

# BUSCANDO OBJETOS



## STEMJAM Teaching Guide

Developing make spaces to promote creativity  
around STEM in schools

Acronym: STEMJAM

Project no. 2016-1-ES01-KA201-025470

[www.stemjam.eu](http://www.stemjam.eu)



Co-funded by the  
Erasmus+ Programme  
of the European Union

# BUSCANDO OBJETOS

## RESUMEN

La idea es programar el mBot para buscar un área limitada.

Usando el robot para encontrar un objeto perdido en un área limitada. Esta área está limitada por la línea negra. El objeto podría ser un bloque alto o una botella y se encuentra usando un sensor de ultrasonidos. También se puede hacer uso de una brújula para mantener la dirección.

## OBJETIVOS DIDÁCTICOS

Los estudiantes aprenderán:

- ❖ A utilizar la brújula.
- ❖ A programar diferentes bucles.
- ❖ Entender los ángulos.

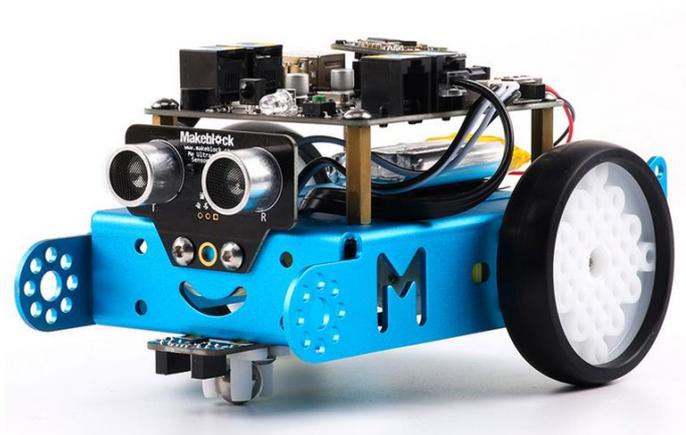
Materia STEM:            Ciencia             Tecnología             Ingeniería             Matemáticas

Nivel educativo:            12-14 años             14-16 años

## LISTADO DE MATERIALES

Los componentes electrónicos y dispositivos necesarios para esta actividad son:

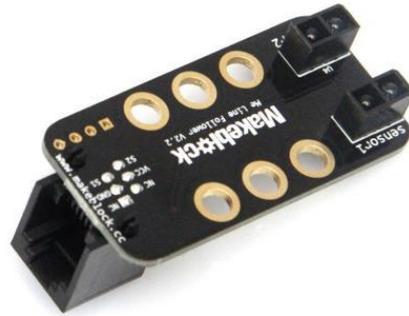
- mBot => Ref. 90054



❖ Me Sensor de Ultrasonidos:



❖ Me Seguidor de Líneas:



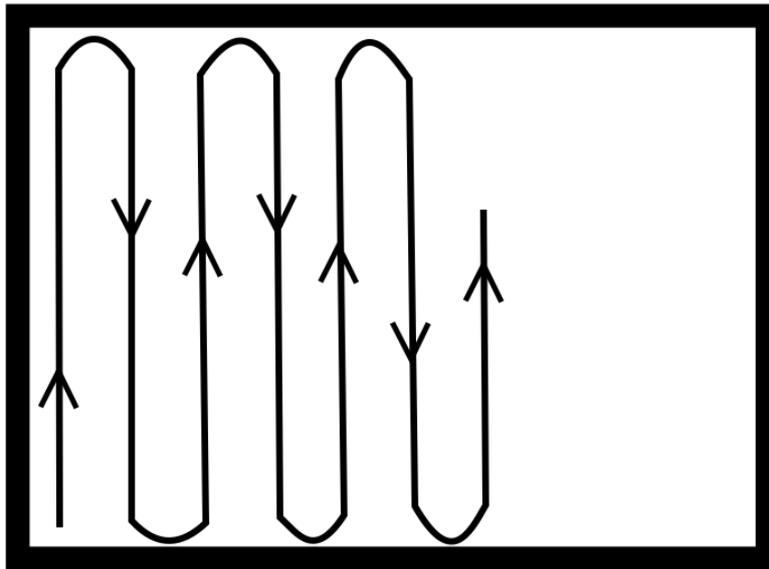
❖ Tablero con área limitada.

❖ Una brújula.

ELEMENT	ID	CABLE	AMOUNT	PORT 1			PORT 2			PORT 3				PORT 4				P.MOT 1	P.MOT 2
				Y	B	W	Y	B	W	Y	B	W	B	Y	B	W	B	W*	W*
Mbot Robot 2'4G																			
Motor 1	W*		1													W*			
Motor 2	W*		1														W*		
Me 7-Segment serial display	B		1	B															
Me Ultrasonic sensor	Y		1					Y											
Me Line Follower	B		1			B													
Me Compass	W		1										W						
RJ25 cables			4																
Structures and beams																			
Laptops			1																
Attrezzo (not essential)																			

## DESCRIPCIÓN DE LA ACTIVIDAD

El robot debe escanear el área. La idea es que el robot vaya así:

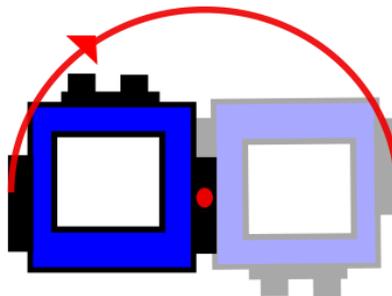


Cuando el robot detecta una línea negra, debe hacer un giro de 180 grados, pero de tal manera que pueda evitar la búsqueda en la misma área una vez más y que no se pierda. También debemos prestar atención al hecho de que un robot debe girar a la izquierda y a la derecha alternativamente. Esa breve descripción se puede dividir en los siguientes problemas a resolver:

- ❖ Movimiento hacia delante.
- ❖ Giro de 180°.
- ❖ Alternar giros.
- ❖ Gire habilitando la exploración de toda el área, pero evitando repetir la ruta.

### Problema: Escanear el área “muerta”

El último problema parece ser el más fácil de resolver. Para girar a la derecha, la rueda izquierda debe ir y la rueda derecha debe detenerse.



## Problema: Alternar giros

El siguiente problema es que cada vez que el robot llega a la línea negra, cambian los giros, el robot gira: izquierda, derecha, izquierda, derecha, etc.

Lista de los pasos:

Paso 1:  $K=0$

Paso 2: Repetir hasta que el sensor de ultrasonidos sea mayor que 5.

Paso 3: Movimiento hacia delante.

Paso 4: Si el sensor de línea es igual a negro y  $k = 0$ , entonces giramos  $180^\circ$  a la izquierda y  $k=1$ , o si el valor del sigue líneas es igual a "negro" y  $k=1$ , giramos  $180^\circ$  a la derecha y  $k$  será igual a 0, o bien, volvemos al paso 3.

```
mBot Program
wait until on board button pressed
wait until on board button released
set k to 0
forever
  if line follower Port2 = 0 and k = 0 then
    o180L
    set k to 1
  else
    run forward at speed 100
  if line follower Port2 = 0 and k = 1 then
    0180P
    set k to 0
  else
    run forward at speed 100
```

o180L

```
define o180L
run forward at speed 0
set motor M2 speed 100
wait 4 secs
```

0180P

```
define 0180P
run forward at speed 0
set motor M1 speed 100
wait 4 secs
```

## Problema: ¿Cómo hacer un giro de 180 grados con precisión?

Para definir el tiempo de funcionamiento, éste no funciona bien, ya que la velocidad depende de la potencia de la batería. En este proyecto el robot debe girar con gran precisión.

Podemos usar la brújula e indicar el ángulo para cambiar.

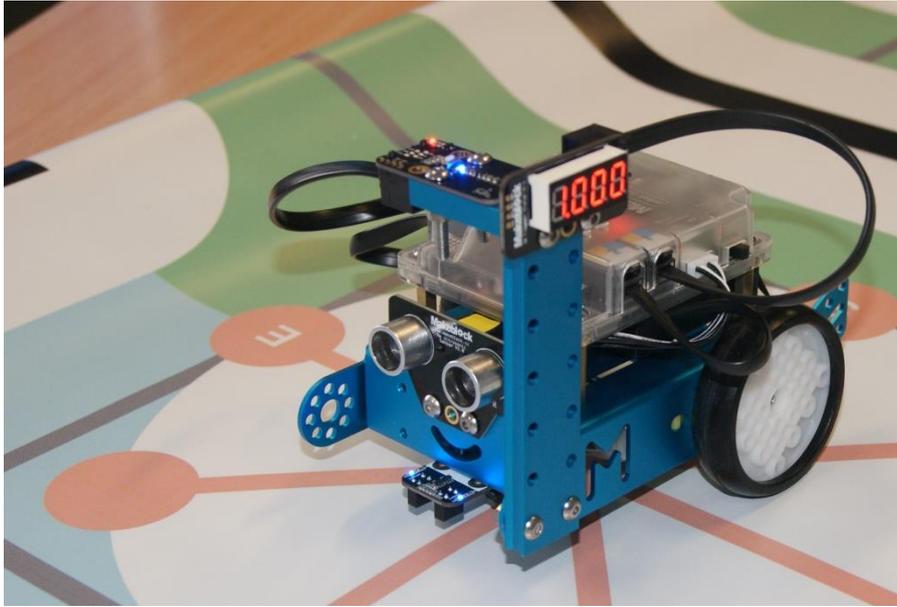
## La brújula

La brújula es el sensor, que da el número entre 0 y 360.

0 o 360 significa Norte. El número 90 significa Este, 180 - Sur, 270 - Oeste.

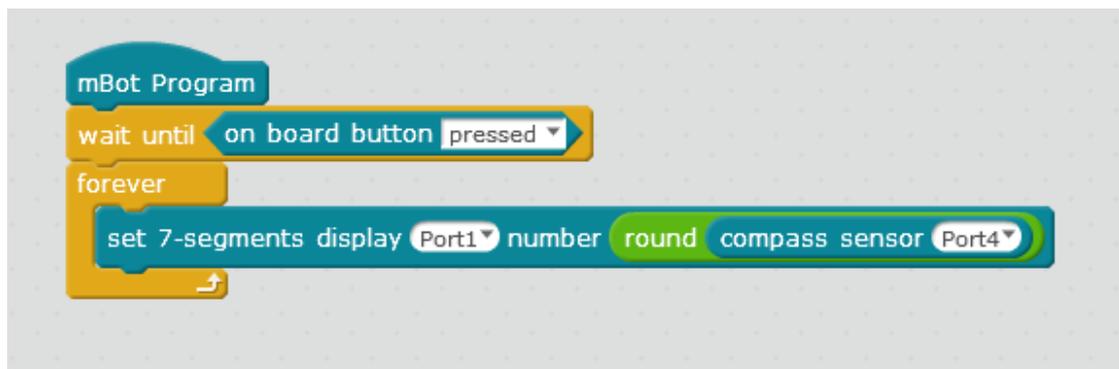
Cada programa que use una brújula debe usarse en el modo Arduino. Antes de cada uso debe calibrarse la brújula. Durante la calibración, el sensor puede detectar algunas interferencias locales y ajustarse al lugar.

## Construyendo la brújula:



La brújula debe montarse de forma horizontal. El eje X puede ser paralelo al eje del robot. La brújula es sensible a los materiales ferromagnéticos (hierro o níquel), por eso debes intentar montarla lo más alto posible. La cubierta del robot está hecha de aluminio que tiene una influencia infinitesimal en la lectura de la brújula, pero los motores que producen un campo electromagnético no son neutrales a sus lecturas. La brújula también es sensible a la postura: cuanto más horizontal sea el montaje, más precisas serán las lecturas. Después de un montaje y calibración cuidadosos, no debemos aceptar errores de medición superiores a 2 grados.

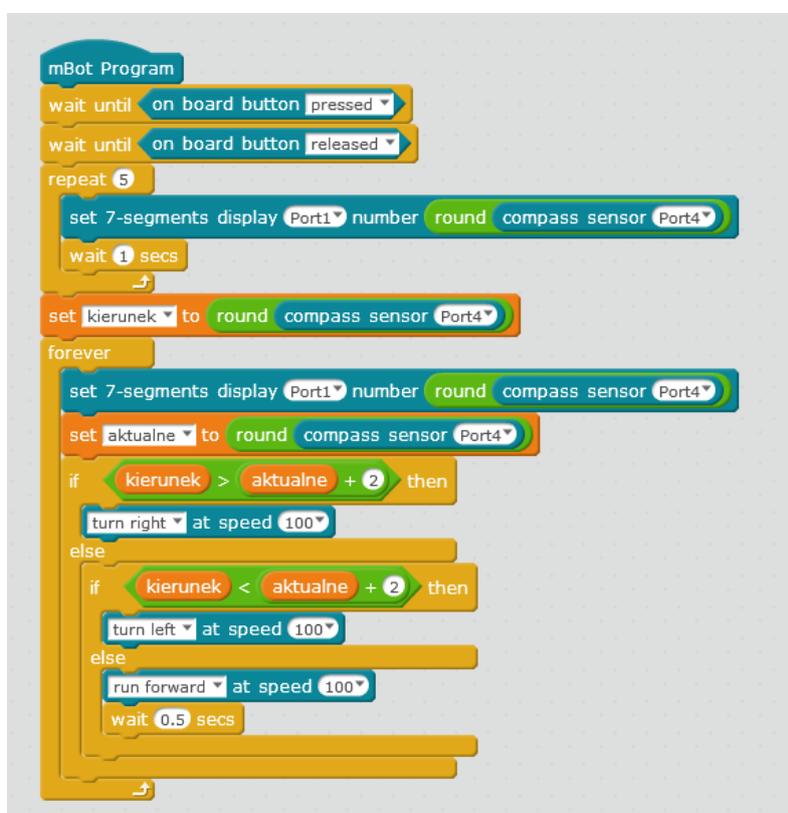
## Proceso de calibración:



Ejecutar el robot. Presione el botón blanco en el módulo de la brújula: el led azul debe parpadear. Usted está en un modo de calibración. Gire el robot lenta y suavemente alrededor de su eje vertical en más de 360 grados. Durante el giro el robot debe estar en posición horizontal. La mejor idea es colocar el robot en una hoja grande de papel y darle la vuelta sosteniendo una de las esquinas. Después del turno, presione nuevamente el botón blanco en el módulo para salir del modo de calibración. LED comienza a brillar continuamente.

### Problema: función hacia delante

Cada vez que queremos que el robot avance hacia delante, en lugar del habitual bloque de "avanzar", podemos copiar el procedimiento de que al principio de la ruta se va a recordar la pista y, a continuación, solo el robot cambia la dirección establecida, una corrección de la ruta va a ocurrir.



```
mBot Program
wait until on board button pressed
wait until on board button released
repeat 5
  set 7-segments display Port1 number round compass sensor Port4
  wait 1 secs
set kierunek to round compass sensor Port4
forever
  set 7-segments display Port1 number round compass sensor Port4
  set aktualne to round compass sensor Port4
  if kierunek > aktualne + 2 then
    turn right at speed 100
  else
    if kierunek < aktualne - 2 then
      turn left at speed 100
    else
      run forward at speed 100
      wait 0.5 secs
```

### Problema: giro de 180°

El siguiente bloque del programa corresponde al giro de 180 grados. En este momento, cuando el robot supera la línea negra, se guarda la dirección actual. A continuación, hay que encontrar la dirección contraria.

Si la dirección es menor que 180, la dirección del contador se cuenta sumando 180. Si el robot muestra un número mayor que 180, la dirección del contador se puede establecer restando 180.

Cuando conocemos ese valor, le decimos al robot que gire hasta que obtengamos la lectura correcta de la brújula.



```

define o180L
  set Kompas to round compass sensor Port1
  set granica to 0
  if Kompas < 180 then
    set granica to Kompas + 180
  else
    set granica to Kompas - 180
  repeat until round compass sensor Port1 = granica
  set Kompas to round compass sensor Port4
  set 7-segments display Port1 number round compass sensor Port4
  run forward at speed 0
  set motor M1 speed 100
run forward at speed 0

```

```

define 0180P
  set Kompas to round compass sensor Port4
  set granica to 0
  if Kompas < 180 then
    set granica to Kompas + 180
  else
    set granica to Kompas - 180
  repeat until round compass sensor Port1 = granica
  set Kompas to round compass sensor Port4
  set 7-segments display Port1 number round compass sensor Port4
  run forward at speed 0
  set motor M2 speed 100
run forward at speed 0

```

## Programa principal

Cuando resolvimos cada uno de los problemas por separado, es hora de poner estos elementos juntos.

Programa puede ser descrito como sigue:

- ❖ El robot comienza y sigue recto de acuerdo con las lecturas de la brújula hasta que se extiende sobre la línea negra.
- ❖ Cuando corre sobre la línea negra, gira a la derecha o izquierda.

```
mBot Program
wait until on board button pressed
wait until on board button released
set k to 0
repeat 3
  set 7-segmentsdisplay Port1 number round compasssensor Port4
  wait 1 secs
forever
  set 7-segmentsdisplay Port1 number round compasssensor Port4
  set kierunek to round compasssensor Port4
  repeat until line follower Port2 = 0
    set aktualne to round compasssensor Port4
    set 7-segmentsdisplay Port1 number round compasssensor Port4
    if kierunek > aktualne + 2 then
      turn right at speed 100
    else
      if kierunek < aktualne + 2 then
        turn left at speed 100
      else
        run forward at speed 100
        wait 0.5 secs
  if line follower Port2 = 0 and k = 1 then
    0180P
    set k to 0
  if line follower Port2 = 0 and k = 0 then
    0180L
    set k to 1
```

Ahora tenemos que agregar la condición, que detiene el robot cuando se acerca al objeto que se está buscando.

```
mBot Program
wait until on board button pressed
wait until on board button released
set k to 0
repeat 3
  set 7-segments display Port1 number round compass sensor Port4
  wait 1 secs
repeat until ultrasonic sensor Port3 distance < 6
  set 7-segments display Port1 number round compass sensor Port4
  set kierunek to round compass sensor Port4
  repeat until line follower Port2 = 0 or ultrasonic sensor Port3 distance < 6
    set aktualne to round compass sensor Port4
    set 7-segments display Port1 number round compass sensor Port4
    if kierunek > aktualne + 2 then
      turn right at speed 100
    else
      if kierunek < aktualne + 2 then
        turn left at speed 100
      else
        run forward at speed 100
        wait 0.5 secs
  if line follower Port2 = 0 and k = 1 then
    0180P
    set k to 0
  if line follower Port2 = 0 and k = 0 then
    0180L
    set k to 1
run forward at speed 0
set led on board all red 0 green 150 blue 0
play tone on note C4 beat Half
play tone on note E4 beat Half
set led on board all red 150 green 0 blue 0
```

## EVALUACIÓN DE LOS ESTUDIANTES

Después de la actividad, los estudiantes sabrán:

- ❖ Explicar cómo usar la brújula.
- ❖ Interpretar las lecturas de la brújula.
- ❖ Usa la brújula para mantener la dirección del robot.

### BIBLIOGRAFÍA

<https://trobot.pl/sklep/czujniki-moduly/makeblock-kompas/>

<http://learn.makeblock.com/en/me-compass/>

### ESCALABILIDAD

La brújula se puede usar para enseñar elementos de geografía; los estudiantes usan una brújula analógica para comparar las lecturas.