

# Manual para formadores

Desarrollo de competencias STEM con robótica

«Toda la robótica 4.0»



**Ubicación: Bodø**

**Organización: FIRST Scandinavia**

**12-14 de noviembre, 2019**

## Tabla de contenidos

The EV3 robot (Lesson 15-16)	4
The lobby and how to get around	4
Building and operating the driving base	5
How to make a new program	Error! Bookmark not defined.
The hardware	Error! Bookmark not defined.
The palette menu – green section	6
The palette menu – orange section	7
The palette menu – yellow section	8
The palette menu – red section	8
Parallel programming	Error! Bookmark not defined.
Exercise 1 Driving 100 cm	10
Exercise 2 How can you display the distance the robot moved in the robot display?	10
Exercise 3 Turning 180 degrees	11
Exercise 4 Discuss an algorithm and program	11
Exercise 5 Street-driving / labyrinth driving	12
<i>Reflection</i>	<i>Error! Bookmark not defined.</i>
Introducing sensors (Lesson 17-18)	12
<i>Ultrasonic sensor</i>	<i>Error! Bookmark not defined.</i>
<i>Using the sensor block</i>	<i>Error! Bookmark not defined.</i>
Exercise 6 Street driving with robot using sensor	14
Exercise 7 Street driving with robot using different sensor	14
Variables (Lesson 25-26)	15
<i>Example 1</i>	15
Using the switch	Error! Bookmark not defined.
<i>Example 2 Using a variable to control your robot</i>	17
Exercise 8 Counting numbers using a variable	17
Exercise 9 “Talking robot” using variables and sensors	18
Exercise 10 Counting black lines and show on display	19
Loop interrupt	20
Exercise 11 Counting 10 black lines, show on display and then stop driving	20
<i>Example 3 Using sensors and logic to control your robot</i>	21
Exercise 12 Program your robot to drive and then stop moving when OR condition occurs using two sensors	21

<b>Exercise 13a Reading water level, control motor and display</b>	<b>22</b>
<b>Exercise 13b Reading water level and turn on alarm system</b>	<b>23</b>
<i>Example 4 Math</i>	24
<i>Example 5 Follow-me robot using variable and switch</i>	24
<b>Exercise 14 Controlling the robot with your hand</b>	<b>25</b>
<i>Example 6 Using a constant</i>	25
<b>Exercise 15 Increasing and decreasing speed (Lesson 27-28)</b>	<b>Error! Bookmark not defined.</b>
<i>Example 7 Using My Block on algorithms</i>	27
<b>Exercise 16 Parallel programming using three motors and sensor</b>	<b>27</b>
<i>Example 8 Arrays</i>	28
<b>Exercise 17 Using variables and arrays</b>	<b>29</b>
<b>The palette menu – blue section (Lesson 21-22)</b>	<b>29</b>
<i>Example 9 Bluetooth connection</i>	30
<b>Exercise 18 Alarm system</b>	<b>32</b>
<b>Daisy chaining and EV3 (Lesson 19-20)</b>	<b>32</b>
<b>Data logging (Lesson 23-24)</b>	<b>33</b>
<b>Exercise 19 Velocity</b>	<b>35</b>
<b>Possible solutions</b>	<b>37</b>
<i>Exercise 1</i>	37
<i>Exercise 2</i>	37
<i>Exercise 3</i>	38
<i>Exercise 4</i>	<b>Error! Bookmark not defined.</b>
<i>Making an algorithm</i>	39
<i>Exercise 6</i>	39
<i>Exercise 8</i>	40
<i>Exercise 9</i>	41
<i>Exercise 10</i>	42
<i>Exercise 11</i>	43
<i>Exercise 12</i>	43
<i>Exercise 13a</i>	44
<i>Exercise 13b</i>	44
<i>Exercise 14</i>	45
<i>Exercise 15</i>	45
<i>Exercise 16</i>	46
<i>Exercise 17</i>	47
<i>Exercise 18</i>	48
<i>Exercise 19</i>	48

## El robot EV3 (Lecciones 15-16)

El robot EV3 es un robot de 3 ruedas con ruedas de dirección diferenciales y una rueda de rotación libre para mantener el cuerpo en equilibrio.

El robot EV3 tiene un bloque. Esta es el cerebro del robot y está conectado a todos los motores y sensores a través de puertos. Hay cuatro puertos para sensores y cuatro puertos para motor en el robot. Cuando hacemos un algoritmo, necesitamos recopilarlo y enviarlo al bloque. Después, este usa sus puertos para comunicarse con los motores y sensores usados en el algoritmo. Es importante asegurarse de que los puertos para sensores y motores son correctos. Si hay algún problema con los puertos, podrás ver una señal en el bloque.

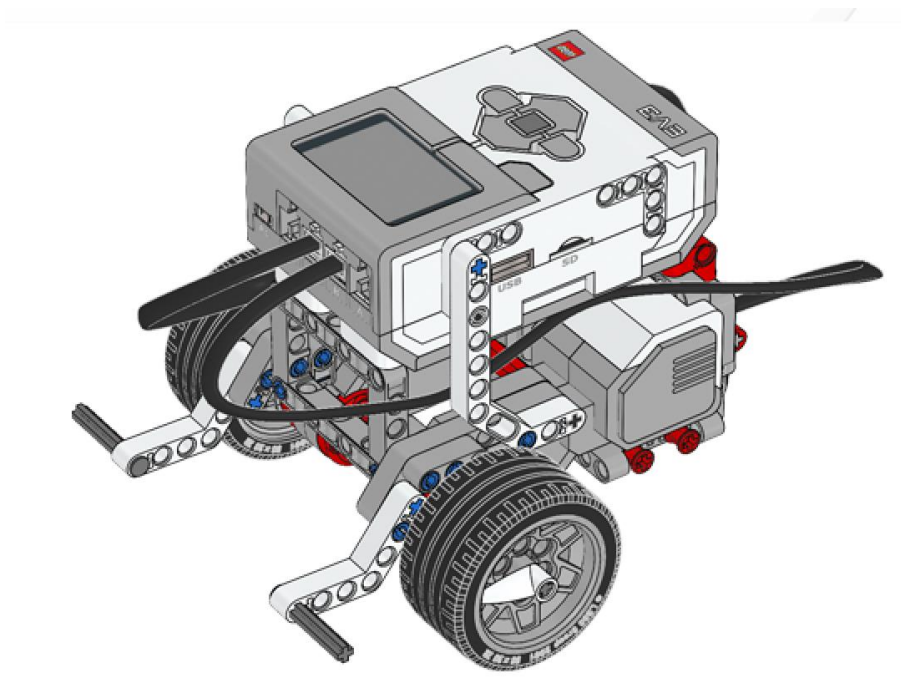
## Pantalla de inicio y forma de moverse

En la pantalla de inicio encontrarás una vista general buena que contiene instrucciones de construcción, lecciones, tutoriales y manuales.



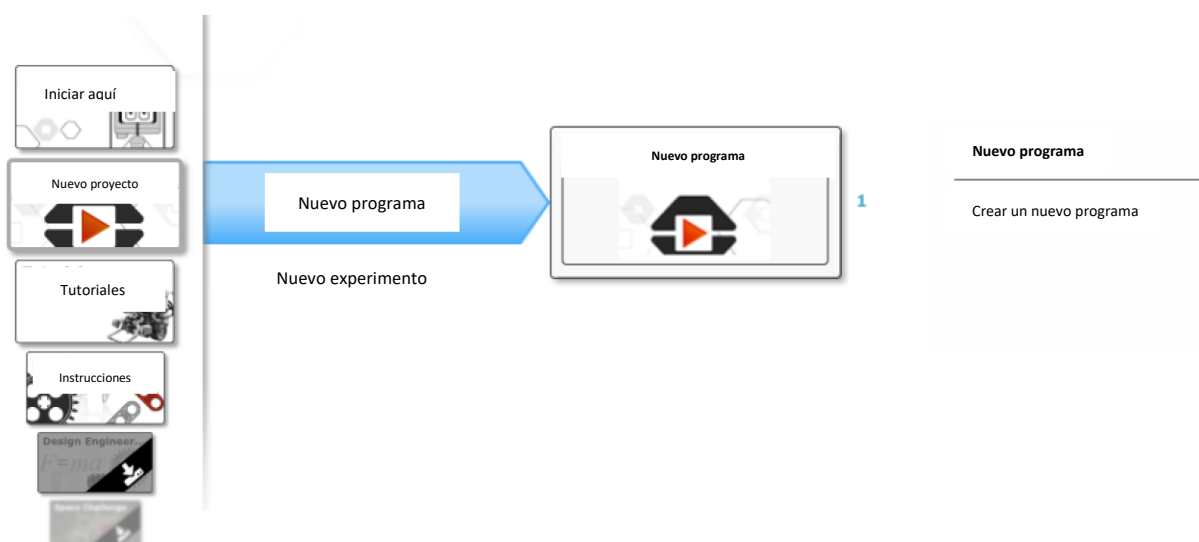
## Construcción y funcionamiento de la base de conducción

Para construir la base de conducción, selecciona las instrucciones en la pantalla de inicio, después, elige ideas de construcción y base de conducción. Para construir esta última, encontrarás una guía paso a paso.



## Cómo crear un programa nuevo

Para empezar, elegimos un nuevo proyecto y programa, o simplemente pulsamos el botón + en la esquina superior izquierda.

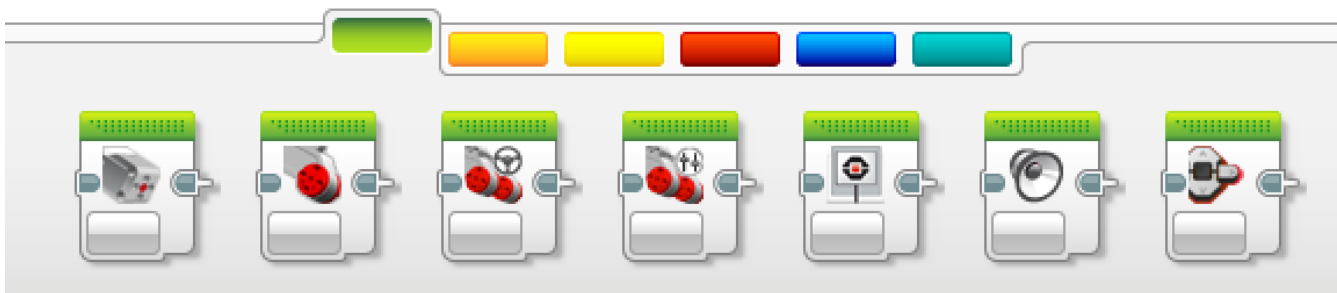


## El hardware

En la zona de hardware, puedes encontrar la información cuando se conecte al ordenador o teclado. Ahí es donde guardas y envías tus programas a tu bloque, usando la flecha de descargar o la flecha de descargar y empezar. También puedes introducir archivos y borrarlos o renombrarlos. Puedes conectar el bloque usando USB, wifi o Bluetooth.



## Menú de la paleta – sección verde

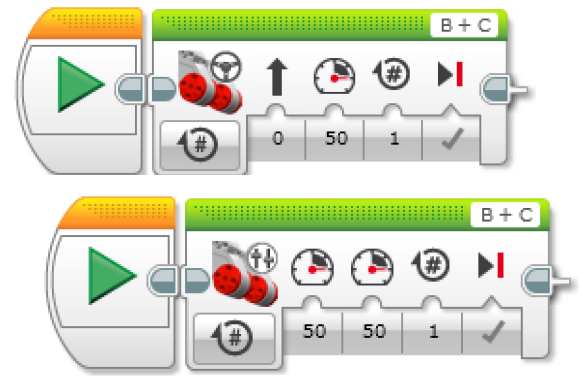


En la sección verde puedes encontrar los bloques de motor, sonido, pantalla y bloques de luz de estado. Vamos a ver el bloque de dirección de movimiento y el bloque de tanque de movimiento.



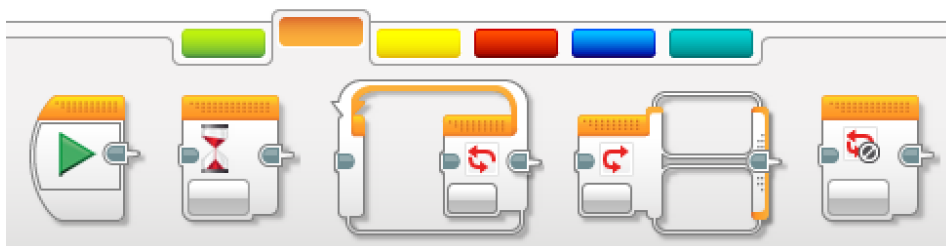
En el menú izquierdo en el bloque de dirección de movimiento puedes elegir encender o apagar el motor, usar el motor unos segundos, los grados o los giros. Vamos a usar todas estas alternativas durante nuestro curso.

Puedes cambiar la dirección, la velocidad y el número de giros. La diferencia entre el bloque de dirección de movimiento y el bloque de tanque de movimiento es que, para el segundo, puedes poner un movimiento diferente en cada rueda.



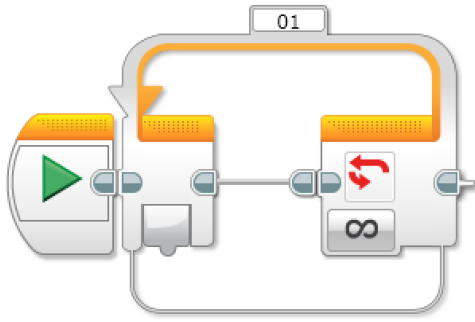
El bloque indicador se puede usar para mostrar resultados de los bloques matemáticos, sensores, medidas o comparaciones. Este bloque puede ser también un texto, un número, una imagen, formas diferentes o reinicio. Si queremos transferir un resultado de un bloque diferente a la pantalla, tenemos que elegirlo en el menú superior derecho (donde pone neutro), esto debe transferirse.

## Manú de la paleta – sección naranja



En esta sección encontramos bloques de control de flujo como empezar, esperar, ciclo, cambiar e interrumpir. En este curso todos estos bloques serán importantes.

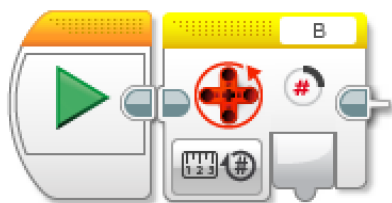
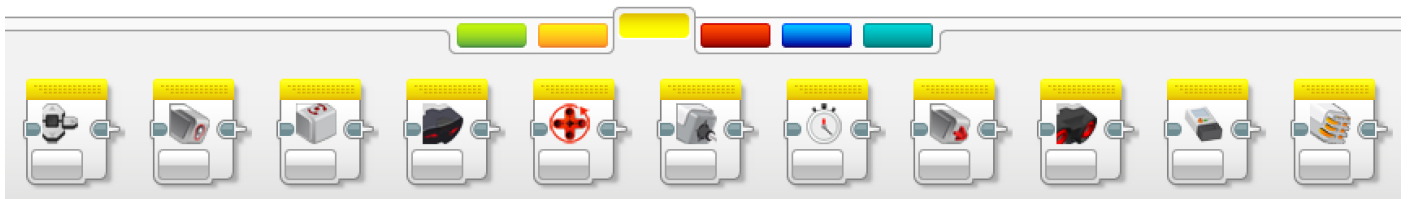




El bloque de ciclo se usa cuando queremos que un programa funcione más de una vez. Podemos hacer que funcione sin límites, con lógica o usando un sensor. Tenemos otro bloque llamado interruptor de ciclo, que se usa para parar otro ciclo si estamos

trabajando en programación paralela. Los otros bloques de control de flujo se introducirán cuando necesitemos usarlos en este curso.

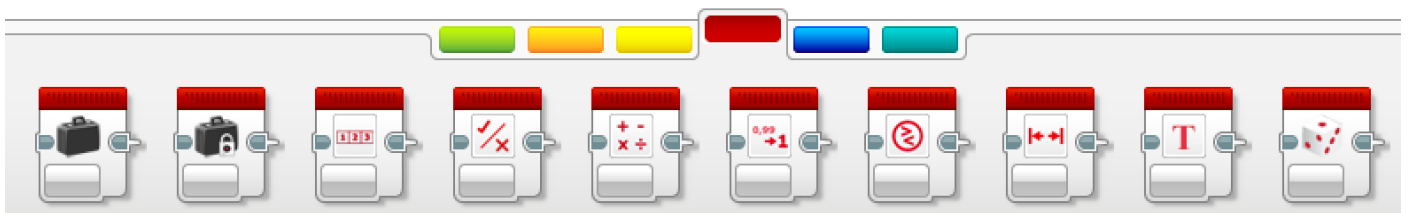
### Menú de la paleta – sección amarilla



En la sección amarilla, puedes encontrar todos los bloques de sensor. Estos se pueden usar para conseguir valores y transferirlos a las matemáticas, la lógica y comparación de bloques o variables. Volveremos más tarde. En el dibujo

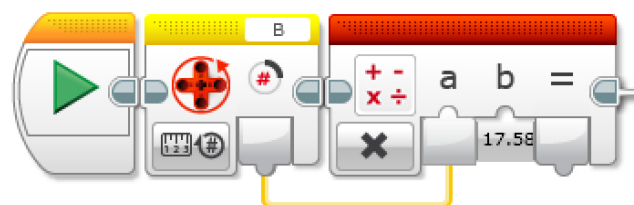
de la izquierda puedes ver un bloque de rotación. Se puede usar para medir o comparar los giros que realiza el motor.

### Menú de la paleta – sección roja



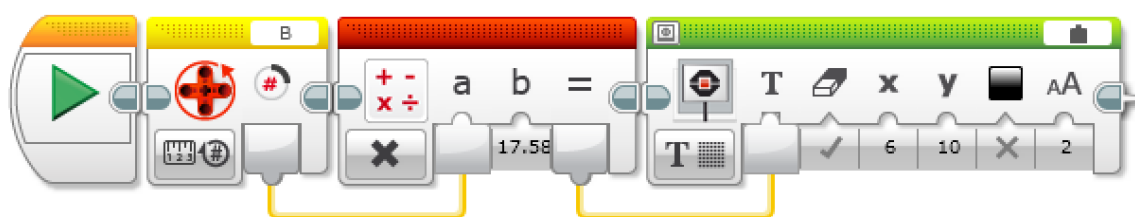


En la sección roja están todos los bloques matemáticos, variables, lógicos y de comparación. En el ejemplo de la derecha, usamos un bloque de medida de giros y



transferimos (enviamos) el resultado a un bloque matemático. Seleccionamos multiplicar en el menú izquierdo del bloque. El número de giros será el valor de a, y entonces ponemos el número de b. (En este caso, el perímetro de las ruedas es 17.584 cm)

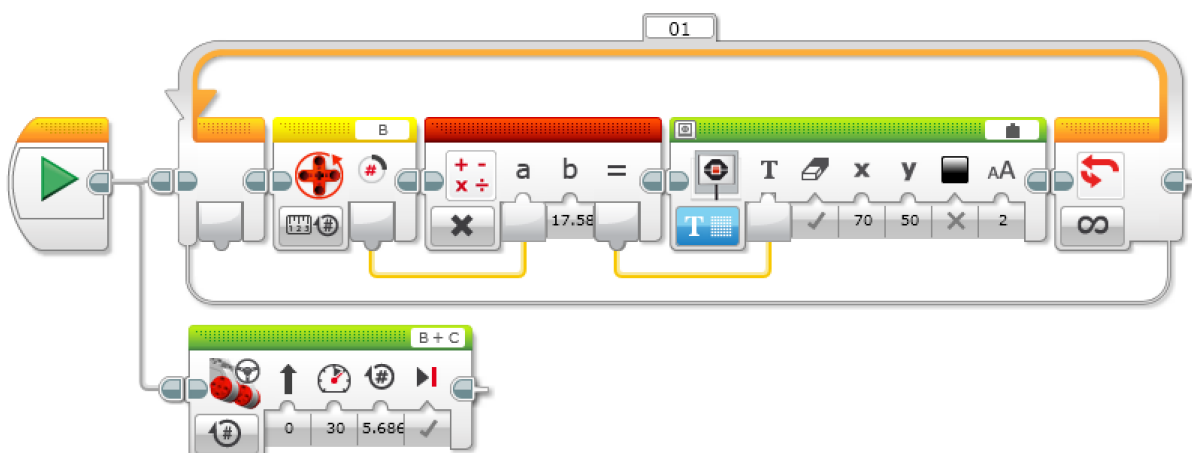
Si quieres mostrar el resultado en la pantalla del robot, necesitamos transferir el resultado (=) al bloque de pantalla. El valor de x e y determina dónde debe aparecer en la pantalla el texto.



Para medir de manera continua tenemos que poner estos bloques en un ciclo. En este ejemplo también es importante que tenemos un algoritmo que activa los motores delanteros para poder medirlo todo.

### Programación paralela

Debemos usar programación paralela. Para hacer funcionar bloques diferentes a la vez, transferimos desde el bloque de inicio.



## Ejercicio 1: conducción de 100 cm

*Usando: bloque de tanque de movimiento y de dirección de movimiento*

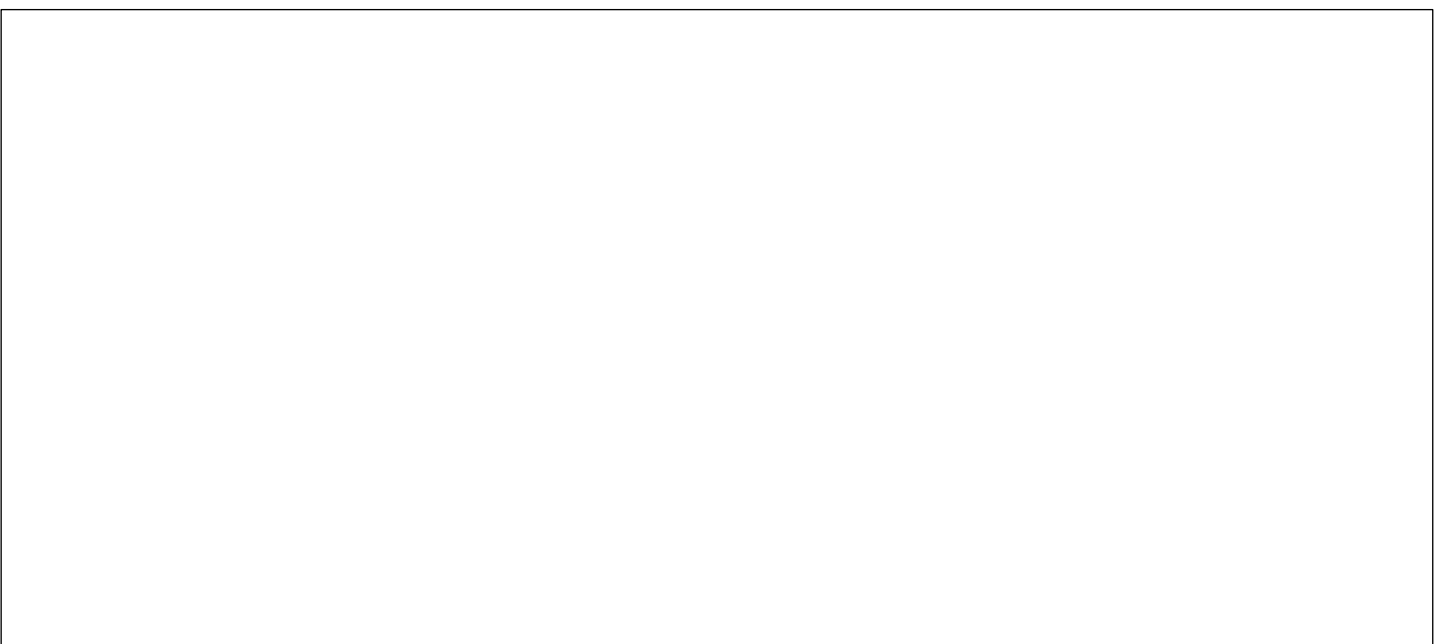
Haz que tu robot recorra exactamente 100 cm calculando y usando el perímetro de su rueda. El diámetro de la rueda estándar es de 56 mm. Usa el bloque de dirección de movimiento y el de tanque de movimiento en dos programas diferentes para solucionarlo. ¿Cuál es la diferencia?



## Ejercicio 2: ¿cómo puedes indicar la distancia que ha recorrido el robot en su pantalla?

*Usando: sensores, matemáticas, ciclo, pantalla y programación paralela*

¿Puedes hacer algunos cambios al programa para mejorarlo aún más?



### Ejercicio 3: giro de 180 grados

*Usando: bloque de tanque de movimiento, bloque de dirección de movimiento y motor grande*

Gira 180 grados usando ambos bloques. También se puede usar un motor grande para girar. Discútelo con tu compañero. ¿Cuál es la diferencia entre estos dos/tres métodos?

### Ejercicio 4: discusión sobre un algoritmo y un programa

*Usando: bloque de dirección de movimiento, bloque de tanque de movimiento y ciclo*

Ya has programado el robot para que recorra 100 cm y has utilizado diferentes maneras de girar el robot. ¿Puedes crear un algoritmo para hacer que el robot se mueva en un cuadrado 100cm? Discútelo con tu compañero y escribe una solución antes de programar.

## Ejercicio 5: conducción urbana / conducción por un laberinto

*Usando: bloque de dirección de movimiento, bloque de tanque de movimiento y ciclo*

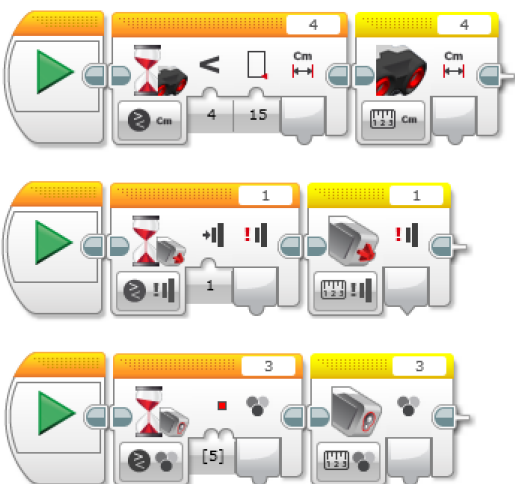
Observa el laberinto. Discute con tu compañero cómo puedes crear un programa que use el robot desde el principio hasta el final sin ninguna intervención humana. Usa los bloques que hemos aprendido hasta ahora.

### Reflexión

¿Qué método y qué bloques crees que funcionan mejor en este ejercicio?

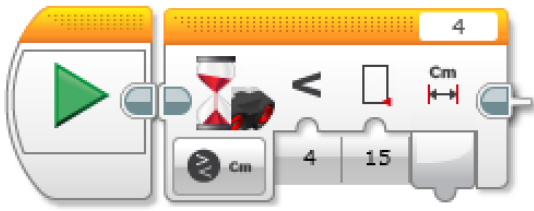
### Introducción de los sensores (lecciones 17-18)

El robot EV3 incluye un sensor ultrasónico, uno táctil, uno de color y un giroscopio. En este curso vamos a usar los tres primeros sensores solucionando retos diferentes.



#### Sensor ultrasónico

Primero, vamos a echarle un vistazo al bloque de espera (color naranja). Este bloque se puede usar para esperar algo, antes de continuar al siguiente bloque de la secuencia. Puedes esperar un tiempo determinado, un cierto valor para un sensor o unos valores de los sensores para cambiar.

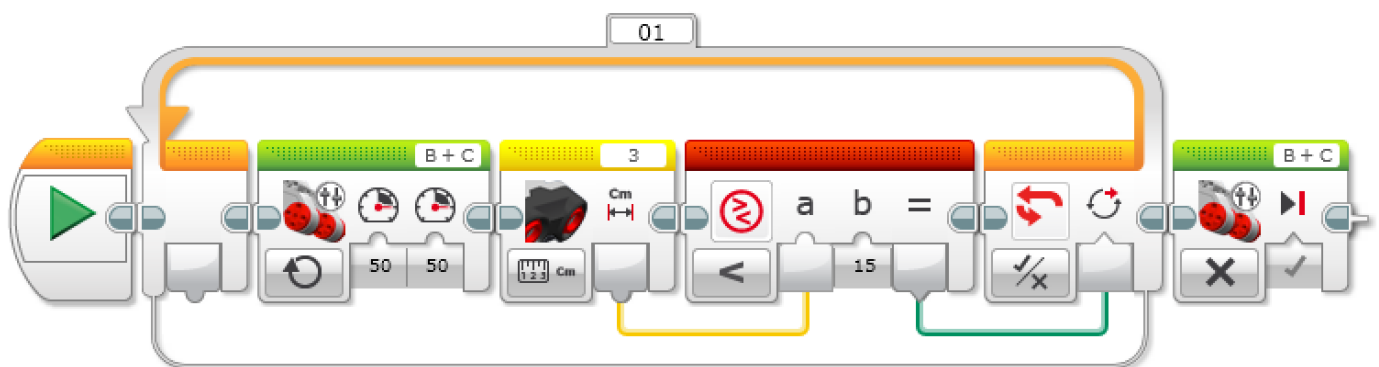


A la izquierda del bloque de espera está la selección de modos. En este caso, elegiremos el sensor ultrasónico. Este sensor se puede usar de diferentes maneras, pero ahora vamos a seleccionar «menor de» y «15 cm». Esto significa que el bloque de espera esperará hasta que suceda «menos de 15 cm».

**Pista:** El bloque de espera no detendrá tu robot. Si hay algún bloque motor funcionando delante del bloque de espera, seguirá funcionando a través de este.

### Usando el bloque de sensores

Para medir, comparar y usar más de un sensor, es muy útil el bloque de sensores. Por ejemplo, nos gustaría programar nuestro robot para que se moviera hasta que el sensor ultrasónico midiera algo menor de 15 cm. Usando el bloque de sensor y el bloque de comparación, y poniéndolo en un ciclo que acaba en lógica, hay una solución.



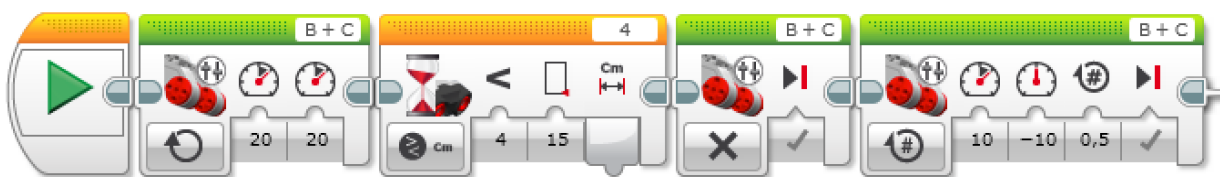
Pero, en este caso también podríamos usar el bloque de espera como el siguiente programa. Esta es una manera más fácil de programar el robot para que haga las mismas operaciones.



### Ejercicio 6: usando un sensor en la conducción urbana con el robot

*Usando: bloque de tanque de movimiento, bloque de espera con sensor, bloque de sonido, bloque de pantalla y programación paralela*

Programa tu robot usando el bloque de espera con un sensor ultrasónico. El robot debería poder moverse por el laberinto utilizando motores y el sensor ultrasónico. El robot también debería hacer un sonido (izquierda/derecha) cuando gira. El programa también debería indicar la distancia que ha recorrido todo el tiempo.



¿Es posible solucionar el laberinto usando uno de los otros sensores?

### Ejercicio 7: usando un sensor diferente en la conducción urbana con el robot

*Usando: bloque de tanque de movimiento, bloque de espera con sensor, bloque de sonido, bloque de sensor, bloque de pantalla y programación paralela*

Programa la base de conducción utilizando sensores, ciclos e interruptores.

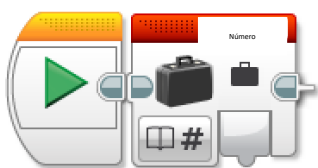
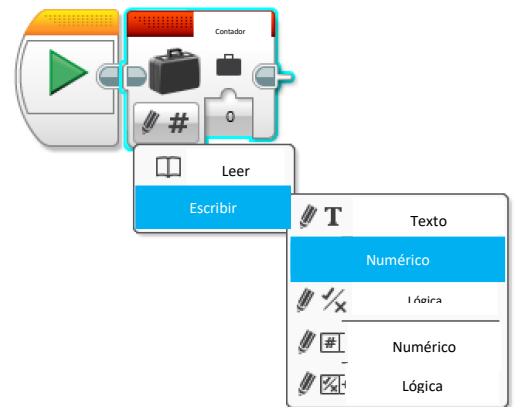
## Variables (lecciones 25-26)

Se puede ver una variable como un «estuche», como el bloque que puede contener números, texto o enunciados de verdadero o falso. También se puede tomar como un conjunto con diferentes datos. Una variable se llama así porque el contenido del estuche puede cambiarse durante la ejecución del programa.

Usar una variable es muy fácil, ya que solo hay tres cosas que podemos hacer con la variable:

1. Definir la variable
2. Escribir datos en la variable
3. Leer datos de la variable

En primer lugar, tenemos que añadir una variable nueva. Lo haremos en el bloque de la variable en la esquina superior derecha. Llamamos variable a lo que contiene (por ejemplo, en este caso, «número») y debemos elegir si la variable debe contener un texto, un número o lógica.



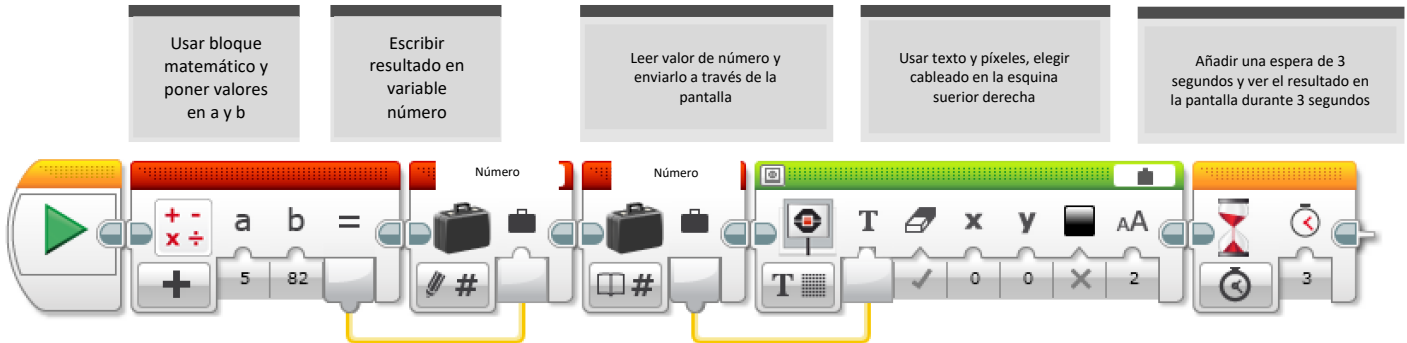
Para conseguir el contenido del bloque de la variable, tenemos que utilizar el bloque de la variable leída. Es importante asegurarnos de que usamos el mismo tipo de contenido (número, texto y lógica)

### Ejemplo 1

En este simple programa usamos el bloque matemático haciendo sumas y poniendo un número para a y b. En este caso 5 y 89. Después, escribimos el resultado de la suma de 5 y 82 en la variable «número».

Para mostrar el resultado, tenemos que poner un bloque de pantalla en nuestro programa. Para ello, usamos la variable «número» en el modo lectura y lo transferimos al bloque de pantalla. Así, tenemos que

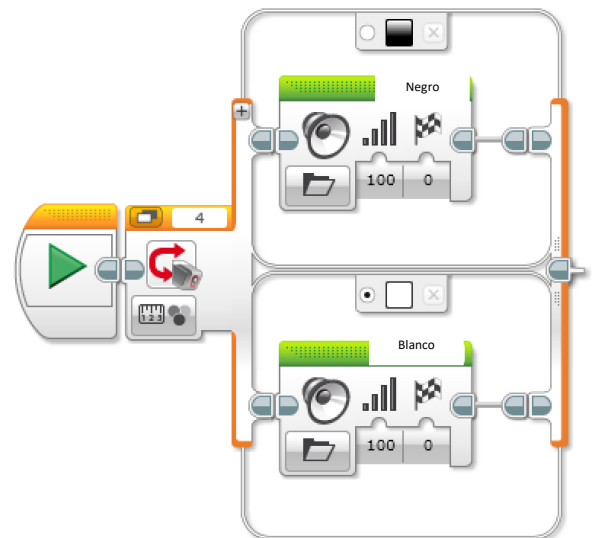
elegir texto y píxeles, y después elegimos «transferido» en la esquina superior derecha del bloque de pantalla. El último bloque es una espera de tres segundos. La razón por la que lo ponemos es que el resultado permanece en pantalla durante tres segundos.



## Uso del interruptor

Usar el interruptor hace más avanzado nuestro programa. Este es más conocido como programación condicional.

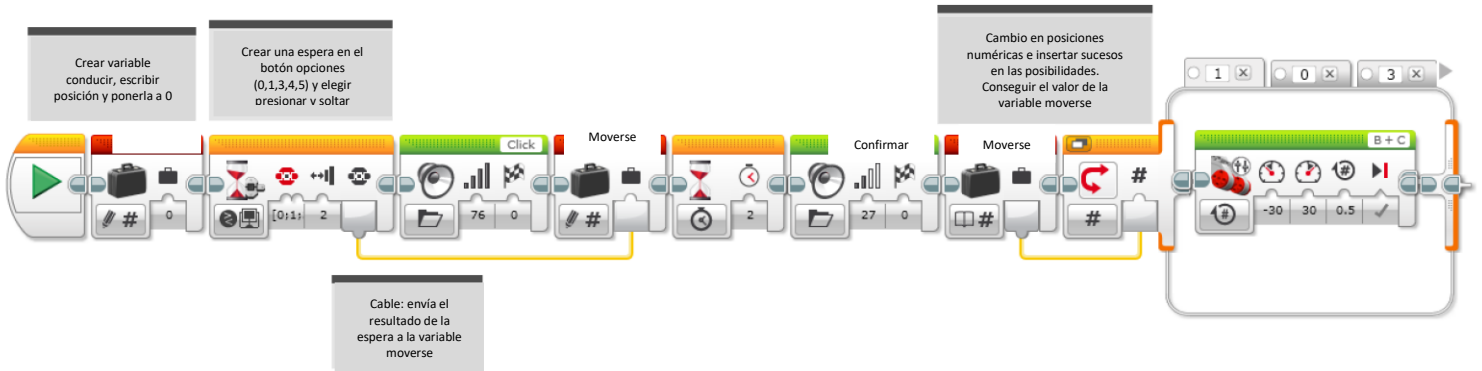
Podemos usarlo para realizar diferentes operaciones para cada color que registra. En el siguiente ejemplo, el programa hará sonar «negro» cuando el sensor de color detecte el color negro y el sonido «blanco» cuando detecte el color blanco. Es posible añadir al interruptor más de solo dos opciones.





## Ejemplo 2: usar una variable para controlar tu robot

Crea una variable «conducción». Después, añade un bloque de espera y elige los botones. Pon un sonido. Escribe el nuevo valor de un bloque de espera transmitiéndolo a la variable «conducción». Espera un segundo. Después, haz sonar un sonido una vez más y lee la variable, transfiriéndola al interruptor que tenga tantas salidas como bloques (0,1,3,4,5). Ponlo en el movimiento correcto para cada número.



## Ejercicio 8: contar números usando una variable

Usando: Variable, matemáticas, espera, pantalla, interruptor y ciclo

Queremos crear un programa que cuente el número de veces que el botón superior de las piezas se pulsa y muestra el número en la pantalla.

Pista:

Usa variable numérica

Usa el interruptor

Usa el ciclo

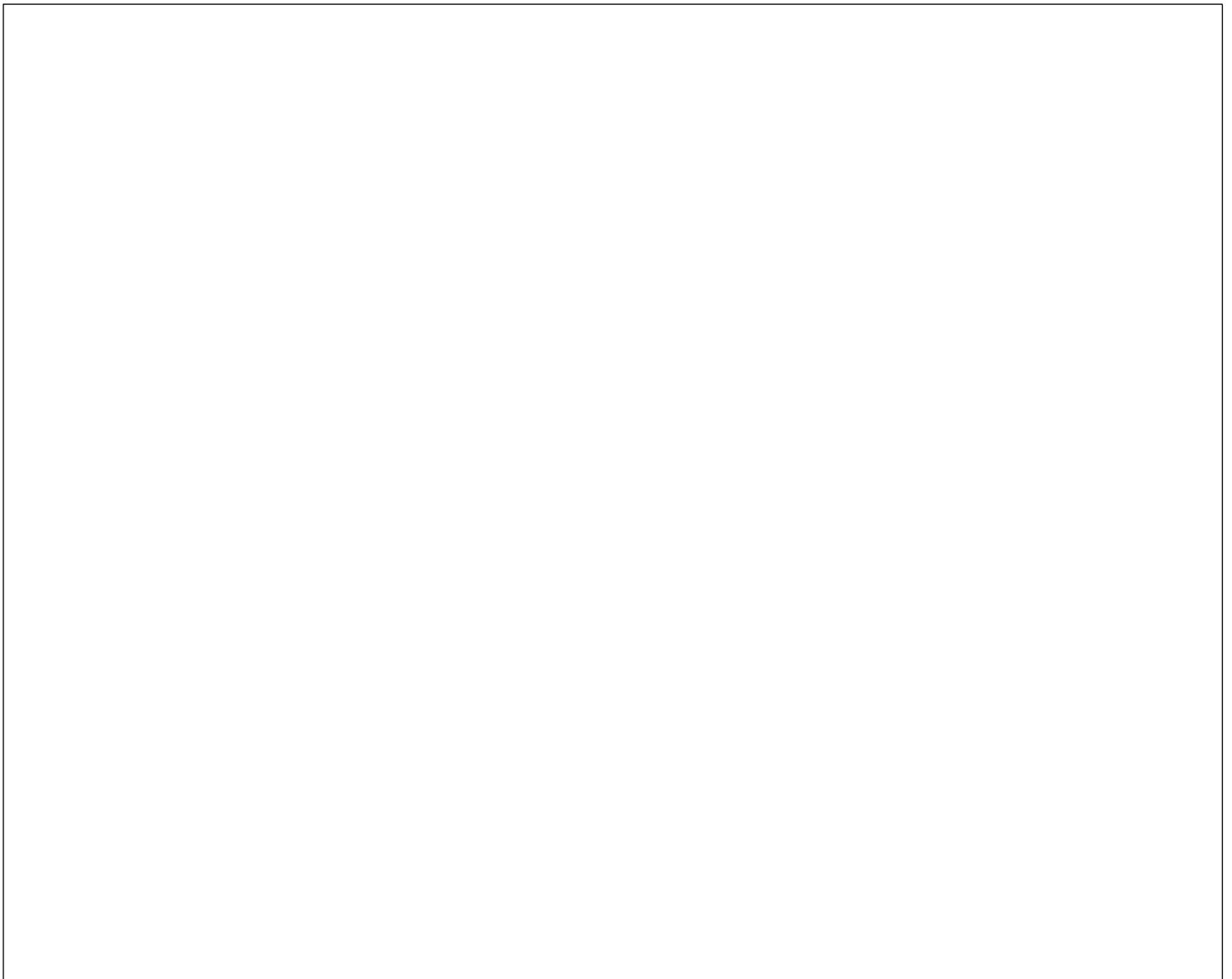
Usa la pantalla

## Ejercicio 9: «robot hablador» usando variables y sensores

*Usando: Variable, espera, pantalla, interruptor y ciclo*

Vamos a programar nuestro robot para que nombre los colores azul, verde, rojo, amarillo y negro. Usamos el mismo método que antes, pero ahora la diferencia es que usamos la variable con texto.

El programa debe hacer lo siguiente: cuando pulsa el botón del centro de la pieza, el programa se inicia y el sensor empieza a registrar colores. Cuando pones el color azul delante del sensor, el robot muestra la palabra «azul» en la pantalla y también reproduce el sonido de la palabra. El programa debería funcionar cinco veces y después terminar.



## Ejercicio 10: contar líneas Negras y mostrarlo en pantalla

Usando: programación paralela, bloque de espera, variables, matemáticas, sensores, interruptor y ciclo

Programa el robot para que se mueva y a la vez haga la cuenta de cuántas líneas negras pasa. Este número debe aparecer en la pantalla del robot.

Pista:

Crea una variable

Usa la programación paralela para hacer que el robot se mueva y cuente a la vez

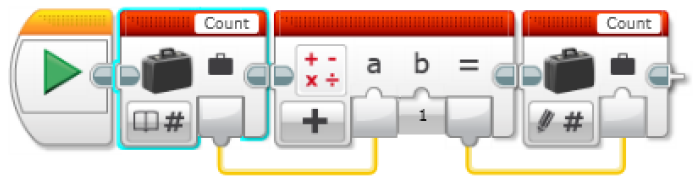
Usa el bloque matemático para sumar 1 a la variable cada vez

Usa cables para recibir y enviar información entre bloques

Usa el bloque de pantalla para mostrar los números

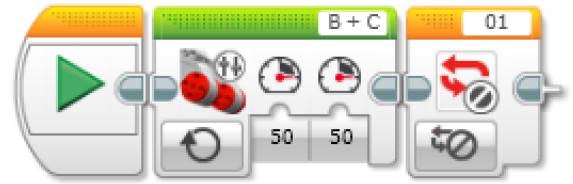
Enciende el sensor de color

Usa el ciclo para repetir



## Interrupción del ciclo

Usando la programación paralela, necesitar usar el interruptor de ciclo para detener todos los ciclos en nuestros algoritmos. Si elegimos no utilizarlo, el ciclo continuará funcionando hasta que apaguemos el programa.



## Ejercicio 11: contar 10 líneas negras, mostrarlo en pantalla y detenerse

*Usando: programación paralela, interruptor, variables, pantalla, espera, matemáticas, comparación e interruptor de ciclo*

Continúa para usar el programa del ejercicio 10. Queremos introducir algunos bloques extra para hacer que el robot deje de moverse y cuente después de pasar 10 líneas.

Pista:

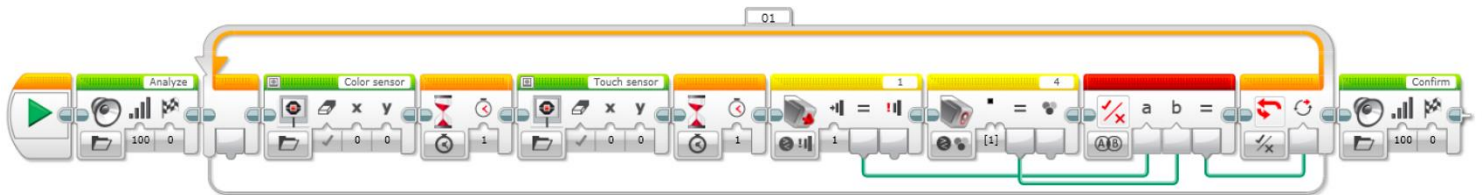
Bloque de comparación

Bloque de variable

La lógica pone fin al ciclo con el interruptor de verdadera o falso

### Ejemplo 3: usar sensores y lógica para controlar tu robot

Observemos cómo podemos programar un robot para que sea dependiente de dos sensores para alcanzar una condición. (se pulsa el sensor táctil y se detecta el color negro). Después, las dos condiciones deben suceder para hacer que el robot confirme. En este programa usamos la condición de AND en el bloque lógico, lo que quiere decir que ambos sensores tienen que ser afectados para hacer que el robot haga su sonido de confirmación.



### Ejercicio 12: programa tu robot para que se displace y luego deje de hacerlo cuando ocurra la condición OR usando dos sensores

*Usando: motor, sensores, lógica y salida del ciclo*

Pista:

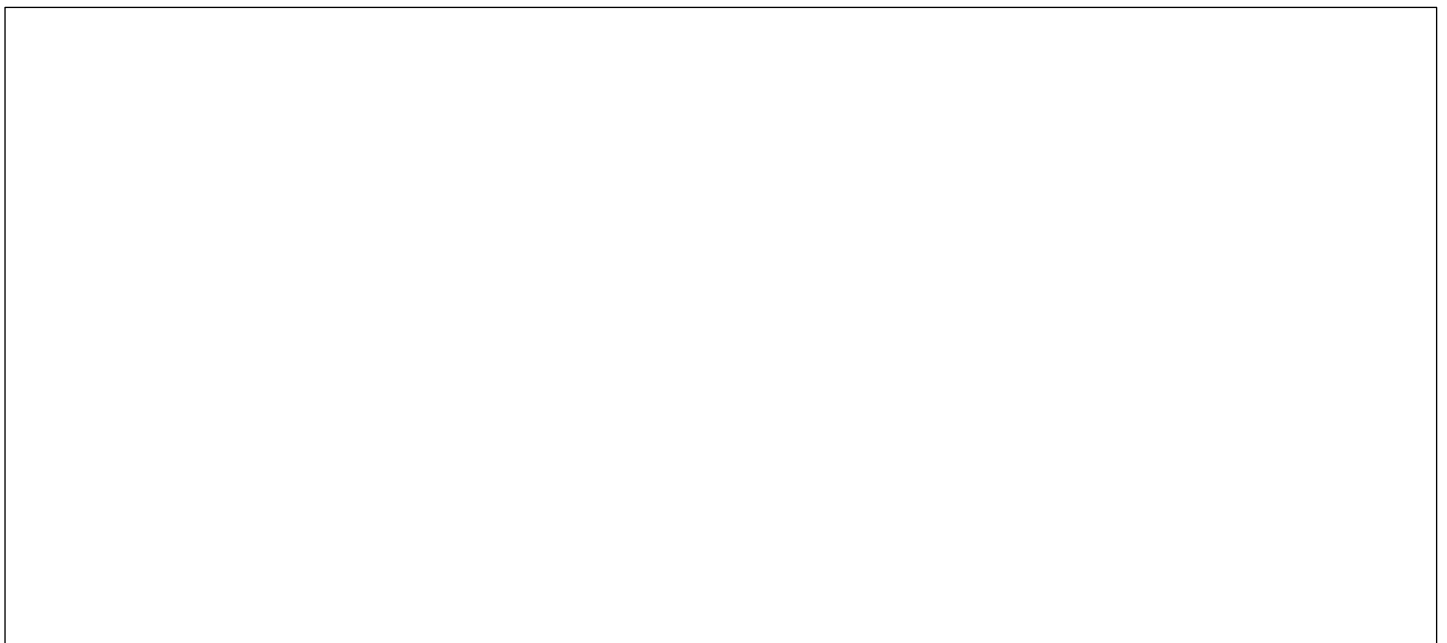
Bloques motores para encender el robot

Sensor táctil y sensor de color

Bloque lógico y ponlo en modo OR

Acaba la lógica en el ciclo

Transferencia entre sensores, lógica y ciclo

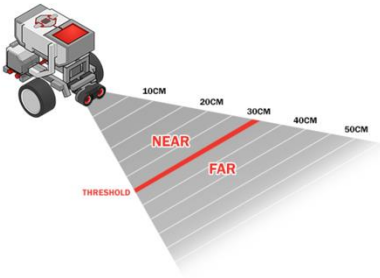


## Ejercicio 13a: lectura del nivel del agua, motor de control y pantalla

*Usando: variable, sensor ultrasónico, comparación, pantalla, interruptor, ciclo y luz de estado*

### Supuesto

Tenemos un tanque con agua al que durante el día le entra y le sale agua por los tubos.



Usando la base de conducción con un sensor ultrasónico, vamos a hacer que un Sistema «lea» el nivel de agua y se asegure de que este nunca exceda un límite.

Cuando el nivel del agua sea demasiado alto, el sistema encenderá un motor y abrirá un hueco, por lo que se saldrá más agua.

El sistema también tendrá una pantalla mostrando el valor del nivel del agua de manera continua. Colocamos el robot a 30 cm de la pared/ caja, etc. El tanque está entre el robot (sensor) y la pared. La luz será verde cuando el nivel esté bien. Cuando el valor esté por debajo del 10 y el motor A empiece a funcionar, la luz se pondrá roja.

### Pista:

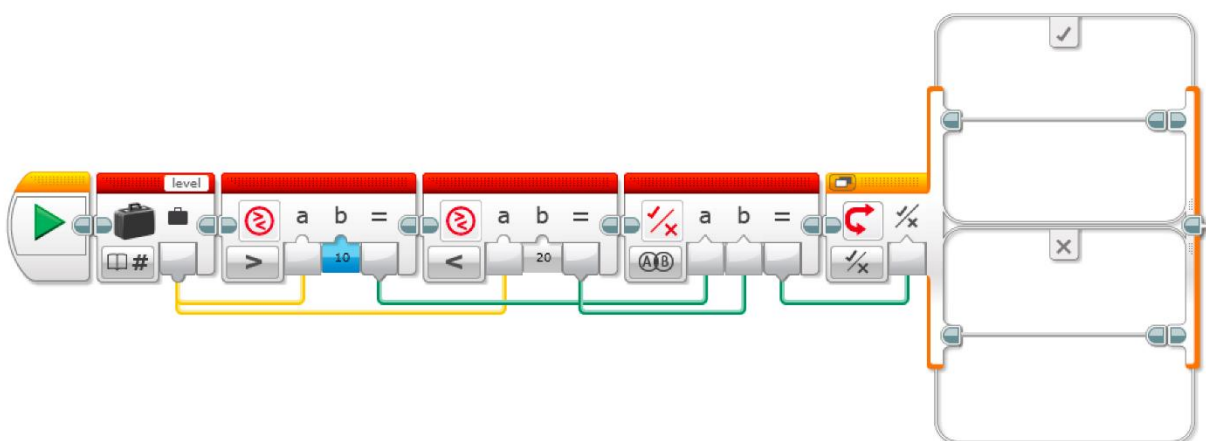
Variable «nivel»  
Medida con el sensor ultrasónico  
Comparación de la variable del sensor  
Luz de estado  
Pantalla  
Interruptor  
Ciclo

## Ejercicio 13b: lectura del nivel del agua y encendido del sistema de alarma

Usando: *variable, sensor ultrasónico, comparación, pantalla, interruptor, ciclo, estado de luz, lógica AND y programación paralela*

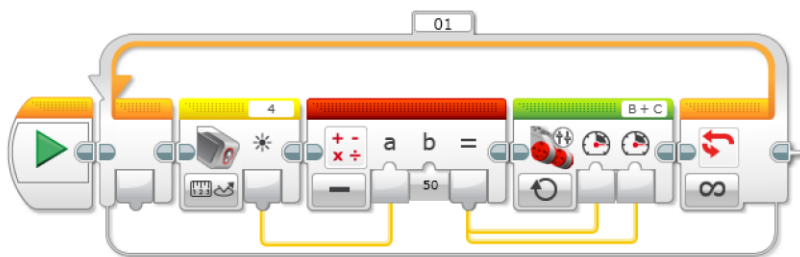
Usando el mismo programa del último retro, vamos a desarrollar ahora un tipo de sistema de alarma que encenderá una alarma si el valor del agua está entre 10 y 20. Usamos el mismo programa y hacemos un algoritmo paralelo que contenga los bloques de alarma

Ya que estamos usando el nivel del agua entre 10 y 20, necesitamos usar un bloque de lógica que sea verdadero para ambos «mayores que 10» y «menores que 20». Solo en este supuesto, en el que sonará la alarma, necesitamos usar un interruptor.



## Ejemplo 4: matemáticas

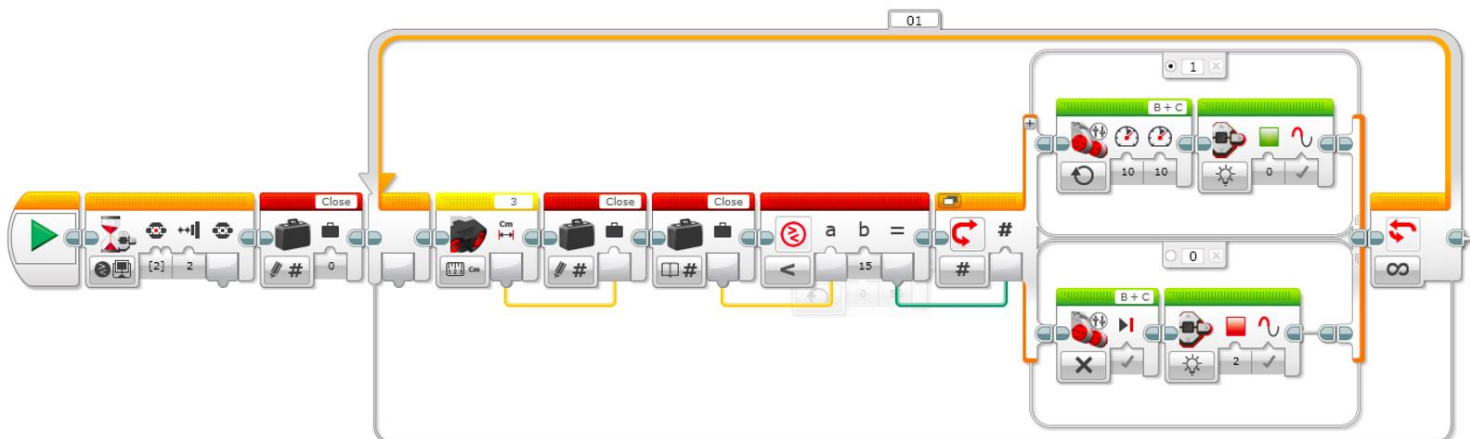
¿Qué hará el siguiente programa?



Discute con compañero. Después, programa y prueba.

## Ejemplo 5: robot sígueme usando una variable y un interruptor

Programa tu robot para empezar con un clic en uno de los botones y después seguir tu mano. Esto lo hará moviéndose cuando tu mano esté a menos de 15 cm de él y parando cuando esta esté a más de 15 cm.

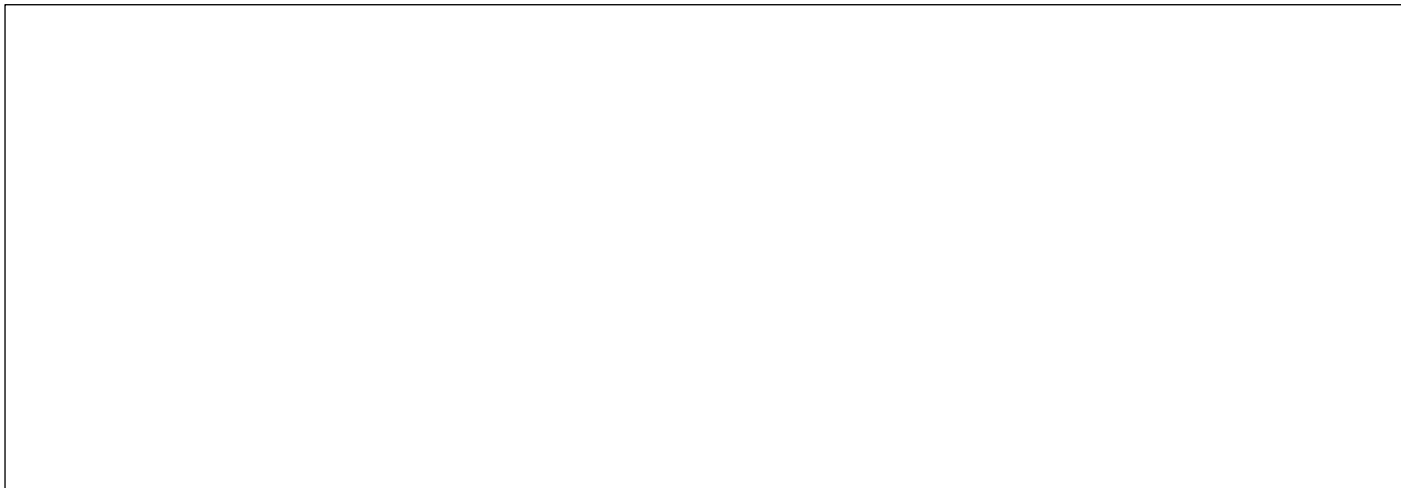


Discute con tu compañero. ¿Cómo Podemos resolver este reto de una manera diferente? ¿es posible crear un programa sin variables?



## Ejercicio 14: controlar el robot con tu mano

Ahora intentemos hacer este ejercicio de forma diferente. Si tu mano está a menos de 20 cm del sensor ultrasónico, el robot se moverá hacia atrás con la luz de estado roja. Si tu mano está a más de 20 cm de dicho sensor, el robot se moverá hacia atrás con la luz de estado verde.

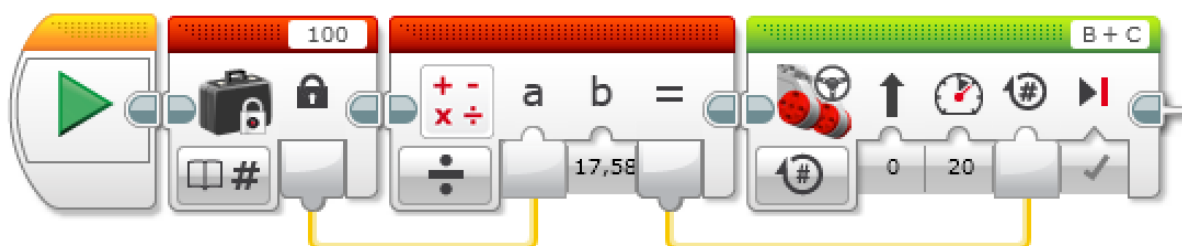


## Ejemplo 6: usar una constante

Una constante es un valor que el programa no puede alterar durante la ejecución normal. El valor es una constante. Al contrario que una variable, que se identifica por cambiar durante la ejecución normal, la constante permanece intacta durante la ejecución.

Usar una constante puede ayudarnos a mover el robot una distancia exacta de 100 cm. Por ejemplo, en el siguiente programa, tenemos una constante determinada de «100». Después, usamos un bloque matemático para dividir la constante con el perímetro de la rueda y encontramos los giros correctos que tiene que hacer el robot para recorrer 100 cm.

Mira el siguiente ejemplo e intenta averiguar qué hará el programa.

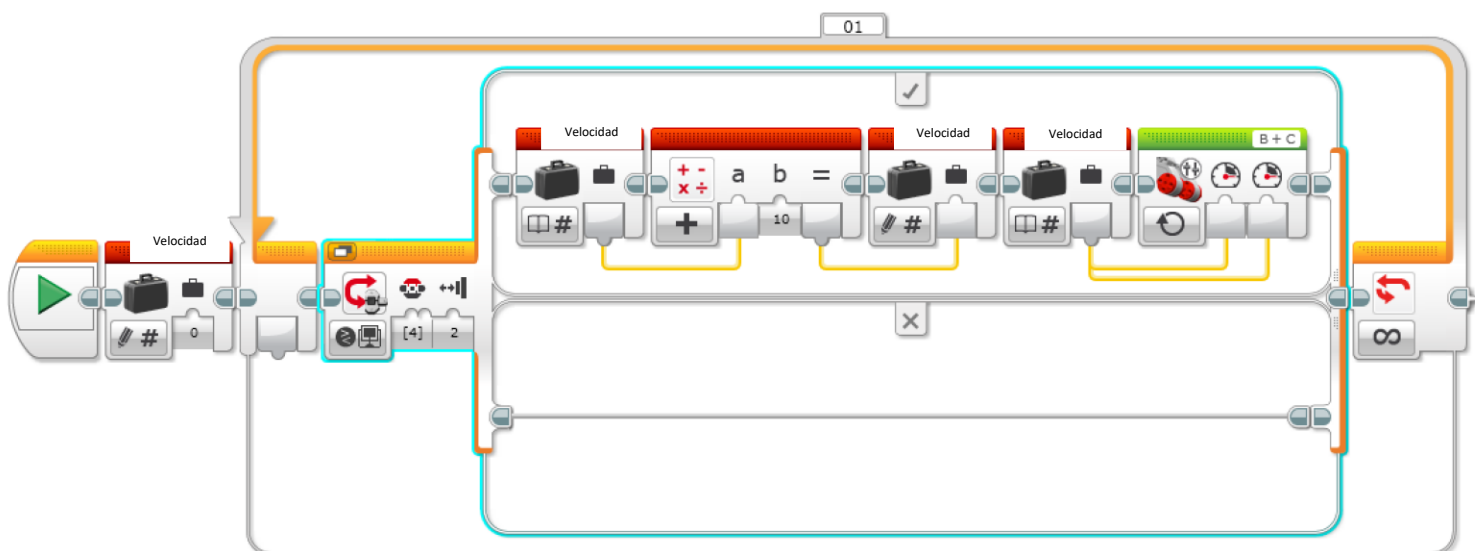


## Ejercicio 15: incremento y disminución de la velocidad (lecciones 27-28)

Mira el siguiente programa. Este hace que el robot incremente su velocidad en 10 cada vez que pulses un botón.

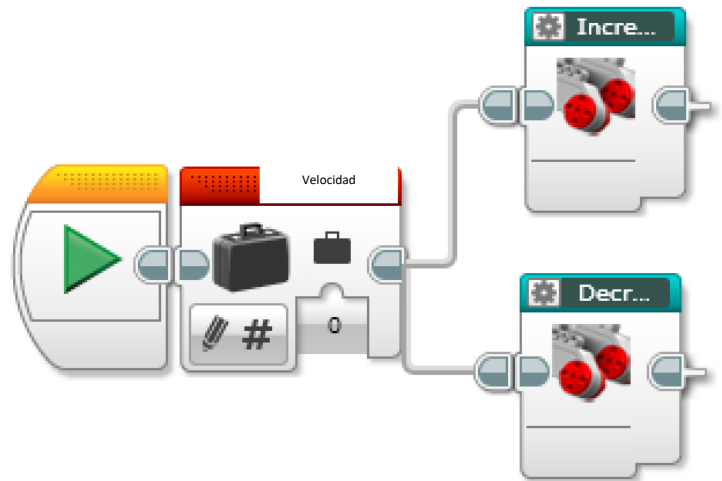
¿Cómo puede el robot disminuir la velocidad si se pulsa el botón de bajar?

Usa la variable «velocidad», disminúyela en 10 cada vez que pulses el botón de bajar.



## Ejemplo 7: uso de mi bloque en algoritmos

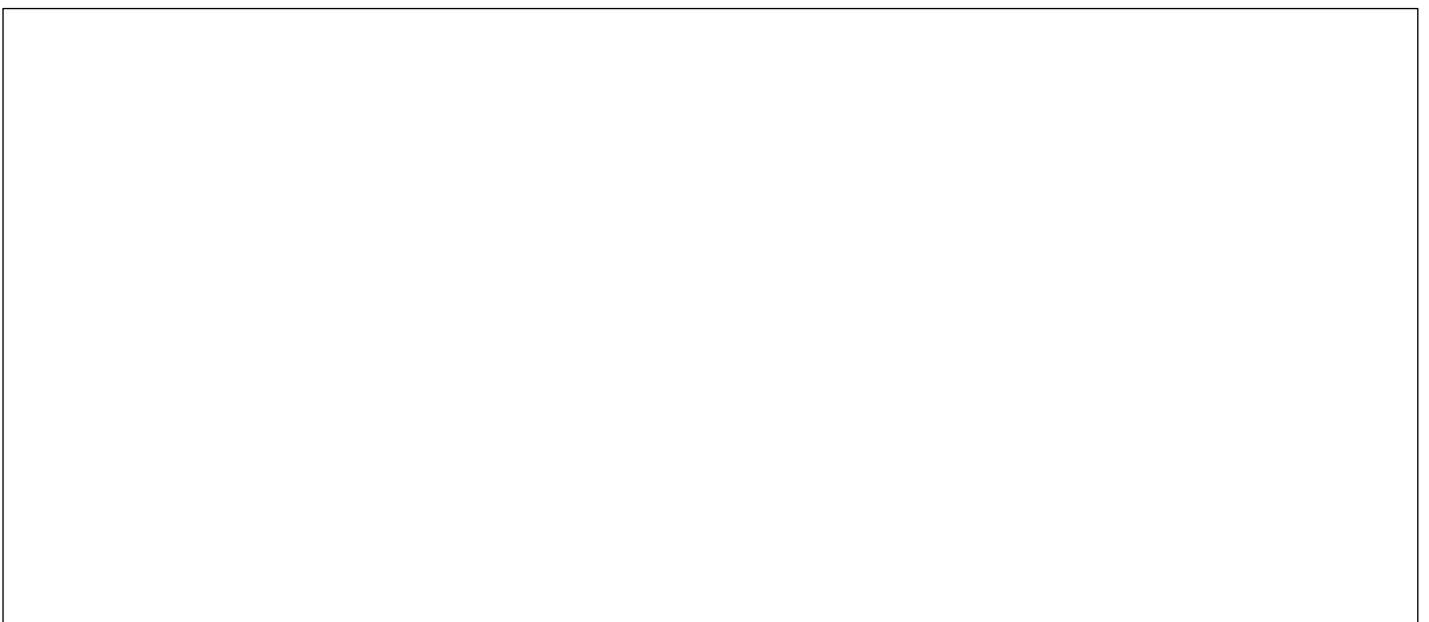
Este programa es un buen ejemplo donde se puede usar mi bloque. Los algoritmos para incrementar y disminuir la velocidad son muy amplios. Destaca el algoritmo que quieras para hacer un «mi bloque», selecciona «herramientas» y «constructor de mi bloque». Elige un dibujo adecuado y ponle nombre al bloque. El resultado es menos Amplio, como puedes ver.



## Ejercicio 16: programación paralela utilizando tres motores y un sensor

*Usando: haces paralelas, sensor ultrasónico, motores (variable, comparación, lógica, fin de ciclo lógico y espera)*

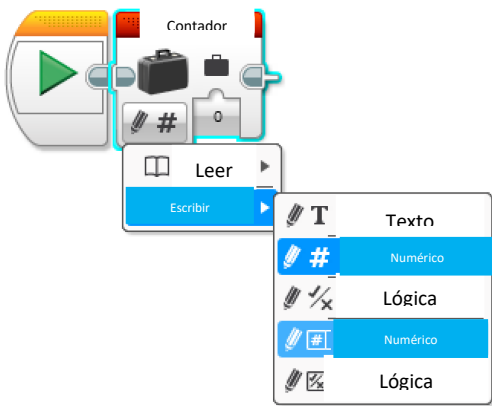
Queremos crear un robot que se mueva y, a la vez, utilice su sensor ultrasónico para registrar cuándo se está acercando a un objeto. Cuando el robot se acerque, disminuirá su velocidad y usará el motor central para mover un objeto de un sitio a otro.



## Ejemplo 8: variedad

Las variedades son útiles. A menudo necesitamos algún lugar para guardar más información, almacenarla y usarla. Una variedad trabaja de la misma manera que una variable. La diferencia es que puedes guardar solo un valor cuando uses una variable, y muchos objetos (números o enunciados de verdadera o falso) usando una variedad.

Necesitamos usar operaciones de variedad para gestionarla. Aquí hay una pequeña introducción sobre cómo de fácil es crear usar una variedad.

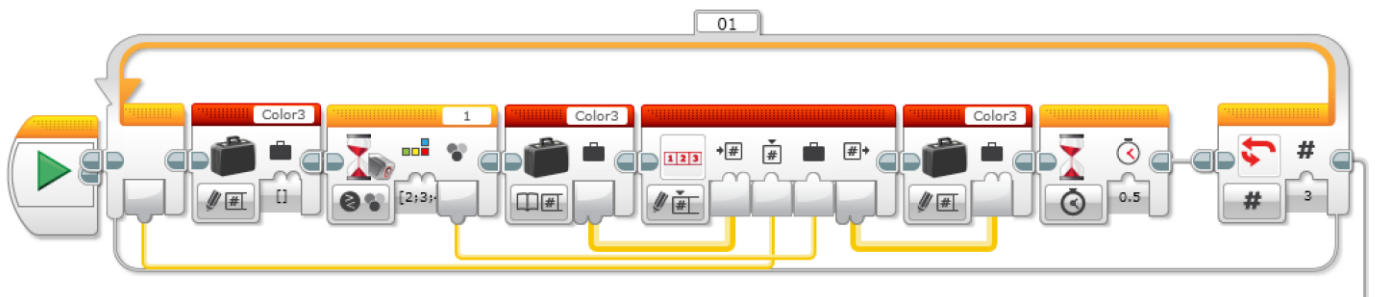
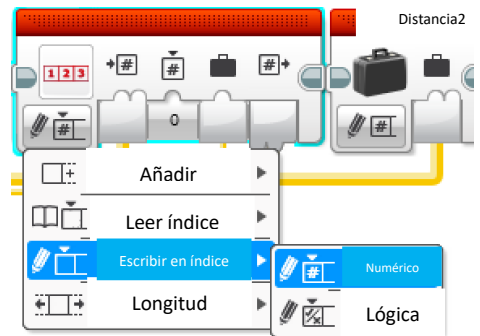


En la sección roja encontramos variables y variedades.

Primero, usamos variedad numérica o lógica. Esta es la información que podemos guardar en una variedad.

Podemos elegir escribir en una variedad o leer de ella.

En el siguiente ejemplo tenemos una variedad que contendrá los colores registrados con el sensor de color. Usamos el bloque de operaciones de variedad y escribimos en el índice numérico. Después, transferimos la variedad de colores como «variedad dentro» Transferimos el valor del sensor al valor en el bloque de operaciones de variedad. El índice podría ser también 0 o Podemos transferirlo a un ciclo. Si usamos un «espera» o unos sensores, podría ser mejor usar una transferencia del ciclo.



## Ejercicio 17: uso de variables y variedades

Cómo hacer que tu robot recuerde cinco comentarios y después siga las instrucciones.

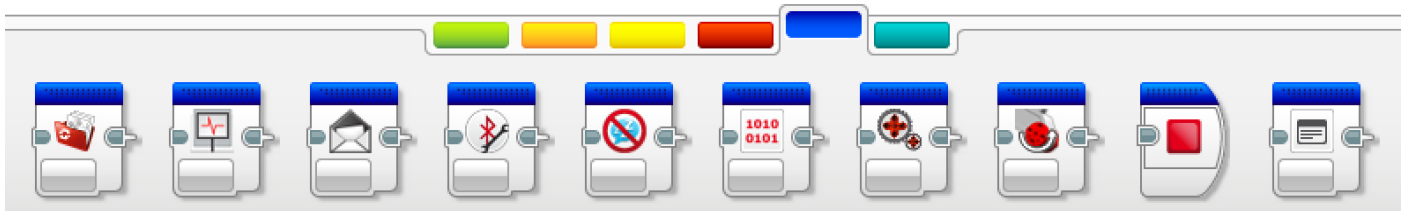
En el ejemplo 2 usamos una variable e hicimos que el robot se moviera a la derecha cuando pulsáramos el botón derecho. En el siguiente ejercicio vamos a crear un programa que nos permita pulsar el botón cinco veces, y después guarde esos cinco valores. En el siguiente paso, el robot leerá todos los valores y los implementará.

Lo llamamos botón de variedad, ya que va a guardar información sobre qué botón se pulsa. Elegimos la variedad numérica.



## Menú de la paleta – sección azul (lecciones 21-22)

En la sección azul encontramos los bloques avanzados.



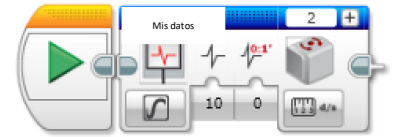
En este curso, necesitaremos utilizar almacenamiento de datos, Bluetooth y bloque de envío de mensajes.

Usar Bluetooth entre dos robots requiere que tengas un programa emisor y un programa receptor. Ambos programas tienen que cargarse en el robot correcto y empezarse.

En el programa, el bloque de Bluetooth necesita iniciarse solo en uno de los programas.



El bloque de almacenamiento de datos se puede usar para almacenar datos diferentes provenientes de sensores y motores.



### Ejemplo 9: conexión Bluetooth

Vamos a crear un sistema simple de alarma, utilizando Bluetooth y dos robots EV3.

- Primero, necesitamos establecer conexión entre los dos robots. Introducimos el menú de herramientas en las piezas y elegimos Bluetooth en los dos.
- Después, elegimos de nuevo Bluetooth
- Finalmente, hacemos clic en conexiones
- La primera vez que conectemos los dos EV3, necesitamos usar un código pin (que suele ser 1234) y hacer clic en ok.

### Emisor y receptor

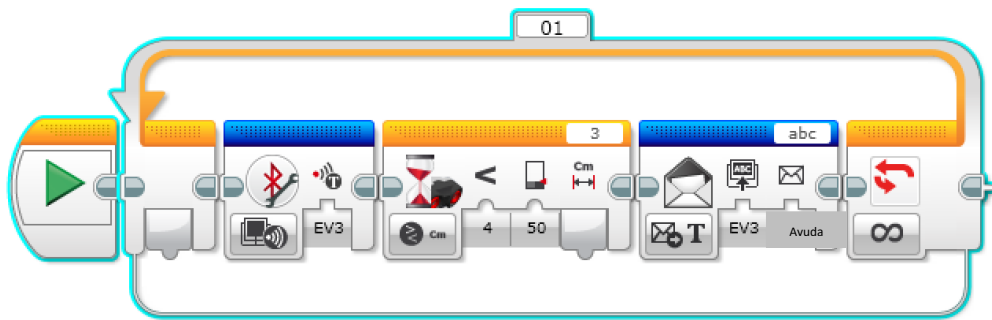
En un sistema de alarma, tenemos una unidad emisora que suele conectarse a un sensor ultrasónico u otro tipo de detector de movimiento. La misión del emisor en nuestro sistema de alarma es registrar movimientos y enviarlos a la alarma, que es una unidad que contiene un sistema de sonido y quizás pida ayuda por teléfono.

Tenemos que hacer un programa para el emisor y un programa para el receptor. Ambos tienen que descargarse correctamente e iniciarse. Solo de esta manera los programas tendrán la posibilidad de alcanzar y enviar los mensajes del otro.

Solo se tiene que iniciar uno de los dos robots (el emisor).

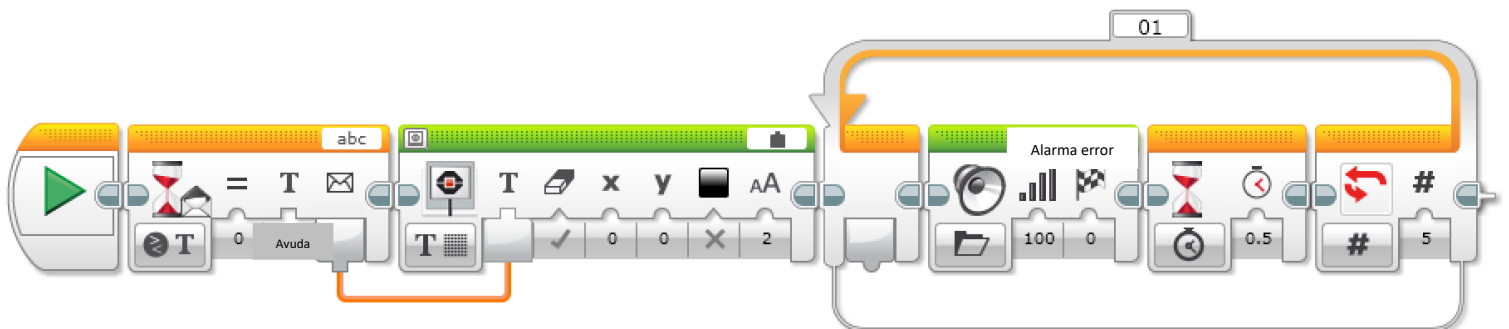
En el siguiente ejemplo, tenemos un sensor ultrasónico que espera los movimientos con la cercanía de 50 cm. Si el sensor registra algo, se envía un mensaje al EV3 (que es el otro robot). El mensaje dice «ayuda».

## Programa emisor:



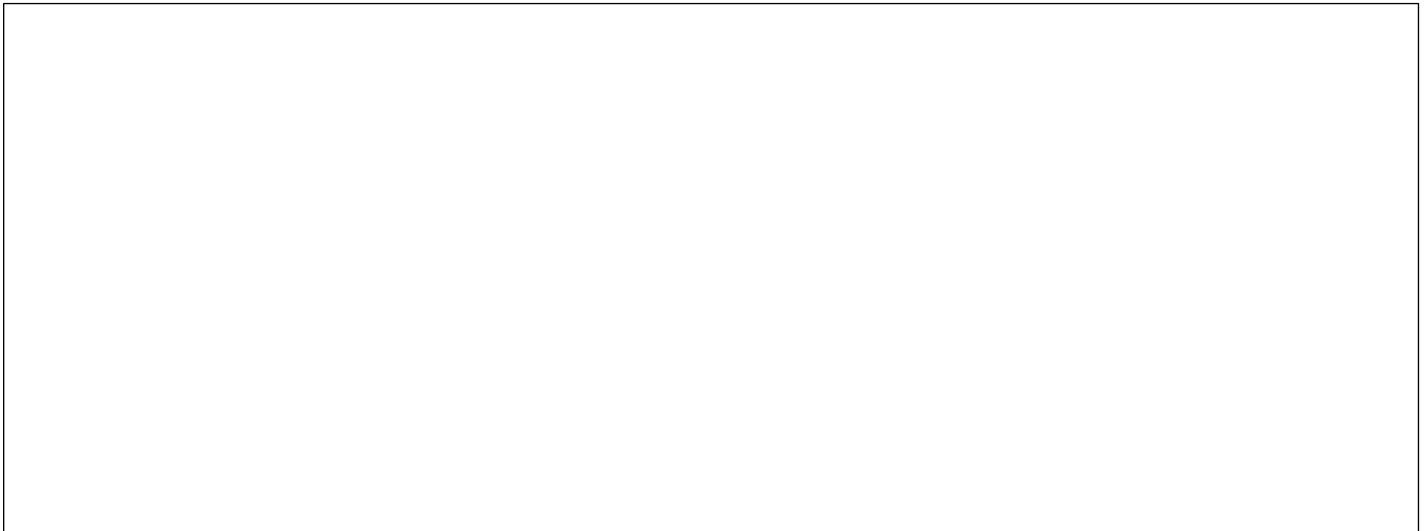
Tenemos otro programa descargado en el receptor, que empieza con un bloque de espera. Espera el mensaje de «ayuda» y, cuando este llega, se envía a la pantalla (en realidad se envía a la agencia de seguridad) y suena una alarma 5 veces (en realidad suena hasta que se apaga).

## Programa receptor:



## Ejercicio 18: sistema de alarma

Haz un sistema de alarma que funcione en una fábrica que produce papel blanco. Cuando el sensor de color registre algún otro color, envía el mensaje «ERROR» al sistema de alarma (pieza principal) que responda con un mensaje de «ERROR» en la pantalla y el sonido «COLOR DETECTADO».



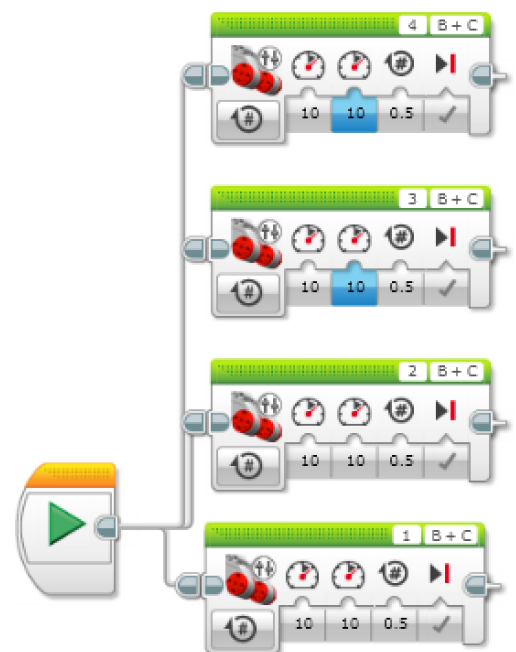
## Cadena margarita y EV3 (lecciones 19-20)

Es posible conectar cuatro EV3 entre sí y, así, hacer un sistema con más sensores y motores.

Para programar los robots en el modo de cadena margarita, debemos introducir las propiedades del proyecto y marcar dicho modo.

Daisy-Chain Mode

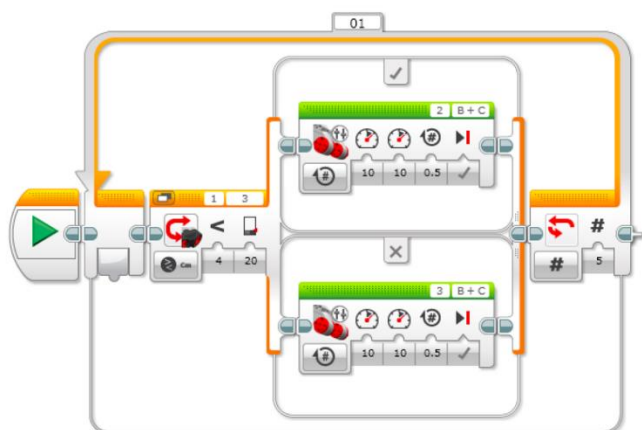
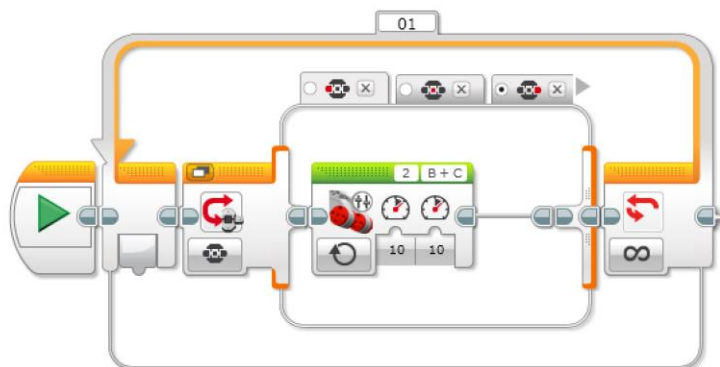
El primer robot conectado al ordenador es el robot 1. Este tendrá motor 1A, 1B y 1C, y sensor 11, 12, 13 y 14 conectado a él. La siguiente pieza se conecta a través del Puerto del ordenador y Puerto USB. Tendrá motor 2A, 2B y así. De esta manera, es posible tener varios motores y sensores conectados





Los únicos bloques disponibles para la programación de esclavos (los demás) son flujo, sensores y bloques motor. Todos los demás se ejecutarán en la pieza principal.

Este ejemplo tiene una pieza que mide en la principal.



En este ejemplo, la pieza principal utiliza el sensor de ultrasonido para controlar los motores en las otras dos piezas.

## Almacenamiento de datos (lecciones 23-24)

En este experimento vamos a medir los diferentes tipos de giros que pueden hacer los robots y vamos a observarlos en el osciloscopio. De esta manera, podemos comparar los giros e intentar entender cuál funciona mejor que los otros.

En el siguiente programa, todos los giros están programados en el mismo programa. En vez de hacerlo de esta manera, podríamos haberlo hecho cuatro programas diferentes y hacerlos funcionar todos, uno a uno.

Estos son los motores que funcionan en el programa:

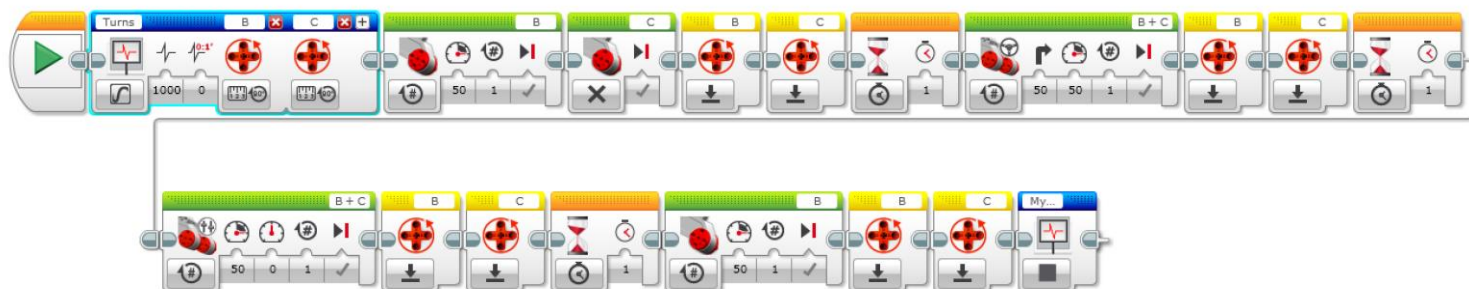
Primero: el motor B solo usa el motor grande y el motor C detenido por completo.

Segundo: motor B y C usando la dirección de movimiento

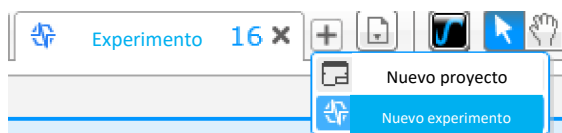
Tercero: motor B y C usando el tanque de movimiento con solo un motor funcionando

Cuarto: Motor B usado (y motor C ignorado)

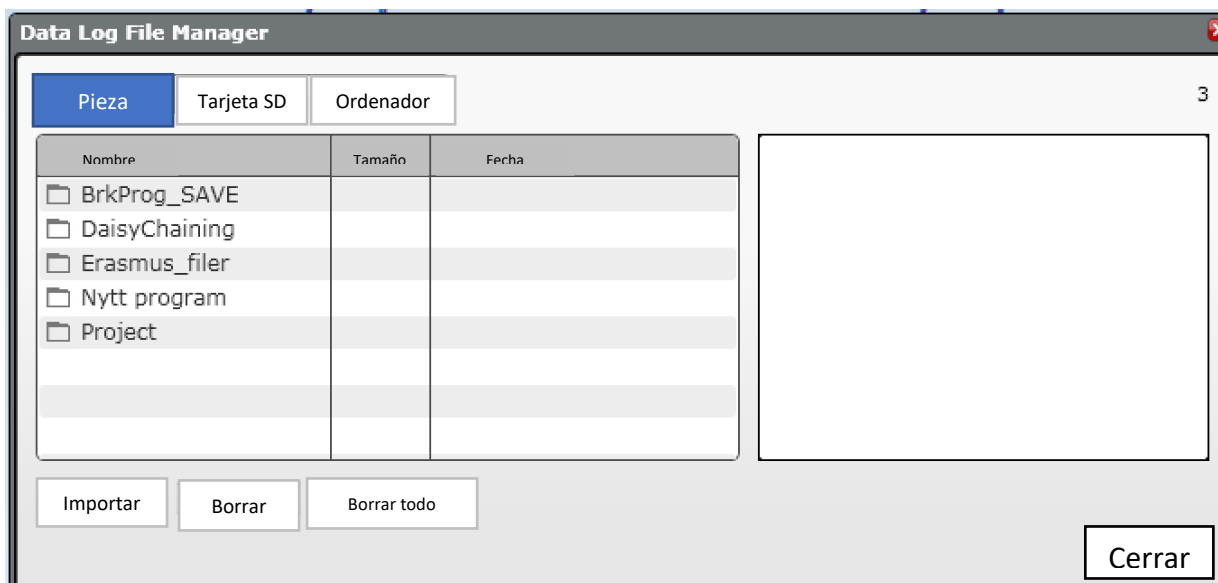
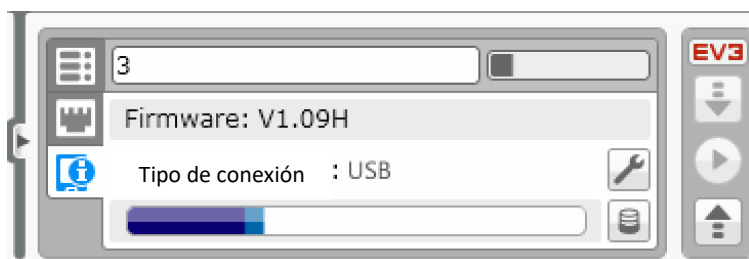
Tenemos que usar el reinicio del motor en ambos motores B y C, para hacer que el sensor de rotación se vuelva a poner a 0. De esta manera, el gráfico también se pondrá a 0 antes de medir el segundo giro.



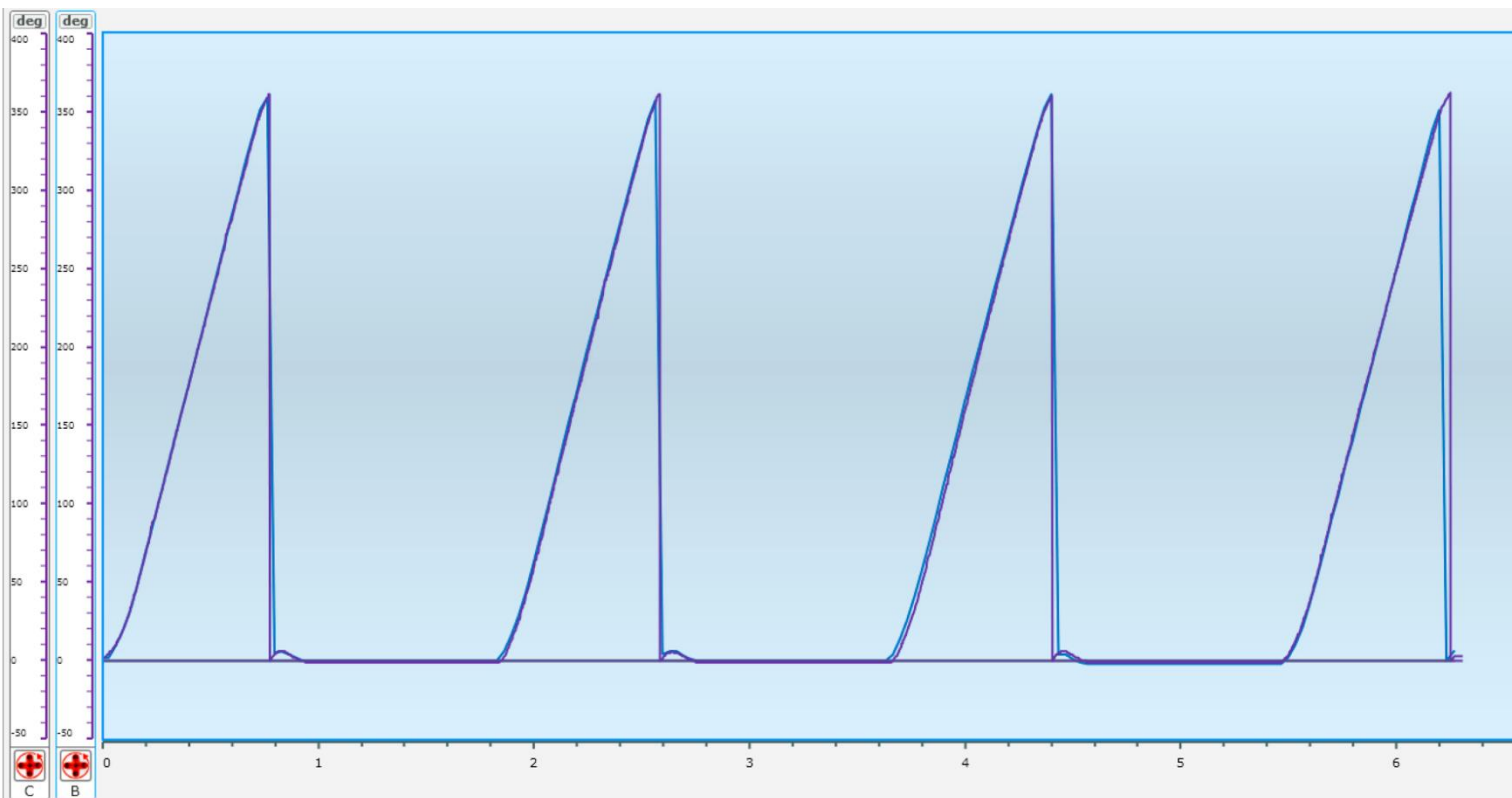
Elige nuevo experimento.



En el sector de hardware, elige «subir» en la esquina derecho. Después, elige el archive que querrías importar al experimento. Primero tienes que hacer funcionar el programa en tu robot.

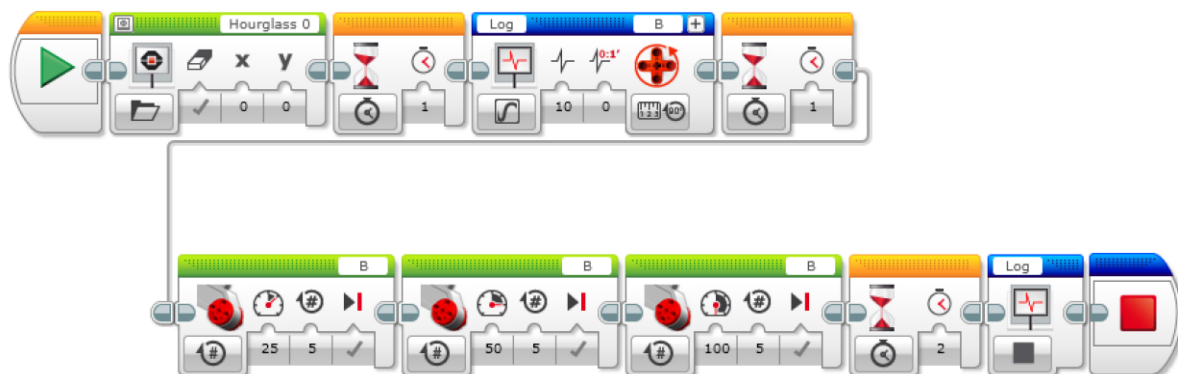


El resultado aparecerá así. En este caso, podemos ver dos pruebas diferentes importadas del mismo experimento. Así, podemos comparar los giros y, como podemos ver, el último giro no fue muy bueno, aunque el primero y el Segundo sí que lo fueron.



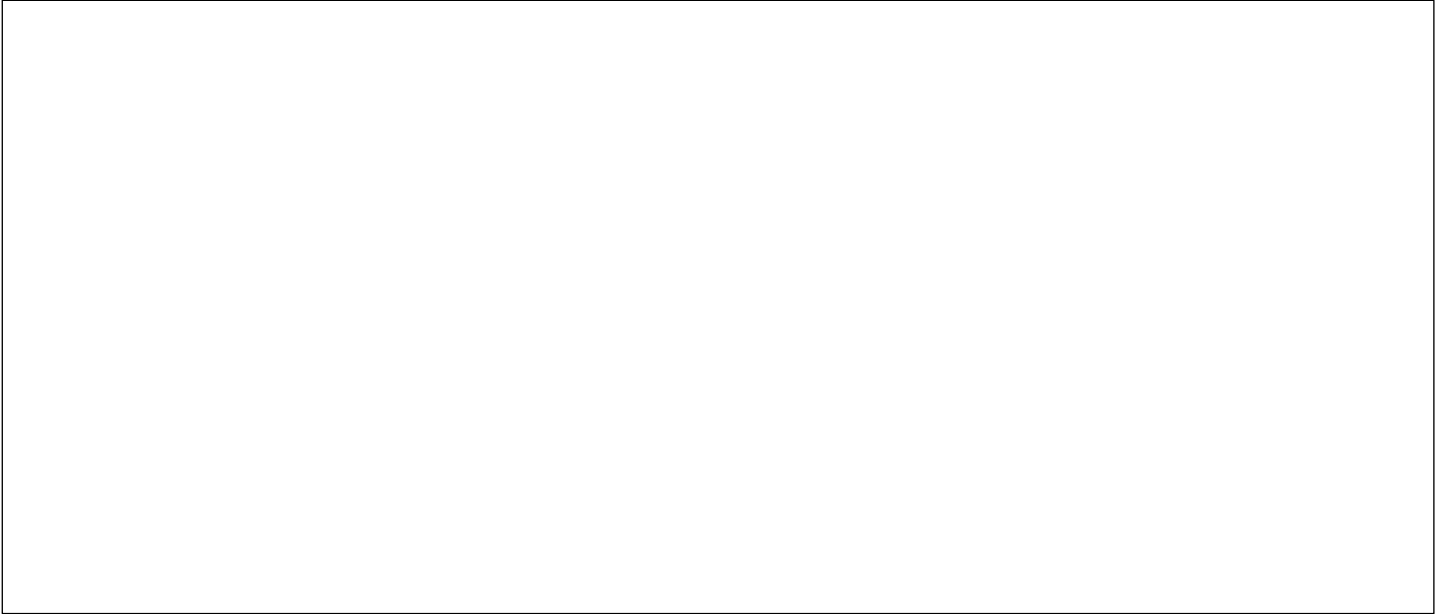
### Ejercicio 19: velocidad

En este experimento, vamos a usar almacenamiento para calcular la velocidad del robot.



Discute con tu compañero ¿qué crees que hará este programa?

Haz que tu robot utilice este programa y, después, reciba el resultado en un experimento nuevo. Pon en el número correcto para los ejes de x e y encuentra la velocidad (aproximada) para el robot.



## Soluciones posibles

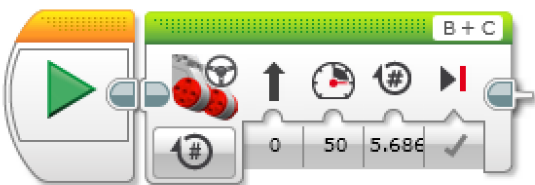
### Ejercicio 1

Solución: recorrer 100 cm usando el bloque de tanque de movimiento



Movimiento de 100 cm  
Usar perímetro ruedas. 17,584. Dividir  $100/17,584 = 5,686$   
Este es el número de giros usados para moverse 100 cm

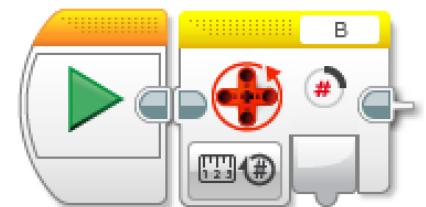
Solución: recorrer 100 cm usando el bloque de dirección



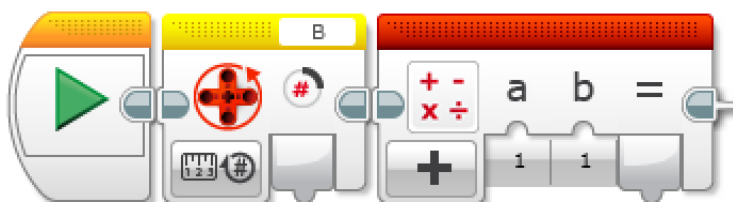
### Ejercicio 2

Solución: mostrar en la pantalla la distancia que recorre el robot

Para mostrar la distancia que recorre el robot, usamos primero el bloque de giro. Elige uno de los motores del robot (B/C) y colócalo para medir los giros.



Usando bloques matemáticos y cableado

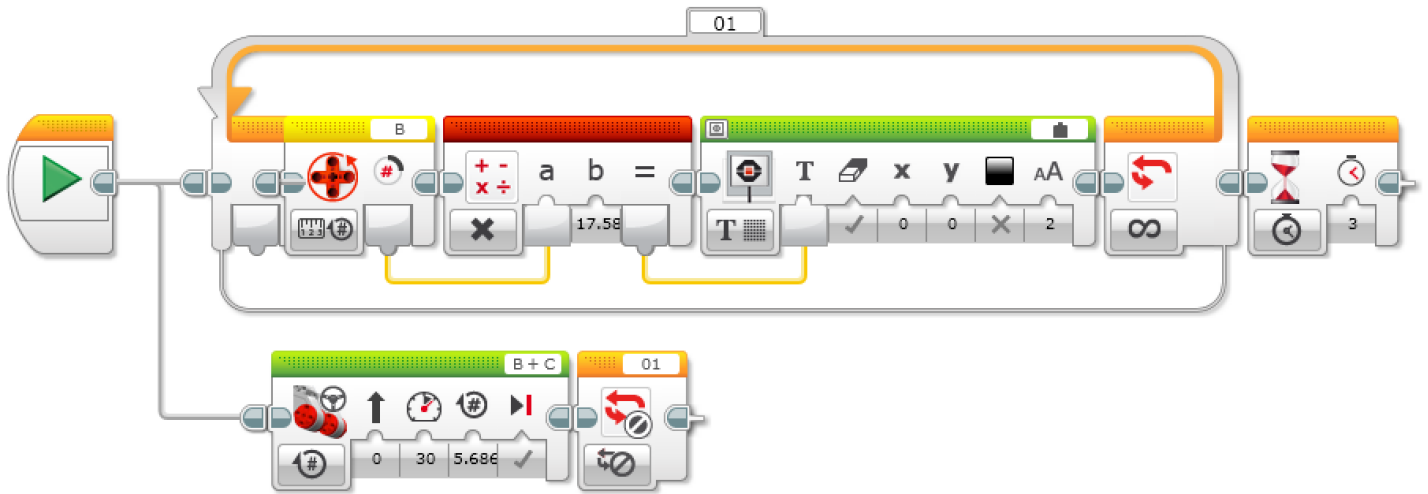


Podemos usar un bloque matemático para resolverlo.

Solución: para mostrar la distancia recorrida, usamos el bloque de giro. Elige uno de los motores del robot y colócalo para medir los giros. Después, transferimos el resultado al bloque matemático y elegimos multiplicar los giros con el valor 17,568 (cm), que es el perímetro de la rueda. El resultado se manda al bloque de pantalla. Para realizar esta transferencia, tenemos que ponerlo en «transferir» en la esquina superior derecha. Después,

los ponemos todos en un ciclo. Necesitamos un algoritmo extra para hacer que el robot se mueva de verdad.

En esta solución hay un interruptor de ciclo en el final del algoritmo de



moto para detener el ciclo 01. Al final de dicho ciclo, hay también una espera de 3 segundos, así, el número de rotaciones se mostrará durante 3 segundos antes de que acabe el programa.

### Ejercicio 3

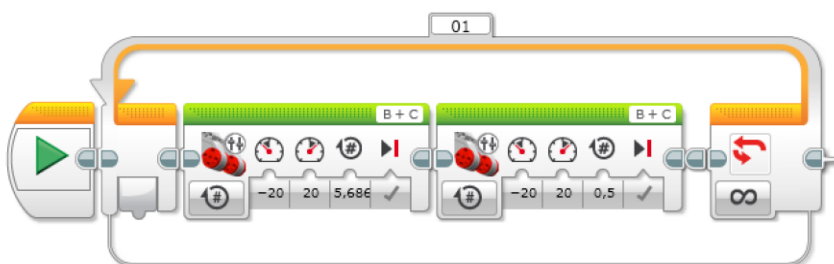
Solución: giro de 180 grados



Mover bloque de tanque  
Usa 2 motores con 1 giro. Uno de los motores con velocidad negativa y el otro, con positiva

### Ejercicio 4

Solución: algoritmo para recorrer 100 cm x 100cm en un cuadrado usando rotaciones. Es importante poner el ciclo a contar 4 veces, si quieres que el robot deje de moverse después de completar el cuadrado.



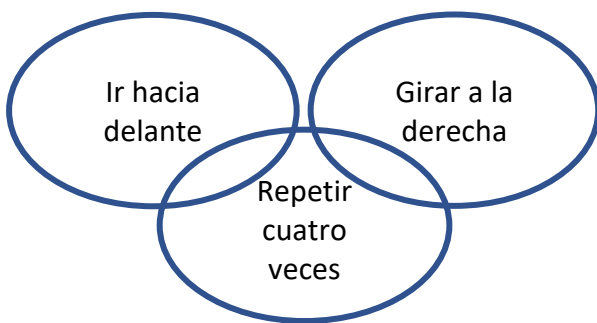
Bloque de movimiento de tanque, usa 5,686 giros para moverse 100 cm.  
Girar 90 grados, necesita 0,5 giros – usar el bloque de movimiento de tanque

## Hacer un algoritmo

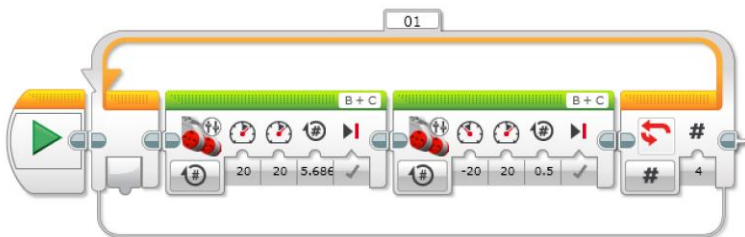
- 1) Definir todo (movimiento, giros, pantalla, Contador, etc.). El programa necesita contener para resolver el caso. «Lluvia de ideas» (enfocada en una solución concreta)



- 2) Reordenar los movimientos en el orden correcto

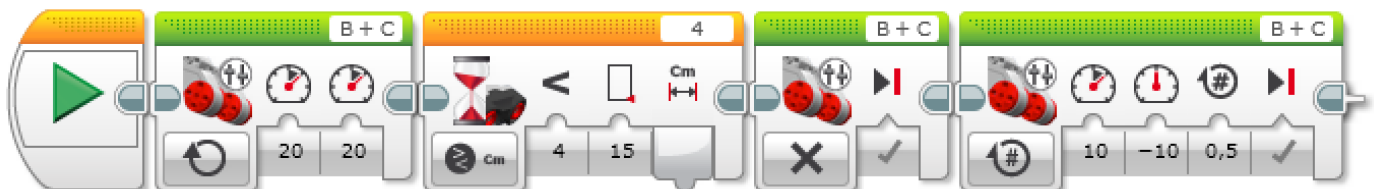


- 3) Decidir qué bloques (o texto) demandan cada movimiento. Pon los bloques juntos y crea un programa.



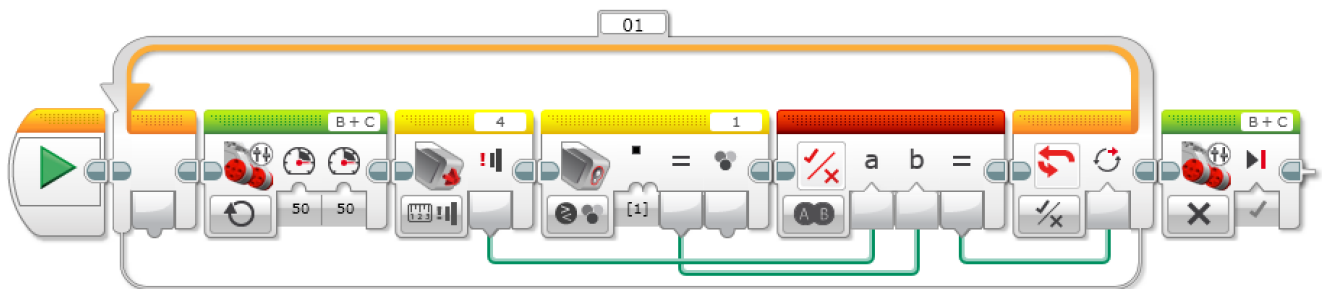
## Ejercicio 6

Solución:



Solución:

Explicación. Utilizamos el bloque motor dejándolo en posición encendido. Después, usamos el sensor táctil y el de color, y añadimos comentarios de los dos sensores al bloque lógico. El bloque lógico se pone en modo OR, que significa que, si el sensor táctil se pulsa o el sensor de color ha detectado el color elegido, el bloque lógico volverá al estado de verdadero. El estado de verdadero se transfiere al ciclo, que quiere decir que el ciclo finalizará cuando el estado lógico se haga verdadero. Después, el motor se detendrá.



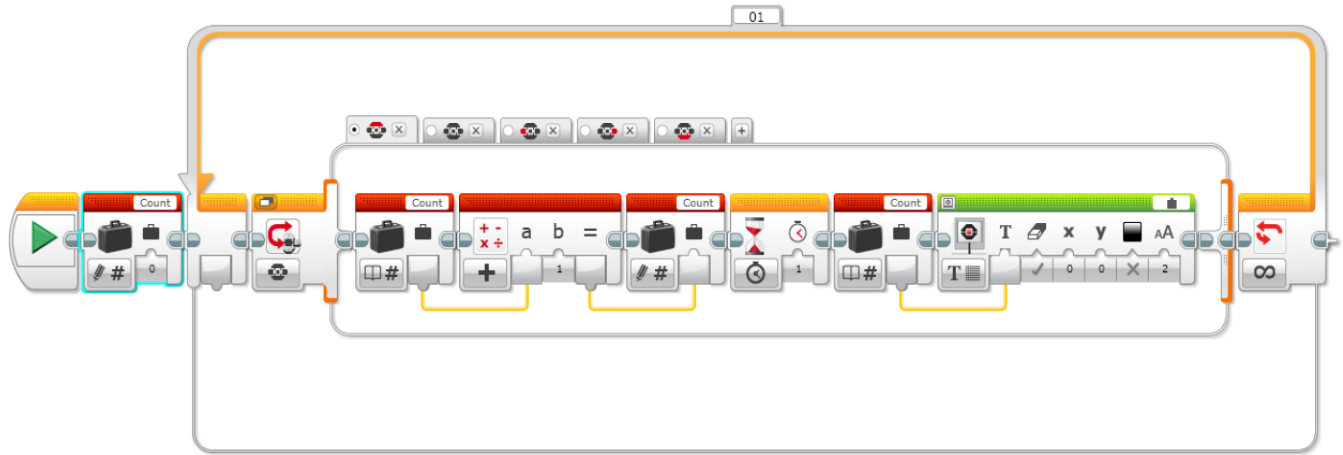
## Ejercicio 8

### Solución:

Este programa muestra cómo puedes hacer que un robot cuente las veces que se pulsa el «botón superior» y muestre este número en la pantalla.

Aquí estamos haciendo una variable que se llama contador. Más tarde, usamos un interruptor y define qué pasará si se pulsa es «botón superior».





## Ejercicio 9

### Solución:

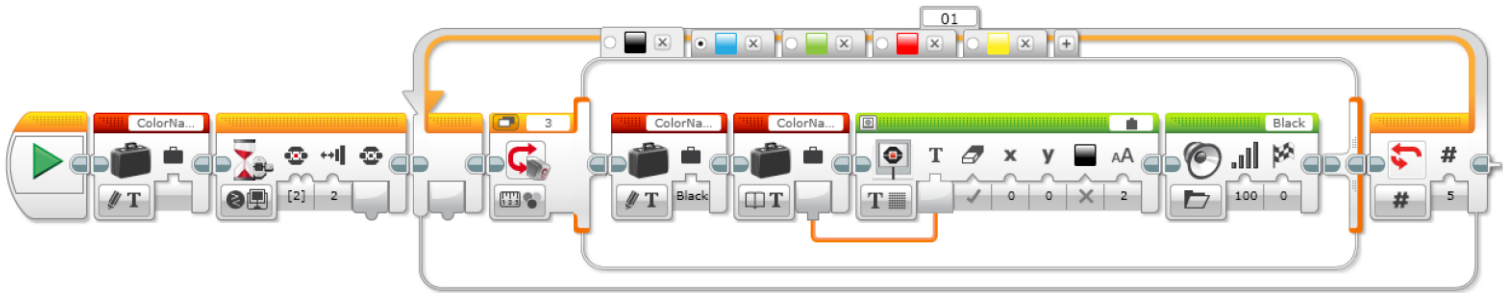
Ahora vamos a crear un programa usando variables y sensores.

Primero, creamos una variable llamada «nombre del color» (o algo más describiendo qué hace la variable). Esta variable debería tener un texto, en este caso «rojo, negro, amarillo, verde y azul». Después, ponemos un bloque de espera utilizando el botón. El botón del centro iniciará el bloque de sensor. El sensor registrar los colores y, dependiendo del color que registre, añadirá el color a la variable «nombre del color» como un texto.

Por ejemplo, el sensor registra el color verde. La sección del interruptor en el programa hace que la variable «nombre del color» mantenga el color verde. Posteriormente, debemos leer la variable y conseguir el color de él. Para ponerlo en la pantalla, usamos el cableado. Es importante poner el bloque de pantalla en posición de transferencia en la esquina superior derecha. También debemos recordar poner el bloque de pantalla para que mantenga el texto y los píxeles.

También es posible ponerlo en el sonido correcto. En este caso, el sonido «verde» en el mismo programa.

En el ejemplo de abajo, hay también un ciclo. El programa funcionará cinco veces y después finalizará.

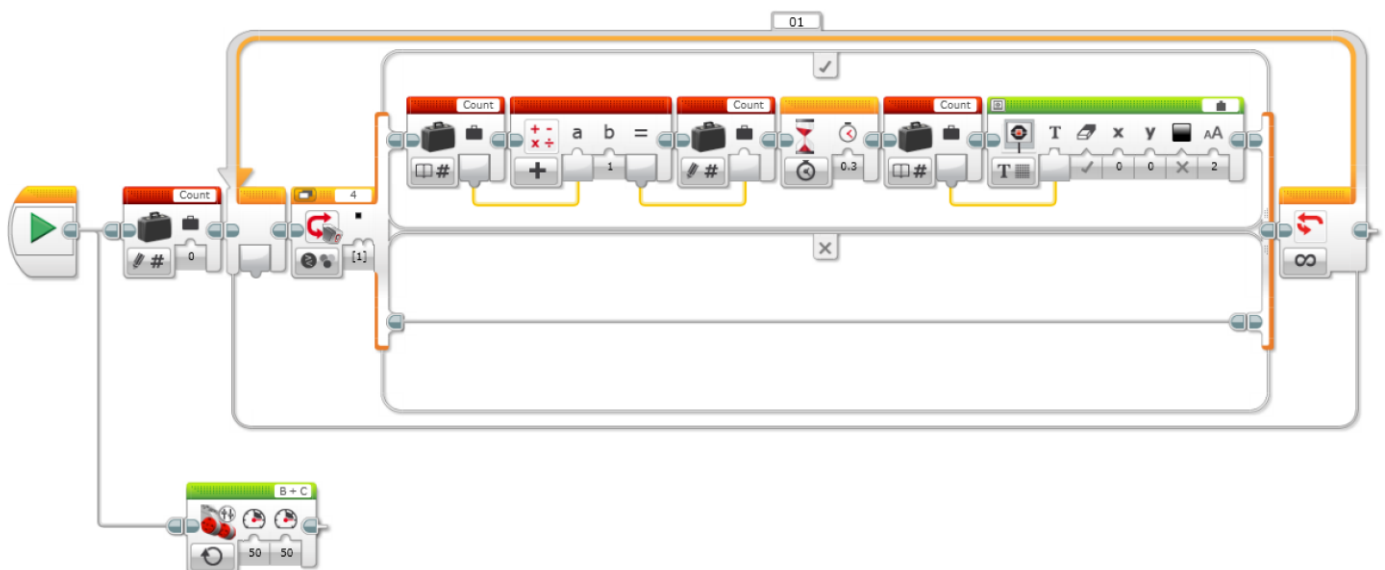


### Ejercicio 10

Solución: programar el robot para que se mueva y, a la vez, cuente las líneas negras que pasa durante su camino. Deberíamos utilizar programación paralela. El número de líneas negras debería aparecer en la pantalla, así pues, necesitaríamos usar variables y matemáticas.

Tenemos que crear una variable. Nombra a esta «contador» y usa una espera para el color. Elige el color negro. Tenemos que usar un interruptor donde definimos qué debería pasar si aparece el color negro delante del sensor de color.

Se lee la variable «contador» y se eleva uno, usando la suma en un bloque matemático.



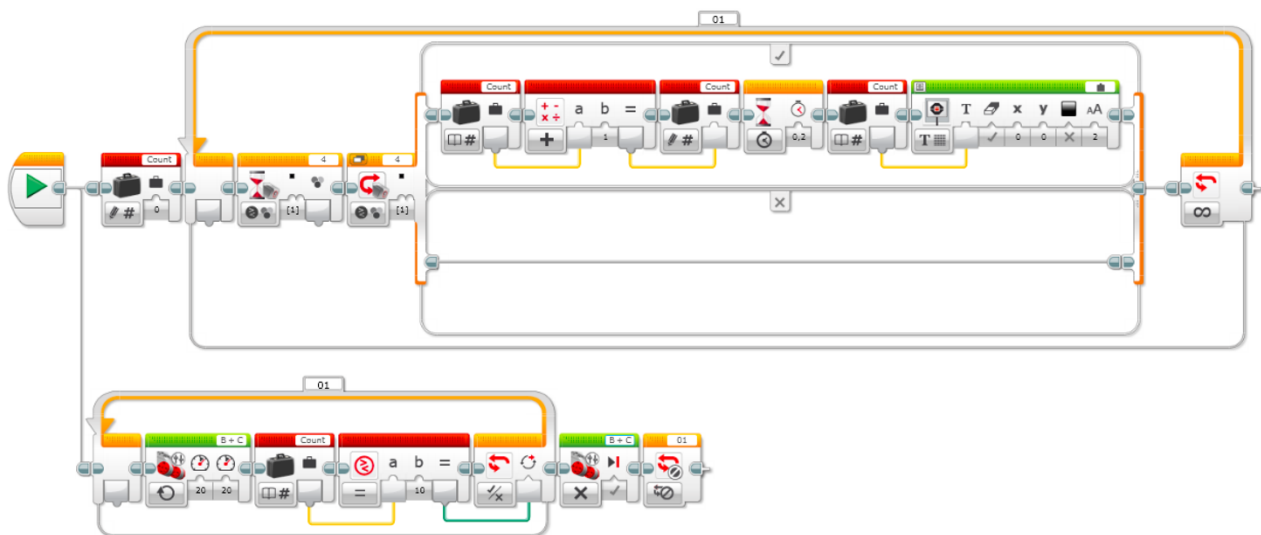
Escribimos el resultado en la variable «contador». Una pausa de un Segundo y se lee dicha variable. Tras esto, se pone en la pantalla. Para poner un valor de una variable en el bloque de pantalla, tenemos que

elegir «transferir» en el menú superior derecho. Después, transferimos. Pon todo en un ciclo, ya que queremos seguir contando las líneas negras que pase el robot.

### Ejercicio 11

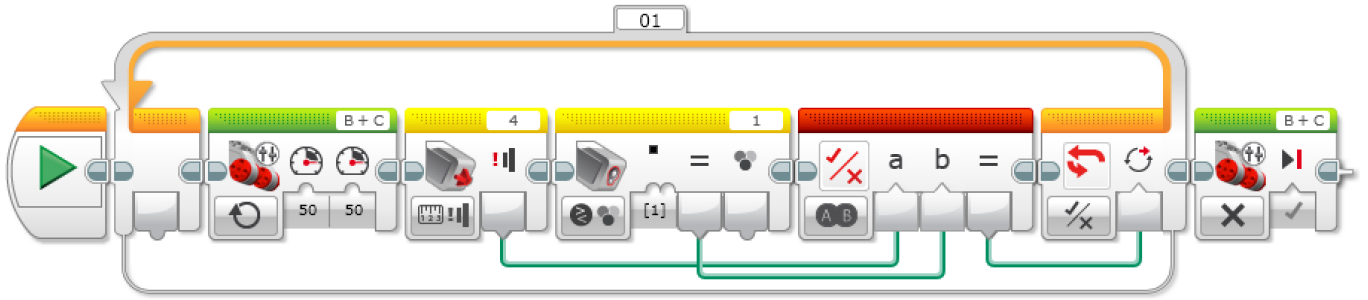
Solución: para hacer que el robot deje de moverse tras pasar 10 líneas negras, añadimos algunos bloques extra al algoritmo que contiene el bloque motor. En este algoritmo, leemos el valor de la variable «contador» y lo compara con un valor seleccionado. Usamos el bloque de comparación y usa la elección =. Luego, ponemos el valor b a 10 y transfiere este valor al verdadero/falso de la lógica al ciclo. Esto significa que cuando el ciclo tenga un verdadero lógico, finalizará. Debemos poner un bloque de «fin de motor» para detener el motor. Finalmente, ponemos un interruptor de ciclo al final del programa, para terminar la ejecución del ciclo contador.

Solución:

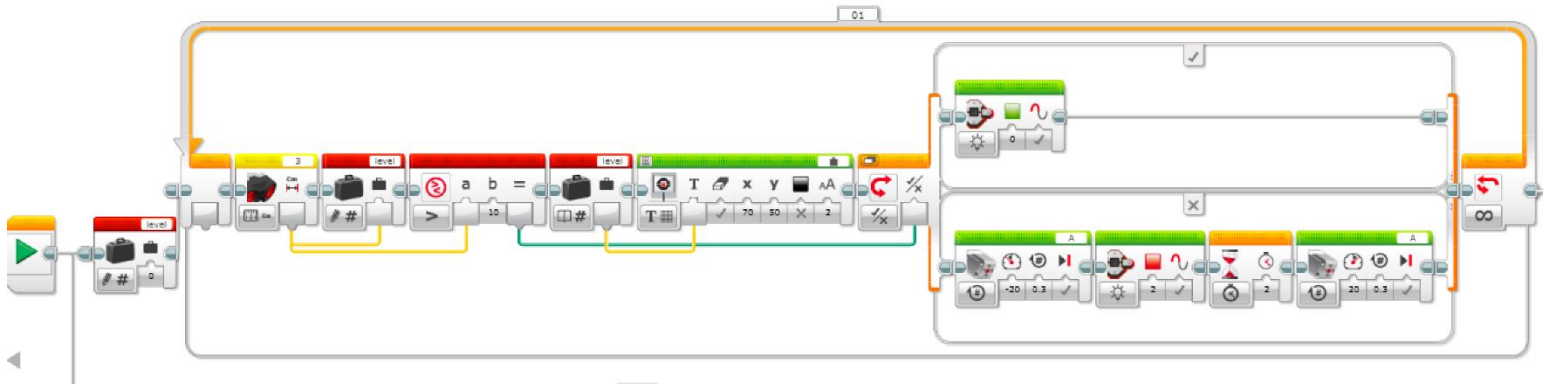


### Ejercicio 12

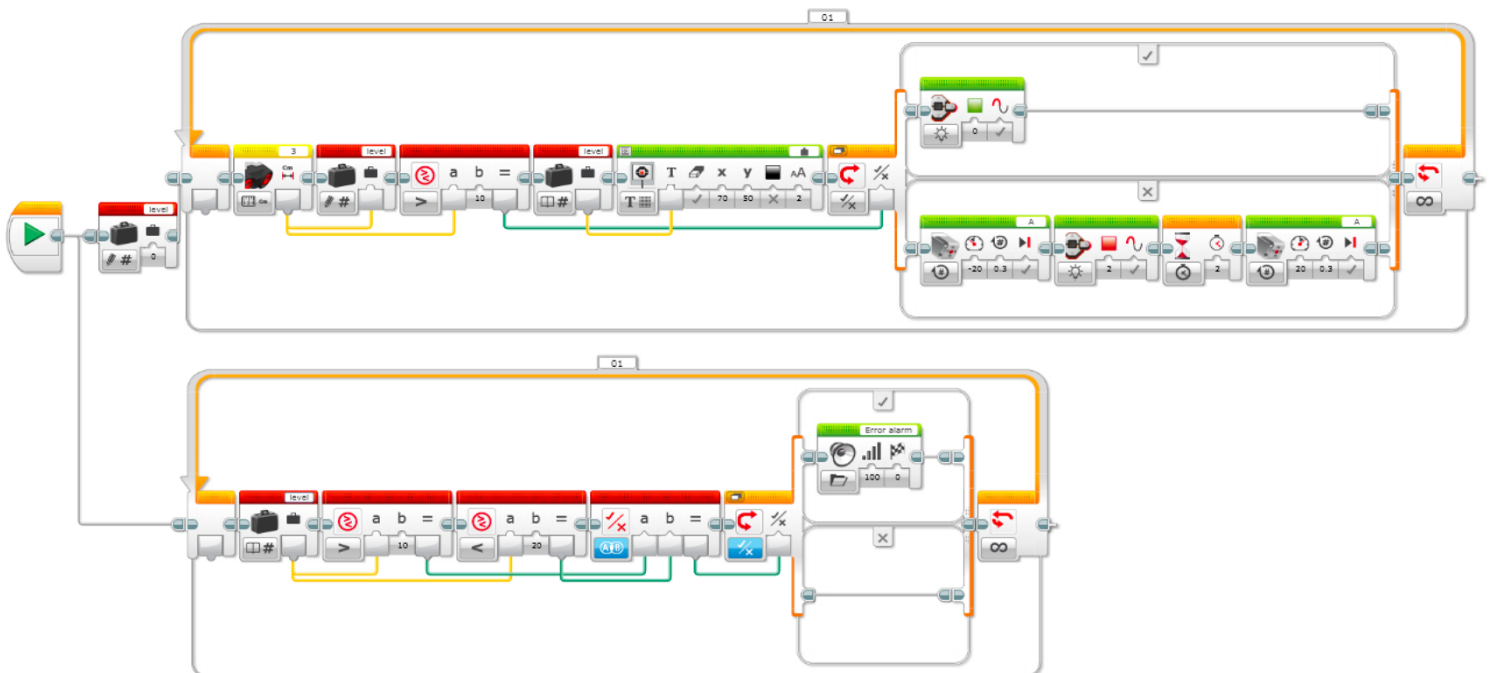
Queremos programar un robot que se mueva hasta que el sensor táctil se pulse y el sensor de color detecte el color negro. Cuando se condigan estas condiciones, el ciclo se detiene en un VERDADERO lógico. Después, el motor se detiene.



Ejercicio 13a  
Solución:



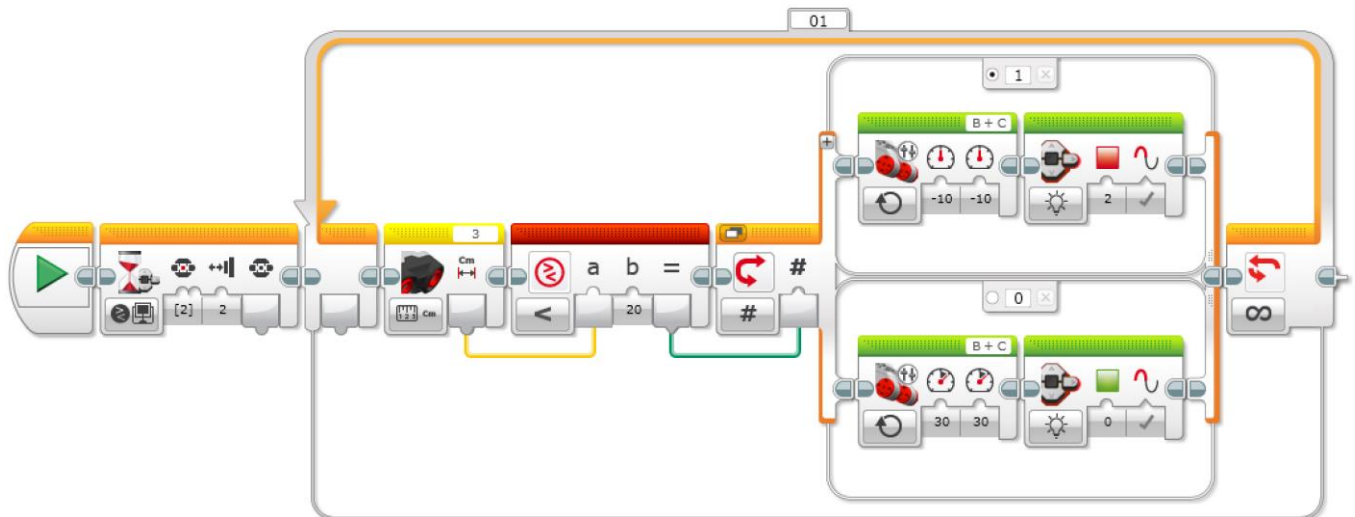
Ejercicio 13b  
Solución:



## Ejercicio 14

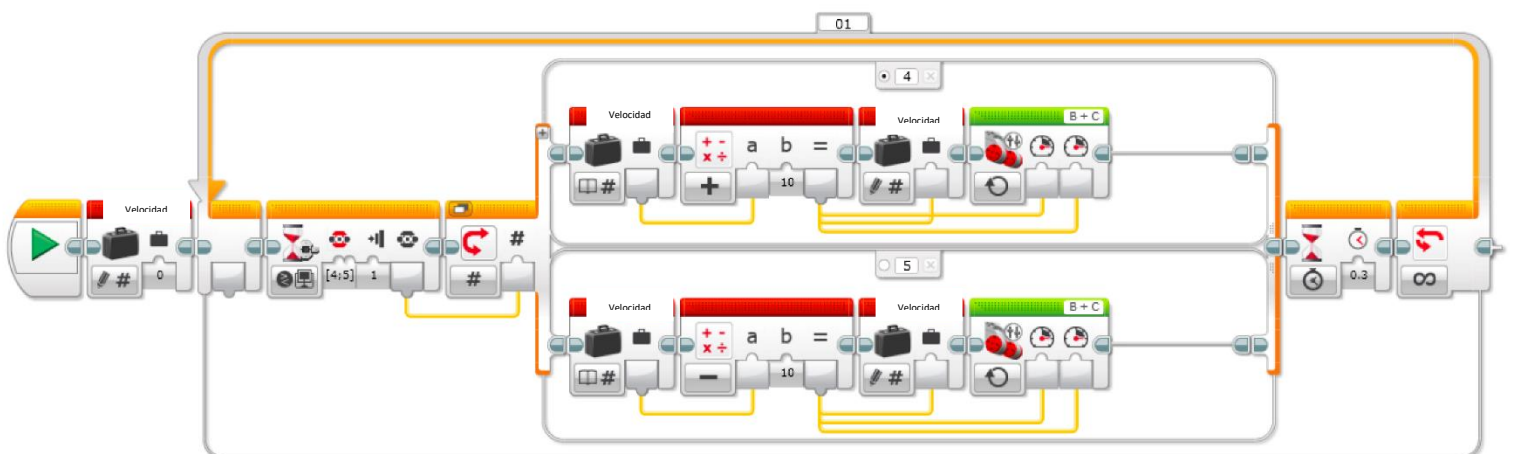
### Solución:

Si tu mano está más cerca de 20 cm del sensor ultrasónico, el robot se moverá hacia atrás con la luz de estado roja. Si tu mano está más lejos de 20 cm de dicho sensor, el robot irá hacia delante con la luz de estado verde.



## Ejercicio 15

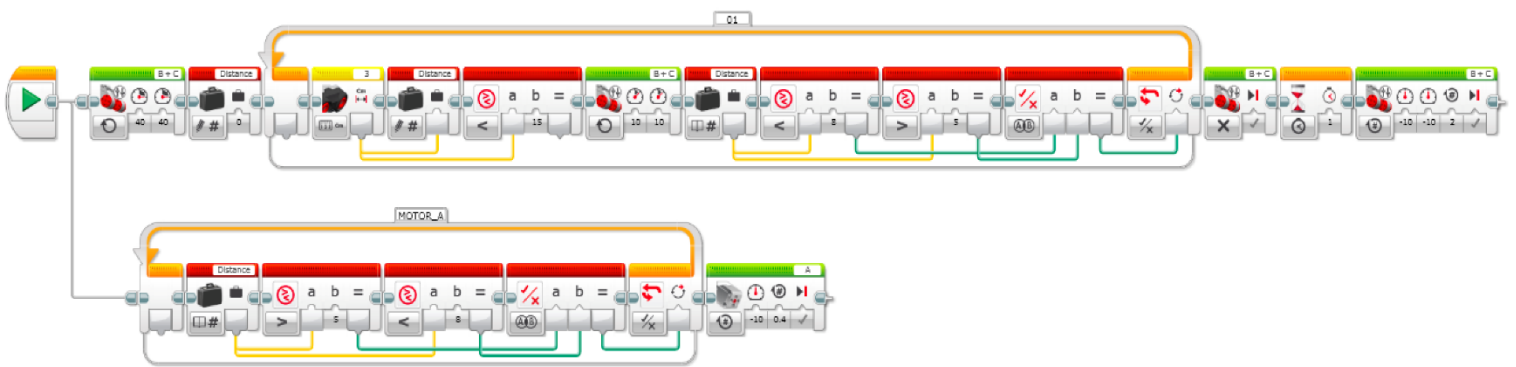
### Solución:



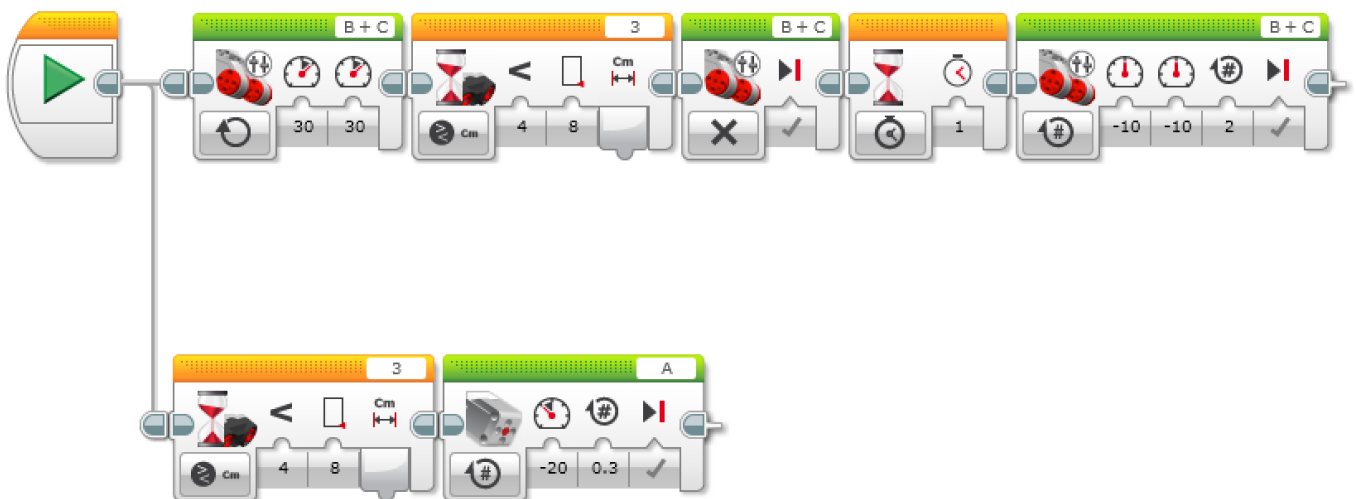
## Ejercicio 16

### Solución 1

Este reto se puede resolver de diferentes maneras. Aquí hay dos soluciones que hacen que el robot haga las mismas acciones. Debido a que el robot no debería representar ningún dato en la pantalla, no necesitamos usar bloques de variable y comparación como en la primera solución. El bloque de espera hace la misma comparación y el programa se puede resolver de una manera mucho más fácil.



### Solución 2



## Ejercicio: 17

### Solución:

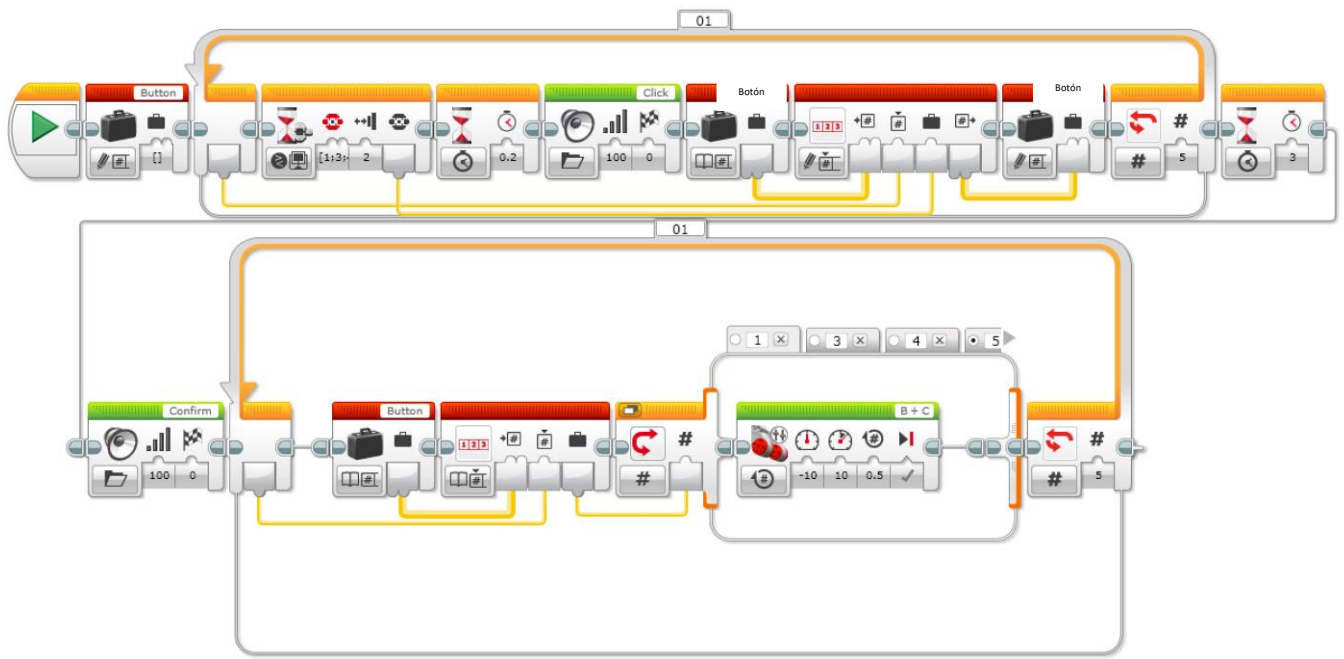
Creamos una variable llamada botón.

Después, necesitamos un ciclo que repita cinco veces, ya que el robot debería recordar cinco entradas diferentes de los botones.

Por cada botón pulsado: lee de la variedad, envía la información de la variedad a las operaciones de variedad, consigue el índice y el valor del botón y envíalo de vuelta al conjunto.

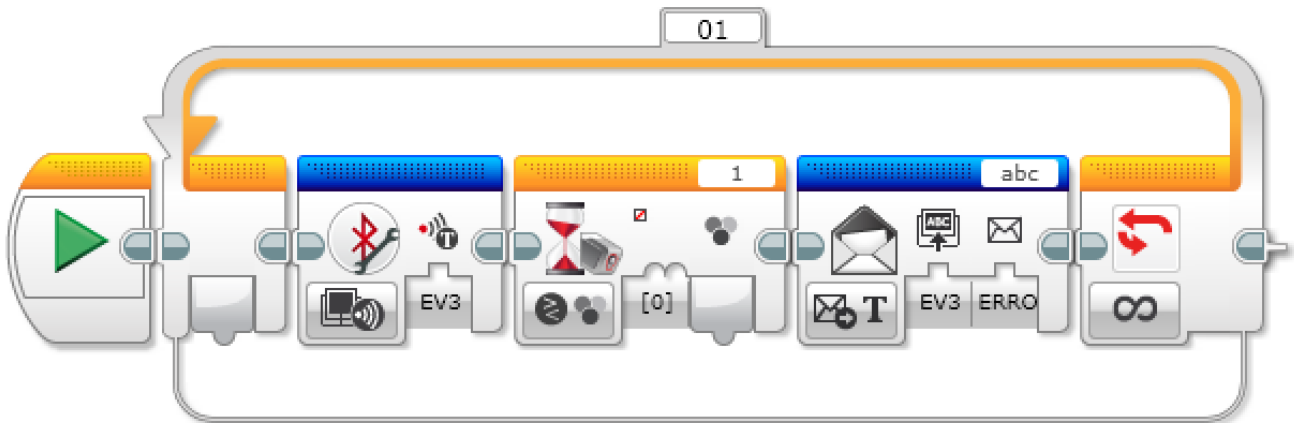
Después esperamos durante 3 segundos.

Lee del botón, envíalo a las operaciones del conjunto, consigue el índice correcto, envía el valor para cambiar al modo numérico (es decir, que conseguir un 1,3,4 o 5) y, después, haz funcionar el motor adecuado.

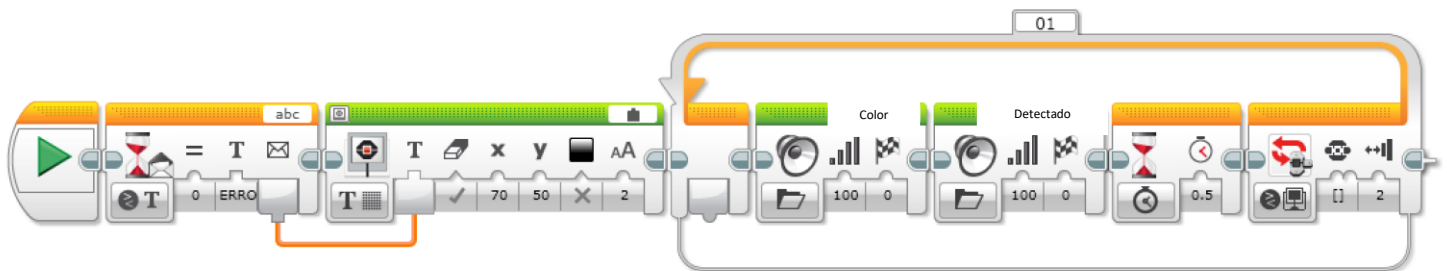


## Ejercicio 18

Programa emisor:



Programa receptor:



## Ejercicio 19

En este experimento vamos a usar almacenamiento para encontrar la velocidad del robot.





Como podemos ver en la ventana del gráfico, hay una parada de un segundo al principio. Después, el robot empieza a moverse durante siete segundos en la primera ronda, 3,4 segundos (8,2-11.6) la siguiente ronda y, finalmente, 2,2 segundos en la última ronda.

Velocidad: 0,1256 m/s – 0,2512 m/s – 0,399 m/s (aproximadamente)

