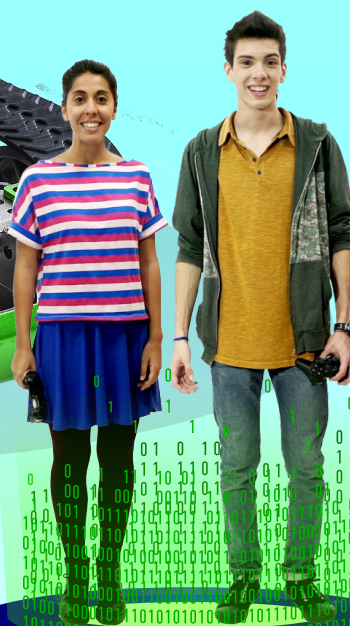
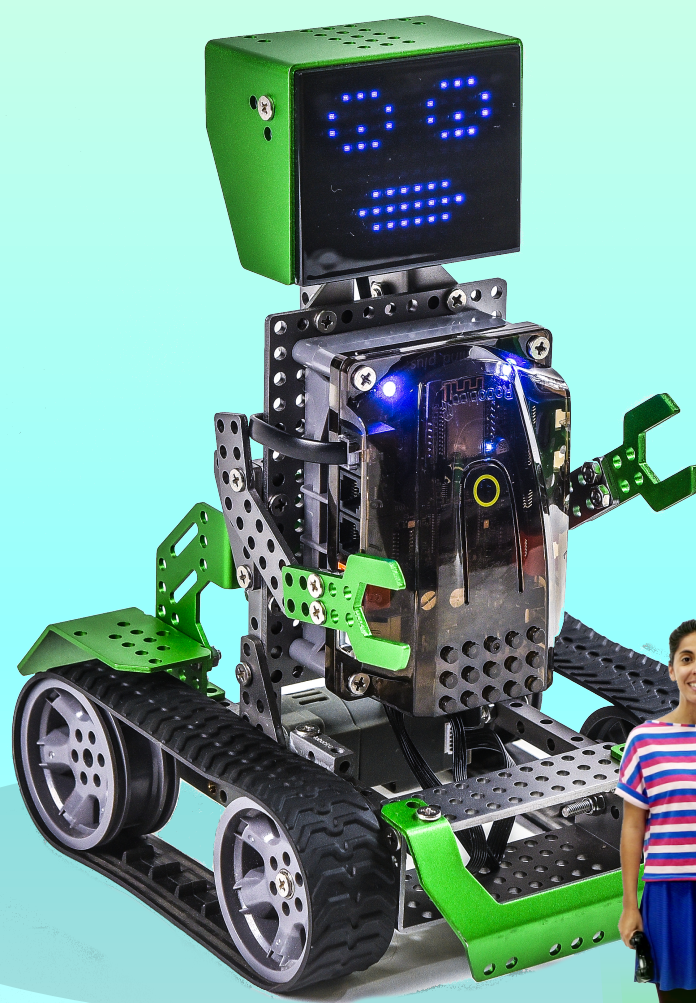


EMOTIBOT

Robótica y programación para nivel secundario



**APRENDER
CONECTADOS**



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

Autoridades

Presidente de la Nación

Mauricio Macri

Jefe de Gabinete de Ministros

Marcos Peña

Ministro de Educación, Cultura, Ciencia y Tecnología

Alejandro Finocchiaro

Secretario de Gobierno de Cultura

Pablo Avelluto

Secretario de Gobierno de Ciencia, Tecnología e Innovación Productiva

Lino Barañao

Titular de la Unidad de Coordinación General del Ministerio de Educación, Cultura, Ciencia y Tecnología

Manuel Vidal

Secretaria de Innovación y Calidad Educativa

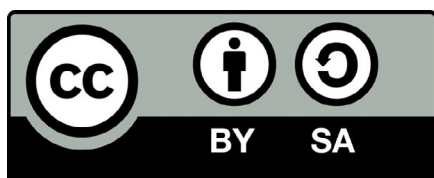
Mercedes Miguel

Subsecretario de Coordinación Administrativa

Javier Mezzamico

Directora Nacional de Innovación Educativa

María Florencia Ripani



Prólogo

Nos encontramos en un mundo que se rediseña permanentemente. Un mundo lleno de desafíos, de cambios constantes y nuevas tecnologías que convergen con nuestro día a día. La incorporación de recursos pedagógicos innovadores puede generar la construcción de nuevos conocimientos, útiles para desarrollar el presente y avanzar hacia el futuro.

Nuestra obligación es crear nuevas realidades para la sociedad a partir de los talentos, habilidades y capacidades de nuestras niñas y niños. Debemos emplear la tecnología para despertar el pensamiento creativo y brindar las herramientas que motiven a nuestros estudiantes a emprender el desafío de vivir y materializar sus sueños.

Tenemos una sociedad definida e intervenida cada vez más por la tecnología, donde las niñas, niños y jóvenes son protagonistas de los cambios. Es por eso que debemos formarlos para que sean artífices de sus propias vidas y formadores de una mejor comunidad.

El Plan **Aprender Conectados** posee el espíritu y los materiales tecnológicos y pedagógicos necesarios para llevar a la educación “un paso más allá”, otorgando a los estudiantes oportunidades para adquirir nuevos conocimientos, como la programación y la robótica, para poder concretar sus objetivos y proyectos.

Queremos un sistema educativo con docentes que enseñen a aprender con creatividad, donde las nuevas tecnologías formen parte de la educación y nos asistan en el camino hacia la sociedad que deseamos integrar. Nuestro mayor desafío es despertar lo mejor de cada uno de los chicos y chicas y lo más necesario para su futuro, una verdadera educación de vanguardia.



Alejandro Finocchiaro

Ministro de Educación, Cultura, Ciencia y
Tecnología de la Nación

Presentación

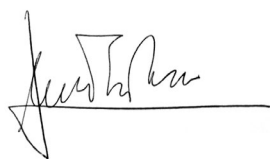
El Plan **Aprender Conectados** es la primera iniciativa en la historia de la política educativa nacional que se propone implementar un programa integral de educación digital, programación y robótica en todos los niveles de la educación obligatoria.

Aprender Conectados es una política de innovación pedagógica y tecnológica tanto para el desarrollo de las competencias de educación digital, como de las capacidades y saberes fundamentales. La propuesta busca dar respuestas a un contexto de cambio permanente, en el cual las habilidades relacionadas con las tecnologías digitales se han convertido en unas de las más valoradas para la construcción del conocimiento, la integración social y el desarrollo del país.

La alfabetización digital requiere un abordaje integral con una visión articulada de todo el sistema educativo, que garantice igualdad y equidad para las niñas, niños y jóvenes de todos los niveles educativos, especialmente cuando el uso de la tecnología digital se inicia en forma cada vez más temprana.

En este marco, en 2018, se aprobaron los Núcleos de Aprendizajes Prioritarios (NAP) de Educación Digital, Programación y Robótica, para la educación obligatoria. Argentina se convirtió, de este modo, en el primer país en la región en integrar estos saberes a la currícula y extender el inicio formal de la alfabetización digital a la sala de 4 años.

El inicio de la construcción de nuevos saberes en todos los niveles de la educación obligatoria será una tarea fundamental para la integración plena en la cultura digital, que demandará el esfuerzo conjunto de toda la comunidad educativa, recuperando prácticas previas y generando propuestas innovadoras que permitan construir futuro y disfrutar de la aventura de aprender.



María Florencia Ripani

Directora Nacional de Innovación Educativa



Mercedes Miguel

Secretaria de Innovación y Calidad Educativa

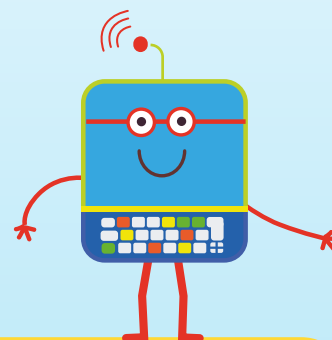


Visión: el futuro es hoy

El país está lleno de héroes. No tienen capas, ni vuelan, sino un gran deseo de aprender. Pero, mientras el mundo está cambiando, seguimos enseñándoles sobre una realidad que poco a poco está quedando atrás. Necesitamos crear las oportunidades para que puedan entender el futuro que hoy ya es realidad: el de códigos, algoritmos, robots y océanos de información que hoy habitan el gran cosmos de redes digitales. Necesitamos acercarles los recursos para que ellos puedan construir juntos y ser nuestros superhéroes del futuro. Porque el sueño de cada niña, niño y joven es la historia de nuestro futuro, de nuestro destino y de nuestra Argentina.



APRENDER CONECTADOS

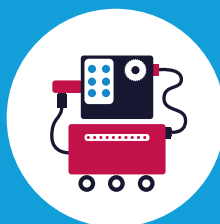


V V V V

**Preparar a los estudiantes para
un uso crítico un uso seguro,
crítico y creativo de la
tecnología digital.**



**Tecnología
emergente**



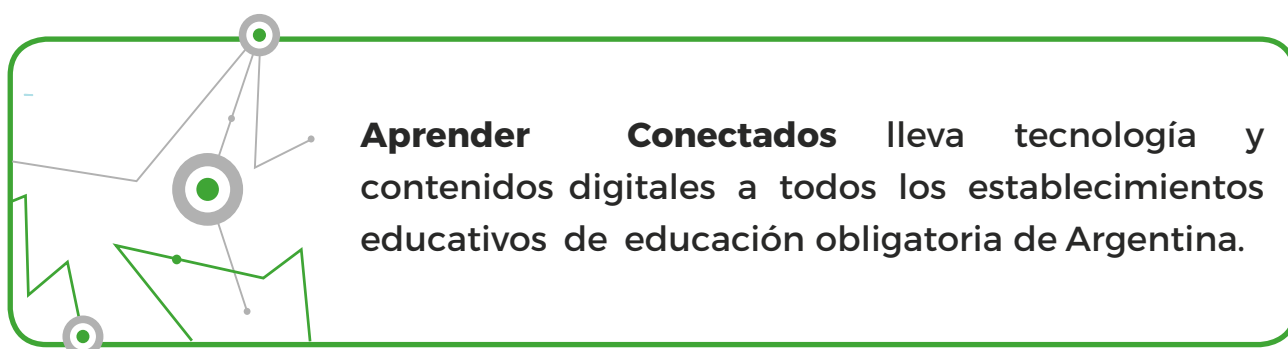
**Innovación
pedagógica**



Aprender Conectados

El Plan **Aprender Conectados** del Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación (MECCyT), es una política integral de innovación pedagógica y tecnológica que comprende como núcleos centrales, el desarrollo de contenidos, el equipamiento tecnológico, la conectividad y la formación docente, que ayude tanto al desarrollo de las competencias de educación digital, como de las capacidades y saberes fundamentales.

Aprender Conectados desarrolla contenidos de alfabetización digital que sistematizan e integran las diferentes competencias y saberes relacionados con el manejo de los nuevos lenguajes producidos por las tecnologías de la información y la comunicación, que demanda la Ley de Educación Nacional N° 26.206. Para esto, prevé la entrega de tecnología educativa emergente en toda la educación obligatoria, con un plan pedagógico orientado a proveer a docentes y estudiantes los recursos que necesitan para un aprendizaje significativo y de calidad.



Este plan llega a todos los establecimientos educativos de gestión estatal del país con un aula digital móvil, que incluye una gran cantidad de equipamiento específico para iniciar a los estudiantes en la educación digital, la programación y la robótica, y para promover la calidad en los aprendizajes.

Estos recursos se acompañan con contenidos innovadores, para que las escuelas- en tanto responsables de los procesos de enseñanza y aprendizaje- se posicionen como nodos de alfabetización digital y guíen a los estudiantes para que puedan realizar un uso seguro, crítico y creativo de la tecnología. De este modo, las escuelas podrán cumplir un rol fundamental en la formación de las niñas, niños y jóvenes de nuestro país.

1. El Plan Aprender Conectados fue creado mediante el Decreto Presidencial N° 386/18 y reglamentado mediante la Resolución del Ministerio de Educación N°1410/18.

Nuevas oportunidades de aprendizaje

“

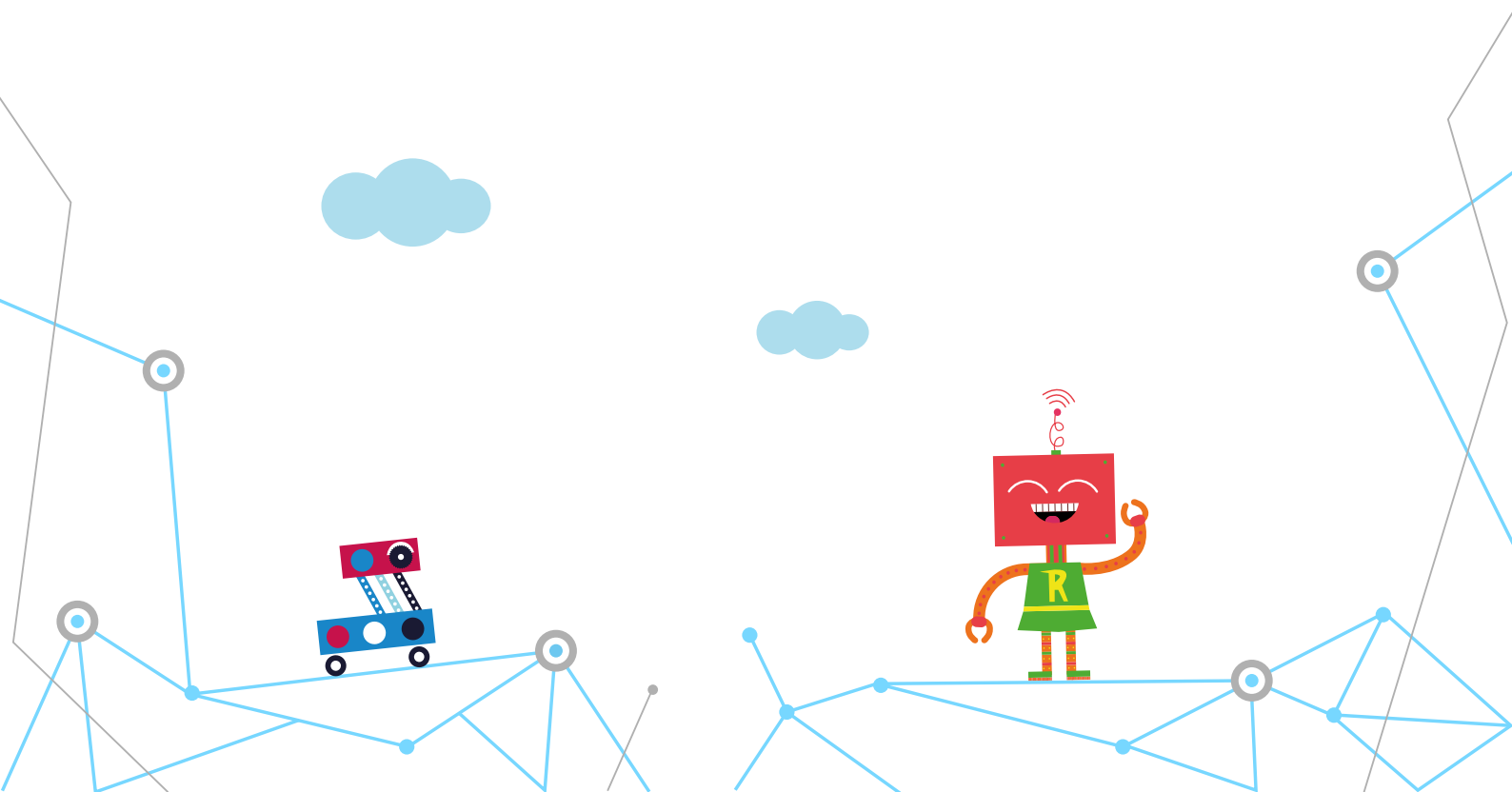
La integración de **Aprender Conectados** aporta tecnología y contenidos digitales que generan nuevas oportunidades para reconocer y construir la realidad: abre una ventana al mundo, facilita la comunicación y la iniciación a la producción digital. Pero, además, promueve la valoración crítica de las tecnologías de la información y la comunicación desde edades tempranas. Esto ofrece un medio para la resolución de problemas –relacionados con la vida cotidiana– fomentando la creatividad, el trabajo colaborativo, una actitud activa y crítica.

El contacto con los recursos digitales se produce a edades cada vez más tempranas. Sin embargo, la mera interacción con la tecnología no garantiza el desarrollo de competencias o capacidades para su uso crítico, siendo esta una tarea que se requiere abordar desde el sistema educativo. Es fundamental, entonces, fomentar la alfabetización digital desde el nivel inicial, pasando por la educación primaria y hasta el fin de la secundaria. Esto es particularmente importante en poblaciones vulnerables, porque permite promover igualdad de posibilidades de aprendizaje y generar mayores oportunidades para su futuro.

”

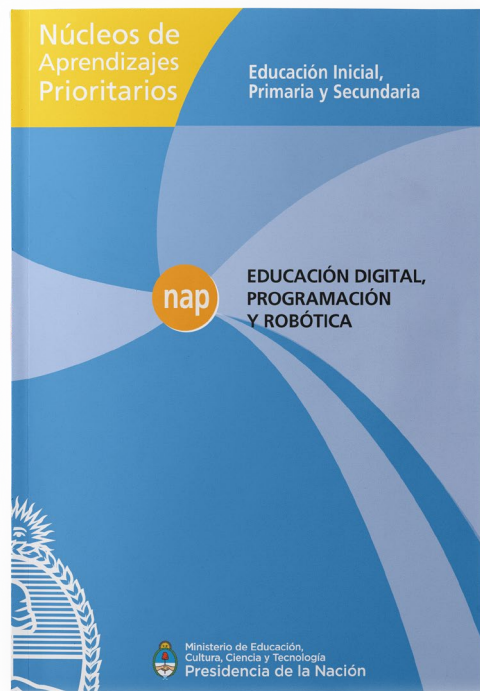
María Florencia Ripani

Directora Nacional de Innovación Educativa



Integración curricular

El Plan **Aprender Conectados** contempla el desarrollo de contenidos de alfabetización digital para su adopción en los diseños curriculares jurisdiccionales. En este marco, en 2018, Argentina se convirtió en el primer país de América Latina en integrar la programación y la robótica en toda la educación obligatoria. Este logro se concretó mediante la aprobación de los Núcleos de Aprendizajes Prioritarios (NAP) de Educación Digital, Programación y Robótica.

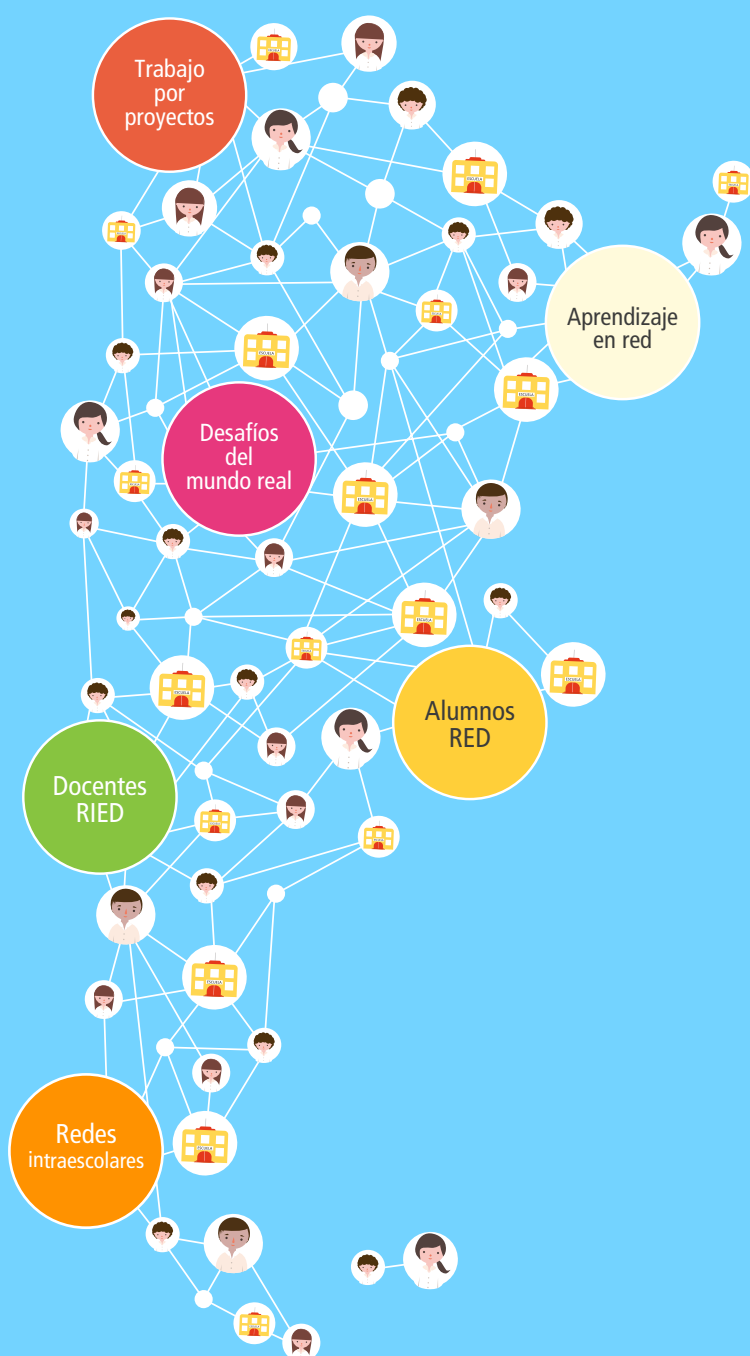


Los NAP abarcan un campo de saberes interconectados y articulados, orientados a promover la alfabetización digital, entendida como el desarrollo del conjunto de competencias y capacidades necesarias para que los estudiantes puedan integrarse plenamente en la cultura digital, incluyendo su participación activa en el entramado de los medios digitales.

2. Los NAP de Educación Digital, Programación y Robótica fueron aprobados mediante la Resolución N° 343/18 del Consejo Federal de Educación. Esta resolución establece un plazo de 2 años para que todas las provincias y la Ciudad Autónoma de Buenos Aires integren los NAP a sus documentos curriculares, desarrollen un plan de formación docente continuo, los integren en la currícula de la formación docente inicial y realicen acciones dirigidas a la comunidad educativa con miras a promover el aprendizaje de la educación digital, la programación y la robótica.

Comunidades de aprendizaje

En un mundo en el cual la colaboración es uno de los valores fundamentales, se propone la integración de los recursos tecnológicos a través de redes, que generen relaciones de cooperación y aprendizaje.



Referente Institucional de Educación Digital (RIED)

Es un docente de la escuela que es nominado por el equipo de conducción para favorecer la integración de la educación digital en proyectos educativos y en acciones con la comunidad.

Alumnos Referentes de Educación Digital (RED)

Son estudiantes que, por sus propios intereses y deseos, contribuyen a la construcción de proyectos y el aprendizaje entre pares. A partir de sus habilidades para el uso de recursos digitales, construyen una relación solidaria con los docentes para facilitar su integración.

Introducción

El material que se presenta a continuación propone una serie de actividades adecuadas al nivel, en un marco de creatividad, colaboración y juego, para desarrollar habilidades que permitan a los estudiantes ser productores de nuevos proyectos originales y diversos, relacionados con las problemáticas de las comunidades educativas y otros aspectos socioculturales relevantes.



Robótica y programación en la escuela resulta un medio y a la vez un fin educativo, es sinónimo de potenciar el desarrollo de habilidades y competencias en los estudiantes promoviendo el acceso a contenidos que los desafían a analizar, investigar, compartir y comunicarse con otros, resolver diversas problemáticas sociales, crear nuevas oportunidades y prepararse para su integración en el mundo del futuro.

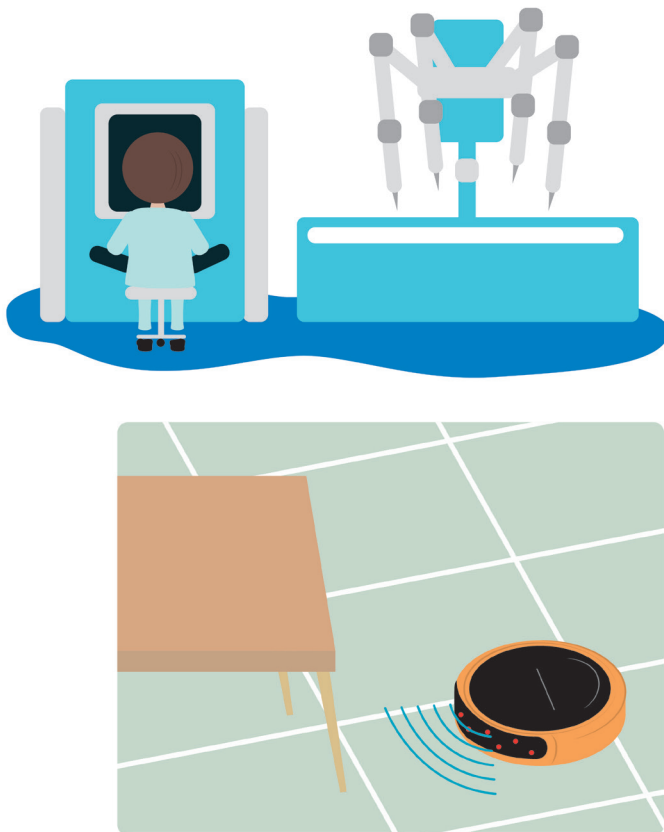
Estamos presentes en un momento histórico en el cual los robots salieron de la ciencia ficción hacia la vida real asociados a los circuitos digitales, los sistemas embebidos y la inteligencia artificial. Progresivamente estos campos de conocimiento ganan protagonismo y relevancia en distintos ámbitos del desarrollo social y económico. En este sentido, resulta fundamental pensar en propuestas educativas que involucren sistemas digitales de control, automatización, vinculados con programación y las ciencias de la computación, relacionados a la resolución de situaciones y problemas del mundo real, dónde se apliquen combinación de estrategias y conceptos para generar soluciones concretas.

En robótica y programación, confluyen tanto la lógica y la abstracción como la imaginación, la expresión y la capacidad de idear y de construir, en forma individual o con otros, promoviendo el trabajo en equipo, la colaboración y el aprendizaje entre pares, capacidades y competencias necesarias como modos de construcción de conocimiento, de interacción social.

¿Qué es un robot?

Un robot es un agente artificial electrónico o electromecánico, un dispositivo funcional y programable, capaz de realizar una acción o serie de acciones por sí solo para cumplir un objetivo específico a partir de la capacidad de percibir el mundo que lo rodea, procesar esta información y actuar en consecuencia.

Ejemplos:



ROBOT CIRUJANO

Cuenta con varios brazos que otorga mayor precisión y destreza que la mano humana. No es completamente autónomo ya que ejecuta las decisiones tomadas por un profesional. Permite tener una visión en alta definición del campo de operación e incluso realizar intervenciones quirúrgicas a distancia ya que el profesional lo puede controlar el robot remotamente.

ROBOT ASPIRADORA

Puede limpiar sin intervención de un humano. Cuenta con un sistema de navegación que le permite detectar obstáculos y construir un mapa de todos los recovecos del hogar.

Hay robots que, además, aprenden a partir de su interacción con el ambiente, con humanos o con otros dispositivos, utilizando algoritmos de aprendizaje automático: programas informáticos capaces de generalizar comportamientos a partir del análisis de datos, integrando conceptos de inteligencia artificial.

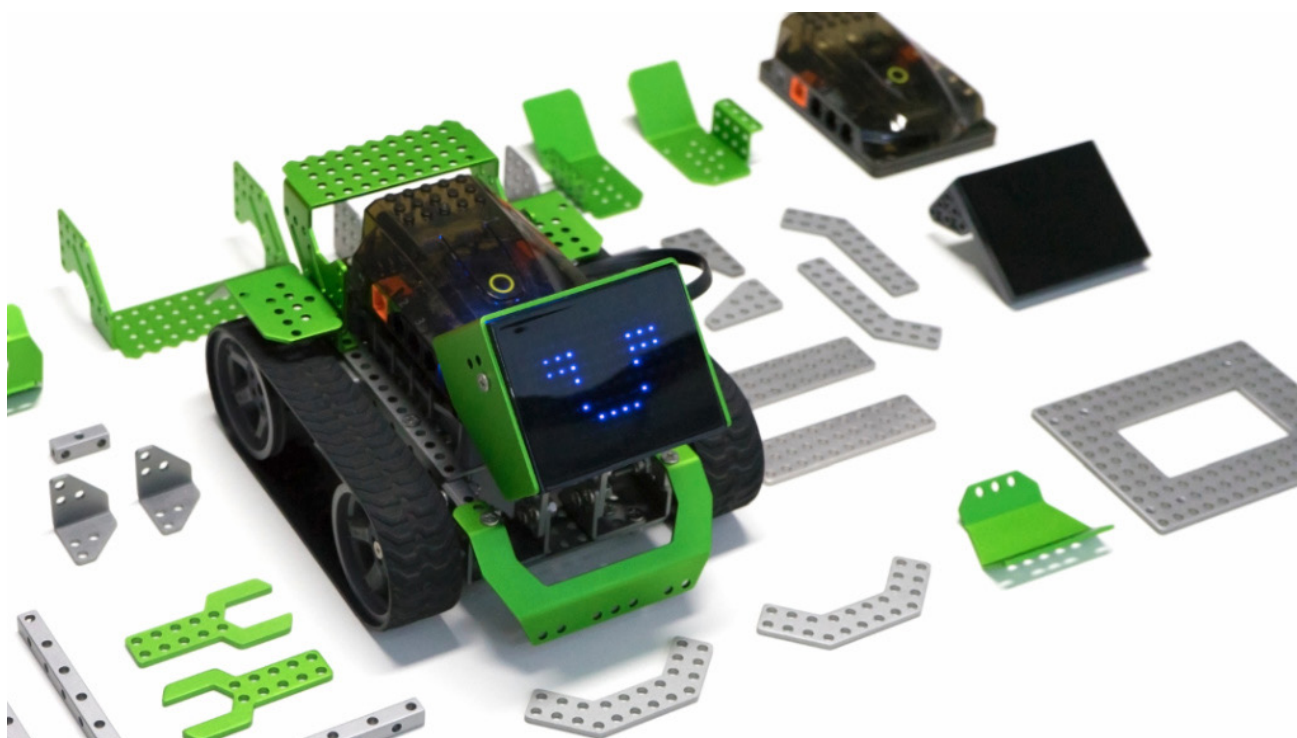
Los bots son programas computacionales que procesan de forma autónoma la información que reciben y son capaces de tomar decisiones en tiempo real. Un ejemplo son los chat bots interactivos que nos sugieren mejores precios, dónde cenar, qué espectáculo ver o qué servicio contratar y hasta realizar un diagnóstico a partir del análisis que realizan.

Desde un punto de vista funcional, un robot está compuesto por dos niveles: el nivel físico y el de procesamiento. El nivel físico comprende la estructura electromecánica, los circuitos electrónicos, la placa programable, la fuente de alimentación y los dispositivos que le permiten interactuar con el entorno. Estos últimos son de dos tipos:

- **Sensores:** capturan información física (luz, temperatura, humedad, distancia, sonido, etc.) y la convierten en señales digitales, legibles por una computadora.
- **Actuadores:** motores, por ejemplo, permiten llevar adelante cualquier acción programada, pueden ser de distinto tipo: hidráulicos (se accionan por presión del agua), neumáticos (por presión del aire) y eléctricos (los más utilizados).

El nivel de procesamiento está constituido por los elementos computacionales y el software que regulan el comportamiento del robot.

Emotibot y software MyQode



El robot posee una estructura metálica a la que se pueden anexas piezas de expansión como vigas, placas y otros sensores.

Las rutinas de programación se generan a partir del recurso multiplataforma MyQode, basado en Scratch 3.0 y compatible con Arduino.

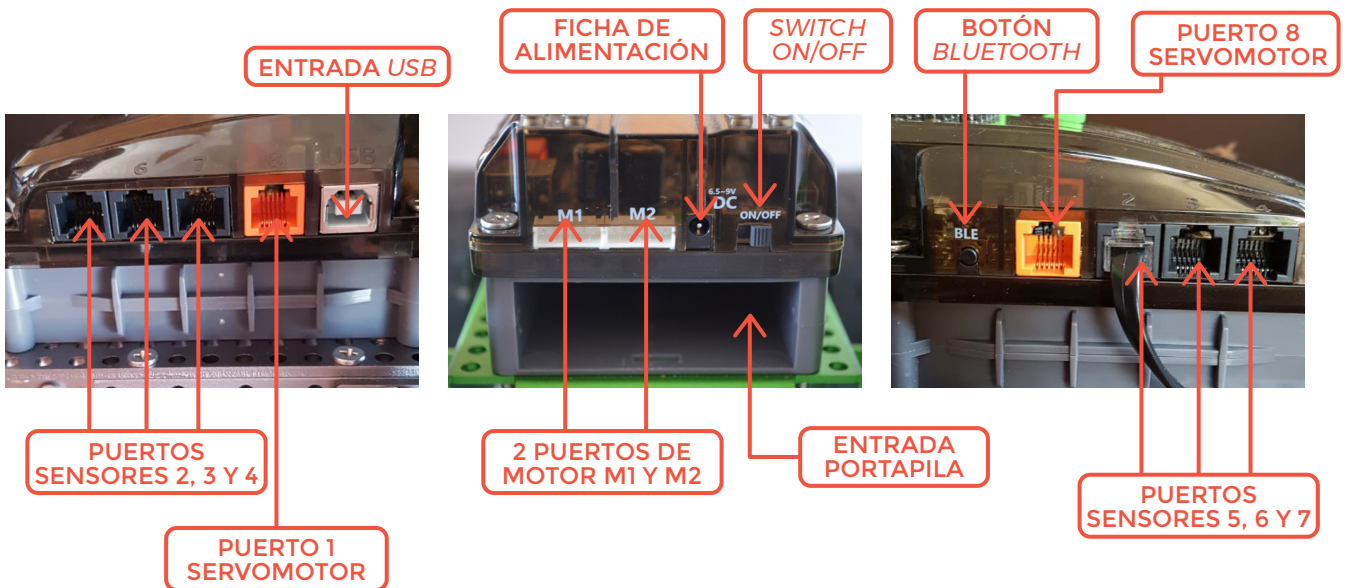
Piezas y características

 Placa Qmind Plus x 1	 Soporte de batería x 1	 Batería x 1	 Panel LED x 1	 Motor x 2
 Llanta x 2	 Rueda x 2	 Sensor de luz x 1	 Sensor ultrasónico x 1	 Sensor de sonido x 1
 Sensor Giroscopio x 1	 Destornillador x 1	 Llave inglesa x 1	 Base rectangular x 1	 Bracket U Grande x 1
 Tornillo M4x42mm x 10	 Tornillo M3x8mm x 5	 Tornillo M4x20mm x 10	 Tornillo M4x9mm x 35	 Tuerca M4 x 60
 Tornillo M4x30mm x 10	 Tornillo M4x15mm x 30	 Tuerca con traba x 5	 Cable Micro USB x 1	 Cable USB x 1
 Oruga x 2	 Viga 6 agujeros x 2	 Viga 10 agujeros x 2	 Eje Bola x 1	 Cable RJ25 150mm x 2
 Cable RJ25 200mm x 3	 Viga 2 agujeros x 1	 Bolillero x 4	 Tubo plástico chico x 2	 Tubo plástico grande x 2
 Placa triangular x 4	 Bracket L x 2	 Placa angular x 2	 Placa rectangular x 2	 Placa angular C x 2
 Placa brazo x 2	 Bracket U chico x 1	 Paragolpes Chico x 1	 Guardabarro Der x 1	 Guardabarro izq x 1
 Cable motor x 2	 Aleron x 1	 Bracket U x 1	 Paragolpes grande x 1	 Manual de usuario x 1

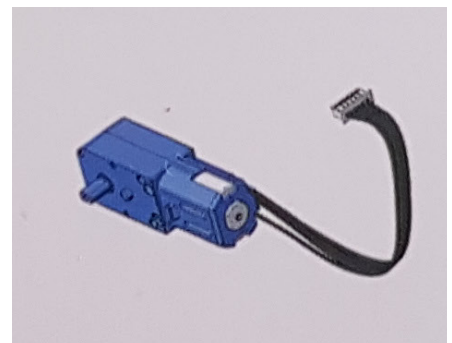
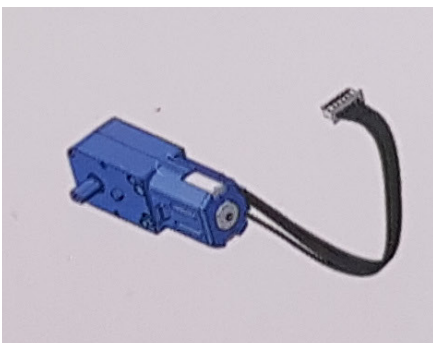
- Sensor ultrasónico con *LED RGB* incorporados (color programable)
- Sensor de sonido
- Sensor de luz
- Giroscopio
- Matriz de *LED* 14x10
- Microcontrolador *Qmind plus*
- Batería de litio recargable de 2200mAh
- 4 Ruedas, 2 cubiertas, 2 orugas
- 2 Motores
- Piezas constructivas para chasis
- Tuercas y tornillos, destornillador, llave fija.
- Cables RJ11 y cable *USB*.

Microcontrolador *Q-mind Plus*

Es una placa programable que permite al robot ser comandado. *Q-mind Plus* es una adaptación de *Arduino Uno* que ofrece fácil y rápida interpretación de conexión a puertos.



Actuadores o componentes de salida



Se trata de elementos electrónicos que conectamos en las salidas del microcontrolador y ejecutan acciones, fundamentalmente encender y apagar. Los más usuales son los motores, *led*, dispositivos de sonido, relés y servos.

Emotibot tiene *leds* RGB en placa y en sensor ultrasónico, *buzzer* para emisión de sonido, 2 motores para generación de movimiento y una matriz de 14 x 10 *leds*.



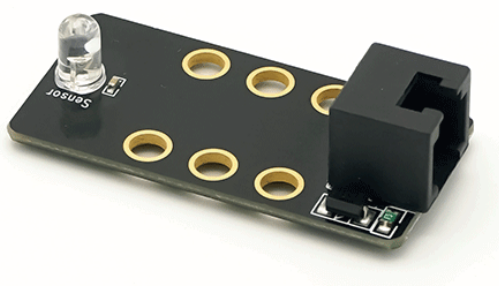
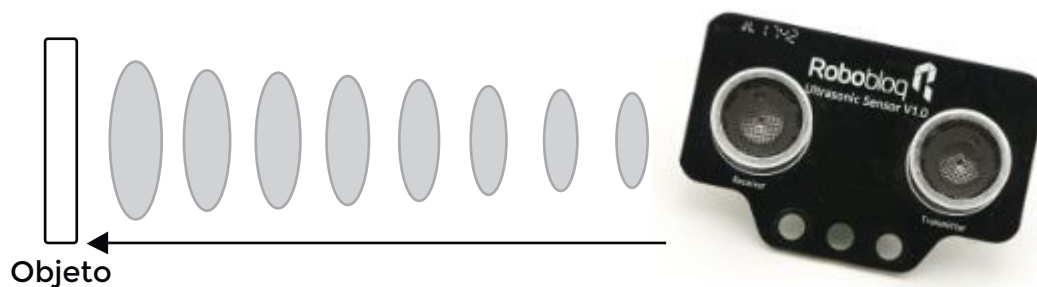
BOTÓN INTERRUPTOR

En la placa microcontroladora se encuentra un botón que puede ser programado para darle encendido a una rutina de programación generada.

Sensores

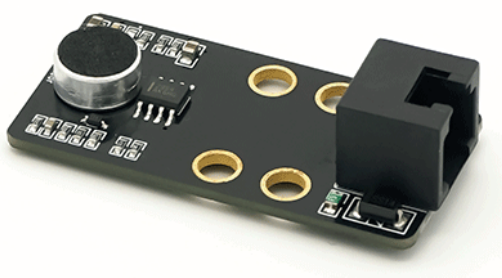
SENSOR ULTRASÓNICO:

es un módulo que detecta la distancia existente entre el robot y un objeto. Permite percibir el entorno a través de la emisión y recepción de ondas ultrasónicas. A través de este sensor ingresamos datos para poder utilizarlos en la programación. Por ejemplo, se puede programar al robot para que evite automáticamente obstáculos y que frene ante ellos.



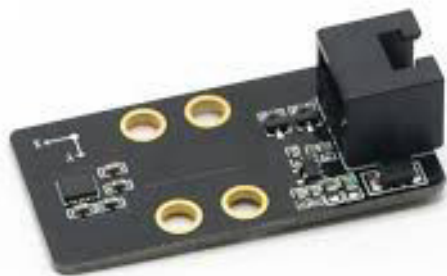
SENSOR DE LUZ:

es un módulo que detecta el nivel de luz y produce una señal representativa respecto a la cantidad de luz detectada



SENSOR DE SONIDO:

es un módulo capaz de registrar niveles de ruido analizando el sonido ambiental y mostrando los datos recopilados en dB (decibeles).

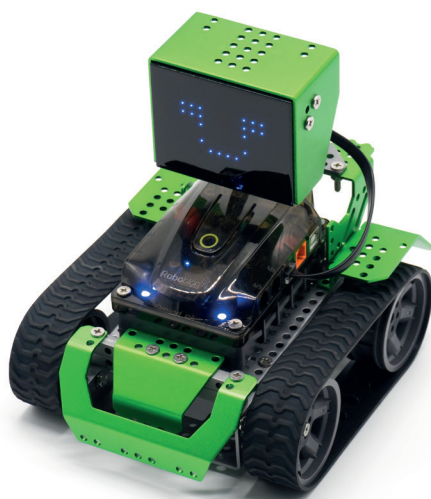


GIROSCOPIO:

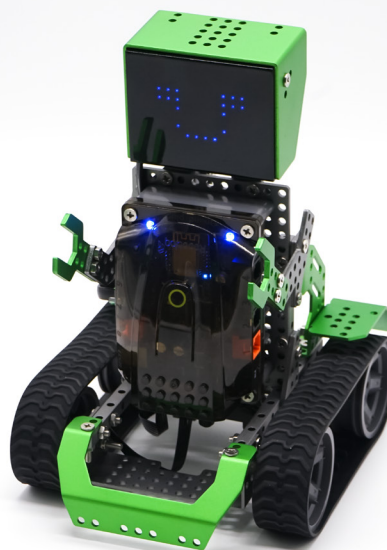
es un módulo que detecta la posición y rotación exacta en los 3 ejes: X, Z e Y.

Construcción paso a paso de Emotibot

Emotibot puede tomar al menos 6 formas distintas. Cada modelo de armado, con su guía paso a paso de construcción, se encuentra en el anexo.



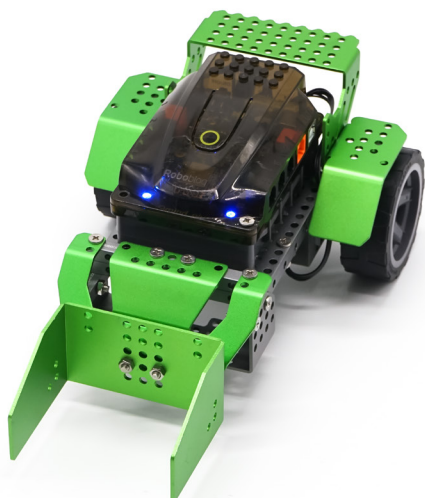
CAPTAIN ALLOY (1)



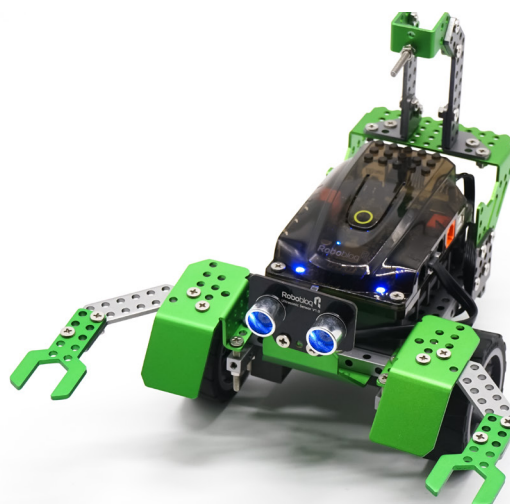
CAVALIER (2)

(1) Ver anexo de construcción - p.100.

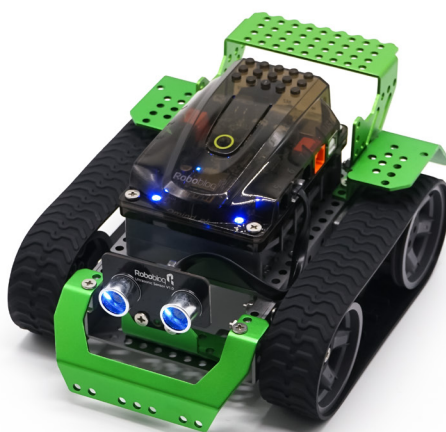
(2) Ver anexo de construcción - p.105.



THE DOZER (3)



SCORPOID (4)



THE VOYAGER (5)

Potencialidades/ Armados creativos

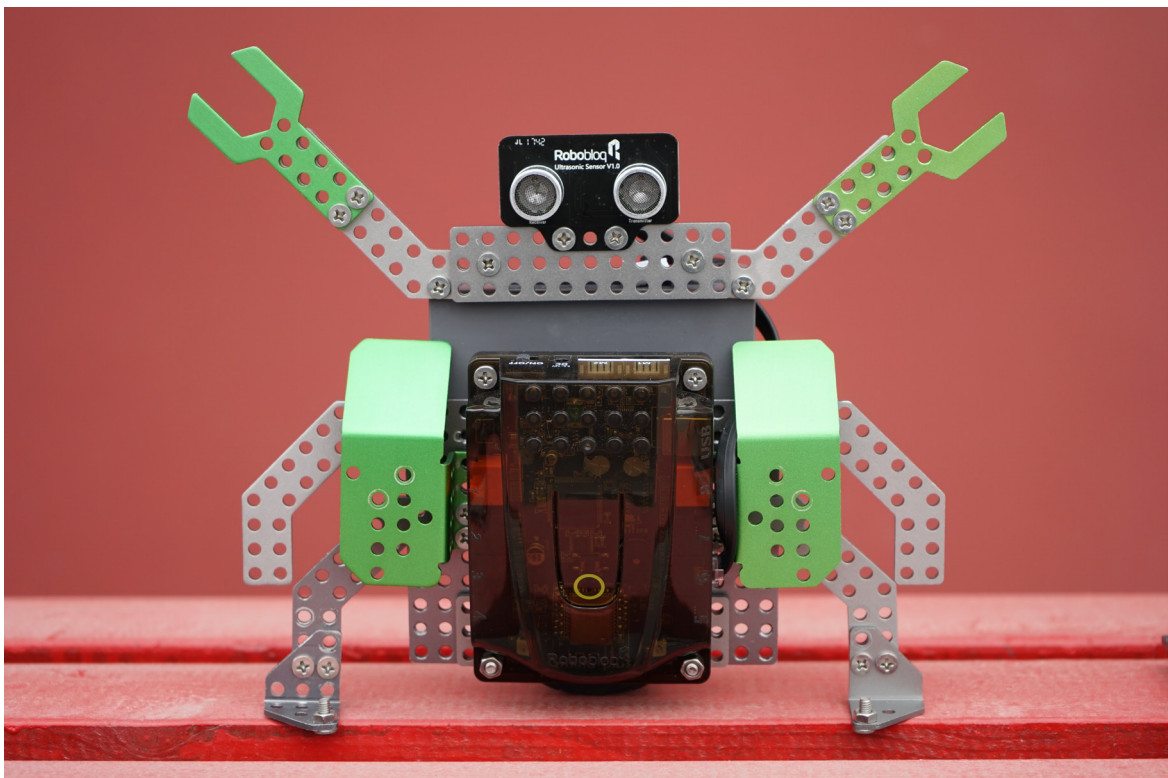
Emotibot también permite generar otras variables de armado. Explorar con creatividad es una de las posibilidades que más divertidas que posee el dispositivo.

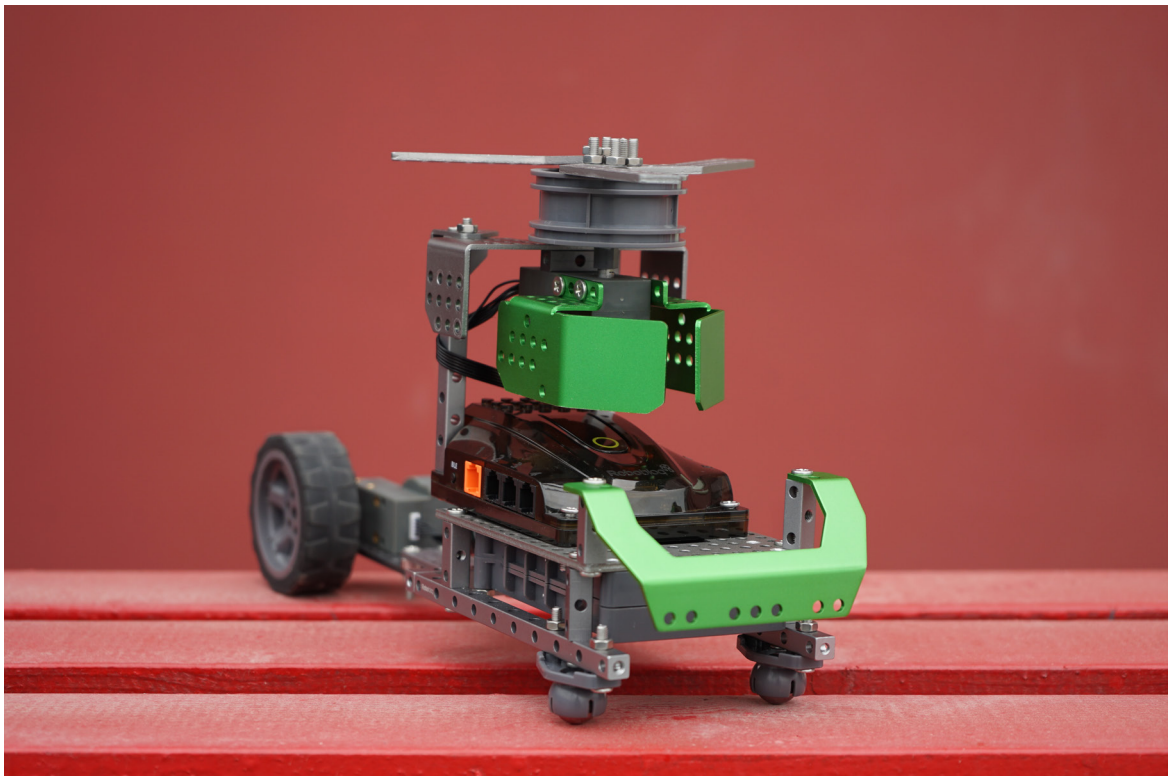
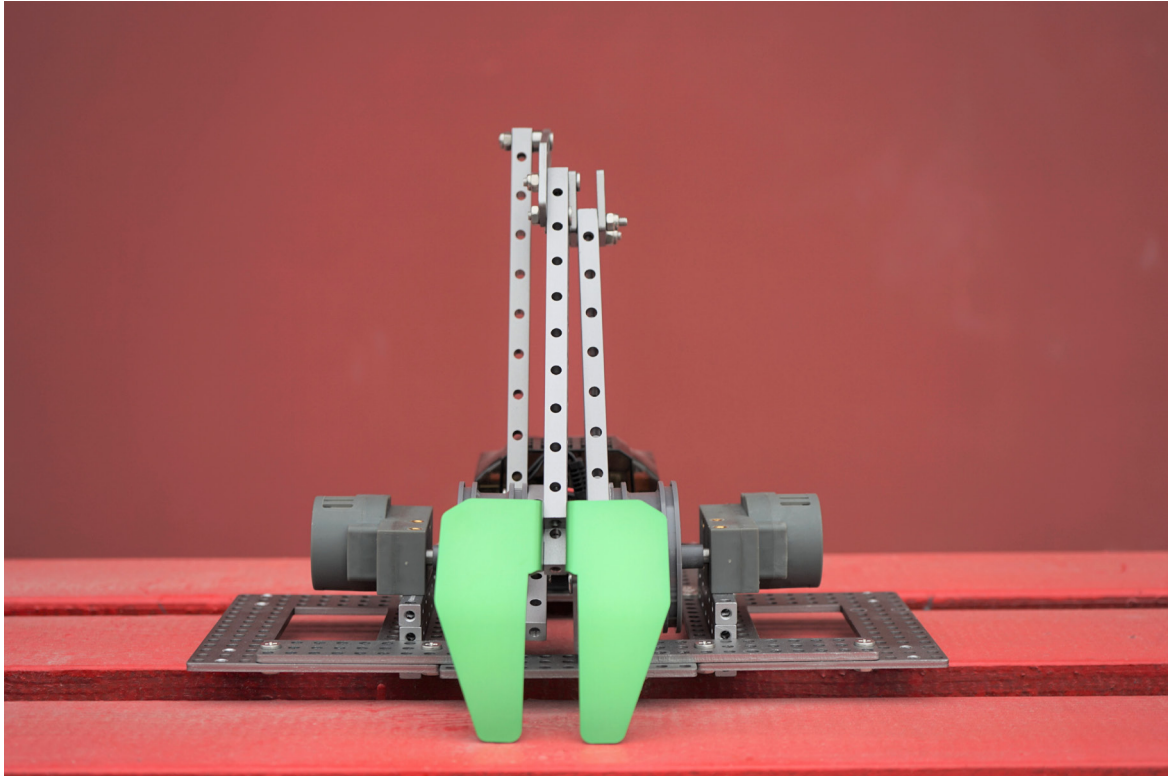
(3) Ver anexo de construcción - p.110.

(4) Ver anexo de construcción - p.117.

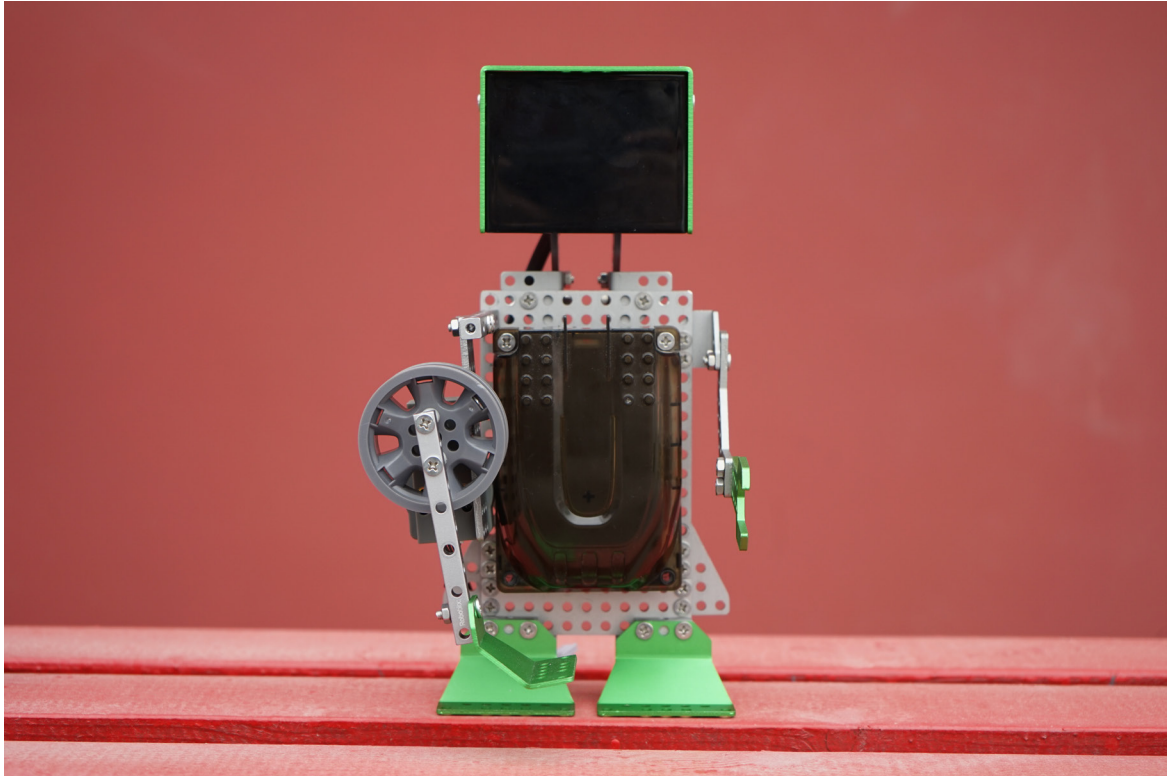
(5) Ver anexo de construcción - p.113.

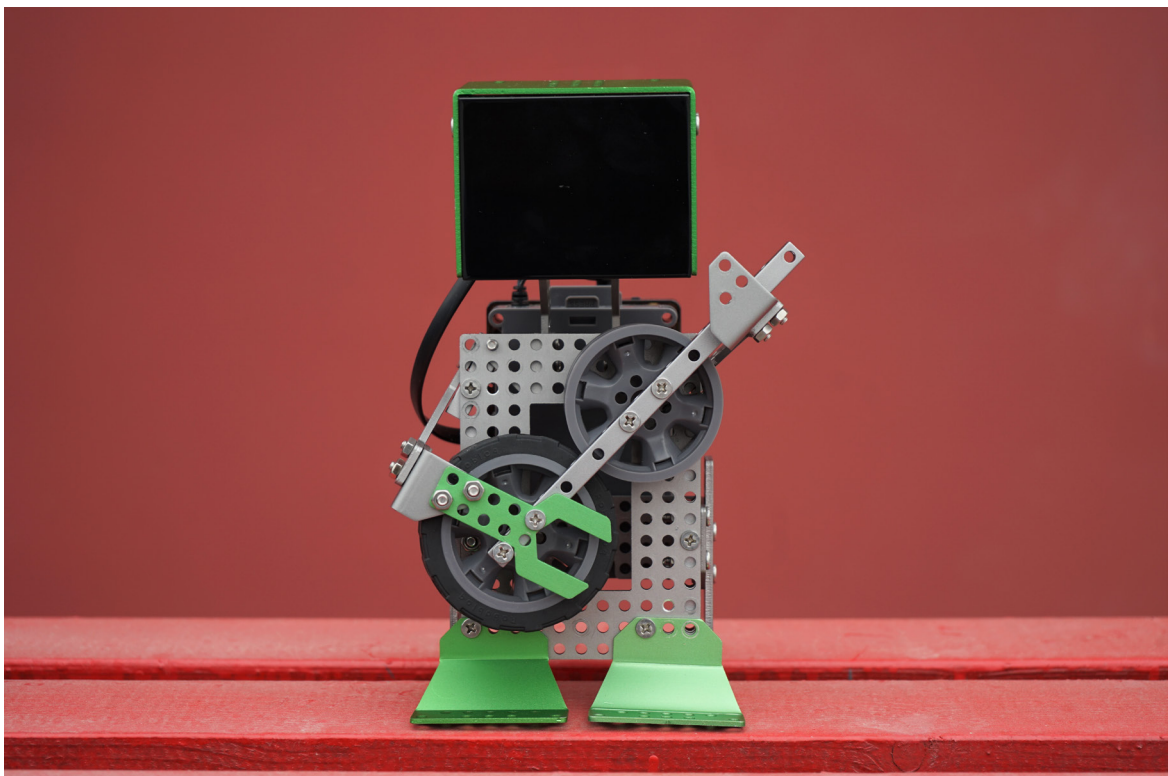
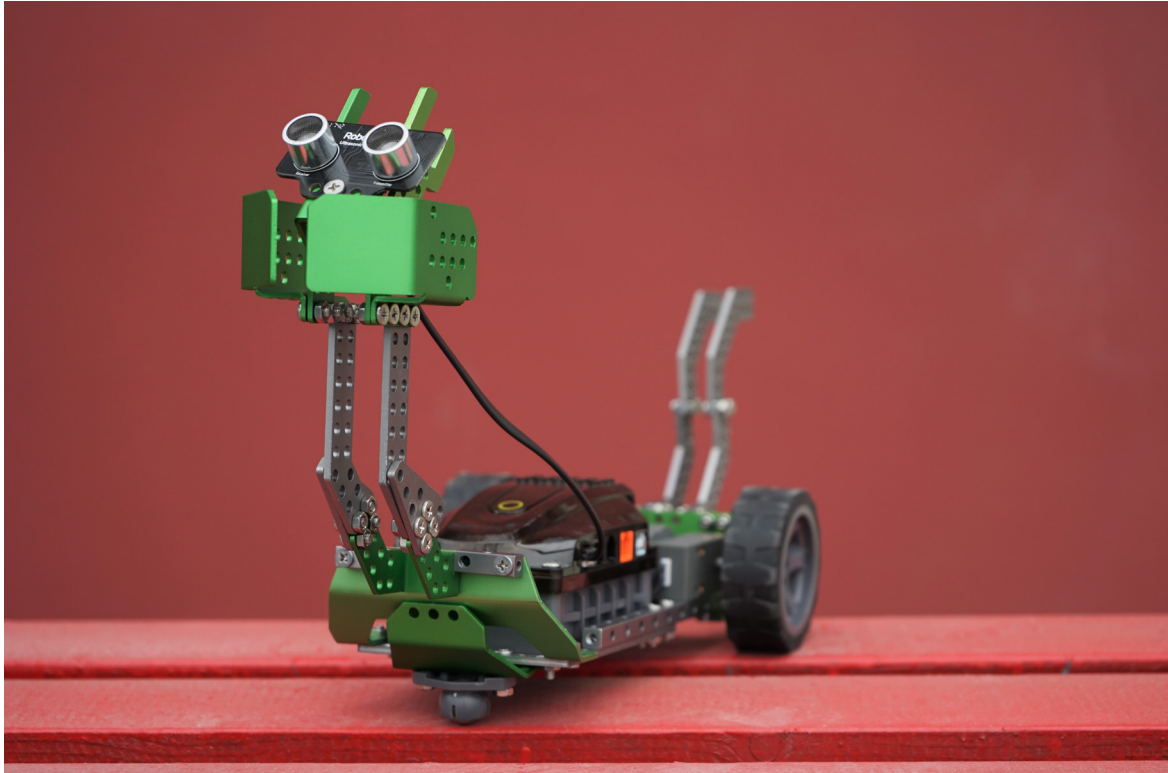
APRENDER CONECTADOS





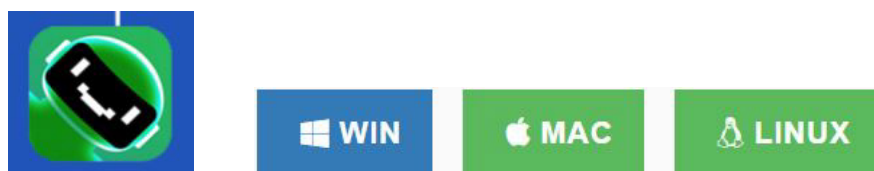
APRENDER CONECTADOS





Recurso MyCode

Basado en el desarrollo de código abierto *Scratch 3.0*, *MyCode* es un entorno gráfico muy intuitivo diseñado con bloques de instrucciones categorizados en paneles por colores.



Es compatible con todos los sistemas operativos y se puede instalar desde el siguiente link:
<https://www.robobloq.com/support/download>

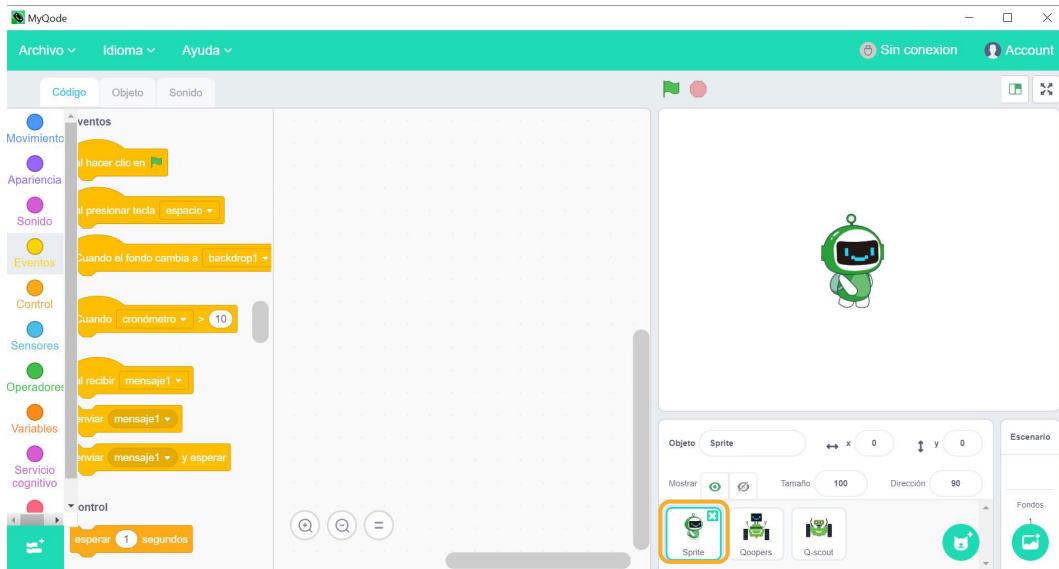
Con este recurso se puede crear programas tanto para programar el robot como para realizar animaciones, simulaciones, videojuegos, etc.



Seguir los siguientes pasos para programar a **Emotibot** :

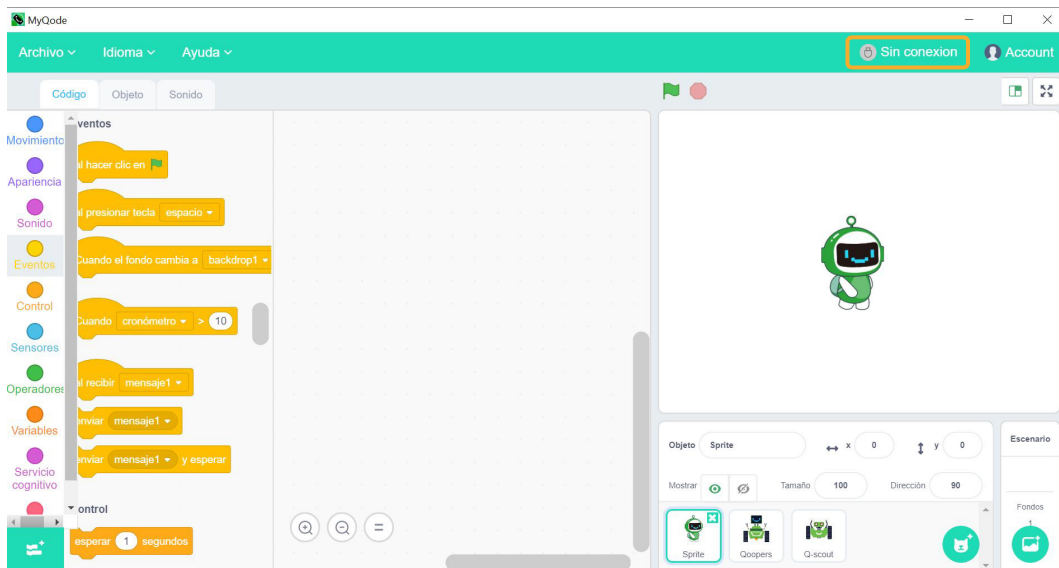
1- Abrir el *software MyCode* y seleccionar el objeto *Sprite*

APRENDER CONECTADOS

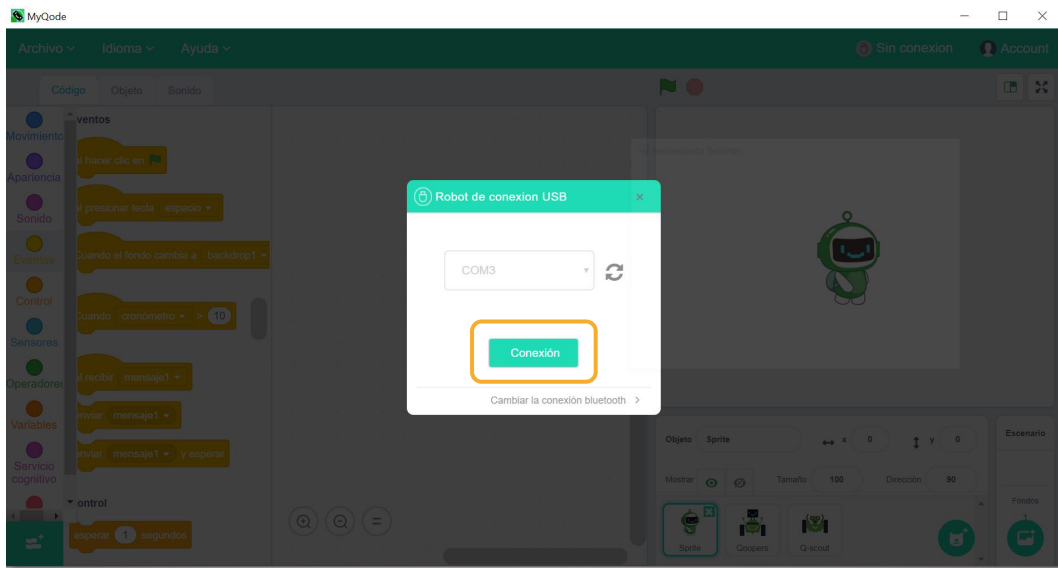


2- Conectar el robot al puerto USB

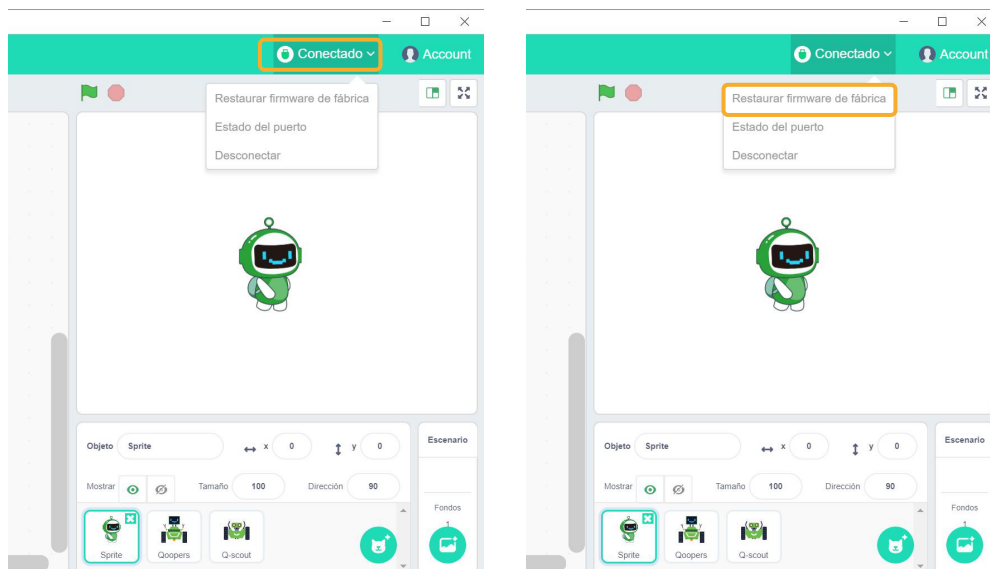
3- Hacer clic en “Sin conexión” para establecer conexión con el puerto COM.



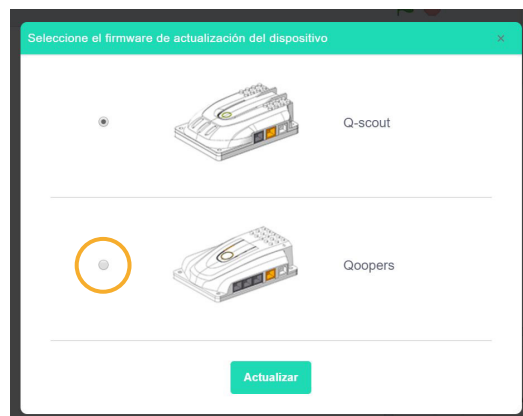
4- Se abre una ventana emergente que detecta el puerto conectado. Apretar el botón “Conexión”



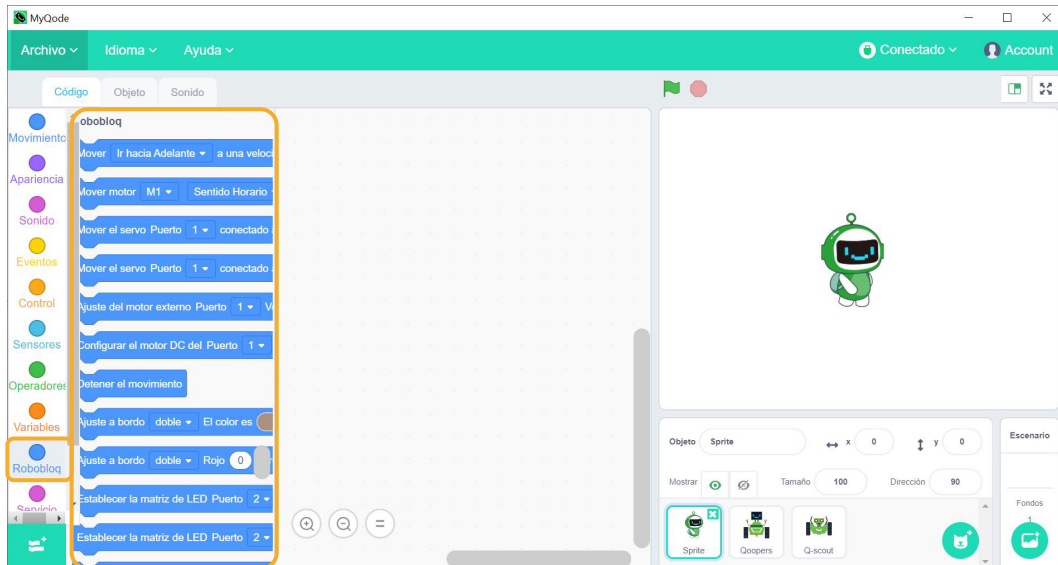
5- Cuando el estado pasa de “Sin conexión” a “Conectado”, proceder a desplegar la solapa de acciones y seleccionar “Restaurar *firmware* de fábrica”.



6- Aparece una ventana emergente que permite elegir la placa del robot. **Emotibot** tiene la placa identificada con el nombre *Qoopers*.



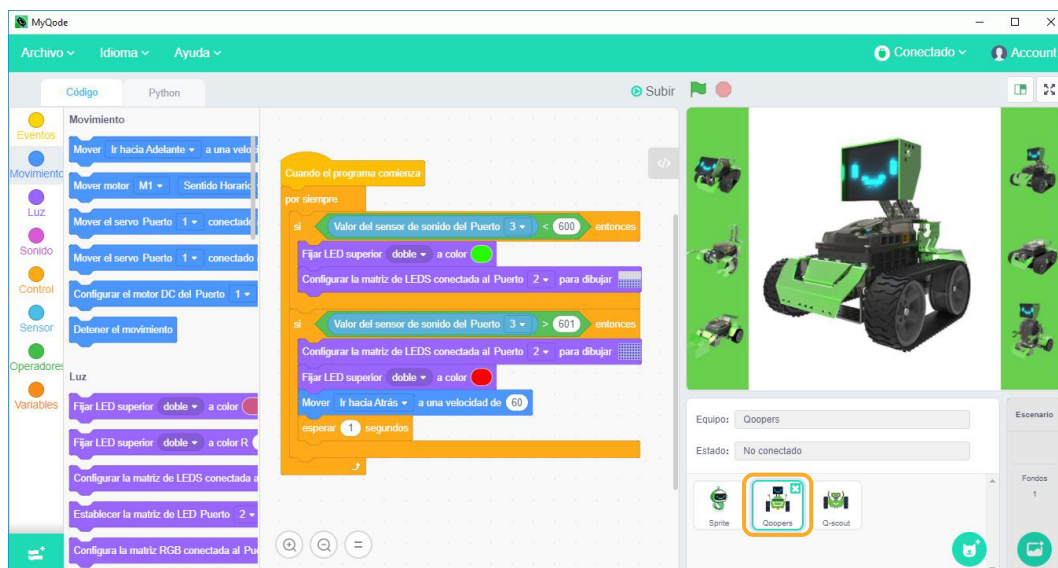
7- Automáticamente se suma a los paneles de comandos, uno nuevo llamado “Robobloq” que permite ingresar instrucciones para el robot:



De esta manera, la secuencia generada puede ser iniciada a partir de cualquier evento, volviéndose **Emotibot** un dispositivo posible de reaccionar ante comandos de inteligencia artificial, por ejemplo. Los programas se cargan en el robot a través del cable USB o Bluetooth.

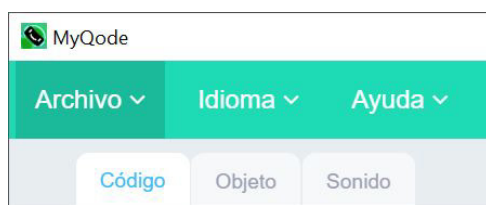
Emotibot autónomo

Una vez que el código es cargado en **Emotibot**, éste puede ejecutar la rutina en forma autónoma si se utilizó el evento adecuado para que esto suceda.

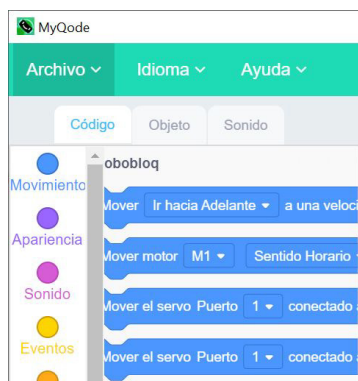


Pestaña Código

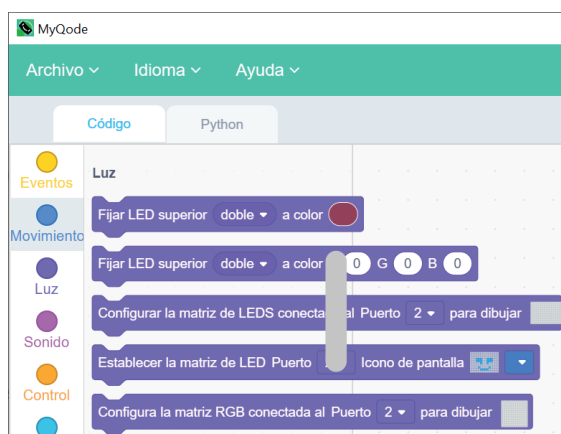
Las rutinas de programación se construyen a partir de instrucciones ó comandos que se arrastran a la ventana de trabajo. Estas instrucciones se encuentran dentro de categorías que están organizadas de acuerdo a la función que realizan.



CATEGORÍA EVENTOS: contiene comandos a partir de los cuales, la rutina de programación inicia, reacciona o se altera ante determinado evento.



CATEGORIA MOVIMIENTO: contiene comandos que permiten encender, apagar y configurar el funcionamiento de los motores.



CATEGORIA LUZ: contiene comandos que encienden luces en la placa programable, en el sensor ultrasónico y en la matriz de *leds*. En este último componente es posible customizar la ubicación de los *leds* encendidos, no así su color. En el caso de los *leds* del sensor ultrasónico y los de la placa, al ser todos *RGB*, es posible personalizar el color.

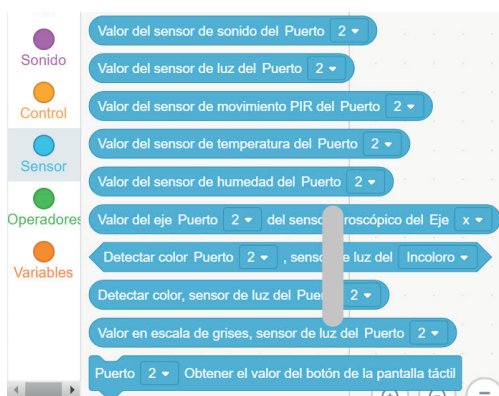
APRENDER CONECTADOS



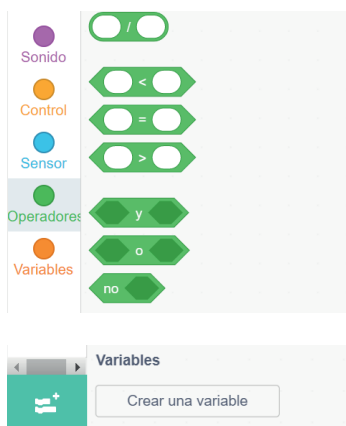
CATEGORIA SONIDO: es un módulo capaz de registrar niveles de ruido analizando el sonido ambiental y mostrando los datos recopilados en dB (decibelios).



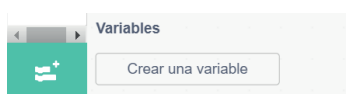
CATEGORIA CONTROL: contiene comandos que manejan tiempos de espera, bucles de repetición y condicionales.



CATEGORIA SENSOR: los comandos de este panel arrojan valores detectados por los sensores que pueden estar conectados a **Emotibot**.



CATEGORIA OPERADORES: Este panel otorga instrucciones para realizar operaciones lógicas y matemáticas que permiten generar comparativas entre los datos.

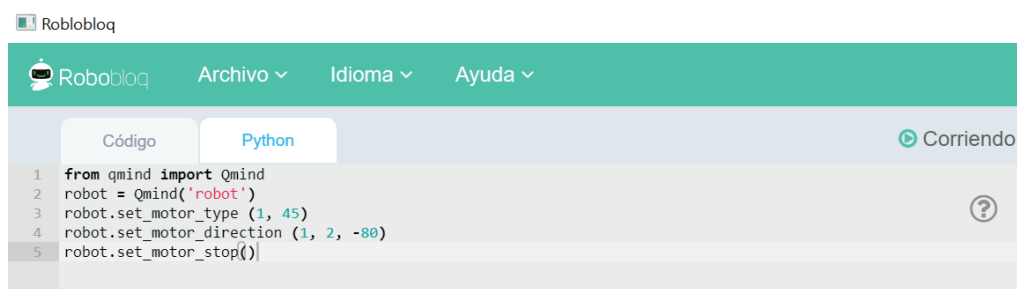


CATEGORIA VARIABLES: ofrece la posibilidad de creación de variables. Puede haber muchas variables generadas en una misma rutina de programación, es importante que tengan un nombre representativo de acuerdo a su función o al dato que alojan.

Pestaña *Python*

Al acceder es posible comenzar a escribir código en lenguaje de programación *Python* para luego subirlo al robot.

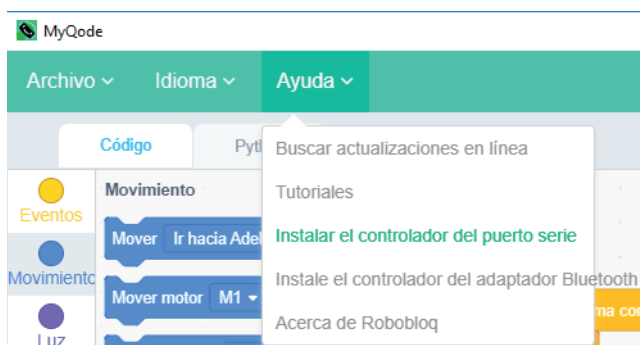
En esta rutina se programa a **Emotibot** para que se mueva hacia adelante, gire y luego se detenga.



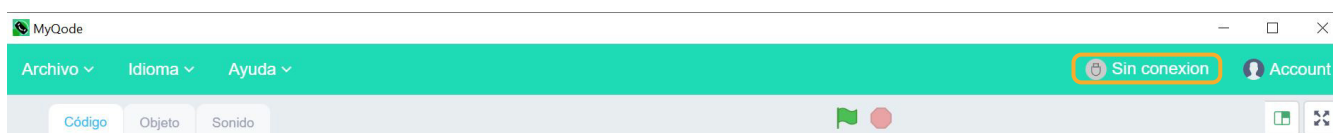
- En los dos primeros renglones se nombra al robot y se determina la placa a utilizar.
- En la tercer línea se señala el movimiento hacia adelante (Parámetro 1: adelante 1, atrás 2, izquierda 3, derecha 4) y la velocidad de 45 rpm (Parámetro 2: rango de velocidad -90 to 90).
- En la cuarta línea se determina el motor que se va a usar (Parámetro 1: motor izquierdo o motor derecho, rango 1 to 2); el sentido de rotación (Parámetro 2: sentido horario o antihorario, rango 1 to 2) y la velocidad (Parámetro 3: rango de velocidad -90 to 90)
- En la quinta línea, los motores se detienen.

Conexión *USB*

Cuando se conecta por primera vez, es necesario instalar el controlador que permite la lectura de los puertos serie (en donde esta conectado el USB):



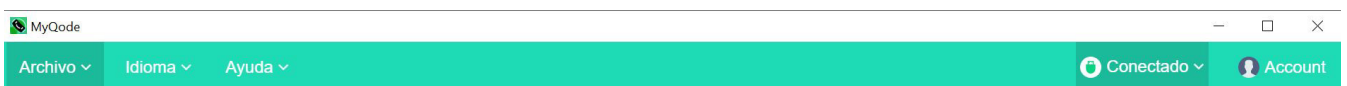
Seleccionar el botón “Sin conexión”:



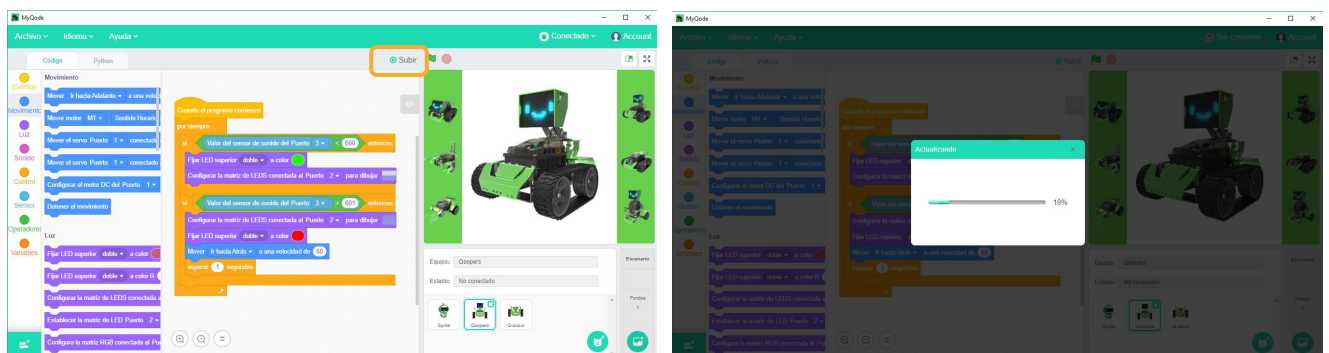
Aparecerá la ventana emergente con la información de detección del puerto. Para que la lectura sea posible, el robot debe estar conectado por cable *USB* a la *netbook*. Al detectar el puerto *COM*, pulsar el botón “Conexión”:



Luego, el estado cambiará a “Conectado”:



Ahora estarán las condiciones para cargar la rutina a Emotibot. Para ello, es necesario seleccionar el botón “Subir” y esperar que el proceso finalice.



Vinculación *Bluetooth*

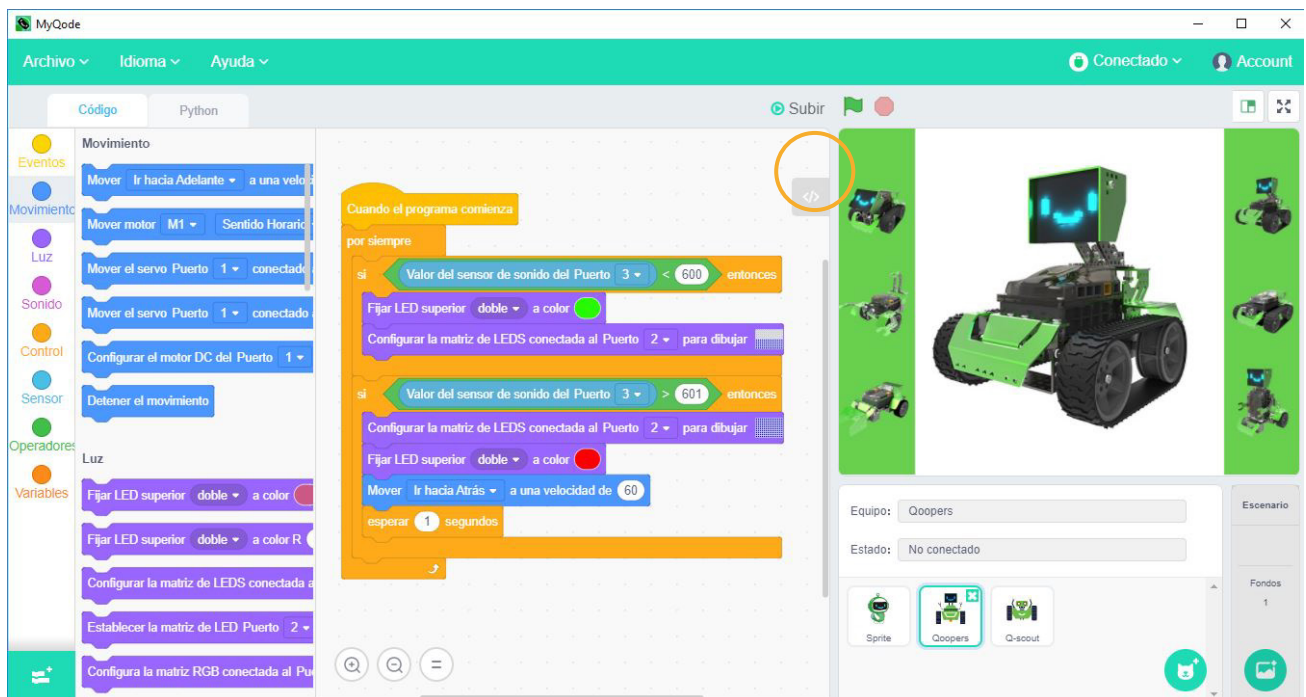
Será necesario instalar el controlador que permitirá la vinculación con el módulo Bluetooth de la placa. Luego, en la ventana emergente, seleccionar “Cambiar la conexión Bluetooth”. Recordar tener activo el Bluetooth de la netbook para que la lectura sea efectiva.

Una vez generada la rutina de programación, es posible subirla a placa para que el robot funcione autónomamente. Para ello, es necesario seleccionar el botón “Subir” y esperar que el proceso finalice.



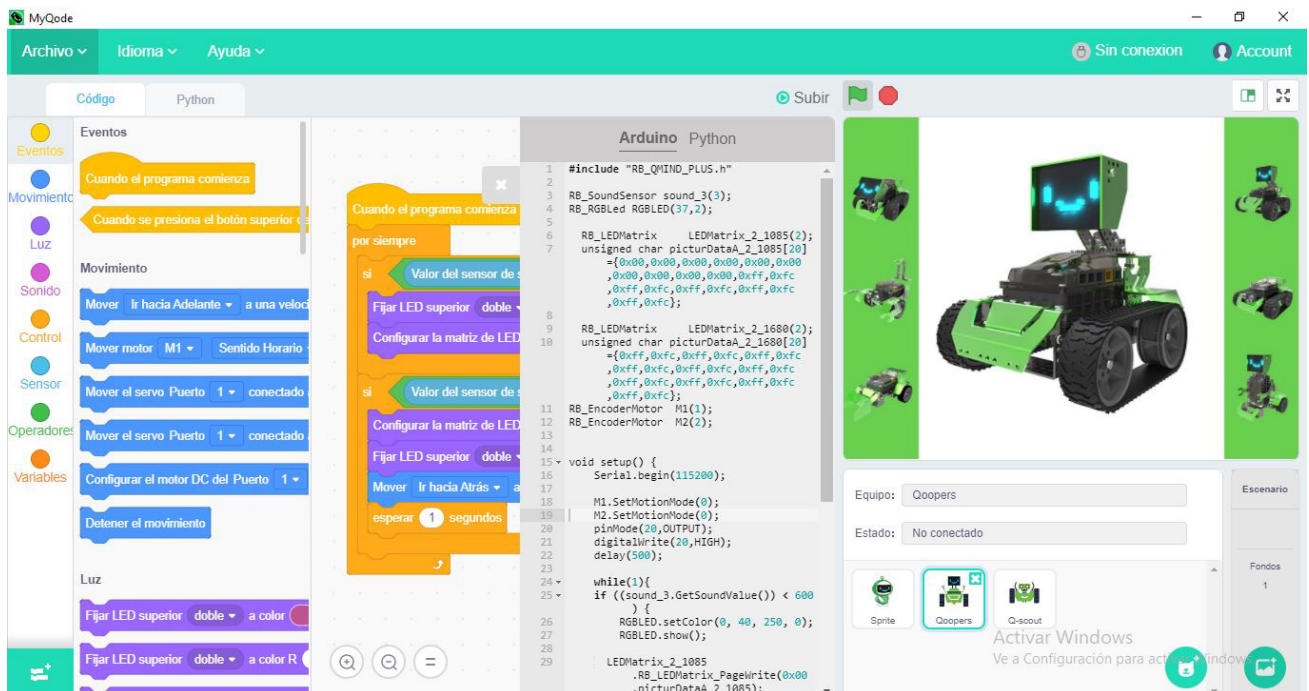
Traducción de bloques a *Arduino IDE*

El *software* permite el pasaje del entorno gráfico al textual presionando el siguiente botón:

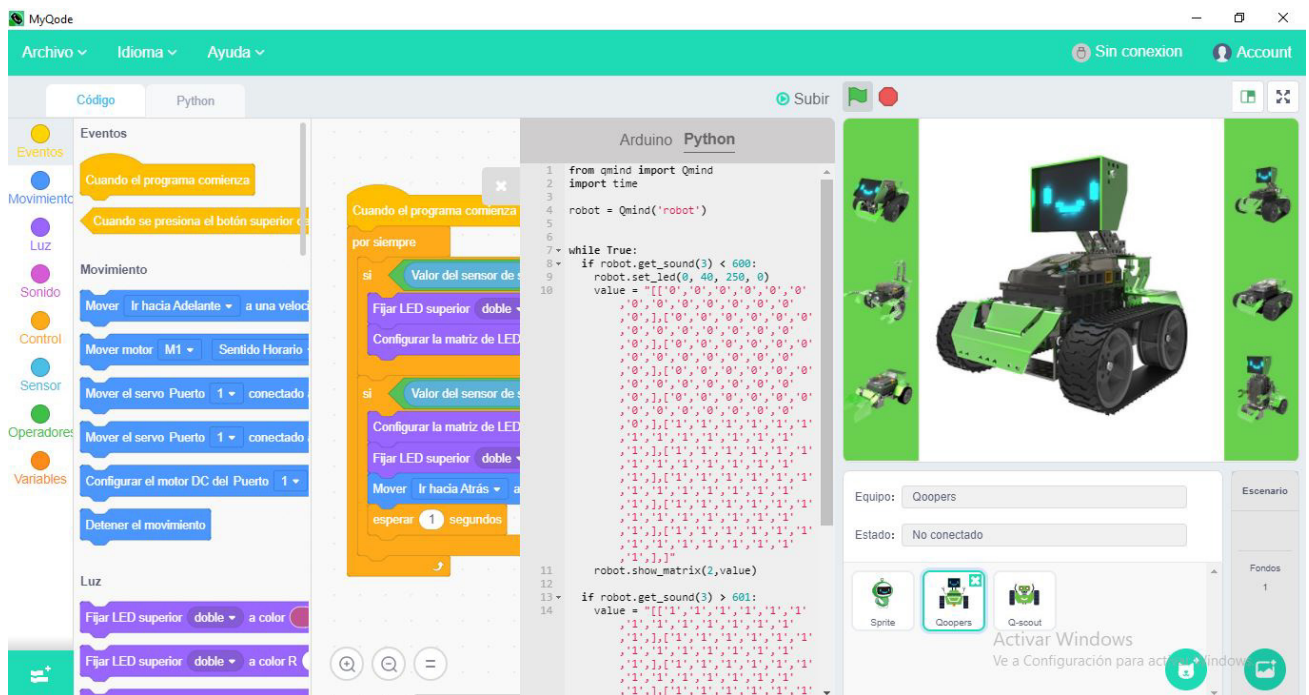


La ventana emergente traduce el código en bloques a Arduino IDE. Esto permite la incorporación de la lectura de código escrito de manera gradual.

APRENDER CONECTADOS

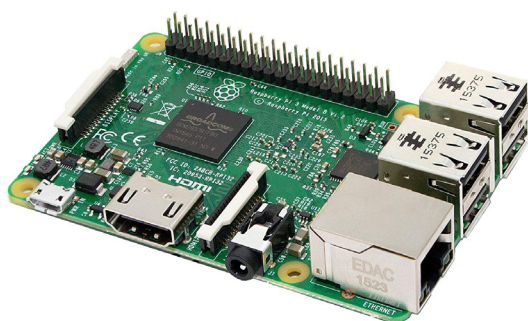


Siguiendo el mismo procedimiento de pasaje de bloque a texto, es posible elegir código escrito en *Python*, en lugar de elegir *Arduino IDE*.



Potencialidades/ Compatibilidad con *Raspberry Pi*

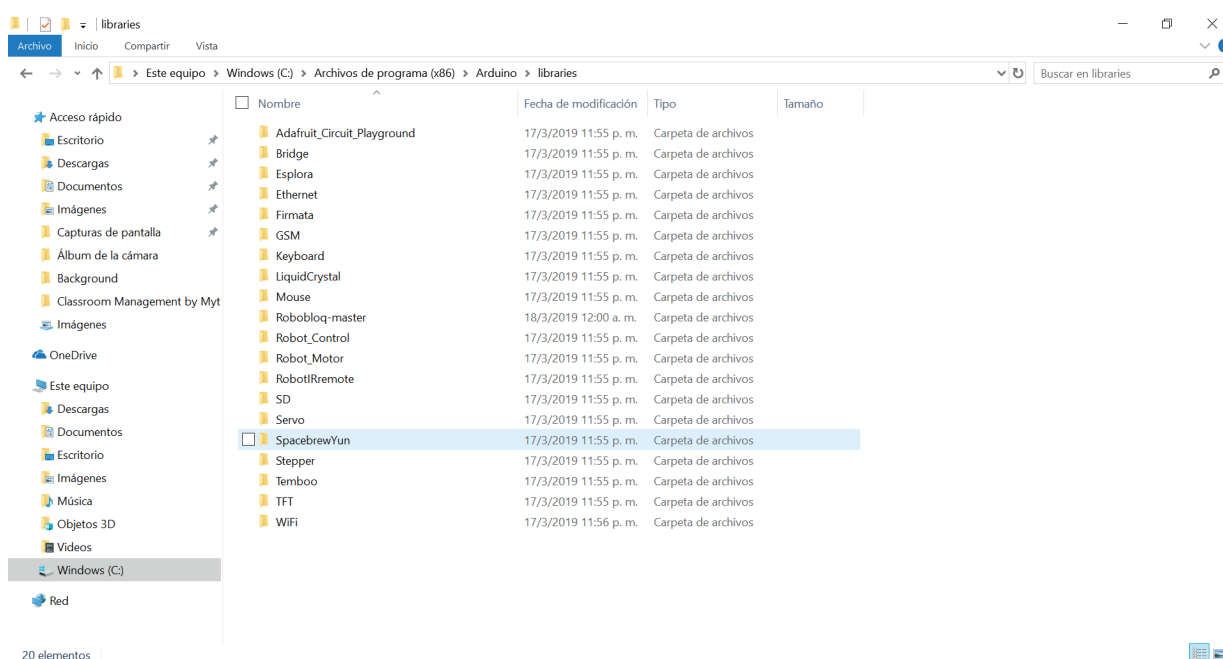
La placa *Raspberry Pi* se puede conectar a **Emotibot** mediante un cable USB 2.0. *Q-mind Plus* al ser una adaptación de *Arduino Uno*, es totalmente compatible con *Raspberry Pi*, y su combinación permite la creación de otro tipo de proyectos.



Raspberry Pi es un computadora de tamaño reducido, portable y bajo costo. Su sistema operativo está grabado en una tarjeta *SD*, la cual también almacena datos. Permite la conexión de otros complementos (sonido *mouse*, monitores, etc) a través de los puertos de la placa. Se utiliza para para desarrollar proyectos en programación y robótica con amplios alcances.

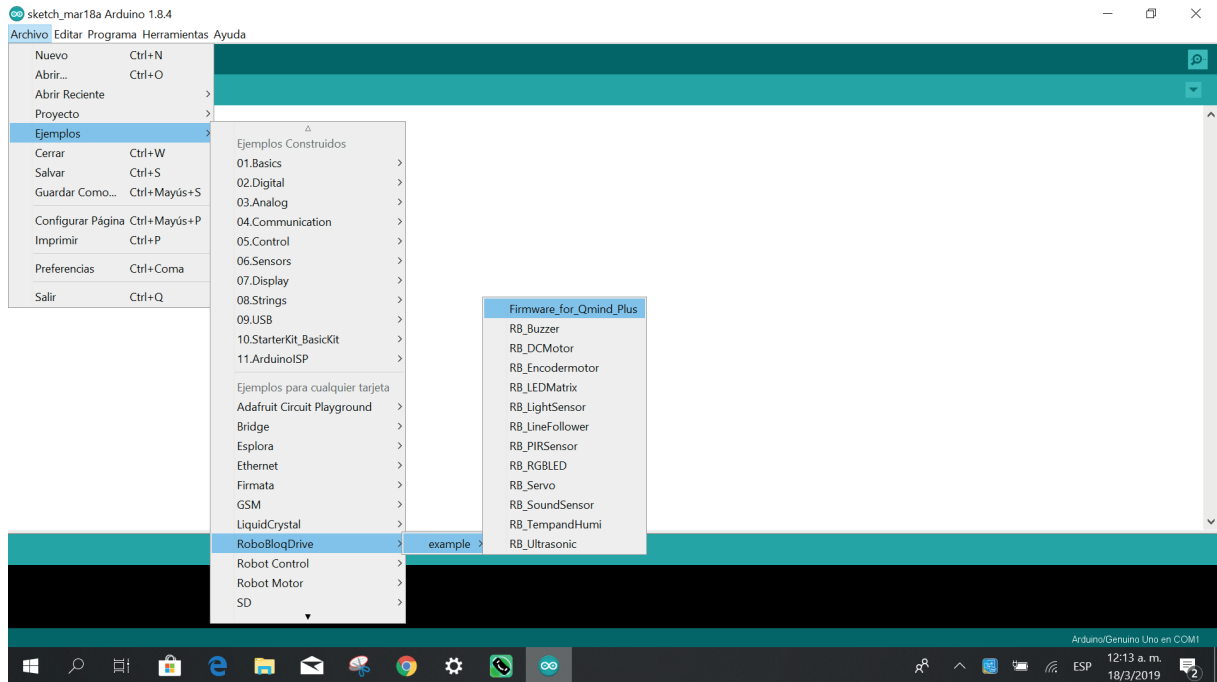
Para conseguir que la *RPi* se comuniquen con *Q-mind plus* es preciso admitir la programación en *Arduino IDE*. Pasos para lograr que **Emotibot** admita programación en lenguaje C++.

1. Instalar el controlador de *Arduino IDE* según su sistema operativo.
 - Para instalar en *Windows*: <https://downloads.arduino.cc/arduino-1.8.4-windows.exe>
 - Para instalar en *Mac OS X*: <https://downloads.arduino.cc/arduino-1.8.4-macosx.zip>
 - Para instalar en *Linux*: <https://downloads.arduino.cc/arduino-1.8.4-linux32.tar.xz>
2. Descargar e instalar la biblioteca de *MyQode*: <https://codecademy.com/Robobloq2018/Robobloq/zip/master>
3. Descomprimirla y mover el archivo "*Robobloq Master*" a la ruta `\\c\\program files\\Arduino\\libraries`, tal y como lo muestra la siguiente imagen:



3. Abrir el IDE de *Arduino*.

4. Seguir la ruta "archivo-ejemplos- RoboBloqDrive-Example- Firmware for Q-mind plus" tal y como lo muestra la siguiente captura de pantalla:



Una vez realizados todos los pasos, es posible codificar **Emotibot** desde la *Raspberry Pi*.

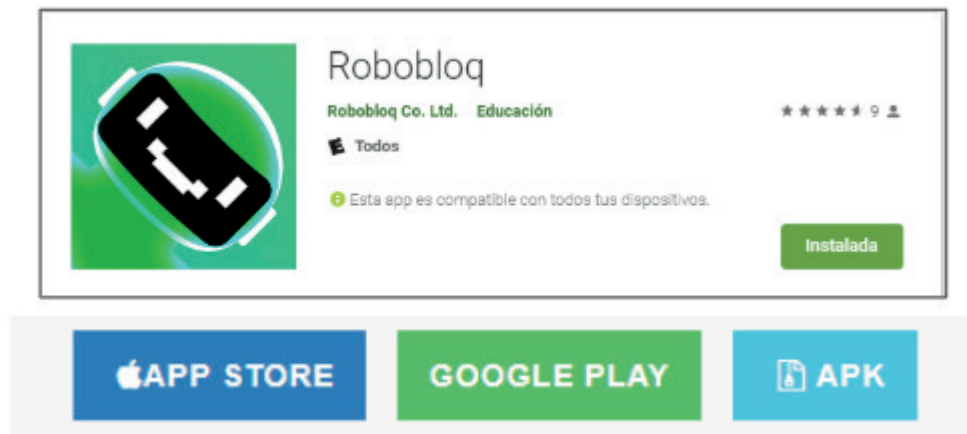
APP ROBOBLOQ

El término app es la abreviatura de la palabra inglesa *Application*. Una app, es una aplicación de *software* diseñada para ejecutarse en dispositivos móviles.

Emotibot puede ser controlado y programado a través de una *app* llamada *Robobloq*.

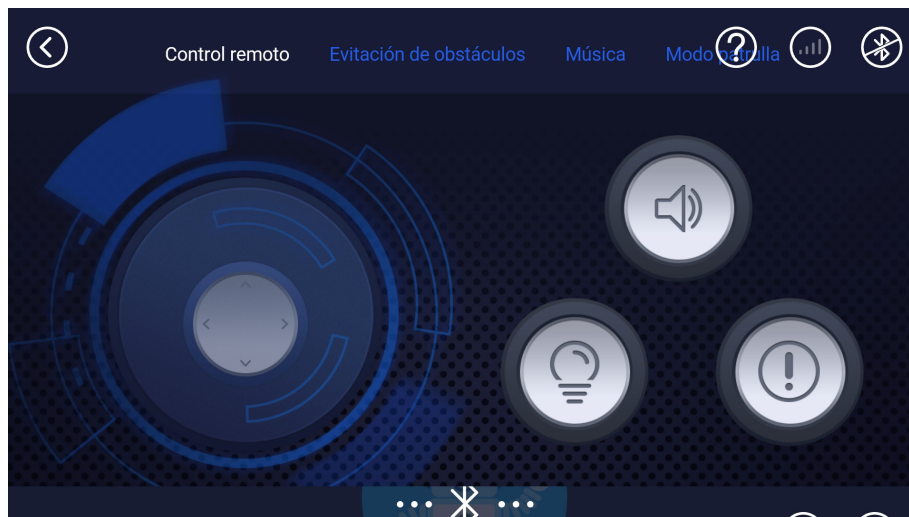
La app Robobloq es de sencilla descarga y está disponible para dispositivos Android compatible con sistemas operativos IOS 8.0+, ANDROID 64BIT PROCESSOR 5.0+.

APRENDER CONECTADOS

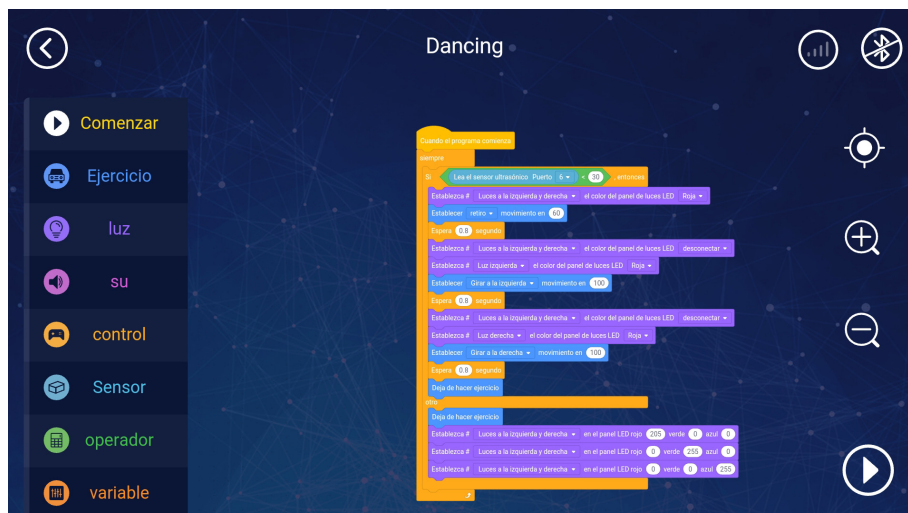


Esta *app* ofrece la posibilidad de comandar o programar 4 robots distintos bajo dos modalidades de uso:

- **MODO CONTROL:** a través de un tablero digital que tiene el aspecto de un *joystick* virtual es posible dirigir en simultáneo a cada acción del usuario, el comportamiento del robot.



- **MODO PROGRAMACIÓN:** permite generar una rutina de programación



APP ROBOBLOQ - MODO CONTROL

Para iniciar con el robot *Emotibot* en su modalidad control debemos seguir estos pasos:

PASO 1: Seleccionar a **Emotibot**.

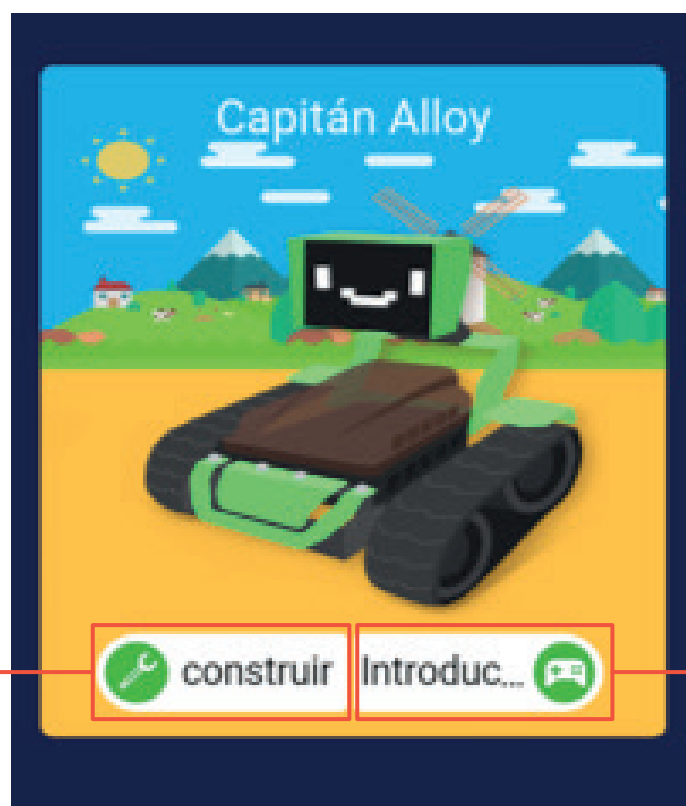


PASO 2: Accede a los modelos de **Emotibot** y sus respectivas guías de construcción e información.



APRENDER CONECTADOS

Ejemplo de opciones a desplegar
en cada modelo:



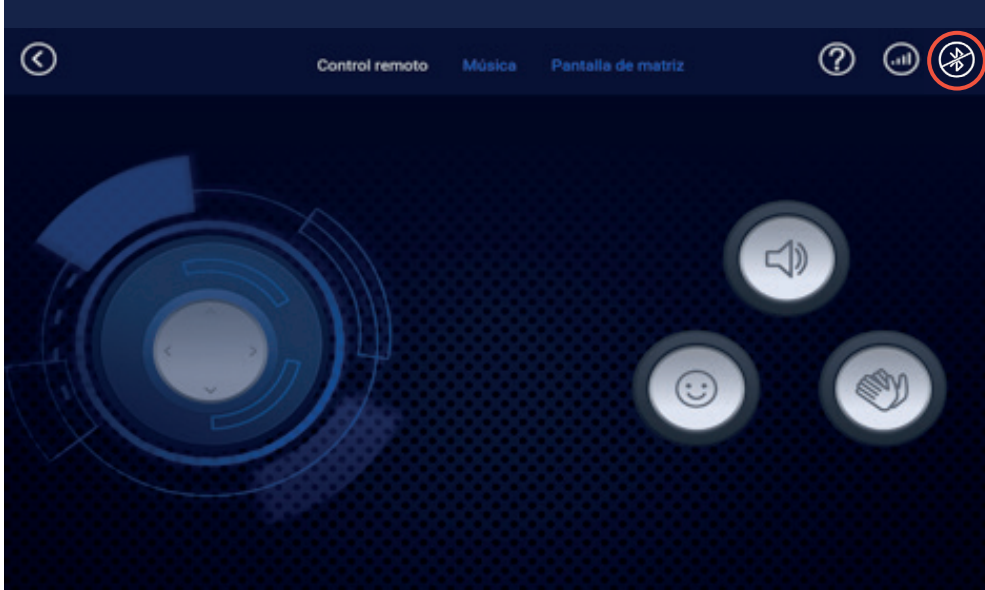
PASO 3:

Construir el robot según el instructivo y finalizar el armado del modelo.



PASO 4:

Vincular con **Emotibot** desde la tableta a través del botón de activación *Bluetooth*.



PASO 5:

Al presionar el botón, la pantalla enseña un cambio de estado en el proceso de vinculación.

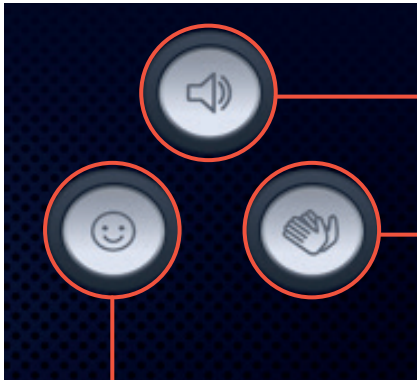


PASO 6:

Desde el tablero o mando de control podremos controlar al robot.



Botones de Control Remoto



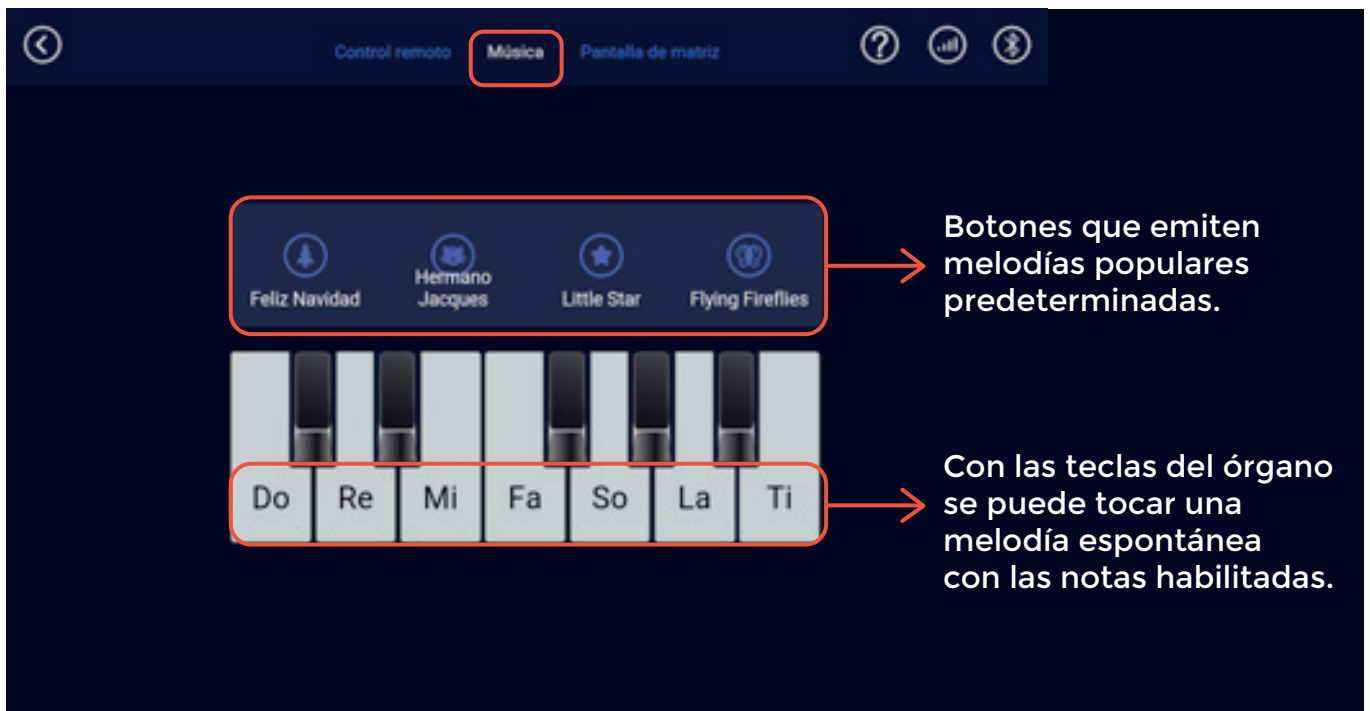
→ Apretando este botón, el robot emite un sonido tipo *beep*.

→ Presionando este botón, el robot gira hacia un lado y hacia el otro.

→ Apretando este botón, cambia el diseño de la matriz de *led*. Hay 8 modelos disponibles.

Pestaña música

Accediendo a esta opción es posible escuchar melodías populares o ingresar notas para generar música propia.

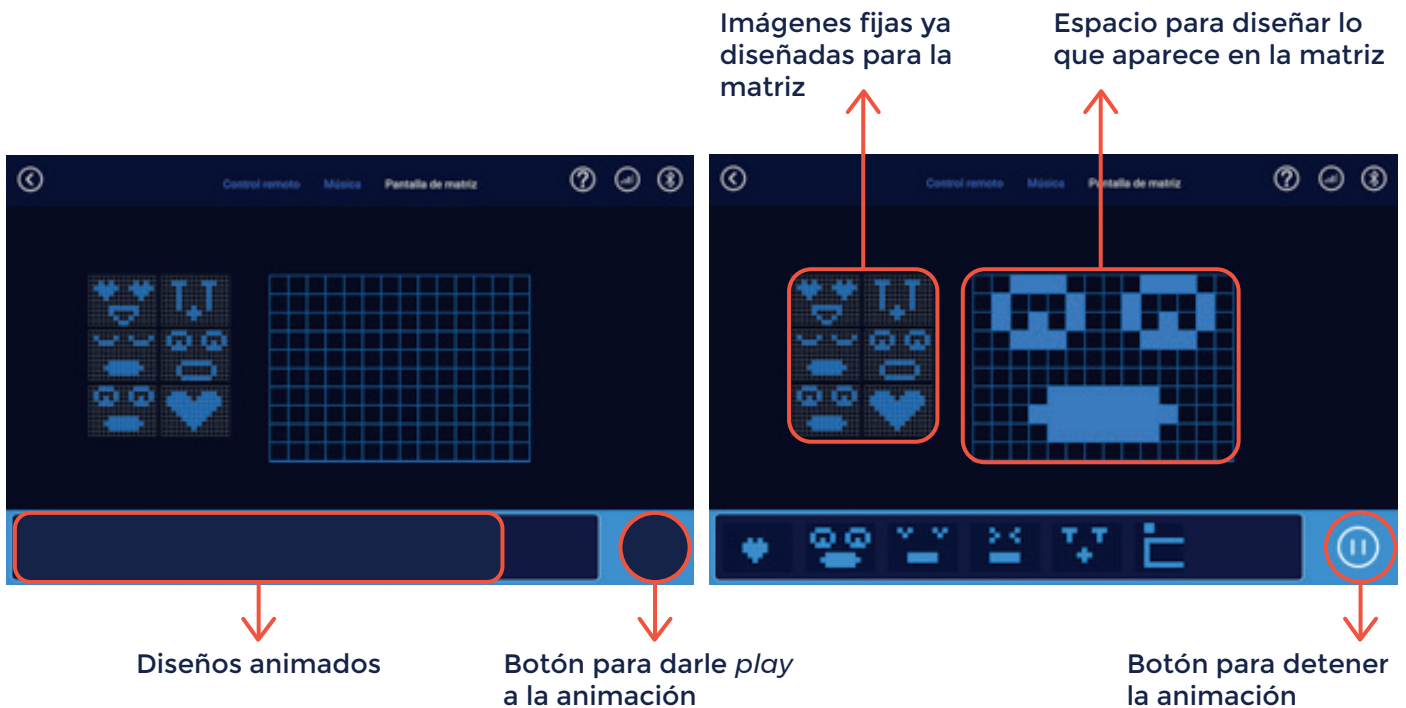


→ Botones que emiten melodías populares predeterminadas.

→ Con las teclas del órgano se puede tocar una melodía espontánea con las notas habilitadas.

Pestaña Pantalla de matriz

Permite seleccionar qué *leds* se verán encendidos y cuales se verán apagados dentro de la matriz de *leds*. También ofrece diseños preestablecidos y animaciones ya configuradas para transferir a **Emotibot**.



APP ROBOBLOQ - MODO PROGRAMACIÓN

Emotibot puede ser programado desde un *smartphone* o tableta, arrastrando los bloques de comandos a la ventana de trabajo. Es fácil e intuitivo identificar los distintos paneles porque están categorizados por color.

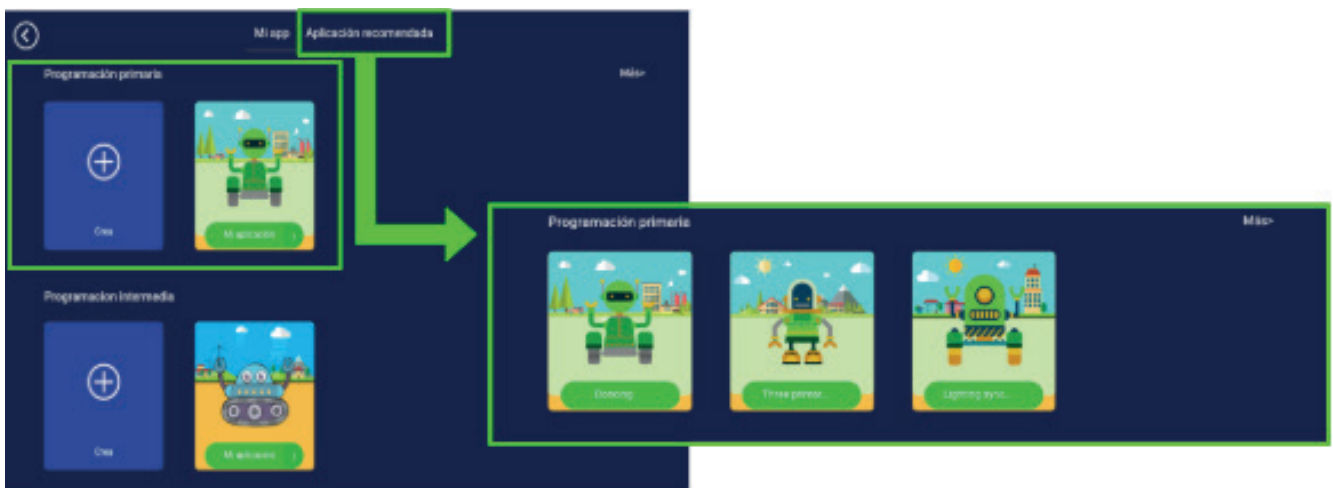
Dentro de la *app*, en la opción **Mi app**, es posible programar el robot bajo dos entornos de programación distintos:

- PRIMARIA: Cuenta con comandos icónicos únicamente.
- INTERMEDIA: construcción de código a través de bloques.



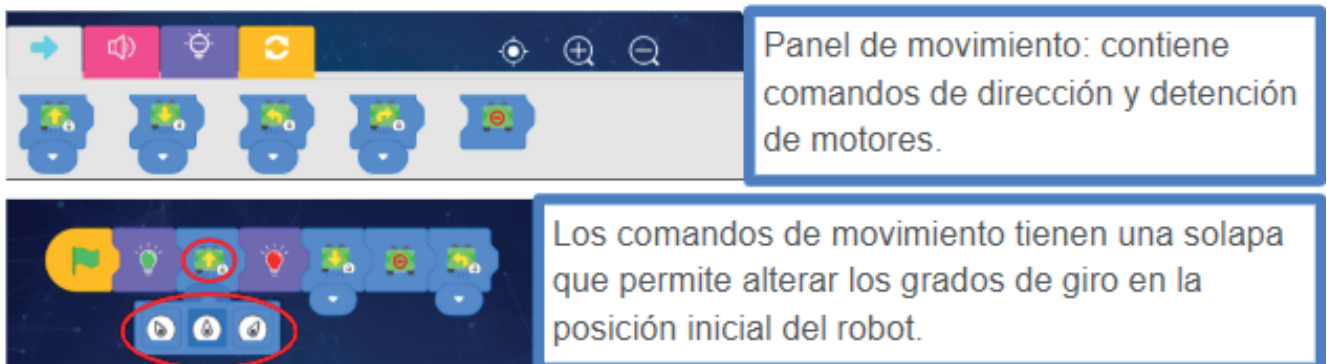
Mi app / Programación Primaria:

Esta modalidad cuenta con la opción **Crear** un proyecto de programación desde 0 y con la alternativa de utilizar algunos proyectos prediseñados con rutinas ya armadas que se pueden ejecutar o editar.





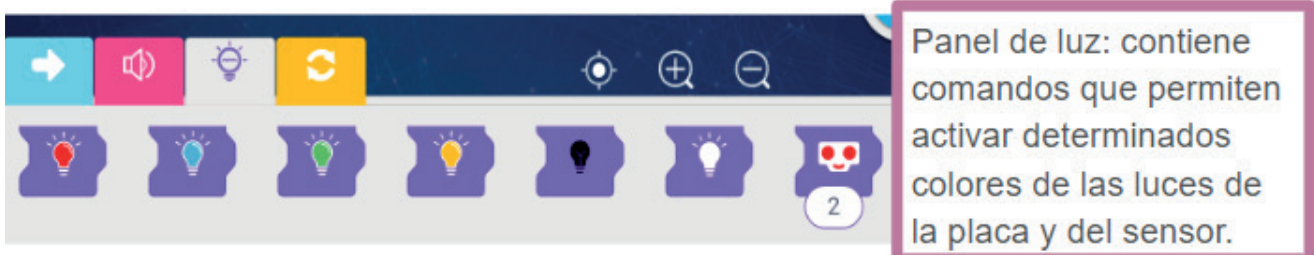
PANEL DE MOVIMIENTO



PANEL DE SONIDO



PANEL DE LUZ

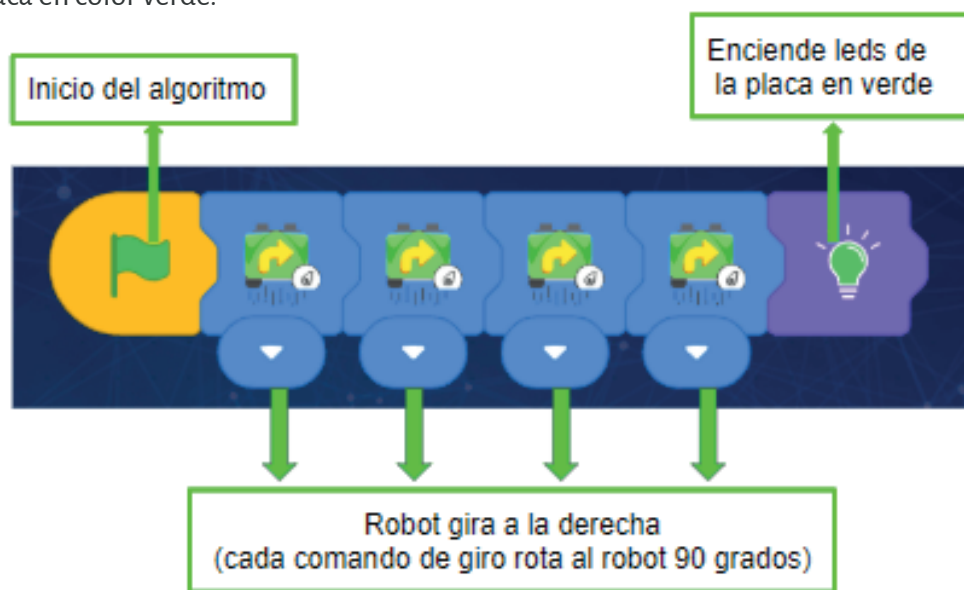


PANEL DE CONTROL



Ejemplo de actividad:

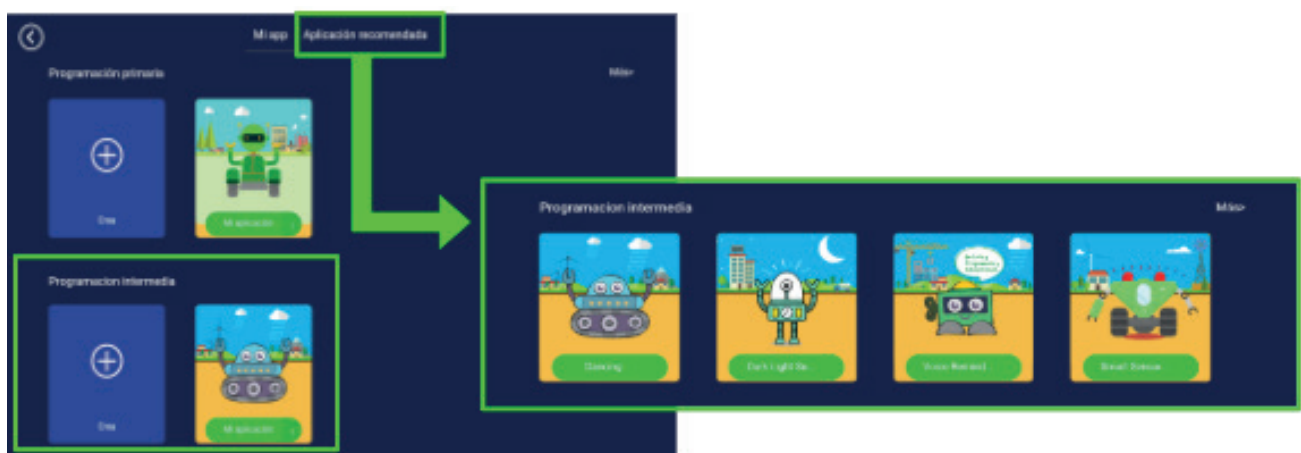
En la siguiente rutina de programación el robot gira 4 veces, generando un círculo. Al finalizar enciende las luces de la placa en color verde.



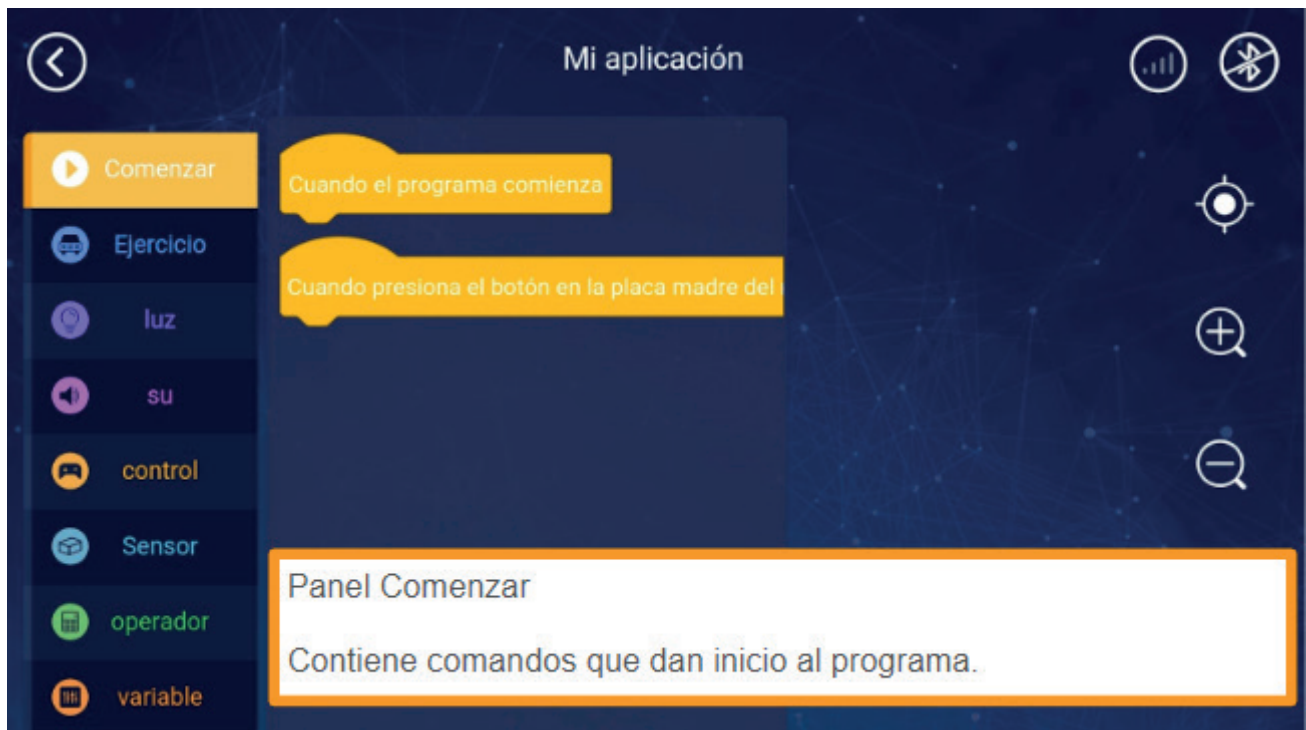
Mi app / Programación Intermedia:

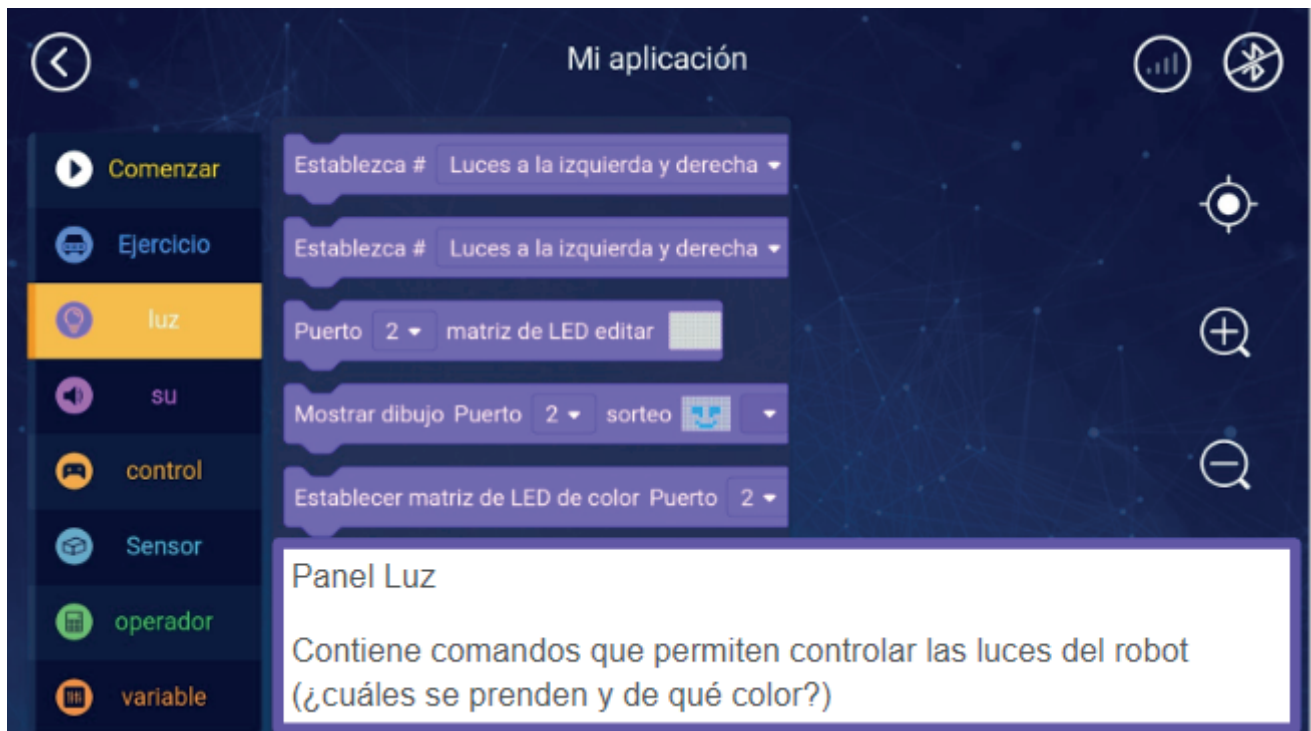
Entorno que cuenta con bloques que contienen texto y más comandos de instrucciones.

Esta modalidad presenta la opción de **Crear** (señalizada con el signo +) un proyecto de programación de principio a fin y una alternativa para utilizar algunos proyectos prediseñados con rutinas ya armadas que se pueden ejecutar o también editar.



Los paneles que incluyen instrucciones son:







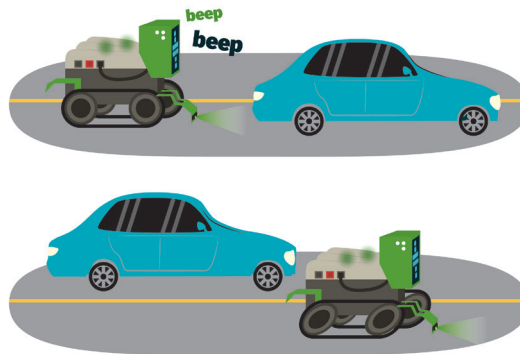


Ejemplo de actividad:

A continuación se presenta una rutina donde se utiliza un dato tomado a través del sensor de ultrasonido.



EMOTIBOT AMBULANCIA



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Básico de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

La creación, la reutilización, la reelaboración y la edición de contenidos digitales en diferentes formatos, entendiendo las características y los modos de representación de lo digital.

OBJETIVOS CURRICULARES ARTICULADOS

El reconocimiento, uso y análisis de funciones en situaciones problemáticas que requieran:

- Interpretar y producir fórmulas que modelicen variaciones lineales y no lineales en función de la situación;
- interpretar el significado de la media y el modo para describir los datos en estudio.

DESARROLLO

Utilizando los motores medir los tiempos del robot para llegar a destino en una emergencia ¿de qué manera se optimizan las acciones para ganar tiempo?. Pensando en situaciones en donde llegar rápido es lo importante, ¿cómo sería la rutina del robot?

Los/las estudiantes desarrollan un robot ambulancia. El mismo contará con:

- sensor de ultrasonido para medir distancias. Ante la presencia de un obstáculo, los motores reducen su velocidad y se toca bocina para que el obstáculo se corra.
- emisión permanente de sonido al estilo sirena de ambulancia.
- un diseño en la matriz de *leds* que tenga una cruz de emergencia.
- *leds* encendidos en placa y en sensor ultrasónico. Todos de color verde.

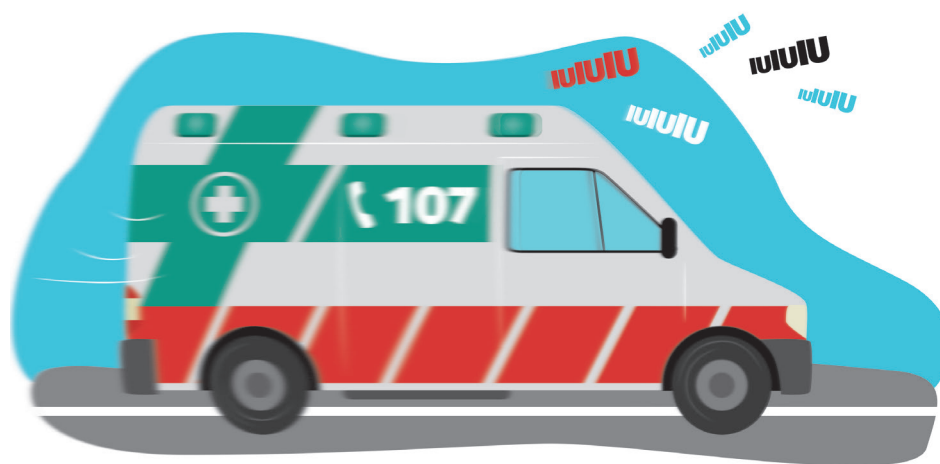
CIERRE, RECOMENDACIONES

Sugerimos programar diferentes sonidos como un camión de bomberos.

MOMENTO 1

Analicemos el comportamiento de una ambulancia en situación de emergencia:

- ¿Qué sonido emite la sirena?
- ¿En qué momento emite sonido?
- ¿De qué color es la luz de la sirena?
- ¿A qué velocidad puede ir una ambulancia?



MOMENTO 2

- Transferimos lo investigado al robot.
- Construimos el modelo LI'L GUARDIAN: (*)



(*) Ver anexo de construcción - p.96.

APRENDER CONECTADOS

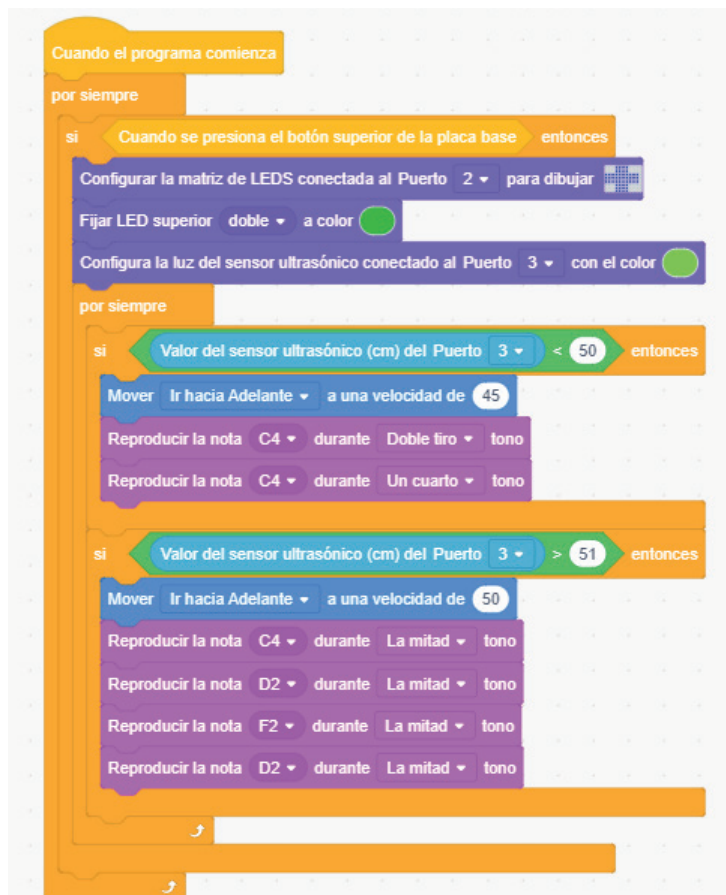
Una vez construido, le agregamos un sensor de ultrasonido en el paragolpe y lo conectamos para que detecte a su paso los obstáculos que estén frente a él.



- Construimos una pista recta con obstáculos (que simulan ser otros autos en movimiento).
- Programamos un robot ambulancia para que vaya sobre este trayecto. Para esto necesitamos:
 - un sensor de ultrasonido para medir distancias.
 - ante la presencia de un obstáculo, los motores reducen su velocidad y el robot toca bocina hasta que el obstáculo se corre.
 - emisión de sonido al estilo sirena de ambulancia.
 - un diseño en la matriz de *leds* que tenga una cruz de emergencia.
 - *leds* encendidos en placa y en sensor ultrasónico. Todos de color verde.

Posible rutina de programación (recordemos que es sólo un ejemplo, es probable que sean necesarias modificaciones en la velocidad que precisan los motores y en la distancia de detección):

MOMENTO 3



En esta rutina estamos programando a **Emotibot** para que:

- Cuando el robot esté encendido con el programa ya guardado en placa,
- ejecuta por siempre que:
- solo si presionamos el botón de la placa,
- la matriz de *led* conectada al puerto 2 dibuje una cruz,
- ambos *leds* de la placa enciendan en verde,
- la luz del sensor ultrasónico encienda en verde,
- por siempre consultá:
- si el valor detectado por el sensor ultrasónico conectado al puerto 3, es menor a 50 cm, entonces:
- el robot avanza hacia adelante a una velocidad de 45 rpm,
- reproduce la nota C4 durante un doble pulso,
- reproduce la nota C4 durante un cuarto de pulso,

- si el valor detectado por el sensor ultrasónico conectado al puerto 3, es mayor a 51 cm, entonces:
- el robot avanza hacia adelante a una velocidad de 50 rpm,
- reproduce la nota C4 durante la mitad de pulso,
- reproduce la nota D2 durante la mitad de pulso,
- reproduce la nota F2 durante la mitad de pulso,
- reproduce la nota D2 durante la mitad de pulso

CIERRE, RECOMENDACIONES

Te sugerimos programar diferentes sonidos que emite un camión de bomberos.

Sonido en emergencia

Existe un código por la cantidad de toques de sirena en un camión

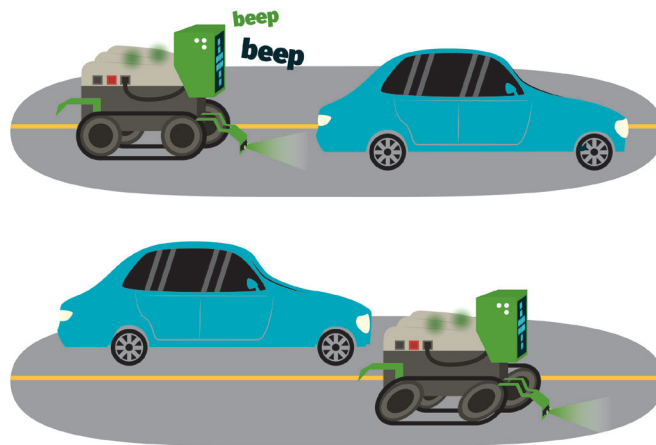
- un toque de **sirena** es festejo,
- tres toques es incendio afuera,
- cuatro es incendio local,
- cinco es accidente o salvataje,
- seis, ahogado.

Audio: Declaraciones de Gustavo Nicola para Radio Líder Argentina

<https://www.villamariaya.com/aprenda-no-se-asuste-significa-cuando-suenan-la-sirena-bomberos/>



EMOTIBOT AMBULANCIA



¡Emotibot al rescate!

Para comenzar:

Construí este modelo de robot:



- Anexale al frente un sensor ultrasónico.
- Planteá un área por la que se va a mover el robot con algunos obstáculos en el camino.
- Pensá una secuencia de programación en la suceda lo siguiente:
 - en la matriz de *led* esté dibujada una cruz de emergencia,
 - las luces en placa estén encendidas de color verde,
 - las luces del sensor ultrasónico sean de color verde,

- el sensor ultrasónico detecte obstáculos a una distancia menor de 50 cm,
- ante la detección de un obstáculo, el robot disminuye su velocidad y toca bocina,
- si no hay obstáculos en su camino, el robot avanza a mayor velocidad con la sirena encendida.

Agregá el código:

Posible rutina de programación (es sólo un ejemplo, es probable que sean necesarias modificaciones en la velocidad que precisan los motores y en la distancia de detección):



EMOTIBOT MEDIDOR DE SONIDO AMBIENTE



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Básico de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

La resolución de problemas a partir de su descomposición en partes pequeñas, aplicando diferentes estrategias, utilizando entornos de programación tanto textuales como icónicos, con distintos propósitos, incluyendo el control, la automatización y la simulación de sistemas físicos.

OBJETIVOS CURRICULARES ARTICULADOS

El uso de relaciones entre variables en situaciones problemáticas que requieran:

- modelizar variaciones uniformes y expresarlas eligiendo la representación más adecuada a la situación;
- desarrollar una rutina de programación que permita verificar la presencia de contaminación sonora.
- reflexionar y crear acciones de prevención y concientización.

DESARROLLO

La contaminación sonora es una problemática mundial que altera las condiciones normales del ambiente.

Los/as estudiantes son pensadores críticos, identifican, promueven acciones preventivas y crean soluciones a situaciones reales de la vida diaria.

Los/as estudiantes generan una rutina de programación en un robot que mide la contaminación sonora a través del sensor de sonido y nos concientiza en torno a ella. Programar estos datos para que sean expresados en la matriz de *led*, prendiéndose la mitad si la cantidad de sonido ambiente es tolerable y encendiéndose totalmente si es intolerable. También las luces en placa se pondrán de color verde cuando el sonido sea bajo, y de color rojo si fuera alto. En simultáneo cuando esto sucede, el robot retrocederá huyendo de la contaminación sonora.

CIERRE, RECOMENDACIONES

Los/las estudiantes alterarán la rutina y le darán emociones a la matriz de *led*, mostrando una cara feliz o asustada según cada caso.

MOMENTO 1

La docente organiza la clase en grupos y realiza una lluvia de ideas sobre la temática “contaminación”, estrategia sugerida para conocer los conocimientos verdaderos o erróneos de los estudiantes. Puede además presentar algunos videos como disparador, por ejemplo las conversaciones que promueve la organización TED en diferentes lugares del mundo:

- Greta Thunberg, Noviembre 2018, El argumento cautivador para actuar ya ante el cambio climático de Greta Thunberg

https://www.ted.com/talks/greta_thunberg_the_disarming_case_to_act_right_now_on_climate

- Mathias Basner, 2018, Por qué el ruido es malo para tu salud y qué puedes hacer al respecto de Mathias Basner

https://www.ted.com/talks/mathias_basner_why_noise_is_bad_for_your_health_and_what_you_can_do_about_it

Proponemos a cada grupo que cree una definición frente a la pregunta
¿Qué es la contaminación?

“La contaminación es encontrar en el ambiente la presencia de sustancias que afectan negativamente las condiciones de vida de los seres vivos...”

¿Qué sabemos de cada tipo de contaminación existente? Reunidos escribimos lo que sabemos al respecto:

Contaminación atmosférica



Contaminación acuática



(3) Recuperado de <https://pxhere.com/es/photo/1549165>

(4) Recuperado de <https://pixnio.com/es/diverso/basura/el-agua-la-contaminacion-los-peces-matar>

Contaminación del suelo



(5)

Contaminación acústica



(6)

Contaminación radioactiva



(7)

Contaminación térmica



(8)

MOMENTO 2

Vemos un video de Jonathan Rossiter, 2016 “Un robot que se come la contaminación”

https://www.ted.com/talks/jonathan_rossiter_a_robot_that_eats_pollution?language=es#t-22804

- ¿Identificaron la problemática que se plantea en el video? ¿Cuál es?
- En la presentación, ¿se propone alguna solución para el problema planteado? ¿Cuál es y qué características tiene?
- ¿Podrían los robots alertarnos sobre la contaminación?
- ¿De qué manera un robot colaboraría con los ruidos en la escuela?

(5) Recuperado de <https://pixabay.com/es/photos/relleno-sanitario-papelera-volcado-1396596/>

(6) Recuperado de https://es.m.wikipedia.org/wiki/Archivo:Qantas_b747_over_houses_arp.jpg

(7) Recuperado de <https://pxhere.com/es/photo/873036>

(8) Recuperado de <https://pixabay.com/es/photos/planta-de-energ%C3%ADa-geot%C3%A9rmica-67538/>

- ¿Se podría solucionar alguna problemática relacionada con la contaminación sonora utilizando un programa informático? Si consideras que si es posible ¿para realizar que acción?

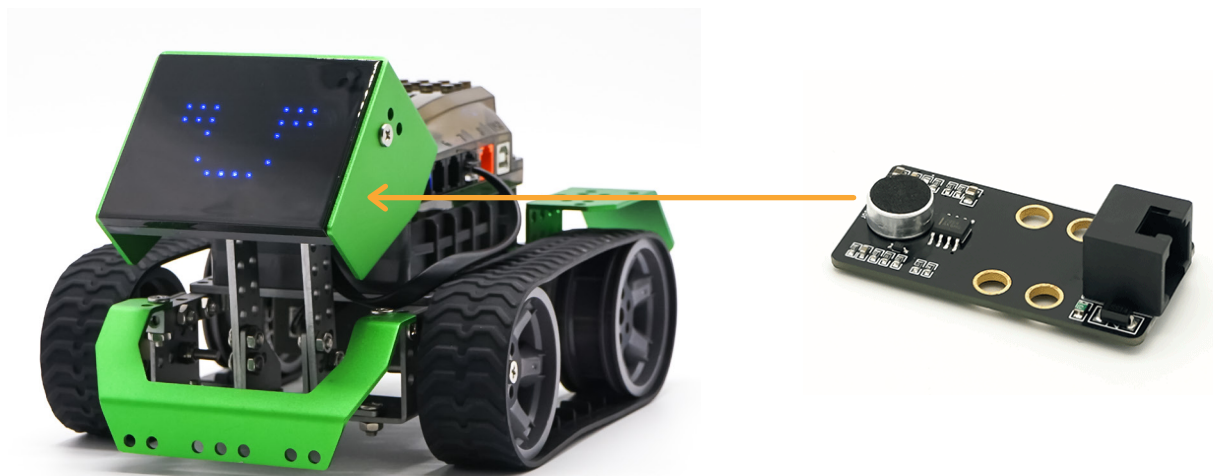
Frente al cuestionario nos escuchamos y luego con **EMOTIBOT** (modelo **LI'L GUARDIAN(*)**) y el entorno de programación, haremos el desarrollo de acciones tendientes a concientizar sobre la temática “contaminación ambiental”.

MOMENTO 3

Unos de los factores principales para la concientización de la contaminación es la prevención. A menudo cuando visualizamos conductas o determinadas realidades, es cuando tomamos conciencia de lo importante que es corregirlas. Usamos muchas veces la tecnología para evidenciar situaciones problemáticas y también para solucionarlas.

¡Utilicemos a **Emotibot** para que nos reporte el grado de contaminación sonora existente en la escuela!

Al siguiente modelo le anexamos el sensor de sonido para lograr medir los datos del ambiente. Podemos dejarlo fijo donde queramos, lo que no nos podemos olvidar es de conectarlo a algún puerto de la placa.

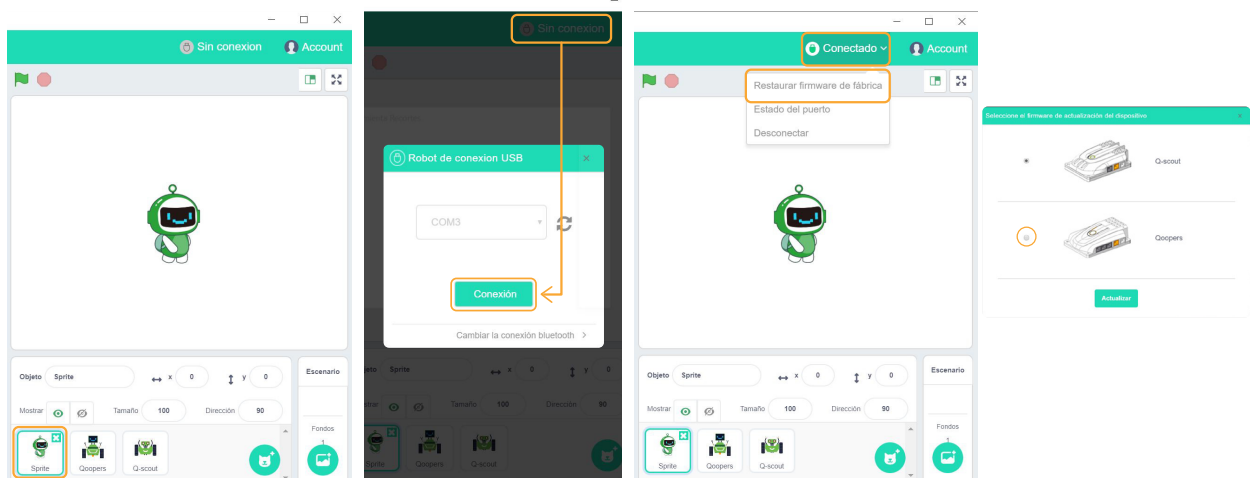


(*) Ver anexo de construcción - p. 96.

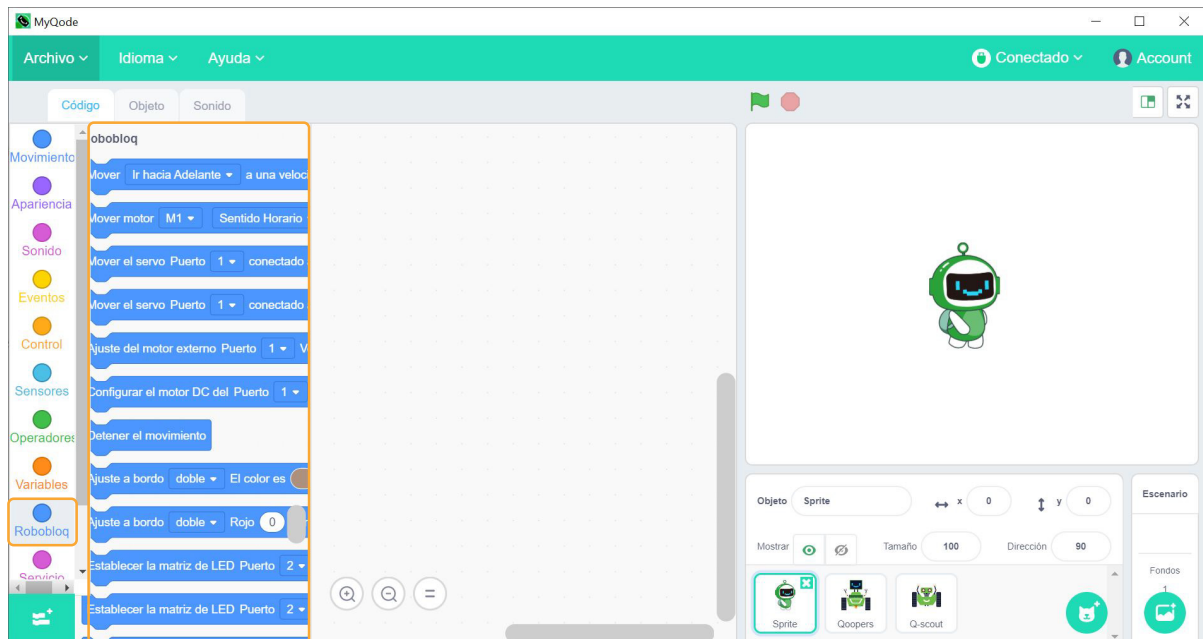
APRENDER CONECTADOS

Utilizaremos los datos que nos arroja el sensor de sonido para programar acciones en la matriz de *led*, prendiéndose la mitad si la cantidad de sonido ambiente es tolerable y encendiéndose totalmente si es intolerable. También las luces en placa se pondrán de color verde cuando el sonido sea bajo, y de color rojo cuando sea alto. En simultáneo cuando esto sucede, el robot retrocederá huyendo de la contaminación sonora.

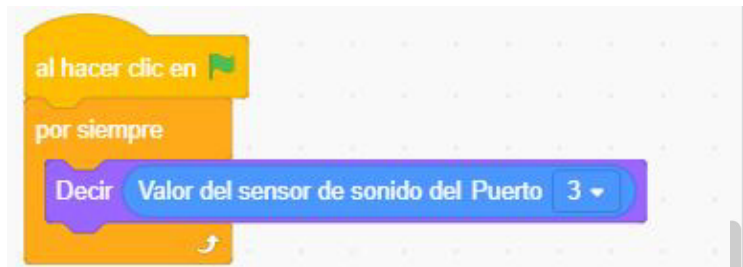
Para evidenciar el dato más alto que detecta el sensor, debemos leer su estado exponiéndolo a un sonido fuerte. Para leer los datos que ingresan al sensor, realizar estos pasos:



- 1- seleccionamos el objeto *Sprite*.
- 2- conectamos el robot al puerto *USB*.
- 3- hacemos clic en “Sin conexión”.
- 4- se abre una ventana emergente que detecta el puerto conectado. Presionamos el botón “Conexión”.
- 5- cuando el estado pasa de “Sin conexión” a “Conectado”, desplegamos la solapa de acciones y seleccionamos “Restaurar *firmware* de fábrica”.
- 6- nos aparece una ventana emergente que nos da a elegir la placa del robot. En el caso de **Emotibot** tiene la placa identificada con el nombre *Qoopers*.
- 7- automáticamente se suma a los paneles de comandos, uno nuevo que se llama “*Robobloq*” que nos permite ingresar instrucciones para el robot.



8- Generamos la siguiente rutina. Ella nos permite visualizar los datos que ingresa el sensor en el globo de diálogo de *Sprite*.

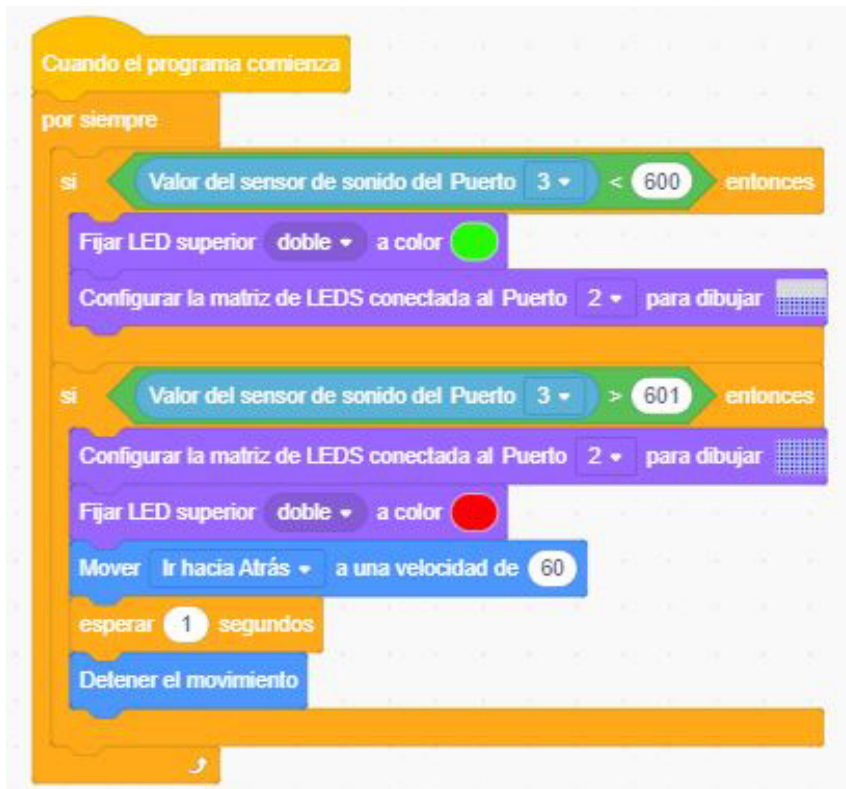


Para entender los parámetros en los que se mueve el sensor, debemos hacer silencio y luego gritar cerca de él o aplaudir. Así veremos el máximo silencio que registra y el máximo ruido y estableceremos el parámetro de detección. Por ejemplo, que ante el mayor silencio de ambiente posible, el sensor arroja 250 y ante el mayor ruido 700. Ahí determinamos entonces que la media es 500 y podemos indicar en nuestra programación que a partir de 500, inicie el estado referente a la detección de contaminación sonora.

El parámetro varía según el espacio y la cantidad de ruido que haya en él. El sensor de sonido puede medir de 0 a 1023 valores referentes a decibeles (unidad de medida de intensidad sonora).

Ejemplo de rutina:

APRENDER CONECTADOS



- Cuando el robot esté encendido con el programa ya guardado en placa,
- por siempre consultá:
- si el valor detectado por el sensor de sonido conectado al puerto 3, es menor a 600, entonces:
- ambas luces en placa se encienden de color verde,
- la matriz de *leds* conectada al puerto 2, dibuja _____ (medía matriz pintada),
- si el valor detectado por el sensor de sonido conectado al puerto 3, es mayor a 601, entonces:
- la matriz de *leds* conectada al puerto 2, dibuja _____ (toda la matriz pintada),
- ambas luces en placa se encienden de color rojo,
- el robot va hacia atrás a una velocidad de 60 rpm,
- durante 1 segundo
- detiene su movimiento

CIERRE, RECOMENDACIONES

Alteremos la rutina para darle emociones a la matriz de *led*, mostrando una cara feliz o asustada según cada caso.

EMOTIBOT MEDIDOR DE SONIDO AMBIENTE



¡Prevenimos la contaminación sonora con Emotibot!

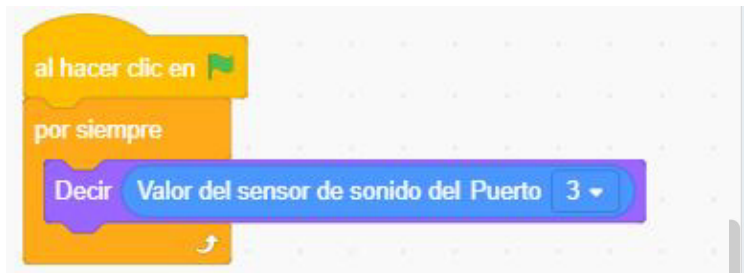
Para comenzar:

1- Agregá el sensor de sonido a este modelo para medir los datos del ambiente:



2- Creá la siguiente rutina para leer en tiempo real los datos que toma el sensor:

APRENDER CONECTADOS



- 3- Establece el parámetro de detección del sensor: el mínimo y el máximo que registra. Determina el valor medio a partir del cual, hay contaminación sonora.
- 4- El desafío es programar utilizando los datos que arroja el sensor de sonido para que la matriz led encienda de una forma si la cantidad de sonido ambiente es tolerable y de otra cuando es intolerable. También las luces en placa se pondrán de color verde cuando el sonido sea bajo, y de color rojo cuando sea alto. En simultáneo cuando esto sucede, el robot retrocederá huyendo de la contaminación sonora.
- 5- Agregá el código:
- 6- Adaptá este ejemplo de rutina de programación al parámetro que detecta el sensor de sonido en el ambiente donde estás.

<p>Cuando el programa comienza</p> <p>por siempre</p> <p>si <i>Valor del sensor de sonido del Puerto 3 < 600</i> entonces</p> <p>Fijar LED superior <i>doble</i> a color ●</p> <p>Configurar la matriz de LEDS conectada al Puerto 2 <i>para dibujar</i></p> <p>si <i>Valor del sensor de sonido del Puerto 3 > 601</i> entonces</p> <p>Configurar la matriz de LEDS conectada al Puerto 2 <i>para dibujar</i></p> <p>Fijar LED superior <i>doble</i> a color ●</p> <p>Mover <i>Ir hacia Atrás</i> a una velocidad de <i>60</i></p> <p>esperar <i>1</i> segundos</p> <p>Detener el movimiento</p>	<p><i>Cuando el robot esté encendido con el programa ya guardado en placa,</i></p> <p><i>por siempre consultá:</i></p> <p><i>si el valor detectado por el sensor de sonido conectado al puerto 3, es menor a 600, entonces:</i></p> <p><i>ambas luces en placa se encienden de color verde,</i></p> <p><i>la matriz de leds conectada al puerto 2, dibuja _____ (media matriz pintada),</i></p> <p><i>si el valor detectado por el sensor de sonido conectado al puerto 3, es mayor a 601, entonces:</i></p> <p><i>la matriz de leds conectada al puerto 2, dibuja _____ (toda la matriz pintada),</i></p> <p><i>ambas luces en placa se encienden de color rojo,</i></p> <p><i>el robot va hacia atrás a una velocidad de 60 rpm,</i></p> <p><i>durante 1 segundo</i></p> <p><i>detiene su movimiento</i></p>
--	--

Te proponemos ahora a programar emociones en la matriz de *led*, mostrando una cara feliz o asustada según el caso?

GIROS CON EMOTIBOT



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Básico de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

La comprensión general del funcionamiento de los componentes de *hardware* y *software*, y la forma en que se comunican entre ellos y con otros sistemas, entendiendo los principios básicos de la digitalización de la información y su aplicación en la vida cotidiana.

OBJETIVOS CURRICULARES ARTICULADOS

El análisis y construcción de figuras, argumentando sobre la base de propiedades, en situaciones problemáticas que requieran:

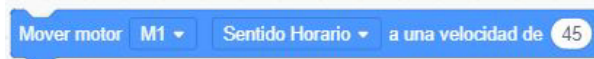
- determinar puntos que cumplan condiciones referidas a distancias y construir circunferencias, círculos, mediatrices y bisectrices como lugares geométricos;
- desarrollar una rutina de programación para que el robot gire, conociendo el movimiento de los motores.

DESARROLLO

En grupos los/las estudiantes piensan cuántos y cuáles objetos conocen que realizan giros e intentan explicarlos. Por ejemplo, ¿el giro es hacia la derecha o hacia la izquierda?, ¿para qué deben girar?

Observan a su alrededor objetos que giran. Los pueden encontrar muy visibles o no tanto. Analizando mecanismos, van comprendiendo que acciones de giros se pueden optimizar y también transformar.

Estudiamos distintos mecanismos de giro y reflexionamos: ¿Cómo lograr que el robot avance en línea recta y que gire utilizando el siguiente comando?:



Los/las estudiantes programan cada motor para dominar el comportamiento del robot. Exploran tiempos y velocidades a aplicar. Analizan todas las variables posibles para generar distintas alternativas de giros.

**CIERRE,
RECOMENDACIONES**

Los/las estudiantes cambian las orugas del robot por cubiertas individuales para las ruedas. Generan y exploran cómo varió el comportamiento del robot luego del cambio. Prueban los 3 casos dados y lo vuelcan a una nueva tabla.

¿Hubo cambios? ¿cuáles?

¿A qué pensás que se debe esta diferencia entre orugas y ruedas individuales?

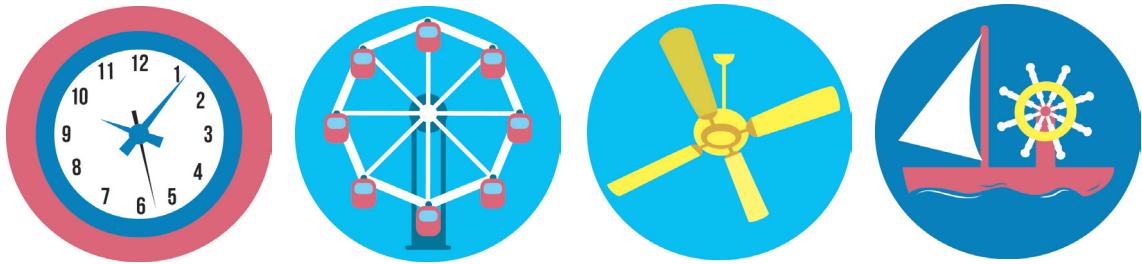
¿Te animás a generar 2 casos nuevos?

MOMENTO 1

Podemos realizar un recorrido por la escuela o el barrio en búsqueda de imágenes de objetos que giran.

Es importante orientar la búsqueda a mecanismos(*) con giros menos evidentes que una rueda, por ejemplo grifos o volantes de autos.

En las siguientes ilustraciones, encontramos objetos que giran de diversa forma.



- ¿Cómo giran?
- ¿Identificaron dónde está el eje de rotación en cada caso?
- ¿Hay algún caso entre los citados, en el que el mecanismo de giro se transforme en otro mecanismo de acción? Si creemos que sí, especifiquemos en qué caso y qué es lo que sucede allí.

MOMENTO 2

Un poco de historia.

Aquí analizamos el proceso de optimización del mecanismo de la bicicleta. El objetivo se centró en realizar mejores giros con menor esfuerzo del usuario.

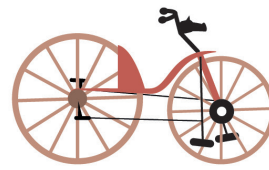
(*) **Mecanismo:** Conjunto de las partes de una máquina(**) en su disposición adecuada para asegurar su funcionamiento efectivo.

(**) **Máquina:** Conjunto de aparatos combinados para recibir cierta forma de energía y transformarla en otra más adecuada, o para producir un efecto determinado.

La cadena de la bicicleta es una banda de transmisión de movimiento que hace girar la rueda trasera muy rápido debido a que está conectada a una rueda pequeña que, al girar, multiplica la fuerza aplicada a los pedales. Las primeras bicicletas no contaban con cadenas ni pedales y eran mucho más lentas y pesadas. Observamos cómo evolucionó la bicicleta desde los primeros modelos hasta las más modernas



1817



1839



1879



1960



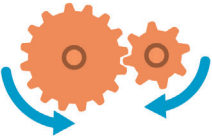
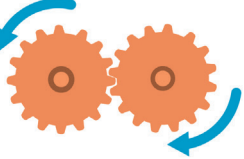
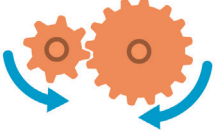
1992

- En la bicicleta de 1817, ¿cómo andaría el usuario sin pedales?
- ¿El modelo de 1839 es óptimo? ¿por qué?, ¿cuál sería la diferencia con las siguientes? Pensemos en el movimiento y la fuerza que debe aplicar el usuario para hacerla andar y analicemos el por qué del cambio de posición de los pedales.
- En la de 1879, ¿cómo pensás que influyó la diferencia de tamaño de las ruedas en cuanto a la facilidad para manejarla?, ¿por qué?
- Comparemos la bicicleta de 1960 con la de 1992 (modelo actual) y analicemos:
 - ¿qué sucede cuando aplicamos la fuerza para generar el movimiento de los pedales en la bicicleta de 1960? ¿Cuánto espacio nos trasladamos: poco o mucho?

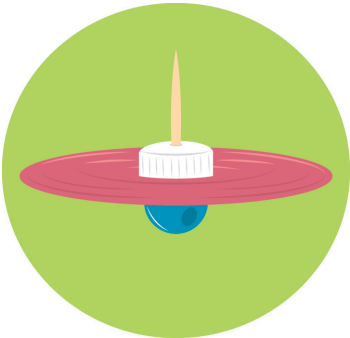
- Supongamos que el diámetro de la rueda es equivalente a 35 cm, ¿cuál sería el perímetro? Sacando el perímetro de la rueda, sabremos los cm que recorre la misma en cada vuelta.
- Supongamos que el diámetro de la rueda de la bicicleta de 1992 es de 65 cm, ¿cuánto recorrido haríamos en cada vuelta de rueda?

• Hoy en día, cada bicicleta está descripta con un número de rodado. ¿A qué refiere ese número? ¡A investigar!

Vinculamos el mecanismo de la bicicleta y todos los casos vistos con los siguientes ejemplos de acción en engranajes:

	<p>En este ejemplo, si la fuerza es aplicada sobre el engranaje grande:</p> <ul style="list-style-type: none"> • ¿el chico daría más o menos vueltas que el grande? <p>Considerando que el resultado de la acción es el comportamiento del engranaje chico:</p> <ul style="list-style-type: none"> • ¿estamos hablando de un mecanismo que reduce o multiplica las vueltas?
	<p>En este ejemplo, ambos engranajes son iguales.</p> <ul style="list-style-type: none"> • ¿varía el resultado dependiendo en cual aplique la fuerza? • ¿qué sucede en este caso?
	<p>En este ejemplo, si la fuerza es aplicada sobre el engranaje chico:</p> <ul style="list-style-type: none"> • ¿el grande daría más o menos vueltas que el chico? <p>Considerando que el resultado de la acción es el comportamiento del engranaje grande:</p> <ul style="list-style-type: none"> • ¿estamos hablando de un mecanismo que reduce o multiplica las vueltas?

Construimos unos juguetes simples que usan giros para moverse. Haremos trompos rápidos creados con materiales descartables.

<p>Materiales:</p>	<ul style="list-style-type: none"> • 1 palito de brochette o escarbadiantes. • 1 CD. • 1 tapa de refresco. • Pegamento. • 1 canica.
	<ul style="list-style-type: none"> • ¿Cómo se genera la acción en este caso? • ¿Hay algún elemento en particular que permite que el impulso de giro aplicado se sostenga en el tiempo y permita que el trompo siga girando?, ¿cuál es ese elemento? • ¿Creés que hay una compensación de peso aplicada al diseño del trompo para que funcione como tal? ¿Qué pasaría si en vez de usar el centro del CD, el escarbadiante, tapita y canica en el centro, estuviesen a un costado?, ¿funcionaría?, ¿por qué? • ¿Podemos alterar el objeto para que sostenga el giro por más tiempo?, ¿cómo lo haríamos?

MOMENTO 3

Programamos a **Emotibot** (modelo **Cavalier(*)**) para que pueda realizar giros tanto en diferentes direcciones como con diversa velocidad.

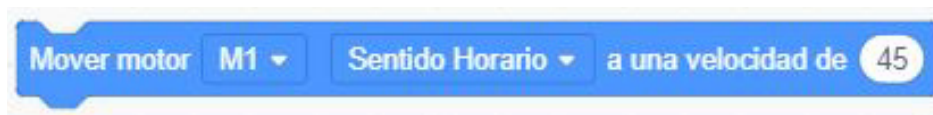


(*) Ver anexo de construcción - p.105.

APRENDER CONECTADOS

Al programar a **Emotibot** para que avance o gire, utilizamos un parámetro de medición que tiene que ver con la cantidad de vueltas que realiza por minuto. Esto se traduce en revoluciones por minuto (o rpm). A través de la medición de la cantidad de vueltas que el motor da por minuto, se puede determinar su velocidad. La velocidad de cada motor puede variar entre 0 y 255 rpm, donde 0 significa la mínima de velocidad aplicada (ya sin movimiento) y 255, la máxima.

El robot presenta 2 motores. Para lograr el giro de todo el objeto, ambos deben comportarse distinto. Usemos este comando para determinar las especificaciones de movimiento de cada motor.



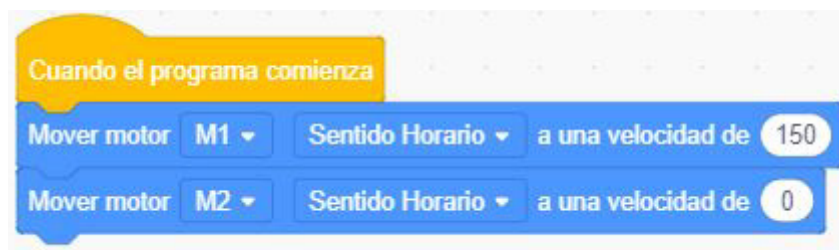
Programemos las siguientes pruebas:

Caso 1



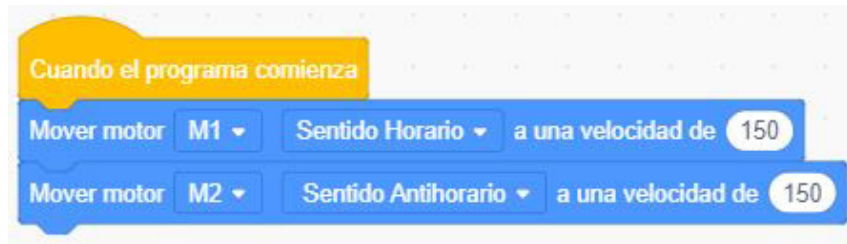
Motor 1 va en sentido de las agujas del reloj a una velocidad de 200 rpm mientras que motor 2 va en la misma dirección a una velocidad de 50 rpm. ¿Qué sucede?

Caso 2



Motor 1 va a 150 rpm mientras motor 2 está en 0 rpm. ¿Qué sucede?

Caso 3



Motor 1 va en 150 rpm en sentido de las agujas del reloj y motor 2 va en 150 en sentido contrario a las agujas del reloj.
¿Qué sucede?

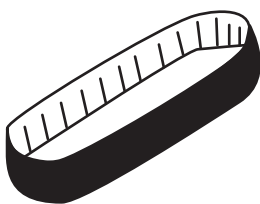
Luego de hacer las 3 pruebas, ¿en qué caso el giro fue el más efectivo?, ¿por qué?

Según cada caso, los estudiantes generan una tabla con los distintos comportamientos del robot:

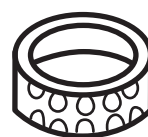
	Dirección de rotación Motor 1	Dirección de rotación Motor 2	Velocidad de rotación Motor 1	Velocidad de rotación Motor 2	Robot gira ¿hacia la derecha o hacia la izquierda?	Robot llega a cubrir un giro de 180° en ¿cuántos segundos?
Caso 1						
Caso 2						
Caso 3						

CIERRE, RECOMENDACIONES

Cambiamos las orugas del robot por cubiertas individuales para las ruedas:



Oruga



Cubierta individual para una rueda

Exploramos cómo varió el comportamiento del robot luego del cambio.

- ¿Hubo cambios? ¿cuáles? ¿por qué?
- Pensá dos casos nuevos para generar otros comportamientos y analízalos.

Probamos los 3 casos dados en el MOMENTO 3 y en grupos, armamos nuevas tablas.

Según cada caso, los estudiantes generan una tabla con los distintos comportamientos del robot:

	Dirección de rotación Motor 1	Dirección de rotación Motor 2	Velocidad de rotación Motor 1	Velocidad de rotación Motor 2	Emotibot gira ¿hacia la derecha o hacia la izquierda?	Emotibot llega a cubrir un giro de 180° en ¿cuántos segundos?
Caso 1						
Caso 2						
Caso 3						
Caso 4						
Caso 5						
Caso 6						

Hacemos ahora, una rutina de programación calculando exactamente cuánto tiempo y a qué velocidad debe avanzar el robot para cumplir el siguiente desafío:

Emotibot parte desde un punto designado y señalado en el piso. Debe llegar a la pared y en lugar de chocar con ella, girar hacia la izquierda.

¡Sin sensores! ¡Sólo con cálculos matemáticos **Emotibot** puede lograrlo!

GIROS CON EMOTIBOT



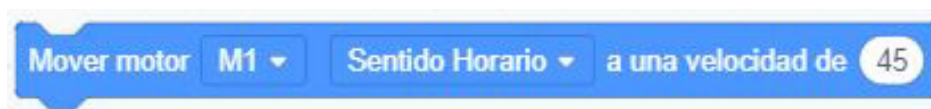
¡Mareando a Emotibot!

Para comenzar:

Construí a **Emotibot** (modelo *Cavalier*) para que pueda realizar giros tanto en diferentes direcciones como con diversa velocidad.



El robot presenta 2 motores. Para lograr el giro de todo el objeto, ambos deben comportarse distinto. Usemos este comando para determinar las especificaciones de movimiento de cada motor.



Programá las siguientes pruebas:

Caso 1:

Motor 1 una va en sentido de las agujas del reloj a una velocidad de 200 rpm mientras que motor 2 va en la misma dirección a una velocidad de 50 rpm.

Caso 2:

Motor 1 va a 150 rpm mientras motor 2 está en 0 rpm.

Caso 3:

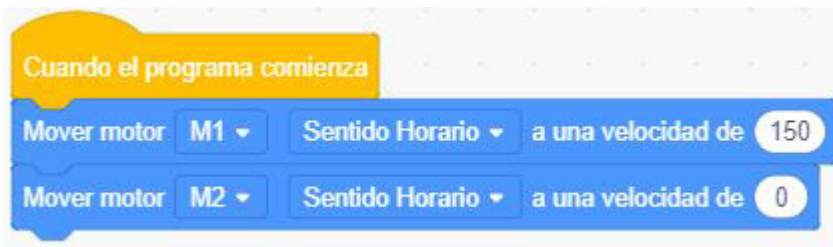
Motor 1 va en 150 rpm en sentido de las agujas del reloj y motor 2 va en 150 en sentido contrario a las agujas del reloj.

Agregá el código:

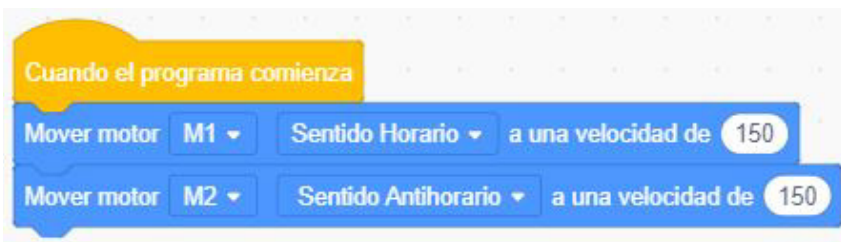
Caso 1



Caso 2



Caso 3



Luego de hacer las 3 pruebas, ¿qué giro fue el más efectivo?, ¿qué lo vuelve más operativo?

¿Te animás a hacer una rutina de programación calculando exactamente cuánto tiempo y a qué velocidad debe avanzar el robot para cumplir el siguiente desafío?:

Emotibot parte desde un punto designado y señalado en el piso. Debe llegar a la pared y en lugar de chocar con ella, girar hacia la izquierda ¡No uses sensores! ¡Sólo con cálculos matemáticos podés lograrlo!

EMOTIBOT INTELIGENCIA ARTIFICIAL



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Orientado de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

el análisis crítico de las perspectivas futuras y el impacto sobre la interacción entre el hombre y los entornos digitales, incluyendo los usos de la inteligencia artificial para la resolución de distintos problemas sociales y en diferentes ámbitos.

OBJETIVOS CURRICULARES ARTICULADOS

La interpretación y elaboración de información estadística en situaciones problemáticas que requieran:

- organizar conjuntos de datos discretos y acotados para estudiar un fenómeno, comunicar información y/o tomar decisiones, analizando el proceso de relevamiento de los datos; hipotetizar sobre el uso efectivo de IA en la vida.

DESARROLLO

Los/las estudiantes se sorprenden y utilizan cada día más los asistentes virtuales que, por ejemplo, proveen los teléfonos inteligentes y los relojes interactivos, entre otros. Buscan *apps* de traductores en tiempo real, interactúan con juegos que se vuelven “más inteligentes” a partir de las acciones de los usuarios. ¿Existen cámaras que deciden solas cuando es el mejor momento de tomar una fotografía?, ¿puede una *app* identificar algún objeto del alrededor y buscarlo en la tienda más cercana?

En esta actividad, analizan cómo desarrollar un programa de inteligencia artificial en el que se generen alteraciones en **Emotibot** a partir de datos que ingresan por la cámara Web de la *netbook*. El docente los contextualiza con el concepto IA y aplicaciones en campo real.

CIERRE, RECOMENDACIONES

Una vez creada la rutina de programación los/las estudiantes realizarán una segunda rutina con el objetivo de desarrollar en *Sprite*, un *chatbot*.

MOMENTO 1

¿Qué es la Inteligencia Artificial (IA)? ¿De qué manera es útil en la vida de las personas?

En estos últimos años hemos escuchado de manera creciente hablar sobre la temática IA. Computadoras y robots son capaces de aprender y mejorar su rendimiento a través de la utilización de patrones que les permitirá tomar decisiones de acuerdo al algoritmo que rige su comportamiento. Pero, aún así, no podemos dejar de hacernos algunas preguntas. ¿Una máquina puede pensar?

“¿Se volverán las máquinas más inteligentes que los seres humanos?

No, responde Jean-Gabriel Ganascia: se trata de un mito inspirado por la ciencia ficción” (1)

Se propone la lectura del artículo y su posterior análisis para abordar a una definición de IA en grupo.

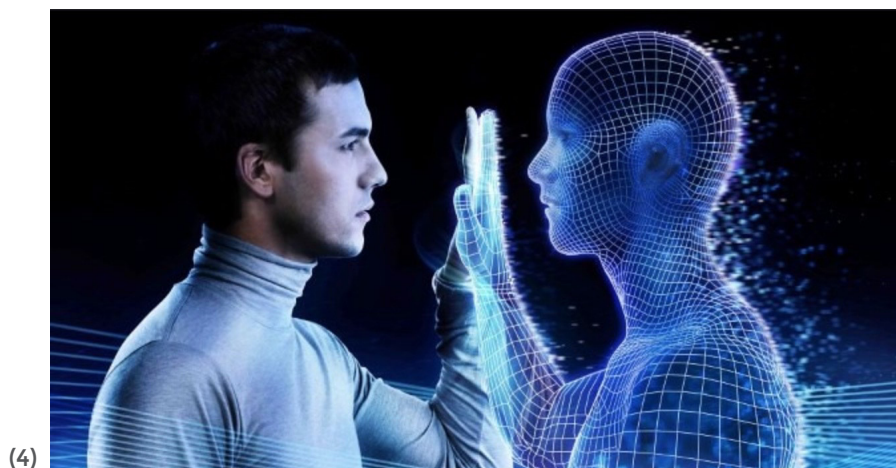
MOMENTO 2

Buscamos imágenes que traducen lo que puede ser la IA y su interacción con los humanos.

Además proponemos ver el siguiente video, Eli Pariser : Attenti alle “gabbie di filtre” in rete (Cuidado con las “jaulas de filtro” en la red)

https://www.ted.com/talks/eli_pariser_beware_online_filter_bubbles

(1) (cita *Inteligencia artificial. Promesas y amenaza*, 2018, Unesco recuperado https://unesdoc.unesco.org/ark:/48223/pf0000265211_spa 03/10/2019).



Se espera generar en los/las estudiantes el pensamiento crítico sobre la temática.

¿Consideramos que con la IA, las máquinas pueden reaccionar ante nuestras emociones?, ¿qué imaginamos que podrían hacer?, ¿cuál sería el objetivo de esta interacción?

¿Conocemos algún caso en que una máquina o *software* reaccione a un estímulo emocional de un usuario?

Podemos profundizar leyendo el siguiente artículo y viendo el video sugerido en él.

<https://www.bbvaopenmind.com/humanidades/sociologia/en-busca-de-la-nueva-empatia-humano-maquina/>

(3) <https://pixabay.com/es/illustrations/evoluci%C3%B3n-inteligencia-artificial-3778196/>

(4) <https://www.flickr.com/photos/150936425@N04/32502383612>

MOMENTO 3

Desafío 1:

Es posible programar un *software* o robot para que sea empático a nuestras emociones. Para ello haremos una rutina de programación en la que **Emotibot** (modelo **LI'L GUARDIAN(*)**) cambie de estado animo por cada emoción del usuario. Veamos un ejemplo:



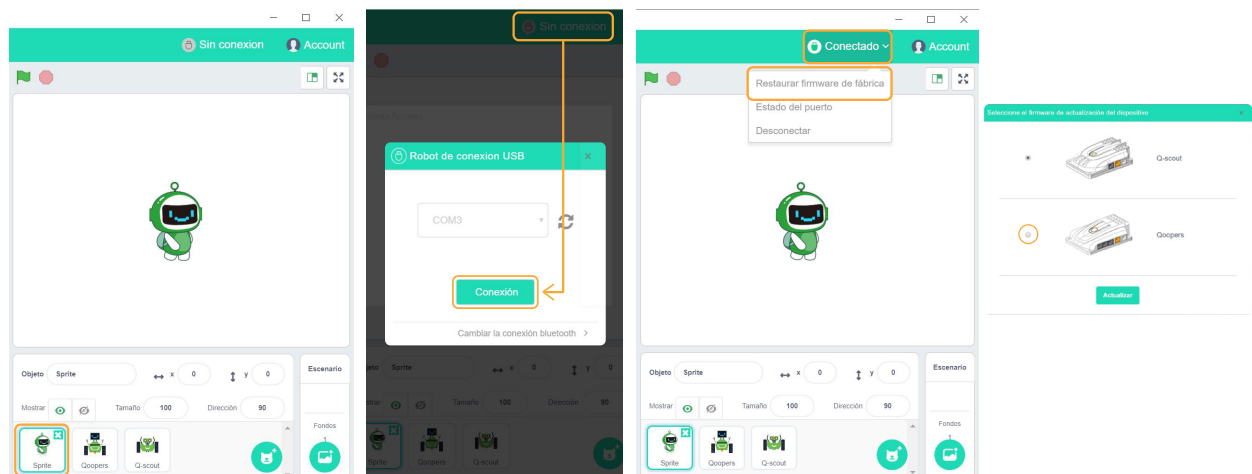
En esta rutina indicamos lo siguiente:

- al hacer clic en banderita verde,
- por siempre consulta:
- la detección de emociones faciales por cada segundo de registro,
- si la emoción es “feliz”, entonces,
- enciende la luz del sensor ultrasónico (conectado en puerto 3) en color verde,
- establece en la matriz de led (conectado en puerto 2) un ícono de pantalla con cara feliz,
- si la emoción es “triste”, entonces,
- enciende la luz del sensor ultrasónico (conectado en puerto 3) en color azul,
- establece en la matriz de led (conectado en puerto 2) un ícono de pantalla con cara triste.

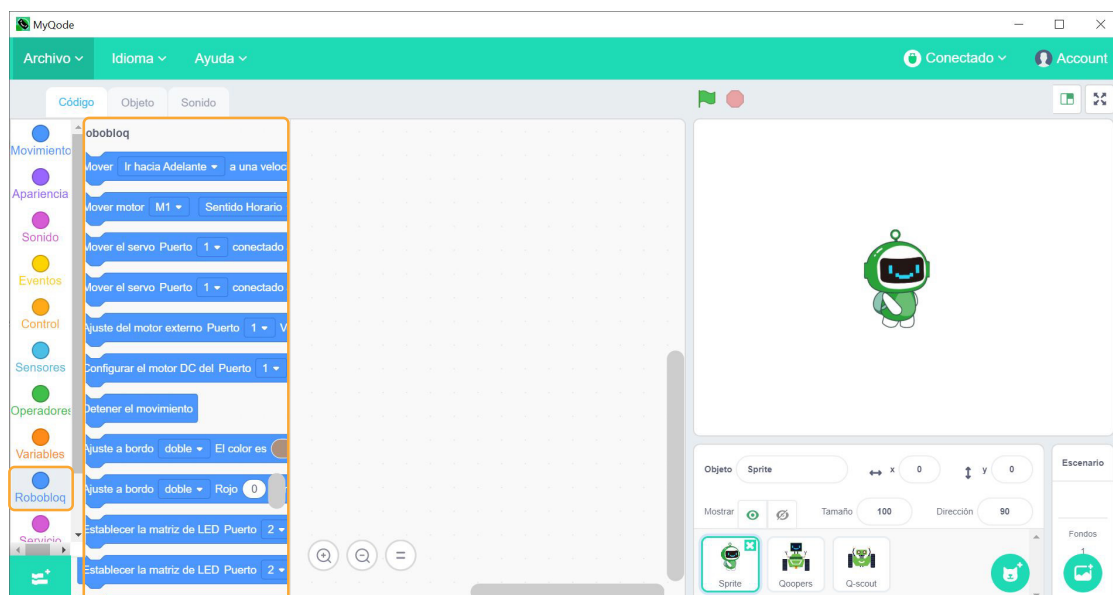
(*) Ver anexo de construcción - p.96.

APRENDER CONECTADOS

Para trabajar los comandos de IA con interacción del robot, abrimos *MyQode* y seguimos los próximos pasos:



- 1- seleccionamos el objeto *Sprite*.
- 2- conectamos el robot al puerto *USB*.
- 3- hacemos clic en “Sin conexión”.
- 4- se abre una ventana emergente que detecta el puerto conectado. Apretamos el botón “Conexión”.
- 5- cuando el estado pasa de “Sin conexión” a “Conectado”, desplegamos la solapa de acciones y seleccionamos “Restaurar *firmware* de fábrica”.
- 6- aparece una ventana emergente que nos da a elegir la placa del robot. En el caso de **Emotibot** tiene la placa identificada con el nombre *Qoopers*.
- 7- automáticamente se suma a los paneles de comandos, uno nuevo que se llama “*Robobloq*” que nos permite ingresar instrucciones para el robot.



Desafío 2:

Tomando de referencia la secuencia en donde hemos puesto feliz o triste a **Emotibot** (Desafío 1) pensamos en otras emociones. Programaremos el comportamiento del robot en relación a cada emoción.

CIERRE, RECOMENDACIONES

Una vez explorado el panel “Servicios cognitivos” que contiene los comandos IA, realizaremos una tercer rutina de programación con el objetivo de desarrollar un *chatbot digital*.

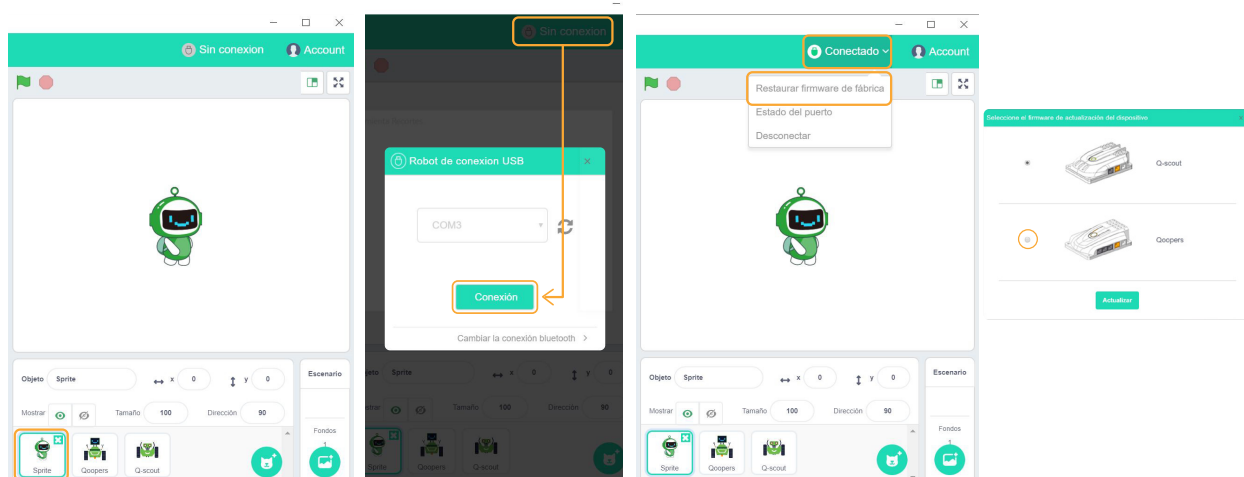
- ¿Conocés algún *chatbot*?
- ¿Dónde los encontrás?, ¿en computadoras?, ¿en celulares?, ¿en páginas Web?
- ¿Charlamos con uno? podemos visitar la página: <http://deixilabs.com/alizia.html>
- ¿Hacemos un *chatbot* de entretenimiento con un objeto de *MyQode*? Usemos comandos de Inteligencia Artificial.

EMOTIBOT INTELIGENCIA ARTIFICIAL



¡Generamos empatía con Emotibot!

¡Trabaja los comandos de IA interactuando con el robot!
Abrí MyQode y seguí los pasos:

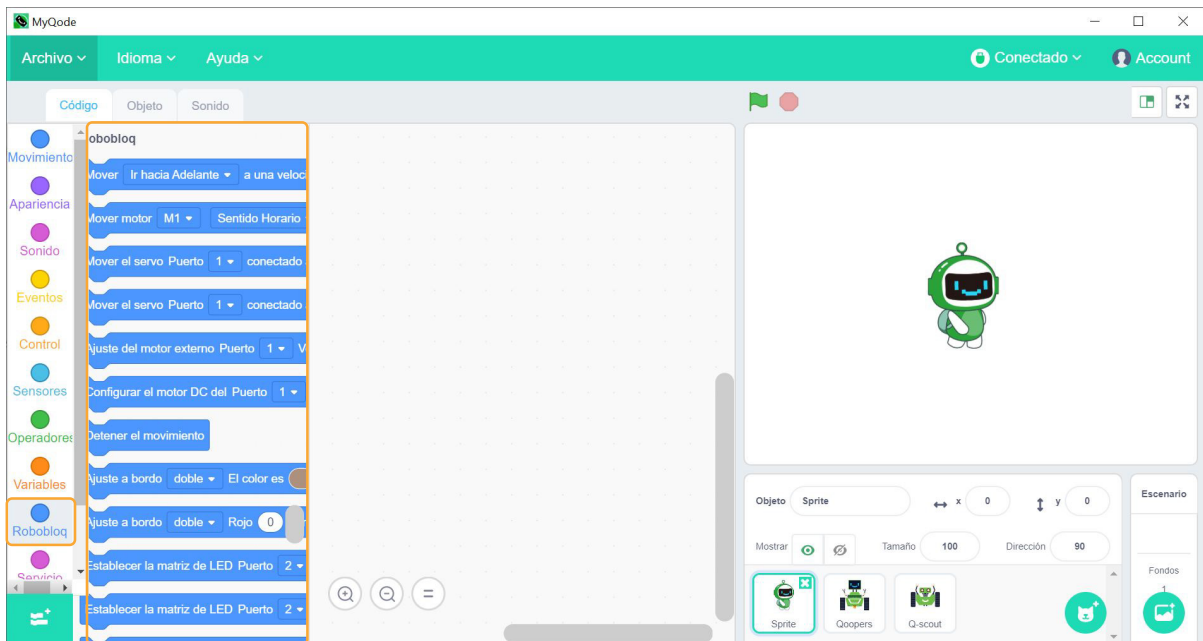


- 1- seleccionar el objeto *Sprite*.
- 2- conectar el robot al puerto *USB*.
- 3- hacer clic en “Sin conexión”.
- 4- se abre una ventana emergente que detecta el puerto conectado. Hacer clic sobre el botón “Conexión”.
- 5- cuando el estado pasa de “Sin conexión” a “Conectado”, desplegar la solapa de acciones y seleccionar “Restaurar *firmware* de fábrica”.

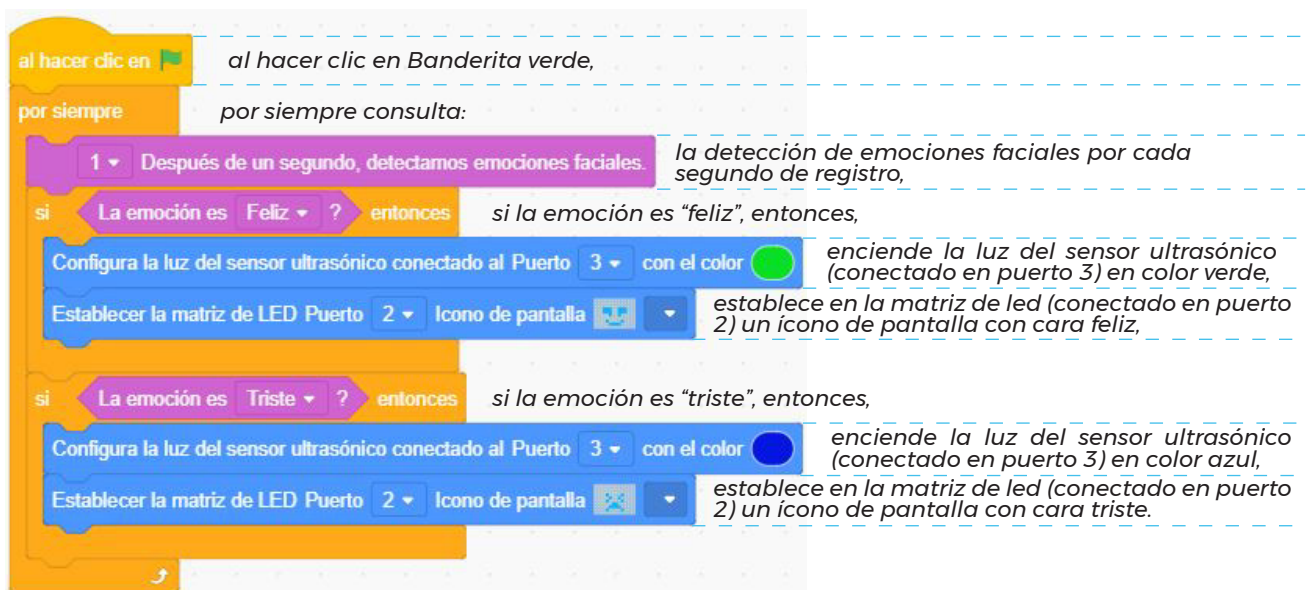
APRENDER CONECTADOS

6- aparece una ventana emergente que nos da a elegir la placa del robot. **Emotibot** tiene la placa identificada con el nombre *Qoopers*.

7- automáticamente se suma a los paneles de comandos, uno nuevo que se llama “*Robobloq*” y que permite ingresar instrucciones para el robot:

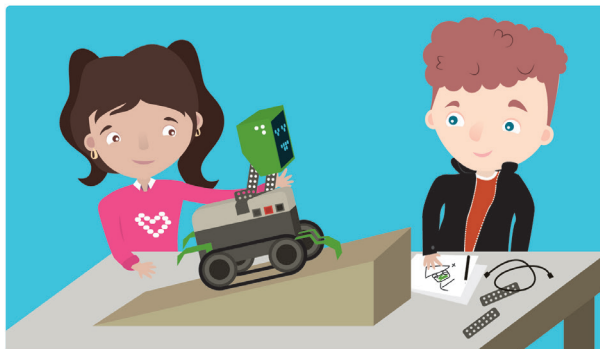


Agregá el código:



Tomando de referencia esta secuencia, pensá en otras emociones y animate a programar el comportamiento del robot en relación a cada emoción?

EMOTIBOT TODOTERRENO



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Orientado de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

La aplicación de sus habilidades analíticas, de resolución de problemas y de diseño para desarrollar proyectos de robótica o programación física, de modo autónomo, crítico y responsable, construyendo soluciones originales a problemas de su entorno social, económico, ambiental y cultural.

OBJETIVOS CURRICULARES ARTICULADOS

La interpretación y la determinación de la correlación lineal entre dos variables aleatorias en situaciones que impliquen:

- la indagación de alguna asociación entre sus valores, permitiendo definir tendencias entre ellos.
- crear una rutina de programación para que el robot afronte diversos desafíos.

DESARROLLO

Los teléfonos inteligentes tienen varios sensores para diversos fines: de proximidad, de luz ambiente, brújula digital, barómetro, acelerómetro y giroscopio. Un giroscopio, añade el registro de la rotación del dispositivo en todos sus ejes. En clase se propone analizar este y otros dispositivos que tengan este sensor.

El desafío se basa en sensar, utilizando un giroscopio, la posición del robot. Diseñar y construir un terreno con subidas y bajadas, rampas y desafíos en donde el robot tenga que cambiar su angulación mientras lo recorre y adquiere más velocidad. Los/las estudiantes reflexionan sobre la lógica del comportamiento de los motores para superar la pendiente y no caer en la bajada. ¿Cómo se puede programar el robot para que se adecúe a subidas y bajadas de terreno?

CIERRE, RECOMENDACIONES

Los/las estudiantes revisan la programación realizada y la ajustan pensando en un suelo más húmedo, más pedregoso, más empinado, etc.

MOMENTO 1

¿De qué manera actúa un giroscopio? ¿Para qué es útil?

León Foucault empleó el giroscopio en 1852 para demostrar la rotación del planeta Tierra.

El primer uso que se le dio a los giroscopios fue en el estudio de los movimientos de rotación y traslación de la Tierra así como también los planetas y objetos en el espacio. También se utilizó el giroscopio en navegación para contrarrestar la oscilación de las embarcaciones.

(1)



El mítico inventor Dean Kamen basó la invención de un dispositivo de transporte de autobalanceo cuyo funcionamiento se basa en la utilización de un giroscopio. Es el elemento fundamental que informa al patín cómo mantenerse en equilibrio y si debe avanzar o frenar.

(2)



También algunos mandos de consolas de juegos se basa en esta tecnología para distinguir sus movimientos en el aire.

(3)



(1) https://es.m.wikipedia.org/wiki/Archivo:Foucault%27s_gyroscope.jpg
(2) <https://www.flickr.com/photos/136833700@N04/21835999610>
(3) https://es.m.wikipedia.org/wiki/Archivo:Wii_console.png

APRENDER CONECTADOS



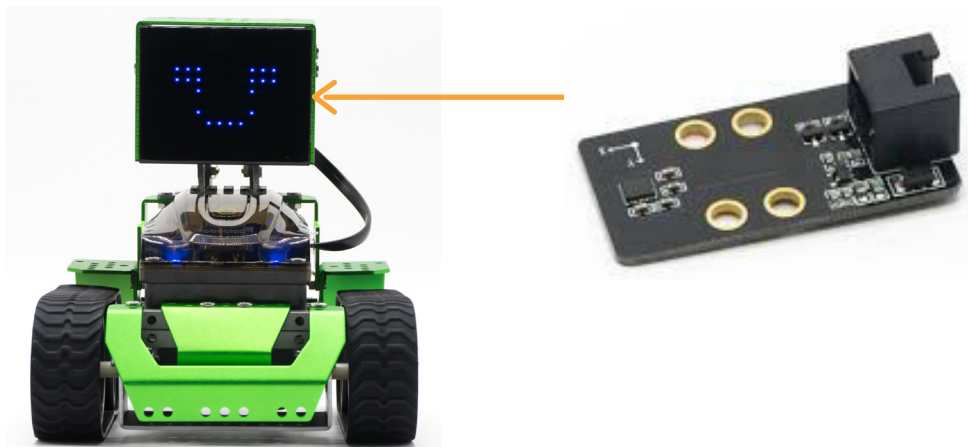
Dispositivos móviles, también , identifican si están en que posición están para mostrar una vista diferente de la información en pantalla.

Casi todos los tipos de estos dispositivos disponen de un giroscopio.

En la actualidad los aviones, los misiles, las naves espaciales y los satélites utilizan esta tecnología para saber si se encuentran nivelados respecto a la Tierra.

MOMENTO 2

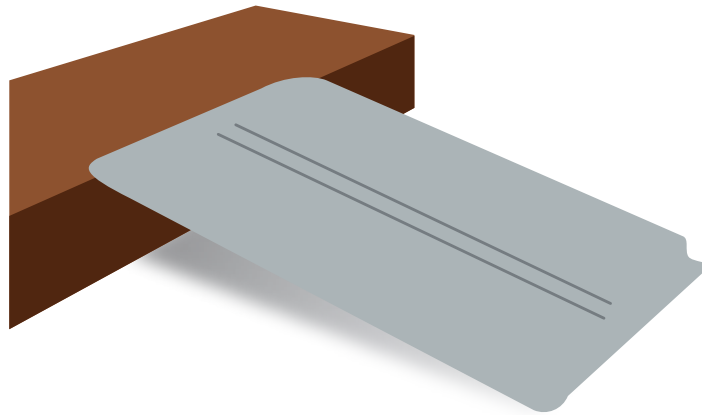
Construimos el **Emotibot** modelo (*Captain Alloy*^(*)) y le anexamos el sensor giroscopio.



(4) <https://www.flickr.com/photos/iphonedigital/34597403825>

(*) Ver anexo de construcción - p.100.

Creamos una rampa para el robot. La misma puede ser construída con algún material rígido que puede ser madera o cartón reforzado y alcanzará una altura de 20 cm, teniendo un ancho de 50 cm y un largo de 90 cm.



MOMENTO 3

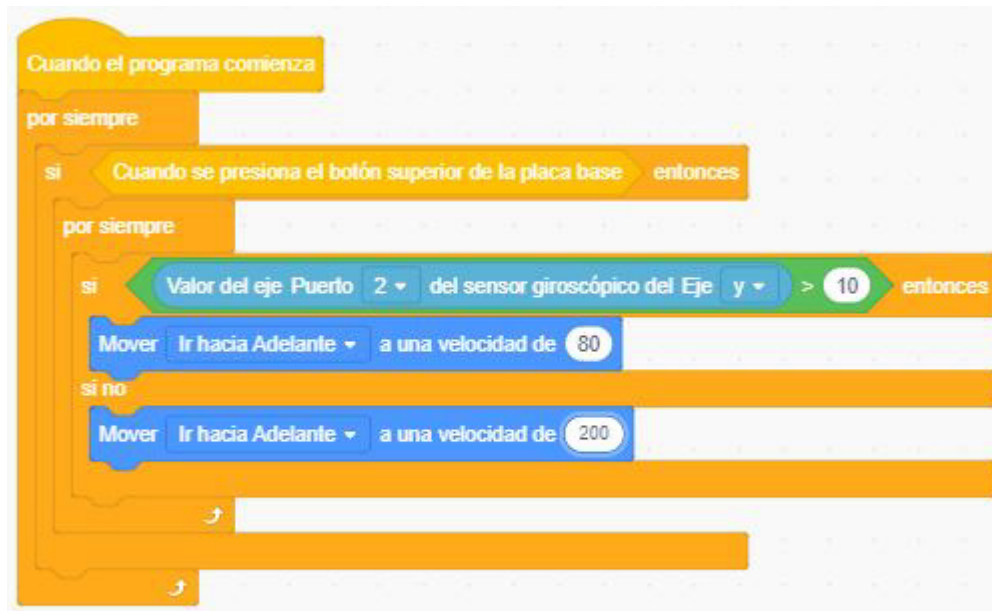
El desafío se basa en sensar, utilizando un giroscopio, la posición del robot. **Emotibot** sube la rampa y al hacerlo cambia su angulación de base. Por acción de la gravedad **Emotibot** necesita fuerza para subir la rampa.

Reflexionemos sobre la lógica del comportamiento de los motores para superar la pendiente. ¿Cómo se puede programar el robot para que se adecue a subidas de terreno?

Desafío 1:

Emotibot avanza recorriendo 1,5 mt en suelo previo a la pendiente. Esta acción la realizará con la mínima velocidad requerida para que los motores trasladen todo el cuerpo del robot. Cuando llegue a la pendiente su angulación cambiará, permitiéndonos ingresar otro dato mediante el sensor. Cuando detectemos que varía el eje y en nuestro giroscopio, los motores precisarán más velocidad para superar la pendiente y llegar al objetivo. Haremos las mediciones y la rutina de programación, ¿sobre qué eje (X, Y, o Z) debemos detectar la variación de valores del giroscopio ?

Ejemplo de rutina.



Desafío 2:

Emotibot al subir la rampa precisa girar y bajarla para volver al suelo. Alteraremos la rutina para conseguir este desafío. Vamos a tener que graduar la velocidad de los motores para que superen la bajada sin que se caiga. Haremos las mediciones y la rutina de programación correspondiente. ¿Qué tendremos que sumarle a la rutina del desafío 1? Registremos los datos en una tabla:

	En suelo	En subida (de 0 a 20 cm de altura)	En bajada (de 20 cm de altura a 0 cm)
Velocidad que usa Emotibot en ambos motores			

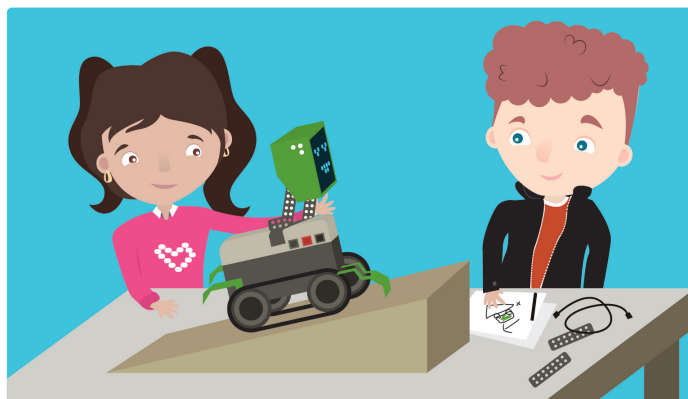
CIERRE, RECOMENDACIONES

Ajustamos la rampa y la hacemos más empinada.

1. Anotamos a cuántos cm del piso se ubicaría su altura máxima.
2. Modificamos la rutina de programación utilizada en MOMENTO 3.
3. Completamos la tabla.

	En suelo	En subida (de 0 a ___ cm de altura)	En bajada (de ___ cm de altura a 0 cm)
Velocidad que usa Emotibot en ambos motores			

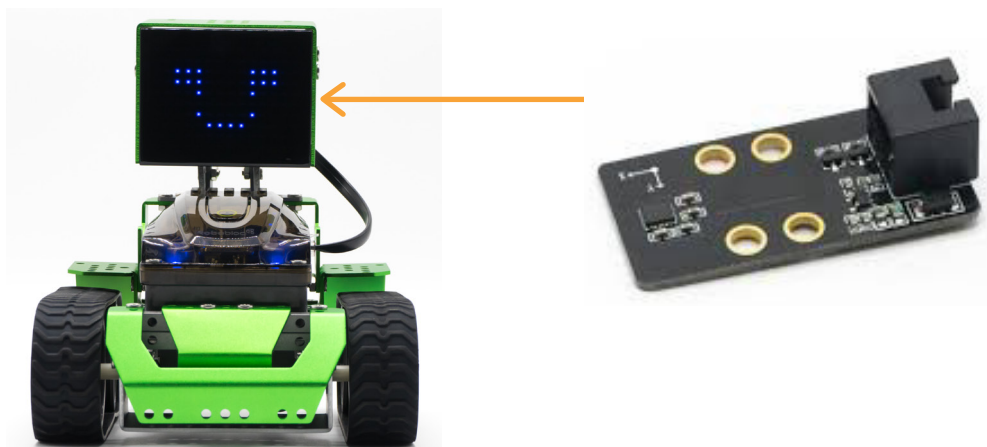
EMOTIBOT TODOTERRENO



¡Equilibrando a Emotibot!

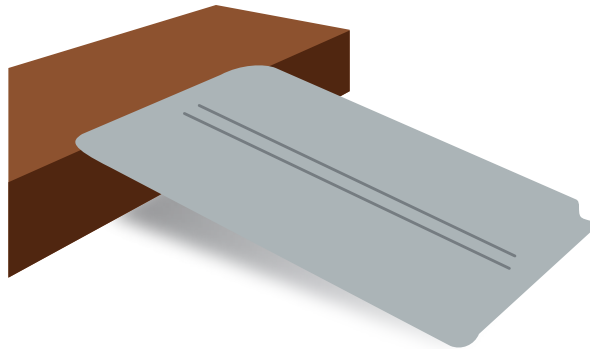
Para comenzar:

Anexale al **Emotibot** modelo *Captain Alloy* el sensor giroscopio.



Crea una rampa para el robot. La misma puede ser construída con algún material rígido que puede ser madera o cartón reforzado y alcanzará una altura de 20 cm, teniendo un ancho de 50 cm y un largo de 90 cm.

APRENDER CONECTADOS



El desafío se basa en sensar, utilizando un giroscopio, la posición del robot. **Emotibot** sube la rampa y al hacerlo cambia su angulación de base. Por acción de la gravedad **Emotibot** necesita fuerza para subir la rampa. Reflexioná sobre la lógica del comportamiento de los motores para superar la pendiente. ¿Cómo se puede programar el robot para que se adecue a subidas de terreno?

Agrega el código:

El siguiente es solo un ejemplo:



Emotibot al subir la rampa precisa girar y bajarla para volver al suelo. Modificá la rutina para conseguir este desafío. Debés graduar la velocidad de los motores para que superen la bajada sin que se caiga. ¿Qué habrá que sumarle a la rutina del desafío 1?

EMOTIBOT MEDIDOR DE LUZ



NIVEL SECUNDARIO

NAP ROBÓTICA, PROGRAMACIÓN Y EDUCACIÓN DIGITAL

Durante el Ciclo Orientado de la Educación Secundaria, la escuela ofrecerá situaciones de enseñanza que promuevan en los alumnos y alumnas:

- La aplicación de sus habilidades analíticas, de resolución de problemas y de diseño para desarrollar proyectos de robótica o programación física, de modo autónomo, crítico y responsable, construyendo soluciones originales a problemas de su entorno social, económico, ambiental y cultural.

OBJETIVOS CURRICULARES ARTICULADOS

La modelización de situaciones que involucren el uso de números reales mediante recursos tecnológicos y de cálculo mental, lo que supone:

- expresar las soluciones mediante diferentes escrituras,
- acotar el error en función de lo que se busca resolver y comunicar,
- sintetizar en una secuencia de programación el cuidado de la energía.

DESARROLLO

Reconocer el exceso de consumo eléctrico es un gran problema que presentan las ciudades. El consumo innecesario de energía, genera que se busquen alternativas que permitan que nos transformemos en agentes más comprometidos con el cuidado del medio ambiente. La domótica surge de una propuesta sustentable de optimización de energía.

Muchas casas o sectores de hogares pueden presentar automatización en el encendido y apagado de la luz, regulación de tiempo activo de un aire acondicionado, automatización en el sistema de alarma, entre otros.

En el caso de que exista mucha luz natural en el ambiente de una casa, no es necesario tener encendidas lámparas. Es posible generar un sistema de optimización en este punto.

El desafío consta en generar que el robot detecte luz y reaccione ante estas 3 condiciones con su matriz de led de manera que se cumpla la condición:

APRENDER CONECTADOS

*si no hay luz del exterior -> encender toda la matriz de led
*si hay luz del exterior -> no encender la matriz de led
*si hay mediana intensidad de luz exterior -> encender la mitad de la matriz

CIERRE, RECOMENDACIONES

Los/las estudiantes investigan cómo se comporta un panel solar que sigue la órbita diurna del Sol.
Desafío: ¿Podría **Emotibot** seguir la luz?

MOMENTO 1

¿Qué es la domótica?

La domótica estudia la aplicación de diversos sistemas en las casas con la finalidad de aumentar la seguridad, el confort y la capacidad de gestión energética de la vivienda.

Visualizar el capítulo 13 de la Serie Somos Digitales ¿Dónde están los robots?

https://www.youtube.com/watch?v=_pa-VNOSPik

¿De qué manera la domótica puede ayudarnos a bajar el consumo de energía eléctrica?

Una casa domótica nos brinda una serie de ventajas, por un lado la seguridad por otro, el confort que proporcionan:

- Iluminar la casa con la fuente de energía que se desee y determinar la intensidad y función de la misma.
- Calefaccionar cada espacio de una casa con diferentes temperaturas según la necesidad.
- Cerrar las puertas y ventanas en función de las condiciones climatológicas.
- Cortar automáticamente el suministro de agua o gas cuando se detecte una pérdida.

MOMENTO 2

Investiguemos acerca de emprendimientos individuales, nacionales o internacionales para cuidar el medio ambiente. Qué sabemos acerca de energías limpias, energías renovables:

Se denomina “**energías renovables**” a aquellas fuentes energéticas basadas en la utilización del sol, el viento, el agua o la biomasa vegetal o animal - entre otras-. Se caracteriza por no utilizar combustible fósiles -como sucede con las energías convencionales-, sino recursos capaces de renovarse ilimitadamente. Su impacto ambiental es de menor magnitud dado que además de no emplear recursos finitos, no generan contaminantes. Sus beneficios van desde la diversificación de la matriz energética del país hasta el fomento a la industria nacional; y desde el desarrollo de las economías regionales hasta el impulso al turismo. (2)

(2) <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables>



(3)

Organizados en grupos, asignar a cada uno un tema para investigar:

- Energía eólica (4)
- Energía solar (5)
- Biomasa (6)
- Biogás (7)
- Biocombustibles (8)
- Pequeños Aprovechamientos Hidroeléctricos (9)
- Geotermia (10)
- ¿Qué son las energías renovables? (10)
- Energías renovables Objetivos de Desarrollo Sostenible (11)
- Cambio ambiental (12)

¡Organicemos la información en una presentación multimedia!

(3) <https://pixabay.com/es/photos/solarpark-de-parques-e%C3%B3licos-1288842/>

(4) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#1>

(5) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#2>

(6) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#3>

(7) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#4>

(8) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#5>

(9) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#6>

(10) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables#7>

(11) Recuperado de <https://www.argentina.gob.ar/energia/energia-electrica/renovables/que-son-las-energias-renovables>

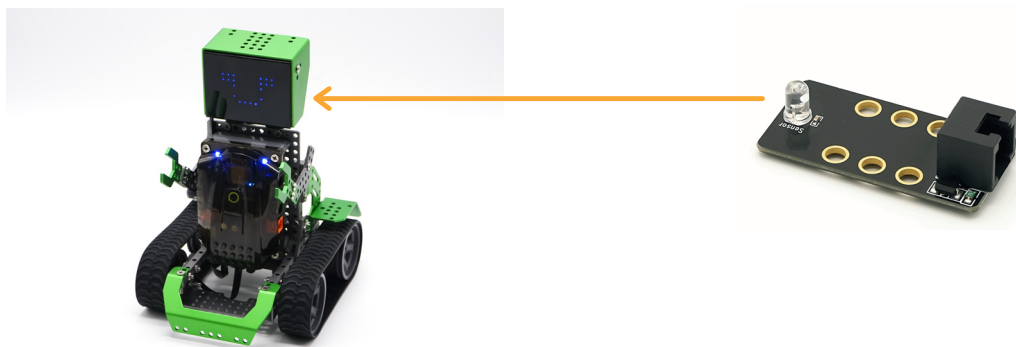
(12) Recuperado de <https://www.educ.ar/recursos/132532/energias-renovables>

(13) Recuperado de <https://www.educ.ar/recursos/120450/resumen>

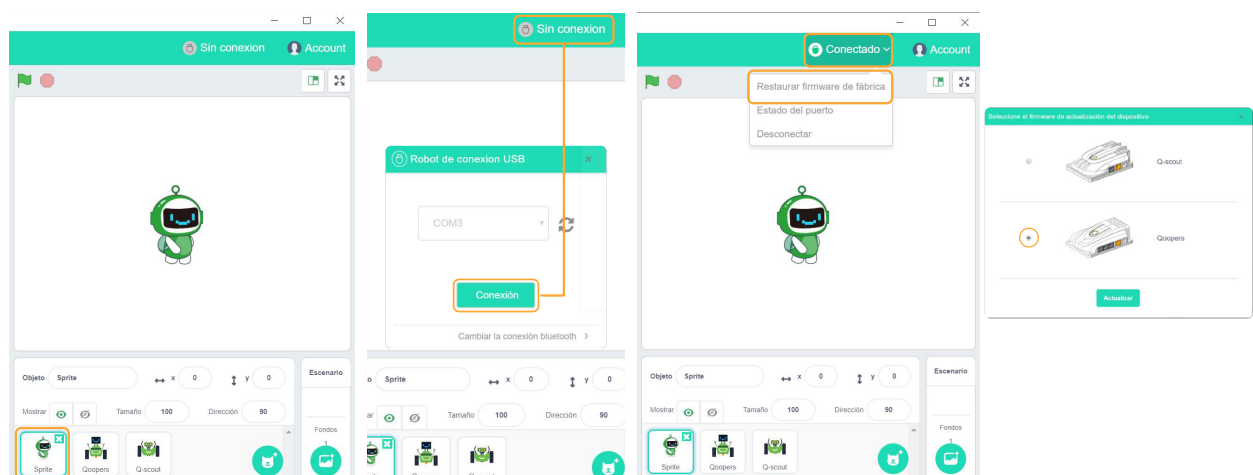
MOMENTO 3

Desarrollemos un sistema para optimizar la cantidad de luz artificial que utilizamos. Para ello precisamos poder medir la cantidad de luz natural existente en un espacio mediante un sensor de luz.

Anexemos a **Emotibot** (modelo **Cavalier(*)**) el módulo sensor de luz.



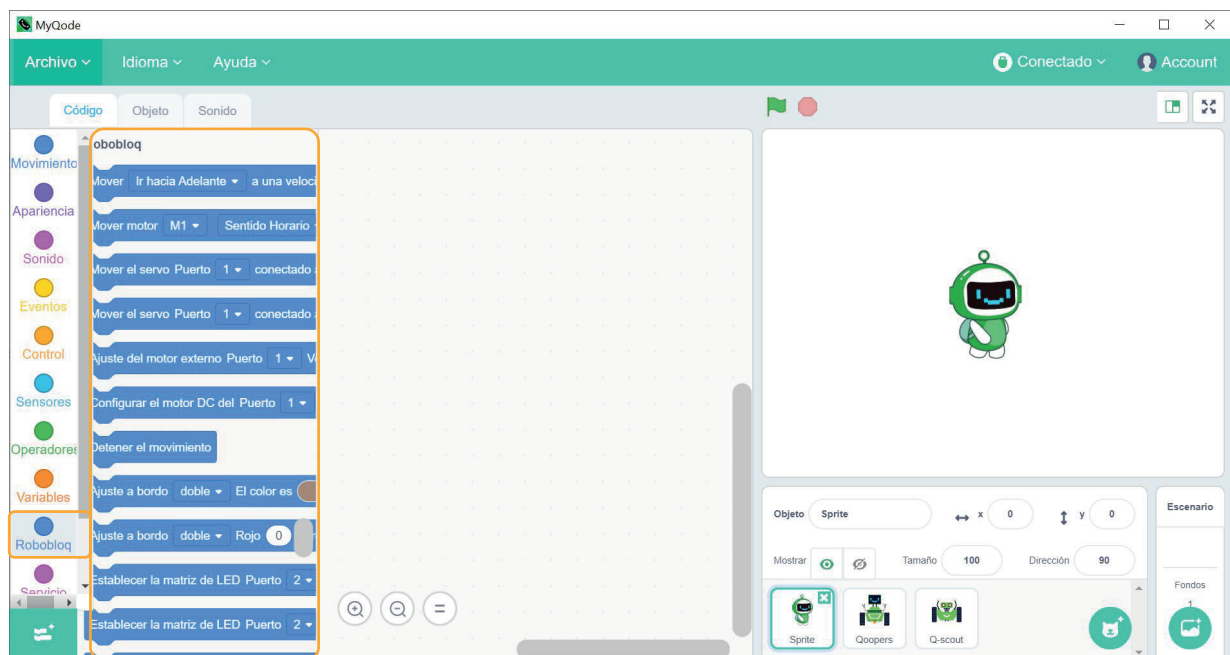
Para registrar el dato más alto que detecta el sensor, debemos leer su estado exponiéndolo a una luz dirigida y fuerte y seguir los siguientes pasos para leer los datos que ingresan al sensor:



(*) Ver anexo de construcción - p.105.

APRENDER CONECTADOS

- 1- seleccionamos el objeto *Sprite*.
- 2- conectamos el robot al puerto *USB*.
- 3- hacemos clic en “Sin conexión”.
- 4- se abre una ventana emergente que detecta el puerto conectado. Presionamos el botón “Conexión”.
- 5- cuando el estado pasa de “Sin conexión” a “Conectado”, desplegamos la solapa de acciones y seleccionamos “Restaurar *firmware* de fábrica”.
- 6- aparece una ventana emergente que nos da a elegir la placa del robot. En el caso de **Emotibot** tiene la placa identificada con el nombre *Qoopers*.
- 7- automáticamente se suma a los paneles de comandos, uno nuevo que se llama “*Robobloq*” que nos permite ingresar instrucciones para el robot.



- 8- generamos la siguiente rutina. Ella nos permite visualizar los datos que ingresa el sensor en el globo de diálogo de *Sprite*.



Para entender los valores que oscila el sensor, debemos dejarlo a oscuras y medir. Luego exponerlo a una luz fuerte y cercana. Así veremos el máximo nivel de luz que registra y el mínimo y estableceremos el parámetro de detección. Por ejemplo, supongamos que ante un ambiente totalmente oscuro, el sensor arroja un valor de 10 y ante uno altamente iluminado, 1000, determinamos que la media es 500 y podemos generar un estado de acción ante cada rango de detección.

Los valores varían según el espacio y la cantidad de luz que haya en él. El sensor puede medir de 0 a 1023 valores referentes a intensidad lumínica.

El desafío consta en generar que el robot detecte luz y reaccione ante estas 3 condiciones con su matriz de *led*:

*si no hay luz del exterior -> encender toda la matriz de *led*

*si hay luz del exterior -> no encender la matriz de *led*

*si hay mediana intensidad de luz exterior -> encender la mitad de la matriz

Ejemplo de rutina:



CIERRE, RECOMENDACIONES

La energía solar es inagotable, motivo por el cual, pertenece a la categoría de fuentes de energía renovable.

¿De qué manera recolectan la luz los paneles solares?

Los paneles solares recolectan la energía solar y la transforman en energía eléctrica.

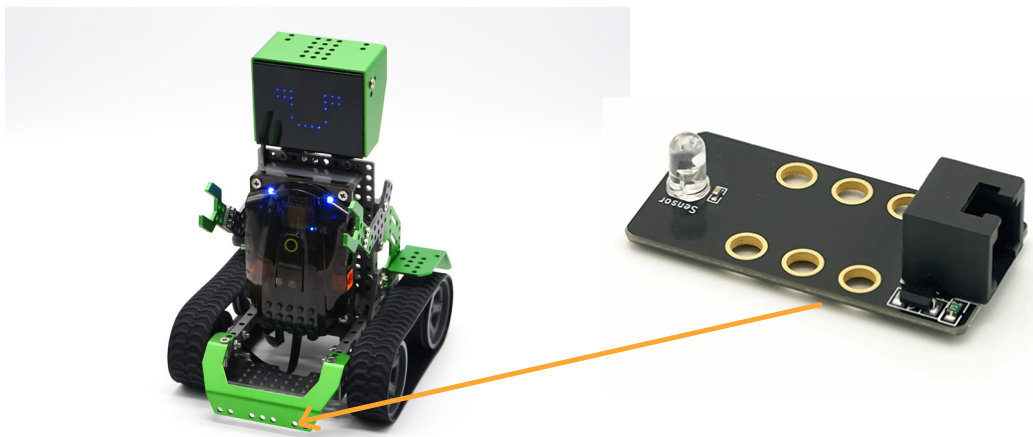
Observemos la siguiente imagen:

(14)



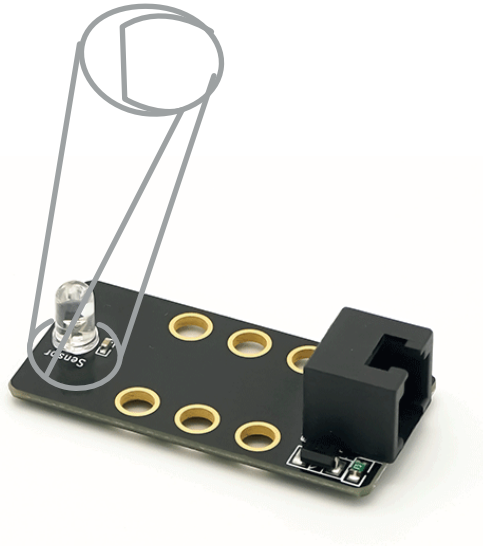
Para recolectar la mayor cantidad de energía, es indispensable que los paneles sigan la órbita diurna del Sol. Para ello se instala un rastreador de doble eje, cuya función es girar el panel para seguir el Sol, captando así la mayor cantidad de energía posible.

¿Podría **Emotibot** mostrarnos de qué manera seguir la luz? Para ello cambiamos la ubicación del sensor hacia el parabolismo del robot.



(14) https://es.m.wikipedia.org/wiki/Archivo:Paneles_solares_en_Cari%C3%B1ena,_Espa%C3%B1a,_2015-01-08,_DD_09-12_PAN.JPG

Vamos a tener que reducir el campo de detección del sensor, armando un cono de papel alrededor del lente.



Desarrollemos entonces la rutina de programación para cumplir los siguientes objetivos:

- con una linterna cuando le demos luz al sensor, **Emotibot** irá hacia adelante para acercarse a la luz.
- cada vez que no detecte luz, quedará detenido.

EMOTIBOT MEDIDOR DE LUZ



¡Emotibot un robot iluminado!

Para comenzar:

1- Al siguiente modelo agregá el sensor de luz para poder medir los datos del ambiente:



2- Utilizá el modo interactivo del *software* para escribir los datos que arroja el sensor dentro del globo de diálogo de *Sprite*.



APRENDER CONECTADOS

3- Establece el parámetro de detección del sensor: valores mínimos y máximos que detecta. También determina el valor medio.

4- El desafío consta en programar al robot para que detecte luz y reaccione ante estas 3 condiciones con su matriz de *led*:

*si no hay luz del exterior -> encender toda la matriz de *led*

*si hay luz del exterior -> no encender la matriz de *led*

*si hay mediana intensidad de luz exterior -> encender la mitad de la matriz

Agregá el código:

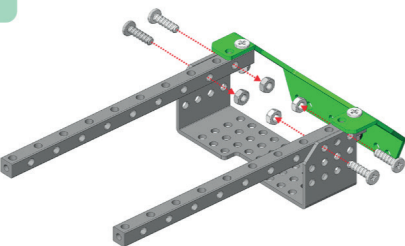
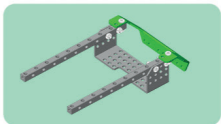
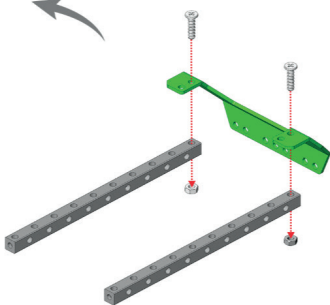
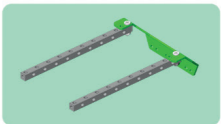
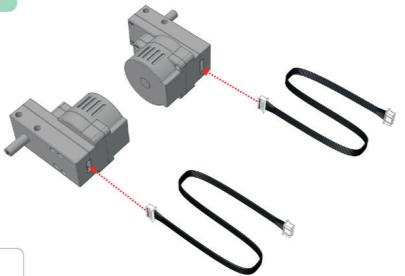
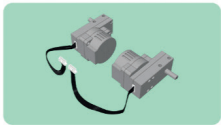
Adecuá este ejemplo de rutina de programación al parámetro que detecta el sensor de luz en el ambiente donde estás.



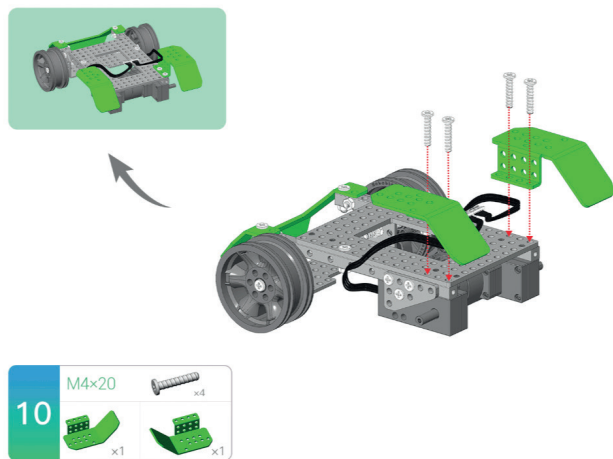
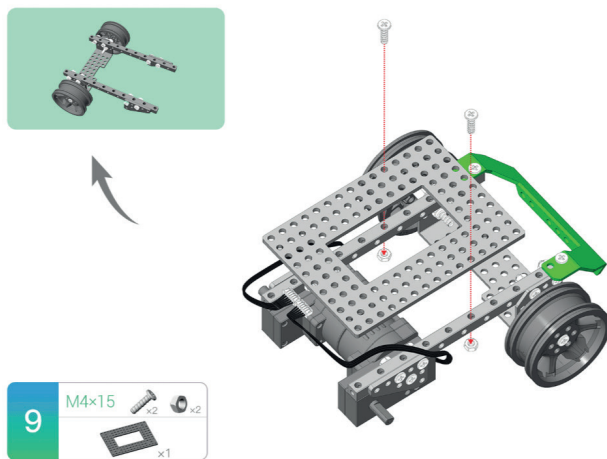
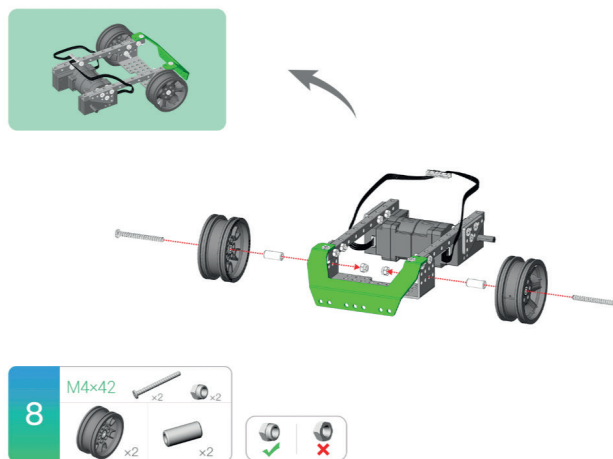
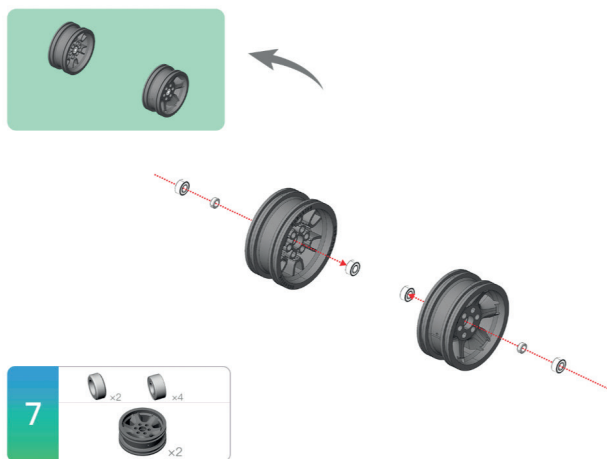
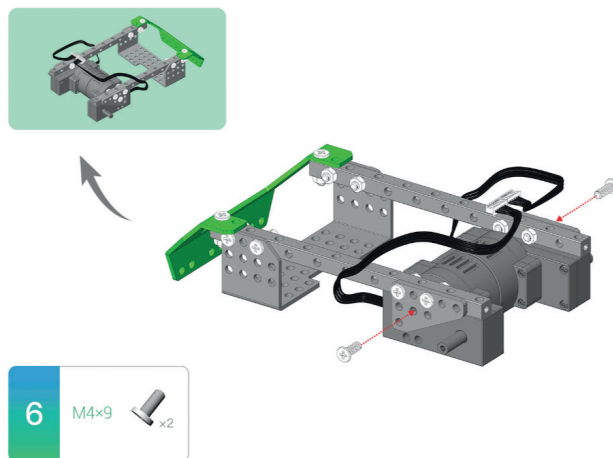
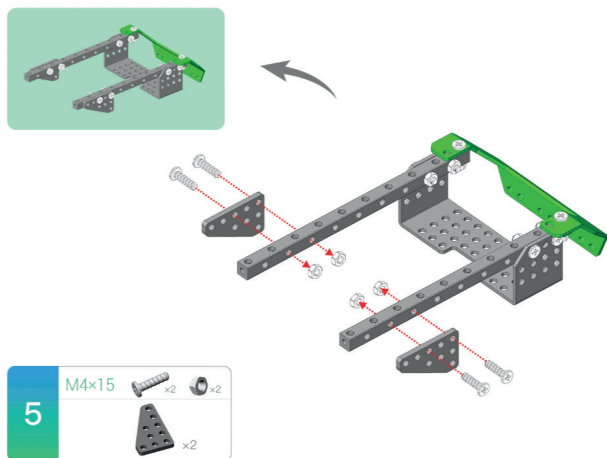
Modificá la rutina para darle emociones a la matriz de *led*, mostrando una cara feliz en caso de detectar luz, una expresión seria, si lee mediana intensidad de luz de ambiente o asustada si detecta oscuridad

Anexo: guía de construcción

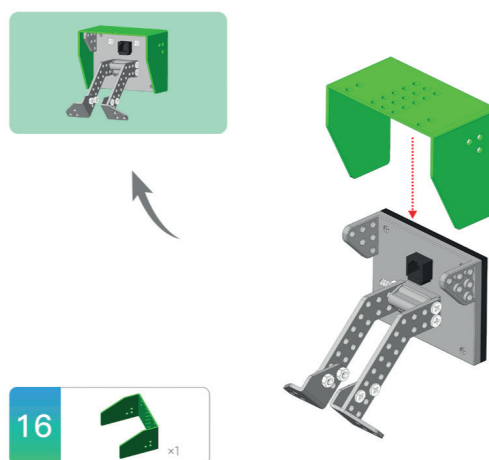
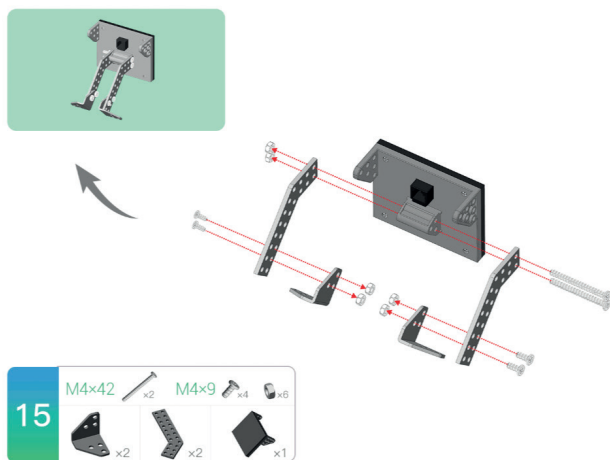
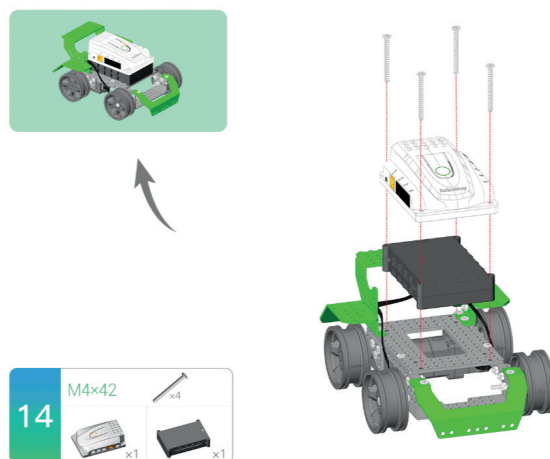
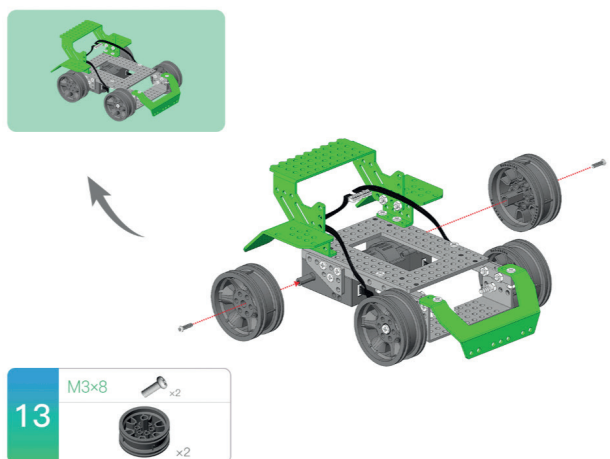
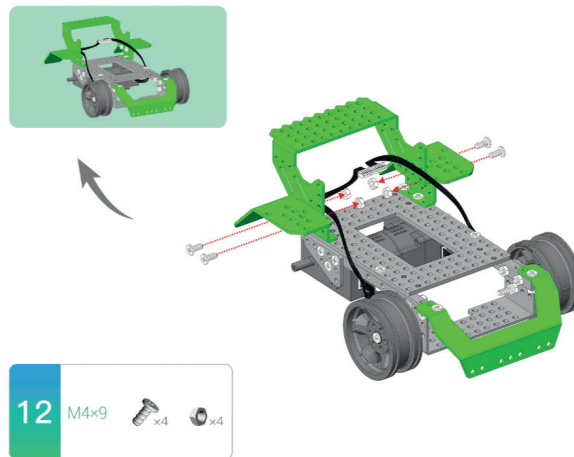
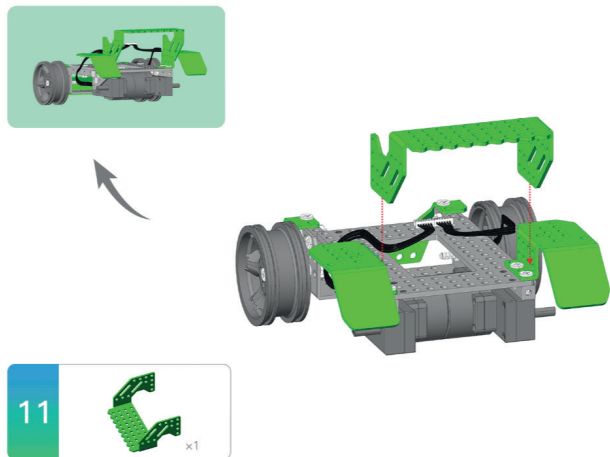
MODELO *LI'L GUARDIAN*



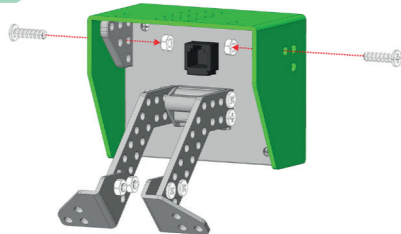
APRENDER CONECTADOS



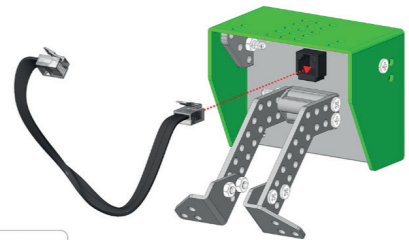
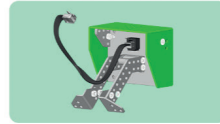
APRENDER CONECTADOS



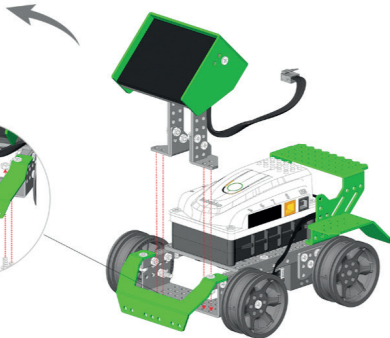
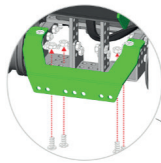
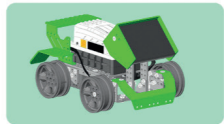
APRENDER CONECTADOS



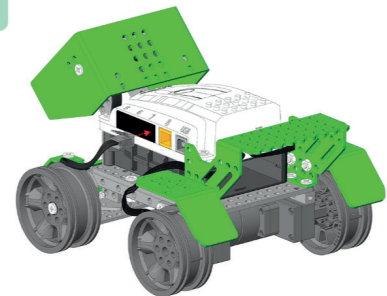
17 M4x15  x2  x2



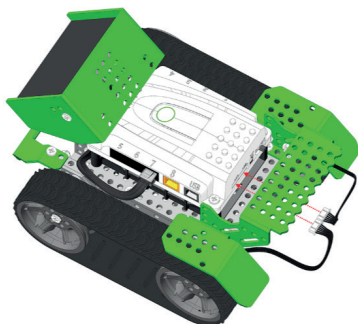
18  x1



19 M4x9  x4  x4

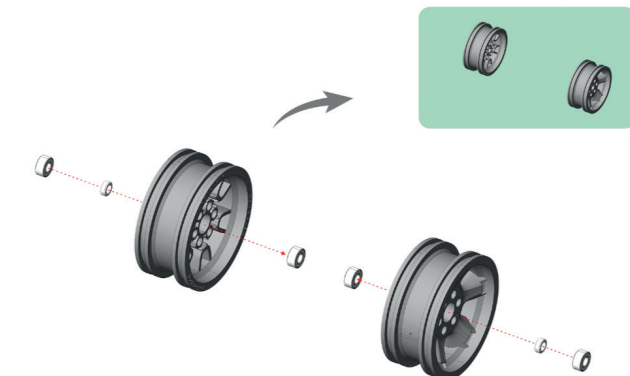
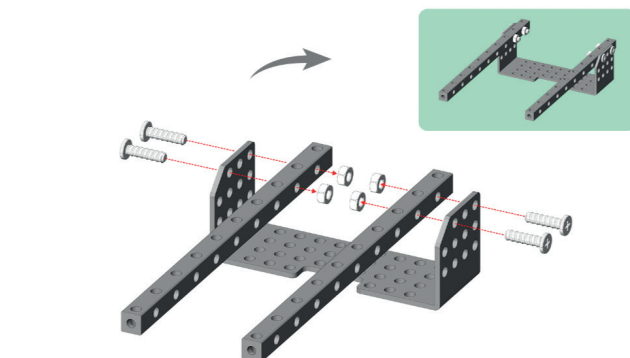
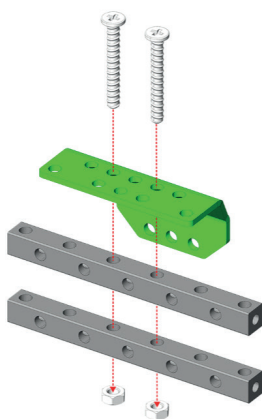
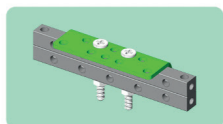
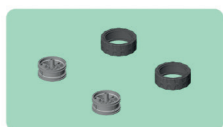
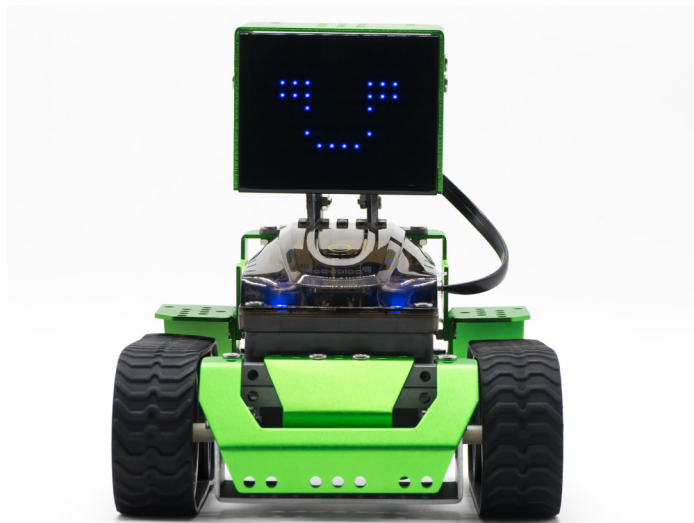


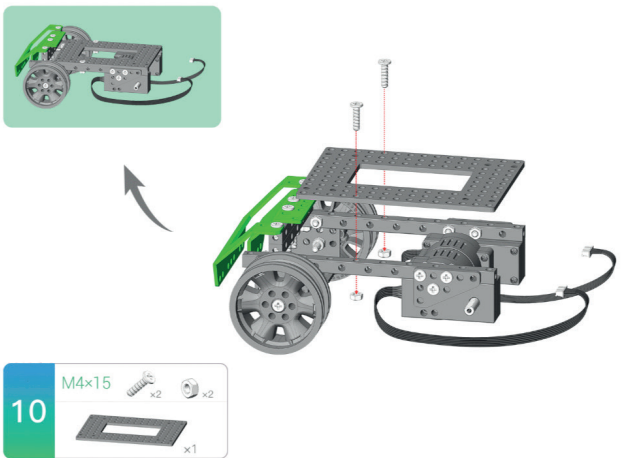
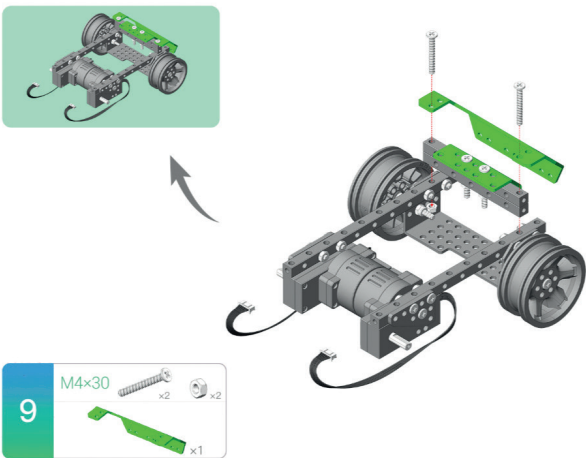
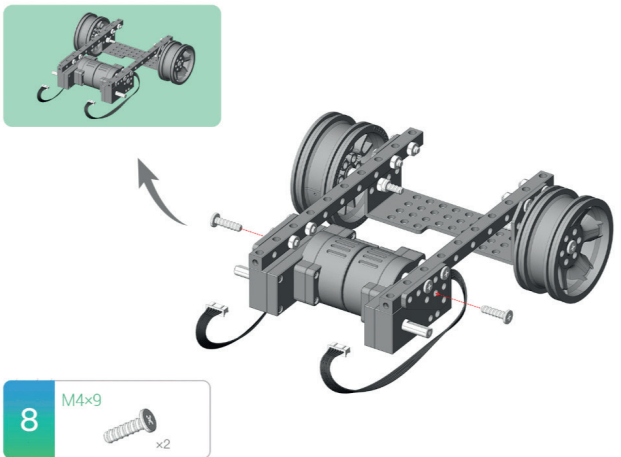
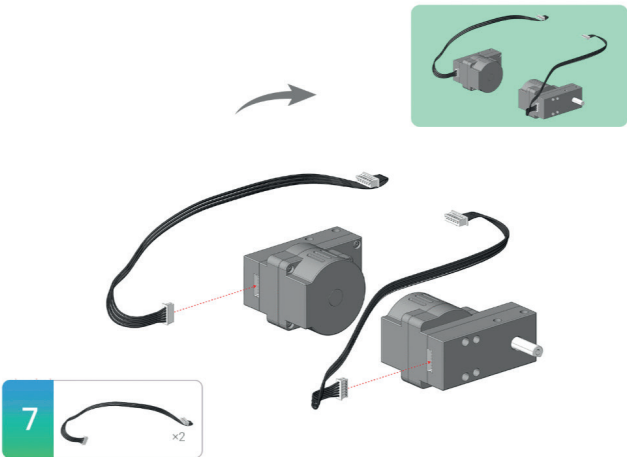
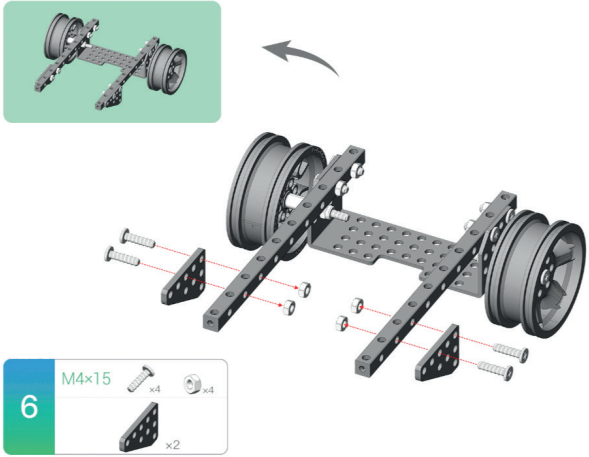
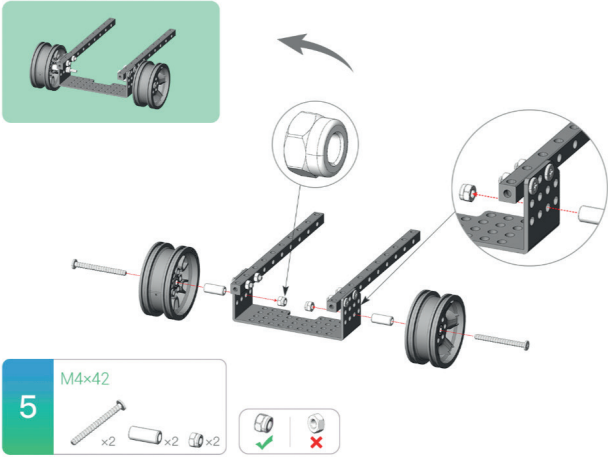
20



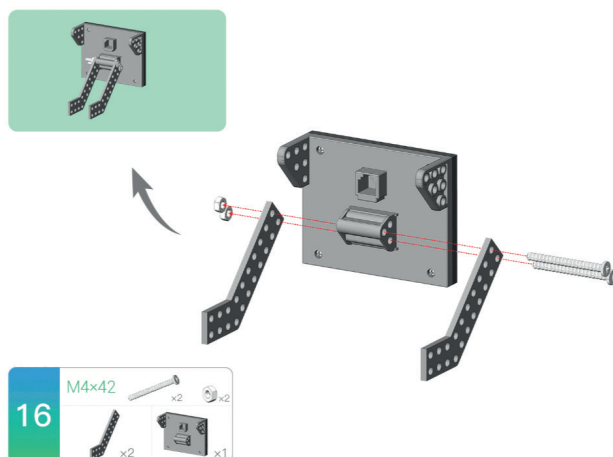
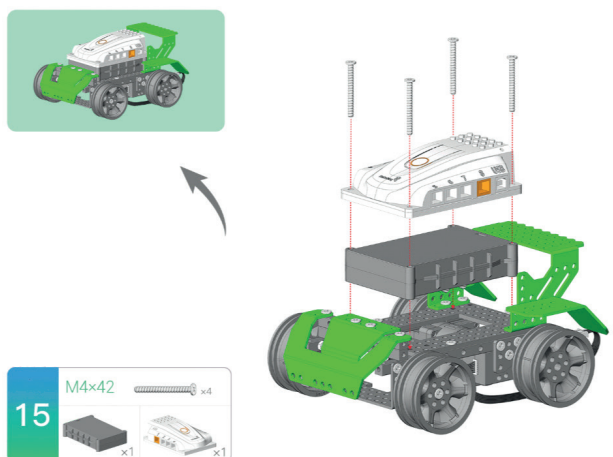
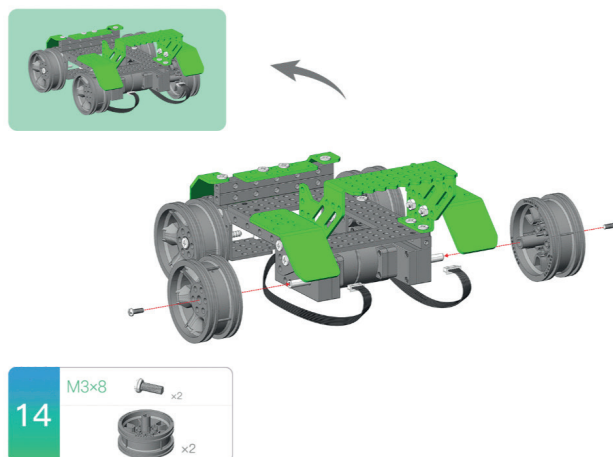
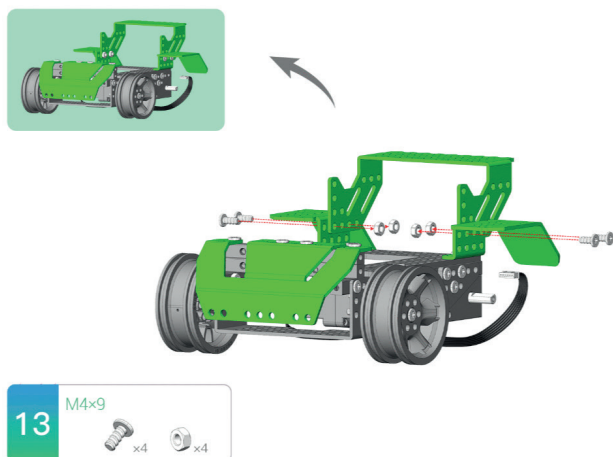
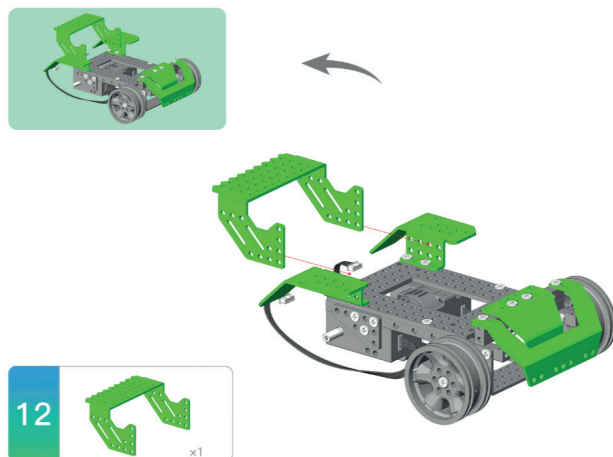
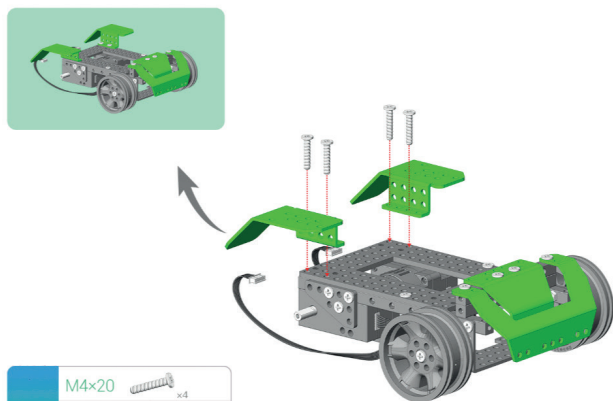
21

MODELO CAPTAIN ALLOY

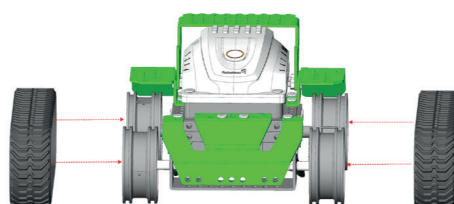
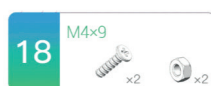
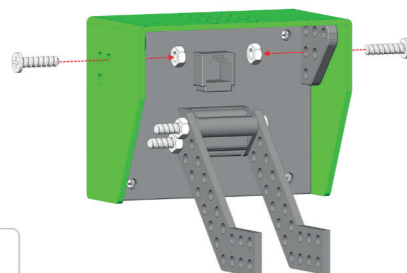
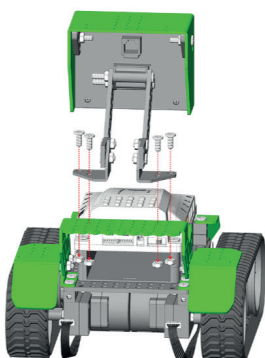
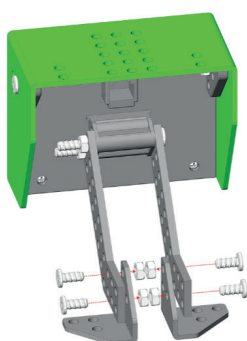
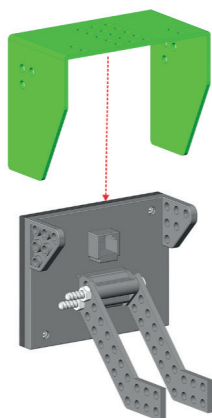


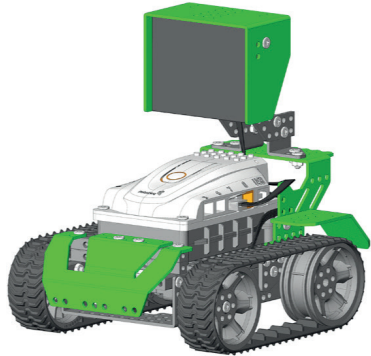


APRENDER CONECTADOS



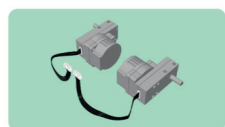
APRENDER CONECTADOS

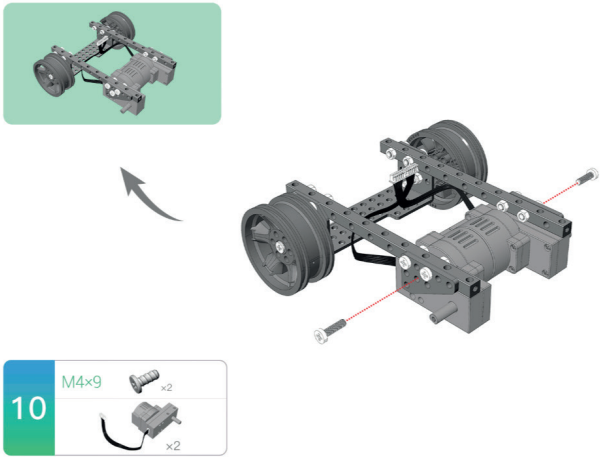
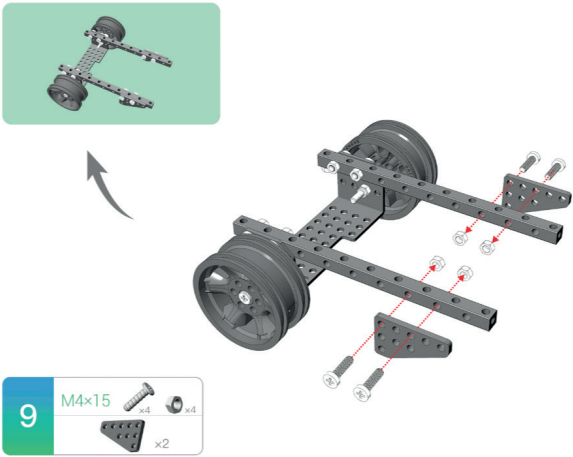
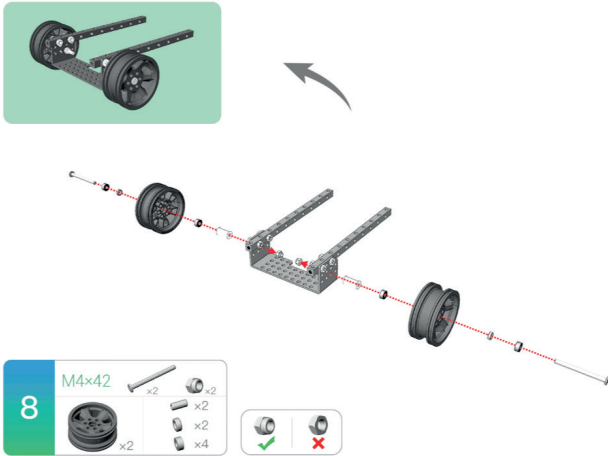
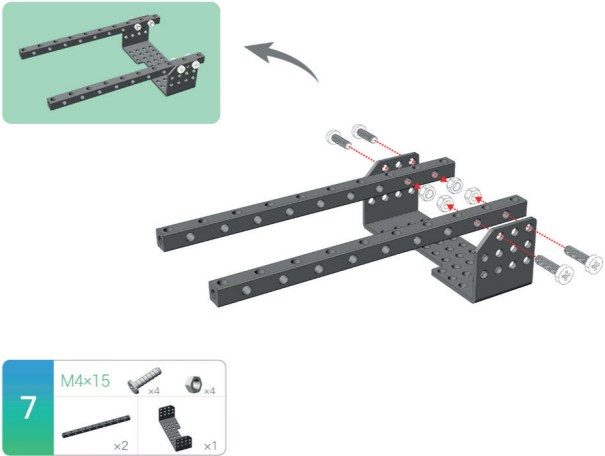
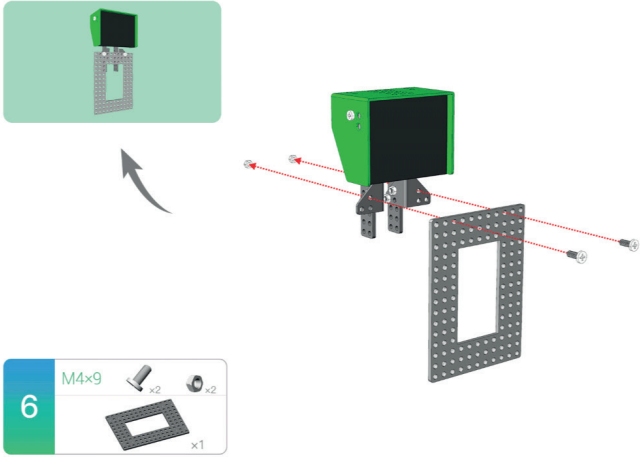
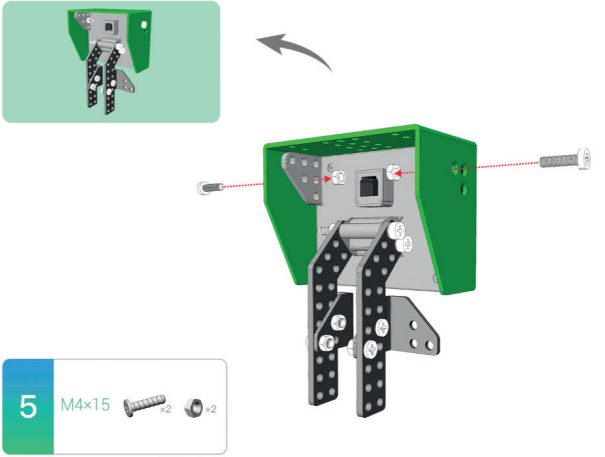


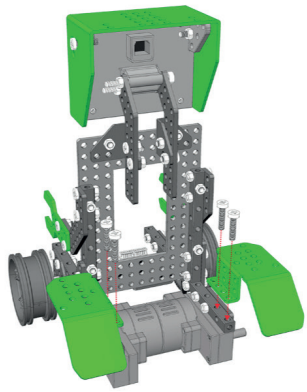
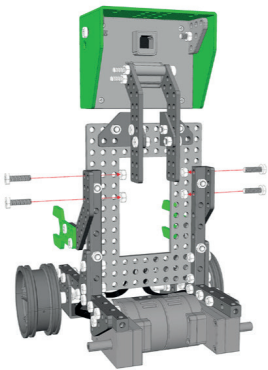
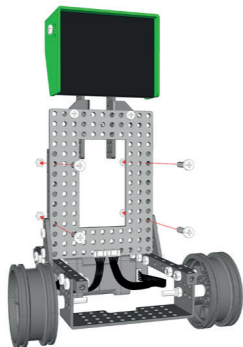
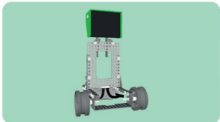
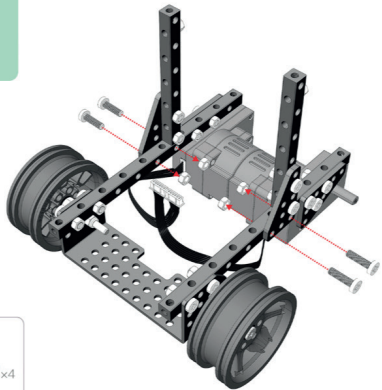
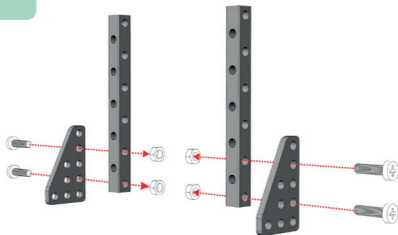


23

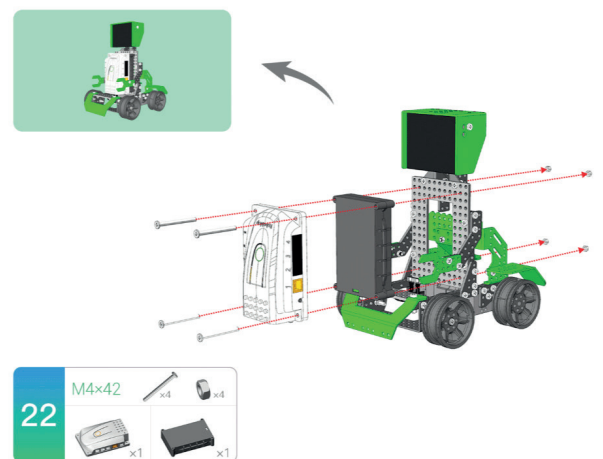
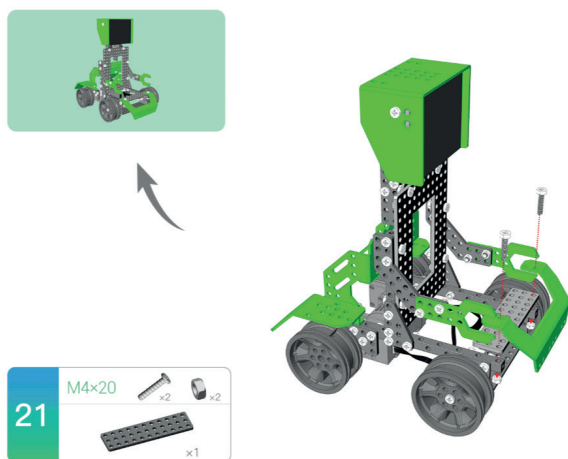
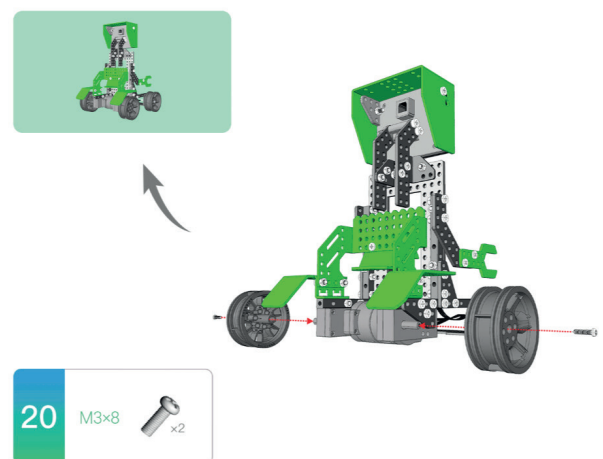
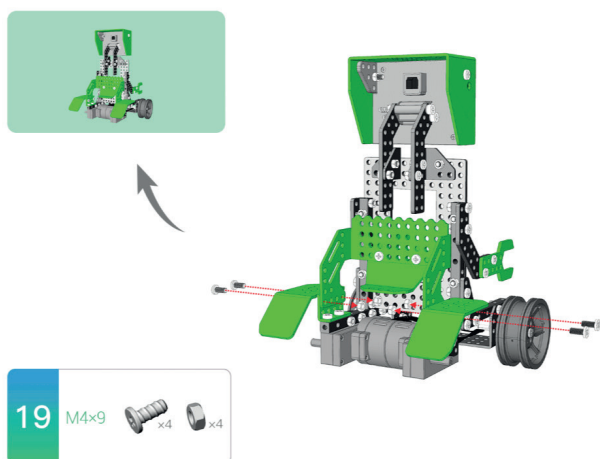
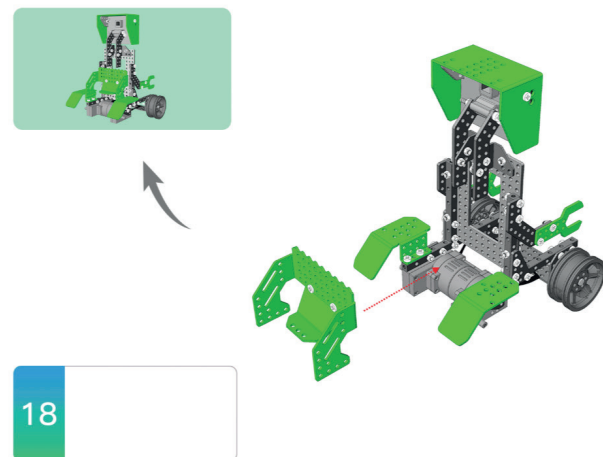
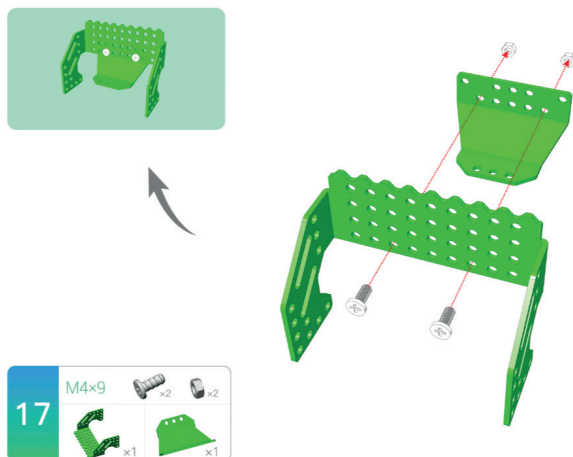
MODELO CAVALIER

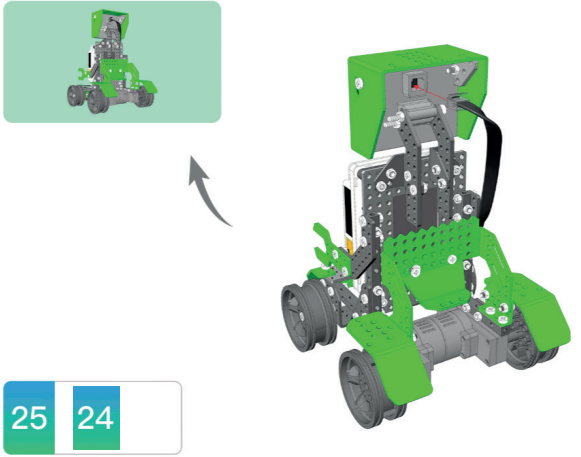
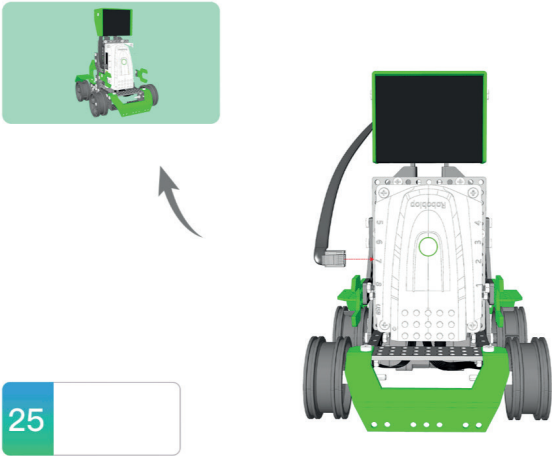
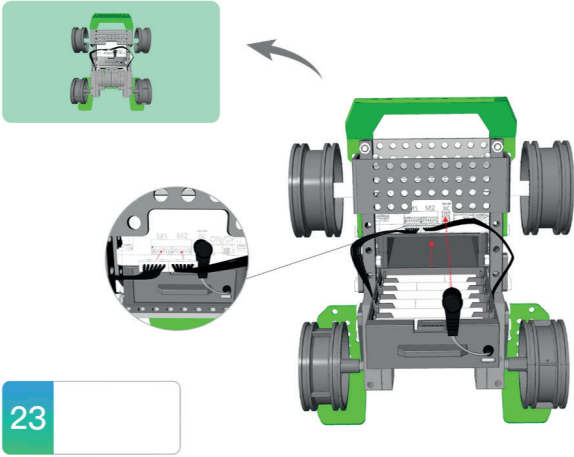




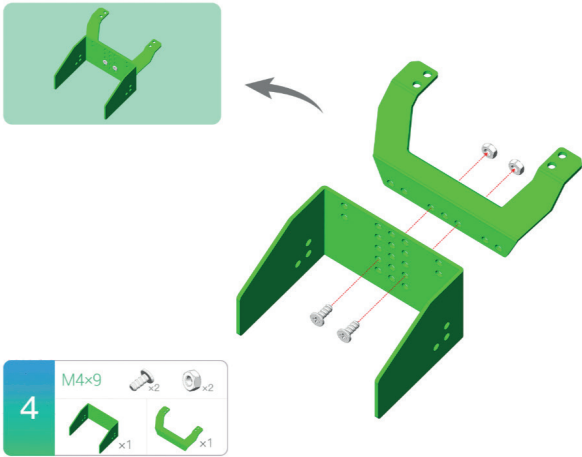
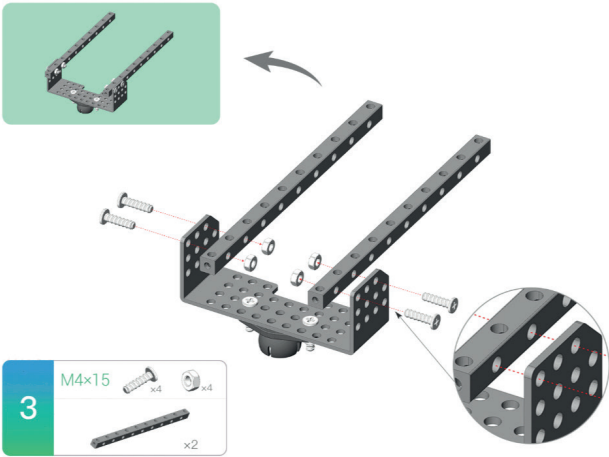
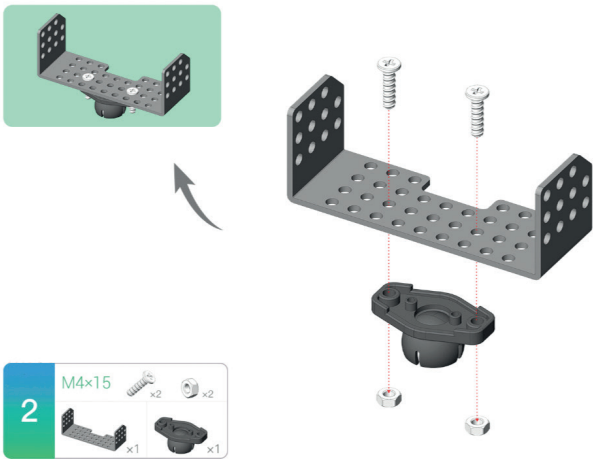
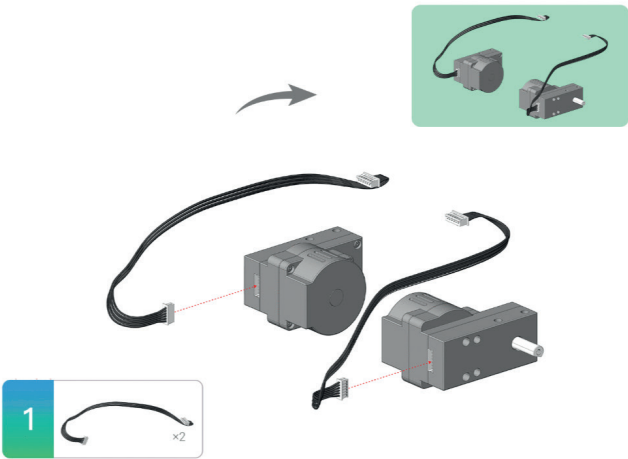
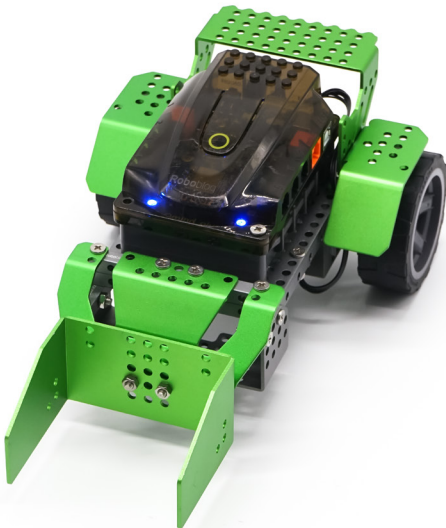


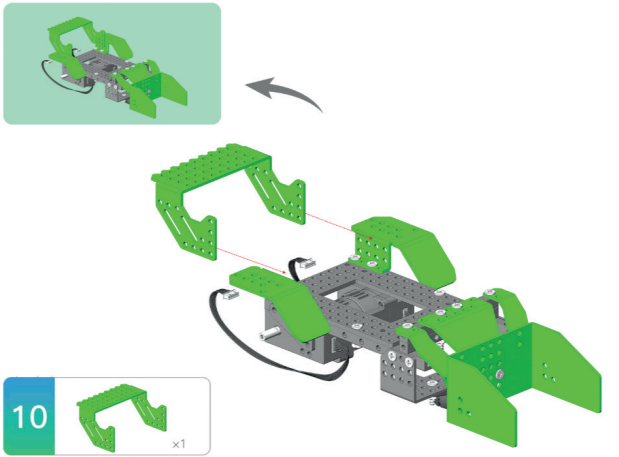
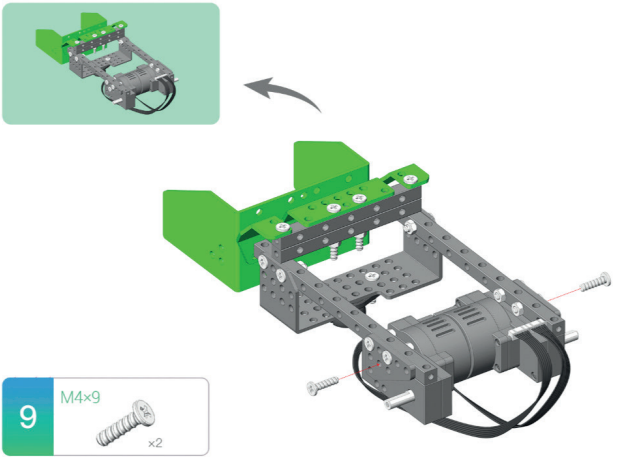
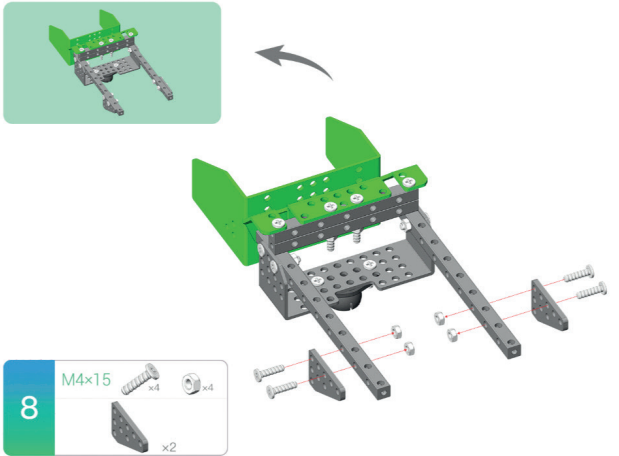
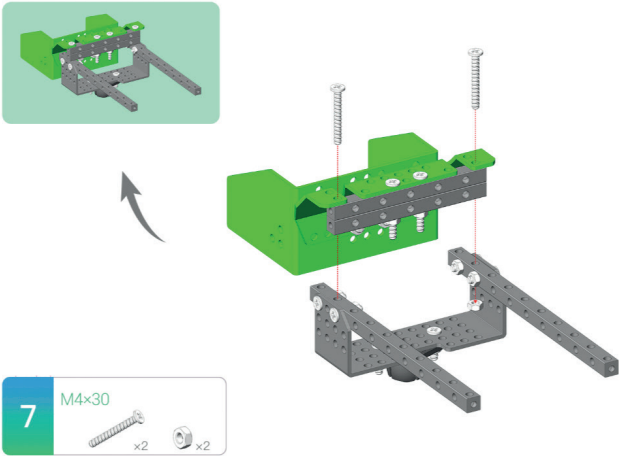
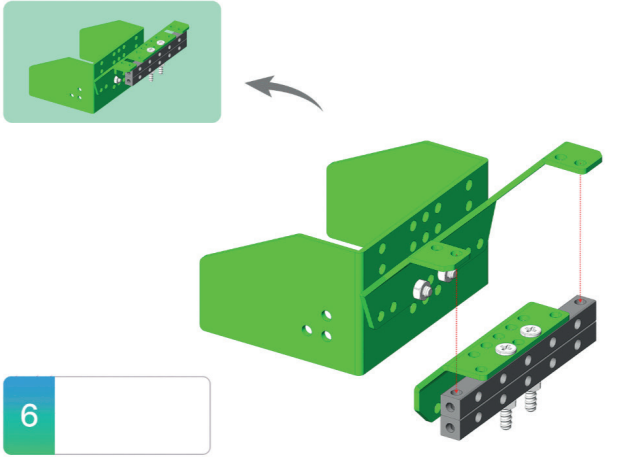
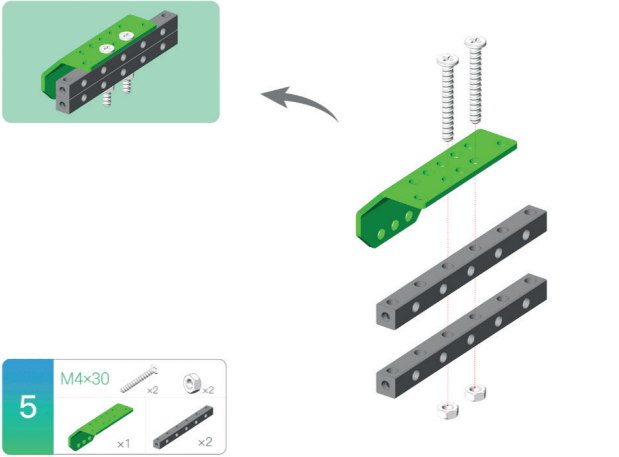
APRENDER CONECTADOS

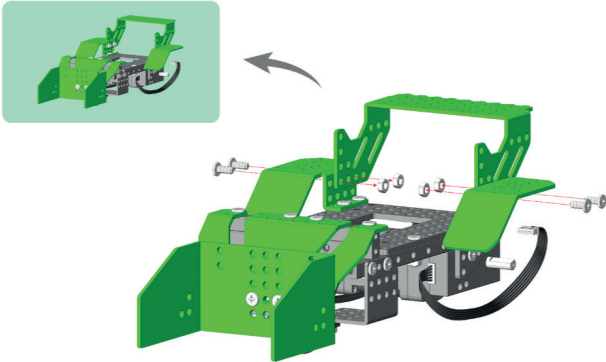




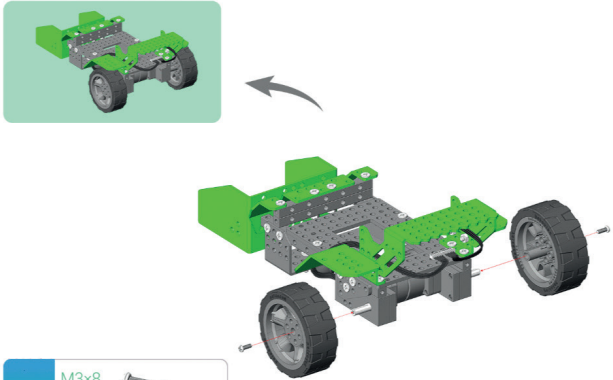
MODELO *THE DOZER*



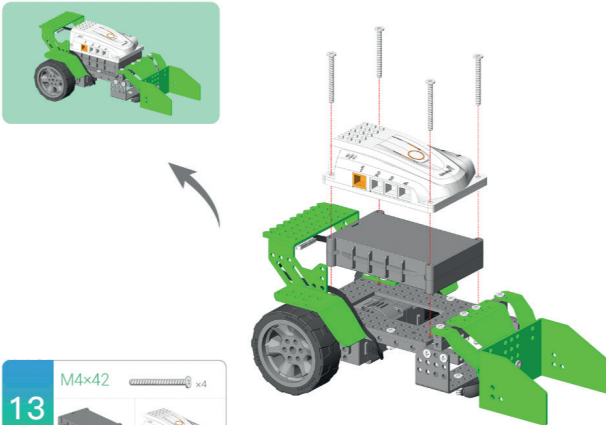




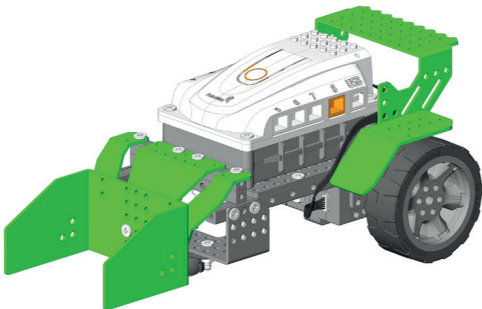
11 M4x9
x4 x4



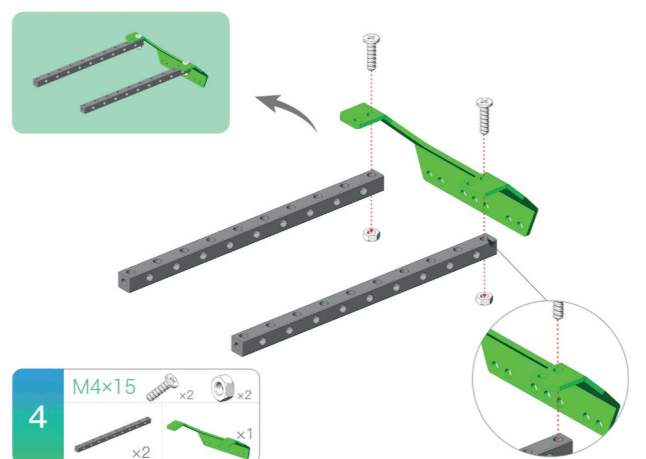
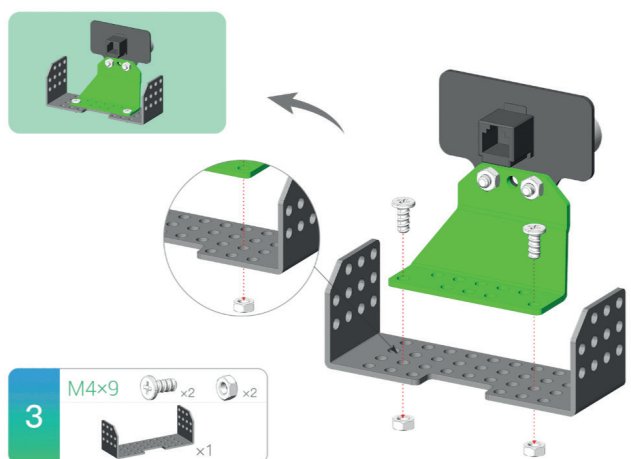
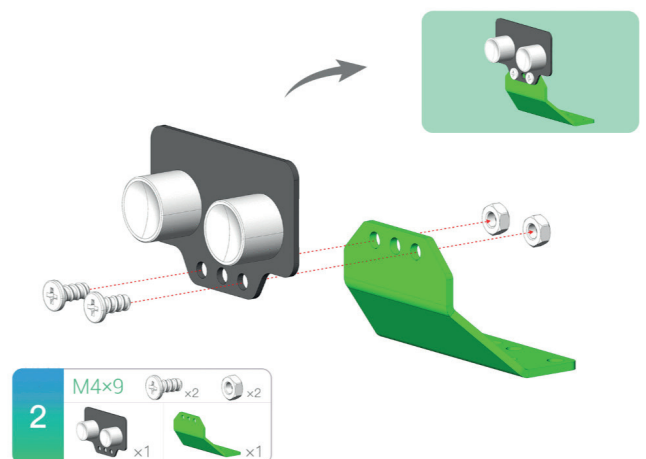
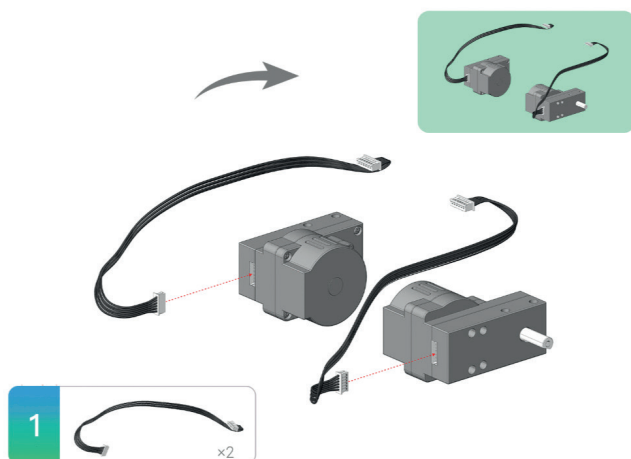
12 M3x8
x2 x2



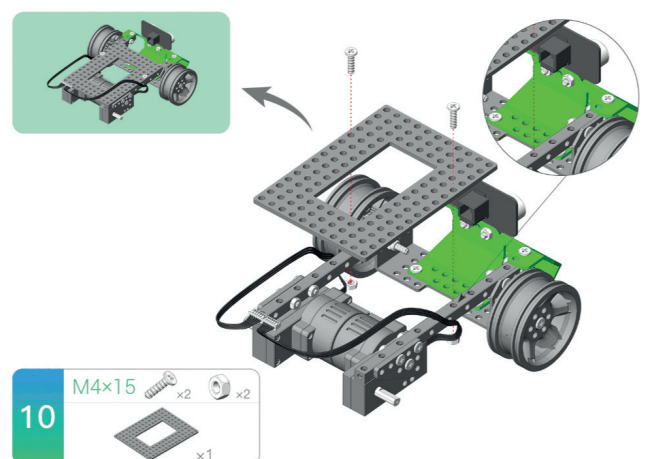
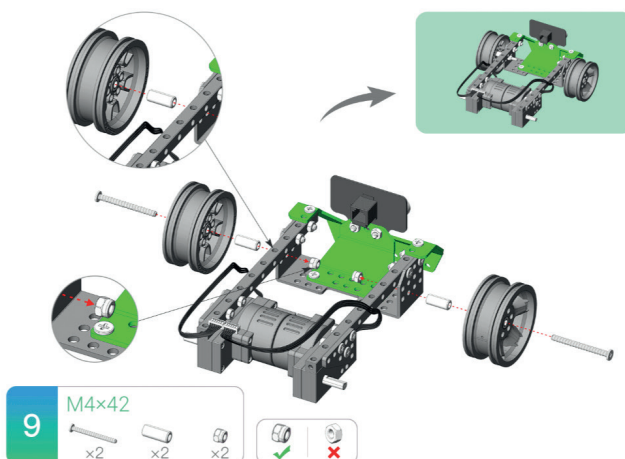
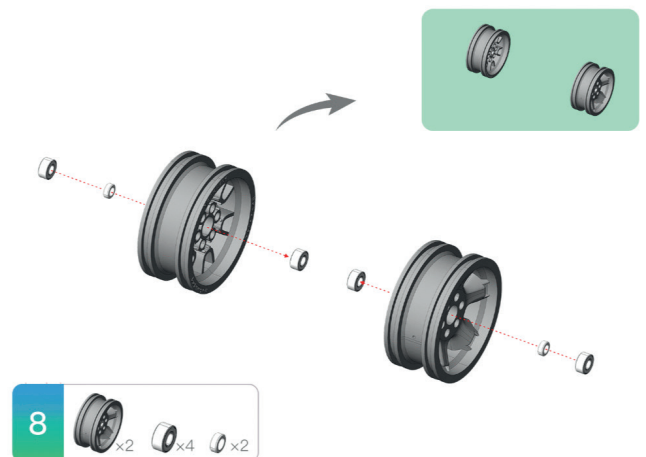
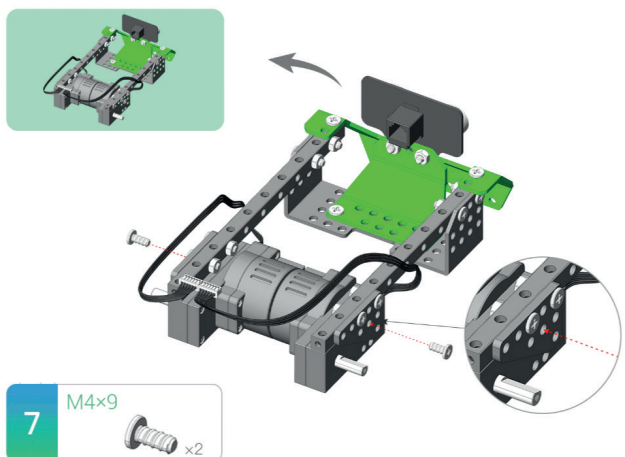
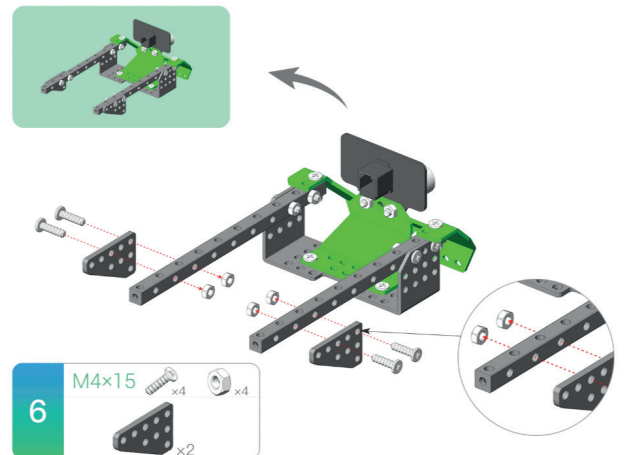
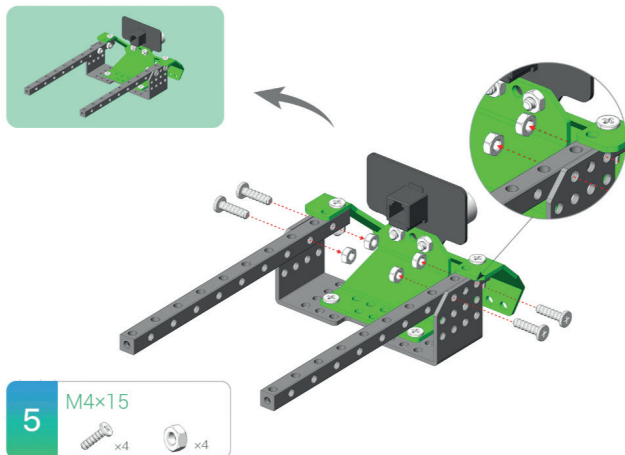
13 M4x42
x1 x1



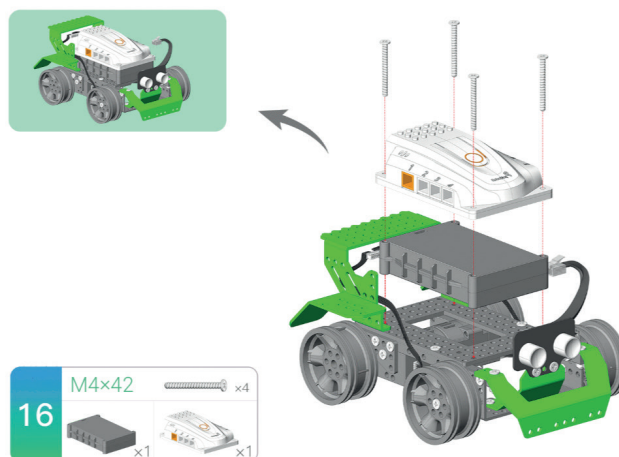
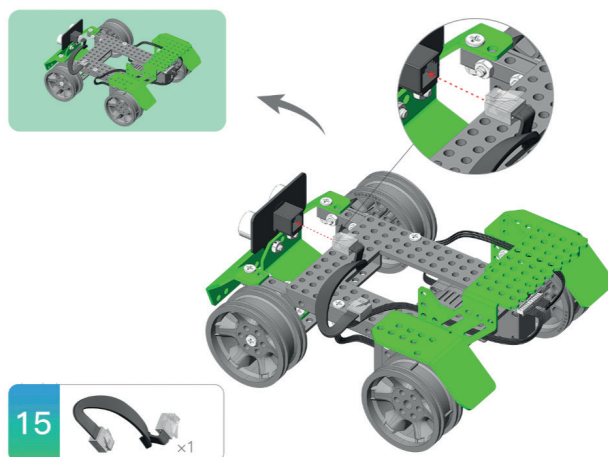
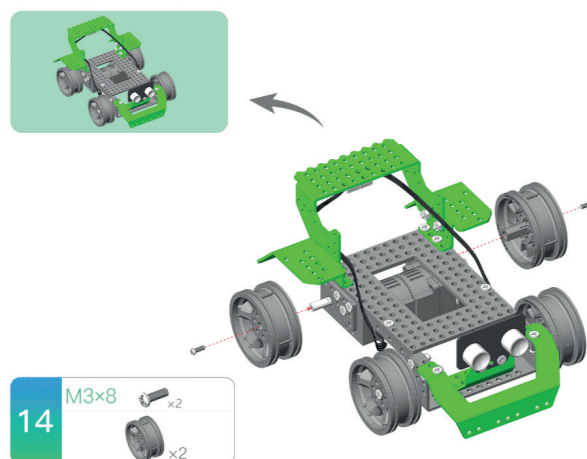
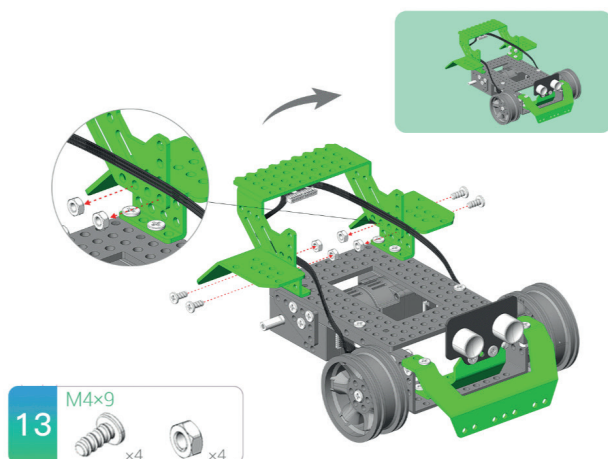
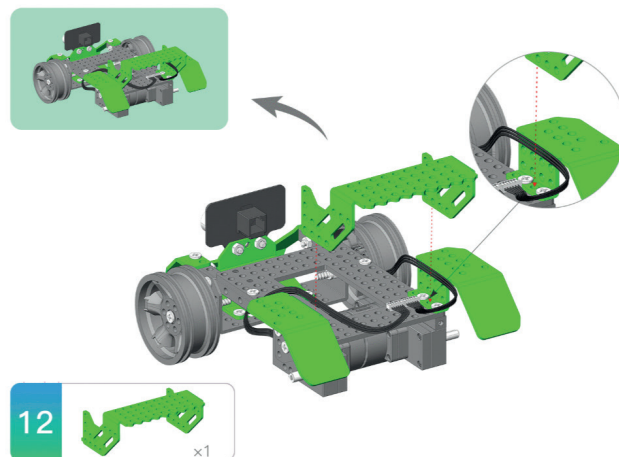
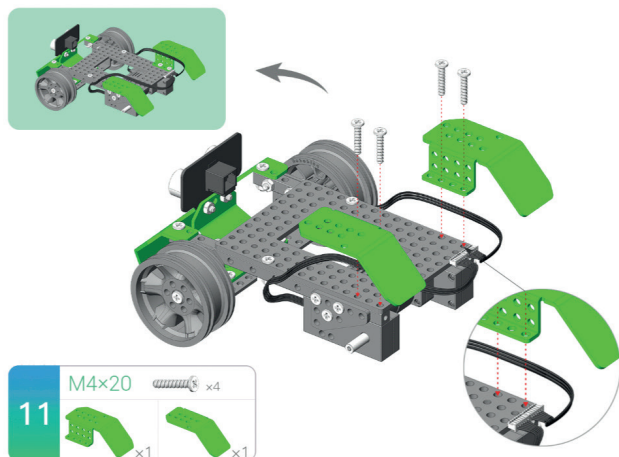
MODELO THE VOYAGER

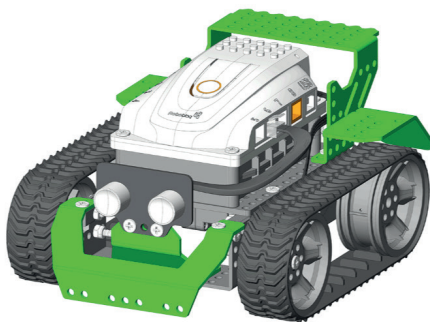
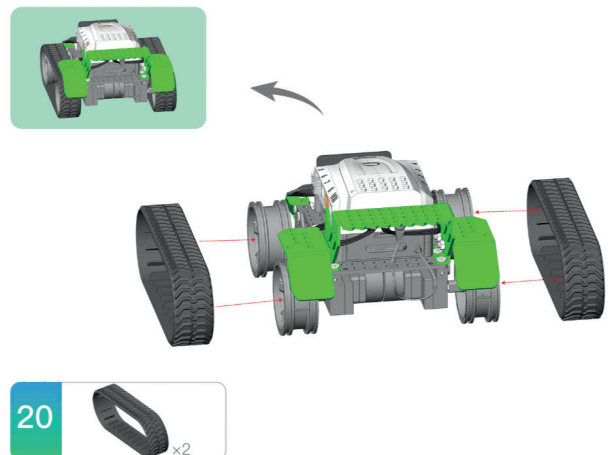
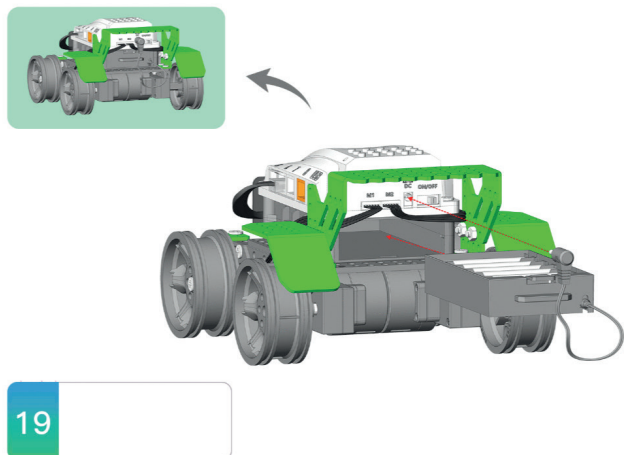
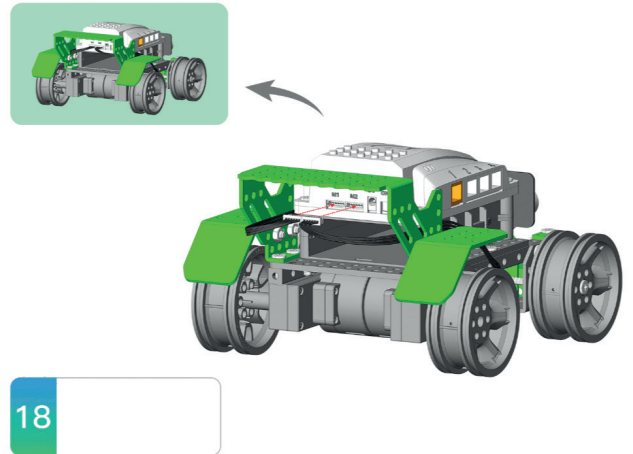
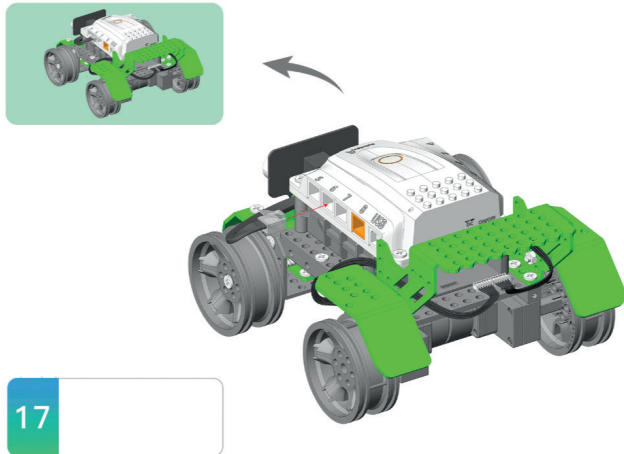


APRENDER CONECTADOS

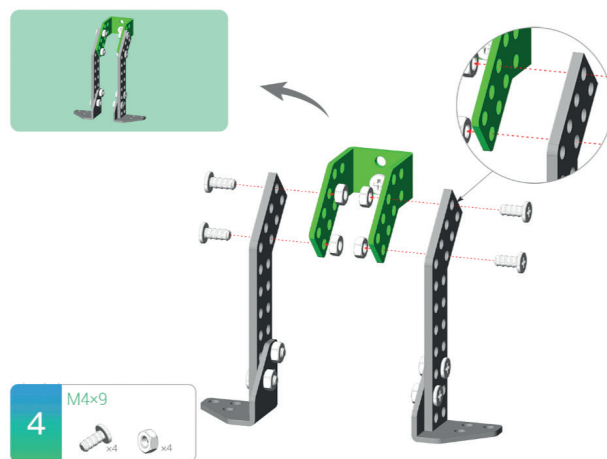
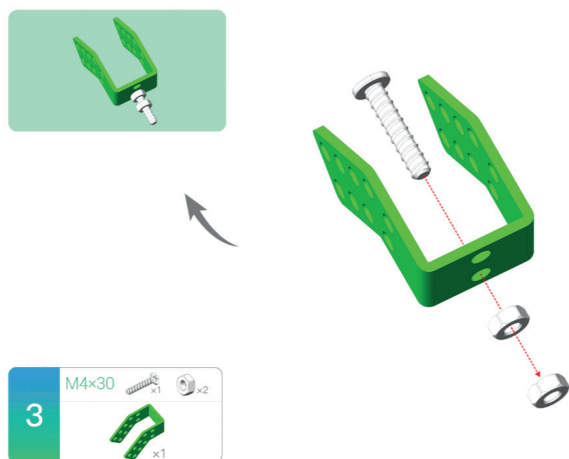
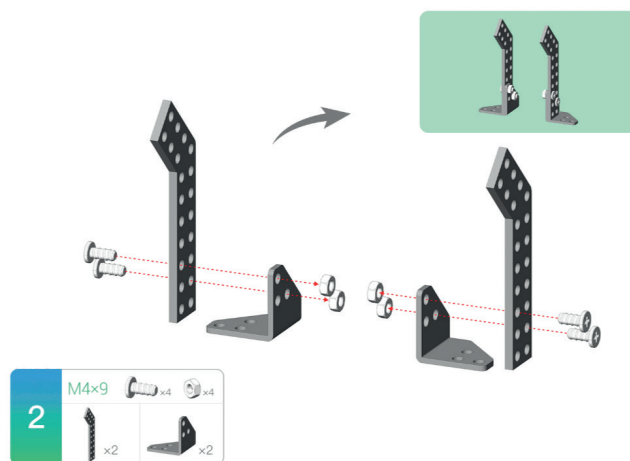
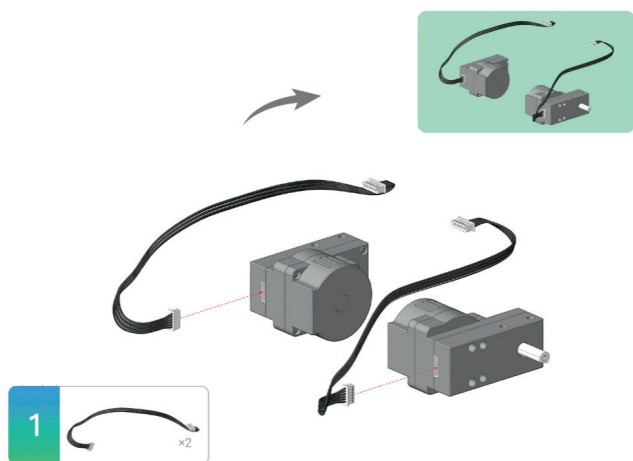
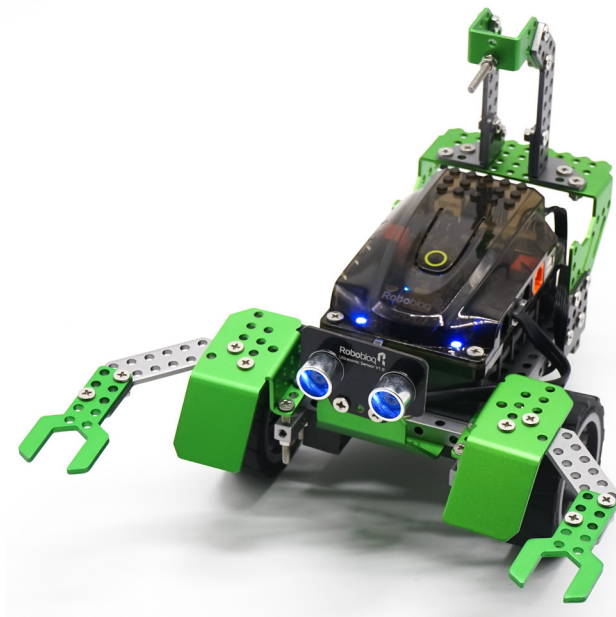


APRENDER CONECTADOS

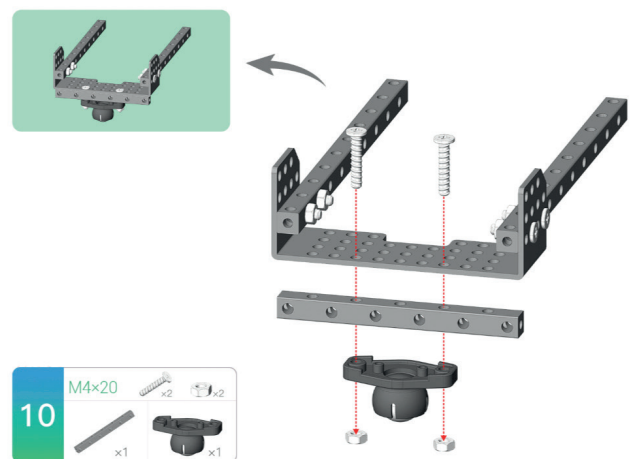
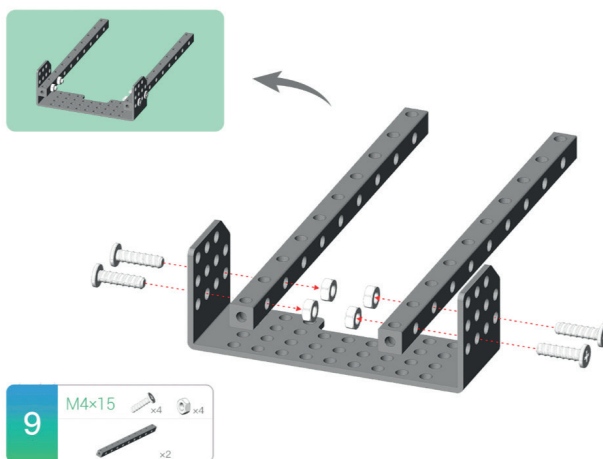
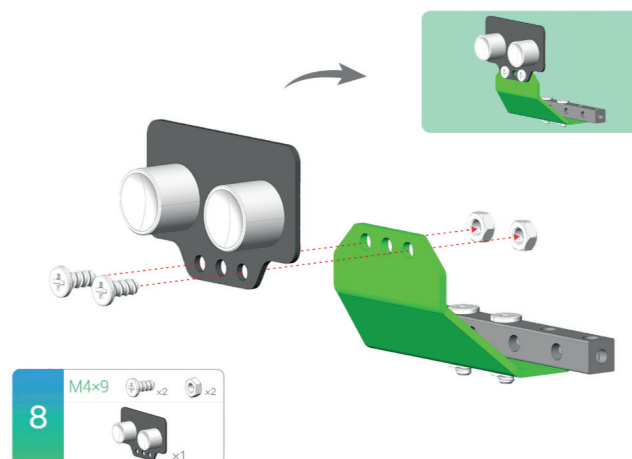
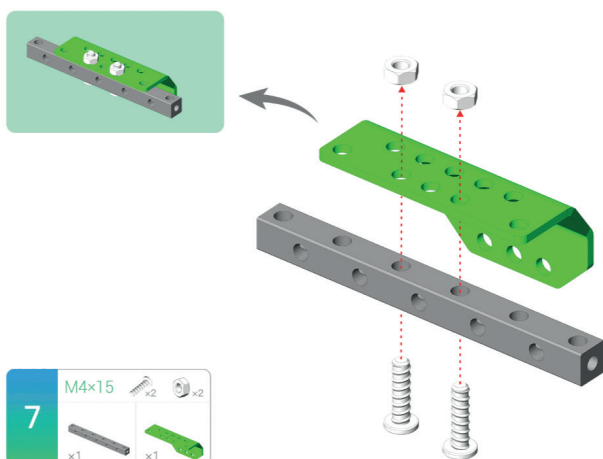
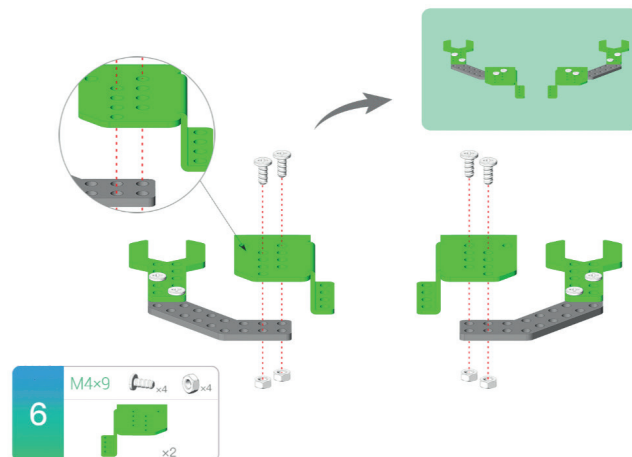
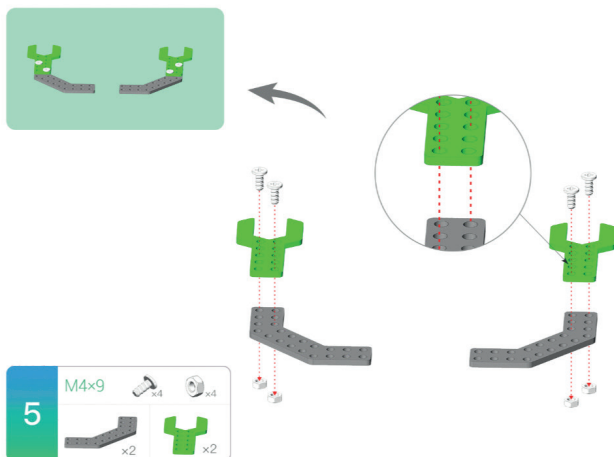




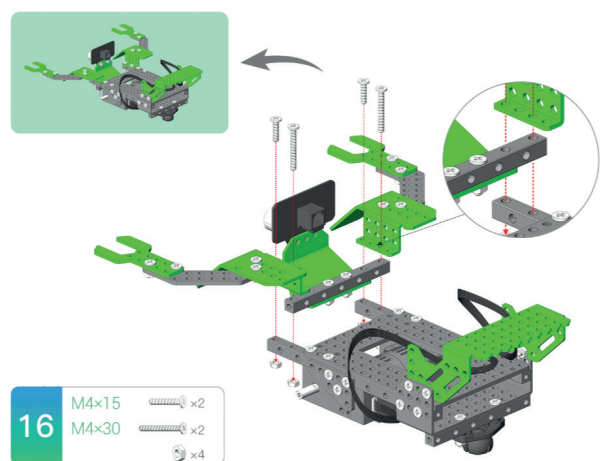
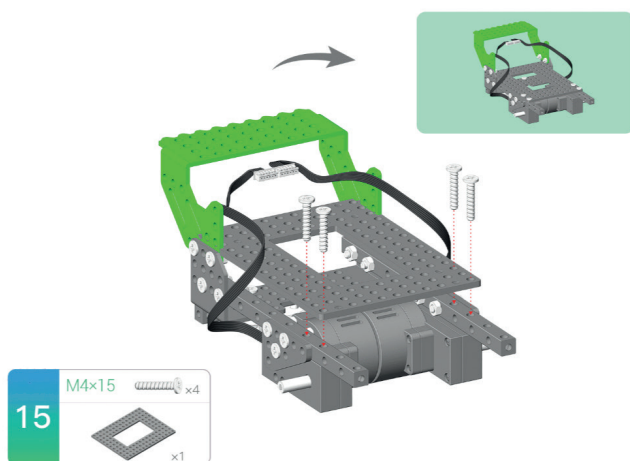
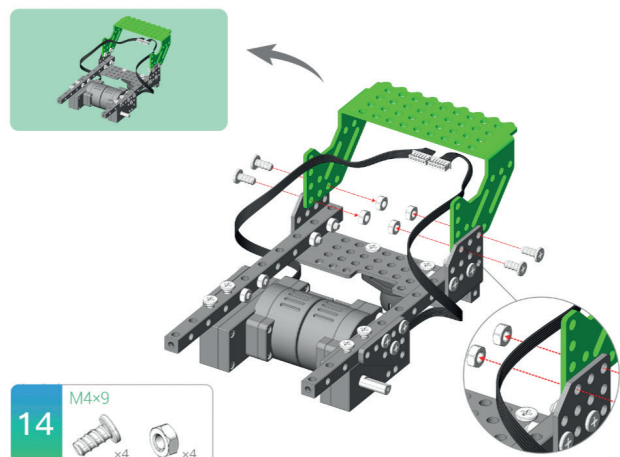
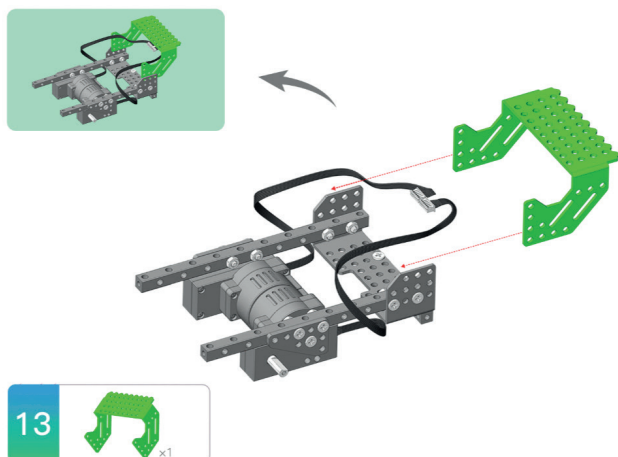
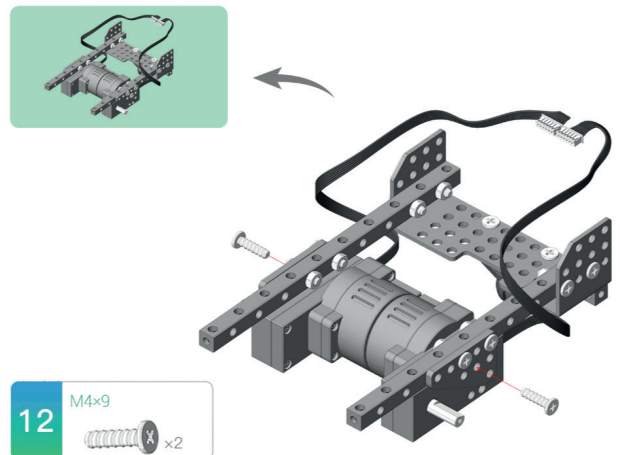
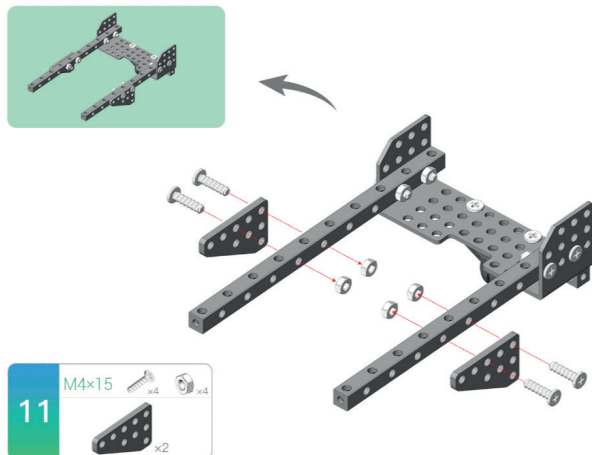
MODELO SCORPOID



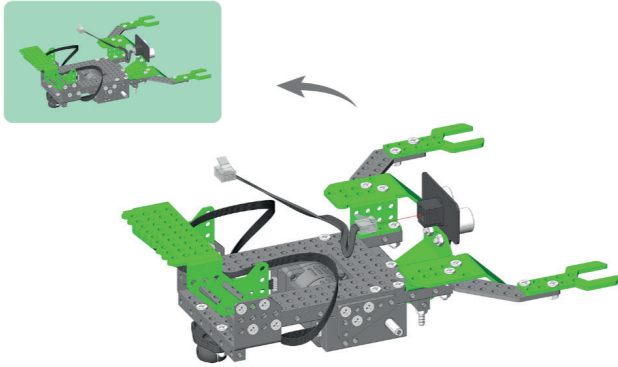
APRENDER CONECTADOS



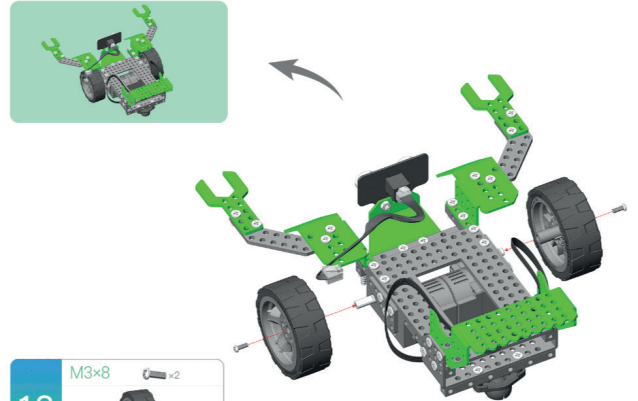
APRENDER CONECTADOS



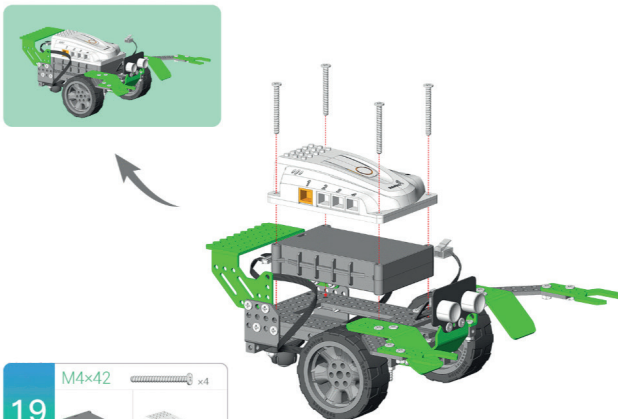
APRENDER CONECTADOS



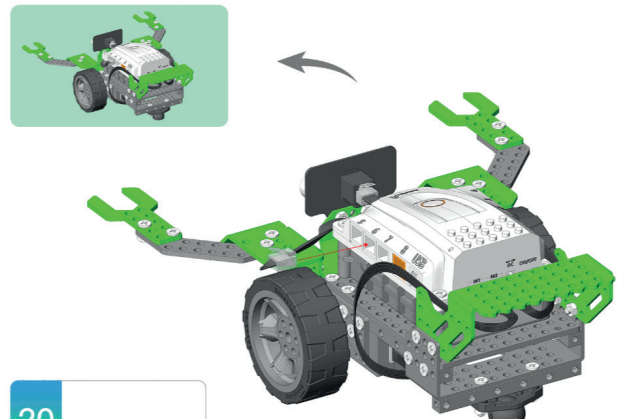
17



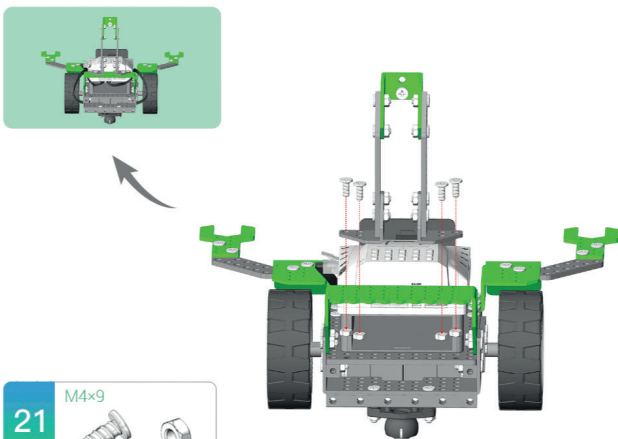
18



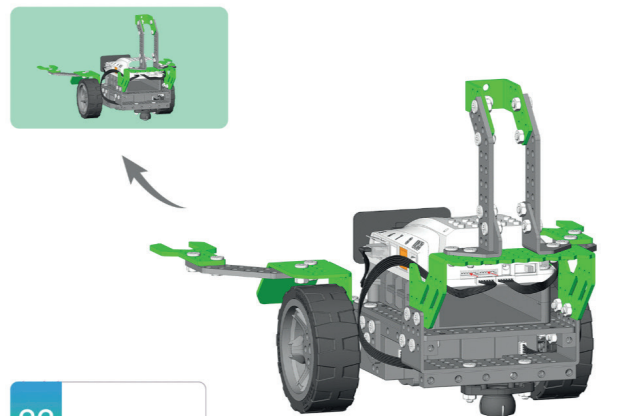
19



20



21



22



APRENDER CONECTADOS



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