



**Universidade Federal da Bahia  
Escola Politécnica  
Colegiado do Curso de Eng. Elétrica**



Lucas Pinheiro Soares

## **Projeto e Construção de um Robô Móvel *Car-Like* Equipado com LiDAR 2D**

Orientador(a): Prof. Dr. Tiago Trindade Ribeiro

Salvador-Ba – Brasil  
12 de Dezembro de 2023



Lucas Pinheiro Soares

## **Projeto e Construção de um Robô Móvel *Car-Like* Equipado com LiDAR 2D**

Projeto apresentado ao Curso de Graduação em Engenharia Elétrica da Universidade Federal da Bahia como parte dos requisitos para a obtenção do grau de Engenheiro(a) Eletricista.

Orientador(a): Prof. Dr. Tiago Trindade Ribeiro

Salvador-Ba – Brasil

12 de Dezembro de 2023



Lucas Pinheiro Soares

## **Projeto e Construção de um Robô Móvel *Car-Like* Equipado com LiDAR 2D**

Projeto apresentado ao Curso de Graduação em Engenharia Elétrica da Universidade Federal da Bahia como parte dos requisitos para a obtenção do grau de Engenheiro(a) Eletricista.

Trabalho aprovado. Salvador-Ba – Brasil, 12 de Dezembro de 2023:

---

Orientador(a): Prof. Dr. Tiago Trindade  
Ribeiro

---

Prof. Dr. André Gustavo Scolari Conceição

---

Prof. Dr. Paulo César Machado de Abreu Farias

Salvador-Ba – Brasil  
12 de Dezembro de 2023



*Este trabalho é dedicado ao meu pai, à minha mãe e à minha família, que me apoiaram por toda a minha vida.*



# Agradecimentos

Agradeço a Márcia Muniz, minha namorada, por todo o apoio e por estar sempre ao meu lado , à minha família por todo o amor, festas e alegrias, especialmente aos meus pais e a minha irmã por todo o incentivo. A Juan Prazeres pelo companheirismo e amizade, a Roberto Machado e Sara Santana por não me deixar desistir, a Felipe Araújo por trilhar a jornada ao meu lado, à Algetec pelo apoio durante minha formação. E principalmente à Equipe Carpoeira Baja pelas amizades que lá fiz.



*“Fear tends to come from ignorance. Once I knew what the problem was, it was just a problem, nothing to fear”.*  
*(Patrick Rothfuss, The Name of the Wind)*



# Resumo

Este projeto apresenta o desenvolvimento de um robô car-like equipado com um LiDAR 2D capaz de receber e transmitir dados. O objetivo do trabalho é projetar e construir um protótipo onde seja possível desenvolver atividades e estudos relacionados à robótica e computação, envolvendo mapeamento, localização e navegação. O projeto do robô parte das premissas de projeto, escolha de componentes e desenvolvimento dos sistemas do LiDAR, direção de Ackermann e chassis.

O sistema embarcado foi projetado levando em consideração requisitos de conectividade rápida utilizando o ESP32 como plataforma computacional, recebendo e tratando dados recebidos dos sensores, transmitindo e recebendo dados através do protocolo UDP e fazendo controle dos motores e servos.

Controladores PID discretos para velocidade de motores de tração e rotação do LiDAR foram projetados e sintonizados considerando critérios de desempenho, utilizando o método do lugar das raízes para o sistema dinâmico discretizado, levantado por análise transitória.

Todo o projeto foi modelado e impresso em 3D, cumprindo todos os requisitos de projeto. Foram coletados dados experimentais dos sistemas, apresentando resultados satisfatórios, possibilitando a exploração do protótipo nas atividades designadas.

**Palavras-chave:** Robô móvel. LiDAR 2D. SLAM. Ackermann. Car-Like.



# Abstract

This project presents the development of a car-like robot equipped with a 2D LiDAR capable of receiving and transmitting data. The goal of the work is to design and build a prototype where it is possible to carry out activities and studies related to robotics and computing, involving mapping, localization, and navigation. The robot project is based on design premises, component selection, and the development of LiDAR systems, Ackermann steering, and chassis.

The embedded system was designed taking into account fast connectivity requirements using the ESP32 as the computing platform, receiving and processing data from sensors, transmitting and receiving data through the UDP protocol, and controlling motors and servos.

Discrete PID controllers for the speed of traction motors and rotation of the LiDAR were designed and tuned considering performance criteria, using the root locus method for the discretized dynamic system raised by transient analysis.

The entire project was modeled and 3D printed, meeting all design requirements. Experimental data from the systems were collected, showing satisfactory results, enabling the exploration of the prototype in the designated activities.

**Keywords:** Mobile robot. 2D LiDAR. SLAM. Ackermann. Car-Like.



# **Lista de ilustrações**

Figura 1 – Robô antibomba . . . . .	25
Figura 2 – Proteus, robô da Amazon que ajuda na logística . . . . .	26
Figura 3 – Ballie, robô doméstico da Samsung . . . . .	27
Figura 4 – Geometria de Ackerman . . . . .	32
Figura 5 – Diagrama de controle Clássico . . . . .	34
Figura 6 – Funcionamento do Sensor de Triangulação . . . . .	36
Figura 7 – Funcionamento do Sensor ToF . . . . .	37
Figura 8 – Interface de Sinais do Encoder de Quadratura . . . . .	38
Figura 9 – LiDAR SICK . . . . .	40
Figura 10 – Exemplo de sistema embarcado utilizando microcontrolador . . . . .	44
Figura 11 – Utilização de firmwares em sistemas embarcados . . . . .	44
Figura 12 – Servo . . . . .	49
Figura 13 – Geometria de direção trapezoidal . . . . .	50
Figura 14 – Motor CC . . . . .	51
Figura 15 – Placa de desenvolvimento ESP32 . . . . .	53
Figura 16 – Módulo sensor VL53L0X . . . . .	54
Figura 17 – Anel rotativo . . . . .	55
Figura 18 – Sensor de obstáculo . . . . .	55
Figura 19 – Diagrama elétrico . . . . .	57
Figura 20 – Pinagem do ESP32 . . . . .	58
Figura 21 – Suporte de validação da reflexão do laser, sensor destacado em laranja	60
Figura 22 – Gerador de engrenagens no OpenSCAD . . . . .	61
Figura 23 – Suporte de validação do LiDAR com espelho, espelho e sensor destacados em laranja . . . . .	62
Figura 24 – Suporte de validação impresso . . . . .	62
Figura 25 – Mudança do sistema do LiDAR . . . . .	63
Figura 26 – Sistema de detecção de volta . . . . .	64
Figura 27 – Sistema de teste de direção, engrenagem primária, braço de suporte e manga de eixo destacados em laranja . . . . .	65
Figura 28 – Sistema de teste de direção impresso . . . . .	65
Figura 29 – Projeto completo, sensor, suporte das baterias e regulador de tensão destacados em laranja . . . . .	66
Figura 30 – Projeto completo impresso . . . . .	67
Figura 31 – Exemplo de String de leitura do LiDAR . . . . .	68
Figura 32 – Levantamento da curva . . . . .	70
Figura 33 – Lugar das raízes do sistema discreto . . . . .	71

Figura 34 – Lugar das raízes do sistema com controlador	72
Figura 35 – Ciclo de leitura de distância	74
Figura 36 – Ensaio comparativo a 365mm/s	76
Figura 37 – Ensaio comparativo a 469mm/s	76
Figura 38 – Ensaio comparativo a 600mm/s	77
Figura 39 – Teste com carga de 1kg	77
Figura 40 – Ensaio comparativo de esforço	78
Figura 41 – Teste de detecção de obstáculos	80
Figura 42 – Resultado do teste de detecção de obstáculos, ponto fora da região do teste não representados	81
Figura 43 – Teste de detecção de vaga	81
Figura 44 – Resultado do teste de detecção de vaga, pontos fora da região do teste não representados	82
Figura 45 – Teste de ângulo de visão	82
Figura 46 – Resultado do teste de ângulo de visão, pontos adquiridos destacados em laranja	83
Figura 47 – Teste de raio de curva	84

# **Lista de tabelas**

Tabela 1 – Diferença entre Sistemas Holonômicos e Não Holonômicos . . . . .	30
Tabela 2 – Comparativo entre dispositivos controladores . . . . .	46
Tabela 3 – Alocação de portas . . . . .	58
Tabela 4 – Teste de tempo de volta - 150mm/s . . . . .	78
Tabela 5 – Teste de tempo de volta - 500mm/s . . . . .	78
Tabela 6 – Limitação de velocidade pela quantidade de medidas . . . . .	79
Tabela 7 – Teste de precisão de medição . . . . .	80
Tabela 8 – Resultado do teste de ângulo de visão . . . . .	83
Tabela 9 – Lista de comandos . . . . .	93
Tabela 10 – Lista de componentes . . . . .	97



# Lista de abreviaturas e siglas

<b>CC</b>	Corrente Contínua
<b>CI</b>	Círculo Integrado
<b>DDOF</b>	<i>Differential Degrees Of Freedom</i>
<b>DOF</b>	<i>Degrees Of Freedom</i>
<b>EVA</b>	Etil Vinil Acetato
<b>IA</b>	Inteligência Artificial
<b>ICR</b>	<i>Instantaneous Centre of Rotation</i>
<b>IoT</b>	<i>Internet of Things</i>
<b>LiDAR</b>	<i>Light detection and ranging</i>
<b>PWM</b>	<i>Pulse Width Modulation</i>
<b>PID</b>	Proporcional, Integral e Derivativo
<b>SLAM</b>	<i>Simultaneous Localization and Mapping</i>
<b>ToF</b>	<i>Time-of-Flight</i>
<b>UDP</b>	<i>User Datagram Protocol</i>



# Lista de símbolos

$\gamma$	Ângulo de viragem
$\gamma_{int}$	Ângulo de viragem interno
$\gamma_{ext}$	Ângulo de viragem externo
$L$	Distância entre os eixos dianteiro e traseiro
$R$	Raio de curvatura
$W$	Largura entre os eixos de rotação vertical das rodas dianteiras
$\tau$	Constante de tempo
$\tau_d$	Atraso da função
$A$	Ganho dos sistemas
$s$	Frequência complexa
$kp$	Ganho proporcional
$ki$	Ganho integral
$kd$	Ganho derivativo
$y_k$	Saída discreta
$u_k$	Entrada discreta
$T$	Período de amostragem
$z$	Variável complexa discreta
$rel$	Relação de engrenagens
$t_1$	Tempo do último pulso do encoder
$t_2$	Tempo do penúltimos pulso do encoder
$p$	Quantidade de pulsos do encoder
$d$	Diâmetro do pneu
$\sigma$	Parte real dos sistemas em frequência
$\alpha$	Posição do zero do controlador discreto
$e$	erro discreto



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>25</b>
<b>1.1</b>	<b>Contextualização e justificativa</b>	<b>25</b>
<b>1.2</b>	<b>Objetivos</b>	<b>27</b>
1.2.1	Objetivos específicos	27
<b>1.3</b>	<b>Organização do trabalho</b>	<b>28</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>29</b>
<b>2.1</b>	<b>Robótica</b>	<b>29</b>
2.1.1	Car-like e geometrias de direção veicular	31
2.1.1.1	Mapeamento	32
<b>2.2</b>	<b>Sistemas de controle</b>	<b>33</b>
2.2.1	Sensoreamento	35
2.2.1.1	Sensor de distância	35
2.2.1.1.1	Sensor Ultrassônico	35
2.2.1.1.2	Sensor de Triangulação	35
2.2.1.1.3	Sensor de Tempo de voo - ToF ( <i>Time of Flight</i> )	36
2.2.1.2	Encoder	38
2.2.1.3	LiDAR	39
2.2.2	Modelagem de sistemas	39
2.2.3	Controle PID	41
2.2.4	Discretização por Tustin	42
2.2.5	O lugar das raízes	42
<b>2.3</b>	<b>Sistemas embarcados</b>	<b>43</b>
<b>3</b>	<b>DESENVOLVIMENTO</b>	<b>47</b>
<b>3.1</b>	<b>Premissas do projeto</b>	<b>47</b>
3.1.1	Velocidade de movimentação	47
3.1.2	Características do LiDAR	47
3.1.3	Tamanho do protótipo	47
3.1.4	Autonomia	48
3.1.5	Conectividade	48
3.1.6	Suspensão e direção	48
3.1.7	Tração	48
<b>3.2</b>	<b>Escolha dos componentes</b>	<b>49</b>
3.2.1	Direção	49
3.2.2	Motores CC	50

3.2.3	Bateria . . . . .	51
3.2.4	Plataforma computacional . . . . .	52
3.2.5	Sensor de distância . . . . .	53
3.2.6	Outros componentes . . . . .	54
3.2.6.1	Sistema do LiDAR . . . . .	54
3.2.6.2	Sistema de alimentação . . . . .	54
3.2.6.3	Módulo ponte H . . . . .	56
<b>3.3</b>	<b>Diagrama elétrico</b> . . . . .	<b>56</b>
<b>3.4</b>	<b>Modelagem 3D</b> . . . . .	<b>59</b>
3.4.1	LiDAR . . . . .	59
3.4.2	Testes iniciais e ajustes . . . . .	61
3.4.3	Sistema de Direção . . . . .	63
3.4.4	Chassi . . . . .	64
<b>3.5</b>	<b>Aquisição e controle</b> . . . . .	<b>67</b>
3.5.1	Comunicação . . . . .	67
3.5.2	Direção . . . . .	68
3.5.3	Leituras de velocidade . . . . .	69
3.5.4	Controle dos motores . . . . .	69
3.5.5	Leitura de distância . . . . .	73
<b>4</b>	<b>ANÁLISE DE RESULTADOS</b> . . . . .	<b>75</b>
<b>4.1</b>	<b>Testes do controlador de velocidade</b> . . . . .	<b>75</b>
4.1.1	Degrau . . . . .	75
4.1.2	Testes de esforço . . . . .	75
4.1.3	Compensação de curva . . . . .	77
<b>4.2</b>	<b>Validação do LiDAR</b> . . . . .	<b>79</b>
4.2.1	Testes de leitura . . . . .	79
4.2.2	Ângulo de visão . . . . .	80
<b>4.3</b>	<b>Teste de raio de curva</b> . . . . .	<b>83</b>
<b>5</b>	<b>CONCLUSÃO</b> . . . . .	<b>85</b>
<b>5.1</b>	<b>Sugestão de trabalhos futuros</b> . . . . .	<b>85</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>87</b>
	<b>APÊNDICE A – TABELA DE COMANDOS</b> . . . . .	<b>93</b>
	<b>APÊNDICE B – FICHA TÉCNICA</b> . . . . .	<b>95</b>
	<b>APÊNDICE C – LISTA DE COMPONENTES</b> . . . . .	<b>97</b>

# 1 Introdução

## 1.1 Contextualização e justificativa

A robótica alcançou seu maior sucesso na indústria através da utilização de braços robóticos manipuladores para a realização de tarefas minuciosas como o processo de soldagem de chassis e posicionamento de componentes eletrônicos com precisão sobre-humana, porém, apesar de seu sucesso, lhes falta a capacidade de deslocamento para a realização de tarefas onde mais forem necessários([PETER, 2011](#)).

Os robôs moveis não possuem essa limitação, assim o estudo e desenvolvimento de sistemas robóticos móveis é de extrema importância no avanço tecnológico não só industrial, mas no cotidiano da humanidade.

Robôs são utilizados na indústria, dentre outros fatores, devido à sua precisão, eficiência e a capacidade de realizar tarefas que trariam riscos a humanos seja devido ao ambiente, seja devido à forma como a atividade tem que ser exercida.

Um robô pode operar em situações de risco e em ambientes não acessíveis por humanos, como incêndios e desastres, ou na presença de gases nocivos e risco de explosão (Figura 1).

Figura 1 – Robô antibomba



Fonte: ([CASTELLI, 2013](#))

A precisão de posicionamento e capacidade de executar exatamente a mesma operação repetidas vezes é de extrema importância na indústria para manter a padronização

de produtos e garantir o seu funcionamento. A utilização de sensores permite a identificação e correção de qualquer desvio na execução da tarefa desejada.

A eficiência está relacionada à capacidade de otimização do processo pelo qual as atividades devem ser exercidas, tanto em questão de velocidade quanto no gasto energético e uso de matéria prima.

A indústria 4.0 chegou elevando a escala da otimização com a integração entre os robôs requerendo ainda mais a robótica móvel (Figura 2), aplicando metodologias de aprendizado de máquina e inteligência artificial e permitindo a integração entre robôs e operadores humanos de forma harmoniosa e segura.

Figura 2 – Proteus, robô da Amazon que ajuda na logística



Fonte: ([MENDES, 2022](#))

O convívio homem-máquina vem sendo cada vez mais cotidiano com a interação com tecnologias autônomas que vão desde a utilização de robôs aspiradores de pó ao lançamento de cada vez mais modelos de carros com sistemas de *Park Assist* ou até *Full Self-Driving*, e o desenvolvimento de robôs para assistência pessoal, como o Ballie da Samsung ([TRONNIC, 2020](#)) (Figura 3).

Robôs moveis também já são encontrados ou estão sendo desenvolvidos em diversas outras áreas de interesse: como no entretenimento, com os shows de drones voando em formação ([NOGUEIRA, 2019](#)), na agricultura, com grandes colheitadeiras autônomas ([VÉDERA, 2021](#)) e drones utilizados para fertilização ([FORNTEIRAS, 2023](#)), na segurança pública, com drones de vigilância ([ALEXANDRINO, 2022](#)), na área militar, com caças autônomos ([MARCUS, 2022](#)), entregas e logística, com robôs funcionando na entrega de suprimentos médicos ([VIEIRA, 2019](#)) e organização de mercadorias em galpões ([MENDES, 2022](#)), dentre outras muitas áreas. A versatilidade no uso de robôs é notável.

A vasta gama de possibilidades demonstra a importância do estudo e desenvol-

Figura 3 – Ballie, robô doméstico da Samsung



Fonte: ([TRONNIC, 2020](#))

vimentos de novas tecnologias voltadas à robótica móvel. Assim, este trabalho planeja construir um robô móvel onde seja possível desenvolver conhecimentos da robótica e computação a respeito de limitações cinemáticas e o modelo de direção de Ackermann, integração de sensores, aquisição de dados, sistema de controle, comunicação, modelagem e impressão 3D dentre outros, possibilitando a implementação de algoritmos diversos relacionados à localização, mapeamento e orientação, permitindo a comunicação em com plataformas diversas.

## 1.2 Objetivos

O objetivo do trabalho é projetar um pequeno robô móvel com rodas na configuração *car-like*, equipado com um *Light detection and ranging* (LiDAR) 2D, capaz de receber e transmitir dados e responder a comandos de controle de movimentação e ajustes de configurações.

### 1.2.1 Objetivos específicos

- Estabelecer uma comunicação sem fio rápida.
- Realizar tarefas de localização e navegação com ou sem ajuda de outros sistemas.
- Estabelecer um protocolo com comando simples para movimentação, como acelerar, frear, virar à direita, dar ré, etc.
- Projetar aspectos físicos simplificados para facilitar a modelagem do sistema, como um chassi quadrado ou eixos de direção verticais.
- Controlar o comportamento dinâmico de forma que seja previsível e com ajustes rápidos.

### 1.3 Organização do trabalho

O seguinte trabalho está estruturado em sete capítulos para a melhor organização e compreensão do desenvolvimento do projeto.

No capítulo 1 é feita a introdução, onde é desenvolvida a contextualização do assunto a ser discutido, as justificativas e relevância do projeto no cotidiano.

No capítulo 2 é apresentada a fundamentação teórica, onde são explorados os assuntos relacionados ao projeto, envolvendo a robótica e as peculiaridades de um robô *car-like*, a utilização de sensores, os sistemas de controle e comunicação.

No capítulo 3, o projeto foi desenvolvido. Foram estabelecidas premissas para orientar o restante do trabalho em direção a objetivos claros, com limitações e características de operação definidas. Com as premissas estabelecidas, a escolha dos componentes foi realizada, optando entre componentes e tecnologias semelhantes. Em seguida, a organização dos componentes começou a partir do diagrama elétrico. Posteriormente, passou-se para o projeto e testes do [LiDAR](#), sistema de direção e na elaboração do chassi, que unifica todos os sistemas. Por fim, ocorreu a elaboração dos sistemas de controle, comunicação e aquisição de dados.

No capítulo 4 são discutidos os testes e validações pelos quais os sistemas foram submetidos, analisando o desempenho e funcionamento do protótipo.

Por fim, no capítulo 5, os critérios são revisados e ocorre a avaliação do projeto e sugestões para trabalhos futuros

## 2 Fundamentação teórica

A seguir, serão explorados os tópicos relevantes para o embasamento teórico deste trabalho.

### 2.1 Robótica

A relevância da robótica na sociedade contemporânea é um tema crucial que tem sido amplamente explorado nos estudos de especialistas pelo mundo. ([PETER, 2011](#)) destaca a importância crescente da automação e da inteligência artificial na otimização de processos do cotidiano. Sua pesquisa destaca como a robótica aumenta a eficiência e desempenha um papel vital na realização de diversas tarefas, em especial as perigosas ou monótonas, liberando os seres humanos para se concentrarem em atividades mais complexas e criativas.

Na esfera da pesquisa, os estudos na área da robótica impulsionam inovações, proporcionando *insights* sobre como a automação e a inteligência artificial podem transformar radicalmente o modo como conduzimos investigações científicas, além do que estimula a colaboração entre especialistas de diversas áreas, fortalecendo as pesquisas interdisciplinares.

No contexto da robótica móvel, a navegação autônoma emerge como um elemento fundamental, e conforme discutido por ([SIEGWART; NOURBAKHS; SCARAMUZZA, 2011](#)), sua relevância destaca-se especialmente no que diz respeito à capacidade dos robôs de operarem de forma independente em ambientes dinâmicos. Ao enfatizarem aspectos como percepção ambiental e tomada de decisões, é possível viabilizar o desenvolvimento de sistemas robóticos capazes de se locomoverem com eficiência e segurança, com aplicações práticas que podem ir desde a logística industrial até operações de busca e salvamento.

Outro conceito importante para ser abordado nos estudos relacionados a robótica móvel, é o de holonomicidade. Crucial na escolha e no projeto de sistemas de locomoção para robôs, a holonomicidade influencia na seleção de tipos de rodas, mecanismos de direção e estratégias de controle. Ter um sistema holonômico pode facilitar a navegação em ambientes complexos e contribuir para a agilidade e eficiência do robô. Contrastando com os sistemas holonômicos, os sistemas não-holonômicos possuem certas restrições de movimento, que limitam a capacidade de alcançar poses específicas sem realizar manobras complexas. Como exemplo, um veículo com rodas não holonômico pode ter dificuldades em realizar movimentos laterais sem fazer curvas.

Na tabela 1, pode ser visto o comparativo entre os sistemas holonômicos e não

holonômicos:

Tabela 1 – Diferença entre Sistemas Holonômicos e Não Holonômicos

Característica	Sistemas Holonômicos	Sistemas Não-Holonômicos
Movimentação	Possui liberdade para se movimentar em todas as direções do espaço de configuração, podendo atingir qualquer pose desejada sem restrições adicionais significativas	Possuem restrições significativas que limitam sua capacidade de movimentação em certas direções. Essas restrições podem surgir devido a características específicas do sistema, como rodas fixas ou direcionais em um robô
Relações Matemáticas	Restrições cinemáticas holonômicas podem ser expressas sem o uso de derivadas em relação ao tempo, tornando as equações de movimento mais simples. De modo que as restrições são frequentemente expressas como funções algébricas	Envolvem frequentemente derivadas em relação ao tempo. Restrições não podem ser integradas para fornecer uma expressão puramente em termos de variáveis de posição
Exemplo	Veículo com rodas direcionais que podem girar em qualquer direção	Veículo com rodas fixas em posição específica, limitando movimentação

Fonte: Produzido pelo autor, baseado em ([SIEGWART; NOURBAKHSH; SCARAMUZZA, 2011](#))

Uma maneira alternativa de descrever um robô holonômico baseia-se na relação entre os graus de liberdade diferenciais *Differential Degrees Of Freedom* (**DDOF**), de um robô e os graus de liberdade *Degrees Of Freedom* (**DOF**), de seu espaço de trabalho.

**DDOF**, representa o número de variáveis independentes necessárias para descrever totalmente a movimentação de um robô, considerando velocidades associadas. Já o **DOF**, é o número de variáveis independentes necessárias para descrever a posição e orientação de um robô em um determinado instante, sem levar em conta as velocidades associadas.

Um robô é holonômico se, e somente se,  $DDOF = DOF$ . Intuitivamente, isso ocorre porque é somente através de restrições não holonômicas (impostas por rodas direcionáveis ou fixas padrão) que um robô pode alcançar um espaço de trabalho com graus de liberdade que excedem seus graus de liberdade diferenciais,  $DOF > DDOF$ . Exemplos incluem configurações de direção diferencial e bicicletas/triciclos. Em robótica móvel, chassis úteis geralmente devem atingir poses em um espaço de trabalho com dimensionalidade 3, então, em geral, exigimos  $DOF = 3$  para o chassis. No entanto, as habilidades "holonômicas" de manobrar ao redor de obstáculos sem afetar a orientação e rastrear um alvo enquanto seguem uma trajetória arbitrária são considerações adicionais importantes. Por essas razões, a forma específica de holonomia mais relevante para a robótica móvel é aquela de  $DDOF = DOF = 3$ . Essa classe de configurações de robô é definida como omnidirecional: um robô omnidirecional é um robô holonômico com  $DDOF = 3$  ([SIEGWART; NOURBAKHSH; SCARAMUZZA, 2011](#)).

A igualdade  $DDOF = DOF$  é uma característica dos robôs holonômicos, indicando que suas restrições cinemáticas permitem movimentação flexível e fluida no espaço.

Abordagem que é especialmente relevante em robôs omnidirecionais, onde a capacidade de se movimentar em qualquer direção é crucial para manobras eficientes.

### 2.1.1 Car-like e geometrias de direção veicular

Em alguns estudos de robótica móvel, o modelo de bicicleta é frequentemente utilizado como uma simplificação da dinâmica de veicular. Esse modelo considera um robô móvel de quatro rodas, com configuração semelhante aos carros, como uma bicicleta, onde as rodas traseiras são agrupadas em uma única roda fixa e a roda dianteira gira sobre seu eixo vertical ([PETER, 2011](#)). Nesse contexto, surge o termo "*car-like*".

A geometria de Ackermann é um conceito crucial na análise e no design de sistemas de direção para veículos, incluindo robôs móveis. Tal geometria refere-se a uma configuração específica das rodas de um veículo, onde as rodas dianteiras não possuem ângulo de rotação igual entre si. Em um sistema de direção Ackermann, as rodas dianteiras são projetadas para convergir ao ponto chamado "ponto de Ackermann", ou *Instantaneous Centre of Rotation* ([ICR](#)), centro de rotação instantâneo, ao virar. Isso permite que o veículo realize curvas mais suaves e reduz o desgaste dos pneus durante manobras. Essa configuração é especialmente valiosa quando se trata de manobras precisas e curvas apertadas ([SIEGWART; NOURBAKHS; SCARAMUZZA, 2011](#)).

As relações trigonométricas na geometria de Ackermann são fundamentais para entender e modelar o movimento de veículos com direção *car-like*. Para entendimento do modelo, um veículo com rodas dianteiras que podem girar e rodas traseiras fixas será usado como exemplo, conforme pode ser visto na Figura 4:

- $L$  como a distância entre os eixos dianteiro e traseiro do veículo,
- $\gamma$  como o ângulo de viragem das rodas dianteiras,
- $R$  como o raio de curvatura da trajetória seguida pelas rodas traseiras.

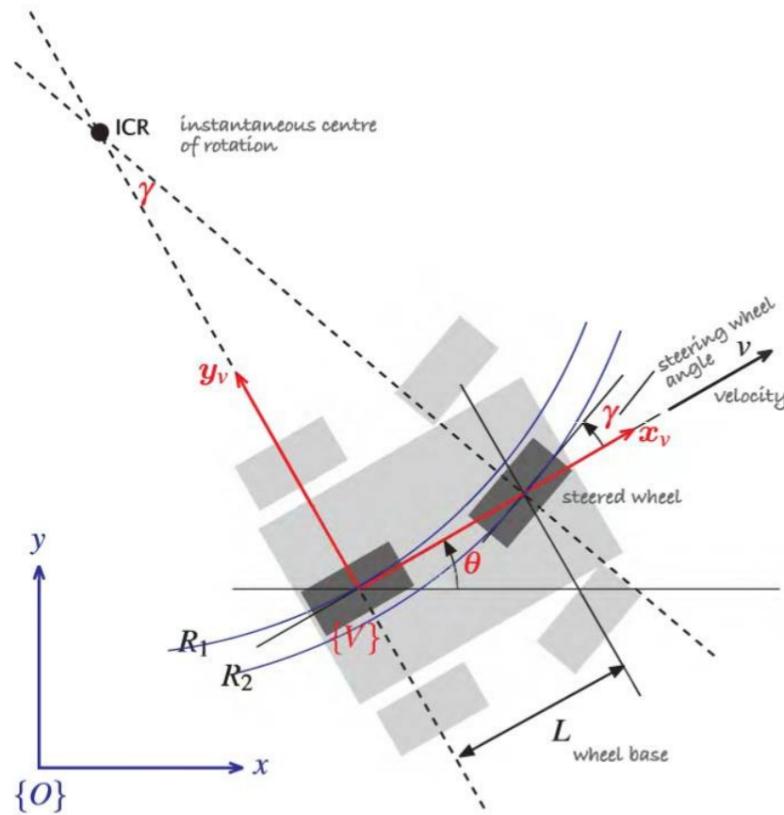
Com base nessas notações, as relações trigonométricas típicas na geometria de Ackermann são:

1. Relação de Curvatura (Relação de Ackermann):

$$\tan \gamma = \frac{L}{R}. \quad (2.1)$$

Essa relação expressa a tangente do ângulo de viragem ( $\gamma$ ) em termos da distância entre os eixos dianteiro e traseiro ( $L$ ) e do raio de curvatura ( $R$ ).

Figura 4 – Geometria de Ackerman

Fonte: ([PETER, 2011](#))

Tendo o raio de curva e sabendo a largura entre os eixos de rotação vertical das rodas dianteiras ( $W$ ), é possível calcular o ângulo de viragem da roda interna ( $\gamma_{int}$ ) e externa ( $\gamma_{ext}$ ) por geometria, resultando em (2.2) e (2.3).

$$\gamma_{int} = \text{atan}\left(\frac{L}{R - \frac{W}{2}}\right). \quad (2.2)$$

$$\gamma_{ext} = \text{atan}\left(\frac{L}{R + \frac{W}{2}}\right). \quad (2.3)$$

### 2.1.1.1 Mapeamento

O mapeamento desempenha um papel crucial na autonomia e eficácia operacional de robôs móveis, capacitando-os a interagir de forma segura e inteligente com o ambiente. É uma ferramenta fundamental para garantir que os robôs possam realizar tarefas complexas e se adaptar a ambientes variados e dinâmicos de maneira autônoma.

Para uma localização precisa, a posição dos pontos de referência (por exemplo, paredes, balizas artificiais, etc.) para auto-localização deve ser medida com precisão

e incluída no sistema de mapeamento. Infelizmente, essa pode ser uma tarefa árdua, dispendiosa e demorada quando o tamanho do ambiente é grande ou quando o ambiente sofre modificações artificiais ou contém objetos dinâmicos.

Avanços recentes tanto na robótica quanto na visão computacional têm gerado resultados na melhoria das técnicas para reconhecimento, mapeamento, criação e modificações autônomas de um mapa ambiental. Naturalmente, os sensores de um robô móvel têm alcance limitado, de modo que o robô deve explorar fisicamente seu ambiente para construir um mapa. Portanto, o robô não só deve criar um mapa, mas também fazê-lo enquanto se move e se localiza para explorar o ambiente. Na comunidade de robótica, isso é frequentemente chamado de problema de *Simultaneous Localization and Mapping* ([SLAM](#)), Localização e Mapeamento Simultâneos. A relevância do problema [SLAM](#) para a comunidade de robótica se deve ao fato de que a solução para esse problema tornaria um robô verdadeiramente autônomo ([SIEGWART; NOURBAKHSH; SCARAMUZZA, 2011](#)).

Métodos como [SLAM](#) permitem que o robô atualize continuamente sua posição no mapa enquanto explora o ambiente, permitindo que os robôs compreendam e interajam de forma autônoma em ambientes complexos e dinâmicos.

Assim, o uso do [LiDAR](#) no contexto deste projeto não apenas aprimora a capacidade de navegação do robô, mas também permite a exploração de tecnologias essenciais para a autonomia e eficácia das operações de veículos robóticos *car-like*.

## 2.2 Sistemas de controle

Sistemas realimentados, na teoria de controle, são sistemas dinâmicos nos quais a saída é constantemente comparada com uma referência, e essa comparação é utilizada para ajustar o comportamento do sistema. Essa prática de realimentação é crucial para corrigir discrepâncias entre a saída atual e a referência desejada, assegurando uma resposta mais precisa e estável. No contexto da teoria de controle, a adoção de sistemas realimentados é essencial para a implementação de estratégias de controle que almejam atingir e manter condições específicas de desempenho, como estabilidade e precisão. Controladores empregam a realimentação para ajustar de forma contínua as ações de controle, otimizando assim o desempenho do sistema em resposta a variações e perturbações.

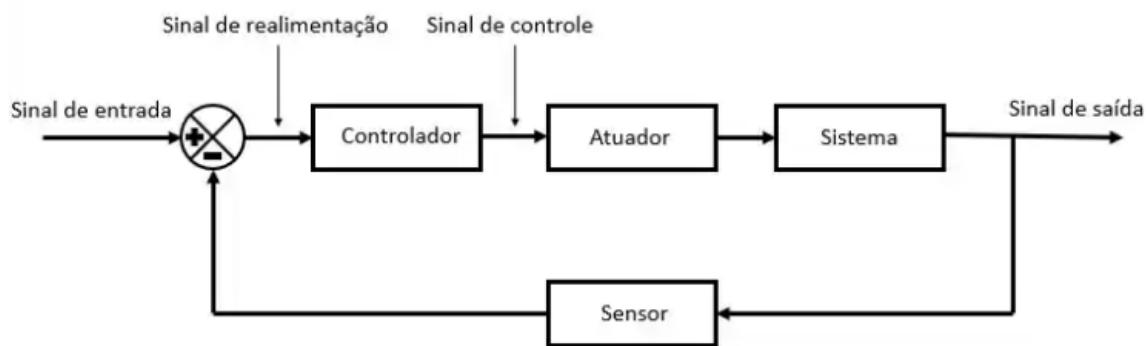
A teoria de controle concentra-se na análise e projeto de sistemas de controle, visando atingir e manter condições de desempenho desejadas, como estabilidade, precisão e eficiência, por meio de técnicas matemáticas. Controladores, amplamente empregados em diversas aplicações industriais, aeroespaciais, automotivas e de processamento, desdobram-se em métodos de projeto lineares, como os controladores PID e LQR, além de outras abordagens, como controle híbrido, adaptativo e robusto. A escolha do método de controle baseia-se nas características do sistema e nas exigências do projeto. Todos esses controladores

compartilham a característica de serem sistemas realimentados, nos quais o controlador emite um sinal ao atuador da planta, utilizando a saída como referência para atingir um valor desejado. Um exemplo prático é o sistema de controle de temperatura, onde a medição ambiental é comparada com o valor de referência, ajustando o sistema de aquecimento. Sistemas realimentados são comumente referidos como sistemas de malha fechada, implicando o uso de controle realimentado para reduzir o erro do sistema ao corrigir a diferença entre a saída e a referência, caracterizando a essência do termo malha fechada ([GIRARDI et al., 2023](#)).

Conforme destacado por ([HENDRICKS; JANNERUP; SØRENSEN, 2008](#)), os elementos primordiais de um sistema de controle compreendem:

- a) A planta do sistema, representando o objeto sujeito a controle;
- b) Os atuadores, dispositivos capacitados a intervir na planta do sistema, como, por exemplo, uma fonte de tensão ou um mecanismo regulador de vazão de um fluido;
- c) Os sensores, incumbidos de monitorar a saída atual, possibilitando ajustes para adequá-la ao valor desejado;
- d) O controlador, responsável por conduzir a planta ao estado desejado.

Figura 5 – Diagrama de controle Clássico



Fonte: ([GIRARDI et al., 2023](#))

Outro aspecto relevante em sistemas de controle com realimentação é a habilidade de rejeição de perturbações. Perturbações, ou distúrbios, constituem entradas indesejáveis que impactam o sinal de saída. A eficácia na rejeição de perturbações varia conforme as características específicas do controlador desenvolvido, sendo este fator contemplado nos requisitos de projeto, influenciando na velocidade de resposta do sistema a tais interferências ([GIRARDI et al., 2023](#)).

## 2.2.1 Sensoreamento

### 2.2.1.1 Sensor de distância

Sensores de distância são dispositivos eletrônicos projetados para medir a distância entre o sensor e um objeto-alvo. Atualmente existem três principais métodos comuns de medição: o sensor ultrassônico, o sensor de triangulação e o de tempo de voo. Cada método possui suas próprias vantagens e limitações, sendo escolhido de acordo com as necessidades específicas da aplicação em questão.

#### 2.2.1.1.1 Sensor Ultrassônico

Os sensores de distância ultrassônicos são dispositivos que utilizam ondas sonoras de alta frequência para medir a distância entre o sensor e um objeto. Esses sensores são comumente empregados em aplicações onde é necessário detectar a presença de obstáculos, realizar medições de distância ou auxiliar na navegação de robôs e veículos autônomos em geral ([ELEKTRA, 2022](#)).

Os sensores ultrassônicos são compostos por dispositivos eletrônicos conhecidos como transdutores ultrassônicos. Esses transdutores desempenham a função de gerar e captar pulsos ultrassônicos em uma frequência específica. O processo de medição é iniciado quando o sensor emite um sinal de pulso sonoro em direção a um objeto. Como resposta a esse estímulo sonoro, o sensor recebe um sinal que corresponde à reflexão dessa onda na superfície do objeto. Ao receber um sinal com a mesma frequência dos pulsos emitidos, é possível calcular a distância até o objeto com base no tempo que os pulsos levaram para ser refletidos de volta pelo objeto ([ALBUQUERQUE; THOMAZINI, 2005](#)).

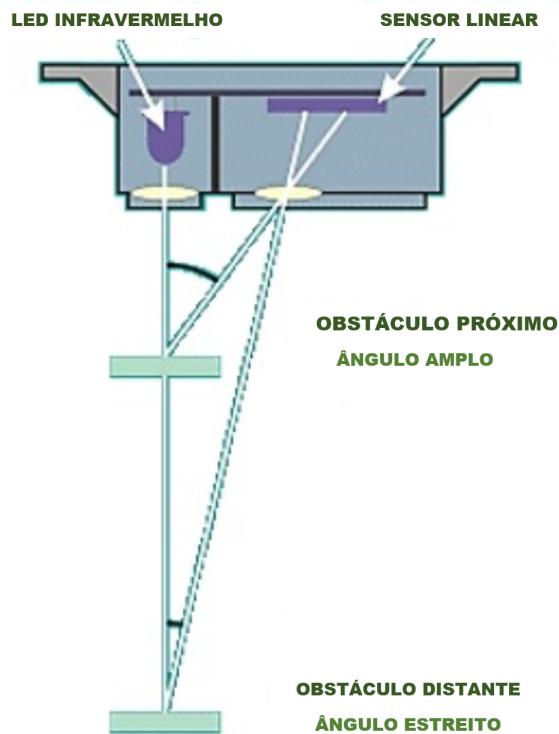
Sensores ultrassônicos não são afetados por cor ou luminosidade, tornando-os úteis em ambientes variados. Entretanto, superfícies absorventes ou irregulares podem afetar a precisão da medição, pois parte do som pode ser absorvida ou refletida de maneira imprevisível. Além disso, variáveis ambientais como temperatura, umidade e presença de ruído têm impacto na velocidade de propagação dos pulsos ultrassônicos, resultando em atrasos no sinal de reflexão e, consequentemente, gerando imprecisões nas medições. Em tais situações, é comum observar que a resposta do sensor não corresponde exatamente ao sinal de reflexão do objeto, uma vez que a presença de ruídos no ambiente pode ocasionar a detecção de ecos falsos ([INDUSTRIAL, 2018](#)).

#### 2.2.1.1.2 Sensor de Triangulação

Sensores de triangulação são dispositivos ópticos que utilizam princípios geométricos para medir distâncias e determinar a posição de objetos. Esses sensores operam com base no conceito de triangulação, que envolve a formação de um triângulo a partir de pontos conhecidos, permitindo a determinação de distâncias e ângulos ([MECHATRONICS, 2016](#)).

O funcionamento básico de um sensor de triangulação envolve a emissão de um feixe de luz em direção ao objeto a ser medido. Esse feixe de luz pode ser um laser ou uma fonte de luz estruturada. O feixe é direcionado para o objeto e, em seguida, é capturado por um sensor óptico, normalmente linear ([MECHATRONICS, 2016](#)). A ilustração do funcionamento do sensor de triangulação pode ser visto na Figura 6 :

Figura 6 – Funcionamento do Sensor de Triangulação



Fonte: ([MECHATRONICS, 2016](#)) - Adaptado pelo autor

Dentre as principais aplicações dos sensores de triangulação, podemos citar: medição de distância de alta precisão; perfilometria; Visão e dimensionamento 3D; e no campo da robótica e navegação, para auxiliar na detecção de obstáculos, especialmente em ambientes dinâmicos. No entanto, esses sensores podem ser sensíveis a condições ambientais, como iluminação, e podem exigir calibração para garantir resultados precisos.

#### 2.2.1.1.3 Sensor de Tempo de voo - ToF (*Time of Flight*)

O sensor de tempo de voo, *Time-of-Flight* (ToF), é um tipo de sensor óptico usado para medir a distância entre o sensor e um objeto, com base no tempo que a luz leva para viajar até o objeto e retornar ao sensor. Esse tipo de sensor aplica pulsos de luz, costumeiramente na forma de luz infravermelha, para medir a distância de forma precisa e eficiente ([KELLER; KOLB, 2009](#)).

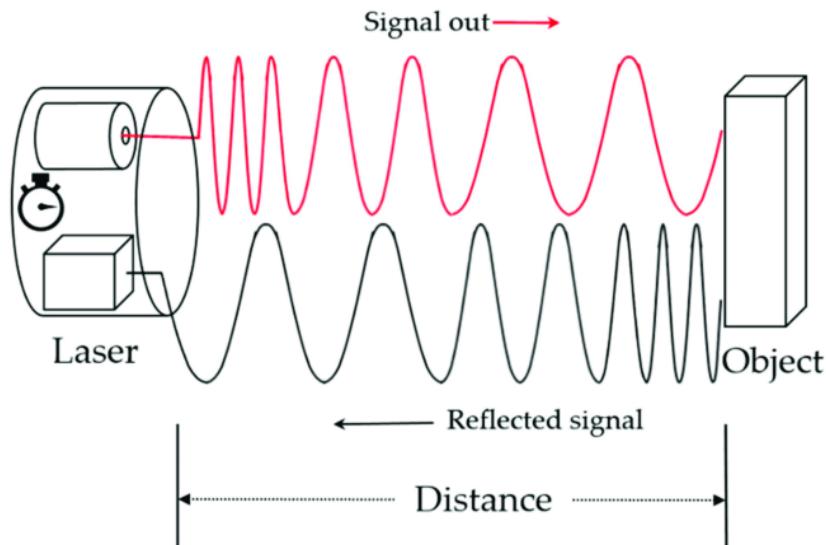
De acordo com (ZHENG et al., 2019), as principais características e funcionamento do sensor de tempo de voo são:

1. Emissão da Luz: O sensor emite pulsos de luz, geralmente na faixa infravermelha, na direção do objeto que se deseja medir.
2. Reflexão da Luz: A luz que incide sobre o objeto é refletida, e retorna ao sensor.
3. Detecção do Tempo de Viagem: O sensor mede o intervalo de tempo entre a emissão do pulso e a detecção do seu retorno.
4. Cálculo da Distância: Com base no tempo de voo da luz e na velocidade conhecida da luz, o sensor calcula a distância até o objeto com base na fórmula (2.8):

$$\text{distância} = \frac{\text{tempo de voo} \cdot \text{velocidade da luz}}{2}. \quad (2.4)$$

O esquemático do funcionamento do sensor ToF pode ser visto na Figura 7.

Figura 7 – Funcionamento do Sensor ToF



Fonte: (ZHENG et al., 2019)

Das diversas aplicações listadas para o sensor ToF, pode-se destacar: navegação e mapeamento robótico; análise de movimento e detecção de objetos dinâmicos; visão computacional e reconstrução 3D; segmentação de imagens para aplicações em robótica; sistemas de controle e monitoramento ambiental; aplicações na área da Biomedicina e Saúde (NAIR et al., 2013) (FUCHS et al., 2010).

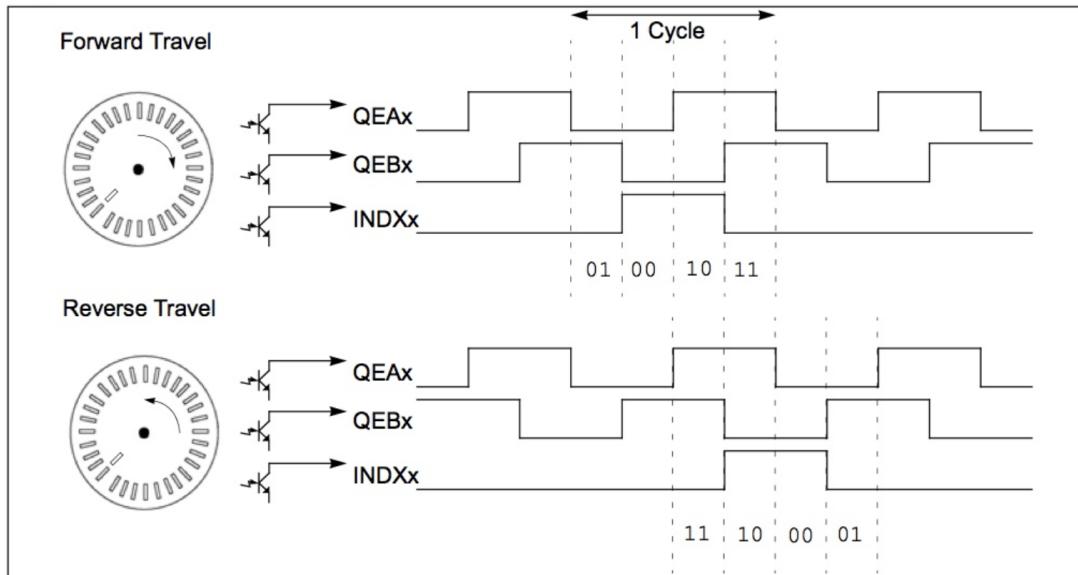
### 2.2.1.2 Encoder

Encoder é um dispositivo sensor que converte movimento rotacional em sinais elétricos, proporcionando informações sobre a posição angular, velocidade ou deslocamento angular de um eixo ou objeto em movimento. Existem dois tipos principais de encoders: encoders absolutos, que fornecem a posição exata em relação a uma referência conhecida, e encoders incrementais, que geram pulsos incrementais à medida que o eixo se move (MATOS et al., 2012). Os encoders são amplamente utilizados em aplicações industriais, robótica, sistemas de controle de motores e outras áreas onde a precisão na medição de posição ou velocidade é crucial para o funcionamento eficiente dos sistemas.

Dentre os tipos de encoders, podemos destacar o encoder de quadratura, que comumente se enquadra na classificação dos encoders incrementais. Este tipo de encoder, gera dois sinais de saída (canais A e B) com pulsos defasados. Cada pulso representa um incremento no movimento angular do eixo ao qual o encoder está acoplado. O uso de dois canais defasados permite não apenas detectar a direção do movimento (avanço ou retrocesso) ao observar a ordem dos pulsos, mas também contar com precisão o número exato de pulsos, resultando em uma medição mais fina da posição ou velocidade angular (REPINALDO, 2015).

Na Figura 8 é possível ver o esquema da interface de sinais de um encoder de quadratura:

Figura 8 – Interface de Sinais do Encoder de Quadratura



Fonte: ([RISCADO, 2013](#))

### 2.2.1.3 LiDAR

**LiDAR**, cujo o termo é uma abreviação do inglês de *Light Detection and Ranging* (Detecção e Varredura por Luz), é um sensor remoto que utiliza pulsos de luz laser para medir distâncias e criar representações do ambiente ao seu redor. O LiDAR é frequentemente empregado em diversas aplicações, incluindo mapeamento topográfico, sensoriamento remoto, navegação autônoma de veículos, e em sistemas de segurança, entre outros (CENTENO; MITISHITA, 2007) (COELHO; VARGAS, 2007).

O funcionamento básico do Lidar envolve o envio de pulsos de laser para o ambiente, e a medição do tempo que leva para esses pulsos retornarem após atingirem alvos dispostos no ambiente. Com base no tempo de viagem e na velocidade da luz, o LiDAR pode calcular as distâncias até os objetos. Ao girar ou movimentar o feixe laser, é possível obter informações tridimensionais detalhadas sobre o ambiente, criando um "mapa" em nuvem de pontos (WAGNER et al., 2004).

O sistema **LiDAR** integra uma fonte de energia intrínseca, sendo nesse caso o próprio laser, que emite radiação eletromagnética em comprimentos de onda curtos (1-10  $\mu\text{m}$ ). Essa emissão é sensível às variações atmosféricas, conferindo ao **LiDAR** o status de método direto na aquisição de dados, o que o classifica como um sensor ativo (ANDERSEN; REUTEBUCH; MCGAUGHEY, 2006).

O **LiDAR** aprimorou a automação em vários setores, do automotivo à logística e armazenamento. Os sensores **LiDAR** são usados na indústria para detecção de obstáculos e aprimoramento de precisão e eficiência e segurança na operação de braços robóticos, empilhadeiras ou equipamentos de transporte, evitando colisões e aprimorando a localização pra operações de precisão.

Sensores **LiDAR** bidimensionais funcionam em apenas um plano e medem distâncias e ângulos em varredura. Seus dados são utilizados principalmente na navegação e detecção de objetos, tanto em ambientes internos quanto externos, já que não dependem das condição de iluminação do local.

O **LiDAR** é de grande valia em aplicações que exigem alta precisão espacial, como em carros autônomos para detecção de obstáculos, em mapeamento topográfico para planejamento urbano, ou em drones para monitoramento ambiental. Sua capacidade de fornecer dados em tempo real o torna uma ferramenta essencial em diversas áreas (SICK, 2013) (Figura 9) .

### 2.2.2 Modelagem de sistemas

Para descrever o comportamento dinâmico de um sistema, é possível empregar uma função de transferência. Essa função descreve como uma entrada modifica a saída de um sistema quando as condições iniciais são nulas (BITTENCOURT, 2007).

Figura 9 – LiDAR SICK



Fonte: ([SICK, 2013](#))

A função de transferência é, por definição, a transformada de Laplace da saída dividida pela transformada de Laplace da entrada, assumindo condições iniciais nulas. Portanto, essas funções são comumente expressas como a razão de dois polinômios em  $s$  como em (2.5).

$$H(s) = \frac{N(s)}{D(s)} = \frac{b_0 + b_1 s + \dots + b_q s^q}{a_0 + a_1 s + \dots + a_n s^n}. \quad (2.5)$$

Em um sistema de ordem elevada, é provável que alguns modos sejam mais significativos para descrever o comportamento do sistema do que outros. Esses modos mais proeminentes são referidos como modos dominantes.

Considerando a função de transferência de primeira ordem expressa como (2.9)

$$H(s) = \frac{k}{\tau s + 1}. \quad (2.6)$$

A resposta ao impulso dessa função pode ser obtida aplicando a transformada inversa da transformada de Laplace da seguinte maneira:

$$h(t) = L^{-1}H(s) = \frac{k}{\tau} e^{\frac{-t}{\tau}}, t > 0. \quad (2.7)$$

Assim, para modelar esse sistema usando apenas dados de entrada e saída, é possível aplicar um sinal degrau à entrada e analisar sua saída. O valor de regime permanente será igual ao parâmetro  $k$ , e o tempo em que a saída atinge 63% desse valor será a constante de tempo  $\tau$ .

Atrasar uma função temporal por  $\tau_d$  unidades de tempo é representado no domínio de Laplace multiplicando a transformada da função por

$$e^{-\tau_d s}. \quad (2.8)$$

Assim uma função completa de primeiro grau pode ser representada por:

$$H(s) = \frac{A}{\tau s + 1} e^{-\tau_d s}. \quad (2.9)$$

### 2.2.3 Controle PID

O método de Controle Proporcional, Integral e Derivativo (**PID**), que representa Proporcional, Integral e Derivativo, é uma técnica amplamente empregada em sistemas dinâmicos para ajustar e estabilizar seu comportamento. Incorporando três componentes essenciais - proporcional (P), integral (I) e derivativo (D) - o controlador PID responde à magnitude do erro atual, acumulação de erros ao longo do tempo e taxa de variação do erro, respectivamente. Essa abordagem é valiosa em sistemas que demandam ajustes dinâmicos, como automação industrial, controle de processos e robótica, proporcionando estabilidade, precisão e rápida resposta transitória.

O método de controle PID, conforme afirma ([FRANKLIN; POWELL; EMAMI-NAEINI, 2015](#)), evoluiu ao longo do tempo, inicialmente com a consideração apenas da parte proporcional. Posteriormente, após estudos mais aprofundados, a inclusão da ação integral revelou-se crucial para eliminar o erro de regime permanente em referências e perturbações constantes. A adição da parte derivativa, também conhecida como termo antecipativo, completou o controlador PID, cuja função de transferência é expressa pela Equação a seguir:

$$C(s) = kp + \frac{ki}{s} + kds. \quad (2.10)$$

Os ganhos do controlador **PID**, impactam diretamente a resposta do controlador às variações no sinal de erro, influenciando a estabilidade e a resposta transitória do sistema de controle. O ganho proporcional,  $kp$ , determina a magnitude da resposta do controlador em relação ao erro, sendo que um valor maior resulta em uma resposta mais forte e rápida, mas pode ocasionar oscilações e instabilidade. Por outro lado, um valor menor proporciona uma resposta mais suave, mas pode resultar em um tempo de resposta mais lento e erro em regime permanente ([GIRARDI et al., 2023](#)).

O ganho integral  $ki$  do controlador PID ajusta a resposta com base no tempo de duração do erro. Um  $ki$  maior proporciona uma resposta mais agressiva para eliminar o erro de regime permanente, mas pode causar oscilações. Um  $ki$  menor resulta em uma resposta mais suave, porém menos eficaz na eliminação do erro. O ganho derivativo  $kd$  ajusta a resposta com base na taxa de variação do erro. Um  $kd$  maior suaviza a resposta do controlador para variações rápidas do erro, mas pode aumentar as oscilações e a instabilidade. Um  $kd$  menor resulta em uma resposta mais suave, mas potencialmente mais lenta. A escolha dos ganhos é um compromisso entre rapidez, estabilidade e suavidade da

resposta, sendo crucial para o desempenho ideal do sistema. O PID, de implementação simples, encontra aplicações em diversas áreas, incluindo o controle de conversores CC-CC, como o conversor boost, essencial em aplicações que demandam um nível de tensão constante (ÖZDEMİR; ERDEM, 2018).

### 2.2.4 Discretização por Tustin

De acordo com (BITTENCOURT, 2007), supondo que os valores dos sinais de entrada e saída de um sistema contínuo sejam medidos a intervalos regulares de  $T$  segundos, e que esse período  $T$  seja suficientemente pequeno, é possível fazer a aproximação discreta da integral de uma função contínua pela soma da área dos trapézios formados a cada período  $T$  por

$$y_k = y_{k-1} + T \frac{u_k + u_{k-1}}{2}, \quad (2.11)$$

que pode ser representado pela transformada z por

$$H(z) = \frac{T}{2} \frac{z+1}{z-1}. \quad (2.12)$$

Sabendo que a integral em Laplace é representada por

$$G(s) = \frac{1}{s}, \quad (2.13)$$

igualando as equações, é possível chegar a

$$s = \frac{2}{T} \frac{z-1}{z+1} \quad (2.14)$$

### 2.2.5 O lugar das raízes

O lugar das raízes é uma representação no plano complexo de todas as posições admissíveis para os polos de um sistema de malha fechada (ARAÚJO, 2007).

As condições são formadas pelo caso onde um sistema  $G(s)$  em série com um controlador  $C(s)$  estiver ligado em malha fechada com realimentação unitária formando a equação (2.15), o ponto só fará parte dos polos do sistema caso o denominador seja igual a zero. Assim, são estabelecidas as condições baseadas em (2.16) para ganho e fase na equações (2.17) e (2.18).

$$G_{MF}(s) = \frac{G(s)}{1 + G(s)H(s)} \quad (2.15)$$

$$1 + G(s)H(s) = 0 \rightarrow G(s)H(s) = -1 \quad (2.16)$$

$$|G(s)H(s)| = 1 \quad (2.17)$$

$$\angle G(s)H(s) = \pm 180^\circ(2k + 1), k = 0, 1, 2, 3, \dots \quad (2.18)$$

Através da formação do diagrama é possível identificar o comportamento do sistema e prever o comportamento das curvas a depender dos polos e zeros do controlador, funcionando como uma ferramenta muito importante para o projeto de controladores tanto contínuos, quanto discretos.

## 2.3 Sistemas embarcados

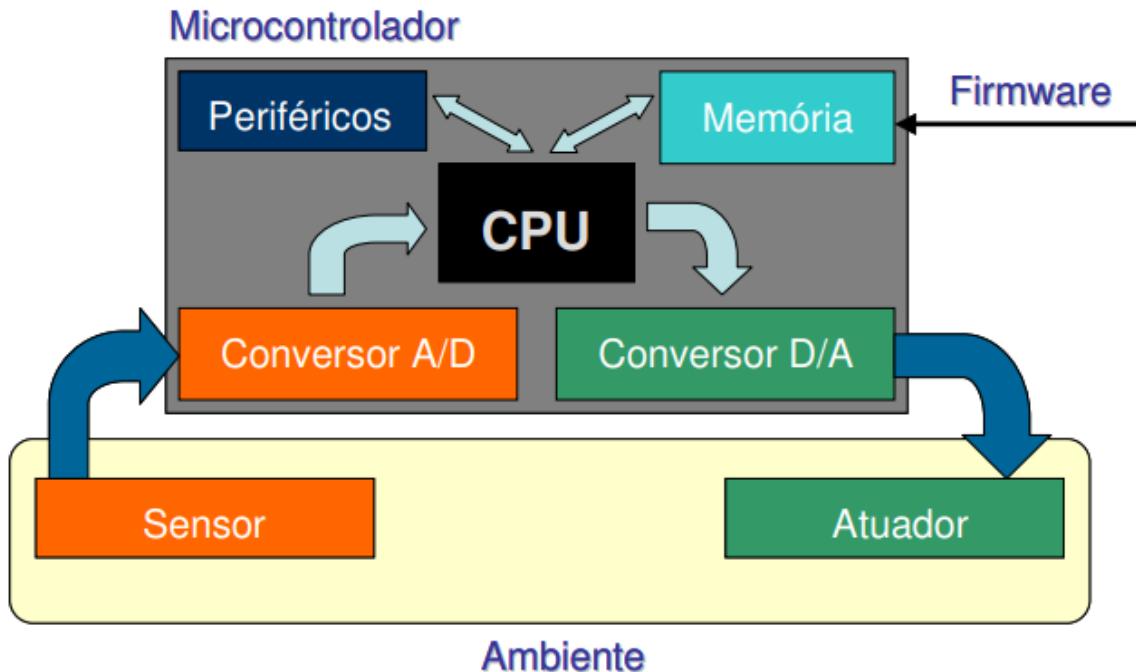
Um sistema embarcado consiste na integração de capacidade computacional em um circuito integrado, equipamento ou sistema, destinada a realizar uma tarefa específica. Diferentemente dos computadores de uso geral, que suportam uma variedade de aplicativos, os sistemas embarcados são concebidos para desempenhar uma única função predeterminada. O usuário final não tem acesso ao programa embutido no dispositivo, mas interage por meio de interfaces, como teclados e displays, se o sistema for projetado para tal (Figura 10) ([CUNHA, 2007](#)).

A nomenclatura "embarcado" (tradução do termo em inglês "*Embedded Systems*") vem do conceito de que esses sistemas são geralmente projetados para operar independentemente de uma fonte de energia fixa, como uma tomada ou gerador. As principais características distintivas desses sistemas incluem sua capacidade computacional e a independência operacional. Outros aspectos relevantes variam conforme os tipos de sistemas, modos de operação e requisitos específicos associados às aplicações embarcadas ([CHASE; ALMEIDA, 2007](#)).

Cada sistema embarcado é constituído por uma unidade de processamento, um circuito integrado anexado a uma placa de circuito impresso. Esses sistemas apresentam capacidade de processamento de informações proveniente de um software que é internamente processado pela unidade, indicando que o software está incorporado na unidade de processamento. Todo software inserido nesse contexto é categorizado como *firmware* (Figura 11) ([BALL, 2002](#)).

Dentro do escopo dos sistemas embarcados, aspectos relevantes incluem diferentes tipos de aplicações. Conforme elencado por ([CUNHA, 2007](#)), destacam-se os tipos de aplicações para os Sistemas Embarcados:

Figura 10 – Exemplo de sistema embarcado utilizando microcontrolador



Fonte: ([CHASE; ALMEIDA, 2007](#))

Figura 11 – Utilização de firmwares em sistemas embarcados



Fonte: ([CHASE; ALMEIDA, 2007](#))

1. Uso cotidiano: Sistemas onde geralmente há uma significativa interação entre com o usuário, frequentemente por meio de terminais de vídeo ou monitores. Semelhantes a computadores de mesa, porém compactados. Exemplos: videogames, conversores de TV, caixas eletrônicos.
2. Sistemas de Controle: Controle em malha fechada, com realimentação em tempo

- real. Utilizados em motores, processos químicos, controle de voo, entre outros.
3. Processamento de Sinais: Processamento eficiente de grandes volumes de informação em curtos intervalos. Exemplos: tratamento de áudio, filtros, modems, compressão de vídeo, radares e sonares.
  4. Comunicações e Redes: Chaveamento e distribuição de informações em sistemas de telefonia, telecomunicações e internet.

Dentre os controladores disponíveis no mercado, pode-se destacar o ESP32, que é um microcontrolador de baixo custo e de baixo consumo de energia, conhecido por suas capacidades avançadas de conectividade, incluindo suporte a Wi-Fi e Bluetooth. O ESP32 é fabricado pela *Espressif Systems* e é amplamente utilizado em projetos de sistemas embarcados, especialmente na área de *Internet of Things (IoT)*. Ele integra um processador de 32 bits, memória flash, Wi-Fi, Bluetooth e diversos periféricos, oferecendo uma solução completa em um único chip ([KURNIAWAN, 2019](#)).

A tabela 2 mostra o comparativo do ESP32 com o Arduino e placas da família Jetson, muito utilizados no mercado para o controle de sistemas embarcados:

Fronte as informações apresentadas, a relevância do ESP32 para sistemas embarcados reside na integração da conectividade avançada, eficiência energética e ampla versatilidade de aplicações que o ESP32 oferece, posicionando esse microcontrolador como uma opção preferencial entre os desenvolvedores em diversas áreas, evidenciando sua capacidade de atender às demandas complexas e variadas inerentes a projetos de sistemas embarcados.

Tabela 2 – Comparativo entre dispositivos controladores

	<b>ESP32</b>	<b>Arduino Uno</b>	<b>Raspberry Pi 3</b>	<b>Jetson</b>
Conectividade	Wi-Fi e Bluetooth integrados, porém pouca disponibilidade de portas USB	Não possui conectividade sem fio integrada	Wi-Fi, Bluetooth e grande disponibilidade portas USB integradas	Oferece grande variedade de opções de conectividade, incluindo Ethernet, USB, HDMI, entre outros
Potência de Processamento	Satisfatória. Processador de 32 bits, porém com capacidade inferior ao Raspberry Pi 3	Processador de 8 bits com velocidade inferior	Alta capacidade de processamento, capaz de executar sistemas operacionais completos como o Linux	Projetadas para desempenho de processamento intensivo, especialmente em tarefas relacionadas à inteligência artificial
Programação	Linguagem Arduino e a IDE Arduino	IDE Arduino (popular por sua simplicidade)	Supora várias linguagens, incluindo Python, e pode executar sistemas operacionais completos	Supora linguagens como CUDA C/C++, Python, TensorRT, OpenCV, Caffe/Caffe2, ROS, e linguagens de programação genéricas como C e C++. (São programadas usando o ambiente de desenvolvimento JetPack, da NVIDIA)
Eficiência energética	Consumo baixo-moderado, sendo mais eficientes em termos de energia	Consumo baixo-moderado, sendo mais eficientes em termos de energia	Consumo baixo-moderado, sendo mais eficientes em termos de energia	Alto consumo energético devido ao seu poder de processamento
Custo	Acessível, com boa disponibilidade no mercado	Acessível	Mais caro que microcontroladores, e com restrições de disponibilidade no mercado	Alto Custo. normalmente, as placas Jetson são mais caras em comparação com microcontroladores como ESP32, Arduino Uno e Raspberry Pi 3
Finalidade	Projetados para aplicações embarcadas gerais, ideal para aplicações IoT e projetos que requerem conectividade sem fio	Ideal para aplicações ligadas ao aprendizado em eletrônica, e iniciativas do tipo "DIY"(do inglês "Do It Yourself")	Versátil, pode ser usado como um microcomputador de baixo custo	Ideal para aplicações de inteligência artificial, aprendizado de máquina e visão computacional avançada

Fonte: Produzido pelo autor, baseado em (AHMED; MARZOG; ABDUL-RAHAIM, 2021) (SÜZEN; DUMAN; ŞEN, 2020) (ARDUINO, 2015)

# 3 Desenvolvimento

## 3.1 Premissas do projeto

Servindo como guia para as escolhas e direcionamentos deste trabalho, foram estabelecidas características mínimas necessárias para proporcionar a obtenção de resultados planejados para o projeto.

As características foram estabelecidas visando à simplificação da modelagem e funcionamento do sistema, porém mantendo sua generalidade e flexibilidade. Foram determinadas de forma a reduzir as incertezas, e atingir o nível de confiança mínimo necessário para permitir a aplicação do projeto em situações diversas.

Foram estabelecidas algumas características básicas relacionadas à velocidade de movimentação, frequência e alcance mínimos do [LiDAR](#), tamanho do protótipo, autonomia, conectividade e funcionamento mecânico.

### 3.1.1 Velocidade de movimentação

Para assegurar a previsibilidade da movimentação e a segurança de que qualquer eventual colisão não danificaria o protótipo, a premissa estabelecida foi de que o veículo não precisa ser rápido, com uma meta de velocidade máxima de apenas 0,5 m/s, sendo possível que se realize tarefas em tempo razoável reduzindo as incertezas da movimentação.

É importante que se tenha completo controle da velocidade, assim, é necessário que se alcance o erro nulo em regime permanente.

### 3.1.2 Características do LiDAR

Para assegurar que as tarefas possam ocorrer de maneira satisfatória foi estabelecida uma meta de velocidade mínima do [LiDAR](#) e distância mínima de leitura, que devem ser de pelo menos 2Hz, com 1m de distância mínima de leitura. Isso é importante para obter dados de ambiente e permitir a tomada de decisões rápidas e precisas.

### 3.1.3 Tamanho do protótipo

O protótipo deve ser projetado de forma a ser imprimível em uma impressora 3D comum com tamanho de construção de 235 x 235 mm. As peças devem se manter nesse limite para diminuir o tempo de fabricação, garantir a portabilidade e reduzir as potências necessária para o projeto.

### 3.1.4 Autonomia

A bateria do projeto deve ter uma autonomia mínima de 15 minutos e ser recarregável. O tempo é importante para permitir a realização das tarefas e a realização de testes diversos, sem que as baterias se esgotem. Devido às dimensões físicas do protótipo, é provável que a característica definida seja facilmente atingida e até mesmo superada, tendo em vista que o consumo energético não será elevado e os modelos de baterias comerciais disponíveis tem capacidade energética satisfatória.

### 3.1.5 Conectividade

Uma premissa importante é a capacidade de estabelecer comunicação com o robô por meio do protocolo *User Datagram Protocol* ([UDP](#)) em uma rede Wi-Fi autônoma gerada localmente. Isso permite a flexibilidade na conexão com diferentes dispositivos devido à popularidade do Wi-Fi, evitando a necessidade de acesso a redes privadas e eliminando a utilização de cabos e a comunicação rápida devido à velocidade de comunicação elevada do protocolo.

### 3.1.6 Suspensão e direção

De acordo com ([SIEGWART; NOURBAKHSH; SCARAMUZZA, 2011](#)) robôs com mais de três rodas necessitam de um sistema de suspensão para manter o contato de todas as rodas com o solo em terrenos irregulares, porém em aplicações *indoor* é possível utilizar pneus macios que servem como uma suspensão primitiva. Assim, no protótipo foram utilizados pneus de borracha com preenchimento em espuma em todas as rodas.

Além da utilização de pneus de borracha, recomenda-se a aplicação do robô sobre uma superfície coberta com folhas de Etil Vinil Acetato ([EVA](#)) para ampliar o efeito de suspensão e o atrito entre o pneu e o solo, especialmente em situações onde a precisão do movimento é crítica.

Quanto à direção de Ackermann, a abordagem adotada foi a utilização de servo motores em cada roda dianteira em vez de recorrer a geometrias trapezoidais convencionais. Essa escolha proporciona maior flexibilidade no controle de cada roda dianteira, permitindo ajustes mais precisos na trajetória do robô. Além disso, essa configuração contribui para simplificar o sistema de direção, aumentando a eficiência operacional do robô.

### 3.1.7 Tração

O sistema de tração vai ser disposto em apenas em uma das rodas traseiras do protótipo. Essa premissa visa à simplificação da modelagem e construção do sistema de tração do carro. Apesar de aumentar as incertezas quanto a derrapagens, a solução vai

evitar a necessidade do uso de um diferencial, componente com muitas partes móveis e de difícil modelagem, e o empacotamento de um segundo sistema nas rodas dianteiras que já se encarregam do sistema de direção.

A baixa velocidade do carro auxilia para que o atrito seja suficiente para manter o pneus na rota desejada. As curvas vão ter que ser compensadas já que o perímetro interno é menor que o externo, mas não envolvem muita complexidade nem tempo de processamento.

## 3.2 Escolha dos componentes

O processo de seleção dos componentes para o projeto, considerou características específicas do robô, requisitos de desempenho, custo e eficiência energética, levando em conta as premissas previamente definidas. A seção aborda a escolha dos sistemas de direção, tração e energia, sensores, plataforma computacional e materiais complementares.

A fim de fornecer uma visão abrangente das escolhas feitas ao longo do processo, uma lista completa de componentes utilizados na construção do robô é apresentada no Apêndice C. Esta lista permite a replicação do projeto, possibilitando futuros melhoramentos.

### 3.2.1 Direção

Para o sistema de direção foram escolhidos um par de micro servos 9g SG90 ([MPJA, 2021](#)) (Figura 12), em cada roda dianteira devido a vários fatores relacionados ao mercado, como custo e disponibilidade, e características de desempenho do motor.

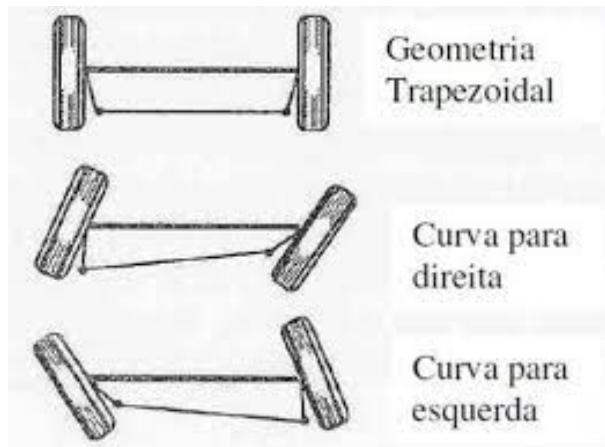
Figura 12 – Servo



Fonte: ([MPJA, 2021](#))

A utilização de servos individuais ao invés de um atuador central e uma geometria de direção (Figura 13) traz a vantagem de conseguir o ajuste do ângulo correto para qualquer nível de esterçamento das rodas o que evita derrapagens dos pneus, auxiliando na modelagem do sistema para as etapas de controle.

Figura 13 – Geometria de direção trapezoidal



Fonte: ([GILLESPIE, 2021](#))

Os servos são dispositivos compactos e versáteis amplamente utilizados em aeromodelismo, robótica e automação.

Uma das principais vantagens dos servos é o seu tamanho e peso pequenos. Isso permite a integração dos servos sem o comprometimento do espaço disponível para outros componentes ou prejudicar a mobilidade do veículo.

Além disso, são conhecidos por sua precisão e simplicidade de controle, através da manipulação da largura de pulsos do nível alto a 50Hz . Isso é importante quando se trata da direção, já que a precisão na orientação do veículo é fundamental para garantir a confiabilidade e previsibilidade do sistema.

Servos funcionam sem a necessidade de módulos de driver ou encoder e apesar da discrepância entre *datasheets* de diferentes fontes, onde a tensão máxima indicada varia de 4.8 a 7.2 V, na prática ([WORKSHOP, 2020](#)), funcionam com alimentação de 6V, comum na alimentação de motores DC que serão utilizados no projeto.

Outro aspecto a ser considerado é a ampla disponibilidade no mercado e baixo custo, o que facilita a aquisição e a substituição em caso de manutenção ou reparos.

### 3.2.2 Motores CC

Para a tração do protótipo e rotação do LiDAR foi escolhido um motor com redução e encoder já integrados ([DFROBOT, 2018](#)) (Figur 14). Os componentes são essenciais

para o controle e leitura da velocidade de rotação e para a elevação do torque no eixo, capaz de acompanhar os comandos de aceleração e desaceleração dos controladores.

Figura 14 – Motor CC



Fonte: ([DFROBOT, 2018](#))

A integração dos componentes é importante para manter os sistemas compactos e diminuir a quantidade de componentes do protótipo.

O motor escolhido possui uma redução de 34,02:1 da rotação do motor para a ponta do eixo, disponibilizando uma rotação de até 210RPM e até 10kg • cm de torque. Esse motor permite que o sistema de tração alcance sua velocidade máxima especificada sem a necessidade de reduções adicionais, simplificando o projeto.

O encoder possui dois canais em quadratura e emite 11 pulsos por revolução do motor antes da redução, permitindo uma resolução alta mesmo em baixas velocidades.

A desvantagem do motor é o peso e o preço elevado comparado a outros modelos do mercado, mas a qualidade construtiva, velocidade final e integração do encoder compensam as desvantagens.

### 3.2.3 Bateria

Para a alimentação do protótipo, foram escolhidas baterias do tipo 18650, mais especificamente o modelo Samsung ICR18650-26F ([SAMSUNG, 2009](#)). Elas apresentam uma série de características que as tornam uma opção vantajosa e eficiente. Essas baterias, seguem o padrão de formato 18mm x 65mm, são células de íon de lítio recarregáveis, com tensão média de 3,7V e podem ser ligadas em série ou paralelo para fornecer a tensão e corrente necessária para o projeto.

Essas baterias destacam-se pela sua alta capacidade de armazenamento, corrente máxima de descarga e segurança de operação. As baterias possuem sistemas de proteção

contra curto-circuito, sobrecarga e descarga excessiva, garantindo a segurança durante o uso do robô além de uma boa autonomia e a garantia do bom funcionamento dos sistemas.

Além disso, o formato compacto e cilíndrico das baterias facilita a integração no design do robô contribuindo para a otimização do espaço disponível, assegurando eficiência na disposição dos componentes e garantindo uma estética mais agradável.

Além dos aspectos de eficiência e design, a decisão também considera a durabilidade e a facilidade de manutenção das baterias. Sua vida útil, com perdas de 30% de capacidade em 300 ciclos, reduz a necessidade de substituições frequentes, aumentando a confiabilidade do sistema. A ampla disponibilidade dessas baterias no mercado facilita a obtenção e substituição em caso de necessidade.

Utilizando o ponto de consumo médio dos motores como maiores consumidores de potência, e as informações de autonomia para o intervalo de consumo das baterias, é possível estimar o tempo de autonomia das baterias através de (3.1), sendo  $t$  a autonomia em horas,  $C_{min}$  a capacidade mínima em  $mAh$ ,  $\eta$  a capacidade relativa ao fluxo de descarga em % e  $I$  o consumo em  $mA$ , superando o facilmente as metas de autonomia.

$$t_h = \frac{C_{min} \cdot \eta(I)}{I} \rightarrow \frac{2550mAh \cdot 0,95_{0,5C}}{2 \cdot 600mA + 100mA} = 1,86h. \quad (3.1)$$

### 3.2.4 Plataforma computacional

As opções de sistemas computacionais considerados para o processamento do protótipo, foram as placas Jetson da NVIDIA, Raspberry Pi e o ESP32.

As placas Jetson, conhecidas pelo poder computacional, são comuns em aplicações de controle e inteligência artificial. Contudo apresentam um custo muito elevado em comparação às outras opções.

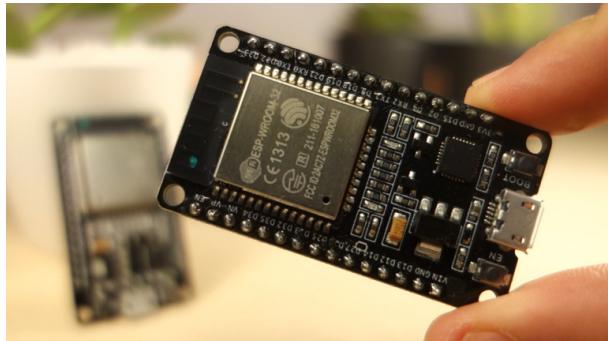
O ESP32, conhecido por sua eficiência energética, preço acessível e integração de Wi-Fi e Bluetooth, oferece um equilíbrio mais adequado entre desempenho e economia, comparado às placas Jetson. O ESP32 atende de maneira satisfatória aos requisitos computacionais do projeto, ao lidar com algoritmos para processamento de dados provenientes de um [LiDAR](#) 2D de baixo custo.

A Raspberry Pi, historicamente uma escolha popular em projetos de computação embarcada, enfrentou desafios de disponibilidade de mercado e aumento de preço devido à escassez provocada pela pandemia de COVID-19. Isso aumentou o apelo do ESP32 como uma alternativa viável para o projeto.

Assim, uma placa de desenvolvimento baseada em ESP32 ([ESPRESSIF, 2023](#)) (Figura 15) foi escolhida como plataforma computacional para o protótipo. A decisão foi guiada pela necessidade de equilibrar desempenho computacional e acessibilidade

econômica e mercadológica, tendo em vista as limitações causadas pela pandemia.

Figura 15 – Placa de desenvolvimento ESP32



Fonte: ([TUTORIALS, 2018a](#))

### 3.2.5 Sensor de distância

O sensor de distância escolhido para constituir o **LiDAR** foi o modelo VL53L0X ([STMICROELECTRONICS, 2022](#)). Com ele foram comparados os sensores de distância mais disponíveis no mercado hobista, o sensores Sharp, o VL53L1X e os modelos de sensores ultrassônicos.

A escolha do VL53L0X em detrimento dos outros sensores levou em conta principalmente seu custo, tamanho, consumo e taxa de amostragem, além da resolução da medição.

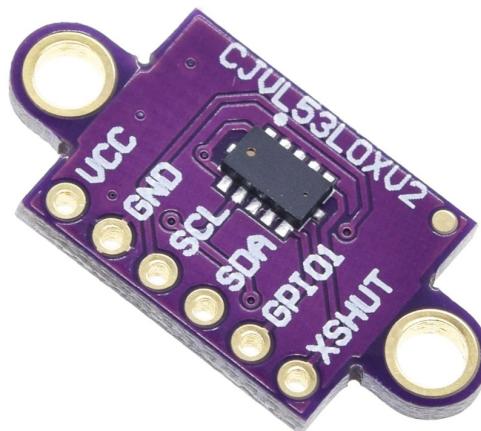
O VL53L0X, desenvolvido pela STMicroelectronics, é um sensor do tipo **ToF** compacto e preciso. Possui uma resolução de até 1 mm e é capaz de medir distâncias de até 2 metros a uma velocidade de até 50Hz, ótimo para situações que demandam precisão sem a necessidade um alcance muito grande.

Comparativamente, os sensores Sharp, embora sejam amplamente utilizados e ofereçam boas características de precisão, apresentam um consumo de energia mais elevado, custam mais caro e podem ser sensíveis a variações nas condições ambientais. Já o VL53L1X, também da STMicroelectronics, é uma versão mais avançada do VL53L0X, com maior alcance, porém, com custo também mais elevado e maior complexidade não justificam plenamente sua inclusão no projeto.

Quanto aos sensores ultrassônicos, embora sejam amplamente utilizados em aplicações de medição de distância, possuem algumas limitações, como uma taxa de amostragem mais baixa e uma resposta mais lenta em comparação com os sensores baseados em tecnologia **ToF**. Além disso, a propagação do som pode ser afetada por condições ambientais, o que pode comprometer a confiabilidade das medições.

Assim, a escolha do VL53L0X para a implementação do LiDAR (Figura 16) teve como objetivo aprimorar a precisão, a eficiência energética e a estabilidade das leituras de modo a satisfazer os requisitos mantendo os custos baixos.

Figura 16 – Módulo sensor VL53L0X



Fonte: ([CIRCUITO, 2023a](#))

### 3.2.6 Outros componentes

#### 3.2.6.1 Sistema do LiDAR

Devido ao encoder do motor ser do tipo incremental e à redução adicionada ao sistema de rotação, não é possível identificar a posição inicial do LiDAR, recorreu-se então à implementação de um sensor de volta. O sistema foi desenvolvido empregando um sensor de obstáculo ([CIRCUITO, 2023b](#)) (Figura 18) motivado pela sua sensibilidade ajustável, facilidade de acesso no mercado e custo acessível.

Foi adicionado também um anel rotativo ([SENRING, 2023](#)) (Figura 17) para permitir a rotação do sensor de distância sem que os cabos se torçam. O anel possui 6 vias e suporta uma velocidade de rotação de até 300RPM, superior às especificações de projeto.

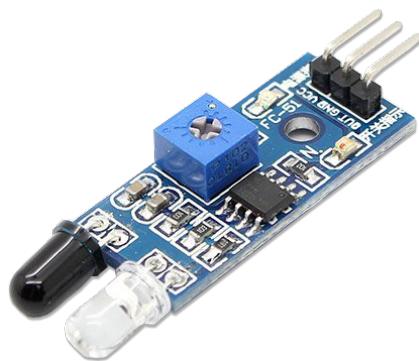
#### 3.2.6.2 Sistema de alimentação

Para a alimentação do projeto foram selecionados componentes complementares ao sistema. Foram utilizados um suporte duplo para baterias com tampo e chave, um fusível, um indicador de bateria e um módulo regulador de tensão ajustável.

Figura 17 – Anel rotativo

Fonte: ([SENRING, 2023](#))

Figura 18 – Sensor de obstáculo

Fonte: ([CORE, 2023](#))

O suporte duplo para baterias com chave desempenha um papel crucial na organização e segurança das células de energia, oferecendo um controle prático sobre o fornecimento de energia.

Um fusível de 3,5A foi integrado ao circuito, age como uma salvaguarda, prevenindo possíveis danos em casos de sobrecarga ou curto-circuito, assegurando a integridade do sistema. Seu valor foi definido levando em consideração a corrente de máxima potência dos motores (1.1A) como maiores consumidores, dando uma margem para consumo dos

outros componentes e assegurando margem para a corrente de partida dos motores (3,2A).

O indicador de bateria desempenha um papel importante ao oferecer uma visualização imediata do estado da carga, proporcionando uma gestão eficaz da autonomia do robô móvel evitando que as baterias sejam descarregadas excessivamente.

O módulo regulador de tensão ajustável é fundamental para manter uma tensão de saída estável, mesmo à medida que as baterias se descarregam durante o uso, assegurando que os componentes do robô recebam a tensão adequada para operação.

### 3.2.6.3 Módulo ponte H

Um módulo driver duplo com ponte H baseado no Circuito Integrado ([CI](#)) L298N ([STMICROELECTRONICS, 2000](#)) foi selecionado para controlar tanto o sistema de tração quanto o [LiDAR](#) do robô móvel. Esse componente oferece a capacidade de controlar motores de corrente contínua bidirecionais, sendo fundamental para o funcionamento eficaz do sistema de tração. A escolha do L298N baseia-se em sua acessibilidade, preço e pela facilidade de uso e implementação, tornando-o uma opção prática para o controle dos motores.

## 3.3 Diagrama elétrico

O desenvolvimento do diagrama das conexões elétricas do protótipo baseou-se pela tensão de alimentação dos componentes. O módulo ESP32 e o driver dos motores Corrente Contínua ([CC](#)) possuem reguladores de tensão internos e podem ser alimentados com até 20V. Os motores [CC](#) e servo motores possuem tensão de operação de até 6V e o sensor de distância e encoders 3,3V.

A solução idealizada para suprir a energia do sistema foi a utilização de um regulador de tensão estabilizando a tensão das baterias e alimentando os servos, o driver e o ESP32, e este fornecendo, através de seus reguladores internos, a tensão de 3,3V para o sensor de distância e encoders.

Assim, para suprir tensão de 6 V duas baterias foram utilizadas em série, fornecendo uma tensão contínua no intervalo de 5,9V a 8,4V de acordo com a carga da bateria. As baterias foram acomodadas no suporte duplo com chave liga/desliga para comandar a alimentação do circuito, e foi adicionado um fusível de vidro 5x20mm no espaço livre do suporte das baterias para a proteção contra curto-circuitos e sobrecarga. O regulador foi ajustado para fornecer 6V na saída e o ESP32, servos e o driver dos motores foram conectados ao regulador.

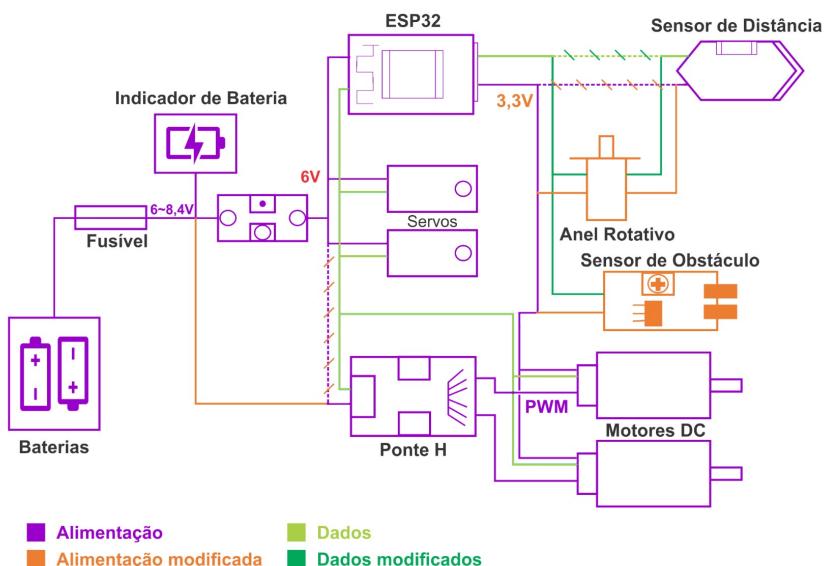
Todos os sinais de comando e alimentação dos sensores foram centralizados no ESP32. A tensão de operação do controlador é 3,3V, assim, através de um dos pinos da

placa é possível ter acesso ao regulador interno e a aproveitá-lo para a alimentação de outros componentes. Como o controlador opera a 3,3V suas saídas digitais não ultrapassam essa tensão, o que pode trazer incompatibilidades com alguns componentes que operam em tensões de 5V normalmente, demandando a utilização de circuitos adaptadores.

No protótipo, tanto os encoders quanto o sensor de distância operam nominalmente a 3,3V, restando examinar a operação dos servos e o driver dos motores com tensões menores, assim, de acordo com (WORKSHOP, 2020) e (TUTORIALS, 2018b), os sinais de comando poderiam ser ligados diretamente sem a necessidade de demais componentes.

Com isso, o diagrama inicial pôde ser projetado de acordo com a Figura 19. Contudo, após a etapa de projeto inicial e a impressão do chassis, nos testes iniciais algumas modificações foram demandadas.

Figura 19 – Diagrama elétrico



Fonte: o autor, 2023

A primeira modificação foi a inclusão do anel rotativo para fazer a ligação do sensor de distância ao ESP32 e a inclusão do sensor de obstáculos como sensor de volta para referenciar o [LiDAR](#), sua tensão de operação é compatível com a tensão de 3,3V e não precisou de adaptações.

A outra modificação foi a troca da alimentação do driver dos motores. A ponte H possui uma queda de tensão de 2V a 4V na saída, com a alimentação de 6V, a queda de tensão estava reduzindo muito a velocidade final dos motores. Assim, a alimentação do driver foi modificada para ser fornecida diretamente pelas baterias ao invés do regulador.

As ligações dos componentes ao ESP32 respeitaram o funcionamento de cada pino, visto que alguns cumprem papéis especiais nas sequências de *boot* do [CI](#) ou nos protocolos

de comunicação.

As designações se basearam na tabela criada por ([TUTORIALS, 2018a](#)), que resume o tipo de operações de cada terminal. O sensor de distância foi alocado aos pinos necessários para a comunicação via protocolo I2C e as demais conexões foram feitas evitando os pinos com funções especiais (Figura 20).

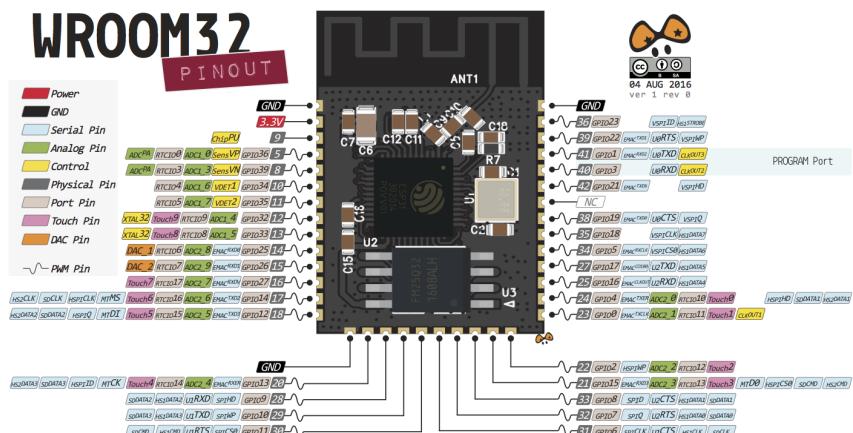
Assim, com as modificações introduzidas e os pinos alocados, a Tabela 3 e o diagrama da Figura 19 puderam ser definidos.

Tabela 3 – Alocação de portas

Função	GPIO
Enable - Motor de tração	4
Direção 1 - Motor do LiDAR	13
Led de debug	14
Direção 1 - Motor de tração	16
Direção 2 - Motor de tração	17
Direção 1 - Motor do LiDAR	18
Enable - Motor do LiDAR	19
i2C SDA - Sensor de distância	21
i2C SCL - Sensor de distância	22
Reset - Sensor de distância	23
Encoder 1 - Motor de tração	25
Servo da direita	26
Servo da esquerda	27
Encoder 1 - Motor do LiDAR	32
Encoder 2 - Motor de tração	33
GPIO - Sensor de distância	34
Encoder 2 - Motor do LiDAR	35
Sensor de volta	39

Fonte: o autor, 2023

Figura 20 – Pinagem do ESP32



Fonte: ([TUTORIALS](#), 2018a)

## 3.4 Modelagem 3D

Todo o processo de modelagem 3D foi feito no Blender utilizando formas primitivas, como cubos e cilindros, servindo de fundação para as peças mais complexas. Foram utilizadas ferramentas de edição de malha como translação, rotação, escala, extrusões, chanfros e operações de soma e subtração entre peças, além de transformações diretas de faces, arestas e pontos dos modelos.

O Blender é uma ferramenta que abrange toda a cadeia de desenvolvimento 3D, desde modelagem até captura de movimento para jogos a animações, sendo uma solução de código aberto e gratuita. Funciona em diversos sistemas operacionais, incluindo Linux, Windows e Macintosh. Uma das características notáveis do Blender é sua capacidade de modelar formas simples e orgânicas, as ferramentas de modelagem do Blender oferecem uma ampla gama de opções para atender às necessidades dos usuários. Essa acessibilidade, aliada à sua comunidade ativa, faz do Blender uma escolha popular para aqueles que buscam uma solução abrangente e gratuita para suas necessidades de modelagem 3D ([BLENDER, 2023](#)).

Como estratégia de modelagem, todos os componentes previamente selecionados foram modelados para estabelecer a referência necessária para a criação dos pontos de fixação e encaixe entre os componentes, utilizando folgas de 0,125 mm nas peças de encaixe por interferência.

O processo de modelagem foi iniciado a partir de testes envolvendo o [LiDAR](#) por ser identificado como ponto crítico do projeto, em seguida foram desenvolvidos protótipos para o sistema de direção e por fim, a união dos sistemas com a modelagem do chassi.

### 3.4.1 LiDAR

A ideia inicial da implementação do [LiDAR](#) foi a rotação do laser emitido pelo sensor de distância através da reflexão deste em um espelho giratório com incidência de 45°

A solução da reflexão do feixe de laser precisava ser validada para a continuidade do desenvolvimento do projeto.

Os primeiros testes foram feitos apenas para comprovar a possibilidade de reflexão do feixe de laser e o nível de interferência de uma película de *smartwatch* que poderia ser utilizada como proteção do sensor. Foi desenvolvido um suporte que garantisse o ângulo de 45° para a utilização de um espelho qualquer com um encaixe para a película. O modelo do suporte pode ser visualizado na Figura 21, com destaque para o sensor em laranja.

Nos testes iniciais ficou comprovado o funcionamento da reflexão do feixe por espelhos e a sua completa obstrução por parte da película que acabou por ser descartada.

Figura 21 – Suporte de validação da reflexão do laser, sensor destacado em laranja



Fonte: o autor, 2023

O segundo modelo foi desenvolvido já validando o sistema de rotação e o posicionamento do espelho no anel superior.

As peças foram modeladas tendo o componente sensor do módulo VL53L0X como centro para evitar a dependência do ângulo de posicionamento para o ajuste de *offset* dos dados lidos.

Uma base fixa foi projetada levando em consideração os parafusos de fixação e a posição dos pinos de comunicação do modulo, mantendo o sensor fixo como centro do sistema.

Um espelho de 20x20x4mm foi posicionado no anel superior com o rolamento servindo de união entre os componentes formando o angulo de 45º em relação ao sensor.

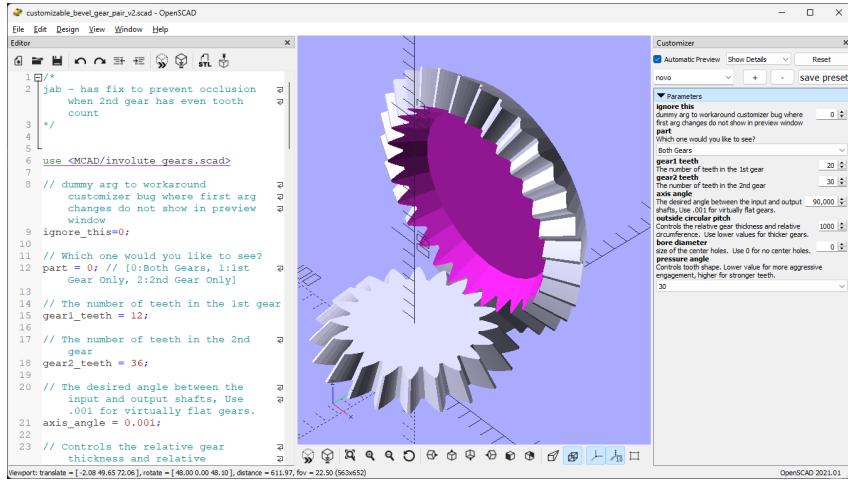
Utilizando a o código criado por ([JIMB15122, 2019](#)) para o OpenSCAD foram geradas engrenagens cônicas para fazer a ligação entre o motor e o anel superior (Figura 22). As engrenagens servem para reduzir a velocidade do motor e para fazer a rotação do movimento em 90º e facilitar o encapsulamento dos componentes.

As engrenagens exportadas do OpenSCAD foram modificadas para se ajustar à escala dos outros componentes e a engrenagem secundária foi posicionada de forma concêntrica ao rolamento aplicando todas as transformações à engrenagem primária para manter o espaçamento do conjunto.

Com as engrenagens posicionadas, o eixo do motor foi alinhado com a engrenagem primária.

As distâncias entre os componentes já alocados foram ajustadas para que o feixe de laser não fosse obstruído e os suportes e encaixes foram projetados mantendo o espaçamento

Figura 22 – Gerador de engrenagens no OpenSCAD



Fonte: o autor, 2023

o padrão de folgas de 0,125mm.

A engrenagem secundária foi projetada sobre trilhos de forma que pudesse ser trocada sem a necessidade de refazer todo o anel superior caso a relação de engrenagens tivesse que ser ajustada tendo seu movimento travado pelo encaixe do rolamento.

No anel superior do [LiDAR](#) foram introduzidos dois furos onde seria possível inserir parafusos para ajustar o centro de massa do sistema caso existisse muita vibração devido à rotação.

O projeto do suporte pode ser visualizado na Figura 23 com destaque para sensor e o espelho em laranja.

### 3.4.2 Testes iniciais e ajustes

Com o sistema projetado, as peças foram impressas (Figura 24) e testadas em bancada.

Após os testes foi identificada uma limitação do sistema para a leitura das distâncias fora do alcance do sensor.

Enquanto com obstáculos próximos, o sensor lia a distância corretamente, mas a partir de um limiar, que dependia da reflexibilidade do objeto, o valor da leitura diminuía conforme o objeto se afastava até um limite próximo de 30mm, quando o sensor deveria indicar 8190mm como indicador de ausência de obstáculos.

O problema foi causado pelo tamanho e distância do espelho. O sensor lê distâncias corretamente quando refletido em espelhos maiores e com maior afastamento. porém em distâncias pequenas, com o espelho de apenas 20x20mm, não consegue ignorar a existência

Figura 23 – Suporte de validação do LiDAR com espelho, espelho e sensor destacados em laranja



Fonte: o autor, 2023

Figura 24 – Suporte de validação impresso



Fonte: o autor, 2023

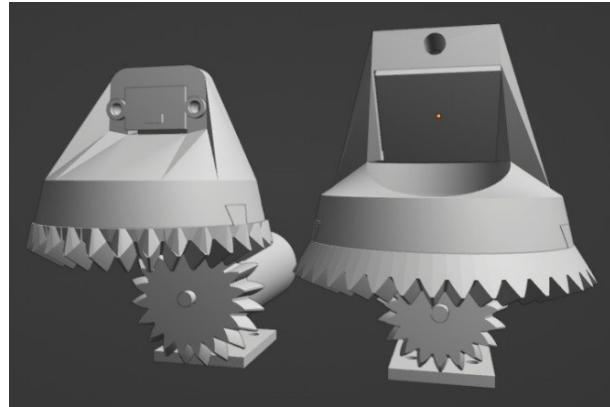
do espelho como obstáculo.

Para solucionar o problema, o sistema de reflexão do feixe de laser foi substituído por um sistema que rotaciona o próprio sensor. Foi adicionado um anel rotativo de 6 vias que através de escovas mantém os contatos dos fios que entram e dos fios que saem sem que eles se torçam a uma velocidade de até 300RPM.

O anel rotativo foi adicionado e o sensor foi realocado na modelagem onde as peças

foram refeitas, impressas e testadas, agora com a rotação e leitura corretas. Na Figura 25 pode ser visto o sistema novo à esquerda e o antigo à direita.

Figura 25 – Mudança do sistema do LiDAR



Fonte: o autor, 2023

Com o sistema de medição corrigido, os testes de ajuste de velocidade e leitura do ângulo que foram feitos já no chassi, após os testes do sistema de direção.

Durante os testes de leitura do ângulo ficou evidente a falta de referencial de posição inicial do sensor, o que acarretou em mais uma modificação no sistema. Um suporte para um sensor de obstáculos foi adicionado ao motor e uma lingueta foi adicionada ao fundo do anel superior do LiDAR (Figura 26), podendo assim, identificar a passagem da lingueta no sensor e usar a posição como referencial para a contagem dos pulsos do encoder do motor.

### 3.4.3 Sistema de Direção

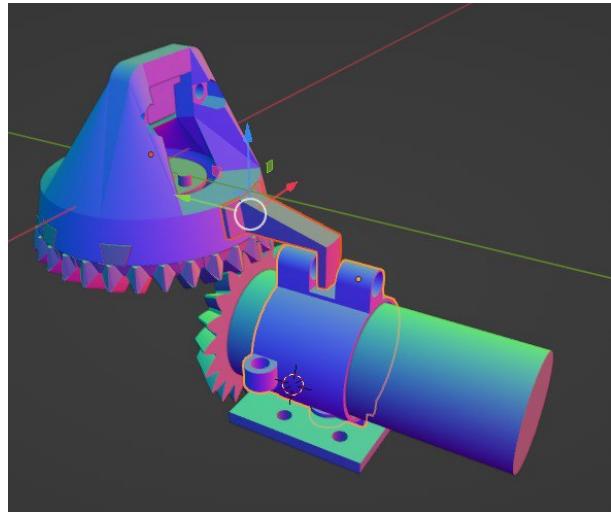
Como foi optado por não utilizar suspensão, e para evitar a complexidade da modelagem do sistema de direção, o eixo de rotação vertical das rodas dianteiras foi posicionado no centro das rodas, ortogonalmente ao chão.

As rodas foram modeladas de forma a comportar os pneus de 64mm com encaixes para dois rolamentos de 10x5x4mm (diâmetro externo, diâmetro interno e espessura) para garantir uma melhor estabilidade. O encaixe é feito direto na manga de eixo e fixadas por um parafuso M3 junto a uma bucha.

A manga de eixo possui acessos para o parafuso de fixação da roda com uma porca embutida e para o eixo vertical de direção. O posicionamento angular é feito através de uma engrenagem ligando a manga de eixo ao servo motor no chassi.

O sistema de teste foi projetado para alcançar uma rotação de  $\pm 25^\circ$  máximos. A engrenagem foi feita utilizando o mesmo código de OpenSCAD do LiDAR, mas sem

Figura 26 – Sistema de detecção de volta



Fonte: o autor, 2023

rotação da transmissão e com relação 1,57 (55 e 35 dentes) para aproveitar um pouco mais do curso de 180° do servo, ganhando mais precisão, mas sem perder muita velocidade de movimentação.

Na Figura 27 é possível ver a modelagem do sistema de teste de direção, com destaque para a engrenagem primária, o braço de suporte da manga de eixo, a manga de eixo e a bucha de trava da roda.

O sistema foi impresso para testes (Figura 28) e sem muitos problemas, o projeto foi incorporado ao chassi com alguns reforços e permitindo uma maior amplitude de movimentação,  $\pm 35^\circ$ .

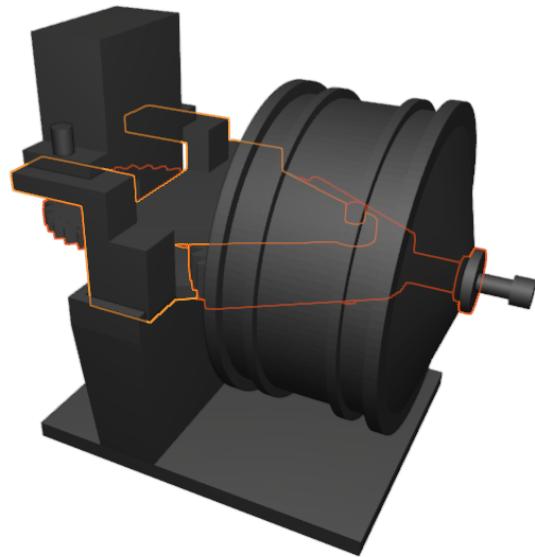
#### 3.4.4 Chassi

O chassi foi feito tomando o LiDAR como centro e as rodas como extremos, mantendo as dimensões dentro da área necessária para que fosse impresso na mesa de 235x235mm.

O sistema de direção foi posicionado e espelhado para as duas rodas dianteiras, fazendo os ajustes na angulação dos braços de fixação e proximidade do espaçamento das rodas para comportar a nova angulação máxima, aliados a reforços na parte superior e inferior dos braços para evitar flexões.

Os componentes foram distribuídos levando em conta o esquemático elétrico e o balanceamento do peso, focando em manter o peso ligeiramente na parte traseira para aumentar a tração, que poderia ter um desempenho ruim por ser unilateral.

Figura 27 – Sistema de teste de direção, engrenagem primária, braço de suporte e manga de eixo destacados em laranja



Fonte: o autor, 2023

Figura 28 – Sistema de teste de direção impresso



Fonte: o autor, 2023

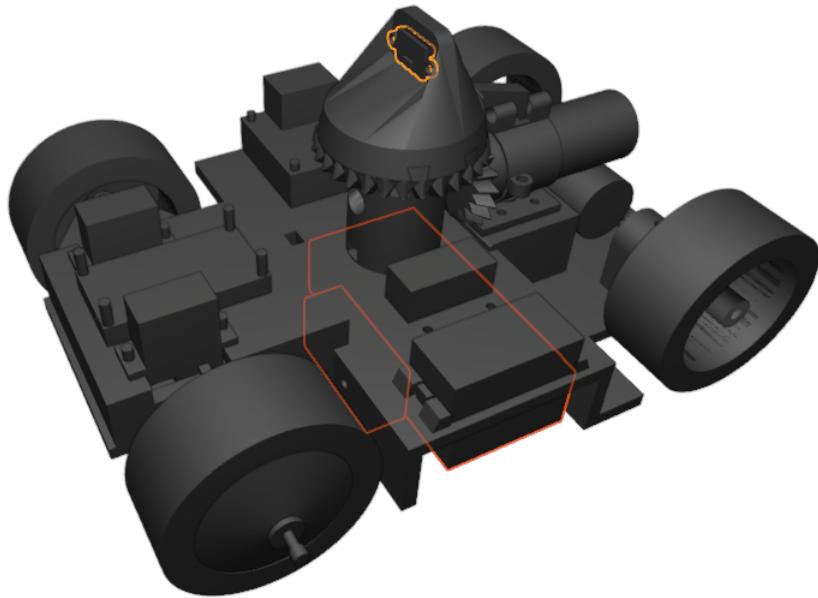
Devido a isso, o motor do **LiDAR** foi posicionado para a traseira que também comportaria o motor de tração, assim, o suporte das baterias foi posicionada na parte central no lado inferior do chassi, onde poderia ser aberto para a troca e recarregamento das baterias, teria a chave acessível e poderia distribuir energia para ambos os setores do carro, sem obstruir a torre do **LiDAR**.

O motor de tração foi ligado diretamente à roda, no lado aposto às baterias para equilibrar o peso, garantindo uma velocidade máxima de até 0,69 m/s, suficientes para cumprir os requisitos de velocidade do projeto, enquanto a roda oposta foi fixada em dois rolamentos encaixados no chassi, fixada por um parafuso M4 com uma bucha adaptadora.

Os componentes eletrônicos foram distribuídos pensando na passagem do cabeamento, com o regulador de tensão e mostrador de baterias próximos ao suporte de baterias, o driver dos motores na parte central e o ESP32 na parte frontal, próximo aos servos e contatos do anel rotativo.

A altura do **LiDAR** foi ajustada para que nenhum componente entrasse no raio de visão do sensor e as fixações e encaixes foram modelados. Na Figura 29 é possível ver a modelagem do sistema, com destaque para suporte das baterias, o regulador de tensão e o sensor.

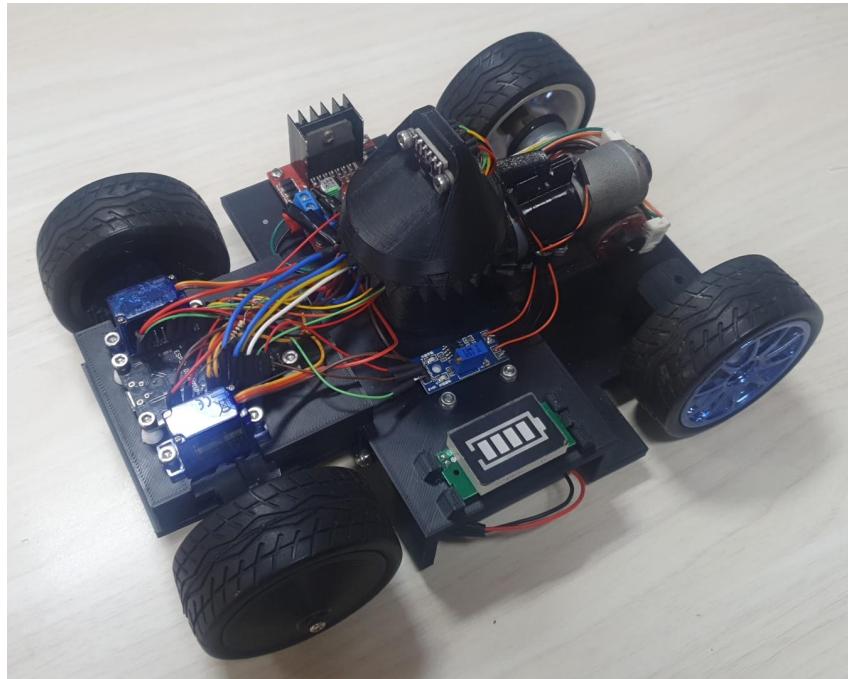
Figura 29 – Projeto completo, sensor, suporte das baterias e regulador de tensão destacados em laranja



Fonte: o autor, 2023

Assim, todos os componentes foram impressos e o protótipo foi montado (Figura 30).

Figura 30 – Projeto completo impresso



Fonte: o autor, 2023

## 3.5 Aquisição e controle

### 3.5.1 Comunicação

O protótipo foi programado para gerar uma rede Wi-Fi própria com nome especificado como "UDP\_Comm" que permite apenas uma conexão com IP fixo, para simplificar as configurações de conexão. Os IPs gerados são: 192.168.4.22, para o *host*; e 192.168.4.25 para o dispositivo conectado.

O protocolo **UDP** é suportado por diversos tipos de dispositivos, assim a comunicação estabelecida foi baseada em comandos simples do tipo string contendo um comando e um valor auxiliar, como por exemplo "Amostragem50", comando que altera o intervalo em que o protótipo envia informações de velocidade para o dispositivo para 50ms. A porta utilizada para comunicação foi a 8080.

Uma lista dos comando programados pode ser encontrada no Apêndice A.

As bibliotecas utilizadas para estabelecer a conexão Wi-Fi e a comunicação foram as bibliotecas "WiFi.h" ([SWEETYMHAISKE, 2020](#)) e sua integrante "WiFiUdp.h". A biblioteca "Wifi.h" foi modificada para limitar a quantidade de conexões e manter os valores de IP fixos através das alterações [3.1](#) e [3.2](#) no arquivo "WiFiGeneric.cpp" da biblioteca.

Código 3.1 – Alteração do incremento de IP

```
155 dhcp_ipaddr=dhcp_ipaddr==0 ? ap_ipaddr+3 : dhcp_ipaddr;
```

Código 3.2 – Alteração da quantidade máxima de conexões

```
159 lease.end_ip.addr=lease.start_ip.addr+0;
```

O controlador envia a velocidade e ângulo da roda virtual do modelo do robô em um período que pode ser alterado por comando, e a quantidade de pontos e as medições de distância do lidar a cada volta, tomado a frente do carro como ponto inicial e girando no sentido horário. As informações são separadas por "\_", tendo a *string* de velocidade e ângulo dois itens e as de distância 17, um para a quantidade de pontos e os seguintes os dados de distância com limite de até 16 pontos. Os pontos não utilizados aparecem como 0 (Figura 31).

Figura 31 – Exemplo de String de leitura do LiDAR

<b>String:</b>
<b>16_147_120_110_180_179_147_643_676_713_8190_371_311_576_8190_395_433</b>

Fonte: o autor, 2023

### 3.5.2 Direção

O sistema de direção é controlado pelo posicionamento dos servos ligados a cada roda dianteira. O controle dos ângulos se baseia no comando de posicionamento da roda virtual do modelo de acordo com a Figura 4 levando em conta a redução estabelecida pelas engrenagens.

Os ângulos se basearam nos parâmetros construtivos, onde  $W$  é a distância entre os eixos de rotação vertical das rodas dianteiras e  $L$  a distância entre os eixos dianteiro e traseiro. As rodas virtuais são posicionadas entre as rodas dianteiras e traseiras no centro do carro, assim primeiramente foi calculada a distância do ICR para o eixo traseiro da roda virtual,  $R$ , utilizando (3.2) e com isso, os ângulos de rotação de cada roda a partir de (3.3) e (3.4) sendo  $rel$  a relação de engrenagens do sistema de direção (53:33 dentes),  $\gamma$  o ângulo de rotação da roda virtual,  $\gamma_{int}$  o ângulo da roda interna e  $\gamma_{ext}$  o ângulo da roda externa à rotação.

$$R = \frac{L}{\tan \gamma}. \quad (3.2)$$

$$\gamma_{int} = (\text{atan}\left(\frac{L}{R - \frac{W}{2}}\right)) \cdot \text{rel.} \quad (3.3)$$

$$\gamma_{ext} = (\text{atan}\left(\frac{L}{R + \frac{W}{2}}\right)) \cdot \text{rel.} \quad (3.4)$$

### 3.5.3 Leituras de velocidade

A leitura de velocidade de rotação tanto do motor de tração quanto do motor do LiDAR foi feita a partir dos seus encoders.

O encoder dos motores emite 11 pulsos por revolução do motor antes da redução, aliado à redução de 34,02:1 do motor para o eixo, a cada rotação do eixo o encoder emite 374,22 pulsos.

Para calcular a velocidade de deslocamento do carro, leva-se em consideração o perímetro da roda e o intervalo de tempo entre um pulso e outro do encoder. Resultando em (3.5), onde  $vel$  é a velocidade do carro em metros por segundo,  $d$  é o diâmetro do pneu,  $p$  a quantidade de pulsos do encoder,  $r$  é a relação da redução e  $t_1$  e  $t_2$  são os tempos dos últimos pulsos do encoder em segundos.

$$vel = \frac{\frac{\pi \cdot d}{p \cdot r}}{t_1 - t_2} \rightarrow \frac{\frac{\pi \cdot 0,063}{11 \cdot 34,02}}{t_1 - t_2}. \quad (3.5)$$

### 3.5.4 Controle dos motores

A atuação dos motores é realizado por um sinal *Pulse Width Modulation (PWM)* para controlar a potência fornecida ao motor, juntamente a dois sinais digitais que controlam a direção de rotação enviados para a ponte H, um para cada motor.

A velocidade de rotação dos motores **CC** é sensível à carga aplicada a eles, assim é necessário fazer o ajuste de potência aplicada para se manter uma velocidade desejada independente do torque exigido.

Para isso, foi levantada a curva característica do sistema de tração através da resposta ao degrau, em condições normais de funcionamento.

O sinal **PWM** é controlado através do valor de um número inteiro positivo de 8bits enviado no comando de controle do sinal das portas do controlador, resultando em um range de 0 a 255, sendo 0 igual a 0% de potência e 255, 100% de potência.

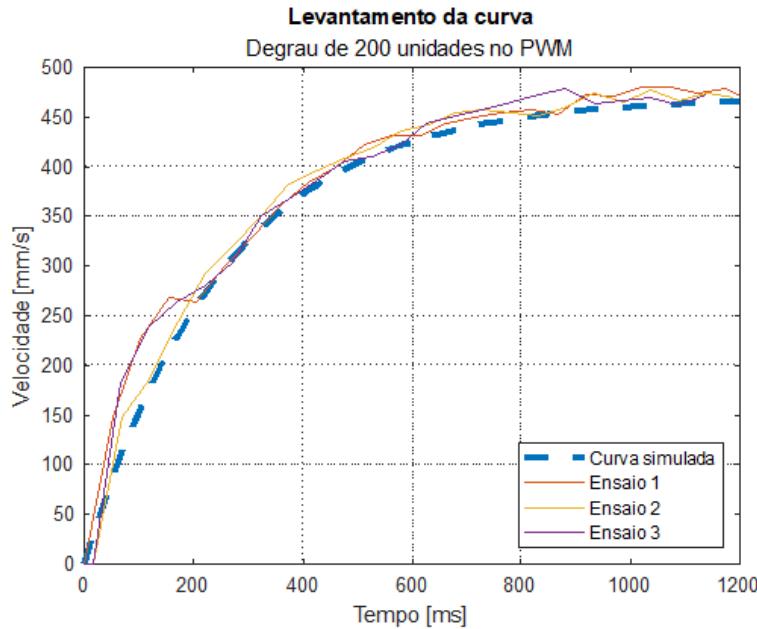
Durante os testes iniciais, foi identificado que o protótipo possui uma zona morta de 80 unidades para se manter em movimento, assim, para o levantamento da curva de funcionamento, foram feitos três ensaios em três configurações de potência diferentes, utilizando 150, 200 e 255 como valores de *duty cycle* do **PWM**.

Os ensaios foram realizados aplicando degraus no sinal PWM com o carro em uma linha reta sobre folhas de EVA coladas no chão. Os dados de leitura de velocidade e tempo a partir da execução do degrau foram enviados inicialmente com período de 50ms, armazenados e analisados. Como apresentaram resolução suficiente para o levantamento das características do sistema, a taxa de amostragem foi mantida em todos os testes de controle de velocidade.

Com os dados obtidos, os parâmetros  $A$ ,  $t_d$  e  $\tau$  de (2.9) foram definidos para as três curvas a partir da média dos parâmetros de cada amostra. A curva com degrau 200 foi selecionada como base por apresentar uma amplitude semelhante à definida no projeto. Obteve-se assim (3.6) obtendo a curva da Figura 32. Percebe-se que o sistema dinâmico de um motor DC! (DC!) pode ser corretamente representado por um sistema de primeiro grau.

$$\frac{A \cdot e^{-s \cdot t_d}}{\tau \cdot s + 1} \rightarrow \frac{2,345}{0,253 \cdot s + 1}. \quad (3.6)$$

Figura 32 – Levantamento da curva



Fonte: o autor, 2023

A equação foi discretizada pelo método de Tustin com tempo de amostragem igual a 50ms, período a ser utilizado na temporização do *loop* de controle de velocidade, resultando em (3.7)

$$\frac{0,2109 \cdot z + 0,2109}{z - 0,8201}. \quad (3.7)$$

De acordo com as premissas de projeto, definiu-se como objetivo do controlador eliminar o erro em regime permanente e manter o tempo de acomodação abaixo de 0,5s. Utilizando a (3.8) obteve-se o valor mínimo para  $\sigma$ .

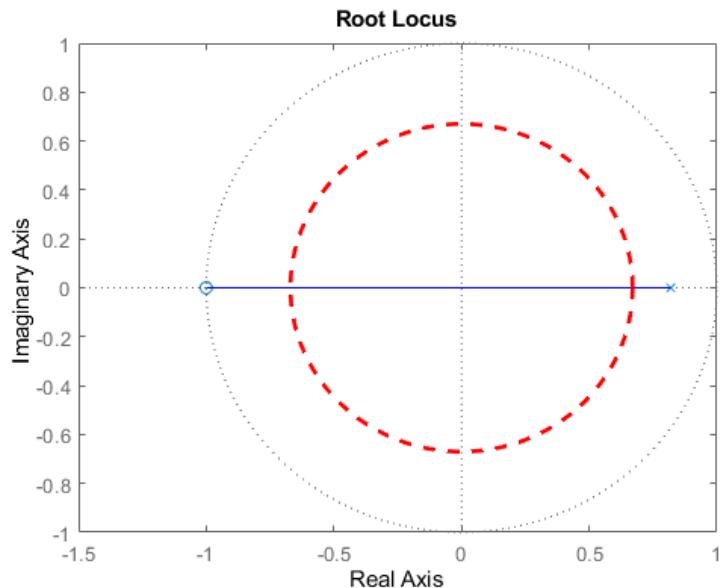
$$t_s(2\%) \leq \frac{4}{\sigma} \rightarrow \sigma \geq 8. \quad (3.8)$$

Passando para o domínio discreto a partir de (3.9), chegou-se a (3.10) onde  $r$  é o raio máximo e  $T$  o período de amostragem, formando o lugar das raízes da Figura 33, onde a região demarcada em vermelho representa a região de interesse.

$$z = e^{t \cdot s}. \quad (3.9)$$

$$r \leq e^{-\sigma \cdot T} \rightarrow r \leq 0,6703. \quad (3.10)$$

Figura 33 – Lugar das raízes do sistema discreto



Fonte: o autor, 2023

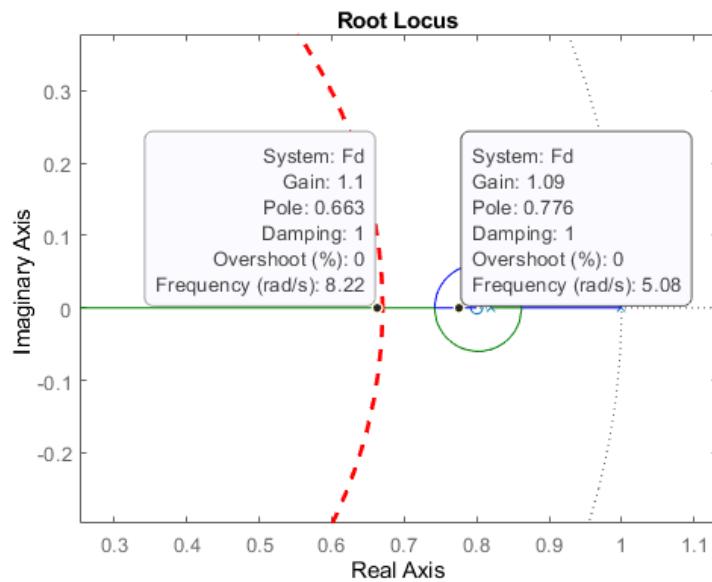
Com isso foi projetado um controlador PI com formato semelhante a (3.11) onde  $K$  é o ganho e  $\alpha$  é a posição do zero do controlador. A parte derivativa não foi necessária, já que o PI foi suficiente para trazer o sistema à região de interesse.

$$K \cdot \frac{z - \alpha}{z - 1}. \quad (3.11)$$

Respeitando os objetivos de controle, o zero do controlador foi posicionado próximo ao polo do sistema para trazer o sistema para a região de interesse sem que o zero possuísse muita influência, elevando o sobressinal do sistema, e sem acelerar muito o sistema, de forma a não alterar muito o comportamento do sistema devido à saturação do controle. Assim, o zero foi posicionado em 0,8 e a partir da simulação visualizada na Figura 34, foi selecionado o ganho 1,1 para o sistema entrar na região de interesse, resultando em (3.12).

$$1,1 \cdot \frac{z - 0,8}{z - 1}. \quad (3.12)$$

Figura 34 – Lugar das raízes do sistema com controlador



Fonte: o autor, 2023

Para a implementação do controlador, a equação foi levada ao formato (3.13), resultando em (3.14) que pode ser implementado no tempo na forma (3.15) de acordo com (ANGÉLICO, 2017)

$$K_P + \frac{K_I}{z - 1}. \quad (3.13)$$

$$1,1 + \frac{0,22}{z - 1}. \quad (3.14)$$

$$u[k] = K_P \cdot e[k] + K_I \cdot \sum_{v=0}^k e[v] \rightarrow u[k] = 1,1 \cdot e[k] + 0,22 \cdot \sum_{v=0}^k e[v]. \quad (3.15)$$

O mesmo controlador foi aplicado tanto ao motor de tração quanto ao motor de rotação do [LiDAR](#) com a adequação de escala de velocidade linear do carro para velocidade angular do [LiDAR](#). Apesar da diferença de esforço a qual os motores serão submetidos, os motores são iguais e o transitório da curva não é significativo para o [LiDAR](#), já que não sofrerá alteração constante de velocidade nem a incidência de perturbações, tendo o erro zero em regime permanente da velocidade garantido pelo controlador PI.

Foi adicionada também uma rotina anti wind-up para amenizar os efeitos de saturação nos transitórios.

Por fim, como a tração do carro é feita apenas em uma das rodas traseiras, foi feita a compensação da diferença de distância percorrida pela roda interna e externa nas curvas, mantendo a velocidade alvo referente à velocidade da roda virtual traseira.

O tempo necessário para percorrer o perímetro referente a um intervalo angular tem que ser o mesmo tanto para as rodas internas quanto para as externas, assim, tomando a roda virtual como referência e as equações de distância (3.16) e (3.17), fazendo  $\frac{v_{int,ext}}{v_v}$ , e isolando  $v_{int}$  ou  $v_{ext}$  chega-se a (3.18) e (3.19) onde  $v_v$  é a velocidade na roda virtual do modelo do robô,  $v_{int}$  é a velocidade na roda interna,  $v_{ext}$  é a velocidade na roda externa,  $R$  é o raio de curvatura e  $t$  é o tempo.

$$2 \cdot \pi \cdot R = v_v \cdot t \rightarrow v_v = \frac{2 \cdot \pi \cdot R}{t}. \quad (3.16)$$

$$2 \cdot \pi \cdot (R \pm W) = v_{int,ext} \cdot t \rightarrow v_{int,ext} = \frac{2 \cdot \pi \cdot (R \pm W)}{t}. \quad (3.17)$$

$$v_{int} = v_v \cdot \frac{R - W}{R}. \quad (3.18)$$

$$v_{ext} = v_v \cdot \frac{R + W}{R}. \quad (3.19)$$

### 3.5.5 Leitura de distância

A leitura de distância foi feita utilizando a biblioteca de *header* "VL53L0X.h" de ([POLOLU, 2022](#)) com adaptações para controlar o inicio das medições sem a interrupção da execução do código devido ao uso de *delays*.

A biblioteca apresenta dois métodos de aquisição dos dados lidos pelo sensor: o primeiro faz uma leitura única com o comando "*readRangeSingleMillimeters()*" onde a aquisição da distância é iniciada assim que o comando é executado, porém a execução é travada até que a leitura seja completada; o segundo método é a leitura contínua em intervalos fixos ou em leituras consecutivas com a obtenção dos dados através do comando

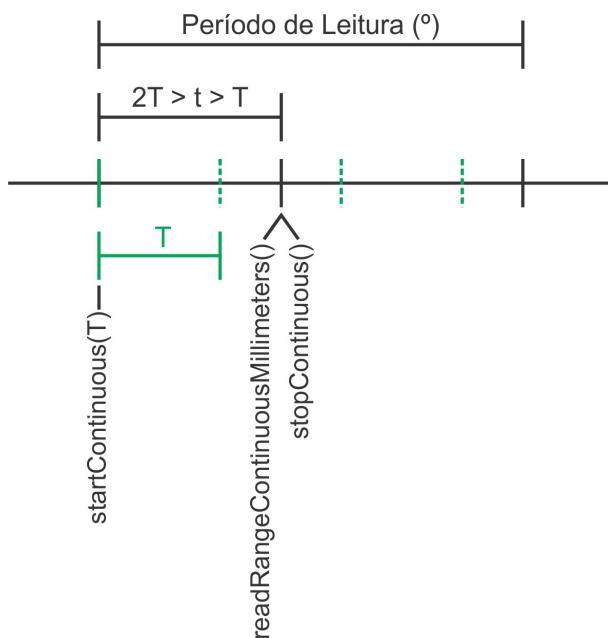
"readRangeContinuousMillimeters()". A configuração da leitura contínua é feita através dos comandos "startContinuous()", que inicia o ciclo de leituras e configura o período em seu argumento através de uma variável do tipo *int* indicando o período em milissegundos, e o comando "stopContinuous()" que interrompe o ciclo de leituras. O comando de aquisição da leitura é rápido mas não é simples sincronizar o período de leitura do sensor com o posicionamento do sensor.

Para fazer as leituras, o sensor foi configurado no modo rápido, com tempo máximo de leitura de 20ms, e foi feito um ciclo de inicio e parada de leituras em modo contínuo para se controlar o ponto onde a leitura era feita.

No modo contínuo, o comando de aquisição da leitura recebe imediatamente o valor da ultima leitura não entregue, caso a ultima leitura já tenha sido adquirida, a execução do programa espera que a leitura atual seja finalizada e então entregue para o controlador.

Para evitar paralisações no programa e ter controle do inicio da leitura na posição correta, foi feita uma rotina onde ao chegar na posição em que a leitura deveria ser tomada (Figura 35), o comando de inicio de aquisição contínua é dado, com um período inferior ao tempo de deslocamento entre um ponto de medida e outro, para que se inicie a primeira leitura do ciclo, segue-se a execução do programa e após o tempo da finalização da primeira leitura e antes da finalização da segunda, os comando de obtenção da leitura e o de parada de leitura são dados. Assim, nenhum comando aguarda a finalização de qualquer processo para a continuidade do programa e pode-se garantir a posição da tomada da leitura.

Figura 35 – Ciclo de leitura de distância



Fonte: o autor, 2023

# 4 Análise de resultados

## 4.1 Testes do controlador de velocidade

### 4.1.1 Degrau

Para a validação do controlador de velocidade foram realizados ensaios onde foram aplicados degraus com o mesmo valor final dos degraus aplicados para o levantamento da curva do sistema, com o objetivo de identificar se os objetivos de controle foram alcançados e compara o comportamento do sistema em malha aberta e malha fechada.

O *setpoints* estabelecidos foram 365mm/s referente ao ensaio com degrau de 150 unidade do [PWM](#), 469mm/s referente ao degrau de 200 unidades e 600mm/s referente ao degrau de 255 unidades. O ensaio a 600mm/s estaria fora das premissas de projeto, porém devido à compensação de velocidade nas curvas, é possível que o valor seja alcançado.

Para cada valor foram realizados três ensaios que apresentaram comportamento bem semelhante entre si como é possível visualizar nas Figuras [36](#), [37](#) e [38](#). Todos os testes atingiram o objetivo de tempo de estabilização abaixo de 500ms. Os testes a 365mm/s e 469mm/s apresentaram melhorias significativas com relação ao sistema em malha aberta, apresentando pequenas oscilações que podem ser causadas por ruídos ou algum pequeno atraso de propagação não identificado do sistema.

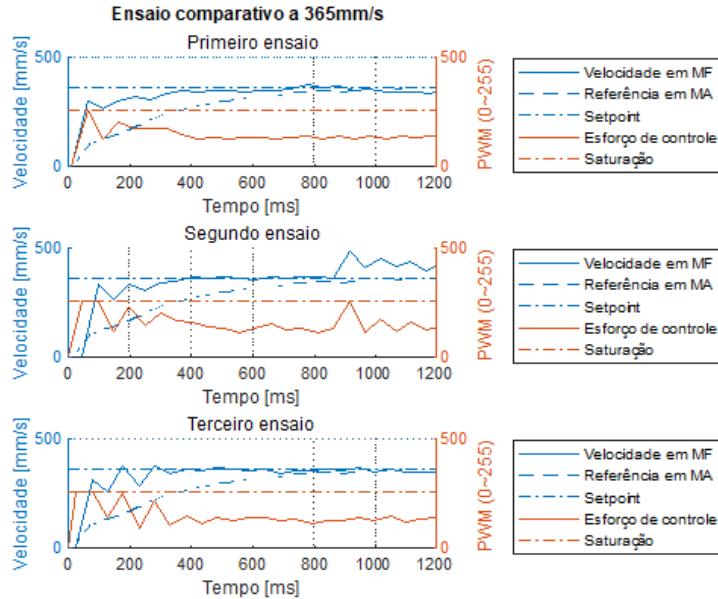
### 4.1.2 Testes de esforço

Foram realizados testes adicionando e retirando esforço do sistema para identificar o comportamento do controlador. Para a comparação da resposta a diferentes esforços foram realizados ensaios com o carro suspenso, no chão e com uma carga de 1kg (Figura [39](#)), todos com um degrau de 300mm/s. Os resultados podem ser visualizados na Figura [40](#).

Percebe-se a diferença no esforço do controlador em manter a velocidade alvo. No teste suspenso o sistema reage rápido demais ao esforço de controle, o que causa um *overshoot* considerável, mas que retorna rapidamente ao *setpoint* estabelecido. O teste no chão apresenta um resultado rápido com um pequeno *overshoot* recuperado em algum tempo. O teste com carga apresenta um período grande de saturação que causa um *overshoot* considerável por um tempo longo, mas controlado a partir de 1s.

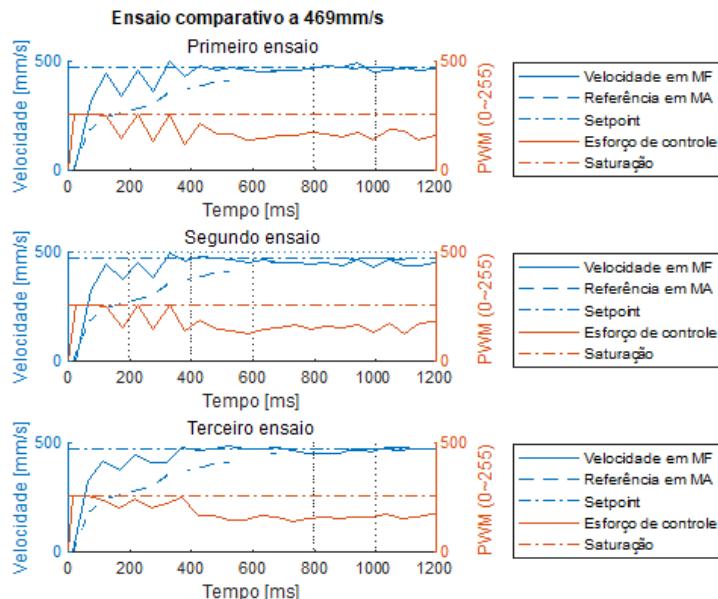
É perceptível a diferença no nível final do esforço de controle entre os ensaios, onde o esforço se estabiliza em valores menores no ensaio suspenso com relação ao ensaio com carga.

Figura 36 – Ensaio comparativo a 365mm/s



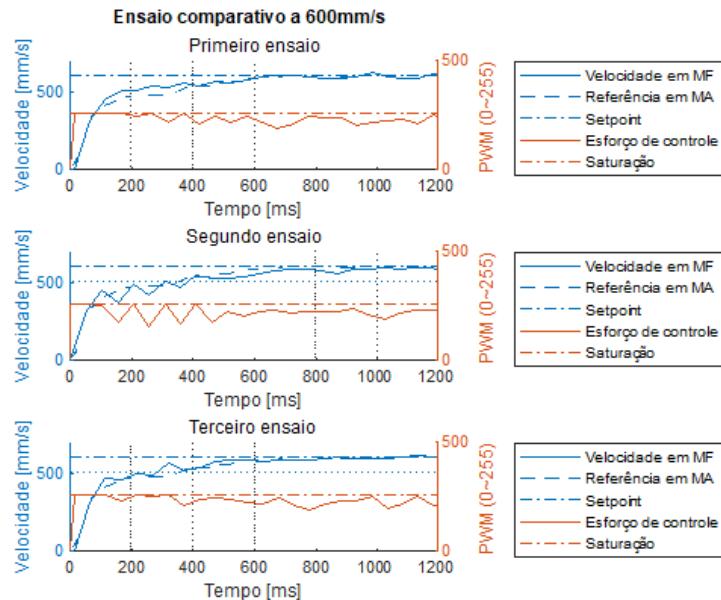
Fonte: o autor através do MatLab

Figura 37 – Ensaio comparativo a 469mm/s



Fonte: o autor, 2023

Figura 38 – Ensaio comparativo a 600mm/s



Fonte: o autor, 2023

Figura 39 – Teste com carga de 1kg



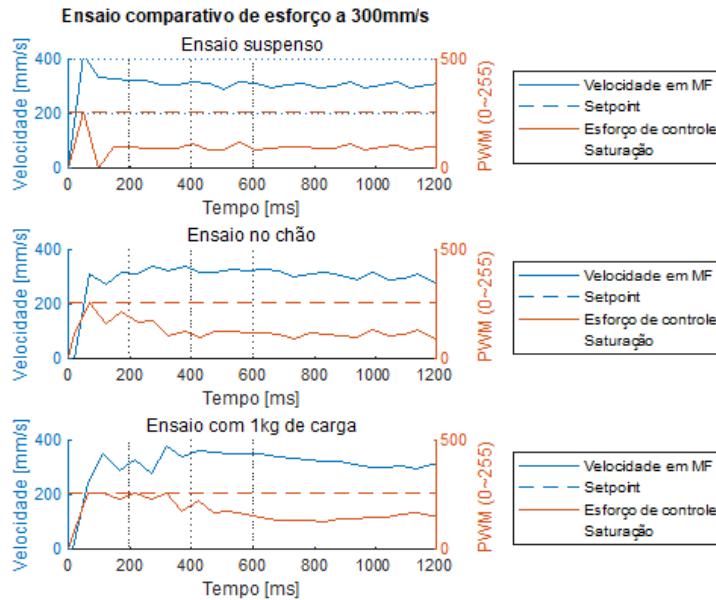
Fonte: o autor, 2023

### 4.1.3 Compensação de curva

Para validar a compensação da velocidade nas curvas o carro foi posto a girar em círculos com esterçamento máximo em velocidades lentas e altas sobre placas de EVA e foi medido o tempo que levava a dar uma volta completa tanto no sentido horário quanto no anti-horário.

Analizando os dados das tabelas 4 e 5, é possível perceber uma diferença de 16,73%

Figura 40 – Ensaio comparativo de esforço



Fonte: o autor, 2023

em velocidades lentas e 15,66% em velocidades altas em relação ao tempo de rotação no sentido horário para o tempo no sentido anti-horário. Essa diferença provavelmente é causada pelo aumento de esforço entre movimentar o carro pela parte externa do círculo (sentido anti-horário) ou pela parte interna (sentido horário) o que causa mais derrapagem, reduzindo a velocidade geral do carro.

Tabela 4 – Teste de tempo de volta - 150mm/s

Volta	Tempo - Anti-Horário [s]	Tempo - Horário [s]
1	11,38	13,50
2	11,69	13,70
3	11,89	13,60
Média	11,65	13,6

Fonte: o autor, 2023

Tabela 5 – Teste de tempo de volta - 500mm/s

Volta	Tempo - Anti-Horário [s]	Tempo - Horário [s]
1	3,41	4,02
2	3,56	4,11
3	3,56	4,05
4	3,50	4,06
Média	3,51	4,06

Fonte: o autor, 2023

## 4.2 Validação do LiDAR

O [LiDAR](#) apresentou medições estáveis principalmente em distâncias pequenas mas apresentou instabilidade quanto à frequência de medições. Fazendo até 10 coletas por revolução o sistema apresentou comportamento normal, mas a partir de 12 medições a velocidade de rotação teve que ser limitada para que se obtivessem dados estáveis. As limitações podem ser visualizadas na Tabela 6.

Tabela 6 – Limitação de velocidade pela quantidade de medidas

Quantidade de medidas por revolução	12	14	16
Período mínimo de revolução [ms]	516	614	700

Fonte: o autor, 2023

### 4.2.1 Testes de leitura

Foram realizados três testes de leitura do [LiDAR](#), no primeiro teste foi avaliada a precisão de leitura e nos outros dois a capacidade de detectar obstáculos e traços do ambiente

Para o teste de precisão do [LiDAR](#) foi colocado um obstáculo fixo na frente do protótipo e foram realizados ensaios com o obstáculo em diferentes distâncias, fazendo a medição da distância real através de uma trena e obtendo dados de dez medidas em ângulo 0º do sensor girando em velocidade máxima.

Os dados obtidos podem ser vistos na Tabela 7. Percebe-se que o erro cresce conforme as distâncias aumentam, apesar da precisão não diminuir muito. O problema pode ser causado pela largura do cone de visão do sensor ou pela rotação do sensor enquanto faz as medições. A discrepância pode ser um problema caso o protótipo esteja sendo utilizado para mapeamento, porém podem ser feitos mais testes e implementada alguma correção de escala nos valores caso o comportamento seja estável em diferentes situações. Para a utilização apenas para orientação, o erro em baixas distâncias é pequeno o suficiente para não comprometer a atividade e tomadas de decisões.

Nos outros dois testes, foram dispostos obstáculos ao redor do carro e avaliadas as leituras de três voltas do [LiDAR](#), fazendo 16 leituras por volta, com relação à localização dos obstáculos. Os resultados podem ser visualizados nas Figuras 41 e 42 para os testes de detecção de obstáculos e nas Figuras 43 e 44 para os testes de detecção de vagas. As leituras são indicadas pelo círculos laranjas, pontos distantes, fora da região do teste, não foram representados. Foram representados também o cone teórico de visão do sensor (25º) e dois círculos indicando as distâncias 250mm e 500mm do sensor para referência.

As discrepâncias das leituras também podem ser justificadas pelo tamanho do cone de visão teórico do sensor (25º) e pela rotação do sensor enquanto este faz a leitura de

Tabela 7 – Teste de precisão de medição

Valor real (trena) [mm]	254.0	452.0	687.0	998.0
<b>Medida 1</b>	263	465	703	1130
<b>Medida 2</b>	263	474	737	1078
<b>Medida 3</b>	256	458	745	1071
<b>Medida 4</b>	254	469	709	1132
<b>Medida 5</b>	264	474	721	1073
<b>Medida 6</b>	255	470	711	1121
<b>Medida 7</b>	255	470	705	1063
<b>Medida 8</b>	253	467	707	1135
<b>Medida 9</b>	261	469	731	1131
<b>Medida 10</b>	266	467	723	1082
<b>Média [mm]</b>	259	469	716	1102
<b>Erro [mm]</b>	5	17	29	104
<b>Erro [%]</b>	1,97	3,76	4,22	10,42
<b>Desvio padrão [mm]</b>	5	5	15	30
<b>Desvio padrão [%]</b>	1,97	1,11	2,18	3,01

Fonte: o autor, 2023

Figura 41 – Teste de detecção de obstáculos



Fonte: o autor, 2023

distância.

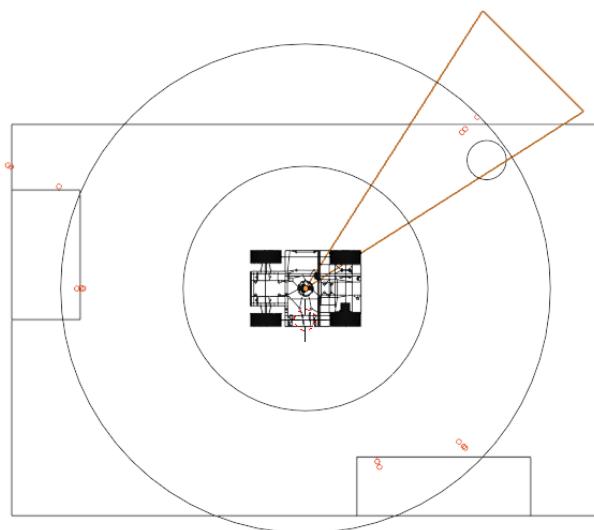
Os obstáculos foram identificados e as discrepâncias não comprometeriam a realização de atividades de localização.

#### 4.2.2 Ângulo de visão

Foi realizado um teste de validação do ângulo do cone de visão, que de acordo com ([STMICROELECTRONICS, 2022](#)) é de 25°, e a consistência do ângulo em que a próxima medida deve ser tomada. O teste é importante pois uma região muito grande compromete a precisão do sensor.

Para validar o ângulo do cone de visão do sensor de distância, foram feitas marcações no chão, formando um círculo com 500mm de raio, o protótipo foi colocado no centro

Figura 42 – Resultado do teste de detecção de obstáculos, ponto fora da região do teste não representados



Fonte: o autor, 2023

Figura 43 – Teste de detecção de vaga

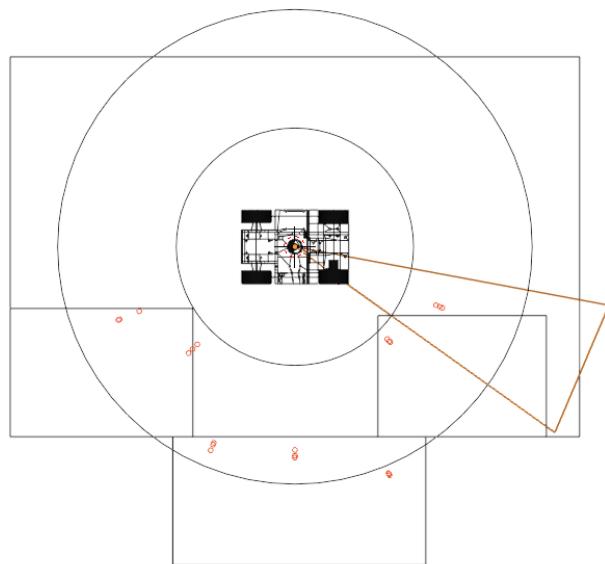


Fonte: o autor, 2023

fazendo medições com 4 pontos por revolução à velocidade máxima e foram colocados obstáculos nas bordas externas do campo de visão em cima do círculo formado. Os cantos dos obstáculos foram marcados quando estivessem no limiar de ser identificados pelo sensor (Figura 45).

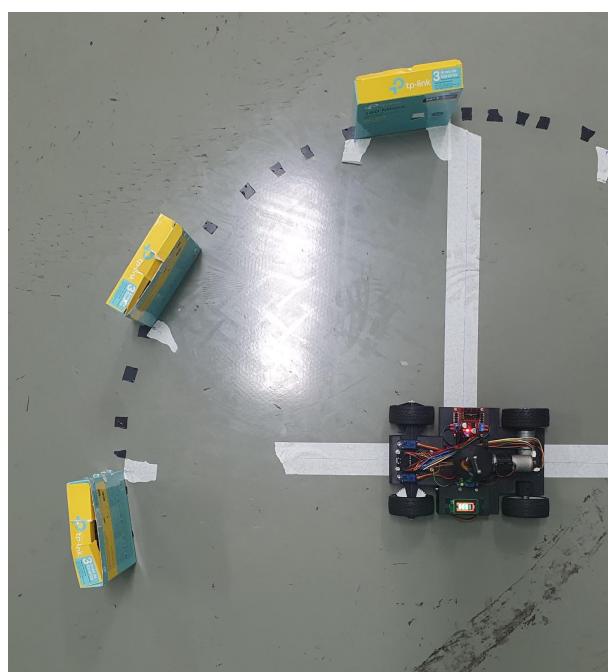
Após a marcação de um intervalo de detecção e outro de não-detecção, as distâncias entre os pontos foram medidas com trena e usando as dimensões dos triângulos formados pelas distâncias entre os pontos e os raios do círculo, foi possível calcular os ângulos do

Figura 44 – Resultado do teste de detecção de vaga, pontos fora da região do teste não representados



Fonte: o autor, 2023

Figura 45 – Teste de ângulo de visão



Fonte: o autor, 2023

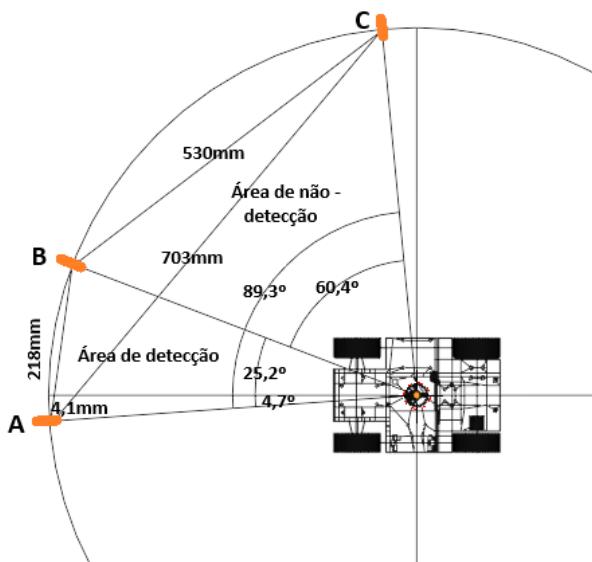
cone de visão e o intervalo entre uma medição e o início da próxima. O resultado pode ser visto na Tabela 8 e na Figura 46.

Tabela 8 – Resultado do teste de ângulo de visão

Retas Distância [mm]	A-B	B-C	C-D
Ângulo do triângulo formado [º]	218	530	703
	25,2	60,4	89,3

Fonte: o autor, 2023

Figura 46 – Resultado do teste de ângulo de visão, pontos adquiridos destacados em laranja



Fonte: o autor, 2023

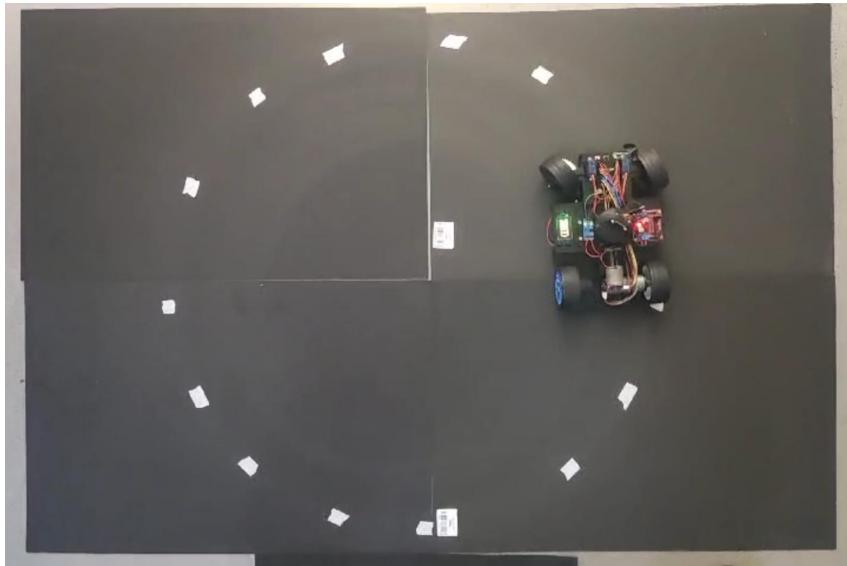
É possível identificar que o raio de visão calculado ( $25,2^\circ$ ) é bem parecido com o raio teórico encontrado em (STMICROELECTRONICS, 2022) ( $25^\circ$ ). O ângulo entre o início de uma medição e a próxima apresentou uma discrepância quando calculado pela soma dos ângulos formados pelos triângulos adjacentes ( $85,6^\circ$ ) quando comparado com o ângulo obtido pela distância entre os pontos ( $89,3^\circ$ ), essa diferença pode ter sido causada por erros durante a medição, ou pelo mal posicionamento de algum dos obstáculos.

Percebe-se também um erro de zero no início das medições causado devido à identificação da passagem da lingueta ser feita na borda se subida do sinal, podendo ser corrigido facilmente por código.

### 4.3 Teste de raio de curva

Foram realizados testes de raio de curva onde o carro foi disposto a andar em círculos com esterçamento máximo sobre placas de EVA onde foram colocadas marcações sobre o caminho percorrido 47.

Figura 47 – Teste de raio de curva



Fonte: o autor

O carro não apresentou variações no raio de curva tanto com relação ao sentido de giro, quanto à velocidade de deslocamento.

A roda traseira externa foi usada como referência e o raio de curva teórico pode ser calculado a partir da equação (4.1). Sendo  $R$  o raio de curva teórico,  $L$  a distância entre eixos (163,8mm),  $\gamma$  o ângulo de esterçamento ( $30^\circ$ ) e  $W$  a distância entre os eixos de rotação vertical das rodas dianteiras (128,9mm).

$$R = \frac{L}{\tan \gamma} + \frac{W}{2} = 348,16. \quad (4.1)$$

O raio do círculo formado pelas marcações foi medido a partir da metade da média das distâncias entre as marcações opostas e confirmado a partir da intersecção dos arcos formados por um cordão com o tamanho calculado a partir de diversas marcações do círculo. O tamanho encontrado foi de 361,0mm medido com uma trena.

O resultado apresenta uma diferença de 12,8mm o que representa 3,69% do valor teórico. A diferença bastante pequena e pode ser causada pela derrapagem dos pneus.

# 5 Conclusão

Este trabalho detalhou todo o processo de projeto de um robô móvel *car-like* com um sistema de [LiDAR](#) 2D capaz de desenvolver tarefas relacionadas à robótica e Computação.

Foram desenvolvidas todas as etapas do projeto desde a fundamentação teórica, passando pelas premissas e escolhas de componentes, desenvolvimento dos sistemas do protótipo, programação do sistema embarcado, controle e sensoriamento e os testes e avaliação de resultados.

O produto final do projeto foi capaz de atender aos objetivos especificados respeitando todas as premissas básicas estabelecidas, apresentando limitações apenas no sistema [LiDAR](#) quanto à velocidade de aquisição quando utilizando mais pontos por revolução.

O protótipo foi capaz de responder aos comandos transmitidos tanto de computadores, quanto celulares, se movimentando satisfatoriamente. Todas as comunicações entre o protótipo e o usuário foram feitas através do sinal Wi-Fi usando o protocolo [UDP](#).

O protótipo pode ser utilizado para o desenvolvimento de algoritmos de Inteligência Artificial ([IA](#)), validação de controladores, novas metodologias para sistemas embarcados, veículos autônomos.

Os dados obtidos foram suficientes para a identificação de obstáculos. Apesar de apresentar desvios consideráveis em distâncias maiores, isso não é problema para ambientes menores e para o desenvolvimento de atividades de docagem, estacionamento e mapeamento de uma zona próxima.

Todos os aspectos construtivos foram documentados no Apêndice B e o projeto foi ajustado para manter a simplicidade e previsibilidade de funcionamento, tendo êxito em fazer o controle do tempo de acomodação e eliminação do erro em regime permanente da velocidade e rotação do [LiDAR](#).

## 5.1 Sugestão de trabalhos futuros

Para trabalhos futuros sugere-se o teste do sistema em aplicações mais complexas e que apresentem funcionalidades específicas para o meio acadêmico e o cotidiano.

Um exemplo seria a utilização dos dados do [LiDAR](#) para a alimentação de uma rede neural capaz de desviar de obstáculos ou estacionar o protótipo autonomamente.

Como aperfeiçoamento do projeto sugere-se o desenvolvimento do sistema [LiDAR](#) com a implementação de lentes para diminuir o cone de visão aumentando sua exatidão

ou desenvolvendo uma biblioteca que permita um melhor controle do comportamento do sensor, de forma que não interfira na execução dos outros sistemas e apresente maior velocidade do ciclo de medidas.

## Referências

- AHMED, A. S.; MARZOG, H. A.; ABDUL-RAHAIM, L. A. Design and implement of robotic arm and control of moving via iot with arduino esp32. *International Journal of Electrical & Computer Engineering* (2088-8708), v. 11, n. 5, 2021. Citado na página 46.
- ALBUQUERQUE, P. U. B. d.; THOMAZINI, D. Sensores industriais: fundamentos e aplicações. São Paulo: Érica, 2005. Citado na página 35.
- ALEXANDRINO, H. H. F. *Iguatemi instala vigilância eletrônica via drone*. 2022. Acesso em: 23/11/2023. Disponível em: <<https://portalterradaluz.com.br/curtas/seguranca-iguatemi-instala-vigilancia-eletronica-via-drone/>>. Citado na página 26.
- ANDERSEN, H.-E.; REUTEBUCH, S. E.; MCGAUGHEY, R. J. A rigorous assessment of tree height measurements obtained using airborne lidar and conventional field methods. *Canadian Journal of Remote Sensing*, Taylor & Francis, v. 32, n. 5, p. 355–366, 2006. Citado na página 39.
- ANGÉLICO, B. *LAB5 – Projeto de Controlador PI Digital*. 2017. Acesso em: 23/11/2023. Disponível em: <[https://edisciplinas.usp.br/pluginfile.php/3441791/mod\\_resource/content/4/LAB5\\_BAA\\_2017.pdf](https://edisciplinas.usp.br/pluginfile.php/3441791/mod_resource/content/4/LAB5_BAA_2017.pdf)>. Citado na página 72.
- ARAÚJO, F. M. U. de. Sistemas de controle. 2007. Citado na página 42.
- ARDUINO, B. *Arduino Uno*. [S.l.]: Obtenido de <http://www.arduino.cc/en/Main/ArduinoBoardUno>, 2015. Citado na página 46.
- BALL, S. *Embedded microprocessor systems: real world design*. [S.l.]: Elsevier, 2002. Citado na página 43.
- BITTENCOURT, M. C. Identificação de sistemas dinâmicos lineares - métodos paramétricos e não paramétricos. 2007. Citado 2 vezes nas páginas 39 e 42.
- BLENDER. *The softwares*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://www.blender.org/about/>>. Citado na página 59.
- CASTELLI, I. *Como funcionam os robôs antibombas?* 2013. Acesso em: 23/11/2023. Disponível em: <<https://www.tecmundo.com.br/robotica/38103-como-funcionam-os-robos-antibombas-.htm>>. Citado na página 25.
- CENTENO, J. A. S.; MITISHITA, E. A. Laser scanner aerotransportado no estudo de áreas urbanas: A experiência da ufpr. *XIII Simpósio Brasileiro de Sensoriamento Remoto*, v. 4, p. 3645–3652, 2007. Citado na página 39.
- CHASE, O.; ALMEIDA, F. Sistemas embarcados. *Mídia Eletrônica. Página na internet:<www.sabajovem.org/chase>*, capturado em, v. 10, n. 11, p. 13, 2007. Citado 2 vezes nas páginas 43 e 44.
- CIRCUITO, C. *Sensor de Distância Laser - VL53L0X*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://curtocircuito.com.br/sensor-de-distancia-vl53l0x.html>>. Citado na página 54.

CIRCUITO, C. *Sensor de Obstáculo Infravermelho IR*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://curtocircuito.com.br/sensor-de-obstaculo-infravermelho.html>>. Citado na página 54.

COELHO, A. H.; VARGAS, R. M. A. Geração de modelos digitais de terreno a partir de dados de laser scanner aerotransportado em área de floresta usando o software livre grass. In: *SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO*, v. 13, p. 3653–3660, 2007. Citado na página 39.

CORE, R. *Sensor de Obstáculo*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://www.robocore.net/sensor-robo/sensor-de-obstaculo>>. Citado na página 55.

CUNHA, A. F. O que são sistemas embarcados. *Saber Eletrônica*, v. 43, n. 414, p. 1–6, 2007. Citado na página 43.

DFROBOT. *Metal DC Geared Motor w/Encoder - 6V 210RPM 10Kg.cm*. 2018. Acesso em: 23/11/2023. Disponível em: <<https://www.dfrobot.com/product-1617.html>>. Citado 2 vezes nas páginas 50 e 51.

ELEKTRA. *Você sabe o que é e como funciona o sensor ultrassônico?* 2022. Acesso em: 21/11/2023. Disponível em: <<https://tipotemporario.com.br/elektra/blog/voce-sabe-o-que-e-e-como-funciona-o-sensor-ultrassonico/>>. Citado na página 35.

ESPRESSIF. *ESP32 Series Datasheet*. [S.l.], 2023. Acesso em: 23/11/2023. Disponível em: <[https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)>. Citado na página 52.

FORNTEIRAS, G. sem. *Drone Pulverizador: Revolucionando a agricultura com precisão e eficiência*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://geosemfronteiras.org/blog/drone-pulverizador-na-agricultura/>>. Citado na página 26.

FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. *Feedback control of dynamic systems*. [S.l.]: Pearson London, 2015. v. 33. Citado na página 41.

FUCHS, S. et al. Cooperative bin-picking with time-of-flight camera and impedance controlled dlr lightweight robot iii. In: IEEE. *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.], 2010. p. 4862–4867. Citado na página 37.

GILLESPIE, T. *Fundamentals of vehicle dynamics*. [S.l.]: SAE international, 2021. Citado na página 50.

GIRARDI, A. R. et al. Controle por realimentação de estados aplicado a conversores boost. Blumenau, SC., 2023. Citado 2 vezes nas páginas 34 e 41.

HENDRICKS, E.; JANNERUP, O.; SØRENSEN, P. H. *Linear systems control: deterministic and stochastic methods*. [S.l.]: Springer, 2008. Citado na página 34.

INDUSTRIAL, M. *O que é um sensor ultrassônico*. 2018. Acesso em: 21/11/2023. Disponível em: <<http://fatecjahu.edu.br/ferramentas/ojs/index.php/revista/article/view/110/39>>. Citado na página 35.

JIMB15122. *Bevel Gear Matched Pair Generator*. 2019. Acesso em: 23/11/2023. Disponível em: <<https://www.thingiverse.com/thing:3682990>>. Citado na página 60.

- KELLER, M.; KOLB, A. Real-time simulation of time-of-flight sensors. *Simulation Modelling Practice and Theory*, Elsevier, v. 17, n. 5, p. 967–978, 2009. Citado na página 36.
- KURNIAWAN, A. *Internet of Things Projects with ESP32: Build exciting and powerful IoT projects using the all-new Espressif ESP32*. [S.l.]: Packt Publishing Ltd, 2019. Citado na página 45.
- MARCUS, J. *Como drones armados estão criando 'nova era da guerra'*. 2022. Acesso em: 23/11/2023. Disponível em: <<https://www.bbc.com/portuguese/geral-60272353>>. Citado na página 26.
- MATOS, N. M. R. d. et al. Análise do funcionamento de um servomotor de corrente alternada com ímãs permanentes. 2012. Citado na página 38.
- MECHATRONICS, N. *TUTORIAL SENSOR DE DISTANCIA SHARP*. 2016. Acesso em: 21/11/2023. Disponível em: <[https://naylampmechatronics.com/blog/55\\_tutorial-sensor-de-distancia-sharp.html](https://naylampmechatronics.com/blog/55_tutorial-sensor-de-distancia-sharp.html)>. Citado 2 vezes nas páginas 35 e 36.
- MENDES, F. *Como funciona o Proteus, 1º robô autônomo a trabalhar na Amazon*. 2022. Acesso em: 23/11/2023. Disponível em: <<https://www.uol.com.br/tilt/noticias/redacao/2022/06/22/proteus-robo-amazon.htm>>. Citado na página 26.
- MPJA. *SG90 MINI SERVO MOTOR*. [S.l.], 2021. Acesso em: 23/11/2023. Disponível em: <<https://www.mpja.com/download/31002md%20data.pdf>>. Citado na página 49.
- NAIR, R. et al. A survey on time-of-flight stereo fusion. In: SPRINGER. *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications: Dagstuhl 2012 Seminar on Time-of-Flight Imaging and GCPR 2013 Workshop on Imaging New Modalities*. [S.l.], 2013. p. 105–127. Citado na página 37.
- NOGUEIRA, J. G. *Conheça os incríveis shows de luzes feitos por drones que acontecem na China*. 2019. Acesso em: 23/11/2023. Disponível em: <<https://www.mundoconectado.com.br/drones/conheca-os-incriveis-shows-de-luzes-feitos-por-drones-que-acontecem-na-china/>>. Citado na página 26.
- ÖZDEMİR, A.; ERDEM, Z. Double-loop pi controller design of the dc-dc boost converter with a proposed approach for calculation of the controller parameters. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, SAGE Publications Sage UK: London, England, v. 232, n. 2, p. 137–148, 2018. Citado na página 42.
- PETER, C. *Robotics, vision and control: fundamental algorithms in MATLAB*. [S.l.]: Springer, 2011. Citado 4 vezes nas páginas 25, 29, 31 e 32.
- POLOLU. *VL53L0X library for Arduino*. 2022. Acesso em: 23/11/2023. Disponível em: <<https://github.com/pololu/vl53l0x-arduino>>. Citado na página 73.
- REPINALDO, J. P. *Controle de velocidade de um servomotor utilizando software Labview Real-Time*. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2015. Citado na página 38.

- RISCADO, D. *Encoders em Quadratura e Micro-Controlador com QEI para Controlo dos Motores*. 2013. Acesso em: 22/11/2023. Disponível em: <<https://stlittle.wordpress.com/2013/02/11/encoders-em-quadratura-e-micro-controlador-com-qei-para-controlo-dos-motores/>>. Citado na página 38.
- SAMSUNG, E. B. D. *SPECIFICATION OF PRODUCT for Lithium-ion Rechargeable Cell*. [S.l.], 2009. Acesso em: 23/11/2023. Disponível em: <<https://www.batteryspace.com/prod-specs/ICR18650-26F.pdf>>. Citado na página 51.
- SENRING. *M220 Series Medium Capsule Slip Ring*. 2023. Acesso em: 23/11/2023. Disponível em: <<https://www.senring.com/capsule-slip-ring/m220.html>>. Citado 2 vezes nas páginas 54 e 55.
- SICK. *O que é LiDAR?* 2013. Acesso em: 22/11/2023. Disponível em: <<https://www.sick.com/br/pt/produtos-e-solucoes/produtos/sensores-radar-e-lidar/sensores-lidar/c/g575802>>. Citado 2 vezes nas páginas 39 e 40.
- SIEGWART, R.; NOURBAKHSH, I. R.; SCARAMUZZA, D. *Introduction to autonomous mobile robots*. [S.l.]: MIT press, 2011. Citado 5 vezes nas páginas 29, 30, 31, 33 e 48.
- STMICROELECTRONICS. *L298 Datasheet*. [S.l.], 2000. Acesso em: 23/11/2023. Disponível em: <<https://www.makerhero.com/img/files/download/L298-Datasheet.pdf>>. Citado na página 56.
- STMICROELECTRONICS. *ESP32 Series Datasheet*. [S.l.], 2022. Acesso em: 23/11/2023. Disponível em: <<https://www.st.com/resource/en/datasheet/vl53l0x.pdf>>. Citado 3 vezes nas páginas 53, 80 e 83.
- SÜZEN, A. A.; DUMAN, B.; ŞEN, B. Benchmark analysis of jetson tx2, jetson nano and raspberry pi using deep-cnn. In: IEEE. *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. [S.l.], 2020. p. 1–5. Citado na página 46.
- SWEETYMHAISKE. *WiFi.h*. 2020. Acesso em: 23/11/2023. Disponível em: <<https://github.com/espressif/arduino-esp32/blob/master/libraries/WiFi/src/WiFi.h>>. Citado na página 67.
- TRONNIC. *Conheça o Samsung Ballie*. 2020. Acesso em: 21/11/2023. Disponível em: <<https://r1.community.samsung.com/t5/lifestyle/conhe%C3%A7a-o-samsung-ballie/td-p/7287609>>. Citado 2 vezes nas páginas 26 e 27.
- TUTORIALS, R. N. *ESP32 Pinout Reference: Which GPIO pins should you use?* 2018. Acesso em: 23/11/2023. Disponível em: <<https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>>. Citado 2 vezes nas páginas 53 e 58.
- TUTORIALS, R. N. *ESP32 with DC Motor and L298N Motor Driver – Control Speed and Direction*. 2018. Acesso em: 23/11/2023. Disponível em: <<https://randomnerdtutorials.com/esp32-dc-motor-l298n-motor-driver-control-speed-direction/>>. Citado na página 57.
- VIEIRA, L. *Drones entregam sangue, vacinas e medicamentos em Gana, na África*. 2019. Acesso em: 23/11/2023. Disponível em: <<https://noticias.r7.com/tecnologia-e-ciencia/fotos/drones-entregam-sangue-vacinas-e-medicamentos-em-gana-na-africa-17052019>>. Citado na página 26.

- VéDERA. *China desenvolve colheitadeira autônoma*. 2021. Acesso em: 23/11/2023. Disponível em: <<https://vederabrasil.com/china-desenvolve-colheitadeira-autonomia/>>. Citado na página 26.
- WAGNER, W. et al. *From single-pulse to full-waveform airborne laser scanners: Potential and practical challenges*. [S.l.]: na, 2004. v. 35. Citado na página 39.
- WORKSHOP, D. *Servo Motors with the ESP32*. 2020. Acesso em: 23/11/2023. Disponível em: <<https://dronebotworkshop.com/esp32-servo/>>. Citado 2 vezes nas páginas 50 e 57.
- ZHENG, L. et al. Lane-level road network generation techniques for lane-level maps of autonomous vehicles: A survey. *Sustainability*, MDPI, v. 11, n. 16, p. 4511, 2019. Citado na página 37.



# APÊNDICE A – Tabela de comandos

Todos os comandos são compostos por um comando e um valor auxiliar de acordo com a Tabela 9.

Tabela 9 – Lista de comandos

Comando	Valor auxiliar	Descrição
go	0 255	Comando de controle do PWM de motor do LiDAR
Parar	X	Comando para parar a movimentação do protótipo
Points	4 16	Comando para configurar a quantidade de pontos por revolução do LiDAR
Amostragem	50 5000	Comando para configurar o tempo de amostragem da velocidade e ângulo das rodas
Deg	0 255	Comando de controle do PWM de motor de tração
Vel	0 500	Comando de controle de velocidade do protótipo em mm/s
Acelerar	X	Comando para acelerar o protótipo
Re	X	Comando para desacelerar o protótipo
Dir	-30 30	Comando para configurar o ângulo de esterçamento da roda virtual do modelo
Direita	X	Comando para incrementar a direção para a direita
Esquerda	X	Comando para incrementar a direção para a esquerda
off	X	Comando para desativar o LiDAR

Fonte: o autor, 2023



## APÊNDICE B – Ficha técnica

- Comprimento total = 225,89mm
- Distância entre eixos = 163,80mm
- Largura do chassi = 155,40mm
- Distância entre eixos de rotação das rodas dianteiras = 128,90mm
- Esterçamento máximo =  $\pm 30^\circ$



# APÊNDICE C – Lista de Componentes

Tabela 10 – Lista de componentes

Quantidade	Componente
500g	Filamento PLA
1	Módulo ESP32
1	Sensor VL53L0X
2	Motor DC 210RPM
1	Anel Rotativo
1	Módulo ponte H
1	Sensor de Obstáculos
4	Rodas
1	Regulador de tensão
2	Bateria 18650
2	Micro servo 9g

Fonte: o autor, 2023