	<b>INSTITUCIÓN EDUCATIVA JOSÉ EUSEBIO CARO</b> <b>TECNOLOGÍA E INFORMÁTICA</b>
<b>Docente</b>	Jesús Eduardo Madroñero Ruales
<b>Propósito del taller</b>	Comprender los aspectos básicos de la programación en MakeCode. Recordar los conceptos de los aspectos y elementos básicos de la plataforma MakeCode y la tarjeta Micro:bit. Aplicar los conceptos básicos de programación en MakeCode.
<b>Competencias</b>	Identificación y solución de problemas a través de procesos tecnológicos. Los estudiantes contribuyen de manera constructiva a equipos de trabajo, asumiendo varios roles y responsabilidades para trabajar efectivamente hacia un objetivo común.

## INTRODUCCIÓN A LA PROGRAMACION: PLATAFORMA MAKECODE Y MICRO:BIT

### ¿QUÉ ES MAKECODE?

Microsoft MakeCode [1], es un editor en línea basado en la web que le permite crear programas utilizando bloques con funciones específicas. También se conoce como lenguaje de programación gráfico y es compatible con todos los navegadores y plataformas web modernos.

El sitio web de MakeCode utiliza cookies para análisis, contenido personalizado y anuncios. No necesita una cuenta de usuario para crear y guardar proyectos con MakeCode. Todos los proyectos se guardan en el caché local del navegador.

### INGRESANDO A MAKECODE

El enlace de ingreso es el siguiente: <https://makecode.microbit.org/>. La plataforma permite programar y simular el funcionamiento de la tarjeta Micro:bit.



**Figura 1 - Ingresando a MakeCode**

En la interfaz que se observa en la figura 1, se detallan diferentes botones: Para crear nuevos proyectos, importar programas desarrollados, revisión de tutoriales, ajustes generales y retorno a la interfaz inicial.

### ¿QUÉ CONTIENE MAKECODE?

- **Simulador:** Proporciona una salida sin el hardware real mientras que está construyendo el código. Los siguientes botones pueden usarse para controlar el comportamiento del simulador.
  - o **Iniciar/detener el simulador:** detiene el programa y se reinicia desde el principio.

- **Reiniciar el simulador:** reinicia el programa. (salida) desde el principio.
- **Cámara lenta:** muestra la salida en cámara lenta.
- **Silenciar audio:** silencia el audio cuando está trabajando con música y voz.
- **Iniciar en pantalla completa:** muestra el simulador en modo de pantalla completa.
- **Cuadro de herramientas:** proporciona bloques en categorías. También le permite buscar o agregar más extensiones al cuadro de herramientas, si están disponibles.
- **Área de codificación:** área donde se compilará su código a bloques o el código que escriba en JavaScript.

## CONTROLES DEL EDITOR

En la siguiente figura se detallan las áreas y los controles básicos del editor MakeCode:

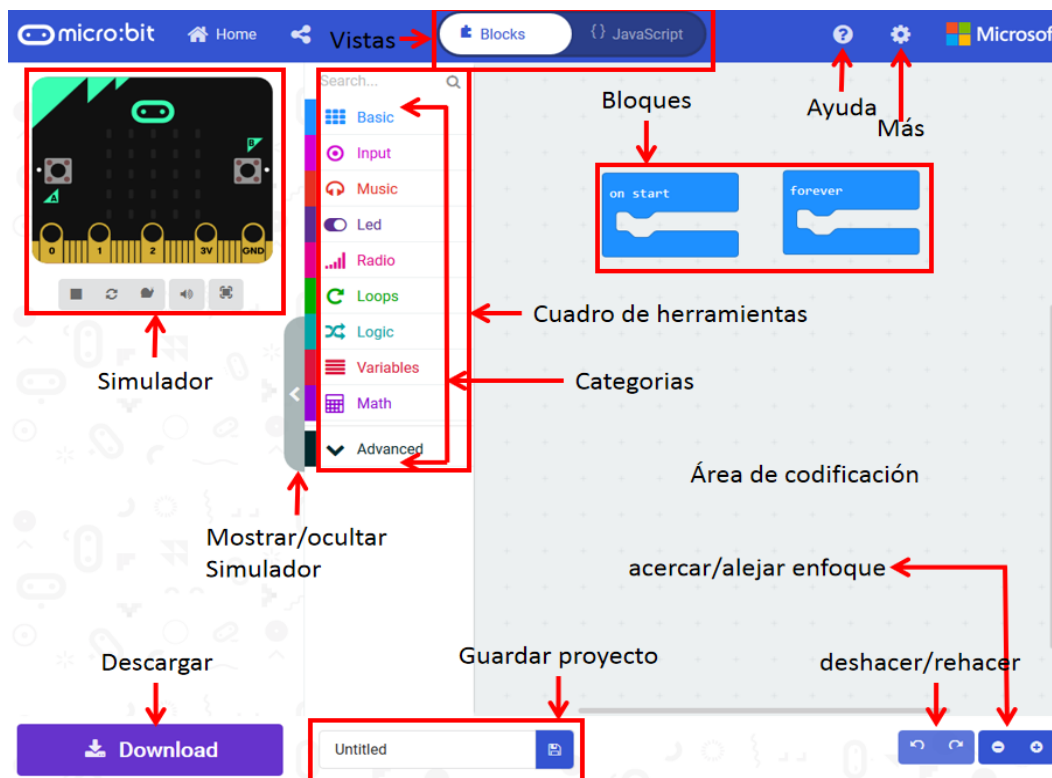


Figura 2 - Áreas y controles básicos de MakeCode

A continuación, se describen cada una de las áreas y controles básicos de la interfaz:

- **Inicio:** Lo lleva a la pantalla de inicio (<https://makecode.microbit.org/>), que muestra proyectos recientes y otras actividades.
- **Compartir:** Muestra la ventana Compartir proyecto que le permite publicar su proyecto en la nube pública e incrustar su proyecto en una página web con diferentes opciones.
- **Bloques o JavaScript:** Le permite cambiar la vista del código de Bloques a JavaScript o viceversa. Presione uno de los botones de vista en la parte superior central de la ventana.
- **Ayuda:** Muestra un menú con opciones de ayuda como soporte, referencia, bloques, JavaScript y hardware.
- **Más... (icono del engranaje):** Le permite acceder a la configuración del proyecto, agregar extensiones, eliminar el proyecto actual, eliminar todos los proyectos, elegir un idioma y emparejar micro: bit vía bluetooth con un solo clic.
- **Deshacer y rehacer:** Le permite deshacer y rehacer los cambios recientes que realice en Bloques o JavaScript con los botones Deshacer y Rehacer en la parte inferior derecha de la ventana del editor.
- **Acercar y Alejar:** Los botones de enfoque cambian el tamaño de los bloques cuando está trabajando en la vista Bloques. Cuando trabajas con el código en la vista de JavaScript, los botones de enfoque cambian el tamaño del texto.
- **Guardar proyecto:** Puede escribir un nombre para su proyecto y guardarlo. Escriba un nombre para el proyecto en el cuadro de texto y presione el icono del disco para guardar.
- **Descargar:** El botón Download copiará su programa en una unidad de su computadora.
- **Mostrar/Ocultar el simulador:** El botón se puede usar para mostrar u ocultar el simulador con un clic.

## ¿QUÉ ES LA MICRO:BIT?

La micro:Bit es una pequeña tarjeta programable, con un costo asequible a cualquier bolsillo. Aun cuando su tamaño es muy reducido, incorpora gran cantidad de sensores y actuadores lo que unido a que usa un software Open Source, hacen de la micro:BIT una plataforma ideal para introducirse en el mundo de la programación básica.

## CARACTERÍSTICAS DE LA MICRO:BIT

La micro:BIT incorpora:

- **25 LEDs:** Se pueden programar de forma independiente y permiten mostrar números, letras e imágenes. Si el texto o la cifra no caben en el display se desplazan de forma automática.
- **Sensor de Luz:** Los LEDs también tiene la posibilidad de ser usados como sensor de luz ambiente.
- **Pulsadores:** Existen 2 botones, etiquetados como A y B. Se puede detectar la pulsación independiente de cada uno de ellos, así como la pulsación simultánea de ambos.
- **Conectores:** Situados en la parte inferior de la placa, dispone de 25 conexiones que permiten conectar otros sensores y actuadores. 5 de las conexiones (0,1,2 3v. y GND) se encuentran sobredimensionadas, para facilitar la conexión mediante pinzas.
- **Sensor de temperatura:** Permite conocer a la micro:BIT la temperatura ambiente. Las unidades son los grados Celsius.
- **Acelerómetro:** Activada cuando se mueve la placa, permite conocer aceleraciones y giros a los que se somete la placa.
- **Brújula digital:** Permite conocer la desviación respecto el Norte Magnético. También permite detectar la presencia de campos magnéticos próximos. Al iniciar su uso entra en modo de calibración.
- **Radio:** Permite conectarse inalámbricamente con otras micro:BITs.
- **Bluetooth:** Ideal para conectarse e intercambiar datos inalámbricamente con otros dispositivos (móviles, tablets, ordenadores, etc) que dispongan de este tipo de conexión.
- **USB:** Usado para descargar los programas a la memoria de la tarjeta y para alimentar eléctricamente la micro:BIT.
- **Conector de batería:** Permite suministrar electricidad mediante dos pilas AAA o una batería. La tarjeta carece de interruptor, por lo que cuando se conecta la fuente de alimentación se ejecuta de forma automática el código que haya en memoria.

## EJEMPLOS DE PROGRAMACION POR BLOQUES

**Programa 1 “Hola Mundo”:** Se requiere desarrollar un algoritmo que muestre en la matriz de LEDs de la tarjeta Micro:BIT, el siguiente mensaje: “Hola Mundo!”.

**Solución:** En la interfaz de MakeCode se agrega el siguiente código (algoritmo) por bloques. Los resultados se detallan a continuación:

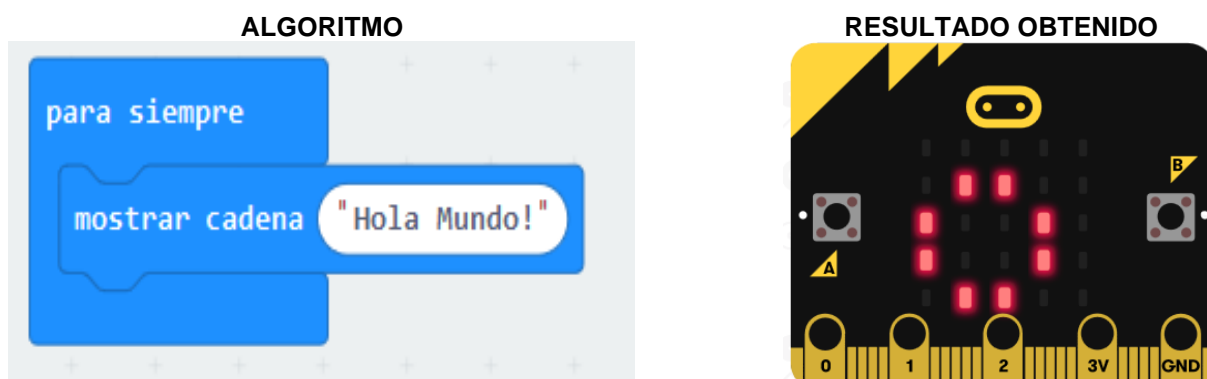


Figura 3 - Algoritmo desarrollado y resultado obtenido para el ejemplo 1

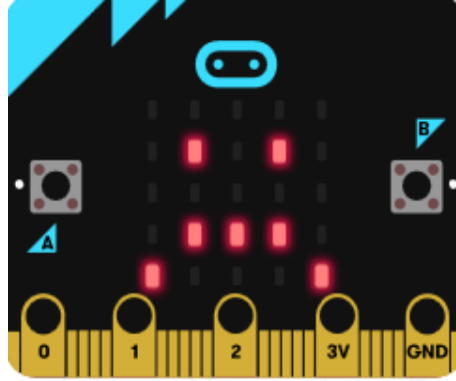
**Programa 2 Utilizando botones:** Se requiere desarrollar un algoritmo que muestre en la matriz de LEDs, un ícono de cara alegre si se está presionado el botón A de la tarjeta Micro:BIT. Si se presiona el botón B, debe mostrar un ícono de cara triste.

**Solución:** Luego de desarrollar el algoritmo, uno de los resultados obtenidos es el siguiente:

### ALGORITMO



### RESULTADO A



### RESULTADO B

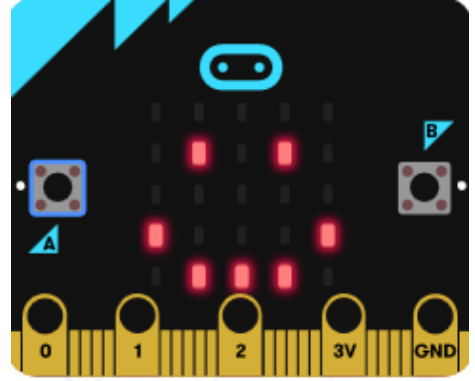


Figura 4 - Algoritmo desarrollo y resultados obtenidos para el ejemplo 2

**Programa 3 Uso de sensores:** Se requiere desarrollar un algoritmo que permita medir la temperatura y el nivel de luz con la tarjeta y presentar el valor medido en la matriz de LEDs. Además, se requiere utilizar el sensor de movimiento para borrar la información medida.

**Solución:** Se hace uso del sensor de temperatura que trae la tarjeta y se presentan los valores medidos al presionar A. Se hace uso del botón B, para presentar los valores medidos de nivel de luz. Para borrar la información medida se hace uso del acelerómetro de la tarjeta. En la siguiente figura se detalla el algoritmo desarrollado:

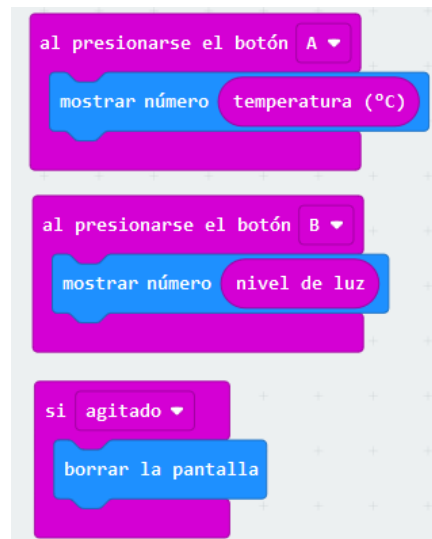
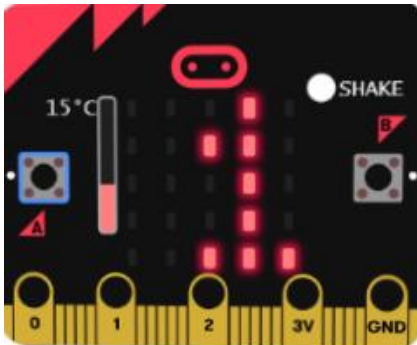


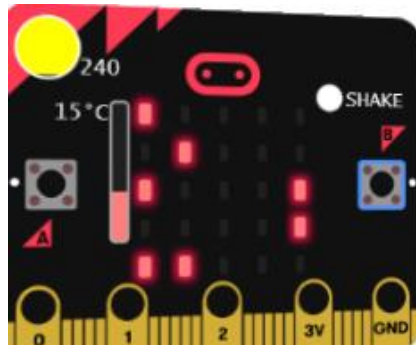
Figura 5 - Algoritmo del ejemplo 3

La interfaz de MakeCode, permite simular variaciones de temperatura y nivel de luz, como se detalla en los resultados obtenidos:

### TEMPERATURA



### NIVEL DE LUZ



### ACELEROMETRO

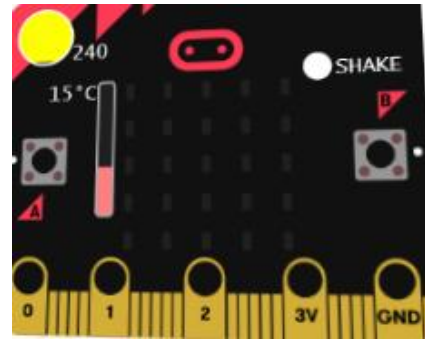


Figura 6 - Resultados obtenidos para el ejemplo 3

## RECURSOS COMPLEMENTARIOS

[1] Curso de robótica (2020, 15 de abril). Qué es Micro:bit? [video]. <https://youtu.be/q1sFUJOSkyc>

[2] BrycoGeek (2019, 23 de noviembre). TUTORIAL: Micro:bit en Español | Ejemplos de Programación Sencillos [video]. [https://youtu.be/MY4jDrN9\\_cc](https://youtu.be/MY4jDrN9_cc)

## CYBERGRAFIA

[1] <http://microes.org/>

[2] <https://agelectronica.blog/2019/07/25/programacion-a-bloques-con-microbit/>

[3] <https://blog.330ohms.com/2019/08/13/tutorial-microbit-0-el-editor-de-bloques/>

### ACTIVIDAD CONCEPTUAL

1. ¿Qué es MakeCode?
2. Describir los aspectos básicos de MakeCode.
3. ¿Qué es la tarjeta Micro:bit?
4. Describir las características básicas de la tarjeta Micro:bit.
5. Consultar en internet acerca de los **sistemas embebidos** y responder las siguientes preguntas:
  - a. ¿Qué son los sistemas embebidos?
  - b. ¿Cuál es su importancia en la actualidad?
6. Consultar en internet que son los sensores, microprocesadores y actuadores, y responder las siguientes preguntas:
  - a. ¿Qué son los sensores, microprocesadores y actuadores?
  - b. ¿En qué se diferencian los sensores de los actuadores?
  - c. ¿Cuál es la importancia de estos tres dispositivos en la actualidad?
  - d. ¿Es posible integrar sensores, microprocesadores y actuadores a la tarjeta Micro:BIT?
7. ¿De qué elementos electrónicos se compone la tarjeta Micro:BIT? Describir cada uno de estos.
8. Describir tres posibles utilidades de la tarjeta Micro:BIT en la vida cotidiana.

### ACTIVIDAD DE CODIFICACIÓN

1. Diseñar un algoritmo que presente en la matriz de LEDs, el mensaje: "Hola" seguido del nombre del estudiante que diseña el algoritmo.
2. Diseñar un algoritmo que al presionar el pulsador A, presente en la matriz de LEDs, el nombre del estudiante. Al presionar el pulsador B, debe presentar la edad del estudiante.
3. Diseñar un algoritmo en MakeCode, que utilice como entrada, los pulsadores y muestre en la matriz de LEDs la siguiente información:
  - a. Un ícono de carita feliz, al presionar el pulsador A.
  - b. Un ícono de carita triste, al presionar el pulsador B.
  - c. Un ícono personalizado al presionar la combinación A+B.

**Nota:** Para todos los algoritmos desarrollados se debe compartir el enlace del proyecto, mediante el formulario en intec-jec.