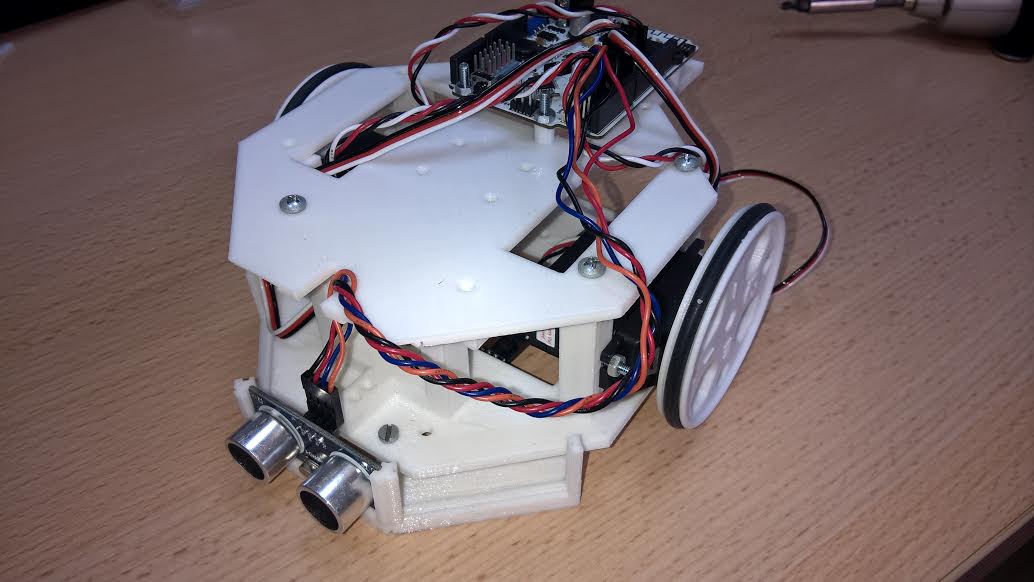
|  |
| --- |
| Software para Robots Jordán Pascual : pascualjordan@uniovi.es |

|  |  |
| --- | --- |
| 9 | Robots Móviles - Sigue líneas |

# Introducción

En esta práctica será una toma de contacto con el robot móvil



# Conceptos

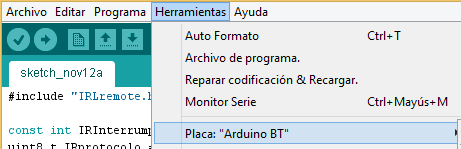
Componentes del robot:

* 2x Servomotores rotación continua (ruedas)
* Sensor de distancia ultrasónico
* **2x Sensores infrarrojos de detección de línea**

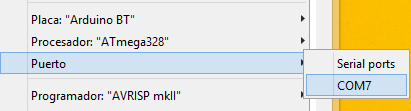
Placa:

Es una versión de la **Arduino BT, muy similar a la placa que utilizamos anteriormente.**

* Tiene una palanca de encendido/apago en un lateral (Roja y Azul).
* La conexión al PC se realiza por micro USB
* Debemos configurar el tipo de placa en Arduino sino no subirá los programas



* No suele indicar el nombre de la placa en el puerto, no obstante, sí que puede estar conectada igualmente.



* Suele tardar unos minutos en subir los programas, más que la placa Arduino Uno.
* Tiene Bluetooth integrado

**Esquema de componentes:**

|  |  |
| --- | --- |
| Componente | Conexión (Pin digital) |
| Sensor de línea central-izquierdo | 2 |
| Sensor de lineal central derecho | 3 |
| Servo izquierdo | 8 |
| Servo derecho | 9 |
| Ultrasonidos Trig | 4 |
| Ultrasonidos Echo | 5 |

|  |
| --- |
| **Importante**. Vamos a respetar siempre las posiciones de los pines que se muestran en el siguiente esquema.  El objetivo es que todos los robots estén conectados de igual manera y el código pueda ser probado en cualquiera de ellos. |

Componentes que pueden requerir calibración:

|  |  |
| --- | --- |
|  | **Sensores infrarrojos de detección de línea:** el sensor emite luz infrarroja de constante, tiene un receptor que se activará (HIGH) si la luz infrarroja “rebota” contra algo y vuelve al sensor.    Los colores claros hacen rebotar la onda, los oscuros la absorben, si la onda no rebota contra nada el receptor no recibirá ninguna señal (será lo mismo que contra un color oscuro). |
| http://www.cetronic.es/sqlcommerce/ficheros/dk_93/productos/151185040-1.jpg | **Servomotores de rotación continua:** si al enviar la señal de "detenido" un servomotor continúa moviéndose significa que está mal calibrado. |

### Avanzar y retroceder

|  |
| --- |
| Para avanzar y retroceder debemos mover los dos servomotores que controlan las ruedas en el mismo sentido, para eso uno tiene que girar de forma horaria y otro anti horaria (ya que los vemos desde arriba) . |

Dependiendo de la señal que se les envié, los servomotores se pueden mover a mayor o menor velocidad.

|  |
| --- |
| #include <Servo.h>  Servo servoIzq**;**  Servo servoDer**;**  int pinServoIzq **=** 8**;**  int pinServoDer **=** 9**;**  void setup**()**  **{**  servoIzq**.**attach**(**pinServoIzq**);**  servoDer**.**attach**(**pinServoDer**);**  **}**  void loop**()**  **{**  // Atras  servoIzq**.**write**(**180**);** // Vel Máxima  servoDer**.**write**(**0**);**  delay**(**3000**);**  servoIzq**.**write**(**100**);** // Próximo a 90, despacio  servoDer**.**write**(**80**);**  delay**(**3000**);**    servoIzq**.**write**(**90**);** // Detener  servoDer**.**write**(**90**);**  delay**(**3000**);**    // Alante  servoIzq**.**write**(**80**);** // Próximo a 90, despacio  servoDer**.**write**(**100**);**  delay**(**3000**);**  servoIzq**.**write**(**0**);** // Vel Máxima  servoDer**.**write**(**180**);**  delay**(**3000**);**  **}** |

### Cambios de dirección

Se trata de un robot que funciona con **locomoción deferencial**, no tiene ruedas directrices (como un coche), solamente tiene ruedas motrices. Dependiendo del sentido y de velocidad de giro y sentido de las ruedas se puede cambiar la dirección del robot

**Cambio de dirección con movimiento.**

Una rueda permanece parada, mientras la otra gira.

En lugar de estar parada la primera rueda podría girar simplemente más despacio que la segunda, en el momento en que las dos ruedas no se muevan a la misma velocidad se producirá un cambio en la dirección.

|  |
| --- |
| #include <Servo.h>  Servo servoIzq**;**  Servo servoDer**;**  int pinServoIzq **=** 8**;**  int pinServoDer **=** 9**;**  void setup**()**  **{**  servoIzq**.**attach**(**pinServoIzq**);**  servoDer**.**attach**(**pinServoDer**);**  **}**  void loop**()**  **{**  servoIzq**.**write**(**90**);**  servoDer**.**write**(**180**);**  delay**(**3000**);**  servoIzq**.**write**(**90**);**  servoDer**.**write**(**0**);**  delay**(**3000**);**  **}** |

**Cambio de dirección sobre el propio eje**

Las dos ruedas giran a la misma velocidad en sentidos contrarios.

Aproximadamente debemos girar las ruedas durante 1000ms para hacer un giro de 180 grados, este tiempo puede variar en función del robot y su calibración.

|  |
| --- |
| #include <Servo.h>  Servo servoIzq**;**  Servo servoDer**;**  int pinServoIzq **=** 8**;**  int pinServoDer **=** 9**;**  void setup**()**  **{**  servoIzq**.**attach**(**pinServoIzq**);**  servoDer**.**attach**(**pinServoDer**);**  **}**  void loop**()**  **{**  servoIzq**.**write**(**180**);**  servoDer**.**write**(**180**);**  delay**(**1000**);**  servoIzq**.**write**(**90**);**  servoDer**.**write**(**90**);**  delay**(**3000**);**  servoIzq**.**write**(**0**);**  servoDer**.**write**(**0**);**  delay**(**1000**);**  servoIzq**.**write**(**90**);**  servoDer**.**write**(**90**);**  delay**(**3000**);**  **}** |

### Detección de línea

Los sensores que utilizamos para detectar la línea son digitales, retornan un 0 si consiguen leer la línea, de lo contrario retornarán un 1.

El siguiente programa hace que el robot avance hasta que detecta una línea en el suelo. Al detectarla gira 180 grados y continúa caminando de nuevo.

|  |
| --- |
| #include <Servo.h>  int NO\_LINEA **=** 1**;**  int LINEA **=** 0**;**  Servo servoIzq**;**  Servo servoDer**;**  int pinServoDer **=** 9**;**  int pinServoIzq **=** 8**;**  int pinIrDer **=** 3**;**  int pinIrIzq **=** 2**;**  void setup**()**  **{**  pinMode**(** pinIrDer **,** INPUT**);**  pinMode**(** pinIrIzq **,** INPUT**);**    servoIzq**.**attach**(**pinServoIzq**);**  servoDer**.**attach**(**pinServoDer**);**  **}**  void loop**()**  **{**  servoIzq**.**write**(**0**);**  servoDer**.**write**(**180**);**    **if** **(**digitalRead**(**pinIrDer**)** **==** LINEA **||** digitalRead**(**pinIrIzq**)** **==** LINEA**){**  servoIzq**.**write**(**180**);**  servoDer**.**write**(**180**);**  delay**(**1000**);**  **}**  **}** |

### 