

INFORME DE TESIS - QUISPE & BRINGAS

por ERICK DANGELO QUISPE GUTIERREZ

Fecha de entrega: 30-abr-2024 04:29p.m. (UTC-0500)

Identificador de la entrega: 2216666999

Nombre del archivo: TESIS,_BRINGAS_QUISPE_2023.docx (9.35M)

Total de palabras: 10840

Total de caracteres: 60572

¹
**UNIVERSIDAD CATÓLICA DE TRUJILLO
BENEDICTO XVI**

**FACULTAD DE INGENIERÍA Y ARQUITECTURA
PROGRAMA DE ESTUDIOS DE INGENIERÍA
MECÁNICO ELÉCTRICA**



**“INFLUENCIA DE UN CÓDIGO DE PROGRAMACIÓN EN
ARDUINO PARA LA D²⁰ETECCIÓN DE OBJETOS EN BASTÓN
PARA INVIDENTES EN EL DISTRITO DE MOCHE 2023”**

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE
INGENIERO MECÁNICO ELÉCTRICO**

AUTORES

Bringas Chacón, Walter Scott
Quispe Gutiérrez, Erick Dangelo

ASESOR

Ing. Saldaña Milla, Fernando Arístides

<https://orcid.org/0000-0002-7274-6974>

LÍNEA DE INVESTIGACIÓN

Procesos y Tecnología

**TRUJILLO – PERÚ
2023**

DECLARATORIA DE ORIGINALIDAD

²³ Señor(a) Decano(a) de la Facultad de Ingeniería:

Yo Mg. Ing. Fernando Arístides Saldaña Milla ²³ con DNI N° 18135414, como asesor del trabajo de investigación **“INFLUENCIA DE UN CÓDIGO DE PROGRAMACIÓN EN ARDUINO PARA LA DETECCIÓN DE OBJETOS EN BASTÓN PARA INVIDENTES”** desarrollado por los bachilleres Walter Scott Bringas Chacón con DNI N° 72049900 y Erick Dangelo Quispe Gutiérrez con DNI N° 74080308, ² egresados del Programa Profesional de Mecánica Eléctrica., considero que dicho trabajo de titulación reúne los requisitos tanto técnicos como científicos y corresponden con las normas establecidas en el reglamento de titulación de la Universidad Católica de Trujillo Benedicto XVI y en normativa para la presentación de trabajos de titulación de la Facultad de Ingeniería y Arquitectura. Por tanto, autorizo la presentación del mismo ante el organismo pertinente para que sea sometido a evaluación por la comisión de la clasificación designado por el Decano de la Facultad de Ingeniería y Arquitectura.

³ Se declara también que el porcentaje de similitud o coincidencia es menor o igual ¹ al 20 %, estándar permitido por el Reglamento de grados y títulos de la Universidad Católica de Trujillo.



Mg. Ing. Fernando Arístides Saldaña Milla
ASESOR

AUTORIDADES UNIVERSITARIAS

Mons. Dr. Héctor Miguel Cabrejos Vidarte, O.F.M.

Arzobispo Metropolitano de Trujillo

Fundador y Gran Canciller de la Universidad Católica Trujillo Benedicto XVI

Mons. Dr. Luis Miranda Díaz

Rector de la Universidad Católica de Trujillo Benedicto XVI

Dra. Mariana Silva Balarezo

Vicerrectora Académica

Dra. Ena Cecilia Obando Peralta

Vicerrectora de Investigación

2

Mg. Breitner Díaz Rodríguez

Decano de la Facultad de Ingeniería y Arquitectura

Mons. Ricardo Exequiel Angulo Bazauri

Gerente de Desarrollo Institucional

Dra. Teresa Reátegui Marín

Secretaria General

DEDICATORIA

A DIOS

*Por permitir tener unida a mi familia,
Por siempre cuidarnos y brindarnos salud,
Por nunca dejarnos de lado.*

A MI MADRE

*Por permitirme que nunca pase hambre,
Que siempre tenga un techo donde pasar la noche,
Por cuidarme y brindarme amor incondicional,
Por ser el mejor ejemplo de la palabra MADRE.*

A MI PADRE

*Por poner las bases de conocimiento y aprendizaje,
Por enseñarme la cultura del trabajo y del esfuerzo,
Y enseñarme que no existe la palabra cansancio.
Por ser el mejor ejemplo de la palabra PADRE.*

A MI HERMANO

*Por permitirme ser su ejemplo,
Por motivarme a ser mejor, con su competencia,
Por cuidar de mis padres cuando no estuve
Por ser el mejor ejemplo de la palabra HERMANO.*

AGRADECIMIENTO

¡Gracias DIOS! Por permitir que pueda disfrutar de mi familia, gracias por echar una mano en cada decisión, por ser la motivación para cumplir mis metas con excelencia, gracias por siempre apoyarme, agradecida con la vida que me demuestra cada día lo magnifico y justo que puede llegar hacer, y gracias por permitirme vivirla y disfrutarla. El camino nunca fue fácil, pero gracias a su comprensión, a su amor, a su apoyo incondicional, he logrado cumplir un objetivo más en mi vida. Este reconocimiento es de ustedes.

De manera especial, quiero agradecer a:

¡MADRE Y PADRE! ¡JULIA Y JUAN!, Mi mayor ejemplo de perseverancia y dedicación, esfuerzo y trabajo, responsabilidad y honestidad, ... Me demostraron que puedo lograr absolutamente todo lo que me proponga en la vida.

A mi amigo y tutor Mg. Ing. Fernando Arístides Saldaña Milla, agradecerle por la confianza brindada, el apoyo cuando lo necesitaba, la oportunidad de aprender y participar en diferentes actividades que contribuían en mi crecimiento personal.

A la Universidad Católica de Trujillo Benedicto XVI, mi alma máter, por permitirme desarrollarme como profesional, por inculcarme valores éticos y morales, y permitirme compartir gratos momentos con grandes personas y profesionales.

***EIAutor
Walter Scott Bringas Chacón***

DEDICATORIA

A DIOS

*Por brindar salud a mi familia y por
siempre guiarme en el camino del bien.*

A MI ETERNA MADRE

*Por su infinito amor, paciencia y apoyo brindado
en todas las decisiones que tomé y tomaré.*

A MI PADRE

*Por ser un ejemplo real de cómo afrontar
las distintas situaciones difíciles de la vida.*

A MIS HERMANAS Y HERMANO

*Por siempre demostrarme su cariño y comprensión,
y ser apoyo de mis padres cuando yo no estaba cerca.*

A MI PROMETIDA

*Por ser una mujer única y por brindarme su comprensión
y apoyo cuando más lo necesité. Eternamente agradecido.*

AGRADECIMIENTO

Al culminar esta etapa, no puedo dejar de agradecer a mis Padres, Marin y Lidubina por ser un claro ejemplo de cómo afrontar la vida a pesar de las adversidades, a mi familia que siempre me animó a salir adelante, a mis hermanas Yaniré y Aliz por siempre estar con mamá, a mi hermano Eddy por estar con papá, a Dios por sus desafíos y a mi prometida Lizbeth por aparecer en el momento correcto. Sin ustedes no habría sido posible

También quisiera agradecer a:

Cada persona que se cruzó en mi ³⁷ camino, ya que directa o indirectamente, me han ayudado a afrontar este reto de la vida universitaria, y me han ayudado a culminarla con éxito.

A mi tutor Mg. Ing. Fernando Arístides Saldaña Milla, agradecerle por el inmenso apoyo durante mi estadía en la universidad, sus enseñanzas y consejos contribuyeron enormemente en mi desarrollo profesional y personal.

¹ A la Universidad Católica de Trujillo Benedicto XVI, mi alma máter, por permitirme alcanzar mi sueño de ser profesional, por inculcarme valores éticos y morales, y permitirme compartir gratos momentos con mis compañeros.

El autor
Erick Dangelo Quispe Gutiérrez

DECLARATORIA DE AUTENTICIDAD

Nosotros, **Bringas Chacón Walter Scott** con DNI **72049900** y **Quispe Gutiérrez Erick Dangelo** con DNI 74080308, egresados del Programa de Estudios de **Pregrado de Ingeniería Mecánico Eléctrico** de la Universidad Católica de Trujillo Benedicto XVI, damos fe que hemos seguido rigurosamente los procedimientos académicos y administrativos emanados por la Facultad de **Ingeniería y Arquitectura**, para la elaboración y sustentación del informe de tesis titulado: **“INFLUENCIA DE UN CÓDIGO DE PROGRAMACIÓN EN ARDUINO PARA LA DETECCIÓN DE OBJETOS EN BASTÓN PARA INVIDENTES”**, el cual consta de un total de 72 páginas, en las que se incluye 12 tablas y 55 figuras, más un total de 10 páginas en anexos. Dejamos constancia de la originalidad y autenticidad de la mencionada investigación y declaramos bajo juramento en razón a los requerimientos éticos, que el contenido de dicho documento corresponde a nuestra autoría respecto a redacción, organización, metodología y diagramación. Asimismo, garantizamos que los fundamentos teóricos están respaldados por el referencial bibliográfico, asumiendo un mínimo porcentaje de omisión involuntaria respecto al tratamiento de cita de autores, lo cual es de nuestra entera responsabilidad.

Los autores



DNI 72049900
Bringas Chacón Walter Scott



DNI 74080308
Quispe Gutiérrez Erick Dangelo

ÍNDICE

DECLARATORIA DE ORIGINALIDAD	ii
AUTORIDADES UNIVERSITARIAS	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
AGRADECIMIENTO	26
AGRADECIMIENTO	vii
ÍNDICE	ix
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS	xi
RESUMEN	14
ABSTRACT	15
I. INTRODUCCIÓN	16
II. METODOLOGÍA	35
2.1. Enfoque	35
2.2. Diseño de Investigación	35
2.3. Población, muestra y muestreo	36
2.4. Técnica e Instrumentos de recojo de datos	36
2.5. Técnicas de procesamiento y análisis de información	46
2.6. Aspectos éticos en investigación	48
III. RESULTADOS	49
3.1. Programación de sensor HC – SR04	49
3.2. Modulo de detección de objetos	55
3.3. Influencia del código de programación	61
IV. DISCUSIÓN	64
V. CONCLUSIONES	66

VI. RECOMENDACIONES	67
VII. REFERENCIAS BIBLIOGRÁFICAS	68
ANEXOS	72

ÍNDICE DE TABLAS

<i>Tabla 01 Parámetros de los tipos de Arduino</i>	
<i>Tabla 02 Parámetros técnicos del Sensor HC-SR04</i>	
<i>Tabla 03. Parámetros técnicos del motor vibrador.</i>	
<i>Tabla 04. Parámetros técnicos de pantalla LCD 1602</i>	
<i>Tabla 05. Formato de registro de Datos técnicos de Fabricación</i>	
<i>Tabla 06. Formato de registro de Equipamiento</i>	
<i>Tabla 07. Formato de registro de los parámetros medidos.</i>	
<i>Tabla 08. Formato de registro de tareas generales.</i>	
<i>Tabla 09. Formato de registro – Prueba de inspección mecánica</i>	
<i>Tabla 10. Formato de registro – Prueba de inspección eléctrica</i>	
<i>Tabla 11. Pruebas y Resultados obtenidos Llanely B. Martínez, Jairo A. Molina.</i>	
<i>Tabla 12. Pruebas y Resultados obtenidos Walter S. Bringas & Erick D. Quispe.</i>	

ÍNDICE DE FIGURAS

Figura 01. Gráfica de porcentajes de lo que usan las personas invidentes como apoyo. ____	
Figura 02. Layout de Conexión de dispositivos a un bus de comunicación. _____	
Figura 03. Diagrama de flujo de procesos industriales (Flujo y velocidad) _____	
Figura 04. Listado de los Software de Programación más representativos de la Industria. _____	
Figura 05. Logo y descripción del programa AUTODESK Tinkercad _____	
Figura 06. Tipos de Arduino en la programación. _____	
Figura 07. Interpretación de sensores digitales y analógicos. _____	
Figura 08. Imagen de un Sensor HC-SR04 _____	
Figura 09. Funcionamiento del Protocolo de comunicación del Sensor HC-SR04 _____	45
Figura 10. Consideraciones del Sensor HC-SR04 _____	
Figura 11. Motor de vibración Arduino. _____	
Figura 12. Pantalla LCD1602 _____	
Figura 13. Esquema metodológico _____	
Figura 14. Simulación en Arduino de sensor HC-SR04 (Software) _____	
Figura 15. Simulación en Arduino de sensor HC-SR04. (Hardware) _____	
Figura 16 Simulación en Arduino de motor vibrador. (Software) _____	
Figura 17 Simulación en Arduino de motor vibrador. (Hardware) _____	
Figura 18 Simulación en Arduino de pantalla LCD 1602. (Software) _____	
Figura 19 Simulación en Arduino de pantalla LCD 1602. (Hardware) _____	
Figura 20 Simulación en TINKERCAD de sensor HC-SR04 + motor de vibración _____	
Figura 21 Simulación en TINKERCAD de pantalla LCD 1602. _____	
Figura 22. Simulación en el software de Arduino- Primer avance _____	

Figura 23 Simulación en el software de Arduino-Segundo Avance _____	
Figura 24 Simulación en el software de Arduino-Tercer Avance _____	
Figura 25. Simulación en Arduino de Hardware _____	
Figura 26. Conexión virtual de Arduino + Periféricos. _____	
Figura 27. Conexión física de Arduino + Periféricos _____	
Figura 28. Recolección de datos. Periféricos de entrada + Arduino + PC. _____	
Figura 29. Recolección de datos por SERIAL BEGIN. _____	

Figura 30. Recolección de datos por pantalla LCD 1602 _____

Figura 31. Recolección de datos por librería LIQUID CRYSTAL. _____

Figura 32. Recolección de datos – Excel. _____

Figura 33. Esquema de programación. _____

Figura 34. Unit test – Código de sensor ultrasónico HC-SR04 _____

Figura 35. Unit test – Código de motor vibrador. _____

Figura 36. Unit test – código de pantalla LCD1602 _____

Figura 37. Integration Test – Código prefinal _____

Figura 38. EXCEL – HCSR04 vs Puerta _____

Figura 39. EXCEL – HCSR04 vs Mueble _____

Figura 40. EXCEL – HCSR04 vs Cesto de ropa _____

Figura 41. EXCEL – HCSR04 vs Balón _____

Figura 42. EXCEL – HCSR04 vs Vereda _____

Figura 43. EXCEL – HCSR04 vs Poste de alumbrado _____

Figura 44. EXCEL – HCSR04 vs Personas _____

Figura 45. EXCEL – HCSR04 vs Puerta _____

Figura 46. EXCEL – HCSR04 vs Mueble _____

Figura 47. EXCEL – HCSR04 vs Cesto de ropa _____

Figura 48. EXCEL – HCSR04 vs Balón _____

Figura 49. EXCEL – HCSR04 vs Vereda _____

Figura 50. EXCEL – HCSR04 vs Poste de alumbrado _____

Figura 51. EXCEL – HCSR04 vs Personas _____

Figura 52. Margen de error en (cm) – HCSR04 vs Puerta _____

Figura 53. Margen de error en (cm) – HCSR04 vs Vereda _____

Figura 54. Margen de error en (cm) – HCSR04 vs Poste de alumbrado _____

Figura 55. Margen de error en (cm) – HCSR04 vs Personas _____

RESUMEN

La discapacidad visual es un gran problema que afecta a la población mundial, y en específico a nuestros adultos mayores por inercia de la propia vejez; así como también a aquella población que viven en regiones pobres y de escasos recursos. Actualmente, la tecnología ha logrado grandes avances en instrumentos, equipos y/o procesos para poder frenar esta problemática; siendo éstos muy difíciles de acceder para la mayoría. Para ello es que se ha realizado un prototipo de bastón inteligente para que el usuario que proceda a usarlo pueda desplazarse y realizar sus actividades diarias con un poco de normalidad. Bajo ese sentido, la persona invidente debe sentir seguridad al momento de usar el prototipo de bastón, y por eso se evaluó la influencia del código de programación en Arduino presente en el bastón inteligente. Se realizaron varias pruebas al prototipo en conjunto, o por sistemas separados (eléctrico, programación, estructura), y pruebas al sensor de ultrasonido HCSR04. El otro lote de pruebas se hizo en un entorno real para garantizar el funcionamiento del código de programación, y gracias a esto se pudo realizar ajustes en el rango de captación de distancia y modificar el código. Al término de la investigación, se comparó los resultados obtenidos con otras investigaciones y esto permitió respaldar nuestros resultados porque nuestro rango de error en la prueba de detección de los objetos de ensayo es como máximo ± 0.06250 m. Este valor permite garantizar el buen comportamiento del prototipo de bastón creado por los autores.

Palabras Clave: Bastón inteligente, Código de programación, Software Arduino IDE, Sensor ultrasónico, Tinkercad, Seguridad. Rango de detección de distancias.

ABSTRACT

Visual disability is a major problem that affects the world population, and specifically our older adults due to the inertia of old age itself; as well as to that population that lives in poor regions with few resources. Currently, technology has made great advances in instruments, equipment and processes to stop this problem; These being very difficult to access for the majority. For this reason, a prototype of a smart cane has been made so that the user who proceeds to use it can move around and carry out their daily activities with a bit of normality. In this sense, the blind person must feel safe when using the cane prototype, and that is why the influence of the Arduino programming code present in the smart cane was evaluated. Several tests were carried out on the prototype as a whole, or by separate systems (electrical, programming, structure), and tests on the HCSR04 ultrasound sensor. The other batch of tests was done in a real environment to ensure the functioning of the programming code, and thanks to this we were able to make adjustments to the distance capture range and modify the code. At the end of the test, we compared the results obtained with other investigations and this allowed us to support our results because our error range in the detection test of the test objects is a maximum of ± 0.06250 m. This value ensures the good behavior of the cane prototype created by the authors.

Key words: Smart Cane, Programming code, Arduino IDE Software, Ultrasonic Sensor, Tinkercad, Security, Detection range of distance.

I. INTRODUCCIÓN

En un mundo creado para la habilidad de observar, la visión, la más dominante de nuestros sentidos, toma un papel trascendental en nuestra vida diaria y en nuestro desarrollo. El recién nacido depende de la visión para reconocer y vincularse con su madre; el niño pequeño, para dominar el equilibrio y aprender a caminar; el estudiante, para caminar a la escuela, leer y aprender; la mujer joven, para participar en la fuerza laboral; y el adulto mayor, para mantener su independencia. Sin el sentido de la vista, el ser humano perdería aproximadamente el 80% de información que se recibe del entorno; con el cual se puede reconocer colores, objetos, texturas, tamaños, movimiento, etc. (World Health Organization [WHO], 2019, p. V).

A nivel mundial se han estimado aproximadamente ³¹ 2 200 millones de personas que sufren de alguna discapacidad visual y al menos 1 000 millones de estos casos se podrían haber evitado con una correcta atención primaria de salud. Los estudios realizados por la OMS han establecido que las cataratas es de las principales afecciones que son causantes del deterioro en la visión de la persona, pero también, ¹³ los errores de refracción, la degeneración relacionada con la edad, el glaucoma, y la afección principal que deteriora de la visión cercana es la presbicia (Organización Mundial de la Salud [OMS], 2021). Así mismo, es importante agregar que, la mayoría de personas con discapacidad en la vista son mayores de edad (82% mayores de 50 años), y también pobres, pues aproximadamente ²² el 90% de personas con discapacidad visual viven en países tercermundistas; además de que la pérdida de visión se agrupa en zonas rurales y desfavorables, donde el riesgo de perder la visión es entre 10 y 40 veces más que en países desarrollados. (Paul Riordan, E. Cunningham Jr., E., 2012)

La discapacidad visual es un peligro al cual se enfrenta nuestra sociedad. Los menores de edad con daño visual grave tienden a tener problemas a nivel lingüístico, emocional, social y cognitivo. Por otro lado, también afecta a la población adulta causando desempleo, depresión y ansiedad que pueden ser perjudiciales y en los casos ²⁸ de los adultos mayores, los problemas de visión pueden contribuir al aislamiento social, dificultad para caminar con riesgo de caídas y lesiones graves. (OMS, 2021).

Un estudio realizado en Latinoamérica afirma que han existido aproximadamente 3 millones de personas que han quedado en su totalidad sin visión, el 60% de ellas se ha debido a cataratas y problemas de refracción, mientras que el 20% por glaucoma y diabetes. Cabe recalcar, que de todas estas personas que han perdido la visión, al menos un 70% se encuentran en una situación de pobreza o pobreza extrema. (Ministerio de Salud [MINSA], 2018). En el Perú existen aproximadamente 801 mil invidentes y 600 mil sufren de alguna discapacidad visual que comprometen su calidad de vida. De la misma manera, se ha mencionado que existen 60 597 personas con discapacidad en la región norte, donde se registra el 18.1% en La Libertad; además el 13% de las deficiencias generalizadas son problemas de la visión y el 53% de los casos de discapacidad son severos. (Consejo Nacional para la Integración de Persona con Discapacidad [CONADIS], 2020)

El desarrollo de la tecnología y su incremento en el siglo XXI ha permitido optimizar procesos en el ámbito industrial, y facilitar la calidad de vida de las personas con alguna discapacidad específica, se han visto beneficiadas con los avances innovadores que la tecnología ha desarrollado, logrando que personas con alguna enfermedad o discapacidad pueden realizar acciones rutinarias sin ningún tipo de problema. (León Meza, 2021). En ese sentido, para ayudar a las personas invidentes o personas que cuentan con alguna discapacidad visual, ya sea total o parcial; la industria ha fabricados prototipos como bastones (una de las más baratas), visores (una de las más caras) y algún prototipo híbrido de investigación realizada por estudiantes en las universidades de todo el mundo.

Uno de estos prototipos se trata de un calzado háptico, proyecto de un estudiante de Licenciatura en Informática de la Universidad de la Plata. Según explica el autor, las personas que sufren con alguna discapacidad visual se colocan el calzado, el cual tiene comunicación inalámbrica a un smartphone mediante bluetooth, dónde anteriormente se ingresaron coordenadas de inicio y fin. Con esa información y las coordenadas obtenidas del GPS incorporado al teléfono, el código de programación fue desarrollado para calcular en tiempo real, un trayecto desde el origen donde se encuentra el pie al destino anhelado; utilizando la tecnología geocoding y ruteo. (Dupleich & García, 2015)

Los anteojos electrónicos, o también llamados Visores eSight; son un dispositivo que mejora la visibilidad de las personas parcialmente ciegas o de visión reducida. Según su creador, las personas que no han perdido completamente la vista aún conservan cierta capacidad visual concentrada en la visión periférica. eSight incrementa la funcionalidad de

las partes que aún están activas del ojo para compensar la pérdida de actividad en las demás partes. El visualizador, el cual va sujeto alrededor de la cabeza, contiene una cámara pequeña, el cual va captando imágenes en vivo de todo aquello que mire el usuario. Incluye también un panel al lado derecho del visor que funciona como control remoto y se pueden ajustar algunas características como la distancia y el enfoque. Un estudio presentado en la reunión de fin de año en ARVO (The Association for Research in Vision and Ophthalmology) en la cual se trató los beneficios del uso del visor eSight en un grupo de siete hombres y seis mujeres. Los resultados expuestos mostraron una mejora en la captación de medidas de distancia y agudeza visual cercana (DVA, NVA), velocidad de lectura (RS) y sensibilidad al contraste (CS) en un tiempo de uso de 3 meses continuo del dispositivo. El precio del visor eSight oscila entre \$5,950.00 dólares, lo cual significa que el acceso a esta tecnología no está al alcance de las personas vulnerables que más sufren esta discapacidad, cómo se mencionó anteriormente.

Una de las opciones más económicas con la cual la tecnología ha apoyado a los invidentes, es un bastón blanco o bastón inteligente, aspecto muy importante en nuestra investigación. Hay un sinnúmero de prototipos, pero el más comercial es un bastón llamado WeWalk fabricado por una empresa del mismo nombre. Este dispositivo funciona gracias a un sistema de sensores de ultrasonido que detectan obstáculos hasta por encima de la altura del pecho, alertando mediante una vibración al usuario final. Sumando la conectividad a la navegación GPS del dispositivo y otras características más, su costo llega a ser relativamente menor al eSight, con un valor aproximado de \$599.00 dólares; siendo un monto relativamente difícil de alcanzar para el grupo de personas que más sufren esta discapacidad.

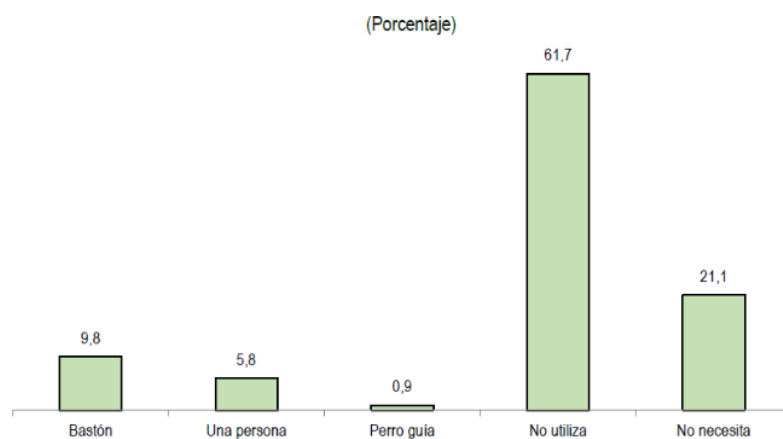
Es inevitable afirmar que la ceguera ocupa un puesto importante al hablar de discapacidades humanas a nivel mundial, y este se puede transformar en un riesgo mucho mayor, debido a los diferentes fenómenos epidemiológicos; como el incremento de expectativa de vida, características sociodemográficas, la desigualdad socioeconómica que hay entre países, etc. Lo que da como resultado una problemática emergente que necesita ser abordada con altísima prioridad por parte de nuestras autoridades de salud.

En el Perú, según ⁴⁹ el (Instituto de Estadística e Informática [INEI],2018) el 9.8% de las personas con discapacidad visual usan el bastón blanco como guía al momento de

movilizarse, mientras que el 61.7% no utiliza un bastón blanco o utilizan un bastón sencillo sin los avances tecnológicos como se puede apreciar en la siguiente figura.

Figura 01

Personas con discapacidad visual, aun usando anteojos, según apoyo.



Nota: Tomado del Instituto Nacional de Estadística e Informática.

Nota: En la figura se puede apreciar que el bastón blanco presenta un 9.8% como elemento de movilización para los invidentes.

¹⁵ Según Dos Santos, Medola, Cinelli, García y Sandnes (2020), el bastón blanco es la herramienta de apoyo más utilizada, ya que permite ubicar elementos peligrosos y asimismo indicar a las personas de alrededor que el usuario es invidente, para que reciban apoyo en las situaciones que resulten complicadas; como subir a un vehículo, subir escaleras, cruzar calles transitadas, entre otras. De la misma manera, según investigadores de (Massachusetts Eye and Ear, 2017) el cerebro de las personas invidentes desarrollan la habilidad de percepción gracias a lograr distinguir semejanzas o diferencias con los sentidos del cuerpo humano, ya que establecen nuevas conexiones por ausencia de información visual, pero el grado de autonomía funcional disminuye al salir a las calles, ya que se encuentran en constante peligro, porque no controlan lo que sucede a su alrededor; por tal motivo toma importancia el uso del bastón blanco como herramienta de apoyo.

Queda demostrado que el bastón blanco es las herramientas más usado cuando se habla de movilidad por los invidentes, no obstante, el bastón blanco presenta ciertas

desventajas y falencias a la hora de su utilización, ya que, no reconoce obstáculos hasta no antes de haber un contacto, tampoco las dimensiones de los objetos o desniveles de los mismos, por eso, junto con el uso de la tecnología se busca mejorar la herramienta de apoyo más usado por aquellas personas con esta discapacidad.

Alrededor del mundo, se han aplicado diversas formas de estudio de diseño de un bastón blanco inteligente usando múltiples componentes que mejoran el desempeño de este. En Latinoamérica ha habido grandes investigaciones para ayudar a las personas invidentes, en Paraguay, Lezcano y Cabral (2016) realizaron un artículo titulado “Prototipo de bastón inteligente para invidentes en lenguaje Arduino”, donde establecieron el diseño de un prototipo de bastón electrónico utilizando el sensor ultrasónico HY - SRF05 además de un sistema de teledetección y módulos de comunicación Ublox NEO6m & GPRS shield SIM900.

En Ecuador, Paucar Espinoza (2018) realizó una investigación titulada “Gafas y bastón inteligente para una persona invidente”, en el cual desarrolló un prototipo de bastón inteligente con el agregado de unas gafas el cual le sirvió para indicar la distancia a la que se encuentra un obstáculo para que el portador logre evitarlo. De igual manera, en México, López, Navarro y Silva (2018) realizaron un artículo titulado “**Detección de objetos por medio de escaneo ultrasónico**” donde detallaron el funcionamiento de los sensores ultrasónicos y ventajas frente a otros dispositivos actualmente, en el campo de reconocimiento de patrones, por medio del Arduino Nano y una interfaz desarrollada en LabView.

En Colombia, Paiba Diaz (2020) realizó una tesis titulada “**Implementar un bastón inteligente en Arduino para facilitar la identificación de objetos circundantes y productos de un almacén para la población invidente**”, aquí el autor mencionó que existen pocos estudios sobre la detección de objetos y que hay muy pocas pruebas en los prototipos de los bastones inteligentes desarrollados. Es decir, el avance tecnológico en esta discapacidad ha sido muy baja, por eso es importante seguir en la línea de investigación. Es por eso por lo que, se han elaborado investigaciones como la de Marcillo y Collantes (2020) quienes en su investigación titulada “Diseño de un prototipo de un bastón inteligente con sensores para personas con discapacidad visual y física en la carrera de Ingeniería en computación y redes”, mostraron un prototipo de bastón electrónico capaz de encontrar obstáculos a una distancia superior a 1 metro.

A su vez, se han trabajado investigaciones de complemento para el bastón que puedan ayudar mucho más en el traslado de las personas invidentes. Tal es el caso de Bravo Dussan (2020) quien desarrolló una tesis titulada “Prototipo de brazaletes como complemento al bastón para personas con discapacidad visual”³³ en la cual estableció mejoras en un prototipo de bastón agregando un brazaletes que permite aumentar la frecuencia de vibración al detectar los obstáculos a través de los sensores ultrasónicos. También se han diseñado usos específicos en los bastones, tal es el caso de Martínez y Molina (2021) quienes en la tesis titulada “Construcción de bastón inteligente para personas con discapacidad visual”¹⁴ establecieron la forma no sólo de la detección de obstáculos sino también la forma de poder detectar desniveles; y León Mexa (2021) desarrolló una investigación titulada “Diseño y evaluación de prototipo de bastón con sensores electrónicos para personas no videntes de la unidad de rehabilitación del hospital IESS Quito Sur” en la cual desarrolló un prototipo de bastón inteligente para ayudar en terapia a personas invidentes de un hospital con el fin de ayudarlos en su movilización.

En Bolivia, Huanca Coaquira (2022), realizó un diseño de un prototipo de bastón inteligente usando un código de programación en Arduino en la cual estableció un rango de captación máximo de 230 cm para cada sensor ultrasónico (utilizaron dos sensores en su bastón, en la parte inferior y superior para un mayor alcance de objetos), ya que a una mayor distancia el sensor no registraba el regreso de la señal con exactitud y por ende el código no podía procesar la distancia a la que se encontraba el objeto. Ella concluyó que, trabajando a menor distancia de captación es suficiente para que el usuario pueda advertir la cercanía del objeto y poder evitarlo.

Martínez Fuentes (2014) , realizó una investigación enfocada a la plataforma Arduino Uno R3 y al sensor ultrasónico HC - SR04, en donde estableció que sin buenos componentes electrónicos para el hardware del sistema, y el algoritmos para software (conjunto lógico de instrucciones o el código de programación) , no se procesaría la señal eléctrica que otorga el sensor HC-SR04 al circuito interno de procesamiento y control, de aquí se pudo deducir que; es importante la optimización del rango de captación de distancia del sensor, ya que sin una buena lectura no se podrá procesar la información y por ende el prototipo no podría entregar un valor correcto al usuario invidente portador del bastón.

Por toda la información mencionada que respalda la investigación, se puede observar que, al tratarse de prototipos o bastones inteligentes, el método más sencillo,

económico y rápido, es la fabricación de un prototipo de bastón inteligente integrado por un código de programación ensamblado en una placa de Arduino. Y en base a toda la información recopilada y estudiada es que se planteó el siguiente problema general a través de la pregunta: ¿Cómo influye un código de programación en Arduino en la detección de objetos de un Bastón Blanco para invidentes?; así mismo, se plantearon los siguientes problemas específicos: ¿Cuáles son las características del módulo ultrasónico? ¿Cómo se realizará la implementación de la programación en el módulo de detección de objetos de un Bastón Blanco de invidentes?

25 Ivana Harari [Coordinadora del proyecto, (2019)] y un equipo, formado por investigadores y alumnos de la Facultad de Informática de la Universidad Nacional de la Plata (UNLP), desarrolló un kit tecnológico 30 para personas con discapacidad visual. El prototipo está diseñado con un bastón, anteojos inteligentes, sensores infrarrojos de proximidad, que permite identificar objetos de la cintura hacia arriba por consecuencia de las gafas y de la cintura hacia abajo por la ubicación de los sensores en el bastón. El proyecto busca soluciones simples y de bajo costo, por lo que, se desarrolla en la filosofía de software libre, dejando el código abierto y a servicio de la comunidad científica.

A nivel nacional también se han realizado investigaciones innovadoras, Lizárraga González (2018) en su investigación titulada “Propuesta para el diseño de un bastón electrónico para personas invidentes que mejorará la calidad de su desplazamiento diario” pudo desarrollar la elaboración de un bastón inteligente con productos de fácil alcance y económicos. También, Vela Inoñan (2019) 11 en su tesis titulada “Diseño e implementación de un bastón ergonómico con sistema de posicionamiento global para mejorar el desplazamiento de personas invidentes en el centro La Unión Nacional de Ciegos en el Perú”, expuso que las pruebas realizadas del bastón ergonómico construido por sensores ultrasónicos, microcontroladores, etc., sí logra incrementar el desplazamiento de las personas. Y finalmente Aragón Loza (2020) en su trabajo titulado “Revisión de la literatura de modelos matemáticos para el tránsito de personas con discapacidad visual a través de rutas accesibles y seguras” estableció no solo un modelo de detección de obstáculos, sino también, agrega un modelo de navegación y accesibilidad a través de rutas seguras y asequibles para las personas invidentes.

Las investigaciones de estudiantes que buscan realizar un prototipo de bastón inteligente económico para contribuir con las personas que no cuentan con los recursos

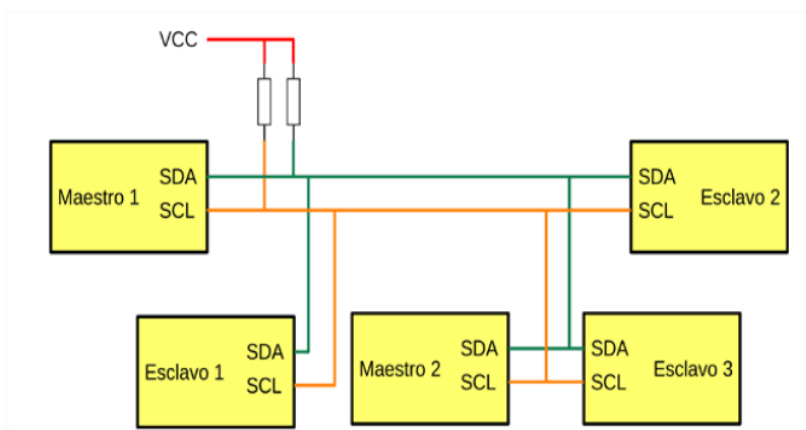
necesarios y que sufren con alguna discapacidad visual; traen consigo una herramienta en común, y es que está claro que en la mayoría de los diseños de bastones inteligentes se utilizan el lenguaje Arduino, ya que se trata de un código abierto de fácil acceso con periféricos de entrada y salida con costos bajos.

I2C – Esquema de comunicación y elementos

El I2C es un protocolo de comunicación que permite la transferencia de bits entre dos o más dispositivos digitales y solamente con dos cables de comunicación, SDA (Intercambio de datos) y SCL (Señal de reloj). Existe dos elementos básicos, un master; que se encarga de iniciar y terminar la comunicación, además de enviar bits de dirección y de memoria, generar bit de lectura, transmitir bits de datos y confirmar la recepción de datos. Y un *slaver*, que es el elemento que suministra la información de interés, es decir el que envía la información en paquetes de bits.

Figura 02

Lay out de Conexión de dispositivos a un bus de comunicación.



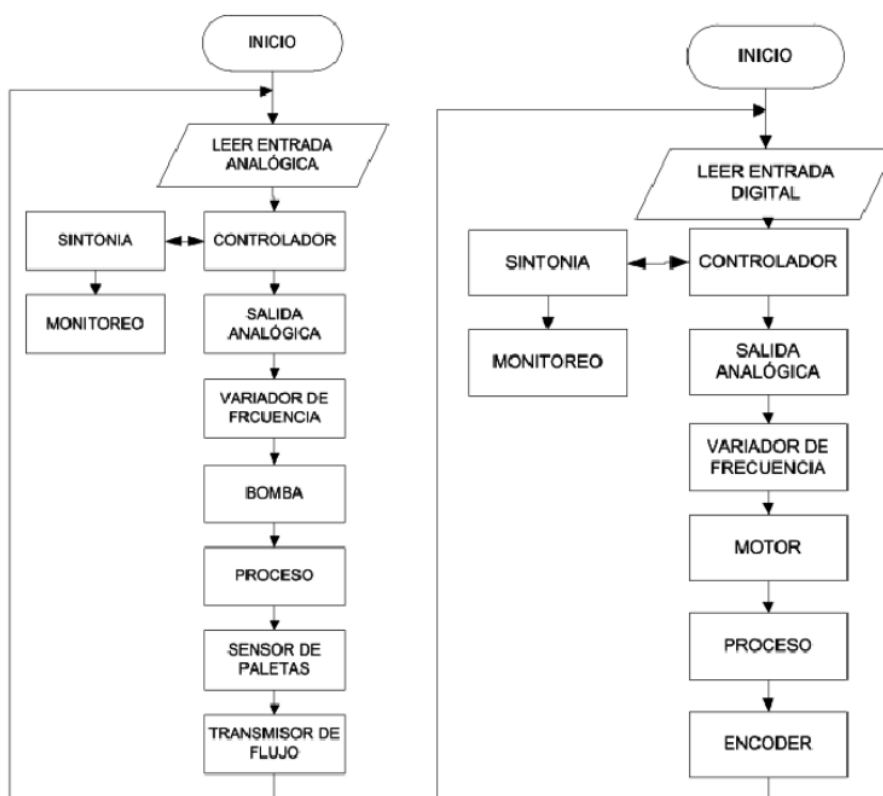
Nota: Tomado de la página de programación <https://hetpro-store.com/TUTORIALES/i2c/>

Lógica de control

Es importante tener en cuenta algunos términos que nos ayudarán a comprender el objeto de estudio, como elemento importante se tiene la lógica de control; se trata de la agrupación de algoritmos y estructuras de control que determina la de ejecución de un programa. A su vez, un algoritmo está definido como los pasos sistemáticos para resolver un problema, los cuales pueden ser graficados mediante un flujograma.

Figura 03

Diagrama de flujo de procesos industriales (Flujo y velocidad)



Nota: Tomada de RESEARCHGATE, como ejemplo de diagrama de flujo en la física.

Software de programación

Un software como tal es la parte lógica de una computadora, es el encargado de decirle que hacer al hardware y se puede diferenciar tres tipos de software: software de sistema (permite administrar y gestionar recursos relacionados a la memoria), software de aplicación (programas que se crean con el software de programación) y software de programación (Conjunto de utilidades y herramientas que permiten desarrollar programas informáticos).

Figura 04

Software de Programación más representativos de la Industria.



Nota: Tomado de <https://tallerdehardwareutd.wordpress.com/software-de-programacion/>

IDE

El entorno de desarrollo integrado permite cubrir el desarrollo de un software, te facilita aplicaciones, herramientas, modificar, crear, compilar, depurar y todo lo necesario; generalmente en un mismo lenguaje para sacar la mayor productividad que un desarrollador necesita.

Los IDE pueden estar enfocados a un mismo lenguaje para evitar paradigmas en la programación, como, por ejemplo; Ensembler, JavaScript, Python, C#, C + +. Estos últimos son los lenguajes en los que está basado el Arduino. Sin embargo, muchos IDE son

capaces de interpretar muchos lenguajes de programación, como; Eclipse, .NET, NetBeans, Microsoft Visual Studio, entre otros.

Definido todo lo anterior, para poder simular un código de programación es necesario una plataforma de simulación, con el protocolo de comunicación correcto.

Tinkercad

Es un programa de Autodesk gratuito de modelado 3D que se puede ejecutar desde un navegador web. Pero no solo es una plataforma para crear modelos de impresiones 3D a través de la geometría sólida construida, sino que, a partir del 2017, Autodesk fusiono a *Tinkercad con 123D Circuits* (Circuits.io), que básicamente es un laboratorio de circuitos electrónico basado en navegadores que admiten microcontroladores Arduino uno, placas *Micro:bit o chips ATtiny*. De la misma manera la funcionalidad de crear un código de programación utilizando *CodeBlocks o* simplemente crear un código basado en texto.

A pesar de ser una herramienta sencilla y no tan versátil con forme a su biblioteca de programación y electrónica, Tinkercad ofrece funciones avanzadas o simulaciones de placas múltiples para usuarios experimentados.

Figura 05

Logo y descripción del programa AUTODESK TINKERCAD



Nota: Tomado de <https://www.tinkercad.com/dashboard>

Por tal motivo se eligió Arduino UNO, ya que es una placa de *hardware* y *software* abierto, flexible y fácil de usar más popular del mundo *maker*. Ya que el hardware libre permite al programador crear sus propias placas y el *software* libre es el código accesible al público en general para el que quiera usarlo y modificarlo.

Arduino

²² Arduino es una placa basada en un microcontrolador ATMEL y lenguaje de comunicación I2C. Esta posee una interfaz de entrada que permite conectar diferentes tipos de periféricos e interpretar la información de los mismos, por ejemplo: sensores de luz, de ultrasonidos, cámaras, teclados, entre otros. También tiene ¹³ una interfaz de salida que se encarga de trasladar la información procesada a otros periféricos, como pantallas, piezos, luces, etc.

Figura 06

Tipos de Arduino en la programación



Nota: Tomada de Ingeniería Mecafenix

<https://www.ingmecafenix.com/electronica/programacion/arduino/>

Tabla 01

Comparación de parámetros de los tipos de Arduino

Parámetro	Arduino Uno	Arduino Mega 2560	Arduino YUN	Arduino Due	Arduino Gemma
Microcontrolador	ATmega328P	ATmega2560	ATmega32U4	AT91SAM3X8E	ATtiny85
Voltaje (VDC)	5	5	5	3.3	3.3
³² Digital (I/O) pins	14 (4 for PWM Output)	54 (15 for PWM Output)	20	-	3
Pins entrada analógica	6	6	-	12	1
Corriente DC por I/O (mA)	20	20	40	130	20
⁵ Memoria Flash	32 KB (5 KB for bootloader)	256 KB (8 KB for bootloader)	⁵ 5 KB (4 KB for bootloader)	512 KB for use applications	⁵ 8 KB (2.75 KB for bootloader)
SRAM	2 KB	8 KB	2.5 KB	96 KB (64 KB and 32 KB)	512 KB

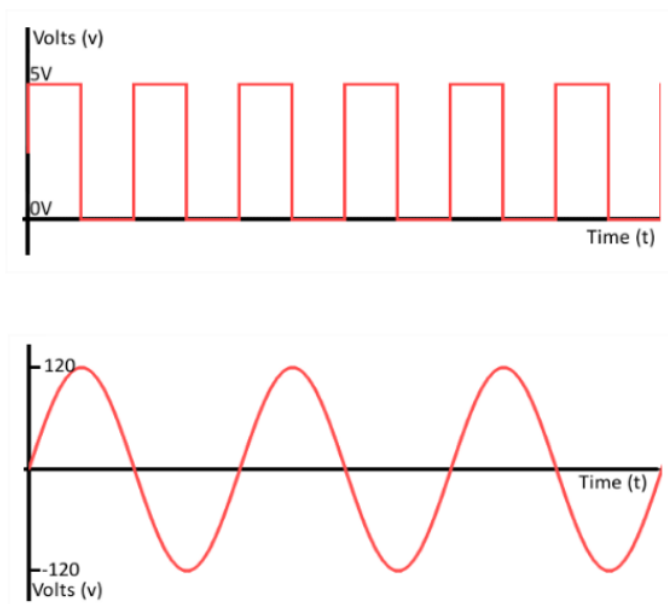
Nota: De acuerdo a los parámetros de construcción de algunos Arduino y la complejidad del proyecto se pueden usar varios, algunos sobredimensionados, pero siguen cumpliendo con los requisitos, sin embargo, se está evaluando cómo influye un código de programación, por lo que será evaluado en aplicaciones web o con el mínimo gasto propio.

Como se observó anteriormente el Arduino posee un interfaz de entrada donde se conectan periféricos, algunos de ellos son los sensores. Estos son los dispositivos que capturan magnitudes físicas o químicas y los transforman en variables eléctricas. Por ejemplo, las magnitudes físicas como: temperatura, distancia, presión, fuerza, radiación, entre otros, y los transforma en magnitudes eléctricas como, tensión, corriente eléctrica, resistencia. etc.

Existen los tipos de sensores digitales (estas son salidas en dos estados; como si fuera un binario, 1 o 0) y los sensores analógicos (emiten las señales en proporciones a lo que están midiendo).

Figura 07

Interpretación de sensores digitales y analógicos.



Nota: Tomado de ²⁹ <https://solectroshop.com/es/blog/que-son-los-sensores-analogicos-todo-sobre-su-funcionamiento-n91>

Un **sensor** que funciona perfectamente en el estudio, son los de ultrasonido, estos se encargan de medir distancias mediante el tiempo de transcurso por eco, es decir, el tiempo que demora entre el haz de ultrasonido del emisor y la recepción del mismo, además, son muy simples, tienen mucha precisión y son relativamente baratos. Se usan para evitar el contacto con objetos, medir los mismos, señalar aproximaciones o que tan alejados están los objetos, entre otros ejemplos.

¹⁸ **Sensor ultrasónico HC - SR04**

Es un circuito electrónico que detecta o mide la distancia de objetos en un rango de 2 cm a 400 cm. Es compatible con Arduino debido a que su alimentación es de 5 VDC y 1.5 mA.

Figura 08

Imagen del Sensor HC SR04



Nota: Tomado de [Medidor de distancias por Ultrasonidos \(compic.es\)](http://Medidor de distancias por Ultrasonidos (compic.es))

Tabla 02

Parámetros técnicos del Sensor HC SR04

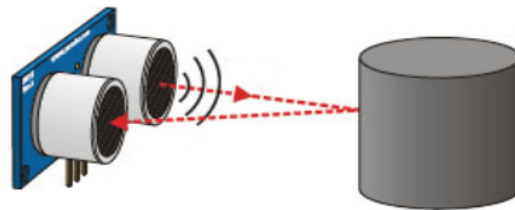
Parámetros técnicos	
Tensión de alimentación	5VDC
Corriente de alimentación	1.5 mA
Ángulo de apertura	15°
Señal de disparo	10 us
Rango de medición	2 cm - 400 cm
Dimensión	45x20x15 mm
Interfaz	
Contacto de potencia	-5V DC
Trig	Señal de entrada

Echo	Señal de salida
GND	Tierra

El sensor Trig emite una ráfaga de ultrasonido (40 KHz), que viaja por el aire y detecta los objetos, estos rebotan y son recibidos por el sensor Echo.

Figura 09

Funcionamiento del Protocolo de ¹³ comunicación del Sensor HC SR04



Nota: El tiempo ⁹ en que detecta el sensor Echo el retorno del ultrasonido es el equivalente a la distancia entre el sensor Trig y cualquier objeto que se encuentre delante. Sin embargo, se debe tener las siguientes consideraciones:

- Que el objeto no se encuentre fuera de su rango.
- El ángulo de medición no puede ser menor a 45°
- El objeto no puede reflejar el ultrasonido.

Figura 10

Consideraciones del Sensor HC SR04



Nota: Es importante que el objeto a detectar se encuentre entre un rango de 0-45° para un mejor desempeño.

El presente estudio hace énfasis en la distancia máxima de medición de un sensor de ultrasonido, la cual será medida en varias pruebas para generar una ecuación matemática que nos permita contrastar posteriormente con los cálculos obtenidos con la siguiente fórmula.

Ecuación 01

Calcular distancia del Sensor HC SR04

$$D (m) = \frac{(T (s) * C \left(\frac{m}{s}\right))}{2}$$

7
Nota: Donde D es la distancia, T es el tiempo entre la emisión y recepción, y C es la velocidad del sonido. (El valor se multiplica por 0.5 ya que T es el tiempo de recorrido de ida y vuelta).

Motor Vibrador

Como señal de salida se tiene un motor de vibración, este es un componente electromecánico que al momento de recibir cargas entre 3VDC - 5VDC, emitirá un efecto de vibración. Entre más se aproxime a su voltaje máximo, este vibrará más fuerte, de lo contrario más leve.

Figura 11

Motor de vibración Arduino.



Nota: Tomado de [Sensor Piezoeléctrico de Vibración para Arduino UNO \(solectroshop.com\)](http://solectroshop.com)

Tabla 03
Parámetros técnicos del motor vibrador

Parámetros técnicos	
Voltaje de alimentación	3VDC - 5VDC
Corriente de consumo	74mA - 130mA
Dimensiones	8mm*3.2mm
Peso	1g
Material	Metálico

Nota: Tomado de Datasheet adjunto en anexos.

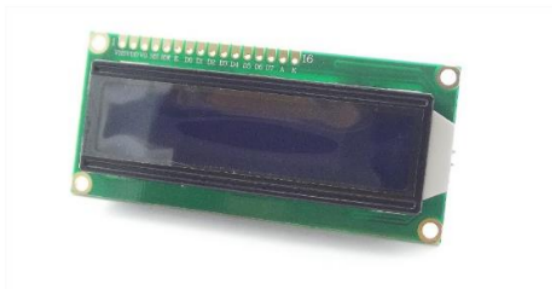
LCD 1602

Para visualizar los datos obtenidos por el periférico ultrasónico HC SR04 se puede hacer de diferentes maneras, la más sencilla es llamando mediante código al monitor serial, cuya pantalla es propia del software de Arduino Uno. Por otro lado, si las pruebas son en campo, se complica un poco el uso del mismo, ya que, se tendría que tener un ordenador cerca todo el tiempo o simplemente agregarle un *shield datalogger* al Arduino para guardar

datos mediante una micro-SD. Sin embargo, se le puede agregar físicamente una pantalla LCD alfanumérico para visualizar o monitorear los parámetros, es la solución más práctica, de menor consumo y de características alfanuméricas.

Figura 12

Pantalla LCD1602.



Nota: Tomado de Datasheet adjunto en anexos.

Tabla 04

Parámetros técnicos de pantalla LCD 1602

Parámetros técnicos	
VSS	GND
VDD (V)	5
VEE	Potenciómetro
RS	D8
RW	GND
EN	D9

Nota: Tomado de Naylamp Mechatronic.

Existen algunas investigaciones y prototipos de bastones inteligentes, sin embargo, no se observa que las personas invidentes los usen, sino todo lo contrario, se sigue

usando el bastón tradicional. Esto puede ser a consecuencia de lo costoso del producto y la tecnología que pueda tener, no obstante, se puede replicar con componentes electrónicos relativamente baratos, por eso se planteó como objetivo general evaluar la influencia de un código de programación en Arduino para la detección de objetos de un bastón blanco de invidentes; el objetivo general se dividió en dos objetivos específicos y fueron: Programar el sensor ultrasónico en Arduino y lograr la implementación de la programación en módulo de detección de objetos de un Bastón blanco de invidentes. Teniendo como resultado a través de la comprobación de la hipótesis general que realmente si influye el código de programación para la detección de objetos de un Bastón Blanco de invidentes.

43

II. METODOLOGÍA

2.1. Enfoque

La presente investigación tiene un enfoque cuantitativo, porque se realizó una medición contrastada en datos numéricos los cuáles fueron analizados para poder realizar la formulación de la hipótesis mediante el diseño de un código de programación.

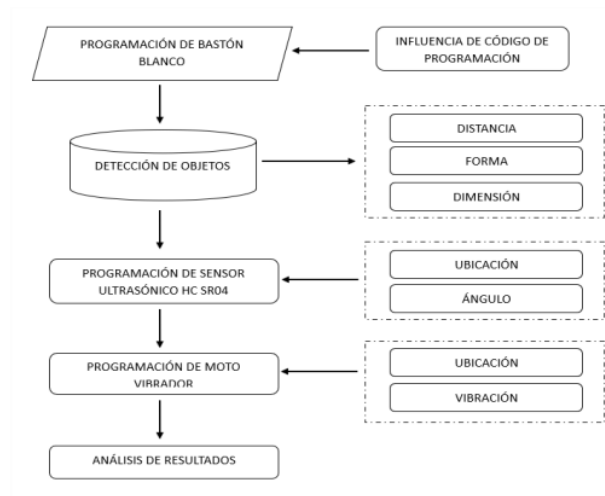
1

2.2. Diseño de Investigación

El diseño metodológico de la investigación es pre experimental, ya que se manipuló la variable independiente, que es el código de programación para evaluar la detección de objetos que viene a ser la variable dependiente. Además, porque el diseño del prototipo es un primer acercamiento al problema de investigación.

Figura 13

Esquema metodológico.



Nota: Esquema realizado por los autores.

2.3. Población, muestra y muestreo

Población: Todos los códigos de programación en Arduino utilizados en detección de objetos para un bastón blanco.

Muestra: El código de programación de Arduino.

Muestreo: El código de programación de Arduino.

2.4. Técnica e Instrumentos de recojo de datos

Se empleó el método *Unit Tests* y *Integration Tests*, como marco de filosofía DevOps, con la finalidad de inspección y funcionamiento a cada componente.

Unit Tests: Son pruebas unitarias o que son aplicadas en una parte de la aplicación completa. Tiene como finalidad comprobar que el componente o sistema no tenga defectos que puedan ser problema para el todo.

Figura 14

Simulación en ARDUINO de sensor HC-SR04. (Software)


```
Unit_Test_SENSOR_SR04
Archivo Editor Programa Herramientas Ayuda

Unit_Test_SENSOR_SR04

int TRIG = 10;
int ECHO = 9;
int Duracion, Distancia;

void setup()
{
  pinMode(TRIG, OUTPUT); // trig como salida
  pinMode(ECHO, INPUT); // echo como entrada
  Serial.begin(9600); // inicializacion de comunicacion serial a 9600 bps
}

void loop()
{
  digitalWrite(TRIG, HIGH); // generacion del pulso a enviar
  delay(10);
  digitalWrite(TRIG, LOW);
  Duracion = pulseIn(ECHO, HIGH);

  Distancia = Duracion / 58.2;
  Serial.print("Distancia (cm) ");
  Serial.println(Distancia); // envio de valor de distancia por monitor serial
  delay(500);
}

El Sketch usa 3158 bytes (98) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 206 bytes (104) de la memoria dinámica, dejando 1542 bytes para las variables locales. El máximo es 2048 bytes.
Automa usa en 20160
```

Figura 15
Simulación en ARDUINO de sensor HC-SR04. (Hardware)

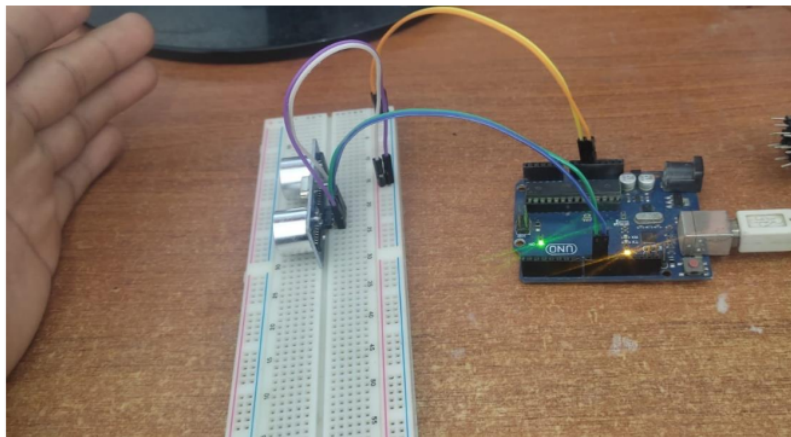


Figura 16
Simulación en ARDUINO de motor vibrador. (Software)

```
Unit_Test_Motor_Vibraci_n Arduino 1.8.19
Archivo Editor Programa Herramientas Ayuda

Unit_Test_Motor_Vibraci_n

void setup()
{
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
}

void loop()
{
  digitalWrite(8, HIGH);
  delay(100);
  digitalWrite(8, LOW);
  delay(100);
  digitalWrite(9, HIGH);
  delay(1000);
  digitalWrite(9, LOW);
  delay(100);
}

I

Salida
El Sketch usa 1004 bytes (3%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 9 bytes (0%) de la memoria dinámica, dejando 2039 bytes para las variables locales. El máximo es 2048 bytes.
Además usa en 0140
```

Figura 17
Simulación en ARDUINO de motor vibrador. (Hardware)

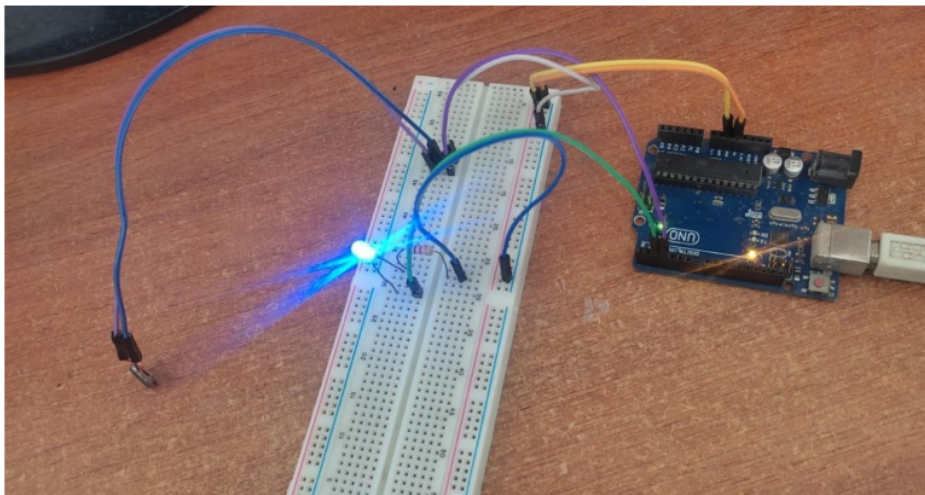


Figura 18
Simulación en ARDUINO de pantalla LCD 1602. (Software)

```
Unit_Tests_LCD_1602 Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda
Unit_Tests_LCD_1602

#include <Wire.h> // librería de comunicación por I2C
#include <LCD.h> // librería para funciones de LCD
#include <LiquidCrystal_I2C.h> // librería para LCD por I2C

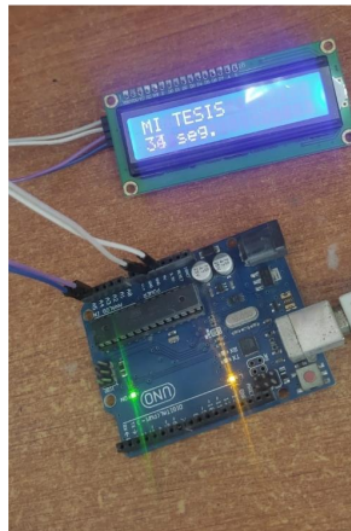
LiquidCrystal_I2C lcd (0x27, 2, 1, 0, 4, 5, 6, 7);

void setup()
{
  lcd.setBacklightPin(3, POSITIVE); // puerto F3 de PCF8574 como positivo
  lcd.setBacklight(HIGH); // habilita iluminación posterior de LCD
  lcd.begin(16, 2); // 16 columnas por 2 líneas para LCD 1602A
  lcd.clear(); // limpia pantalla
}

void loop()
{
  lcd.setCursor(0, 0); // ubica cursor en columna 0 y línea 0
  lcd.print("MI TESIS"); // escribe el texto
  lcd.setCursor(0, 1); // ubica cursor en columna 0 y línea 1
  lcd.print(millis() / 1000); // función millis() / 1000 para segundos transcurridos
  lcd.print(" seg."); // escribe seg.
}

El Sketch usa 4038 bytes (12%) del espacio de almacenamiento de programa. El máximo es 32768 bytes.
Las variables Globales usan 264 bytes (14%) de la memoria dinámica, dejando 1754 bytes para las variables locales. El máximo es 2048 bytes.
```

Figura 19
Simulación en ARDUINO de pantalla LCD 1602. (Hardware)



Integration Tests: Son el segundo paquete de pruebas, es decir, como funcionan juntando las partes unitarias. Tiene como finalidad encontrar defectos que no son tan obvios en varias partes de la aplicación.

Figura 20
Simulación en TINKERCAD de sensor HC-SR04 + motor de vibración.

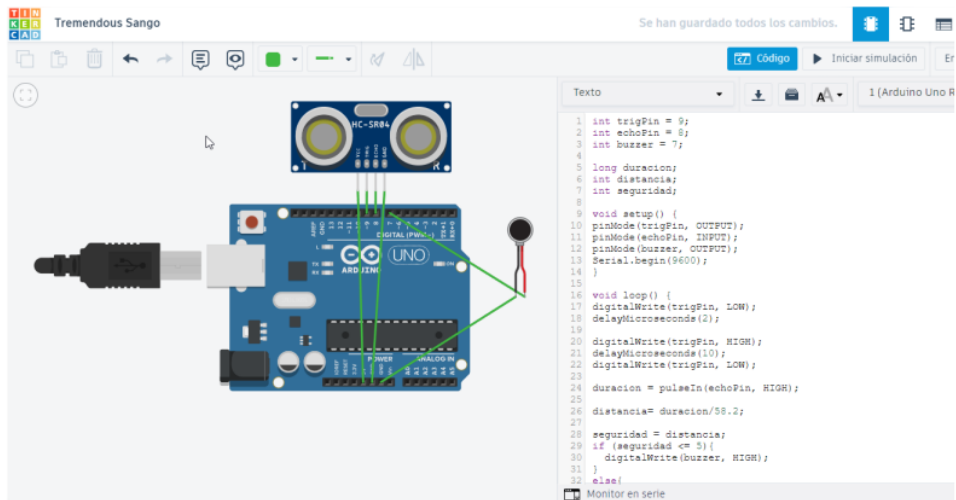


Figura 21
Simulación en TINKERCAD de pantalla LCD1602.

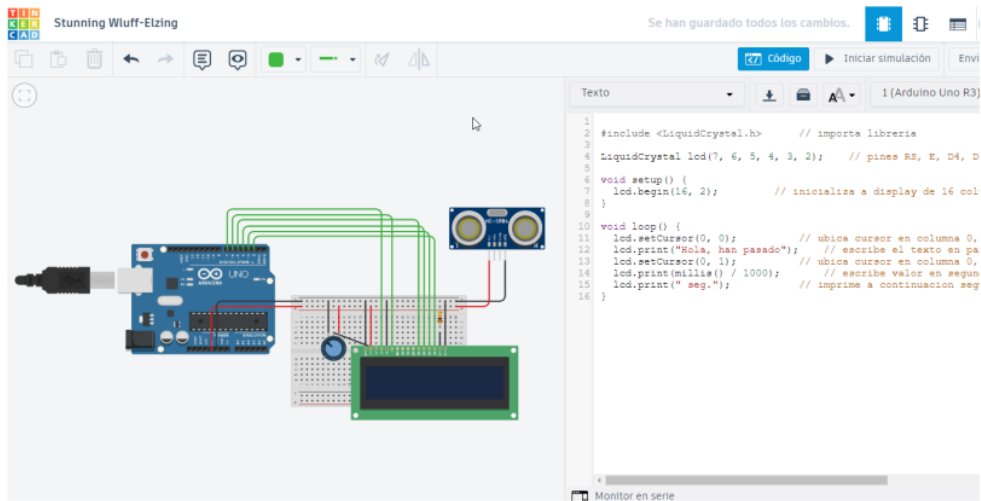


Figura 22
Simulación en el software de ARDUINO-Primer Avance.

```
AVANCE_1
const int trig = 8, echo = 9;
long duracion;
int distancia;

void setup()
{
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);

  Serial.begin(9600);
}

void loop()
{
  digitalWrite(trig, HIGH);
  delay(10);
  digitalWrite(trig, LOW);

  duracion = pulseIn(echo, HIGH);
  distancia = (duracion * 0.034) / 2;
  Serial.println(distancia);
  delay(200);
}

Compilado
El sketch usa 3152 bytes (9%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 105 bytes (9%) de la memoria dinámica, dejando 1660 bytes para las variables locales. El máximo es 2048 bytes.
Arduino Uno en COM3
```

Figura 23
Simulación en el software de ARDUINO-Segundo Avance.

```
AVANCE_3
if (distancia <= 40 || distancia >= 30) {
  delay(500);

  digitalWrite(led, HIGH);
  delay(distancia * 1);
  digitalWrite(led, LOW);
}
if (distancia <= 30 || distancia >= 20) {
  delay(500);

  digitalWrite(led, HIGH);
  delay(distancia * 5);
  digitalWrite(led, LOW);
}
if (distancia <= 20 || distancia >= 10) {
  delay(500);

  digitalWrite(led, HIGH);
  delay(distancia * 10);
  digitalWrite(led, LOW);
}
if (distancia <= 10 || distancia >= 1) {
  delay(500);

  digitalWrite(led, HIGH);
  delay(distancia * 20);
  digitalWrite(led, LOW);
}
}

Compilado
El sketch usa 3414 bytes (10%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 130 bytes (9%) de la memoria dinámica, dejando 1700 bytes para las variables locales. El máximo es 2048 bytes.
Arduino Uno en COM3
```

Figura 24
Simulación en el software de ARDUINO-Tercer Avance.

```
sketch_007a_Arduino 1.8.10
Archivo Editar Programa Herramientas Ayuda
sketch_007a
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

const int Trig = 9, Echo = 10, motor=11;
long duracion;
int distancia;
int ultimoDistancia = 0;
LiquidCrystal_I2C lcd(0x27,16,2);
void setup()
{
  pinMode(Trig, OUTPUT);
  pinMode(Echo, INPUT);
  pinMode(motor, OUTPUT);
  Serial.begin(9600);
  lcd.begin();
  lcd.backlight();
}
void loop()
{
  digitalWrite(Trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(Trig, LOW);
  duracion = pulseIn(Echo, HIGH);
  distancia = (duracion * 0.034) / 2;
  Serial.println(distancia);
  if(distancia != ultimoDistancia)
  {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Distancia: ");
  }
}

// Sketch usa 5990 bytes (11%) del espacio de almacenamiento de programa. El máximo es 32768 bytes.
// Las variables globales usan 302 bytes (24%) de la memoria dinámica, dejando 1448 bytes para las variables locales. El máximo es 2048 bytes.
// Arduino Uno en C/C++

sketch_007a_Arduino 1.8.10
Archivo Editar Programa Herramientas Ayuda
sketch_007a
digitalWrite(motor, HIGH);
delay(distancia* 2);
digitalWrite(motor, LOW);

if (distancia <= 100 && distancia >= 70){
  lcd.setCursor(0,1);
  lcd.print(" LEJOS");
  delay(500);
}

digitalWrite(motor, HIGH);
delay(distancia* 8);
digitalWrite(motor, LOW);
}

if (distancia <= 70 && distancia >= 30){
  lcd.setCursor(0,1);
  lcd.print(" CERCA");
  delay(500);
}

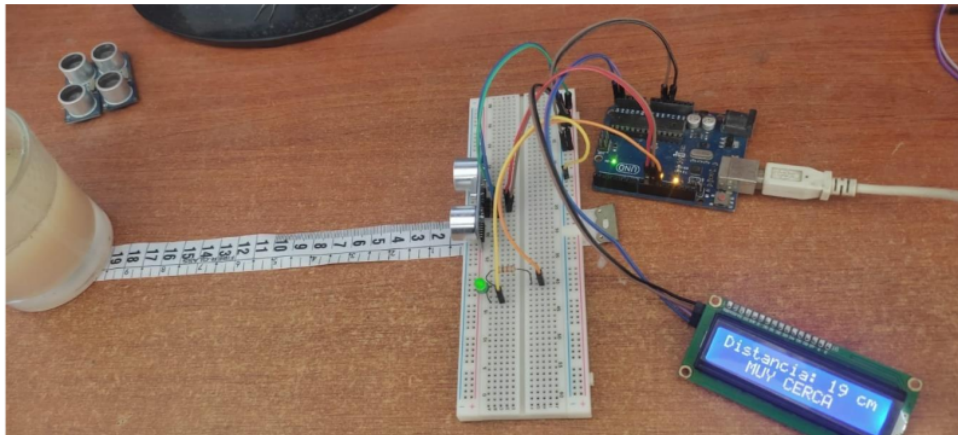
digitalWrite(motor, HIGH);
delay(distancia* 10);
digitalWrite(motor, LOW);
}

if (distancia <= 30 && distancia >= 11)
  lcd.setCursor(0,1);
  lcd.print(" MUY CERCA");
  delay(500);
}

digitalWrite(motor, HIGH);
delay(distancia* 20);
digitalWrite(motor, LOW);
}

// Sketch usa 5990 bytes (11%) del espacio de almacenamiento de programa. El máximo es 32768 bytes.
// Las variables globales usan 302 bytes (24%) de la memoria dinámica, dejando 1448 bytes para las variables locales. El máximo es 2048 bytes.
// Arduino Uno en C/C++
```

Figura 25
Simulación en ARDUINO de Hardware.



Una vez se simuló el comportamiento del código de programación en TINKERCAD y ARDUINO, este algoritmo debe llevarse a una prueba física, para evaluar su correcto desempeño frente a obstáculos reales. También el registro de información, recolección de datos y evaluar su influencia frente a otros códigos de programación, además, por tratarse de Arduino como software, se puede hacer uso de la técnica de fuentes abiertas, es decir, recurrir a la información pública y gratuita.

Figura 26

Conexión virtual de Arduino + Periféricos.

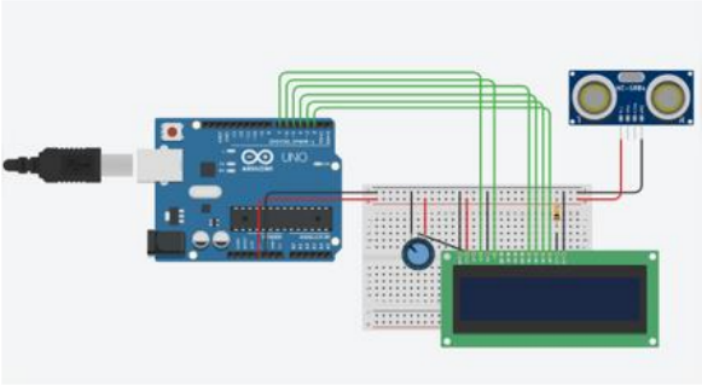
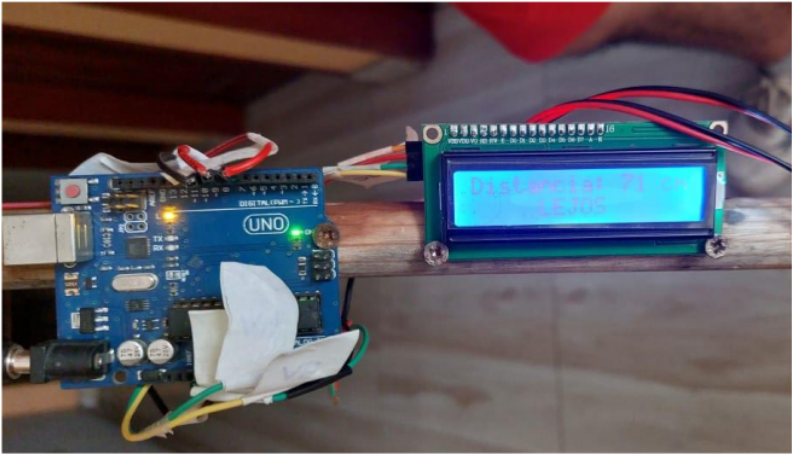


Figura 27

Conexión física de Arduino + Periféricos.



Instrumentos:

Para la recolección de datos que permitieron evaluar la influencia del código de programación se utilizó en primer lugar el monitor serial (Serial.begin(9600)), ya que, las primeras pruebas se realizaron conectadas a un pc a través de un cable USB 2.0.

Figura 28

Recolección de datos. Periférico de entrada + Arduino + PC.

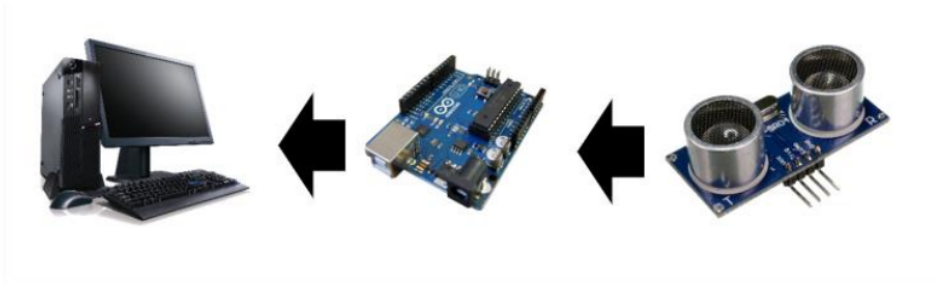
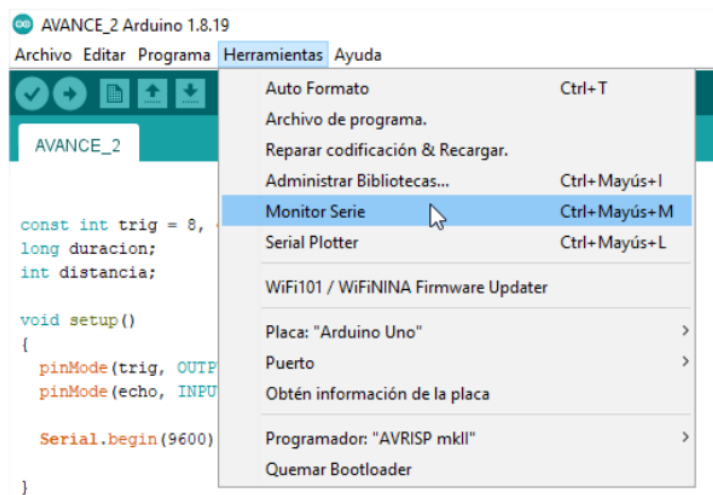


Figura 29

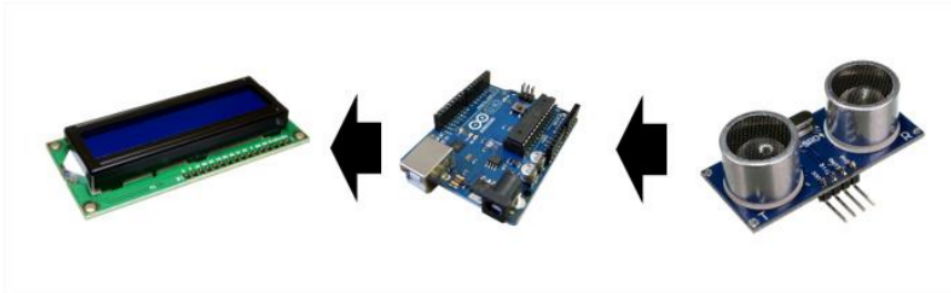
Recolección de datos por SERIAL.BEGIN



Las siguientes pruebas no tenían que ser conectadas a un pc, ya que, se necesitó recolectar información lo más real posible, es decir, en campo abierto. Por eso se hicieron dos modificaciones, la primera es sustituir la conexión a PC por USB 2.0, a simplemente una batería de 9V, que servirá como alimentación de la placa de Arduino. Y la recolección de datos será mediante la visualización en la pantalla LCD 1602 + adaptador LCM1602.

Figura 30

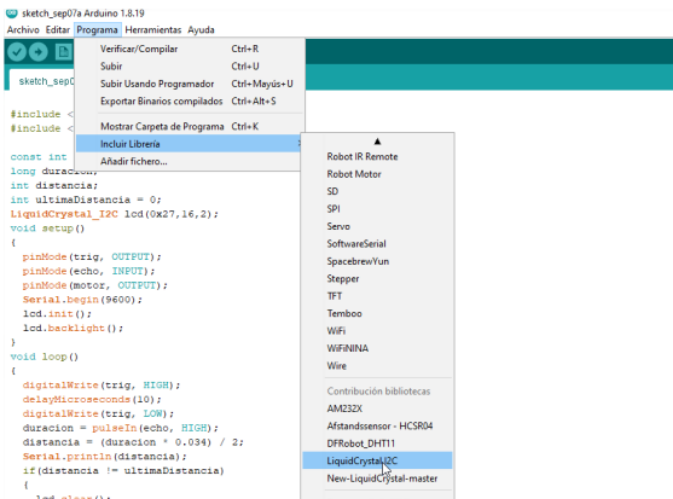
Recolección de datos por Pantalla LCD1602



Debido a que la pantalla LCD1602 usa su propia librería en el IDE de Arduino, en algunas ocasiones tiene que descargarse de otras plataformas. *LiquidCrystal* permite la comunicación con pantallas LCD alfanuméricas.

Figura 31

Recolección de datos por librería LIQUIDCRYSTAL



2.5. Técnicas de procesamiento y análisis de información

Siguiendo la filosofía DevOps y sus pruebas por *Unit tests* se usó el siguiente formato de tablas para determinar el correcto funcionamiento de cada componente como unidad y también el conjunto de los mismos.

Tabla 05

Formato de registro de Datos Técnicos de Fabricación

DATOS TÉCNICOS DE FABRICACIÓN
Tensión de Alimentación (V)
Corriente de Alimentación (A)
Señal de disparo
Ángulo de apertura
Rango de Medición
Dimensiones

Tabla 06

Formato de registro de Equipamiento

EQUIPAMIENTO						
ÍTEM	DESCRIPCIÓN DE EQUIPOS	CANTIDAD	ESTADO		FUNCIÓN	
			Aprobado	Observado	Aprobado	Observado
1						
2						
3						
4						
5						
6						
OBSERVACIONES:			Sin Observaciones			

Para registrar las mediciones eléctricas y pruebas de continuidad en circuitos y bloques de pines, se empleó el siguiente formato

Tabla 07

Formato de registro de los parámetros medidos.

ITEM	DESCRIPCIÓN DE EQUIPO	MARCA	MODELO
1			
2			
3			
	Fecha Calibración		
	Nº Certificación		
	Nombre certificador		

Tabla 08

Formato de registro de tareas generales.

ITEM	OPERACIÓN	EFECTUADA	APROBADO	OBSERVACIONES
1	Conformidad de planos eléctricos y mecánicos, según Datasheet.			
2	Verificación de conexión y pines			
3	Verificación de cada componente electrónico			
4	Codificación de cables y equipos			
	OBSERVACIONES:			

Tabla 09

Formato de registro - Prueba de inspección mecánica

ITEM	OPERACIÓN	EFECTUADA	APROBADO	OBSERVACIONES
1	Verificación de soldadura de componentes electrónicos			
2	Conformidad de dispositivos de maniobra			
	OBSERVACIONES:			Sin Observaciones

Tabla 10

Formato de registro - Prueba de inspección eléctricas

ITEM	OPERACIÓN	EFECTUADA	APROBADO	OBSERVACIONES
1	Continuidad entre pines			
2	Alimentación			
3	Operatividad			
OBSERVACIONES:			Sin Observaciones	

Para la recolección de datos por distancia capturada se usó; el sensor HC SR04 como sensor de distancia, la pantalla LCD 1602 para la visualización de esa distancia capturada y Excel como registro de información.

Figura 32

Formato de Recolección de datos en Excel.

PRUEBA	Recolección de datos
FECHA	
AUTORES	Bringas / Quispe
SERVICIO	Detección
DISPOSITIVO	HC - SR04
OBJETO	
ÁNGULO BASTÓN	
ÁNGULO SENSOR	

NIVEL	DISTANCIA ESTÁNDAR (cm)	(1) DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
MUY CERCA	20						
	40						
	60						
	80						
CERCA	100						
	120						
	140						
	160						
LEJOS	180						
	200						
	220						
	240						
MUY LEJOS	260						
	280						
	300						

2.6. Aspectos éticos en investigación

La presente exploración se realizó de manera transparente, sin las manipulaciones de los datos obtenidos. De la misma manera, por tratarse de un software de código abierto, queda a la disposición de cualquier persona que quiera usarlo con la finalidad de futuras investigaciones.

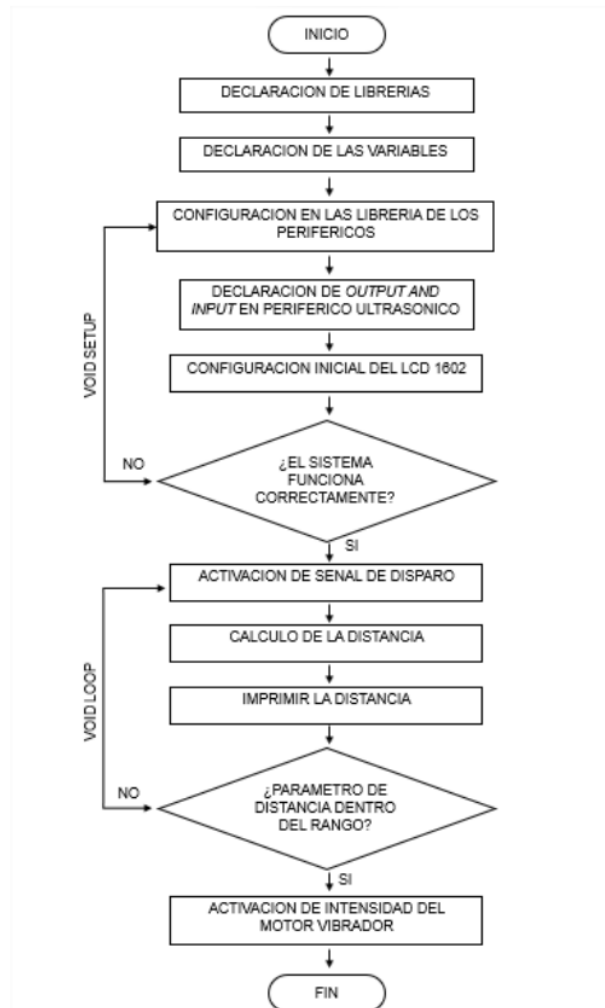
III. RESULTADOS

3.1. Programación de sensor HC – SR04

La programación del microcontrolador ATEMega328P, se realizó mediante el software de Arduino UNO, siguiendo la lógica del esquema de programación mostrada a continuación:

Figura 33

Esquema de programación.



Sin embargo, para llegar al código de programación final, se ha tenido que programar los componentes como unidad para el testeo de su correcto funcionamiento, para

luego pasar a la evaluación como integración. A continuación, se demostró el código de programación aplicado en el proyecto, desde el *Unit Test* hasta el *Integration Test*.

Figura 34

Unit Test – Código de sensor ultrasónico HC SR 04

```
const int trig = 8, echo = 9;
long duracion;
int distancia;
void setup()
{
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  Serial.begin(9600);
}
void loop()
{
  digitalWrite(trig, HIGH);
  delay(10);
  digitalWrite(trig, LOW);
  duracion = pulseIn(echo, HIGH);
  distancia = (duracion * 0.034) / 2;
  Serial.println(distancia);
  delay(200);
}
```

Figura 35

Unit Test – Código de motor vibrador

```

void setup()
{
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
}
void loop()
{
  digitalWrite(8, HIGH);
  delay(300);
  digitalWrite(8, LOW);
  delay(300);
  digitalWrite(9, HIGH);
  delay(2000);
  digitalWrite(9, LOW);
  delay(100);
}

```

Figura 36

Unit Test – Código de pantalla LCD1602

```

#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd (0x27, 2, 1, 0, 4, 5, 6, 7);
void setup()
{
  lcd.setBacklightPin(3,POSITIVE);
  lcd.setBacklight(HIGH);
  lcd.begin(16, 2);
  lcd.clear();
}

```

```

void loop()
{
  lcd.setCursor(0, 0);
  lcd.print("MI TESIS");
  lcd.setCursor(0, 1);
  lcd.print(millis() / 1000);
  lcd.print(" seg.");
}

```

Figura 37

Integration Test – Código pre-final

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
const int trig = 9, echo = 10, motor=11;
long duracion;
int distancia;
int ultimaDistancia = 0;
LiquidCrystal_I2C lcd(0x27,16,2);
void setup()
{
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(motor, OUTPUT);
  Serial.begin(9600);
  lcd.init(); lcd.backlight();
}
void loop()
{
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  duracion = pulseIn(echo, HIGH);
  distancia = (duracion * 0.034) / 2;
}

```



```
Serial.println(distancia);

if(distancia != ultimaDistancia)

{

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Distancia: ");

lcd.print(distancia);

lcd.print(" cm");

ultimaDistancia = distancia;

}

delay(500);

if (distancia <= 300 && distancia >= 220){

lcd.setCursor(0,1);

lcd.print(" MUY LEJOS");

delay(500);

digitalWrite(motor, HIGH);

delay(distancia* 1);

digitalWrite(motor, LOW);

}

if (distancia <= 220 && distancia >= 140){

lcd.setCursor(0,1);

lcd.print(" LEJOS");

delay(500);

digitalWrite(motor, HIGH);
```

```
    delay(distancia* 5);

    digitalWrite(motor, LOW);

}

if (distancia <= 140 && distancia >= 80){

    lcd.setCursor(0,1);

    lcd.print(" CERCA");

    delay(500);

    digitalWrite(motor, HIGH);

    delay(distancia* 10);

    digitalWrite(motor, LOW);

}

if (distancia <= 80 && distancia >= 1){

    lcd.setCursor(0,1);

    lcd.print(" MUY CERCA");

    delay(500);

    digitalWrite(motor, HIGH);

    delay(distancia* 20);

    digitalWrite(motor, LOW);

}

}
```

3.2. Modulo de detección de objetos

Las pruebas del código de programación se realizaron en diferentes objetos, de diferentes formas y tamaños, con la finalidad de buscar los mejores rangos para ajustar en el prototipo de bastón inteligente.

Figura 38

EXCEL – HCSR04 vs Puerta

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	(1) DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023		20	22	21	22	21.67	8.33	DETECTA
AUTORES	Bringas / Quispe	MUY CERCA	40	37	41	38	38.67	3.33	DETECTA
SERVICIO	Detección		60	60	58	59	59.00	1.67	DETECTA
DISPOSITIVO	HC - SR04		80	82	81	79	80.67	0.83	DETECTA
OBJETO	Puerta		100	100	101	100	100.33	0.33	DETECTA
ÁNGULO BASTÓN		CERCA	120	120	121	119	120.00	0.00	DETECTA
ÁNGULO SENSOR			140	140	139	140	139.67	0.24	DETECTA
			160	158	159	158	158.33	1.04	DETECTA
		LEJOS	180	180	179	178	179.00	0.56	DETECTA
			200	200	201	199	200.00	0.00	DETECTA
			220	221	220	221	220.67	0.30	DETECTA
			240	242	241	241	241.33	0.56	DETECTA
		MUY LEJOS	260	261	259	260	260.00	0.00	DETECTA
			280	280	279	278	279.00	0.36	DETECTA
			300	296	296	297	296.33	1.22	DETECTA

Figura 39

EXCEL – HCSR04 vs Mueble

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023		20	17	18	19	18.00	10.00	
AUTORES	Bringas / Quispe	MUY CERCA	40	39	39	40	39.33	1.67	DETECTA
SERVICIO	Detección		60	54	57	58	56.33	6.11	DETECTA
DISPOSITIVO	HC - SR04		80	77	79	80	78.67	1.67	DETECTA
OBJETO	Mueble		100	95	97	98	96.67	3.33	DETECTA
ÁNGULO BASTÓN		CERCA	120	118	117	121	118.67	1.11	DETECTA
ÁNGULO SENSOR			140	135	137	138	136.67	2.38	DETECTA
			160	153	157	155	155.00	3.13	DETECTA
		LEJOS	180	173	177	178	176.00	2.22	DETECTA
			200	209	204	205	206.00	3.00	DETECTA
			220	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			240	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
		MUY LEJOS	260	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			280	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			300	ND	ND	ND	#VALORI	#VALORI	NO DETECTA

Figura 40

EXCEL – HCSR04 vs Cesto de ropa

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023		20	20	23	21	21.33	6.67	DETECTA
AUTORES	Bringas / Quispe	MUY CERCA	40	41	41	40	40.67	1.67	DETECTA
SERVICIO	Detección		60	62	60	61	61.00	1.67	DETECTA
DISPOSITIVO	HC - SR04		80	81	79	79	79.33	0.83	DETECTA
OBJETO	Cesto de ropa		100	101	98	100	99.67	0.33	DETECTA
ÁNGULO BASTÓN		CERCA	120	123	119	121	121.00	0.83	DETECTA
ÁNGULO SENSOR			140	141	135	137	137.67	1.67	DETECTA
			160	159	157	161	159.00	0.63	DETECTA
		LEJOS	180	178	180	179	179.00	0.56	DETECTA
			200	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			220	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			240	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
		MUY LEJOS	260	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			280	ND	ND	ND	#VALORI	#VALORI	NO DETECTA
			300	ND	ND	ND	#VALORI	#VALORI	NO DETECTA

Figura 41

EXCEL – HCSR04 vs Balón

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023								
AUTORES	Bringas / Quispe								
SERVICIO	Detección	MUY CERCA	20	17	18	19	18.59	10.00	DETECTA
DISPOSITIVO	HC - SR04		40	40	38	39	39.00	2.50	DETECTA
OBJETO	Balón		60	56	57	60	57.67	3.89	DETECTA
			80	76	78	81	76.33	2.08	DETECTA
			100	97	99	98	98.00	2.00	DETECTA
ÁNGULO BASTÓN		CERCA	120	117	119	120	118.67	1.11	DETECTA
ÁNGULO SENSOR			140	135	139	141	138.33	1.19	DETECTA
			160	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
		LEJOS	180	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			200	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			220	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
		MUY LEJOS	240	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			260	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			280	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			300	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA

Figura 42

EXCEL – HCSR04 vs Vereda

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023								
AUTORES	Bringas / Quispe								
SERVICIO	Detección	MUY CERCA	20	20	20	21	20.33	1.67	DETECTA
DISPOSITIVO	HC - SR04		40	40	40	39	39.67	0.83	DETECTA
OBJETO	Vereda		60	61	59	60	60.00	0.00	DETECTA
			80	79	78	81	79.33	0.83	DETECTA
			100	106	102	103	103.67	3.67	DETECTA
ÁNGULO BASTÓN		CERCA	120	117	119	118	118.00	1.67	DETECTA
ÁNGULO SENSOR			140	139	142	138	139.67	0.24	DETECTA
			160	161	160	159	160.00	0.00	DETECTA
		LEJOS	180	176	176	178	176.67	1.85	DETECTA
			200	195	198	197	196.67	1.67	DETECTA
			220	218	219	218	218.33	0.76	DETECTA
		MUY LEJOS	240	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			260	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			280	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			300	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA

Figura 43

EXCEL – HCSR04 vs Poste de alumbrado

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023								
AUTORES	Bringas / Quispe								
SERVICIO	Detección	MUY CERCA	20	19	19	20	19.33	3.33	DETECTA
DISPOSITIVO	HC - SR04		40	37	38	40	38.33	4.17	DETECTA
OBJETO	Poste de alumbrado		60	58	58	59	58.33	2.78	DETECTA
			80	77	78	81	78.67	1.67	DETECTA
			100	99	98	100	99.00	1.00	DETECTA
ÁNGULO BASTÓN		CERCA	120	119	118	120	119.00	0.83	DETECTA
ÁNGULO SENSOR			140	138	139	139	138.67	0.96	DETECTA
			160	158	157	159	158.00	1.25	DETECTA
		LEJOS	180	175	177	178	176.67	1.85	DETECTA
			200	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			220	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
		MUY LEJOS	240	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			260	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			280	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			300	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA

Figura 44

EXCEL – HCSR04 vs Personas

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	13/09/2023								
AUTORES	Bringas / Quispe								
SERVICIO	Detección	MUY CERCA	20	18	18	20	18.67	6.67	DETECTA
DISPOSITIVO	HC - SR04		40	37	38	41	38.67	3.33	DETECTA
OBJETO	Personas		60	56	58	59	57.67	3.89	DETECTA
			80	78	79	77	78.00	2.50	DETECTA
			100	99	101	99	99.67	0.33	DETECTA
ÁNGULO BASTÓN		CERCA	120	119	120	118	119.00	0.83	DETECTA
ÁNGULO SENSOR			140	139	138	141	139.33	0.48	DETECTA
		LEJOS	160	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			180	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			200	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			220	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
		MUY LEJOS	240	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			260	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			280	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
			300	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA

Replanteamiento

Los resultados de las mediciones en los objetos presentan una tendencia, con respecto a la forma del objeto, se pudo observar en la figura 40 donde ya nos comienza a presentar una falla de medición a los 200 cm de distancia del bastón hacia el objeto, mientras más irregular es la forma, la falla por medición es más evidente, también se puede observar en la figura 41, donde la falla se presenta a los 160 cm. Si bien estos objetos solamente se pueden encontrar dentro de una vivienda, no significa que sea menos importante, puesto que, el uso del bastón blanco es mecanizado por el cerebro de la persona invidente.

Con respecto a los resultados de la figura 43 y la figura 44, que son obstáculos que están muy presentes al momento de caminar en el exterior, se encontraron fallos a distancias de 200 cm y 160 cm respectivamente, entonces se pudo concluir que el sensor HCSR04 funciona según sus rangos en mayormente figuras planas, donde el rebote del haz del ultrasonido es más eficiente, mientras que en figuras circulares, ovaladas, o simplemente irregulares, los rangos son menores para sacar su mayor eficiencia.

Por tal motivo el código se ajustó a los rangos donde se puede aprovechar su mayor eficiencia para la persona invidente, entonces, el código final termina de la siguiente manera:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
const int trig = 9, echo = 10, motor=11;
long duracion;
int distancia;
int ultimaDistancia = 0;
LiquidCrystal_I2C lcd(0x27,16,2);
void setup()
{
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(motor, OUTPUT);
  Serial.begin(9600);
  lcd.init(); lcd.backlight();
}
```

```

void loop()
{
digitalWrite(trig, HIGH);
delayMicroseconds(10);
digitalWrite(trig, LOW);
duracion = pulseIn(echo, HIGH);
distancia = (duracion * 0.034) / 2;
Serial.println(distancia);

if(distancia != ultimaDistancia)
{
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Distancia: ");
lcd.print(distancia);
lcd.print(" cm");
ultimaDistancia = distancia;
}

delay(500);

if (distancia <= 160 && distancia >= 100){

lcd.setCursor(0,1);
lcd.print(" LEJOS");
delay(500);
digitalWrite(motor, HIGH);
delay(distancia* 5);
digitalWrite(motor, LOW);
}
}

```

```

if (distancia <= 100 && distancia >= 60){

lcd.setCursor(0,1);

lcd.print(" CERCA");

delay(500);

digitalWrite(motor, HIGH);

delay(distancia* 5);

digitalWrite(motor, LOW);

}

if (distancia <= 60 && distancia >= 20){

lcd.setCursor(0,1);

lcd.print(" MUY CERCA");

delay(500);

digitalWrite(motor, HIGH);

delay(distancia* 10);

digitalWrite(motor, LOW);

}

}

```

Luego de la modificación del código, se obtuvo los siguientes resultados:

Figura 45

EXCEL – HCSR04 vs Puerta

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	(1) DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023	MUY CERCA	20	21	20	19	20.00	0.00	DETECTA
AUTORES	Bringas / Quispe		40	41	41	40	40.67	1.67	DETECTA
SERVICIO	Detección		60	61	61	60	60.67	1.11	DETECTA
DISPOSITIVO	HC - SR04		80	81	81	80	80.67	1.11	DETECTA
OBJETO	Puerta		100	101	102	102	101.67	1.67	DETECTA
ÁNGULO BASTÓN		LEJOS	100	101	102	102	101.67	1.67	DETECTA
ÁNGULO SENSOR			120	123	121	122	122.00	1.67	DETECTA
			140	143	142	143	142.67	1.90	DETECTA
			160	161	162	160	161.00	0.63	DETECTA

Figura 46

EXCEL – HCSR04 vs Mueble

PRUEBA		Recolección de datos		NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA		30/09/2023		MUY CERCA	20	21	21	19	20.33	1.67	DETECTA
AUTORES		Bingas / Quispe			40	40	42	41	41.00	2.50	DETECTA
SERVICIO		Detección			60	60	58	59	59.00	1.67	DETECTA
DISPOSITIVO		HC - SR04			60	60	58	59	59.00	1.67	DETECTA
OBJETO		Mueble			80	60	58	59	59.00	1.67	DETECTA
ÁNGULO BASTÓN					80	81	79	80	80.00	0.00	DETECTA
ÁNGULO SENSOR					100	102	103	101	102.00	2.00	DETECTA
					100	102	103	101	102.00	2.00	DETECTA
					120	121	121	122	121.33	1.11	DETECTA
					140	140	142	141	141.00	0.71	DETECTA
					160	159	158	161	159.33	0.42	DETECTA

Figura 47

EXCEL – HCSR04 vs Cesto de ropa

PRUEBA		Recolección de datos		NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA		30/09/2023		MUY CERCA	20	20	20	20	20.00	0.00	DETECTA
AUTORES		Bingas / Quispe			40	41	39	40	40.00	0.00	DETECTA
SERVICIO		Detección			60	62	61	60	61.00	1.67	DETECTA
DISPOSITIVO		HC - SR04			60	52	61	60	61.00	1.67	DETECTA
OBJETO		Cesto de ropa			80	81	79	80	80.00	0.00	DETECTA
ÁNGULO BASTÓN					100	101	101	100	100.67	0.67	DETECTA
ÁNGULO SENSOR					100	101	101	100	100.67	0.67	DETECTA
					120	123	119	118	120.00	0.00	DETECTA
					140	141	137	138	138.67	0.95	DETECTA
					160	159	158	158	158.33	1.04	DETECTA

Figura 48

EXCEL – HCSR04 vs Balón

PRUEBA		Recolección de datos		NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA		30/09/2023		MUY CERCA	20	20	19	19	19.33	3.33	DETECTA
AUTORES		Bingas / Quispe			40	39	38	39	38.67	3.33	DETECTA
SERVICIO		Detección			60	60	58	59	59.00	1.67	DETECTA
DISPOSITIVO		HC - SR04			60	60	58	59	59.00	1.67	DETECTA
OBJETO		Balón			80	80	78	79	79.00	1.25	DETECTA
ÁNGULO BASTÓN					100	99	99	98	98.67	1.33	DETECTA
ÁNGULO SENSOR					100	99	99	98	98.67	1.33	DETECTA
					120	121	118	119	119.33	0.56	DETECTA
					140	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA
					160	ND	ND	ND	#VALOR!	#VALOR!	NO DETECTA

Figura 49

EXCEL – HCSR04 vs Vereda

PRUEBA		Recolección de datos		NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA		30/09/2023		MUY CERCA	20	19	20	20	19.67	1.67	DETECTA
AUTORES		Bingas / Quispe			40	39	40	39	39.33	1.67	DETECTA
SERVICIO		Detección			60	58	58	59	58.33	2.78	DETECTA
DISPOSITIVO		HC - SR04			60	58	58	59	58.33	2.78	DETECTA
OBJETO		Vereda			80	77	78	79	78.00	2.50	DETECTA
ÁNGULO BASTÓN					100	98	99	98	98.33	1.67	DETECTA
ÁNGULO SENSOR					100	98	99	98	98.33	1.67	DETECTA
					120	117	118	119	118.00	1.67	DETECTA
					140	141	142	142	141.67	1.19	DETECTA
					160	161	160	161	160.67	0.42	DETECTA

Figura 50

EXCEL – HCSR04 vs Poste de alumbrado

PRUEBA		Recolección de datos		NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA		30/09/2023		MUY CERCA	20	19	20	20	19.67	1.67	DETECTA
AUTORES		Bingas / Quispe			40	40	39	39	39.33	1.67	DETECTA
SERVICIO		Detección			60	58	58	60	58.67	2.22	DETECTA
DISPOSITIVO		HC - SR04			60	58	58	60	58.67	2.22	DETECTA
OBJETO		Poste de alumbrado			80	79	78	79	78.67	1.67	DETECTA
ÁNGULO BASTÓN					100	99	101	99	99.67	0.33	DETECTA
ÁNGULO SENSOR					100	99	101	99	99.67	0.33	DETECTA
					120	118	119	119	118.67	1.11	DETECTA
					140	139	139	138	138.67	0.95	DETECTA
					160	158	159	158	158.33	1.04	DETECTA

Figura 51

EXCEL – HCSR04 vs Personas

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023		20	19	20	19	19.33	3.33	DETECTA
AUTORES	Bingas / Quispe	MUY CERCA	40	38	38	41	39.00	2.50	DETECTA
SERVICIO	Detección		60	58	58	57	57.67	3.89	DETECTA
DISPOSITIVO	HC - SR04		80	58	58	57	57.67	3.89	DETECTA
OBJETO	Personas	CERCA	90	78	79	77	78.00	2.50	DETECTA
ANGULO BASTÓN			100	99	101	99	99.67	0.33	DETECTA
ANGULO SENSOR			100	99	101	99	99.67	0.33	DETECTA
		LEJOS	120	119	120	119	119.33	0.56	DETECTA
			140	NA	NA	NA	#VALOR!	#VALOR!	NO DETECTA
			160	NA	NA	NA	#VALOR!	#VALOR!	NO DETECTA

Como se puede observar los ajustes de la reprogramación del código, nos presenta menos márgenes de error, y valores casi por completos en los rangos de 20 cm a 160 cm, en casi todas objetos planos o irregulares, esto significa mayor eficiencia. Por tal motivo, se usó este código como final, para seguir a la fase de evaluación frente a otros algoritmos.

3.3. Influencia del código de programación

Para poder evaluar el algoritmo de programación frente a otros algoritmos, se llevó a una tabla en común, como se nos muestra en el artículo de Llanely B. Martínez Núñez, Jairo A. Molino Tinoco, en la etapa de resultados que nos muestra su prototipo.

Tabla 11

Pruebas y Resultados obtenidos Llanely B. Martínez, Jairo A. Molina.

<i>Llanely B. Martínez, Jairo A. Molina</i>				
Tipo de obstáculo	Cara lisa frente al sensor	Liso con esquina frente a sensor	Redondo	Amorfo
Distancia máxima de detección	2.56m	1.53m	2.06m	2.03m
Error promedio	+/- 0.06m	+/- 0.1m	+/- 0.08m	+/- 0.12m

Para esto, se recolectó los datos de las figuras con las características correspondiente, de acuerdo al tipo de obstáculo, para posteriormente convertir el margen de error en las unidades respondientes, de la misma manera la distancia máxima de detección, para finalmente obtener nuestra tabla final.

Figura 52

Margen de error en (cm) – HCSR04 vs Puerta

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	(1) DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023								
AUTORES	Bringas / Quispe	MUY CERCA	20	21	20	19	20.00	0.00	DETECTA
SERVICIO	Detección		40	41	41	40	40.67	1.67	DETECTA
DISPOSITIVO	HC - SR04		60	61	61	60	60.67	1.11	DETECTA
OBJETO	Puerta	CERCA	60	61	61	60	60.67	1.11	DETECTA
			80	81	82	81	81.33	1.67	DETECTA
			100	101	102	102	101.67	1.67	DETECTA
			100	101	102	102	101.67	1.67	DETECTA
ÁNGULO BASTÓN		LEJOS	120	123	121	122	122.00	1.67	DETECTA
ÁNGULO SENSOR			140	143	142	143	142.67	1.90	DETECTA
			160	161	162	160	161.00	0.63	DETECTA
							MARGEN DE ERROR EN (CM)	0.006250	

Figura 53

Margen de error en (cm) – HCSR04 vs Vereda

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023								
AUTORES	Bringas / Quispe	MUY CERCA	20	19	20	20	19.67	1.67	DETECTA
SERVICIO	Detección		40	39	40	39	39.33	1.67	DETECTA
DISPOSITIVO	HC - SR04		60	58	58	59	58.33	2.78	DETECTA
OBJETO	Vereda	CERCA	60	58	58	59	58.33	2.78	DETECTA
			80	77	78	79	78.00	2.50	DETECTA
			100	98	99	98	98.33	1.67	DETECTA
			100	98	99	98	98.33	1.67	DETECTA
ÁNGULO BASTÓN		LEJOS	120	117	118	119	118.00	1.67	DETECTA
ÁNGULO SENSOR			140	141	142	142	141.67	1.19	DETECTA
			160	161	160	161	160.67	0.42	DETECTA
							MARGEN DE ERROR EN (CM)	0.00417	

Figura 54

Margen de error en (cm) – HCSR04 vs Poste de Alumbrado

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023								
AUTORES	Bringas / Quispe	MUY CERCA	20	19	20	20	19.67	1.67	DETECTA
SERVICIO	Detección		40	40	39	39	39.33	1.67	DETECTA
DISPOSITIVO	HC - SR04		60	58	58	60	58.67	2.22	DETECTA
OBJETO	Poste de alumbrado	CERCA	60	58	58	60	58.67	2.22	DETECTA
			80	79	78	79	78.67	1.67	DETECTA
			100	99	101	99	99.67	0.33	DETECTA
			100	99	101	99	99.67	0.33	DETECTA
ÁNGULO BASTÓN		LEJOS	120	118	119	119	118.67	1.11	DETECTA
ÁNGULO SENSOR			140	139	139	138	138.67	0.95	DETECTA
			160	159	159	160	159.33	0.42	DETECTA
							MARGEN DE ERROR EN (CM)	0.00417	

Figura 55

Margen de error en (cm) – HCSR04 vs Personas

PRUEBA	Recolección de datos	NIVEL	DISTANCIA ESTÁNDAR (cm)	DISTANCIA CAPTURADA (cm)	(2) DISTANCIA CAPTURADA (cm)	(3) DISTANCIA CAPTURADA (cm)	DISTANCIA PROMEDIO (cm)	MARGEN DE ERROR (%)	ESTADO DEL SENSOR
FECHA	30/09/2023								
AUTORES	Bringas / Quispe	MUY CERCA	20	19	20	19	19.33	3.33	DETECTA
SERVICIO	Detección		40	38	38	41	39.00	2.50	DETECTA
DISPOSITIVO	HC - SR04		60	58	58	57	57.67	3.89	DETECTA
OBJETO	Personas	CERCA	60	58	58	57	57.67	3.89	DETECTA
			80	78	79	77	78.00	2.50	DETECTA
			100	99	101	99	99.67	0.33	DETECTA
			100	99	101	99	99.67	0.33	DETECTA
ÁNGULO BASTÓN		LEJOS	120	119	120	119	119.33	0.56	DETECTA
ÁNGULO SENSOR			140	NA	NA	NA	#VALORI	#VALORI	NO DETECTA
			160	NA	NA	NA	#VALORI	#VALORI	NO DETECTA
							MARGEN DE ERROR EN (CM)	0.005556	

Tabla 12*Pruebas y Resultados obtenidos Walter S. Bringas, Erick D. Quispe.*

<i>Walter S. Bringas, Erick D. Quispe</i>				
Tipo de obstáculo	Cara lisa frente al sensor	Liso con esquina frente a sensor	Redondo	Amorfo
Distancia máxima de detección	1.60m	1.60m	1.60m	1.20m
Error promedio	+/- 0.00625m	+/- 0.00417m	+/- 0.00417m	+/- 0.00556m

IV. DISCUSIÓN

Las pruebas de medición del código de programación en el bastón inteligente fueron realizadas en un entorno acorde al movimiento diario que puede tener una persona invidente en la calle o en su hogar; esto nos permite dar un análisis del comportamiento del sensor y del código en una situación real.

En la realización del primer lote de pruebas, se obtuvieron errores de lectura de los objetos puestos a prueba a una distancia promedio de 300 cm; es decir, en distancias mayores el sensor presentaba errores de captación, y el valor que nos mostraba la pantalla LCD del bastón blanco procesado por el Arduino era inexacto. Esta metodología de prueba se compara a la realizada en otras investigaciones donde también se usó un sensor de ultrasonido HC-SR04 y objetos de prueba para medición, similares. (Martínez-Molina et al, 2019; Shah-Khan et al, 2021). En cada investigación, los datos obtenidos comprueban que la eficiencia del funcionamiento del sensor frente a objetos con formas irregulares depende de la posición y la distancia a la que se encuentra el objeto a medir. Si esto no se cumple, el proceso de captación del sensor no logra realizarse eficazmente y provoca un rompimiento en el eslabón del resultado final, que es lograr advertir al usuario portador de que está próximo a un objeto cercano.

Para mejorar ese aspecto, se realizó un ajuste necesario en el código del sensor, ajustando el rango de distancias de captación de este, con el fin de evitar las inexactitudes mencionadas; esta solución también fue aplicada en otras investigaciones donde buscaban mejorar el sistema automatizado de un bastón inteligente (Ayala Cruz, 2011; Ñiacasha Utreras, 2004). En el último lote de pruebas, los datos obtenidos comprueban una mejor captación de distancias por parte del sensor y el código de programación; lo cual nos permite tener un margen de error promedio necesario para que el prototipo de bastón inteligente pueda ser tomado en cuenta para un posible uso, o como base para el desarrollo de un prototipo más sofisticado.

Al comparar los resultados obtenidos con otras investigaciones similares, en las que usan el mismo sensor, pero cambian los ajustes del código de programación, se pudo contrastar que el margen de error de captación es menor en cada uno de los tipos de obstáculos que se presentaron en la medición de los objetos de ensayo. En principio, nuestros

rangos de captación del sensor, tienen un margen de error general entre $\pm 0.0417 - 0.00625$ m; a una distancia de prueba máxima de 160 cm. Mientras que otros estudios obtuvieron rangos que oscilaron entre $\pm 0.08 - 0.12$ m; a una distancia de prueba máxima de 240 cm (Martínez-Molina et al). Cabe mencionar que hay investigaciones que a pesar de disminuir el rango de captación un 50%, es decir, con una distancia de prueba máxima de 80 cm, con un código de programación similar obtuvieron resultados que oscilaron entre $\pm 0.0041 - 0.0057$ m. (Collantes Rodríguez,2020).

V. CONCLUSIONES

El código de programación realizado en Arduino influye de manera significativa en la detección de objetos a través de un sensor de ultrasonido para un bastón inteligente, ya que permite realizar ajustes en los rangos de captación de distancia, contribuyendo a una lectura más precisa del sensor y a su vez, contribuyendo a una mayor seguridad en el uso del prototipo de parte del usuario invidente.

La presente investigación revela una realidad que no se puede obviar, sin la realización de las pruebas de ensayo de detección en un entorno real, no sólo una, sino varias veces; no hubiese sido posible poder entregar los resultados contrastados en el posible uso del bastón inteligente; ni se hubiese podido realizar la comparación con otros estudios que trabajaron un algoritmo similar.

El integrado del Arduino permitió el desarrollo del proyecto gracias a su interfaz de fácil manejo para crear, compilar y cargar códigos en los microcontroladores Arduino. Sin esta herramienta, los lotes de prueba hubiesen sido más tardíos y los costos de conseguir otro software y microcontrolador hubiesen sido más elevados.

VI. RECOMENDACIONES

Es importante estudiar a fondo el sistema de captación de señal del prototipo de bastón inteligente y es que con un poco más de recursos se hubiese podido conseguir sensores de ultrasonido mucho más sofisticados, y una placa de Arduino mejor compacta, que permita obtener mejor precisión en la detección de objetos.

La programación es una rama de la ingeniería que actualmente es muy importante en la industria y ha tenido un apogeo mucho mayor gracias a la automatización; por consiguiente, ser alguien capacitado en estos temas nos harán profesionales más competitivos ante el sistema laboral.

Esta investigación está enfocada en presentar una solución ante la problemática de la ceguera que aquejan nuestra sociedad. Por eso buscamos incentivar la investigación y la creatividad junto con la mejora de proyectos que ya se han realizado con anterioridad en nuestra casa de estudios con el fin de seguir sirviendo a la comunidad.

VII. REFERENCIAS BIBLIOGRÁFICAS

- ⁸ Aragón-Loza, M.A. (2020). *Revisión de la literatura de modelos matemáticos para el tránsito de personas con discapacidad visual a través de rutas accesibles y seguras*. [Tesis de pregrado, Pontificia Universidad Católica del Perú]. Archivo digital ¹²
[https://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/18490/ARAGON_LOZA_MARCO_REVISI%
c3%93N_LITERATURA_MODELOS.pdf?sequence=1&isAllowed=y](https://tesis.pucp.edu.pe/repositorio/bitstream/handle/20.500.12404/18490/ARAGON_LOZA_MARCO_REVISI%c3%93N_LITERATURA_MODELOS.pdf?sequence=1&isAllowed=y)
- ¹² Archivo digital. <http://repositorio.umsa.bo/xmlui/handle/123456789/29622>
- Ayala-Cruz, E.L. (2011). *Diseño y construcción del prototipo de un sistema electrónico por ultrasonido para medir distancias aplicada a un bastón blanco*. [Tesis de pregrado, Universidad Politécnica Salesiana Sede Cuenca]. ⁴
- ⁶ Bauer C., Hirsch G., Zajac L., Koo BB., Collignon O. y Merabet L. (22 de mayo de 2017). *Multimodal MR-imaging reveals large-scale structural and functional connectivity changes in profound early blindness*. ⁶
<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0173064>
- Bravo-Dussan, J.G. (2019). *Prototipo de brazalete como complemento al bastón para personas con discapacidad visual*. Archivo digital.
<https://revistas.sena.edu.co/index.php/GRINNDA/article/view/2720/3247>
- Cbenga Dada, E. (octubre 05, 2017). *Smart Walking Stick for Visually Impaired People Using Ultrasonic Sensors and Arduino*. Academia: Accelerating the world's research. Vol 9. doi: 10.21817/ijet/2017/v9i5/170905302
- Collantes-Rodríguez, B.D. *Diseño de un prototipo de un bastón inteligente con sensores para personas con discapacidad visual y física en la carrera de Ingeniería en Computación y Redes*. [Tesis de Pregrado, Universidad Estatal del Sur de Manabí]. Archivo digital. ¹⁰
<http://repositorio.unesum.edu.ec/bitstream/53000/2249/1/COLLANTES%20RODRIGUEZ%20BRYAN%20DAMIAN.pdf>
- Descubre Arduino (s.f.). *Tabla comparativa con los tipos de microcontroladores Arduino más utilizados en proyectos electrónicos*. ⁵
<https://descubrearduino.com/comparamos-varias-de-las-arduino-mas-utilizadas/>

- Dos Santos, A., Medola, F., Cinelli M., García A. y Sandnes F. (17 de febrero de 2020). *Are electronic white canes better those traditional canes? A comparative study with blind and blindfolded participants*. Springer Link.
<https://link.springer.com/article/10.1007/s10209-020-00712-z#:~:text=Conclusions,to%20the%20traditional%20white%20cane.>
- Dupleich, J.L. García, M.R. García, C.B. (2015) AVANCES. LA TECNOLOGÍA Y PERSONAS CIEGAS. <http://sedici.unlp.edu.ar/handle/10915/50984>
- Fernández-Llontop. R.J. (2021). *Bastón sensorial geo localizador inteligente para apoyar en el desplazamiento de personas invidentes en la Organización Regional de Ciegos del Perú-Chiclayo*. [Tesis de Pregrado, Universidad Católica Santo Toribio de Mogrovejo]. Archivo digital.
https://tesis.usat.edu.pe/bitstream/20.500.12423/3213/1/TL_FernandezLlontop Roger.pdf
- Harari I., Altoaguirre P. y Torales R. (2019). *Desarrollan un bastón y anteojos inteligentes para personas con discapacidad visual y ceguera*. Archivo digital.
<https://noticiasdelaciencia.com/art/33234/desarrollan-un-baston-y-anteojos-inteligentes-para-personas-con-discapacidad-visual-y-ceguera>
- HETPRO (s.f.). *Sensor ultrasónico HC SR04 para crear una alarma con Arduino*.
<https://hetpro-store.com/TUTORIALES/sensor-hc-sr04/>
<https://dspace.ups.edu.ec/bitstream/123456789/1080/12/UPS-CT002123.pdf>
- Huanca Coaquira. Y. (2022). *Bastón blanco con sensores ultrasónicos y vibraciones*. [Trabajo de Grado, Universidad Mayor de San Andrés].
- Lanuez-Bayolo, M. (2006). *Metodología de la investigación educativa*. Archivo digital. <https://www.ucm.es/data/cont/media/www/pag-55163/2Metodos.pdf>
- León-Meza, X.E. (2021). *Diseño y evaluación de prototipo de bastón con sensores electrónicos para personas no videntes de la unidad de rehabilitación del hospital IESS Quito Sur*. [Tesis de pregrado, Universidad de Guayaquil].
- Lescano-Irala J.G. y Cabral-Franco M.C. (2015). *Prototipo de bastón inteligente para invidentes*. Archivo digital
<http://publicaciones.uni.edu.py/index.php/rseisa/article/view/140>
- Lizárraga-González, C.R. (2018). *Propuesta para el diseño de un bastón electrónico para personas invidentes que mejorará la calidad de su desplazamiento diario*. [Tesis de pregrado, Universidad Continental de Arequipa]. Archivo digital.

https://repositorio.continental.edu.pe/bitstream/20.500.12394/5202/1/IV_FIN_10_TE_Lizarraga_Gonzales_2018.pdf

López-Ojeda, J. A., Navarro-Morales, M. J. y Silva Cruz, E. M. *Detección de objetos*

³⁶
por medio de escaneo ultrasónico. Archivo digital.

⁴⁴
<http://www.itcelaya.edu.mx/ojs/index.php/pistas/article/view/1650/1447>

M. M. Shah, M. N. R. Khan, M. R. A. Khan, M. M. H. Plabon and M. A. Razzak, "A Cost-Effective Smart Walking Stick for Visually Impaired People," *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, 2021, pp. 1582-1585, doi: 10.1109/ICCES51350.2021.9489010.

Marquina-García, D.C. (2021). *Bastón Inteligente basado en sensores para mejorar el desplazamiento de personas con discapacidad visual de la Asociación Fe y Esperanza de la ciudad de Tarapoto*. [Tesis de Grado, Universidad Nacional de San Martín - Tarapoto]. Archivo digital. <http://hdl.handle.net/11458/4004>

Martínez Fuentes, V. (2014). *Introducción a la plataforma Arduino y al Sensor ultrasónico HC-SR04: experimentando en una aplicación para medición de*
¹⁹
distancias. ⁴⁸ [Trabajo de Fin de Grado, Universidad Carlos III de Madrid].

⁴¹
Martínez-Núñez, Ll. B., Molina-Tinoco, J. A., Peñaloza-Mendoza, G. R., Hernández-Sánchez, N.K. y Guizar-Rojas, E. *Construcción de bastón inteligente para personas con discapacidad visual*. Archivo digital.

<http://memoriascnib.mx/index.php/memorias/article/view/81/77>

¹⁴
MINSA (17 de noviembre de 2014). *En el Perú cerca de 160 000 personas son invidentes y unas 600 000 sufren de alguna discapacidad*. Gob.pe.

<https://www.gob.pe/institucion/minsa/noticias/31145-en-el-peru-cerca-de-160-000-personas-son-invidentes-y-unas-600-000-sufren-de-alguna-discapacidad-visual>

Ñiacasha-Utreras, N. L. (2004). *Desarrollo de un dispositivo que mida la distancia a un objeto emulando el efecto de un bastón blanco para personas invidentes*. [Proyecto de control, Escuela Politécnica Nacional de Quito]. Archivo digital: <http://bibdigital.epn.edu.ec/handle/15000/5096>

OMS (26 de febrero de 2021). *Ceguera y discapacidad visual*. Organización Mundial de la Salud. ¹⁵ <https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment>

- Ontiveros-Paredes, S.D., Rojas-Balbuena. D., Martínez-Paredes, J. (2014). *Diseño y construcción de un bastón blanco electrónico para personas invidentes*. Archivo digital. <https://www.redalyc.org/articulo.oa?id=61448035002>
- Paiba-Díaz, C.Y. (2020). *Implementar un bastón inteligente para facilitar la identificación de objetos circundantes y productos de un almacén para la población invidente*. [Proyecto de Grado, Universidad Santo Tomas]. Archivo digital <https://repository.usta.edu.co/bitstream/handle/11634/29849/2020carlospaiba.pdf?sequence=6&isAllowed=y>
- Paucar-Espinoza, L. G. (2018). *Gafas y bastón inteligente para una persona invidente*. [Tesis de pregrado, Pontificia Universidad Tecnológica Israel]. Archivo digital <http://repositorio.uisrael.edu.ec/bitstream/47000/1566/1/UISRAEL-EC-ELDT-378.242-2018-011.pdf>
- Paul Riordan, E. Cunningham Jr., E. (2012) *Oftalmología general* (pp. 400-404) (18ª ed.). MC Graw Hill Lange.
- Proyectos con Arduino (s.f.). *Sensor ultrasónico HC SR04 para medir distancias con Arduino*. <https://proyectosconarduino.com/sensores/sensor-de-distancia-hc-sr04/>
- S. Mohapatra, S. Rout, V. Tripathi, T. Saxena and Y. Karuna, "Smart Walking Stick for Blind Integrated with SOS Navigation System," *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, 2018, pp. 441-447, doi: 10.1109/ICOEI.2018.8553935.
- Srinidhi Srinivasan, M.R (abril 25, 2019). *Smart Walking Stick*. 3rd International Conference on Trends in Electronics and Informatics (ICOEI). [10.1109/ICOEI.2019.8862753](https://doi.org/10.1109/ICOEI.2019.8862753)
- Vela-Inoñan. E.J. *Diseño e implementación de un bastón ergonómico con sistema de posicionamiento global para mejorar el desplazamiento de personas invidentes en el centro "La unión nacional de ciegos del Perú"*. [Tesis de Pregrado, Universidad de Ciencias y Humanidades]. Archivo digital. <https://repositorio.uch.edu.pe/handle/20.500.12872/505>
- World Health Organization (2019). *World report on Vision*. <https://www.iapb.org/es/learn/resources/the-world-report-on-vision/>

Anexo 1: Instrumentos de recolección de la información

Ficha Técnica de HC – SR04

HC-SR04 Ultrasonic Sensor

Elijah J. Morgan

Nov. 16 2014

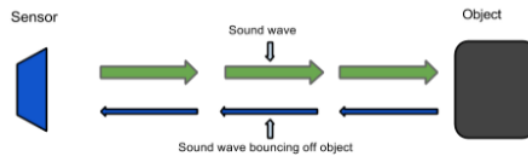
The purpose of this file is to explain how the HC-SR04 works. It will give a brief explanation of how ultrasonic sensors work in general. It will also explain how to wire the sensor up to a microcontroller and how to take/interpret readings. It will also discuss some sources of errors and bad readings.

1. How Ultrasonic Sensors Work
2. HC-SR04 Specifications
3. Timing chart, Pin explanations and Taking Distance Measurements
4. Wiring HC-SR04 with a microcontroller
5. Errors and Bad Readings



1. How Ultrasonic Sensors Work

Ultrasonic sensors use sound to determine the distance between the sensor and the closest object in its path. How do ultrasonic sensors do this? Ultrasonic sensors are essentially sound sensors, but they operate at a frequency above human hearing.



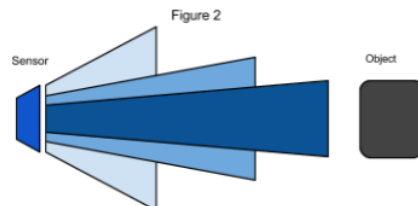
The sensor sends out a sound wave at a specific frequency. It then listens for that specific sound wave to bounce off of an object and come back (Figure 1). The sensor keeps track of the time between sending the sound wave and the sound wave returning. If you know how fast something is going and how long it is traveling you can find the distance traveled with equation 1.

$$\text{Equation 1. } d = v \times t$$

The speed of sound can be calculated based on the a variety of atmospheric conditions, including temperature, humidity and pressure. Actually calculating the distance will be shown later on in this document.

It should be noted that ultrasonic sensors have a cone of detection, the angle of this cone varies with distance, Figure 2 show this relation. The ability of a sensor to

detect an object also depends on the objects orientation to the sensor. If an object doesn't present a flat surface to the sensor then it is possible the sound wave will bounce off the object in a way that it does not return to the sensor.



2. HC-SR04 Specifications

The sensor chosen for the Firefighting Drone Project was the HC-SR04. This section contains the specifications and why they are important to the sensor module. The sensor modules requirements are as follows.

- Cost
- Weight
- Community of hobbyists and support
- Accuracy of object detection
- Probability of working in a smoky environment
- Ease of use

The HC-SR04 Specifications are listed below. These specifications are from the Cytron Technologies HC-SR04 User's Manual (source 1).

- Power Supply: +5V DC
- Quiescent Current: <2mA
- Working current: 15mA
- Effectual Angle: <15°
- Ranging Distance: 2-400 cm
- Resolution: 0.3 cm
- Measuring Angle: 30°
- Trigger Input Pulse width: 10uS
- Dimension: 45mm x 20mm x 15mm
- Weight: approx. 10 g

The HC-SR04's best selling point is its price; it can be purchased at around \$2 per unit.

3. Timing Chart and Pin Explanations

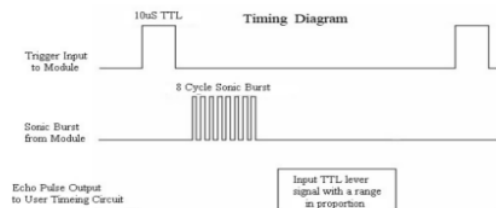
The HC-SR04 has four pins, VCC, GND, TRIG and ECHO; these pins all have different functions. The VCC and GND pins are the simplest -- they power the HC-SR04. These pins need to be attached to a +5 volt source and ground respectively. There is a single control pin: the TRIG pin. The TRIG pin is responsible for sending the ultrasonic burst. This pin should be set to HIGH for 10 μ s, at which point the HC-SR04 will send out an eight cycle sonic burst at 40 kHz. After a sonic burst has been sent the ECHO pin will go HIGH. The ECHO pin is the data pin -- it is used in taking distance measurements. After an ultrasonic burst is sent the pin will go HIGH, it will stay high until an ultrasonic burst is detected back, at which point it will go LOW.

Taking Distance Measurements

The HC-SR04 can be triggered to send out an ultrasonic burst by setting the TRIG pin to HIGH. Once the burst is sent the ECHO pin will automatically go HIGH. This pin will remain HIGH until the the burst hits the sensor again. You can calculate the distance to the object by keeping track of how long the ECHO pin stays HIGH. The time ECHO stays HIGH is the time the burst spent traveling. Using this measurement in equation 1 along with the speed of sound will yield the distance travelled. A summary of this is listed below, along with a visual representation in Figure 2.

1. Set TRIG to HIGH
2. Set a timer when ECHO goes to HIGH
3. Keep the timer running until ECHO goes to LOW
4. Save that time
5. Use equation 1 to determine the distance travelled

Figure 3
Source 2



To interpret the time reading into a distance you need to change equation 1. The clock on the device you are using will probably count in microseconds or smaller. To use equation 1 the speed of sound needs to be determined, which is 343 meters per second at standard temperature and pressure. To convert this into more useful form use equation 2 to change from meters per second to microseconds per centimeter. Then equation 3 can be used to easily compute the distance in centimeters.

$$\text{Equation 2. Distance} = \frac{\text{Speed}}{170.15 \text{ m}} \times \frac{\text{Meters}}{100 \text{ cm}} \times \frac{1e6 \mu\text{S}}{170.15 \text{ m}} \times \frac{58.772 \mu\text{S}}{\text{cm}}$$

$$\text{Equation 3. Distance} = \frac{\text{time}}{58} = \frac{\mu\text{S}}{\mu\text{S/cm}} = \text{cm}$$

4. Wiring the HC-SR04 to a Microcontroller

This section only covers the hardware side. For information on how to integrate the software side, look at one of the links below or look into the specific microcontroller you are using.

The HC-SR04 has 4 pins: VCC, GND, TRIG and ECHO.

1. VCC is a 5v power supply. This should come from the microcontroller
2. GND is a ground pin. Attach to ground on the microcontroller.
3. TRIG should be attached to a GPIO pin that can be set to HIGH
4. ECHO is a little more difficult. The HC-SR04 outputs 5v, which could destroy many microcontroller GPIO pins (the maximum allowed voltage varies). In order to step down the voltage use a single resistor or a voltage divider circuit. Once again this depends on the specific microcontroller you are using, you will need to find out its GPIO maximum voltage and make sure you are below that.

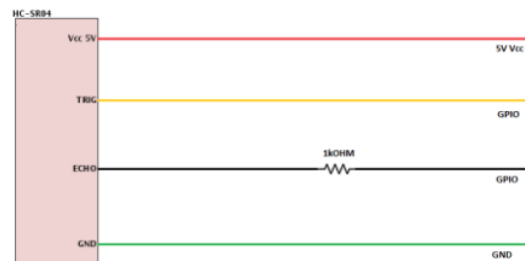


Figure 4

5. Errors and Bad Readings

Ultrasonic sensors are great sensors -- they work well for many applications where other types of sensors fall short. Unfortunately, they do have weaknesses. These weaknesses can be mitigated and worked around, but first they must be understood. The

first weakness is that they use sound. There is a limit to how fast ultrasonic sensors can get distance measurements. The longer the distance, the slower they are at reporting the distance. The second weakness comes from the way sound bounces off of objects. In enclosed spaces it is possible, if not probable that there will be unintended echos. The echos can very easily cause false short readings. In Figure 2 a pulse was sent out. It bounced off of object 1 and returned to the sensor. The distance was recorded and then a new pulse was sent. There was another object farther away, so that when the new pulse reaches object 1, the first signal will reach the sensor. This will cause the sensor to think that there is an object closer than is actually true. The old pulse is smaller than the new pulse because it has grown weaker. The longer the pulse exists the weaker it grows until it is negligible. If multiple sensors are being used, the number of echos will increase along with the number of errors. There are two main ways to reduce the number of errors. The first is to provide shielding around the sensor. This prevents echos coming in from angle outside what the sensor should actually pick up. The second is to reduce the frequency at which pulses are sent out. This gives more time for the echos to dissipate.

Anexo 2: Operacionalización de variables

TÍTULO	FORMULACIÓN DEL PROBLEMA	HIPÓTESIS	OBJETIVOS	VARIABLES	DIMENSIONES	METODOLOGÍA
Influencia de un código de programación en Arduino para la detección de objetos en bastón para invidentes en el distrito de Moche-2023.	<p>Problema General:</p> <p>¿Cómo influye un código de programación en Arduino para la detección de objetos en un bastón para invidentes?</p>	<p>Hipótesis General:</p> <p>Un código de programación en Arduino influye significativamente para la detección de objetos de un bastón para invidentes.</p>	<p>Objetivo General:</p> <p>Evaluar la influencia de un código de programación en Arduino para la detección de objetos de un bastón para invidentes.</p>	<p>Variable Independiente:</p> <p>Código de programación</p>	<p>Lógica de Control (I)</p> <p>Software de programación (I)</p>	<p>Tipo: Aplicada</p> <p>Diseño: Cuantitativa</p> <p>Población y muestra: Todos los códigos de programación en Arduino utilizados en detección de objetos para bastón blanco y el código de programación en Arduino.</p>
	<p>Problemas Específicos:</p> <p>¿Cuáles son las características del modelo del sensor ultrasónico?</p> <p>¿Cómo se realizará la implementación de la programación en el módulo de detección de objetos en un bastón para invidentes?</p>		<p>Objetivos Específicos:</p> <p>Programar el sensor ultrasónico en Arduino.</p> <p>Implementación de la programación en módulo de detección de objetos de un bastón para invidentes.</p>	<p>Variable Dependiente:</p> <p>Detección de objetos</p>		

Anexo 3: Matriz de consistencia

34

Variable	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores	Instrumentos	Escala de Medición
Código de Programación	Conjunto de instrucciones que ordenan al Arduino ejecutar. Dicho código está estructurado según las guías correspondientes a un lenguaje de programación específico. (León Mexa, 2021)	<p>16 La traducción del lenguaje de programación a las instrucciones binarias se realizará utilizando el Arduino Uno R3</p>	Lógica de control	Diagrama de flujo	Software de Arduino	Bloque de flujo
			Software de programación	Arduino IDE	Hardware de Arduino	Bits
Variable Dependiente						
Detección de objetos	<p>17 Es un sensor HC-SR04 de distancia que utiliza el ultrasonido para determinar la distancia de un objeto en un rango de 2 a 450 cm. (Martínez Fuentes, 2014)</p>	<p>El emisor (TRIG) emite pulsos de ultrasonido (kHz) que rebotan en el objeto detectado, el receptor (ECO) cambia a (5V) por el tiempo equivalente del pulso y esto es interpretado por el microcontrolador.</p>	Funcionamiento de ultrasonido HC SR04	Ultrasonido HC SR04	Registro de distancia de objetos	Centímetros

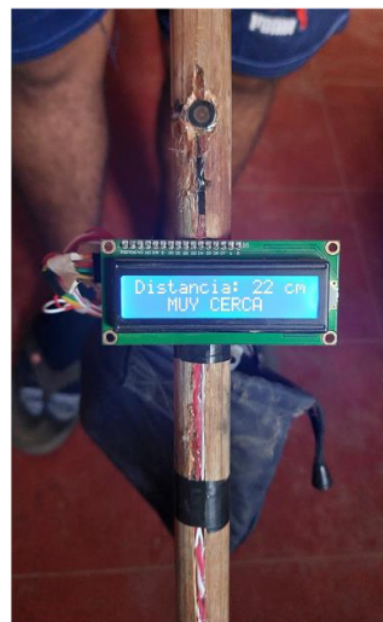
Anexo 4: Fotos

9

Sensor HC – SR04 vs Vereda



Sensor HC – SR04 vs Puerta



9

Sensor HC – SR04 vs Poste



Sensor HC – SR04 vs Mueble



9

Sensor HC – SR04 vs Cesto



Sensor HC – SR04 vs Balón



INFORME DE TESIS - QUISPE & BRINGAS

INFORME DE ORIGINALIDAD

13%

INDICE DE SIMILITUD

12%

FUENTES DE INTERNET

1%

PUBLICACIONES

8%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Catolica de Trujillo Trabajo del estudiante	2%
2	core.ac.uk Fuente de Internet	1%
3	Submitted to Mountain Lakes High School Trabajo del estudiante	1%
4	repositorio.unsm.edu.pe Fuente de Internet	1%
5	descubrearduino.com Fuente de Internet	<1%
6	Submitted to IUBH - Internationale Hochschule Bad Honnef-Bonn Trabajo del estudiante	<1%
7	Submitted to Universidad Autónoma Metropolitana-Xochimilco Trabajo del estudiante	<1%
8	alicia.concytec.gob.pe Fuente de Internet	<1%

9	docplayer.es Fuente de Internet	<1 %
10	www.scielo.org.co Fuente de Internet	<1 %
11	Submitted to Universidad Tecnológica del Peru Trabajo del estudiante	<1 %
12	cybertesis.unmsm.edu.pe Fuente de Internet	<1 %
13	ri-ng.uaq.mx Fuente de Internet	<1 %
14	laccei.org Fuente de Internet	<1 %
15	tesis.pucp.edu.pe Fuente de Internet	<1 %
16	piperlab.es Fuente de Internet	<1 %
17	www.tecnologiaypedagogia.net Fuente de Internet	<1 %
18	Submitted to Universidad Tecnológica Indoamerica Trabajo del estudiante	<1 %
19	www.repositorio.usac.edu.gt Fuente de Internet	<1 %

20	www.slideshare.net Fuente de Internet	<1 %
21	repository.usta.edu.co Fuente de Internet	<1 %
22	rinacional.tecnm.mx Fuente de Internet	<1 %
23	repositorio.uct.edu.pe:8080 Fuente de Internet	<1 %
24	repositorio.untrm.edu.pe Fuente de Internet	<1 %
25	unexcai11-101.blogspot.com Fuente de Internet	<1 %
26	www.habilidadesparaadolescentes.com Fuente de Internet	<1 %
27	sol.sbc.org.br Fuente de Internet	<1 %
28	www.colibri.udelar.edu.uy Fuente de Internet	<1 %
29	Submitted to Universidad Industrial de Santander UIS Trabajo del estudiante	<1 %
30	repositorio.unesum.edu.ec Fuente de Internet	<1 %
31	www.ambito.com	

Fuente de Internet

<1 %

32

www.element14.com

Fuente de Internet

<1 %

33

repository.unad.edu.co

Fuente de Internet

<1 %

34

es.slideshare.net

Fuente de Internet

<1 %

35

publicaciones.uni.edu.py

Fuente de Internet

<1 %

36

repositorio.espe.edu.ec

Fuente de Internet

<1 %

37

www.cyberprimo.com

Fuente de Internet

<1 %

38

Submitted to Universidad Católica de Santa
María

Trabajo del estudiante

<1 %

39

onlinelibrary.wiley.com

Fuente de Internet

<1 %

40

rehatrans.com

Fuente de Internet

<1 %

41

repositorio.uan.edu.co

Fuente de Internet

<1 %

42

repositorio.urp.edu.pe

Fuente de Internet

<1 %

43

repositorio.uta.edu.ec

Fuente de Internet

<1 %

44

e-catalog.nlb.by

Fuente de Internet

<1 %

45

repositorio.umsa.bo

Fuente de Internet

<1 %

46

www.clubensayos.com

Fuente de Internet

<1 %

47

de.slideshare.net

Fuente de Internet

<1 %

48

dspace.ups.edu.ec

Fuente de Internet

<1 %

49

risti.xyz

Fuente de Internet

<1 %

50

Submitted to uncedu

Trabajo del estudiante

<1 %

Excluir citas

Activo

Excluir coincidencias < 10 words

Excluir bibliografía

Activo