



TECNOLOGICO DE ESTUDIOS SUPERIORES DE ECATEPEC

DIVISION DE INGENIERIA EN SISTEMAS COMPUTACIONALES

**Sistema de navegación autónoma de interiores con
tecnología LIDAR 2D**

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIERO EN SISTEMAS COMPUTACIONALES

P R E S E N T A N:

Berenice Concepción Velasco Reyes

Omar Hernández Ramírez

DIRECTOR: Griselda Barrera Cortes

Codirector: Francisco Jacob Ávila Camacho

Codirector: Edgar Corona Organiche

ECATEPEC DE MORELOS, EDO. DE MÉXICO, 2022.



GOBIERNO DEL
ESTADO DE MÉXICO

DICTAMEN DE LIBERACIÓN Y CUMPLIMIENTO DE
REQUISITOS ACADÉMICOS PARA TITULACIÓN

FO-TESE-DA-78
TÍTOS



DIRECCIÓN ACADÉMICA
DIVISIÓN INGENIERÍA EN SISTEMAS COMPUTACIONALES

TES_PRF-20222-008
09 DE NOVIEMBRE DEL 2022

Datos generales

Nombre del egresado:

VELASCO	REYES	BERENICE CONCEPCIÓN
Apellido paterno	Apellido materno	Nombres (s)

Matrícula: 201321284 Teléfono: 55-5749-7890 Móvil: 55-5878-8665 Correo Electrónico: Bvreyes08@gmail.com 201321284@tese.edu.mx

Opción de titulación: **TESIS PROFESIONAL**
Únicamente en las opciones I, II, VII y X

PROYECTO: **SISTEMA DE NAVEGACIÓN AUTÓNOMA DE INTERIORES CON TECNOLOGÍA LIDAR 2D**

ASESOR (ES): **DRA. GRISELDA CORTES BARRERA, DR. FRANCISCO JACOB ÁVILA CAMACHO, M. EN C. EDGAR CORONA ORGANICHE**

Dictamen:

Con base en la solicitud de opción de titulación No. (1) correspondiente al periodo escolar (2022 - 2), presentada por el alumno de referencia en fecha **21 DE FEBRERO DEL 2022** y después de haber sido desarrolladas las actividades requeridas en dicha opción, el Consejo Académico Divisional, conformado por los integrantes que firman al calce, ha determinado que SI (X) NO () han sido cubiertos en su totalidad los requisitos que permitan a esta División Académica liberar al alumno de los trámites académicos para tal efecto e iniciar el trámite de titulación en la Unidad de Registro y Certificación de este Tecnológico de Estudios Superiores de Ecatepec.

Observaciones:

Consejo Académico Divisional

M. EN C. IRVING CARDIEL ALCOZAR GUILLERMO Presidente de la Academia de Ciencias Básicas	LIC. MARÍA REBECA GARCÍA PONCE Presidenta de la Academia de Ciencias Sociales
M. EN D. DULCE ANAHI FLORES NUÑEZ Presidenta de la Academia de Ciencias Económico y Administrativas	M. EN C. ROSA MARÍA HERNÁNDEZ ROJAS Presidenta de la Academia de Ciencias de la Ingeniería
I.S.C. FRANCISCO JAVIER HERNÁNDEZ BARAJAS Presidente de la Academia de Diseño de la Ingeniería	M. EN C. XÓCHITL RAQUEL WONG COHÉN Presidenta de la Academia de Ingeniería Aplicada

Elaboró:	Revisó y Visto Bueno:
M. EN C. MERCEDES FLORES FLORES SECRETARIO DEL CONSEJO ACADÉMICO DIVISIONAL	ING. JOSÉ LUIS CHÁVEZ ROJAS ENCARGADO DE LA DIVISIÓN ISC



GOBIERNO DEL
ESTADO DE MÉXICO

DICTAMEN DE LIBERACIÓN Y CUMPLIMIENTO DE
REQUISITOS ACADÉMICOS PARA TITULACIÓN

FO-TESE-DA-78
TÍTOS



DIRECCIÓN ACADÉMICA
DIVISIÓN INGENIERÍA EN SISTEMAS COMPUTACIONALES

TES_PRF-20222-007
09 DE NOVIEMBRE DEL 2022

Datos generales

Nombre del egresado:

HERNÁNDEZ RAMÍREZ OMAR
Apellido paterno Apellido materno Nombres (s)

Matrícula: 201410864 Teléfono: 55-5749-7890 Móvil: 55-6745-7093 Correo Electrónico: omar-hr@outlook.com 201410864@tese.edu.mx

Opción de titulación: **TESIS PROFESIONAL**

Únicamente en las opciones I, II, VII y X

PROYECTO: **SISTEMA DE NAVEGACIÓN AUTÓNOMA DE INTERIORES CON TECNOLOGÍA LIDAR 2D**

ASESOR (ES): **DRA. GRISELDA CORTES BARRERA, DR. FRANCISCO JACOB ÁVILA CAMACHO, M. EN C. EDGAR CORONA ORGANICHE**

Dictamen:


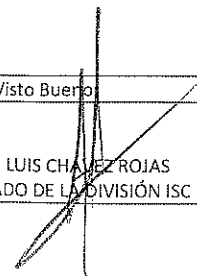
Con base en la solicitud de opción de titulación No. (1) correspondiente al periodo escolar (2022 - 2), presentada por el alumno de referencia en fecha **21 DE FEBRERO DEL 2022** y después de haber sido desarrolladas las actividades requeridas en dicha opción, el Consejo Académico Divisional, conformado por los integrantes que firman al calce, ha determinado que **SI (X) NO ()** han sido cubiertos en su totalidad los requisitos que permitan a esta División Académica liberar al alumno de los trámites académicos para tal efecto e iniciar el trámite de titulación en la Unidad de Registro y Certificación de este Tecnológico de Estudios Superiores de Ecatepec.

Observaciones:

Consejo Académico Divisional

<p> M. EN C. IRVING CARDIEL VALCOCER GUILLERMO Presidente de la Academia de Ciencias Básicas</p>	<p> LIC. MARÍA REBECA GARCÍA PONCE Presidenta de la Academia de Ciencias Sociales</p>
<p> M. EN D. DULCE ANAHÍ FLORES NUÑEZ Presidenta de la Academia de Ciencias Económicas y Administrativas</p>	<p> M. EN C. ROSA MARÍA HERNÁNDEZ ROJAS Presidenta de la Academia de Ciencias de la Ingeniería</p>
<p> I.S.C. FRANCISCO JAVIER HERNÁNDEZ BARAJAS Presidente de la Academia de Diseño de la Ingeniería</p>	<p> M. EN C. XOCHITL RAQUEL WONG COHÉN Presidenta de la Academia de Ingeniería Aplicada</p>

LICENCIATURA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

<p>Elaboró:</p> <p> M. EN C. MERCEDES FLORES FLORES SECRETARIO DEL CONSEJO ACADÉMICO DIVISIONAL</p>	<p>Revisó y Visto Bueno</p> <p> ING. JOSÉ LUIS CHÁVEZ ROJAS ENCARGADO DE LA DIVISIÓN ISC</p>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Dedicatoria

O.- Con mucho amor y cariño a mi hija Antonella que es mi mayor motivación y la razón de mi existencia. Si algún día lee esto, sabrá que todo fue siempre por ella.

A mis padres por el inmenso esfuerzo que han hecho para que yo salga adelante pese a todas las dificultades y hostilidades de la vida, ustedes están siempre en mi corazón y sin duda alguna este éxito es más suyo que mío.

A Berenice que es mi cómplice y compañera de vida; por su apoyo incondicional e invaluable y sobre todo por tanto amor que ha hecho todo esto posible.

B.-Mi bella Antonella eres la única razón de cada día, de cada acción siempre serás la más grande prueba de que con el 1% basta para que todo sea posible.

Tres personas especiales, siempre fueron, son y serán la mayor inspiración y ejemplo de persistencia, resistencia, resiliencia y fortaleza ustedes sembraron esta semilla que hoy junto a nuestra promesa damos por lograda.

Mamá este y todos los logros son tanto tuyos como míos pues eres la única persona que siempre ha creído en mí, gracias por siempre obligarme a dar más y jamás dejarme a pesar de los años te amo.

Omar por todo el esfuerzo que en conjunto emos puesto en cada paso que damos, por la paciencia, la tolerancia y el amor tan inmenso que tenemos que siempre sean la base para todo aquello que cosechemos mano a mano.

Agradecimientos

O.-El principal agradecimiento es a Dios porque de alguna u otra manera siempre colocó en mi andar todos los medios necesarios para poder llegar hasta este punto.

A mis padres que con mucho amor, paciencia y esfuerzo me educaron y me pusieron las herramientas necesarias para salir adelante en un mundo donde las exigencias son cada vez más grandes.

A mi esposa que nunca dejó de creer en mí y que es un motor fundamental que siempre supo y sabrá sacar la mejor versión de mí.

Y a todos mis amigos, maestros y personas que en su momento me brindaron su apoyo a lo largo de toda esta carrera.

B.-A mis Profesores en especial a la Mtra. Griselda Cortes por todo su apoyo a lo largo de este proyecto, el interés y la paciencia y sobre todo comprensión en todo momento.

A mi Mamá por todo el impulso que me brindo en cada etapa de mi formación su apoyo y su confianza y creer siempre en mí, a mi Papá por enseñarme que aunque no tengas el apoyo siempre podrás, a mis hermanos por siempre brindarme su apoyo incondicional

A mis amigos que a lo largo de nuestra carrera siempre estuvieron conmigo trabajando en equipo y apoyándonos siempre en todo momento.

A mi compañero de vida Omar que desde nuestro inicio ha sido mi cómplice en cada aventura, meta, paso y objetivos, mi mayor respaldo por todo lo vivido y lo que está por venir gracias.

Resumen

El avance tecnológico crece día con día en el mundo y en cada país surgen nuevas herramientas creadas para facilitar actividades cotidianas de los seres humanos o bien dar respuesta y soluciones a cosas que antes parecían no tenerlas o era sumamente complicado realizar, nos ayuda a llegar a lugares donde es imposible para un humano, a explorar más allá de lo conocido.

“La historia de los robots comenzó mucho antes de lo que se suele pensar; desde la Grecia antigua, los más reconocidos filósofos empezaron a asomar la posibilidad de crear artefactos o artilugios que sustituyeran el trabajo humano, específicamente en aquellas actividades relacionadas con la limpieza de fincas y el cultivo de los alimentos.”

La robótica y automatización han tenido sus inicios desde mucho antes que la revolución industrial con las maquinas que se crearon para la industria, pero fue en ese punto donde se dio un gran avance para el desarrollo de esta área y con el paso del tiempo se ha extendido más a cada parte de nuestra vida como trabajo, escuela hogar, transporte etc. Es por ello, por lo que el presente proyecto se desarrolla un módulo movimiento autónomo implementado mediante tecnología, el LIDAR fin de esto es que este pueda ser implementado o agregado a otras herramientas que con ello cumplan una función mayor y con esto agilizar su operatividad.

El involucrarnos en esta parte nos abre una gran oportunidad al aporte y avance de tecnologías en nuestro país ya que aunque en México existen ya aplicaciones y creaciones sobre el tema contamos con un considerable rezago tecnológico en comparación con otros países y mayormente se opta por importar estas tecnologías del extranjero, también buscamos demostrar que el desarrollar

tecnología puede hacerse de manera accesible utilizando componentes que están al alcance de muchos y nos pueden brindar los requerimientos necesarios en este caso trabajamos principalmente con los siguiente elementos:

Arduino UNO, Controlador puente H L298N, Sensor laser LIDAR lite v3 e. Se desarrollo un prototipo de robot, y la implementación de un algoritmo el cual su objetivo es darte a este robot la autonomía de operar su libre movimiento sin la intervención humana, apoyados de el sensor LIDAR que utiliza un haz de luz pulsado que se puede denominar un simple láser con una baja potencia. La principal función es poder medir la distancia que existe entre él y el objeto en cuestión al poder medir el tiempo que transcurre entre la emisión del haz de luz y la detección de la señal que se refleja sobre este.

Índice

Contenido

Dedicatoria.....	2
Capítulo I.....	9
Introducción.....	9
1.1 Antecedentes.....	11
1.2 Planteamiento del Problema.....	14
1.3 Objetivo General.....	15
1.4 Objetivos Específicos.....	15
1.5 Justificación.....	16
1.6 Hipótesis.....	18
1.7 Aporte.....	18
Marco Teórico.....	19
2.2 Arquitectura de un robot móvil.....	21
2.3 Requerimientos y Configuración para Robots Móviles Autónomos.....	22
2.4 Sistema empotrado Arduino.....	23
2.6 Entradas y salidas analógicas.....	26
2.7 Motores y librerías de servos.....	26
2.8 Tecnología de robots.....	27
2.9 Sistema de engranes en robótica.....	28
2.10 Servomotores de transmisión directa.....	28
2.12 Sensores e instrumentos.....	29
2.13 Sensor de luz light detection and ranging.....	29
2.14 Técnicas de simultaneous localization and mapping.....	30
CAPITULO III.....	31
Sistema de navegación autónoma de interiores con tecnología LIDAR 2D.....	31
3.1 Requerimientos y necesidades del prototipo utilizando técnicas de SLAM localización y mapeo simultaneo.....	32
3.1.1 Especificaciones técnicas del sensor light detection and ranging.....	34
3.1.2 Especificaciones Técnicas Arduino.....	35
3.2 Algoritmo de toma de decisiones de desplazamiento.....	36
. 3.3 Diseño y construcción del prototipo físico del Robot.....	38

3.4 Integración del Software programado a la estructura del prototipo.....	39
3.5 Integración del Software programado a la estructura de prototipo.....	41
Capitulo IV.....	44
Resultados	44
4.1 Uso esperado del prototipo.....	45
4.2 Presentación de estructura	45
4.3 Datos obtenidos del sensor	46
4.4 Resultados de la evasión de obstáculos con el prototipo	47
4.5 Pruebas de funcionalidad en un ambiente con obstáculos	47
Conclusiones y trabajo futuro	52
Trabajo futuro	54

Índice de Figuras

<i>Fig 1</i> Metodología propuesta	32
<i>Fig 2</i> Arduino Uno	33
<i>Fig 3</i> Lidar 3V	33
<i>Fig 4</i> Bloque 1 del diagrama de proceso de autonomía	37
<i>Fig 5</i> Bloque 2 Diagrama de proceso de autonomía	38
<i>Fig 6</i> Montaje de componentes	39
<i>Fig 7</i> Diagrama de Conexión	jError! Marcador no definido.
<i>Fig 8</i> Prototipo final	46
<i>Fig 9</i> Proceso de recopilación de datos	46
<i>Tabla 1</i> Especificaciones LIDAR	35
<i>Tabla 2</i> Especificaciones Arduino	36
<i>Tabla 3</i> Impactos por circuito de ejecución	48
<i>Tabla 4</i> Impactos calibrando el servomotor	50
<i>Ilustración 1</i> Gráfica de impactos por circuito	49
<i>Ilustración 2</i> , Espectro de visión del ojo humano	49
<i>Ilustración 3</i> Gráfica de comportamiento del servomotor	51

Capítulo I

Introducción

Se presentan las bases para realizar el proyecto, se definen los objetivos, estado del arte de los trabajos, se delimita el área de investigación previa junto al planteamiento del problema.

Un LIDAR (acrónimo del inglés LIDAR, Light Detection and Ranging o Laser Imaging Detection and Ranging) es un dispositivo que permite determinar la distancia desde un emisor láser a un objeto o superficie utilizando un haz láser pulsado. La distancia al objeto se determina midiendo el tiempo de retraso entre la emisión del pulso y su detección a través de la señal reflejada. En general, la tecnología LIDAR tiene aplicaciones en geología, sismología y física de la atmósfera. También se investiga su uso en vehículos, especialmente los autónomos.

Dadas las múltiples aplicaciones que se le pueden dar a esta clase de sensores, se pretende desarrollar el módulo de autonomía para un robot que tendrá aplicaciones de sanitización, utilizando sensores LIDAR que le permitan ubicarse espacialmente en un entorno y que haga recorridos a la vez que genera un mapa de la zona.

En este caso particular utilizamos un sensor LIDAR de la marca GARMIN modelo LITE v3, para realizar pruebas en un modelo a escala reducida que simulará los eventos que tendrá que generar el robot original.

La encomienda es que este robot pueda realizar una ruta si chocar con ningún objeto, a la vez que genera un mapa y una base de conocimientos sobre el lugar para futuras trayectorias y desplazamientos de desinfección.

El adicionar módulos que adicionen y agilicen tareas a herramientas ya existentes siempre brindará una mayor eficacia y eficiencia a sus aplicaciones, con ello dará una mejor inversión y rentabilidad, el crear un módulo de movimiento autónomo independiente y adaptable garantiza la seguridad del uso de otros sistemas en este caso el Robot Sanitizador no pondrá en riesgo al ser humano por exposición a la tecnología desinfectante que utiliza ya que al tomar sus propias decisiones de desplazamiento no ocupa ser operado ni monitoreado lo cual también repercute en un buen ahorro de recurso monetarios y humanos, estos mismos beneficios pueden ser empleados en otras áreas y ramas de la industria.

1.1 Antecedentes

En 2017 J. Dominic realizó un trabajo sobre “El control de un vehículo autónomo” utilizando como herramientas: encoders, LIDAR IMUs. En donde su principal objetivo de este proyecto fue el estudio y desarrollo de la navegación, localización y algoritmos de evitación de obstáculos para un robot móvil. Para detectar el entorno y los obstáculos el sensor LIDAR (detección de luz y rango) se utilizará como sistema de percepción principal. Para lograr una conducción completamente autónoma, se deben combinar varios subsistemas, como los sistemas de percepción, un sistema de localización y sistemas de navegación y planificación de trayectorias. Además de los algoritmos utilizados, la precisión de la localización depende de la precisión de los sensores.

Finalmente en este proyecto lo que se busca es adquirir conocimientos en el campo de la navegación autónoma sistemas y los algoritmos y métodos de localización utilizados en robótica móvil (Dominik, 2018).

En el proyecto titulado “Un método para controlar un robot y / o un sistema de transporte autónomo sin conductor”, desarrollado por L. R. Miller en 2019 nos habla sobre la base de una identificación de objetos basada en sensores que incluye la generación de un par de puntos, características de los contornos de la superficie 2D sobre la base de los contornos de la superficie 2D de los objetos a identificar.

Este proyecto fue realizado mediante las herramientas de Raspberry PI y sensor LIDAR; con estas se adquiere una nube de puntos del entorno utilizando el sensor de distancia, se estima una normal de superficie para cada punto y se generan las características de pares de puntos correspondientes del entorno.

Como resultado en base a un método de votación, las características del entorno se comparan con las características del modelo para generar de manera eficiente una hipótesis de pose, que se somete a una optimización y una verificación de

coherencia para finalmente ser aceptadas o rechazadas como una adquisición (Miller, 2017).

Por otra parte, en el proyecto que lleva por nombre “Método para construir un mapa mientras se realiza un trabajo en segundo plano” que fue propuesto en 2019 por K. J. George nos menciona que para que los dispositivos robóticos autónomos o semiautónomos funcionen de forma independiente o con una entrada mínima y / o control externo dentro de un entorno, es útil un proceso de mapeo para la construcción autónoma de un mapa del entorno de trabajo.

Lo cual se logra cuando se proporciona un proceso ejecutado por un robot, que incluye: atravesar, a una primera posición, una primera distancia en dirección hacia atrás; después de recorrer la primera distancia, girar 180 grados en una primera rotación; después de la primera rotación, atravesando, a una segunda posición, una segunda distancia en la segunda dirección; y después de atravesar la segunda distancia, girar 180 grados en una segunda rotación de modo que el campo de visión del sensor apunte en la primera dirección; todo esto se logró a partir de la utilización principalmente de Arduino uno, sensor LIDAR (George, 2019).

El presente desarrollo de A. A. Ebrahimi que lleva por nombre “Sistema de mapeo y seguimiento con características en espacio tridimensional” fue realizado en el año 2019 se refiere y tiene como base métodos y herramientas como: Sensor LIDAR, métodos LK-SURF, Robust Kalman Filter, HAR-SLAM y Landmark Promotion SLAM methods. LK-SURF, que se basan en una técnica de procesamiento de imágenes que combina el seguimiento de características de Lucas-Kanade con características robustas aceleradas para realizar un seguimiento espacial y temporal utilizando imágenes estéreo para producir características 3D que se pueden rastrear e identificar.

The Robust Kalman Filter es una extensión del algoritmo del filtro de Kalman que mejora la capacidad de eliminar observaciones erróneas mediante el análisis de componentes principales y la regla de rechazo de valores atípicos X84.

Hierarchical Active Ripple SLAM es una nueva arquitectura SLAM que rompe el espacio de estado tradicional de SLAM en una cadena de espacios de estado más pequeños, lo que permite que se produzcan múltiples objetos rastreados, múltiples sensores y múltiples actualizaciones en tiempo lineal con almacenamiento lineal con respecto al número de objetos rastreados, puntos de referencia y ubicaciones estimadas de objetos (Ebrahimi, 2019).

Continuando con los precursores de esta rama tenemos el trabajo de H. Huang que lleva por nombre “Un robot modular autónomo” el cual fue desarrollado en 2019 en el cual se incluye un sistema de retención de accesorios que retiene dos o más accesorios intercambiables para realizar tareas únicas, por ejemplo, limpieza a vapor, aspirar, cortar césped, etc. Los accesorios pueden colocarse secuencialmente en la trayectoria de desplazamiento del robot y configurarse para realizar tareas complementarias.

El robot comprende un cuerpo principal; un sistema de impulsión unido al cuerpo principal; un sistema de retención de archivos adjuntos; y un sistema de control. El sistema de transmisión está configurado para mover el cuerpo principal. El sistema de retención de accesorios está configurado para acoplar dos o más accesorios intercambiables al cuerpo principal, y cada uno de los accesorios intercambiables puede configurarse para realizar una tarea. El sistema de control puede configurarse para controlar de forma autónoma el sistema de accionamiento y los accesorios intercambiables (Hong, 2020).

R. Greggs desarrollo en 2020 el proyecto “Un método para crear un mapa basado en láser 3D” que se lleva a cabo con: GPS, sensor LIDAR y métodos de mapeo en el cual se incluye: planificar una ruta y adquirir datos en un proceso de movimiento de un robot móvil; cargar datos adquiridos utilizando una herramienta de creación de mapas; realizar la alineación inicial con los datos láser adyacentes; realizar una alineación precisa con los datos láser adyacentes, realizar una asociación de bucle cerrado a la misma área atravesada por el robot móvil y establecer una restricción de bucle cerrado; realizar una solución de optimización global para un mapa completo para el que se establece la restricción de bucle

cerrado; y superponer todos los datos del láser y generar un mapa láser. El método no depende del GPS y no está restringido por señales de GPS (Gregg, 2019).

El siguiente artículo llamado “Robots Móviles autónomos y plataformas de navegación” escrito en 2020 por K. J. Z Cruz, toma como objetivo principal: Desarrollar una vigilancia tecnológica sobre literatura existente de robots móviles autónomos, plataformas de navegación utilizados en procesos logísticos.

Nos muestra el siguiente desenlace; los usos de la robótica estaban enfocados inicialmente en el sector industrial, donde el principal objetivo era la automatización masiva en los servicios para aumentar la productividad, la flexibilidad, la calidad, pero, sobre todo, la mejora de la seguridad para disminuir el riesgo de las personas en la realización de tareas peligrosas.

En áreas como: salud, educación, movilidad, electrónica, mecanizado, plásticos, mobiliario y equipamiento, hacen parte de los más beneficiados con la aplicación de la robótica en los diferentes procesos y tareas que desarrollan. Sin embargo, con el paso de los años se ha fortalecido también a la industria agrícola (Cruz, 2020).

1.2 Planteamiento del Problema

Existen herramientas que nos facilitan muchas tareas en todo aspecto de nuestra vida, cada vez más complementarlas se vuelve parte fundamental para que estas puedan brindar mayor eficacia, un ejemplo claro de esto puede ser el IOT (Internet de las cosas), los carros que implementan sistemas inteligentes para localización y estacionado automático, y es en este punto donde encontramos una ventana de oportunidad para dar un plus a otras herramientas y cumplan su función con mejor eficacia y menos recursos, tomando como herramienta previa un Robot sanitizador cuya función era la de desinfectar áreas mediante rayos ultravioleta los cuales exponen al humano y no puede operarse manualmente más que vía

remota pero esto puede resultar una tarea difícil y pesada ya que si el espacio es grande puede haber problemas de cálculo, pérdida de señal, etc.

Es así como nace la necesidad de implementar a esta herramienta un módulo de autonomía la cual permita que mediante un algoritmo este pueda realizar sus cálculos de espacio y ubicación y tomar propiamente la decisión sobre su movimiento, con ello reduce sus tiempos de operación, pero siendo este un módulo independiente tiene la capacidad de adaptarse a cualquier otra herramienta y brindarle un plus que magnifique su uso como ejemplo, aspiradoras, podadoras, pequeños vehículos, pero también pueden transportarse por lugares donde sea difícil el acceso y tránsito al humano como para exploraciones, zonas de desastres y como en el ámbito científico se han desarrollado robots que pueden ser lanzados a altas profundidades del mar, espacio y otros sitios de poca seguridad.

Con la actual situación este tipo de tecnología se ha implementado importantemente en el área de entregas de paquetes empresas, como un ejemplo de esto denota la empresa de Amazon quien opto por la entrega de paquetes vía dron que, aunque parecieran diferentes a nuestro modulo no es así, y (Ripipsa, 2020) a que estas ocupan tecnología de navegación y ubicación misma que se implementa a menor escala con nuestro proyecto.

1.3 Objetivo General

Diseñar y construir un prototipo robótico autónomo basado en el uso de LIDAR que opere de manera independiente que sea adaptable a diferentes ambientes.

1.4 Objetivos Específicos.

- Identificar requerimientos y necesidades del prototipo utilizando técnicas de SLAM localización y mapeo simultaneo
- Crear el algoritmo de toma de decisiones de desplazamiento del prototipo autónomo.
- Diseñar y construir prototipo físico del Robot móvil
- c
- Implementar y probar el prototipo de movimiento autónomo.

1.5 Justificación

La automatización puede minimizar los requisitos de mano de obra o reemplazar completamente a los humanos para tareas tediosas. Pero principalmente, puedes llevar a tu equipo de trabajo a ser más eficiente, productivo, y a que dedique más tiempo a tareas que generan valor.

Arduino y LIDAR son las herramientas que emplearemos para la realización del módulo autónomo, aunque existen componentes de mayor sofisticación y usos estos son los que presentan mayor accesibilidad y pueden cumplir con la función que se tiene asignada, los datos que obtenidos por el sensor de luz son recabados por la placa en la cual se tiene cargado el programa del algoritmo de toma de decisiones en donde en conjuntos con los servos y motores se da el movimiento a muestra estructura.

A menudo se comienza a pequeña escala, pero la implementación crece a medida que se vuelve más fácil la operación de distintas herramientas. Por ejemplo, la mayoría de las plantas de fabricación utilizan robótica automatizada para fabricar sus productos. El aporte humano es solo para definir el proceso y supervisarlos. Al mismo tiempo, las líneas de ensamblaje robóticas hacen todo el ensamblaje por

sí mismas, con materias primas convertidas en productos terminados automáticamente a un ritmo rápido.

La automatización se está volviendo más popular en todos los sectores, incluso se han incorporado técnicas de inteligencia artificial y aprendizaje automático en software / hardware y máquinas de nivel inferior para maximizar sus usos, es por ello por lo que innovar y explotar en esta rama es no solo necesario sino conveniente en cualquier ámbito.

Los efectos sociales del impacto de nuestro proyecto causarían un gran aporte a la incentivación de la creación de herramientas tecnológicas, pues esta nos abre una oportunidad laboral y profesional, dentro de los efectos negativos encontramos que esta herramienta al poderse implementar sobre otras que operan regularmente las personas puede causar que algunos empleos ya no sean requeridos, pero a su vez también exigiría la creación de nuevas áreas para el mantenimiento y supervisión de la implementación de esta nueva tecnología. En el área económica, los efectos que se tendrían en las empresas en las que sus servicios ocupen herramientas que pueden utilizar la implementación de este módulo, podrían tener una baja significativa en sus gastos administrativos y operativos por ejemplo en una empresa de sanitización ya que se reducirían sus tiempos de operación, así como también la mano de obra que se involucra en dichos servicios, entre las otras áreas en las que podemos adaptar el módulo como pueden ser, servicios de limpieza, recolección entre otros. Los efectos en el área tecnológica serían sumamente positivos pues incentivar, participar e innovar en esta área siempre significaría un gran avance en el desarrollo de un país para que este pueda tener competitividad y con ello requerir más profesionistas y generar más empleos y mejores oportunidades de vida.

1.6 Hipótesis

Los sistemas de navegación autónoma de interiores con tecnología LIDAR 2D permiten que se creen herramientas tecnológicas que controlen su propio movimiento en espacios determinados, de esta manera se agilizan y reducen sus tiempos de operación

1.7 Aporte

Sentar un precedente al desarrollo de nuevas tecnologías consideramos es sumamente importante para el desarrollo comercial y de crecimiento de un país aprovechar las herramientas locales en vez de las extranjeras y con ello crear nuevos mercados, áreas y mejores oportunidades para nuestra comunidad.

La adaptación al mundo tecnológico de otras herramientas puede traer consigo una gran ganancia y una buena distribución de los recursos ya que esta idea puede escalarse y ser implementada en áreas como: seguridad, el área de limpieza y desinfección, el área de la salud, para realizar traslados, búsquedas entre muchas otras.

Capítulo II

Marco Teórico

Presentamos el desarrollo del marco teórico basados en una cuidadosa selección de los autores y trabajos previos, bases en los cuales retomamos cada uno de los componentes que pueden servirnos para la realización del prototipo, se retoman temas como movimientos, tipos de sensores, placas Arduino, métodos y librerías que se pueden implementar.

2.1 Robots teledirigidos y autónomos

Se define como robots autónomos a los que tienen la aplicabilidad y capacidad de poder ejecutar actividades y tareas sin la necesidad de algún tipo de comando y control directamente explícito de los humanos. Los Robots autónomos son funcionales en lugares como: industrias y comercios, además de áreas de trabajo tan diferentes como: el agua, el aire y el espacio. Estos se han evolucionado y perfeccionados gracias al desarrollo de la inteligencia artificial pues les han permitido ser capaces de realizar diversas actividades en entornos cooperativos que pueden ser desde complejos y dinámicamente cambiantes (Ripipsa, 2020).

En diferencia un robot tele operado o bien teledirigido ocupa la intervención de operación por un usuario a distancia, mediante conexiones inalámbricas de distintos tipos, el mayor ejemplo de esto son los robots espaciales, pues estos son controlados vía remota desde la estación espacial. En ambos casos son herramientas de diferentes tipos y diseños las cuales tienen su mayor uso en la industria, para completar tareas que son difíciles o peligrosas para los seres humanos, también para llegar a sitios donde los humanos aún no han podido, pero aun con todo esto sus usos son limitados y no tan conocidos aun en las grandes economías. (Torres, 2017)

Los orígenes de la robótica se pueden remontar a mucho tiempo atrás ya que la necesidad de crear herramientas que nos faciliten el realizar una acción o bien evitar el peligro de que alguna actividad implique a sido desde el comienzo de la humanidad, los primeros prototipos robóticos creados fueron los tipos brazo que se utilizaron para manipular material radioactivo por los años 50 (Ruiz-de-Garibay, 2017).

2.2 Arquitectura de un robot móvil

La arquitectura de un robot móvil se da a partir en diferentes campos ya que es un sistema que opera de manera conjunta con otros subsistemas los principales son: Sistema cognitivo, sistema electrónico y sistema mecánico, a partir de estos en cada caso de creación y diseño de prototipos robóticos se emplean los materiales según sean la exigencias y necesidades de las actividades y tareas a cumplir de cada uno; desde su forma más básica hasta la más compleja siempre veremos esos tres principales sistemas operado en conjunto para lograr su fin y cumplir con la definición de robot (Murlanda, 2018).

Entendamos como *sistema cognitivo* el algoritmo o aplicación programada dentro del procesador que es el encargado de la toma de decisiones es aquí donde se procesan los datos recabados cualesquiera que sean, esto también involucra las decisiones que debe ejecutar el sistema mecánico el cual es el encargado de realizar las funciones de movimiento y desplazamiento, acompañado del sistema eléctrico que es quien surte la energía y distribuye la misma para cada acción y tarea a ejecutarse, y aunque dependiendo de cada prototipo pueden agregarse más subsistemas para sus funciones especializadas todos parten de estos tres quienes comprenden la esencia para cualquier herramienta robótica y a partir de esto se estructura la arquitectura. *“La arquitectura debe ofrecer los medios por los cuales el sistema pueda lograr sus metas eficientemente, satisfacer restricciones de tiempo real, promover la tolerancia a las fallas y proporcionar seguridad para vehículo y su entorno”* (Fillottrani, 2018)

Arquitectura de control la cual está estructurada por; capacidad de abordar múltiples objetivos de forma simultánea, capacidad de integración de la información de múltiples sensores de diferente procedencia, robustez ante fallos en elementos del sistema de adaptación ante nuevos entornos, capacidad de extensión y modificación a lo largo de su vida, capacidad para considerar posibilidades y valorar consecuencias interacción con el entorno para percibir cambios en el y responder adecuadamente. Elementos básicos y áreas

Percepción, interpretación e integración sensorial, modelado del mundo, reconocimiento planificación a nivel de tarea, gestión de recursos, monitorización de la actividad, navegación, planificación detallada de movimientos y acciones control de actuadores (Lope, 2019).

2.3 Requerimientos y Configuración para Robots Móviles Autónomos

Los requerimientos para un robot siempre serán diferentes según las funciones y objetivos para los cuales este diseñado por ejemplo no ocuparíamos los mismos requerimientos para un submarino tele controlado que para un robot industrial, aunque ambos tienen las mismas bases y métodos para definir sus requerimientos que se realizaran siempre de una manera jerárquica y en diferentes campos como lo son los requerimientos funcionales los cuales se indican por; ser capaz de moverse por un entorno controlado mostrando diferentes grados de “inteligencia”, teniendo un entorno relativamente controlado donde se minimizan las interferencias que pudieran anular, mermar o engañar seriamente los sensores, disponer de locomoción, disponer de algún mecanismo que permita al autómatas desplazarse por el medio, capacidad de percepción (Onofre, 2021).

Dentro de los requerimientos físicos no podemos olvidar que el crear un robot, tiene unas necesidades de materiales dispositivos, cada uno con de ellos con una serie de requisitos de funcionamiento, por ejemplo: necesitamos un ordenador dentro del robot, pero para que este funcione necesita energía estos son los elementos que ocupan que estos requerimientos sean cumplidos; Movimiento y Sentido: el robot necesita unos dispositivos para moverse y otros para recibir información del medio, estos dispositivos por desgracia no son fáciles de conectar

al ordenador directamente y necesitan de cierta adaptación tanto eléctrica como lógica (Onofre, 2021).

Para ello se ha creado la interfaz de control, que no solo controla, sino que también alimenta y adapta electrónicamente los sensores y dispositivos de actuación. Interfaz de control: Necesita una fuente de energía para poder funcionar Esta fuente debe estar separada de la alimentación de los motores dado que la electrónica de control es muy sensible al ruido electrónico (Morales, 2007).

2.4 Sistema empotrado Arduino

Como uno de los sistemas de la arquitectura de un prototipo robótico nos indica debemos tener la parte cognitiva, lógica que es donde se procesan los datos y se toman las decisiones según el algoritmo previamente diseñado y programado lo marque, pero le corresponde a una computadora llevar este proceso para este caso nos apoyamos de la plataforma Arduino que se podría definir como una microcomputadora es una placa con un microcontrolador. El microcontrolador de Arduino posee lo que se llama una interfaz de entrada, que es una conexión en la que podemos conectar en la placa diferentes tipos de periféricos. La información de estos periféricos que conectes se trasladará al microcontrolador, el cual se encargará de procesar los datos que le lleguen a través de ellos (Serma, 2019).

El tipo de periféricos que puedas utilizar para enviar datos al microcontrolador depende en gran medida de qué uso le estés pensando dar. Pueden ser cámaras para obtener imágenes, teclados para introducir datos, o diferentes tipos de sensores. También cuenta con una interfaz de salida, que es la que se encarga de llevar la información que se ha procesado en el Arduino a otros periféricos. Estos periféricos pueden ser pantallas o altavoces en los que reproducir los datos procesados, pero también pueden ser otras placas o controladores. (Reyes & Cid, 2016)

Arduino nació en el año 2005 en el instituto de diseño interactivo de ivrea (Italia), apareció por la necesidad de contar con un dispositivo para utilizar en las aulas,

al día de hoy es un proyecto de hardware y software libre, su entorno de programación es multiplataforma se puede ejecutar e instalar en diferentes sistemas operativos. El modelo del microcontrolador que lleva la placa es el modelo ATmega328p de la marca Atmel. La “P” del final significa que este chip incorpora tecnología picopower, que permite un consumo eléctrico menor, tiene una arquitectura tipo AVR (Orrente, 2015).

2.5 Arquitectura abierta del sistema Arduino

Arduino es una plataforma de desarrollo basada en una placa electrónica de hardware libre que incorpora un microcontrolador re-programable y una serie de pines hembra. Estos permiten establecer conexiones entre el microcontrolador y los diferentes sensores y actuadores de una manera muy sencilla (principalmente con cables dupont). Arduino es una plataforma de hardware libre, se pueden crear y modificar sus propias placas a partir de una misma base y puede darle el uso que quiera libremente, de manera que con una de estas placas se pueden crear diferentes tipos de microordenadores. Y lo mismo pasa con el software libre que ofrece la plataforma, un entorno donde cada persona puede programar y crear las aplicaciones que quiere poner en su placa Arduino mediante un lenguaje de programación propio (Baturone, 2001).

2.5.1 Funciones y Características de los pins

GPIO significa entradas y salidas de propósito general puede que a un ingeniero le diga algo, pero la verdad, a mi esta descripción no me dice absolutamente nada. Al tratarse de circuitos eléctricos lo más lógico es pensar que sea controlando la corriente y la tensión. Y precisamente, esto es lo que hacemos con Arduino, permitir o no permitir circular la corriente. Como si fuera un interruptor por ejemplo, un pin digital de Arduino es útil cuando quieres encender un LED al apretar

un pulsador. Arduino Nano es una placa de desarrollo de tamaño compacto, completa y compatible con protoboards, basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital (de los cuales 6 pueden ser usando con PWM), 6 entradas analógicas, un cristal de 16Mhz, conexión Mini-USB, terminales para conexión ICSP y un botón de reseteo, posee las mismas capacidades que un Arduino UNO (Luis de Valdes Hernandez, 2019).

2.5.2 Entradas y salidas digitales

Las entradas y salidas digitales (nivel lógico) son los bloques más básicos de los sistemas de hardware digitales construcción. También se les conoce comúnmente como GPIO. Una señal de nivel lógico tiene dos estados: baja y alta, respectivamente, 0 y 1 o falsas y verdaderas. Las representaciones reales de estos estados en términos de voltaje y corriente varían en función de los circuitos que los generan, a su vez, influyen en los circuitos que los utilizan (MIOIO OPENSAT, 2018).

2.5.3 Salida digital

Cuando se utiliza como salida, en sus modos de funcionamiento por defecto, los pines MIOIO representan los niveles lógicos de la siguiente manera : un nivel lógico BAJO hará que el pin para forzar una tensión de 0 V (" tierra") en lo que se conecta a la misma, un nivel lógico ALTO hará que el pin para forzar una tensión de 3,3 V en lo que se conecta a la misma. (Peña, 2021)

2.5.4 Entrada digital

Cuando se utilizan los pines MIOIO como entradas, detectan el nivel de tensión y reportan su estado en consecuencia.

Cuando se detecta 0 V , el pin lee BAJO, cuando se detecta 3.3V, el pin a leer ALTO pins 5V- tolerante, toleran hasta 5V, y tratarlo como ALTO, cuando no hay nada conectado en el pin y nada le obliga a un voltaje, los pines pueden dar cualquier valor y no se puede confiar en el. Esto se conoce como modo " flotante" , y es el modo por defecto para los pines de entrada (Peña, 2021).

2.6 Entradas y salidas analógicas

Una señal analógica es un tipo de señal que cambia sus valores de forma continua a través del tiempo. Por norma general en los autómatas las entradas analógicas son más escasas, más lentas y caras que las entradas digitales. En el caso de Arduino uno disponemos de un número variable de entradas analógicas, que en el caso de Arduino Uno y Mini Pro son 6, y en el Arduino Mega son 16. Este es un número más que respetable de entradas analógicas, que rivaliza o supera a autómatas tradicionales de coste muy superior (Llamas, 2020).

2.7 Motores y librerías de servos

Un servomotor es un tipo especial de motor que permite controlar la posición del eje en un momento dado. Está diseñado para moverse determinada cantidad de grados y luego mantenerse fijo en una posición. Para la construcción de robots se utilizan motores paso a paso y/o servomotores. Estos se caracterizan por ser permitir un movimiento controlado y por entregar un mayor par de torsión (torque)

que un motor DC común. En este artículo daremos un vistazo a lo que son los servomotores, sus usos y principales características y haremos algunas pruebas experimentales sobre consumo de energía. Un servomotor (o servo) es un tipo especial de motor con características especiales de control de posición. Al hablar de un servomotor se hace referencia a un sistema compuesto por componentes electromecánicos y electrónicos (Gonzalez, 2016).

2.8 Tecnología de robots

La robótica es una de las expresiones de la tecnología cuya aplicación se ha extendido a diversos contextos de la vida del hombre. Además de sus diversas aplicaciones en la industria, hace presencia facilitando y mejorando actividades como: los vuelos no tripulados, el estudio del mundo submarino, la limpieza de piscinas, la exploración del espacio exterior con robots como el Opportunity, el Spirit, el Rocky IV, la Misión Robótica Juno y el Curiosity; en el ámbito del entretenimiento: la creación de robots, como Aibo de Sony que simulan características de una mascota, robots que pueden jugar fútbol, robots móviles, humanoides y muchos otros en los cuales se aplican los últimos adelantos tecnológicos en sonido, reconocimiento y síntesis de voz, e inteligencia artificial en pocas palabras un robot es un sistema autónomo, en forma de dispositivo, que interactúa con su entorno. La estructura de un robot es el conjunto de elementos mecánicos que le dan forma y soportan los demás elementos que lo componen. (Márquez, 2019)

Los robots pueden mostrar muchos tipos estructuras dependiendo del fin para el que estén diseñados: brazo robótico articulado, vehículo para desplazamiento de cargas, paneles solares que siguen automáticamente al Sol, etc. Pero todos ellos están formados fundamentalmente por los siguientes componentes: sensores, sistemas de control, actuadores, herramientas y elementos que suministran energía (Ramírez, 2016).

2.9 Sistema de engranes en robótica

Los engranes son ruedas dentadas de distintos tamaños que encajan entre sí, y a través de este complemento mecánico ocurre la transmisión de movimiento hacia el resto de la maquinaria. Partes de un engrane Diente: Es aquel que efectúa el esfuerzo de empuje y transmite la potencia, tienen un perfil que se debe tener en cuenta en su diseño y fabricación. Corona: Es la parte donde se encuentran los dientes. Cubo: Parte céntrica mediante la cual la rueda del engranaje queda fijada a su eje. Cabeza o Cresta. Parte superior del diente. Cara. Parte superior de cada extremo lateral de cada diente. Flanco: Es la cara interior del diente, es la zona donde ocurre el rozamiento entre los engranajes. Fondo o valle. Espacio que separa un diente del otro (Santiago, 2019).

2.10 Servomotores de transmisión directa

El término «transmisión directa» se puede aplicar a cualquier motor que impulsa directamente una carga o un rotor sin elementos de transmisión como engranajes, poleas o cadenas. Mas a menudo el término hace referencia a motores sin escobillas, de imán permanente o síncrono que transmiten el par directamente a la carga o el rotor. Los motores de transmisión directa funcionan prácticamente igual que la mayoría de los motores CC sin escobillas. Los imanes están fijados al rotor del motor y los bobinados están colocados en el estátor del motor. Puesto que los bobinados reciben alimentación producen campos electromagnéticos que, o bien atraen, o repelen los imanes del rotor. Un sistema adecuado de interruptores o conmutación de la alimentación de los bobinados provoca un movimiento controlado. Existen motores de transmisión directa lineales y rotatorios, pero las versiones rotatorias son con diferencia las más usadas (Celera Motion, 2019).

2.12 Sensores e instrumentos

Existen diferentes tipos de instrumentos de medición que se encargan de capturar los datos de diferentes variables para poder ejercer una supervisión constante. Por ejemplo, la medición de la calidad del aire, del agua, de voltaje, movimiento y luz, son algunas de las que más requieren de un constante monitoreo para: la seguridad del personal, la optimización de recursos y materias primas, el control de los procesos, puesta en marcha o interrupción de acciones según los datos monitoreados, ahora, los instrumentos de medición deben estar acompañados de un sistema que permita recopilar los datos medidos, almacenarlos y analizarlos para desarrollar acciones predeterminadas y automatizadas, o la toma de decisiones por parte del personal a cargo del monitoreo. Este monitoreo se hace sencillo cuando cuentas con proveedores de soluciones especializadas, en las que se incluyen los siguientes elementos: Sensores que se encargan de recopilar los datos medidos y de transmitirlos por medio de la red (de internet o radiofrecuencia) para ser almacenados y procesados y sistema o software en el que se almacena y presenta la información, para que el personal encargado del monitoreo lo pueda analizar (SENSOR GO, 2021).

2.13 Sensor de luz light detection and ranging

Light Detection And Ranging (LIDAR) es una tecnología de teledetección basada en el láser. La idea que subyace en LIDAR es bastante sencilla: apuntar un pequeño láser a una superficie y medir el tiempo que tarda el láser en volver a su fuente. Esta tecnología se utiliza en los sistemas de información geográfica (SIG) para producir un modelo digital de elevación (DEM) o un modelo digital del terreno (DTM) para la cartografía en 3D. LIDAR principio es bastante fácil de entender: 1 emite un pulso de láser en una superficie 2. atrapa el láser reflejado de vuelta a la LIDAR fuente del pulso con sensores 3 mide el tiempo que el láser viajó 4

calcula la distancia de la fuente con la fórmula "Distancia = (Velocidad de la luz x Tiempo transcurrido) / 2, este proceso es repetido un millón de veces por los instrumentos LIDAR y termina produciendo un complejo mapa del área estudiada conocido como una nube de puntos 3D (YELLOWSCAN 2020).

2.14 Técnicas de simultaneous localization and mapping

SLAM proviene de las siglas inglesas Simultaneous Localization And Mapping, mapeo y localización simultáneos. Se trata de una técnica de navegación que permite a un robot o vehículo, construir un mapa del entorno y al mismo tiempo navegar por él, gracias a la información que captan sus sensores en tiempo real. El desplazamiento autónomo de una máquina a través de un entorno desordenado siguiendo una ruta óptima desde una posición inicial hasta una posición final. Algo que las personas vemos tan sencillo y natural como es el caminar por un lugar, para una máquina, es una de las tareas más complejas y exigentes que puede acometer. Las principales cosas que un robot debe tener en cuenta a la hora de navegar son: Detectar el entorno y los obstáculos de este, colocar en un mapa la información capturada, planificar una ruta o trayectoria libre de colisiones, navegar por la ruta elegida teniendo en cuenta posibles cambios en la posición de los obstáculos o la aparición de nuevos (José Luis Martínez, 2020)

CAPITULO III

Sistema de navegación autónoma de interiores con tecnología LIDAR 2D

Se presenta el método y herramienta elegidas para la realización del prototipo para el módulo de movimiento autónomo podemos ver a detalle cada uno de los componentes la manera en que se conectan, la función y el porqué de cada pieza, el algoritmo de decisión es programado en el IDE de Arduino, a partir de ahí se realizan las pruebas de funcionamiento y se realizan los ajustes.

La Fig 1 Metodología propuesta muestra cuatro etapas de desarrollo a) análisis, se basó sobre los principios ya existentes acerca del tema para elegir el diseño del prototipo, algoritmo y materiales a utilizar, la creación del algoritmo se basó en los principios de SLAM, b) En la etapa de desarrollo estructural se integran las herramientas Arduino, sensor LIDAR, servomotores, montajes y circuitos, el algoritmo es programado en código Arduino y montado sobre la placa) la fase de pruebas verifica como principalmente la conectividad entre el LIDAR y Arduino con la recolección de datos d) en la última etapa se presenta la documentación del proyecto, así como el prototipo final.

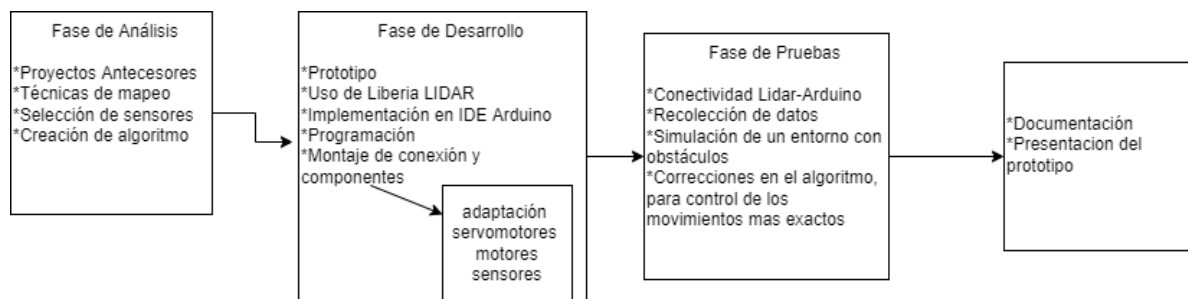


Fig 1 Metodología propuesta

3.1 Requerimientos y necesidades del prototipo utilizando técnicas de SLAM localización y mapeo simultaneo

Basados en el objetivo general del módulo la selección de materiales se realizó en base a un examen de comparativas en donde el costo/utilidad y la compatibilidad fueron los puntos a considerar para la elección de los componentes.

En este caso particular utilizamos un sensor LIDAR de la marca GARMIN modelo LITE v3, para realizar pruebas en un modelo a escala reducida que simulará los eventos que tendrá que generar el robot original.

Para la implementación del prototipo a escala se utilizaron los siguientes materiales: 1 chasis para robot móvil a tres ruedas, dos moto-reductores DC a 12V, dos llantas para moto-reductores, una rueda loca de 30 mm de diámetro aproximadamente, 1 capacitor de 470 μ F a 16 V, 2 baterías recargables modelo 18650 de 3.7v 6000 mA/h, un porta pila doble para baterías 18650 y 1 controlador puente H L298N.



Fig 2 Arduino Uno

La Fig 2 es una placa basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital (de los cuales 6 pueden ser usando con PWM), 6 entradas analógicas, un cristal de 16Mhz, conexión USB, conector jack de alimentación, terminales para conexión ICSP y un botón de reseteo. Tiene toda la electrónica necesaria para que el microcontrolador opere, simplemente hay que conectarlo a la energía por el puerto USB ó con un transformador AC-DC, dentro del proyecto es la base principal ya que es el microcontrolador que posee el algoritmo programado en donde son procesados los datos y ejecutadas las instrucciones del mismo, tiene conectados los sensores, motores y alimentación de energía.



Fig 3 Lidar 3V

La Fig 3 es un sensor de proximidad, alcance fiable y potente para aplicaciones de drones, robots o vehículos no tripulados. Compacto, ligero y con bajo consumo de energía, el usuario puede ajustar la exactitud, el alcance de funcionamiento y el tiempo de medición cuenta con comunicaciones vía I2C y PWM, requiere una fuente de alimentación y un microcontrolador externo para ejecutar una aplicación cuando los requisitos de espacio y peso son ajustados, es la solución de detección y medición de distancia óptica compacta y de alto rendimiento ideal para aplicaciones de drones, robots o vehículos no tripulados. Con una solución de procesamiento de señal de chip único y un mínimo de hardware, este sensor es altamente configurable puede utilizarse como el bloque de construcción básico para aplicaciones donde los espacios reducidos, la ligereza, el bajo consumo de energía y alto rendimiento son factores importantes. La encomienda es que este robot pueda realizar una ruta si chocar con ningún objeto, a la vez que genera un mapa y una base de conocimientos sobre el lugar para futuras trayectorias y desplazamientos de desinfección.

3.1.1 Especificaciones técnicas del sensor light detection and ranging

En la presente tabla se especifica toda la información técnica de nuestro sensor principal, el cual hace parte principal de este proyecto, se detallan cada una de sus características y componentes, es importante explicarlos por qué a partir de estas especificaciones se decidieron los complementos como, batería para alimentar todo el prototipo, tamaños y espacios de los demás componentes y algunas funciones programadas en el algoritmo.

Tabla 1 Especificaciones LIDA (Russo, 2021).

Tamaño de la unidad (alto x ancho x profundidad):	20 x 48 x 40 mm
Peso:	22 g
Resolución:	1 cm
Precisión:	+/- 2,5 cm a una distancia superior a 1 m.
Rango:	0-40 m
Frecuencia de actualización:	hasta 500 Hz
Interfaz:	I2C o PWM
Potencia (voltaje de funcionamiento):	4,75-5,5 V CC; 6 V máx.
Consumo de corriente:	<2 ma en inactividad; <100 mA en funcionamiento continuo
Temperatura de funcionamiento:	-20 a 60 °C
Longitud de onda láser/potencia máxima:	905 nm/1,3 vatios
Divergencia del haz:	4 m radián x 2 m radián
Abertura óptica:	12,5 mm
Aprobaciones regulatorias:	Producto láser de clase 1 con clasificación en/iec 60825-1 2007 eL sistema contiene componentes que no pueden ser reutilizadas.

3.1.2 Especificaciones Técnicas Arduino

En la siguiente tabla se muestran las especificaciones de nuestro segundo componente principal la placa de Arduino, este es el cerebro del prototipo ya que por medio de él se llevó a cabo el procesamiento de datos recopilados por el sensor LIDAR, aquí se almaceno el código programado para cumplir con las sentencias del movimiento, aquí se conectaron todos los elementos del sistema de energía, sistema mecánico y el lógico.

Tabla 2 Especificaciones Arduino (Artero, 2018).

Microcontrolador	ATMega328P
Velocidad de reloj	16 MHz
Voltaje de trabajo	5V
Voltaje de entrada	7,5 a 12 voltios
Pinout	54 pines digitales (15 PWM) y 16 pines analógicos
Memoria	256 KB Flash (8KB para bootloader), 8KB RAM y 4KB Eeprom.

3.2 Algoritmo de toma de decisiones de desplazamiento.

En la Fig 4 Bloque 1 del diagrama de proceso de autonomía se muestra la primera fase del proceso que se siguió para llevar a cabo la construcción de este prototipo, como primer paso se realizó la selección de las herramientas lógicas para la programación como lo fueron las bibliotecas para el uso del LIDAR, se prosigue a montar la parte técnica de conexión de todos los elementos como lo son sensores, servomotores, sistema de energía, etc se muestra la secuencia a de talle de cómo se realizó la conexión de todos los elementos.

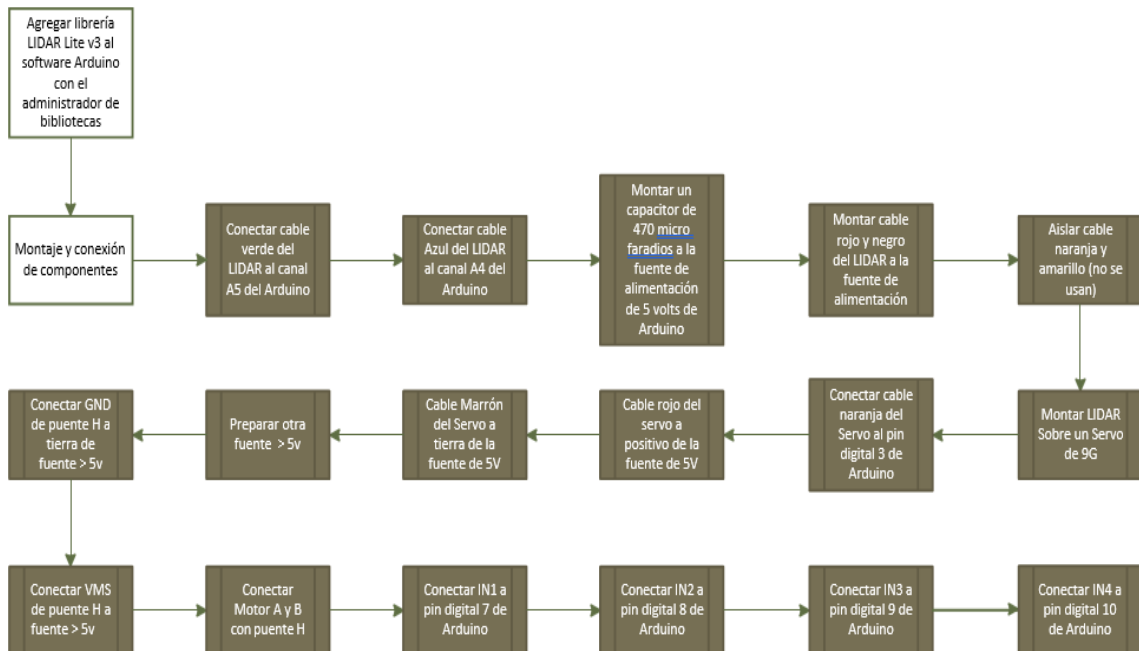


Fig 4Bloque 1 del diagrama de proceso de autonomía

En la segunda parte del diagrama de proceso el cual esta ilustrado en la

Fig 5Bloque 2 Diagrama de proceso de autonomía se declaran las clases para utilizar el servo motor y las definiciones de variables, como segunda sección están las banderas para verificar si existen obstáculos o el área es libre, junto con la variable del limite para el objeto y sensor LIDAR, 4 enteros son los que se encargaran de llevar el control del movimiento de los motores que están conectados a los pines del Arduino.

En la tercera etapa se crean los métodos para el avance hacia delante a través de un giro en espejo de las ruedas, en el giro para retroceder se copia la lógica del método de avanzar adelante, en un sentido inverso. Los giros de derecha se llevarán con los motores 1 y 2 aquí el giro se da en sentido opuesto y no en espejo como en los movimientos anteriores, de igual manera para el giro a la izquierda se invierten los pines, el diagrama está dividido en 2 etapas la lógica y la técnica en la cual nos centramos en la creación de métodos de movimientos y la conexión física de cada uno de los componentes electrónicos.

Métodos de movimiento

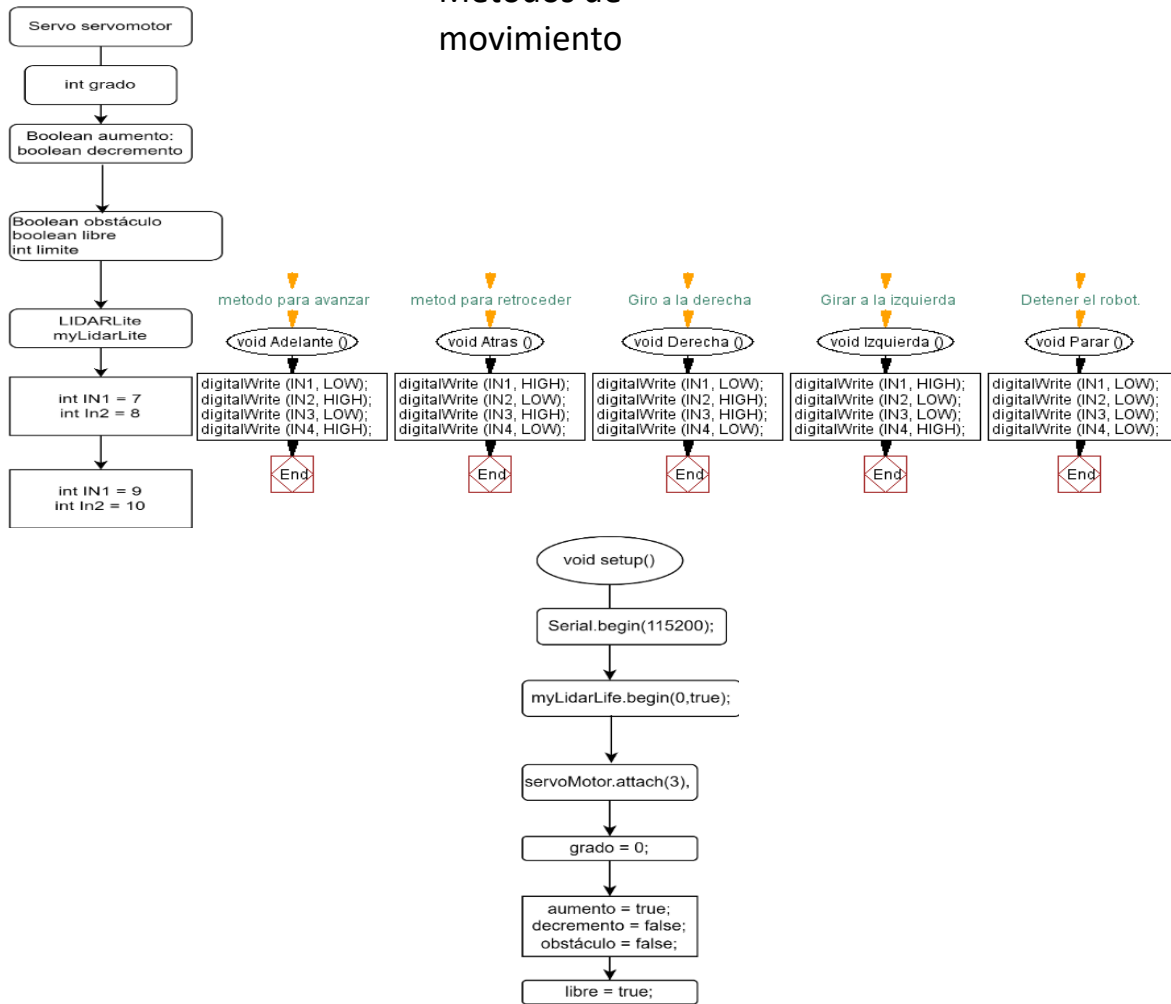


Fig 5Bloque 2 Diagrama de proceso de autonomía

3.3 Diseño y construcción del prototipo físico del Robot

Se utilizó una base rectangular como esqueleto y soporte el prototipo de material acrílico el cual nos brindará un peso ideal para poder ser soportado y llevado por los servomotores, cuatro llantas que se conectaron a cada servo con el fin de dar el equilibrio de un vehículo a tamaño escala, para el ensamblado se optó por poner el LIDAR sobre una altura más alta que el resto de los elementos del prototipo para que el punto de visión de un buen campo y la detección del haz de luz no se vea entorpecida y este pueda cumplir con su objetivo.

3.4 Integración del Software programado a la estructura del prototipo

En la Fig 6 se muestran las conexiones necesarias para el funcionamiento del robot. En esta arquitectura diseñada se dividen las conexiones en los siguientes tipos:

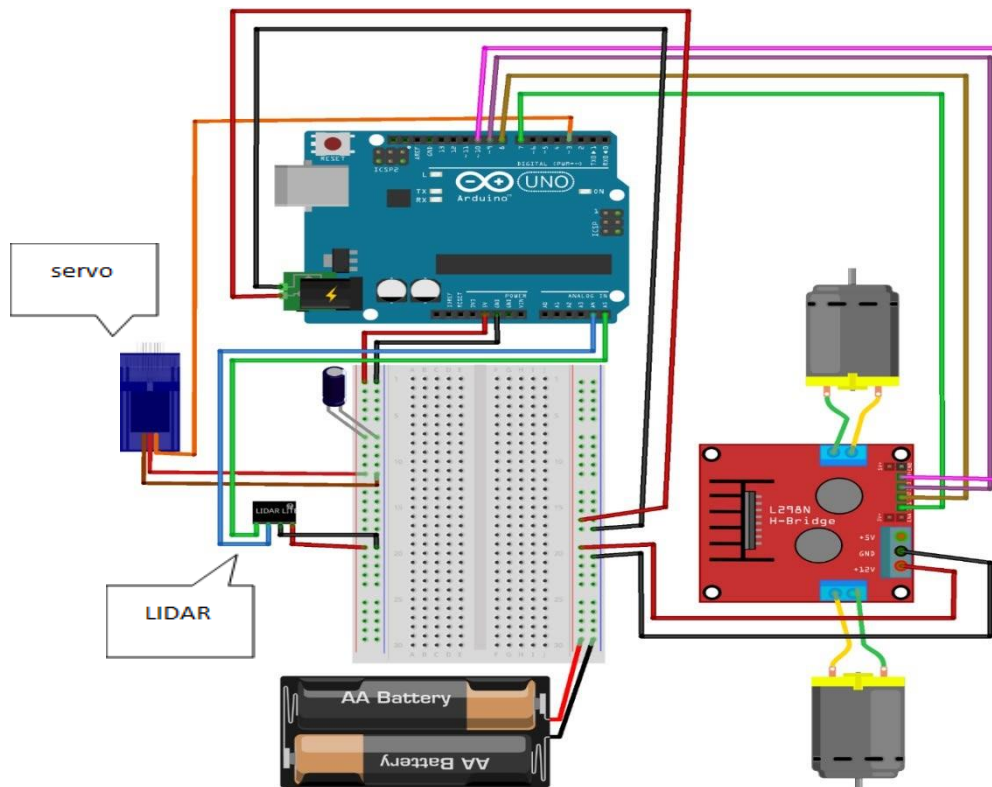


Fig 6 Diagrama de conexión

Conexiones de tipo lógica: son aquellas en las que se mapean las rutas que van a dirigir las señales que van a salir del Arduino hasta el actuador, en este caso las conexiones de tipo lógica están dadas y definidas por los pines digitales del Arduino, que en este caso se utilizaron los pines número 6, 7, 8 y 9 para enviar las señales al puente H en las entradas In1, In2, In3, In4 que serán las encargadas de mandar a su vez las señales a los motores.

Por otro lado, se puede apreciar que en el puerto 4 del Arduino también se tiene mapeado el controlador para un servomotor el cual se programará para que pueda

realizar un movimiento de 0 180 grados para realizar un barrido de la zona frontal en la que se colocará el sensor LIDAR.

Se emplean de igual modo las entradas y salidas análogas A1 y A2 para conectar los controladores del sensor LIDAR que mediante estos pines realizará el envío y recepción de datos capturados.

Puente H: El puente H recibe cuatro señales provenientes del Arduino mediante sus puertos IN1, IN2, IN3, IN4, para que de este modo pueda definir en sentido mandar las señales de giro, si en sentido horario o antihorario dependiendo de la polaridad que este reciba producto de las señales que emite el arduino.

A su vez el puente H manda señales a través de sus puertos M1 a M4 en los cuales fungen como salidas de energía que llevan las polaridades para girar en uno o en otro el sentido del motor.

Servo: Funciona como estructura de soporte para el LIDAR ya que este al realizar un giro en 0 a 180 grados, permite que el láser encargado de realizar el escaneo pueda obtener un ángulo de referencia óptimo que amplía su espectro de visión para la interpretación de objetos que obstaculicen su avance. El servo recibe una señal del pin 4 digital que le va indicando el momento en el que tiene que detenerse y cambiar ruta.

Sensor LIDAR El sensor LIDAR es un tipo de sensor basado en escaneo mediante Un lidar o lidar1 (acrónimo del inglés LIDAR, Light Detection and Ranging o Laser Imaging Detection and Ranging) es un dispositivo que permite determinar la distancia desde un emisor láser a un objeto o superficie utilizando un haz láser pulsado. La distancia al objeto se determina midiendo el tiempo de retraso entre la emisión del pulso y su detección a través de la señal reflejada. El LIDAR es un sistema que permite obtener una nube de puntos del terreno tomándolos mediante un escáner láser aerotransportado (airborne light scanner, ALS). Para realizar este escaneado se combinan dos movimientos. Uno longitudinal dado por la trayectoria del avión y otro transversal mediante un espejo móvil que desvía el haz de luz láser emitido por el escáner.

Para conocer las coordenadas de la nube de puntos se necesita la posición del sensor y el ángulo del espejo en cada momento. Para ello el sistema se apoya en un sistema GPS diferencial y un sistema de navegación inercial (INS). Conocidos estos datos y la distancia sensor-terreno obtenida con el distanciómetro obtenemos las coordenadas buscadas.

3.5 Integración del Software programado a la estructura de prototipo.

A continuación, se detalla con precisión en el siguiente código comentado la lógica implementada para esta primera fase de pruebas de movimiento utilizando el sensor LIDAR v3.

<pre> 1 #include <Wire.h> 2 #include <LIDARLite.h> 3 #include <Servo.h> 4 Servo servoMotor; 5 int grado; //valores de ida y vuelta del servo 6 boolean aumento; 7 boolean decremento; 8 boolean obstaculo; 9 boolean libre; 10 int limite; 11 LIDARLite myLidarLite; // Motor A 12 int IN1 = 7; 13 int IN2 = 8; // Motor B 14 int IN3 = 9; 15 int IN4 = 10; </pre>	<pre> //MÉTODO PARA AVANZAR 16 void Adelante () { //Direccion motor A 17 digitalWrite (IN1, LOW); 18 digitalWrite (IN2, HIGH); //Direccion motor B 19 digitalWrite (IN3, LOW); 20 digitalWrite (IN4, HIGH); } //Y ahora para moverse en el sentido contrario: 21 void Atras () { //Direccion motor A 22 digitalWrite (IN1, HIGH); 23 digitalWrite (IN2, LOW); //Direccion motor B 24 digitalWrite (IN3, HIGH); } </pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Línea 1-4 Importación de las librerías de los componentes externos a utilizar como lo son el LIDAR, servos y Ros que es el paquete encargado del movimiento
Línea 5-1.-En esta sección tenemos la declaración de variables que estaremos utilizando en los métodos entre las cuales están las de detección y grados.

Línea 16-20.- Método para avanzar en dirección continua se definen las variables que corresponden a los puertos que se van a emplear del Arduino, en los cuales van :

Puerto	Nombre
7	Entrada 1 del motor A
8	Entrada 2 del motor A
9	Entrada 1 del motor B
10	Entrada 2 del motor A

De este modo en los pines 7, 8, 9 y 10 del Arduino van a funcionar para emitir señales que van a hacer girar a los motores en un sentido determinado el objeto myLidarLite pertenece a la clase LidarLite la cual es una librería propia del fabricante que contiene los métodos necesarios para acceder a las funciones que provee el sensor Lidar.

Se definen los métodos para avanzar y retroceder. Los motores con esta combinación de bits giran en el mismo sentido para que las ruedas puedan ser impulsadas hacia adelante o hacia atrás invirtiendo el valor de las entradas.

<pre>// Giro a la derecha 26 void Derecha () { //Direccion motor A 27 digitalWrite (IN1, LOW); 28 digitalWrite (IN2, HIGH); //Direccion motor B 29 digitalWrite (IN3, HIGH); 30 digitalWrite (IN4, LOW); } //Girar a la izquierda 31 void Izquierda () { //Direccion motor A 32 digitalWrite (IN1, HIGH); 33 digitalWrite (IN2, LOW);</pre>	<pre>//Detener el robot. 36 void Parar () { //Direccion motor A 37 digitalWrite (IN1, LOW); 38 digitalWrite (IN2, LOW); //Direccion motor B 39 digitalWrite (IN3, LOW); 40 digitalWrite (IN4, LOW); } 41 void setup() { 42 Serial.begin(115200); 43 myLidarLite.begin(0, true); 44 myLidarLite.configure(3); 45 servoMotor.attach(3); 46 grado = 0; 47 aumento = true; 48 decremento = false;</pre>	<pre>49 obstaculo = false; 50 libre= true; 51 limite=20; 52 pinMode (IN1, OUTPUT); 53 pinMode (IN2, OUTPUT); 54 pinMode (IN3, OUTPUT); 55 pinMode (IN4, OUTPUT); }</pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Se definen los métodos que contienen las combinaciones necesarias para hacer que los motores giren en sentidos opuestos dando lugar a la rotación hacia la izquierda o derecha. Posteriormente se define el método para detener el robot. Cuando todas las entradas se establecen en Low, los motores se detienen. A partir de la línea 26 se presentan los métodos programados para el movimiento

de los motores quienes dan la dirección del camino para el robot, como son izquierda, derecha, detener.

<pre> 56 void loop() { 57 if (aumento) { 58 if (grado < 125) { 59 if(obstaculo){ 60 Izquierda(); 61 delay(100); 62 libre = true; 63 obstaculo = false; 64 } 65 if(libre){ 66 if(myLidarLite.distance()<limite){ 67 obstaculo = true; 68 Parar(); 69 delay(100); 70 }else{ 71 Adelante(); 72 } 73 } 74 } 75 grado =grado+3; 76 servoMotor.write(grado); 77 78 delay(1); 79 // 80 } else { 81 aumento = false; 82 decremento = true; 83 } 84 } </pre>	<pre> 85 if (decremento) { 86 if (grado > 65) { 87 if(obstaculo){ 88 Derecha(); 89 delay(100); 90 libre = true; 91 obstaculo = false; 92 } 93 if(libre){ 94 if(myLidarLite.distance()<limite){ 95 obstaculo = true; 96 Parar(); 97 delay(100); 98 }else{ 99 Adelante(); 100 } 101 } 102 grado = grado-3; 103 servoMotor.write(grado); 104 // 105 //Serial.println(myLidarLite.distance() 106); 107 //Serial.println(grado); 108 delay(1); 109 } else { 110 aumento = true; </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

línea 56 -96 como inicialmente el ángulo del servo inicia en 0, este va ir aumentando, hasta llegar a 125 grados, cuando el contador de los grados llega a este valor, se detiene y regresa, si en el lapso de que aumenta el contador de grados se detecta un obstáculo se gira a la izquierda durante 100 ms e intenta verificar si se libró del obstáculo. si durante el aumento de grados se encuentra libre, el robot avanza hasta encontrar un obstáculo. El contador de grados va a aumentar de 3 en tres para que sea tan lento el ciclo de escaneo una vez que el contador de grados llega a 125, aumento entra en falso y se activa el procedimiento de decremento.

Capitulo IV

Resultados

En este capítulo mostraremos los resultados que arrojaron las pruebas preparadas para medir el funcionamiento del prototipo realizado, para medir la eficiencia y utilidad del algoritmo creado para el módulo de autonomía, así como presentar la estructura final.

4.1 Uso esperado del prototipo

La encomienda es que este robot pueda realizar una ruta sin chocar con ningún objeto, a la vez que genera una base de conocimientos sobre el lugar para trazar futuras trayectorias y desplazamientos, el prototipo realizado logra cumplir con las tareas previstas. En los primeros recorridos de prueba arrojaba márgenes de error debido a los tiempos en que ejecutaba las instrucciones de movimiento, se realizaron los ajustes correspondientes, el módulo después de esto realiza la evasión de obstáculos de una manera óptima con la cercanía necesaria, también se provee que recuerde las rutas ya recorridas para evitar el duplicar caminos por lo que nos da el resultado esperado en cuanto a los cálculos previstos.

4.2 Presentación de estructura

El armado de prueba del prototipo para demostrar la funcionalidad del módulo de autonomía se muestra en la Fig 7 Prototipo final quedo a modo de un pequeño carro compuesto por tres llantas para lograr un buen equilibrio, aunque 4 llantas podrían dar una mejor estabilidad se opta por un sistema de triada ya que de esta manera se establece que los movimientos son exactos y precisos ya que el prototipo conserva un buen equilibrio y exactitud en los movimientos de esquivación de obstáculos, con lo que se puede comprobar que si es implementado en algún sistema de 4 ejes se cumplirá la función con gran éxito, la ubicación del LIDAR es punto clave para el buen funcionamiento ya que este debe tener un amplio campo de visión inferior y superior, para un buen campo lateral se incluyó un servo que fue programado para realizar un movimiento de lado a lado por un tiempo promedio de manera que no afecte la detección, es sistema que alimenta está integrado por una pequeña batería la cual soporta toda la carga, cabe mencionar que trasplantar este módulo a otras herramientas no debe generar el mayor problema pues la estructura del algoritmo y del código

fueran hechas para agregar más componentes, adaptarse o incluso funcionar con los LIDAR simultáneos si la necesidad lo amerita.

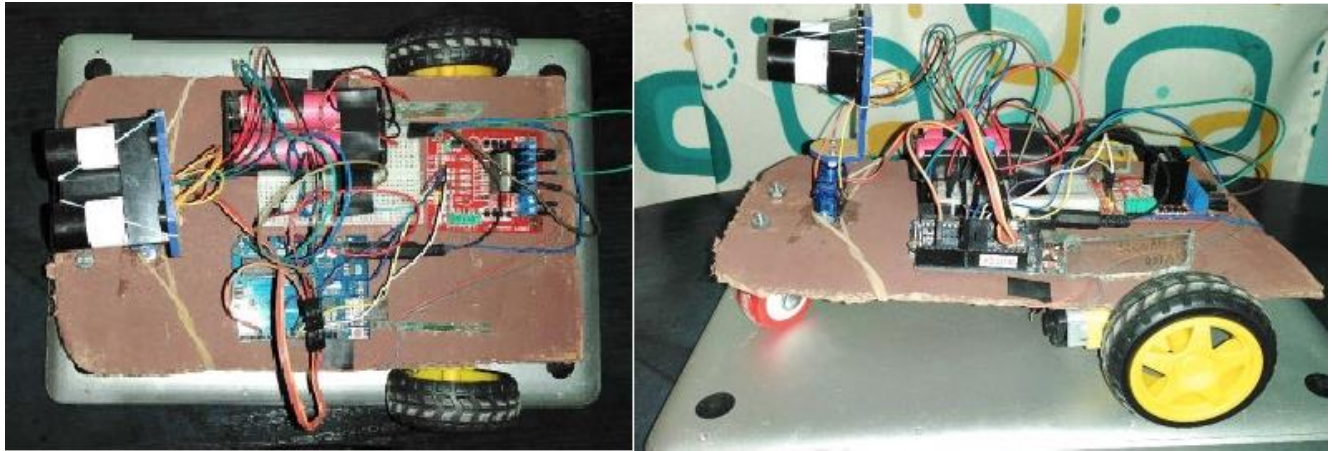


Fig 7 Prototipo final

4.3 Datos obtenidos del sensor

La placa de Arduino puede recopilar datos de los sensores, almacenarlos y enviarlos a otro dispositivo para su posterior análisis, para analizar datos con un Arduino se lleva a cabo un proceso serial que se compone de la siguiente manera
Proceso de recopilación de datos:

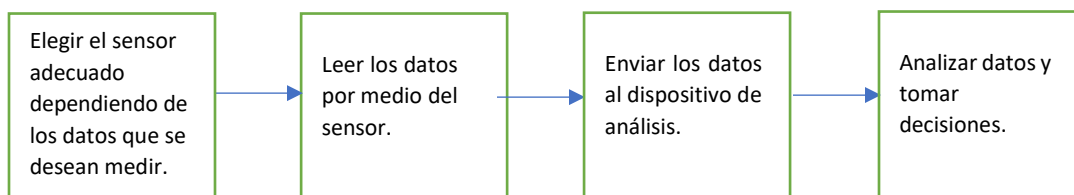


Fig 8 Proceso de recopilación de datos

Aplicado este proceso a nuestro prototipo el sensor LIDAR V3, recopila los datos por medio de la medición del haz de luz a un objeto es decir mide la distancia los tiempos en que esta es reflejada, estos tiempos los tomamos por medio del Arduino nuestro dispositivo de análisis mediante el programa que fue previamente

programado y cargado, aquí es donde a partir de estas recopilaciones se ejecutaban cada una de las sentencias del mismo y se producían los movimientos de acuerdo a la toma de decisiones. Los tiempos se midieron en distancias dadas en CM de acuerdo con el láser es decir entre más cerca de 0 estaba quería decir que el obstáculo se encontraba más cerca.

4.4 Resultados de la evasión de obstáculos con el prototipo

La evasión de obstáculos por parte del robot ocurre cuando el sensor LIDAR detecta que hay un obstáculo que impide su avance, y que además se encuentra dentro de la ruta generada por el comportamiento de navegación, recordemos que la detección se hace a través de los datos que arroja el haz de luz del LIDAR, cuando este choca mide la distancia y el tiempo en el cual la luz regresa, con esto el código puede medir que tan cerca y en qué dirección se encuentra este obstáculo, y cambiara la ruta hacia los lados, o en reversa este algoritmo tiene por fin el recordar las rutas ya recorridas y trazar nuevos caminos de escape, cabe mencionar que este proceso es realizado en segundos para que el movimiento sea lo mas fluido y no se noten paradas súbitas, o giros agresivos que desestabilicen el curso recto del prototipo.

4.5 Pruebas de funcionalidad en un ambiente con obstáculos

Se realizaron las pruebas con el prototipo creado a forma de vehículo, primero se realizaron en un espacio amplio de 4 paredes en el cual la navegación se dio en dirección rectilínea hasta que encontraba el tope de una de las paredes y cambiaba su dirección hacia donde tenía mayor campo de visión el sensor, es decir en primer punto fue de frente y giro a la izquierda, al encontrar nuevamente el muro el giro volvió a ser a la izquierda ya que al comparar las distancias entre ese punto y el de la izquierda tomo el que daba un mayor ámbito de visión, los tiempo de respuesta de los giros y velocidad de frenando se dio sin impacto

importante, se dirigió de manera fluida, se empezaron a agregar algunos obstáculos en las cuatro direcciones a partir de aquí empezamos a observar la eficiencia que tenía el vehículo en un entorno más saturado, la altura inicial del prototipo era de 20 Cm el LIDAR se ubicó inicialmente a 17 cm del suelo.

Algunas de las modificaciones que se hicieron fue subir el LIDAR a 20cm del suelo, ya que notamos que al tenerlo a 17 cm sobre la superficie del prototipo este tenía dificultades para captar algunos de los obstáculos que estaban a este nivel, y se realizó una tabla de medidas para obtener la eficiencia del LIDAR a la altura de 20 cm, como se muestra en la siguiente tabla:

Tabla 3 Impactos por circuito de ejecución

Número de ejecución	17 cm	18 cm	19 cm	20 cm
1	7	4	3	0
2	6	5	3	1
3	6	9	3	1
4	5	4	2	0
5	9	3	6	0
6	10	2	2	1
7	12	6	3	0
8	5	2	3	0
9	8	6	1	1
10	5	6	4	0
11	8	5	2	1
12	10	8	0	0
13	12	5	2	1
14	14	5	2	0
15	12	2	1	0
16	10	5	2	0
17	9	1	1	0
18	9	8	1	0
Proimedio	8.72	7.39	3.06	1.11

Por lo que al generar la gráfica de estado, se llegó a la conclusión de que la altura óptima para el LIDAR par que puedas evadir la mayoría de obstáculos es de 20 cm y se coteja la información antes mencionada con la siguiente gráfica:

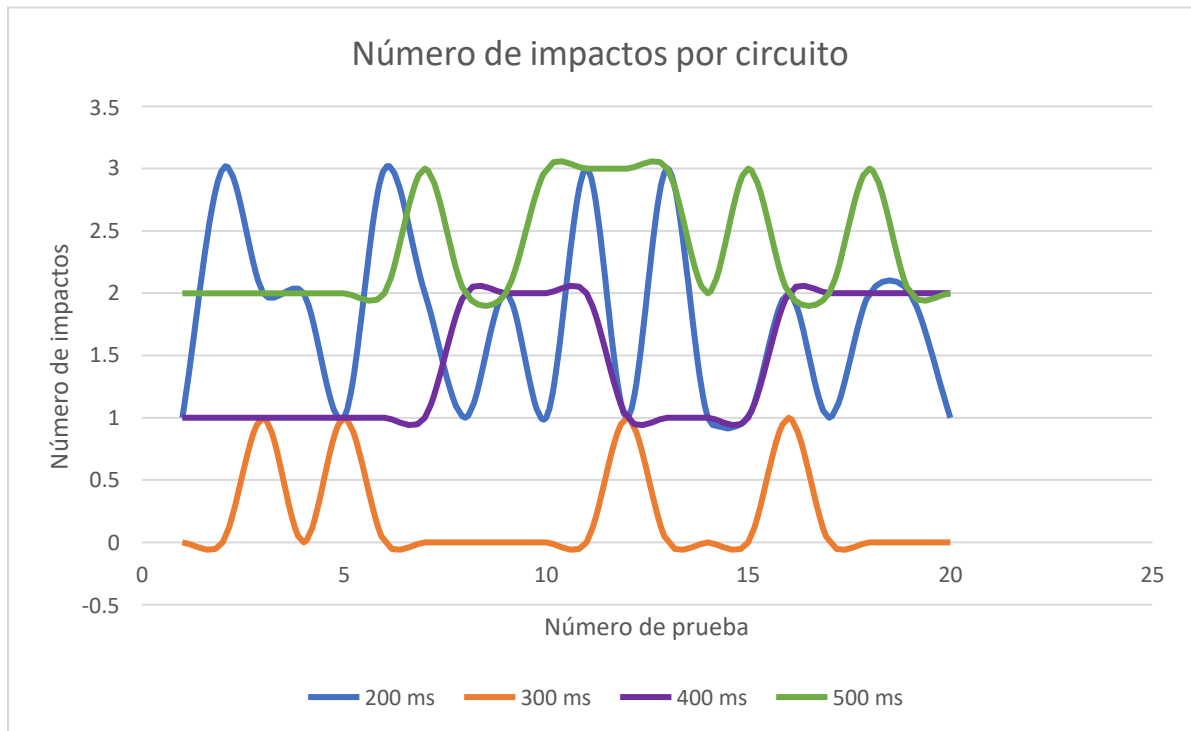


Ilustración 1 Gráfica de impactos por circuito

En cuanto a los obstáculos verticales no se tuvo mayor inconveniente con la adaptación hecha en la altura del sensor. Para definir el ángulo de visión del sensor LIDAR se adaptó un servomotor que permitiría girarlo cada intervalo de tiempo para mantener un espectro muy similar al del ojo humano basado en la siguiente Ilustración 2:

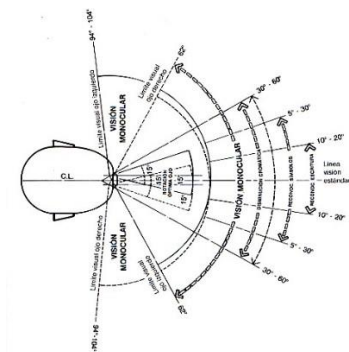


Ilustración 2, Espectro de visión del ojo humano

Por lo cual se estableció que para una correcta interpretación del entorno que recorre el robot se debería de colocar un espectro de visión de -62° a 62° para cubrir todo el frente. Para ello se analizaron diferentes escalas de tiempo en milisegundos para la rotación del LIDAR como se muestra en la Tabla 4

Tabla 4 Impactos calibrando el servomotor

Número de ejecución	200 ms	300 ms	400 ms	500 ms
1	1	1	2	1
2	0	1	2	2
3	2	0	3	2
4	1	0	2	1
5	1	0	1	2
6	0	0	1	2
7	0	0	2	1
8	1	0	1	2
9	2	0	1	1
10	1	1	1	2
11	0	0	1	1
12	0	1	1	2
13	1	0	0	1
14	2	1	2	2
15	1	1	0	2
16	0	0	1	1
17	1	0	1	2
18	1	0	2	1
Promedio de impactos	0.83	0.33	1.33	1.56

Generando la gráfica respectiva se determinó que el tiempo en milisegundos mas estable para el giro del LIDAR es de 300 ms, cómo se muestra en la siguiente imagen:

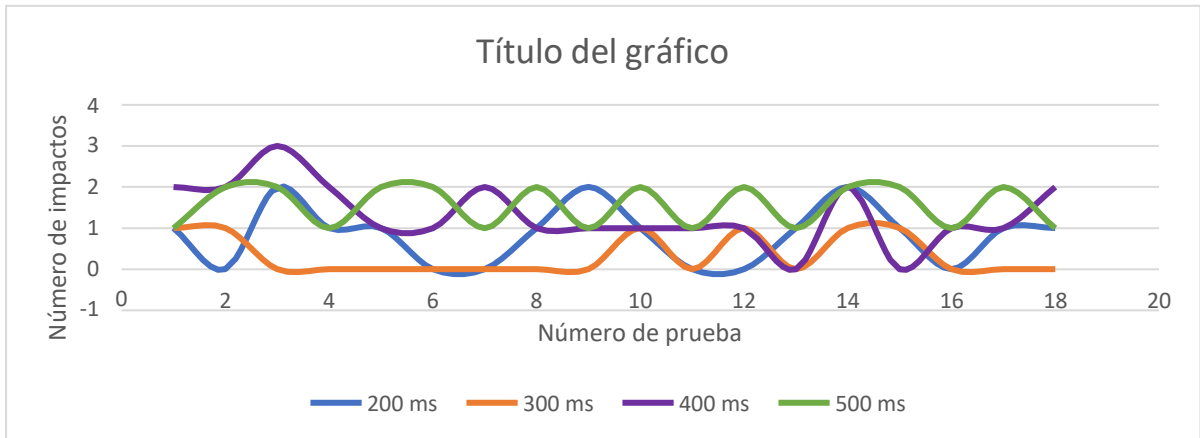


Ilustración 3 Gráfica de comportamiento del servomotor

Dados los resultados anteriores se determinó que los parámetros para una navegación eficiente son de 300 ms como intervalo de tiempo para cambiar el ángulo de visión del sensor LIDAR calibrado de -62° a 62° con una altura de 20 cm del suelo al cabezal del sensor con respecto a las dimensiones antes mencionadas del robot.

En la Tabla 5 vamos a observar los principales objetos que se ocuparon para las pruebas del prototipo, que serán también los mas usados para los entornos que los utilice en el cual se representa la información de alcance y detección.

Tabla 5 Objetos reconocidos

Objeto	Distancia máxima	Distancia mínima	Reconocido	Altura mts
Muros	60 Cm	2 Cm	si	2
Barras verticales	60 Cm	2 Cm	si	2
Barras horizontales	60 Cm	2 Cm	si	0.25
Sillas	60 Cm	2 Cm	si	1.25
Otros	60 Cm	2 Cm	no	0.05

Conclusiones y trabajo futuro

En este apartado se presentan las conclusiones después de toda la práctica y conocimientos adquiridos al culminar el desarrollo del prototipo del módulo de autonomía, así como también la visión que dejo a futuro basada en las ventanas de oportunidad que logramos identificar como puntos clave.

Se ha diseñado, construido y programado un sistema capaz de medir distancias en tiempo real y a partir de estas se obtiene la navegación con movimiento autónomo y detección de obstáculos. Los objetivos principales para cumplir en este proyecto son: navegación autónoma y evasión de obstáculos, para dicha detección se han investigado las distintas posibilidades de sistemas de medición de distancia con el objetivo de encontrar el que más se adapte a nuestras necesidades.

Se decidió que la mejor opción es el empleo de sensores láser, más concretamente del sensor LIDAR, debido a sus buenas características de precisión para su bajo coste y relativamente baja dificultad computacional, basados en una comparativa que ofreciera la eficiencia que buscamos realizamos la demás selección de herramientas en este caso para el procesamiento de datos se opta por montar una arquitectura Arduino ya que es lo más común y práctico para estos casos, según la literatura y otros proyectos antecesores en los cuales tomamos como referencia y bases para la creación de nuestro prototipo, el diseño de la estructura con la cual demostraríamos la funcionalidad de este módulo fue creada a partir de las características de los componentes se realiza un vehículo de tres ruedas, a modo de simular un pequeño robot, los materiales que usamos son relativamente económicos.

El uso de LIDAR se implementa mayormente con Slam de Ros aunque aquí se optó por el diseño de un algoritmo propio, este mismo fue plasmado sobre el lenguaje de programación de Arduino. La navegación dependía totalmente de los datos recabados por el sensor los cuales se procesaban a través de los pines de entrada de Arduino aquí aprendimos el cómo funcionan los dos tipos de señales y cuáles son las que se requieren según la necesidad, se deduce que los movimientos deben ser divididos y por cada uno creados métodos, de esta manera era más fácil que se interpretaran correctamente los datos numéricos recibidos.

En las primeras pruebas se tuvieron que realizar correcciones sobre tiempos y valores de algunas variables para que el movimiento no fuera torpe ni pausado, después de cada uno de estos ajustes se puede considerar como exitosa la implementación del algoritmo y el armado de la estructura pues en conjunto dieron los resultados esperados.

El introducirnos en esta área de oportunidad nos deja claro que hay mucho por explorar y explotar ya que a pesar de que los avances se han dado a pasos agigantados en todo el mundo en nuestro país en específico hay mucho rezagó en estos temas y todas las investigaciones y proyectos están principalmente creadas en países extranjeros, el incentivar a trabajar en estas áreas podría darnos un nuevo campo muy productivo en donde desarrollarnos.

Los usos que se pueden tener con esta herramienta desarrollada son principalmente la implementación a otras herramientas como pueden dispositivos de desinfección en donde la operación humana sea poco segura.

Trabajo futuro

Este módulo puede usarse de manera complementaria sobre otras herramientas que requieren ser tele operadas por la intervención humana, y esta pueda representar un riesgo, como lo son dispositivos de limpieza, excavaciones, exploraciones en áreas de siniestro o contaminadas, o bien desarrollar una propuesta para implementar la tecnología que ofrece por ejemplo la industria automotriz que en la actualidad operan ya con esta incluida pero los precios son elevados y se tiene que adquirir completamente una unidad nueva, se podría abrir una ventana de oportunidad en la que sobre los autos que no cuentan con dicha característica se pueda acondicionar de manera externa, viéndolo como un módulo de asistencia autónoma para el estacionado, podemos crear algún sistema de videovigilancia móvil, para ciertas áreas en específico, lo alcances que puede tener y usos son demasiado variados y con grandes oportunidades de crecimiento.

Referencias

- YELLOWSCAN 2020. (s.f.). *YELLOWSCAN*. Recuperado el 02 de Marzo de 2022, de <https://www.yellowscan-lidar.com/es/knowledge/how-lidar-works/#:~:text=Detecci%C3%B3n%20y%20alcance%20de%20la,en%20volver%20a%20su%20fuente.>
- Artero, O. T. (2018). *Arduino*. Madrid: Libros Ur.
- Baturone, A. O. (2001). *Robótica manipuladores y robots móviles*. España: Marcombo & Alfa y Omega.
- Celera Motion. (2019). *Celera Motion*. Recuperado el 24 de Febrero de 2022, de <https://www.celeramotion.com/zettlex/es/asistencia/documentacion-tecnica/motores-de-transmision-directa-y-sensores-directos/>
- Cruz, K. J. (Mayo de 2020). Obtenido de https://repositorio.sena.edu.co/bitstream/handle/11404/6927/Informe_de_vigilancia_tecnologica_sobre_robots.pdf?sequence=1
- Dominik, J. (19 de Julio de 2018). Obtenido de <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=%2Fnetahtml%2FPTO%2Fsearch-bool.html&r=1&f=G&l=50&co1=AND&d=PTXT&s1=%2220180203460+A1%22&OS=>
- Ebrahimi, A. A. (25 de Abril de 2019). Obtenido de <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=%2Fnetahtml%2FPTO%2Fsearch-bool>.
- Fillottrani, E. M. (2018). *Sedeci.edu*. Recuperado el 4 de Enero de 2022, de http://sedeci.unlp.edu.ar/bitstream/handle/10915/21720/Arquitectura_de_Software.pdf;jsessionid=F11036762D7F397F4537E05E927011CF?sequence=1
- George, K. J. (3 de Enero de 2019). Obtenido de . Jan 3, 2019
- Gonzalez, A. G. (Diciembre de 2016). *Bolanosdj*. Recuperado el 22 de 2022 de Febrero , de <https://www.bolanosdj.com.ar/MOVIL/ARDUINO2/IntroServos.pdf>
- Gregg, R. (19 de Diciembre de 2019). Obtenido de <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=%2Fnetahtml%2FPTO%2Fsearch-bool.html&r=1&f=G&l=50&co1=AND&d=PTXT&s1=%2220190384315+A1%22&OS=>
- Hong, H. (12 de Marzo de 2020). Obtenido de <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=%2Fnetahtml%2FPTO%2Fsearch-bool.html&r=1&f=G&l=50&co1=AND&d=PTXT&s1=%2220200080860+A1%22&OS=>
- José Luis Martínez. (2020). *Clem*. Recuperado el 02 de Marzo de 2022, de <https://clem.es/noticia/navegacion-slam-robots-que-construyen-mapas-514>
- Llamas, L. (2020). *Luis Llamas*. Recuperado el 22 de Febrero de 2022, de <https://www.luisllamas.es/entradas-analogicas-en-arduino/>

- Lope, J. d. (2019). *Diafi.upm*. Recuperado el 4 de Enero de 2022, de <http://www.dia.fi.upm.es/~jdlope/slides/arquit.pdf>
- Luis de Valdes Hernandez. (2019). *Programar Fácil*. Recuperado el 22 de Febrero de 2022, de <https://programarfacil.com/blog/arduino-blog/pines-digitales-de-arduino/>
- Márquez, J. M. (2019). *Tecnología, programación y robótica*. Oxfon Educación.
- Miller, L. R. (Enero de 2017). Recuperado el 2021, de <https://repositorio.upct.es/bitstream/handle/10317/6449/tfg-reb-aut.pdf?sequence=1&isAllowed=y>
- MIOIO OPENSAT. (2018). *MIOIO*. Recuperado el 22 de Febrero de 2022, de <http://www.mioio.es/entradassalidas-digitales/#:~:text=Las%20entradas%20y%20salidas%20digitales,1%20o%20falsas%20y%20verdaderas.>
- Morales, R. T. (2007). *ddd.uab*. Recuperado el 4 de Enero de 2022, de https://ddd.uab.cat/pub/trerecpro/2007/hdl_2072_9045/PFCDAvilaMolina.pdf
- Murlanda, J. F. (2018). *Repositorio Unicauca*. Recuperado el 04 de Enero de 2022, de <http://repositorio.unicauca.edu.co:8080/bitstream/handle/123456789/479/ARQUITECTURA%20GEN%C3%89RICA%20PARA%20SISTEMAS%20DE%20ROB%C3%93TICA%20MOV%C3%8DL.pdf?sequence=3&isAllowed=y>
- Onofre, R. L. (2021). *UN FRAMEWORK BASADO EN ROS PARA LA NAVEGACIÓN DE ROBOTS MÓVILES AUTÓNOMOS*. Mexico.
- Orrente, O. (2015). *Arduino. Curso practico de formacion*. España.
- Peña, C. (2021). *Arduino. Entradas y salidas: digitales y analógicas*. Argentina: Plandos S.A.
- Ramírez, P. A. (2016). Aprendizaje de y con robótica. *Revista Educacion*, 1(1), 21.
- Reyes, F., & Cid, J. (2016). *ARDUINO: Aplicado en Robótica, mecatronica e ingenierías*. México: AlfaOmega.
- Ripipsa. (2020). *Ripipsa*. (Ripipsa S.A. de C.V.) Recuperado el 04 de Enero de 2022, de <https://ripipsacobots.com/robots-autonomos/>
- Ruiz-de-Garibay, J. (2017). *Robótica: Estado del arte*. Recuperado el 04 de Enero de 2022, de <https://d1wqtxts1xzle7.cloudfront.net/5533677/robotica-with-cover-page-v2.pdf?Expires=1641311880&Signature=KUB3VBsarDaHqSnwOdxLdJVMTzdTauFaDevo6ie4zhYGGm7ynrMfqMC27s2qGohiGAOv6cNdBafHxo1NOBWPrCS9HNzzL8pMybE0MjBckU0Ah~1Ol~vzuzfoXjQCIRmz3eBtjodivOHsppXtRj8EH>
- Russo, D. (2021). *Intelliget Human Systems Integration 2021*. USA: Springer.
- Santiago, J. G. (2019). *uaeh.edu*. Recuperado el 23 de Febrero de 2022, de https://www.uaeh.edu.mx/docencia/P_Presentaciones/prepa3/2020/mecanismos-que-usan-robots.pdf

SENSOR GO. (2021). *SENSOR GO*. Recuperado el 01 de Marzo de 2022, de <https://sensorgo.mx/que-son-los-instrumentos-de-medicion-y-por-que-debes-implementarlos-en-tu-empresa/>

Serma, E. (2019). *DESARROLLO E INNOVACIÓN DE INGENIERIA*. Antioquia: IAI.

Torres, J. M. (2017). *Control Bilateral de Robots Teleoperados*. España: Editorial Academica Española.

Anexos

Anexo 1 Código de funcionamiento

Se detalla con precisión en el siguiente código comentado la lógica implementada para esta primera fase de pruebas de movimiento utilizando el sensor LIDAR v3.

```
#include <Wire.h>
#include <LIDARLite.h>
#include <Servo.h>
//variable para el servo
Servo servoMotor;
//Variable para llevar el conteo de los grados
int grado;
//valores de ida y vuelta del servo
boolean aumento;
boolean decremento;
//variables para verificar si el sensor detecta un objeto cercano
boolean obstaculo;
boolean libre;
int limite;
//Objeto de la librería LidarLite para controlar el sensor
LIDARLite myLidarLite;
// Motor A
int IN1 = 7;
int IN2 = 8;
// Motor B
int IN3 = 9;
int IN4 = 10;
//MÉTODO PARA AVANZAR
void Adelante ()
{
//Direccion motor A
digitalWrite (IN1, LOW);
digitalWrite (IN2, HIGH);
//Direccion motor B
digitalWrite (IN3, LOW);
digitalWrite (IN4, HIGH);
}
//Y ahora para moverse en el sentido contrario:
void Atras ()
{
//Direccion motor A
digitalWrite (IN1, HIGH);
digitalWrite (IN2, LOW);
//Direccion motor B
digitalWrite (IN3, HIGH);
```

```

digitalWrite (IN4, LOW);
}
// Giro a la derecha
void Derecha ()
{
//Direccion motor A
digitalWrite (IN1, LOW);
digitalWrite (IN2, HIGH);
//Direccion motor B
digitalWrite (IN3, HIGH);
digitalWrite (IN4, LOW);
}
//Girar a la izquierda
void Izquierda ()
{
//Direccion motor A
digitalWrite (IN1, HIGH);
digitalWrite (IN2, LOW);
//Direccion motor B
digitalWrite (IN3, LOW);
digitalWrite (IN4, HIGH);
}
//Detener el robot.
void Parar ()
{
//Direccion motor A
digitalWrite (IN1, LOW);
digitalWrite (IN2, LOW);
//Direccion motor B
digitalWrite (IN3, LOW);
digitalWrite (IN4, LOW);
}
void setup() {
Serial.begin(115200); // Inicializamos el puerto serial a esta frecuencia para que tenga buena
conectividad con el LIDAR
myLidarLite.begin(0, true); // Colocamos esta configuración para establecer un a comunicación
a 400 HZ
myLidarLite.configure(3); // Este valor se puede cambiar por un valor enre 1-5 para ajustar las
sensibilidad y el alcance del LIDAR
servoMotor.attach(3); //Puerto en el que está trabajando el servomotor
grado = 0;
aumento = true;
decremento = false;
obstaculo = false;
libre= true;
limite=20;
// Declaramos todos los pines de los motores como salidas
pinMode (IN1, OUTPUT);

```

```

pinMode (IN2, OUTPUT);
pinMode (IN3, OUTPUT);
pinMode (IN4, OUTPUT);
}
void loop() {
// como inicialmente el ángulo del servo inicia en 0, este va ir aumentando, hasta llegar a 125
grados
if (aumento) {
//cuando el contador de los grados llega a este valor, se detiene y regresa.
if (grado < 125) {
//si en el lapso de que aumenta el contador de grados se detecta un obstáculo se gira a la
izquierda
//durante 100 ms e intenta verificar si se libró el obstáculo
if(obstaculo){
Izquierda();
delay(100);
libre = true;
obstaculo = false;
}
//si durante el aumento de grados se encuentra libre, el robto avanza hasta encontrar un
obstáculo.
if(libre){
if(myLidarLite.distance())<limite){
obstaculo = true;
Parar();
delay(100);
}else{
Adelante();
}
}
// el contador de grados va a umentar de 3 en tres para que sea tan lento el ciclo de escaneo.
grado =grado+3;
servoMotor.write(grado);
//Serial.println(myLidarLite.distance());
//Serial.println(grado);
delay(1);
//una vez que el contadr de gradios llega a 125, aumento entra en falso
//y se activa el procedimiento de decremento
} else {
aumento = false;
decremento = true;
}
}
if (decremento) {
if (grado > 65) {
if(obstaculo){
Derecha();
libre = true;

```

```
obstaculo = false;
}
if(libre){
if(myLidarLite.distance()<limite){
obstaculo = true;
Parar();
delay(100);
}else{
Adelante();
}
}
grado = grado-3;
servoMotor.write(grado);
//
//Serial.println(myLidarLite.distance());
//Serial.println(grado);
delay(1);
} else {
aumento = true;
decremento = false;
}
}
}
```