



Trabajo de Grado

Seminario De Robótica

Corporación Universitaria Remington.

Facultad de Ingeniería.

Ingeniería de Sistemas.

Federico Lisalda Muñoz

Duberney Sarchi Velez

Tutor: Jonathan Stick Campos Núñez.

Opción de Trabajo de grado Seminario-Diplomado

2024



## Contenido

Resumen .....	5
Palabras Claves .....	5
Glosario .....	6
Objetivo General .....	8
Objetivos Específicos .....	8
¿Qué es Arduino?.....	9
La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales. .....	9
La robótica educativa. ....	9
¿Por qué utilizar Arduino?.....	9
¿Qué se puede hacer con Arduino? .....	10
¿Solo existe un tipo de Arduino? .....	10
Protoboard.....	12
Importancia de la robótica .....	12
Importancia de la robótica en las ramas del conocimiento. ....	13
Aporte de la robótica a la enseñanza en la modernidad .....	13
Tres leyes de la robótica de Isaac Asimov .....	14
Y ¿qué es un robot? .....	14
Clasificación de la robótica según su actividad y propósito.....	15
Tipos de robots por cronología: .....	15
Primera Generación: .....	15
Segunda Generación: .....	15
Tercera generación: .....	15
Cuarta Generación:.....	15
Quinta Generación:.....	15
Tipos de robots según movilidad: .....	15
Robots articulados o brazos robóticos: .....	15
Vehículo de guiado automático (AGV): .....	15
Robots móviles autónomos: .....	15
Humanoides: .....	15
Cobots: .....	16
Zoomórficos: .....	16
Androides:.....	16



Poliarticulados: .....	16
Robótica espacial: .....	16
Tipos de robots por función o sector:.....	16
Robots industriales: .....	16
Robots domésticos: .....	16
Robots educativos: .....	17
Robots militares: .....	17
Robots médicos:.....	17
PWM.....	17
Marco teórico .....	17
Ejercicio práctico en Arduino: .....	17
Configurar grados de un servomotor: .....	19
Acercamiento a la araña robótica: .....	23
Montaje y conexiones de la araña robótica: .....	26
Programación: .....	30
Conclusiones .....	37
Referencias bibliográficas. ....	38



## INDICE DE IMÁGENES

Imagen 1 Modelos de Arduino .....	11
Imagen 2 Protoboard .....	12
Imagen 3 Código para encender un led .....	17
Imagen 4 Demostración del led encendido.....	18
Imagen 5 Demostración de led apagado.....	19
Imagen 6 Código de configuración de giro en grados para el servomotor.....	20
Imagen 7 Giro del servomotor a 30° .....	21
Imagen 8 Giro del servomotor a 120° .....	22
Imagen 9 Araña robótica .....	23
Imagen 10 Diagrama de conexiones del robot arácnido.....	24
Imagen 11 Denominación de los servomotores.....	25
Imagen 12 Identificación de código de colores de los servomotores .....	26
Imagen 13 Conexión de servomotores al arduino .....	27
Imagen 14 Conexión del Arduino a la protoboard .....	28
Imagen 15 Conexión de servomotores en la parte de control.....	29
Imagen 16 Ilustraciones de conexiones siguiendo código de colores .....	30
Imagen 17 Demostración de la araña robótica no funcional .....	34
Imagen 18 Araña robótica en proceso de levantamiento.....	35
Imagen 19 Araña robótica puesta en pie .....	36



## Resumen

La robótica es una interdisciplina de la ciencia de las que se encarga la ingeniería, incluye la mecánica, la electrónica, la informática y la inteligencia artificial, así como el diseño, construcción, programación y operación de robots. Son máquinas capaces de realizar tareas de manera autónoma o semiautónoma con el fin de automatizar y/o agilizar procesos, siempre mediante la búsqueda de la mejora y el progreso.

En este proyecto como opción de grado, se desea demostrar mediante un sencillo ejemplo la utilidad e importancia de la robótica en la actualidad.

## Palabras Claves

Arduino, Servomotor, Robótica, Sensores y programación.



## Glosario

**Robótica:** Es la rama de la ingeniería y la ciencia dedicada al diseño, construcción, operación y uso de robots. (ChatGPT, 2024).

**Microcontrolador:** Es un circuito integrado que contiene un procesador, memoria y periféricos de entrada/salida en un solo chip, utilizado para controlar dispositivos electrónicos. (ChatGPT, 2024).

**Servomotor:** Es un dispositivo que combina un motor eléctrico con un sensor de posición y un sistema de control. Este conjunto permite controlar con precisión la posición, velocidad y aceleración del motor. (ChatGPT, 2024).

**Arduino:** Plataforma de hardware y software libre utilizada para la creación de proyectos electrónicos interactivos. (ChatGPT, 2024).

**Arduino Uno:** Es una de las placas de desarrollo más populares de la plataforma Arduino. Está basada en el microcontrolador ATmega328P y cuenta con 14 pines de entrada/salida digital, 6 entradas analógicas, una conexión USB, un conector de alimentación, y un botón de reinicio. (ChatGPT, 2024).

**Entradas:** Son señales o datos que el microcontrolador recibe desde el exterior a través de sus pines, permitiéndole interactuar con sensores, botones u otros dispositivos. (ChatGPT, 2024).

**Salidas:** Son señales o datos que el microcontrolador envía hacia el exterior a través de sus pines, permitiéndole controlar actuadores, luces, motores u otros dispositivos. (ChatGPT, 2024).

**Sensores:** Son dispositivos que detectan cambios físicos o ambientales (como temperatura, luz, presión, humedad, etc.) y convierten esta información en señales eléctricas para que puedan ser procesadas por un sistema electrónico. (ChatGPT, 2024).

**Electrónica:** Rama de la ingeniería y la física que estudia y aplica el comportamiento y control de los electrones en dispositivos y sistemas electrónicos. (ChatGPT, 2024).

**PWM:** (Modulación por Ancho de Pulso) es una técnica utilizada para controlar la cantidad de energía entregada a un dispositivo electrónico. (ChatGPT, 2024).

**void setup():** Es una función que se ejecuta una vez al inicio del programa (arduino) para configurar los pines y la comunicación antes de entrar en el bucle principal. (ChatGPT, 2024).

**void loop():** Es una función que se ejecuta repetidamente y contiene el código principal que el Arduino debe ejecutar continuamente. (ChatGPT, 2024).



**pinMode:** Código que se utiliza en arduino para definir que un pin de la placa se utilizará de entrada o de salida. (ChatGPT, 2024).

**digitalWrite:** Código que permite generar una salida en un pin determinado de la placa. (ChatGPT, 2024).

**delay:** Código que permite generar una pausa de voltaje en un determinado pin de la placa. (ChatGPT, 2024).

**High:** Significa un estado de voltaje alto (generalmente 5V) en un pin digital, utilizado para encender componentes como leds o activar dispositivos. (ChatGPT, 2024).

**Low:** Significa un estado de voltaje bajo (0V) en un pin digital, utilizado para apagar componentes o desactivar dispositivos. (ChatGPT, 2024).



### Objetivo General

Desarrollar un robot “araña” funcional y autónoma para el proyecto final en el seminario de robótica como opción de grado.

### Objetivos Específicos

Conocer los elementos necesarios para llevar a cabo el desarrollo del robot y sus configuraciones básicas.

Realizar proyectos de prueba sencillos para comprender cómo funciona el Arduino y sus componentes.

Ejecutar el montaje del robot y toda su programación, así como llevar a cabo pruebas de ensayo y error para encontrar posibles fallas y corregirlas a tiempo.

Lograr que el robot trabaje de manera autónoma cumpliendo con el objetivo principal del proyecto que es que la araña se ponga en pie de manera correcta.

Documentar todos los temas vistos en el seminario, al igual que las actividades realizadas con sus respectivas explicaciones durante el desarrollo de este.



## ¿Qué es Arduino?

Arduino es una plataforma de hardware de código abierto, basada en una placa sencilla que contiene un microcontrolador, esta placa conecta el mundo físico con el mundo virtual, o bien el mundo analógico con el digital, además de que controla sensores, alarmas, sistemas de comunicación y actuadores físicos los cuales funcionan de acuerdo con la programación que se crea en el entorno de desarrollo (IDE) de Arduino.

El software de Arduino está basado en una licencia libre para que se descargue sin problema y los usuarios comiencen a adquirir habilidades de programación con el uso de las principales instrucciones y funciones. (Pedrera, 2017).

## La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales.

La creciente importancia que tiene la tecnología en el mundo hoy en día y su continuo desarrollo, hace que la tecnología, en sí misma, se convierte en parte integral del proceso de formación en la niñez y la juventud. Por esta razón es importante desarrollar propuestas en las que se ofrezca a niños y jóvenes la posibilidad de entrar en contacto con las nuevas tecnologías; esto es posible a través del manejo de herramientas de software y hardware, como prototipos robóticos y programas. (Bravo Sánchez, 2012).

## La robótica educativa.

La robótica se ha ido introduciendo poco a poco en la educación, dotando a los alumnos de herramientas tecnológicas propias de este siglo XXI. Por medio de la robótica los alumnos pueden experimentar con robots, construirlos, programarlos, resolver problemas, etc. Esto siempre en un contexto de trabajo en equipo que potencia la cooperación y la solución de problemas. (Pinto-Salamanca, 2010).

## ¿Por qué utilizar Arduino?

Arduino fue creado con el propósito de ser una plataforma extremadamente fácil de usar en comparación con otras, lo que la hace ideal para los desarrolladores más



experimentados como para principiantes ya que ahora sus proyectos se pueden realizar mucho más rápido y son menos laboriosos; más aún cuando nos referimos a estudiantes, personas que apenas se están aventurando en temas de programación. Otro factor que hace que Arduino sea muy atractivo es su filosofía de software libre, es decir, la gente puede utilizarlo para crear varios proyectos sin coste alguno por los derechos de utilización de la plataforma y se puede distribuir de forma gratuita, si así lo desean, lo que permite aprender de todo tipo de proyectos tanto para aficionados como para quienes se encuentran estudiando temas afines a la ingeniería. Esto trae muchos beneficios; además de crear y distribuir varias bibliotecas nuevas y herramientas para ayudar al desarrollo de los proyectos todos los días, cuenta con una comunidad con miles de personas que revelan información y detalles acerca de lo que se crea y aportan documentación, tutoriales y repositorios sobre el funcionamiento de Arduino. Estas son también algunas razones por las que la popularidad de Arduino está creciendo entre los desarrolladores.

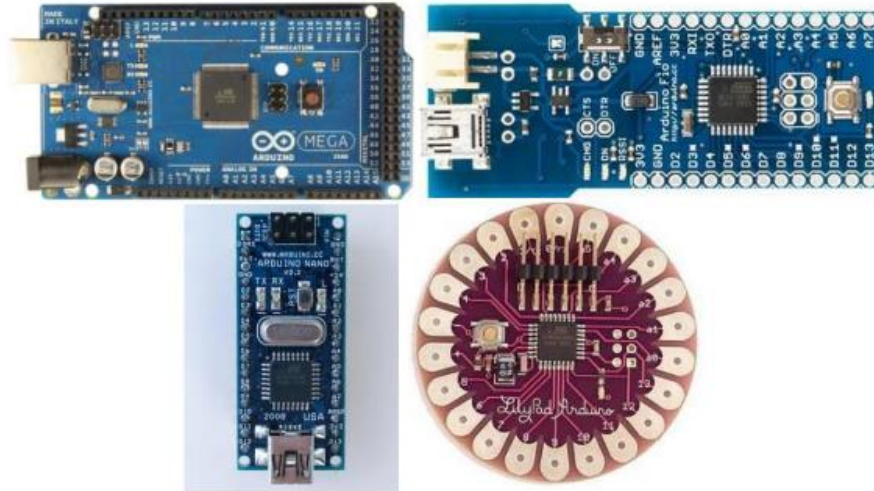
Como recomendación personal invitamos a todos quienes lean este documento y estén interesados en aprender algo de electrónica o robótica, realizar sus proyectos en Arduino, es una herramienta fácil de manejar y agradable, perfecta para empezar con estos temas. Además, pueden realizar un sinnúmero de simulaciones sin tener que invertir dinero en componentes ni elementos, pero podrán adquirir bastantes conocimientos.

### ¿Qué se puede hacer con Arduino?

¡Casi cualquier cosa! Basta con tener el equipo necesario y podrá crear todos los diseños que tan solo limitados por su imaginación o por las leyes de la física, por supuesto.

### ¿Solo existe un tipo de Arduino?

No. Hay muchos tipos de Arduino que puede utilizar dependiendo de lo que quiera hacer, con diferentes formas y configuraciones de hardware. El Arduino Uno es el más utilizado por el Mega Arduino, por ejemplo, tiene más puertos de entrada posibilitando la creación de dispositivos más grandes y complejos. El Arduino Nano, como el nombre lo dice es una versión abreviada de un arduino común, para la creación de objetos de electrónica más pequeña. A continuación, puede ver algunas fotos con algunos diversos tipos de Arduino que existen hoy en día.



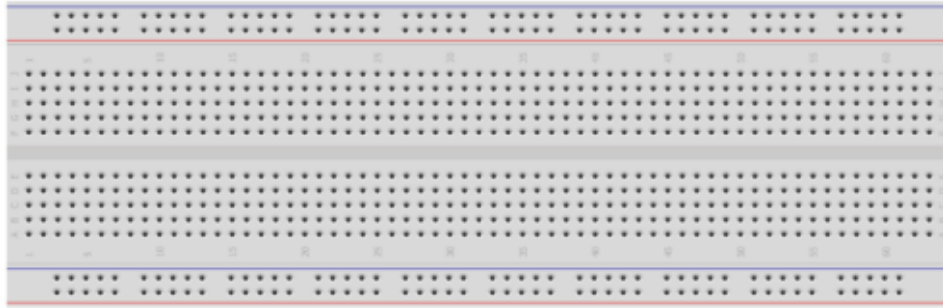
1

*Imagen 1 Modelos de Arduino*

Respectivamente, estos modelos son Arduinos Mega, Arduino Uno, Arduino Nano y Arduino LilyPad.

Cada uno tiene una funcionalidad diferente que justifica su creación. El LilyPad, por ejemplo, está diseñado para ser capaz de ser utilizado en la ropa, se puede coser directamente sobre los tejidos. Para obtener más información sobre los distintos modelos de la Arduino, puede consultar el sitio web oficial: [www.arduino.cc](http://www.arduino.cc).

## Protoboard



*Imagen 2 Protoboard*

Para conectar todos estos componentes con el fin de tener un circuito funcional, tenemos que unir (puentear) sus entradas y salidas de alguna manera, de modo que la corriente que pasa a través de una pieza continúe propagándose a la próxima. Por lo general, lo hacemos a través de un proceso de soldadura; es decir, calentamos un metal maleable y conductor para que se derrita y una de las dos partes que lo utilizan como una especie de pegamento. El acto de soldar es una acción delicada ya que requiere práctica y habilidad, tanto para obtener resultados satisfactorios como para evitar quemaduras, que a menudo pueden ser fatales (una máquina de soldadura puede causar quemaduras de tercer grado). Además del peligro obvio, el proceso de soldar es en cierto modo “permanente” porque las piezas unidas son difíciles de retirar, algo que puede ser muy laborioso si hay una gran cantidad de piezas soldadas. (Pedrera, 2017).

Una protoboard consta de una zona de alimentación horizontal, donde la línea azul corresponde al riel con carga negativa y el riel de color rojo corresponde a la carga positiva, y una zona de alimentación vertical, donde se realizan las conexiones necesarias.

### Importancia de la robótica

Para el siglo XXI, la robótica hoy en día juega un papel fundamental y esencial en el desarrollo tecnológico debido a que utiliza múltiples áreas para poder construirse y ejecutarse, por ello, uno de los grandes retos es en el ámbito educativo, debido a que la enseñanza ha ido cambiando con el pasar del tiempo, y las instituciones educativas deben estar a la par de dicha evolución por lo tanto, es primordial el fomentar en los estudiantes ciertas habilidades y/o competencias que les permitan ser personas activas, flexibles, creativas y trabajen en equipo, y sean capaces de aportar soluciones innovadoras a los problemas que se les presente de manera cotidiana.



El uso de las tecnologías en la educación permite acceder a contenidos educativos, así como a mejorar el planteamiento para el desarrollo de habilidades y competencias que requieren los alumnos, así que la robótica educativa es un recurso eficaz para el trabajo interdisciplinario y enriquecer los procesos de enseñanza y aprendizaje. (López, 2019)

De ahí, que la robótica contenga gran importancia en las ramas del conocimiento, pues en temas de tecnología es esencial para resolver problemas cotidianos en el ámbito laboral; más aún si tenemos en cuenta el auge que tiene la inteligencia de las cosas y la inteligencia artificial en la actualidad. Teniendo en cuenta que la tecnología tiende a un progreso evolutivo a futuro, por ende, la robótica.

### Importancia de la robótica en las ramas del conocimiento.

La robótica educativa es parte de las STEM<sup>1</sup> (Ciencia, Tecnología, Ingeniería y Matemáticas) las cuales involucran un proceso con mayor innovación que da paso a la solución de problemas de la vida real. (Ortiz, 2019).

### Aporte de la robótica a la enseñanza en la modernidad

Hoy en día la robótica con un enfoque pedagógico se convierte en una estrategia de enseñanza para diferentes áreas como lo son: las matemáticas, ciencias e informática; con ello, se genera un ambiente de aprendizaje donde el estudiante desempeña un papel fundamental y el trabajo colaborativo se ve fortalecido al diseñar y simular situaciones, por medio de la construcción de robots.

De esta forma, los estudiantes adquieren conocimientos, actitudes, valores y habilidades, obteniendo un pensamiento lógico y formal para el logro de competencias. (García, 2017).

Por ello, es de gran valor que el estudiante realice estudios autónomos y autodidactas para que desarrolle habilidades de manera significativa y se desempeñe con méritos en el área de la programación de la robótica.

Dentro de los principales beneficios de la robótica en la educación se encuentran los siguiente:

- Motiva la curiosidad científica y tecnológica.
- Participación y colaborativa.
- Uso de tecnología como una solución a problemas.
- Hace uso de la innovación para la solución de fallas.

La enseñanza de la robótica en algunos currículos hace uso de kits como Lego Mindstorm EV3, sin embargo, para la creación de proyectos requieren el uso y conocimiento de otras

---

<sup>1</sup> Ciencia, Tecnología, Ingeniería y Matemáticas.



áreas escolares como: mecatrónica, electrónica y computación, es por ello, que se debe crear una sincronía para generar un conocimiento unificado para diseñar un sistema global.

Con el uso de este tipo de herramientas como la construcción de un pequeño robot o mecanismo el cual puede ser manipulado mediante un software, permite que los estudiantes aprendan, utilizando el método de ensayo y error, a programar y a conseguir que realice tareas sencillas. De esa manera, se puede trabajar mediante proyectos lo que permite desarrollar capacidades que en el modelo laboral actual y futuro serán necesarias, por lo que, en la robótica educativa este tipo de metodología de aprendizaje apoya a experimentar con todos los campos que pueden relacionarse con la tecnología y la ciencia, fomentando un espíritu emprendedor donde el alumno es el protagonista de su aprendizaje. (Robotix, 2019).

Cuando los jóvenes comienzan a programar, también inician con el desarrollo de su pensamiento computacional, según Carlos Casado, profesor de los Estudios de Informática, Multimedia y Telecomunicación de la UOC, la programación y la robótica en general son una herramienta que ayuda a trabajar a la resolución de problemas de cualquier tipo, obligando a los estudiantes a imaginar cual podría ser el resultado antes de empezar y así trabajar un pensamiento más abstracto para decidir cuál es la mejor opción, y posteriormente realizarlo en la computadora. (Casado, 2016).

### Tres leyes de la robótica de Isaac Asimov

1. Un robot no puede dañar a un ser humano o, por inacción, permitir que un ser humano sufra daños.
2. Un robot debe obedecer las órdenes que le den los seres humanos, excepto cuando tales órdenes entren en conflicto con la Primera Ley.
3. Un robot debe proteger su propia existencia siempre que dicha protección no entre en conflicto con la Primera o Segunda Ley. (Asimov, 1950).

### Y ¿qué es un robot?

Es una máquina programable que posee cierto grado de inteligencia y es capaz de ejecutar tareas de manera automática en función de las decisiones, basándose en la estructura de su programa. (Hernández, 2017).



## Clasificación de la robótica según su actividad y propósito

### Tipos de robots por cronología:

En este caso, se distinguen hasta cinco tipos de robots, según las etapas por las que ha ido pasando la robótica hasta el momento actual.

#### Primera Generación: robots manipuladores.

Son aquellos que pueden coger y mover objetos, pero tienen unos movimientos muy limitados.

#### Segunda Generación: robots en aprendizaje.

Recogen la información del entorno para poder hacer movimientos más complejos.

#### Tercera generación: robots reprogramables.

Son aquellos equipados con sensores y en los que se usan lenguajes de programación para variar sus funciones según las necesidades en cada momento.

#### Cuarta Generación: robots móviles.

En la cuarta generación aparecen los primeros robots inteligentes, capaces de interpretar el entorno en tiempo real.

#### Quinta Generación: robots con inteligencia artificial.

Es la etapa que se encuentra actualmente en desarrollo. Pretende imitar al ser humano y son autónomos.

### Tipos de robots según movilidad:

Otra clasificación muy interesante de los robots tiene mucho que ver con su movilidad. Según su desempeño, su capacidad de movimiento y tomar de decisiones se pueden diferenciar de este modo.

**Robots articulados o brazos robóticos:** Tienen una capacidad muy reducida; pero son muy buenos aliados para mover productos, manipular herramientas, empaquetar (mano de obra) ...

**Vehículo de guiado automático (AGV):** Este se mueven por una pista predefinida y normalmente necesitan la supervisión de un humano.

**Robots móviles autónomos:** Los llamados AMR se pueden mover y tomar decisiones por sí mismos, prácticamente en tiempo real. Incorporan sensores y un equipo de procesamiento a bordo para llevar a cabo sus funciones.

**Humanoides:** Normalmente son un tipo de AMR, con formas humanas y con funciones similares a de las personas.



**Cobots:** Son aquellos que están diseñados con el propósito de trabajar codo con codo con los humanos, ayudando con tareas peligrosas o repetitivas.

**Zoomórficos:** Dentro de este grupo se consideran a los andróides, ya que se subdivide en dos categorías caminadores y no caminadores.

**Híbridos:** Robots de difícil clasificación cuya estructura se sitúa en la combinación con alguna de las anteriores, bien por su Conjunción o Yuxtaposición.

**Andróides:** Son los robots que intentan reproducir total y/o parcialmente la forma y el comportamiento cinemático del ser humano.

**Poliarticulados:** Son los robots de diversas formas y configuraciones, cuyas características son básicamente sedentarios y están estructurados para mover sus elementos terminales en un determinado espacio de trabajo.

**Robótica espacial:** Son los robots diseñados para la actividad espacial como la exploración planetaria o la asistencia en órbita.

#### Tipos de robots por función o sector:

La segunda clasificación empleada para los robots atiende a sus funciones y el entorno en el que se desarrollan, es decir, el sector para el que han sido concebidos (como la industria, la ganadería, la educación, la sanidad, la logística...).

De este modo, se pueden encontrar robots militares, industriales, de servicios, educativos, de investigación, médicos o domésticos, principalmente, aunque la lista podría ser tan extensa como posibilidades de uso existen. Cada uno de ellos tiene una funcionalidad concreta. Por lo que se sobreentiende la importancia de los robots en el mundo actual, siendo capaces de realizar actividades de manera parcial o totalmente autónoma, agilizando y automatizando procesos en diferentes áreas de la industria.

Cabe recalcar que los procesos tecnológicos tienden a ser más demandados con el paso de los años; por lo cual toma relevancia el estudio de la robótica.

A continuación, se resumen las principales.

**Robots industriales:** Estos tipos de robots tienen un claro enfoque en la cadena de producción y realizan actividades rutinarias y repetitivas. Por ejemplo, pueden encargarse de categorizar productos dentro de un almacén o de participar en una cadena de montaje moviendo productos de un lado a otro.

**Robots domésticos:** Son aquellos que ayudan con las tareas de limpieza y vigilancia de la casa. En este grupo se pueden mencionar los robots aspiradores de limpieza, los robots cortacésped, los robots de cocina que preparan la receta de principio a fin o las cámaras de seguridad conectadas.



**Robots educativos:** En este grupo se pueden incluir aquellas máquinas destinadas al desarrollo cognitivo o al aprendizaje de una materia. Por ejemplo, los kits de robótica para niños.

**Robots militares:** Por su parte, los robots militares se encargan de actuar como apoyo de los ejércitos en ciertas operaciones, como transportar material o ayudar a detectar la presencia de explosivos.

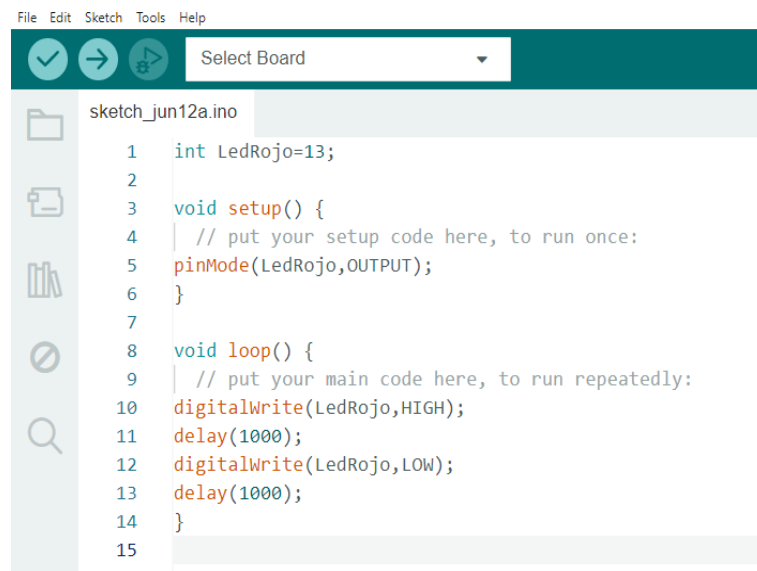
**Robots médicos:** Pueden servir de apoyo en el sector sanitario, por ejemplo, para ayudar a personas con movilidad reducida, para trasladar maquinaria o medicinas, incluso para participar en intervenciones quirúrgicas. (EcuRed, 2020).

## PWM

Pueden construirse diversos generadores de señales tanto con dispositivos analógicos como digitales. Con cualquier dispositivo analógico el procedimiento es el mismo, se trata de generar una señal que oscila alrededor de un voltaje con amplitud y frecuencia controlada que puede ser tratada para obtener señales de tipo senoidal, diente de sierra y cuadradas. Los amplificadores operacionales pueden tratar esta oscilación al comparar valores de voltaje de esta señal. Pueden realizarse más operaciones como la integración y la comparación para producir las señales típicas. (Ortiz E. A.).

## Marco teórico

Ejercicio práctico en Arduino:



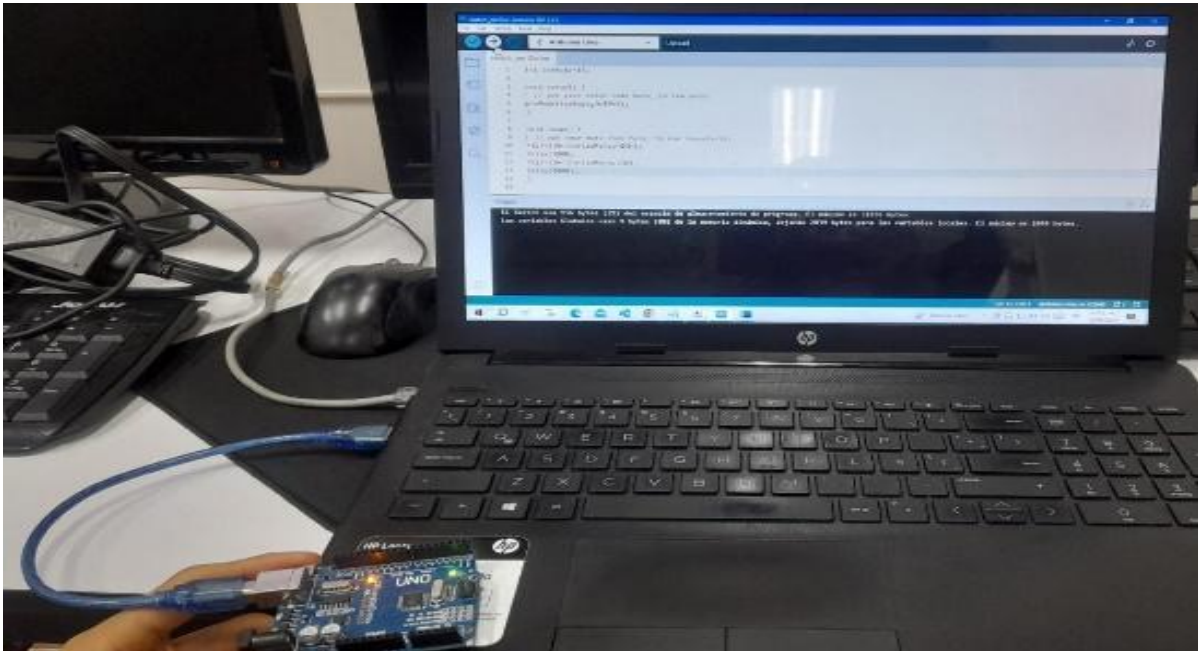
```
File Edit Sketch Tools Help
[Icons] Select Board
sketch_jun12a.ino
1 int LedRojo=13;
2
3 void setup() {
4   // put your setup code here, to run once:
5   pinMode(LedRojo,OUTPUT);
6 }
7
8 void loop() {
9   // put your main code here, to run repeatedly:
10  digitalWrite(LedRojo,HIGH);
11  delay(1000);
12  digitalWrite(LedRojo,LOW);
13  delay(1000);
14 }
15
```

*Imagen 3 Código para encender un led*

En este sencillo ejercicio aprendimos a programar Arduino para encender un Led, que es el proceso más básico que existe en dicha plataforma. En pocas palabras el código

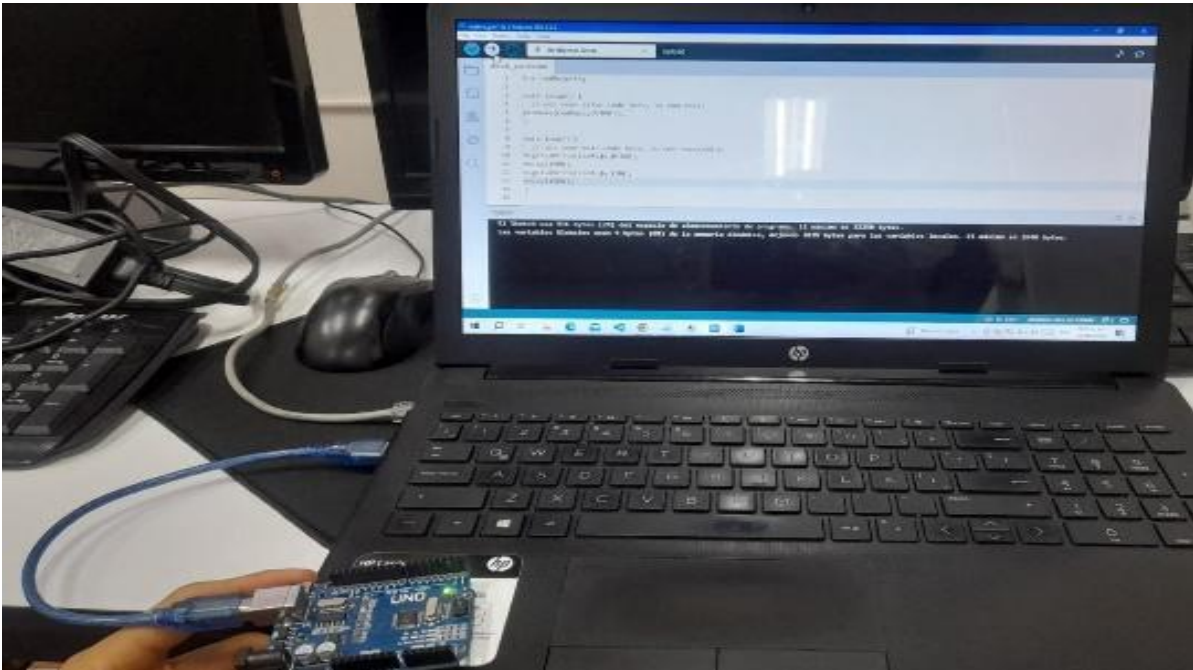
que hemos generado cumple la función de encender un led en el pin #13 de la placa durante un segundo, luego de esto procede a apagarlo por un segundo; luego de este segundo de “pausa”, lo enciende de nuevo y se repite el ciclo de forma infinita.

Encender un led en Arduino es el equivalente a imprimir en pantalla un “Hola mundo” que es lo más básico y sencillo en temas de programación.



*Imagen 4 Demostración del led encendido*

En la anterior imagen logramos comprobar que el led del pin #13 de la placa se enciende.



*Imagen 5 Demostración de led apagado*

En la anterior imagen evidenciamos que el led del pin #13 de la placa se apaga. De esta manera podemos observar que el ejercicio se ha realizado de manera correcta.

#### Configurar grados de un servomotor:

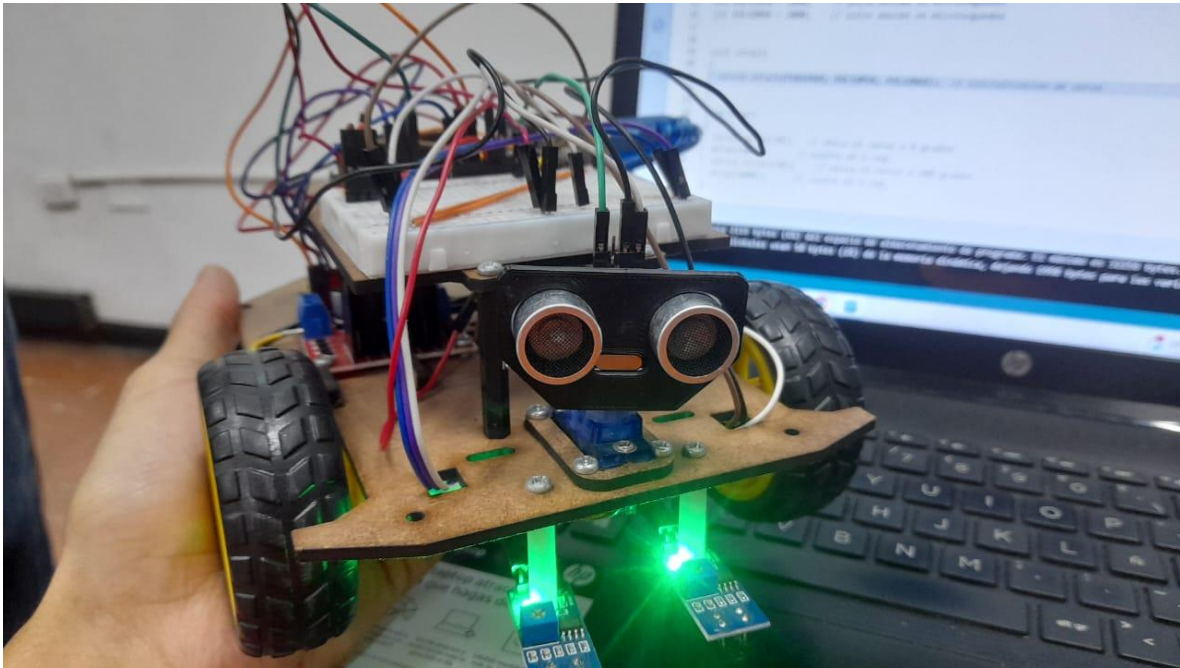
En este momento tenemos un robot, el cual cuenta con un servo motor, que a su vez se encuentra atornillado a un sensor de proximidad. El objetivo es lograr que el servo motor gire, de manera que el sensor rote a las posiciones (grados °) establecidas en el código. Para esto es importante programar la placa de Arduino, para que dicho sensor tenga movimiento. A continuación, se comparte el código encargado de generar dicha movilidad.



```
File Edit Sketch Tools Help
Arduino Uno
sketch_jun13a.ino
7
8 */
9
10 #include <Servo.h> // incluye librería de Servo
11
12 Servo servo1; // crea objeto
13
14 int PINSERVO = 11; // pin 11 conectado a señal del servo
15 int PULSOMIN = 1000; // pulso mínimo en microsegundos
16 int PULSMAX = 2000; // pulso máximo en microsegundos
17
18
19 void setup()
20 {
21   servo1.attach(PINSERVO, PULSOMIN, PULSMAX); // inicialización de servo
22 }
23
24 void loop()
25 {
26   servo1.write(30); // ubica el servo a 0 grados
27   delay(5000); // espera de 5 seg.
28   servo1.write(120); // ubica el servo a 180 grados
29   delay(5000); // espera de 5 seg.
30 }
31
Output
El Sketch usa 2116 bytes (6%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
```

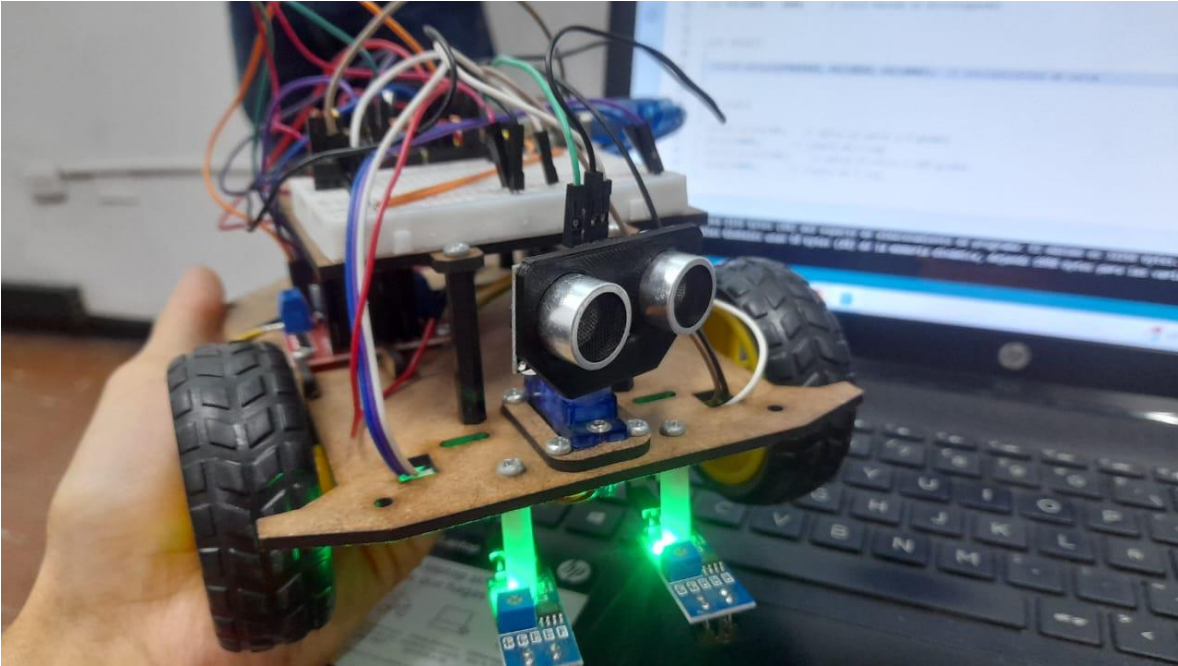
*Imagen 6 Código de configuración de giro en grados para el servomotor*

Configuramos el servomotor para que realice un giro de entre 30° a 120°.



*Imagen 7 Giro del servomotor a 30°*

En la imagen anterior podemos observar como el robot se ha configurado, para que el servo motor gire ubicándose en 30°.

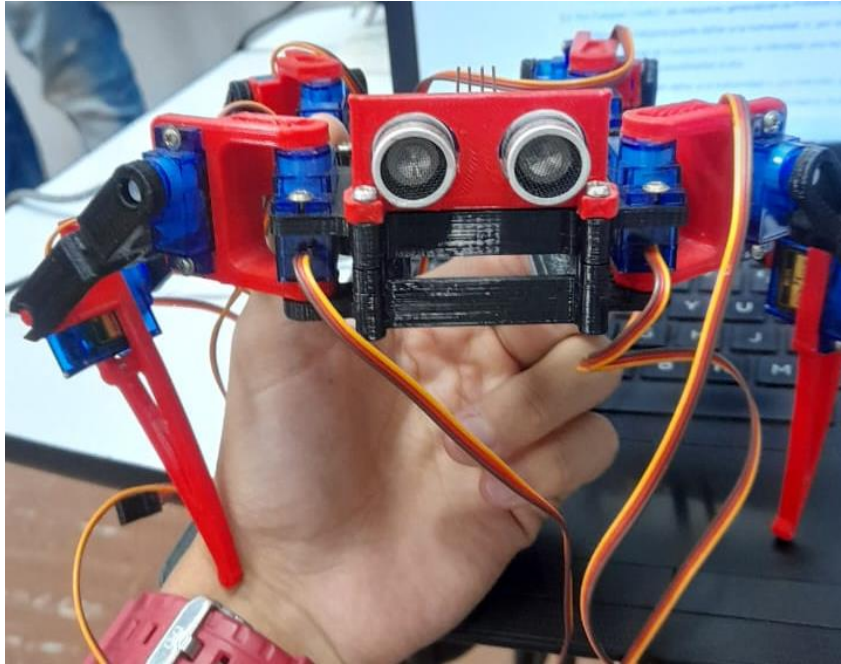


*Imagen 8 Giro del servomotor a 120°*

En la imagen anterior se observa que el robot es capaz de girar el servo motor a sus 120°.

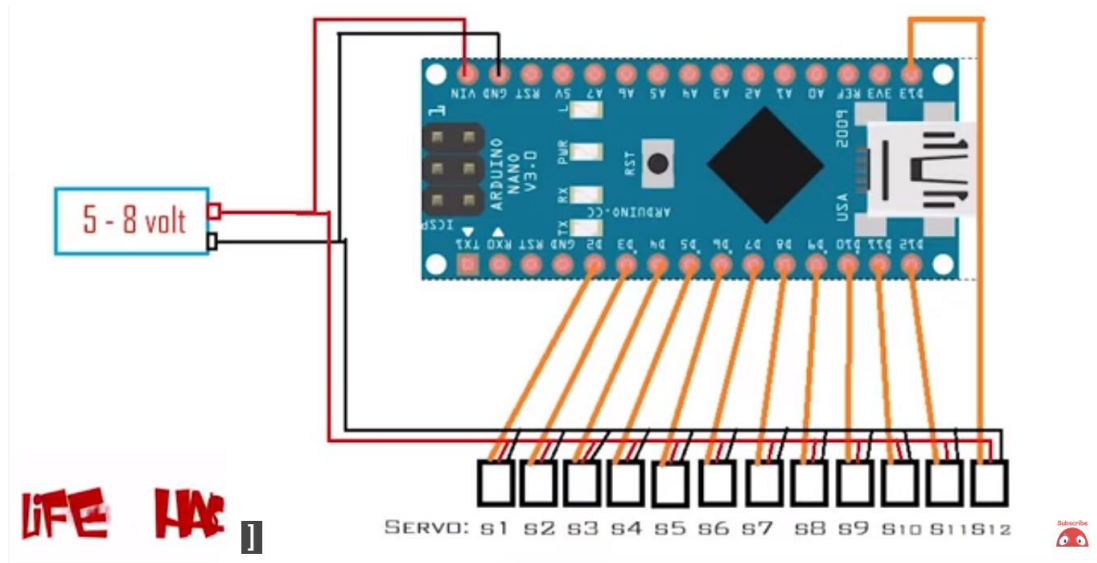
Hemos cumplido con el propósito. Una vez que hemos comprendido los conceptos básicos de Arduino y las configuraciones del servo motor estamos listos para aventurarnos con la idea final del proyecto que consta de hacer que la araña robótica evada de forma correcta los obstáculos y se desplace sin interferencias.

Acercamiento a la araña robótica:



*Imagen 9 Araña robótica*

En esta imagen podemos visualizar la araña robótica con la que vamos a trabajar, igualmente se pueden ver sus componentes tales como, servomotores, sensores, cables y su estructura física. A continuación, se procede a realizar el ensamble del robot mencionado anteriormente.



*Imagen 10 Diagrama de conexiones del robot arácnido*

(Lifehacker, 2018).

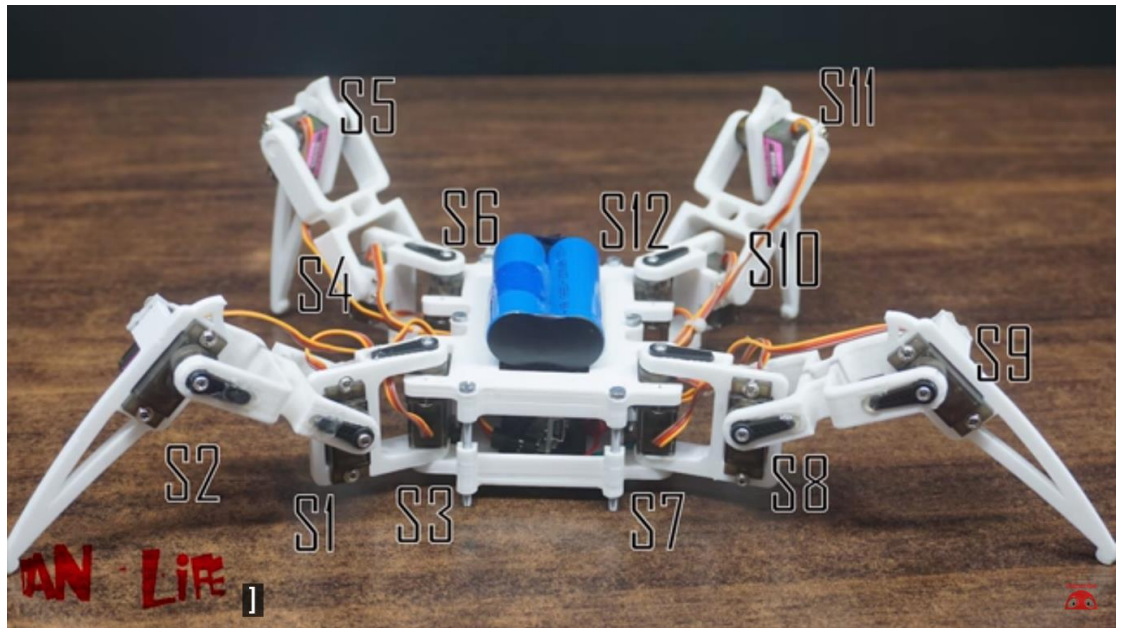
En esta figura, se muestra la conexión de potencia y control de los servomotores que utilizaremos para llevar a cabo el proyecto deseado; clasificándolos de la siguiente manera:

Fuente de alimentación: cable rojo.

GND o Negativo: cable negro.

Cable de control: cable naranja.

S1: servomotor #1, S2: servomotor #2... S12: servomotor #12.

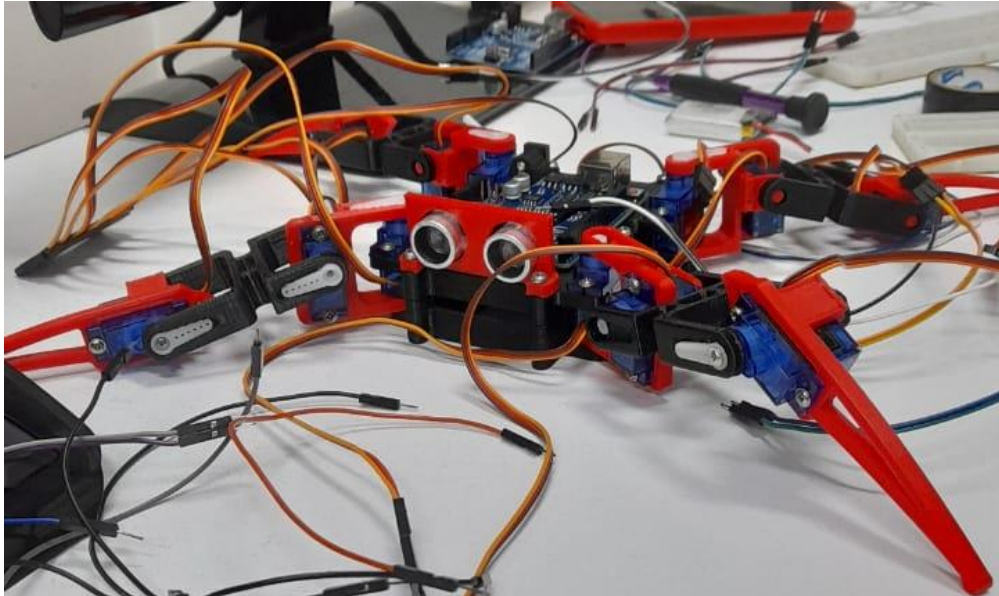


*Imagen 11 Denominación de los servomotores*

(Lifehacker, 2018).

En este diagrama se observa la denominación que le daremos a cada servomotor, en relación con la figura 1.

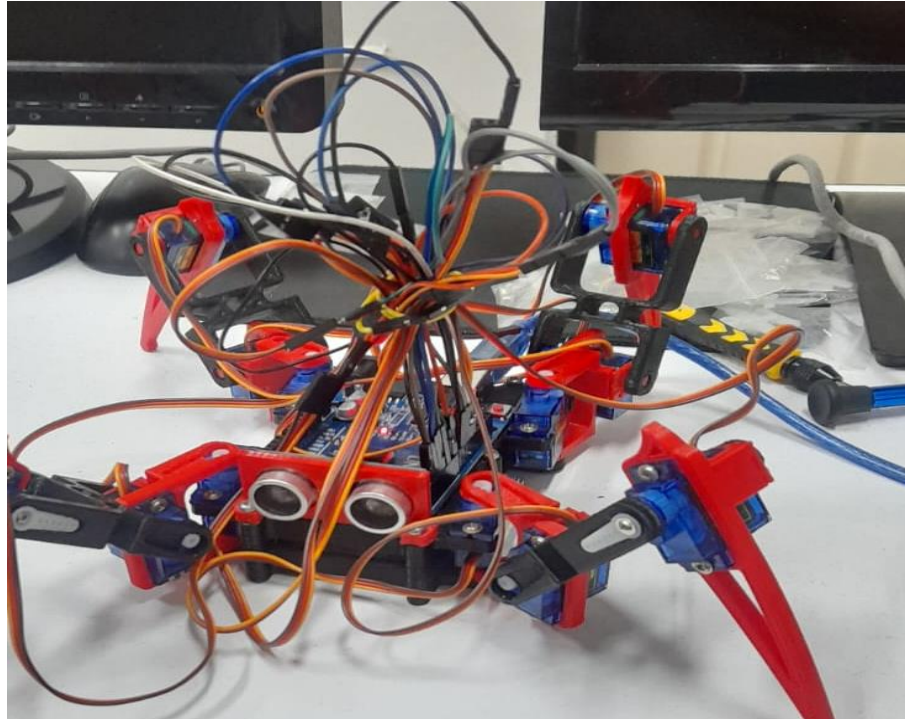
Montaje y conexiones de la araña robótica:



*Imagen 12 Identificación de código de colores de los servomotores*

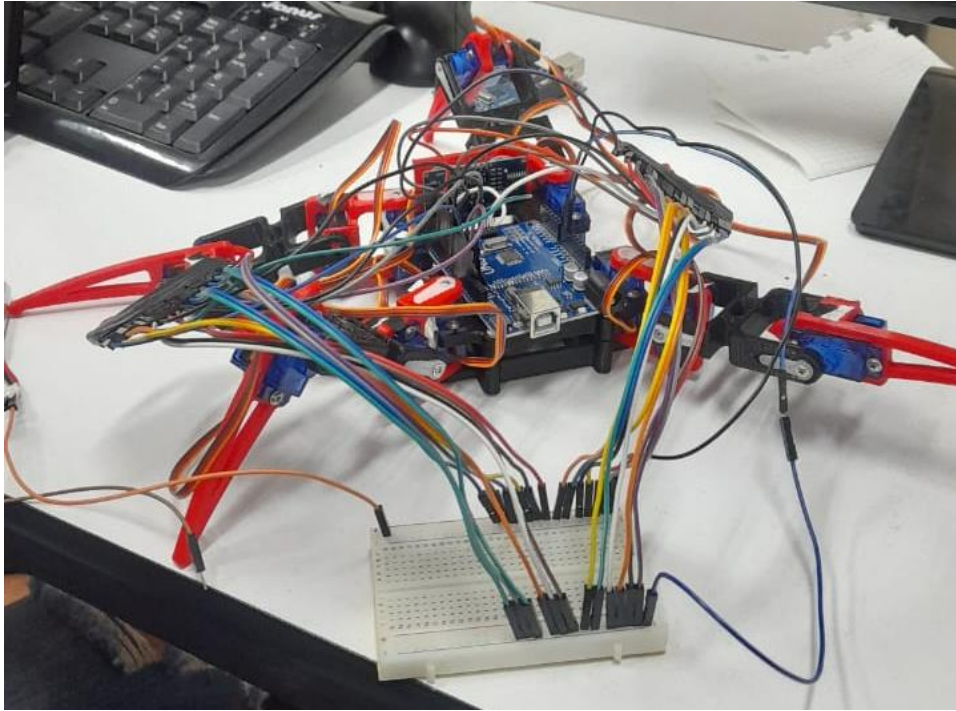
Hasta ahora nuestra araña robótica no es capaz de cumplir ninguna función por sí sola, para ello, debemos de realizar las conexiones necesarias para que sea capaz de levantarse en sus 4 extremidades y posteriormente pueda caminar libremente.

Lo primero que haremos es identificar el código de colores que trae cada servomotor, para así establecer los cables que irán al positivo, señal-control y el ground.



*Imagen 13 Conexión de servomotores al arduino*

Hemos conectado el cableado de potencia y control de manera provisional, realizando pruebas para observar el comportamiento del robot y sus posibles fallas, para realizar las respectivas correcciones. En este momento el robot se ha energizado desde el computador directamente a la tarjeta de Arduino.

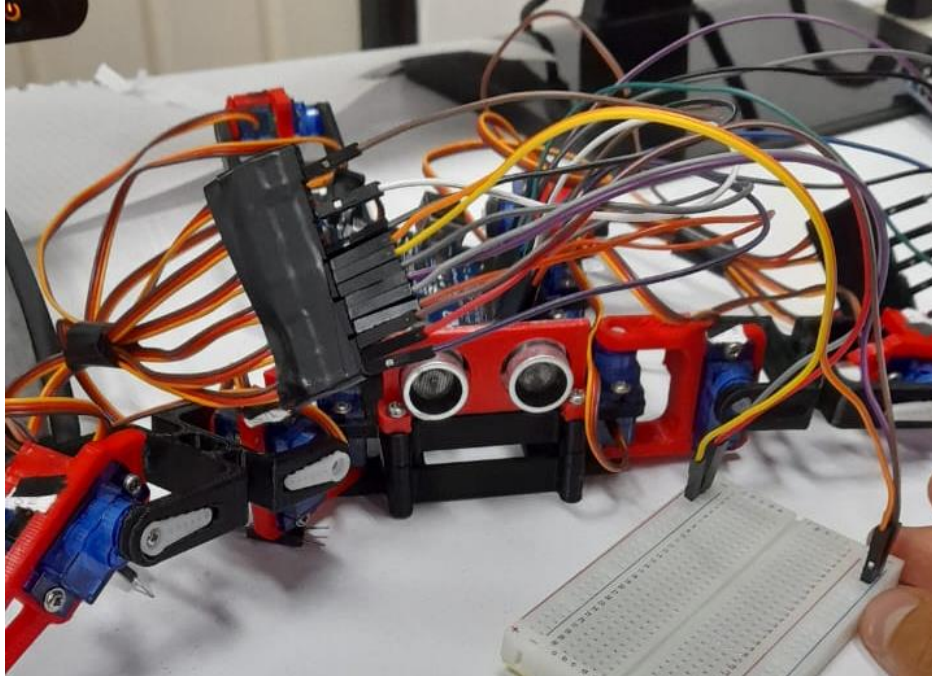


*Imagen 14 Conexión del Arduino a la protoboard*

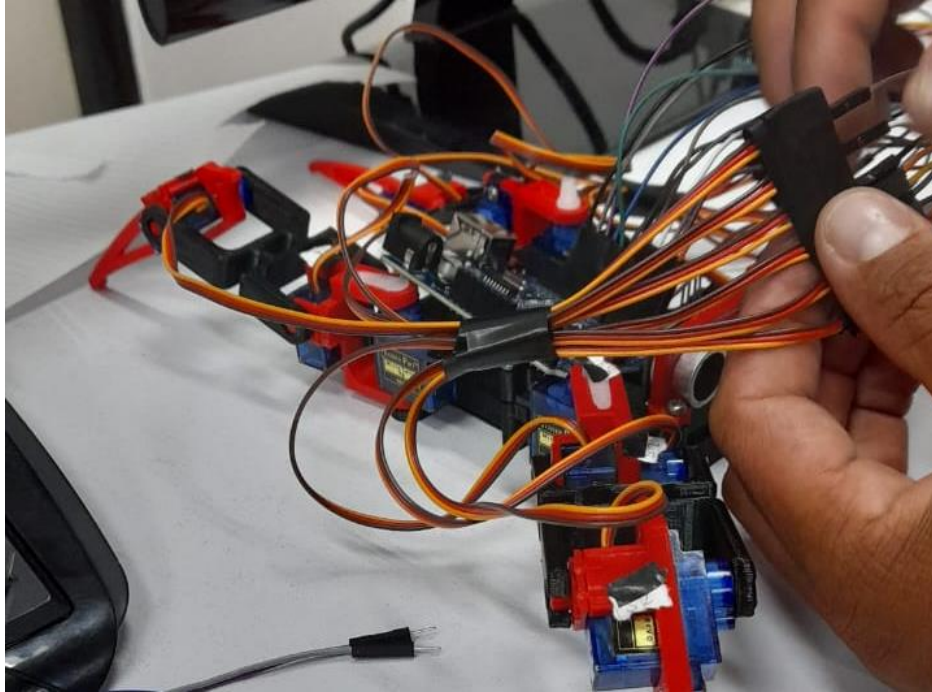
En la imagen anterior se logra observar que hemos realizado modificaciones en las conexiones de los cables de manera que sea más ordenado y fácil a la hora de revisar posibles fallas.

Ya se han realizado las respectivas conexiones de los pines de la tarjeta Arduino a la protoboard, de igual manera se está energizando la protoboard desde una batería externa independiente a la del computador; desde la que se transfiere el voltaje requerido a la tarjeta, la cual se encarga de alimentar los servomotores de nuestra araña.

Como se muestra en las siguientes imágenes:



*Imagen 15 Conexión de servomotores en la parte de control*



*Imagen 16 Ilustraciones de conexiones siguiendo código de colores*

Luego de tener nuestra araña robótica correctamente conectada, está lista para que se le ingrese el código que se encargará de ponerla en pie.

#### Programación:

A continuación, se comparte el código con el cual haremos que nuestra araña se ponga en pie:

```
#include <Servo.h> // incluye la libreria de Servo
```

```
// Declaración de los servos para las articulaciones de las patas delanteras
```

```
Servo servoPatIzqDelArt1; // Articulación 1 de la pata delantera izquierda
```

```
Servo servoPatIzqDelArt2; // Articulación 2 de la pata delantera izquierda
```

```
Servo servoPatIzqDelArt3; // Articulación 3 de la pata delantera izquierda
```

```
Servo servoPatDerDelArt1; // Articulación 1 de la pata delantera derecha
```

```
Servo servoPatDerDelArt2; // Articulación 2 de la pata delantera derecha
```



```
Servo servoPatDerDelArt3; // Articulación 3 de la pata delantera derecha
```

```
// Declaración de los servos para las articulaciones de las patas traseras
```

```
Servo servoPatIzqTrasArt1; // Articulación 1 de la pata trasera izquierda
```

```
Servo servoPatIzqTrasArt2; // Articulación 2 de la pata trasera izquierda
```

```
Servo servoPatIzqTrasArt3; // Articulación 3 de la pata trasera izquierda
```

```
Servo servoPatDerTrasArt1; // Articulación 1 de la pata trasera derecha
```

```
Servo servoPatDerTrasArt2; // Articulación 2 de la pata trasera derecha
```

```
Servo servoPatDerTrasArt3; // Articulación 3 de la pata trasera derecha
```

```
// Límites de pulso en microsegundos
```

```
int PULSOMIN = 700;
```

```
int PULSOMAX = 2350;
```

```
// Pines de los servos para la pata delantera izquierda
```

```
int PatIzqDelArt1 = 10;
```

```
int PatIzqDelArt2 = 9;
```

```
int PatIzqDelArt3 = 8;
```

```
// Pines de los servos para la pata delantera derecha
```

```
int PatDerDelArt1 = 3;
```

```
int PatDerDelArt2 = 2;
```

```
int PatDerDelArt3 = 4;
```

```
// Pines de los servos para la pata trasera izquierda
```

```
int PatIzqTrasArt1 = 12;
```

```
int PatIzqTrasArt2 = 11;
```



```
int PatIzqTrasArt3 = 13;

// Pines de los servos para la pata trasera derecha
int PatDerTrasArt1 = 6;
int PatDerTrasArt2 = 5;
int PatDerTrasArt3 = 7;

void setup() {
  // Inicialización de los servos de la pata delantera izquierda
  servoPatIzqDelArt1.attach(PatIzqDelArt1, PULSOMIN, PULSOMAX);
  servoPatIzqDelArt2.attach(PatIzqDelArt2, PULSOMIN, PULSOMAX);
  servoPatIzqDelArt3.attach(PatIzqDelArt3, PULSOMIN, PULSOMAX);

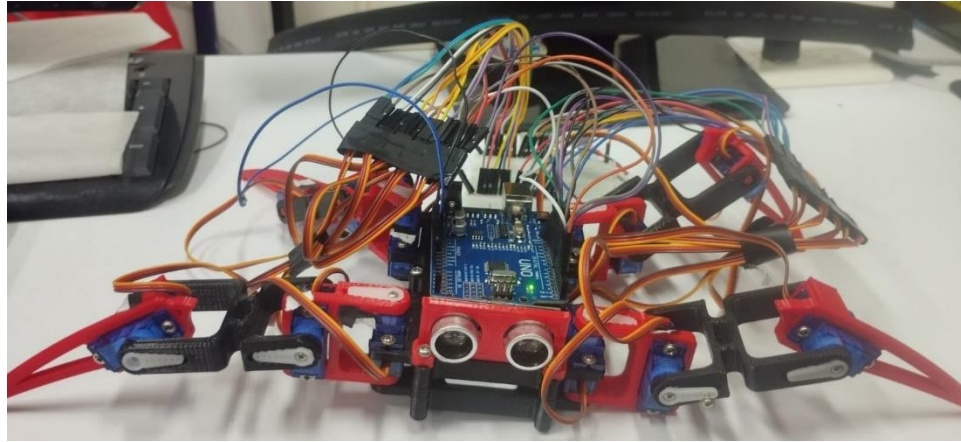
  // Inicialización de los servos de la pata delantera derecha
  servoPatDerDelArt1.attach(PatDerDelArt1, PULSOMIN, PULSOMAX);
  servoPatDerDelArt2.attach(PatDerDelArt2, PULSOMIN, PULSOMAX);
  servoPatDerDelArt3.attach(PatDerDelArt3, PULSOMIN, PULSOMAX);

  // Inicialización de los servos de la pata trasera izquierda
  servoPatIzqTrasArt1.attach(PatIzqTrasArt1, PULSOMIN, PULSOMAX);
  servoPatIzqTrasArt2.attach(PatIzqTrasArt2, PULSOMIN, PULSOMAX);
  servoPatIzqTrasArt3.attach(PatIzqTrasArt3, PULSOMIN, PULSOMAX);

  // Inicialización de los servos de la pata trasera derecha
  servoPatDerTrasArt1.attach(PatDerTrasArt1, PULSOMIN, PULSOMAX);
  servoPatDerTrasArt2.attach(PatDerTrasArt2, PULSOMIN, PULSOMAX);
  servoPatDerTrasArt3.attach(PatDerTrasArt3, PULSOMIN, PULSOMAX);
}
```

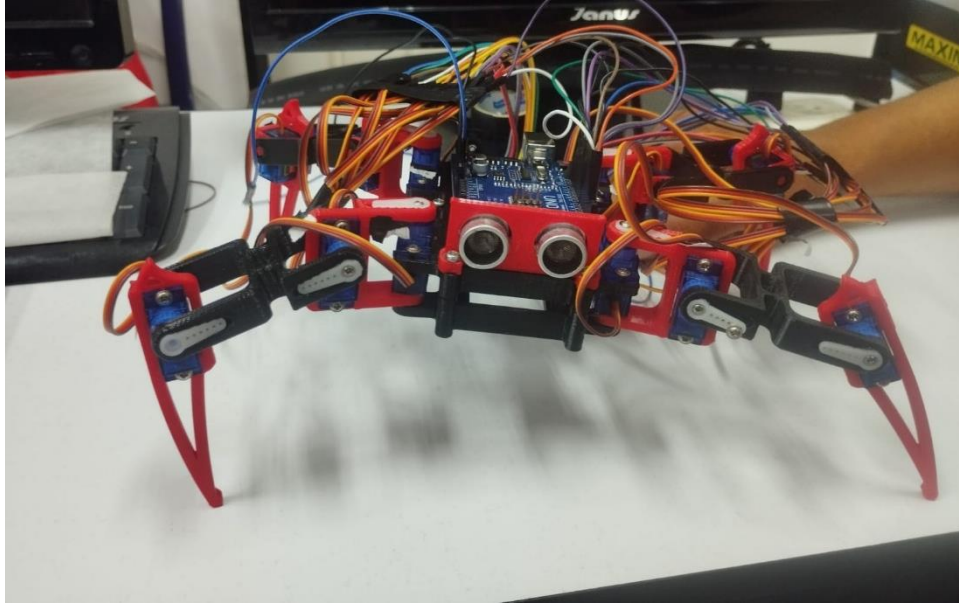


```
void loop() {  
    // Posicionar los servos de las patas delanteras a 90 grados  
    servoPatIzqDelArt1.write(90);  
    servoPatIzqDelArt2.write(90);  
    servoPatIzqDelArt3.write(90);  
  
    servoPatDerDelArt1.write(90);  
    servoPatDerDelArt2.write(90);  
    servoPatDerDelArt3.write(90);  
  
    // Posicionar los servos de las patas traseras a 90 grados  
    servoPatIzqTrasArt1.write(90);  
    servoPatIzqTrasArt2.write(90);  
    servoPatIzqTrasArt3.write(90);  
  
    servoPatDerTrasArt1.write(90);  
    servoPatDerTrasArt2.write(90);  
    servoPatDerTrasArt3.write(90);  
  
    delay(9000); // Agregar un delay para evitar sobrecargar el loop  
}
```



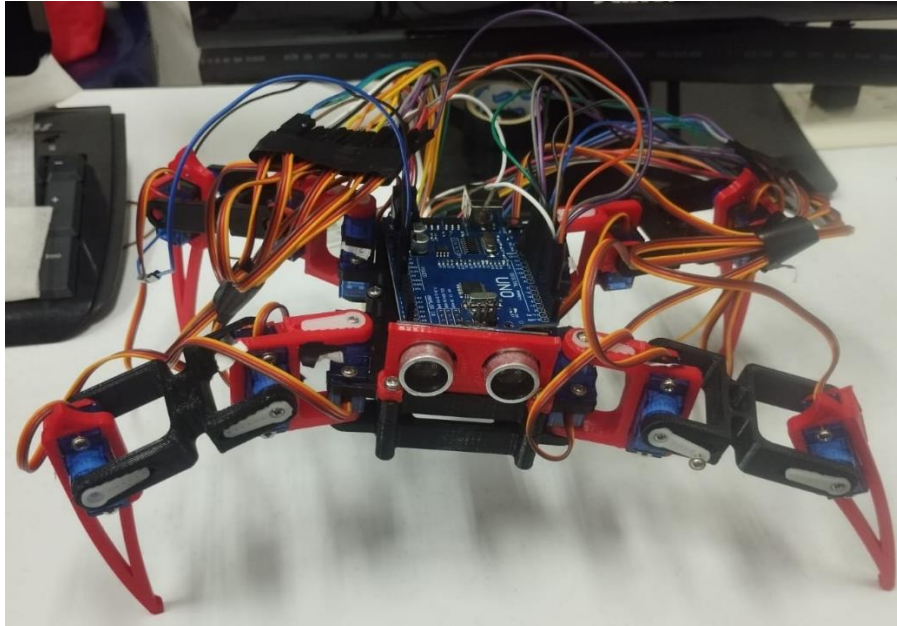
*Imagen 17 Demostración de la araña robótica no funcional*

En la imagen anterior se comparte la evidencia de que la araña robótica aún no se levanta sobre sus extremidades aun estando encendida la tarjeta de Arduino.



*Imagen 18 Araña robótica en proceso de levantamiento*

En la anterior se visualiza como la araña ha cambiado de posición sus articulaciones, con el fin de lograr levantarse sobre sus extremidades.



*Imagen 19 Araña robótica puesta en pie*

La anterior imagen nos permite visualizar como la araña se sostiene de pie de manera autónoma sobre sus propias extremidades, con esto hemos logrado el objetivo que nos trazamos.



## Conclusiones

El desarrollo del robot "araña" evasor de objetos mediante el uso de la plataforma Arduino nos ha permitido alcanzar varios aprendizajes significativos y demuestra la importancia y versatilidad de la robótica en la educación y en la vida cotidiana.

Los conocimientos que hemos adquirido en el proceso de este proyecto han sido muy valiosos y nos han permitido tener o adquirir más conocimiento en el área de programación toda la parte electrónica y demás componentes que llevaron a la ejecución de este robot autónomo (araña).



## Referencias bibliográficas.

Pedrera, A. C. (2017). *Arduino para Principiantes: 2ª Edición*. IT Campus Academy.

*La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales*

(Pinto-Salamanca, 2010) *Uso de la robótica educativa como herramienta en los procesos de enseñanza*. *Ingeniería Investigación y Desarrollo*, 10(1), 15-23.

*Ortiz, E. A. L. Modulador por ancho de pulso PWM.*

*Definiciones de términos electrónicos y robóticos.*

<https://www.youtube.com/watch?v=fiQbOWvqVco>