

TEMA 1 – HACER AL COCHE MOVERSE

Puntos de este tema:

Aprender:



Aprender a usar Arduino IDE



Hacer al coche moverse subiendo un programa

Preparaciones:



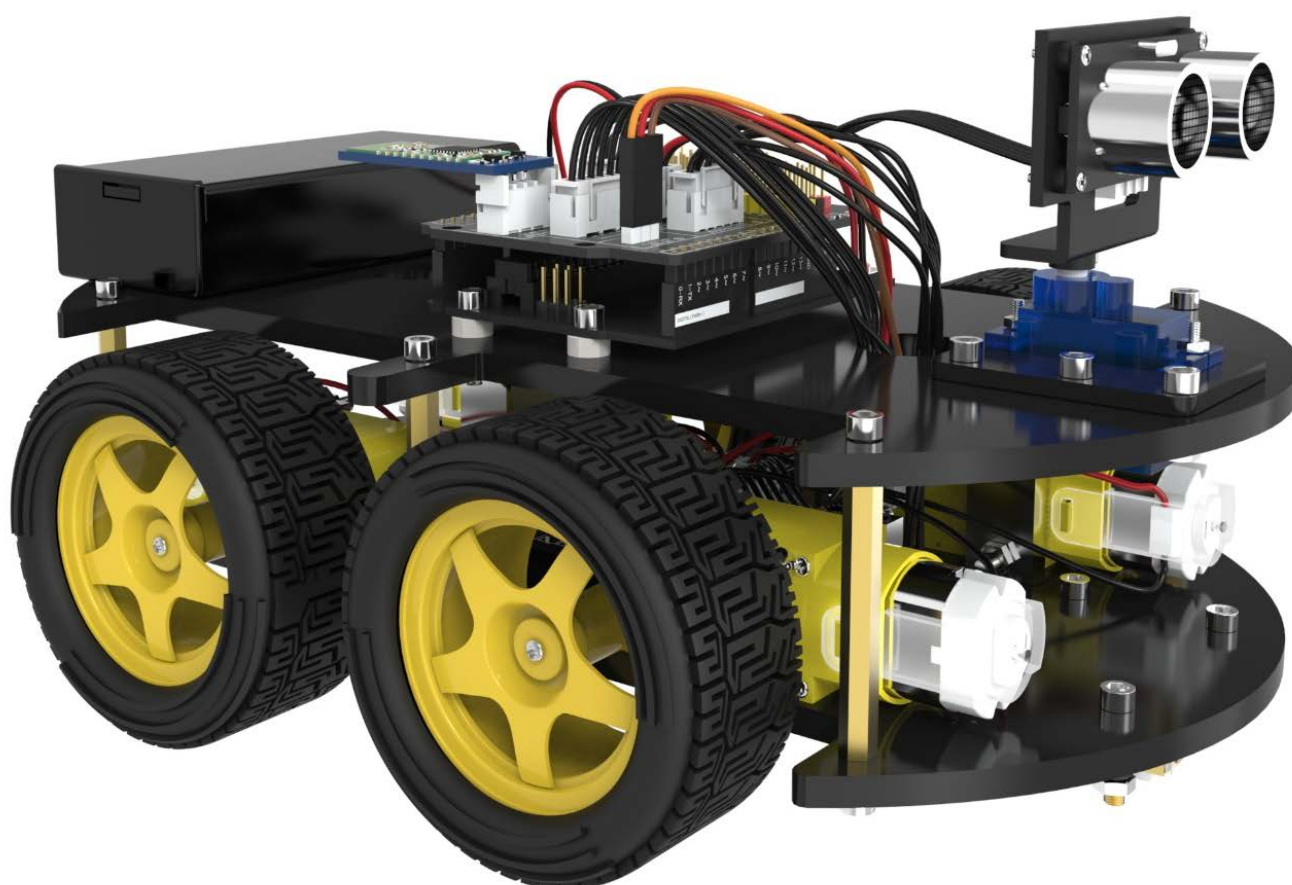
Un coche (con batería)



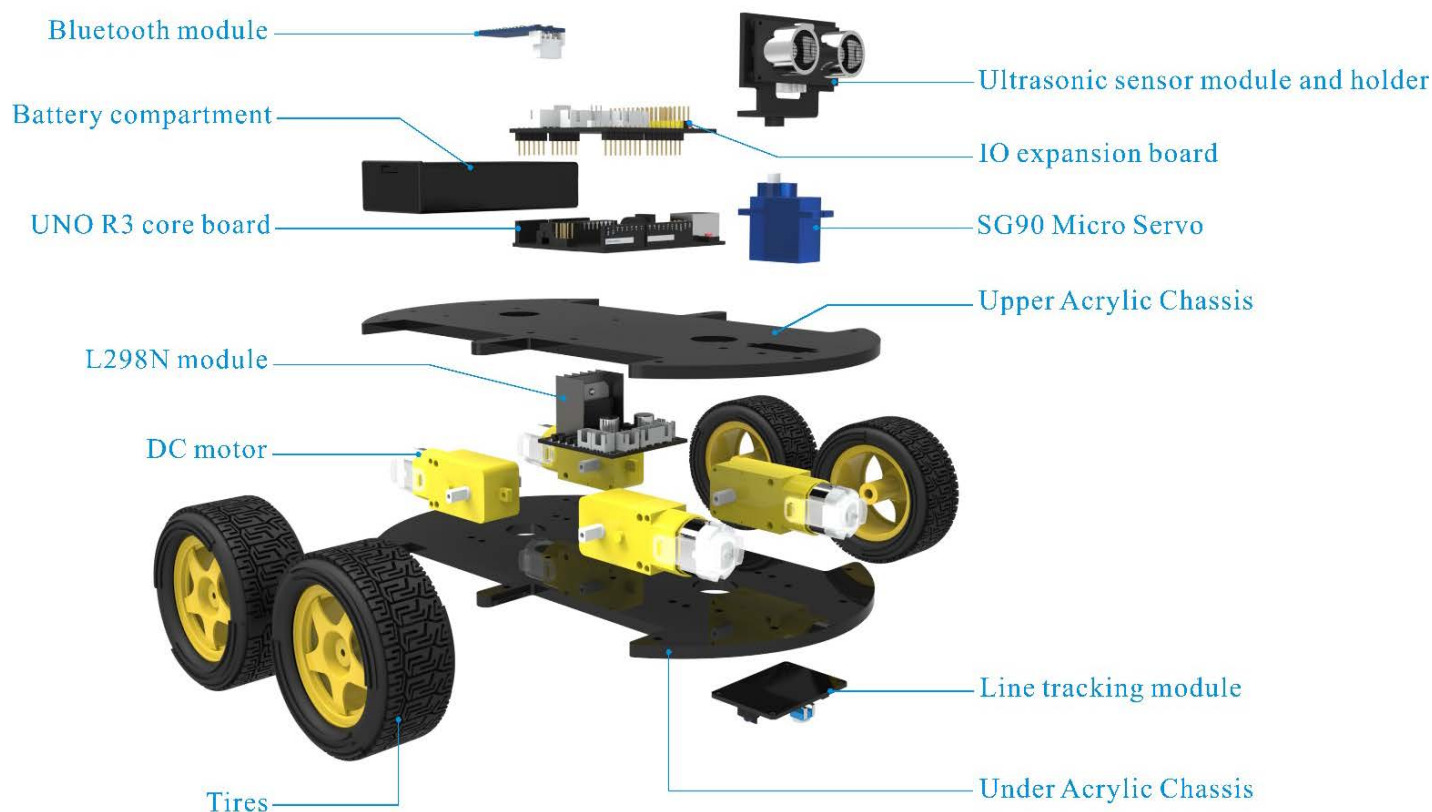
Un cable USB

I . Introducción al coche

Este kit es un kit de vehículos extremadamente flexible diseñado especialmente para la educación, la competencia y el entretenimiento. El panel superior del kit es directamente compatible con el motor de dirección de 9 gramos. También lleva un sensor ultrasónico, una batería y una serie de agujeros fijos para facilitar la instalación de varios sensores. Este es un robot muy divertido y versátil que cumple con los objetivos de aprendizaje y producción. Con él, puede implementar diversas ideas interesantes, como Bluetooth y control remoto por infrarrojos, la eliminación automática de obstáculos y la inspección de línea. Vamos a describir el pequeño vehículo que nos acompañará durante mucho tiempo en el futuro.



Each parts of the car is as below:



Funciones de cada componente:

1. Soporte de la batería con un interruptor: proveer la fuente de alimentación para el vehículo
2. Motor eléctrico + rueda: Dar movimiento al vehículo
3. placa de acrílico: el chasis del coche
4. Placa de conducción del motor L298N: acciona el motor para darle movimiento
5. UNO controlador bordo: el cerebro del coche, controla todas las partes
6. Tarjeta de expansión del sensor V5: combinada con la UNO, hacen que la conexión sea más fácil
7. Servo y plataforma de nube: permiten que el sensor de distancia GP2Y0A21 gire 180 grados
8. Módulo ultrasónico del sensor: medida de la distancia y evita los obstáculos
9. Módulo de seguimiento de línea: sensor blanco y negro para el reconocimiento de los carriles blancos y negros
10. Receptor infrarrojo y teledirigido: proporcionan la función teledirigida infrarroja
11. Módulo Bluetooth: proporciona la función de control de Bluetooth

II. Cargar programa

Cada movimiento del coche esta controlado por un programa, por lo que será necesario tenerlo instalado y correctamente configurado. Usaremos el programa Arduino Software IDE (Integrated Development Environment) como herramienta de programación.

PASO 1: Vaya a <https://www.arduino.cc/en/Main/Software> y busque la siguiente página.



La versión disponible en este sitio web suele ser la última version disponible, y la versión actual puede ser más reciente que la versión de la imagen.

PASO 2: Descargue el software de desarrollo que sea adecuado para su sistema operativo.

En este tutorial utilizaremos Windows como ejemplo.

Puede instalarlo utilizando el paquete de instalación EXE o el paquete verde.

Windows Installer
Windows ZIP file for non admin install

Las siguientes serán las instrucciones para la instalación EXE

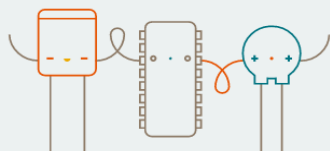
Pulse en “Windows Installer”

Windows Installer

Pulse el botón “JUST DOWNLOAD” para descargar el software.

Support the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). Learn more on how your contribution will be used.



SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **8,808,272** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3

\$5

\$10

\$25

\$50

OTHER

JUST DOWNLOAD

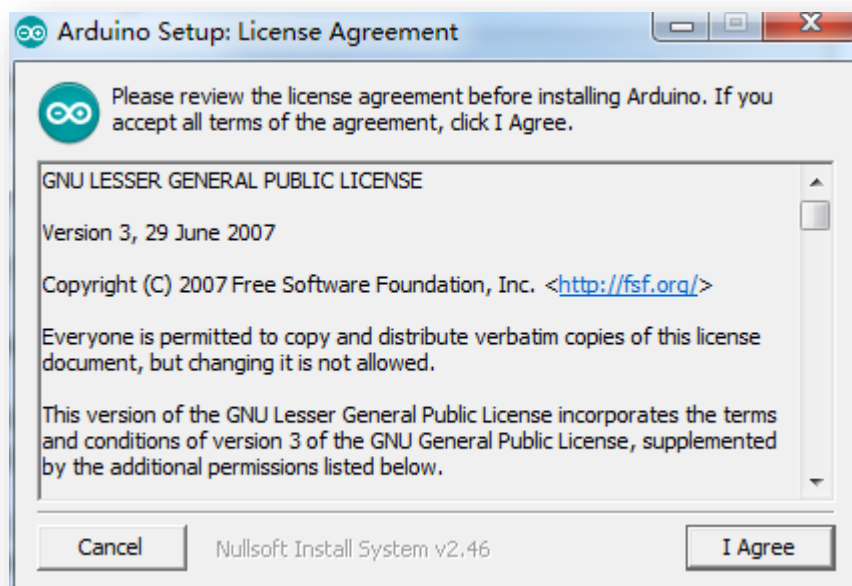
CONTRIBUTE & DOWNLOAD

JUST DOWNLOAD

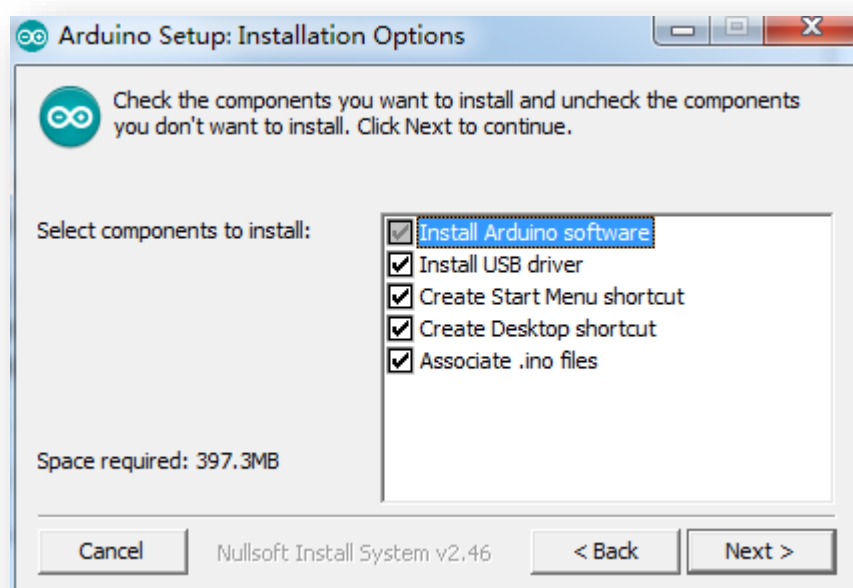
El archivo a descargar

arduino-1.8.2-windows.exe

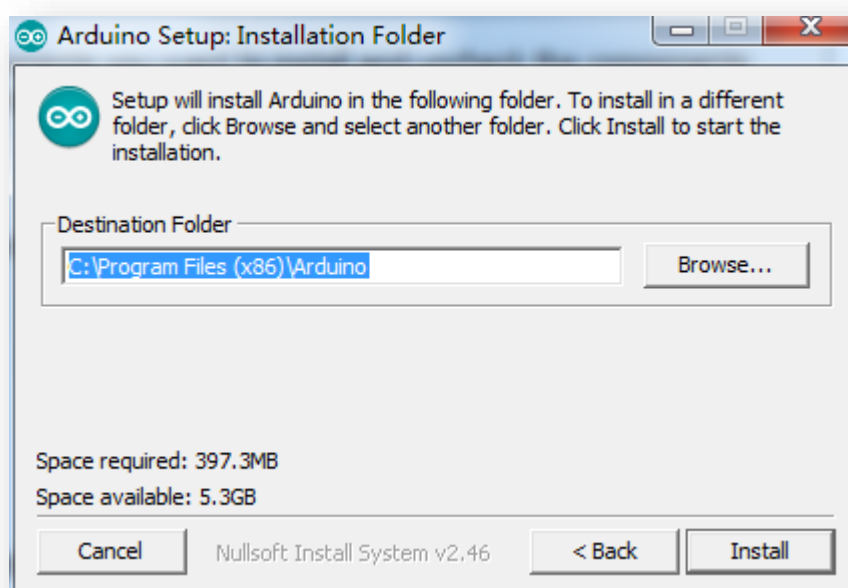
Estos están disponibles en los materiales que ofrecemos, y las versiones de nuestros materiales son las últimas versiones más actualizadas cuando se realizó este tutorial.



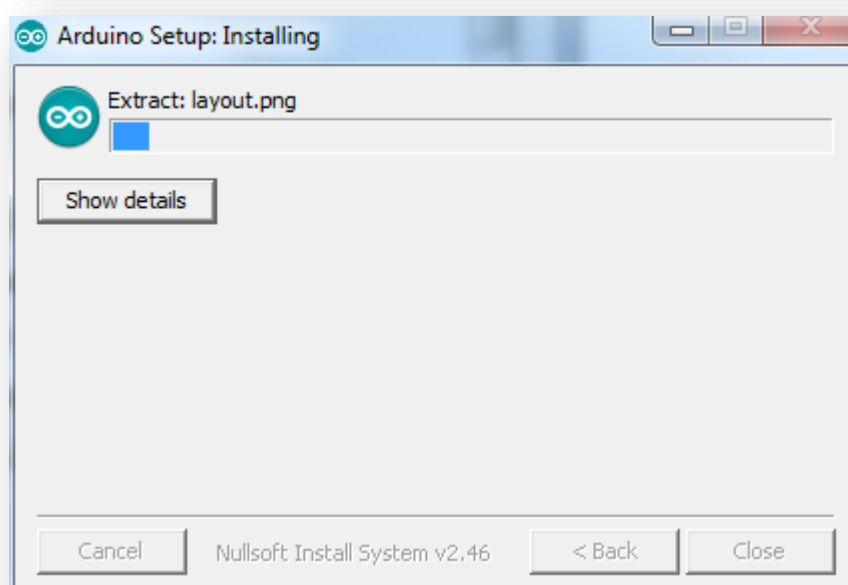
Selecciona "I Agree" para acceder a la siguiente ventana



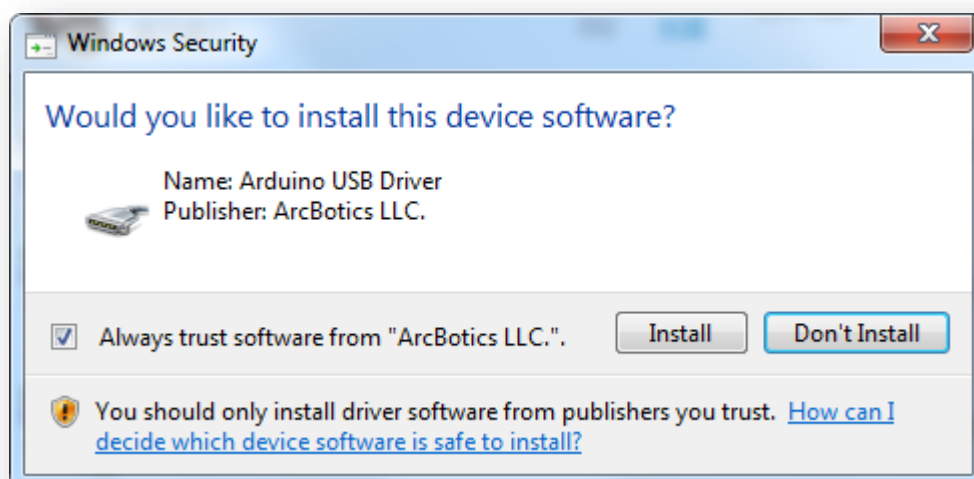
Pulse en "Next" para continuar



Pulse Install para iniciar la instalación



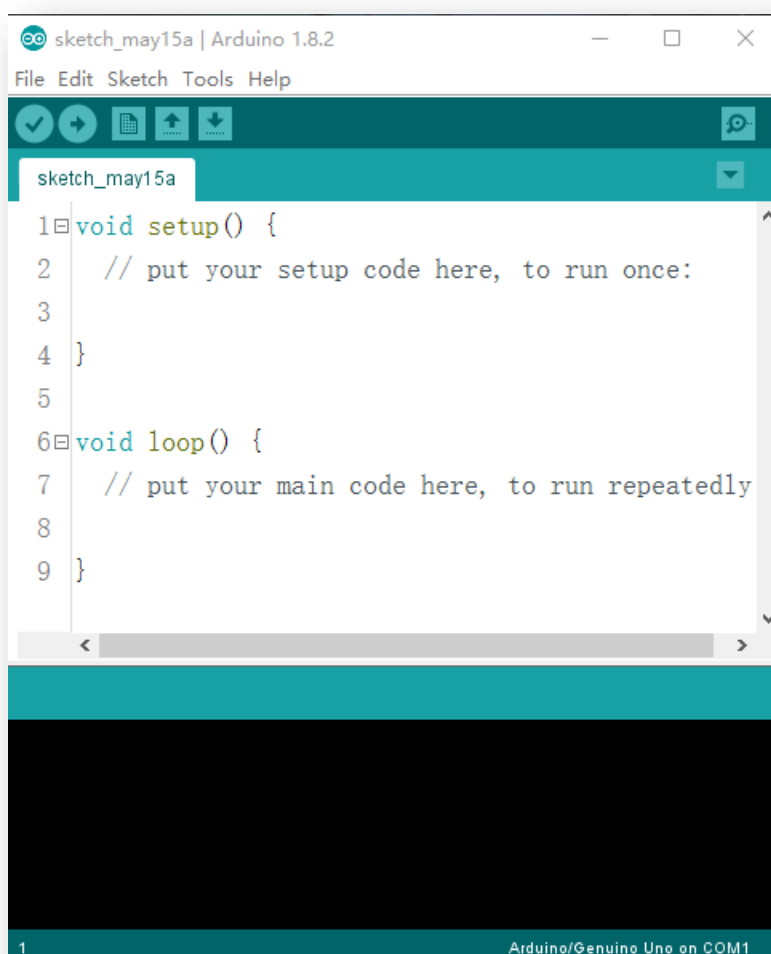
Finalmente, aparece la siguiente interfaz, debes elegir Instalar para asegurar la corrección del desarrollo

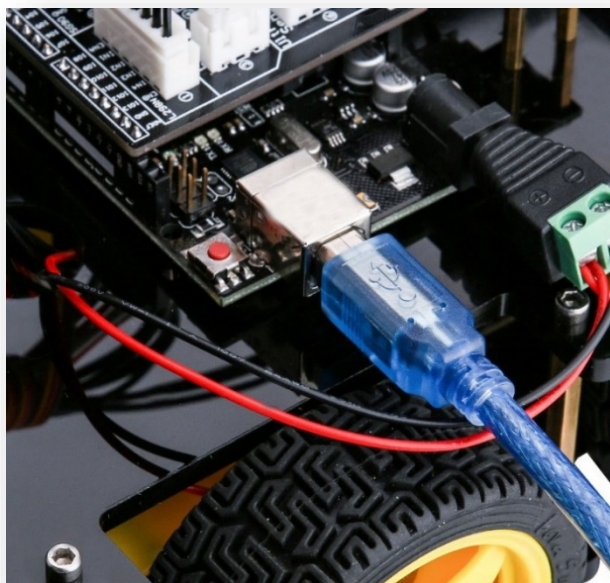


A continuación el siguiente icono aparecerá en el escritorio.



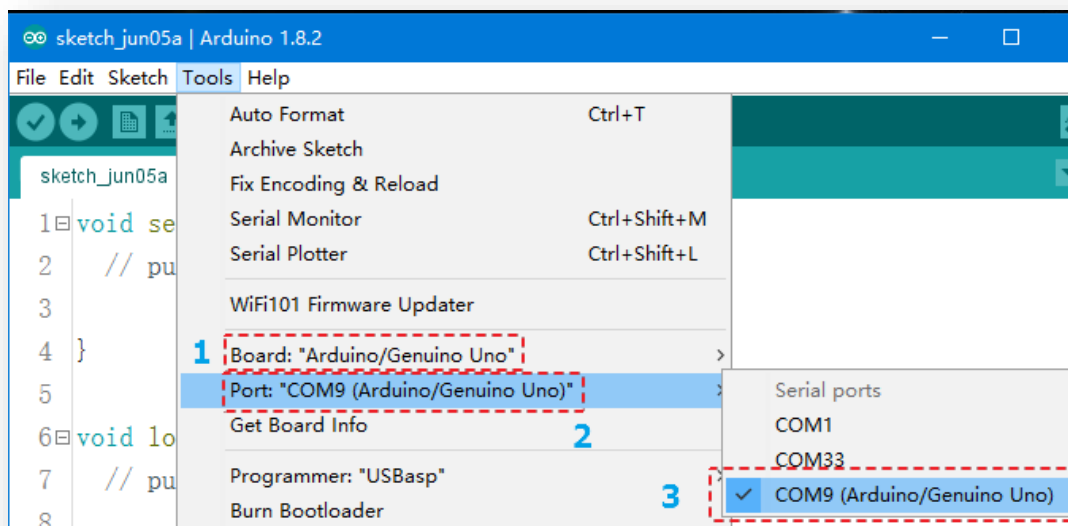
Pulse doble click para entrar en el entorno de desarrollo



PASO3: Conectar el coche al ordenador

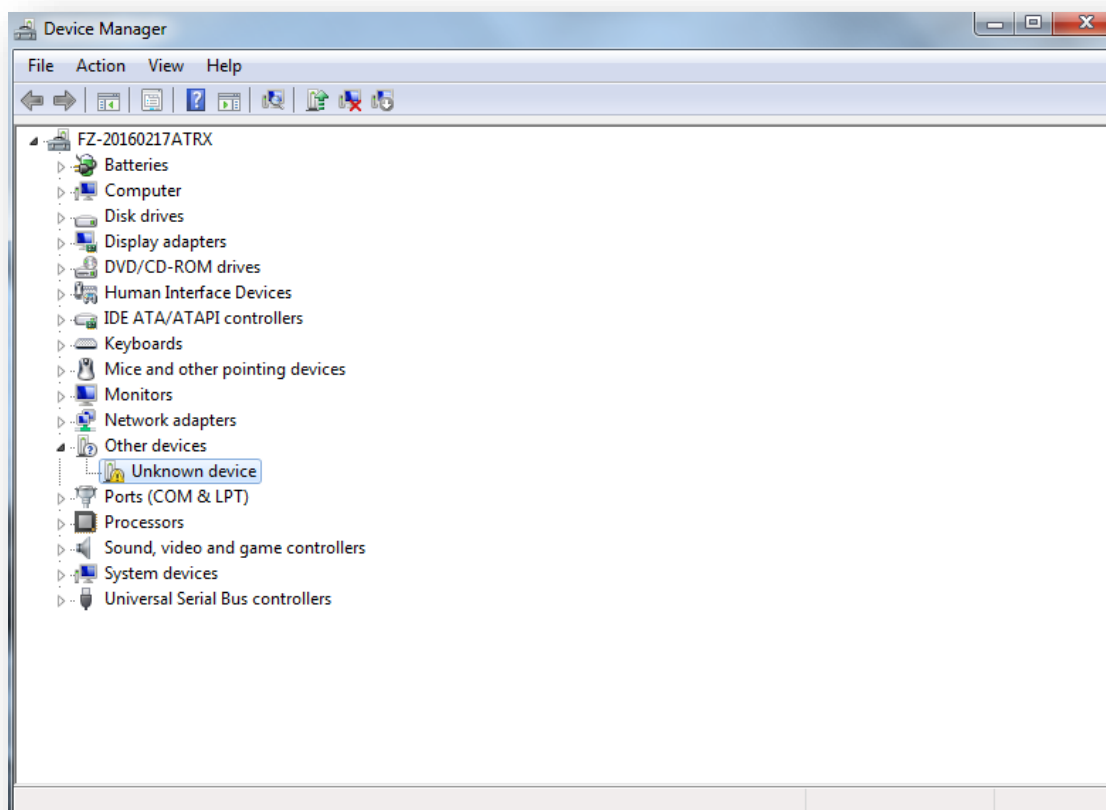
PASO 4: Abrir la siguiente ruta Arduino IDE. Select "Tool" → "Board:" → "Arduino/Genuino Uno".
Select "Tool" → "Port:" → "COM (Arduino/Genuino Uno)".

Cada placa Arduino Uno tiene un número COM diferente en el mismo equipo y usualmente el número COM con el nombre de sufijo "(Arduino / Genuino Uno)" en Arduino 1.8.2. Debe elegir el número COM de la pantalla real.

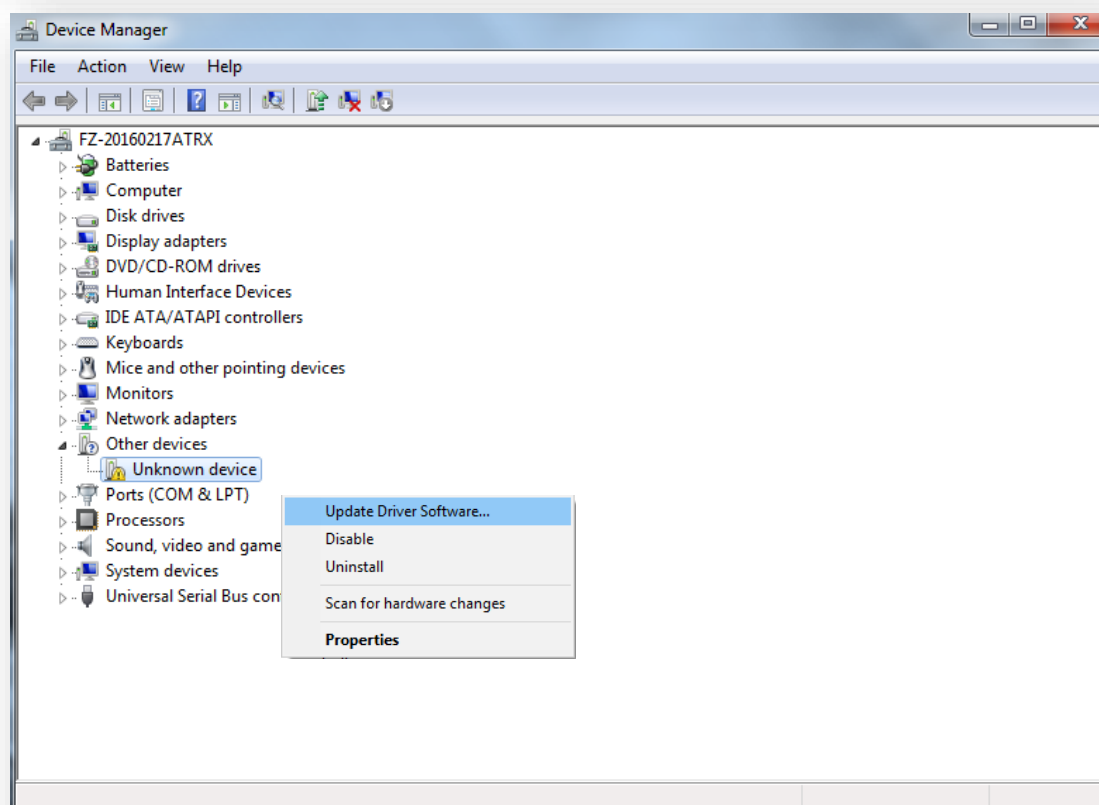


Si puedes ver el puerto “COM (Arduino/Genuino Uno)”, significa que el vehiculo se ha conectado correctamente al ordenador. En ese caso puedes pasar al paso 5 directamente. En otro caso necesitarás instalar el driver de la siguiente manera.

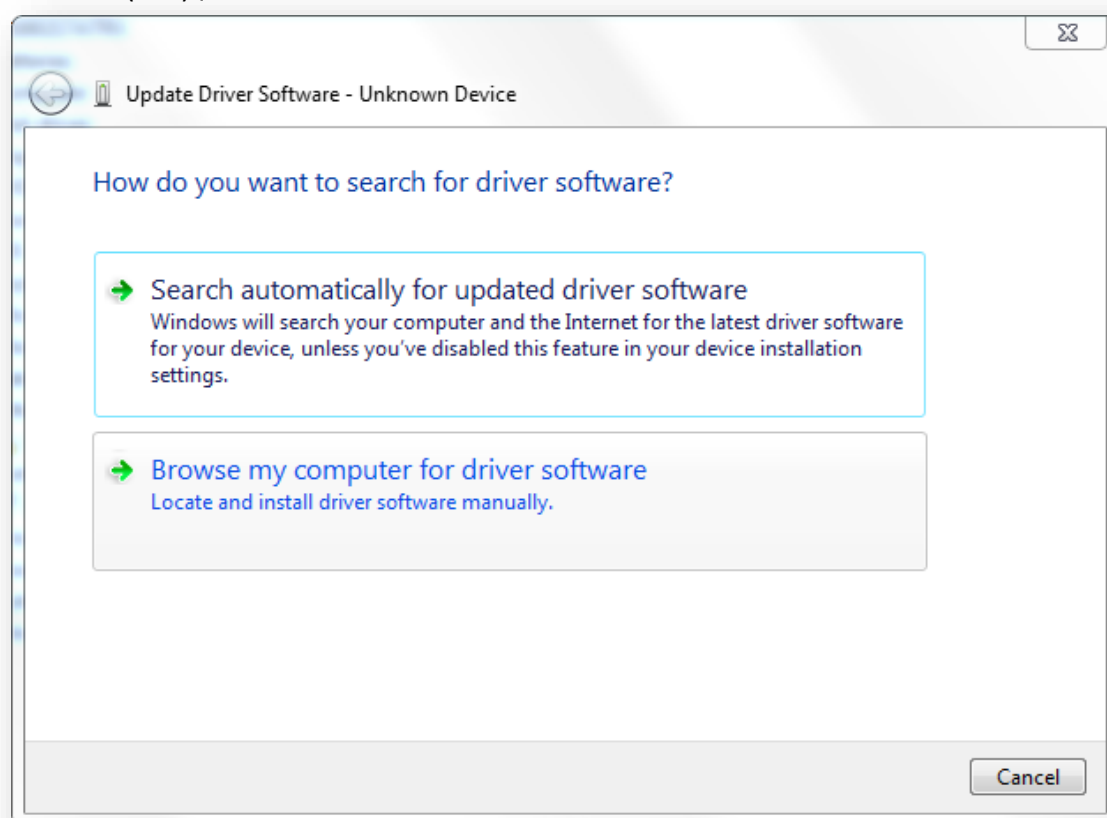
Abrir Device Manager pulsando boton derecho en Mi PC—Panel de administración— Dispositivos



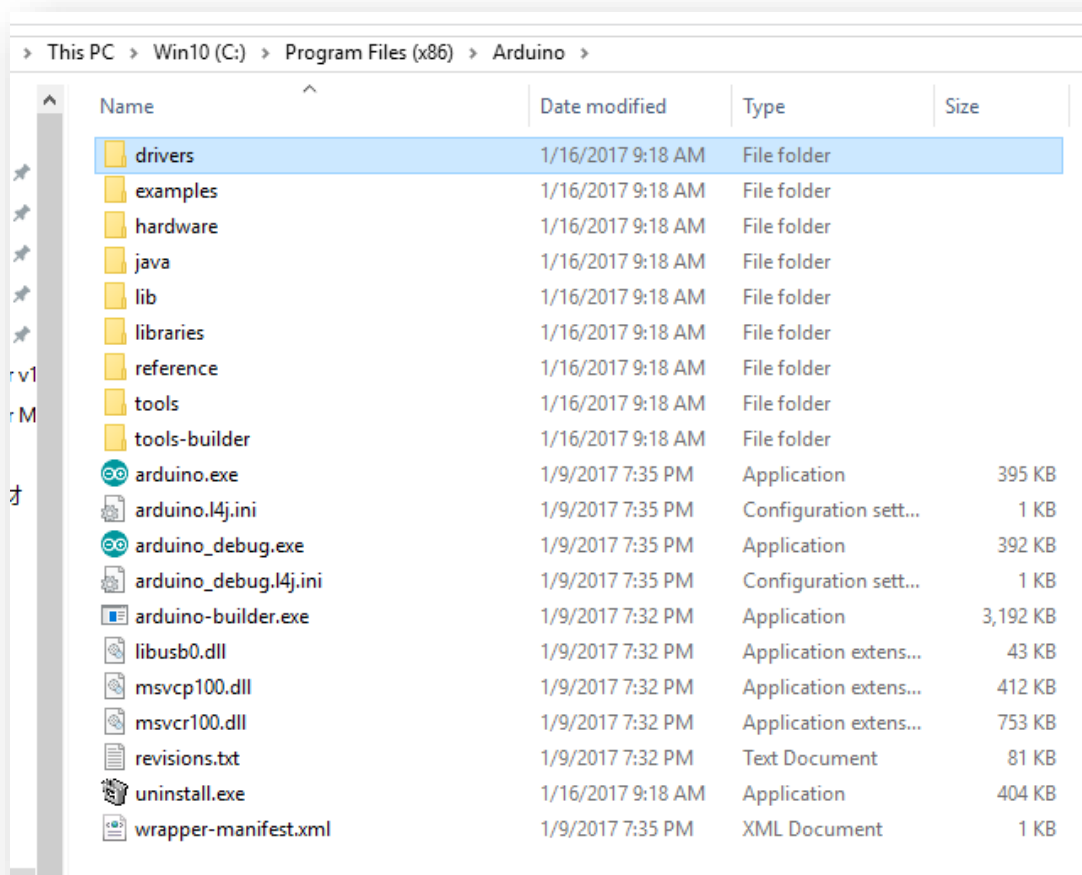
Boton derecho en dispositivo desconocido-----actualizar software dispositivo



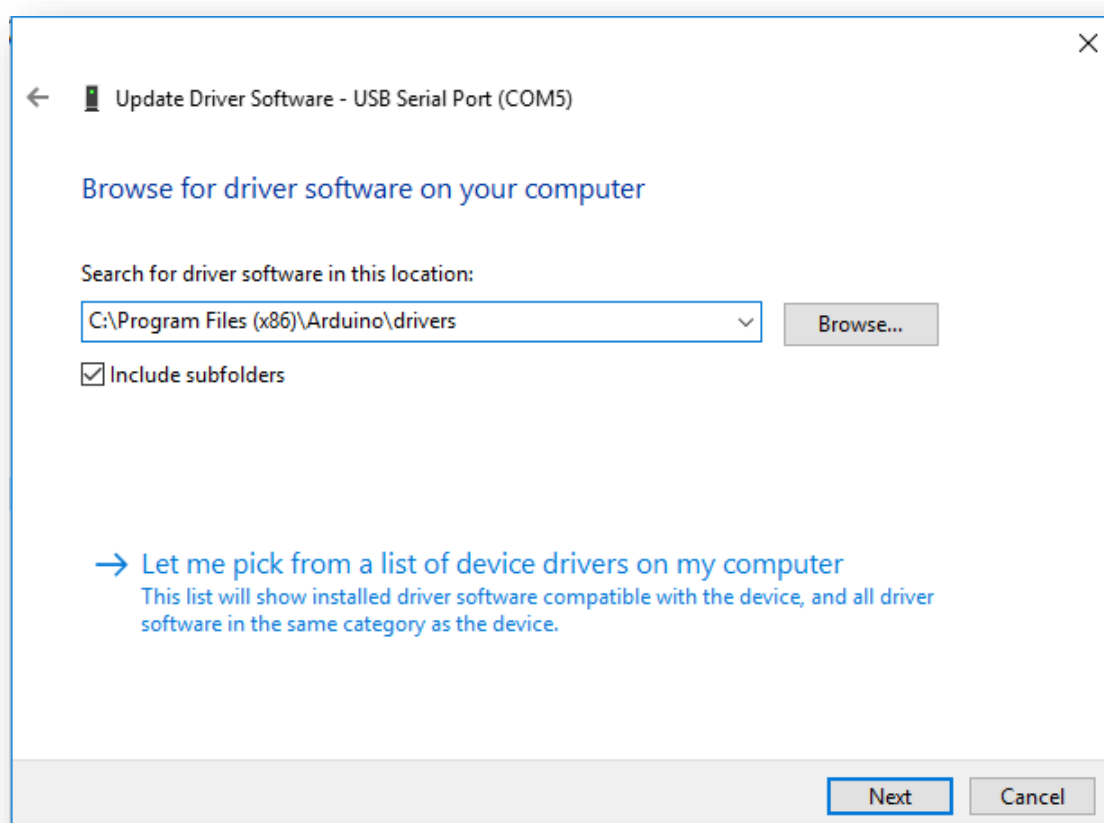
Se muestra que el controlador no se ha instalado, y tiene que hacer clic en Examinar mi pc para que el software del controlador encuentre los controladores. Las unidades se encuentran en la carpeta Arduino. Normalmente, se instalará la carpeta en C:\Program Files (x86)\Arduino.



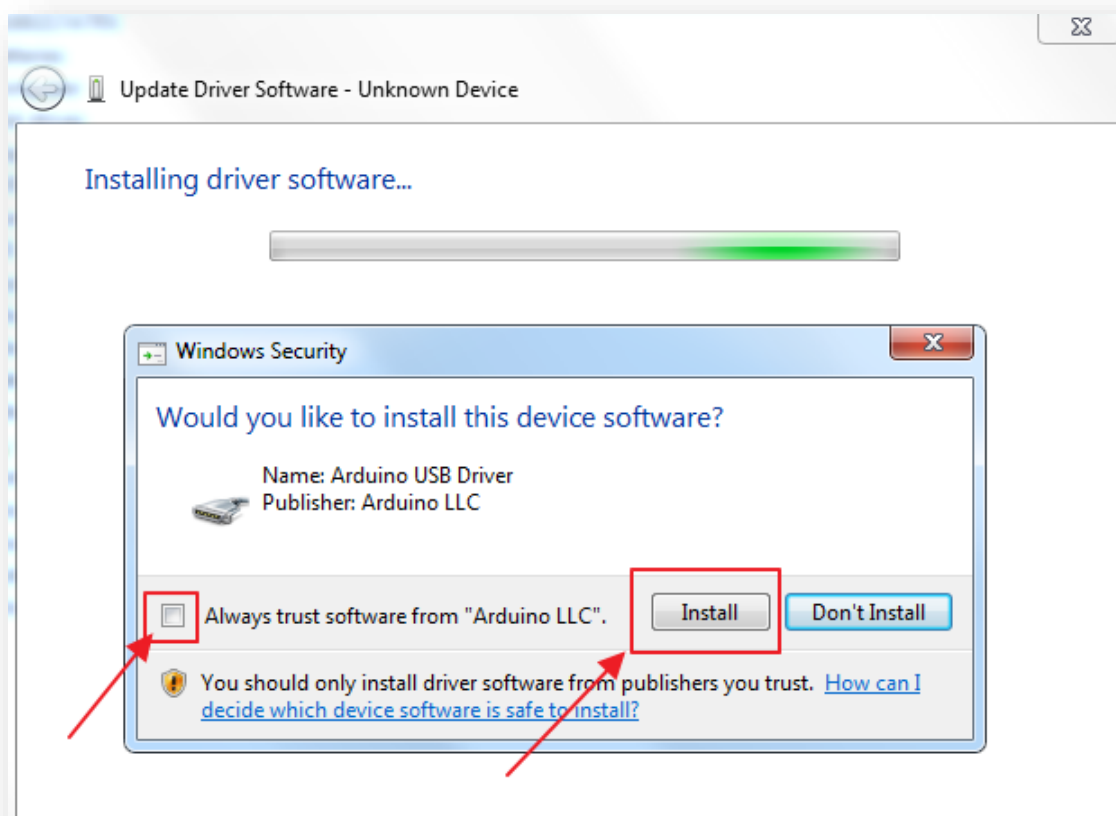
Carpeta de instalación de Arduino

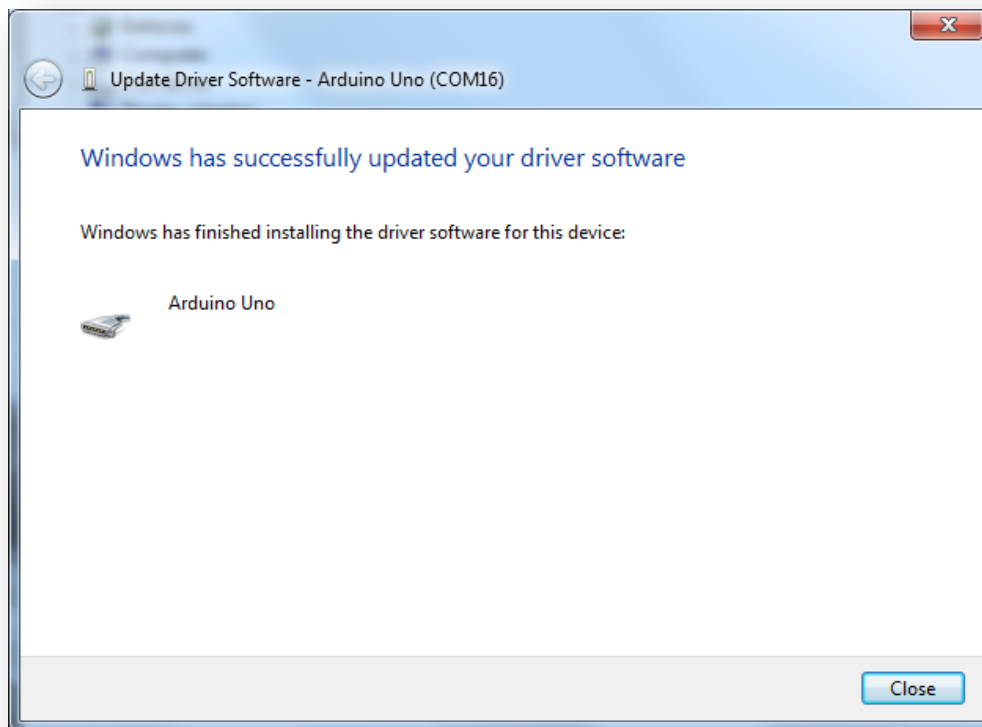


Seleccionar la carpeta de los drivers de Arduino

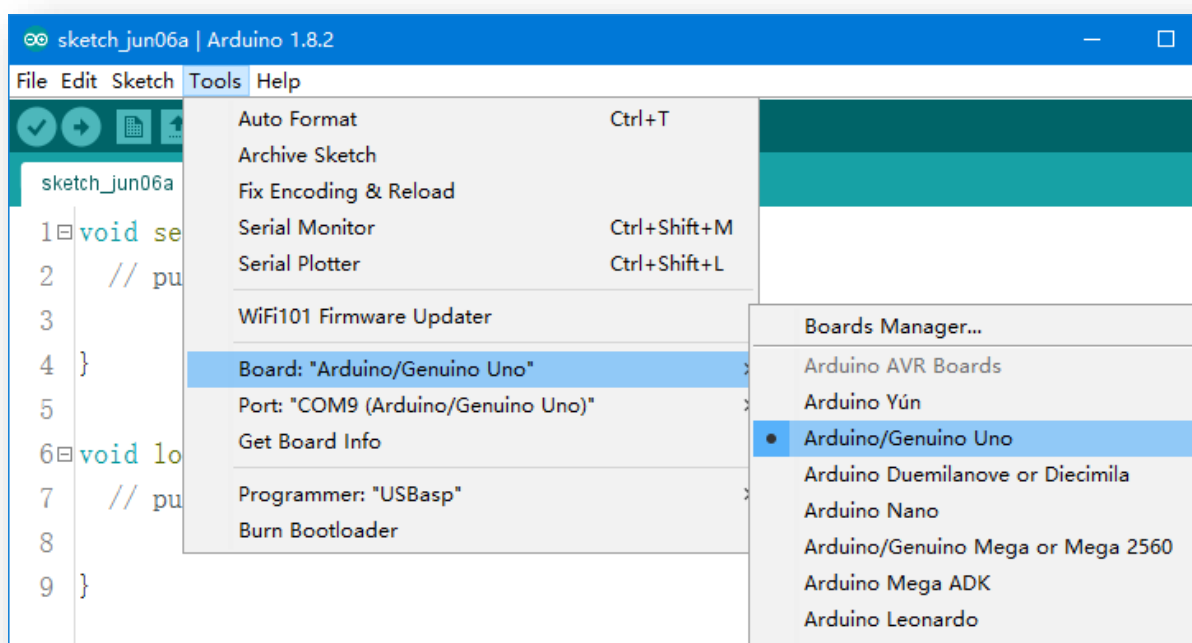


Instalar Arduino USB device

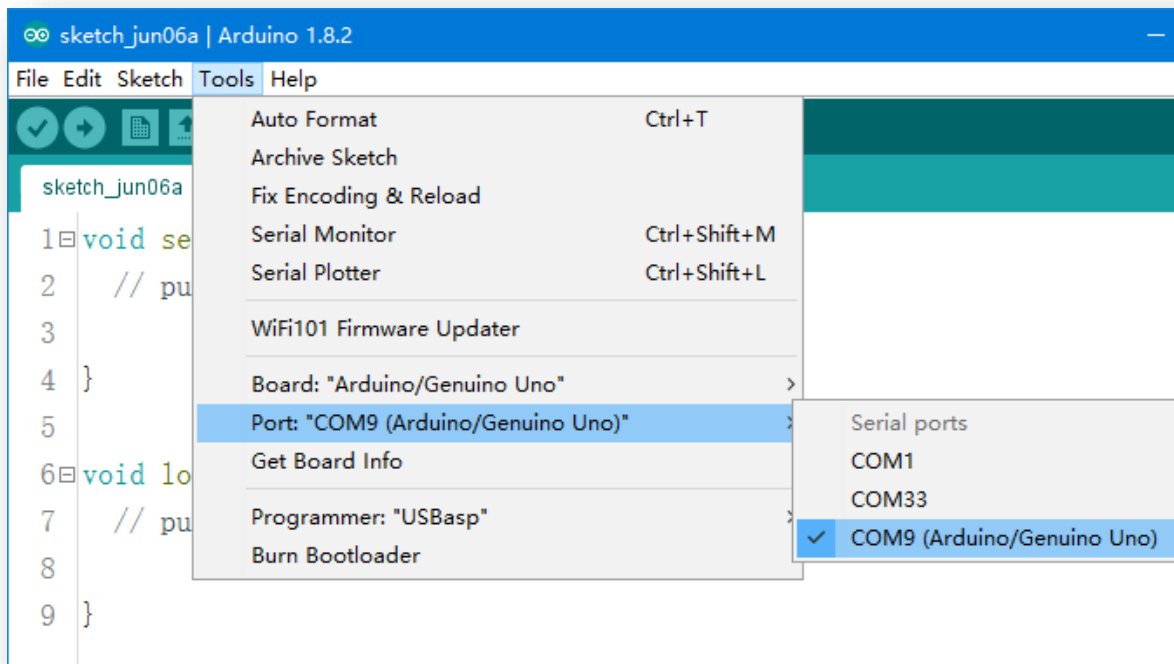




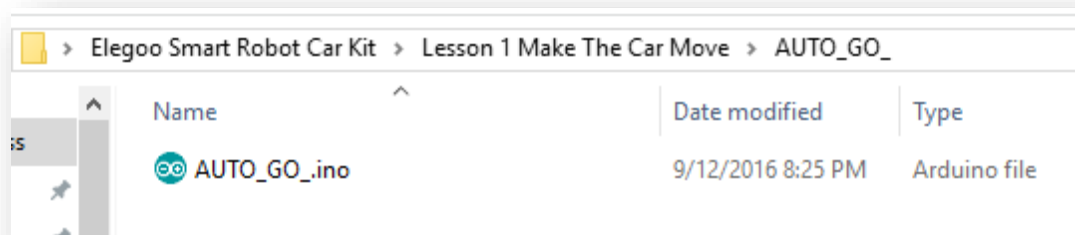
PASO 5: Tras instalar los drivers, abrir el IDE y después pulsar “Tools”→”Board”→
“Arduino/Genuino Uno”.



PASO 6: Pulsar “Tools”→”Port”→COM.



PASO 7: Abir el archivo con el codigo en el directorio “\Lesson 1 Make The Car Move \AUTO_GO_\AUTO_GO.ino” y cargarlo en el controlador de UNO (AVISO: El modulo Bluetooth debe retirarse cada vez que cargue el programa, de lo contrario no podrá cargarse.)

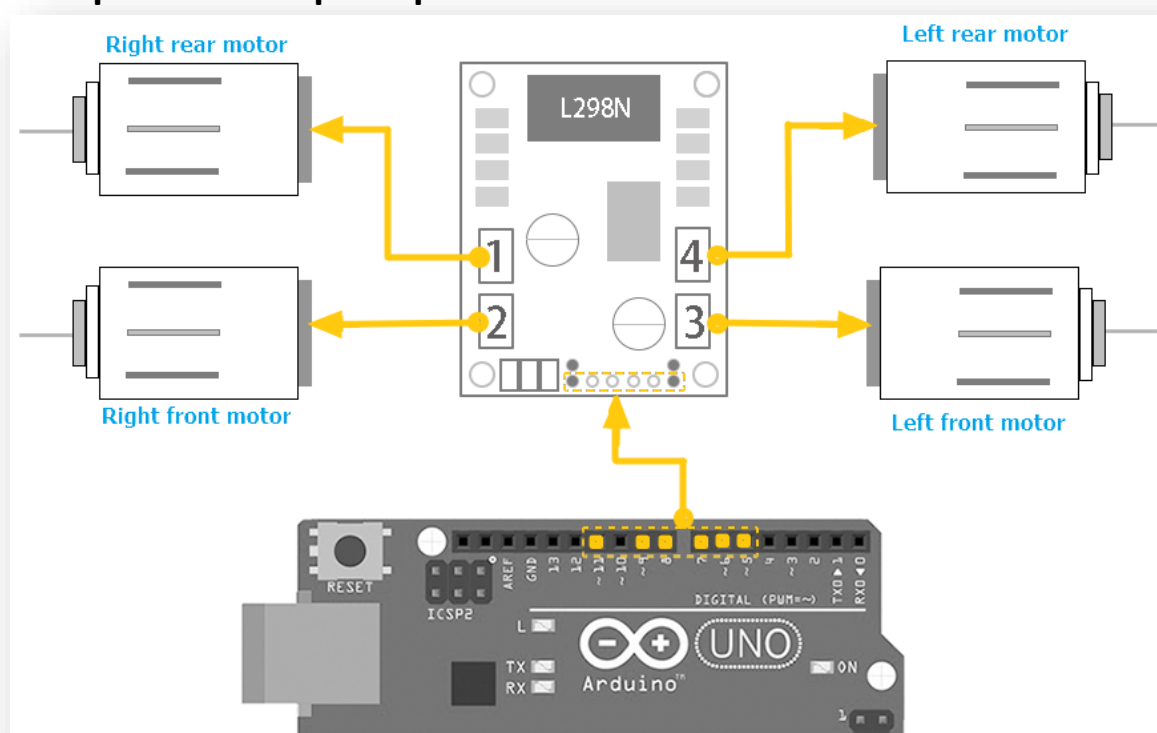




La imagen superior muestra que se ha cargado correctamente.

PASO 8: Echemos un vistazo a los resultados. Cargar el programa en el controlador de UNO. Después de desconectar el coche del ordenador , puede poner activar el interruptor y poner el coche en el suelo. Verá como este se mueve.

III. Descripción de los principios



Como usar el driver del motor de la placa L298N

La definición de los puertos de conexión en la placa L298N se ha marcado anteriormente. Los motores deben estar conectados a la tarjeta L298N como se muestra en la imagen superior, y si encuentra que la dirección de rotación de uno de los motores es opuesta, cambie la posición de conexión de sus cables negros y rojos.

L298N GND se conecta a la caja de batería GND;

L298N VCC se conecta a la caja de batería VCC;

La placa UNO también se conecta a la caja de la batería.

L298N 5V aquí no se puede conectar a UNO 5V;

ENA y ENB controlan la velocidad de los motores izquierdo y derecho por separado por PWM.

IN1, IN2, IN3, IN4: IN1 y IN2 se usan para controlar el motor izquierdo, IN3 and IN4 se usan para el motor derecho. Para el principio observe la tabla: (usemos como ejemplo el motor izquierdo)

ENA	IN1	IN2	ESTADO MOTOR DC
0	X	X	PARAR
1	0	0	FRENAR
1	1	0	ADELANTE
1	0	1	ATRÁS
1	1	1	PITIDO

IV. Hacer al coche moverse

Primer paso: Conducir el motor

Intentaremos mover el motor sin control de velocidad, pues nos será más sencillo así el escribir el código del programa.

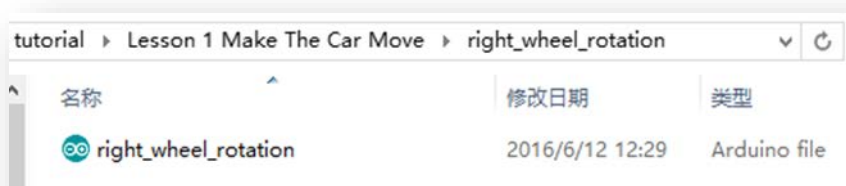
Antes de nada, vamos a ver las conexiones del motor de la placa L298N, usaremos Arduino 5, 6, 7, 8, 9, 11 pins para controlar el coche. Los pines 9 y 11 controlan la rueda derecha. Los pines 7 y 8 controlan la rueda izquierda y los pines 5 y 6 controlan ENA and ENB.

La conexión será la siguiente

L298N	V5 placa de expansión
ENA	5
ENB	6
IN1	7
IN2	8
IN3	9
IN4	11

Basado en la table superior, diseñaremos un programa simple para hacer moverse a la rueda derecho 0.5s en dirección positiva, parar 0.5s, girar 0.5s en dirección negative y parar 0.5s. Y la rueda repetirá la reacción.

Conectar la placa de control UNO al ordenador, abrir el archive de codigo de la ruta “\Lesson 1 Make The Car Move\right_wheel_rotation\ right_wheel_rotation.ino”. Subir el programa a la placa de UNO.



Vista previa del código:

```
//www.elegoo.com

//    Right motor truth table
//Here are some handy tables to show the various modes of operation.
//  ENB      IN3      IN4      Description
//  LOW   Not Applicable  Not Applicable  Motor is off
//  HIGH      LOW      LOW      Motor is stopped (brakes)
//  HIGH      LOW      HIGH      Motor is on and turning forwards
```

```

// HIGH      HIGH      LOW      Motor is on and turning backwards
// HIGH      HIGH      HIGH     Motor is stopped (brakes)

// define IO pin
#define ENB 6
#define IN3 9
#define IN4 11

//init the car
void setup() {
  pinMode(IN3, OUTPUT); //set IO pin mode OUTPUT
  pinMode(IN4, OUTPUT);
  pinMode(ENB, OUTPUT);
  digitalWrite(ENB, HIGH); //Enable right motor
}

//mian loop
void loop() {
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH); //Right wheel turning forwards
  delay(500);              //delay 500ms
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW); //Right wheel stoped
  delay(500);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW); //Right wheel turning backwards
  delay(500);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, HIGH); //Right wheel stoped
  delay(500);
}

```

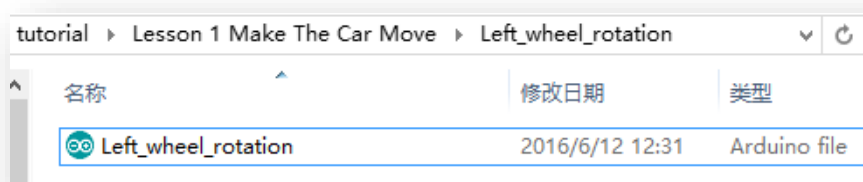
Desconectar del ordenador, y después encienda la Fuente de alimentación del coche. Verá como la rueda derecha se mueve tal y como esperaba.

Si el coche no se mueve, pulse el botón de reiniciar en la placa UNO.

Si la dirección de movimiento del motor es distinta de la deseada , puede cambiar la conexión de los cables negro y rojo del motor a la placa L298.

Luego hacemos a la rueda izquierda girar en el mismo sentido

Conecte la placa de control UNO al ordenador, abra el archivo con el código de la ruta “Lesson 1 Make The Car Move\Left_wheel_rotation\ Left_wheel_rotation.ino”. Y cargue el archivo a la placa UNO.



Vista previa del código:

```
//www.elegoo.com

//    Left motor truth table
//Here are some handy tables to show the various modes of operation.
//  ENA      IN1      IN2      Description
//  LOW   Not Applicable   Not Applicable   Motor is off
//  HIGH      LOW      LOW      Motor is stopped (brakes)
//  HIGH      HIGH      LOW      Motor is on and turning forwards
//  HIGH      LOW      HIGH      Motor is on and turning backwards
//  HIGH      HIGH      HIGH      Motor is stopped (brakes)

// define IO pin
#define ENA 5
#define IN1 7
#define IN2 8

//init the car
void setup() {
  pinMode(IN1, OUTPUT); //set IO pin mode OUTPUT
  pinMode(IN2, OUTPUT);
  pinMode(ENA, OUTPUT);
  digitalWrite(ENA, HIGH); //Enable left motor
}

//main loop
void loop() {
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW); //Right wheel turning forwards
  delay(500); //delay 500ms
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW); //Right wheel stoped
  delay(500);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH); //Right wheel turning backwards
  delay(500);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, HIGH); //Right wheel stoped
  delay(500);
}
```

Desconectar del ordenador, y después encienda la Fuente de alimentación del coche. Verá como la rueda derecha se mueve tal y como esperaba.

Segundo paso: Moverse hacia adelante y hacia atrás

Después de depurar el coche puede desarrollar programas para hacer al coche moverse.

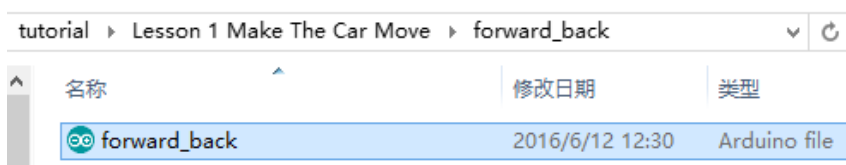
A continuación se muestra como se mueve el coche:

COCHE	Adelante	Atrás	Parar
Rueda Izquierda	Adelante	Atrás	Parar
Rueda derecha	Adelante	Atrás	Parar

COCHE	Giro izquierda	Giro derecha	Parar
Rueda izquierda	Atrás	Adelante	Parar
Rueda derecha	Adelante	Atrás	Parar

A continuación escribiremos un programa simple para hacer al coche moverse hacia adelante 0.5s , después parar 0.5s, después hacia atrás 0.5s y después parar 0.5s.

Conecte la placa de control de UNO al ordenador , abra el archivo de código de la ruta “Lesson 1 Make The Car Move\forward_back\forward_back.ino”. Cargue el programa en la placa UNO.



Vista previa del código:

```
//www.elegoo.com

// Left motor truth table
// ENA      IN1      IN2      Description
// LOW      Not Applicable      Not Applicable      Motor is off
// HIGH      LOW      LOW      Motor is stopped (brakes)
// HIGH      HIGH      LOW      Motor is on and turning forwards
// HIGH      LOW      HIGH      Motor is on and turning backwards
// HIGH      HIGH      HIGH      Motor is stopped (brakes)

// Right motor truth table
// ENB      IN3      IN4      Description
```

```
// LOW   Not Applicable   Not Applicable   Motor is off
// HIGH      LOW           LOW           Motor is stopped (brakes)
// HIGH      LOW           HIGH          Motor is on and turning forwards
// HIGH      HIGH          LOW           Motor is on and turning backwards
// HIGH      HIGH          HIGH          Motor is stopped (brakes)

//   The direction of the car's movement
// Left motor   Right motor   Description
// stop(off)    stop(off)     Car is stopped
// forward      forward       Car is running forwards
// forward      backward      Car is turning right
// backward     forward       Car is turning left
// backward     backward      Car is running backwards

//define the L298n IO pin
#define ENA 5
#define ENB 6
#define IN1 7
#define IN2 8
#define IN3 9
#define IN4 11

void setup() {
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  digitalWrite(ENA, HIGH);
  digitalWrite(ENB, HIGH);
}

void loop() {
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH); //go forward
  delay(500);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW); //stop
  delay(500);
  digitalWrite(IN1, LOW);

```

```
digitalWrite(IN2, HIGH);
digitalWrite(IN3, HIGH);
digitalWrite(IN4, LOW); //go back
delay(500);
digitalWrite(IN1, LOW);
digitalWrite(IN2, LOW);
digitalWrite(IN3, HIGH);
digitalWrite(IN4, HIGH); //stop
delay(500);
}
```

Cargue el programa en la placa UNO y después desconecte del ordenador,encienda la Fuente de alimentación del coche y verá como la rueda derecho se mueve como esperaba

Tercer paso: Escribir el programa

Puede ser difícil para usted escribir todo el programa para que el coche se mueva automáticamente. Así que separamos los movimientos en diferentes funciones, por ejemplo, avanzando y girando a la izquierda. Y cuando escribimos el programa en el paso final, podemos llamar a la función.

A continuación, comenzamos a escribir programas para cada movimiento:

Vista previa del código:

```
void forward(){
    digitalWrite(ENA,HIGH); //enable L298n A channel
    digitalWrite(ENB,HIGH); //enable L298n B channel
    digitalWrite(IN1,HIGH); //set IN1 hight level
    digitalWrite(IN2,LOW); //set IN2 low level
    digitalWrite(IN3,LOW); //set IN3 low level
    digitalWrite(IN4,HIGH); //set IN4 hight level
    Serial.println("Forward");//send message to serial monitor
}

void back(){
    digitalWrite(ENA,HIGH);
    digitalWrite(ENB,HIGH);
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    Serial.println("Back");
}
```

```

void left(){
  digitalWrite(ENA,HIGH);
  digitalWrite(ENB,HIGH);
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,HIGH);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);
  Serial.println("Left");
}

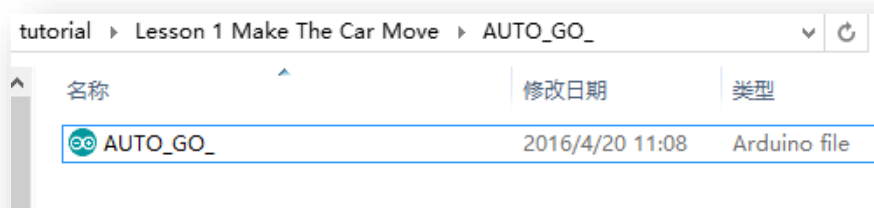
void right(){
  digitalWrite(ENA,HIGH);
  digitalWrite(ENB,HIGH);
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
  digitalWrite(IN3,HIGH);
  digitalWrite(IN4,LOW);
  Serial.println("Right");
}

```

Cuarto paso: Moverse automáticamente

Empezamos a escribir el código para que se mueva automáticamente hacia adelante 0.4s hacia atrás 0.4s – girara a la izquierda 0.4s – girar a la derecha 0.4s.

Conecte la placa de control de UNO al ordenador , abra el archivo de código de la ruta “Lesson 1 Make The Car Move\AUTO_GO_\AUTO_GO_.ino”. Cargue el programa en la placa UNO.



Code preview:

```

//www.elegoo.com

// The direction of the car's movement
// ENA  ENB  IN1  IN2  IN3  IN4  Description
// HIGH HIGH HIGH LOW  LOW  HIGH  Car is runing forward
// HIGH HIGH LOW  HIGH HIGH  LOW  Car is runing back
// HIGH HIGH LOW  HIGH LOW  HIGH  Car is turning left
// HIGH HIGH HIGH LOW  HIGH  LOW  Car is turning right
// HIGH HIGH LOW  LOW  LOW  LOW  Car is stoped

```



```
// HIGH HIGH HIGH HIGH HIGH HIGH Car is stoped
// LOW LOW N/A N/A N/A N/A Car is stoped

//define L298n module IO Pin
#define ENA 5
#define ENB 6
#define IN1 7
#define IN2 8
#define IN3 9
#define IN4 11

void forward(){
    digitalWrite(ENA,HIGH); //enable L298n A channel
    digitalWrite(ENB,HIGH); //enable L298n B channel
    digitalWrite(IN1,HIGH); //set IN1 hight level
    digitalWrite(IN2,LOW); //set IN2 low level
    digitalWrite(IN3,LOW); //set IN3 low level
    digitalWrite(IN4,HIGH); //set IN4 hight level
    Serial.println("Forward");//send message to serial monitor
}

void back(){
    digitalWrite(ENA,HIGH);
    digitalWrite(ENB,HIGH);
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    Serial.println("Back");
}

void left(){
    digitalWrite(ENA,HIGH);
    digitalWrite(ENB,HIGH);
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);
    Serial.println("Left");
}

void right(){
    digitalWrite(ENA,HIGH);
```

```
digitalWrite(ENB,HIGH);
digitalWrite(IN1,HIGH);
digitalWrite(IN2,LOW);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
Serial.println("Right");
}

//before execute loop() function,
//setup() function will execute first and only execute once
void setup() {
  Serial.begin(9600);//open serial and set the baudrate
  pinMode(IN1,OUTPUT);//before using io pin, pin mode must be set first
  pinMode(IN2,OUTPUT);
  pinMode(IN3,OUTPUT);
  pinMode(IN4,OUTPUT);
  pinMode(ENA,OUTPUT);
  pinMode(ENB,OUTPUT);
}

//Repeat execution
void loop() {
  forward(); //go forward
  delay(1000);//delay 1000 ms
  back();    //go back
  delay(1000);
  left();    //turning left
  delay(1000);
  right();   //turning right
  delay(1000);
}
```


Desconectar del ordenador, y después encienda la Fuente de alimentación del coche. Verá como la rueda derecha se mueve tal y como esperaba.

Quinto Paso: El control de velocidad

El código para conseguir controlar la velocidad: avanzar y reducir la velocidad → parar 1s → correr hacia atrás y acelerar →parar 2s.

Conecte la placa de control de UNO al ordenador , abra el archivo de código de la ruta “Lesson 1

Make The Car Move\speed_control\ speed_control.ino Cargue el programa en la placa UNO..

Elegoo Smart Robot Car Kit V3.0 > Lesson 1 Make The Car Move > speed_control			
	名称	修改日期	类型
材	 speed_control.ino	2017/5/11 15:35	Arduino file

Vista previa del código:

```
//www.elegoo.com
#define ENA 5
#define ENB 6
#define IN1 7
#define IN2 8
#define IN3 9
#define IN4 11
void setup() {
  pinMode(IN1,OUTPUT);
  pinMode(IN2,OUTPUT);
  pinMode(IN3,OUTPUT);
  pinMode(IN4,OUTPUT);
  pinMode(ENA,OUTPUT);
  pinMode(ENB,OUTPUT);
}

void loop() {
  //go forward
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);
  //reduce the speed
  for(int i = 255; i >= 0; i--){
    analogWrite(ENB,i);
    analogWrite(ENA,i);
    delay(20);
  }

  //stop
  analogWrite(ENB,0); //speed = 0
  analogWrite(ENA,0);
  delay(1000);

  //runing back
```

```
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
//accelerate
for(int i = 0; i <= 255; i++){
    analogWrite(ENB,i);
    analogWrite(ENA,i);
    delay(20);
}

//stop
digitalWrite(ENB,LOW); //Motor is off
digitalWrite(ENA,LOW);
delay(2000);
}
```