



**UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA – EST**

GLAUCIO MARCULINO MUNIZ

**SOLUÇÃO EM TRANSPORTE DE SUPRIMENTOS:
DESENVOLVIMENTO DE PROTÓTIPO DE VEÍCULO GUIADO
AUTOMATICAMENTE.**

Manaus

2022

GLAUCIO MARCULINO MUNIZ

**SOLUÇÃO EM TRANSPORTE DE SUPRIMENTOS:
DESENVOLVIMENTO DE PROTÓTIPO DE VEÍCULO GUIADO
AUTOMATICAMENTE.**

Projeto de Pesquisa desenvolvido durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Eletrônica da Escola Superior de Tecnologia na Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Engenheiro em Eletrônica.



Orientador: Angilberto Muniz Ferreira Sobrinho, Dr.

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia – EST

Reitor:

Cleinaldo de Almeida Costa.

Vice-Reitor:

Cleto Cavalcante de Souza Leal.

Diretora da Escola Superior de Tecnologia:

Ingrid Sammyne Gadelha Figueiredo, Me.

Coordenador do Curso de Engenharia em Eletrônica:

Bruno da Gama Monteiro, Me.

Banca Avaliadora composta por:

Data da defesa: 25/01/2022.

Prof. Angilberto Muniz Ferreira Sobrinho, Dr. (Orientador)

Prof. Bruno da Gama Monteiro, Me. (Avaliador 1)

Prof. Jozias Parente de Oliveira, Dr. (Avaliador 2)

CIP – Catalogação na Publicação

MUNIZ, Glaucio Marculino

Solução em transporte de suprimentos: desenvolvimento de protótipo de veículo guiado automaticamente / Glaucio Marculino Muniz; [orientado por] Angilberto Muniz Ferreira Sobrinho. - Manaus: 2022.

41 p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia Eletrônica). Universidade do Estado do Amazonas, 2022.

1. *Automated Guided Vehicle (AGV)*. 2. Arduino. 3. Transporte de Suprimentos. I. Sobrinho, Angilberto Muniz Ferreira.

GLAUCIO MARCULINO MUNIZ

**SOLUÇÃO EM TRANSPORTE DE SUPRIMENTOS:
DESENVOLVIMENTO DE PROTÓTIPO DE VEÍCULO GUIADO
AUTOMATICAMENTE.**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Eletrônica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Engenheiro em Eletrônica.

Nota obtida: 9,4 (Nove vírgula quatro)

Aprovada em 25 / 01 / 2022.

Área de concentração: Robótica, Sistemas Embarcados

BANCA EXAMINADORA


Orientador: Angilberto Muniz Ferreira Sobrinho, Dr.


Avaliador: Bruno da Gama Monteiro, Me.


Avaliador: Jozias Parente de Oliveira, Dr.

Manaus

2022

AGRADECIMENTOS

Primeiramente a Deus, por me dar força e não desistir e persistir, sempre acreditando no melhor, e também “*as mulheres da minha vida*”, minha esposa Mayara Muniz e minha filha Melissa Muniz as quais sempre foram minha maior motivação e também não posso esquecer de agradecer a Universidade Estadual do Amazonas – UEA e todo seu corpo docente, que me ajudaram como um centro de informações disponíveis ao nosso alcance, em vários momentos tive a oportunidade de aprender com professores inspiradores.

“Pois está-se à sombra da sabedoria como se está à sombra do dinheiro: a utilidade do saber consiste em que a sabedoria dá vida ao que a possui”.

Eclesiastes 7:12

RESUMO

Este trabalho apresenta o desenvolvimento de um protótipo de veículo guiado automaticamente, do inglês, “*Automated Guided Vehicle*” (AGV), usado para fins de transporte de suprimentos em ambiente produtivo. Neste documento é descrita a construção e programação de componentes servomecanismos controlados por um microcontrolador ATmega328PA da placa Arduino Uno. Também explica a utilização de técnica de navegação para guiar o protótipo AGV através de uma linha preta em contraste com um fundo branco, para alinhar o protótipo a pista foram utilizados dois sensores IR instalados na sua dianteira, na posição próxima ao limite lateral da linha preta, para que quando o sensor detectar um sinal de retorno refletido na cor preta, entenda que deve enviar um comando aos motores para manter o robô no curso estendido. Foram feitas várias simulações e testes para otimizar o código de programação e atingir a melhor performance na tarefa de transporte de suprimentos. O protótipo se utiliza de energia de baterias alcalinas para energizar todo sistema eletrônico e motores, dando autonomia ao AGV.

Palavras-chaves: sistema embarcado, AGV, transporte de suprimentos, robô seguidor de linha, Arduino Uno.

ABSTRACT

This work presents the development of a prototype of an automatically guided vehicle, “Automated Guided Vehicle” (AGV), used for the purpose of transporting supplies in a productive environment. This document describes the construction and programming of servomechanism components controlled by an ATmega328PA microcontroller on the Arduino Uno board. It also explains the use of navigation technique to guide the AGV prototype through a black line in contrast to a white background. , so that when the sensor detects a return signal reflected in black, it understands that it must send a command to the motors to keep the robot on the extended course. Several simulations and tests were carried out to optimize the programming code and achieve the best performance in the task of transporting supplies. The prototype uses energy from alkaline batteries to power the entire electronic system and motors, giving the AGV autonomy.

Keywords: embedded system, AGV, supply transport, line follower robot, Arduino Uno.

LISTA DE FIGURAS

Figura 1 - Arduino modelo RobotDyn Uno.....	10
Figura 2 - Diagrama em blocos.....	10
Figura 3 - Estrutura de motor de corrente contínua.....	13
Figura 4 - Motor DG01D-A130 com caixa de redução e eixo duplo.....	14
Figura 5 – Esquema da ponte H	15
Figura 6 – Funcionamento da ponte H.....	15
Figura 7 – Ponte H L298N.....	17
Figura 8 – Ponte H CI L298N	18
Figura 9 - Pinos de conexão do CI LN298N.....	18
Figura 10 – Símbolo do fotodiodo.....	20
Figura 11 – Circuito do Sensor.....	20
Figura 12 – Verificação da presença de linha.....	21
Figura 13 – Disposição dos sensores.....	22
Figura 14 – Sinal ultrassônico.....	23
Figura 15 – Ondas dos parâmetros de controle.....	23
Figura 16 – PWM Variando a tensão média.....	24
Figura 17 – Definição de ciclo ativo.....	25
Figura 18 – Gráfico PWM com comandos do Arduino.....	25
Figura 19 – Diagrama em blocos das etapas do projeto (Descrição).....	26
Figura 20 – Diagrama em blocos das etapas do projeto (Ilustração).....	26
Figura 21 – Descrição do kit para montagem do AGV.....	27
Figura 22 – “Roda boba”	27
Figura 23 – Montagem dos motores DC no chassi.....	28
Figura 24 – Montagem do suporte de pilhas e ponte H.....	28
Figura 25 – Placa Arduino instalada no chassi.....	29
Figura 26 – Sensores infravermelho e sensor ultrassônico.....	29
Figura 27 – Veículo guiado automaticamente AGV montado.....	30
Figura 28 – Diagrama de ligações dos fios do protótipo.....	31
Figura 29 – Tabela de pinagem.....	31
Figura 30 – Funcionamento dos dois sensores de infravermelho.....	32
Figura 31 – Código fonte para controle dos motores vs sensores.....	33
Figura 32 – Circuito oval para simulação.....	34
Figura 33 – Resultado dos “loops”	35

SUMÁRIO

INTRODUÇÃO	8
1 REFERENCIAL TEÓRICO	10
1.1 MICROCONTROLADOR	10
1.1.1 ATmega328	11
1.2 MOTOR DE CORRENTE CONTÍNUA	12
1.2.1 Caixa de Redução	12
1.3 PONTE H	15
1.3.1 Circuito Integrado L298N	16
1.4 SENSOR	19
1.4.1 Sensores infravermelhos	19
1.4.2 Funcionamento dos sensores infravermelhos	21
1.4.3 Sensores ultrassônicos	22
1.4.4 Funcionamento do sensor ultrassônico	23
1.5 PWM	24
1.5.1 Comandos do Arduino para PWM.	25
2 MATERIAIS E METODOS	26
2.1 MONTAGEM	26
2.2 DIAGRAMA DAS FIAÇÕES	31
3 IMPLEMENTAÇÃO DO PROJETO	32
3.1 LÓGICA DE PROGRAMAÇÃO	32
3.2 SIMULAÇÕES	34
4 RESULTADOS OBTIDOS	35
CONCLUSÃO	36
APÊNDICE A – CÓDIGO DO MICROCONTROLADOR ATMEGA328	38

INTRODUÇÃO

O tema deste trabalho é um protótipo de um veículo guiado automaticamente ou do inglês, *Automated Guided Vehicle* (AGV). AGV é um tipo de robô móvel industrial utilizado para manipular equipamentos por meio de guindastes, elevadores, transportadores, etc., sendo um veículo autônomo capaz de se mover ao longo de caminhos pré-determinados. É utilizado na transferência de cargas em ambientes fabris. Esse sistema possui diversas vantagens, como por exemplo, proteção aos materiais transportados, segurança aos trabalhadores, aumento da produtividade, melhora das condições de trabalho (YAGHOUBI, 2012).

Em 1953, o primeiro veículo guiado automaticamente foi criado e implementado e no final da década de 50, muitos AGVs já estavam operando. A indústria que mais utiliza os AGVs é a indústria de automóveis, além das indústrias de distribuição, papel, têxtil e metal. Os três principais tipos de veículos utilizados atualmente são: tratores, que movem cargas passivas colocadas em plataformas com rodas; carga unitária, que transportam uma única carga colocada sobre o veículo; e bifurcada, que busca, transporta e entrega a carga unitária (YAGHOUBI, 2012).

O presente trabalho explora a oportunidade de automatização no setor logístico em atividades de transporte de suprimentos em ambiente fabril, que ainda são realizados por pessoas e que não tem a mesma agilidade, autonomia e produtividade de uma máquina, utilizando a tecnologia de robótica como ferramenta capaz de mitigar o tempo de transporte. Nesta pesquisa, foi testada a hipótese de que é possível com a implementação de um sistema de transporte de cargas automático, mitigar o tempo de chegada dos suprimentos em uma linha de produção trazendo maior agilidade, autonomia e maior produtividade ao sistema logístico, utilizando um veículo guiado automaticamente (AGV), como alternativa viável de menor custo, capaz de auxiliar as pessoas no transporte logístico em ambiente produtivo, assim aumentando a performance no seu fluxo de trabalho. Além de erradicar os acidentes que podem ser causados por erros humanos durante o percurso. Este AGV será capaz de seguir uma linha demarcada no piso e realizar o transporte de forma autônoma, podendo levar suprimentos a qualquer lugar programado.

Para teste e validação da hipótese acima, foram implementados os seguintes objetivos. Primeiro fez-se um estudo do material necessário para montagem do robô AGV. Em seguida realizou-se a programação, o ajuste de performance dos motores visando a otimização do tempo de transporte de suprimento em questão e a instalação de sensores ultrassônicos para detecção de obstáculos. Após isso foi feita a análise dos resultados obtidos avaliando o parâmetro de tempo e transporte seguro em todo percurso.

Esta pesquisa se justifica, pelo fato que a implementação de um sistema de transporte de suprimentos automatizado, baseado no uso de veículos guiados automaticamente, viabiliza uma otimização nas atividades diárias dos funcionários, maior sincronismo no abastecimento das células de produção, maior velocidade no transporte interno de cargas, e mais segurança, com efetivo incremento de produtividade para indústria e competitividade no mercado.

Este Trabalho está estruturado em capítulos. No capítulo 1 é apresentada uma revisão bibliográfica sobre os principais conceitos e tecnologias que serviram de base para o desenvolvimento do projeto. Neste capítulo são apresentados breves conceitos começando pela placa de controle usada, o microcontrolador ATmega328 usado pela plataforma Arduino Uno. Em seguida vem os componentes usados para construção do protótipo, principalmente os motores e seu controle de velocidade com o uso da tecnologia PWM e também a disposição dos sensores infravermelhos para melhor orientação do AGV sobre a linha preta no piso.

No capítulo 2 são apresentados os materiais e métodos aplicados no projeto, onde são descritas as etapas executadas durante o desenvolvimento do todo sistema, mostrando a relação dos itens necessários para montagem, mostrando os cuidados e observações necessárias para atender os requisitos do projeto, principalmente em sua parte física, a parte de hardware em si.

No capítulo 3 são apresentados no detalhe a execução da implementação do projeto baseado nas etapas do capítulo anterior, traz principalmente a parte lógica da programação, que deve ser bem consolidada para o entendimento das simulações e testes, também mostrado neste capítulo, esta parte do trabalho julgo como a mais importante, pois nesta fase a qual temos acesso aos parâmetros envolvidos na tarefa, no caso é o transporte de suprimentos, podendo fazer diversos experimentos e várias condições, simulando um ambiente fabril, para serem analisados posteriormente.

No capítulo 4 são apresentados os resultados obtidos, mostram-se os resultados dos experimentos realizados na implementação, e faz-se análises baseadas no referencial teórico e no conhecimento adquirido durante todo o processo de pesquisa e implementação do projeto.

E por fim é apresentada a conclusão que aborda a análise do comportamento do dispositivo AGV para a tarefa proposta, sendo de transportar suprimentos em ambiente fabril, automatizando o transporte de cargas internas e dando maior agilidade e produtividade.

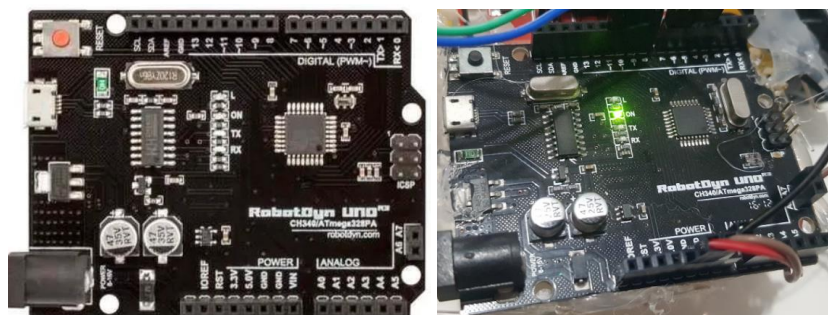
1 REFERENCIAL TEÓRICO

Neste capítulo são apresentados os referenciais utilizados para conceituação, classificação e desenvolvimento aplicados ao projeto.

1.1 MICROCONTROLADOR

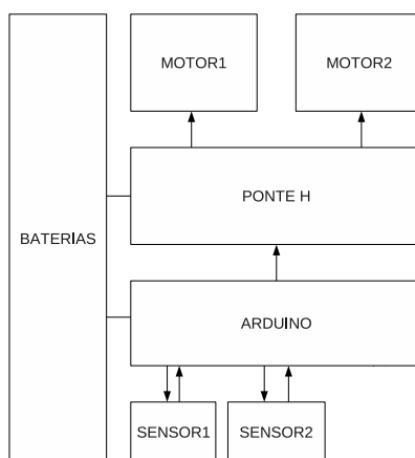
Optou-se por escolher o Arduino modelo RobotDyn Uno, figura 1, com microcontrolador para o projeto. Este dispositivo será responsável pela ligação entre o sensor e o motor. As informações geradas pelos sensores são analisadas pelo microcontrolador e então são passados comandos através da ponte H para o motor executar o movimento apropriado para se manter sobre a linha. O diagrama de blocos dos componentes está apresentado na figura 2.

Figura 1 – Arduino modelo RobotDyn Uno



Fonte: Próprio autor

Figura 2 – Diagrama em blocos



Fonte: Próprio autor

O Arduino é uma plataforma de prototipagem eletrônica *open-source*, baseado em um microcontrolador que se conecta ao computador pessoal através de uma porta serial ou USB, dependendo do modelo utilizado. Ele possui uma linguagem de programação própria, baseada em Wiring, que é implementada em um ambiente de desenvolvimento (IDE), também próprio, baseado em Processing, e que pode ser utilizada em vários sistemas operacionais. Através desta plataforma pode-se montar uma variedade de circuitos de forma fácil e ágil, como, por exemplo, um sistema de sensores, cujos dados podem ser interpretados e utilizados pelo próprio Arduino ou repassados para um PC (*personal computer*, ou computador pessoal). A sua alimentação se dá pela porta USB conectada ao PC ou por uma fonte externa de até 25V (INTERFACING. .,2009).

A linguagem de programação do Arduino é simples e sua sintaxe se assemelha muito ao C++ e ao Java. Um programa típico possui duas funções básicas, o “void setup()”, que é executada logo no início do programa, e o “void loop()”, que é a função executada repetidamente pelo microcontrolador. Fora elas existem diversas outras funções como por exemplo as que fazem com que um pino emita e receba sinais digitais ou analógicos.

1.1.1 ATmega328

A partir de 2013 o ATmega328 é comumente usado em muitos projetos e sistemas autônomos, onde se precisa de um microcontrolador simples, de baixa potência e baixo custo. Talvez a utilização mais comum deste chip seja no popular Arduino, plataforma de desenvolvimento, nomeadamente nos modelos Arduino Uno e o Arduino Nano (WIKIPEDIA, 2018).

O modelo do Arduino escolhido para ser utilizado neste projeto foi o RoboDyn Uno que utiliza um microcontrolador ATmega328PA com interface USB e também um conector de alimentação externa.

Uma série de vantagens fazem do Arduino a escolha para ser utilizada neste projeto, dentre eles estão seu preço atrativo, programação descomplicada e fácil manuseio.

1.2 MOTOR DE CORRENTE CONTÍNUA

Motores elétricos convertem energia elétrica em mecânica. Existem motores de corrente contínua (CC) e de corrente alternada (CA), cada um com diversas variações. Motores de corrente alternada são geralmente usados para máquinas grandes e recebem energia diretamente da rede de distribuição de energia. Segundo (JONES; FLYNN; SEIGER, 1999), robôs móveis usam tipicamente corrente contínua, pois sua fonte de energia é uma bateria.

1.2.1 Caixa de Redução

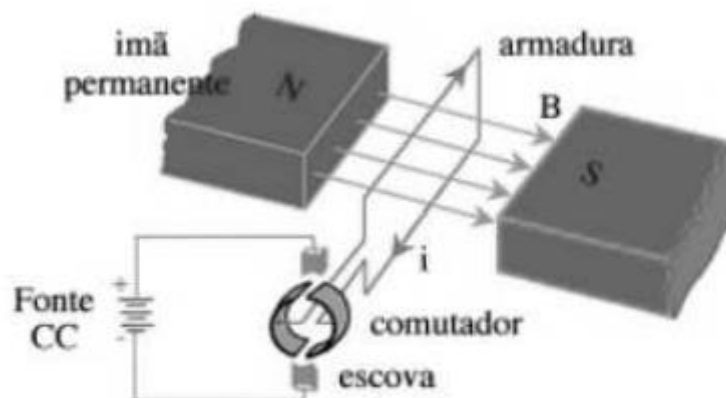
A caixa de redução de velocidade é composta de engrenagens que permitem reduzir a velocidade do movimento de rotação dos motores de uma máquina, transferindo-a para os eixos de tração do equipamento. Para tal, a caixa de redução de velocidade age de acordo com a relação entre o diâmetro do eixo do motor e o diâmetro da roda maior, que viabiliza o movimento.

Para as necessidades de locomoção de um robô, o motor gira em uma velocidade muito alta e com um torque muito baixo. Para inverter essa relação, o motor deve ser ligado a uma caixa de redução que produz uma nova saída que gira mais devagar, porém com um torque maior. Muitos motores CC são vendidos com a caixa de redução já instalada, como é o caso do adquirido para este projeto.

Os motores de corrente contínua têm pelo menos dois terminais que ao se aplicar uma tensão nestes terminais o motor gira em um sentido e invertendo a polaridade da entrada o motor gira no sentido contrário.

Alguns motores CC, como motores de passo, têm mais que dois terminais. Os sinais aplicados nestes terminais energizam partes do motor organizadamente de modo a controlar o ciclo interno do motor. Este controle permite saber a posição do motor em cada instante e obter uma precisão maior da velocidade de rotação do motor. Neste projeto a trajetória do robô pode ser constantemente corrigida a partir da linha, então não será necessário o uso desse tipo de motor.

Figura 3 – Estrutura de motor de corrente contínua



Fonte: (ANTUNES, 2009)

A figura 3 mostra uma simplificação de um motor CC com ímã permanente. Quando um fio mergulhado em um campo magnético é atravessado por uma corrente, surge uma força. O sentido e a intensidade dessa força podem ser calculados pela equação 1.

$$\vec{F} = \vec{iL} \times \vec{B} \quad (1)$$

Duas vezes por ciclo os comutadores mudam de contato, invertendo o sentido da corrente na armadura. Esta mudança do sentido da corrente é essencial para que o torque sobre o comutador esteja atuando sempre no mesmo sentido. A corrente de partida de motores CC é muito grande, as vezes até maior que o valor em regime permanente. Isto ocorre pois quando o motor é ligado, a armadura está parada e, portanto, a força contra eletromotriz E é nula. Logo, toda a tensão sobre o motor é aplicada na resistência da armadura, que é pequena, resultando em uma corrente alta. Isto pode ser melhor compreendido através da equação 2.

$$V = RI + E \quad (2)$$

Por outro lado, essa mesma equação nos mostra que quando a tensão no motor é cessada bruscamente, a energia armazenada na armadura transforma-se em uma corrente reversa. Para prevenir que esta corrente danifique o circuito são colocados diodos em paralelo com os transistores. Os motores comprados para o robô são do modelo DG01D-A130 com caixa de redução e eixo duplo representado na figura 4.

Especificações:

- Tensão de Operação: 3-6VDC
- Torque: 0,35 Kgf/cm (3V) e 0,80 Kgf/cm (6V)
- Sistema magnético com anti-interferência
- Engrenagem com eixo duplo
- Redução: 1:48
- Corrente sem carga: $\leq 200\text{mA}$ @ 6V, $\leq 150\text{mA}$ @ 3V
- Velocidade sem carga: $200 \pm 10\% \text{RPM}$ @ 6V, $90 \pm 10\% \text{RPM}$ @ 3V
- Dimensões: 70 x 37 x 23mm

Figura 4 – Motor DG01D-A130 com caixa de redução e eixo duplo

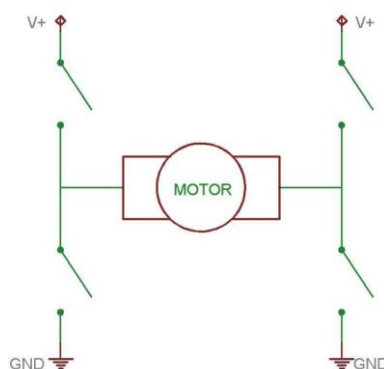


Fonte: (FILIPEFLOP, 2022)

1.3 PONTE H

Dentro do projeto, os motores CC 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 5, 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.

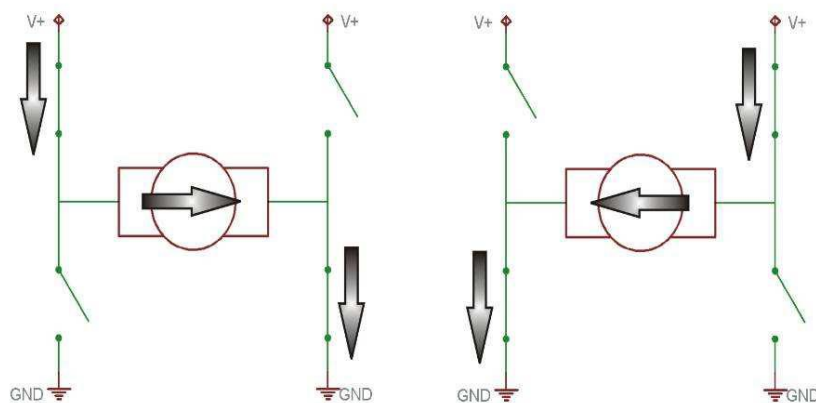
Figura 5 – Esquema da ponte H



Fonte: (PATSKO, 2006)

O fornecimento de corrente ocorre pelo fechamento das chaves duas a duas, de tal forma que a fonte e o terra estejam em posições alternadas dos pólos do motor como indicado na Figura 6.

Figura 6 – Funcionamento da ponte H



Fonte: (PATSKO, 2006)

Caso seja necessário parar o motor CC, a ponte H propicia duas formas distintas para fazê-lo. A primeira consiste em abrir as quatro chaves da ponte, fazendo com que a corrente gerada pela fonte não circule pelo motor, então o movimento nele é apenas inercial e cessa suavemente devido apenas pela força do atrito. A segunda maneira é feita fechando as duas chaves superiores, que estão ligadas a fonte, ou fechando as duas chaves inferiores, ligadas ao terra. Desta forma produz-se um “freio eletrônico”, parando o motor quase instantaneamente. Isso ocorre, pois, as chaves fazem com que o motor entre num “curto-circuito” e seja freado justamente pela corrente inversa gerada pela natureza indutiva do próprio motor, que o obriga a girar em sentido contrário.

Um caso especial que se deve tomar cuidado é de não fechar as quatro chaves ao mesmo tempo ou duas chaves de um mesmo lado, pois isso faz com que “o fluxo da corrente vá direto do polo positivo para o negativo, causando um curto-circuito fatal para a fonte de alimentação e para os componentes eletrônicos envolvidos no circuito” (PATSKO, 2006).

1.3.1 Circuito Integrado L298N

No caso deste projeto foi utilizado o circuito integrado L298N (STMICROELECTRONICS, 2000).

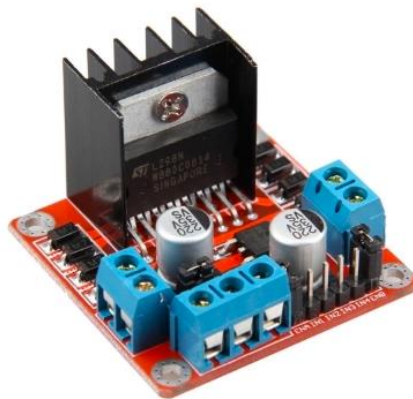
Especificações:

- Tensão de Operação: 4~35v
- Chip: ST L298N
- Controle de 2 motores DC ou 1 motor de passo
- Corrente de Operação máxima: 2A por canal ou 4A máx.
- Tensão lógica: 5v
- Corrente lógica: 0~36mA
- Limites de Temperatura: -20 a +135°C
- Potência Máxima: 25W
- Dimensões: 43 x 43 x 27mm
- Peso: 30g

Na ponte H com componentes discretos e circuito integrado L298N, figura 7, foram utilizados transistores MOSFETs como chaves que para fechá-las basta colocar uma tensão na base deles. Os resistores são adicionados para evitar que haja sobrecarga da corrente e os diodos

para evitar que uma possível corrente reversa gerada ao acionar o freio dos motores possa danificar o circuito.

Figura 7 – Ponte H L298N

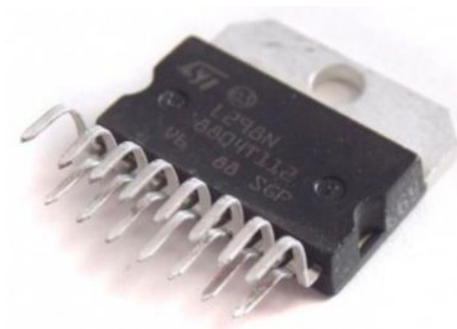


Fonte: (FILIPEFLOP, 2022)

No L298N, figura 9, são implementadas duas pontes H com a existência do “sense” que deve ser conectada a um resistor para o terra com o intuito de controlar a corrente de carga de cada ponte; a existência de quatro pinos “out”, duas para cada ponte, que é por onde sai a corrente controlada; a existência de um pino “Vs” para a entrada da tensão das duas pontes; “VSS” para a alimentação dos blocos lógicos; “GND” para o terra; dois “Enable”, um para cada ponte, que funciona justamente para ativar as pontes; e quatro pinos “Input”, dois para cada ponte, que determinam por qual “out” sairá a tensão fornecida em “Vs”. No datasheet consta ainda quatro estados para a ponte H: no caso do Input1 estar recebendo sinal e do Input2 não, o motor funciona para “frente”, no caso contrário o motor funciona para “trás”. Caso os dois Input estejam iguais, ou seja, ambos recebendo sinal ou ambos não recebendo sinal, temos o freio do motor, e caso enable seja desligado o motor roda apenas com a inércia.

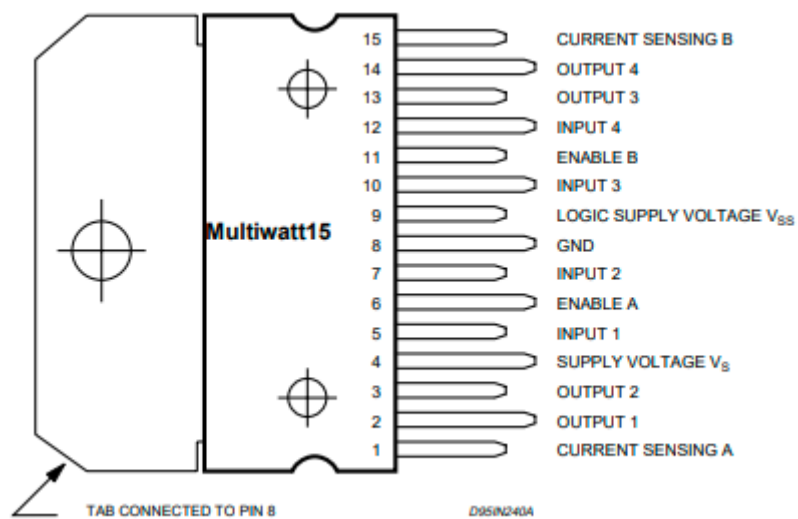
O componente centro integrado L298N tem seu encapsulamento, figura 8, e já possui duas pontes H integradas, ideal para o caso, pois foi necessário controlar dois motores CC de forma independente através do microcontrolador. Além disso outra vantagem é o seu dimensionamento, pois ocupa um espaço muito menor do que duas pontes H feitas com somente com componentes discretos.

Figura 8 – Ponte H CI LN298N



Fonte: (STMICROELETRONICS, 2000)

Figura 9 – Pinos de conexão do CI LN298N



Fonte: (STMICROELETRONICS, 2000, p. 2)

1.4 SENSOR

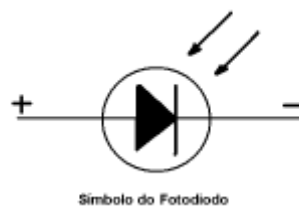
Foram utilizados dois tipos de sensores para o projeto, dois sensores infravermelho para serem instalados na dianteira do chassi e detectar os sinais vindo da linha preta de acordo com o posicionamento do AGV e um sensor ultrassônico para detecção de obstáculos. O movimento do AGV será interrompido quando da detecção de algum obstáculo e só voltará a operar com a intervenção humana, evitando acidentes.

1.4.1 Sensores infravermelhos

Os sensores utilizados no projeto são fototransistores e fotodiodos, funcionando como receptores e emissores respectivamente. Basicamente eles verificam se o protótipo está ou não em cima do seu trajeto e se ele precisa fazer alguma correção na rota.

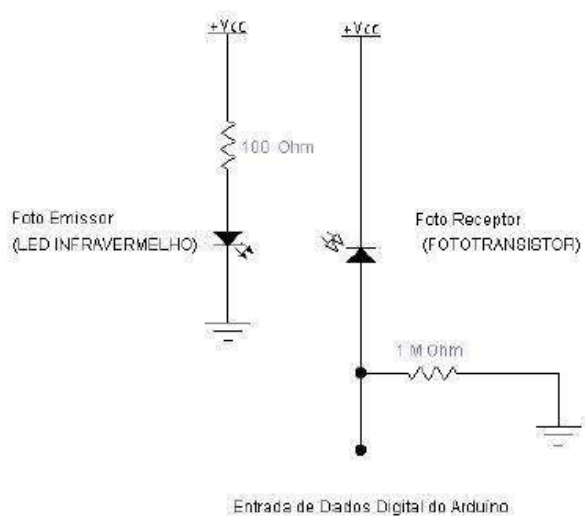
Todo o funcionamento desses sensores está baseado numa propriedade fundamental dos fotodiodos, cujo símbolo está representado na figura 10. Esses componentes eletrônicos quando instalados em polaridade reversa possuem uma resistência elétrica praticamente infinita assim como qualquer diodo em polaridade reversa. No entanto quando um feixe de luz infravermelha é direcionado para o fotodiodo, sua resistência elétrica cai drasticamente e é através dessa propriedade que é possível verificar se o robô móvel está ou não sobre a faixa de cor escura utilizada para orientar o percurso que deverá ser percorrido. O funcionamento do fototransistor é basicamente o mesmo do fotodiodo, entretanto ele é mais sensível ao sinal luminoso pois basta uma pequena corrente na base para que ele altere seu estado verificando então a presença da luz. O diodo emissor (LED) emite um feixe luminoso que reflete sobre uma superfície de cor clara e incide no fototransistor, "receptor", diminuindo sua resistência e permitindo a passagem de corrente que é verificada pelo Arduino. Como a faixa é de cor escura, o feixe luminoso emitido pelo LED é absorvido fazendo com que não haja passagem de corrente pelo fotodiodo. Com isso é verificado se o "carro" está ou não sobre o percurso indicado. Como escolhemos criar um percurso de cor escura o Arduino interpreta a "ausência de luz" como caminho e a presença, no caso a reflexão da luz emitida, como sendo o fundo fora do caminho. A figura 11 mostra o esquema do circuito individual dos sensores:

Figura 10 – Símbolo do fotodiodo



Fonte: (STMICROELETRONICS, 2000)

Figura 11 – Circuito do Sensor

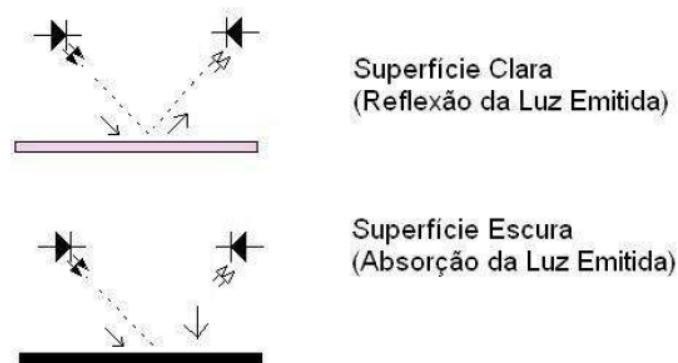


Fonte: Autoria própria

1.4.2 Funcionamento dos sensores infravermelhos

A figura 12 mostra como funciona a verificação da presença ou não do percurso e a disposição do par de sensores.

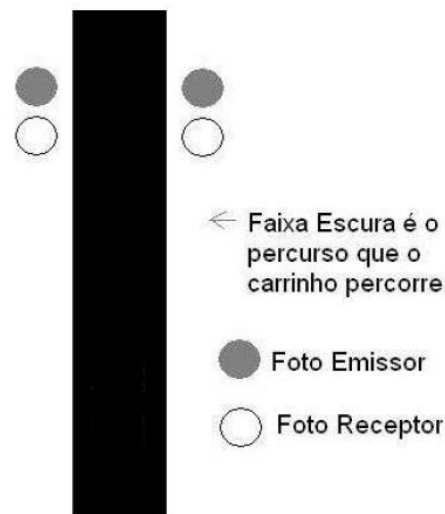
Figura 12 – Verificação da presença de linha



Fonte: Autoria própria

Os sensores estão dispostos em pares (receptor e emissor) próximos um do outro que são distribuídos de tal forma que um fica na faixa central do robô e os outros dois ficam adiantados e mais para as laterais, com uma distância maior que o da faixa preta utilizada como caminho. Inicialmente, foi previsto que seria necessário haver um isolamento lateral entre o emissor e o receptor para que não houvesse interferência na captação do sinal, entretanto após feitos alguns testes verificaram-se que tal isolamento era desnecessário pois não gerava tal interferência, pelo contrário, ele atrapalhava na reflexão da luz. Eles estão levemente inclinados na diagonal para haver uma reflexão mais precisa e dispostos a uma altura aproximadamente dois centímetros do solo. Quanto à quantidade de sensores são dois pares (receptor e emissor) dispostos de maneira semelhante à figura 13:

Figura 13 – Disposição dos sensores



Fonte: Autoria própria

Tal disposição foi escolhida para que os sensores funcionem da seguinte maneira: quando os dois sensores estão sem receber sinal caracteriza “estado presente” o carro está na linha, quando um dos dois sensores da frente fazem a verificação e recebe o sinal refletido da cor preta, define em qual direção o carro deverá seguir caso haja uma mudança de trajetória (uma curva), funcionando como verificador de “estado futuro” do protótipo.

1.4.3 Sensores ultrassônicos

O princípio de funcionamento deste sensor consiste na emissão de sinais ultrassônicos pelo sensor e na leitura do sinal de retorno (reflexo/eco) desse mesmo sinal. A distância entre o sensor e o objeto que refletiu o sinal é calculada com base no tempo entre o envio e leitura de retorno. Sinais Ultrassônicos são ondas mecânicas com frequência acima de 40 KHz.

Como ouvido humano só consegue identificar ondas mecânicas até a frequência de 20KHz, os sinais emitidos pelo sensor Ultrassônico não podem ser ouvidos ou percebidos por nós.

Este sensor é composto de três partes principais:

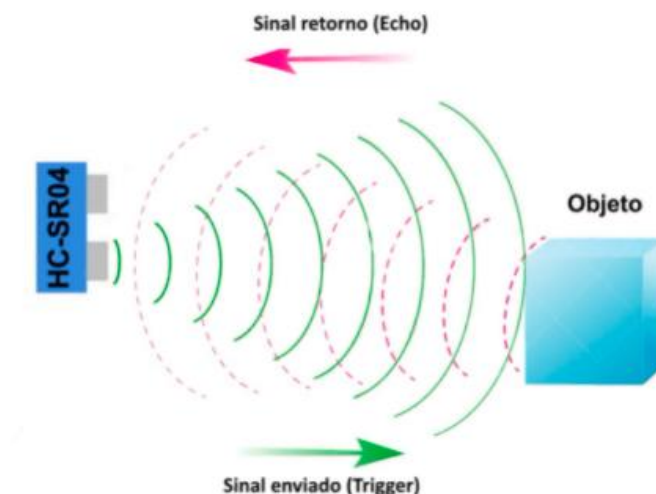
- Transmissor Ultrassônico – Emite as ondas ultrassônicas que serão refletidas pelos obstáculos;
- Um receptor – Identifica o eco do sinal emitido pelo transmissor;

- Circuito de controle – Controla o conjunto transmissor/receptor, calcula o tempo entre a emissão e recepção do sinal;

1.4.4 Funcionamento do sensor ultrassônico

O funcionamento se baseia no envio de sinais ultrassônicos pelo sensor, que aguarda o retorno eco (“echo”) do sinal, e com base no tempo entre envio e retorno, calcula a distância entre o sensor e o objeto detectado, como podemos ver na figura 14.

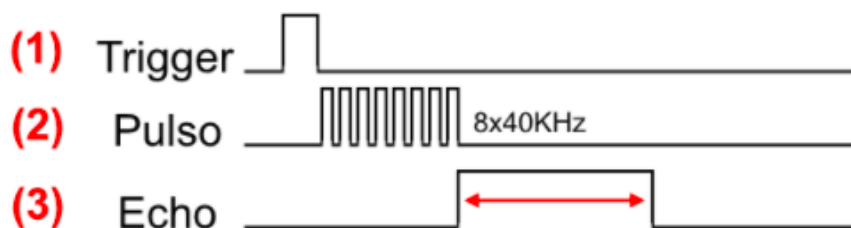
Figura 14 – Sinal ultrassônico



Fonte: (FILIPEFLOP, 2022)

Primeiramente é enviado um pulso de 10 μ s, indicando o início da transmissão de dados. Depois disso, são enviados 8 pulsos de 40 KHz e o sensor então aguarda o retorno (em nível alto/high), para determinar a distância entre o sensor e o objeto, utilizando a equação $\text{Distância} = (\text{Tempo echo em nível alto} * \text{velocidade do som}) / 2$, como é mostrado na figura 15.

Figura 15 – Ondas dos parâmetros de controle



Fonte: (FILIPEFLOP, 2022)

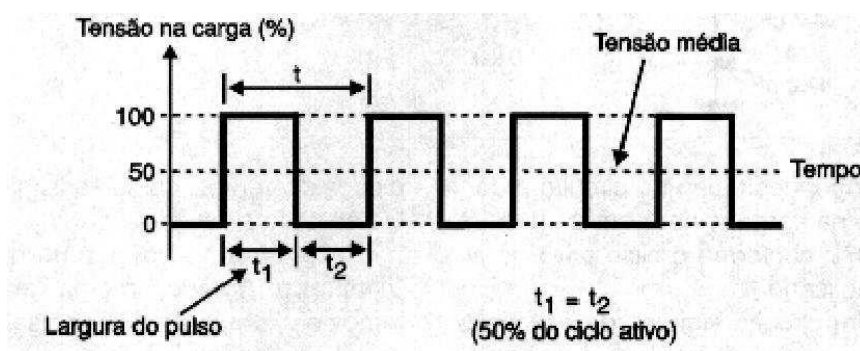
1.5 PWM

Um dos objetivos deste trabalho é poder controlar a velocidade com que os motores do robô giram para melhorar a sua resposta em uma curva. Para isso é necessário que a tensão fornecida para o motor seja variável, entretanto isso não seria possível, à primeira vista, utilizando puramente sinais digitais, que funcionam somente com zero ou tensão máxima. Do inglês, *pulse width modulation* (PWM), ou modulação por largura de pulso, serve justamente como técnica para obter um sinal analógico a partir de um digital.

Tendo um circuito que possui apenas uma chave e uma carga qualquer, quando a chave é ligada passará a potência máxima sobre a carga e quando ela está desligada a potência será nula. A pergunta é como fazer para obter um valor diferente de potência sobre a carga entre este valor nulo e a potência máxima, aí está o ponto em que entra a ideia do PWM (GHIRARDELLO, 2008).

O PWM consiste em um padrão de rápida alternância entre liga e desliga de um período T, na qual, parte do tempo ela estará em estado ativo e parte do tempo em estado desativado. Vemos assim que a potência média aplicada a carga pode ser diferente dos valores digitais e que a tensão sobre a carga varia. Veja figura 16.

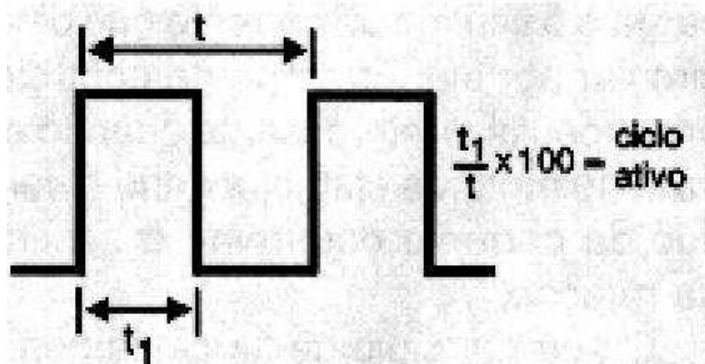
Figura 16 – PWM Variando a tensão média



Fonte: (GHIRARDELLO, 2008)

A relação $(t_1 \div t) \times 100\%$ é chamada de ciclo ativo (*duty-cycle*) do PWM, sendo o tempo em que ele fica ativo igual a t_1 e o período igual a t . Caso o ciclo ativo seja igual a 0% o circuito está desligado e caso o ciclo ativo seja 100% o circuito estará sempre ligado. A figura 17 demonstra o que foi explicado.

Figura 17– Definição de ciclo ativo



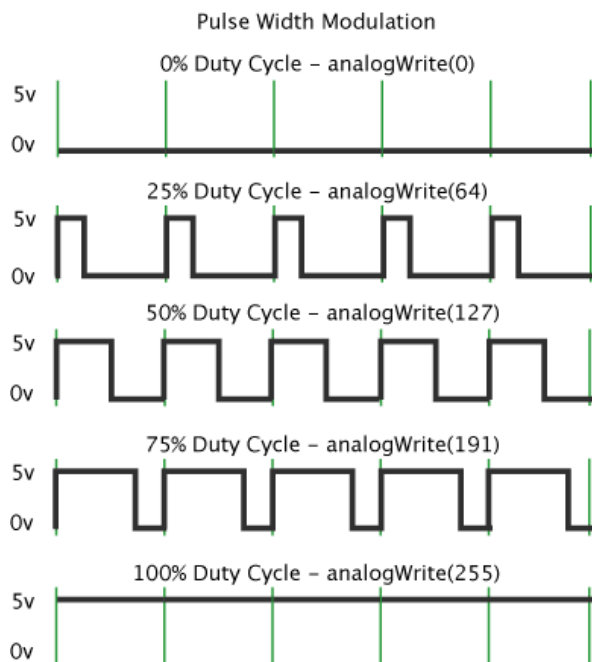
Fonte: (GHIRARDELLO, 2008)

1.5.1 Comandos do Arduino para PWM.

Utilizando o microcontrolador Arduino é possível fornecer este tipo de sinal utilizando as portas adequadas e o comando em código “`analogWrite(valor)`”, em que valor varia de 0 a 255, sendo 255 a tensão máxima.

Na figura 16 pode-se ver um gráfico representando o funcionamento do PWM para valores variados da função “`analogWrite`”. Veja a figura 18.

Figura 18 – Gráfico PWM com comandos do Arduino.



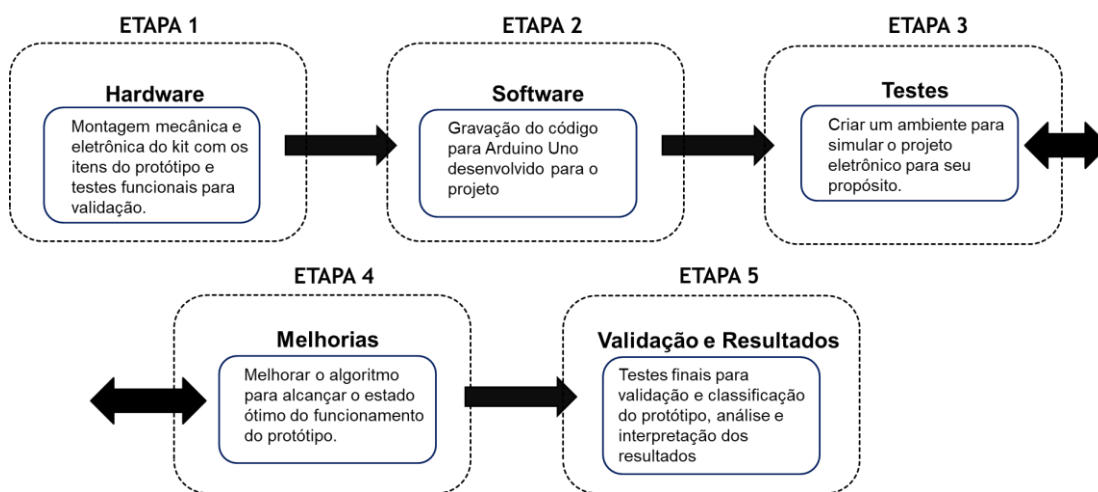
Fonte: (NASCIMENTO; DIAS; ZORZO, 2008)

2 MATERIAIS E METODOS

2.1 MONTAGEM

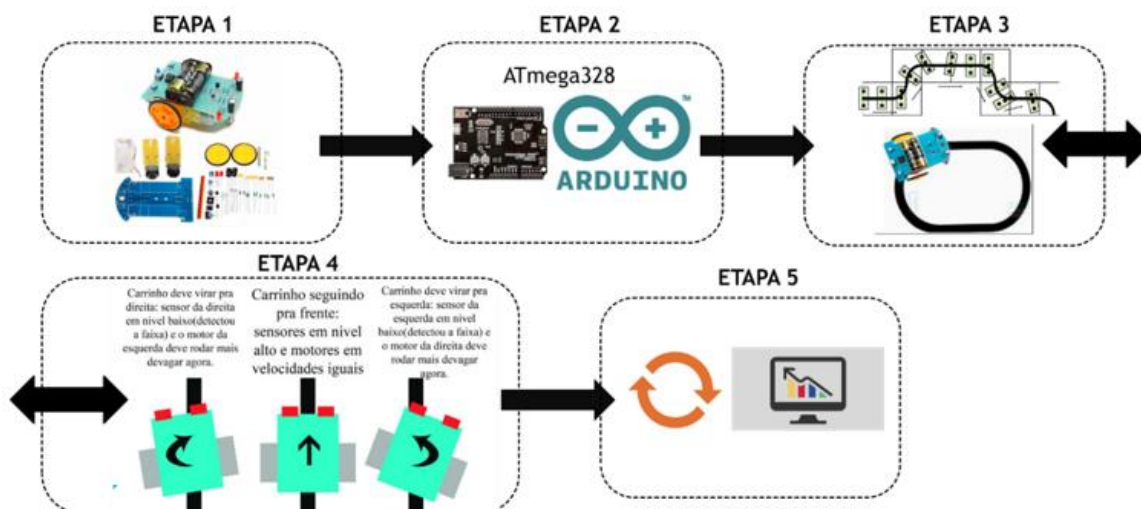
No presente projeto é apresentado o desenvolvimento de um protótipo de um veículo guiado automaticamente, utilizando a tecnologia de robô móvel na categoria AGV para transporte de suprimentos. Para a montagem do protótipo foi feito um estudo dos requisitos para tarefa em questão como pode se melhor entendido observando a figura 19, figura 20 logo abaixo.

Figura 19 – Diagrama em blocos das etapas do projeto (Descrição)



Fonte: Autoria própria

Figura 20 – Diagrama em blocos das etapas do projeto (Ilustração)



Fonte: Autoria própria

Na etapa 1 foi utilizado um kit com os componentes, listados na figura 21:

Figura 21 – Descrição do kit para montagem do AGV

Figura	Descrição
	Kit Chassi 2WD (2 rodas) Robô para Arduino
	Ponte H Dupla L298N
	Adaptador Bateria 9v para Arduino
	Jumpers - Macho/Femea - 20 Unidades de 20cm
	2x Sensores de Obstáculos Reflexivo Infravermelho
	Módulo Sensor de Distância Ultrassônico HC-SR04

Fonte: Aatoria própria

Foi realizada a montagem do kit começando pelo chassi, nele foi instalado todos os componentes, comecei instalando a “roda boba”, como podemos ver na figura 22.

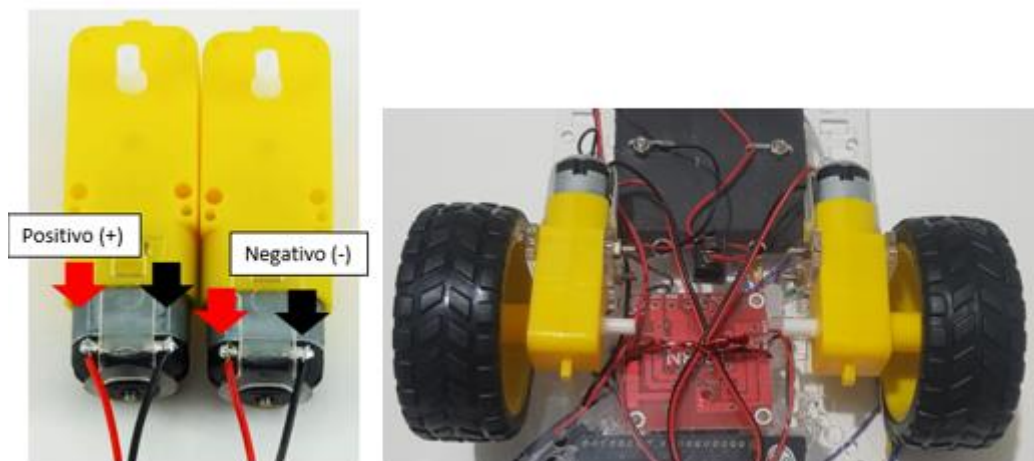
Figura 22 – “Roda boba”



Fonte: Aatoria própria

O próximo passo ainda na etapa 1 foi a montagem dos motores e rodas traseira, tomando cuidado com a devida polaridade dos mesmos, foi feita da seguinte maneira. Veja a figura 23:

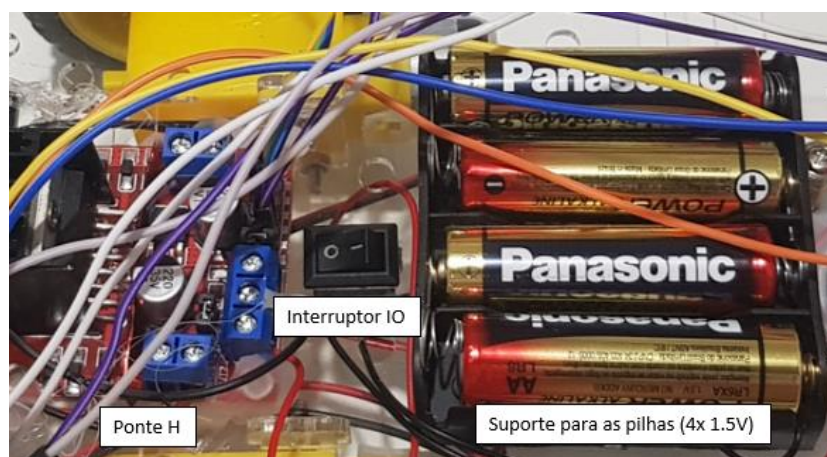
Figura 23 – Montagem dos motores DC no chassi



Fonte: Autoria própria

Continuando a montagem seguindo com cuidado, sempre pensando em balancear o peso dos componentes, para que fiquem bem posicionados, após a instalação dos motores e feito o teste de rotação, foi instalado o suporte para quatro pilhas alcalinas de 1,5 volts, somando 6 volts, que irão alimentar os motores controlado pelo circuito ponte H com tensão de operação entre 6 ~ 35 volts, como podemos ver na figura 24.

Figura 24 – Montagem do suporte de pilhas e ponte H

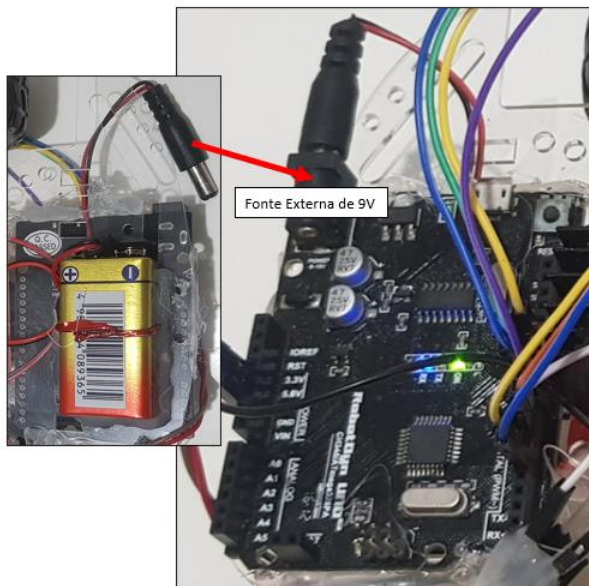


Fonte: Autoria própria

Segue-se com a montagem da placa Arduino Uno e sua fonte externa (Bateria alcalina de 9V), a tensão de alimentação externa para o modelo usado no projeto é de entre 7V e 12V, esta

bateria é dedicada para placa Arduino, podemos ver na figura 25 o local onde foram posicionadas.

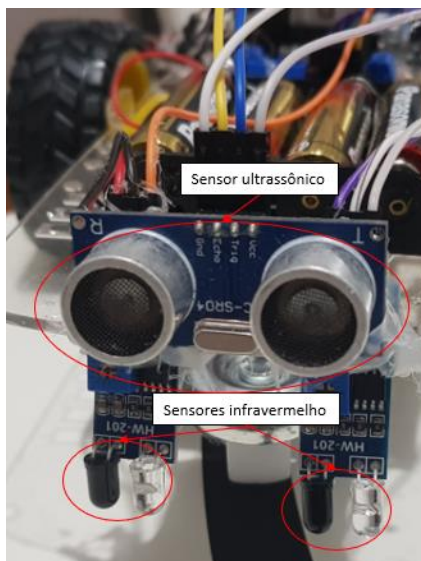
Figura 25 – Placa Arduino instalada no chassi



Fonte: Autoria própria

Finalizando a montagem com os sensores infravermelho de orientação e os sensores ultrassônico para detecção de obstáculos. Veja figura 26.

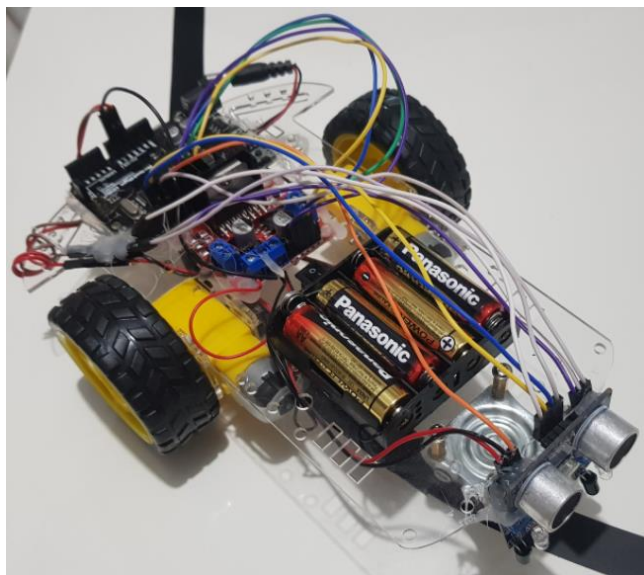
Figura 26 – Sensores infravermelho e sensor ultrassônico



Fonte: Autoria própria

Após finalizada a etapa 1 de montagem de todos componentes, foi iniciada a ligação dos fios conforme os requisitos do projeto, na figura 27 podemos ver o veículo AGV finalizado e no próximo tópico será mostrado em detalhes as conexões elétricas para o devido funcionamento do protótipo.

Figura 27 – Veículo guiado automaticamente AGV montado

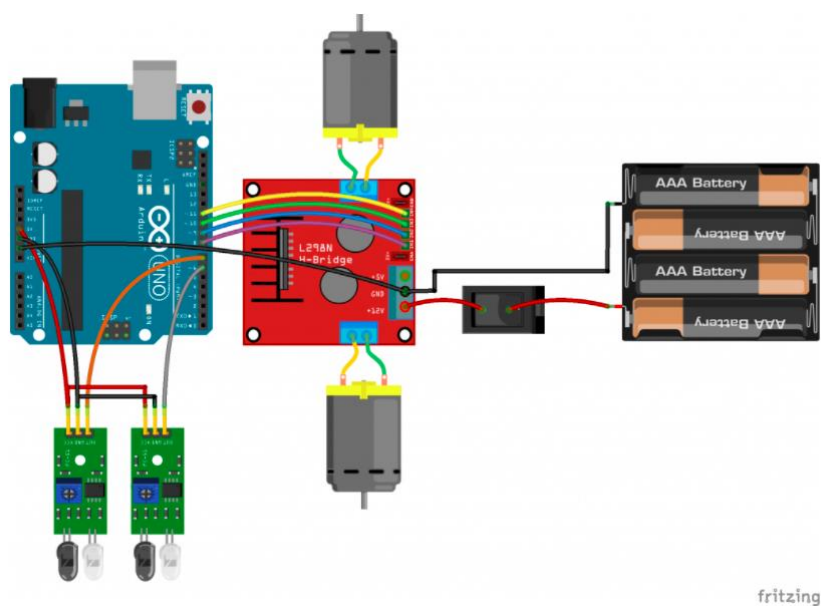


Fonte: Autoria própria

2.2 DIAGRAMA DAS FIAÇÕES

Com a utilização do software de código aberto *fritzing*, foi possível fazer as ligações de forma virtual antes de começar os testes, seguiu-se com as ligações sempre verificando as especificações de cada componente, com relação a sua capacidade de carga e performance para que não danificassem os mesmos, na figura 28 podemos ver o diagrama.

Figura 28 – Diagrama de ligações dos fios do protótipo



Fonte: Autoria própria

Também foi feita esta identificação (figura 29) para seguir com as pinagem no Arduino e nos referenciar na programação do microcontrolador.

Figura 29 – Pinagem

IN1 -> D8
IN2 -> D9
IN3 -> D10
IN4 -> D11
Out_Sensor1 -> D6
Out_Sensor2 -> D7

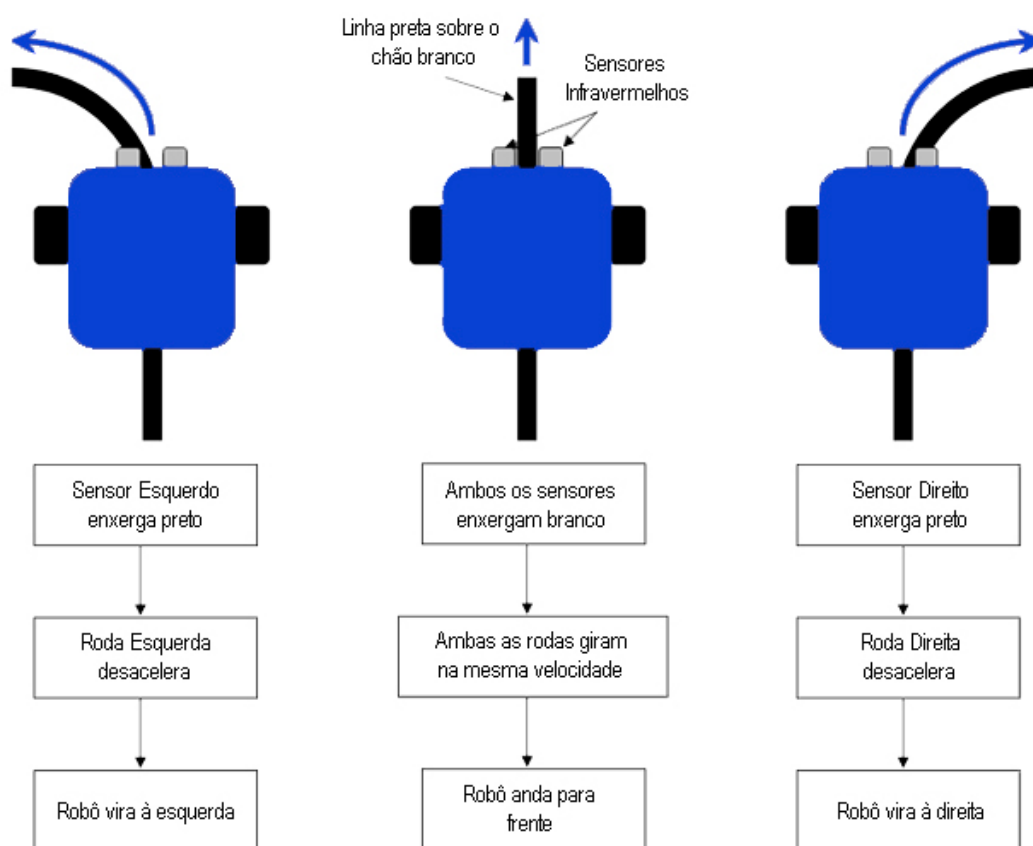
Fonte: Autoria própria

3 IMPLEMENTAÇÃO DO PROJETO

3.1 LÓGICA DE PROGRAMAÇÃO

Na segunda etapa foi feita a programação do microcontrolador. Para programar o Arduino é necessário entender a lógica dos sensores antes de iniciar a programação. Para isso foi feito um desenho para termos uma “visão macro” de todos eventos acontecendo, visto que a programação do Arduino é feita em “loop” e deve-se obedecer a uma sequência lógica. Pode ver visto na figura 30.

Figura 30 – Funcionamento dos dois sensores de infravermelho



Fonte: Autoria própria

Para caracterizar o comportamento dos sensores infravermelhos, foram realizadas medições de tensão de saída de cada um dos sensores. A tensão de cada um dos três sensores foi verificada através do acendimento do “led on” em sete posições do sensor em relação a faixa preta, sendo que o protótipo será posicionado em três direções distintas, veremos o código relacionado a seguir. As sete posições do sensor em relação a faixa preta são:

- Sensor sobre a superfície branca a esquerda da faixa;
- Sensor sobre a borda esquerda da faixa;
- Sensor entre a borda esquerda da faixa e o centro da faixa;
- Sensor sobre o centro da faixa;
- Sensor entre a borda direita da faixa e o centro da faixa;
- Sensor sobre a borda direita da faixa;
- Sensor sobre a superfície branca a direita da faixa.

Nesta parte do código fonte logo abaixo na figura 31 podemos ver as três situações em que a ponte H será “setada” de acordo com a devida orientação do veículo sobre a faixa preta com o fundo branco, vale ressaltar que para atingir o pleno funcionamento de navegação do protótipo foram necessários vários testes para que o mesmo se mantivesse sobre a linha preta em toda extensão.

Figura 31 – Código fonte para controle dos motores vs sensores

```
//Controle dos motores vs sensores.
//Neste processo armazenamos o valor lido pelo sensor na variável que armazena tais dados.
Sensor1 = digitalRead(pin_S1);
Sensor2 = digitalRead(pin_S2);
//Aqui está toda a lógica de comportamento do robô: Para a cor branca atribuímos o valor 0 e,
para a cor preta, o valor 1.
if((Sensor1 == 0) && (Sensor2 == 0)){ // Se detectar na extremidade das faixas duas cores
brancas
analogWrite(M1, velocidade); // Ambos motores ligam na mesma velocidade
analogWrite(M2, velocidade);
}
if((Sensor1 == 1) && (Sensor2 == 0)){ // Se detectar um lado preto e o outro branco
analogWrite(M1, 0); // O motor 1 desliga
analogWrite(M2, velocidade); // O motor 2 fica ligado, fazendo assim o carrinho virar
}
if((Sensor1 == 0) && (Sensor2 == 1)){ // Se detectar um lado branco e o outro preto
analogWrite(M1, velocidade); //O motor 1 fica ligado
analogWrite(M2, 0); // O motor 2 desliga, fazendo assim o carrinho virar no outro sentido
}
}
```

Fonte: Autoria própria

3.2 SIMULAÇÕES

Foi montado um pequeno circuito no formato oval, ver figura 32, para simular curvas e testar o funcionamento do protótipo. Todos os componentes foram colocados a prova, principalmente os sensores infravermelhos, tendo sido necessário reposicioná-los várias vezes avaliando a sua sensibilidade até manter o veículo sobre a faixa preta em toda extensão do circuito.

Figura 32 – Circuito oval para simulação.



Fonte: Autoria própria.

4 RESULTADOS OBTIDOS

Neste capítulo são exibidos os resultados obtidos após todas as etapas de implementação do protótipo. Após diversas simulações no circuito oval e também em diferentes tipos de curvas, obteve-se os seguintes resultados como pode-se ver na figura 33.

Figura 33 – Resultado dos “loops”

Numero de voltas completas	*Grau da curva	Velocidade PWM	Tempo por volta (s)
10	30°	150 (~58%)	18
0	45°	150 (~58%)	0
2	45°	120 (~47%)	26
10	45°	100 (~39%)	32

Fonte: Autoria própria

É possível deduzir, pelos resultados destes “loops”, que o melhor caso foi o primeiro, mas apenas para curvas de 30° graus então isto deverá ser levado em consideração quando for feita a demarcação no piso do chão de fábrica o qual este protótipo é disposto a transportar suprimentos e também para evitar acidentes caso o mesmo não seja efetivo na execução das curvas em que ele deverá passar até chegar no seu destino.

Para o propósito que este protótipo está sendo avaliado nesta pesquisa o seu comportamento durante os testes atende os requisitos de transporte de suprimentos em chão de fábrica, e também foi testado o sensor ultrassônico instalado na parte frontal do veículo para evitar acidentes, em que a distância pode ser setada de acordo com a logística do caminho e distância dos operadores. Durante os testes, o sistema foi configurado para detectar obstáculos a 15 cm de distância. Nessa situação, o AGV para completamente até que tenha uma intervenção humana para que ele volte a operar.

CONCLUSÃO

Neste trabalho foi desenvolvido um protótipo de um veículo guiado automaticamente (AGV) para transporte suprimentos em ambientes fabris. Para isso foram feitas revisões bibliográficas sobre os assuntos: plataforma Arduino Uno equipado com microcontrolador ATmega328PA, motores com caixa de redução e eixo duplo modelo DG01D-A130 , circuito eletrônico chamado ponte H para controle do acionamento e sentido de rotação do motor modelo L298N, sensor infravermelho usado para devida navegação do protótipo sobre uma linha demarcada com um fundo em contraste, neste caso se utilizou a linha preta com fundo branco, e outro sensor ultrassônico para detecção de obstáculos.

A implementação da arquitetura do projeto foi realizada com êxito, de acordo com as etapas descritas no capítulo de materiais e métodos. Com esta arquitetura, foi possível realizar os testes e simulações de forma prática em um cenário apropriado a tarefa proposta a ser validada, foram feitos vários “loops”, voltas em um circuito oval feito com fita preta, em diversas situações, alterando os parâmetros que mais influenciavam no resultado de saída, que é o de transportar com segurança e agilidade auxiliando as pessoas nesta atividade.

Em relação aos resultados obtidos na implementação do projeto, tivemos um melhor resultado e foi validade com dez voltas completas, onde o protótipo se comportou bem em todo percurso, com a velocidade do PWM “setada” em aproximadamente 58% com curvas suaves de 30° Graus de dobra, com o tempo de 18 segundos.

Se baseando nos resultados obtidos, verificar-se que, por meio da tecnologia de um robô móvel e um sistema embarcado é possível trazer melhoria ao setor logístico, solucionando a problemática da ineficiência humana em trabalhos repetitivos que requerem muita destreza e atenção. Desta forma, a hipótese apresentada na introdução teve sua veracidade comprovada.

O próximo passo, que ainda pode ser desenvolvido em ambiente acadêmico, seria a ampliação da pesquisa para aprimorar o protótipo com a integração de uma plataforma IHM para configuração e registro de rotas, e com estes dados trazer melhorias ao processo. Adequar as normas de segurança pertinente para evitar acidentes, como por exemplo a inclusão de amortecedores frontais e emissão de som sutil ao ouvido humano durante o percurso para sinalizar sua movimentação próximo às pessoas.

Com os recursos disponíveis será possível reunir a tecnologia necessários para levar a solução a um patamar que possa fazer a diferença na vida das pessoas no meio produtivo, trazendo maior segurança, trazendo um auxílio ao trabalho para melhorar a qualidade de vida neste ambiente.

REFERÊNCIAS

- ANTUNES, F. L. M. **Motores de Corrente Contínua**. 2009. Disponível em: <[http://www.dee.ufc.br/fantunes/Maquinas EletricasI/Aulas/MotoresCorrenteContinua-Impressao.pdf](http://www.dee.ufc.br/fantunes/MaquinasEletricasI/Aulas/MotoresCorrenteContinua-Impressao.pdf)> Acesso em: 13 jan.2022.
- FILIFELOP, **Motor DC 3-6V com Caixa de Redução e Eixo Duplo**, p. 1, 2022. Disponível em: <<https://www.filifeelop.com/produto/motor-dc-3-6v-com-caixa-de-reducao-e-eixo-duplo/>>. Acesso em: 17 jan. 2022.
- FILIFELOP, **Como conectar o Sensor Ultrassônico HC-SR04 ao Arduino**, p. 1, 2022. Disponível em: < <https://www.filifeelop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/>>. Acesso em: 17 jan. 2022.
- GHIRARDELLO, A. **Apostila sobre Modulac, ̃ao PWM**. 2008. Disponível em: <<http://www.eletronica.org/arqapostilas/apostilapwm.pdf>> Acesso em: 14 jan. 2022.
- INTERFACING with Hardware. [S.l.]: **Arduino**, 2009. Disponível em: <<http://www.arduino.cc/playground/Main/InterfacingWithHardware>> Acesso em: 14 jan. 2022.
- JONES, J. L.; FLYNN, A. M.; SEIGER, A. B. **Mobile Robots: Inspiration to Implementation**. 2. ed. [S.l.]: Natick, Massachusetts: A K Peters, 1999.
- NASCIMENTO, L. P.; DIAS, L. del C.; ZORZO, R. de R. **Cadeira de Rodas Motorizada** — Universidade Tecnológica Federal do Paraná, Curitiba, 2008.
- PATSKO, L. F. **Tutorial Montagem da Ponte H**. 2006. Disponível em: <[http://www.maxwellbohr.com.br/downloads/Tutorial Eletronica - Montagem de uma Ponte H.pdf](http://www.maxwellbohr.com.br/downloads/TutorialEletronica-MontagemdeumaPonteH.pdf)> Acesso em: 17 jan. 2022.
- STMICROELECTRONICS, S. T. **L298 datasheet**. 2000. Disponível em: <<http://www.datasheetcatalog.org/datasheet/SGSThompsonMicroelectronics/mXrqqxz.pdf>> Acesso em: 17 jan. 2022
- WIKIPEDIA, **ATmega328**, p. 1, 2018. Disponível em: <<https://pt.wikipedia.org/wiki/ATmega328>>. Acesso em: 17 jan. 2022.
- YAGHOUBI, S. et al. Designing and Methodology of Automated Guided Vehicle Robots/Self-Guided Vehicle Systems, Future Trends. **International Journal of Research and Reviews in Applied Sciences**, Ahar, Outubro 2012. 6.

APÊNDICE A – CÓDIGO DO MICROCONTROLADOR ATMEGA328

```

//*****
//UNIVERSIDADE DO ESTADO DO AMAZONAS
//
//Projeto de conclusão de curso II
//
//Autor: Glaucio Marculino Muniz
//
//Este programa tem a finalidade controlar dois motores através de um circuito de ponte H de
//
//acordo com os status dos sensores infravermelhos para navegação de um veículo guiado
//
//automaticamente (AGV).
//*****

//Definição dos pinos de controle do motor
#define M1 9 // Pino_Velocidade 1º Motor ( 0 a 255)_ Porta ATV_A ponte H;
#define M2 11 //Pino_Velocidade 2º Motor ( 0 a 255) _ Porta ATV_B ponte H;
#define dir1 8 //Pino_Direção do 1º Motor: Para frente / Para trás (HIGH ou LOW)_ porta
IN1 ponte H;
#define dir2 10 //Pino_Direção do 2º Motor: Para frente / Para trás (HIGH ou LOW)_
porta IN3 ponte H;

//Definição dos pinos dos sensores
#define pin_S1 7
#define pin_S2 6
bool Sensor1 = 0;
bool Sensor2 = 0;

//variável responsável por controlar a velocidade dos motores
int velocidade = 100;

// Definição dos pinos do sensor ultrasonico
int PinTrigger = 4; // Pino usado para disparar os pulsos do sensor
int PinEcho = 5; // pino usado para ler a saída do sensor
float TempEcho = 0;
const float VelocidadeSom_mpors = 340; // em metros por segundo
const float VelocidadeSom_mporus = 0.000340; // em metros por microsegundo

```

```
void setup(){

    //Setamos os pinos de controle dos motores como saída
    pinMode(M1, OUTPUT);
    pinMode(M2, OUTPUT);
    pinMode(dir1, OUTPUT);
    pinMode(dir2, OUTPUT);

    //Setamos a direção inicial do motor como 0, isso fará com que ambos os motores girem
para frente
    digitalWrite(dir1, LOW);
    digitalWrite(dir2, LOW);

    //Setamos os pinos dos sensores como entrada
    pinMode(pin_S1, INPUT);
    pinMode(pin_S2, INPUT);

    // Configura pino de Trigger como saída e inicializa com nível baixo
    pinMode(PinTrigger, OUTPUT);
    digitalWrite(PinTrigger, LOW);
    pinMode(PinEcho, INPUT); // configura pino ECHO como entrada
    // Inicializa a porta serial
    Serial.begin(9600);
    delay(100);

}

void loop(){

    //Sensor ultrasonico - detectar obstáculos
    // Envia pulso para o disparar o sensor
    DisparaPulsoUltrasonico();
    // Mede o tempo de duração do sinal no pino de leitura(us)
    TempEcho = pulseIn(PinEcho, HIGH);
    Serial.println("Distancia em metros: ");
```

```

Serial.println(CalculaDistancia(TempEcho));
Serial.println("Distancia em centimetros: ");
Serial.println(CalculaDistancia(TempEcho)*100);

```

//Condição quando detectar o obstáculo a menos de 4 centímetros de distância do sensor, desligar os motores em velocidade zero.

```

while((CalculaDistancia(TempEcho)*100) <= 15){
  digitalWrite(M1, 0);
  digitalWrite(M2, 0);
  digitalWrite(dir1, 0);
  digitalWrite(dir2, 0);
  digitalWrite(pin_S1, 0);
  digitalWrite(pin_S2, 0);
}

```

//Controle dos motores vs sensores.

//Neste processo armazenamos o valor lido pelo sensor na variável que armazena tais dados.

```

Sensor1 = digitalRead(pin_S1);
Sensor2 = digitalRead(pin_S2);

```

//Aqui está toda a lógica de comportamento do robô: Para a cor branca atribuímos o valor 0 e, para a cor preta, o valor 1.

if((Sensor1 == 0) && (Sensor2 == 0)){ // Se detectar na extremidade das faixas duas cores brancas

```

  analogWrite(M1, velocidade); // Ambos motores ligam na mesma velocidade
  analogWrite(M2, velocidade);
}

```

if((Sensor1 == 1) && (Sensor2 == 0)){ // Se detectar um lado preto e o outro branco

```

  analogWrite(M1, 0); // O motor 1 desliga
  analogWrite(M2, velocidade); // O motor 2 fica ligado, fazendo assim o carrinho virar

```

```
}

if((Sensor1 == 0) && (Sensor2 == 1)){ // Se detectar um lado branco e o outro preto
  analogWrite(M1, velocidade); //O motor 1 fica ligado
  analogWrite(M2, 0); // O motor 2 desliga, fazendo assim o carrinho virar no outro sentido
}
}

// Função para enviar o pulso de trigger
void DisparaPulsoUltrassonico(){
  // Para fazer o HC-SR04 enviar um pulso ultrassonico, nos temos
  // que enviar para o pino de trigger um sinal de nivel alto
  // com pelo menos 10us de duração
  digitalWrite(PinTrigger, HIGH);
  delayMicroseconds(10);
  digitalWrite(PinTrigger, LOW);
}

// Função para calcular a distancia em metros
float CalculaDistancia(float tempo_us){
  return((tempo_us*VelocidadeSom_mporus)/2);
}
```