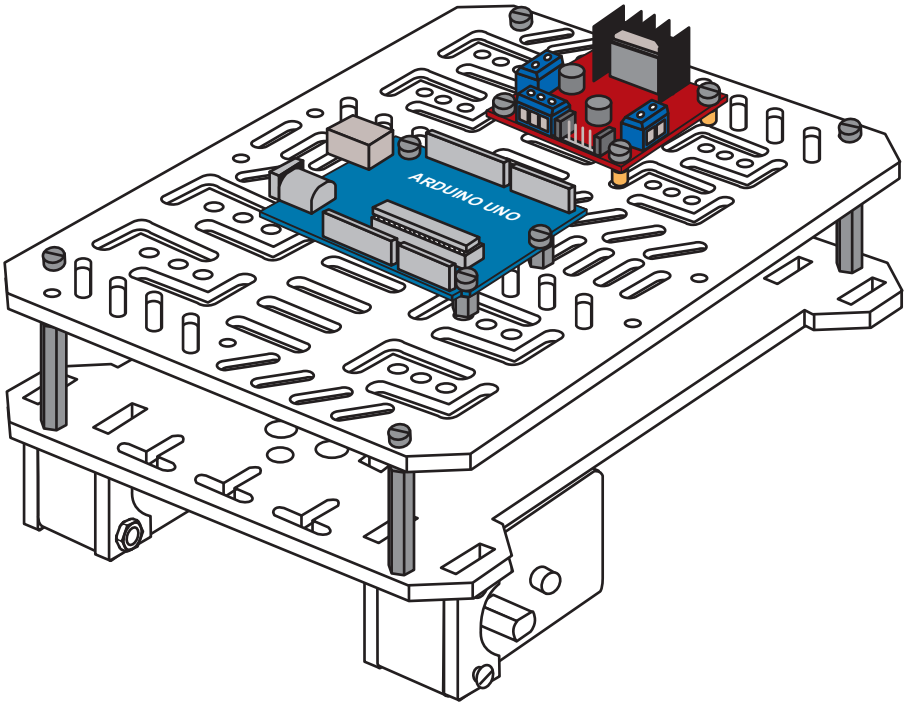
e) Atornillar el controlador de motores con 4 tornillos M3 LOG 464, 4 bulones de plástico y 4 tuercas M3 LOG 480.

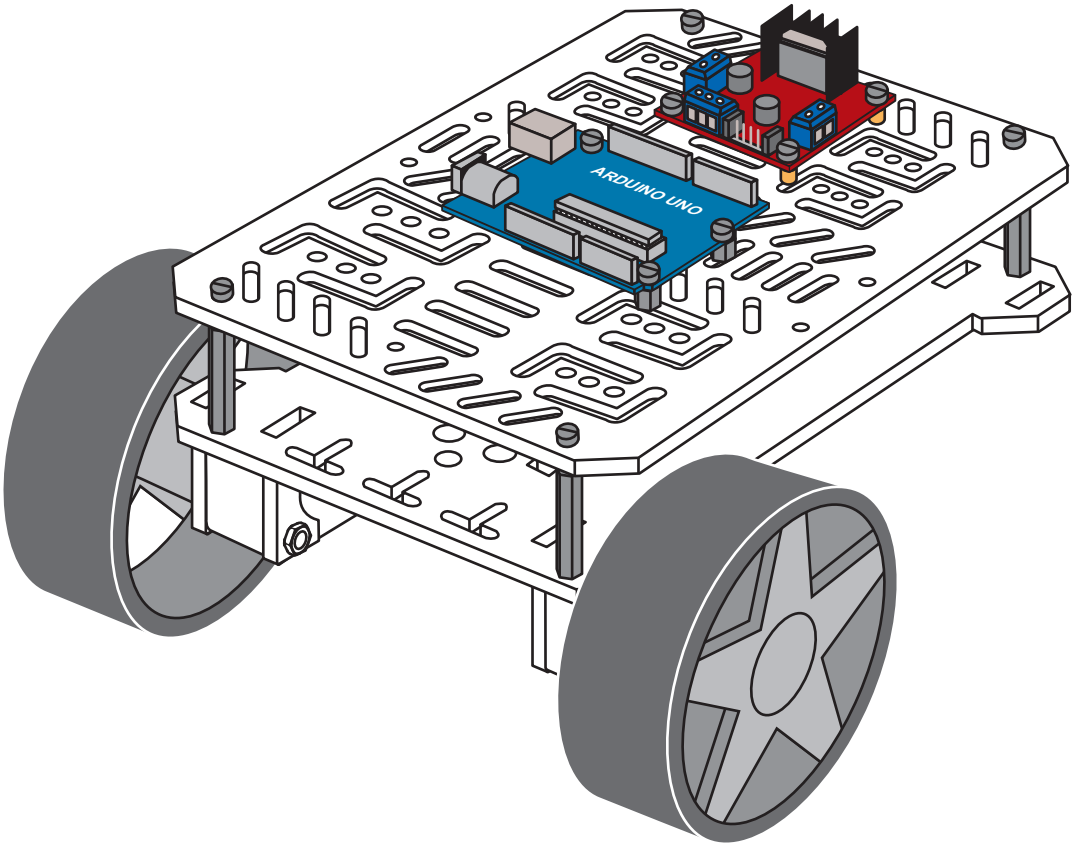


MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

6. CONSTRUCCIÓN CON DIRECCIÓN

a) Montar los motores por debajo de la plataforma móvil para elevarla del suelo.





MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

7. IMPRESIÓN EN 3D

- Descargar los archivos STL desde "Thingiverse"entrando en el perfil de "Micrologt" y buscando los diseños de "Mecanismos dirección coche por bluetooth LOG 4202A" o bien copiando el siguiente enlace **http://www.thingiverse.com/thing:1999249**

- Cargar los archivos STL en un programa para control de Impresoras 3D:

  - Repetier (software libre)
  - Cura (software BQ)
  - XYZware (software XYZ printing)

- Todas los programas anteriormente nombrados son gratuitos y fáciles de instalar. Una vez que interactuemos con el programa, debemos configurar las características de la impresora. Con estos programas podremos configurar los siguientes parámetros:

  - Tipo de plástico (PLA, ABS)
  - Temperatura entre 190º y 210º dependiendo del material
  - Relleno de las piezas
  - Espesor de las capas
  - Soportes de las estructuras

- Es importante tener la impresora calibrada, utilizar cinta de carrocero sobre la cama independientemente de si es caliente o fría y además aplicar pegamento de barra sobre la cama para facilitar la adherencia del plástico.

- Una vez terminada la pieza, despegarla de la cama utilizando una espátula, teniendo cuidado de no romper la pieza.

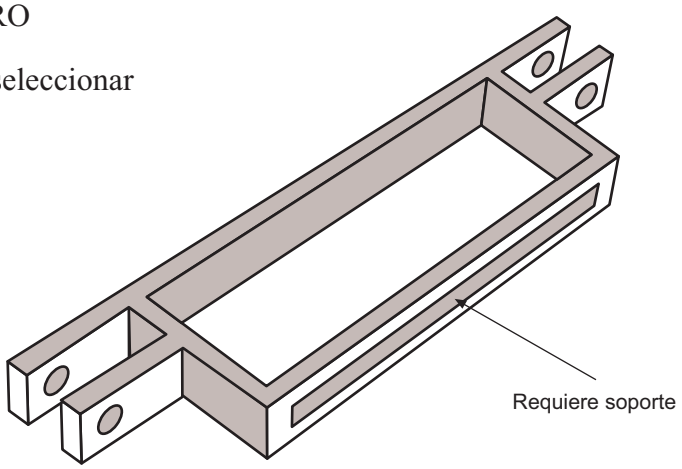
- Algunas piezas pueden requerir soportes. Imprimir utilizando esta herramienta y finalmente retirarlos. Para mejorar el acabado podemos utilizar una lima.

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologt.es; pedidos@micrologt.es

- Imprimir las 5 piezas en un impresora de 3D, que están en los tres archivos STL .

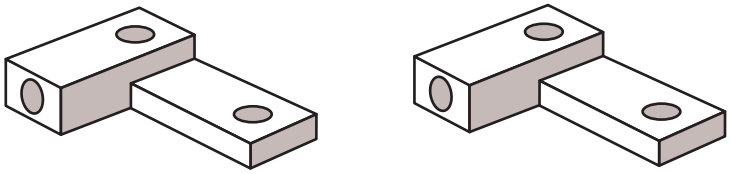
SOPORTE TREN DELANTERO

- Imprimir la pieza tumbada y seleccionar la opción de soportes



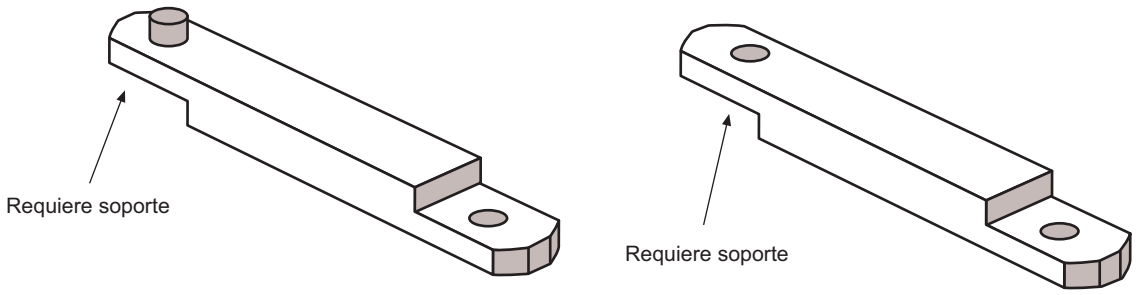
RÓTULAS (imprimir 2 veces)

- Imprimir las piezas tumbadas



BRAZOS DE DIRECCIÓN

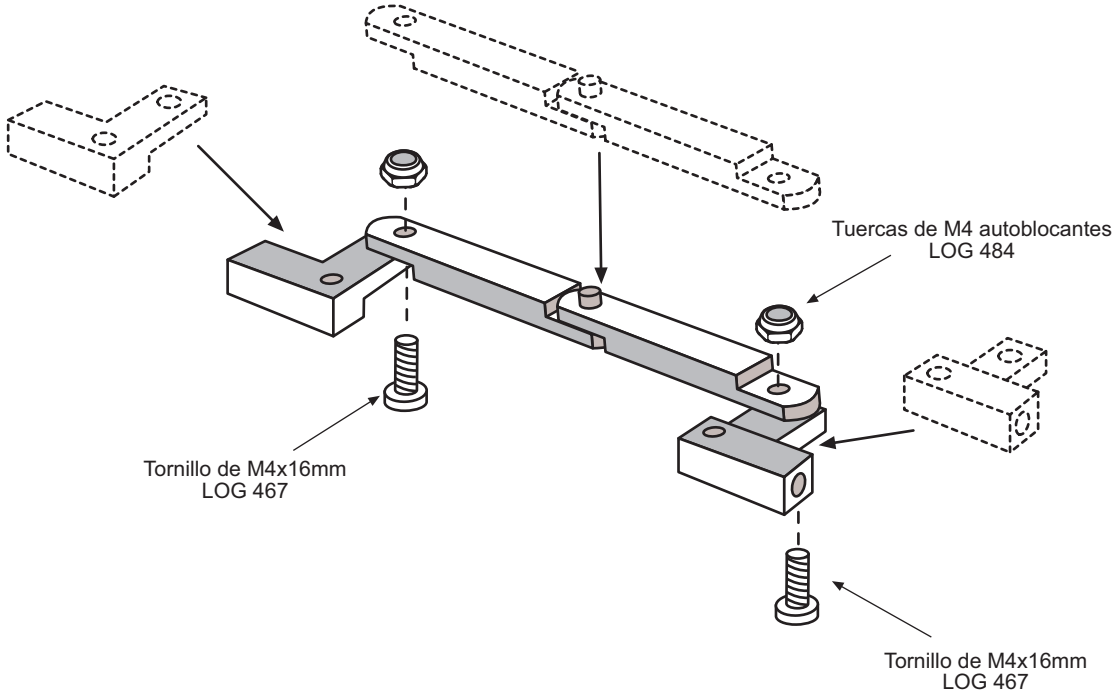
- Imprimir la pieza tumbada y selecionarla opción de soportes



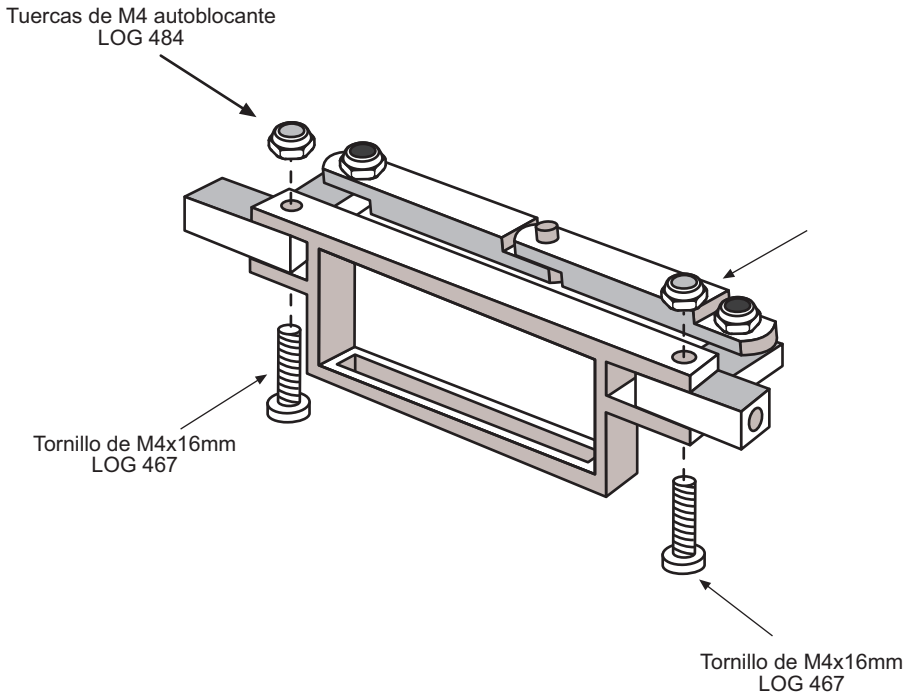
MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

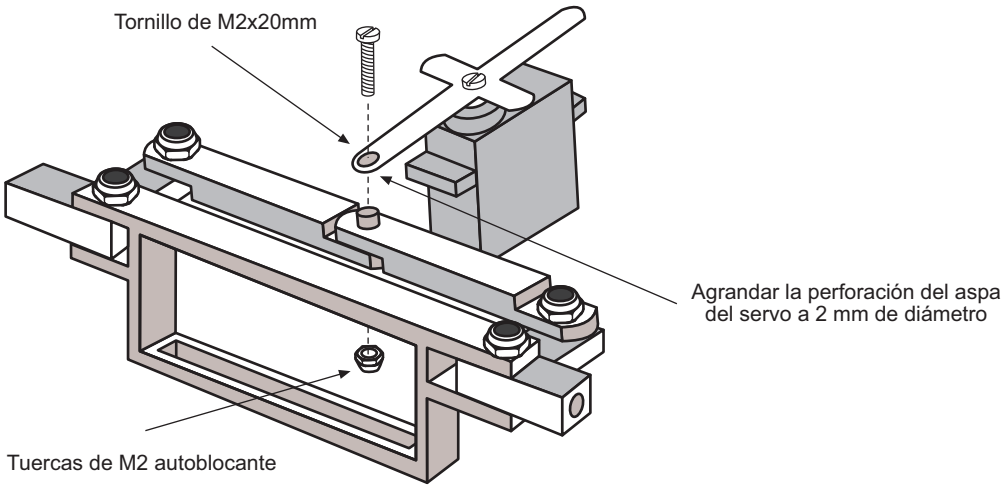
- Atornillar las rótulas a los brazos de la dirección.



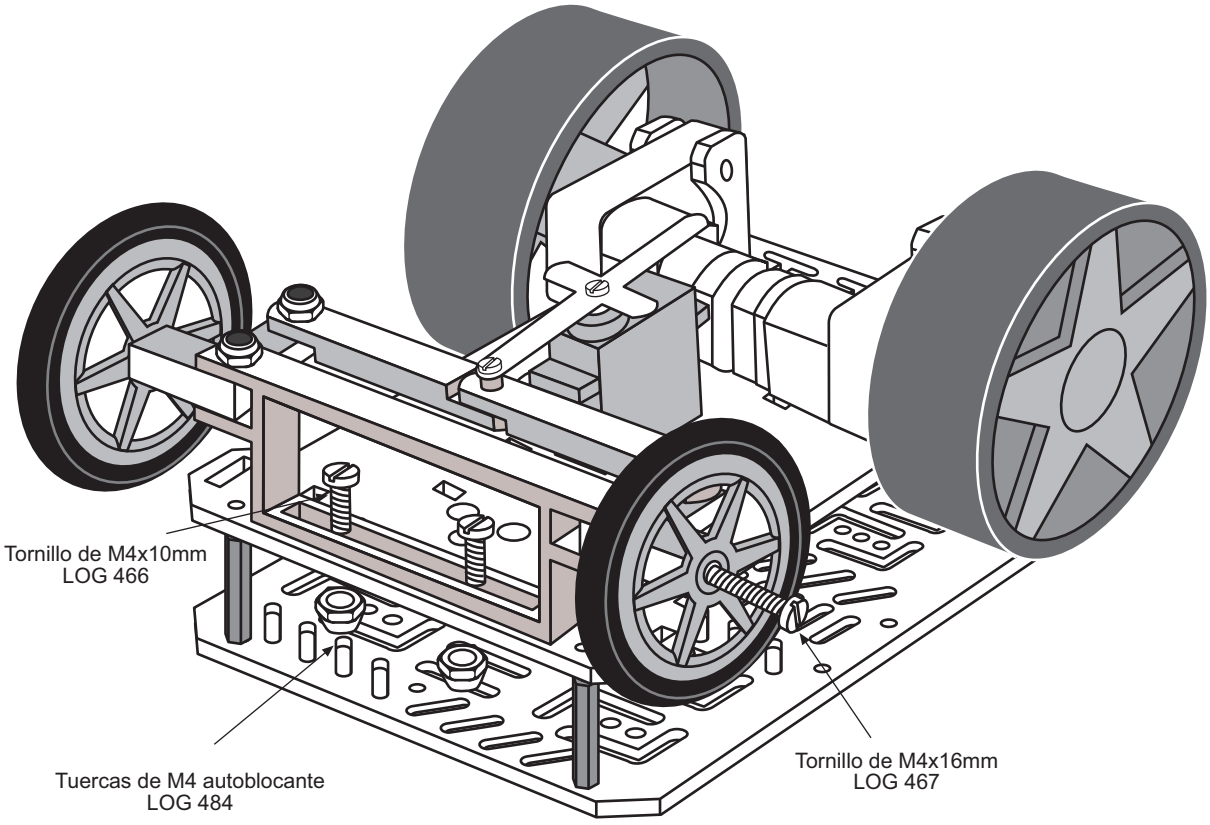
- Atornillar el conjunto de dirección al soporte del tren delantero.



- Atornillar el servo a los brazos de la dirección.
- El servomotor tiene que estar en posición de 90º.



- Atornillar el conjunto a la plataforma.
- Pegar el servomotor con pegamento termofusible a la plataforma.
- Atornillar las ruedas al tren delantero.

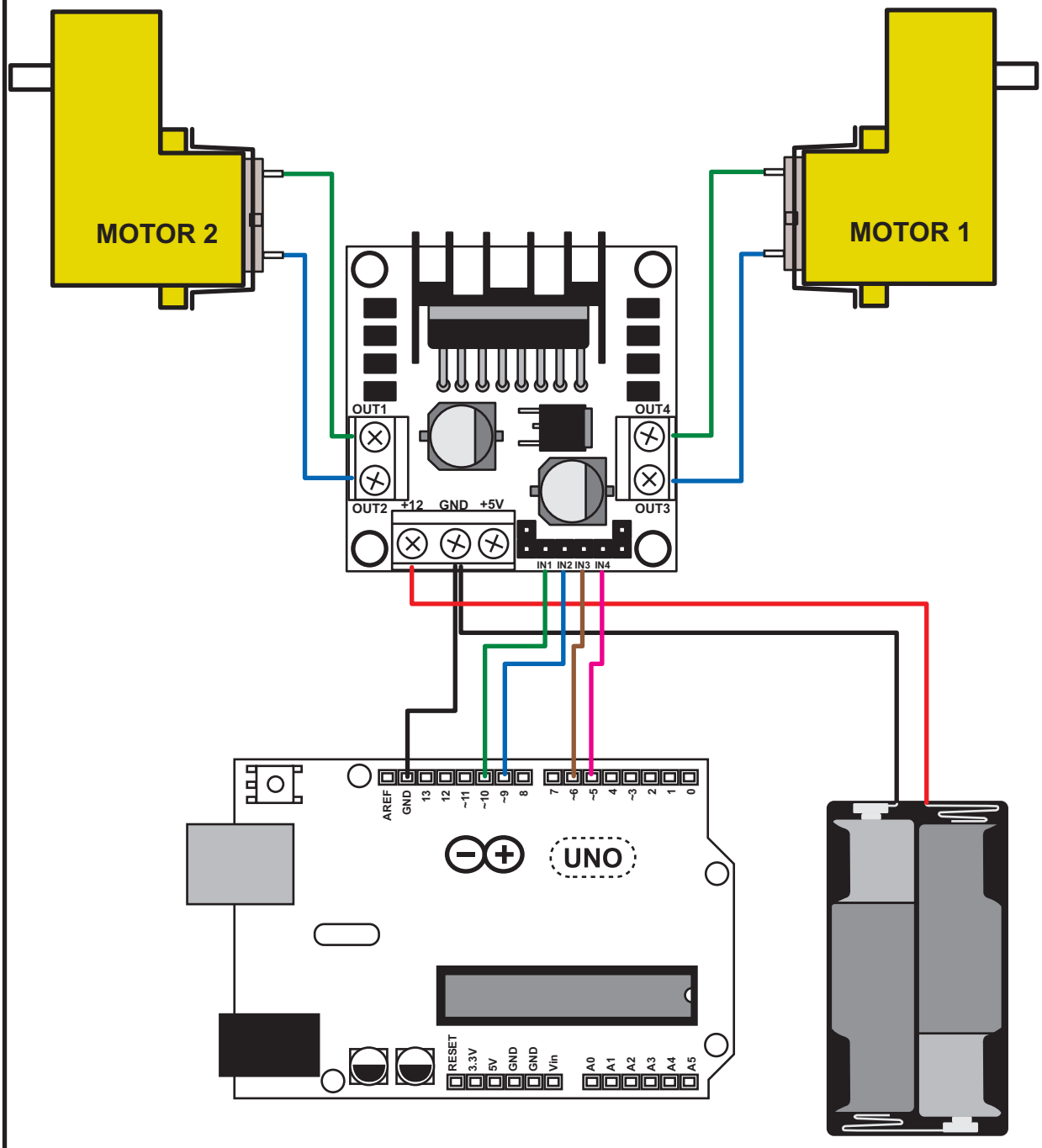


MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologg.es; pedidos@micrologg.es

8. 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).



MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

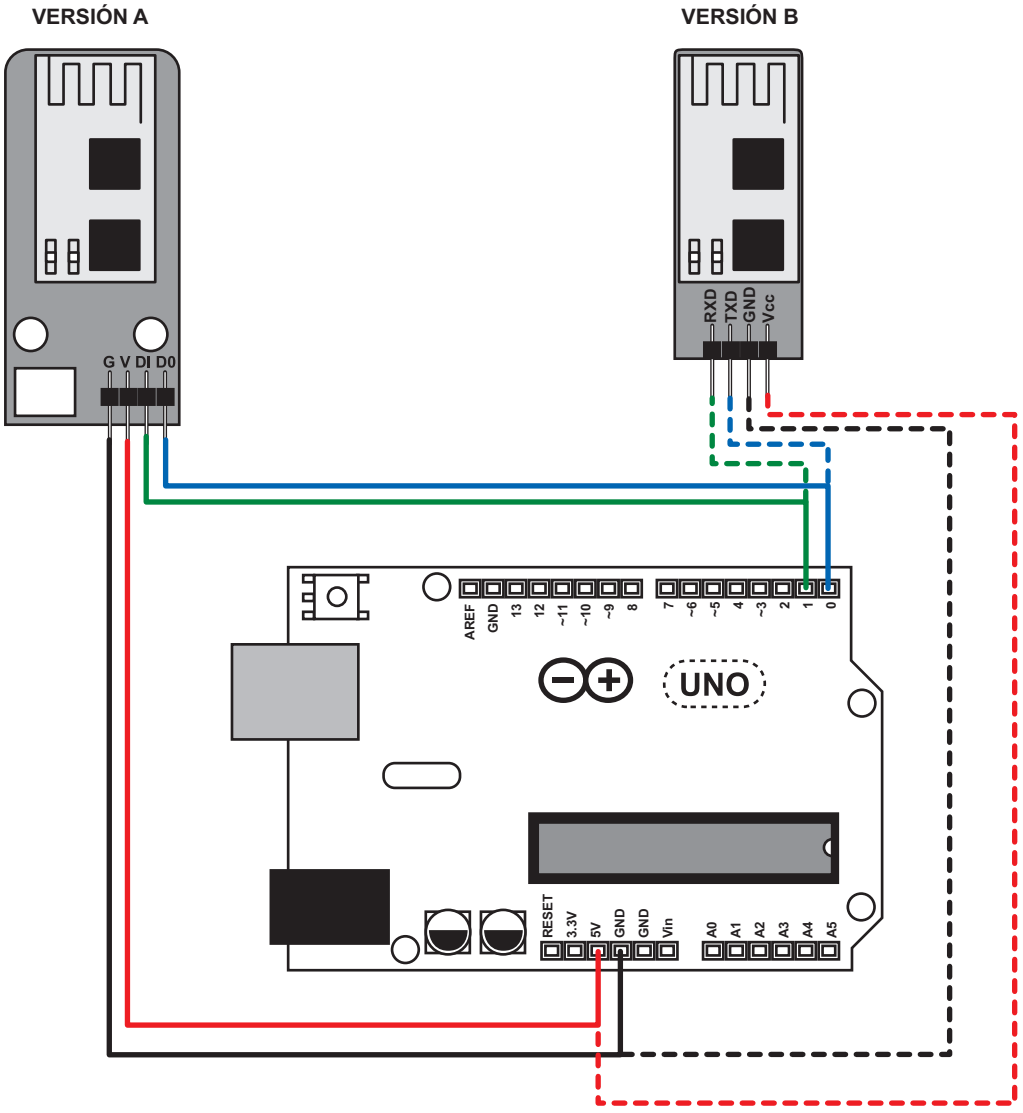
9. CONEXIÓN DEL SHIELD DE BLUETOOTH

- El shield bluetooth es el encargado de la comunicación bidireccional entre la tarjeta Arduino y el dispositivo móvil con el que controlaremos el coche.

- El shield bluetooth tiene 4 pines:

  - GND: Se conecta a un pin GND de Arduino
  - V: Se conecta al pin 5V del controlador de motores
  - TXD: Se conecta al pin RX de Arduino
  - RXD: Se conecta al pin TX de Arduino

- Los pines TX y RX son los que se van a encargar de la transmisión y recepción de datos

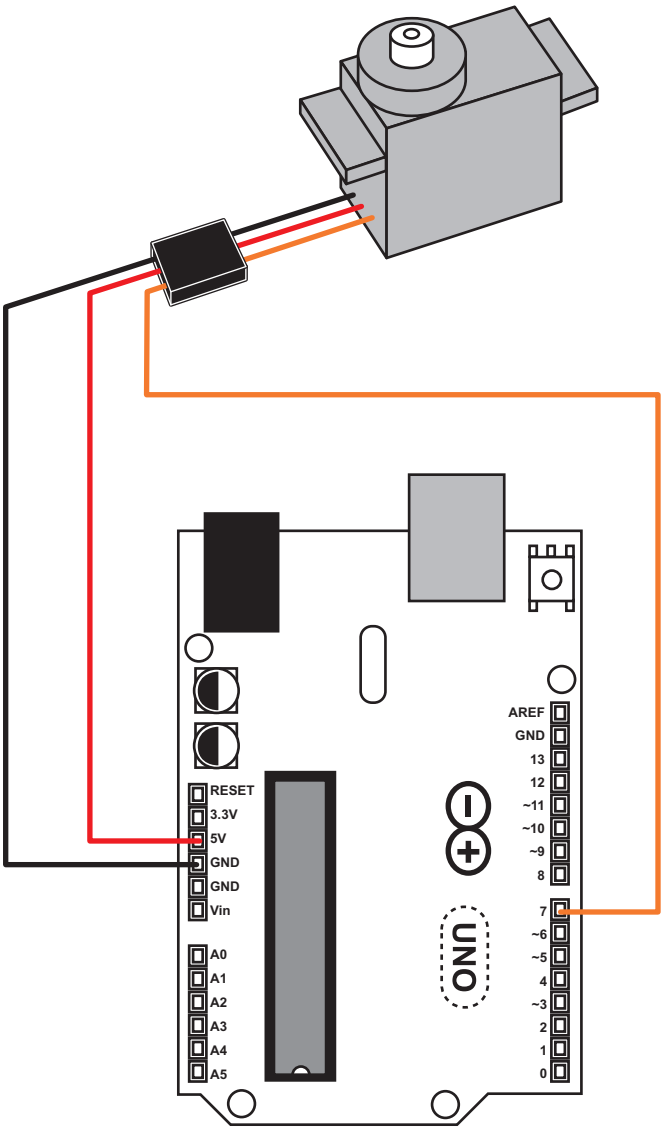


MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologs.es; pedidos@micrologs.es

10. CONEXIÓN DEL SERVOMOTOR

Conectamos el servomotor a la placa Arduino UNO con latiguillos, teniendo en cuenta la siguiente asignación de pines:

- Cable naranja o amarillo: salida 7
- Cable marrón o negro: GND
- Cable rojo: 5V



MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologg.es; pedidos@micrologg.es

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

11. PROGRAMACIÓN CON BOLA RODADURA

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", selecionar donde pone "Placa" y pinchar en "Arduino/Genuino UNO".

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

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

```
/*
Bluetooth

Programa para controlar los movimientos de un vehículo desde un dispositivo
móvil con bluetooth HC-06 LOG 4058.
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
*/


// Definimos los pines para los motores de la plataforma LOG 4080
int motor1Avance = 6;
int motor1Atras = 5;
int motor2Avance = 10;
int motor2Atras = 9;
// almacena el dato con la orden de movimiento que envía el móvil
int estado = 0;
int i = 0;

void setup() {
    pinMode(motor1Avance, OUTPUT);
       //pines del sistema motriz LOG 48 en modo salida
    pinMode(motor1Atras, OUTPUT);
    pinMode(motor2Avance, OUTPUT);
    pinMode(motor2Atras, OUTPUT);
    analogWrite(motor1Avance, 0); //el coche empieza en estado de parado
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 0);
    analogWrite(motor2Atras, 0);
    Serial.begin(9600);
}

void loop() {

//activasmos el puerto serie para la lectura de datos
 if(Serial.available() > 0)
 {
      estado = Serial.read();
 }

// leemos un dato recibido a traves del dispositivo HC-06 LOG 4058
 Serial.println(estado);
```

```
 // si recibimos un c, paramos el vehículo
 if (estado == 'c')
 {
    analogWrite(motor1Avance, 0);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 0);
    analogWrite(motor2Atras, 0);
    Serial.println("parado");
 }

// si recibimos una a el coche LOG 4202 circula recto de frente
else if (estado == 'a')
 {
    analogWrite(motor1Avance, 255);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 248);
    analogWrite(motor2Atras, 0);
    Serial.println("adelante");
 }

// si recibimos una d el coche LOG 4202 gira hacia la derecha
 else if (estado == 'd')
 {
    analogWrite(motor1Avance, 100);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 200);
    analogWrite(motor2Atras, 0);
    Serial.println("derecha");
 }

// si recibimos una b el coche LOG 4202 gira hacia la izquierda
 else if (estado == 'b')
 {
    analogWrite(motor1Avance, 200);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 100);
    analogWrite(motor2Atras, 0);
    Serial.println("izquierda");

 }

// si recibimos una e el coche LOG 4202 retrocede
  else if (estado == 'e')
 {
    analogWrite(motor1Avance, 0);
    analogWrite(motor1Atras, 255);
    analogWrite(motor2Avance, 0);
    analogWrite(motor2Atras, 255);
    Serial.println("atras");
 }
Serial.println(i++);
}
```

// Al cargar el programa, hay que desconectar el bluetooth para que no interfiera en la
// transmisión de datos. Al finalizar la carga volver a conectar el bluetooth.

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologs.es; pedidos@micrologs.es

12. PROGRAMACIÓN CON BOLA RODADURA 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*".

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es

13. PROGRAMACIÓN CON SERVOMOTOR

```
/*
Bluetooth

Programa para controlar los movimientos de un vehículo desde un dispositivo
móvil con bluetooth HC-06 LOG 4058.
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
*/
 #include <Servo.h>
Servo myservo;
int motor1Avance = 6; //Definimos los pines para los motores
int motor1Atras = 5;
int motor2Avance = 11;
int motor2Atras = 9;
int estado = 0; //almacena el dato con la orden de movimiento que envía el móvil
int i = 0;
void setup() {
    myservo.attach(7);
    pinMode(motor1Avance, OUTPUT);   //pines de los motores LOG 48 en modo salida
    pinMode(motor1Atras, OUTPUT);
    pinMode(motor2Avance, OUTPUT);
    pinMode(motor2Atras, OUTPUT);
    analogWrite(motor1Avance, 0); //el coche empieza en estado de parado
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 0);
    analogWrite(motor2Atras, 0);
    Serial.begin(9600, SERIAL_8N1);
}

void loop() {
 if(Serial.available() > 0) //activasmos el puerto serie para la lectura de datos
 {
      estado = Serial.read();
 }

// leemos un dato recibido a traves del dispositivo HC-06 LOG 4058
 Serial.println(char(estado));
 if (estado == 'c') // si recibimos un c, paramos el vehículo
 {
    analogWrite(motor1Avance, 0);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 0);
    analogWrite(motor2Atras, 0);
    Serial.println("parado");
    myservo.write(90);
 }
 else if (estado == 'a')  //si recibimos una "a" el coche circula recto de frente
 {
    analogWrite(motor1Avance, 255);
    analogWrite(motor1Atras, 0);
    analogWrite(motor2Avance, 255);
    analogWrite(motor2Atras, 0);
    Serial.println("adelante");
    myservo.write(90);
 }
```

(continuación)

```
  else if (estado == 'd')   //si recibimos una "d" el coche gira hacia la derecha
   {
      analogWrite(motor1Avance, 255);
      analogWrite(motor1Atras, 0);
      analogWrite(motor2Avance, 255);
      analogWrite(motor2Atras, 0);
      Serial.println("derecha");
      myservo.write(0);
   }
   else if (estado == 'b')  //si recibimos una "b" el coche gira hacia la izquierda
   {
      analogWrite(motor1Avance, 255);
      analogWrite(motor1Atras, 0);
      analogWrite(motor2Avance, 255);
      analogWrite(motor2Atras, 0);
      Serial.println("izquierda");
      myservo.write(180);

   }
    else if (estado == 'e')  //si recibimos una "e" el coche retrocede
   {
      analogWrite(motor1Avance, 0);
      analogWrite(motor1Atras, 255);
      analogWrite(motor2Avance, 0);
      analogWrite(motor2Atras, 255);
      Serial.println("atras");
      myservo.write(90);
   }
   else if (estado == 'i')  //si recibimos una "i" el coche retrocede a la derecha
   {
      analogWrite(motor1Avance, 0);
      analogWrite(motor1Atras, 255);
      analogWrite(motor2Avance, 0);
      analogWrite(motor2Atras, 255);
      Serial.println("atras_derecha");
      myservo.write(45);
   }
   else if (estado == 'o')  //si recibimos una "o" el coche retrocede a la izquierda
   {
      analogWrite(motor1Avance, 0);
      analogWrite(motor1Atras, 255);
      analogWrite(motor2Avance, 0);
      analogWrite(motor2Atras, 255);
      Serial.println("atras_izquierda");
      myservo.write(135);
   }

   else if (estado == 'j')  // si recibimos una "j" el coche avanza a la derecha
   {
      analogWrite(motor1Avance, 255);
      analogWrite(motor1Atras, 0);
      analogWrite(motor2Avance, 255);
      analogWrite(motor2Atras, 0);
      Serial.println("delante_derecha");
      myservo.write(45);
   }
   else if (estado == 'p')  // si recibimos una "p" el coche avanza a la izquierda
   {
      analogWrite(motor1Avance, 255);
      analogWrite(motor1Atras, 0);
      analogWrite(motor2Avance, 255);
      analogWrite(motor2Atras, 0);
      Serial.println("delante_izquierda");
      myservo.write(135);
   }

  }
```

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.micrologg.es; pedidos@micrologg.es

14. 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, disminuiremos 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.

15. 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/4202.zip

16. PRUEBAS

- Aumentar o disminuir la velocidad de los motores.

MICRO-LOG TECNOLOGÍA Y SISTEMAS S.L. C/ Andrés Obispo 37, 28043 Madrid; Tfno: 91 759 59 10; Fax: 91 759 54 80; www.microlog.es; pedidos@microlog.es