

UNIDAD DIDÁCTICA

---

*Iniciación a la programación y la robótica  
mediante el Laboratorio Remoto Labsland*

---

TECNOLOGÍA

Educación Secundaria Obligatoria (ESO) y Bachillerato

Rev: 1.0 (Enero/2017)

Autor: LabsLand Experimentia S.L (contact@labsland.com)

## ÍNDICE

1.	TÍTULO DE LA UNIDAD DIDÁCTICA .....	1
2.	FUNDAMENTACIÓN Y JUSTIFICACIÓN .....	1
3.	MARCO LEGISLATIVO.....	2
4.	INTENCIONES EDUCATIVAS .....	2
4.1	Objetivos.....	2
4.2	Contenidos.....	3
4.3	Competencias.....	3
5.	TEMPORALIZACIÓN.....	4
6.	METODOLOGÍA .....	4
6.1	Fundamentos metodológicos .....	4
6.2	Principios metodológicos.....	4
6.3	Estrategias organizativas del alumnado .....	5
6.4	Equipamiento y recursos.....	6
6.5	Organización del tiempo .....	6
6.6	Organización del espacio .....	7
6.7	Transversalidad .....	7
7.	ACTIVIDADES .....	8
7.1	Introducción a la robótica básica: actuadores y sensores.....	9
7.2	Conociendo la interfaz del Laboratorio Remoto.....	12
7.3	¿Qué es un algoritmo?.....	16
7.4	Nuestro primer programa en el Laboratorio remoto: moviendo el robot.....	19
7.5	Programando los LEDs .....	28
7.6	Bucles.....	28
7.7	Sensores de infrarrojos .....	28
7.8	Condiciones .....	28
7.9	Variables.....	28
7.10	Funciones .....	28
7.11	Seguir la línea .....	28
7.12	Salir del laberinto .....	28
8.	EVALUACIÓN.....	28
8.1	Criterios de evaluación.....	28
8.2	Procedimientos e Instrumentos de Evaluación.....	28
9.	CONCLUSIÓN .....	28
10.	BIBLIOGRAFÍA Y WEBGRAFÍA.....	28

## 1. TÍTULO DE LA UNIDAD DIDÁCTICA

*“Iniciación a la programación y la robótica mediante el Laboratorio Remoto Labsland”*

## 2. FUNDAMENTACIÓN Y JUSTIFICACIÓN

El aprendizaje de la programación es prioritariamente práctico y se obtiene mediante el ejercicio. A través del esfuerzo requerido para discernir la forma más adecuada para resolver cada problema, se adquiere la habilidad para utilizar eficientemente los distintos elementos de código disponibles. Es por ello que no se puede plantear este tipo de estudio de forma puramente teórica, sino que es necesario proponer un objetivo claro hacia el que los alumnos puedan orientar su labor.

Esta Unidad Didáctica está dirigida al alumnado de las etapas de Educación Secundaria Obligatoria (ESO) y Bachillerato, aunque podría ser susceptible de uso en otros cursos escolares teniendo en cuenta la novedad de sus aprendizajes. Entre las asignaturas que más se relacionan con los contenidos destacan Ciencia, Tecnología, Ingeniería y Matemáticas – disciplinas académicas conocidas con el acrónimo CTIM o STEM en lengua anglosajona. A su vez, también tienen cabida el resto de las asignaturas, como son la enseñanza de lenguas extranjeras, lengua castellana, educación artística, por su gran componente transversal.

En esta Unidad se plantea fijar como objetivo principal el conocimiento de la robótica básica, desarrollando actividades de control de robots que utilicen de forma acumulativa las distintas herramientas que proporcionan los lenguajes de programación. La organización de las actividades se basa en esta progresión.

Una vez impartidos unos fundamentos básicos de robótica y algoritmia, se familiariza al alumnado con la interfaz de programación online usada para los experimentos prácticos y se proponen algunas formas de interacción simple con los sistemas actuadores de los dispositivos disponibles. A continuación se explican los bucles de control, de forma que se pueda utilizar adecuadamente los sistemas sensoriales, seguidos de las condiciones, variables y funciones, requeridas para producir programas complejos de control.

La unidad termina con dos proyectos de programación de robots, para afianzar los conocimientos y poner a prueba las habilidades adquiridas por los alumnos: en primera instancia se genera el sistema de control de un robot que sigue un recorrido simple y, aumentando la complejidad del código correspondiente, se permite encontrar la salida de un laberinto. Al finalizar este trabajo, se habrá logrado que los alumnos no sólo hayan adquirido nociones de programación básica, sino que además hayan podido observar su utilidad en un entorno práctico.

### 3. MARCO LEGISLATIVO

Actualmente en España conviven dos ordenaciones de enseñanza en la etapa de Educación Secundaria Obligatoria, que se distribuyen del siguiente modo:

- En Educación Secundaria Obligatoria:
  - El 1º y 3º curso de Educación Secundaria Obligatoria en España obedecerán a la Ley Orgánica 8/2013 de 9 de diciembre para la Mejora de la Calidad Educativa (LOMCE)
  - El 2º y 4º curso de Educación Secundaria Obligatoria en España obedecerán a la Ley Orgánica 2/2006 de 3 de mayo de Educación (LOE)
- En Bachillerato:
  - El 1º curso de Bachillerato en España obedecerá a la Ley Orgánica 8/2013 de 9 de diciembre para la Mejora de la Calidad Educativa (LOMCE)
  - el 2º curso de Bachillerato en España obedecerá a la Ley Orgánica 2/2006 de 3 de mayo de Educación (LOE)
- Educación Secundaria Obligatoria y Bachillerato:
  - Real Decreto 1105/2014 de 26 de diciembre, por el que se establece el currículo básico de la Educación Obligatoria y del Bachillerato (BOE n.º 3, de 3 de enero de 2015)
  - Orden ECD/1361/2015 de 3 de julio, por la que se establece el currículo de Educación Secundaria Obligatoria y Bachillerato para el ámbito de gestión del Ministerio de Educación, Cultura y Deporte y se regula su implantación, así como la evaluación continua y determinados aspectos organizativos de las etapas

Es importante considerar que Cada Comunidad Autónoma deberá atender a la legislación vigente en materia educativa que le corresponda para llevar a cabo su propia programación didáctica y adaptarla de forma efectiva y eficiente.

### 4. INTENCIONES EDUCATIVAS

#### 4.1 Objetivos

Esta Unidad Didáctica contribuirá a alcanzar el siguiente *objetivo general*:

- *Desarrollar en los y las estudiantes capacidades cognitivas de análisis crítico y habilidades de ingeniería mediante la enseñanza de conocimientos de programación y el ensamblaje de robots a través de un lenguaje de programación por bloques.*

En cada una de las actividades que conforman esta unidad didáctica se detallarán los objetivos específicos a los que cada una contribuye, con el propósito de lograr la adquisición del objetivo general.

## 4.2 Contenidos

Los contenidos que se trabajarán en esta Unidad Didáctica son:

1. Conocimiento de un robot y fundamentos básicos de la robótica
2. Interfaz del laboratorio remoto
3. Algoritmos y diagramas de flujo
4. Nuestro primer programa en el Laboratorio remoto: moviendo el robot
5. Unidad LED
6. Bucles
7. Sensores de infrarrojo
8. Condiciones
9. Variables
10. Funciones
11. Robot sigue-línea
12. Laberintos en robótica

En cada actividad se especifican los contenidos específicos correspondientes.

## 4.3 Competencias

Atendiendo a la *Ley Orgánica 2/2006, de 3 de mayo, de Educación (LOE)* esta Unidad Didáctica contribuirá, por orden de prioridad, a la consecución de las siguientes competencias:

- Competencia en tratamiento de la información y competencia digital
- Competencia para aprender a aprender
- Competencia en autonomía e iniciativa personal
- Competencia matemática
- Competencia en el conocimiento y la interacción con el mundo físico
- Competencia en comunicación lingüística
- Competencia social y ciudadana
- Competencia cultural y artística

Atendiendo a la *Ley Orgánica 8/2013, de 9 de diciembre, de Educación (LOMCE)* esta Unidad Didáctica contribuirá, por orden de prioridad, al logro de las siguientes competencias básicas:

- Competencia digital
- Aprender a aprender
- Sentido de la iniciativa y espíritu emprendedor
- Competencia matemática y competencias básicas en ciencia y tecnología
- Comunicación lingüística
- Competencias sociales y cívicas
- Conciencia y expresiones culturales

## 5. TEMPORALIZACIÓN

La presente Unidad Didáctica, como se ha mencionado con anterioridad, se trabajará principalmente en las asignaturas de Ciencia, Tecnología, Ingeniería y Matemáticas (CTIM), siendo posible su uso también en cualquier otra.

Se organizará en 12 actividades teórico-prácticas y se estima que su duración total sea de aproximadamente 24 horas. Puesto que se prevé impartir la asignatura de informática durante 3 horas a la semana, el contenido propuesto se extendería a 8 semanas de clase.

Recuerda que este es un documento demostrativo. Si quieres conocer la distribución temporal del curso completo de robótica, ponte en contacto con nosotros en [contact@labsland.com](mailto:contact@labsland.com)

El reparto de actividades se ha realizado de esta manera debido a que los conceptos se asimilan mejor cuando se imparten en ese orden. Se pretende partir de los métodos más generales y sencillos para finalmente llegar al objetivo final que es la programación de un robot que es capaz de salir de un laberinto.

## 6. METODOLOGÍA

### 6.1 Fundamentos metodológicos

El planteamiento metodológico que se llevará a cabo en esta Unidad Didáctica buscará la integración de contenidos científicos, tecnológicos y organizativos, la capacidad de autoaprendizaje y la capacidad para trabajar en equipo. Para alcanzar los objetivos educativos propuestos en el módulo, la metodología que se plantea perseguirá, principalmente, fomentar la motivación y despertar el interés de los alumnos y alumnas hacia los contenidos correspondientes y citados con anterioridad.

### 6.2 Principios metodológicos

La existencia de unos principios metodológicos y unas estrategias se justifica porque debemos partir de una forma de proceder didáctica, que facilite de manera activa, significativa, globalizadora, constructiva y socializadora; la funcionalidad del proceso de enseñanza

---

aprendizaje, bajo un clima de afecto y confianza, y conforme a un entorno creativo, en el que se favorezca la igualdad de oportunidades entre el alumnado.

Algunos de estos *principios metodológicos* son:

- Favorecer un aprendizaje significativo y progresivo, partiendo de lo que se domina hasta alcanzar las competencias requeridas
- Potenciar un aprendizaje variado mediante la utilización de diferentes técnicas tecnológicas y la variación de actividades prácticas
- Ordenar las actividades didácticas de forma progresiva, de manera que los alumnos y las alumnas tengan una guía sistemática y coherente que favorezca la consecución de los aprendizajes
- Fomentar el trabajo cooperativo y colaborativo, cuando se pueda, para que el alumnado se familiarice con las tareas compartidas, la dedicación de un tiempo, unos espacios y con unos procedimientos para llegar a acuerdos en la programación y elaboración de proyectos
- Motivar la actividad, la investigación y la experimentación en el alumnado de modo que pueda ir construyendo sus propios esquemas de conocimiento activamente a partir de lo que ya conoce y lo que se le ofrece, dejando a un lado la mera contemplación ofreciéndoles estrategias que les ayuden a ser creativos/as, potencien su imaginación y la capacidad de observación
- Es importante evaluar continua, global e individualmente el proceso de enseñanza-aprendizaje y facilitar la participación de los alumnos y alumnas en el proceso de evaluación de todas las actividades realizadas

Se recomienda al profesorado actuar como guía del proceso de aprendizaje, facilitando los recursos necesarios a la vez que asesora y orienta la adquisición de conocimientos, habilidades, destrezas y actitudes relacionadas con la programación y el ensamblaje de robots.

### **6.3 Estrategias organizativas del alumnado**

Se recomienda organizar el aula de manera flexible para propiciar actividades de enseñanza-aprendizaje y evaluación autónomas, así como compartidas con el grupo-clase. Para ello se utilizarán agrupamientos de diversos tipos, teniendo en cuenta el número de personas que cohabiten en el espacio:

- **Gran grupo**

Podría ser interesante este modelo organizativo porque permite que todo el alumnado coopere para realizar producciones que necesiten de implicación conjunta. A su vez, al

---

presentar los proyectos de robótica todos y todas podrían sugerir propuestas de mejora contribuyendo a la generación de *feedback* satisfactorio.

- **Pequeño grupo**

Se podrían formar grupos de 3 y/o 4 personas para desarrollar actividades colectivas fomentando el trabajo cooperativo y colaborativo. De este modo, surgirían mayor cantidad de opiniones e ideas a la hora de realizar los diversos proyectos utilizando la técnica del *Brainstorming* o lluvia de ideas.

- **Individuales**

Este tipo de actividades se podrían realizar para una mayor introspección del alumnado. Además, en los casos en los que no sea posible acceder al laboratorio remoto de forma conjunta, se recurrirá a este estilo de agrupamiento.

#### 6.4 Equipamiento y recursos

Con el propósito de enriquecer e incentivar el desarrollo de las actividades, en esta Unidad Didáctica nos serviremos de los siguientes recursos:

- **Recursos humanos**

- Docente
- Alumnado y sus familias

- **Recursos técnicos y materiales**

- Material fungible (papel, lápiz, etc.)
- Ordenadores con conexión a Internet
- Cuenta Labsland

- **Recursos del entorno**

- Centro educativo y aula ordinaria. También espacios personales como la vivienda familiar que disponga de conexión a Internet

#### 6.5 Organización del tiempo

Atendiendo a la dificultad de cada una de las actividades y al tiempo estimado para la realización de cada una de ellas, la organización temporal propuesta es la siguiente:

Actividad	Teoría	Ejercicios
1	40 min	1 hora y 20 min
2	50 min	1 hora y 10 min

3	50 min	1 hora y 10 min
4	1 hora	1 hora
5	25 min	35 min
6	1 hora	1 hora
7	30 min	45 min
8	1 hora	1 hora
9	30 min	1 hora
10	1 hora y 10 min	1 hora y 20 min
11	1 hora y 20 min	1 hora y 20 min
12	1 hora y 30 min	1 hora

## 6.6 Organización del espacio

La organización espacial va a influir decisivamente en la forma de realizar las actividades. En un aula, debe ser flexible y conforme a las necesidades que vayan surgiendo.

Los únicos *espacios de trabajo* requeridos para las actividades propuestas en esta Unidad Didáctica, dentro del centro educativo, serán:

- **Espacio Ordinario:** El aula de clase, para los ejercicios teóricos y manuales
- **Espacio Específico:** El aula de informática, desde el que los alumnos puedan conectarse al laboratorio remoto online de Labsland

## 6.7 Transversalidad

Uno de los principios y fines de la educación según la Ley 2/2006, de 3 de mayo, de Educación (LOE) y la Ley Orgánica 8/2013, de 9 de diciembre, para la mejora de la calidad educativa (LOMCE) que se citan con anterioridad, es la educación en valores como la libertad personal, la responsabilidad, la ciudadanía democrática, solidaridad, tolerancia, justicia, esfuerzo, cooperación, colaboración, etc. Los temas transversales contribuyen de manera especial a la educación de valores morales y cívicos, entendida ésta como una educación al servicio de la formación de personas capaces de construir racional y autónomamente su propio sistema de valores y, a partir de ellos, capaces también de enjuiciar críticamente la realidad que les ha tocado vivir, e intervenir para conseguir su transformación y mejora (Temas transversales y desarrollo curricular, MEC, 1.993).

Los temas transversales que se trabajarán en esta Unidad Didáctica, a parte de los ya citados, son educación para el consumidor y usuario, educación para la convivencia y paz, educación

para los derechos humanos, educación para la salud, educación ambiental y educación para la igualdad de trato entre los sexos. Esto es así porque no se trata únicamente de que el alumnado aprenda a programar, sino también de que adquiera valores para trabajar en equipo, lograr respetar a sus iguales, crear proyectos que contribuyan a una mejora social y desarrollar sus habilidades creativas, de forma que se aprovechen todos los contenidos para la creación de seres independientes y autónomos en su proceso de aprendizaje.

Sería interesante proponer al alumnado temáticas transversales, de forma que contextualice los distintos proyectos relacionados con la robótica. De este modo, el alumnado se podría sentir más motivado sabiendo que su producto podrá tener una utilidad y repercusión social.

## 7. ACTIVIDADES

Las pautas específicas para organizar la distribución de las actividades se hará tomando como base la secuencia de actividades de *Merrill* (activación, demostración, aplicación e integración) pero, principalmente, atendiendo a las diferentes etapas del proceso de aprendizaje.

Cada una de las actividades se distribuye en *actividades iniciales*, donde se realizarán actividades introductorias, de orientación y motivación, de detección y evaluación de conocimientos e ideas previas con la finalidad de intentar modificar estereotipos y lograr un aprendizaje significativo. También, en *actividades de profundización*, donde se expondrán los contenidos que se tratarán en cada momento, empleando los medios disponibles y aplicando una metodología activa además de proponer actividades individuales y/o grupales, orientadas a afianzar lo explicado. Por último, se plasman *actividades de consolidación*, que tendrán que ver con actividades de investigación, diseño y ejecución de proyectos.

## 7.1 Introducción a la robótica básica: actuadores y sensores

### Objetivos específicos:

- Conocer los componentes internos del Robot y su correspondiente uso.
- Conocer las diferencias entre los dos tipos principales de Robots.

### Contenidos específicos:

- Definición de Robot
- Origen del término Robot
- Sistemas que componen un Robot (control, sensorial, actuación)
- Tipos principales del Robot (manipulativos, móviles)

### Definición de Robot

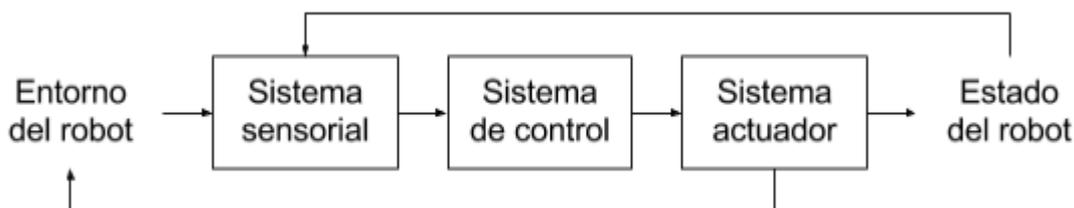
Se conoce como "robot" a toda máquina programable capaz de realizar una o más actividades de forma autónoma. Debe tener la capacidad de recoger información e interactuar hasta cierto punto con su entorno, para modificar su conducta y así tomar decisiones sobre las funciones a realizar.

La palabra "robot" proviene del checo "robota", que significa "trabajo duro o forzado". Fue creada por el escritor Karel Čapek en su obra de teatro "Robots Universales Rossum" (1920), aunque su uso se extendió gracias a las obras de ciencia ficción del escritor estadounidense Isaac Asimov.

### Sistemas principales del Robot

Todo robot consta de tres sistemas principales:

- El sistema de control, generalmente en forma de ordenador auxiliar, que le permite interpretar los datos percibidos y tomar las acciones pertinentes en función de los mismos,
- El sistema sensorial, que engloba todos aquellos dispositivos anexos que le permitan recabar información acerca de su entorno y su propio estado,
- El sistema de actuación, al que pertenecen todos aquellos elementos físicos del robot que le permiten realizar acciones sobre su entorno o sobre sí mismo.



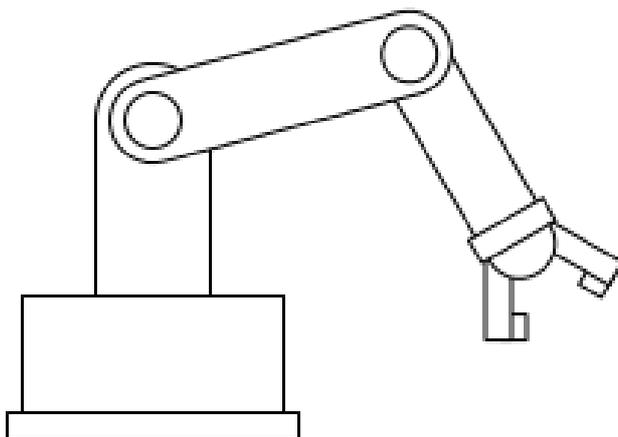
Por simplicidad, la estructura general y el comportamiento de un robot pueden compararse con los de un ser humano. Así, el sistema de control de una persona correspondería a su cerebro, que procesaría la información del sistema sensorial (los sentidos, receptores de dolor, niveles hormonales, etc.) y controlaría la actividad del sistema actuador (movimiento de las articulaciones y todas las funciones naturales de los órganos del cuerpo). En temas posteriores se observará que los métodos de recogida de información y toma de decisiones de los robots son igualmente similares a los de un ser humano.

### Tipos principales de Robot

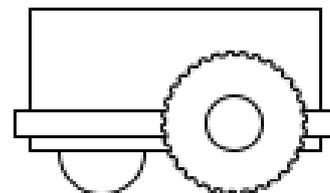
Existen dos tipos principales de robots:

- Los **robots manipuladores** se encuentran normalmente anclados a una superficie, pero están formados por una cadena de articulaciones que les permite alcanzar y realizar acciones en puntos concretos de sus proximidades. Estos son los robots que suelen encontrarse en las cadenas de montaje de las fábricas. El elemento final de un robot manipulador, conocido como "efector final", es con frecuencia intercambiable para poder realizar distintas acciones, tales como recoger objetos, pintar o soldar.
- Los **robots móviles** no disponen de efectores finales, pero tienen la capacidad de desplazarse por su entorno, generalmente mediante ruedas. Normalmente necesitan un sistema sensorial más preciso y exhaustivo que los manipuladores, ya que su área de trabajo puede resultar menos predecible que la correspondiente a una cadena de montaje.

Por supuesto, ambos tipos de robots podrían combinarse entre sí para producir una máquina más compleja, pero desde el punto de vista de la programación, cada parte sería independiente.



Robot manipulador



Robot móvil

### PROPUESTA DE EJERCICIOS

**Motivación:** Se propone a los alumnos pensar en robots ficticios que hayan visto en el cine y la televisión y analicen qué partes de ellos corresponderían a la robótica de manipulación y qué partes a la robótica móvil, teniendo en cuenta que algunas partes podrían formar parte de ambas modalidades o de ninguna de ellas.

Agrupamiento: Toda la clase  
Recursos: No se requieren  
Espacio: Aula de clase  
Tiempo: 20 minutos

*Solución:* Por lo general, toda parte de un robot que tenga articulaciones y pueda desempeñar una actividad concreta, aunque esta sea tan simple como la de una cámara de orientación variable, son manipuladores, mientras que todo elemento que permita al robot cambiar su posición en el espacio pertenece a la robótica móvil. Las piernas de los robots humanoides serían un ejemplo de ambas modalidades, ya que tienen articulaciones que hay que programar pero producen un desplazamiento.

**Desarrollo:** Se propone entonces a los alumnos usar la imaginación para diseñar sus propios robots, combinando elementos de manipulación y móviles.

Agrupamiento: Individual  
Recursos: Papel y lápiz  
Espacio: Aula de clase  
Tiempo: 30 minutos

*Solución:* Puesto que los alumnos probablemente se basarán sobre todo en elementos de ficción, no se espera que el resultado sea realista, sólo que comprendan la diferencia entre la robótica de manipulación y la móvil. De nuevo, es posible que incluyan partes que no correspondan a ninguna de las dos, pero esto es admisible siempre que sepan darse cuenta de ello o argumentar razonadamente si no están de acuerdo.

**Consolidación:** En grupos reducidos, los alumnos pondrán en común sus diseños y discutirán las características de cada uno, proponiendo entre todos mejoras y combinándolos en un único diseño, que al terminar mostrarán al resto de la clase.

Agrupamiento: Grupos reducidos (3 a 6 alumnos)  
Recursos: Papel y lápiz  
Espacio: Aula de clase  
Tiempo: 30 minutos

*Solución:* Los criterios de clasificación de elementos antes descritos se mantienen. El profesor deberá comprobar que los alumnos son capaces de identificar correctamente cada parte de sus diseños.

## 7.2 Conociendo la interfaz del Laboratorio Remoto

### Objetivos específicos:

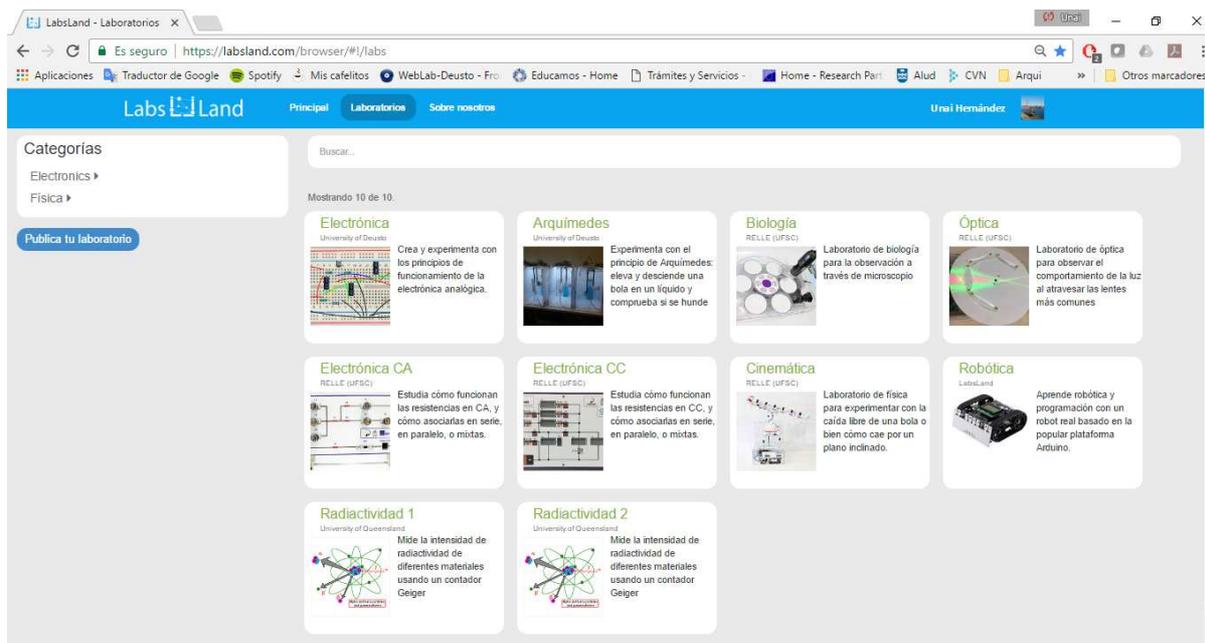
- Conocer el interfaz del laboratorio remoto para acceder a sus diferentes secciones

### Contenidos específicos:

- Definición de sitio web, mapa del sitio web.

## Accediendo al Laboratorio Remoto

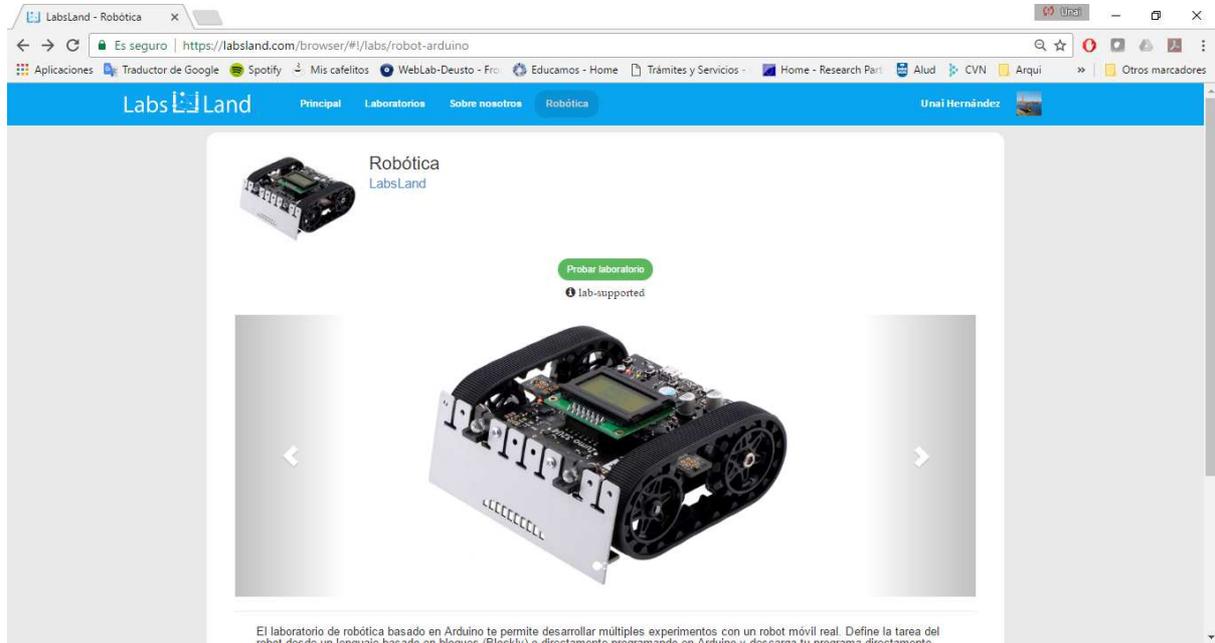
Un laboratorio remoto de robótica es una plataforma que, mediante Internet, permite comprobar el funcionamiento de los programas realizados sobre un robot real. Para acceder al laboratorio remoto de robótica ofrecido por LabsLand, debes en primer lugar registrarte en su página web <https://labsland.com> El registro es muy sencillo y puedes hacerlo usando tu cuenta de LinkedIn, Gmail o Facebook, por ejemplo.



Una vez registrado, podrás acceder al laboratorio de robótica, haciendo click en su enlace:



Desde este enlace accederás a otra pantalla, en la que encontrarás más información del robot y el acceso, haciendo clic en “Probar laboratorio”.



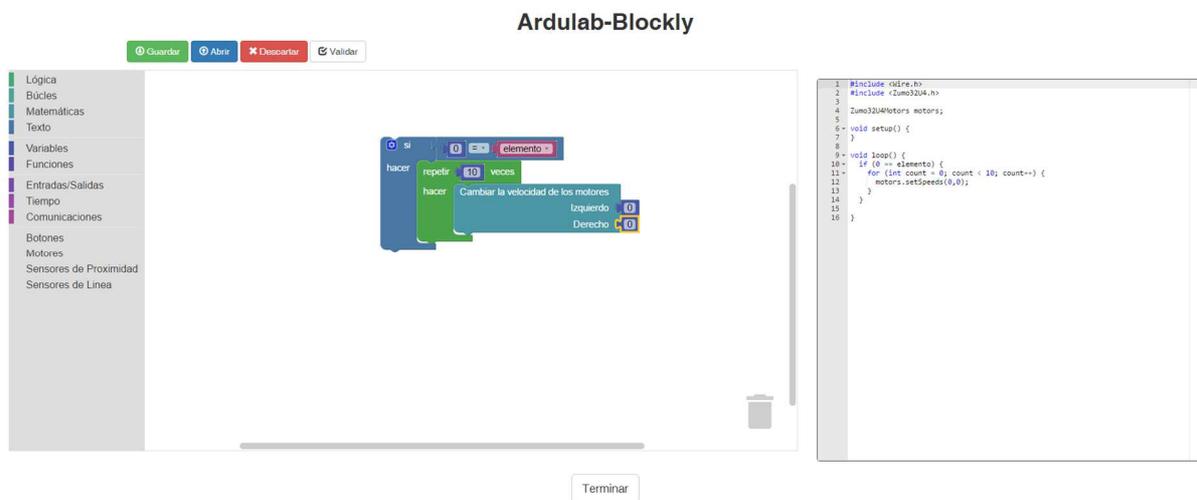
En la siguiente pantalla es donde encontrarás las diferentes formas de interactuar con el laboratorio. El orden lógico es: primero acceder al Interfaz de Programación Basada en Bloques y una vez que has codificado y compilado tu programa sin errores, acceder al Arduino Robot Lab, en donde podrás comprobar cómo funciona el robot con el programa que previamente has codificado.

En una versión extendida del laboratorio (no en la demo) también podrás realizar competiciones entre tus alumnos, de modo que puedan comparar el tiempo que tardan en, por ejemplo, dar la vuelta a los circuitos disponibles en el laboratorio.



### Interfaz de programación basada en bloques

Si hacemos click en esta opción, accederemos a la página de reserva de esta sección del Laboratorio Remoto. Haciendo click en el botón “reservar” podremos construir los programas para nuestros robots.



El entorno utilizado para programar es Blockly, que es un sistema basado en bloques para construir nuestro código visualmente. En la parte izquierda del interfaz tenemos las distintas familias de bloques, agrupadas por colores según su funcionalidad. Así, tendremos bloques para construir sentencias condicionales, bucles, operaciones matemáticas y todo lo que necesitemos.

En la parte central del entorno construiremos nuestro programa, arrastrando los bloques desde la parte izquierda. Para eliminar los bloques, podemos arrastrarlos al icono de la

papelera, en la esquina inferior derecha, o seleccionarlo y pulsar la tecla “suprimir”. En la parte derecha se irá generando el código correspondiente en lenguaje C.

En la parte superior hay una serie de botones que nos permitirán guardar el trabajo realizado, cargar un trabajo que hayamos guardado previamente o descartar el programa que estamos haciendo. El botón “validar” se usa para saber si el programa que hemos construido es válido para ser ejecutado en el robot.

Finalmente, cuando queramos terminar nuestro experimento, haremos click en el botón “terminar” para volver a la página de reserva.

## Arduino Robot Lab

Si hacemos click en esta opción, accederemos a la página de reserva de esta sección del Laboratorio Remoto. Haciendo click en el botón “reservar” podremos visualizar el comportamiento del robot al ejecutar distintos programas.

The screenshot displays the 'Sigue-lineas' (Line Follower) interface. At the top left, a timer shows '00:04:18'. Below it, there are two panels: 'Tus programas:' (Your programs) which is empty, and 'Programas demostrativos:' (Demonstration programs) containing 'lineFollower', 'buttonControl', and 'testButtons'. A 'Programar' button is located below these panels. The central area features a live video feed of a blue robot on a green surface with a black line. The video feed includes a timestamp '07/21/2016 09:14:11' and the camera name 'ardubot\_cam\_1'. To the right of the video are a 'Cronometro' (Timer) and a 'Monitor serial' (Serial Monitor) window. At the bottom, there are three red circular buttons labeled 'A', 'B', and 'C', two grey circular icons representing LEDs, and a 'Terminar' (End) button.

En la parte central de la interfaz podemos ver al robot mientras ejecuta el código con el que lo hemos programado. En la esquina superior izquierda vemos el tiempo que nos queda en la reserva del robot.

En la parte izquierda está la sección de los programas que podemos cargar en el robot. En el bloque superior está la lista de programas que hemos creado nosotros. En el bloque inferior se muestra una serie de programas demostrativos diseñados para mostrar las distintas funcionalidades del robot. Debajo de este bloque está el botón “programar”, con el que podremos programar el robot con el programa seleccionado.

En la parte inferior de la interfaz vemos tres botones rojos, “A”, “B” y “C”, con los que podremos interactuar con el robot. También hay dos iconos que representan los LEDs del robot y que se iluminan cuando lo hacen los LEDs reales.

En la parte derecha hay un bloque que puede mostrar un cronómetro y, en la parte inferior, un bloque que monitoriza la comunicación serial con el robot.

### PROPUESTA DE EJERCICIOS

**Motivación:** Se explica a los alumnos el interfaz del Laboratorio Remoto, sus diferentes secciones y funcionalidades. A continuación se les explica cómo diseñar un mapa esquemático del sitio web, con la interrelación entre las distintas páginas web que lo conforman. Se propone a los alumnos que piensen en posibles mejoras en el interfaz.

Agrupamiento: Toda la clase  
Recursos: No se requieren  
Espacio: Aula de clase  
Tiempo: 30 minutos

*Solución:* Después de la explicación del profesor, éste dejará unos 10 minutos para que los alumnos piensen en qué mejoras incorporarían al interfaz del Laboratorio Remoto.

**Desarrollo:** Se propone a los alumnos que dibujen un mapa esquemático del sitio web del Laboratorio Remoto en el que han pensado.

Agrupamiento: Individual  
Recursos: Papel y lápiz  
Espacio: Aula de clase  
Tiempo: 10 minutos

*Solución:* El esquema del Laboratorio Remoto es conocido por el profesor. Éste se encargará de comparar los diseños de los alumnos con el diseño real para comprobar que los alumnos han accedido a todas las secciones del sitio web.

**Consolidación:** En grupos reducidos, los alumnos pondrán en común sus esquemas y discutirán las particularidades de cada uno, detectando carencias y proponiendo entre todos mejoras en la interfaz. Finalmente, construirán un esquema final que mostrarán al resto de la clase.

Agrupamiento: Grupos reducidos (3 a 6 alumnos)  
Recursos: Papel y lápiz  
Espacio: Aula de clase  
Tiempo: 30 minutos

*Solución:* El profesor deberá comparar los esquemas finales con el esquema real del Laboratorio remoto y valorar las posibles propuestas de mejora por parte de los alumnos.

### 7.3 ¿Qué es un algoritmo?

<b>Objetivos específicos:</b>
-------------------------------

- Desarrollar habilidades para la creación de diagramas de flujo
- Organizar actividades siguiendo el patrón de un algoritmo

**Contenidos específicos:**

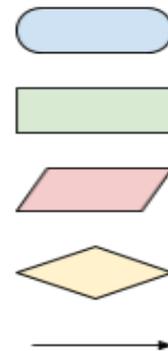
- Definición de algoritmo
- Diagramas de flujo

**Definición de Algoritmo y Diagramas de Flujo**

Se conoce como "algoritmo" a un conjunto de instrucciones concretas que, ejecutadas en orden, permiten la realización de una tarea. Una receta de cocina o las instrucciones de montaje de un mueble son ejemplos cotidianos de algoritmos.

Aunque los algoritmos deben codificarse en algún lenguaje de programación para que pueda un ordenador pueda ejecutarlos, existen formas alternativas de representarlos para facilitar su comprensión por parte de las personas. Los diagramas de flujo sirven para tal fin y los elementos más importantes que los componen son los siguientes:

- El inicio y el final de un algoritmo se representan mediante cápsulas,
- Las acciones individuales, tales como cálculos y tomas de decisiones, mediante rectángulos,
- Los datos de entrada y de salida a usar por las acciones, mediante romboides,
- Los puntos en los que la ejecución puede tomar distintos caminos en función de cierta condición, mediante rombos,
- Los elementos se conectan entre sí en orden mediante flechas.

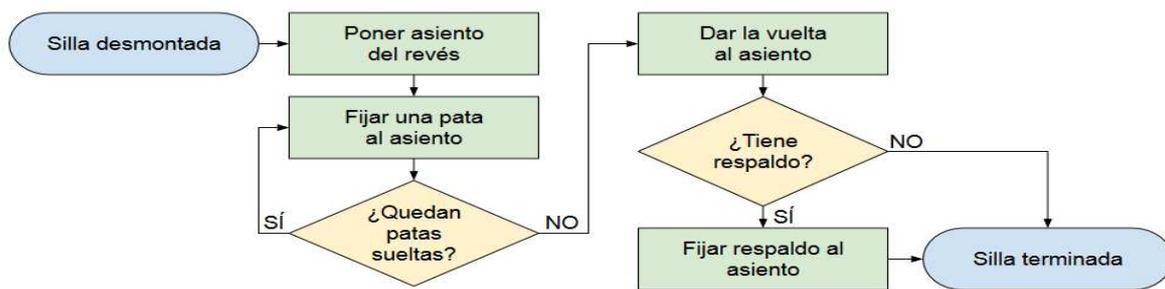


A la hora de escribir un programa en un ordenador, es muy recomendable crear primero un guión o diagrama de flujo que permita recordar qué acciones hay que implementar y en qué orden se ejecutan.

Por ejemplo, supongamos que disponemos de las piezas necesarias para construir una silla o taburete. Las instrucciones de montaje, junto a un posible código de programa y su correspondiente diagrama de flujo, serían las siguientes:

1. Tómesese el asiento del revés
2. Fíjese cada una de las patas al asiento
3. Dése la vuelta a la estructura
4. Si la silla tiene respaldo, fíjese al asiento

```
montar una silla:
dar la vuelta (asiento)
para (cada pata de la silla):
    fijar (pata, asiento)
dar la vuelta (asiento)
si (la silla tiene respaldo):
    fijar (respaldo, asiento)
```



### PROPUESTA DE EJERCICIOS

**Motivación:** Se propone a los alumnos escoger una actividad que realicen de forma cotidiana y describirla en forma de diagrama de flujo.

Agrupamiento: Individual  
 Recursos: Papel y lápiz  
 Espacio: Aula de clase  
 Tiempo: 20 minutos

**Solución:** Ha de tenerse en cuenta que cada alumno puede desempeñar la misma tarea siguiendo diferentes pasos. Se ha de evaluar que se usen correctamente las distintas formas de los diagramas de flujo (entrada y salida, acciones, datos y decisiones).

**Desarrollo:** Para contrastar los diferentes algoritmos que pueden describir una misma tarea, cada alumno dirá en clase cuál ha escogido para el ejercicio anterior y se compararán aquellas que sean similares.

Agrupamiento: Toda la clase  
 Recursos: No se requieren  
 Espacio: Aula de clase  
 Tiempo: 20 minutos

**Solución:** El profesor debe permitir que los alumnos debatan acerca de la eficiencia de sus algoritmos, asegurándose de que comprendan las restricciones temporales impuestas por los mismos. Como norma, el orden propuesto para las acciones en un algoritmo tiene que respetarse.

**Consolidación:** El profesor propondrá una tarea compleja que los alumnos deberán describir en forma de algoritmo. Algunos ejemplos de ello podrían ser “ir a la Luna”, “llegar a presidente” o “construir un robot”. La tarea no tiene por qué ser realista ni factible, puede permitirse que los alumnos usen su imaginación y la fantasía.

Agrupamiento: Grupos reducidos (3 a 6 alumnos)  
 Recursos: Papel y lápiz  
 Espacio: Aula de clase  
 Tiempo: 30 minutos

*Solución:* De nuevo, el profesor sólo debe comprobar que los elementos de los diagramas de flujo se utilicen correctamente y que la secuencialidad de las acciones se respete.

#### 7.4 Nuestro primer programa en el Laboratorio remoto: moviendo el robot

##### Objetivos específicos:

- Identificar cómo se relaciona el movimiento de las ruedas del robot con el movimiento del robot completo
- Aprender a programar el movimiento del Robot

##### Contenidos específicos:

- Definición de motor de corriente continua
- Programación del movimiento de un Robot

#### Definición de motor de corriente continua

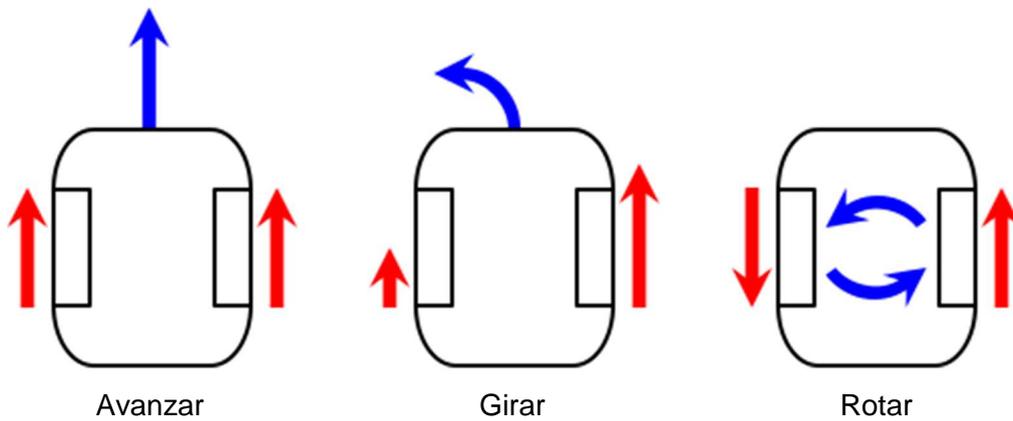
Un **motor de corriente continua** (conocido también como motor CC o DC) es un componente que convierte la energía eléctrica en energía mecánica, provocando un movimiento de rotación continua en su eje.

Este tipo de motores son muy utilizados en un amplio rango de electrodomésticos y aparatos además de en robótica. En nuestro caso utilizaremos este tipo de motores para dotar de movimiento el Robot disponible en **Zumoline**.

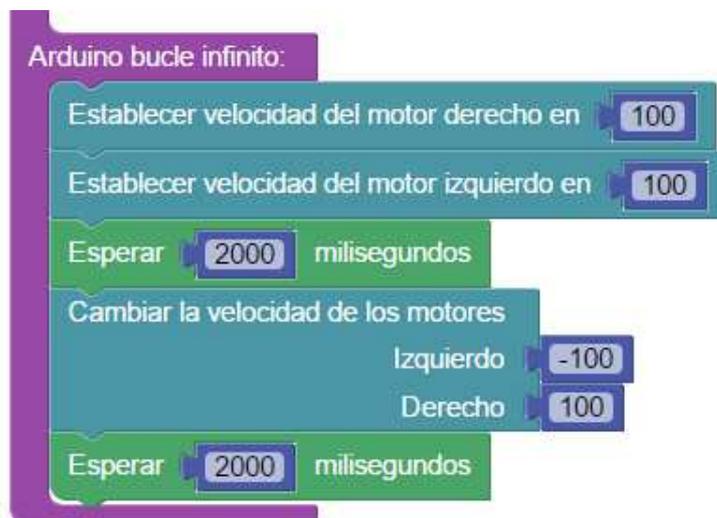


#### Programando el movimiento del robot

Una característica básica de nuestro robot es su capacidad de moverse. Imaginemos que queremos programar nuestro robot para que sea capaz de realizar cinco tipos de movimientos: **avanzar**, **retroceder**, **girar hacia la derecha**, **girar hacia la izquierda** y **girar sobre sí mismo**. Debemos tener en cuenta que estos movimientos deben darse utilizando únicamente dos motores y por lo tanto dos ruedas. La configuración de velocidad para cada rueda en función del movimiento que queremos realizar la podemos ver en la siguiente imagen:

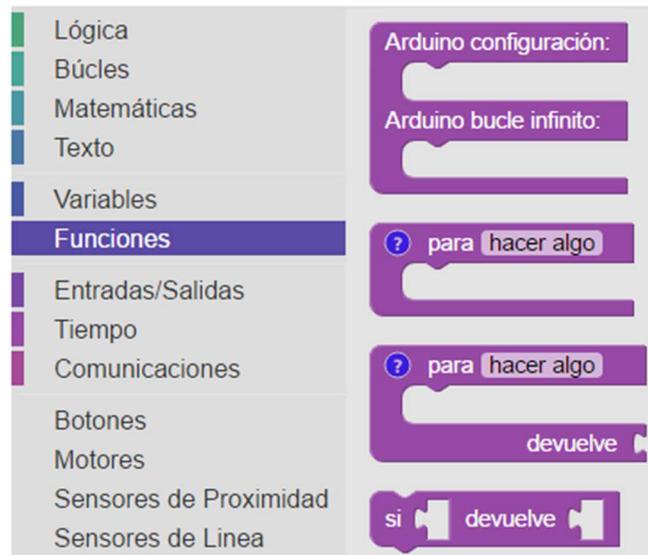


Como vemos, los distintos movimientos que queremos generar dependen principalmente de la diferencia de velocidad entre las dos ruedas. En el caso de que queramos que el robot se mueva hacia la izquierda, tendríamos que dar una **velocidad baja** a la rueda izquierda y una **velocidad alta** a la rueda derecha. Si quisiéramos que el robot avanzara o retrocediera, lo que tendríamos que hacer es dar la **misma velocidad** a cada rueda. En la programación del robot, para la asignación de distintas velocidades introduciremos valores entre 0 y 255; con ello conseguiremos generar las distintas velocidades de rotación. En la siguiente imagen veremos dos ejemplos de programación, una para el caso de **avanzar** y otra para la **rotación hacia la izquierda**:

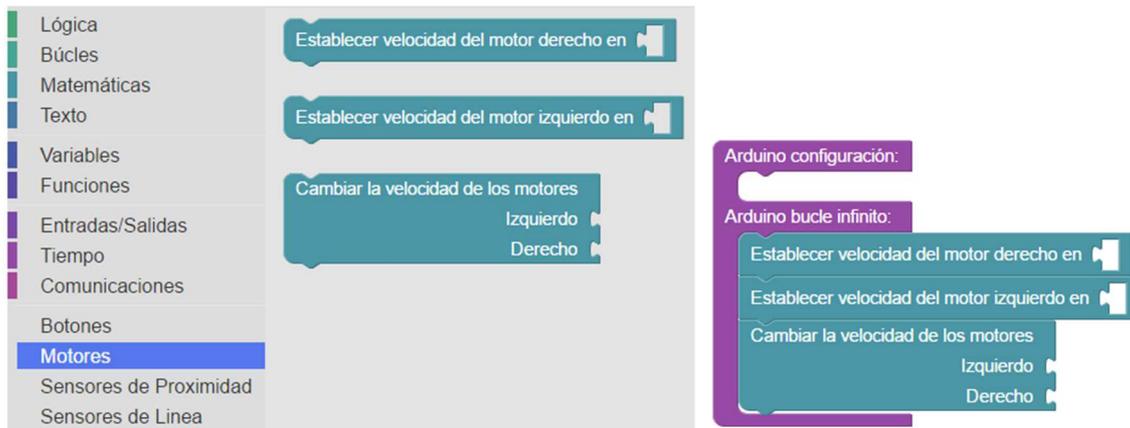


Vemos que en primer lugar avanzamos durante 2 segundos, para luego realizar un giro durante otros dos segundos y vuelta al principio.

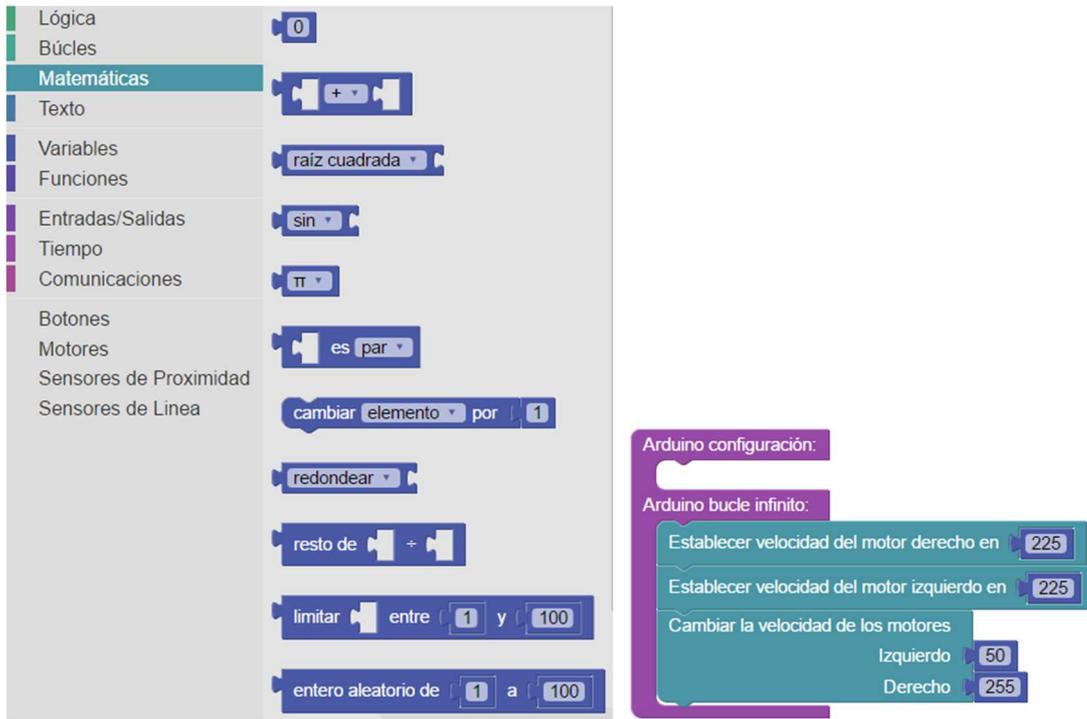
Vamos a ver ahora cómo construir este programa en el Laboratorio remoto. Lo primero es entrar en la sección de "Ardublocks" y seleccionar la familia de bloques "Funciones".



Ahora seleccionamos el primer bloque “Arduino configuración - Arduino bucle infinito” y lo arrastramos hasta la zona del programa en la parte central del entorno. Este bloque hará que los bloques que coloquemos dentro de él se ejecuten continuamente. A continuación, seleccionamos la familia de bloques “Motores” para incluir los bloques relacionados con el movimiento de los motores de la misma forma.

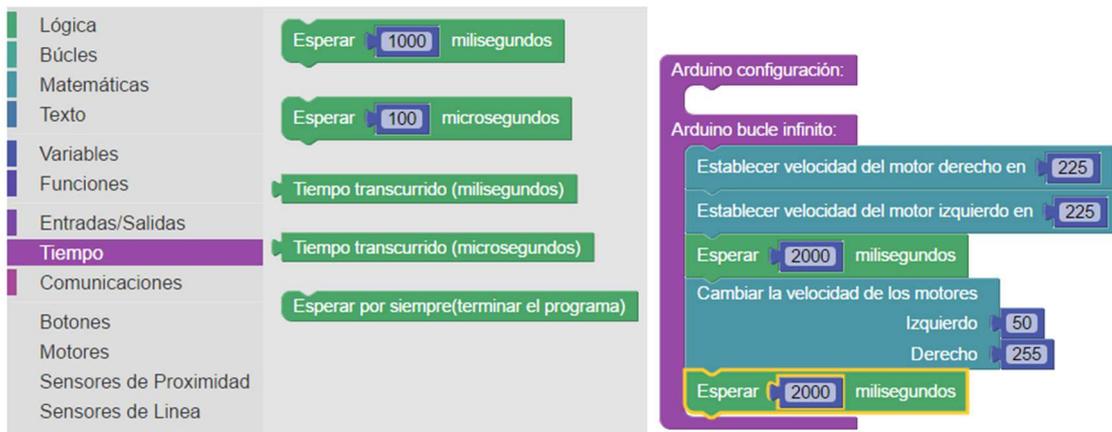


Ahora vamos a añadir la velocidad de giro de los motores. Para ello, seleccionamos la familia de bloques “Matemáticas” y arrastramos el primer tipo de bloques “Un número” a los huecos libres en los bloques correspondientes al movimiento de los motores.



**Un truco:** cuando necesites el mismo tipo de bloque en diferentes lugares, puedes usar Ctrl+C y Ctrl+V para copiar y pegar el mismo bloque y usarlo en otro lugar, con lo que ahorrarás el tiempo de seleccionarlos de uno en uno.

De momento, este programa actuaría sobre los motores del robot pero no hemos definido cuánto tiempo se tienen que mover los motores para que el robot haga lo que nosotros queremos. Para ello, vamos a incluir bloques que controlen el tiempo a través de distintos retardos. Abrimos la familia de bloques “Tiempo” e insertamos bloques de tipo “Esperar - milisegundos” entre el movimiento hacia adelante del robot (es decir después del segundo bloque de movimiento) y después del giro (después del tercer bloque de movimiento). Nuestro programa quedaría así:



Una vez que hemos construido nuestro programa tenemos que ejecutarlo en el robot para ver si funciona como nosotros queremos. Al entrar en la sección “Zumoline” del laboratorio remoto seleccionamos el programa que aparece en la sección “Tus programas” y lo cargamos en el

robot con el botón "Programar". Podemos ver en la parte central entorno como nuestro robot se mueve siguiendo las órdenes que hemos programado. Este sencillo programa no nos permite controlar el robot interactivamente, y lo más probable es que se acabe chocando con alguno de los límites del entorno de operaciones del robot.

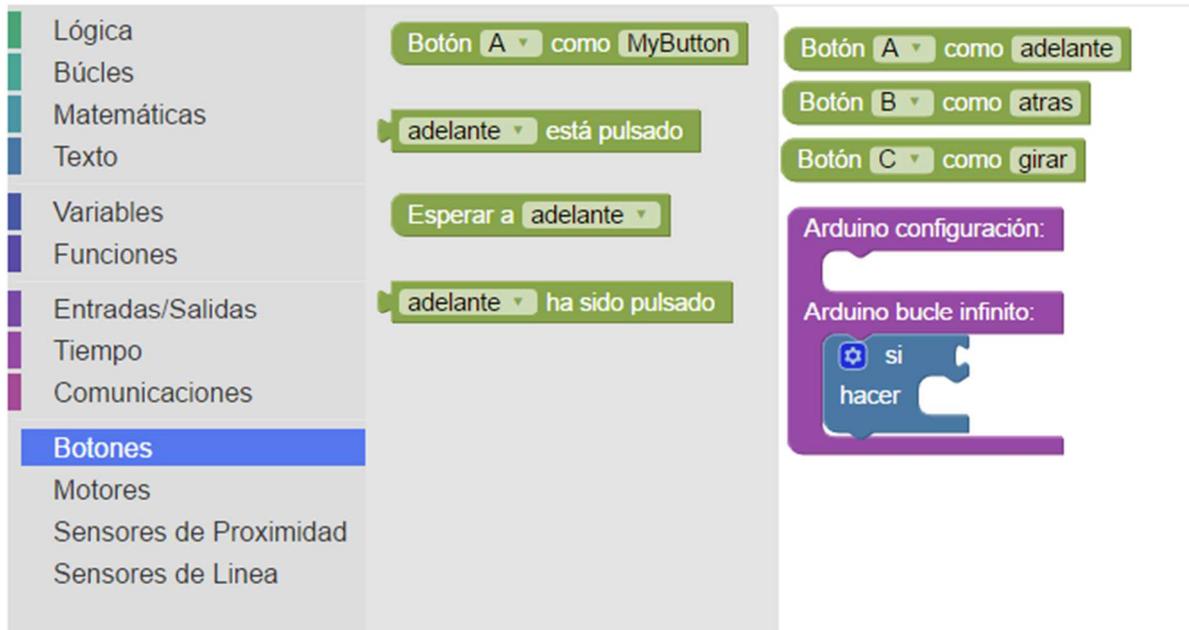
Para tener un mayor control sobre los movimientos del robot, el Laboratorio remoto incluye una serie de botones programables con lo que podemos controlar el robot de forma interactiva. Vamos a asignar a cada botón un movimiento del robot.

De vuelta a la sección "Ardublocks" vamos a construir un programa para que cuando pulsemos el botón A, el robot avance, cuando se pulse el botón "B" el robot retroceda y que cuando se pulse el botón "C" el robot gire.

Para ello, después de incluir un bloque de bucle infinito como en el ejemplo anterior, abriremos la familia de bloques "Lógica" y añadiremos un bloque "si - hacer", como vemos en la siguiente imagen:



Los bloques "si - hacer" ejecutan el bloque que conectemos en la parte "hacer" siempre que se cumpla la condición expresada en el bloque que conectemos en la parte "si". En nuestro ejemplo queremos que cuando se pulsa un botón, el robot se mueva. Vamos a incluir el bloque que informa de que un botón ha sido pulsado. Para ello, abrimos la familia de bloques "Botones" y arrastramos a nuestro programa el bloque "Botón - como". Este bloque nos permite asignar un nombre a los tres botones disponibles en el interfaz del Laboratorio remoto. Asignamos los tres botones como vemos en la siguiente imagen:

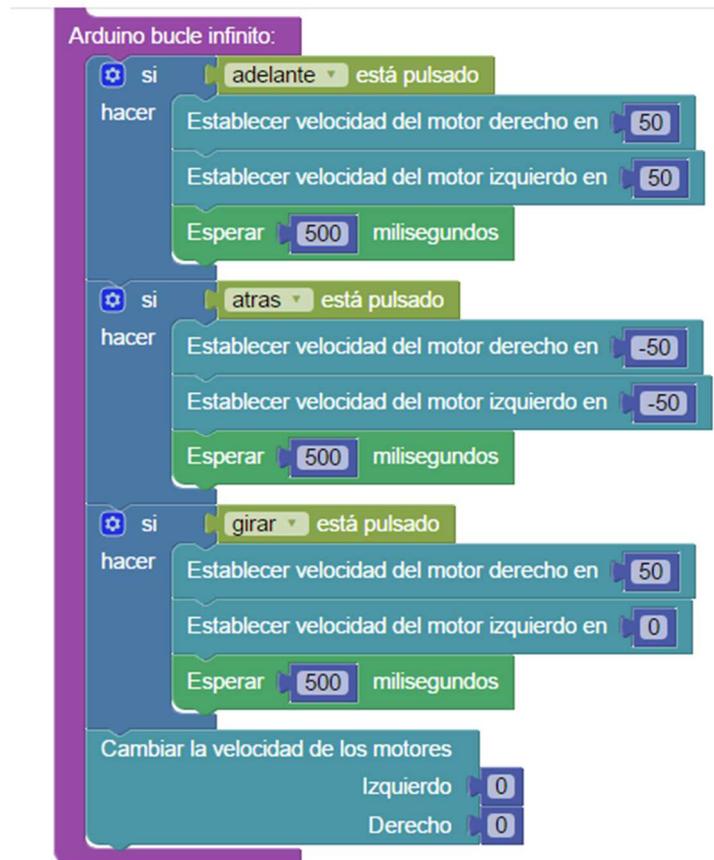


Ahora que hemos definido los nombres de los botones, podemos seleccionar el botón correspondiente dentro del bloque “está pulsado”. Este bloque nos informa si un determinado botón está pulsado y lo vamos a usar para conectarlo en el bloque condicional que hemos añadido previamente como se ve en la siguiente imagen.



También hemos añadido los bloques que actúan sobre los motores como en el ejemplo anterior. En este caso, como es el botón “adelante” la velocidad de los dos motores es la misma para que el robot se mueva adelante. Se ha incluido también un bloque de espera para que los motores se muevan durante ese tiempo, y un bloque “Cambiar velocidad de los motores” para mantener los motores parados siempre que no se pulse un botón.

Ahora vamos a incluir los bloques correspondientes a los otros botones para que el robot gire o vaya hacia atrás, como puede verse en la siguiente imagen:



Igual que en el ejemplo anterior, vamos a la sección del Laboratorio remoto "Zumoline" y programamos nuestro robot con el nuevo ejemplo. En este caso podremos controlar el movimiento del robot de forma interactiva y remota usando los botones A (adelante), B (atrás) y C (girar). Este programa es muy útil para usarlo en combinación con programas más complejos que requieran que el robot comience en una posición inicial concreta. De esta forma podemos colocar manualmente nuestro robot donde queramos.

Nota: en estos ejemplos hemos visto bloques condicionales, bucles, etc. De momento no es necesario conocer cómo funcionan esos bloques en profundidad sino familiarizarse con el proceso de construcción y ejecución de los programas en el Laboratorio remoto. En las siguientes actividades se explicará en mayor profundidad cada familia de bloques.



## Solución y demostración.

A modo de ejemplo de solución al ejercicio anterior, la siguiente secuencia de comandos hace que el robot en primer lugar gire sobre sí mismo (fíjate que la velocidad con la que hemos configurado cada motor es la misma pero con signo contrario) y después, tras esperar 2000ms (2 segundos) realiza otro giro, pero ya no sobre sí mismo, ya que las velocidades son distintas.

Recuerda que para codificar el programa debes en primer lugar acceder al laboratorio denominado “Ardublocks”

**Ardulab-Blockly**

```
1 #include <I2C.h>
2 #include <Uno324.h>
3
4 Uno324Motors motores;
5
6 void setup() {
7 }
8
9 void loop() {
10  motores.setRightSpeed(225);
11  motores.setLeftSpeed(225);
12  delay(2000);
13  motores.setSpeeds(50,255);
14  delay(2000);
15 }
16 }
```

Si queremos ahora comprobar cómo se ejecuta el programa, debemos una vez validado, acceder al laboratorio “Zumoline” y seguir los siguientes pasos:

1. Seleccionar con el ratón el programa que lleva por defecto el nombre “blocks”
2. Pulsar el botón Programar.
3. Observar como el robot ejecuta tu programa a través de la webcam. En función de la velocidad de tu conexión a internet, podrás ver el video con mejor calidad si pulsas el botón “MJPEG” que está sobre la imagen de la webcam.



### 7.5 Programando los LEDs\*

### 7.6 Bucles\*

### 7.7 Sensores de infrarrojos\*

### 7.8 Condiciones\*

### 7.9 Variables\*

### 7.10 Funciones\*

### 7.11 Seguir la línea\*

### 7.12 Salir del laberinto\*

## 8. EVALUACIÓN\*

### 8.1 Criterios de evaluación\*

### 8.2 Procedimientos e Instrumentos de Evaluación\*

## 9. CONCLUSIÓN\*

## 10. BIBLIOGRAFÍA Y WEBGRAFÍA\*

\*Si quieres conocer el contenido de estos apartados y qué más se puede hacer con el Laboratorio Remoto de robótica de LabsLand en tu aula y hasta cómo evaluar las actividades que proponemos, ponte en contacto con nosotros ([contact@labsland.com](mailto:contact@labsland.com)) y te informaremos de cómo desarrollar todas las opciones que te proponemos en el índice de este documento.