



Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Julio Antonio Encalada Cuenca

Sara Gabriela Cruz Naranjo

Nathaly Dayana Torres Bustamante



© Julio Antonio Encalada Cuenca
Sara Gabriela Cruz Naranjo
Nathaly Dayana Torres Bustamante

Primera edición, 2025-11-02

ISBN: 978-9942-53-001-1

DOI: <http://doi.org/10.48190/9789942530011>

Distribución online

③ Acceso abierto

Cita

Encalada, J., Cruz, S., Torres, N. (2025) Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula. Editorial Grupo Compás

Este libro es parte de la colección de la Universidad Técnica de Machala y ha sido debidamente examinado y valorado en la modalidad doble par ciego con fin de garantizar la calidad de la publicación. El copyright estimula la creatividad, defiende la diversidad en el ámbito de las ideas y el conocimiento, promueve la libre expresión y favorece una cultura viva. Quedan rigurosamente prohibidas, bajo las sanciones en las leyes, la producción o almacenamiento total o parcial de la presente publicación, incluyendo el diseño de la portada, así como la transmisión de la misma por cualquiera de sus medios, tanto si es electrónico, como químico, mecánico, óptico, de grabación o bien de fotocopia, sin la autorización de los titulares del copyright.

Prefacio

Introducción

En el siglo XXI, la educación se enfrenta a un reto fundamental: preparar a los niños para un futuro en el que la tecnología y la creatividad serán pilares esenciales. En este contexto, el pensamiento computacional emerge como una de las habilidades más valiosas que los estudiantes pueden desarrollar. Más allá de la programación o el uso de dispositivos, el pensamiento computacional implica abordar problemas de manera estructurada, analizar datos, y encontrar soluciones a través del razonamiento lógico y la innovación.

Según Herrero-Álvarez et al. (2025), la robótica educativa se presenta como una poderosa herramienta para potenciar esta habilidad. Al integrar la robótica en el aula, los docentes tienen la oportunidad de involucrar a los niños de manera activa, permitiéndoles experimentar y aprender conceptos de ciencia, matemáticas, tecnología e ingeniería a través de la práctica (Torres & Inga, 2025). Sin embargo, en el mundo actual, no basta con enseñar estas disciplinas de forma aislada. Es necesario un enfoque más integrador, como el que propone la metodología STEAM (Ciencia, Tecnología, Ingeniería, Arte y Matemáticas), que permite conectar el pensamiento lógico con la creatividad y la expresión artística (Sanz-Camarero et al., 2023).

Este libro está dirigido a futuros docentes y docentes de tercer año de EGB que desean integrar el desarrollo del pensamiento computacional mediante la robótica educativa y el enfoque STEAM en sus aulas. A lo largo de estas páginas, exploraremos cómo la robótica puede ser una excelente plataforma para enseñar a los niños a pensar de manera crítica y creativa, y cómo la metodología STEAM puede enriquecer la experiencia educativa al integrar distintas disciplinas de forma interdisciplinaria.

Objetivo general

El objetivo de este libro es proporcionar a los futuros docentes de tercer año de EGB herramientas, estrategias y recursos prácticos para integrar el desarrollo del pensamiento computacional y la robótica educativa en sus aulas, utilizando la metodología STEAM. A través de un enfoque interdisciplinario, buscamos empoderar a los educadores para que fomenten en los niños habilidades clave como la descomposición, la abstracción, el diseño de algoritmos, y la identificación de patrones, preparándolos para los retos de un futuro cada vez más tecnológico y dinámico. Este libro pretende ser una guía accesible y efectiva para transformar la enseñanza, promoviendo un aprendizaje activo y significativo que inspire a los estudiantes a explorar y crear de manera innovadora en un entorno tecnológico y colaborativo.

Estructura

- El primer capítulo desarrolla los conceptos teóricos básicos del pensamiento computacional, la programación y la metodología STEAM, destacando especialmente la conexión entre estos temas y la utilización de sensores ultrasónicos. Se expone una visión general, pero se profundiza particularmente en el funcionamiento y aplicación de estos sensores como eje central de la fundamentación teórica.
- En el capítulo 2, se analiza cómo organizar proyectos de programación enfocados en sensores ultrasónicos, detallando el papel fundamental de las variables para gestionar sus datos, así como el uso de procedimientos y funciones para resolver problemas prácticos. Además, se trabajan las estructuras de control secuenciales que permiten automatizar procesos con sensores ultrasónicos, enmarcando todo dentro de las áreas de Tecnología, Ingeniería y Matemáticas de la metodología STEAM.
- El capítulo 3 se enfoca en integrar los sensores ultrasónicos dentro de proyectos interdisciplinarios que abordan las cinco dimensiones de STEAM: Ciencia, Tecnología,

Ingeniería, Arte y Matemáticas. Se presentan actividades y secuencias didácticas que permiten aplicar de manera práctica los conocimientos adquiridos previamente, promoviendo la creatividad y el pensamiento crítico en la resolución de retos reales.

- En la figura 1 se muestra un resumen gráfico de los aprendizajes por cada área STEAM que debe lograr el futuro docente para enseñar a los niños. A medida que avanzan los capítulos, en este libro, se van pintando en el gráfico las áreas trabajadas.

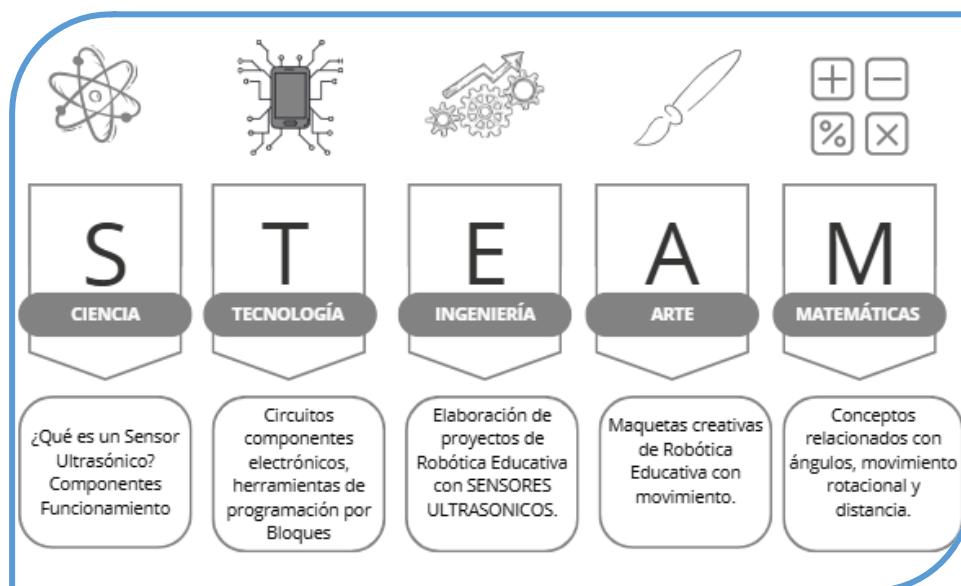
Características pedagógicas del libro

- Enfoque Constructivista: El libro está basado en el principio de que los estudiantes aprenden mejor a través de la experiencia práctica. Por ello, se enfoca en actividades y proyectos que fomentan el aprendizaje activo, donde los estudiantes construyen, programan y experimentan con circuitos electrónicos basados en sensores ultrasónicos, creando su propio conocimiento de forma autónoma y colaborativa (Leung et al., 2024).
- Aprendizaje Basado en Proyectos (ABP): Cada capítulo incluye proyectos prácticos sobre sensores ultrasónicos que permiten a los estudiantes aplicar los conceptos aprendidos a través de la resolución de problemas reales. Los proyectos no solo sirven para consolidar conocimientos técnicos, sino también para desarrollar habilidades como el trabajo en equipo, la creatividad y la resolución de problemas (Hao et al., 2024).
- Enfoque educativo STEAM: Los proyectos de robótica educativa basados en sensores ultrasónicos que se elaboran en este libro adoptan el enfoque STEAM, lo que permite a los estudiantes explorar y aplicar conceptos de Ciencia, Tecnología, Ingeniería, Arte y Matemáticas de manera práctica y colaborativa (Huang et al., 2023).

La presente obra ofrece a los futuros docentes de niños los fundamentos esenciales para abordar la enseñanza aprendizaje del pensamiento computacional utilizando robótica educativa basada en circuitos de sensores ultrasónicos y metodología STEAM. Presenta la información clave, sugiere actividades didácticas y ofrece recursos básicos para su implementación. Además, brinda flexibilidad al docente para adaptar y enriquecer la metodología según sus necesidades y contexto.

La Robótica Educativa con metodología STEAM permite que los niños desarrollen pensamiento crítico, creatividad y habilidades para la resolución de problemas mediante la integración de Ciencia, Tecnología, Ingeniería, Arte y Matemáticas. Fomenta el aprendizaje activo, la curiosidad y la conexión con el mundo real (Kalaitzidou & Pachidis, 2023)

Figura 1
Gráfico de aprendizaje de las áreas STEAM



Nota. En esta figura se observa las áreas de aprendizaje STEAM que el futuro docente debe lograr.

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

En este libro, los principios pedagógicos constructivistas guían la elaboración de actividades educativas basadas en robótica educativa para que los futuros docentes de tercer año de EGB puedan trabajarlas con los niños en sus aulas de clases (Addas et al., 2024). A través de los diferentes capítulos, la robótica educativa se trabaja involucrando circuitos electrónicos basados en sensores ultrasónicos; se proponen una gran variedad de técnicas activas para trabajar con los niños el pensamiento computacional con robótica educativa a través de la metodología STEAM.

Contenido

CAPÍTULO 1: EL PENSAMIENTO COMPUTACIONAL, LA PROGRAMACIÓN Y LA METODOLOGÍA STEAM USANDO SENSOR ULTRASÓNICO	12
1.1. Objetivo del capítulo	13
1.2. Aprendizajes Esperados.....	13
1.3. Introducción: El pensamiento computacional en la programación de sensor ultrasónico.....	13
1.4. Definición de pensamiento computacional asociada al sensor ultrasónico	16
1.5. Definición de programación asociada al sensor ultrasónico.....	20
1.6. Definición de metodología STEAM asociada al sensor ultrasónico	21
1.7. Relación del pensamiento computacional, la programación y la metodología STEAM.....	23
1.8. Resumen del capítulo	26
1.9. Autoevaluación del capítulo.....	26
CAPÍTULO 2: LA PROGRAMACIÓN EN LA RESOLUCIÓN DE PROBLEMAS CONDICIONALES CON SENSOR ULTRASÓNICO	28
2.1. Objetivo del capítulo	29
2.2. Aprendizajes Esperados.....	29
2.3. Introducción: la educación del pensamiento computacional condicional	29
2.4. Organización de la programación de sensores ultrasónicos con <i>mBlock</i>	42
2.5. La importancia y funcionamiento de las variables en programación de sensores ultrasónicos con <i>mBlock</i>	43
2.6. El rol de la programación de procedimientos y funciones en la resolución de problemas con sensores ultrasónicos en <i>mBlock</i>	44

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula	
2.7. Sentencias de control condicionales para sensores ultrasónicos con <i>mBlock</i>	46
2.7. El hardware necesario para trabajar robótica educativa basada en sensores ultrasónicos con <i>mBlock</i>	55
2.9. Resumen del capítulo	78
2.10. Autoevaluación del capítulo.....	79
CAPÍTULO 3: APRENDIZAJE CREATIVO DE PROGRAMACIÓN CON SENsoRES ULTRASÓNICOS.....	83
3.1. Objetivo del capítulo	83
3.2. Aprendizajes Esperados.....	83
3.3. Introducción: La Educación con el Aprendizaje Basada en Proyectos de Sensores Ultrasónicos.....	83
3.4. Aprendizaje Basado en Proyectos (ABP) utilizando sensores ultrasónicos	87
3.5. Secuencia didáctica 1: El sensor ultrasónico y el descubrimiento del mundo a través del sonido	90
3.6. Secuencia didáctica 2: Aprendiendo a programar con <i>mBlock</i>	103
3.7. Secuencia Didáctica 3: Identificación de animales carnívoros y herbívoros.....	118
3.8. Secuencia Didáctica 4: Construcción de una maqueta de un puente levadizo	127
3.9. Resumen del capítulo	138
3.10. Desafíos prácticos propuestos con sensores ultrasónicos	138
3.11. Autoevaluación del capítulo.....	141
CRÉDITOS DE SOFTWARE Y RECURSOS VISUALES	144
REFERENCIAS BIBLIOGRÁFICAS.....	145

CAPITULO 1: EL PENSAMIENTO COMPUTACIONAL, LA PROGRAMACIÓN Y LA METODOLOGÍA STEAM USANDO SENSOR ULTRASÓNICO

1.1. Objetivo del capítulo

Al terminar este capítulo, los futuros profesores contarán con las herramientas necesarias para comprender, analizar y explicar con claridad los conceptos de pensamiento computacional, programación y la metodología STEAM utilizando como elemento central el sensor ultrasónico. Además, podrá identificar de manera crítica las relaciones entre estos elementos y cómo su integración puede enriquecer los procesos de enseñanza y aprendizaje en contextos educativos diversos.

1.2. Aprendizajes Esperados

- Definición de pensamiento computacional y el sensor ultrasónico
- Definición de programación y el sensor ultrasónico
- Definición de metodología STEAM y el sensor ultrasónico
- Relación del pensamiento computacional, la programación y la metodología STEAM

1.3. Introducción: El pensamiento computacional en la programación de sensor ultrasónico

En el proceso de enseñanza-aprendizaje del tercer año de Educación General Básica (EGB), es fundamental incorporar estrategias que fomenten el desarrollo del pensamiento lógico, la creatividad y el interés por la tecnología desde edades tempranas. A esta edad, los niños comienzan a construir habilidades cognitivas más complejas y muestran una gran disposición para explorar, experimentar y resolver problemas. Por ello, se vuelve necesario integrar recursos educativos que promuevan la curiosidad, el análisis y la innovación. Este capítulo se enfoca en tres ejes esenciales para lograr ese propósito: el pensamiento computacional, la programación y la metodología

STEAM, todos ellos aplicados al uso del sensor ultrasónico como una herramienta tecnológica didáctica.

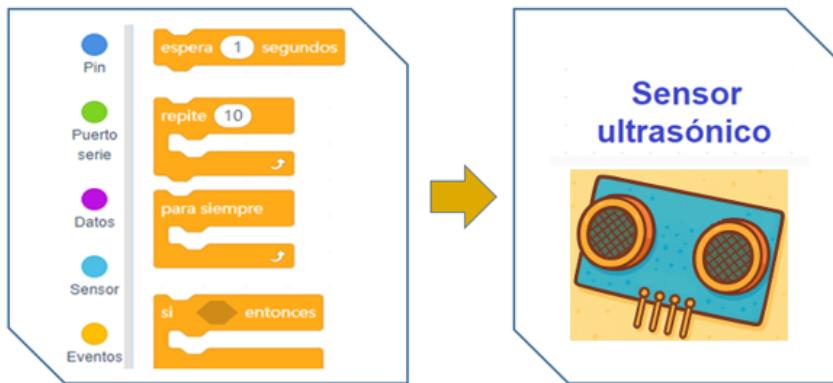
De acuerdo a Zapata-Cáceres et al. (2024), el pensamiento computacional se refiere a la capacidad de enfrentar problemas de manera estructurada, a través de habilidades como la descomposición, la identificación de patrones, el diseño de algoritmos y la evaluación de soluciones. Estas competencias, cuando se articulan con la programación y el uso de herramientas tecnológicas en entornos educativos, favorecen el aprendizaje interdisciplinario y potencian la creatividad y el razonamiento lógico dentro del enfoque STEAM.

Cuando se integra la robótica basada en sensores ultrasónicos con la programación, estas capacidades se traducen en acciones concretas que permiten controlar dispositivos, como el sensor ultrasónico, un componente electrónico que mide distancias mediante ondas de sonido. En el aula, su uso permite a los estudiantes observar cómo una secuencia de instrucciones puede generar un resultado real, fomentando la conexión entre teoría y práctica. Según Huang et al. (2023), la robótica educativa en las áreas STEAM proporciona un entorno de aprendizaje activo donde los estudiantes aplican conocimientos de diferentes áreas para resolver problemas reales, fortaleciendo la comprensión conceptual y las habilidades prácticas

El sensor ultrasónico puede ser programado tanto mediante lenguajes de programación textual como a través de entornos visuales basados en bloques, lo que facilita su comprensión y uso en niveles educativos iniciales. De acuerdo con Dúo-Terrón (2023), la programación de robots en entornos educativos promueve competencias propias del enfoque STEAM, como la resolución de problemas, el pensamiento crítico, la creatividad y la colaboración. Esta versatilidad de la robótica educativa permite que los estudiantes interactúen de manera intuitiva con la tecnología, como se muestra en la figura 2, favoreciendo el aprendizaje práctico y significativo desde una edad temprana.

Figura 2

Interfaz de programación con bloques de un sensor ultrasónico



Nota. En esta figura se observa un sensor ultrasónico en un ambiente de programación con bloques. Fuente: Captura de pantalla adaptada de la interfaz de mBlock v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Por su parte, la metodología STEAM (Ciencia, Tecnología, Ingeniería, Arte y Matemáticas) busca conectar distintas disciplinas para ofrecer una experiencia educativa integral, dinámica y centrada en el aprendizaje activo (Thoma et al., 2023). La inclusión del sensor ultrasónico en proyectos STEAM brinda a los estudiantes oportunidades para construir soluciones tecnológicas sencillas, despertar su interés por el funcionamiento de los dispositivos, y aplicar conceptos matemáticos y científicos en contextos reales. El enfoque STEAM promueve la colaboración, fomenta la creatividad y contribuye al desarrollo de competencias fundamentales para la educación (Terrón, 2025).

A lo largo de este capítulo, se analizará cómo el pensamiento computacional, la programación y la metodología STEAM pueden trabajarse de manera articulada en el tercer año de EGB, utilizando el sensor ultrasónico como recurso central. Además, se presentarán ejemplos prácticos y propuestas pedagógicas que

permitirán a los futuros docentes implementar actividades innovadoras y significativas en el aula. El objetivo es brindar una base teórica y metodológica que facilite la integración de la tecnología en los procesos educativos, promoviendo un aprendizaje más participativo, contextualizado y orientado al desarrollo integral del estudiante.

1.4. Definición de pensamiento computacional asociada al sensor ultrasónico

En términos de Sherwood et al. (2024), el pensamiento computacional es una habilidad cognitiva que permite resolver problemas de manera lógica y ordenada, utilizando procesos similares a los que emplea una computadora. Esta forma de pensamiento no se limita al ámbito de la informática, sino que puede aplicarse en diversas áreas del conocimiento, ya que implica descomponer problemas complejos en partes más pequeñas, identificar patrones, crear algoritmos y evaluar posibles soluciones. En el contexto educativo, especialmente en los primeros años de la Educación General Básica, el pensamiento computacional promueve el desarrollo del razonamiento crítico, la toma de decisiones y la autonomía en la resolución de tareas cotidianas o académicas.

Según Alenezi, (2024), aplicar el pensamiento computacional en el aula no requiere necesariamente el uso de computadoras, aunque sí se potencia con herramientas tecnológicas. Actividades como organizar secuencias de instrucciones, clasificar información o diseñar soluciones paso a paso pueden estimular esta forma de pensamiento desde edades tempranas. En niveles como el tercer año de EGB, su implementación permite que los estudiantes desarrollen habilidades de lógica, planificación y análisis a través del juego, la exploración y la experimentación, sentando las bases para futuros aprendizajes en áreas como la programación, la robótica educativa y las ciencias en general.

Liao et al. (2024) concluye que desde una perspectiva educativa, el pensamiento computacional también se define como una

competencia fundamental para la formación de estudiantes críticos y creativos, capaces de enfrentar desafíos mediante la formulación de soluciones paso a paso. Esta habilidad no solo implica saber cómo resolver un problema, sino también entender y organizar el proceso necesario para llegar a la solución. A través del pensamiento computacional, los estudiantes aprenden a planificar, anticipar resultados, tomar decisiones basadas en la lógica y reflexionar sobre sus propios procesos mentales. Todo esto fortalece su autonomía, su capacidad de análisis y su seguridad al enfrentarse a situaciones nuevas o complejas en distintos ámbitos del conocimiento.

Dong et al. (2025) argumentan que la aplicación del pensamiento computacional con enfoque STEAM en el aula va más allá del uso de computadoras, ya que puede desarrollarse mediante juegos, secuencias de instrucciones, desafíos de lógica, y especialmente a través de herramientas como la programación y la robótica educativa. Por ejemplo, cuando se utiliza un sensor ultrasónico en proyectos escolares, los estudiantes deben pensar cómo estructurar un algoritmo que permita al robot tomar decisiones en función de la distancia detectada. Esta experiencia concreta les permite visualizar el impacto de sus decisiones lógicas en un resultado físico y real, conectando la abstracción con la acción. En este sentido, Sun et al. (2023) señalan que el pensamiento computacional se convierte en una base esencial para integrar la tecnología de manera significativa en los procesos de enseñanza-aprendizaje, promoviendo una educación activa y contextualizada.

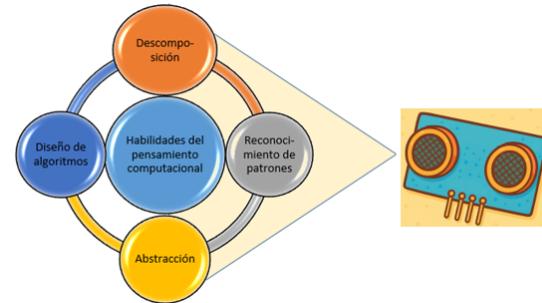
Habilidades que incluye el pensamiento computacional asociadas al sensor ultrasónico

Coenraad et al. (2025) indican que las habilidades cognitivas que incluye el pensamiento computacional son: la descomposición de problemas, el reconocimiento de patrones, la abstracción de información, el desarrollo de algoritmos (ver figura 3). Hijón-Neira

et al. (2024) mencionan que estas habilidades permiten enfrentar problemas complejos de forma lógica, creativa y eficaz, y tienen aplicaciones tanto en entornos tecnológicos como en la vida diaria (véase figura 3). Su desarrollo se vuelve especialmente significativo cuando se incorporan sensores ultrasónicos en actividades de programación y robótica educativa. Al utilizar este tipo de sensor, que detecta distancias mediante ondas sonoras, los estudiantes deben analizar cómo interpretar la información que recoge el entorno y transformarla en respuestas programadas dentro del sistema robótico.

Por ejemplo, al programar un robot para detenerse al detectar un obstáculo cercano o encender una luz cuando algo se aproxima, los estudiantes deben aplicar procesos de pensamiento computacional como la descomposición del problema, la identificación de patrones de comportamiento, la abstracción de los datos relevantes del sensor, y el diseño de algoritmos que ejecuten una acción específica según la distancia medida. Este tipo de experiencias STEAM les permite conectar los conceptos abstractos con resultados reales y visibles, convirtiendo el pensamiento computacional en una herramienta práctica para crear soluciones automatizadas e inteligentes (Dong et al., 2025).

Figura 3
Habilidades del pensamiento computacional relacionadas con sensor ultrasónico



Nota. En esta figura se observan habilidades del pensamiento computacional relacionadas con sensor ultrasónico.

A continuación, se describen las habilidades de pensamiento computacional, las cuales son fundamentales para la resolución de problemas de manera lógica, estructurada y eficiente. Estas habilidades incluyen la descomposición de problemas, el reconocimiento de patrones, la abstracción y el diseño de algoritmos (ver figura 4), elementos esenciales para enfrentar desafíos en entornos digitales y tecnológicos.

Descomposición: La descomposición consiste en dividir un problema complejo en partes más pequeñas y manejables (Pan et al., 2024). En el uso del sensor ultrasónico, esta habilidad se pone en práctica cuando los estudiantes analizan cada componente de una tarea robótica. Por ejemplo, si el objetivo es que un robot evite obstáculos, primero deben separar el problema en subtareas: detectar la distancia, interpretar los datos del sensor, tomar una decisión (girar, detenerse o avanzar), y ejecutar una acción mediante el motor. Esta secuencia ayuda a los niños a entender cómo se construyen soluciones paso a paso y fomenta una planificación ordenada.

Reconocimiento de patrones: El reconocimiento de patrones implica identificar regularidades o comportamientos repetitivos para predecir o anticipar lo que ocurrirá (Palma et al., 2024). En actividades con sensores ultrasónicos, los estudiantes observan cómo el robot responde de forma similar cada vez que detecta un objeto a cierta distancia. Al notar que, por ejemplo, el robot siempre gira hacia la izquierda cuando un obstáculo se encuentra a menos de 10 cm, los niños pueden deducir un patrón de comportamiento y aplicarlo en nuevos contextos. Esta habilidad les permite anticipar resultados y ajustar el código de manera más eficiente.

Abstracción: La abstracción permite enfocarse en los elementos más relevantes de un problema y eliminar los detalles innecesarios (Liu et al., 2024). Al trabajar con el sensor ultrasónico, esta habilidad se activa cuando los estudiantes se concentran únicamente en la información que realmente importa: la distancia que mide el sensor. Por ejemplo, no es necesario saber qué tipo

de objeto hay delante del robot, sino solamente cuán lejos está. Al abstraer esa información clave, los niños pueden programar respuestas efectivas sin distraerse con datos secundarios, desarrollando así una forma de pensar más precisa y funcional.

Diseño de algoritmos: El diseño de algoritmos consiste en crear una secuencia de pasos claros y ordenados para resolver un problema (Åkerfeldt et al., 2024). En el caso del sensor ultrasónico, los estudiantes deben pensar en qué debe hacer el robot dependiendo de las distancias detectadas. Por ejemplo: "Si la distancia es mayor a 20 cm, avanzar; si es menor a 20 cm, detenerse; si es menor a 10 cm, girar". Este proceso de programación permite que los estudiantes visualicen cómo una serie de instrucciones lógicas puede generar un comportamiento inteligente en un robot, fortaleciendo así su capacidad de estructurar soluciones precisas y replicables.

1.5. Definición de programación asociada al sensor ultrasónico

La programación es el proceso de diseñar, escribir y mantener el código fuente de programas informáticos (Alonso-García et al., 2024). Este código es un conjunto de instrucciones que una computadora puede entender y ejecutar para realizar tareas específicas. La programación se lleva a cabo utilizando lenguajes de programación, que permiten a los desarrolladores comunicarse con la máquina de una manera lógica y estructurada.

El objetivo principal de la programación es resolver problemas o automatizar procesos mediante el uso de algoritmos. Un algoritmo es una serie de pasos definidos que conducen a la solución de un problema. Al traducir estos pasos a un lenguaje de programación, el programador crea software que puede realizar desde tareas simples, como sumar números, hasta funciones complejas, como manejar bases de datos o controlar un robot.

Además de escribir código, la programación implica probar, depurar (corregir errores), optimizar y documentar el software.

Sun et al. (2024) indican que existen diferentes formas de programar, entre las que destacan la programación textual y la basada en bloques. La programación textual utiliza directamente el lenguaje de programación escrito, como Python o Java, lo que permite un mayor control y flexibilidad. Por otro lado, la programación por bloques, como la que se encuentra en plataformas como *Scratch* y *mBlock*, es ideal para principiantes, ya que utiliza piezas visuales que se arrastran y encajan como rompecabezas, facilitando la comprensión de los conceptos básicos. Ambas formas promueven el pensamiento lógico y la resolución de problemas, haciendo que la programación sea accesible para personas de todas las edades y niveles de experiencia.

Shanshan y Sen (2024) sostienen que la programación es el procedimiento de elaborar instrucciones comprensibles para una computadora o dispositivo, con el propósito de ejecutar tareas específicas, como la interacción con componentes electrónicos de robótica. Un ejemplo claro es el uso del sensor ultrasónico, común en proyectos de robótica y automatización, el cual permite medir distancias mediante ondas de sonido. A través de la programación, se puede configurar este sensor para detectar obstáculos, calcular la distancia a un objeto y activar respuestas automáticas, como detener un motor o cambiar de dirección. De esta forma, la programación no solo se limita al mundo virtual, sino que también conecta con el entorno físico, permitiendo que los sistemas respondan de manera inteligente a lo que los rodea.

1.6. Definición de metodología STEAM asociada al sensor ultrasónico

Silva-Hormazábal y Alsina (2023) señalan que la metodología STEAM es un enfoque educativo que integra de manera interdisciplinaria cinco áreas clave del conocimiento: Ciencia (Science), Tecnología (Technology), Ingeniería (Engineering), Arte (Arts) y Matemáticas (Mathematics). Su objetivo principal es fomentar el aprendizaje significativo mediante la aplicación práctica de conceptos teóricos en contextos reales o simulados,

promoviendo la creatividad, el pensamiento crítico y la resolución de problemas. A diferencia de los métodos tradicionales, STEAM busca romper con la enseñanza fragmentada y propone una visión más holística y conectada del aprendizaje.

Rodrigues-Silva y Alsina (2023) manifiestan que uno de los pilares de STEAM es el aprendizaje basado en proyectos, donde los estudiantes investigan, diseñan y desarrollan soluciones a problemas reales o hipotéticos. Este enfoque les permite aplicar conocimientos de diversas disciplinas de forma integrada, lo que favorece una comprensión más profunda y duradera. Además, fomenta el trabajo en equipo, la comunicación efectiva y la toma de decisiones, habilidades esenciales tanto en el ámbito académico como en el mundo laboral.

García Fuentes et al. (2023) indican que el componente artístico (Arts) dentro de STEAM tiene un papel fundamental, ya que estimula la imaginación y la innovación. A través del arte, los estudiantes pueden expresar ideas complejas de manera visual o creativa, mejorando su capacidad para pensar de forma divergente. Esta integración también promueve la inclusión de diferentes estilos de aprendizaje, permitiendo que más estudiantes se sientan identificados y motivados a participar activamente en su proceso educativo.

Dentro del enfoque STEAM, el uso del sensor ultrasónico es un excelente ejemplo de cómo se integran distintas áreas del conocimiento en un solo proyecto. Este dispositivo, que permite medir distancias mediante ondas de sonido, puede ser programado para detectar obstáculos en aplicaciones como vehículos autónomos, sistemas de seguridad o robots móviles. Al trabajar con él, los estudiantes aplican conceptos de física (como la velocidad del sonido), tecnología y programación (para procesar las señales del sensor), ingeniería (para integrar el sensor en un sistema funcional) y matemáticas (para calcular distancias). Incluso el arte puede estar presente al diseñar la estructura o el entorno donde se instala el sensor, demostrando cómo un solo

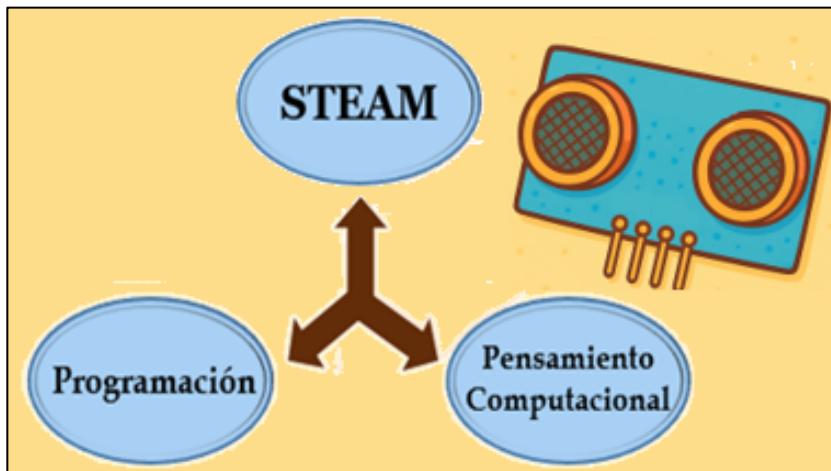
Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula componente tecnológico puede convertirse en el eje de un proyecto STEAM completo.

1.7. Relación del pensamiento computacional, la programación y la metodología STEAM

El pensamiento computacional es una habilidad fundamental en la educación moderna que permite a los estudiantes abordar problemas de manera lógica, estructurada y eficiente. Esta forma de pensar incluye procesos como la descomposición de problemas, el reconocimiento de patrones, la abstracción y el diseño de algoritmos. Estas habilidades no solo se aplican en la informática, sino que son útiles en diversas disciplinas, ya que ayudan a entender y resolver situaciones complejas de forma sistemática. Estos detalles combinan el pensamiento computacional, la programación y la metodología STEAM (ver figura 4).

Figura 4

Relación del pensamiento computacional, programación y STEAM asociadas al sensor ultrasónico



Nota. En esta figura se observa la relación del pensamiento computacional, programación y STEAM asociadas al sensor ultrasónico.

La programación es una herramienta clave para desarrollar el pensamiento computacional, ya que permite poner en práctica estos procesos mediante la creación de soluciones digitales. Al programar, los estudiantes transforman ideas abstractas en acciones concretas que una máquina puede ejecutar. Esto refuerza su capacidad de análisis, precisión y creatividad, fomentando una mentalidad de prueba y error que impulsa el aprendizaje continuo. Programar no solo enseña a resolver problemas, sino también a diseñar nuevas formas de interacción con el mundo digital y físico.

Olivato y Silva (2023) argumentan que la metodología STEAM incorpora tanto el pensamiento computacional como la programación como componentes esenciales de su enfoque interdisciplinario. A través de proyectos integrados, los estudiantes enfrentan desafíos reales que requieren la aplicación conjunta de conocimientos de ciencia, tecnología, ingeniería, arte y matemáticas. En este contexto, el pensamiento computacional se convierte en una estrategia para organizar ideas y resolver problemas, mientras que la programación se convierte en el medio para implementar soluciones. De esta manera, STEAM no solo promueve el aprendizaje técnico, sino que también desarrolla habilidades cognitivas, creativas y sociales necesarias para el siglo XXI.

El sensor ultrasónico, al integrarse en proyectos educativos bajo la metodología STEAM, se convierte en una herramienta poderosa para aplicar el pensamiento computacional y la programación en un contexto tangible. Los estudiantes deben analizar cómo funciona el sensor, descomponer el problema de medición de distancias, identificar patrones en los datos recolectados y diseñar algoritmos que permitan actuar en función de esos datos, por ejemplo, detener un robot cuando se aproxima a un obstáculo. Estas actividades STEAM fortalecen el pensamiento lógico y la capacidad para estructurar soluciones, promoviendo un aprendizaje activo y significativo (Román-Graván & Arrifano-Tadeu, 2025).

Además, la programación del sensor ultrasónico brinda la oportunidad de experimentar con variables físicas reales, como el tiempo de ida y vuelta de una onda sonora, lo que conecta directamente con conceptos de ciencia y matemáticas. Los estudiantes pueden escribir código para calcular distancias, graficar resultados, y ajustar parámetros según el entorno, desarrollando así competencias técnicas junto con habilidades analíticas y creativas. Todo esto se alinea perfectamente con los principios STEAM, donde el uso de la tecnología no es un fin en sí mismo, sino un medio para explorar, innovar y resolver problemas del mundo real (Suparman et al., 2025).

Caso práctico: Un proyecto introductorio para enseñar robótica educativa consiste en utilizar un sensor ultrasónico conectado a una placa controladora y programado con *mBlock* para medir distancias y activar un indicador luminoso o sonoro. La actividad permite que los estudiantes comprendan cómo un dispositivo electrónico puede detectar objetos y responder de forma automática, estableciendo una conexión directa entre los principios del pensamiento computacional y su aplicación práctica en educación STEAM (Cachay, 2022).

Relación con las áreas STEAM:

- Ciencia: estudio de la propagación del sonido y el principio de funcionamiento del sensor ultrasónico.
- Tecnología y Programación: uso de *mBlock* para configurar la lectura del sensor y programar la respuesta automática.
- Ingeniería: montaje del sensor, cableado y ensamblaje del sistema.
- Arte: diseño visual del dispositivo o estructura donde se incorpora el sensor.
- Matemáticas: cálculo de distancias y umbrales de activación a partir de las mediciones.

1.8. Resumen del capítulo

Este capítulo presenta los fundamentos del pensamiento computacional, la programación y la metodología STEAM, vinculados con el uso del sensor ultrasónico como recurso pedagógico. Se explica qué es el pensamiento computacional, sus habilidades clave y su aplicación en la robótica educativa. Se describen los principios de la programación, su relación con el control de dispositivos, y la integración de las áreas STEAM para un aprendizaje interdisciplinario. Finalmente, se expone cómo estos elementos se interrelacionan en proyectos educativos, favoreciendo la creatividad, el trabajo colaborativo y la resolución lógica de problemas en estudiantes de tercer año de EGB.

1.9. Autoevaluación del capítulo

Pregunta 1: ¿Cuál de las siguientes opciones define mejor el pensamiento computacional?

Opciones de respuesta:

- a) La capacidad de crear arte con herramientas digitales.
- b) La habilidad de resolver problemas siguiendo procesos lógicos y estructurados.
- c) El conocimiento de cómo funcionan los sensores electrónicos.
- d) El uso exclusivo de computadoras para programar.

Pregunta 2: ¿Qué función principal cumple un sensor ultrasónico en un proyecto de programación?

Opciones de respuesta:

- a) Detectar colores en un objeto cercano.
- b) Medir la distancia a un objeto mediante ondas sonoras.
- c) Enviar datos de temperatura a un servidor.
- d) Emitir luz infrarroja para detectar movimiento.

Pregunta 3: ¿Cuál es una característica central de la metodología STEAM?

Opciones de respuesta:

- a) Enseñar únicamente matemáticas y tecnología.
- b) Promover el aprendizaje memorístico de teorías científicas.
- c) Integrar ciencia, tecnología, ingeniería, arte y matemáticas en proyectos prácticos.
- d) Separar las materias para facilitar el aprendizaje.

Pregunta 4: ¿Cómo se relaciona la programación con el pensamiento computacional en un proyecto con sensor ultrasónico?

Opciones de respuesta:

- a) La programación reemplaza completamente el pensamiento computacional.
- b) El pensamiento computacional permite diseñar soluciones, mientras que la programación las implementa.
- c) Solo se utiliza programación cuando no hay sensores involucrados.
- d) La programación se usa únicamente para el diseño gráfico del proyecto.

Pregunta 5: ¿Por qué el sensor ultrasónico es un recurso valioso en la metodología STEAM?

Opciones de respuesta:

- a) Porque solo permite trabajar temas de física y matemáticas.
- b) Porque mide la humedad del ambiente con gran precisión.
- c) Porque conecta disciplinas como tecnología, ingeniería y ciencia en un proyecto práctico.
- d) Porque solo se utiliza para actividades de entretenimiento.

Pregunta 6: ¿Cuál es el principal objetivo de la programación?

Opciones de respuesta:

- a) Crear videojuegos únicamente
- b) Resolver problemas o automatizar procesos mediante algoritmos
- c) Enseñar a las computadoras a pensar como humanos
- d) Diseñar hardware para computadoras

Pregunta 7: ¿Qué es un lenguaje de programación?

Opciones de respuesta:

- a) Un sistema operativo de una computadora
- b) Un conjunto de instrucciones que una computadora no puede interpretar

- c) Un medio lógico y estructurado para que los desarrolladores se comuniquen con la máquina
- d) Una forma de crear hardware de forma automática

Pregunta 8: ¿A qué se refiere un algoritmo en el contexto de la programación?

Opciones de respuesta:

- a) A un error en el código que debe corregirse
- b) A un conjunto de pasos definidos que llevan a la solución de un problema
- c) A una lista de programas instalados en una computadora
- d) A una técnica de diseño gráfico para computadoras

CAPÍTULO 2: LA PROGRAMACIÓN EN LA RESOLUCIÓN DE PROBLEMAS CONDICIONALES CON SENSOR ULTRASÓNICO

2.1. Objetivo del capítulo

Al finalizar este capítulo, el docente en formación estará en condiciones de utilizar el pensamiento computacional, especialmente en lo relacionado con las estructuras condicionales, mediante el uso de la plataforma *mBlock* y la implementación de circuitos electrónicos que incluyan sensores ultrasónicos como parte de sus prácticas educativas.

2.2. Aprendizajes Esperados

- Organización de la programación con sensores ultrasónicos
- La importancia y funcionamiento de las variables en programación con sensores ultrasónicos
- El rol de la programación de procedimientos y funciones en la resolución de problemas con sensores ultrasónicos
- Sentencias de control condicionales con sensores ultrasónicos

2.3. Introducción: la educación del pensamiento computacional condicional

La programación con sensores ultrasónicos es una herramienta clave en el desarrollo de sistemas inteligentes capaces de interactuar con su entorno. Estos sensores, que miden distancias a través de ondas sonoras, son ampliamente utilizados en robótica, automatización y vehículos autónomos. Para lograr que un sistema responda adecuadamente a los datos que recibe del sensor, es fundamental organizar el código de manera estructurada y eficiente, facilitando su mantenimiento, legibilidad y escalabilidad. En este capítulo, se explorará cómo estructurar correctamente la programación con sensores ultrasónicos,

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula abordando desde el diseño de algoritmos hasta la implementación práctica.

Uno de los pilares esenciales en este tipo de programación es el uso adecuado de variables. Las variables permiten almacenar y manipular datos cruciales, como la distancia detectada por el sensor, umbrales de seguridad o estados del sistema. Comprender su funcionamiento y saber cuándo y cómo utilizarlas no solo mejora el rendimiento del programa, sino que también permite una toma de decisiones más precisa. Además, el uso de procedimientos y funciones modulariza el código, promoviendo la reutilización y facilitando la depuración, especialmente en proyectos complejos donde se requieren múltiples lecturas del sensor o respuestas condicionadas.

Por otro lado, el control del flujo del programa a través de sentencias condicionales cobra especial relevancia en la programación con sensores ultrasónicos. Estas estructuras permiten que el sistema actúe de manera inteligente, evaluando condiciones específicas -como si un objeto se encuentra a una determinada distancia- y ejecutando diferentes bloques de código según la situación. La correcta aplicación de condicionales, junto con una programación estructurada, variables bien definidas y funciones claras, permite diseñar soluciones eficientes para una amplia gama de problemas prácticos. A lo largo del capítulo, se mostrarán ejemplos y buenas prácticas para aplicar estos conceptos de forma efectiva en proyectos reales.

Es importante indicar que los contenidos desarrollados en este capítulo permiten abordar tres áreas clave de la metodología STEAM: Tecnología, Ingeniería y Matemática. Mediante la enseñanza aprendizaje de pensamiento computacional condicional basado en sensores ultrasónicos se pueden desarrollar competencias de Tecnología, Ingeniería y Matemática.

En el área de la tecnología: La enseñanza de la programación condicional en *mBlock* combinada con circuitos electrónicos,

como *Arduino*, ofrece a los niños una experiencia educativa única que va más allá del simple código. Al integrar el mundo digital con el físico, los estudiantes no solo aprenden a pensar lógicamente y a resolver problemas, sino que también adquieren habilidades prácticas en electrónica, control de dispositivos y diseño de sistemas. De esta forma se desarrollan competencias tecnológicas a través de proyectos que desarrollan también el pensamiento computacional (Breda et al., 2023).

Utilizando bloques de programación condicional, los niños pueden interactuar directamente con sensores, motores y otros componentes electrónicos, lo que les permite comprender cómo el software puede influir en el mundo real. Esta combinación fomenta la creatividad, el pensamiento crítico y la capacidad de innovación, al tiempo que prepara a los estudiantes para el futuro en campos como la robótica, la automatización y el Internet de las Cosas. Las competencias tecnológicas específicas que se pueden desarrollar se mencionan a continuación:

Conexión entre software y hardware. Los niños aprenden a vincular el mundo digital (la programación) con el mundo físico (los componentes electrónicos). Esto les enseña cómo interactúan los programas con el hardware, controlando cosas como luces, sensores, motores y otros dispositivos electrónicos mediante condicionales en el código.

Control de dispositivos físicos. Al utilizar *Arduino*, los niños aprenden a usar la programación para controlar dispositivos reales, como encender o apagar LEDs, activar motores o leer sensores. Por ejemplo, pueden programar un LED para que se encienda si un sensor de temperatura alcanza un valor específico, lo que es un ejemplo práctico de una condición.

Conceptos de electrónica básica. Al trabajar con *Arduino*, los niños adquieren nociones fundamentales de electrónica, como el uso de resistencias, transistores, circuitos y sensores. Entienden cómo se conecta un circuito y cómo los valores de entrada (como

la lectura de un sensor) afectan las salidas (como prender una luz o activar una alarma).

Pensamiento computacional aplicado al mundo físico. La programación condicional en un entorno físico como *Arduino* refuerza el pensamiento computacional, ya que los niños deben considerar cómo la información del mundo real (por ejemplo, el valor de un sensor) puede influir en las decisiones dentro de un programa y generar una acción real (como un motor que se mueve o una pantalla que muestra información).

Integración de sensores y actuadores. Los niños aprenden cómo funcionan los sensores (como el sensor de luz, temperatura o distancia) y cómo los actuadores (motores, LEDs, zumbadores, etc.) responden a las condiciones establecidas en el código. Esto les da una comprensión profunda de cómo los sistemas ciberfísicos interactúan y se automatizan.

Desarrollo de habilidades en diseño de sistemas. Al programar *Arduino*, los niños deben considerar tanto el diseño del circuito como el del software. Esto fomenta habilidades en el diseño de sistemas, donde entienden cómo se conectan diferentes partes de un sistema (hardware y software) para lograr una solución completa.

Experimentación y aprendizaje mediante prueba y error. Al trabajar con *Arduino*, los niños tienen la oportunidad de experimentar con diferentes circuitos y configuraciones. Aprenden a modificar sus programas y circuitos basándose en los resultados que obtienen, lo que refuerza la habilidad de aprender mediante la experimentación, la iteración y la solución de problemas prácticos.

Creatividad e innovación. Combinar programación condicional con *Arduino* fomenta la creatividad, ya que los niños pueden idear y construir proyectos interactivos y útiles. Pueden diseñar proyectos como sistemas de alarma, robots sencillos o dispositivos controlados por temperatura, lo que los anima a

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
pensar fuera de lo convencional y aplicar sus conocimientos de una manera innovadora.

Conceptos de automatización y control. Al usar condicionales para activar ciertos componentes en función de condiciones específicas (por ejemplo, "si la temperatura es mayor que 30°C, entonces enciende el ventilador"), los niños aprenden sobre conceptos clave de la automatización y el control, habilidades esenciales en áreas como la robótica, la domótica y el Internet de las Cosas (IoT).

Fomento de la perseverancia y resolución de problemas técnicos. Trabajar con hardware y software puede involucrar desafíos técnicos que requieren que los niños sean persistentes, ajusten su código, verifiquen sus conexiones o solucionen errores de hardware. Esto les enseña habilidades de depuración y perseverancia, ya que no siempre los circuitos y programas funcionan correctamente a la primera.

Para finalizar, el análisis de estas competencias podríamos indicar que al combinar la programación condicional con *Arduino*, los niños no solo desarrollan competencias de programación y lógica, sino también una comprensión sólida de cómo el software puede controlar y automatizar el hardware, lo que abre puertas a un mundo de posibilidades creativas y técnicas en la electrónica, la robótica y la automatización (ver figura 5).

Figura 5

Enseñanza de tecnología relacionada a circuitos basados en sensor ultrasónico



Nota. En esta figura se observan datos sobre la enseñanza de tecnología relacionada a circuitos con sensor ultrasónico.

En el área de la ingeniería: la enseñanza de la programación condicional combinada con circuitos electrónicos basados en *Arduino* permite a los niños desarrollar competencias clave en el proceso de ingeniería. A través de actividades prácticas, los estudiantes aprenden a aplicar un enfoque estructurado para resolver problemas, desde la definición del problema hasta la implementación y mejora de soluciones STEAM (Herrero et al., 2023).

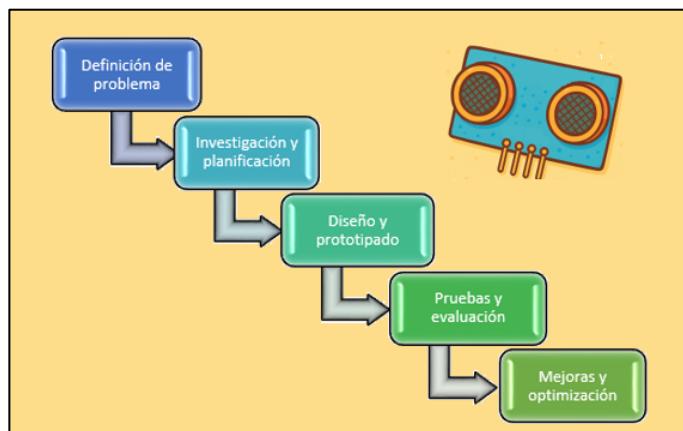
Este enfoque fomenta habilidades como el diseño de sistemas, donde los niños identifican y seleccionan componentes adecuados (sensores, motores, etc.) para cumplir con los requisitos del proyecto. Además, experimentan el proceso iterativo de prototipado y prueba, donde crean modelos iniciales, evalúan su funcionamiento y ajustan tanto el hardware como el software en función de los resultados obtenidos. El análisis y la

depuración del código y los circuitos es una parte esencial, ya que permite a los estudiantes encontrar y corregir fallos, promoviendo la comprensión de las fases de prueba y optimización típicas en ingeniería.

Finalmente, la gestión de proyectos también se ve reflejada (ver figura 6), ya que los niños deben organizar su trabajo, distribuir tareas y seguir una secuencia lógica para alcanzar los objetivos, simulando los desafíos que enfrentan los ingenieros en la vida real. Este enfoque en el proceso de ingeniería fomenta el pensamiento crítico, la resolución de problemas y la capacidad de trabajar de manera sistemática, habilidades fundamentales en cualquier campo de la ingeniería.

Figura 6

Enseñanza de procesos de ingeniería a niños



Nota. En esta figura se observa un proceso de ingeniería para enseñanza a niños.

En el área de Matemática: La enseñanza de la programación condicional combinada con circuitos electrónicos basados en sensores ultrasónicos y otros componentes, no solo fomenta el aprendizaje de conceptos tecnológicos, sino que también desarrolla competencias matemáticas clave. A través de estas actividades, los niños adquieren habilidades prácticas en

geometría, álgebra, cálculo de velocidad y funciones matemáticas. Al integrar estas herramientas tecnológicas con el conocimiento matemático, los estudiantes no solo resuelven problemas de manera creativa, sino que también entienden cómo las matemáticas se aplican en situaciones del mundo real, especialmente en el diseño de sistemas interactivos y automatizados. A continuación, se detallan algunas competencias específicas en esta área de Matemática.

Geometría y medidas espaciales. Los niños aprenden a medir distancias de forma precisa utilizando el sensor ultrasónico, lo que les permite aplicar y reforzar conceptos geométricos clave como la distancia entre puntos, la ubicación en el espacio y la medición de ángulos. Esta experiencia práctica promueve una comprensión más profunda de la geometría a través de la exploración activa y el pensamiento espacial.

Álgebra y operaciones aritméticas. A través de la programación condicional, los niños aplican operaciones algebraicas y aritméticas en tiempo real. Utilizan sumas, restas, multiplicaciones y divisiones para manipular los valores de los sensores y ajustar su comportamiento. Este proceso refuerza su comprensión de las variables, las expresiones matemáticas y la forma en que los datos numéricos influyen en el comportamiento de los circuitos electrónicos.

Cálculos de velocidad y tiempo. Al usar sensores ultrasónicos, los niños pueden calcular la velocidad de los objetos o el tiempo que tardan en recorrer una distancia. A través de fórmulas como la fórmula de la distancia, los estudiantes aprenden a relacionar la distancia, el tiempo y la velocidad. Estos cálculos les permiten entender cómo se pueden modelar fenómenos físicos reales mediante herramientas matemáticas simples, favoreciendo el aprendizaje de conceptos de física aplicada.

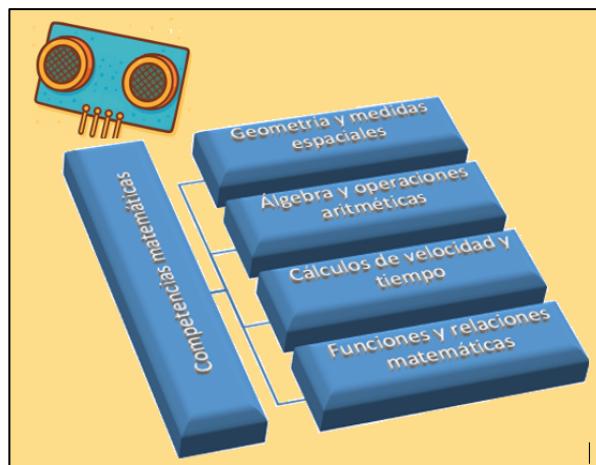
Funciones y relaciones matemáticas. Los niños desarrollan la capacidad de reconocer y aplicar funciones matemáticas al programar la respuesta de las mediciones de distancia de los

sensores ultrasónicos. Aprenden a conectar la entrada (distancia) con la salida (movimiento), lo que les permite comprender cómo se generan decisiones automáticas basadas en relaciones matemáticas. Esta habilidad es esencial en programación y diseño de sistemas cibernéticos.

Analizando estas competencias podemos concluir que al integrar la programación condicional con circuitos que utilizan sensores ultrasónicos y otros componentes, los niños desarrollan competencias matemáticas clave como la geometría, el álgebra, la trigonometría y la estadística (ver figura 7). A través de la programación y la electrónica, los conceptos matemáticos cobran vida en aplicaciones prácticas, lo que no solo refuerza el aprendizaje de las matemáticas, sino que también ayuda a los estudiantes a entender su relevancia en el mundo real.

Figura 7

Enseñanza de Matemática con pensamiento computacional



Nota. En esta figura se observan competencias clave de la enseñanza de Matemática con pensamiento computacional.

Los temas tratados en este capítulo abordan tres áreas fundamentales de la metodología STEAM: Tecnología, Ingeniería y Matemática (ver figura 8). A través del aprendizaje de pensamiento computacional con circuitos electrónicos basados en sensores ultrasónicos, es posible desarrollar competencias de Tecnología, Ingeniería y Matemática de forma innovadora y creativa.

Figura 8

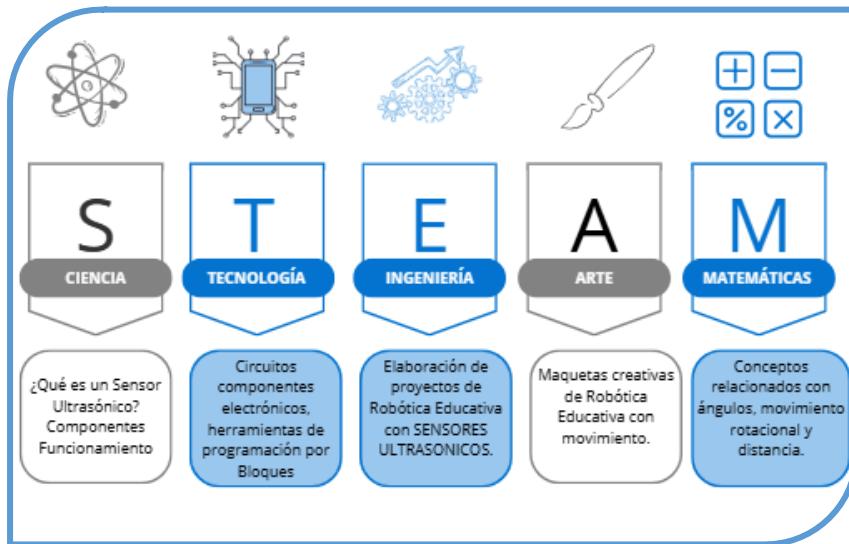


Gráfico de aprendizaje de las áreas STEAM

Nota. En esta figura se observa las áreas de aprendizaje STEAM que el futuro docente debe lograr.

Los aprendizajes mencionados en la anterior figura giran en torno al proceso de ingeniería que se le puede enseñar a un niño para desarrollar un programa en **mBlock** con estructuras condicionales puede ser sencillo y adaptado a su nivel de comprensión. A continuación, se describen las principales fases o pasos de este proceso de manera estructurada, para ello, se combina el sensor ultrasónico con un servomotor ya que el sensor necesita interactuar con otros componentes.

Fase 1. Definir el Problema

Objetivo de la Fase: Determinar qué se quiere lograr con el sensor ultrasónico HC-SR04, es decir que hará el circuito electrónico una vez que el sensor detecte un objeto a una determinada distancia.

Explicación: Antes de programar, es importante definir el propósito del proyecto. En este caso, el niño debe reflexionar sobre cómo el sensor ultrasónico puede detectar un objeto a una determinada distancia para luego interactuar con otros componentes electrónicos, por ejemplo, mover un servomotor.

Actividades educativas. El futuro docente puede dar inicio haciendo varias preguntas a los niños: ¿Qué quieres que haga el servomotor cuando detecte un objeto? ¿Quieres que se mueva rápida o lentamente? ¿A qué distancia debe activarse? ¿Debe moverse en una dirección específica?

Fase 2. Planificar la Solución

Objetivo de la Fase: Desglosar el problema en pequeños pasos para facilitar su implementación.

Explicación: Se define cómo se conectarán los componentes y cuáles serán sus interacciones.

Actividades educativas: Realizar una lista de las acciones que debe realizar el sistema. Primero leer la distancia con el sensor ultrasónico, luego comparar la distancia con un valor límite (ej. 20 cm), si la distancia es menor al límite, mover el servomotor a 90°, en cambio, si la distancia es mayor, volver a 0°, y al finalizar Repetir el proceso continuamente.

Fase 3. Diseñar el Algoritmo

Objetivo de la Fase: Crear una secuencia clara de pasos a seguir.

Explicación: Se organiza el funcionamiento del sistema en instrucciones lógicas.

Actividades educativas. Realizar una representación gráfica de un diagrama de flujo o un pseudocódigo para representar el siguiente programa:

Iniciar.

Leer la distancia con el sensor HC-SR04.

Si la distancia < 20 cm:

Mover el servomotor a 90° .

Si la distancia ≥ 20 cm:

Mover el servomotor a 0° .

Repetir desde el paso 2.

Fase 4. Implementar el Código en *mBlock*

Objetivo de la Frase: Traducir el algoritmo a bloques de programación.

Explicación: Una vez listo el algoritmo, se usa *mBlock* para convertir la lógica en código ejecutable. Al ser un programa por bloques gráficos es más fácil su explicación para que los niños entiendan la programación paso a paso.

Actividades educativas. Arrastrar y conectar bloques que representen el siguiente código:

Leer la distancia del sensor.

Comparar la distancia con el umbral.

Mover el servomotor a la posición correcta.

Usar un bucle para repetir el proceso continuamente.

Fase 5. Probar el Programa

Objetivo de la Fase: Verificar el funcionamiento esperado del sistema.

Explicación: Se ejecuta el programa y se observa si el servomotor responde correctamente a la distancia detectada.

Actividades educativas. Realizar las acciones que se detallan a continuación:

Ejecutar el programa y probar con diferentes distancias.

Verificar si el servomotor responde correctamente.

Anotar posibles errores.

Ejemplo: Si el servomotor no se mueve correctamente, comprobar si el sensor está midiendo bien la distancia.

Fase 6. Depurar y Mejorar el Código

Objetivo de la Fase: Corregir errores y optimizar el funcionamiento.

Explicación: Luego de poner el programa en funcionamiento, se puede llegar a encontrar errores, para aquello es necesario que se realicen ajustes para mejorar el programa.

Actividades educativas. Realizar las acciones que se detallan a continuación:

Revisar conexiones y lógica del código.

Agregar comentarios para mayor claridad.

Probar mejoras, como minimizar el movimiento del servomotor o la distancia del sensor.

Ejemplo: Si el servomotor se mueve demasiado rápido, agregar una pausa para mejorar la respuesta.

2.4. Organización de la programación de sensores ultrasónicos con *mBlock*

La organización de la programación en *mBlock* se caracteriza por su enfoque visual e intuitivo, diseñado para facilitar el aprendizaje de la programación desde los niveles más básicos hasta proyectos más complejos. Su interfaz está dividida en diferentes áreas que guían al usuario durante el proceso de creación del programa, permitiendo mantener una estructura lógica y ordenada. En el área principal se encuentra el escenario o espacio de trabajo, donde se ensamblan los bloques para formar secuencias de instrucciones. A la izquierda se sitúa el panel de bloques, clasificados por colores y funciones específicas, mientras que en la parte superior se encuentran las opciones de control del proyecto, tales como guardar, cargar, cambiar de modo (bloques o código) y conectarse a dispositivos físicos.

Los bloques de *mBlock* están organizados en categorías, cada una representada por un color y enfocada en una función particular. Por ejemplo, los bloques de Movimiento (azules) controlan el desplazamiento de personajes en pantalla; los bloques de Apariencia (rosados) permiten modificar el aspecto, textos o sonidos; y los bloques de Sonido (morados) se usan para reproducir efectos auditivos. Los bloques de Eventos (amarillos) permiten iniciar secuencias cuando ocurre una acción específica, como presionar una tecla o al hacer clic en un objeto.

Una categoría especialmente importante para el trabajo con hardware y sensores ultrasónicos es la de Extensiones, donde se encuentran bloques relacionados con placas como *Arduino*, *micro:bit* o *mBot*. Aquí se pueden utilizar bloques para controlar sensores ultrasónicos y otros dispositivos físicos. Además, los bloques de Control (naranjas) permiten establecer estructuras condicionales, repeticiones y esperas; los Operadores (verdes) realizan cálculos matemáticos y lógicos; y los bloques de Variables y Mis bloques (rojos y vino) permiten almacenar datos y crear funciones personalizadas, respectivamente.

Gracias a esta organización por áreas y colores, *mBlock* facilita que los estudiantes comprendan la lógica detrás de un programa, reconozcan los elementos que deben usar para lograr un objetivo y mantengan su código bien estructurado. Esta organización también promueve el pensamiento computacional al permitir que el usuario divida el problema en partes, pruebe distintas soluciones y modifique fácilmente los bloques según sea necesario. En conjunto, *mBlock* no solo enseña a programar, sino que entrena a pensar como programador.

2.5. La importancia y funcionamiento de las variables en programación de sensores ultrasónicos con *mBlock*

En *mBlock*, las variables son esenciales para almacenar y manejar datos durante la ejecución de un programa. Permiten guardar números, textos u otros valores que pueden cambiar a lo largo del tiempo, lo que brinda flexibilidad y control sobre las acciones del personaje o dispositivo. Gracias a las variables, los estudiantes pueden realizar operaciones matemáticas, llevar conteos, controlar repeticiones o tomar decisiones según condiciones específicas. Su uso facilita la comprensión de conceptos fundamentales de la programación, como entrada, procesamiento y salida de datos, fomentando el pensamiento lógico y la resolución de problemas de manera estructurada.

En la programación con *mBlock*, las variables juegan un papel fundamental al trabajar con sensores ultrasónicos, ya que

permiten almacenar, manipular y reutilizar los datos que estos sensores capturan. Una de las funciones principales del sensor ultrasónico es medir la distancia entre el dispositivo y un objeto. Esta medición se guarda en una variable, lo que facilita su uso en distintas partes del programa, ya sea para mostrar la información en pantalla, compararla con valores de referencia o activar acciones específicas.

El funcionamiento de las variables en *mBlock* es sencillo e intuitivo, lo que lo convierte en una excelente herramienta educativa para introducir a los estudiantes en la lógica de la programación. Las variables pueden crearse fácilmente desde el entorno gráfico y asignarse mediante bloques como "establecer variable a" o "cambiar variable por". Esto permite trabajar con valores dinámicos, como las mediciones del sensor, y realizar cálculos o condiciones basadas en esos datos en tiempo real.

Además, el uso adecuado de variables mejora la organización del programa y facilita su mantenimiento. Por ejemplo, si se desea cambiar el umbral de distancia para una acción determinada (como evitar un obstáculo), solo se necesita modificar el valor almacenado en una variable, sin tener que reescribir múltiples bloques. Esta práctica no solo optimiza el rendimiento del código, sino que también promueve una forma de pensar más estructurada y eficiente en el desarrollo de soluciones con sensores ultrasónicos.

2.6. El rol de la programación de procedimientos y funciones en la resolución de problemas con sensores ultrasónicos en *mBlock*

La programación de procedimientos y funciones en *mBlock* cumple un papel clave en la organización y simplificación del código, especialmente cuando se desarrollan proyectos más complejos. En este entorno visual, los procedimientos se crean dentro de la categoría "Mis bloques", donde el usuario puede diseñar bloques personalizados que agrupan una serie de instrucciones bajo un solo nombre. Esta herramienta permite

reutilizar secuencias de código en distintas partes del programa sin tener que repetir los mismos bloques, lo que mejora la legibilidad y facilita la depuración.

La programación de procedimientos y funciones en *mBlock* permite organizar el código de forma más clara y eficiente, especialmente al trabajar con sensores ultrasónicos. Al crear bloques personalizados que agrupan una serie de instrucciones, es posible reutilizar ese conjunto de acciones en diferentes partes del programa sin tener que escribir el mismo código repetidamente. Esto resulta muy útil, por ejemplo, cuando se necesita leer la distancia del sensor en varios momentos o reaccionar ante obstáculos de forma consistente. Definir funciones específicas para estas tareas mejora la legibilidad y facilita la depuración del programa.

Además, el uso de procedimientos y funciones promueve una forma de pensar modular, que es clave para resolver problemas complejos. En lugar de abordar todo el sistema de forma lineal, los estudiantes pueden dividir un desafío en pequeñas partes manejables: una función para medir la distancia, otra para decidir una acción según esa medida, y otra para ejecutar la respuesta. Esta estrategia no solo hace que el código sea más limpio y ordenado, sino que también ayuda a los alumnos a desarrollar habilidades de planificación, análisis y diseño lógico que son fundamentales en la programación y en la solución de problemas del mundo real.

En el contexto educativo, enseñar a los estudiantes a utilizar procedimientos y funciones en *mBlock* con sensores ultrasónicos fortalece su capacidad para resolver problemas de manera estructurada y eficiente. Por ejemplo, pueden crear una función que lea la distancia detectada por el sensor y, según ese valor, ejecute una acción determinada, como hacer que otro componente electrónico actúe. Al encapsular este tipo de lógica en bloques personalizados, los estudiantes no solo simplifican su código, sino que también comprenden mejor cómo reutilizar y adaptar sus soluciones a diferentes escenarios. Esta práctica

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula fomenta una programación más ordenada y les brinda herramientas esenciales para enfrentar desafíos más avanzados en el futuro, tanto dentro como fuera del entorno de *mBlock*.

2.7. Sentencias de control condicionales para sensores ultrasónicos con *mBlock*

Las sentencias secuenciales en el pensamiento computacional

Las estructuras condicionales en programación permiten que el flujo de ejecución de un programa tome decisiones basadas en condiciones específicas. Estas estructuras se utilizan para controlar el comportamiento del programa, determinando qué acciones deben ejecutarse dependiendo de si una condición es verdadera o falsa.

En el contexto del pensamiento computacional, las estructuras condicionales son fundamentales, ya que proporcionan flexibilidad y permiten que los programas respondan de manera dinámica a diferentes situaciones.

Al aplicar estructuras condicionales con un sensor ultrasónico, el programa puede realizar decisiones basadas en las mediciones de distancia obtenidas. Por ejemplo, si el sensor detecta que un objeto está dentro de un rango específico, el programa puede activar una acción, como encender una luz o mover un motor. Esto se logra mediante instrucciones como "si la distancia medida es menor a un valor determinado, entonces ejecutar acción X". Así, el sensor ultrasónico, al medir la distancia a un objeto, permite que el programa tome decisiones basadas en el entorno y reaccione de forma adecuada, lo que agrega interactividad y adaptabilidad al sistema.

Este enfoque permite que el programa no solo siga una secuencia lineal de instrucciones, sino que también tome decisiones inteligentes en función de las condiciones observadas, lo que es un aspecto clave del pensamiento computacional en la resolución de problemas.

Sintaxis básica de un programa condicional

Un diagrama o programa condicional es un concepto clave en la programación, ya que permite que un programa tome decisiones según ciertas condiciones.

A continuación, se presentan algunos ejemplos de programas condicionales.

Ejemplo 1: Preparar un Sándwich

Imaginemos que queremos dar instrucciones a una persona sobre cómo preparar un sándwich de manera fácil y sencilla paso a paso.

Inicio

Tomar pan

Colocar ingredientes (jamón, queso, lechuga, etc.)

Tapar con otra rebanada de pan

¿Es demasiado grande para comer?

Sí → Cortar el sándwich

No → Seguir al siguiente paso

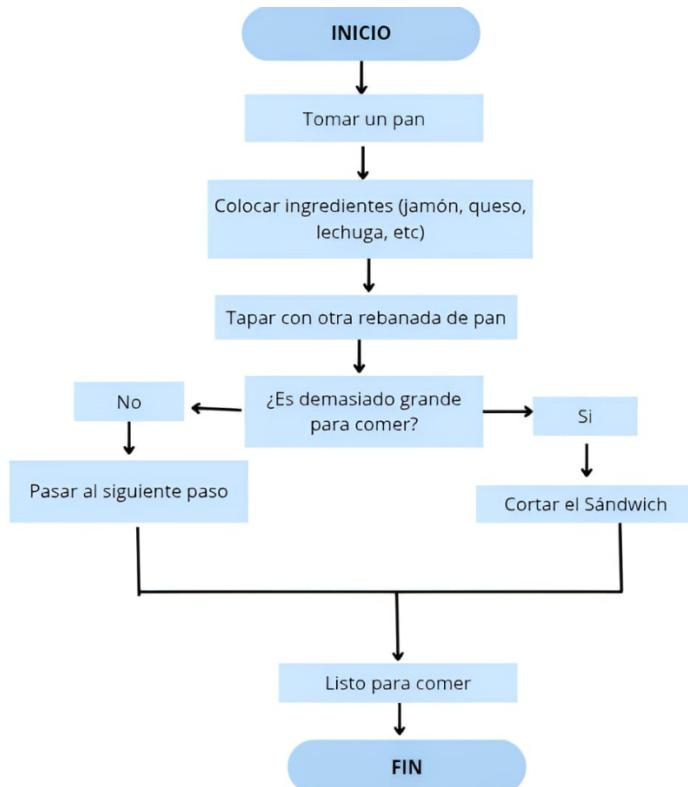
Listo para comer

Fin

A continuación, se muestra una representación gráfica de estos pasos en un programa condicional (ver Figura 9)

Figura 9

Programa para preparar un sándwich



Nota. La figura muestra un programa para preparar un sándwich

Ejemplo 2: Regar una Planta

Este ejemplo muestra cómo seguir el flujo, mientras se toma en cuenta la condición que está presente en el diagrama.

Inicio

Tomar regadera

Llenar con agua

Verter agua en la planta

¿La tierra está húmeda?

No → Agregar más agua y volver a verificar

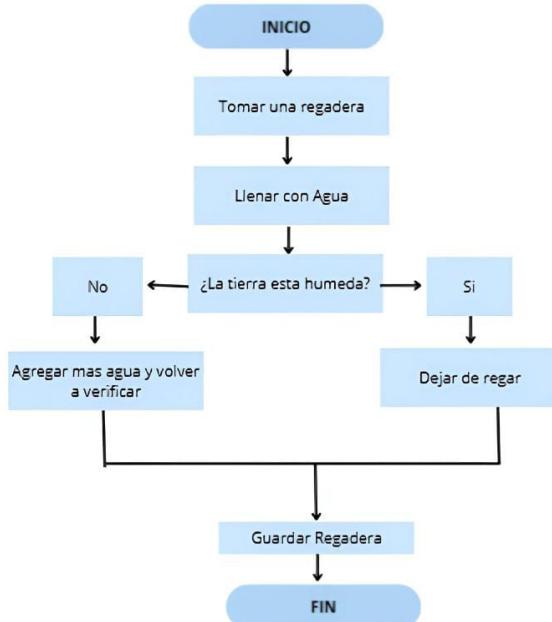
Sí → Dejar de regar

Guardar regadera

Fin

En la Figura 10 se muestra un esquema del flujo condicional de este programa.

Figura 10
Programa de flujo de como regar una planta



Nota. En esta figura se visualizar el programa para regar una planta.

Las estructuras condicionales son fundamentales en la programación, ya que permiten que un programa tome decisiones según ciertas condiciones, modificando su flujo de ejecución de manera dinámica. La comprensión de este concepto es clave en proyectos de robótica educativa y automatización, donde los sistemas deben responder a diferentes situaciones, como detectar obstáculos con sensores ultrasónicos o activar dispositivos según condiciones específicas.

En herramientas como *mBlock*, se pueden programar estructuras condicionales utilizando bloques como "si-entonces" o "si-entonces-sino", lo que permite a los estudiantes visualizar y comprender cómo un programa responde a distintas entradas. A través de estos entornos gráficos, los futuros programadores pueden estructurar programas de manera lógica, fomentando el desarrollo del pensamiento computacional y la resolución de problemas mediante la toma de decisiones en tiempo real.

A continuación, se explicará cómo trabajar condiciones en el lenguaje de programación por bloques *mBlock*, primero haciendo simulaciones con objetos virtuales y luego con robótica educativa específicamente con circuitos electrónicos físicos basados en Sensores Ultrasónicos.

La programación condicional por bloques utilizando el sensor ultrasónico

Caracterización de *mBlock*

mBlock es una plataforma de programación visual basada en bloques, diseñada para enseñar conceptos de programación a niños y principiantes en programación. Está basada en *Scratch* y permite crear proyectos interactivos además de programar robots, dispositivos electrónicos y placas como el *Arduino*, utiliza

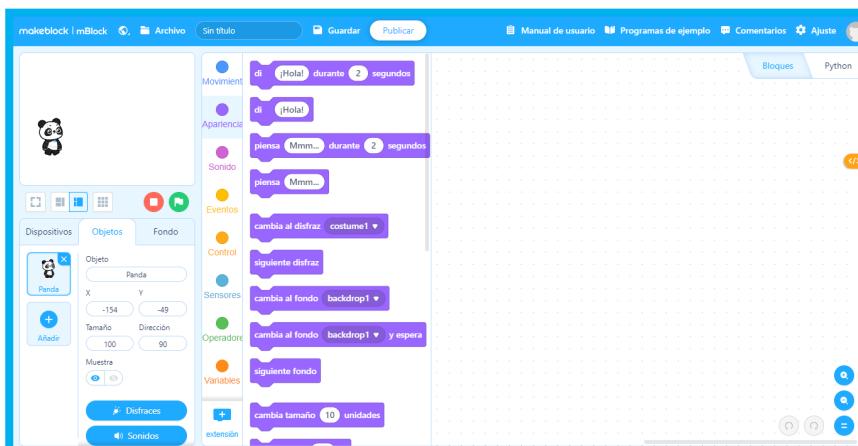
Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula una interfaz intuitiva que facilita la comprensión de la lógica de programación mediante bloques gráficos en lugar de códigos escritos.

El entorno de programación de *mBlock*

Es un entorno de programación visual y fácil de usar que emplea bloques gráficos para representar comandos. Los usuarios pueden arrastrar y enlazar bloques para crear programas, lo que hace más sencillo el aprendizaje de la lógica y las estructuras de programación. También incluye herramientas para manejar robots y dispositivos interactivos. El entorno de *mBlock* se presenta en la siguiente imagen (ver figura 11), donde se destacan las áreas más relevantes de esta herramienta de programación por bloques. El entorno se compone principalmente de un Escenario en el que los diversos Objetos (conocidos como Sprites en inglés) interactuarán y se desarrollarán.

Figura 11

Entorno de programación por bloques de *mBlock*

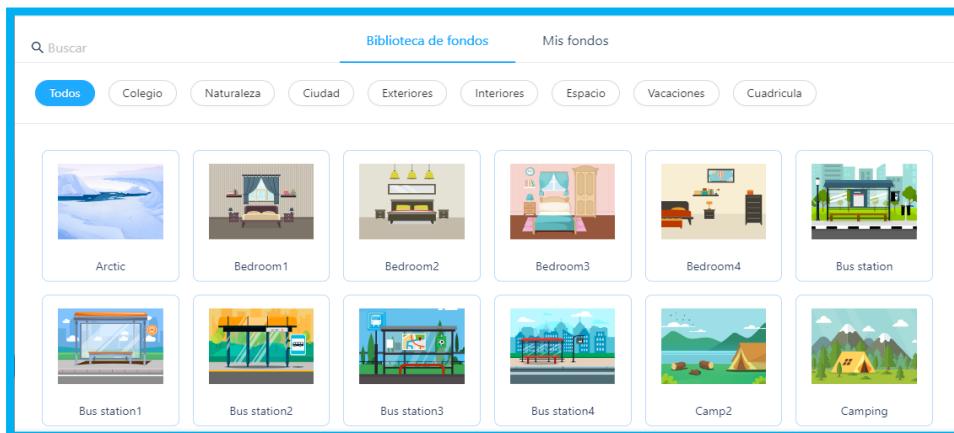


Nota. La figura muestra el Entorno de programación por bloques de *mBlock*. Fuente: Captura de pantalla tomada de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Implementar un escenario

Para dar comienzo y armar escenario primero se debería elegir un Fondo (ver figura 12), este se puede modificar durante la ejecución, *mBlock* tiene una colección de fondos que se pueden utilizar para realizar la animación o el programa. Sin embargo, es importante recordar y recalcar que podemos crear nuestro propio fondo para el escenario e incluso añadir una imagen guardada en nuestro dispositivo.

Figura 12
Fondos disponibles en mBlock



Nota. La figura muestra Fondos disponibles en *mBlock*. Fuente: Captura de pantalla tomada de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Una segunda parte esencial de un escenario son los Objetos, donde se puedes representar distintas representaciones graficas, como los diversos personajes, Motocicletas, plantas, etc. en el escenario virtual de *mBlock*. A los objetos se le puede asignar interacciones para desarrollar lo que conocemos como un programa. Estos pueden aparecer en el escenario con múltiples

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula disfraces, lo que nos permite asociar y definir acciones según las distintas formas de presentación del objeto. En la pestaña Objetos (ver figura 13) podemos empezar el trabajo con diversos personajes en el entorno de programación de *mBlock*.

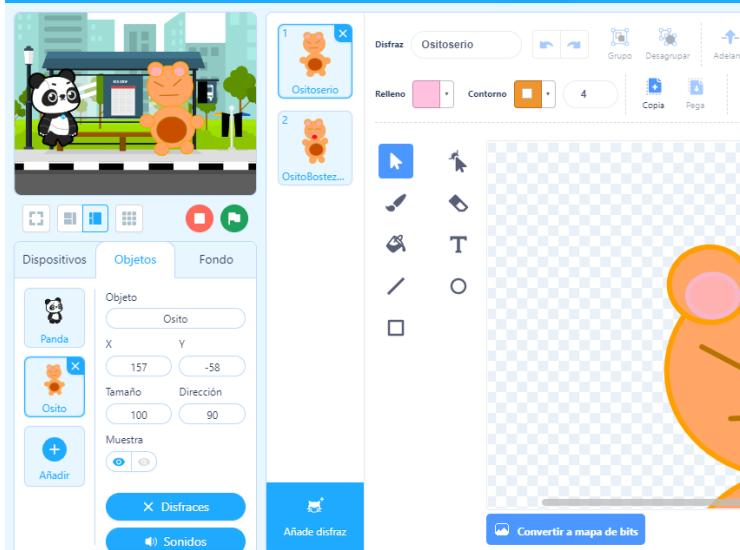
Figura 13
Pestaña Objetos en *mBlock*



Nota. La figura muestra la pestaña Objetos en *mBlock*. Fuente: Captura de pantalla tomada de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

La figura 14 muestra la creación del objeto dibujado, y del cambio del fondo, donde ambos se complementan en la pantalla o escenario. A los escenarios se les debe colocar un nombre apropiado para luego usar esos nombres en la programación por bloques. En este caso, a los escenarios, se les ha colocado como nombres *Ositoserio* y *OsitoBostezando*.

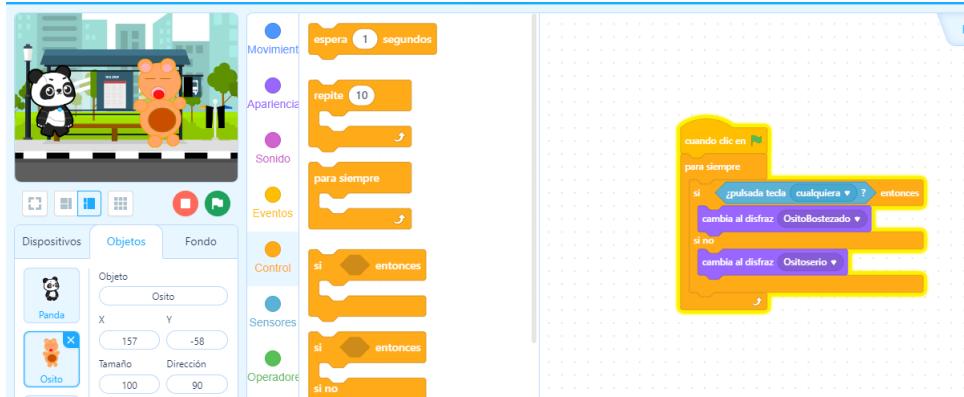
Figura 14
Creación de objeto en mBlock



Nota. La figura muestra la creación de objetos en mBlock. Fuente: Captura de pantalla tomada de la interfaz de mBlock v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Una vez que tenemos listo, todo lo que van dentro de la pantalla del escenario, se comienza a programar estructuras condicionales sobre dicho escenario. En este caso vamos a simular cambio de movimiento con condiciones (ver figura 15).

Figura 15
Simulación en mBlock



Nota. La figura muestra una simulación en mBlock. Fuente: Captura de pantalla tomada de la interfaz de mBlock v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

2.7. El hardware necesario para trabajar robótica educativa basada en sensores ultrasónicos con mBlock

Descripción de un sensor ultrasónico

Un sensor ultrasónico es un dispositivo que mide distancias mediante la emisión y recepción de ondas sonoras de alta frecuencia, imperceptibles para el oído humano. (Fuentes, 2014). Estos funcionan enviando un pulso ultrasónico desde su cabezal (ver figura 16), que rebota en un objeto y regresa al sensor. La distancia se calcula midiendo el tiempo que transcurre entre la emisión y la recepción de la onda reflejada.

Figura 16

Emisión de ondas desde sensor ultrasónico HC-



SR04

Nota. La figura muestra la Emisión de ondas desde sensor ultrasónico. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Estructura de un sensor ultrasónico.

El Sensor ultrasónico está compuesto por un transmisor, que emite pulsos ultrasónicos, y un receptor, que detectar el eco reflejado y lo convierte en una señal eléctrica. Incluye un circuito de control que procesa la información y calcula la distancia basándose en el tiempo de viaje de las ondas.

La mayoría de los sensores cuentan con pines de conexión, como por ejemplo el Sensor ultrasónico HC-SR04, cuentan con VCC, que suministra la alimentación de 5V; GND, que funciona como conexión a tierra; Trig, que activa la emisión de pulso ultrasónicos; y Echo, que envía la señal de respuesta cuando el eco es detectado (ver figura 17).

Figura 17
Sensor ultrasónico HC-SR04



Nota. La figura muestra el diseño y partes del sensor ultrasónico HC-SR04. Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

El sensor ultrasónico PING))) tiene una estructura compacta y se diferencia de otros modelos al contar con solo tres pines de conexión que son: GND, sirve como conexión a tierra del circuito; 5V, que proporciona la alimentación necesaria para su funcionamiento; y SIG, que se encarga tanto de activar la emisión del pulso ultrasónico como de recibir la señal de eco reflejada (ver figura 18).

Figura 18
Sensor ultrasónico PING)))

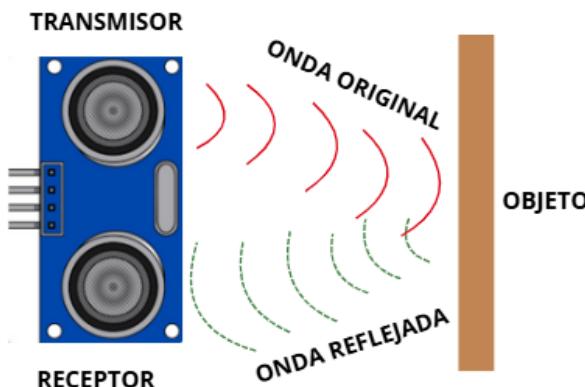


Nota. En esta figura se visualiza el diseño del Sensor PING))). Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

Funcionamiento de un sensor ultrasónico

Un sensor ultrasónico funciona emitiendo ondas de sonido de alta frecuencia que no podemos oír, las cuales viajan hasta encontrar un objeto y luego regresan al sensor como un eco. Al medir el tiempo que tarda en recibir ese eco, el sensor puede calcular la distancia entre él y el objeto (ver figura 19).

Figura 19
Funcionamiento del Sensor ultrasónico



Nota. La figura muestra el reflejo de la onda, para su funcionamiento. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

El funcionamiento del sensor ultrasónico se basa en el principio de propagación de ondas acústicas, similares al sonar utilizado por los submarinos o los radares utilizados en la aviación, pero en lugar de usar ondas electromagnéticas, el sensor ultrasónico emite ondas sonoras de frecuencia muy alta.

Proceso de Funcionamiento del Sensor Ultrasónico:

1. **Emisión de Ondas Ultrasónicas:** El sensor consta de un transductor que tiene dos funciones: emitir y recibir ondas ultrasónicas. El transductor emisor genera pulsos de sonido de alta frecuencia, generalmente por encima de 20 kHz (frecuencias que están fuera del rango audible para los humanos). Estas ondas sonoras viajan a través del aire a la velocidad del sonido, que varía dependiendo de la temperatura y la presión del aire. El sensor emite estos pulsos de forma periódica, generalmente en ráfagas rápidas.
2. **Propagación de las Ondas:** Una vez que las ondas ultrasónicas son emitidas, viajan a través del aire en línea recta hasta que encuentran un obstáculo o superficie. El comportamiento de las ondas ultrasónicas puede verse afectado por las condiciones del medio a través del cual viajan, como la temperatura del aire, la humedad, o la presión atmosférica, ya que la velocidad del sonido varía con estos factores.
3. **Reflexión de las Ondas en el Objeto:** Cuando las ondas ultrasónicas encuentran un objeto, se reflejan de vuelta hacia el sensor. La cantidad de ondas reflejadas depende de las características del objeto, como su tamaño, forma y textura. Las superficies duras, planas y lisas reflejan más ondas, mientras que las superficies absorbentes o blandas reflejan menos.
4. **Recepción del Eco:** El transductor receptor del sensor detecta el eco de las ondas reflejadas. El receptor está diseñado para captar las vibraciones causadas por las ondas de ultrasonido que regresan al sensor. Este receptor mide el tiempo exacto que transcurre entre la emisión del

pulso y la recepción del eco, es decir, el tiempo que tardan las ondas en ir al objeto y regresar al sensor.

5. **Cálculo de la Distancia:** Con el tiempo de vuelo (el tiempo que tarda el pulso de ultrasonido en ir hasta el objeto y volver), el sensor puede calcular la distancia entre el sensor y el objeto. Para hacer este cálculo, se utiliza la fórmula básica:

$$\text{Distancia} = \frac{\text{Velocidad del sonido} \times \text{Tiempo de vuelo}}{2}$$

La distancia se divide entre 2 porque el tiempo de vuelo corresponde al trayecto de ida y vuelta de las ondas ultrasónicas. Por ejemplo, si el tiempo de vuelo es de 0,01 segundos, y la velocidad del sonido es de 343 metros por segundo (en condiciones estándar), la distancia al objeto sería de:

$$\text{Distancia} = \frac{343 \text{ m/s} \times 0,01 \text{ s}}{2} = 1,715$$

Este cálculo da la distancia en metros, y el sensor puede entonces enviar esta información a un microcontrolador o a otro sistema que la utilice para tomar decisiones, como evitar obstáculos o ajustar la posición de un vehículo.

Factores que Afectan el Funcionamiento del Sensor Ultrasónico:

Aunque el principio de funcionamiento es simple, hay varios factores que pueden influir en la precisión del sensor ultrasónico:

1. **Condiciones Ambientales:** La velocidad del sonido en el aire no es constante, sino que varía dependiendo de la

temperatura, la humedad y la presión del aire. En ambientes muy fríos o calurosos, la velocidad del sonido puede cambiar significativamente, lo que afecta la precisión de la medición. Los sensores avanzados suelen tener compensaciones para estos cambios en sus algoritmos de cálculo.

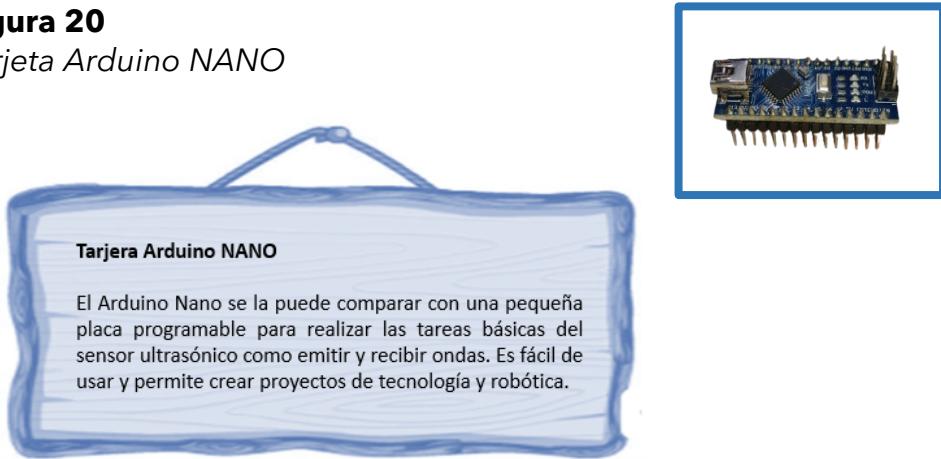
2. **Tipo de Superficie del Objeto:** La forma y el material del objeto que se está midiendo también pueden influir en la precisión. Los objetos blandos o absorbentes, como tela o espuma, no reflejan bien las ondas ultrasónicas, lo que puede llevar a una medición inexacta. Los objetos grandes, planos y sólidos reflejan las ondas de manera más eficiente, lo que resulta en una medición más precisa.
3. **Ángulo de Incidencia:** Si las ondas ultrasónicas golpean una superficie de manera perpendicular (en un ángulo de 90 grados), la reflexión será más fuerte y precisa. Sin embargo, si las ondas inciden en la superficie en un ángulo agudo, las ondas pueden reflejarse en direcciones no deseadas, lo que puede dificultar la recepción del eco y generar lecturas incorrectas.
4. **Rango de Medición:** Los sensores ultrasónicos tienen un rango limitado, dependiendo de la frecuencia de las ondas ultrasónicas y la potencia del transductor. Generalmente, los sensores ultrasónicos tienen un rango de medición de unos pocos centímetros a varios metros. Este rango puede verse afectado por la geometría del objeto y las condiciones ambientales.
5. **Ruido Acústico:** En entornos ruidosos, las ondas ultrasónicas pueden interferir con otras fuentes de sonido, lo que puede dificultar la detección de los ecos. Por esta razón, los sensores ultrasónicos son más efectivos en entornos controlados o donde no haya muchas fuentes de sonido que interfieran con las ondas ultrasónicas.

Componentes electrónicos complementarios

Tarjeta Arduino NANO

La tarjeta *Arduino NANO* es una plataforma de hardware y software de código abierto que facilita la creación de proyectos electrónicos interactivos (ver figura 20). Es una versión compacta de *Arduino*, ideal para proyectos más pequeños o con espacio limitado, funciona como una pequeña computadora programable que puede controlar dispositivos como luces, motores, sensores, y otros componentes electrónicos. La *Arduino NANO* está equipada con un microcontrolador, que actúa como el cerebro de la placa, y pines de entrada y salida (I/O) que permiten conectar diversos componentes electrónicos para su funcionamiento.

Figura 20
Tarjeta Arduino NANO



Nota. La figura muestra un *Arduino NANO*.

El microcontrolador de la tarjeta *Arduino* se programa fácilmente con el software *Arduino IDE*, que te permite escribir y cargar el código necesario para que realice tareas específicas. El lenguaje de programación que utiliza es parecido a C/C++, lo que hace que sea accesible para principiantes. Además, la comunidad de

Arduino es muy activa, ofreciendo una gran cantidad de tutoriales, ejemplos y proyectos listos para usar.

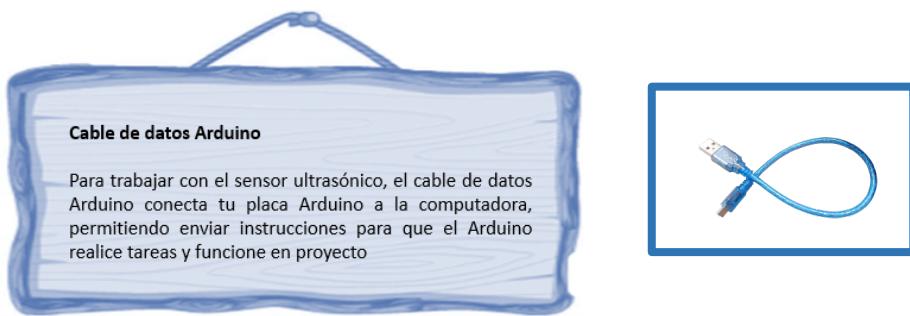
Una de las mayores ventajas de *Arduino* es su simplicidad y flexibilidad. Es perfecta para que los niños y principiantes aprendan sobre electrónica y programación, pero también lo suficientemente potente para que los expertos desarrollen proyectos más complejos. Desde algo tan sencillo como encender un LED hasta la creación de robots y sistemas automatizados, *Arduino* fomenta la creatividad y el aprendizaje práctico en tecnología.

Cable de datos *Arduino NANO*

El cable de datos para *Arduino NANO* permite la conexión entre la placa y un computador. Es un cable USB que se utiliza para cargar programas en la placa y permitir su funcionamiento con distintos componentes electrónicos (ver figura 21).

Figura 21

Cable de datos para Arduino NANO



Nota. La figura muestra un cable de datos para *Arduino NANO*

Características del cable de datos para *Arduino NANO*:

Conecotor USB Tipo A a Mini-USB o Micro-USB:

- La *Arduino NANO* generalmente utiliza un conector Mini-USB o Micro-USB, dependiendo del modelo, para conectar la placa a la computadora.
- El extremo Mini-USB o Micro-USB se conecta al puerto de la tarjeta *Arduino NANO*.
- El extremo USB Tipo A se conecta al puerto USB de la computadora.

Propósito:

- Programación: El cable se utiliza para cargar programas desde el *Arduino IDE* a la placa *Arduino NANO*. Al conectar el *Arduino NANO* a la computadora, se puede transferir código escrito en el entorno de desarrollo *Arduino*.
- Comunicación de datos: Permite que el *Arduino NANO* se comunique con la computadora, lo que es útil para enviar datos de sensores, leer entradas del teclado o de otros dispositivos y procesar resultados en tiempo real.

Conexión y alimentación:

- Además de la transferencia de datos, el cable USB también alimenta el *Arduino NANO* a través del puerto USB de la computadora. Esto significa que no necesitas una fuente de alimentación externa para que el Nano funcione cuando lo conectas por USB.

Compatibilidad:

- Es importante usar un cable que sea compatible con el modelo específico de tu *Arduino NANO*. Algunos modelos antiguos de *Arduino NANO* usan un conector Mini-USB, mientras que las versiones más recientes usan

Micro-USB. Asegúrate de que el cable que adquieras tenga el conector adecuado para tu placa.

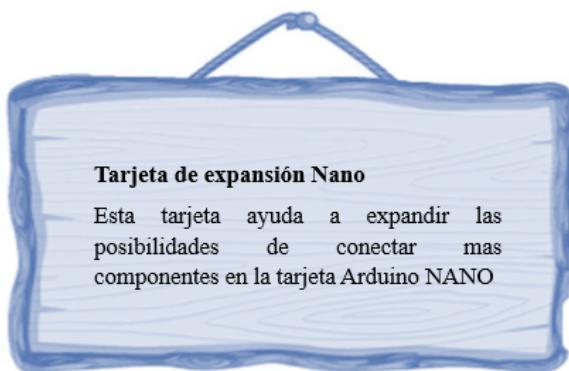
Tarjeta de expansión *Arduino NANO*

Es una tarjeta (de la categoría de los shields) de expansión diseñada para ser compatible con la placa *Arduino NANO* (ver figura 22). Los shields **son** placas adicionales que se pueden montar directamente sobre las placas *Arduino* (como el Nano) para añadirles funcionalidades específicas sin necesidad de usar cables o componentes adicionales dispersos.

Un *Arduino NANO* Shield se conecta a los pines del *Arduino NANO* y expande su capacidad, añadiendo más funcionalidades de manera fácil y rápida. Estos shields pueden incluir una variedad de módulos o componentes, como pantallas LCD, sensores, motores, módulos de comunicación, entre otros. Están diseñados para que encajen perfectamente con la disposición de los pines de la placa Nano, de modo que no sea necesario realizar conexiones complejas con cables.

Figura 22

Tarjeta de expansión para Arduino NANO



Nota. La figura muestra una tarjeta de expansión para Arduino NANO

Cables de conexión Dupont

Los cables Dupont son cables de conexión versátiles y ampliamente utilizados en proyectos de electrónica y robótica (ver figura 23), especialmente cuando se trabaja con plataformas como *Arduino*, *Raspberry Pi*, y otros sistemas de prototipado. Son conocidos por su facilidad de uso y por permitir conexiones rápidas y seguras entre la placa base (como el *Arduino NANO*) y otros componentes electrónicos, como sensores, módulos, y actuadores.

Figura 23

Cables de conexión Dupont para Arduino



Nota. La figura muestra cables de conexión Dupont para Arduino

Diodos LED

El Diodo LED (Light Emitting Diode) es un componente electrónico que emite luz cuando pasa corriente eléctrica a través de él. A diferencia de las bombillas tradicionales, los LEDs son eficientes, de bajo consumo y duran mucho más tiempo. En proyectos con *Arduino*, estos se utilizan comúnmente para mostrar el estado de un circuito o como indicadores visuales en diversas aplicaciones. Son fáciles de controlar y permiten a los usuarios aprender sobre conceptos como la corriente eléctrica y la polaridad (ver figura 24).

Figura 24

Diodos leds



Nota. La figura muestra diodos leds. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Los colores de los LEDs (ver tabla 1) influyen directamente en el voltaje de funcionamiento, lo cual está relacionado con la resistencia necesaria en un circuito. Los LEDs rojos, que requieren entre 1.8V y 2.2V, necesitan menor resistencia, mientras que los LEDs azules y blancos, con voltajes de 3.0V a 3.5V, requieren mayor resistencia. Ajustar la resistencia adecuada garantiza que los LEDs reciban la corriente correcta sin dañarse. La selección de la resistencia depende del voltaje de alimentación y el voltaje del LED, siguiendo la Ley de Ohm para asegurar un funcionamiento seguro y eficiente.

Tabla 1

Colores de LED y su voltaje de funcionamiento

Color del LED	Voltaje de funcionamiento (aproximado)	Corriente típica (mA)
Rojo	1.8V - 2.2V	20-30
Naranja	2.0V - 2.2V	20-30
Amarillo	2.0V - 2.3V	20-30
Verde (normal)	2.0V - 3.0V	20-30

Verde (brillante)	3.0V - 3.2V	20-30
Azul	3.0V - 3.5V	20-30
Blanco	3.0V - 3.5V	20-30
Violeta	3.1V - 3.5V	20-30

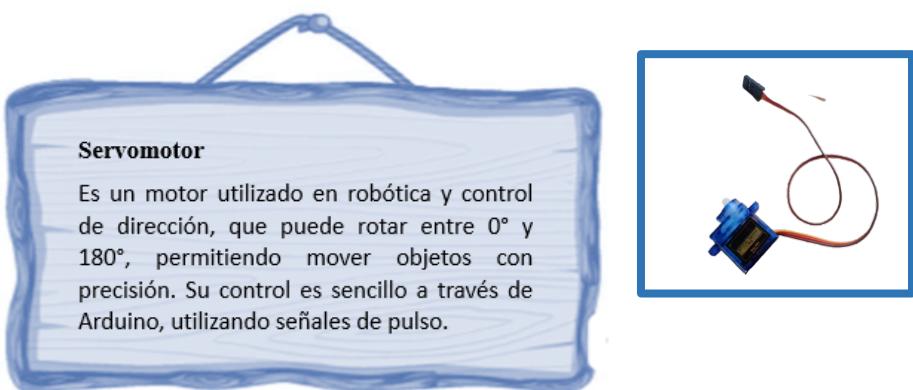
Nota. La tabla muestra los colores más usados de LED y su voltaje de funcionamiento

Servomotor

El servomotor 180° o micro servo SG90 es un tipo de motor que se utiliza para mover objetos a un rango específico de ángulos, generalmente de 0° a 180°. Este motor es muy útil en proyectos de robótica y automatización, ya que permite un control preciso del movimiento. Se controla mediante señales eléctricas enviadas desde la tarjeta *Arduino*, y su capacidad para girar a posiciones específicas lo hace ideal para aplicaciones como mover brazos robóticos o ajustar la dirección de objetos (ver figura 25).

Figura 25

Servomotor para *Arduino*



Nota. La figura muestra un servomotor para *Arduino*

El servomotor de 180° es un componente ampliamente utilizado en proyectos de robótica debido a su capacidad de controlar movimientos precisos en un rango limitado. Estos tienen varios aspectos importantes como:

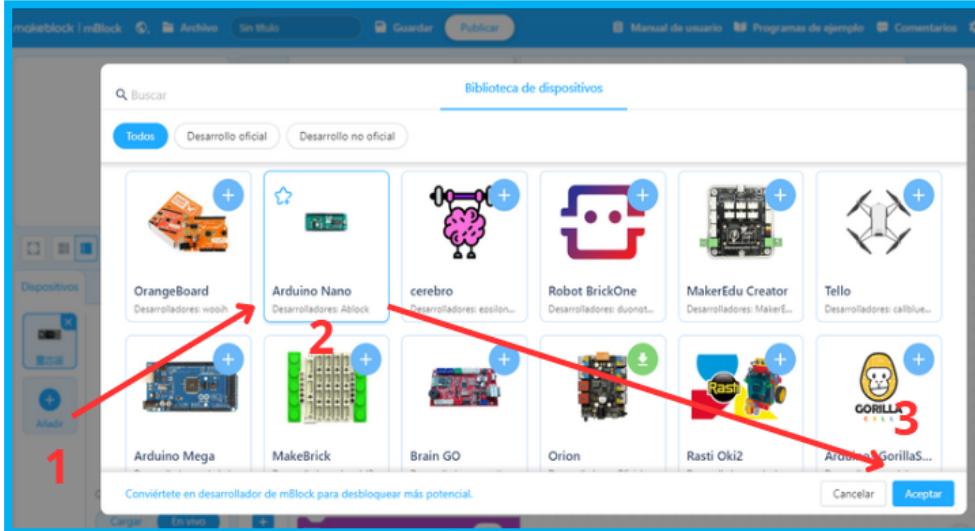
- **Control de precisión:** Permite un movimiento preciso en ángulos específicos, generalmente con incrementos de 1 grado, ideal para tareas que requieren alta precisión.
- **Ángulo de rotación limitado:** Su rango de rotación es de 0° a 180°, lo que lo hace perfecto para aplicaciones donde no se necesita una rotación continua.
- **Composición interna:** Está compuesto por un motor eléctrico, engranajes y un potenciómetro que permite conocer la posición exacta del eje.
- **Fuente de alimentación:** Requiere una fuente de alimentación externa (generalmente 4.8 a 6V), ya que el *Arduino* no puede suministrar suficiente energía.
- **Usos comunes:** Se utiliza en robótica, control de cámaras, sistemas de dirección, modelismo y aplicaciones de automatización.
- **Consumo de energía:** Aunque consume más energía que otros componentes, su eficiencia y precisión lo hacen indispensable para movimientos controlados.
- **Fácil integración con *Arduino*:** Se puede controlar fácilmente con la librería "Servo" de *Arduino*, lo que facilita su uso en proyectos.

2.8. Desafíos lógicos condicionales con sensores ultrasónicos utilizando programación por bloques

El primer paso fundamental para comenzar con la programación de robótica en *mBlock* consiste en agregar la tarjeta *Arduino* que se va a utilizar en el proyecto. Este procedimiento debe realizarse

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula dentro de la biblioteca de dispositivos, tal como se muestra en la figura 26. Añadir la tarjeta correcta garantiza la compatibilidad entre el software y el hardware, permitiendo programar adecuadamente.

Figura 26
Agregación de dispositivos en *mBlock*

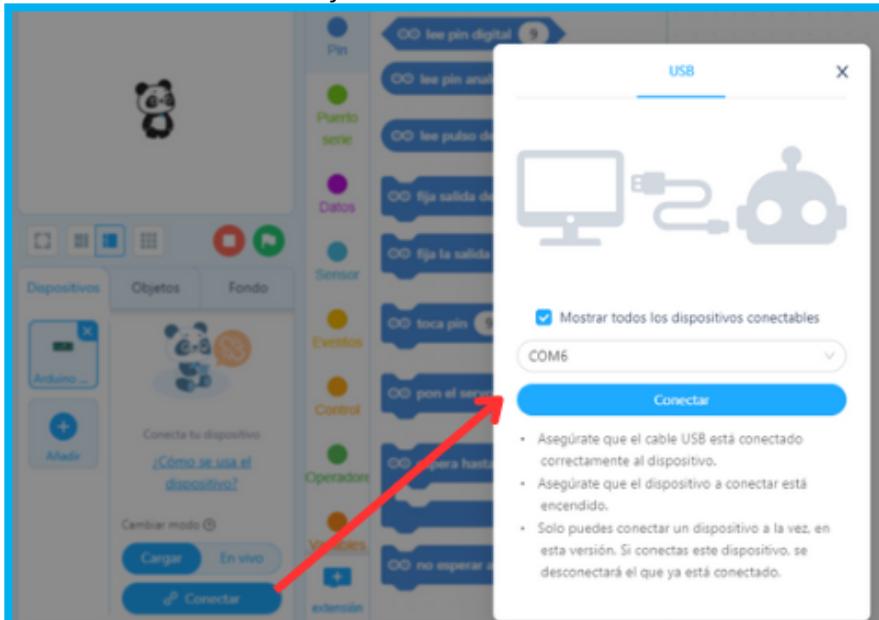


Nota. La figura muestra la agregación de dispositivos en *mBlock*.

Se debe conectar la tarjeta *Arduino NANO* a un puerto USB de la computadora. Ahora debemos conectar la tarjeta *Arduino NANO* al *mBlock* (ver figura 27) de acuerdo al puerto COM que fue detectado por *mBlock*, en este caso se está conectando al puerto COM34.

Figura 27

Conexión de tarjeta Arduino NANO a mBlock



Nota. La figura muestra la conexión de tarjeta Arduino a mBlock. Fuente: Captura de pantalla tomada de la interfaz de mBlock v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

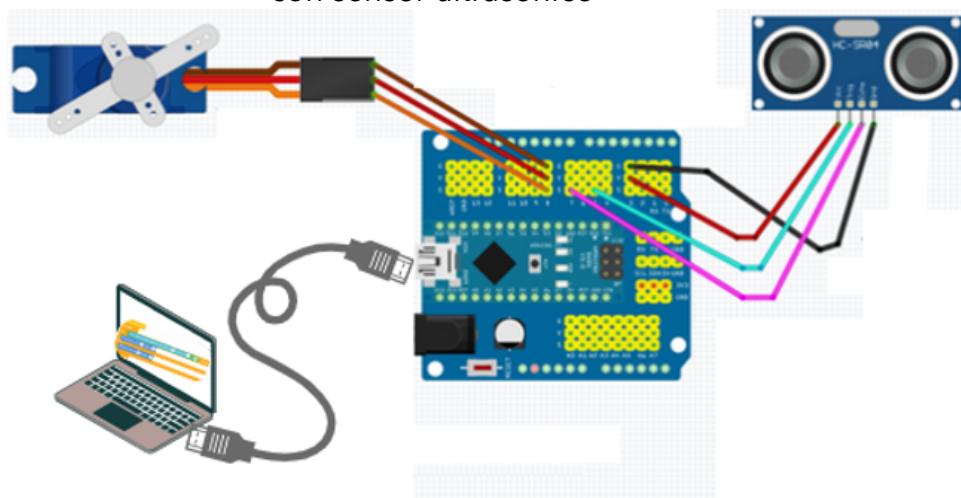
El entorno de programación de *mBlock* ya se encuentra completamente configurado y todo el hardware conectado de forma correcta, lo cual nos permite iniciar con confianza el desarrollo de nuestros desafíos lógicos condicionales. Estos desafíos se basan en el uso de circuitos electrónicos que integran sensores ultrasónicos, permitiendo así la detección de obstáculos y la toma de decisiones automatizadas.

Desafío 1: El objetivo principal es programar y controlar el movimiento de un servomotor utilizando como referencia la

distanza medida por un sensor ultrasónico conectado a la placa *Arduino NANO*. A medida que el sensor detecte distintos rangos de distancia, el servomotor deberá responder con movimientos específicos, permitiendo simular un sistema de respuesta automática ante obstáculos.

Solución de ensamblaje: La figura 28 muestra el ensamblaje del circuito electrónico necesario para el funcionamiento del sensor ultrasónico y el servomotor. En esta configuración, el servomotor se conecta al pin digital 8 de la placa *Arduino NANO*, mientras que el sensor ultrasónico se conecta a los pines adecuados para alimentación y señal. Todo el circuito debe estar correctamente alimentado para su funcionamiento.

Figura 28
*Ensamblaje del desafío 1
con sensor ultrasónico*

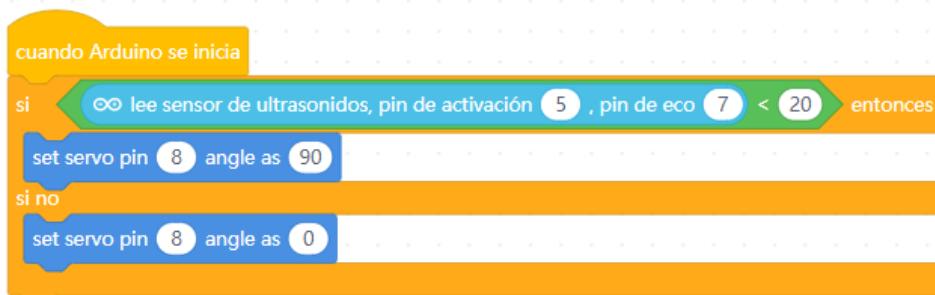


Nota. La figura muestra el ensamblaje del desafío 1 con sensor ultrasónico. Fuente: Captura de pantalla generada de la interfaz de Fritzing. Fritzing (<https://Fritzing.org>)

Solución de programación: La figura 29 presenta la programación por bloques utilizada para controlar el movimiento del servomotor en función de la distancia detectada por el sensor ultrasónico. En esta programación, se leen constantemente los valores del sensor y, según la distancia medida, se envían instrucciones al servomotor para ajustar su posición. Esto permite una respuesta dinámica y automática.

Figura 29

Programación del desafío 1 con sensor ultrasónico



Nota. La figura muestra la programación del desafío 1 con sensor ultrasónico. Fuente: Captura de pantalla tomada de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

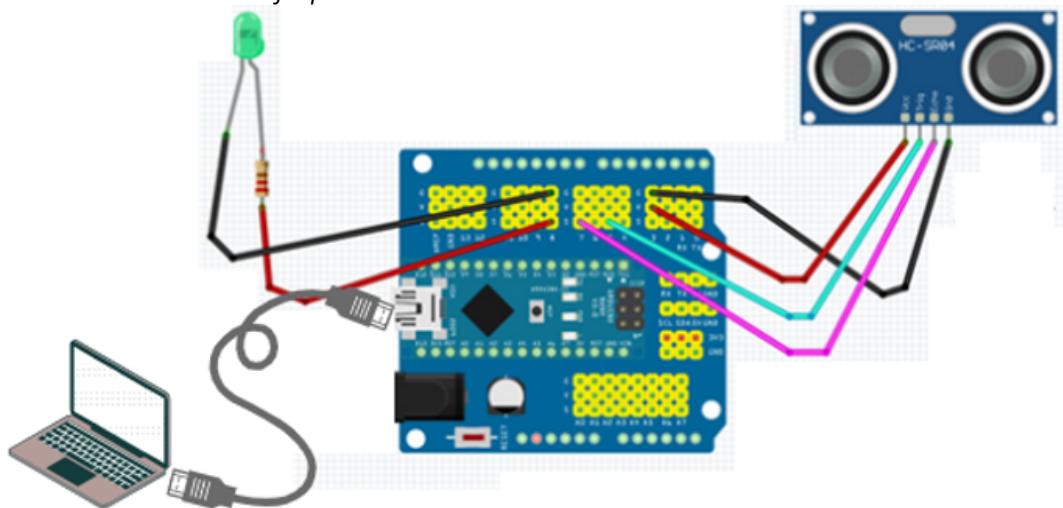
Desafío 2: El objetivo consiste en encender y apagar un LED verde utilizando la lectura de distancia del sensor ultrasónico conectado a la placa *Arduino NANO*. Dependiendo de si un objeto se encuentra dentro de un rango determinado, el LED se activará o desactivará, simulando una sencilla respuesta automatizada.

Solución de ensamblaje: La figura 30 muestra el ensamblaje del circuito electrónico necesario para encender y apagar un LED verde utilizando un sensor ultrasónico. En esta configuración, el sensor se conecta a los pines correspondientes de la placa

Arduino NANO para alimentación y señal, mientras que el LED verde se conecta a un pin digital, a través de una resistencia. Este montaje permite controlar el encendido del LED según la distancia detectada.

Figura 30

Ensamblaje para el desafío 2 con sensor ultrasónico

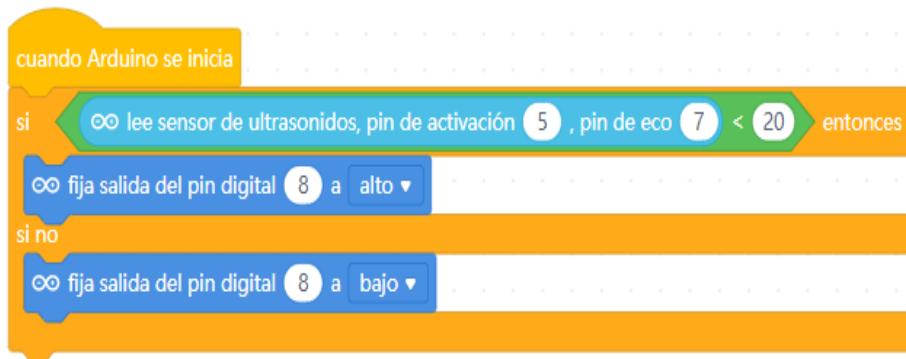


Nota. La figura muestra el ensamblaje para el desafío 2 con sensor ultrasónico. Fuente: Captura de pantalla de la interfaz de *Fritzing*. [Fritzing](https://Fritzing.org) (<https://Fritzing.org>)

Solución de programación: La figura 31 incluye la programación por bloques diseñada para controlar el encendido y apagado del LED verde en función de la distancia medida por el sensor ultrasónico. El código evalúa constantemente los valores obtenidos por el sensor y, si la distancia es menor o mayor a un umbral definido, envía la señal correspondiente al pin digital para activar o desactivar el LED, simulando una acción automatizada basada en proximidad.

Figura 31

Programación para el desafío 2 con sensor ultrasónico



Nota. La figura muestra la programación para el desafío 2 con sensor ultrasónico. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

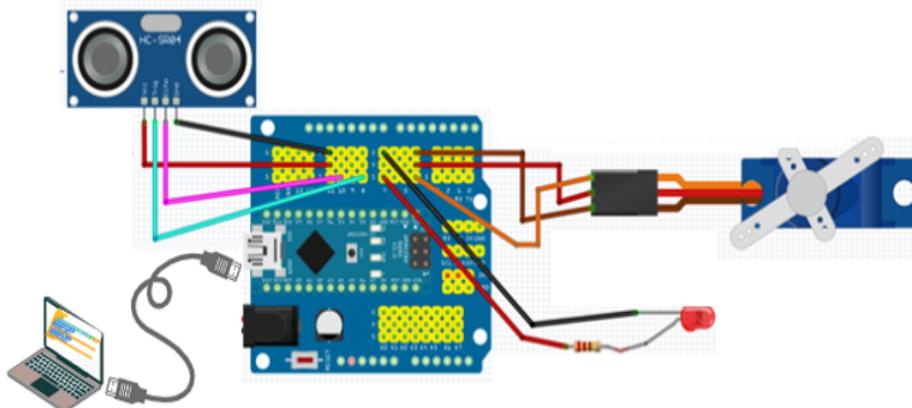
Desafío 3: El propósito de este desafío es encender y apagar un LED al mismo tiempo que un servomotor realiza un movimiento de 90 grados, todo controlado por la lectura de un sensor ultrasónico conectado a la placa *Arduino NANO*. Dependiendo de la distancia detectada por el sensor, se activarán simultáneamente el LED y el movimiento del servomotor, permitiendo simular una respuesta automatizada compuesta que combina señales visuales con acción mecánica. Este desafío integra componentes electrónicos y lógicos, fomentando una mayor comprensión del trabajo en conjunto entre sensores y actuadores.

Solución del ensamblaje: La figura 32 muestra el ensamblaje del circuito electrónico necesario para controlar, a partir de un sensor ultrasónico, tanto el encendido de un LED como el movimiento de un servomotor. En esta configuración, el sensor ultrasónico se conecta a los pines de señal y alimentación de la placa *Arduino NANO*, mientras que el LED se conecta a un pin digital a través de una resistencia. El servomotor, por su parte, se conecta también a un pin digital específico que permita su control. Todos los

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
componentes deben compartir una correcta conexión a tierra y alimentación, asegurando así el funcionamiento conjunto del sistema ante los cambios de distancia detectados por el sensor.

Figura 32

Ensamblaje para el desafío 3 con sensor ultrasónico

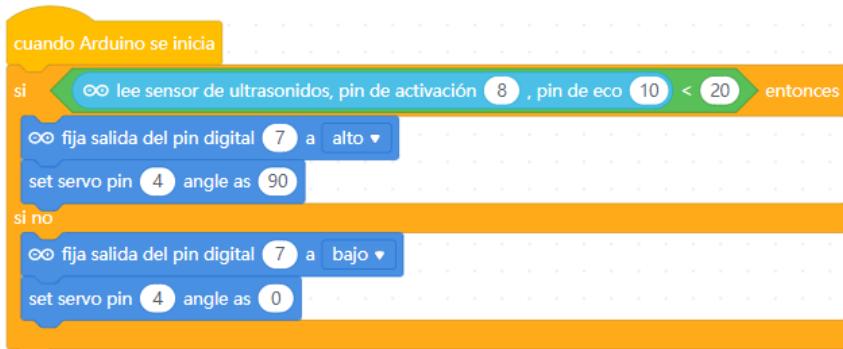


Nota. La figura muestra el ensamblaje para el desafío 2 con sensor ultrasónico. Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Solución de la programación: La figura 33 visualiza la programación por bloques necesaria para coordinar el encendido del LED y el movimiento del servomotor en función de la distancia detectada por el sensor ultrasónico. El programa evalúa continuamente los valores del sensor, y cuando se alcanza una distancia específica, se activa el LED y se ordena al servomotor realizar un movimiento de 90 grados. Si la distancia varía fuera del rango definido, el sistema responde apagando el LED y devolviendo el servomotor a su posición inicial. Esta programación permite una respuesta sincronizada entre los elementos electrónicos según las condiciones del entorno.

Figura 33

Programación para el desafío 3 con sensor ultrasónico



Nota. La figura muestra la programación para el desafío 3 con sensor ultrasónico. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

2.9. Resumen del capítulo

En este capítulo se profundiza en la programación condicional aplicada a sensores ultrasónicos mediante la plataforma *mBlock*. Se abordan la organización del código, el uso de variables, procedimientos y funciones, así como la implementación de sentencias condicionales para la interacción con el entorno físico. También se describe el hardware necesario y se explica cómo estos contenidos permiten desarrollar competencias en tres áreas de la metodología STEAM: Tecnología, Ingeniería y Matemática. A través de ejemplos y fases del proceso de ingeniería, se demuestra cómo integrar la programación y la robótica educativa para resolver problemas reales de manera estructurada y creativa.

2.10. Autoevaluación del capítulo

Pregunta 1: ¿Cuál es la función principal de una estructura condicional en programación?

Opciones de respuesta:

- a) Ejecutar instrucciones en un orden fijo.
- b) Permitir que un programa tome decisiones según condiciones específicas.
- c) Repetir una acción indefinidamente sin cambios.
- d) Organizar datos en listas o tablas.

Pregunta 2: ¿Qué herramienta permite programar estructuras condicionales mediante bloques gráficos?

Opciones de respuesta:

- a) Python
- b) JavaScript
- c) *mBlock*
- d) SQL

Pregunta 3: ¿Cómo funciona un sensor ultrasónico en una estructura condicional?

Opciones de respuesta:

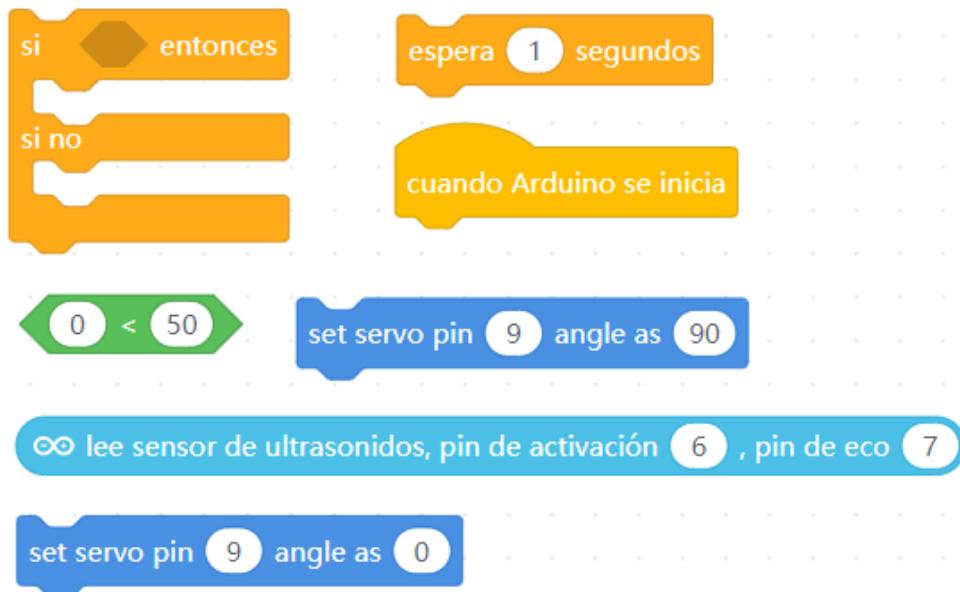
- a) Detectando temperatura y ajustando la velocidad del motor.
- b) Midiendo la distancia a un objeto y activando una acción si se cumple una condición.
- c) Enviando señales infrarrojas para detectar movimiento.
- d) Generando energía a partir de ondas de sonido.

Pregunta 4: Ubicar de forma correcta los bloques de una secuencia de *mBlock*, donde se realiza un movimiento de un servomotor a 90° cuando detecte un objeto cerca del sensor

ultrasónico, con números ubícalos en el orden correcto para formar el código, tacha los bloques que estén mal:

Figura 34

Bloques de mBlock para ubicarlos correctamente



Nota. La figura muestra un conjunto de bloques de *mBlock* para ordenarlos. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Pregunta 5: Según el gráfico que está a continuación escribe el funcionamiento del código.

Figura 35

Bloques de mBlock para describir su funcionamiento



Nota. La figura muestra un conjunto de bloques de *mBlock* para describir su funcionamiento. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Respuesta:

Pregunta 6: Si el tiempo de vuelo medido por un sensor ultrasónico es de 0,01 segundos y la velocidad del sonido es de 343 metros por segundo, ¿cuál sería la distancia al objeto?

Opciones de respuesta:

- a) 1,715 metros
- b) 3,43 metros
- c) 17,15 metros

d) 34,3 metros

Pregunta 7: ¿Cuál es el proceso básico de funcionamiento de un sensor ultrasónico?

Opciones de respuesta:

- a) Emite una señal de radio y mide la cantidad de energía reflejada
- b) Emite una onda sonora, mide el tiempo que tarda en reflejarse y regresa al sensor
- c) Detecta cambios de temperatura en el ambiente
- d) Captura imágenes para calcular la distancia del objeto

Pregunta 8: ¿Cuáles son las principales sentencias condicionales que se utilizan en *mBlock*?

Opciones de respuesta:

- a) "si-entonces" y "si-entonces-sino"
- b) "cuando-hasta" y "mientras-repetir"
- c) "repetir-por siempre" y "detener programa"
- d) "esperar" y "cambiar color"

CAPÍTULO 3: APRENDIZAJE CREATIVO DE PROGRAMACIÓN CON SENORES ULTRASÓNICOS

3.1. Objetivo del capítulo

Al concluir este capítulo, el futuro docente podrá diseñar iniciativas de enseñanza mediante el Aprendizaje Basado en Proyectos, integrando sensores ultrasónicos para fomentar el desarrollo del Pensamiento Computacional. Estas propuestas estarán centradas en el uso de estructuras condicionales y en la implementación de circuitos electrónicos utilizando la plataforma *mBlock*.

3.2. Aprendizajes Esperados

- Caracterización del Aprendizaje Basado en Proyectos con sensores ultrasónicos
- Elaboración de propuestas de secuencias didácticas combinadas con Aprendizaje Basada en Proyectos utilizando sensores ultrasónicos, los proyectos se realizan con enfoque interdisciplinar para enseñanza de pensamiento computacional condicional utilizando programación por bloques en *mBlock*.

3.3. Introducción: La Educación con el Aprendizaje Basada en Proyectos de Sensores Ultrasónicos

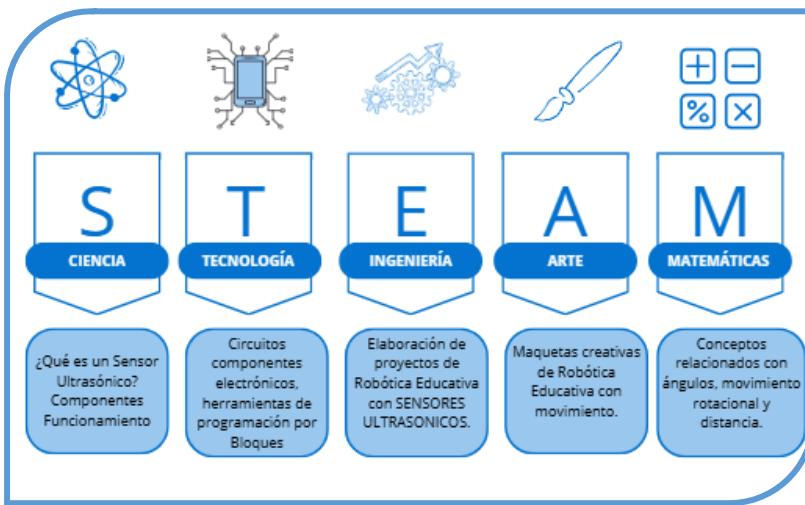
En este capítulo se presentan contenidos que favorecen la integración de las cinco áreas fundamentales que conforman la metodología STEAM (ver figura 36): ciencia, tecnología, ingeniería, arte y matemáticas. A través del diseño y desarrollo de proyectos interdisciplinarios, se promueve una enseñanza articulada y contextualizada, permitiendo al docente implementar

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula propuestas educativas más completas, creativas y vinculadas a situaciones reales del entorno.

Los proyectos con enfoque interdisciplinario presentados abarcan integralmente las áreas de STEAM, tal como se muestra en la figura. Estas experiencias con enfoque STEAM no solo facilitan la comprensión de principios científicos y tecnológicos, sino que también estimulan habilidades blandas como la creatividad, la colaboración y la resolución de problemas, promoviendo un aprendizaje activo, lúdico y significativo para los niños en contextos reales (Arpaci et al., 2023).

Figura 36

Áreas de la metodología STEAM que se abordan en capítulo 3.



Nota. En esta figura se observan las áreas de la metodología STEAM que se abordan en el capítulo 4.

El elemento central de este capítulo es el Aprendizaje Basado en Proyectos (ABP) que se refiere a un enfoque pedagógico que promueve el aprendizaje de los estudiantes mediante la resolución de problemas reales en proyectos educativos. Los estudiantes trabajan de manera activa y colaborativa para investigar, resolver problemas y crear productos que demuestren

su aprendizaje (Seo et al., 2024). Este enfoque fomenta la exploración, el análisis crítico y la creatividad. Algunas características del aprendizaje basado en proyectos son:

El estudiante es el centro de aprendizaje: En el ABP, los estudiantes son protagonistas de su aprendizaje, lo que significa que tienen una gran autonomía en el proceso, ya que son ellos quienes eligen, diseñan, desarrollan y gestionan el proyecto, con el acompañamiento del docente. Fomentando la motivación intrínseca y la responsabilidad del estudiante por su propio aprendizaje, además de las habilidades metacognitivas (Salsabila & Baroroh, 2024).

Interdisciplinariedad: Los proyectos en el ABP integran contenidos y habilidades de diferentes áreas del conocimiento. Esto ayuda a los estudiantes a ver las conexiones entre disciplinas, lo cual es fundamental para una comprensión más global y profunda de los temas. La interdisciplinariedad favorece la transferencia de aprendizajes y permite abordar problemas complejos desde distintas perspectivas (Fitzgerald & Evans, 2024).

Aprendizaje activo: El aprendizaje basado en proyectos es una metodología activa porque involucra a los estudiantes en tareas concretas que requieren esfuerzo cognitivo. Este enfoque promueve la búsqueda, la elección, la discusión, la aplicación, la corrección y el ensayo además de la construcción activa de conocimientos a través de la exploración, investigación y resolución de problemas (Júnior et al., 2024).

Colaboración: Los proyectos en el ABP se basan en el trabajo en equipo, donde los estudiantes deben colaborar para resolver el proyecto propuesto. Los estudiantes aprenden compartir ideas, a discutir, a escuchar a los demás y a tomar decisiones en conjunto, así logrando desarrollar habilidades sociales, de comunicación y

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula de liderazgo, que son fundamentales en el mundo laboral y en la vida cotidiana (Chang et al., 2024).

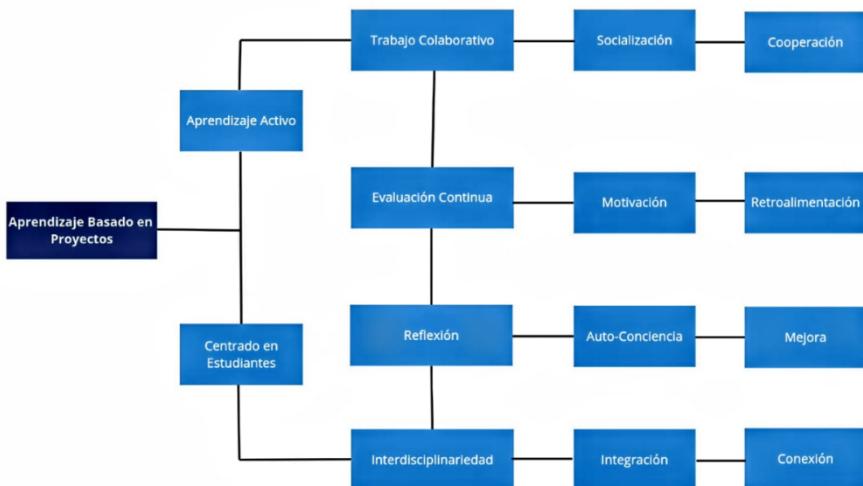
Evaluación continua: En el ABP, la evaluación no es solo un proceso final, sino que es continua y formativa. Se caracteriza por ser integral, formativa, interdisciplinaria y participativa. A lo largo del proyecto, los estudiantes reciben retroalimentación constante del docente, de sus compañeros y de ellos mismos, lo que les permite ajustar y mejorar su trabajo. La evaluación no solo se centra en el resultado final, sino también en el proceso de aprendizaje: cómo investigaron, cómo colaboraron, cómo resolvieron los problemas, y cómo reflexionaron sobre su trabajo (Sircar et al., 2024).

Reflexión: La reflexión es una parte fundamental del ABP, ya que permite a los estudiantes pensar sobre su propio proceso de aprendizaje, es un proceso continuo de reflexión, mejora y evaluación. A través de la reflexión, los estudiantes analizan lo que han hecho bien, lo que podrían mejorar, y cómo su aprendizaje se puede aplicar a otras situaciones. La reflexión también ayuda a los estudiantes a desarrollar habilidades metacognitivas, es decir, la capacidad de pensar sobre su propio pensamiento (Alarfaj et al., 2024).

A continuación, se presenta un resumen gráfico de las características más importantes del Aprendizaje Basado en Proyectos (ver figura 37).

Figura 37

Características de Aprendizaje Basado en Proyectos



Nota. La figura muestra las características de Aprendizaje Basado en proyectos.

3.4. Aprendizaje Basado en Proyectos (ABP) utilizando sensores ultrasónicos

Conceptualización del ABP orientada a sensores ultrasónicos

El Aprendizaje Basado en Proyectos se refiere a un método aplicado a la educación en el que los estudiantes se involucran y participan activamente en el desarrollo de proyectos extensos y desafiantes. Cuando se orienta al uso de sensores ultrasónicos, el ABP permite integrar conceptos de programación, electrónica y física, incentivando a los estudiantes a diseñar soluciones innovadoras mediante la medición de distancias o la automatización de procesos. Esta orientación práctica no solo fortalece el pensamiento lógico y computacional, sino que también facilita la aplicación de conocimientos interdisciplinarios,

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula especialmente en contextos STEAM, potenciando la creatividad, la toma de decisiones y el trabajo colaborativo (Leavy et al., 2023).

Esta metodología implica que los estudiantes trabajen en un proyecto durante un período determinado, con una evaluación que no se limita al resultado final, sino que se lleva a cabo de manera continua. Este seguimiento permite ofrecer retroalimentación constante, facilitando la reflexión sobre el progreso, las competencias adquiridas y el producto final, promoviendo así un aprendizaje más profundo y consciente.

Una de sus principales características es la generación de un producto o solución concreta al finalizar el proyecto. Este resultado puede manifestarse en diferentes formatos, como informes, presentaciones o prototipos, evidenciando la aplicación práctica de lo aprendido. A lo largo del desarrollo, el ABP conecta la teoría con situaciones reales, fomentando un aprendizaje significativo y relevante.

Fases del Aprendizaje Basado en Proyectos orientadas a sensores ultrasónicos

El aprendizaje basado en proyectos se desarrolla en varias fases que guían a los estudiantes desde la exploración de un problema hasta la presentación de una solución. Estas etapas fomentan la investigación, la colaboración, el pensamiento crítico y la aplicación práctica de conocimientos, promoviendo un aprendizaje significativo y centrado en el estudiante. De acuerdo a Navarro Arcas et al. (2024) las fases del ABP son:

Identificación de la problemática: Esta fase inicia con la presentación de una situación significativa, preferiblemente vinculada al contexto de los estudiantes. Se busca que el problema despierte curiosidad y tenga una posible solución mediante el uso de sensores ultrasónicos. Por ejemplo, diseñar un sistema de alerta para evitar colisiones en un robot, o crear un

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
sensor de proximidad para personas con discapacidad visual. Es fundamental que el reto sea abierto y fomente la creatividad.

Investigación y planificación del proyecto: Los estudiantes investigan sobre los sensores ultrasónicos: su funcionamiento, aplicaciones reales, conexión a placas electrónicas y principios básicos de programación. Esta fase incluye la planificación colaborativa del proyecto: definición de roles, elaboración de cronogramas, asignación de tareas, análisis de recursos disponibles y selección de herramientas como *mBlock* o *Arduino*. La fase también contempla la formulación de hipótesis y la identificación de criterios de éxito del proyecto.

Diseño e implementación del prototipo: Aquí se empieza a construir el proyecto: se diseñan los circuitos, se conectan los sensores ultrasónicos, y se programan utilizando entornos como *mBlock*, donde se aplican estructuras condicionales para interpretar datos de distancia. Esta fase favorece la aplicación de conocimientos interdisciplinarios (física, tecnología, matemáticas, computación) y el desarrollo de habilidades prácticas. El estudiante convierte ideas abstractas en soluciones tangibles mediante prueba y error.

Pruebas, ajustes y mejora del proyecto: Una vez montado el prototipo, se realizan pruebas de funcionamiento en diferentes condiciones. Se analizan los resultados, se detectan errores y se implementan mejoras tanto en el diseño del circuito como en la lógica del programa. Esta fase fomenta el pensamiento crítico, la resiliencia ante el error, la experimentación y la toma de decisiones fundamentadas. Es una etapa iterativa en la que el fallo se considera parte esencial del aprendizaje.

Presentación y socialización del proyecto: El proyecto final con sensores ultrasónicos se presenta ante compañeros, docentes u otros miembros de la comunidad educativa. Los estudiantes

explican el proceso de diseño, los problemas que enfrentaron, las decisiones tomadas y el funcionamiento del sistema creado. Esta etapa refuerza la comunicación oral, el uso de vocabulario técnico, la argumentación y la confianza en sí mismos. Puede incluir una demostración práctica o la elaboración de un informe o póster científico.

Evaluación y retroalimentación: Se valora el proyecto de sensores ultrasónicos desde una perspectiva integral. Se consideran aspectos técnicos, el grado de innovación, el trabajo en equipo, la gestión del tiempo, y el desarrollo de habilidades blandas y cognitivas. La evaluación puede incluir rúbricas, autoevaluación, coevaluación y retroalimentación del docente. Esta etapa promueve la reflexión sobre el propio aprendizaje y sienta las bases para futuros proyectos.

3.5. Secuencia didáctica 1: El sensor ultrasónico y el descubrimiento del mundo a través del sonido

Secuencia didáctica 1

“El sensor ultrasónico y el descubrimiento del mundo a través del sonido”

Objetivo General

El niño comprenderá qué es un sensor ultrasónico, cómo funciona y por qué es especial en la medición de distancias, reconociendo su principio de emisión y recepción de sonido a través del eco y su aplicación en tecnologías cotidianas.

Presentación

Esta secuencia didáctica tiene como objetivo guiar al niño en el descubrimiento del uso y entendimiento del sensor ultrasónico, mostrándoles su funcionamiento, ventajas y presencia en el mundo real, mientras realiza actividades prácticas y creativas para consolidar el conocimiento.

Estructura de la secuencia didáctica

Sesión 1: Introducción. Explorando el sonido.

Sesión 2: Eco y rebote. Simulando el sensor ultrasónico.

Sesión 3: Radar humano. Jugando a ser un sensor.

Sesión 1: Introducción: Explorando el sonido

Objetivo específico:

Introducir el concepto de las ondas sonoras y cómo viajan, conociendo su relación con el funcionamiento del sensor ultrasónico.

Materiales:

- Resorte o cuerda larga
- Tambor o caja de cartón
- Globos inflados
- Pizarra o cartel para dibujar ondas

Actividades:

Actividad 1: Bienvenida y Presentación

Da la bienvenida a los niños y explica que en esta sesión aprenderán sobre cómo viaja el sonido, que es la base del funcionamiento de los sensores ultrasónicos. Comenta que el sonido no viaja de cualquier manera, sino que se propaga en forma de ondas, y esas ondas pueden rebotar. Inicia con una pregunta: ¿Alguna vez han gritado en un lugar grande y han escuchado que su voz rebota? ¿Cómo creen que el sonido viaja?

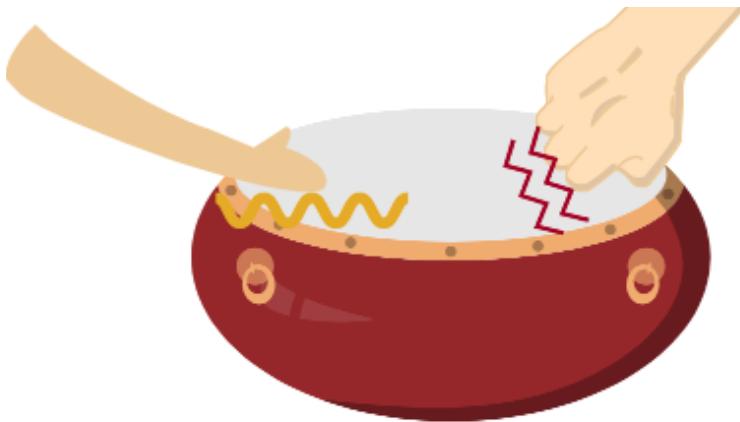
Comenta con ellos las respuestas brevemente, además de dar ejemplos de los ecos.

Actividad 2: Demostración práctica

Usa un resorte o una cuerda larga para mostrar cómo se propagan las ondas cuando se mueve un extremo. Hazlo en vivo para que los niños puedan ver la propagación y la vibración. Pídeles que toquen el tambor o la caja de cartón mientras alguien más la golpea para sentir las vibraciones (ver figura 38).

Figura 38

Tambor para actividad vibraciones



Nota. En esta figura se visualiza de forma animada la actividad práctica de sentir la vibración después del golpe. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Explica que esas vibraciones son ondas, algo parecido a cómo funciona el sonido.

Actividad 3: Comparación visual

En la pizarra, dibuja diferentes tipos de ondas: las ondas del sonido, las del agua y otras más, luego relaciona cómo las ondas del sonido son similares a las ondas en el agua cuando se tira una piedra, a continuación, haz preguntas: ¿Por qué el sonido puede rebotar en las paredes y cambiar de dirección? Esto con el fin de ayudarle a analizar mejor sobre las ondas de sonidos, luego socializa con ellos sobre la respuesta.

Actividad 4: Reflexión

Al final de la sesión, invita a los niños a compartir lo que aprendieron sobre las ondas sonoras. Se puede preguntar: ¿Cómo creen que las ondas del sonido son similares a las ondas en un sensor ultrasónico?, da una explicación sobre la relación o similar entre ellos, puede usar ejemplo de los sonidos con el sensor, al concluir, puede realizar una pregunta: ¿Qué les gustaría explorar más sobre cómo viajan las ondas?

Evaluación de la sesión

Puede pedirles a los niños que escriban como creen que se funciona el Sensor ultrasónico siguiendo la explicación de las ondas de sonidos.

Sesión 2: Eco y Rebote: Simulando el Sensor ultrasónico

Objetivo específico:

Introducir la idea de que los sensores ultrasónicos usan el eco para medir distancias.

Materiales:

- Tubo de cartón largo (de toallas de papel)
- Pelota pequeña
- Cronómetro

Actividades:

Actividad 1: Bienvenida

Da la bienvenida a los niños y explíquales que en esta sesión aprenderán cómo los sensores ultrasónicos pueden medir distancias usando el sonido.

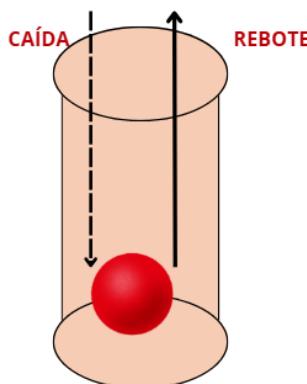
Coméntales sobre cómo algunos animales, como los murciélagos, usan el eco para "ver" en la oscuridad. Pregúntales: ¿Alguna vez han gritado en un lugar vacío y escuchado cómo se rebota el sonido? Explícales qué es sobre el eco y como el sonido regresa de donde inicio.

Actividad 2: Demostración práctica

Usando un tubo, deja que los niños lancen una pelota dentro de él y midan el tiempo que tarda en salir. Esto servirá como analogía para entender cómo los sensores ultrasónicos miden la distancia mediante el eco. Luego, pídeles que repitan el ejercicio con tubos de diferentes longitudes y comparan los tiempos. Relaciona estos tiempos con el concepto del rebote del sonido en los sensores ultrasónicos, explicando cómo el emisor envía una señal y el receptor la detecta después del rebote (ver figura 39).

Figura 39

Representación de la caída y rebote de la pelota.



Nota. En esta figura se observa la caída de la pelota y el rebote que genera. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Actividad 3: Discusión grupal

Haz preguntas a los niños sobre cómo cambia el tiempo en función del tamaño del tubo. Relaciona estos cambios con el funcionamiento de los sensores ultrasónicos, explicando que, al igual que el eco rebota dentro del tubo, los sensores envían una señal y miden el tiempo que tarda en regresar. Explica cómo esta medición permite a los sensores calcular la distancia hasta un objeto.

Actividad 4: Reflexión

Al final de la sesión, invita a los niños a compartir lo que han aprendido sobre el eco y los sensores ultrasónicos. Pregúntales: ¿Cómo creen que el eco ayuda a los sensores a detectar objetos? y ¿Qué pasaría si el eco no regresara al sensor? Anima a los niños a reflexionar y a explicar sus ideas con ejemplos.

Evaluación de la sesión

Pídeles a los estudiantes que se dividan en grupos pequeños para responder preguntas relacionadas con el funcionamiento de los sensores ultrasónicos. A través de este ejercicio, se fomentará la reflexión y el trabajo colaborativo, permitiendo que cada equipo explique sus respuestas con ejemplos concretos y relacione el concepto con situaciones de la vida real. En caso de dudas, los integrantes podrán debatir entre ellos para llegar a una conclusión conjunta.

Sesión 3: Radar Humano: Jugando a ser un Sensor

Objetivo específico:

Simular el funcionamiento de un sensor ultrasónico de manera práctica, ayudando a los niños a entender cómo detecta obstáculos.

Materiales:

- Venda para cubrir los ojos
- Objetos pequeños (botellas, cajas, conos)
- Un compañero para hacer ruidos

Actividades:

Actividad 1: Bienvenida

Da la bienvenida y explica que hoy aprenderán cómo los sensores ultrasónicos detectan objetos sin verlos. Comenta sobre cómo los autos usan sensores para evitar chocar con otros objetos sin tener que verlos, haz una pregunta: ¿Cómo creen que el sensor sabe cuándo detenerse o cuándo hay algo cerca? Para iniciar la actividad práctica.

Actividad 2: Simulación en parejas

Para esta actividad escoge a dos niños al azar o voluntario, uno de los niños se pondrá una venda y será el "sensor ultrasónico", otro niño hará ruidos con las manos cerca de diferentes objetos para representar el sonido que el sensor puede detectar. El niño con los ojos vendados debe decir si el objeto está cerca o lejos dependiendo del sonido.

Haz que cambien de roles para que todos tengan la oportunidad de ser el "sensor" y realicen la misma actividad analizando si verifican que el objetivo está cerca o lejos (ver figura 40).

Figura 40

Representación animada de la actividad.



Nota. En esta figura se visualiza de forma animada la actividad en ejecución, donde se genera el ruido que el sensor debe detectar. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Actividad 3: Comparación con la realidad

Puede explicar cómo el ejercicio realizado en clase es similar a la forma en que los sensores ultrasónicos funcionan en robots y automóviles para detectar objetos. Se debe destacar que, al igual que en el experimento, estos sensores emiten ondas sonoras que rebotan en los objetos y regresan al sensor, lo que les permite calcular la distancia. Para ilustrar esta aplicación en la vida real, se pueden mostrar ejemplos de cómo los sensores ultrasónicos se utilizan en sistemas de asistencia para estacionamiento en vehículos, en robots de navegación autónoma o en dispositivos de seguridad. Esta explicación ayuda a los estudiantes a comprender el concepto de una manera práctica y significativa, y les permite conectar lo aprendido con situaciones cotidianas

Actividad 4: Reflexión

Al final de la sesión, invita a los niños a compartir sus experiencias sobre el ejercicio, realizando una serie de preguntas tales como:

¿Fue difícil identificar si el objeto estaba cerca o lejos solo con el sonido? ¿Cómo creen que los sensores ultrasónicos ayudan a los autos y robots a evitar accidentes?

Evaluación de la sesión

Pide a los alumnos escribir su experiencia, tanto como siendo el sensor o creando el sonido, comparar las experiencias con los demás.

Sesión 4: Construyendo un "Detector de Distancia" de papel

Objetivo específico:

Representar cómo un sensor ultrasónico mide distancias de manera visual y creativa.

Materiales:

- Regla
- Cartón o cartulina
- Tijeras y pegamento
- Hilo y clip

Actividades:

Actividad 1: Bienvenida y Presentación

Primero da el saludo de Bienvenida y prosigue a explicar que, en esta sesión, los niños construirán un modelo de un detector de distancia, similar a cómo los sensores ultrasónicos miden la distancia a los objetos. Comenta sobre la importancia de los sensores en robots y autos para evitar accidentes y mejorar la seguridad. Luego, plantea la pregunta: ¿Cómo creen que los robots o autos saben qué tan lejos están de un objeto?

Actividad 2: Creación del modelo

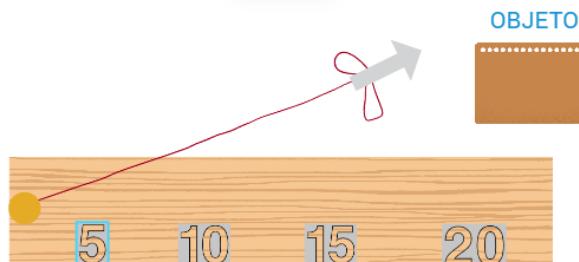
Para construir un detector de distancia, necesitarás algunos materiales simples como cartulina o cartón, una regla, tijeras, pegamento, hilo (común o de lana), clips o pinzas, marcadores o crayones para decorar, y una flecha o triángulo de cartulina. La cartulina servirá como base para el detector, y con la regla, podrás trazar una escala de distancias en la parte inferior. Puedes hacer la escala de 0 cm a 50 cm o 100 cm, dependiendo del tamaño de tu cartulina, y marcar líneas horizontales a intervalos regulares, como cada 5 o 10 cm.

La flecha que crearás con la cartulina representará el "sensor ultrasónico". Este sensor se moverá a lo largo de la escala para indicar las distancias. Asegúrate de que la flecha sea lo suficientemente grande para deslizarse cómodamente sobre la cartulina. Para simular cómo se mueve un sensor, corta un pedazo de hilo largo (unos 30 cm) y ata un extremo a un clip, que será el peso que permita que la flecha se desplace por la escala.

Una vez que tengas la flecha y el hilo listos, pega un extremo del hilo a la cartulina, dejando que el clip se mueva libremente sobre la escala. Esto simula cómo un sensor ultrasónico se mueve para medir distancias, mientras que el clip representa el sensor en movimiento. Los niños pueden utilizar este "detector" para medir distancias en el aula, moviendo el clip y observando cómo la flecha marca diferentes puntos en la escala (ver figura 41).

Figura 41

Representación gráfica de la actividad *Detector de Distancia*



Nota. En esta figura se observa la actividad práctica que sirve para detectar la distancia de un objeto, esta escala va desde el 0 hasta el 20 de 5 en 5. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Para hacer la actividad más interesante, los niños pueden decorar su detector con dibujos relacionados con los sensores ultrasónicos, como robots o autos, o agregar colores y gráficos en la escala de distancias. Este ejercicio no solo es creativo, sino que también ofrece una forma visual de entender cómo funcionan los sensores ultrasónicos, simulando cómo miden la distancia a través de la reflexión de ondas sonoras.

Actividad 3: Estimación de distancias

Los niños usarán su detector para medir distancias de objetos en el aula, simulando cómo funciona un sensor ultrasónico. Luego, podrán comparar los resultados entre ellos y discutir cómo se realiza la medición, reflexionando sobre las similitudes con el funcionamiento de un sensor ultrasónico real

Reflexión

Al final de la sesión, invita a los niños a compartir lo que aprendieron al crear su detector. Pregunta: ¿Cómo creen que este modelo simula el trabajo de un sensor ultrasónico? ¿Qué otras

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

cosas creen que pueden medir utilizando esta idea de los sensores?

Evaluación de la sesión

Los niños deberán realizar una lluvia de ideas donde comparten sus ideas sobre lo que creen que son los sensores ultrasónicos y cómo funcionan. Puedes guiarlos con preguntas como: ¿Qué creen que hace un sensor ultrasónico? ¿En qué situaciones creen que se usan los sensores ultrasónicos? ¿Qué tipos de objetos pueden medir los sensores?

Cierre y Reflexión Final

Objetivo específico:

El niño resumirá lo aprendido sobre el sonido y los sensores ultrasónicos.

Actividades:

Actividad 1: Reflexión grupal

Reúne a los niños y pídeles que comparten lo que aprendieron sobre el sonido y los sensores ultrasónicos. Puedes hacer preguntas como: ¿Cómo viaja el sonido? ¿Qué es el eco y cómo lo usan los sensores ultrasónicos? ¿Cómo los sensores ultrasónicos pueden ayudar a los autos y robots?

Actividad 2: Refuerzo visual

Haz un repaso de las imágenes de sensores ultrasónicos en autos, robots y otros dispositivos. Refuerza la idea de que estos sensores

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula están en muchas tecnologías que usamos para hacer nuestras vidas más seguras y fáciles.

Evaluación de la sesión

Pídeles que realicen un organizador gráfico sobre lo que se vio en las secciones anteriores, da una retroalimentación sobre los temas tratados ayudándolos a recordar y así completar la actividad con éxito.

3.6. Secuencia didáctica 2: Aprendiendo a programar con *mBlock*

Secuencia didáctica 2

**"Aprendiendo a
programar con *mBlock*"**

Objetivo General

Comprender el funcionamiento del sensor ultrasónico junto al servomotor y al LED, aplicándolos en un proyecto práctico de robótica con *mBlock* usando *Arduino NANO*.

Presentación

Esta secuencia didáctica tiene como objetivo guiar al niño en el descubrimiento del funcionamiento del sensor ultrasónico junto al servomotor y al LED.

Estructura de la secuencia didáctica

Sesión 1: Sensor Ultrasónico y Servomotor

Sesión 2: Sensor Ultrasónico y Led

Sesión 3: Explorando el Sensor Ultrasónico, Led y el Servomotor

Sesión 4: Proyecto “Puerta Automática”

Sesión 1: Sensor ultrasónico y Servomotor

Objetivo: Aprender cómo usar el sensor ultrasónico para medir distancias y controlar el movimiento de un servomotor.

Materiales:

Placa *Arduino* o *Arduino NANO*

Sensor ultrasónico HC-SR04

Servomotor

Cables de conexión

Actividades:

Actividad 1: Introducción al Proyecto

Dar la Bienvenida, explicar cómo el sensor ultrasónico mide distancias utilizando ondas sonoras y cómo un servomotor puede moverse a diferentes ángulos mediante señales, dar

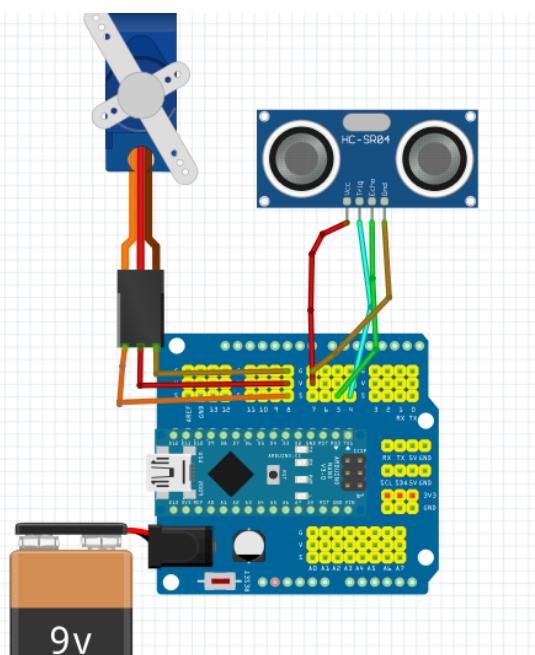
preguntas de introducción ¿Cómo creen que se relacionan el sensor ultrasónico con el servomotor?

Actividad 2: Conexión de los Componentes

Conectar el sensor ultrasónico y el servomotor a la placa *Arduino* utilizando cables. En el sensor ultrasónico se realiza la conexión del Trig: 4S, Eccho: 5S, VCC: 7V, GND: 7G. Y el servomotor va unido al pin (ver figura 42).

Figura 42

Conexión realizada en *Arduino* sobre el Sensor ultrasónico con Servomotor



Nota. En esta figura se visualiza la conexión realizada para conectar el sensor con el servomotor. Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Actividad 3: Programación del Proyecto

Mediante el uso de *mBlock* realizar la programación en bloques. Se usan varios paneles entre ellos (ver figura 43).

Figura 43

Código realizado en *mBlock* para realizar el movimiento del servomotor guiada por el sensor ultrasónico.



Nota. En esta figura se observa el código realizado en *mBlock* para realizar el movimiento del servomotor con el sensor. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Actividad 4: Prueba y Ajustes

Cargar el programa en *Arduino* y probar el funcionamiento del sensor ultrasónico y servomotor. Los estudiantes deben verificar que el servomotor responda correctamente al cambio de distancia.

Reflexión:

Preguntar a los estudiantes cómo podrían modificar el código para que el servomotor realice otros movimientos (como rotar 180 grados) o cómo podría mejorar la precisión del sensor.

Sesión 2: Sensor ultrasónico y Led.

Objetivo: Aprender cómo usar el sensor ultrasónico para medir distancias y activar un LED según esa distancia.

Materiales:

Placa *Arduino* o *Arduino NANO*

Sensor ultrasónico HC-SR04

LED

Resistencia 220Ω

Cables de conexión

Actividades:

Actividad 1: Introducción al Proyecto

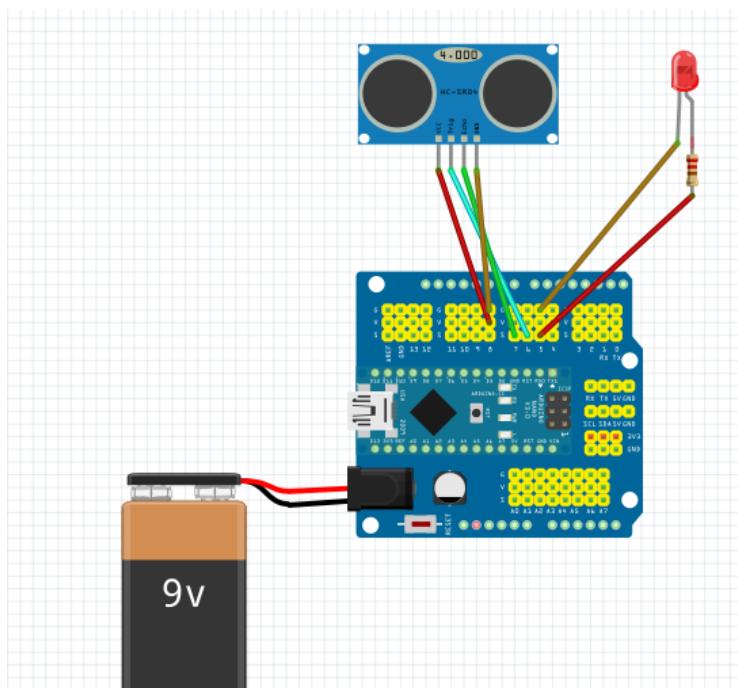
Explicar cómo el sensor ultrasónico puede ser utilizado para activar dispositivos como LEDs en función de la distancia medida.

Actividad 2: Conexión de los Componentes

Conectar el sensor ultrasónico y el LED a la placa *Arduino*. El LED debe estar conectado a un pin digital (con una resistencia para protegerlo). En el sensor ultrasónico, en Trig: 6S, Echo: 7S, VCC: 8V y GND: 8G, el LED conectado al Pin 5(ver figura 44).

Figura 44

Conexión el sensor ultrasónico con un led.



Nota. En esta figura se observa la conexión del sensor ultrasónico con un led. Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Actividad 3: Programación del Proyecto

Escribir el código para activar el LED cuando la distancia medida sea menor que un valor determinado (ver figura 45).

Figura 45

Código para el funcionamiento del sensor ultrasónico con led.



Nota. En esta figura se visualiza el código de conexión en *mBlock*. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Actividad 4: Prueba y Ajustes.

Cargar el código en *Arduino* y verificar que el LED se encienda cuando un objeto esté cerca del sensor (menor a 40 cm) y se apague cuando la distancia sea mayor.

Reflexión:

Preguntar a los estudiantes cómo podrían ajustar el proyecto para que el LED parpadee en lugar de encenderse de forma continua.

Sección 3: Explorando el Sensor ultrasónico, Led y Servomotor.

Objetivo: Integrar tanto el servomotor como el LED para que trabajen juntos en función de la distancia medida por el sensor ultrasónico.

Materiales:

Placa *Arduino* o *Arduino NANO*

Sensor ultrasónico HC-SR04

Servomotor

LED y resistencia (220Ω)

Cables de conexión

Actividades:

Actividad 1: Introducción al Proyecto

Explicar cómo se pueden combinar ambos dispositivos (servomotor y LED) para realizar una acción conjunta. Por ejemplo, el servomotor se mueve cuando la distancia es baja y el LED se enciende cuando la distancia es corta.

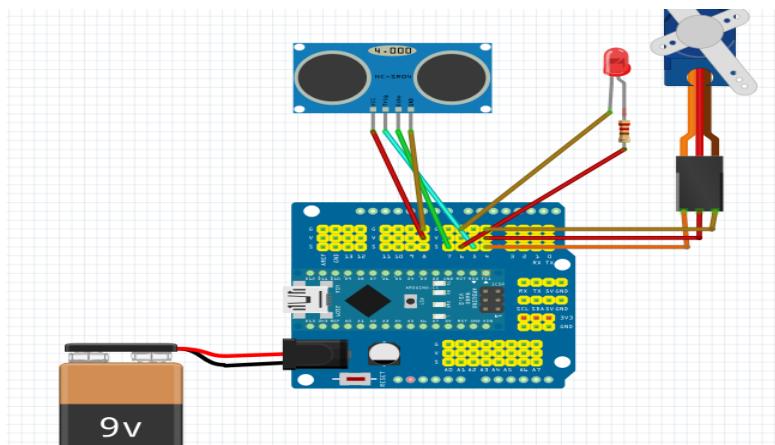
Actividad 2: Conexión de los Componentes

Conectar todos los componentes (sensor ultrasónico, servomotor, LED) a la placa *Arduino*. Asegurarse de que cada componente esté bien conectado a los pines correctos.

En el sensor ultrasónico, en Trig: 5S, Echo: 7S, VCC: 8V y GND: 8G, el LED conectado al Pin 6, el servomotor va conectado al Pin 4(ver figura 46).

Figura 46

Conexión del sensor ultrasónico, led y servomotor.



Nota. En esta figura se observa la conexión del sensor ultrasónico, led y servomotor. Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Actividad 3: Programación del Proyecto

Crear un código que haga lo siguiente:

Mover el servomotor a 90 grados si la distancia es menor a 40 cm y encender el LED cuando la distancia sea menor a 40 cm y apagarlo cuando sea mayor de esa cantidad (ver figura 47).

Figura 47

Código el funcionamiento de la conexión con sensor ultrasónico, led y servomotor.



Nota. En esta figura se observa la conexión del código para el funcionamiento del sensor ultrasónico, led y servomotor. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Actividad 4: Prueba y Ajustes

Cargar el código y verificar que tanto el servomotor como el LED respondan correctamente a la distancia medida por el sensor ultrasónico.

Reflexión:

Discutir cómo la integración de varios componentes permite crear un sistema más complejo. Preguntar a los estudiantes cómo podrían modificar el proyecto para que el servomotor gire a diferentes posiciones dependiendo de la distancia.

Sesión 4: Proyecto “Puerta Automática”

Objetivo: Los estudiantes construirán una maqueta de una puerta automática que se abre y cierra en función de la distancia medida por un sensor ultrasónico. El LED se

encenderá cuando el objeto esté cerca y se apagará cuando esté lejos.

Materiales:

Placa *Arduino* o *Arduino NANO*

Sensor ultrasónico HC-SR04

Servomotor

LED (220Ω de resistencia)

Cables de conexión

Cartón, madera o material plástico para construir la puerta

Pegamento, tijeras, y regla

Base para montar la maqueta (puede ser un cartón o una base de madera)

Cinta métrica o regla para medir distancias

Actividades:

Actividad 1: Introducción y Explicación del Proyecto

Dar el inicio o introducción hablando sobre el sensor ultrasónico, el servomotor, el LED y cómo estos interactúan en el proyecto. Explicar el concepto de pensamiento computacional secuencial, ya que los estudiantes deberán programar el *Arduino* para que se ejecuten ciertas acciones según la entrada del sensor.

Presentar el proyecto y explicar a los estudiantes que van a crear una maqueta de una puerta automática que se abrirá cuando un objeto se acerque y se cerrará cuando se aleje. También, se utilizará un led que se encenderá cuando el objeto esté cerca.

Actividad 2: Diseño y Construcción de la Maqueta

Para la maqueta se necesitan varias piezas y materiales:

Puerta: Utilizar cartón, madera o material plástico para construir una puerta que se mueva mediante el servomotor.

Base: Colocar la maqueta sobre una base sólida (puede ser cartón o madera) donde se montarán los componentes.

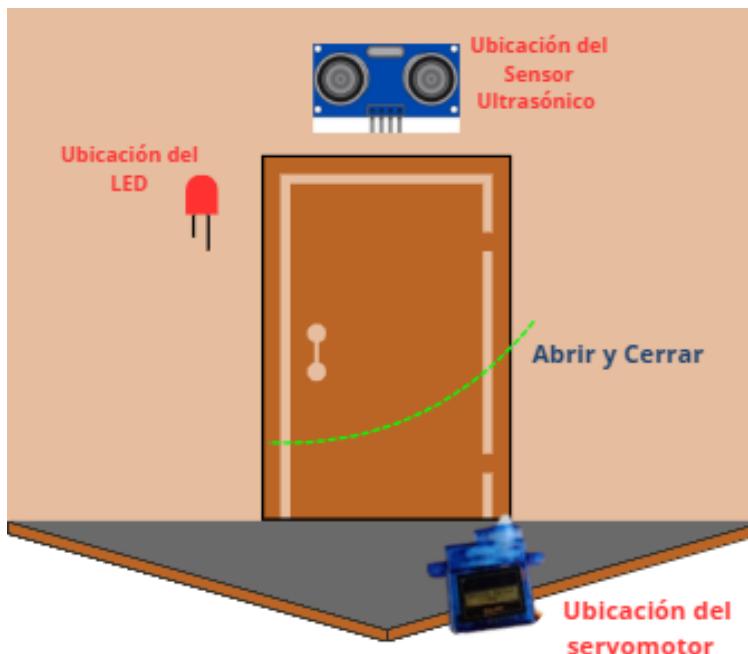
Montaje del Sensor ultrasónico: Colocar el sensor en la parte superior de la puerta para que pueda medir la distancia desde un objeto en frente.

Montaje del Servomotor: Colocar el servomotor de modo que pueda mover la puerta (por ejemplo, moviéndola de 0 a 90 grados, ya sea en la parte superior o inferior).

LED: Colocar el LED en un lugar visible para indicar que el sistema está activo (ver figura 48).

Figura 48

Diseño de la maqueta con la ubicación de cada componente.



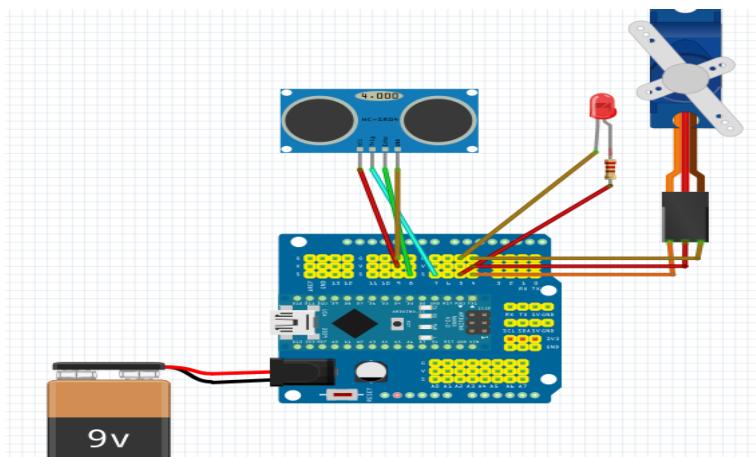
Nota. En esta figura se observa el diseño animado de maqueta donde se implementan el servomotor, led y sensor ultrasónico. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Actividad 3: Conexión de los Componentes Electrónicos

Conectar el Sensor ultrasónico: Conectar los pines Echo 8S y Trig 7S del sensor ultrasónico a los pines del *Arduino*, para el Servomotor se realiza la conexión en el pin 4 al *Arduino* y finalmente el led va conectado el ánodo a la resistencia para luego conectarla al pin 5 del *Arduino* y el cátodo a GND (ver figura 49). Asegurarse de que todos los componentes estén bien conectados a la placa base y listos para ser programados.

Figura 49

Conexión para realizar el funcionamiento de la puerta automática



Nota. En esta figura se visualiza la conexión realizada en Arduino.
Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Actividad 4: Programación del Proyecto

Dar una introducción a la Programación y explicar a los estudiantes cómo usar el sensor ultrasónico para medir la distancia y cómo activar el servomotor y el LED en función de esa medición.

En *mBlock* los estudiantes deben programar el *Arduino* para que, cuando la distancia medida sea menor a un valor determinado (por ejemplo, 10 cm), el servomotor mueva la puerta y el LED se encienda. Si la distancia es mayor, el servomotor cerrará la puerta y el LED se apagará (ver figura 50).

Figura 50

Código de la conexión de la puerta automática.



Nota. En esta figura se observa el código de la conexión realizada en *mBlock* para la puerta automática. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Actividad 5: Prueba y Ajustes

Los estudiantes deben cargar el código en el *Arduino* y verificar que el servomotor mueva la puerta y que el LED se encienda o apague correctamente según la proximidad del objeto al sensor. Si la puerta no se mueve correctamente o el LED no responde, deben ajustar la programación o la conexión de los componentes.

Actividad 6: Optimización y Presentación Final

Los estudiantes pueden hacer ajustes en la estructura de la puerta o mejorar la alineación del sensor ultrasónico para que funcione con mayor precisión. Cada grupo de estudiantes presentará su maqueta, demostrando cómo el

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
sistema funciona y explicando el proceso de diseño y programación que siguieron.

Reflexión Final

Los estudiantes deben reflexionar sobre los desafíos que enfrentaron durante el proyecto, cómo aplicaron el pensamiento computacional para solucionar problemas y qué mejorarían si tuvieran más tiempo o recursos.

3.7. Secuencia Didáctica 3: Identificación de animales carnívoros y herbívoros

Secuencia didáctica 3

“Identificación de animales carnívoros y herbívoros”

Objetivo General

Los estudiantes aprenderán a clasificar los animales en carnívoros y herbívoros mediante una maqueta interactiva, en la cual los LEDs indicarán si el animal seleccionado es de un tipo u otro al acercar la mano al sensor ultrasónico. Este proyecto integrará conceptos de biología, tecnología y programación mediante el uso de Arduino Nano y *mBlock*.

Presentación

Este proyecto está orientado a la enseñanza de la clasificación de animales según su alimentación. Es una forma de involucrar a los niños en la biología y la ciencia, mientras aprenden de manera práctica a diferenciar carnívoros y herbívoros. Utilizando circuitos electrónicos con Arduino, LEDs y un sensor ultrasónico, los estudiantes podrán explorar conceptos biológicos de forma interactiva.

Estructura de la secuencia didáctica

Sesión 1: Introducción a la alimentación de los animales y sus características

Sesión 2: Construcción de la maqueta y disposición de los animales

Sesión 3: Programación en *mBlock* (Interacción con el sensor ultrasónico y LEDs)

Sesión 4: Programación en *mBlock*

Sesión 5: Presentación y reflexión del proyecto

Recursos necesarios:

- Arduino NANO
- Placa Expansiva Nano
- Sensor ultrasónico HC-SR04
- 4 LEDs (2 rojos y 2 amarillos)
- Resistencias de 220Ω

- Cables de Conexión
- Cartulina o materiales reciclados para la maqueta
- Imágenes o figuras de animales (dos carnívoros y dos herbívoros)
- Computadoras con acceso a *mBlock*

Duración:

La secuencia didáctica se desarrollará en 5 sesiones de 2 horas cada una.

Sesión 1: Introducción a la alimentación de los animales y sus características

Objetivos:

Introducir a los estudiantes en el concepto de carnívoros y herbívoros.

Presentar los componentes electrónicos del proyecto (*Arduino*, Leds y sensor ultrasónico).

Actividades:

Iniciar una explicación sobre los carnívoros y herbívoros sobre sus diferencias en su alimentación, dientes y hábitos, además de realizar una lluvia de ideas de los animales que serán elegidos para la maqueta (dos carnívoros y dos herbívoros) y dar un color a cada clasificación.

Para hacerlo más dinámico y variado, pídeles que han grupos de 2 a 3 estudiantes donde ellos decidirán los recortes de animales que utilizaran.

Dar una retroalimentación sobre el *Arduino NANO*, Leds y Sensor ultrasónico, sus características además de dar ejemplos del funcionamiento de cada uno.

Evaluación de la sesión

Preguntas de reflexión sobre los tipos de alimentación y su importancia en la cadena alimenticia.

Sesión 2: Construcción de la maqueta y disposición de los animales

Objetivos:

Diseñar la maqueta con los animales ubicados en sus respectivas secciones.

Conectar los Leds y el sensor ultrasónico para representar cada tipo de animal.

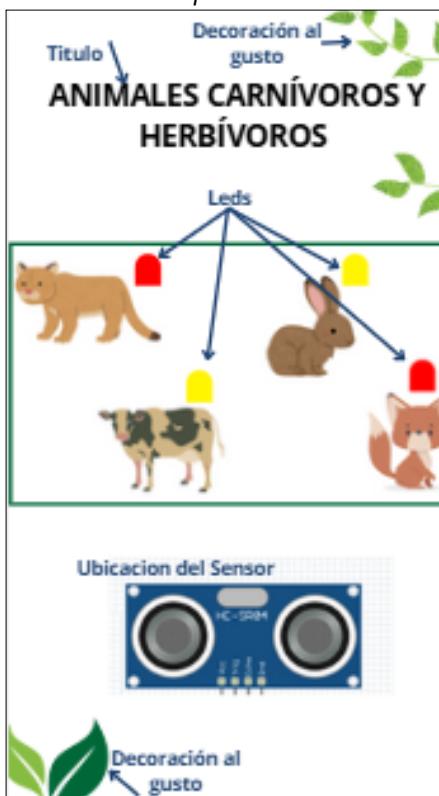
Actividades:

Diseño de la maqueta: los estudiantes decidirán la ubicación de los animales y leds, utilizando imágenes de los animales seleccionados, recortar y pegar en la base de la cartulina, colocarlo de manera desordenada, con el título correspondiente.

Dejar un espacio en la cartulina en cada animal y colocarle los leds encima de ellos con el color propuesto, por ejemplo: rojo para carnívoros, Amarillo para herbívoros. Ubicar al sensor ultrasónico en un punto estratégico para la detección en la maqueta, de preferencia en la parte inferior de la cartulina (figura 51).

Figura 51

Cartulina con los materiales implementados.



Nota. En esta figura se observa la representación de la cartulina ya con los implementos ubicados. Fuente: Elaboración propia con recursos gráficos de Canva (<https://www.canva.com>).

Evaluación de la sesión

Verificar que los estudiantes coloquen correctamente los Leds en la maqueta.

Sesión 3: Programación en mBlock (Interacción con el sensor ultrasónico y LEDs)

Objetivos:

Programar en *mBlock* el encendido de los Leds en función de la distancia del sensor ultrasónico.

Actividades:

Explicación del código en *mBlock* para encender Leds según la distancia medida por el sensor, cada led prenderá según el sensor. Si el objeto se acerca al sensor se deberá encender el led de los animales carnívoros y al alejarlo se encenderá el de los herbívoros y se apagará el de los carnívoros.

La conexión de los leds se llevará a cabo en los Pines 4 y 5 para el rojo: El Ánode del Led Rojo va conectado a la resistencia de 220, luego al Pin 4S y el Cátodo en el 4G, así mismo con el otro rojo ocupando el Pin 5S y 5G

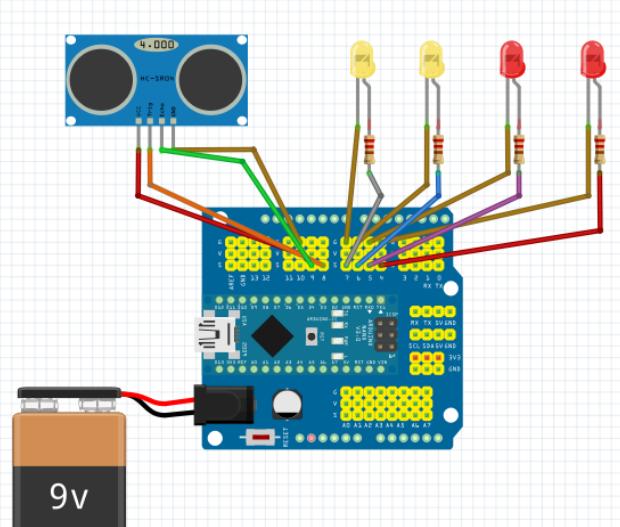
Para los leds Amarillos se utilizarán, los pines 6 y 7: El Ánode del Led Amarillo va conectado a la resistencia de 220, luego al Pin 6S y el Cátodo en el 6G, de igual manera con el otro led amarillo ocupando el Pin 7S y 7G

La conexión del Sensor ultrasónico realiza la conexión en: VCC: 10V, Trig: 8S, Echo: 9S y GND: 10G.

Todo va conectado al *Arduino NANO* tal y como se visualiza en la figura 52.

Figura 52

Conexión de los componentes



Nota. En esta figura se observa la conexión correspondiente de los leds y el sensor ultrasónico en el Arduino. Fuente: Captura de pantalla de la interfaz de Fritzing. Fritzing (<https://Fritzing.org>)

Prueba del código y corrección de errores.

Evaluación de la sesión

Verificar que los LEDs se enciendan correctamente al acercar la mano al sensor.

Sesión 4: Programación en *mBlock*

Objetivos:

Realizar la programación para el circuito electrónico en *mBlock*.

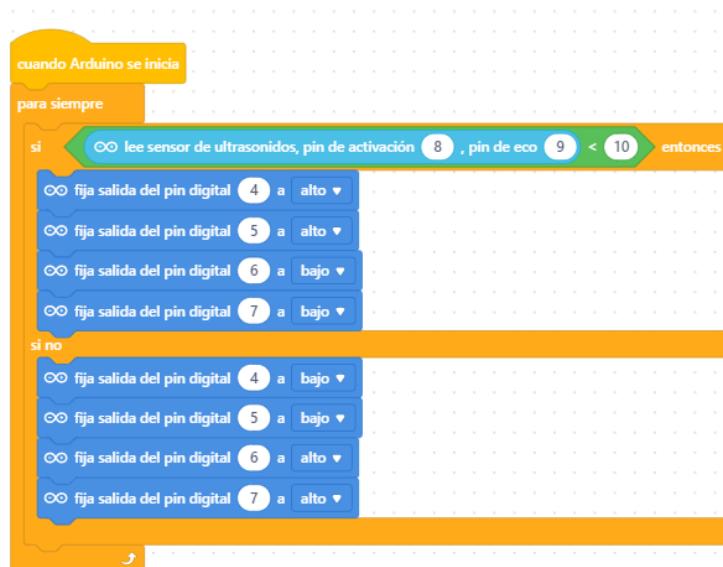
Reflexionar sobre el impacto de la alimentación de los animales en el ecosistema.

Actividades:

En *mBlock* realizar el código que cumpla con la función de que al acercar la mano o un objeto se prendan las luces amarillas, indicando a los animales herbívoros y al alejarla prenda los leds rojos de los carnívoros (figura 53).

Figura 53

Código para el sensor con leds



Nota. En esta figura se visualiza el código para la conexión. Fuente: Captura de pantalla de la interfaz de *mBlock* v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Evaluación de la sesión

Revisión del código y funcionamiento de la maqueta.

Preguntas abiertas sobre la importancia de la clasificación de los animales.

Sesión 5: Presentación y reflexión del proyecto

Objetivos:

Presentar la maqueta interactiva y explicar su funcionamiento.

Reflexionar sobre lo aprendido y cómo se integran ciencia y tecnología.

Actividades:

Para concluir con el proyecto puede realizar una exposición donde los estudiantes explicaran cómo funciona el sensor con los leds, además de realizar una explicación sencilla sobre los animales escogidos con la clasificación que corresponden cada uno y porque escogieron el color de los leds en cada clasificación.

Evaluación de la sesión

Verificación de la correcta comprensión del tema y funcionamiento de la maqueta. Evaluación final en la que los estudiantes respondan preguntas sobre carnívoros y herbívoros.

Conclusión

Este proyecto permite a los estudiantes explorar la clasificación de los animales de manera interactiva, combinando biología y tecnología. Al programar el sensor ultrasónico y los LEDs, los niños aprenden a aplicar la programación en problemas científicos reales, fomentando el aprendizaje basado en proyectos (ABP) y la metodología STEAM.

3.8. Secuencia Didáctica 4: Construcción de una maqueta de un puente levadizo

Secuencia didáctica 4

“Construcción de una maqueta de un puente levadizo”

Objetivo General

Los estudiantes aprenderán a construir una maqueta de un puente que se levanta al acercarse un objeto, utilizando un sensor ultrasónico para detectar la proximidad, un servomotor para levantar el puente y dos LEDs para indicar el estado del puente.

Presentación

Este proyecto está orientado a enseñar conceptos ingeniería, programación, matemáticas y física mediante un puente levadizo, utilizando un sensor ultrasónico, leds y servomotor puede ser muy divertido y educativo además de implementarlo en una maqueta divertida. Para guiar a los niños en la elaboración de este proyecto se puede armar una secuencia didáctica que integre la metodología STEAM y

Estructura de la secuencia didáctica

Sesión 1: Introducción al Proyecto y Diseño de la Maqueta

Sesión 2: Introducción a los Componentes Electrónicos y la conexión

Sesión 3: Programación de la Elevación del Puente y los LEDs

Sesión 4: Integración de la Maqueta y Ajustes del Sistema

R

Sesión 5: Presentación Final y Evaluación

- Placa Expansiva *Arduino NANO*
- Sensor ultrasónico HC-SR04
- Servomotor SG90
- 2 LEDs (uno amarillo para indicar que bajar, y uno rojo para indicar que subir)
- Resistencias de 220Ω
- Palitos de helado
- Silicona fría o caliente
- Pintura o marcadores
- Un pedazo de fomix o cartulina
- Cables
- Computadoras con acceso a *mBlock*

Duración:

La secuencia didáctica se llevará a cabo a lo largo de 5 sesiones de 2 horas cada una.

Sesión 1: Introducción al Proyecto y Diseño de la Maqueta

Objetivo: Introducir a los estudiantes en el concepto del puente levadizo automatizado y comenzar la construcción de la maqueta física del puente.

Actividades:

Dar inicio explicando qué es un puente levadizo y cómo el sensor ultrasónico puede detectar objetos para hacer que el puente se levante automáticamente, además de realizar seguir las luces ya sea rojo o amarilla.

Los estudiantes se dividen en grupos y empiezan a discutir y planificar cómo construirán el puente.

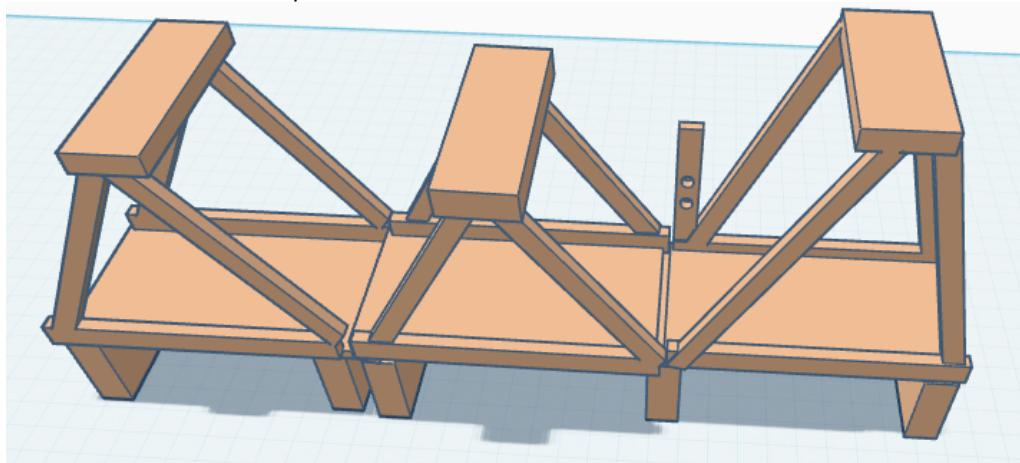
Construcción del Puente: Usando materiales como palitos de helado y pegamento, los estudiantes crean la estructura básica del puente, el cual podrán personalizarlo al gusto, dar indicaciones de la creación del puente.

1. Pegar ocho palos de helado de costado formando un piso o en fila. Hacer tres de esos pisos.
2. En los costados de cada piso, colocar un palito donde haciendo uso de varios palitos formar triángulos.
3. El piso que va en el centro pegar un lado al fomix o cartulina al piso siguiente para poder realizar el movimiento de arriba hacia abajo.
4. En la parte inferior poner columnas de palitos para tener altura.

Observa la figura 54 para tener una mejor idea.

Figura 54

Estructura de la maqueta



Nota. En esta figura se observa la estructura básica de la maqueta del puente elevadizo. Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

Además de discutir cómo funciona el servomotor y cómo el sensor ultrasónico ayudará a controlar el movimiento del puente y dejarle el lugar para los componentes.

Evaluación de la sesión

Se evalúa la creatividad y la organización del diseño de la maqueta. Además, se observa si los estudiantes comprenden los conceptos del puente levadizo, el funcionamiento del servomotor y la interacción del sensor ultrasónico.

Sesión 2: Introducción a los Componentes Electrónicos y la conexión.

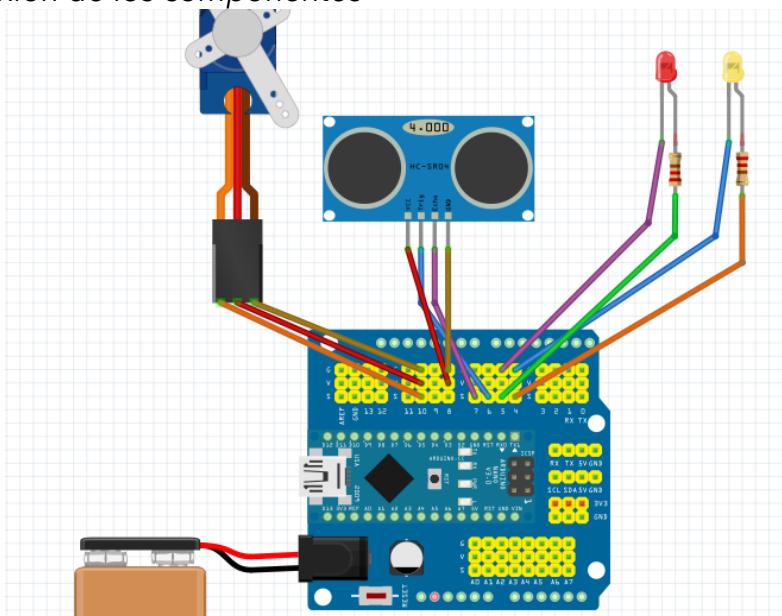
Objetivo: Familiarizar a los estudiantes con los componentes electrónicos (sensor ultrasónico, servomotor y LEDs) y realizar la conexión en el *Arduino NANO*.

Actividades:

Los estudiantes conectan el sensor ultrasónico, donde se ubicarán en los pines correspondientes, el servomotor (para mover el puente) y los LEDs (para indicar los estados de movimiento) a la placa *Arduino NANO*, además de realizar la prueba del sensor ultrasónico.

En el servomotor se utilizarán el Pin 10 (el naranja en el S, el rojo en V y el café en G), los leds utilizarán el Rojo pin 5 y el Amarillo el 4, en el caso del sensor ultrasónico se necesitarán varios pines, el VCC: 8V, el GND: 8G, TRig: 6S y Echo: 7S (ver figura 55).

Figura 55
Conexión de los componentes



Nota. En esta figura se visualiza la conexión realizada con todos los componentes. Fuente: Captura de pantalla de la interfaz de *Fritzing*. *Fritzing* (<https://Fritzing.org>)

Evaluación de la sesión

Se evalúa la correcta conexión de los componentes y la capacidad de los estudiantes para escribir y probar un código básico de lectura del sensor ultrasónico.

Sesión 3: Programación de la Elevación del Puente y los LEDs

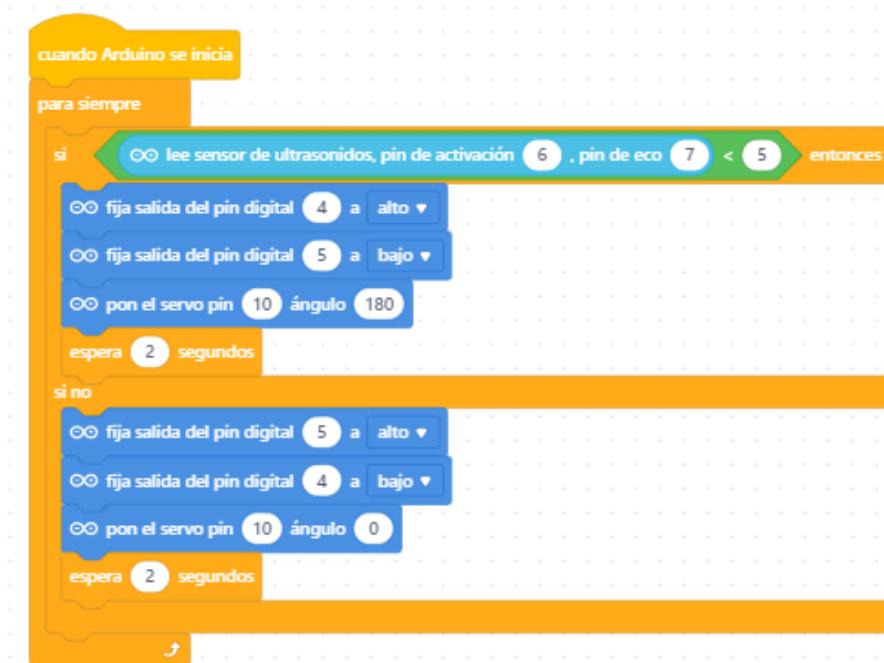
Objetivo: Programar el puente para que se eleve automáticamente cuando el sensor detecte un objeto cercano, y para que los LEDs se enciendan según el estado del puente (subiendo o bajando).

Actividades:

Programación del Control de Movimiento: Escribir el código para que el servomotor mueva el puente cuando el sensor ultrasónico detecte un objeto a una distancia específica. Uso de LEDs: Programar dos LEDs: Amarillo, para indicar que el puente está bajando y Rojo, para indicar que el puente está subiendo (ver figura 56).

Figura 56

Programación hecha en mBlock



Nota. En esta figura se observa la programación realizada en mBlock donde se realiza la conexión del sensor ultrasónico, leds y servomotor. Fuente: Captura de pantalla tomada de la interfaz de mBlock v5.4.0. Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Prueba del Sistema: Los estudiantes prueban el código para asegurarse de que el puente se eleve al acercarse un objeto y que los LEDs cambien de acuerdo al movimiento del puente.

Evaluación de la sesión

Se evalúa si el código permite que el puente se levante y baje según lo detectado por el sensor ultrasónico, y si los LEDs se encienden correctamente según el estado del puente.

Sesión 4: Integración de la Maqueta y Ajustes del Sistema

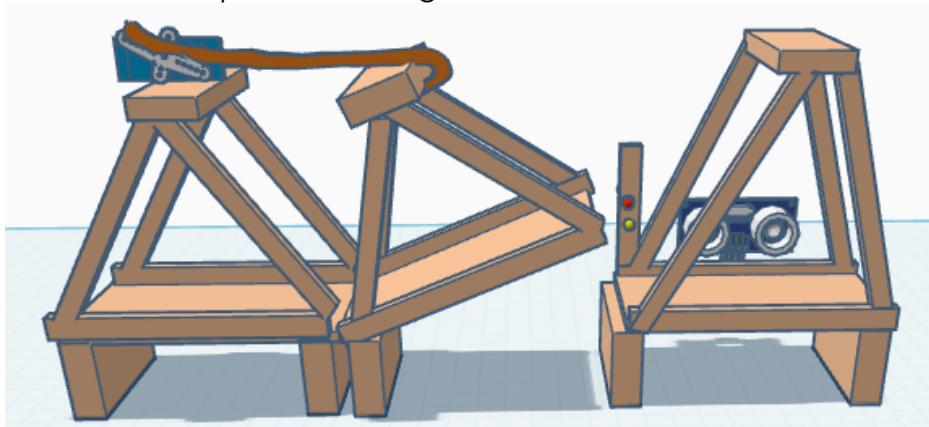
Objetivo: Completar la maqueta física del puente y realizar ajustes en la programación y el sistema para asegurar que funcione correctamente.

Actividades:

Los estudiantes completan la maqueta física del puente, asegurándose de que el servomotor y los LEDs estén bien integrados en la estructura (ver figura 57). En caso de ser necesarios realizar ajustes en la programación para mejorar la precisión del sensor ultrasónico y asegurarse de que el puente se levante y baje suavemente.

Figura 57

Maqueta con los componentes integrados.



Nota. En esta figura se visualiza la maqueta con los componentes. Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

Realizan pruebas del sistema completo, asegurándose de que el puente se levante al detectar un objeto y que los LEDs se iluminen correctamente (Amarillo para Baja, Rojo para Sube).

Evaluación de la sesión

Se evalúa la integración de los componentes en la maqueta y la efectividad de los ajustes realizados.

Sesión 5: Presentación Final y Evaluación

Objetivo: Los estudiantes presentan sus proyectos finales, explican cómo funciona su puente y reflexionan sobre el proceso de aprendizaje.

Actividades:

Cada grupo presenta su maqueta de puente, explica cómo se construyó y cómo funciona el Sensor ultrasónico, el servomotor y los LEDs.

Realizar una reflexión sobre lo que aprendieron durante el proceso, discuten los desafíos que enfrentaron y cómo los solucionaron.

Evaluación Final: Los proyectos se evalúan según los siguientes criterios:

Funcionamiento correcto del puente (movimiento al detectar el objeto).

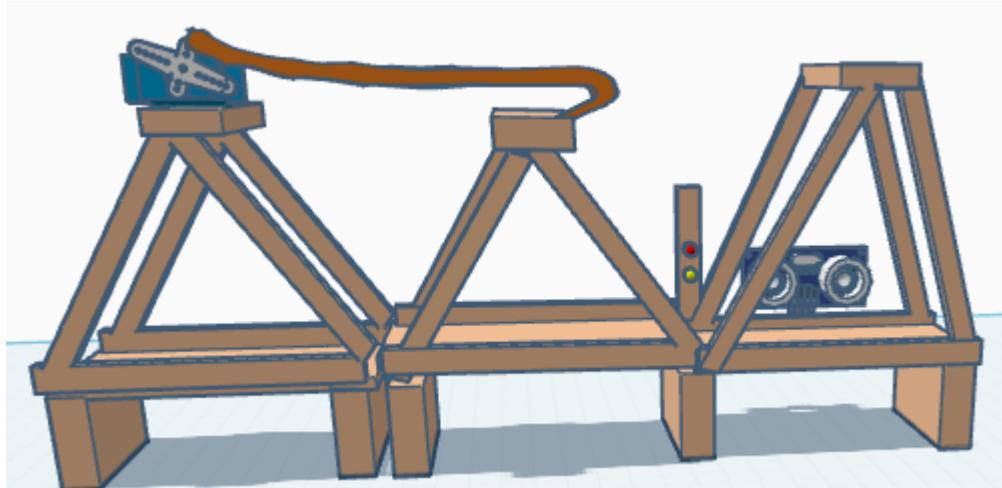
Uso adecuado de los LEDs y del Servomotor.

Creatividad y calidad en la presentación de la maqueta.

Comprensión del funcionamiento del Sensor ultrasónico, el servomotor y los LEDs. Tal y como se demuestra la figura 58 y la figura 59.

Figura 58

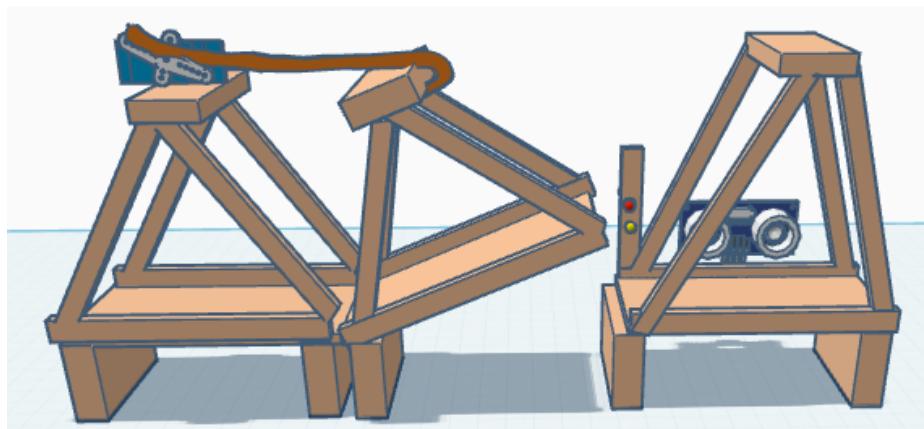
Maqueta en funcionamiento con LED Amarillo con sensor ultrasónico



Nota. En esta figura se observa el funcionamiento del LED amarillo siguiendo las órdenes del sensor en bajar. Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

Figura 59

Maqueta en funcionamiento del LED Rojo con el Sensor ultrasónico



Nota. En esta figura se visualiza el funcionamiento del LED rojo siguiendo las órdenes del Sensor ultrasónico. Fuente: Captura de pantalla generada en **Tinkercad**. Autodesk, Inc. (<https://www.Tinkercad.com>)

Evaluación de la sesión

La evaluación se basa en el funcionamiento del proyecto (si el puente se levanta correctamente, el uso de los LEDs y la integración de todos los componentes), la creatividad en el diseño de la maqueta y la capacidad de los estudiantes para explicar su proyecto.

Conclusión:

Este proyecto proporciona a los estudiantes una experiencia práctica en tecnología, ingeniería, física y matemáticas, permitiéndoles aprender cómo los sensores, los servomotores y los LEDs trabajan en conjunto para crear una solución automatizada. Además, fomenta el trabajo en equipo, la

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
creatividad y la resolución de problemas, habilidades que son fundamentales en el mundo actual.

3.9. Resumen del capítulo

Este capítulo aborda el aprendizaje creativo de la programación con sensores ultrasónicos mediante el Aprendizaje Basado en Proyectos (ABP). Se presentan secuencias didácticas que integran las cinco áreas STEAM: Ciencia, Tecnología, Ingeniería, Arte y Matemática, fomentando la experimentación, la creatividad y el pensamiento crítico. Cada propuesta combina la programación en *mBlock* con aplicaciones prácticas, desde la exploración del sonido hasta el diseño de sistemas mecánicos, permitiendo que los futuros docentes adapten y apliquen estrategias innovadoras para motivar y desarrollar competencias en estudiantes de tercer año de EGB.

3.10. Desafíos prácticos propuestos con sensores ultrasónicos

Estos desafíos están pensados para ampliar y profundizar las aplicaciones prácticas vistas en el capítulo, añadiendo condiciones, combinaciones de funciones y pequeños elementos de programación que aumentan la complejidad sin introducir nuevos componentes electrónicos. De esta forma los estudiantes desarrollan su pensamiento computacional de forma creativa.

Desafío 1: Puerta automática con luz de cortesía y cierre temporizado

Diseñar un sistema que, al detectar la presencia de una persona, abra una puerta mediante servomotor y encienda un LED como luz de cortesía. Después de un tiempo determinado, el sistema debe cerrar la puerta automáticamente y apagar la luz.

Relación STEAM:

- Ciencia: uso del sensor ultrasónico para medir distancia y detectar presencia.
- Tecnología y Programación: implementación de temporizadores para apertura y cierre automáticos.
- Ingeniería: integración del LED y servomotor en la estructura de la puerta.
- Arte: diseño estético de la entrada y su iluminación.
- Matemáticas: configuración del tiempo de encendido de la luz y del ciclo de apertura/cierre.

Desafío 2: Barrera automática con semáforo de espera y conteo de accesos

Construir una barrera que se abra con un servomotor al detectar un objeto cercano, mientras un LED rojo indica espera y un LED verde señala paso libre. El sistema llevará un conteo de cuántas veces se ha abierto la barrera.

Relación STEAM:

- Ciencia: detección de proximidad con ultrasonido y conversión a señales visuales.
- Tecnología y Programación: uso de variables para registrar conteos.
- Ingeniería: montaje del mecanismo de barrera y su soporte.
- Arte: diseño visual del semáforo y la base.
- Matemáticas: registro de datos y totales de accesos.

Desafío 3: Control de acceso con aviso de tiempo restante

Crear un sistema que abra una compuerta al detectar una persona, pero que indique con un LED parpadeante cuando faltan pocos segundos para que se cierre.

Relación STEAM:

- Ciencia: medición de distancia y control de señales visuales.

- Tecnología y Programación: combinación de temporizadores y parpadeo de LED.
- Ingeniería: acoplamiento del servomotor a la compuerta.
- Arte: diseño del entorno de acceso.
- Matemáticas: cálculo del tiempo de parpadeo previo al cierre.

Desafío 4: Paso peatonal inteligente

Desarrollar un sistema que detecte la presencia de peatones, active un LED verde para indicar paso y mueva un servomotor que baje una barrera lateral. Después de unos segundos, volver a la posición inicial con LED rojo encendido.

Relación STEAM:

- Ciencia: detección de proximidad y señalización visual.
- Tecnología y Programación: coordinación entre dos salidas (LED y motor) con temporizadores.
- Ingeniería: instalación de la barrera y luces.
- Arte: decoración del paso peatonal en maqueta.
- Matemáticas: determinación de tiempos seguros para el cruce.

Desafío 5: Portón doble con apertura secuencial

Diseñar un sistema con dos servomotores que simulen puertas dobles, programadas para abrirse una después de la otra al detectar proximidad. Un LED se enciende cuando el portón está completamente abierto.

Relación STEAM:

- Ciencia: sincronización de acciones mecánicas y señales visuales.
- Tecnología y Programación: programación de secuencias con retardo.
- Ingeniería: montaje de dos servomotores en estructura de portón.

- Arte: presentación estética del portón.
- Matemáticas: cálculo del intervalo entre la apertura de cada hoja.

3.11. Autoevaluación del capítulo

Pregunta 1: ¿Cuál es la principal característica del Aprendizaje Basado en Proyectos (ABP)?

- a) Se basa en la memorización de contenidos.
- b) Se enfoca en la producción de un resultado o solución concreta.
- c) No permite el trabajo en equipo.
- d) Solo se evalúa el producto final del proyecto.

Pregunta 2: ¿Cuál es el propósito de la fase de exploración o introducción al proyecto?

- a) Elaborar un informe final.
- b) Evaluar a los estudiantes antes de comenzar el proyecto.
- c) Despertar el interés y activar conocimientos previos.
- d) Asignar calificaciones sin considerar el proceso.

Pregunta 3: ¿Por qué es importante la fase de planificación en el ABP?

- a) Porque permite organizar las tareas y establecer un plan de trabajo.
- b) Porque se eligen los estudiantes que van a participar en el proyecto.
- c) Porque en esta fase se presenta el producto final.
- d) Porque se asignan exámenes para evaluar el conocimiento teórico.

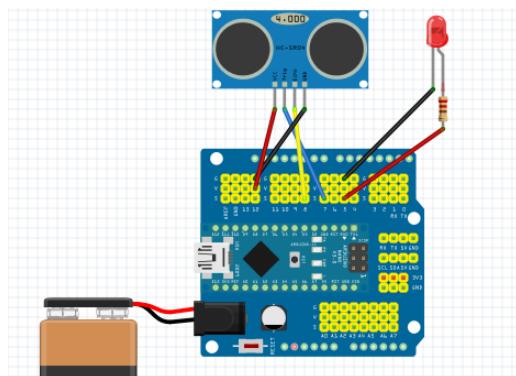
Pregunta 4: Ubica en los siguientes cuadros las fases del Aprendizaje Basado en Proyectos.



Pregunta 5: Analiza el funcionamiento del circuito junto a su código, que se muestra en el siguiente gráfico, y escribe su funcionamiento y en qué proyecto se lo podría implementar.

Figura 60

Bloques en mBlock y circuito en Fritzing para analizar el funcionamiento



Nota. En esta figura se visualiza bloques en *mBlock* y circuito en *Fritzing* para analizar el funcionamiento. Fuente: Captura compuesta de las interfaces de *Fritzing* y *mBlock*. *Fritzing* (<https://Fritzing.org>) y Makeblock Co., Ltd. (<https://www.mBlock.cc>)

Pregunta 6: ¿Qué se debe hacer primero al programar un sensor ultrasónico en *mBlock*?

- a) Conectar el sensor al puerto correcto y seleccionar el bloque de distancia ultrasónica
- b) Encender el motor del robot
- c) Ajustar la velocidad de la luz
- d) Crear un bloque de repetición infinita sin condiciones

Pregunta 7: En *mBlock*, ¿cómo puede un programador usar el sensor ultrasónico para evitar obstáculos?

- a) Dibujando una línea en la pantalla
- b) Programando una condición "si la distancia es menor a un valor, entonces girar"
- c) Usando un bloque de sonido para alertar
- d) Ajustando la luz LED del robot

Pregunta 8: En un proyecto ABP que utiliza un sensor ultrasónico, ¿cuál podría ser un objetivo principal?

- a) Crear un videojuego de carreras
- b) Construir un sistema automático que detecte y evite obstáculos
- c) Diseñar una página web de recetas
- d) Dibujar figuras geométricas en papel

CRÉDITOS DE SOFTWARE Y RECURSOS VISUALES

Las imágenes contenidas en este libro fueron generadas mediante herramientas de simulación, diseño y programación educativa. Se agradece a las siguientes plataformas:

- **mBlock:** (Makeblock Co., Ltd.): Utilizado para crear simulaciones de programación por bloques. Algunas imágenes muestran sprites y componentes visuales propiedad de Makeblock. Sitio oficial: <https://www.mBlock.cc>
- **Tinkercad:** (Autodesk, Inc.): Utilizado para diseñar y capturar modelos 3D y circuitos electrónicos. Las imágenes incluyen componentes prefabricados proporcionados por la biblioteca gráfica de **Tinkercad**. Sitio oficial: <https://www.Tinkercad.com>
- **Fritzing:** Utilizado para esquemas eléctricos en vistas de protoboard, esquemático y PCB. Las capturas de pantalla fueron generadas por el autor a partir de sus propios diseños. Sitio oficial: <https://Fritzing.org>

Todas las imágenes se usan con fines ilustrativos y educativos. Se respetan los términos de uso y propiedad intelectual de cada plataforma. En caso de reutilización, consulte las licencias individuales de cada software.

REFERENCIAS BIBLIOGRÁFICAS

Addas, A., Naseer, F., Tahir, M., & Khan, M. N. (2024). Enhancing Higher-Education Governance Through Telepresence Robots and Gamification: Strategies for Sustainable Practices in the AI-Driven Digital Era. *Education Sciences*, 14(12), Article 12.

<https://doi.org/10.3390/educsci14121324>

Åkerfeldt, A., Kjällander, Susanne, & and Petersen, P. (2024). A research review of computational thinking and programming in education. *Technology, Pedagogy and Education*, 33(3), 375-390.

<https://doi.org/10.1080/1475939X.2024.2316087>

Alarfaj, M., Mohamed, S. R., Chtourou, S., Enshasy, H., Aboulnaga, A., & Hassan, M. (2024). Experience of Project-Based Learning: Challenges, Assessment, and Analysis. *International Journal of Engineering Pedagogy (iJEP)*, 14(3), Article 3. <https://doi.org/10.3991/ijep.v14i3.43849>

Alenezi, A. (2024). TEACHING COMPUTATIONAL THINKING IN SAUDI ARABIAN SCHOOLS USING SMARTPHONE APPS.

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Information Technologies and Learning Tools, 103(5),

Article 5. <https://doi.org/10.33407/itlt.v103i5.5811>

Alonso-García, S., Fuentes, A.-V. R., Navas-Parejo, M. R., &

Victoria-Maldonado, J.-J. (2024). Enhancing computational thinking in early childhood education with educational robotics: A meta-analysis. *Helijon*, 10(13).

<https://doi.org/10.1016/j.heliyon.2024.e33249>

Arpacı, I., Dogru, M. S., Kanj, H., Ali, N., & Baharı, M. (2023). An

Experimental Study on the Implementation of a STEAM-Based Learning Module in Science Education.

Sustainability, 15(8), Article 8.

<https://doi.org/10.3390/su15086807>

Breda, A., Garcia, V., & Santos, N. (2023). Teachers' Perceptions

of STEAM Education. *International Journal of Technology*

in Education, 6(4), Article 4.

<https://doi.org/10.46328/ijte.563>

Chang, Y., Choi, J., & Şen-Akbulut, M. (2024). Undergraduate

Students' Engagement in Project-Based Learning with an

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Authentic Context. *Education Sciences*, 14(2), Article 2.

<https://doi.org/10.3390/educsci14020168>

Coenraad, M., Rangel ,Alessandra, & and Dunbar, K. (2025).

Utilizing a computational thinking engagement inventory
to support inclusive computational thinking pathways.

Journal of Research on Technology in Education, 57(1),

201-217.

<https://doi.org/10.1080/15391523.2024.2398520>

Dong, J., Yu, S., Choi, K., & Jeon, M. (2025). A child-robot theater
afterschool program can promote children's
conceptualization of social robots' mental capacities and
engagement in learning. *Frontiers in Robotics and AI*, 12.

<https://doi.org/10.3389/frobt.2025.1529421>

Dúo-Terrón, P. (2023). Analysis of Scratch Software in Scientific
Production for 20 Years: Programming in Education to
Develop Computational Thinking and STEAM Disciplines.
Education Sciences, 13(4), 404.

<https://doi.org/10.3390/educsci13040404>

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Fitzgerald, M. S., & Evans, K. B. (2024). Integrating Digital Tools

to Enhance Access to Learning Opportunities in Project-based Science Instruction. *TechTrends*, 68(5), 882-891.

<https://doi.org/10.1007/s11528-024-00975-w>

Fuentes, V. M. (2014). *Introducción a la plataforma Arduino y al sensor ultrasónico HC-SR04*.

García Fuentes, O., Raposo Rivas, M., & Figueira, M. E. M. (2023).

El enfoque educativo STEAM: Una revisión de la literatura.

Revista Complutense de Educación, 34(1), Article 1.

<https://doi.org/10.5209/rced.77261>

Hao, L., Tian, K., Salleh, U. K. M., Chin, H. L., Ge, S., & Cheng, X. (2024). THE EFFECT OF PROJECT-BASED LEARNING AND PROJECT-BASED FLIPPED CLASSROOM ON CRITICAL THINKING AND CREATIVITY FOR BUSINESS ENGLISH COURSE AT HIGHER VOCATIONAL COLLEGES. *Malaysian Journal of Learning and Instruction*, 21(1), Article 1.

<https://doi.org/10.32890/mjli2024.21.1.6>

Herrero, A. C., Recio, T., Tolmos, P., & Vélez, M. P. (2023). From the Steam Engine to STEAM Education: An Experience

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

with Pre-Service Mathematics Teachers. *Mathematics*,

11(2), Article 2. <https://doi.org/10.3390/math11020473>

Herrero-Álvarez, R., Arnay, R., Segredo, E., Miranda, G., & León,

C. (2025). Using RoblockLLy in the Classroom: Bridging

the Gap in Computer Science Education Through Robotics

Simulation. *IEEE Transactions on Learning Technologies*,

18, 39-52. <https://doi.org/10.1109/TLT.2024.3520329>

Hijón-Neira, R., Pizarro, C., French, J., Palacios-Alonso, D., &

Çoban, E. (2024). Computational Thinking Measurement

of CS University Students. *Applied Sciences*, 14(12), Article

12. <https://doi.org/10.3390/app14125261>

Huang, C.-Y., Cheng, B.-Y., Lou, S.-J., & Chung, C.-C. (2023).

Design and Effectiveness Evaluation of a Smart

Greenhouse Virtual Reality Curriculum Based on STEAM

Education. *Sustainability*, 15(10), Article 10.

<https://doi.org/10.3390/su15107928>

Júnior, J. V. de M., Silva, R. M. G. da, & Medeiros, B. C. (2024).

Benefícios da abordagem ágil para Aprendizagem

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Baseada em Projetos (ABPJ). *Revista de Gestão e Projetos*,

15(2), Article 2. <https://doi.org/10.5585/gep.v15i2.25676>

Kalaitzidou, M., & Pachidis, T. P. (2023). Recent Robots in STEAM Education. *Education Sciences*, 13(3), Article 3.

<https://doi.org/10.3390/educsci13030272>

Leavy, A., Dick, L., Meletiou-Mavrotheris, M., Paparistodemou, E., & Stylianou, E. (2023). The prevalence and use of emerging technologies in STEAM education: A systematic review of the literature. *Journal of Computer Assisted Learning*, 39(4), 1061-1082.

<https://doi.org/10.1111/jcal.12806>

Leung, A. C. Y., Liu, D. Y. W., Luo, X., & Au, M. H. (2024). A constructivist and pragmatic training framework for blockchain education for IT practitioners. *Education and Information Technologies*, 29(12), 15813-15854.

<https://doi.org/10.1007/s10639-024-12505-5>

Liao, J., Zhong, L., Zhe, L., Xu, H., Liu, M., & Xie, T. (2024).

Scaffolding Computational Thinking With ChatGPT. *IEEE*

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

Transactions on Learning Technologies, 17, 1628-1642.

<https://doi.org/10.1109/TLT.2024.3392896>

Liu, Z., Gearty, Z., Richard, E., Orrill, C. H., Kayumova, S., &

Balasubramanian, R. (2024). Bringing computational

thinking into classrooms: A systematic review on

supporting teachers in integrating computational thinking

into K-12 classrooms. *International Journal of STEM*

Education, 11(1), 51. <https://doi.org/10.1186/s40594-024-00510-6>

Navarro Arcas, A. R., Llorca Schenk, J. M., Jara, C. A., & Sentana

Gadea, I. (2024). Educational Impacts on Robotic

Engineering Students of an International Online Project-

Based Learning Course. *Education Sciences, 14(12),*

Article 12. <https://doi.org/10.3390/educsci14121305>

Olivato, J. A., & Silva, J. C. (2023). Interdisciplinary teaching

practices in STEAM education in Brazil. *London Review of*

Education, 21(1), Article 1.

<https://doi.org/10.14324/LRE.21.1.38>

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula
Palma, N. M., Lucena, F. J. H., Rodríguez, J. M. R., & Navas-Parejo,

M. R. (2024). Desarrollo del pensamiento computacional "desenchufado" mediante resolución de problemas: Una revisión sistemática y meta-análisis. *Revista de Investigación en Educación*, 22(3), Article 3.

<https://doi.org/10.35869/reined.v22i3.5755>

Pan, Y., Adams, E. L., Ketterlin-Geller, L. R., Larson, E. C., & Clark, C. (2024). Enhancing middle school students' computational thinking competency through game-based learning. *Educational Technology Research and Development*, 72(6), 3391-3419.

<https://doi.org/10.1007/s11423-024-10400-x>

Rodrigues-Silva, J., & Alsina, Á. (2023). STEM/STEAM in Early Childhood Education for Sustainability (ECEfS): A Systematic Review. *Sustainability*, 15(4), Article 4.

<https://doi.org/10.3390/su15043721>

Román-Graván, P., & Arrifano-Tadeu, P.-J. (2025). La robótica en la formación docente universitaria: Un análisis comparativo de las percepciones entre España y Portugal.

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

IJERI: International Journal of Educational Research and

Innovation, 23. <https://doi.org/10.46661/ijeri.11021>

Salsabila, N. M., & Baroroh, R. U. (2024). Assessment Of Arabic Writing Skills In Differentiated Learning Based On Project-Based Learning. *Ijaz Arabi Journal of Arabic Learning*, 7(2), Article 2. <https://doi.org/10.18860/ijazarabi.v7i2.25429>

Sanz-Camarero, R., Ortiz-Revilla, J., & Greca, I. M. (2023). The Impact of Integrated STEAM Education on Arts Education: A Systematic Review. *Education Sciences*, 13(11), Article 11. <https://doi.org/10.3390/educsci13111139>

Seo, S., Van Orman, D. S. J., Beattie, M., Paxson, L. C., & Murray, J. (2024). Transforming Learning Orientations Through STEM Interdisciplinary Project-Based Learning. *Education Sciences*, 14(11), Article 11.

<https://doi.org/10.3390/educsci14111154>

Shanshan, S., & Sen, G. (2024). Empowering learners with AI-generated content for programming learning and computational thinking: The lens of extended effective use

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

theory. *Journal of Computer Assisted Learning*, 40(4),

1941-1958. <https://doi.org/10.1111/jcal.12996>

Sherwood, H., Culp, K. M., Ferguson, C., Kaiser, A., Henry, M., &

Negron, A. (2024). Teacher Practices for Formatively

Assessing Computational Thinking with Early Elementary

Learners. *Education Sciences*, 14(11), Article 11.

<https://doi.org/10.3390/educsci14111250>

Silva-Hormazábal, M., & Alsina, Á. (2023). Exploring the Impact of

Integrated STEAM Education in Early Childhood and

Primary Education Teachers. *Education Sciences*, 13(8),

Article 8. <https://doi.org/10.3390/educsci13080842>

Sircar, M., Orr, S., Madis, C., & DiMaria, S. (2024).

Conceptualizing an Initial Framework to Support

Discipline-Rich Project-Based Learning in STEM. *Education*

Sciences, 14(7), Article 7.

<https://doi.org/10.3390/educsci14070793>

Sun, D., Looi, C.-K., Li, Y., Zhu, C., Zhu, C., & Cheng, M. (2024).

Block-based versus text-based programming: A

comparison of learners' programming behaviors,

Enseñanza y aprendizaje del pensamiento computacional: exploración de sensores ultrasónicos en proyectos educativos. Aplicaciones prácticas para el aula

computational thinking skills and attitudes toward

programming. *Educational Technology Research and*

Development, 72(2), 1067-1089.

<https://doi.org/10.1007/s11423-023-10328-8>

Sun, L., You, X., & Zhou, D. (2023). Evaluation and development

of STEAM teachers' computational thinking skills: Analysis

of multiple influential factors. *Education and Information*

Technologies, 28(11), 14493-14527.

<https://doi.org/10.1007/s10639-023-11777-7>

Suparman, S., Juandi, D., Turmudi, T., Martadiputra, B. A. P.,

Helsa, Y., Masniladev, M., & Suherman, D. S. (2025).

Computational Thinking in Mathematics Instruction

Integrated to STEAM Education: A Systematic Review and

Meta-Analysis. *TEM Journal*, 949-963.

<https://doi.org/10.18421/TEM141-84>

Terrón, P. D. (2025). Exploring STEAM disciplines with robotics in

the Future Classroom Lab in Primary Education. *Digital*

Education Review, 47, 141-153.

<https://doi.org/10.1344/der.2025.47.141-153>

Thoma, R., Farassopoulos, N., & Lousta, C. (2023). *Teaching STEAM through universal design for learning in early years of primary education: Plugged-in and unplugged activities with emphasis on connectivism learning theory. Teaching and Teacher Education, 132*, 104210.

<https://doi.org/10.1016/j.tate.2023.104210>

Torres, I., & Inga, E. (2025). Fostering STEM Skills Through Programming and Robotics for Motivation and Cognitive Development in Secondary Education. *Information, 16*(2), Article 2. <https://doi.org/10.3390/info16020096>

Zapata-Cáceres, M., Marcelino, P., El-Hamamsy, L., & Martín-Barroso, E. (2024). A Bebras Computational Thinking (ABC-Thinking) program for primary school: Evaluation using the competent computational thinking test.

Education and Information Technologies, 29(12), 14969-14998. <https://doi.org/10.1007/s10639-023-12441-w>

Julio Antonio Encalada Cuenca
jencalada@utmachala.edu.ec

Universidad Técnica de Machala, Facultad de Ciencias Sociales, Carrera de Pedagogía de las Ciencias Experimentales <https://orcid.org/0000-0002-8120-2047> Julio Encalada es docente e investigador en la Universidad Técnica de Machala. Magíster en Educación Informática, cuenta con amplia experiencia en docencia e investigación en el campo de la robótica educativa, el pensamiento computacional y la metodología STEAM. Su trabajo promueve la integración de tecnologías en entornos pedagógicos innovadores y orientados al aprendizaje activo.

Sara Gabriela Cruz Naranjo
scruz@utmachala.edu.ec

Universidad Técnica de Machala, Facultad de Ciencias Sociales, Carrera de Pedagogía de las Ciencias Experimentales <https://orcid.org/0000-0002-8772-2051> Sara Cruz es magíster en Tecnología e Innovación Educativa, es docente titular en la Universidad Técnica de Machala, comprometida con la transformación de los procesos de enseñanza-aprendizaje. Su labor impulsa la integración de tecnologías en el ámbito educativo, fomentando metodologías innovadoras que fortalecen la formación académica y el desarrollo de competencias digitales en sus estudiantes.

Nathaly Dayana Torres Bustamante
ntorres10@utmachala.edu.ec

Universidad Técnica de Machala, Facultad de Ciencias Sociales, Carrera de Pedagogía de las Ciencias Experimentales <https://orcid.org/0009-0003-5740-3433> Nathaly Torres es investigadora independiente, enfocada en el desarrollo de proyectos educativos y tecnológicos. Su trabajo impulsa la innovación en procesos de enseñanza-aprendizaje, promoviendo el uso de herramientas digitales. Comprometida con la formación integral, aporta conocimientos que fortalecen las competencias digitales en contextos educativos contemporáneos.

Compás
capacitación e investigación

ISBN: 978-9942-53-001-1



9 789942 530011