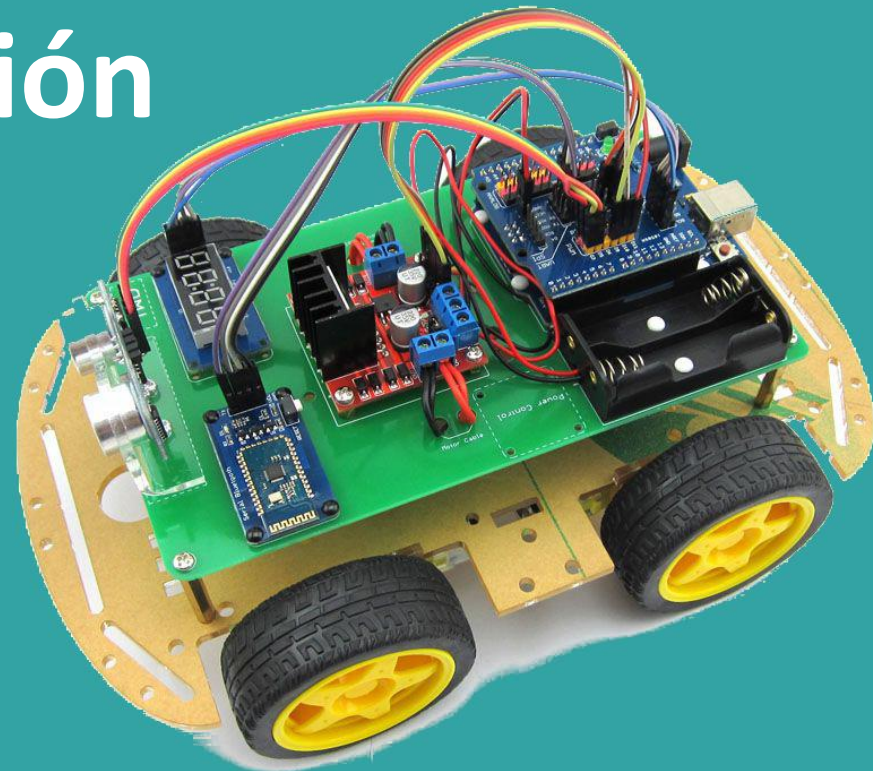


ARDUINO

Taller de introducción a la robótica



Jorge Guilló



Qué es Arduino

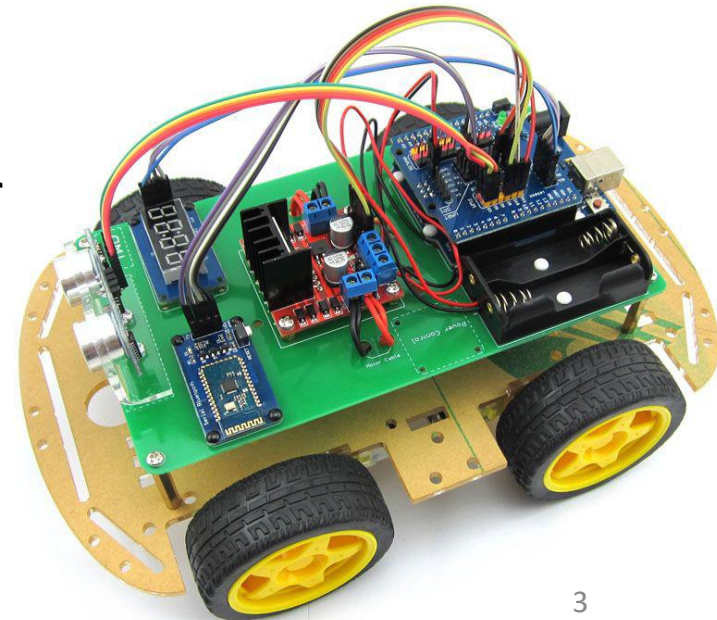
- Plataforma open-source para desarrollo de proyectos electrónicos

<https://arduino.cc>

- Consiste en una placa con un microcontrolador, pines de conexión para conectar distintos tipos de hardware (módulos) y un puerto USB para conectar al ordenador.
- Para programar el microcontrolador, la plataforma proporciona un lenguaje de programación basado en C++, un entorno de desarrollo IDE y múltiples librerías de control.

Qué vamos a hacer en el taller

- Vamos a montar un kit Smart Car
 - <https://es.aliexpress.com/item/32795674146.html>
 - Tracción de 4 ruedas con accionamiento diferencial del lado izquierdo y derecho
 - Velocidad variable
 - Movimiento tipo tanque
 - Conducción autónoma con detección de obstáculos por sonar
 - Conducción manual por bluetooth
 - Pantalla numérica

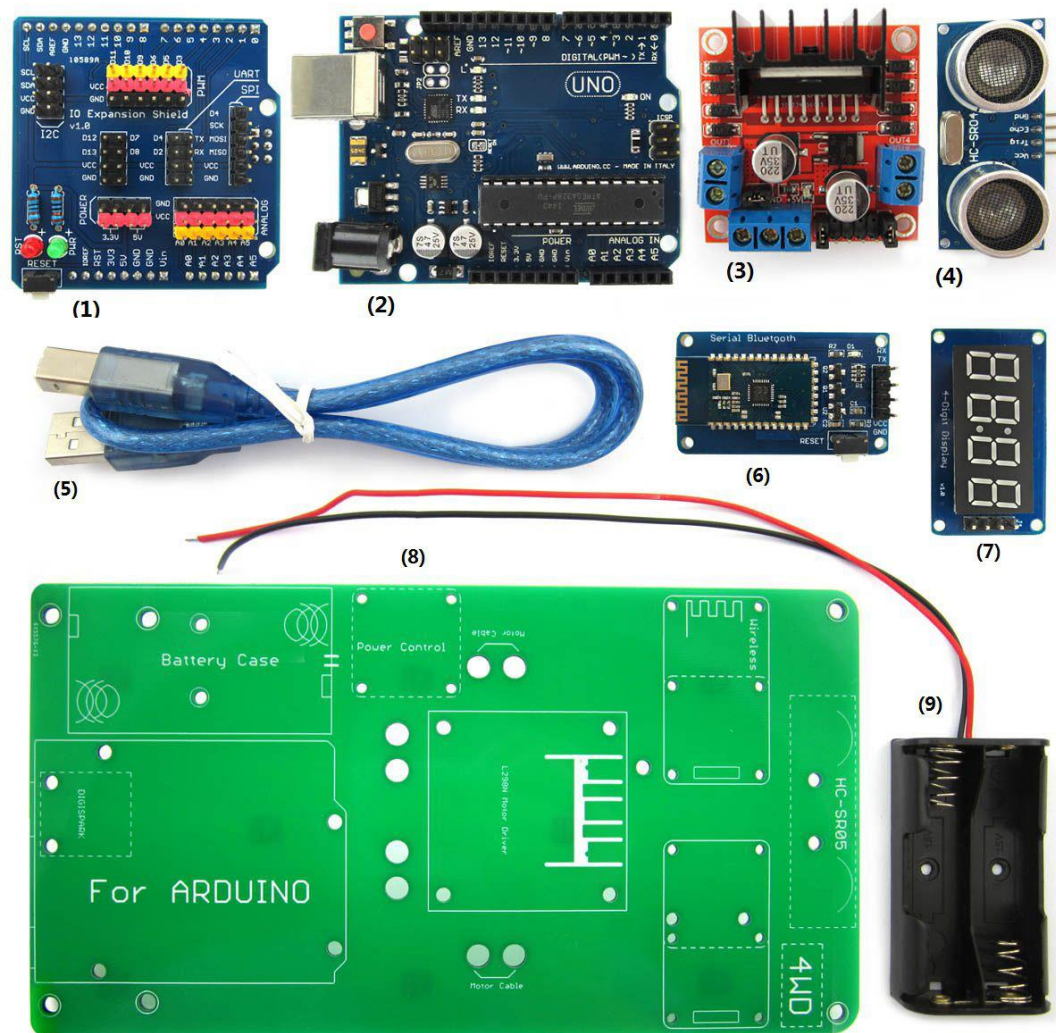




Descargas necesarias

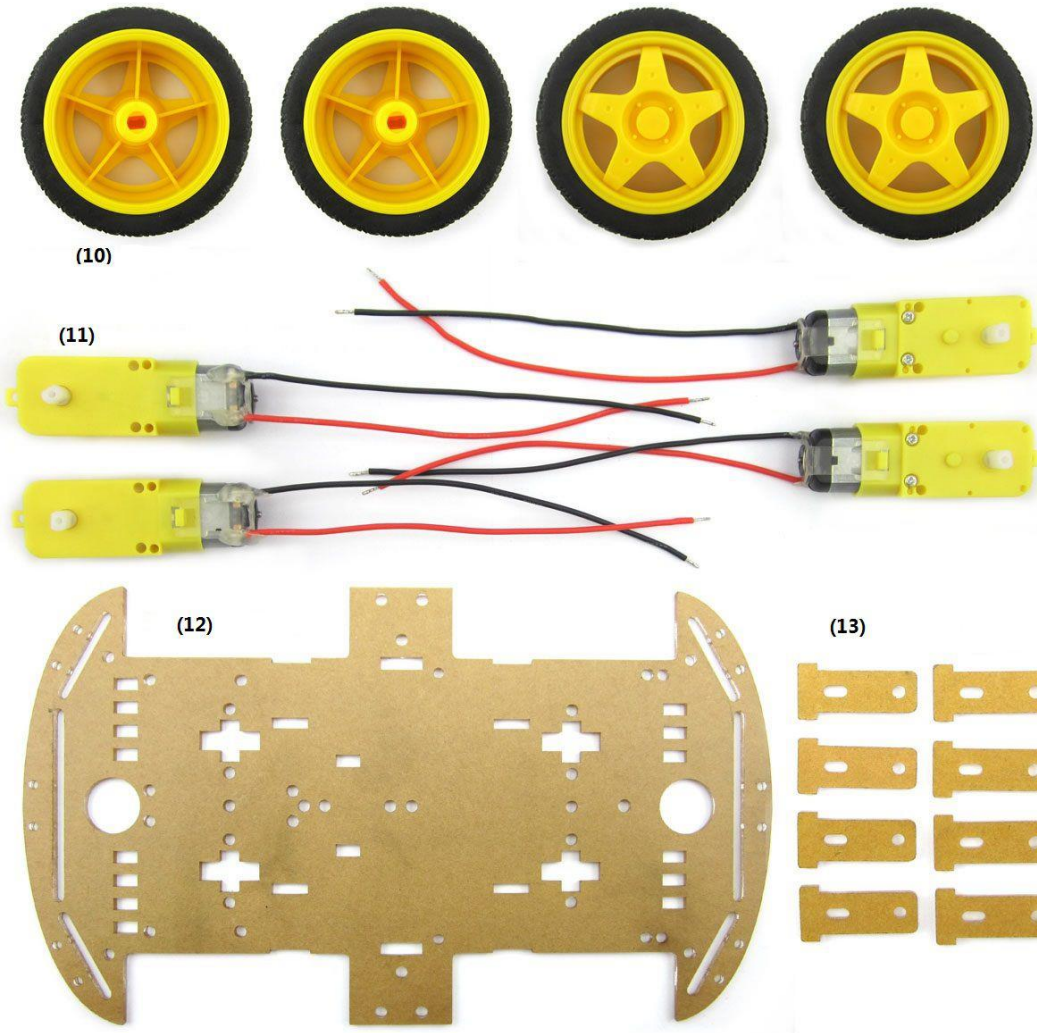
- Necesitaremos instalar el IDE de Arduino:
 - <https://www.arduino.cc/en/software>
- Y los archivos necesarios para el taller:
 - <https://github.com/jguillo/taller-arduino>
- Además, instalaremos en el móvil la app **Arduino Bluetooth RC Car**
 - <https://play.google.com/store/apps/details?id=braulio.calle.bluetoothRCcontroller>

Contenido del Kit



1. Placa de expansión E/S
2. Arduino UNO
3. Módulo de control de motores L298N
4. Módulo sensor de distancia (ultrasonidos)
5. Cable USB
6. Módulo Bluetooth
7. Pantalla 4 dígitos
8. Placa de montaje
9. Compartimento pilas
 - Usa 2 pilas de litio 1450, de 3,6V (7,2V totales)
 - El tamaño es similar a las AA pero el voltaje es distinto. 5

Contenido del Kit



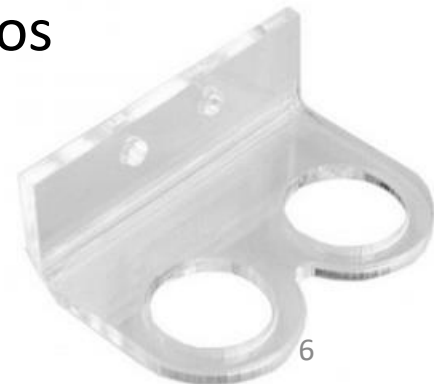
10. Ruedas

11. Motores

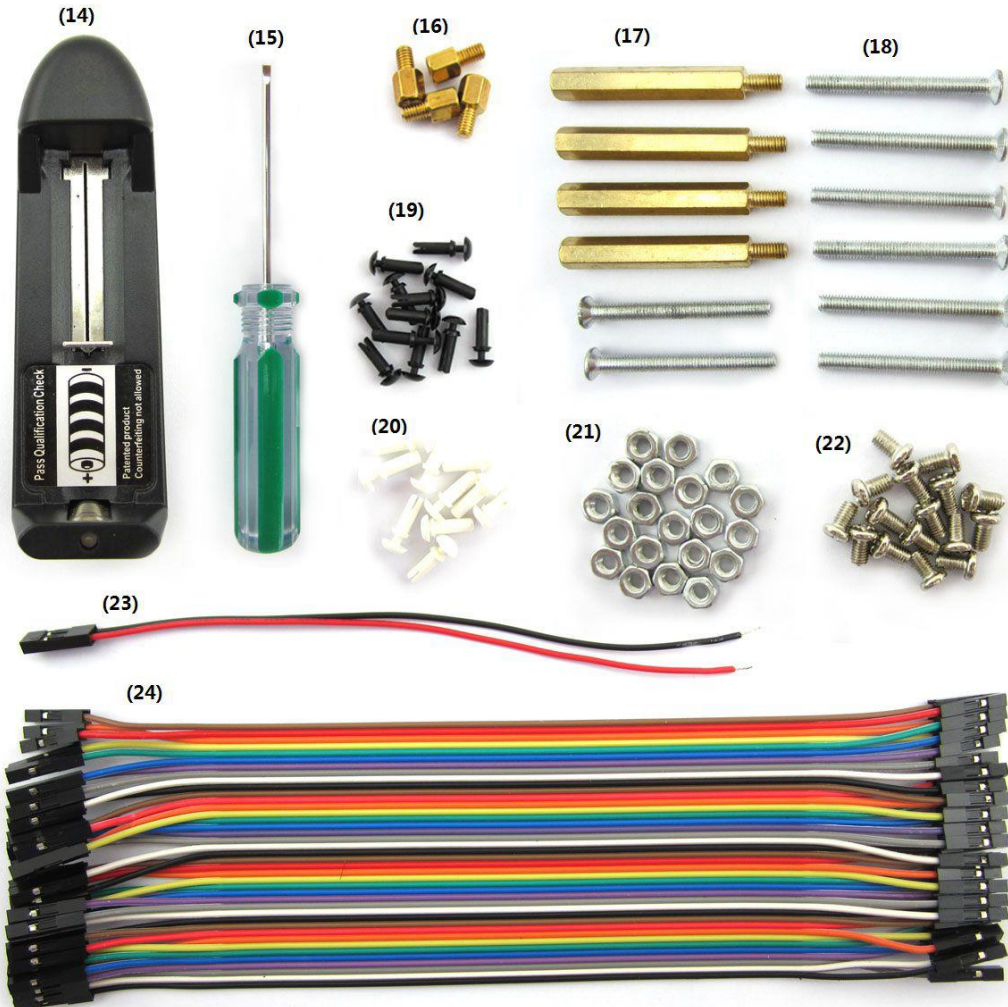
12. Placa inferior

13. Soportes de motores

Soporte del sensor de ultrasonidos



Contenido del Kit



- 14. Cargador de baterías
- 15. Destornillador
- 16. Elevadores 6 mm
- 17. Elevadores 30 mm
- 18. Tornillos 30 mm
- 19. Remaches nylon
- 20. Remaches nylon
- 21. Tuercas
- 22. Tornillos 5mm
- 23. Cable alimentación
- 24. Cables de conexión



Arduino UNO

Es la placa Arduino más habitual

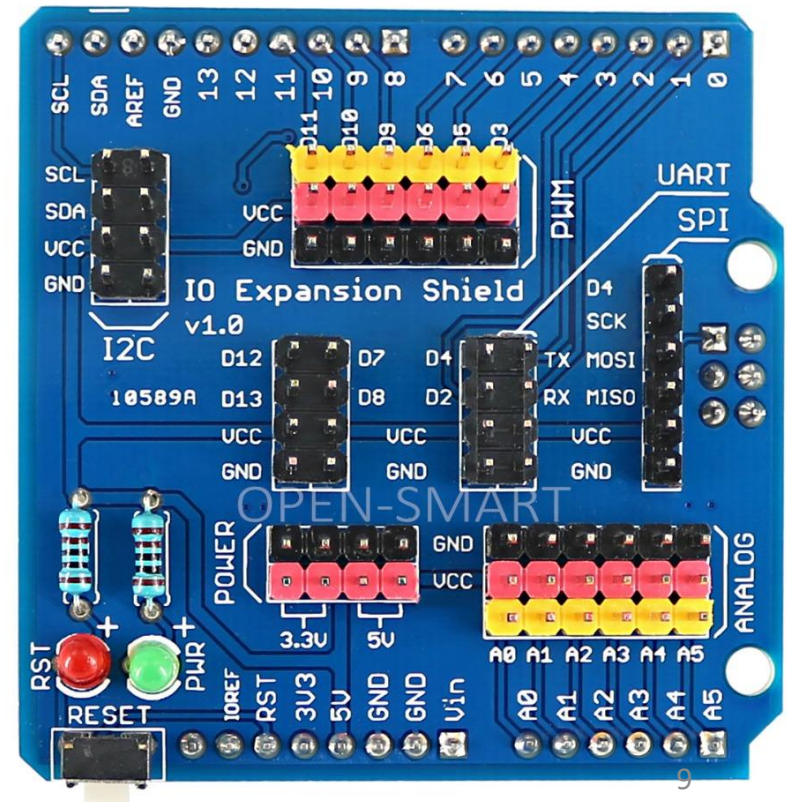
- Microcontrolador 8 bits ATmega328P
- 32 Kb memoria Flash (almacenamiento programa)
- 2 Kb memoria RAM
- 1 Kb memoria EEPROM (almacenamiento persistente)
- 14 pines E/S digital (0-13)
 - 6 de ellos permiten salida analógica PWM (3, 5, 6, 9, 10, 11)
 - 0 - 1: Comunicación serie con el ordenador
 - 13: Controla el LED integrado en la placa
- 6 pines de entrada analógica (A0-A5)
 - Se pueden usar también como pines E/S digitales
- Voltaje de trabajo: 5V
- Voltaje de entrada: 7V – 12V





Placa de expansión E/S (IO Expansion Shield)

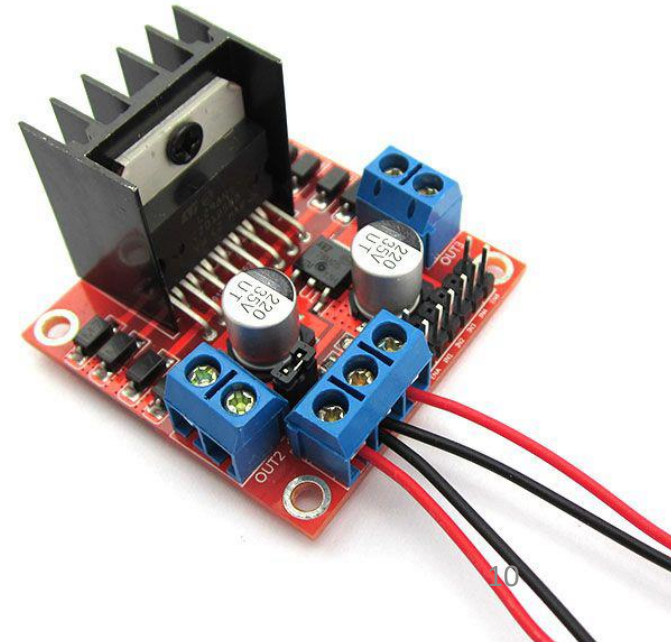
- Facilita la conexión de dispositivos a la placa
 - Proporciona un par de pines de alimentación (+5V y GND) por cada pin de E/S
 - Expone los pines de tipo estándar (bluetooth, I2C, SPI) en su disposición más común





Módulo de control de motores L298N

- Cables de alimentación
 - Recibe corriente de las baterías y la envía a los motores
 - 2 voltajes de alimentación, uno para los motores (6-24V) y otro para la electrónica (5V)
- Cables de control (3 cables x 2 motores)
 - ENA / ENB controla la velocidad
 - A más voltaje, más velocidad
 - IN1-IN2 / IN3-IN4 controlan la dirección de giro
 - HIGH LOW - Sentido horario
 - LOW HIGH - Sentido antihorario
 - HIGH HIGH o LOW LOW - Frenado



Sensor de distancia HC-SR04

- Emite un pulso de ultrasonidos y mide el tiempo que tarda en retornar el eco
- Se conecta con dos cables:
 - Trigger: Salida para ordenar el envío del pulso
 - Echo: Entrada para recoger los datos
- Lo usaremos a través de la librería NewPing



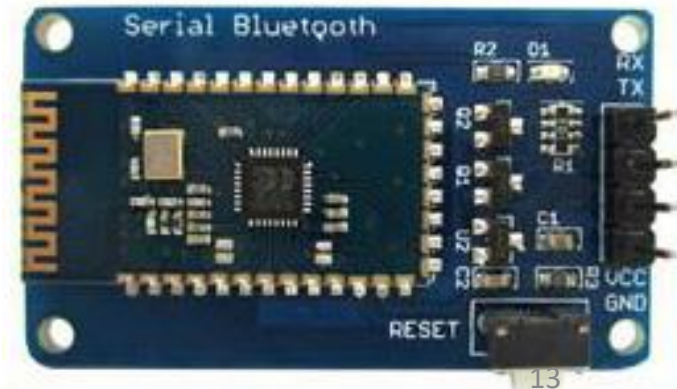
Pantalla digital 7 segmentos

- Permite mostrar 4 dígitos en modo 7 segmentos, más unos puntos de hora
- Se conecta por I2C, que es un protocolo estándar de comunicación, que usa dos cables y permite conectar varios dispositivos en un único bus.
 - SDA: Datos
 - SCL: Reloj (pulsos de sincronización)
- Cada dispositivo concreto define:
 - Una dirección (identificador numérico) para que el microcontrolador pueda indicar con qué dispositivo se comunica.
 - Los comandos que acepta y el protocolo de comunicación con el microcontrolador
- Lo usaremos a través de la librería DigitalTube (TM1637), que abstrae el protocolo de comunicación I2C.



Módulo Bluetooth SPP-C

- Gestiona por completo la conexión Bluetooth.
 - Es más común el módulo HC-06, pero el uso es similar.
- Establece un puente de comunicación en serie con el móvil u ordenador.
 - Si aún no está conectado por Bluetooth se puede configurar con distintos comandos AT
- Se conecta con dos cables (transmisión TX y recepción RX)
 - El pin TX del módulo se conecta al pin RX del microcontrolador y viceversa.
- Usaremos la librería estándar SoftwareSerial para comunicarnos con el módulo y recibir datos del móvil.

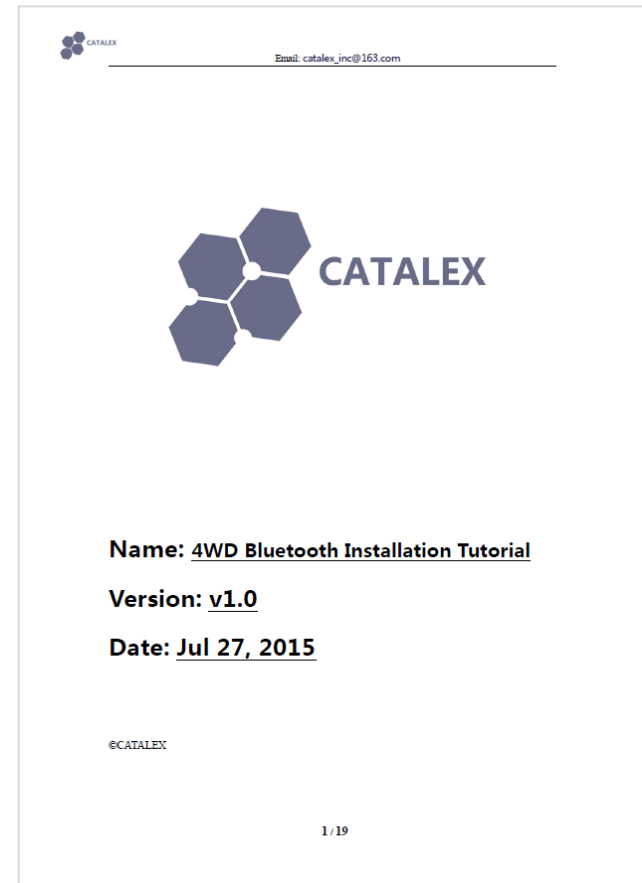




Montaje del kit

Seguiremos las instrucciones de:
**4WD Bluetooth Hardware
Installation Tutorial.pdf**

- Este documento se encuentra en la carpeta manuales
- Prestar mucha atención al orden de los pines de conexión, en especial no invertir las conexiones de VCC y GND





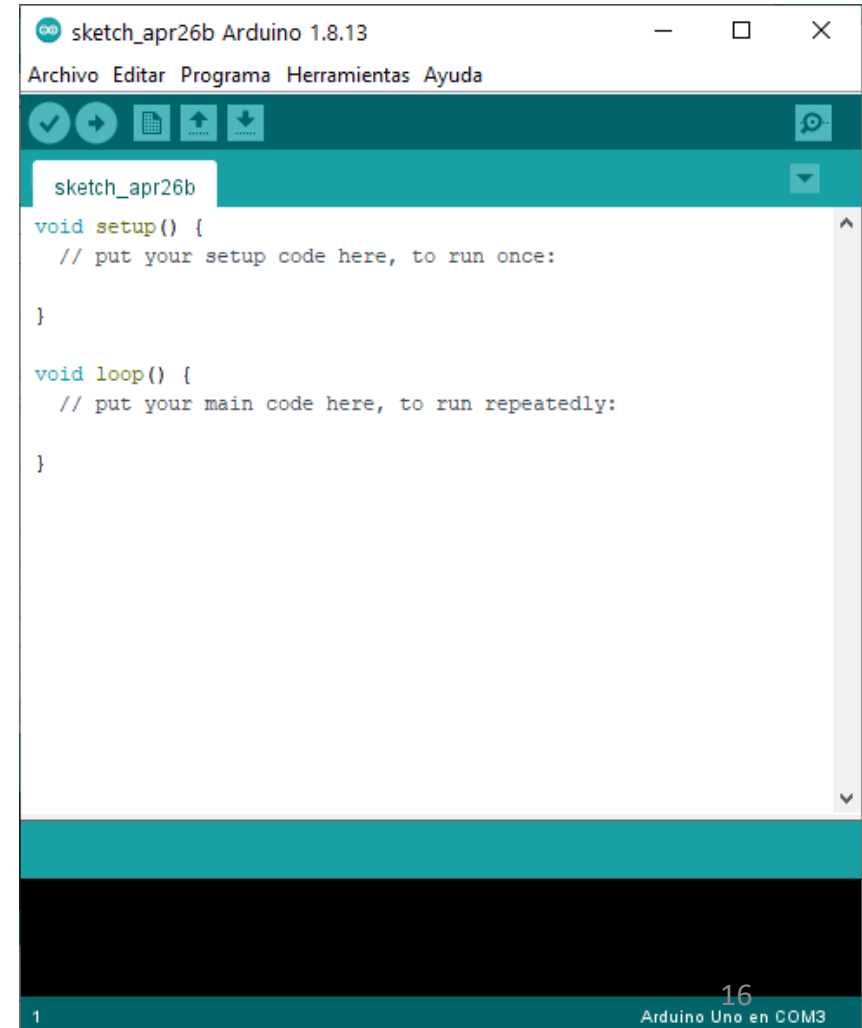
Preparación IDE Arduino

- Instalar el IDE Arduino
- Descomprimir los archivos descargados de <https://github.com/jguillo/curso-arduino>
- En las preferencias del IDE, elegir la carpeta **arduino** como carpeta de proyecto
 - La carpeta incluye todos los archivos de código y las librerías necesarias



Uso del IDE

- En el menú Herramientas, elegir
 - Placa “Arduino Uno”
 - Puerto COM que aparezca al conectar el cable USB
- Desde el menú Herramientas se abre también el Monitor Serie
 - Permite leer y enviar mensajes al Arduino por el puerto serie.
- Una vez escrito el programa, se usa el icono de la flecha para enviar el programa a la placa Arduino.





Estructura de un programa Arduino

- Un programa Arduino consta de dos funciones principales
 - `setup()` para inicializar el funcionamiento del programa
 - Se ejecuta sólo una vez al encender la placa
 - `loop()` se ejecuta continuamente
 - Una vez ejecutado el `setup`, el programa consiste en un bucle infinito de llamadas a `loop`.
 - Implementa el funcionamiento principal del programa.
- Se pueden añadir otras funciones para llamarlas desde las anteriores.



Test Pantalla

- Cargar el programa **display_test**
- La pantalla muestra un avance rápido de dígitos



Test Sonar

- Cargar el programa **sonar_test**
- En el monitor serie aparece la distancia medida por el sensor
- Cargar el programa **sonar_display_test**
- La distancia aparece en la pantalla numérica



Test Bluetooth

- Cargar el programa **bt_test**
- Consultar los comandos AT del manual **SPP-CA Bluetooth module AT commands.pdf**
- Abrir el Monitor Serie y configurarlo a 9600 baudios y envío de NL+RT como fin de línea
- Enviar AT
 - Debe devolver OK
- Consultar el nombre del dispositivo (AT+NAME)
- Cambiar el nombre del dispositivo a ArduinoN (indicando el número del Kit)
 - Ej. AT+NAMEArduino5
- Consultar la dirección Bluetooth (AT+LADDR) y apuntarla



Test Bluetooth

- Instalar la app **Arduino Bluetooth RC Car**
- En el icono de opciones, entrar en Settings
 - Face screen UP to stop car
 - Data stream frequency – On change/touch
- En el icono de opciones, entrar en Connect to Car
 - Buscar el nombre de nuestro módulo Bluetooth
 - Emparejar usando el pin 1234
- Comprobar en el Monitor Serie que se reciben los comandos correctamente.





Comandos bluetooth enviados por la app

G ↖	F ↑	I ↗
L ←	S STOP	R →
H ↙	B ↓	J ↘

	Luces delanteras ON	W
	Luces delanteras OFF	w
	Luces traseras ON	U
	Luces traseras OFF	u
	Claxon ON	V
	Claxon OFF	v
	Extra (modo auto) ON	X
	Extra (modo auto) OFF	x
	Parada completa	D

% velocidad	0	10	20	30	40	50	60	70	80	90	100
comando	0	1	2	3	4	5	6	7	8	9	q



Test controlador motores

- Cargar el programa **motor_test**
- Desconectar el cable USB y poner las baterías
- Dejar el coche en el suelo
- Cuando termina la cuenta atrás el coche debe hacer los siguientes movimientos de forma continua:
 - Avanza 1 segundo
 - Retrocede 1 segundo
 - Gira a la izquierda 1 segundo
 - Gira a la derecha 1 segundo
 - Espera 2 segundos
- Si los movimientos no corresponden con lo esperado, comprobar las conexiones o cambiar las asignaciones de pines



Programa completo

- Cargar el programa **taller_robot**
- Desconectar el cable USB y poner las baterías
- Dejar el coche en el suelo
- La pantalla muestra guiones
- Abrir la app y enlazar con el módulo bluetooth
- Una vez enlazado, la pantalla realiza una cuenta atrás de 5 segundos y el coche ya está listo para ser conducido desde la app.
 - Se puede controlar la velocidad con la barra de desplazamiento
 - Pulsando el icono del triángulo se activa o desactiva el modo autónomo
- En modo autónomo:
 - El coche avanza continuamente hasta encontrar un obstáculo
 - Al encontrar un obstáculo: frena, gira hasta que ya no tiene el obstáculo delante y continúa avanzando.