

Explore STEM & Coding with

EDU:BIT

on start

start melody power up ▼ repeating once ▼

set all RGB pixels to



forever

if IR sensor triggered then

Set servo S1 ▼ position to 40 degrees

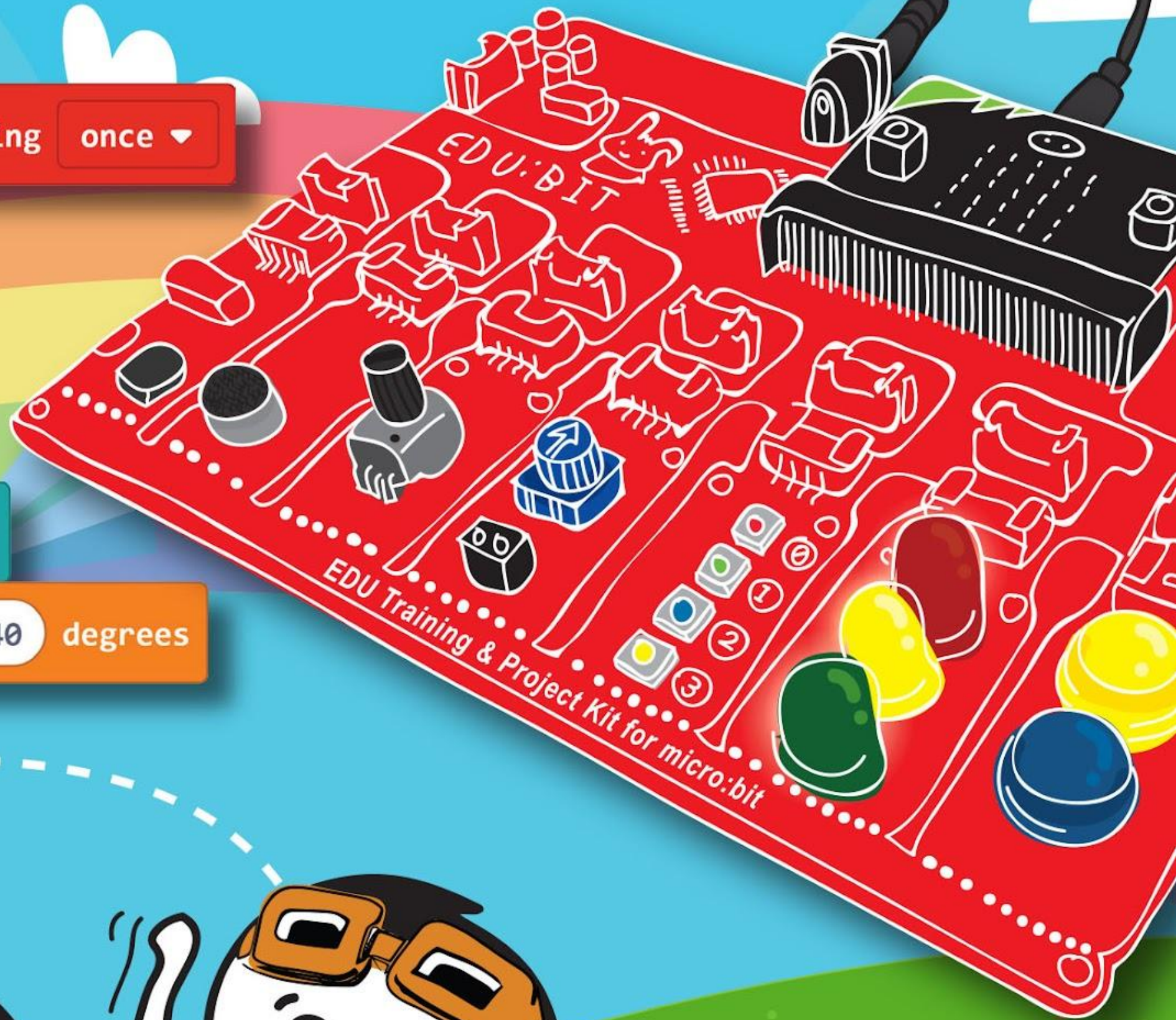
show leds



else

Set servo S1 ▼ position to 120 degrees

show arrow East



NOTA DE EDUTEAM @ CYTRON

Querido _____ ,
(nombre del niño)

¿Alguna vez has oído hablar de micro: bit? Es una pequeña placa programable diseñada en el Reino Unido y distribuida ampliamente en todo el mundo para animar a los niños a aprender a programar, de una manera fácil y divertida.

Los ingenieros de Cytron lo han mejorado haciendo la placa EDU: BIT para que aprendas a programar poco a poco. Tienes el Music Bit con zumbador piezoeléctrico y jack de audio para reproducir música, el Bit Sound para detectar sonidos, el Bit Potenciómetro para control analógico, el Bit Infrared para detectar objetos, el Bit RGB para mostrar luces de colores, el Bit Semáforo con los LED rojos, amarillos y verdes, y finalmente el Button Bit, que tiene una versión más grande de los botones ya presentes en la tarjeta micro: bit. Con este kit también tienes un Motor DC y un Servo Motor disponibles para jugar. Que bonito todo esto

Podemos empezar, ¿verdad? En las próximas páginas exploraremos y crearemos algunos juegos clásicos de la infancia como piedra, papel o tijera, serpientes y escaleras, trastos, show de talento, tornado, simón dice y otros juegos divertidos. ¡Sigue la guía paso a paso para construir los juegos y diviértete con tus amigos! Eres libre de modificar el código para crear tu propia versión de los juegos.

Al final de cada capítulo, habrá una prueba que requiere que aplique lo que ha aprendido para crear una aplicación para su clase. Pruébalo, y si tienes problemas, siempre estaremos aquí para ayudarte.

¿Estás listo? entonces partamos para este viaje emocionante: ¡diviértete aprendiendo y explorando!

Saludos,

Adán y ana



Explora STEM y programación con EDU: BIT Kit Tutorial y Proyectos

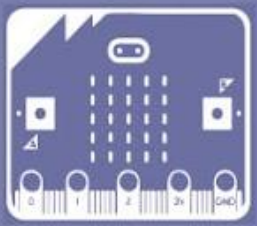
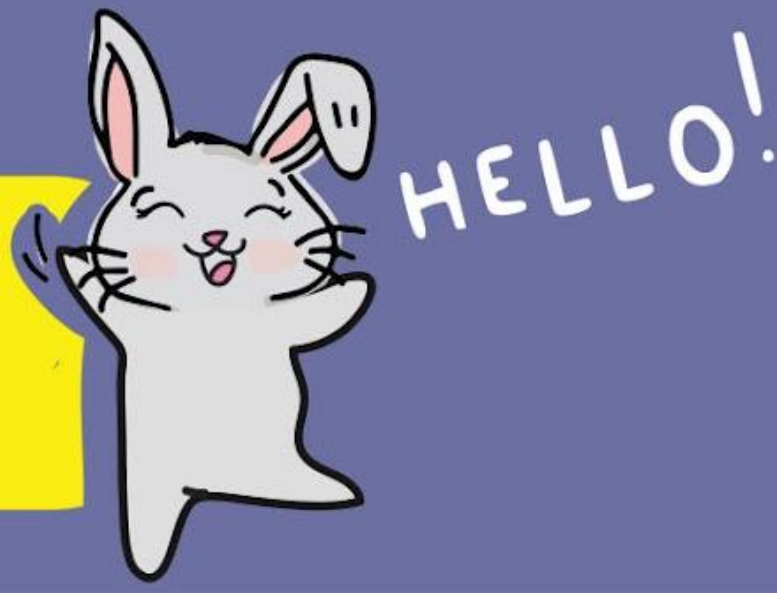
Escrito por Cheryl Ng, SC Lim y Adrian Teo
Traducido por José Manuel Ruiz Gutiérrez

Ilustrado por Suhana Oazmi

2022

Publicado por  **Cytron**
Technologies

Contenido



Capítulo 1: ¡Hola, mundo! (Matriz de LED en micro: bit)

Al principio y para siempre

1 - 11



Capítulo 2: ¡Juguemos a piedra, papel o tijera! (Botón de bits)

Programación de variables y eventos

12 - 25



Capítulo 3: Escuchemos algo de música ~ (Bit Music))

Funciones en programación

26 - 38



Capítulo 4: Victoria, derrota o empate ~ (Bit Semaphore)

Salida digital

39 - 47



Capítulo 5: Dado digital... Lancémoslo ~ (Bit Infrared))

Entrada digital, matriz y bucles while

48 - 60



Capítulo 6: Tackle (potenciómetro de bits)

Entrada analógica, programación condicional

61 - 74



Capítulo 7: ¡Escuchemos los aplausos! (Sonido de bits)

Cambiar entre modos en un programa

75 - 87



Capítulo 8: ¡Vamos a dar un paseo! (Motor de corriente continua)

Control del sentido de giro y velocidad de un motor DC

88 - 95



Capítulo 9: Penalti... ¡Gol! (Servo motor)

Control de la posición angular de un servomotor

96 - 104



Capítulo 10: Mastermind, ¿puedes encontrar el código secreto? (bits RGB)

modelo de color RGB

105 - 113



Capítulo adicional: Juego Simon con LED

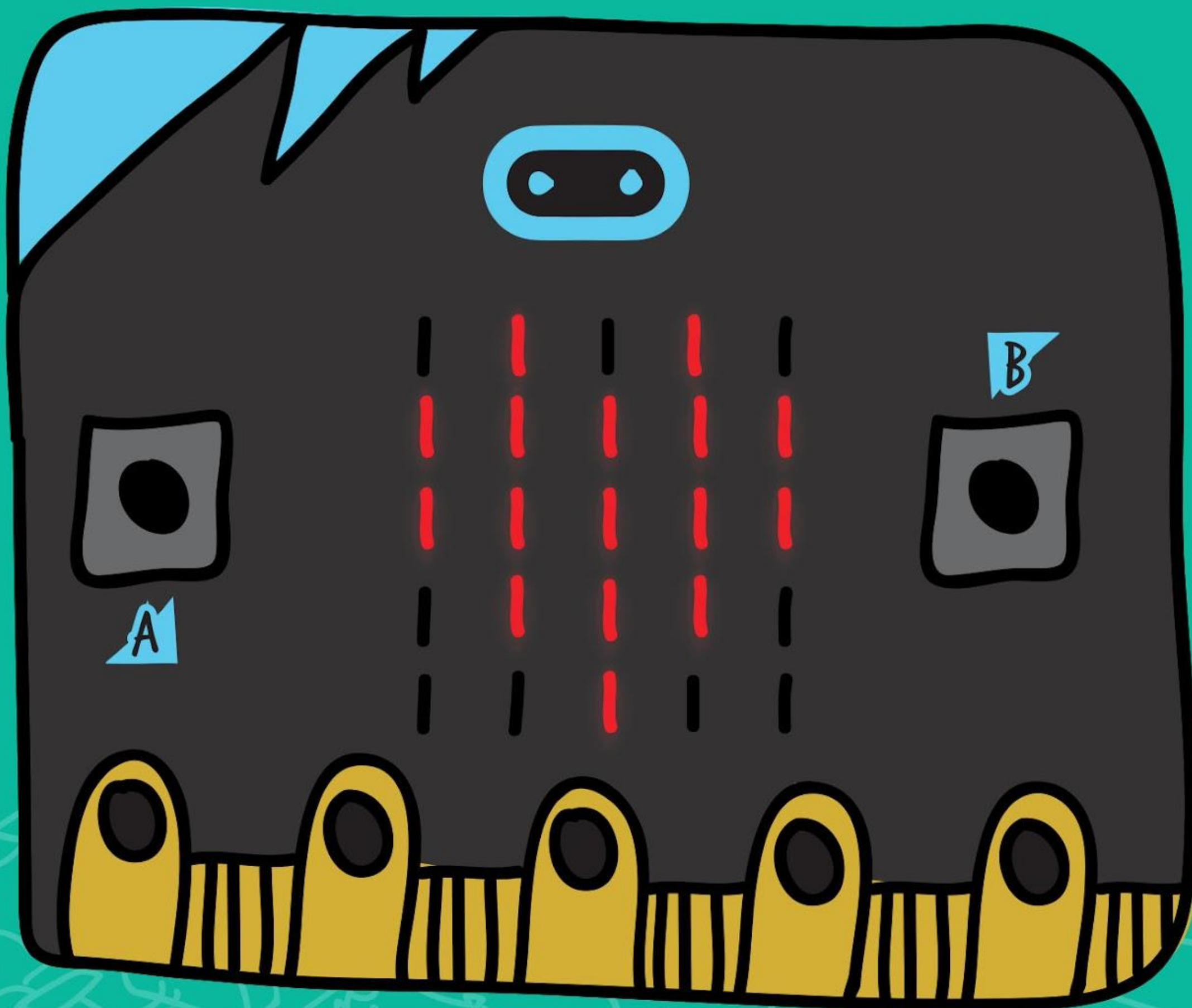
Comunicación por radio

114 - 124

CAPÍTULO 1

> ¡Hola Mundo!_

Matriz LED en micro: bit

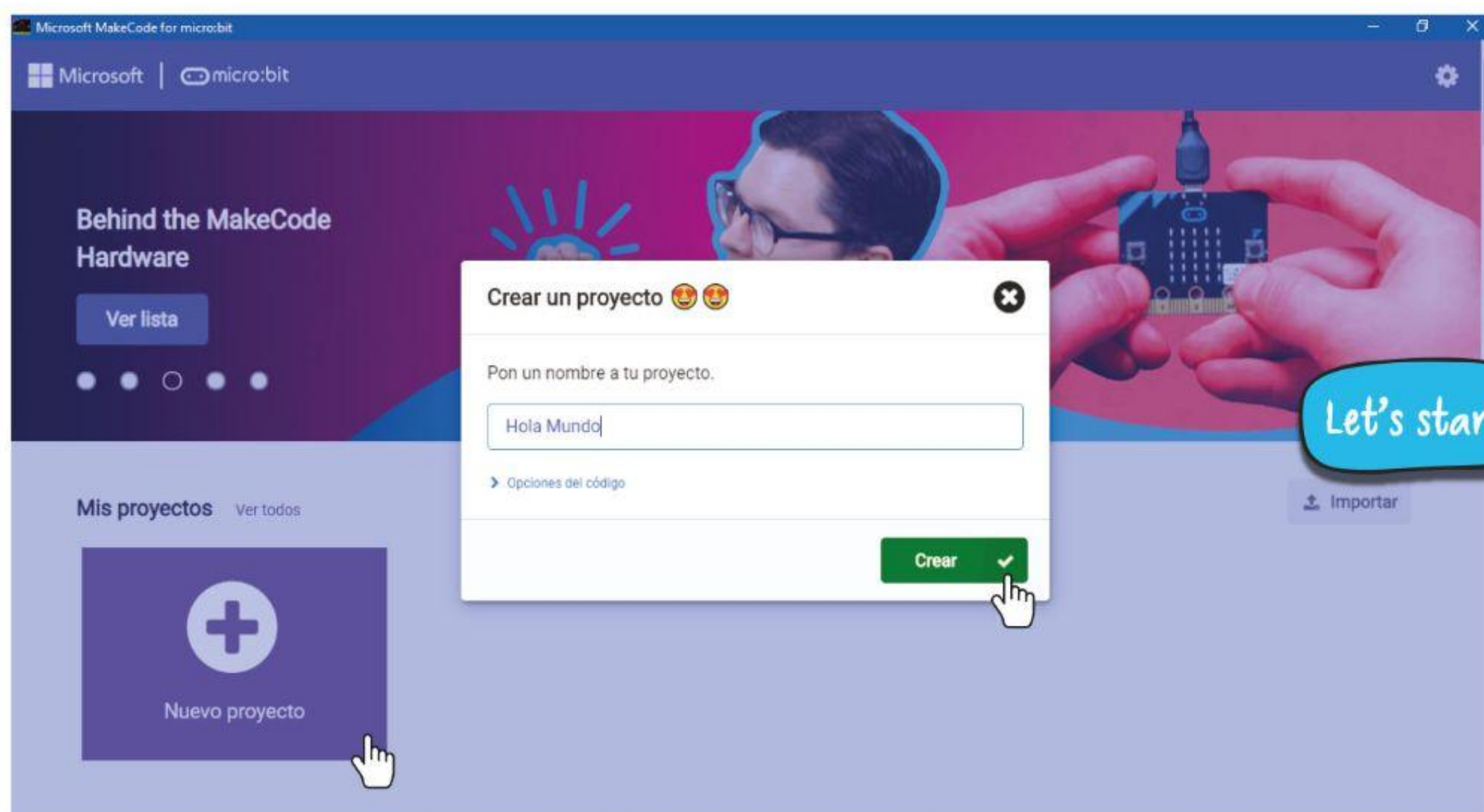


Si Escanéame!

CAPÍTULO 1: ¡Hola, mundo!!

PROGRAMAMOS!

Paso 1 Abre tu navegador y ve a <https://makecode.microbit.org/>
Haz clic en '**Nuevo proyecto**'. Escribe el nombre de tu proyecto y luego haz clic en '**Crear**'.



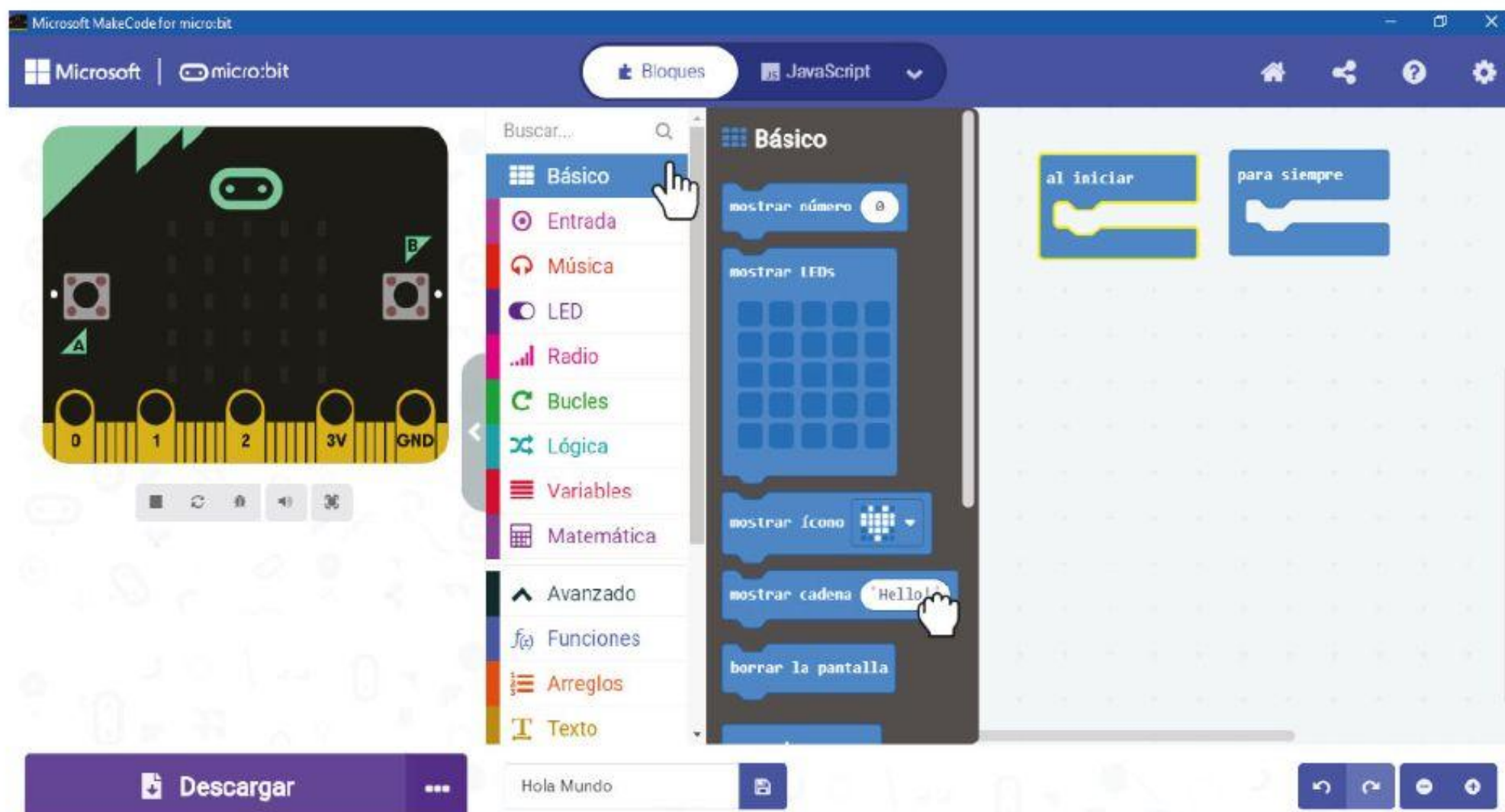
Verás esta página del editor Microsoft **MakeCode** que te permite programar fácilmente tu EDU: BIT utilizando el modo de arrastrar y soltar..



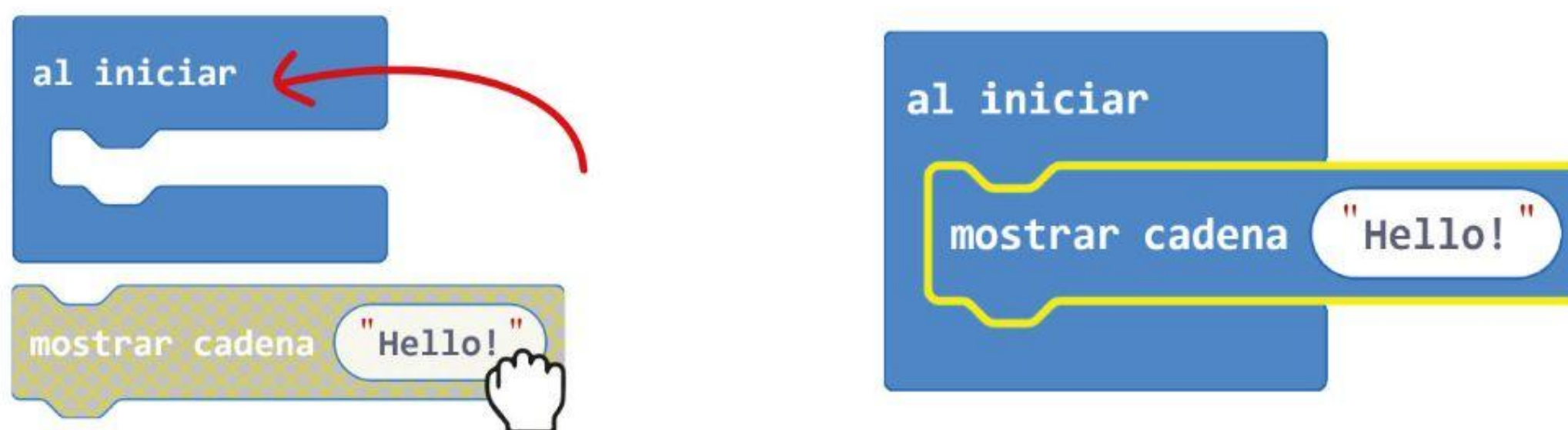
- 1) Publica y comparte tu proyecto
- 2) Elige si programar con Blocks, con Javascript o Python
- 3) Abre el menú de ayuda
- 4) Cambia la configuración, agrega extensiones y empareja un dispositivo
- 5) SIMULADOR: muestra cómo se verá su programa como si se estuviera ejecutando en un micro: bit
- 6) CAJA DE HERRAMIENTAS / LISTA DE CATEGORÍAS: haz clic para ver los bloques de código disponibles en cada categoría. Todos los bloques de la misma categoría tienen el mismo color.
- 7) ESPACIO DE TRABAJO: los programas se construyen encajando los bloques en esta área
- 8) Descarga tu código en el micro:bit
- 9) Dale un nombre y guarda el proyecto actual en tu computadora
- 10) Crear un repositorio de GitHub
- 11) Deshacer rehacer
- 12) Acercar / alejar

CAPÍTULO 1: ¡Hola, mundo!

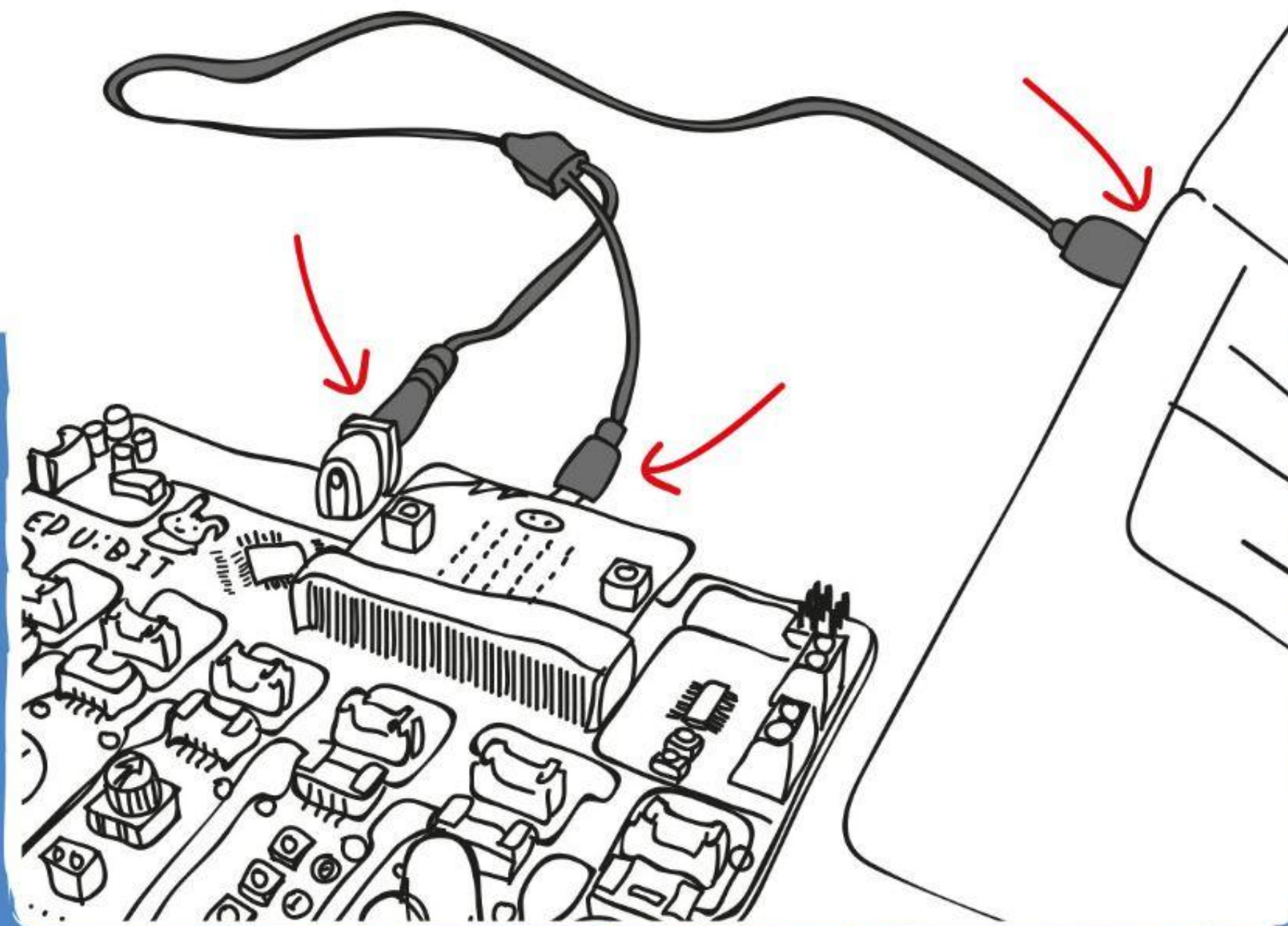
Paso 2 Haz clic en **[Básicos]** y luego selecciona el bloque **[mostrar cadena]**.



Paso 3 Haz clic e inserta el bloque **[mostrar cadena]** en el contenedor **[al inicio]**.

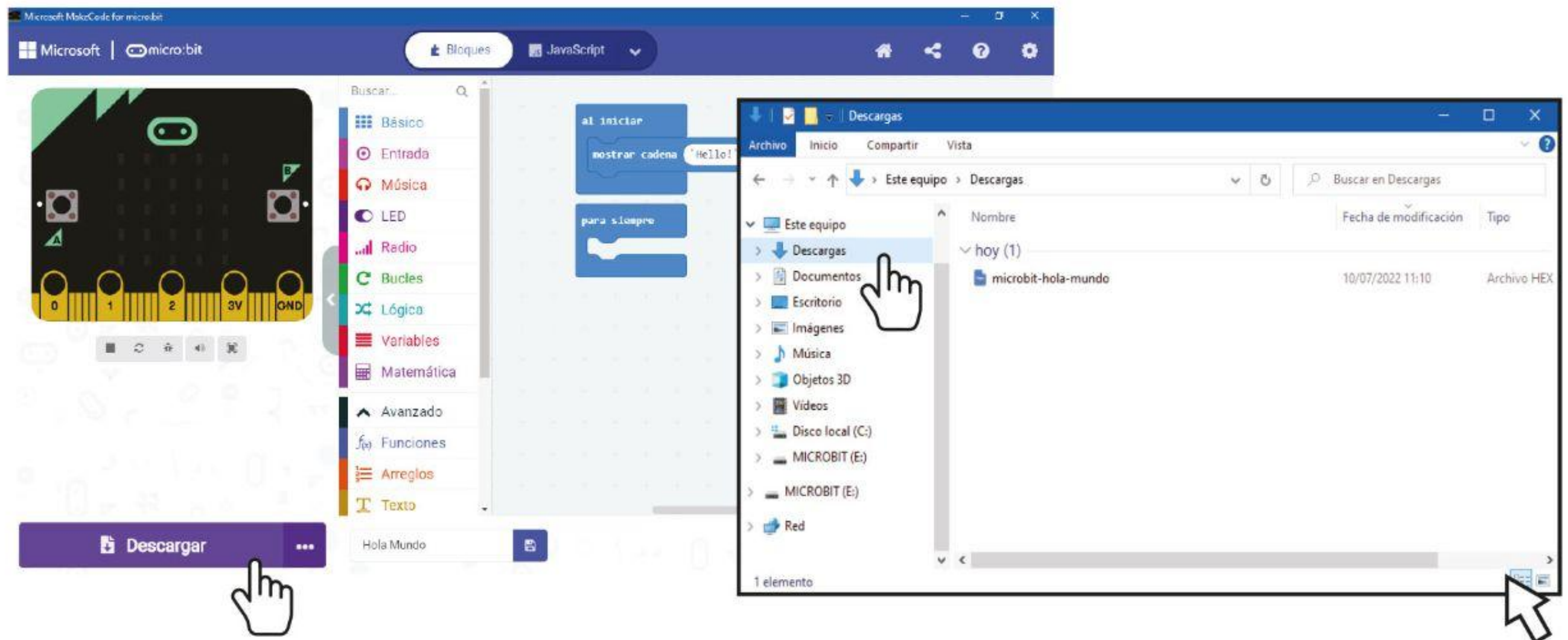


Paso 4 Conecta el cable USB a tu computadora y a la placa EDU:BIT como se muestra en la figura. Recuerda encender EDU: BIT cambiando el interruptor a ON.





Paso 5 Haz clic en el botón **[Descargar]**. En la ventana emergente que aparece, elige descargar tu proyecto a la unidad MICROBIT. Cierra la ventana cuando aparezca "Descarga completada".

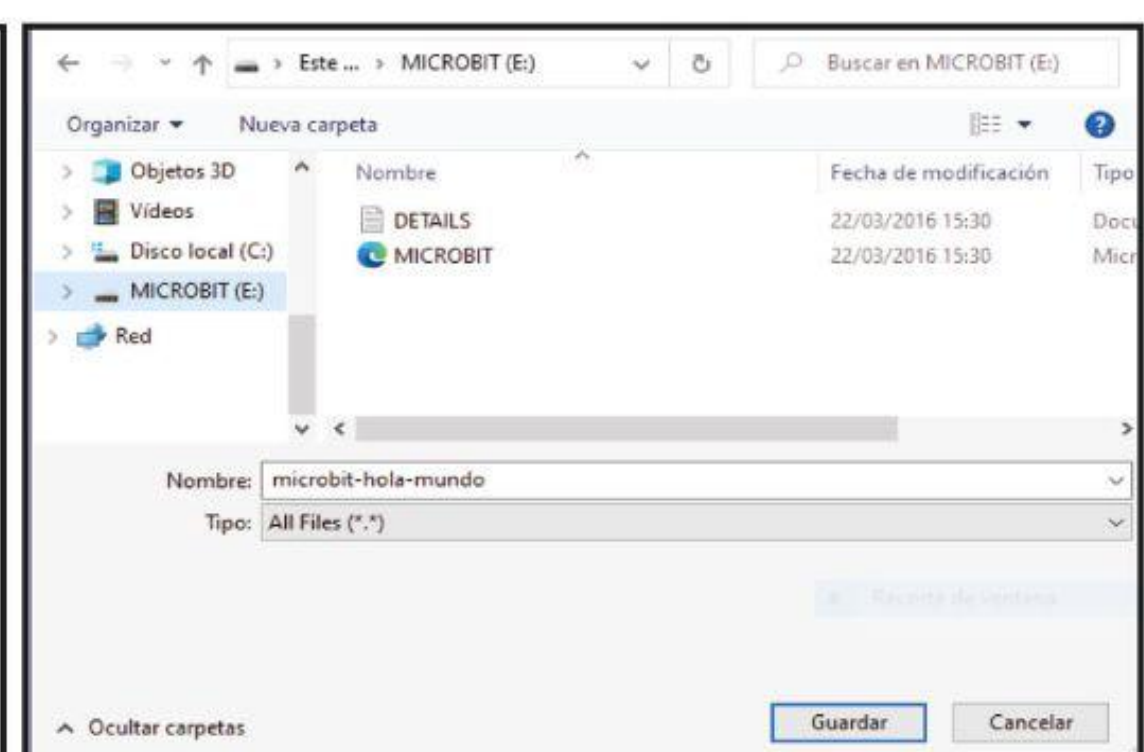
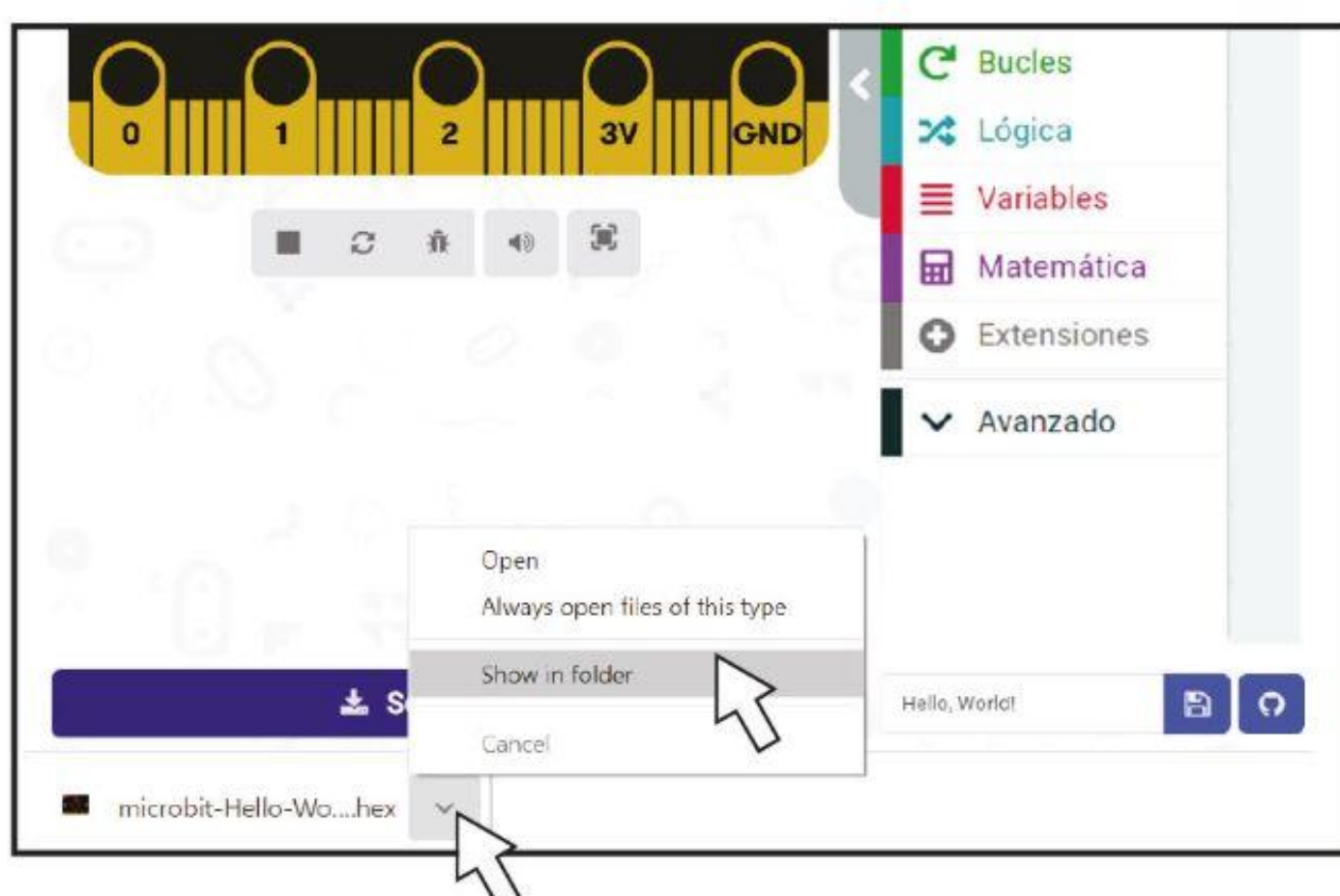


Este proceso de transferencia del código a la tarjeta se denomina Flasheo. El LED naranja detrás del micro: bit parpadea durante la transferencia y cuando se completa, el código se ejecutará automáticamente.



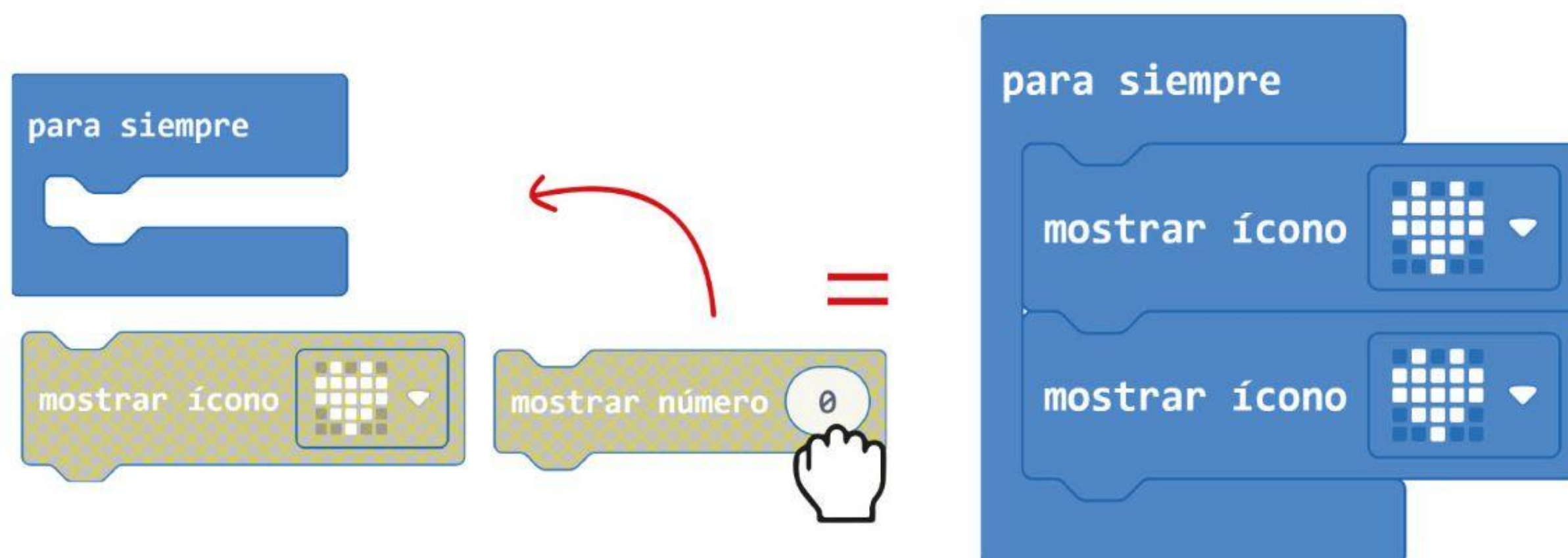
NOTA!

Si la ventana emergente no aparece, significa que el archivo se descargó automáticamente en la ubicación donde su navegador guarda todas las descargas. Haz clic derecho en el archivo .hex descargado que aparecerá en la parte inferior de la ventana y selecciona 'Mostrar en carpeta'. Haz clic y arrastra el archivo "microbit-xxxx.hex" descargado a la unidad MICROBIT, como si estuvieras copiando el archivo en una memoria USB

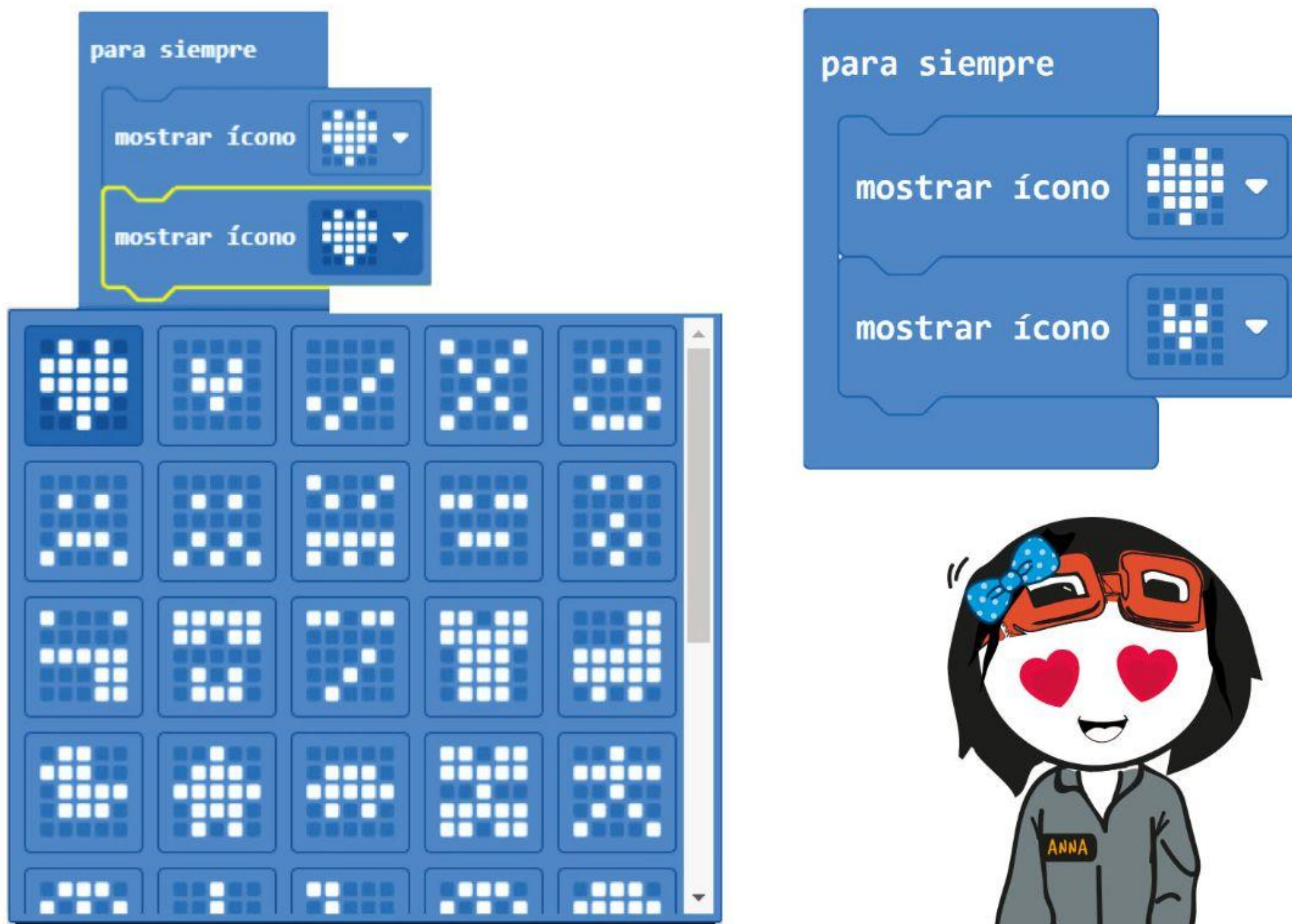


CAPITULO 1 : ¡Hola, Mundo!

Paso 6 Haz clic en **[Básico]** y luego haz clic en el bloque **[mostrar ícono]**. Repite para agregar otro bloque **[mostrar ícono]**. Haz clic y coloca los bloques **[mostrar ícono]** en el contenedor **[para siempre]**.



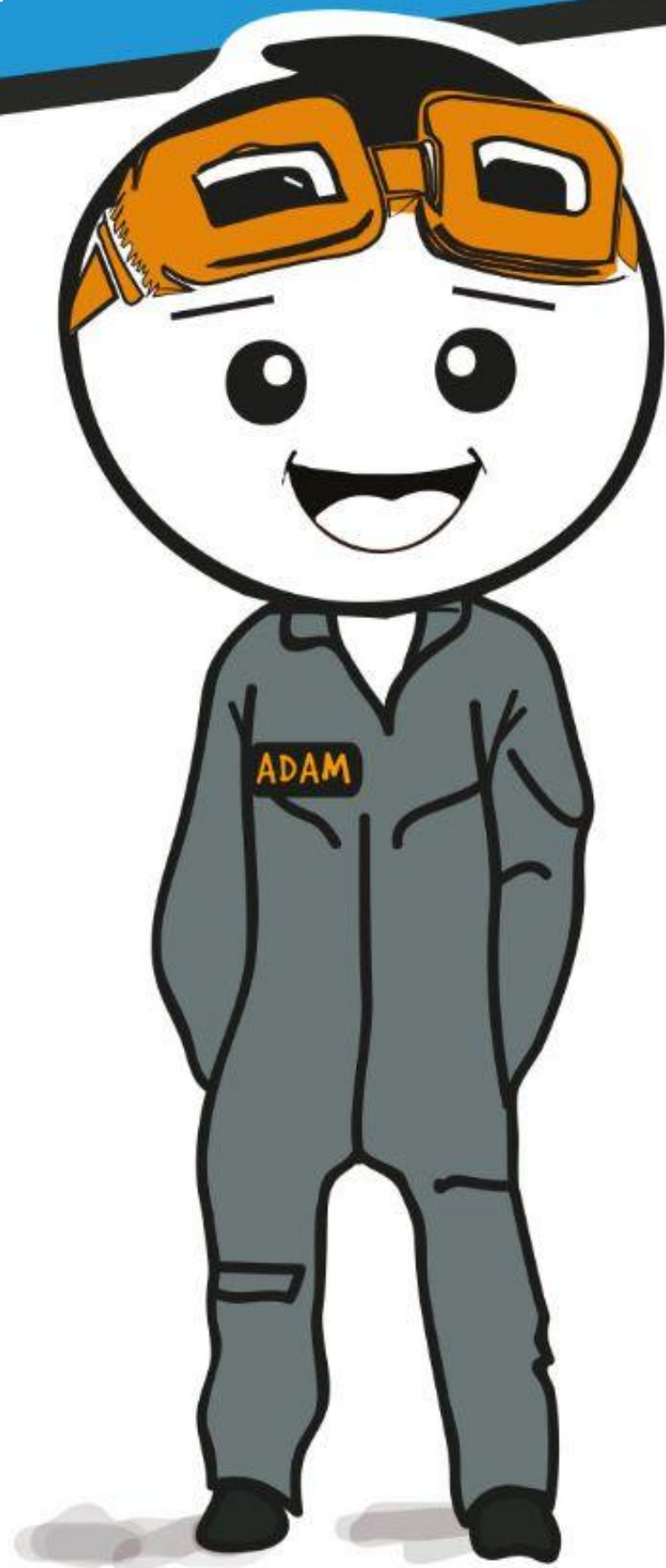
Paso 7 Haz clic con el botón izquierdo en el segundo icono de bloque **[mostrar ícono]** y selecciona la imagen del corazón pequeño en la ventana emergente. Transfiere el código a tu tarjeta EDU: BIT.



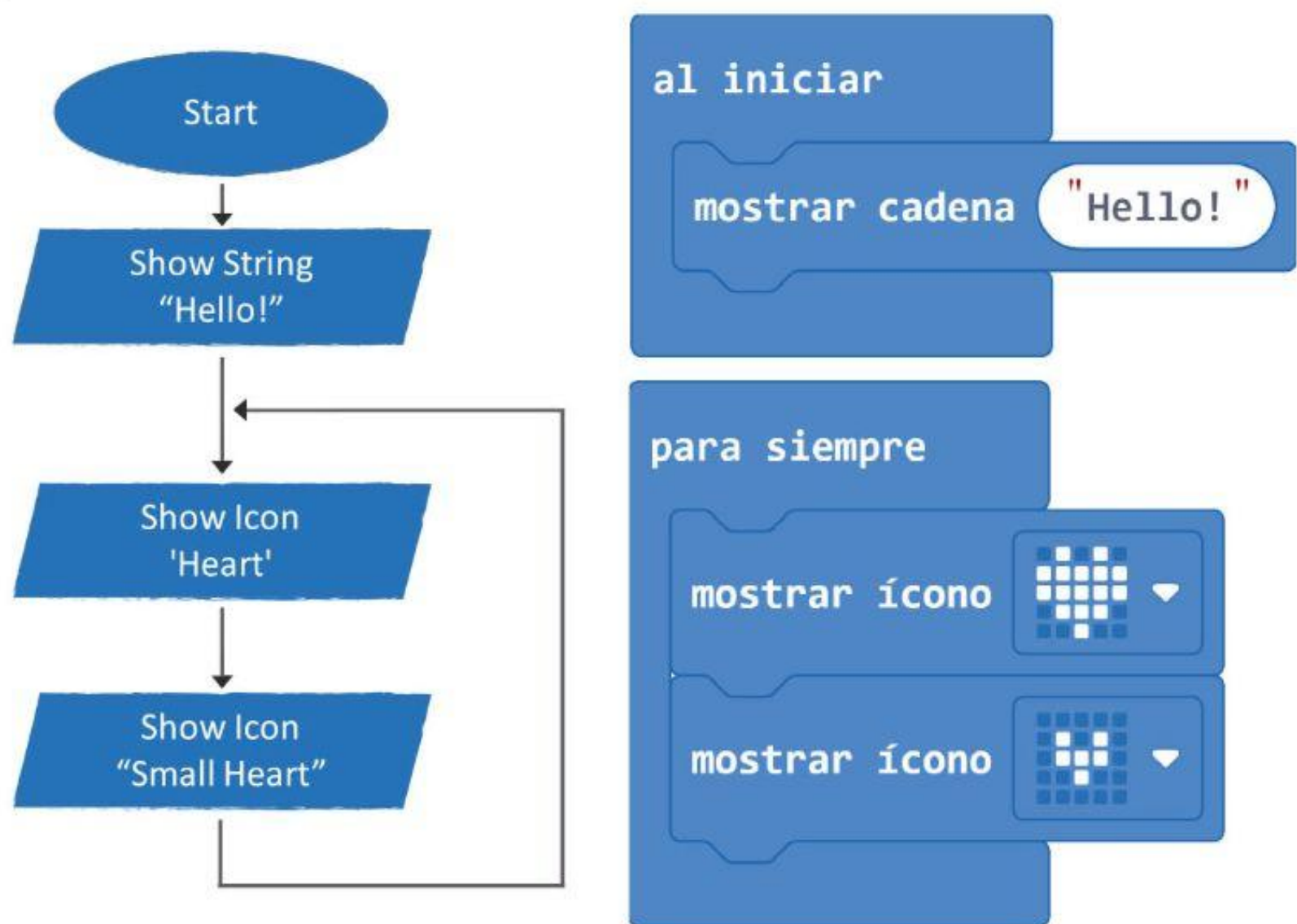
Es amor a primera vista. ¿Ves la animación de un corazón que late?



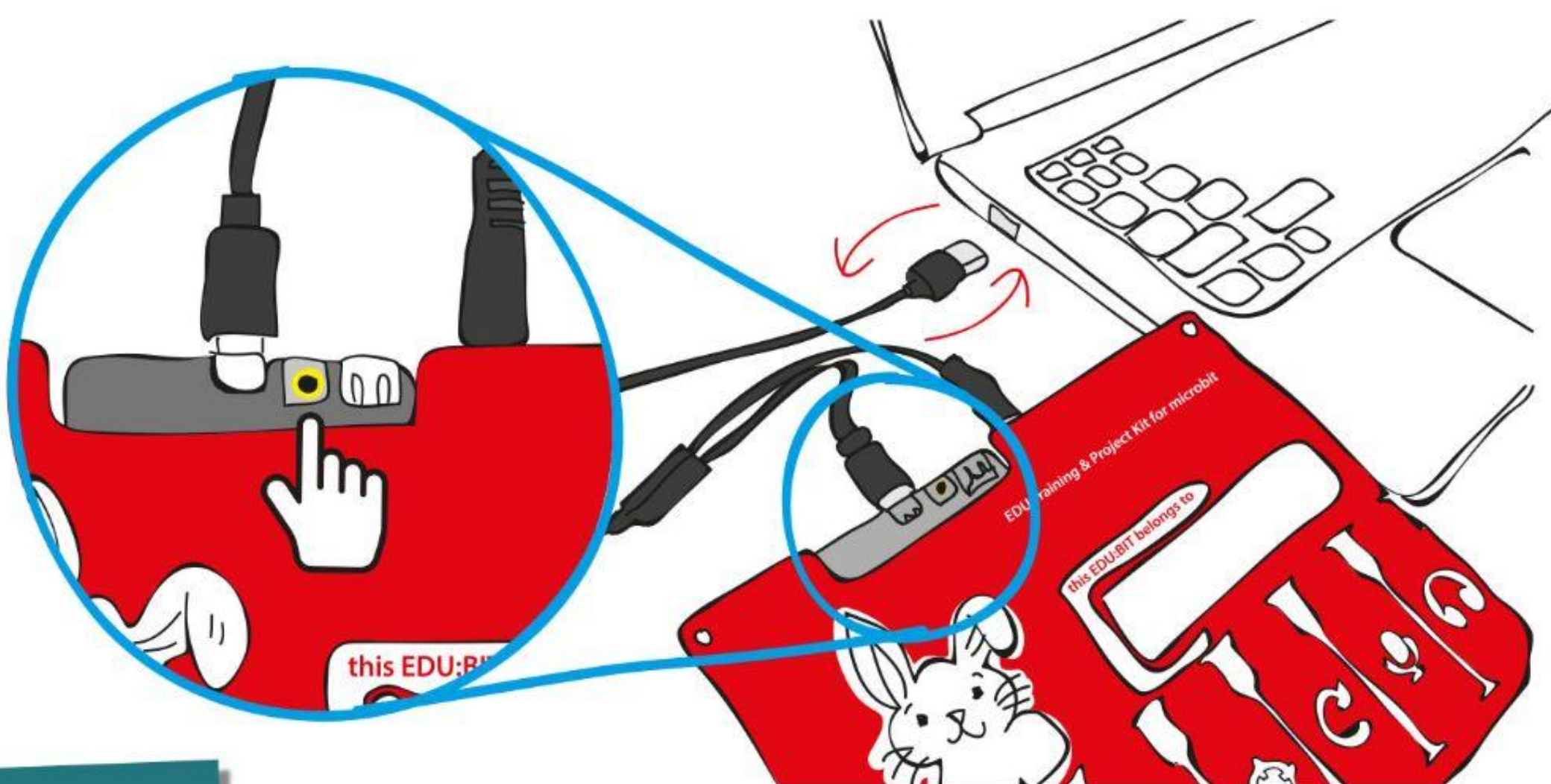
MODIFICAR EL CÓDIGO



¿Te diste cuenta de que el texto "¡Hola!" se desplaza en la pantalla solo una vez, pero el icono del corazón continúa en bucle? ¿Por qué?



El bloque **[al iniciar]** ejecuta el código al inicio (una vez). El bloque **[por siempre]** ejecuta el código nuevamente cuando llega al final (por siempre).

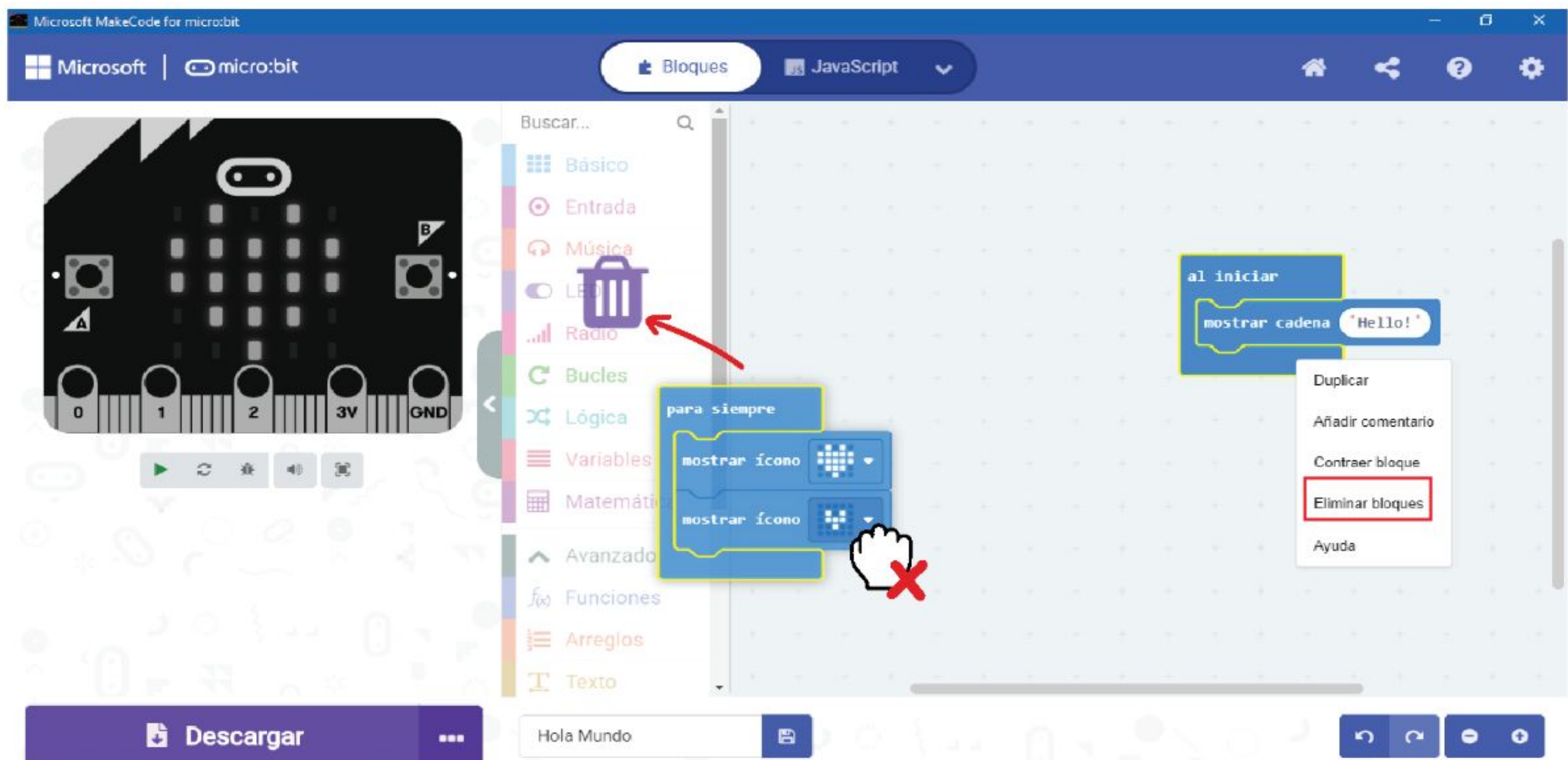


NOTA!

Si deseas volver a iniciar el programa desde el principio, simplemente presiona el botón RESET o desenchufa el cable USB y luego vuelve a enchufarlo para reiniciar la placa

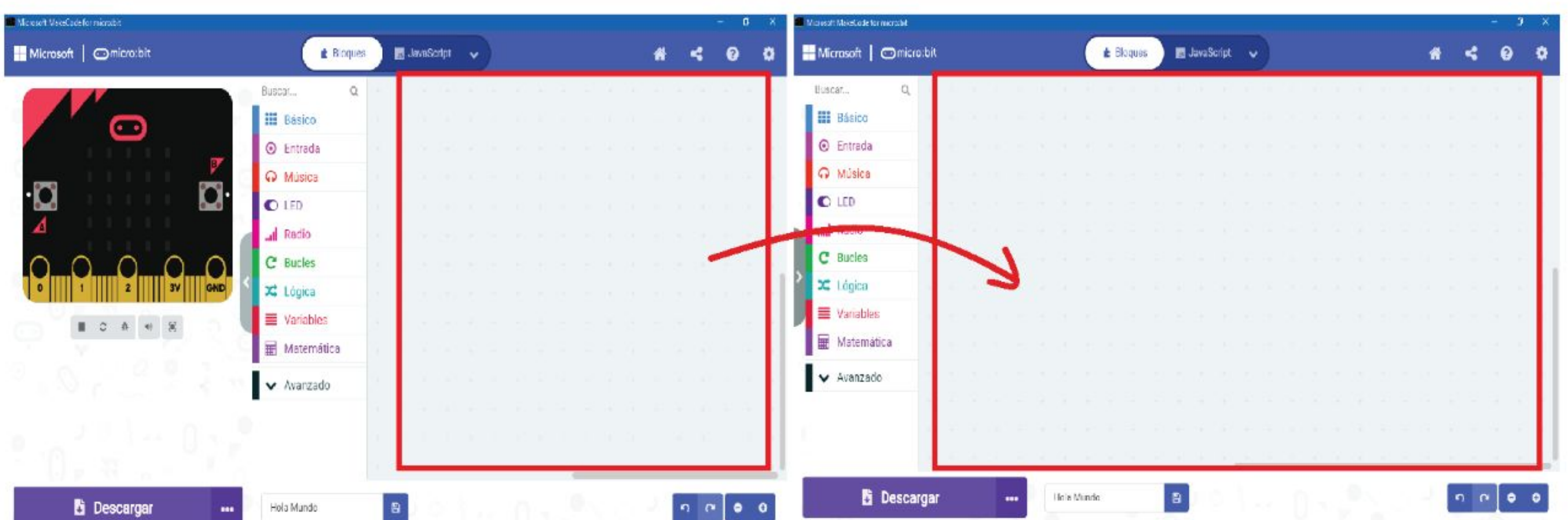
¡Consejo rápido #1!

Puedes eliminar bloques haciendo clic y arrastrando bloques no deseados en el área de la caja de herramientas. Suelta cuando aparezca el icono de la papelera para eliminar los bloques. Alternativamente, puedes hacer clic con el botón derecho y seleccionar "Eliminar bloque".



Consejo rápido # 2!

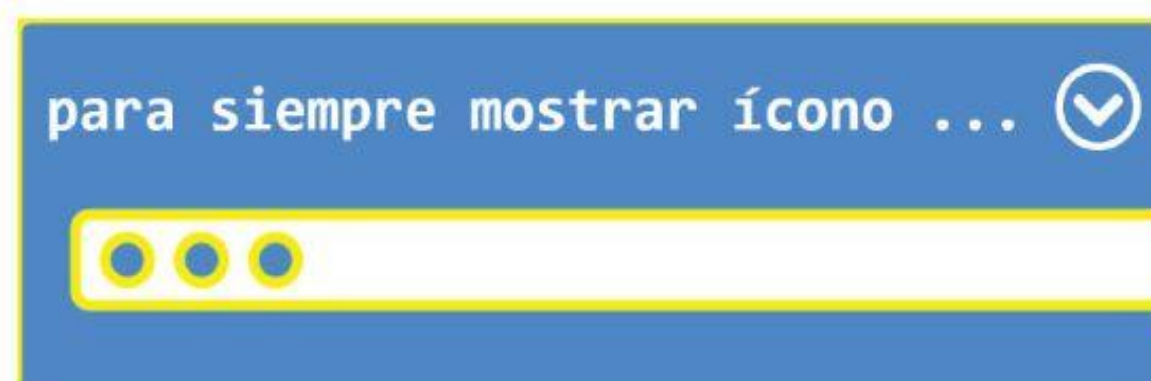
Si no estás utilizando la ventana del simulador, puedes hacer clic en la pestaña para ocultarla y tener más espacio para tus bloques de código





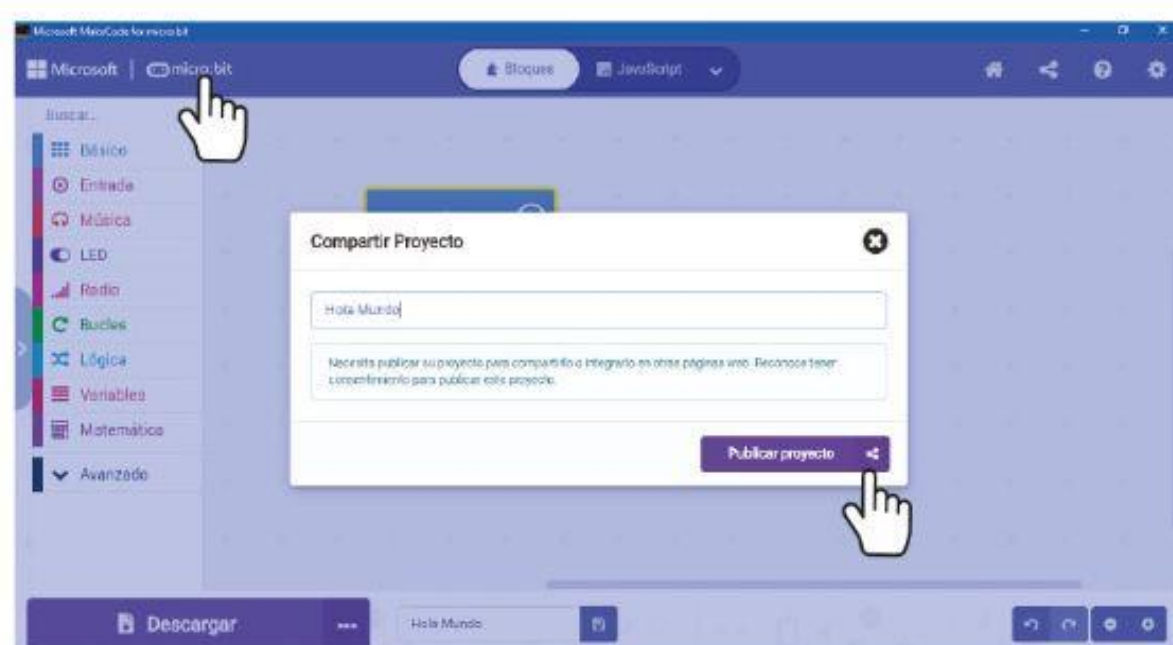
¡Consejo rápido #3!

Puedes colapsar un conjunto de bloques haciendo clic derecho en el contenedor y seleccionando 'Colapsar bloque'. Para expandir un conjunto comprimido de bloques, simplemente haz clic en el icono con la flecha hacia abajo..



¡Consejo rápido #4!

Puedes compartir tu código con tu maestro o amigos publicando tu proyecto y luego enviando la URL de tu proyecto. Para hacer esto, haz clic en **[Compartir]** y luego haz clic en **[Publicar proyecto]** en la ventana emergente. Verás una nueva ventana emergente con la URL de tu proyecto. Haz clic en el botón **[Copiar]** para copiar la URL del proyecto.



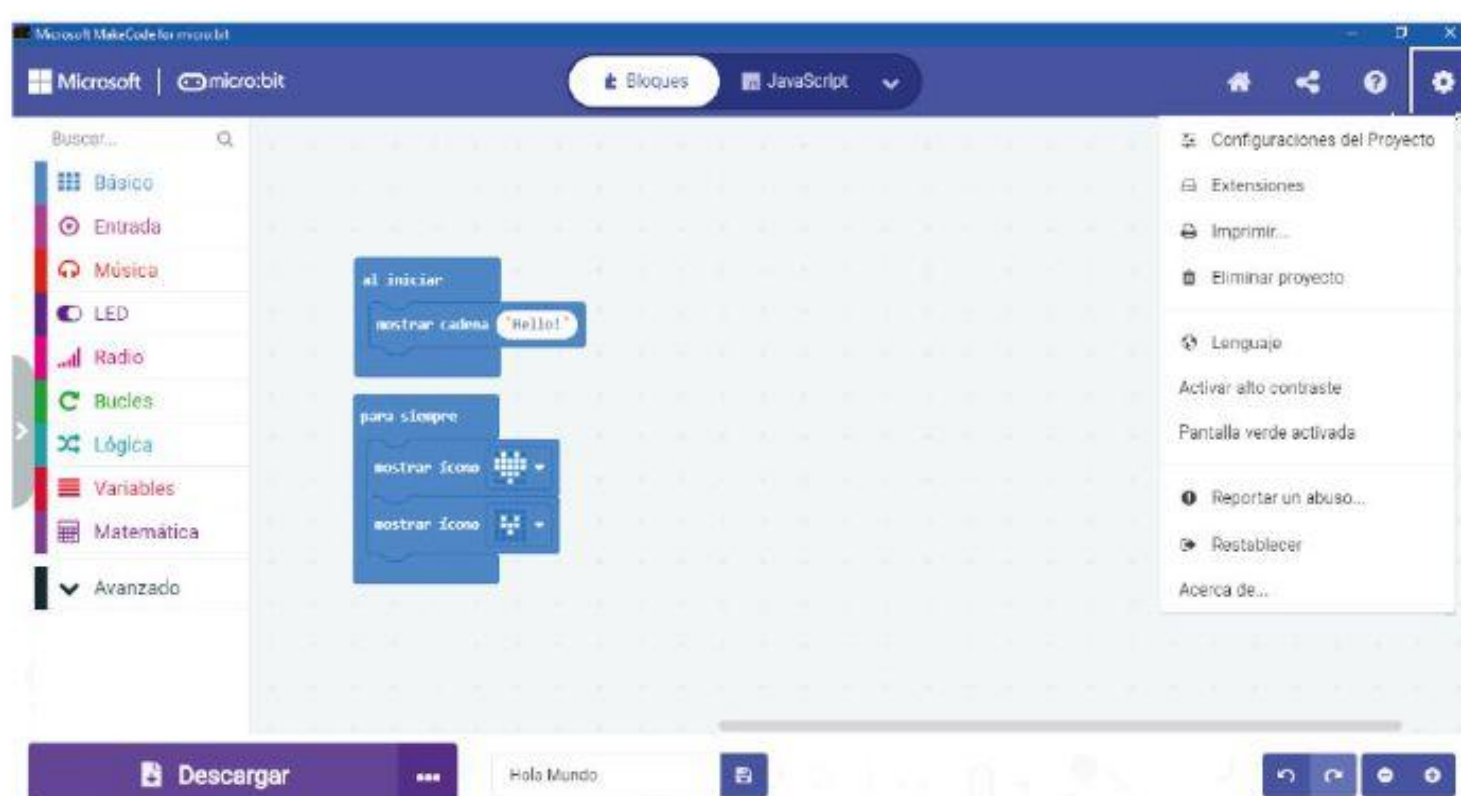
¡Consejo rápido #5!

El profesor o los amigos verán esta página cuando abran la URL de su proyecto en su navegador. Podrán ver tu código y también editarlo haciendo clic en **[Editar Código]**.



¡Consejo rápido #6!

¿Sabes que puedes emparejar el dispositivo para descargar con un solo clic? Para hacer esto, haz clic en el ícono de ajustes y luego selecciona 'Emparejar dispositivo'.



NOTA!

Necesitas el firmware más reciente en tu dispositivo micro: bit y usar el nuevo Edge o Chrome como navegador. Sigue las instrucciones aquí si necesita actualizar el firmware: <https://microbit.org/get-started/user-guide/firmware/>

Asegúrate de que su EDU: BIT esté conectado a tu PC y luego haz clic en el botón **[Emparejar dispositivo]** en la ventana emergente. Luego selecciona BBC micro: bit CMSIS-DAP o DAPLink CMSIS-DAP de la lista y luego haz clic en **[Conectar]**.



Una vez que hayas emparejado tu dispositivo, puedes mostrar su código directamente en tu EDU: BIT cuando haga clic en el botón **[Descargar]**. ¡Inténtalo!

Si tienes problemas para emparejar tu dispositivo, puedes consultar

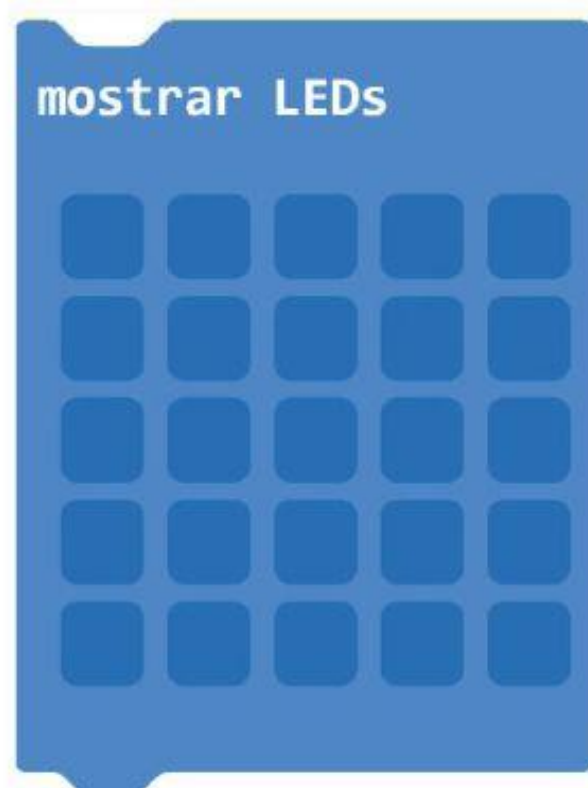
<https://makecode.microbit.org/device/usb/webusb/troubleshoot> para obtener más información.





EXPLORA OTROS BLOQUES

1 Usa el bloque **[mostrar LED]** para dibujar tus íconos y el bloque **[mostrar número]** para mostrar números.



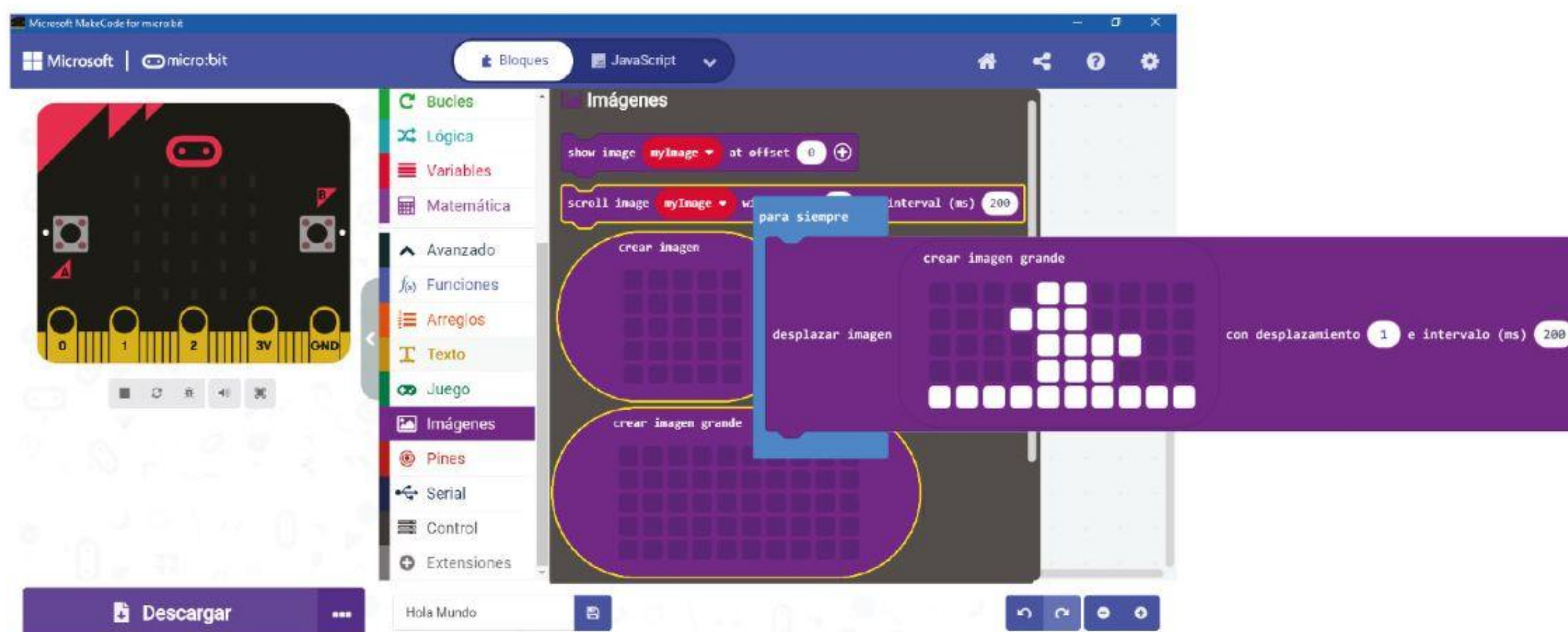
¿Sabías que?

1000 milisegundos (ms) = 1 segundo

#2 Agrega un bloque **[pausa]** para ralentizar el programa. Esta función detiene el programa durante el número establecido de milisegundos (ms).



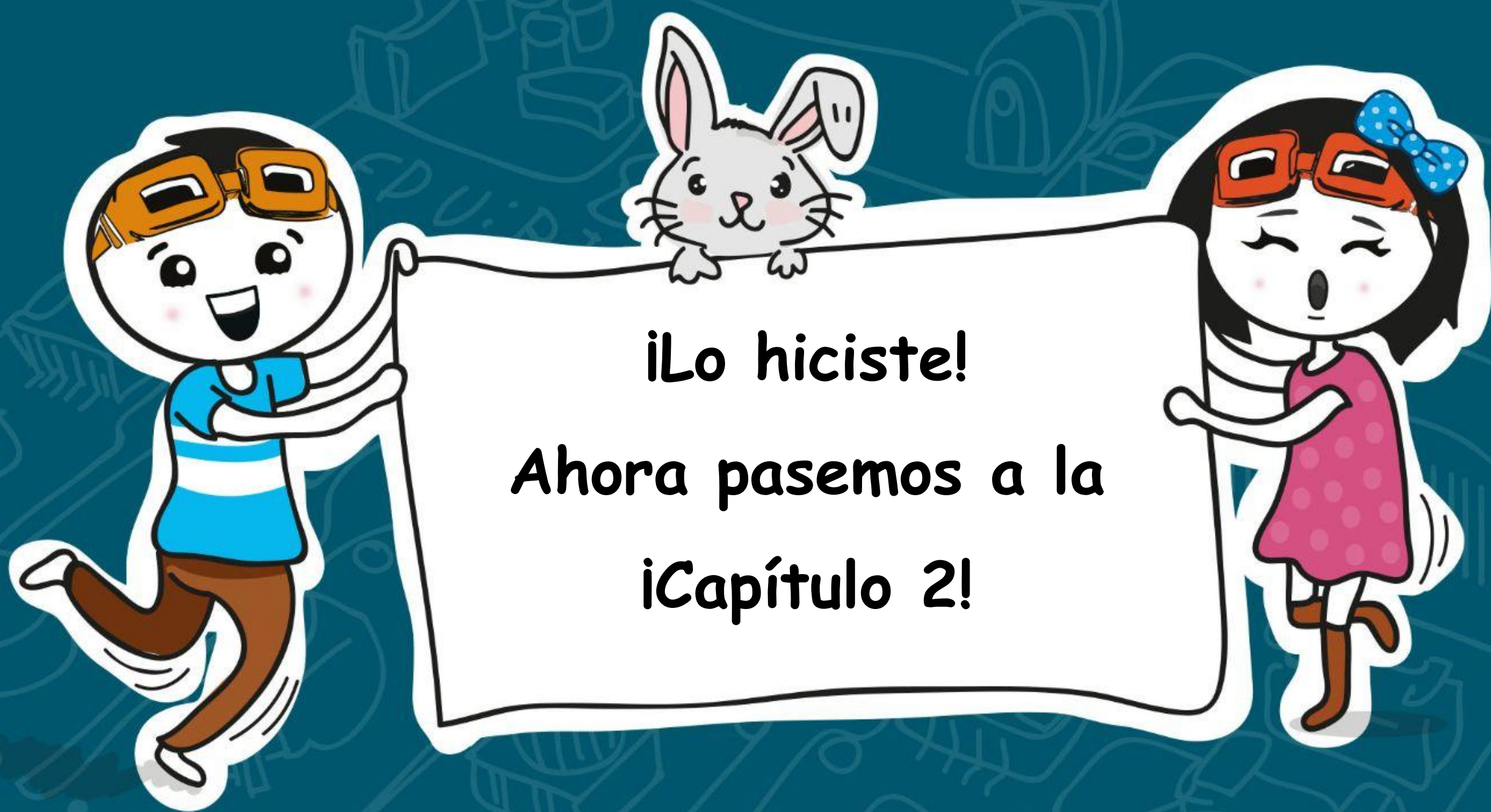
3 Para desplazar una imagen en la pantalla de matriz de LED, puedes utilizar el bloque **[desplazar imagen _ con _ desplazamiento _ a intervalos de _ (ms)]**, junto con el bloque **[crear imagen]** o el bloque **[crear imagen grande]**, en la sección **[Imágenes]** (en la categoría **Avanzado**).



Cuando ejecutes el programa, verás pequeños patitos moviéndose en la pantalla de matriz de LED, uno tras otro.

DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para que funcione como un tablón de anuncios digital.	
en el inicio	Muestra una animación simple para llamar la atención y luego desplaza el nombre de su clase.
siempre	Desplaza la fecha de hoy y otra información importante para la clase.en el inicio



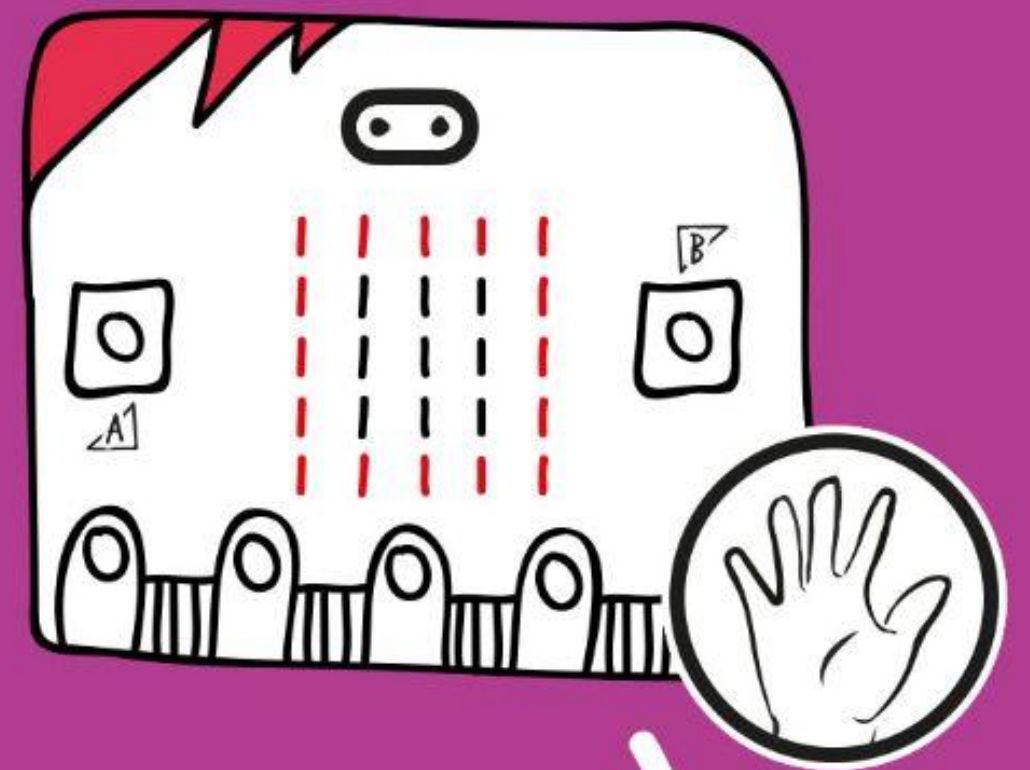
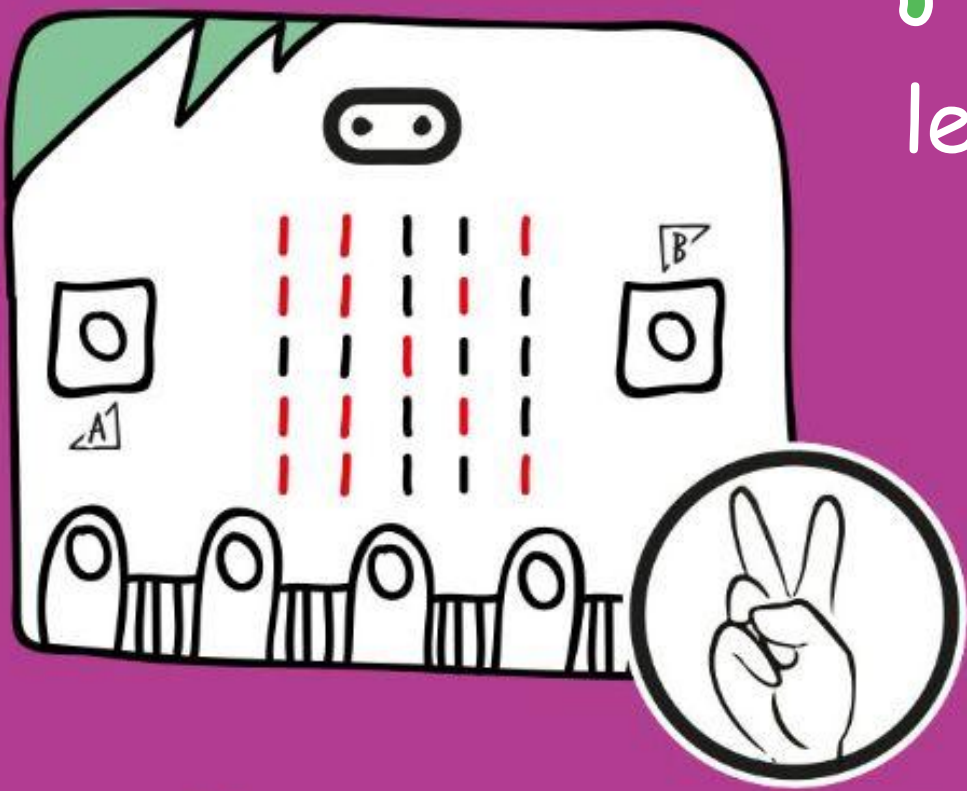
CAPÍTULO 2

Vamos a jugar ¡Piedra Papel tijeras!

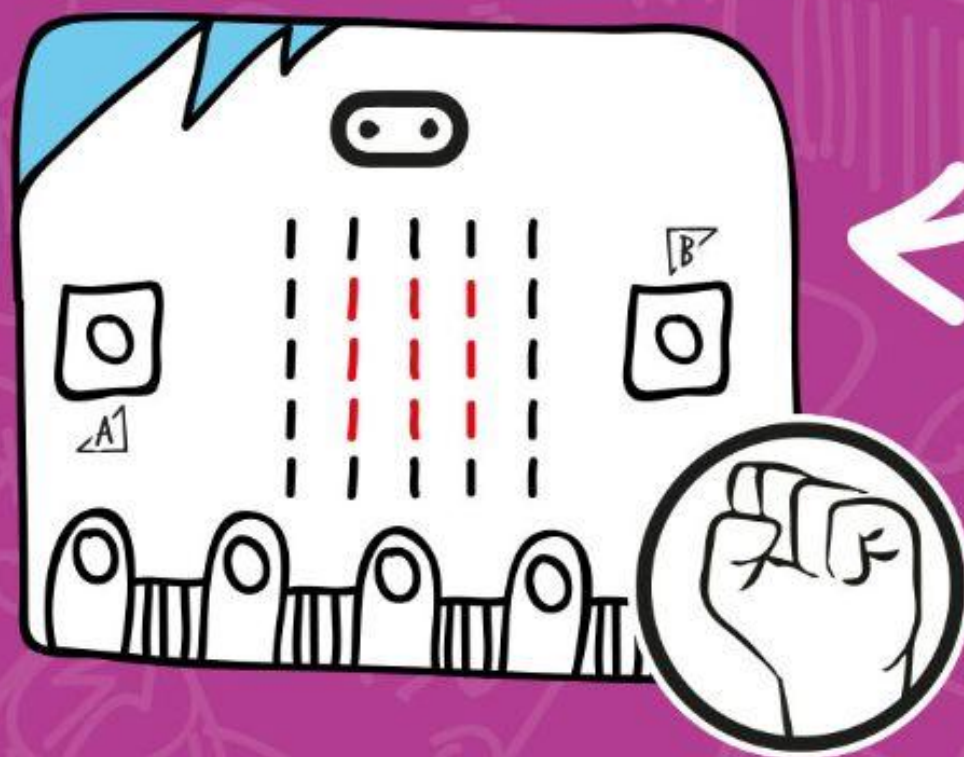
Botones en Micro:bit y Bit Button

TIJERAS

le ganaron la carta



PIEDRA
vence a las tijeras



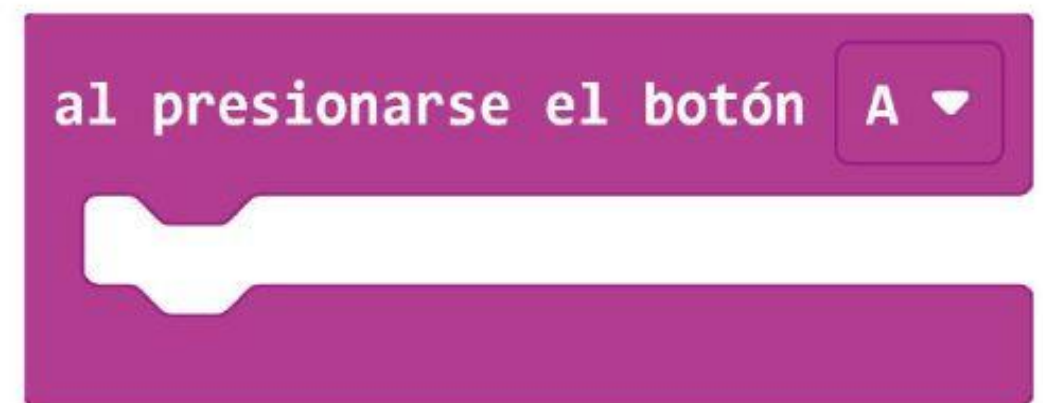
PAPEL
golpea la piedra



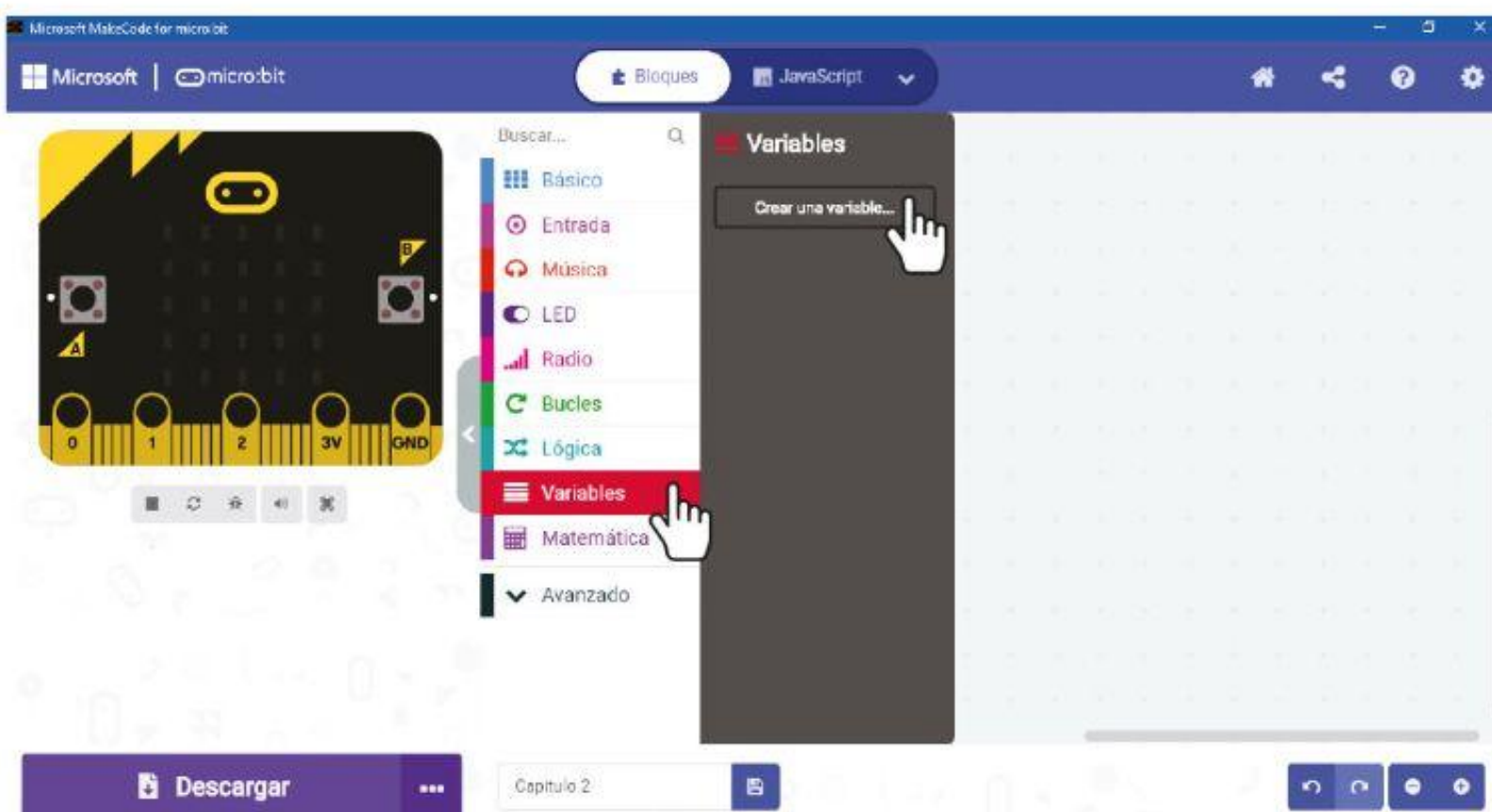
CAPÍTULO 2 : ¡Piedra Papel tijeras!

PROGRAMAMOS!

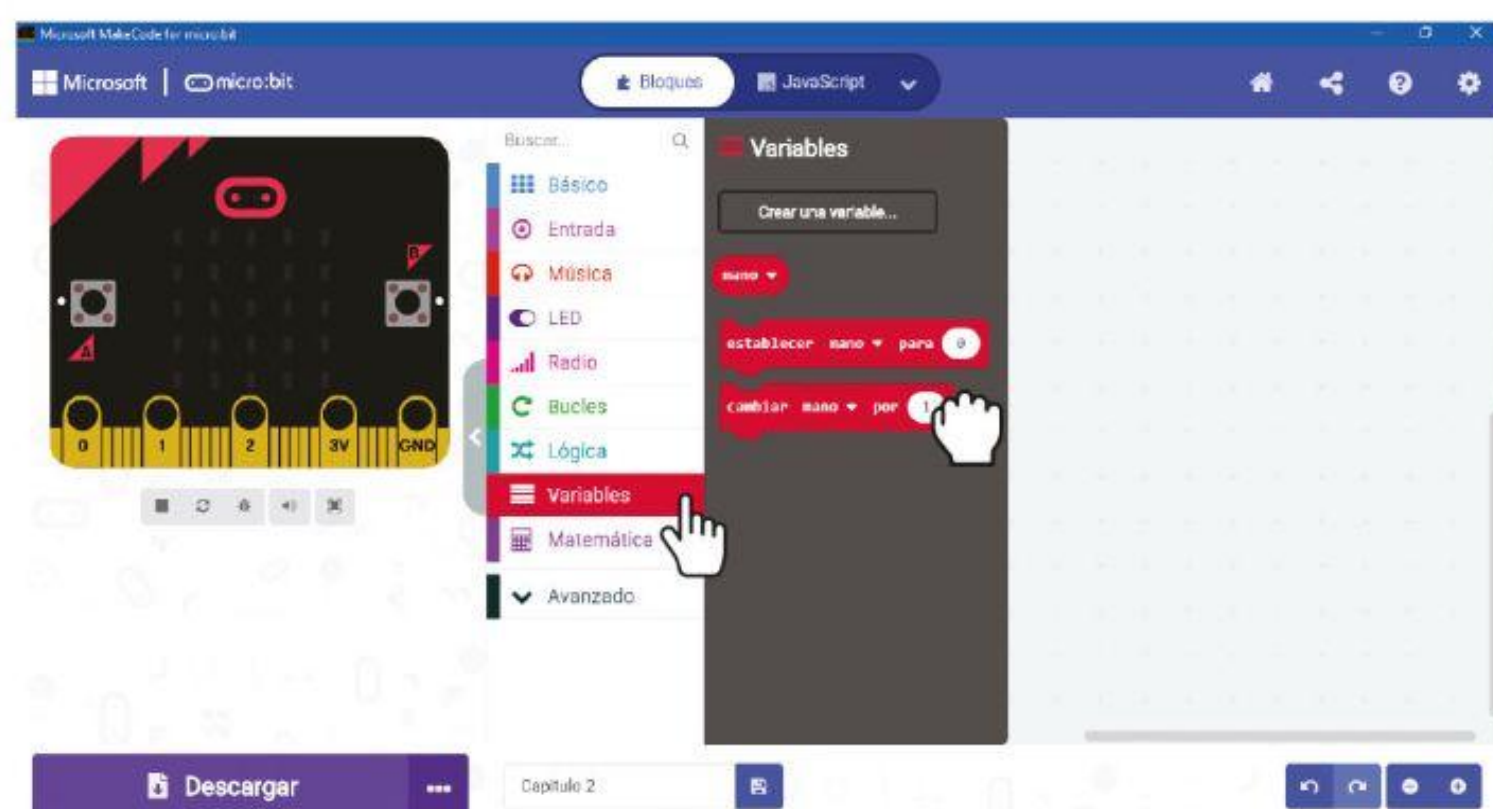
Paso 1 Ve a <https://makecode.microbit.org/> (o simplemente haz clic en el icono **Página Principal** si ya estás en el Editor de MakeCode) crea un nuevo proyecto. Haz clic en la categoría **[Entradas]** y luego selecciona el bloque **[al presionar el botón _]**.



Paso 2 Haz clic en la categoría **[Variables]** y luego selecciona **[Crear una variable]**. Escribe 'mano' en la ventana emergente y luego haz clic en Aceptar.



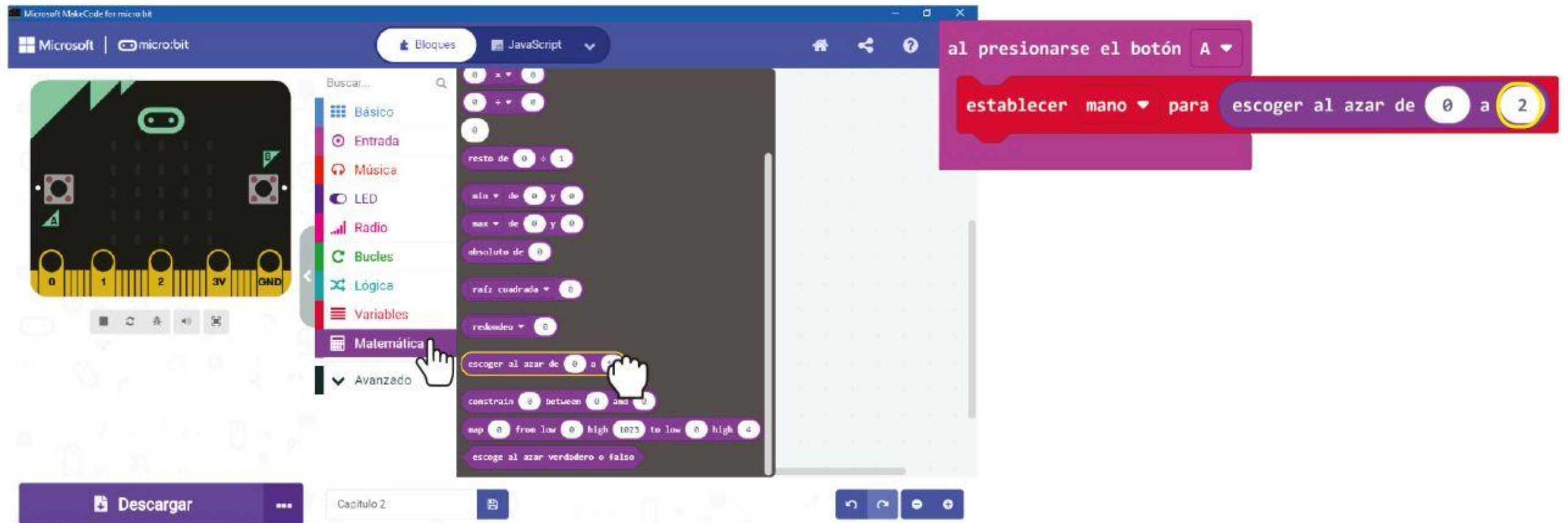
Paso 3 Haz clic en la categoría **[Variables]** y luego selecciona el bloque **[establecer_ en_]**. Ajusta el bloque dentro del bloque **[cuando presionas el botón_]**.



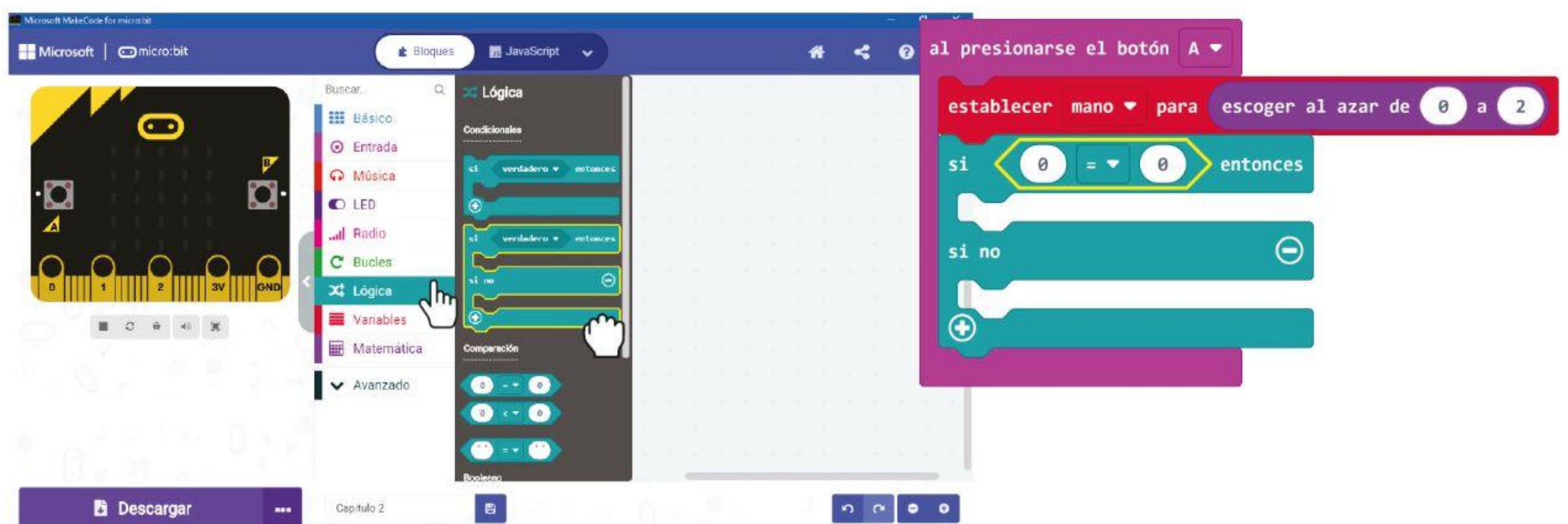
CAPÍTULO 2 : ¡Piedra Papel tijeras!



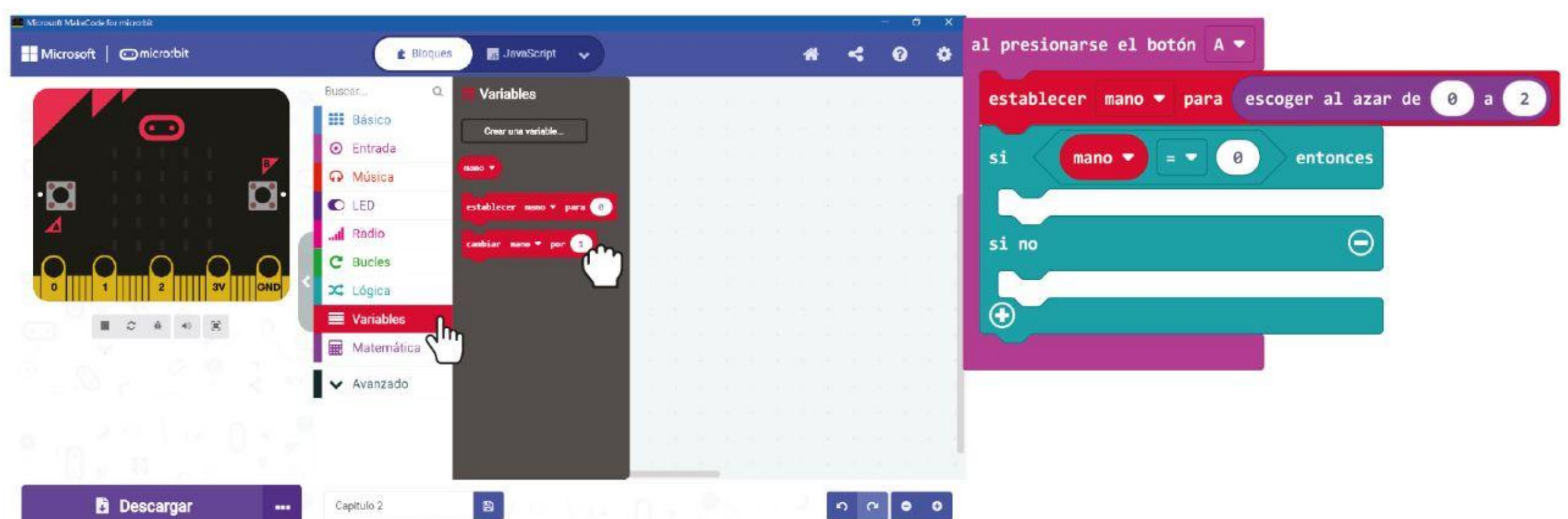
Paso 4 Haz clic en la categoría **[Matemáticas]** y luego selecciona el bloque **[valor aleatorio de _a _]**. Cambia el número 10 al valor 2.



Paso 5 Haz clic en la categoría **[Lógica]** y luego selecciona el bloque **[si_entonces]** y el bloque de comparación **[_ = _]**. Coloca el bloque de comparación en la ranura 'si'.

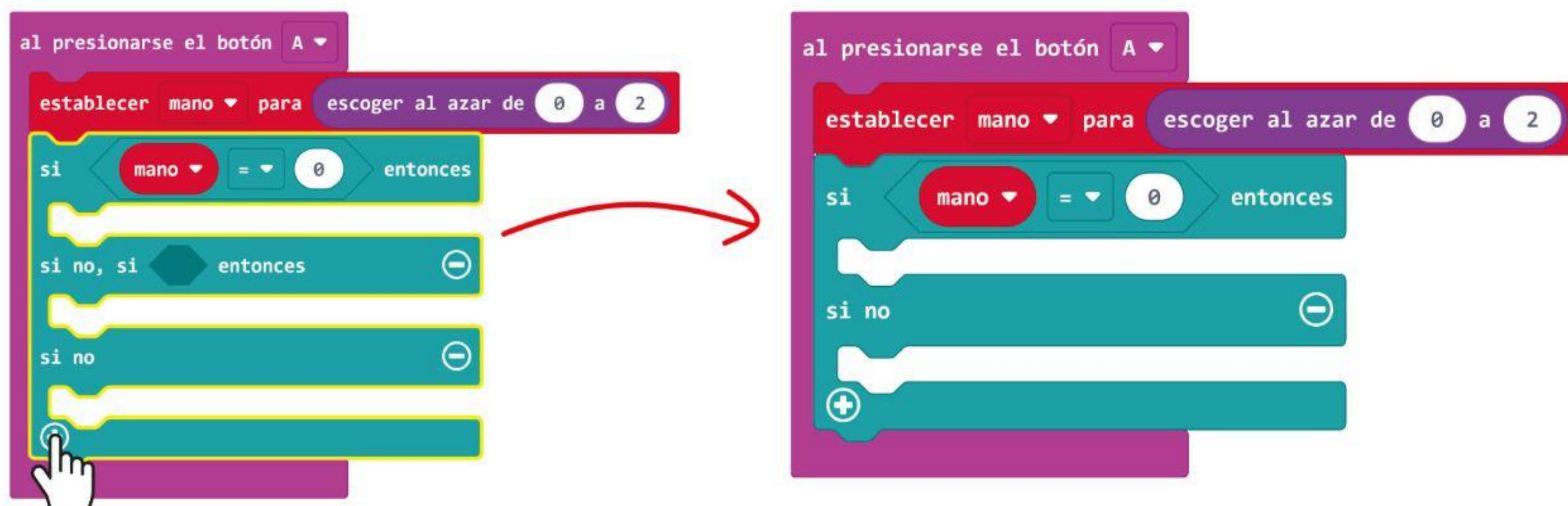


Paso 6 Haz clic en la categoría **[Variables]** y luego selecciona el bloque **[mano]**. Coloca el bloque dentro del bloque de comparación.

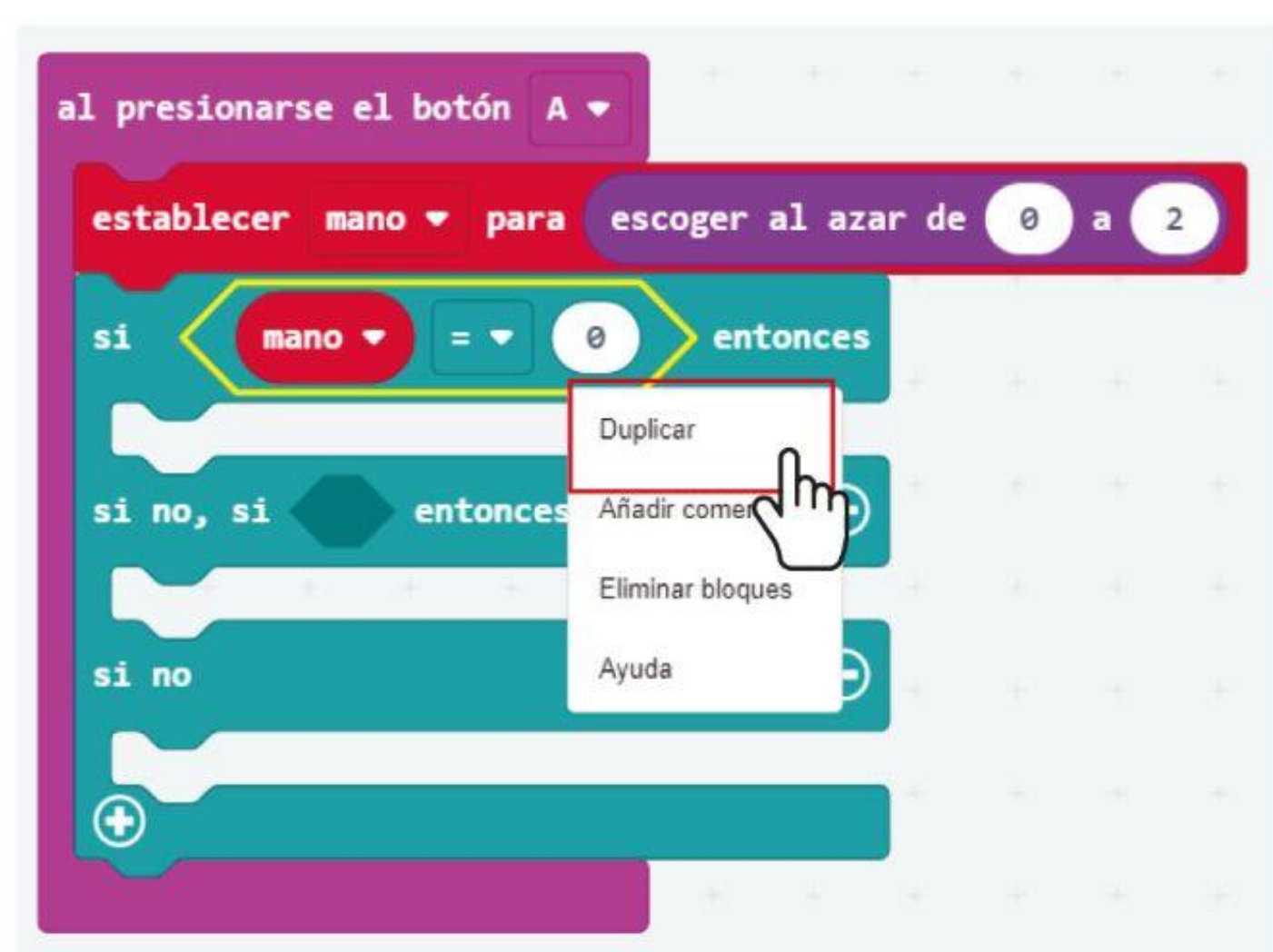


CAPÍTULO 2 : ¡Piedra Papel tijeras!

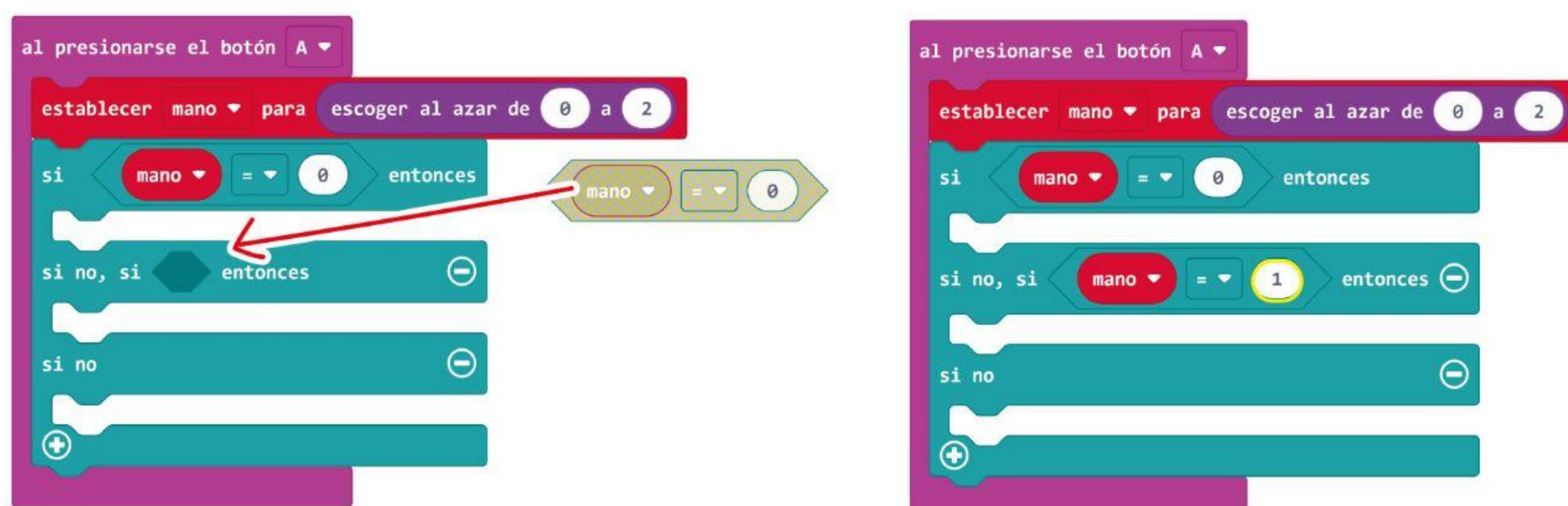
Paso 7 Haz clic en el ícono + para agregar una condición [de lo contrario] al bloque 'si'.



Paso 8 Haz clic derecho en el bloque de comparación y selecciona 'Duplicar'.



Paso 9 Ajusta el bloque duplicado en la ranura 'si no' y cambia el número 0 al valor 1.





Paso 10 Agrega los bloques **[Básico] : [mostrar LED]** en las ranuras 'sí', 'de lo contrario sí' y 'de lo contrario'. Haz clic en los cuadrados de los bloques. **[mostrar LED]** para crear las imágenes como en el ejemplo que se muestra a continuación.

→

Transfiere el código a tu EDU: BIT y podrás jugar a Piedra, Papel, Tijera con tus amigos. Cada vez que presiones el botón A en el micro: bit o el botón amarillo, la matriz de LED mostrará aleatoriamente "piedra", "papel" o "tijeras".

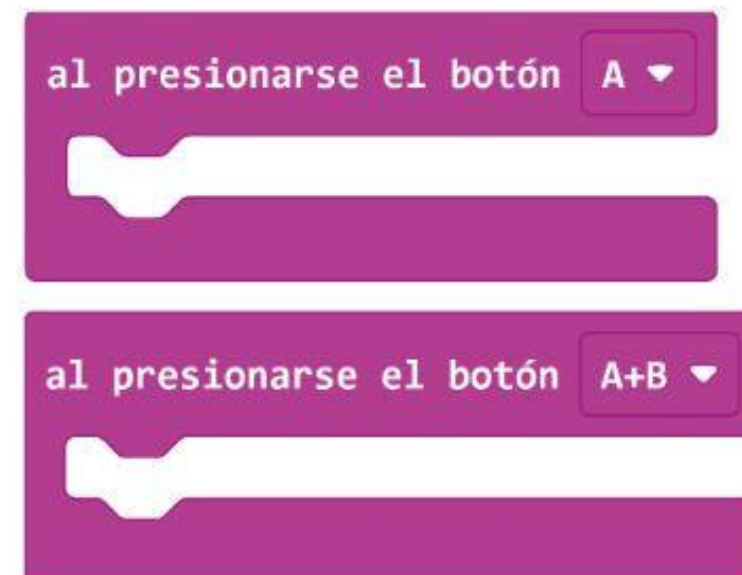
Recordatorio: si deseas conservar el proyecto, recuerda guardarlo en una carpeta de tu computadora haciendo clic en el botón Guardar.





Mejoremos un poco este juego asignando 3 vidas a cada jugador. Para hacer esto, debe crear una nueva variable llamada 'adivina' y agregar los siguientes bloques de código.

Paso 11 Haz clic en la categoría **[Entradas]** y selecciona el bloque **[al presionarse el botón_]**. Duplica el bloque y cambia la configuración a 'Botón B' y 'Botón A + B' respectivamente.



Paso 12 Haz clic en la categoría **[Variable]** y selecciona **[Crear una variable]**. Escribe 'adivina' en la ventana emergente y luego haz clic en OK.



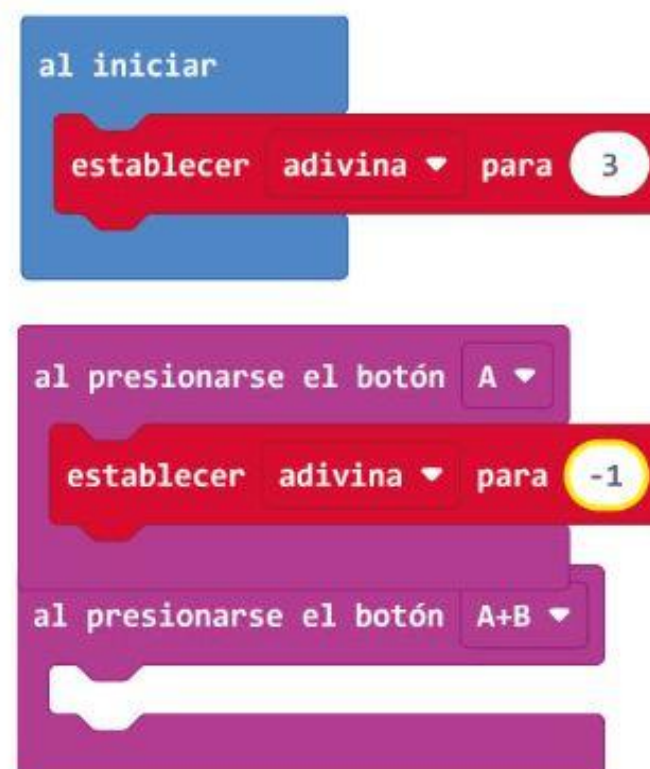
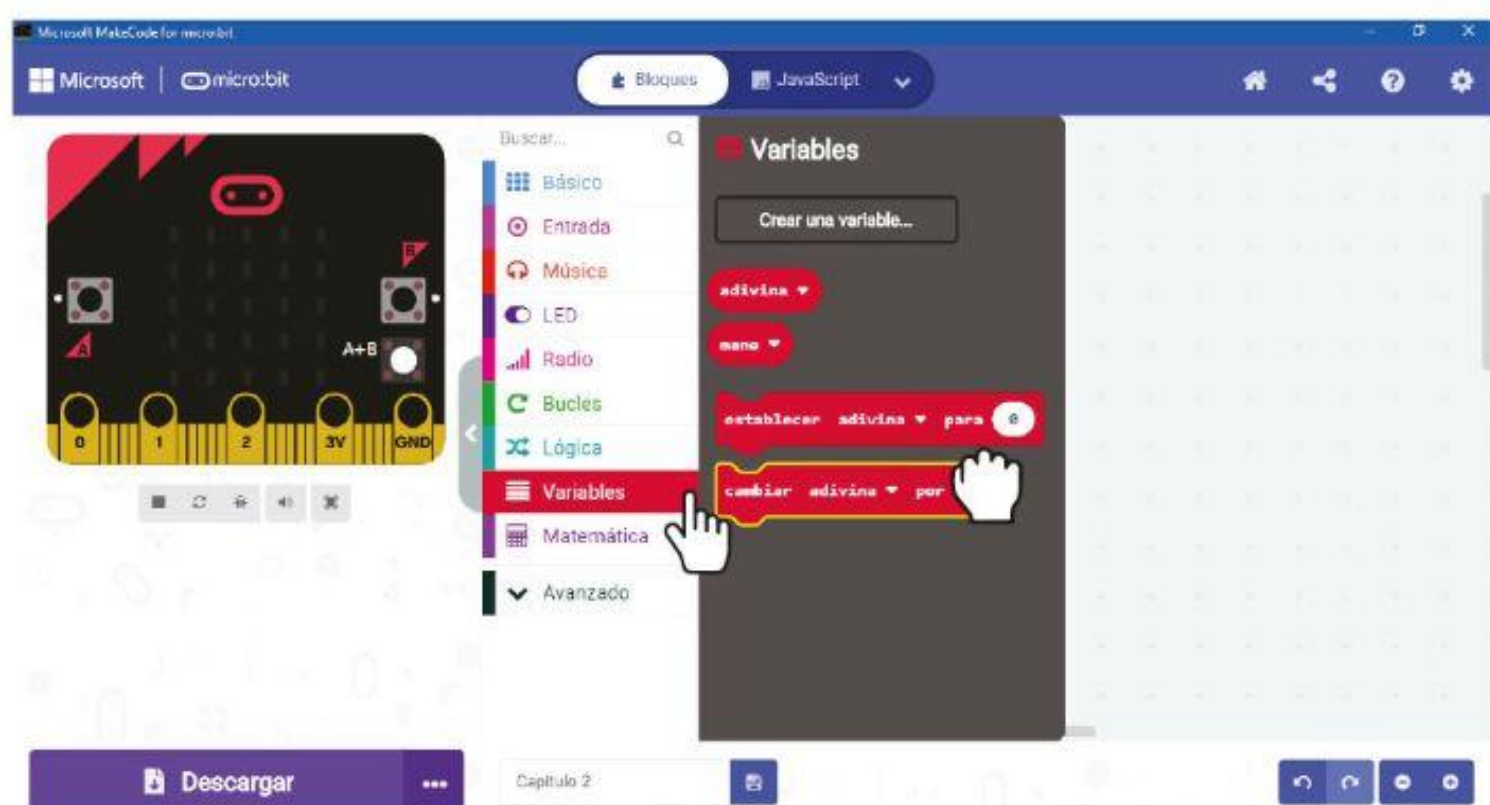
Paso 13 Haz clic en la categoría **[Variables]** y luego selecciona el bloque **[establecer _ para _]**. Inserta el bloque dentro del bloque **[Básico: al iniciar]**. Configura la variable 'adivina' y cambia el valor a 3.



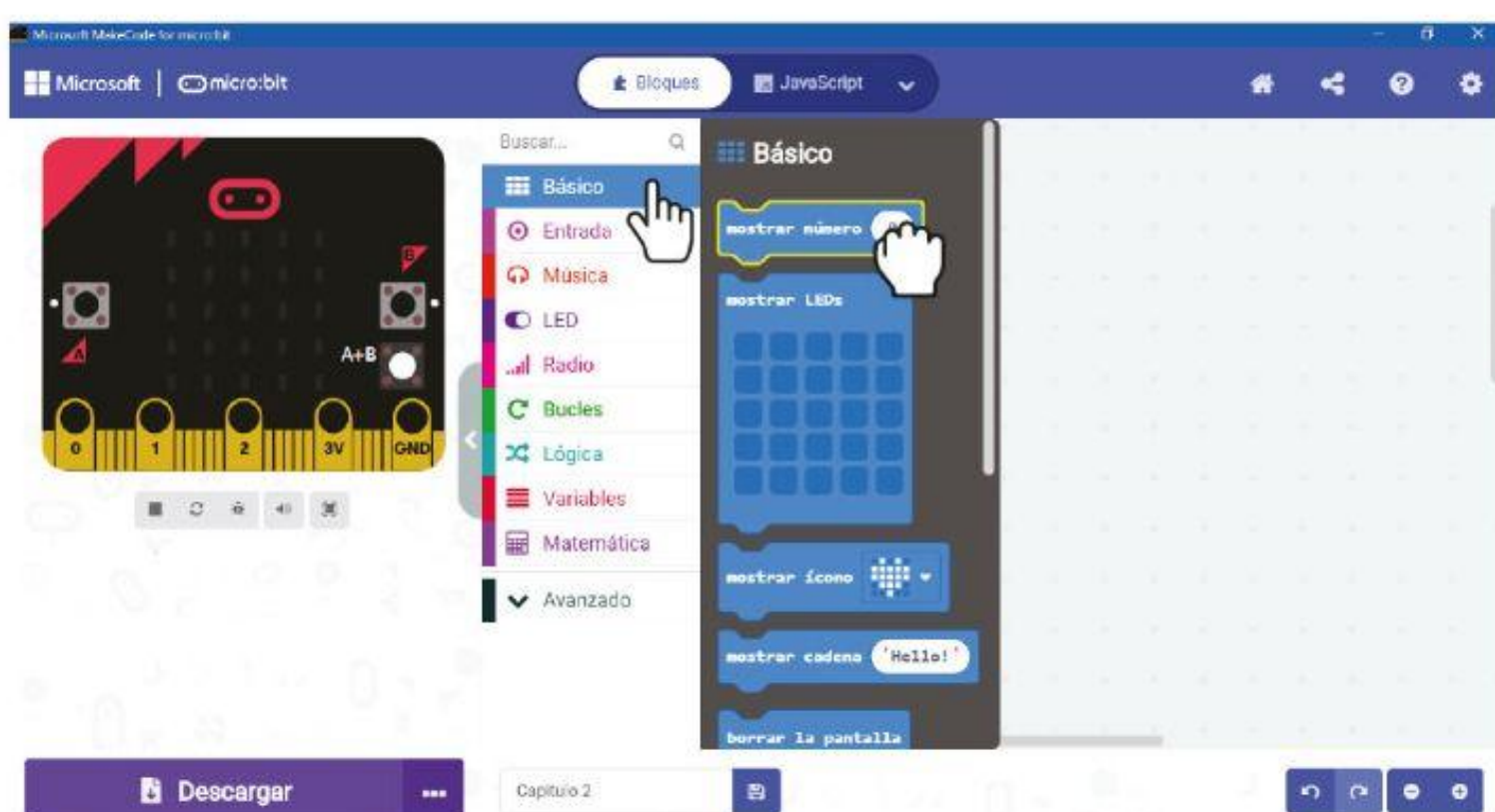
CAPÍTULO 2 : ¡Piedra Papel tijeras!



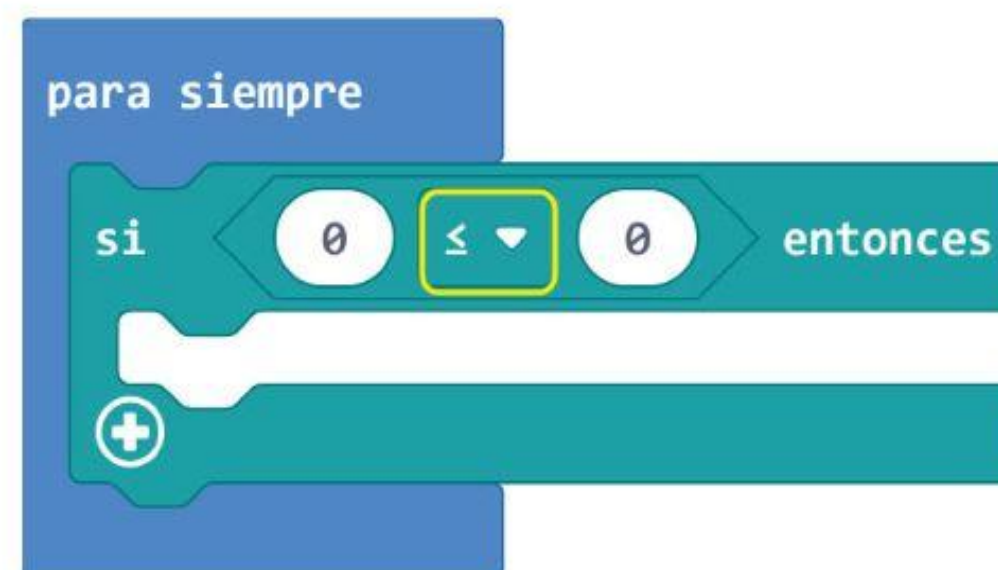
Paso 14 Vuelve a hacer clic en la categoría **[Variables]** y selecciona el bloque **[cambiar _ por _]**. Encaja el bloque en el bloque **[al presionarse el botón _]**. Establece la variable 'adivina' y cambia el valor a -1.



Paso 15 Haz clic en la categoría **[Básicos]** y selecciona el bloque **[mostrar número]**. Duplícalo y encaja los bloques en ambos bloques **[al presionarse el botón _]**.

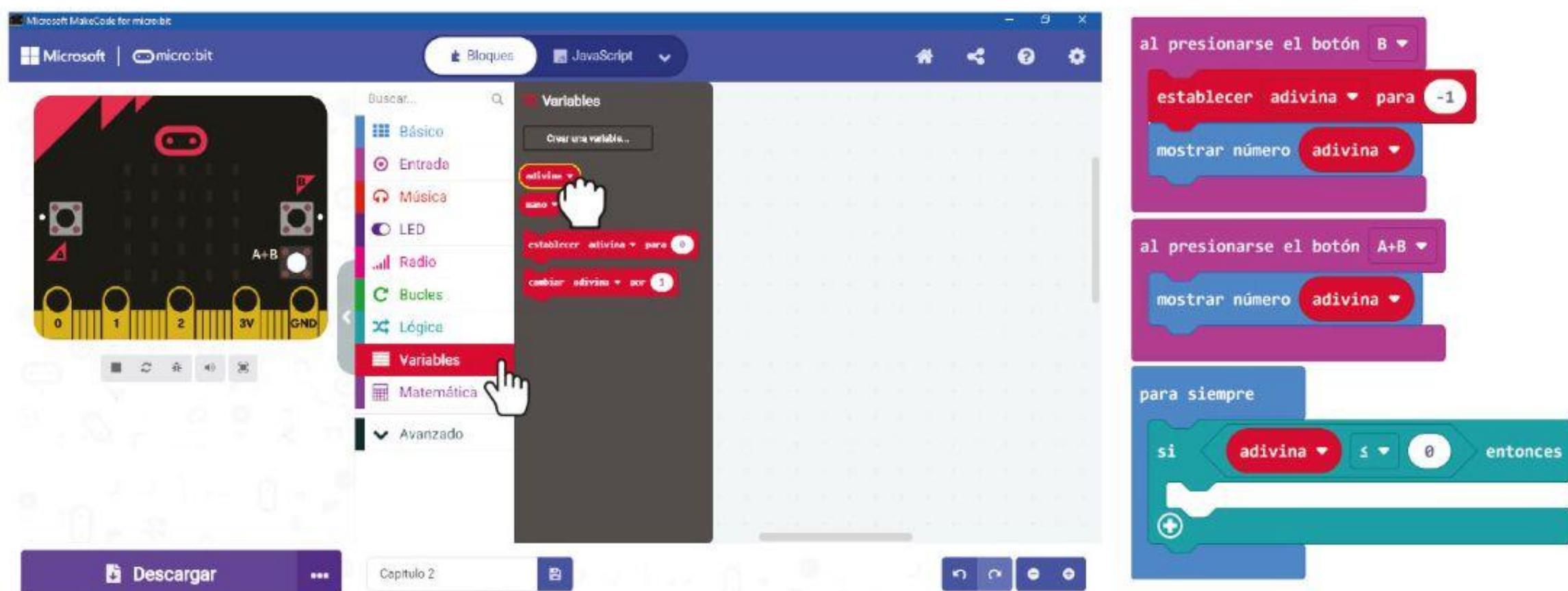


Paso 16 Haz clic en la categoría **[Lógica]** agrega el bloque **[si_entonces]** y el bloque de comparación **[_ = _]**. Ajústalos en el bloque **[Básico]** : **[para siempre]** y cambia el signo a '<='.

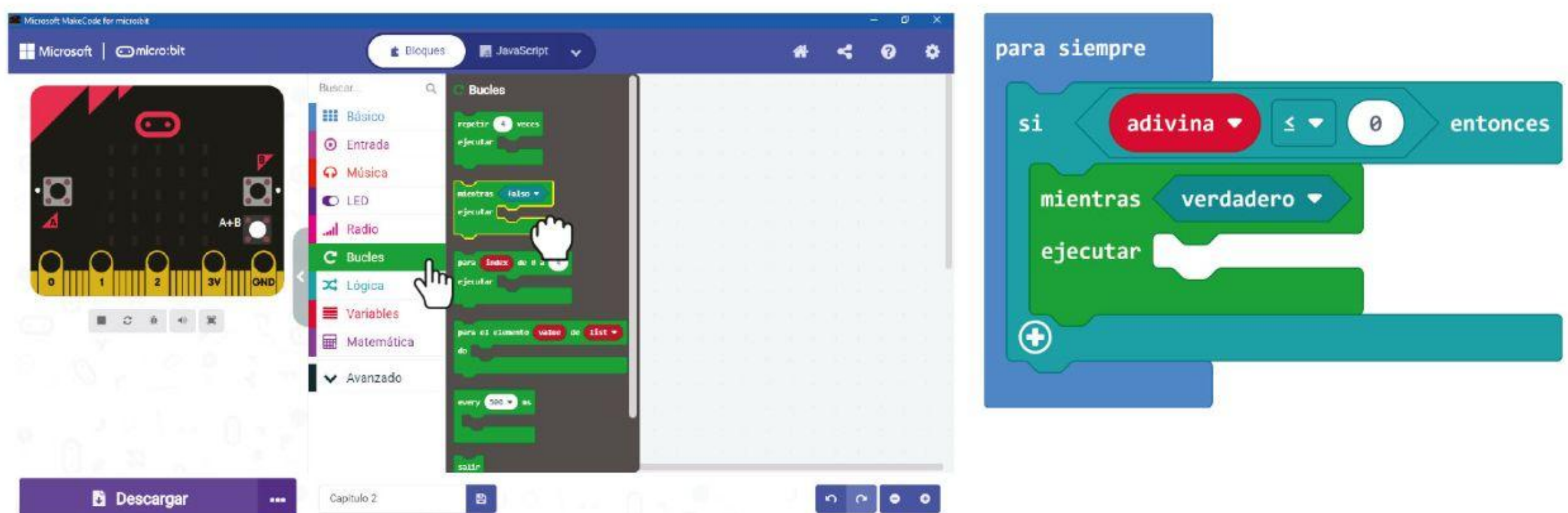


CAPÍTULO 2 : ¡Piedra Papel tijeras!

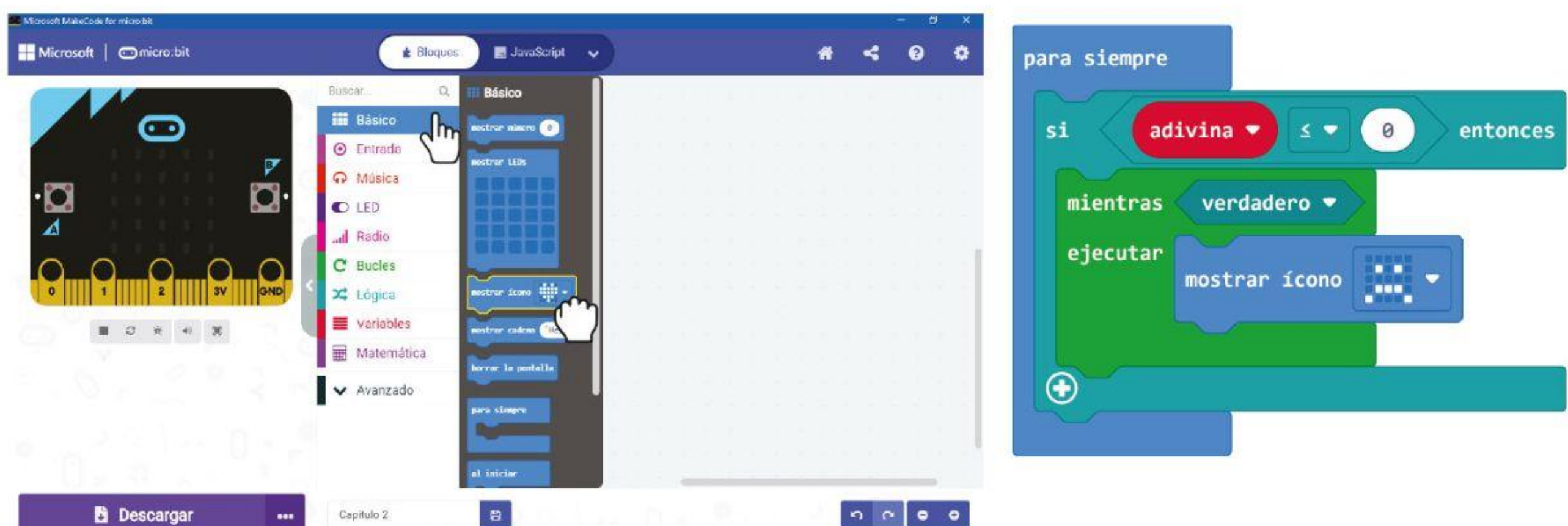
Paso 17 Haz clic en la categoría **[Variables]** y selecciona el bloque **[adivina]**. Duplicalo y ajusta ambos a los bloques. **[mostrar número]**, está a la izquierda del bloque de comparación **[=]**.



Paso 18 Haz clic en la categoría **[Bucles]** y selecciona el bloque **[mientras _ejecutar]**. Colócalo en la ranura del bloque **[si _ entonces]**.



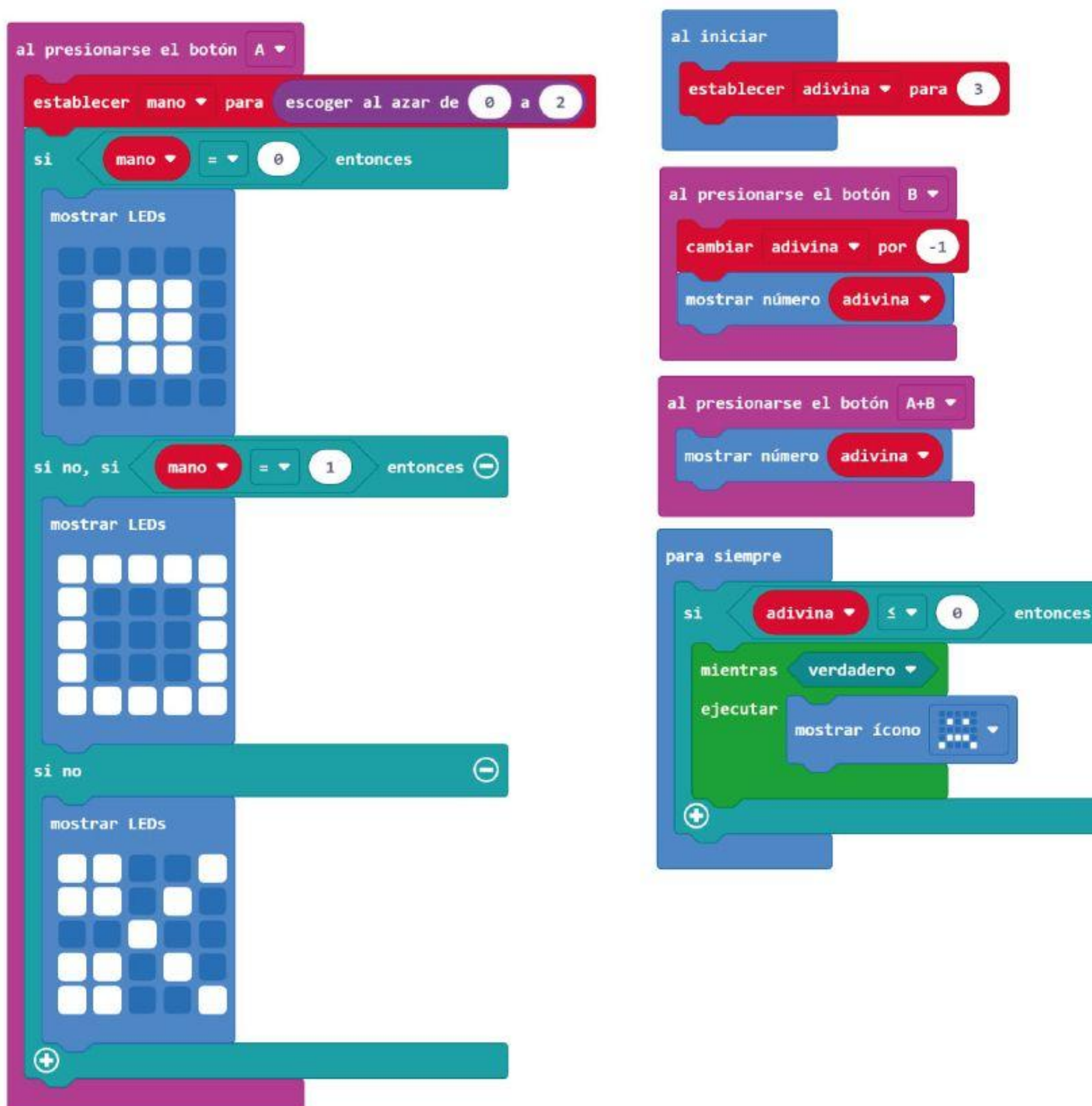
Paso 19 Haz clic en la categoría **[Básico]** y selecciona el bloque **[mostrar ícono]**. Colócalo en la ranura del bloque **[mientras _ejecutar]** y cambia el icono a uno 'cara triste'.



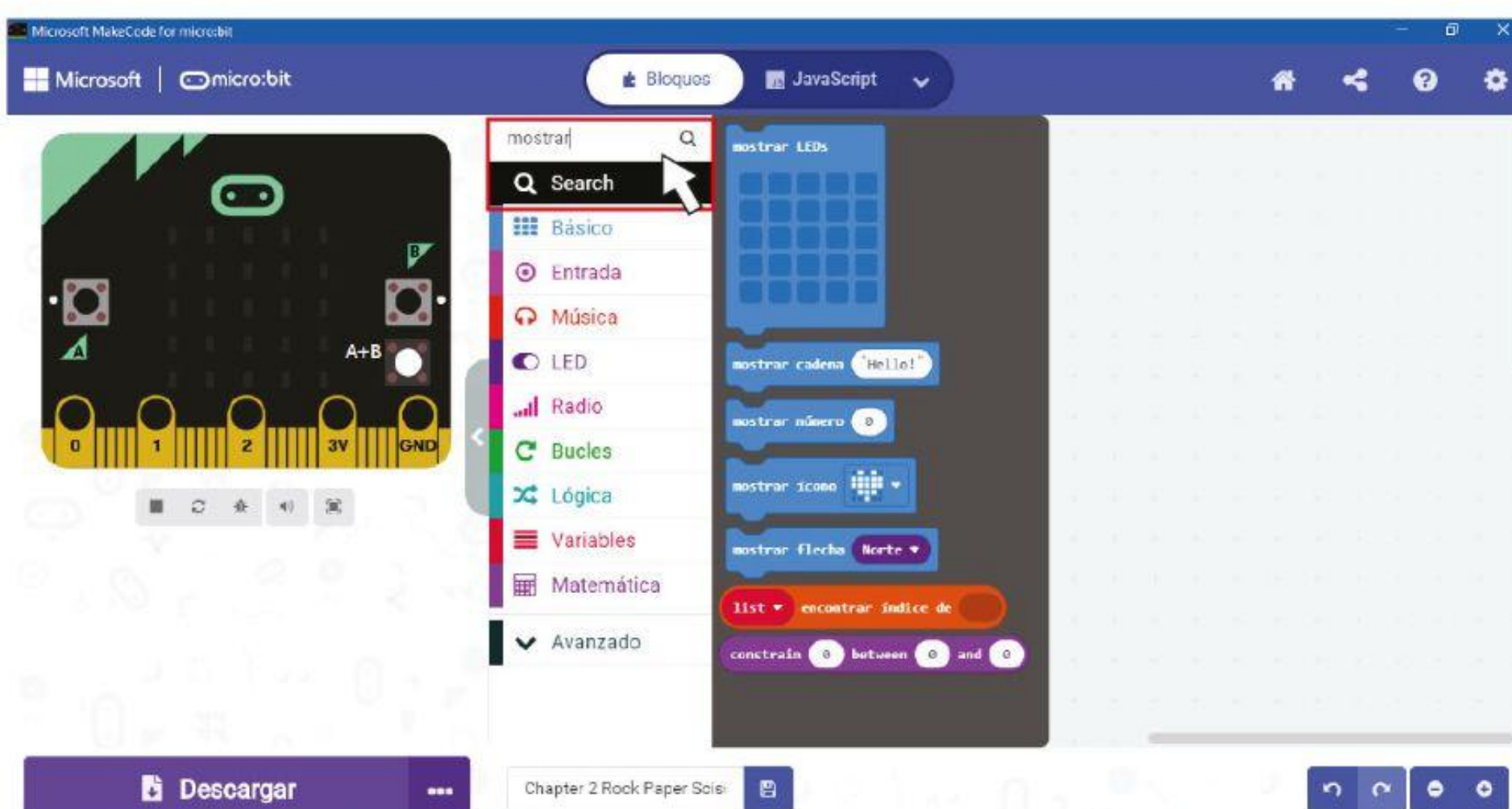
CAPÍTULO 2 : ¡Piedra Papel tijeras!



Paso 20 Aquí está el código completo. Transfiérelolo a tu EDU:BIT y diviértete jugando con tus amigos para determinar quién es el rey o la reina del juego Piedra, Papel o Tijera.

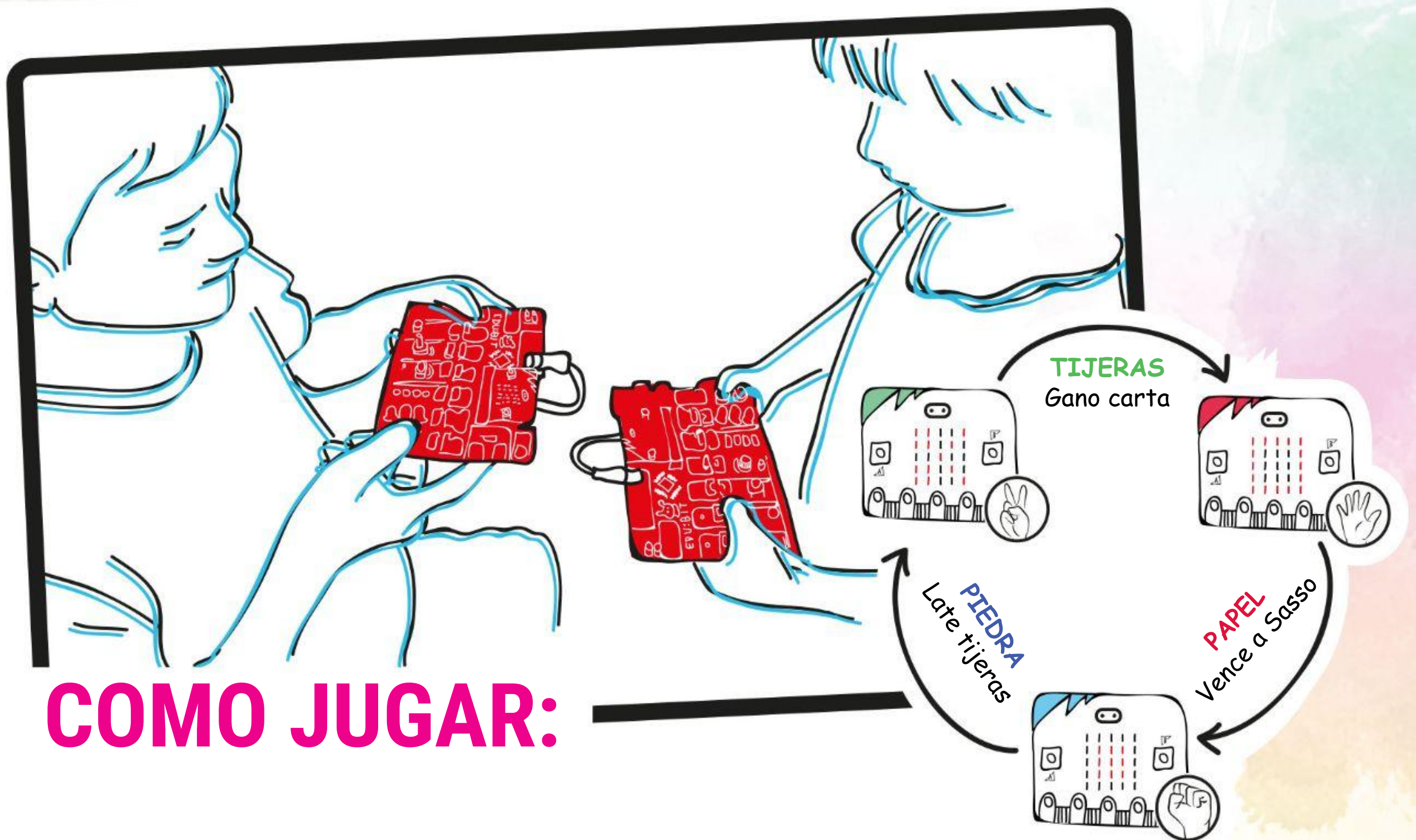


Aquí hay un consejo para ti...
Todos los bloques de código están agrupados por color. Busca los bloques que necesitas en la categoría del mismo color.
Alternativamente, puedes escribir palabras clave en el cuadro de búsqueda.



Vamos a jugar

Piedra, Papel, Tijeras - Versión Avanzada



COMO JUGAR:

- Párate frente a tu oponente. Cuando ambos jugadores estén listos, presione el botón amarillo (Botón A) para ver aleatoriamente piedra, papel o tijera.
- Compara y decide quién gana.
- Si pierdes, debes presionar el botón azul (botón B) en tu EDU: BIT una vez para quitarle la vida.
- Pulsa los botones amarillo y azul al mismo tiempo para comprobar el número de vidas restantes..
- Si un jugador pierde 3 veces, es Game Over y su EDU: BIT mostrará una cara triste

NOTA!

- Para jugar otra ronda, debes REINICIAR la tarjeta para empezar.
- Si no tienes un amigo con quien jugar, siempre puedes jugar contra el simulador en tu MakeCode Editor

MODIFICAR EL

CÓDIGO

En la programación de computadoras, las **variables** se utilizan para almacenar información o valores que se pueden cambiar durante la ejecución (cuando el programa se está ejecutando). Puedes pensar en una variable como un sobre que contiene una hoja de papel con información escrita en ella. La hoja de papel se puede quitar y reemplazar con otra hoja de papel que tenga nueva información.

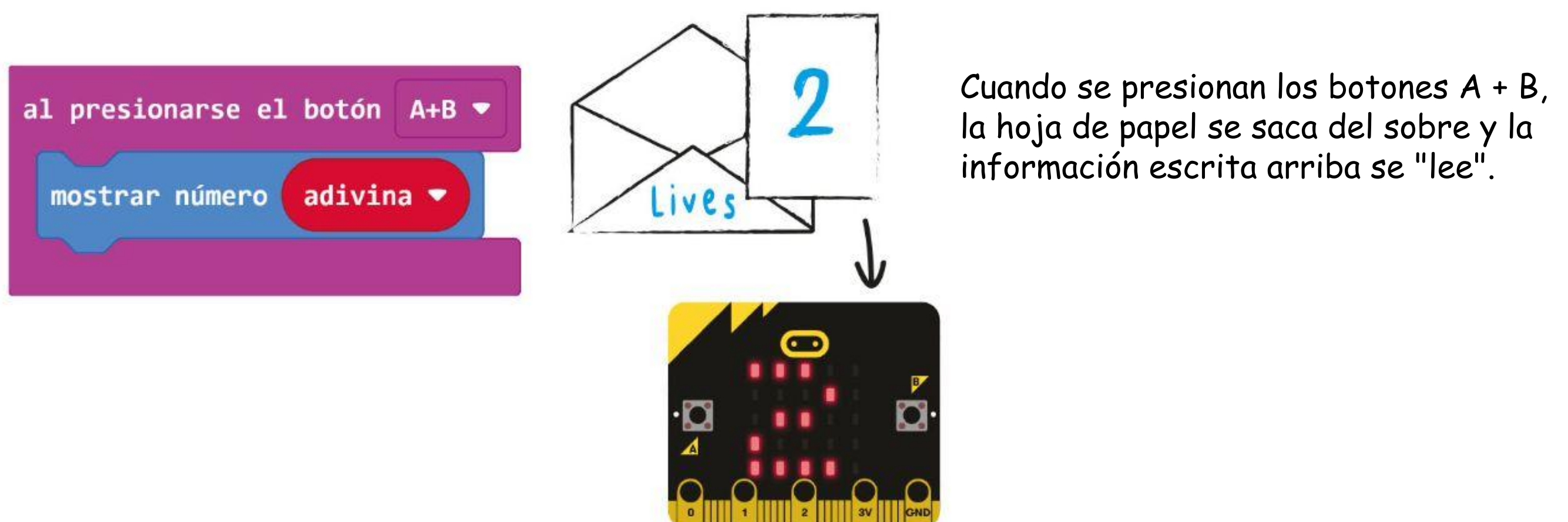
Anteriormente en nuestro código creamos una variable llamada 'vida' y le asignamos un valor inicial de 3.



Luego, cada vez que se presiona el botón B, el valor se reduce en 1.



Al presionar los botones A + B simultáneamente, se mostrará el valor actual de la variable 'vida' en la matriz de LED





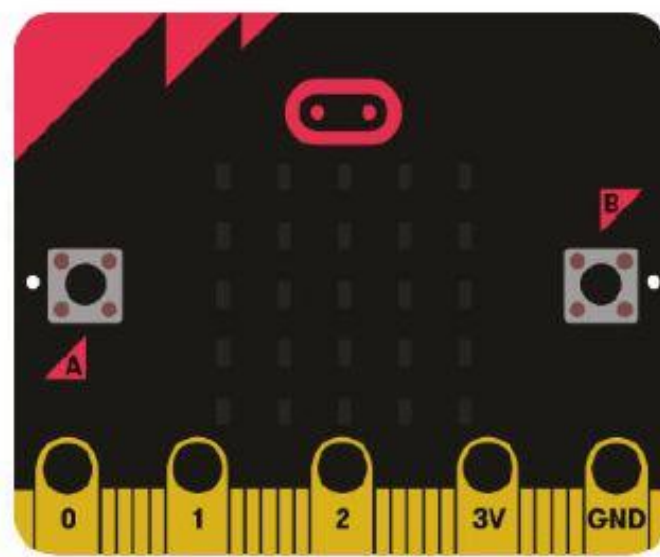
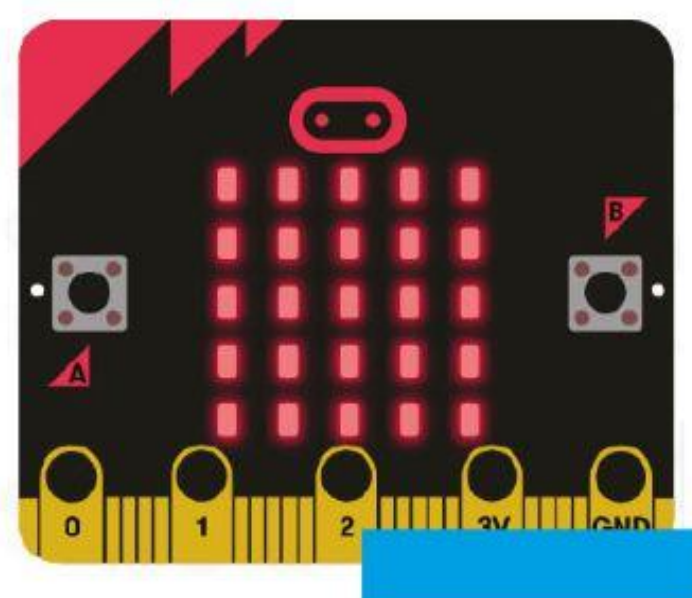
EXPLORA OTROS BLOQUES

Más allá del bloque **[al presionarse el botón]**, también puedes usar otros bloques en la categoría **[Entradas]** para llevar a cabo la planificación de eventos. Las acciones realizadas por el usuario, como presionar un botón o agitar la tarjeta, se conocen como 'eventos'.

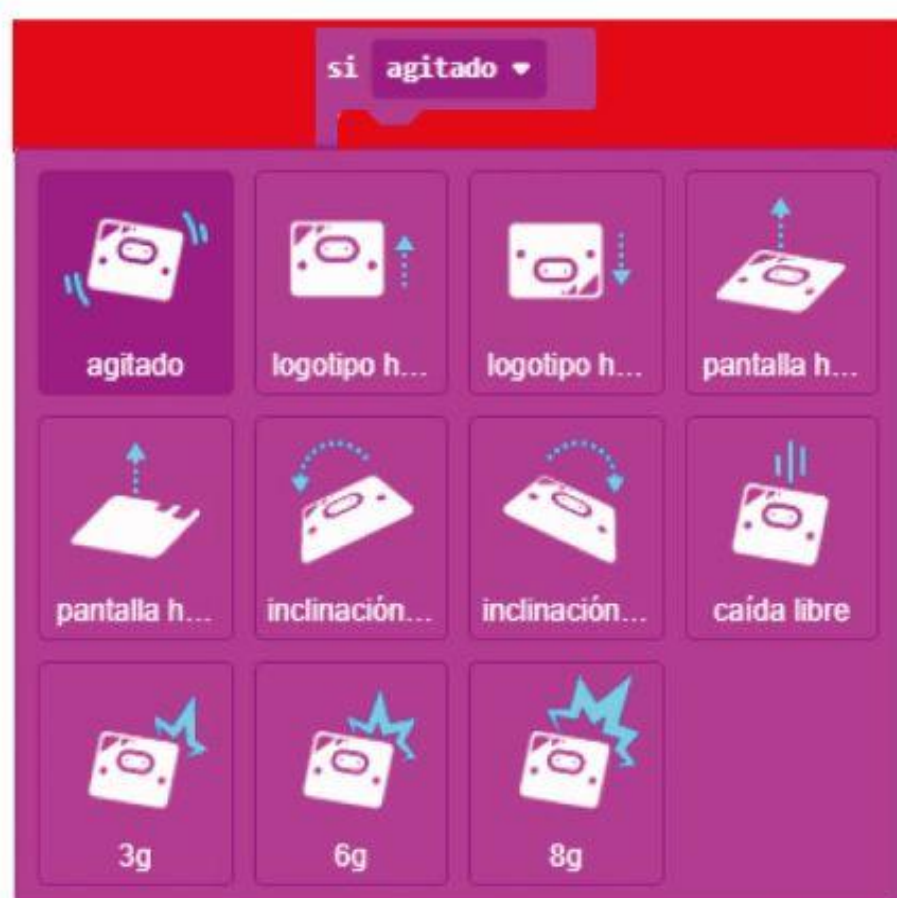
El siguiente código, por ejemplo, hará que la matriz de LED se encienda durante 1 segundo cuando se agite la placa. Darle una oportunidad.



Resultado



Si haces clic en el botón **[si agitado]** presente en el bloque, aparecerá un menú emergente que muestra una selección de otros eventos. Intenta programar EDU: BIT para mostrar un icono diferente para cada uno de estos eventos. ¡Diviértete explorando!



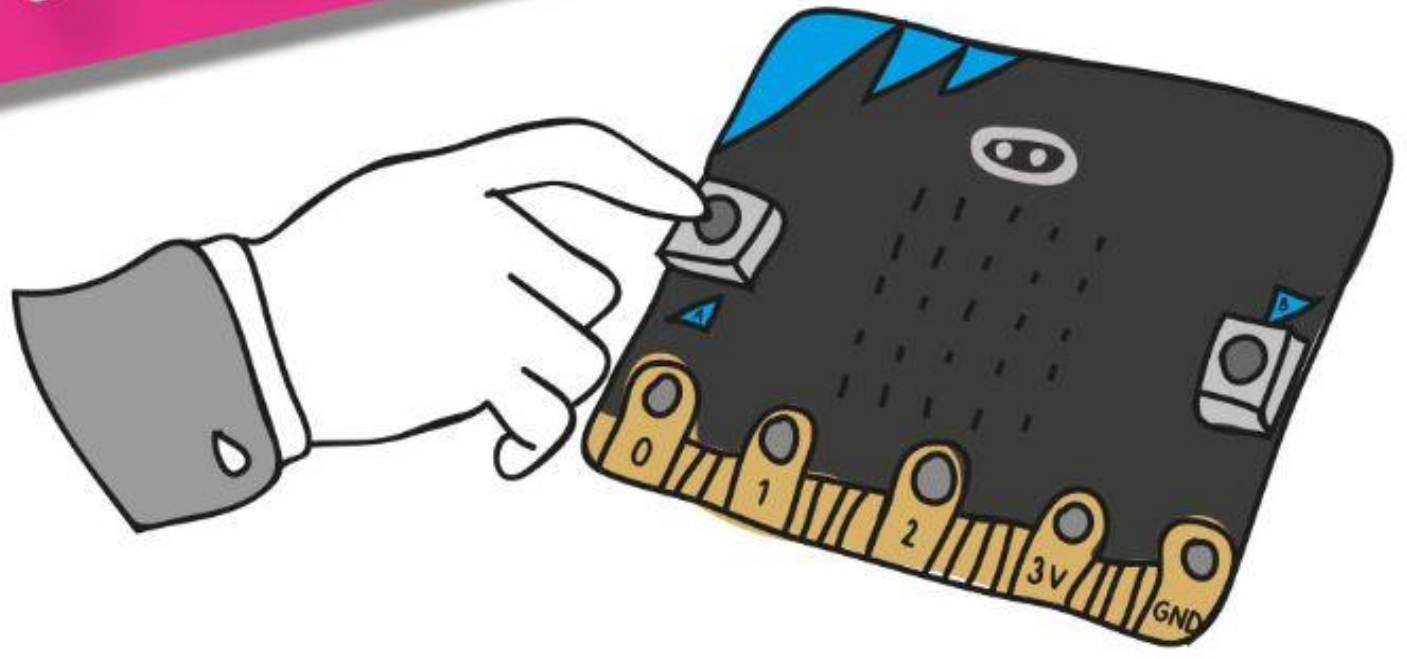
EDU: BIT puede detectar el evento de sacudida o conocer su orientación en el espacio porque hay un sensor de movimiento en la tarjeta micro: bit.



COSAS DIVERTIDAS!

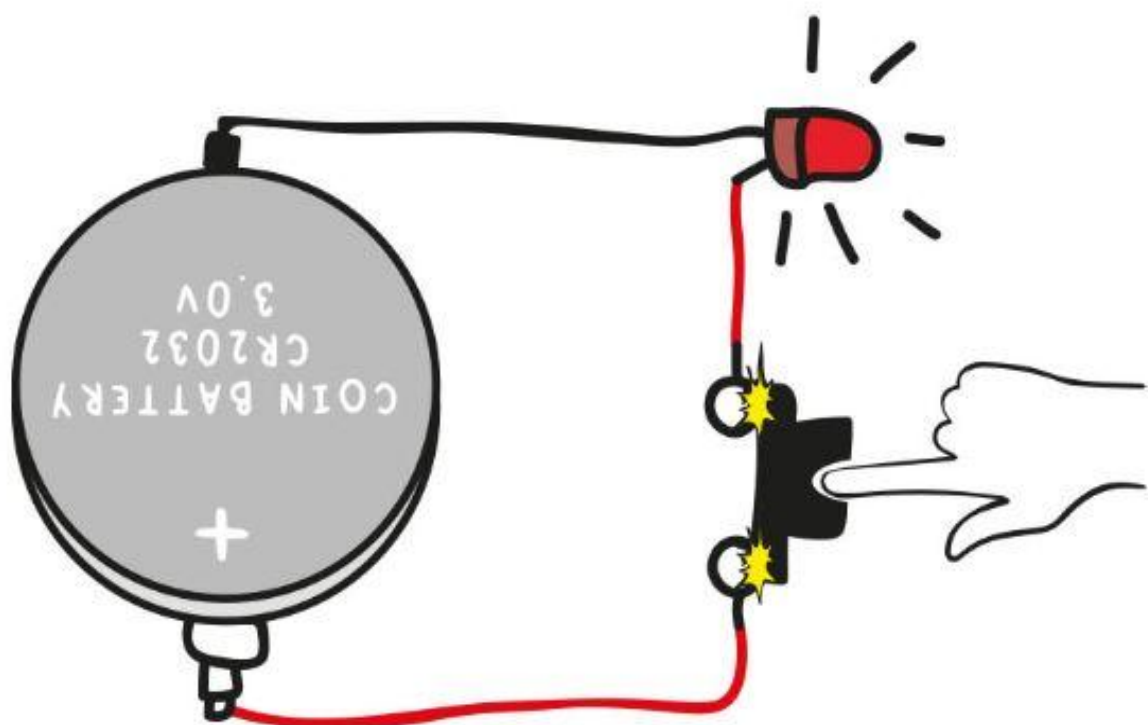
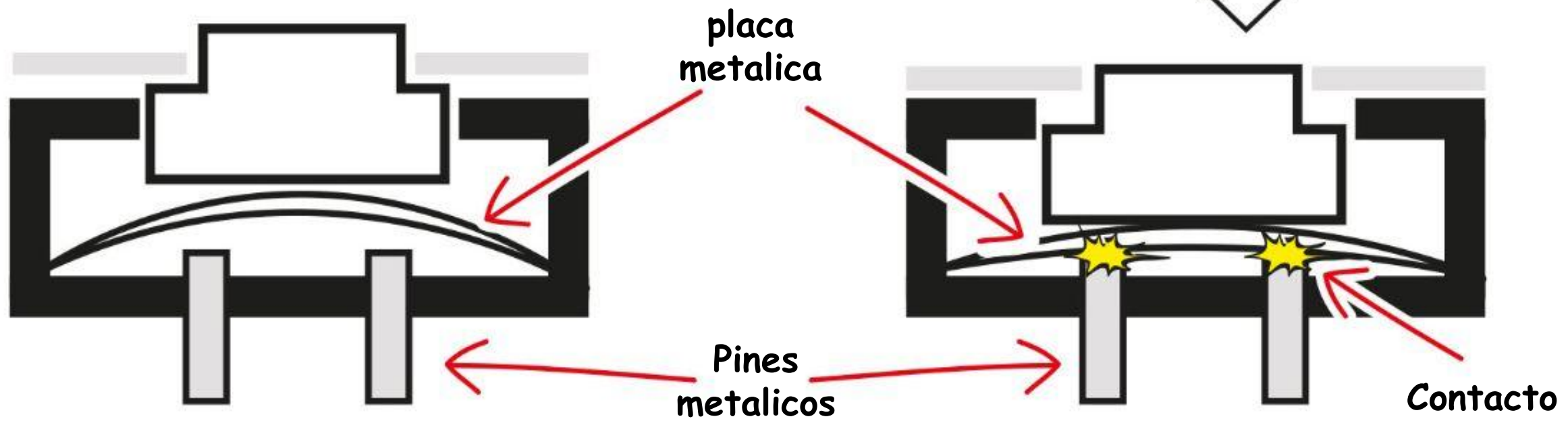


Un **botón** es un dispositivo de entrada con solo dos estados posibles: presionado o no Presionado.



NO PULSADO

PULSADO



¡Cuando se presiona el botón, se cierra el circuito eléctrico y se enciende el LED! ¿Adivina qué sucede cuando sueltas el botón?



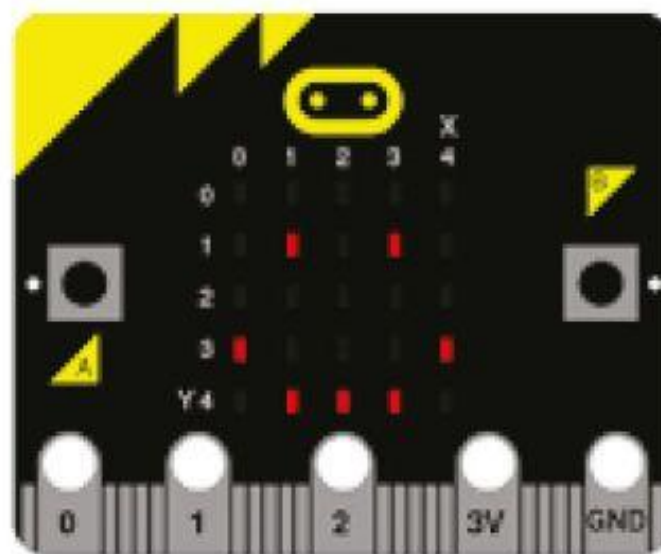
Los botones negro, gris, verde y blanco se utilizan normalmente para funciones ON/OFF, mientras que el ROJO se utiliza como botón de pánico o emergencia para parar maquinaria.

¡Aprende más!



youtu.be/t_Qujjd_38o

DESAFÍO DE LA APLICACIÓN



Programa EDU: BIT para que funcione como contador para registrar la presencia de alumnos. Las niñas deben presionar el Botón A al ingresar al salón de clases; los niños tienen que presionar el botón B.

En el inicio	Mostrar una cara sonriente. Establecer las variables Niñas = 0 y Niños = 0
Cuando presionas el botón A (amarillo)	Aumenta la variable Niñas en 1
Cuando presionas el botón B (azul)	Incrementa la variable Niños en 1
Cuando presionas los botones A + B	Se desplaza en la pantalla de LED esta información: Total = (Niñas + Niños); Chicas = (Chicas); Chicos = (Chicos)

CAPÍTULO 3

Escuchemos algo de música ~

Bit Música (zumbador piezoeléctrico + conector de audio)



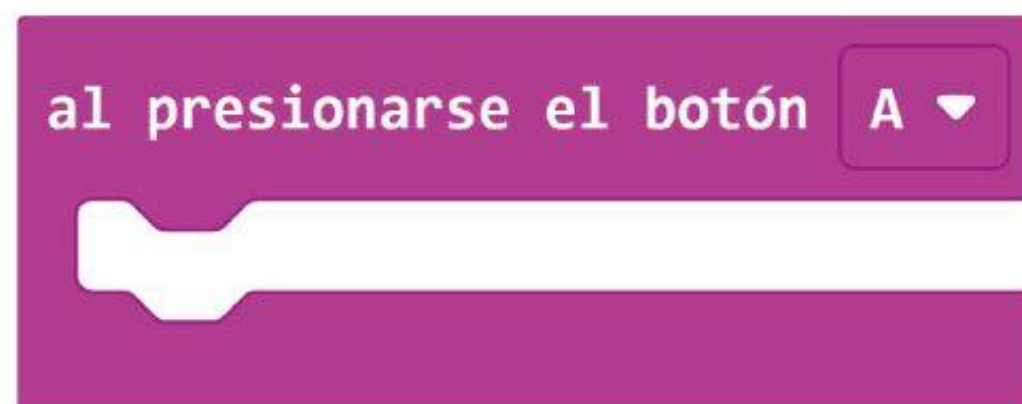
¡Escanéame!



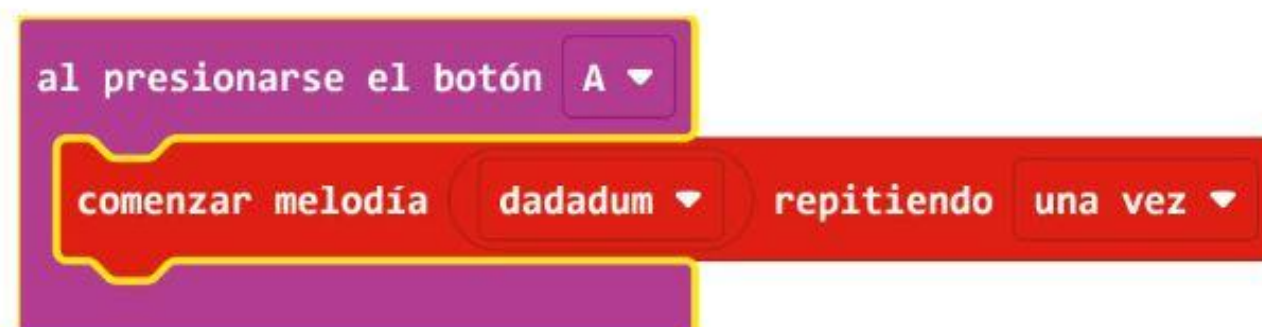
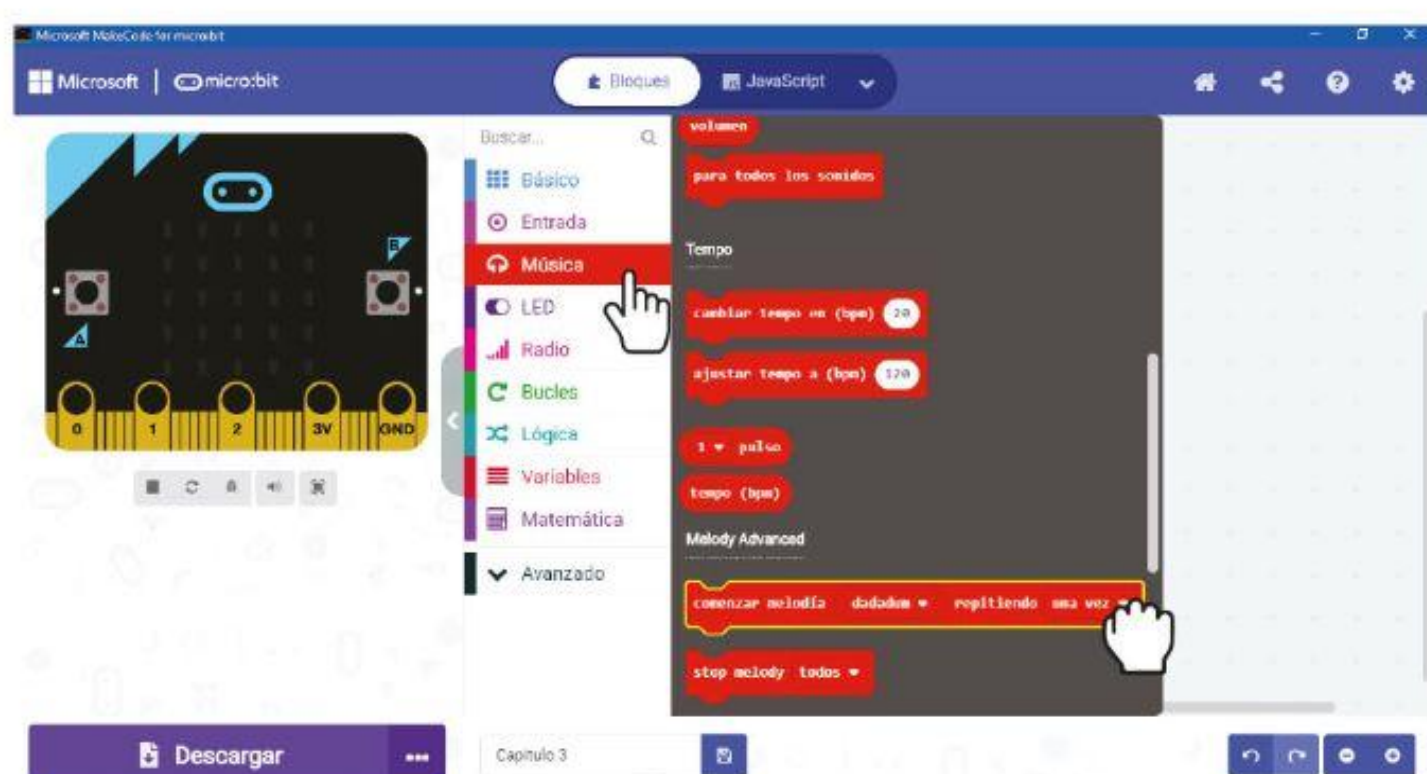
CAPÍTULO 3 : Escuchemos algo de música ~

PROGRAMAMOS!

Paso 1 Crea un nuevo proyecto en el Editor de MakeCode. Haz clic en la categoría **[Entradas]** y luego selecciona el bloque **[al presionarse el botón_]**.



Paso 2 Haz clic en la categoría **[Música]** y luego selecciona el bloque **comenzar melodía _ repitiendo _**.



Paso 3 Haz clic en **[dadadum]** y selecciona la melodía de 'cumpleaños' de la lista desplegable.



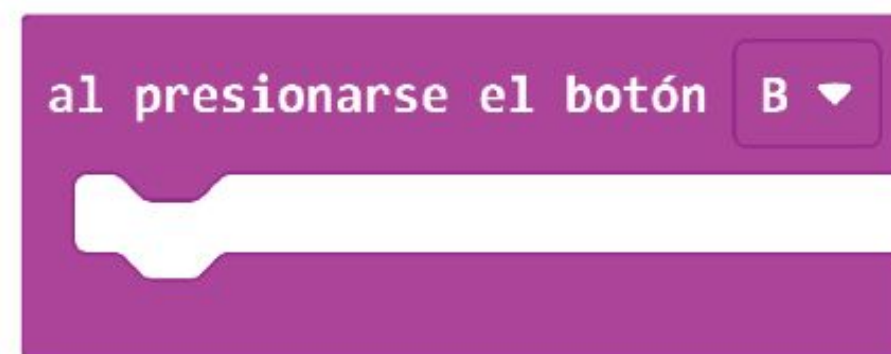
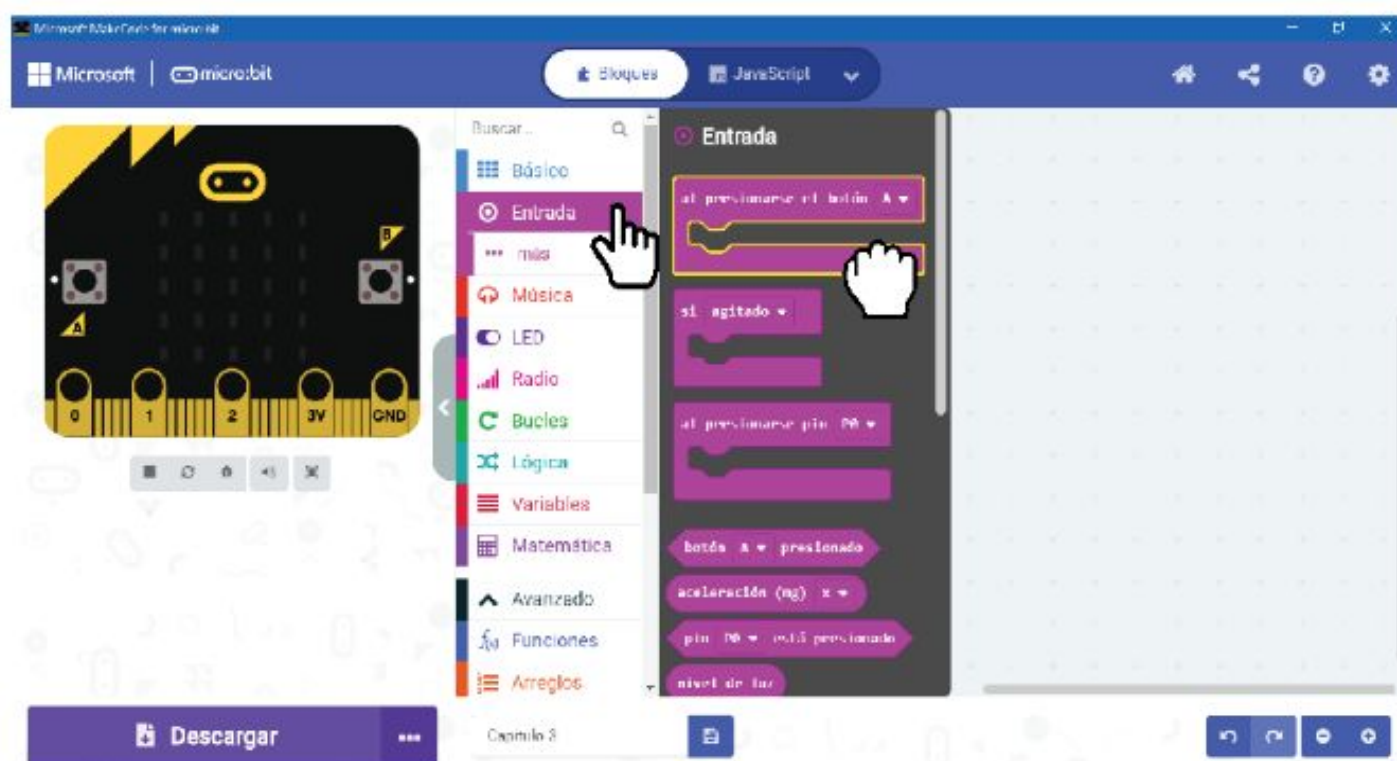
Haz clic en el botón A de tu simulador en la pantalla. ¿Oyes un sonido familiar? Diviértete probando las otras melodías también ~* asegúrate de que los altavoces de tu computadora estén encendidos



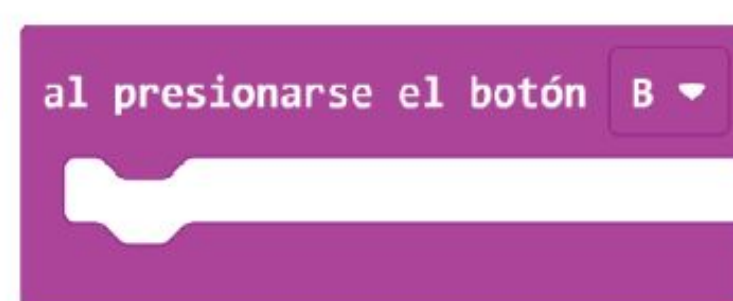
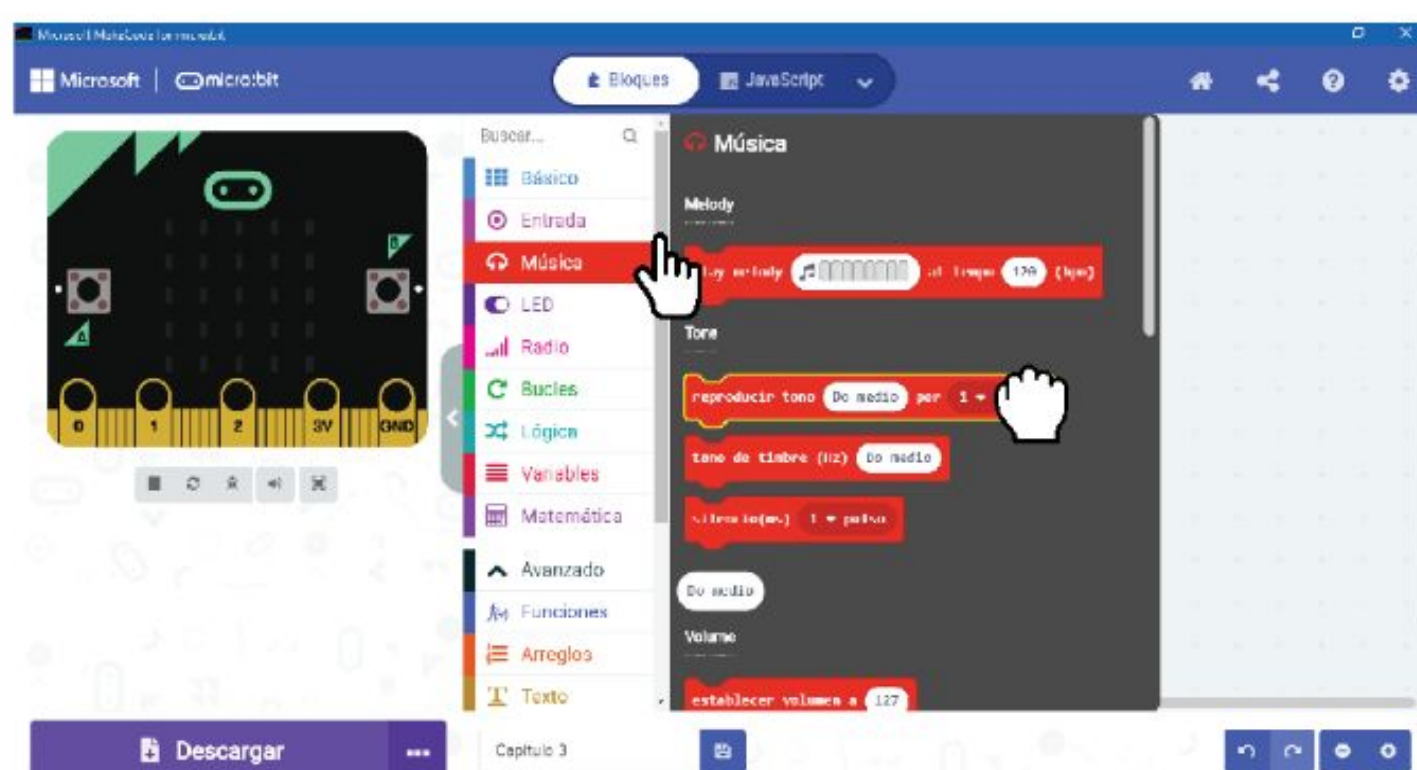
Además de la lista de melodías preestablecidas, también puedes programar EDU: BIT para reproducir cualquier canción que desees. Tratemos de tocar esta melodía pegadiza ~



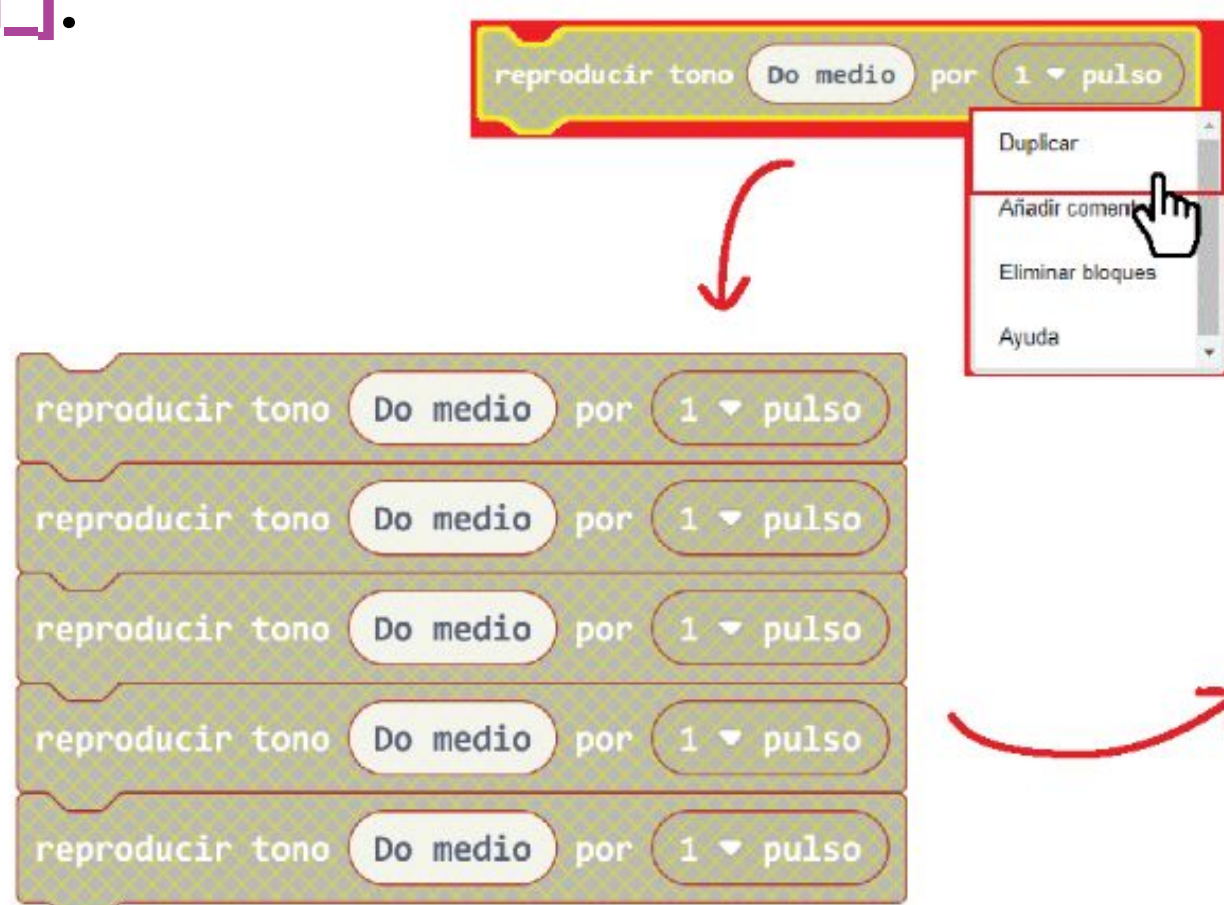
Paso 4 Haz clic en la categoría **[Entradas]** y luego selecciona el bloque **[al presionarse el botón _]**. Selecciona el botón **“B”**.



Paso 5 Haz clic en la categoría **[Música]** y luego en el bloque **[reproducir tono_ por_]**.

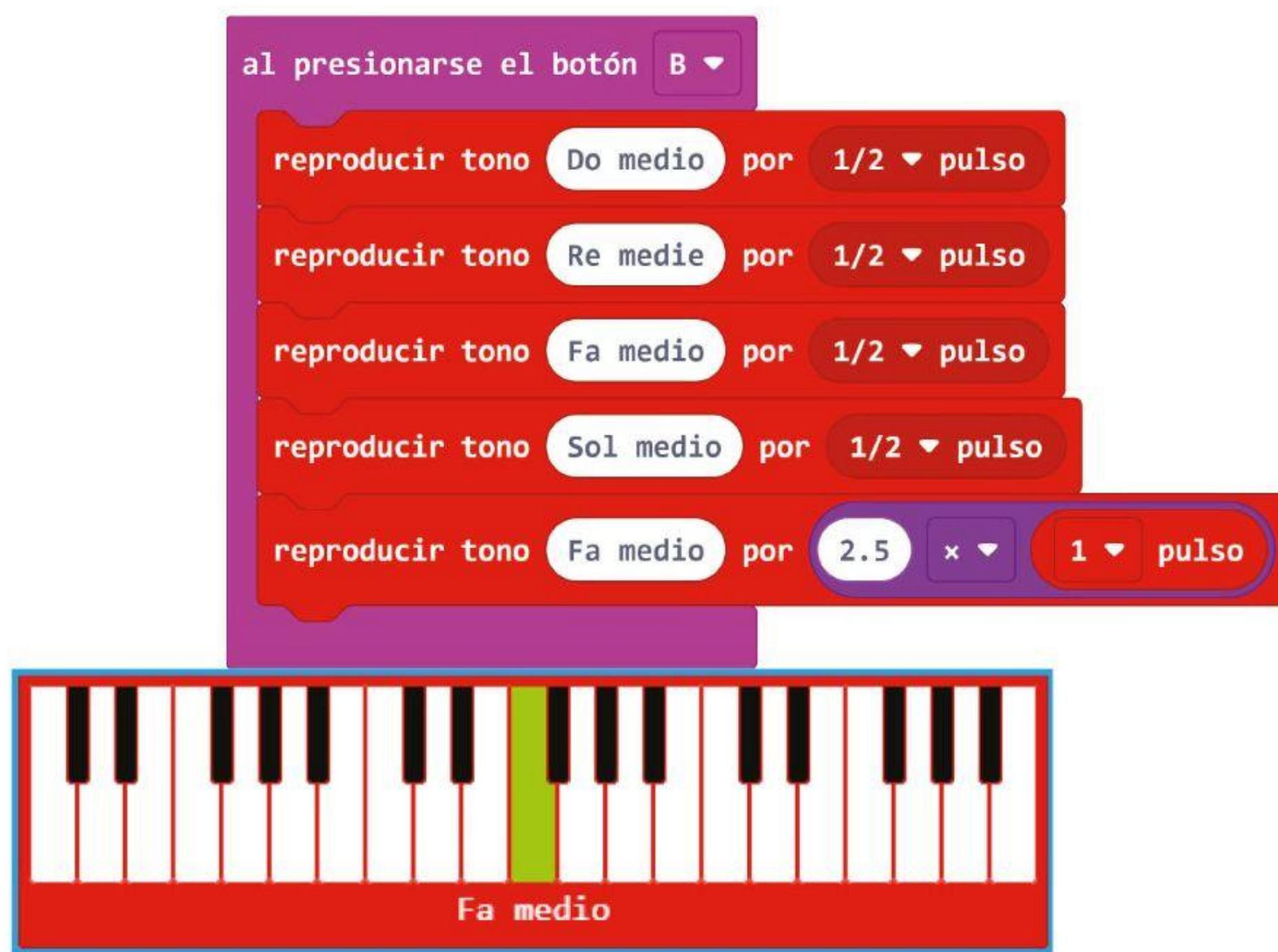


Paso 6 En el espacio de trabajo, haz clic con el botón derecho en el bloque **[reproducir tono_ por_]** y luego 'Duplicar'. Repite hasta que tengas un total de 5 bloques **[reproducir tono_ por_]**. Coloca los bloques dentro **[al presionarse el botón_]**.



CAPÍTULO 3 : Escuchemos algo de música ~

Paso 7 Selecciona 'nota' y 'duración' de estos bloques **[reproduce tono_ por_]** como el ejemplo de código a continuación (C - D - F - G - F):



NOTA!

El bloque morado se tomó de la categoría [Matemáticas].



Haz clic en el botón B de tu simulador en la pantalla. ¿Puedes adivinar la canción?



Cuando estés programando es recomendable revisar periódicamente el código para asegurarte de que lo estás haciendo bien. Puedes utilizar el simulador para este fin..

Paso 8 Continúa programando el resto de la canción agregando más bloques **[reproducir tono_ por_]** y cambia el 'tono' y los 'tiempos' según corresponda. Puede consultar la página siguiente para ver las notas y los tiempos..



I Will Follow You



I
will
fol-
low
you,

Fol-
low
you
wher-
ev-
er
you
may
go,

There
is
n't
an
o-
cean
too
deep

al presionarse el botón B ▼

reproducir tono Do medio por 1/2 ▼ pulso

reproducir tono Re medie por 1/2 ▼ pulso

reproducir tono Fa medio por 1/2 ▼ pulso

reproducir tono Sol medio por 1/2 ▼ pulso

reproducir tono Fa medio por 2.5 x ▼ 1 ▼ pulso

silencio(ms) 1/2 ▼ pulso

reproducir tono Do medio por 1/2 ▼ pulso

reproducir tono Re medie por 1/2 ▼ pulso

reproducir tono Fa medio por 1/2 ▼ pulso

reproducir tono Re medie por 1/2 ▼ pulso

reproducir tono Fa medio por 1/2 ▼ pulso

reproducir tono La medio por 1/2 ▼ pulso

reproducir tono Do alto por 1.5 x ▼ 1 ▼ pulso

reproducir tono La medio por 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

silencio(ms) 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono La medio por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Do alto por 2 ▼ pulso

silencio(ms) 1/2 ▼ pulso

A
moun-
tain
so
high
it
can
keep,

Keep
me
a-
way
...
...

reproducir tono Do alto por 1/2 ▼ pulso

silencio(ms) 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Re alto por 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

reproducir tono Si medio por 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

silencio(ms) 1/2 ▼ pulso

reproducir tono Do alto por 1/2 ▼ pulso

reproducir tono La medio por 1/2 ▼ pulso

reproducir tono Sol medio por 1/2 ▼ pulso

reproducir tono La medio por 1/2 ▼ pulso

reproducir tono Sol medio por 1/2 ▼ pulso

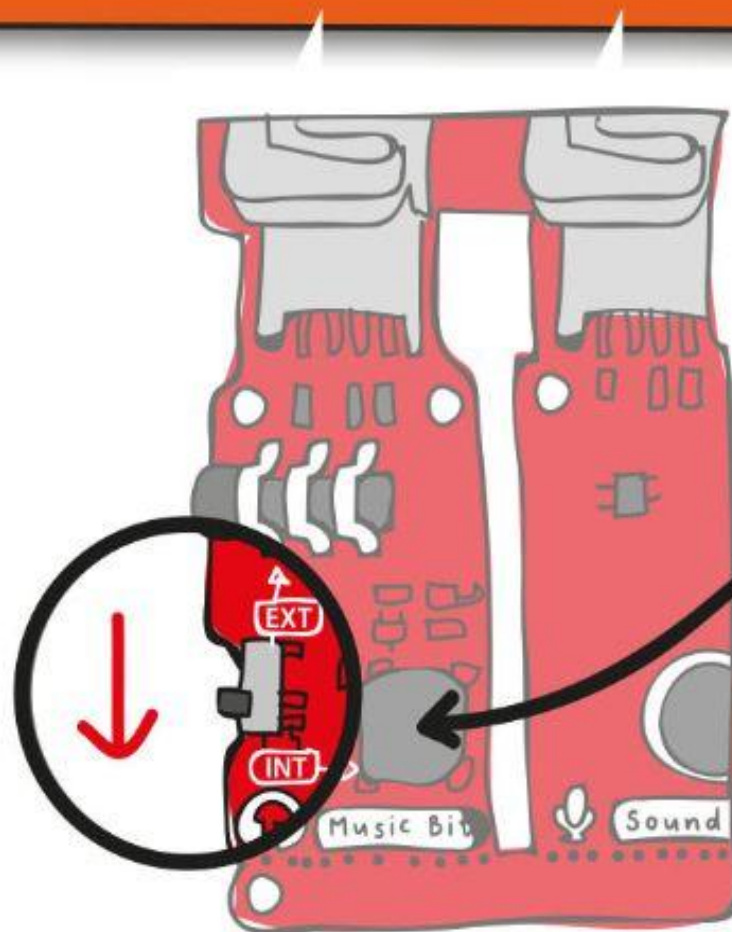
reproducir tono Fa medio por 2.5 x ▼ 1/2 ▼ pulso

silencio(ms) 1 ▼ pulso



CAPÍTULO 3 : Escuchemos algo de música ~

EDU: BIT reproducirá la canción "Te seguiré" cada vez que presiones el botón B. Recuerda conectar la fuente de alimentación de su EDU: BIT y encender el zumbador piezoeléctrico moviendo el interruptor a la posición INT (interna).

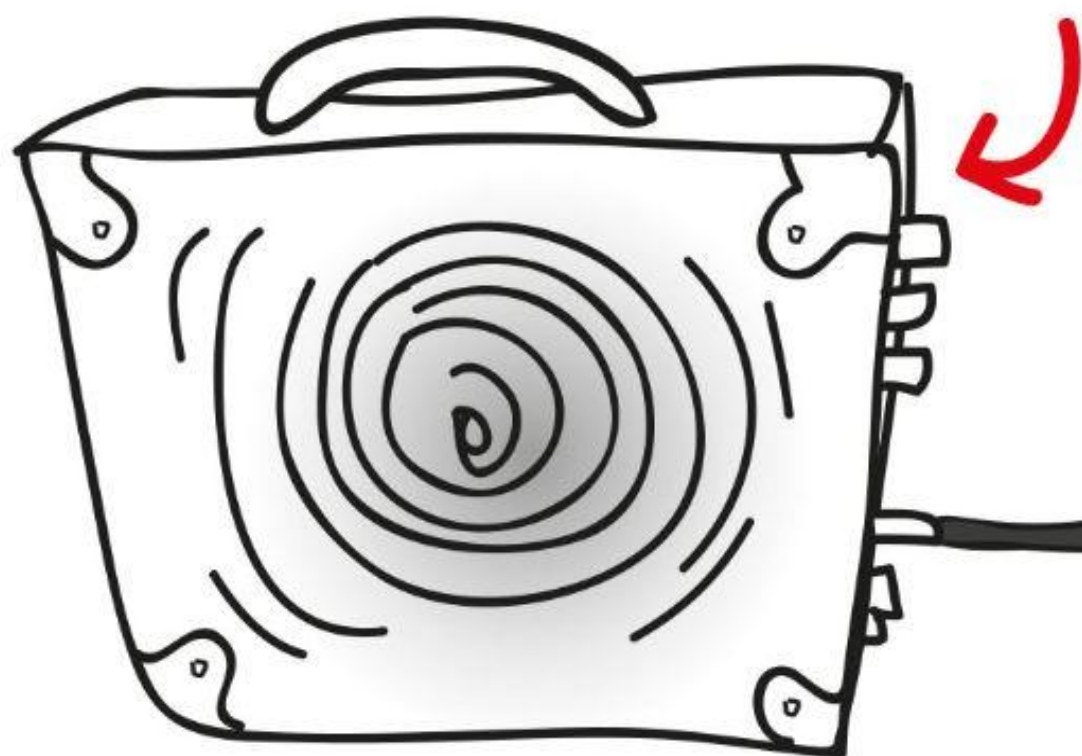


Buzzer
Piezoeléctrico

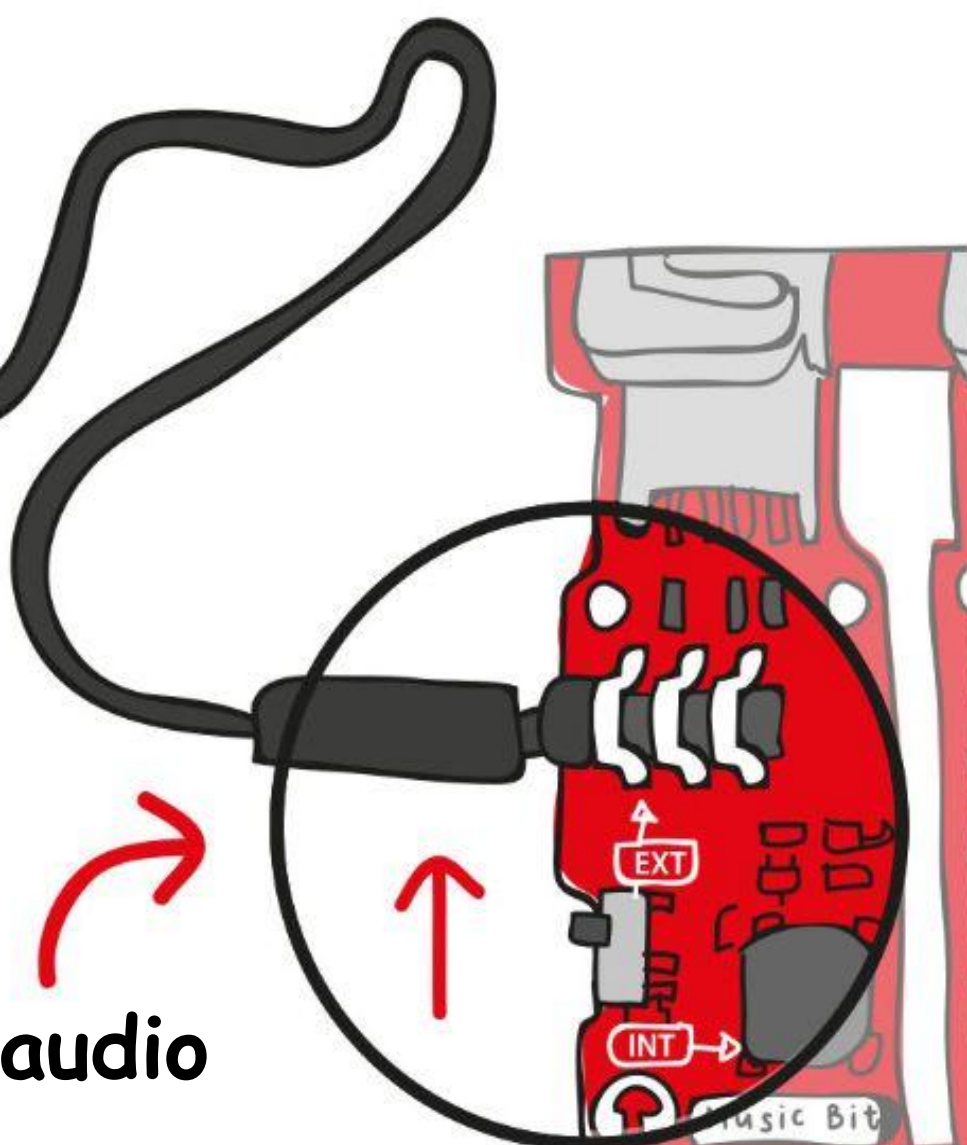


Alternativamente, puedes conectar altavoces externos o auriculares al conector de audio en EDU: BIT. Deberás mover el interruptor a la posición EXT (externa).

Altavoz



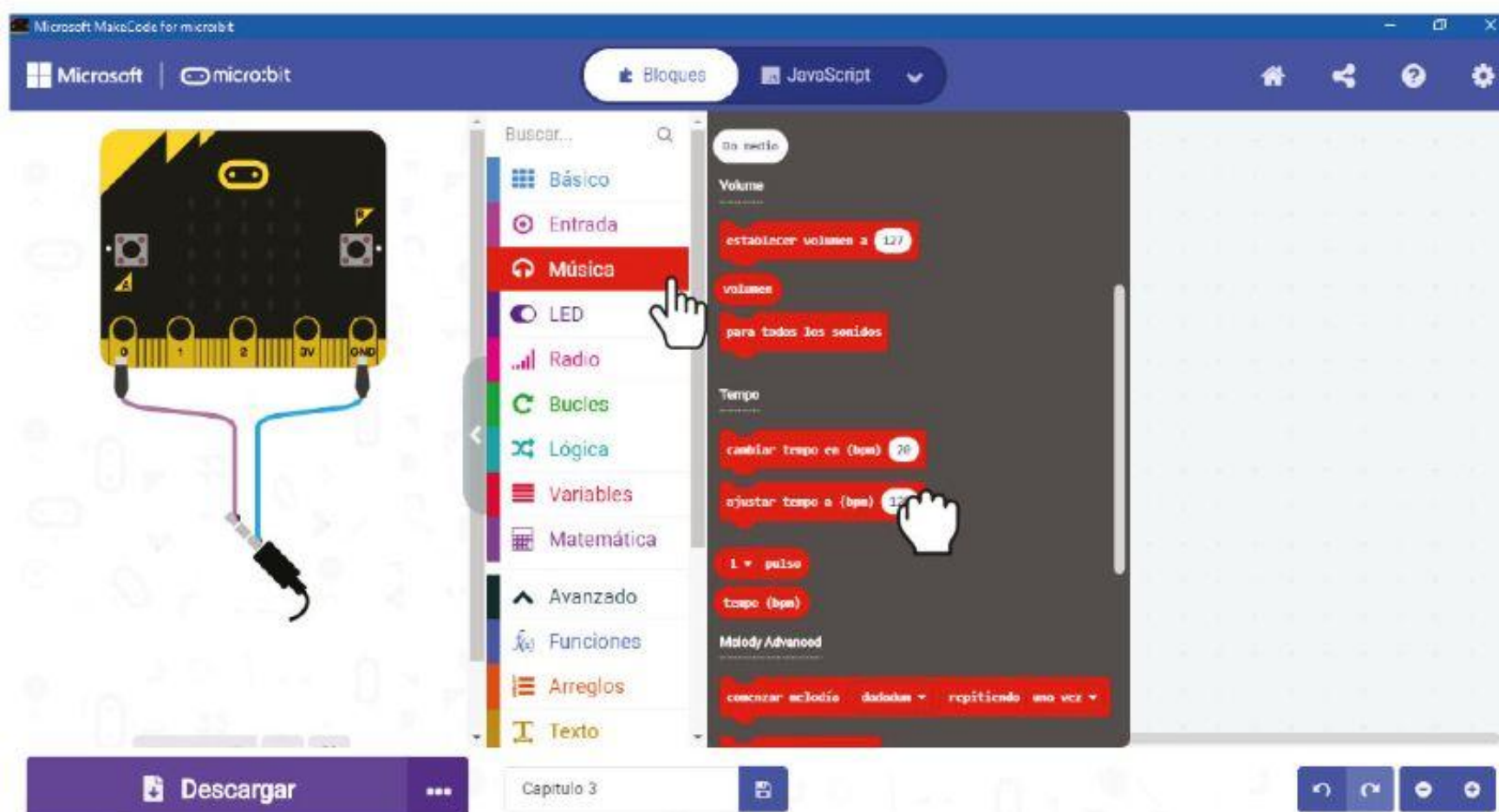
Conector audio





¿La música es demasiado baja? o demasiado fuerte? Puede agregar un bloque [establecer volumen _] para cambiar el volumen, que puede tener un valor entre 0 y 255 (volumen máximo).

Paso 10 Haz clic en la categoría **[Música]** y luego en el bloque **[establecer volumen a _]**. Encaja el bloque dentro del bloque. **[al iniciar]** y cambia el valor del volumen a 200.

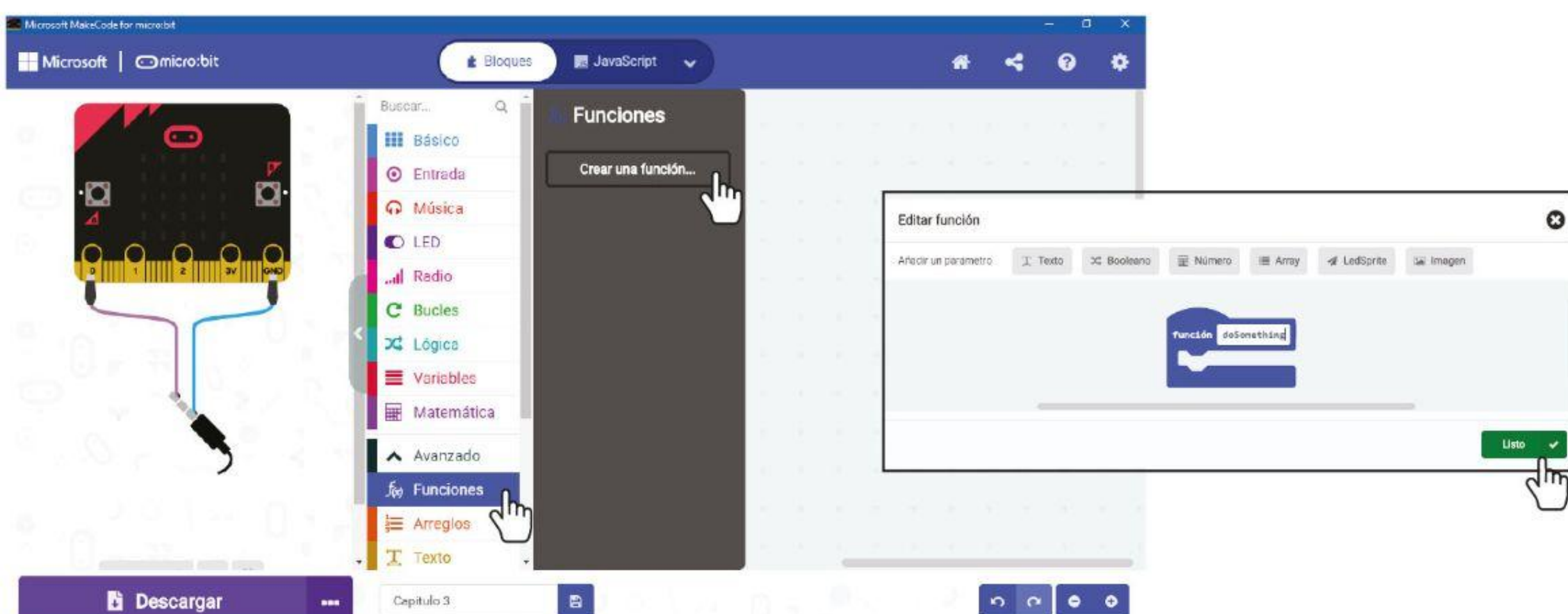


NOTA!

Una vez que se ha definido una función, se puede usar varias veces en su programa sin tener que volver a escribir los mismos bloques de código cada vez.

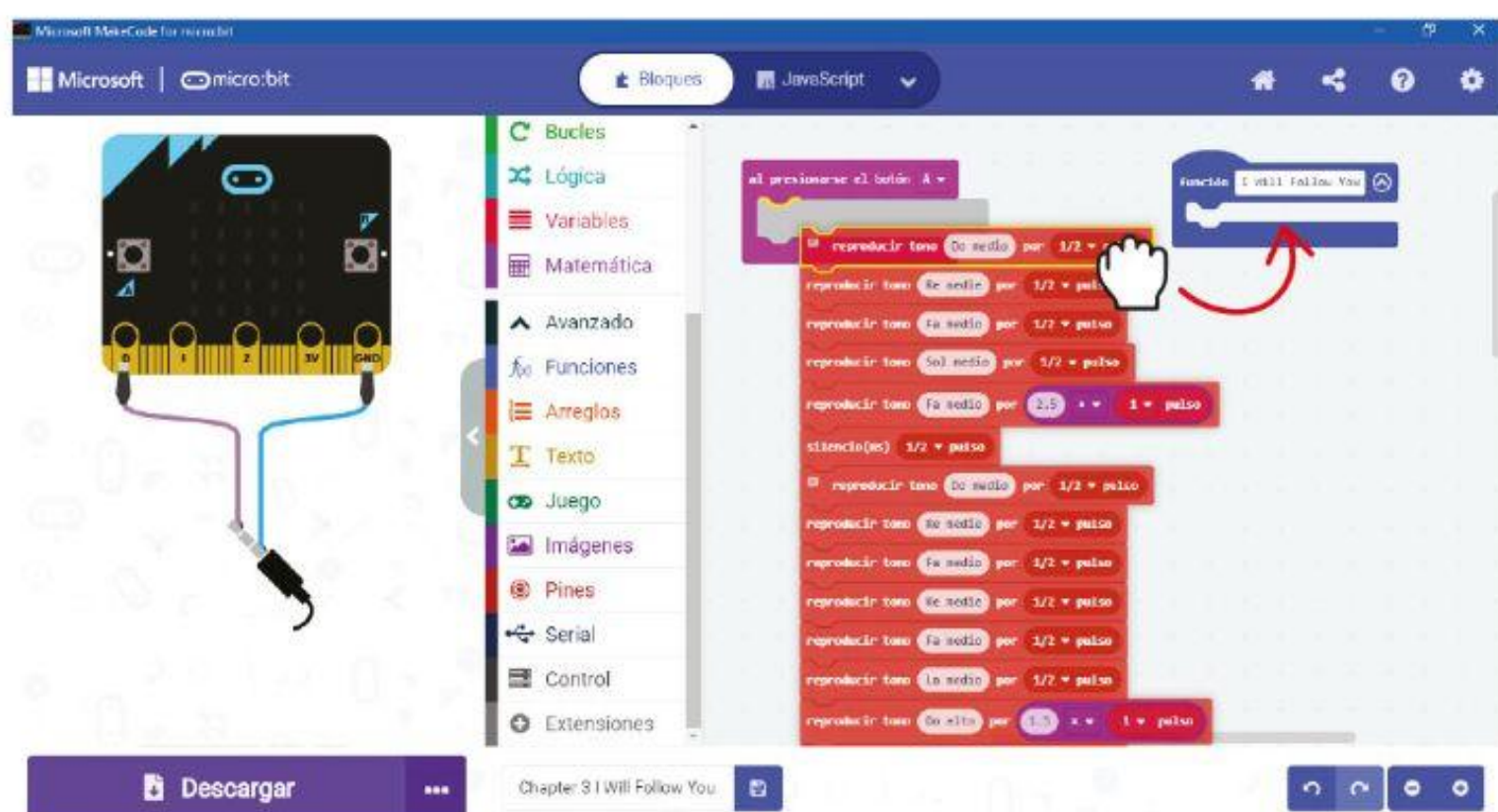
Podemos agrupar bloques de código que realizan una tarea específica en una función, por ejemplo tu código para reproducir la canción 'I Will Follow You'.

Paso 11 Haz clic en la categoría **[Avanzado]** y luego en la categoría **[Funciones]**. Selecciona **[Crear una función...]** y renombra 'doSomething' como **'I Will Follow You'** la ventana emergente. Luego haz clic en **'Listo'**.

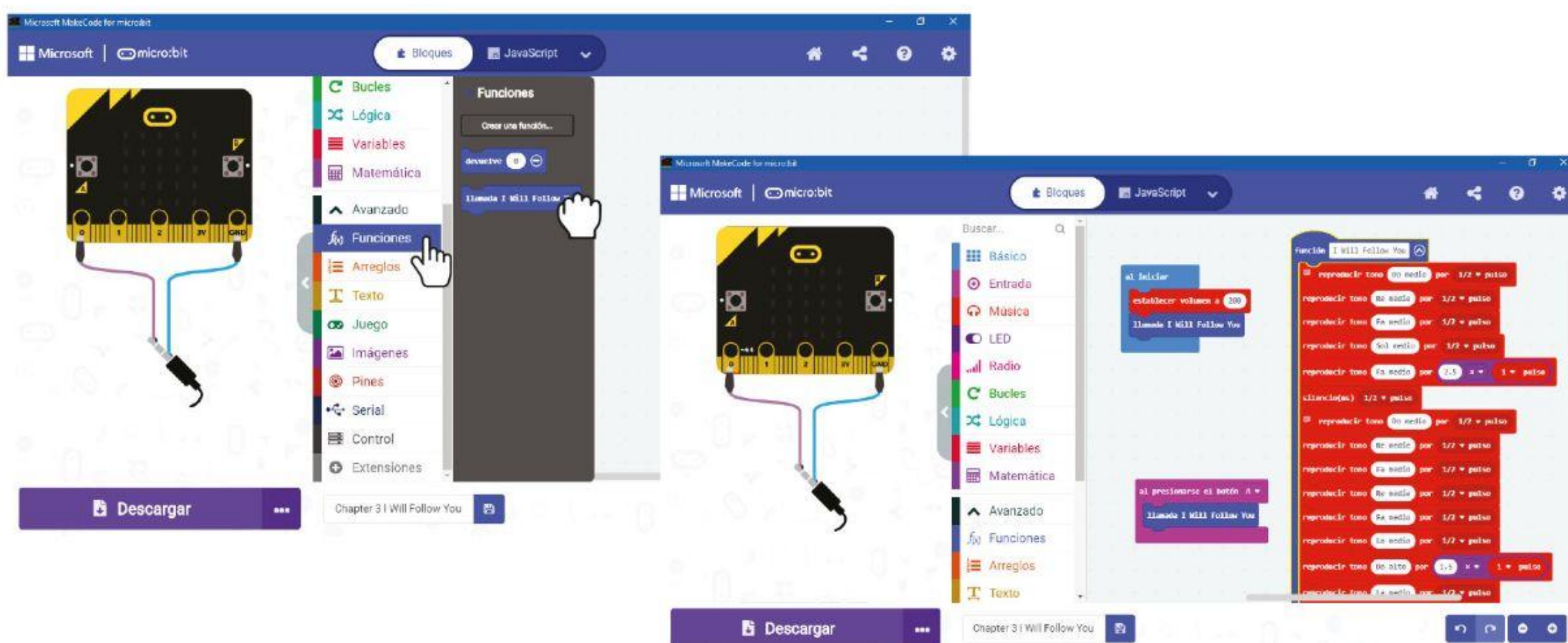


CAPÍTULO 3 : Escuchemos algo de música ~

Paso 12 Un bloque **[función I Will Follow You]** aparecerá en tu Editor. Haz clic en el bloque superior dentro del bloque **[al presionar el botón B]**, manten presionado y arrastra todos los bloques dentro de la ranura **[función I Will Follow You]**.



Paso 13 Haz clic en la categoría función y selecciona el bloque **[llamada I Will Follow You]**. Duplica el bloque **[llamada I Will Follow You]** y encájalo dentro del bloque **[al iniciar]** y en el bloque **[al presionarse el botón B]**. Este es el código de muestra:

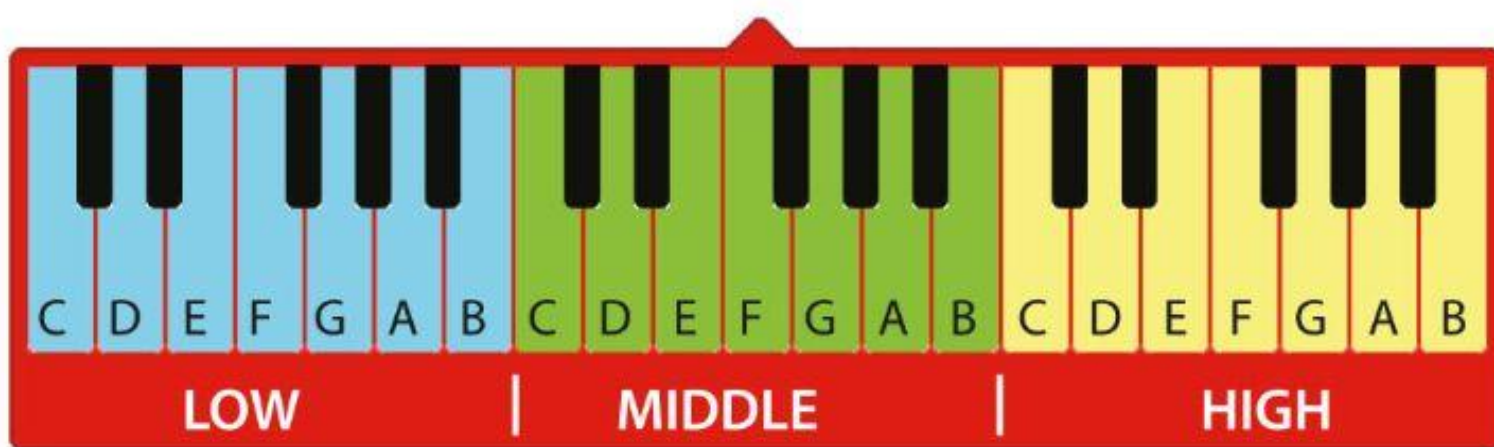


Paso 14 Transfiere el código completo a tu EDU: BIT. Disfruta la música ~

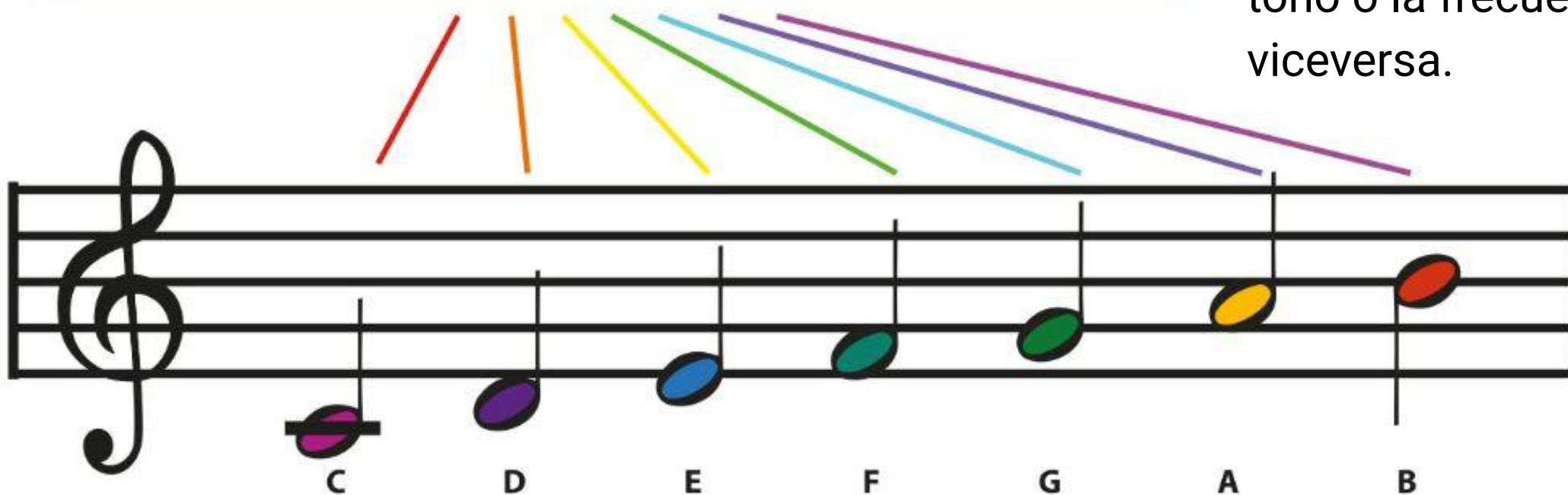
MODIFICAR EL CÓDIGO



Puedes programar EDU: BIT para reproducir otras canciones si sabes leer música. Aquí hay una guía simple para ayudarlo a "descifrar" una partitura musical.



La posición de una nota musical en el pentagrama (es decir, las cinco líneas horizontales) nos dice qué nota tocar. Cuanto más alta sea la posición de la nota en el pentagrama, mayor será el tono o la frecuencia del sonido, y viceversa.



Sign	Rest	Relative Length	Duration
		Whole Note	4 beats
		Half Note	2 beats
	or	Quarter Note	1 beat
		Eighth Note	1/2 beat
		Sixteenth Note	1/4 beat

Se utilizan diferentes notaciones musicales para decirnos cuánto tiempo se debe tocar una nota..

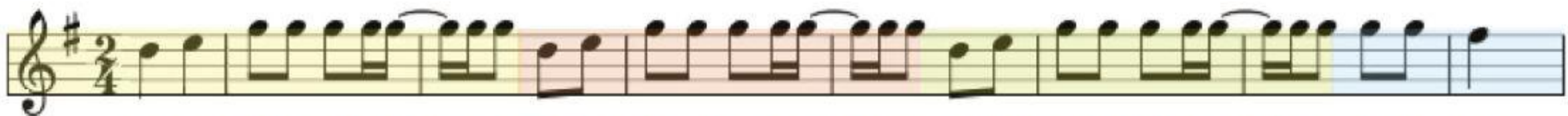
MODIFICAR EL CÓDIGO

Usando las pistas, ¿puedes "descifrar" la siguiente melodía?



Baby Shark

♩ = 115



Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark Doo Doo Doo Doo Doo Doo Ba -by Shark.

Line 1	Ba	-by	Shark	doo	doo,	doo	doo	doo	doo
Note	High D	High E	High G		High G	High G	High G		High G
Beat	1		1/2	1/2	1/2		1/2	1/4	1/2

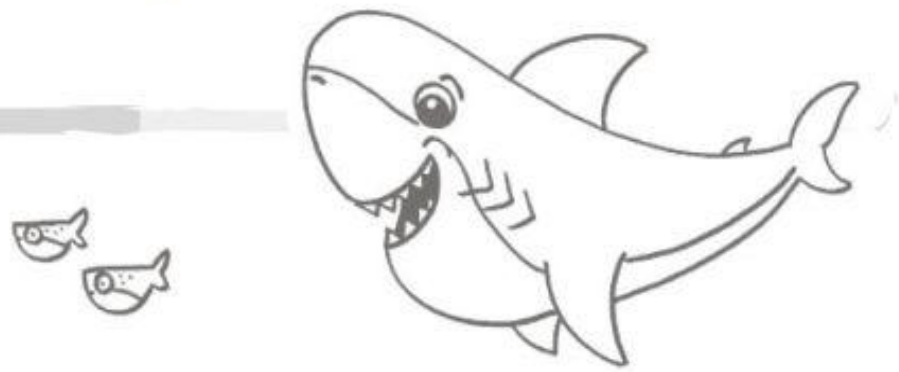
Line 2	Ba	-by	Shark	doo	doo,	doo	doo	doo	doo
Note	High D		High G	High G	High G		High G	High G	High G
Beat	1/2	1/2	1/2		1/2	1/4	1/2		1/2



Repeat the same for Line 3

Line 4	Ba	-by	Shark
Note		High G	High F#
Beat	1/2	1/2	

Program EDU:BIT to play Baby Shark tune when the yellow button (Button A) and blue button (Button B) are pressed at the same time.



NOTA!

Utiliza el bloque [establecer volumen a _] para ajustar el volumen de la melodía.



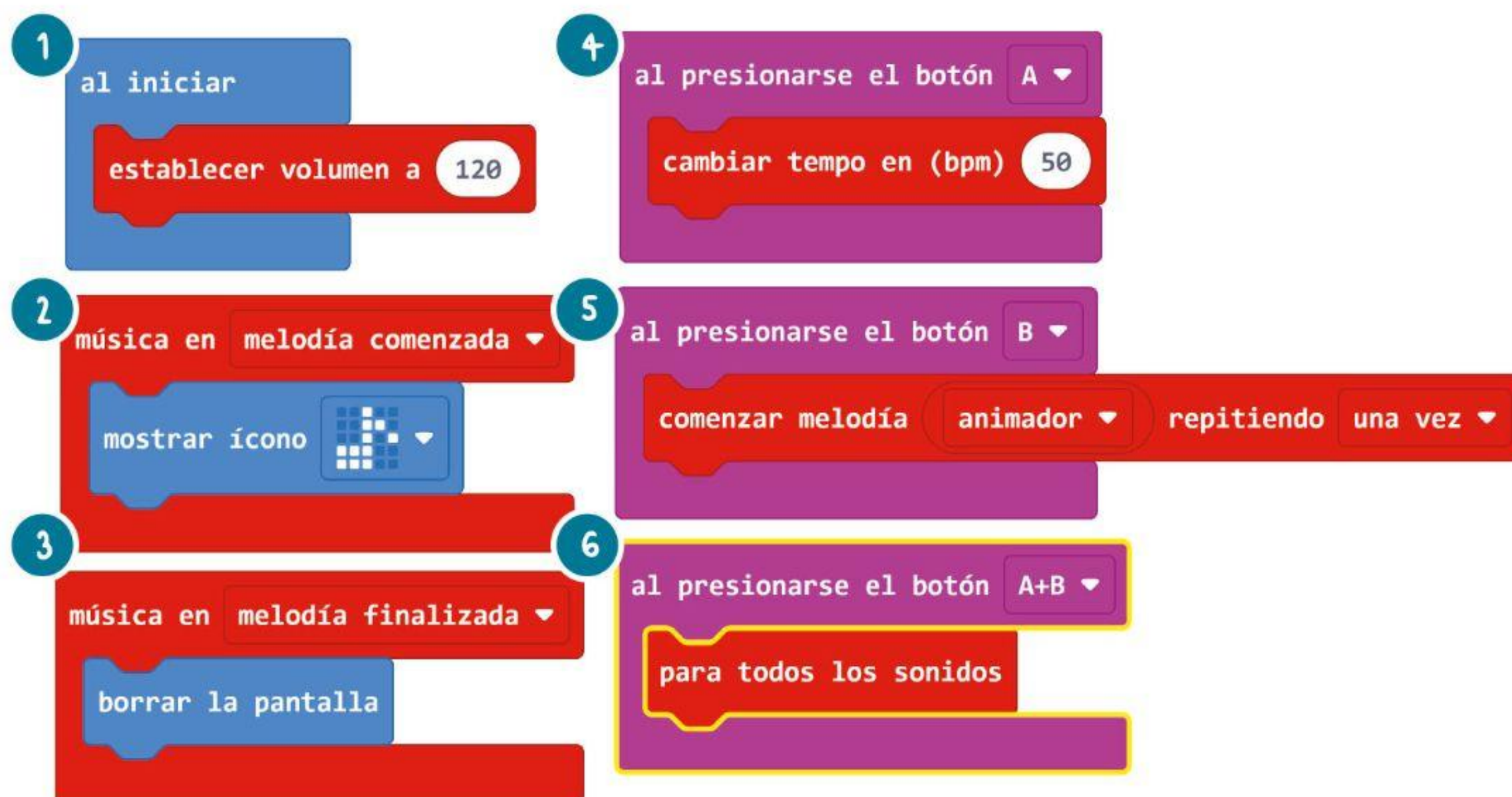
EXPLORA OTROS BLOQUES

1 Puedes establecer el "ritmo" (es decir, la velocidad de tu canción) usando el bloque **[cambiar tempo en (bpm)]**. Cuanto mayor sea el valor de bpm (pulsaciones por minuto), más rápida o más lenta será la melodía. Usa el bloque **[cambiar tempo en (bpm)]** para cambiar el tempo.

#2 Usa el bloque **[stop melody]** para detener una melodía que se está reproduciendo.

#3 3 También puedes usar el bloque **[música en_]** y sus condiciones, como melodía iniciada o finalizada, para desencadenar eventos en su código

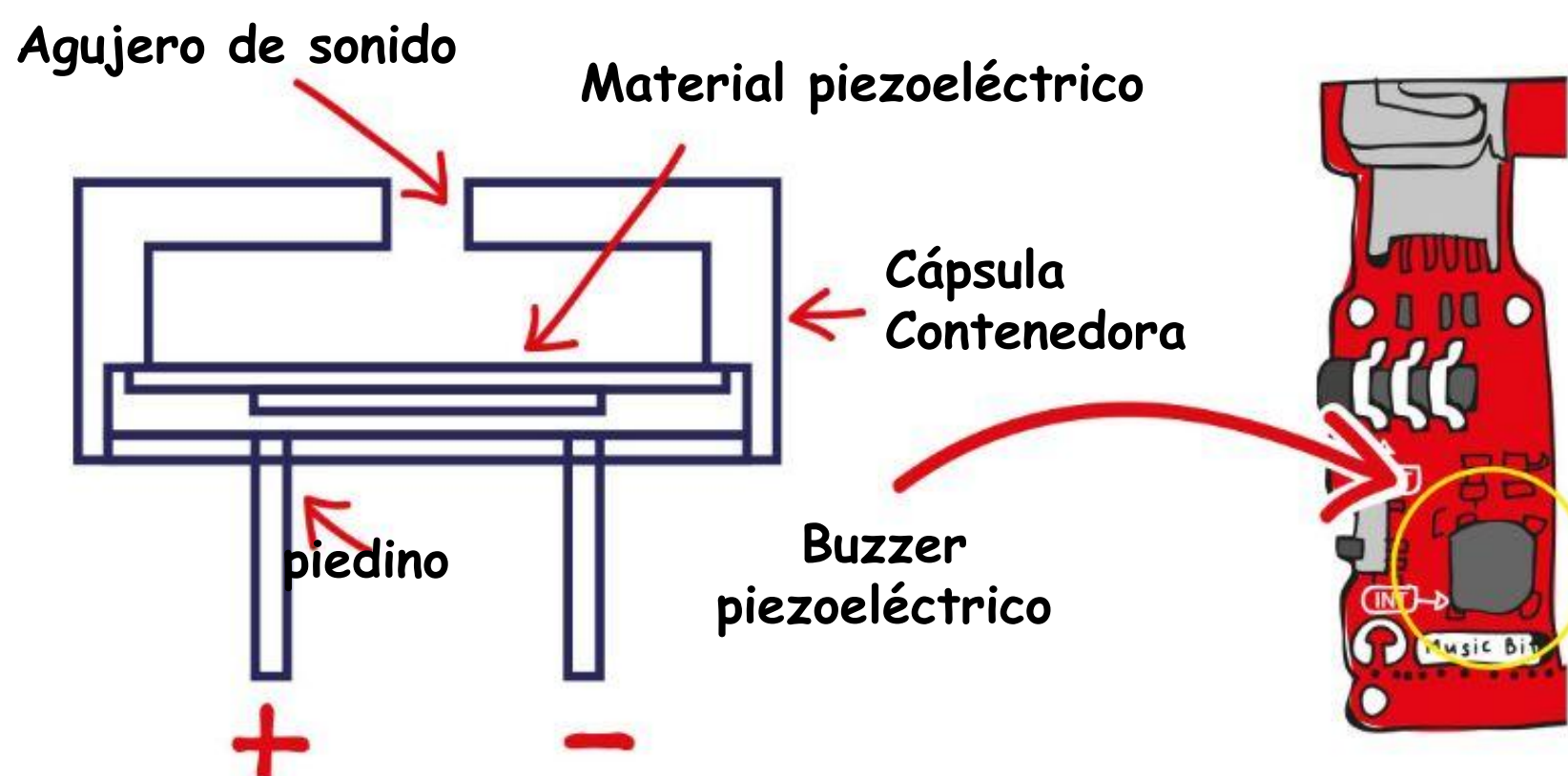
Aquí hay un código de muestra



- 1 En este programa, el tempo inicial es establecido a 120 bpm.
- 2 Al comienzo de la melodía, la matriz LED mostrará el ícono de la nota musical.
- 3 Al final de la melodía, todos los LED de la matriz de visualización se apagarán.
- 4 Cada vez que se presiona el botón A, el tempo aumenta en 50 bpm.
- 5 La melodía del 'animador' se reproducirá cada vez que se presione el botón B.
- 6 La melodía se detiene cuando los botones A + B se presionan juntos.

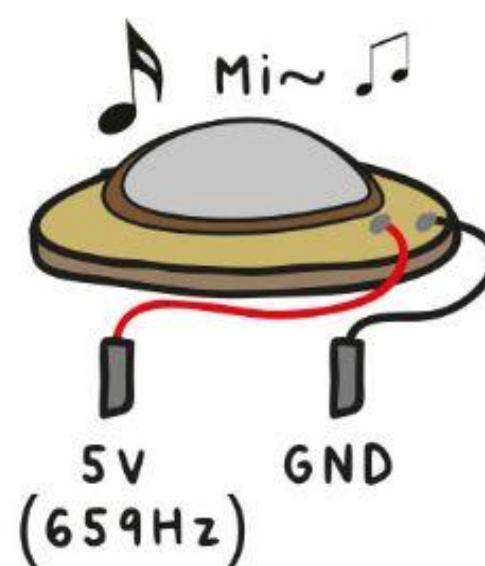
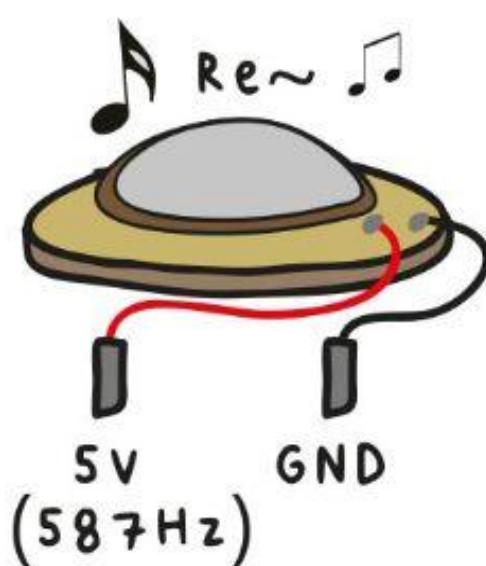
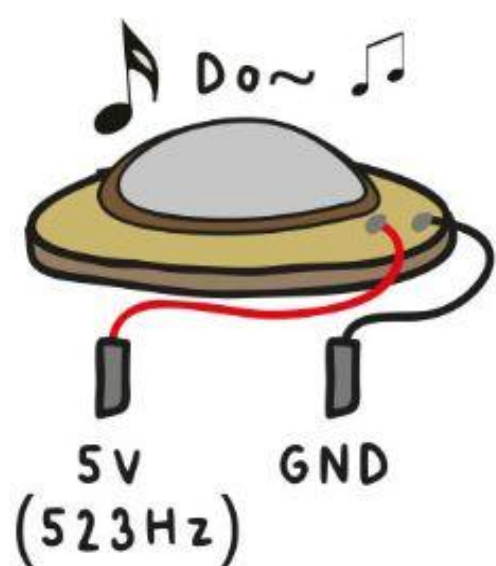
COSAS DIVERTIDAS

Un **zumbador piezoeléctrico** se usa comúnmente para producir sonido al hacer vibrar un material piezoeléctrico a través de una señal eléctrica que lo atraviesa.



Al cambiar la frecuencia de la señal eléctrica, cambia la velocidad de las vibraciones; y por lo tanto el zumbador piezoeléctrico produce sonidos de diferentes tonos.

ELEMENTO PIEZOELÉCTRICO



El oído humano puede escuchar sonidos en el rango de frecuencia entre 20 Hz y 20 000 Hz. Cualquier sonido por debajo de 20 Hz se denomina infrasónico, mientras que por encima de 20 000 Hz se denomina ultrasónico.

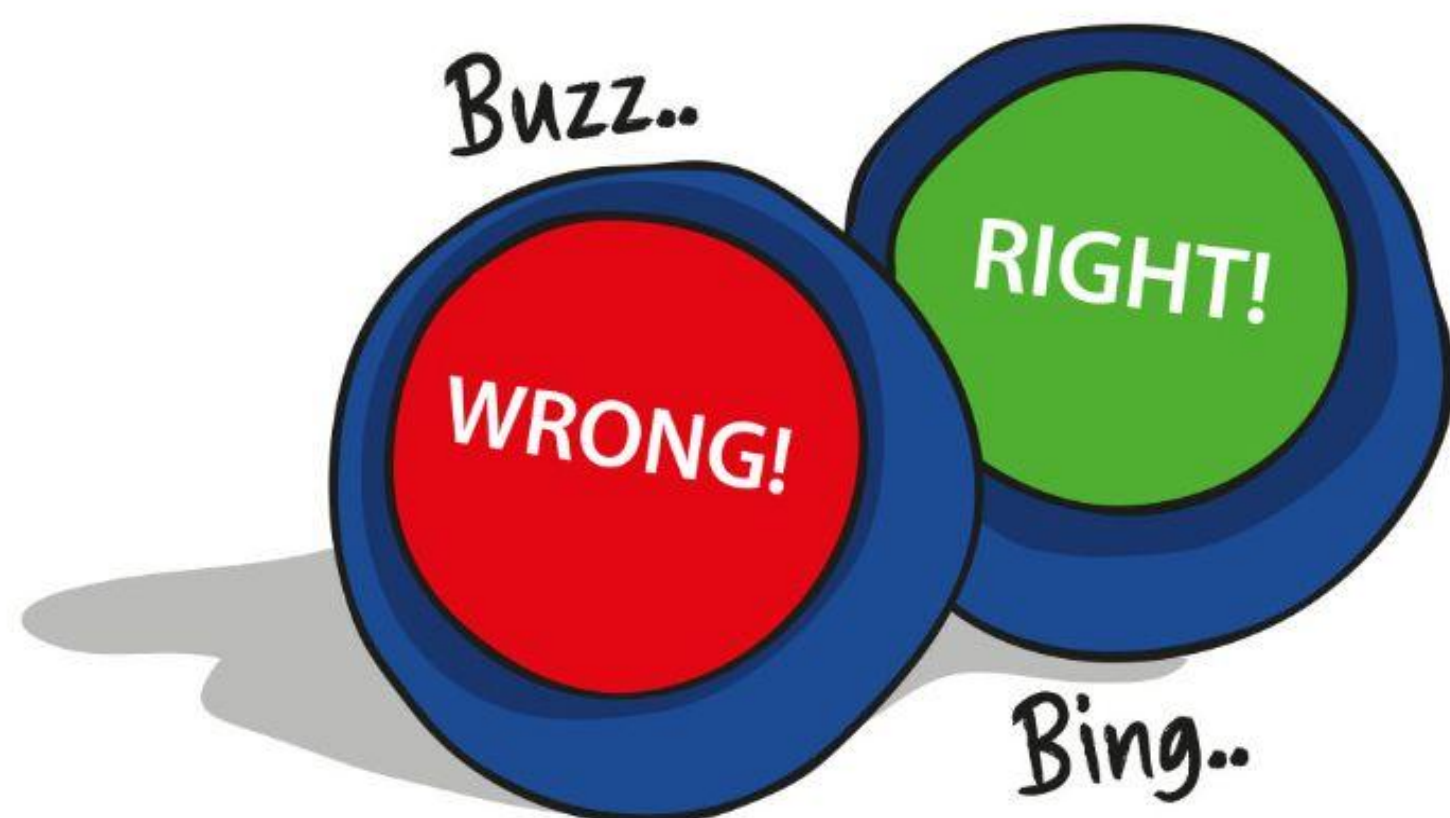


¡Aprende más!



youtu.be/cxfPNc4Wefo

DESAFÍO DE LA APLICACIÓN



Programa EDU:BIT para que funcione como indicador sonoro de un juego de preguntas, para indicar respuestas correctas o incorrectas.	
En el inicio	Mostrar una cara sonriente.
Cuando presionas el botón A (amarillo)	Muestra el icono ✓ y toca la melodía de "encendido" una vez.
Cuando presionas el botón B (azul)	Muestra el ícono □ y toca la melodía "wawawawa" una vez.
Cuando presionas los botones A + B	Limpia la pantalla.

CAPÍTULO 4

Victoria, Derrota o Empate ~

Semáforo de Edu:bit



¡Pavo!



¡Escanéame!

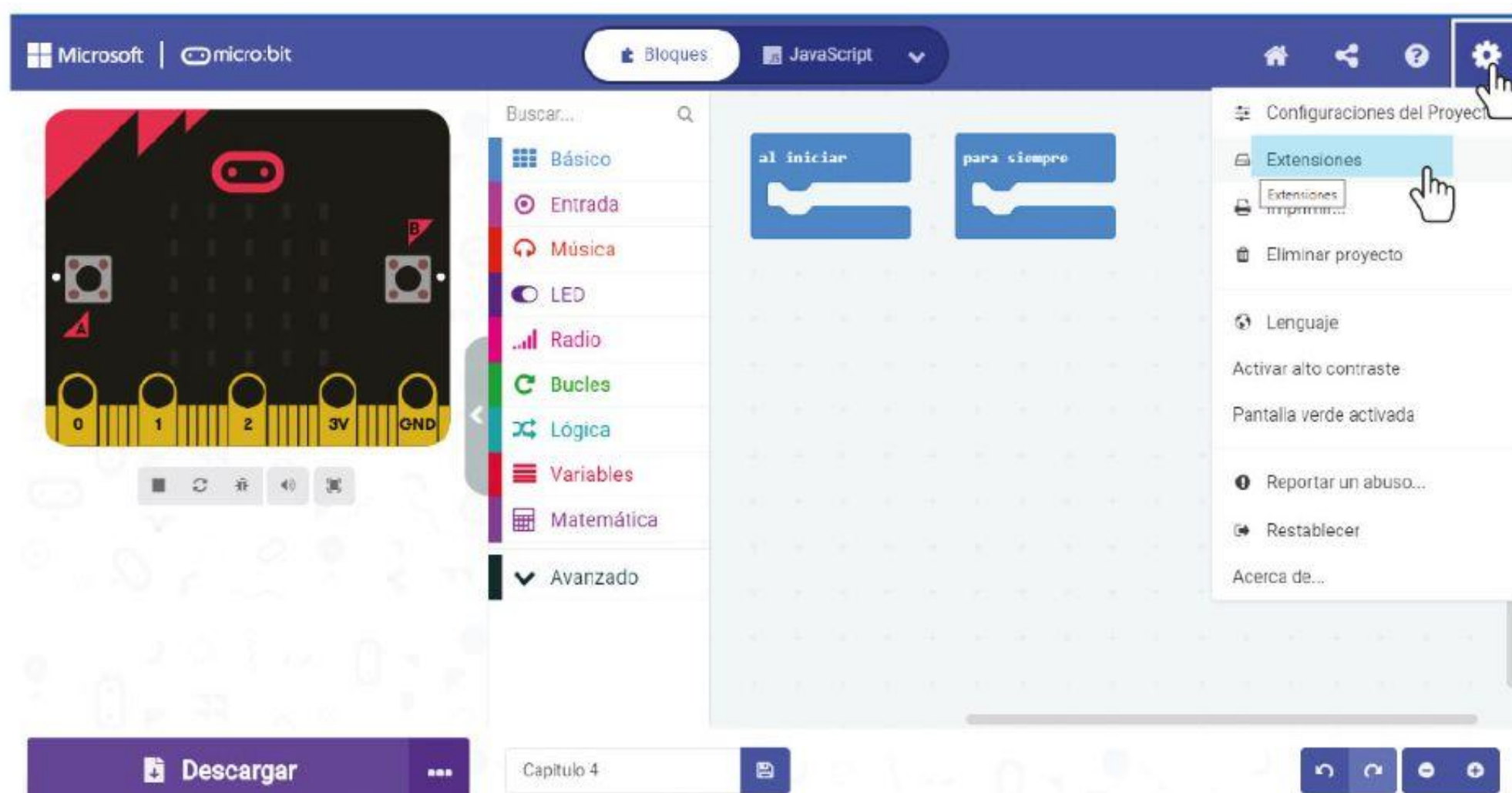


¿Ha notado un grupo de LED de color rojo, amarillo y verde en su EDU: BIT? Es el Bit Semáforo. Para programarlo, debe agregar la extensión EDU: BIT a su Editor MakeCode. Las extensiones son un conjunto de bloques personalizado que se agregan al editor para permitirnos programar fácilmente accesorios de la micro:bit, como nuestra placa EDU:BIT.



PROGRAMAMOS!

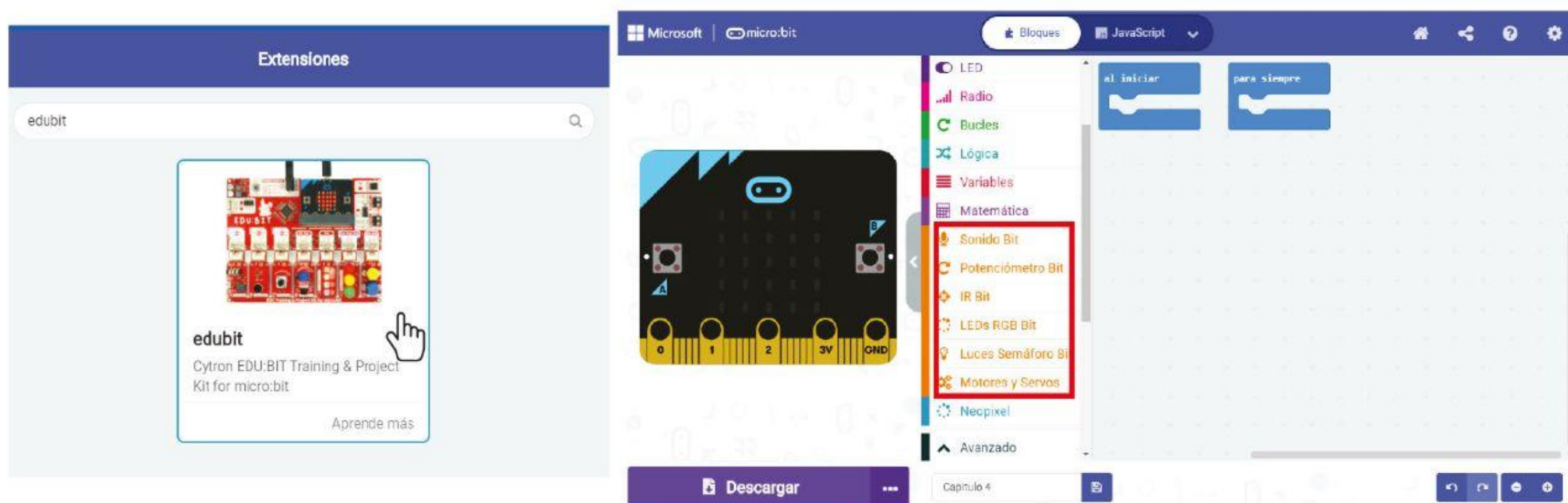
Paso 1 Crea un nuevo proyecto en el Editor de MakeCode. Haz clic en el icono de engranaje y selecciona 'Extensiones'. * Necesitas una conexión a internet para agregar extensiones.



Paso 2 Escribe "edubit" en el cuadro de búsqueda y presione Entrar..

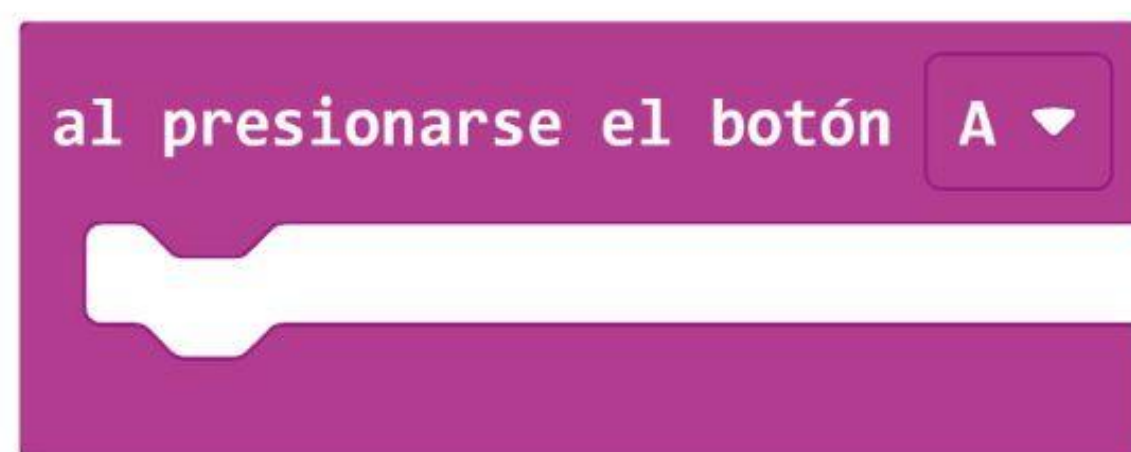


Paso 3 Haz clic en la extensión 'edubit'. Espera a que se cargue y luego notarás la siguiente categoría nueva en su Editor de MakeCode.

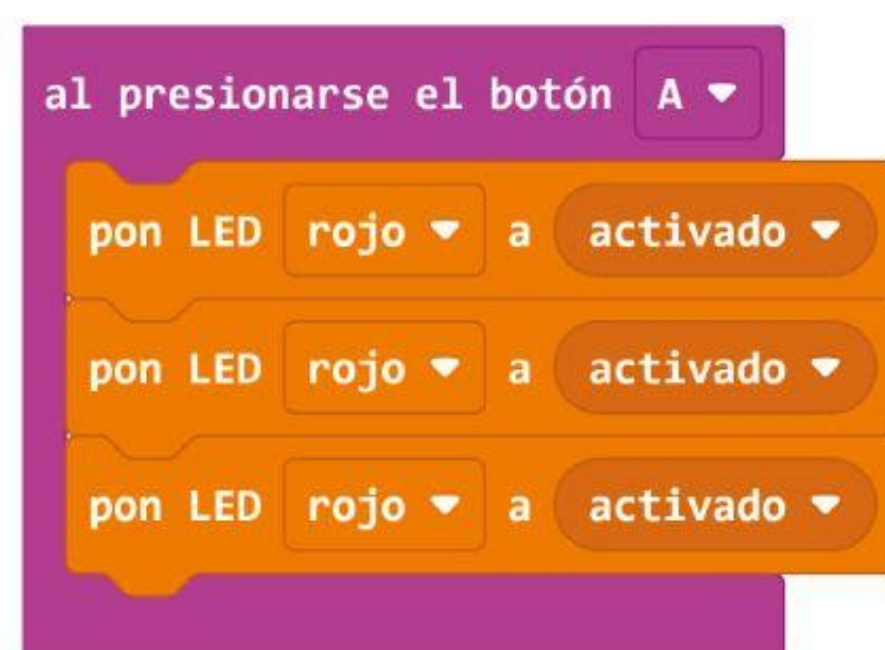
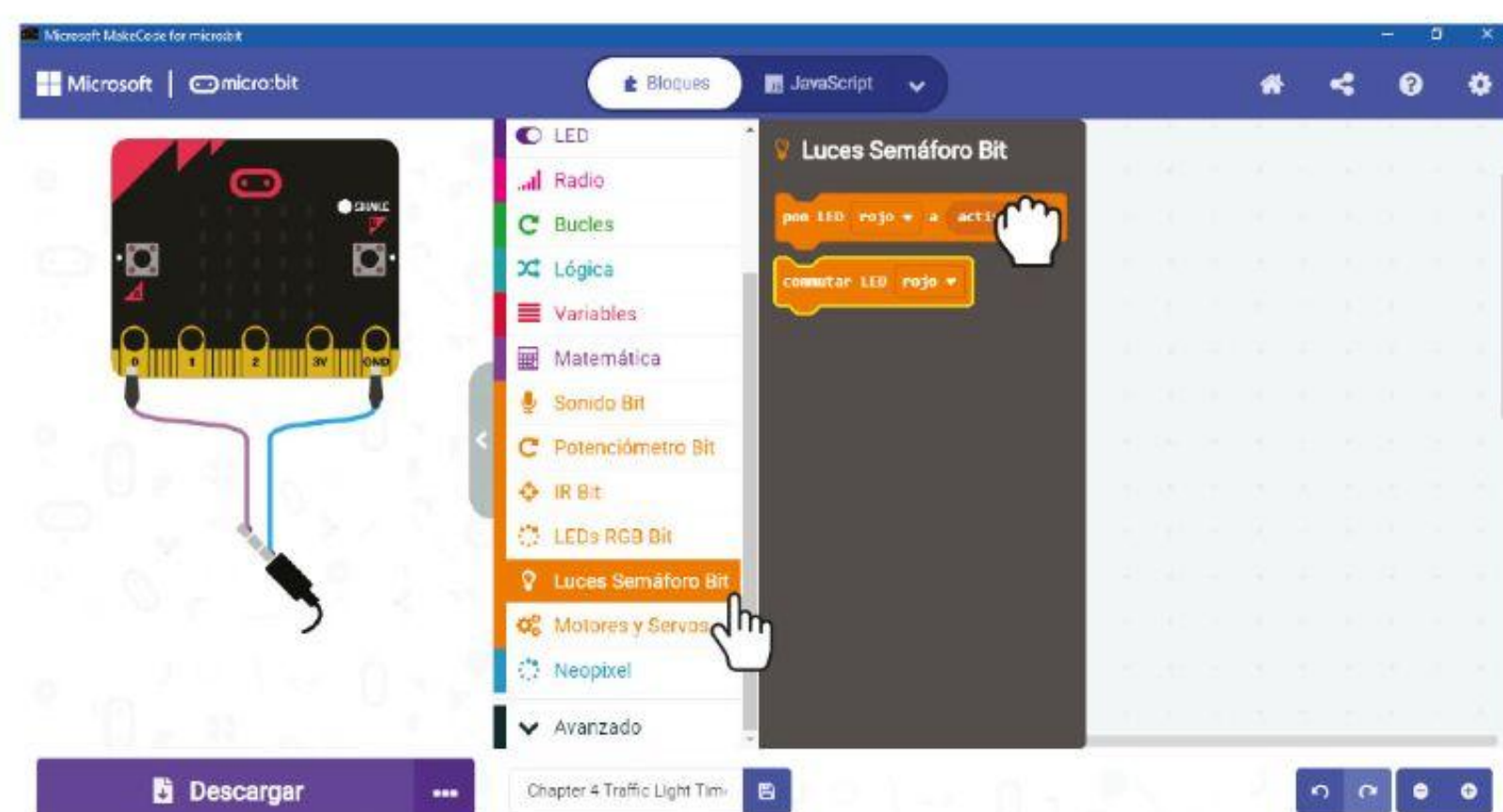


CAPÍTULO 4 : Victoria, Derrota, Empate ~

Paso 4 Haz clic en la categoría **[Entrada]** y selecciona el bloque **[al presionarse el botón _]**.



Paso 5 Haz clic en la categoría **[Luces Semáforo Bit]** y selecciona el bloque **[pon LED _ a _]**. En el espacio de trabajo, haz clic derecho en el bloque **[pon LED _ a _]** y luego haz clic en 'Duplicar'. Repite hasta que tengas 3 bloques. **[pon LED _ a _]**. Coloca los bloques dentro del bloque **[al presionarse el botón _]**.

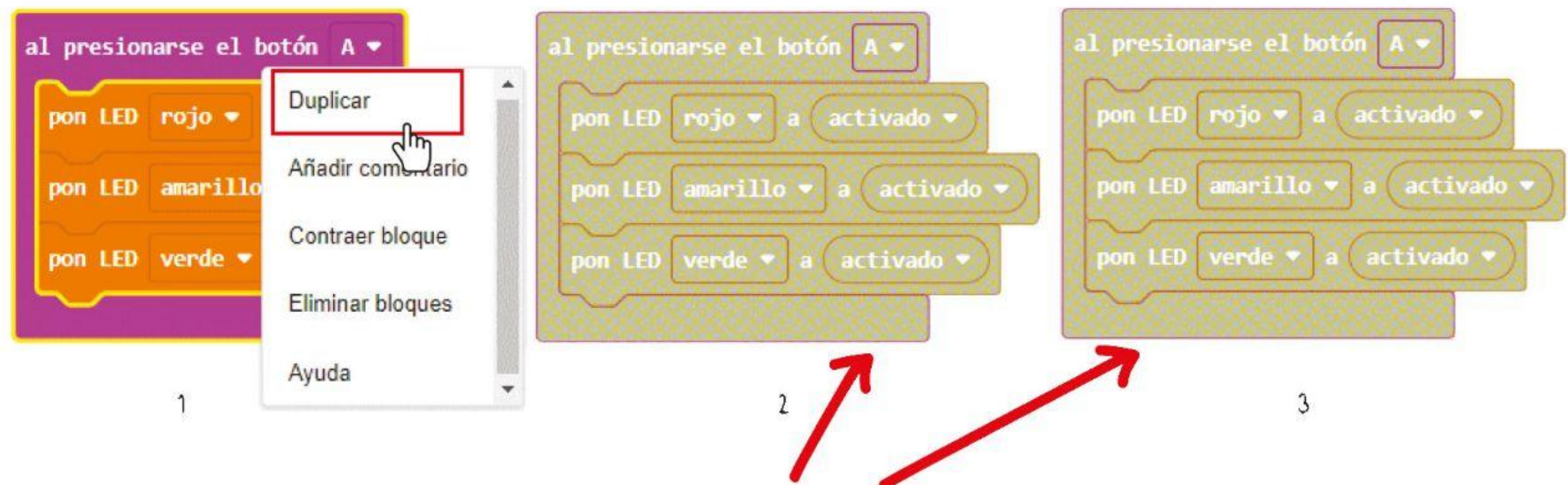


Paso 6 Haz clic en la selección de color y cámbiala en el segundo y tercer bloque para **'amarillo'** y **'verde'** respectivamente.



CAPÍTULO 4 : Victoria, Derrota, Empate ~

Paso 7 Haz clic derecho en el bloque **[al presionarse el botón _]** y selecciona 'Duplicar'. Repite para obtener tres grupos con los mismos bloques.



*Estos bloques están deshabilitados y no se ejecutarán porque hay más de un bloque **[al presionarse el botón A]**.

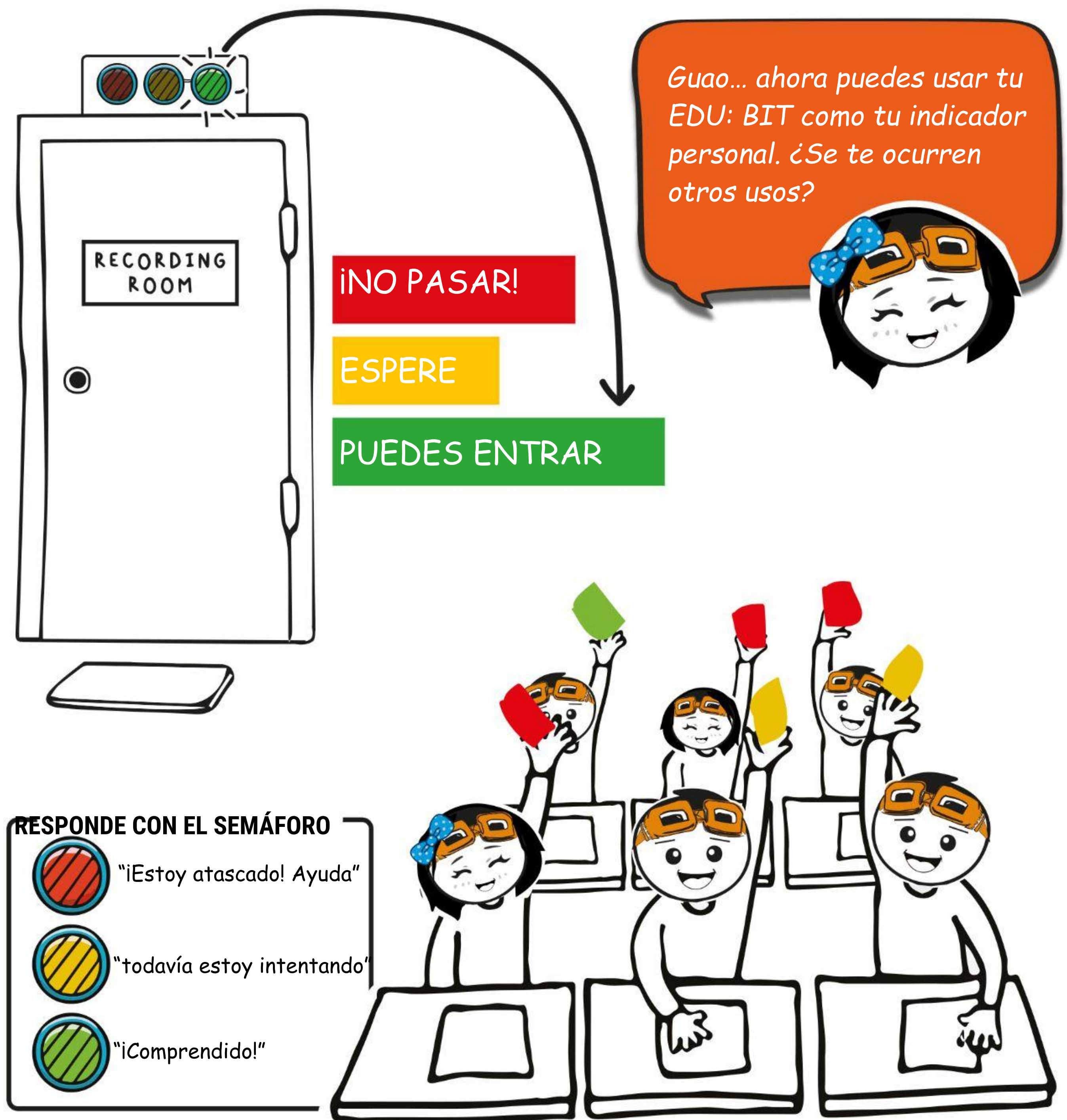
Paso 8 Cambia "A" en el segundo y tercer bloque **[al presionarse el botón _]** por "B" y "A+B" respectivamente.

Paso 9 Cambia el estado de los LED de ENCENDIDO a APAGADO de la siguiente manera.



Paso 10 Transfiere el código a su EDU: BIT y observa qué sucede cuando presionas el Botón A, el Botón B y luego ambos botones al mismo tiempo.

CAPÍTULO 4 : Victoria, Derrota, Empate ~



El LED, o diodo emisor de luz, es un ejemplo de un dispositivo de salida digital. Solo tiene dos estados posibles: ENCENDIDO o APAGADO; entonces ON se representa comúnmente por 1 (uno) y OFF por 0 (cero).



También puedes programar tu EDU:BIT para que funcione como temporizador. Aquí está el código de ejemplo.



El temporizador comienza cuando agitas EDU: BIT

Sonidos para señalar el inicio del temporizador.

LED verde encendido.

LED amarillo encendido.

LED rojo encendido.

Suena wawawawaaa para indicar el fin de los tiempos.

Cambia el LED rojo 10 veces

```

si agitado
  reproducir tono Do medio por 1 pulso
  pon LED rojo a desactivado
  pon LED amarillo a desactivado
  pon LED verde a activado
  pausa (ms) 2000
  pon LED rojo a desactivado
  pon LED amarillo a activado
  pon LED verde a desactivado
  pausa (ms) 2000
  pon LED rojo a activado
  pon LED amarillo a desactivado
  pon LED verde a desactivado
  pausa (ms) 2000
  comenzar melodía wawawawaa repitiendo una vez
  repetir 4 veces
    ejecutar
      conmutar LED rojo
      pausa (ms) 500
  
```

En este código de ejemplo, cada LED se enciende durante 2000 ms (2 segundos).

Si quisieras encender cada LED durante 1 minuto, ¿qué valor debes ingresar aquí?

Aquí hay un consejo para ti:
1 minuto = 60 segundos

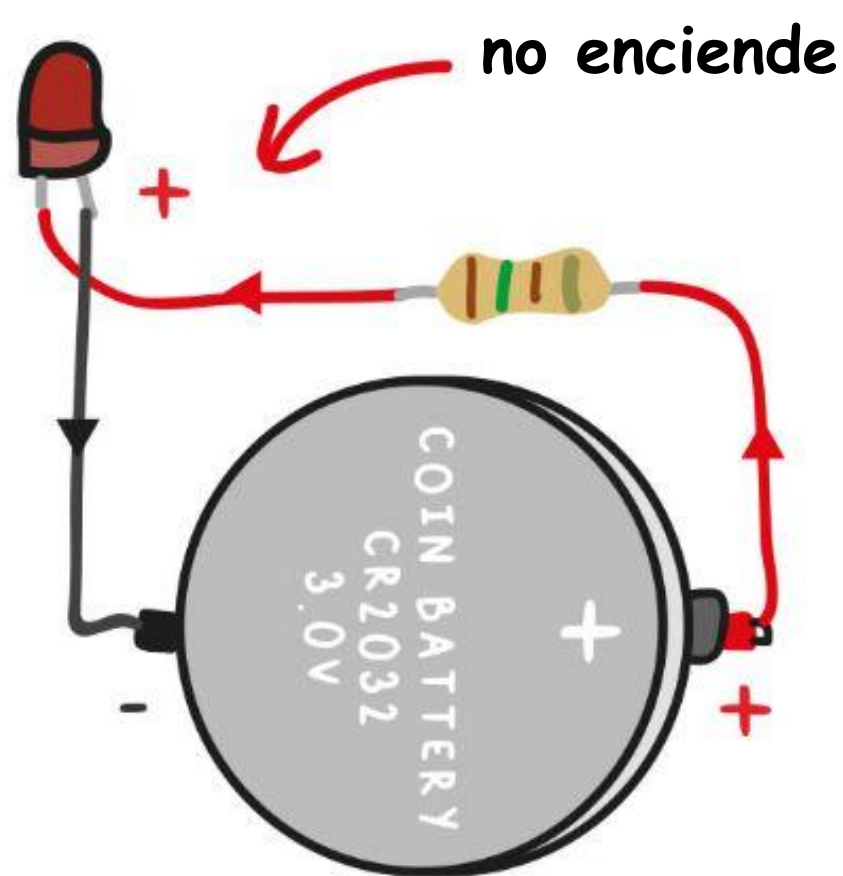
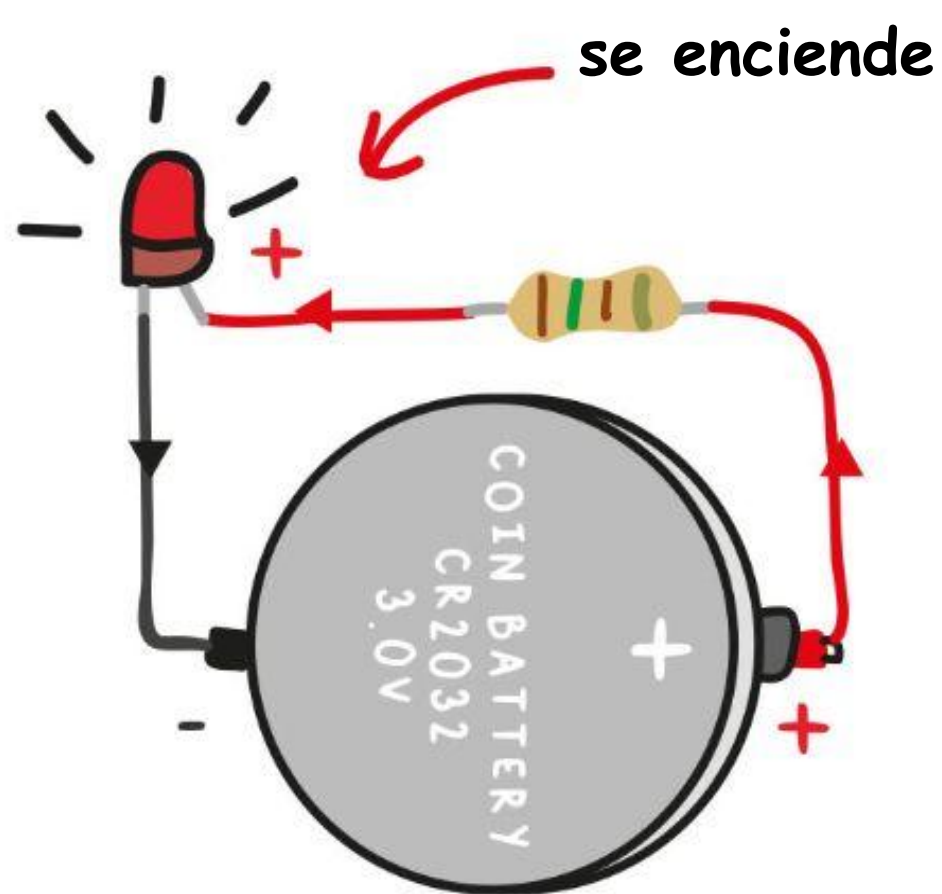
pon LED rojo a activado



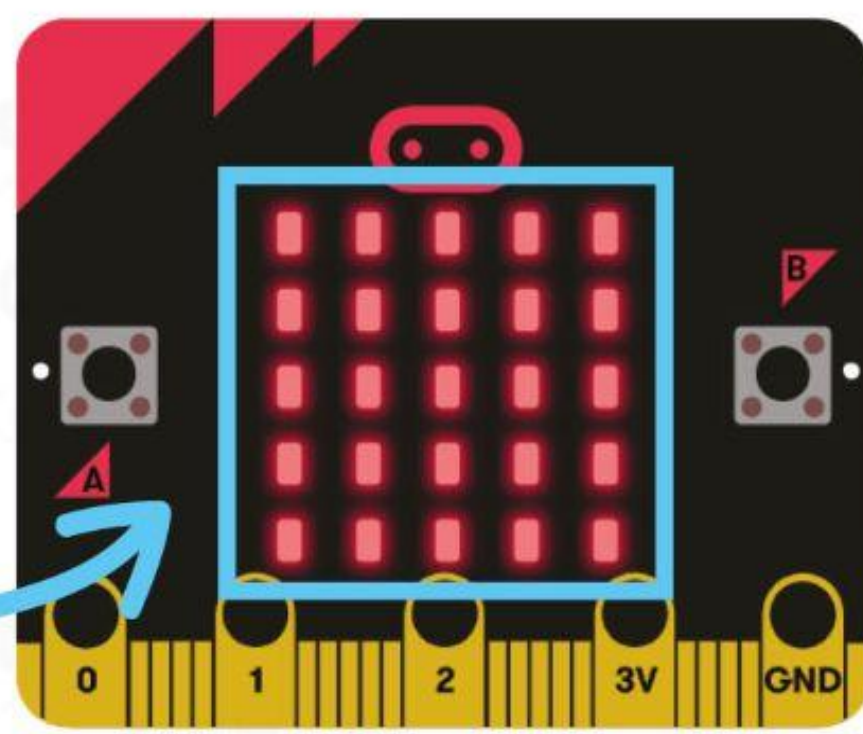
"Alternar" significa cambiar de un estado a otro. Si el estado actual es ON, cambiará a OFF; y viceversa. Entonces, cuando cambiamos el estado del LED repetidamente, el LED parpadeará.

¡COSAS DIVERTIDAS!

Un **diodo emisor de luz (LED)** es un dispositivo semiconductor que produce luz a partir de la electricidad. Tiene 2 terminales, positivo y negativo. Cuando un LED está conectado con la polaridad correcta y la corriente fluye a través de él, emite luz.



Los LEDs utilizados en micro: bit son los de la tecnología de montaje superficial (SMT), que permite que sean muy pequeños.



Además de los del micro: bit, hay otros 41 LED SMT en la placa EDU: BIT. ¿Puedes identificarlos a todos?



¡Aprende más!



youtu.be/qqBmvHD5bCw

DESAFÍO DE LA APLICACIÓN

Programa EDU:BIT para que funcione como contador de puntos y también como cronómetro para un juego como "Gana, pierde o empata" y "Charada".	
En el inicio	Establecer variable Equipo A = 0 Establecer variable Equipo B = 0
Cuando presionas el botón A (amarillo)	Cambia el Equipo A por 1. Muestra la puntuación del equipo A.
Cuando presionas el botón B (azul)	Cambia el Equipo B por 1. Muestra la puntuación del equipo B.
Cuando presionas los botones A + B	Desplácese por las puntuaciones del Equipo A y el Equipo B.
Cuando sacudes	Inicie un temporizador durante 1 minuto encendiendo el LED verde (durante 30 segundos), luego el LED amarillo (durante 20 segundos) y finalmente el LED rojo (durante 10 segundos). Toca la melodía "wawawawaa" cuando se acabe el tiempo. Parpadee el LED rojo 10 veces.

Aquí hay un consejo para ti. Necesitarás dos variables llamadas Equipo A y Equipo B respectivamente.



Vamos a jugar

Win, Lose or Draw~



CÓMO JUGAR:

- Divide la clase en 2 equipos: Equipo A y Equipo B.
- Un miembro del Equipo A comenzará eligiendo una carta al azar. Después de leer en silencio la palabra en la tarjeta, agite EDU: BIT para iniciar el temporizador (1 minuto).
- Luego puede empezar a dibujar en la pizarra para que los miembros de su equipo adivinen la palabra. ¡No está permitido hablar ni gesticular!
- El equipo A anota un punto (presione el botón A o el botón amarillo) si un miembro del equipo adivina la palabra antes de que se acabe el tiempo.
- Si el equipo A comete un error, el equipo B puede intentar robar un punto respondiendo.
- Ambos equipos se turnan para adivinar hasta el final del juego.
- ¡El equipo con la mayor cantidad de puntos gana!

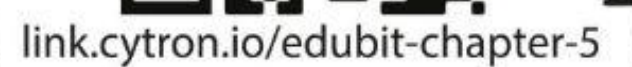
NOTA!

Escanea aquí



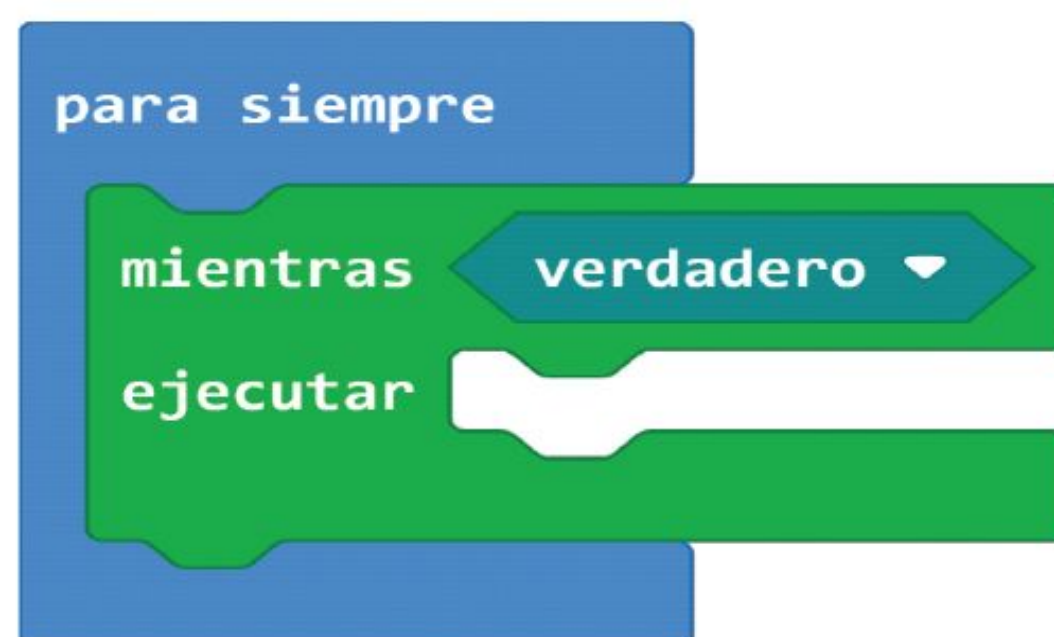
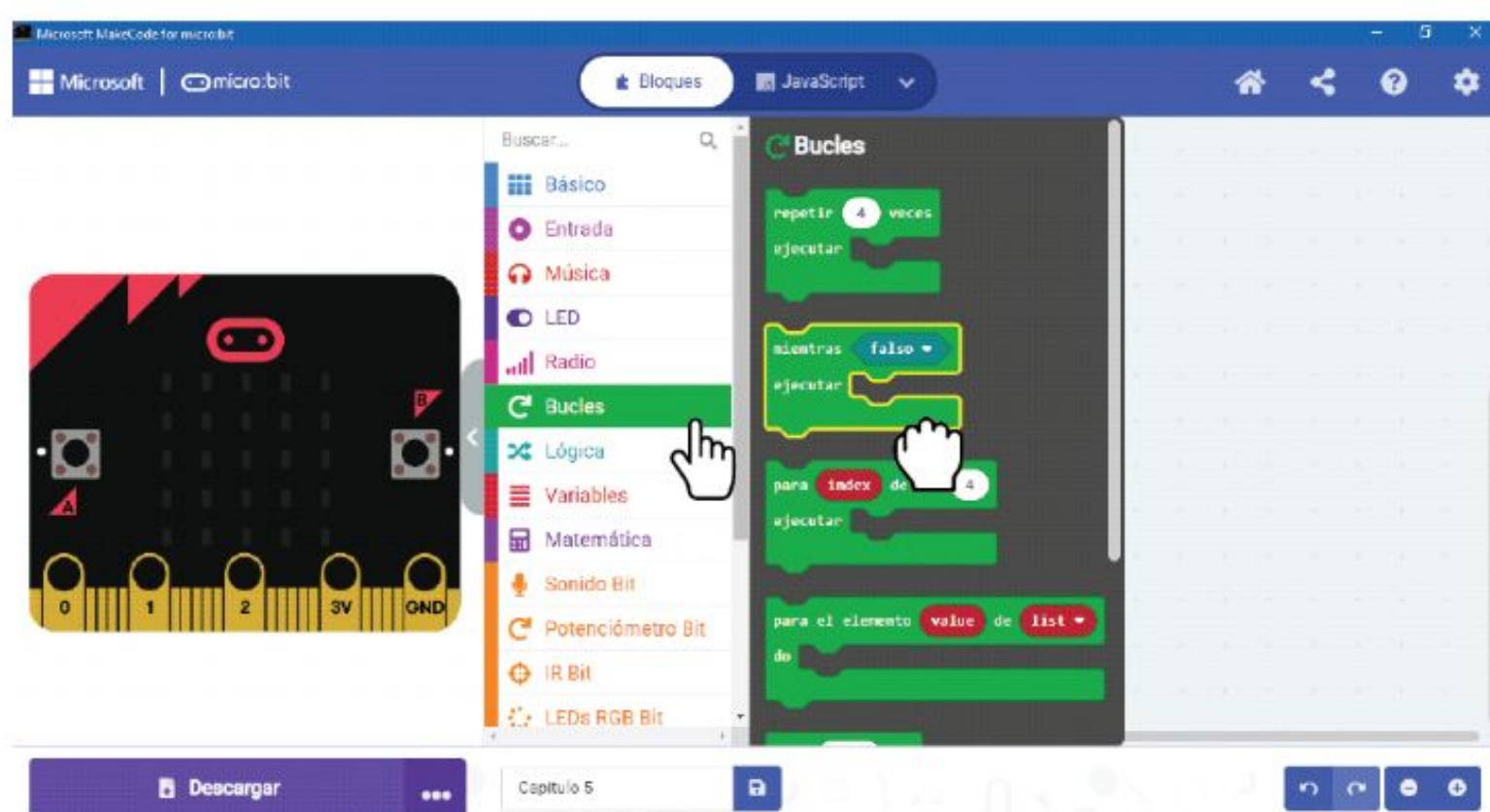
Escanea aquí para descargar tarjetas imprimibles con palabras. Si no eres bueno dibujando, puedes probar el juego Charades. Se aplican las mismas reglas, pero en lugar de dibujar, puede usar gestos. ¡Que lo pases bien!

Bit infrarojo

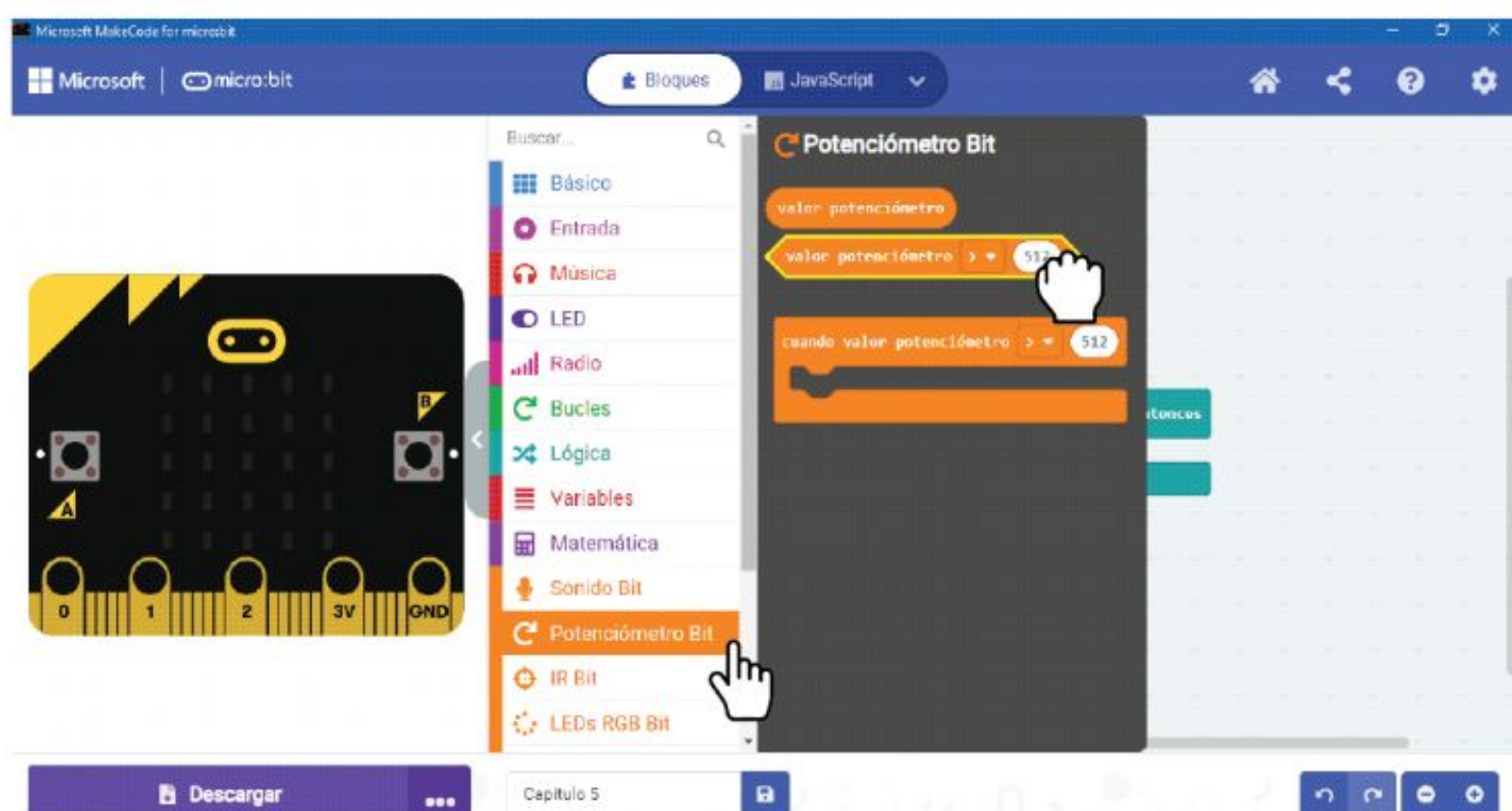


PROGRAMAMOS!

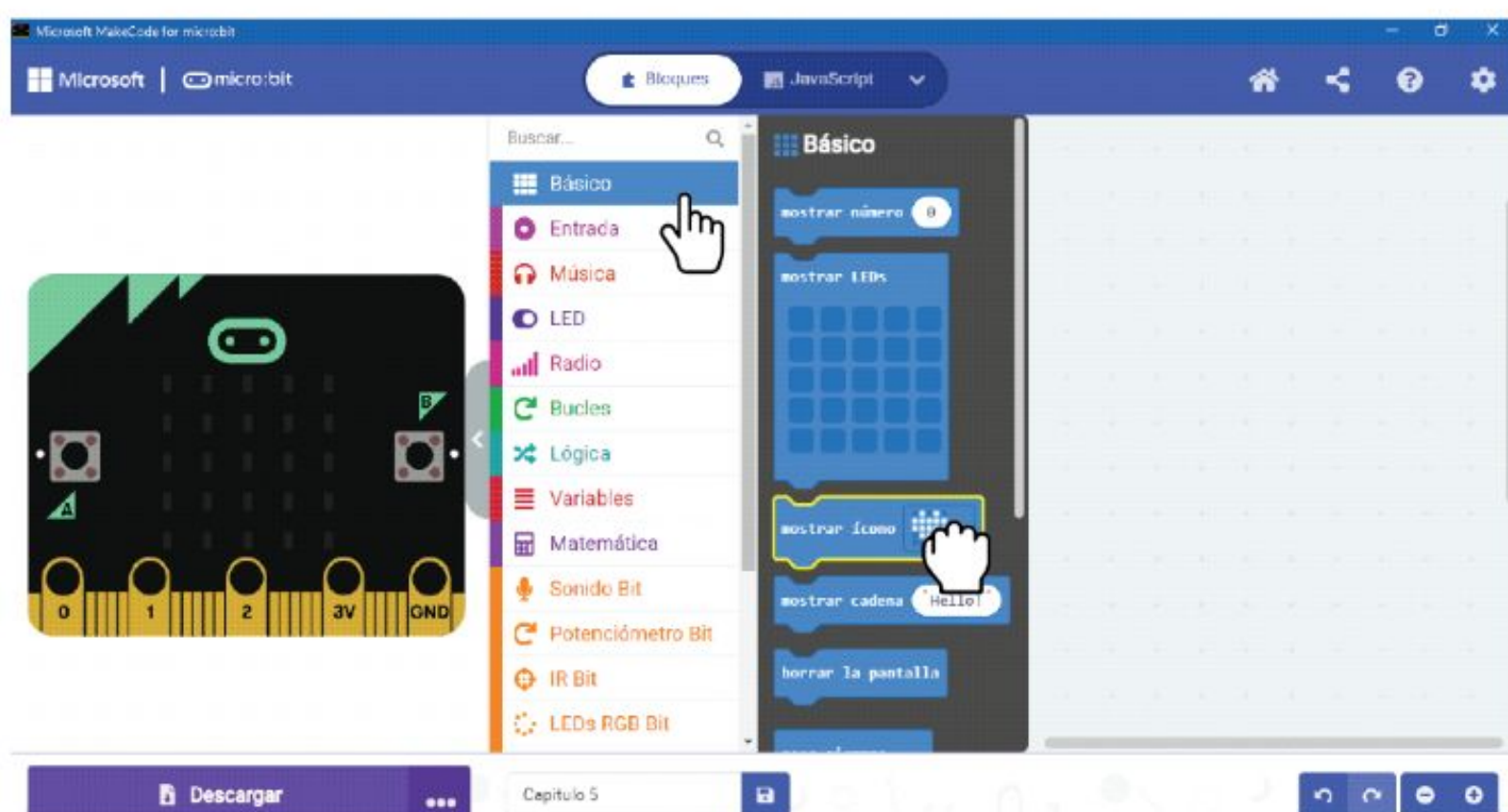
Paso 1 Crea un nuevo proyecto en el Editor MakeCode y agrega la extensión EDU: BIT (puedes consultar la página 40). Haz clic en la categoría **[Bucles]** y luego selecciona el bloque **[mientras_ejecutar]**. Encaja el bloque en la ranura **[para_siempre]**.



Paso 2 Haz clic en la categoría **[IR Bit]** y luego selecciona el bloque **[IR activado]**. Encaja el bloque en la ranura de la condición relativa al bloque **[mientras_ejecutar]**.



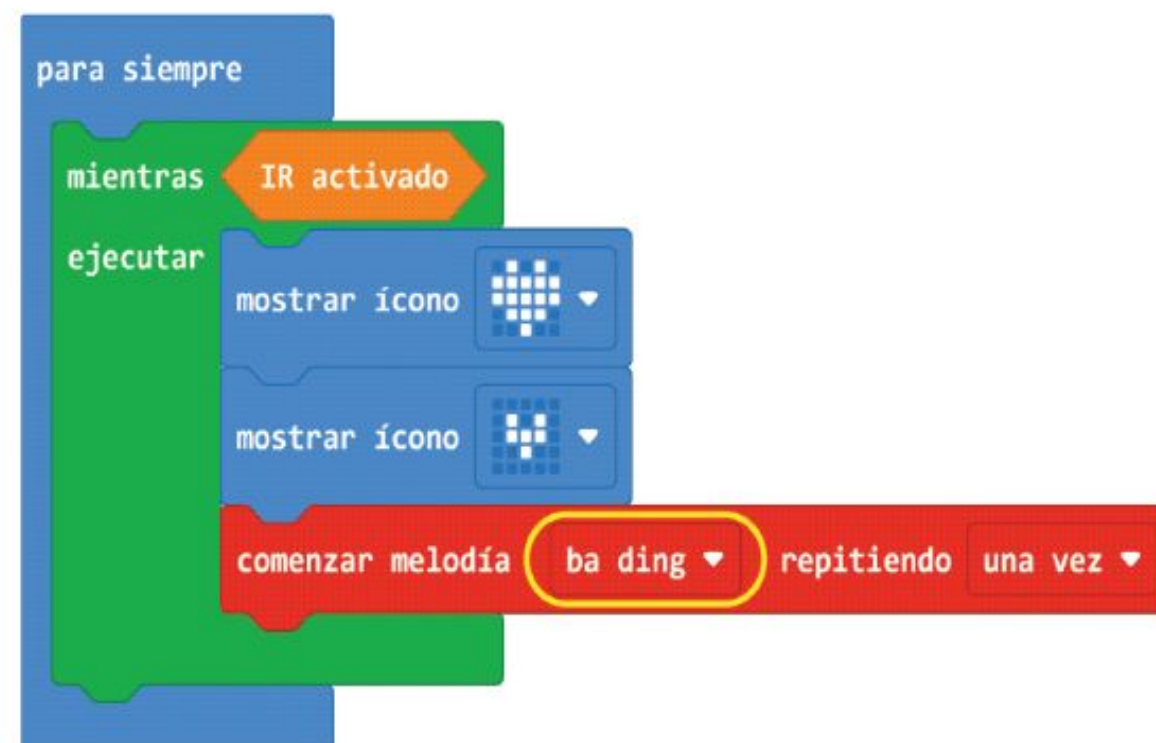
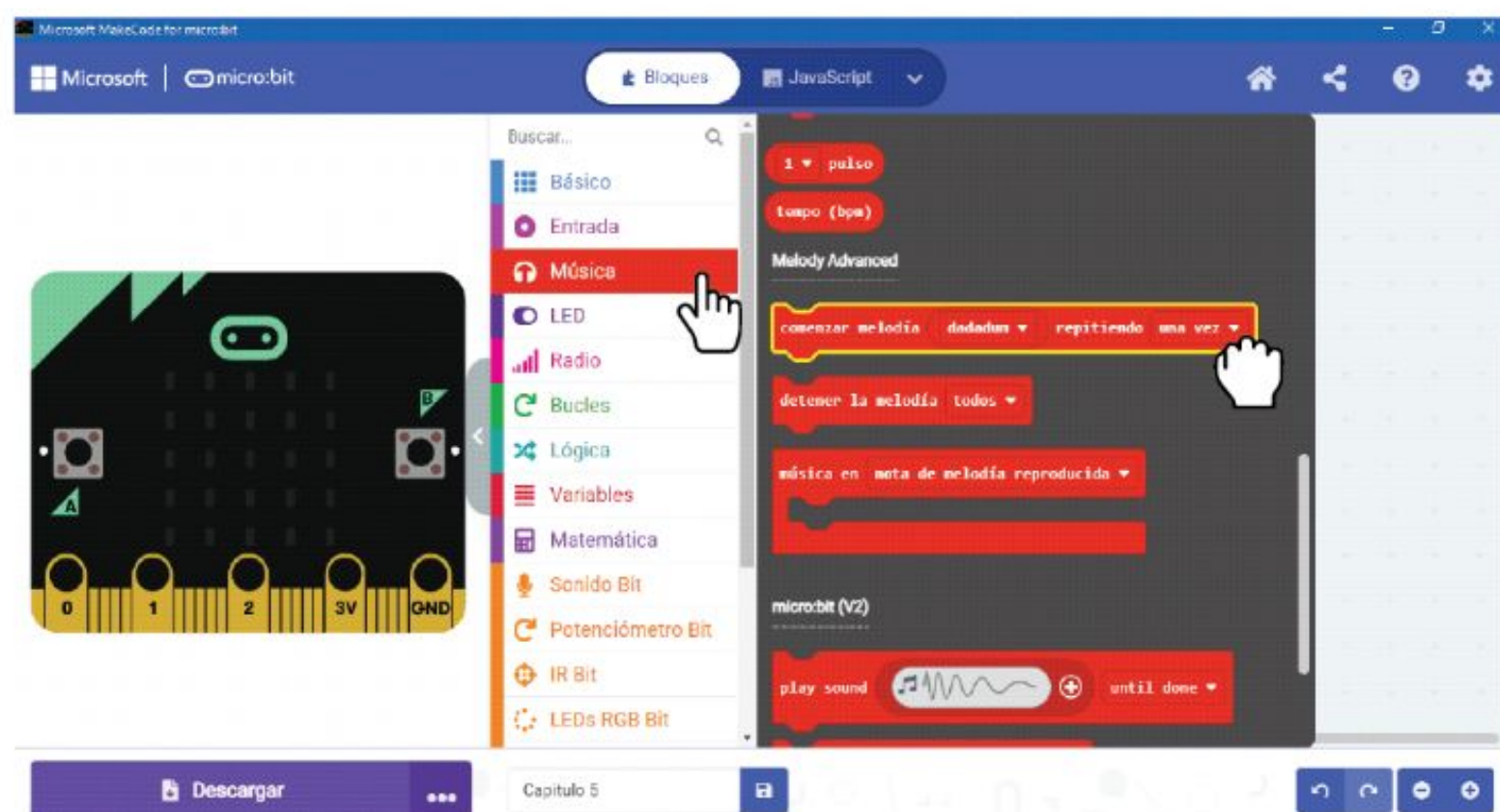
Paso 3 Haz clic en la categoría **[Básico]** y agrega dos bloques **[mostrar ícono]**. Cambia uno de los íconos a un "pequeño corazón". Encaja ambos bloques en el bloque **[mientras_ejecutar]**.



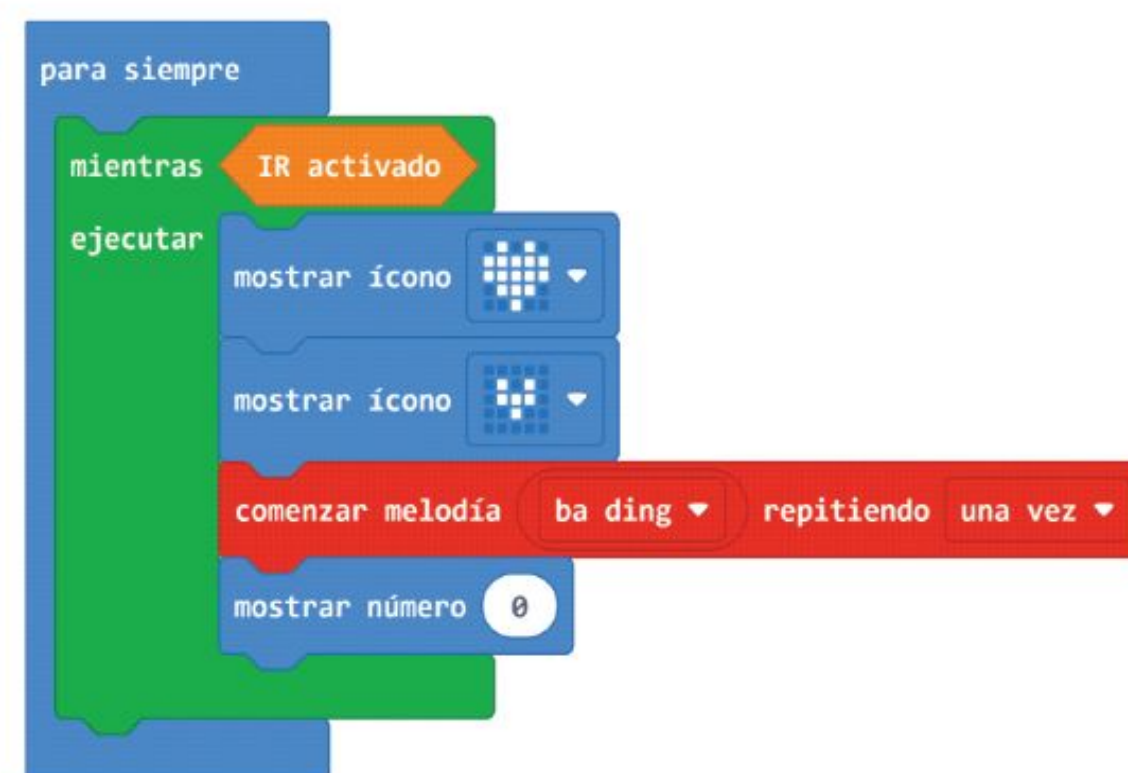
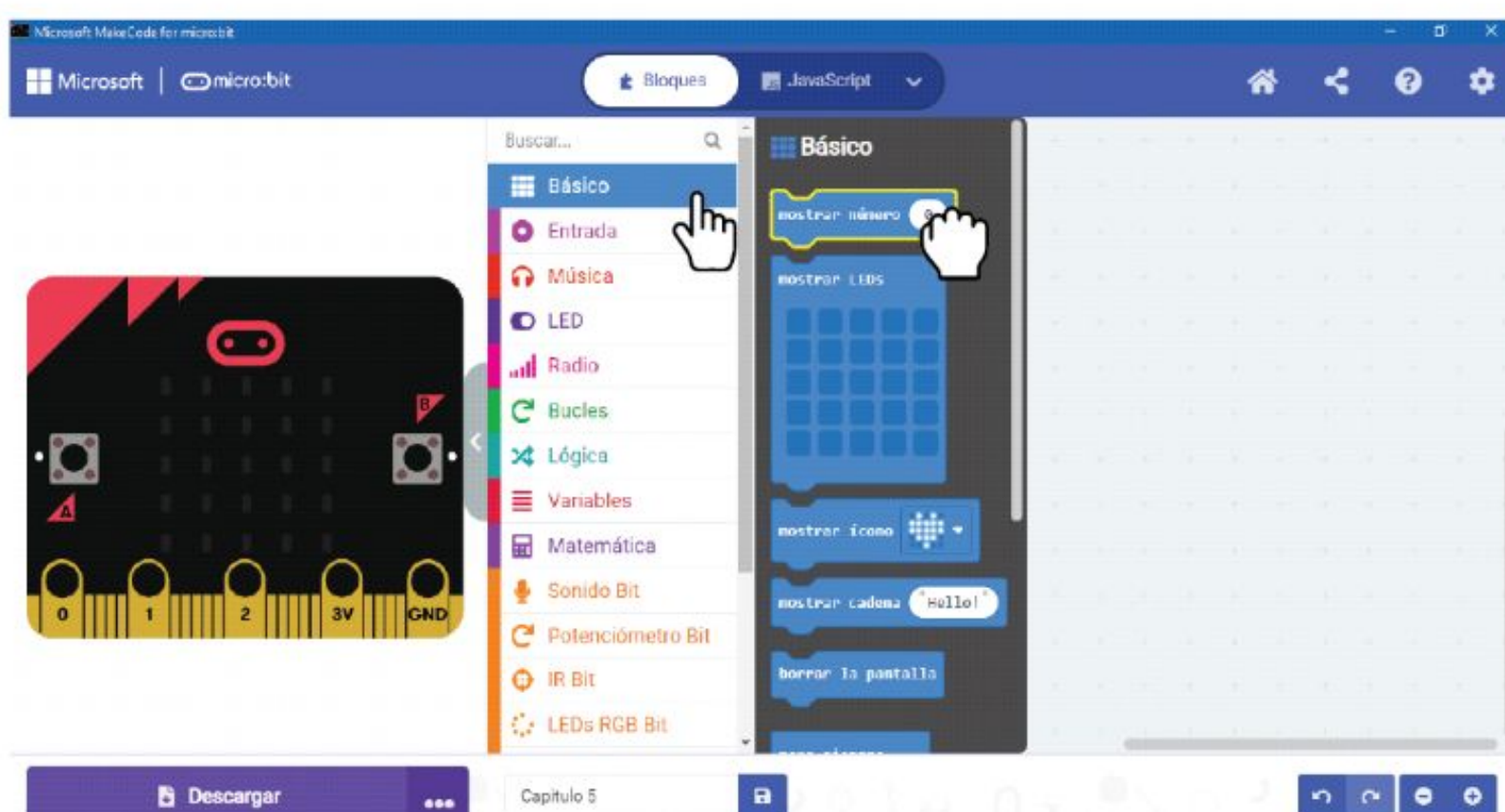
CAPÍTULO 5 : Datos digitales... Vamos a tirarlo~



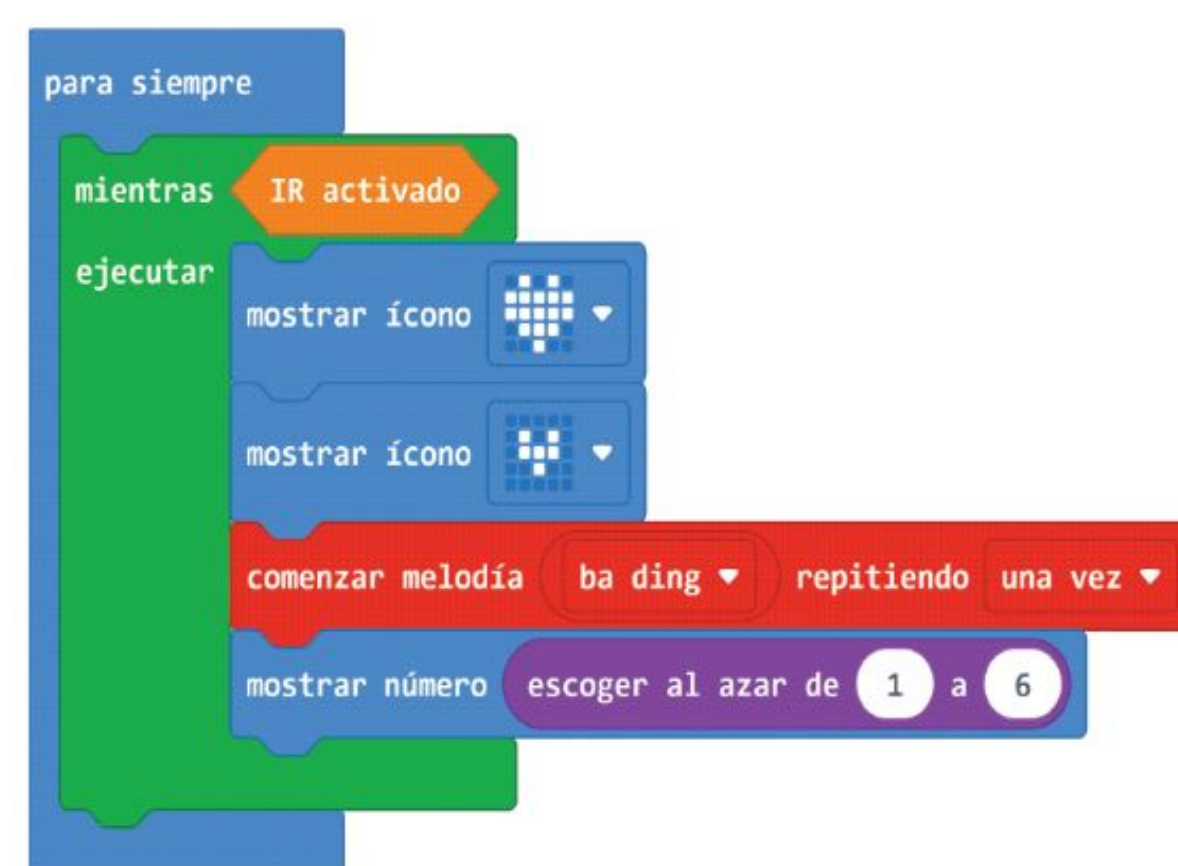
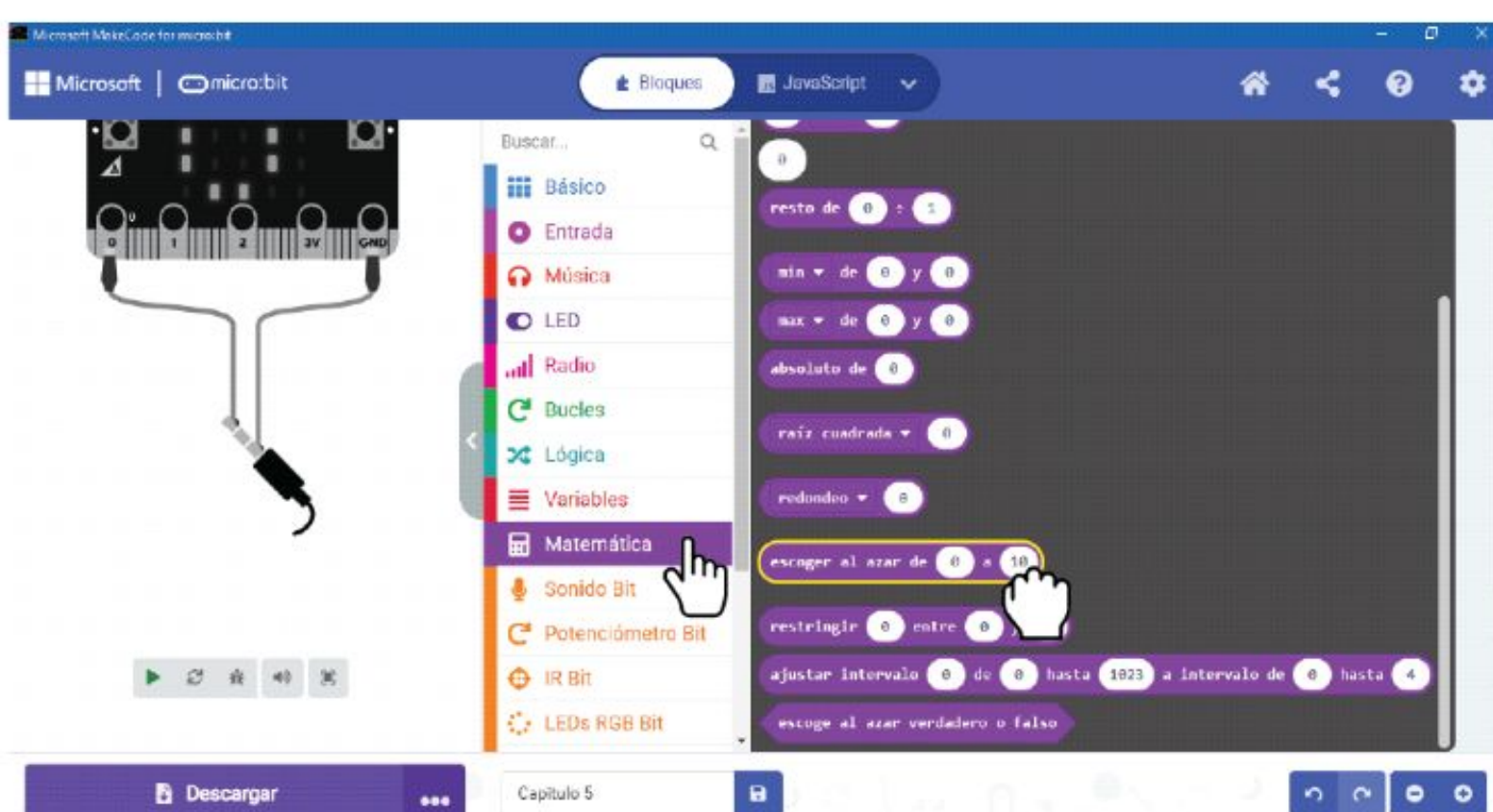
Paso 4 Haz clic en la categoría **[Música]** y selecciona el bloque **[comenzar melodía _ repitiendo]**. Cambia la melodía “dadadum” in “ba ding”.



Paso 5 Haz clic en la categoría **[Básico]** y luego selecciona el bloque **[mostrar número]**.

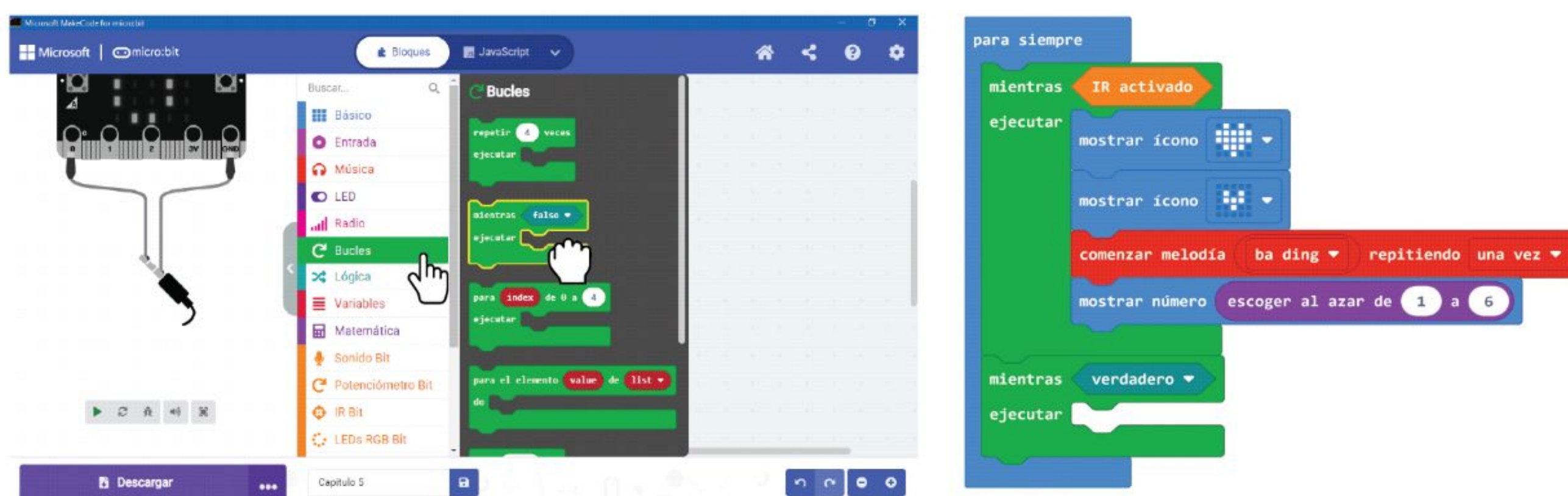


Paso 6 Haz clic en la categoría **[Matemática]** y luego selecciona el bloque **[escoger al azar de _ a _]**. Establece números de 1 a 6.

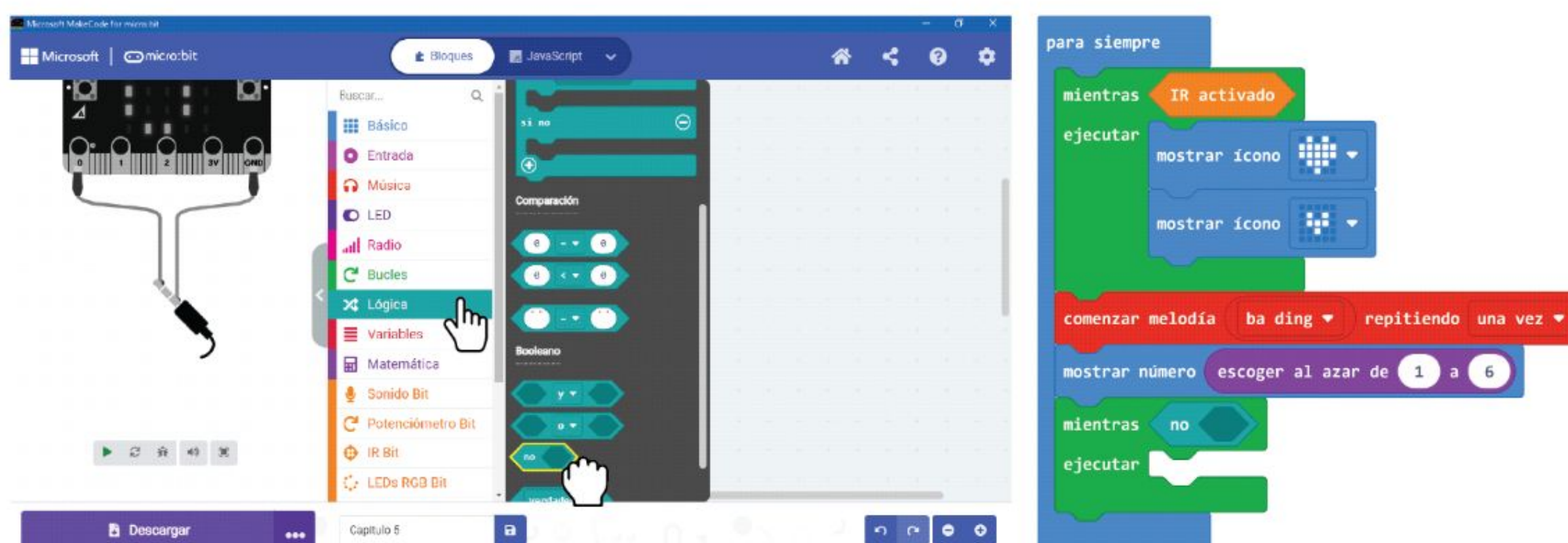


CAPÍTULO 5 : Datos digitales... Vamos a tirarlo~

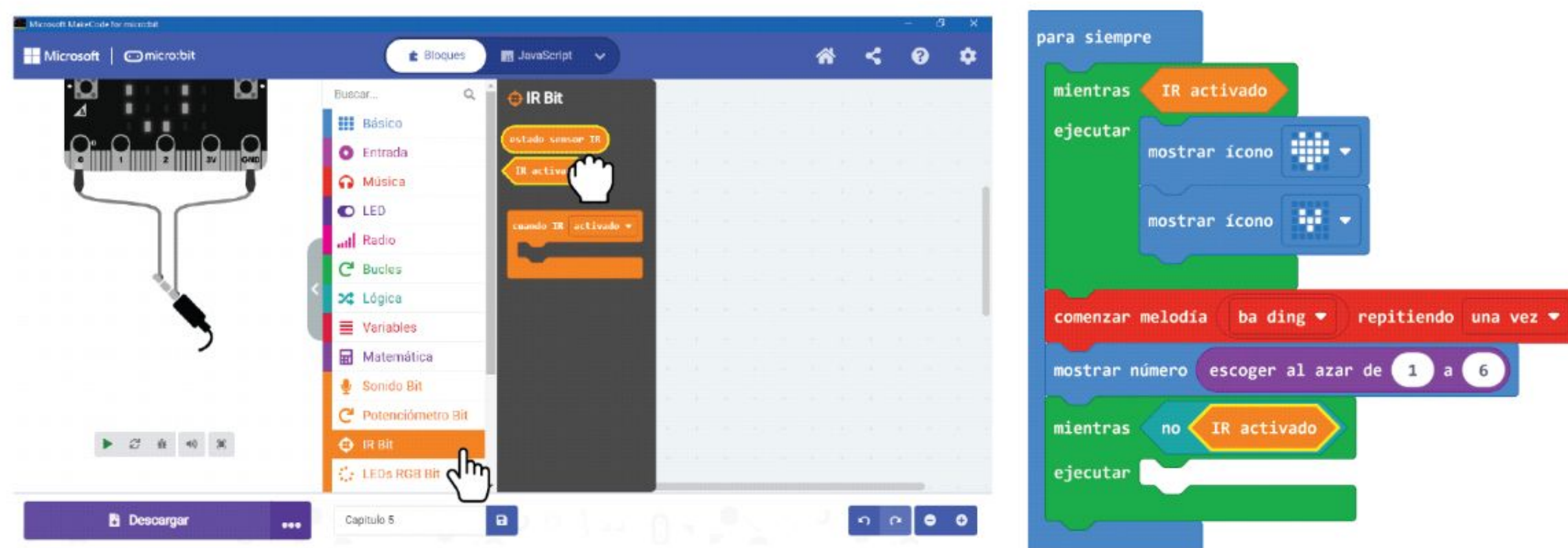
Paso 7 Haz clic en la categoría **[Bucles]** y luego selecciona el bloque **[mientras _ ejecutar]**.



Paso 8 Haz clic en la categoría **[Lógica]** y luego selecciona el bloque booleano **[no _]**. Encaja el bloque en la ranura de la condición relativa al bloque **[mientras _ ejecutar]**.



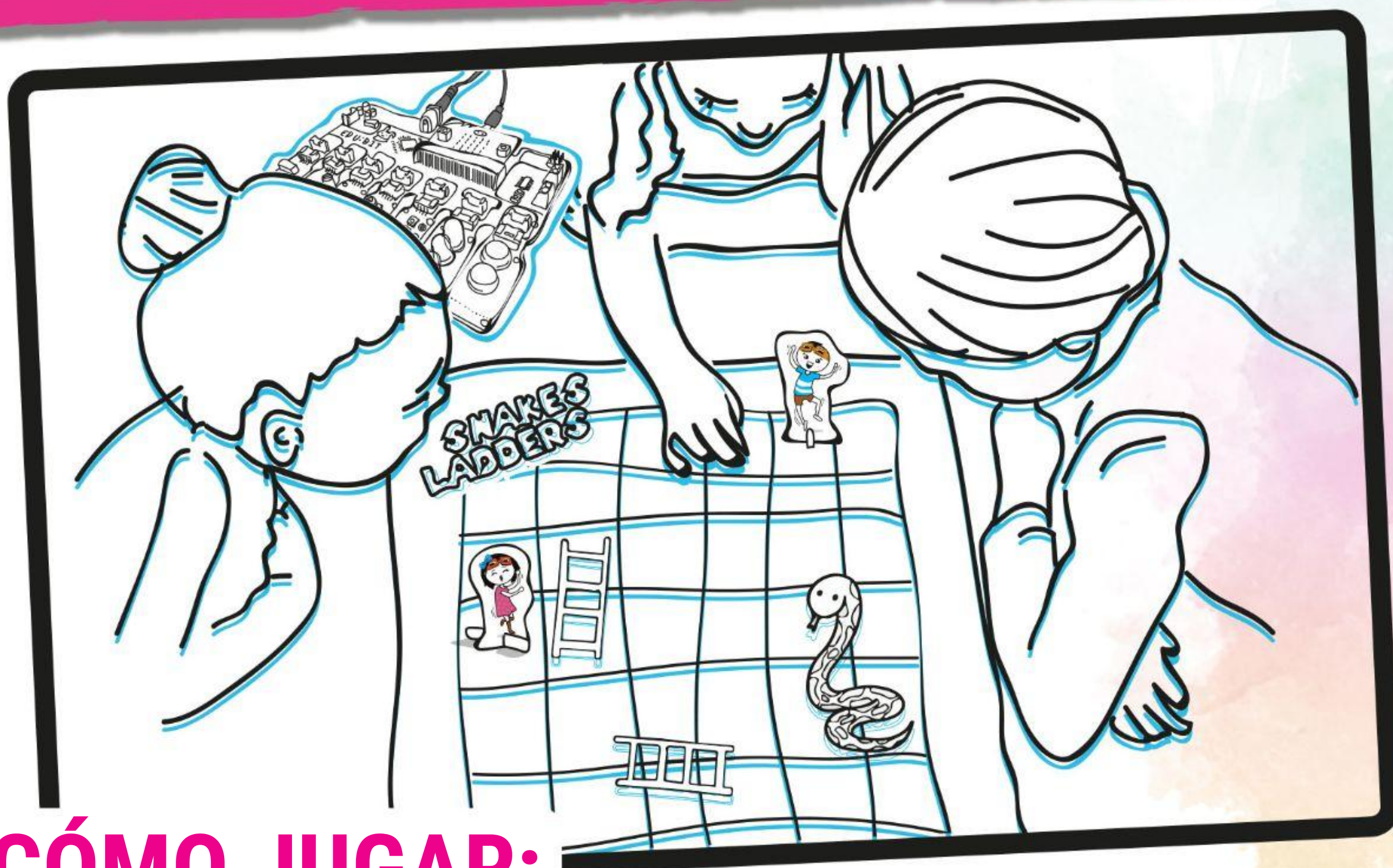
Paso 9 Haz clic en la categoría **[IR Bit]** y luego selecciona el bloque **[IR activado]**. Encaja el bloque en la ranura bloque en blanco **[no _]**.



Paso 10 Transfiere el código a tu EDU: BIT.

Vamos a jugar

Juego Escaleras y Serpientes



CÓMO JUGAR:

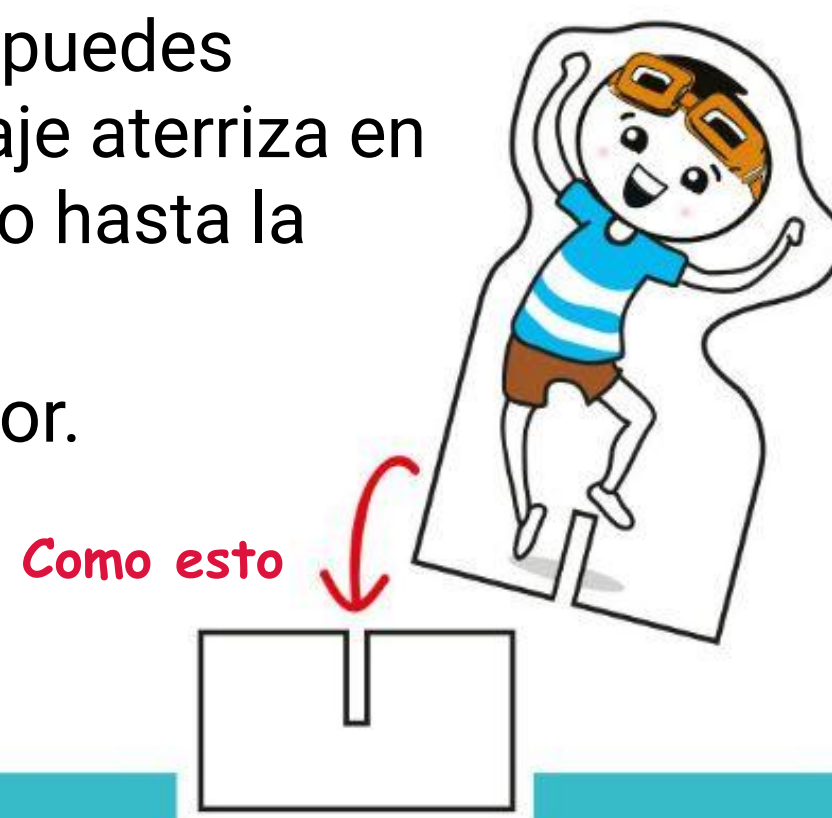
Cada jugador elige una pieza de un personaje y la coloca en el espacio marcado como "Empezar aquí".

Los jugadores se turnan para "tirar el dado": coloca la palma de tu mano sobre el bit infrarrojo. Cuando vea la animación del corazón que late, retire la palma de la mano.

Mueve el carácter hacia adelante el número de espacios indicado en la pantalla de matriz de LED (entre 1 y 6).

Si tu personaje cae en la parte inferior de una escalera, puedes moverte a la parte superior de la escalera. Si tu personaje aterriza en la cabeza de una serpiente, debes deslizarte hacia abajo hasta la punta de la cola de la serpiente.

El primer jugador en llegar al número 100 es el ganador.
¡Que lo pases bien!



NOTA!

El tablero de juego de Serpientes y Escaleras y las piezas de los personajes se incluyen en la caja. Extrae los personajes y sus soportes, luego júntalos para hacer las fichas de personajes.

MODIFICAR EL CÓDIGO



Usamos primero el bloque [mientras _ ejecutar] de la categoría [Bucles]. ¿Sabes cómo funciona un ciclo de bricolaje?

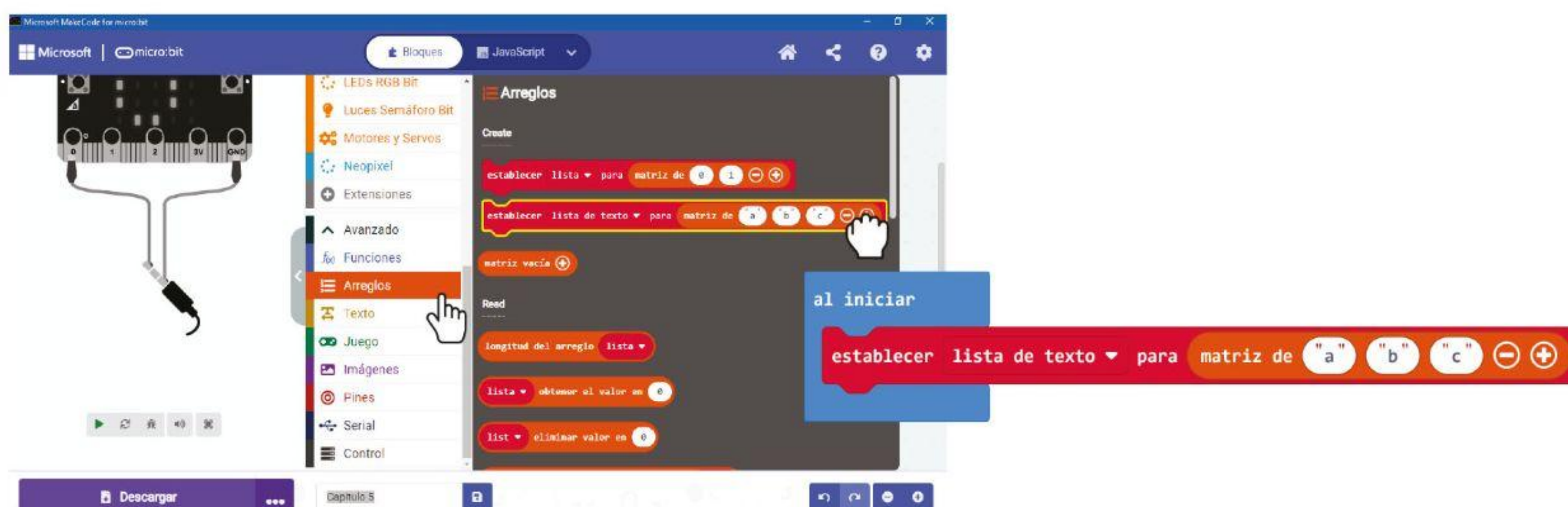
Cuando un programa llega a un bloque [mientras _ ejecutar], verifica la condición. Si se cumple la condición (es decir, es VERDADERO), el programa ejecutará los bloques de código dentro del bloque [mientras _ ejecutar]. Continuará reciclando, pero tan pronto como NO se cumpla la condición (es decir, se vuelve FALSO), el programa saldrá del ciclo y ejecutará los siguientes bloques de código.



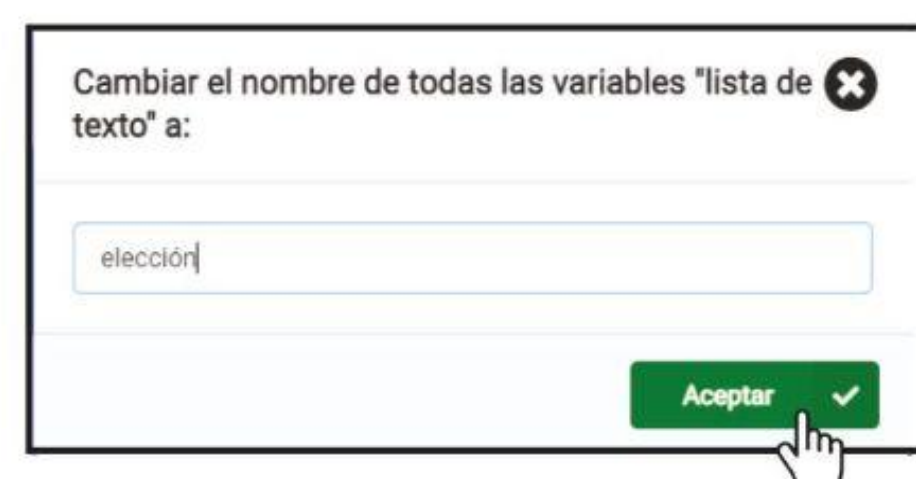


Oltre ad usare EDU:BIT come un dado digitale, puoi anche modificare il codice precedente per fare in modo che EDU:BIT ti aiuti quando hai diverse possibilità di scelta e non riesci a decidere cosa scegliere, per esempio - cosa mangiare per pranzo? Yum yum~

Paso 11 Haz clic en la categoría **[Avanzado] : [Arreglos]** y selecciona el bloque **[establecer lista de texto para matriz de _]**. Encaja el bloque en la ranura **[Básico] : [al iniciar]**.



Paso 12 Haz clic en el bloque **[lista de texto]** y selecciona **“Renombrar la variable”**. Escribe **“Opciones de almuerzo”** en la ventana emergente y luego haga clic en Aceptar.



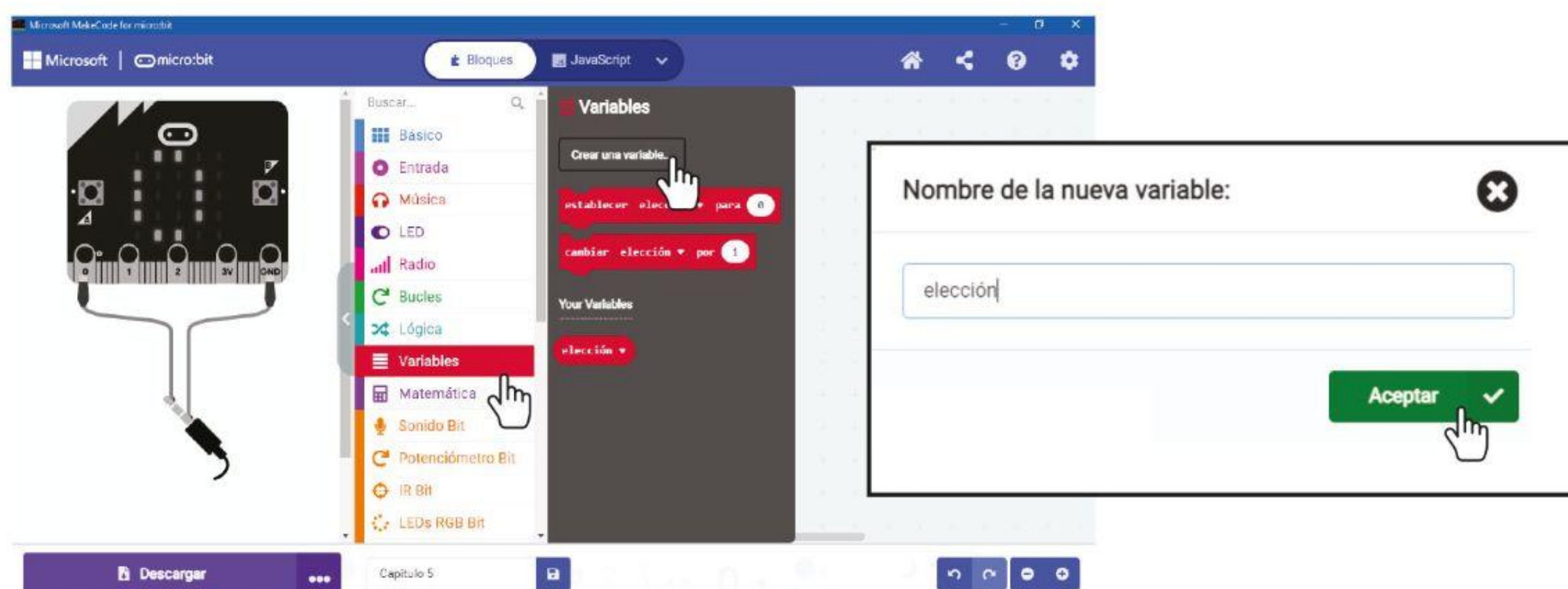
Paso 13 Haz clic en cada bloque de los elementos de la matriz y escribe una opción de almuerzo en cada bloque.



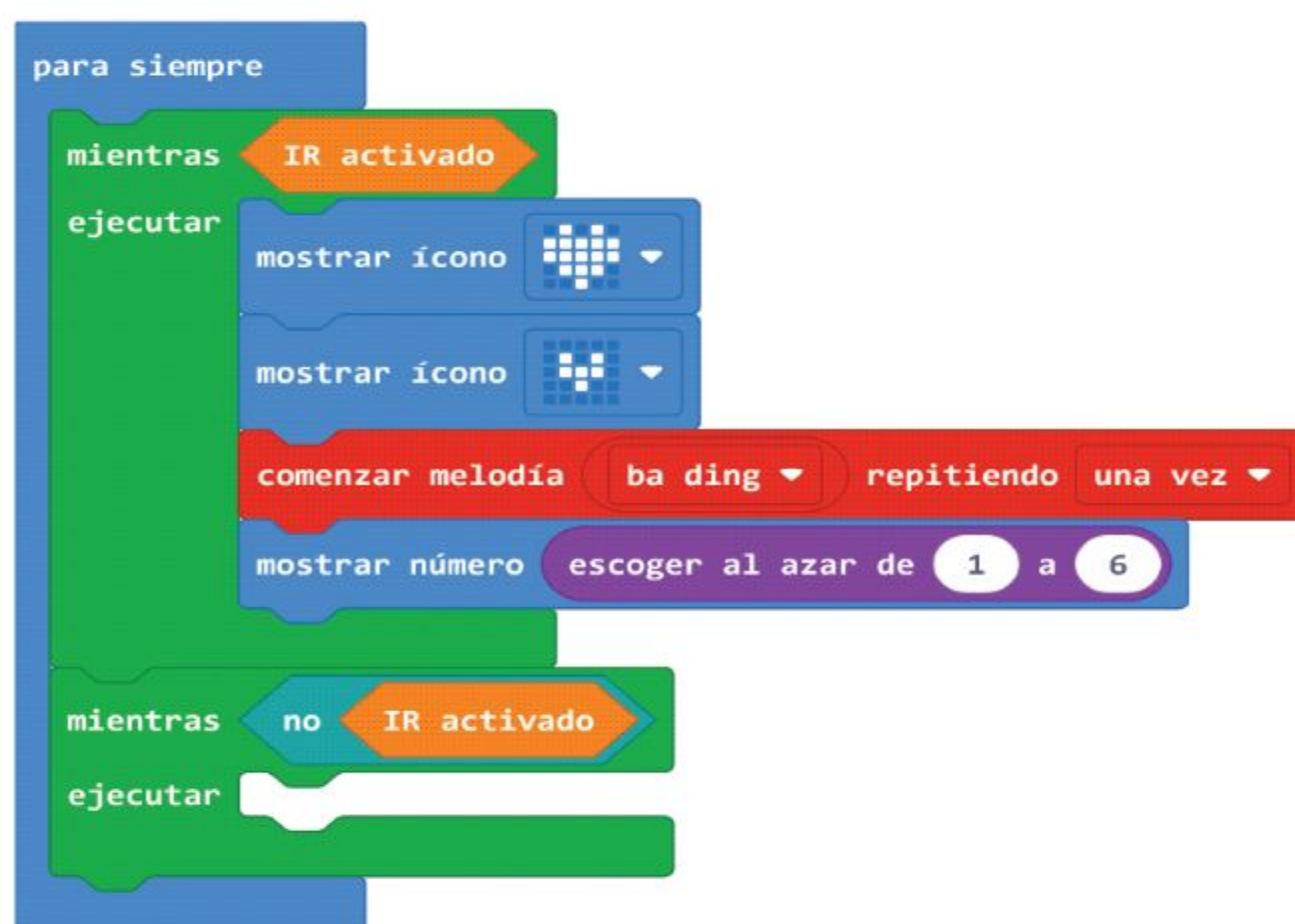
Haz clic en este botón para agregar más opciones.

CAPÍTULO 5 : Datos digitales... Vamos a tirarlo~

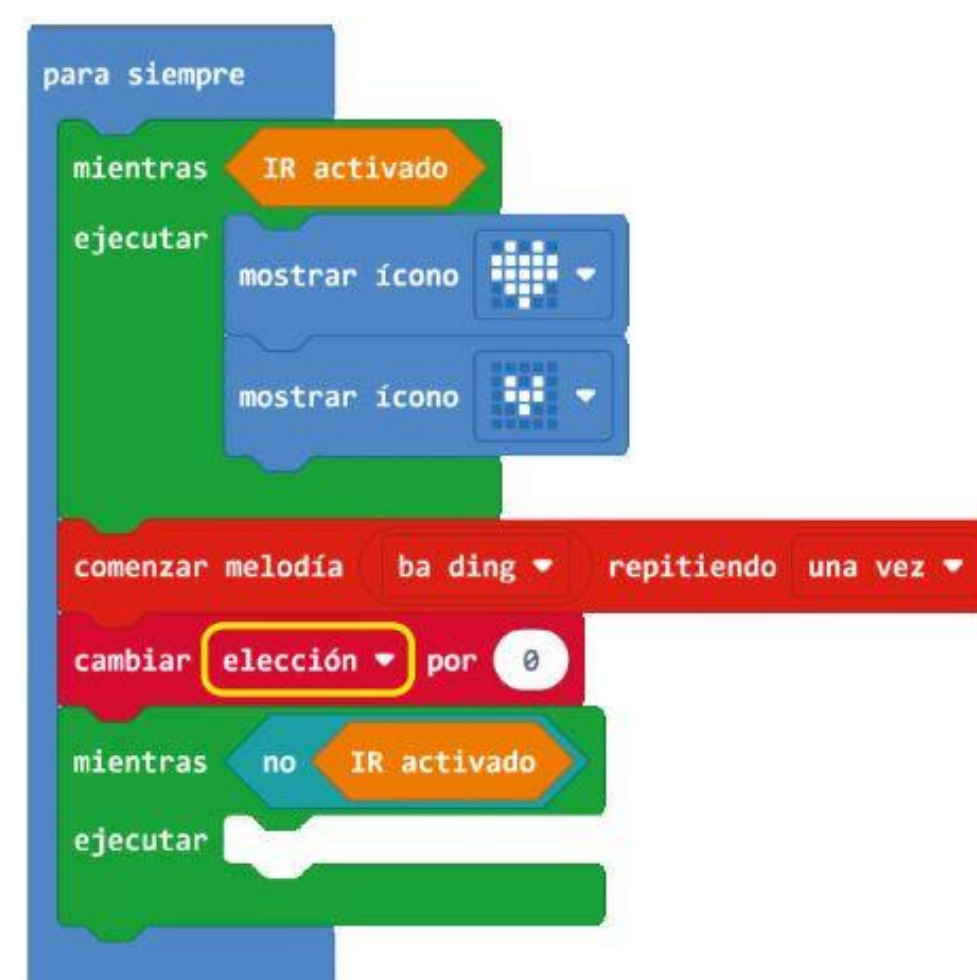
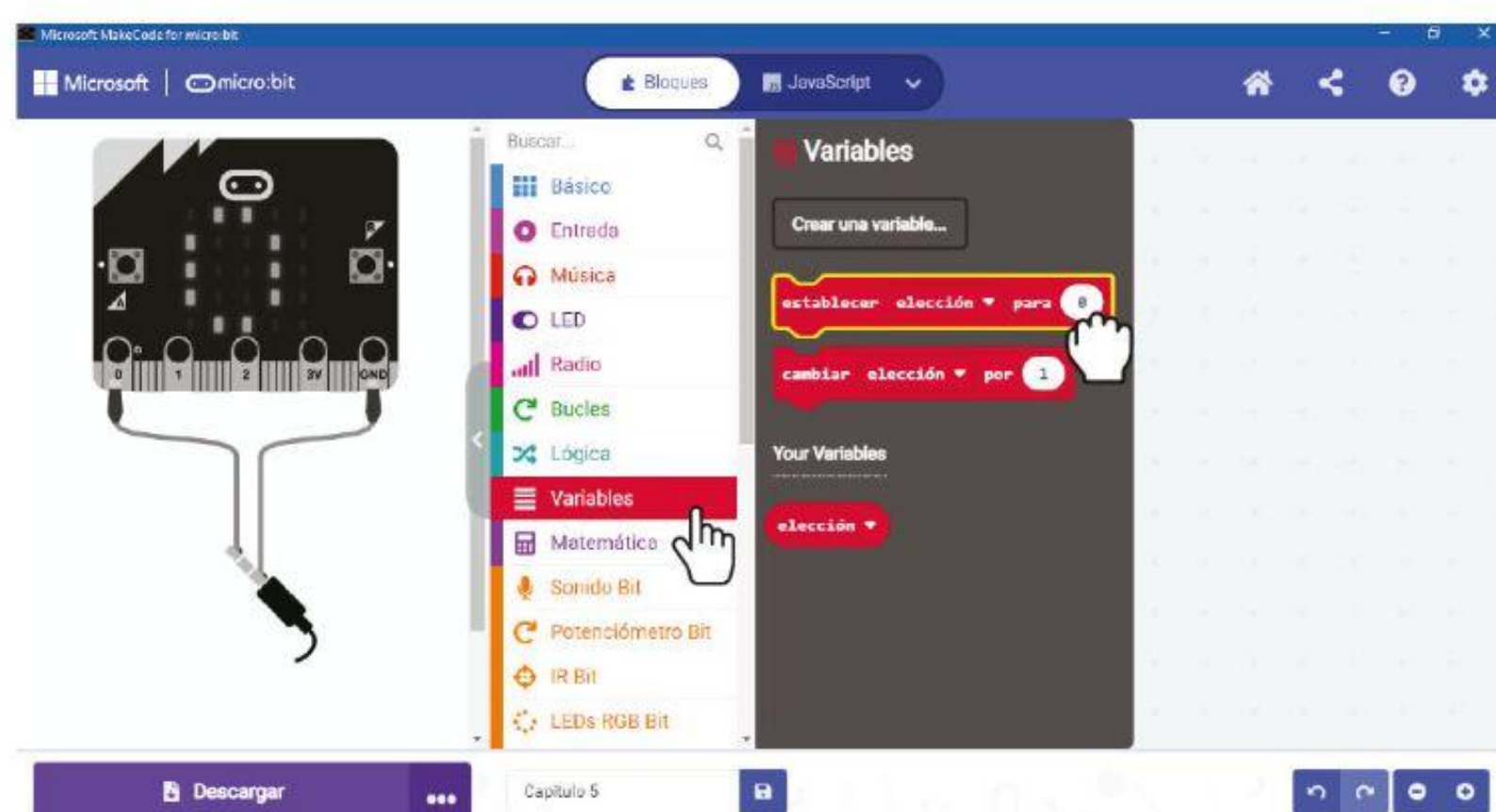
Paso 14 Haz clic en la categoría **[Variables]** y crea una nueva variable llamada 'elección'.



Paso 15 Haz clic derecho en el bloque **[mostrar número [escojer al azar de _ a _]]** y selecciona 'Eliminar bloques'.



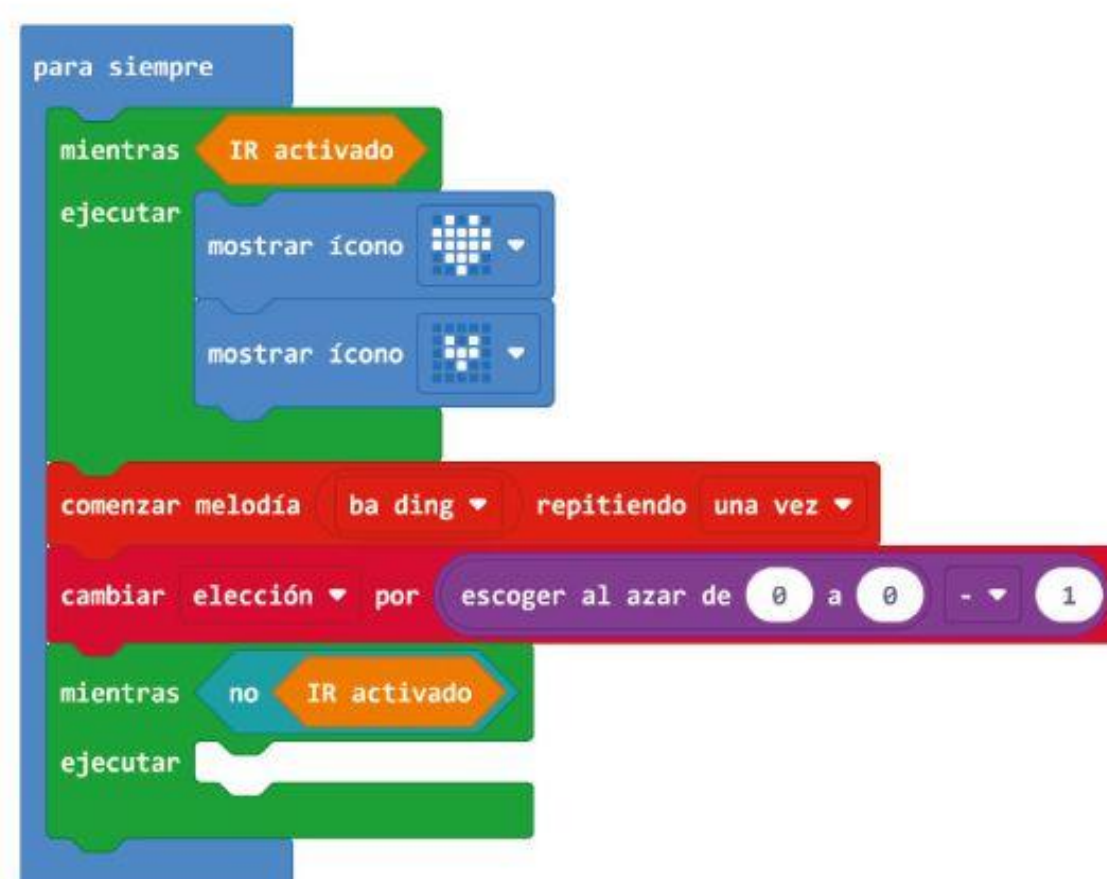
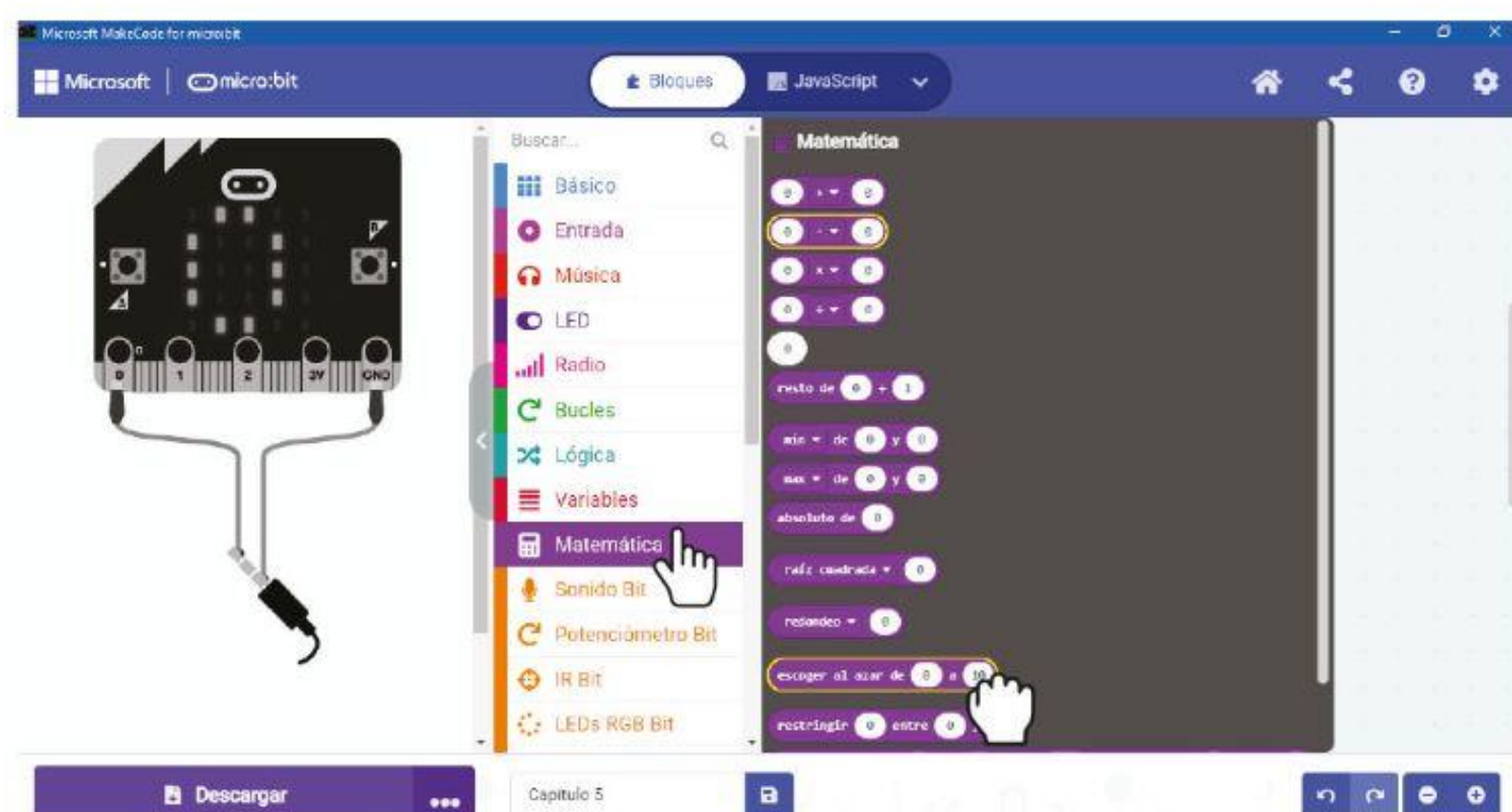
Paso 16 Haz clic en la categoría **[Variables]** y selecciona **[establecer _ para _]**. Coloca el bloque entre **[comenzar melodía _ ripitiendola _]** y el bloque **[mientras _ ejecutar]**. Cambia la variable a 'elección'.



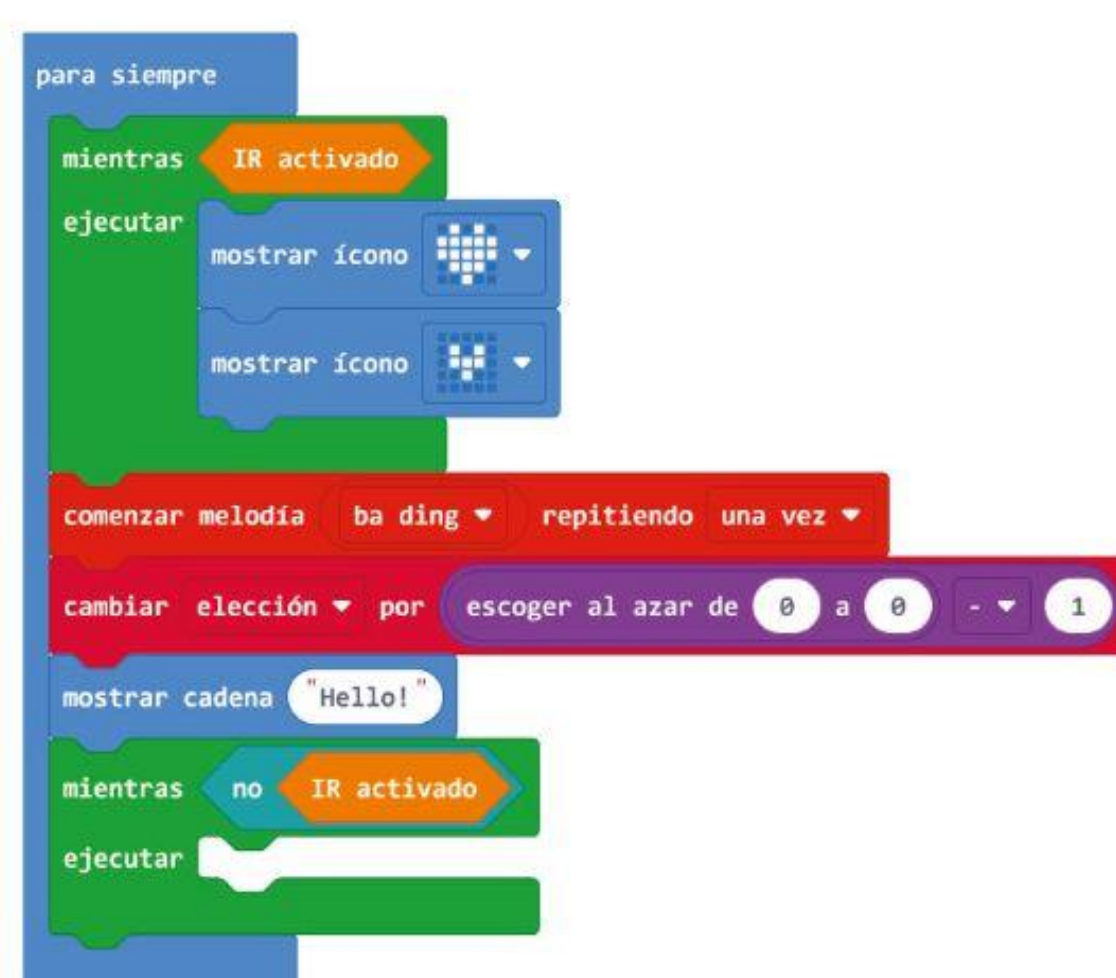
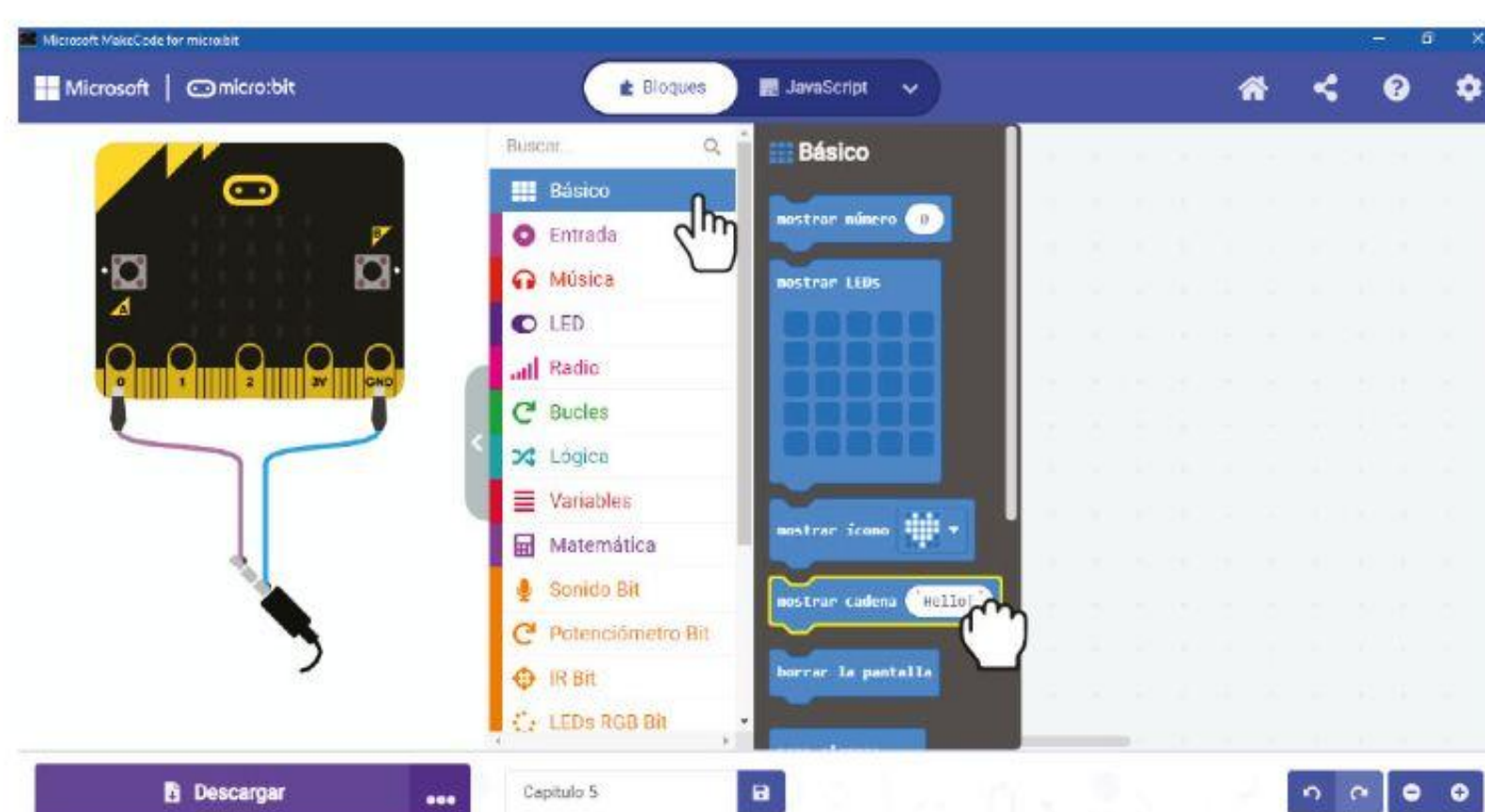
CAPÍTULO 5 : Datos digitales... Vamos a tirarlo~



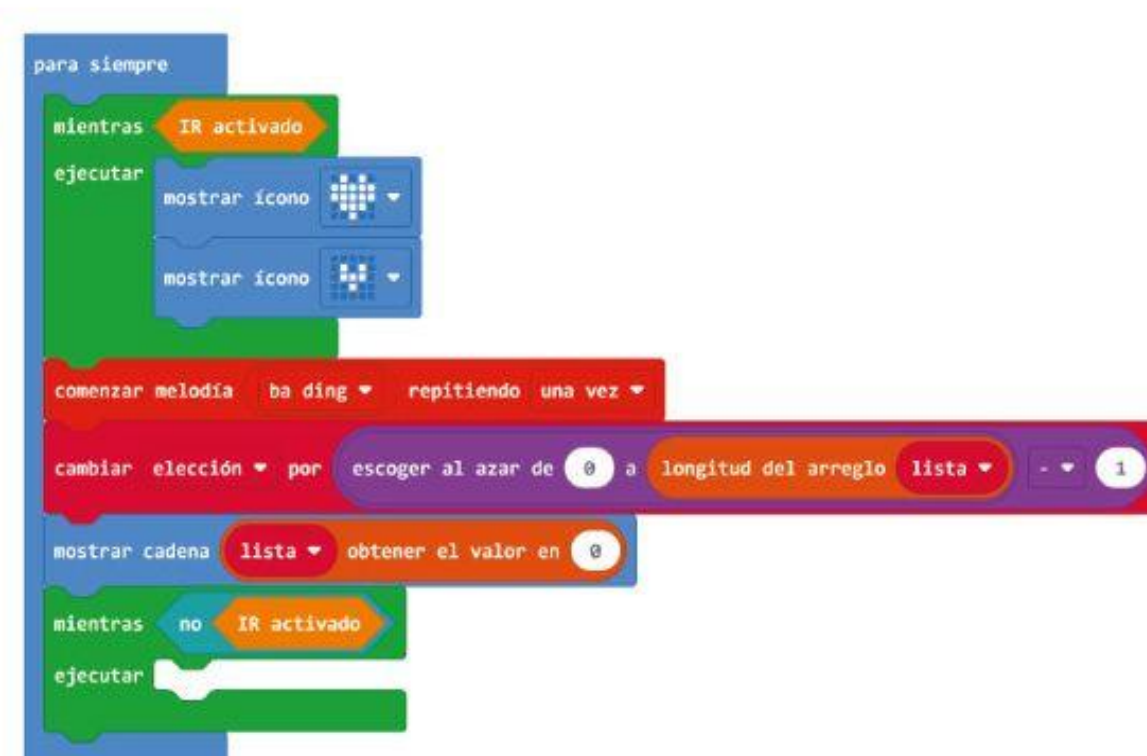
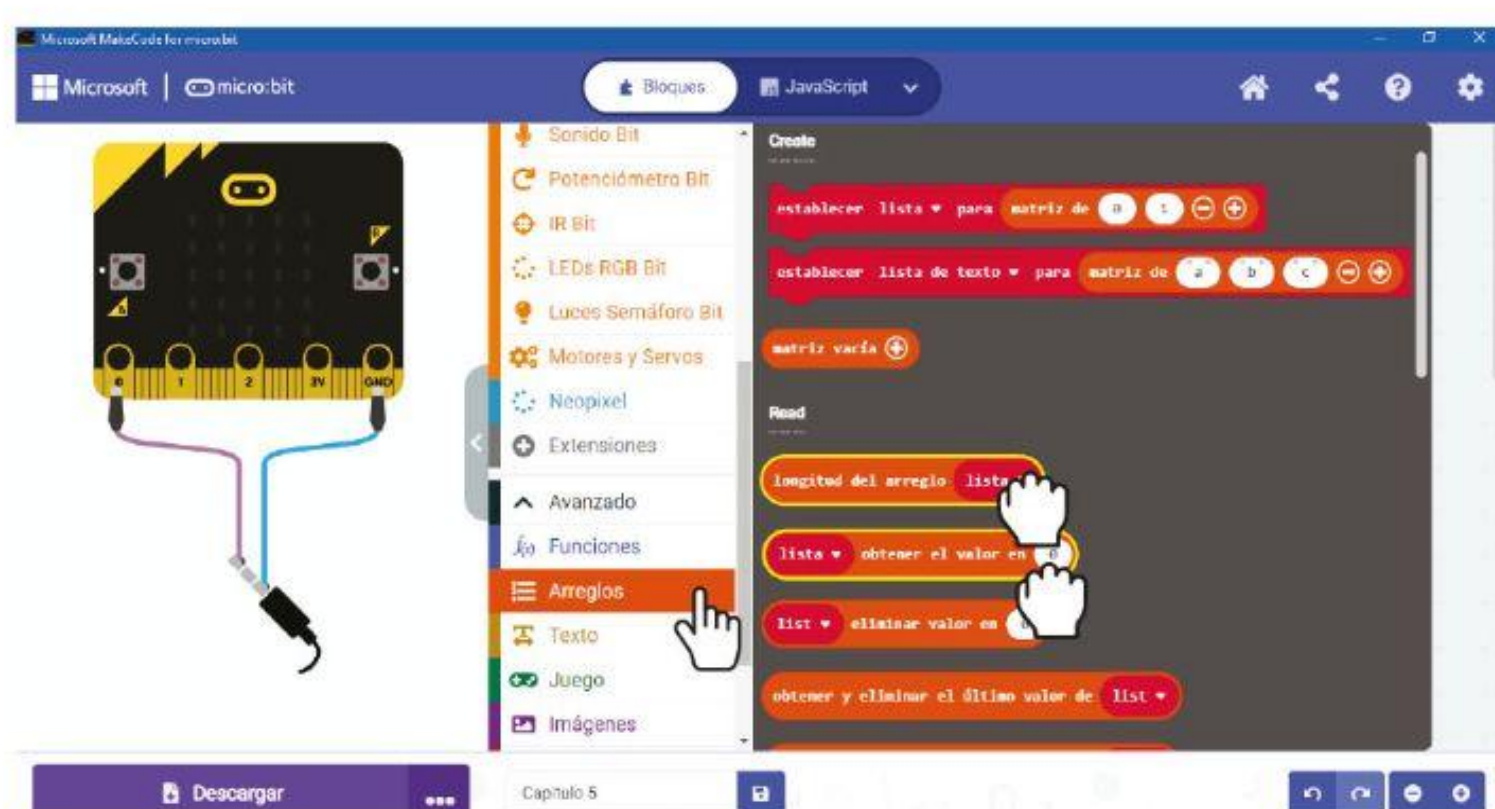
Paso 17 Haz clic en la categoría **[Matemática]** agrega el bloque **[escojer al azar de_a_]**. y agrega el bloque 1.



Paso 18 Haz clic en la categoría **[Básico]** y selecciona el bloque **[mostrar cadena]**.

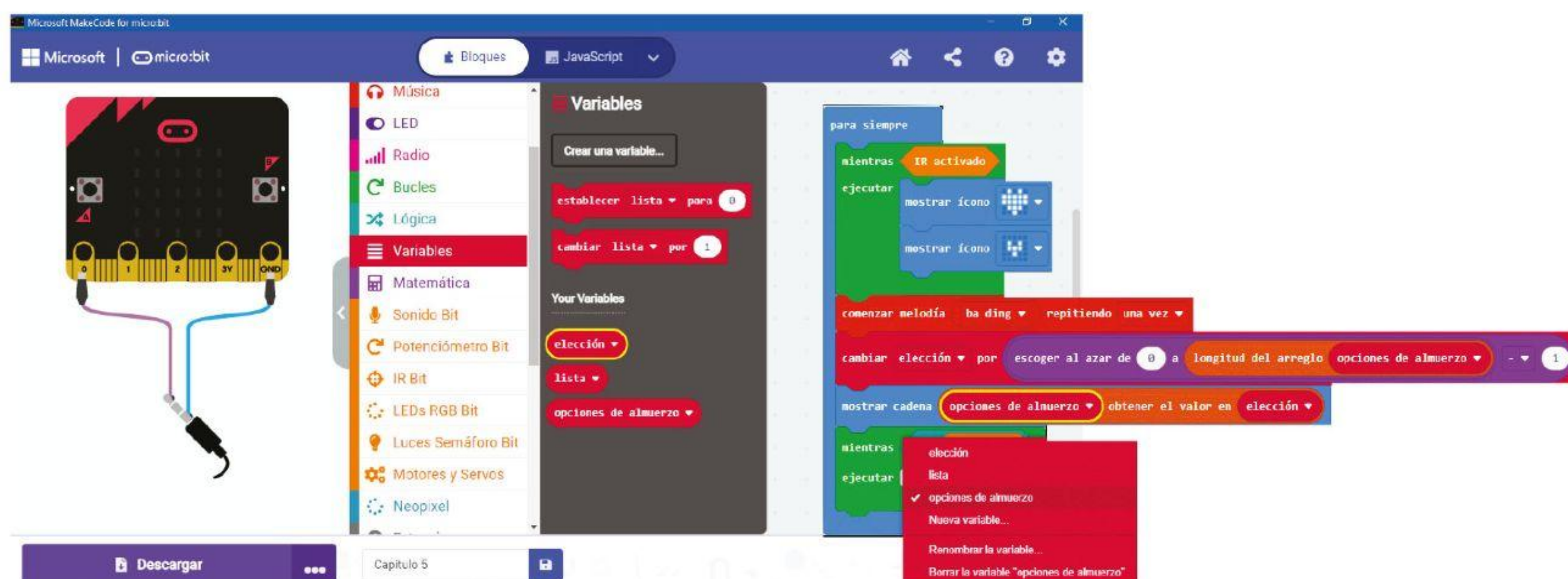


Paso 19 Haz clic en la categoría **[Avanzado]** : **[Arreglos]** y agrega los bloques **[longitud del arreglo]** e **[_ obtener el valor _]**.

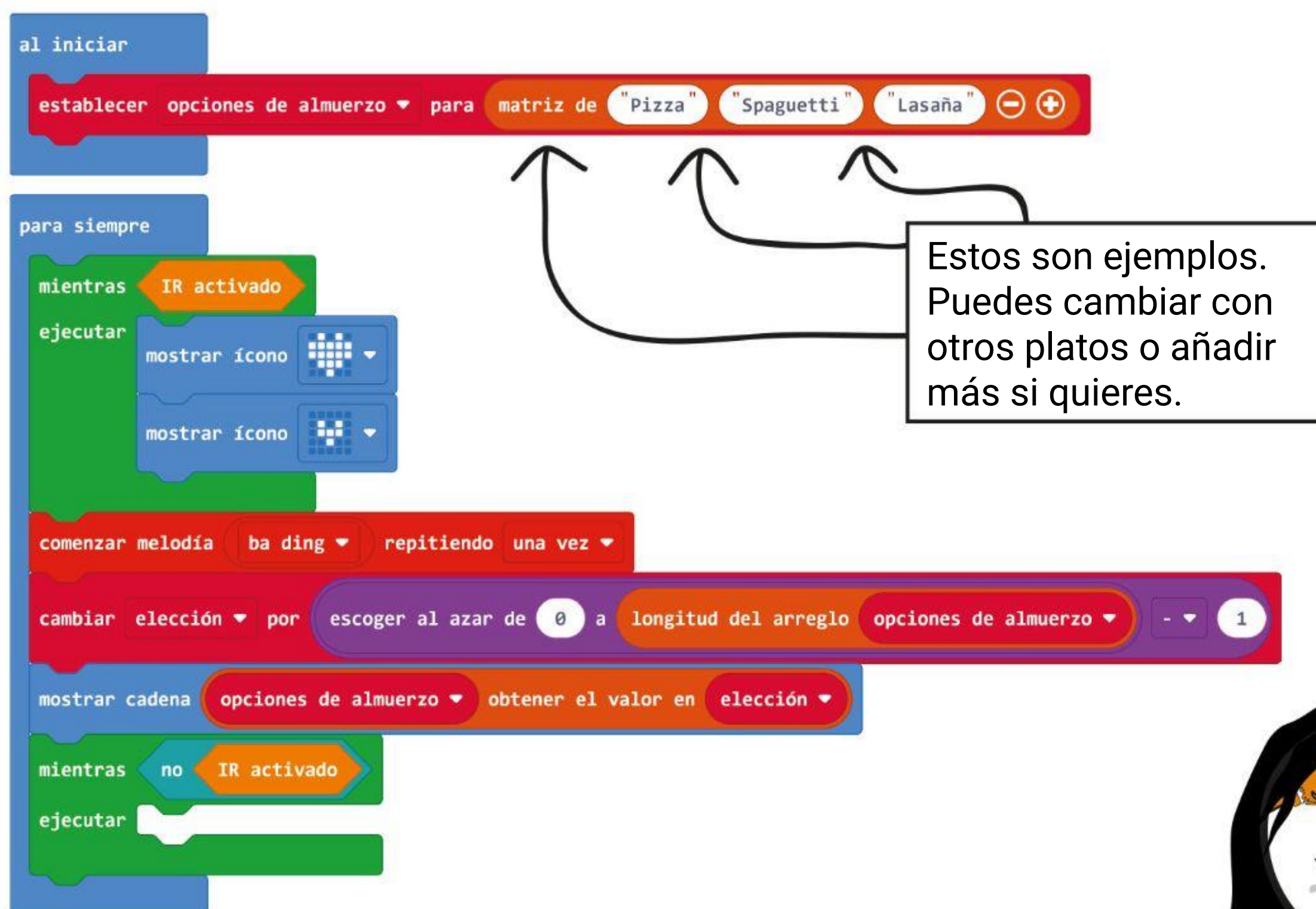


CAPÍTULO 5 : Datos digitales... Vamos a tirarlo~

Paso 20 Haz clic en **[Variables]** y cambia la variable a 'Opciones de almuerzo' para ambos bloques. Finalmente haz clic en la categoría **[Variables]** y selecciona el bloque **[Elección]**. Colócalo en la ranura vacía del bloque. **[_ obtener el valor en _]**.



Aquí está su código completo para el programa "¿Qué hay para cenar?":

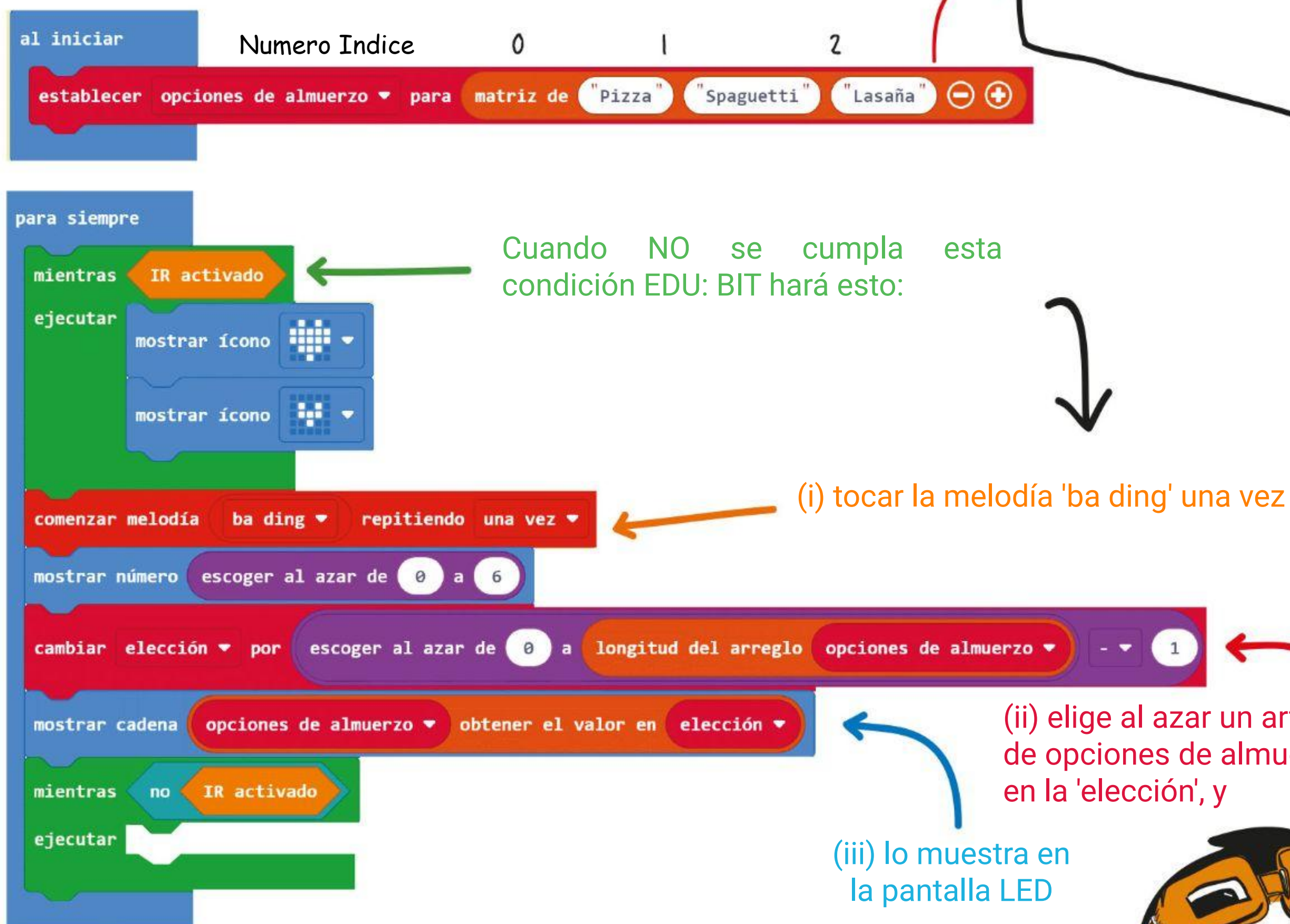
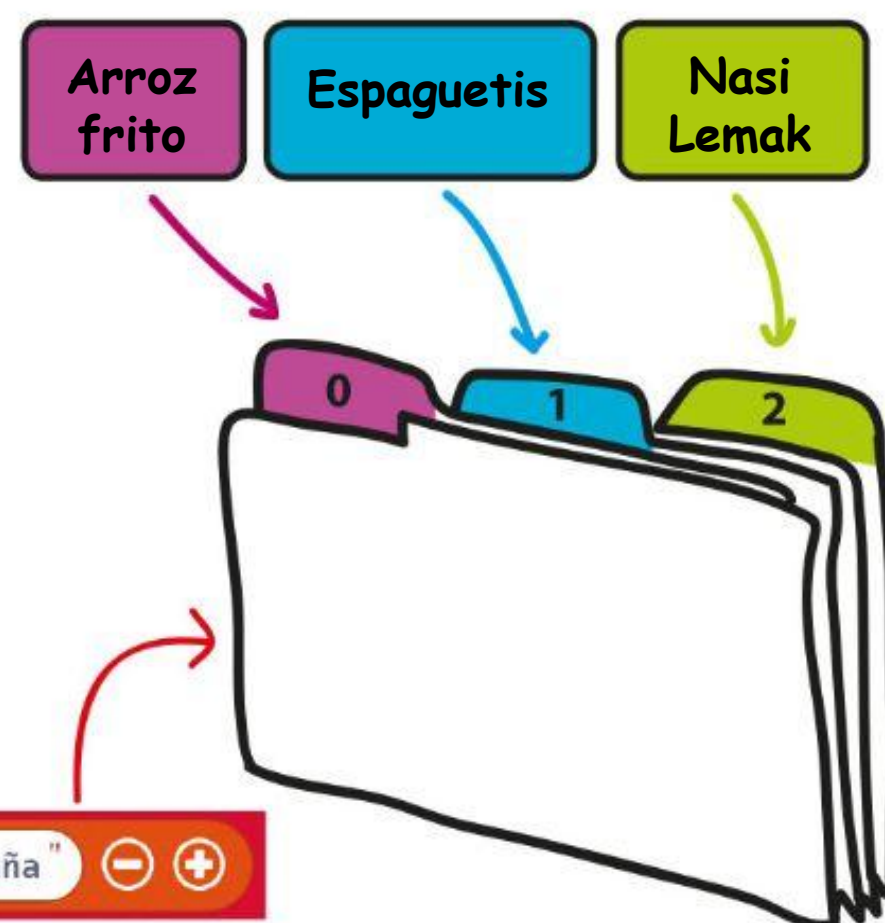


La próxima vez que no estés seguro de qué comer, puedes hacer que EDU: BIT decida por tí, activando el Bit infrarrojo y luego retirando tu mano. También puedes cambiar este código para ayudarte a decidir qué juego jugar con tus amigos. ¿Sabes qué tienes que cambiar para hacer esto?

MODIFICAR EL CÓDIGO

Un **array** es una lista o colección de variables relacionadas. Puede considerarse como una carpeta con varias secciones, y cada sección se usa para contener una parte de la información. Usamos una matriz para que podamos modificar fácilmente el código cuando necesitamos agregar o eliminar elementos de una lista.

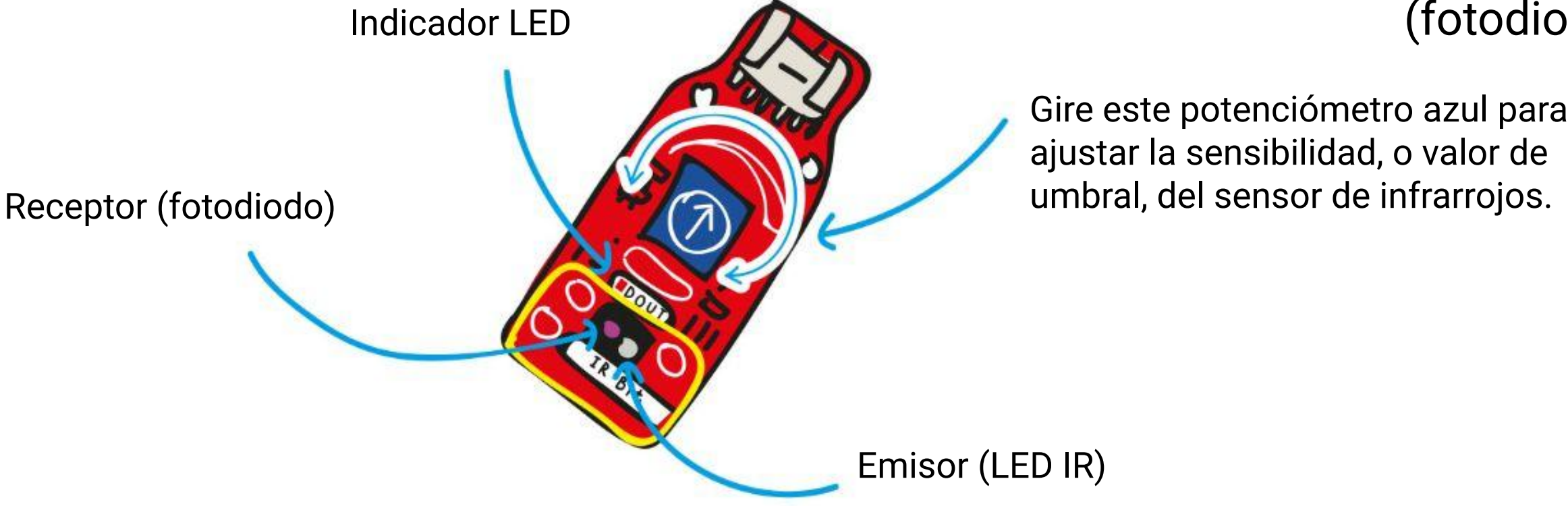
En este código, por ejemplo, hemos creado una matriz con tres elementos y los llamamos "Opciones de almuerzo". Podemos cambiar fácilmente la comida representada por cada elemento. También podemos añadir opciones o reducir el número de elementos de la lista simplemente haciendo clic en los botones + o -.



En la programación, comenzamos a contar desde 0 en lugar de 1. Por lo tanto, "Arroz frito" está en el número de índice 0 en la lista y el último elemento "Nasi lemak" está en el número de índice 2, a pesar de ser el tercer elemento.



Un **sensor de infrarrojos (IR)** es un dispositivo electrónico que se utiliza comúnmente para detectar obstáculos. El sensor consta de dos partes: un emisor (LED IR) y un receptor (fotodiodo).



¿Cómo funciona?

El LED infrarrojo (LED IR) emite luz infrarroja que se reflejará de regreso en el receptor si hay un objeto frente al sensor. El bit de infrarrojos se “activará” si la cantidad de luz reflejada es superior al umbral establecido. Cuando se activa, el indicador LED en el bit infrarrojo se encenderá.

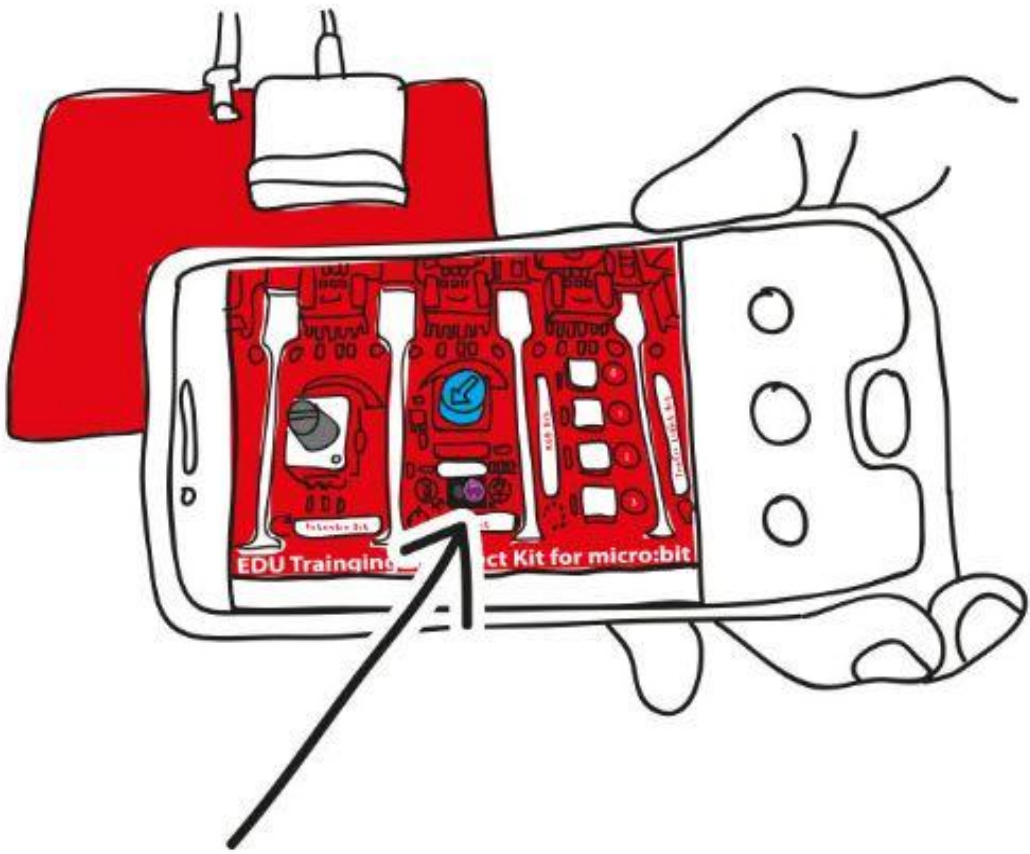
Si no hay ningún objeto o el objeto está demasiado lejos, muy poca luz infrarroja se reflejará en el receptor. Por lo tanto, el bit de infrarrojos no se activará.

Sin embargo, es posible que el sensor de infrarrojos no funcione como se espera en las siguientes condiciones:



¿Sabías que?

La luz infrarroja es invisible a simple vista, sin embargo, se puede ver la luz infrarroja simplemente mirando el LED infrarrojo a través de la cámara de un teléfono.



DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para que funcione como indicador de malas posturas.	
En el inicio	Desplaza el texto recordatorio "Presta atención a tu postura"
Cuando el sensor de infrarrojos está activo	Muestra una cara sonriente en la pantalla y encienda el LED verde.
Cuando el sensor de infrarrojos no está activo	Toca la melodía 'ba ding' una vez. Muestra una cara triste en la pantalla y parpadea el LED rojo.

¿Cómo funciona?

Fija EDU: BIT al respaldo de su silla como se muestra. Siéntate cómodamente con la postura correcta. Ajusta el mando del potenciómetro azul en el bit infrarrojo hasta que el indicador LED se ilumine (el bit infrarrojo detecta su espalda). Este proceso se llama calibración.



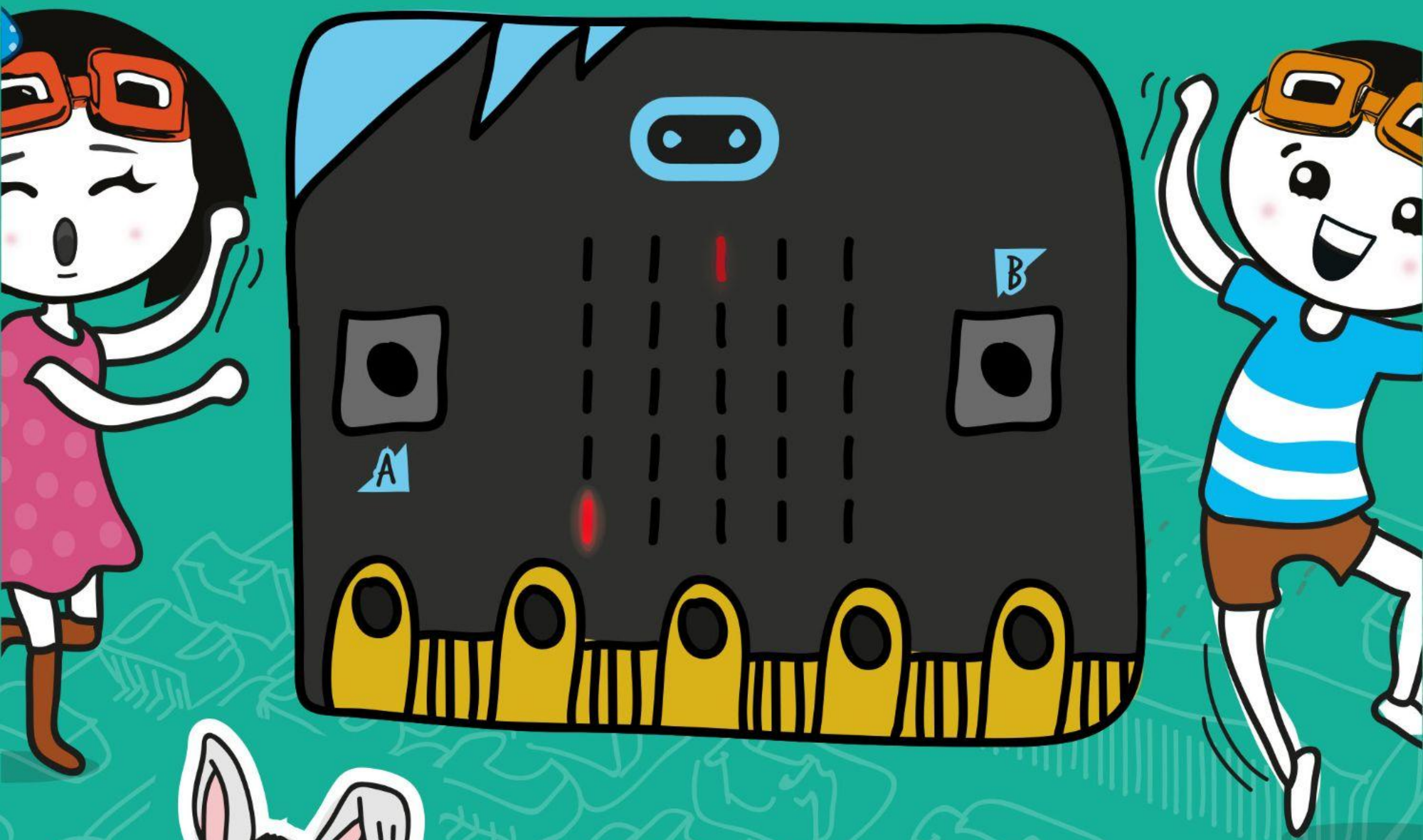
Deberás volver a calibrar el bit infrarrojo si llevas una camiseta de otro color. ¿Sabes por qué?



CAPÍTULO 6

Corre que te pillo

Bit Potenciómetro



¡Escanéame!

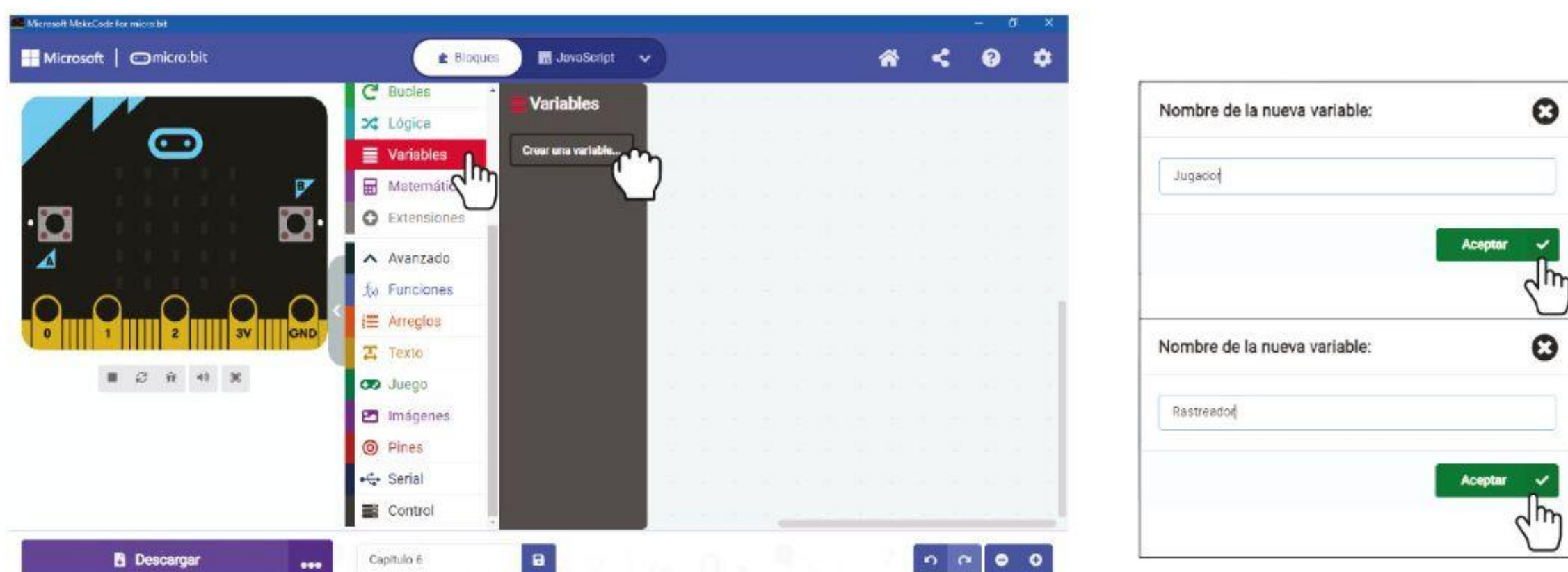


¡PROGRAMAMOS!

Paso 1 Crea un nuevo proyecto en el Editor de MakeCode y agrega la extensión EDU: BIT (puedes consultar la página 40). Hacer clic en **[Avanzado]** y luego en la categoría **[Juego]**. Selecciona el bloque **[iniciar cuenta regresiva (ms) _]**, encájalo en el bloque **[al iniciar]** y cambia el valor a **30000**.



Paso 2 Haz clic en la categoría **[Variables]** y haz clic **[Crear una variable]**. Escribe 'Jugador' en la ventana emergente y luego haz clic en Aceptar. Crea otra variable llamada 'Rastreador'.



Paso 3 Haz clic en la categoría **[Variables]** y luego seleccione el bloque **[establecer _ para _]**. Duplica el bloque **[establecer _ para _]** y encaja ambos en el bloque **[al iniciar]**. Establece las variables en 'Jugador' e 'Rastreador' respectivamente.



CAPÍTULO 6 : Corre que te pillo

Paso 4 Haz clic en **[Avanzado]** y luego en la categoría **[Juego]**. Selecciona el bloque **[create sprite at x:_ y:_]**. Duplica y encaja los bloques en bloques **[establece _ para _]**. Cambia el valor a **x:0 y:5** por 'Rastreador' e **x:2 y:0** por 'Jugador'.



Paso 5 Haz clic en la categoría **[Avanzado]** : **[Juego]** y selecciona el bloque **[_ girar _ (°) _]**. Pon el bloque en la ranura **[al iniciar]**. Selecciona la variable 'Jugador' y establece los grados a **90**.

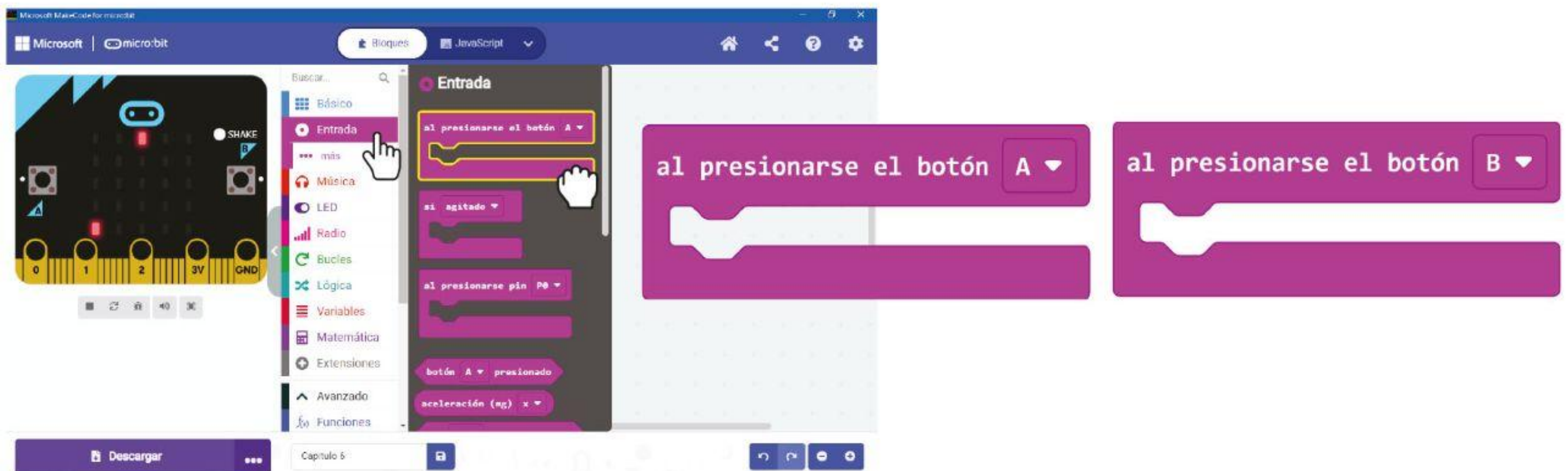


Paso 6 Haz clic en la categoría **[Avanzado]** : **[Juego]** y selecciona el bloque **[_ establece _ en _]**. Selecciona la variable 'Jugador', cambia 'x' en 'brillo' y configúralo en **50**.





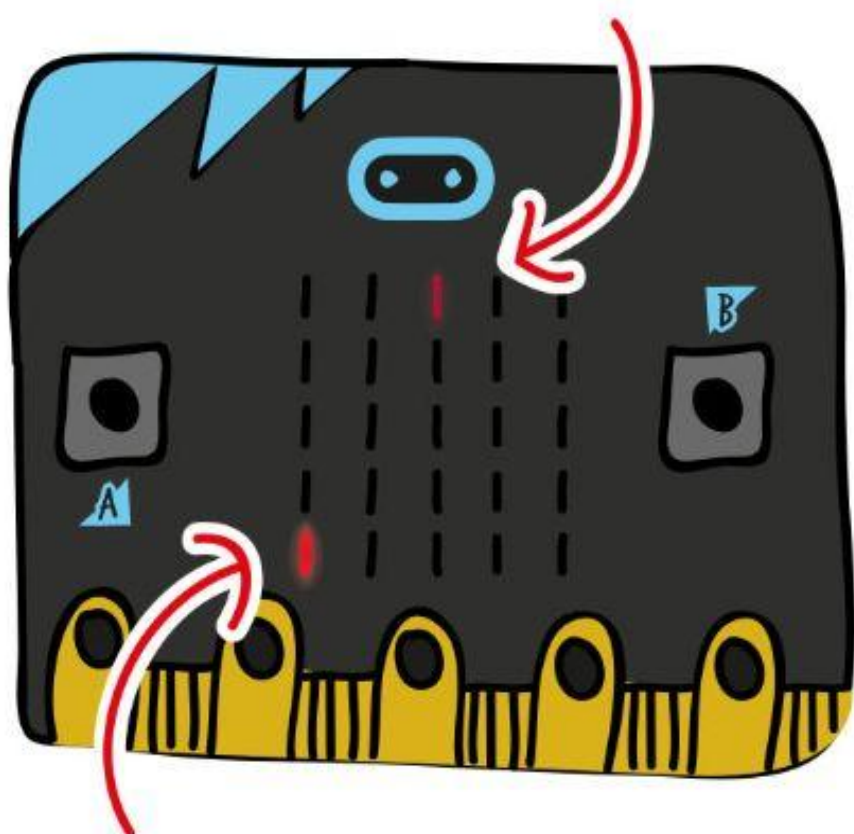
Paso 7 Haz clic en la categoría **[Entrada]** y selecciona el bloque **[al presionar el botón _]**. Duplica el bloque y selecciona el botón 'B' En el segundo **[al presionar el botón _]**.



Paso 8 Haz clic en la categoría **[Juego]** y selecciona el bloque **[_ desplazar _]**. Duplica y pon los bloques en las ranuras. **[al presionar el botón _]**. Selecciona la variable **'Jugador'** para ambos bloques y cambia los valores a **-1** (botón A presionado) y **1** (botón B presionado).



Sprite Jugador



Sprite Rastreador

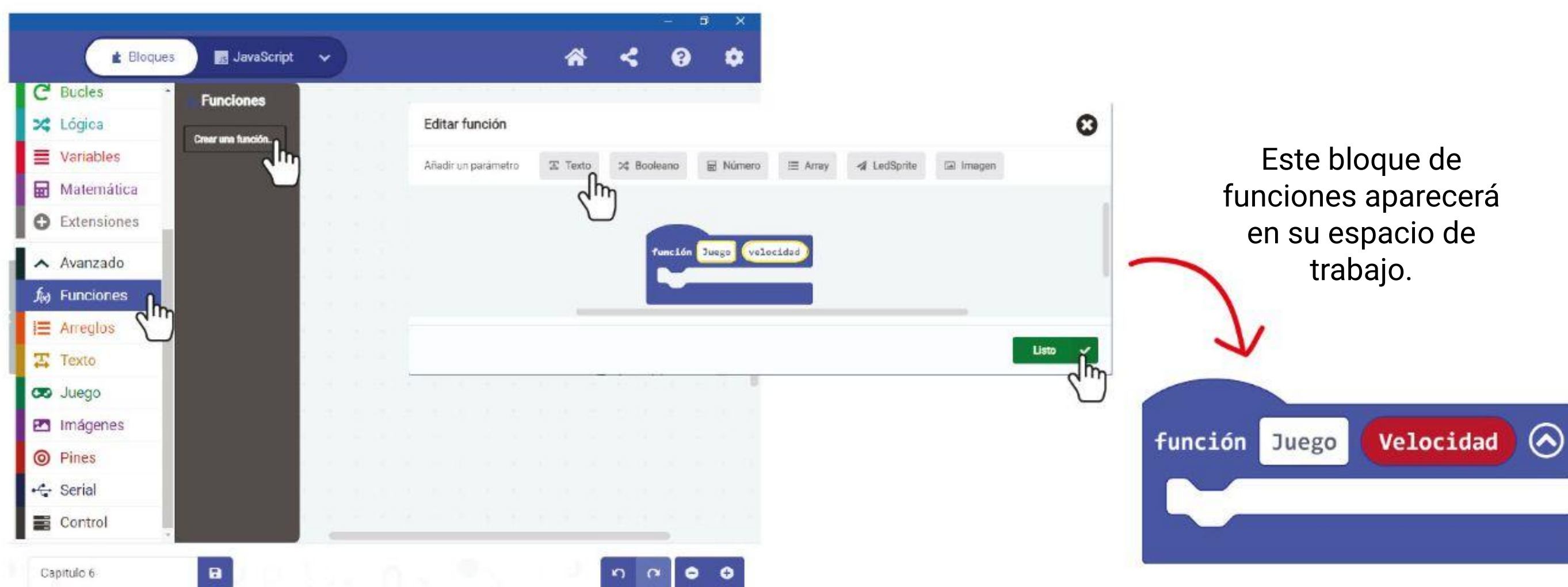
Transfiere el código a tu EDU: BIT. Cuando presionas el botón azul (B), ¿nota que el LED del atenuador se está moviendo hacia abajo? ¡Este es el sprite de Jugador!

Un sprite es como una "pequeña criatura de LED" que puedes controlar. ¿Qué pasará si presionas el botón amarillo (A)?

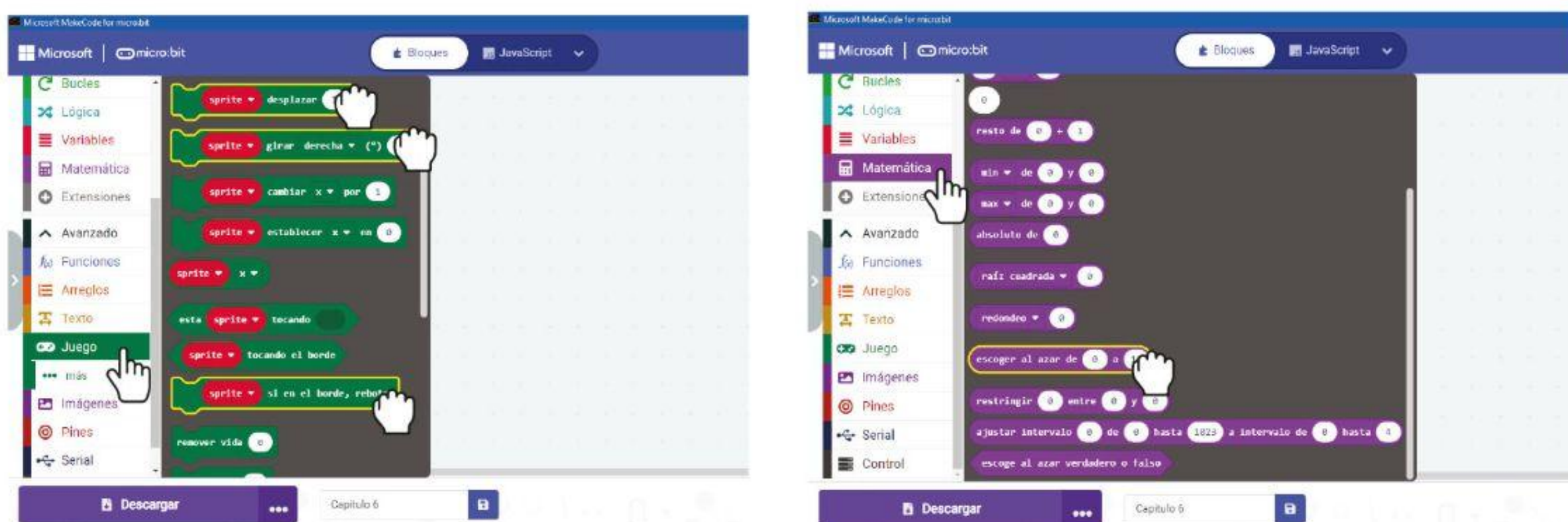


CAPÍTULO 6 : Corre que te pillo

Paso 9 Haz clic en la categoría **[Avanzado]** y selecciona la categoría **[Funciones]**, luego haz clic **[Crear una función...]**. En la ventana de edición de funciones, cambia el nombre de 'doSomething' a '**Juego**'. Luego haz clic **[Número]** para agregar un parámetro y cambia el nombre de 'num' por '**velocidad**' en el bloque de funciones. Luego haz clic en '**Listo**'.



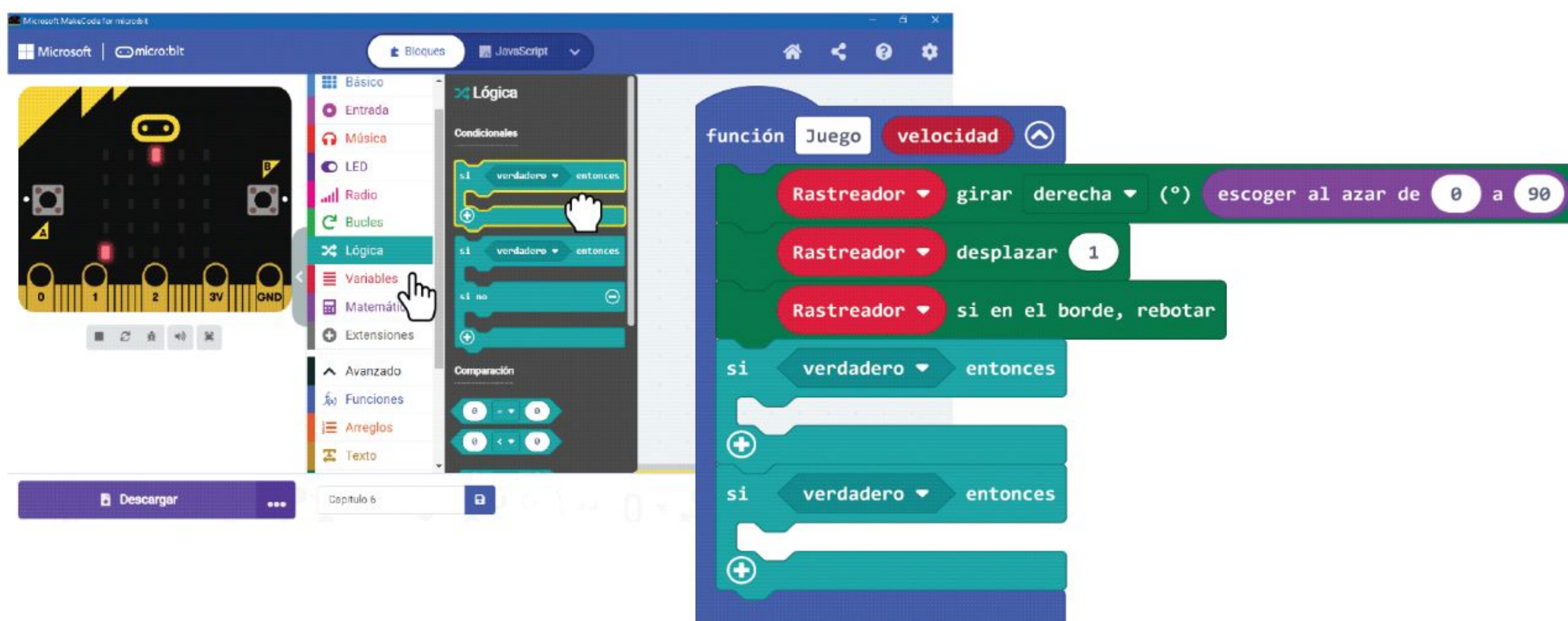
Paso 10 Continúa construyendo el código agregando bloques de las categorías **[Avanzado]** : **[Juego]** y **[Matemática]** Como se muestra abajo. Recuerda cambiar la variable '**Rastreador**' y el valor a **90**.



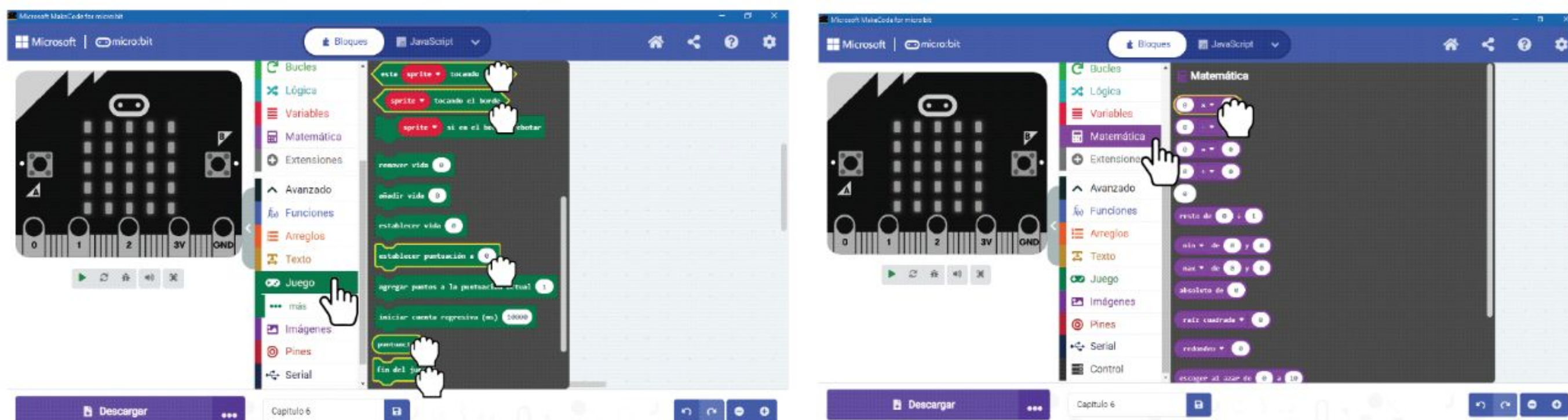
CAPÍTULO 6 : Corre que te pillo



Paso 11 Añade dos bloques **[si-entonces]** de la categoría **[Lógica]** a tu código.



Paso 12 Continúa construyendo el código agregando bloques de las categorías **[Avanzado]** : **[Juego]** y **[Matemática]** Como se muestra abajo.

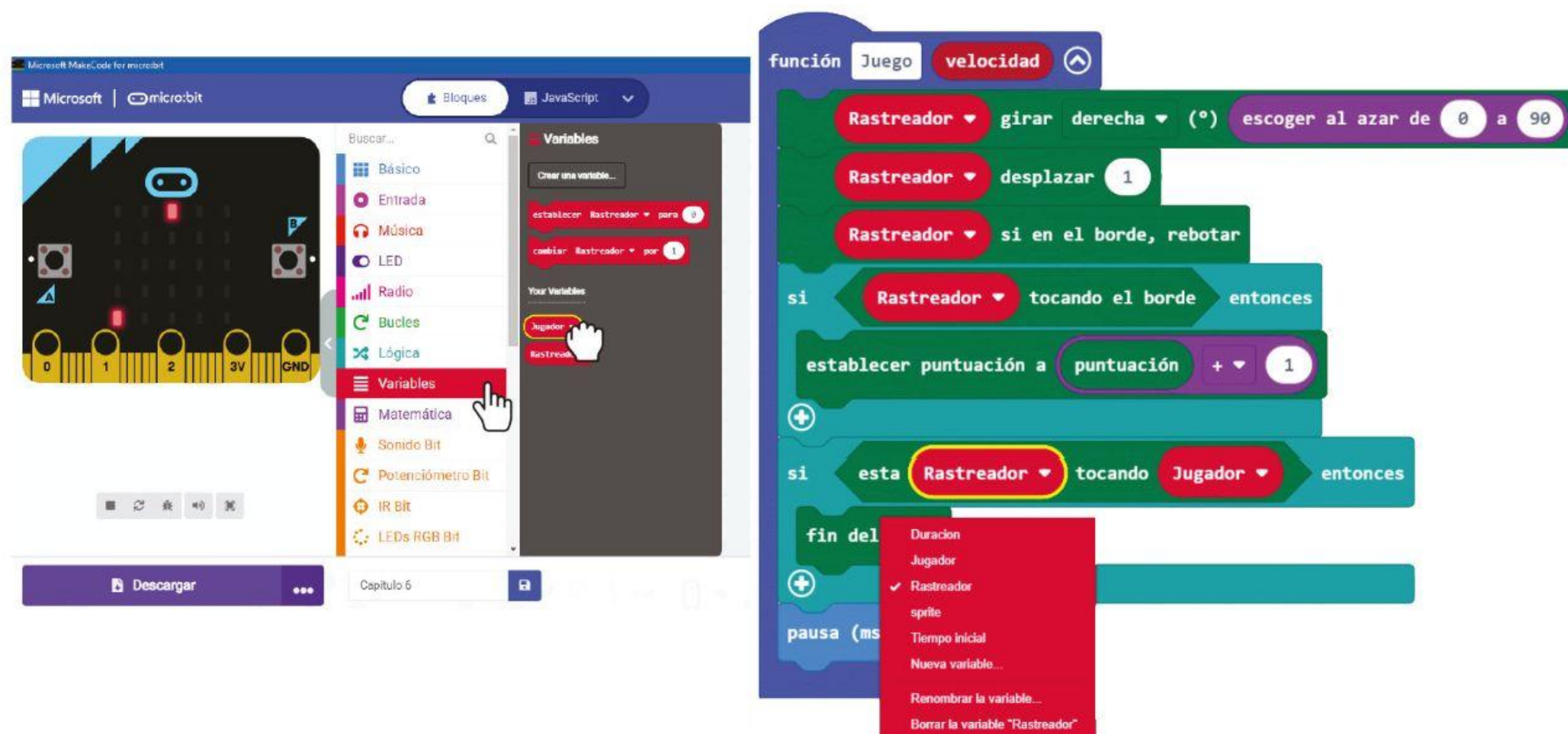


¡Seguid así!



CAPÍTULO 6 : Corre que te pillo

Paso 13 Cambia ambos bloques **[sprite]** a **'Rastreador'** haciendo clic en ellos y luego seleccionando **'Rastreador'**. Haz clic en la categoría **[Variables]** y selecciona el bloque **[Jugador]**. Colócalo en la ranura vacía del bloque. **[_ está tocando _]**.



Paso 14 Haz clic en la categoría **[Básico]** y selecciona el bloque **[pausa _ (ms)]**. Agrégalo al código. Haz clic en **[velocidad]** en el bloque de funciones, manten presionado y arrastra hacia la ranura vacía del bloque **[pausa _ (ms)]**.



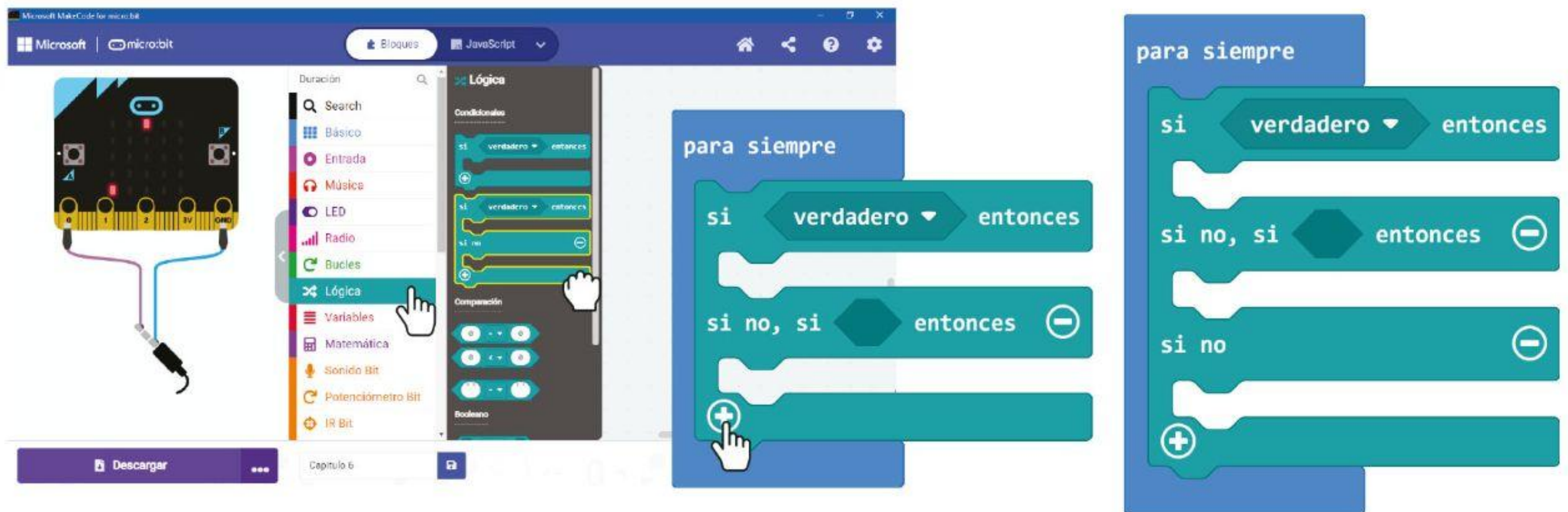
Puedes añadir vivacidad al juego tocando una melodía cuando el Rastreador toca al Jugador. ¿Puedes averiguar qué bloque agregar y dónde colocarlo?



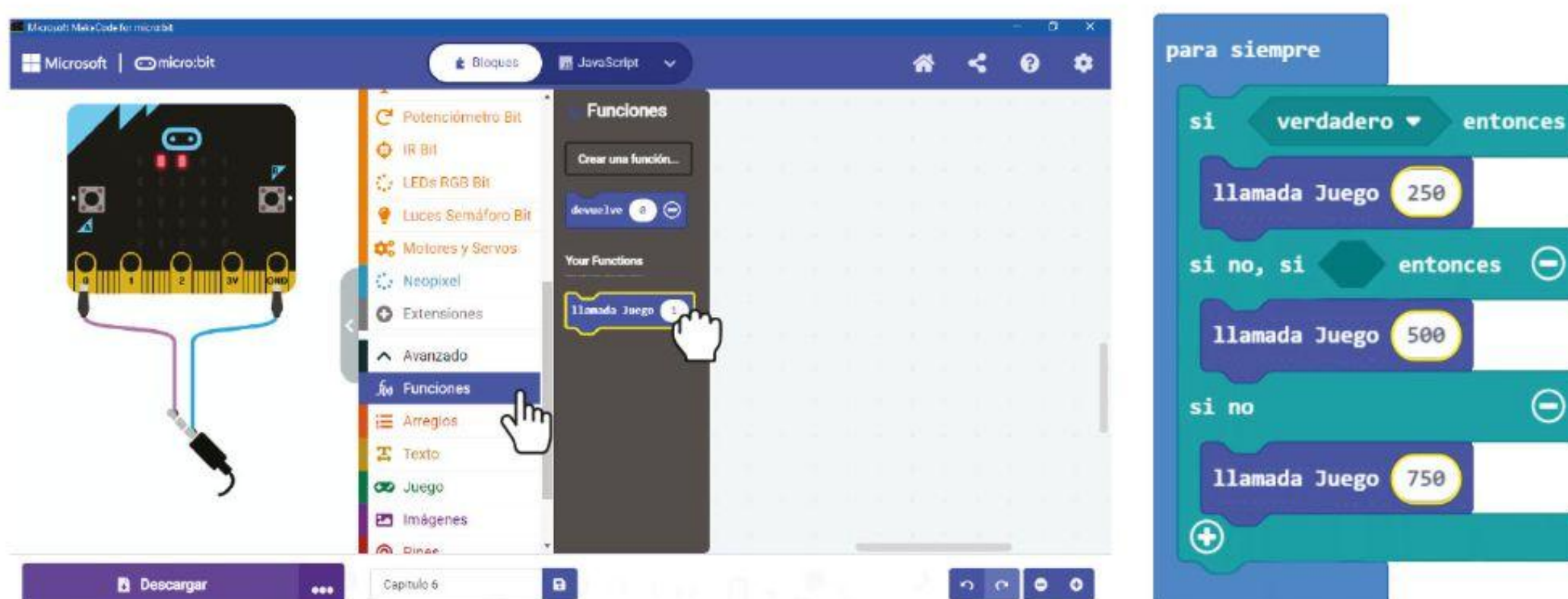


¡Ahora agreguemos diferentes niveles de dificultad al juego!

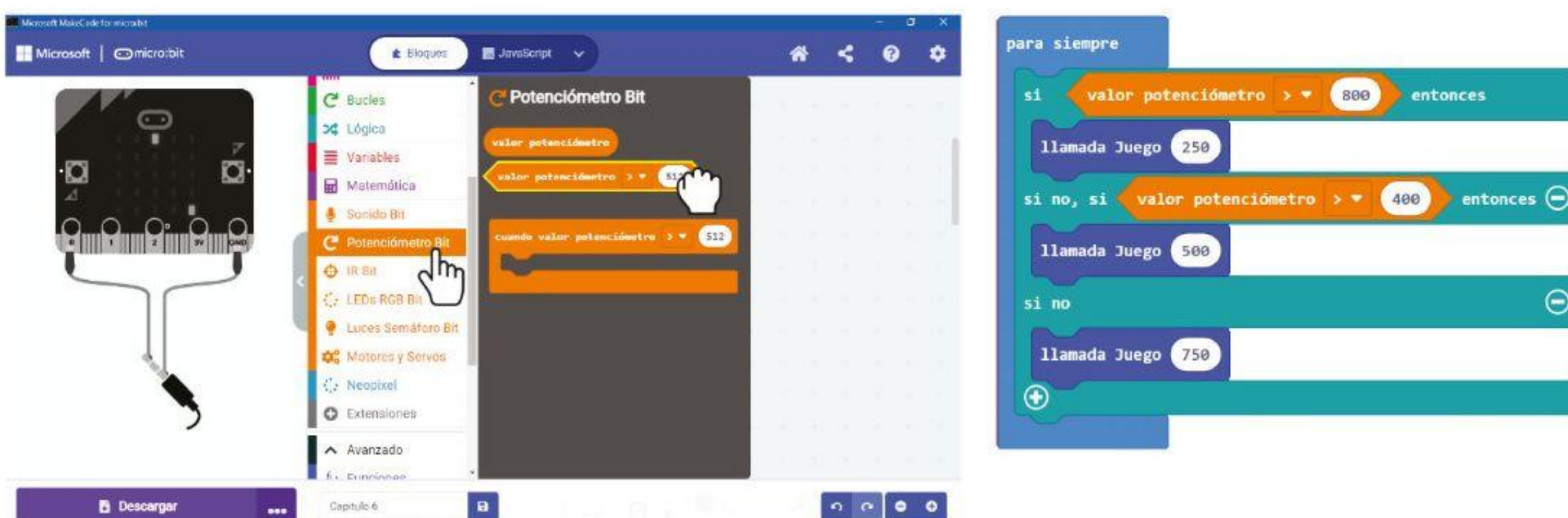
Paso 15 Haz clic en la categoría **[Lógica]** y selecciona el bloque **[si-entonces-si no]**. Coloca el candado en la ranura. **[para siempre]**. Haga clic en el botón (+) para agregar otra condición.



Paso 16 Haz clic en la categoría **[Funciones]** y selecciona el bloque **[llamada Juego _]**. Duplica el bloque y ajusta los bloques en cada una de las ranuras **[si-entonces-si no]**. Cambia el valor de los bloques. **[llamada Juego _]** a **250**, **500** y **750** respectivamente.



Paso 17 Haz clic en la categoría **[Potenciómetro Bit]** y selecciona el bloque **[valor potenciometro > _]**. Duplica y coloca bloques en las ranuras de condición de bloque **[si-entonces-si no]**. Establece el valor en **800** para el primer bloque y **400** para el segundo.



CAPÍTULO 6 : Corre que te pillo

Aquí está el código completo:

The image displays the complete Scratch code for the game 'Corre que te pillo'. The code is organized into several sections:

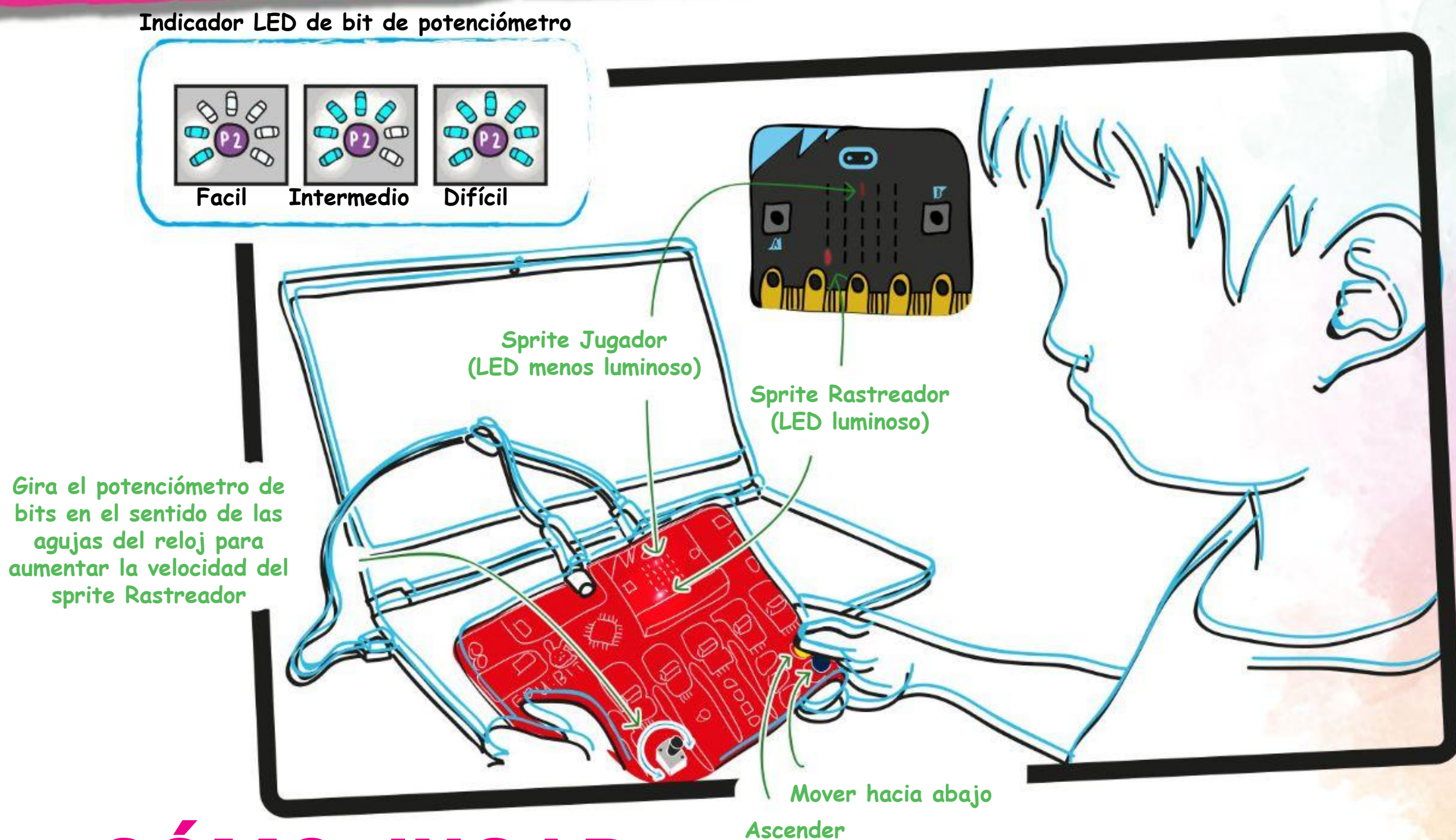
- Initialization (al iniciar):** Sets up a 30,000 ms countdown timer, creates the 'Rastreador' (piller) and 'Jugador' (runner) sprites at specific coordinates, rotates the runner 90 degrees, and sets its brightness to 50.
- Button Controls:** Two event-driven blocks for buttons A and B. Button A moves the runner down (-1), and Button B moves the runner up (1).
- Game Function (función Juego):** A custom function that takes a 'velocidad' (speed) parameter. It rotates the piller randomly, moves it 1 step, and checks for collisions with the runner. If a collision occurs, it triggers the 'fin del juego' (game over) event and plays the 'wawawawaaa' sound.
- Game Loop (para siempre):** A 'forever' loop that checks the potentiometer's value. If it's above 800, it calls the 'Juego' function with a speed of 250 ms. If it's between 400 and 800, it calls the function with a speed of 500 ms. If it's below 400, it calls the function with a speed of 750 ms.

The code is presented in a visual format with color-coded blocks and a corresponding list of instructions on the left side of the image.

Paso 18 Transfiere el código completo a tu EDU:BIT y disfruta jugando Corre que te pillo con tus amigos.

Vamos a jugar

Corre que te pillo



CÓMO JUGAR:

Cuando está encendido, el sprite Rastreador continuará moviéndose en direcciones aleatorias.

Mueve el objeto Jugador hacia arriba o hacia abajo para evitar el objeto Rastreador. Presiona el botón amarillo (botón A) para subir y el botón azul (botón B) para bajar.

Fin del juego si el jugador es "tocado" por el Rastreador, o después del límite de tiempo de 30 segundos.

Cada vez que el sprite de Rastreador "toca" el borde, ganas 1 punto. ¡El Jugador con la puntuación más alta es el ganador! Disfruta !

¡AYUDA!

1 Para obtener una puntuación más alta dentro del límite de 30 segundos, puedes aumentar la velocidad del sprite Rastreador para que "toque" el borde más grueso.
2 Después de que termine el juego, puedes presionar los botones A + B al mismo tiempo para comenzar un nuevo juego. Esta es una función integrada en los bloques [Juego].

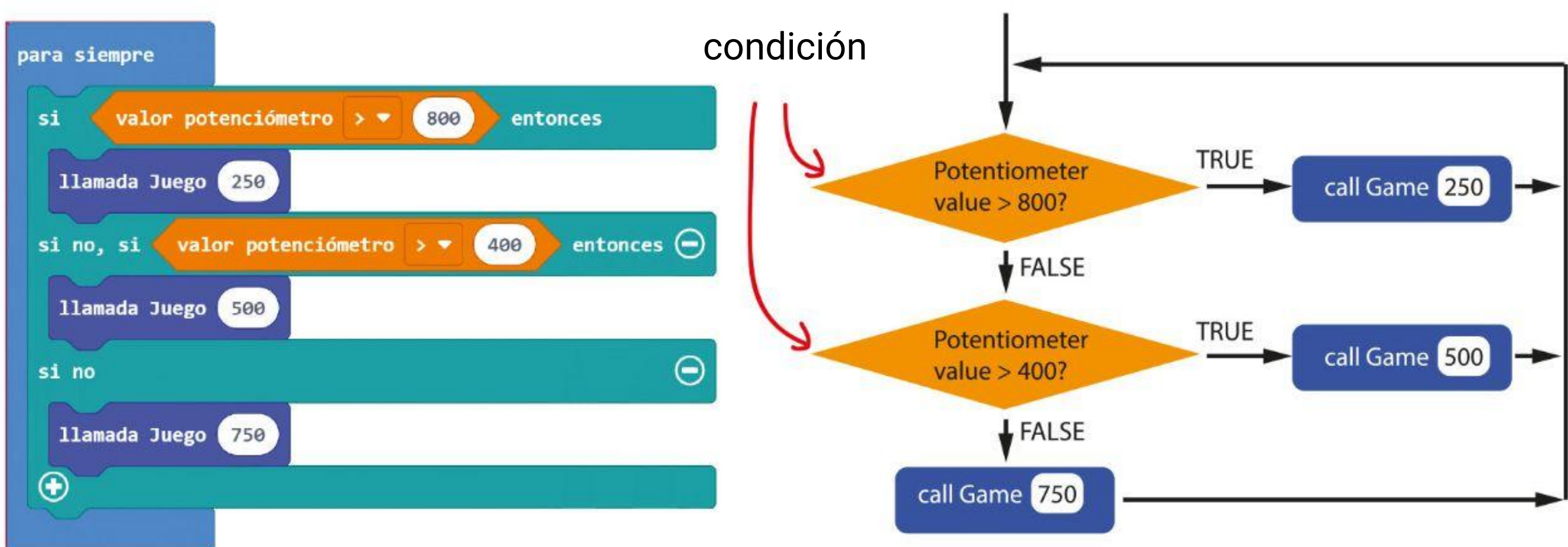
MODIFICAR EL CÓDIGO

En programación, las **expresiones condicionales** se utilizan para tomar decisiones. En MakeCode, usamos los bloques condicionales [si-entonces] o [si-entonces-de lo contrario] de la categoría **[Lógica]**. El programa verifica la instrucción con la condición y si es VERDADERO, ejecuta el código en el bloque dentro de la condición. De lo contrario, si es FALSO, se mueve al siguiente bloque de código.



Cuando tenemos varias condiciones, el programa evaluará las condiciones en secuencia, de arriba a abajo, y ejecutará el código correspondiente a la primera condición que sea VERDADERA. De esa manera, cuanto mayor sea la condición, mayor será su prioridad sobre los de abajo.

Por ejemplo, este código del juego determina la velocidad de movimiento del objeto Rastreador comparando el valor del potenciómetro con umbrales predeterminados.



si valor del potenciómetro > 800, llama a la función Juego (con la velocidad variable a 250 ms),
si no si valor del potenciómetro > 400, llame a la función JJuego (velocidad = 500 ms),
si no llamar a la función Juego (velocidad = 750 ms)



EXPLORA OTROS BLOQUES

#1 Usa el bloque **[Básico]:[mostrar número]** con el bloque **[valor potenciómetro]** para leer y mostrar la posición actual del potenciómetro.

mostrar número valor potenciómetro

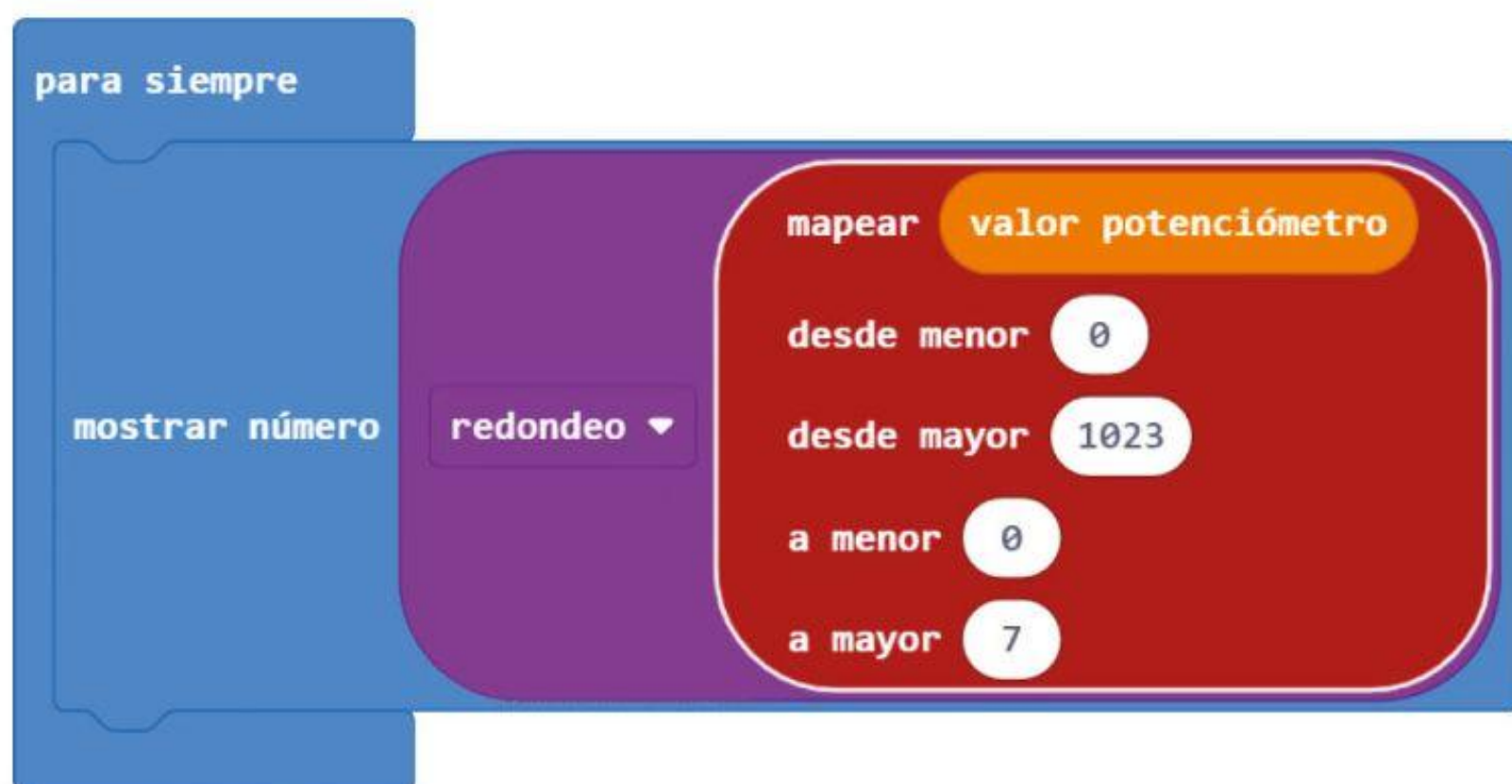
#2 El potenciómetro devuelve un valor entre 0 y 1023. Puedes usar el bloque **[mapear _ desde menor _ a mayor _ a menor_ a mayor_]** de la categoría **[Avanzado]:[Pines]** para mapear la lectura a un rango que sea más significativo.



#3 El bloque que mapear devolverá un número con dígitos decimales (por ejemplo, 1,68 o 3,998). Para redondear el número usa el bloque **[redondeo _]** de la categoría **[Matemática]**.

redondeo 0

Aquí hay un código de ejemplo para mapear la lectura del bit del potenciómetro en un rango de 0 a 7. El valor se redondea y se muestra en la matriz de LED.



Recuerda activar EDU:
BIT para obtener
lecturas precisas.

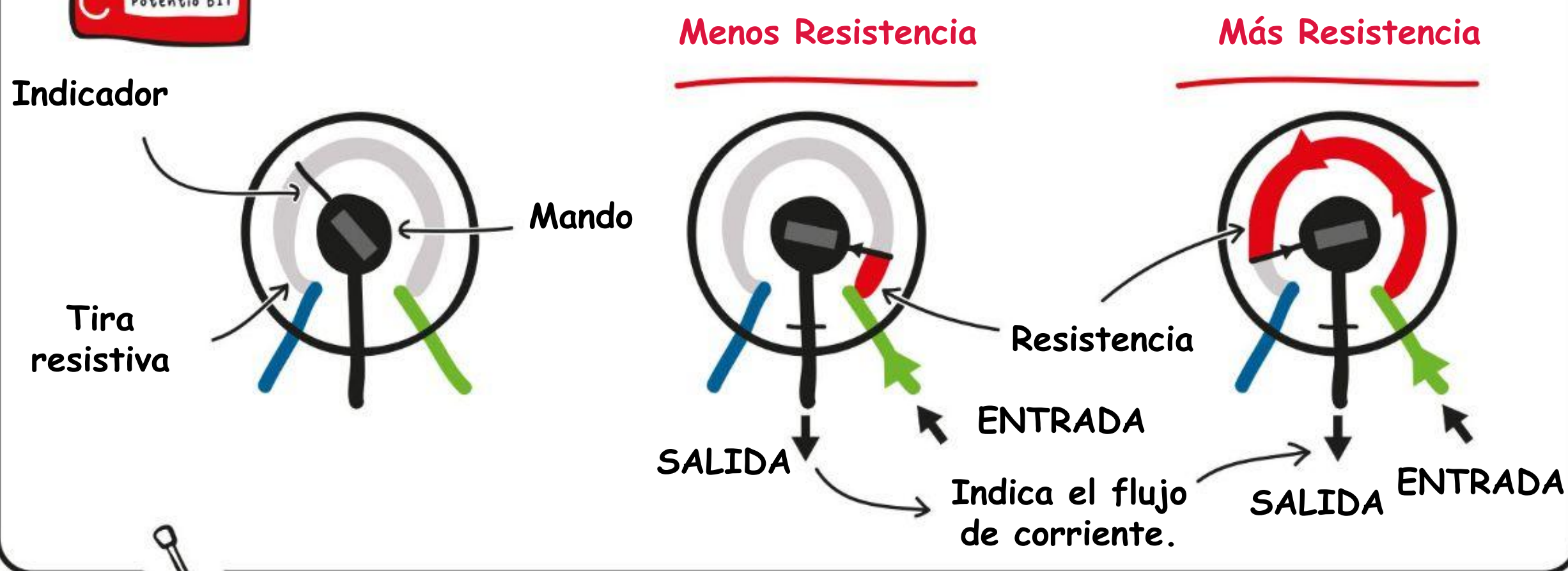


¡COSAS DIVERTIDAS!



Los **potenciómetros** son resistencias variables con una resistencia que se puede ajustar fácilmente con una mando giratorio o un control deslizante.

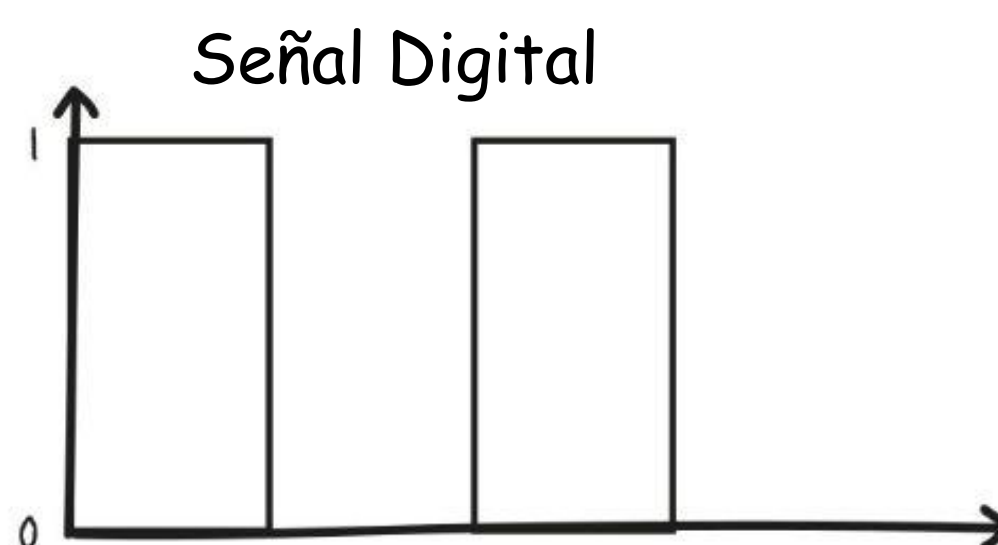
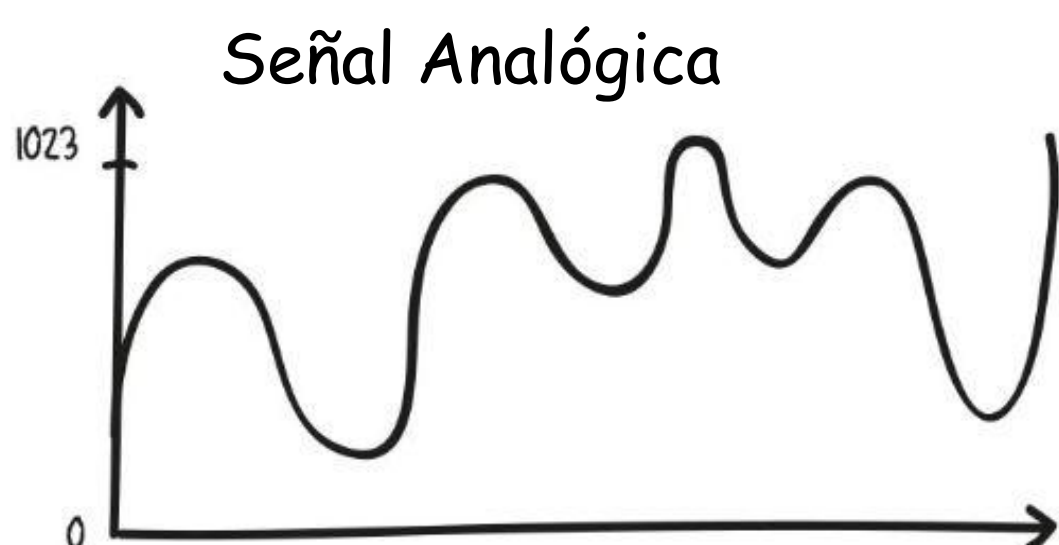
Si tienes un potenciómetro de $10.000\ \Omega$, puedes obtener un valor de resistencia entre $0\ \Omega$ y $10\ 000\ \Omega$ cambiando la posición del indicador en el mando giratorio.



Aquí hay algunos usos comunes:

- Volumen de audio de un altavoz
- Comprobación de la frecuencia de una radio
- Comprobación de la temperatura de un calentador de agua

El potenciómetro en EDU: BIT es un dispositivo de entrada de tipo analógico. Mide el potencial eléctrico y convierte el voltaje medido (entre $0V$ y $3.3V$) en un número entero entre 0 y 1023 .



DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para que funcione como temporizador. Utiliza el Potenciómetro para ajustar la duración (entre 0 y 60 segundos), el botón A para activarlo y el botón B para resetearlo.	
En el inicio	Establecer modo 0
Cuando presionas el botón A (amarillo)	Establecer el modo 1 Establece hora inicial Mostrar cara sonriente
Cuando presionas el botón B (azul)	Establecer modo 0
Siempre	<div>Comprueba siempre el modo</div> <ul style="list-style-type: none">• Modo IF = 0, luego configure la Duración al valor redondeado del potenciómetro mapeado entre 0 y 60 y muestre la Duración en la matriz de LED.• DE LO CONTRARIO SI modo = 1, comprobar si (hora actual - Hora inicial) > (Duración x 1000). Si es VERDADERO, reproduce la melodía wawawawaa y establece el modo 2.• De lo CONTRARIO, muestra una cara triste.

Aquí tienes algunos consejos:

- # 1 Debes crear tres variables: Modo, Hora de inicio y Duración.*
- # 2 El bloque de tiempo actual (ms) está en la categoría [Entradas].*
- # 3 Usa esta declaración condicional para verificar si se acabó el tiempo.*



tiempo de ejecución (ms)

- ▼

Tiempo inicial ▼

≥ ▼

Duracion ▼

× ▼

1000

CAPÍTULO 7

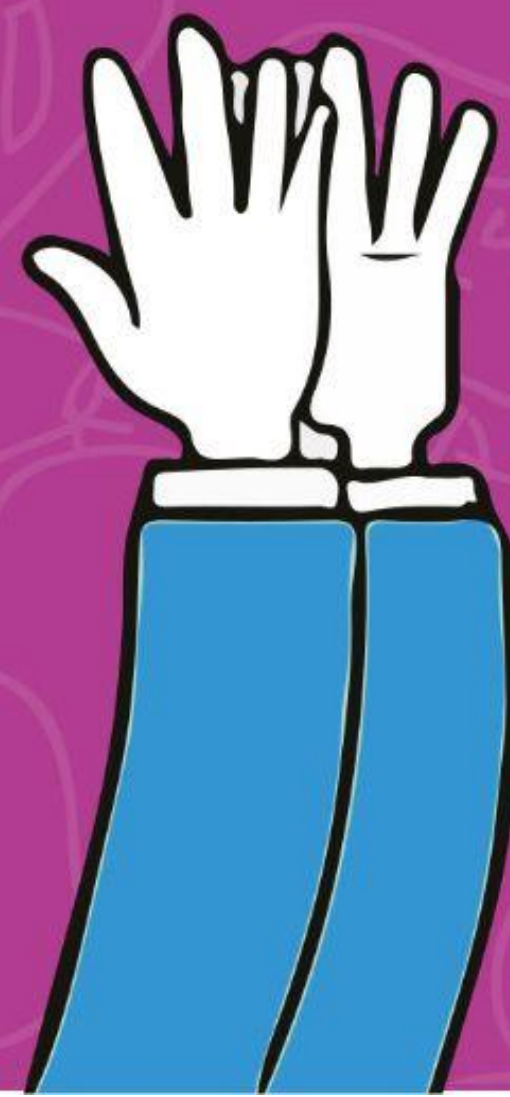
¡Escuchemos los aplausos!

Sonido Bit



¡Escanéame!

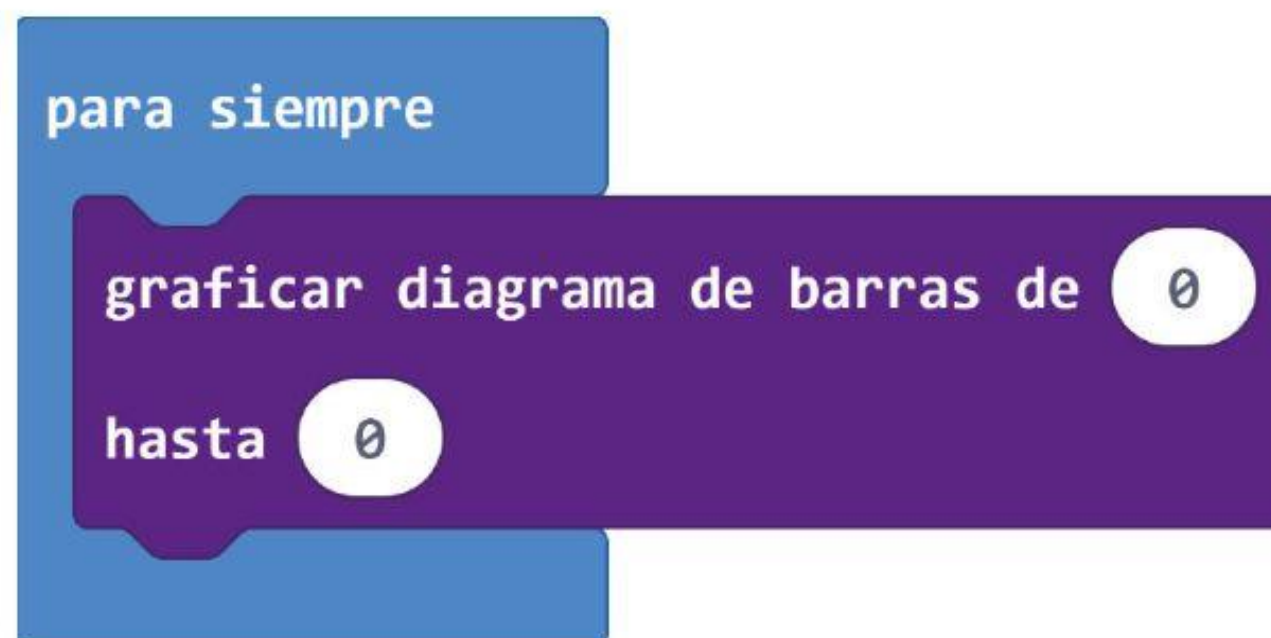
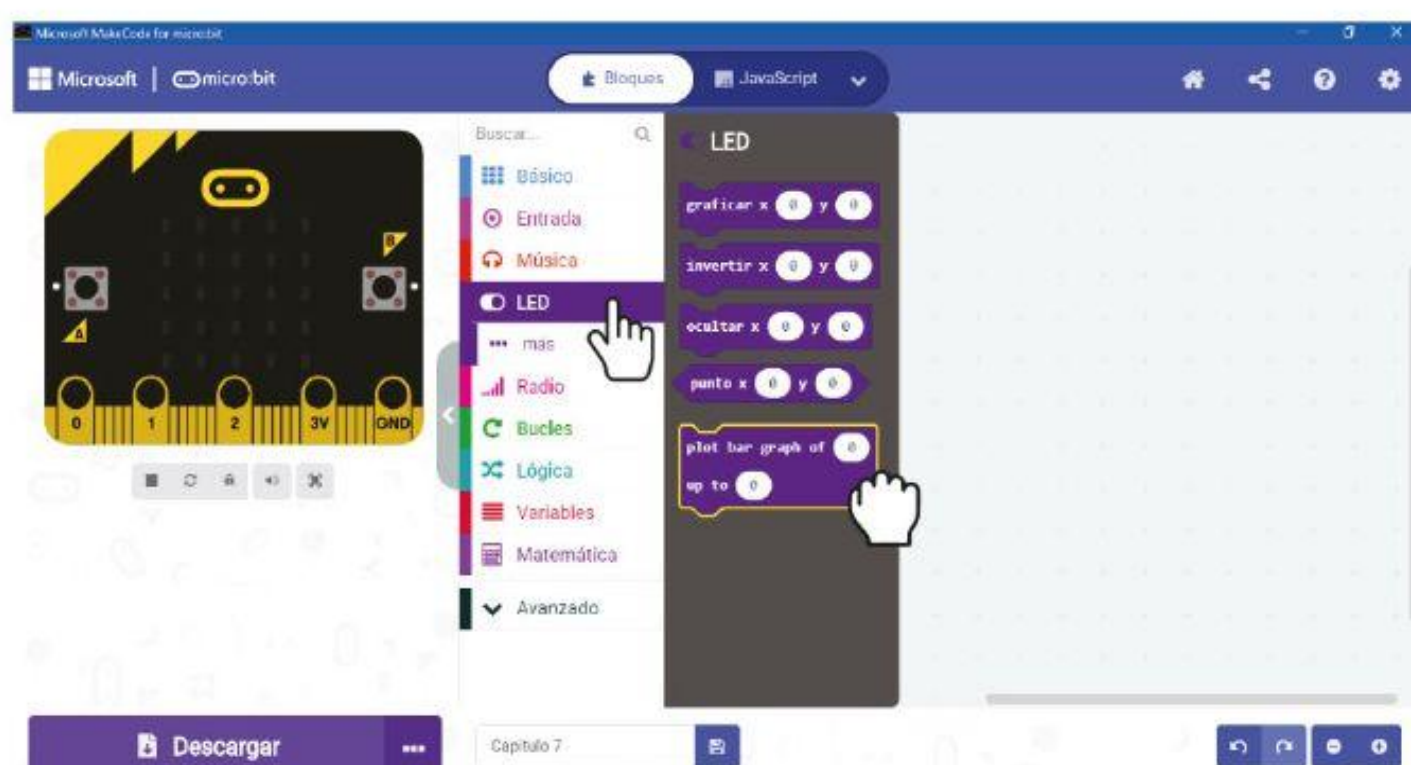
link.cytron.io/edubit-chapter-7



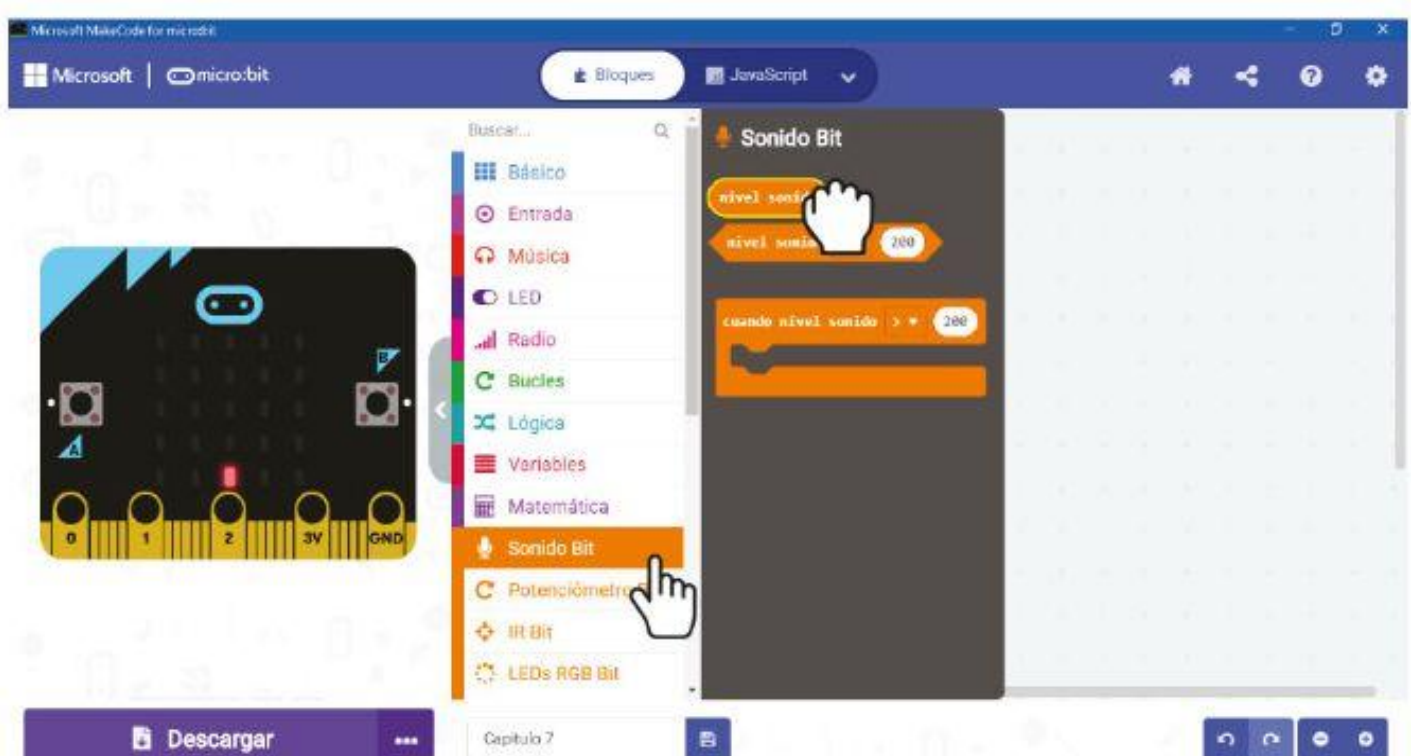


PROGRAMAMOS!

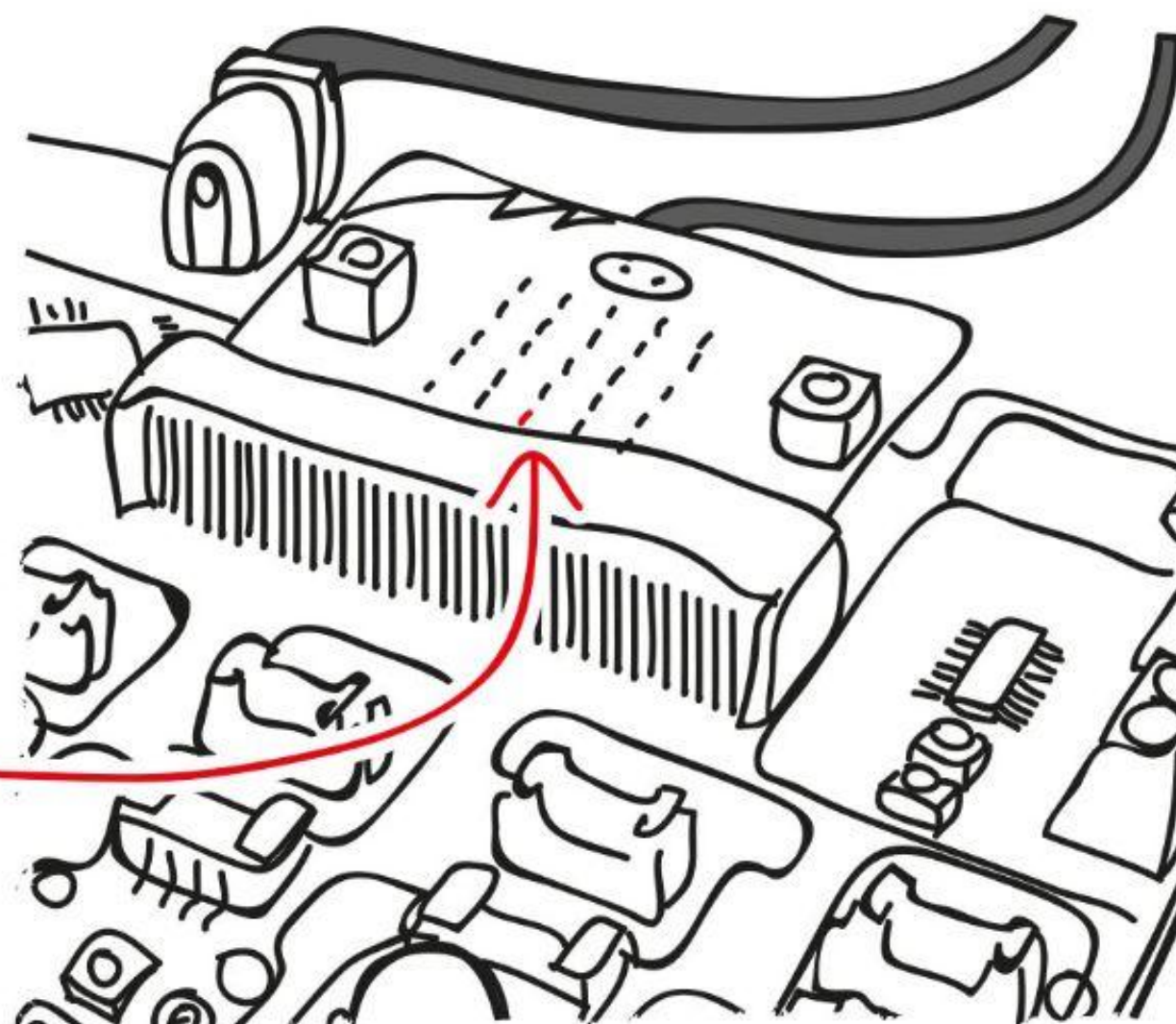
Paso 1 Crea un nuevo proyecto en el Editor de MakeCode y agrega la extensión EDU: BIT (puedes consultar la página 40). Haz clic en la categoría **[Led]** y selecciona el bloque **[graficar diagrama de barras de _ hasta _]**. Encaja el bloque en la ranura **[para siempre]**.



Paso 2 Haz clic en la categoría **[Sonido Bit]** y selecciona el bloque **[nivel sonido]**. Encaja el bloque en el bloque **[graficar diagrama de barras de _ hasta _]** y cambia el segundo valor 0 a 1023.



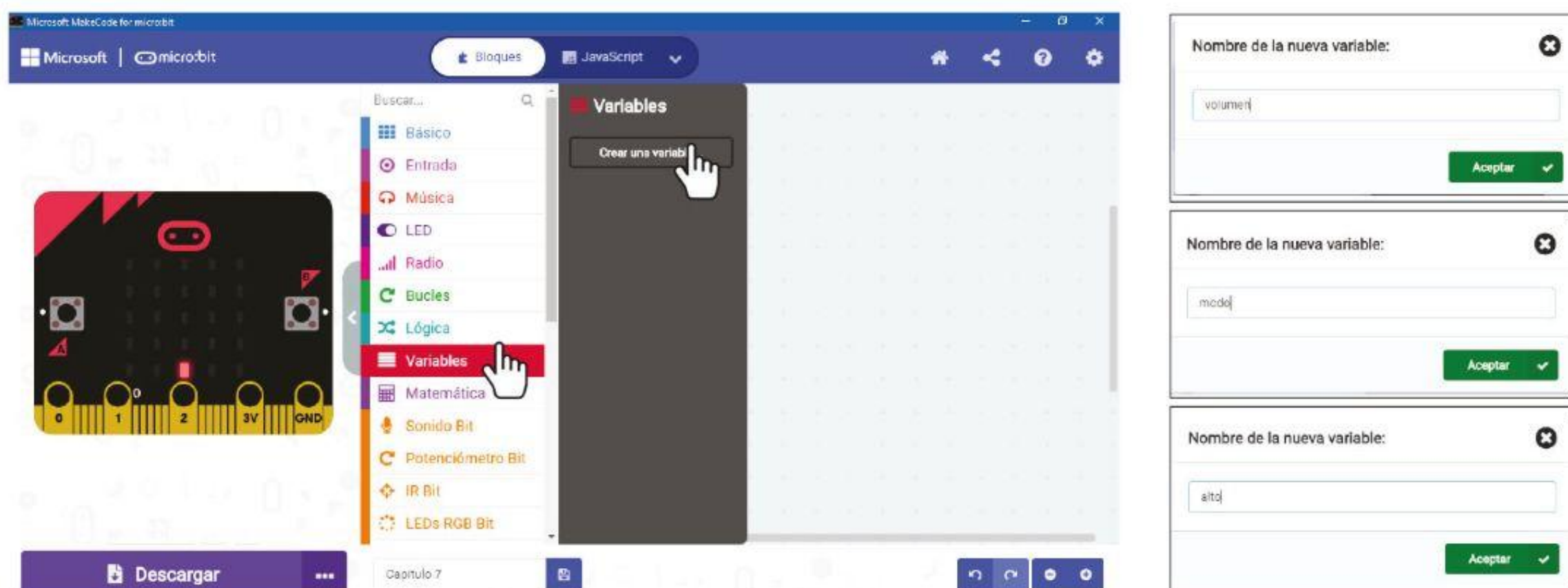
Paso 3 Transfiere el código a tu EDU: BIT. Observa la pantalla de matriz de LED mientras aplaudes o tamborileas con los dedos sobre la mesa.



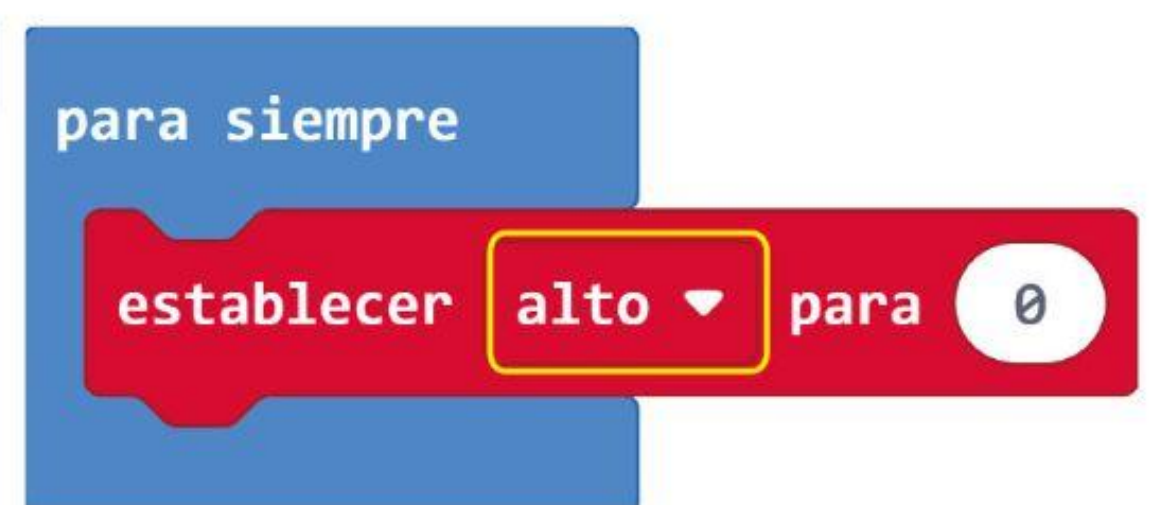
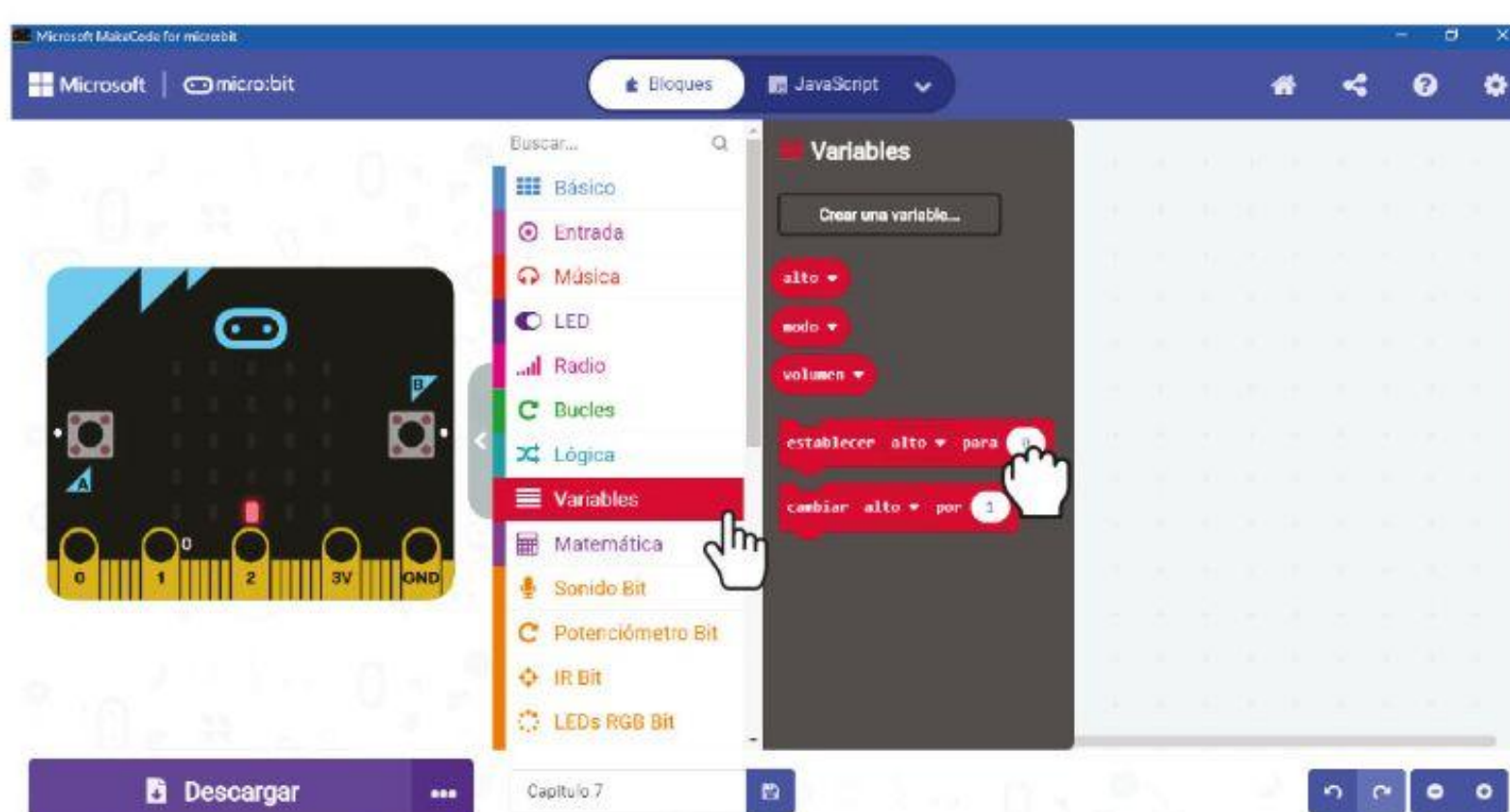


Ahora vamos a convertir EDU: BIT en un contador de aplausos.

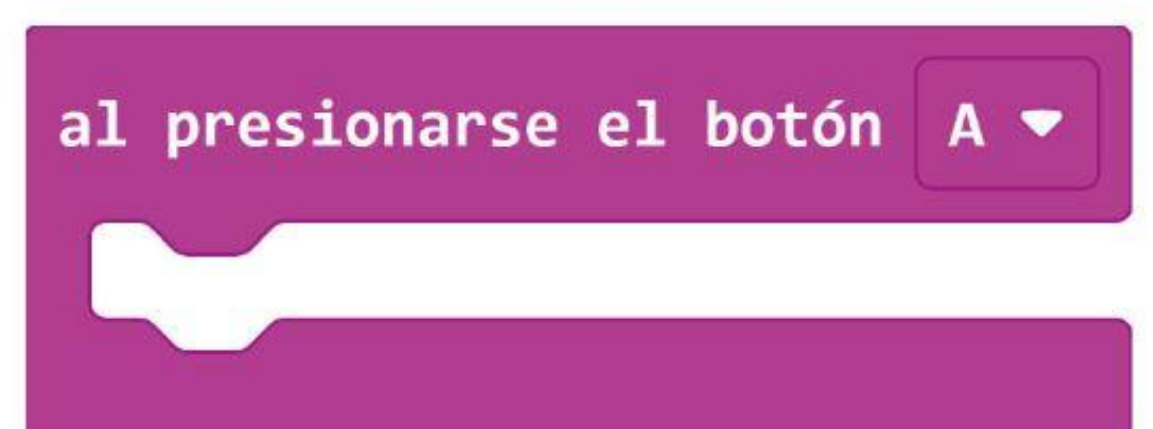
Paso 4 Inicia un nuevo proyecto y agrega la extensión EDU: BIT. Haz clic en la categoría **[Variables]** y luego haz clic **[Crear una variable]**. Escribe 'modo' en la ventana emergente y luego haz clic en Aceptar. Crea otras dos variables llamadas 'volumen' y 'alto'.



Paso 5 selecciona el bloque **[establecer _ para _]** de la categoría **[Variables]**. Encaja el bloque en la ranura **[al iniciar]**. Establece la variable en 'modo'.



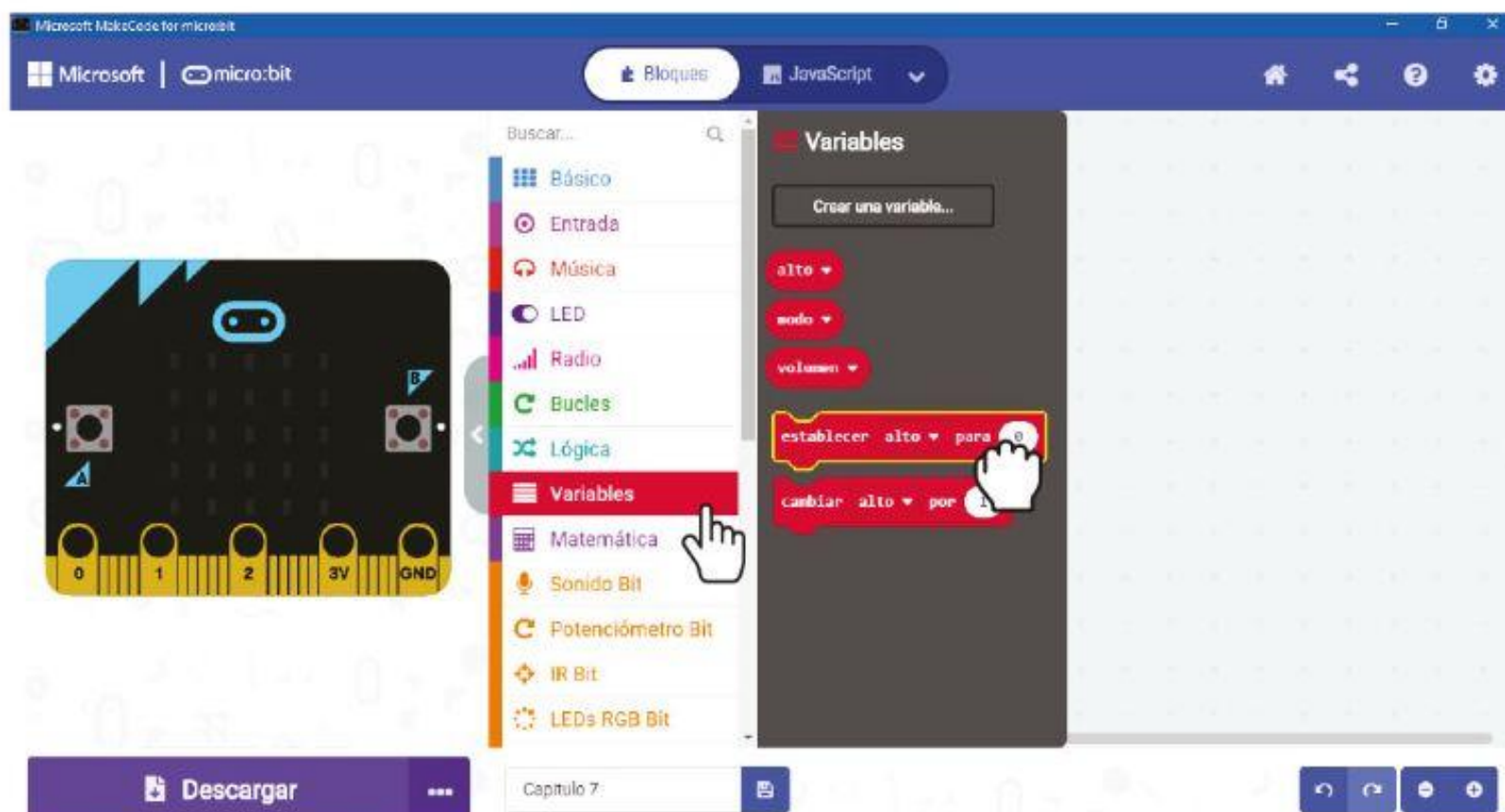
Paso 6 Haz clic en la categoría **[Entrada]** y luego selecciona el bloque **[al presionar el botón _]**.



CAPÍTULO 7 : ¡Escuchemos los aplausos!



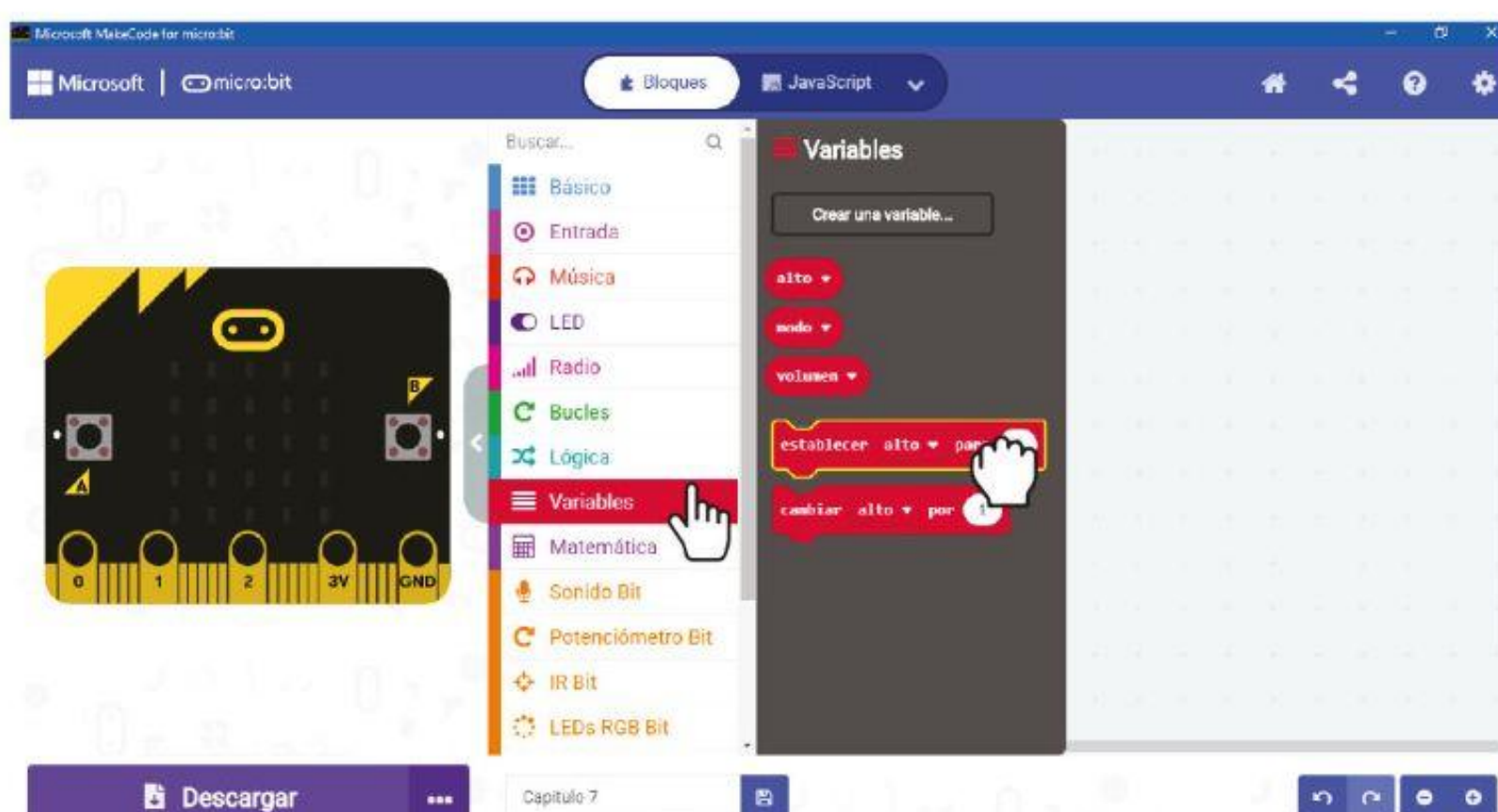
Paso 7 Toma dos bloques **[establece _ para _]** de la categoría **[Variables]** y encaja los bloques en la ranura **[al presionar el botón _]**. Establece la primera variable en 'modo' y el valor a 1, y la segunda variable a 'alto' con el valor predeterminado 0.



Paso 8 Haz clic en la categoría **[IR Bit]** y selecciona el bloque **[cuando IR activado]**.

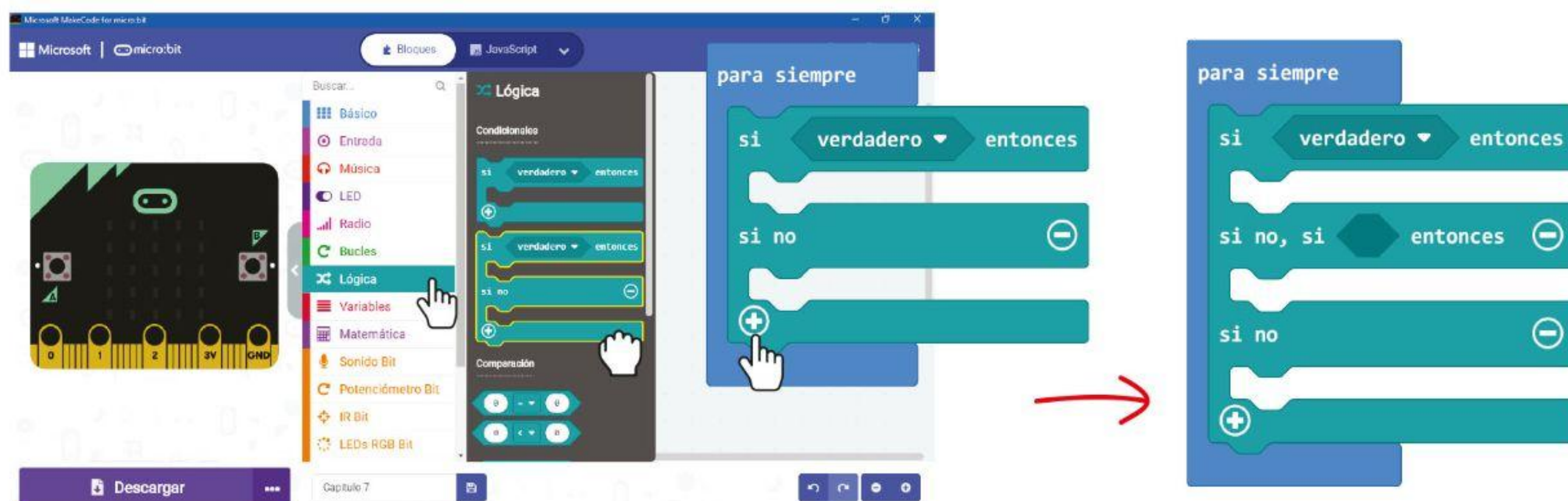


Paso 9 Haz clic en la categoría **[Variables]** y luego selecciona el bloque **[establecer _ para _]**. Encaja el bloque en la ranura **[cuando IR activado]** y cambia la variable a 'modo' y el valor a 2.



CAPÍTULO 7 : ¡Escuchemos los aplausos!

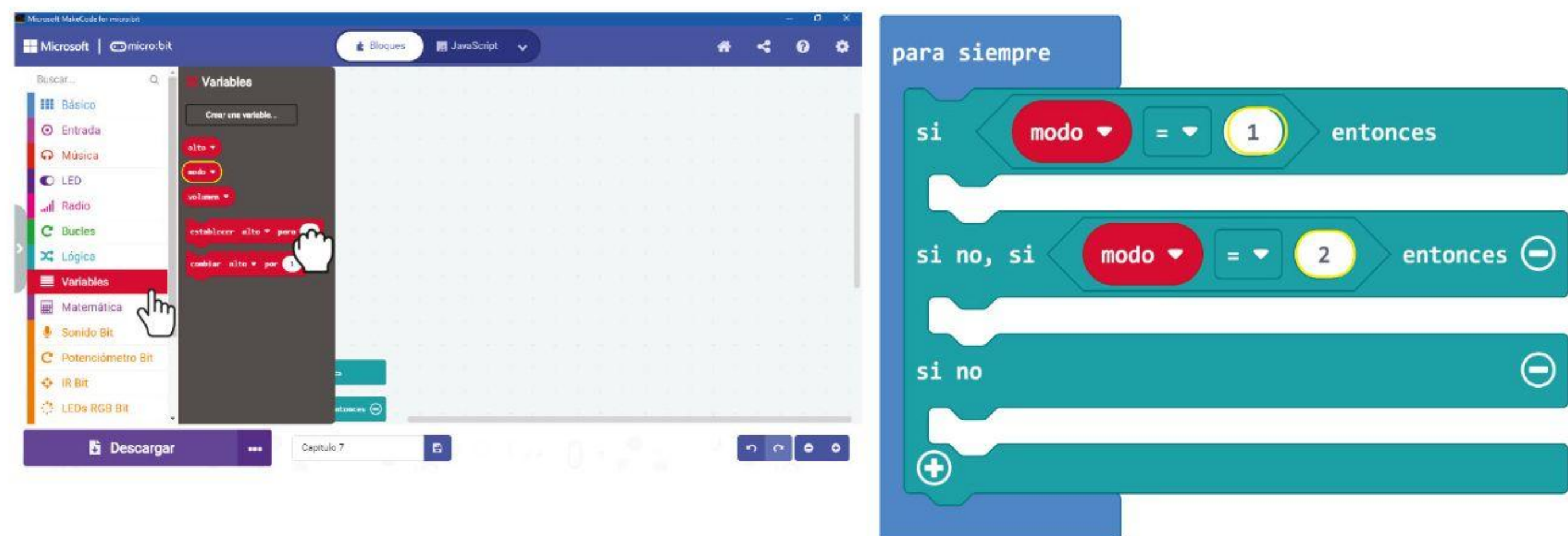
Paso 10 Haz clic en la categoría **[Lógica]** y luego selecciona el bloque **[si-entonces-si no]**. Encaja el bloque en la ranura **[para siempre]**. Haz clic en el ícono más (+) para agregar otra condición de lo contrario al bloque.



Paso 11 Haz clic en la categoría **[Lógica]** y luego selecciona el bloque comparación **[=]**. Duplica el bloque y encaja los bloques en las ranuras del bloque **[si-entonces-si no]**.



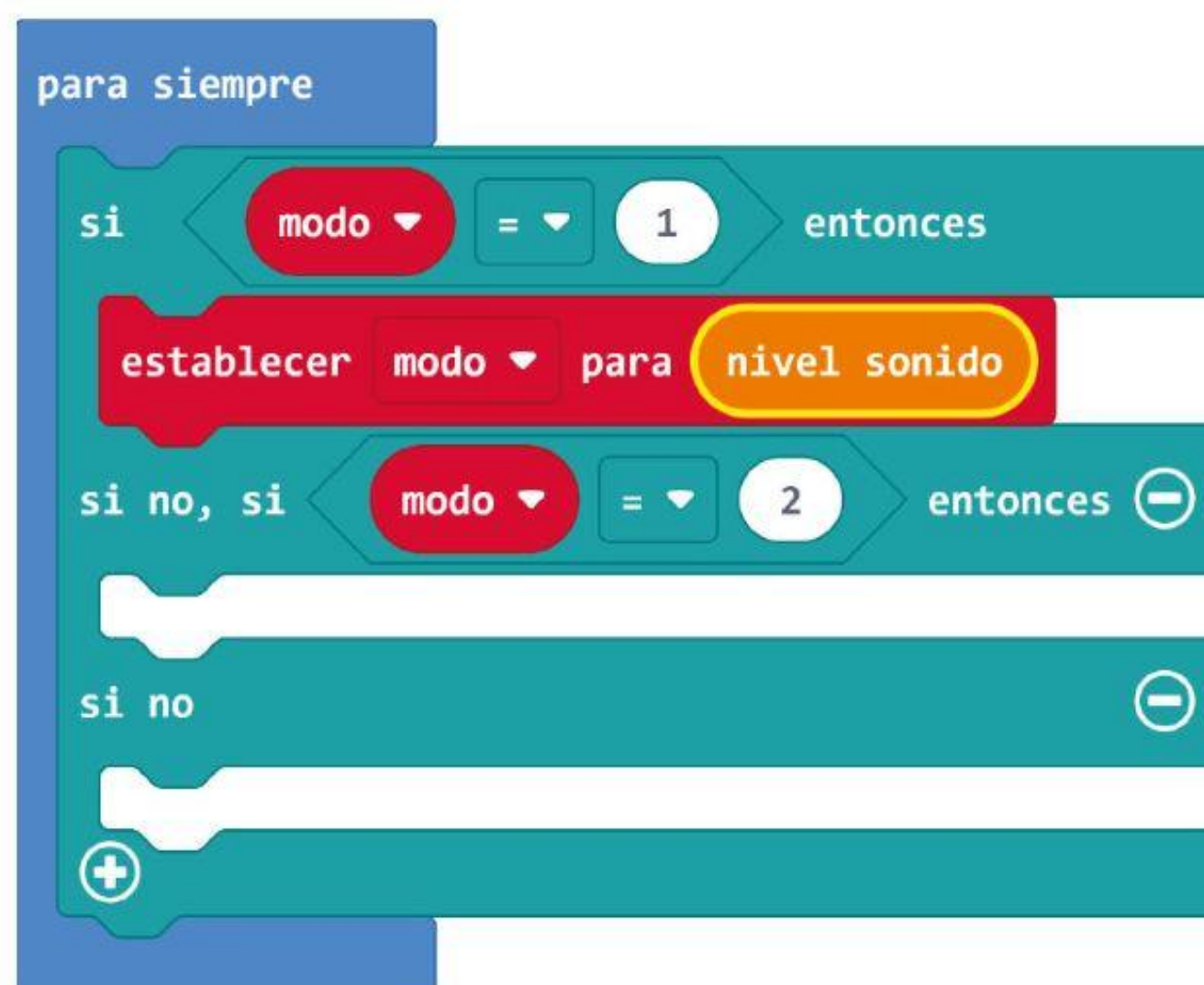
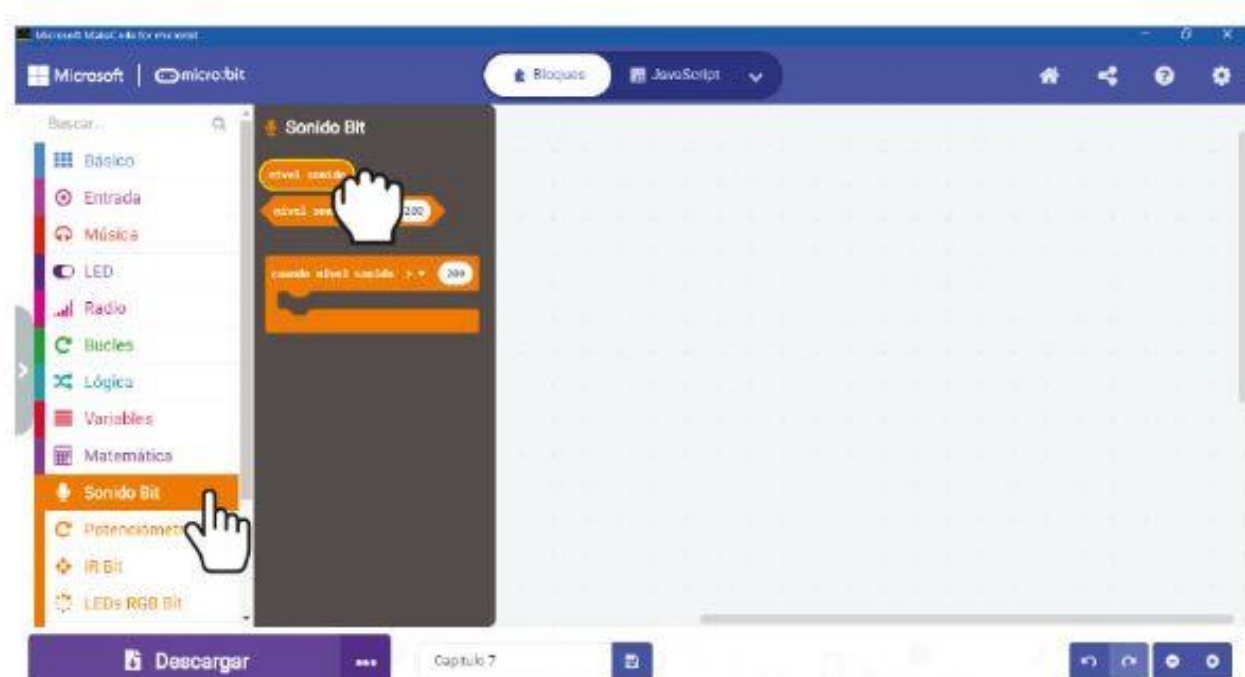
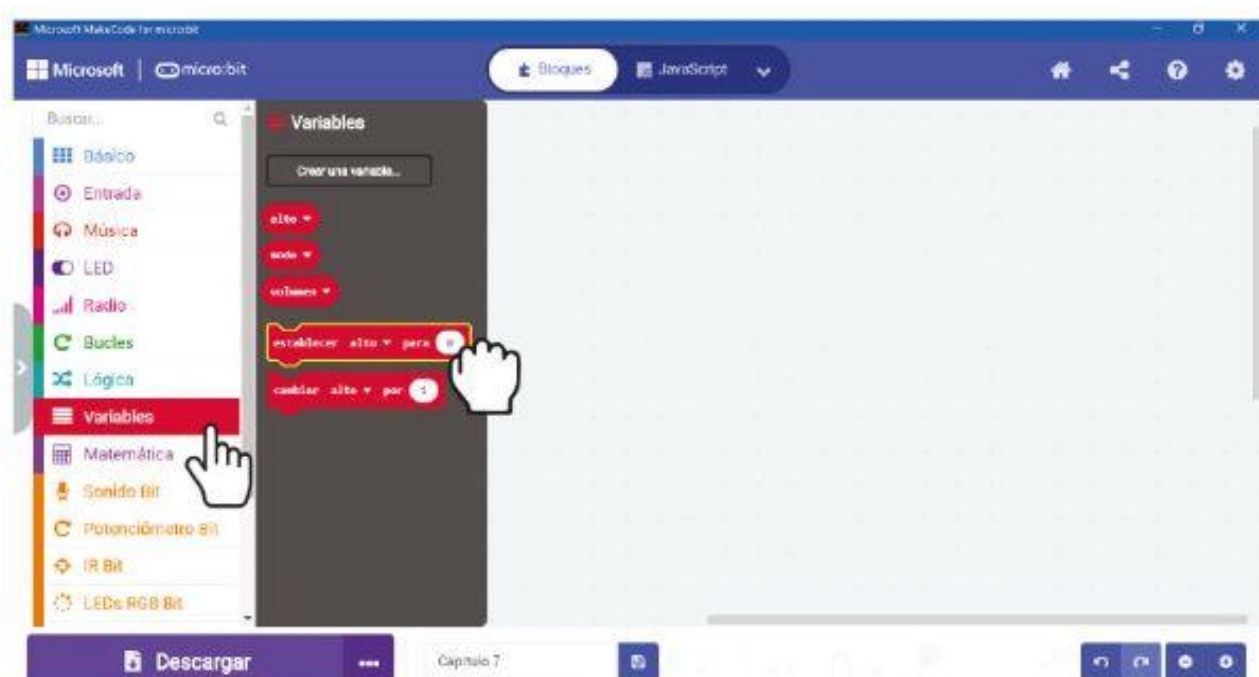
Paso 12 Toma **[modo]** de la categoría **[Variables]** y colócalo en la ranura a la izquierda de los bloques de comparación. Establece las otras ranuras en 1 y 2 respectivamente.



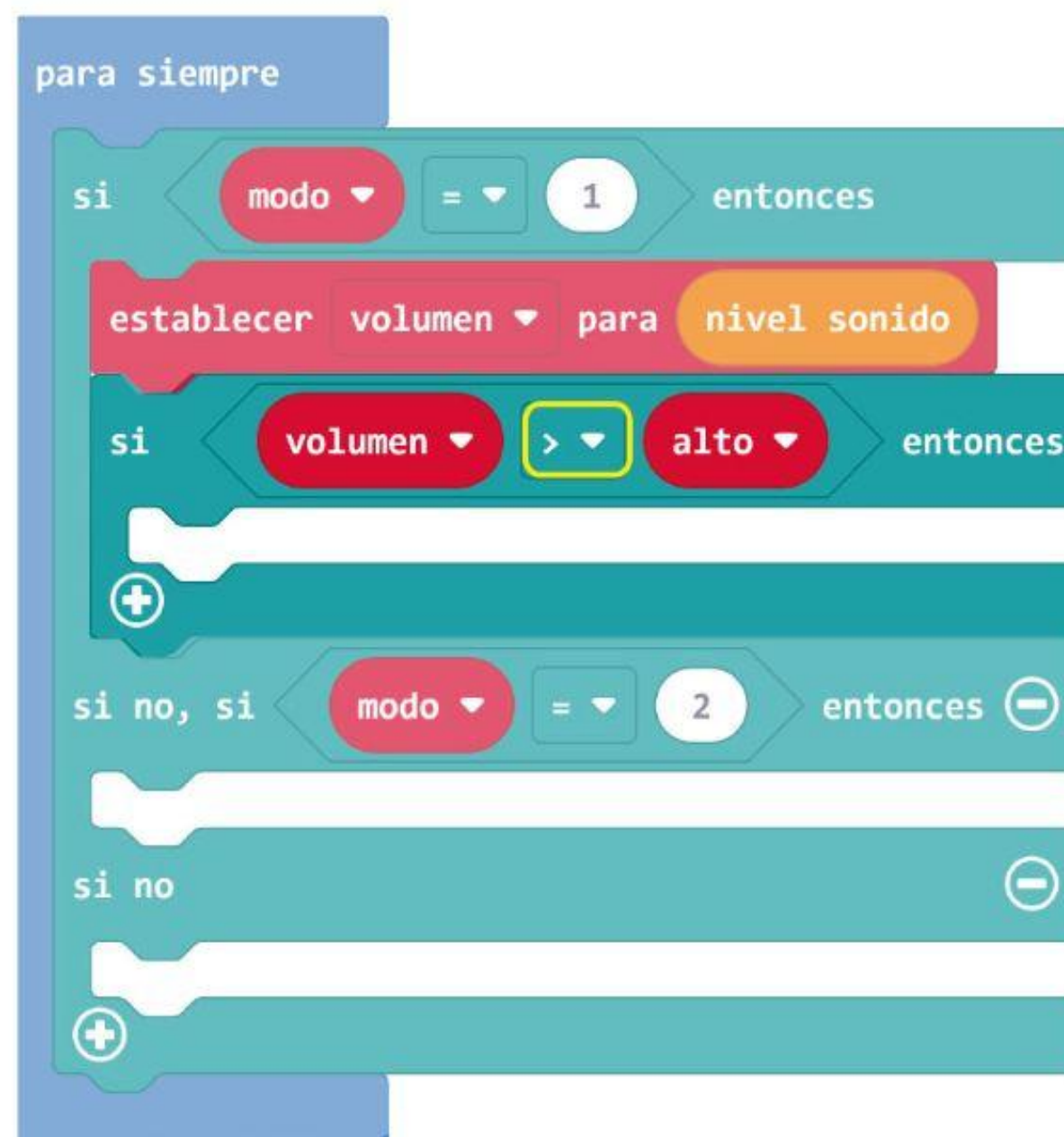
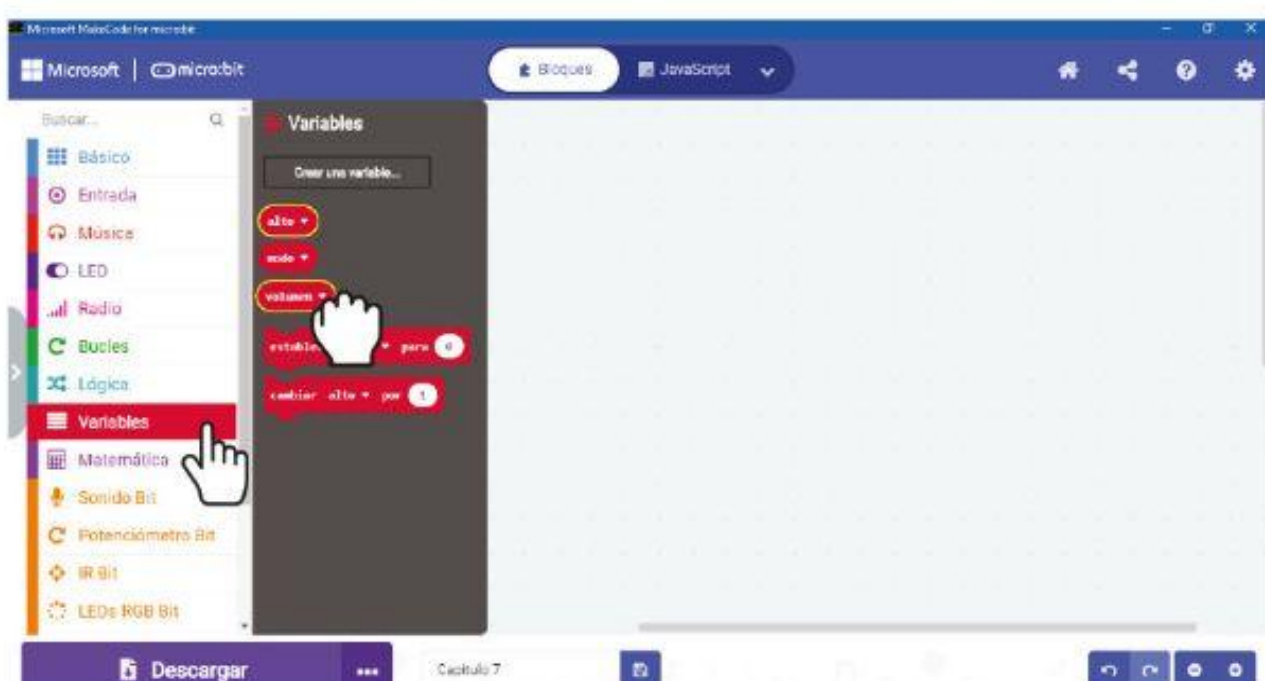
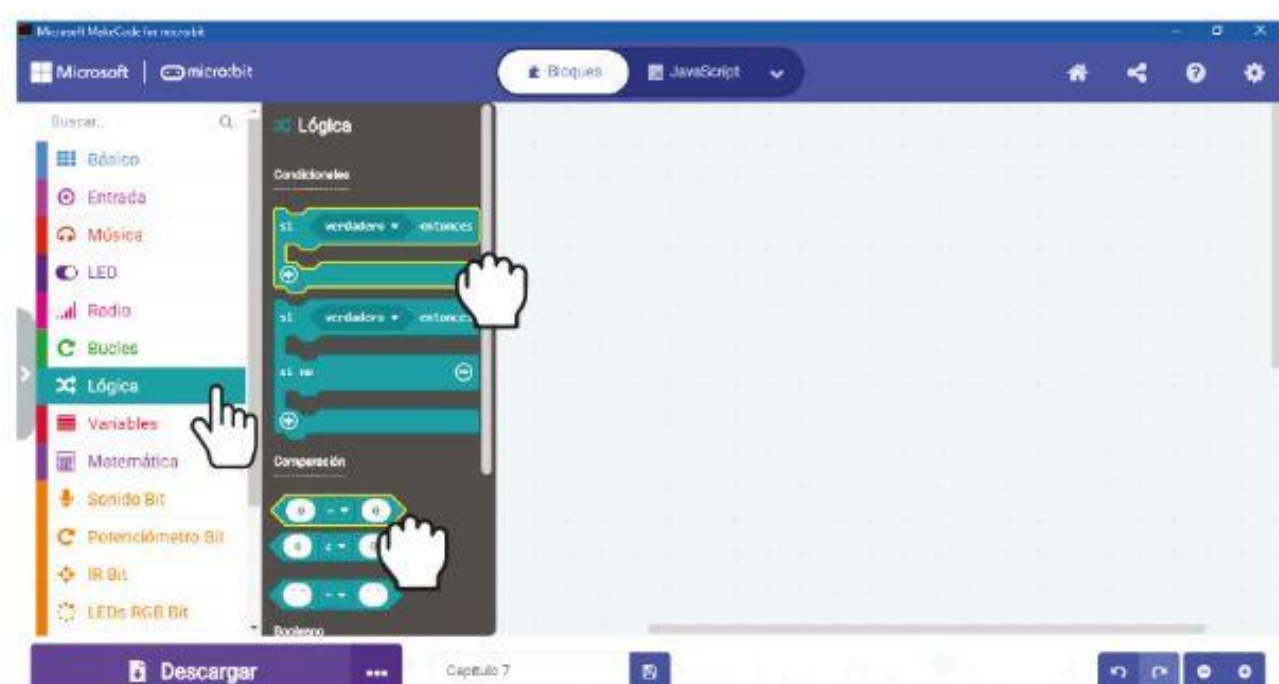
CAPÍTULO 7 : ¡Escuchemos los aplausos!



Paso 13 Coge el bloque **[establecer _ para _]** de la categoría **[Variables]** e insertalo en la primera ranura **[si-entonces-si no]**. Establece la variable en **'volumen'** y encaja el bloque **[nivel sonido]** de la categoría **[Sonido Bit]** en la ranura de valor.

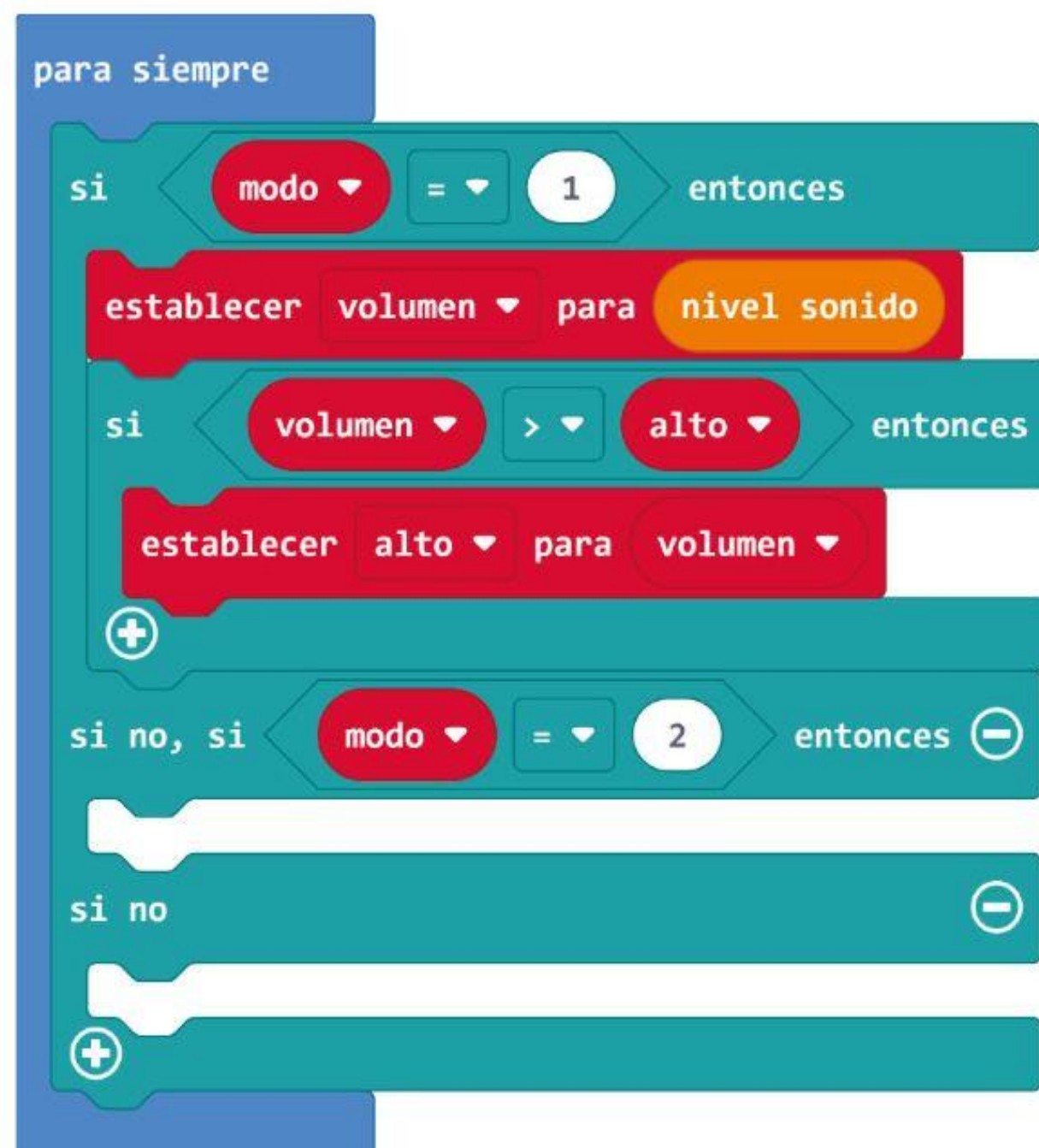
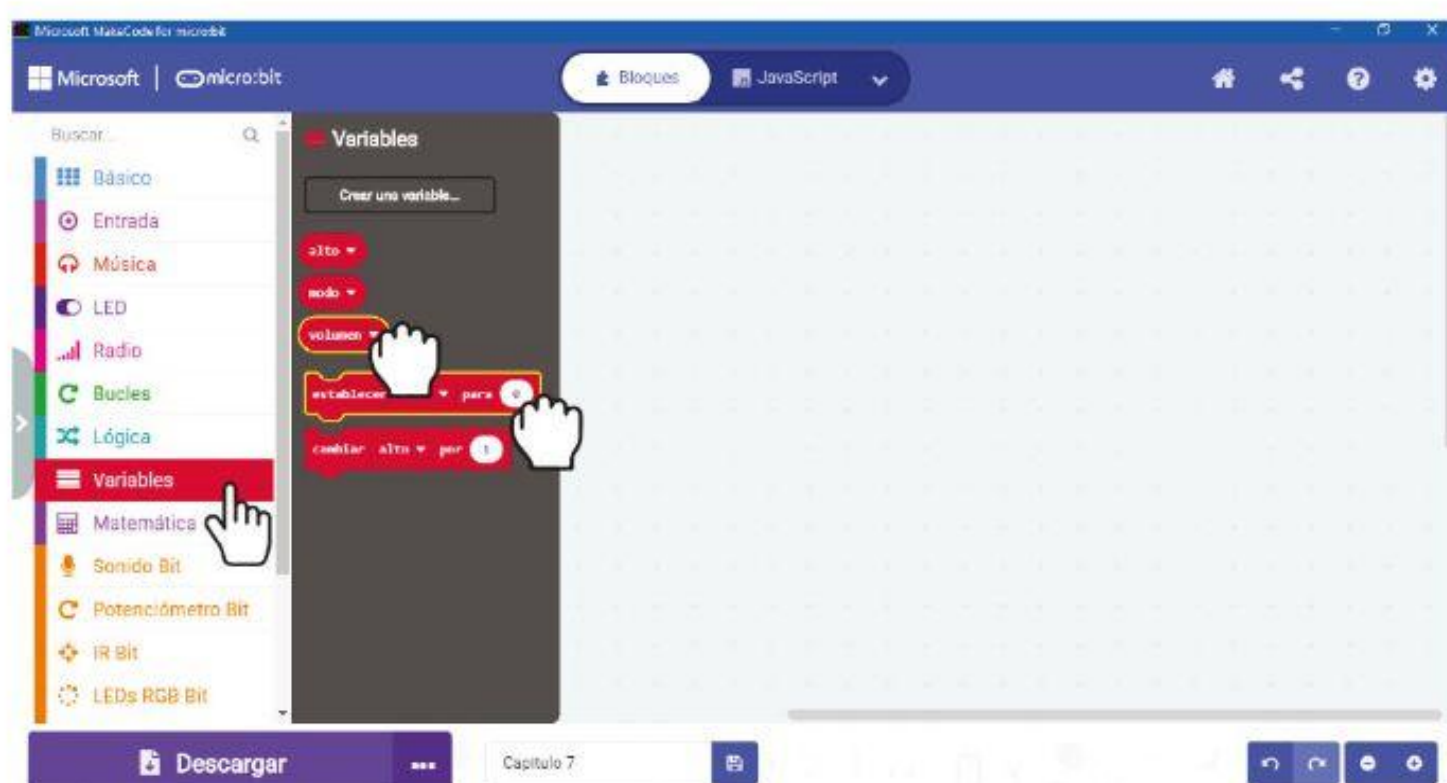


Paso 14 Haz clic en la categoría **[Lógica]** y luego selecciona el bloque **[si-entonces]** y el bloque de comparación **[_ = _]**. Cambia el símbolo **=** por **>**. Encaja **[volumen]** y **[alto]** de la categoría **[Variables]** en las ranuras del bloque de comparación.

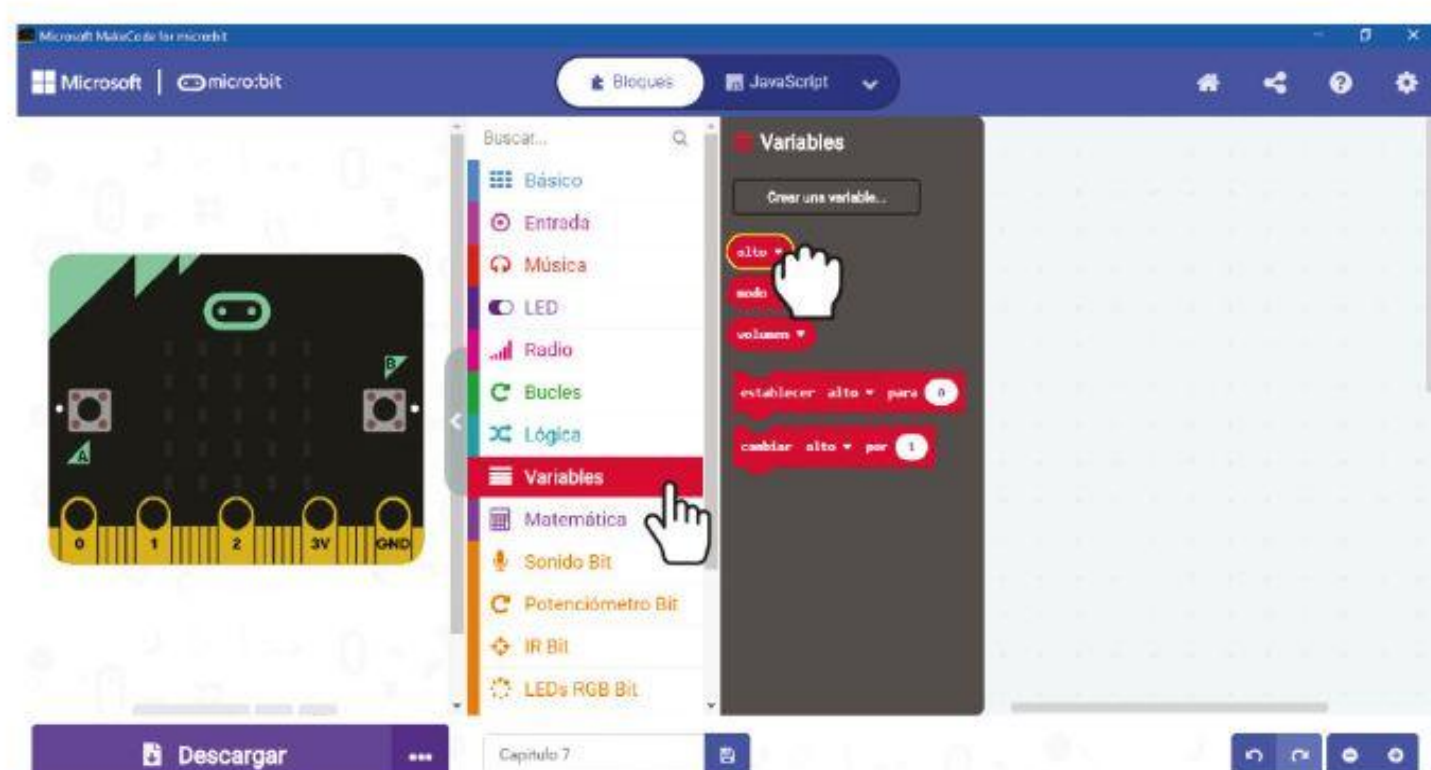


CAPÍTULO 7 : ¡Escuchemos los aplausos!

Paso 15 Haz clic en la categoría **[Variables]** y luego selecciona el bloque **[establecer _ para _]**. Encaja el bloque en la ranura **[si-entonces]** y **[volumen]** de la categoría **[Variables]** en la ranura de valor.



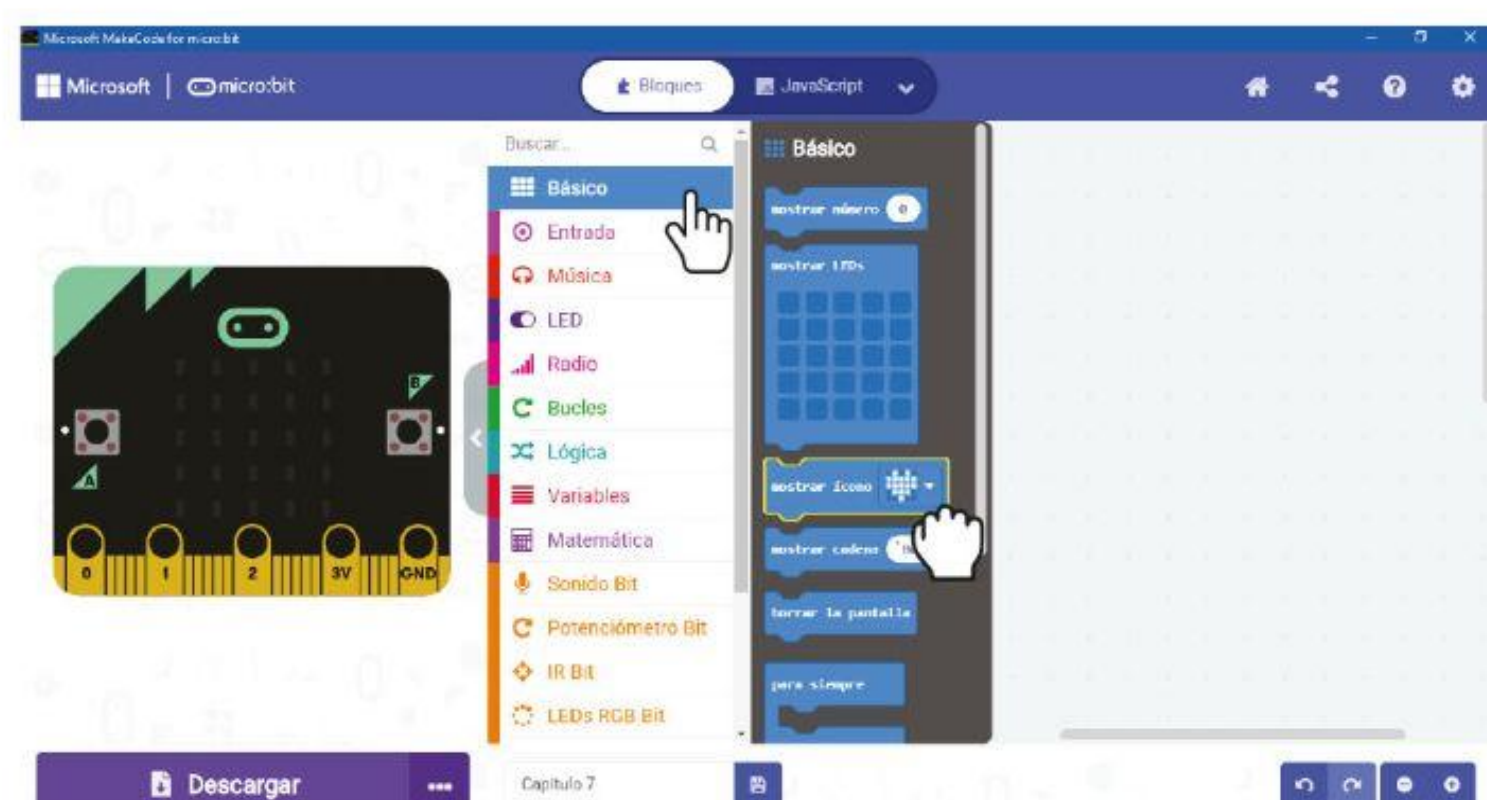
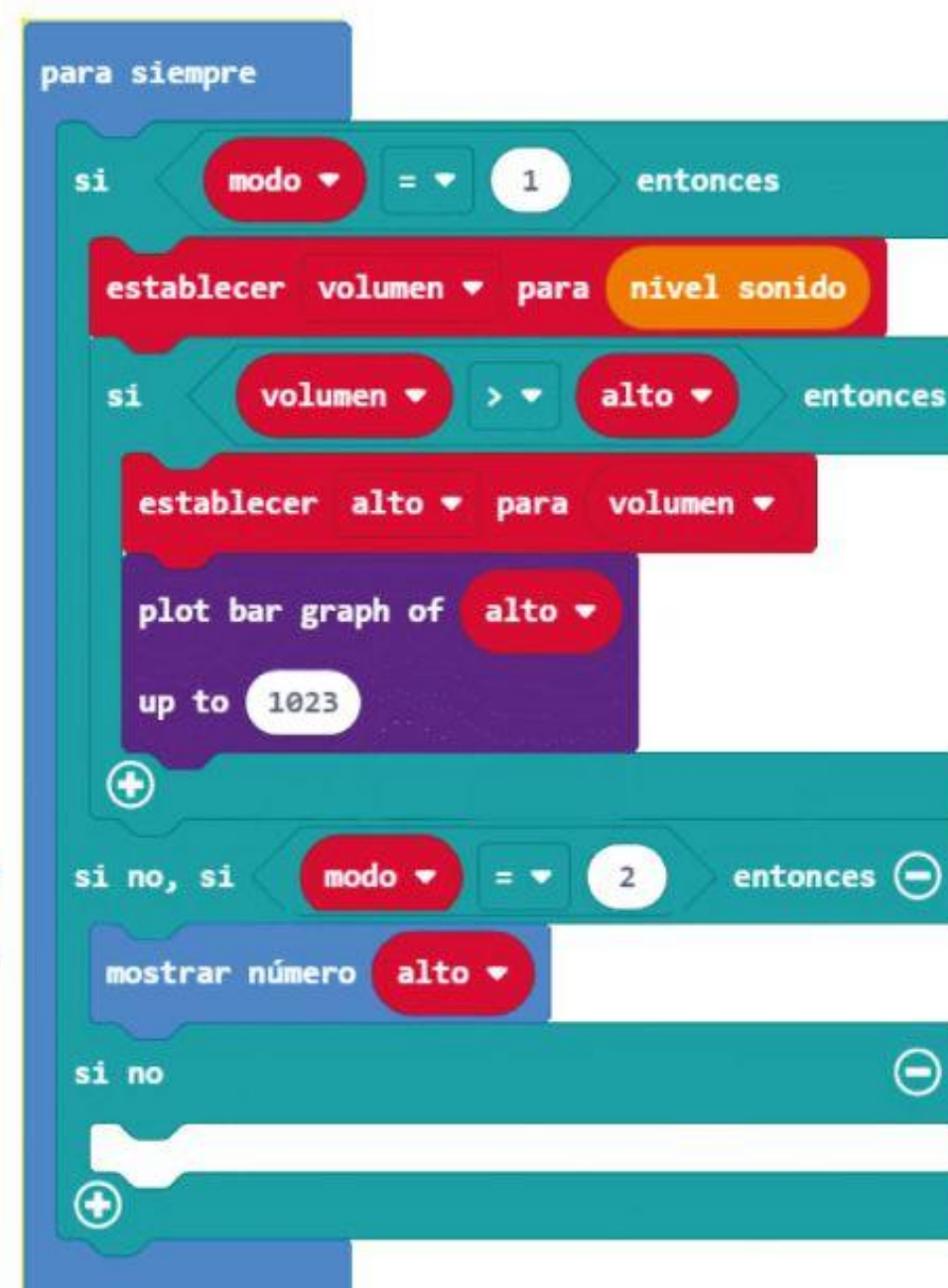
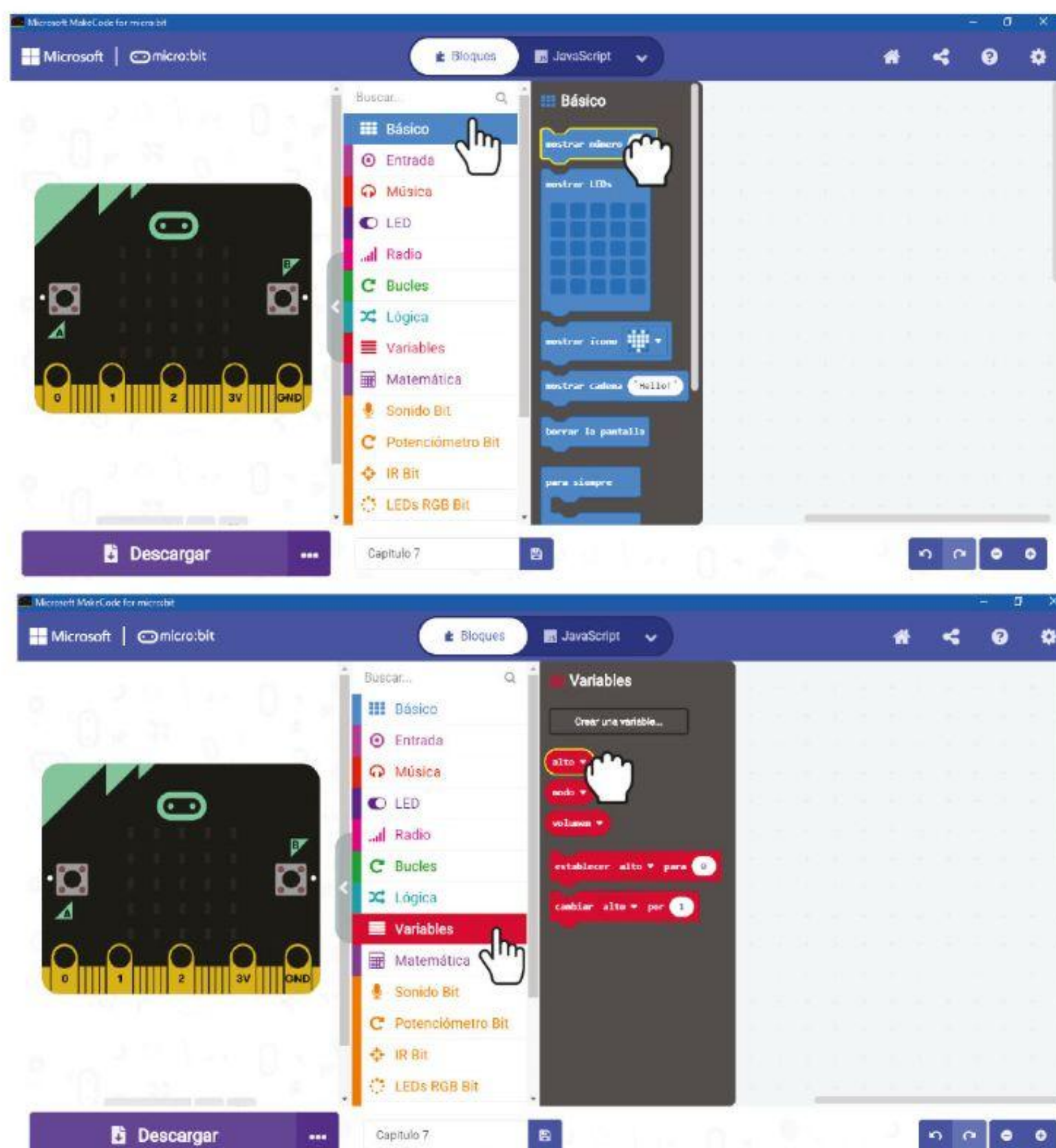
Paso 16 Haz clic en la categoría **[Led]** y selecciona el bloque **[graficar diagrama de barras de _ hasta _]**. Haz clic en la categoría **[Variables]** y selecciona el bloque **[volumen]**. Introduce el bloque en **[graficar diagrama de barras de _ hasta _]** y cambia el valor a 1023.



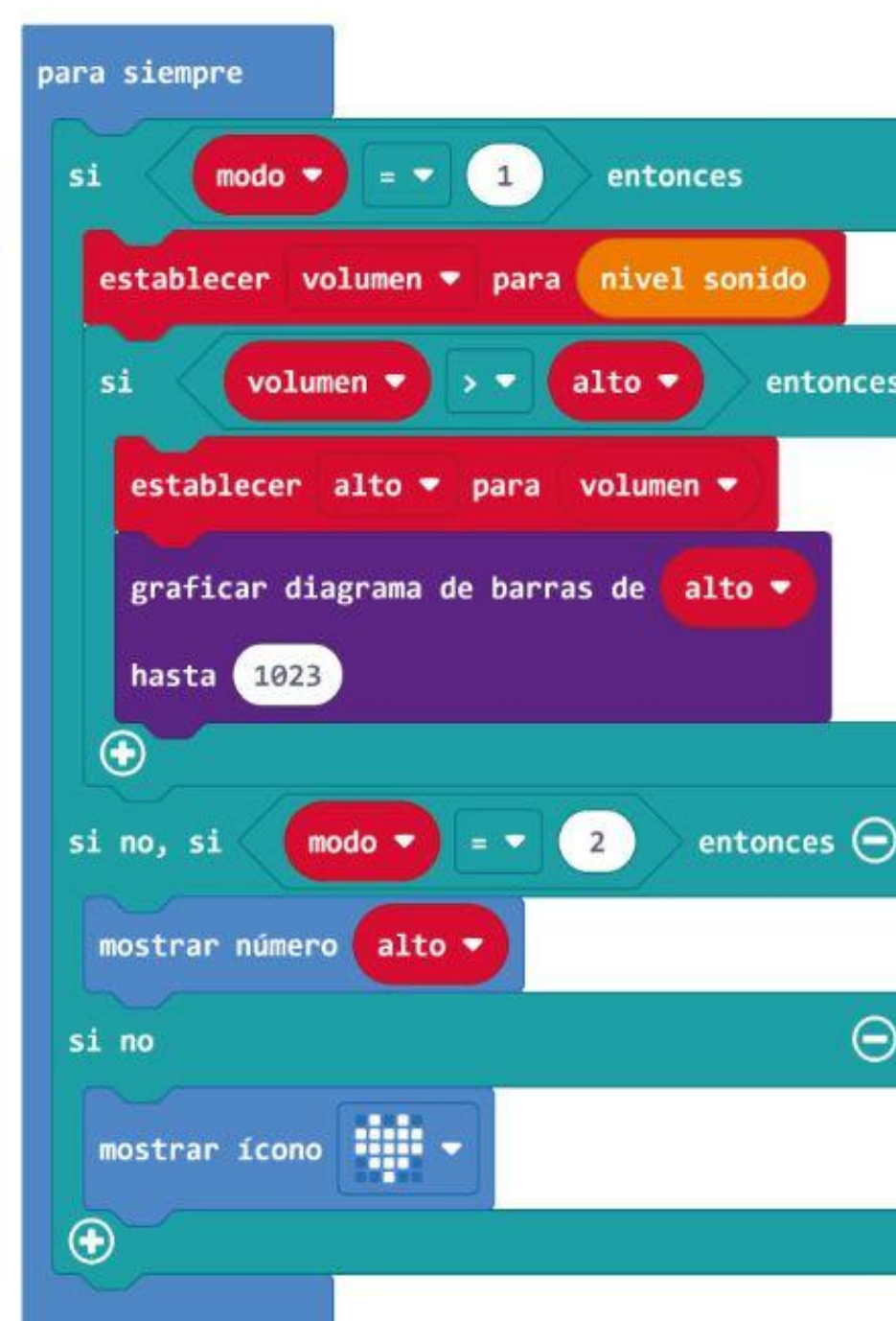
CAPÍTULO 7 : ¡Escuchemos los aplausos!



Paso 17 Haz clic en la categoría **[Básico]** y selecciona el bloque **[mostrar número]**. Colócalo en la segunda ranura del bloque. **[si-entonces-si no]**. Coge el bloque **[alto]** de la categoría **[Variables]** y colócalo en la ranura del bloque **[mostrar número]**.



Paso 18 Haz clic en la categoría **[Básico]** y selecciona el bloque **[mostrar ícono]**. Colócalo en la última ranura del bloque. **[si-entonces-si no]**.



CAPÍTULO 7 : ¡Escuchemos los aplausos!

Aquí está el código completo:

Al inicio, configura el 'modo' en 0.

Cuando se presiona el botón A, cambia el 'modo' a 1 y establece la variable 'alto' a 0

Cuando se activa el bit infrarrojo, cambia el 'modo' a 2

Comprueba siempre el modo actual.

Si 'modo' = 1 (botón A presionado), dibuja un gráfico de barras del ruido más alto detectado en la pantalla de matriz de LED.

Cuanto mayor sea el ruido, más LED se iluminan y viceversa.

Si 'modo' = 2 (sensor de infrarrojos activado), muestra el valor actual de la variable 'alto' (nivel de ruido detectado).

Si 'modo' no es ni 1 ni 2, muestra el icono del corazón.

```
al iniciar
  establecer modo para 0
```

```
al presionarse el botón A
  establecer modo para 1
  establecer alto para 0
```

```
cuando IR activado
  establecer modo para 2
```

```
para siempre
  si modo = 1 entonces
    establecer volumen para nivel sonido
    si volumen > alto entonces
      establecer alto para volumen
      graficar diagrama de barras de alto
      hasta 1023
    +
  si no, si modo = 2 entonces
    mostrar número alto
  si no
    mostrar ícono [icono de corazón]
  +
```

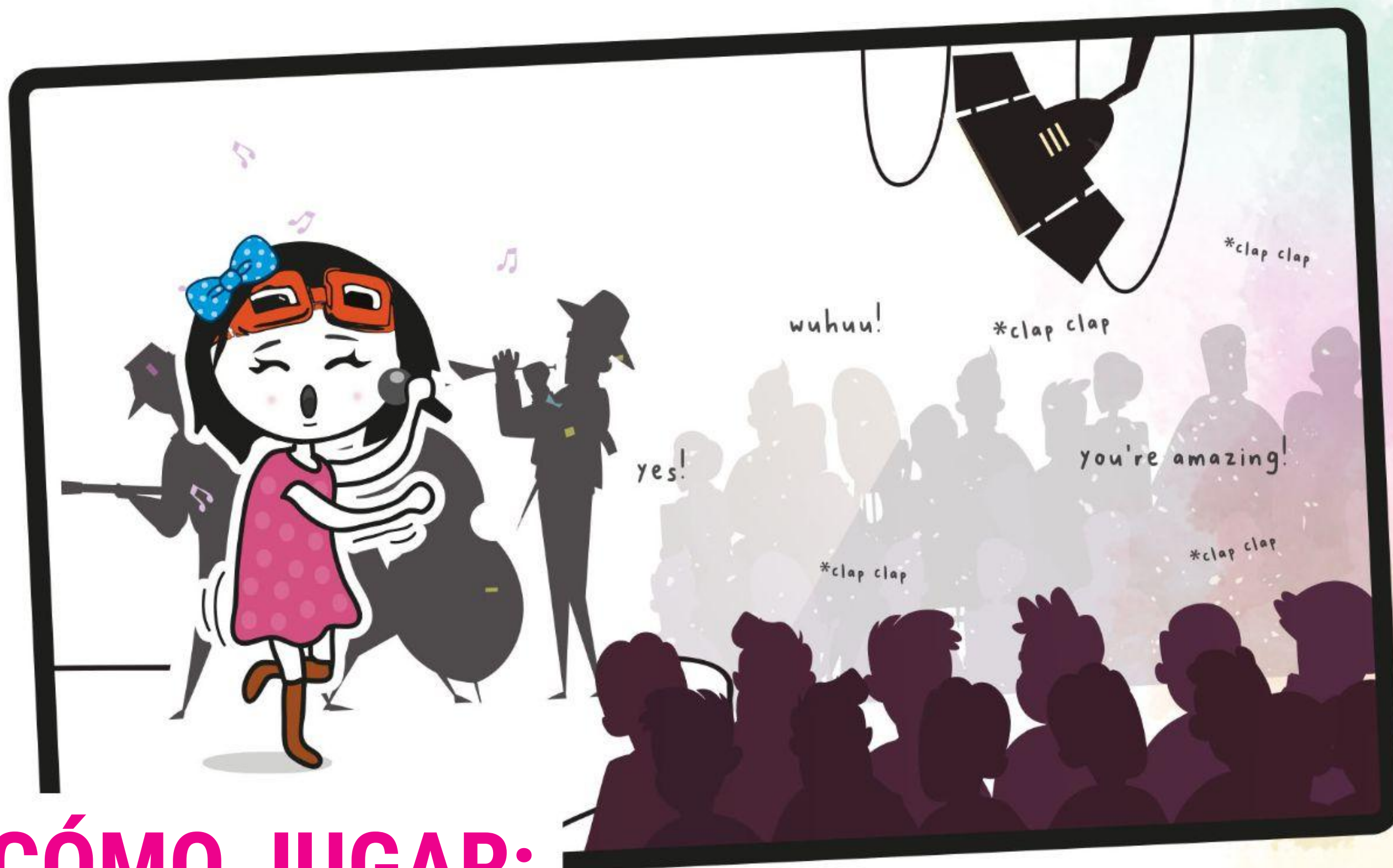
¡ Problemos !



Paso 19 Transfiere el código a tu EDU: BIT y tendrás un medidor de aplausos listo para tu Talent Show.

Vamos a jugar

¡Escuchemos los aplausos!



CÓMO JUGAR:

Los participantes disponen de un tiempo determinado para preparar una breve actuación, ya sea individualmente, en pareja o en equipo. Puedes elegir una canción, un baile o incluso contar un chiste.

Cuando todos estéis listos, turnaros para mostrar vuestro talento. Después de cada actuación, el público responde aplaudiendo: cuanto más les gustó el espectáculo, más fuerte fue el aplauso.

Cuando finalicen los aplausos, activa el Bit Infrarrojo para recorrer la partitura (el nivel de sonido más alto registrado).

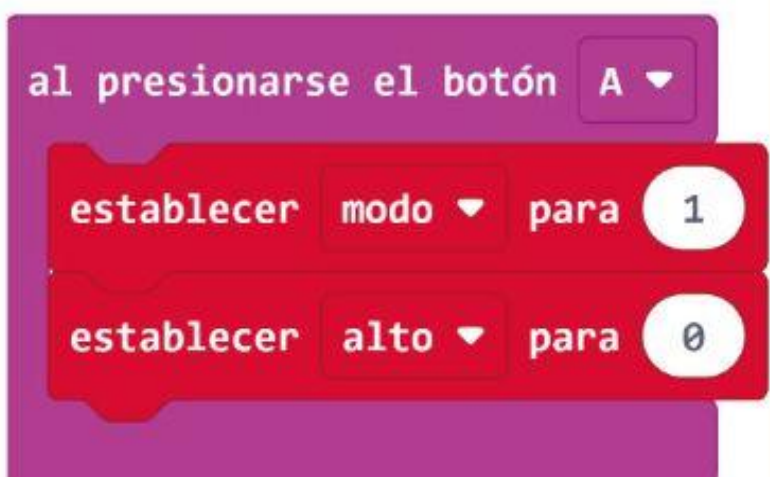
Recuerda presionar el botón A para restablecer la puntuación antes de la próxima actuación.

El ganador es el individuo, la pareja o el equipo que recibe el aplauso más fuerte. ¡Disfruten!

MODIFICAR EL CÓDIGO

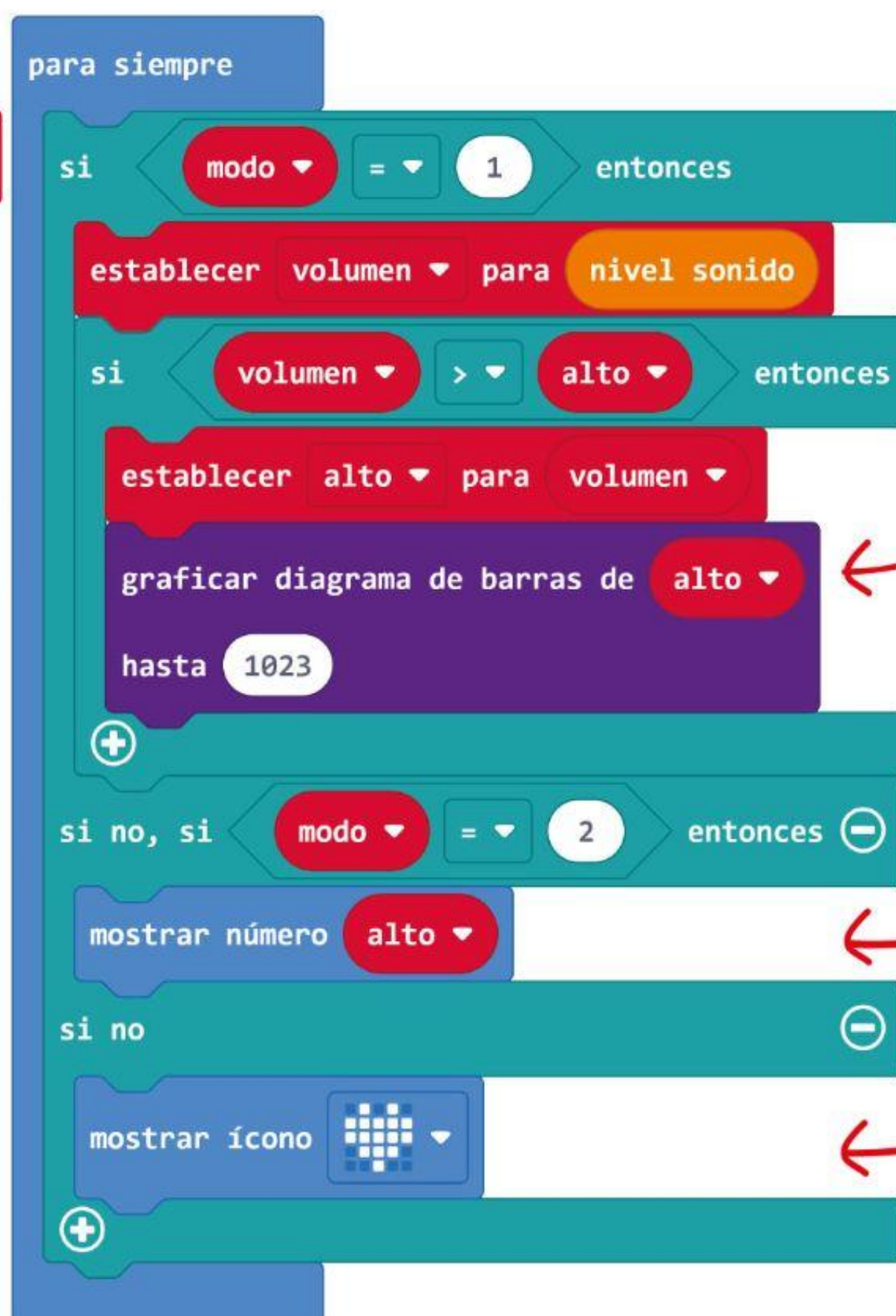
Cuando tenemos múltiples tareas para hacer en un programa, podemos usar eventos desencadenantes para pasar de una tarea a otra. Para que el programa funcione sin problemas, usamos un ciclo que verifica constantemente el modo actual y luego ejecuta los bloques de código correspondientes.

Establece el modo en 0 (en espera) al inicio



Bloques que cambian de modo siguiendo eventos

Siempre verifica cuál es el modo actual y luego realiza la tarea (bloques de código) asignada a ese modo en particular



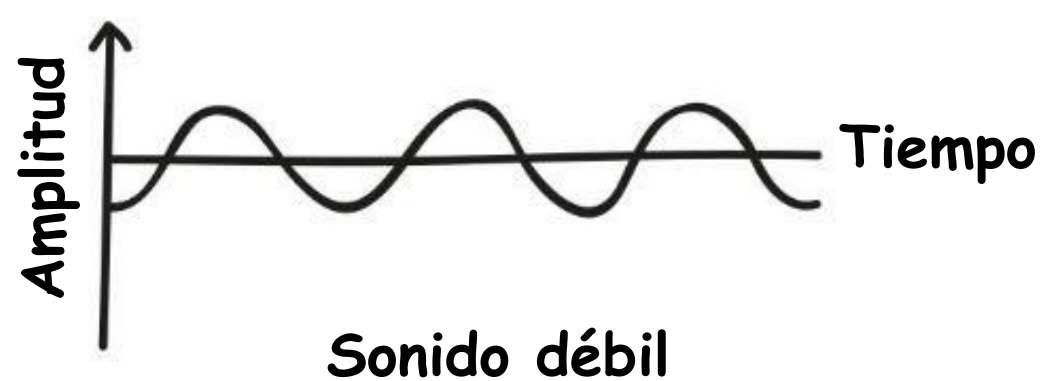
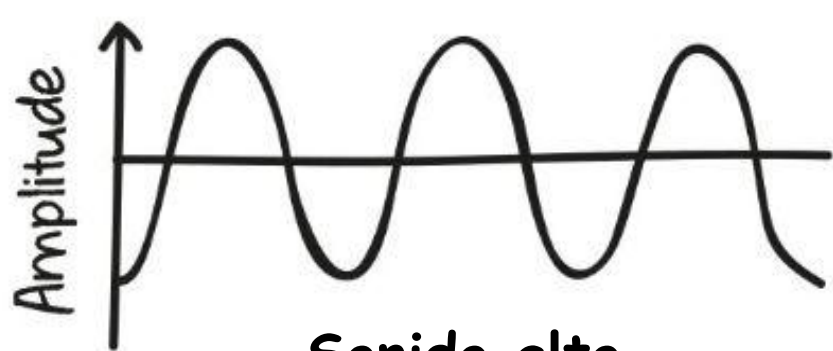
Si tienes modos/tareas adicionales para incluir, siempre puedes agregar bloques activados por eventos, como [cuando se agita] y [cuando el nivel de sonido > _], y luego haz clic en el botón (+) en [si-entonces-de lo contrario] bloque para agregar otras condiciones.

¡COSAS DIVERTIDAS!



El sonido se produce cuando un objeto vibra, por ejemplo, un tambor cuando se golpea. La vibración provoca el movimiento de las moléculas de aire (medio) a su alrededor y crea ondas sonoras.

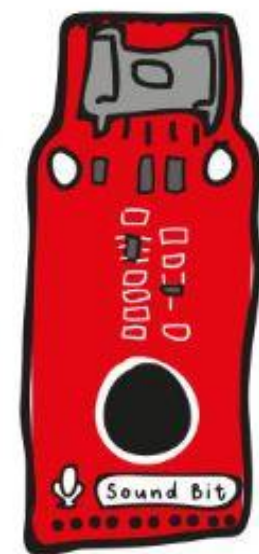
Un **sensor de sonido** es un módulo que detecta la intensidad de las ondas sonoras (es decir, el volumen) y las convierte en una señal eléctrica.



De alguna manera, un sensor de sonido funciona de manera similar a nuestros oídos, que convierten las vibraciones del aire en señales electroquímicas que luego el cerebro traduce en los sonidos que conocemos.



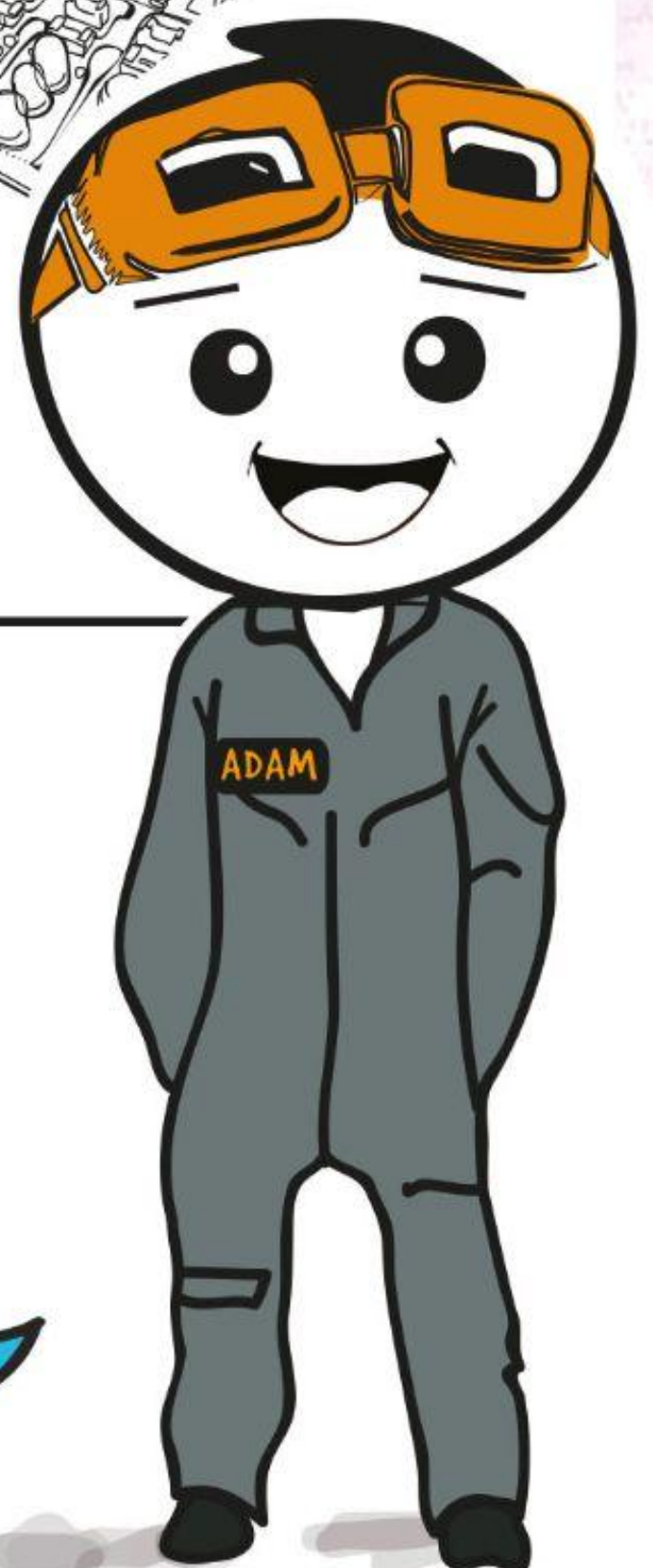
Sonido detectado



Aquí hay algunos usos comunes:

- Detector de alarma antirrobo
- Luces activadas por sonido
- Monitoreo sonoro de recién nacidos

¿Crees que nuestro sensor de sonido puede detectar cualquier ruido en el espacio? ¿Por qué sí o por qué no?



DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para que funcione como monitor de ruido en el aula. Enciende los LED en el Traffic Light Bit para indicar el nivel de ruido.

Nivel de ruido	Intervalo de sonido	Semáforo de bits
Demasiado ruido; reducir el volumen	de () a 1023	LED rojo
Ligeramente ruidoso; presta atención al volumen	de () a ()	LED amarillo
Ruido aceptable. ¡Bueno!	de 0 a ()	LED verde

Aquí tienes algunos consejos ...

1 Deberás determinar los valores de umbral para cada nivel de ruido por adelantado

#2 Para un monitoreo más estable, promedia las lecturas de ruido obtenidas a intervalos regulares



¿Demasiado fácil? Prueba este desafío más difícil. Cambia el código para que los valores de umbral sean relativos a la posición del potenciómetro.



CAPÍTULO 8

¡Vamos a dar un paseo!

Motor DC



Let's Play / Chapter 8
EDU:BIT TWISTER

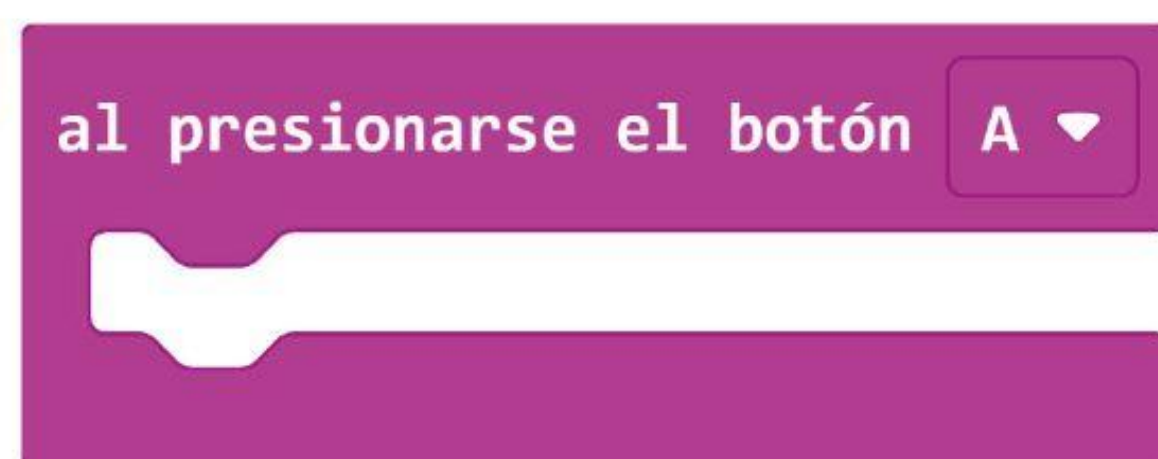
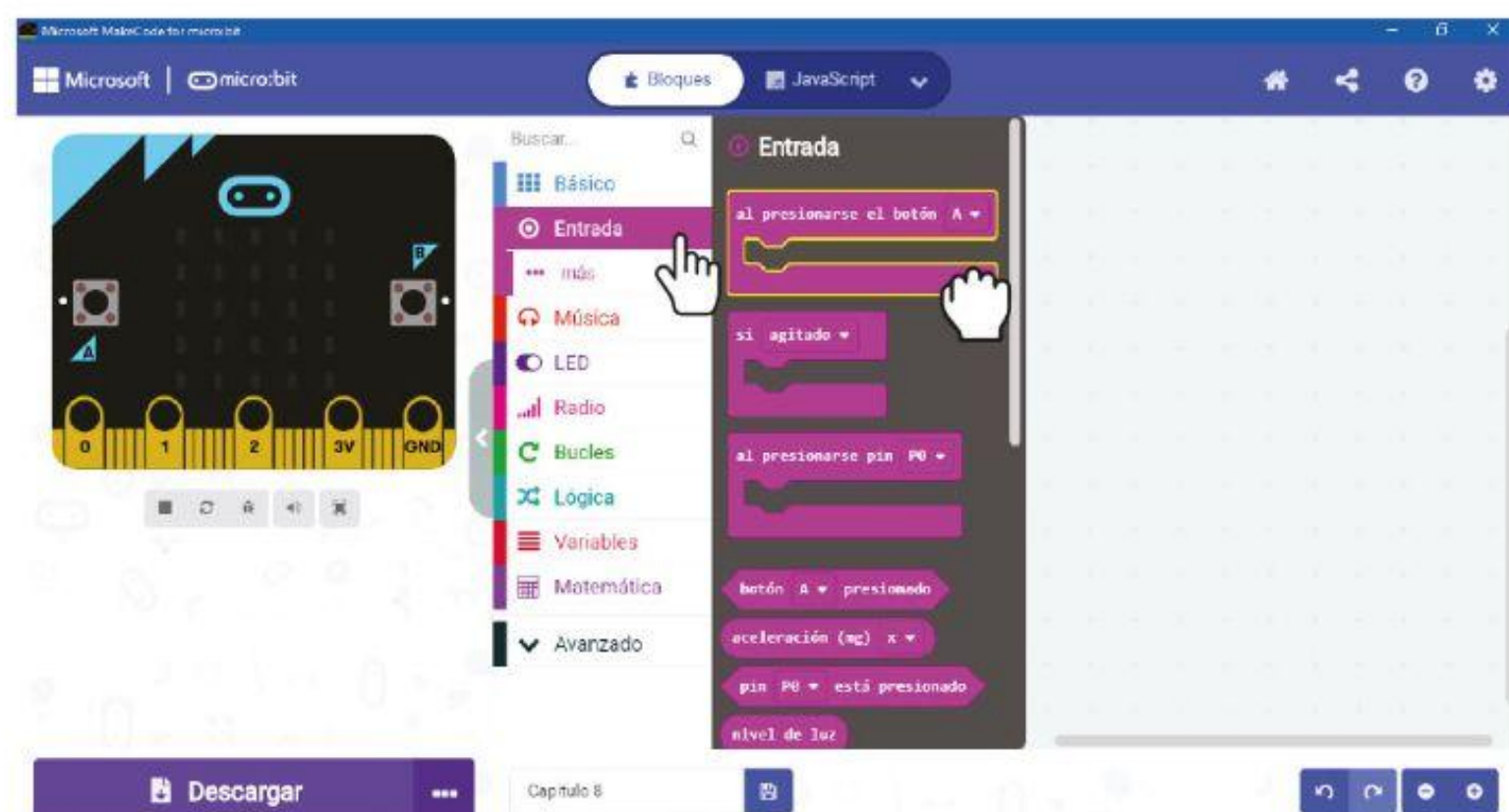
www.cytron.io

EDU Training & Project k

CAPÍTULO 8 : ¡Vamos a dar un paseo!

¡PROGRAMAMOS!

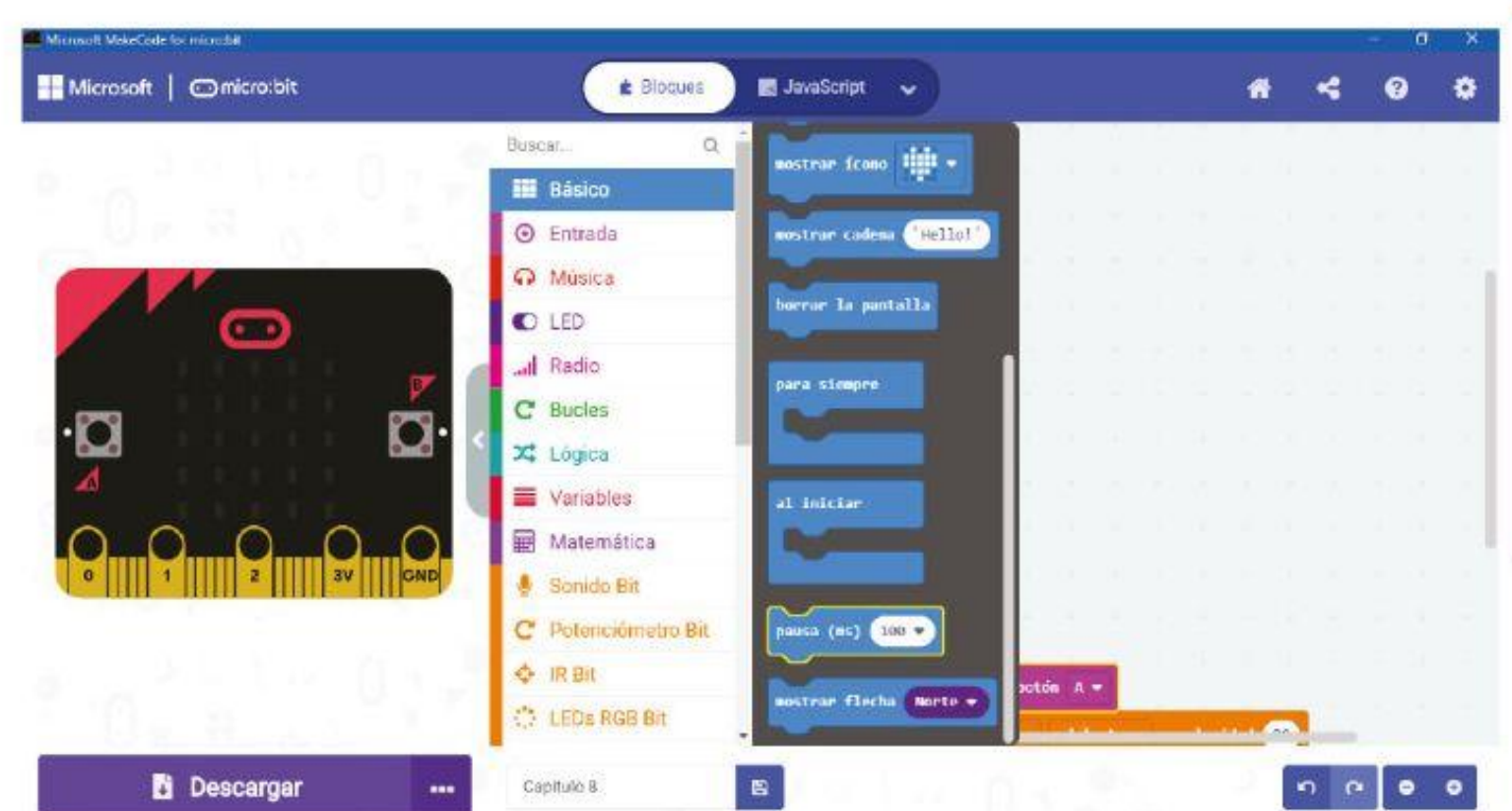
Paso 1 Crea un nuevo proyecto en el editor de MakeCode y agrega la extensión EDU: BIT (puedes consultar la página 40). Haz clic en la categoría **[Entrada]** y selecciona el bloque **[al presionarse el botón _]**.



Paso 2 Haz clic en la categoría **[Motores y Servos]** y agrega los bloques **[Activa motor _ a la velocidad _]** y **[Frena motor _]**. Cambia el valor de la velocidad a **80**.



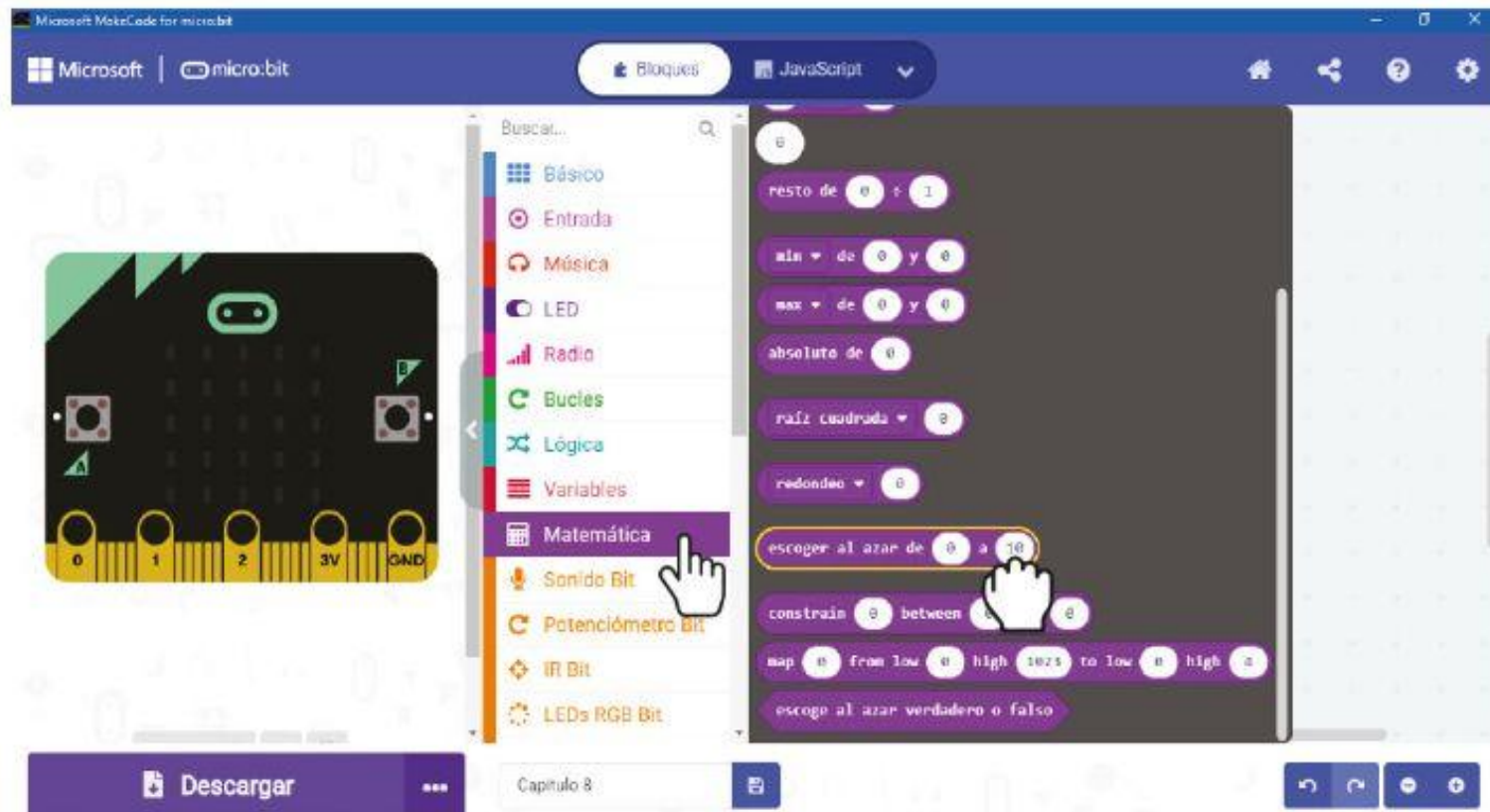
Paso 3 Haz clic en la categoría **[Básico]** y selecciona el bloque **[pausa (ms) _]**. Coloca el bloque entre el bloque **[Activa motor _ a la velocidad _]** y el bloque **[Frena motor _]**.



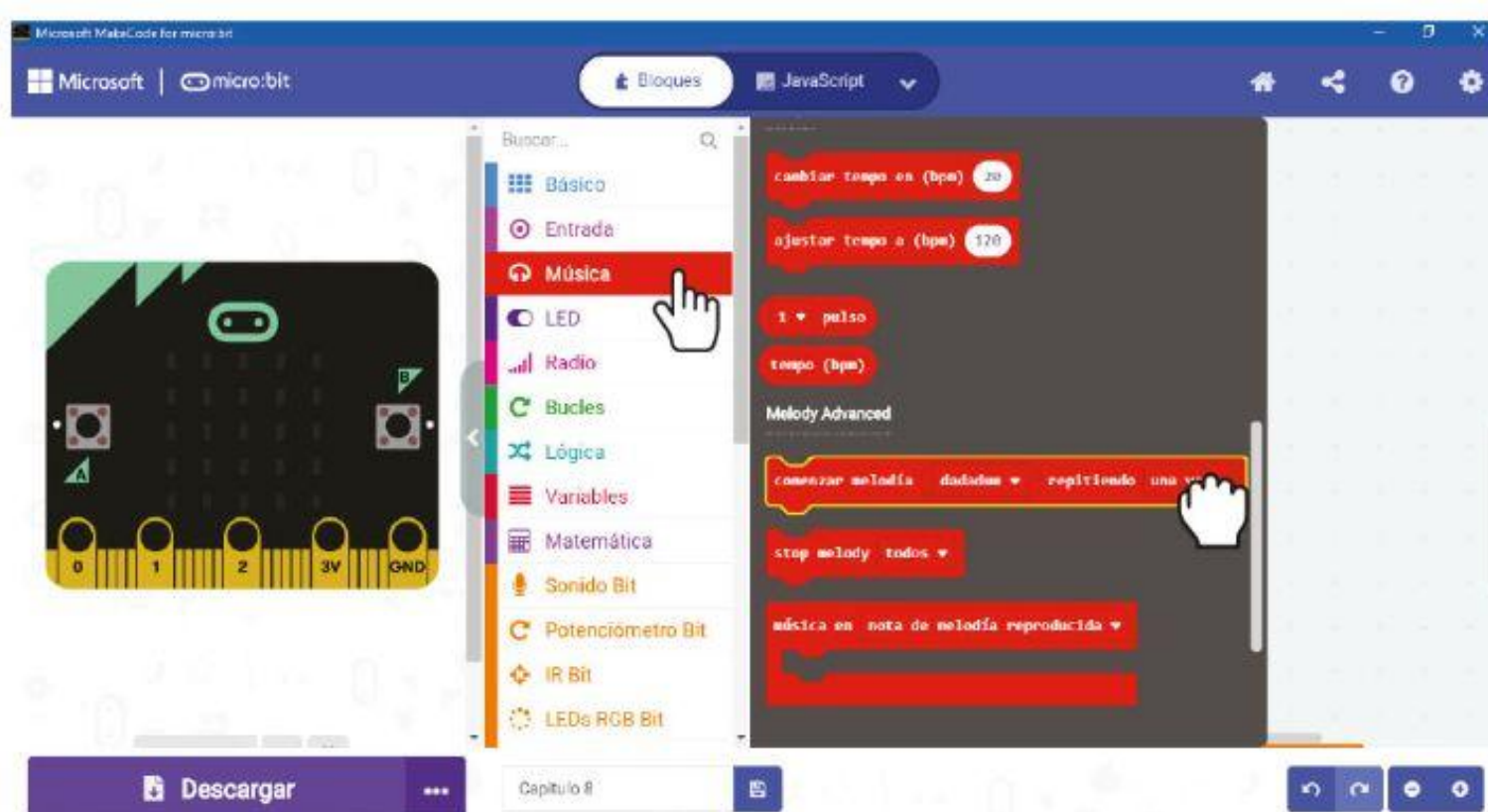
CAPÍTULO 8 : ¡Vamos a dar un paseo!



Paso 4 Haz clic en la categoría **[Matemática]** y selecciona el bloque **[escoger al azar de _ a _]**. Ponlo en el bloque **[pausa (ms) _]** y cambia los valores a **200** y **1000** respectivamente.



Paso 5 Haz clic en **[Música]** y selecciona el bloque **[comenzar melodía _ repitiendo _]**. Cambia la melodía a **"ba ding"** (o elige una melodía que quieras).



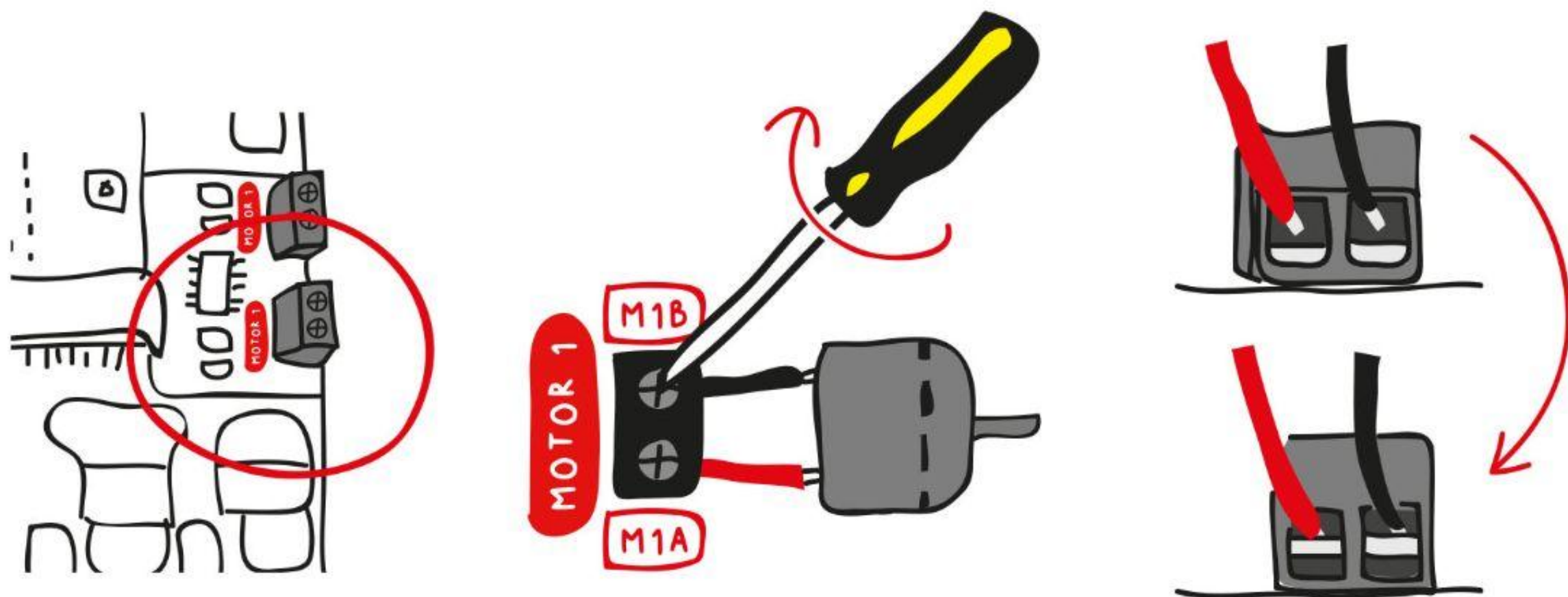
Paso 6 Transfiere el código completado a su EDU: BIT.

Podemos usar este código para cualquier juego que requiera un objeto que gire aleatoriamente. Básicamente, el motor comienza a funcionar cuando se activa, luego se detiene después de un tiempo elegido al azar.



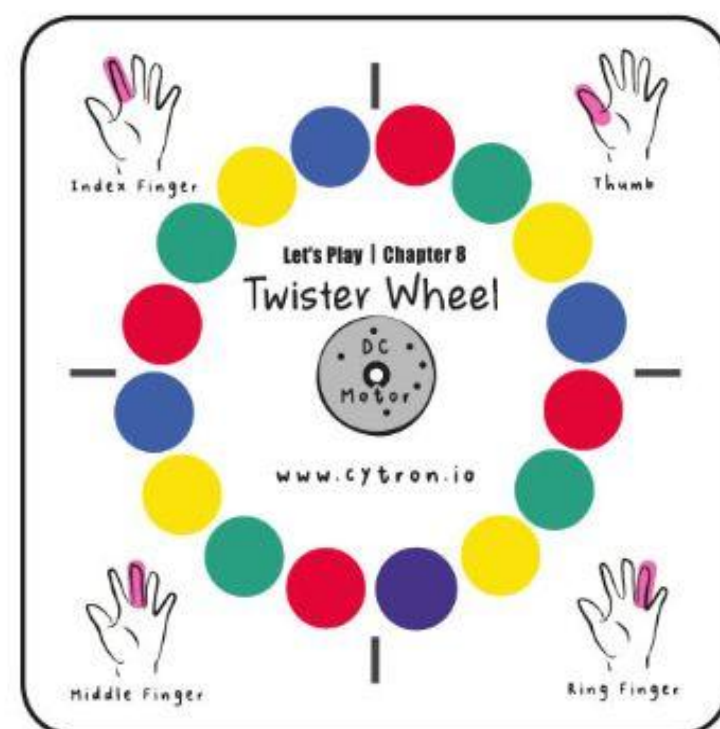
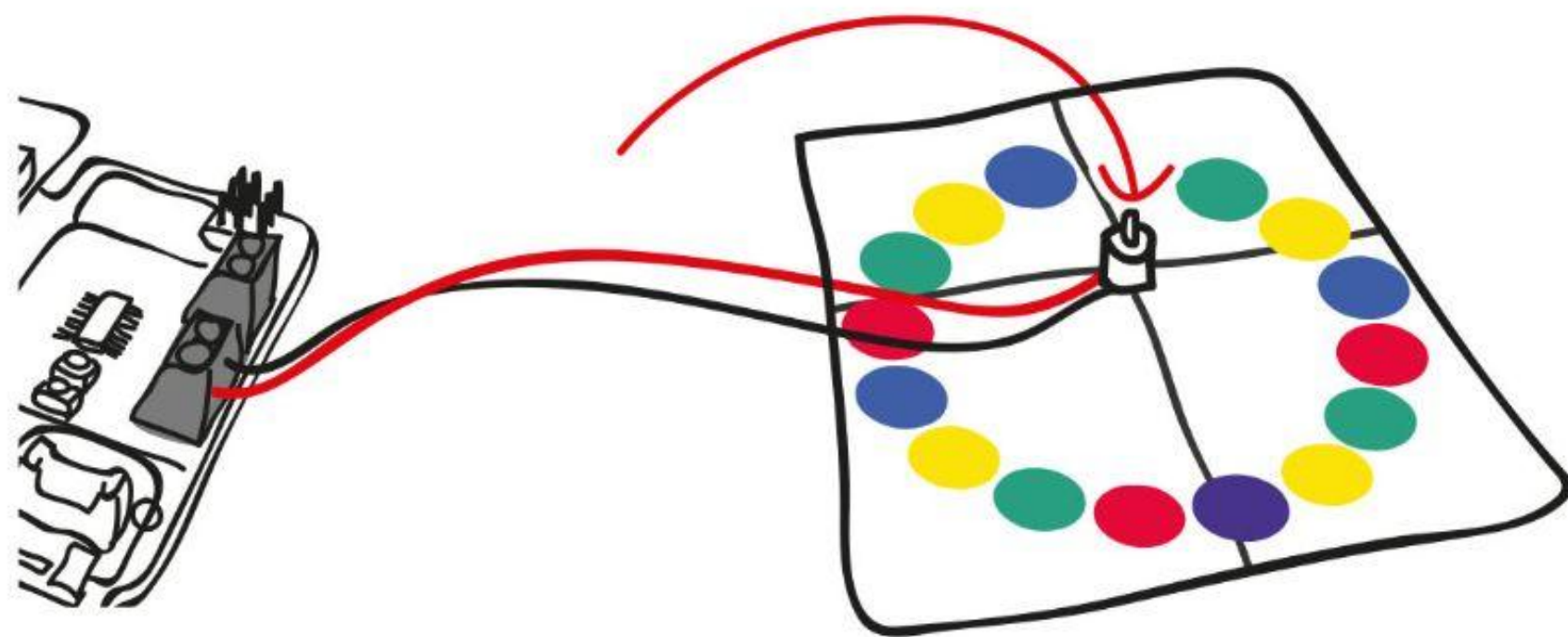
CAPÍTULO 8 : ¡Vamos a dar un paseo!

Paso 7 Conecta el motor de CC a los terminales MOTOR 1: (i) inserta los cables pelados y luego (ii) aprieta los tornillos con el destornillador provisto, para mantener las conexiones firmes y ajustadas.

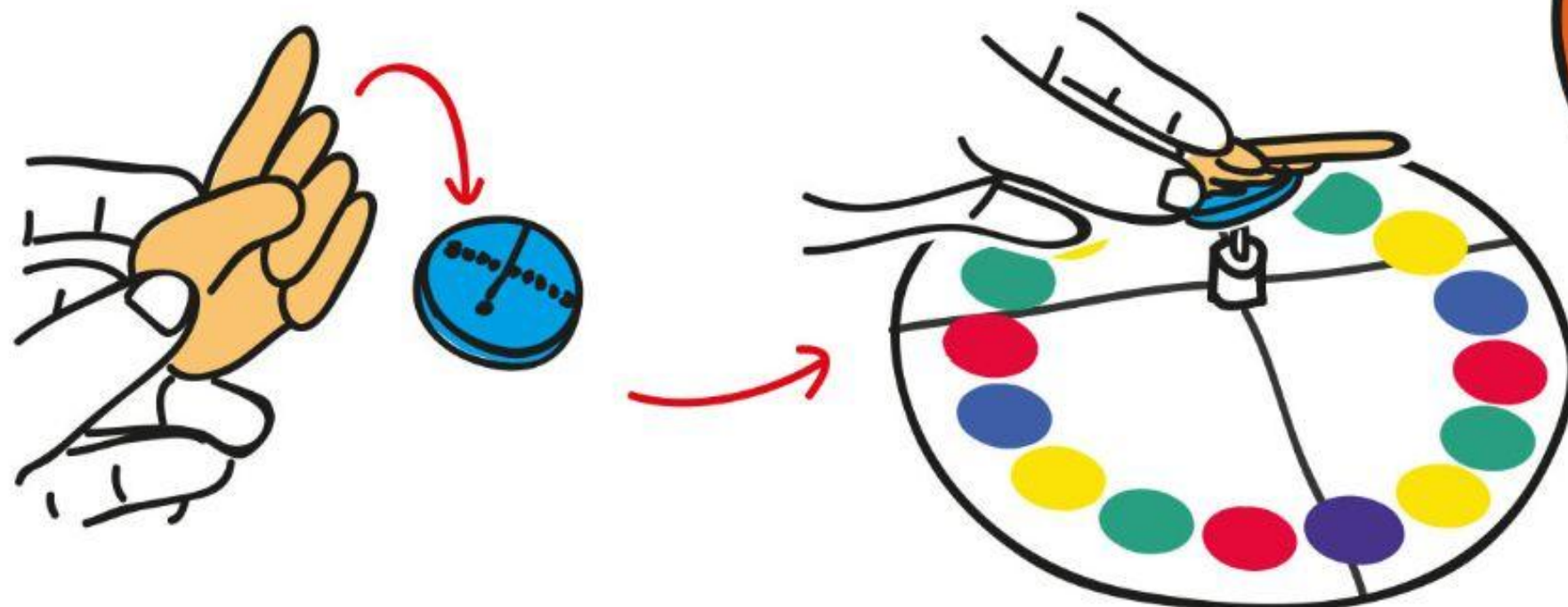


Presiona el botón amarillo (botón A) para intentarlo. Si el motor no gira, verifica que la conexión de los cables a los terminales esté bien apretada y que la EDU: BIT esté encendida.

Paso 8 Usa adhesivo, como cinta de doble cara o pegamento caliente, para unir el motor de CC al centro de la rueda Twister (como se indica).



Paso 9 Saca el puntero y pégalo al disco de plástico con adhesivo. Luego fija el disco en el cigüeñal.



La rueda Twister, el puntero y el mapa del juego se incluyen en la caja.

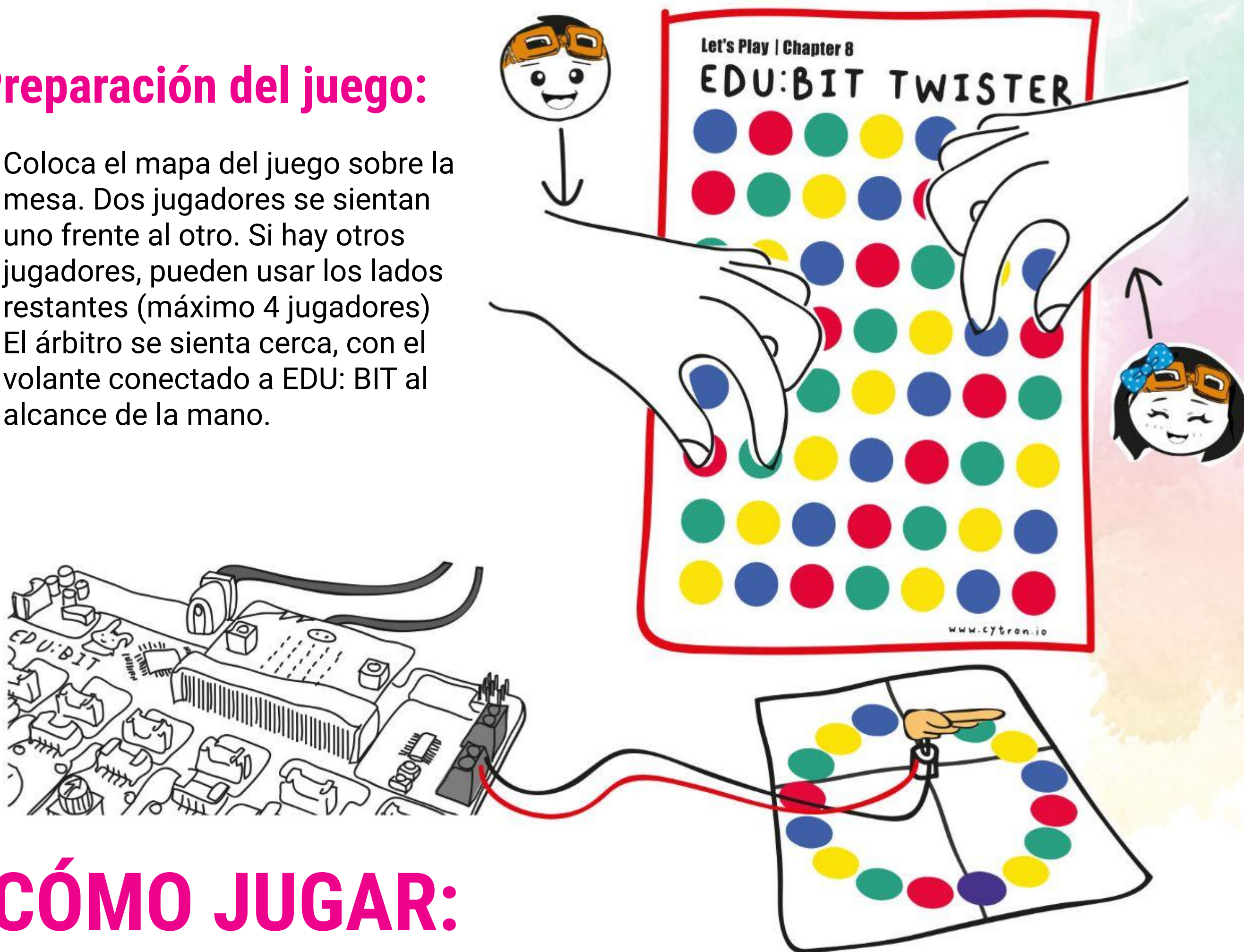


Vamos a jugar

¡Vamos a dar un paseo!

Preparación del juego:

- Coloca el mapa del juego sobre la mesa. Dos jugadores se sientan uno frente al otro. Si hay otros jugadores, pueden usar los lados restantes (máximo 4 jugadores)
- El árbitro se sienta cerca, con el volante conectado a EDU: BIT al alcance de la mano.



CÓMO JUGAR:

En este juego, los jugadores se turnan para colocar los dedos sobre los círculos de colores del tablero, según lo indique el árbitro.

El papel del árbitro es presionar el botón amarillo (o botón A) para girar el puntero y luego comunicar el dedo y el color que señala el puntero, por ejemplo: "dedo índice; rojo".

Cuando sea tu turno, debes escuchar las instrucciones del árbitro y colocar el dedo que te indiquen en un círculo del color correcto. Si ese dedo ya está en un círculo de ese color, debe intentar moverlo a otro círculo del mismo color.

Si fallas en completar un movimiento, estás fuera del juego.

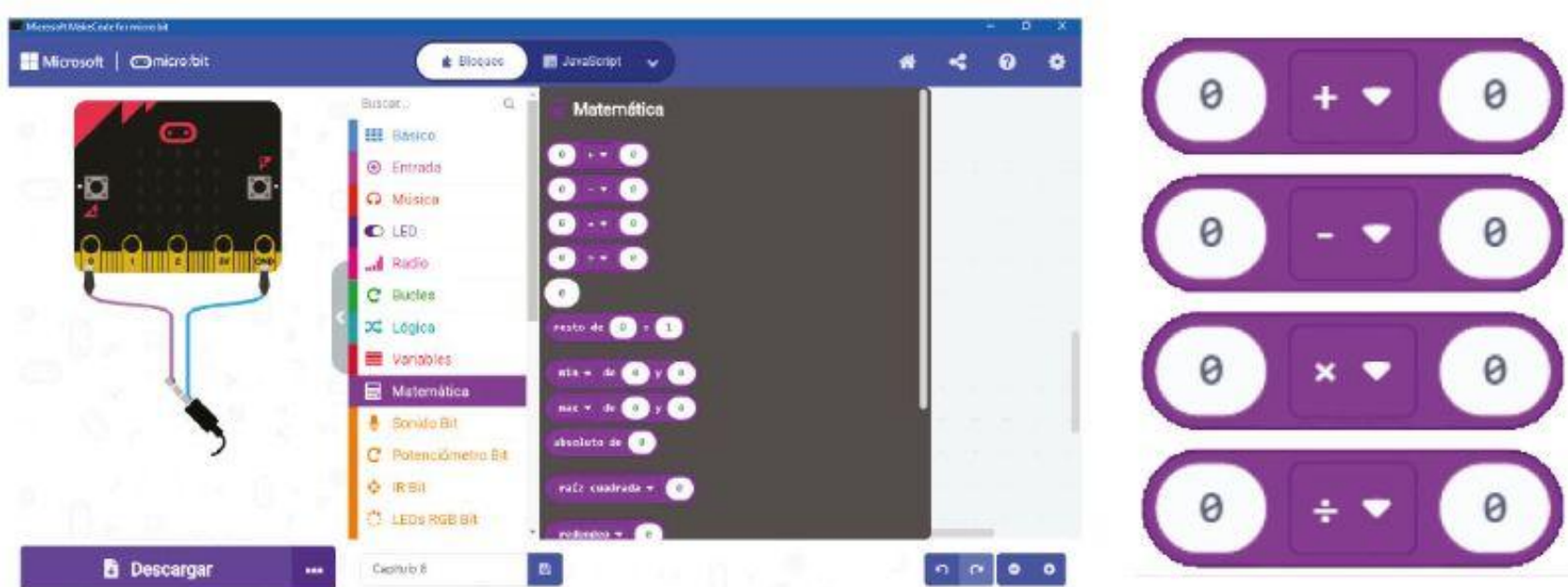
¡El último jugador que queda en el juego es el GANADOR!



EXPLORA OTROS BLOQUES

Puedes usar bloques en la categoría **[Matemática]** para realizar operaciones aritméticas en sus variables.

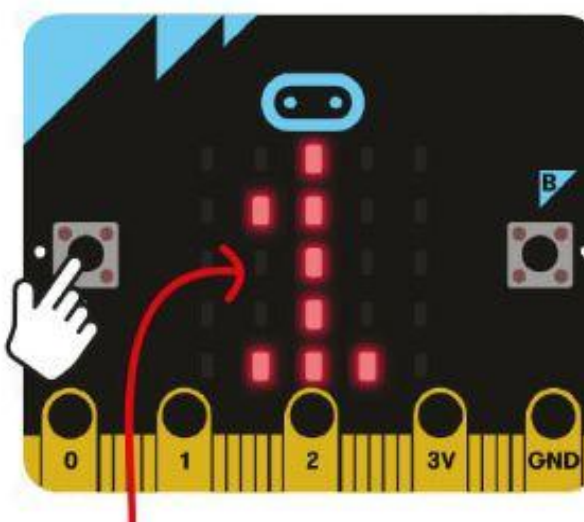
#1 Usa los siguientes bloques para sumar, restar, multiplicar o dividir.



#2 Puedes usar el bloque **[resto de _ ÷ _]** para saber cuánto queda si un número no se puede dividir exactamente entre otro.

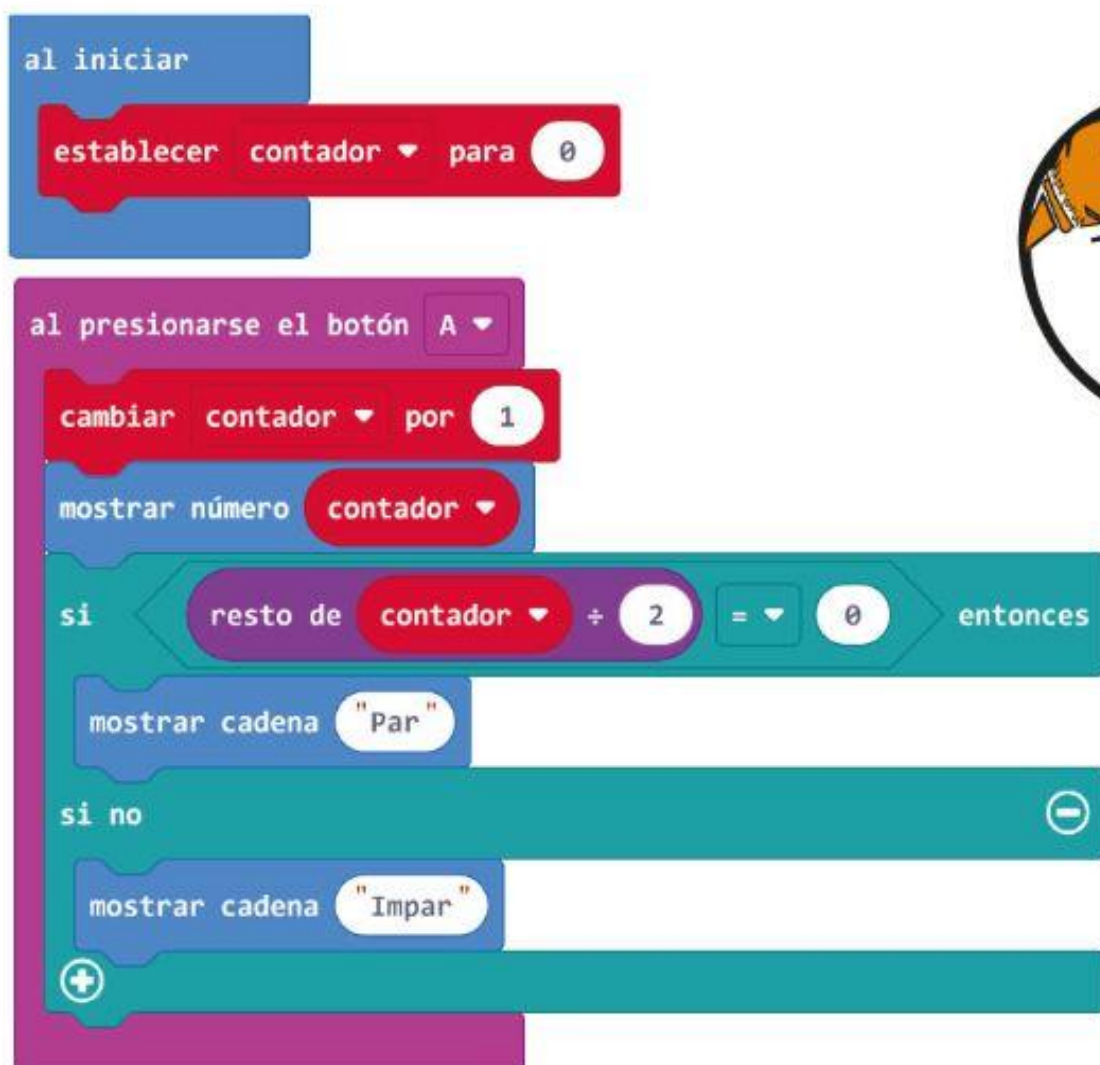


$$\begin{array}{r} 6 \\ 2 \overline{)13} \\ \underline{12} \\ 1 \end{array}$$



resto de la división

#3 También puedes usar el bloque **[resto de _ ÷ _]** para decidir si un número es par o impar. Simplemente divide por 2. Si el resto es "1", entonces es "impar"; si el resto es "0", entonces es "par". ¡Prueba!

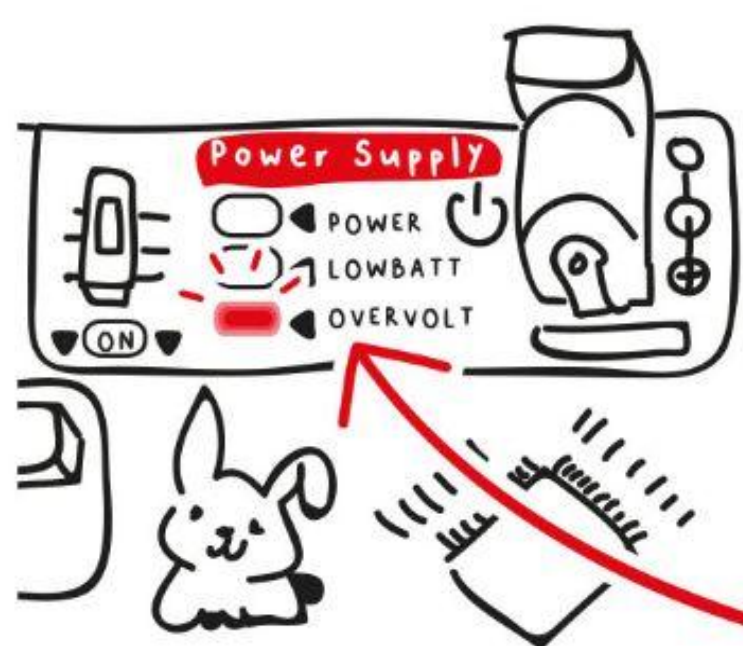


Notaste que EDU: BIT muestra "impar" cuando presionas el botón A por primera vez (1 = número impar) y luego "par" la próxima vez (2 = número par)? ¿Qué se mostrará cuando presione el botón A la 99ª vez?

¡COSAS DIVERTIDAS!

Un motor de corriente continua, o **motor de CC**, es un dispositivo eléctrico giratorio que convierte la energía eléctrica en energía mecánica.

Necesitamos aplicar un voltaje de entrada para hacer que el motor de CC gire. Podemos controlar la velocidad de rotación ajustando el voltaje de entrada. Cuanto mayor sea el voltaje de entrada, más rápido girará el motor. El voltaje recomendado para un motor de CC utilizado en EDU: BIT es de 3,6 V a 6 V.



! ATENCIÓN !

Aplicar un alto voltaje a un motor (más allá del voltaje recomendado) acortará la vida útil del motor a largo plazo.



Indicador excesivo

Puedes controlar fácilmente la dirección de rotación y la velocidad del motor de CC utilizando el siguiente bloque de programa.

Active motor M1 ▶ adelante ▶ a velocidad 128

✓ M1
M2
todos

✓ adelante
atrás

Speed 255

Dirección de rotación

Hay dos terminales de motor de CC en la placa EDU: BIT. Seleccione la terminal correcta.

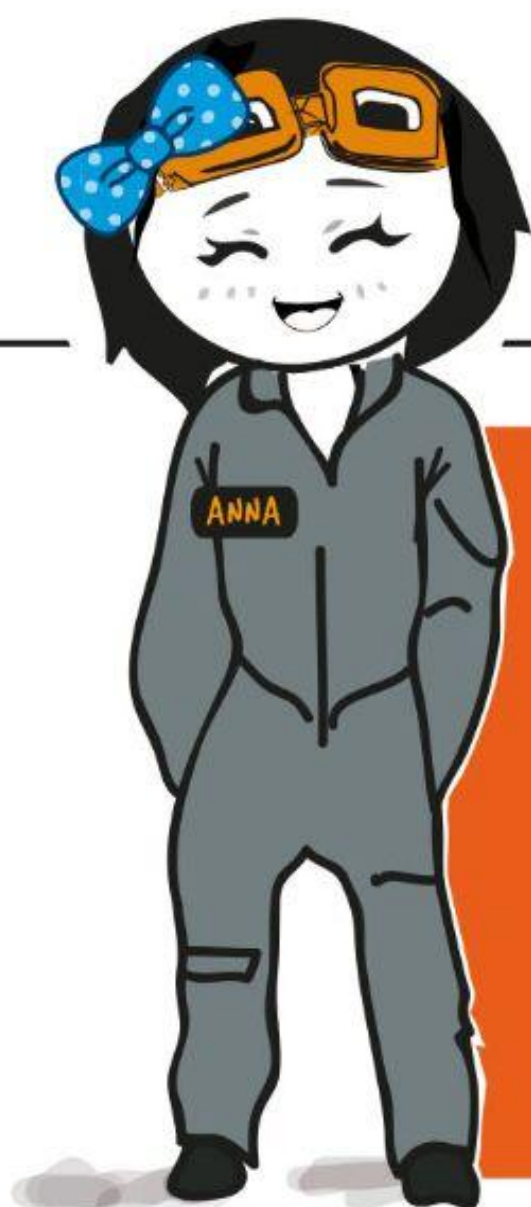
Este es un valor relativo que va de 0 a 255. Cuanto mayor sea el valor, más rápido funcionará el motor.

¿Sabes que EDU: BIT ha integrado un circuito para probar un motor? Presione los botones blancos (marcados como M1A, M1B, M2A y M2B) para verificar si la conexión está bien apretada y si el motor funciona bien.



DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para hacerlo funcionar como un ventilador activado por sonido y cuya velocidad es controlada por el Potenciómetro Bit.	
En el inicio	Muestra un icono de corazón (o uno de tu elección). Establece la variable Modo en 0
Cuando nivel sonido > _ (umbral)	Cambia la variable Modo por 1
Por siempre	<p>Configura la variable 'Velocidad' para mapear la posición del potenciómetro de 0-1023 a 0-255.</p> <p>Siempre verifique los Modos</p> <ul style="list-style-type: none">• Si el Modo es un número par, entonces detiene el motor M1.• DE LO CONTRARIO, SI Modo es un número impar, haga funcionar el motor M1 de acuerdo con el valor de la variable 'Velocidad' (según el mapa de la lectura del potenciómetro).



Aquí tienes algunos consejos ...

- # 1: Deberás determinar el umbral de activación de sonido para encender y apagar el motor.
- #2: Tienes que crear dos variables: Modo y Velocidad.
- # 3: Conecta las aspas del ventilador al eje del motor y luego ejecuta el programa. Si no escuchas el aire soplando cuando gira el motor, debes cambiar la dirección de rotación en el código.

CAPÍTULO 9

Penalti... ¡Gol!

Servomotor

Ready!

Get set...

GO!!!!!!

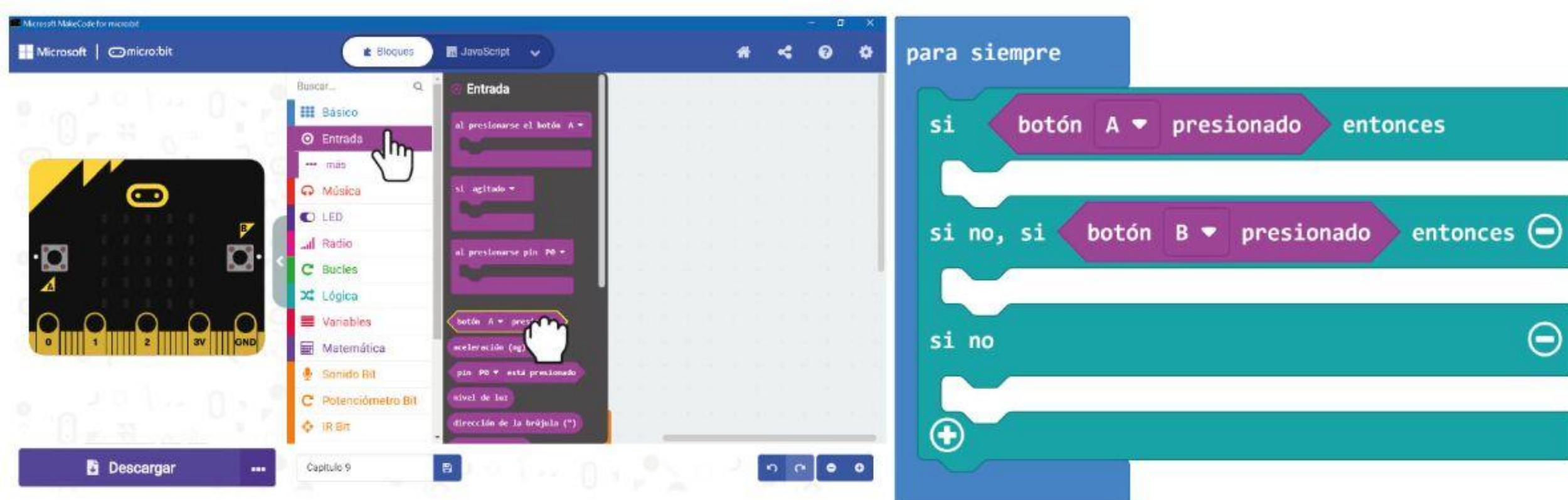


PROGRAMAMOS!

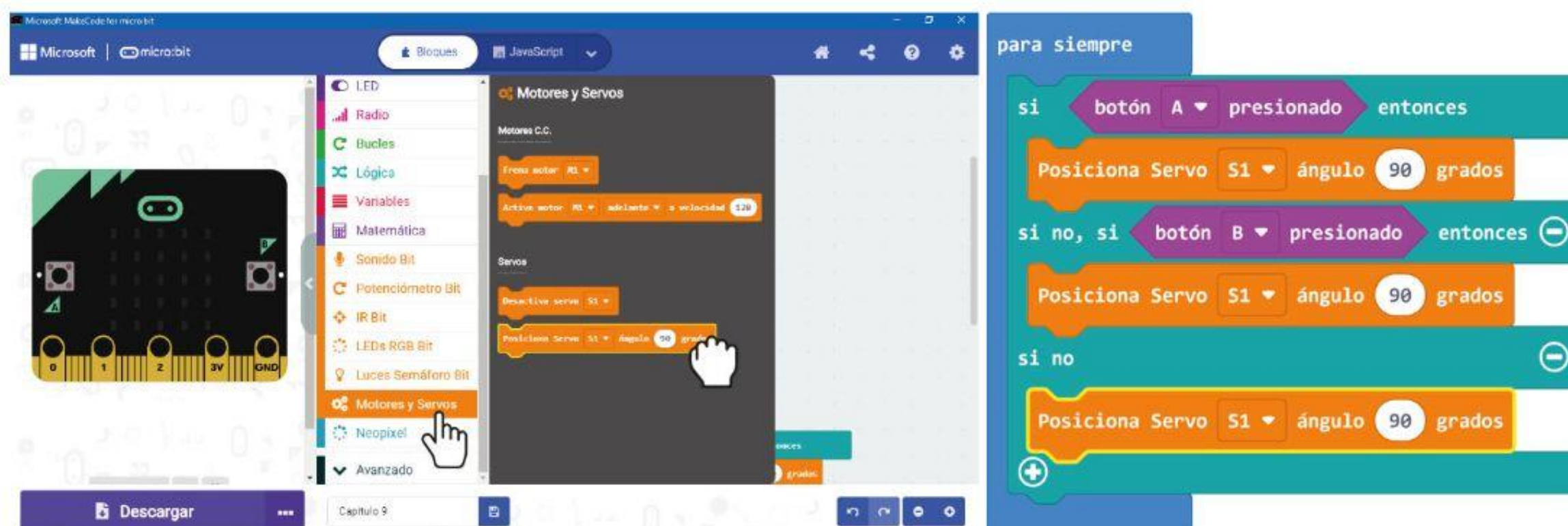
Paso 1 Crea un nuevo proyecto en el editor MakeCode y agrega la extensión EDU: BIT (puedes consultar la página 40). Haz clic en la categoría **[Lógica]** y selecciona el bloque **[si-entonces-si no]**. Encaja el bloque en la ranura **[para siempre]**. Haz clic en el ícono más (+) para agregar otra condición al bloque.



Paso 2 Haz clic en la categoría **[Entrada]** y selecciona el bloque **[botón _ presionado]**. Duplica el bloque y colócalo en las ranuras de condición de bloque **[si no, si _ entonces]**. Cambia el segundo bloque a 'pulsador B'.

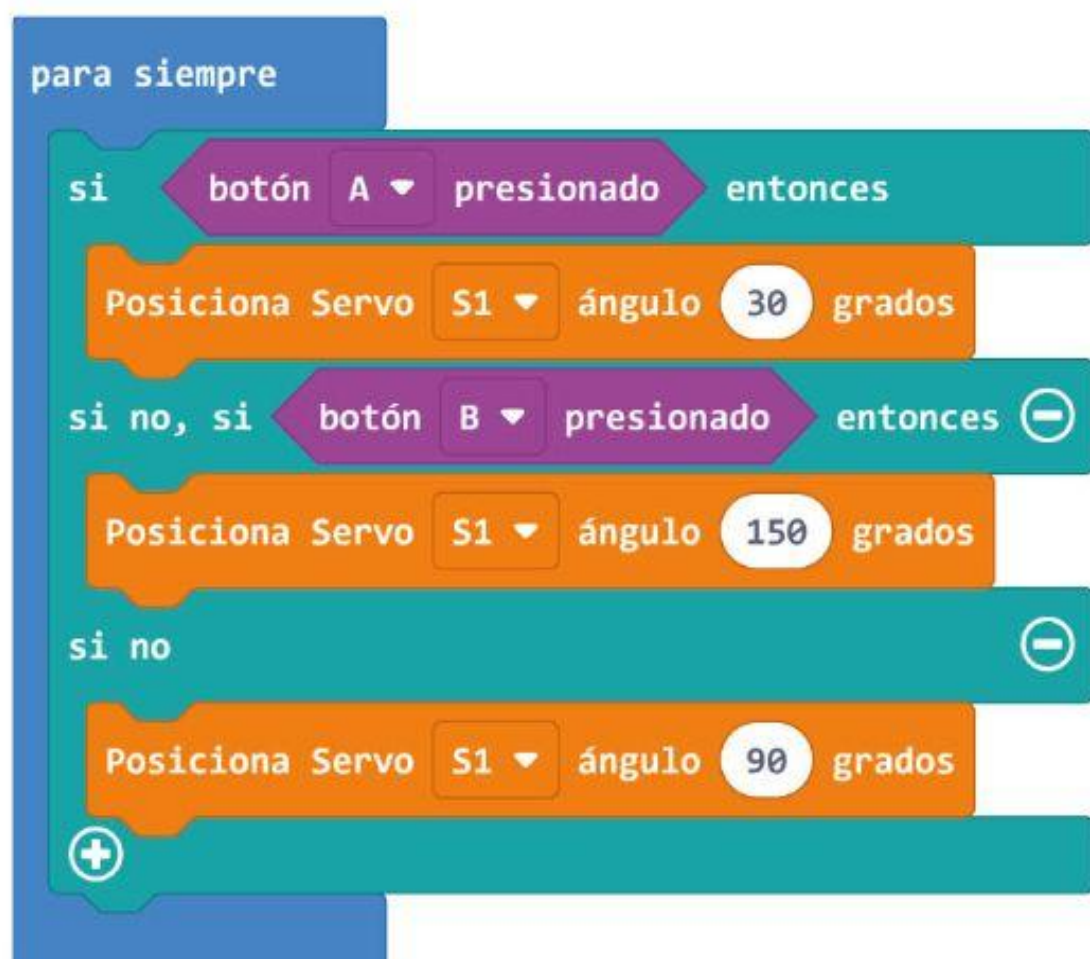


Paso 3 Haz clic en la categoría **[Motores y Servos]** y selecciona el bloque **[Posiciona servo _ ángulo _ grados]**. Duplica el bloque y adjúntalo a cada una de las ranuras del bloque **[si no]**.

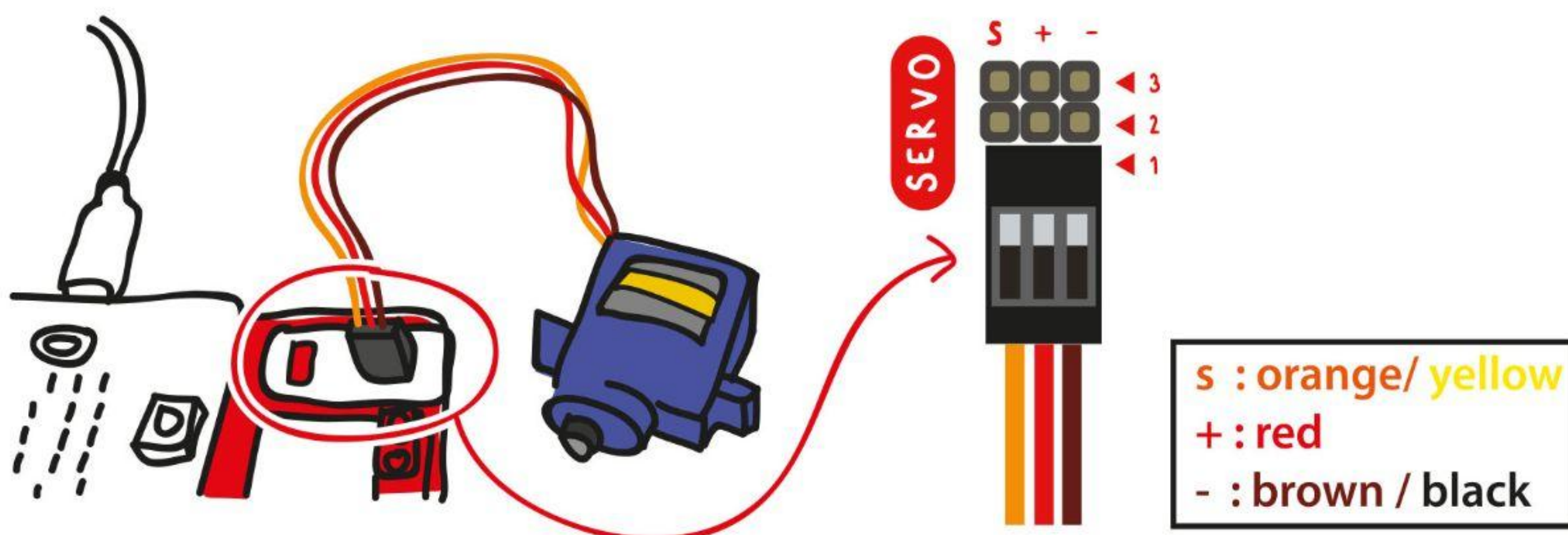




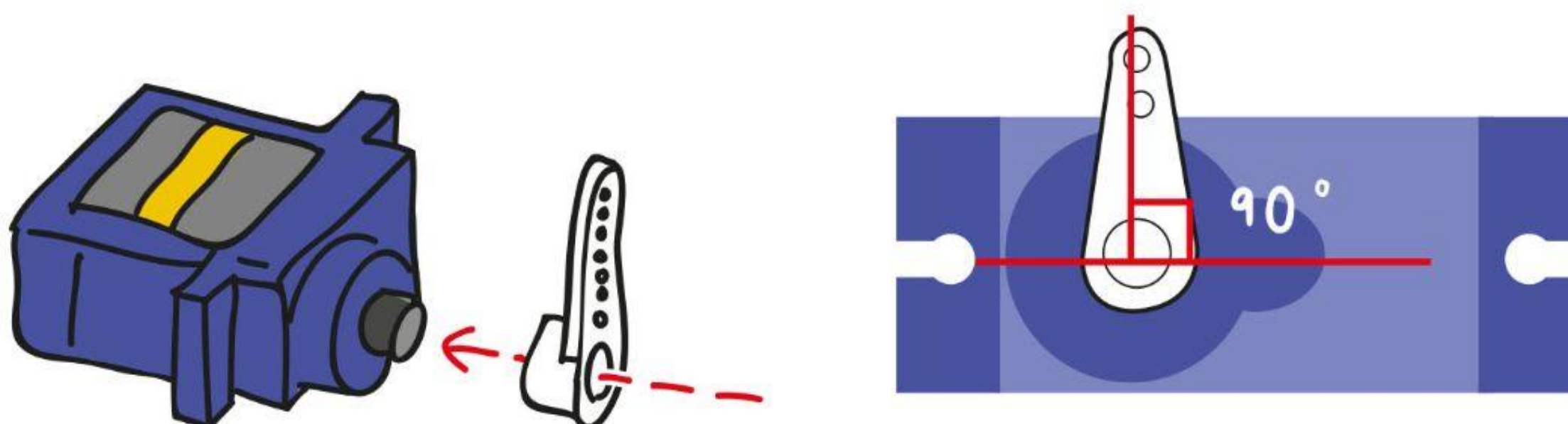
Paso 4 Cambia los valores del primer y segundo bloque a 30 y 150 respectivamente. Transfiere el código a tu EDU: BIT.



Paso 5 Conecta el cable del servomotor al Servo Port 1 en EDU: BIT, como se muestra a continuación.

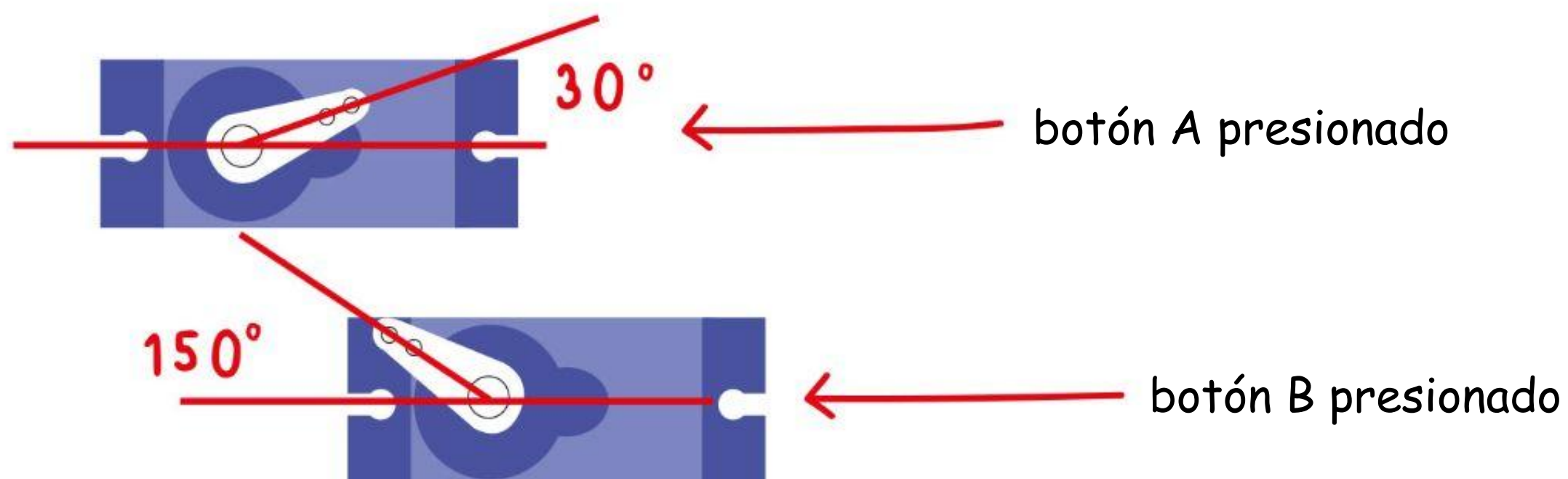


Paso 6 Enciende EDU: BIT y luego conecta un brazo de servo al eje del servomotor a 90 grados, como se muestra a continuación.

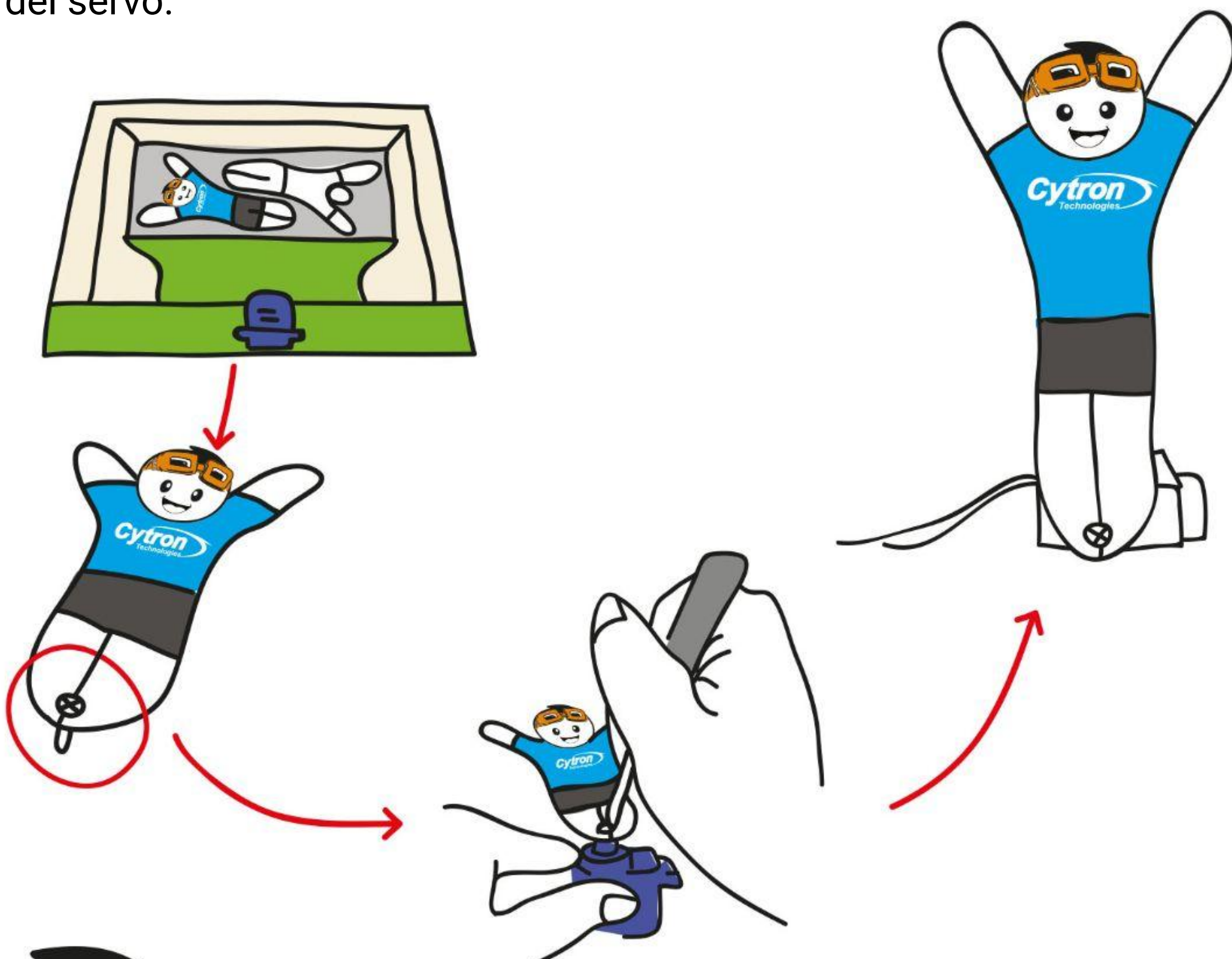


CAPÍTULO 9 : Penalti... ¡Gol!

Paso 7 Presiona el Botón A y luego el Botón B, como prueba.



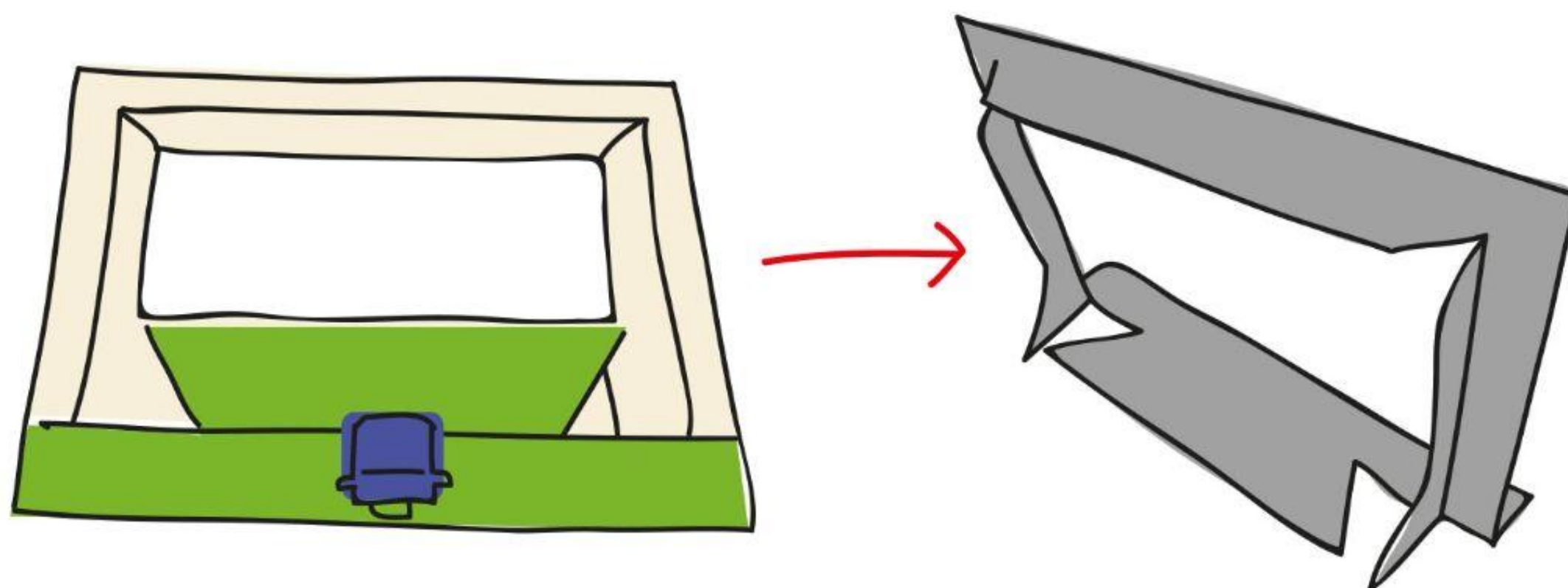
Paso 8 Saca el guardián de la hoja de material provista. Con el destornillador y el tornillo provistos, asegura el portero al brazo del servo. Usa adhesivo adicional, como cinta de doble cara o pegamento caliente, para asegurar firmemente el portero al brazo del servo.



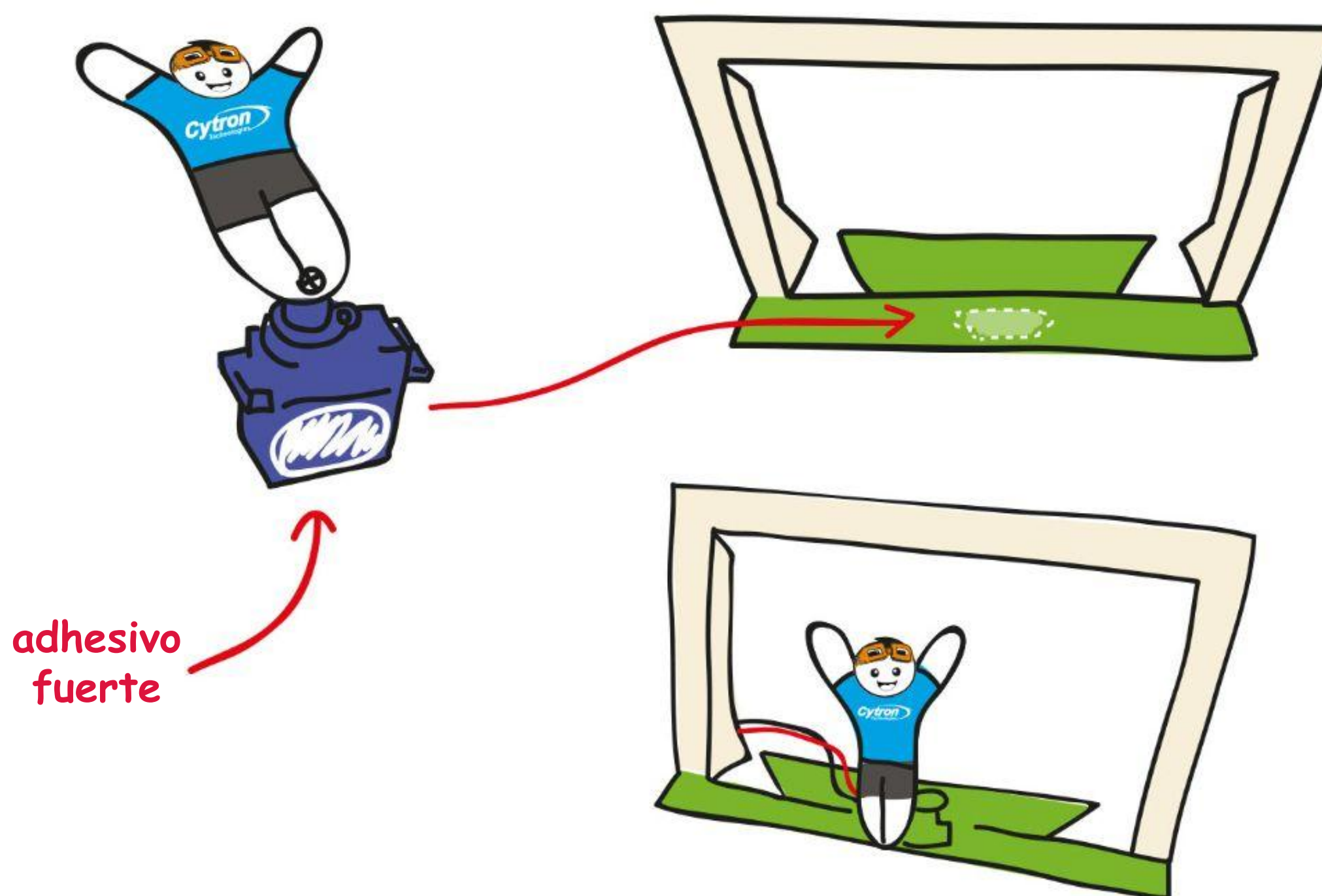
Si quieres, puedes usar la otra imagen proporcionada y diseñar tu propio uniforme para el portero.



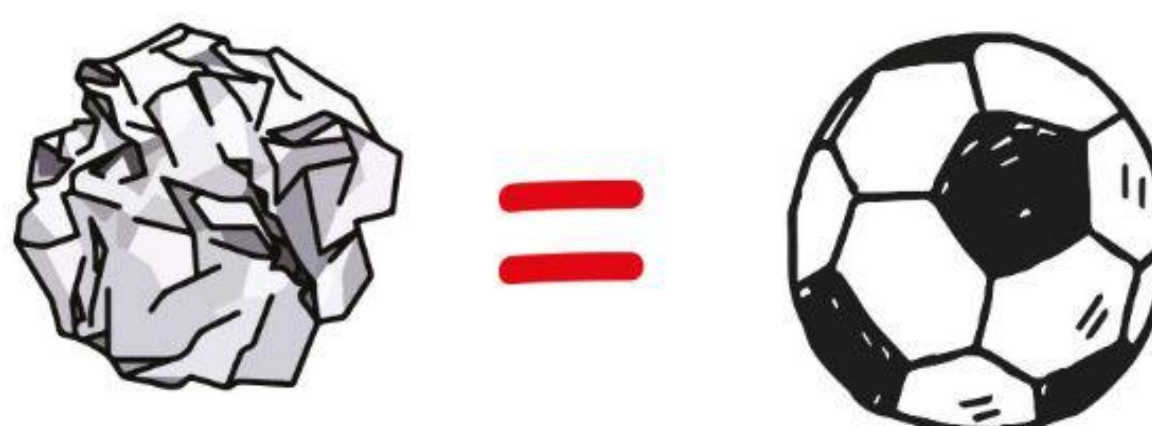
Paso 9 Saca la puerta y colócala como se muestra a continuación.



Paso 10 Usa un adhesivo fuerte, como cinta de doble cara o pegamento caliente, para asegurar firmemente el servomotor en su lugar como se indica.

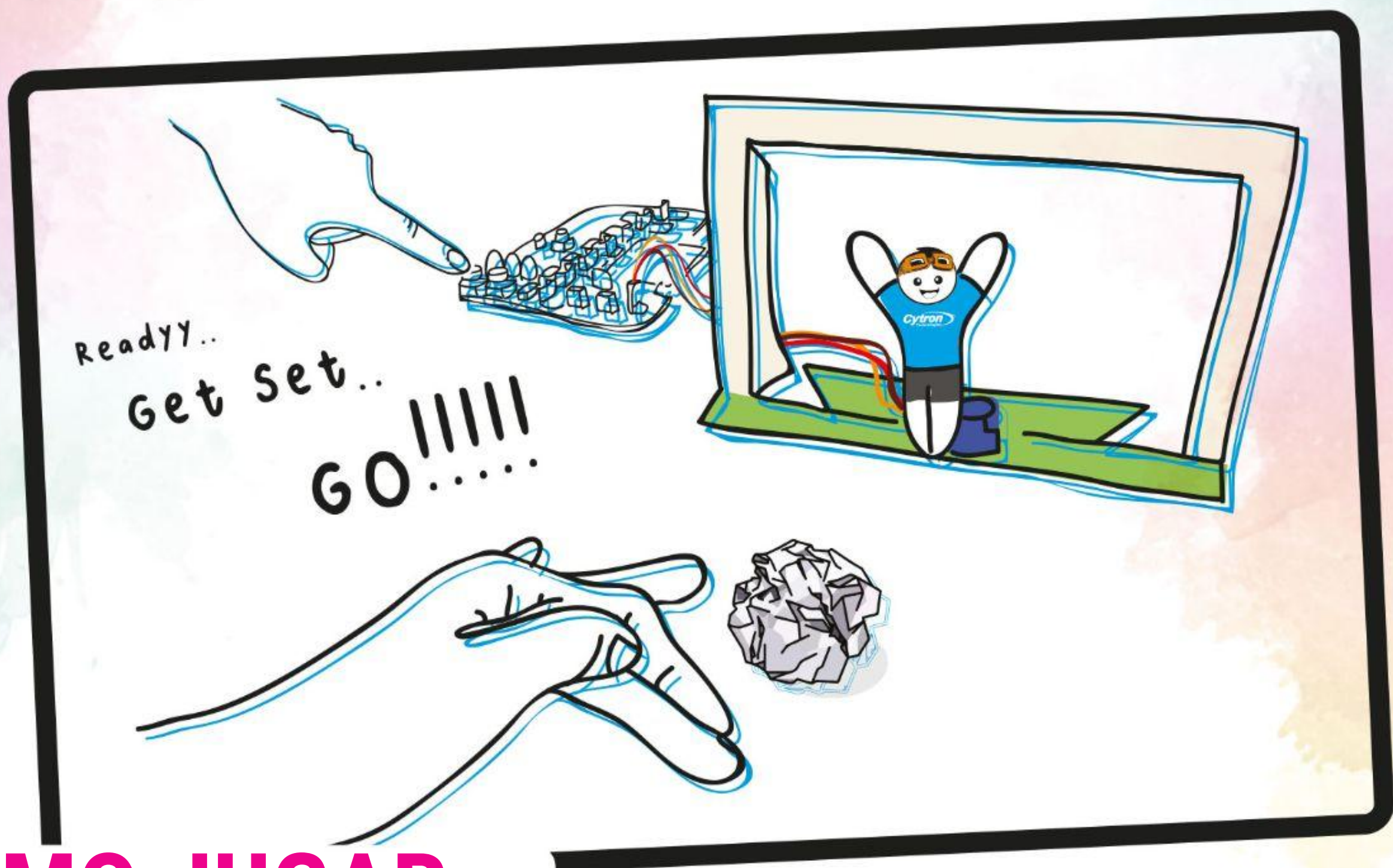


Paso 11 Arruga un pequeño trozo de papel para usarlo como "balón de fútbol" para que tengamos todo lo que necesitamos para una tanda de penaltis divertida. ¿Estás listo?



Vamos a jugar

Penalti... Gol !!!



CÓMO JUGAR:

Colocar la portería y marcar el lugar desde el que ejecutar el penalti (aproximadamente a 1 metro de la portería, ajustando la distancia a los jugadores más pequeños).

Los jugadores se turnan para ser el futbolista o el portero.

El futbolista golpea el balón con el dedo hacia la portería.

El portero intenta bloquear el balón moviéndose hacia la izquierda presionando el botón amarillo (A) o hacia la derecha presionando el botón azul (B).

Para cada ronda, cada jugador tiene 5 posibilidades. El jugador que anota más goles es el ganador.



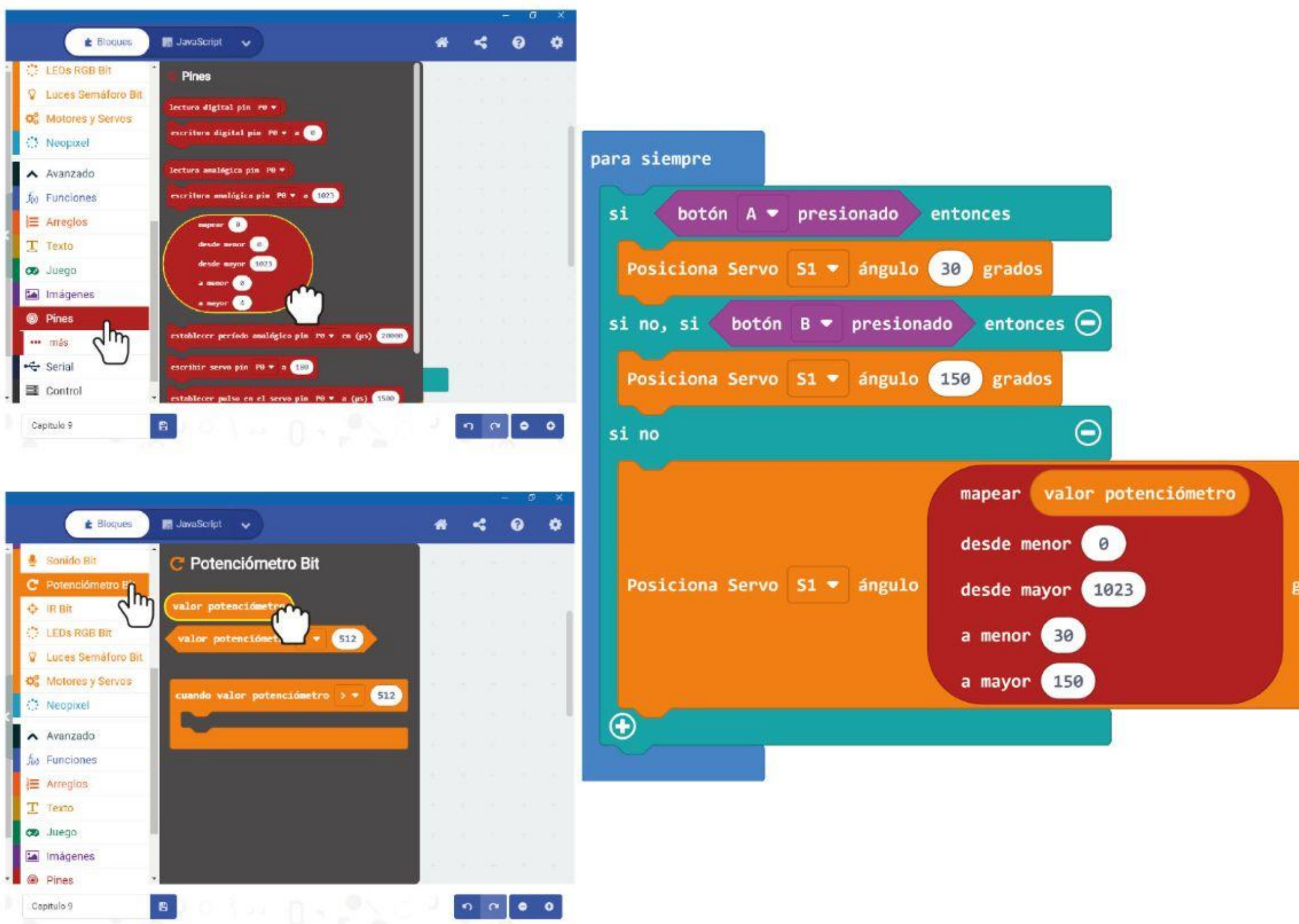
¿Sabías que? Los tiros de penalti se juegan para determinar el equipo ganador en un partido de fútbol, cuando el marcador es igual al final del partido y se mantiene igualado después de la prórroga. Cada equipo puede hacer cinco tiros. El jugador tira desde el punto de penalti hacia la portería defendida únicamente por el portero del equipo contrario. La victoria se otorga al equipo que anota más goles.



En el código anterior, giramos al portero hacia la izquierda o hacia la derecha presionando los botones. Podemos modificar el código para controlar la posición del portero usando el Potenciómetro Bit.

Paso 12 Haz clic en la categoría **[Avanzado]** y selecciona la categoría **[Pin]**. Añade el bloque **[mapear _ desde menor _ desde mayor _ a menor _ a mayor _]** a tu código.

Paso 13 Haz clic en la categoría **[Potenciómetro Bit]** y selecciona el bloque **[valor potenciómetro]**. Encaja el bloque en el bloque **[mapear _ desde menor _ desde mayor _ a menor _ a mayor _]** y cambia los dos últimos valores a **30** y **150** respectivamente.



Paso 14 Transfiere el código a tu EDU: BIT. Ahora puedes controlar al portero usando el potenciómetro de bits. ¡Que lo pases bien!

Si quieres practicar los lanzamientos de penaltis tú mismo, puedes cambiar el código en el "modo de entrenamiento", lo que hace que el portero gire a la izquierda y a la derecha continuamente. ¡Prueba!



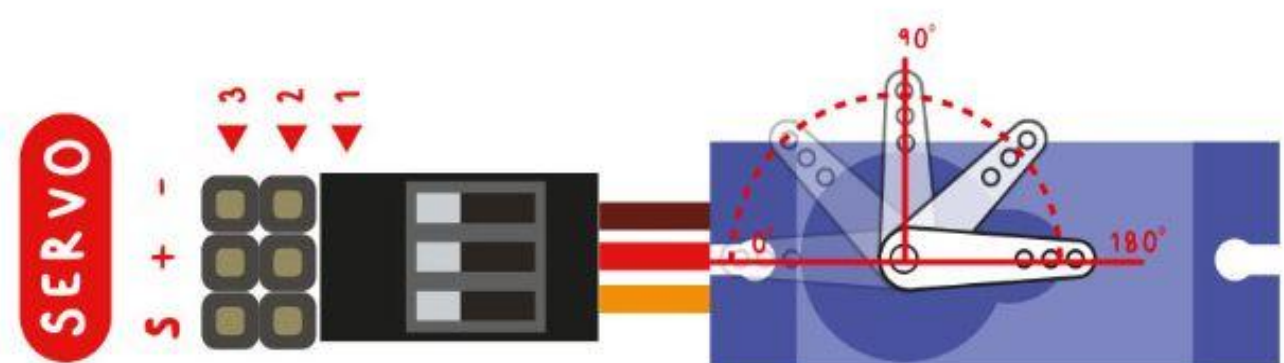
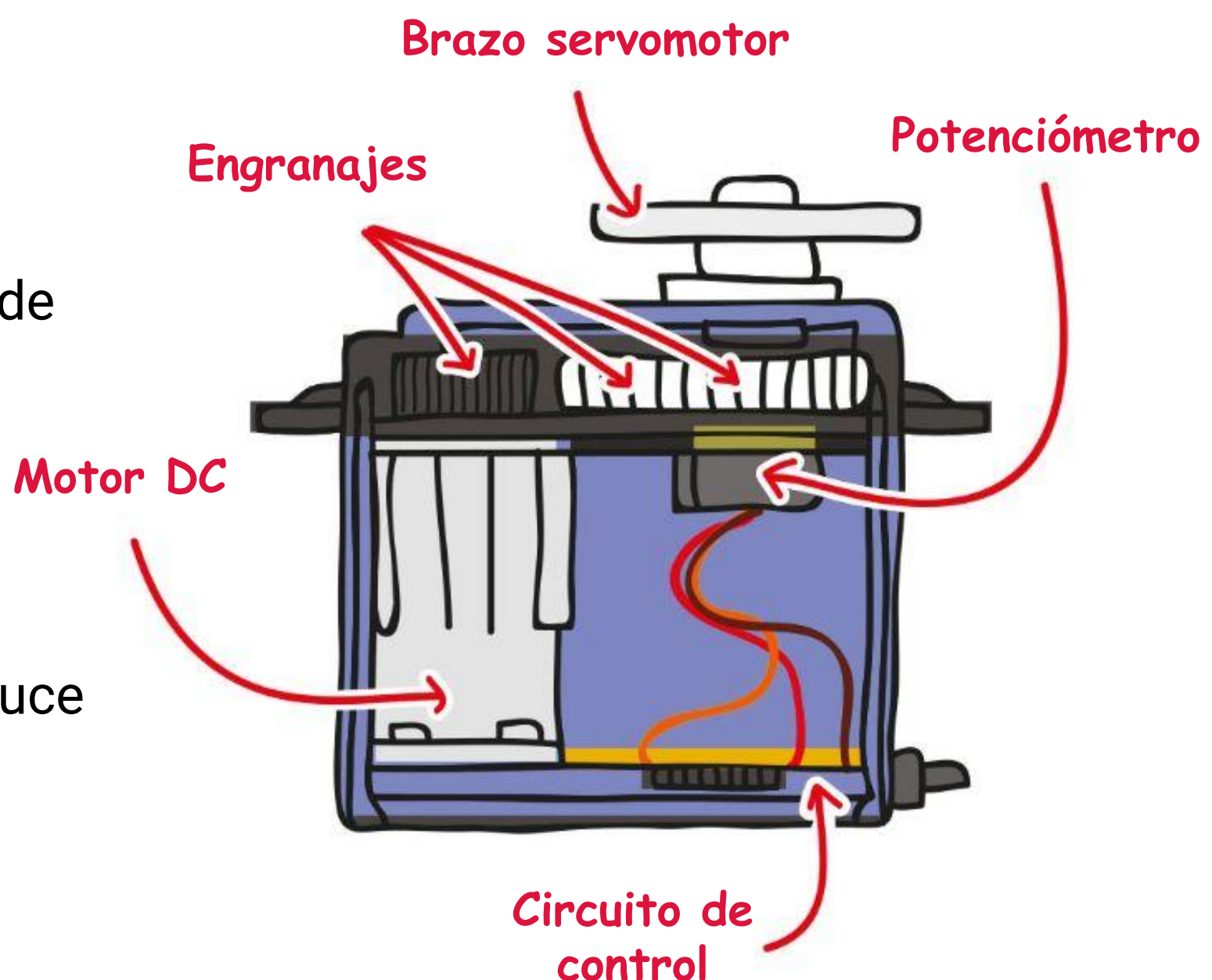
¡COSAS DIVERTIDAS!

El servomotor del kit EDU: BIT también se conoce como servomotor RC (controlado por radio). Es muy utilizado en coches de juguete controlados por radio y en pequeños robots para controlar sus movimientos.

Un servomotor utiliza un sistema de tres hilos para alimentación (+), tierra (-) y señal(es) de control. Por lo general, consta de un motor de CC, engranajes, un potenciómetro (sensor de posición) y un circuito de control.

El controlador integrado traduce los comandos en forma de pulsos en posiciones expresadas en grados. El servomotor girará constantemente y permanecerá en la posición que corresponda a los pulsos recibidos.

A diferencia del motor de CC que gira continuamente, podemos controlar la rotación de un servomotor hasta un ángulo deseado en un rango de 0 a 180 grados.

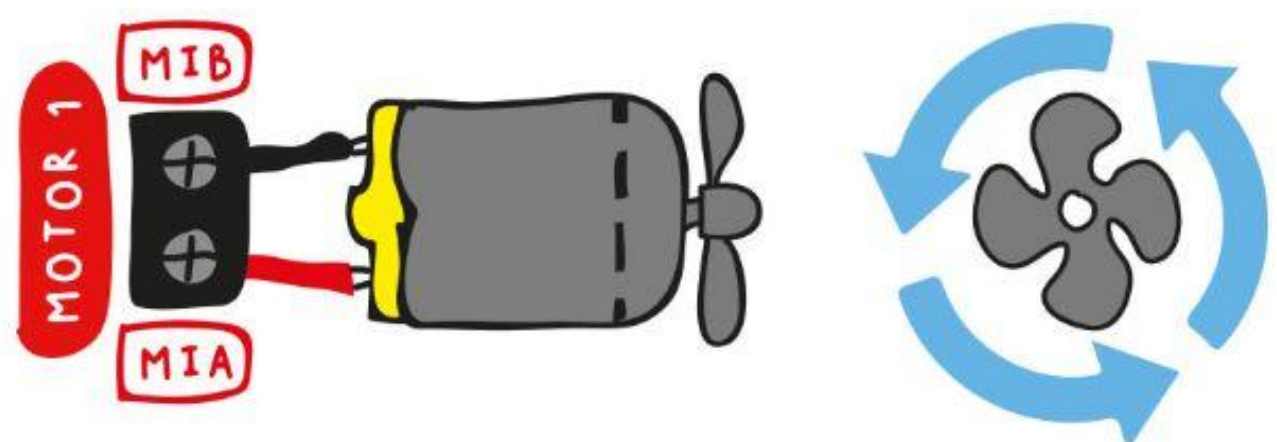


rotación 0-180°

Servomotor

V S

Motor DC



rotación 360°

DESAFÍO DE LA APLICACIÓN

Programa EDU: BIT para que funcione como un metrónomo. Cuando esté encendido, gira el puntero (unido al brazo del servo) hacia la izquierda y luego hacia la derecha repetidamente a un ritmo constante. Configura el tiempo para que sea controlable por el bit del potenciómetro. Cuando se presiona el botón amarillo (botón A), muestra el tiempo actual (por ejemplo, 120 pm).

Aquí tienes algunos consejos ...

1: Necesitas crear dos variables: Tiempo y Retraso.

2: Un rango típico de un metrónomo está entre 40 y 200 bpm.

#3: Un tiempo de 60 pm (o pulsaciones por minuto) significa que la aguja oscila de un extremo al otro un total de 60 veces en un minuto, es decir, una vez cada segundo.

4: Cuanto más rápido sea el tiempo, menor será el retraso.



"Un metrónomo es un dispositivo que produce un sonido a intervalos regulares que el usuario puede configurar, normalmente en pulsaciones por minuto (pm). Los músicos usan el dispositivo para practicar tocando con un tempo musical constante. Los metrónomos suelen incluir un movimiento sincronizado visible".

wikipedia -

CAPÍTULO 10

Mastermind, ¿puedes encontrar el código secreto?

LEDs RGB Bit

buscador de código



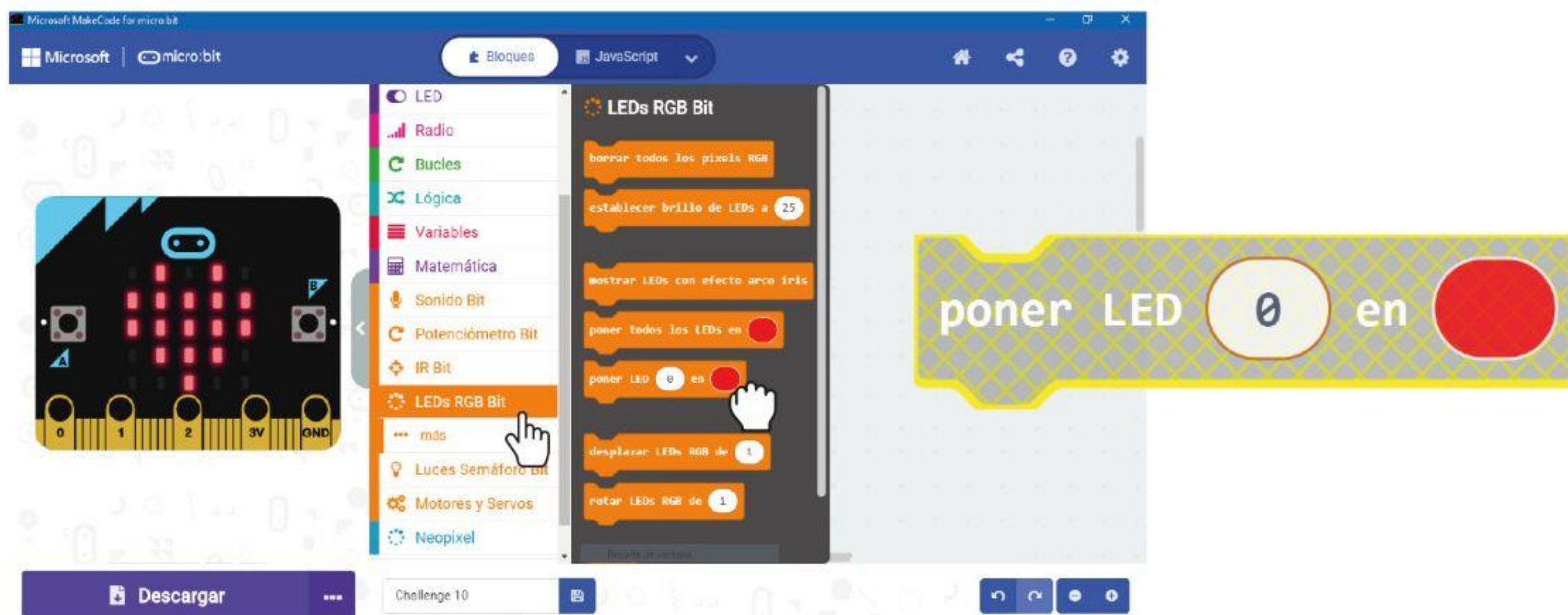
creador de código



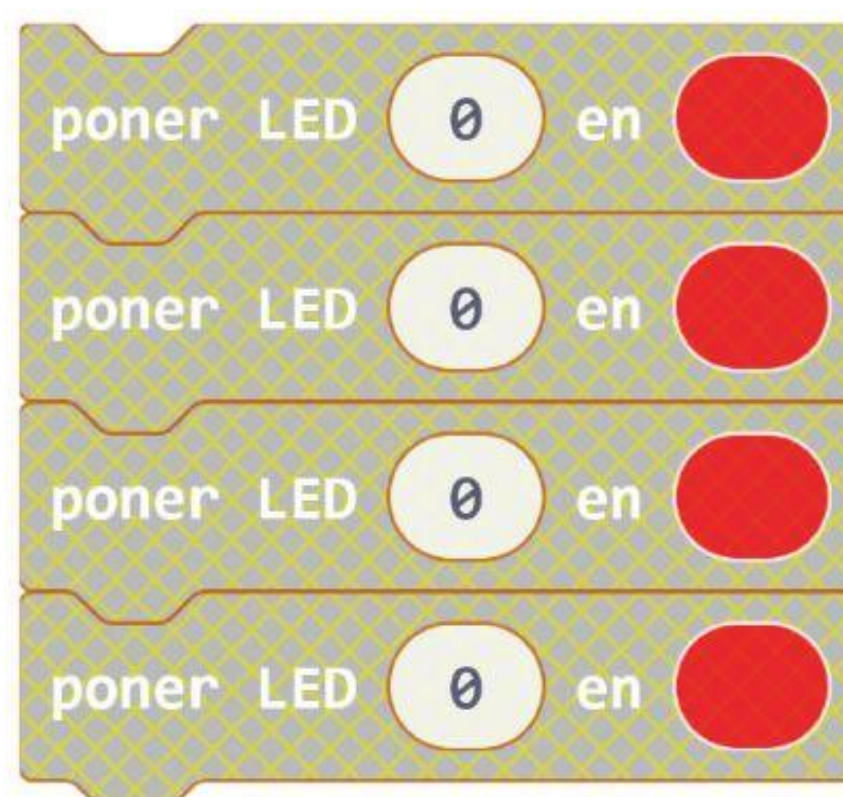


PROGRAMAMOS!

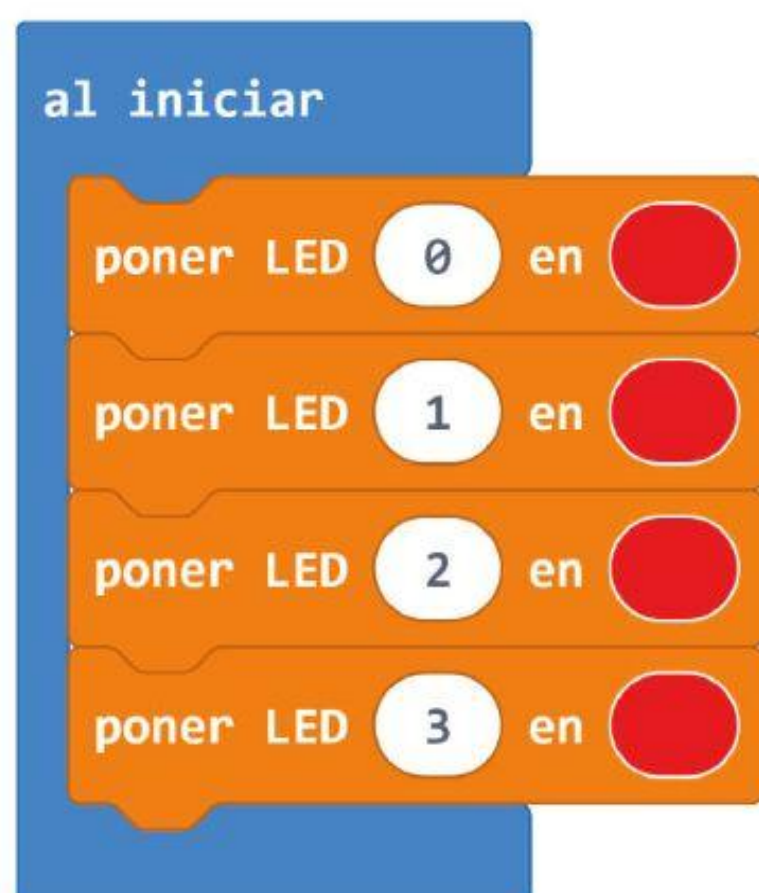
Paso 1 Crea un nuevo proyecto en el editor de MakeCode y agrega la extensión EDU: BIT. Haz clic en la categoría **[LEDs RGB Bit]** y selecciona el bloque **[poner LED _ en _]**.



Paso 2 En el espacio de trabajo, haz clic derecho en el bloque **[poner LED _ en _]** y luego haz clic en 'Duplicar'. Repite hasta que tengas 4 bloques. **[poner LED _ en _]**.



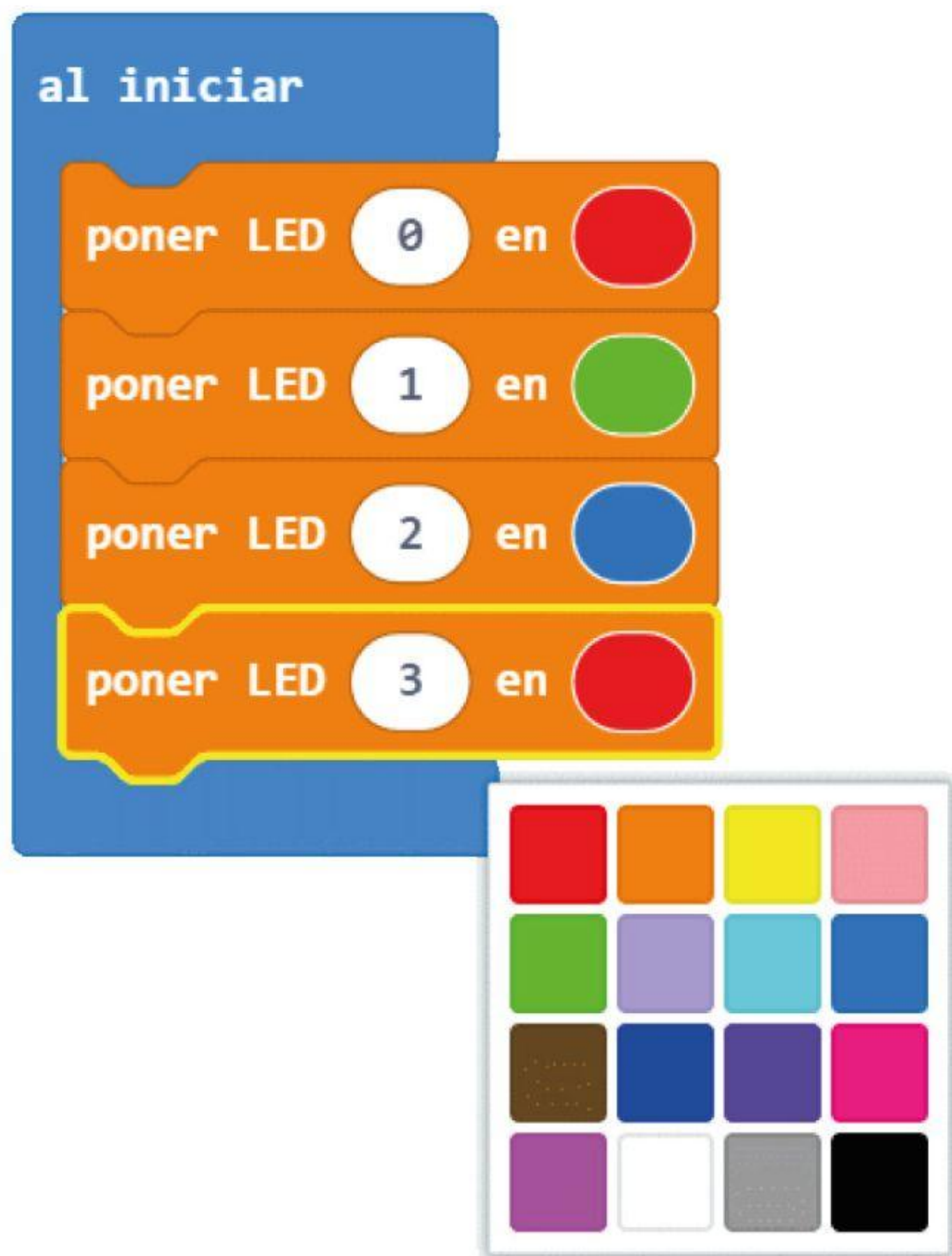
Paso 3 Coloca los bloques en la ranura del bloque. **[al iniciar]**. Cambia el número de píxeles de 0 a 1, 2 y 3 para el segundo, tercer y cuarto bloque.



Hay 4 LED RGB en el bit RGB y a cada uno se le asigna un número de identificación (de 0 a 3). Usa este número para programar cada LED individualmente.

CAPÍTULO 10 : Mastermind

Paso 4 Transfiere el código a tu EDU: BIT.



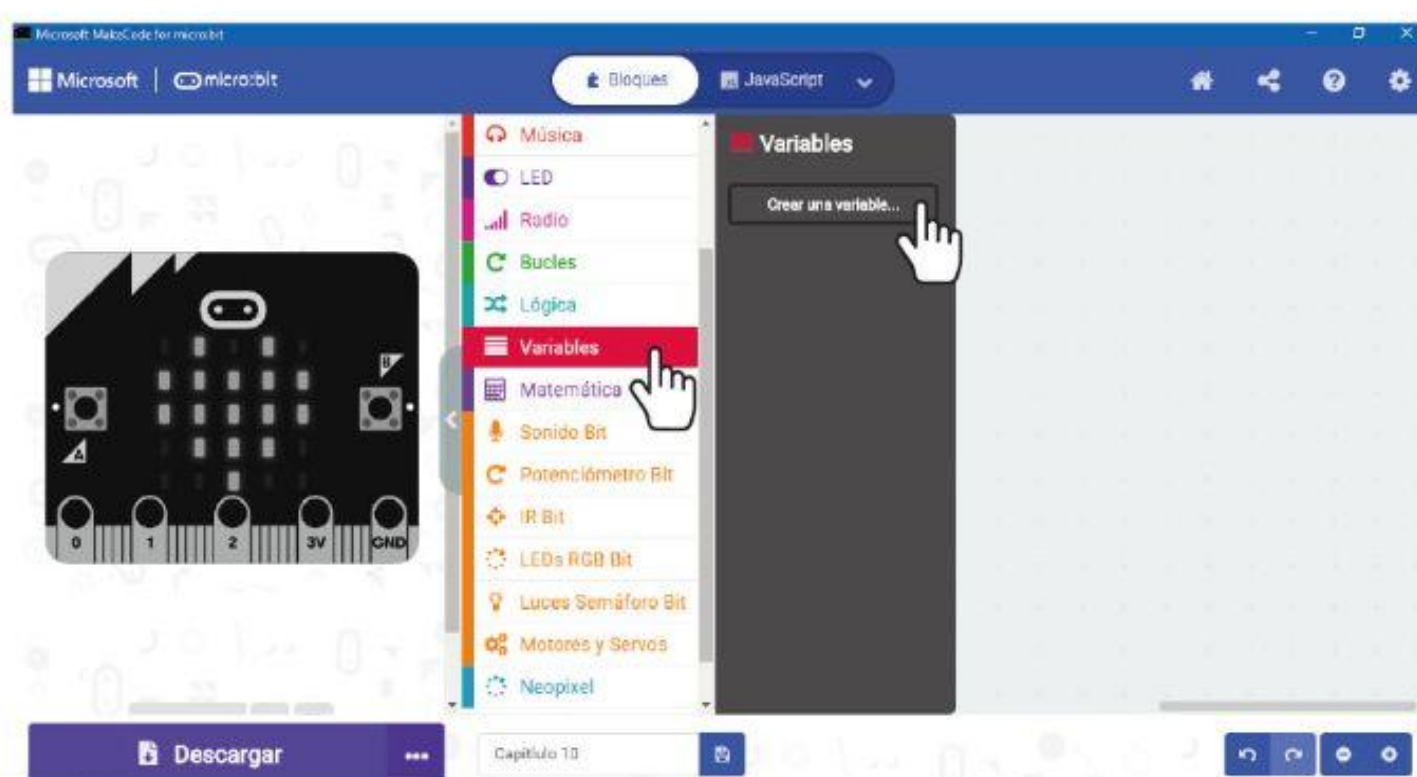
Cuando enciendas la placa, los cuatro LED en el bit RGB se encenderán de acuerdo con los colores establecidos.

Puedes cambiar fácilmente el color haciendo clic en el bloque y seleccionando otro color disponible de la paleta de colores.

¡Pruébalo!



Paso 5 Agrega dos nuevas variables y llámalas - “Color y posición correctos” e “Color correcto pero posición incorrecta”.



Nombre de la nueva variable:

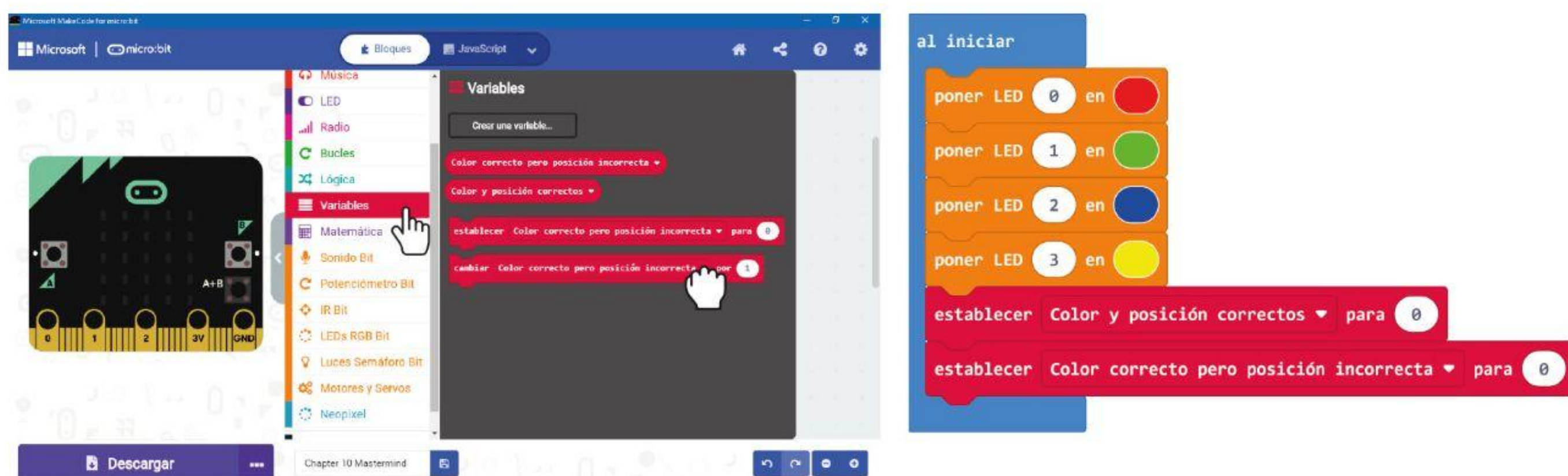
Aceptar

Nombre de la nueva variable:

Aceptar



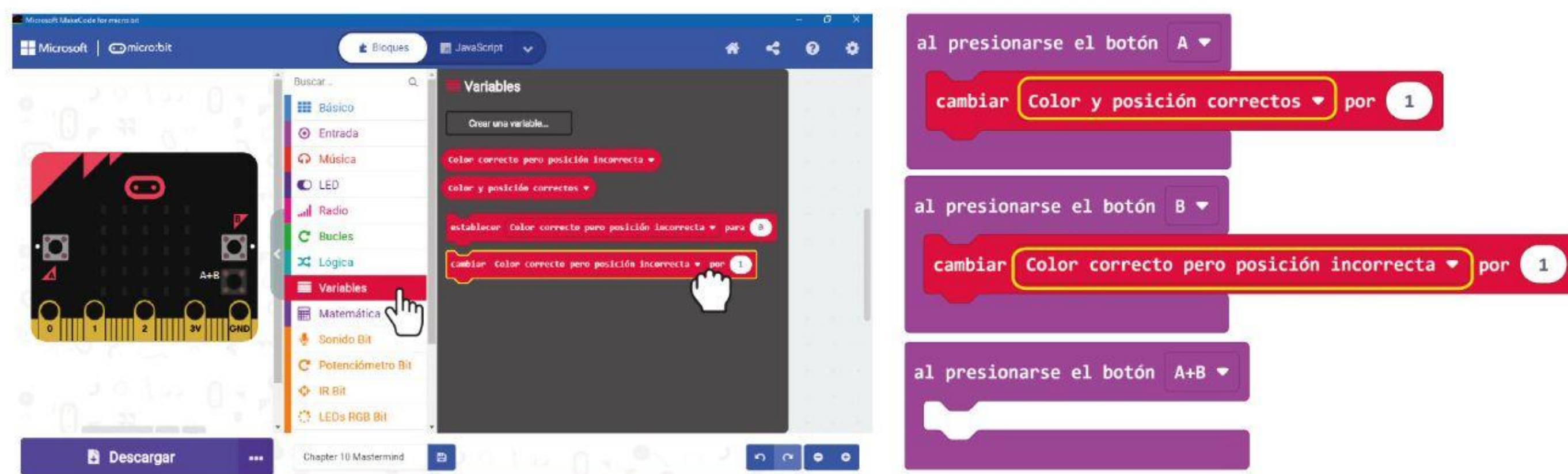
Paso 6 Haz clic en la categoría **[Variables]** y selecciona el bloque **[establecer _ para _]**. Duplica el bloque y encaja ambos en el bloque **[al iniciar]**. Establece una variable para “Color y posición correctos” y la otra para “Color correcto pero posición incorrecta”.



Paso 7 Haz clic en la categoría **[Entrada]** y selecciona el bloque **[al presionarse el botón _]**. Duplica el bloque. Cambia el segundo bloque a **B** y el tercer bloque a **A + B**.

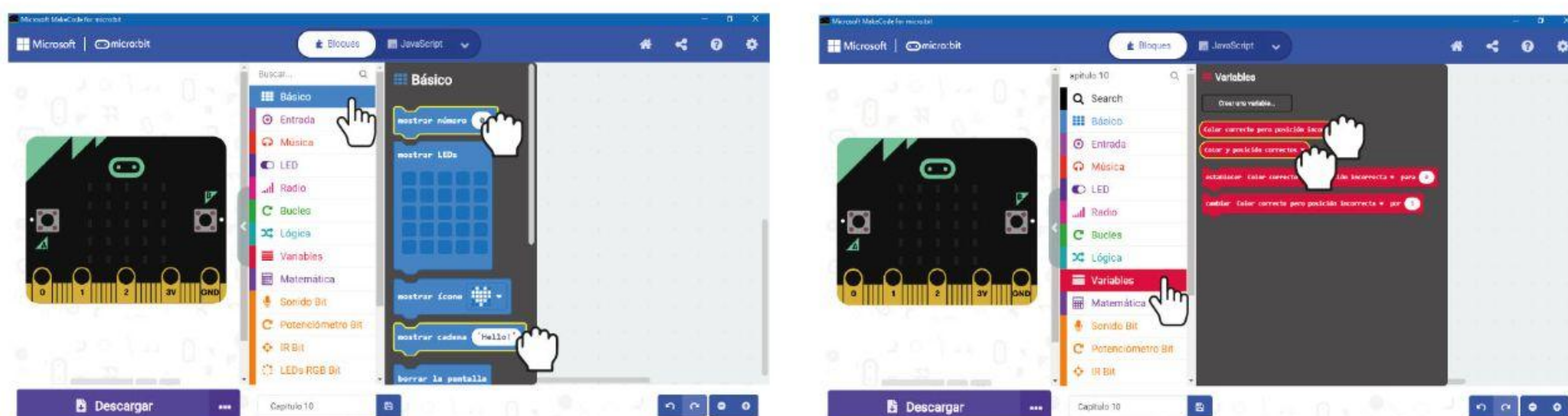


Paso 8 Haz clic de nuevo en la categoría **[Variables]** y selecciona el bloque **[cambia _ por _]**. Duplica y pega los bloques en los bloques **[al presionarse el botón A]** y **[al presionarse el botón B]**. Establece una variable para “Color y posición correctos” y la otra para “Color correcto pero posición incorrecta”.

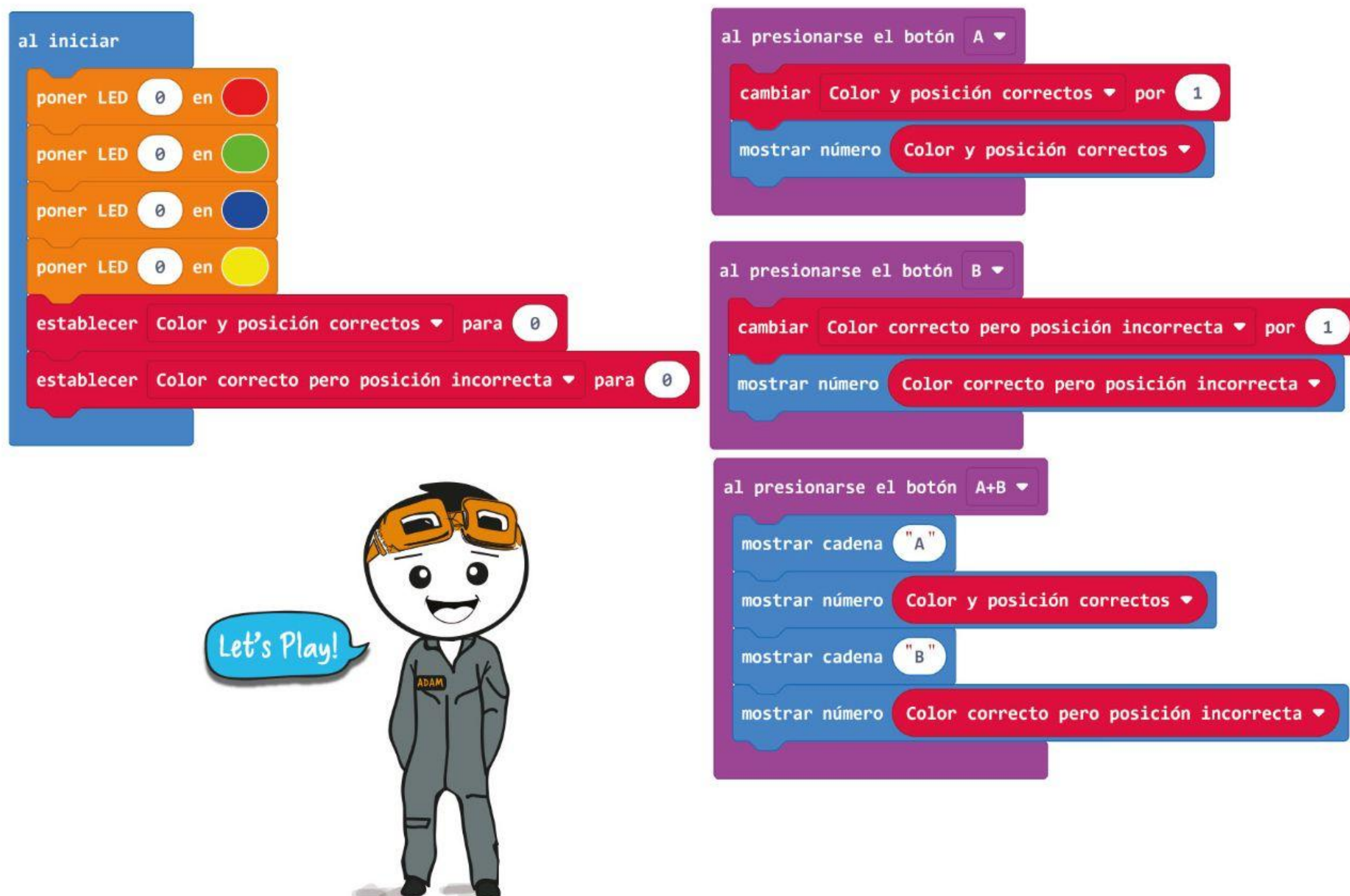


CAPÍTULO 10 : Mastermind

Paso 9 Agrega [mostrar número] y [mostrar cadena] de la categoría [Básico], así como bloques [Color y posición correctos] y [Color correcto pero posición incorrecta] de la categoría [Variables].



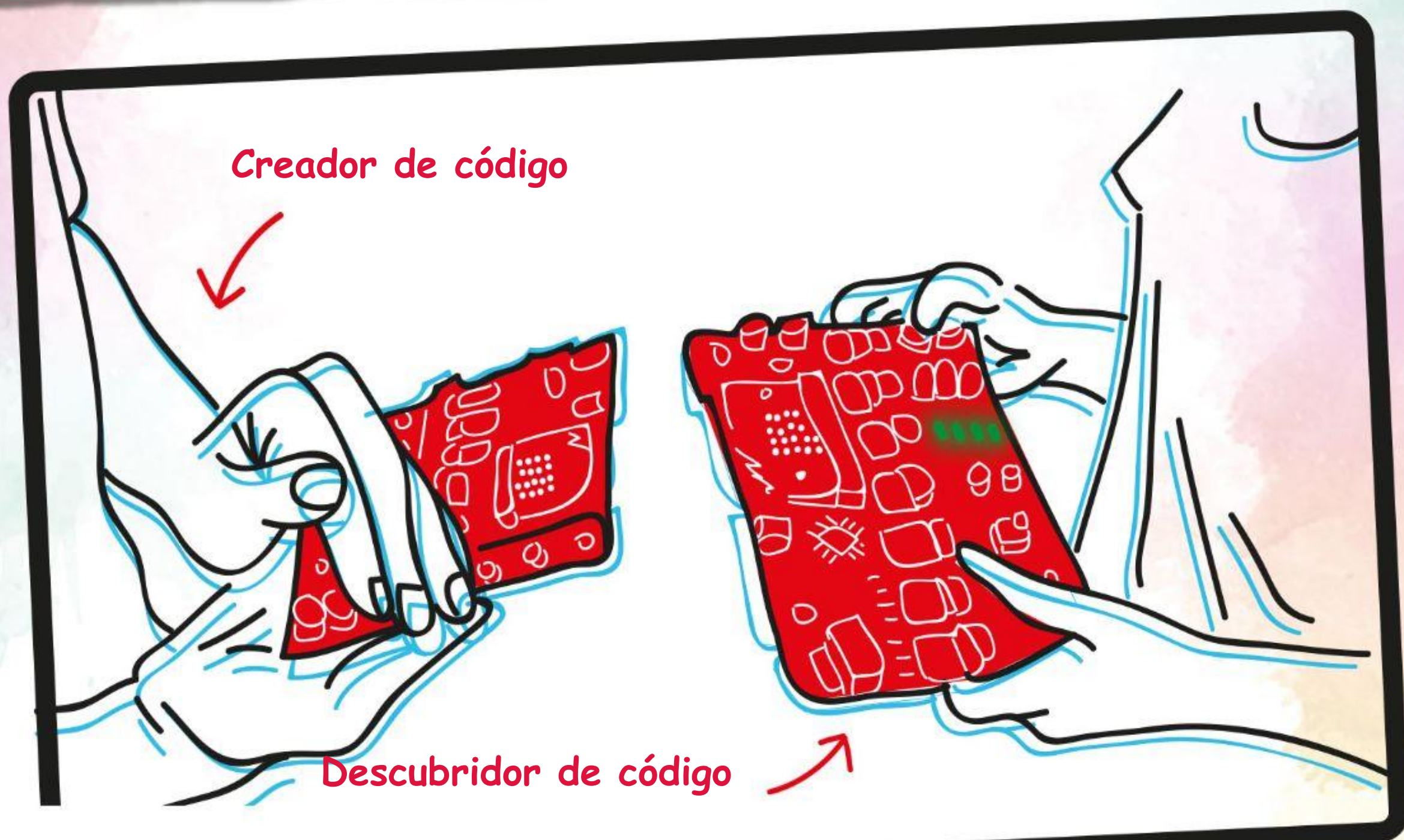
Paso 10 Cambia "Hello!" en el bloque [mostrar cadena] por "A =" y "B =" respectivamente. Aquí está el código completo.



Paso 11 Transfiere el código a tu EDU: BIT. Ahora estás listo para jugar a Mastermind.

Vamos a jugar

Mastermind, ¿puedes encontrar el código secreto?



CÓMO JUGAR:

- El jugador 1, el creador del código, establece el código secreto encendiendo los cuatro LED al azar. Para comenzar, limita los colores a solo rojo y verde. Recuerda tapar tu tarjeta para que el otro jugador no pueda ver la secuencia de colores.
- El jugador 2, el buscador de códigos, intenta adivinar el código secreto. El jugador 2 enciende los LED RGB en su EDU: BIT y se los muestra al Creador de Código.
- Verifica el código y luego presiona el botón amarillo (A) y / o el botón azul (B) en su EDU: BIT para indicar cuántos LED están encendidos con "color y posición correctos" y "color correcto pero posición incorrecta".
- El Buscador de Código puede presionar ambos botones simultáneamente para "leer" el resultado.
- Repite los pasos 2 y 3 hasta que el Buscador de Código adivine correctamente la secuencia de colores (máximo 10 intentos).
- Cambia los papeles para jugar otra ronda.

Cómo ganar:

Ganas el juego si encuentras el código (adivina la secuencia de colores). Si fallas, la victoria es para el Creador de Código.



EXPLORA OTROS BLOQUES

1 Puedes configurar los LED en el bit RGB para mostrar los colores del arco iris reemplazando el bloque [poner LED _ en] con el bloque [mostrar LEDs con efecto arco iris].

al iniciar

mostrar LEDs con efecto arco iris

2 Puedes crear un efecto de movimiento brillante colocando un bloque [desplazar LEDs RGB de _] en un bloque [para siempre]. Recuerda agregar un bloque [pausa] para ralentizar el programa para que pueda ver el efecto. Aquí hay un código de muestra:

al iniciar

poner LED 0 en 
poner LED 1 en 
poner LED 2 en 
poner LED 3 en 

para siempre

rotar LEDs RGB de 1
pausa (ms) 500

3 También puedes traducir píxeles uno por uno usando el bloque [rotar LEDs RGB de _] en un bloque [para siempre]. Deberás agregar un bloque [pausa] para ralentizar el programa y poder ver el efecto, mientras que los píxeles se apagan uno por uno. Aquí hay un código de muestra:

al iniciar

poner LED 0 en 
poner LED 1 en 
poner LED 2 en 
poner LED 3 en 

para siempre

desplazar LEDs RGB de 1
pausa (ms) 500

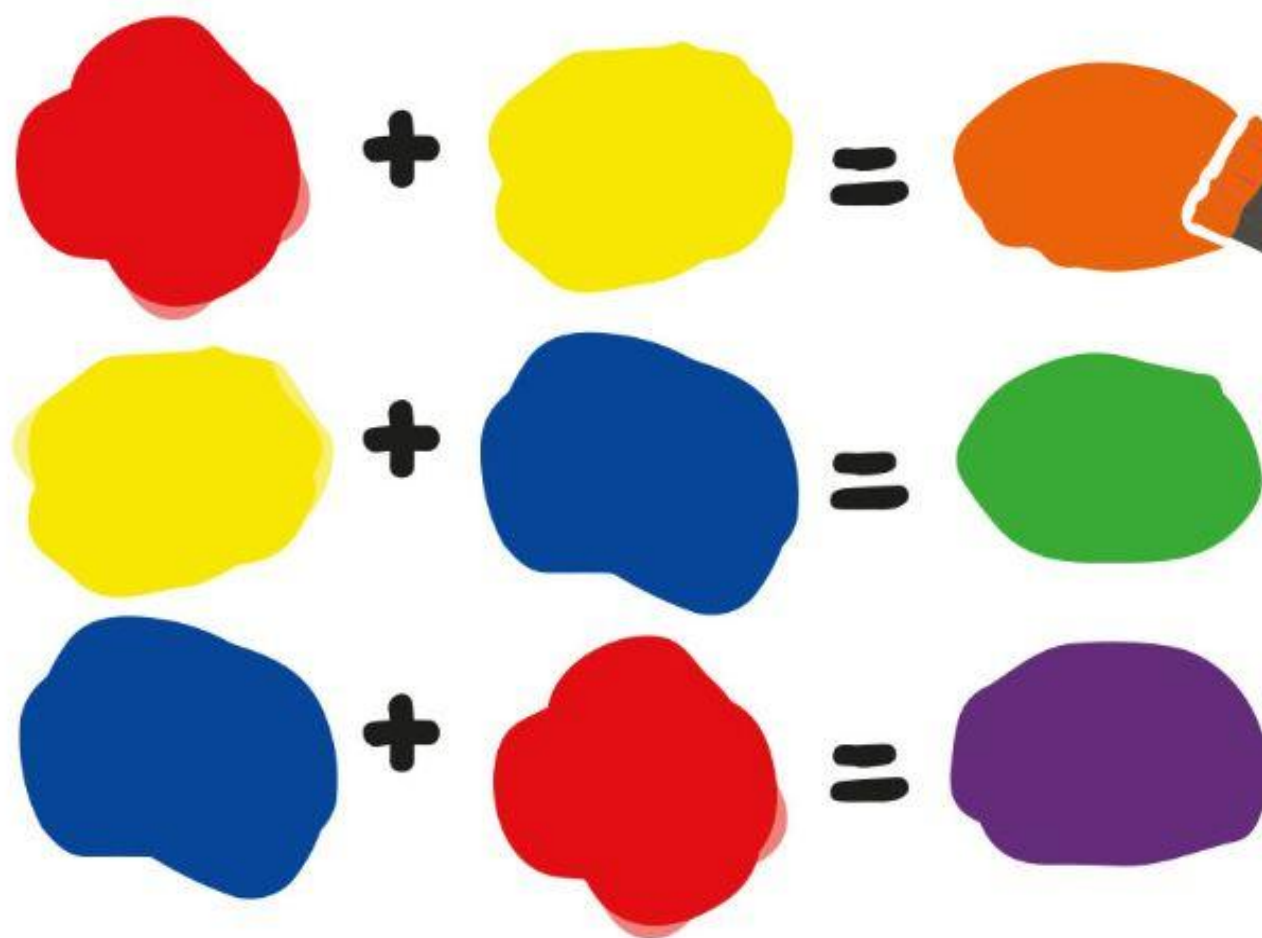


Puedes cambiar la dirección del efecto para los puntos # 2 y # 3 anteriores cambiando la configuración de un valor positivo a un valor negativo.

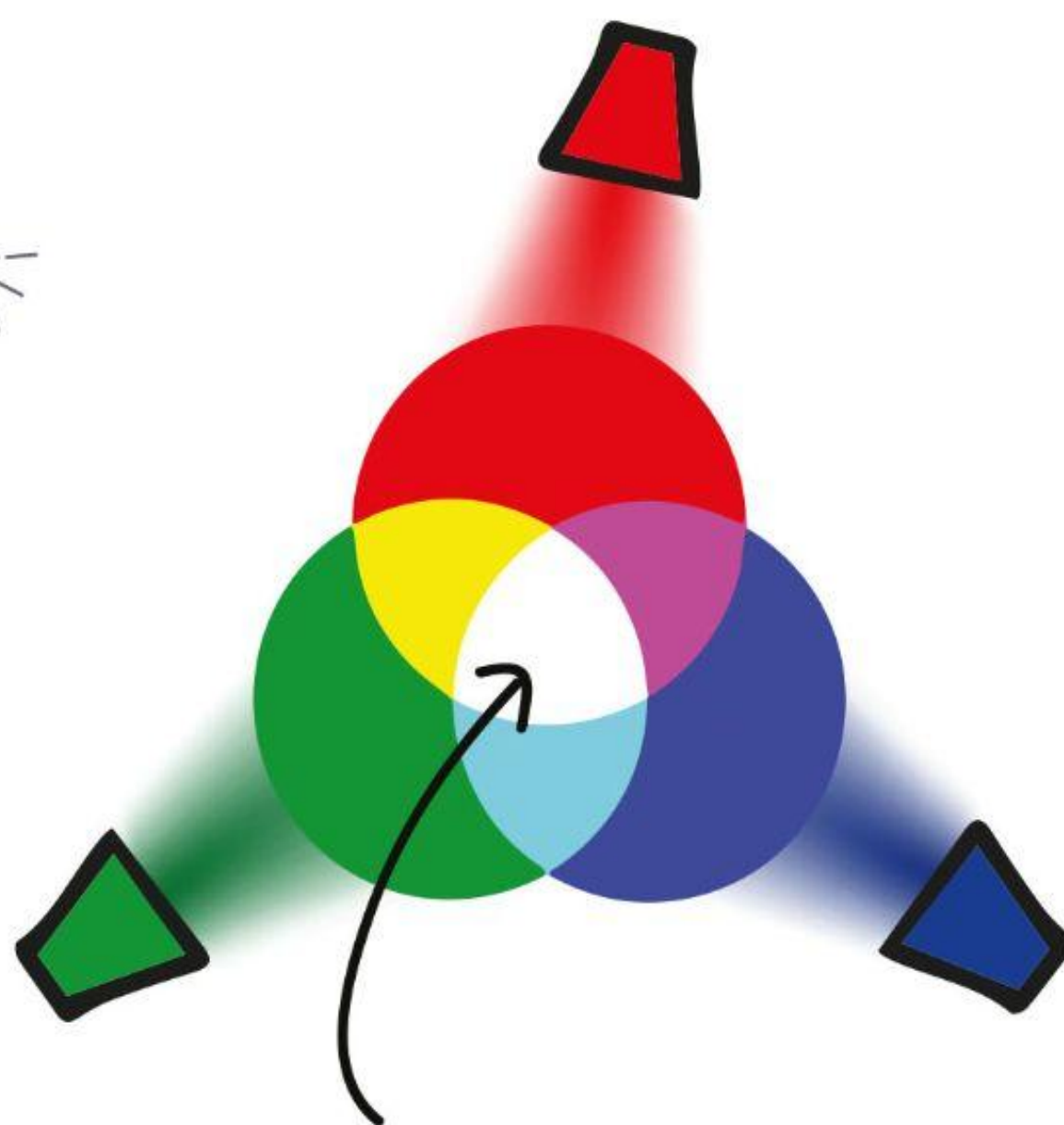
¡COSAS DIVERTIDAS!



En la clase de arte probablemente hayas aprendido que los 3 colores primarios son rojo, amarillo y azul. Y cuando los mezclas obtienes este resultado:



Sin embargo, para dispositivos que usan luz para mostrar colores, por ejemplo, la pantalla de tu televisor y computadora, se utiliza el modelo de color RGB.



En este modelo los tres colores primarios son en cambio rojo (R), verde (G) y azul (B); ¡y cuando se combinan crean una luz blanca!

DESAFÍO DE LA APLICACIÓN

Programa EDU:BIT para hacerlo funcionar como dispositivo para entrenar juegos de memoria.

¿Cómo funciona?

- Para comenzar, incline EDU: BIT hacia la izquierda para encender los LED del bit RGB con una secuencia de colores aleatoria durante unos segundos.
- Deberás observar y luego decir la secuencia de colores después de que los LED se hayan apagado.
- Para verificar la respuesta, presione el botón azul (B) para encender los LED en la misma secuencia que antes.
- Si respondes correctamente, presiona el botón amarillo (A) para actualizar y ver su puntaje. El juego termina si respondes mal.
- Puedes ajustar el nivel de dificultad del juego girando el eje del potenciómetro de bits para aumentar o disminuir el tiempo que los LED permanecen encendidos.
- ¡El jugador con el más alto puntaje gana!

Aquí tienes algunos consejos:

1: debes crear dos variables: puntaje y secuencia.

2: primero debes configurar la secuencia de colores (el color de cada LED RGB). Usa más colores o más secuencias para hacer el juego más difícil. Y viceversa, es posible que desees limitar la cantidad de colores/secuencias para los jugadores más jóvenes.



CAPÍTULO BONUS

Juego Simon con LED

Radio Comunicación

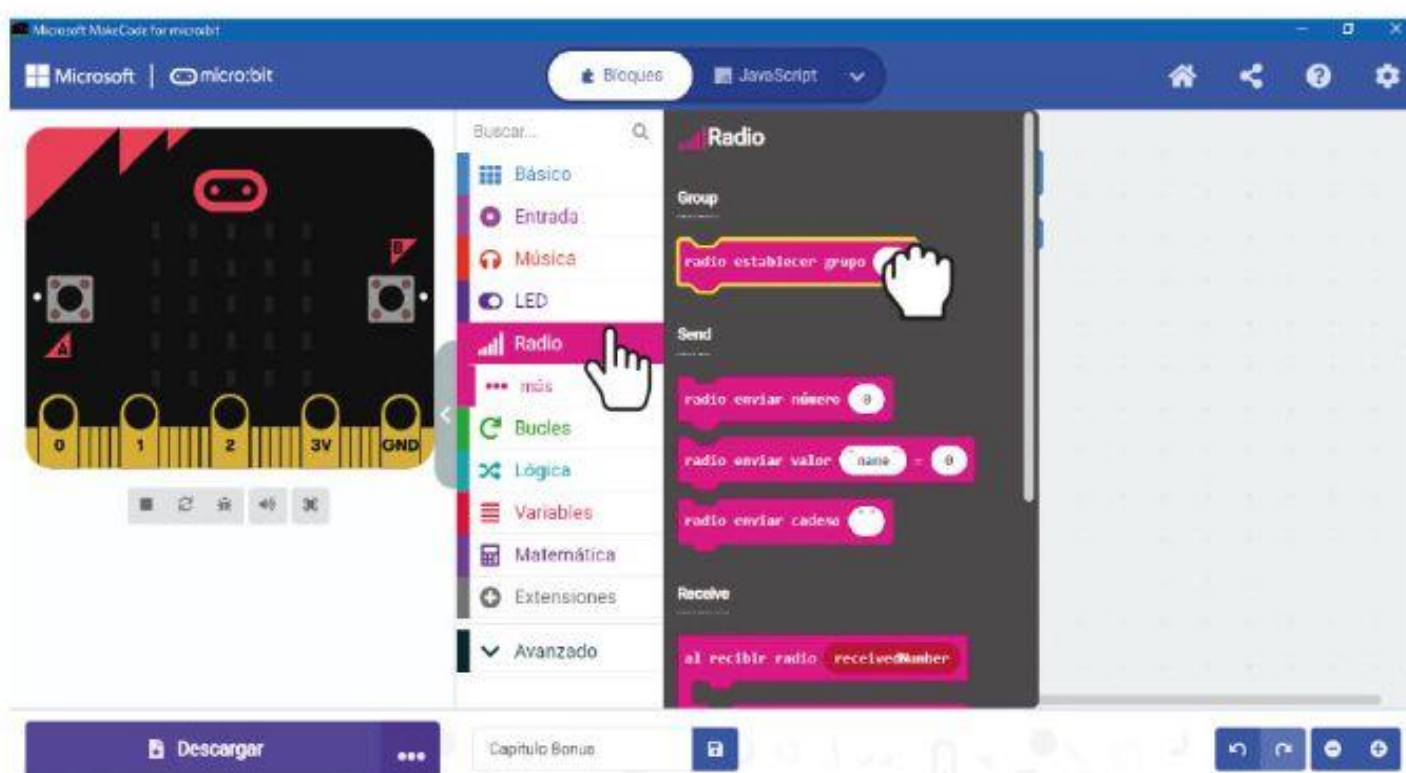




En cualquier forma de comunicación necesitamos al menos dos partes interesadas: un transmisor y un receptor. En este juego necesitamos dos EDU: BIT que se comunican entre sí enviando y recibiendo señales de radiodifusión.

PROGRAMAMOS!

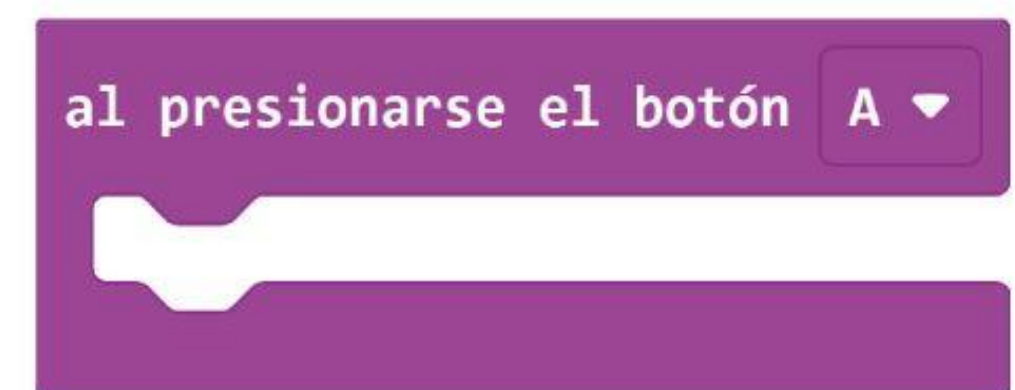
Paso 1 Crea un nuevo proyecto en el editor de MakeCode y agrega la extensión EDU: BIT. Haz clic en la categoría **[Radio]** y selecciona el bloque **[radio establecer grupo _]**. Encaja el bloque dentro del bloque **[al iniciar]**.



Paso 2 En tu editor de MakeCode, haz clic en la categoría **[Básico]** y luego agrega el bloque **[mostrar ícono]** al bloque **[al iniciar]**.



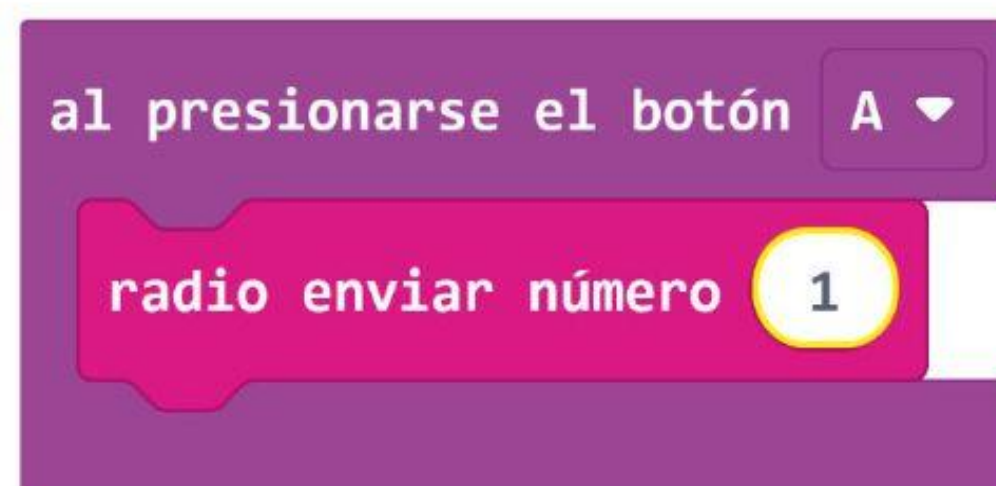
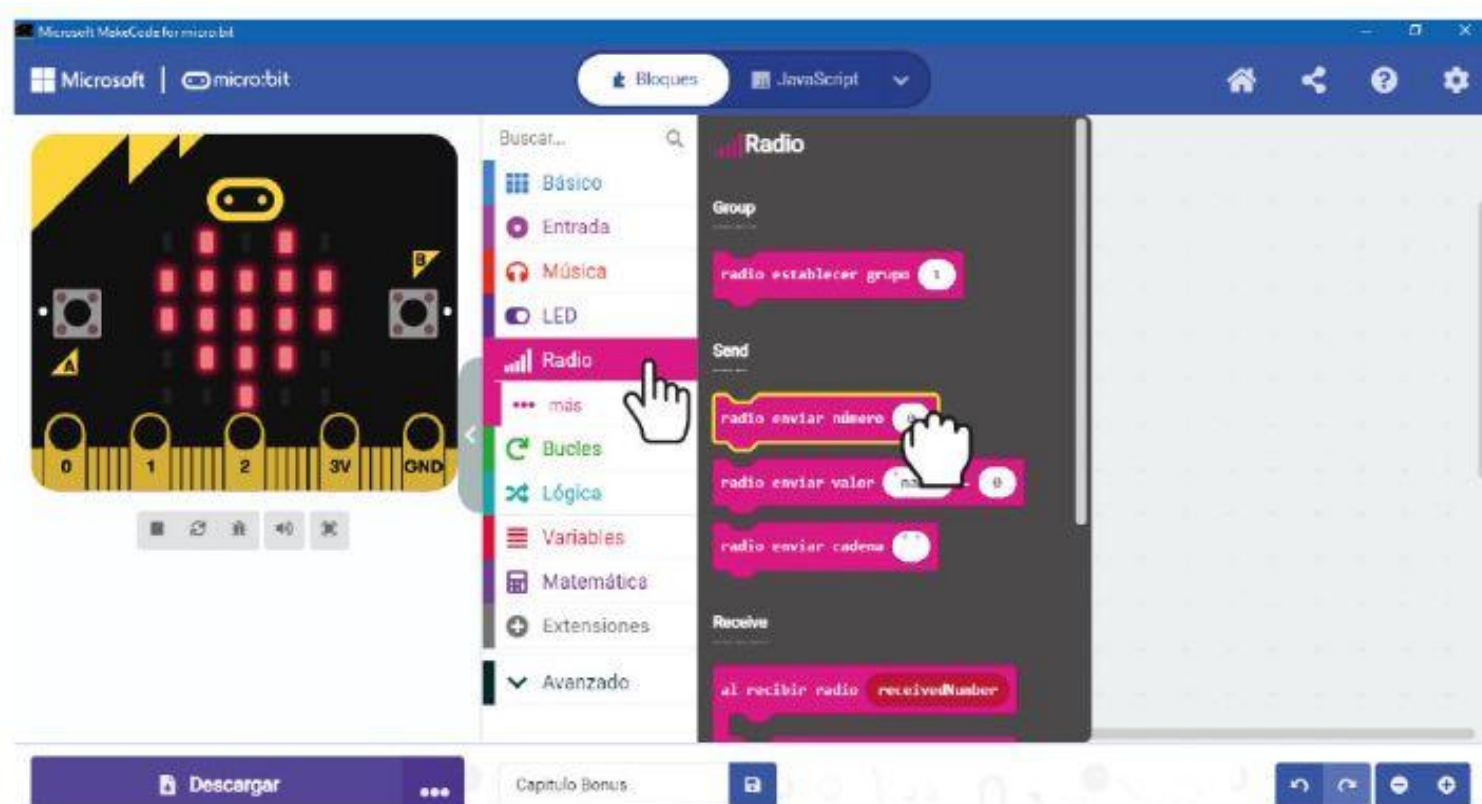
Paso 3 Haz clic en la categoría **[Entrada]** y selecciona el bloque **[al presionarse el botón _]**.



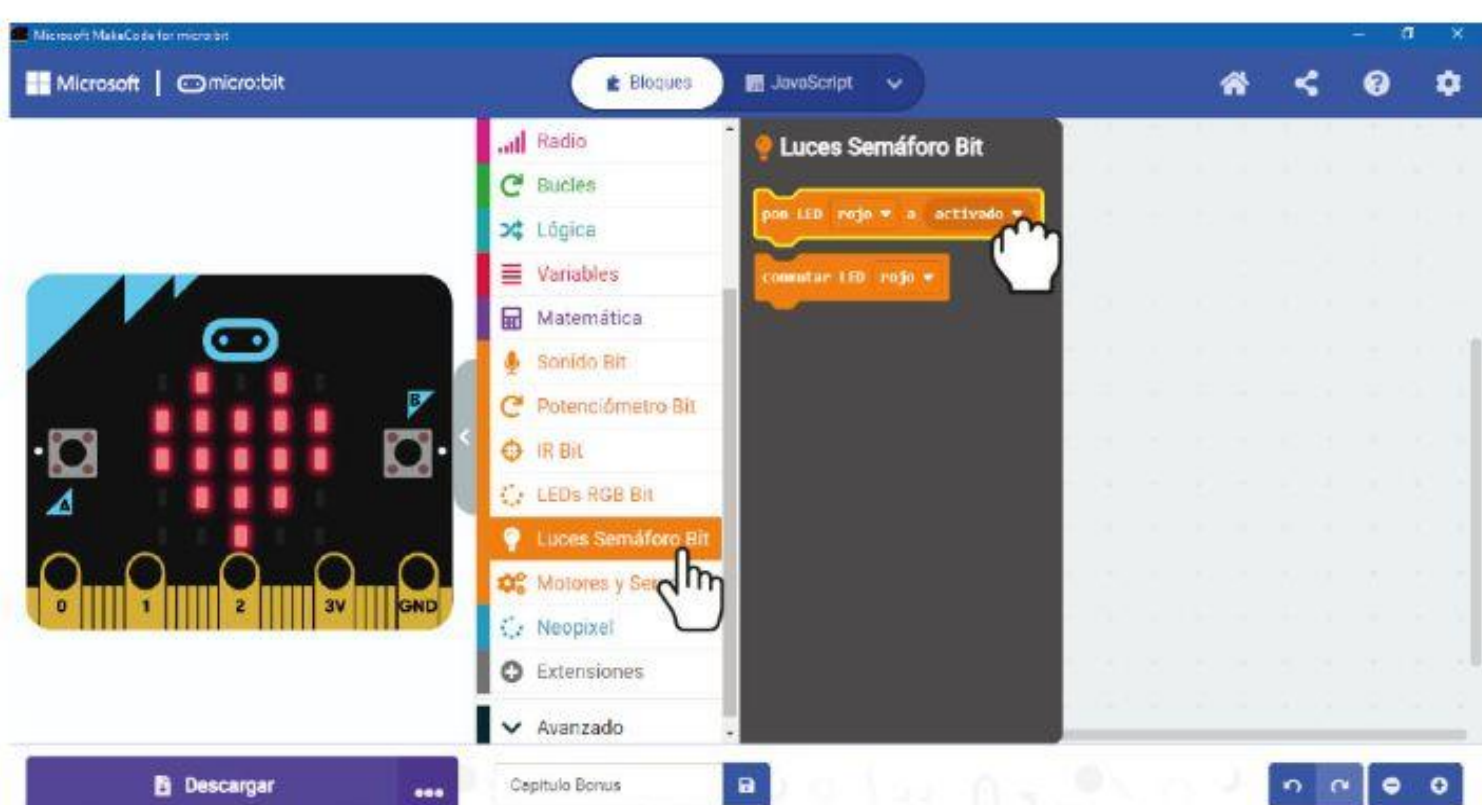
CAPÍTULO BONUS : Juego Simon con LED



Paso 4 Haz clic en la categoría **[Radio]** y selecciona el bloque **[radio - enviar número _]**. Cambia el valor a **1**.



Paso 5 Haz clic en la categoría **[Luces Semáforo Bit]** y selecciona el bloque **[pon LED _ a _]**. Duplícalo y encaja ambos bloques en el bloque. **[al presionarse el botón _]**. Cambia la configuración en el segundo bloque a **'desactivado'**.



Paso 6 Haz clic en la categoría **[Básico]** y selecciona **[pausa (ms) _]**. Encaja el bloque entre los bloques. **[poner LED _ a _]** y cambia el valor a **500**.



CAPÍTULO ADICIONAL: Juego Simon con LED

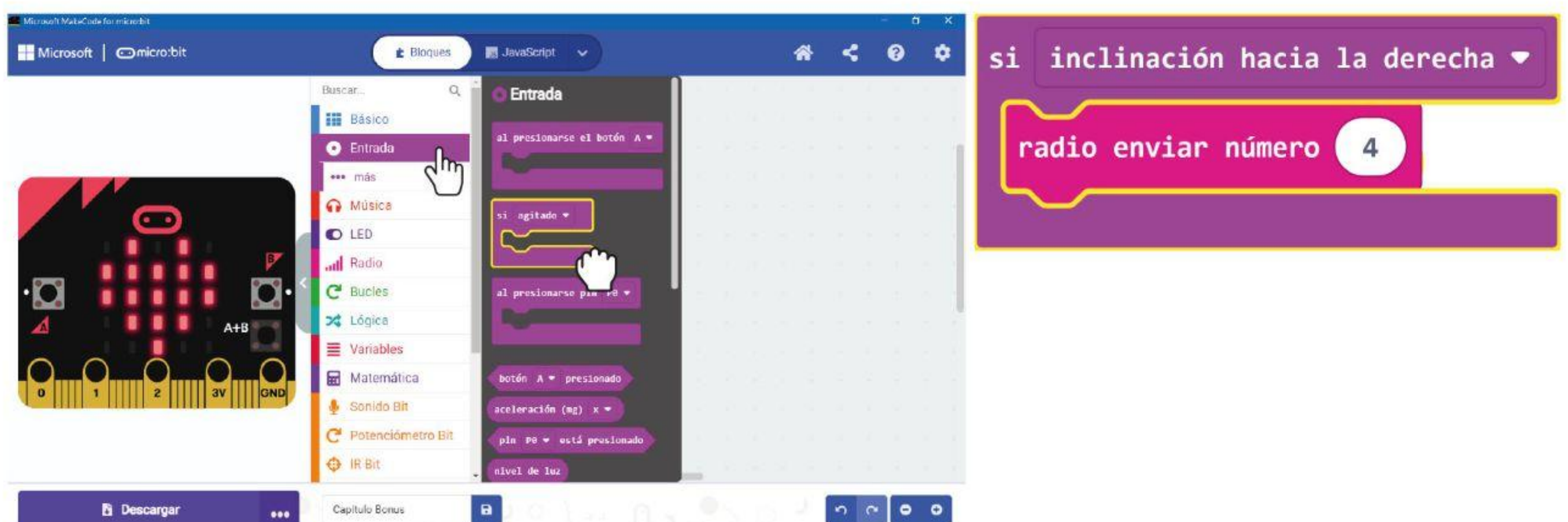
Paso 7 Haz clic derecho en el bloque **[al presionar el botón_]** y luego selecciona 'Duplicar'. Repite para obtener tres grupos de los mismos bloques.



Paso 8 Cambia la configuración del botón, el valor del número y el color del LED para los bloques duplicados de la siguiente manera:



Paso 9 Haz clic en la categoría **[Entradas]** y selecciona el bloque **[cuando _]**. Cambia la configuración a 'inclinarse a la izquierda'. Haz clic en la categoría **[Radio]** y selecciona el bloque **[radio - numero de envío _]**. Cambia el valor a 4.

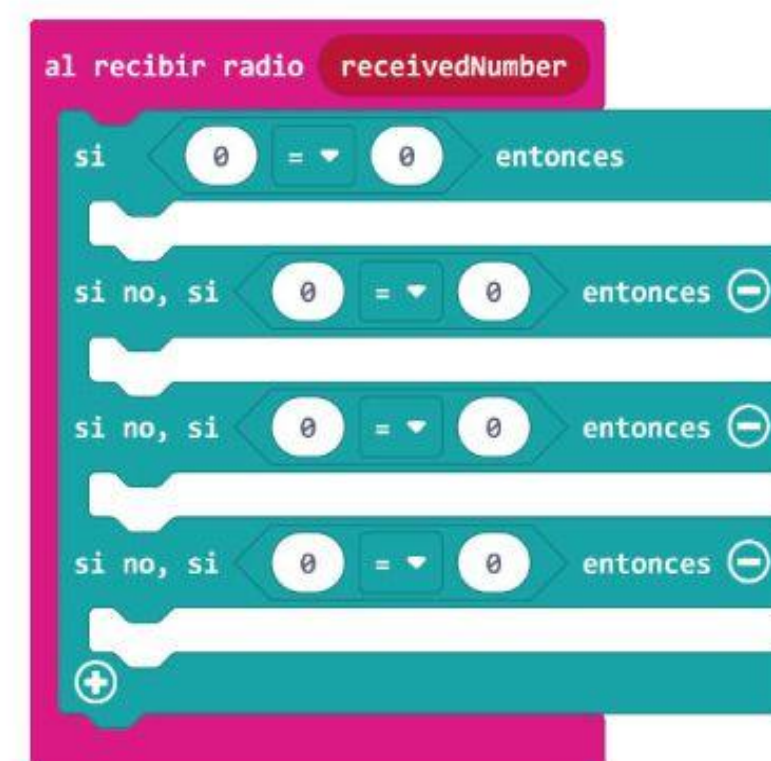
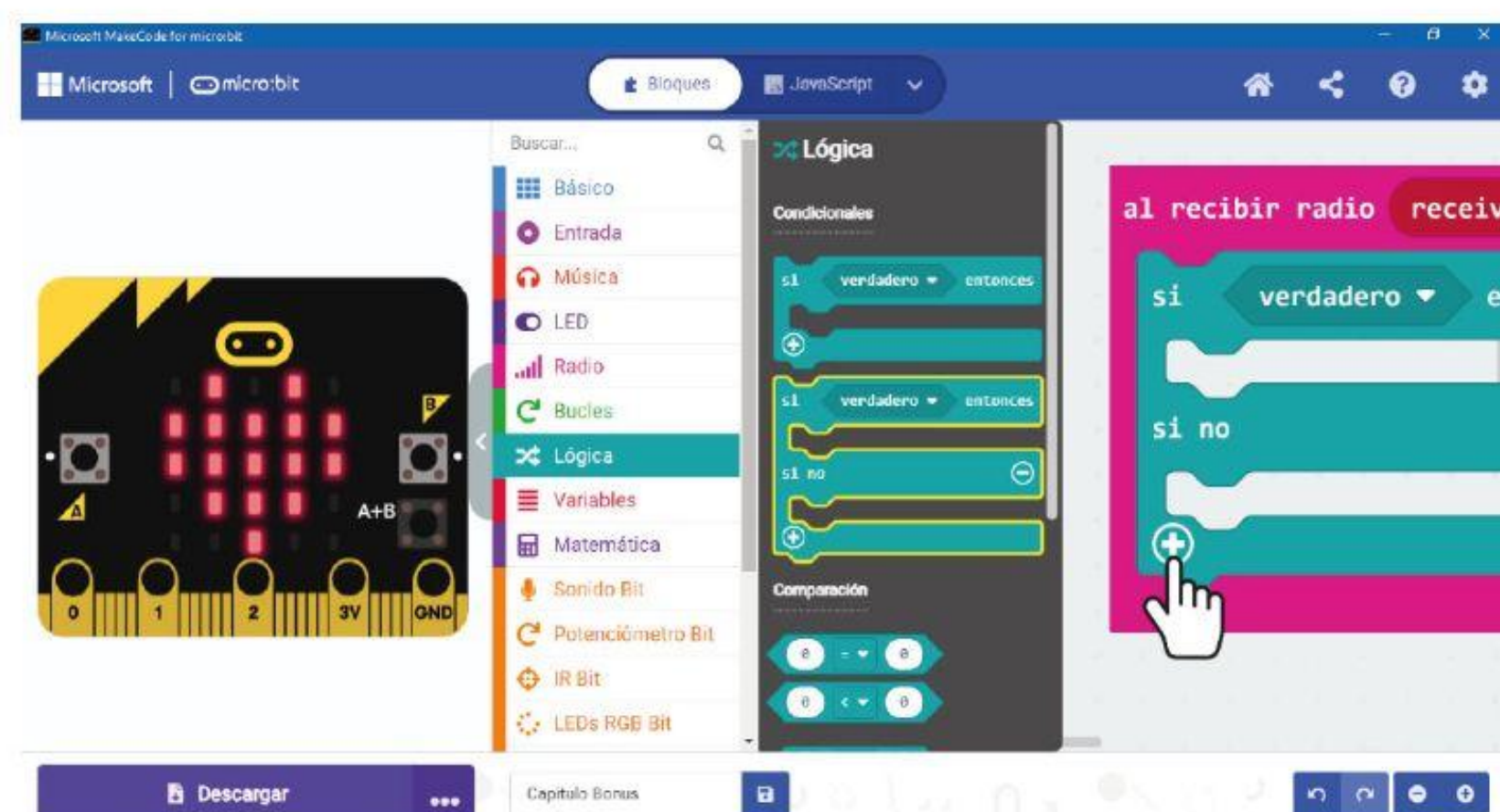




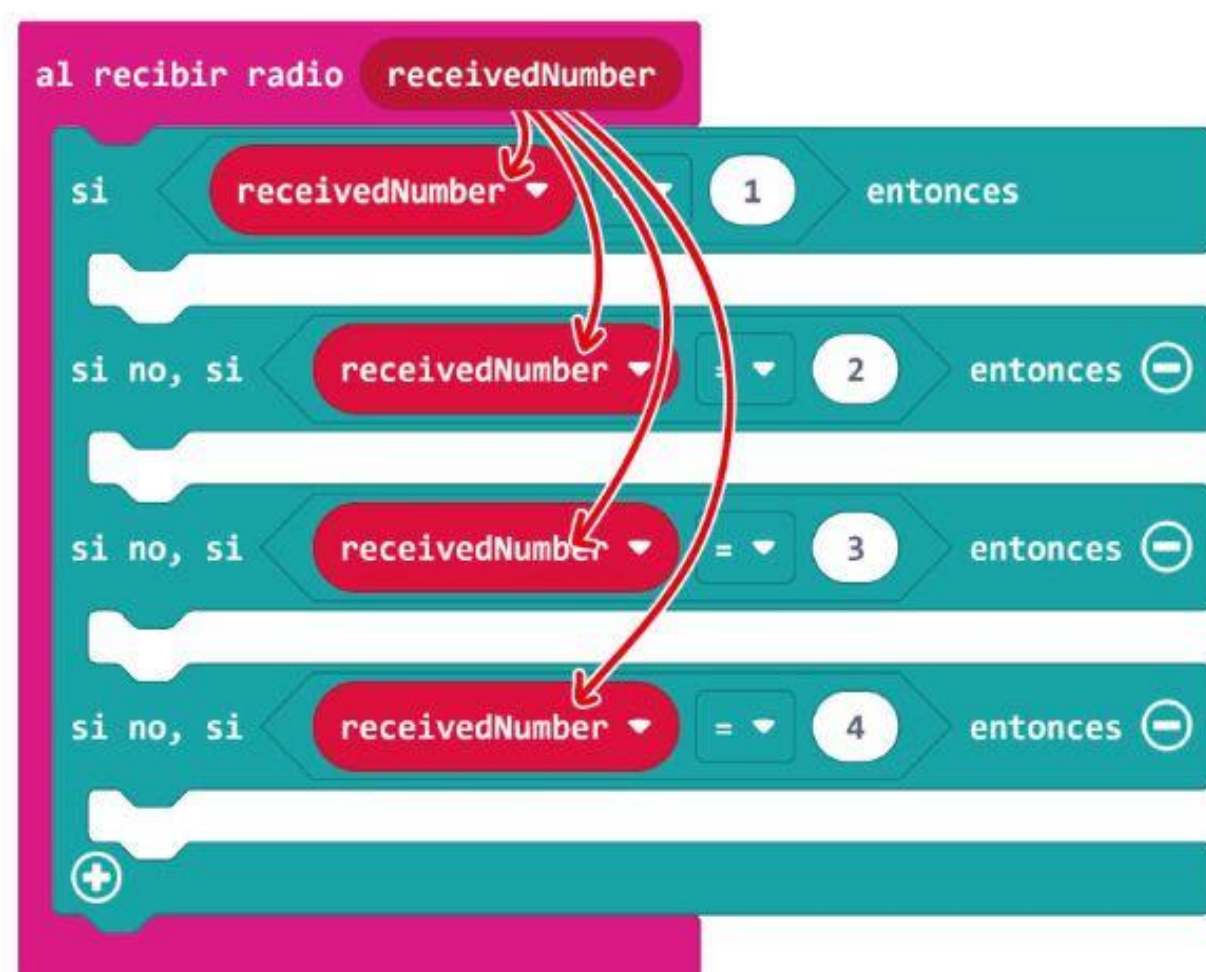
Paso 10 Haz clic en la categoría **[Radio]** y selecciona el bloque **[radio - al recibir señal _]**.



Paso 11 Haz clic en la categoría **[Lógica]** y selecciona el bloque **[si-entonces-de lo contrario]**. Haz clic en el botón + para agregar tres condiciones **[de lo contrario]** y haz clic en el botón - para eliminar la condición **[de lo contrario]**. Adjunta el bloque de comparación **[Lógica] : [_ = _]** en cada una de las ranuras "si" y "de lo contrario si".

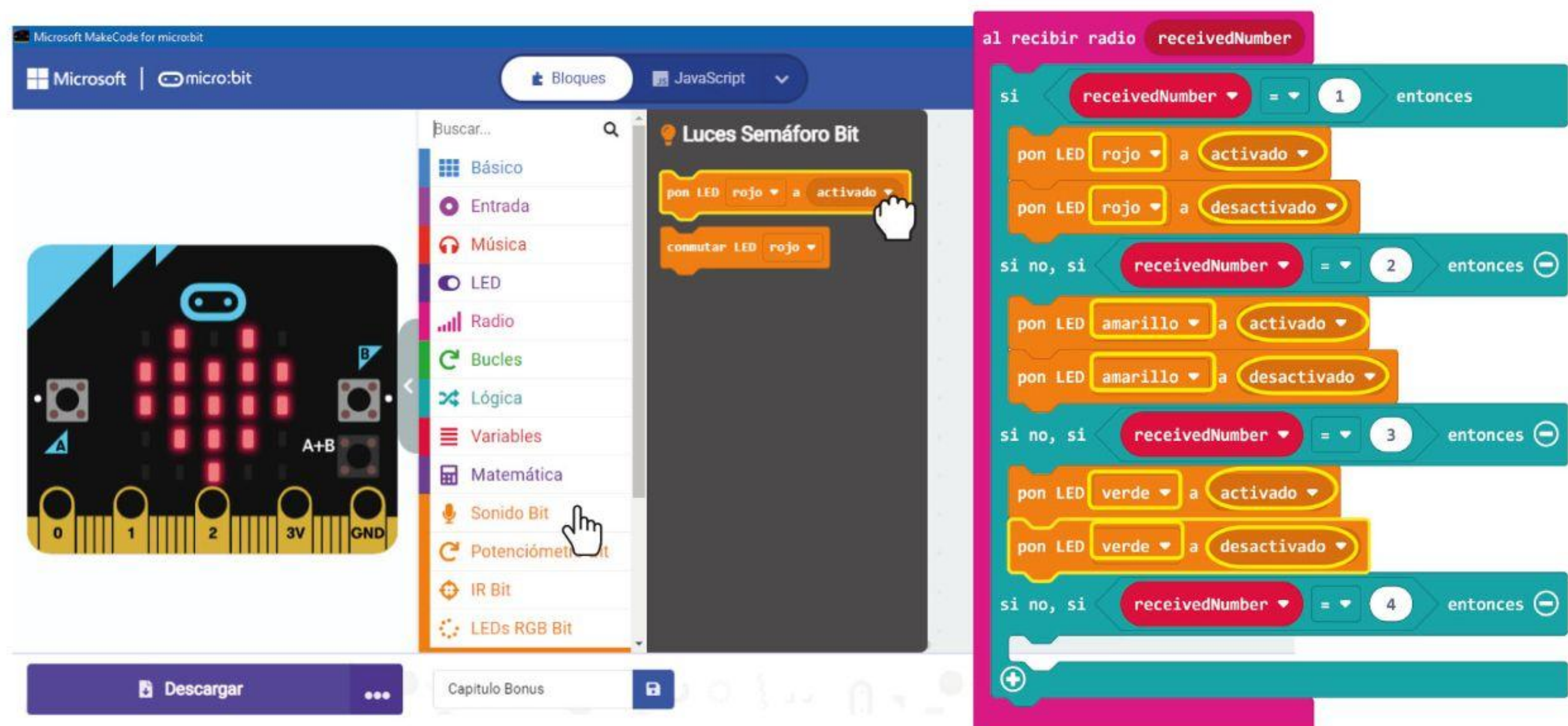


Paso 12 Haz clic y arrastra la variable **'recibidoNumero'** a los bloques de comparación como se muestra y cambia los valores a 1, 2, 3 y 4 respectivamente.

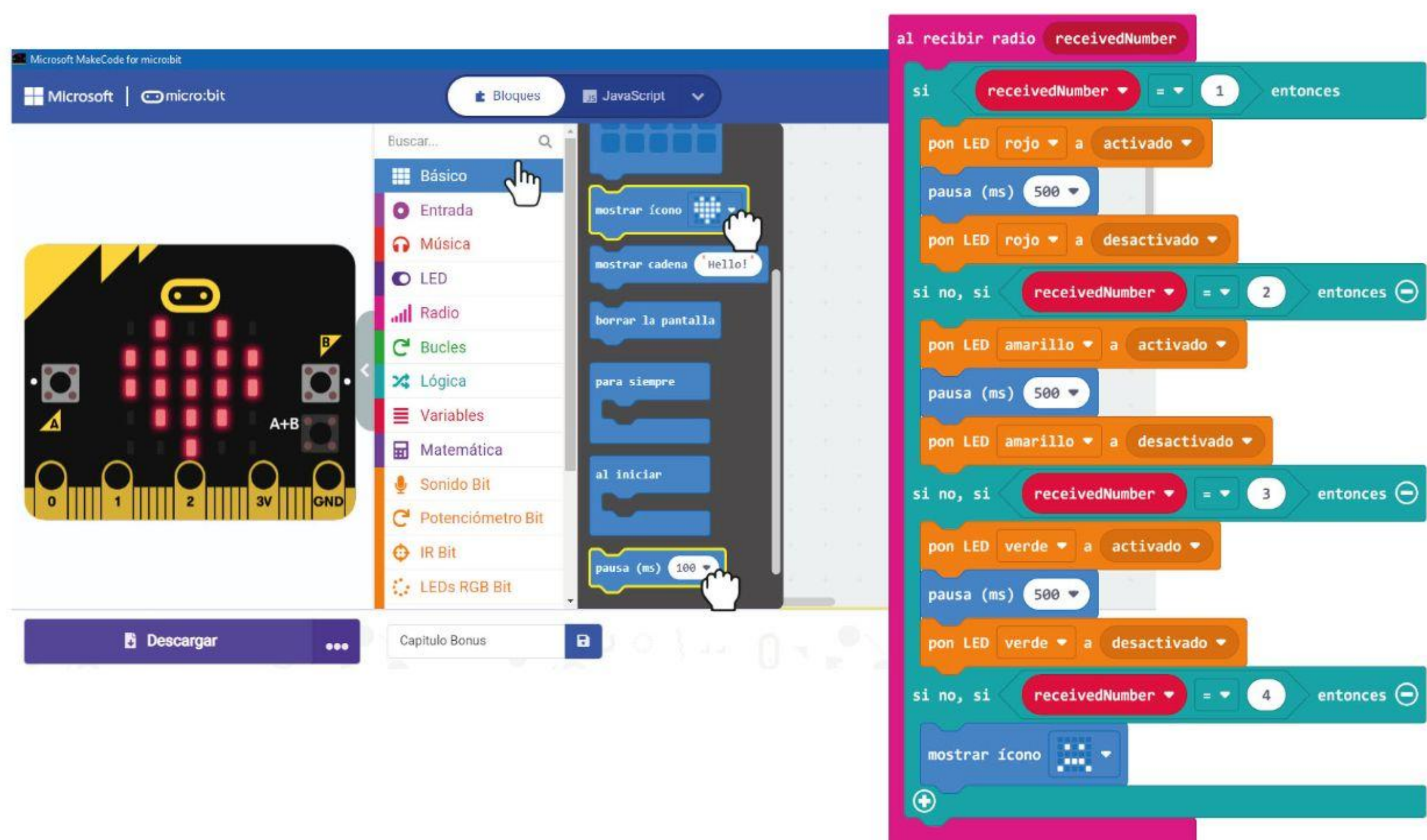


CAPÍTULO BONUS : Juego Simon con LED

Paso 13 Haz clic en la categoría **[Luces semáforo Bit]** y selecciona el bloque **[pon LED _ a _]**. Duplica el bloque y colócalo en las tres primeras ranuras. Cambia el color y la configuración de encendido/apagado como se muestra.



Paso 14 Haz clic en la categoría **[Básicos]** y selecciona **[pausa (ms) _]**. Duplica y ajusta los bloques entre los bloques **[poner LED _ a _]** y cambia el valor a **500**. Agrega el bloque **[Básico] : [mostrar icono]** en la última ranura "de lo contrario si" y cambia el icono a "cara triste".





Paso 15 Haz clic en la categoría **[Musica]** y añade **[comenzar melodía _ ripitiéndola _]** para completar tu código. Cambia la melodía a “wawawawaa” (o cualquier melodía de tu elección).

Aquí está el código completo:

```
al iniciar
  radio establecer grupo 1
  mostrar ícono
```

Para enviar o recibir señales de transmisión, todas las tarjetas EDU: BIT que se comunican entre sí deben configurarse con el mismo grupo de radio. Puede ser un número entre 0 y 255.

```
al presionarse el botón A
  radio enviar número 1
  pon LED rojo a activado
  pausa (ms) 500
  pon LED rojo a desactivado

al presionarse el botón B
  radio enviar número 2
  pon LED amarillo a activado
  pausa (ms) 500
  pon LED amarillo a desactivado

al presionarse el botón A+B
  radio enviar número 3
  pon LED verde a activado
  pausa (ms) 500
  pon LED verde a desactivado

si inclinación hacia la derecha
  radio establecer grupo 4
```

Estos son bloques de eventos. Cuando se activa, EDU: BIT enviará el comando correspondiente (es decir, 1, 2, 3 o 4) a otras tarjetas EDU: BIT vecinas que estén configuradas con el mismo grupo de radio.

Cuando EDU: BIT recibe un comando, verificará el valor y luego ejecutará el bloque de código correspondiente..

```
al recibir radio receivedNumber
  si receivedNumber = 1 entonces
    pon LED rojo a activado
    pausa (ms) 500
    pon LED rojo a desactivado
  si no, si receivedNumber = 2 entonces
    pon LED amarillo a activado
    pausa (ms) 500
    pon LED amarillo a desactivado
  si no, si receivedNumber = 3 entonces
    pon LED verde a activado
    pausa (ms) 500
    pon LED verde a desactivado
  si no, si receivedNumber = 4 entonces
    mostrar ícono
```

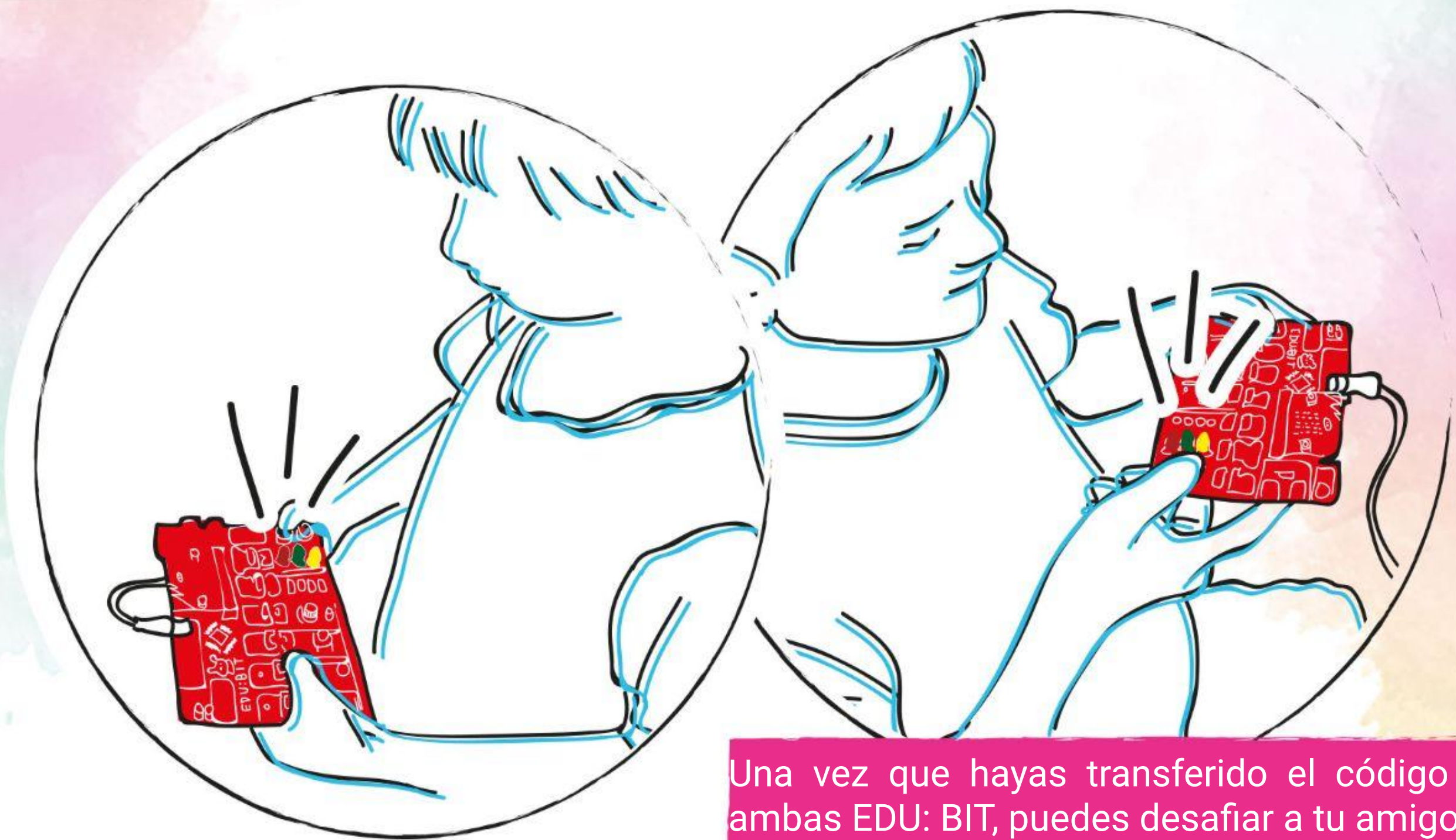
Paso 16 Transfiere el código completo a tu EDU: BIT y también a la de tu amigo.



Cuando ambos EDU: BIT están encendidos, puedes enviar señales de radio para encender los LED en el tablero EDU: BIT de tu amigo; y viceversa, tu amigo también puede pulsar los botones de su EDU:BIT para encender los LEDs de tu placa.

Vamos a jugar

Juego Simon con LED



Una vez que hayas transferido el código a ambas EDU: BIT, puedes desafiar a tu amigo a una versión interactiva del juego Simon Says.

CÓMO JUGAR:

Ambos jugadores se turnan para ser "Simon". Presiona los botones para encender el LED rojo, amarillo y verde cuando sea tu turno.

Para comenzar el juego, el jugador 1 enciende UNO de los LED en el bit de semáforo.

El jugador 2 observa y luego enciende el mismo LED, seguido de otro LED.

El juego continúa con los jugadores turnándose para repetir la última secuencia y luego encender otro LED para agregar a la secuencia..

Si uno de los jugadores enciende el LED incorrecto (o en la secuencia incorrecta), el oponente inclinará su EDU: BIT para finalizar el juego.

.El perdedor tendrá que resetear su EDU:BIT para empezar una nueva partida

Este juego comienza fácil pero se vuelve más largo y más complejo después de cada turno. Para ganar necesitarás observar la secuencia cuidadosamente. Es un juego divertido para entrenar tu memoria.





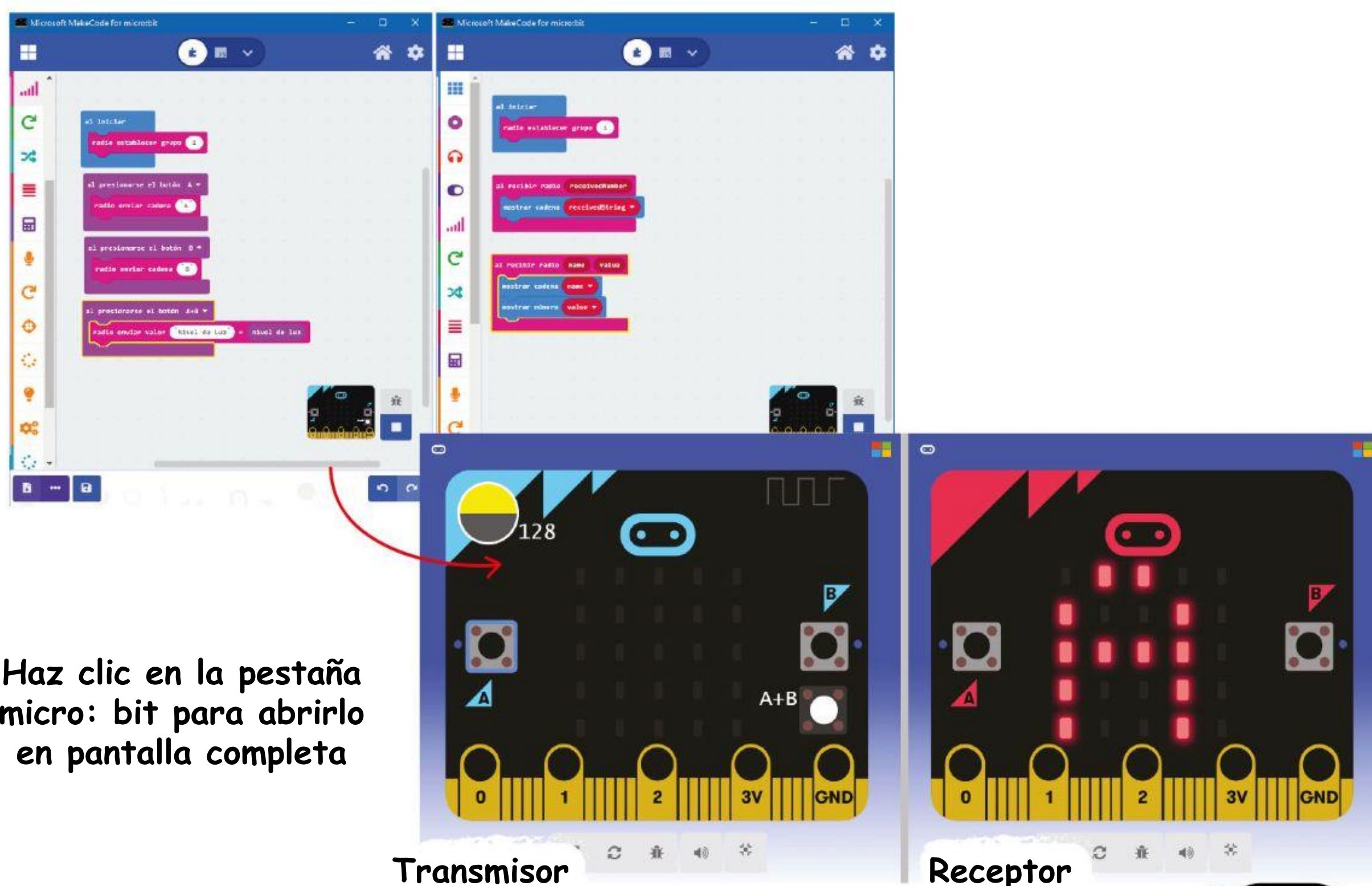
EXPLORA OTROS BLOQUES

#1 Además de enviar números, también puedes enviar mensajes de texto usando el bloque **[radio - enviar cadena _]**. Necesitarás el bloque **[radio - al recibir radio _]** para recibir una señal de cadena de texto transmitida. La longitud máxima de la cadena es de 19 caracteres.

#2 Usa el bloque **[radio - enviar valor "_" = _]** y el bloque **[radio - al recibir "_" "_"]** si necesitas enviar y/o recibir texto o números juntos. La longitud máxima de la cadena es de 8 caracteres.



Si no tienes varias EDU: BIT, aún puedes probar la función de comunicación por radio. Ve a makecode.com/multi para escribir su código para el "transmisor" y el "receptor". Puede ver el resultado en las ventanas del simulador.



Haz clic en la pestaña micro: bit para abrirlo en pantalla completa

Transmisor

Receptor

Cuando presionas el botón A en el micro: bit del "transmisor", el otro micro: bit recibirá la señal de radio y mostrará la cadena recibida, es decir, 'A'. ¿Qué pasa si presionas los botones A + B?

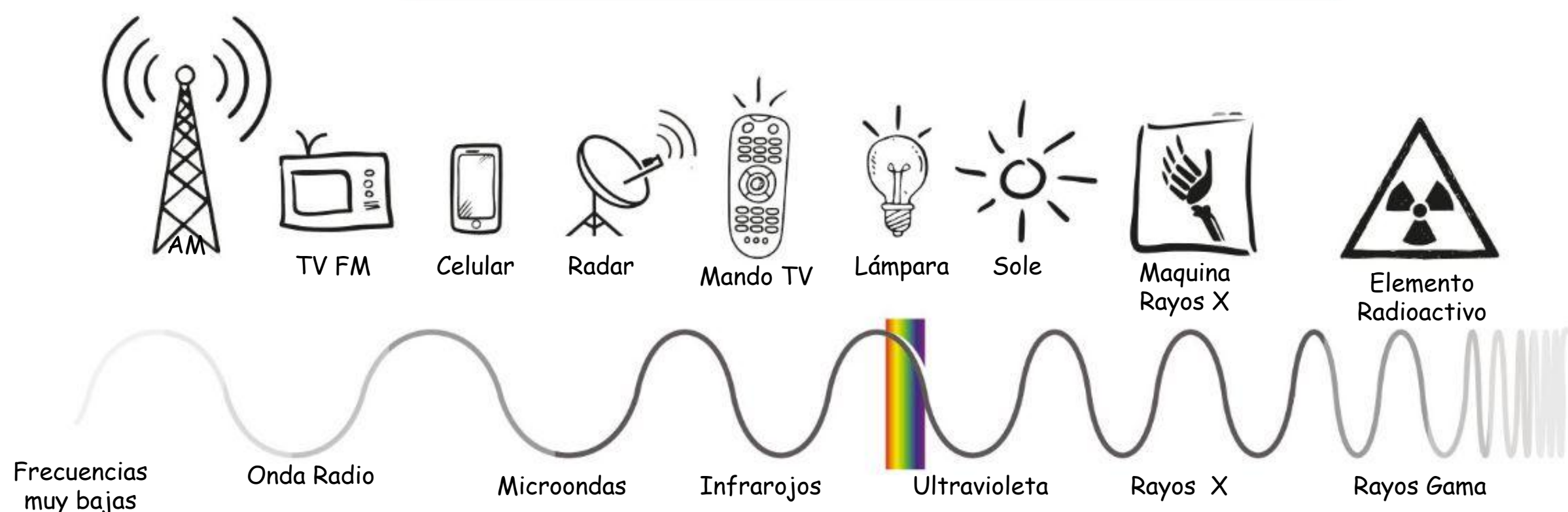


¡COSAS DIVERTIDAS!



La antena transmite señales en forma de ondas electromagnéticas, que también se utilizan habitualmente para las transmisiones de radio y televisión, así como las vía satélite.

Espectro electromagnético



NOTA!



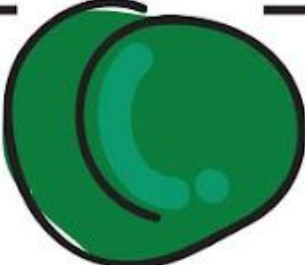
Para que su EDU: BIT envíe y reciba señales de transmisión de los otros EDU: BIT, deberás configurarlos todos con el mismo grupo de radio.

DESAFÍO DE LA APLICACIÓN

Configura una red de comunicación de respuesta para tu clase

Cómo funciona?

Configura cada EDU: BIT en la clase con el mismo grupo de radio. El EDU:Bit del profesor está configurado para desplazar el texto cuando recibe una señal de "cadena" y enciende los LED en el bit de semáforo cuando recibe una señal de "número", por lo que...

Número Recibido	LED Acceso	¿Que significa?
1	ROJO 	A / No / Falso
2	AMARILLO 	B / Tal vez / no estoy seguro
3	VERDE 	C / Si / Verdadero

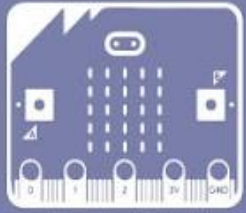
EDU:BIT del estudiante está configurado para enviar una cadena con el nombre del estudiante y luego enviar un número (1 o 2 o 3) para encender los LED en la EDU del maestro: BIT cuando está activado.

- Presione el botón A para enviar el número 1.
- Presione el botón B para enviar el número 2.
- Presione los botones A + B para enviar el número 3.

Aquí hay un consejo para ti. Asigna nombres cortos (con solo dos o tres caracteres) a cada estudiante para reducir el tiempo de desplazamiento del texto.



Aprendí a ...



- ☐ Programar EDU: BIT para mostrar texto y animaciones en la matriz LED.
- ☐ Descargar, guardar, publicar y editar archivos MakeCode .hex.



- ☐ Utilizar bloques de entrada para la programación basada en eventos.
- ☐ Crear y utilizar variables.



- ☐ Utilizar el zumbador piezoeléctrico del Music Bit para reproducir melodías sencillas.
- ☐ Crear y utilizar funciones.
- ☐ Leer partituras.



- ☐ Programar EDU: BIT para controlar los LED en el bit de semáforo.
- ☐ Agregar extensiones al Editor de MakeCode.



- ☐ Programar EDU: BIT para detectar objetos usando el Bit Infrarrojo.
- ☐ Usar bucles while.
- ☐ Crear y utilizar matrices.



- ☐ Programar EDU: BIT para leer valores analógicos del Potenciómetro Bit.
- ☐ Mapear las lecturas de las entradas analógicas.
- ☐ Utilizar bloques lógicos para la programación condicional.



- ☐ Programar EDU: BIT para detectar sonidos con el Sound Bit.
- ☐ Usar eventos para cambiar entre diferentes modos.



- ☐ Programar EDU: BIT para controlar un motor DC - dirección de rotación y control de velocidad.
- ☐ Usar bloques matemáticos para realizar operaciones aritméticas.



- ☐ Programar EDU: BIT para controlar un servomotor - posición angular.



- ☐ Programar EDU: BIT para encender LED RGB en el Bit RGB con diferentes colores/patrones.



- ☐ Programar EDU: BIT para enviar y recibir señales de radio

** Marca las casillas si dominas la habilidad; de lo contrario, vuelve al capítulo correspondiente para revisar.

NOTA DE EDUTEAM @ CYTRON

¡¡¡FELICIDADES!!!

Has completado todos los capítulos y aprendido a programar con el Editor MakeCode. Esperamos que hayas disfrutado jugando nuestra selección de juegos populares para niños. Te felicitamos por desarrollar algunas aplicaciones útiles para tu clase.

A estas alturas ya deberías tener una buena comprensión de lo que se puede hacer con micro: bit y todos los elementos adicionales que se incluyen en su placa EDU: BIT. Psss... ¿sabes que puedes despegar cada Elemento Bit del tablero?

Adelante, despégalos si quieres. Una vez que se hayan separado del tablero, puedes construir nuevos proyectos con los diversos Elementos Bits; deberás conectarlos utilizando los cables plug-and-play suministrados.

Y ahora es el momento de ponerse el sombrero del pensador y crear nuevos juegos y aplicaciones. No podemos esperar a ver qué increíbles proyectos puedes hacer.

Recuerda compartir tus creaciones con nosotros. Escribenos un correo electrónico o déjanos un mensaje en nuestra página de Facebook. Nos encantaría saber de tí.

Saludos,
Adam & Anna



¡Actualízanos
sobre tu
progreso!



[link.cytron.io/
edubit-resource-hub](https://link.cytron.io/edubit-resource-hub)



