

Tabla de Contenidos

Max:Bot	4
Introducción a Max:Bot	4
¿Cómo definir un robot?	4
Lista de componentes del Max:Bot	5
Una breve introducción a Micro:bit	6
Que es Micro:bit?	6
Cómo programar Micro:bit	8
Interfaz gráfica de programación MAKECODE	8
Programación con Micropython.....	9
Cómo programar Micro:bit a través de Makecode?	9
Abrir MakeCode	10
Interfaz de programación	10
Descargue el programa y cárguelo en micro:bit.....	12
Capítulo 1: Iniciar Max:Bot	14
Objetivos:	14
Componentes	14
1.1 Ir hacia adelante	15
Información básica	15
Programar	15
Ejercicios.....	17
1.2 Ir en Círculos	18
Información básica:	18
Por favor, tenga en cuenta que la " function " y la " call function " se encuentran en " Advanced " => " Functions ". Pero ambos no están listados en él directamente. La manera de encontrarlos es así:	19
1. La manera de encontrar la " function ": " Advanced" => " Functions" => " Make a Function". Haga clic en " Make a Function ", nómbralo y haga clic en " OK ". Entonces encontrará que este bloque de " function " aparece automáticamente en el área de programación.	19
2. Otra forma de encontrar la " call function ": al mismo tiempo, el bloque " function " aparece en la zona de programación, el bloque " call function " aparece en la zona de " Functions " de la zona de funciones.....	19
Por favor, recuerde que la " call function " sólo aparece cuando se crea la " function "	19
Programar	19
Ejercicios.....	22
1.3 Formar un cuadrado	22
Información básica	22
Programar	23
Ejercicios.....	26
Capítulo 2: Max:Bot Inteligente	27
Objetivos:	27
3. Programación asistida por diagramas	27

Digital Modules	27
2.1 El Max:Bot "tímidо"	28
Información básica	28
Programar	30
Ejercicios.....	34
2.2 Escapar del laberinto	35
Información básica	35
Programa	37
Ejercicios.....	39
Capítulo 3: Sensor de Luz	39
Objetivos	39
Programación asistida por diagramas.....	40
Módulos digitales.....	40
3.1 Aventura al brillo	40
Información básica	41
Programa	41
Ejercicios.....	42
2.2 El Robot Polilla	43
Información básica	43
Programa	44
Ejercicios.....	45
Capítulo 4 Ultrasonido, el arma secreta	46
Objetivos:	46
Módulos Digitales.....	47
4.1 Medidas de distancia por ultrasonidos	47
Información básica	49
Programa	49
Ejercicios.....	52
4.2 Protección del automóvil	53
Información básica	54
Programa	54
Ejercicios.....	57
Capítulo 5: Max:Bot Go!	58
.....	58
Objetivos:	58
¿Cómo utilizar la función "and"?	58
Módulos digitales.....	58
5.1 Max:Bot despierta!	59
Información básica	60
Programa	61
Ejercicios.....	63
5.2 Rastrea la pista	64
Información básica	65

Programa	65
Ejercicios.....	68
Capítulo 6: Radiocomunicación.....	68
Objetivos:	69
Módulos digitales.....	69
6.1 Di Hola!	69
Información básica	72
Programa	72
Ejercicios.....	75
6.2 Carrera de detección de movimiento.....	76
Información básica	78
Programa	79
Ejercicios.....	84
Capítulo 7: El espectáculo de luz y sonido.....	84
Objetivos:	85
Módulos Digitales.....	85
7.1 Tira de luces arcoíris.....	86
Información básica	86
Programa	87
Ejercicios.....	89
7.2 A poner Música!.....	90
Información básica	90
Programa	90
Ejercicios.....	92
7.3 El espectáculo de luz y sonido!.....	93
Información básica	93
Programa	94
Ejercicios.....	101

* Sea bienvenido al blog de DFRobot: www.dfrobot.com/blog

Max:Bot

Introducción a Max:Bot

Max:Bot, una plataforma móvil fácil de usar con las interfaces mejor diseñadas, puede proporcionar varias funciones para sus usuarios. Puede funcionar perfectamente cuando se conecta a Micro:bit. Además, Max:Bot lleva casi todas las capacidades fundamentales de un robot. Sin embargo, lo que puede sorprenderle más es la carcasa brillante totalmente metálica que es fácil de montar.

- La mejor combinación con Micro:bit
- Configuración rápida y fácil de usar
- Diseño de aluminio, más sólido y duradero
- Disponible para actualizar y conectar a tarjetas de expansión. Soporta más de 100 módulos digitales de Gravedad.

¿Cómo definir un robot?

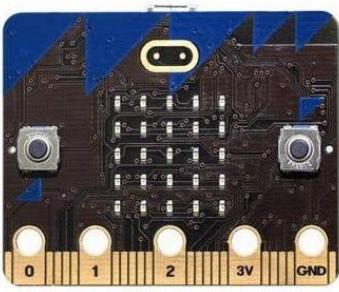
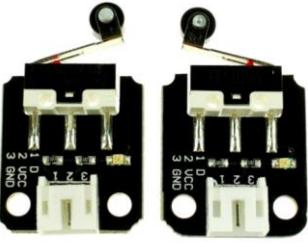
Generalmente, los robots son dispositivos automatizados con cierta potencia de cálculo. Al igual que las criaturas vivientes, los robots tienen sus propios "órganos":

El cerebro: micro:bit actúa como el cerebro. Con su propio sistema lógico, la placa puede procesar la información y enviar los comandos correspondientes.

El sistema sensorial: Max:Bot está conectado con sensores de diferentes funciones, por lo que obtenemos el sistema sensorial. Por ejemplo, un sensor de escala de grises y un sensor de colisión se pueden utilizar como el ojo y la mano para recibir información externa.

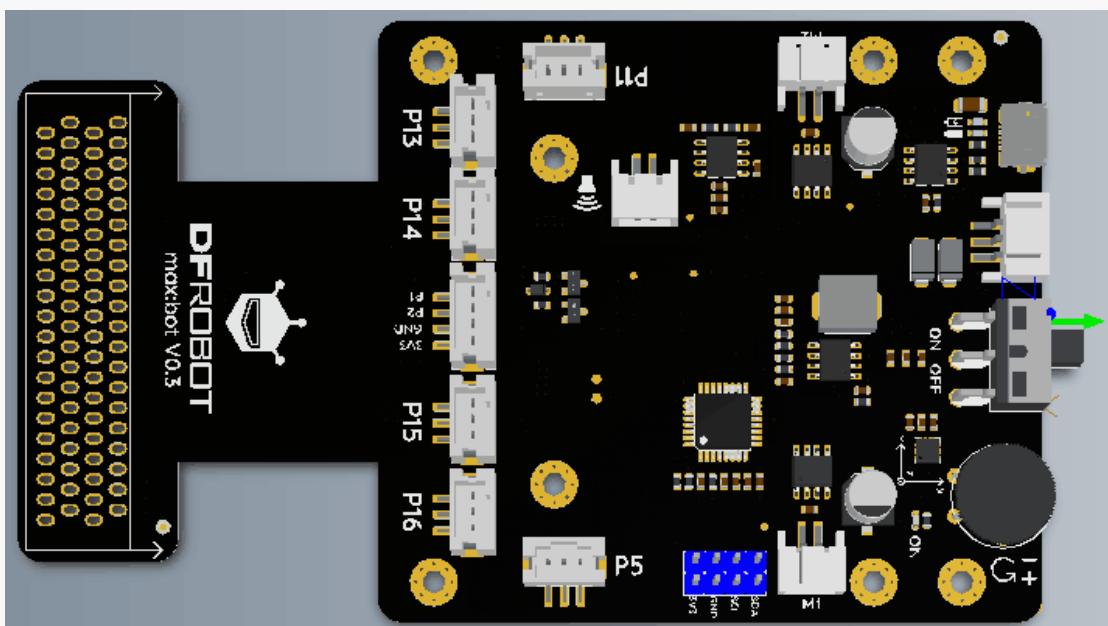
El sistema ejecutivo: Este sistema puede cambiar nuestro entorno o enviar mensajes al mundo exterior a través de gestos, lenguajes, expresiones, etc. Las tiras de LEDs y los módulos de altavoces digitales se pueden conectar al Max:Bot para enviar masajes. Además, hay un panel LED con 25 LEDS en el micro:bit y se pueden utilizar para mostrar mensajes y patrones.

Lista de componentes del Max:Bot

Nombre	Imágenes	Pin(s) a conectar
micro:bit		
Sensor de colisión (Left) Sensor de colisión (Right)		P13、P16
Sensor de Ultrasonido		P1、P2
Rastreador de líneas		P14、P15
Tira de Leds		P5、P11

Motor		P8、P12

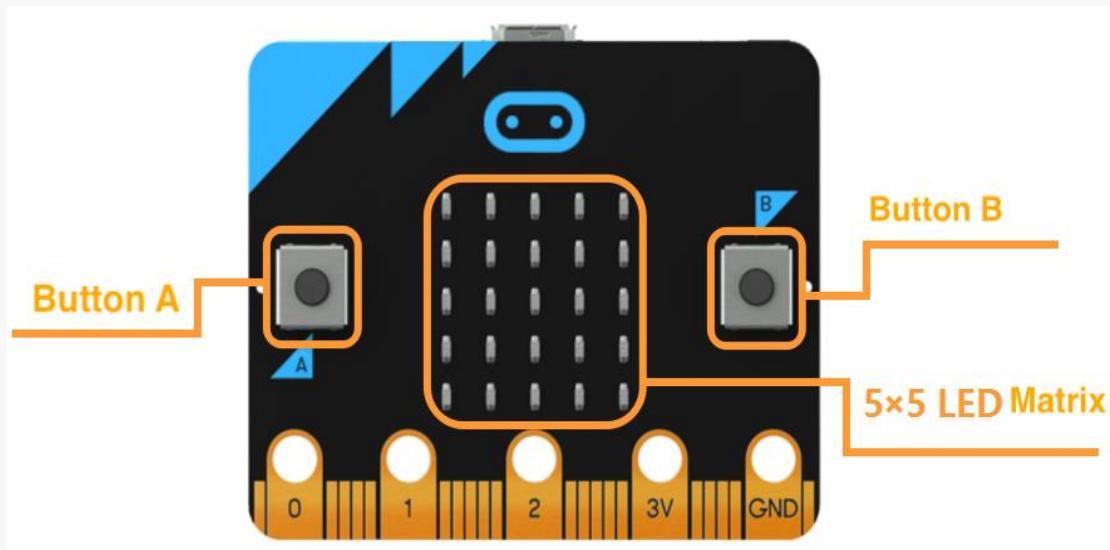
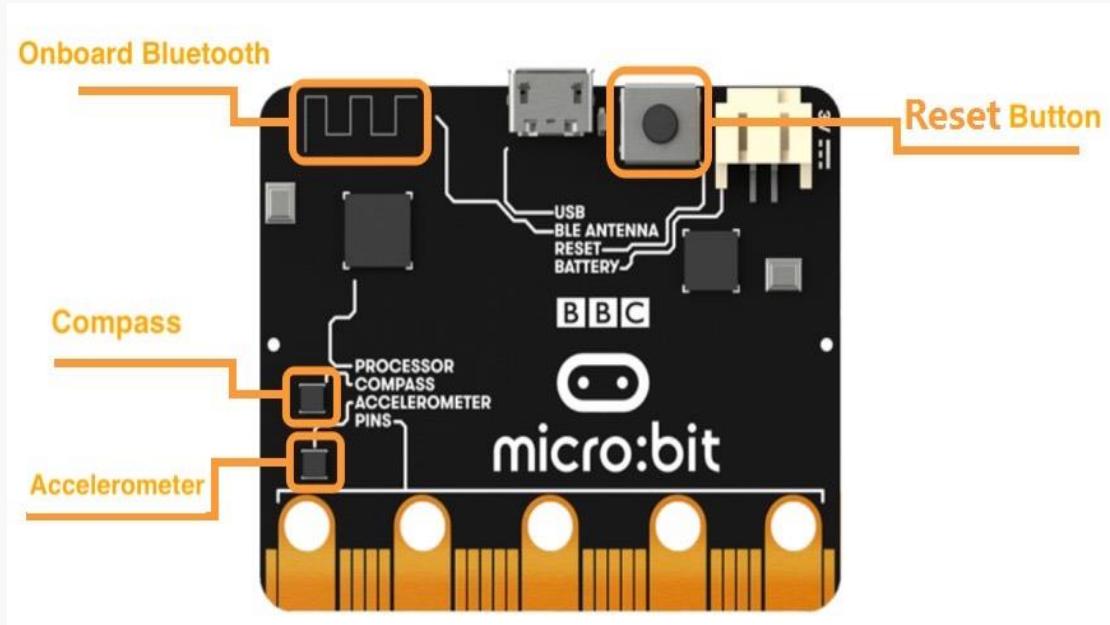
Conecte todos los componentes listados arriba a la placa de abajo, entonces podremos tener el cuerpo del Max:Bot .



Una breve introducción a Micro:bit

Que es Micro:bit?

Micro:bit es un microcontrolador de bolsillo fácil de usar, potente y económico diseñado para que niños y principiantes aprendan a programar, permitiéndoles introducir fácilmente ideas en juegos digitales de "hágalo usted mismo", proyectos interactivos y robótica.



Gracias a sus puertos de E/S externos y al soporte de hardware, Micro:bit es muy adecuado para el aprendizaje y desarrollo de diversos tipos de robots.

- Una pequeña placa similar al tamaño de una tarjeta de crédito (4cm x 5cm)
- Módulos integrados, como: acelerómetro, brújula y módulo Bluetooth® inteligente.
- Un microcontrolador de bolsillo

- Una matriz LED 5x5 (también soporta detección de luz)

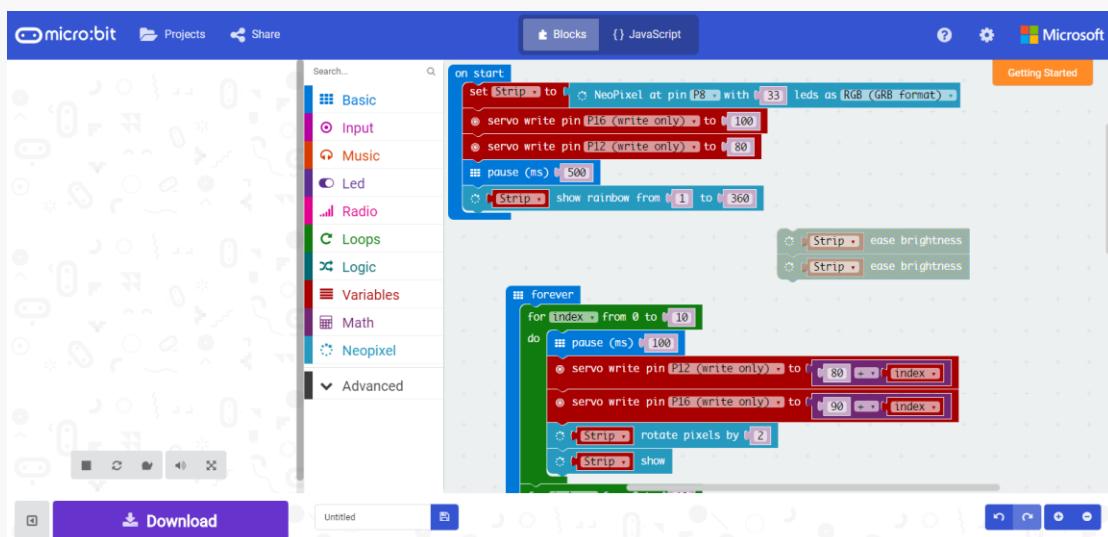
Sensores de luz, temperatura y otros dispositivos de detección comunes

Equipado con el procesador M0 de ARM, Micro:bit puede ejecutar la mayoría de las funciones fundamentales de un robot.

Cómo programar Micro:bit

Interfaz gráfica de programación MAKECODE

Para los principiantes, el mérito de Micro:bit reside en sus ricos lenguajes de programación e interfaces inclusivas.



Tomemos como ejemplo la plataforma de programación en línea makecode de Microsoft.

Los usuarios pueden arrastrar y colocar los módulos relevantes en la página web para leer los datos de los sensores y para procesar los datos rápidamente a través de ciertas funciones lógicas incluyendo bucle, juicio, etc. El desarrollo de micro:bit redujo enormemente el coste de programación para asegurar que la mayoría de los alumnos se centraran en el control de robots.

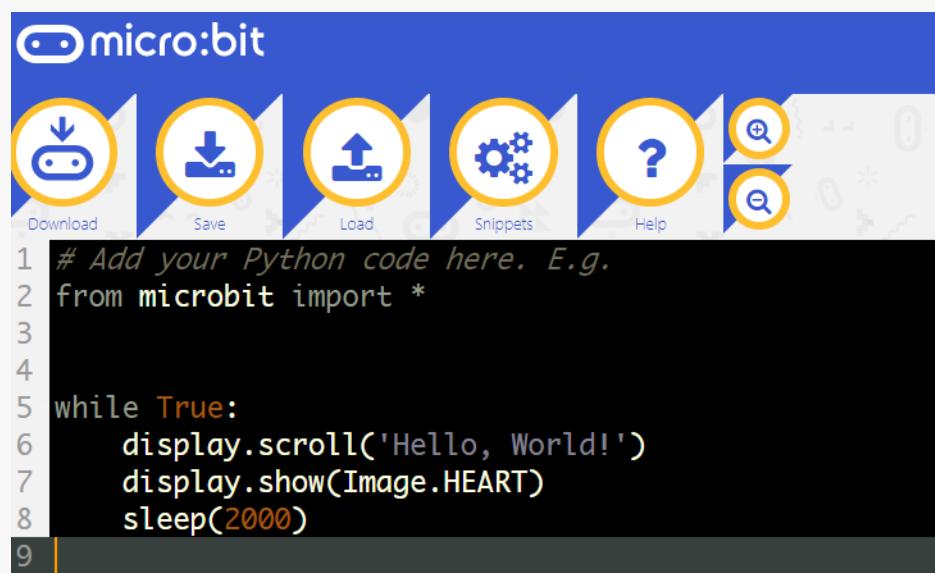
Haga clic en los siguientes enlaces para iniciar su viaje de programación en línea:

[Http://microbit.dfrobot.com.cn/index.html](http://microbit.dfrobot.com.cn/index.html) o,

[Https://makecode.microbit.org/#](https://makecode.microbit.org/#)

Programación con Micropython

Los usuarios avanzados pueden utilizar Python para programar Micro:bit, que es más potente y más utilizado. Micropython está desarrollado en base al lenguaje Python. Continúa usando el lenguaje Python pero es más poderoso en sus funciones.



En este tutorial, aprenderemos a usar la programación gráfica de makecode.

Luego continuaremos con la guía inicial de Python.

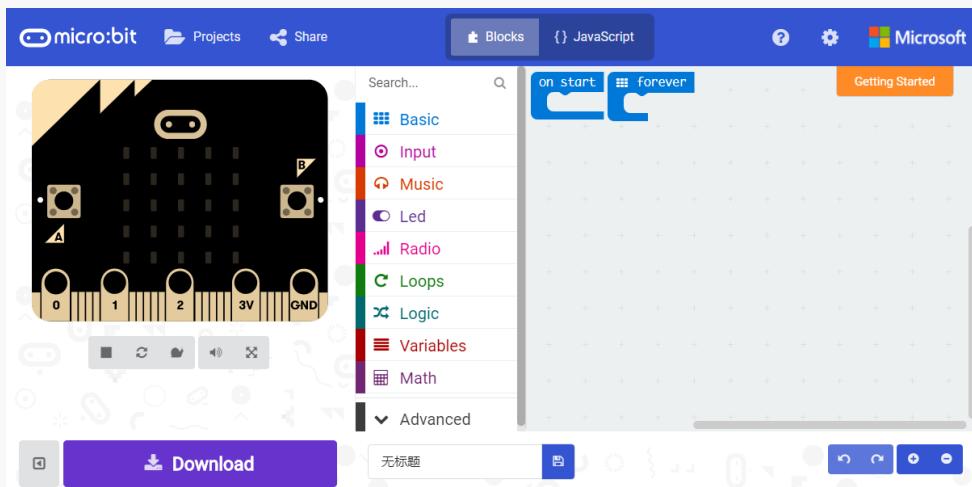
Cómo programar Micro:bit a través de Makecode?

Abrir MakeCode

Abra MakeCode desde el siguiente enlace:

[Https://makecode.microbit.org/#](https://makecode.microbit.org/#)

Así es como se ve el MakeCode cuando se abre por primera vez.



MakeCode siempre guardará el programa en su almacenamiento local. Así que cada vez que se cierra el navegador web, sólo tiene que recargar la página y el programa de edición aparecerá de nuevo en la ventana.

Interfaz de programación

Basado en funciones, la interfaz MakeCode incluye cinco partes:



Ventana de simulación: Simula el estado de funcionamiento del micro: bit. Durante el proceso de programación, siempre puede comprobar cómo se ve su programa a través de la ventana.

Área de funciones: Donde se pueden encontrar todos los bloques de función, incluyendo entrada, salida, bucle, lógica, etc.

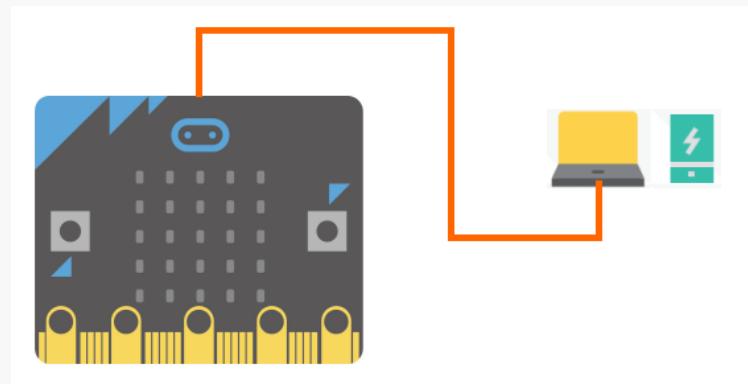
Área de programación: Arrastre los bloques desde el "Área de funciones", apílelos y construya su programa aquí.

Ajustes: Aquí puede seleccionar el idioma .

Área de nombres: Puedes nombrar tu proyecto aquí. El valor predeterminado es " Untitled".

Descargue el programa y cárguelo en micro:bit

1. Conecte el micro:bit al ordenador a través del cable USB.



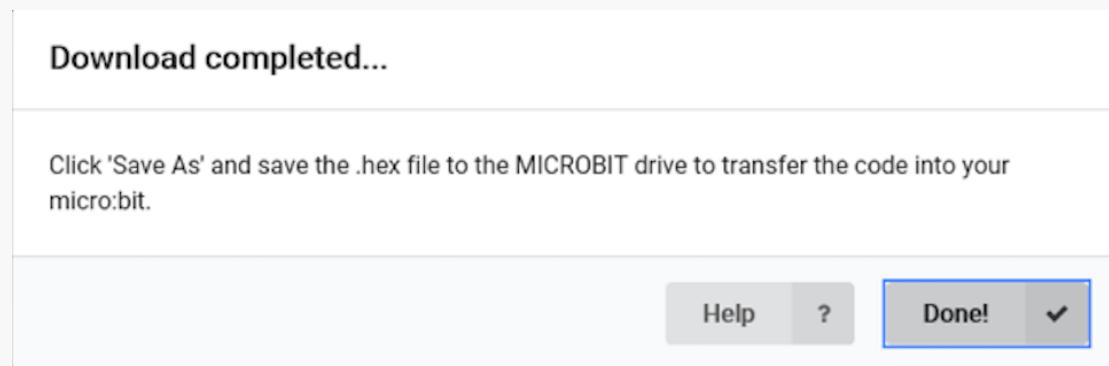
2. Antes de programar, debemos asegurarnos de que la placa base ha sido reconocida por el ordenador. Cuando el micro: bit está conectado, un directorio "MICROBIT" aparecerá en "Mi PC".



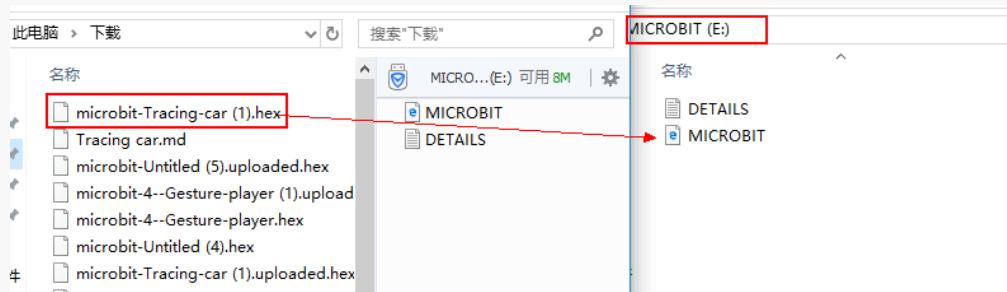
3. Haga clic en " download " y guarde el archivo ".hex " programado.



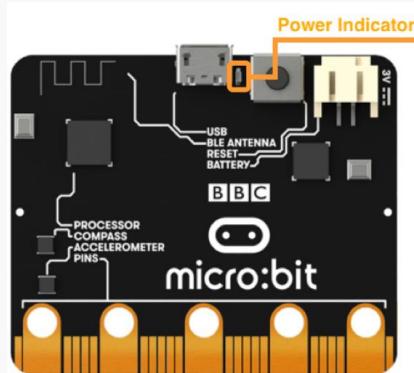
Luego verá el cuadro de diálogo que aparece a continuación:



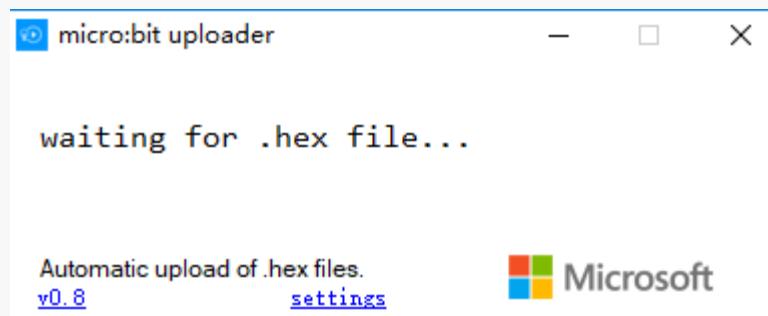
4. Busca el archivo ".hex" guardado y muévelo a "MICROBIT".



Durante el proceso de descarga, el indicador de encendido en la parte posterior del micro:bit parpadeará. Cuando termine, dejará de parpadear y seguirá funcionando.

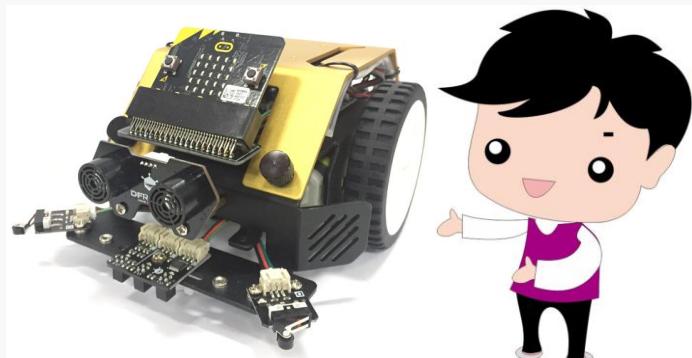


Consejos: Durante la programación, usted puede utilizar Micro:bit Uploader para cargar automáticamente el archivo ".hex" en "MICROBIT".



Capítulo 1: Iniciar Max:Bot

Hola, él es Max:Bot, tu compañero robot personal. Creo que ya has estado obnubilado con su brillante piel. Debes querer jugar con él ahora. Lo primero que queremos hacer es ponerlo en funcionamiento.



Objetivos:

1. Poner las ruedas en marcha.
2. Aprende a conducir el Max:Bot.
3. Recorrer en un patrón.

Componentes

Nombre	Imagen	Función
--------	--------	---------

Motor		Los motores accionan las ruedas de Max:Bot.
-------	---	---

1.1 Ir hacia adelante

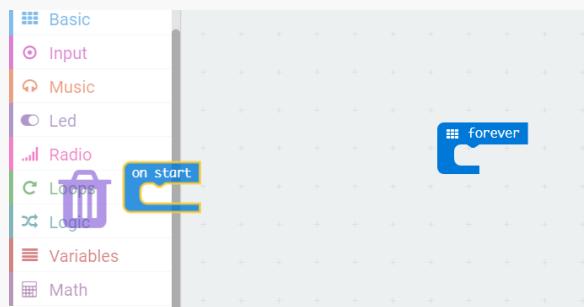
Información básica

Bloque de función	Imágenes	Funciones								
"servo write"		<p>La función de "servo write" se utiliza para controlar la velocidad y las direcciones de los motores.</p> <p>El valor 0 hace que las ruedas retrocedan;</p> <p>Valor 180 hace mover las ruedas hacia adelante;</p> <p>Valor 90 detiene el vehículo.</p> <table border="1" data-bbox="865 1426 1357 1560"> <tr> <td>Numbers</td> <td>0</td> <td>90</td> <td>180</td> </tr> <tr> <td>Speed and directions</td> <td>MAX</td> <td>0</td> <td>MAX</td> </tr> </table>	Numbers	0	90	180	Speed and directions	MAX	0	MAX
Numbers	0	90	180							
Speed and directions	MAX	0	MAX							
“forever”		Todas las funciones que se ejecutan repetidamente deben estar colocadas en él. El bucle "forever" es el bucle mayor.								

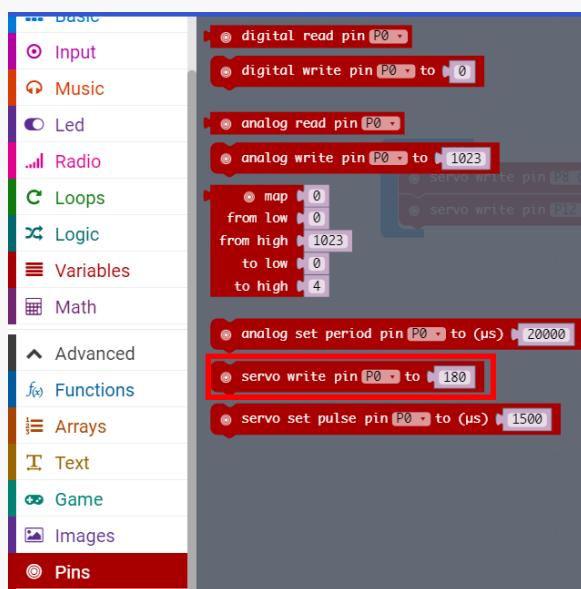
Programar

- (1) Abra makecode: <http://microbit.dfrobot.com.cn/index.html>

- (2) El módulo "on start" no se utilizará en esta parte. Así que tenemos que borrarlo. Sólo tenemos que arrastrarlo y soltarlo en el área funcional. A continuación, se borrará automáticamente. Este método también está disponible para todas las demás funciones.

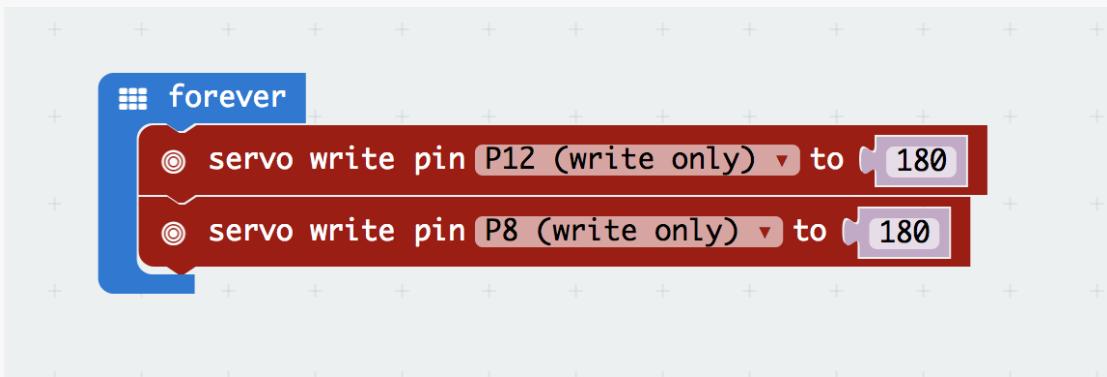


- (3) Busque la función "servo write" de "Pins" en el área de funciones. Esta función controla la velocidad de los motores.

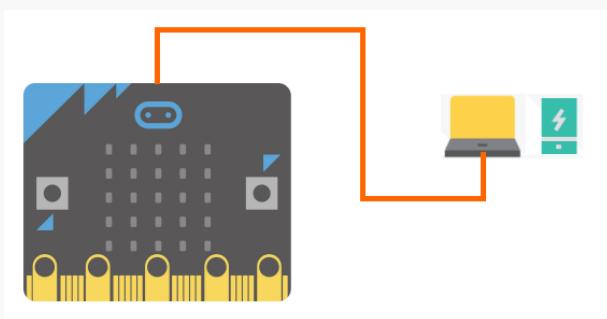


- (4) Ponga la función "servo write" en el bucle "forever".

Nota: la rueda izquierda debe estar conectada a P12, y la derecha a P8.



- (5) Ahora por favor cárguelo a Max:Bot . Wow! ¿Puedes ver eso? Max:Bot va hacia adelante!



Después de la descarga, debe quitar el cable USB del ordenador. Y el módulo de la batería debe estar conectado al Max:Bot.

Si Max:Bot sigue sin moverse, por favor, compruebe si lo ha encendido.

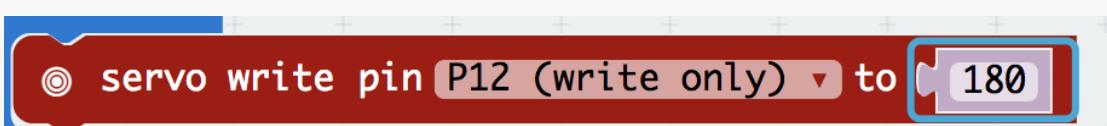


Ejercicios

Eres tú quien permite que Max:Bot siga adelante. Él realmente lo aprecia.

Pero también quiere saber cómo retroceder. ¿Lo ayudarás con esto?

Consejos: el número en la función "servo write" controla la velocidad, intente establecer diferentes valores.



1.2 Ir en Círculos

Información básica:

Bloque de función	Imágenes	Funciones
"function"		<p>Si se ejecutan acciones continuas repetidamente, podemos usar la "function" para reemplazar la serie de acciones. De esta manera se puede simplificar el bucle principal. Tenga en cuenta: La "function" no debe colocarse en el bucle "forever". Se trata de un subbucle independiente.</p>
"call function"		<p>Si las acciones programadas en la "function" deben ser utilizadas en el bucle "forever". A continuación, encontraremos y colocaremos la "call function" en el bucle "forever" para activar las acciones programadas en la "function". Tenga en cuenta: En el bucle "forever", la "call function" es un uso correspondiente de la "function".</p>

Por favor, tenga en cuenta que la "function" y la "call function" se encuentran en "Advanced" => "Functions". Pero ambos no están listados en él directamente. La manera de encontrarlos es así:

1. La manera de encontrar la "function": "Advanced" => "Functions" => "Make a Function". Haga clic en "Make a Function", nómbralo y haga clic en "OK". Entonces encontrará que este bloque de "function" aparece automáticamente en el área de programación.
2. Otra forma de encontrar la "call function": al mismo tiempo, el bloque "function" aparece en la zona de programación, el bloque "call function" aparece en la zona de "Functions" de la zona de funciones.

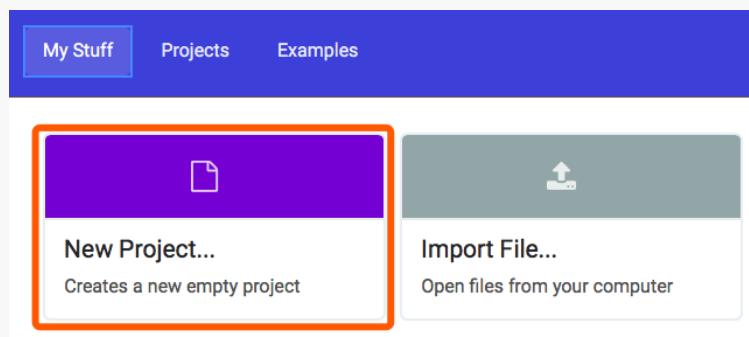
Por favor, recuerde que la "call function" sólo aparece cuando se crea la "function".

Programar

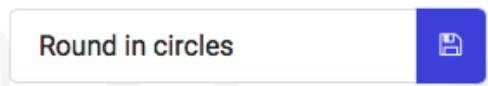
- (1) Comienza un nuevo proyecto y llámalo "Round in circles"



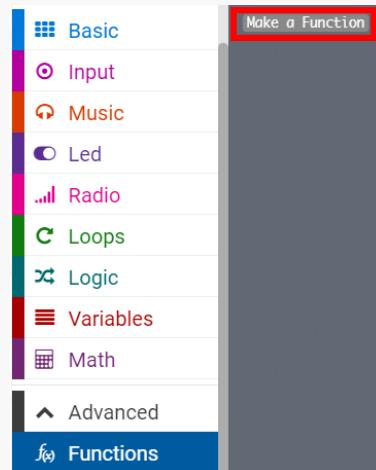
Por favor tenga en cuenta: Para evitar cualquier confusión, tenemos que empezar un nuevo proyecto, de lo contrario makecode lo guardaría directamente en el último archivo de proyecto.



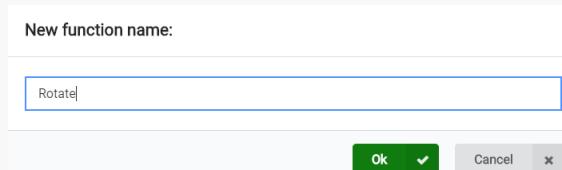
Nómbralo: Round in circles



- (2) Aquí necesitamos crear un sub-bucle. Va como: " Advanced"
=> "Functions" => "Make a Function".



Haga clic en " Make a Function " y nómbralo como: Rotar. A continuación, el bloque de " function " aparece automáticamente en la zona de programación. Y al mismo tiempo que se crea, la " call function " aparece en las " functions " del área de funciones.



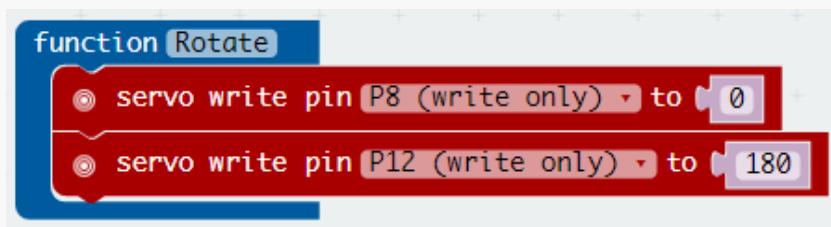
- (3) Cómo hacer que Maxbot gire?

Antes de programar, tendremos que familiarizarnos con el principio de "dirección diferencial".

En nuestra vida diaria, lo que permite a un coche girar es la diferente velocidad de rotación de las ruedas motrices. Tanto los embragues que están montados en los ejes del lado izquierdo y derecho como el freno se pueden utilizar para controlar las ruedas motrices para que giren a diferentes velocidades. En la actualidad, casi todos los vehículos de orugas, como los tanques, se rigen por este principio.

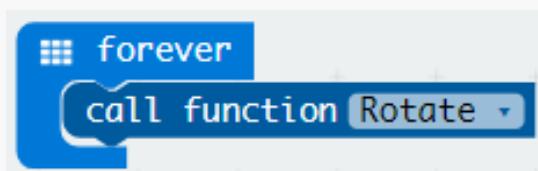
Por lo tanto, la clave del éxito es girar las dos ruedas a una velocidad diferente.

Colocar las funciones de "servo write" en el subbucle de "function".

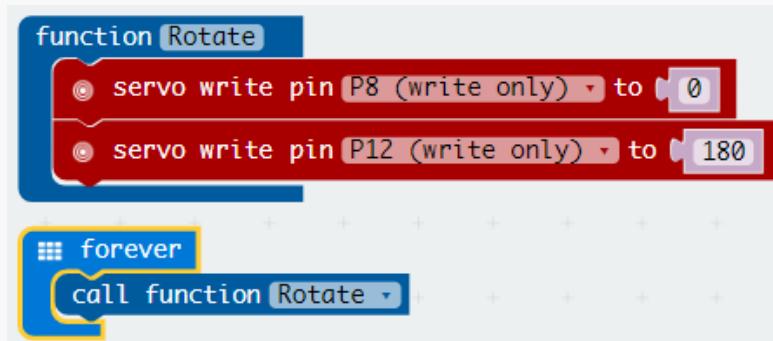


El "180" es la velocidad de rotación superior en el sentido de las agujas del reloj, mientras que el "0" es la velocidad de rotación superior en sentido contrario.

Arriba hemos creado el sublazo llamado "Rotate". Ahora tenemos que colocar la "call function" de "Functions" en el bucle "forever". Así, las acciones en la "function" serán ejecutadas por el bucle "forever".



(4) Combine todos los bloques de función listados arriba, tendremos el programa final como se muestra a continuación:



Descargue el programa a Max:Bot, luego girará en círculos en el sentido de las agujas del reloj.

No olvide quitar el cable USB y encender el Max:Bot

Ejercicios

Hemos aprendido a hacer que Max:Bot rote. ¿Puedes hacer que gire a una velocidad diferente?

Consejo: El valor de "servo write" también controla la velocidad del robot.

Establezca diferentes valores para obtener otros resultados.

◎ servo write pin [P12 (write only)] to [180]

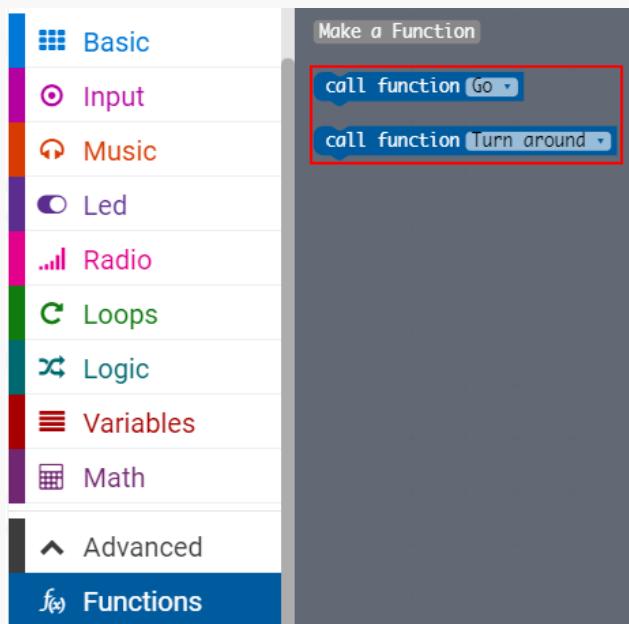
1.3 Formar un cuadrado

Información básica

Bloques de función	Imágenes	Funciones
"pause"	<pre> [forever v] pause (100 ms) end </pre>	La función "Pause" mantiene el mismo estado durante un cierto período de tiempo.

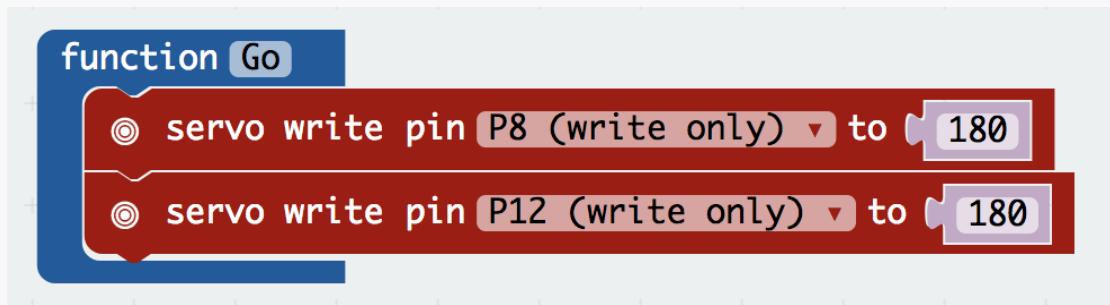
Programar

- (1) Comenzar un nuevo proyecto y nombrarlo "Round in squares"
- (2) Cree otros dos sub bucles y llámelos "Go" y "Turn Left". ¿Has recordado cómo crearlos? Lo revisaremos juntos: " Advanced"=>"Functions"=>" Make a Function".

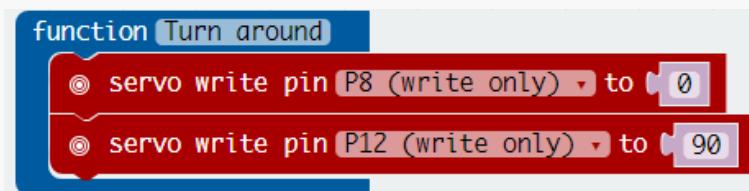


Como todos sabemos que, si uno quiere caminar y formar de cuadrado, primero debe caminar derecho por un rato (como 1 minuto), luego girar 90 grados y seguir caminando hacia adelante por otro minuto.... Estos movimientos se repetirán hasta que termine un cuadrado.

- (3) Por lo tanto, primero tendremos que hacer que siga recto. Como se ha mencionado en partes anteriores, la función de "servo write" puede utilizarse para controlar la velocidad de las ruedas y hacer que Max:Bot avance. Así que lo usaremos en esta parte. Ajuste ambos valores a "0" para mantener las dos ruedas girando a la misma velocidad. Y luego pon el "servo write" en "function(Go)".



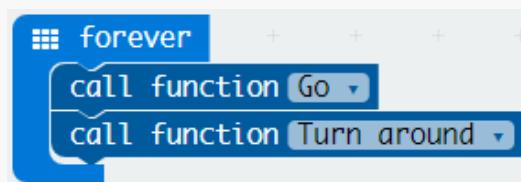
- (4) También hemos aprendido el principio de "dirección diferencial" para hacer girar el Max:Bot. Pero aquí, tenemos que ajustar los valores para que gire sólo 90 grados. Poner P8 (rueda izquierda) como "0" y P12 (rueda derecha) como "90". El valor "90" significa que la rueda debe permanecer quieta.



En comparación con la "function(Rotate)", los valores "0" y "90" ajustados en la "function(Turn around)" significan:

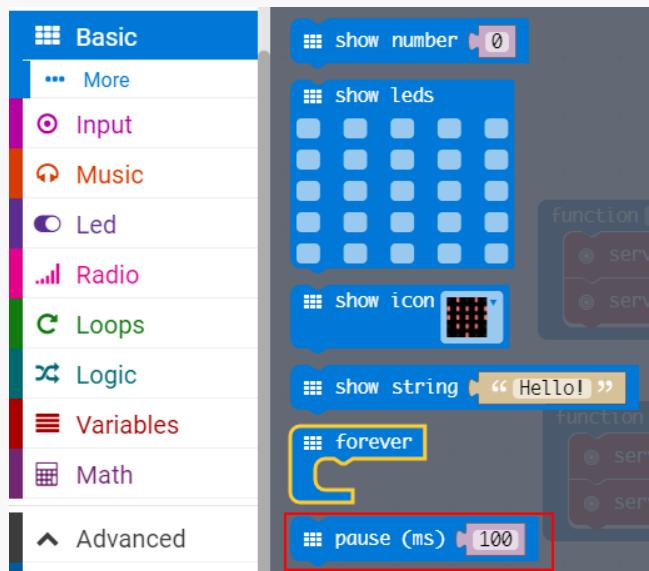
El "90" en P12 significa que la rueda derecha gira 0 grados. El "0" en P8 significa que la rueda izquierda gira a la velocidad máxima en sentido contrario a las agujas del reloj. Así, Max:Bot gira a la derecha.

- (5) Coloque tanto la "call function(Go)" como la "call function (Turn around)" en el bucle "forever". El programa final será el siguiente:

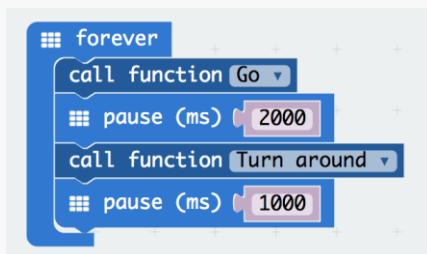


¿Ahora el Max:Bot hace el cuadrado? Supongo que no. Tenemos que ser más pacientes. Sin dolor no hay ganancia.

Para asegurar que Max:Bot realice la tarea, tenemos que darle tiempo suficiente para avanzar y girar. Aquí necesitamos la función "pause" para ajustar el tiempo. Puede encontrarla en "Basic" del área funcional. Colócalo en el bucle "forever".

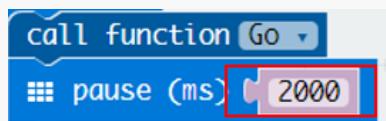


El programa es como se muestra a continuación:

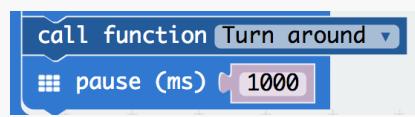


Por favor, tenga en cuenta:

- a. El "2000" determina la longitud lateral del cuadrado.

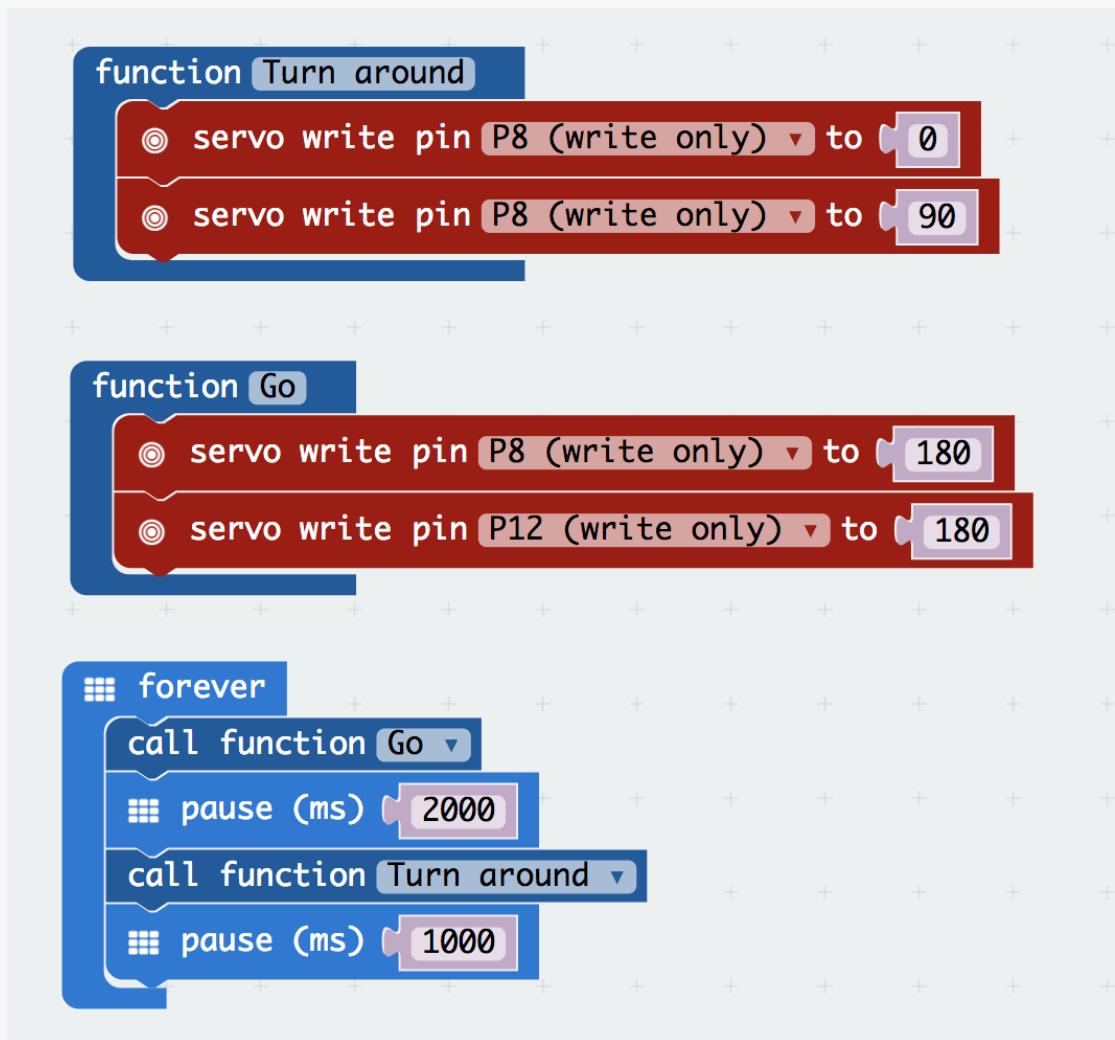


- b. El "1000" determina el ángulo de giro.



Diferentes baterías producen diferentes voltajes. Por lo tanto, los "2000" y "1000" aquí sólo se utilizan como referencia. Hay que intentar ajustarlos en función de la situación real. Recuerden lo que acabamos de decir: Sin dolor no hay ganancia. ¡Sólo inténtalo!

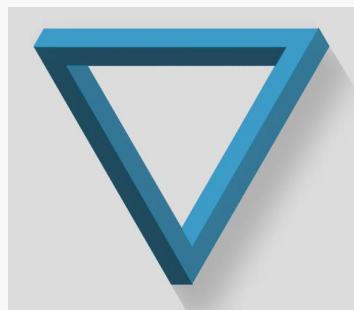
- (6) Combine todas las funciones listadas arriba, el programa final es el siguiente. Asegúrate de que has hecho cada paso correctamente y luego descárgalo a Max:bot.



¡Los has logrado!

Ejercicios

Una vez que lo hayas logrado te será fácil hacer lo mismo pero en forma de triángulo, haz la prueba!



Capítulo 2: Max:Bot Inteligente

Puede que ahora tengas un sentido de realización porque Max:Bot puede ir a donde quieras que vaya. Pero te has sentido un poco más desanimado porque siempre hay algo que lo bloquea, como un escritorio o un armario. No te preocupes, en este capítulo enseñaremos a Max:Bot cómo evitar estos obstáculos.



Objetivos:

1. ¿Cómo usar el sensor digital de colisión?
2. ¿Cómo utilizar la función "if then"?
3. Programación asistida por diagramas

Digital Modules

Módulos digitales	Imágenes	Funciones
-------------------	----------	-----------

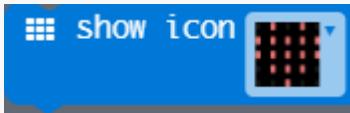
Sensor digital de colisión		El sensor digital de colisión puede, como un botón, detectar obstáculos. Como muestra la segunda imagen a la izquierda, hay dos sensores digitales de colisión (marcados en rojo) conectados al Max:Bot por separado.
----------------------------	---	---

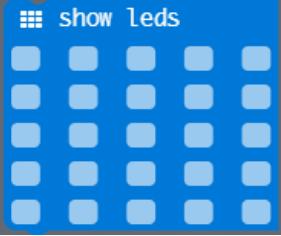
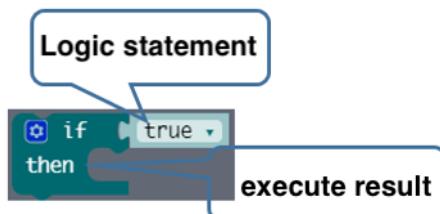
2.1 El Max:Bot "tímido"

A veces, al ser tocado, Max:Bot se vuelve lindo y tímido. Sí, puede sentirte.



Información básica

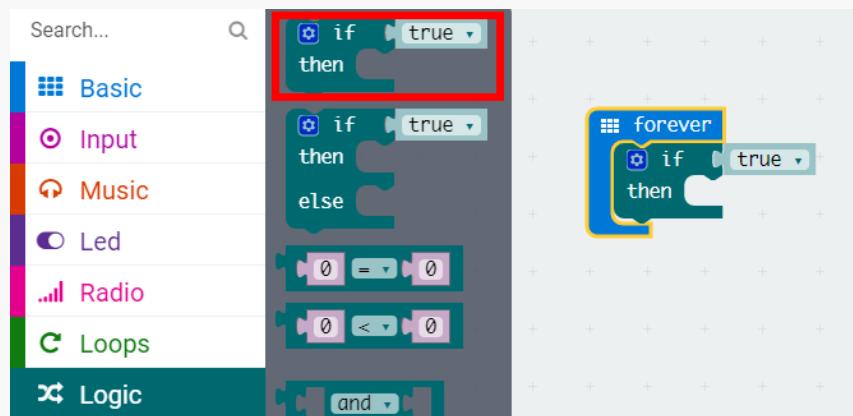
Bloque de función	Imágenes	Funciones
"show icon"		En la función " show icon " se pueden seleccionar muchos íconos. Puedes usarlo para elegir tus íconos favoritos.

"show leds"		Hay 25 puntos celestes dentro del bloque, puede hacer clic en cada uno de ellos para editar un patrón. Estos puntos representan los 25 leds del Micro:bit.
"if then"		El "if then" se utiliza para juzgar qué función se ejecutará. Lo que significa que si se cumple la sentencia lógica a la derecha de "if", debe ejecutarse la frase de código a la derecha de "then", mientras que la siguiente función debe ejecutarse si no se cumple la sentencia lógica.
"=="		El operador "==" se utiliza para controlar el rango de un valor. Por ejemplo: Como podemos ver en la imagen de la izquierda, hay dos puntos rojos claros. Si colocamos "digital read(P1)" en el punto izquierdo y cambiamos el valor del punto derecho por "1", significa que el valor de P1 es igual a 1. Dentro del operador "==" hay muchas otras alternativas, como ">", "<", etc.
"digital read"		"Digital read" se utiliza para leer el estado del pin. Hay dos estados: un alto voltaje (presentado como 1) y

		un bajo voltaje (presentado como 0).
--	--	--------------------------------------

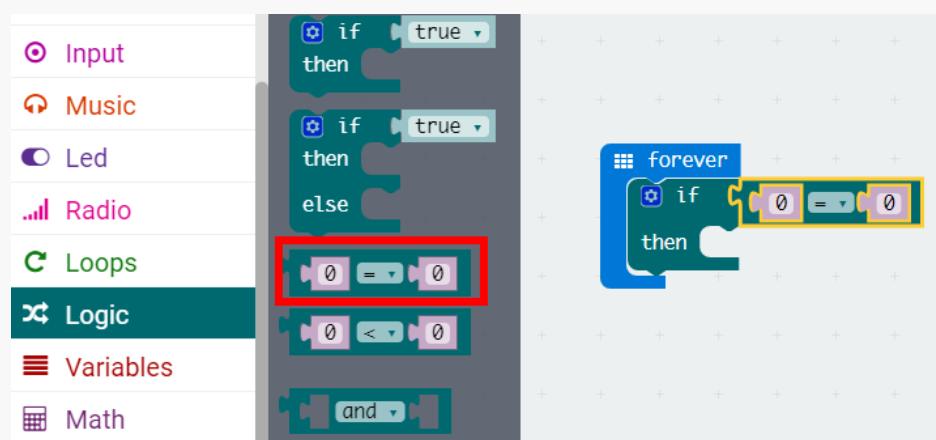
Programar

- (1) Comenzar un nuevo proyecto y nombrarlo como "el Max:Bot tímido"
- (2) Encuentre la función "if then" de "Logic" en el área de funciones y colóquela en el bucle "forever".

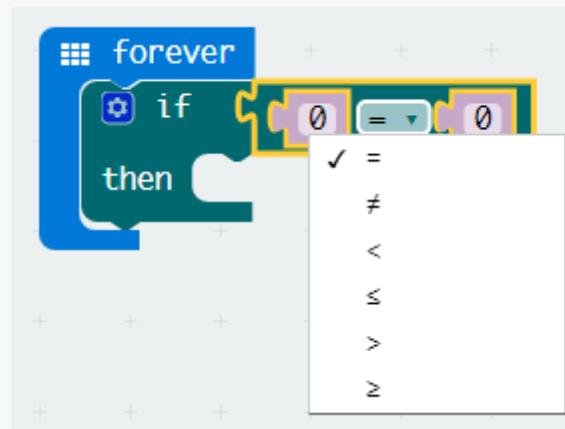


Si se cumple la sentencia lógica a la derecha de "if", debe ejecutarse la función a la derecha de "then", mientras que la siguiente función debe ejecutarse si no se cumple la sentencia lógica.

- (3) El operador " $=$ " se encuentra en la " Logic ".



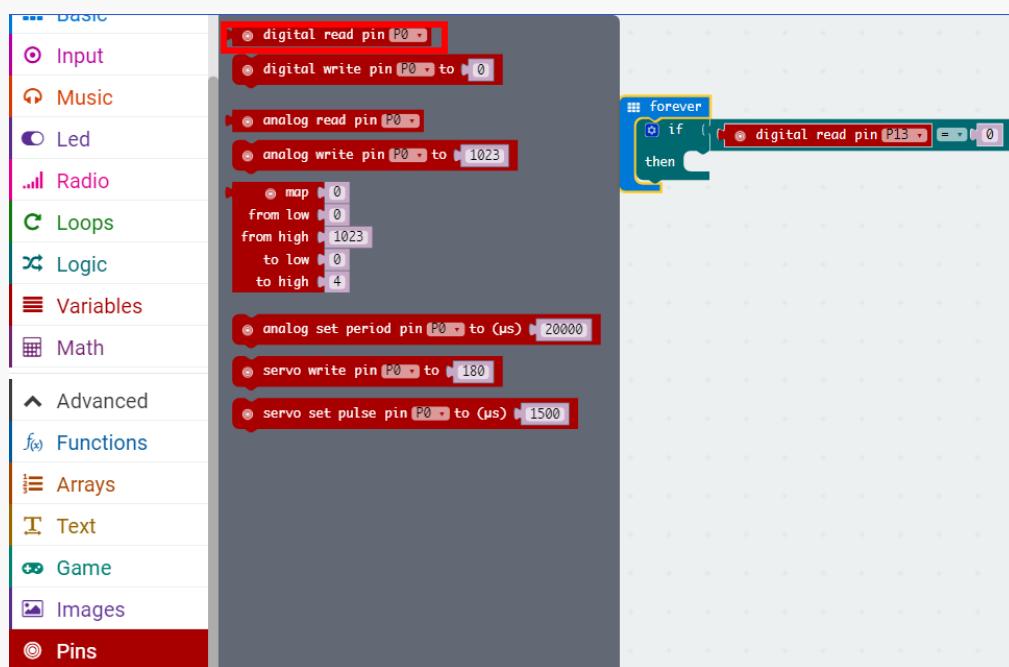
Además de " $=$ ", hay muchos otros operadores, como " $>$ ", " $<$ ", " \neq ", " \leq ", " \geq ", que se pueden seleccionar en el menú desplegable.



(4) ¿Cómo controlar el sensor digital de colisión?

Siempre que Max:Bot sea tocado por otros, se convertirá inmediatamente en un robot tímido. ¿Por qué se vuelve tan sensible? Porque hemos conectado sensores digitales de colisión.

En esta parte se aplicará la función de " digital read " para determinar si los sensores digitales de colisión están en contacto o no. Conecte uno de los sensores digitales de colisión al Pin13.

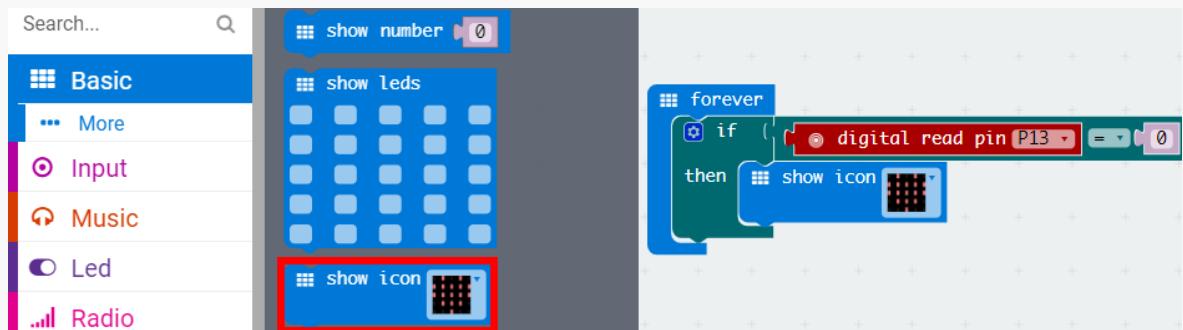


¿Tienes idea de la respuesta que va a dar Max:Bot? La siguiente tabla se puede utilizar como referencia.

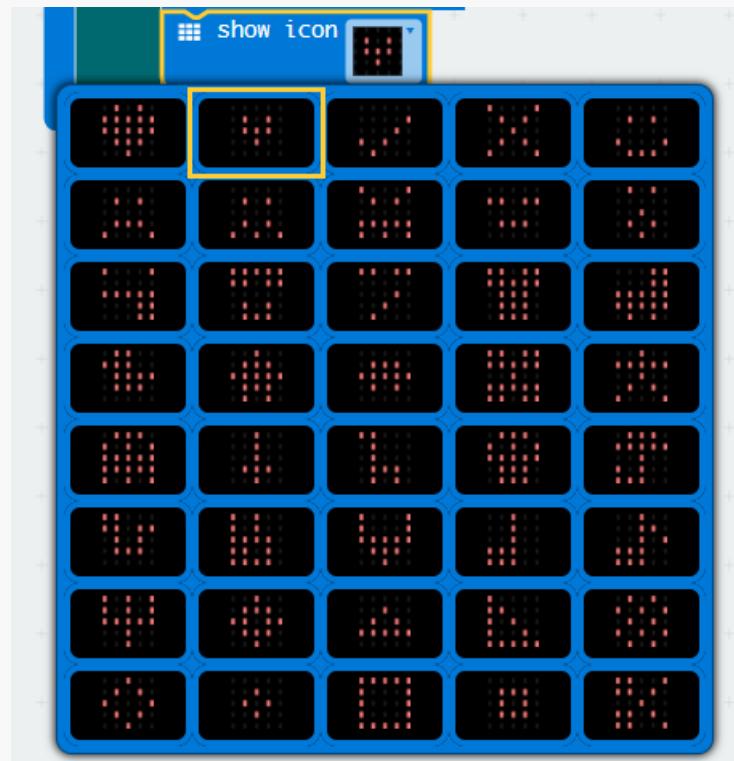
Status	Values
Being touched (the indicator turns on)	0
Untouched (the indicator turns off)	1

(5) Haz latir el corazón

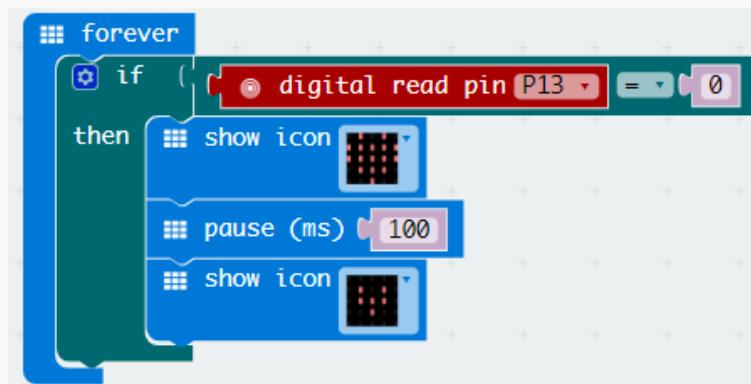
Una vez que ha sido tocado por otros, Max:Bot inmediatamente se volverá tímido y se acelerará su corazón. ¿Pero cómo podemos hacer que su corazón late realmente? La función "show icon" realiza el trabajo. Puede encontrarla en "Basic" del área funcional.



Como se puede ver en la imagen de abajo, hay un microtriángulo en la parte superior derecha del "show icon". Haga clic en él y elija el icono del corazón más pequeño.



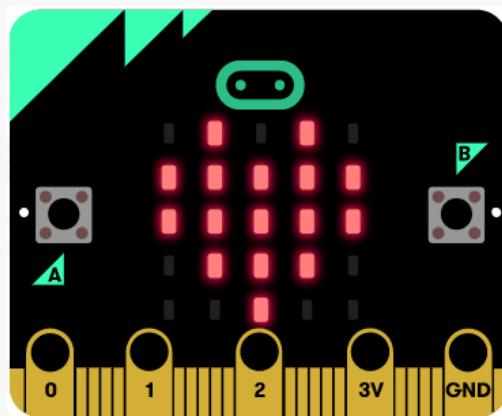
Para hacerla más parecida a un latido real, necesitamos añadir la función "pause" y otro ícono de corazón más grande. Al juntar todas las funciones listadas arriba, tendremos el siguiente programa.



- (6) Descargue el programa a Max:Bot.

Ahora cuando toques Max:Bot, te mostrará el latido de su corazón.

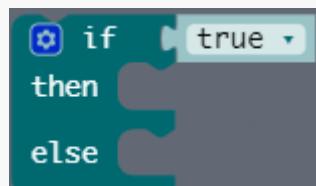
¡Qué lindo robot!



Ejercicios

No hace falta decir que Max:Bot es un compañero muy agradable. Ya hemos creado su latido. Ahora queremos hacerle una cara sonriente cuando esté intacto.

Usaremos la función "if else" para hacerle sonreír. También puede encontrarla en "Logic" del área funcional.



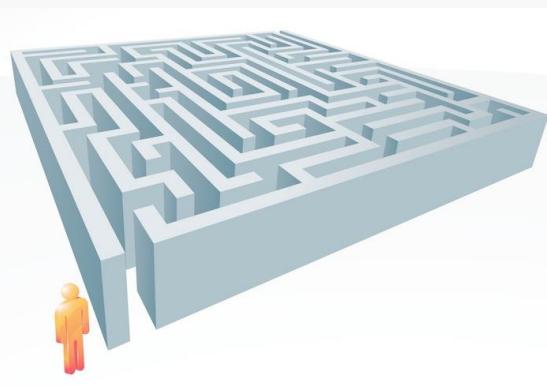
En comparación con el "if then", hay un "else" adicional en el "if else". ¿Qué es esto?

Como hemos aprendido en la parte anterior que, en la función "if then", si se cumple la sentencia lógica a la derecha de "if", se debe ejecutar la función a la derecha de "then", mientras que la función siguiente se debe ejecutar si no se cumple la sentencia lógica.

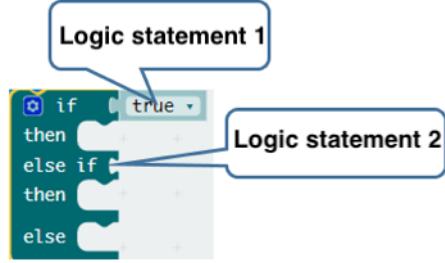
En la función "if else", si no se cumple la sentencia lógica a la derecha de "if", debe ejecutarse la sentencia a la derecha de "else".

2.2 Escapar del laberinto

Como el sensor digital de colisión convierte a Max:Bot en un robot sensorial, ¿qué tal si lo colocamos en un laberinto? ¿Encontrará él mismo una salida?



Información básica

Bloques de función	Imágenes	Funciones
Múltiple "if else"		Dado que la función "if else" no es nueva para nosotros, la función "if else" múltiple no es más que una función "if else" superpuesta. Sin embargo, el bloque de función múltiple "if else" no es un bloque de función listo para usar, sino que evoluciona a partir del "if else" y cumple la misma función. La forma de construir la función "if else" múltiple es la siguiente: Como podemos ver en la imagen de la izquierda, en la esquina superior izquierda de la función "else if" hay un ícono

azul con forma de engranaje. Haga clic en él y arrastre la frase "else if" desde el lado izquierdo a la parte central entre el "if else". Entonces encontrará que la función "if else" ha evolucionado hasta convertirse en la función "if else" múltiple. Seguramente puede repetir la acción anterior para hacer una función "si no" múltiple más larga.

Pero aquí, para ilustrar su función, pondremos como ejemplo la función "if else" múltiple arrastrada una vez:

Como podemos ver en la imagen de la izquierda, en la función múltiple "if else" hay la sentencia lógica 1 y la sentencia lógica

2. Lo que significa:

Habrá un juicio por la información recibida. Si se cumple la sentencia lógica 1, se debe ejecutar la función a la derecha del primer "then".

Mientras que se cumple la sentencia lógica 2, se debe

		ejecutar la función a la derecha del segundo "then".
--	--	--

Programa

- (1) Comenzar un nuevo programa y nombrarlo "escapar del laberinto"

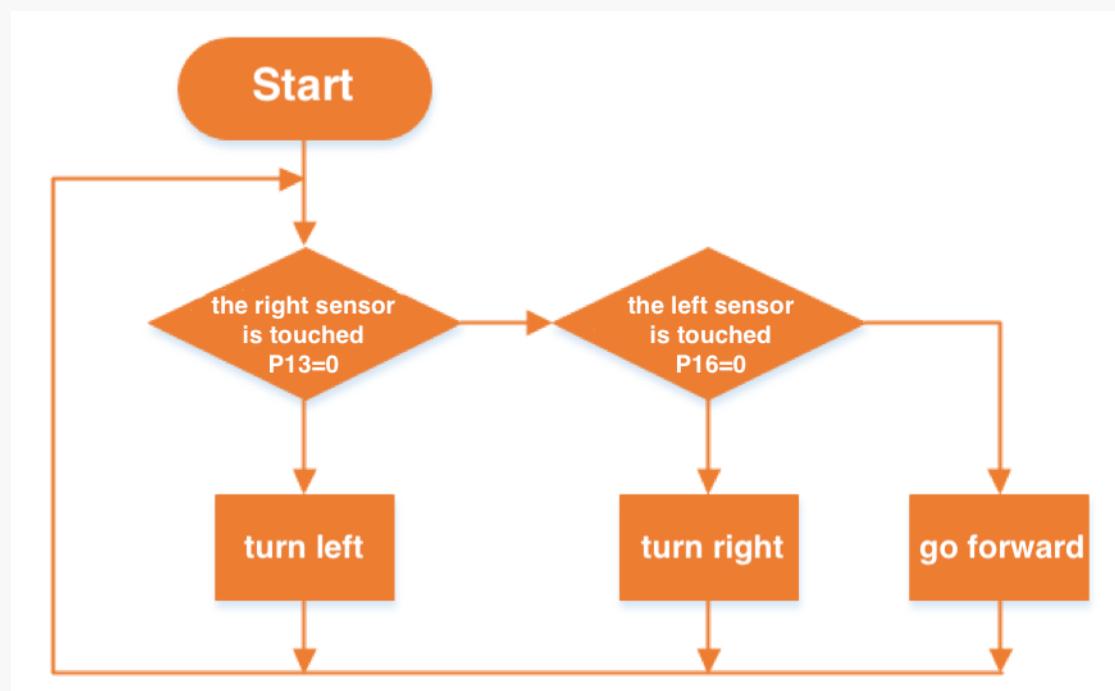
Ayuda a Max:Bot a escapar del laberinto

Antes de programar, primero debemos saber qué haría Max:Bot para evitar obstáculos.

Si se toca el sensor de colisión izquierdo, el Max:Bot girará a la derecha;

Si se toca el sensor de colisión derecho, el Max:Bot girará a la izquierda.

El siguiente diagrama puede ser tomado como guía:

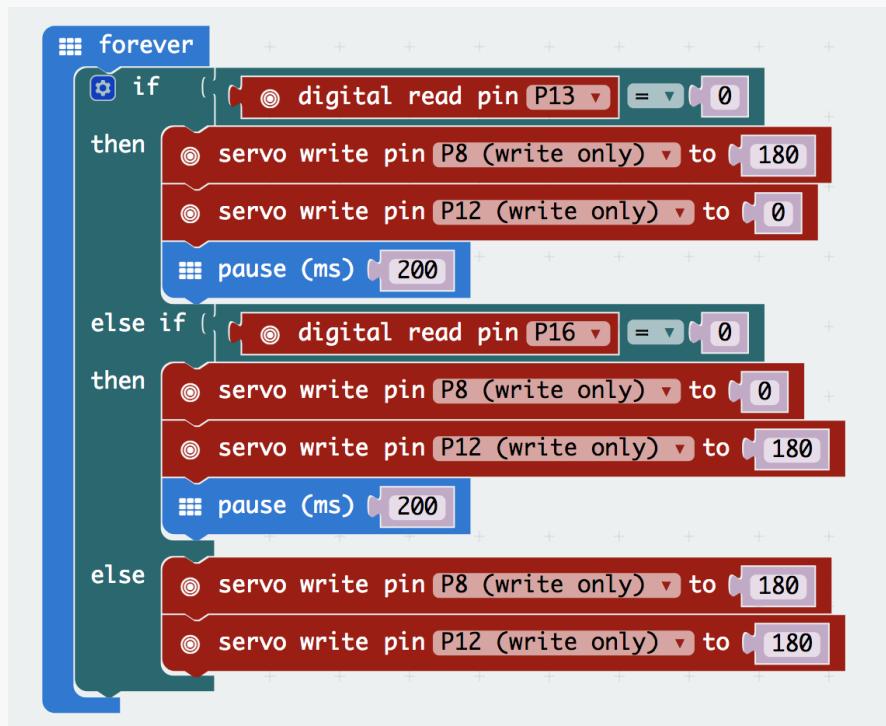




Con la ayuda del diagrama, por favor intente programarlo.

- (2) Algunos de los programas aprendidos anteriormente se combinarán en esta parte para ayudar a Max:Bot a escapar del laberinto. Tales como: girar a la izquierda, girar a la derecha, seguir recto, etc. Al juntar todos los programas mencionados anteriormente, tenemos el programa final como se muestra a continuación:

Por favor, tenga en cuenta: Las diferentes baterías emiten diferentes voltajes. Por lo tanto, el valor que aparece a continuación en la función "pause" sólo se utiliza como referencia. Tienes que intentar ajustarlos en función de la situación real.

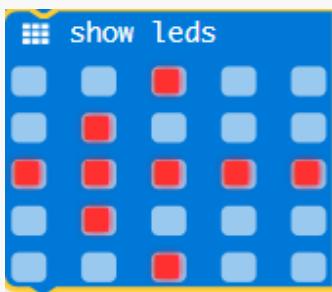


Ejercicios

Max:Bot, un aprendiz muy rápido, ahora ha dado otro paso adelante!

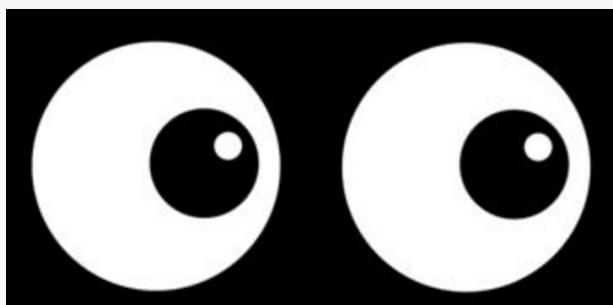
¿Recuerdas el "latido" que hicimos para él? ¿Qué tal si nos muestra su dirección cuando esté listo para girar?

Consejos: Es posible que necesite la ayuda de la función "show leds".



Capítulo 3: Sensor de Luz

En capítulos anteriores vimos como Max:Bot es como una criatura viva con ojos, capaz de reconocer y evitar obstáculos, ¿qué más podemos explorar sobre su sistema sensorial?



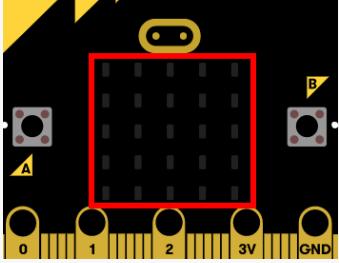
Te sorprenderá que Max:Bot pueda incluso sentir el brillo de la luz. En este capítulo, trabajaremos juntos para aprender cómo lograr esto.

Objetivos

1. ¿Cómo utilizar el sensor de luz?
2. ¿Cómo utilizar el módulo de visualización digital?

Programación asistida por diagramas

Módulos digitales

Módulos digitales	Imágenes	Funciones
Sensor de luz		La matriz LED 5x5 de Micro:bit se utiliza para mostrar patrones y detectar luz. Integra componentes sensibles a la luz que pueden transferir el brillo de la luz a los valores de salida.

3.1 Aventura al brillo

Cuando se está en un entorno con luces fuertes, la gente suele entrecerrar los ojos instintivamente para protegerse. Al estar en un entorno con poca luz, no somos capaces de distinguir objetos. Aquí vienen las preguntas: es lo que influye en el brillo de la luz. ¿Cómo los colores de las luces influyen en su brillo?



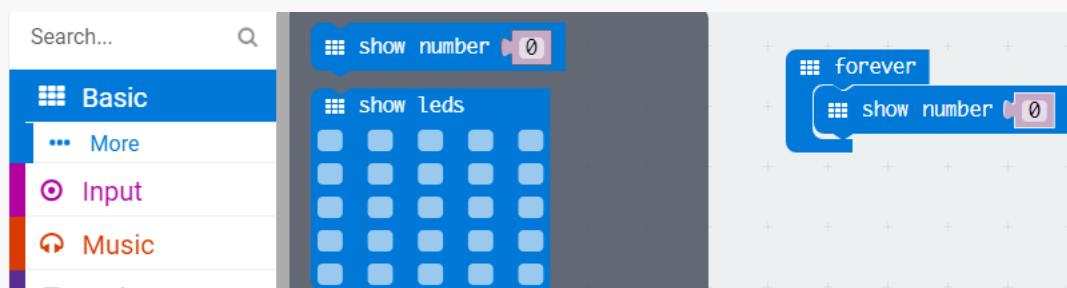
Para aclarar lo anterior, deberíamos pedir ayuda a Max:Bot. Como dice el refrán: aprende a caminar antes de correr. Así que primero necesitaremos usar Micro:bit para transferir el brillo a los valores de salida.

Información básica

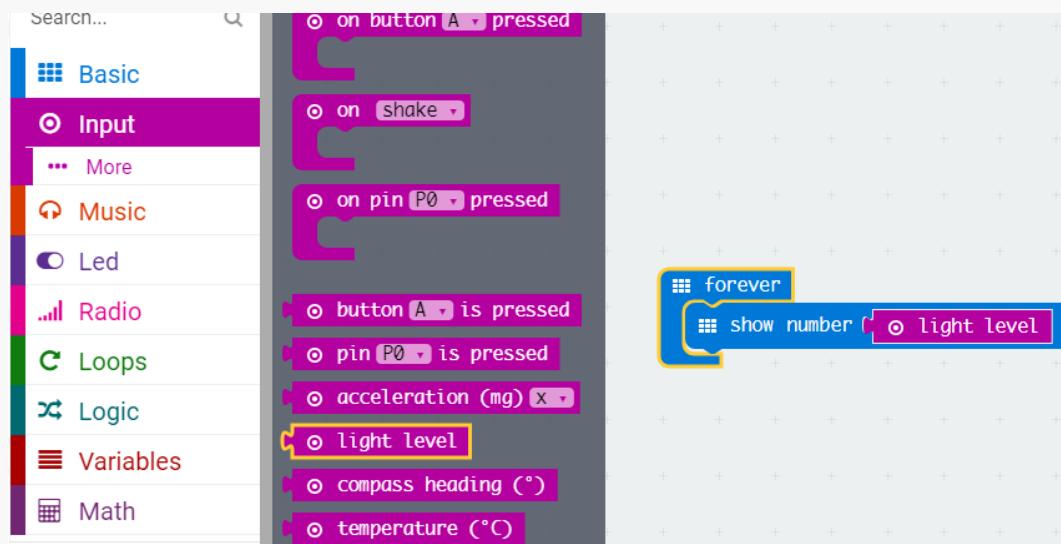
Bloques de función	Imágenes	Funciones
"light level"		Podemos utilizar la función " light level " para leer el valor de luminosidad. Va de 0 a 255, lo que significa que el brillo es de débil a fuerte.
"show number"		La función " show number " se utiliza para mostrar el número recibido.

Programa

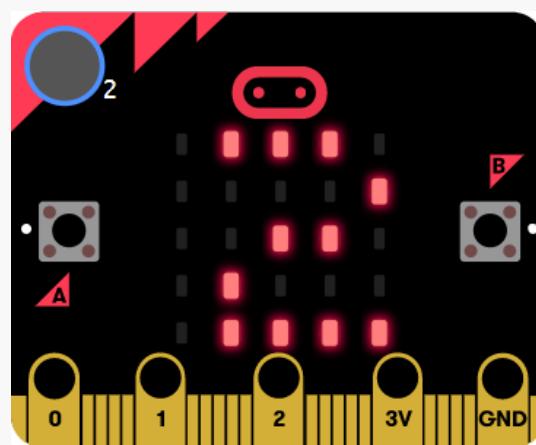
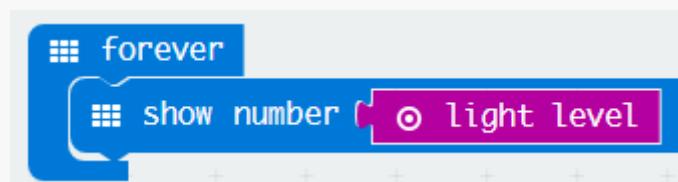
- (1) Comienza un nuevo programa y llámalo "aventura al brillo".
- (2) Busque la función " show number " de "Basic" y colóquela en el bucle " forever ". El número inicial es "0".



- (3) Como hemos mencionado anteriormente, la función " show number " puede ser utilizada para mostrar el número que ha sido leído por la función " light level ". Por lo tanto, en este paso, necesitamos primero encontrar la función " light level " de la " Input " en el área de función, y luego colocarla en la función " show number ". Colóquelos en el bucle "forever".



- (3) Descargue el programa a Max:Bot. El brillo detectado se mostrará inmediatamente en la matriz de LEDs 5x5 de Micro:bit, entonces podremos ver el brillo con nuestros propios ojos.



Ejercicios

Hay diferentes habitaciones con diferentes luces en su casa, ¿por qué no ir y detectar su brillo? Puede registrarlos en la siguiente tabla.

Entorno	Brillo

2.2 El Robot Polilla

¿Has visto alguna vez, por la noche, cientos de polillas volando contra los faroles? Esto se debe a que la polilla es una especie de insecto fotostático. Suena muy interesante, ¿verdad? Como hemos aprendido el principio de cómo Max:Bot puede sentir el brillo, ahora podemos aplicarlo en un robot polilla.



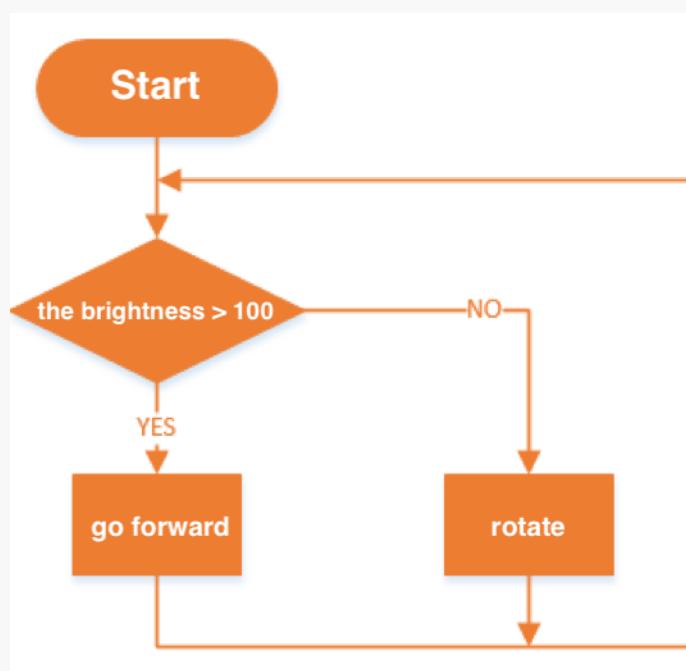
Información básica

Bloques de función	Imágenes	Funciones
"if else"		<p>Como podemos ver en la imagen de la izquierda que en la función "if else" se encuentra la sentencia lógica a la derecha de "if", el resultado de la ejecución 1 a la derecha de "then" y el resultado de la ejecución 2 a la derecha de "else". Lo que significa:</p>

		Si se cumple la sentencia lógica, el resultado de ejecución 1 debe ejecutarse, mientras que el resultado de ejecución 2 debe ejecutarse si no se cumple.
--	--	--

Programa

- (1) Inicie un nuevo programa y llámelo "el robot polilla"
- (2) Cuando el brillo alcanza un cierto rango de valor, el robot polilla avanzará hacia la luz. Si no se alcanza el valor, permanece y gira. El siguiente diagrama puede proporcionar asistencia a su programa.

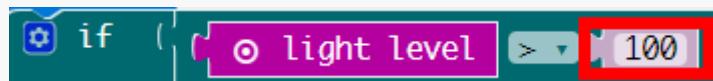


- (3) En esta parte, el operador "=" se utilizará para juzgar si el brillo alcanza o no el valor establecido. Y la función de "servo write" se utiliza para hacer que el robot avance o gire. La función "if else" se utiliza aquí para juzgar cuándo debe avanzar y cuándo debe permanecer y

girar. Al juntar todas las funciones listadas arriba, tendremos el programa final de la siguiente manera:



Tenga en cuenta: el valor de luminosidad no debe ser ni demasiado grande ni demasiado pequeño, sino que debe mantenerse dentro de un rango razonable.



Si el valor es demasiado grande, el robot polilla no avanzará hasta que se dé una luz bastante fuerte.

Si el valor es demasiado pequeño, es probable que el robot polilla no deje de avanzar.

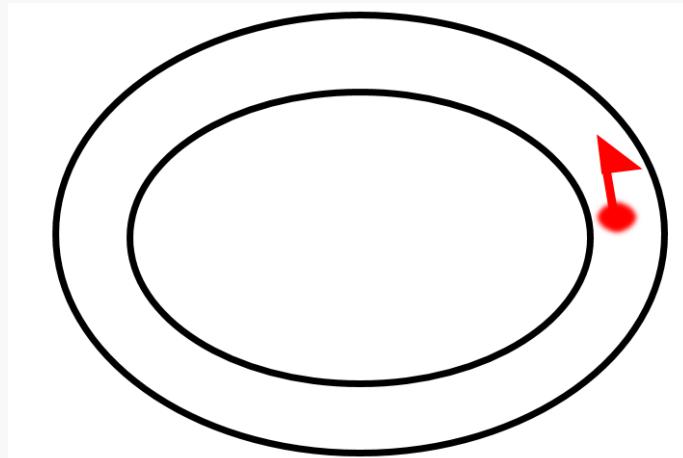
Así que tienes que pensarlo dos veces y establecer un valor razonable.

- (4) Descargue el programa final a Max:Bot . Ahora puedes jugar un juego de detección de luz con este robot polilla.

Ejercicios

Hagamos una carrera para el jugador del robot polilla.

Puedes invitar a tus amigos a unirse a la carrera. Cada uno de ustedes usará la misma linterna eléctrica para guiar al robot de la polilla hacia adelante. El ganador será la persona que complete una vuelta en el menor tiempo posible. La línea de salida es una bandera roja.



Consejos: Mantener su velocidad dentro de un rango razonable es la clave del éxito.

Capítulo 4 Ultrasonido, el arma secreta

En el capítulo anterior, hemos enseñado a Max:Bot a utilizar sus "ojos" (los sensores digitales de colisión y el sensor de luz) para evitar obstáculos y percibir el brillo de la luz. Sin embargo, ¿alguna vez has notado que hay otro par de ojos reales en su cara. ¿Qué pueden hacer?

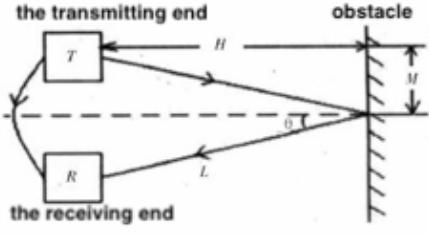


Son ojos con super poderes. Pero tienes que prometerme que lo mantendremos en secreto. Este par de ojos son el arma secreta de Max:Bot. Aprendámoslos juntos.

Objetivos:

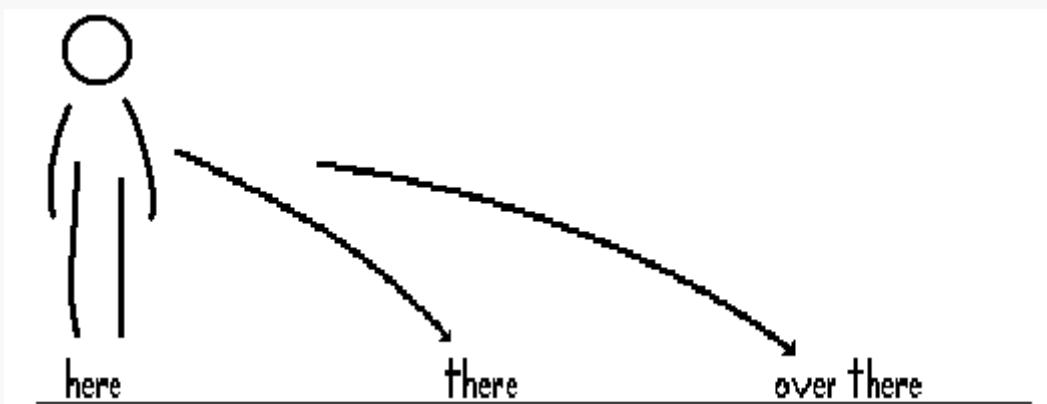
1. ¿Cómo funciona el sensor de ultrasonido?
2. ¿Cómo añadir un paquete?
3. Programación asistida por diagramas

Módulos Digitales

Módulos Digitales	Imágenes	Funciones
Sensor de Ultrasonido		 <p>El sensor de ultrasonido puede enviar ultrasonido a través del extremo de transmisión. Una vez encontrados los obstáculos, el ultrasonido enviado será rebotado inmediatamente y recibido por el extremo receptor. Así Max:Bot puede detectar obstáculos.</p>

4.1 Medidas de distancia por ultrasonidos

Lo que puede sorprenderte de nuevo es que Max:Bot es más una herramienta fiable que un buen compañero. Una vez que tenemos Max:Bot, todas las demás herramientas de medición, como reglas, cintas, etc., pueden ser desechadas. En resumen, Max:Bot puede medir la distancia con facilidad.



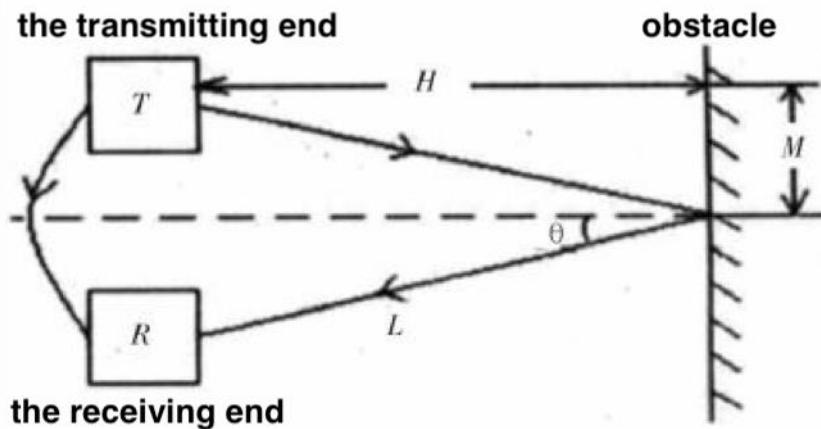
¿Qué es el ultrasonido?

Como sabemos, al vibrar, los objetos producen ondas sonoras. Algunos de ellos pueden ser escuchados por nuestros oídos, mientras que otros no. Los científicos nombraron los tiempos de vibración por segundo como la frecuencia del sonido con su unidad llamada Hertz. Casi todos los humanos son capaces de oír las frecuencias de sonido que van de 20 a 20000 Hertz.



Sin embargo, la mayoría de nosotros no es capaz de oír las frecuencias de sonido que son superiores a 20000 Hertz o inferiores a 20 Hertz. Por lo tanto, para aquellas frecuencias de sonido que son superiores a 20000 Hertz, los científicos tienen otro nombre para ellos, es el "ultrasonido".

¿Cómo puede el sensor de ultrasonido medir la distancia?

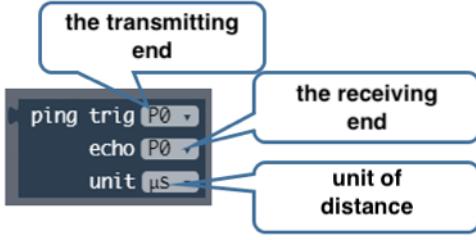


Como hemos aprendido arriba el sensor ultrasónico puede, a través del aire, enviar ultrasonido a través del extremo de transmisión. Una vez encontrados los obstáculos, el ultrasonido enviado será rebotado inmediatamente y

recibido por el extremo receptor. Por lo tanto, el sensor ultrasónico puede calcular la distancia a través de la fórmula: $S=340t/2$. Entre ellos, "S" es la distancia; "340" significa viajes ultrasónicos a 340 metros por segundo en aire; "t" es la cantidad total de tiempo; "/2" significa que el tiempo que necesitamos para el cálculo es el tiempo que el ultrasonido viaja hasta el obstáculo. Pero la "t" (la cantidad total de tiempo) es el tiempo tanto para transmitir como para recibir, por lo que la "t" necesita ser dividida en dos.

Puede que ahora esté convencido de que Max:Bot es una herramienta profesional y potente. Incluso puede calcular la distancia en un segundo! ¡Vamos a intentarlo!

Información básica

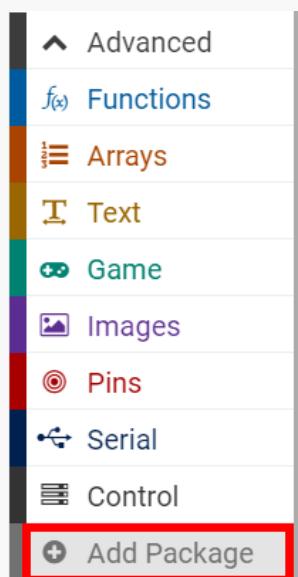
Bloques de función	Imágenes	Funciones
"ping unit"		<p>Para permitir que el sensor de ultrasonido calcule la distancia, necesitamos primero conectarlo correctamente. Como podemos ver en la imagen de la izquierda, el extremo transmisor debe estar conectado a la pin detrás de "ping trig" y el extremo receptor a la pin detrás de "echo". Además, necesitamos seleccionar la unidad correcta (cm) a la derecha de "unit".</p>

Programa

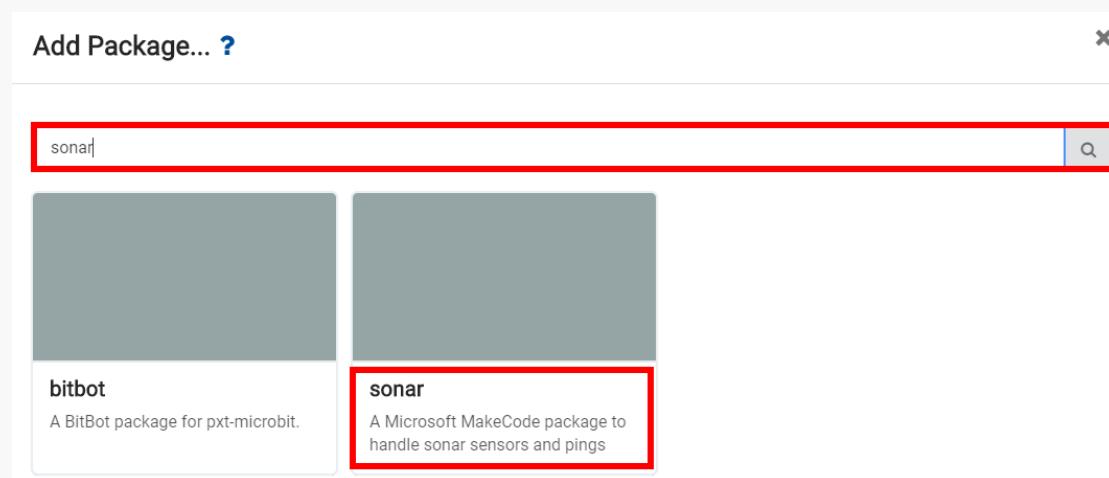
- (1) Comenzar un nuevo programa y nombrarlo como "medidas de distancia por ultrasonidos"

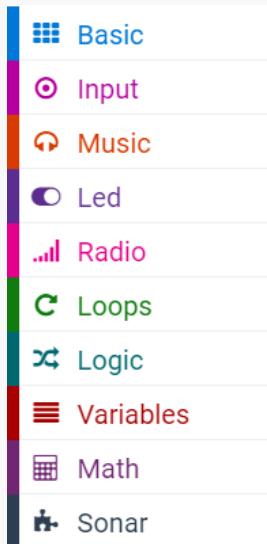
(2) Añadir paquete

El área funcional ha enumerado algunas funciones básicas. Sin embargo, para los requisitos de funciones especiales, tenemos que utilizar el "Add package". Necesitamos añadir la función "sonar" en esta parte.

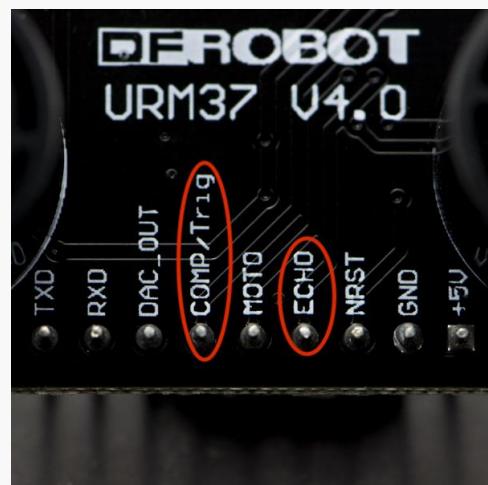
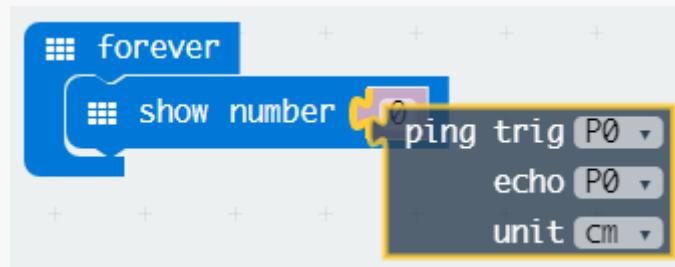


Haga clic en "Add package" y busque "sonar". Luego haga clic en el paquete llamado "sonar" y será listado automáticamente en el área de funciones.





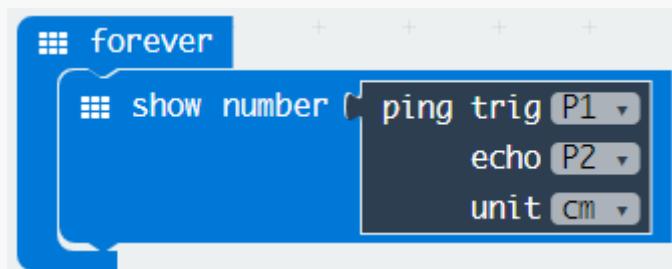
- (3) Primero tenemos que colocar la función "show number" de "Basic" en el bucle "forever". A continuación, coloque la función "ping unit" de "Sonar" en la función "show number".



La imagen de arriba es del sensor de ultrasonido. Entre ellos, el "COMP/Trig" representa el extremo de transmisión y debe conectarse a P1, y "ECHO" el extremo de recepción a P2. Durante la

programación, en la función "ping unit", debemos cambiar el Pin de "trig" a P1, y el "echo" a P2.

- (4) Al juntar todas las funciones listadas arriba, tendremos el programa final como se muestra a continuación. Descárgalo a Max:Bot y se mostrará la distancia calculada entre Max:Bot y el obstáculo.



Importante: Tenga en mente encender Max:Bot. Para hacer el cálculo más preciso, es mejor mantener la distancia entre Max:Bot y el obstáculo entre 5cm y 300cm.

Ejercicios

Arriba hemos aprendido a usar Max:Bot para medir una distancia. ¿Tienes idea de cómo usarla para medir tu altura? ¡Sólo inténtalo!



1. **Consejos:** Para hacer el cálculo más preciso, es posible que necesite hacer algo más:
2. 1. Preste atención a dónde y en qué dirección se encuentra el sensor ultrasónico.
3. 2. Para evitar desviaciones mayores, es mejor ajustar Max:Bot dentro de 10cm.

4.2 Protección del automóvil

Creas o no, hoy en día los accidentes de tráfico se producen casi en cualquier lugar y en cualquier momento. Los coches nos hacen la vida más cómoda, pero al mismo tiempo nos ponen en peligro. ¿Cómo podemos ayudar a los conductores a ser más sensibles a esos peligros potenciales? Max:Bot hace el trabajo.



Información básica

Bloques de función	Imágenes	Funciones
"item"		La función "item" representa una variable. Podemos usarlo para realizar un efecto dinámico.
"set item to"		Podemos utilizar la función "set to" para fijar un valor para la variable "item". La imagen a la izquierda significa: el valor de la variable llamada "item" es igual a 0.
"and"		Cuando se utiliza la función " and ", significa que tanto la parte izquierda como la derecha deben coincidir simultáneamente.

Programa

- (1) Iniciar un nuevo programa y nombrarlo como " protección del automóvil ".
- (2) Item

¿Cómo utilizar la variable "item"?

El sistema sensorial de Max:Bot lleva las detecciones en tiempo real al entorno cambiante. Por ejemplo, el brillo puede variar de una luz a otra. Esto significa que Max:Bot requiere una función especial para detectar y transformar el brillo de la variable en un valor determinado. Así tenemos la función "set item to". Entre ellos, el "item" en el medio es la variable detectada. Toda la función significa: ajustar el valor de la variable llamada "item".

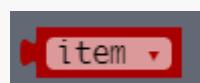
Busque la función "set item to" de "Variables" y colóquela en el bucle "forever".



En esta parte, los datos que se han detectado en tiempo real son la distancia que recorre el ultrasonido. Por lo tanto, necesitamos establecer el valor de la variable de acuerdo a la distancia detectada. Hemos aprendido a utilizar la función "ping unit" para calcular la distancia, por lo que, en esta parte, la función "ping unit" debe situarse en la función "set item to".

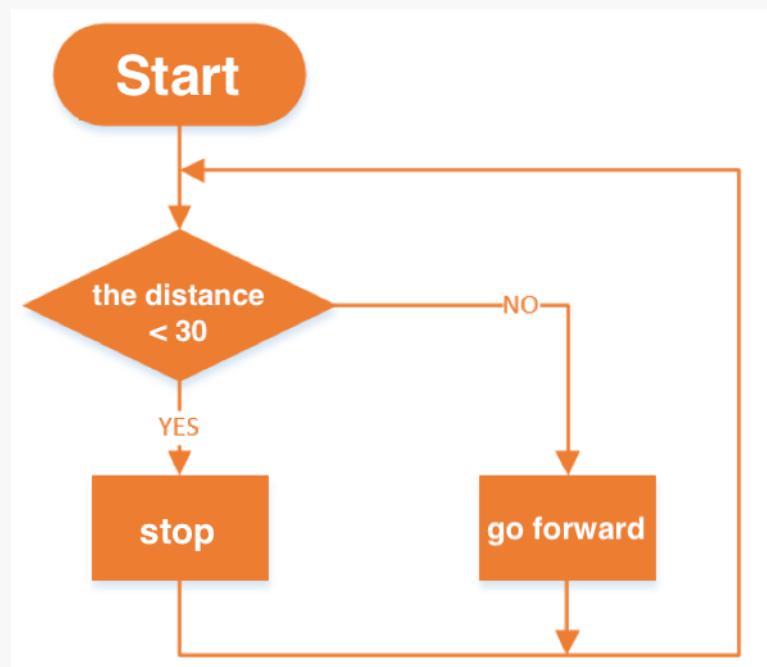


Puesto que el valor de la variable llamada "item" ha sido fijado, ahora podemos usar la variable "item" directamente en las partes posteriores. La variable "item" de las "Variables" se encuentra en el área de funciones.

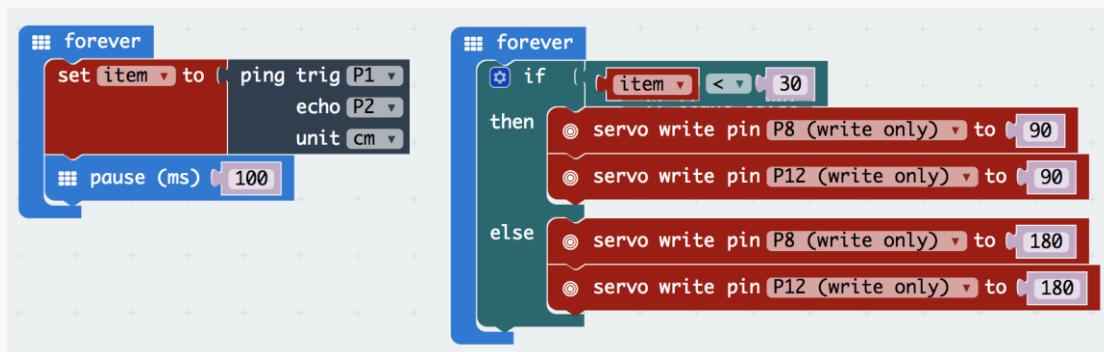


- (3) Una vez que la distancia entre el obstáculo por delante y el Max:Bot es menor que el valor preestablecido, el Max:Bot se detendrá automáticamente para evitar una colisión por detrás.

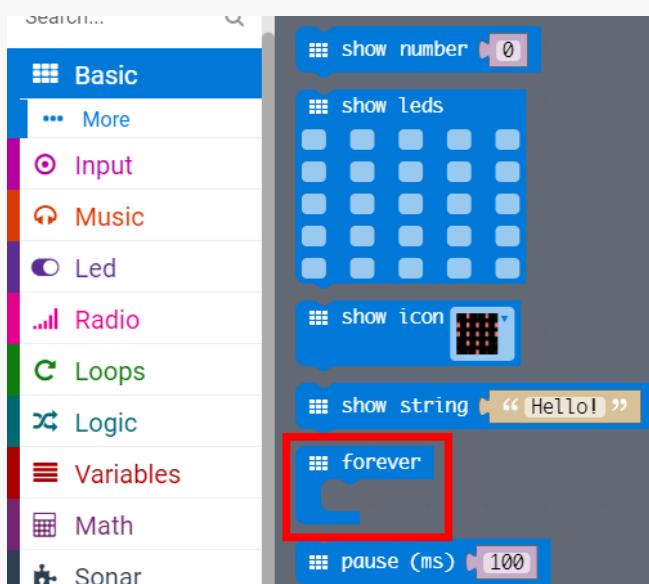
Suena como una muy buena idea, pero ¿cómo lograrlo? Podemos utilizar el siguiente diagrama como guía.



- (4) Combinar todos los bloques de función listados arriba, el programa final es el que se muestra a continuación. Descárguelo a Max:Bot . Entonces usted podrá disfrutar de un viaje agradable y seguro!



Como habrás notado, hay dos bucles "forever". Esto se debe a que necesitamos que Max:Bot opere dos programas simultáneamente sin influenciarse el uno al otro. Por un lado, debe detectar el valor de la variable y, por otro, debe determinar por sí mismo cuándo detenerse y cuándo avanzar. Así que necesitamos usar dos bucles "forever" en esta parte. Encontrará el bucle "forever" desde "Basic" en el área de funciones.



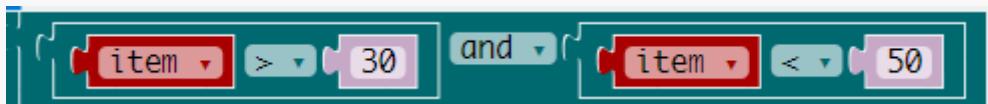
Ejercicios

Al conducir, el coche no se detendrá por completo, sino que irá disminuyendo gradualmente la velocidad. Cuando la distancia entre él y el vehículo que circula por delante se acerca cada vez más, su velocidad se reducirá en consecuencia.

¡Vamos a intentarlo!



Consejos: la función "and" se puede utilizar para juzgar diferentes distancias.



Capítulo 5: Max:Bot Go!

¿Recuerdas cuántos ojos tiene Max:Bot? En primer lugar, tiene los "ojos" (los sensores digitales de colisión y el sensor de luz) para evitar obstáculos y percibir la luminosidad; a continuación, tiene el ojo en la cara (el sensor ultrasónico) para medir la distancia entre éste y el obstáculo. ¿Se preguntará si tiene otros ojos? La respuesta es absolutamente sí. En este capítulo, se introducirá otro ojo. Max:Bot mantiene este ojo hacia el suelo para evitar caerse de una altura. Por cierto, con la ayuda de este ojo, Max:Bot puede seguir un trazo fijo. ¡Explorémoslo!



Objetivos:

1. ¿Cómo utilizar el sensor de seguimiento de línea?

¿Cómo utilizar la función "and"?

Módulos digitales

Módulos digitales	Imágenes	Funciones
-------------------	----------	-----------

El sensor de seguimiento de línea		El sensor de seguimiento de línea permite al Max:Bot trazar una línea blanca sobre fondo negro o una línea negra sobre fondo blanco.
-----------------------------------	---	--

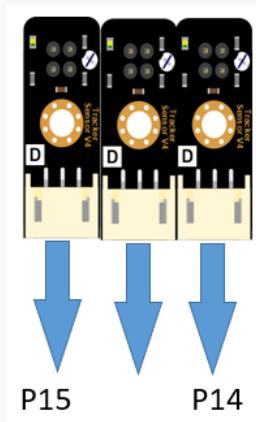
5.1 Max:Bot despierta!

¡Cuidado! ¡El acantilado está adelante! ¡Tenemos que parar ahora! Tranquilo, Max:Bot nunca hará algo tan absurdo. Gracias al sensor de seguimiento de línea, siempre puede reaccionar ante el peligro en el último momento.



Pero ¿cómo funciona el sensor de seguimiento de línea?

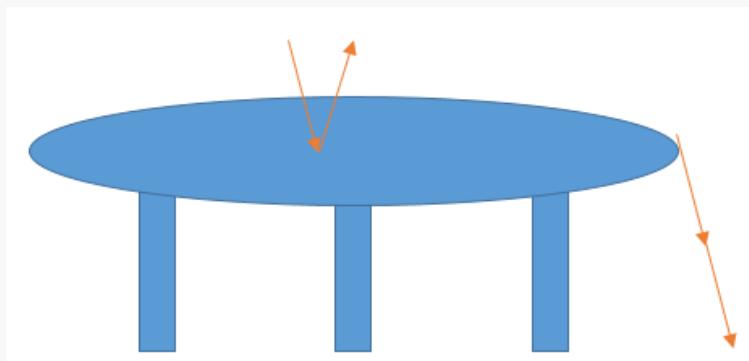
Hay tres sondas en el sensor de seguimiento de línea. Pero en esta parte, usaremos sólo dos de ellos. Conectar la sonda izquierda a P15 y la derecha a P14.



Además, tanto la sonda izquierda como la derecha pueden utilizarse como sensores infrarrojos separados.



Esto significa que ambas sondas pueden transmitir y recibir infrarrojos. La forma en que funciona la sonda es similar a la del sensor ultrasónico. Sin embargo, no calculará la distancia de acuerdo a la información recibida sino sólo para juzgar si hay obstáculos o no.



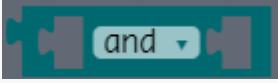
¿Cómo funciona el sensor de seguimiento de línea?

Al ser colocadas sobre una mesa, ambas sondas transmiten infrarrojos. El infrarrojo transmitido se encuentra con el obstáculo (la mesa) y luego es rebotado y recibido por ellos. Así, el indicador se enciende y el Max:Bot recibe un alto voltaje (presentado como 1).

Mientras que cuando Max:Bot llega al borde del escritorio, el infrarrojo transmitido no será rebotado. Así, el indicador se apaga y Max:Bot recibe un bajo voltaje (presentado como 0).

Información básica

Bloques de función	Imágenes	Funciones

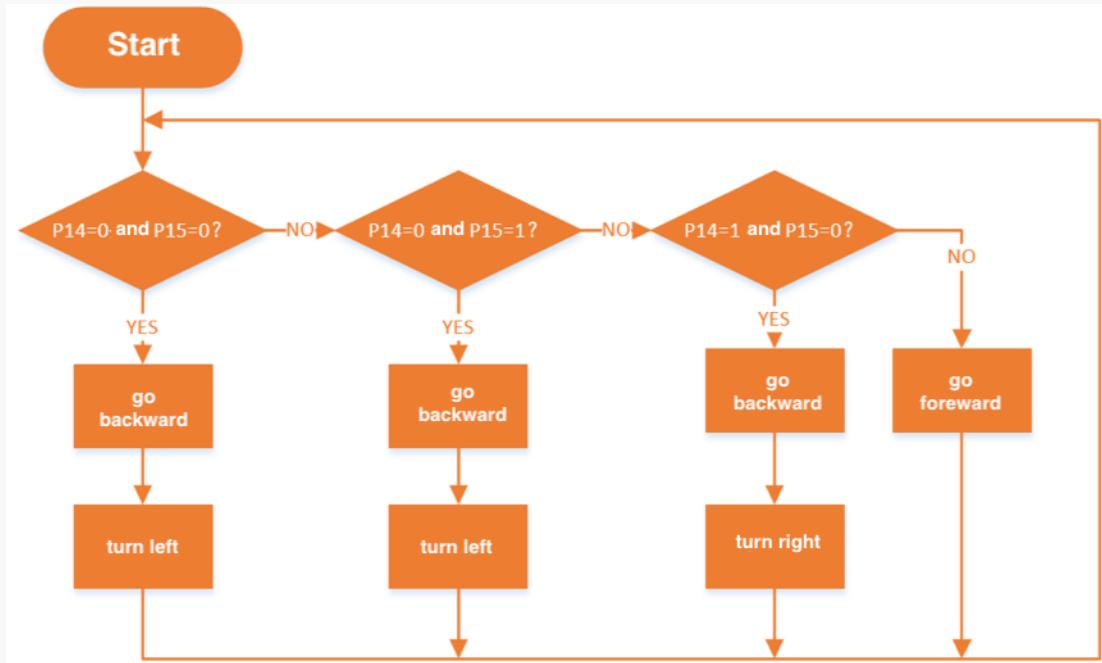
"and"		Cuando se utiliza la función " and ", significa que tanto la parte izquierda como la derecha deben coincidir simultáneamente.
-------	---	---

Programa

- (1) Inicie un nuevo programa y llámelo "Max:Bot despierta!".
- (2) En este programa, sólo utilizaremos las sondas izquierda y derecha del sensor de seguimiento. Conéctelos por separado a P15 y P14. A continuación se presentan cuatro situaciones que deben tenerse en cuenta:

status	description	status of the probes
	both probes detect obstacles(the desk)	P14=1; P15=1;
	both probes detect nothing	P14=0; P15=0;
	the left probe detect an obstacle, and the right probe detect nothing	P14=0; P15=1;
	the left probe detect nothing, and the right probe detect an obstacle	P14=1; P15=0;

- (3) De acuerdo a las cuatro situaciones potenciales, ahora podemos dibujar un diagrama para programar Max:Bot.

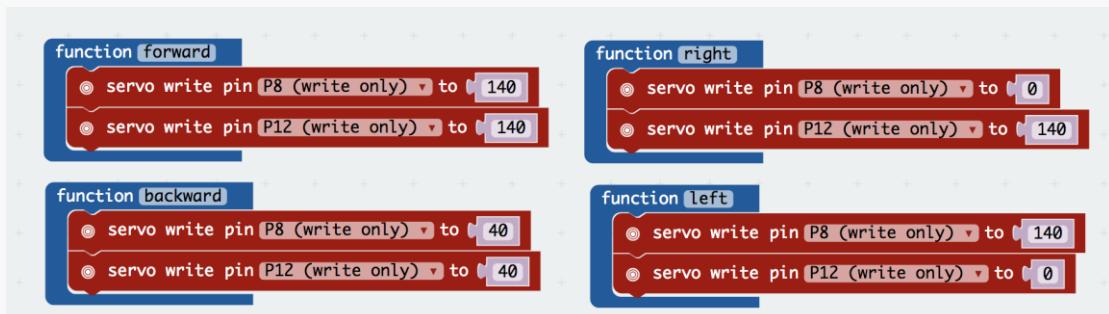
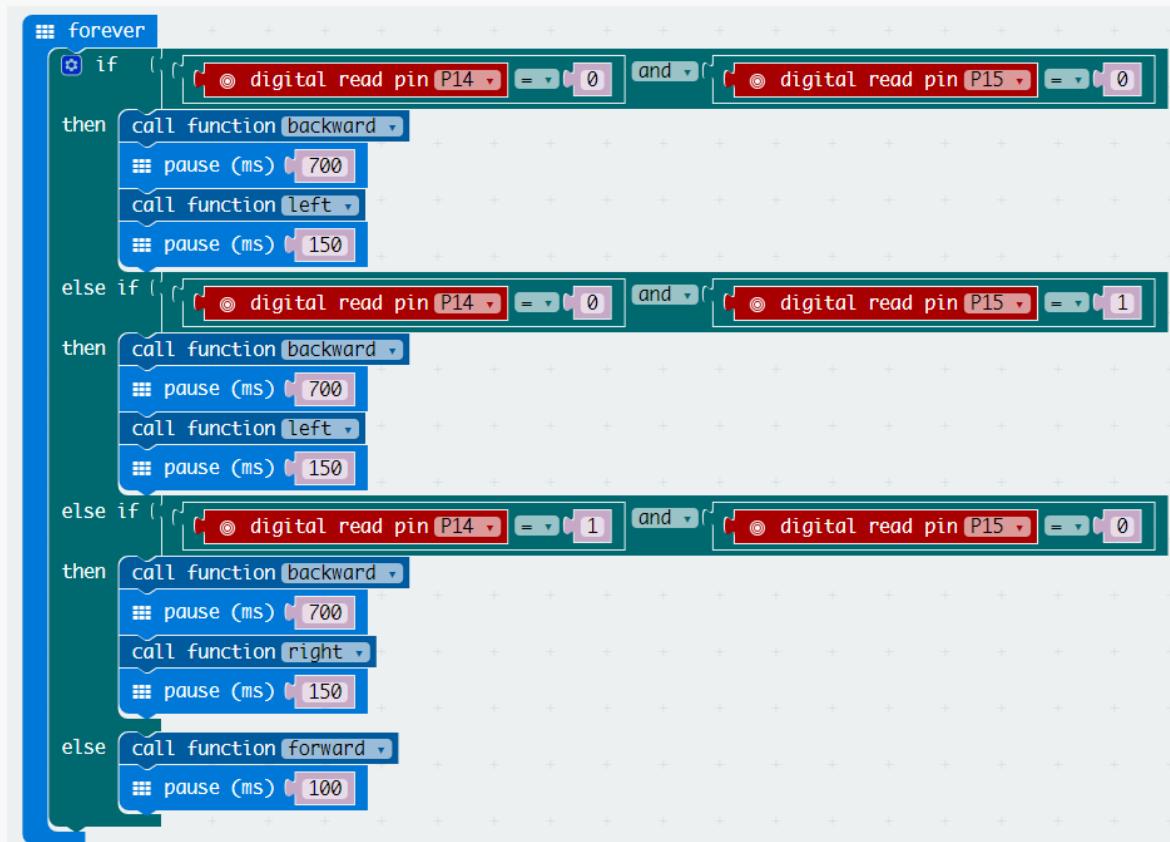


- (4) A continuación hemos aplicado numerosas funciones que se han aprendido anteriormente. Revisémoslos juntos.

En primer lugar, tenemos que usar el sub-bucle "function" para permitir a Max:Bot girar a la derecha, avanzar, girar a la izquierda y retroceder.

La función múltiple "if else" aquí se usa para juzgar qué acción debería tomar Max:Bot: El primer "if then" significa que ambas sondas detectan el precipicio, Max:Bot retrocederá y girará a la izquierda; el segundo "if else" indica que la sonda derecha detecta el precipicio, Max:Bot retrocederá y girará a la izquierda; el tercer "if else" indica que la sonda izquierda detecta el precipicio, Max:Bot retrocederá y girará a la derecha. El último "else" significa que, si no se cumple ninguno de los tres "if then", Max:Bot seguirá adelante.

Al juntar todas las funciones listadas arriba, el programa final quedara como se muestra a continuación. Descárguelo a Max:Bot . El robot inteligente ahora puede bailar libremente sobre la mesa.



Consejos: La velocidad del Max:Bot debe mantenerse dentro de un rango razonable para evitar caerse de la mesa.

Ejercicios

¿Puede encontrar otros objetos que deban evitarse al caer de una altura?



Hoy en día, las herramientas cada vez más avanzadas son recibidas por los seres humanos con los brazos abiertos. Como el robot barredor auto cargable que puede limpiar el suelo a tiempo. Para evitar caídas desde las escaleras, debe programarse con función anti caídas.

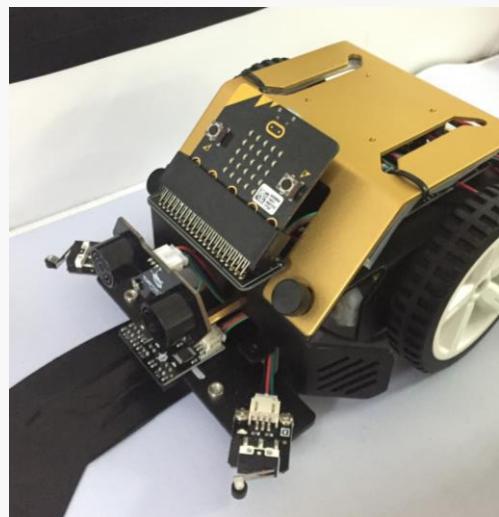
Hay una serie de sondas en el borde inferior del robot barredor para protegerlo de caídas desde las escaleras.

Una vez que haya pensado en otros objetos que deberían protegerse de caerse, puede anotarlos en el espacio en blanco que aparece a continuación. Cuanto más, mejor.

-
-
-

5.2 Rastrea la pista

Dale a Max:Bot una pista lo suficientemente larga, él la rastreará por todo el mundo.



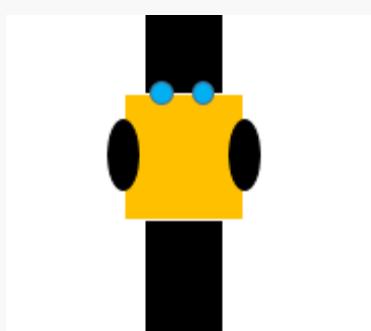
¡Juguemos a la búsqueda del Tesoro!

Información básica

Bloque de función	Imágenes	Funciones
"and"		Cuando se utiliza la función " and ", significa que tanto la parte izquierda como la derecha deben coincidir simultáneamente.

Programa

- (1) Comenzar un nuevo proyecto y nombrarlo como "Rastrea la pista"
- (2) En este capítulo, necesitamos primero pavimentar una pista negra con un lápiz negro o una cinta negra. Entonces deberíamos programar para permitir que Max:Bot rastree a lo largo de ella.

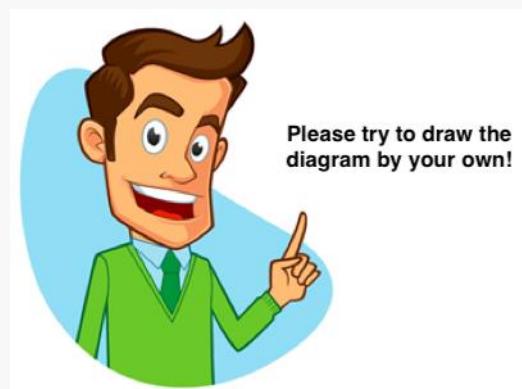


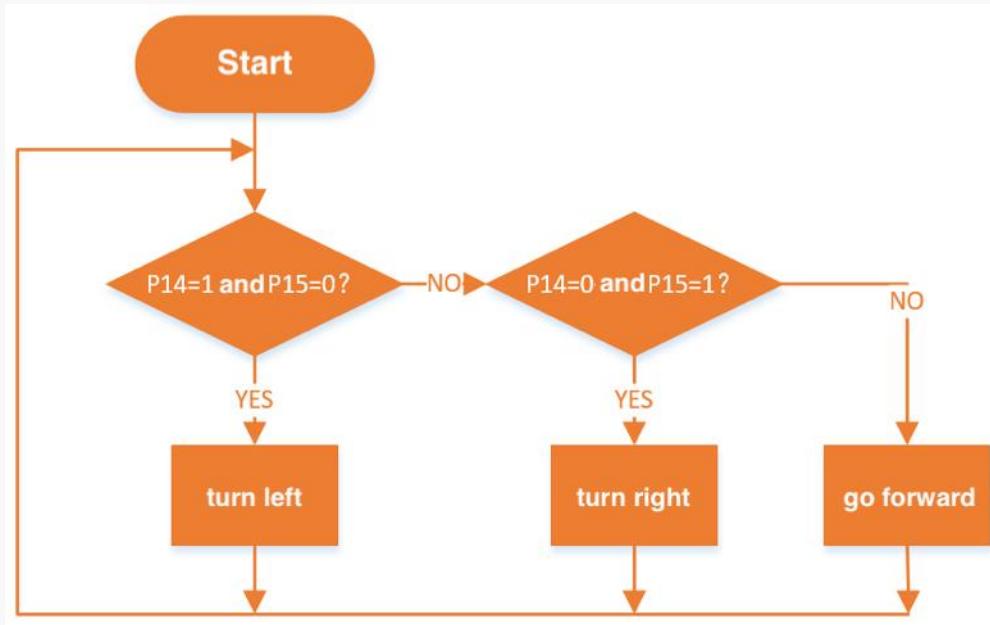
Tenga en cuenta: la pista negra debe ser lo suficientemente ancha para que tanto la sonda izquierda (P14) como la derecha (P15) de los sensores de seguimiento puedan detectarla simultáneamente.

- (3) Se deben tener en cuenta las siguientes consideraciones:

status	description	status of the probes
	both the probes detect the black track	P14=0; P15=0;
	the left probe detects the track, while the right detects nothing	P14=0; P15=1;
	the left probe detects nothing, while the right detects the track	P14=1; P15=0;

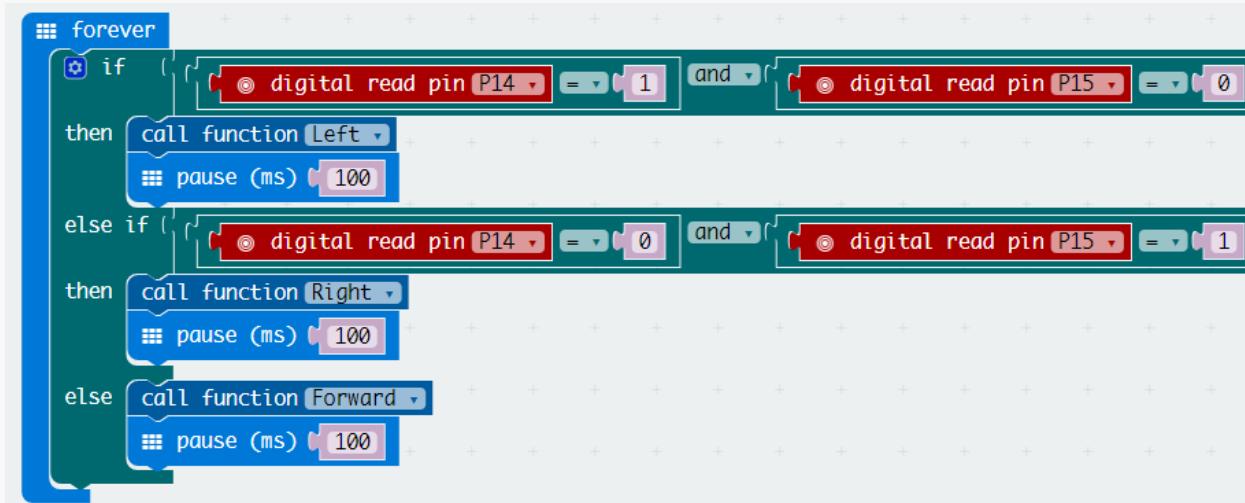
- (4) Como se ha analizado anteriormente, ahora podemos dibujar un diagrama para la programación.

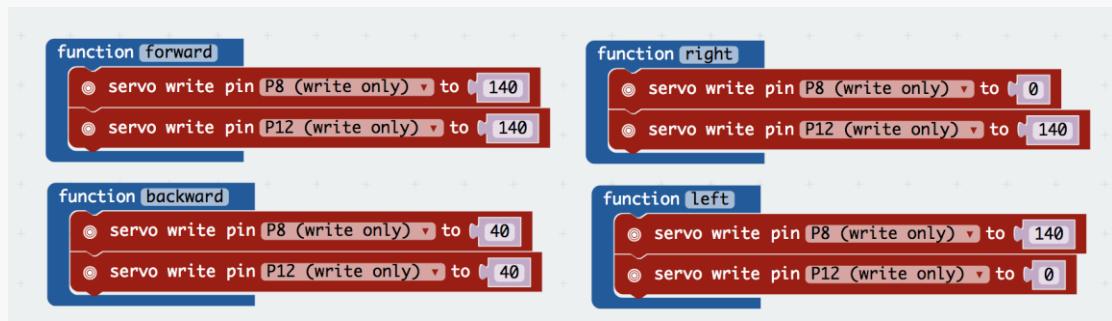




- (5) En este programa, al igual que en el programa final de la última parte, utilizamos la "function" para permitir a Max:Bot girar a la derecha, avanzar y girar a la izquierda. La función "is else" múltiple se aplica para indicar a Max:Bot cuándo girar a la izquierda, cuándo girar a la derecha y cuándo seguir adelante. Combine todos los bloques de función listados arriba, tendremos el programa final como se muestra a continuación.

Descargue el programa final al Max:bot. A continuación, se irá y rastrear a lo largo de la línea negra

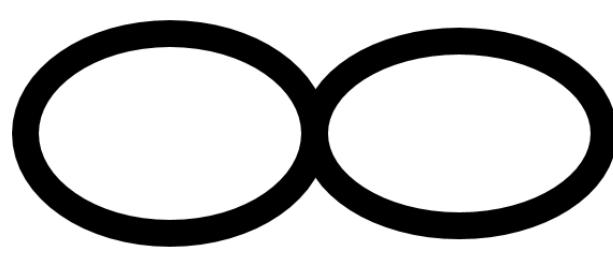




Sugerencias: Para mantener a Max:Bot siempre rastreando a lo largo de la pista, necesitamos mantener su velocidad y tiempo de pausa dentro de un rango razonable.

Ejercicios

Max:Bot ahora puede viajar alrededor del mundo por la pista negra! Puede que no lo creas porque algunas carreteras son bastante complejas. ¿Es Max:Bot capaz de seguir pistas complicadas como las siguientes?



En el curso de la programación, usted puede encontrar algunos problemas. Anótalos y discute con tus amigos para encontrar la solución.

-
-
-

Capítulo 6: Radiocomunicación

Con un Max:Bot, puedes controlar casi todo. ¿Y si tienes dos Max:Bots en la mano?



Como hemos aprendido que Micro:bit es el cerebro de Max:Bot . Aparte de las potentes funciones que hemos experimentado en capítulos anteriores, Micro:bit tiene otra función que puede realizar el control remoto.

¡Explorémoslo!

Objetivos:

1. ¿Cómo utilizar la función de comunicación por radio?
2. ¿Cómo utilizar el acelerómetro?

Módulos digitales

Modulo digital	Imágenes	Funciones
Micro:bit	A photograph of a Micro:bit microcontroller board. It features a central circular connection point, two small buttons on the sides, and five pins labeled 0, 1, 2, 3V, and GND at the bottom. The board is dark with gold-colored components.	Micro:bit incorpora un módulo de radio y un acelerómetro.

6.1 Di Hola!

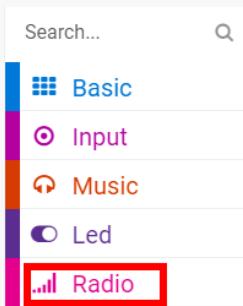
Cuando dos Max:Bots se encuentran, ¿qué harían? Deben ser decir hola. Pero, ¿cómo lograrlo?



La función de comunicación por radio hace el trabajo. Cuando dos Max:Bots se encuentran, uno de ellos envía una señal a través de Micro:bit, mientras que el otro recibe la señal y responde a ella.

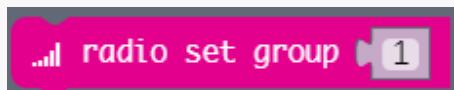
¿Cómo utilizar la función "Radio"?

Necesitaremos dos micro: bits en esta parte, uno envía el mensaje (extremo transmisor), el otro recibe la señal (extremo receptor). El micro: bit se comunica por radio.



En primer lugar, es necesario conocer las tres funciones más utilizadas en "Radio":

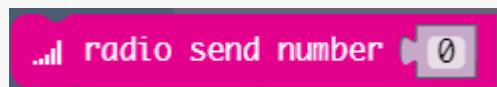
1、 "set group":



Descripción de la función: Hay un total de 255 canales de radio diferentes. El transmisor y el receptor deben estar en el mismo grupo para poder comunicarse.

Por favor, tenga en cuenta: se deben configurar diferentes canales de radio para diferentes jugadores para evitar cualquier confusión.

2、"send number" y "send string":

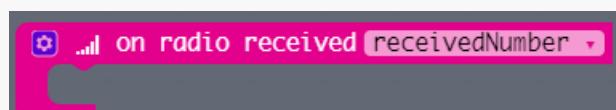


Descripción de la función: enviar un número o una variable por radio.

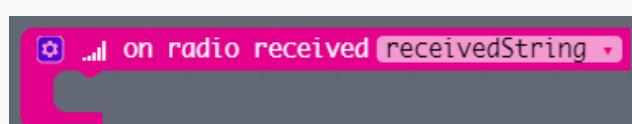


Descripción de la función: enviar una cadena por radio.

3、"on radio received number" y "on radio received string":

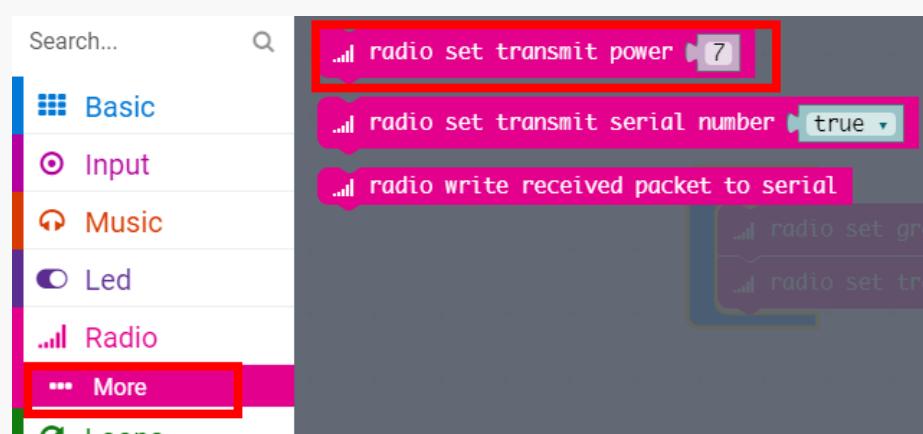


Descripción de las funciones: Al recibir un número o una variable, ejecutará el programa dentro del bucle "on radio received number".



Descripción de las funciones: Al recibir una cadena, ejecutará el programa dentro del bucle "on radio received string".

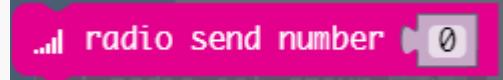
4、"set transmit power"



Se pueden seleccionar 7 grados para la potencia de transmisión de radio.

Cuanto más alto es el grado, más lejos transmite la potencia.

Información básica

Bloques de función	Imágenes	Funciones
"on start"		Las funciones que se colocan en el "on start" se ejecutarán una sola vez.
"set group"		Defina el canal de radio.
"radio send number"		"0" envía el número "0" por radio
"on radio received number"		Al recibir un número, éste ejecutará el programa dentro del bucle "on radio received".
"set transmit power"		Ajuste el grado de potencia de transmisión de radio.

Programa

El extremo de transmisión

- (1) Comenzar un nuevo proyecto y nombrarlo como "Hola transmisor"
- (2) Ajuste inicial del extremo de transmisión

Las funciones que se colocan en "on start" se ejecutarán una sola vez. Lo que significa que tanto el "set group" como "set transmit power" se ejecutarán sólo una vez después del encendido.

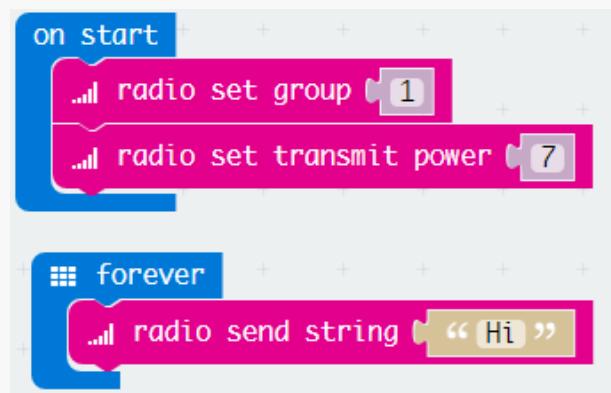


(3) Configurar el contenido de la transmisión

Dado que los Max:Bot se saludan unos a otros, necesitamos usar la función "radio send string" para enviar el "Hi" por radio.

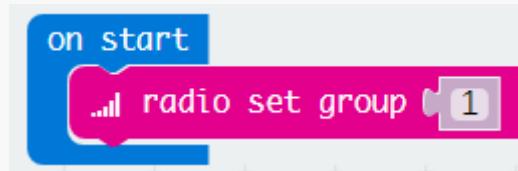


(4) Combinar todos los bloques de función listados arriba, el programa final del final de la transmisión es el que se muestra a continuación. Descárguelo en el extremo de transmisión.



El extremo receptor

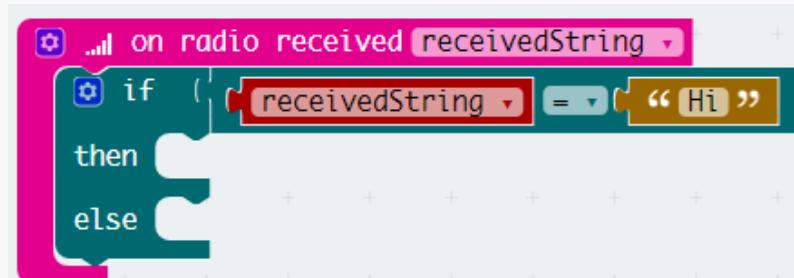
- (1) Comenzar un nuevo proyecto y nombrarlo como "Hola receptor"
- (2) Configurar el mismo canal de radio que el extremo de transmisión mediante "set group". Colócalo en "on start".



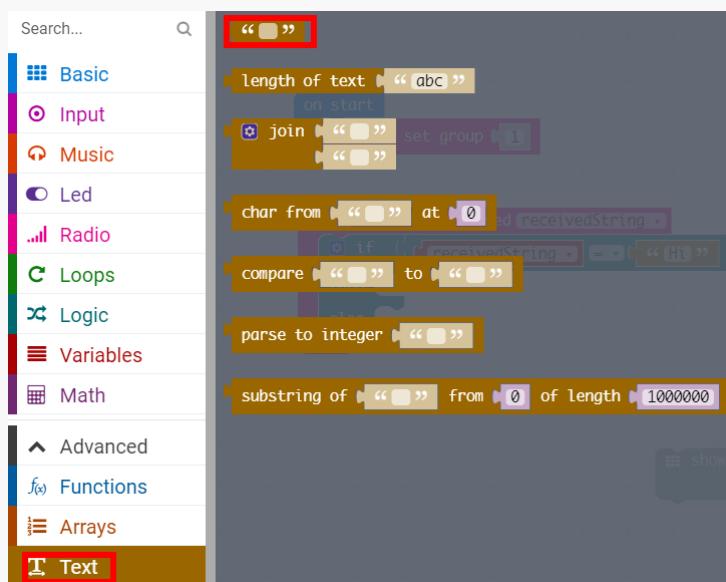
- (3) La función "on radio received string" se utilizará para juzgar si la información transmitida se recibe o no.



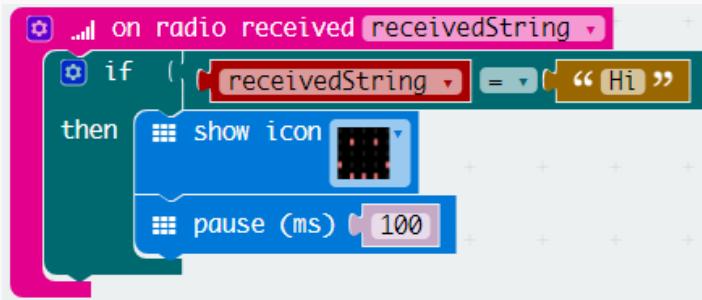
- (4) A continuación, utilizamos la función "if else" para juzgar si la información recibida es "Hi" o no.



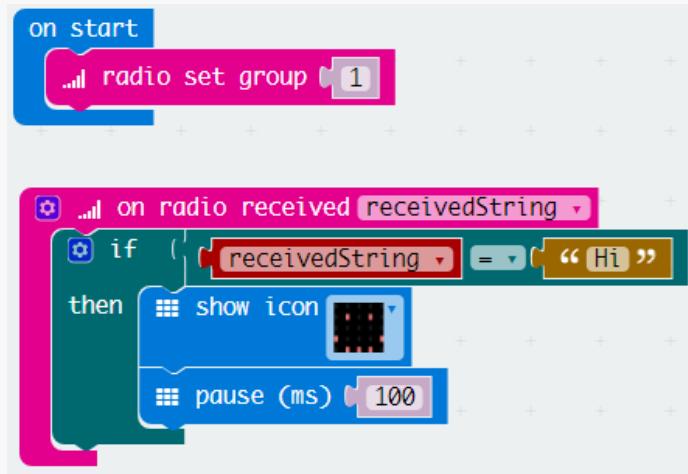
Tenga en cuenta: " Hi " está en forma de texto. Por lo tanto, la función "" se utilizará en esta parte para leer el texto. Puede encontrarla en "Text" en el área de funciones.



- (5) Si Max:Bot recibe el "Hi" por radio, el robot mostrará una cara sonriente. Aplicaremos la función "show icon" y la función "pause" para generar un rostro sonriente.



- (6) Al juntar todas las funciones listadas arriba, el programa final será el siguiente. Descárguelo en el extremo receptor.



Encienda el Max:Bot receptor y el Max:Bot transmisor, la cara sonriente podrá verse en el Max:Bot receptor.

Ejercicios

En algunas ocasiones, como en las películas de espías, necesitamos identificarnos a través de un cierto código. ¡Ahora juguemos un juego de espías!

Un Max:Bot envía un código, como 1231; mientras que el otro recibe el código, éste mostrará el código correspondiente, como 1231. Si falla, se probará que no es uno de los tuyos.



6.2 Carrera de detección de movimiento

¿Has jugado alguna vez a un juego de detección de movimiento, como las carreras? Este tipo de carreras requiere que conduzcas el coche en la pantalla a través de movimientos corporales reales.



Con Micro:bit, puedes jugar al juego de detección de movimiento. El acelerómetro incorporado hace el trabajo.

¿Qué es el acelerómetro?

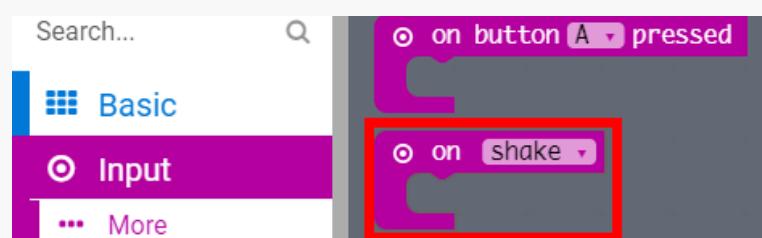
Un acelerómetro es un dispositivo que mide la aceleración adecuada. El acelerómetro incorporado se puede utilizar en combinación con otras funciones para indicar los movimientos de Micro:bit. Incluyendo su dirección de movimiento, ángulo, gesto, etc.

Este tipo de acelerómetros también se pueden encontrar en los teléfonos móviles, que permiten a sus usuarios jugar juegos de detección de movimiento. Además, el acelerómetro, que sirve como una de las partes más

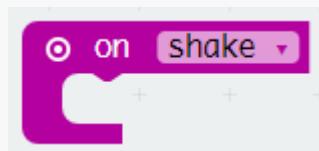
importantes del brazalete inteligente, puede leer los movimientos para proporcionar una función de recuento de pasos.



En el área de funciones, podemos utilizar la función "on" de la "Input" para leer los movimientos del Micro:bit.



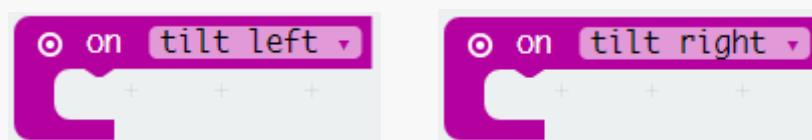
1. "on shake"



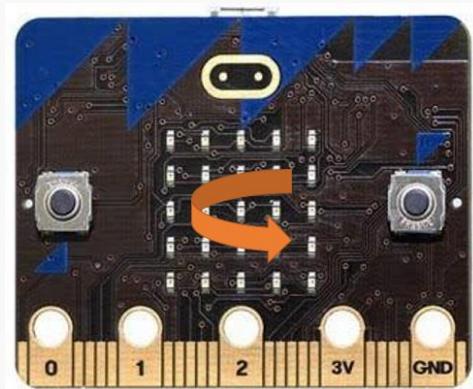
Descripción de la función: juzgar si el Micro:bit está temblando o no.

Esta función es similar a la función de conteo de pasos de un podómetro.

2. "on tilt left" y "on tilt right"



Descripción de la función: para juzgar si Micro:bit se inclina hacia la izquierda o hacia la derecha.



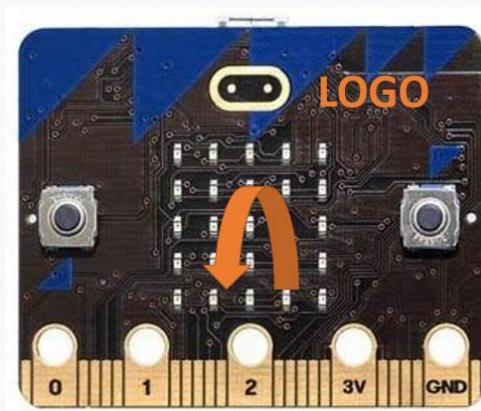
3. "on logo up" y "on logo down"



Descripción de la función: juzgar si Micro:bit se inclina hacia arriba o hacia abajo. Entre los cuales, el lado positivo tiene un LOGO que se



parece a:



Información básica

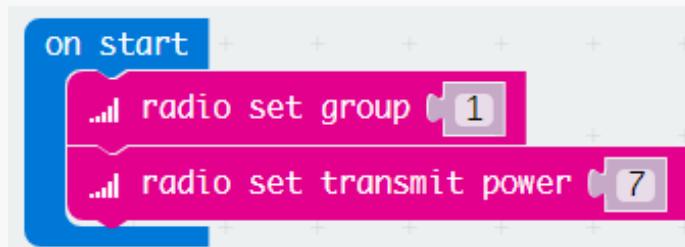
Bloques de función	Imágenes	Funciones
"on shake"		Para juzgar si Micro:bit está temblando o no.
"on tilt left"		Para juzgar si Mirco:bit se inclina a la izquierda o a la derecha.

"on logo up"		Para juzgar si Micro:bit se inclina hacia el lado del LOGO o no.
"show arrow"		Para mostrar una flecha en Micro:bit. El "North" significa que la flecha apunta hacia el norte. Hay otras siete direcciones que se pueden seleccionar.

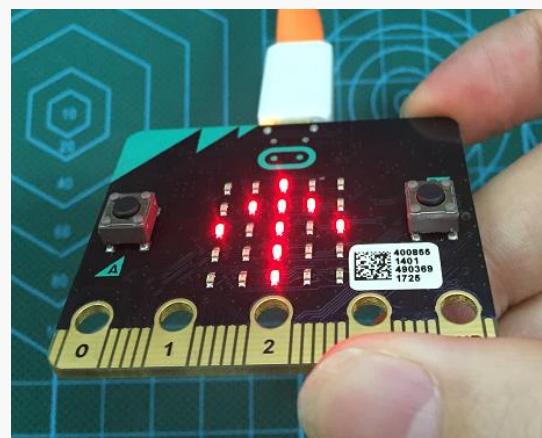
Programa

El extremo transmisor

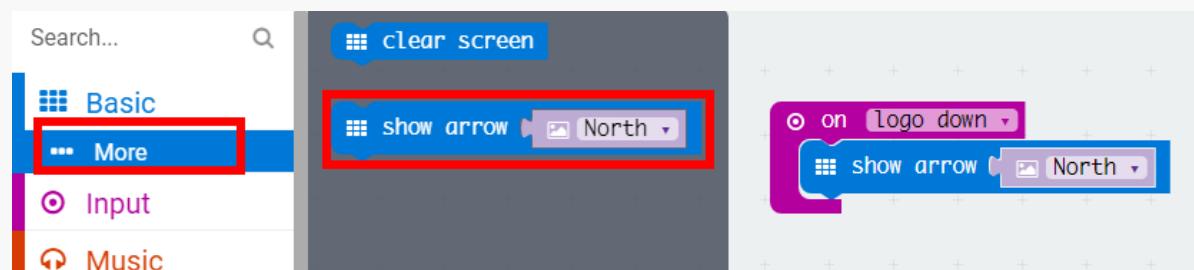
- (1) Comenzar un nuevo proyecto y nombrarlo como "sensing racing the transmitting end".
- (2) Utilice el botón "on start" para ajustar por primera vez el extremo transmisor.



- (3) Cuando se inclina hacia abajo (el lado sin LOGO), Micro:bit muestra una flecha. Y Max:Bot va hacia la dirección que indica la flecha.



Atención: la función " show arrow " no se encuentra directamente en el menú " Basic ". Primero debemos hacer clic en el botón " Basic ", luego aparecerá el botón " More " bajo el botón " Basic ". Haga clic en " More " y encontrará la función " show arrow ". Coloque la función " show arrow " en el " on logo down ". Lo que significa: en el momento en que el Micro:bit se inclina hacia el lado negativo, muestra una flecha apuntando hacia el norte.



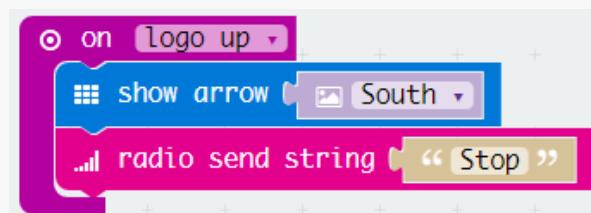
Necesitamos usar la función "radio send string" para enviar el "Go" por radio. Coloque la " Radio Send String " en el " on logo down ".



(4) Además, tenemos que programar los otros tres estados:

Detenerse

Coloque las funciones " show arrow " y " radio send string " en el " on logo up ". Ajuste la flecha apuntando hacia el sur y envíe por radio el "Stop".



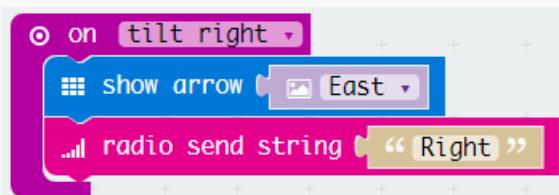
Girar a la izquierda

Coloque las funciones "show arrow" y "radio send string" en el "on tilt left". Ajuste la flecha apuntando hacia el oeste y envíe por radio el "Left".



Gire a la derecha

Coloque las funciones "show arrow" y "radio send string" en el "on tilt right". Ajuste la flecha apuntando hacia el este y envíe por radio el "Right".



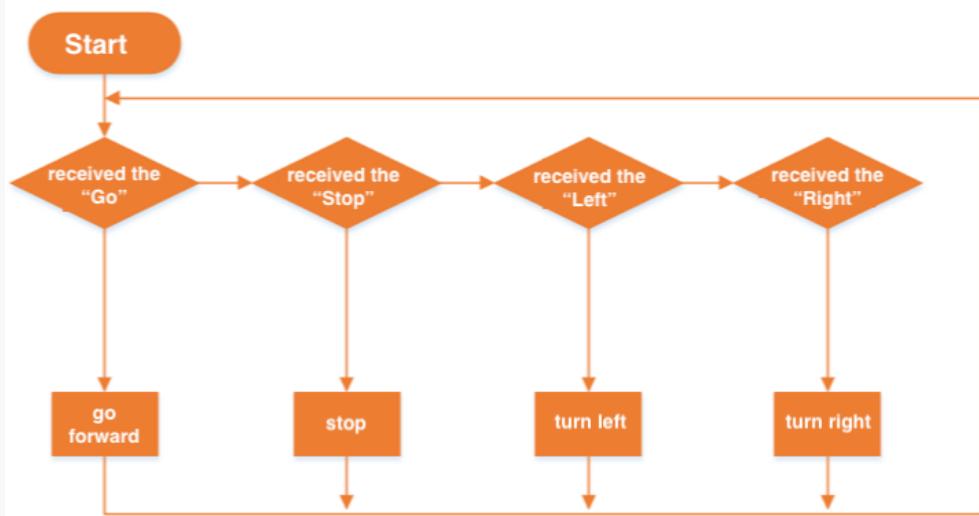
- (5) Al juntar todas las funciones listadas arriba, tendremos el programa final como se muestra a continuación. Descárguelo en el extremo de transmisión.



El extremo receptor

- (1) Comenzar un nuevo proyecto y nombrarlo como "sensing racing the receiving end"
- (2) Hay que programar los movimientos correspondientes de Max:Bot en función de los mensajes recibidos por el receptor.

Podemos programar el extremo receptor de acuerdo con el siguiente diagrama.



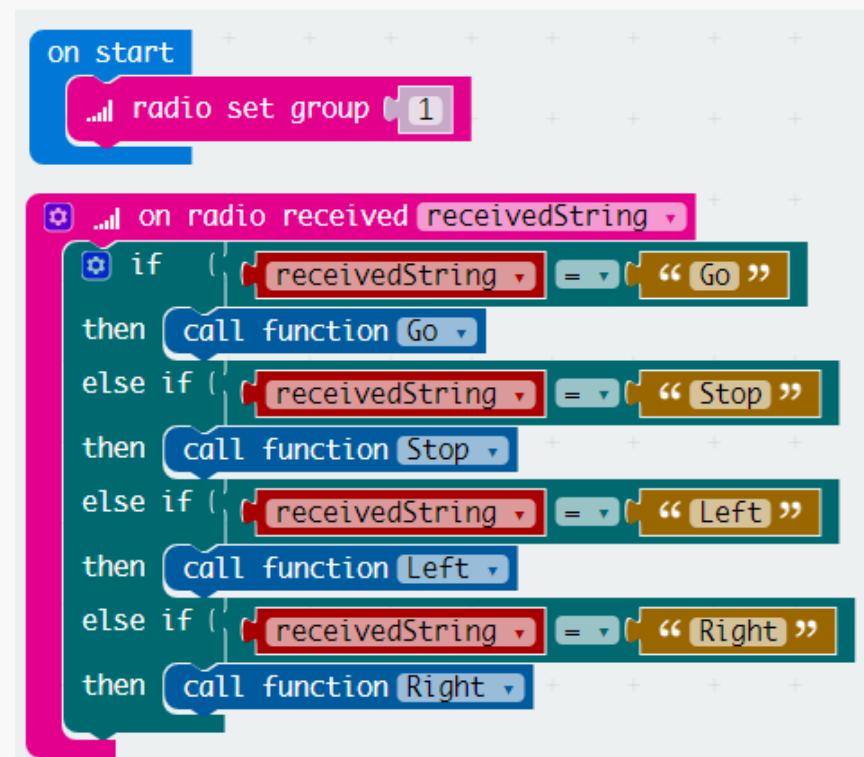
- (3) Aquí necesitamos primero usar el "servo write" para controlar la velocidad de las ruedas y realizar movimientos como: avanzar, parar, girar a la izquierda y girar a la derecha.

A continuación, la función "received string" y el operador "=" se utilizan para juzgar qué palabras se reciben.

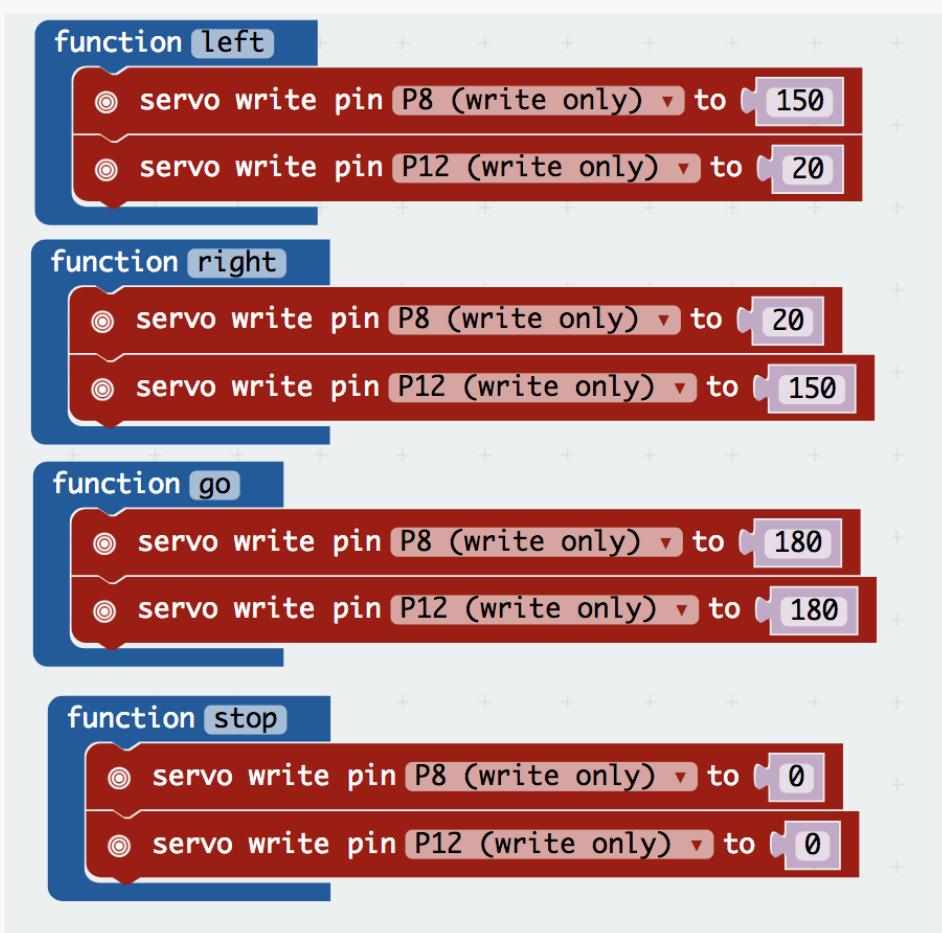
El múltiple "if else" se aplica para determinar cuándo debe Max:Bot avanzar, detenerse, girar a la izquierda y girar a la derecha.

Combinar todos los bloques de función listados a continuación, el programa final es el siguiente.

Descárguelo a Max:Bot . Entonces podemos invitar a nuestros amigos a jugar el juego de carreras.

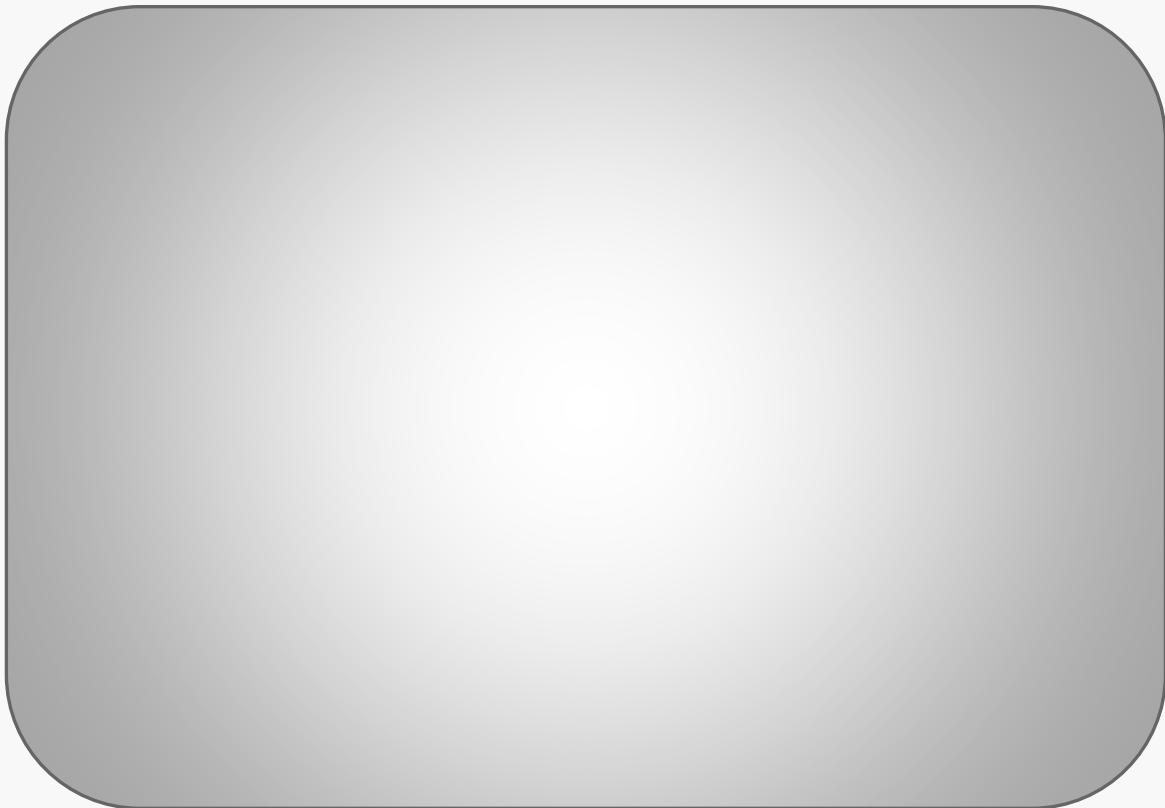


```
when green flag clicked
repeat (5)
    if receivedString = "Go" then
        call function Go
    else if receivedString = "Stop" then
        call function Stop
    else if receivedString = "Left" then
        call function Left
    else if receivedString = "Right" then
        call function Right
```



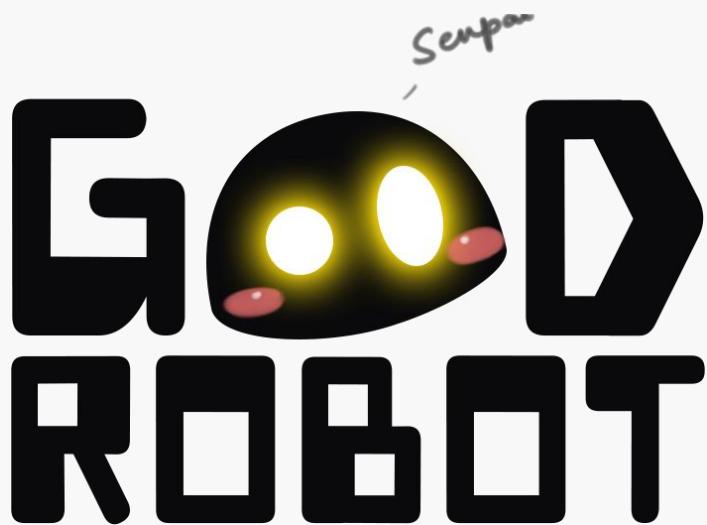
Ejercicios

Al jugar una carrera de detección de movimiento, cuanto mayor sea el ángulo de giro de las manos, mayor será el ángulo de giro del coche de carreras en la pantalla. Cómo lograrlo a través de Max:Bot? Trate de diseñar su propio diagrama en el espacio en blanco a continuación.



Capítulo 7: El espectáculo de luz y sonido

Max:Bot, nuestro compañero de confianza, nos ha llevado a escapar del laberinto y ganar la carrera de detección de movimiento. Puede que todavía tengas una sensación de desamparo porque Max:Bot nunca te habla. Pero en realidad, incluso puede reproducir un espectáculo de luz y sonido! Exploraremos más!



Objetivos:

1. ¿Cómo usar el módulo de la tira de LEDs?
2. ¿Cómo utilizar el módulo de altavoces digitales?
3. ¿Cómo programar una pieza musical?
4. ¿Cómo hacer una "Tira Led Fluida"?

Módulos Digitales

Módulos Digitales	Imágenes	Funciones
Tira de Leds		Hay muchos leds en la tira. A través de la programación se pueden configurar diferentes leds con diferentes colores, brillos y frecuencias de flash.
Módulo de altavoz digital		El módulo de altavoz digital es un módulo de salida que se utiliza para reproducir música.

7.1 Tira de luces arcoíris

En ocasiones especiales, como tu fiesta de cumpleaños, varios festivales, etc., a todos nos gustaría decorar nuestras casas con tiras de luz de colores.



En esta parte, usaremos la tira de LEDs para decorar Max:Bot.

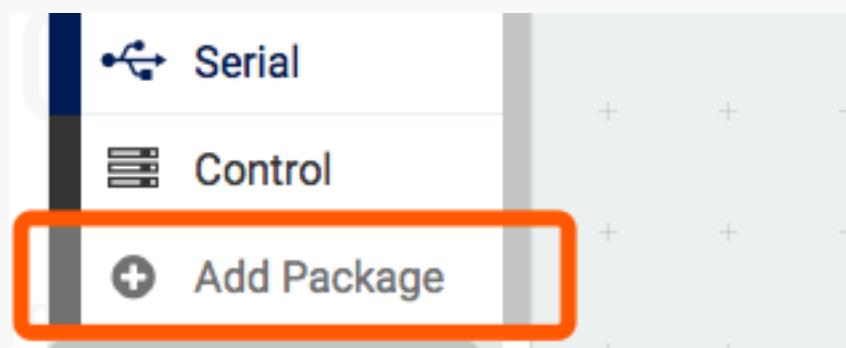
Información básica

Bloques de función	Imágenes	Funciones
"Neopixel"	<p>Pin value of the light strip total amount of LEDs color mode</p>	<p>La función "Neopixel" se puede utilizar para configurar el número de leds incluidos, el pin que se va a conectar y los colores que se van a mostrar.</p>
"range leds"	<p>the LEDs light up from 0 the first four LEDs light up</p>	<p>El "range leds" se utiliza para configurar cuántos leds (del número total de leds incluidos</p>

		físicamente en la tira) estarán bajo control del programa
"show rainbow"		La función " show rainbow " se utiliza para configurar el efecto de luz.

Programa

- (1) El bloque de funciones para el control de la tira de luz RGB no aparece en la caja de herramientas al iniciar un nuevo proyecto. Para añadir esta función, necesitamos importar manualmente el "Neopixel" desde la parte inferior de la página. Haga clic en "Add Package" y para abrir el administrador de funciones de extensión, seleccione "Neopixel". El Neopixel aparecerá en la lista de funciones.



Add Package... ? x

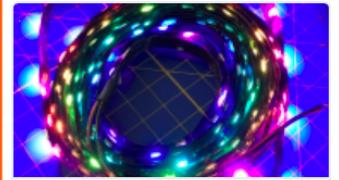
Search or enter project URL... q

devices

Camera, remote control and other Bluetooth services

bluetooth

Bluetooth services



neopixel
AdaFruit NeoPixel driver

C Loops

Logic

Variables

Math

Neopixel More

Advanced

Functions

Arrays

Text

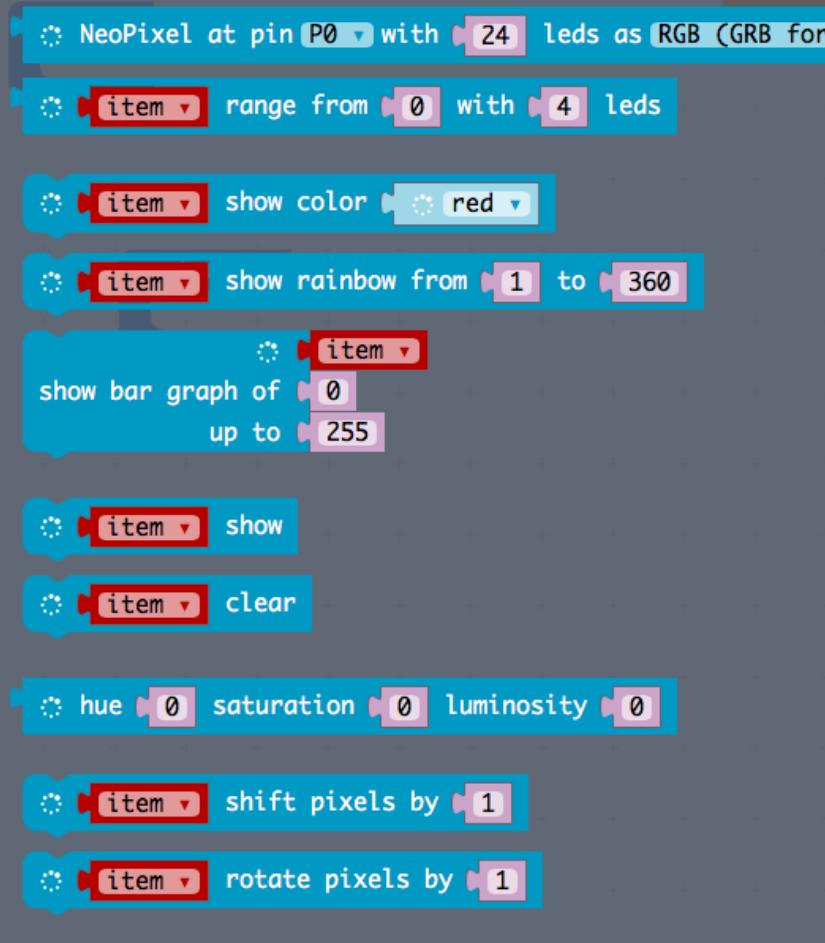
Game

Images

Pins

Serial

Control



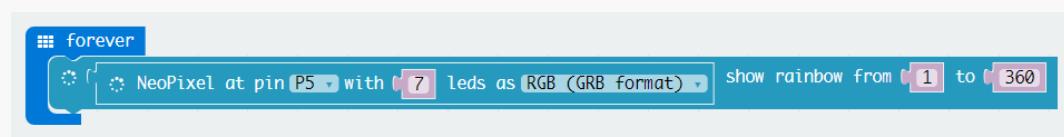
The Scratch script consists of the following blocks:

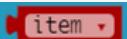
- Control: `when green flag clicked`
- Neopixel: `NeoPixel at pin P0 with 24 leds as RGB (GRB format)`
- Neopixel: `item range from 0 with 4 leds`
- Neopixel: `item show color red`
- Neopixel: `item show rainbow from 1 to 360`
- Neopixel: `item show bar graph of 0 up to 255`
- Neopixel: `item show`
- Neopixel: `item clear`
- Neopixel: `hue 0 saturation 0 luminosity 0`
- Neopixel: `item shift pixels by 1`
- Neopixel: `item rotate pixels by 1`

(2) Programar el efecto de luz arco iris

Para empezar a utilizar la función "Neopixel", primero configuraremos cuántos leds están incluidos en la tira de luz y a qué pin se conecta.

A continuación, utilice la función "show rainbow" para configurar el efecto de luz. Este efecto sólo se aplicará a los leds que definimos en nuestro último paso. El parámetro dentro del bloque de función se utiliza para establecer el rango de gradiente de color, 1 representa el rojo mientras que 360 representa el azul, cuanto más cerca estén estos dos números, menor será el cambio de gradiente:



Entre ellos, el  desde  show rainbow from 1 to 360 la función bock se sustituye por la función "Neopixel".

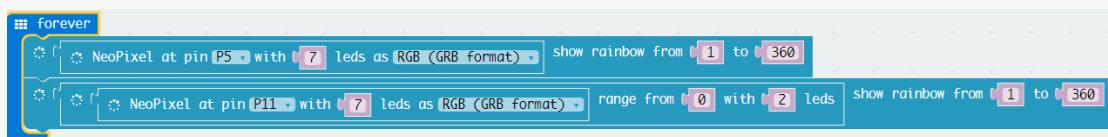
(3) Descargar el programa a Max:Bot .



La tira de LEDs ahora puede exhibir el efecto de luz de arco iris.

Ejercicios

Conecte dos tiras de luz por separado al Pin5 y al P11. Programe las tiras de luz como se indica a continuación.



¿Puede decirnos las diferencias entre las dos tiras de luz? Grábalas y compártelas con tus amigos!

-
-
-

7.2 A poner Música!

Escucha música suave y tus temores insensatos y tus celos desaparecerán. ¿Qué pasa si añadimos un efecto de sonido mientras jugamos? Por ejemplo, una pieza de una canción animada se reproduce para celebrar la victoria.

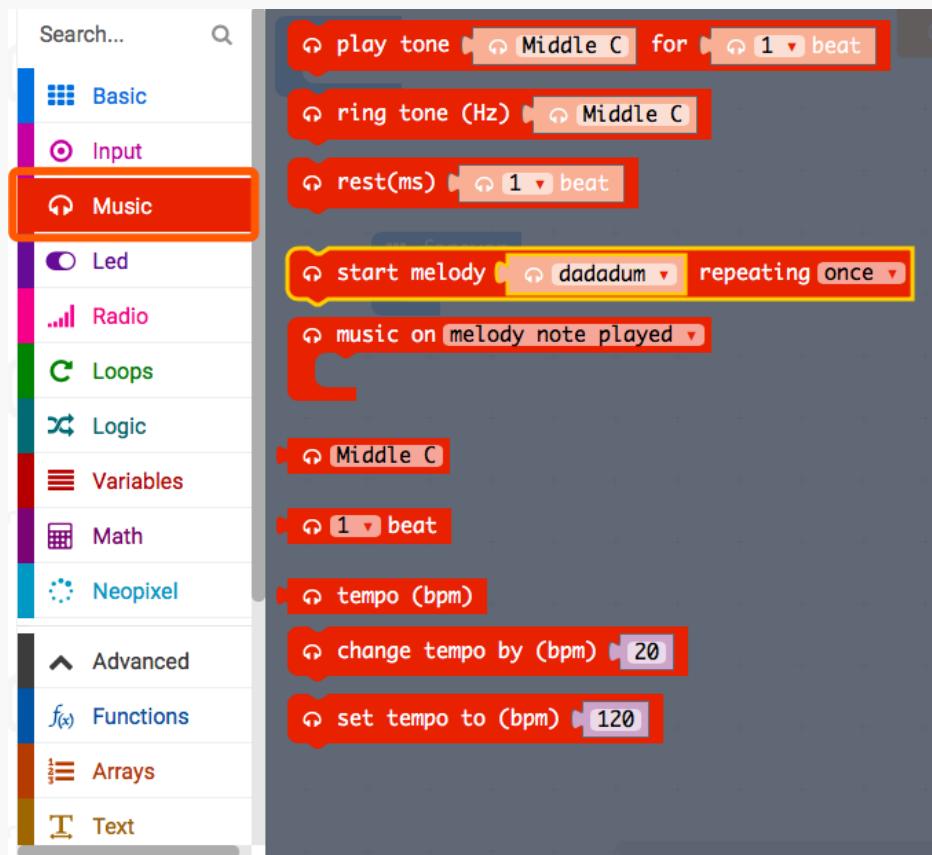


Información básica

Bloques de Función	Imágenes	Funciones
"start melody"		La función "start melody" se utiliza para seleccionar las canciones que aparecen en Makeode.
"play tone"		La función "play tone" se utiliza para ajustar el tono y los tiempos.

Programa

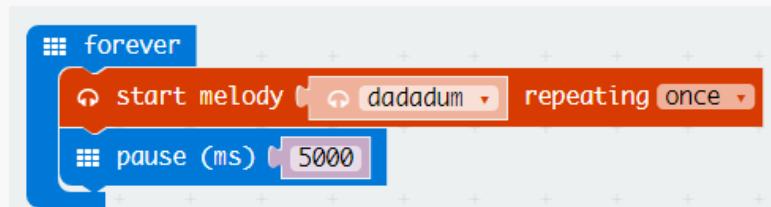
- (1) Comenzar un nuevo proyecto y nombrarlo como "a poner música"
- (2) La función "start melody" se encuentra en la función "Music" del área de función. Encuéntralo y colócalo en el bucle "forever".



- (3) Hay muchas melodías que se pueden seleccionar de la lista desplegable de "start melody". Aquí elegimos el "dadadum" para repetir "once".



- (4) En esta parte, usaremos la función "pause" para dar tiempo suficiente a Max:Bot a terminar su canción - el "dadadum". Combinando todos los bloques de función listados arriba, nuestro programa final será el siguiente:



- (5) Descárgalo a Max:Bot y escucharás su primera canción - el "dadadum".

Ejercicios

¿Qué pasa si quieres que Max:Bot cante una canción compuesta por ti? Él puede hacer tu sueño realidad. Usaremos la función "play tone" para ajustar el tono y los tiempos.



Como se ha mostrado en el diagrama siguiente, cada tono corresponde a una nota determinada.

Middle C	Middle D	Middle E	Middle F	Middle G	Middle A	Middle B
1(Do)	2(Re)	3(Mi)	4(Fa)	5(So)	6(La)	7(Si)

Aplicando la función "play tone", podemos trabajar juntos para editar la "Twinkle, Twinkle Little Star". Ven y únete a nosotros!

小星星

1=C 1 1 5 5 6 6 5 — 4 4 3 3 2 2 1 —
一 闪 一 闪 亮 晶 晶， 满 天 都 是 小 星 星，

5 5 4 4 3 3 2 — 5 5 4 4 3 3 2 —
挂 在 天 上 放 光 明， 它 是 我 们 的 小 眼 睛。

1 1 5 5 6 6 5 — 4 4 3 3 2 2 1 —
一 闪 一 闪 亮 晶 晶， 满 天 都 是 小 星 星。

7.3 El espectáculo de luz y sonido!

¿La Navidad es tu fiesta favorita? ¿Te gusta decorar el árbol de Navidad? En este capítulo, trabajaremos de la mano para crear un robot especial para árboles de Navidad que pueda ofrecer un espectáculo de luz y sonido para su familia y sus invitados!



Información básica

Bloques de función	Imágenes	Funciones
"repeat"	<p>The image shows a Scratch script consisting of a 'repeat' loop block with a value of 4. Inside the loop is a 'do' block which contains an empty script area.</p>	<p>Podemos primero colocar una función en "repeat" y luego utilizarla para determinar cuántas veces se repetirá la función.</p>
"change"	<p>The image shows a Scratch script consisting of a 'change' block. It has a dropdown menu labeled 'the name of a variable' and a value input field labeled 'the changing value'. The value field contains the number 1.</p>	<p>El "change" se utiliza para determinar cómo cambia una variable. Por ejemplo: nombramos la variable como "item" y fijamos el valor cambiante como "1". Lo que significa que cada vez que se ejecute la función "change", la variable añadirá 1. Como</p>

		<p>hemos aprendido antes, las funciones del bucle "forever" se ejecutarán repetidamente. Así, la primera vez que se ejecuta la función "change", la variable es "item", mientras que por segunda vez se repite, la "variable" se convertirá en "item+1". La variable continuará cambiando como se indica a continuación:</p> <p>Item → item+1 → (item+1)+1</p>
"while do"	<p>A Scratch script consisting of a green "while" control hat with a "true" condition and a grey "do" control body containing an empty script.</p> <p>conditions judgement</p>	<p>El bucle "while do" se utiliza para juzgar si se cumple o no la condición a la derecha del "while". En caso afirmativo, se ejecutará la función a la derecha de "do".</p>
"clear"	<p>A Scratch script consisting of a blue "clear" control hat with a "item v" parameter.</p>	<p>Con la función "clear" se apaga el efecto de luz.</p>
"show"	<p>A Scratch script consisting of a blue "show" control hat with a "item v" parameter.</p>	<p>Con la función "show" se activa el efecto de luz.</p>

Programa

- (1) Comenzar un nuevo proyecto y nombrarlo como "espectáculo de luz y sonido"
- (2) Utilice la función "play tone" para editar el "Jingle Bells".

| 3 3 3 — | 3 3 3 — | 3 5 1 · 2 | 3 — — — |

Jingle bells Jingle bells Jingle all the way

Las dos primeras frases de "Jingle Bells" mantienen las mismas notas y ritmos.

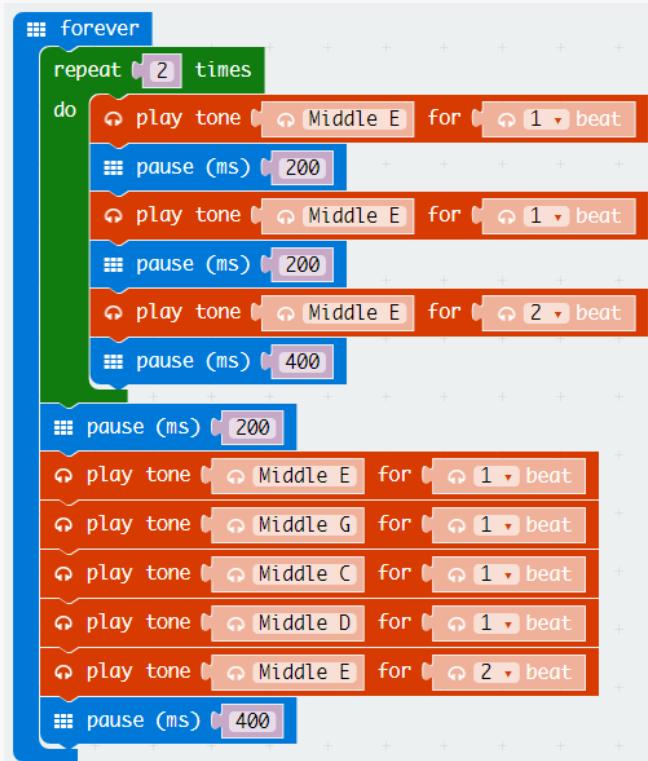
Así que podemos aplicar la función de "repeat" para hacer el trabajo.

Como hemos aprendido antes, el Mi Medio es igual a la nota 3(Mi). Así que podemos programar las dos primeras frases de la siguiente manera:



Puede añadir la función "pause" para que Max:Bot sea más profesional.

Según las frases de "Jingle Bells", la música completa va de la siguiente manera:



(3) Programar el efecto luminoso correspondiente

En este paso, aprenderemos cómo convertir las dos tiras de luz en dos "Tiras de Led Fluidas".

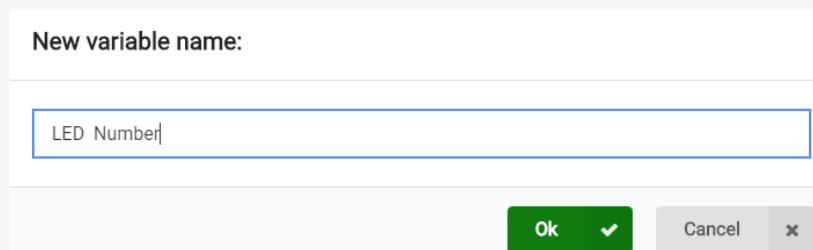
Antes de comenzar, debemos conocer que es la Tira Led de Fluide.

Significa que los leds de la tira se encenderán uno a uno hasta que se enciendan todos los leds.

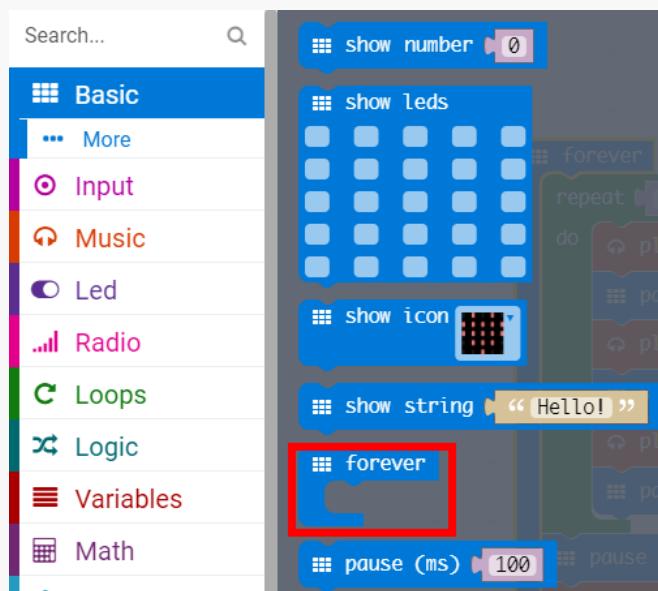


Por lo tanto, el número de LEDs a iluminar debe ser una variable. Por lo tanto, necesitamos crear una variable en primer lugar.

Busque "Make a Variable" a partir de las "Variables" en el área de funciones. Haga clic y nómbrelo como " LED Number ". Haga clic en Aceptar.



Colocar otro bucle "forever" desde el "Basic" en el área de programa.

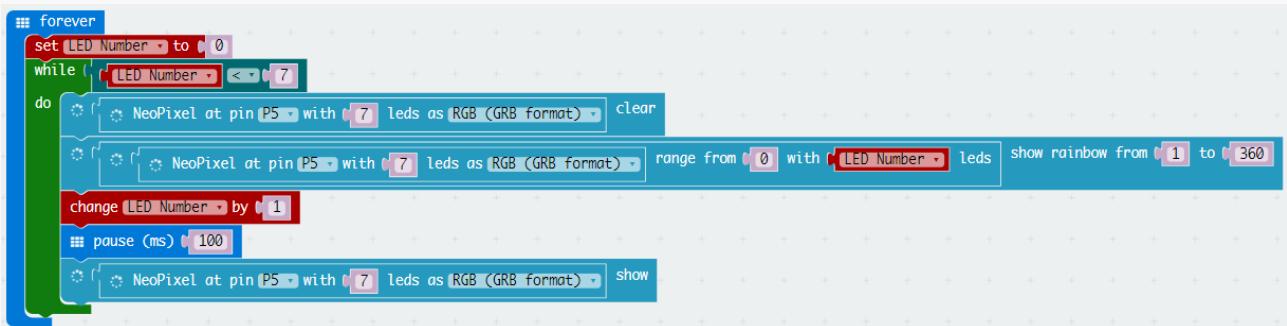


El pin P5 conectado a la tira izquierda viene con 7 leds. Para añadir un efecto de luz arco iris a la tira, necesitaremos establecer el rango de color de 1-360 (el rojo cambia gradualmente a azul). La variable creada "LED Numbers" debe colocarse en la función "range from".



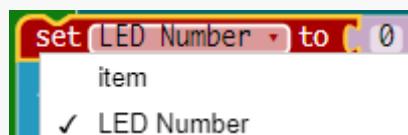
La variable "LED Number" se utiliza para determinar cuántos leds de la banda izquierda se van a iluminar.

Puesto que hay 7 leds en la tira, el número de la variable " LED Number " debe limitarse a 7. El valor inicial es 0.

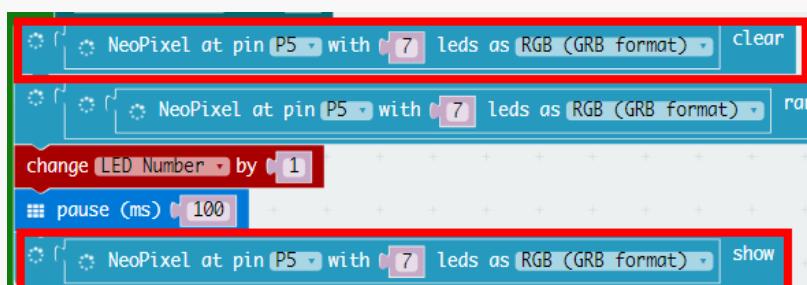


Al programar, debemos tener en cuenta los siguientes aspectos:

- Cuando usamos la función "set to", debemos hacer clic en el menú desplegable (el pequeño triángulo) para seleccionar la variable "LED Number".



- Si está programando la tira de LEDs izquierda, ajuste la clavija como P5. Mientras que para el derecho, el Pin 11.
- Para realizar el efecto de la luz fluida, debemos colocar las funciones "clear" y "show" en el bucle "forever".



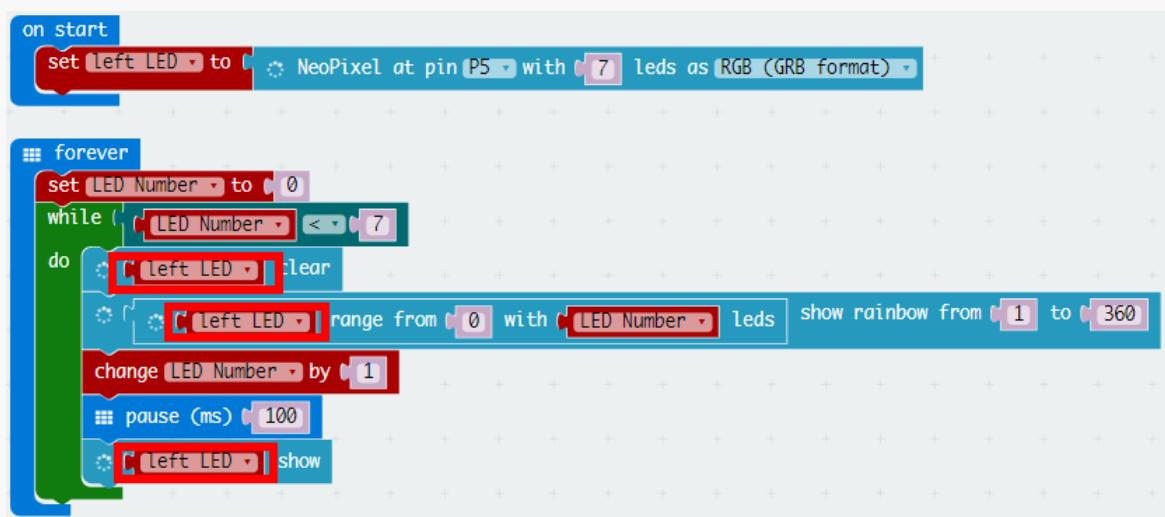
Como podemos ver desde arriba, la función "Neopixel" que controla la tira de LEDs se utiliza tres veces en el programa. Para simplificar el

bucle mayor, podemos configurar un variable "left LED" para reemplazar la función larga "Neopixel".

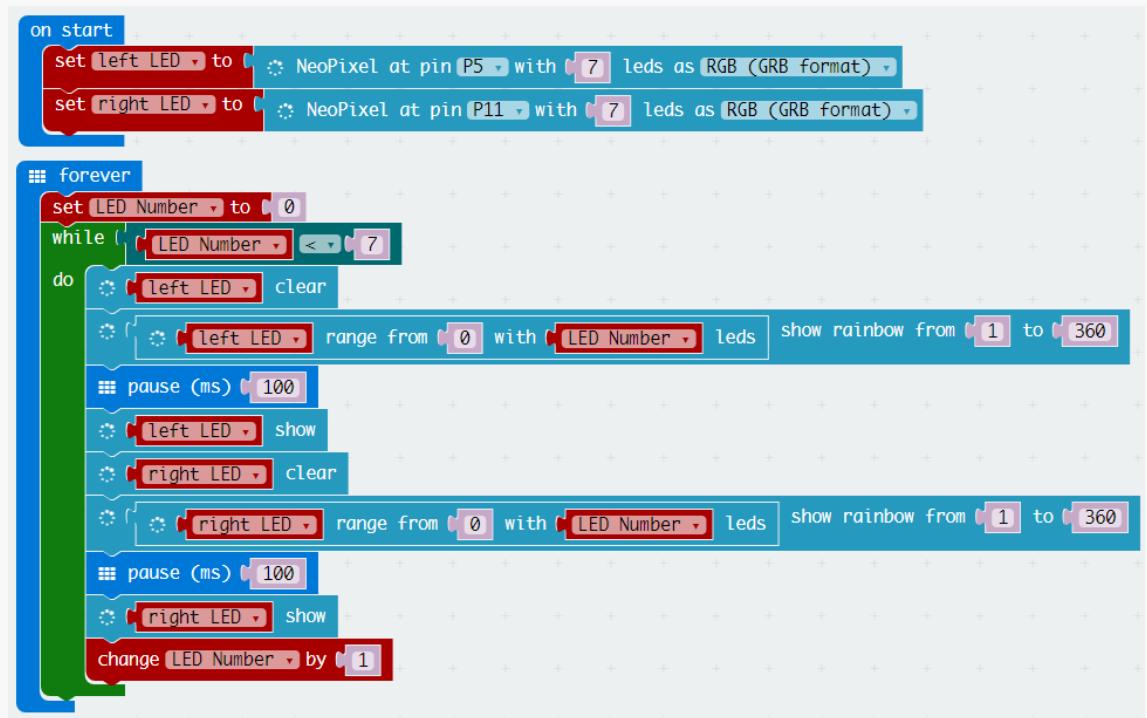
Primero tenemos que crear una variable y darle el nombre de "left LED".

A continuación, coloque el largo "Neopixel" a la derecha de la función "set to". Esto significa que toda la función "Neopixel" será sustituida por la variable "left LED". La función "set to" se encuentra en las "Variables".

Colocar las funciones programadas en el "on start".

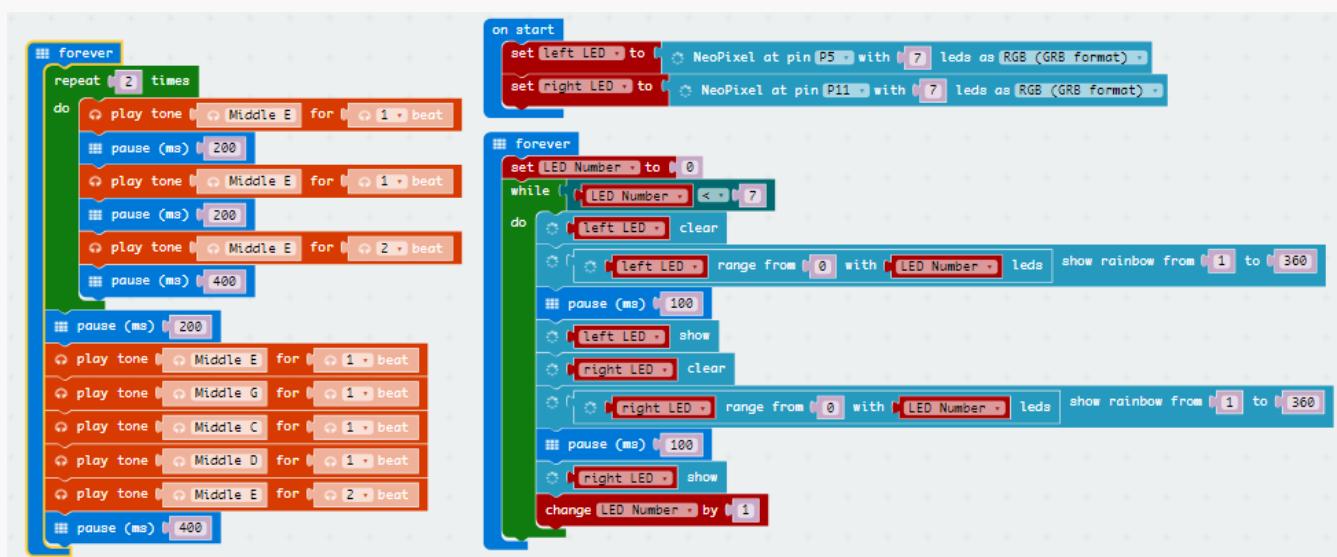


Por lo tanto, podemos aplicar el mismo procedimiento para programar la tira de LEDs derecha. Combinando todas las funciones enumeradas anteriormente, el programa final de efecto de luz fluida será el siguiente:



Arriba hemos aprendido a tocar "Jingle Bells" y a mostrar el efecto de la luz fluida. Para que Max:Bot pueda realizar el espectáculo de luz y sonido, tenemos que combinar todas las funciones listadas en ambos programas juntos. El programa final será el siguiente:

Descargue el programa final a Max:Bot . ¡Ahora él es capaz de reproducir el sonido personalizado y el espectáculo de luz para su próxima Navidad!



Ejercicios

Dado que su robot de Navidad es capaz de reproducir un espectáculo de luz y sonido, ¿por qué no pedirle que salga y envíe regalos para usted?

En esta guía de introducción, acabamos de incorporar algunas funciones básicas de Max:Bot . En realidad, posee habilidades mágicas que nunca son exactamente exploradas. Pero lo que sí creemos es que, a través de sus mentes creativas, todos esos misterios serán eventualmente revelados a todos nosotros. Por favor visite el blog de DFRobot y comparta sus ideas con personas de ideas afines!

Bienvenido al blog de DFRobot!