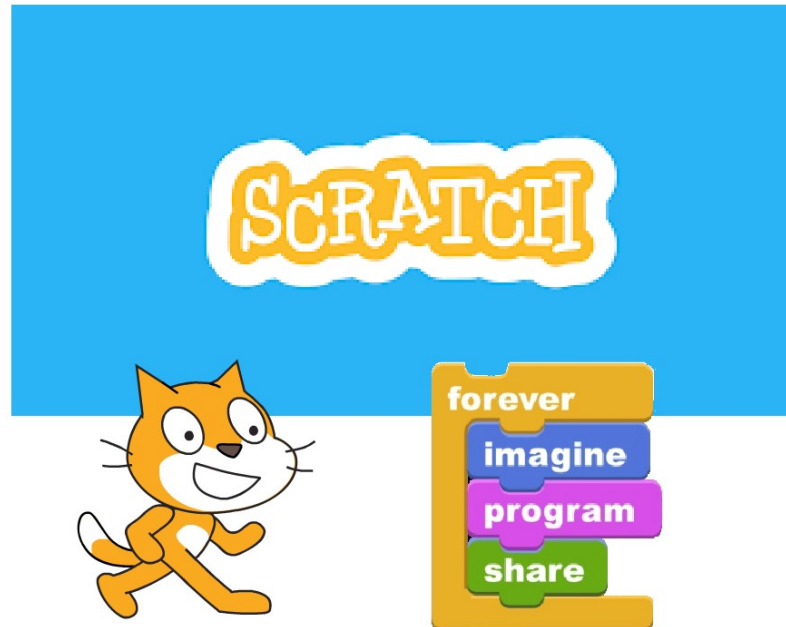


SCRATCH

la aventura de la programación



Jesús Urcera López

Nov. 2018 – May. 2019

• Licencia y reconocimientos

- *Esta obra está bajo una licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional.*

<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

- *Todo el material de soporte de Scratch, incluyendo imágenes, sonidos, videos código de ejemplo, que ayuda a los usuarios a construir sus proyectos está licenciado bajo licencia [Creative Commons Attribution-ShareAlike 2.0 license](https://creativecommons.org/licenses/by-sa/4.0/deed.es).*

https://scratch.mit.edu/terms_of_use

- *Todo el material de soporte de mBlock pertenece a su propietario Makeblock Co. Ltd. , siendo su uso en este documento, única y exclusivamente, con fines docentes.*

<https://www.makeblock.com/terms-of-use>

- *Todo el material de soporte de mBot pertenece a sus propietarios Makeblock Co. Ltd. y Smart Products Connection S.A., siendo su uso en este documento, única y exclusivamente, con fines docentes.*

<http://www.spc-makeblock.es/aviso-legal/>

- **Aplicaciones en Robótica con mBot**
 - *¿Qué es un robot?*
 - *Sensores*
 - *Actuadores*
 - *Placa de control mCore para mBot*
 - *Entorno de desarrollo*
 - *Programación*
 - *Luces y sonidos*
 - *Esquiva objetos*
 - *Sigue líneas*

• ¿Qué es un robot?

- *Es un sistema electromecánico programable y autónomo capaz de reaccionar a cambios en su entorno según las condiciones dadas en la programación que lo gobierna.*
- *La reacción se genera mediante **actuadores** que convierten magnitudes eléctricas en magnitudes físicas o químicas.*
- *Los cambios en el entorno se obtienen de **sensores** que transforman magnitudes físicas o químicas en señales eléctricas comprensibles por el robot. Existen dos tipos de sensores: los analógicos que proporcionan valores continuos en un rango y los digitales que devuelven solo dos valores, encendido y apagado o “1” y “0”.*
- *En el **programa** se define como debe actuar el robot ante cada uno de los estímulos externos. El programa se ejecuta en una **placa de control** programable a la que están conectados sensores y actuadores.*

Curso de iniciación a SCRATCH – C.E.I.P. La Garena

- Robots educativos



ArduBot



mBot



crumbleBot



RaspiRobot



Zowi



BeeBot

• Sensores

- Son elementos que generan señales o información eléctrica partiendo de una magnitud física o química.
- Características:
 - Funcionan dentro de un rango de variación de la magnitud que miden.
 - Linealidad: indica si la salida es proporcional a la entrada durante todo el rango de medición del sensor.
 - Resolución: que es la mínima variación de la magnitud a medir que pueden detectar.
 - Precisión: que nos indica la variación entre el valor medido y el valor real de la magnitud medida.
- Se pueden clasificar por la magnitud que miden: de choque, de luz, de temperatura, de humedad, de campo magnético, de CO2, etc.
- También se pueden clasificar por el tipo de señal que devuelven: analógica o digital. Dentro de las analógicas pueden ser de tipo resistivo, de variación de tensión, de variación de corriente y con respecto a las digitales pueden devolver un valor digital básico (0/1) o una señal codificada siguiendo una secuencia o protocolo.
- Para convertir señales analógicas en digitales y poder tratarlas digitalmente se usa un ADC (convertor analógico-digital).



LDR



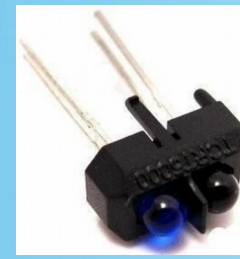
Temperatura



Choque



Gases



Infrarrojos



Ultrasonidos

• Actuadores

- Son elementos que convierten señales o información eléctrica producida por el controlador en una magnitud normalmente física.
- Se pueden clasificar por la magnitud o señal que generan:
 - Movimiento
 - Luz
 - Color
 - Sonido
 - Posición
 - etc.



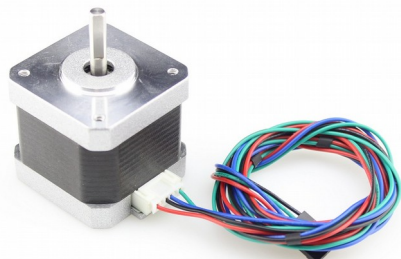
LED



Motor DC



Servomotor



Motor paso a paso



Zumbador



LED RGB

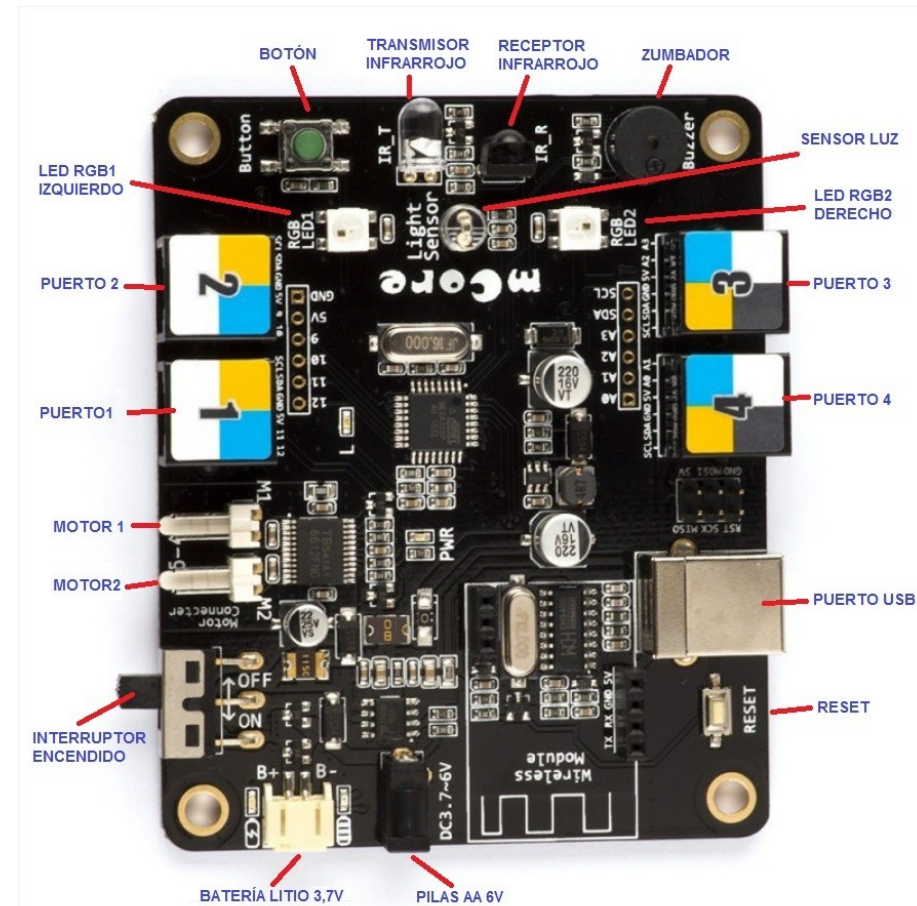


Electroválvula

Curso de iniciación a **SCRATCH** – C.E.I.P. La Garena

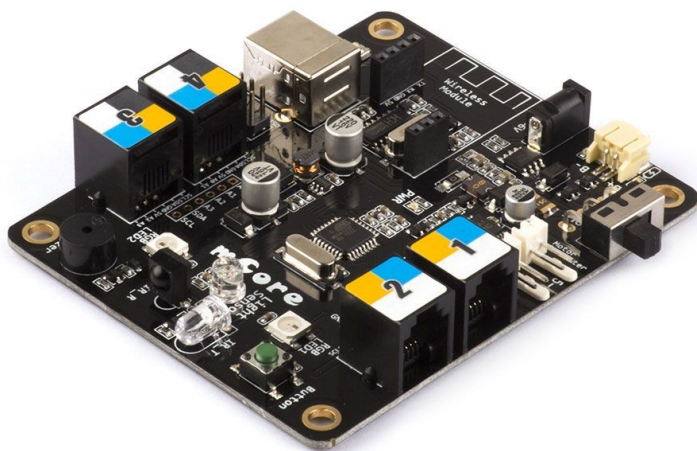
• Placa mCore para mBot

- Placa: mCore (basada en Arduino)
- Microcontrolador: Atmega328
- Alimentación: 4 pilas AA o batería de litio de 3,7V
- Accesorios: Sensor de luz, botón, infrarrojos, ultrasónico, seguidor de línea, zumbador, Led RGB, transmisor.
- Comunicación: USB, Bluetooth, Serie inalámbrica 2.4G
- Conexiones: 2 motores y 4 sensores
- Dimensiones: 17 x 13 x 9 cm (montado)
- Peso: 400gr



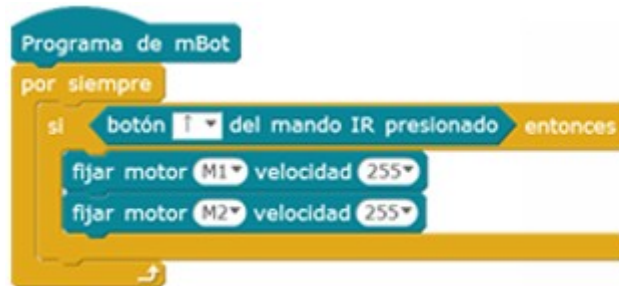
• Placa mCore: módulos de expansión.

- Los módulos o sensores que podemos conectar a la placa vienen clasificados por colores. Dicho color debe corresponder con el color del puerto al que queremos conectarlo. Por ejemplo, el puerto 1 dispone de tres colores (amarillo, azul y blanco), esto indica que podremos conectar en él cualquier módulo cuyo color del RJ25 sea uno de esos colores.
- Los colores son: Rojo (motores), Amarillo (interfaz digital), Azul (interfaz digital dual), Gris (puerto serie, Bluetooth), Negro (interfaz analógica y dual) y Blanco (Puerto I2C). Este sistema hace que la conexión de accesorios sea muy fácil y sencilla.



• El robot educativo mBot

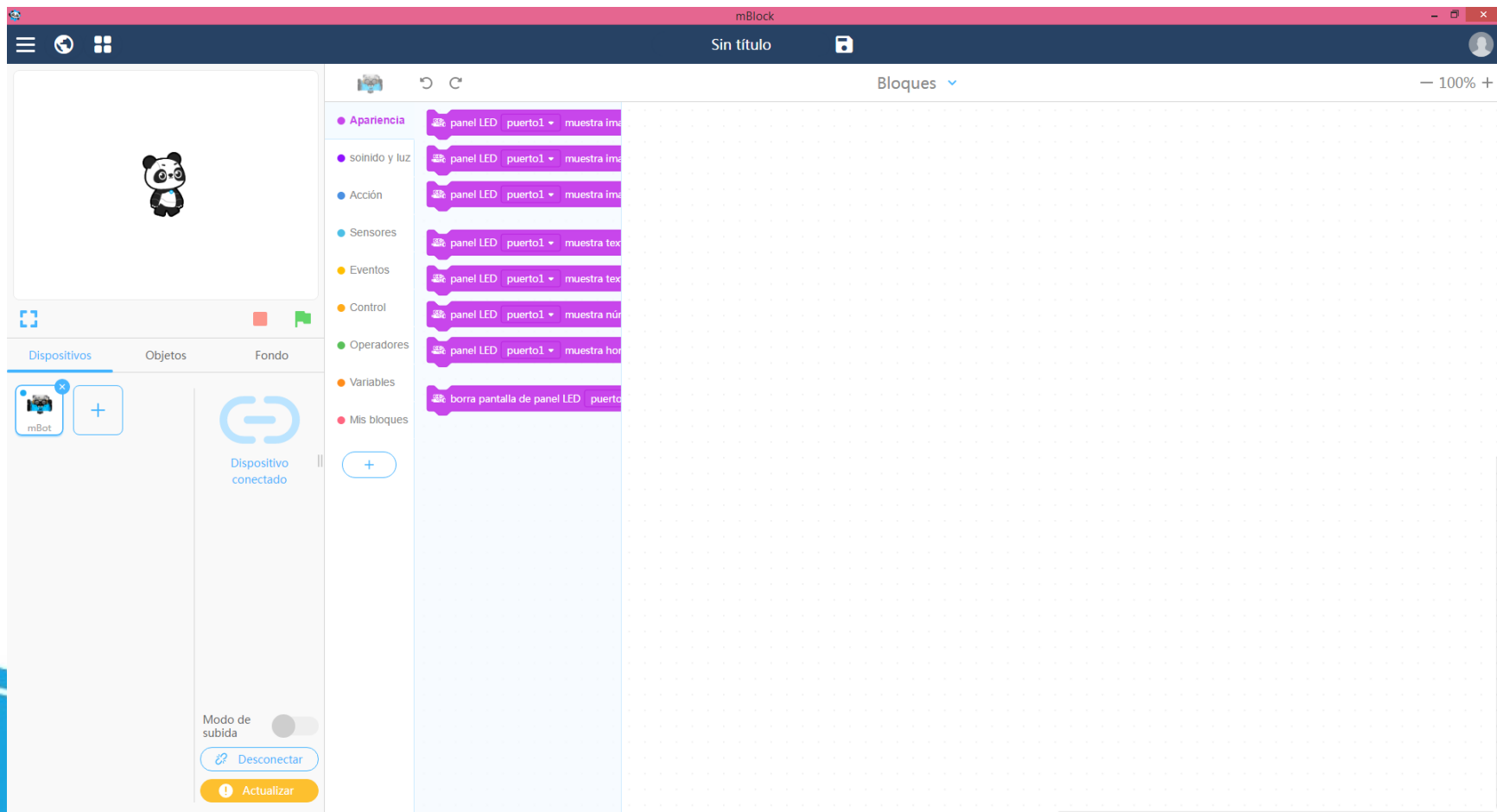
- mBot es un robot educativo ideal para iniciarse en la programación y robótica desde educación primaria. Está basado en Arduino Uno es muy sencillo de utilizar ya que no necesita cableado ni soldaduras gracias a sus conectores RJ25, parecidos a los de teléfono y a la programación visual por bloques similar a Scratch.



Robot mBot de makeBlock

Curso de iniciación a SCRATCH – C.E.I.P. La Garena

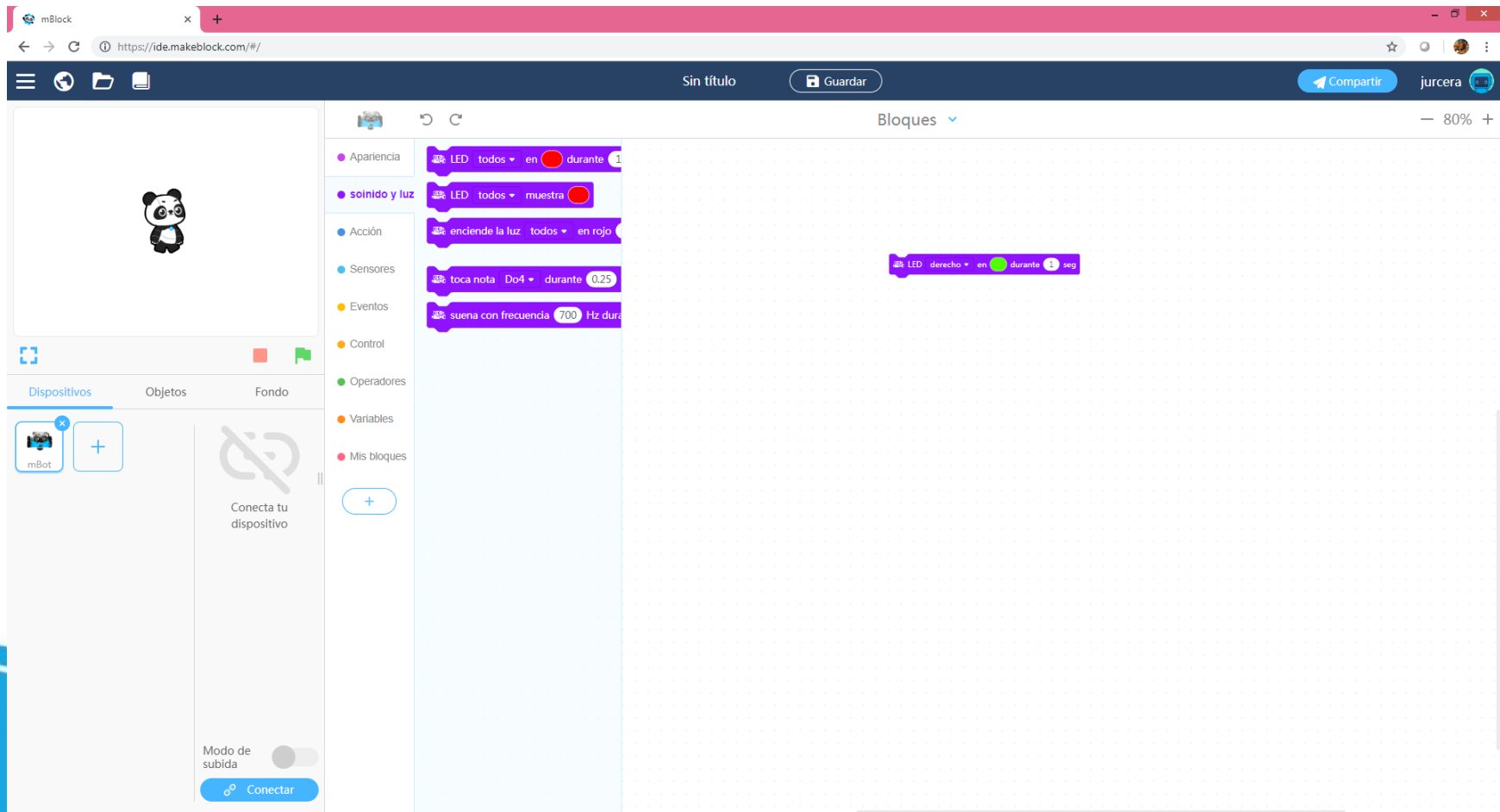
- El entorno de desarrollo offline mBlock 5
 - Permite trabajar sin conexión a Internet. Con conexión a Internet es posible acceder a nuestros proyectos online. Descarga: <http://www.mblock.cc/mblock-software/>
 - Es necesario instalar mLink para poder conectarse con el robot.



Curso de iniciación a SCRATCH – C.E.I.P. La Garena

• El entorno de desarrollo (IDE) online

- Permite realizar los programas y compartirlos online sin necesidad de instalar el IDE.
- Es necesario acceso a Internet e instalar mLink para poder conectarse con el robot.
- <https://ide.makeblock.com/>

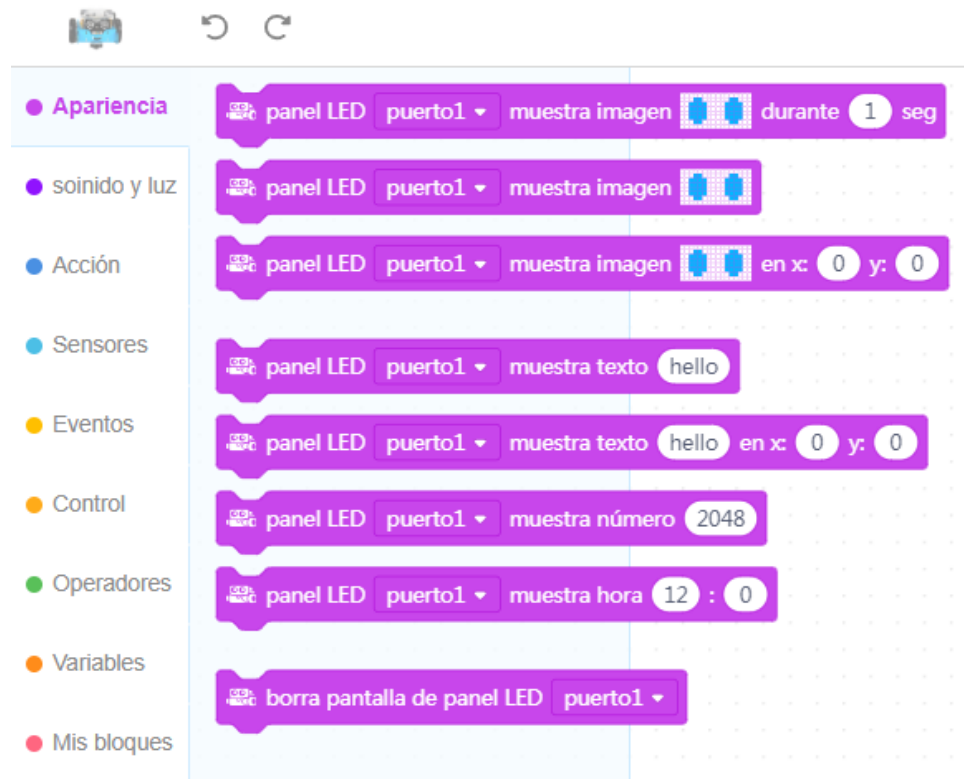


• Programación

- La programación puede realizarse de tres maneras:
 - Mediante el IDE online en modo bloques y en modo texto (Arduino).
 - Mediante el IDE offline mBlock en modo bloques y en modo texto (Arduino).
 - Desde el IDE de Arduino instalando las librerías de MakeBlock.
- En este curso solo se verá la programación gráfica por bloques.
- Existen nueve grupos de bloques similares a Scratch pero orientados a mBot.
 - Apariencia
 - Sonido y luz
 - Acción
 - Sensores
 - Eventos
 - Control
 - Operadores
 - Variables
 - Mis bloques

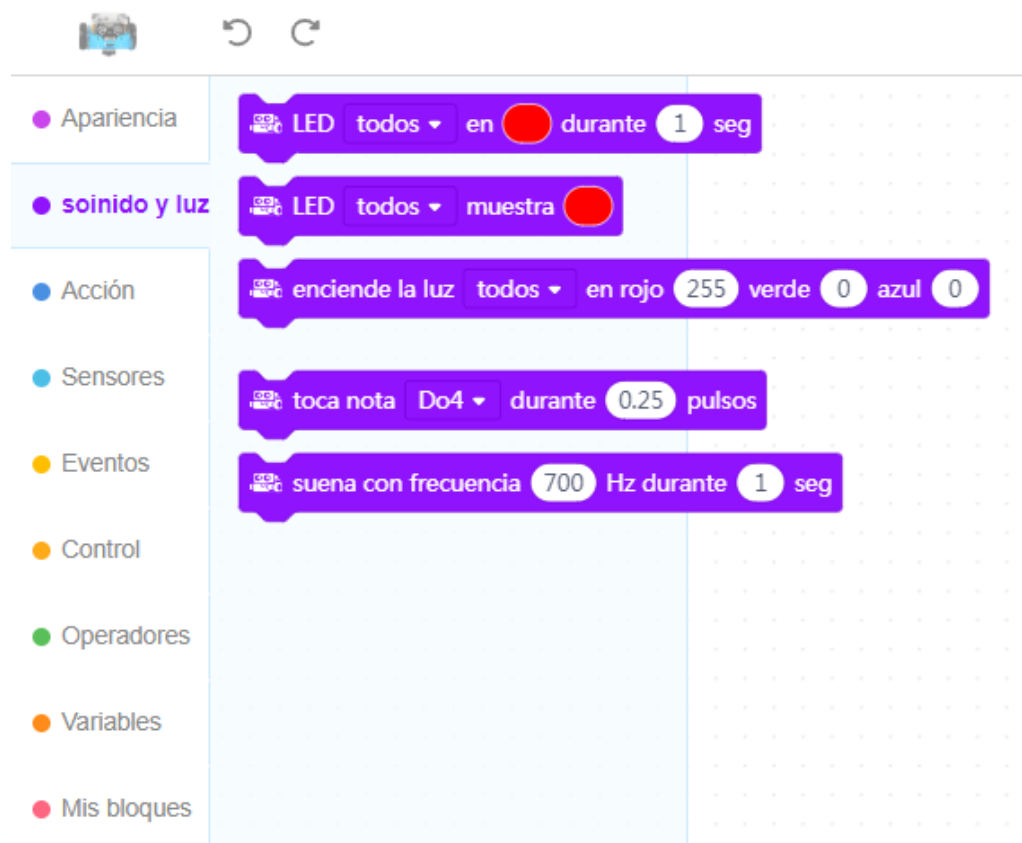
• Bloques de Apariencia

- Para mBot aparecen los siguientes bloques de apariencia. Es necesario disponer del módulo **panel LED** para poder usarlos.



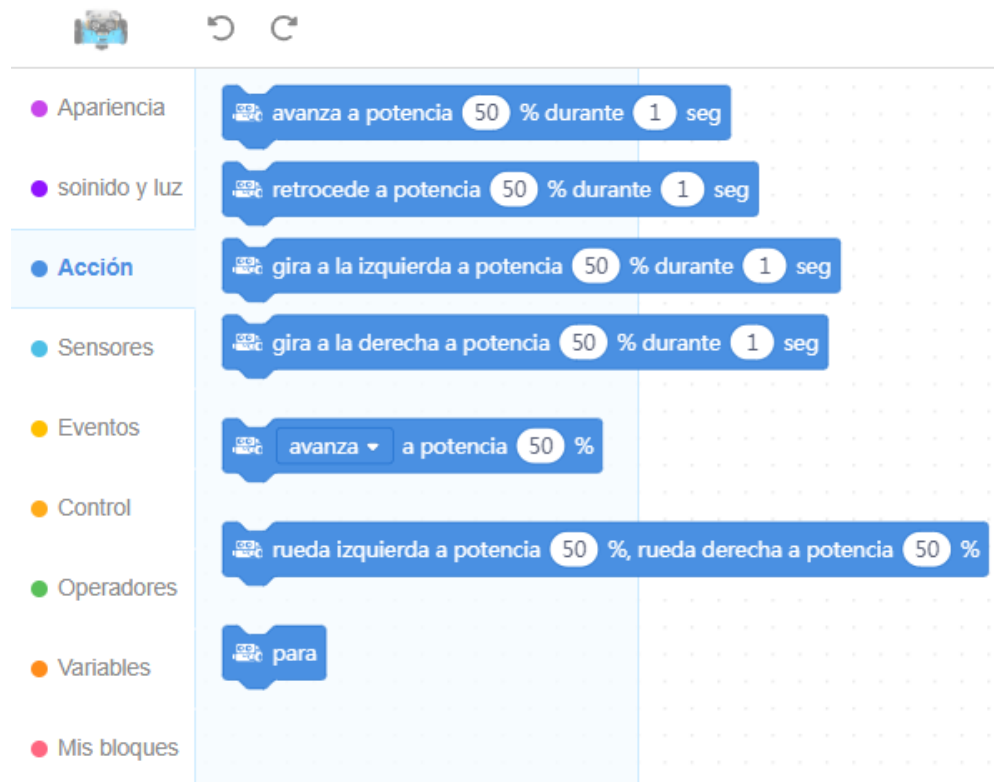
• Bloques de Sonido y luz

- Permiten trabajar con los dos LED RGB derecho e izquierdo y con el zumbador. Para mBot aparecen los siguientes bloques de sonido y luz.



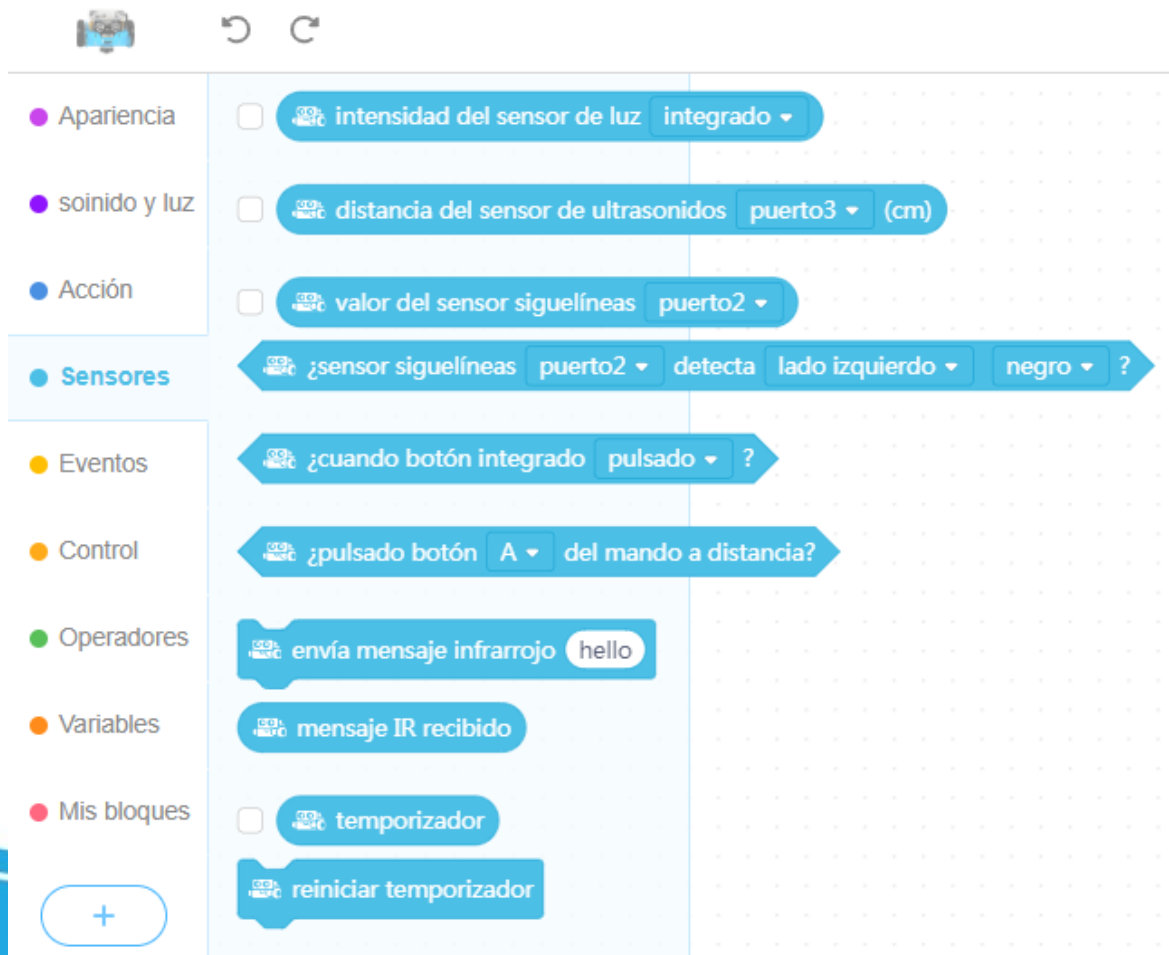
• Bloques de Acción

- Permiten trabajar con los dos motores derecho e izquierdo. Para mBot aparecen los siguientes bloques de Acción.



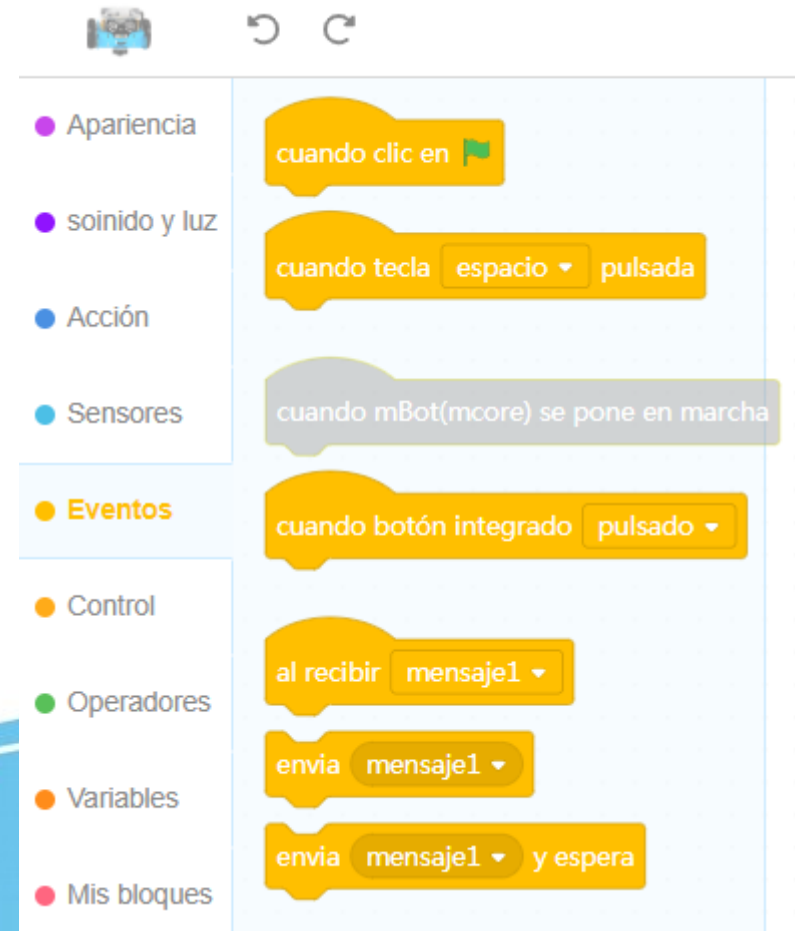
• Bloques de Sensores

- Permiten trabajar con los sensores de mBot. Para mBot aparecen los siguientes bloques de Sensores.



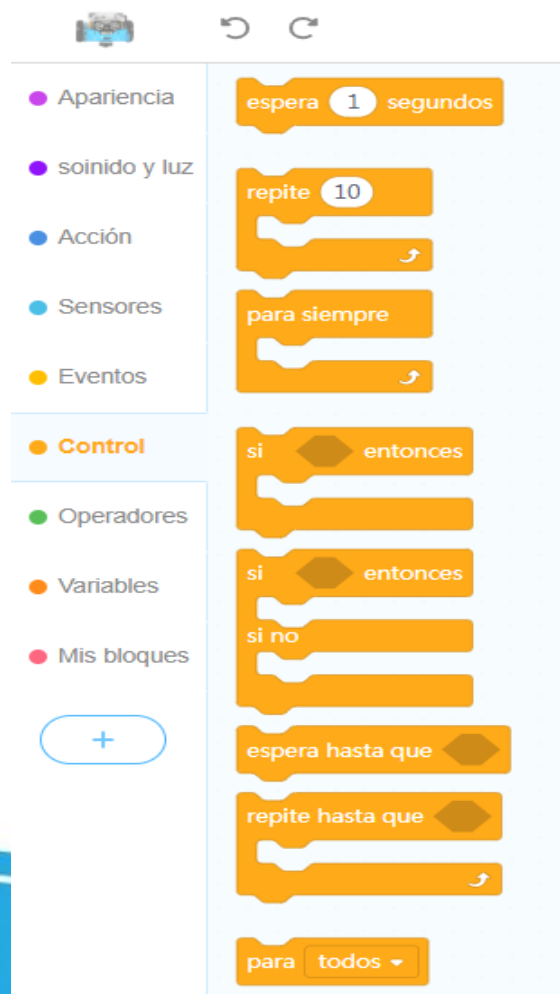
• Bloques de Eventos

- Permiten trabajar con eventos en modo de subida (izquierda) y en modo normal (derecha). Para mBot aparecen los siguientes bloques de Eventos.



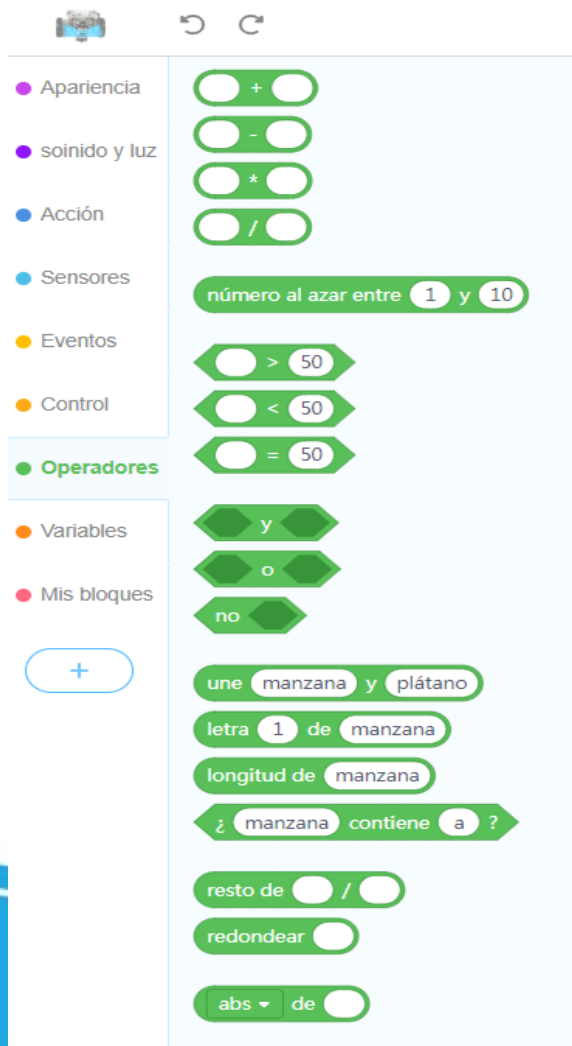
• Bloques de Control

- Permiten trabajar con bloques de control. Para mBot aparecen los siguientes bloques de Control.



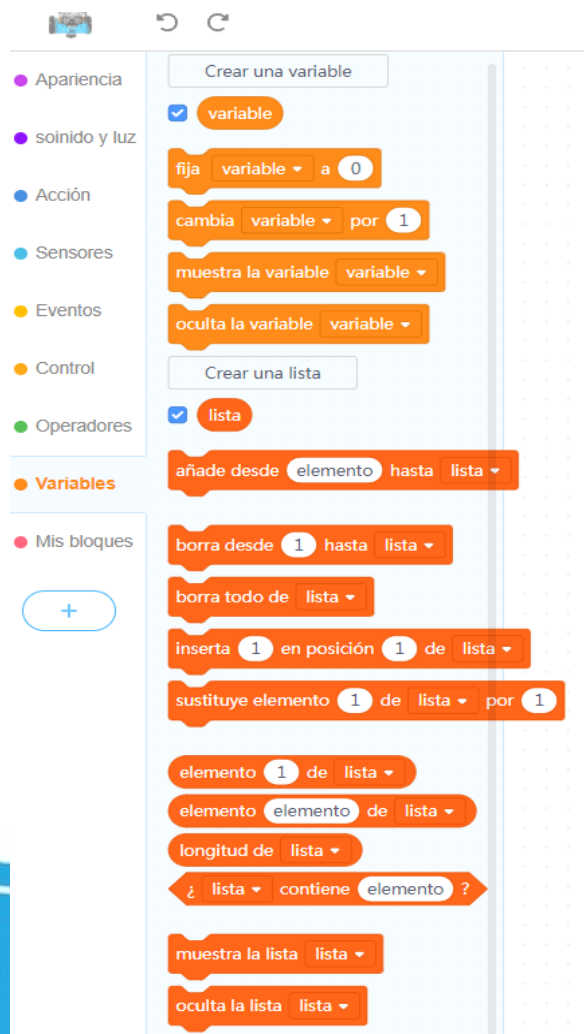
• Bloques de Operadores

- Permiten trabajar con bloques de operadores. Para mBot aparecen los siguientes bloques de Operadores.



• Bloques de Variables

- Permiten crear y trabajar con variables y listas. Para mBot aparecen los siguientes bloques de Variables.



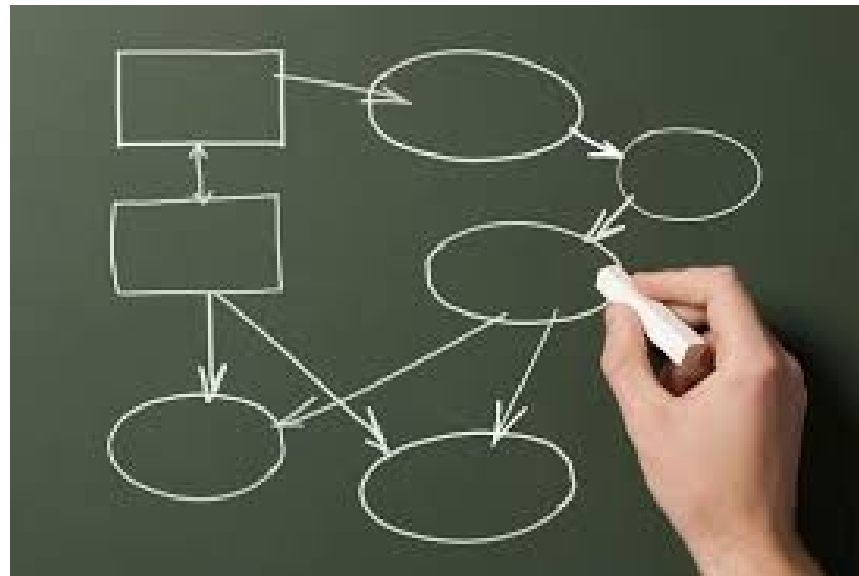
• Bloques de Mis bloques

- Permiten crear y trabajar con mis propios bloques. Para mBot aparecen los siguientes bloques para trabajar con Mis Bloques.

The screenshot displays the Scratch 'Mis bloques' (My Blocks) interface. On the left, a sidebar lists categories: Apariencia, sonido y luz, Acción, Sensores, Eventos, Control, Operadores, Variables, and Mis bloques. The 'Mis bloques' category is active, showing a 'Crear un Bloque' button and three custom blocks labeled 'parte1', 'parte2', and 'parte3'. A 'cuando clic en' event block is connected to a 'repite' loop containing 'parte1', 'parte2', 'parte1', and 'parte3'. A 'Crear un bloque' dialog box is open, showing a text input for 'nombre del bloque' and four options: 'Añadir una entrada número', 'Añadir una entrada texto', 'Añadir una entrada verdadero/falso', and 'Añadir una etiqueta'. The main workspace shows three 'definir' blocks for 'parte1', 'parte2', and 'parte3', each containing a sequence of actions like 'LED todos muestra', 'toca nota', and 'LED todos muestra' with various parameters.






• Algoritmo

- *Es una serie de instrucciones o pasos ordenados que se necesitan para realizar una actividad o resolver un problema.*



• Diagrama de flujo

- *Representa de forma gráfica los pasos necesarios para realizar una tarea o resolver un problema.*
- *Elementos y funciones.*

Símbolo	Nombre	Función
	Inicio / Final	Representa el inicio y el final de un proceso
	Línea de Flujo	Indica el orden de la ejecución de las operaciones. La flecha indica la siguiente instrucción.
	Entrada / Salida	Representa la lectura de datos en la entrada y la impresión de datos en la salida
	Proceso	Representa cualquier tipo de operación
	Decisión	Nos permite analizar una situación, con base en los valores verdadero y falso

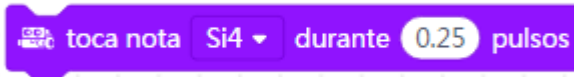
• Práctica 1: Música y luces

- *En este ejemplo mBot va a interpretar una melodía que va a acompañar con luces.*
- *La melodía es “Pequeña canción” para flauta que adaptaremos para que la interprete mBot.*
- *Asignaremos a cada nota un color y haremos que el led derecho y el izquierdo cambien de color con las notas.*
- *Al existir una parte que se repite generaremos bloques propios para no repetir partes del código.*
- *Como solo lo vamos a ejecutar una sola vez y en secuencia no vamos a necesitar un diagrama de flujo.*

• Práctica 1: Música y luces

- Con el bloque **tocar nota durante** iremos construyendo la melodía ajustando la duración de las notas y con el bloque **LED todos muestra** elegimos colores.
- Ajustaremos la duración de la blanca a 1 pulso, de la negra a 0,50 pulsos y de la corchea a 0,25 pulsos para mantener los tiempos según indica la partitura.

Redonda	○	4 Tiempos
Blanca	♪	2 Tiempos
Negra	♩	1 Tiempo
Corchea	♪♪	1/2 Tiempo
Semicorchea	♪♪♪	1/4 Tiempo
Fusa	♪♪♪♪	1/8 Tiempo
Semifusa	♪♪♪♪♪	1/16 Tiempo



Flauta dulce (3º de Primaria)

NOTAS EN LA FLAUTA:

Mano izquierda

Mano derecha

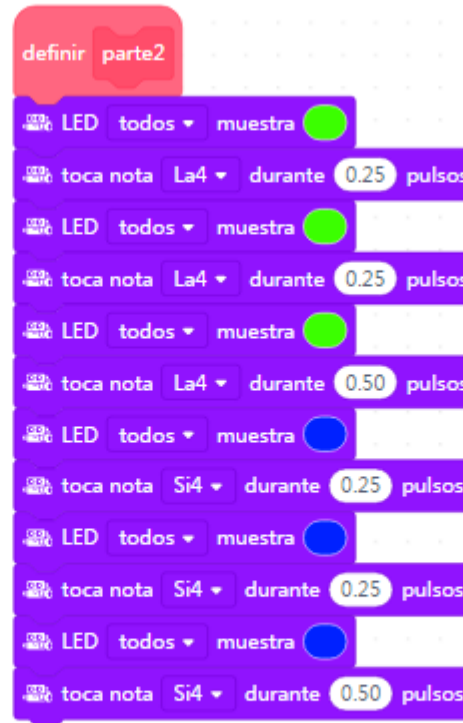
Sol La Si

PEQUEÑA CANCIÓN:

Si La Sol La Si Si Si La La La Si Si Si Si La Sol La Si Si Si La La Si La Sol

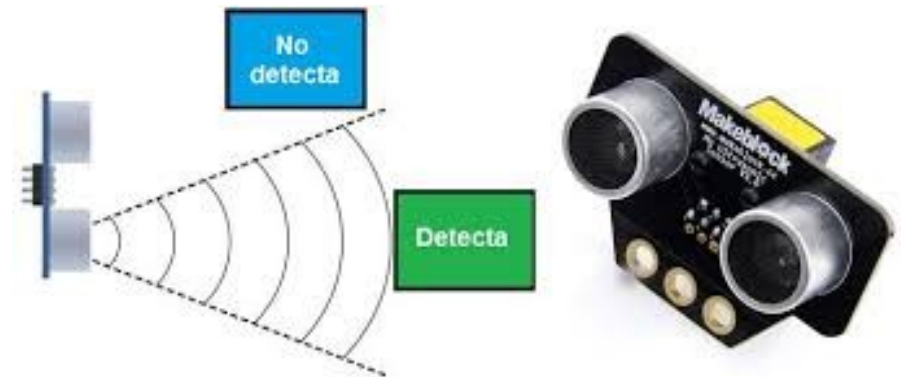
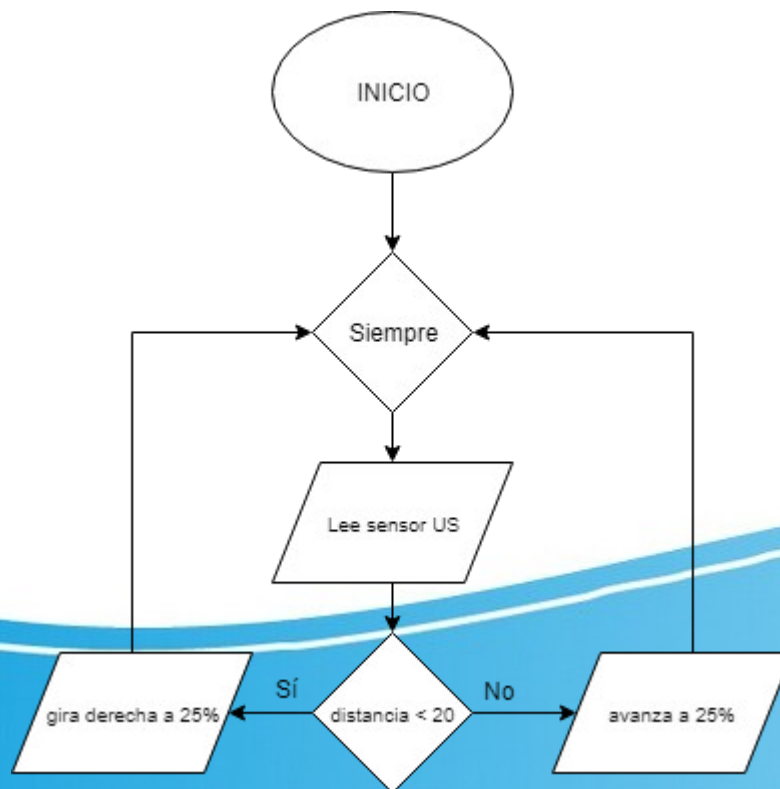
• Práctica 1: Música y luces

- *El resultado final sería el siguiente.*



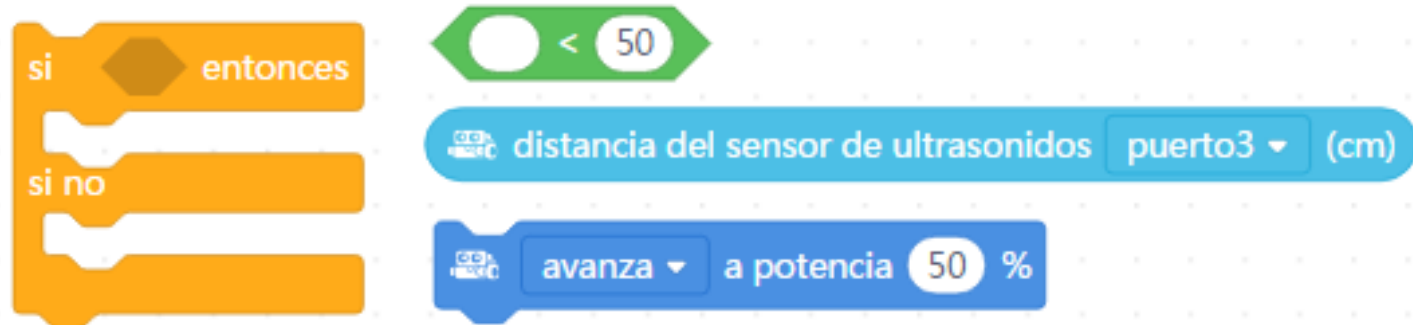
• Práctica 2: Evitar objetos

- *En este ejemplo mBot va a desplazarse evitando chocar con los objetos que se encuentre en su camino.*
- *Usaremos el sensor de ultrasonidos, conectado al puerto 3, para detectar los obstáculos.*
- *El diagrama de flujo sería el siguiente.*



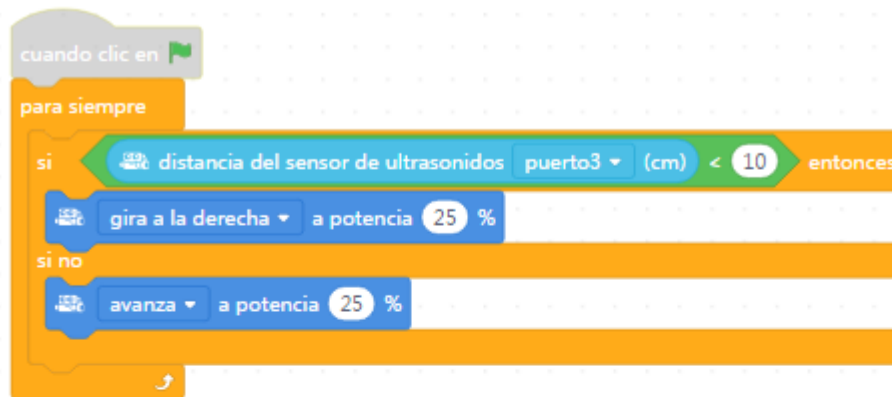
• Práctica 2: Evitar objetos

- Usaremos el bloque **distancia del sensor de ultrasonidos** del grupo **Sensores** para medir la distancia.
- Usaremos el comparador **menor que** del grupo **Operadores** para comparar la distancia medida por el sensor de ultrasonidos.
- Usaremos el bloque **avanza a potencia** del grupo **Acciones** para mover los motores.
- Usaremos la condición **si entonces si no** del grupo **Control** para tomar la decisión de que hacer.




• Práctica 2: Evitar objetos

- *El programa sería.*



```
cuando clic en [bandera verde]
para siempre
si [distancia del sensor de ultrasonidos puerto3 (cm) < 10] entonces
  gira a la derecha a potencia 25 %
si no
  avanza a potencia 25 %
```

Para ejecutar conectado por USB

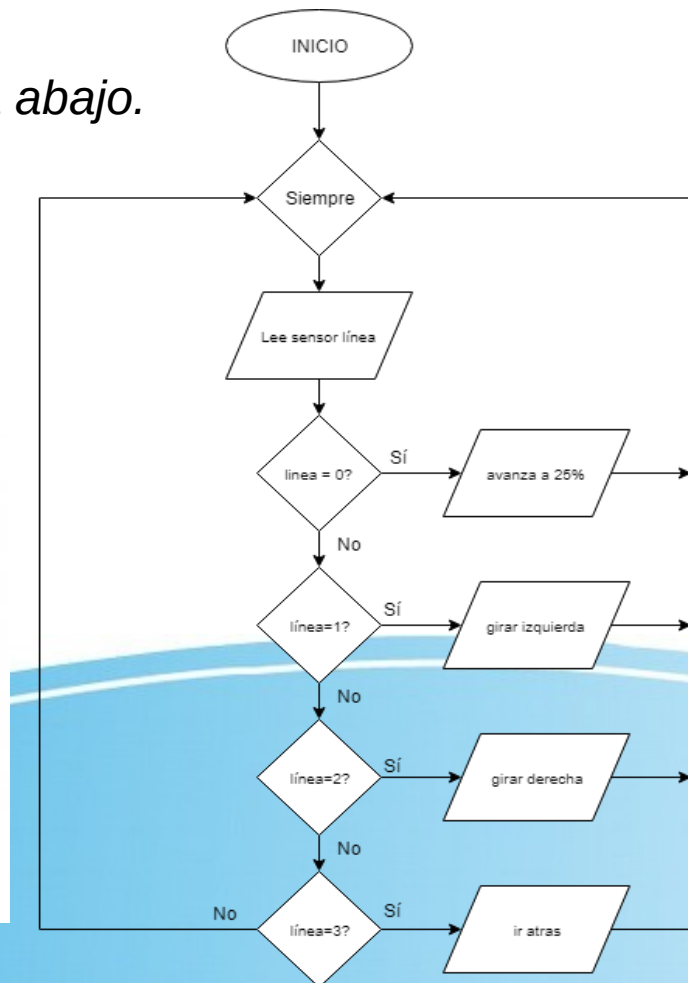


```
cuando mBot(mcore) se pone en marcha
para siempre
si [distancia del sensor de ultrasonidos puerto3 (cm) < 10] entonces
  gira a la derecha a potencia 25 %
si no
  avanza a potencia 25 %
```

Para ejecutar con mBot desconectado

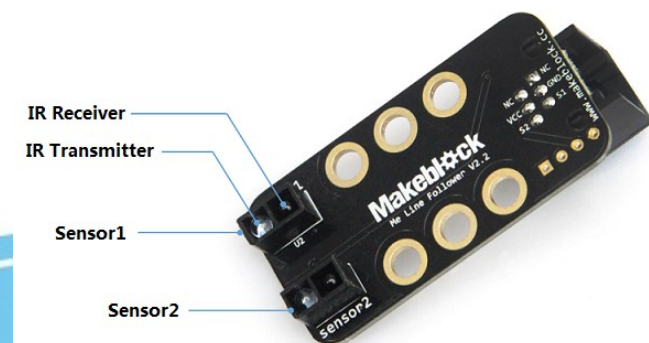
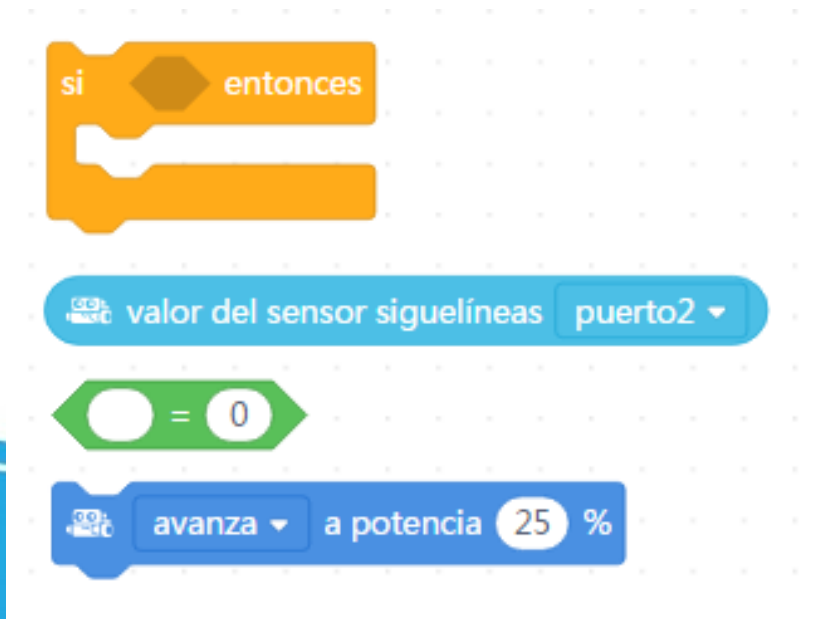
• Práctica 3: Seguir línea

- En este ejemplo mBot va a desplazarse siguiendo una línea negra, de un determinado ancho, dibujada en el suelo.
- Usaremos el sensor de líneas, conectado al puerto 2, para detectar si está o no sobre la línea.
- El sensor de línea funciona como se indica abajo.
- El diagrama de flujo sería el siguiente.



• Práctica 3: Seguir línea

- Usaremos el bloque **valor del sensor siguelíneas** del grupo **Sensores** para saber si estamos sobre la línea o no.
- Usaremos el comparador **igual que** del grupo **Operadores** para comparar la distancia medida por el sensor de ultrasonidos.
- Usaremos el bloque **avanza a potencia** del grupo **Acciones** para mover los motores.
- Usaremos la condición **si entonces** del grupo **Control** para tomar la decisión de que hacer.



- **Práctica 3: Seguir línea**
 - *El programa sería.*

```
cuando mBot(mcore) se pone en marcha
espera hasta que ¿cuando botón integrado pulsado ?
para siempre
  si valor del sensor siguelíneas puerto2 = 0 entonces
    avanza a potencia 25 %
  si valor del sensor siguelíneas puerto2 = 1 entonces
    gira a la izquierda a potencia 25 %
  si valor del sensor siguelíneas puerto2 = 2 entonces
    gira a la derecha a potencia 25 %
  si valor del sensor siguelíneas puerto2 = 3 entonces
    retrocede a potencia 25 %
```



Curso de iniciación a **SCRATCH** – C.E.I.P. La Garena

- *Cuestiones y dudas.*

