

# APRENDER CONECTADOS



Ministerio de Educación,  
Cultura, Ciencia y Tecnología  
Presidencia de la Nación

# RobotLab: Usando el sensor de ultrasonido y la grúa robótica



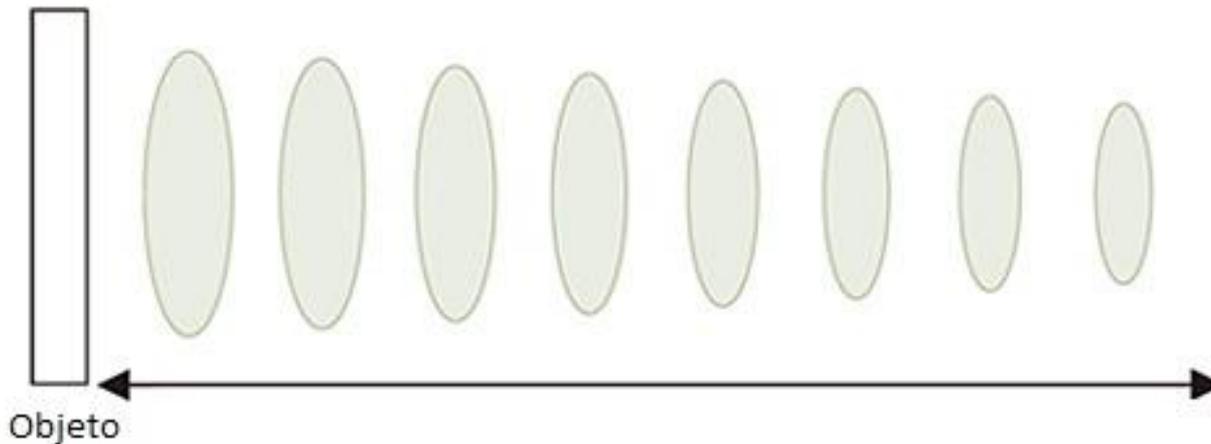
Este tutorial explica cómo se programa el dispositivo RobotLab, para que recorra un camino, se detenga cuando encuentra un obstáculo a cierta distancia y accione el brazo de la grúa robótica para tomar un elemento.

Se pueden plantear actividades de rescate o categorización de objetos tomando como referencia esta rutina de programación.



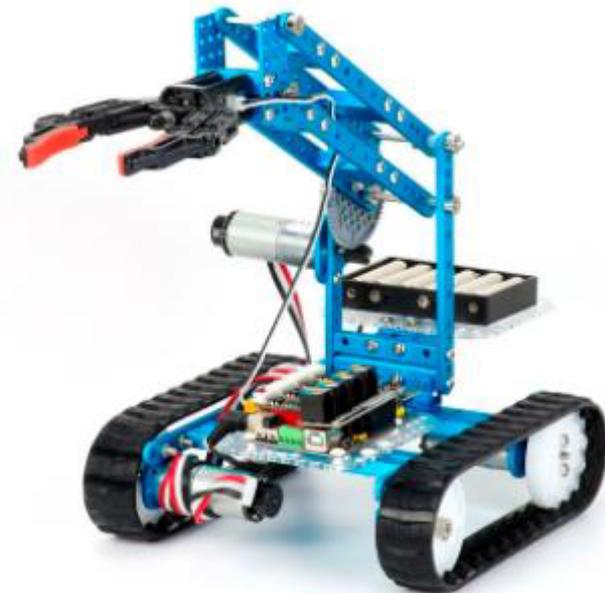
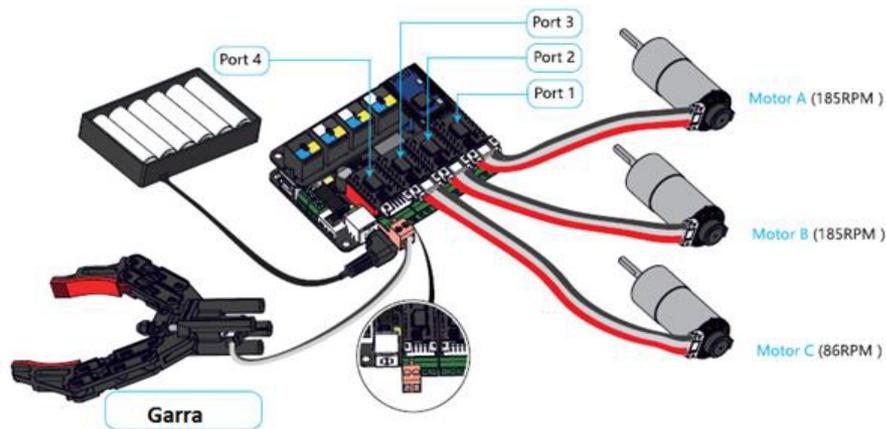
# Sensor de ultrasonido

Es un sensor que permite medir distancia mediante el uso de ondas ultrasónicas. El cabezal emite una onda y la recibe reflejada al retornar desde el objeto.



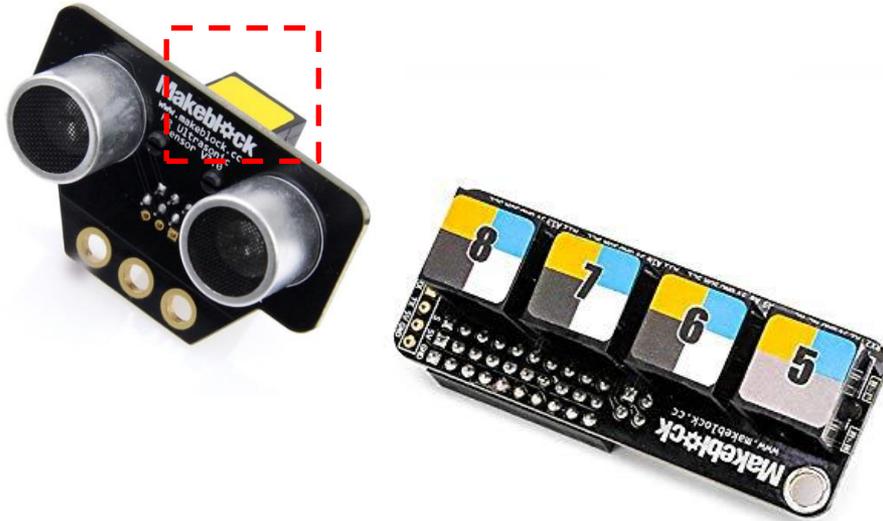
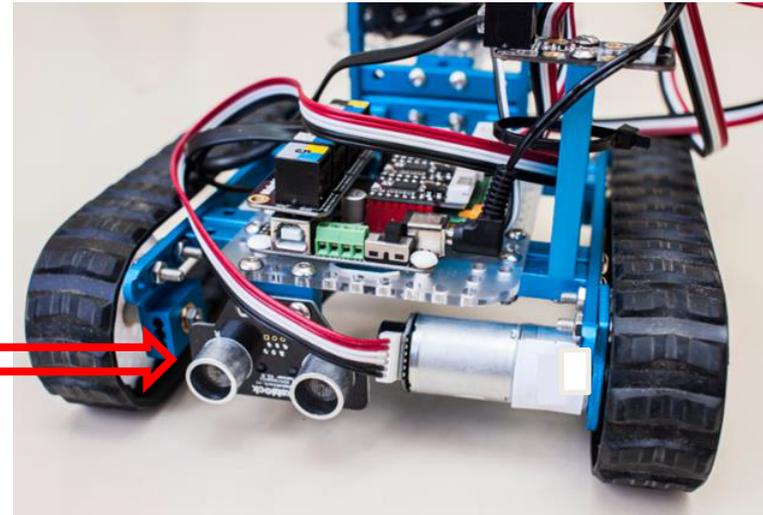
RobotLab permite la construcción de diversos modelos. Entre ellos, el RoboticArm Tank permite recorrer diferentes suelos y tomar objetos con su garra robótica.

Tanto los motores como los sensores deben ser conectados a una placa llamada MegaPi que permite su programación.



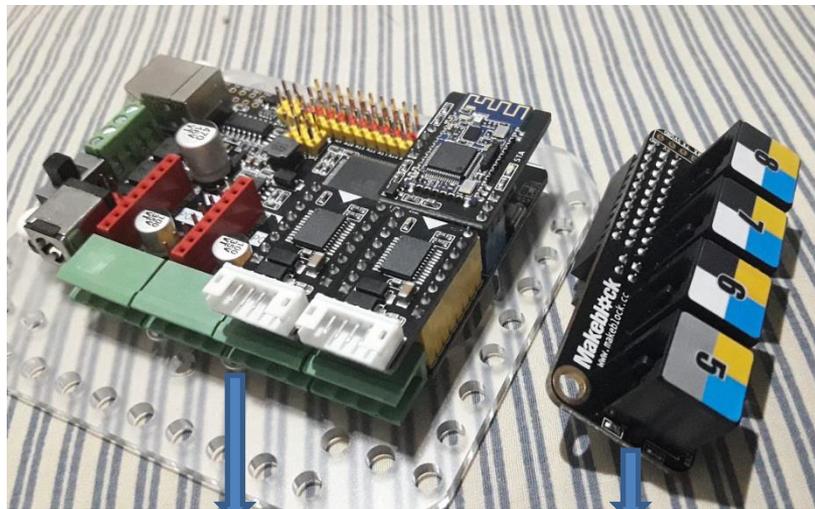
1

Al armar el Robotic Arm Tank (o cualquier otro modelo), es necesario incorporar el sensor de ultrasonido a la construcción.



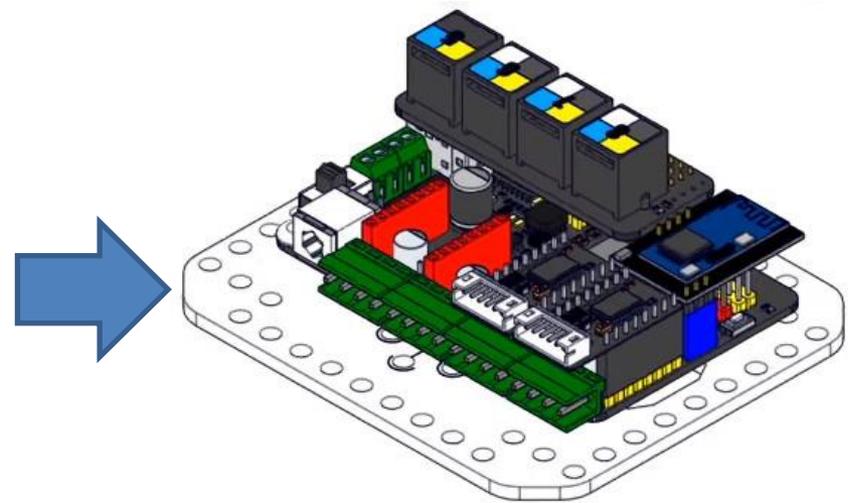
La ficha del sensor de ultrasonido esta categorizada con color amarillo, lo cual indica que su conexión es posible en los puertos que muestran la misma categorización de color.

Una vez realizada la conexión del sensor al puerto correspondiente del Shield RJ25, debemos adosar este último a la placa MegaPi.



Placa MegaPi

Shield RJ25



# Programando con mBlock



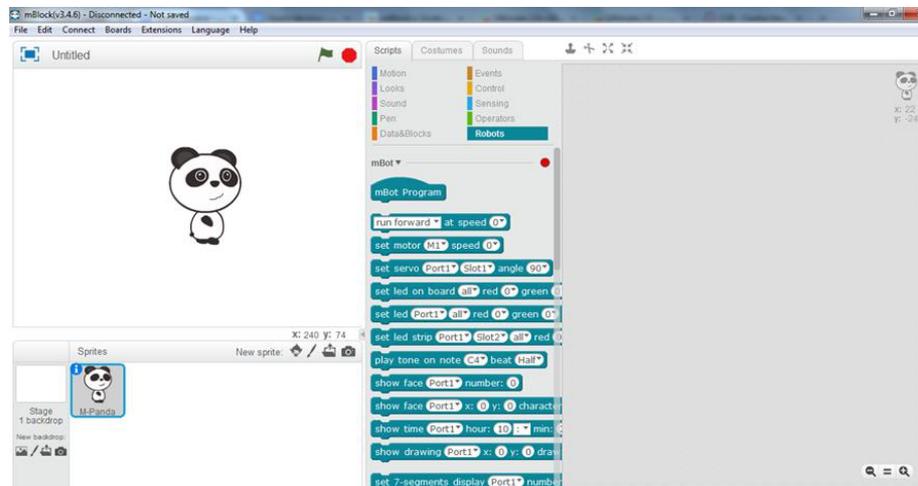
mblock 3

Based on Scratch 2.0.

Supports makeblock robots and other Arduino based hardware.

Para realizar esta actividad, es necesario:

- Construir el dispositivo robótico, equipado con el sensor de ultrasonido,
- Utilizar el entorno de programación mBlock para programar su funcionamiento.



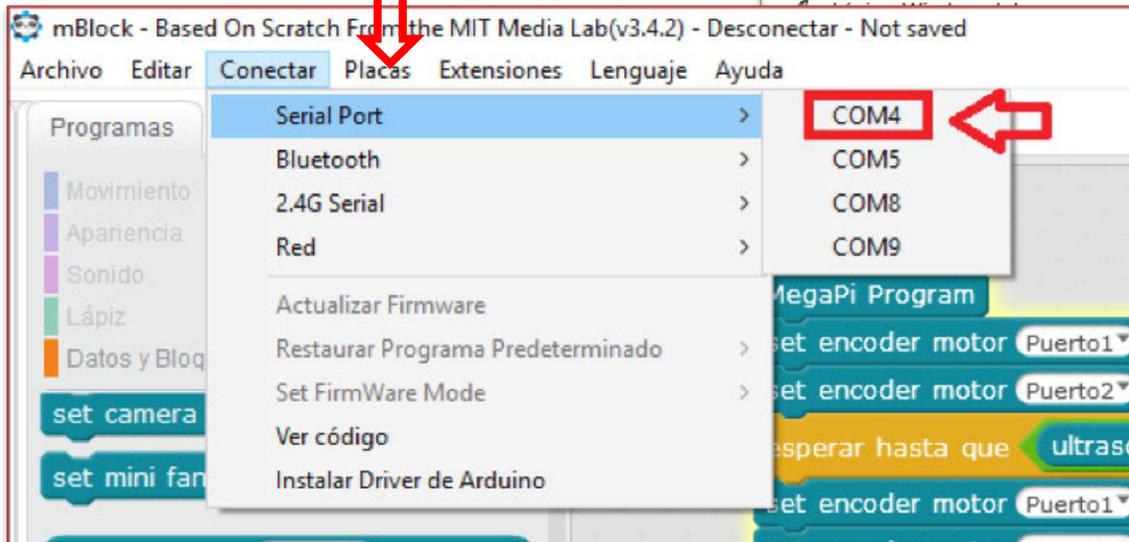
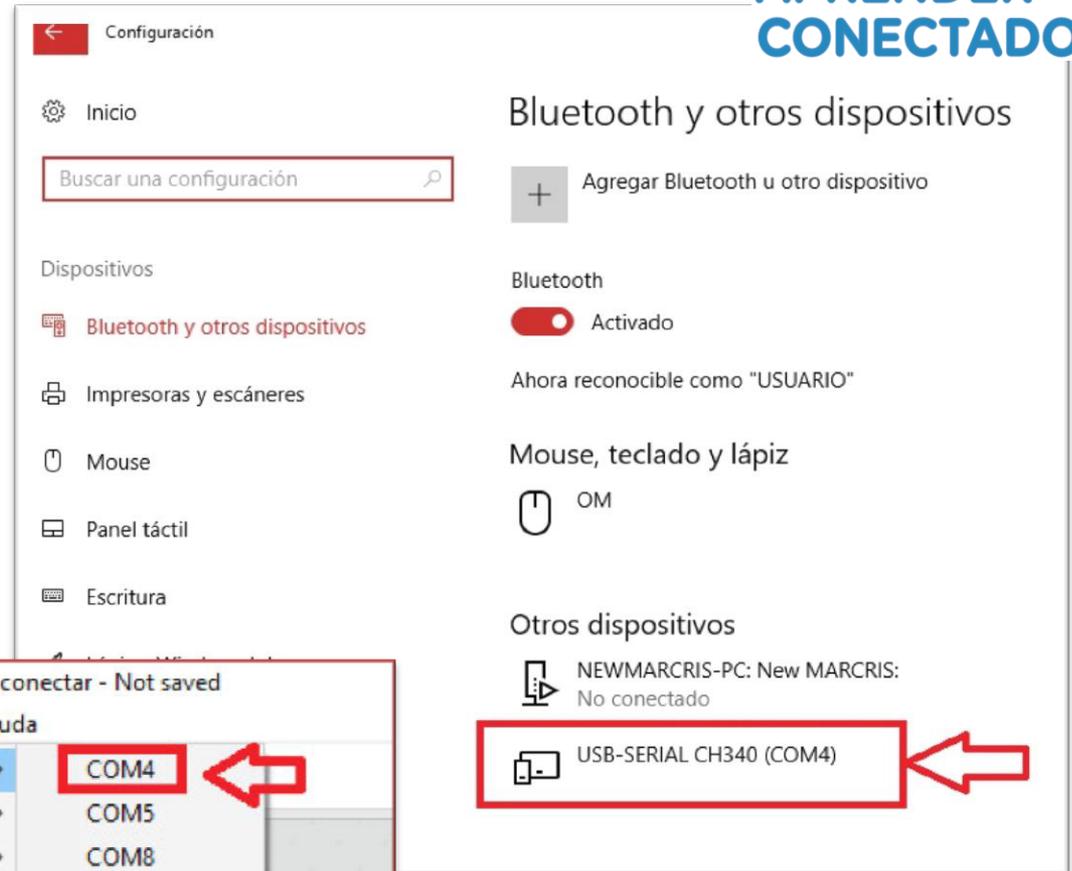
Pantalla de inicio del entorno de programación.



4

Para vincular al robot con la computadora a través del cable USB, debemos conectarlo, ir a configuración y dentro de “otros dispositivos” detectar el puerto activo nombrado COM.

Ya con ese dato, desde mBlock, se debe establecer conexión con el Serial Port correspondiente.



Señalar que la placa a programar es la MegaPi.  
Para trabajar en robótica, utilizaremos el panel “Robots”.

The screenshot shows the mBlock(v3.4.5) interface. The 'Placas' menu is open, listing various Arduino boards. 'Ultimate 2.0 (Mega Pi)' is selected with a checkmark. The 'Robots' category is highlighted in the block palette. The main workspace shows a program for an Arduino board with the following blocks: 'Programa de Arduino', 'leer pin digital 9', 'leer pin analógico (A) 0', 'read pulse pin 13 timeout 20000', 'fijar salida pin digital 9 a HIGH', 'fijar pin PWM 5 a 0', 'reproducir tono 9 en nota C4 beat 1', 'fijar ángulo del pin 9 del servo a 90', 'serial write text hola', 'serial available bytes', 'serial read byte', 'read ultrasonic sensor trig pin 13 echo', 'cronómetro', and 'reiniciar cronómetro'. The 'MegaPi' board is selected at the bottom of the workspace.

Comenzamos a escribir el código.  
Utilizamos un comando de inicio de rutina con la placa MegaPi

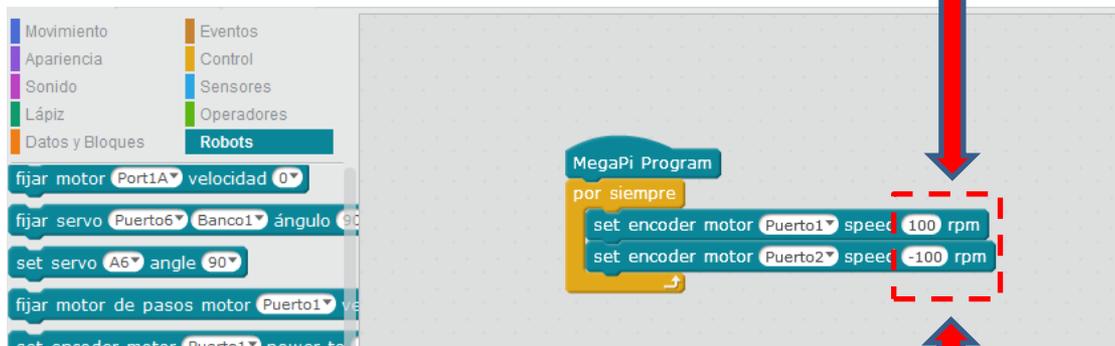
The screenshot displays the mBlock (v3.4.5) software interface. The main workspace shows a panda character on a white background. The left sidebar contains a palette of objects, including a 'M-Panda' object. The right sidebar shows a list of categories: Movimiento, Apariencia, Sonido, Lápiz, Datos y Bloques, Eventos, Control, Sensores, Operadores, and Robots. The 'Robots' category is selected, and a 'MegaPi Program' block is visible in the code area. The code block contains the following commands:

```
MegaPi Program
correr adelante a velocidad 0
correr adelante 1000 degrees at the
fijar motor Port1A velocidad 0
fijar servo Puerto6 Banco1 ángulo 90
set servo A6 angle 90
fijar motor de pasos motor Puerto1
set encoder motor Puerto1 power to
set encoder motor Puerto1 speed 180
set encoder motor Puerto1 rotate 100
fijar display 7-segments Puerto6 número
set led Puerto6 todos red 0 green
fijar tira led Puerto6 Banco2 todos
show face Puerto6 number: 0
mostrar cara Puerto6 x: 0 y: 0 cara
mostrar tiempo Puerto6 hora: 10 :
show drawing Puerto6 x: 0 y: 0 dra
fijar LED del sesnor de sonidos Puerto
set camera shutter Puerto6 as Press
set mini fan Puerto6 blow clockwise
```

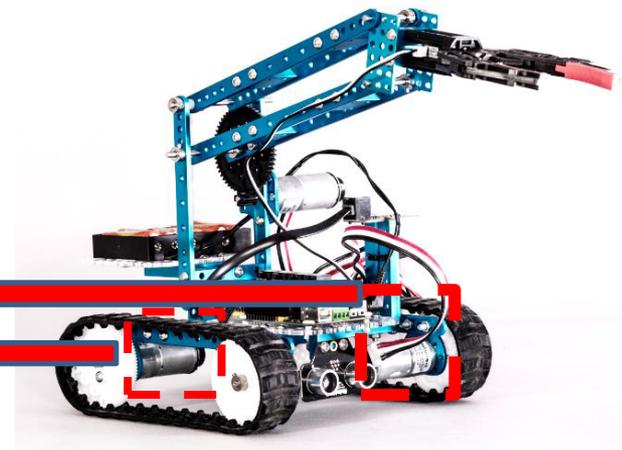
Dentro del panel naranja “Control” encontramos bucles y comandos de tiempo. Vamos a seleccionar un “por siempre” para indicar que el código se lea y se ejecute en loop permanente.

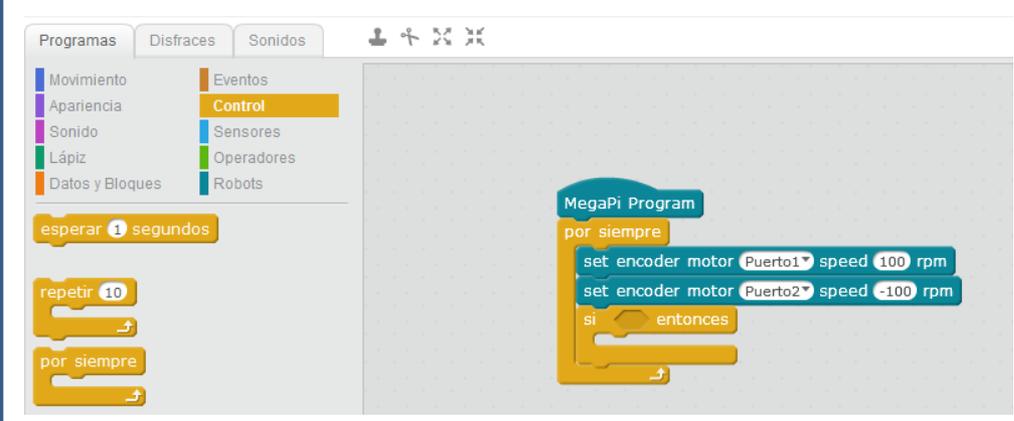
The screenshot shows the mBlock (v3.4.5) software interface. The window title is "mBlock(v3.4.5) - Desconectar - Not saved". The menu bar includes "Archivo", "Editar", "Conectar", "Placas", "Extensiones", "Lenguaje", and "Ayuda". The main workspace displays a cartoon panda character. The left sidebar shows the "Objetos" panel with a "M-Panda" object selected. The central panel is divided into categories: "Programas", "Disfraces", "Sonidos", "Eventos", "Control", "Sensores", "Operadores", and "Robots". The "Control" category is highlighted in orange. A list of control blocks is visible, including "esperar 1 segundos", "repetir 10", "por siempre", "si entonces", "si entonces", "si no", "esperar hasta que", "repetir hasta que", "detener todos", "cuando comience como clon", "crear clon de este objeto", and "borrar este clon". The "por siempre" block is highlighted in orange. The right sidebar shows the "MegaPi Program" block with a "por siempre" block selected. The coordinates "x: -24 y: 13" are displayed in the bottom right corner of the workspace.

Dentro del panel “Robots” elegimos el comando “set encoder motor..... speed.....rpm”. El mismo nos permite programar el encendido de motores (debiendo nombrar los puertos en donde están conectados) y la velocidad con la que se moverán. La dirección de rotación de cada motor, la designamos alterando en positivo o negativo el número que indica la velocidad asignada.

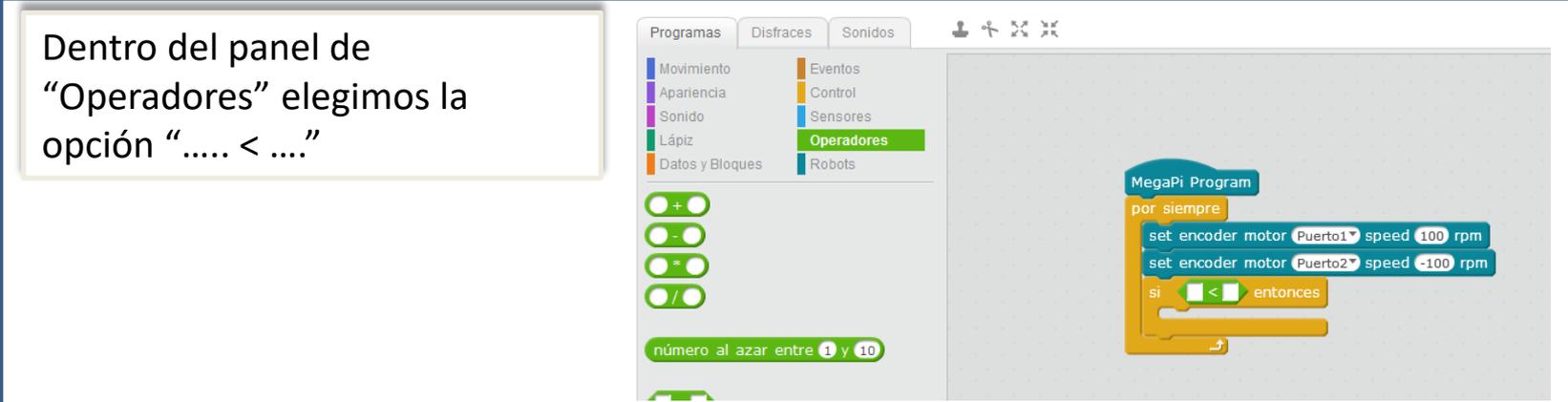


Si observamos el diseño, tenemos un motor que mueve una rueda trasera y el otro motor, ubicado en sentido invertido que lo hace con una delantera. Por ello, para que el robot realice la acción de ir hacia adelante, uno de los motores debe girar en sentido contrario, o en negativo.





Dentro del panel de “Control” elegimos un condicional.



Dentro del panel de “Operadores” elegimos la opción “..... < ....”

11

```
MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < entonces
```

Seleccionamos el comando “ultrasonic sensor ... distance” para que en la rutina se lean los datos ingresados a través del sensor. Debemos nombrar el puerto en el que aquel esta conectado.

12

Generamos una variable de cm de detección. En este ejemplo indicamos que se cumple la condición si halla un obstáculo a una distancia “menor a 15 cm”

```
MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < 15 entonces
```

13

```

MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < 15 entonces
    set encoder motor Puerto1 speed 0 rpm
    set encoder motor Puerto2 speed 0 rpm
    set encoder motor Puerto3 speed 150 rpm
  
```

Generamos que los motores de las ruedas se frenen y se active el tercer motor referente al brazo de la grúa.

14

Luego de 2 segundos, frenamos el brazo de la grúa y activamos la garra robótica para que se abra.

```

MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < 15 entonces
    set encoder motor Puerto1 speed 0 rpm
    set encoder motor Puerto2 speed 0 rpm
    set encoder motor Puerto3 speed 150 rpm
    esperar 2 segundos
    set encoder motor Puerto3 speed 0 rpm
    fijar motor Port4B velocidad 255
  
```



```

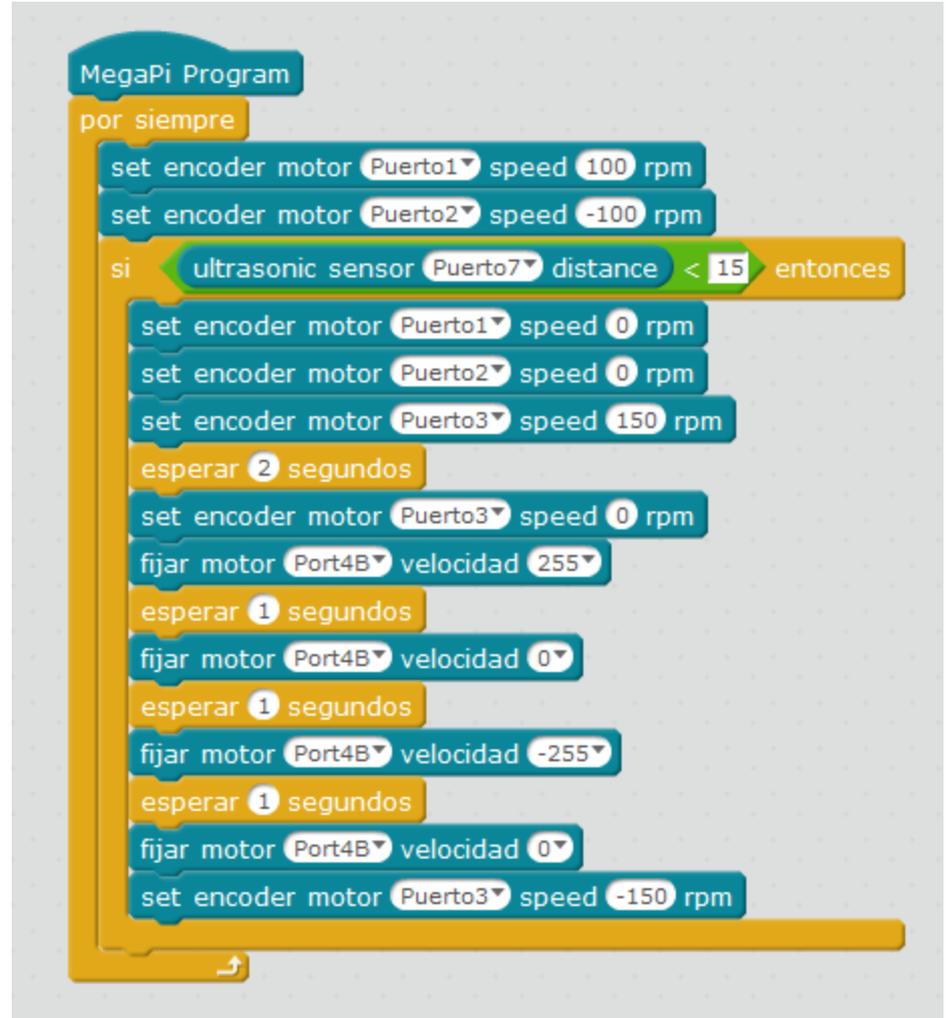
MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < 15 entonces
    set encoder motor Puerto1 speed 0 rpm
    set encoder motor Puerto2 speed 0 rpm
    set encoder motor Puerto3 speed 150 rpm
    esperar 2 segundos
    set encoder motor Puerto3 speed 0 rpm
    fijar motor Port4B velocidad 255
    esperar 1 segundos
    fijar motor Port4B velocidad 0
    esperar 1 segundos
    fijar motor Port4B velocidad -255
    esperar 1 segundos
    fijar motor Port4B velocidad 0
    set encoder motor Puerto3 speed -150 rpm
  
```

Luego de 1 segundo, frenamos el movimiento de la garra y activamos el motor en sentido inverso, generando que la garra robótica se cierre. Un segundo de espera en esta posición y luego subimos el brazo de grúa.



# Rutina terminada

De esta forma queda el código terminado. En este caso, determinamos que el robot avance hasta que encuentre un obstáculo a 15cm de distancia de él. Cuando se genera esta detección el brazo de la grúa robótica baja para tomar el objeto con la garra y alzarlo. Se pueden plantear actividades de rescate o categorización de objetos tomando como referencia esta rutina de programación.



```
MegaPi Program
por siempre
  set encoder motor Puerto1 speed 100 rpm
  set encoder motor Puerto2 speed -100 rpm
  si ultrasonic sensor Puerto7 distance < 15 entonces
    set encoder motor Puerto1 speed 0 rpm
    set encoder motor Puerto2 speed 0 rpm
    set encoder motor Puerto3 speed 150 rpm
    esperar 2 segundos
    set encoder motor Puerto3 speed 0 rpm
    fijar motor Port4B velocidad 255
    esperar 1 segundos
    fijar motor Port4B velocidad 0
    esperar 1 segundos
    fijar motor Port4B velocidad -255
    esperar 1 segundos
    fijar motor Port4B velocidad 0
    set encoder motor Puerto3 speed -150 rpm
```