

# ESPAÑOL - MIXE

## Una introducción al

# FASCINANTE MUNDO DE LA ROBÓTICA

Responsable técnico:  
Jesús Santiguillo Salinas

Diseño e ilustración:  
Melisa del C. Muñoz Vázquez

Traducción:  
Vanessa N. Gómez Jiménez



ESPAÑOL - MIXE

Una introducción al

# FASCINANTE MUNDO DE LA ROBÓTICA



Primera edición, Oaxaca, México

*Una introducción al fascinante mundo de la robótica, primera edición*

Universidad del Papaloapan (UNPA)

Consejo Nacional de Ciencia y Tecnología (CONACYT)

© Por autoría, Jesús Santiaguillo Salinas

© Diseño e ilustraciones, Melisa del Carmen Muñoz Vázquez

© Hiram Nezahualcoyothl García Lozano

Por la compilación

© Rafael Fernando González Zarate

Por la adaptación

© José Luis Nájera Sánchez

Por la recopilación

© Luis Alberto Hernández Zuccolotto

Por la corrección

© Vanessa Nayeli Gómez Jiménez

© Leticia Gómez Jiménez

© Alfonso Díaz Vargas

Por la traducción

ISBN 978-607-9320-50-8

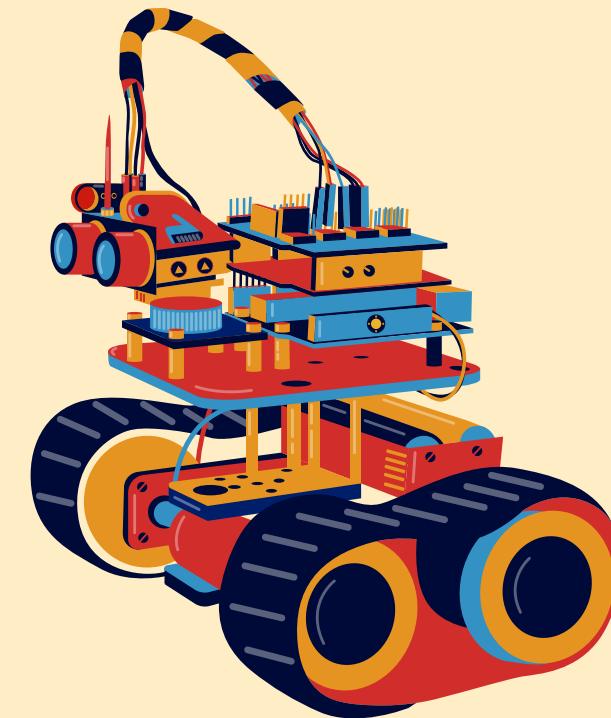
© Cisnegro ] lectores de alto riesgo [

Queda prohibida la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, la fotocopia o la grabación sin la previa autorización por escrito de los editores.

ESPAÑOL - MIXE

Una introducción al

# FASCINANTE MUNDO DE LA ROBÓTICA



# Índice



## Introducción

## 01 CAPÍTULO

### Introducción a la Robótica

#### 1.1 Historia de la robótica

1.1.1 Robots en el cine

1.1.2 Repaso histórico de la robótica

#### 1.2 Clasificación de la robótica

1.2.1 Robots manipuladores

1.2.2 Robots móviles

1.2.2.1 Robots terrestres

1.2.2.2 Robots acuáticos

1.2.2.3 Robots aéreos

#### 1.3 Aplicaciones de la robótica

1.3.1 Robots de telemando

1.3.2 Robots en la medicina

1.3.3 Robots en la industria

1.3.4 Trajes de rehabilitación y combate

1.3.5 Robots de Boston Dynamics

06

08

10

11

12

16

17

22

24

25

25

26

26

28

30

32

34

## 02 CAPÍTULO

### Componentes de un robot móvil

#### 2.1 Nuestro robot móvil

36

#### 2.2 Componentes de nuestro robot móvil

38

##### 2.2.1 Controlador

40

2.2.1.1 Arduino - Nano

40

##### 2.2.2 Dispositivos de Entrada-Salida

41

##### 2.2.3 Sensores

42

2.2.3.1 Sensor óptico

42

2.2.3.2 Sensor ultrasónico

42

##### 2.2.4 Actuadores

43

2.2.4.1 Puente H 1293D

43

2.2.4.2 Motor reductor con llanta

43

## 03 CAPÍTULO

### Programación básica del robot móvil

44

#### 3.1 Programación en Arduino

46

#### 3.2 Programas para el uso de los componentes

52

3.2.1 Mi primer programa Blink

52

3.2.2 Programa para el uso del LCD

53

3.2.3 Programa para el uso de los motores

54

3.2.6 Programa para el uso de los sensores ópticos

55

3.2.4 Programa para el uso del sensor ultrasónico

56

3.2.5 Programa para el uso de los botones

57

#### 3.3 Programas para tareas específicas

58

3.3.1 Robot seguidor de líneas

58

3.3.2 Robot autónomo evasor de obstáculos

60

# Introducción



Este elibro surge como una propuesta de proyecto de la Universidad del Papaloapan dentro de la Convocatoria 2021 para la Elaboración de propuestas de Proyectos para el Fomento y Fortalecimiento de las Vocaciones Científicas del Consejo Nacional de Ciencia y Tecnología (CONACyT).

El objetivo de este elibro es acercar a los grupos vulnerables del país, en particular a los niños, niñas y jóvenes, al conocimiento científico y tecnológico de vanguardia, contribuyendo, por medio de la elaboración de material digital, interactivo y de acceso en línea sobre robótica, a fomentar y fortalecer las vocaciones científicas motivando la creatividad e innovación de esta población, con el fin de consolidar una cultura del conocimiento.

Se pretende beneficiar a comunidades indígenas o afromexicanas, población marginada socioeconómica y/o geográficamente, principalmente jóvenes entre 12 a 24 años, facilitando el acceso al conocimiento y tecnología referente a la robótica, considerando también que la gran cantidad de la población objetivo habla una lengua materna distinta al español o no cuentan con los recursos económicos suficientes.

Este elibro consta de 3 secciones: Introducción a la Robótica, Componentes de un Robot y Programación Básica de Robots.

En la Sección Introducción a la Robótica se hablará de los principios de la robótica, su evolución a través de los años y del estado del arte actual de la robótica. Se presentarán los diversos tipos de robots manipuladores y móviles, así como sus características básicas como su configuración y espacio de trabajo, para el caso de los robots manipuladores; el sistema de locomoción y dirección para robots móviles.

En la Sección Componentes de un Robot se presenta un módulo robótico de robot móvil diferencial construido por la Universidad del Papaloapan, se explicarán cada uno de los componentes que lo forman como lo son sensores, actuadores y placas electrónicas de control.

Por último, en la Sección de Programación Básica de Robots se darán las bases de programación del robot móvil diferencial en Arduino y se presentarán los respectivos programas para la utilización de cada uno de los componentes electrónicos que se emplean en el robot.

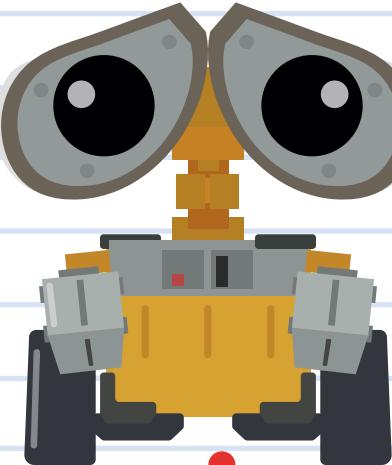
Para finalizar la Sección, se presentarán programas donde se extienden las funcionalidades del robot para realizar tareas más complejas como son: seguimiento de líneas, y movimiento autónomo con evasión de objetos.

# Cap. 01

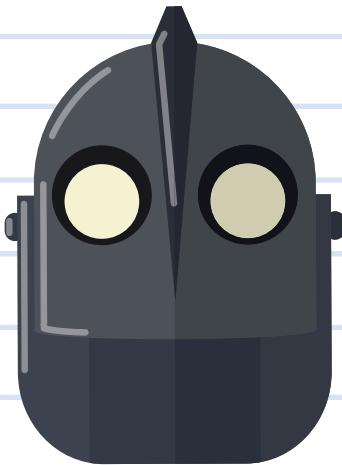
## Introducción a la robótica

# Historia

**Robots en el cine**



WALL-E es un robot diseñado para limpiar la basura que cubre a la Tierra después de que fuese devastada y abandonada por el ser humano.



El Gigante de Acero es un robot proveniente del espacio capaz de autorepararse, que entabla una gran amistad con un niño de 9 años llamado Hogarth.

Durante los últimos años, los robots no solo nos han maravillado con extraordinarias historias de ciencia ficción si no que, utilizando como base las universidades y los centros de investigación, han incursionado en muchas ramas de la industria. En la literatura o en el cine, los robots tienen comportamiento humano, y más aún, ellos tienen sentimientos de amistad, odio o amor.



En la realidad los robots solo son sistemas mecánicos o electromecánicos controlados, diseñados para desempeñar tareas específicas o generales. Si bien, estos robots pueden tener forma humana o humanoide, realizan algunas tareas en forma más precisa y por más tiempo que una persona.

Bumblebee es un personaje ficticio perteneciente al universo Transformers. Es llamado el "pequeño hermano" de los Autobots.  
Ocupación: Espionaje, Exploración.

*El dotar a uno de estos mecanismos de cualquier tipo de sentimiento no se ha logrado aún, pero...*

**¿quién sabe en el futuro?**

# Historia

Existen muchas definiciones del término “robótica”. Sin embargo, a nuestro parecer, la siguiente definición, del Dr. Fernando Reyes Cortés, en su libro “Robótica”, es la más adecuada debido a su sencillez.

***“La robótica se define como una disciplina científica que se basa en la investigación y desarrollo de sistemas mecánicos, llamados robots, que se diseñan para realizar una serie de aplicaciones industriales, comerciales, entre otras”***

### Autómatas

Ejemplo:  
Autómata de Jacques (1739)

Los griegos llamaban **Autómata** a las máquinas que imitan la figura y movimientos de un ser animado.

No se tiene registro exacto de cuál fue el primer autómata, pero se cree que los primeros se remontan al antiguo Egipto, aproximadamente en el año 1300 A.C., donde sistemas hidráulicos y mecánicos permitían a los sacerdotes mover a sus gigantescos dioses.



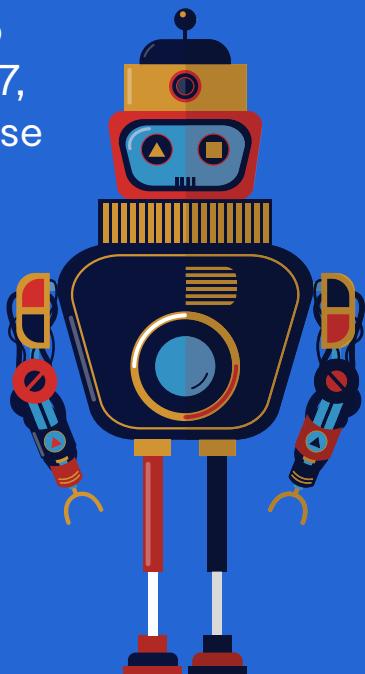
El término Robot proviene del checo **ROBOTA** y lo usó por primera vez el escritor Karel Capek en 1917, en su obra Rossum's Universal Robots, para referirse a máquinas con forma humanoide. Esta palabra fue traducida al inglés como robot.

Otros escritores retomaron las narrativas sobre robots, por ejemplo Isaac Asimov (quien acuñó el término robótica) que dentro de su extensa bibliografía se encuentra el libro “yo robot”, en el cual se presentan las tres leyes de la robótica, que hasta hoy se siguen utilizando.

El sector industrial se dió cuenta del gran potencial que tenían estos sistemas, al ser incorporados en sus procesos.

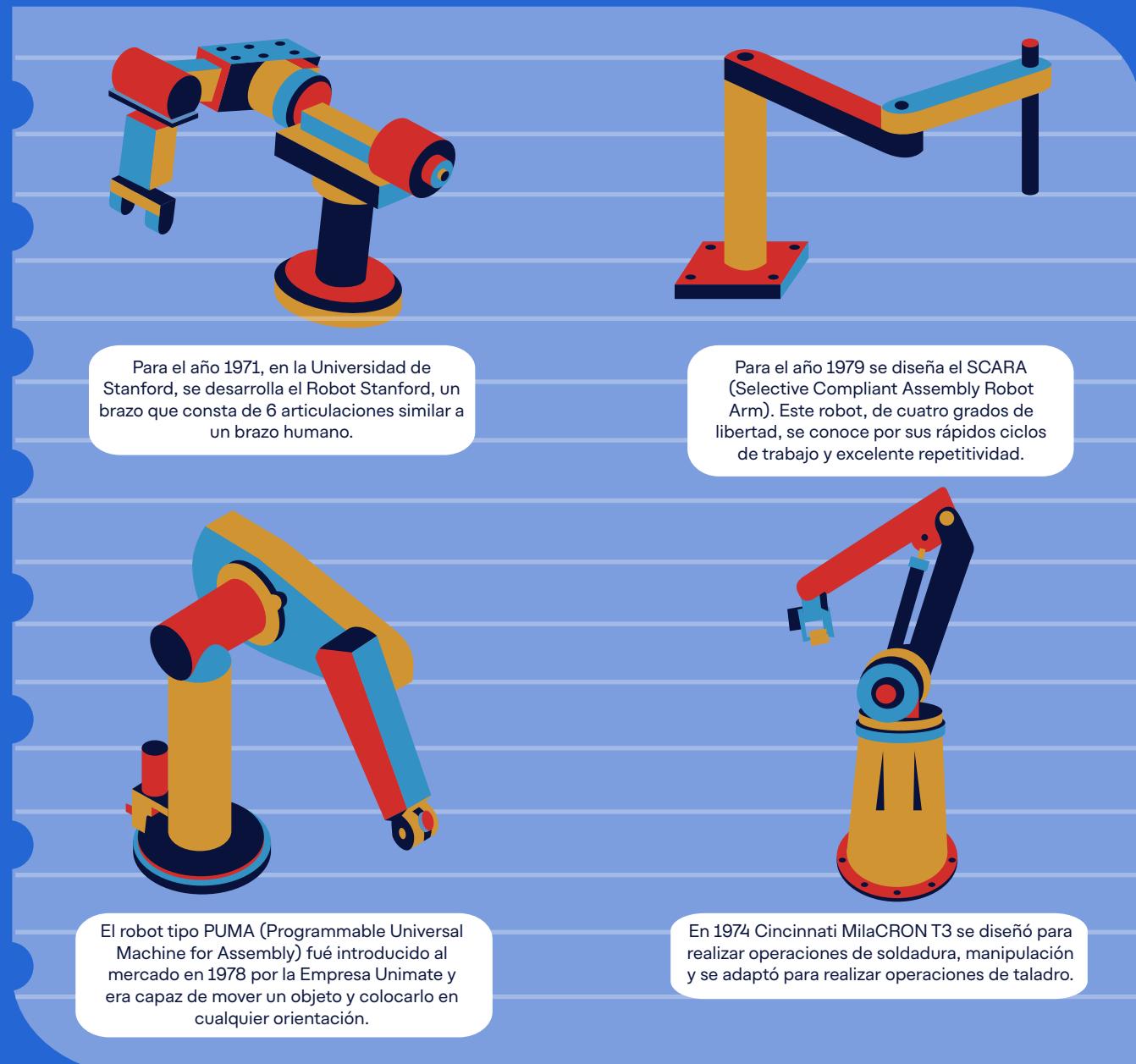
- En el año 1961, el robot Unimate fué instalado en una cadena de montaje en la empresa General Motors.
- En 1963 se diseña el primer Robot Palletizing (robots que sirven para apilar bienes sobre una tarima).

*Desde ese momento, los robots se integraron al sector productivo.*



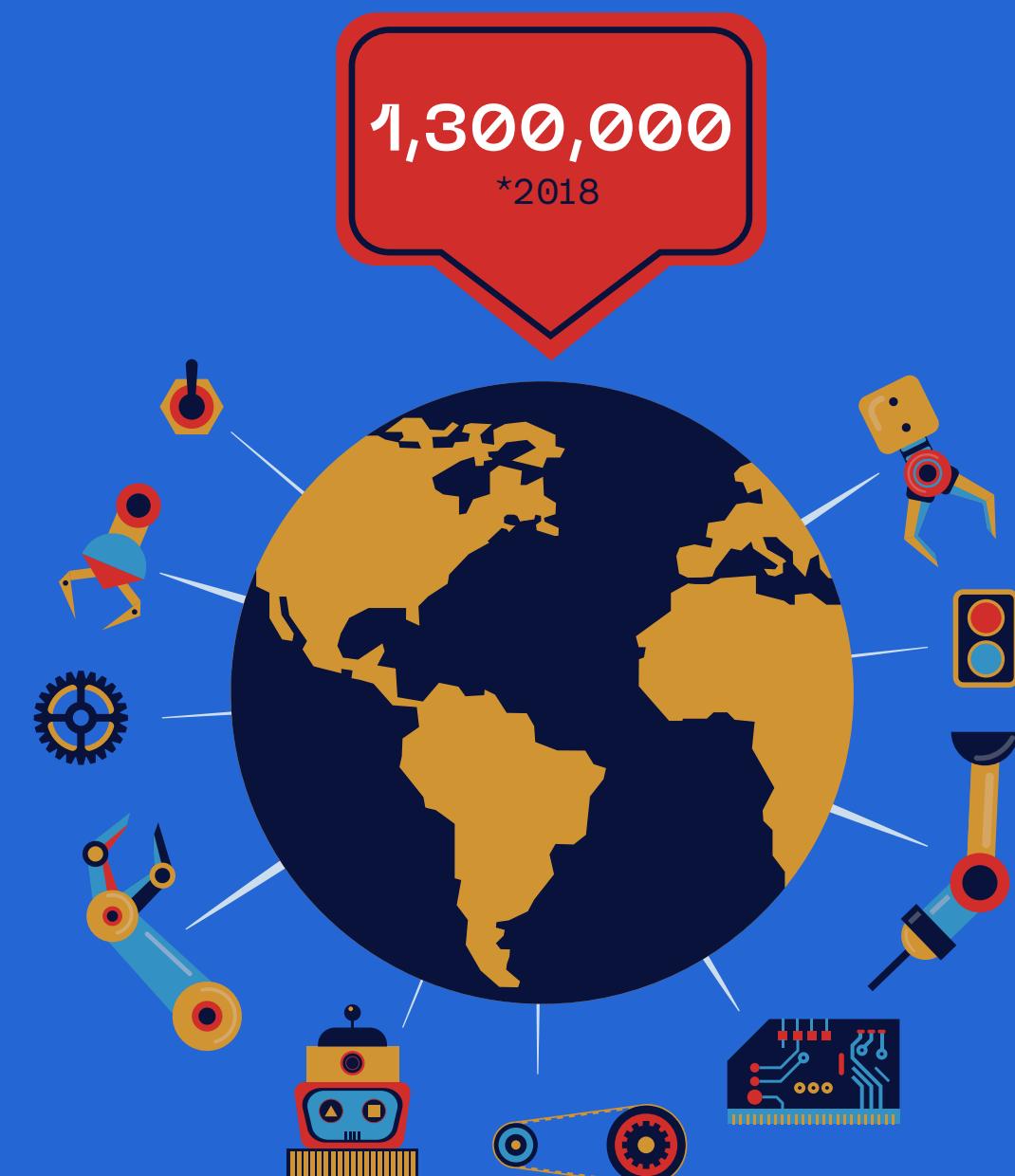
# Historia

Por su parte, las universidades de todo el mundo encuentran a la robótica como un área de investigación joven y con mucho potencial. Gran cantidad de robots fueron diseñados para resolver distintos tipos de problemas:



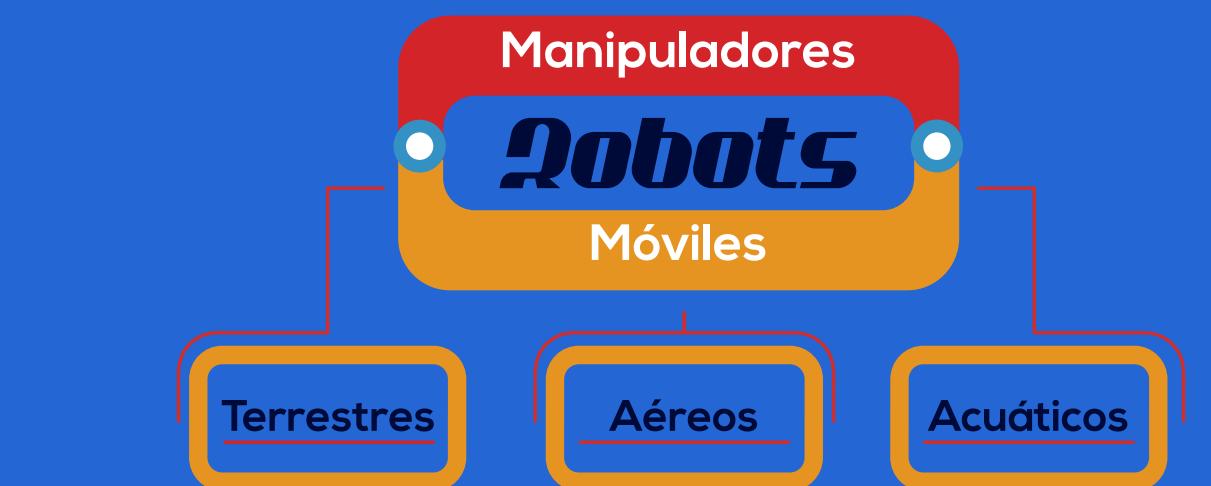
El periódico español El País, en su publicación de la sección de economía del 7 de febrero del 2016, llama a los robots como la cuarta revolución industrial y compara el efecto que estos tienen en la industria actual con el que tuvo la máquina de vapor en la época preindustrial.

De esta forma, hoy en día es área clave y estratégica para todo país en desarrollo. Hasta el 2008 se encontraban operando más de 1,000,000 de robots en el mundo y se estimó que en el 2018 existían 1.3 millones de robots industriales.



# Clasificación 1.2

Existen diversas formas de organizar y caracterizar la gran variedad de robots que existen. Una de ellas es la que se muestra en la siguiente figura.

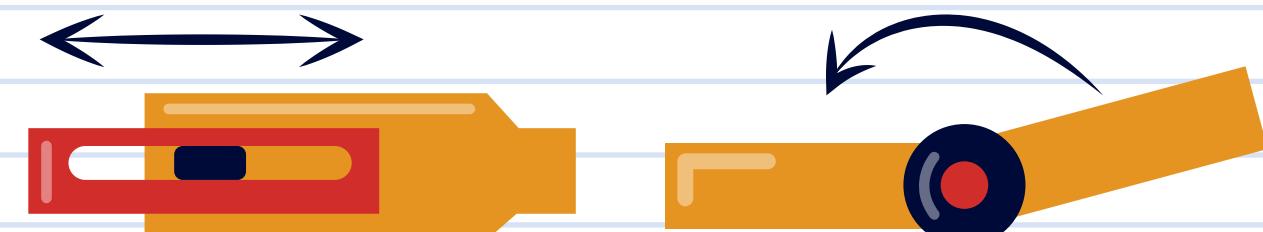


Un robot manipulador está constituido principalmente por tres elementos: eslabones, articulaciones y un efecto final.

Los eslabones son elementos rígidos. Son los que le dan soporte y estructura al robot. El primer eslabón es fijo y el último eslabón es conocido como herramienta o efecto final. Para unir dos eslabones es necesario utilizar una articulación.

## Manipuladores

La forma de movimiento que tienen las articulaciones puede ser de tres tipos:



Prismático

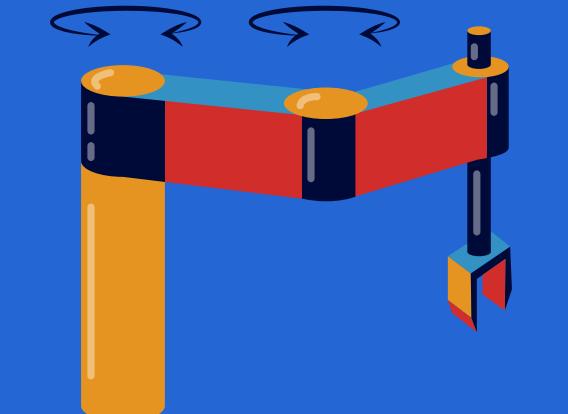
Rotacional



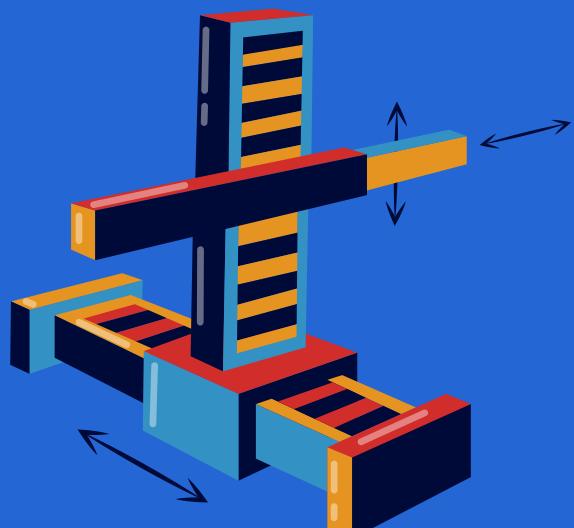
Esférico

# Clasificación

## Manipuladores



Cuando dos eslabones están conectados por medio de una articulación rotacional, el movimiento que pueden tener es similar al de una puerta con bisagras. Cuando un eslabón está fijo, el otro puede girar en sentido horario o anti-horario, con centro en la articulación.



El eslabón prismático permite el movimiento lineal entre dos eslabones. Es muy parecido al desplazamiento de una ventana corrediza. El movimiento solo es hacia adelante o hacia atrás.

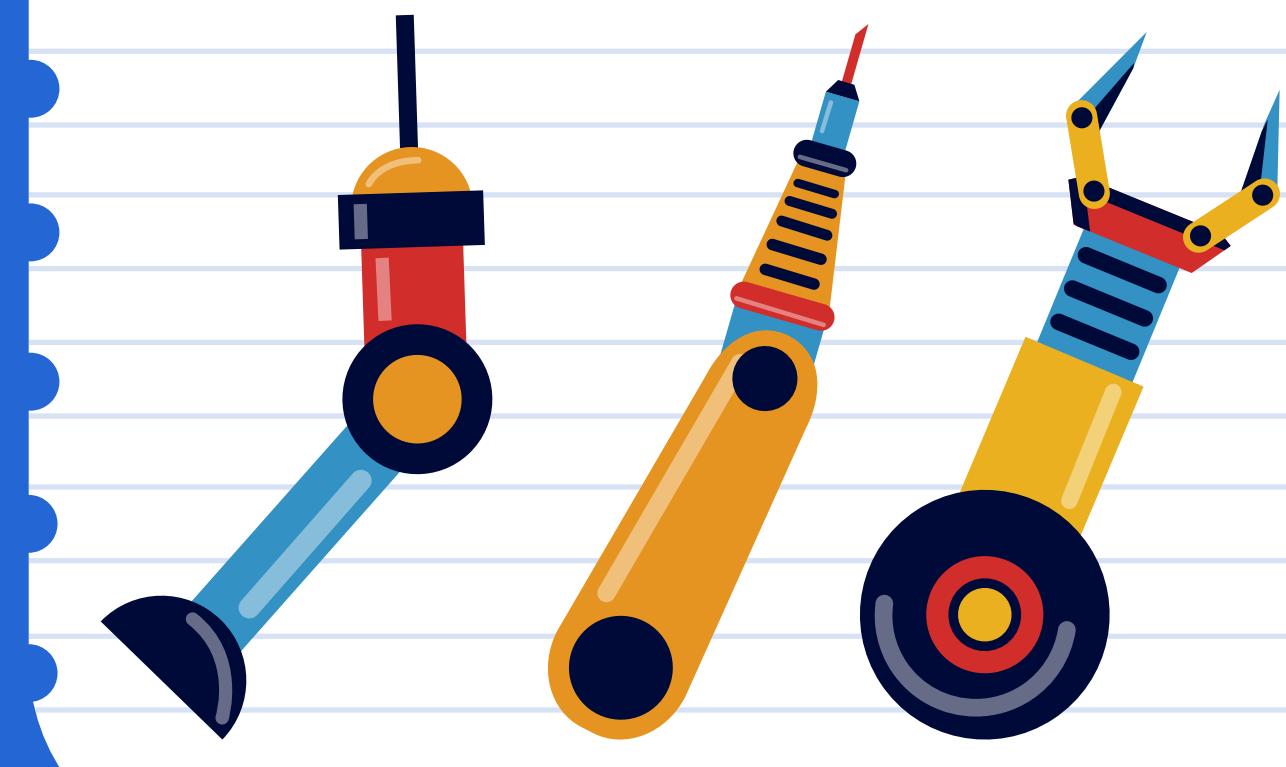
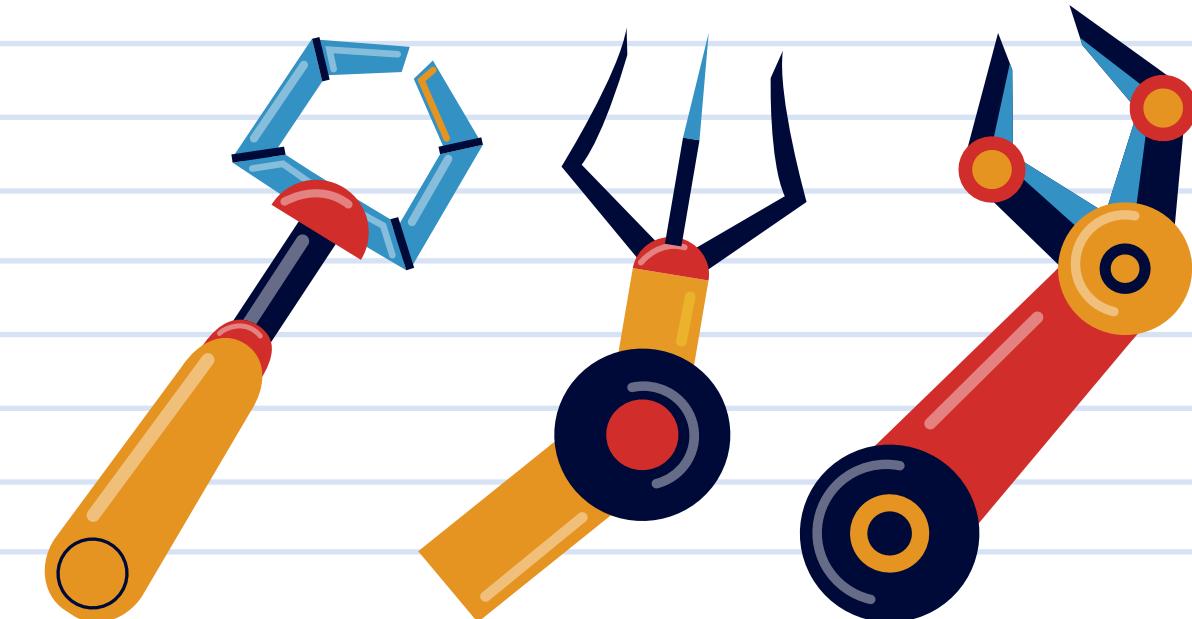


La articulación esférica permite realizar tres rotaciones, similar a los movimientos que podemos hacer con la cabeza o con la muñeca. Por ejemplo, con nuestra cabeza podemos girar de izquierda a derecha (señal de no). Podemos realizar movimientos laterales de un lado a otro y realizar movimiento del cuello de adelante hacia atrás (indicación de afirmación).

## Manipuladores

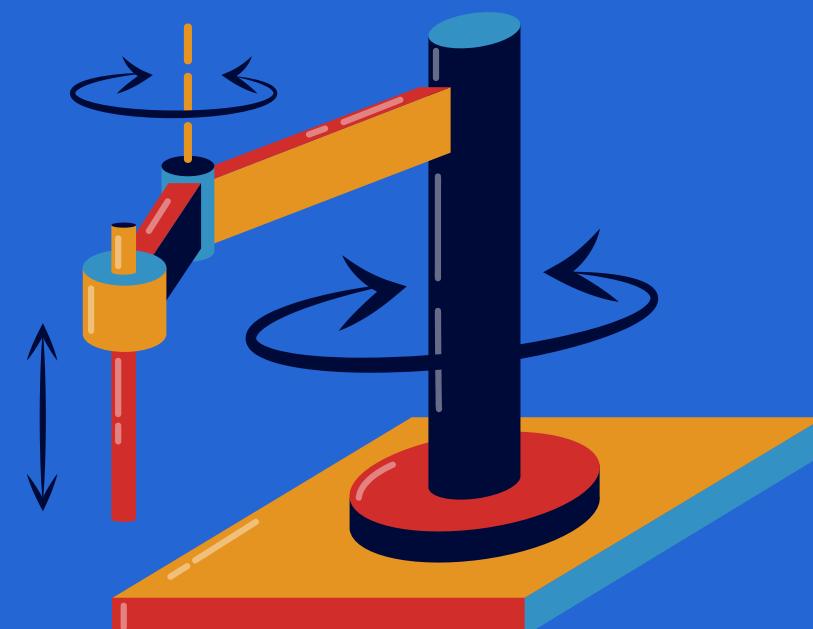
### Efecto Final

Herramienta especial que permite al robot realizar una aplicación particular.



# Clasificación

## Tipos de robots manipuladores



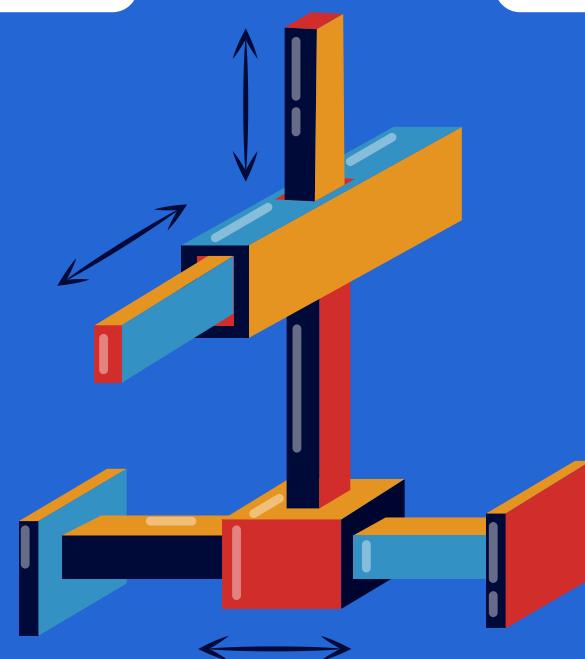
Robot SCARA



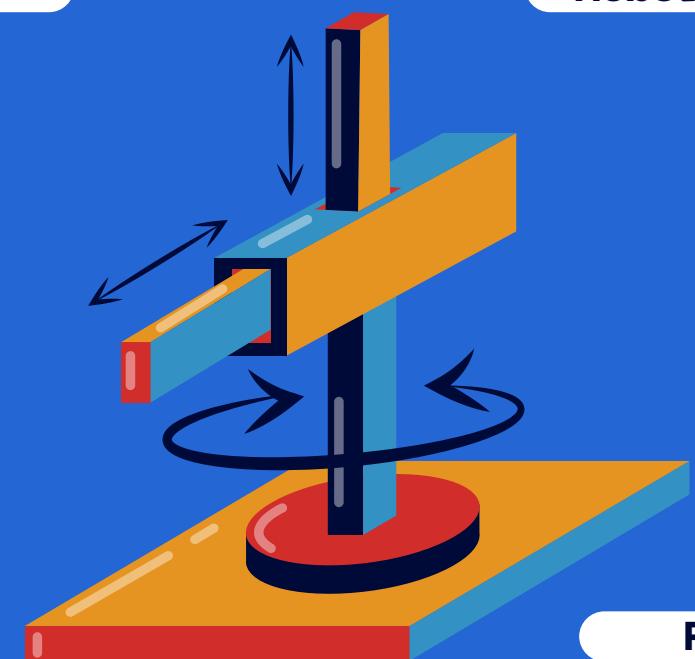
Robot Esférico



Robot Antropomórfico



Robot Cartesiano



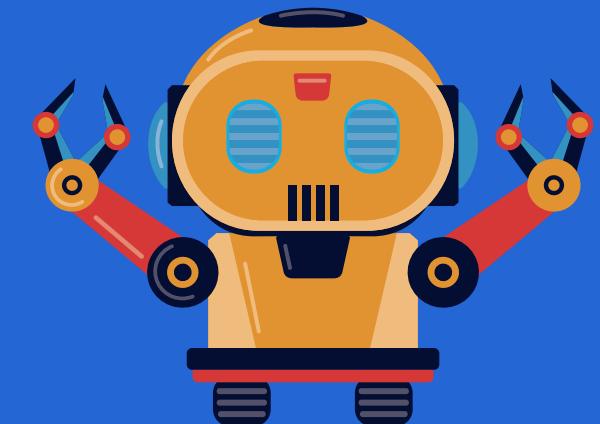
Robot Cilíndrico

# Clasificación

## Robot móvil

Como ya se mencionó, existe gran variedad de robots móviles. Ahora nos vamos a centrar en los robots móviles terrestres con ruedas. Estos robots son la solución más simple para lograr desplazamiento en superficies planas y, en el caso de no tener obstáculos, es posible alcanzar velocidades relativamente altas.

Los robots móviles con ruedas pueden ser clasificados por medio de la geometría en que se encuentran sus ruedas, así como la forma en que dan tracción y dirección a las mismas. Esto se conoce como la configuración de su sistema de locomoción.



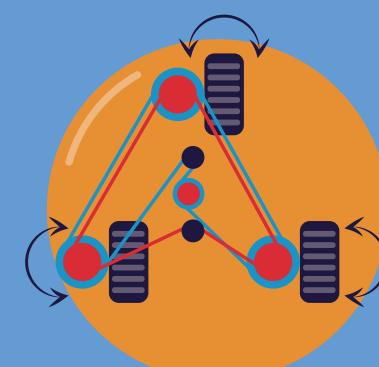
### Tipos de Robot Móvil Terrestre



#### Robot con ruedas omnidireccionales

Las ruedas omnidireccionales son ruedas de diseños complejos que permiten una movilidad en varias direcciones. Estas ruedas fueron patentadas en 1919 por J. Grabowiecki en Estados Unidos y su diseño ha ido evolucionando hasta la fecha.

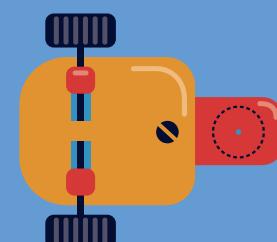
La característica principal de este tipo de robot es el desplazamiento sin cambio de orientación del robot.



#### Configuración Síncrona

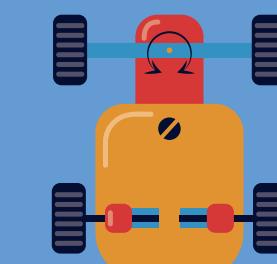
En esta configuración, las ruedas (generalmente tres) son utilizadas para dar dirección y tracción al robot. Para avanzar, las tres ruedas giran a la misma velocidad. Para cambiar de dirección, las tres ruedas se orientan en el mismo sentido al mismo tiempo. Esta sincronía se logra por medio de dos bandas.

### Tipos de Robot Móvil Terrestre



#### Tracción Diferencial

En esta configuración dos ruedas conectadas cada una a su propio motor (tracción independiente), se montan en el robot a lo largo de un eje común. Por lo general una tercera rueda que gira libremente (rueda loca) es colocada en la parte trasera del robot, para mantener el equilibrio.



#### Configuración Ackerman

La configuración Ackerman está compuesta de cuatro ruedas, dos delanteras que dan dirección y dos traseras la que proporcionan la tracción del vehículo.



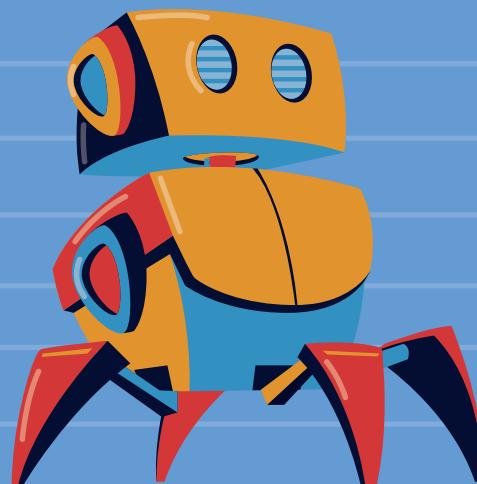
#### Triciclo clásico

En la configuración triciclo clásico la rueda delantera se utiliza tanto para dirección como para tracción. El eje trasero está constituido por dos ruedas laterales, paralelas de orientación fija, las cuales giran libremente.

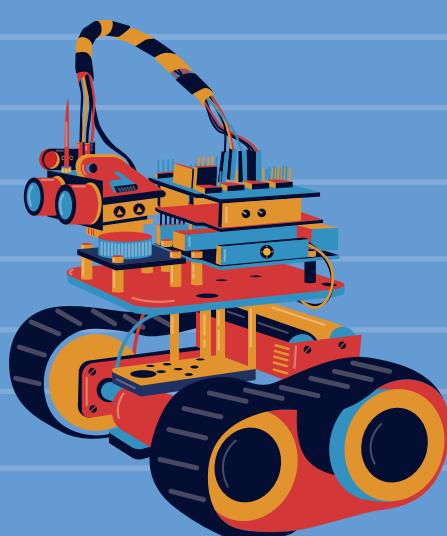
# Clasificación

## Otros Robots Móviles Terrestres

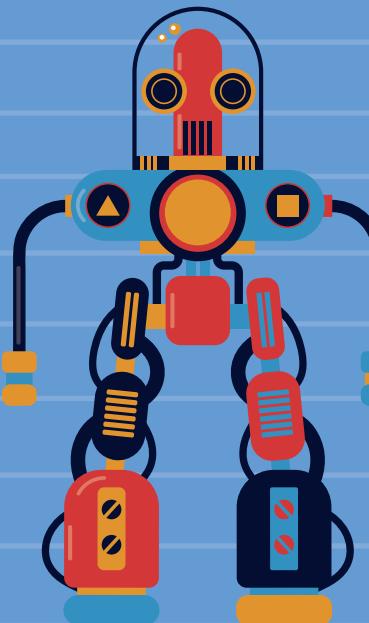
Robot con patas



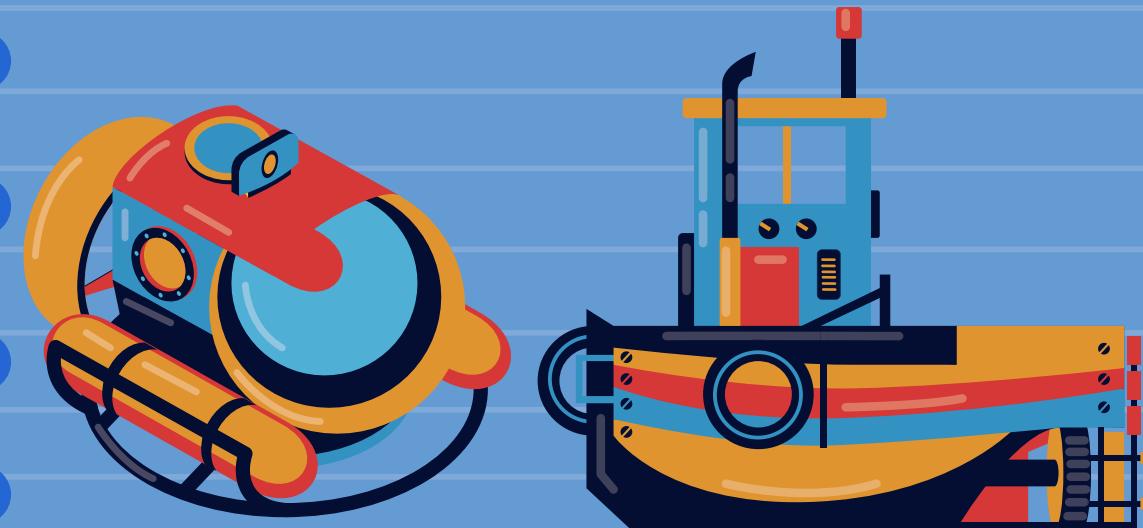
Robot oruga



Robot humanoide



## Robots Móviles Acuáticos



Submarino

Barcos

## Robots Móviles Aéreos



Ala rotativa



Ala fija

# Aplicaciones 1.3

Dada la gran evolución y desarrollo de la robótica, sus ramas de aplicación son muy amplias. Algunas de ellas son: Robots de telemundo, Asistencia médica, Asistencia en rehabilitación, Trajes de combate, Humanoides y Robots industriales. A continuación, se mostrarán cada una de ellas.

## Robots de telemundo

Estos robots son controlados de manera remota por un operador humano. Un ejemplo de ellos son los drones. Este tipo de vehículo aéreo no tripulado recibe su nombre del inglés drone que significa “Zumbido” y pueden ser utilizados en:

- **Eventos:** como partidos de fútbol, conciertos, periodismo, cine, entre otros.
- **Entregas:** de paquetería menor a 2.2 Kg por empresas como Fedex y Amazon.
- **Situaciones de emergencia:** acceso a zonas de difícil acceso, desastres naturales, búsqueda de personas, control de incendios, vigilancia, etc.
- **Zonas rurales:** monitoreo de grandes extensiones de terrenos agrícola, búsqueda de plagas en cultivos, malezas o el control de rebaños

En esta misma categoría se encuentran los robots que tienen la capacidad de adentrarse en las profundidades marinas. Con ellos se busca:

- *Preservar el ambiente marino*
- *La inspección del casco y los tanques internos en barcos*
- *La intervención en desastres marinos.*

Otros robots de telemundo son los utilizados para:

- *Demolición en espacios muy reducidos o peligrosos.*
- *Trabajo industrial pesado.*
- *Demolición a gran escala en grandes bóvedas o presas.*



Ejemplo:  
Robot aéreo de  
telemundo

# Aplicaciones



## Asistencia médica

### Robot PILLO

Entre los Robots de asistencia médica se encuentran los de almacenaje y distribución de medicamentos.

La empresa Pillo Health ha sido la encargada de crear al **Robot PILLO** sanitario que permite resolver dudas en materia de salud, realizar videoconferencias directas con médicos y dispensar los medicamentos que necesita tomar a diario el usuario a través de un algoritmo de reconocimiento facial.



### Robot Da Vinci

La robótica es de gran utilidad en la medicina; debido a que al ser programable y tener gran precisión, permite al especialista acceder de mejor manera a áreas de riesgo o áreas en donde no se debe generar error alguno.

Los robots quirúrgicos como el **Da Vinci**, permiten a los cirujanos realizar intervenciones complejas con resultados muy favorables, dichas intervenciones pueden ser tanto de exactitud como de fuerza.



# Aplicaciones

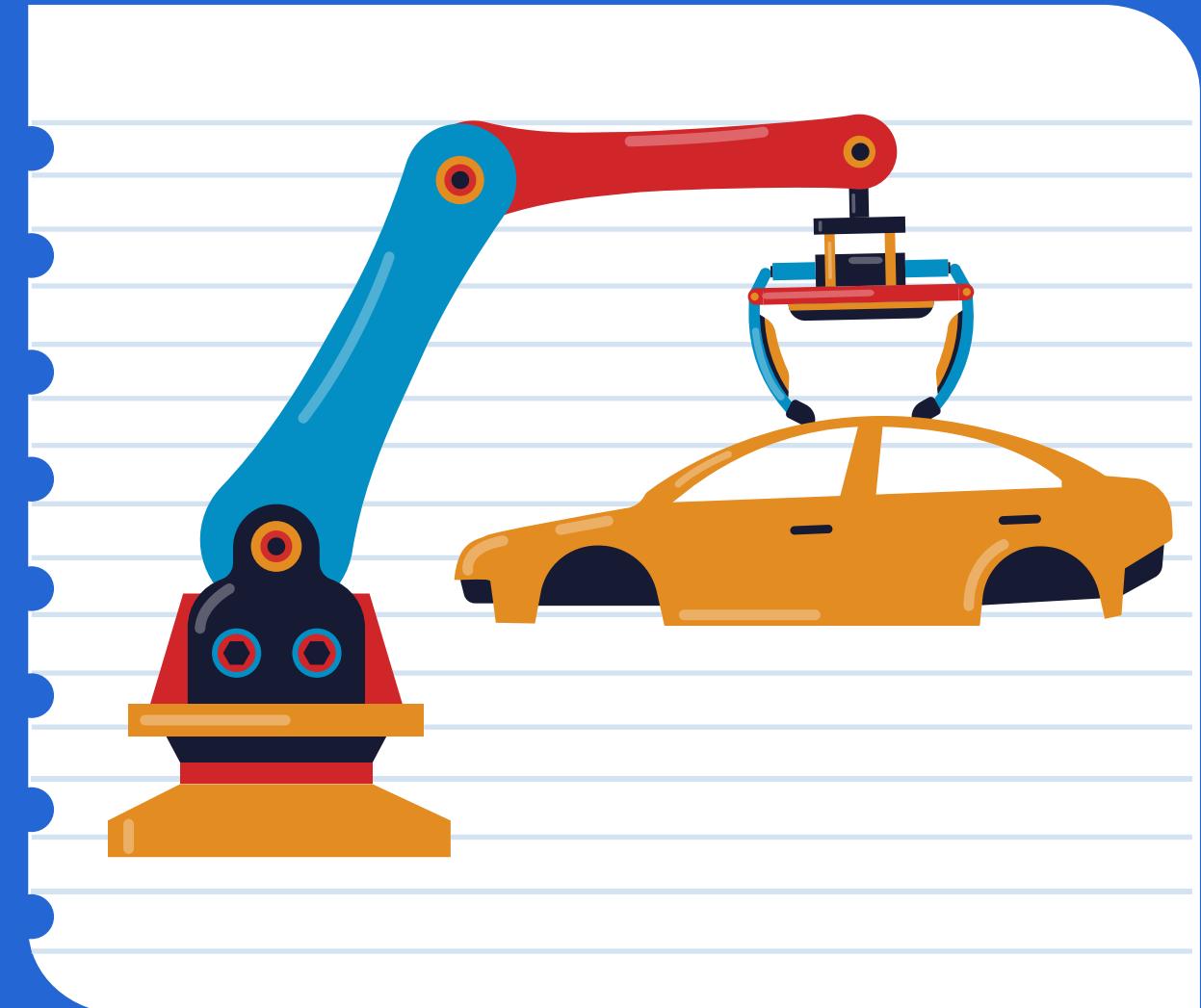
## Robots Industriales

La mayor parte de los robots industriales son brazos con controles que se manejan a través de una computadora. Estos robots tienen herramientas diferentes, según su trabajo. Algunos de sus usos regulares son la manipulación de materiales delicados, soldadura, carga, descarga y corte de elementos pesados, aplicación de productos como pintura, baños de metales o ácidos, medición de sólidos, líquido, gases, o aplicación de altas presiones, e incluso pueden funcionar para la supervisión del control de calidad de los materiales fabricados.

La robótica en la industria automotriz empezó a desarrollarse, sobre todo con la concepción de los modelos más grandes de camionetas, a principios de la década de los 90's . Los fabricantes de automóviles eran capaces de garantizar un producto uniforme con un mínimo de lesiones a los trabajadores. Actualmente el 50 % de todos los robots en uso se utilizan en la fabricación de automóviles.

El Instituto Americano de Robótica (RIA) postula la siguiente definición de robot industrial:

*Un robot industrial es un manipulador reprogramable multifuncional diseñado para mover materiales, piezas, herramientas o artefactos especiales, mediante movimientos variables programados para la ejecución de tareas en forma inteligente.*



De esta forma, muchas de las tareas que normalmente se realizaban por personas en forma artesanal, fueron sustituidas por este tipo de máquinas especializadas.

# Aplicaciones

## Asistencia de rehabilitación

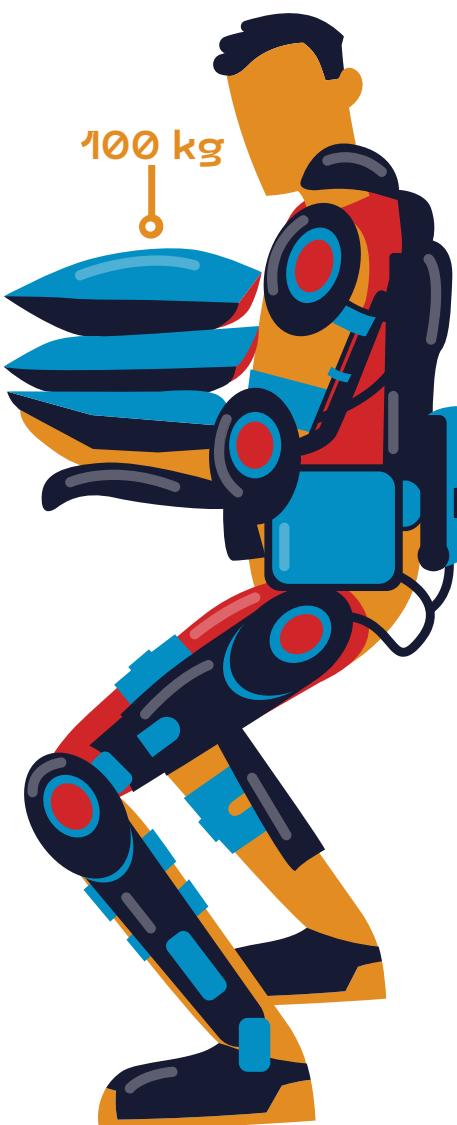
Este tipo de robots se conocen como exoesqueletos, esta tecnología permite a la persona que la usa realizar movimientos adecuados y con la fuerza necesaria según el diagnóstico médico. Estos robots tienen un papel muy importante en mantener a personas de edades avanzadas activas y saludables por más tiempo, ya sea en el trabajo o en casa.



## Trajes de combate

En Rusia se han presentado los nuevos trajes de combate para sus fuerzas especiales inspirados en ciencia ficción. Desarrollado por el Instituto Central de Investigación para la Construcción de Máquinas de Precisión, este peculiar traje esconde un exoesqueleto motorizado, protección balística, y hasta un casco con visor electrónico.

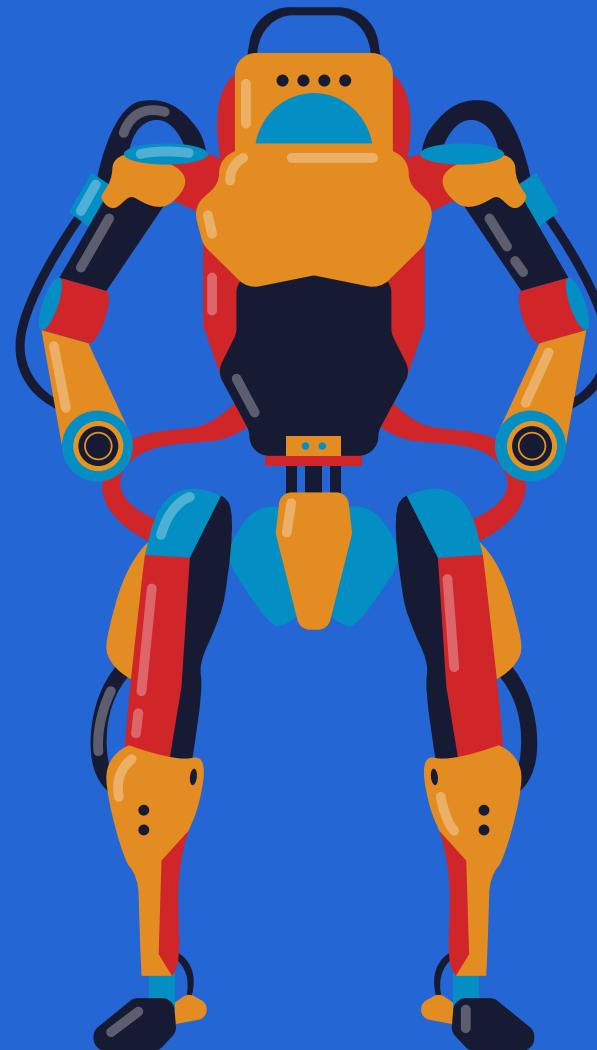
**El Exoesqueleto Asistencial Híbrido (HAL)** es un traje desarrollado por la Universidad Japonesa de Tsukuba y la empresa robótica CYBERDYNE. Ha sido diseñado para ampliar las capacidades físicas de sus usuarios, haciendo posible extender las capacidades del hombre a levantar 50 kilos con cada mano o más de cien con el impulso de las piernas.



# Aplicaciones

## Robots Humanoides y Droides

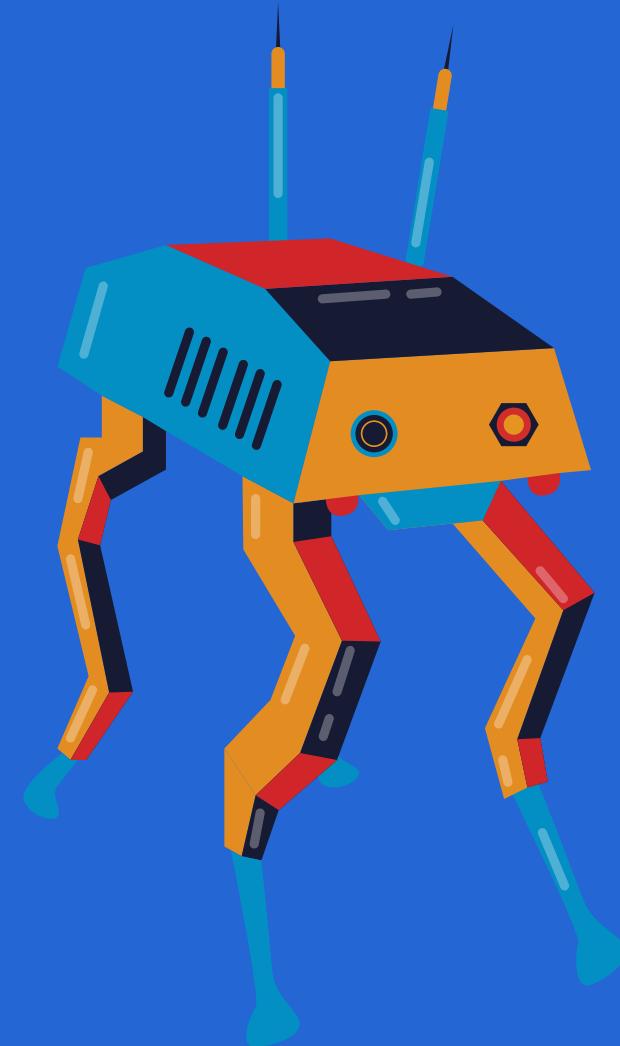
Boston Dynamics es una empresa de ingeniería y robótica que se especializa en la construcción de robots. En sus inicios trabajó en conjunto con el Instituto de Tecnología de Massachusetts (MIT), desarrollando robots que corrían y maniobraban como animales. Ahora buscan nuevos horizontes combinando los principios del control dinámico y el equilibrio con sofisticados diseños mecánicos, electrónica de vanguardia y software para percepción, navegación e inteligencia. Entre sus principales diseños se encuentran:



ATLAS



WILDCAT



BIGDOG



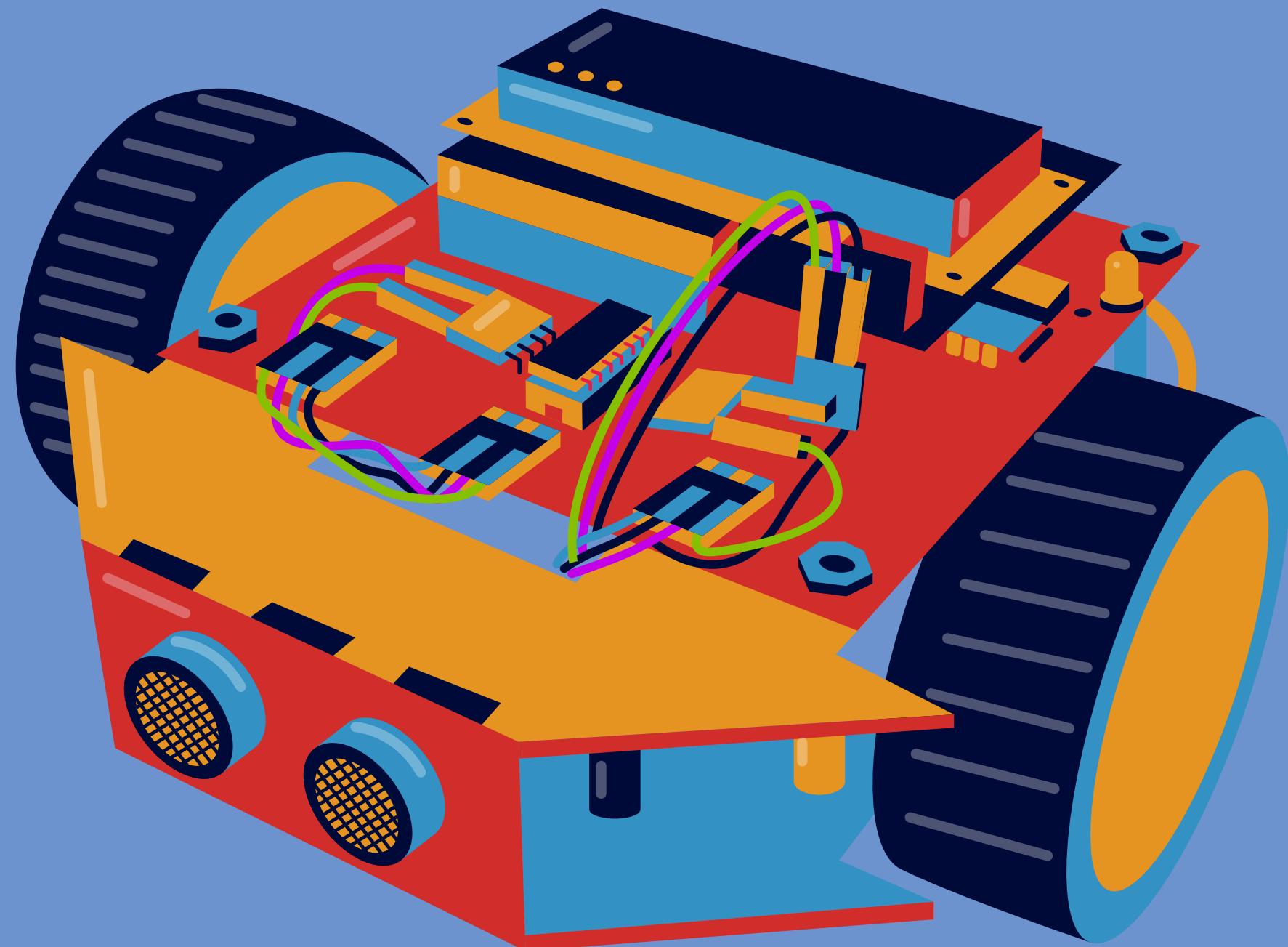
SPOT

Los invitamos a visitar la página [www.bostondynamics.com](http://www.bostondynamics.com) en donde encontrarán videos asombrosos de estos y otros robots.

Cap. 02

# Componentes de un robot móvil

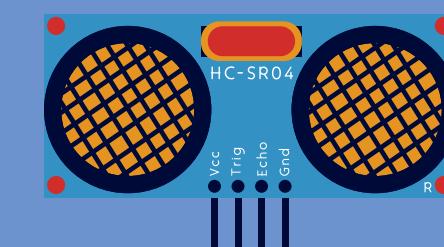
# Nuestro robot móvil



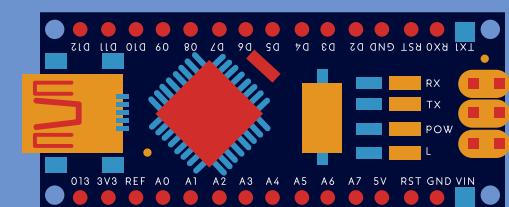
2.1

## COMPONENTES

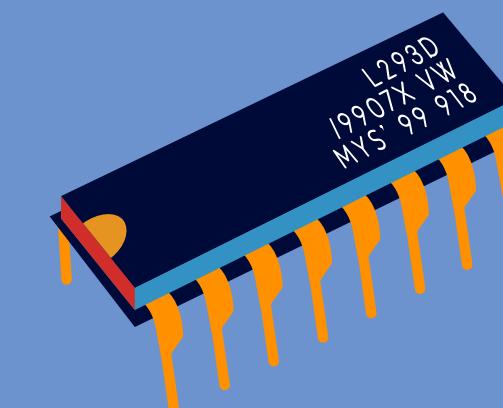
Sensor Ultrasónico  
SRF04



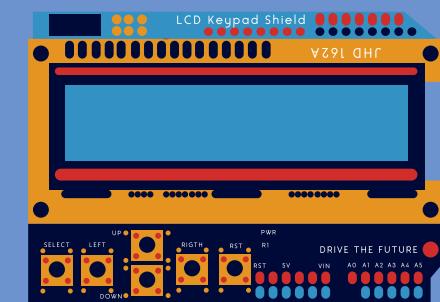
NANO



Puente H L293D



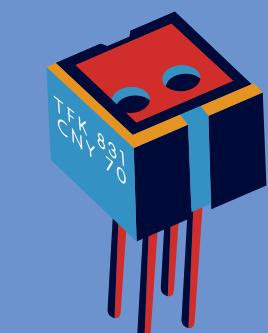
LCD Keypad Shield



Motor Reductor  
con Llanta



Sensor Óptico  
CNY70



# Componentes

2.2

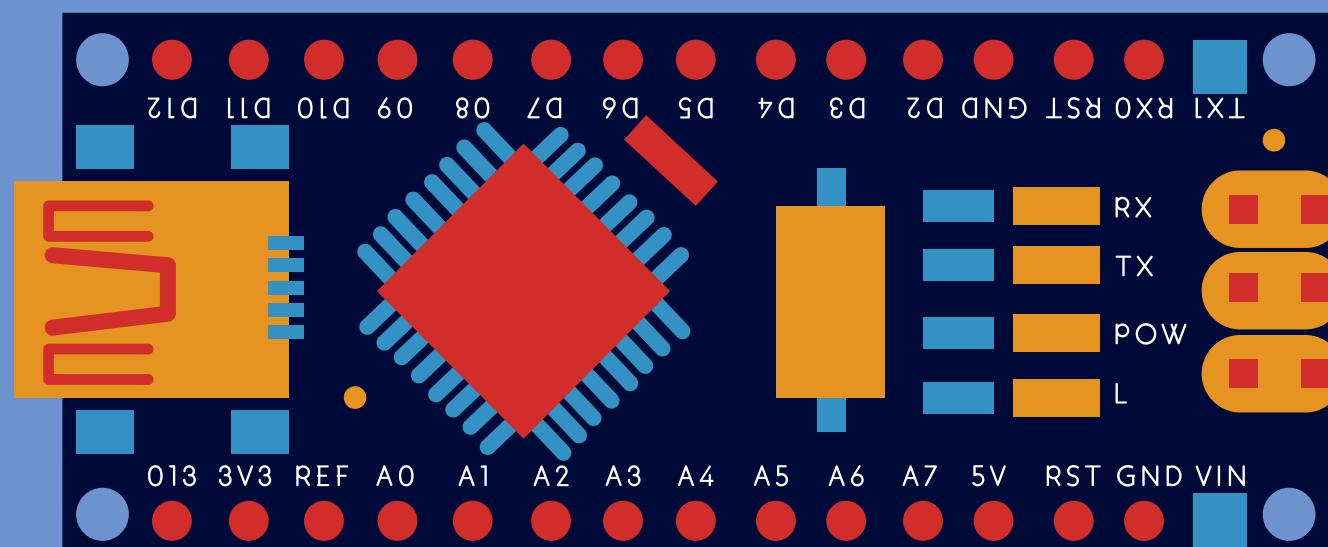
Controlador

Dispositivo de entrada/salida

## Arduino - NANO

Un controlador Arduino es una plataforma de prototipos interactivos. Está pensado para artistas, diseñadores, (como hobby) o para cualquiera interesado. En este módulo robótico se utiliza una plataforma de hardware libre basada en Arduino a la que llamaremos Controlador NANO.

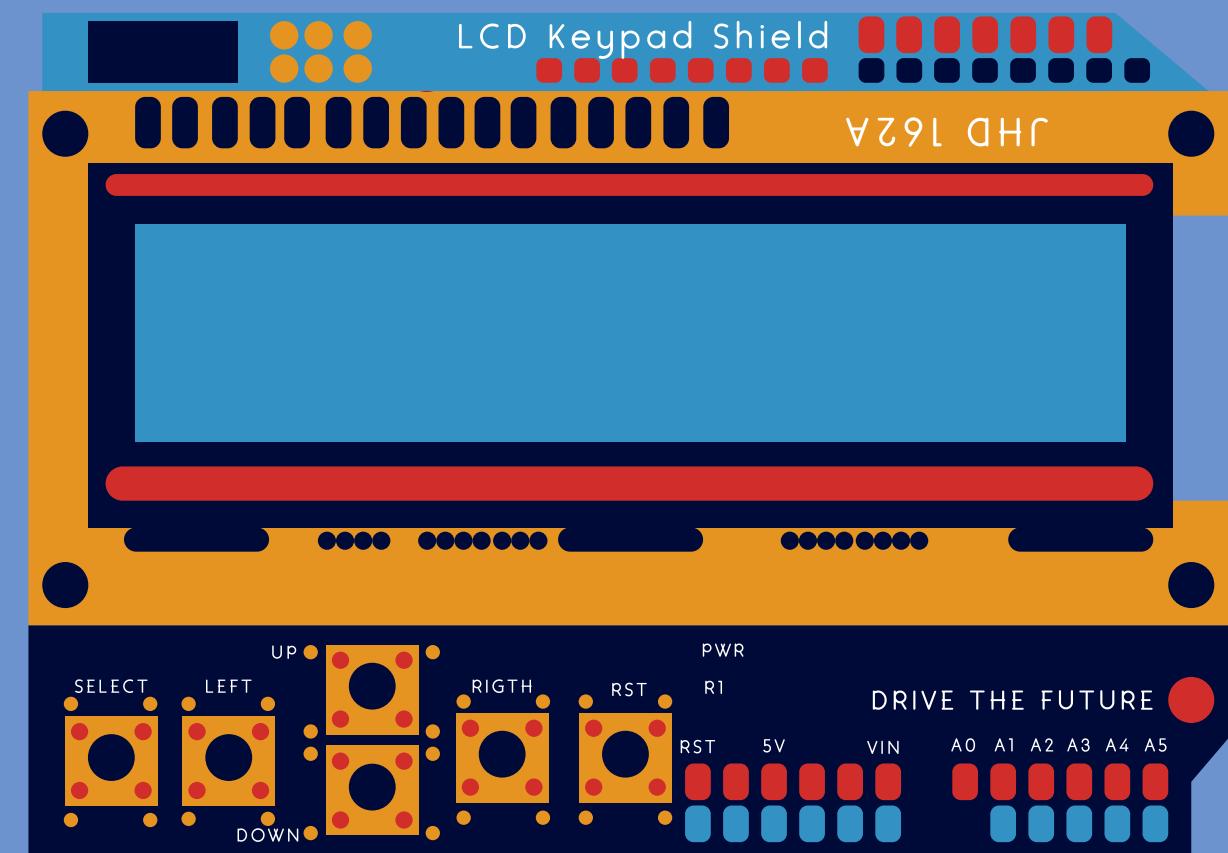
Se puede sentir el entorno mediante sensores y se pueden controlar luces, motores y otros artefactos. De esta manera, los proyectos basados o compatibles con Arduino, como el Controlador NANO, pueden interactuar con el entorno o funcionar por sí solos.



## LCD Keypad Shield

La LCD es un dispositivo diseñado para mostrar información en forma gráfica.

LCD significa Liquid Crystal Display (Display de cristal líquido). Esta cuenta con 5 botones para entrada de información mas un conjunto de terminales o conexiones.

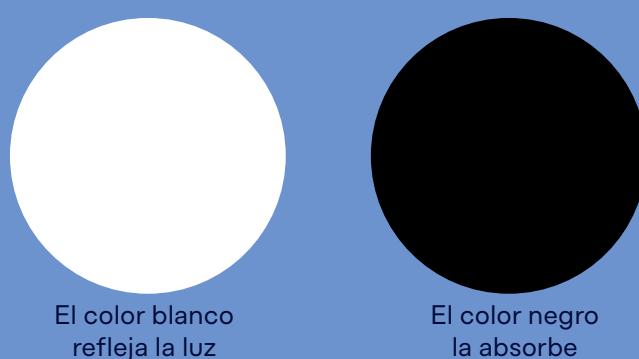
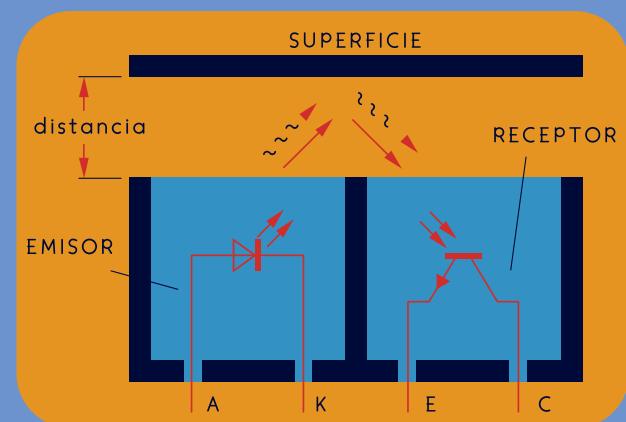


# Componentes

## Sensores

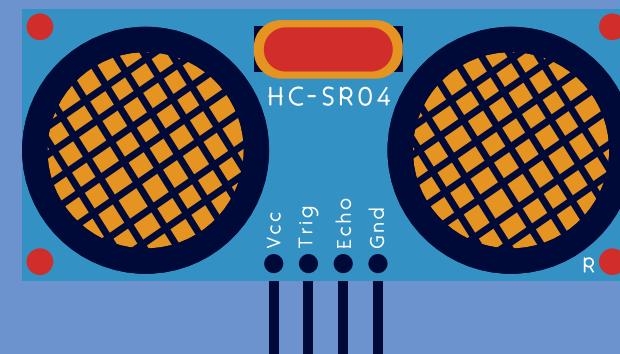
### Sensor Óptico CNY70

El Controlador NANO necesita un sensor parecido a los “ojos”. Estos sensores se conocen como CNY70.



### Sensor Ultrasónico SRF04

Para que el Controlador NANO pueda medir la distancia a un objeto, se puede utilizar ecolocalización.



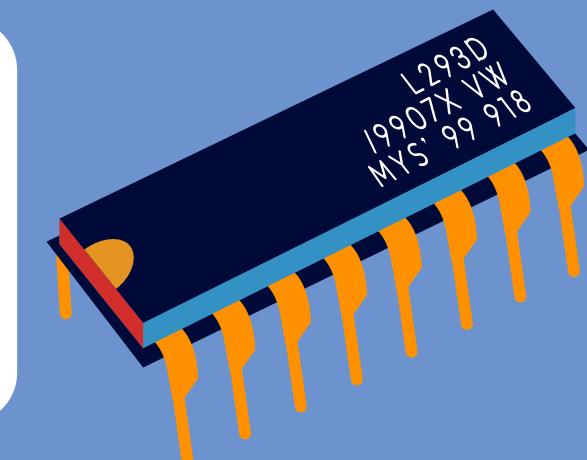
## Actuadores

### Puente H L293D

Para que el Controlador NANO pueda hacer que dispositivos de mayor potencia se puedan mover, como los motores de CD, se necesitan circuitos adicionales.

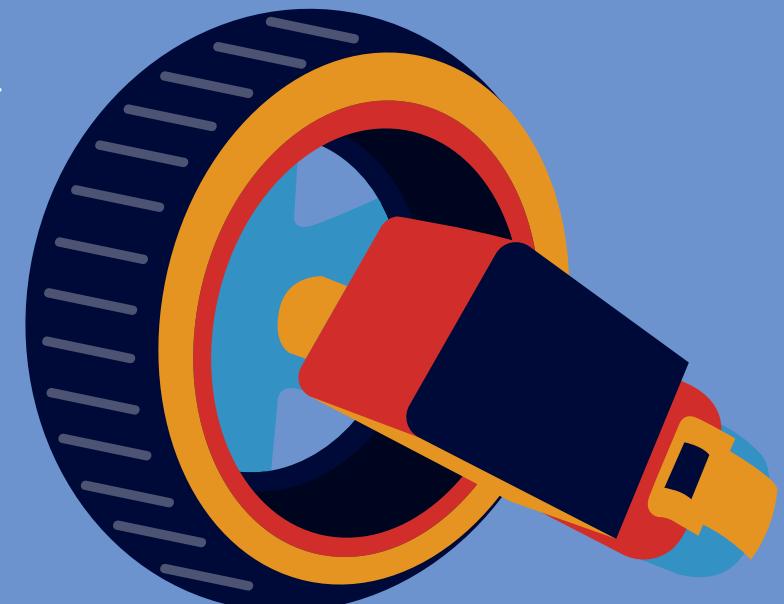
EN	1A	2A	FUNCIÓN
H	L	H	Giro a la derecha
H	H	L	Giro a la izquierda
H	L	L	Detención rápida
H	H	H	Detención rápida
L	X	X	Detención rápida

L=bajo H=alto X=no afecta



### Motor reductor con llanta

Para que el robot pueda desplazarse, se utilizan un par de motores con llantas. Estos motores se conectan a la etapa de potencia para que el Controlador Nano los pueda mover. Puede ser alimentado hasta 6V con 200 RPM y 3V con 150 RPM.



# Cap. 03

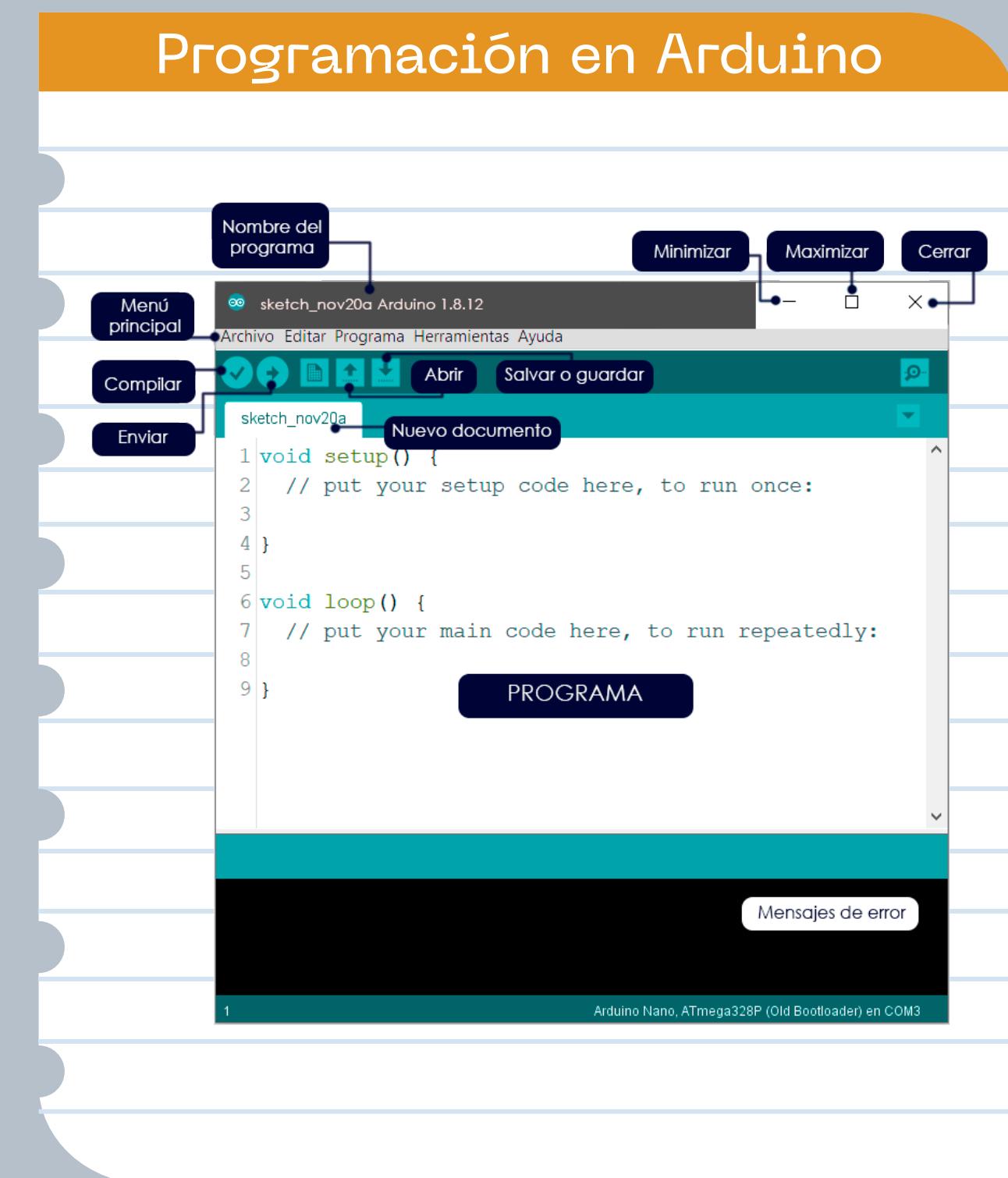
## Programación básica del Robot móvil

# Programación en Arduino

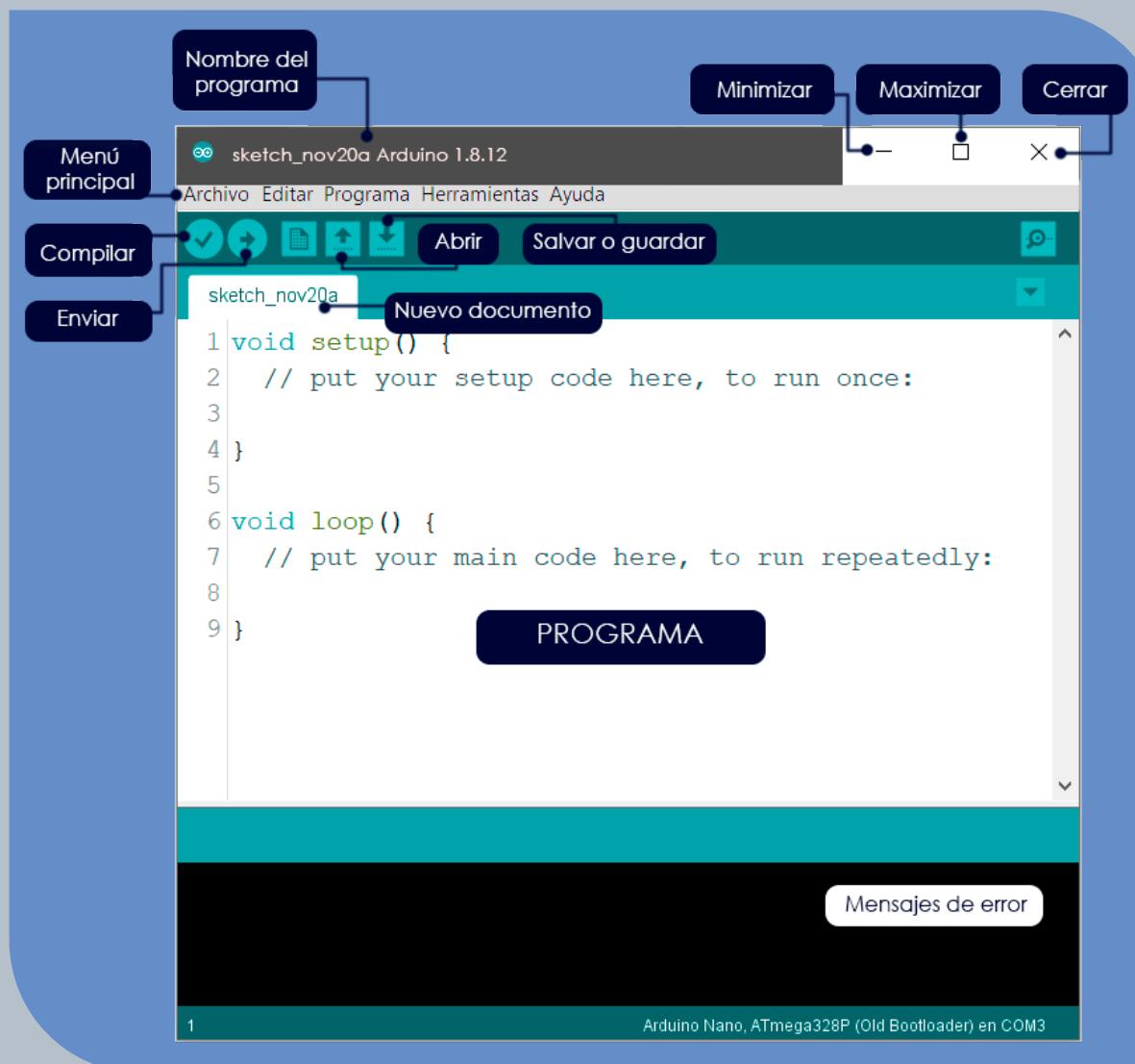
Las tareas que debe realizar el robot deben ser previamente descritas en forma tal que el usuario pueda explicar que es lo que se realizará. A este conjunto de indicaciones, expresadas en un lenguaje particular, se le conoce como programa.

Estos programas deben ser cargados al cerebro del robot, la tarjeta controladora Arduino, a fin de que él comprenda estas indicaciones y las lleve a cabo. En particular, para los dispositivos Arduino, se cuenta con un entorno de desarrollo integrado (Arduino IDE) en donde, el usuario escribe las instrucciones que desea que el Arduino realice.

Este mismo IDE tiene la capacidad de verificar que la sintaxis de todas las instrucciones escritas sea correcta, indicando cuando hay errores. Y en el caso de que no encuentre este tipo de errores, lo puede enviar al Arduino el cual realizará exactamente y de forma precisa lo indicado.



# Programación en Arduino



En la parte superior se encuentra, siempre en un color contrastante, la barra de título donde centrado se puede leer el nombre del programa y en la esquina superior derecha los clásicos botones de Minimizar, Maximizar y Cerrar la ventana. En un color más claro, en este caso gris, se encuentra la barra de menú principal.

Cada botón de esta barra da acceso a un menú desplegable en el que aparecen distintas opciones asociadas a dicho botón. Por ejemplo, en el menú de Archivo están las opciones de abrir, cerrar, nuevo, etc.

Un poco más abajo de la barra de menú principal, se encuentra la barra de botones. Desde ella se accede de una forma rápida, fácil e intuitiva a las opciones más utilizadas en cada programa.

- El primer botón que encontramos es el de compilar. Este botón, representado por un símbolo “palomita” de verificación, permite verificar si todas las instrucciones están escritas correctamente, así como convertir esta información a un código que el Arduino puede comprender y ejecutar. Si presionas este botón y no tienes ningún error en tu código, entonces tu programa está listo para ser ejecutado por tu Arduino.
- El siguiente botón a la derecha es una flecha que se conoce como enviar o subir. Es aquí donde, después de compilar correctamente el programa, se puede enviar al Arduino para que se realice todo lo que programaste.
- Los tres siguientes botones son muy simples si has tenido un acercamiento mínimo a una computadora, estos son Nuevo, que abre una nueva ventana de trabajo. Abrir, que permite seguir escribiendo un programa previamente realizado y Guardar, que almacena los cambios del programa actual.

Justo debajo de esta barra está el espacio destinado a tu programa. Es aquí donde se escriben las instrucciones que debe seguir el Arduino. Es importante notar que al ser un lenguaje de programación sensible a mayúsculas y minúsculas, debes tener mucho cuidado de escribir tal cual se te indique.

# Programación en Arduino

Cada programa que escribas debe tener de manera obligatoria tres secciones.

En el inicio se encuentra la sección conocida como “Sección de declaración de variables”, la cual nosotros denominamos la sección del QUE.

En esta sección se debe de poner lo QUE se va a ocupar (para los más conocedores, las variables, definiciones, librerías, etc.). Es aquí donde se escriben los nombres de las variables que se pueden utilizar en todo el programa.

La siguiente sección es delimitada por llaves de apertura y cierre, son las primeras instrucciones a ejecutar en el programa. Se denomina llamada al void setup().

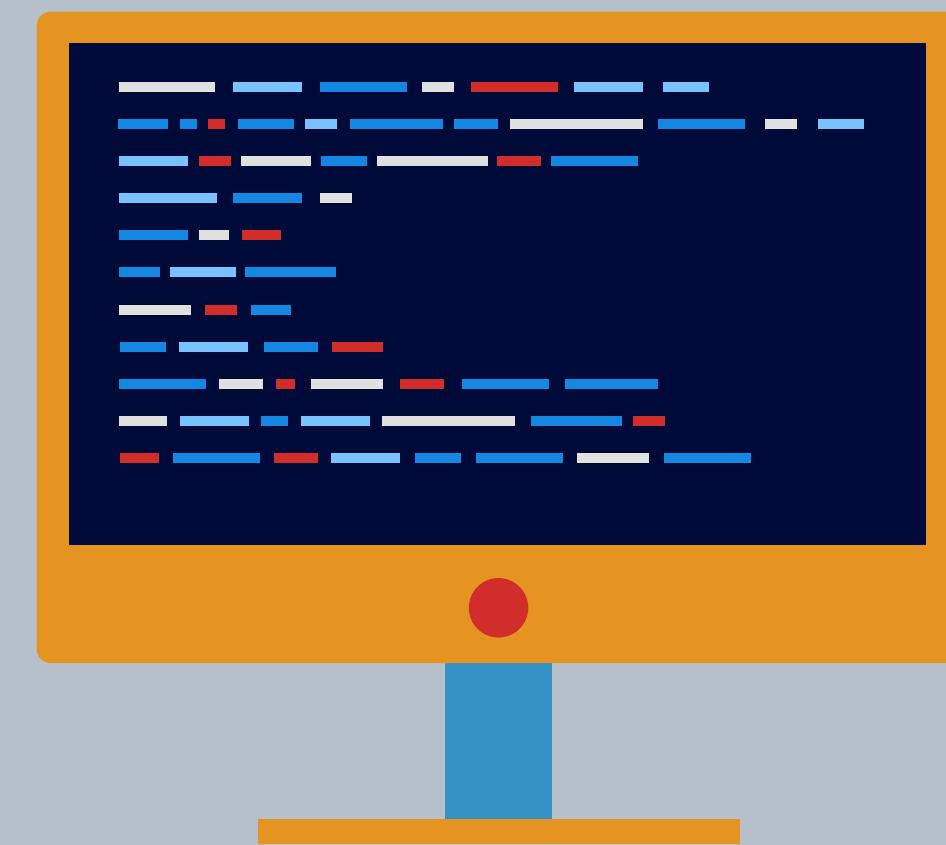
Esta sección la llamamos el COMO, por que aquí se indica COMO se van a inicializar las variables, si son de entrada o salida, analógicas o digitales, entre otras características que poco a poco irás entendiendo.

Una característica importante de esta sección es que todo lo que este aquí solo se ejecuta una vez, en el momento de encender o re-iniciar el Arduino.

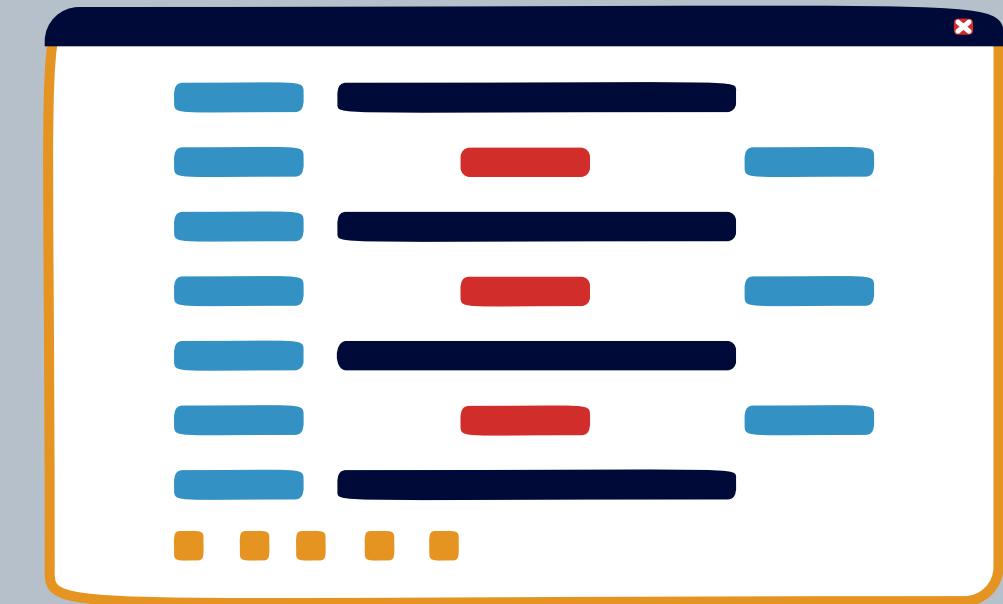
Un poco más abajo, después del COMO, se encuentra la sección llamada “void loop()”. También delimitada por llaves, incluye las instrucciones precisas de CUANDO se deben de activar salidas, leer entradas, CUANDO se utilizan condiciones para cambiar el flujo de un programa.

Como se puede observar, llamamos a esta sección CUANDO. Lo que se encuentra en esta sección se irá ejecutando instrucción por instrucción hasta llegar a la llave de cierre ( } ), en este momento el flujo del programa regresa a donde se abre la llave ( { ).

Esta es la razón del nombre en inglés loop, qué significa lazo, como un lazo amarrado por sus extremos, no importa por donde empieces siempre regresas al inicio, donde están los nudos.



# Programas 3.3 para el uso de los componentes



Blink

```
1 //////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: Blink.- Programa que enciende un LED por 1seg y lo apaga 1seg //
4 //////////////////////////////////////////////////////////////////
5
6 // ¿Qué? --> tipo de dato usamos:
7 int led = 14; //Se declara la variable led como un dato de tipo entero y se le
8 //asigna el pin número 14.
9
10 // ¿Cómo? --> usamos los datos (entrada o salida):
11 void setup(){
12   pinMode(led, OUTPUT); //Inicializa el pin digital 14 (led) como salida.
13 }
14
15 // ¿Cuándo?
16 void loop(){
17   digitalWrite(led, HIGH); //Enciende el LED (HIGH es 1 binario o 5[V])
18   delay(1000); //Espera por 1s
19   digitalWrite(led, LOW); //Apaga el LED (LOW es 0 binario o 0[V])
20   delay(1000); //Espera por 1s
21 }
```

LCD

```
1 //////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: LCD.- Programa que muestra en el LCD el mensaje "Hola mundo" //
4 //           y el valor de una variable //
5 //////////////////////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 // ¿Cómo? --> usamos los datos (entrada o salida):
12 void setup(){
13   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
14 }
15
16 // ¿Cuándo?
17 void loop(){
18   lcd.setCursor(0,0); //Coloca el cursor en la posición 0,0 del LCD
19   lcd.print("Hola Mundo"); //Imprime el mensaje "Hola mundo"
20   lcd.setCursor(0,1); //Coloca el cursor en la posición 0,1 del LCD
21   lcd.print(millis() / 1000); //Imprime la información de la variable millis
22 }
```

# Programas para el uso de los componentes

## Motores

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: Motores.- Programa que hace que el robot avance 2seg y se detenga 1 seg  //
4 ///////////////////////////////////////////////////
5
6 // ¿Qué? --> tipo de dato usamos:
7 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
8 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
9 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
10 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
11
12 void setup() {
13 // put your setup code here, to run once:
14 pinMode(13,OUTPUT);
15 pinMode(ad1,OUTPUT);
16 pinMode(at1,OUTPUT);
17 pinMode(ad2,OUTPUT);
18 pinMode(at2,OUTPUT);
19 pinMode(Adelante,INPUT);
20 pinMode(Atras,INPUT);
21 pinMode(Derecha,INPUT);
22 pinMode(Izquierda,INPUT);
23
24 Serial.begin(9600);
25
26 }

```

## Componente CNY70

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: CNY70.- Programa que lee el valor de los sensores CNY70 y los muestra en el LCD  //
4 //                                              //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 int ValorCNY_I; //Variable para guardar el valor del CNY Izquierdo
12 int ValorCNY_D; //Variable para guardar el valor del CNY Derecho
13 int ValorCNY_C; //Variable para guardar el valor del CNY Centro
14
15 // ¿Cómo? --> usamos los datos (entrada o salida):
16 void setup(){
17 lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
18 }
19
20 // ¿Cuándo?
21 void loop(){
22 ValorCNY_D = analogRead(A4); //Leemos el valor de los CNY's
23 ValorCNY_C = analogRead(A5);
24 ValorCNY_I = analogRead(A6);
25
26 lcd.setCursor(0,1); //Borramos datos del LCD
27 lcd.print("          ");
28 lcd.setCursor(0,0); //Mandamos a imprimir el valor del CNY Izquierdo
29 lcd.print("IZQ");
30 lcd.setCursor(0,1);
31 lcd.print(ValorCNY_I);
32 lcd.setCursor(6,0); //Mandamos a imprimir el valor del CNY Centro
33 lcd.print("CENT");
34 lcd.setCursor(6,1);
35 lcd.print(ValorCNY_C);
36 lcd.setCursor(13,0); //Mandamos a imprimir el valor del CNY derecho
37 lcd.print("DER");
38 lcd.setCursor(12,1);
39 lcd.print(ValorCNY_D);
40 delay(200); //Se requiere de un tiempo de retardo por estabilidad
41 }

```



# Programas para el uso de los componentes

## Ultrásónico

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)           //
3 // Programa: Ultrásónico.- Programa que lee el valor del sensor ultrásónico  //
4 //             y lo muestra en el LCD                   //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 long distancia; //Variable para guardar el valor de la distancia al objeto
12 long tiempo; //Variable para guardar el tiempo que tarda en rebotar el pulso
13
14 // ¿Cómo? --> usamos los datos (entrada o salida):
15 void setup(){
16   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
17   pinMode(13, OUTPUT); //Coloca al pin 13 como salida: genera pulso ultrásónico
18   pinMode(12, INPUT); //Coloca al pin 12 como entrada: detecta el tiempo de
19   //rebote del pulso ultrásónico
20
21 // ¿Cuándo?
22 void loop(){
23   digitalWrite(13,LOW); //Por cuestiones de estabilidad del sensor
24   delayMicroseconds(5);
25   digitalWrite(13, HIGH); //Envío del pulso ultrásónico
26   delayMicroseconds(10);
27   tiempo=pulseIn(12, HIGH); //Mide el tiempo transcurrido entre el envío del
28   //pulso ultrásónico y el rebote del mismo
29   distancia= int(0.017*tiempo); //Fórmula para calcular la distancia
30
31   lcd.setCursor(0,0); //Mandamos a imprimir la distancia calculada
32   lcd.print("Distancia=");
33   lcd.setCursor(10,0);
34   lcd.print(distancia);
35   lcd.print("cm ");
36   delay(500);
37 }

```

## Botones

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)           //
3 // Programa: Botones.- Programa que muestra en el LCD un mensaje con el nombre del //
4 //             botón presionado y su valor analógico      //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 int CDA_teclado = 0; //Variable que guarda la información del teclado
12
13 // ¿Cómo? --> usamos los datos (entrada o salida):
14 void setup(){
15   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
16   lcd.setCursor(0,0);
17   lcd.print("Presiona boton");
18 }
19
20 // ¿Cuándo?
21 void loop(){
22   CDA_teclado = analogRead(A7); //Lee el valor del pin 0 del puerto analógico. Cada botón
23   //tiene asignado alguno de los siguientes valores: 50,150,300,450,650,1000
24   if(CDA_teclado < 50){ //Se presiona el botón derecho 50
25     lcd.setCursor(0, 1);
26     lcd.print("Derecha      ");
27     lcd.setCursor(8, 1);
28     lcd.print(CDA_teclado );
29     delay(200);
30   }else if(CDA_teclado < 150){ //Se presiona el botón arriba 150
31     lcd.setCursor(0, 1);
32     lcd.print("Arriba       ");
33     lcd.setCursor(8, 1);
34     lcd.print(CDA_teclado );
35     delay(200);
36   }else if(CDA_teclado < 400){ //Se presiona el botón abajo 400
37     lcd.setCursor(0, 1);
38     lcd.print("Abajo        ");
39     lcd.setCursor(7, 1);
40     lcd.print(CDA_teclado );
41     delay(200);
42   }else if (CDA_teclado < 550){ //Se presiona el botón izquierdo 550
43     lcd.setCursor(0, 1);
44     lcd.print("Izquierda    ");
45     lcd.setCursor(10, 1);
46     lcd.print(CDA_teclado );
47     delay(200);
48   }else if (CDA_teclado < 850){ //Se presiona el botón selección 850
49     lcd.setCursor(0, 1);
50     lcd.print("Seleccion    ");
51     lcd.setCursor(12, 1);
52     lcd.print(CDA_teclado );
53     delay(200);
54   }else if(CDA_teclado > 1000){ //No se presiona ningún botón 1000
55     lcd.setCursor(0, 1);
56     lcd.print("No boton     ");
57     lcd.setCursor(10, 1);
58     lcd.print(CDA_teclado );
59     delay(200);
60   }
61 }

```



# Programas 3.4

## para tareas específicas

### Seguidor de líneas

```

1 ///////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: Seguidor.- Programa para realizar el seguimiento de un linea //
4 //             negra sobre un fondo blanco //
5 ///////////////////////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include<LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd (6,7,2,3,4,5); //Pines del LCD
10 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
11 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
12 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
13 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
14 int ValorCNY_I; //Variable para guardar el valor del CNY Izquierdo
15 int ValorCNY_D; //Variable para guardar el valor del CNY Derecho
16
17 // ¿Cómo? --> usamos los datos (entrada o salida):
18 void setup(){
19   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
20   pinMode(MD1, OUTPUT); //Coloca a los pines de los motores como salida
21   pinMode(MD2, OUTPUT);
22   pinMode(MI1, OUTPUT);
23   pinMode(MI2, OUTPUT);
24 }
25

```

```

26 // ¿Cuándo?
27 void loop(){
28   ValorCNY_I = analogRead(A6); //Leemos el valor de los CNY's
29   ValorCNY_D = analogRead(A4);
30   //Sólo se utilizan 2 sensores, el izquierdo y el derecho. El robot se coloca
31   //sobre la línea, cuidando que la línea esté entre los sensores
32
33   lcd.setCursor(0,1); //Borramos datos del LCD
34   lcd.print("          ");
35   lcd.setCursor(0,0); //Mandamos a imprimir el valor del CNY Izquierdo
36   lcd.print("IZQ");
37   lcd.setCursor(0,1);
38   lcd.print(ValorCNY_I);
39   lcd.setCursor(12,0); //Mandamos a imprimir el valor del CNY derecho
40   lcd.print("DER");
41   lcd.setCursor(12,1);
42   lcd.print(ValorCNY_D);
43
44   if (ValorCNY_D < 650){ //El sensor derecho va a un costado de la linea
45     digitalWrite(MD1, HIGH); //El motor derecho avanza durante 200 miliseg
46     digitalWrite(MD2, LOW);
47     delay(200);
48     digitalWrite(MD1, LOW); //Se detiene 100 miliseg
49     delay(100);
50   }else{ //El sensor derecho entra a la linea
51     digitalWrite(MD1, LOW); //El motor derecho se detiene durante 200 miliseg
52     digitalWrite(MD2, LOW);
53     delay(200);
54   }
55   if (ValorCNY_I < 650){ //El sensor izquierdo va a un costado de la linea
56     digitalWrite(MI1, HIGH); //El motor izquierdo avanza durante 200 miliseg
57     digitalWrite(MI2, LOW);
58     delay(200);
59     digitalWrite(MI1, LOW); //Se detiene 100 miliseg
60     delay (100);
61   }else{ //El sensor izquierdo entra a la linea
62     digitalWrite(MI1, LOW); //El motor izquierdo se detiene durante 200 miliseg
63     digitalWrite(MI2, LOW);
64     delay(200);
65   }
66 }

```



# Programas para tareas específicas

## Evasor de obstáculos

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: Evasor.- Programa para hacer que el robot avance de forma      //
4 //               autónoma evadiendo obstáculos al frente de él      //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include<LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd (6,7,2,3,4,5); //Pines del LCD
10 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
11 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
12 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
13 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
14 long distancia; //Variable para guardar el valor de la distancia al objeto
15 long tiempo; //Variable para guardar el tiempo que tarda en rebotar el pulso
16
17 // ¿Cómo? --> usamos los datos (entrada o salida):
18 void setup(){
19   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
20   pinMode(MD1, OUTPUT); //Coloca a los pines de los motores como salida
21   pinMode(MD2, OUTPUT);
22   pinMode(MI1, OUTPUT);

```

```

23   pinMode(MI2, OUTPUT);
24   pinMode(13, OUTPUT); //Coloca al pin 13 como salida: genera pulso ultrasónico
25   pinMode(12, INPUT); //Coloca al pin 12 como entrada: detecta el tiempo de
26 }                                //rebote del pulso ultrasónico
27
28 // ¿Cuándo?
29 void loop(){
30   digitalWrite(13, LOW); //Por cuestiones de estabilidad del sensor
31   delayMicroseconds(5);
32   digitalWrite(13, HIGH); //Envío del pulso ultrasónico
33   delayMicroseconds(10);
34   tiempo=pulseIn(12, HIGH); //Mide el tiempo transcurrido entre el envío del
35                                //pulso ultrasónico y el rebote del mismo
36   distancia= int(0.017*tiempo); //Fórmula para calcular la distancia
37
38   lcd.setCursor(0,0); //Mandamos a imprimir la distancia calculada
39   lcd.print("Distancia=");
40   lcd.setCursor(10,0);
41   lcd.print(distancia);
42   lcd.print("cm ");
43
44 if(distancia < 35){ //La distancia del robot al obstáculo es < 35cm
45   digitalWrite(MD1, LOW); //El robot se detiene por 500 miliseg
46   digitalWrite(MD2, LOW);
47   digitalWrite(MI1, LOW);
48   digitalWrite(MI2, LOW);
49   delay(500);
50   digitalWrite(MD1, HIGH); //El robot gira a la derecha por 1 seg
51   digitalWrite(MD2, LOW);
52   digitalWrite(MI1, LOW);
53   digitalWrite(MI2, LOW);
54   delay(1000);
55   digitalWrite(MD1, LOW); //El robot se detiene por 500 miliseg
56   digitalWrite(MD2, LOW);
57   digitalWrite(MI1, LOW);
58   digitalWrite(MI2, LOW);
59   delay(500);
60 }else{ //La distancia del robot al obstáculo es > 35cm
61   digitalWrite(MD1, HIGH); //El robot avanza por 100 miliseg
62   digitalWrite(MD2, LOW);
63   digitalWrite(MI1, HIGH);
64   digitalWrite(MI2, LOW);
65   delay(100);
66 }
67 }

```





**CONACYT**  
Consejo Nacional de Ciencia y Tecnología

E S P A Ñ O L - M I X E

Una introducción al

# FASCINANTE MUNDO DE LA ROBÓTICA



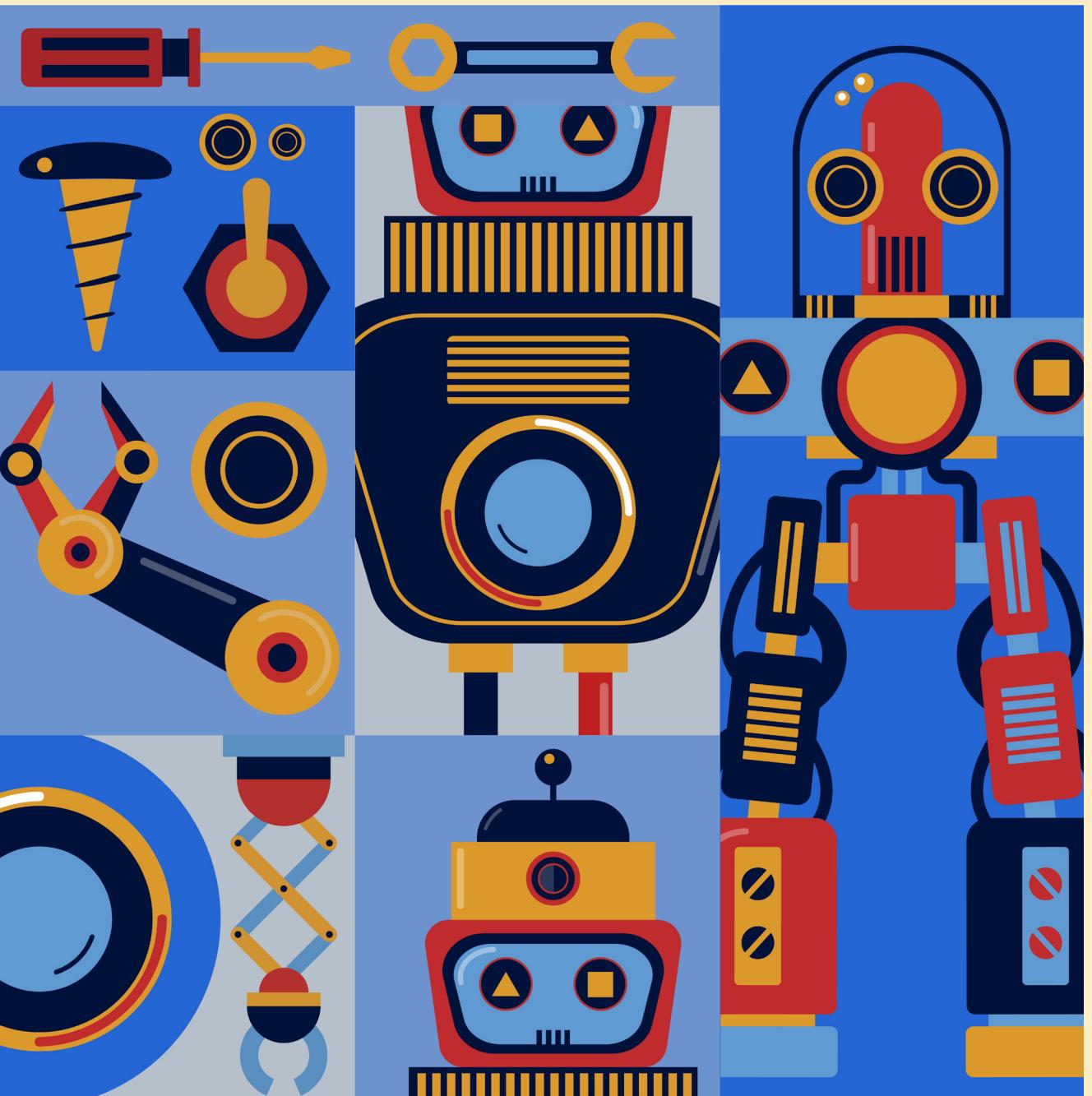
Responsable técnico:  
Jesús Santiguillo Salinas

Colaboradores:  
Hiram N. García Lozano  
Rafael F. González Zarate  
José L. Nájera Sánchez  
Luis A. Hernández Zuccolotto

Diseño e ilustraciones:  
Melisa del C. Muñoz Vázquez

Traducción:  
Vanessa N. Gómez Jiménez

Asistentes de traducción:  
Leticia Gómez Jiménez  
Alfonso Díaz Vargas



ESPAÑOL - MIXE  
Ja tsuj tsyo'ntajken JA  
**NÄXWIINY**  
**ROBÓTICA**

Ejxy'ejtpē mitē nyajää wëep  
ti tunép jätjép:  
Jesús Santiaguillo Salinas

Wunmääny Ajanyák  
tanépétääjép:  
Melisa del C. Muñoz Vázquez

Jakyukénwunmääny:  
Vanessa N. Gómez Jiménez





ESPAÑOL - MIXE  
**Ja tsuj tsyo'ntajken JA  
NÄXWIINY  
ROBÓTICA**



Ejxy'ejtpë mitë nyajää'wëëp  
ti tunëp jäjtëp:  
Jesús Santiguillo Salinas

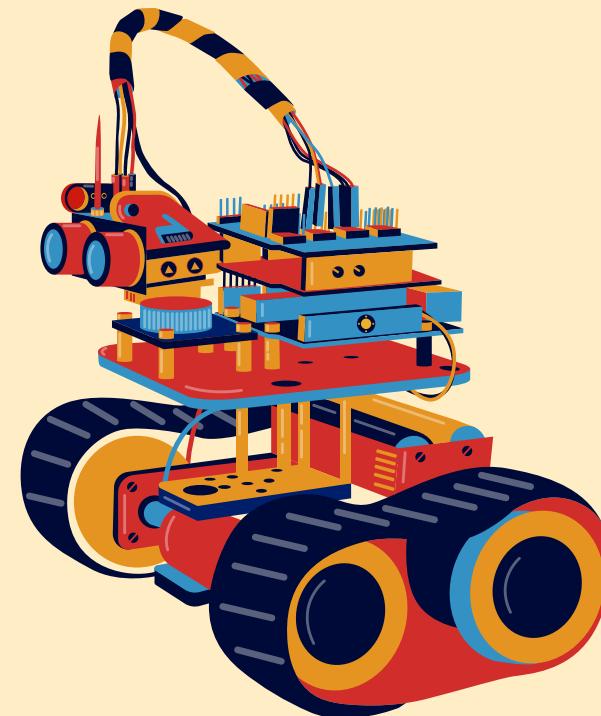
Pukkë'pë-puxäjëpë:  
Hiram N. García Lozano  
Rafael F. González Zarate  
José L. Nájera Sánchez  
Luis A. Hernández Zuccolotto

Wunmää'ny Ajanyäx  
tanëpëktääjkëpë:  
Melisa del C. Muñoz Vázquez

Jakyukënwunmää'ny:  
Vanessa N. Gómez Jiménez

Pukkë'pë-puxäjëpë:  
Leticia Gómez Jiménez  
Alfonso Díaz Vargas

ESPAÑOL - MIXE  
**Ja tsuj tsyo'ntajken JA  
NÄXWIINY  
ROBÓTICA**



# Índice



## Tsyo' ntajken

## 01 CAPÍTULO

### Ja pujx tsyo' ntajken

#### 1.1 Tsä ojts tsyoo'ntä'äky

1.1.1 Ja pujxtsyë jëts ja cine

1.1.2 Ìxpëjkën Tsä ojts tsyoo'ntä'äky ja pujxtsyë

#### 1.2 Ja kawënë'n ja pujxtë mitë robot tyejtëp

1.2.1 Robots manipuladores

1.2.2 Ja pujxtsyë mitë oy'yatë'ëtstëp kawënääk ampy

1.2.2.1 Ja pujxtsyë móviles näxwënyë'pyëtë

1.2.2.2 Ja pujxtsyë mitë nøyumtëp

1.2.2.3 Ja pujxtsyë mitë yiktuntëp ja control

#### 1.3 Mä tsyuntë Yä'ät pujxtsyë

1.3.1 Robots de telemundo

1.3.2 Ja pujxtsyë mitë pyutëjkëtëp yikxon

1.3.3 Ja pujxtsyë mitë pyutëjkëtëp jam industrias

1.3.4 Ja tsëyëpëtë ja' mitë ja jää'y napyëktääjkëtëp

1.3.5 Pujxtsyë Boston Dynamics

70

72

74

75

76

80

81

86

88

89

89

90

90

92

94

96

98

## 02 CAPÍTULO

### Ti yik tuntëp ja pujxtsyë

#### 2.1 Myëjawyeen pujx móvil

#### 2.2 Ti myëëtëp ja'n myëjawyeen pujx móvil

##### 2.2.1 Controlador

2.2.1.1 *Arduino - Nano*

2.2.2 Dispositivos Mä tyiknäxtë jëts tyikpëtsëmtë

2.2.3 Sensores

2.2.3.1 *Sensor óptico*

2.2.3.2 *Sensor ultrasónico*

2.2.4 Actuadores

2.2.4.1 *Puente H 1293D*

2.2.4.2 *Motor reductor con llanta*

100

102

104

104

104

105

106

106

106

106

107

107

## 03 CAPÍTULO

### Programación básica del robot móvil

#### 3.1 Programación en Arduino

#### 3.2 Programas para el uso de los componentes

3.2.1 Jan myëjawyeen programa Blink

3.2.2 Programa para el uso del LCD

3.2.3 Programa para el uso de los motores

3.2.6 Programa para el uso de los sensores ópticos

3.2.4 Programa para el uso del sensor ultrasónico

3.2.5 Programa para el uso de los botones

#### 3.3 Programas para tareas específicas

3.3.1 Robot seguidor de líneas

3.3.2 Robot autónomo evasor de obstáculos

108

110

116

116

117

118

119

120

121

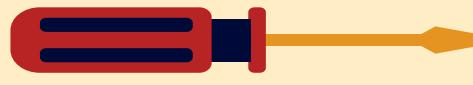
122

122

124



# Ja pujx tsysó'ntajken



Yä'at nëky tsysó'ntä'äky tsäm tu'uk ja propuesta ja proyecto mä yi ejxpëjktääjkën Universidad del Papaloapan ojts yik mo'otë ja mëj wënmää'ny 2021 ojts yik tuny ja propuesta mitë xyëew'ajtypy proyectos para el formento y fortalecimiento de las vocaciones científicas del consejo Nacional de Ciencia y Tecnología (CONACyT).

Yä'ät wënmää'ny nëky tsyséjkypy ku ja ayoopë'jää'ytyë yä näxwiinyët yik putékëtëp, pën ma'äse ojts yik pawënmaytsye ja mutsk kiïxytyë, mijxtsyë, wäjtyëjktë jëts kiixye'të, ja wëjen científico jëts tecnológico de vanguardia tpääätët, jëts yik putékëtëp, mëët ja tunpajt digital, jëts tyik'ëyëtëp mä ja internet tyejtëp, jëts ja wejën kajën tpääätëp, mä yi unä'äjktë yik pukkäjpxëtëp jëts yik tuk'ejtëp ja mëj jëts tükatsy wënmää'ny.

Jatë'n yik pawënmay kü yik'putékëtëp ja käjp ayuujk jëts ja tükatsy jää'y ja jakam käjp, jëts ja ayopëjää'ytyë, jawyeen ja unä'äjktë mitë myëëtëtëp makmäjkts o ee'px maktäxk jyumëjt, mitë yikxon yikpawënmaätyëp ja jää'y mitë tükatsy ja y'ääw y'ayujk kyäjpxtëp mitë ka' amixän o ka' ja myeeny jyënkääp myëëtëtëp jëts ja ejxpëjkën tpääätëp.

Yä'ät nëky myëëtëp tükëek ja wënmää'ny: jäwyeen ejxpëjkën, ti ja robótica tyijpy, ti yik tuntëp, jëts ja mitë yik tejp programación básica de Robot.

Mitë myëjawyeen yi kajpxy'ätwämp, tsu ja pujx ojts tsysoo'ntä'äky, tsä ojts tyikatsy jumëjt ämp, jëts tsutso ejxyäm ja tsuj'äjten yi robótica tmëëtë. Yik nänky'ejxää'nítip tsä tyékätstëp ja pujx mitë atëtstëp jëts yë'ëtyëp, mitë yiktejtëp sistema de locomoción y dirección para robots móviles.

Jä'ätso ampyë, mä ja componentes de un robot yik nymätsiäkanye ja módulo robótico móvil diferencial, pën ojts tyik'ëyëtë ja Universidad del Papaloapan, Yiknëmatyä'äkwämp tu'uk jaty ja componentes mitë myëëtëtëp, ja sensores, actuadores jëts ja placas electrónicas de control jëts tsutso tejxtë.

Yä yjipkixy, mä ja yik mo'tëp wyinajty ja base de programación del robot móvil diferencial en Arduino jëts wyinajty yik tuk'ejxtëp ja programas tsä tyiktu'ntëp tu'uk jats ja componentes electrónicos mitë yiktuntëp ja robot.

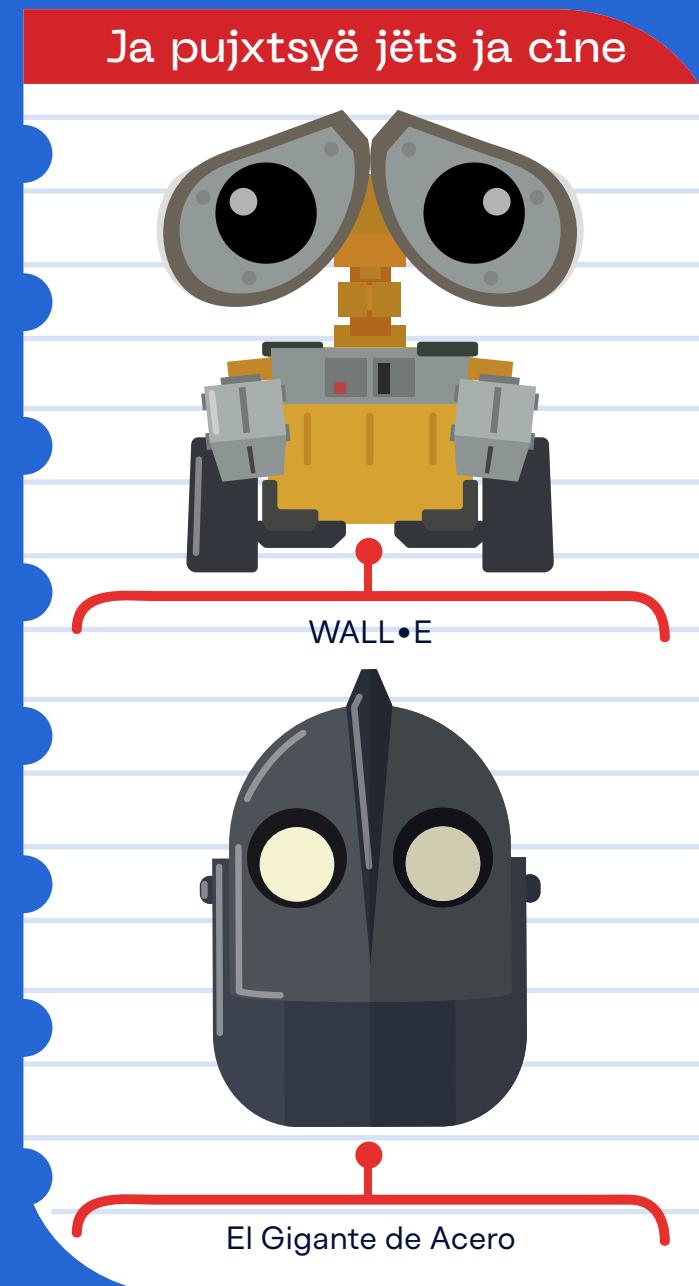
Jëts jatë'n nïky jïpkixëmpy ya'ät sección de programación básica de robot, yik na'änkyexänëyëtëp ja programas mä yik nymätsiäky tsutsë ja robot tsysuntë jëts ja tsep tunk tyk'ëyëtëp tsäm ja: seguimiento de línea y movimiento autónomo con evasión de objetos.

# Cap. 01

## Ja, pujx tsyo' ntajken

Ja pujx tsysoo'ntä'äky

# Tsä ojts tsyoo'ntä'äky



Ja pujxtsyé jëts ja cine

Ka' ja'yë të xyikjotkujk'äjtën mëet ja mëj nëmatyä'äky mitë tyejtëp Ciencia ficción nayëtë'n ojts tyiku'ntë säm ja pudëjk äjtën ja ejxpëjktääjkëñ jëts ja Centro de Investigación, ja pujxtsyé ojts tsyékëtë mä ja may jëts tékatsy ampye ja industrias.

Ja literatura jëts ja cine, tyuk'eyetëp ja jää'ytyë, ku ja robot tmëetë ja ëy wunmää'ny, jëts ja wenkëm nayëtë'n.



Yä'ät pujxtsyé myëtëtëp ja sistemas mecánicos jëts ja electromecánicos, nayïtëñ tuntë ja tunk mitë yik ta anamptëp.

Jatë'ën kyaxë'ëkte sa'äm tu'ukë jääy, ka'apx ja tunk yik tukëxtë, jëts ma'äsë kajaa tsyuntë...

¿Ti ntanajää'wa'amp wyënhäjty?

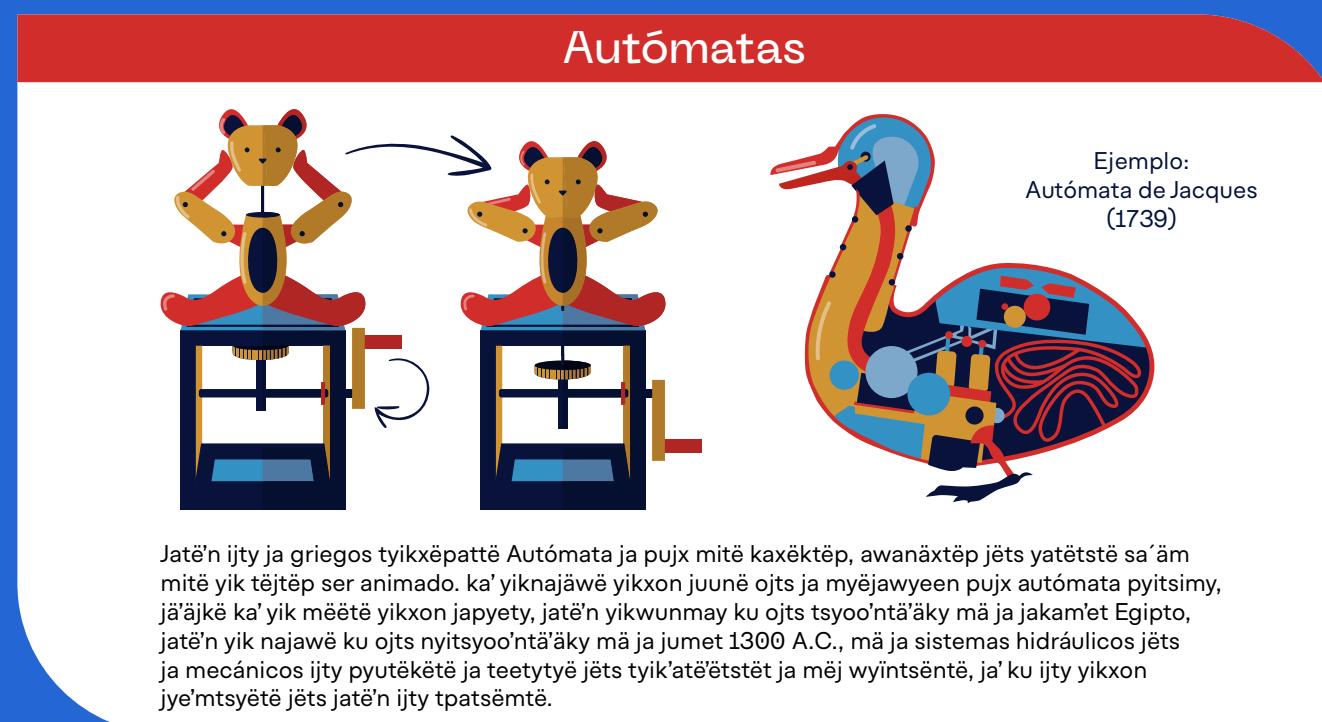
Ja pujx tsyo'ntajken

# Tsä ojts tsyoo'ntä'äky

# ROBOT

Jam may'ampy tsä yik nëmatyä'äky ja "robótica". Jëts jatë'n yik wënmay ku jaak ka'pxy ja Dr. Fernando Reyes Cortes, tnëmatyä'äky mä ja nëky txëewë "Robótica" ja' jaak ka'pxy jëts ëy tsuj tnëmatyä'äky.

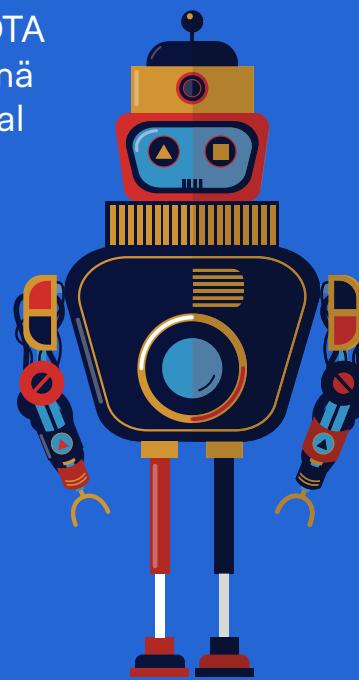
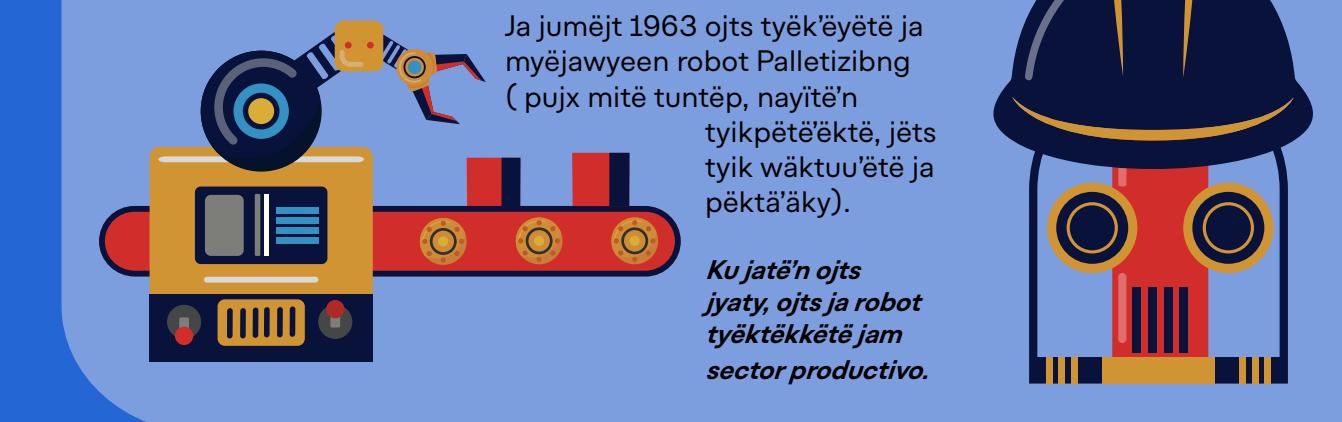
*"Ja robótica yik'ejxép tsäm tu'uk ja ejxpéjkén científico mitë yiktuntëp ja kajaa'ejxpéjkén jëts mitë yik tejp desarrollo de sistemas mecánicos, ja' yik xépajtëp robots, yë' yik'ëe'yëtëp ja tyikatsy tunk tsäm ja industrielas, comerciales, jëts jatu'ukpëtë".*



Yä'ät xyëew mitë yiktejp robot, memp mä yi checo ROBOTA jëts pën myijawyeen ojts tyiktuny ja jaapyë Karel Capek mä ja jumëjt 1917, ja nëky ojts tyik xépattë Rossum's Universal Robots, jatë'n ijty tiyy ja pujx tyuny mitë ja'yë kaxéktëp. yä'ät xyëew robot ojts tyiktékätstë mä ja äaw ayuujk inglés.

Ja jää'ytyï mitë tyunk'äjtëp ja jaaky ja jo'tsk, ojts tyik tunkojmëtë ja nïmatyä'äky ja robotëjya' tsäm ja Isaac Asimov (pën ojts txëtijy robótica) mä tnïmatyä'äky ja kajaa tsyënää'yën, jajp yik pääty ja neky mitë xyëew "yo robot", mä myiny tékëek ja kutujkén de la robótica, mitë yämpäät njaak yiktuu'nyëmp.

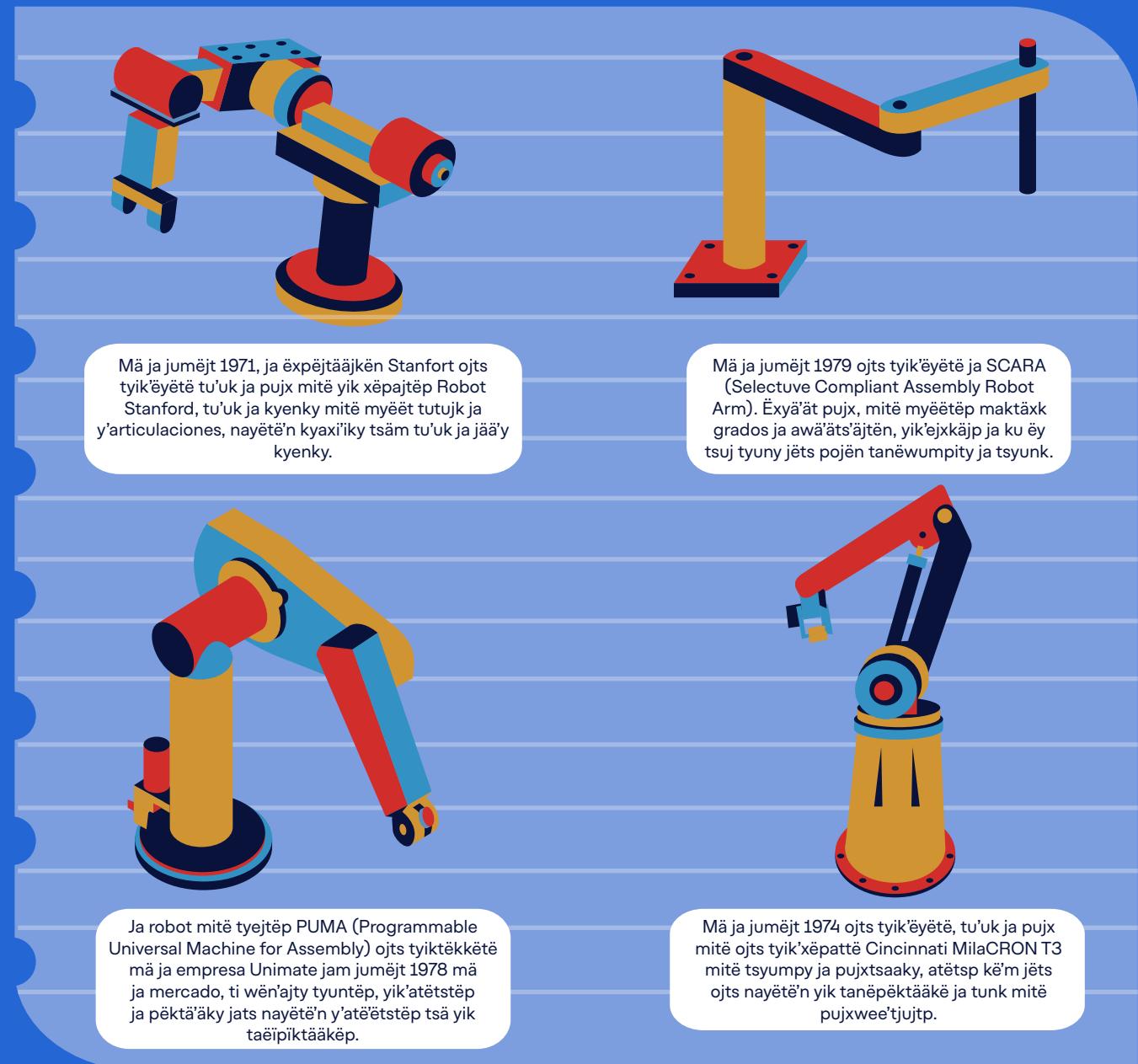
Ja mitë yik tejtëp sector industrial ojts t'ejxtë ja mëj putékëtën mitë ja pujx kyëjäktëp, ku tyuny mä ja tyunk tyik'eyëtë. Mä ja jumëjt 1961, ja robot Unimate ojts tyiktékëtë mä ja empresa txëewi General Motors.



Ja pujx tsyo'ntajken

# Tsä ojts tsyoo'ntä'äky

Pats ja mëj ïxpëjtääjkën mä nëtukki'yëtë ja näxwinyët t'ejxtëp ja robótica tsäm tu'uk ja nämñem ïxpëjtääjk, mitë kajaa wunmääny jëts tank myëetëp.



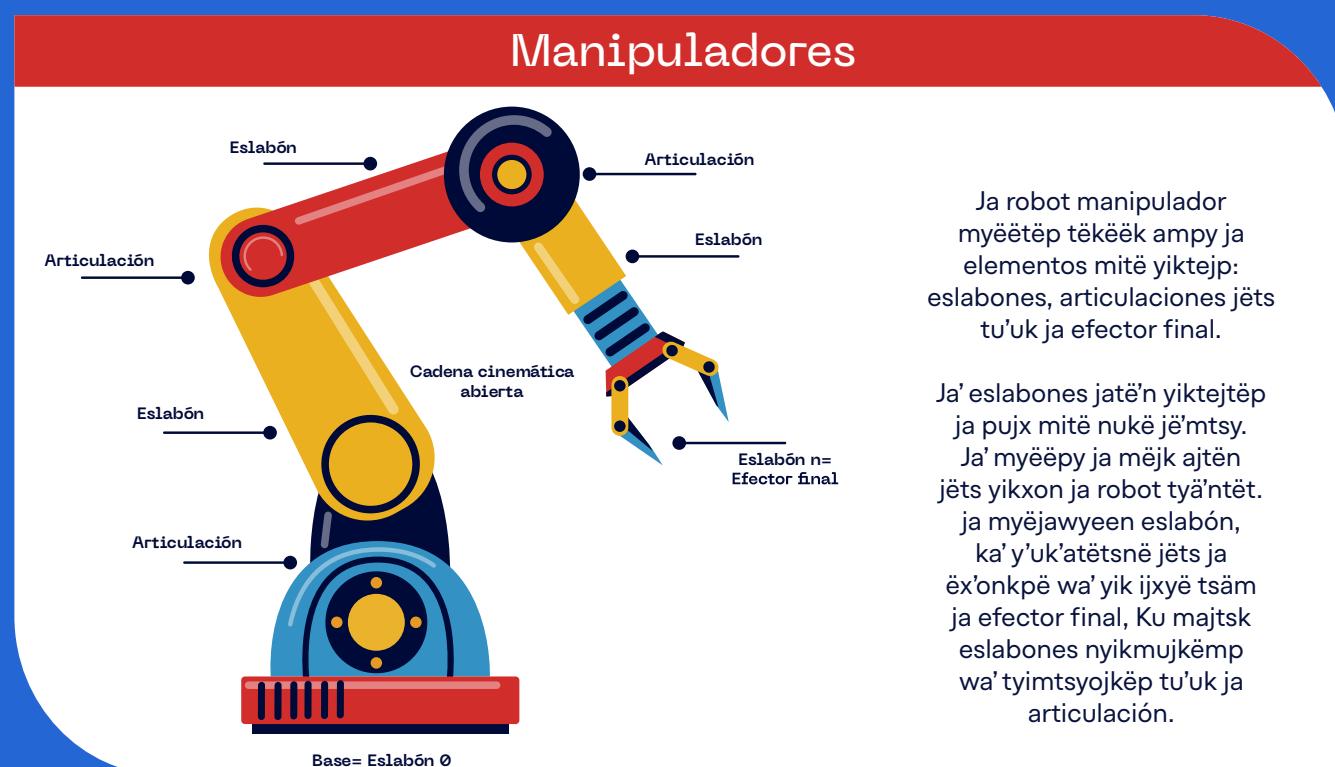
Ja nëky amëxän mitë yik xëpajtëp El país, mä myiny economía, wëxujk ämp ja myumajtskpë po' (7 de febrero) 2016 ja robot ojts txëtejtë, cuarta revolución industrial, jëts tamë'ejxëtë tsä ijty tyuntë ja pujx mitë pujxky yiktuntëp mä ja época preindustrial jëts tsä ëxyäm tyuntë jam industria.

Ëxyäm yi näxwiinyët yë' tsojkeyep ets yë' tsyu' untëp ets ja käjp yakpë. ja jumëjt 2008 wa' ojts tyuntë 1,000,000 ja robot yä näxwiiny jëts ojts yik ijxy ku ja jumëjt 2018 jaa wyinäjty nyityunë 1.3 millones robots industrielos.



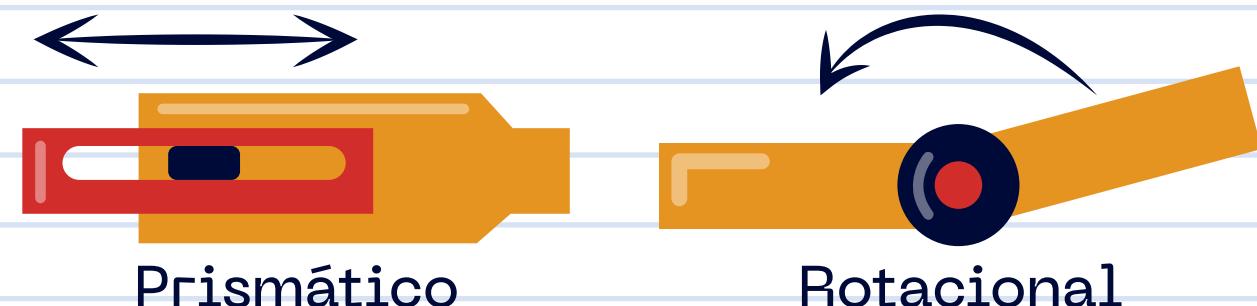
# Tsä y'a'äätspaky 1.2

Ja may ampy tsutsë yik tuntë jëts yik ejxkäpt ja tükatsy jats ja robots mitë ya näxiiny nmëëta'amp. yä yik kajpx'ätwä'äny jëts yik tuk'ejxwämp tsä yik tejtë tu'uk jaty.



## Manipuladores

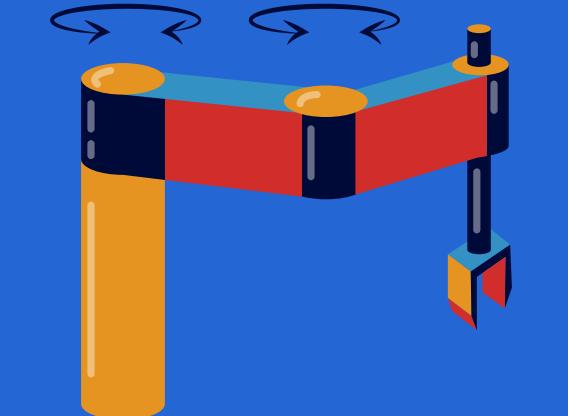
Yä'ät articulaciones atëtstëp tükëek ampy:



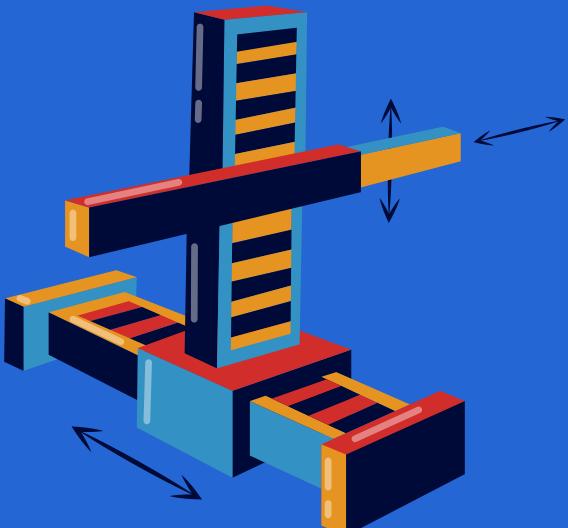
## Esférico

# Tsä y'a'äätspaky

## Manipuladores



Ku majtsk ja eslabones nyamyukyëtë mä ja articulacion rotacional, tsutsë wyinajty y'atëtstëp, jatë'n tsäm tu'uk ja akäj mitë yik tumpy ja bisagras. Ku tu'uk ja eslabon nukë tyanë, ey ja jatu'ukpë y'awëtetp jatë'n tsäm ja horario o ja anti-horario.



Ja eslabon prismático tyumpy ja movimiento lineal mä tu'uk jaty ja eslabones tyuntë. tym jatë'n y'ati'itsy tsäm tu'uk ja ventana corrediza. jatë'n ja eslabon prismático y'ati'itsy wénkëm jëts jëxki'ipy.

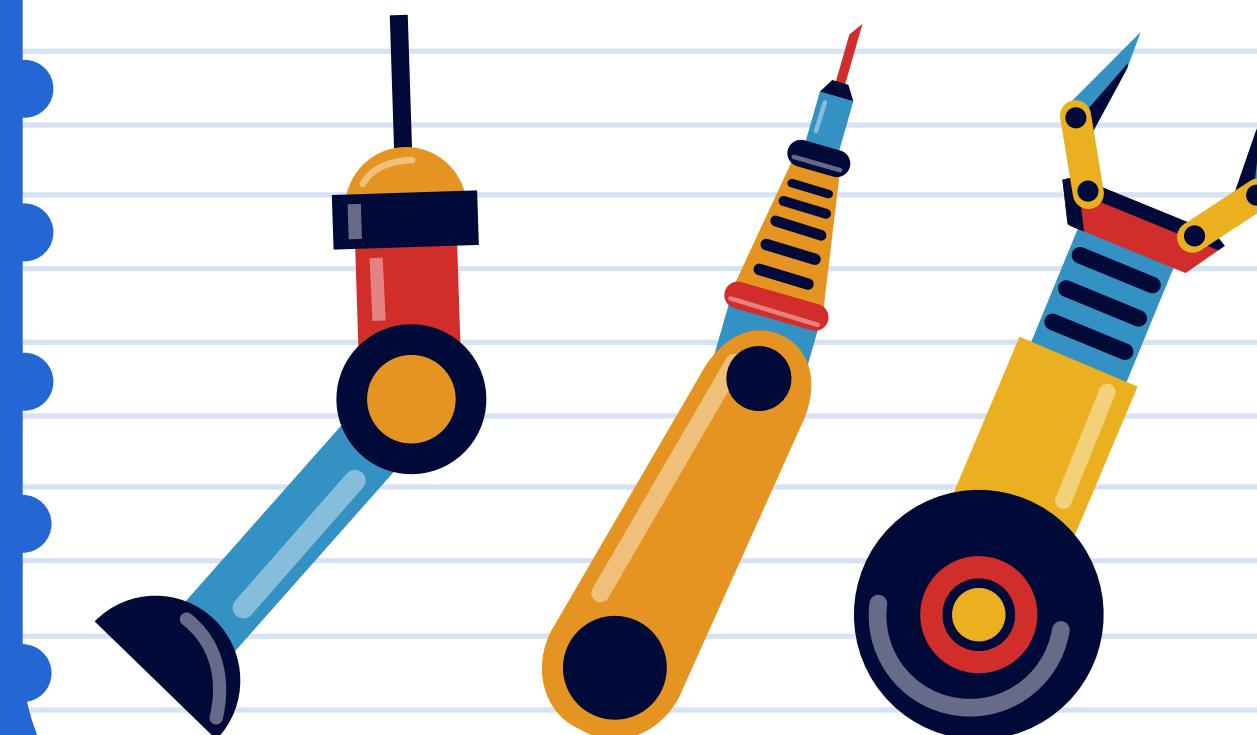
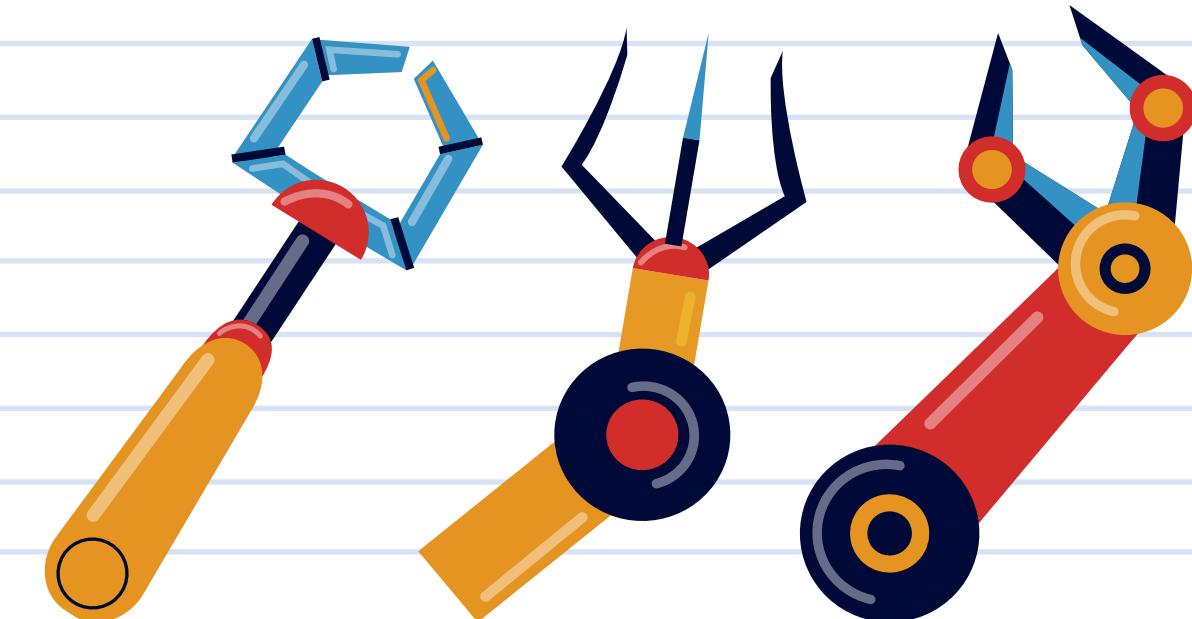


Ja articulación esférica tyumpy tékéek ja rotaciones, jatë'n kyax'iiky tsäm atëm yi'nkupäjk uk yin kë' nyik'atëeëjtsëmp, (akää'ny jëts anäjy ampy) (nayte'n tsäm yi'nkupäjk nyik'atijjtsyëmp ku nii'myëmp ku ka'). Jëts nayitë'n y'ati'itsy wénkëm jets jëxkëm tsäm jeexyëp nii'myëmp ku jëe.

## Manipuladores

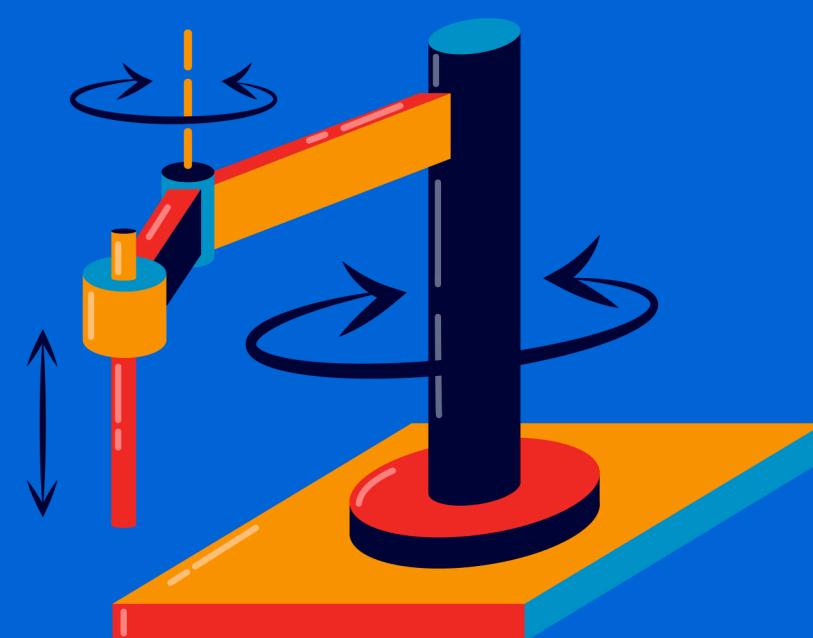
### Efecto Final

Yä'ät tumpajt mitë putëjkëyëp ja robot jëts tyik'ëyëp ja tyunk.



# Tsä y'a'äätspaky

## Tipos de robots manipuladores



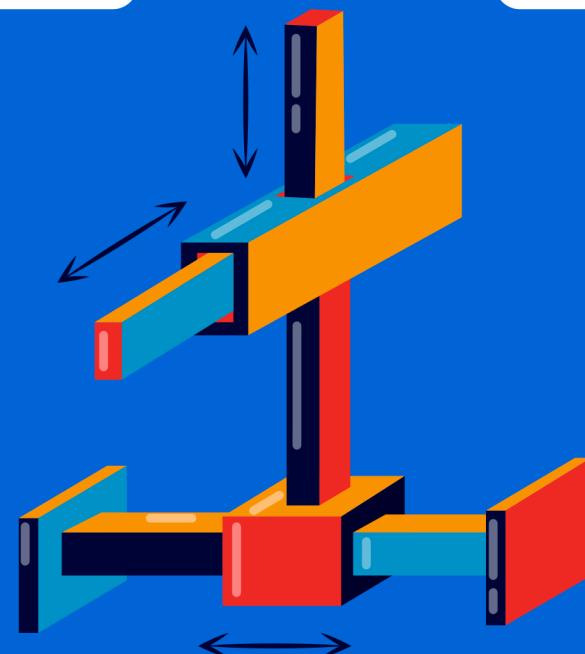
Robot SCARA



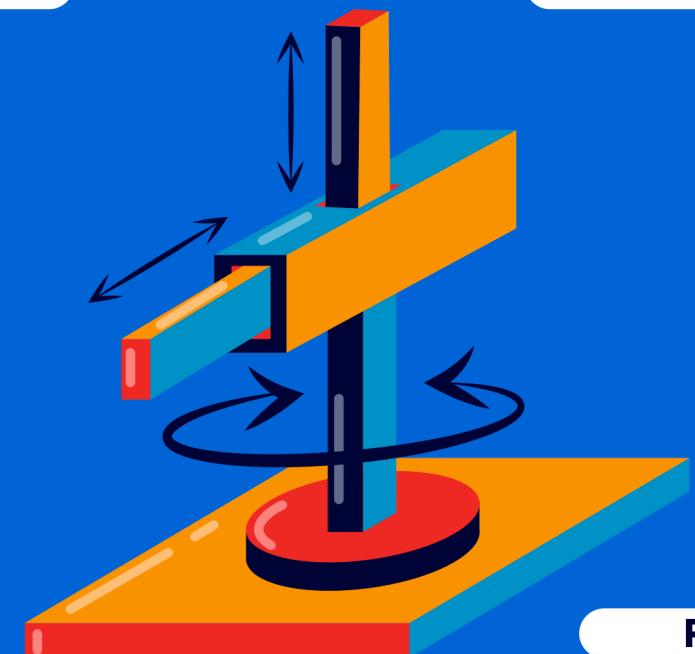
Robot Esférico



Robot Antropomórfico



Robot Cartesiano



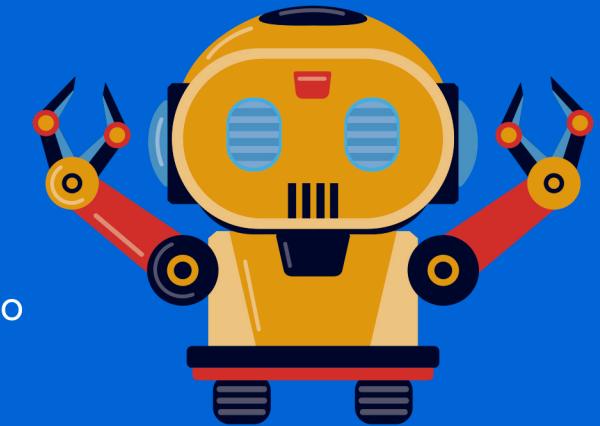
Robot Cilíndrico

# Tsä y'a'äätspaky

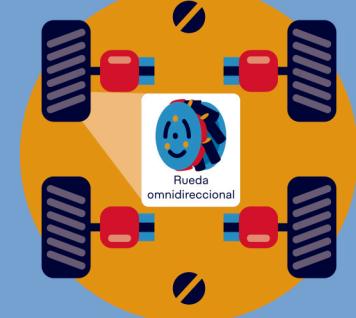
## Robot móvil

Tsäm tē yik nümatyä'äky, ja may'ampy ja robot atëtspë. Naytë'n ixäm yik kajpxwä'äny ja' mitë yik'tejtëp robots atëtspë näjxkijxy mëet ja tyiyuupyë. Yä'ät pujx tyim étye jëts y'atëtstëp mä ja jëynyäjax, jëts pënë ka' ja atsejpën tee, éy wyinajty tpääätë ja kajaa pujtk.

Ja robots atëtspë mëet ja tyiyuupyë, yikpëktäktëp tsäm tu'uk jaty ja llantas t'ejxtë jëts tmëëtëtë, jatë'n tsäm tyuntë jëts tsutsoo y'atëetstë. yä'ät yik ijxyëp tsäm ja configuración ja sistema de locomoción.



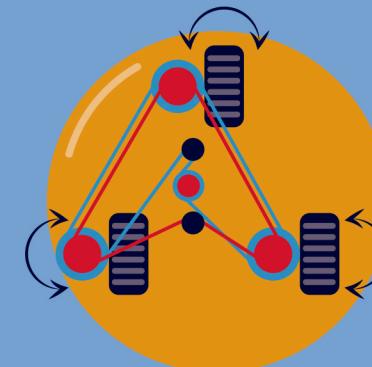
### Tipos de Robot Móvil Terrestre



Robot con ruedas omnidireccionales

Ja ruedas omnidireccionales myëëtëtëp ja tyikatsy jëts tsep ja diseño, oy'atëetstëp kawënäak ampy. Ja J.Grabowiecki ojts yikmëëy ja awäät'säjtën jëts ja têyuupyë tyiktékëtëp jam Estados Unidos, ja jumejt 1919, ja diseño té tyékäjtsnë jumëjt ampy jëts exämpäät.

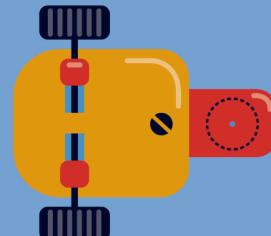
Yä'ät robot yik'ijxyëp ku tukjëep ampy ja'yë ja tyuu, jëts ka' tyik tekatsy.



Configuración Síncrona

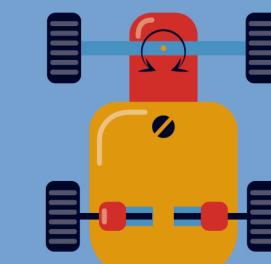
Yä'ät ampy (tékëek ja ruedas) tyiktuntë jëts tmo'të tu'uk ja tyuu' jëts tracción ja robot. Ku yétye, tékëek ja têyuupyë y'awëtetë, jëts tum jatë'n y'atëetstë, ku ja tyuu' nyiktékajtsënt, ja têyuupyë tumnayatén'ampy y'atëtstëp tuknäx. Yä'ät ampy tu'uk jats ja têyuupyë ja tyunk tyik'eyëtë mëet majtsk ja bandas.

### Tipos de Robot Móvil Terrestre



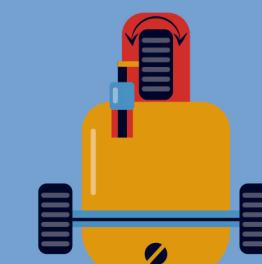
Tracción Diferencial

Majtsk ja têyuupyë tyuntë mëet tu'uk jaty ja motores (tracción independiente), jam yik pëktääky mä tukween ja robot yikti eje comün. ja myëtékëek têyuupyë awëtejtp oy'ojsääj (ja'yikxemëëpy têyuupyë local) yikpëktäkp jyëxki'py'ampy ja robot, jëts ka' kyëtäät uk wenk y'atëetsp.



Configuración Ackerman

Ja configuración Ackerman myëëtëp maktäxk ja têyuupyë, majtsk wyënk'i'py, ja' yäjktëp ja tuu', jëts naytë'n majtsk jyëxki'py mitë yäjktëp ja tracción.



Triciclo clásico

Ja configuración triciclo clásico myëëtëp tu'uk ja rueda wënkëm mitë yiktumpy jëts myë'ëyëp ja tyuu'jëts ja tracción.

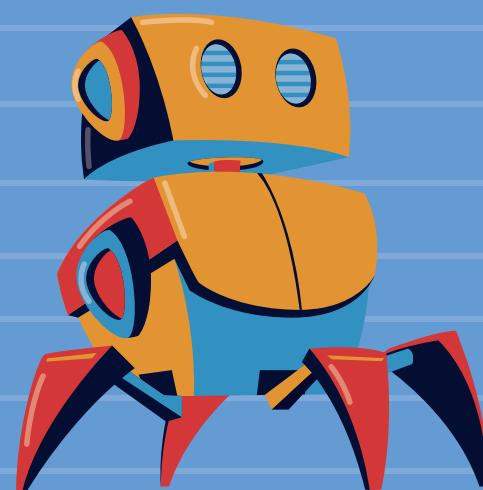
Ja yëxki'pyët eje myëëtëp majtsk ja pää'têyuupyë, laterales, paralelas de orientación fija, mitë awëtejtp oy'ojsää.

Ja pujx tsyo'ntajken

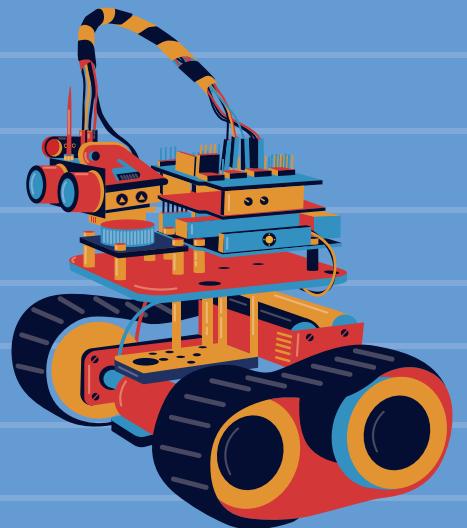
# Tsä y'a'äätspaky

Jajp jatu'ukpiky ja robots móviles näxwënyë'pyëtë:

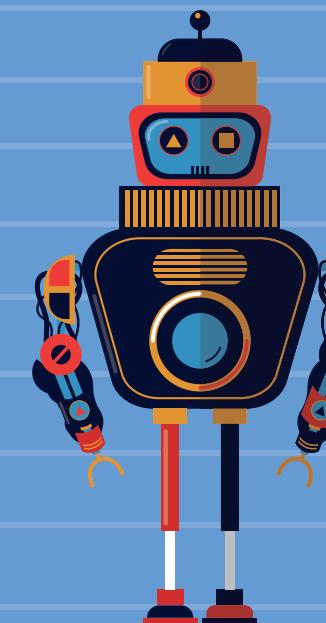
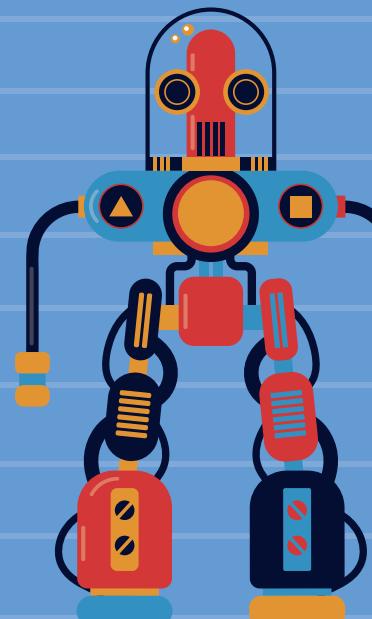
Robots tekymyëët (jatë'n kyaxë'ëktë tsäm ja pëejxy ).



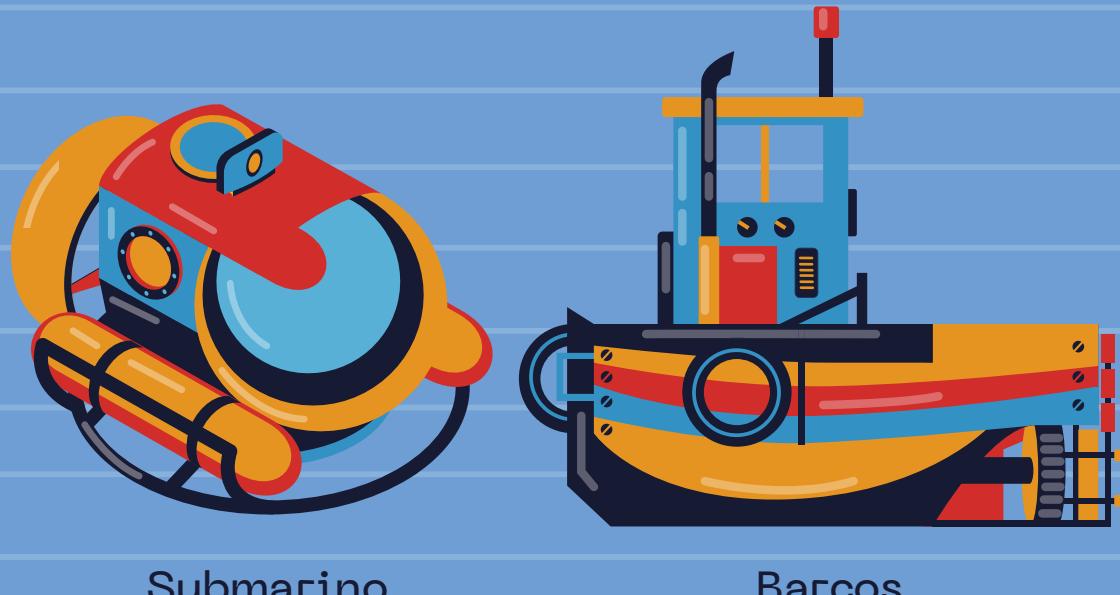
Robots con orugas (jatë'n yiktejtë ku kyaxë'ëktë tsäm ja mutsk jëyujktë).



Robots humanoides/majtskteky. (mitë jää'yën kaxë'ktëp).



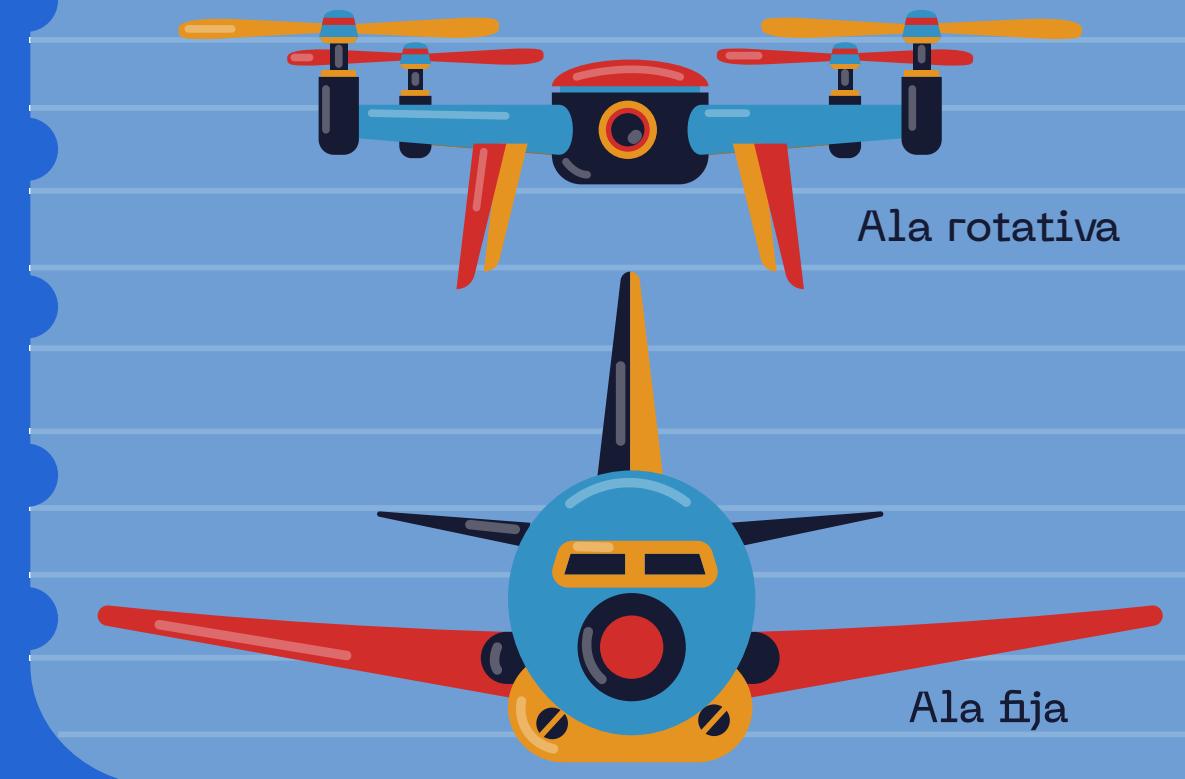
Ja pujxtsyë mitë nëyumtëp



Submarino

Barcos

Ja pujxtsyë mitë yiktuntëp ja control



# Mä tsyuntë Yä'ät pujxtsyë

1.3

Tëe ja robótica tyékatsy jumëjt ampy, yä'ät ejxpéjkën kajaa ja tyunk tmëetë, wan t'uk'ijxyäm mitë ja'ty: Robot de telemando (ja' mitë yik atëtstëp jakam'ampy mëet tu'uk ja pujx); Asistencia médica (mitë ja tséyepë yiktuntëp); Robots industriales (ja' mitë ja jää'y yiktuntëp jam tyunk jëpäm); Asistencia en rehabilitación. (ja pujx mitë putéjkëp jëts ja jää'y y'atëetstëp); Trajes de combate (ja' mitë ja jää'y napyéktääjkëtëp); Humanoides y droides (ja' mitë kaxéktëp tsäm ja jää'y).

## Ja' mitë yik atëtstëp jakam'ampy mëet tu'uk ja pujx

Yä'ät pujx yiktuntëp ja control, pën yik'atëtstëp jakam'ampy ja jää'ytyë.

Ejxëm ja drones jatë'n yik'tejtëp ja pujx mitë kakwëtejtëp:

- Yä'ät pujx yik tuntëp mä ja kuyäjtënnaxy, uk ja ëepëtë, xuxpëtë, naytë'n tyiku'ntë ja jää'ytyë mitë pelicula yik'ëe'yëtëp, kawënäkpiy ja tyunk tmëetëtë.
- Nayitë'n tuny yä'ät pujx, kyajxtëp jëts pyakontëp oy tyii'jë, ti tsojkëyëp ku ja'yë jye'mtsyët 2.2 kg, naytë'n tyuntë mëet ja empresas mitë yik xëtejtëp Fedex jëts Amazon.
- Ku ja ayoo'n tyunyë jyatyë: tëjkëtëp jëts jä'tëp oy'omäej, oy ja tuu' nukë tsep uk wénkë'm jä'tëp ja', ku ja ayoo'n jyaty kyupety y'ejxaatyëp ja jää'y, jëts naytë'n y'ijxyëttë.
- ja mutskkäjptë: putéjkëtëp jëts t'ëxä'ätëp ja jeyujk mitë ja nepk wenk tyuntëp.

Xäätsö'ampy naytë'n yikpääty ja pujx mitë nøyumtëp, jëts tyékkëtë nøyëtypy. ti yik ejxaapy:

- Jëts ja ujts, jeyujktë mitë jam nøyotp tsënaatyëp yik'ejxy'ettëp.
- naytë'n t'ejxtëp jëts t'ejxyëttëp ja cascós jëts tanques mitë yi barcos yiktuntëp.
- naytë'n tyuntë ku ja mëj ayoo'n yikpääty nøyëtypy.



Ejemplo:  
Robot aéreo de  
telemando



# Mä tsyuntë Yä' ät pujxtsyë



Mitë ja tsëyëpë yiktuntëp

## Robot PILLO

Yä'ät pujx yik'tejtëp robots de almacenamiento y distribución de medicamento.

Ja empresa pilo Health ojts tyik'eyë ja robot mitë xyëew'ajtypy PILLO sanitario yä'ät pujx pyutëjkëp ja tsëyëpë, jëts ja matyä'äky tyäktëp jakam ampy, ku ja jää'y ka' tmëtapyäaty tsutsë ja tsëey t'uuktët jëts tsä tyektu'ntët. yä'ät pujx yiktumpy ja exkäjpën wunpojk.



## Robot Da Vinci

Ja robótica pyutëjkëtëp yikxon ja tsëyëpëtë; ku ja tsep tank ka'pxy tsuj ja tyik'eyëtë, ku Tu'uk ampy ja neekxy nu ku ama'äy. Ja pujx tutä'äky ja tank tyik'eyëtë jëts ja neekxy kaja ayo'on tpa'ät.

Ja pujx mitë yiktejtëp quirúrgicos tsäm ja da Vinci, pyutëjkëtëp ja tsëyëpëtë jëts tyuntëp ja tsep tank, ka'pxy ja tyunk tyikpëtsëmtë.



# Mä tsyuntë Yä'ät pujxtsyë

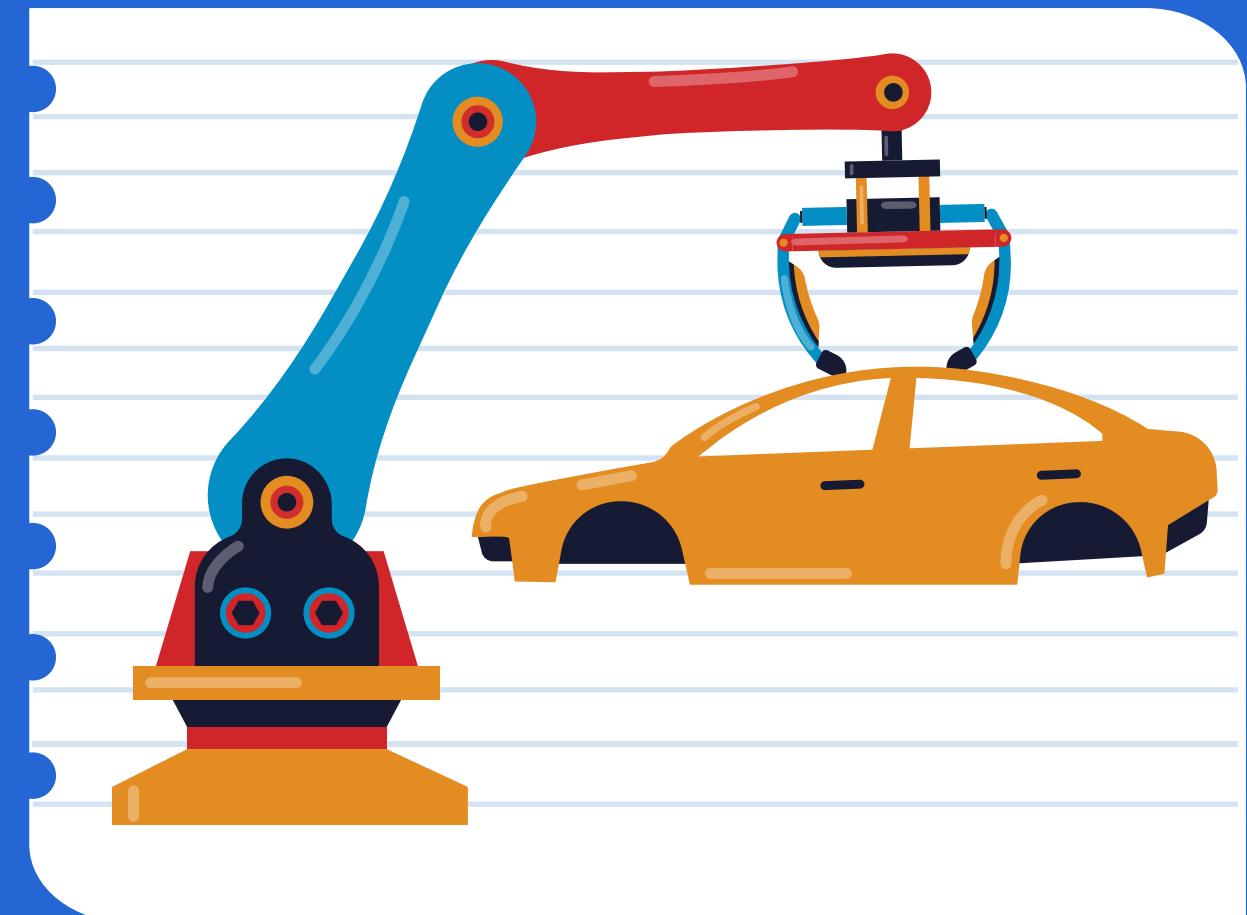
Ja' mitë ja jää'y yiktuntëp jam tyunk jëp'äm

Ja robots industrielas jatë'n t'ejxtë tsäm ja jää'y ja kyenky, yä'ät pujxtë tuntëp mëet tu'uk ja computadora. Ëy ku najää'wä'ämp ku yä'ät robots tükatsy ja tyunpajt tmëetëtë, jëts tükäjstëp ja tyunk mitë yik'ëe'yëtëp. Yä'ät pujx yiktuntëp ja jää'y jëts tyik'atë'ëtstëp ja amäay pëktä'äky, ja pujx atsaapyë, naytë'n ku ja pëktä'äky yikpatsimy jëts yikkatsjuky, jëts tatsukkëtë ja pujxtë mitë nukë jë'mtsy, naytë'n t'uktë, jëts tyiktse'të mëet ja pujx jëts ja ácidos, nayï'kyejpxtëp ja pëktä'äky mitë yiktejtëp sólidos, líquido, gases jëts ja aplicación de altas presiones, naytë'n t'ejxtë ku ja pëktä'äky ëy tsuj, ka'pxy pyëtsë'mt.

Ojts ja robótica yeeky jam mä yiktiy industria automotriz, naytë'n ojts tyiktuntë ku ja tükatsy mëj pujxwee'tpë ojts pyëtsë'mtë, jëts ojts tsyoo'ntä'äky jam década de los 90's. ja jää'ytyë mitë ojts tyik'ëyëtë ja pujx wee'tpëtë, yäjktëp wyinäjty ëy, tsuj, jëts ka'pxy ja pujxtë, yä'ät pujx ojts naytë'n yikpawenmay jëts ja jää'ytyë ka'ap ja ayoon tpäätëp, ejxyäm ja industrias mitë tyunk'äjtëp jëts tyik'ëyëtëp ja pujx wee'tpë yiktuntëp ja kujpk ja pujx mitë yiktejtëp robot.

Ja Instituto Americano de Robótica (RIA) nyëmatyäkypy ti ja robot industrial tyijp:

*Ja robot industrial, yik'ijxyëp ku ja jää'ytyë tmo'të ja ana'mën, naytë'n tmumatowtë jëts ja tyunk ëy, ka'pxy tyik'ëyëtë, yä'ät pujx ojts yik pawunmay jëts tyik'atë'ëtsp ja tükatsy pëktä'äky, jëts ja tunpajt.*



Jatë'n ojts tyikatsy, ja pujx putëjkënyëtëp wën'äjty mä ja tükatsy tunk, jëts ja jää'ytyë ka' t'uktunëtët ja mëjk tunk.

Ja pujx tsyo'ntajken

# Mä tsyuntë Yä'ät pujxtsyë

Ja pujx mitë putëjkëp jëts ja jää'y y'atë'ëtstëp

Jajp jatu'kpiky ja robot mitë yiktejtëp exoesqueletos, ku ja jää'y tyiktuntë yä'ät pujx, putëjkëtëp jëts ja y'atëtstëp ku ja pä'äm tmëëtëtë. Yä'ät pujx pyutëjkëtëp ja majää'ytyë jëts ja jënts y'etëp, mä ja tyunk tmëëtëtë jëts mä nyäxtë tyékëtë.



Ja' mitë ja jää'y napyëktääjkëtëp

Jam Rusia ojts tyik'ëyëtë ja jempyë wet mitë yiktejtëp trajes de combate, jam tyiku'ntë mä ja fuerzas especiales. yä'ät wet ojts tyik'ëyëtë ja Instituto Central de Investigación para la construcción de Máquinas de Presión, yä'ät wet yu'tsypy tu'uk ja exoesqueleto motorizado, protección balística, jëts ja casco mëët ja visor electrónico.

**Jajp jatu'uk ja robot mitë yikxëpajtëp Exoesqueleto Asistencial Híbrido (HAL)** pën ojts tyik'ëyëtë, tu'uk ja Universidad Japonesa mitë xyëëw Tsukuba. mëët ja empresa robótica CYBERDYNE. ojts ja'tyik'ëyëtë jëts ja jää'y tyikpëtëktëp wëxijkxy myäjk kilos mëët tu'uk jaty ja kyë', jëts ku ja tyeky pyutëkëyëp ëy ja tukmakë'pxy kilos tyikpëtëëktëp.

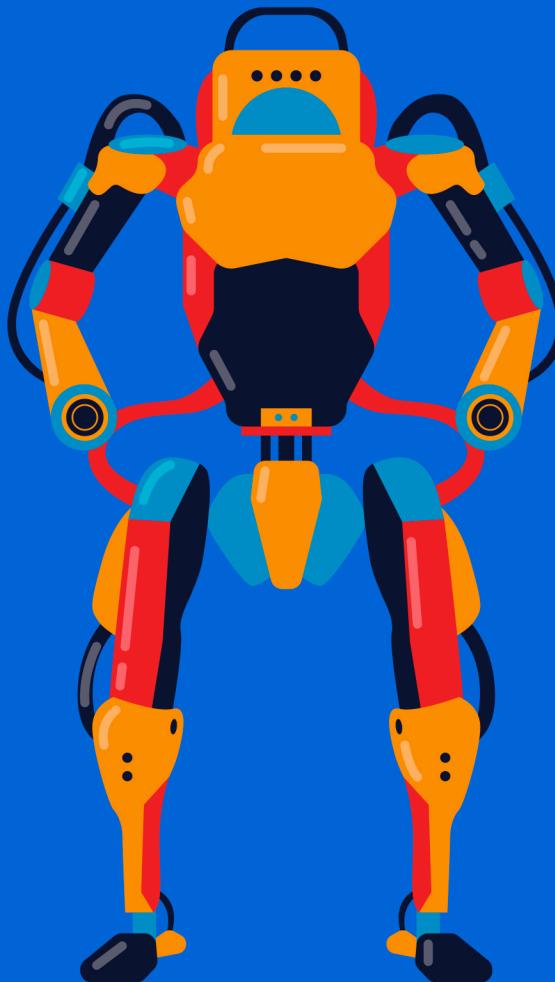


Ja pujx tsyo'ntajken

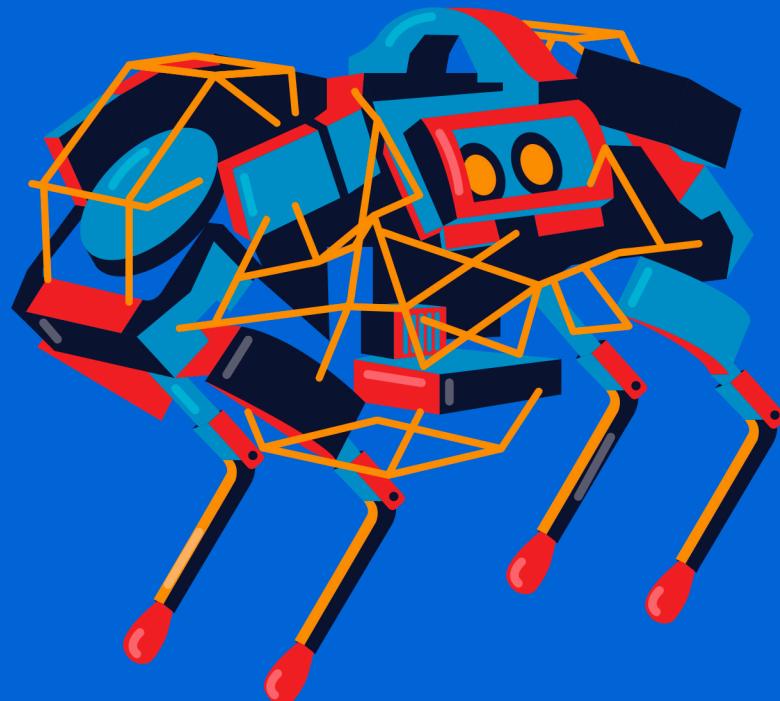
# Mä tsyuntë Yä' ät pujxtsyë

## Ja' mitë kaxë'ktëp tsäm ja jää'y

Boston Dynamics jatë'n yik'ijxyëp tu'uk ja empresa de ingeniería jëts robótica, mitë tyunk'äjtëp ja pujx tyik'ëyëtëp. ku ojts tsyoo'ntä'äktë, ojts ja tyuntë mëet ja Instituto Tecnológico de Massachusetts (MIT), ja jää'ytyë mitë jam tuntëp ojts tyik'ëyëtë ja robots mitë pujttëp jëts tuntë tsäm ja jyujky'ättë. ti y'ëjxaatyëp ja mëj jëts kajaa wunmää'ny mä tyiku'ntë ja principios del control dinámico jëts ja equilibrio mëet ja sofisticados diseños mecánicos, electrónica de vanguardia jëts ja software para percepción, navegación jëts inteligencia. Ja diseño mitë ja yik'ëë'yëtëp yikxëpajtëp:



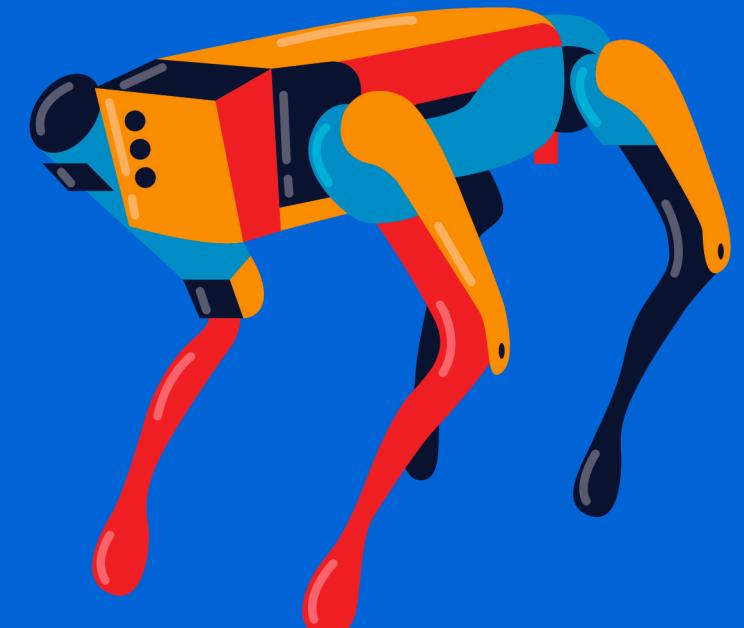
ATLAS



WILDCAT



BIGDOG



SPOT

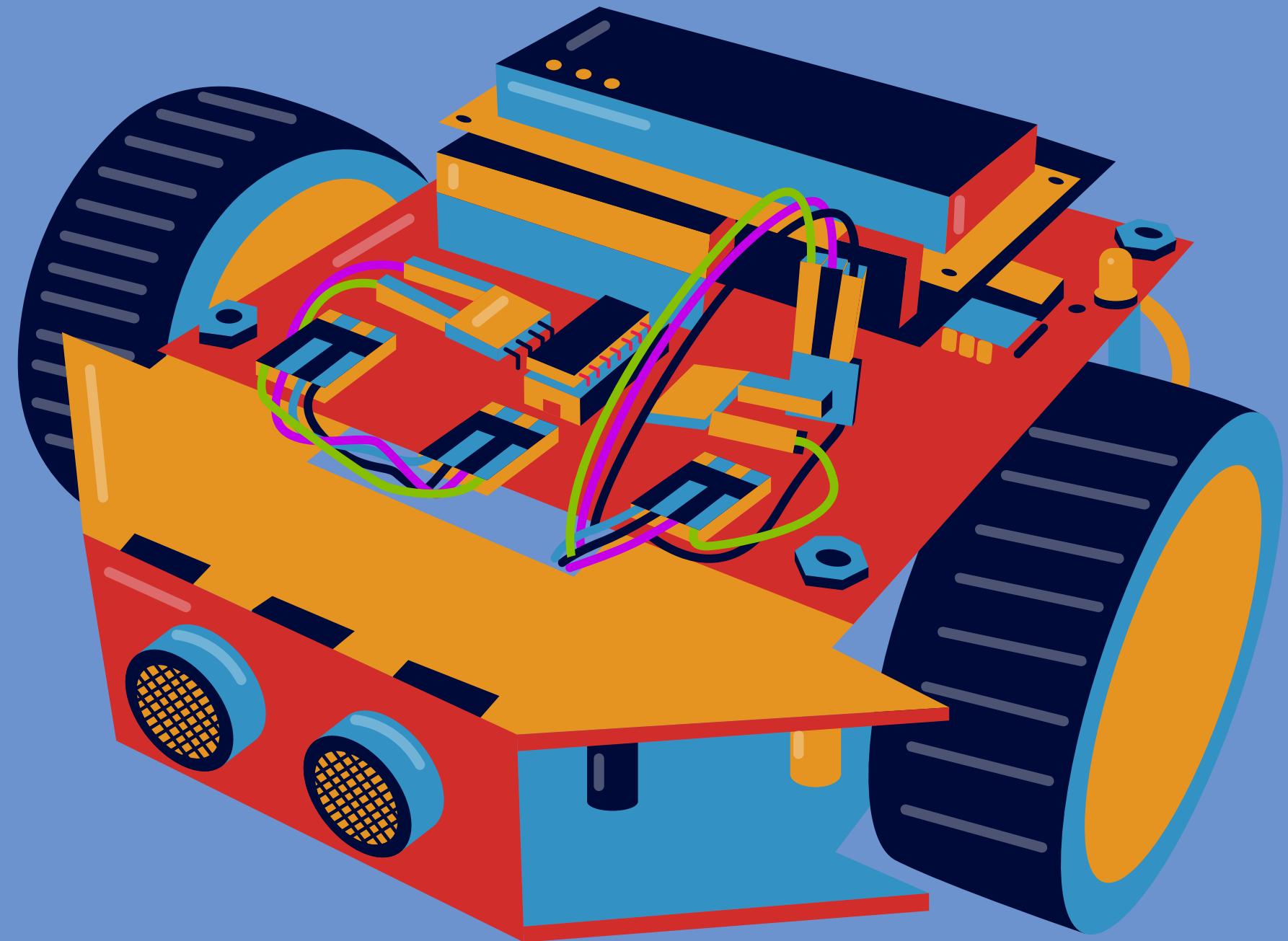
Myikwäätsootëp just x'ejxtët xko'tët yä'ätpë awä'äts'äjtën [www.bostondynamics.com](http://www.bostondynamics.com) mä xpääätët tijaty ja pujx kë'm atëtspë tyuntëp ka mupikyëm.

Cap. 02

# Ti yik tuntëp ja pujxtsyë

Ti yik tuntēp ja pujxtyé

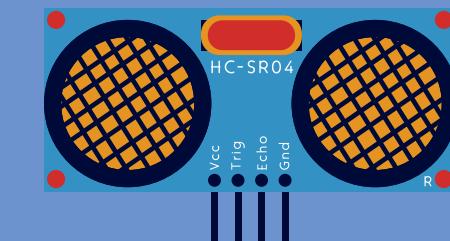
# Myëjawyeen pujx móvil



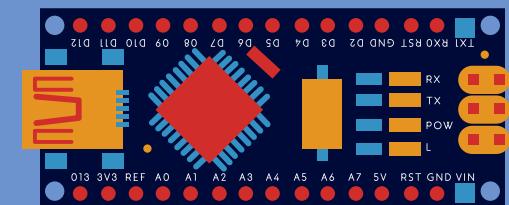
21

Ti myëetëp ja'n  
myëjawyeen pujx móvil

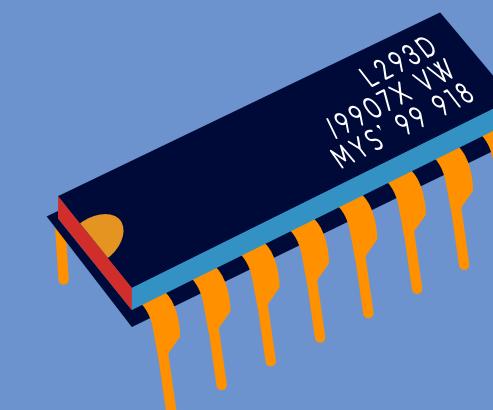
Sensor Ultrasónico  
SRF04



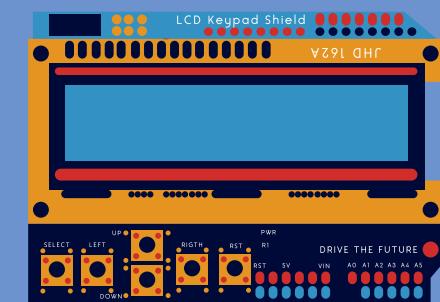
NANO



Puente H L293D



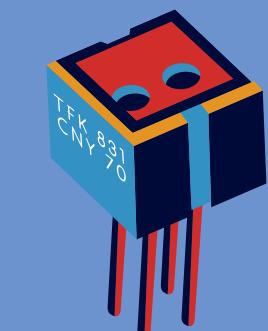
LCD Keypad Shield



Motor Reductor  
con Llanta



Sensor Óptico  
CNY70



# Componentes

2.2

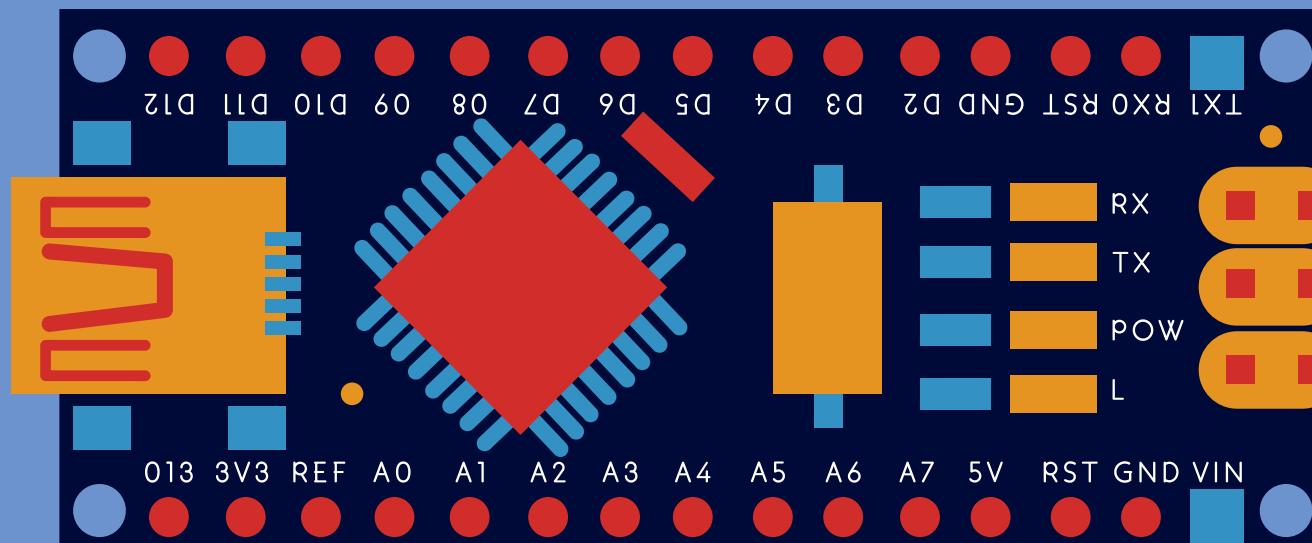
## Controlador

### Arduino - NANO

Plataforma de prototipos interactivos. ja' yiktawunmaatyëp ja jää'ytyë mitë eëtëp, ajtstëp uk oy o tii tyuntëp, uk tsuj t'ejxtë jëts tsuj tjawëtë yä'ätpë pujx. Yä'ät módulo robótico yiktuntëp tu'uk ja plataforma de hardware libre basada en Arduino mitë yiktejtëp Controlador NANO.

Yikjää'wip mä nyaxy ukmää y'ity mëet ja sensores, jëts ëy naytë'n yik tanëpëktääkëtë ja jäjtë, motores jëts jatu'uk pujxtë.

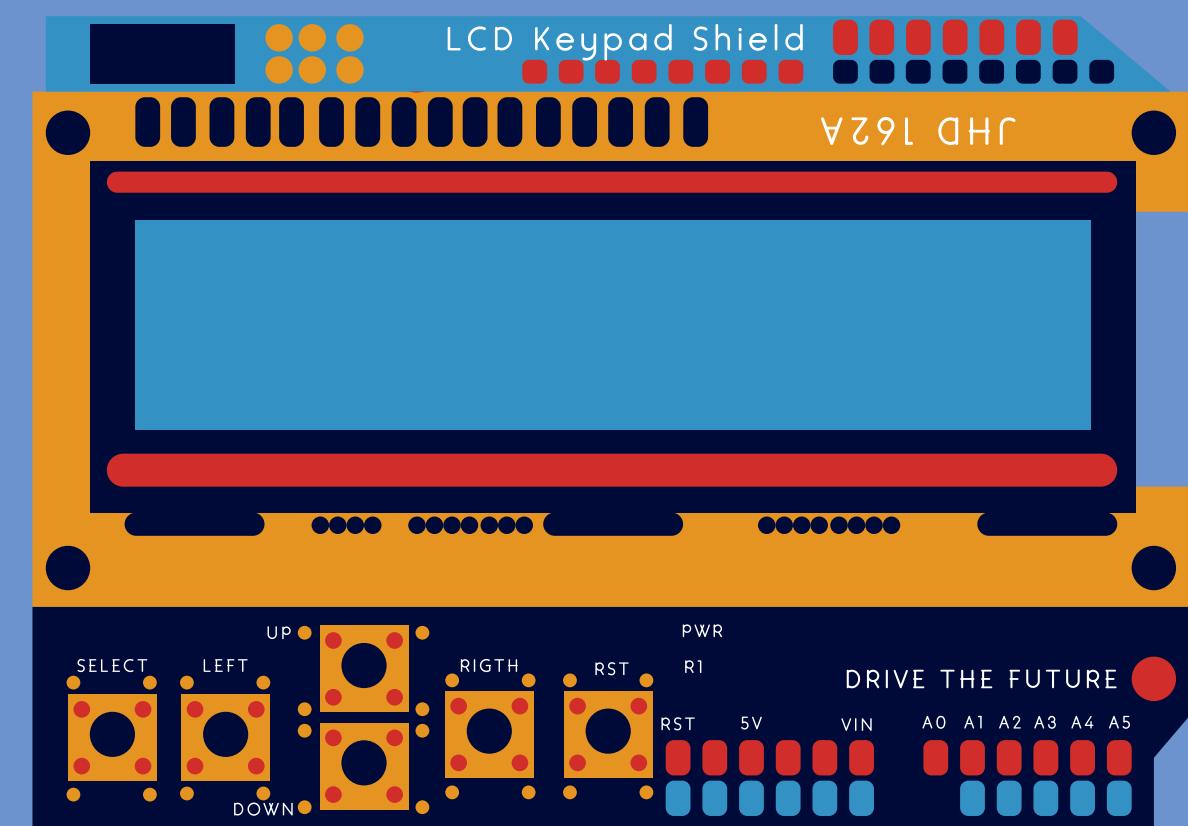
Yä'ät ampy, ja proyectos mitë tuntëp mëet Arduino, tsäm ja pujx mitë yiktejp Controlador NANO, oy tyuntëp natyu'uk jats mä nyäxtë jëts tyékëtë.



## Dispositivos Mä tyiknäxtë jëts tyikpëtsëmtë

### LCD Keypad Shield

LCD, jatë'n txëewë tu'uk ja pujx mitë ojts tyik'ëyëtë jëts nyänky'ejxëp ja tékatsy némattyä'äky, jatë'n ja yik'ijxyë amëxân tsäm ja forma gráfica. Ti ja LCD tyijpy, mä ja ääww ayuujk inglés, Liquid Crystal Display (Display de Cristal Líquido). Yä'ät pujx myëëtëp makoxkween ja atse'tpë, mä tyiknäxtë ja ääw ayuujk jëts naytë'n tmëëtëtë ja terminales jëts ja conexiones.

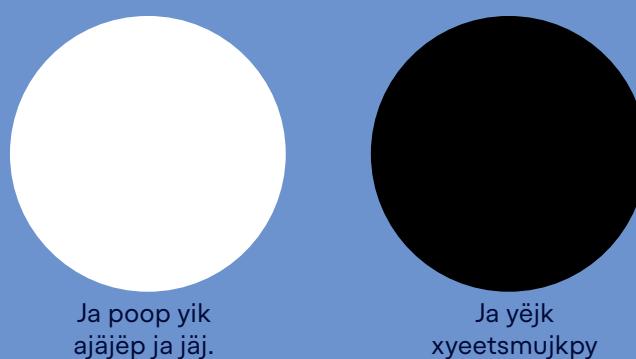
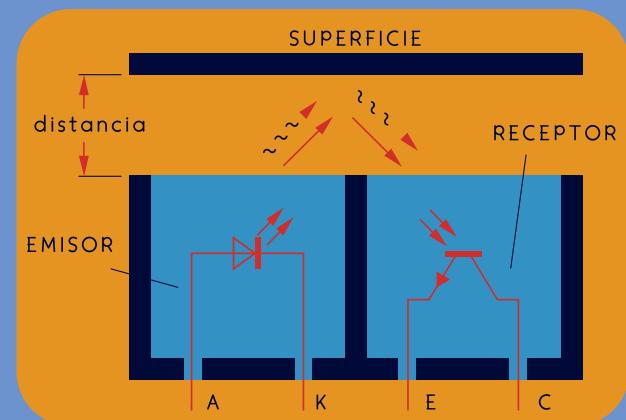


# Componentes

## Sensores

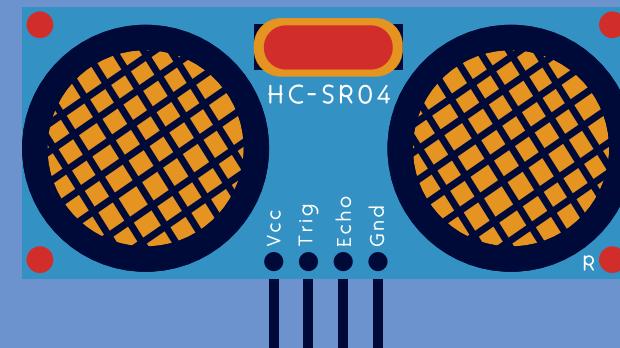
### Sensor Óptico CNY70

Yäät controlador NANO tsyékypy tu'uk ja sensor mitë jatë'n kaxë'kp tsäm ja "ween". yäät sensores yik'ijxyétüp tsäm ja CNY70.



### Sensor Ultrasónico SRF04

Jëts ja controlador NANO kyejxpy ja jakam uk ja wënkon pëktääky, ëy tyiku'ntët ja ecolocalización.



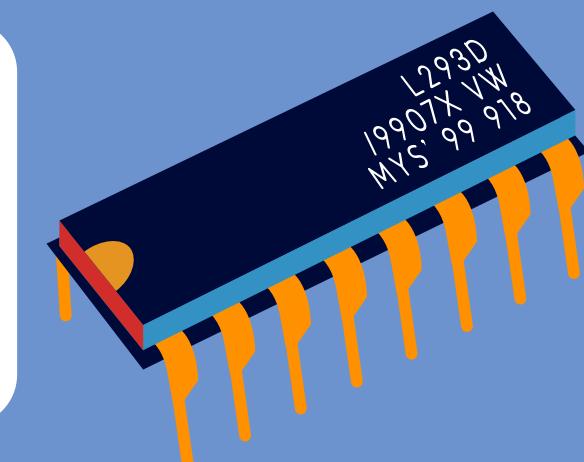
## Actuadores

### Puente H L293D

Ja controlador NANO tyumpy jëts ja dispositivos mitë mejk tuntëp y'atë'etstëp, tsäm ja motores de CD, tsojkëyëp naytë'n ja circuitos adicionales.

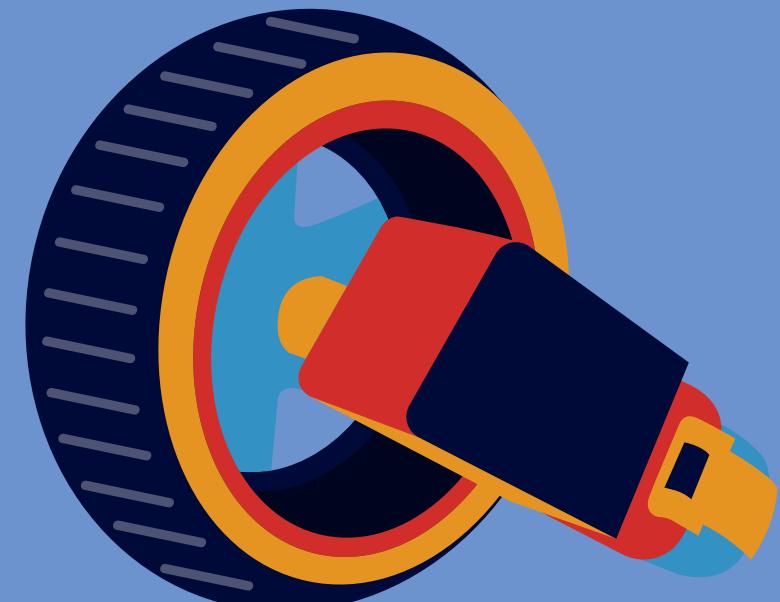
EN	1A	2A	FUNCIÓN
H	L	H	Awëtejtëp akää'ny ampy
H	H	L	Awëtejtëp anäjy ampy
H	L	L	Pojen tyanëtë
H	H	H	Pojen tyanëtë
L	X	X	Pojen tyanëtë

L=bajo H=alto X=no afecta



### Motor reductor con llanta

Jëts ja robot y'atë'etst, tsojkëyëp majtsk ja motores mëet ja tiyuupyï. yäät motores namyujkëtëp mä ja etapa de potencia tejë, jëts ja controlador NANO tyik'atë'etsp. Oy yäät pujx tyëktuntët hasta 6v (tutujk v) con 200 RPM y 3v (tëkëëk v) con 150 RPM.



# Cap. 03

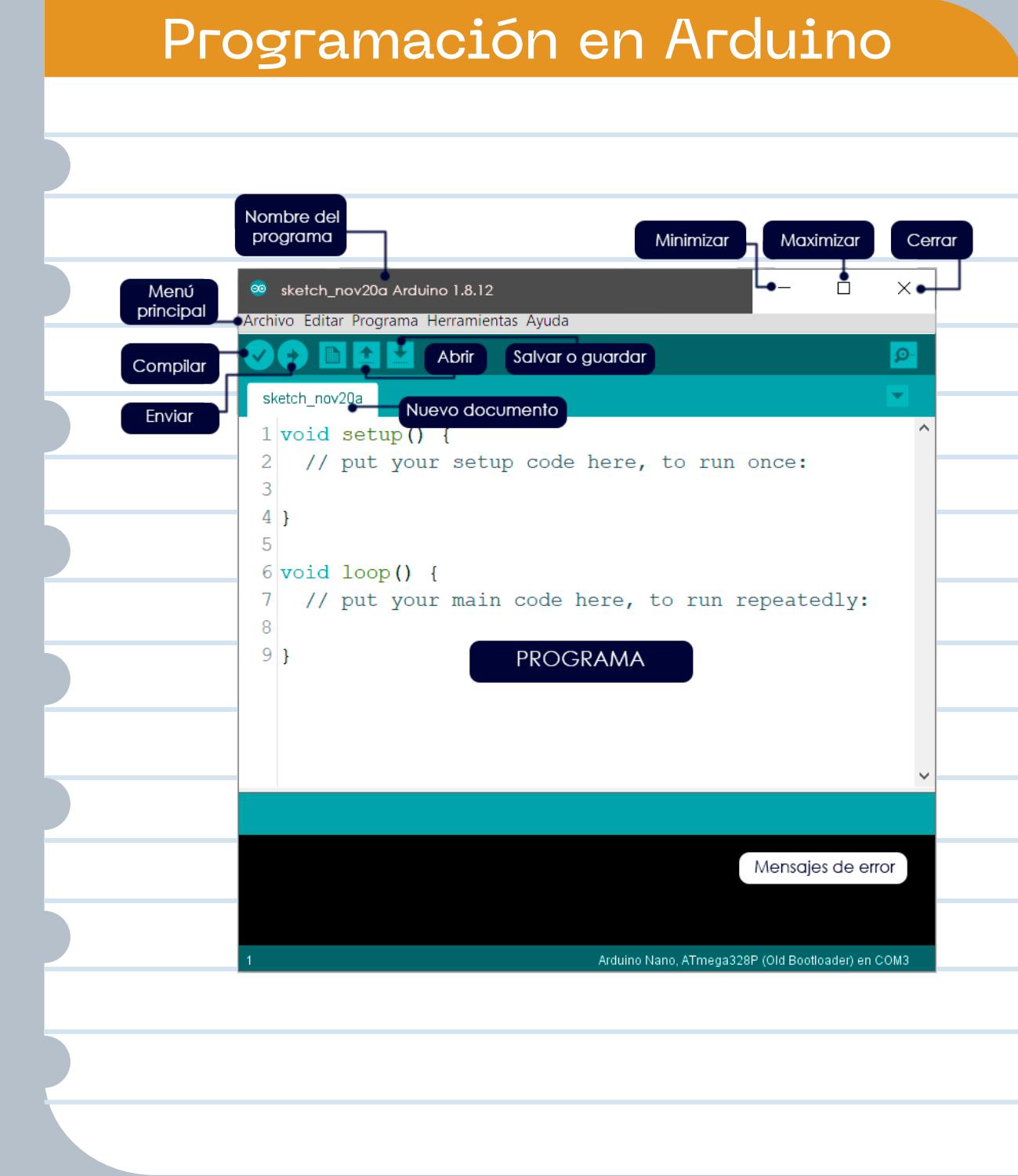
## Programación básica del Robot móvil

# Programación en Arduino

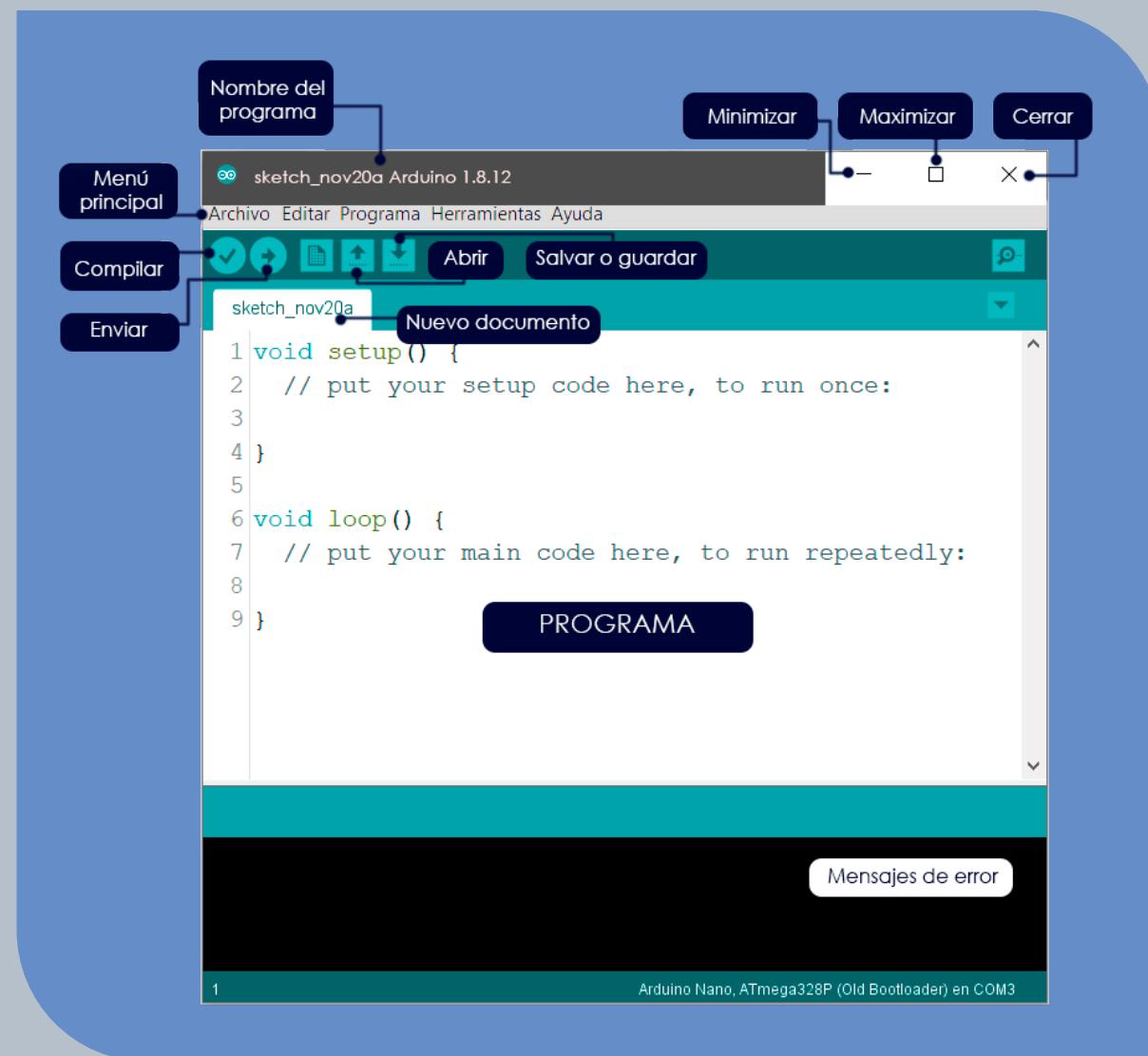
Mitë myëjawyeen tsojkëyëp ja ana'mën jëts ja robot ëy ka'pxy ja tyunk tyék'ëytëp, ja jää'ytyë tya'akuwää'nëtëp ja pujxtë jëts këttu'uk jaty ja tyunk tyék'ëytët. mä ja tükatsy äaw ayuujk tyiktu'ntë, yä'ät ana'mën yik ijxyëp tsäm ja programa.

Yä'ät programas, yiknäjxtëp jam mä ja robot ja kyukë'ëxy tmëetë, jatë'n yiktejtë ja robot ja kyukë'ëxy, tarjeta controladora Arduino, jëts ja pujx tpatu'ntët jëts tnëmato'tët ëy jëts ka'pxy ja tunk mitë ja jää'ytyë y'amëtootëp. Yä'ät pujxtë Arduino myëëtëtëp tu'uk ja tunpajt digital mitë yik tejp entorno de desarrollo integrado (Arduino IDE), jam ja jää'y tjä'të ja ana'mën mitë tsojktëp jëts ja Arduino tu'nt.

Nayë yä'ät IDE myëtëp ja majataäjkën jëts t'ejxtëp pënë ka'pxy ja ana'mën të yikjä'äy, naytë'n tnëkäjpxtë pënë ka'apxy ja äaw ayuujk të yik jää'y, jëts naytë'n ku kaa kyapxyë, jëts ku kapxy ja jyaapyajtën, ku ka tpäätë ja tükëëyën jaaky, kyajxtëp jam mä ja Arduino, pën yik'ëëyëtëp wyëñäjty tyimkyapxy ja tyunk, mitë yik'amëtootëp.



# Programación en Arduino



Anëkijxpy'ampy yikpätp tu'uk ja barra mä myiny ja programa jë xyëew, jëts jatu'uk ampy yikpätp ja atse'tpë mitë nëmtëp ku nyikmujtska'amp, nyikmëja'amp jëts nyik'atujkëm. Mä ja barra jam kyaxi'iky jam yik pääty ja menú principal.

Tu'uk jaty ja atse'tpë tmëetëtë ja awä'äts'äjtën ku xkuppanëp jëts ja menú pyëtsë'mt, kawënääk'ampy ja opciones pyitsimy. jam ja menú de archivos tmëetëtë ja tükatsy opciones, tsäm ku nyik awääjtsënt, nyik atujkënt, uk tu'uk jemy archivo nyék'awääjtsëm.

Jaak apëtki'ipy mä ja barra de menu principal kyaxi'iky, yik pätp ja barra de botones. jaatso'ampy pojën jëts jäjtnë ntëjka'am mä ja opciones mitë mouse yiktumpy jajp programa jëtypy.

- Ja myëjawyeen atse'tpë nëmp compilar, yä'ät atse'tpë jatë'n nyänky'ijxyëyë mëët tu'uk ja "palomita" yë' nyikajpxpy pënë ka'pxy ja äaw ayuujk të yik jää'y. Ëy yä'ät äaw ayuujk nyëktëkäjtsëm jëts ja Arduino tjääkyukët jëts tu'nt, ku yik kuppanë ja atse'tpë, jëts ka' yik pääty ja tyik'ëyën mä ja äaw ayuujk të yik jää'y, ka'pxynyë wyënäjty ja programa, jëts ja Arduino tyék'ëeyënyëp ja tyunk.
- Ja jatu'uk atse'tpë, mitë anäjy'ampy kaxë'kp, yik'ijxyëp ku tyikpety jëts tkexy ja äaw ayuujk. Ku ja programa ka'pxy të nyaxy, yikkajxp jam Arduino jëts tu'nt ja ana'mën mitë ja jää'y të tjä'të.
- Ja tükëek atse'tpë nukë jäjtnë yiknëmatëy pënë tëe ja computadora xyiktunë, ja myëjawyeen nëmp Nuevo, yik awatsypy tu'uk ja jempyë ventana, ja myumajtsk nëmp Abrir, tyumpy jëts njaak jää'yënt ja programa mitë ojts nyik'ëe'ya'amp, Guardar, mitë pyëjk ji'kpy ja äaw ayuujk ku nyik'atujknë.

Apëtkipy'ampy mä ja barra yikpääty, yik mëëtëp ja awä'äts'äjtën jëts yik jää'y ja ana'mën mitë ja Arduino wyinäjty pyatumpy. Ëy ku'njäämyajtsënt ku yä'ät äaw ayuujk, ja mayúscula jëts minúscula tutä'äky jëts ka'pxy yikjää't.

# Programación en Arduino

Tu'uk jaty ja programas tmëetë kuwänii tükéek ja secciones.

Mä tsyoo'ntä'äky yik pätp ja sección mä nyimy "Sección de declaración de variables", mitë yik xëpajtëp sección del QUE.

Yä'ät sección, yikpëktäkp ti nyiktunwäänyäm (ja' mitë yiktejtëp variables, definiciones, liberías, etc.). xyäätso'ampy tjä'të mitë yikxëpajtëp variables, mitë wyinäjty yiktuntëp mä ja programa.

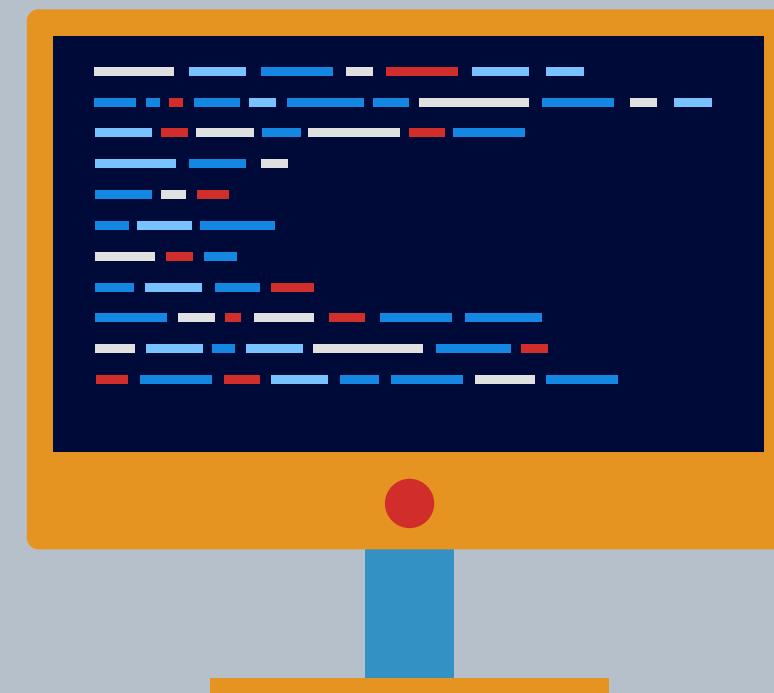
Xyäätso'ampy yiktuntëp ja awääjts mitë tyejtëp amëxän llaves de apertura y cierre. ja' myëjawyeen tsojkëyëp jëts nyik'ëe'ya'amp ja'jp programa jëtypy. Jatë'n yikxëtiy void setup(). yä'ät sección yiktejp COMO, yä yik nëkajpx tsutsë nyiktsoon'tääjkën ja variables. pënë tëjkëtëp uk pëtsëmtëp, analógicas o digitales.

Ëy ku najää'wä'ämp ku tukki'yi mitë ejxyää sección yikpätp, ja'yë tuk'onk yiktuny, ku ja Arduino nyik tëpyëjkën uk nyik tsoo'ntääjkën jatëkoojk.

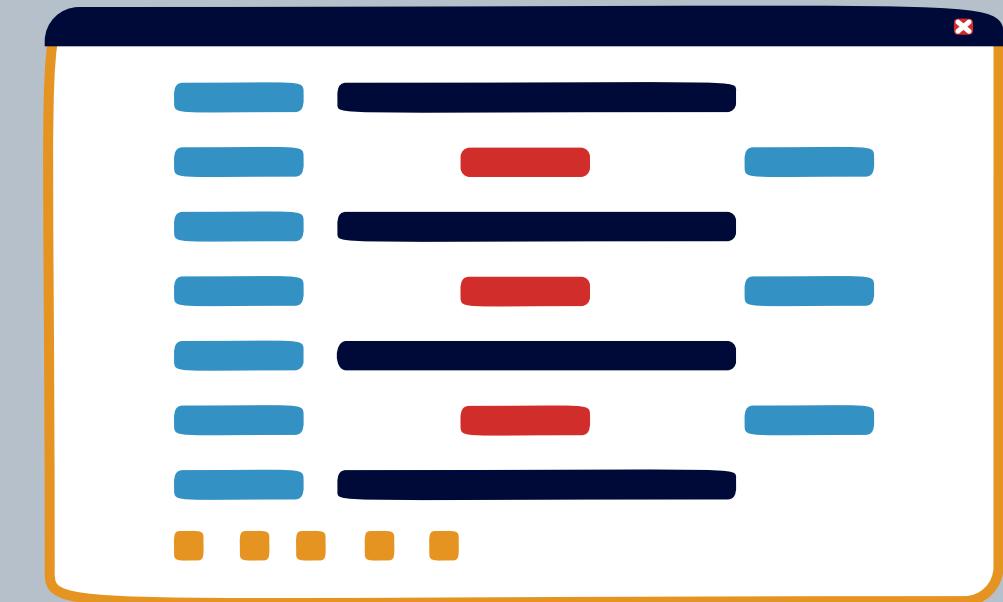
Jaak apëtki'py'ampy mä ja COMO (tsutso) kyaxi'iky, yikpätp ja sección mitë yik tejp "void loop()". mä naytë'n tyiku'ntë ja awääjts mitë amëxän yiktejp llaves, ja' mä myiny ja ka'pxy ana'mën, (CUANDO) juunë y'ëyë jëts nyiktuunëm ja awääjtsën jëts ja pëtsëe'mën, ku nyiktuunën ja condiciones jëts tyiktëkätstët, ja' mitë yik tejp flujo de un programa.

Tsäm ejxa'n'iixyäm, yä'ät sección yik tejp (CUANDO) juunë. mitë ejxyäätso'ampy yik pätp tyumpy wyenäjty tu'uk jaty ja ana'mën, kunëm njä'äjtëm mä ja awääjts y'atuky (llave de cierre) ( ) , yäätso'ampy ja flujo del programa wyumpity, mä ja awääjts y'awä'ätsy (abrir llave) ( { ).

Pats jatë'n tyik xëpattë loop, mitë tyijpy tejpxy, tsäm tu'uk ja tejpxy yik'axëtsy, ka' yik tunk'aty määtso'ampy xyiktsoontä'äky, ejtp wyumpity mä ojts tsyoo'ntä'äky, mä yik axëtsy.



# Programas 3.3 para el uso de los componentes



Blink

```
1 //////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: Blink.- Programa que enciende un LED por 1seg y lo apaga 1seg //
4 //////////////////////////////////////////////////////////////////
5
6 // ¿Qué? --> tipo de dato usamos:
7 int led = 14; //Se declara la variable led como un dato de tipo entero y se le
8 //asigna el pin número 14.
9
10 // ¿Cómo? --> usamos los datos (entrada o salida):
11 void setup(){
12   pinMode(led, OUTPUT); //Inicializa el pin digital 14 (led) como salida.
13 }
14
15 // ¿Cuándo?
16 void loop(){
17   digitalWrite(led, HIGH); //Enciende el LED (HIGH es 1 binario o 5[V])
18   delay(1000); //Espera por 1s
19   digitalWrite(led, LOW); //Apaga el LED (LOW es 0 binario o 0[V])
20   delay(1000); //Espera por 1s
21 }
```

LCD

```
1 //////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: LCD.- Programa que muestra en el LCD el mensaje "Hola mundo" //
4 //           y el valor de una variable //
5 //////////////////////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 // ¿Cómo? --> usamos los datos (entrada o salida):
12 void setup(){
13   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
14 }
15
16 // ¿Cuándo?
17 void loop(){
18   lcd.setCursor(0,0); //Coloca el cursor en la posición 0,0 del LCD
19   lcd.print("Hola Mundo"); //Imprime el mensaje "Hola mundo"
20   lcd.setCursor(0,1); //Coloca el cursor en la posición 0,1 del LCD
21   lcd.print(millis() / 1000); //Imprime la información de la variable millis
22 }
```

# Programas para el uso de los componentes

## Motores

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: Motores.- Programa que hace que el robot avance 2seg y se detenga 1 seg  //
4 ///////////////////////////////////////////////////
5
6 // ¿Qué? --> tipo de dato usamos:
7 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
8 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
9 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
10 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
11
12 void setup() {
13 // put your setup code here, to run once:
14 pinMode(13,OUTPUT);
15 pinMode(ad1,OUTPUT);
16 pinMode(at1,OUTPUT);
17 pinMode(ad2,OUTPUT);
18 pinMode(at2,OUTPUT);
19 pinMode(Adelante,INPUT);
20 pinMode(Atras,INPUT);
21 pinMode(Derecha,INPUT);
22 pinMode(Izquierda,INPUT);
23
24 Serial.begin(9600);
25
26 }

```

## Componente CNY70

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: CNY70.- Programa que lee el valor de los sensores CNY70 y los muestra en el LCD  //
4 //                                              //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 int ValorCNY_I; //Variable para guardar el valor del CNY Izquierdo
12 int ValorCNY_D; //Variable para guardar el valor del CNY Derecho
13 int ValorCNY_C; //Variable para guardar el valor del CNY Centro
14
15 // ¿Cómo? --> usamos los datos (entrada o salida):
16 void setup(){
17 lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
18 }
19
20 // ¿Cuándo?
21 void loop(){
22 ValorCNY_D = analogRead(A4); //Leemos el valor de los CNY's
23 ValorCNY_C = analogRead(A5);
24 ValorCNY_I = analogRead(A6);
25
26 lcd.setCursor(0,1); //Borramos datos del LCD
27 lcd.print("          ");
28 lcd.setCursor(0,0); //Mandamos a imprimir el valor del CNY Izquierdo
29 lcd.print("IZQ");
30 lcd.setCursor(0,1);
31 lcd.print(ValorCNY_I);
32 lcd.setCursor(6,0); //Mandamos a imprimir el valor del CNY Centro
33 lcd.print("CENT");
34 lcd.setCursor(6,1);
35 lcd.print(ValorCNY_C);
36 lcd.setCursor(13,0); //Mandamos a imprimir el valor del CNY derecho
37 lcd.print("DER");
38 lcd.setCursor(12,1);
39 lcd.print(ValorCNY_D);
40 delay(200); //Se requiere de un tiempo de retardo por estabilidad
41 }

```



# Programas para el uso de los componentes

## Ultrásónico

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)           //
3 // Programa: Ultrásónico.- Programa que lee el valor del sensor ultrásónico  //
4 //             y lo muestra en el LCD                   //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 long distancia; //Variable para guardar el valor de la distancia al objeto
12 long tiempo; //Variable para guardar el tiempo que tarda en rebotar el pulso
13
14 // ¿Cómo? --> usamos los datos (entrada o salida):
15 void setup(){
16   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
17   pinMode(13, OUTPUT); //Coloca al pin 13 como salida: genera pulso ultrásónico
18   pinMode(12, INPUT); //Coloca al pin 12 como entrada: detecta el tiempo de
19   //rebote del pulso ultrásónico
20
21 // ¿Cuándo?
22 void loop(){
23   digitalWrite(13,LOW); //Por cuestiones de estabilidad del sensor
24   delayMicroseconds(5);
25   digitalWrite(13, HIGH); //Envío del pulso ultrásónico
26   delayMicroseconds(10);
27   tiempo=pulseIn(12, HIGH); //Mide el tiempo transcurrido entre el envío del
28   //pulso ultrásónico y el rebote del mismo
29   distancia= int(0.017*tiempo); //Fórmula para calcular la distancia
30
31   lcd.setCursor(0,0); //Mandamos a imprimir la distancia calculada
32   lcd.print("Distancia=");
33   lcd.setCursor(10,0);
34   lcd.print(distancia);
35   lcd.print("cm ");
36   delay(500);
37 }

```

## Botones

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)           //
3 // Programa: Botones.- Programa que muestra en el LCD un mensaje con el nombre del //
4 //             botón presionado y su valor analógico      //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include <LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd(6,7,2,3,4,5); //Pines del LCD
10
11 int CDA_teclado = 0; //Variable que guarda la información del teclado
12
13 // ¿Cómo? --> usamos los datos (entrada o salida):
14 void setup(){
15   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
16   lcd.setCursor(0,0);
17   lcd.print("Presiona boton");
18 }
19
20 // ¿Cuándo?
21 void loop(){
22   CDA_teclado = analogRead(A7); //Lee el valor del pin 0 del puerto analógico. Cada botón
23   //tiene asignado alguno de los siguientes valores: 50,150,300,450,650,1000
24   if(CDA_teclado < 50){ //Se presiona el botón derecho 50
25     lcd.setCursor(0, 1);
26     lcd.print("Derecha      ");
27     lcd.setCursor(8, 1);
28     lcd.print(CDA_teclado );
29     delay(200);
30   }else if(CDA_teclado < 150){ //Se presiona el botón arriba 150
31     lcd.setCursor(0, 1);
32     lcd.print("Arriba       ");
33     lcd.setCursor(8, 1);
34     lcd.print(CDA_teclado );
35     delay(200);
36   }else if(CDA_teclado < 400){ //Se presiona el botón abajo 400
37     lcd.setCursor(0, 1);
38     lcd.print("Abajo        ");
39     lcd.setCursor(7, 1);
40     lcd.print(CDA_teclado );
41     delay(200);
42   }else if (CDA_teclado < 550){ //Se presiona el botón izquierdo 550
43     lcd.setCursor(0, 1);
44     lcd.print("Izquierda    ");
45     lcd.setCursor(10, 1);
46     lcd.print(CDA_teclado );
47     delay(200);
48   }else if (CDA_teclado < 850){ //Se presiona el botón selección 850
49     lcd.setCursor(0, 1);
50     lcd.print("Seleccion    ");
51     lcd.setCursor(12, 1);
52     lcd.print(CDA_teclado );
53     delay(200);
54   }else if(CDA_teclado > 1000){ //No se presiona ningún botón 1000
55     lcd.setCursor(0, 1);
56     lcd.print("No boton     ");
57     lcd.setCursor(10, 1);
58     lcd.print(CDA_teclado );
59     delay(200);
60   }
61 }

```

# Programas 3.4

## para tareas específicas

### Seguidor de líneas

```

1 ///////////////////////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA) //
3 // Programa: Seguidor.- Programa para realizar el seguimiento de un linea //
4 //             negra sobre un fondo blanco //
5 ///////////////////////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include<LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd (6,7,2,3,4,5); //Pines del LCD
10 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
11 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
12 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
13 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
14 int ValorCNY_I; //Variable para guardar el valor del CNY Izquierdo
15 int ValorCNY_D; //Variable para guardar el valor del CNY Derecho
16
17 // ¿Cómo? --> usamos los datos (entrada o salida):
18 void setup(){
19   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
20   pinMode(MD1, OUTPUT); //Coloca a los pines de los motores como salida
21   pinMode(MD2, OUTPUT);
22   pinMode(MI1, OUTPUT);
23   pinMode(MI2, OUTPUT);
24 }
25

```

```

26 // ¿Cuándo?
27 void loop(){
28   ValorCNY_I = analogRead(A6); //Leemos el valor de los CNY's
29   ValorCNY_D = analogRead(A4);
30   //Sólo se utilizan 2 sensores, el izquierdo y el derecho. El robot se coloca
31   //sobre la línea, cuidando que la línea esté entre los sensores
32
33   lcd.setCursor(0,1); //Borramos datos del LCD
34   lcd.print("          ");
35   lcd.setCursor(0,0); //Mandamos a imprimir el valor del CNY Izquierdo
36   lcd.print("IZQ");
37   lcd.setCursor(0,1);
38   lcd.print(ValorCNY_I);
39   lcd.setCursor(12,0); //Mandamos a imprimir el valor del CNY derecho
40   lcd.print("DER");
41   lcd.setCursor(12,1);
42   lcd.print(ValorCNY_D);
43
44   if (ValorCNY_D < 650){ //El sensor derecho va a un costado de la linea
45     digitalWrite(MD1, HIGH); //El motor derecho avanza durante 200 miliseg
46     digitalWrite(MD2, LOW);
47     delay(200);
48     digitalWrite(MD1, LOW); //Se detiene 100 miliseg
49     delay(100);
50   }else{ //El sensor derecho entra a la linea
51     digitalWrite(MD1, LOW); //El motor derecho se detiene durante 200 miliseg
52     digitalWrite(MD2, LOW);
53     delay(200);
54   }
55   if (ValorCNY_I < 650){ //El sensor izquierdo va a un costado de la linea
56     digitalWrite(MI1, HIGH); //El motor izquierdo avanza durante 200 miliseg
57     digitalWrite(MI2, LOW);
58     delay(200);
59     digitalWrite(MI1, LOW); //Se detiene 100 miliseg
60     delay (100);
61   }else{ //El sensor izquierdo entra a la linea
62     digitalWrite(MI1, LOW); //El motor izquierdo se detiene durante 200 miliseg
63     digitalWrite(MI2, LOW);
64     delay(200);
65   }
66 }

```

# Programas para tareas específicas

## Evasor de obstáculos

```

1 ///////////////////////////////////////////////////
2 // Autor: Universidad del Papaloapan (UNPA)      //
3 // Programa: Evasor.- Programa para hacer que el robot avance de forma      //
4 //               autónoma evadiendo obstáculos al frente de él      //
5 ///////////////////////////////////////////////////
6
7 // ¿Qué? --> tipo de dato usamos:
8 #include<LiquidCrystal.h> //Librería para manejar al LCD
9 LiquidCrystal lcd (6,7,2,3,4,5); //Pines del LCD
10 int MD1 = 8; //Variable para el conector 1 del motor derecho: pin digital 8
11 int MD2 = 9; //Variable para el conector 2 del motor derecho: pin digital 9
12 int MI1 = 10; //Variable para el conector 1 del motor izquierdo: pin digital 10
13 int MI2 = 11; //Variable para el conector 2 del motor izquierdo: pin digital 11
14 long distancia; //Variable para guardar el valor de la distancia al objeto
15 long tiempo; //Variable para guardar el tiempo que tarda en rebotar el pulso
16
17 // ¿Cómo? --> usamos los datos (entrada o salida):
18 void setup(){
19   lcd.begin(16, 2); //Establece dimensiones del LCD, 16 caracteres y 2 filas
20   pinMode(MD1, OUTPUT); //Coloca a los pines de los motores como salida
21   pinMode(MD2, OUTPUT);
22   pinMode(MI1, OUTPUT);

```

```

23   pinMode(MI2, OUTPUT);
24   pinMode(13, OUTPUT); //Coloca al pin 13 como salida: genera pulso ultrasónico
25   pinMode(12, INPUT); //Coloca al pin 12 como entrada: detecta el tiempo de
26 }                                //rebote del pulso ultrasónico
27
28 // ¿Cuándo?
29 void loop(){
30   digitalWrite(13, LOW); //Por cuestiones de estabilidad del sensor
31   delayMicroseconds(5);
32   digitalWrite(13, HIGH); //Envío del pulso ultrasónico
33   delayMicroseconds(10);
34   tiempo=pulseIn(12, HIGH); //Mide el tiempo transcurrido entre el envío del
35                                //pulso ultrasónico y el rebote del mismo
36   distancia= int(0.017*tiempo); //Fórmula para calcular la distancia
37
38   lcd.setCursor(0,0); //Mandamos a imprimir la distancia calculada
39   lcd.print("Distancia=");
40   lcd.setCursor(10,0);
41   lcd.print(distancia);
42   lcd.print("cm ");
43
44 if(distancia < 35){ //La distancia del robot al obstáculo es < 35cm
45   digitalWrite(MD1, LOW); //El robot se detiene por 500 miliseg
46   digitalWrite(MD2, LOW);
47   digitalWrite(MI1, LOW);
48   digitalWrite(MI2, LOW);
49   delay(500);
50   digitalWrite(MD1, HIGH); //El robot gira a la derecha por 1 seg
51   digitalWrite(MD2, LOW);
52   digitalWrite(MI1, LOW);
53   digitalWrite(MI2, LOW);
54   delay(1000);
55   digitalWrite(MD1, LOW); //El robot se detiene por 500 miliseg
56   digitalWrite(MD2, LOW);
57   digitalWrite(MI1, LOW);
58   digitalWrite(MI2, LOW);
59   delay(500);
60 }else{ //La distancia del robot al obstáculo es > 35cm
61   digitalWrite(MD1, HIGH); //El robot avanza por 100 miliseg
62   digitalWrite(MD2, LOW);
63   digitalWrite(MI1, HIGH);
64   digitalWrite(MI2, LOW);
65   delay(100);
66 }
67 }

```

E S P A Ñ O L - M I X E

**Ja tsuj tsyo'ntajken JA  
NÄXWIINY  
ROBÓTICA**



Ejxy'ejtpë mitë nyajää'wëëp  
ti tunëp jäjtëp:

Jesús Santiguillo Salinas

Pukkë'pë-puxäjëpë:  
Hiram N. García Lozano  
Rafael F. González Zarate  
José L. Nájera Sánchez  
Luis A. Hernández Zuccolotto

Wunmää'ny Ajanyäx  
tanëpëktääjkëpë:  
Melisa del C. Muñoz Vázquez

Jakyukënunmää'ny:  
Vanessa N. Gómez Jiménez

Pukkë'pë-puxäjëpë:  
Leticia Gómez Jiménez  
Alfonso Díaz Vargas

E S P A Ñ O L - M I X E

Ja tsuj tsyo'ntajken JA  
NÄXWIINY  
ROBÓTICA

Este elibro surge como una propuesta de proyecto de la Universidad del Papaloapan dentro de la Convocatoria 2021 para la Elaboración de propuestas de Proyectos para el Fomento y Fortalecimiento de las Vocaciones Científicas del Consejo Nacional de Ciencia y Tecnología (CONACyT).

El objetivo de este elibro es acercar a los grupos vulnerables del país, en particular a los niños, niñas y jóvenes, al conocimiento científico y tecnológico de vanguardia, contribuyendo, por medio de la elaboración de material digital, interactivo y de acceso en línea sobre robótica, a fomentar y fortalecer las vocaciones científicas motivando la creatividad e innovación de esta población, con el fin de consolidar una cultura del conocimiento.



ISBN: 978-607-9320-50-8

9 786079 320508