





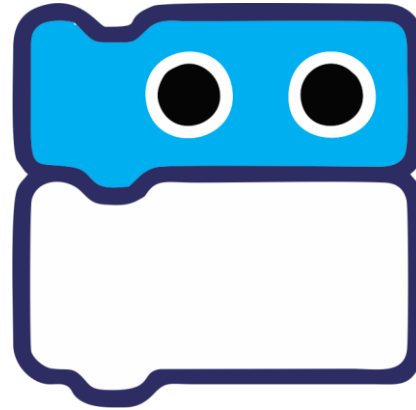
Robot to NEA
Academy



GRÍA DE PROGRAMACIÓN



MBLOCK



OTTO



CONFIGURACIÓN DE MBLOCK

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo Editar Conectar Placas **Extensiones** Lenguaje Ayuda

Untitled

- Administrar Extensiones Ctrl+Shift+T
- Restaurar Extensiones
- Limpiar cache

En primer lugar, vamos a ir al administrador de **EXTENSIONES...**

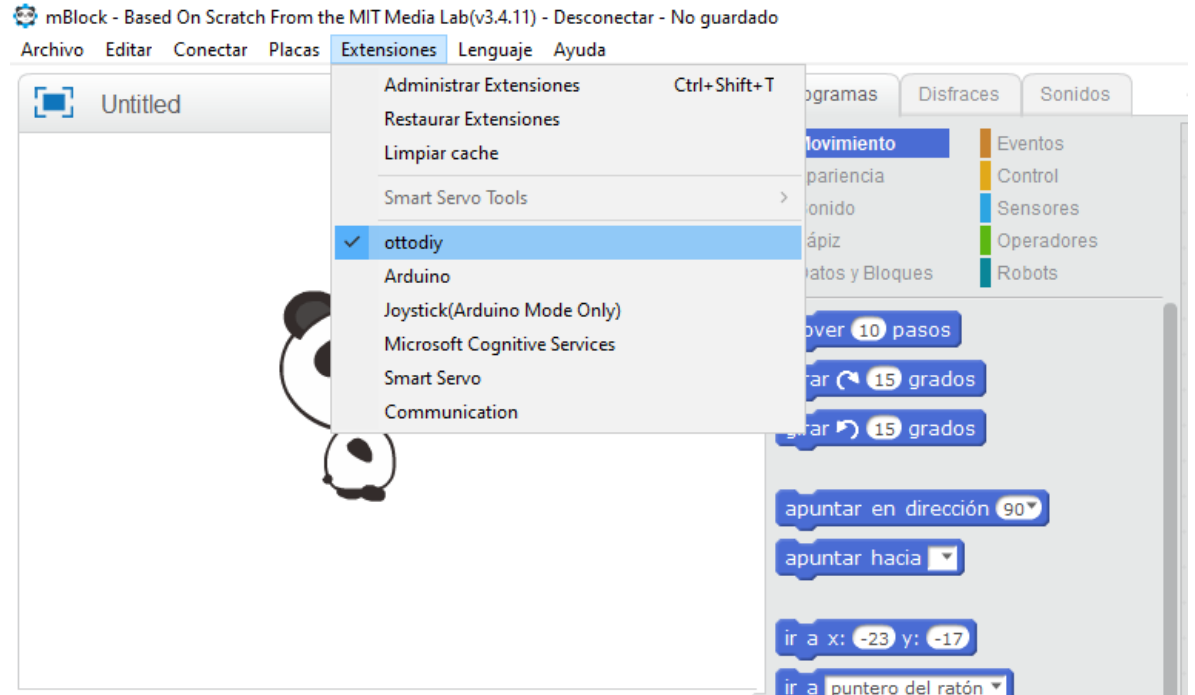
Después buscamos y bajamos la extensión **OTTODIY**

Extension	Author	Version	Buttons
ottodiy	Hien Phan	4.1	Descargado
ottodiyTR	Mehmet Aytakin HÜZ	2.0	Bajar



CONFIGURACIÓN DE MBLOCK

...luego seleccionamos **OTTODIY** como única extensión.





CONFIGURACIÓN DE MBLOCK

Ahora vamos a **PLACAS** y elegimos la placa: **ARDUINO NANO (MEGA328)**

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo Editar Conectar **Placas** Extensiones Lenguaje Ayuda

Untitled

- Arduino
 - Arduino Uno
 - Arduino Leonardo
 - ✓ Arduino Nano (mega328)
 - Arduino Mega 1280
 - Arduino Mega 2560
- Makeblock
 - Starter/Ultimate (Orion)
 - Me Uno Shield
 - mBot (mCore)
 - mBot Ranger (Auriga)
 - Ultimate 2.0 (MegaPi)
 - MegaPi Pro
- Otros
 - PicoBoard

Programas Disfraces Sonidos

- Movimiento
- Apariencia**
- Sonido
- Lápiz
- Datos y Bloques
- Eventos
- Control
- Sensores
- Operadores
- Robots

decir ¡Hola! por 2 segundos

decir ¡Hola!

pensar Hmm... por 2 segundos

pensar Hmm...

mostrar

esconder

cambiar disfraz a Panda-h



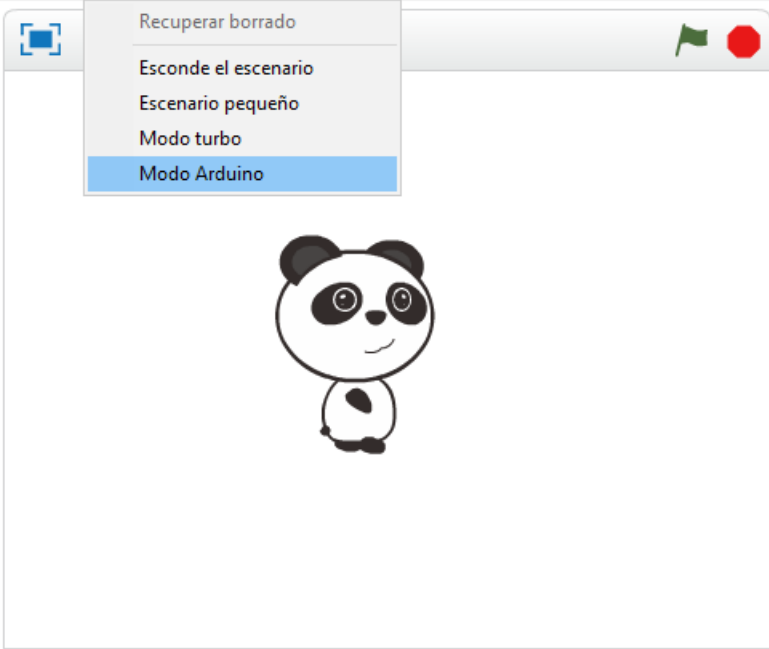
CONFIGURACIÓN DE MBLOCK

Después en **EDITAR**,
seleccionamos el
MODO ARDUINO...

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo **Editar** Conectar Placas Extensiones Lenguaje Ayuda

- Recuperar borrado
- Esconde el escenario
- Escenario pequeño
- Modo turbo
- Modo Arduino**



Programas Disfraces Sonidos

- Movimiento
- Apariencia**
- Sonido
- Lápiz
- Datos y Bloques
- Eventos
- Control
- Sensores
- Operadores
- Robots

decir ¡Hola! por 2 segundos

decir ¡Hola!

pensar Hmm... por 2 segundos

pensar Hmm...

mostrar

esconder

cambiar disfraz a Panda-b



CONFIGURACIÓN DE MBLOCK

...y vamos a ver como cambia el entorno de trabajo

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo Editar Conectar Placas Extensiones Lenguaje Ayuda

Programas

- Movimiento
- Apariencia
- Sonido
- Lápiz
- Datos y Bloques
- Eventos
- Control**
- Sensores
- Operadores
- Robots

esperar 1 segundos

repetir 10

por siempre

si entonces

si entonces

si no

esperar hasta que

repetir hasta que

x: 23
y: -17

Atrás Subir a Arduino Editar con IDE de Arduino

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9 }
10
11 void loop() {
12   _loop();
13 }
14
15 void _delay(float seconds){
16   long endTime = millis() + seconds * 1000;
17   while(millis() < endTime)_loop();
18 }
19
20 void _loop() {
21 }
```

send encode mode
 modo binario modo de caracteres

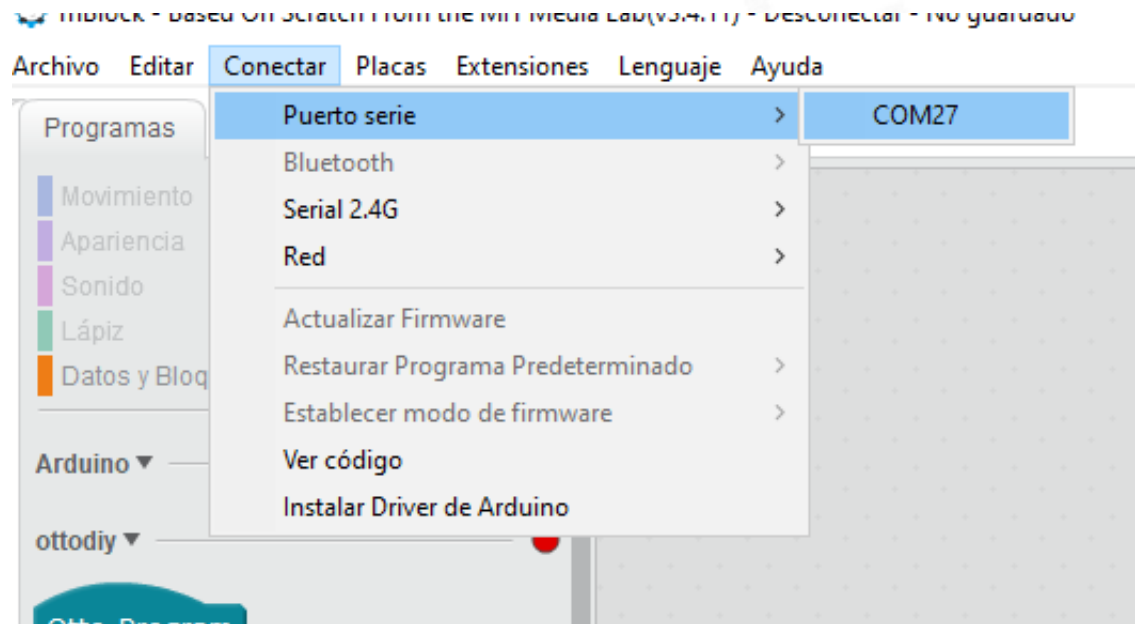
recv encode mode
 modo binario modo de caracteres

Enviar



CONECTAR A OTTO CON MBLOCK

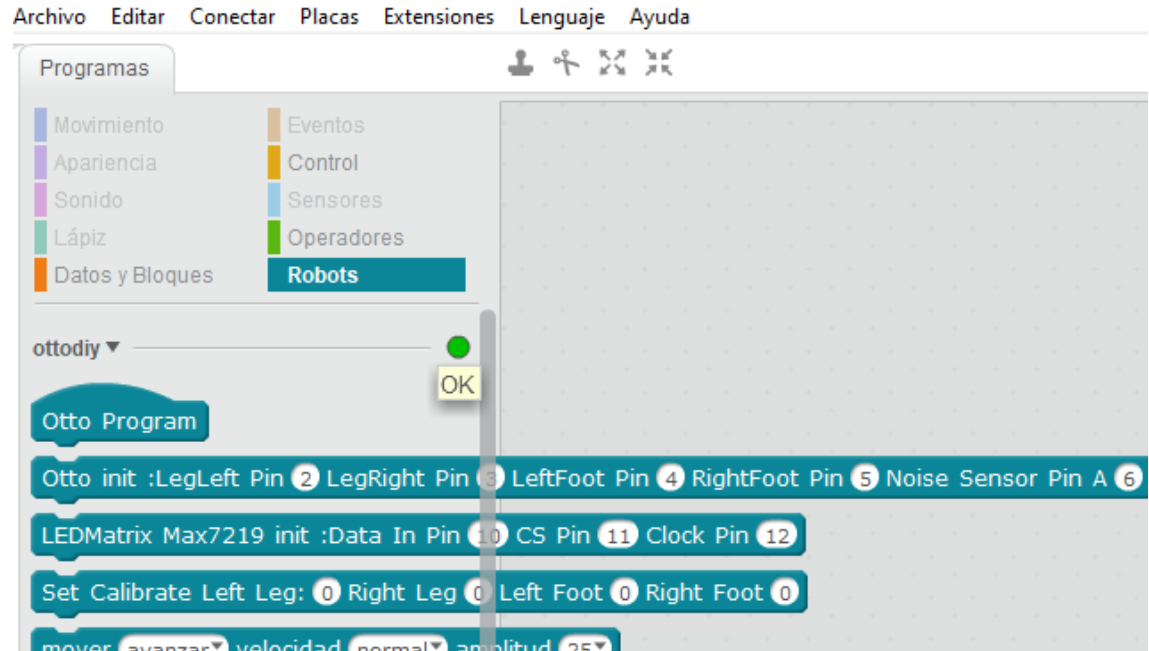
Una vez que conectamos el cable USB al robot y a la PC, vamos a ir a la pestaña **CONECTAR** y en puerto serie marcamos el puerto **COM_** que nos aparezca





CONECTAR A OTTO CON MBLOCK

Para comprobar que la conexión esta hecha, en los bloques de **ROBOT**, tendríamos que ver que el indicador de color, marque **VERDE** . Y si colocamos el cursor sobre él, vemos el mensaje de **OK**.





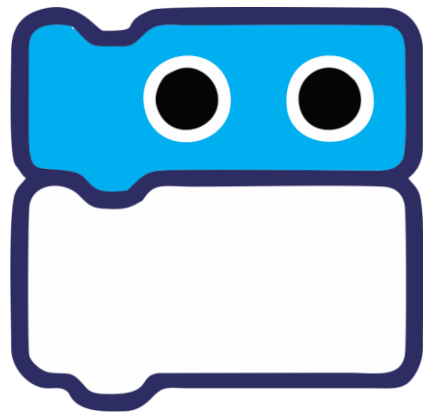
BLOQUES PARA PROGRAMAR A OTTO

En la categoría de bloques **ROBOT**, vamos a ver los bloques destinados a **OTTO**

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = 0;
6 double angle_deg = 0;
7
8 void setup() {
9   // Initialize pins
10  // ...
11  void loop() {
12    _loop();
13  }
14
15 void _delay(float long endTime = millis() + long; while(millis() < endTime);
16
17 void _loop() {
21 }
```



PROGRAMAMOS MOVIMIENTOS





PROGRAMAMOS A OTTO

El primer bloque siempre tiene que ser el bloque cabecera **OTTO PROGRAM.**

Abajo del bloque cabecera ponemos el **bloque de inicializar al robot.**

En este bloque vamos a encontrar la **configuración de los pines** de los motores, el sensor ultrasónico y el buzzer.

The screenshot shows the Arduino IDE interface. On the left, the 'Robots' block palette is visible, with the 'Otto Program' block selected. The main workspace contains a block-based program for Otto robot. The blocks are: 'Otto Program', 'Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise Sensor Pin A 6 BuzzerPin 13 SRF04 Trig 8 SRF04 Echo 9', 'LEDMatrix Max7219 init :Data In Pin 4', 'Set Calibrate Left Leg: 0 Right Leg 0', 'mover avanzar velocidad normal', 'gesto feliz', 'reproducir conexión', 'reproducir nota C4 alta medio', 'mostrar boca sonrisa', 'mostrar animación de boca pequeño u', 'Borrar boca', 'Medir Distancia', 'Medir ruido', 'medir luz en izquierda', 'botón llave1 pressed', 'serial write text hola', 'bytes disponibles en el serial', and 'byte leído en el serial'. On the right, the C++ code is displayed, corresponding to the blocks. The code includes headers for wiring, software serial, servo, EEPROM, and a custom header 'Otto.h'. It defines variables for pin numbers and a double for angle. The setup function calls Otto.init with the pin numbers. The loop function calls Otto.move with 'avanzar' and 'normal' velocity, Otto.gestor with 'feliz', and Otto.reproducir with 'conexión' and 'C4' note. It also calls Otto.mostrarBoca with 'sonrisa' and Otto.mostrarAnimacion with 'pequeño u'. The code ends with a delay of 1000ms.

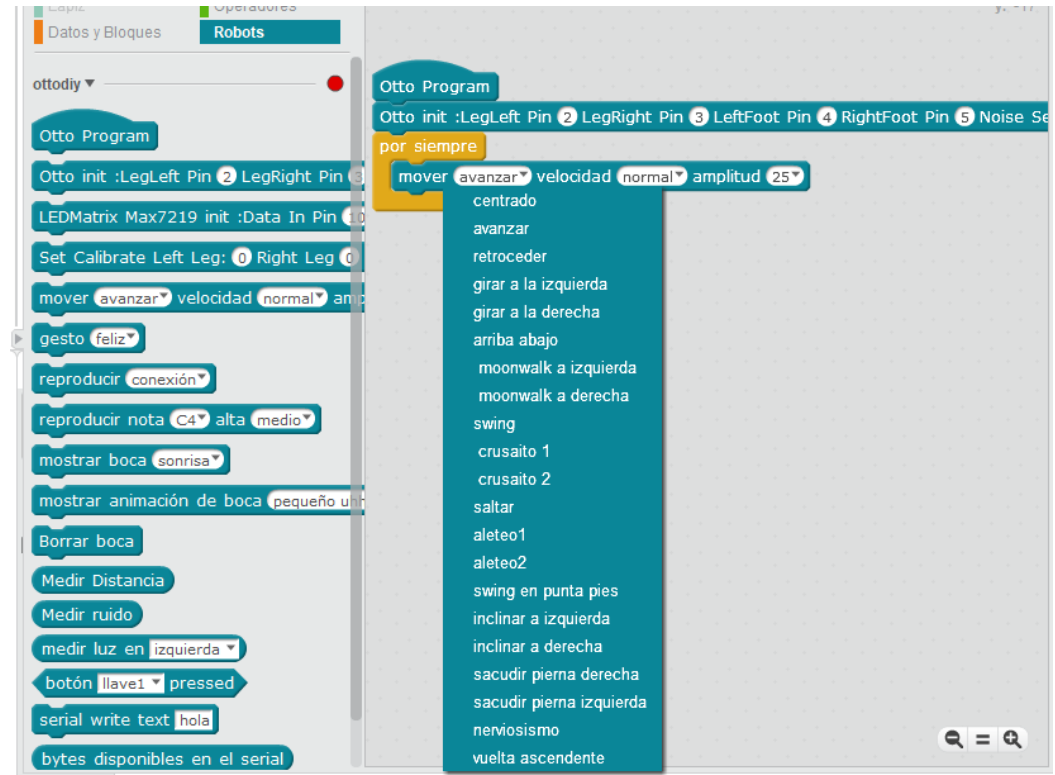
```
4 #include <wiring.h>
3 #include <SoftwareSerial.h>
4
5 #include <Servo.h>
6 #include <EEPROM.h>
7 #include "Otto.h"
8
9 double angle_rad = PI/180.0;
10 double angle_deg = 180.0/PI;
11 Otto Ottobot;
12
13 void setup() {
14     Ottobot.init(2,3,4,5,true,A6,13,8,9);
15 }
16
17 void loop() {
18     _loop();
19 }
20
21 void _delay(float seconds) {
22     long endTime = millis() + seconds * 1000;
```



PROGRAMAMOS A OTTO

Después ponemos el bloque **POR SIEMPRE** y adentro, el bloque de **MOVER**.

El bloque **MOVER**, se puede configurar con varios movimientos (avanzar, retroceder, girar, moonwalk, entre otros).





PROGRAMAMOS A OTTO

También podemos variar la velocidad.

The screenshot shows the LEGO Mindstorms software interface with the 'Robots' tab selected. The program is titled 'Otto Program' and includes the following blocks:

- Otto Program
- Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise Se
- por siempre
- mover avanzar velocidad normal amplitud 25
- gesto feliz
- reproducir conexión
- reproducir nota C4 alta medio
- mostrar boca sonrisa
- mostrar animación de boca pequeño u
- Borrar boca
- Medir Distancia
- Medir ruido
- medir luz en izquierda
- botón llave1 pressed
- serial write text hola
- bytes disponibles en el serial

A dropdown menu is open over the 'mover' block, showing the following options:

- muy rápido
- rápido
- normal
- lento
- muy lento



SUBIMOS EL PROGRAMA

Una vez que completamos el programa, solo nos queda hacer clic en el botón **SUBIR A ARDUINO**, y esperar que se cargue por completo.

The screenshot shows the Arduino IDE interface. On the left, a block-based program is visible with the following structure:

- Otto Program
- Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise
- por siempre
- mover avanzar velocidad normal amplitud 25

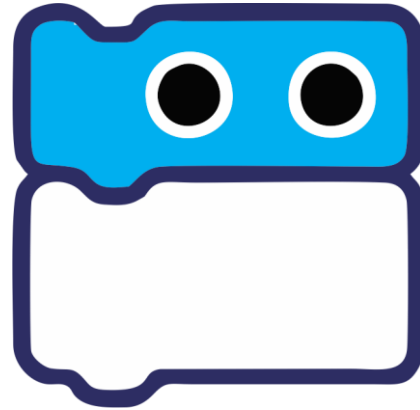
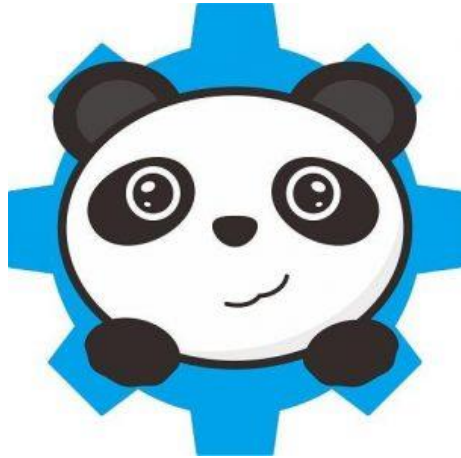
On the right, the C++ code is displayed:

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 #include <Servo.h>
6 #include <EEPROM.h>
7 #include "Otto.h"
8
9 double angle_rad = PI/180.0;
10 double angle_deg = 180.0/PI;
11 Otto Ottobot;
12
13 void setup() {
14     Ottobot.defaultInit();
15     Ottobot.init(2,3,4,5,ts);
16 }
```

Navigation buttons 'Atrás' and 'Subir a Arduino' are visible at the top right of the code editor area.



PROGRAMAMOS UN BAILE USANDO EL SENSOR ULTRASÓNICO





PROGRAMAMOS A OTTO

En primer lugar colocamos el bloque **OTTO PROGRAM** y abajo el bloque **OTTO INIT...**

The screenshot shows the Arduino IDE interface. On the left, the 'Robots' library is open, displaying a list of blocks for the Otto robot. The 'Otto Program' block is highlighted at the top. Below it, the 'Otto init' block is shown with its pin configuration: LegLeft Pin 2, LegRight Pin 3, LeftFoot Pin 4, RightFoot Pin 5, Noise Sensor Pin A, BuzzerPin 13, SRF04 Trig 8, and SRF04 Echo 9. Other blocks include LEDMatrix Max7219, Set Calibrate Left Leg, mover, gesto, reproducir, reproducir nota, mostrar boca, mostrar animación de boca, Borrar boca, Medir Distancia, Medir ruido, medir luz en, botón, serial write text, bytes disponibles en el serial, and byte leído en el serial.

On the right, the C++ code for the Otto robot is displayed. The code includes headers for wiring, SoftwareSerial, Servo, EEPROM, and 'Otto.h'. It defines a double variable for angle_rad and a double variable for angle_deg. The setup function initializes the Otto robot with the specified pin configuration. The loop function calls _loop(). A delay function is also defined.

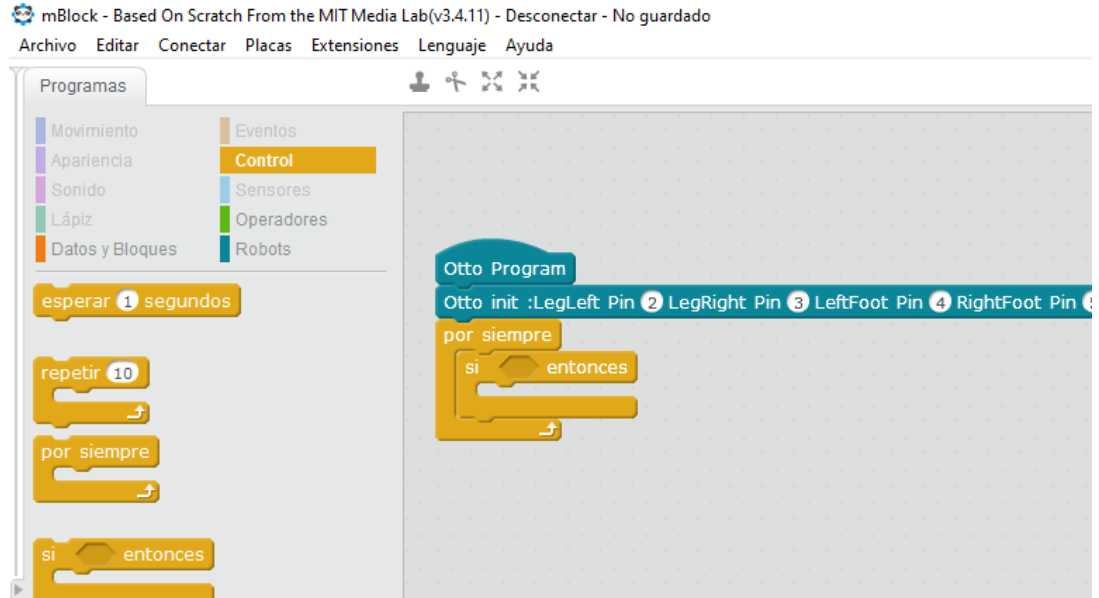
```
4 #include <wiring>
3 #include <SoftwareSerial.h>
4
5 #include <Servo.h>
6 #include <EEPROM.h>
7 #include "Otto.h"
8
9 double angle_rad = PI/180.0;
10 double angle_deg = 180.0/PI;
11 Otto Ottobot;
12
13 void setup() {
14     Ottobot.init(2,3,4,5,true,A6,13,8,9);
15 }
16
17 void loop() {
18     _loop();
19 }
20
21 void _delay(float seconds) {
22     long endTime = millis();
    ...
23     long seconds * 1000
```

send encode mode modo binario modo de caracteres
recv encode mode modo binario modo de caracteres



PROGRAMAMOS A OTTO

...después buscamos el bloque **POR SIEMPRE** y el **SI ENTONCES**, y los colocamos como vemos en la imagen





PROGRAMAMOS A OTTO

Luego agarramos el bloque **MENOR QUE (“<”)**, que se encuentra en la categoría **OPERADORES**

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo Editar Conectar Placas Extensiones Lenguaje Ayuda

Programas

- Movimiento
- Apariencia
- Sonido
- Lápiz
- Datos y Bloques
- Eventos
- Control
- Sensores
- Operadores**
- Robots

Otto Program

Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise

por siempre

si < entonces

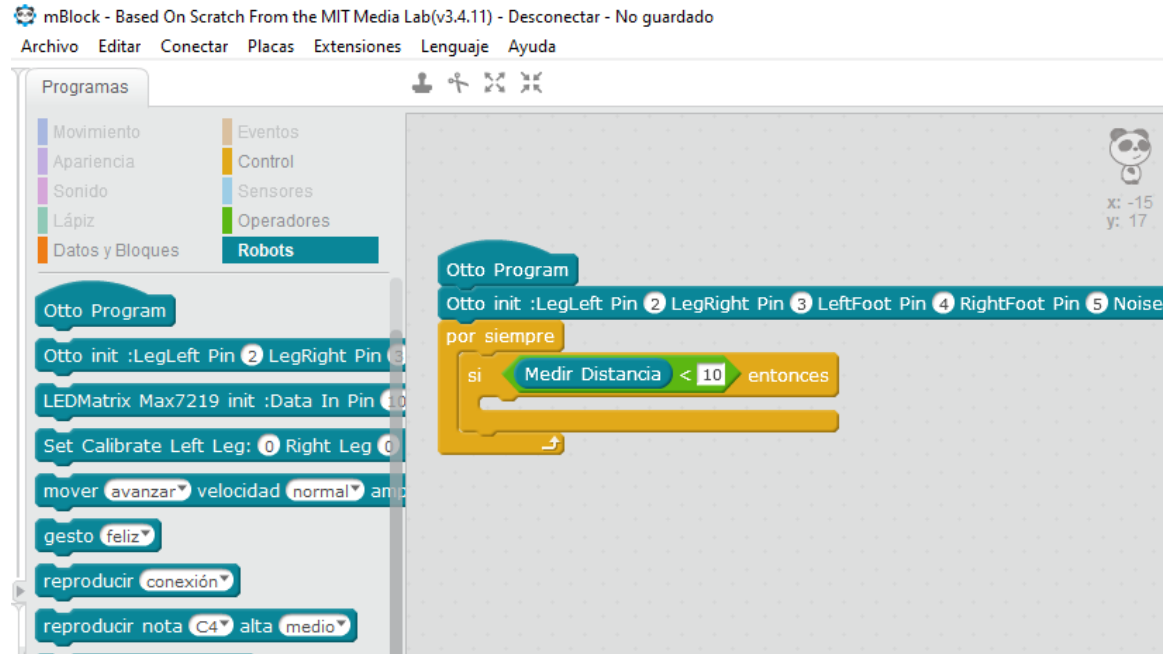
1
1
1
1
1
1
1
1
1
1



PROGRAMAMOS A OTTO

Ahora buscamos el bloque del sensor **ULTRASÓNICO: MEDIR DISTANCIA**, y lo colocamos dentro del **primer espacio** del sensor **MENOR QUE**.

En el **segundo espacio** ponemos la distancia en la cual se va a **activar el sensor**, por ejemplo 10





SUBIMOS EL PROGRAMA

Dentro del bloque **SI ENTONCES**,
ponemos el bloque de **MOVER**, y lo
configuramos como se ve en la
imagen

The screenshot shows the mBlock software interface. At the top, it says "mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado". Below that is a menu bar with "Archivo", "Editar", "Conectar", "Placas", "Extensiones", "Lenguaje", and "Ayuda".

The main workspace is divided into two panes. On the left is the "Programas" pane, which shows a list of categories: Movimiento, Apariencia, Sonido, Lápiz, Datos y Bloques, Eventos, Control, Sensores, Operadores, and Robots. The "Robots" category is selected. Below the categories, there is a dropdown menu for "ottodiv" and a list of blocks: "Otto Program", "Otto init :LegLeft Pin 2 LegRight Pin 3", "LEDMatrix Max7219 init :Data In Pin 10", "Set Calibrate Left Leg: 0 Right Leg 0", "mover avanzar velocidad normal amplitud 40", and "gesto feliz".

On the right is the main script area, which shows a script for "Otto Program". The script starts with "Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise". This is followed by a "por siempre" (forever) loop. Inside the loop, there is a "si Medir Distancia < 10 entonces" (if distance measured < 10 then) block. Inside this "if" block, there is a "mover avanzar velocidad normal amplitud 40" (move forward velocity normal amplitude 40) block.



SUBIMOS EL PROGRAMA

Después ponemos un bloque de **ESPERAR** y completamos agregando un **movimiento más** y una **nota musical**

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Desconectar - No guardado

Archivo Editar Conectar Placas Extensiones Lenguaje Ayuda

Programas

- Movimiento
- Apariencia
- Sonido
- Lápiz
- Datos y Bloques
- Eventos
- Control
- Sensores
- Operadores
- Robots

ottodiy ▾

Otto Program

Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise

LEDMatrix Max7219 init :Data In Pin 10

Set Calibrate Left Leg: 0 Right Leg 0

mover avanzar velocidad normal amplitud 40

gesto feliz

reproducir conexión

Otto Program

Otto init :LegLeft Pin 2 LegRight Pin 3 LeftFoot Pin 4 RightFoot Pin 5 Noise

por siempre

si Medir Distancia < 10 entonces

mover avanzar velocidad normal amplitud 40

esperar 1 segundos

mover retroceder velocidad normal amplitud 40

esperar 1 segundos

reproducir nota A4 alta medio



SUBIMOS EL PROGRAMA

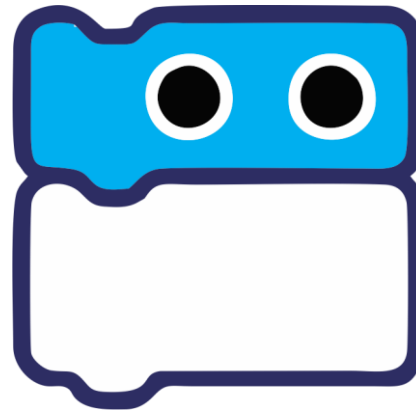
Como último paso subimos el programa al robot

```
Atrás Subir a Arduino Editar con
```

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 #include <Servo.h>
6 #include <EEPROM.h>
7 #include "Otto.h"
8
9 double angle_rad = PI/180.0;
10 double angle_deg = 180.0/PI;
11 Otto Ottobot;
12
13 void setup() {
14     Ottobot.defaultInit();
15     Ottobot.init(2,3,4,5,true,A6,13,8,9);
16 }
```

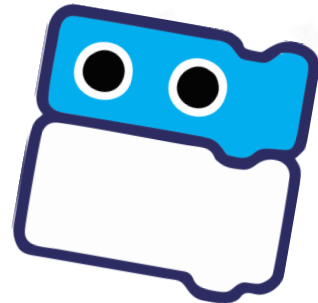



¿TE ANIMAS A ESCRIBIR EL SIGUIENTE PROGRAMA?





A PROGRAMAR!



```
Otto Program X: -15
Otto init :LegLeftPin 2 LegRightPin 3 LeftFootPin 4 RightFootPin 5 Noise Sensor Pin A 6 Buz
por siempre
si Medir Distancia < 10 entonces
  mover avanzar velocidad normal amplitud 40
  esperar 1 segundos
  mover swing velocidad normal amplitud 40
  esperar 1 segundos
  mover swing velocidad normal amplitud 40
  esperar 1 segundos
  mover avanzar velocidad normal amplitud 40
  esperar 1 segundos
  reproducir nota A4 alta medio
  reproducir nota A4 alta medio
  reproducir nota A4 alta medio
  reproducir nota F4 alta cuarto
  reproducir nota C5 alta cuarto
  reproducir nota A4 alta medio
  reproducir nota F4 alta cuarto
  reproducir nota C5 alta cuarto
  reproducir nota A4 alta medio
```

CONTINUA ACA

```
esperar 1 segundos
reproducir nota E5 alta medio
reproducir nota E5 alta medio
reproducir nota E5 alta medio
reproducir nota F5 alta cuarto
reproducir nota C5 alta cuarto
reproducir nota A4 alta medio
reproducir nota F4 alta cuarto
reproducir nota C5 alta cuarto
reproducir nota A4 alta medio
```

