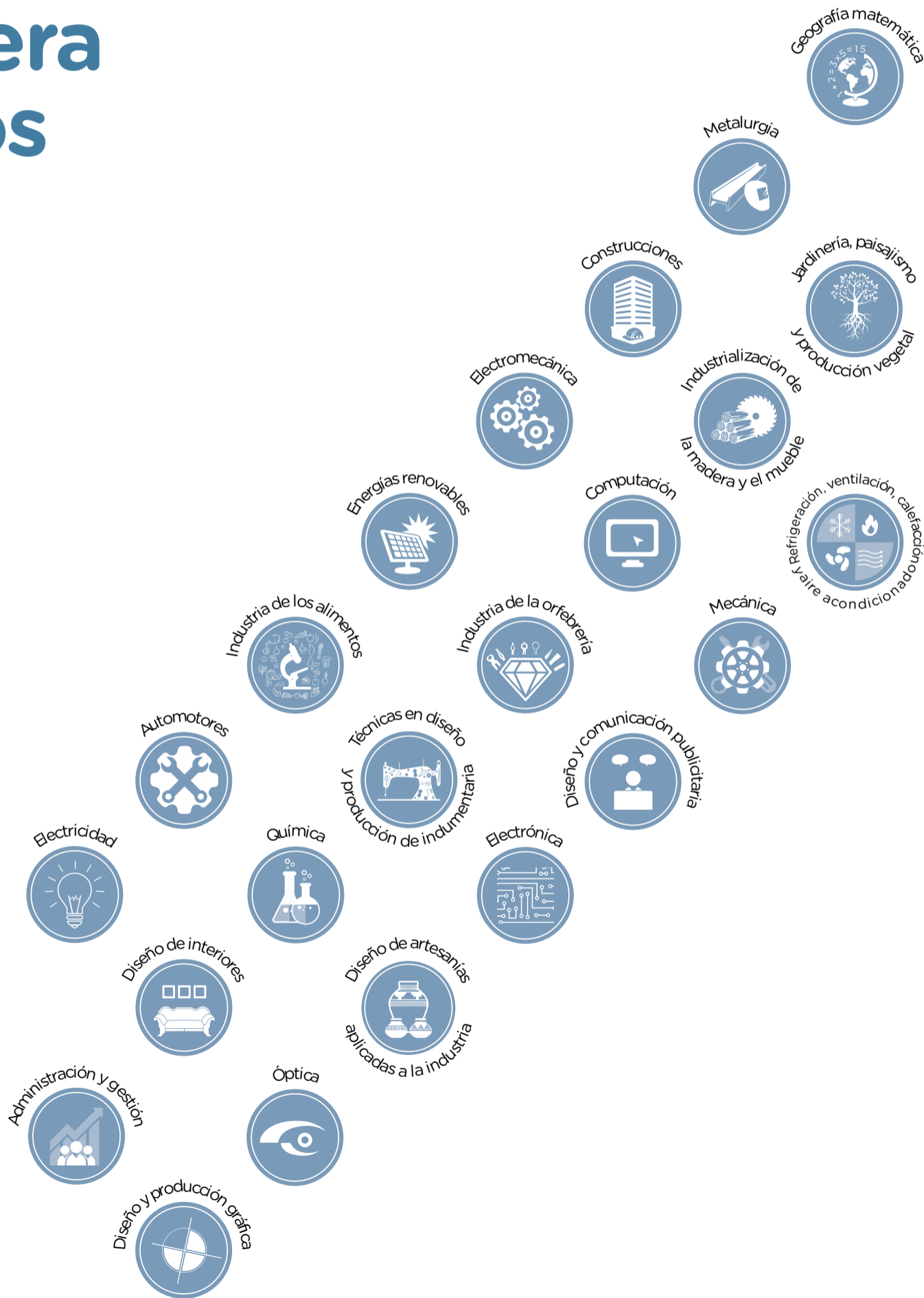


La carrera de autos

G.C.A.B.A. | Ministerio de Educación | Dirección General de Planeamiento Educativo | Gerencia Operativa de Currículum.



Jefe de Gobierno

Horacio Rodríguez Larreta

Ministra de Educación

María Soledad Acuña

Jefe de Gabinete

Manuel Vidal

Subsecretaria de Coordinación Pedagógica y Equidad Educativa

María Lucía Feced Abal

Subsecretario de Carrera Docente

Oscar Mauricio Ghillione

Subsecretario de Tecnología Educativa y Sustentabilidad

Santiago Andrés

Subsecretario de Gestión Económico Financiera y Administración de Recursos

Sebastián Tomaghelli

Subsecretaria de la Agencia de Aprendizaje a lo Largo de la Vida

Eugenia Cortona

Directora Ejecutiva de la Unidad de Evaluación Integral de la Calidad y Equidad Educativa

Carolina Ruggero

Directora General de Educación de Gestión Privada

María Constanza Ortiz

Director General de Planeamiento Educativo

Javier Simón

Directora General de Educación Digital

Rocío Fontana

Gerente Operativo de Currículum

Eugenio Visiconde

Gerenta Operativa Tecnología e Innovación Educativa

Sandra Coronel

Dirección General de Planeamiento Educativo (DGPLEDU) Gerencia Operativa de Currículum (GOC)

Eugenio Visiconde

Equipo Nivel Secundario. Modalidad Técnico Profesional: Miguel Rubés (coordinación), Irma Sicardi (generalista).

Especialistas: Liliana Kurzrok (Matemática), Octavio Javier da Silva Gillig (Robótica).

Subsecretaría de Tecnología Educativa y Sustentabilidad (SSTES)

Dirección General de Educación Digital (DGED)

Gerencia Operativa Tecnología e Innovación Educativa (INTEC)

Sandra Coronel

Especialistas de Educación Digital: Julia Campos (coordinación), Sandra Coronel, Josefina Gutiérrez.

Equipo Editorial de Materiales y Contenidos Digitales (DGPLEDU)

Coordinación general: Silvia Saucedo.

Coordinación editorial: Marcos Alfonzo.

Asistencia editorial: Leticia Lobato.

Edición y corrección: Víctor Sabanes.

Corrección de estilo: Sebastián Vargas.

Diseño gráfico y desarrollo digital: Gabriela Ognio.

Este material contiene las actividades para estudiantes presentes en *La carrera de autos* ISBN 978-987-549-999-7

Se autoriza la reproducción y difusión de este material para fines educativos u otros fines no comerciales, siempre que se especifique claramente la fuente. Se prohíbe la reproducción de este material para venta u otros fines comerciales.

Las denominaciones empleadas en este material y la forma en que aparecen presentados los datos que contiene no implican, de parte del Ministerio de Educación del Gobierno de la Ciudad Autónoma de Buenos Aires, juicio alguno sobre la condición jurídica o nivel de desarrollo de los países, territorios, ciudades o zonas, o de sus autoridades, ni respecto de la delimitación de sus fronteras o límites.

La mención de empresas o productos de fabricantes, en particular, estén o no patentados, no implica que el Ministerio de Educación del Gobierno de la Ciudad Autónoma de Buenos Aires los apruebe o recomiende de preferencia a otros de naturaleza similar que no se mencionan.

Fecha de consulta de imágenes, videos, textos y otros recursos digitales disponibles en Internet: 1 de marzo de 2022.

© Gobierno de la Ciudad Autónoma de Buenos Aires / Ministerio de Educación. Dirección General de Planeamiento Educativo / Gerencia Operativa de Currículum, 2022. Carlos H. Perette y Calle 10 – C1063 – Barrio 31 - Retiro - Ciudad Autónoma de Buenos Aires.

© Copyright © 2022 Adobe Systems Software. Todos los derechos reservados. Adobe, el logo de Adobe, Acrobat y el logo de Acrobat son marcas registradas de Adobe Systems Incorporated.

¿Cómo se navegan los textos de esta serie?

Los materiales de Educación Técnica cuentan con elementos interactivos que permiten la lectura hipertextual y optimizan la navegación. Estos reflejan la interactividad general de la serie.

Para visualizar correctamente la interactividad se sugiere bajar el programa [Adobe Acrobat Reader](#) que constituye el estándar gratuito para ver e imprimir documentos PDF.



Pie de página

Volver a vista anterior — Al clicar regresa a la última página vista.



— Ícono que permite imprimir.



— Folio con flechas interactivas que llevan a la página anterior y a la página posterior.

Portada



— Flecha interactiva que lleva a la página posterior.

Itinerario de actividades

Actividad 1

Carreras de cuarto de milla

1

Organizador interactivo que presenta la secuencia completa de actividades.

Actividades

Carreras de cuarto de milla

Las carreras de cuarto de milla o *picadas*, como se denominan en la Argentina, constan de dos autos que tienen que recorrer 402,336 metros (el equivalente a un cuarto de una milla) en el menor tiempo posible y en línea recta.



Actividad anterior



Actividad siguiente



Botón de navegación.

Posición de la actividad en la secuencia.

Íconos y enlaces

1 Cita o nota aclaratoria. Clic para abrir *pop-up*:

Ovidescim repti ipita voluptis audi iducit ut qui

Los números indican las referencias de notas al final del documento.

El color azul y el subrayado indican un [vínculo](#) a la web o a un documento externo.



Título de la actividad o del anexo

Indica enlace a una actividad o a un anexo.

Itinerario de actividades



Actividad 1

Carreras de cuarto de milla

1



Actividad 2

Programación de un simulador

2



Actividad 3

La pista de carrera

3



Actividad 4

Desarrollo del auto robot

4



Actividad 5

La carrera de robots

5

Carreras de cuarto de milla

Las carreras de cuarto de milla o *picadas*, como se denominan en la Argentina, constan de dos autos que tienen que recorrer 402,336 metros (el equivalente a un cuarto de una milla) en el menor tiempo posible y en línea recta.

a. Investiguen en internet y contesten las siguientes preguntas acerca de las carreras de cuarto de milla que se corren en el Autódromo de Buenos Aires Juan y Oscar Gálvez. Al iniciar la búsqueda en internet, deben prestar especial atención a quiénes escribieron los artículos (si son idóneos/as o especialistas en la temática), a la actualidad del contenido y a las fuentes que se mencionan para sostener o fundamentar la información.

1. ¿Cuántos autos compiten por carrera?
2. ¿Cuánto dura, aproximadamente, cada carrera?
3. ¿Qué características tienen los autos que corren en este tipo de competencia?
4. ¿Qué características tienen las pistas en las que compiten?

b. Ingresen al proyecto [“Simulador carrera de autos”](#) programado en Scratch.



Imagen del programa “Simulador carrera de autos” realizado en Scratch.

Allí verán un auto que se desplaza a velocidad constante sobre una pista recta.

- c. Analicen el código del auto para identificar las variables que determinan los valores iniciales del proyecto. ¿Cuántas variables encontraron? ¿Cómo se llaman?

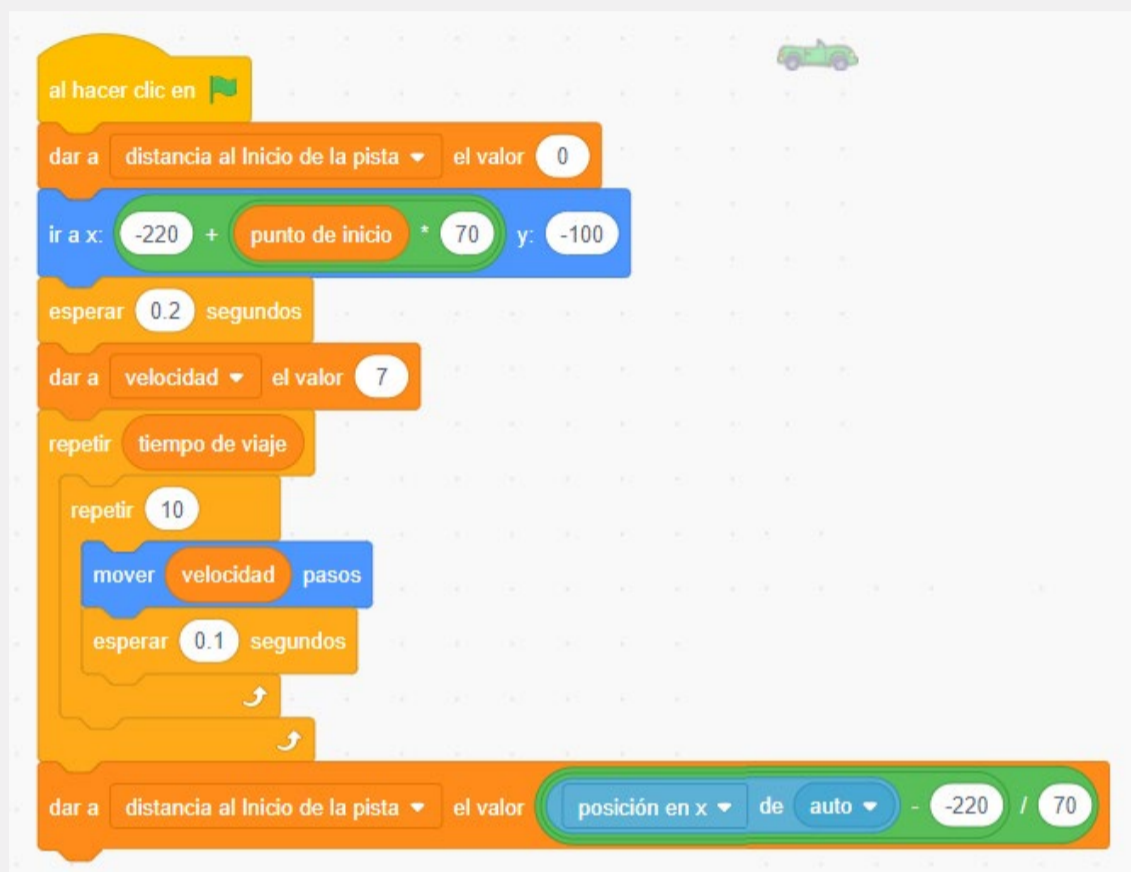


Imagen del programa “Simulador carrera de autos” realizado en Scratch.

1. Cambien los valores iniciales del proyecto y anticipen a qué distancia del inicio de la pista llegará el auto luego de algunos segundos de viaje. Verifiquen si su anticipación fue correcta.
2. Completen la siguiente frase con valores elegidos por ustedes para que sean resueltos por otro grupo de trabajo:

“Si el auto sale a del inicio de la pista, ¿a qué distancia del inicio de la pista llegará a los segundos de viaje?”.

3. Encuentren una manera de resolver la situación a partir de probar con el auto. Para eso, sigan estos pasos:
 - Propongan que el auto salga a dos unidades del inicio de la pista y viaje por tres segundos. Sin usar el simulador, anticipen a qué distancia del inicio de la pista llegará.
 - Revisen con el simulador si su anticipación se verificó. Si no es así, vuelvan a realizarla con otra propuesta.

- d. Luego de haber trabajado con el simulador, contesten:
1. ¿Qué datos tomaron?
 2. ¿Qué les importaba saber?
 3. ¿Pudieron anticipar donde se encontraría el auto? ¿Por qué?
 4. ¿Cuáles son los parámetros que quedan constantes en todo momento del viaje del auto? ¿Cuáles varían?

Sugerimos que registren estas respuestas en un procesador de textos con el objetivo de ir recuperándolas a lo largo de las actividades posteriores. Como la actividad se realiza en pequeños grupos, pueden hacerlo, si tienen conexión a internet, mediante un documento compartido en [Google Docs](#) (pueden consultar el [tutorial de Google Docs](#) en el Campus Virtual de Educación Digital). Allí, un miembro del equipo abrirá un documento y lo compartirá con permiso de edición para que todos puedan agregar sus comentarios. En caso de que no tengan conexión es posible hacerlo con [OpenOffice Writer](#) (pueden consultar el [tutorial de OpenOffice Writer](#) en el Campus Virtual de Educación Digital) y de manera organizada ir completándolo de a uno.

Actividad siguiente



Actividad 2

Programación de un simulador

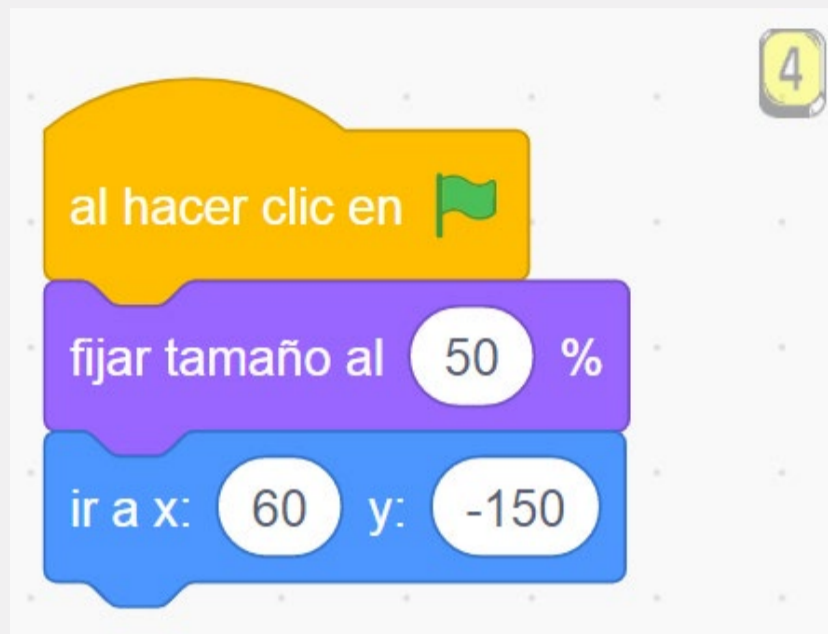
Luego de haber experimentado con [“Simulador carrera de autos”](#), realizaremos nuestro propio programa utilizando [Scratch](#) para hacer simulaciones de este tipo.

Identificación del problema: antes de empezar a programar es importante tener bien claro qué quieren hacer.

- a. Reunidos en grupos, respondan las siguientes preguntas en referencia al código de la actividad 1:
1. El simulador posee 7 objetos llamados “Mojón X”, donde X representa un número del 0 al 6. El código de cada mojón es similar al de los demás salvo por un valor, ¿qué valor es y qué representa?



Actividad 1.
Carreras de cuarto de milla



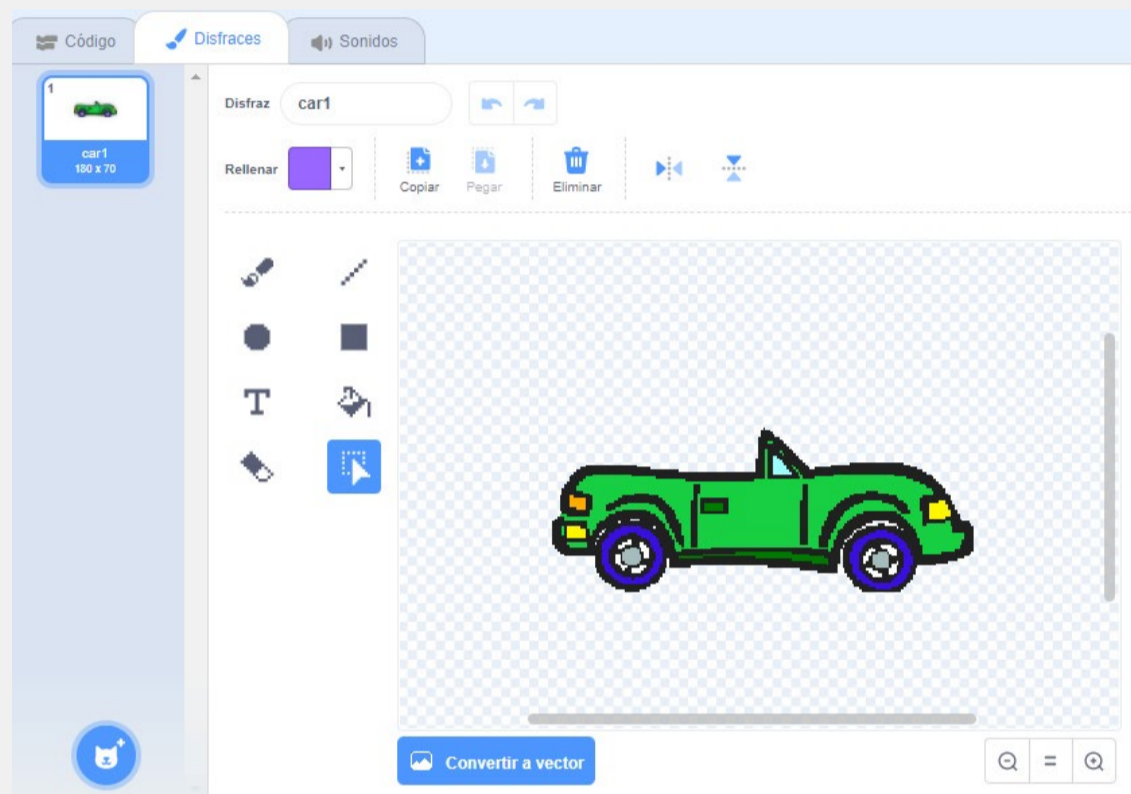
Código de Scratch del mojón 4.

2. El objeto “auto” es el protagonista del simulador. ¿Cuántos ciclos tiene su programa? ¿En relación con qué varían estos ciclos?
3. ¿Qué bloque de Scratch es el encargado de iniciar todas las acciones de todos los objetos en el simulador?
4. ¿Cuál es el bloque más utilizado en el programa del auto? ¿Por qué cada vez que se lo utiliza es con parámetros de entrada diferentes?
5. En pocas palabras, ¿cómo modificarías este programa para que tenga dos autos en vez de uno?

Diseño de la solución: una vez identificado el problema que quieren resolver, hay que pensar en posibles diseños antes de empezar a programar.

- b. Siguiendo con el trabajo en grupo, dibujen en un papel tres pantallas posibles para su programa. Estos dibujos funcionarán como esquemas para cuando hagan sus propias implementaciones. Al momento de realizar estos bocetos, tengan en cuenta qué tipo de fondo les gustaría usar, los autos y cualquier otro objeto que sea necesario para su programa. Podrán dibujar esas pantallas en papel o bien con algún programa de dibujo colaborativo como [Google Jamboard](#). Será importante que puedan agregar esos bocetos de pantallas al documento colaborativo grupal en formato imagen o con un enlace (caso [Google Jamboard](#)) para que quede un registro completo de las actividades.
- c. ¿Qué tipos de autos les gustaría tener en su programa? Evalúen las siguientes opciones y elijan la que prefieran antes de empezar con la implementación:
 1. Autos de la librería de objetos de [Scratch](#).

2. Autos dibujados desde cero en el editor de objetos de Scratch.
3. Autos de la librería de Scratch modificados utilizando el editor de objetos.
4. Autos que no pertenecen a la librería de Scratch agregados por ustedes al programa desde sus computadoras.



El editor de imágenes de Scratch se encuentra en la solapa “Disfraces” de cada objeto.

Implementación de la solución: luego de haber diseñado los principales aspectos del programa que van a realizar, es hora de poner manos a la obra y empezar a escribir código, editar imágenes y hacer que todo funcione.

- d. Para la etapa de implementación pueden dividirse las tareas que tienen que realizar. Por ejemplo, un subgrupo puede realizar el código para los autos y otro puede encargarse de los gráficos (autos, escenarios, mojonos, etc.).

Recuerden que el participante debe poder incorporar la distancia al inicio de la pista a la que sale el auto y que los dos autos no deben ir a la misma velocidad.

Testeo: una vez terminado el programa hay que asegurarse de que todo funciona como quieren, para lo cual deben testear el trabajo usando casos que les permitan revisar el código.

- e. Testeen su programa simulando la situación que se describe a continuación. Luego, contesten las preguntas que siguen.

El primer auto sale a 25 cm del inicio de la pista y viaja a velocidad constante de modo tal que a los 3 segundos se encuentra a 34 cm del inicio de la misma. El segundo viaja de modo tal que a los 3 segundos se encuentra a 24 cm del inicio de la pista y a los 6 segundos se encuentra a 39 cm del inicio de la pista.

1. ¿Es posible saber qué auto gana la carrera? Justifiquen la elección de la respuesta.
2. ¿Qué auto corre con ventajas? ¿Por qué?
3. ¿Se encontrarán los dos autos a la misma distancia del inicio de la pista en algún momento? ¿Cómo es posible saberlo?



Actividad anterior



Actividad siguiente



Actividad 3

La pista de carrera

En esta actividad deberán construir una pista que pueda ser utilizada para las carreras propuestas en la actividad 2 de cuarto de milla, como las que investigaron en la actividad 1.

Antes de empezar, tengan en cuenta las siguientes etapas en el desarrollo de un proyecto:

- Identificación del problema por resolver.
- Diseño de una solución.
- Implementación del diseño.
- Testeo de la implementación.

Identificación del problema por resolver

- a. Reunidos en grupo, revisen lo investigado en la actividad 1, vuelvan al documento compartido en [Google Docs](#) o en [OpenOffice Writer](#) y dibujen un esquema de una pista a escala que tenga las características de una pista para carreras de cuarto de milla.
- b. Confeccionen, en este mismo documento, una lista con las características principales de este tipo de pista y también con los puntos que consideran que pueden ser problemáticos a la hora de producir su propio desarrollo.



Actividad 2.
Programación
de un
simulador



Actividad 1.
Carreras de
cuarto de
milla

Diseño de la solución

- c. Utilicen un programa a elección del grupo para diseñar la pista y tengan en cuenta cada punto de la lista confeccionada previamente. Pueden utilizar software de diseño 2D, como por ejemplo [LibreCAD](#) (pueden consultar el [tutorial de LibreCAD](#) en el Campus Virtual de Educación Digital) o en el caso de querer realizarlo en 3D pueden hacerlo en la plataforma de [Tinkercad](#). Es importante, en este punto, que presten atención a los detalles, ya que lo que dibujen con el *software* lo implementarán más adelante. Asimismo, hay que tener en cuenta que luego harán robots para competir en estas pistas, por lo que es importante tener idea de qué tamaño serán los robots, para que puedan ser utilizados.

Implementación del diseño

- d. A partir del material disponible en el taller, armen las pistas que diseñaron previamente. Pueden utilizar impresoras 3D, si disponen de ellas.

Testeo de la implementación

- e. Chequeen que la implementación realizada coincida con el diseño que hicieron. Luego, prueben que la pista sea funcional al problema que identificaron. Si encuentran desperfectos o diferencias entre el diseño y la implementación, modifiquen lo que consideren necesario. Registren estos cambios en el documento para poder así identificarlos.



Actividad anterior



Actividad siguiente

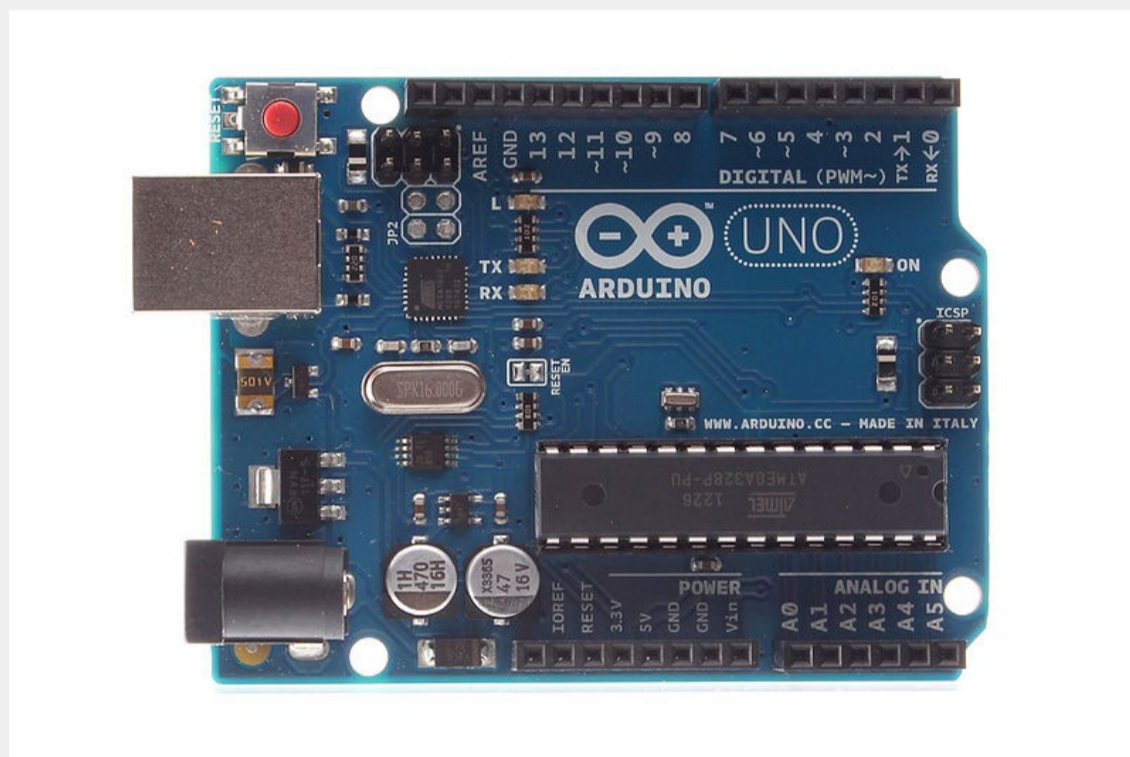


Desarrollo del auto robot

Identificación del problema

- a. Continuando con el trabajo en grupo, imaginen y esquematicen un robot que represente a un auto de carreras para jugar con la pista que armaron en la actividad anterior. Las siguientes preguntas pueden ayudarlos a pensar en aspectos importantes que deben tener en cuenta:

1. ¿Qué características son importantes en el diseño del auto? ¿Su forma, su peso, su tamaño, la cantidad de ruedas?
2. ¿Qué placa controladora les conviene utilizar para el auto robot que van a desarrollar?
3. ¿Cuántas ruedas y cuántos motores conviene que tenga? ¿Cuántas ruedas serían de apoyo y cuántas estarían conectadas a un motor?
4. ¿Qué otros componentes electrónicos necesitarán en el robot?
5. ¿Cómo harían para que toda la electrónica esté bien sujeta al chasis?
6. ¿Qué tendría que hacer el programa que graben en la placa controladora?



Ejemplo de placa controladora que pueden utilizar en el proyecto.

Diseño de la solución

- b. En la etapa de diseño deberán documentar con la mayor precisión posible los aspectos más importantes del robot. Para comenzar, deberán confeccionar una lista de los componentes electrónicos necesarios que van a utilizar. Pueden usar la siguiente tabla como ejemplo para empezar su lista.

Cantidad	Componente
1	Placa programable (pueden completar con el modelo elegido).
	Cables de conexión.
2	Motor.

- c. Mientras confeccionan la tabla de componentes electrónicos pueden ir armando un esquema con las conexiones correspondientes. Deberán diseñar el esquema de conexiones en el apartado de circuitos de [Tinkercad](#) y en la misma plataforma también pueden realizar simulaciones de la programación. Este esquema es indispensable para dejar registro de cómo será la interacción entre las partes del robot y, además, servirá de guía para armar y arreglar el robot cuando sea necesario.
- d. Teniendo en cuenta la electrónica y sus conexiones, es el momento de diseñar el chasis del robot. Pueden hacerlo usando el *software* de diseño 2D o 3D. Es sumamente importante que las piezas que dibujen utilizando estos programas coincidan entre ellas y también tengan las dimensiones necesarias para contener a la electrónica elegida. En el anexo se incluye una explicación detallada de cómo conectar una placa controladora Arduino con dos motores eléctricos, la cual puede ser útil para la etapa de implementación de la solución.



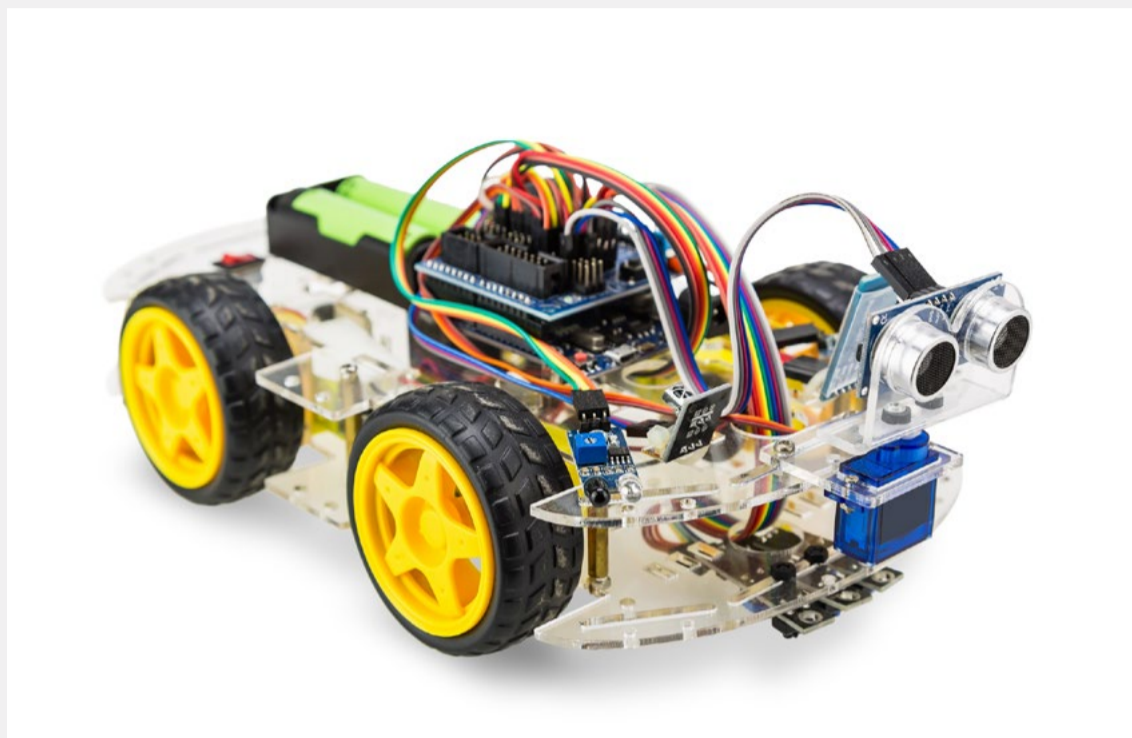
Anexo.
Conexión y
programación
de un robot
con dos
motores

Implementación del diseño

- e. Con los aspectos más importantes documentados, pasen a armar el robot. Pueden usar elementos que tengan a disposición en el taller y también pueden utilizar una impresora 3D, si hay alguna disponible.
- f. Con el robot completo, elijan un *software* que sirva para darle órdenes a placas programables como la que eligieron, por ejemplo [mBlock](#) (pueden consultar el [tutorial de mBlock](#) en el Campus Virtual de Educación Digital), el cual está basado en el entorno del programa [Scratch](#), y escriban el programa que va a ejecutar su robot. Luego, graben el programa en su robot.

Testeo de la implementación

- g. Prueben el funcionamiento de todo lo que hicieron y, si encuentran cosas para mejorar en el robot, hagan las modificaciones que consideren necesarias. Tengan en cuenta que cada modificación en el robot puede implicar revisar la documentación, ya que esta tiene que coincidir con el prototipo final.



Ejemplo de un robot similar a un auto desarrollado con una placa controladora programable.

← Actividad anterior

Actividad siguiente →

Actividad 5

La carrera de robots

- a. Es momento de utilizar todo lo desarrollado en las actividades anteriores. Para eso, presenten sus producciones al resto del curso. Utilizando soporte multimedial, armen presentaciones que muestren el proceso creativo por el que pasaron como grupo y expongan sus resultados frente a la clase. En las presentaciones, muestren los siguientes documentos mientras exponen el proceso creativo por el que pasaron y los intercambios de ideas que hubo dentro del grupo.
 - Primeros bocetos de la pista y del robot. Si los hicieron en papel, pueden escanearlos o sacarles fotos para digitalizarlos.
 - Diseños de partes en 3D tanto del robot como de la pista.
 - Esquemas de conexiones de componentes electrónicos junto con la lista de componentes.
 - Código utilizado para programar la placa.
 - Además, pueden agregar fotos del grupo de trabajo e identificar los roles de cada participante en el trabajo grupal.

Para estas presentaciones pueden elegir diferentes formatos, como por ejemplo:

- Presentaciones
 - » En el caso de poseer conexión las pueden realizar de forma colaborativa en [Google Presentaciones](#).
 - » En el caso de trabajar fuera de línea pueden realizar su presentación con el programa [OpenOffice Impress](#).
- Videos
 - » Pueden realizar un video con las distintas etapas del proceso, mostrando imágenes, capturas de pantalla, relatos en primera persona de las impresiones propias a lo largo del proyecto, registro de las grabaciones de los testeos, etc. Para la edición del video pueden utilizar el programa [OpenShot](#).
- Imágenes interactivas
 - » Pueden realizar imágenes interactivas, sumando el registro de todo el proceso de trabajo. Pueden confeccionarlas en plataformas como [Genially](#).
- Pizarras colaborativas
 - » En el caso de pizarras colaborativas pueden dividir por sectores la información que quieran mostrar y ofrecer un recorrido guiado. Un entorno como [Miro](#), puede ser una opción para esta posibilidad.

Además, pueden agregar, dentro de su presentación o al final de esta, una instancia de retroalimentación con el resto de la clase, por ejemplo, a partir de preguntas sobre el desarrollo o la presentación del proyecto. Para esta dinámica pueden utilizar una plataforma como [Mentimeter](#).

Tutoriales en el Campus Virtual de Educación Digital

- [Google Presentaciones](#)
- [OpenOffice Impress](#)
- [OpenShot](#)
- [Genially](#)
- [Mentimeter](#)

- b. Con el resto de la clase, elijan las pistas que les parezcan más adecuadas para hacer carreras de tipo cuarto de milla.

- c. ¡A correr! Muestren el funcionamiento de sus robots en las pistas elegidas. Utilicen cronómetros para tomar los tiempos de cada robot. Pueden hacer varias carreras y realizar modificaciones en los prototipos que desarrollaron para su mejor funcionamiento.



Actividad anterior



Anexo

Conexión y programación de un robot con dos motores

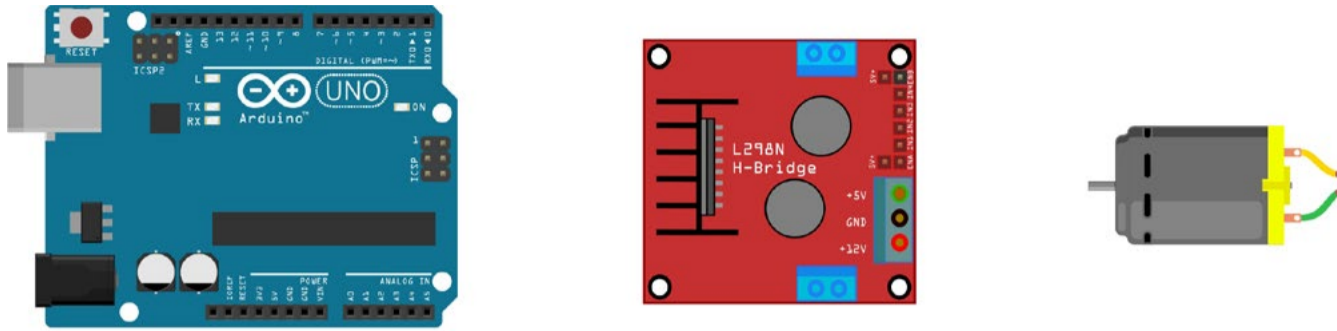
En este anexo se describe, paso a paso, una opción para conectar una placa [Arduino UNO](#) con dos motores de corriente continua y cómo escribir un programa sencillo que permita controlar la velocidad del robot. Esta es una manera en que pueden cablearse las componentes del robot. Hay muchas otras formas de conseguir resultados similares, incluso utilizando otros componentes electrónicos. Las placas [Arduino](#) permiten automatizar de manera sencilla los dispositivos realizados y programar, a través de secuencias de órdenes, el comportamiento de las creaciones.

Los motores que utilicen no pueden conectarse directamente a la placa Arduino. Por lo tanto, necesitarán una placa con un puente H que será el componente que permita controlar los motores y poder asignarles distintas potencias. Luego, desde la placa, podrán pedirle a cada motor, a través de este puente H, que avance o retroceda y que cambie su velocidad. Pero desde el entorno de programación Arduino IDE solamente configurarán la potencia que entregaron al motor.

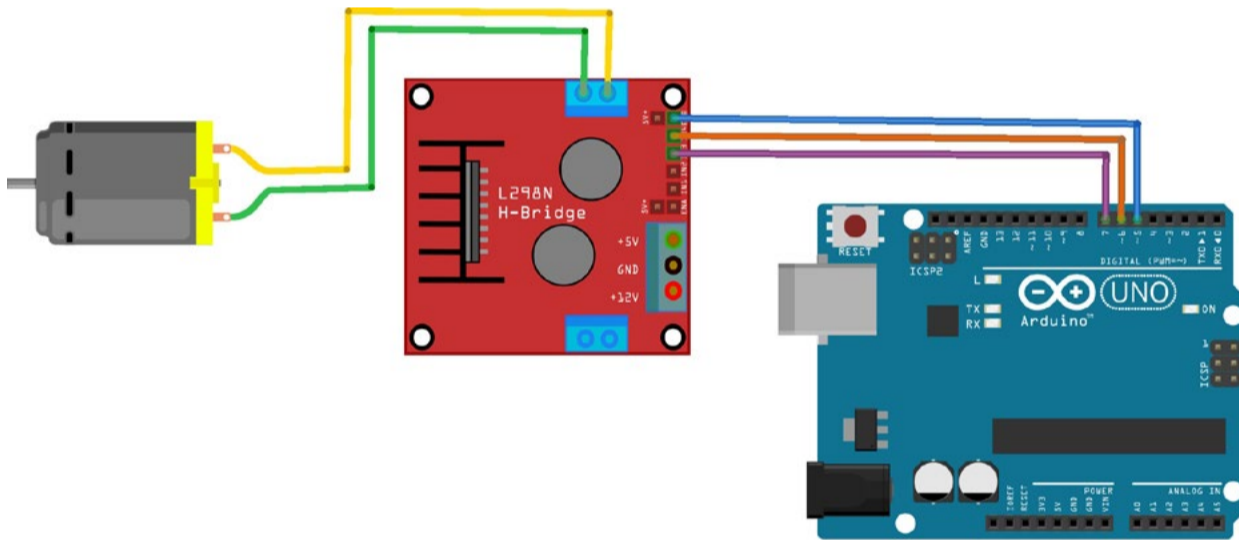
En este ejemplo van a utilizar los siguientes componentes electrónicos para automatizar el robot.

Cantidad	Componente
1	Arduino UNO o similar.
2	Motores de corriente continua.
1	Puente H driver L298.
	Cables de conexión de un solo hilo.
1	Batería 9V.
1	Portapilas de 4 elementos AA.

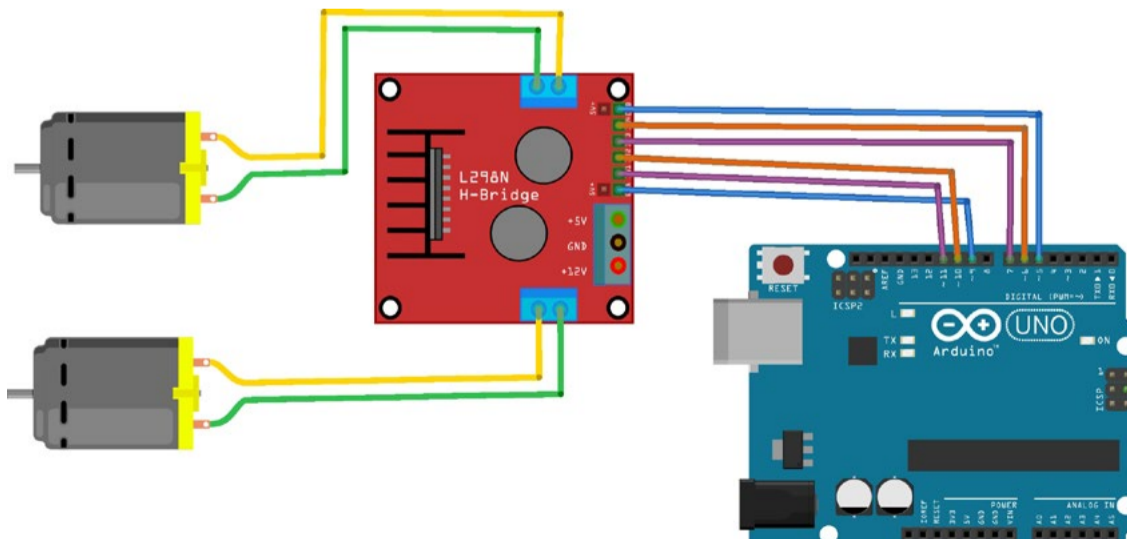
En la siguiente figura, de izquierda a derecha, se muestran los esquemas de la placa Arduino UNO, el puente H driver L298 y un motor de corriente continua. En el caso de los motores, estos suelen ser muy diferentes del que se muestra en estos esquemas, ya que poseen una reducción y una rueda acoplada al eje.



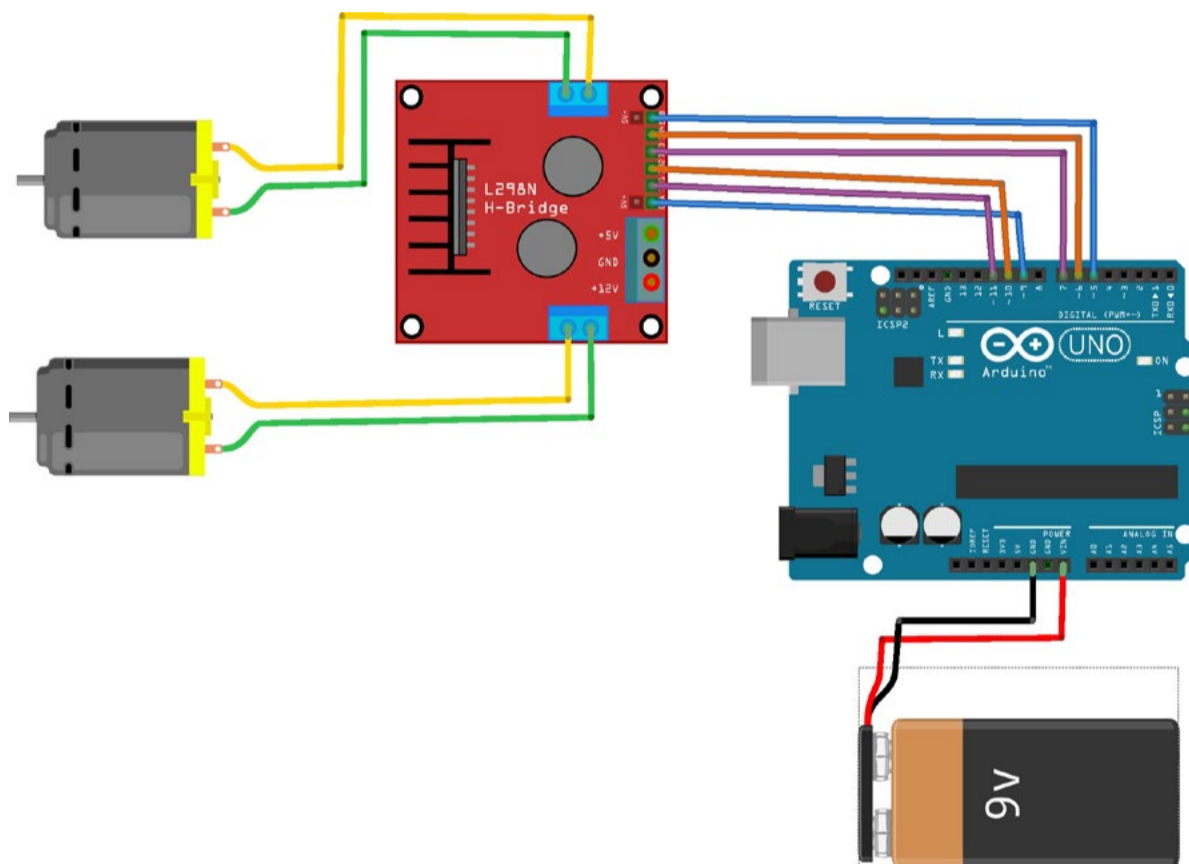
Lo primero que hay que hacer es conectar el motor al puente H. El motor tiene dos terminales: negativo y positivo. Estos, a través de cables de un hilo, van a una de las borneras celestes que posee la placa del puente H. Luego conecten los pines de la placa Arduino también a través de cables de un hilo en las entradas de datos de la placa controladora de motores, como se muestra en el siguiente esquema. En este ejemplo se utilizan los pines 5, 6 y 7 de la placa y se conectan con los pines nombrados como ENB, IN4 e IN3 respectivamente.



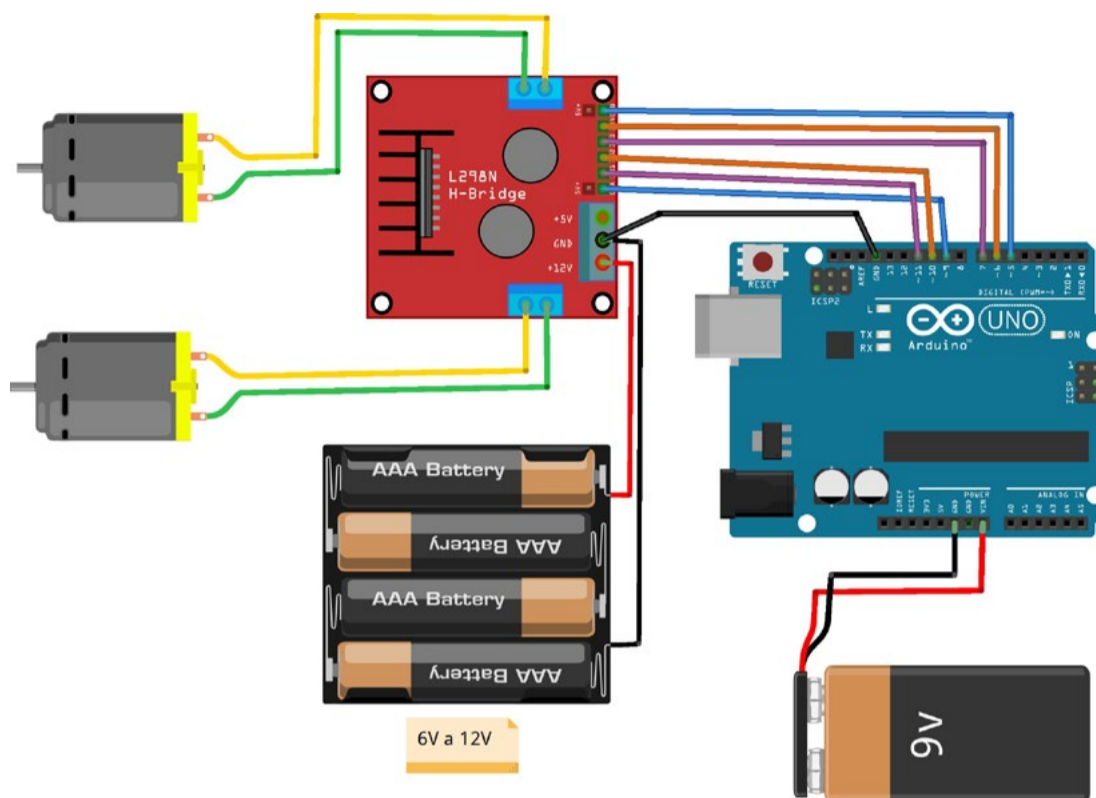
Una vez conectado el primer motor, se repite el cableado con el segundo motor, pero esta vez utilizando los pines 9, 10 y 11 de la placa y los pines ENA, IN2 e IN1 del puente H, tal como se muestra en la siguiente imagen.



Hasta acá tendrán los dos motores conectados a la placa a través del puente H. Lo que deberían hacer ahora es conectar la alimentación. Por un lado, deberán alimentar la placa controladora con una batería de 9V. Esto no sería necesario si la placa estuviese conectada a la computadora desde la que se envían las órdenes a través de un cable USB, ya que este le proveería los 5V que necesitaría para funcionar. Pero si quieren que el robot que desarrollaron sea autónomo, van a tener que independizarse del cable USB. Los cables del conector de la batería se conectan a los pines GND y VIN de la placa.



A continuación deberán conectar la alimentación de los motores. En este ejemplo se utiliza un portapilas de cuatro elementos AA, pero se puede usar cualquier configuración que provee entre 6V y 12V. Es importante notar que este portapilas se conecta directamente a los pines de alimentación del puente H. Sin embargo, el cable que está conectado a GND (tierra) tiene que estar conectado también al pin GND de la placa que utilicen.

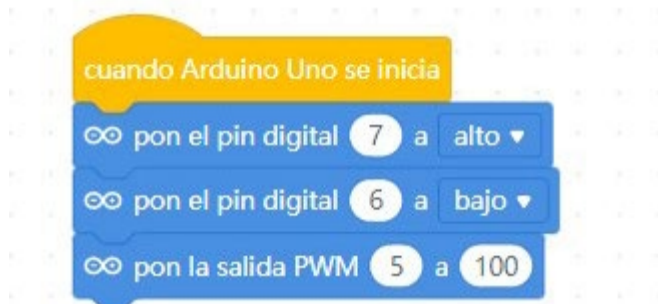


Por último, escriban el siguiente código en el entorno de programación de Arduino y grábenlo en la placa. Pueden empezar programando un motor solo, para asegurarse de que está funcionando bien, y luego agregar el código del segundo motor. Los pines que se utilizan para el primer motor son los numerados como 5, 6 y 7 y son todos pines digitales. Estos tienen que ser configurados como salidas dentro de la función `setup()`. En el código que está a continuación, se utilizan los pines digitales 6 y 7 de la placa Arduino para indicarle al motor en qué dirección girar y el pin 5 para indicarle la velocidad. El pin conectado a ENB tiene que soportar PWM para poder asignarle un valor entre 0 y 255, que representará la velocidad de giro.

Código para programar la placa

```
void setup()
{
  pinMode(7, OUTPUT); //IN3
  pinMode(6, OUTPUT); //IN4
  pinMode(5, OUTPUT); //ENB
}
void loop()
{
  digitalWrite(7, HIGH); //IN3
  digitalWrite(6, LOW); //IN4
  analogWrite(5, 100); //ENB
}
```

En el caso de programar la placa con [mBlock](#), la siguiente imagen se corresponde con la programación anterior:



Para finalizar, se muestra a continuación el código necesario para que los dos motores del robot funcionen con la misma velocidad y al mismo tiempo. La velocidad está establecida en 100 y este valor tiene que estar comprendido entre 0 (frenar) y 255 (velocidad máxima del motor). Por último, es importante notar que, una vez programada la placa, es posible que el robot no avance en línea recta “a la perfección”. En tal caso, a partir de la observación, se puede elegir un motor para que avance un poco más rápido que el otro y así se consiga que todo el conjunto avance recto.

Código para programar la placa

```
void setup()
{
  pinMode(7, OUTPUT); //IN3
  pinMode(6, OUTPUT); //IN4
  pinMode(5, OUTPUT); //ENB

  pinMode(11, OUTPUT); //IN1
  pinMode(10, OUTPUT); //IN2
  pinMode(9, OUTPUT); //ENA
}

void loop()
{
  digitalWrite(7, HIGH); //IN3
  digitalWrite(6, LOW); //IN4
  analogWrite(5, 100); //ENB

  digitalWrite(11, HIGH); //IN1
  digitalWrite(10, LOW); //IN2
  analogWrite(9, 100); //ENA
}
```

En el caso de programar la placa con [mBlock](#), la siguiente imagen corresponde con la programación anterior:



Listado de imágenes

Página 7. Imagen del "Simulador carrera de autos" programado en Scratch.

Disponible en <https://scratch.mit.edu/projects/198879818/editor>

Página 8. Imagen del "Simulador carrera de autos" programado en Scratch.

Página 10. Código del "Simulador carrera de autos" programado en Scratch.

Página 11. Editor de imágenes del "Simulador carrera de autos" programado en Scratch.

Página 14. Placa controladora. Aporte del equipo ETP, Nivel Secundario, de la Gerencia Operativa de Currículum.

Página 16. Robot móvil con placa controladora programable. Aporte del equipo ETP, Nivel Secundario, de la Gerencia Operativa de Currículum.

Página 20. Esquemas de placa Arduino, puente H y motor de corriente continua. Aporte del equipo ETP, Nivel Secundario, de la Gerencia Operativa de Currículum.

Página 21. Esquema de batería de 9V, placa Arduino, puente H y motores de corriente continua. Aporte del equipo ETP, Nivel Secundario, de la Gerencia Operativa de Currículum.

Página 22. Esquema de batería de 9V, placa Arduino, puente H, motores de corriente continua y portapilas AAA. Aporte del equipo ETP, Nivel Secundario, de la Gerencia Operativa de Currículum.

