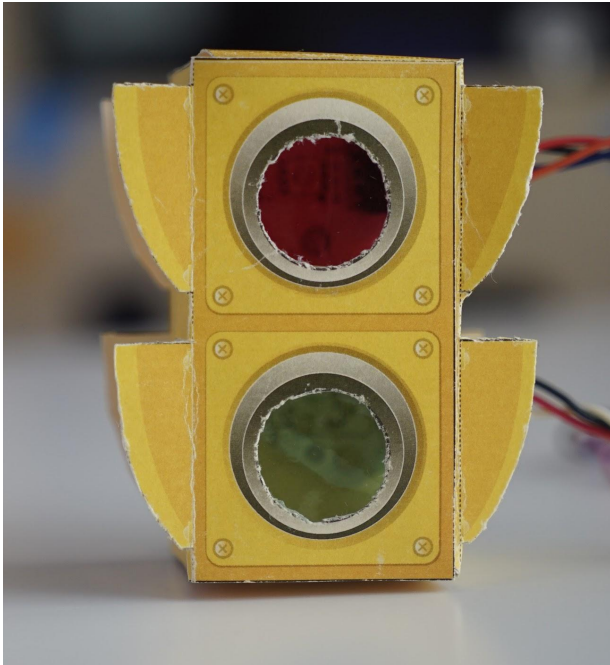


Construye y programa tu semáforo



En este proyecto vas a descubrir cómo construir y programar tu propio semáforo, y cómo coordinar el funcionamiento de varios semáforos a la vez para que, tanto peatones como coches, puedan circular con seguridad.

Nivel de dificultad: Bajo

Tiempo estimado: 2-3 horas

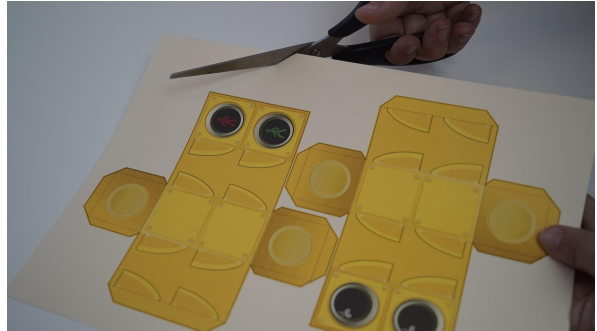
Materiales:

- Plantillas de semáforos
- Cartulinas blancas tamaño DIN A4
- Papel cebolla o celofán de colores rojo, amarillo y verde
- Impresora 2D
- Tijeras
- Pegamento
- Punzón
- Cinta adhesiva

Construye tu semáforo

Ahora que tenemos la programación de nuestro semáforo, vamos a construirlo siguiendo los siguientes pasos:

1- Lo primero que debemos hacer es pedir a nuestro docente la plantilla o recortable de nuestro semáforo. Hay dos modelos diferentes, un semáforo para peatones y otro para vehículos. Debemos recortar el semáforo por las líneas negras exteriores.

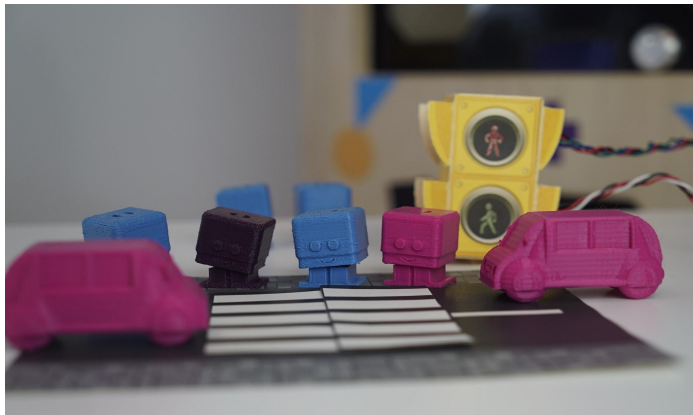
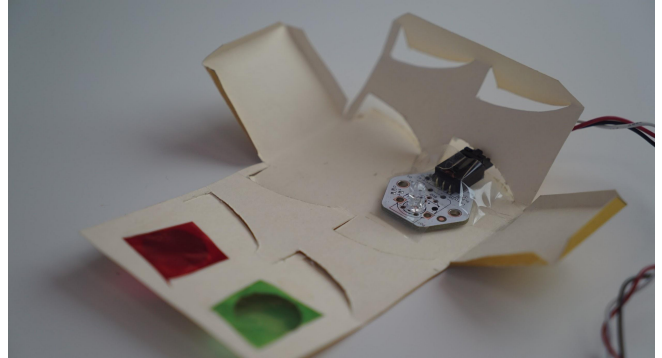


2- Recortamos las aletas laterales de los semáforos. Para ello utiliza un punzón o, si lo deseas, un *cutter*. ¡Protege la mesa donde vas a recortar y hazlo con cuidado!

3. Doblamos las pestañas por las líneas punteadas y colocamos trozos de papel cebolla o celofán del color correspondiente en cada agujero. Para el semáforo de peatones, podemos hacer pequeños agujeros con el punzón sobre las figuras, así los agujeros dejarán pasar la luz simulando que las figuras están formadas por pequeños LED.



4. Introducimos los cables de los LED por las aletas laterales. Colocamos cada LED de forma horizontal en la pared trasera del semáforo mediante cinta adhesiva. Colocamos el LED verde para la luz verde y el LED RGB para la luz roja. Debemos intentar que las luces estén lo más centradas posibles.



5. Ensamblamos el semáforo utilizando pegamento en las pestañas y finalmente, colocamos nuestro semáforo en cualquier lugar de la ciudad. Podemos improvisar un poste para él utilizando un lápiz, una barra de pegamento o cualquier otro elemento que pueda sujetar el semáforo.

¡A programar!

Antes de empezar, es importante que conozcamos cómo funciona Bitbloq. Para ello, vamos a crear un programa que encienda un LED.

¿Cómo conectar el LED a la placa?, ¿y en Bitbloq?

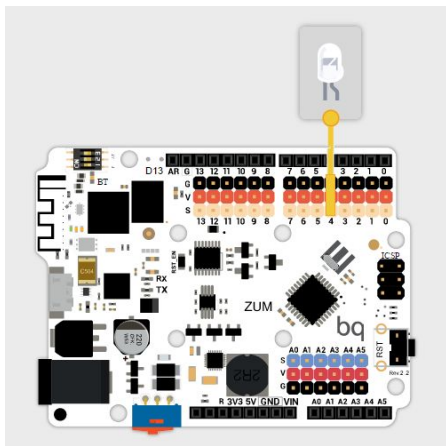
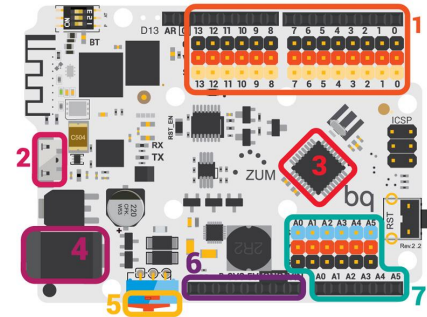
Antes de conectar un componente a la placa, deberemos pensar si es un componente **analógico**, aquellos que pueden tener más de dos valores, o **digital**, aquellos con solo dos valores (0 o 1).

En nuestro caso, el LED es un componente digital ya que solo tiene dos estados, o encendido (1) o apagado (0). Por ello, deberemos conectarlos a los pines digitales de nuestra placa controladora.

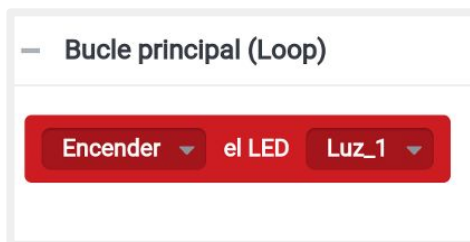
¿Cuáles son los pines digitales de la placa controladora?

Son los que corresponden con el número 1 de la siguiente imagen.

* Debemos intentar no utilizar los pines digitales 0 y 1 de la placa, ya que a través de estos dos pines, la placa se comunica con el ordenador y otros aparatos por lo que en ocasiones pueden dar problemas.



Una vez que tenemos el LED conectado a la placa, deberemos ir a Bitbloq y en la pestaña *Hardware*, arrastrar nuestra placa y el LED, conectándolo al mismo pin que utilizamos en la placa física.



¿Qué bloques necesito para encender un LED?

Una vez que has conectado tu LED, hay que programarlo para que se encienda. Para ello, solo debes seleccionar el bloque del LED en *Componentes* y arrastrarlo dentro del bloque *Loop()*.

Conecta tu placa al ordenador y clicas sobre el botón cargar. Tras cargarse el programa en tu placa, tu LED se encenderá.

Si tienes problemas para encender tu LED, consulta el siguiente enlace:

<http://diwo.bq.com/antes-de-empezar-con-bitbloq-2/>.

Con este programa hemos aprendido cómo encender un LED, y por tanto, cómo encender las luces de nuestro semáforo. Sin embargo, ciertas luces en los semáforos parpadean para avisar de que el semáforo está a punto de cambiar a rojo. Vamos a programar este caso.

¿Cómo programamos que un LED parpadee?

Para que parpadee, primero tiene que encenderse, luego apagarse y así sucesivamente. Por ello, deberemos incorporar una espera, utilizando el bloque *esperar*, que se encuentra en la categoría *Control*.

Una vez que tenemos el programa creado, podemos probar a modificar la velocidad a la que parpadea el LED, cambiando el valor del bloque *esperar*.

Otros ejercicios que podemos hacer para conocer más de la programación en Bitbloq son que dos LED parpadeen a la vez o que dos LED parpadeen alternativamente.

Programando nuestro semáforo

Vamos a programar un semáforo con dos luces, una verde utilizando un LED normal y otra roja, utilizando el LED RGB.

El LED RGB, está formado por tres luces distintas: una roja, una verde y una azul, cuya combinación hace que se obtengan muchos otros colores.

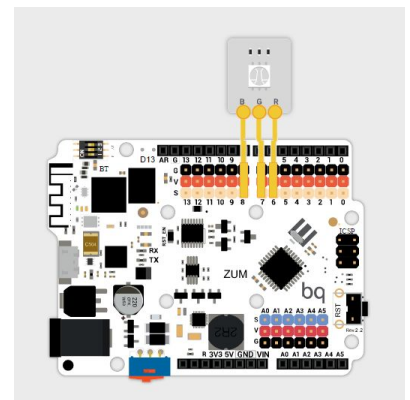
Para programar nuestro LED RGB, lo que necesitamos conocer es lo siguiente:

- **¿Cómo lo conecto?**

Este componente tiene 4 cables. Deberemos conectar el cable negro o GND a cualquiera de los pines digitales de color negro, correspondientes a tierra, y los otros cables a los pines digitales de señal, los mismos en los que los hayamos conectado en Bitbloq.

- **¿Cómo lo apago?**

Para que permanezca apagado, deberemos utilizar el siguiente bloque y poner todos los valores a 0.



Encender el led RGB **led_RGB_0** con un valor de rojo de **0**, un valor de verde de **0** y un valor de azul de **0**

- **¿Puedo crear otros colores?**

Utilizando el bloque anterior pero asignando valores correspondientes al color. Para consultar los valores que tiene un color podemos utilizar el siguiente enlace: http://rapidtables.com/web/color/RGB_Color.htm.

Una vez que conocemos cómo funciona un LED RGB, programamos nuestro semáforo para que se encienda primero la luz verde y luego la luz roja.

Ahora, programaremos que la luz verde parpadee antes de pasar a rojo, con el fin de avisar al peatón o vehículo y que puedan anticiparse y parar con seguridad.

Para realizar esta programación deberemos utilizar dos variables: una en la que almacenaremos el tiempo que tiene que pasar la luz verde o roja encendida y otra variable *cuenta* con la que el programa contará el tiempo que parpadee la luz verde.

Una vez que hemos declarado las variables, deberemos decirle al programa que al iniciarse se apaguen ambos LED.

A continuación, en el bucle principal deberemos primero encender la luz verde y apagar la roja, que espere un tiempo y que empiece a parpadear la luz verde.

Para hacer que la luz verde parpadee deberemos utilizar el bloque *Contar*:

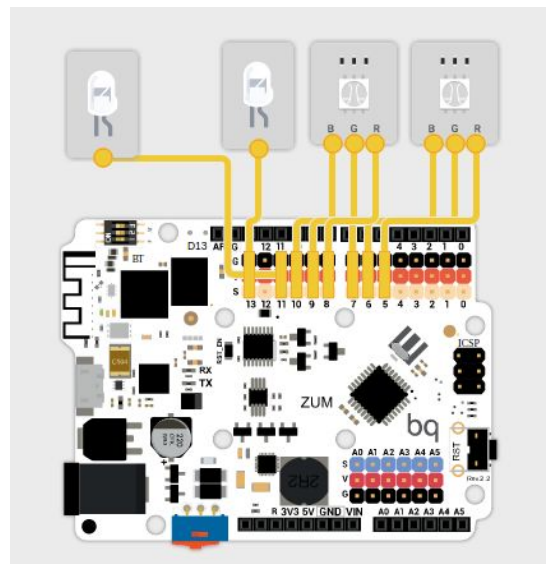


Por último, deberemos apagar la luz verde y encender la luz roja.

Deberemos comprobar que todo funciona y... ¡ya tendremos la programación de nuestro semáforo terminada!

Coordinando dos semáforos

Vamos a programar nuestro semáforo de peatones y el de vehículos para que funcionen de forma sincronizada, de forma que cuando el de vehículos esté en verde, el de peatones esté en rojo y viceversa. Para ello, conectaremos ambos semáforos a la misma placa.



Para que nos sea más fácil recordar, podemos cambiar los nombres de cada componente (por ejemplo, rojo_peaton, verde_peaton, rojo_vehiculo, verde_vehiculo).

Recordemos que queremos que los dos semáforos funcionen sincronizados, por lo que tenemos que juntar sus dos programaciones en una sola, es decir cuando se enciende el LED verde de uno, se deberá encender el LED rojo del otro, y viceversa.

Ampliando los semáforos

Si queremos continuar con la actividad, podemos realizar las siguientes propuestas:

- Imprimir la plantilla del semáforo de 3 luces adjunta en el Anexo I o construye tu propio semáforo (con material reciclado o impresión 3D) y programarlo añadiendo un LED para la indicación en ámbar.
- Incluir sonido al semáforo de peatón con el fin de poder indicar mediante señales acústicas cuando los peatones tienen que pasar.
- Hacer que la señal acústica sea más rápida a medida que se acerque el momento de ponerse en rojo.
- Programar que la luz verde parpadee cada vez más rápido según el tiempo que quede para ponerse en rojo el semáforo.
- Añadir un botón al semáforo de peatón y programar los semáforos para que sólo deje pasar personas tras pulsar el botón. Además, se puede incorporar una pantalla o LCD que muestre mensajes como "Pulse peatón" o "Espere verde".



Anexo I:

Semáforo de 3 luces.

Fundación **MAPFRE**

