

UNIVERSIDADE DE CAXIAS DO SUL  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIAS  
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

GIORDANO COLLEONI

DESENVOLVIMENTO DE UM PROTÓTIPO APLICADO A  
CONSTRUÇÃO DE UM ROBÔ MÓVEL AUTÔNOMO DO TIPO AMR

CAXIAS DO SUL  
2022

**Giordano Colleoni**

**DESENVOLVIMENTO DE UM PROTÓTIPO APLICADO A  
CONSTRUÇÃO DE UM ROBÔ MÓVEL AUTÔNOMO DO TIPO AMR**

Trabalho de Conclusão de Curso apresentado  
como parte dos requisitos para a obtenção do  
título de Engenheiro de Controle e Automa-  
ção da Universidade de Caxias do Sul.

Orientador:  
Prof. Felipe Augusto Tondo

**Caxias do Sul  
2022**

Ficha catalográfica.

COLLEONI, Giordano

DESENVOLVIMENTO DE UM PROTÓTIPO APLICADO A CONSTRUÇÃO DE UM ROBÔ MÓVEL AUTÔNOMO DO TIPO AMR/Giordano Colleoni; Orientador: Prof. Felipe Augusto Tondo

57 folhas, 30 cm.

Trabalho de conclusão do curso de Engenharia de Controle e Automação — Universidade de Caxias do Sul, Centro de Ciências Exatas e Tecnologias, 2022

Inclui referências bibliográficas.

1. Robô Móvel Autônomo, AMR, Sistema de Guiamento. I. REFERENCIAR ORIENTADOR II. Universidade de Caxias do Sul. III. DESENVOLVIMENTO DE UM PROTÓTIPO APLICADO A CONSTRUÇÃO DE UM ROBÔ MÓVEL AUTÔNOMO DO TIPO AMR.

**Giordano Colleoni**

**DESENVOLVIMENTO DE UM PROTÓTIPO APLICADO A  
CONSTRUÇÃO DE UM ROBÔ MÓVEL AUTÔNOMO DO TIPO AMR**

Trabalho de Conclusão de Curso apresentado  
como parte dos requisitos para a obtenção do  
título de Engenheiro de Controle e Automa-  
ção da Universidade de Caxias do Sul.

Orientador:  
Prof. Felipe Augusto Tondo

Aprovado em \_\_\_\_/\_\_\_\_/\_\_\_\_\_

**Banca Examinadora**

-----  
Prof. Felipe Augusto Tondo (orientador)  
Universidade de Caxias do Sul - UCS

-----  
Prof. Alexandre Mesquita  
Universidade de Caxias do Sul - UCS

-----  
Prof. Wagner Grison  
Universidade de Caxias do Sul - UCS

*Dedico este trabalho a todo o curso de engenharia de Controle e Automação da Universidade de Caxias do Sul, corpo docente e discente, a quem fico lisonjeado por dele ter feito parte. A minha esposa Victória, minha razão de viver e maior apoiadora.  
Dedico este trabalho a todos os que me ajudaram ao longo desta caminhada.*

## AGRADECIMENTOS

Agradeço ao apoio, atenção, ensinamentos e dedicação do Prof. Felipe Augusto Tondo e a todos os professores e colegas que trouxeram seu melhor para tornar este meio acadêmico cada vez mais agradável.

Agradeço a minha esposa Victória por todo o apoio e pela ajuda, que muito contribuiu para a realização deste trabalho.

Agradeço aos meus pais e amigos, que incentivaram nos momentos difíceis e compreenderam a minha ausência.

*“A coragem não é a ausência de medo, e sim a capacidade de agir apesar dele”.*  
John McCain

## RESUMO

A quarta revolução industrial trouxe conceitos amplamente abordados atualmente, entre essas novas tecnologias, estão os veículos autônomos que são cada vez mais utilizados na indústria para transporte de materiais e apoio à produção. O uso destes permite uma cadência de ritmo de entrega de produtos fixa, com vantagens associadas ao menor custo e menos problemas de qualidade. Perante isso, este trabalho apresenta o desenvolvimento de um protótipo para um Robô Móvel Autônomo (AMR) a ser utilizado na indústria. Neste documento é abordado o projeto de controle de um sistema de guiamento de um AMR. Gerenciamento de sensores, controle do tracionamento das rodas e sistemas de navegação são discutidos, com o principal objetivo de idealizar e construir um protótipo funcional, viável e de construção nacional. A partir de trajetos definidos são testados sua repetibilidade, em percursos lineares, curvos e mistos, bem como sua velocidade e sua capacidade de carga são validadas, por fim, verifica-se a detecção de obstáculos e testes de parada para garantir a segurança de operação do mesmo, resultando em um veículo autônomo, de baixo custo de implementação e funcional.

**Palavras-chave:** Robô Móvel Autônomo, AMR, Sistema de Guiamento.

## ABSTRACT

The fourth industrial revolution brought concepts widely discussed today, among these new technologies are autonomous vehicles that are increasingly used in industry to transport materials and support production. The use of these allows for a fixed product delivery rhythm cadence, with advantages associated with lower cost and fewer quality problems. In view of this, this work presents the development of a prototype for an Autonomous Mobile Robot (AMR) to be used in industry. This document discusses the control project of an AMR guidance system. Sensor management, wheel drive control and navigation systems are discussed, with the main objective of idealizing and building a functional, viable and nationally built prototype. From defined routes, their repeatability is tested, in linear, curved and mixed routes, as well as their speed and load capacity are validated, finally, obstacle detection and stop tests are verified to guarantee operational safety of the same, resulting in an autonomous vehicle, with low implementation cost and functional.

**Keywords:** Autonomous Mobile Robot, AMR, Guidance System.

## LISTA DE FIGURAS

Figura 1 – Robô Manipulador Industrial KUKA . . . . .	18
Figura 2 – Representação de um AGV . . . . .	19
Figura 3 – Topologias mais habituais para AGV. . . . .	19
Figura 4 – Representação dos sistemas do AMR . . . . .	20
Figura 5 – Representação do Sistema de filoguiamento . . . . .	21
Figura 6 – Representação do sistema de faixas . . . . .	22
Figura 7 – Representação do Sistema fléxivel de triangulação . . . . .	22
Figura 8 – Detecção de área utilizável para locomoção (estrada) . . . . .	23
Figura 9 – Trajeto de um veículo representado pela Odometria . . . . .	24
Figura 10 – Representação de um IMU e suas funções. . . . .	25
Figura 11 – Representação das maneiras de tração e direção . . . . .	26
Figura 12 – Modelo de Scanner da marca SICK . . . . .	28
Figura 13 – Ponte H . . . . .	29
Figura 14 – Representação do Diagrama de blocos da implementação do projeto . . . . .	30
Figura 15 – Croqui do Projeto Mecânico . . . . .	31
Figura 16 – Microcontrolador NodeMCU-32 . . . . .	32
Figura 17 – Fluxograma do Método de Controle do projeto . . . . .	32
Figura 18 – Modelo do sensor HC-sr04 . . . . .	33
Figura 19 – Ponte H Modelo BTS7960 . . . . .	34
Figura 20 – Motor AK360 . . . . .	34
Figura 21 – Funcionamento do Sensor Foto-Elétrico . . . . .	35
Figura 22 – Erro do Sensor Foto-Elétrico . . . . .	36
Figura 23 – Vista Isométrica do Modelamento 3D . . . . .	37
Figura 24 – Vista Explodida do Modelamento 3D . . . . .	37
Figura 25 – Circuito de Ligação do HC-SR04 . . . . .	39
Figura 26 – Posicionamentos dos HC-SR04 . . . . .	39
Figura 27 – Montagem do Sensor Odométrico . . . . .	41
Figura 28 – Montagem da placa de Alimentação e Distribuição 21x5V . . . . .	41
Figura 29 – Esquemático da placa de Alimentação e Distribuição 21x5V . . . . .	42
Figura 30 – Representação do modelo 3D da vista superior da PCB . . . . .	42
Figura 31 – Protótipo finalizado. . . . .	43
Figura 32 – Trajeto Retilíneo . . . . .	44
Figura 33 – AMR Posicionado para iniciar Trajeto . . . . .	44
Figura 34 – Posição Inicial do Rodizio de Apoio . . . . .	45
Figura 35 – Trajeto Curvo . . . . .	45
Figura 36 – Trajeto Curvo . . . . .	46
Figura 37 – Trajeto Misto . . . . .	46
Figura 38 – Trajeto do teste de carga . . . . .	47
Figura 39 – Dispositivo com carga aplicada . . . . .	47

Figura 40 – Curva de Velocidade x Peso . . . . .	48
Figura 41 – Area de Monitoramento . . . . .	48
Figura 42 – Teste de Parada . . . . .	49
Figura 43 – Teste de Parada . . . . .	49

## LISTA DE TABELAS

Tabela 1 – Características dos sistemas de tração . . . . .	26
Tabela 2 – Características dos microcontroladores . . . . .	27
Tabela 3 – Número de linhas do programa base . . . . .	27
Tabela 4 – Sensor HC-SR04 . . . . .	33
Tabela 5 – Lista de componentes do projeto . . . . .	38
Tabela 6 – Medições dos HC-SR04s . . . . .	40
Tabela 7 – Aplicação do Filtro Mediana nas medições . . . . .	40
Tabela 8 – Orçamentos Realizados . . . . .	50

## LISTA DE ABREVIATURAS

AMR	<i>Autonomous Mobile Robot</i>
CAD	<i>Computer Aided Desing</i>
FFT	<i>Fast Fourier Transform</i>
GND	<i>Ground</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IMT	<i>Inertial Measurement Unit</i>
I/O	<i>Input/Output - Entrada/Saída</i>
LED	<i>Light Emitting Diode</i>
M	<i>Metro</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
PID	<i>Controlador Proporcional Integral e Derivativo</i>
USD	<i>Dólar Americano</i>
WiFi	<i>Wireless Fidelity</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>15</b>
1.1	JUSTIFICATIVA	15
1.2	OBJETIVOS DO TRABALHO	16
<b>1.2.1</b>	<b>Objetivo Geral</b>	<b>16</b>
<b>1.2.2</b>	<b>Objetivos Específicos</b>	<b>16</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>17</b>
2.1	O INÍCIO DA ROBÓTICA	17
2.2	ROBÔS MÓVEIS AUTÔNOMOS E SEMI-AUTÔNOMOS NA INDÚSTRIA	18
2.3	SISTEMAS DE POSICIONAMENTO	20
<b>2.3.1</b>	<b>Medidas Absolutas</b>	<b>21</b>
2.3.1.1	Sistema de filoguiamento	21
2.3.1.2	Sistema de reconhecimento de faixas	21
2.3.1.3	Sistema Flexível	22
2.3.1.4	Sistema de Visão Artificial	23
<b>2.3.2</b>	<b>Medidas Relativas</b>	<b>23</b>
2.3.2.1	Sistema de Odometria	24
2.3.2.2	Sistema Inerciais	24
2.4	SISTEMAS DE TRAÇÃO	25
2.5	COMPONENTES ELETRÔNICOS	26
<b>2.5.1</b>	<b>Microcontroladores</b>	<b>26</b>
<b>2.5.2</b>	<b>Scanner</b>	<b>28</b>
<b>2.5.3</b>	<b>Driver de motor</b>	<b>28</b>
<b>3</b>	<b>METODOLOGIA PARA IMPLEMENTAÇÃO DO PROTÓTIPO</b>	<b>30</b>
3.1	METODOLOGIA DE CONTROLE	30
3.2	PROJETO MECÂNICO	31
3.3	MICROCONTROLADOR	31
3.4	SENSOR ULTRASSÔNICO	32
3.5	DRIVER DO MOTOR	33
3.6	MOTOR	34
3.7	MONITORAMENTO GERAL E MONITORAMENTO WIFI	34
3.8	ANÁLISE DO MOVIMENTO	35
3.9	DETALHAMENTO MECÂNICO	36
3.10	CALIBRAÇÃO DOS COMPONENTES ELETRÔNICOS	38
<b>3.10.1</b>	<b>Sensor Ultrassônico HC-SR04</b>	<b>39</b>
<b>3.10.2</b>	<b>Sensor Odométrico</b>	<b>41</b>
<b>3.10.3</b>	<b>Placa de Alimentação e Distribuição 21x5V.</b>	<b>41</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>43</b>
4.1	TESTE PRÁTICO DE TRAJETÓRIA	43
<b>4.1.1</b>	<b>Trajetoária Retilínea</b>	<b>43</b>
<b>4.1.2</b>	<b>Trajetoária Curva</b>	<b>45</b>
<b>4.1.3</b>	<b>Trajetoária Mista</b>	<b>46</b>
4.2	TESTES DE CARGA E VELOCIDADE	46
4.3	TESTES DE SEGURANÇA	48

4.4	ORÇAMENTOS . . . . .	49
<b>5</b>	<b>CONCLUSÃO . . . . .</b>	<b>51</b>
5.1	TRABALHOS FUTUROS . . . . .	51
	<b>REFERÊNCIAS . . . . .</b>	<b>52</b>

# 1 INTRODUÇÃO

Segundo Kotze (2008), o problema de manuseio de materiais no campo industrial consiste em como transportar matérias-primas, produtos em andamento e bens manufaturados entre diferentes locais de produção de sistemas de fabricação, armazéns entre outros locais. Dependendo do tipo de produto para manusear e do transporte para realizar, existem diferentes soluções no mercado, como: o sistema de correias, rolo e transportadores verticais, os elevadores, os veículos guiados automatizados (AGVs) e, um pouco mais tecnológicos, os Robôs Móveis Autônomos (AMRs).

Os sistemas automatizados possuem algumas vantagens inerentes ao seu design conforme citado por Maarif e Moyo (2019), sendo as principais como: a redução do produto, a capacidade de deslocamento para áreas perigosas sem preocupação com a segurança do operador, a capacidade de rastrear e registrar automaticamente o trajeto do produto e a redução de trabalho. Além disso, características como flexibilidade e adaptabilidade fazem desses sistemas automatizados uma ferramenta importante quanto a eficiência da produção.

Este assunto já é abordado por Das e Pasan (2016a) que utiliza certo tipo de controle para produzir sinais analógicos com o microcontrolador e controlar a velocidade de rotação dos motores das rodas, além do sistema de visão do dispositivo. Outro ponto importante desses sistemas autônomos, são seus módulos de posicionamento de direção que guiam os trajetos a serem percorridos pelos mesmos.

Diante do contexto apresentado acima, é proposto um protótipo de um Robo Móvel Autônomo a ser utilizado na indústria, sendo previsto o desenvolvimetro de um controle de um sistema de posicionamento que, a partir dos sinais recebidos pelos sensores de posicionamento presentes no AMR, faça o controle de tração dos motores presentes nas rodas, bem como mensure possíveis desalinhamentos e variações nos movimentos das rodas através de encoders para corrigir em tempo real a trajetória.

## 1.1 JUSTIFICATIVA

A crescente demanda de transportes de carga de forma autônoma utilizando um agregado de tecnologias existentes no mercado, possibilita a fomentação de criação e implementação de produtos nacionais que atendam essa demanda, pois atualmente as soluções existentes são massivamente importadas, e com isso está vinculado um alto preço nos produtos com um prazo de entrega demasiado grande.

Inovando em robôs para transportes de carga, surge a criação de um protótipo de um AMR, fabricado nacionalmente. Além disso pode-se criar uma expertise tecnológica na indústria em sistemas de transporte e movimentação de cargas, pelo fato deles serem uma evolução tecnológica e possuírem um melhor controle na movimentação que os seus ante-

cessores, os AGVs, ajudando a contornar a escassez de fornecedores em âmbito nacional, podendo melhorar no planejamento e prazo de entrega dessas soluções.

Bem como, justifica-se este projeto ser desenvolvido por um graduando em Engenharia de Controle e Automação, pois engloba-se as diversas áreas do curso, seja no desenvolvimento da parte mecânica, na estratégia da montagem elétrica dos componentes no protótipo, ou ainda na parte de programação do mesmo.

## 1.2 OBJETIVOS DO TRABALHO

Neste capítulo aborda-se o objetivo geral do trabalho bem como posteriormente os objetivos específicos.

### 1.2.1 Objetivo Geral

Implementação de um protótipo para desenvolvimento de um robô móvel autônomo (AMR) a ser utilizado na indústria. Além disso, a implementação prevê a criação do módulo de controle do sistema de posicionamento deste robô.

### 1.2.2 Objetivos Específicos

Para se atingir o objetivo geral pretende-se:

- Pesquisar sobre aplicações de *AMRs* na indústria;
- Pesquisar e definir os sensores de posicionamento que serão aplicados no módulo de controle;
- Definir a plataforma microcontroladora para aquisição e processamento dos sinais;
- Propor uma metodologia de trabalho para implementação do módulo de controle de um sistema de posicionamento para robôs móveis autônomos;
- Desenvolver o projeto mecânico do protótipo a ser construído;
- Implementar um algoritmo de posicionamento e trajetória do robô AMR;
- Avaliar a repetibilidade nos testes funcionais, com a criação de uma trajetória padrão;

## 2 REFERENCIAL TEÓRICO

Este capítulo apresenta os conceitos básicos de robótica móvel, de instrumentação e controle de veículos autônomos, com ênfase nas novas tecnologias computacionais utilizadas pela indústria, abordando as principais partes que compõem o AMR.

### 2.1 O INÍCIO DA ROBÓTICA

A robótica é uma área do conhecimento que gradativamente tem evoluído nos últimos anos, porém, sabe-se que o estudo e o projeto de robôs vêm sendo desenvolvidos há vários séculos. Segundo Jung et al. (2005) é difícil estabelecer a data de origem dos primeiros autômatos (ancestrais dos atuais robôs), mas pode-se estabelecer um marco na história da humanidade como sendo a construção da primeira máquina de calcular, criada por Pascal em 1642, pois a partir desta época, inúmeros autômatos foram desenvolvidos, sendo somente a partir de 1923 que o termo “robot” começou a ser empregado, sendo usado pela primeira vez por Karel Capek (CAPÁK et al., 2019).

A fim de se obter flexibilidade e volume de produção em 1959, George Devol e Joseph Engelbergerm apresentaram o primeiro robô industrial e dois anos mais tarde é instalado para manipulação de peças quentes retiradas de um processo de fundição. Desde então se desencadeou uma revolução na indústria de processos principalmente com a implantação de novos dispositivos e tecnologias, consolidando como necessários os manipuladores robóticos, (IFR, 2013).

A literatura apresenta várias maneiras de classificar os robôs industriais que segundo Filho (2006) podem variar de acordo com a forma, tipo de atividade, geometria, entre outros, conforme suas particularidades. Contudo, como a proposta deste trabalho enfatiza um robô móvel autônomo, não serão apresentados todos os modelos industriais fixos, apenas será contextualizado de maneira geral, pois os mesmos o antecedem e de certa forma contribuíram para sua criação.

A Figura 1 apresenta o robô do tipo manipulador industrial com 6 eixos de movimentação, com base fixa. Somente no último século, surgiram os robôs móveis, que possuem a capacidade de se deslocar de modo guiado, de forma semi-autônoma ou autônoma. Apesar de existirem referências históricas mais antigas de dispositivos móveis comandados remotamente, pode-se considerar que sua origem foi influenciada pelo desenvolvimento de artefatos militares, como por exemplo, veículos autônomos teleguiados ou mesmo guiados como as bombas desenvolvidas durante a 2<sup>a</sup> Guerra Mundial (JUNG et al., 2005).

Figura 1 – Robô Manipulador Industrial KUKA



Fonte: (KUKA, 2021).

## 2.2 ROBÔS MÓVEIS AUTÓNOMOS E SEMI-AUTÓNOMOS NA INDÚSTRIA

Segundo Montiel, González e Sepúlveda (2010) um robô movel procura conjugar num só dispositivo a mobilidade de um veículo autónomo e a capacidade de manuseamento e manipulação dos robôs convencionais, como os braços fixos vistos anteriormente. A utilização de robótica móvel em áreas na indústria e na logística é dominada por dois tipos de dispositivos: o AGV e o AMR.

Como o trabalho de Souza e Royer (2013) relata, o AGV é uma das tecnologias mais interessantes para transportes de materiais nas indústrias, principalmente no que diz respeito ao pagamento de produção (logística), coletar o material do armazém e transportá-lo até a linha de montagem. Seu transporte é seguro e eficaz, trazendo grandes benefícios à produção e aos usuários deste sistema.

O AGV tem locomoção autónoma, feito através de baterias, o que lhes permite trabalhar 24 horas por dia, sendo assim mais eficiente do que o sistema tradicional feito por equipamentos manuais com necessidade de intervenção humana. Além disso, há um melhor controle devido às operações serem todas computadorizadas, bem como os dados compilados de volumes de material, tendo assim uma visão mais real do estoque e da logística da fábrica em tempo real. Kim, Tanchoco e Koo (1999). Pode-se ver um dos modelos de AGV na Figura 2.

Figura 2 – Representação de um AGV

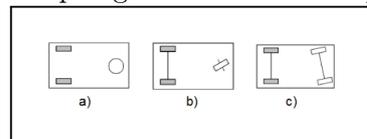


Fonte: (SOUZA; ROYER, 2013). Adaptado.

Na literatura existem diversas topologias de estruturas para robôs móveis terrestres, muitas utilizadas por AGVs, que se aplicam a diversas situações. Bastos et al. (2019) em seu trabalho ilustra várias estruturas possíveis de veículos autônômicos terrestres, sendo as mais utilizadas destacadas na Figura 3. Estas serão melhor explicadas nos parágrafos que seguem.

Pode-se ver três topologias diferentes de veículos na indústria, a primeira possui tração independente, ou seja cada roda é motorizada com um motor, possibilitando diferentes velocidades para cada roda tracionada, as rodas escuras, com a estabilidade de um ponto de apoio, representado pelo círculo claro. A segunda topologia representada consiste no tracionamento traseiro único, com a direção sendo definida pelo ângulo em que a roda dianteira se encontra. Por fim, a terceira tipologia consiste basicamente no mesmo ideal da segunda, com a adição de mais uma roda dianteira, para aumentar a estabilidade em curvas.

Figura 3 – Topologias mais habituais para AGV.



Fonte: (BASTOS et al., 2019). Adaptado.

Contudo, como Carneiro, Bandeira et al. (2021) salientam, os AGVs são robôs que operam em ambientes especialmente tratados para sua operação, tornando-se inflexíveis e frágeis no que diz respeito ao trajeto frente a possíveis alterações no ambiente, sendo considerados robôs de “autonomia fraca”. Contrastante a esta característica está a capacidade dos robôs móveis autônomos (AMRs) de alterar suas ações de acordo com as interpretações processadas do ambiente em que estão inseridos, sendo por esta razão considerados robôs de “autonomia forte”.

A fundamental diferença entre um AGV e um AMR consiste no princípio básico no funcionamento do sistema de navegação. Enquanto o primeiro utiliza um sistema rígido

composto por uma rota pré-definida, normalmente utilizando condutores localizados no chão, que é seguido por um dispositivo de leitura. Os AMRs utilizam uma rota definida por um nível mais alto de tecnologia, usando para-choques de segurança, dispositivos ópticos de varredura, sensores de obstáculos, dispositivos de alerta de áudio e visual, conexão Wi-Fi, módulos GPS, entre outros, que possibilitam uma definição de múltiplas possíveis trajetórias, fornecendo assim, através das informações fornecidas por estes sensores, a melhor rota possível (JUNG et al., 2005).

Segundo Das e Pasan (2016b), como pode ser visualizado na Figura 4, os principais componentes do sistema AMR são: (a) Veículo em cinza; (b) Sistema de posicionamento em amarelo e (c) Sistema de Controle da tração em verde. A fim de entender a melhor forma e o que é necessário para projetar o módulo de guiamento nos próximos capítulos, são abordados dois desses principais sistemas, o sistema de posicionamento, em amarelo na imagem, que é composto pelos sensores e métodos que rastreiam e informam a posição do robô, e o sistema de tração, em verde na imagem, que é composto mecanicamente pela força motriz que traciona as rodas e algum sensor de natureza odométrica, como um encoder, por exemplo.

Figura 4 – Representação dos sistemas do AMR



Fonte: Autor.

### 2.3 SISTEMAS DE POSICIONAMENTO

Os métodos, fórmulas e sensores que compõem os sistemas de posicionamentos dos robôs autônomos é atualmente estudado e explorado, existindo no mercado uma variedade de soluções. Segundo Moreira (2014) a escolha da solução mais indicada depende de uma diversidade de fatores, tais como as condições do ambiente, a flexibilidade solicitada e os custos associados à implementação do sistema.

Ainda Moreira (2014), exemplifica dois tipos de medições para esses sistemas de nave-

gação e posicionamento, a diferença entre as mesmas consiste na relação com o referencial onde estas se efetuam. No primeiro tipo, nas medições absolutas, a medida é feita em relação à origem do ambiente onde o sistema robótico se encontra. De outro modo, as medidas relativas são feitas em referência a um ponto, neste caso, o referencial do robô, estas são caracterizadas por estarem relacionadas à dinâmica do robô.

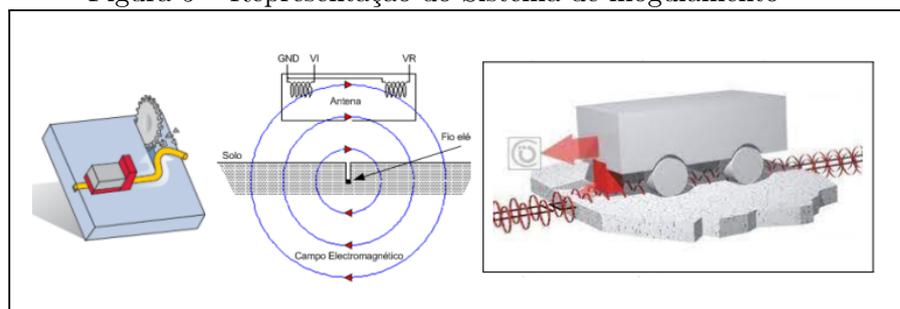
### 2.3.1 Medidas Absolutas

Nesta subseção discute-se os principais sistemas de medição com referencial absoluto, aquele que é obtido em relação a origem do ambiente onde o AMR se encontra.

#### 2.3.1.1 Sistema de filoguiamento

De acordo com Santos (2013) sistemas filoguiados consistem no seguimento, por parte do robô, de um condutor enterrado a poucos centímetros no chão, o qual é percorrido por uma corrente elétrica, criando desta forma um campo magnético. O campo criado é detectado por uma antena colocada no robô móvel, muito próxima ao chão, onde o condutor está, permitindo ao robô uma correção contínua da sua trajetória, de forma que este não se perca tão facilmente. No entanto, Lima et al. (2010) ressalta que estes sistemas estão limitados ao uso de velocidades controladas, de forma que o sistema robótico nunca deixe de sentir o efeito do campo magnético. A Figura 5 representa um sistema de filoguiamento, demonstrando o fio enterrado no chão e o campo gerado pelo condutor com a antena receptora do veículo.

Figura 5 – Representação do Sistema de filoguiamento



Fonte: (LIMA et al., 2010).

#### 2.3.1.2 Sistema de reconhecimento de faixas

Nos sistemas baseados em faixas, as trajetórias são definidas por fitas magnéticas ou linhas coloridas pintadas no chão. A detecção da linha é feita por meio de um sensor, que detecta o campo magnético da fita ou a mudança de gradiente. Este sistema é bastante semelhante ao filoguiado, no entanto possibilita uma troca de trajeto mais prática e rá-

pida, por consequencia acarretando em menores custos (ARAÚJO et al., 2017).

No entanto, em ambientes muito movimentados, este sistema revela algumas desvantagens, por exemplo, na durabilidade das marcas implementadas, uma vez que as fitas danificam-se mais facilmente e as linhas pintadas estão expostas a maior sujidade, diminuindo a eficiência no seu seguimento por parte do robô. Na Figura 6 apresenta-se um exemplo do sistema de reconhecimento de faixas.

Figura 6 – Representação do sistema de faixas



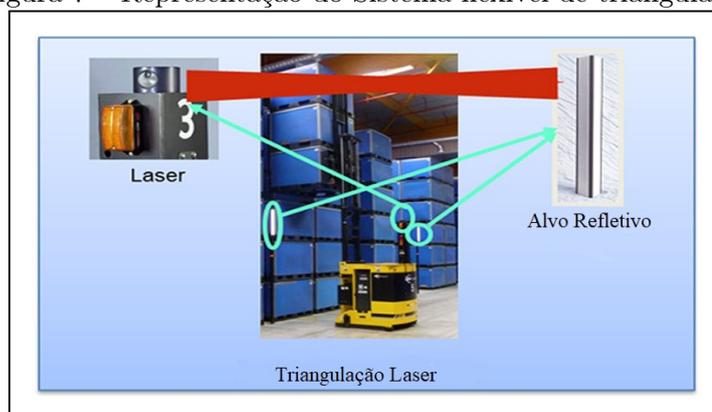
Fonte: (ARAÚJO et al., 2017). Adaptado.

### 2.3.1.3 Sistema Flexível

Dentro das trajetórias flexíveis, um método bastante adotado na indústria é a navegação baseada em triangulação e/ou trilateração. Este método consiste na detecção da localização do robô através de postes ou faróis refletores, colocados em postes ou paredes, e no uso de um laser rotativo, que permite a detecção destes refletores (MOREIRA, 2014).

De forma a efetuar a sua localização, é necessário que o laser consiga detectar pelo menos três refletores. Sendo a posição de cada refletor constante, o laser permite obter a distância do robô a cada um dos refletores, bem como a sua orientação relativamente aos mesmos, permitindo uma localização eficaz e precisa do sistema robótico. Pode-se ver um exemplo disto citado na Figura 7.

Figura 7 – Representação do Sistema flexível de triangulação



Fonte: (MOREIRA, 2014). Adaptado.

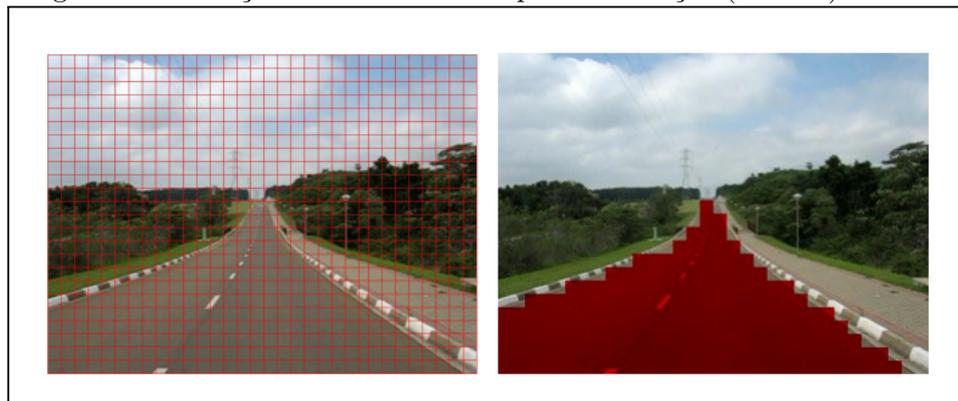
### 2.3.1.4 Sistema de Visão Artificial

Sistemas robóticos baseados em visão artificial despertam interesse na comunidade científica, levando ao aparecimento de diversas metodologias para a localização destes sistemas.

O trabalho de Lee, Hyun e Park (2013) sugere um sistema de localização baseado em marcadores de referência, recorrendo a uma câmara web para detectar estes marcadores. Nos marcadores encontra-se uma letra maiúscula ou um triângulo, indicando a direção. Sendo assim um sistema de baixo custo de implementação. Quando o marcador, de tamanho conhecido, é visualizado pelo robô, a relação posicional entre o robô e o marcador pode ser calculada. Recorrendo a transformadas de distância é possível reconhecer os caracteres presentes nos marcadores. Verificou-se que com quatro triângulos direcionais e dez letras foi possível um nível de reconhecimentos na ordem dos 98.87%, tornando este um sistema de localização viável para ambientes interiores.

O trabalho de Cucchiara, Perini e Pistoni (2007) sugere a utilização de técnicas de estereoscopia. Primeiramente são pintadas linhas paralelas ou perpendiculares aos eixos coordenados, possibilitando a correção da orientação do robô. Posteriormente, a câmara detecta as imagens e assim torna-se possível calcular a distância. Desta forma, consegue-se corrigir a posição do veículo robótico ao longo do seu trajeto. Na Figura 8 pode-se ver o mapeamento de uma estrada, o sistema identifica o local onde o veículo pode navegar, o qual visualmente pode ser visto em vermelho.

Figura 8 – Detecção de área utilizável para locomoção (estrada)



Fonte: (SHINZATO, 2010). Adaptado.

### 2.3.2 Medidas Relativas

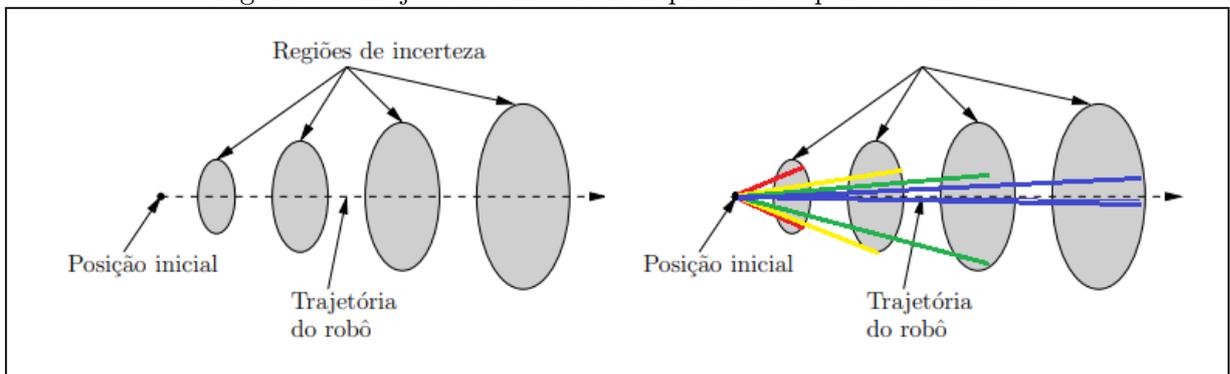
Nesta subsecção discute-se os principais sistemas de medição com referencial relativo, aquele que é obtido em relação a um ponto referenciado apenas pelo robô e não um ponto geral no ambiente onde o AMR se encontra.

### 2.3.2.1 Sistema de Odometria

A odometria é um dos métodos mais utilizados para estimar a posição de um robô, sendo mais viável que alguns dos demais sistemas apresentados neste capítulo para implementar, além de permitir taxas de amostragem altas. Uma vez que a odometria consiste na integração incremental do movimento das rodas ao longo do tempo, o erro associado a esta medição é desta forma acumulado, originando possíveis erros na estimação da posição do robô.

Apesar do seu principal inconveniente, o erro acumulativo citado, considera-se a odometria uma importante parte do sistema de navegação, tendo esta, um papel importante com o sensoramento, com medidas de posicionamento absolutas, tornando a estimação da posição mais viável. Para exemplificar, na Figura 9, há oito linhas coloridas, todas iniciando da posição inicial, sendo quatro duplas demonstrando possíveis trajetotos com finais diferentes, ou seja, as duas linhas vermelhas possuem o mesmo comprimento, mas não acabam no mesmo ponto, assim como as amarelas, verdes e azuis. Isso demonstra que a odometria consegue fornecer uma possível área onde o veículo pode estar, cabendo a um segundo sistema de medição fornecer esta informação mais precisa. (PEREIRA, 2019)

Figura 9 – Trajeto de um veículo representado pela Odometria



Fonte: (BEZERRA, 2004). Adaptado.

### 2.3.2.2 Sistema Inerciais

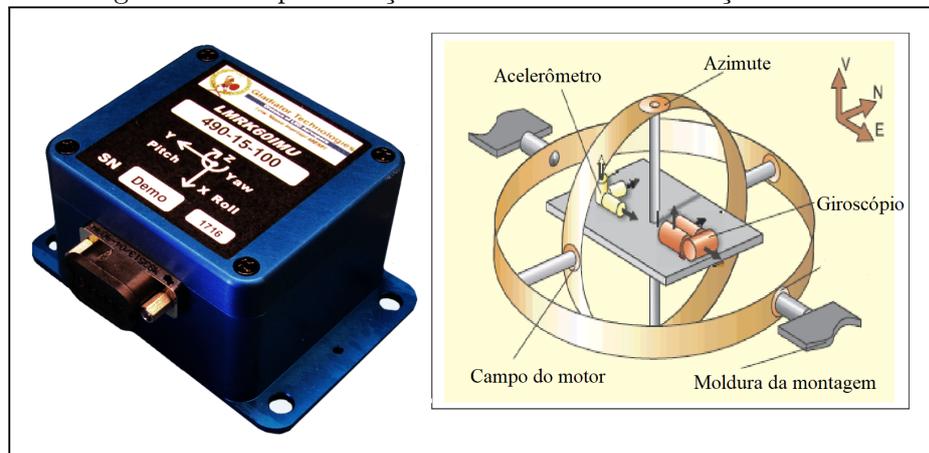
Outra forma de ter conhecimento da posição de um robô, recorrendo a medidas relativas, é com recursos a IMUs. Um IMU é um dispositivo eletrônico que mede a velocidade, orientação e forças que atuam sobre um corpo, sendo este, o componente principal de sistemas de navegação baseados em forças inerciais. Os principais componentes de um IMU são giroscópios e acelerômetros. Pode-se ver a representação destes componentes na Figura 10.

O trabalho de Zmuda, Elesev e Morton (2008) destaca algumas desvantagens associa-

das a este tipo de medição. Um fator importante a ser considerado são as derrapagens a que o robô se encontra sujeito, induzindo este sistema a erros. O autor ainda enfatiza, que na ocorrência de movimentos bruscos, pode ocorrer uma instabilidade no sensor levando alguns segundos para retornar aos seus valores baixos.

Alguns IMUs utilizam magnetômetros, que baseiam suas medições no campo magnético terrestre, estes não recomendados para a aplicação, pois desta forma, os materiais ferromagnéticos presentes na estrutura dos edifícios e na construção do robô (motores, estrutura, etc) influenciam grandemente nas medições por este obtidas.

Figura 10 – Representação de um IMU e suas funções.



Fonte: (ZMUDA; ELESEV; MORTON, 2008). Adaptado.

## 2.4 SISTEMAS DE TRAÇÃO

Segundo Das e Pasan (2016a) existem quatro maneiras básicas de fornecer propulsão e direção. A Figura 7 e a Tabela 1 apresentam estas quatro formas. Alguns tipos são mais fáceis de construir, outros têm melhores características de direção. Dois desses projetos são totalmente simétricos e podem ser operados em ambas as direções. Neste projeto, o AMR em análise possuirá o sistema de Tracionamento Central com Apoio, assim possuindo motor independente em cada roda. Esse sistema foi escolhido devido à facilidade de controle e precisão do movimento.

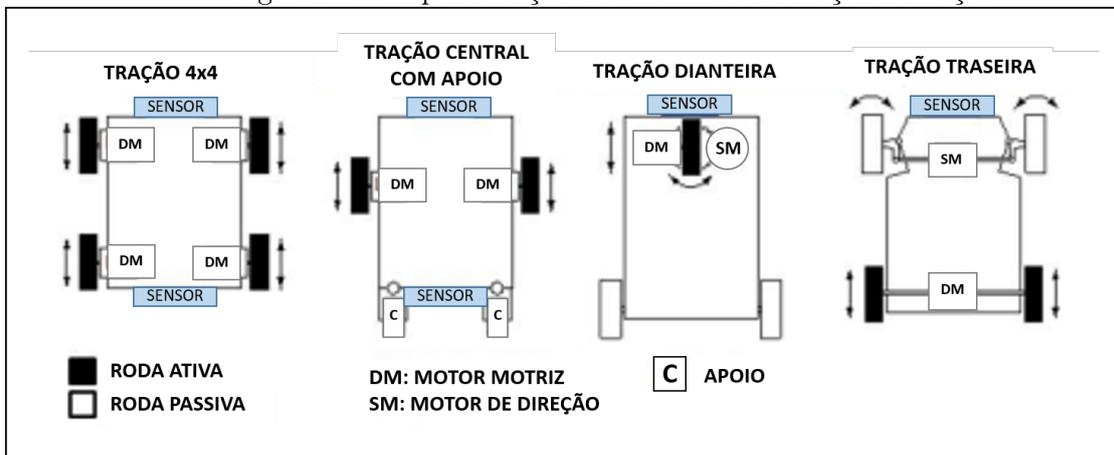
No sistema de acionamento central, os movimentos retos ou movimentos de giro são regulados pela velocidade de rotação de cada motor. Para movimentos retos, a velocidade motora da direita e esquerda deve ser a mesma, enquanto para o movimento de giro a velocidade do motor para a direita e para a esquerda é distinguida. Por exemplo, se o AMR virar para a direita, a velocidade do motor é ajustada para que o motor direito seja mais lento do que o motor esquerdo para que a mecânica do conjunto se movimente para a direita, e vice-versa.

Tabela 1 – Características dos sistemas de tração

Denominação	Simplicidade	Dirigibilidade	Reversão
Tração 4x4	Muito Simples	Pouco Precisa	Sim
Tração Central com Apoio	Simple	Precisa	Sim
Tração Dianteira	Mediano	Muito Precisa	Não
Tração Traseira	Complexo	Precisa	Difícil

Fonte: (DAS; PASAN, 2016b) Adaptado.

Figura 11 – Representação das maneiras de tração e direção



Fonte: (DAS; PASAN, 2016b) Adaptado.

## 2.5 COMPONENTES ELETRÔNICOS

Nesta seção discute-se os principais componentes eletrônicos que estarão presentes no módulo de controle do sistema de posicionamento projetado.

### 2.5.1 Microcontroladores

Microcontroladores são determinados microprocessadores que podem ser programados para diversas funções específicas por incluírem processador, memória e controle de periféricos de entrada e saída. É constante o seu uso na área da automação residencial, bem como no monitoramento e gerência de dados provenientes de sensores diversos na indústria, visando trazer agilidade e maior eficiência na tarefa de troca de dados (RODRIGUES, 2013).

Os trabalhos de Sharp e Vagapov (2017) e Maier, Sharp e Vagapov (2017) fazem análises de alguns microcontroladores mais comuns utilizados para pequenas automações e aplicações tecnológicas, dentre eles, pode-se citar, ESP32, ESP 8266 e CC3220. Conforme

pode-se verificar na Tabela 2 o ESP32 tem o maior poder de processamento e número de portas, tendo um baixo custo em relação ao desempenho oferecido.

Tabela 2 – Características dos microcontroladores

Detalhes	ESP 32	ESP 8266	CC3220
CPU	32 bit dual-core 160/240 MHz	32 bit singe-core 80 a 160 MHz	ARM CortexM4 80 MHz
SRAM	520 KB	36 KB	256 KB
FLASH	2 MB	4 MB	1 MB
Tensão	2,2 a 3,6 V	3 a 3,6 V	2,3 a 3,6 V
Programação	(C,C++,Lua)	(C,C++,Lua)	C
Open Source	Sim	Sim	Não
WIFI	802.11	802.11	802.11
Bluetooth	4.2BR+BLE	-	-
UART	3	2	2
GPIO	32	17	21
PWM	8	-	6
ADC	18 (12 bit)	1 (10 bit)	4 (12 bit)
DAC	2(8 bit)	-	-
Tamanho	25x18x3 mm.	24x16x3 mm.	21x18x2,5mm.
Preço	\$5	\$7	\$18

Fonte: (SHARP; VAGAPOV, 2017) Adaptado.

Uma outra vantagem dos modelos ESP em relação aos demais microcontroladores é sua possibilidade de programação em diversas linguagens, assim pode-se utilizar a linguagem com melhor performance e menor tamanho para cada aplicação, visto por Rak e Wiora (2021) em seu trabalho. Conforme a Tabela 3 demonstra, a linguagem C++ possui um menor número de linhas em comparação a linguagem C, além de ser possível também, programar em Lua, caso queira ter uma redução ainda maior de linhas.

Tabela 3 – Número de linhas do programa base

Nº Teste	Nome do Teste	C++	Lua	MP	C
1	Blink LED	12	12	8	37
2	Serial	8	3	5	32
3	ADC	10	6	8	36
4	Servo	13	10	7	43
5	Servidor	47	15	15	150

Fonte: (RAK; WIORA, 2021) Adaptado.

### 2.5.2 Scanner

Scanner é um sensor que utiliza tecnologia *time-of-flight* (tempo de percurso), que significa que calcula o tempo que leva para o laser refletir em um objeto e retornar ao scanner, pode ser montado em um veículo ou carro de transferência conduzido automaticamente, para eliminar o risco de colisões com objetos ou pessoas no ambiente (KEYENCE, 2022).

É válido mencionar que para todo projeto de movimentação a legislação solicita que haja um monitoramento de segurança, afim de evitar colisões e acidentes. Um scanner cumpre esta função, como o modelo da marca SICK na Figura 12 (BERLIN; GRANATH, 2021).

Figura 12 – Modelo de Scanner da marca SICK



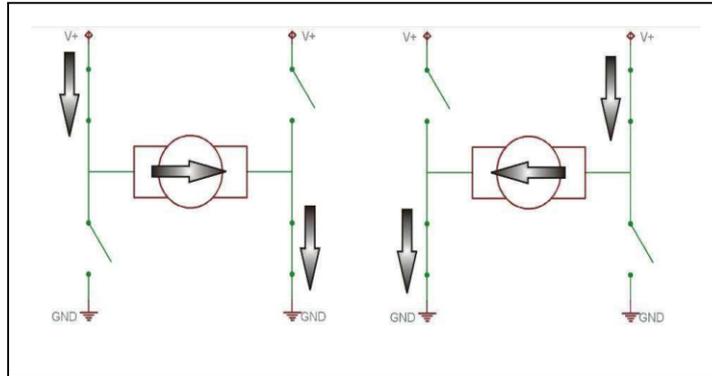
Fonte: (BERLIN; GRANATH, 2021).

### 2.5.3 Driver de motor

A função de um motor elétrico é converter energia elétrica em energia mecânica. Existem comercialmente motores de corrente contínua (CC) e de corrente alternada (CA), cada um com diversas variações de tamanho, força e conseqüentemente consumo. Motores de corrente alternada são geralmente usados para maquinário de grande porte, pois recebem energia diretamente da rede de distribuição elétrica, por outro lado, robôs móveis usam tipicamente corrente contínua, pois sua fonte de energia é uma bateria (SILVA, 2016).

Os motores presentes no projeto precisam ser acionados nos dois sentidos, portanto, há a necessidade de controlar o sentido da corrente fornecida para que o motor inverta o sentido de rotação. Esse controle é feito pela ponte H, Figura 13, que controla o sentido da corrente para o motor através da disposição de quatro chaves eletrônicas em que o motor é posicionado entre elas, de modo que formem uma letra “H”, por esse motivo o seu nome (MUNIZ et al., 2022).

Figura 13 – Ponte H



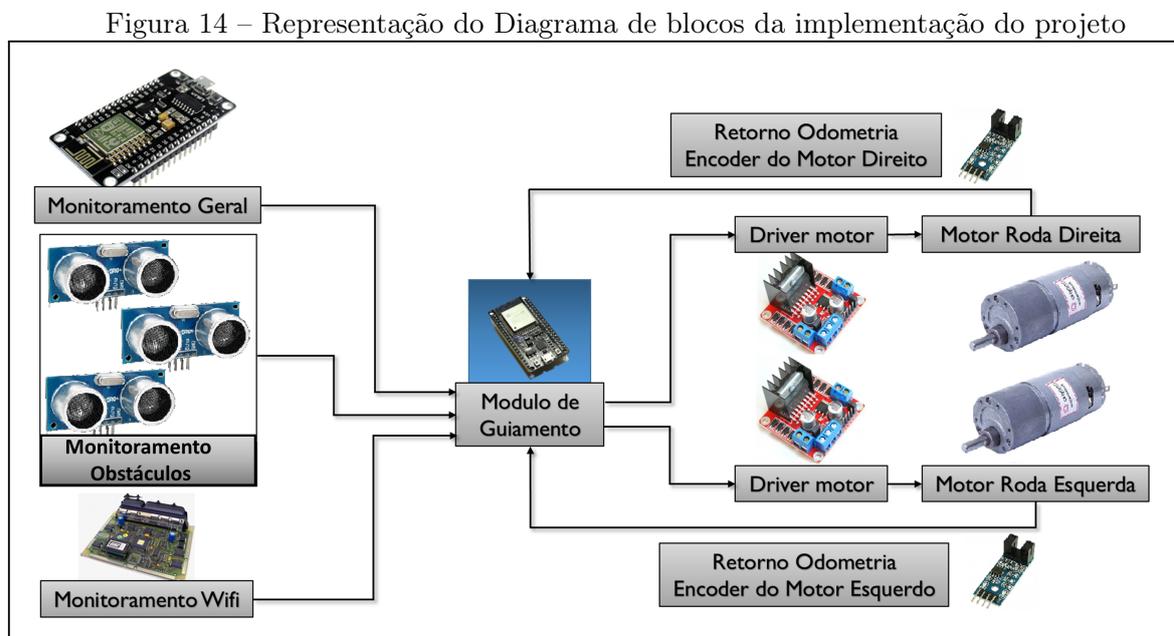
Fonte: (MUNIZ et al., 2022).

### 3 METODOLOGIA PARA IMPLEMENTAÇÃO DO PROTÓTIPO

Nesta seção serão descritas as etapas do desenvolvimento do protótipo, bem como do projeto e controle do sistema de guiamento do AMR, usando controle de sensores e controle do tracionamento das rodas. A seguir será apresentada a descrição da metodologia utilizada no projeto

#### 3.1 METODOLOGIA DE CONTROLE

A Figura 14 representa, em diagrama em blocos, a implementação do projeto da perspectiva de *hardware* e comunicação.<sup>1</sup>



Fonte: Autor.

O objetivo é implementar um veículo protótipo que seja semelhante ao produto original construído posteriormente, o qual consiste em uma estrutura metálica de aproximadamente 450 x 450 mm com capacidade de carregar até 10 kg, com a topologia de tracionamento central com duas rodas e mais apoio. Possuirá os sensores e dispositivos presentes neste trabalho, consistindo em um veículo com capacidade de locomoção autônoma, que será testado em um ambiente controlado. Maiores informações construtivas serão abordadas nos tópicos que seguem.

<sup>1</sup>Inicialmente definiu-se a utilização de um sistema de visão com scanner para detecção de obstáculos. Devido à falta de acesso ao recurso que surgiu no decorrer do trabalho optou-se por trocar por um sistema ultrassônico

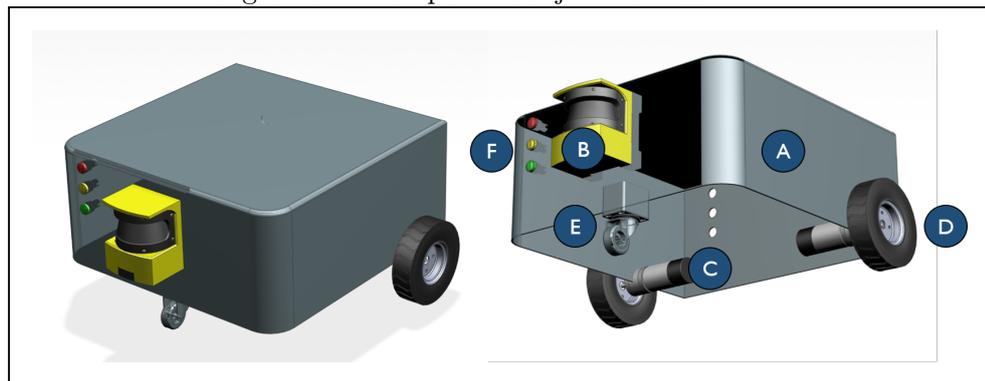
### 3.2 PROJETO MECÂNICO

O projeto mecânico será elaborado pelo autor, tanto a etapa relacionada ao projeto 3D desenvolvido em *software* CAD e detalhamento para produção, quanto a montagem manual de todos os componentes do projeto.

Na concepção estrutural do projeto foi utilizado chapas de aço de baixo carbono SAE 1010/1020, que são o tipo de aço mais utilizado na indústria para estruturas mais simples que não necessitam de alta resistência mecânica e/ou a corrosão. Serão utilizadas chapas na espessura de 1,2 a 3 mm. com dobras e abas para montagem por parafusos, eliminando assim o processo de solda, buscando facilitar a montagem mecânica.

O croqui do projeto 3D do veículo pode ser visto na Figura 15. Pode-se ver de forma simplificada: (a) Estrutura de aço; (b) Scanner; (c) Motor com Encoder; (d) Rodas de Tração; (e) Rodizio de apoio e (f) LEDs de operação.

Figura 15 – Croqui do Projeto Mecânico



Fonte: Autor.

### 3.3 MICROCONTROLADOR

O microcontrolador a ser utilizado será a placa NodeMCU-32S, visível na Figura 16, pelo fato do controlador possuir capacidade de processamento para a aplicação com a finalidade de auxiliar no desenvolvimento do projeto e apresentar uma solução funcional, conforme Akbar (2021) e Saha et al. (2021) comprovaram em seus projetos.

O microcontrolador além de conter o ESP32, conta com um cristal de 40MHz, antena embutida e porta micro USB para alimentação e programação, além de pinos GPIOs, Conversor Analógico Digital com 12 bits de resolução, 2 DAC com 8 bits de resolução, além de possuir conexão WiFi nativa e Bluetooth V4.2, grande diferencial perante a outros microcontroladores de mesmo valor, oferecendo assim baixo consumo de energia e alto desempenho de potência a um custo acessível (PEREIRA; OLIVEIRA, 2019).

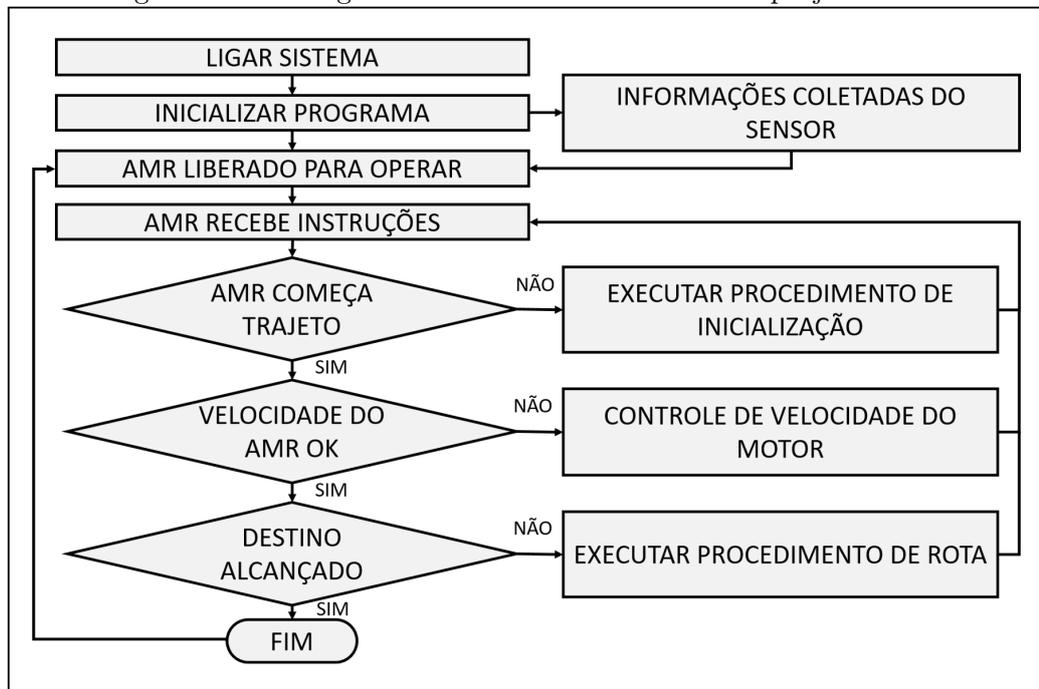
Figura 16 – Microcontrolador NodeMCU-32



Fonte: (PEREIRA; OLIVEIRA, 2019).

O fluxo de comunicação das informações ocorre pela solicitação/envio de comandos ou parâmetros para o controlador que é responsável pela leitura e controle de um sensor de referenciamento, encoder e atuador. O fluxograma apresentado na Figura 17 divide o método em etapas principais e subdivisões.

Figura 17 – Fluxograma do Método de Controle do projeto

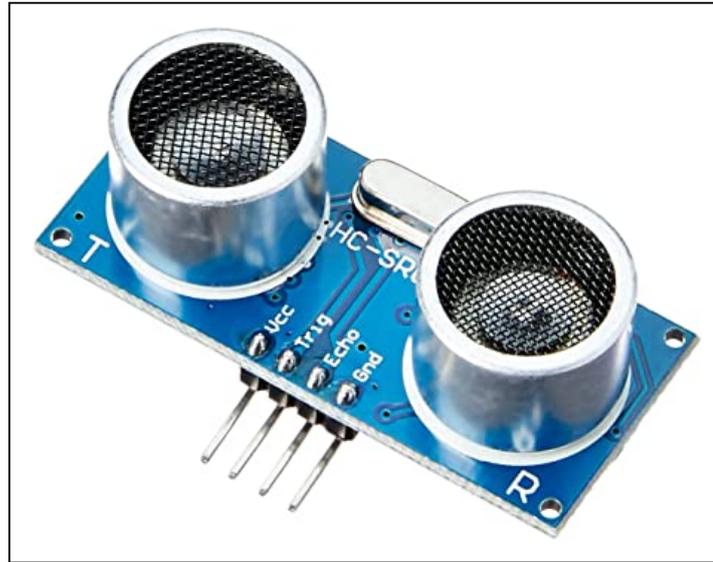


Fonte: Autor.

### 3.4 SENSOR ULTRASSÔNICO

O sensor ultrassônico HC-SR04 aplicado ao projeto está ilustrado na Figura 18 e tem suas principais características listadas na Tabela 4.

Figura 18 – Modelo do sensor HC-sr04



Fonte: (LOUTFIA et al., 2017)

Tabela 4 – Sensor HC-SR04

Recurso	Especificação	Unidade
Alimentação	5	VDC
Corrente de Operação	2	mA.
Angulo Alcance	+/- 15	º
Resolução linear	3	mm.
Zona Cuidado Máx.	4	m.
Zona Perigo Mín.	3	cm.

Fonte: 4Hobby

Primeiramente o sensor ultrassônico irá fazer a leitura do retorno da propagação do som e indicar se há algum obstáculo presente na área frontal próxima ao veículo, ocasionando a alteração de trajeto, caso o obstáculo esteja na Zona de Cuidado ou até mesmo a paralização do veículo caso esteja na Zona de Perigo.

### 3.5 DRIVER DO MOTOR

Para desenvolvimento do projeto foi escolhido o modelo BTS7960, pela facilidade em adquirir o item, bem como a sua grande utilização no ramo de controle de motores, que pode ser evidenciado por Syukriyadin et al. (2018). O Módulo Driver Ponte H - BTS7960 trabalha com um motor de tensão de 5 a 45V, com tensão lógica de 3.3 a 5V, suportando no máximo 43A. Na Figura 19 pode-se ver o modelo escolhido para a prototipagem.

Figura 19 – Ponte H Modelo BTS7960



Fonte: (FILIPEFLOP, 2022).

### 3.6 MOTOR

Considerando o acesso ao recurso, bem como suas características técnicas foi escolhido o Modelo AK360 da fabricante Akiyama, possui caixa de redução já integrada, resultando em alto torque e baixa rotação ideal para o projeto. Este motor selecionado opera em 12 a 24 VDC, possui velocidade nominal de 39 RPM, velocidade máxima de 44 rpm, torque nominal de 2,2 kgcm com potência máxima de 2w, usando até 0,5 A, com um peso de 160g. Pode ser visto na Figura 20.

Figura 20 – Motor AK360



Fonte: (4Hobby, 2022).

### 3.7 MONITORAMENTO GERAL E MONITORAMENTO WIFI

Em seguida explica-se sobre o monitoramento geral e o monitoramento WiFi que compõem a implementação do módulo. O bloco de monitoramento geral consiste na inicialização do sistema e a leitura dos sensores e instruções de modo a liberar o AMR para operação, bem como parar a operação por fatores de segurança ou pelo comando do

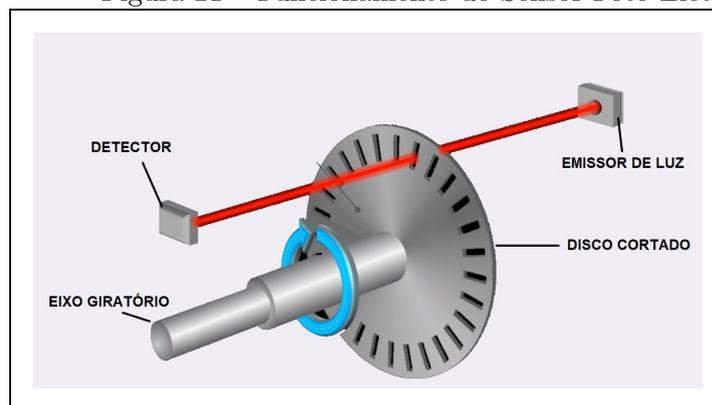
operador de desligar o veículo, por exemplo em um fim de turno, ou para colocá-lo para carregar.

O bloco Wifi consiste no recebimento e envio de informações pela rede, no projeto o microcontrolador ESP32 já possui integrado conexão WiFi Nativa, facilitando este processo. Assim, podendo utilizar o protocolo MQTT, por ser um protocolo de mensagens de simples concepção, baixa exigência de recursos computacionais e baixos requisitos de banda de Internet, tornando-o um dos principais protocolos de comunicação com as maiores plataformas IoT do mercado (AI et al., 2021).

### 3.8 ANÁLISE DO MOVIMENTO

Afim de entender e analisar o movimento do robo móvel proposto, deve-se evidenciar mais minuciosamente um dos sistemas a ser utilizado, para ter-se um controle de direcionamento, basicamente, na parte do sistema de Odometria, será utilizado um sensor foto-eletrico, que simula de forma mais simples o funcionamento de um encoder incremental. Como pode-se ver na Figura 31, o sensor consiste em um emissor de luz, um detector e um disco cortado, com cortes equidistantes de forma radial que é acoplado ao eixo da roda. Quando o disco girar o feixe de luz emitido sera receptado a cada vez que passar por cada corte presente no disco, assim, se o disco possui 30 recortes, quando o receptor receber os 30 pulsos, é sinal de que a roda girou 360 graus.

Figura 21 – Funcionamento do Sensor Foto-Elétrico



Fonte: (REALPARS, 2022).

Apartir disso pode-se concluir pelas equações que se seguem, a distância percorrida pela roda, partindo do cálculo do perímetro da roda, representado pela primeira equação, posteriormente verifica-se a resolução do disco do sensor, sabendo quantos recortes o disco no projeto possui, calcula-se quanto cada recorte do disco corresponde em mm.

$$P = 2.\pi.R \quad (3.1)$$

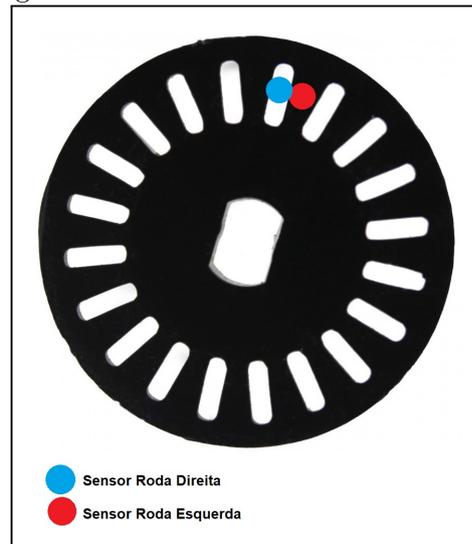
$$RE = P/N \quad (3.2)$$

Onde,  $P$  é o perímetro da roda,  $R$  é o raio,  $N$  é o número de recortes no disco e  $RE$  é a resolução do sensor. Após isso, multiplicando o valor de  $RE$  pela quantidade de pulsos  $Q$  que o sensor recebeu, é possível identificar a distância percorrida  $D$  pela roda dada por:

$$D = RE.Q \quad (3.3)$$

Apartir do que foi explicado, pode-se implicar um erro relacionado na medição em relação da posição do feixe de luz no disco, considerando as duas possibilidades representadas na Figura 22, em que cada uma represente uma roda do robô móvel, e que percorra um movimento retilíneo, em que basicamente as duas rodas irão girar igualmente, é possível ter uma diferença de mais ou menos um no número de pulsos recebidos pelos sensores.

Figura 22 – Erro do Sensor Foto-Elétrico



Fonte: Autor.

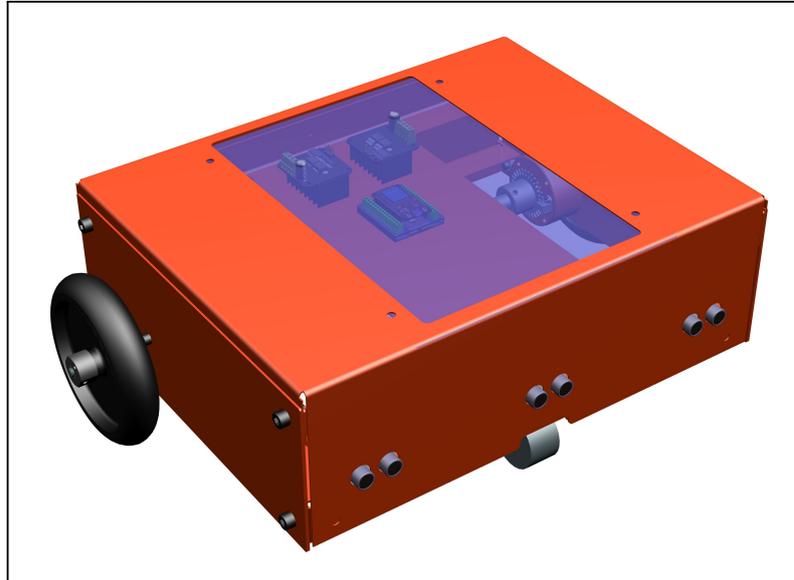
Assim, a diferença, ou o erro do sensor é um  $RE$ , ou seja a própria resolução do sensor em mm, implicando na distância (mm) em que a roda direita pode ter percorrido a mais que a esquerda, ou vice-versa, é também por este motivo que não é possível se ter uma precisão tão eficaz com a odometria, pois este erro pode ser confundido com derrapagens das rodas por exemplo, acarretando em não poder fazer correções de rota em medidas sensibilizadas pelos sensores menores que  $1 RE$ .

### 3.9 DETALHAMENTO MECÂNICO

O projeto mecânico do AMR foi elaborado por meio do *software* Nx CAD, modelado tridimensionalmente e posteriormente analisado e validado. Para a produção das peças cortadas à laser, foram realizados os DWG's e PDF's dos mesmos, e dos itens usinados,

foram feitos os PDF's com descrição do material e medidas necessárias. Pode-se ver na Figura 23 a vista isométrica renderizada do conjunto.

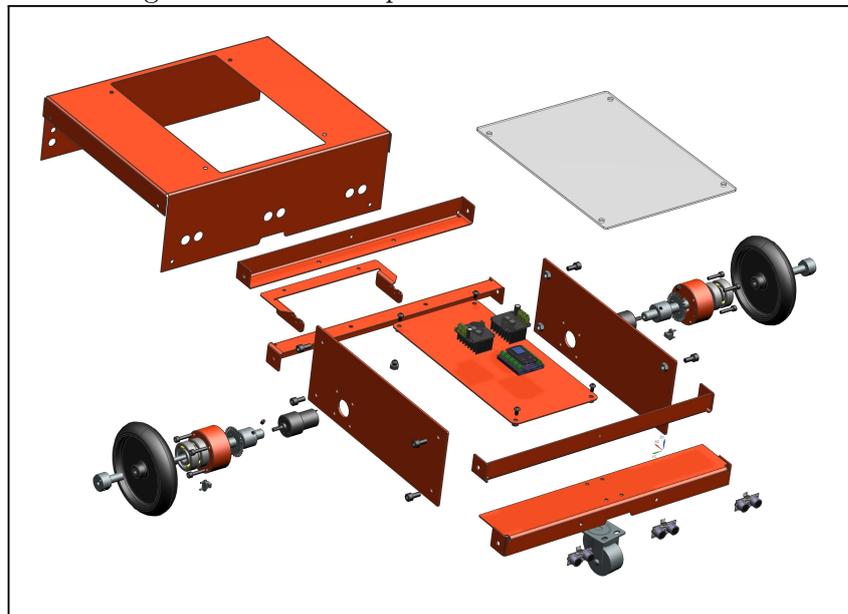
Figura 23 – Vista Isométrica do Modelamento 3D



Fonte: Autor.

Pode-se ver na Figura 24 a vista explodida do conjunto. O programa gera uma lista de componentes automaticamente, com código do item, descrição e quantidade necessária para o projeto, na Tabela 5. pode ser visto todos esses itens.

Figura 24 – Vista Explodida do Modelamento 3D



Fonte: Autor.

Tabela 5 – Lista de componentes do projeto

Código	Descrição	Quantidade
100000430996	Regua	1
100000424153	Regua	3
100000424139	Lateral Interna	2
100000424599	Bucha Rolamento	2
100000426064	Chapa Base Elétrico	1
100000426878	Rolamento 6303	4
100000298206	Parafuso Allen M6x25	8
10000013956	Parafuso Allen M8x15	8
10000013956	Porca Sextavada M8	10
10000430977	Rodizio Livre Ø2"	1
10000298456	Parafuso Cabeça Abaulada M6x10	6
210001923	Porca Sextavada M6	4
210001923	Porca Sextavada M8 Auto-frenante	8
100000431198	Eixo Roda	2
100000431196	Roda Ø6"4151	2
100000433941	Bucha Eixo	2
100000441273	Motor Yohama	2
100000442321	ESP32	1
100000442340	Driver BTS 7960H	2
100000441272	Sensor Odométrico	2
10000298242	Parafuso Cabeça Abaulada M3x6	4
10000298787	Parafuso sem cabeça M6x8	4
100000443980	Chapa Fixação Motores	1
100000444247	Sensor Ultrassônico HC-Sr04	3
100000451991	Fechamneto	1
100000456806	Placa Policarbonato	1

Fonte: Autor.

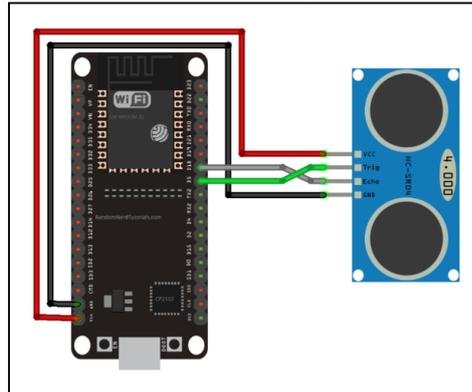
### 3.10 CALIBRAÇÃO DOS COMPONENTES ELETRÔNICOS

Afim de garantir o funcionamento do projeto, uma importante etapa é a calibração e validação dos sensores utilizados no projeto. Os itens que se seguem tratam da aferição dos sensores ultrassônicos, dos sensores odométricos e a definição e construção da sua placa de alimentação.

### 3.10.1 Sensor Ultrassônico HC-SR04

Para testar os sensores ultrassônicos, foi utilizado a biblioteca padrão *ultrasonic.h*, a qual facilita e implementa as suas funções básicas, o código pode ser visto em Apêndice A. Pode-se ver na Figura 25 o esquema de ligação do sensor com a placa ESP32.

Figura 25 – Circuito de Ligação do HC-SR04



Fonte: Autor.

Para validar a medição do sensor, colocou-se o veículo a uma distância pré-determinada de uma parede de forma inclinada, a uma distância de 65 cm, 66 cm e 67 cm respectivamente de cada sensor. Para posicioná-lo em relação a parede, utilizou-se uma trena métrica, visto que a sua resolução e seu erro são de 1 mm, dez vezes mais preciso que a resolução do sensor ultrassônico. Conforme Figura 26.

Figura 26 – Posicionamentos dos HC-SR04



Fonte: Autor.

Apartir da leitura que aparece na porta serial, pode-se ver a medição lida pelo sensor, percebe-se alguns valores que se diferem dos demais, conforme demarcados em vermelho, que podem ser vistos na Tabela 6.

Tabela 6 – Medições dos HC-SR04s

Sensor Esquerdo	Sensor Central	Sensor Direito
55	65	64
66	64	64
66	55	64
67	64	64
66	55	64
66	63	64
66	56	64
67	64	65
66	53	63
66	65	63
67	65	65

Fonte: Autor.

Afim de validar as medições dos sensores na operação do veículo, foi aplicado um filtro, que consiste na leitura de 5 medições e posteriormente a aplicação da mediana nestas medições, de modo a ignorar possíveis interferências no sistema e evitar uma instabilidade, o código pode ser visto em Apêndice B. Conforme na Tabela 7, as primeiras cinco colunas, correspondem a medições do sensor, já ordenadas de forma crescente, sendo que os valores mais distoantes estão nas extremidades, e serão desconsiderados, utilizando-se apenas o valor da terceira coluna. Percebe-se como exemplo, na sétima linha, uma medição realizada de 53 cm, entende-se que esta medição consistiu de alguma interferência, pois é extremamente mais alta que as medições anteriores, utilizando-se do filtro, esta medida distoante não será considerada.

Tabela 7 – Aplicação do Filtro Mediana nas medições

1ª Med	2ª Med	3ª Med	4ª Med	5ª Med	Med Considerada
52	52	52	52	52	52
9	9	10	10	11	10
5	5	6	6	6	6
29	30	30	30	31	30
25	26	28	38	52	28
52	52	52	52	52	52
3	3	4	5	53	4
2	3	3	4	4	3

Fonte: Autor.

### 3.10.2 Sensor Odométrico

Para testar os sensores odométricos, foi utilizado o sistema de interrupção do arduino, pelo comando *attachInterrupt*, o qual aciona um contador a cada vez que a entrada definida recebe um pulso, a aplicação pode ser vista em Apêndice C. Os dois sensores utilizados no projeto foram montados um em cada lado do protótipo, um montado no eixo da roda esquerda e outro no eixo direito, conforme pode-se ver na Figura 27.

Figura 27 – Montagem do Sensor Odométrico

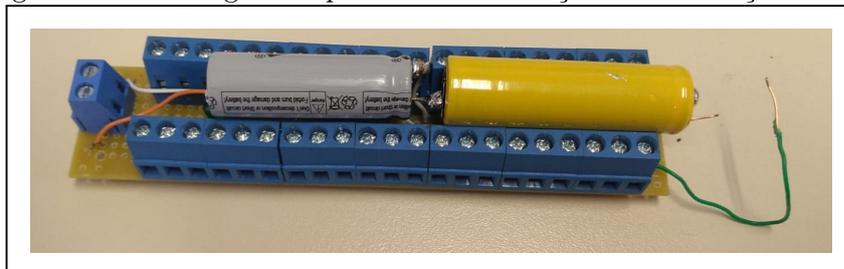


Fonte: Autor.

### 3.10.3 Placa de Alimentação e Distribuição 21x5V.

Na implementação do protótipo, verificou-se que era necessário energizar pelo menos quinze pinos, entre a alimentação dos sensores e do arduino. Pode-se ver na Figura 28 a montagem em placa perfurada, possuindo vinte e uma saídas de 5V e GND, além de possuir um circuito acoplado a uma chave que habilita o carregamento das baterias ou energiza as saídas ditas anteriormente.

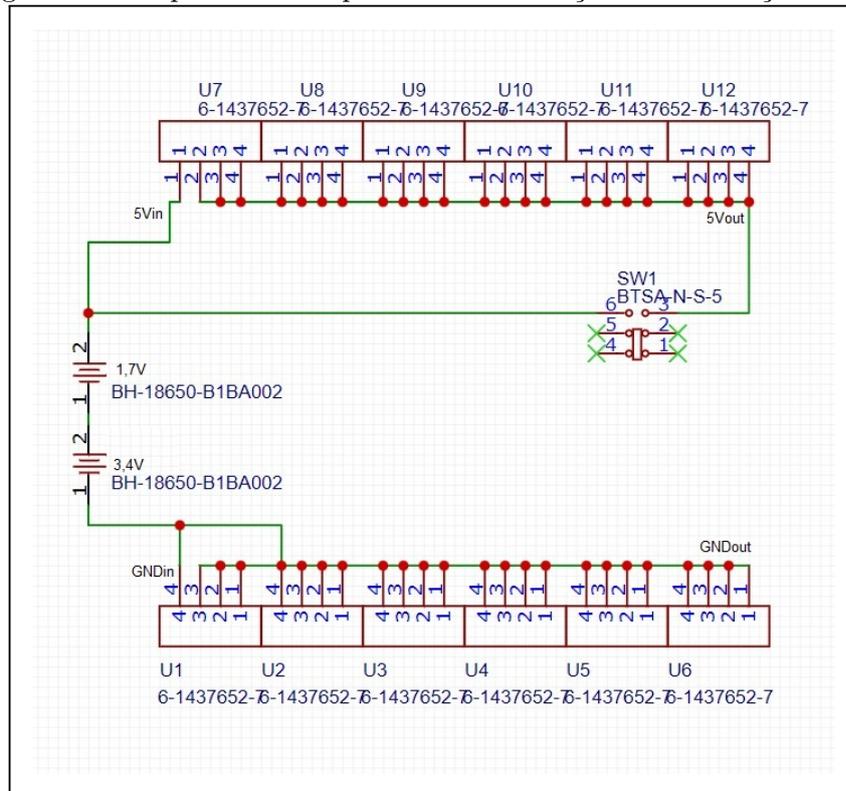
Figura 28 – Montagem da placa de Alimentação e Distribuição 21x5V



Fonte: Autor.

O esquemático elétrico da placa foi desenvolvido pelo ambiente da plataforma EasyEDA, conforme é possível observar na Figura 29.

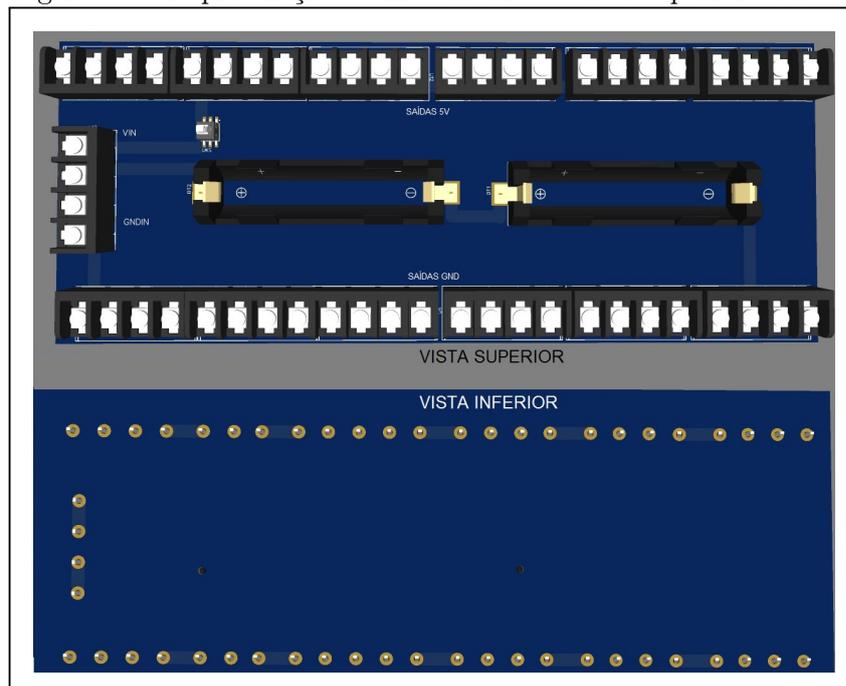
Figura 29 – Esquemático da placa de Alimentação e Distribuição 21x5V



Fonte: Autor.

O PCB da placa foi desenvolvido pelo ambiente da plataforma EasyEDA, conforme é possível observar na Figura 30.

Figura 30 – Representação do modelo 3D da vista superior da PCB



Fonte: Autor.

## 4 RESULTADOS E DISCUSSÕES

Neste capítulo serão apresentados os resultados experimentais conforme a metodologia descrita. Na primeira etapa serão apresentados os testes de validação individual de cada bloco do circuito e os resultados da caracterização da saída. Na segunda etapa será apresentado e discutido os resultados de testes práticos e seus parâmetros finais resultantes. A Figura 31 apresenta o protótipo implementado.

Figura 31 – Protótipo finalizado.



Fonte: Autor.

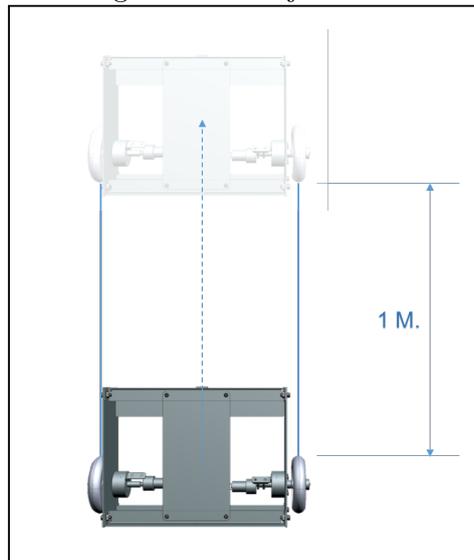
### 4.1 TESTE PRÁTICO DE TRAJETÓRIA

Para verificar possíveis desalinhamentos e validar os comandos básicos de movimentação dos motores e sensores, foram traçados e testadas três trajetórias as quais o AMR deverá seguir. O veículo sem transparência demonstra o ponto de partida e o com transparência demonstra o ponto de chegada, cada figura possui algumas cotas de modo a orientar o que está sendo verificado e o que é importante mensurar. Ao lado de cada figura aparece a comparação do ponto final ideal e real com o erro expresso em mm.

#### 4.1.1 Trajetória Retilínea

Esse teste consiste no deslocamento linear do AMR por um metro, conforme visto na Figura 32.

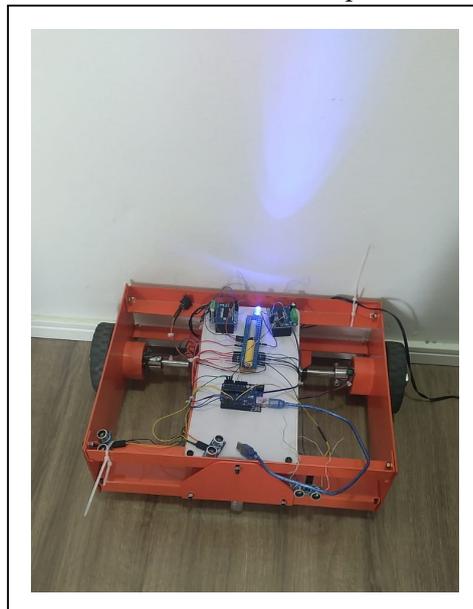
Figura 32 – Trajeto Retilíneo



Fonte: Autor.

Para realizar esse teste o protótipo foi posicionado de costas, encostado em uma parede para ter-se uma referência fixa, conforme visto na Figura 33.

Figura 33 – AMR Posicionado para iniciar Trajeto



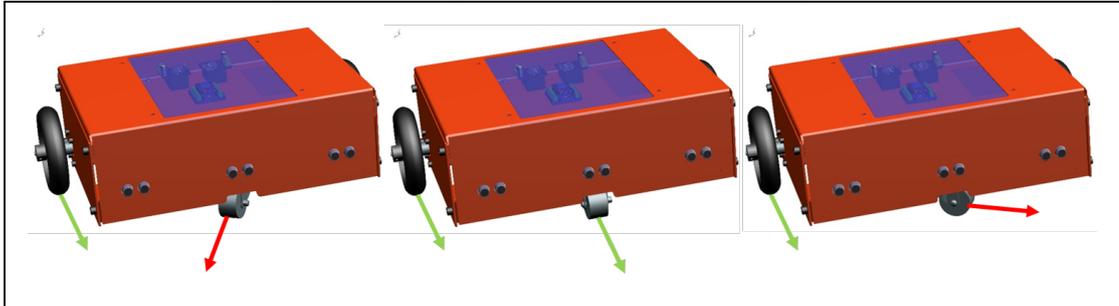
Fonte: Autor.

O teste foi repetido e ajustado conforme verificava-se desalinhamentos, baseando-se na análise das medições dos encoders. Após os resultados estabilizarem e os últimos cinco trajetos terem o mesmo erro, com uma variação entre eles menor que 1 cm, o teste foi finalizado e definiu-se o erro.

Um ponto importante observado, que influencia nos resultados do trajeto percorrido, é a posição do rodízio de apoio, por ele não ser uma esfera e sim um rodízio giratório,

o mesmo oferece um atrito e um deslocamento de modo a aumentar o desalinhamento e consequentemente o erro do sistema. Na Figura 34 pode-se ver 3 condições diferentes para o início desse teste de trajetória, dependendo para qual lado o mesmo está posicionado inicialmente.

Figura 34 – Posição Inicial do Rodizio de Apoio



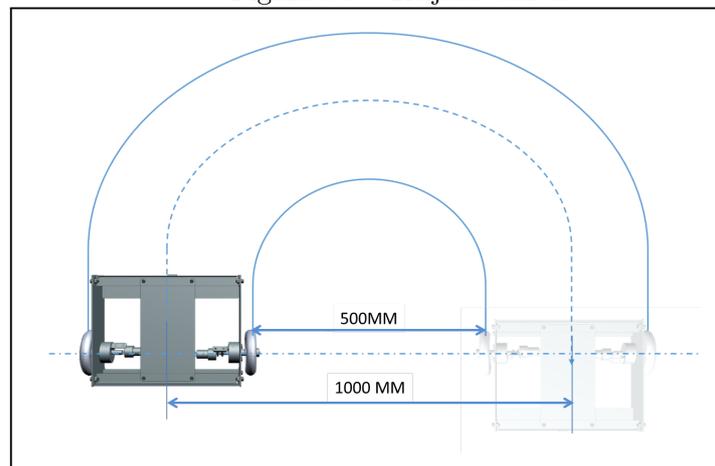
Fonte: Autor.

No teste de trajeto retilíneo, o AMR percorreu o trajeto com um erro final médio de 3%, ou 30 mm. Este erro pode ser causado por algum desalinhamento presente na estrutura das rodas, e até mesmo uma pequena folga no sistema de tração entre eixos e motores. A importância de validar este trajeto é fundamental para entender como será dado o percurso da trajetória.

#### 4.1.2 Trajetória Curva

Em veículos de transporte de cargas, principalmente os industriais, é comum ter-se qual o raio mínimo de curvatura que é possível percorrer, conforme pode-se ver na Figura 35. Foi definido um raio mínimo de curvatura de 1000 mm, medido da linha média da curva do trajeto.

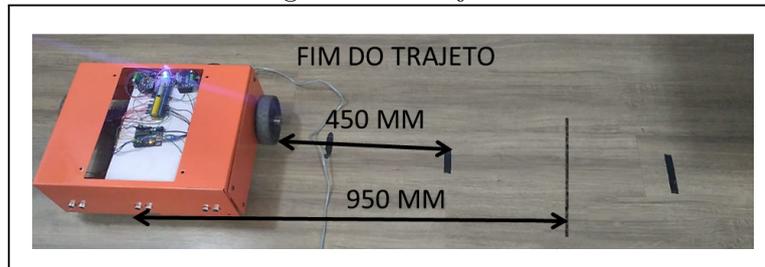
Figura 35 – Trajeto Curvo



Fonte: Autor.

Foram feitas tentativas considerando este valor, resultando em um raio de 950 mm, sem se ter grandes arrastes e desvios das rodas, deste modo o protótipo atendeu ao raio mínimo especificado anteriormente. Pode-se ver o mesmo na Figura 36.

Figura 36 – Trajeto Curvo

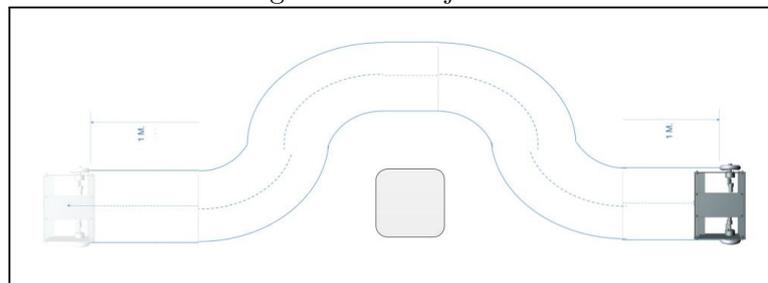


Fonte: Autor.

#### 4.1.3 Trajetória Mista

O teste consiste na junção do trajeto linear do AMR por um metro, e posteriormente a detecção de um obstáculo realizando o desvio para chegar ao destino final, conforme visto na Figura 37.

Figura 37 – Trajeto Misto



Fonte: Autor.

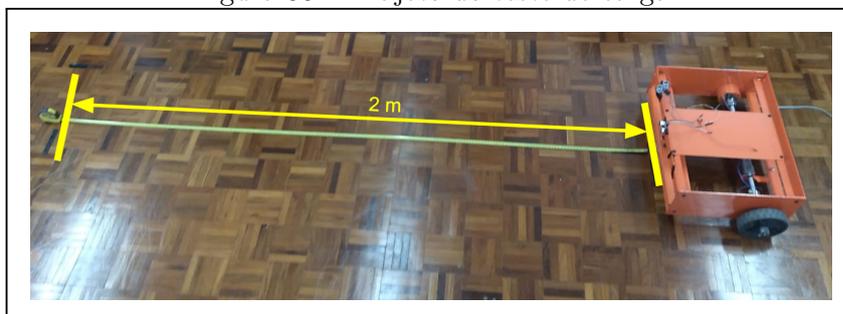
O trajeto programado consiste no trajeto retilíneo de 5 metros, de modo que quando o AMR detecte o objeto a uma distância de 1 metro, decidindo se irá contorná-lo pela direita, ou pela esquerda, analisando o menor deslocamento. De modo que os movimentos para iniciar o contorno fiquem salvos e sejam refeitos inversamente, para posicionar o AMR na reta final da trajetória. Nestes trajetos o AMR desenvolveu com um erro aproximado de 7,5%, de modo a ter um deslocamento linear entre 50 a 70 mm. fora da posição final, bem como terminar o trajeto rotacionado a até 5°.

#### 4.2 TESTES DE CARGA E VELOCIDADE

Afim de verificar o atendimento de transporte da carga especificada neste trabalho, realizou-se cinco testes de circuito retilíneo, o qual o protótipo precisava percorrer uma

distância de dois metros, conforme Figura 38, com diferentes pesos.

Figura 38 – Trajeto do teste de carga



Fonte: Autor.

Realizou-se a trajetória com os pesos de zero, cinco e dez quilos, respectivamente conforme Figura 39, também foi testado a carga máxima que o mecanismo suporta carregar, resultando no valor de 16 Kg.

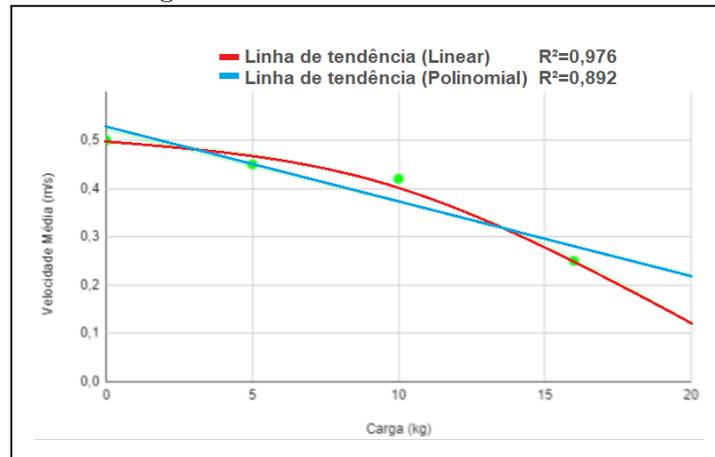
Figura 39 – Dispositivo com carga aplicada



Fonte: Autor.

Afim de visualizar a relação carga por velocidade, pode-se montar um gráfico, bem como verificar a relação deles. Na Figura 40 pode-se ver que a velocidade cai drasticamente próximo a carga máxima testada. se for traçar uma linha de tendência entre os pontos, percebe-se que o coeficiente de determinação de regressão ( $R^2$ ) da linha do tipo polinomial de segundo grau, está mais próximo de um, isto fica implicitamente visual verificando que a linha fica mais próxima de todos os pontos do que da linha de tipo linear, bem como o coeficiente linear está mais longe de um do que o anterior. Validando a ideia de que a perda de velocidade é maior e não linear com maiores cargas, devido principalmente a perdas mecânicas relacionadas ao próprio desempenho do motor, que também não é linear.

Figura 40 – Curva de Velocidade x Peso

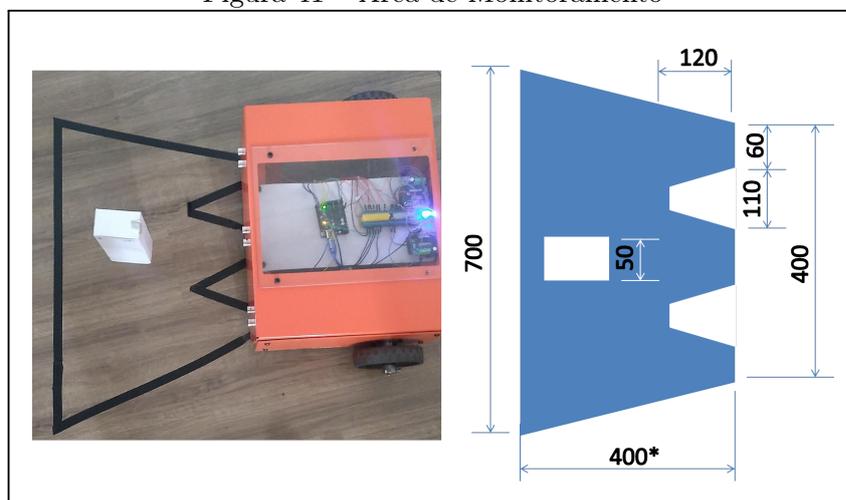


Fonte: Autor.

#### 4.3 TESTES DE SEGURANÇA

Para testar a área de segurança do veículo, primeiramente ergue-se a parte traseira, de modo a travar os movimentos, contudo deixando as rodas livres para giro, posteriormente testou-se pontos com um objeto de 50 mm, simulando a largura de um tornozelo humano, em que interrompesse a rotação das rodas, se detectasse um objeto a sua frente, em uma distância menor que 400 mm. Com estes testes chegou-se ao resultado da Figura 41, o qual consiste na área monitorada, em azul, e se percebe que existe dois pontos cegos, entre os sensores ultrassônicos. Salienta-se que a medida de 400 mm é configurável, de acordo com a programação, passível de chegar até um valor de 4000 mm. Adotou-se a medida de 400 mm em função da velocidade média do veículo, bem como a média de alguns produtos já existentes no mercado.

Figura 41 – Area de Monitoramento



Fonte: Autor.

Foi realizado o teste de parada do protótipo, de modo a mensurar a distância necessária para paralisá-lo em relação a um obstáculo fixo. De modo prático, colocou-se o AMR a uma distância linear de 80 cm de uma parede, fazendo-o avançar até que detectasse e parasse completamente seu movimento. Conforme visto na Figura 42.

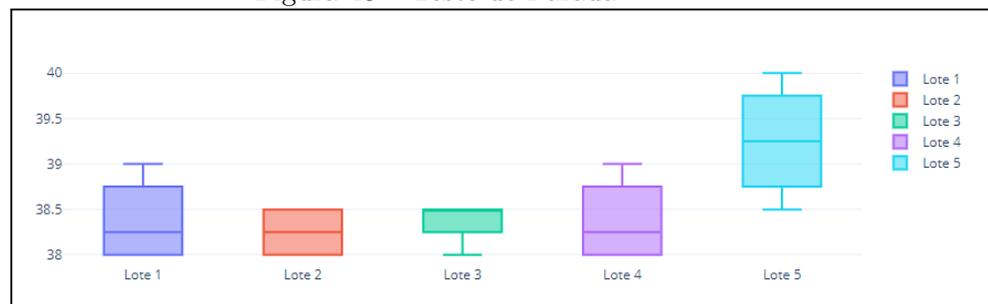
Figura 42 – Teste de Parada



Fonte: Autor.

Apartir dos dados coletados, pode-se montar um gráfico com valores medidos. Conforme visto na Figura 43, foram realizados cinco lotes de 10 testes cada, o gráfico de *boxplot* demonstra os valores mínimos, médios, o primeiro quartil e o terceiro quartil. Percebe-se valores menores que os 40 cm. definidos, pelo fato do tempo e distância que os motores levam para parar totalmente o veículo.

Figura 43 – Teste de Parada



Fonte: Autor.

#### 4.4 ORÇAMENTOS

A fim de contribuir com a análise de viabilidade do projeto, cada componente do veículo foi orçado, e para se ter uma confiabilidade nos custos, realizou-se a média entre 3 orçamentos de um mesmo item, sendo um destes, em loja física, próxima a região onde este

trabalho foi realizado, e os outros dois orçamentos em lojas virtuais, conforme Tabela 8. Não foram consideradas as horas gastas em montagem mecânica, programação e ajustes no projeto. Pode-se visualizar que a parte mecânica corresponde a 28% do valor total do orçamento, bem como os dois motores somam juntos 20% do orçamento, sendo estes dois, os itens de maior custo.

Tabela 8 – Orçamentos Realizados

Recurso	Preço Médio Unitário (R\$)	Quantidade (Un.)
Esp32	54,58	1
Born Esp32	66,56	1
Estrutura Aço	385,70	1
Rolamento 6303ZZ	18,51	4
Driver H BTS7960	62,07	2
Sensor Encoder	26,27	2
Motor Akiyama	135,03	2
Sensor HC-Sr04	15,30	3
Kit Parafusos/Porcas	30,00	1
Materiais Elétricos	107,40	1
Total = R\$ 1430,89		

Fonte: Autor.

## 5 CONCLUSÃO

O trabalho apresenta a implementação de um protótipo de um veículo autônomo, consistindo em uma estrutura metálica de aproximadamente 450 x 450 mm, com capacidade de carregar até 16 Kg, com a topologia de tracionamento central, duas rodas e apoio, possuindo um sistema de guiamento autônomo, que direciona e contorna obstáculos até o veículo alcançar o ponto de destino informado. A importância desse trabalho consiste em fomentar a criação e implementação de produtos nacionais autônomos que atendam a demanda de transporte de cargas, pois atualmente as soluções existentes são de grande parte importadas com altos custos e prazos de entrega.

O conjunto mecânico e o sistema de tração se portaram como esperado suportando a carga mínima de 10 kg definida inicialmente no trabalho. O sistema de guiamento, consistindo de sensores ultrassônicos e sensores odométricos, validam um sistema de posicionamento, com detecção de obstáculos na parte frontal do veículo, sendo possível mensurar os erros dos sensores através da leitura dos valores enviados pela serial, possibilitando assim a correção do trajeto. O protótipo permitiu a aplicação e junção das áreas mecânica, elétrica e programação, gerando um conceito de um veículo autônomo de baixo custo, funcional e autônomo.

### 5.1 TRABALHOS FUTUROS

Como extensão do projeto, é passível implementar um sistema para usuários, que possa definir os pontos de destinos e ter informações em tempo real do veículo, como por exemplo, autonomia das baterias e posição em tempo real.

Um segundo ponto, que pode ser abordado juntamente com o anterior, é analisar (verificando possíveis melhorias) para produzir em massa o projeto, transformando o protótipo em produto.

## REFERÊNCIAS

- AI, C. et al. Agv scheduling system based on mqtt protocol. In: IEEE. **2021 4th International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)**. [S.l.], 2021. p. 1387–1392.
- AKBAR, D. A. **Rancang bangun charger otomatis AGV berbasis NodeMCU ESP 32**. 2021. 7–11 p.
- ARAÚJO, J. R. C. d. et al. **Estudo e implementação de técnicas para identificação e controle de trajetória para um AGV usando Arduino**. [S.l.]: Universidade Federal de Campina Grande, 2017.
- BASTOS, F. F. et al. **Estudo e implementacao de controladores fuzzy e pid para controle de direcao e velocidade de um agv com visao computacional**. [S.l.]: Universidade Federal Rural do Semi-Árido, 2019.
- BERLIN, F.; GRANATH, S. **Obstacle Detection and Avoidance for an Automated Guided Vehicle**. 2021.
- BEZERRA, C. G. **Localização de um robô móvel usando odometria e marcos naturais**. 2004.
- CAPÁK, T. et al. **Optimization of traction unit for low-cost automated guided vehicle**. 2019. 012064 p.
- CARNEIRO, R. d. S.; BANDEIRA, M. S. et al. **Projeto básico de robô seguidor de linha controlado por Arduino**. 2021.
- CUCCHIARA, R.; PERINI, E.; PISTONI, G. **Efficient Stereo Vision for Obstacle Detection and AGV Navigation**. 2007. 291–296 p.
- DAS, S. K.; PASAN, M. **Design and methodology of automated guided vehicle-a review**. 2016. 29–35 p.
- \_\_\_\_\_. \_\_\_\_\_. 2016. 29–35 p.
- FILHO, S. C. T. **Modelagem e otimização de um robô de arquitetura paralela para aplicações industriais**. 2006.
- IFR. **International Federation of Robotics**. 2013. Disponível em: <[www.ifr.org/history](http://www.ifr.org/history)>.
- JUNG, C. R. et al. **Computação embarcada: Projeto e implementação de veículos autônomos inteligentes**. 2005. 1358–1406 p.
- KEYENCE. **SAFETYSCANNER**. 2022. Disponível em: <[https://www.keyence.com.br/landing/safety/pr\\_sz-v\\_safetyscanner.jsp](https://www.keyence.com.br/landing/safety/pr_sz-v_safetyscanner.jsp)>.
- KIM, C.; TANCHOCO, J.; KOO, P.-H. **AGV dispatching based on workload balancing**. [S.l.]: Taylor & Francis, 1999. 4053–4066 p.
- KOTZE, B. **An integrated control system for an Automatic Guided Vehicle (AGV)**. [S.l.]: Central University of Technology, Free State, 2008. 4–12 p.

- KUKA. **Especificação KUKA KR60**. 2021. Disponível em: <[http://www.kuka-robotics.com/res/sps/e6c77545-9030-49b1-93f5-4d17c92173aa:\\_Spez\\_KR\\_60\\_100\\_P\\_en.pdf](http://www.kuka-robotics.com/res/sps/e6c77545-9030-49b1-93f5-4d17c92173aa:_Spez_KR_60_100_P_en.pdf)>.
- LEE, J.; HYUN, C.-H.; PARK, M. **A vision-based automated guided vehicle system with marker recognition for indoor use**. [S.l.]: Multidisciplinary Digital Publishing Institute, 2013. 10052–10073 p.
- LIMA, D. et al. **Localização absoluta de robôs móveis em ambientes industriais**. 2010.
- LOUTFIA, E. et al. **3D model reconstruction from aerial ortho-imagery and LiDAR data**. 2017.
- MAARIF, E.; MOYO, T. **Driving control module for low cost industrial automated guided vehicle**. 2019. 012016 p.
- MAIER, A.; SHARP, A.; VAGAPOV, Y. **Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things**. 2017. 143–148 p.
- MONTIEL, O.; GONZÁLEZ, A.; SEPÚLVEDA, R. **Vision Based Obstacle Detection Module for a Wheeled Mobile Robot**. [S.l.]: IntechOpen, 2010.
- MOREIRA, N. M. da S. **Localização de robôs autônomos em ambiente industrial**. 2014.
- MUNIZ, G. M. et al. **Solução em transporte de suprimentos: desenvolvimento de protótipo de veículo guiado automaticamente**. [S.l.]: Universidade do Estado do Amazonas, 2022.
- PEREIRA, D. d. A. **Localização de AGVs baseado na detecção de marcadores RFIDs, magnéticos e GPS-indoor**. Tese (Doutorado), 2019.
- PEREIRA, M. C.; OLIVEIRA, R. L. de. **AUTOMAÇÃO RESIDENCIAL HÍBRIDA DE BAIXO CUSTO PARA MAIOR ACESSIBILIDADE E CONFORTO DE PESSOAS PORTADORAS DE NECESSIDADES ESPECIAIS**. 2019.
- RAK, F.; WIORA, J. **Comparison of ESP programming platforms**. 2021. 77–86 p.
- RODRIGUES, K. d. O. **Microcontroladores e Webservices**. 2013.
- SAHA, R. et al. **Monitoring air quality of Dhaka using IoT: Effects of COVID-19**. 2021. 715–721 p.
- SANTOS, E. A. da S. **Logística baseada em AGVs**. 2013.
- SHARP, A.; VAGAPOV, Y. **Comparative analysis and practical implementation of the ESP32 microcontroller module for the Internet of Things**. 2017.
- SHINZATO, P. Y. **Sistema de identificação de superfícies navegáveis baseado em visão computacional e redes neurais artificiais**. 2010.

SILVA, L. A. **Solução em transporte de suprimentos: protótipo de veículo guiado automaticamente.** [S.l.]: Universidade Tecnológica Federal do Paraná, 2016.

SOUZA, J. de; ROYER, R. **IMPLANTAÇÃO DE UM SISTEMA AGV–VEÍCULO GUIADO AUTOMATICAMENTE: UM ESTUDO DE CASO DEPLOYING A AGV SYSTEM-AUTOMATICALLY GUIDED VEHICLE: A CASE STUDY.** 2013.

SYUKRIYADIN, S. et al. **Permanent magnet DC motor control by using arduino and motor drive module BTS7960.** 2018. 012023 p.

ZMUDA, M. A.; ELESEV, A.; MORTON, Y. T. **Robot localization using re and inertial sensors.** 2008. 343–348 p.

**APÊNDICE A**

```
//Leitura de distância com o sensor HC-SR04
include <Ultrasonic.h>
Ultrasonic ultrassomC(4,A3); // define o nome do sensor(ultrassom) e onde esta ligado o
trig(4) e o echo(a0) respectivamente
Ultrasonic ultrassomE(7,A2);
Ultrasonic ultrassomD(8,A1);
long distancia;
long distancia1;
long distancia2;
void setup()
{
Serial.begin(9600);
}
void loop()
{
distancia = ultrassomC.Ranging(CM); // ultrassom.Ranging(CM) retorna a distancia em
centímetros(CM) ou polegadas(INC)
distancia1 = ultrassomE.Ranging(CM);
distancia2 = ultrassomD.Ranging(CM);
Serial.print(distancia); //imprime o valor da variável distancia
Serial.print("\n");
Serial.print(distancia1);
Serial.print("\n");
Serial.print(distancia2);
Serial.println("cm");
delay(200);
}
```

## APÊNDICE B

```

    include <Ultrasonic.h>
include <stdio.h>
include <stdlib.h>
Ultrasonic ultrassomC(4,A3); // define o nome do sensor(ultrassom) e onde esta ligado o
trig(4) e o echo(a0) respectivamente
Ultrasonic ultrassomE(7,A2);
Ultrasonic ultrassomD(8,A1);
int n,aux,aux1,aux2;
int distancia[]=0,0, 0, 0,0;
int distancia1[]=0,0, 0, 0,0;
int distancia2[]=0,0, 0, 0,0;
int compfun(const void*a,const void*b) {
return ( *(int*)a - *(int*)b); }
void setup()
{
Serial.begin(9600);
}
void loop()
hcsr04();
delay(1000);
void hcsr04()
for( n = 0; n < 5; n++)
distancia[n] = ultrassomE.Ranging(CM);// retorna a distancia em centímetros(CM)
distancia1[n] = ultrassomC.Ranging(CM);
distancia2[n] = ultrassomD.Ranging(CM);
Serial.print("\n");
qsort(distancia, 5, sizeof(int),compfun);
qsort(distancia1, 5, sizeof(int),compfun);
qsort(distancia2, 5, sizeof(int),compfun);
Serial.println("\n");
aux = distancia[2];
aux1= distancia[2];
aux2 = distancia[2];
for( n = 0; n < 5; n++)
Serial.print(distancia1[n]);
Serial.print("\n");
Serial.print(distancia1[2]);

```

## APÊNDICE C

```
Programa: Sensor de velocidade Arduino
int rpmE=0; //quantidade de pulsos esquerdo
int rpmD=0; //quantidade de pulsos direito
int dif=0; //diferença da leitura dos encoders
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  attachInterrupt(0, contador, FALLING);
  attachInterrupt(1, contador1, FALLING);
  delay(1000);
  printarpm();
}
void contador()
{
  //Incrementa Contador Esquerdo
  rpmE++;
}
void contador1()
{
  //Incrementa Contador Direito
  rpmD++;
}
void printarpm()
{
  Serial.print(rpmE);
  Serial.print("\n");
  Serial.print(rpmD);
  Serial.print("pulsos");
  Serial.println("\n");
  Serial.println(ajuste);
}
```