Mini-Taller de Robótica Móvil

Programa un robot de sumo











Actividad orientada a estudiantes de Enseñanza Secundaria y Bachillerato

Mini-Taller de Robótica Móvil Programa un robot de sumo













Planificación de Actividades



■ Introducción del Taller

- ¿Qué es un robot?¿Qué partes tiene?
- El robot TuBot
 - □ Motores
 - □ Sensores
- Entorno de programación
 - Descripción general
- Taller de robótica
 - Moviendo el robot
 - Detectando obstáculos
 - Detectando el tatami
 - Compitiendo en Sumo





Competición de Sumo



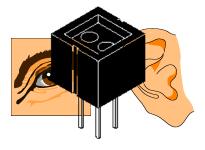


Hispabot'04 Final Panzer - Bestia



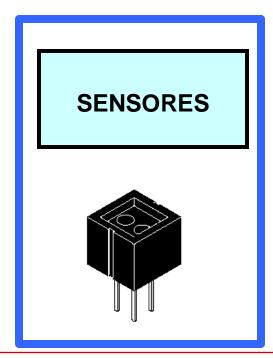


SENSORES







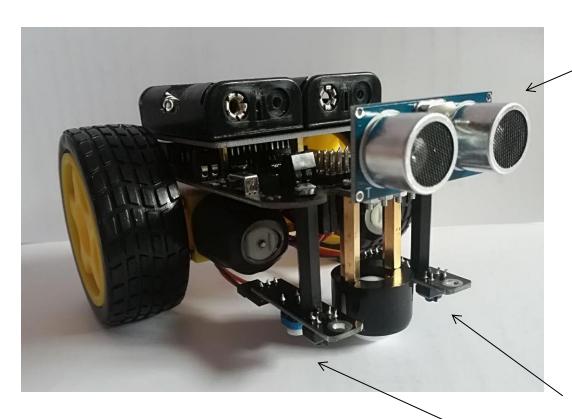




Sensores de TuBot



Sensor de Distancia por Ultrasonidos



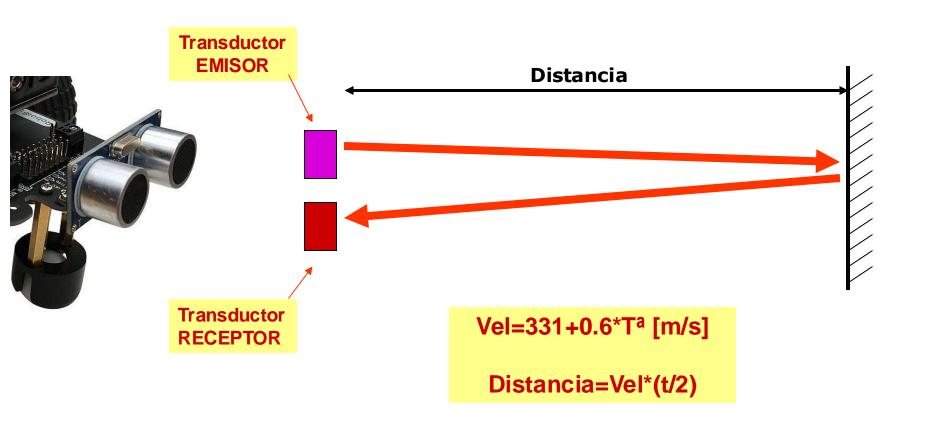
Sensor de Suelo Sensores infrarrojos reflectivos



Medida de Distancia: Ultrasonidos



- El sonido tiene una velocidad de propagación de 330m/s aprox.
- Se mide el tiempo que tarda el sonido en ir y volver

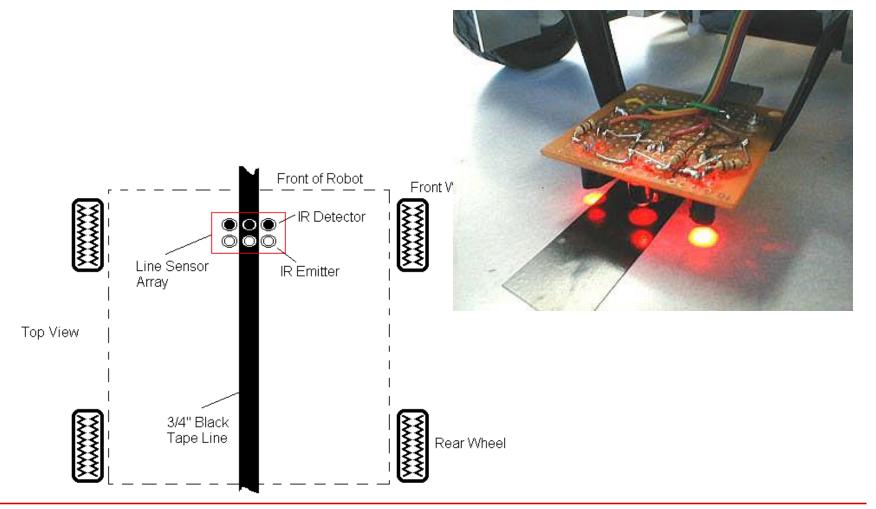




Detección de Línea o Suelo



☐ Sensor de infrarrojos reflectivo: detección de blanco o negro

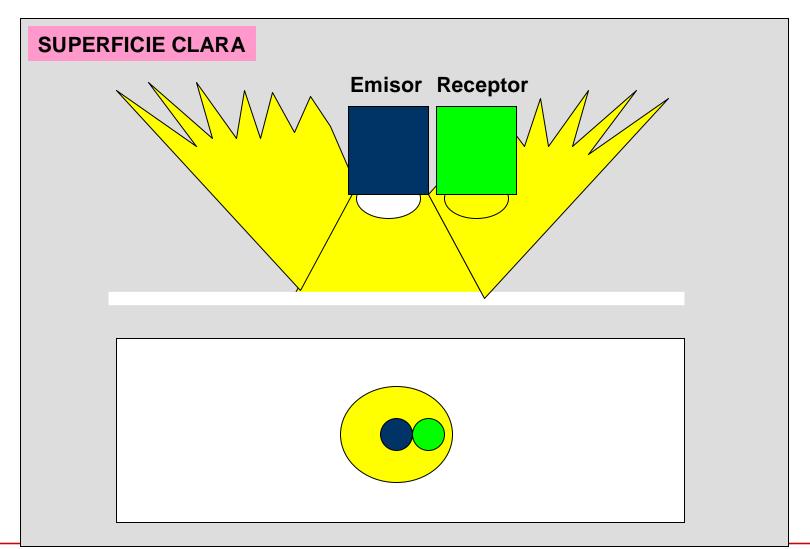




Detección de Línea o Suelo



Sensor de infrarrojos reflectivo: detección de blanco o negro

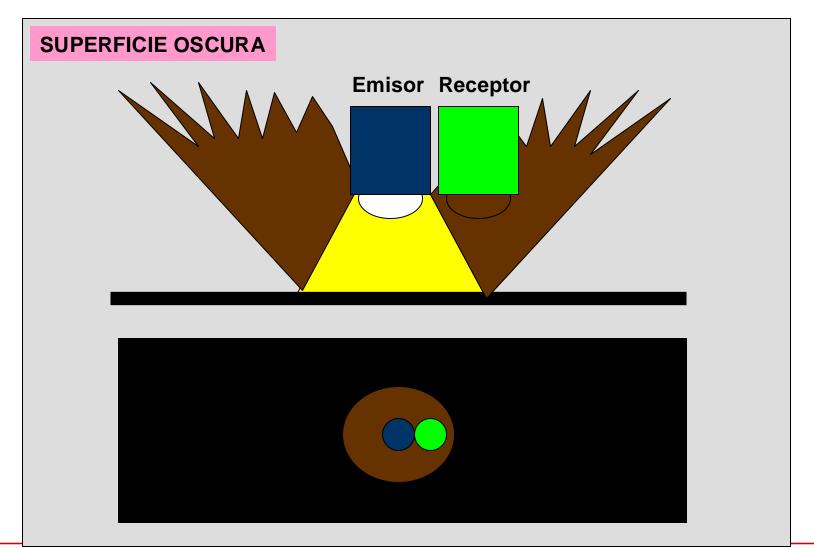




Detección de Línea o Suelo



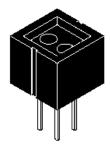
Sensor de infrarrojos reflectivo: detección de blanco o negro







SENSORES



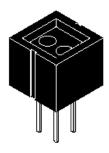
ACTUADORES







SENSORES

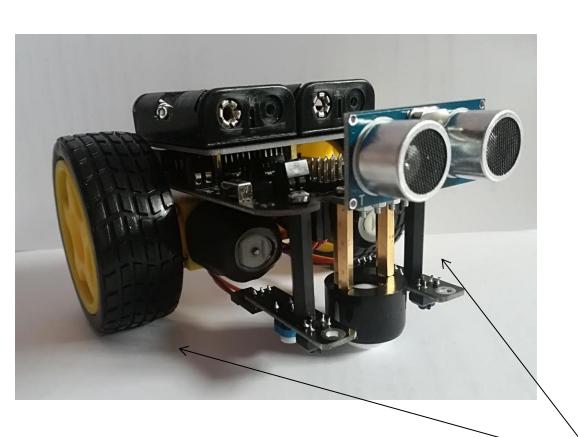


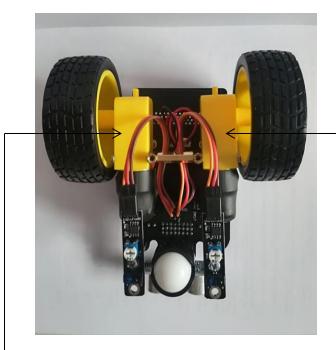




Actuadores de TuBot



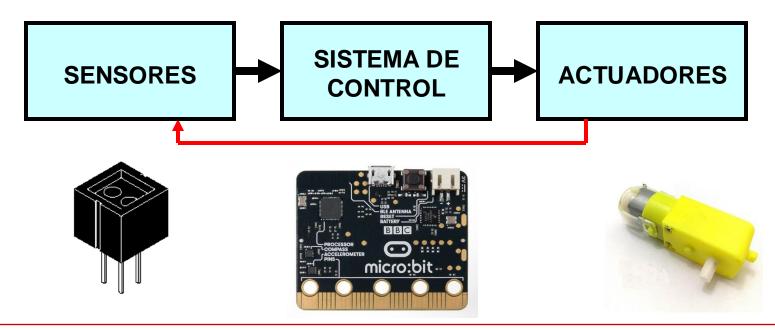




Dos motores

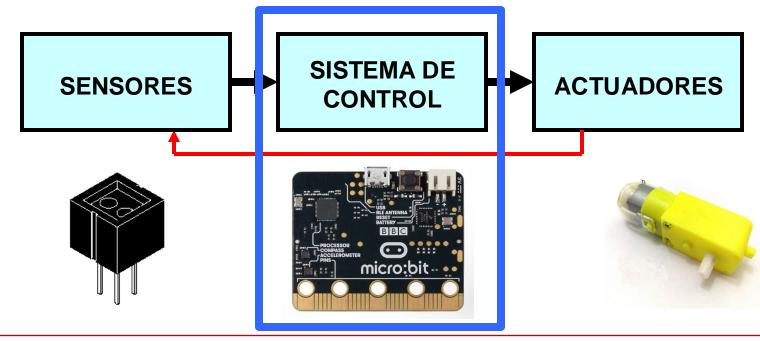












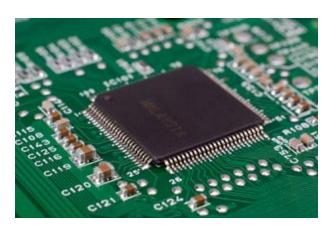


Programa y compilador



- ☐ Un **Programa** se puede escribir en muchos **lenguajes** diferentes.
- El microprocesador realmente entiende sólo órdenes en código máquina (unos y ceros)





Source: http://pacotraver.files.wordpress.com/2011/11/interprete.jpg

 $http://2.bp.blogspot.com/_Pm8qvnCsVOI/TCwv_SAuznI/AAAAAAAAAAA/9asQgJGiQMw/s1600/MICRO.jpg$



Programa y compilador



- ☐ Un **Programa** se puede escribir en muchos **lenguajes** diferentes.
- El microprocesador realmente entiende sólo órdenes en código máquina (unos y ceros)
- Para hacerlo más fácil vamos a utilizar un lenguaje gráfico que el ordenador traduce a código máquina (unos y ceros)

```
on start

iii show string ("Hello!")

iii show number (0)

iii clear screen

iii clear screen

iii clear screen

iii show string ("YES")

else

iii show string ("YES")

else

iii show string ("NO")

else

iii show string ("IDON'T KNOW")

iii show number (8)
```



MICROBIT: buen sistema de iniciación



- Plataforma diseñada como elemento de iniciación a la programación y a la electrónica.
- Muy sencillo de utilizar.
- Dispone de una enorme comunidad de usuarios.





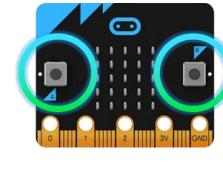
micro:bit

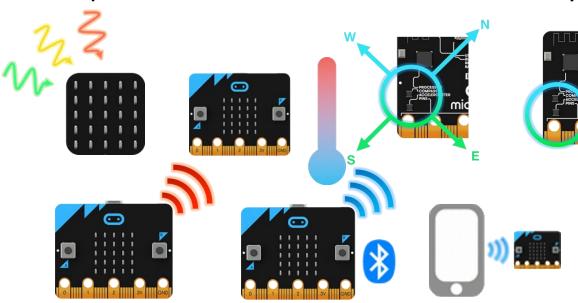


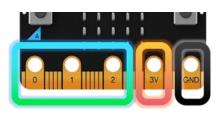
MICROBIT: buen sistema de iniciación



- □ Matriz de 5x5 LEDs capaz de visualizar mensajes
- Dos pulsadores
- □ Sensor de luz ambiente
- Sensor de temperatura
- Brújula
- □ Acelerómetros en tres ejes
- □ Capacidad de comunicación Bluettoth y radio



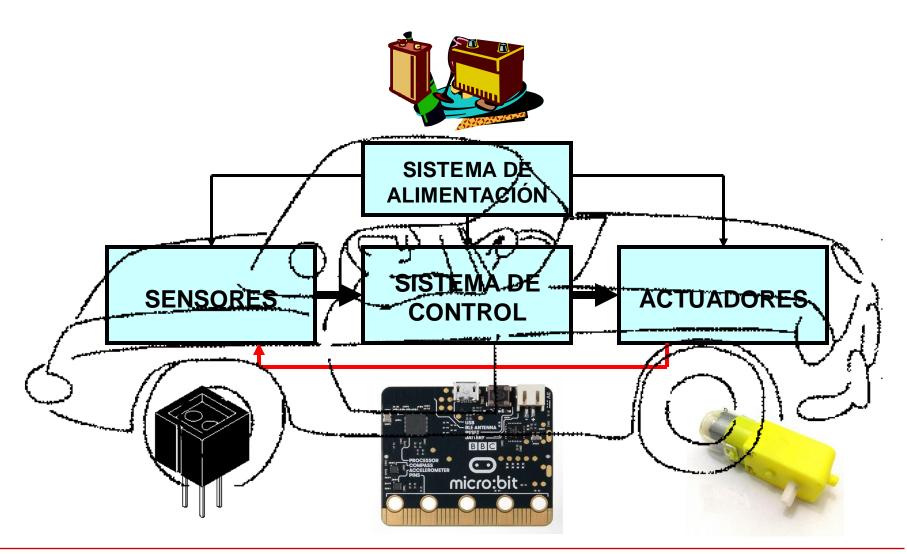




http://microbit.org/es/

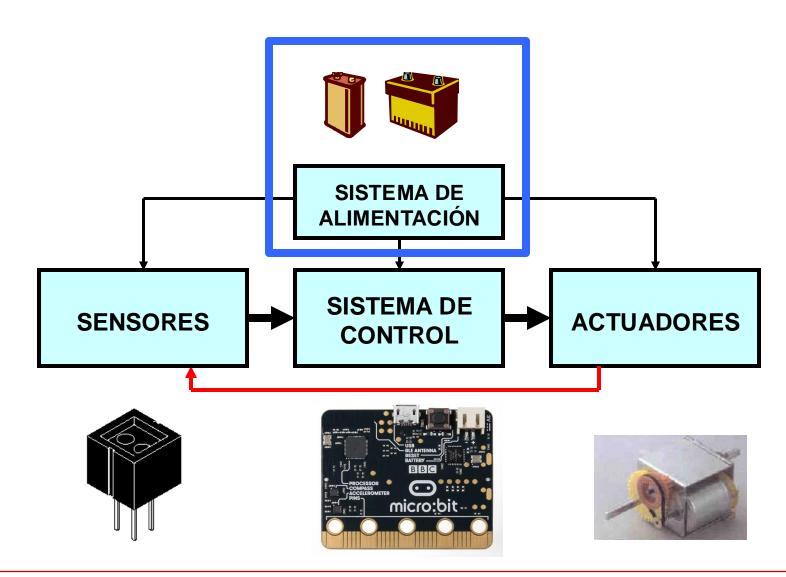






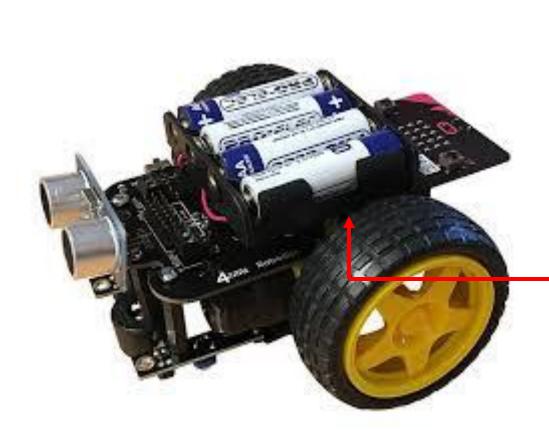












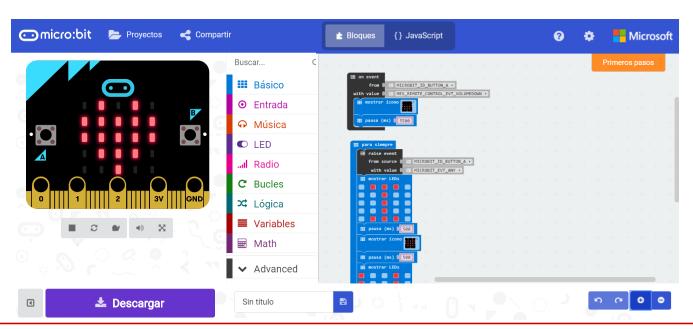




Programación con MicroBit



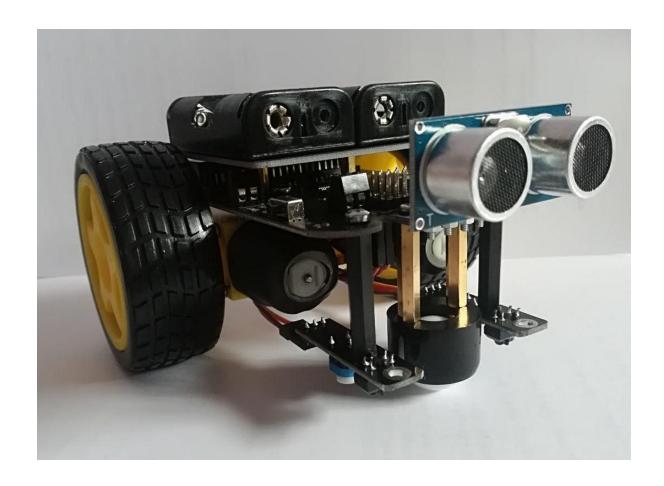
- □ Vamos a utilizar un lenguaje de programación gráfico: Editor de Bloques de JavaScript desarrollado por Microsoft MakeCode para MicroBit
 - Software de programación OnLine
 - Su uso es gratuito
 - Versión web → https://makecode.microbit.org/#lang=es
 - Si te quedas sin internet: iNo pasa nada! Sigue funcionando ③





Programación







Paso 0.a: Metodología



- 1. En el Taller te iremos enseñando **paso a paso** cómo programar el robot para que haga lo que tú quieras.
- 2. El robot ya está montado. Si ves que hay algún cable o tornillo suelto (o que se está soltando) díselo a los profesores.
- 3. Sigue las instrucciones paso a paso. No te saltes ninguno.
- 4. Lee detenidamente las instrucciones de cada uno de los pasos.
 - Si algo no entiendes pregunta a los profesores.
- 5. En cada paso **se parte de un ejemplo** ya hecho. Hay que cargarlo, probarlo y entender lo que hace.
 - Si no entiendes lo que hace. Pregunta a los profesores



Paso 0.b: Metodología



- En cada paso se proponen ejercicios para que hagas modificando el programa de base.
 - Hay ejercicio marcados en verde (fáciles de hacer) y otros marcados en rojo (para los más avanzados, no es necesario hacerlos) hay otros de dificultad intermedia de color naranja.
- 7. Las **primeras instrucciones son más guiadas**, las últimas requerirán que pongas más de tu parte.
- 8. Si terminas antes que los demás, intenta mejorar el robot para que sea más competitivo.



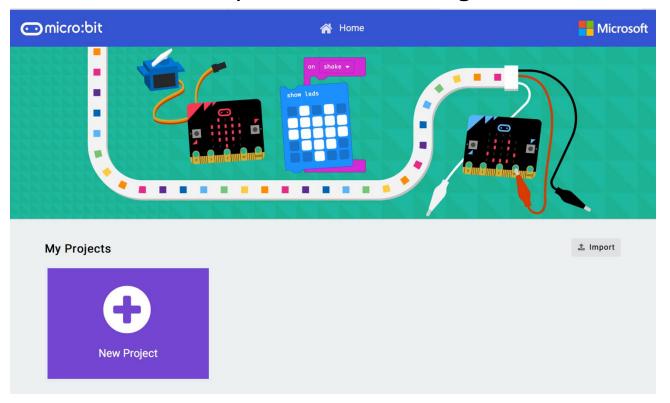
Paso 1a: Abriendo el entorno de programación



Abre la aplicación MicroBit a ser posible con Chrome:

https://makecode.microbit.org

La aplicación tiene la apariencia de la figura



□ Pulsa en "New Project"

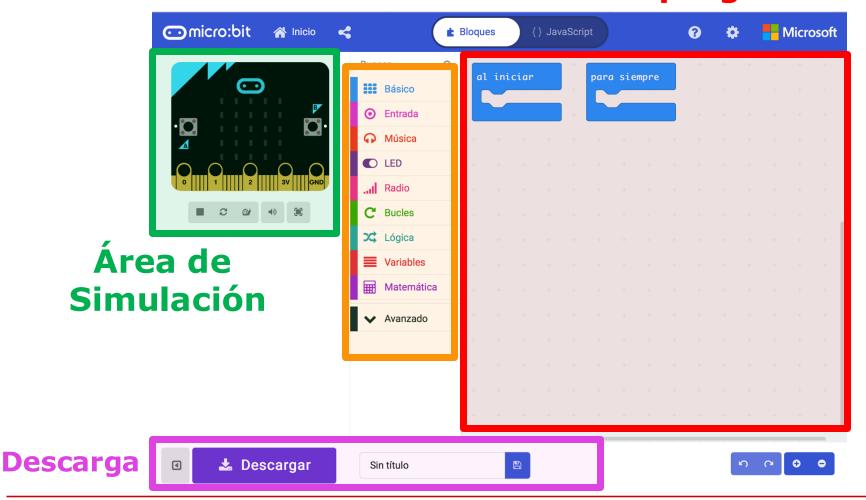


Paso 1a: Abriendo el entorno de programación



La aplicación tiene la apariencia de la figura

Área de Instrucciones Área de programación

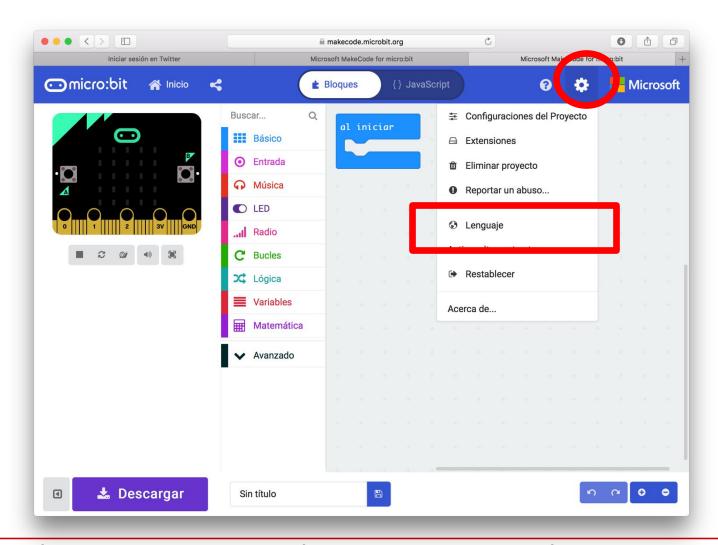




Paso 1a: Abriendo el entorno de programación



Configuración del idioma



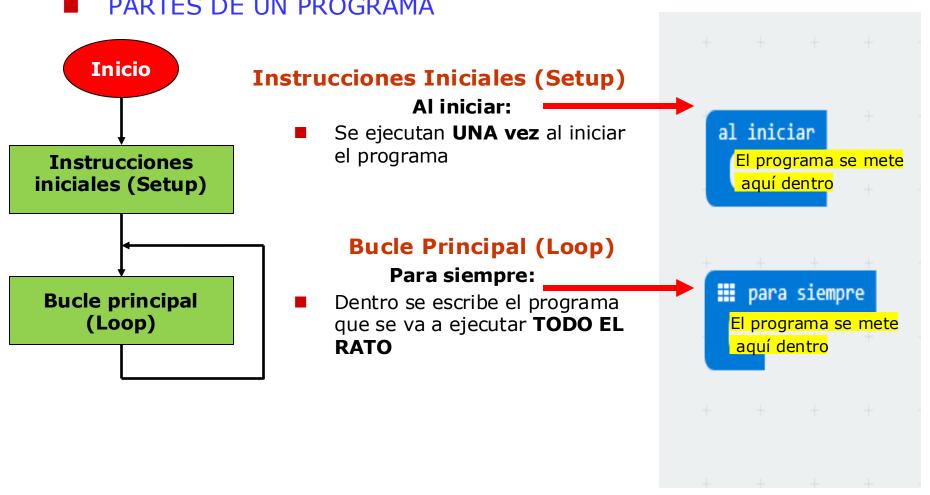


Paso 1b: Abriendo el entorno de programación



Área de Programación

PARTES DE UN PROGRAMA





Paso 1c: Abriendo el entorno de programación



☐ En cada paso habrá un ejemplo a ejecutar

Para cargar los proyectos:

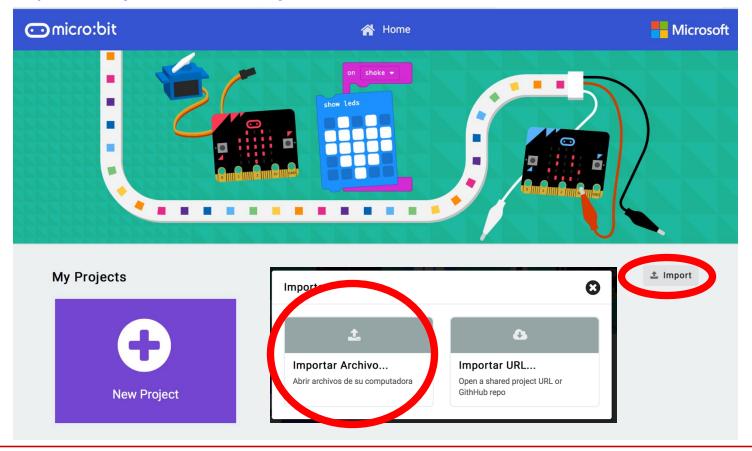
- 1. Se selecciona **PROYECTOS**, en la parte superior
- 2. Seleccionar **IMPORTAR ARCHIVO** desde el ordenador
- 3. Con cada ejercicio nuevo, abrir su PASO correspondiente:
 - Paso 3 EncendiendoUnLed.hex
 - Paso_4_SensorDeDistancia.hex
 - Paso_5_DetectandoObjeto.hex
 - Paso_6_MoviendoMotores.hex
 - Paso_7_DetectandoObstaculos.hex
 - Paso_8_SensoresSueloConFunciones.hex



Paso 2.a: Abriendo el primer programa



- □ Carga el programa Paso_3_EncendiendoUnLED.hex
 - Volver a la página principal: https://makecode.microbit.org/#
 - ☐ También pulsando en "Home" o "Inicio" (Arriba a la izquierda)
 - Importar (a la derecha)

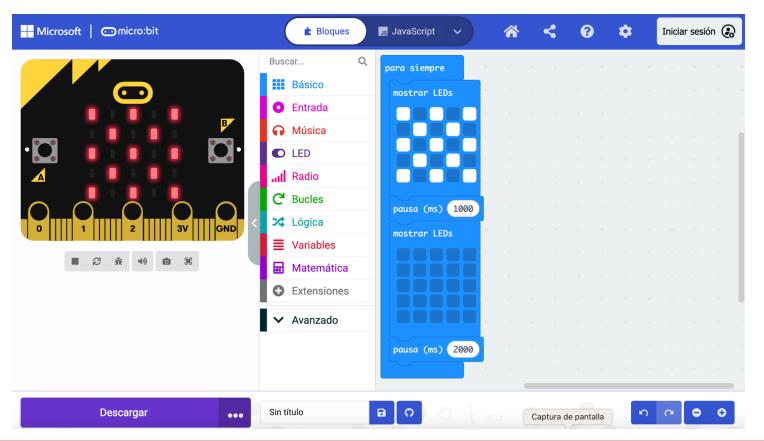




Paso 2.b: Abriendo el primer programa



- □ Carga el programa Paso_3_EncendiendoUnLED.hex
 - Proyectos (Arriba a la izquierda) → Importar Archivo
 - Este programa hace que los LEDs se enciendan y se apaguen





Paso 2.c: Abriendo el primer programa



☐ Área de Programación

Permite introducir un programa en la tarjeta de control

Usaremos la pantalla de Bloques **Bloques** JavaScript (JavaScript es más complejo) Buscar... para siempre Básico A la izquierda: mostrar LEDs Entrada Los bloques que se van Música a utilizar C LED Radio C Bucles pausa (ms) (1000 A la derecha: El programa 1 Lógica mostrar LEDs Variables **Matemática** Extensiones Los colores del menú de Avanzado bloques es el mismo que el de los bloques que lo pausa (ms) 2000 contienen



Paso 2.d: Abriendo el primer programa



☐ Área de Programación

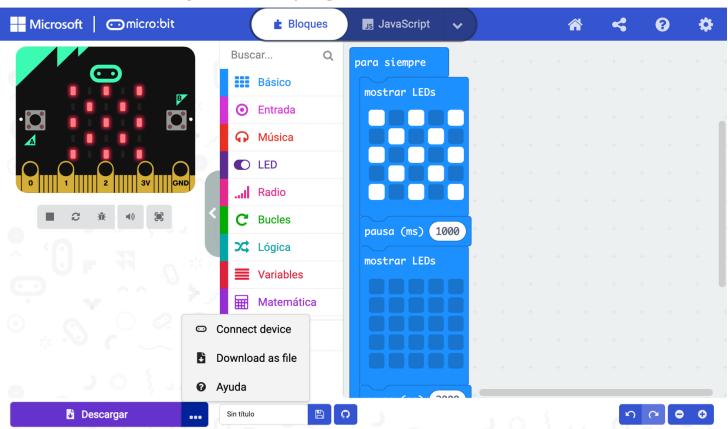
Permite introducir un programa en la tarjeta de control

Usaremos la pantalla de Bloques Js JavaScript Bloques (JavaScript es más complejo) Básico Buscar Básico A la izquierda: mostrar número 0 Los bloques que se van Música mostrar LEDs a utilizar C LED Radio C Buck A la derecha: El programa Lógica ■ Variables mostrar ícono **Matemática** Extensiones Los colores del menú de mostrar cadena ("Hello!" Avanzado bloques es el mismo que el borrar la pantalla de los bloques que lo contienen





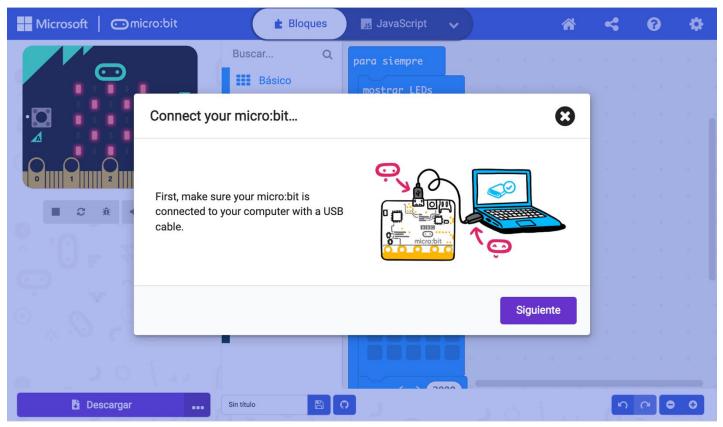
- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Conecta la tarjeta MicroBit al ordenador con el cable USB
 - □ Aparecerá un nuevo disco USB
 - Conectar la tarjeta con el programa







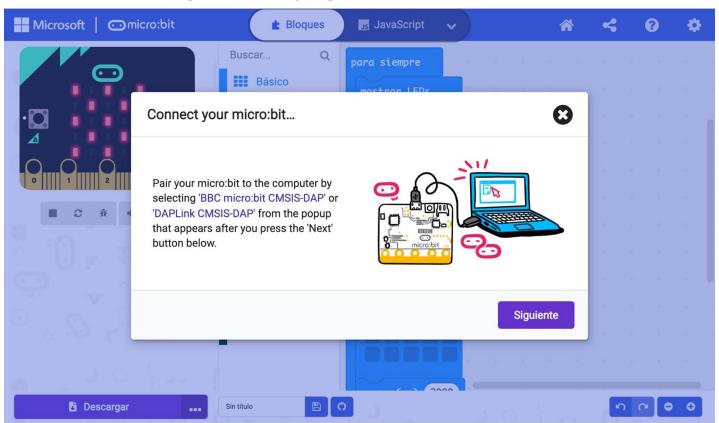
- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Conecta la tarjeta MicroBit al ordenador con el cable USB
 - ☐ Aparecerá un nuevo disco USB
 - □ Conectar la tarjeta con el programa







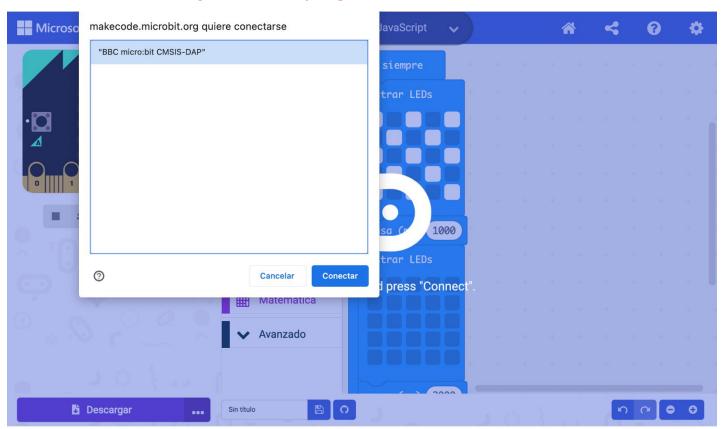
- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Conecta la tarjeta MicroBit al ordenador con el cable USB
 - ☐ Aparecerá un nuevo disco USB
 - Conectar la tarjeta con el programa







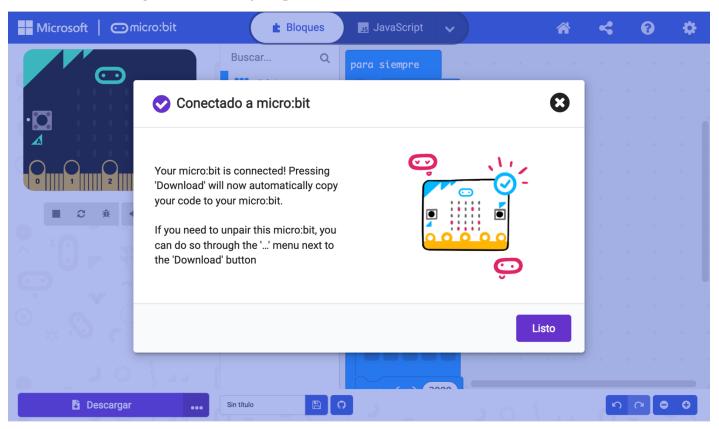
- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Conecta la tarjeta MicroBit al ordenador con el cable USB
 - ☐ Aparecerá un nuevo disco USB
 - □ Conectar la tarjeta con el programa







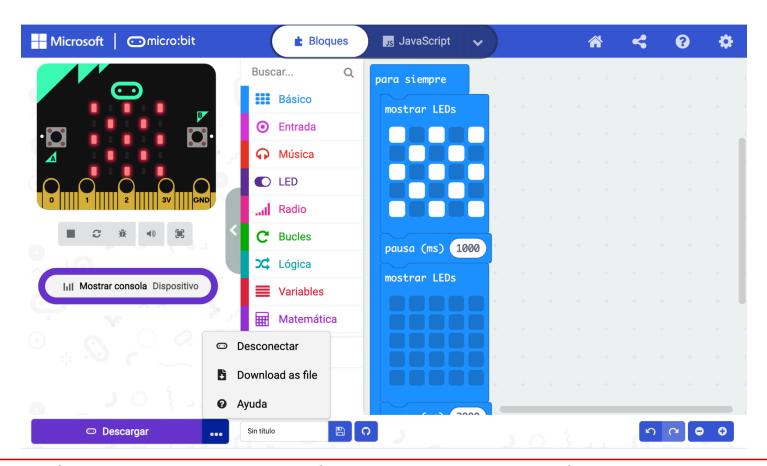
- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Conecta la tarjeta MicroBit al ordenador con el cable USB
 - □ Aparecerá un nuevo disco USB
 - □ Conectar la tarjeta con el programa







- Una vez hecho el programa hay que descargarlo en la tarjeta para probarlo
 - ☐ Descargar el programa una vez conectado





Paso 3: Programa el parpadeo de LEDs



□ Carga el programa Paso_3_EncendiendoUnLED.hex



Observa que realiza el programa:

Enciende los LEDs durante **1000** milisegundos. **Apaga** los LEDs durante **2000** milisegundos.

□ Ejercicio

Modifica el programa para que parpadee más rápido. Pruébalo.

□ Ejercicio

Cambia el dibujo a mostrar por la pantalla

□ Ejercicio

Haz que antes de nada muestre un corazón durante 2 segundos una sola vez

□ Ejercicio

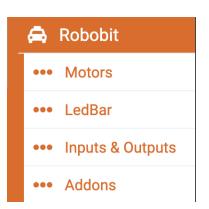
Haz que se muestre tu nombre al inicio.

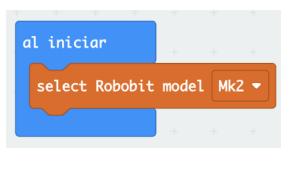


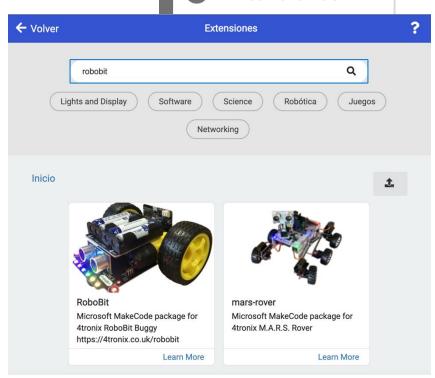
Cargando la extensión ROBOBIT



- Cargando la extensión ROBOBIT
 - Existe una extensión donde hay bloques específicos del robot ROBOBIT que utilizamos.
 - Ya está cargada en los ejemplos pero si se quisiera cargar desde cero habría que cargarla
 - En el menú seleccionar extensiones (en gris)
 - En el recuadro de búsqueda poner "robobit"
 - □ Seleccionar la extensión
 - ☐ En los proyectos hay que indicar el modelo en Al iniciar"







Extensiones





□ Abre el Paso_4_SensorDeDistancia.hex

Para utilizar el sensor de distancia ...

Hay que utilizar el bloque "read sonar as cm" de la sección "Inputs & Outputs" de la extensión Robobit.





También hay que utilizar una variable donde guardar el dato medido

Las variables sirven para almacenar datos

Los bloques relacionados en: ■ Variables

El Bloque "Fijar" guarda un valor en una variable.

Las variables se pueden definir en "Ejecutar una variable"







☐ Abre y descarga el **Paso_4_SensorDeDistancia.hex** en el robot

Observa lo que realiza el programa

Envía al ordenador el valor de distancia medido tras introducirlo en la variable distancia. En el "Consola del dispositivo" deberían aparecer números.





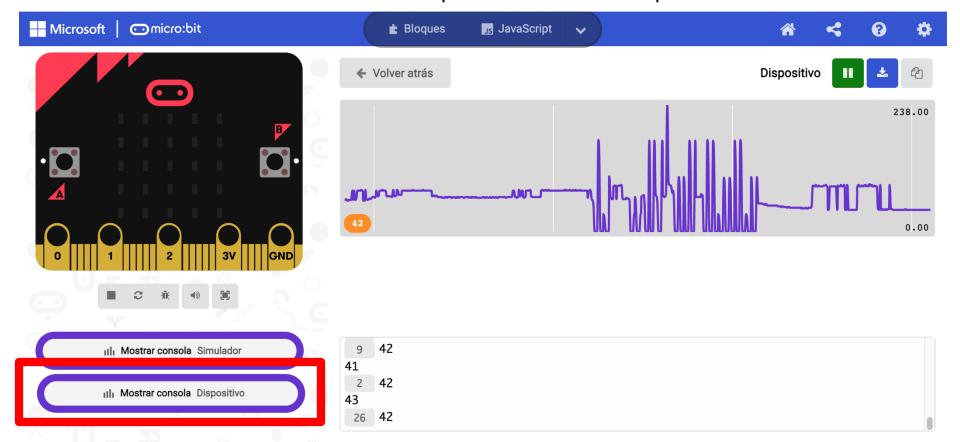




Abre y descarga el Paso_4_SensorDeDistancia.hex en el robot

Observa lo que realiza el programa

Envía al ordenador el valor de distancia medido tras introducirlo en la variable distancia. En el "Consola del dispositivo" deberían aparecer números.







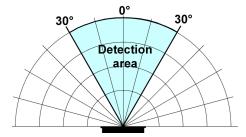
☐ Abre y descarga el **Paso_4_SensorDeDistancia.hex** en el robot

Observa lo que realiza el programa

Envía al ordenador el valor de distancia medido tras introducirlo en la variable distancia. En el "Monitor Serie" deberían aparecer números.







□ Ejercicio

Comprueba las medidas que genera el sensor al mover un objeto delante. Prueba también a desplazarlo lateralmente para ver el ángulo de detección.





Abre y descarga el Paso_4_SensorDeDistancia.hex en el robot

Observa lo que realiza el programa

Envía al ordenador el valor de distancia medido tras introducirlo en la variable distancia. En el "Monitor Serie" deberían aparecer números.





□ Ejercicio

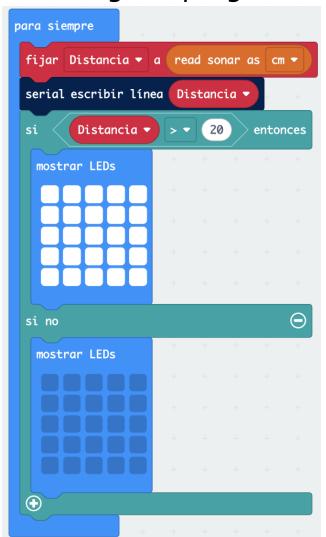
Cambia las unidades de medida en us para ver el tiempo que tarda el ultrasonido en rebotar



Paso 5: Detectando un objeto



☐ Carga el programa Paso_5_DetectandoObjeto.hex



Observa lo que realiza el programa

Si detecta un objeto a más de 20 cm enciende el display y si no lo mantiene apagado.

□ Ejercicio

 Modifica el programa para encienda la pantalla cuando la distancia sea menor de 10cm.

□ Ejercicio

 Modifica el programa para que identifique tres distancias diferentes.

☐ Ejercicio

 Modifica el programa para que el LED parpadee cuando detecte un objeto cercano.



Paso 6: Moviendo los motores



Moviendo los motores

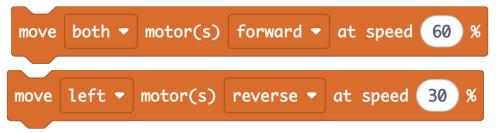
Para mover el robot completo



- Para mover el robot completo durante un intervalo de tiempo
 - Durante ese tiempo el programa no avanza y no se leen los sensores



Para mover los motores de forma individual



Para corregir la asimetría de los motores

```
bias left ▼ by 5 %
```



Paso 6: Moviendo los motores



☐ Carga el programa Paso_6_MoviendoMotores.hex

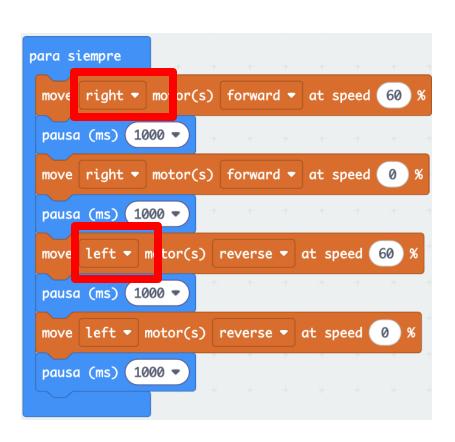




Paso 6: Moviendo los motores



☐ Carga el programa Paso_6_MoviendoMotores.hex



□ Ejercicio:

Cambia el programa para que se mueva todo el robot en línea recta 2 segundos hacia delante y 2 segundos hacia atrás.

□ Ejercicio

 Modifica el programa para que el robot realice la trayectoria de un cuadrado girando en las esquinas.



Paso 7: Detectando obstáculos

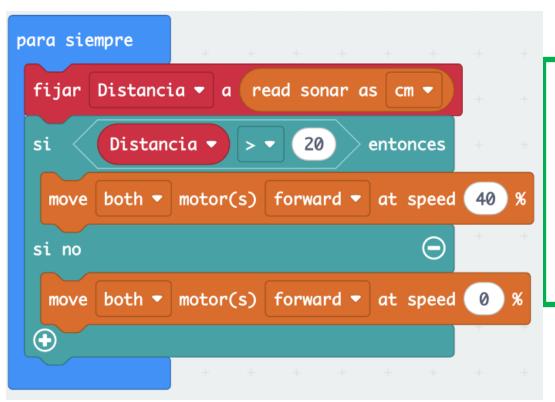


□ Carga el programa Paso_7_DetectandoObstaculos.hex

Observa lo que realiza el programa

Si la distancia (que lee el ultrasonidos) es mayor que 20 cm, se mueven las dos ruedas.

Si no, es decir, si la distancia es menor o igual que 20 cm, se paran.



□ Ejercicio

Modifica el programa para que el robot gire en el sitio y, cuando detecte un **objeto a menos de 10 cm**, vaya a por él en línea recta.



Paso 8: Leyendo los sensores de suelo

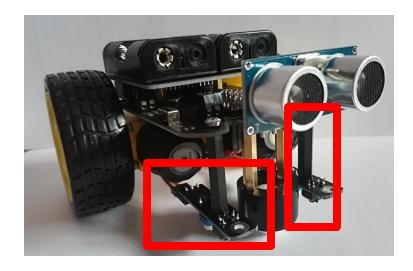


☐ Carga el programa Paso_8_SensoresSuelo.hex

Para utilizar los sensores de suelo_...

Se utilizan los bloques Read Line Sensor indicando el sensor (derecho o izquierdo). Puede tomar el valor 0 (blanco) o 1 (negro).





Los sensores tienen un led que se enciende cuando detecta blanco y se apaga cuando detecta negro. El umbral puede ajustarse con un destornillador



Paso 8: Leyendo los sensores de suelo



☐ Carga el programa Paso_8_SensoresSuelo.hex

Observa que realiza el programa:

Representa en pantalla una línea del lado del sensor que detecta blanco



Si (detecta blanco solo sensor izquierdo)

Pinta línea a la derecha

Si no, si (detecta solo sensor derecho)

Pinta línea a la izquierda

Si no, si detectan los dos sensores Pinta dos líneas

Si no

No pinta nada



Paso 9.a: Nos movemos sin salirnos



□ Teniendo en cuenta todo lo visto anteriormente, realiza los siguientes ejercicios:

l Ejercicio:

 Realiza un programa que haga que el robot avance cuando esté sobre negro y se detenga cuando llegue a la línea blanca.

□ Ejercicio:

- Realiza un programa que haga para que el robot avance hasta que detecte blanco y en ese momento, retroceda durante un cierto tiempo, gire, y siga avanzando.
- El robot debería moverse dentro del Tatami sin salirse.

□ Ejercicio:

- Modifica el programa para que, si mientras está girando detecta un oponente, vaya a por él.
- Este ya es un programa que detecta al oponente



Paso 9.b: Luchador de sumo



☐ Ya estás listo para programar el luchador de sumo. Busca al oponente, detéctale, empújale, pero no te salgas tú solo.



Mini-Taller de Robótica Móvil

Programa un robot de sumo











Actividad orientada a estudiantes de Enseñanza Secundaria y Bachillerato