

# M2Clase 4 - Pensamiento Computacional y Electrónica



## ¡Bienvenidas y bienvenidos a nuestra clase 4!

La semana pasada profundizamos aspectos acerca de la programación y su relación con el Pensamiento Computacional. Esta semana buscaremos descifrar qué hay más allá de la programación; ¿Cuál es su relación con la robótica? , para así seguir encontrando puentes entre la Educación Digital en la Cultura Tecnológica.

### MÁS ALLÁ DE LA PROGRAMACIÓN

---



**Recapitulando**



**Abordaje de la programación... propuestas para el aula**



**Más allá de la programación... Simuladores para experimentar y aprender**



**Actividad 1- Propuestas para compartir**

### ANALIZANDO SIMULADORES

---



**Simulador Vr Vex**



**Simulador Tinkercad**



**Simulador Microbit**



**Actividad exploratoria: Tinkercad**

**A MODO DE CIERRE**

---



**Spoiler alert**



**Bibliografía**

# Recapitulando

---

Hemos visto que **programar** es una actividad que nos otorga una primera comprensión de buena parte de nuestro entorno vital. **Indica un modo en el que otros han resuelto situaciones y problemas a través del Pensamiento Computacional.** Este modo de pensar y operar está asociado a las tecnologías digitales, aunque también hemos visto que la relación fuerte entre los instructivos y los algoritmos permite pensar en un modo general de pensamiento que podemos llamar “técnico”.

---

**Si logramos que los alumnos y alumnas experimenten todos estos modos de comunicar, tendrán la posibilidad de explotar las prestaciones de gran parte de la tecnologías que nos rodean diariamente, y desarrollar, además, habilidades que fomentan su despliegue mental y físico.**

Podemos decir entonces que los beneficios relacionados con la programación, son mucho más de los inicialmente pensados.

A continuación comentamos algunos de ellos.

1

**Desarrollo de la creatividad:** La programación se fundamenta en la espiral del pensamiento creativo de Mitchel Resnick. El alumnado parte de su

imaginación para crear un proyecto, bien sea un juego, una historia... A continuación, jugará y disfrutará de su creación y orgulloso/a lo compartirá con sus compañeros/as, lo que le llevará a reflexionar sobre posibles cambios o modificaciones para mejorarlo. Así volverá al punto inicial donde la imaginación se convierte en algo con lo que poder interactuar.

2

**Mejora la capacidad de resolución de problemas y desarrolla el pensamiento computacional:** Programar es tomar decisiones, los/as niños/as se encuentran constantemente con problemas que deben resolver. A través de la programación aprenden a definir problemáticas u objetivos, a determinar las herramientas que tienen para solucionarlo o conseguirlo y a decidir la forma más adecuada de hacerlo, donde los algoritmos son la clave.

3

**Fomenta la tolerancia a la frustración al mismo tiempo que mejora la autoestima y la confianza del alumnado:** Al programar son muchos los errores que nos podemos encontrar y tenemos que ir resolviendo. A largo plazo esto les ayuda a entender que lograr sus objetivos requiere de tiempo y esfuerzo. Ver que finalmente lo logran, les hace entender que todo el trabajo realizado merece la pena, mejorando su autoestima y confianza en sí mismos al verse capaces de lograrlo.

4

**Fortalece la autonomía de los niños y las niñas:** Generalmente el software diseñado para iniciar a los/as niños/as en el mundo de la programación es muy intuitivo, permitiendo al niño/a experimentar en función de su curiosidad y de forma autónoma. Los/as niños/as tienen claro a dónde quieren llegar y una vez que entienden la herramienta les gusta ir probando las posibilidades que les brinda.

El pensamiento computacional ayuda a tomar decisiones de una manera ordenada, secuenciada, lógica, sin ambigüedades, algo que a veces resulta difícil de observar en el ámbito de las ciencias de corte más social. Aprender a programar empodera al sujeto frente a un mundo lleno de objetos digitales programados por otros.

CONTINUAR

# Abordaje de la programación... propuestas para el aula

---

Como hemos visto, las propuestas pedagógicas que trabajan el desarrollo del Pensamiento Computacional van mucho más allá del abordaje basado en la programación, la codificación o la robótica la programación.

En este sentido se han establecido cuatro tipos de experiencias pedagógicas para abordar el Pensamiento Computacional en diferentes contextos educativos. ¿Cuáles son?

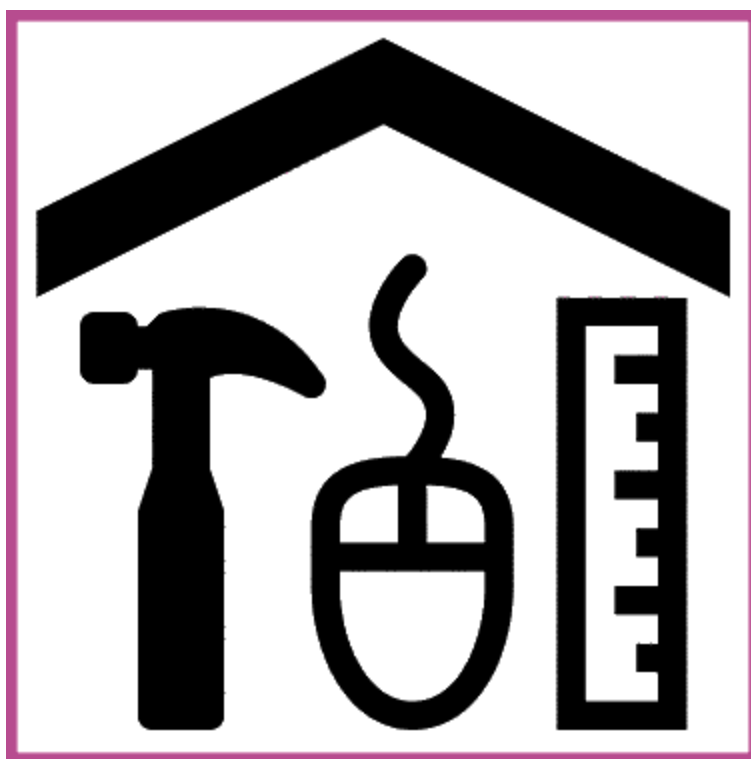
**Unplugged:** son experiencias que se trabajan al margen de computadoras y que se centran en la exploración y aplicación de los conceptos del Pensamiento Computacional. Nos pueden servir para un escenario analógico, en el que la escuela no dispone de equipamiento tecnológico pues no precisan de este y son fáciles de aplicar para docentes con poca experiencia



**Tinkering:** estas experiencias persiguen la modificación de un objeto preexistente y se basan en la búsqueda de un contexto de investigación. Los alumnos deben reflexionar e investigar sobre los posibles cambios a realizar y las implicaciones de estos cambios. Las

actividades de Tinkering suelen implicar el tratamiento de datos y prácticas de simulación, abordando la resolución de problemas de forma computacional. Tienen mayor carga tecnológica que las experiencias Unplugged.

*Making:* involucran la creación de ideas a partir de objetos y con un resultado único. Suelen llevar consigo procesos de creación de prototipos, lo que también implica reflexiones estructuradas y previas al desarrollo del concepto. Las experiencias making pueden llevarse a cabo de forma desconectada (utilizando, por ejemplo, materiales de autoconstrucción, objetos cotidianos, materiales reutilizables...) o implicar una creación digital a través de la computación y la programación.







**Remixing:** estas experiencias implican mezclar componentes de objetos ya existentes para crear unos nuevos con un determinado propósito. Para ser capaces de crear objetos a partir de esta “remezcla”, el alumnado debe ser capaz de descomponer un objeto existente en sus pequeñas funcionalidades o partes, de forma que puedan extrapolar su uso a nuevos prototipos. Por tanto, este tipo de experiencias requieren un mayor desarrollo cognitivo en relación con el Pensamiento Computacional y suelen trabajarse una vez se hayan realizado con anterioridad los otros tres tipos de actividades.

Todas estas propuestas pueden combinarse y adaptarse a multitud de contextos para ofrecer a todos los niños y niñas las mismas oportunidades de aprendizaje, en cualquier circunstancia y lugar.

Independientemente de los recursos involucrados en estos tipos de experiencias pedagógicas, es necesario recordar que buena parte del desarrollo de la Cultura

Tecnológica se asocia a un perfil activo, con alguna capacidad, al menos parcial, de abrir “cajas negras”. Esta actitud “*maker*” es capaz de desarrollar pensamiento técnico para la resolución de problemas, y cuenta con algún contenido específico para desarrollar su solución en el plano concreto, por ejemplo, a través de un programa.

**CONTINUAR**

# Más allá de la programación... Simuladores para experimentar y aprender

---

## ¿Qué entendemos por "simulador"?

---

**Un simulador es una máquina que reproduce el comportamiento de un sistema en ciertas condiciones, lo que permite que la persona que debe manejar dicho sistema pueda entrenarse. Los simuladores suelen combinar partes mecánicas o electrónicas y partes virtuales que le ayudan a generar una reproducción precisa de la realidad.**

Los pilotos de avión, los conductores de coches de competición y los cirujanos suelen practicar con simuladores. Hay simuladores de cabinas de aviación profesionales y lúdicas como el juego Flight Simulator. En ese caso el usuario (futuro Piloto) va ganando experiencia en un ambiente similar al que verá en la cabina del avión con los mismos instrumentos y controles. Sin embargo, al realizar una maniobra incorrecta podrá ver los resultados sin causar daños. Es una forma de ganar experiencia en un entorno virtual controlado.



El simulador permite al estudiante aprender de manera práctica, a través del descubrimiento y la construcción de situaciones hipotéticas. Un simulador tiene la ventaja de permitirle al estudiante desarrollar la destreza mental o física a través de su uso y ponerlo en contacto con situaciones que pueden ser utilizadas de manera práctica. Si son usados en trabajo colaborativo, estimulan el trabajo en equipo al estimular la discusión del tema.

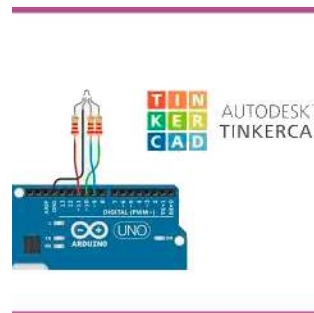
## Ejemplos de simuladores



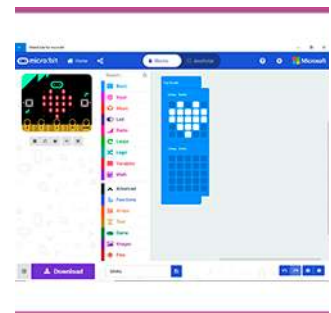
[Edumedia](#) contiene experimentos interesantes para utilizar como recursos educativos.



[vr.vex](#) que permite hacer el programa para el robot usando la programación en bloques y luego probarlo y verlo en acción en un simulador 3D.



[Tinkercad](#) es una herramienta online y gratuita. Ofrece la posibilidad de montar, programar y simular circuitos con Arduino



[Microsoft MakeCode](#) para micro:bit es una plataforma gratuita de código abierto para crear proyectos con BBC micro:bit.

El uso del simulador es muy útil en el campo educativo. Permite:

- Ganar experiencia tanto en el campo de la resolución de circuitos sin emplear componentes evitando cortocircuitos.
- Practicar secuencias de programación previamente a pasarlas a una placa de control.
- Resolver problemas de circuitos y programación más allá de contar o no con equipamiento robótico.

**CONTINUAR**

# Actividad 1- Propuestas para compartir

---

**i** **Aclaración: En este espacio se comparte solamente la consigna de la actividad realizada en el Trayecto. No requiere de participación.**



Compartí las siguientes propuestas vinculadas a conceptos trabajados en el texto.

1

**Una** propuesta pedagógica de Educación Tecnológica relacionada con **alguno de los cuatro** tipos de experiencias pedagógicas para abordar el Pensamiento Computacional en diferentes contextos educativos. (**Unplugged - Tinkering - Making - Remixing**). Puede ser que ya la hayas realizado en tu práctica docente o bien, pensada a partir de los conceptos desarrollados en el módulo.

2

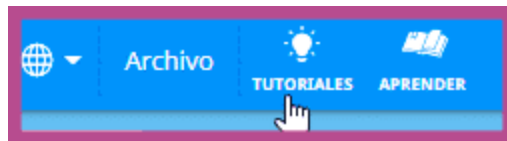
¿Haz incluido el uso de simuladores en el aula? En caso que la respuesta haya sido "Si", ¿podés compartir brevemente su experiencia?

CONTINUAR

# Simulador Vr Vex

---

**VEXcode VR** es una plataforma de programación on-line con la que puede **programar un robot VEX Virtual**, utilizando el lenguaje de **programación en bloques** Scratch 3.0. El robot virtual está basado en la línea VEX GO. Contiene **distintos escenarios**, los que adicionalmente **traen actividades prediseñadas con desafíos** para los usuarios.



<https://vr.vex.com/>

---

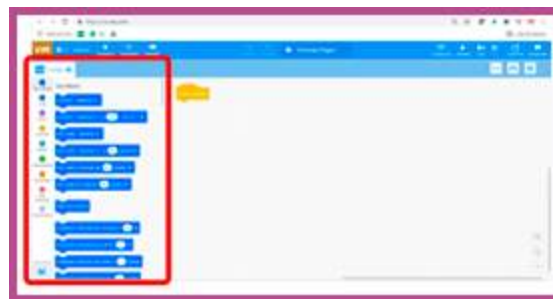
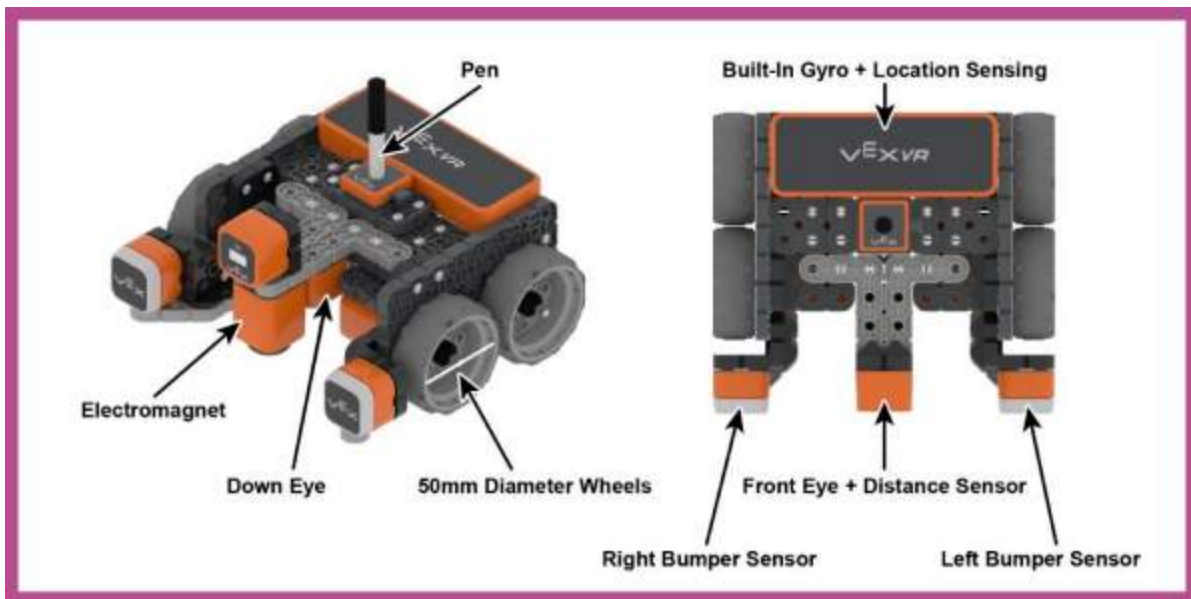
Posee pequeños vídeos explicando cómo se hacen las acciones más básicas con bloques. Al ingresar a la página, aparece la ventana de videotutoriales. Haciendo clic en la lámpara se abre un página donde se puede acceder a todos los tutoriales.

El robot cuenta con las siguientes características:

- Utiliza la base motriz (Drive Train) para navegar.
- Cuenta con los siguientes dispositivos: 1 electroimán, 2 sensores de ojo (distancia y color), 2 sensores de choque (bumper) a cada lado, encoders incorporados en sus motores (sensores de giro), 1 giroscopio y 1 sensor de posición (estilo GPS).



- Contiene un lápiz para realizar actividades en algunos de los escenarios



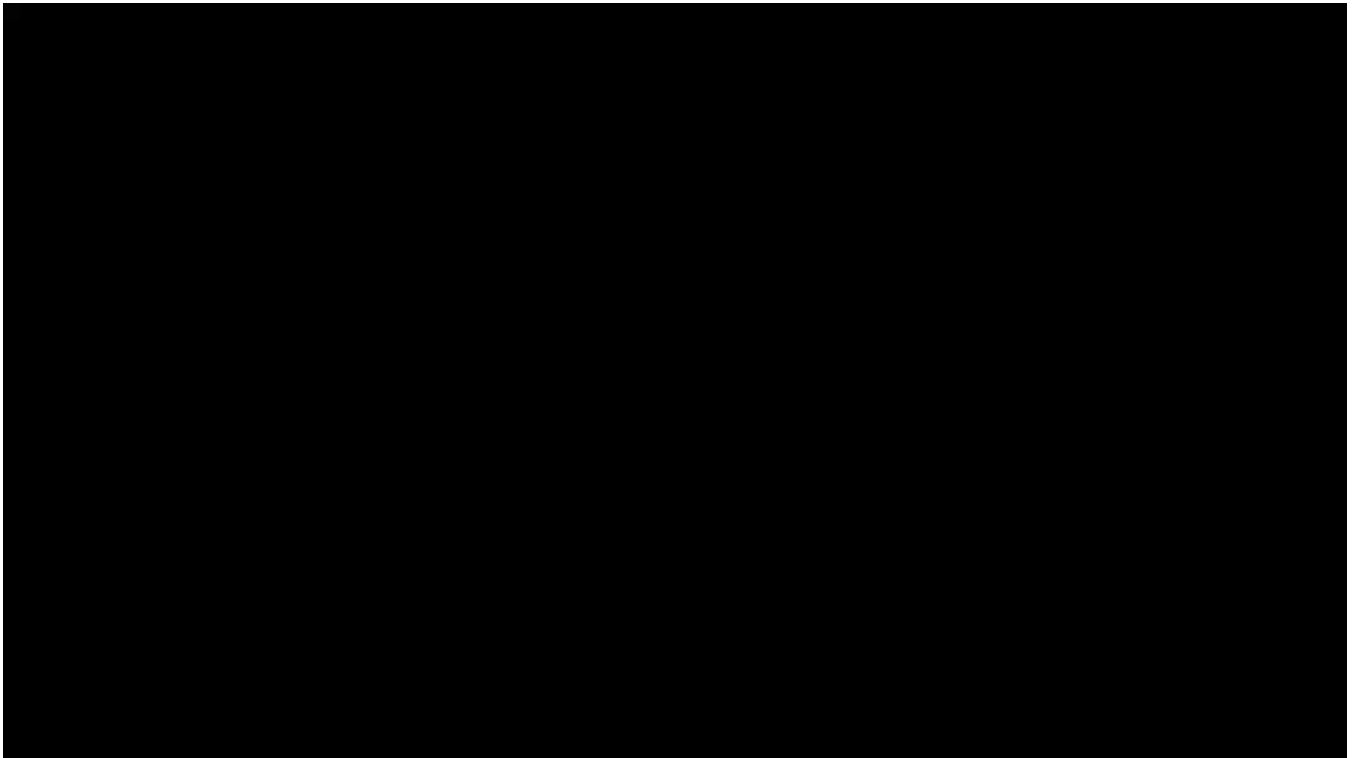
<https://vr.vex.com/>

El entorno de trabajo es muy similar a Scratch. Los bloques de programación están divididos por colores.

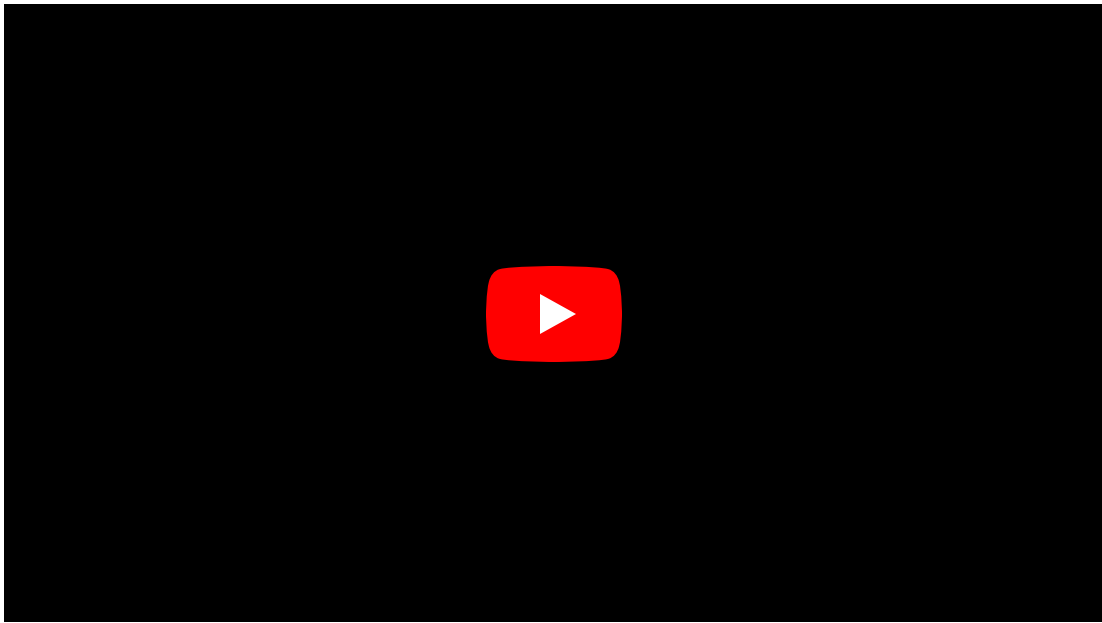
OJO: Estos bloques pueden cambiar según la zona de juegos que se haya elegido en el playground. Se debe elegir ANTES DE EMPEZAR A PROGRAMAR la zona de juegos en la que se desea ejecutar el programa.

---

**Para profundizar sobre las características de este simulador, las y los invitamos a visualizar los siguientes videotutoriales.**



 **YOUTUBE**



**Aprendiendo a programar un robot con el simulador VEX code VR (Realizando cuatro cuadros seguidos).**

**VER EN YOUTUBE >**

**CONTINUAR**

# Simulador Tinkercad

---

**Tinkercad** es una **herramienta online** ofrecida por **Autodesk**. Se utiliza de forma **gratuita** y sólo requiere crearse una cuenta de usuario. Entre sus utilidades, probablemente la más conocida es la de diseñar piezas en 3D. Sin embargo, **ofrece** también una posibilidad realmente interesante y es la de **montar, programar y simular circuitos con Arduino**.

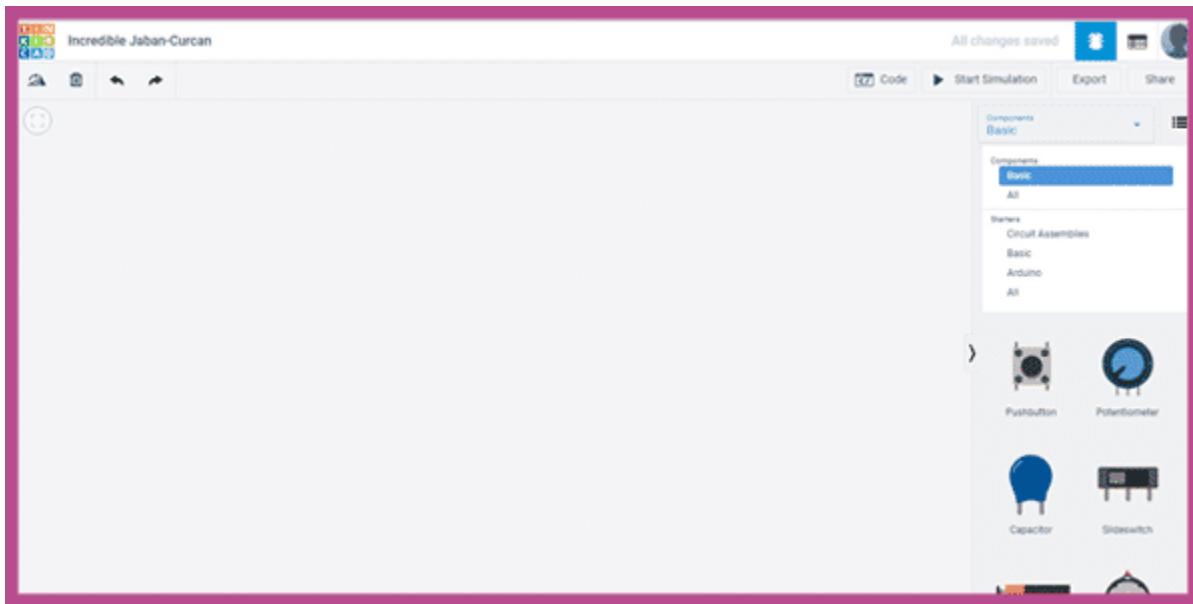


<https://www.tinkercad.com/>

---

Para ello, deberemos crearnos una cuenta de usuario y acceder. Seleccionando la opción **“Circuitos”** podremos empezar a crear nuestros circuitos clicando sobre **“Crear nuevo circuito”**

Tinkercad asignará a nuestro nuevo proyecto un nombre por defecto (*Incredible Jaban-Curcan, en la imagen*) y ofrece una serie de **componentes** en la parte derecha de la pantalla



<https://www.tinkercad.com/>

---

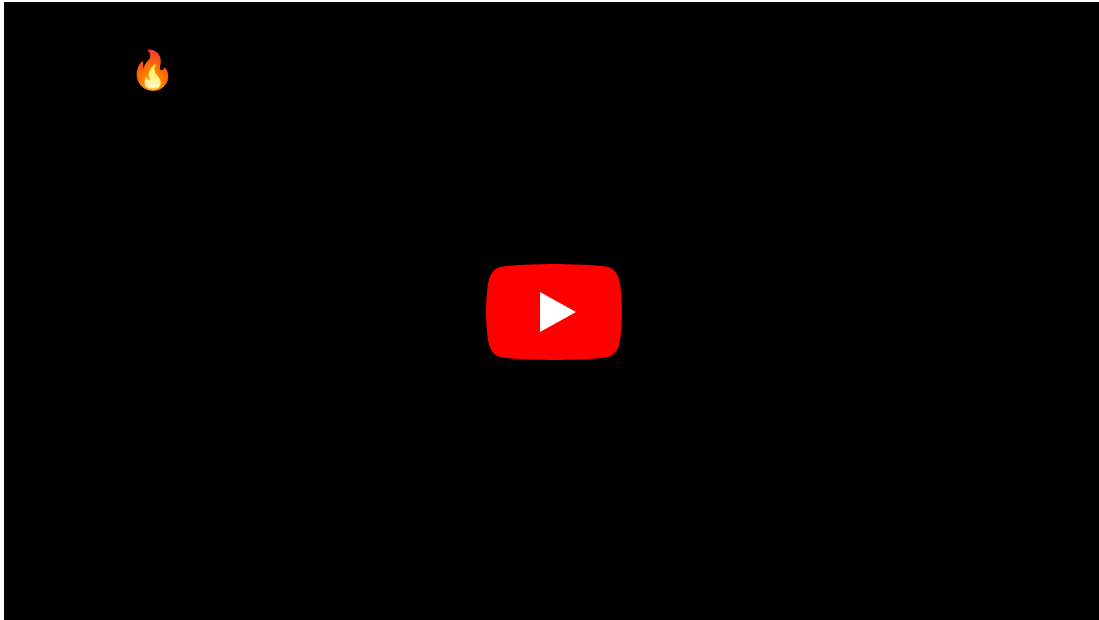
El cuadro de selección nos ofrece las opciones “Basic” y “All”, según queramos componentes elementales o de uso más común como pulsadores, LEDs, resistencias fijas, potenciómetros, baterías, transistores, placas de prototipado etc. u opciones más avanzadas como circuitos integrados, osciloscopios, control remoto o tiras de LEDs.




Sea cual sea la opción, clicando sobre el componente y arrastrándolo al área de trabajo podremos ir ensamblando nuestros circuitos.

---

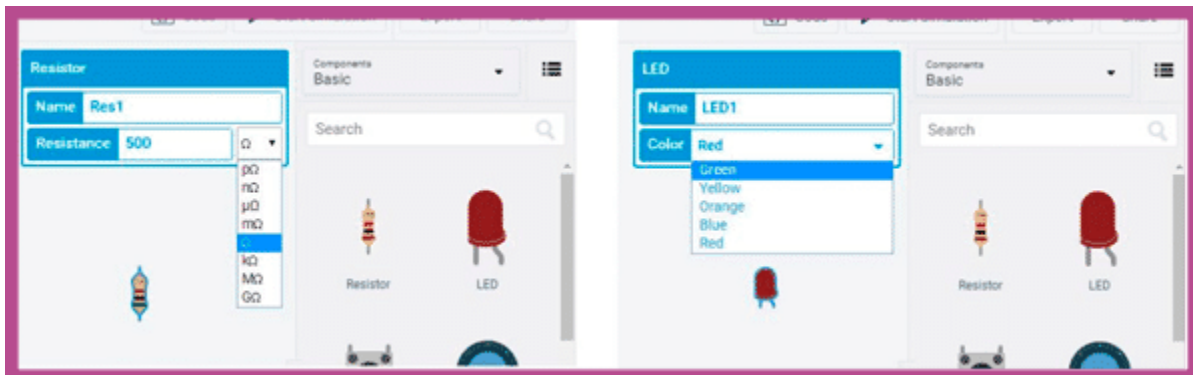
**Para profundizar en el entorno de Tinkercad, los y las invitamos a visualizar el siguiente video:**

 YOUTUBE



 **¿Como puedo comenzar sin tener un Arduino ni materiales? Desde cero con tinkercad  - Clase #2 **

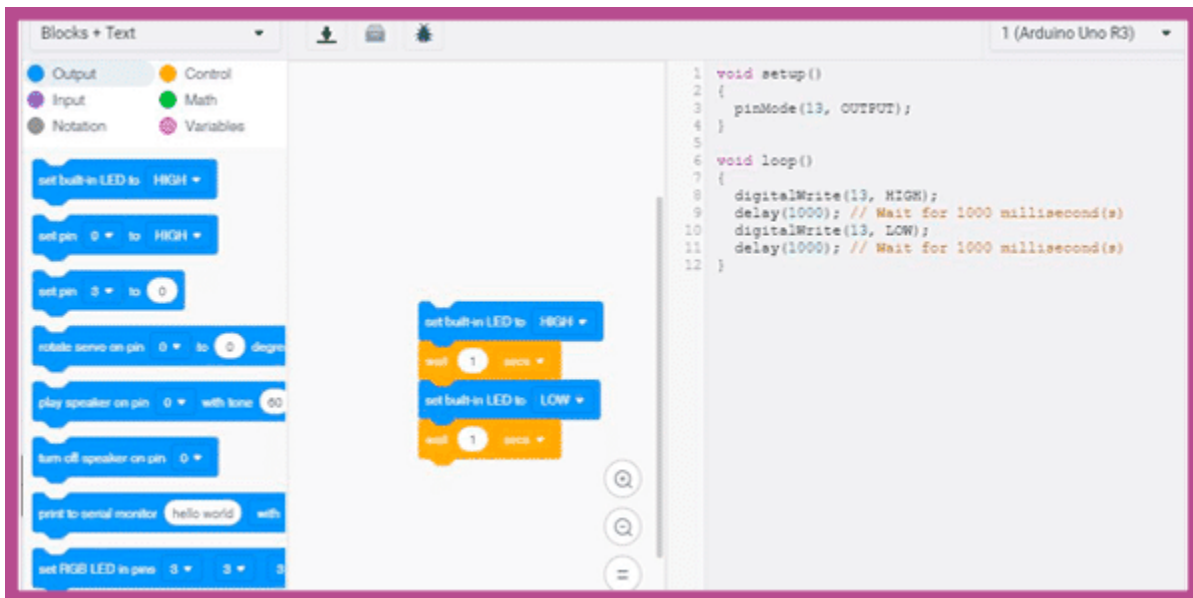
VER EN YOUTUBE >



Además, podremos nombrar cada componente y cambiar ciertas características, tales como valores, color, etc.

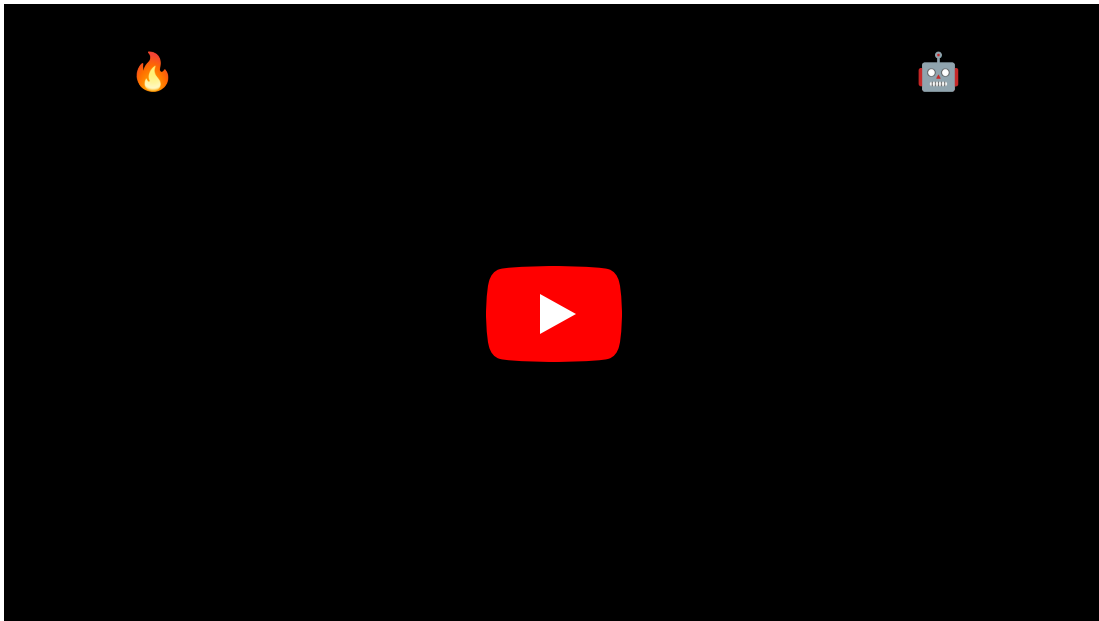
Una vez montado el circuito con una placa controladora Arduino, podremos **escribir el código**. Para ello, habrá que clicar sobre el icono “Code” y se optará por la opción de **programación por bloques, texto o mixta**. Ésta última muestra las dos opciones (bloques y texto) simultáneamente.

Escribimos el programa para la placa controladora (en la imagen, un led que prende y apaga) y a continuación podemos simular el circuito eligiendo la opción “Empezar Simulación”



Las y los invitamos a realizar la siguiente práctica para ahondar en las posibilidades que nos da este simulador

 YOUTUBE



**🔥 Mas leds y uso de la protoboard con tinkercad 🤖 - Clase #7 🚀**

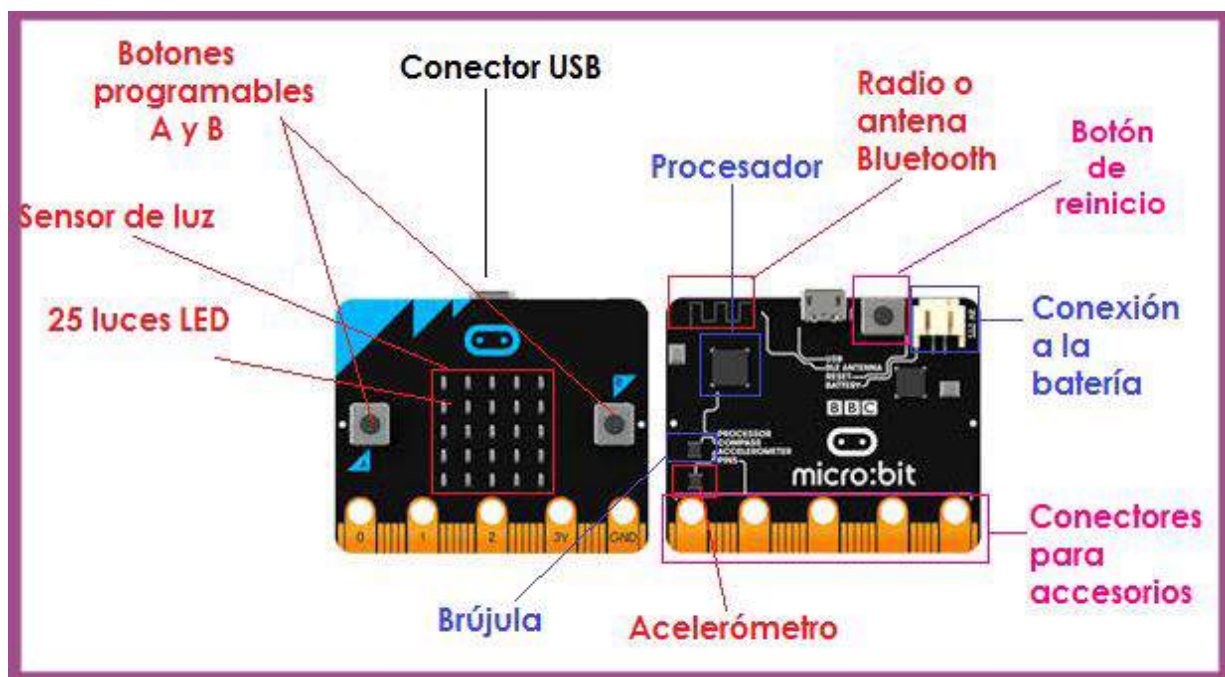
**VER EN YOUTUBE >**

**CONTINUAR**



# Simulador Microbit

Microbit es una pequeña tarjeta programable. A pesar de su tamaño muy reducido, **incorpora gran cantidad de sensores y actuadores**. Utiliza software *Open Source*. Es ideal para introducirse en el mundo de la programación de robots. Está compuesta por:



- Microprocesador con 512kB de memoria flash y 128 kB de RAM.
- Display con 25 LEDs programables de forma individual.
- 2 botones programables.
- Display luces LED.
- Pines de entrada y salida.

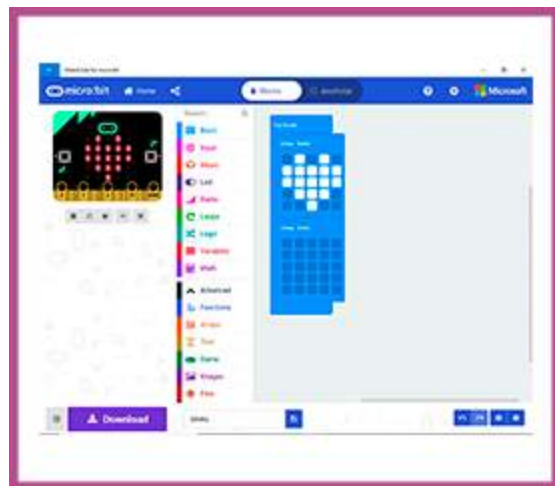
- Sensores de luz, temperatura y movimiento.
- Conectividad Bluetooth.
- USB y Conector para batería externa.
- Micrófono digital.
- Altavoz integrado.

Micro:bit es útil para implementar automatización, desde la robótica o la programación. Trae sensores incorporados, lo que se vincula con que sea potente. El sensor permite recuperar datos del exterior y a su vez, el comportamiento de la placa se refleja por una matriz de led.

El entorno de programación es online, lo que permite que se pueda hacer una programación con un simulador. Es decir, no siempre es necesario contar con la placa física para trabajar.

Permite crear casi cualquier cosa que pase por la mente: desde un videojuegos, ropa inteligente, música, robots, entre otros.

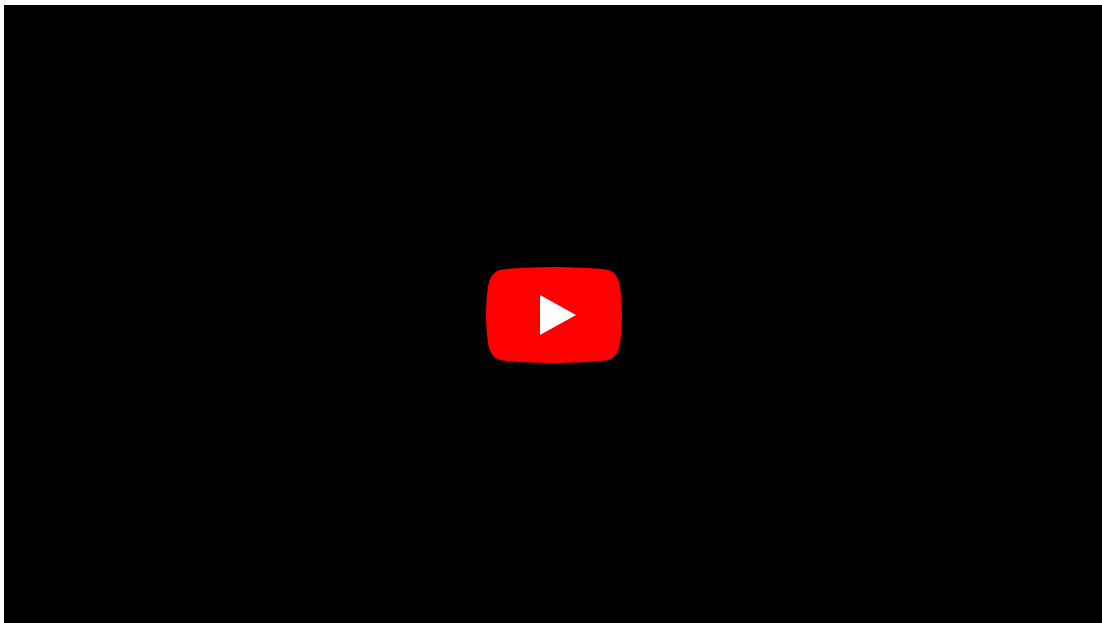
Hay varias plataformas que permiten programar con la micro:BIT, entre ellas destaca MakeCode, tanto en su versión online como offline. Que posee un simulador para trabajar con la placa aunque no dispongamos de ella físicamente.



<https://makecode.microbit.org/#>

---

**Para profundizar sobre las características de este simulador, las y los invitamos a visualizar el siguiente videotutorial.**



**CONTINUAR**

# Actividad exploratoria: Tinkercad

---

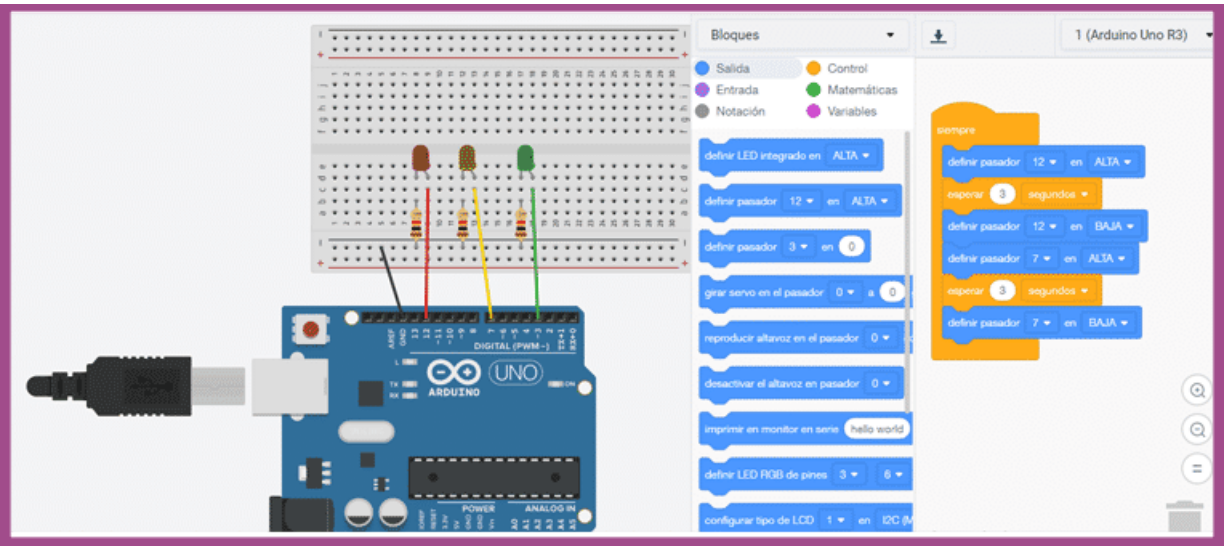
**i** **Aclaración: En este espacio se comparte solamente la consigna de la actividad realizada en el Trayecto. No requiere de participación.**

Durante esta clase comenzamos a explorar diferentes simuladores que permiten trabajar la programación y comenzar a adentrarnos en la robótica educativa, más allá de los dispositivos que tengamos disponibles.

En nuestro próximo módulo, seguiremos ahondando en ideas sobre la generación de secuencias de instrucciones, que puedan repetirse en forma cíclica.

Para finalizar nuestro módulo, les proponemos la siguiente actividad a partir del simulador [Tinkercad](#).

En la siguiente imagen se muestra un circuito con tres luces, sus respectivas resistencias, una placa Arduino y una placa de pruebas. Incluyendo un código para dicha construcción.



A partir de las imágenes mostradas las y los invitamos a responder:



## Actividad 1

---

- ¿Qué salidas están activas?
- ¿Cuál es el tiempo que está activa cada una de las salidas ?
- Con esa programación, ¿Se encienden todas las luces?
- ¿Hay algún tipo de diagrama que me pueda representar el comportamiento del sistema?
- Reproduzcan el esquema utilizando Tinkercad. Para ello, deberán crear una cuenta de usuario. Luego, realicen una simulación que permita encender las tres luces de manera intermitente simulando un semáforo. Deberán descargar el archivo (desde la opción "Enviar a") y adjuntarlo en el espacio de entrega.

---

**Los procesos y tecnologías de control por programa que son una variante del control temporizados logran su propósito mediante la predeterminación de la secuencia y duración de las operaciones intervinientes, por esa razón constituyen tecnificaciones y promueven la delegación de acciones técnicas más sofisticadas a los artefactos con sus efectos sobre la regularidad. Estas tecnificaciones son útiles cuando no tengo posibilidad o no decido medir para controlar se basan en una estimación de tiempo previamente definida.**

## Spoiler alert

---

**SPOILER  
ALERT!**

Durante esta semana comenzamos a explorar diferentes simuladores que permiten trabajar la programación y comenzar a adentrarnos en la robótica educativa, más allá de los dispositivos que tengamos disponibles.

Nuestra próxima parada es el encuentro presencial obligatorio, en el cual realizaremos una integración de los contenidos abordados hasta el momento.

**¡Los y las esperamos!**

**CONTINUAR**



# Bibliografía

---



- Gobierno de la Ciudad Autónoma de Buenos Aires. (2020) [Diseño curricular. Educación Digital, Programación y Robótica](#). Nivel Primario / 1a ed, Buenos Aires. Ministerio de Educación e Innovación del GCABA.
  
- Gobierno de la Ciudad Autónoma de Buenos Aires. (2012). Educación Tecnológica. En [Diseño curricular para la Escuela Primaria—Segundo Ciclo: Tomo 1](#) (pp. 381-426). Ministerio de Educación e Innovación.