



MONTAGEM DE BANCADA DE TESTE PARA RECONHECIMENTO DE SINALIZAÇÃO VIÁRIA E SOFTWARE DE TOMADA DE DECISÃO PARA VEÍCULOS AUTÔNOMOS

Alex Luiz Cibien e Silva¹

Marcos Welliton Daros Felipe²

Fernando Guessi Placido³

Everlise Maestrelli⁴

Marcos Antônio Jeremias Coelho⁵

Resumo: O projeto desenvolve uma bancada de testes e um sistema de tomada de decisão destinado ao reconhecimento de sinalização viária e à tomada de decisão em veículos autônomos, integrando sensores de visão computacional, RFID, rede CAN e um display LCD. Usando um método de simulação de Monte Carlo para gerar dados, o sistema analisa e responde a entradas como placas de sinalização e comandos do pedal, o sistema conta com um sensor de RFID e um sistema de visão computacional. A comunicação ocorre através de um protocolo serial desenvolvido, entre um microcontrolador ATmega328P e um Raspberry Pi, que facilita o envio de informações para o display, enquanto um filtro de média móvel melhora a precisão da leitura dos dados do potenciômetro. Os testes realizados mostraram a eficiência no reconhecimento de placas e a estabilidade na decisão. A bancada de teste permite que o sistema execute ações de acordo com um nível de prioridade pré-definido, onde placas de pedestres e de pare têm prioridade para garantir a segurança. Os resultados mostram que cobrir mais condições no sistema de decisão pode aumentar a sua eficácia em ambientes do mundo real. Conclui-se que a solução é uma plataforma promissora para pesquisa em tecnologia autônoma e oferece potencial para aprimoramentos futuros, incluindo diversidade de controles, integração com redes automotivas e melhorias na comunicação com sensores e sistemas de veículos conectados.

Palavras-chave: Carro Autônomo. Sinalização Viária. Rede CAN. Python.

1. INTRODUÇÃO

Veículos de alto nível de automação estão se aproximando de seu lançamento para comercialização. Porém ainda se faz necessário constantes evoluções em sistemas embarcados e comportar as novas tecnologias que incluem

¹ Graduando em Engenharia Mecatrônica, 2024-2. E-mail: alexci@hotmail.com

² Prof. do Centro Universitário UniSATC. E-mail: marcos.welliton@satc.edu.br

³ Prof. do Centro Universitário UniSATC. E-mail: fernando.placido@satc.edu.br

⁴ Prof. do Centro Universitário UniSATC. E-mail: everlise.maestrelli@satc.edu.br

⁵ Prof. do Centro Universitário UniSATC. E-mail: marcos.coelho@satc.edu.br



sensores de proximidade, câmeras e visão computacional, leituras de *tags*, sensores infravermelhos, entre outras, para que seja possível um meio de locomoção seguro, reduzindo o número de acidentes e elevando a conectividade visto as tendências de implantação de cidades inteligentes nas grandes metrópoles (Borges, 2024).

Atualmente, o excesso de veículos nas grandes cidades gera problemas na fluidez do tráfego, ocasionando atrasos, acidentes leves ou fatais, com a tendência na elevação de vendas de veículos, problemas como acidentes e engarrafamento, intensificando principalmente em datas comemorativas, como no carnaval, onde Barcellos (2024) apontou que em 2024 nas rodovias estaduais de Santa Catarina, oito fatalidades aconteceram por embriaguez, excesso de velocidade e ultrapassagem em local proibido. Em 2023, foram registradas duas mortes. Dentro dessas situações, veículos autônomos podem aprimorar a porção de prevenção de acidentes, já que diminui a possibilidade de erros sem um motorista no volante (Nascimento, 2023).

Para um padrão dentro das tecnologias entre as automações veiculares, a associação *SAE International* dividiu em seis níveis de automação, onde o nível zero é o veículo sem assistência dos controles, até o nível 5 em que o veículo passa a ser 100% autônomo sem qualquer necessidade de interferência do usuário para dirigibilidade (SAE BRASIL, 2024).

Com o objetivo de aumentar a segurança, os veículos autônomos com comunicação entre si, que usam redes Vanet (redes temporárias de comunicação entre veículos ou ad-hoc veiculares), em que todos os integrantes estão cientes do que o ocorre em volta e onde cada um está no momento, assim não há perda de sincronia e o trânsito fica mais fluído, tendo que se chama de Mente Coletiva ou Mente de Colmeia, a autonomia proporcionada facilita e organiza o ambiente de trânsito, diminuindo estresse, atrasos e a redução de acidentes (Nascimento, 2023).

No contexto de novas tecnologias autônomas e cidades inteligentes, este projeto propõe o desenvolvimento de um sistema de controle e tomada de decisão para veículos autônomos com foco na leitura de placas de sinalização. Além disso, realizar a construção de uma bancada unindo o sistema de propulsão, sistema de visão computacional e leitura de *tags*, fazendo uso de métricas e simulações, em que seja possível a implementação em outros modelos que tenham essa mesma plataforma de inversor.

2. REVISÃO BIBLIOGRÁFICA

Este capítulo tem como finalidade explorar as principais tecnologias, bem como métodos e estudos na área. As informações pertinentes foram reunidas e organizadas de forma a garantir todo o embasamento teórico e fundamental para a execução e desenvolvimento do trabalho em questão.

2.1. MOBILIDADE AUTÔNOMA

Cidades conectadas estão se tornando realidade, e com isso, a mobilidade autônoma é um dos pilares que devem estar instaurados nessas cidades, sendo um meio de transporte que visa facilitar a locomoção seja para o trabalho, para casa, em viagens e demais afazeres (Borges, 2024).

No Brasil, apenas carros autônomos de níveis 2 e inferiores, exigem a presença de um condutor habilitado por trás do volante. Não há legislação específica a esse meio de transporte, portanto limitando apenas a uma direção com supervisão do motorista (Capomaccio, 2023).

Catalogando os veículos com base no seu nível em automação, a SAE J3016, divide em seis níveis: nível zero, o veículo não possui intervenção do sistema veicular, nível um, com manutenção da aceleração para manter a certa velocidade, nível dois, o veículo é capaz de acelerar e frear sem intervenção de motorista, nível três, em que a autonomia se faz presente na direção do veículo, nível quatro, o motorista não precisa estar mais atento ao seu redor, podendo este realizar outras tarefas que desejar enquanto dirige, e o nível cinco, o veículo é 100 % autônomo, o veículo leva a qualquer ponto que o usuário desejar (SAE BRASIL, 2024). A Figura 1 detalha esses níveis que podem ser encontrados de automação veicular.

Figura 1: Níveis de autonomia veicular.



Fonte: Adaptado de SAE INTERNATIONAL (2024)

Atualmente nos Estados Unidos, a empresa de desenvolvimento autônomo *Waymo*, subsidiária do *Google*, possui veículos autônomos conhecidos como robotáxis. Segundo a agência Reuters (2024), eles estão em funcionamento em *Los Angeles* e em São Francisco, que aprovaram o aumento de frota e uso desses veículos para transporte público. Na Figura 2 mostra um desses modelos de robotáxi em teste.

Figura 2: Robotáxi Waymo durante testes em São Francisco.



Fonte: REUTERS/Paresh

O veículo é uma modificação de um veículo elétrico da marca Jaguar. A proposta da *Waymo* é adaptar um veículo comum com a tecnologia autônoma, inserindo e reconfigurando equipamentos e atuadores no automóvel.

2.2. CARROS ELÉTRICOS

A tecnologia embarcada nos veículos está tomando um rumo direcionado a eletrificação automotiva. Os veículos elétricos trouxeram uma evolução significativa em relação a eficiência do powertrain (trem de força é onde inclui o trajeto de força do motor a roda), pois com o eletromagnetismo é possível possuir torque com o mínimo de rotação, já nos motores a combustão, há torque e potência apenas com o motor em rotações mais altas (Hughes; Drury, 2019).

No território brasileiro os veículos elétricos estão tendo um grande aumento no número de vendas e emplacamentos. De acordo com a ABVE Data (2024) (Associação Brasileira do Veículo Elétrico) registrou que no mês de março de 2024 (13.613 vendas) houve um crescimento no emplacamento de veículos elétricos em



127% superando o mês de março de 2023 (5.989 vendas), que já havia representado uma grande evolução com março de 2022 (55,5%). A Tabela 1 distribui essas quantidades por tipos de veículos elétricos.

Tabela 1: Quantidade de vendas de eletrificados leves no Brasil por mês e trimestre.

Tecnologia	mar/24	mar/23	Total - 1º tri/2024	% - mar/24	% - 1º tri/2024
BEV (Battery Electric Vehicle)	6.137	587	14.134	45,0%	39,0%
PHEV (Plug-in Hybrid Electric Vehicle)	3.128	2.095	10.632	23,0%	29,5%
HEV FLEX (Hybrid Electric Vehicle Flex)	2.020	1.917	5.117	15,0%	14,0%
HEV (Hybrid Electric Vehicle)	1.221	346	3.351	9,0%	9,5%
MHEV (Mild Hybrid Electric Vehicle)	1.107	1.044	2.856	8,0%	8,0%
TOTAL	13.613	5.989	36.090		

Fonte: Adaptado de ABVE Data (2024)

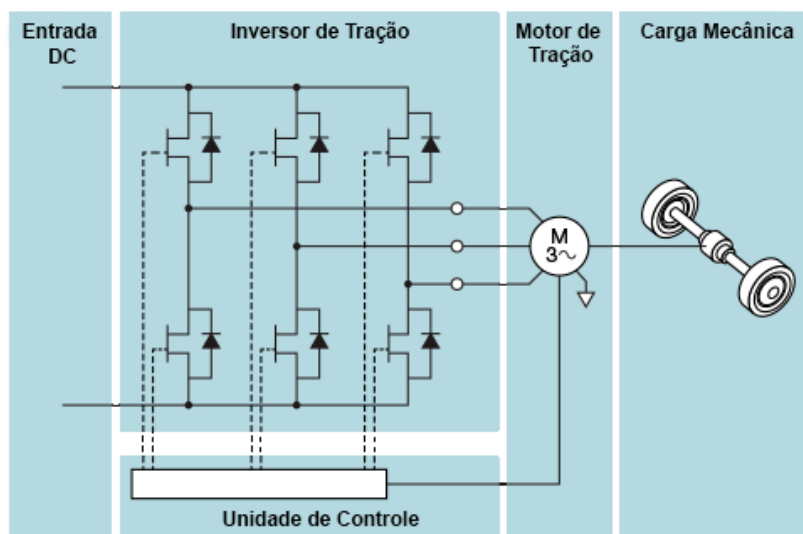
Tendo uma comparação de janeiro a março de 2024, as vendas de eletrificados alcançaram 36.090 e bateram mais um recorde, com aumento de 145 % sobre o mesmo período em 2023 (14.786).

2.2.1 Inversor de tração

Um inversor de tração é um componente dos veículos elétricos e seu principal objetivo é transformar a corrente contínua do banco de baterias do veículo em corrente alternada fornecendo energia ao motor de tração. Seu principal benefício é a eficiência energética por permitir o controle da rampa de aceleração, assim não forçando o motor e mantendo o torque disponível para movimentar cargas.

A Figura 3 ilustra o caminho da energia da bateria até ser transformada em energia mecânica pelo motor, sendo controlado por um inversor, que é dividido em dois blocos de componentes controladores e de potência: ponte inversora e unidade de controle.

Figura 3: Diagrama de funcionamento de um inversor de frequência.



Fonte: Adaptado de Tektronix (2024)

A unidade de controle (controlada por um microcontrolador interno) gera um sinal PWM (*Pulse Width Modulation*) por meio de algoritmos DSP (*Digital Signal Processing*) responsável por auxiliar e manter o tempo e frequência do sinal, posteriormente é comutado pela ponte inversora (drivers de gate) essa responsável por compreender o sinal da unidade de controle em uma onda senoidal, que faz com que o motor seja controlado pelo inversor de tração (TEKTRONIX, 2024).

2.3. VISÃO COMPUTACIONAL

Visão computacional é um campo da inteligência artificial que busca capacitar máquinas a interpretar e compreender o mundo visual de maneira semelhante ao ser humano. Em sua essência, envolve a análise e a extração de informações úteis de imagens, vídeos ou sequências de dados visuais, utilizando algoritmos avançados para identificar padrões, objetos e ações (Sivaraman; Trivedi, 2013).

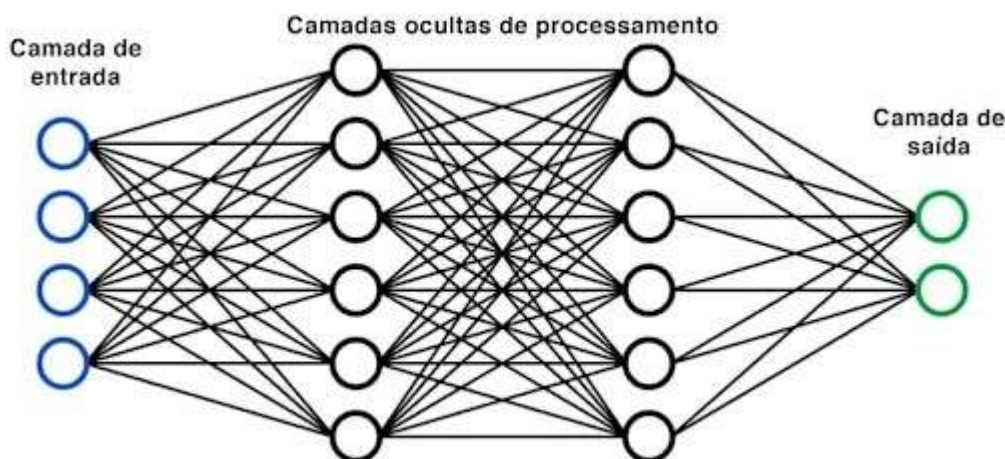
A identificação de sinais de trânsito é uma das aplicações da visão computacional. Por meio de algoritmos de processamento de imagem, esta tecnologia identifica e interpreta sinais de trânsito, possibilitando que os veículos obedeçam as normas de circulação. Diversos tipos de sinais de trânsito podem ser detectados pela visão computacional, como os de limite de velocidade, estacionamento obrigatório, proibição de conversão e sentido único. Os algoritmos empregados para

reconhecimento de sinais utilizam técnicas como segmentação de imagens, classificação de padrões e redes neurais convolucionais para identificar os sinais em imagens capturadas por câmeras de trânsito (Sivaraman; Trivedi, 2013).

2.3.1 Rede neural artificial

É um tipo de rede usado em inteligências artificiais (IA's) que busca implementar modelos matemáticos assimilando-se a uma rede neural biológica. Essa rede tem inspiração direta com o cérebro humano e o modelo de sinapses dos nossos neurônios para fazer conexões (Ferneda, 2006). A Figura 4 apresenta uma estrutura de rede neural genérica amplamente utilizada.

Figura 4: Representação de um modelo de rede neural artificial.



Fonte: Opencadd (2022)

Os aprendizados para redes neurais podem ser caracterizados por três influências:

- Aprendizado supervisionado: onde um terceiro infere na rede a resposta intencionada para a entrada e a rede aprende em cima dessas etiquetas, um exemplo da técnica é a SVM;
- Aprendizado não supervisionado: a IA tenta encontrar padrões de forma autônoma, um exemplo é a clusterização (K-Means);
- Aprendizado por reforço: o terceiro afere a resposta apresentada pela rede, um exemplo desse aprendizado é o *Q-Learning* (Ferneda, 2006).



2.4. SENSORES RFID

Visando substituir a leitura óptica em sistemas de registro e leitura de produtos, a tecnologia Identificador de Rádio Frequência (do inglês *Radio-Frequency Identifier*, RFID) utiliza de etiquetas (do inglês *tags*), que armazena dados de forma passiva ou ativa, para posterior leitura dessas informações (Ciriaco, 2009).

O sistema RFID possui dois métodos para armazenagem de dados dentro dessas etiquetas:

- Ativa: seu custo é maior e é necessário uma alimentação elétrica para transmitir seu sinal sobre uma distância razoável, além de permitir armazenamento em memória RAM capaz de guardar até 32 kB.
- Passiva: utilizam a rádio frequência do leitor para transmitir o seu sinal e normalmente têm com suas informações gravadas permanentemente quando são fabricadas. Contudo, algumas destas etiquetas permitem gravação (Ciriaco, 2009).

2.5. REDES AUTOMOTIVAS

Dentro de um sistema automotivo eletroeletrônico podem ser encontrados sensores, atuadores, módulo de injeção, módulo de carroceria, painel, tela multimídia e módulo ABS (*Anti-lock Braking System*), ESP (*Electronic Stability Program*), para que isso tudo possua comunicação entre si é necessário um meio físico que possua comunicação entre esses módulos. As redes automotivas entram como uma solução e, principalmente, facilitação para transporte de dados de um módulo para outro, com a redução de fios em meio aos chicotes e formando uma centralização distribuída. Hoje em dia as redes, que são mais utilizadas nos veículos urbanos são: Rede LIN (Local Interconnect Network), linha K, rede SAEJ1850 e Rede CAN (do inglês, *Controller Area Network*) (Nolte; Hansson; Bello, 2005).

A rede LIN desenvolvida por um consórcio entre as grandes empresas automotivas (Audi, BMW, Daimler-Chrysler, Volcano, Volkswagen e Volvo) e a fabricante de componentes eletrônicos Motorola, com o lançamento da LIN 1.1 em 2000, começou a ser utilizada pela primeira vez em veículos de fabricação em série em 2001. Depois em 2003 com o lançamento da rede LIN 2.0, se espalhou o uso das demais montadoras liderando junto com a rede CAN atualmente no mercado. A rede



LIN é muitas vezes usada como uma rede auxiliar da rede CAN, onde uma realiza tarefas comuns, enquanto a outra realiza atividades com maior instantâneas, que necessitam de uma alta demanda de dados e atualizações (Morais, 2012).

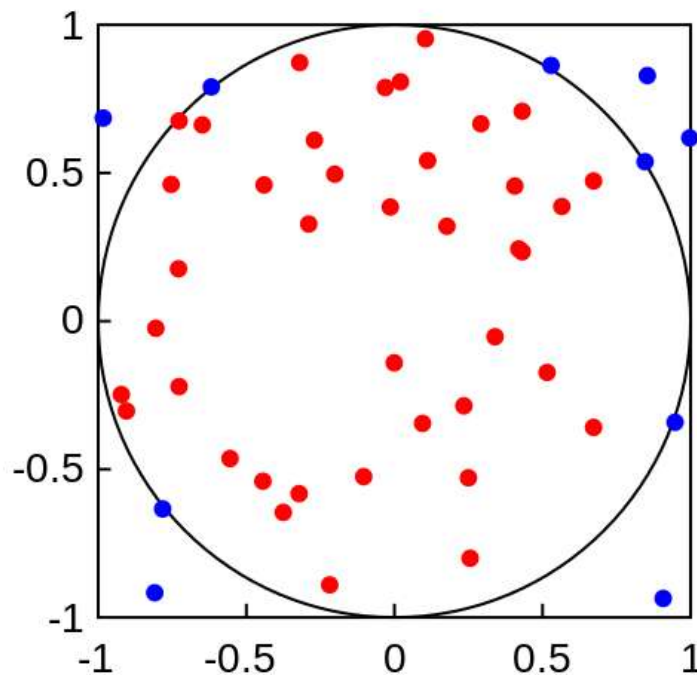
Proposta pela empresa Robert Bosch GmbH por volta de 1980, a rede CAN foi projetada para interligar sistemas automotivos de controle, simplificando a fiação, o que antes era uma distribuição centralizada, ou seja, todos os sensores, atuadores eram calculados e comandados em uma única unidade de controle e processamento, localizada normalmente próximo ao motor (Morais, 2012).

A rede CAN possui uma arquitetura com base em uma comunicação por barramento, em que todas as informações são distribuídas para os nós que estão conectados à rede. Em cada nó da rede está presente uma unidade controladora embarcada, e um transceptor que tem conexão direta com a rede CAN. Os nós da rede (controladores) tem a função de enviar informações para outros nós, e são eles quem decidem qual informação usar (periféricos) ou descartar para fim próprio (Morais, 2012).

2.6. MÉTODO DE MONTE CARLO

O método de Monte Carlo é uma abordagem estatística utilizada para resolver problemas complexos de simulação, destacando-se em aplicações de aprendizado de máquina e inteligência artificial (IA). Fundamentado na geração de números aleatórios, o método permite a construção de modelos para prever cenários possíveis em sistemas onde há grande incerteza. Este método é útil para avaliar a probabilidade de eventos em problemas cujo comportamento analítico exato seria inviável de se resolver. O avanço computacional tornou possível rodar milhões de iterações com Monte Carlo, em tempo relativamente curto, aprimorando a precisão das previsões em IA e ampliando sua aplicação em áreas como reconhecimento de padrões e análise de imagens. A Figura 5 ilustra a utilização desse método para calcular a probabilidade de os pontos ficarem para fora do círculo (Sutton; Barto, 2014).

Figura 5: Método de Monte Carlo em ação para medir probabilidade.



Fonte: Hoja del Lunes (2022)

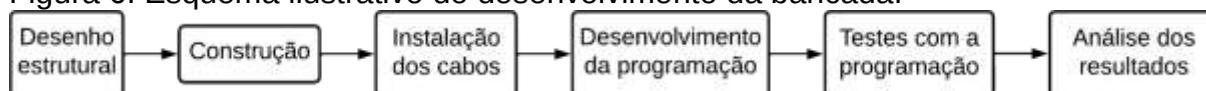
A abordagem de Monte Carlo inclui várias etapas, tal como a definição do modelo matemático, a identificação das variáveis aleatórias e a execução de repetidas iterações para obter uma distribuição de resultados. Os estudos com esse método mostram que essa metodologia é eficaz não apenas em prever valores médios esperados, para o desenvolvimento de técnicas de *Monte Carlo Tree Search* (MCTS, em português, Árvore de procura de Monte Carlo), amplamente usadas em jogos de estratégia e otimização de políticas em aprendizado de máquina. Esta característica tem tornado o método uma ferramenta essencial para análises de risco, planejamento e controle de projetos e tomada de decisão sob incerteza (Rubinstein; Kroese, 2008).

3. PROCEDIMENTO EXPERIMENTAL

Nesta seção, serão descritos os métodos experimentais utilizados para conduzir o desenvolvimento de uma bancada de testes. O objetivo é integrar e validar a comunicação entre os seguintes componentes: inversor, sistema de visão computacional, leitor RFID, sistema de tomada de decisão e display LCD. A bancada será realizada com o intuito de desenvolver o protótipo e adquirir dados, para posterior análise com uso de métricas e matriz de confusão.

Como método de validação do sistema de tomada de decisão, foi utilizado o método de Monte Carlo, para geração dos sinais de entrada. Analisando meios em que poderiam ser realizados a bancada, a Figura 6 demonstra o diagrama de blocos elaborado com os processos que foram realizados visando alcançar os objetivos propostos.

Figura 6: Esquema ilustrativo do desenvolvimento da bancada.

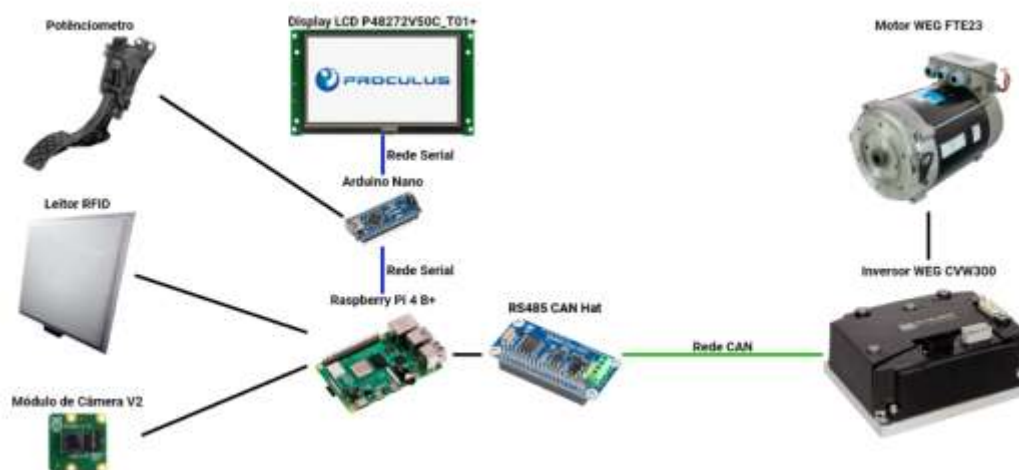


Fonte: Do Autor (2024)

O desenvolvimento do projeto iniciou com a aquisição dos componentes, grande parte foi adquirida pela própria instituição, os quais serão descritos ao decorrer do projeto. Os experimentos foram realizados nos laboratórios da Engenharia Mecatrônica localizados no campus Criciúma do centro universitário UNISATC, em que possuem ferramentas de medição e análise que serão descritas adiante.

O sistema de tomada de decisão terá como entrada um sensor de potenciômetro representando o pedal, um sistema de visão computacional, com a redundância do sistema de sensor RFID confirmando qual a placa que foi registrada pelos sensores e como saída um sinal para informar ao inversor se será necessário diminuir a velocidade, acelerar ou parar lentamente. A Figura 7 mostra um diagrama de blocos do sistema com os demais periféricos.

Figura 7: Sistema pretendido na bancada.



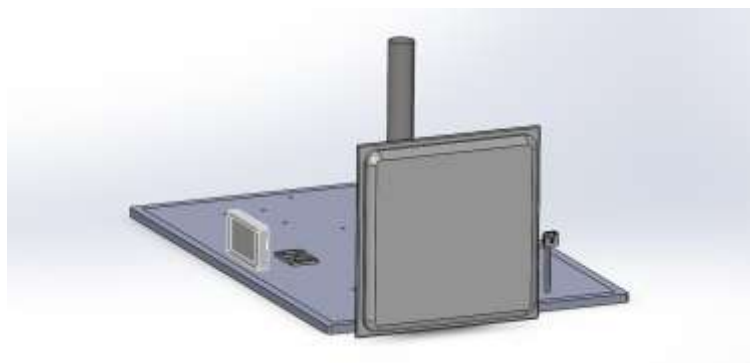
Fonte: Do Autor (2024)

A linguagem escolhida foi o Python, por se tratar de uma linguagem de fácil aprendizado e comum nos trabalhos de Brolese (2023) e Niero (2023). Foram utilizados o formato de bibliotecas para facilitar a comunicação e organização de código para posterior uso. Um display da marca Proculus modelo P48272V50C_T01+ (Victor Vision, São Carlos, Brasil) foi utilizado para apresentar informações como o estado do inversor, sua velocidade em km/h, sua temperatura em tempo real, informações úteis para a bancada tais como a leitura do potenciômetro, da câmera e do RFID. A bancada de teste foi equipada com um sistema de automação nível 2, onde a aceleração, frenagem e manutenção de velocidade constante são executadas.

3.1. DESIGN DA BANCADA DE TESTE

Equipada com os sensores RFID, sistema de visão computacional, display LCD, Raspberry Pi, motor e inversor, a bancada de teste foi desenvolvida no software de desenho e simulações 3D Solidworks®. A Figura 8 detalha o processo inicial da montagem.

Figura 8: Bancada de teste em desenvolvimento.



Fonte: Do Autor (2024)

3.2. TESTES COM OS COMPONENTES

O projeto teve início com o desenvolvimento de programações para testes com cada componente do projeto separado. Uma IDE (do inglês, *Integrated Development Environment*) foi utilizada para desenvolver os programas, no qual possui ferramentas para compilação, depuração e edição do programa. O microcomputador Raspberry Pi 4 (Maker Hero, Florianópolis, Brasil) foi utilizado como plataforma para executar estes testes, visto que possui entradas e saídas

programáveis e comunicação SPI, Serial e um conector para módulo de câmeras, que foram utilizados para comunicação com os módulos utilizados no projeto.

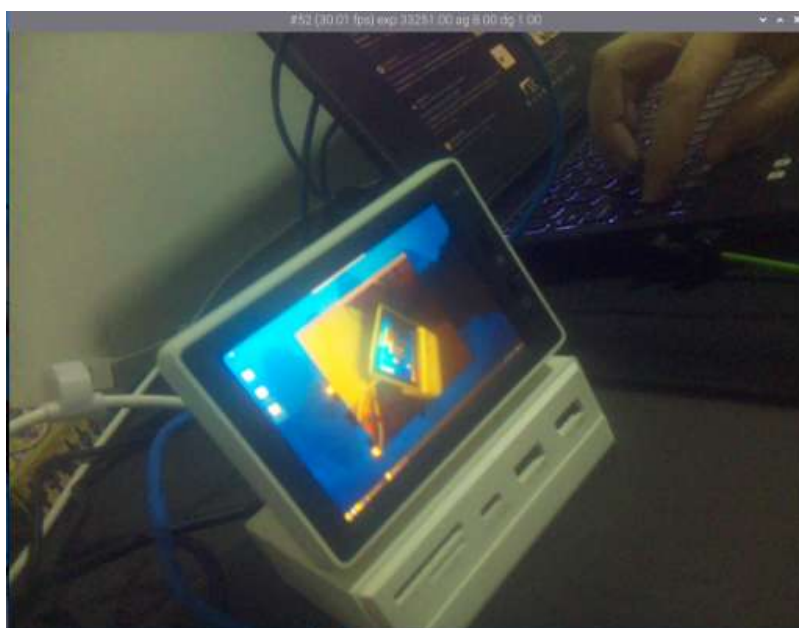
Os testes se fazem necessários para validar o funcionamento de cada módulo e ferramenta que foi utilizado dentro do projeto, além de permitir uma organização e desenvolvimento do código final de maneira modular e simplificada. As bibliotecas auxiliam no desenvolvimento, bem como a depuração do código, a repetibilidade para novos projetos também é favorecida.

3.2.1 Camera Module V2

A câmera (*Maker Hero*, Florianópolis, Brasil) é um módulo vendido separadamente conta com um sensor Sony IMX219 com 8 MP de resolução, sendo possível uma captura de vídeo em 1080p 30 FPS, ou a 720p 60 FPS e resoluções de imagens estáticas de até 3280x2464 (PI FOUNDATION, 2024).

Para teste do módulo de câmera foi utilizado a biblioteca *rpicam-apps*, esta já contida no microcomputador, ela é compatível com a V2 e foi baseada na biblioteca anterior *libcamera*. A Figura 9 possui a imagem estática do teste realizado.

Figura 9: Visão da câmera na janela do programa.



Fonte: Do Autor (2024)

O teste com o módulo de câmera validou a aquisição de imagens estáticas por meio da biblioteca `picam-apps`, utilizando a interface de linha de comando.

3.2.2 Módulo rede CAN

O módulo de rede CAN (*Waveshare*, Shenzhen, China) se faz necessário já que o inversor WEG utilizado conta com essa rede, facilitando o seu uso e reduzindo os cabos no circuito. O Inversor de modelo CVW300A0400D0NB66G2 (WEG, Jaraguá do Sul, Brasil) destinado a tração elétrica para controle de motores de indução trifásicos de baixa tensão em veículos elétricos (WEG, 2019). Um teste foi realizado para estabelecer uma rota de comunicação entre o inversor e o módulo rede CAN. A Figura 10 mostra esse módulo que permite o Raspberry Pi comunicar-se com a rede CAN.

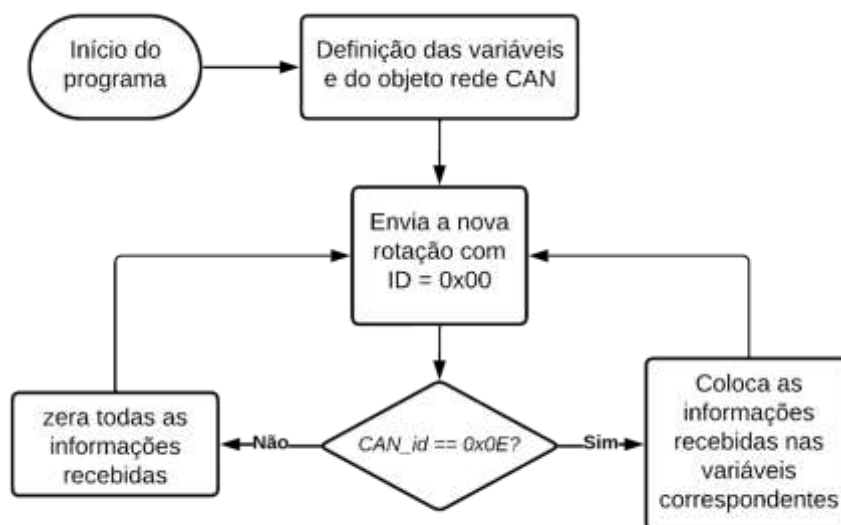
Figura 10: Módulo *RS485 CAN Hat*.



Fonte: Waveshare (2024)

O teste foi iniciado com uma comunicação básica entre dois Raspberry Pi 3 e 4, os pacotes enviados continham uma mensagem remetendo a um ping de informações, enviada para a rede até chegar ao outro Raspberry Pi, permitindo estabelecer uma base para o início do protocolo de comunicação com rede CAN.

Figura 11: Fluxograma da biblioteca da rede CAN.



Fonte: Do Autor (2024)

A Figura 11 ilustra o laço de repetição que a biblioteca de rede CAN exerce quando é feito um envio de uma mensagem e quando se recebe uma mensagem, essas já configuradas em ambos os lados.

3.2.3 Leitor de *Tags* em placas com RFID

O sistema de leitura de *tags* em placas, programação desenvolvida por Niero(2023), escrita em Python, tem como principal objetivo um sistema de identificação e redundância para sinalizações de trânsito.

Seu funcionamento será com etiquetas fixadas em placas e um sensor instalado na bancada para identificar qual a placa que está no leitor. A Figura 12 mostra o modelo de antena utilizado para montagem na bancada. A comunicação com essa antena é por TCP/IP (Ethernet) com o Raspberry Pi(NIERO, 2023).

Figura 12: Antena para leitura das *tags* RFID.



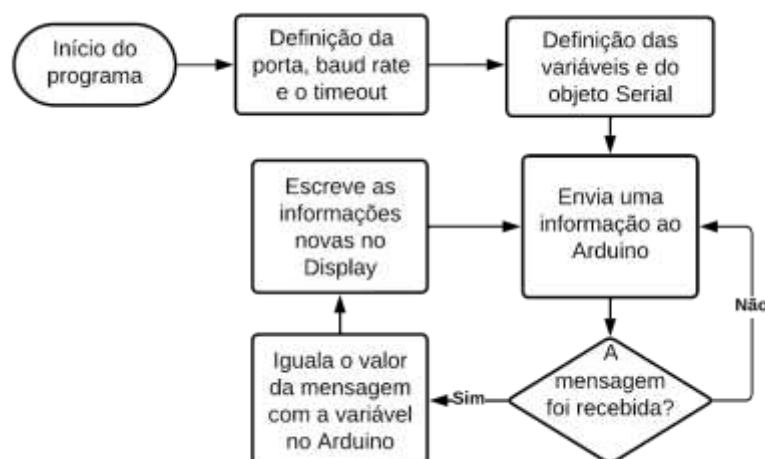
Fonte: Via Onda (2023)

3.2.4 Display LCD

O display é o responsável por exibir informações relevantes ao usuário. Nisso foi escolhido o estado do inversor, velocidade de saída ao inversor em km/h, sua temperatura em tempo real, bem como informações úteis para a bancada tais como a leitura do potenciômetro, da câmera e do RFID.

Visto que o Raspberry Pi 4 não possui entradas analógicas em seus pinos, foi necessário atribuir um circuito com essa função. Optou-se por um circuito integrado ou um microcontrolador para tal tarefa. O modelo de display também possuía uma limitação, não há biblioteca desenvolvida em Python ao Raspberry Pi para permitir uma comunicação direta entre ambos. Foi optado o uso de um microcontrolador ATmega328P, visto que possui leitura analógica em até 6 pinos e contém uma biblioteca desenvolvida pela própria fabricante. A Figura 13 apresenta o fluxograma do programa testado para adquirir os valores de tensão e posição do potenciômetro.

Figura 13: Fluxograma do teste com o display e o conjunto Serial.



Fonte: Do Autor (2024)

De modo a melhorar o sinal do potenciômetro, um filtro de média móvel foi desenvolvido, já para a sua usabilidade no projeto, foi necessário ser adaptado de modo a ter posições de frenagem e aceleração no mesmo equipamento, para isso a sua posição para a esquerda o valor retornado é negativo até -100, já para a direita, seu valor retornado é positivo até 100.

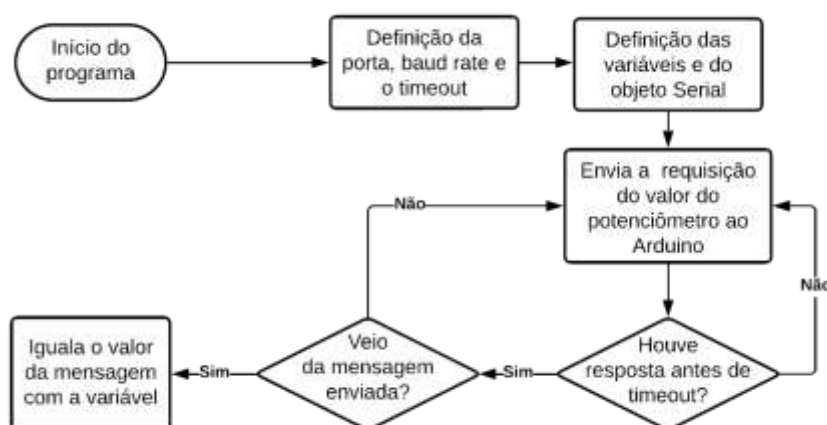
3.3. PROGRAMAÇÃO DO RASPBERRY PI 4

Após realizados os testes com cada equipamento a ser utilizado separadamente, o próximo passo foi a programação completa do projeto. Início com o módulo de câmera e a visão computacional de leitura de sinalização viária, utilizando o código de Brolese (2023), programação essa desenvolvida em Python, que tem como principal objetivo a leitura de placas de sinalização viária. O projeto em questão da visão computacional, emprega redes neurais convolucionais para avaliar a precisão da rede em diferentes e adversas situações reais. O objetivo final é aplicar essa tecnologia em um veículo elétrico, e o sistema desenvolvido em conjunto permitir que ele tome decisões baseadas no aprendizado obtido. As dependências da programação são as bibliotecas *matplotlib*, *os*, *random*, *openpyxl*, *time*, *datetime*, *os*, *tensorflow*, *sklearn* e *opencv-python*, essas podendo ser instaladas utilizando o terminal do Raspberry Pi caso não instaladas nativamente.

3.3.1 Programação do Protocolo Serial

Foi realizado um protocolo serial que requisita informações ao ATmega ou enviar mensagens em um determinado tempo sem necessidade de requisição, apenas confirmando que a mensagem foi recebida com sucesso e chegou completa. O início da função quando esta é chamada se dá com a escolha da porta, taxa de transmissão e o período máximo para chegar uma mensagem. Foi optado por usar 115200 bits por segundo por se tratar de um valor comum em microcontroladores ATmega328P e proporciona uma taxa de transmissão suficiente ao projeto que possui pacotes curtos de informação. A Figura 14 mostra um fluxograma estrutural de como é realizado essa comunicação de um sistema com outro bem como cada passo desde o início da biblioteca.

Figura 14: Fluxograma do Protocolo Serial.



Fonte: Do Autor (2024)

A estrutura do protocolo possui uma função de requisição de dados, que está cadastrada em ambos os lados (neste programa foi implementado a leitura de posição e tensão no sensor de potenciômetro), uma função que faz a recepção da mensagem, interpretando a mensagem e salvando em sua respectiva variável, uma função designada para comparação se essa mensagem é alguma das que estão cadastradas e outra que envia a confirmação ou não dessa mensagem que chegou está correta, esta função é utilizada para responder e finalizar a comunicação entre os dois microcontroladores e permitir uma nova comunicação.



3.3.2 Programação do sistema de decisão

É o principal programa, responsável por executar e tomar decisões. Neste, foi levado em consideração quais são as possibilidades que estarão no sistema bem como suas ações para quando aplicado na bancada adquirir as saídas dos sensores.

Foi estabelecido uma ordem para qual sensor e qual placa ou ação do pedal terá prioridade, para proporcionar segurança as pessoas ao redor do teste e robustez durante os testes realizados, a Tabela 2 detalha essas entradas.

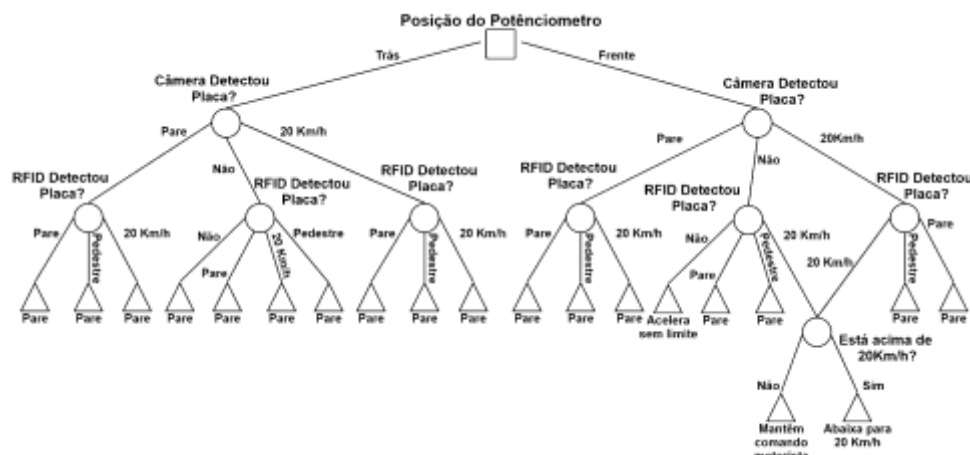
Tabela 2: Prioridade de sinais de entrada no sistema de decisão

Posição	Sinal
1º	PLACA DE PEDESTRE
2º	PLACA DE PARE
3º	SINAL DE FREIO POTENCIÔMETRO
4º	PLACA DE 20 KM/H
5º	SINAL DE ACELERAR POTÊNCIOMETRO
6º	SEM LEITURA

Fonte: Do Autor (2024)

Para os sensores, foram levados em consideração uma prioridade, visto que seus funcionamentos são diferentes um do outro, proporcionando maior ou menor garantia de seu valor retornado. O sensor de potenciômetro foi optado por ser o primeiro, o sensor de RFID o segundo e o sensor de câmera do sistema de visão computacional o terceiro. Na prioridade da sinalização, tem-se a placa de pedestre prioritária para evitar possíveis acidentes, a placa de pare recebe a segunda posição por se tratar de danos materiais, já o sinal de potenciômetro pode ser acionado em qualquer momento do teste, não sendo bloqueado em caso de alguma falha, a placa de 20 Km/h, o sinal de acelerar no potenciômetro e sem leitura nos sensores estão em último lugar para permitir ao usuário acelerar ou dirigir com maior velocidade e segurança. A árvore de decisões ilustrada na Figura 15 funciona juntamente com a Tabela 2, em que cada entrada de sensor irá definir as saídas.

Figura 15: Árvore de tomada de decisão.



Fonte: Do Autor (2024)

Na Figura 16 está ilustrado um fluxograma das saídas com base nas escolhas da árvore de decisão e entrada dos sensores, com essa implementação a saída dessa função pode acionar o motor para frente ou frear o veículo. Para segurança, se nenhuma das condições atender o resultado é parar.

Figura 16: Fluxograma da execução das ações.

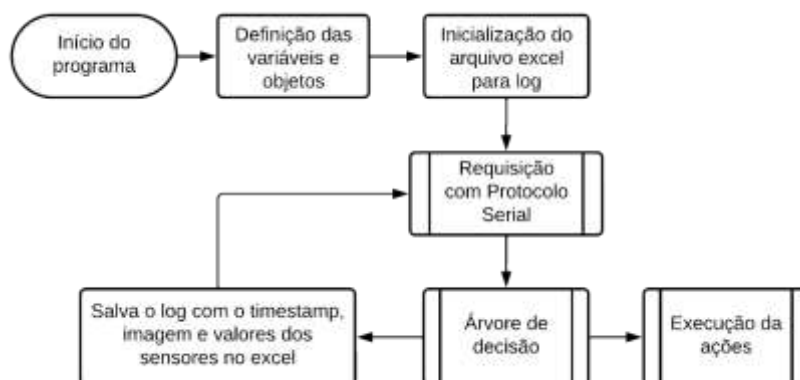


Fonte: Do Autor (2024)

A Figura 17 apresentada a seguir é o fluxograma da programação do sistema de tomada de decisão, que inicia definindo as variáveis a serem utilizadas pelo programa inicialização dos objetos e o arquivo de gravação de dados para análise e geração da matriz de confusão, assim que realizado as requisições de dados e envios de informações ao ATmega328P, o programa executa as ações, detalhado as suas condições na Figura 16, e depois envia os dados para uma planilha de dados.

Para a imagem ser salva dentro de uma célula da planilha é necessário estar com o Excel atualizado para a versão 2410 ou posterior.

Figura 17: Fluxograma da programação do sistema de tomada de decisão.



Fonte: Do Autor (2024)

Aplicando todos os módulos de programas explicados tem-se o programa finalizado e permitindo a gravação dos dados de entrada e saída dos testes.

4. RESULTADOS E DISCUSSÃO

Realizados os testes e programações, foram analisados os resultados e os equipamentos instalados, bem como as simulações realizadas de modo a ser apresentado nessa seção. Os procedimentos descritos nesta seção foram fundamentais para validar a integração e o funcionamento dos módulos experimentais. Com os dados coletados e os sistemas testados, o projeto está preparado para avançar para a etapa de análise e otimização do sistema de decisão.

4.1. BANCADA DE TESTE

A bancada de testes foi montada inserindo os componentes cruciais para permitir o desenvolvimento em sua plataforma. Foram instalados na base os cabos, sensores, Raspberry Pi e o ATmega328P de modo a ficar organizado a distribuição de fios. A Figura 18 mostra o resultado da montagem dessa bancada.

Figura 18: Bancada de teste em rota de finalização.



Fonte: Do Autor (2024)

A bancada de testes pode ser utilizada para desenvolvimento dentro da área de veículos autônomos para leitura de placas de sinalização, comunicação com o sistema de rede CAN e melhorias nos códigos já desenvolvidos.⁶

4.2. PROTOCOLO SERIAL

No protocolo serial desenvolvido teve como resultado a leitura do potenciômetro onde a Tabela 3 mostra o teste realizado com a leitura analógica comprovando que foi válido utilizar um filtro analógico, neste teste foi utilizado um potenciômetro Linear com resistência 10kΩ e um filtro de média móvel com 25 pontos de aquisição de valor e a (1 utilizada para descobrir a porcentagem de melhoria.

$$P.M. = \left(\frac{\text{Erro médio do sinal puro} - \text{Erro médio do sinal filtrado}}{\text{Erro médio do sinal puro}} \right) * 100 \quad (1)$$

⁶ Os códigos utilizados no trabalho estão disponíveis em repositório aberto: <https://github.com/lexcibien/Sistema-tomada-de-decisao>

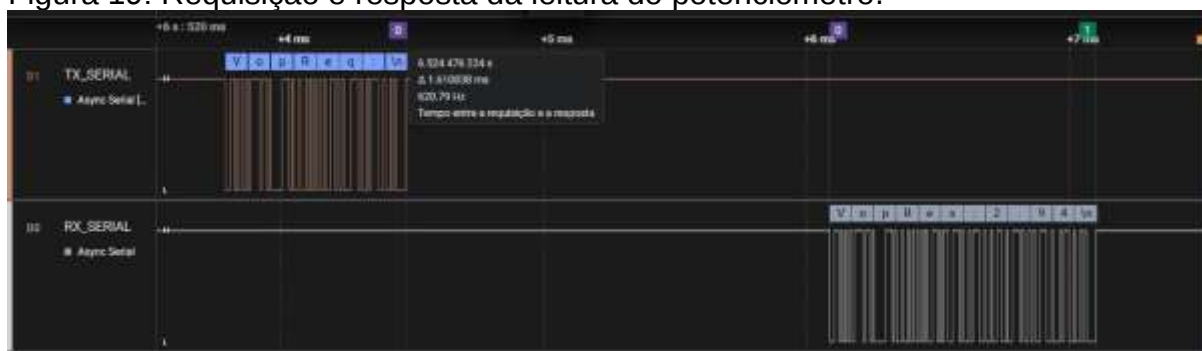
Tabela 3: Melhoria da leitura de tensão com filtro e sem filtro.

Média de erro no sinal sem filtro	0,003 V
Média de erro no sinal com filtro	0,001 V
Porcentagem de melhoria	66,66 %

Fonte: Do Autor (2024)

A Figura 19 é um teste realizado utilizando um analisador de lógica para visualização do protocolo serial realizando a requisição do potenciômetro, o Raspberry Pi fazendo a pergunta ao ATmega328P e este respondendo com o valor requisitado.

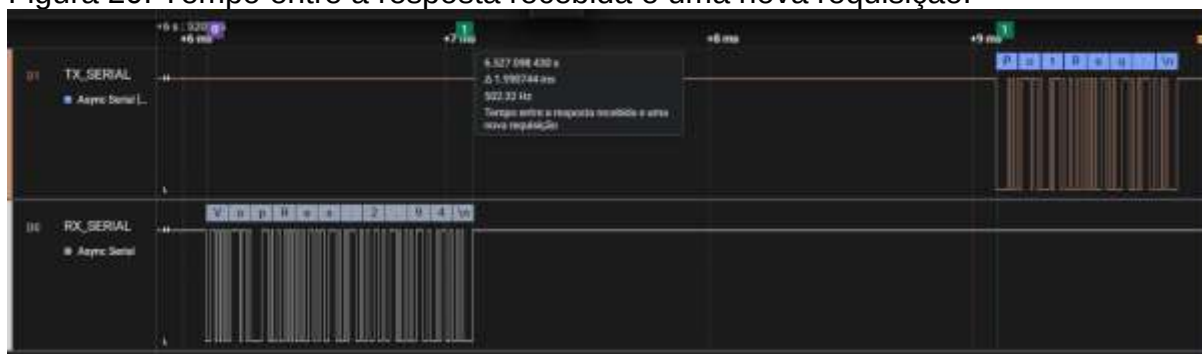
Figura 19: Requisição e resposta da leitura do potenciômetro.



Fonte: Do Autor (2024)

Pode-se notar que o tempo necessário para a requisição do Raspberry Pi ser reconhecida pelo ATmega328P é de 1,6 milissegundos sendo um valor interessante para a aplicação, podendo receber de modo rápido a informação de um para outro. Esse tempo em específico deve-se em decorrência de a programação realizar condições para cada entrada de mensagem.

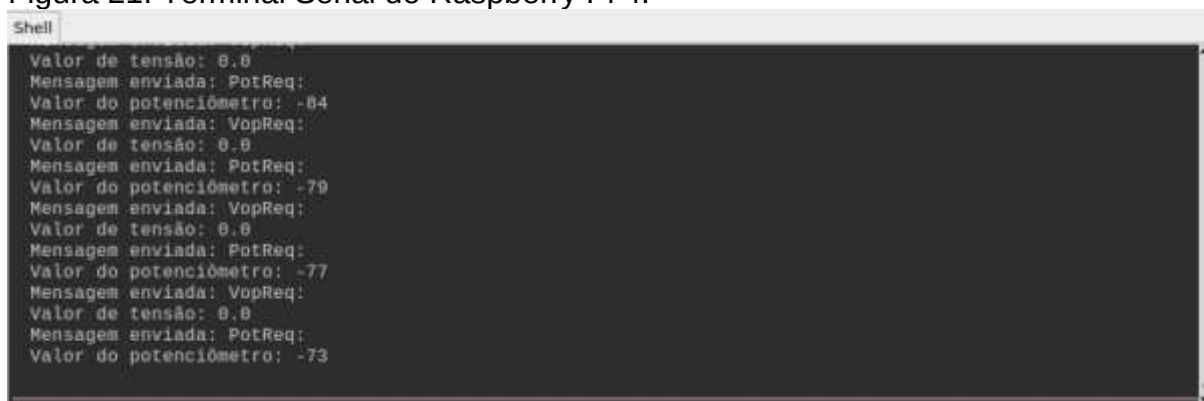
Figura 20: Tempo entre a resposta recebida e uma nova requisição.



Fonte: Do Autor (2024)

A Figura 20 demonstra que é possível realizar uma aquisição rápida de modo a ocupar apenas 2 milissegundos de um tempo para outra, esse tempo deve-se a impressões de dados para visualizar no terminal. A Figura 21 está presente os pedidos que o Raspberry Pi 4 realiza ao ATmega328P. A mensagem é composta por uma *string* “PotReq:” e depois o outro lado envia uma resposta “PotRes: -77” onde possui a *string* de conferência e o valor desse sensor requisitado.

Figura 21: Terminal Serial do Raspberry Pi 4.



```

Shell
Valor de tensão: 0.8
Mensagem enviada: PotReq:
Valor do potenciômetro: -84
Mensagem enviada: VopReq:
Valor de tensão: 0.8
Mensagem enviada: PotReq:
Valor do potenciômetro: -79
Mensagem enviada: VopReq:
Valor de tensão: 0.8
Mensagem enviada: PotReq:
Valor do potenciômetro: -77
Mensagem enviada: VopReq:
Valor de tensão: 0.8
Mensagem enviada: PotReq:
Valor do potenciômetro: -73
  
```

Fonte: Do Autor (2024)

4.3. SIMULAÇÃO DO SISTEMA DE TOMADA DE DECISÃO

Tabela 4: Matriz de confusão da simulação das entradas e saídas do sistema.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2779	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	2678	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	2670	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	2811	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	2742	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	2771	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	2789	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	2836	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	2822	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	2745	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	5415	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	5697	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	5554	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	5509	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5479	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5607	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5599	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5789	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5572	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5573
Total de resultados da simulação não encontrados no sistema de tomada de decisão: 16563																				

Fonte: Do Autor (2024)

A Tabela 4 é a matriz de confusão proveniente das simulações realizadas com o método de Monte Carlo e a programação do sistema de tomada de decisão.

Figura 22: Casos de cada índice no sistema de tomada de decisão.

1	Potenciômetro = Trás , Câmera = Pare , RFID = Pare
2	Potenciômetro = Trás , Câmera = Pare , RFID = Pedestre
3	Potenciômetro = Trás , Câmera = Pare , RFID = 20 Km/h
4	Potenciômetro = Trás , Câmera = Não detectou , RFID = Não detectou
5	Potenciômetro = Trás , Câmera = Não detectou , RFID = Pare
6	Potenciômetro = Trás , Câmera = Não detectou , RFID = 20 Km/h
7	Potenciômetro = Trás , Câmera = Não detectou , RFID = Pedestre
8	Potenciômetro = Trás , Câmera = 20 Km/h , RFID = Pare
9	Potenciômetro = Trás , Câmera = 20 Km/h , RFID = Pedestre
10	Potenciômetro = Trás , Câmera = 20 Km/h , RFID = 20 Km/h
11	Potenciômetro = Frente , Câmera = Pare , RFID = Pare
12	Potenciômetro = Frente , Câmera = Pare , RFID = Pedestre
13	Potenciômetro = Frente , Câmera = Pare , RFID = 20 Km/h
14	Potenciômetro = Frente , Câmera = Não detectou , RFID = Não detectou
15	Potenciômetro = Frente , Câmera = Não detectou , RFID = Pare
16	Potenciômetro = Frente , Câmera = Não detectou , RFID = Pedestre
17	Potenciômetro = Frente , Câmera = Não detectou , RFID = 20 Km/h
18	Potenciômetro = Frente , Câmera = 20 Km/h , RFID = 20 Km/h
19	Potenciômetro = Frente , Câmera = 20 Km/h , RFID = Pedestre
20	Potenciômetro = Frente , Câmera = 20 Km/h , RFID = Pare

Fonte: Do Autor (2024)

A simulação de Monte Carlo foi realizada utilizando 100.000 iterações com números que simulam as entradas dos sensores e a saída esperada. Dentro dessas iterações 83.437 foram resultados contidos no sistema de decisão com 83,43% e 16.563 resultados não encontrados com 16,56%.

Com base nos resultados tem-se uma concentração nos valores de índices maiores, isso pode ser em decorrência das que algumas condições não terem sido consideradas. Foi gerado também os resultados onde não foram possíveis encaixar na matriz de confusão, certas condições possíveis onde o RFID poderia passar por uma placa e não detectar e o sistema de visão computacional conseguir detectar detalhe esse que foi ignorado durante o desenvolvimento da árvore de decisão.

5. CONCLUSÃO

Neste trabalho foi desenvolvido e simulado um sistema de tomada de decisão para sensores de detecção de placas de sinalização viária. Também foi desenvolvido e testado um protocolo serial para a leitura do sensor de pedal de aceleração e frenagem e aferição do seu valor, desenvolvida uma programação para



comunicação com o display LCD Victor Vision, desenvolvido uma programação para comunicação entre um inversor WEG e o Raspberry Pi, desenvolvimento e montagem de uma bancada de testes para desenvolver pesquisas sobre veículos autônomos.

Os resultados foram satisfatórios com mais de 83% de resultados assertivos com a simulação do sistema de tomada de decisão, uma precisão de leitura do sensor de posição do pedal melhorada em aproximadamente 70%, a aquisição do seu valor pelo protocolo serial entre um ATmega328P e o Raspberry Pi. Porém para melhorar ainda mais a segurança é necessário abranger mais condições de placas entre os sensores com ênfase para situações em que o sensor de RFID pode não detectar uma placa e a visão computacional sim.

A inclusão de novos casos no sistema de decisão visa cobrir situações mais complexas, como placas parcialmente obstruídas ou ambientes de baixa luminosidade. Já os testes com a rede CAN buscam validar a comunicação entre diferentes módulos do sistema, garantindo maior estabilidade e segurança operacional.

Para trabalhos futuros a inclusão de mais casos dentro do sistema de tomada de decisão de modo a cobrir todas as situações possíveis, a realização de testes com o inversor WEG e validações com a rede CAN, avaliando sua robustez, o teste com o display LCD para exibir as informações importantes ao usuário, o teste das programações utilizando a bancada de teste como plataforma para averiguações de robustez do sistema e dos sensores e o desenvolvimento de mais opções de controle junto com a bancada de teste.

REFERÊNCIAS

ABVE DATA (São Paulo). Associação Brasileira do Veículo Elétrico. **Eletrificados Crescem 145% No Trimestre**. 2024. Disponível em: <https://abve.org.br/eletrificados-crescem-145-no-trimestre-e-tem-segundo-melhor-mes-da-historia/>. Acesso em: 20 abr. 2024.

BARCELLOS, Mariana. **Acidentes em rodovias estaduais de Santa Catarina aumentam 46% no Carnaval 2024**. Número de mortes também subiu de duas em 2023 para oito este ano. 2024. Disponível em: <https://www.nsctotal.com.br/noticias/acidentes-em-rodovias-estaduais-de-santa-catarina-aumentam-46-no-carnaval-2024>. Acesso em: 21 abr. 2024.

BORGES, Rafaela. **Carro autônomo**. Disponível em: <https://www.uol.com.br/carros/reportagens-especiais/transporte-do-futuro---carro-autonomo/>. Acesso em: 09 maio



2024.

BOSCH, Robert. **CAN Specification V2.0**. Stuttgart: Robert Bosch GmbH, 1991.

BROLESE, Bruno Scremin. **RECONHECIMENTO DE SINALIZAÇÃO VIÁRIA ATRAVÉS DE VISÃO COMPUTACIONAL E REDES NEURAIS NO AUXÍLIO DA MOBILIDADE ELÉTRICA EM VEÍCULOS ELÉTRICOS E AUTÔNOMOS**. 2023. 37 f. TCC (Graduação) - Curso de Engenharia Mecatrônica, Centro Universitário, UNISATC, Criciúma, 2023.

CIRIACO, Douglas. **Como funciona a RFID?** 2009. Disponível em: <https://www.tecmundo.com.br/tendencias/2601-como-funciona-a-rfid-.htm>. Acesso em: 01 jun. 2024.

FERNEDA, Edberto. Redes neurais e sua aplicação em sistemas de recuperação de informação. **Ciência da Informação**, [S.L.], v. 35, n. 1, p. 25-30, abr. 2006. IBICT. <http://dx.doi.org/10.1590/s0100-19652006000100003>.

FOUNDATION, Raspberry Pi. **Raspberry Pi Camera Module 2**. 2024. Disponível em: <https://www.raspberrypi.com/products/camera-module-v2/>. Acesso em: 01 ago. 2024.

MORAIS, Daniel Cardoso de. **Controlador CAN em VHDL**. 2012. 54 f. TCC (Graduação) - Curso de Engenharia Elétrica, Universidade Federal de Campina Grande, Campina Grande, 2012.

NASCIMENTO, Livia Cristina Silva do. **ANÁLISE DE REDES VEICULARES MODELADAS COMO REDES TEMPORAIS E AGREGADAS UTILIZANDO A TEORIA DA INFORMAÇÃO**. 2023. 23 f. TCC (Graduação) - Curso de Ciência da Computação, Universidade Federal de Alagoas, Arapiraca, 2023.

NIERO, Edmar dal Molin. **USO ESTRATÉGICO DE IDENTIFICADORES POR RÁDIO FREQUÊNCIA EM SISTEMAS DE MOBILIDADE AUTÔNOMA: MELHORANDO A SEGURANÇA E EFICIÊNCIA DOS VEÍCULOS AUTÔNOMOS**. 2023. 17 f. TCC (Graduação) - Curso de Engenharia Mecatrônica, Centro Universitário, UNISATC, Criciúma, 2024.

HUGHES, Austin; DRURY, Bill. **Electric Motors and Drives: fundamentals, types and applications**. 5. ed. Oxford, Cambridge: Newnes, 2019. 511 p.

REUTERS (San Francisco). California approves Waymo robotaxi services in LA, SF neighboring cities. 2024. Disponível em: <https://www.reuters.com/business/autos-transportation/california-approves-waymo-robotaxi-services-la-sf-neighboring-cities-2024-03-01/>. Acesso em: 27 abr. 2024.

RUBINSTEIN, Reuven Y.; KROESE, Dirk P. **Simulation and the Monte Carlo Method**. 2. ed. Nova Iorque: John Wiley & Sons, 2008. 345 p. (Wiley Series in Probability and Statistics).



SAE BRASIL. **Carros autônomos: serão eles o novo “padrão” da indústria?** 2024. Disponível em: <https://saebrasil.org.br/noticias/carros-autonomos/>. Acesso em: 10 maio 2024.

SIVARAMAN, Sayanan; TRIVEDI, Mohan Manubhai. Looking at Vehicles on the Road: a survey of vision-based vehicle detection, tracking, and behavior analysis. IEEE Transactions on Intelligent Transportation Systems, [S.L.], v. 14, n. 4, p. 1773-1795, dez. 2013. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/tits.2013.2266661>.

SUTTON, Richard S.; BARTO, Andrew G. **Reinforcement Learning: An Introduction**. 2. ed. Cambridge, Londres: The Mit Press, 2014. 352 p.

TEKTRONIX (EUA). **EV Traction Inverter & Motor Measurements and Analysis**. 2024. Disponível em: <https://www.tek.com/en/solutions/industry/automotive-test-solutions/ev-traction-inverters-and-motors>. Acesso em: 20 abr. 2024.

THOMAS NOLTE, HANS HANSSON, LUCIA LO BELLO, 10., 2006, Catania, Itália. **Automotive communications-past, current and future**. Västerås, Suíça; Catania, Itália: Department Of Computer Engineering And Telecommunications, 2005.

WEG (Brasil). **CVW300 - INVERSOR DE FREQUÊNCIA PARA TRAÇÃO ELÉTRICA**. 3. ed. Jaraguá do Sul: Weg, 2019. 8 p. Disponível em: <https://static.weg.net/medias/downloadcenter/h19/hd1/WEG-CVW300-50041420-catalogo-pt.pdf>. Acesso em: 21 jun. 2024.