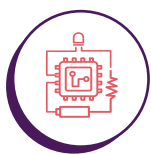


Sesión

2

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Aplicar bloques de programación secuencial para el control de máquinas de estados.



Realizar conexiones entre la *micro:bit* y actuadores externos (LED) en el entorno *Tinkercad*.



Simular código de programación y montajes electrónicos con la *micro:bit* desde el entorno *Tinkercad*.

Material para la clase

- ☐ Acceso a *Tinkercad* - cuenta de estudiante o personal.
- ☐ Anexo 2.1.

Duración sugerida

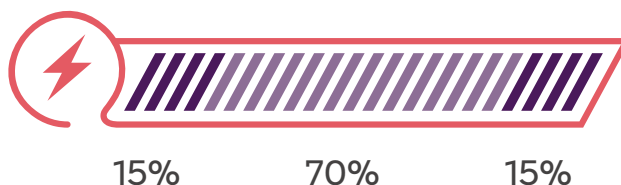
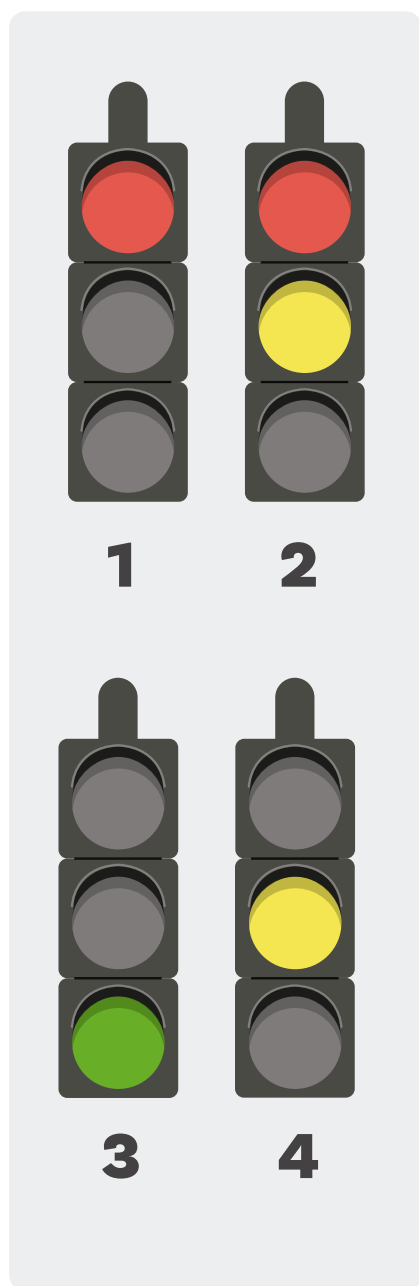


Figura 1. Semáforo



Lo que sabemos,

lo que debemos saber

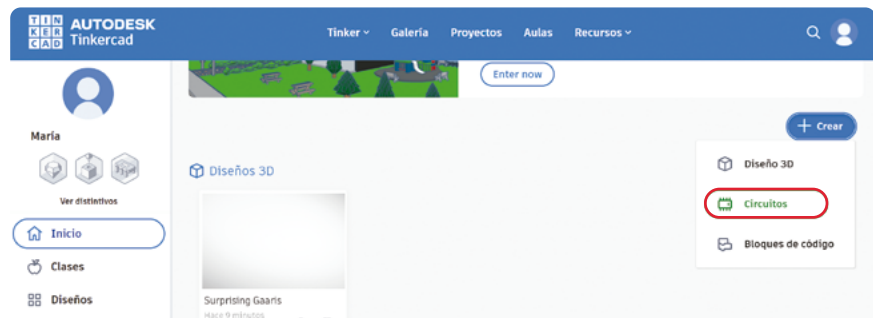


Esta sección corresponde al 15% de avance de la sesión

Ahora que entendemos cómo funcionan los sistemas automatizados, es hora de mejorar el flujo de tránsito en la ciudad del futuro. En esta sesión aprenderemos a programar un semáforo utilizando la *micro:bit* y algunos LED, para garantizar que tanto los vehículos como los peatones se muevan de manera segura y ordenada. Cada semáforo actuará como un guardián que previene accidentes y atascos, haciendo nuestras calles más seguras para todos los ciudadanos(as).

Un semáforo es un ejemplo de **máquina de estados**, es decir, un dispositivo cuyo comportamiento se define por varios estados, transiciones entre estos y acciones que se llevan a cabo en respuesta a ciertos eventos o condiciones. Por ejemplo, una lavadora es una máquina de estados, ya que inicia con el llenado, luego pasa al lavado, continúa con el enjuague y, finalmente, al secado. De manera similar, un semáforo funciona como una máquina de estados, y sus estados y transiciones pueden describirse de la siguiente manera, ver Figura 1:

- **Estado 1**
luz roja prendida, luz amarilla apagada, luz verde apagada (5 segundos).
- **Estado 2**
luz roja prendida, luz amarilla prendida, luz verde apagada (2 segundos).
- **Estado 3**
luz roja apagada, luz amarilla apagada, luz verde encendida (5 segundos).
- **Estado 4**
luz roja apagada, luz amarilla encendida, luz verde apagada (2 segundos).

Figura 2. Opción de crear circuitos en *Tinkercad*

Iniciemos con la programación en *Tinkercad* de un semáforo clásico de 4 estados, para lo cual primero debes dirigirte a www.tinkercad.com y abrir tu cuenta, una vez allí haz clic en el botón Crear y luego en la opción Circuitos como se muestra en la *Figura 2*.

Con base a tus experiencias anteriores con *Tinkercad*, en este simulador encontrarás una gran zona gris a la que llamaremos “Lienzo” y una barra vertical a la izquierda donde encontrarás los componentes. Dicha barra cuenta con una opción de filtro y un buscador de componentes para facilitar encontrar el componente requerido.

Glosario



Máquina de estados: un sistema con estados finitos, transiciones y acciones, cambiando entre estados según condiciones o eventos. Por ejemplo, un semáforo o una lavadora.



Actuadores: dispositivos que transforman la energía en movimiento, luz o cualquier otra acción física, permitiendo que un sistema automatizado interactúe con su entorno. Por ejemplo, motores, LED, parlantes, entre otros.



Pin GND: conector en la *micro:bit* que sirve como referencia de polo negativo para completar los circuitos eléctricos.

Figura 4. Conexión de resistencia a pin 0

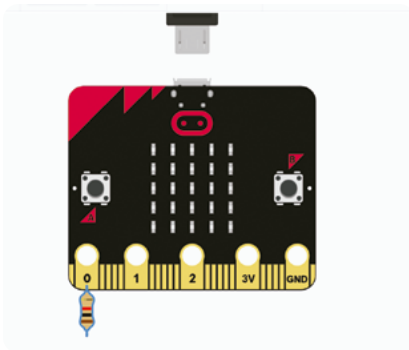
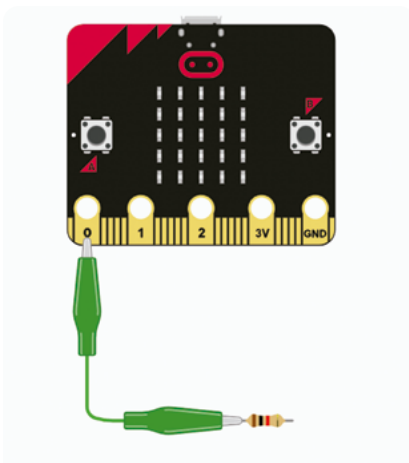


Figura 5. Conexión de resistencia con caimanes



Manos a la obra

Conectadas

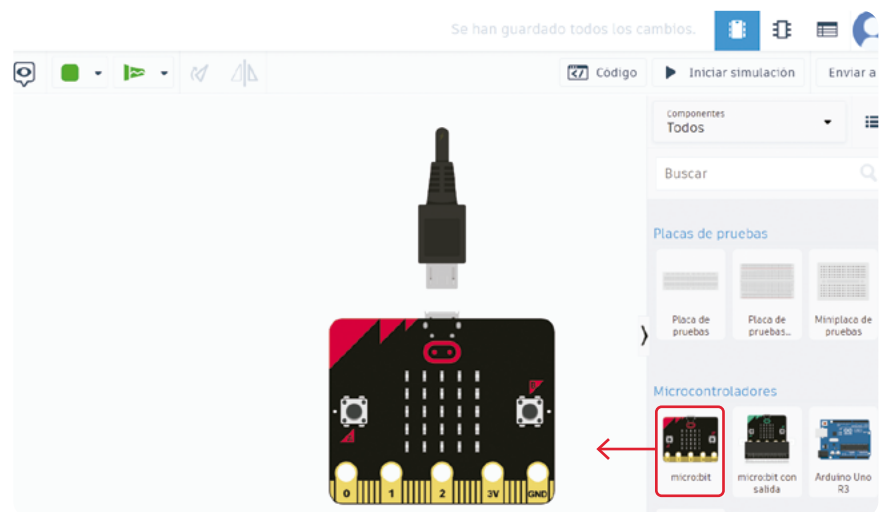


Esta sección corresponde al 85% de avance de la sesión

Sigue los pasos para hacer este montaje y simulación en el entorno *Tinkercad*:

Paso 1: busca entre los componentes la *micro:bit*, haz clic sostenido sobre ella y llévala al lienzo como se muestra en la *Figura 3*.

Figura 3. Seleccionar *micro:bit* en la lista de componentes en *Tinkercad*



Paso 2: busca entre los componentes la “Resistencia” y conecta al pin 0 tal como lo muestra la imagen *Figura 4*.

En caso de que tengas la *micro:bit*, cuentes con los componentes y quieras hacer montajes físicos, puedes usar cables caimán, tal como se muestra en la *Figura 5*.

O simplemente pela los extremos de un cable conductor y amarra las partes descubiertas a las terminales. No te preocupes por la electricidad en estos montajes, la corriente y el voltaje son demasiado pequeños para hacerte daño.

Para efectos de esta guía no usaremos cables-caimán en *Tinkercad*.

Figura 6. Conexión de led a resistencia

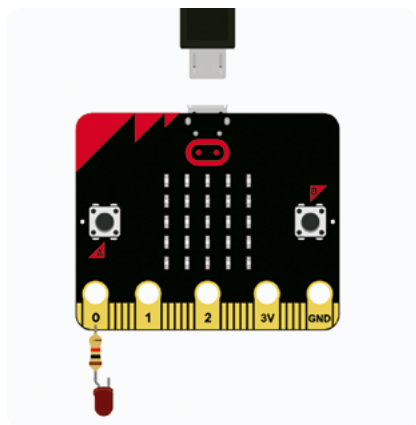
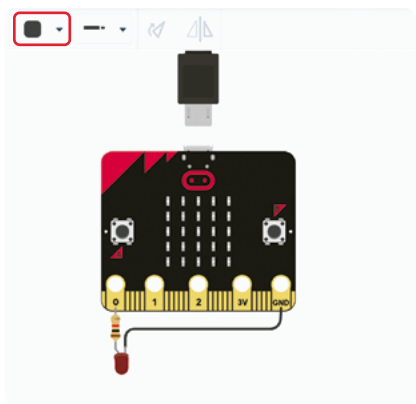


Figura 7. Cátodo del LED



Figura 8. Opción de cambio de color de cableado



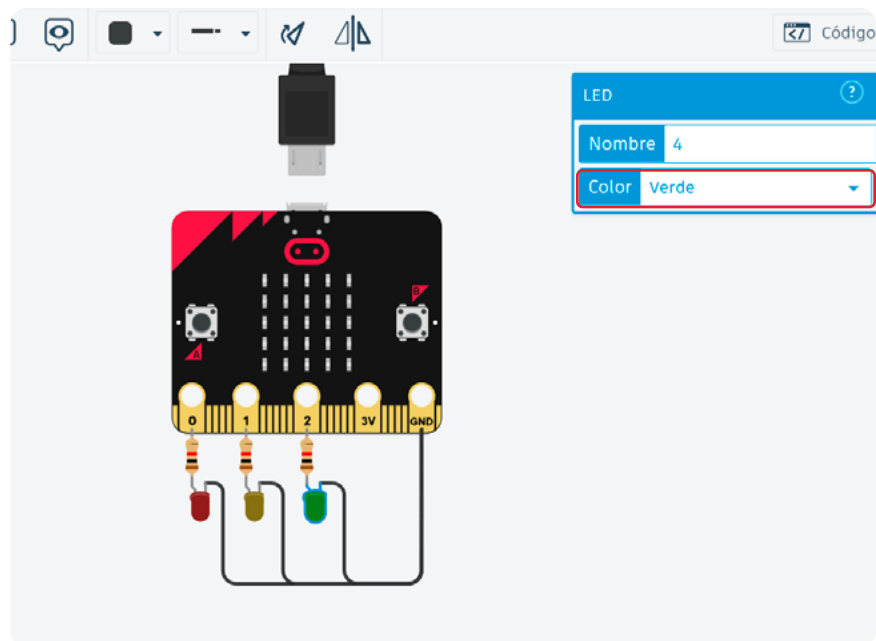
Paso 3: busca entre los componentes el LED, arrástralo al lienzo y hazlo girar oprimiendo la tecla “R” o dando clic sobre la herramienta.

Cuando esté girado conéctalo con la resistencia, tal como lo muestra la *Figura 6*.

Paso 4: haz un clic sostenido en la terminal del LED llamada “cátodo” como en la *Figura 7* y extiende el cable hasta el pin GND de la *micro:bit*. Puedes cambiar el color del cable dando clic sobre el cuadrado de colores, ver *Figura 8*.

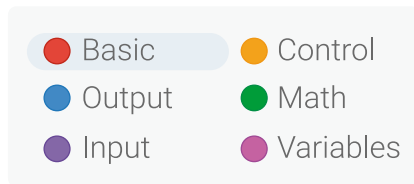
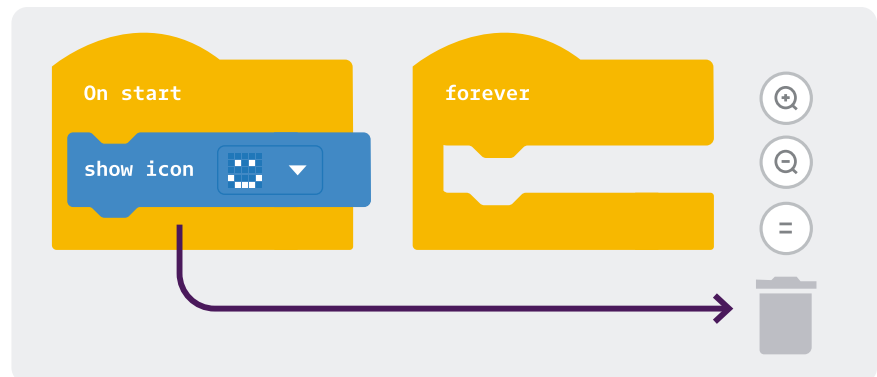
Paso 5: siguiendo el mismo procedimiento realiza el montaje de los otros dos LED, puedes cambiar el color del led en las opciones emergentes que aparecen cuando das clic sobre alguno de ellos, ver *Figura 9*.

Figura 9. Conexiones de LEDS a los pines del *micro:bit*



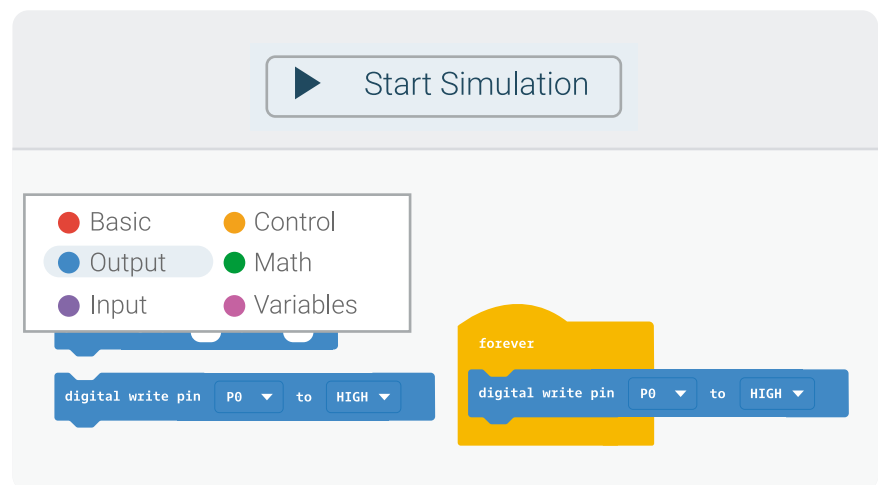
Una vez terminado el montaje vamos a iniciar con la programación, para lo cuál debes hacer clic en la opción “Código”.

Una vez en la zona de programación encontrarás una caneca donde puedes desechar los bloques que no vas a utilizar, *Figura 10*.

**Figura 10.** Opción para desechar los bloques

Al igual que en *Scratch*, encontrarás los bloques con los mismos colores de la paleta a la que pertenecen. Igualmente, los bloques se encajan uno debajo o dentro del otro.

Paso 6: busca en la paleta “Output” o “Salida” la opción pin 0-HIGH y conéctalo dentro del bucle “Forever” o “Para siempre”. Esta opción hará que nuestro LED rojo se encienda todo el tiempo. Puedes comprobarlo haciendo clic sobre el botón como se ve en la *Figura 11*.

Figura 11. Bloque con instrucciones “Output”

Paso 7: detén la simulación y programa el primer estado del semáforo, rojo conectado a pin 0 en HIGH o encendido, amarillo conectado a pin 1 en LOW o apagado y verde conectado a pin 2 también en LOW. Coloca el tiempo de espera para cada uno de los estados, ver *Figura 12*.

Anexo

Anexo 2.1

Anexo 2.1 Simulador

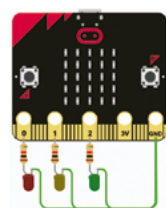


Figura 12. Programación de escrituras digitales en pines



Paso 8: programa los otros tres estados restantes teniendo en cuenta que no todos tienen los mismos tiempos. Simula el montaje con el código terminado y observa su comportamiento.

El código completo lo puedes observar en el Anexo 2.1.



¿Funciona cada uno de los cuatro estados? ¿Qué tendrías que hacer si quisiéramos hacer que dure dos veces más el semáforo en rojo que en verde?

Observa que este semáforo se programa de manera secuencial, un estado debajo del otro, cuando acaba uno inicia el otro.

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

De forma individual, regresa a revisar los aprendizajes de la sesión. Elige la opción de respuesta que mejor describa lo que alcanzaste.

- 1 ¿Puedes reconocer el entorno de simulación de circuitos electrónicos *Tinkercad*?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no

2 ¿Puedes conectar LEDs a la *micro:bit* y encenderlos y apagarlos de forma secuencial?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

3 ¿Puedes simular montajes electrónicos programados en *micro:bit* desde el entorno *Tinkercad*?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

En caso que tus respuestas a hayan sido “Parcialmente” o “Aún no”, discute con tu compañera o compañero de grupo el trabajo realizado sobre la simulación realizada en Tinkercad, verifica la solución que dieron a uno de los ejercicios planteados y consulta con tu docente las preguntas que aún tengas.

Las preguntas que te proponemos ahora pensar las siguientes preguntas:



Si regresamos al reto, ¿para qué serviría lo que hemos aprendido en esta sesión?

Algunos semáforos son inteligentes, dado que pueden saber si hay vehículos o no esperando.

¿Podrías indicar en palabras lo que habría que hacer para que este programa funcione?

Utiliza este espacio final para crear un esquema que resuma lo aprendido durante la actividad conectada, incluyendo definiciones de términos clave y ejemplos del uso de los conceptos aplicados al programar el semáforo.