

1. Preparados

- [¿Qué es mBot?](#)
- [mBot y mBlock](#)
- [Arduino: El corazón de mBot](#)
- [Componentes exteriores](#)
- [Programación mBlock](#)
- [Preparando mBlock](#)
- [IMPORTANTE: ACTUALIZAR FIRMWARE](#)
- [EN VIVO ¿Eso qué es?](#)
- [EN VIVO conexión 2.4 Bluetooth o cable](#)
- [CARGAR](#)
- [EN VIVO vs CARGAR](#)
- [A JUGARRRRR !!!!](#)

¿Qué es mBot?

No confundas

mBot objetivo de este curso, basado en Arduino unos 125€ con matriz LED

<https://www.robotix.es/es/mbot>

mBot2 basado en Cyberpi unos 200€ <https://www.robotix.es/es/mbot-2>

mBot es un robot educativo de la empresa [Makeblock](#), que persigue los siguientes objetivos:

1. El objetivo principal es **desarrollar el pensamiento computacional** en el alumnado motivado por la ejecución de órdenes en algo físico como es el robot.
2. El robot está diseñado para su **uso escolar**: resistente y económico
3. Basado en hardware **libre** y software **libre**

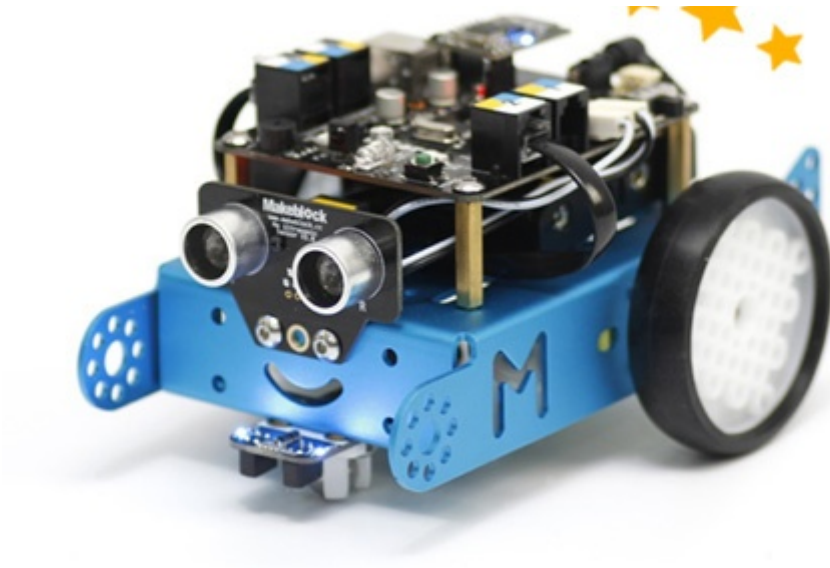
El objetivo 1 da como resultado que el kit de mBot sólo permite una configuración posible más los complementos que tiene el kit que proporciona el kit de CATEDU, **no hay tiempos en construcción, sólo en programación**, esto es otra filosofía diferente frente a otras alternativas como los robots de LEGO, que el tiempo en construcción y creatividad es importante. No queremos defender qué alternativa es buena y cual mala, sino que el docente tiene que decidir qué objetivos quiere perseguir, según lo que quiere tiene que decidir qué producto es el adecuado.

El objetivo 2 No podemos hablar de este objetivo pues actualmente este equipo no se fabrica Las características y precios de mBot 2.4G se puede ver en [este enlace](#).

El objetivo 3 el hardware libre está materializado en que se basa en la placa **ARDUINO**, que lo han personalizado con más sensores y conexiones rápidas RJ11 (la clavija de teléfono fijo). El software libre en este robot está en **el programa mBlock** que está basado en el software de programación [Scratch](#), diseñado para desarrollar el pensamiento computacional en los niños, ampliamente usado en todo el mundo, el cual le añaden unas librerías propias del robot, dando como resultado [mBlock](#).

El software mBlock es el resultado de instalar el [firmware de la placa Arduino](#), instalar las [librerías](#) del robot en el Scratch, [actualizar](#), etc... recomendamos lo práctico: Descargar el [mBlock](#) directamente que lo tiene todo ya preparado.

Dentro de esta filosofía de libertad, los agujeros son compatibles con LEGO ampliando las posibilidades.

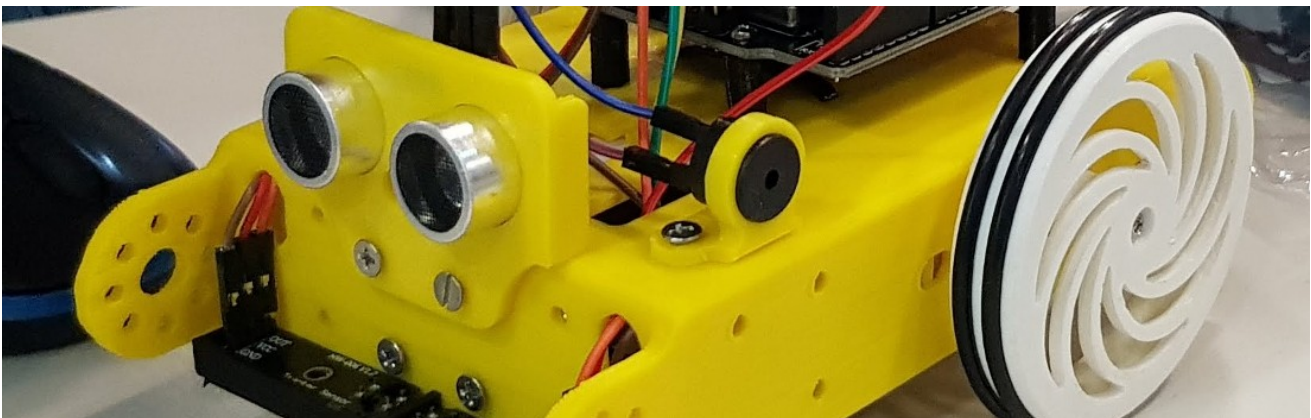


Fuente: <http://makeblock.es/>

¿Se puede uno fabricar un "mBot"?

SI, pues prácticamente es un Arduino con motores y sensores bastante estándares, y encima el programa mBlock es gratis, lo único es que no te saldrá tan perfecto. ¿Cómo se hace? para esto ya hay un curso en Aularagon MCLON <https://libros.catedu.es/books/mclon-con-nanoarduino>

y un grupo en Twitter: @mClonRobot



¿Se puede simular?



¿Qué no tienes robot? no pasa nada, hay buenos simuladores

<https://scratch.mit.edu/projects/788113358/editor/>

mBot y mBlock

Requisitos de conceptos previos

Es necesario haber realizado una **FORMACIÓN BÁSICA EN SCRATCH** la formación exclusiva de [Scratch](#) en [Aularagón](#) es suficiente.

Requisitos de material

Necesitas el **kit de robótica mBot** de MakeBlock que presta CATEDU :

- [mBot con mochila de comunicación 2.4G](#). (actualmente solo se vende la versión con Bluetooth)
- [Matriz de LEDs 8x16](#) donde aumentaremos la capacidad de interacción de nuestro simpático mBot.

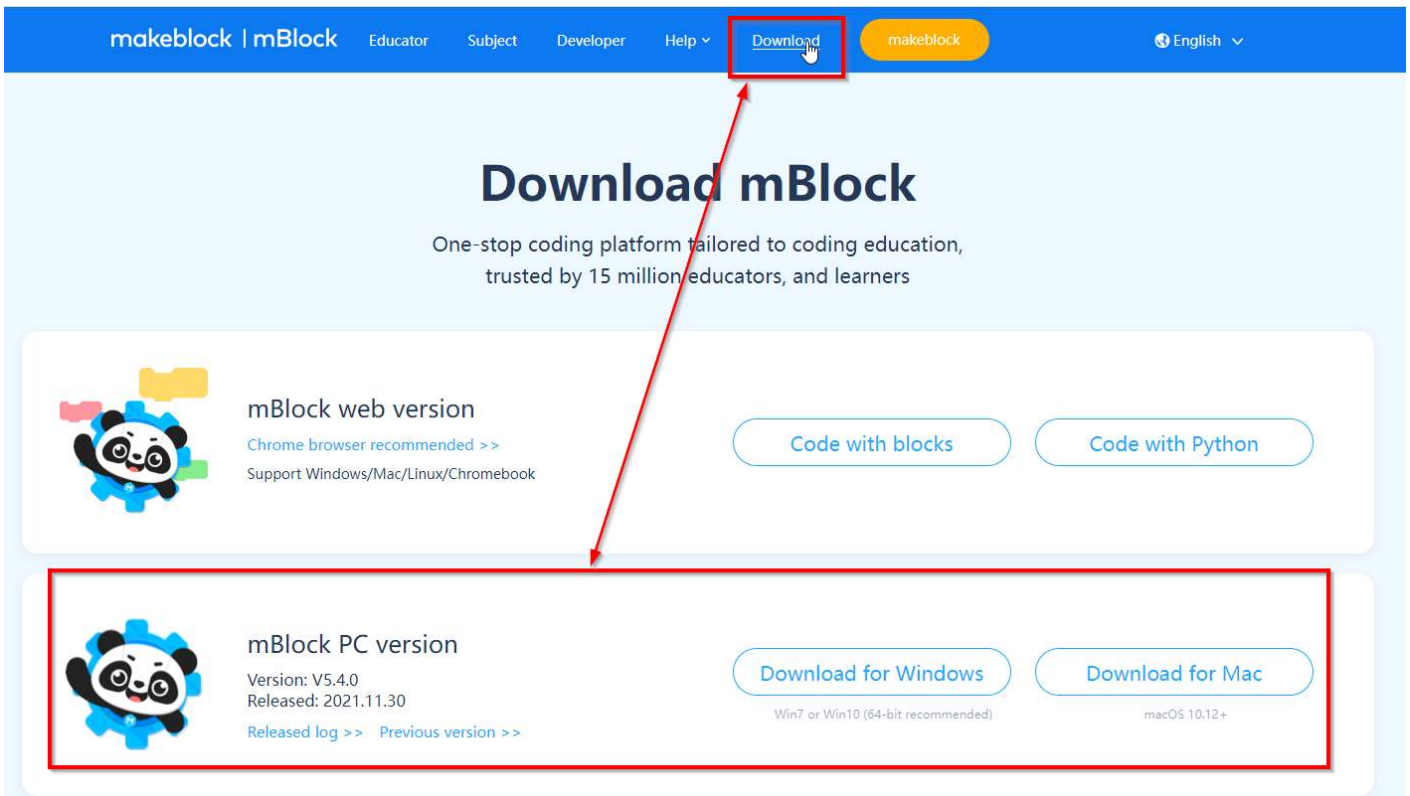
Requisitos de software y Hardware

Instalar el **mBlock** en la [página oficial de descarga mBlock http://www.mblock.cc/](http://www.mblock.cc/).

Disponible en diversos sistemas operativos y su instalación no presenta ningún problema.

- mBlock for PC: aconsejado para el aula, y recomendamos conexión 2.4G: **Windows, Mac, Linux, ChromeOS,**
- mBlock for Mobile, que no aconsejamos para el aula porque necesita Bluetooth y se produce interferencias: Android, **iPad** .

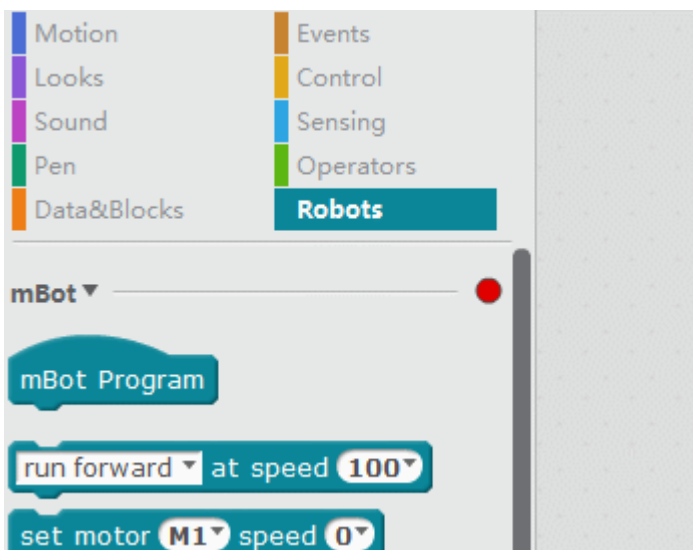
Recomendamos usar la versión descargada y así no usar Internet, tenemos versión Windows y versión Mac, para Linux ver [Robótica con Vitalinux](#)



Hay que destacar que en Windows, la primera vez que se quiere comunicar con el robot **salta el Firewall de Windows** bloqueándolo, pero mostrando un diálogo si se permite o no esta comunicación externa, clicar en **PERMITIR**.

Se actualiza muy a menudo, mejorando cada vez más sus prestaciones, lo que implica que su descarga es cada vez más pesada, más de cien megas, pero después prescinde de Internet.

El programa es muy sencillo si estás acostumbrado al SCRATCH :





Fuente: <http://makeblock.es/>

Qué curioso !

mBlock está basado en Scratch y simplemente ha añadido extensiones para trabajar mBot, pero también ha añadido extensiones para trabajar Arduino, no es capricho, es porque mBot es un Arduino con dos motores y sensores.

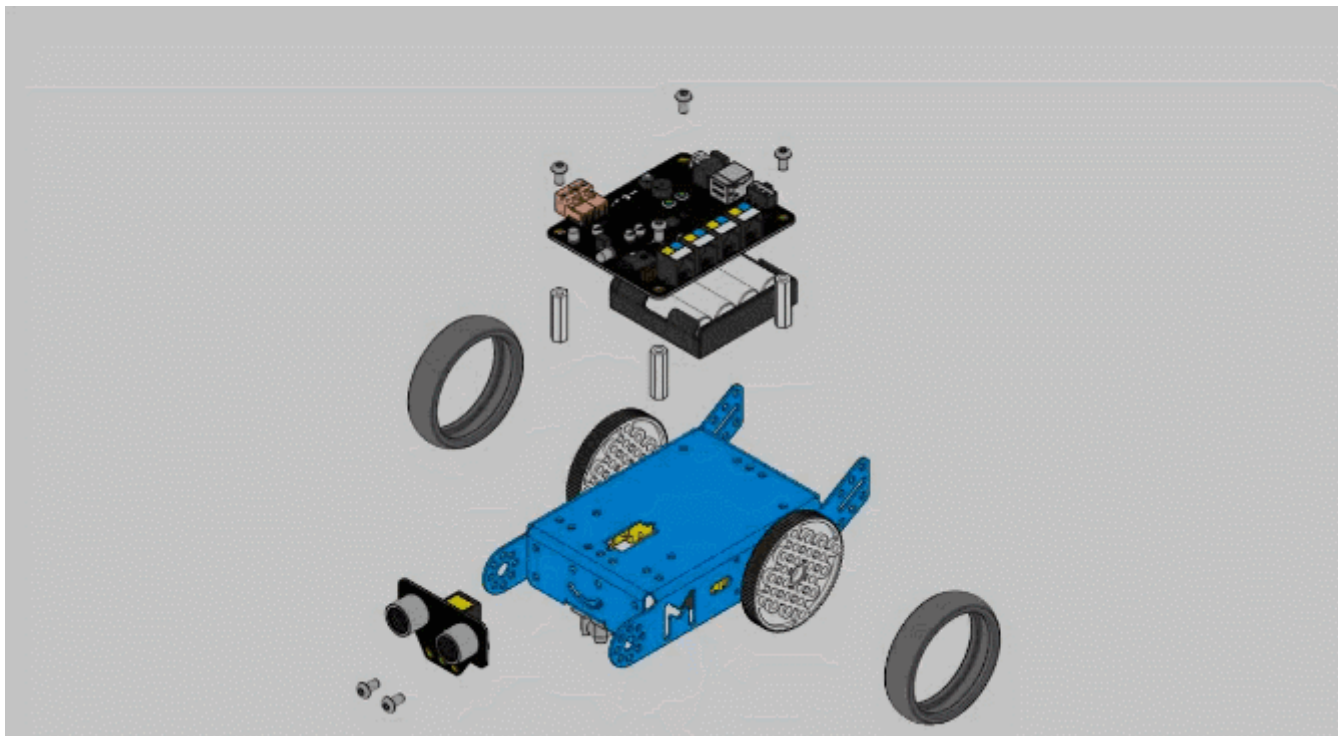
En CATEDU hemos experimentado distintas formas de programar el Arduino de forma sencilla para alumnos de primaria y encontramos mBlock como el mejor programa para hacerlo. Nuestros cursos de -Arduino utilizamos mBlock.

Un vistazo a mBlock en la página de Robotopia :

<https://www.makeblock.es/soporte/mblock/>

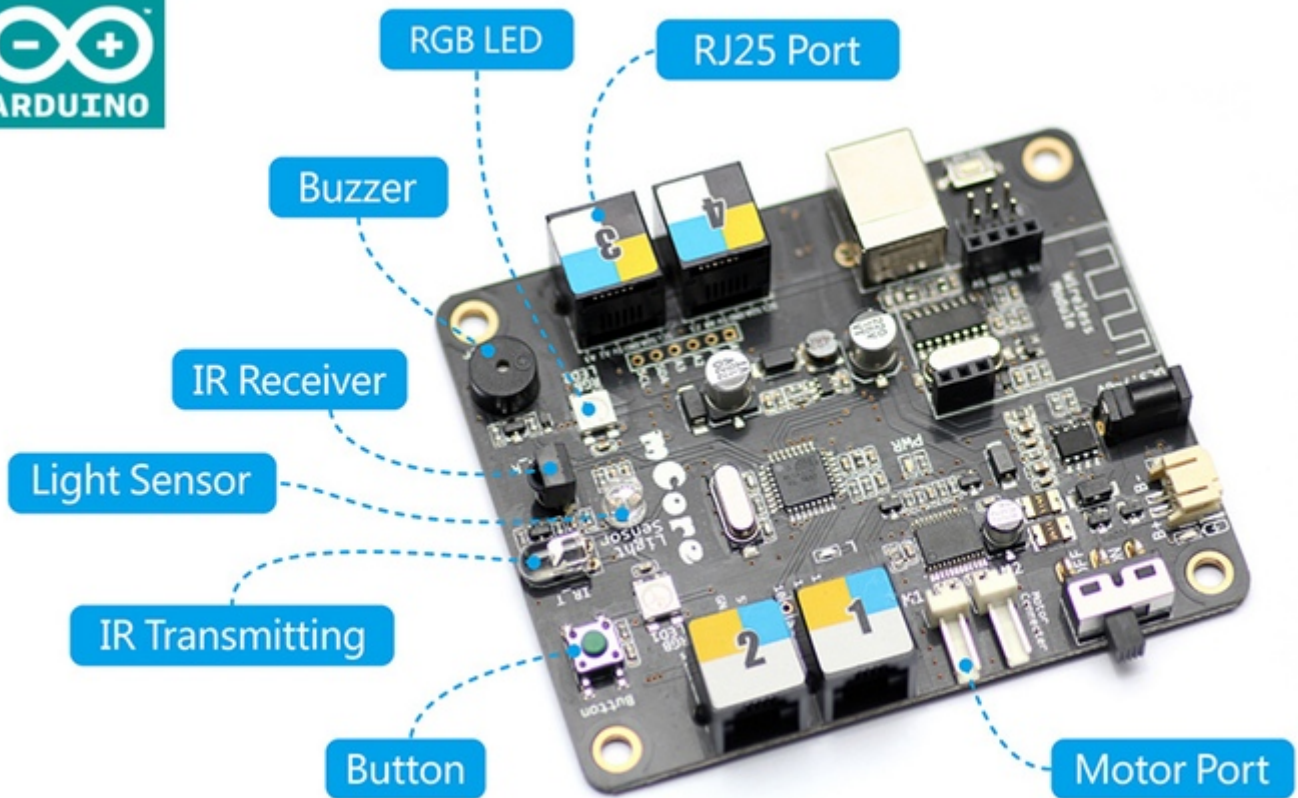
Arduino: El corazón de mBot

El kit es fácil de montar ([instrucciones](#)), pero si nos fijamos el corazón es la placa de Arduino:



Fuente: <http://makeblock.es/>

Nos tenemos que fijar en estos componentes de la placa:



Fuente: <http://makeblock.es/>

- Los R25 donde montaremos los diferentes componentes externos, tienen colores para indicar qué componentes son compatibles.
- **Transmisor de Infra Rojos** para interactuar con el mando a distancia, un elemento de interacción, pero ojo: común a todos los robots que estén en el aula
- **Botón** que permitirá interactuar
- Puertos de motor, donde conectaremos los motores de las ruedas.
- **Buzzer** o pequeño altavoz, no puede hacer maravillas, sólo tonos simples
- RGB Led o luces de los tres colores básicos.
- **Light Sensor** o sensor de luz, que nos servirá como otra vía de interacción.
- También tiene una **conexión USB** para conectarse con el ordenador con un cable.
 - Esta conexión USB a la vez puedes usarse como alimentación. Es una opción a considerar si se nos ha agotado las pilas y queremos seguir programando. O si quieres que se mueva el robot sin cable, y te has quedado sin baterías y no tienes pilas, pon un powerbank en ese puerto y a funcionar.
 - La conexión USB funciona también como cargador si en vez de pilas usamos la batería de litio que hay debajo de la placa



- Para que entiendas qué diferencia hay entre una placa Arduino y esta placa de mBot, esta placa es un Arduino con ...
 - Algunos elementos integrados: sensor luz, botón, transmisor IR, buzzer, leds...
 - Tiene puertos RJ25 para conectar elementos exteriores fácilmente.

Componentes exteriores

Los componentes exteriores no importantes a la hora de programar del Kit son:

- Dos [motores 200rpm 6V](#) conectados a las ruedas.
- [Mochila de comunicación 2.4G](#) que permite conectarse con el ordenador de forma inalámbrica.

Componentes exteriores importantes a la hora de programar y que trataremos en este curso son:

- [Sensor de Línea](#) para utilizarlo por ejemplo como sigue líneas. EN ESTE CURSO LO CONSIDERAREMOS CONECTADO EN EL **PUERTO 2**.
- [Sensor de distancia por ultrasonidos](#) para utilizarlo por ejemplo como evita-obstáculos. EN ESTE CURSO LO CONSIDERAREMOS CONECTADO EN EL **PUERTO 3**.
- [Matriz de leds 8x16](#) para expresar símbolos y caracteres. Este componente no está en el [kit standard mBot](#), pero sí en el [que presta CATEDU](#). EN ESTE CURSO LO CONSIDERAREMOS CONECTADO EN EL **PUERTO 1**.

Puedes elegir montar estos componentes en otros puertos, simplemente tienes que tenerlo en cuenta en la programación.
Si utilizas el kit de Catedu TE PEDIMOS NO MODIFICAR LOS PUERTOS PARA NO TENER PROBLEMAS CON LOS SIGUIENTES COMPAÑEROS QUE UTILICEN EL EQUIPO.





Fuente de las imágenes: <http://makeblock.es/>

Programación mBlock

DOS FORMAS DE PROGRAMAR EN MBLOCK :

OPCIÓN Programación en vivo

El programa reside en el ordenador, y en la placa hay instalado un Firmware para ir escuchando y ejecutando lo que manda el ordenador.

- **VENTAJAS**

- Te permite interactuar el robot y el ordenador, por ejemplo podemos hacer que cuando el detector de humedad detecte agua, que salga por pantalla un fondo acuático, o que pulsando una tecla del teclado se encienda un LED en la placa...

- **DESVENTAJAS**

- **hay que cargar dentro del robot el Firmware exclusivo de mBlock** para que Arduino haga caso a mBlock
- Hay que tener **nuestro ordenador como intermediario, se come los recursos** y puede que nuestro programa vaya lento
- Por supuesto necesita tener ordenador conectado al robot, o sea, trabaja como un esclavo del ordenador.

OPCIÓN Programación cargar a la placa

Todos los programas editores de Arduino (tanto los que programan con código como el Arduino IDE) como los editores de programas gráficos en bloque (mBlock, Snap4Arduino, Arduinoblocks, ...) permiten cargar el programa en la placa. Las ventajas y desventajas son las opuestas de trabajar en vivo.

EN VIVO ¿Qué es eso?

Existe una posibilidad de utilizar la placa "en vivo" frente a "cargar" el programa en la placa. Es decir, interactuando con el ordenador. El programa está en el PC. En la placa hay un **firmware** que le dice que este a las órdenes del PC. De esta manera podemos por ejemplo:

- Enviar órdenes desde el ordenador a la placa.

Por ejemplo que al pulsar la tecla espacio que se encienda el led D13

- Enviar información desde la placa al ordenador

Por ejemplo que muestre por pantalla nos muestre la cantidad de luz, que registra el sensor LDR, etc...



Que nosotros sepamos, estos programas permiten la programación en vivo :

- **mBlock** placas: Arduino, Microbit, Raspberry Pi, ... robots de Makeblock: mBot, Cyberpi...

- **EchidnaScratch** CURSO DE ECHIDNA

VENTAJAS LA PROGRAMACIÓN EN VIVO PERMITE MUCHO JUEGO Y POSIBILIDADES A LA HORA DE ELABORAR PROYECTOS

INCONVENIENTES: Necesitas el ordenador encendido y conectado al robot.

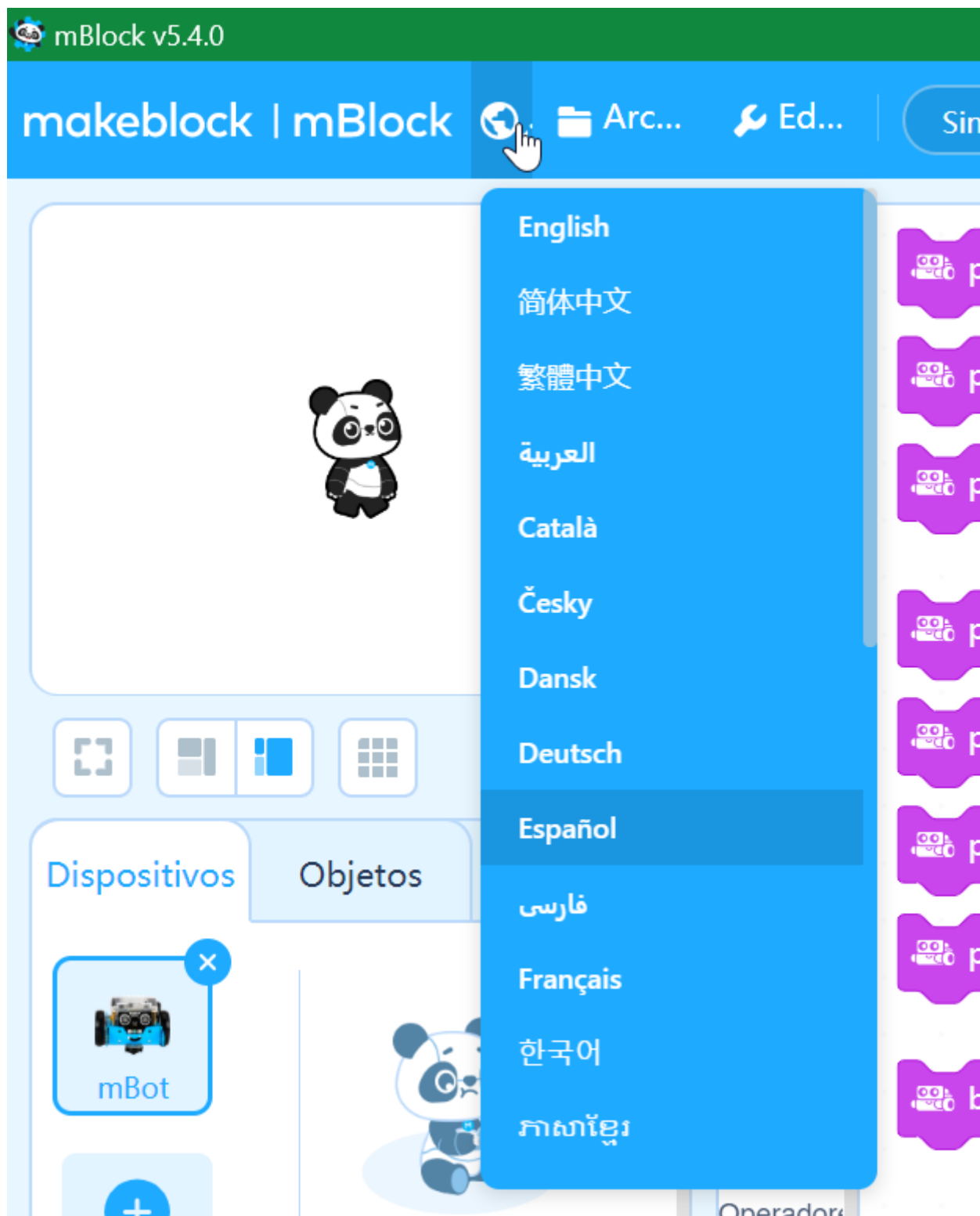
Preparando mBlock

Descargamos y ejecutamos **mBlock** desde la página de descargars de mBlock

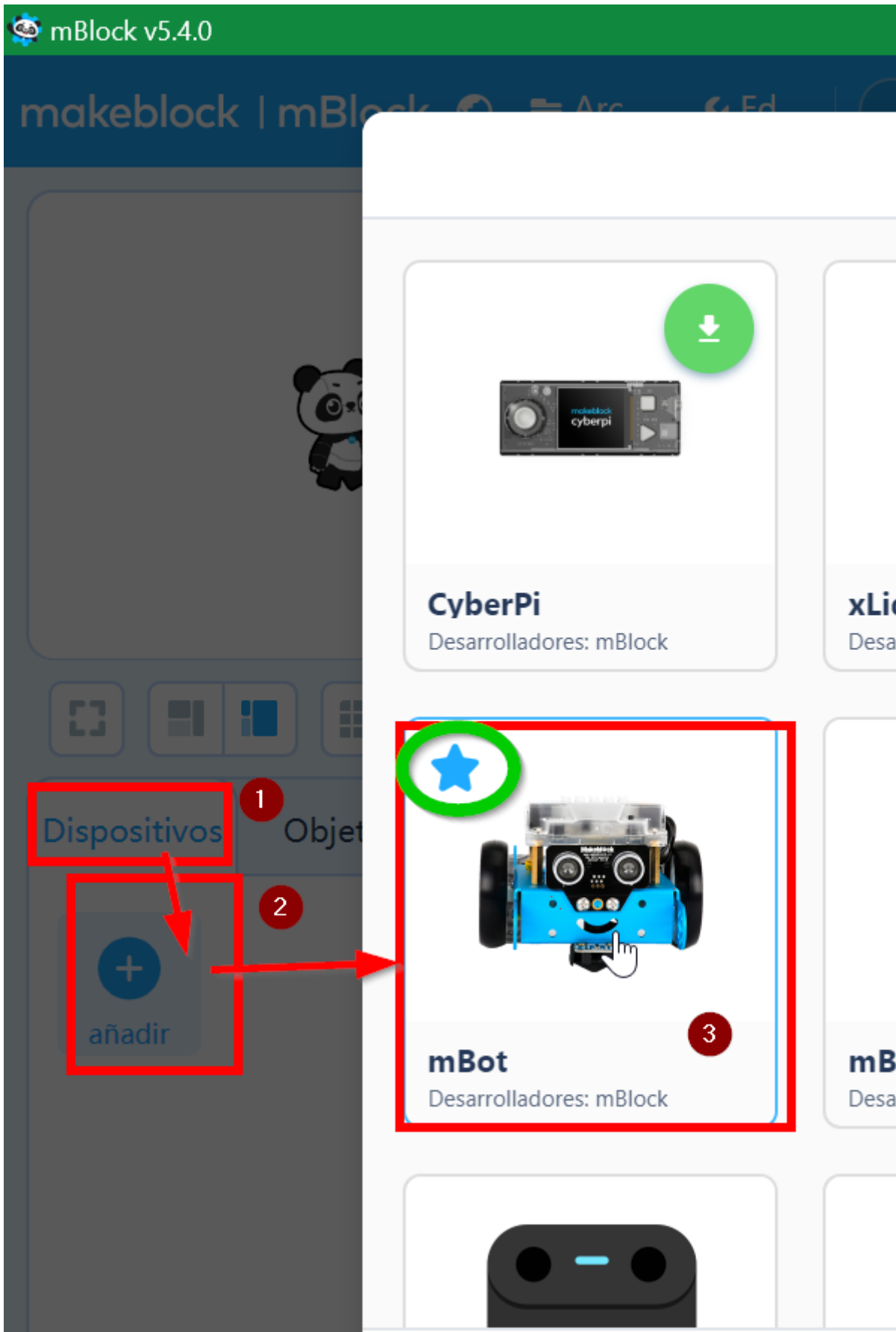
<https://www.mblock.cc/en/download/>

The screenshot shows the mBlock download page. A red box highlights the 'Download' link in the top navigation bar. A red arrow points from this box to the 'mBlock PC version' section. Another red box highlights the 'mBlock PC version' section, which includes the mBlock logo, version information (V5.4.0, Released: 2021.11.30), and download links for Windows and Mac. The 'mBlock web version' section is also visible, featuring a 'Code with blocks' button and a 'Code with Python' button.

Podemos configurar el lenguaje



Vamos a Dispositivos, vemos pues como mBlock esta preparado para controlar distintos tipos de robot, destacamos Microbit, Arduino, bueno, al grano, vamos a añadir mBot





- Podemos **quitar** los dispositivos que no sean mBot pues no vamos a trabajar con ellos
- Si lo marcamos mBot pulsando en la estrella, ya saldrá predeterminado.

IMPORTANTE: ACTUALIZAR FIRMWARE

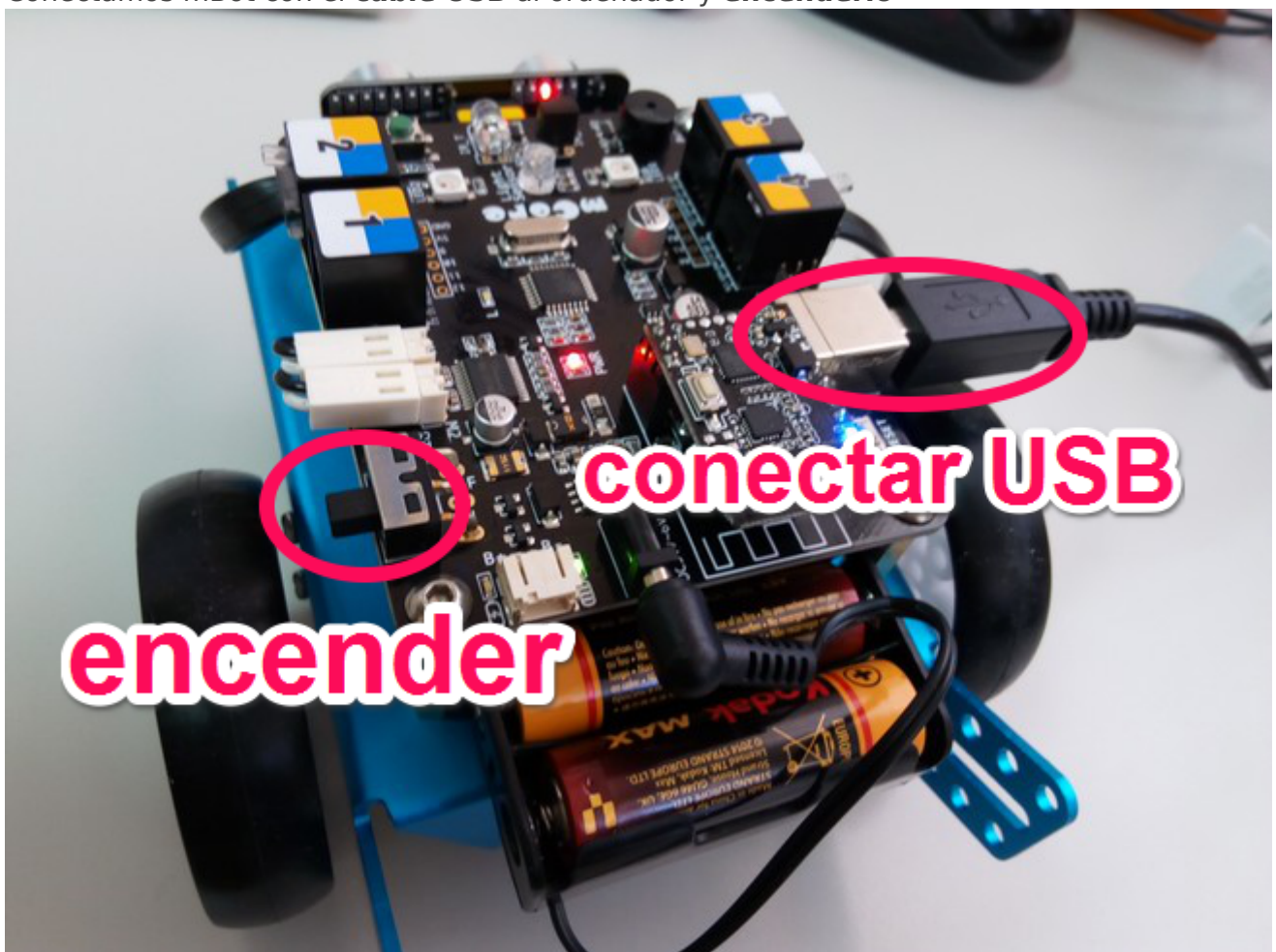
P.-Enciendes tu mBot, y de repente ¡¡se pone a hacer cosas !!! pero si no le programado aún nada!!! ¿Cómo es posible?

R.-Seguramente tu equipo mBot, el anterior alumno **ha dejado un programa** dentro del mBot

ACTUALIZAREMOS EL FIRMWARE SIEMPRE QUE MBOT NO RESPONDA A MBLOCK

¿Cómo se hace?

1. Conectamos mBot con el **cable USB** al ordenador y **encenderlo**



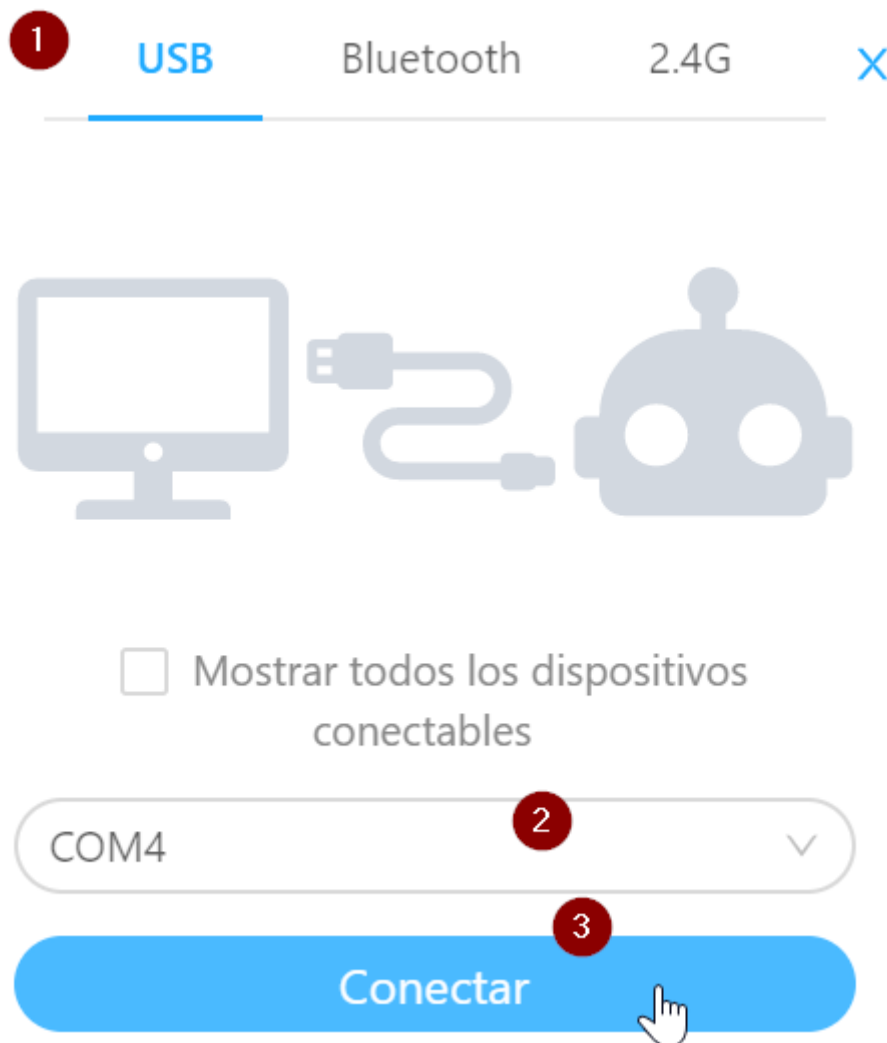
Fuente el autor

2. Pulsamos en mBlock "CONECTAR"





3. En el diálogo seleccionamos **USB** y de todos los puertos USB seleccionamos el que esta conectado (suele ser el que propone) y **CONECTAR**



4. Una vez que la conexión ha sido un éxito, le damos a **ACTUALIZAR**





5. Sale un diálogo para **elegir qué firmware** quieres, si el de fábrica o el que existe en Internet, no hay mucha diferencia de versiones

Actualizaciones de firmware del dispositivo

Dispositivo:

mBot

Versión del firmware:

Firmware en internet(06.01.107)



Cancelar

Actualizaciones



6. Sale un mensaje de que **se ha actualizado** y advierte que SE VA DESCONECTAR EL DISPOSITIVO

La actualización ha terminado.



¡La actualización del firmware ha terminado!

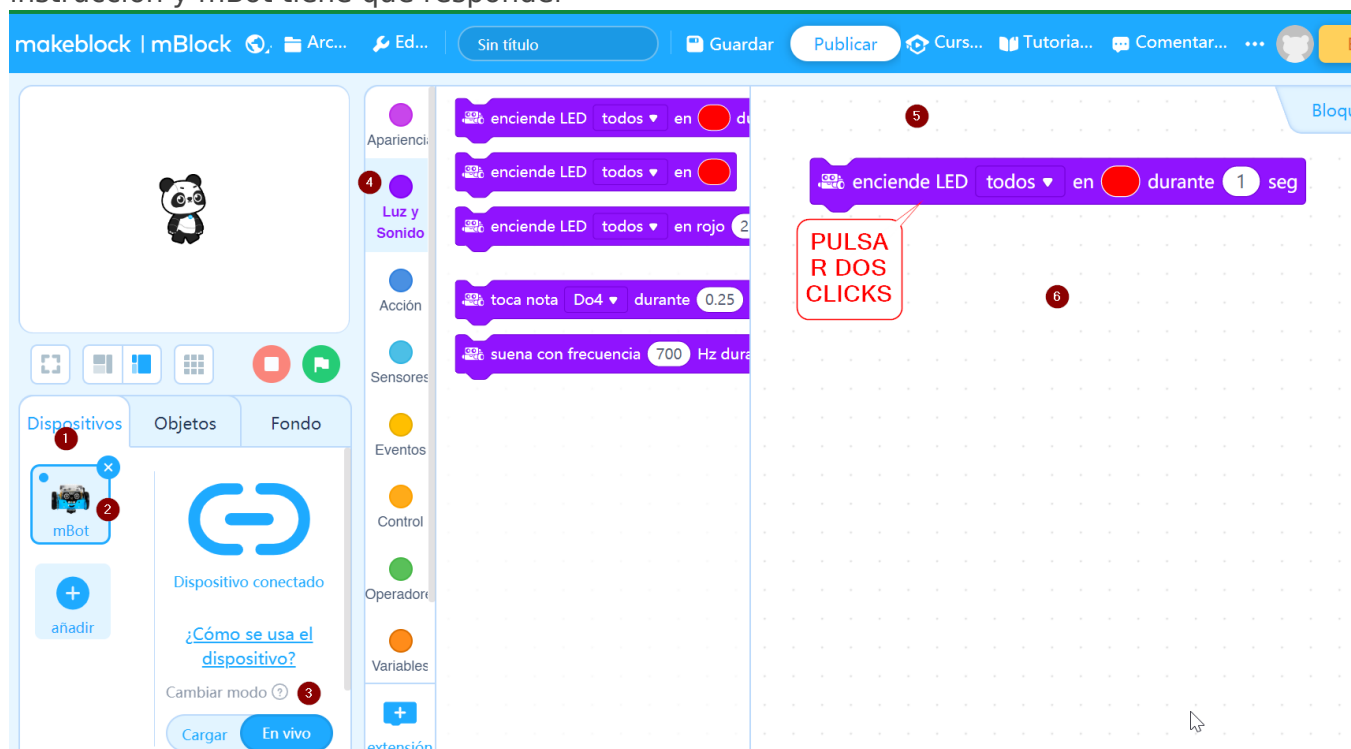
- Para aprovechar la actualización, puedes reiniciar el dispositivo y conectarlo de nuevo.

Aceptar

7. Volvemos a **conectar el dispositivo**, es decir, repetimos los pasos PASO 2 Y PASO 3



8. **PROBARLO** vamos al dispositivo mBot, nos aseguramos que esta en modo EN VIVO, elegimos una instrucción sencilla, por ejemplo vamos a LUZ Y SONIDO elegimos ENCIENDE LED TODOS EN ROJO DURANTE 1 SEGUNDO damos DOS CLICKS en esa instrucción y mBot tiene que responder



EN VIVO ¿Eso qué es?

DOS FORMAS DE PROGRAMAR EN MBLOCK :

OPCIÓN Programación en vivo

El programa reside en el ordenador, y en la placa hay instalado un Firmware para ir escuchando y ejecutando lo que manda el ordenador.

- **VENTAJAS**

- Te permite interactuar el robot y el ordenador, por ejemplo podemos hacer que cuando el detector de humedad detecte agua, que salga por pantalla un fondo acuático, o que pulsando una tecla del teclado se encienda un LED en la placa...

- **DESVENTAJAS**

- **hay que cargar dentro del robot el Firmware exclusivo de mBlock** para que Arduino haga caso a mBlock
- Hay que tener **nuestro ordenador como intermediario, se come los recursos** y puede que nuestro programa vaya lento
- Por supuesto necesita tener ordenador conectado al robot, o sea, trabaja como un esclavo del ordenador.

OPCIÓN Programación cargar a la placa

Todos los programas editores de Arduino (tanto los que programan con código como el Arduino IDE) como los editores de programas gráficos en bloque (mBlock, Snap4Arduino, Arduinoblocks, ...) permiten cargar el programa en la placa. Las ventajas y desventajas son las opuestas de trabajar en vivo.

EN VIVO ¿Qué es eso?

Existe una posibilidad de utilizar la placa "en vivo" frente a "cargar" el programa en la placa. Es decir, interactuando con el ordenador. El programa está en el PC. En la placa hay un **firmware** que le dice que este a las órdenes del PC. De esta manera podemos por ejemplo:

- Enviar órdenes desde el ordenador a la placa.

Por ejemplo que al pulsar la tecla espacio que se encienda el led D13

- Enviar información desde la placa al ordenador

Por ejemplo que muestre por pantalla nos muestre la cantidad de luz, que registra el sensor LDR, etc...



Que nosotros sepamos, estos programas permiten la programación en vivo :

- **mBlock** placas: Arduino, Microbit, Raspberry Pi, ... robots de Makeblock: mBot, Cyberpi...

- **EchidnaScratch** CURSO DE ECHIDNA

VENTAJAS LA PROGRAMACIÓN EN VIVO PERMITE MUCHO JUEGO Y POSIBILIDADES A LA HORA DE ELABORAR PROYECTOS

INCONVENIENTES: Necesitas el ordenador encendido y conectado al robot.

¿Qué hace mBot EN VIVO dependiendo del ordenador?

En el modo EN VIVO, mBot obedece a un programa interno (**Firmware**) que es el que hemos puesto al ACTUALIZAR EL FIRMWARE

Este **Firmware** lo que le ordena a la placa Arduino es:

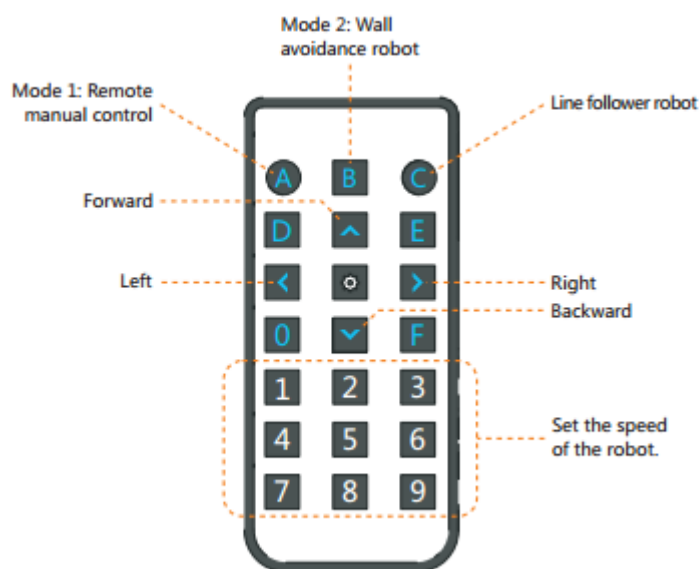
1. Haz caso a las instrucciones del mando de IR ver figura de abajo
2. Haz caso a la conexión con el ordenador y a las instrucciones que vengan por ahí

Esto tiene su importancia:

- El punto 1 te dice que no puedes hacer un programa con el Mando IR pues prevalece el del Firmware, por lo tanto no se puede utilizar en tu programación el mando IR en el modo "dependiendo del ordenador" pero sí en el modo "independiente del ordenador".
- El punto 2 te dice que tus programas se ejecutan en el ordenador y se lo comunica al robot luego:
- Si desconectamos el ordenador o la conexión, dejan de funcionar, o mejor dicho se mBot se queda atascado en la última instrucción ejecutada.
- Tus programas van un poco lentos, pues dependen del ordenador. El mismo programa lo pasas al modo "independiente del ordenador" y va mucho más rápido.



Instrucciones del mando IR en el firmware que viene por defecto (modo dependiendo del ordenador):



Mode 1: Remote manual control

Users can use buttons to control the direction and speed of mBot.

Mode 2: Wall avoidance robot

A robot that can avoid walls and obstacles while moving.

Mode 3: Line follower robot

Line follower is a robot that can follow a path. The path can be visible like a black line on a white surface (or vice-versa)

Fuente de las imágenes: <http://makeblock.es/>

AL ACTUALIZAR EL FIRMWARE ESTAS DICIENDO AL MBOT QUE TRABAJE EN VIVO

EN VIVO conexión 2.4 Bluetooth o cable

Tenemos dos opciones de comunicarnos con mBot, inalámbrica 2.4G o con cable, aconsejamos la primera:

Conexión inalámbrica 2.4G

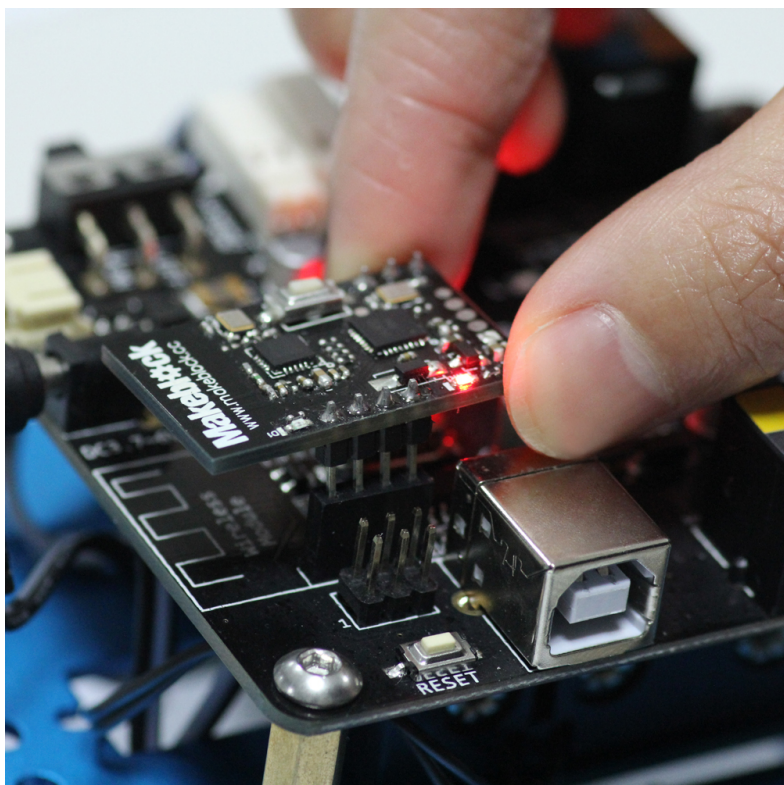
La conexión inalámbrica se realiza a través de la [mochila 2.4G](#). Insertamos el pincho USB de la mochila (no necesita ningún driver, el ordenador lo interpreta como un ratón inalámbrico).



Pincho USB 2.4G fuente Makeblock.es

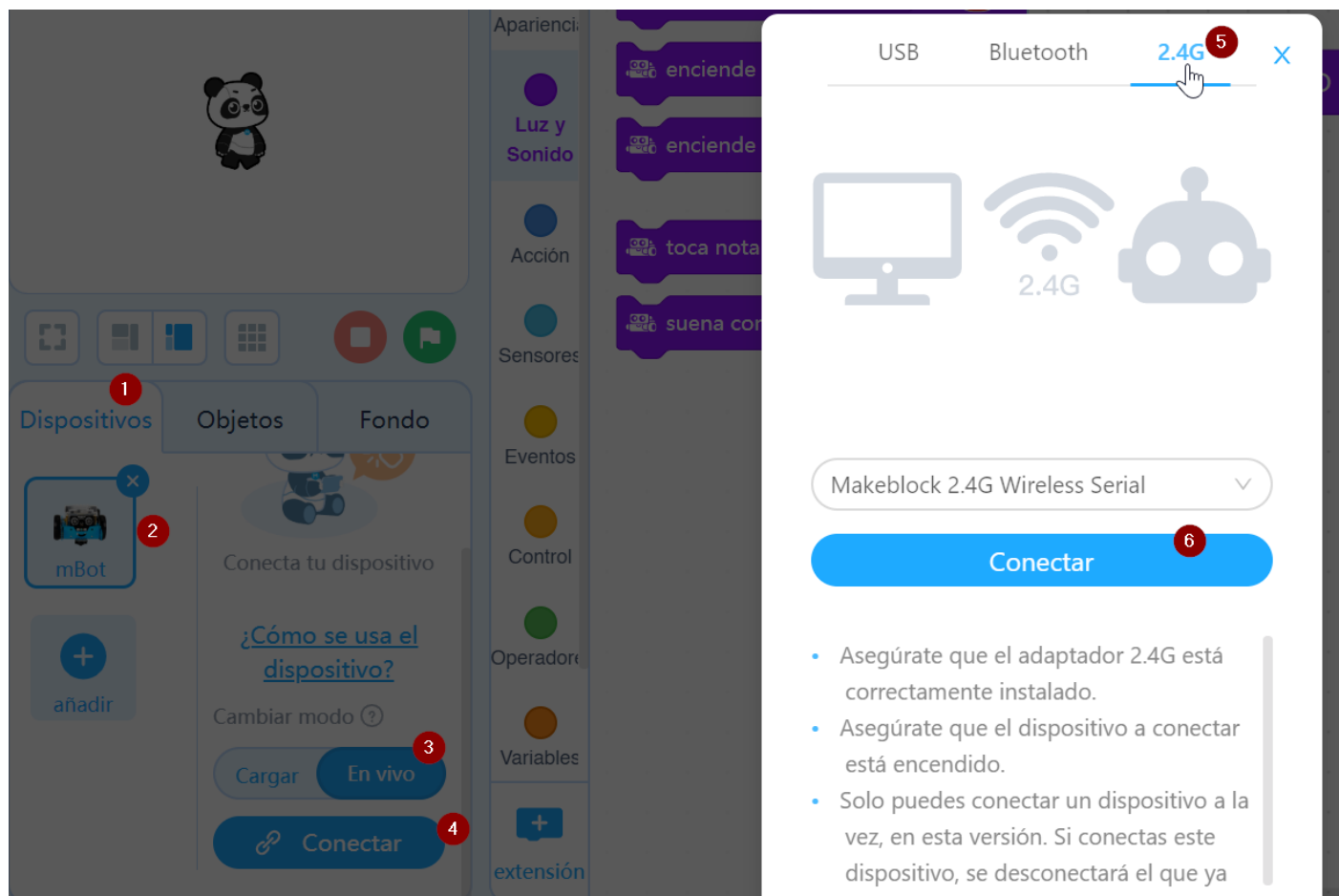
Cada pincho va asociado a un robot (o mejor dicho: a la [mochila 2.4G](#) están emparejados, si el led de la mochila parpadea es que ha perdido conexión, podemos emparejarlo haciendo una pulsación larga en el botón que hay en la mochila 2.4G.





Mochila 2.4G fuente makeblock.es

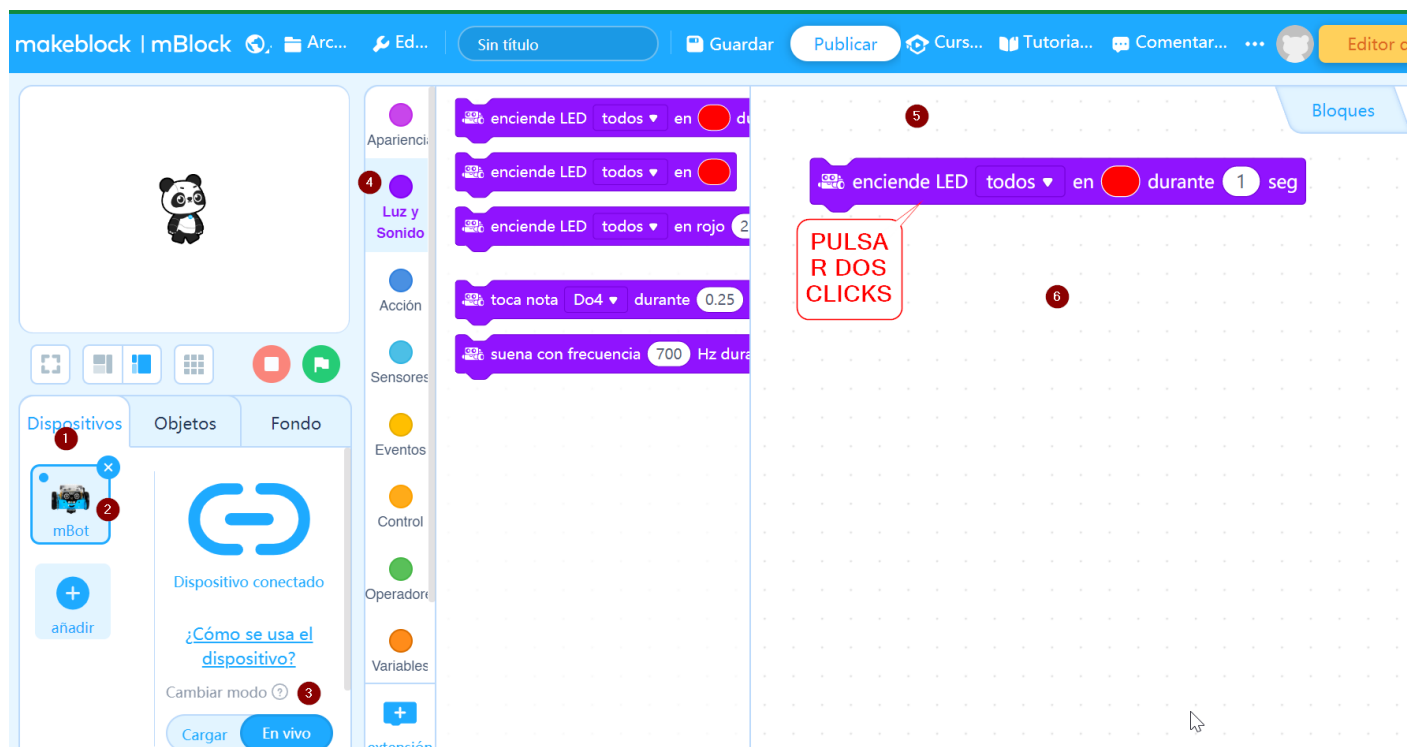
y conectamos el robot con el programa en **Conectar- 2.4G Serial- Conectar**



Si sale que esta conectado 2.4G ya podemos trabajar SOLO EN VIVO

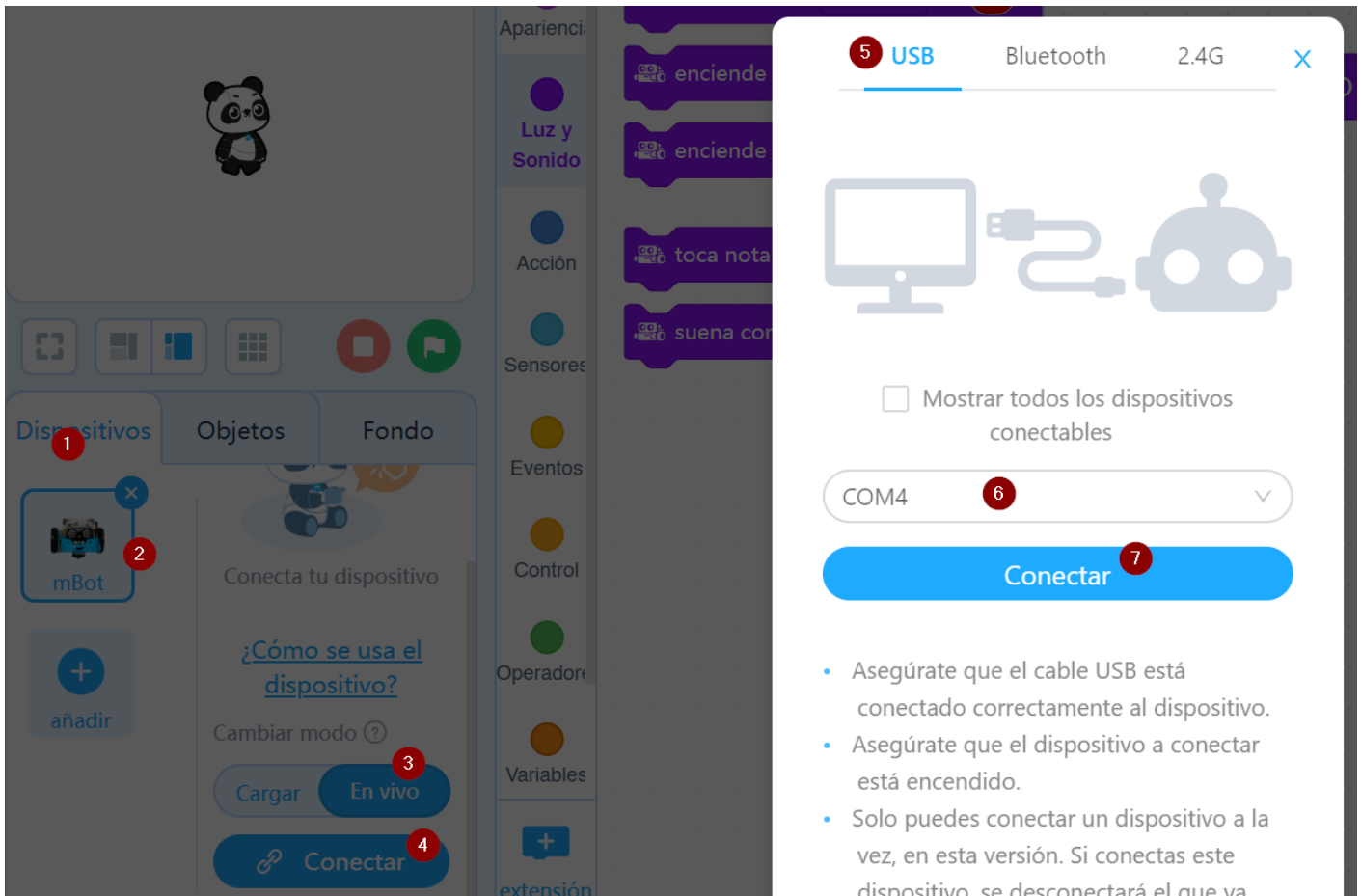


Podemos probarlo haciendo dos clicks a la instrucción de encender led rojo durante un segundo



Conexión por cable

Existe la posibilidad de conexión por cable, útil por ejemplo si se ha perdido el pincho, seguir programando hasta la espera de la compra de otro, o por ejemplo si se agotan las pilas, por cable el robot puede seguir funcionando. Evidentemente con la pérdida de libertad de movilidad.



No hay que olvidar que si optamos por la conexión Serial por el puerto USB, TENEMOS QUE DESCONECTAR LA OTRA CONEXIÓN INALÁMBRICA:

CARGAR

La opción de cargar tu programa en el mBot se hace simplemente

1. Conectar el mBot por CABLE como lo hemos visto anteriormente
2. Realizas tu programa
3. Pulsa CARGAR
4. Pulsa SUBIR y el programa va al mBot



ATENCIÓN, UNA VEZ CARGADO EL PROGRAMA, **TE HAS "CARGADO" EL FIRMWARE** por lo tanto si quieres volver a trabajar con el mBot **EN VIVO** tienes que volver a ACTUALIZAR EL FIRMWARE tal y como lo hemos visto

EN VIVO vs CARGAR

VENTAJAS EN VIVO

- Se trabaja y se programa **MUY FÁCILMENTE** simplemente se ponen instrucciones en la pantalla de bloques y se van ejecutando en vivo
- una ventaja muy considerable **PUEDES INTERACTUAR CON EL ORDENADOR Y CON LOS OBJETOS esto da mucha libertad**
 - puedes con el teclado, ratón dar órdenes al robot
 - puedes visualizar en la pantalla del ordenador lo que lee mBot (por ejemplo la distancia del sensor US)
 - puedes que los objetos (el panda o lo que sea...) interactúe
 - con la opción CARGAR puedes interactuar con los objetos si utilizas la extensión

BROADCAST ver <https://libros.catedu.es/books/robotica-educativa-con-mbot/page/programacion-mblock>

DESVENTAJAS EN VIVO

- Dependes del **ordenador**, es decir, el ordenador tiene que estar encendido
- **lentitud**, las órdenes van y vienen, no es lo mismo que si se ejecuta desde el mismo mBot, programas que implican rapidez, como el sigue líneas no funcionan bien en vivo, al menos que bajes la velocidad de los motores.

Cargar un programa tiene las ventajas y desventajas al revés de en vivo

OJO una vez que le das a CARGAR si quieres trabajar EN VIVO tienes que ACTUALIZAR EL HARDWARE



A JUGARRRRRR !!!!

¿Lo probamos? venga!!! monta este sencillo programa. Las flechas te indican dónde está cada instrucción :



Lo tienes aquí <https://planet.mblock.cc/project/2999030>

y como decía Joaquín Prat en el **precio justo A jugaaarr !!**



Fuente <http://lossesentas-mispersonajes.blogspot.com.es/2015/08/joaquin-prat.html>

(si entiendes esto último, es que tienes la edad óptima para aprender a programar ;)



CATEDU

PONLE MÁS DIVERSIÓN

Pega en el mBot con cinta aislante una varilla sujeta globos, pon en un extremo un globo y en el otro extremo una aguja de coser también con cinta aislante



Se lo pasan genial !!!

<https://www.youtube.com/embed/54exNJ369n4>