



**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS  
UNIDADE ARAXÁ**

**RUBENS MORENO ALVES DE OLIVEIRA**

**SISTEMA EMBARCADO PARA MONITORAMENTO  
AUTOMOTIVO EM TEMPO REAL**

Araxá, MG

2022

**RUBENS MORENO ALVES DE OLIVEIRA**

**SISTEMA EMBARCADO PARA MONITORAMENTO  
AUTOMOTIVO EM TEMPO REAL**

Trabalho de Conclusão de Curso apresentado ao Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá, como requisito parcial para obtenção do grau de Bacharel em Engenharia de Automação Industrial.

Orientador: Prof. Dr. Leandro R. Mattioli

**ARAXÁ-MG**

**2022**

**RUBENS MORENO ALVES DE OLIVEIRA**

**SISTEMA EMBARCADO PARA MONITORAMENTO  
AUTOMOTIVO EM TEMPO REAL**

Trabalho de Conclusão de Curso apresentado ao  
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá,  
como requisito parcial para obtenção do grau de  
Bacharel em Engenharia de Automação Industrial.

Araxá, 18 de julho de 2022

**BANCA AVALIADORA**

---

Leandro Resende Mattioli, Dr.

Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

---

Domingos Sávio de Resende, Dr.

Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

---

Kleber Lopes Fontoura, Dr.

Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

**ARAXÁ-MG**

**2022**

## **AGRADECIMENTOS**

Primeiramente a Deus por ter me dado saúde e força para superar as dificuldades.

A minha mãe, por sempre me guiar com o coração quando priorizei a razão.

A meu pai, por sempre me guiar com a razão quando priorizei o coração.

Ao meu amor, Carolayne, pelo apoio dado nos dias ruins, pelo impulso dado nos dias bons.

Ao meu orientador Leandro, pela sua dedicação, paciência e disponibilidade em me orientar, mas, principalmente, pelas palavras amigas e conselhos.

Aos amigos, que sempre estiveram ao meu lado, pela amizade incondicional e pelo apoio técnico e mental demonstrado ao longo de todo o período de tempo em que me dediquei a este trabalho.

## RESUMO

A integração e o armazenamento de dados de diferentes fontes podem ser observados em tecnologias como Internet das Coisas, comunicação *machine-to-machine* (M2M) e computação em nuvem, promovendo a integração entre sistemas físicos e virtuais. Ainda assim, observa-se uma ociosidade da integração automobilística nos sistemas que usam dessas tecnologias. De fato, um grande problema enfrentado em frotas automotivas, sendo particulares ou empresariais, é a coleta de dados de bordo de forma preventiva e mais minuciosa. Vários aspectos devem ser observados em relação às variáveis de processos relevantes, que influenciam no custo, na segurança e na manutenção. Este trabalho trata do desenvolvimento de um sistema embarcado para a telemetria e a detecção de frenagem brusca em automóveis. A solução é baseada no microcontrolador ESP32, no computador de placa única BeagleBone Black e na rede MQTT, popularizada pelas aplicações IoT. São coletados dados do veículo por meio de um scanner automotivo OBDII, que são complementados com informações provenientes do módulo de aceleração MPU 6050 e do aplicativo OwnTracker, para a geolocalização. O armazenamento é feito em um banco não-relacional RavenDB. O sistema tem o objetivo de adquirir dados como: consumo de combustível, localização do veículo, velocidade instantânea, estado do motor, temperatura do motor e eventos de frenagem brusca, além de exibi-los em uma interface gráfica de acompanhamento em tempo real e armazená-los local e remotamente. O sistema foi testado em 2 veículos distintos: um para avaliar o desempenho do sistema de telemetria nas redes móveis 3G e 4G, tanto em ambiente urbano como em rodovia, e outro para os cenários de frenagens bruscas. Os resultados se mostraram satisfatórios, em termos de armazenamento de dados de forma remota, em caso de conectividade com a internet, e armazenamento de dados de forma local, prevenindo a perda de dados no caso contrário. No que diz respeito às frenagens bruscas, os testes atenderam às expectativas de captura de imagens conclusivas, usadas no cenário pós-ocorrência para fins de diagnóstico.

**Palavras-chave:** OBD2. Internet das Coisas. Telemetria. Análise Pós-Ocorrência.

## ABSTRACT

The integration and storage of data from different sources can be observed in technologies such as the Internet of Things, machine-to-machine (M2M) communication, and cloud computing, promoting the integration between physical and virtual systems. Even so, there is some idleness in automotive integration with systems that use such technologies. A major problem faced in automotive fleets, whether private or corporate, is the collection of onboard data in a preventive and more detailed way. Several aspects must be observed about the relevant process variables, which influence cost, safety, and maintenance. This work deals with developing an embedded system for telemetry and sudden braking detection in automobiles. The solution is based on the ESP32 microcontroller, the BeagleBone Black single-board computer, and the MQTT network, popularized by IoT applications. Vehicle data is collected through an automotive OBDII scanner, which is complemented with information from the MPU 6050 acceleration module and the OwnTracker application, the latter for geolocation. Storage is done in a non-relational RavenDB database. The system aims to acquire data such as fuel consumption, vehicle location, instantaneous speed, engine status, engine temperature, and sudden braking events, in addition to displaying them in a graphical interface for real-time monitoring and storing them locally and remotely. The system was tested in 2 different vehicles: one to evaluate the telemetry system's performance in 3G and 4G mobile networks, both in urban environments and on highways, and another for the sudden braking scenarios. The results were satisfactory, in terms of remote data storage, whenever good internet connectivity is present, and remote storage locally, preventing data loss otherwise. Concerning hard braking, the tests met the expectations of capturing conclusive images, used in post-occurrence scenarios for diagnostic purposes.

**Palavras-chave:** OBD2. Internet of Things. Telemetry. Post-occurrence analysis.

## LISTA DE FIGURAS

Figura 1 – Conector padrão OBD2.....	18
Figura 2 – Esquema dos pinos.....	19
Figura 3 – Estrutura de publicação e inscrição do MQTT .....	22
Figura 4 – GPS sem imagem de trânsito e GPS com imagem de trânsito.....	24
Figura 5 – Placa de desenvolvimento com ESP32 .....	27
Figura 6 – Sistema embarcado de baixo nível.....	28
Figura 7 – Scanner genérico ODBII .....	29
Figura 8 – Módulo GPS Neo-6m.....	31
Figura 9 – Módulo MPU-6050 .....	32
Figura 10 – Integração sistemas embarcados alto nível-baixo nível.....	33
Figura 11 - Estrutura de comunicação com broker local.....	35
Figura 12 - Estrutura de comunicação com broker remoto .....	36
Figura 13 – Console do navegador.....	38
Figura 14 – Interface gráfica do cliente RavenDB .....	39
Figura 15 – Percurso urbano.....	40
Figura 16 – Percurso rodoviário .....	41
Figura 17 – Distribuição do armazenamento (local-remoto) nas redes 3G e 4G .....	42
Figura 18 – Sinalizador de ponto de frenagem.....	43
Figura 19 – Teste de frenagem brusca a 70 km/h.....	43
Figura 20 – Teste de frenagem brusca a 50 km/h.....	44
Figura 21 – Teste de frenagem brusca a 30 km/h.....	44
Figura 22 – Imagem sem captura de sinalizador .....	45
Figura 23 – Percurso de teste com dados reais .....	46
Figura 24 – Coordenadas armazenadas no percurso .....	47
Figura 25 – Gráfico geral pausado .....	47



## **LISTA DE TABELAS E QUADROS**

Quadro 1 – Descrição dos Pinos do conector OBD-II.....	19
Quadro 2 – Possíveis fontes de dados do sistema .....	25
Tabela 1 - Intervalo de aquisição de mensagens .....	38
Tabela 2 – Relação de imagens capturadas .....	45

## LISTA DE SIGLAS E ABREVIACÕES

CONAMA	Conselho nacional do meio ambiente
DTC	<i>Diagnostic Trouble Code</i> (Código de problema de diagnóstico)
ECU	<i>Electronic Control Unit</i> (Unidade de Controle Eletrônico)
IoT	<i>Internet of Things</i> (Internet das coisas)
ISO	<i>International Organization for Standardization</i> (Organização Internacional de Normalização)
MIL	<i>Malfunction Indicator Light</i> (Luz indicadora de mau funcionamento)
MQTT	<i>Message Queue Telemetry Transport</i> (Transporte de Filas de Mensagem de Telemetria)
OPAS	Organização Pan-Americana da Saúde
OBD	<i>On Board Diagnostics</i> (Diagnóstico de bordo)
PID	<i>Parameter Identifier</i> (Identificador de parâmetro)
SAE	<i>Society of Automotive Engineers</i> (Sociedade dos Engenheiros Automotivos)
VIN	<i>Vehicle identification number</i> (Número de Identificação do Veículo)
3G	Terceira geração de rede móveis
4G	Quarta geração de rede móveis

## SUMÁRIO

1. INTRODUÇÃO .....	13
2. REVISÃO TEÓRICA .....	15
2.1 FRENAGEM BRUSCA.....	15
2.2 OBD.....	16
2.2.1 OBD II.....	18
2.2.2 PROTOCOLOS .....	18
2.2.3 PADRÃO DE COMANDOS.....	20
2.3 MANUTENÇÃO .....	21
2.4 MEDIÇÕES EM TEMPO REAL .....	21
2.5 MQTT .....	22
2.6 RECURSOS DE IMAGEM.....	23
3. MATERIAIS E MÉTODOS .....	25
3.1 LEVANTAMENTO DAS NECESSIDADES DO SISTEMA .....	25
3.1.1 SELEÇÃO DE VARIÁVEIS DE TEMPO REAL .....	26
3.1.2 SELEÇÃO DE VARIÁVEIS HISTÓRICAS .....	26
3.2 DESENVOLVIMENTO DE SISTEMA EMBARCADO .....	26
3.2.1 SISTEMA EMBARCADO DE BAIXO NÍVEL.....	27
3.2.2 ELM 327.....	28
3.2.3 ESTIMAÇÃO DE CONSUMO DE COMBUSTÍVEL.....	29
3.2.4 MÓDULO GPS NEO-6M .....	30
3.2.5 MÓDULO MPU - 6050.....	31
3.2.7 DESENVOLVIMENTO DA APLICAÇÃO IOT .....	33
3.3 AVALIAÇÃO DE DESEMPENHO DO SISTEMA EMBARCADO .....	34
3.4 TRANSMISSÃO DE DADOS .....	34

3.5	DESENVOLVIMENTO DA INTERFACE GRÁFICA COM OS USUÁRIOS ...	36
4.	RESULTADOS.....	37
4.1	VALIDAÇÃO DE REDE 3G E 4G.....	37
4.1.1	TELEMETRIA.....	37
4.1.2	ARMAZENAMENTO DE DADOS.....	39
4.1.3	TESTE URBANO.....	40
4.1.4	TESTE RODOVIÁRIO.....	41
4.2	IDENTIFICAÇÃO DE FRENAGENS BRUSCAS.....	42
4.3	AQUISIÇÃO DE DADOS REAIS.....	46
5.	CONCLUSÕES.....	49
6.	REFERÊNCIAS.....	50

## 1. INTRODUÇÃO

Desde a consolidação das redes globais de computadores, houve uma constante conexão e aproximação mundial no que diz respeito ao transporte de dados em tempo real. Diversos setores da sociedade vêm aproveitando essa conexão e disseminação de dados para melhoria da sociedade, potencialização de empreendimentos, e constante globalização.

No contexto mais atual, essa interconexão pode ser observada em tecnologias como: Internet das coisas (IoT) (RUENGITTINUN, et al., 2017), comunicação *machine to machine* (M2M) (RUENGITTINUN, et al., 2017), Radio Frequency Identification (RFID) (HANS, 2017) e computação em nuvem (HANS, 2017). Todas elas podem ser utilizadas para promover a integração entre sistemas físicos e virtuais (HANS, 2017), a fim de beneficiar diversos setores da sociedade, com diferentes custos e níveis de confiabilidade.

Ainda assim, um grande problema enfrentado em frotas automotivas, sendo particulares ou empresariais, é a coleta de dados de bordo de forma preventiva e mais minuciosa. No que concerne ao armazenamento de dados em meios de transporte, uma forte referência são as caixas pretas dos aviões. Segundo Gorlick (1991), tais aparelhos destinam-se ao registro de mensagens enviadas e recebidas da torre, além de informações sobre as condições do avião, conversas dentro da cabine e um histórico de valores de certas grandezas como velocidade, aceleração, altitude e potência.

É notável a necessidade de estudo de sistemas embarcados que possam proporcionar proteção a atitudes inesperadas dos condutores de automóveis (RUENGITTINUN et al., 2017). Por outra perspectiva, Gupta et al. (2008) denotam como a obtenção de dados de consumo de combustível em tempo real pode impactar a relação custo-benefício de uma frota.

Contudo, a referência de armazenamento de dados, as caixas pretas dos aviões, demandam robustez e possuem alto custo. Diante desse cenário, esse trabalho tem como intuito desenvolver e avaliar um sistema embarcado de monitoramento automotivo que possa captar dados e eventos relevantes de frotas automotivas particulares ou empresariais, com armazenamento físico e em nuvem, propiciando uma maior confiabilidade, em detrimento de seu custo. Por ser um sistema embarcado externo à ECU do veículo, o sistema em questão é genérico o suficiente para poder ser aplicado em outros tipos de veículos e montadoras.

O combustível é insumo primário de frotas automotivas, e a medição e previsão de seu consumo é de extrema importância para a relação custo-benefício das operações. Além desse dado de caráter financeiro, a manipulação de frotas tem relação direta com riscos que podem envolver perdas materiais e humanas. O método apresentado neste trabalho será um modelo para auxiliar na obtenção de dados em frotas, possibilitando ao usuário um maior controle de gastos, consumo de insumos, manutenção e segurança por meio da telemetria do veículo.

O objetivo geral deste trabalho é desenvolver e avaliar o potencial de um sistema embarcado que colete informações sobre consumo de combustível, localização do veículo, velocidade instantânea e eventos de frenagem brusca por meio da aceleração, usando esses dados para proporcionar segurança e economia no controle de frota. Para tanto, o trabalho deve conceber um protótipo de nível funcional que proporcione a captura de dados relevantes do veículo em tempo real e armazenamento, com arquitetura e suporte compatíveis com as necessidades. Além disso, deve haver a implantação do sistema eletrônico em um veículo automotor para validação de sua capacidade operacional e também prover soluções específicas para os casos de baixa cobertura da rede móvel.

## 2. REVISÃO TEÓRICA

No contexto do controle de veículos automotores, vários aspectos devem ser observados em relação às variáveis de processos relevantes. Esses valores são frequentemente ignorados pelo usuário final, sendo usados apenas pela unidade de controle do motor (ECU - *engine control unit*) do veículo, ou, em alguns casos, nem mesmo por ela. Logo, a percepção e o processamento das informações advindas dessas variáveis são extremamente importantes para que seja possível identificar e prever necessidades, definir parâmetros e estabelecer medidas para o alcance de objetivos, no que diz respeito à manutenção, segurança, conforto e redução de custos em frotas de transporte terrestre, atuação agrícola e equivalentes.

### 2.1 FRENAGEM BRUSCA

Segundo a OMS, os acidentes de trânsito devem causar mais de 13 milhões de mortes e deixar 500 milhões de pessoas lesionadas durante a próxima década, especialmente em países de baixa e média renda (OPAS, 2021).

Conforme o artigo 42 do Código de Trânsito Brasileiro, “Nenhum condutor deverá frear bruscamente seu veículo, salvo por razões de segurança.” (CÓDIGO DE TRÂNSITO BRASILEIRO, 1997). Isso se associa ao artigo 29, inciso II : “o condutor deverá guardar distância de segurança lateral e frontal entre o seu e os demais veículos, bem como em relação ao bordo da pista, considerando-se, no momento, a velocidade e as condições do local, da circulação, do veículo e as condições climáticas;” (CÓDIGO DE TRÂNSITO BRASILEIRO, 1997), e também ao artigo 28 “O condutor deverá, a todo momento, ter domínio de seu veículo, dirigindo-o com atenção e cuidados indispensáveis à segurança do trânsito.” (CÓDIGO DE TRÂNSITO BRASILEIRO, 1997). Assim, a frenagem brusca pode acontecer principalmente pela falta de atenção, pela falta de previsão, antecipação, excesso de confiança, avaliações erradas do condutor, negligência, imprudência e imperícia.

Segundo Mao et al. (2021), é notável que eventos de acentuada variação de força  $g$  ( $9,80665 \text{ m/s}^2$ ) podem presumir manobra de frenagem forte, desvio e aceleração brusca. Além disso, esses eventos são frequentes e conclusivos no que diz respeito à identificação de colisões e também no comportamento de motoristas, se comparados a outros eventos que o veículo pode expor, como excesso de velocidade. Corroborando com isso, Simons-Morton et al (2013) concluem que a obtenção dos eventos de frenagem brusca pode ser considerada uma ferramenta preditiva de identificação de risco de acidentes.

Quando um carro está em movimento, os passageiros se encontram com a mesma velocidade que ele. Em uma frenagem os passageiros tendem a manter o estado de movimento sendo projetados para frente, porém isso pode ocorrer de forma suave, ou brusca, conforme a variação de força gravitacional. Diante disso, conforme Simons-Morton et al (2013), os eventos de força  $g$  são considerados elevados quando excedem os seguintes limites: desaceleração longitudinal/frenagem forte ( $\leq -0,45g$ ), aceleração longitudinal/arranques rápidos ( $\geq 0,35g$ ), giro lateral negativo/esquerda ( $\leq -0,50g$ ) e giro lateral positivo/direita ( $\geq 0,50g$ ). Já Liu et al. (2022), de forma empírica, obtiveram valores aproximados, no que diz respeito à desaceleração longitudinal de  $-5m/s^2$  ou  $-0.509858g$ .

## 2.2 OBD

*On-board diagnostics* (OBD) é o sistema de autodiagnóstico que está presente na maioria das viaturas automóveis este permite obter relatórios sobre o estado dos vários componentes do mesmo (RIBEIRO, 2015). No Brasil, segundo a Resolução CONAMA (Conselho Nacional do Meio Ambiente) Nº 315/2002, artigo 10 “Fica estabelecido o porte de dispositivos/sistemas para auto diagnose (OBD), das funções de gerenciamento do motor que exerçam influência sobre as emissões de poluentes do ar, para todos os veículos leves de passageiros e veículos leves comerciais”. Segundo Ribeiro (2015) as primeiras versões deste sistema de autodiagnóstico tinham pouca capacidade de fornecimento de informação acerca do estado do veículo, apenas reportavam a existência de um problema e não informação detalhada do mesmo, como por exemplo o acender de uma luz CHECK ENGINE no painel do veículo caso algum problema fosse detectado no motor.

Com a modernização do sistema de autodiagnóstico surgiu a padronização OBD-II por meio especificamente, da interface OBD-II, que permite ampla análise de variáveis do veículo, e também acesso a rede de diagnósticos. Diante disso, a legislação brasileira complementa a resolução anterior com a abrangência de variáveis do padrão OBD-II, conforme a Resolução CONAMA 354/2004 , que foi dividida em duas fases de implantação, sendo a primeira chamada, OBDBr-1, que contém as seguintes variáveis a serem contempladas:

- a) sensor pressão absoluta ou fluxo de ar;
- b) sensor posição da borboleta;
- c) sensor de temperatura de arrefecimento;

- d) sensor temperatura de ar;
- e) sensor oxigênio (somente o sensor pré-catalisador);
- f) sensor de velocidade do veículo;
- g) sensor de posição do eixo comando de válvulas;
- h) sensor de posição do virabrequim;
- i) sistemas de recirculação dos gases de escape (EGR);
- j) sensor para detecção de detonação;
- l) válvulas injetoras;
- m) sistema de ignição;
- n) módulo controle eletrônico do motor;
- o) lâmpada indicadora de mau funcionamento (LIM);
- p) outros componentes que o fabricante julgue relevantes para a correta avaliação do funcionamento do veículo e controle de emissões de poluentes.

Já o sistema OBDBr-2 deve detectar e registrar a existência de falhas de combustão, deterioração do(s) sensor(es) de oxigênio primário(s) e eficiência de conversão do catalisador que acarretem aumento de emissões, bem como apresentar características mínimas para a detecção de falhas nos seguintes componentes, quando aplicável:

- a) sensores de oxigênio (pré e pós-catalisador);
- b) válvula de controle da purga do cânister; e
- c) outros componentes que o fabricante julgue relevantes para a correta avaliação do funcionamento do veículo e controle de emissões de poluentes.

Em resumo, o padrão OBD-II especifica o conector físico usado para a interface de diagnóstico, o esquema de pinos, os protocolos de transporte de dados, e também os formatos das mensagens que podem ser trocadas acerca dos parâmetros que estão a ser monitorados pelas linhas de diagnóstico.

### 2.2.1 OBD II

A transferência de dados nessas interfaces segue vários padrões, mas nenhum deles pode ser usado diretamente por PCs ou dispositivos inteligentes (ELM ELECTRONICS [s.d.]). OBD (do inglês *On-Board Diagnostic*) é um sistema de autodiagnóstico para acesso ao estado dos sensores de um veículo por meio de um conector padrão (Figura 1). Dentre os dados coletados, destacam-se a velocidade linear, a rotação do motor (RPM), a temperatura do líquido de arrefecimento e as taxas de combustível e oxigênio (AMARASINGHE et al, 2015).



Figura 1 – Conector padrão OBD2

Fonte: <https://blog.mixauto.com.br/como-funciona-o-scanner-automotivo/>

### 2.2.2 PROTOCOLOS

Existem cinco protocolos que podem ser utilizados na interface OBD-II de forma simultânea, porém os fabricantes de veículos automóveis optam majoritariamente por utilizar apenas um. Tais protocolos podem ser facilmente identificados por meio da análise dos pinos presentes no conector (Figura 2), descritos no Quadro 1.

Quadro 1 – Descrição dos Pinos do conector OBD-II

Pino	Descrição
1	À descrição do fabricante
2	Linha positiva do barramento do SAE J1850
3	À descrição do fabricante
4	Massa do chassi
5	Massa de sinal
6	Linha CAN-High da ISO 15765-4 e SAE J2284
7	Linha K da ISO 9141-2 e ISO 14230-4
8	À descrição do fabricante
9	À descrição do fabricante
10	Linha negativa do barramento do SAE J1850
11	À descrição do fabricante
12	À descrição do fabricante
13	À descrição do fabricante
14	Linha CAN-Low da ISO 15765-4 e SAE J2284
15	Linha L da ISO 9141-2 e ISO 14230-4
16	Potencial da bateria

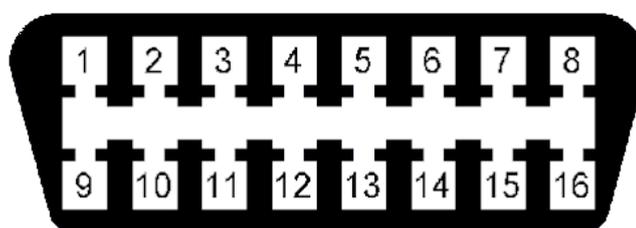


Figura 2 – Esquema dos pinos

Fonte: <https://www.keycentermonterrey.com/blog/configuracion-del-conector-dlc-obd2/>

Os protocolos suportados pela porta OBD-II possuem algumas características de velocidade e sinal que os diferenciam:

- SAE J1850 PWM Protocolo padrão da Ford Motor Company que apresenta uma mensagem com tamanho máximo definido de doze bytes que são transmitidas com uma modulação de largura de pulso a 41.6 kbit/s (RIBEIRO, 2015).

- SAE J1850 VPW Semelhante ao anterior este é um padrão da General Motors e pode apresentar taxas de transmissão de 10.4 kbit/s a 41.6 kbit/s (RIBEIRO, 2015).

- ISO 9141-2 Protocolo assíncrono que opera a uma taxa de 10.4 kbit/s semelhante ao protocolo RS-232, vulgo porta serial, com um tamanho máximo de mensagem de 260 Bytes (RIBEIRO, 2015).

- ISO 14230 KWP2000 Protocolo semelhante aos descrito no ponto anterior que opera a uma taxa que pode variar entre 1.2 kbit/s a 10.4 kbit/s (RIBEIRO, 2015).

- ISO 15765 CAN Protocolo síncrono que opera a uma taxa de 250 kbit/s ou 500 kbit/s, com um tamanho máximo de mensagem de 8 bytes (SÁNCHEZ CARRIZO, 2017).

### 2.2.3 PADRÃO DE COMANDOS

Os identificadores de parâmetros, PID (SAE J1979), são os códigos de comando usados pelo OBD-II para se comunicar com a ECU (SIM; SITOANG, 2014). Quando o usuário deseja obter alguma informação, um dispositivo enviará um PID para o OBD-II, que será lido pela ECU, que retorna as informações solicitadas, como, por exemplo, a temperatura do motor. Segundo Sitohang (2014), existem 9 modos de operação para PID, cada um fornecendo informações diferentes. Os modos disponíveis para PIDs são:

- **Modo 1:** Fornece dados atuais do motor em tempo real.
- **Modo 2:** Fornece informações de falha detectadas no motor.
- **Modo 3:** Códigos de erro padrão (DTC - Diagnostic Trouble Code) que a ECU armazena atualmente.
- **Modo 4:** Envia um comando para a ECU para limpar todos os códigos de erro padrão DTCs e desligar a lâmpada indicadora de mau funcionamento (MIL) se estiver ligada.
- **Modo 5:** Resultados de teste do monitoramento do sensor de oxigênio.
- **Modo 6:** Resultado do teste de outros sensores.
- **Modo 7:** DTCs pendentes.

- **Modo 8:** operação de controle do sistema de bordo.
- **Modo 9:** VIN (Número de Identificação do Veículo)

Quando um dispositivo de diagnóstico envia um PID, a ECU fornecerá uma resposta em uma série de bits que devem ser decodificados pelo dispositivo. As fórmulas para decodificar a resposta da ECU variam de acordo com o PID que é enviado. Essas fórmulas vão desde expressões muito simples (bit de resposta - 40) a fórmulas sofisticadas envolvendo declarações condicionais e cálculos complexos. Sitohang (2014) comenta sobre a obtenção das rotações por minuto (RPM, de PID 12):

$$RPM = (A \times 256 + B) \times 4 \quad (1)$$

A e B em (1) são o primeiro byte e o segundo byte da resposta da ECU, respectivamente.

## 2.3 MANUTENÇÃO

Atualmente, os veículos passam muito mais tempo parados no trânsito do que circulando em ambiente urbano, especificamente em grandes centros com alta densidade de veículos. Parados, porém com o motor em pleno funcionamento, pois os itens de conforto, como o climatizador, por exemplo, precisam funcionar. Desse modo é notável que certos componentes de veículos automotores podem se adaptar melhor a diferentes formas de temporizar a manutenção, podendo ser por quilômetros rodados, ou por horas de funcionamento, como é o mais presente em máquinas pesadas. Hoje em dia, quase todos os sistemas do veículo motorizado foram digitalizados, incluindo o hodômetro digital, que funciona para medir a quilometragem do veículo, e a inspeção periódica do veículo para substituição do lubrificante, sendo esta de suma importância a fim de se manter a durabilidade do motor do veículo (GALVANI et al., 2015).

## 2.4 MEDIÇÕES EM TEMPO REAL

O custo de funcionamento primário ao operar um veículo ou uma frota de veículos é o combustível (GUPTA et al., 2008). Comercialmente, isso tem influência direta nas taxas de

serviço. Logo, o conhecimento do consumo de combustível permite que os proprietários de veículos e administradores de frotas possam estimar seus custos mais efetivamente, a fim de otimizar sua logística e lucros.

Matematicamente a medição de consumo médio de combustível em um certo trajeto é algo relativamente fácil: basta realizar a leitura do hodômetro do veículo antes e depois de um certo trajeto e então dividir a distância percorrida pela quantidade de combustível consumida no trajeto. Porém, esse método é trabalhoso, impreciso, e não fornece o consumo em tempo real, que pode variar ao longo do trajeto, conforme a aceleração do motorista, ou particularidades da pista. O acesso a esse consumo em tempo real permite que o motorista procure recursos para otimizar a maneira de guiar o veículo a fim de reduzir o consumo de combustível, ou até mesmo a otimização da condução de veículos autônomos.

## 2.5 MQTT

O MQTT é um protocolo de mensagens padrão OASIS para a Internet das Coisas (IoT). Ele foi projetado como um transporte de mensagens de publicação/assinatura extremamente leve, ideal para conectar dispositivos remotos com um pequeno espaço de código e largura de banda de rede mínima (MQTT, 2020).

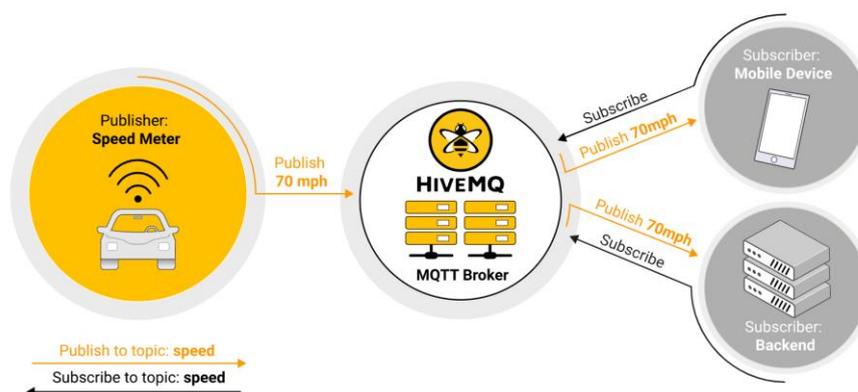


Figura 3 – Estrutura de publicação e inscrição do MQTT

Fonte : <https://www.hivemq.com/blog/mqtt-essentials-part2-publish-subscribe/>

Conforme corrobora Martins (2014), o MQTT foi consolidado por volta de 1999 por Andy Stanford-Clark (IBM) e Arlen Nipper (Eurotech). Trata-se de um protocolo de mensagens com arquitetura de publicação e inscrição, na qual o cliente escolhe as mensagens que quer obter com base na inscrição em um tópico.

Segundo Jaffey (2014) o modelo publicação/inscrição separa o cliente que envia uma mensagem (o publicador) do cliente ou clientes que recebem as mensagens (os assinantes). Ambos passam a ser intermediados por uma terceira camada, o broker. As mensagens a serem transmitidas são publicadas para um endereço (tópico), que se assemelha a uma estrutura de diretórios em um sistema de arquivos, como por exemplo, *casa/sala/temperatura*.

## 2.6 RECURSOS DE IMAGEM

Os veículos automotores, especificamente os carros possuem diversos mecanismos para promoção de segurança de caráter corretivo, quando o modo de condução seguro não é adotado, como airbags, cintos de segurança, entre outros. Já se pensando em mecanismos de segurança de caráter preventivo, pode-se optar por um sistema de percepção ativa do ambiente, a fim de obtenção de dados de práticas que remetem a atos inseguros, ou até mesmo para aprendizado de veículos autônomos.

Sensores diferentes, como sensores acústicos, scanners a laser, radares de micro-ondas e câmeras, podem ser usados para este objetivo. A percepção baseada na visão é a mais semelhante à função de percepção visual humana. As vantagens são que a aquisição de imagem não causa nenhuma interferência entre veículos ou ruído, e a visão computacional pode ser usada como ferramenta para inspecionar ampla gama e obter informações abundantes de cenas, conforme a Figura 4 (Fröhlich, 2013). Portanto, usando o ambiente de direção baseado em visão e sistema de percepção no veículo, os motoristas podem ser assistidos e os participantes protegidos (WANG et al., 2006, tradução nossa).

Já segundo Fröhlich (2013) uma das ferramentas de informação mais usadas atualmente é verificar o conteúdo das câmeras de trânsito ao vivo diretamente antes de iniciar uma viagem. À medida que esses serviços migram cada vez mais para plataformas móveis, os motoristas estão trazendo esses serviços para o carro por meio de seus dispositivos. Assim, embora concebido principalmente para situações pré-viagem, este conteúdo multimídia pode ser utilizado mesmo durante a condução, como por exemplo os serviços de captação de imagens de câmeras de celulares.

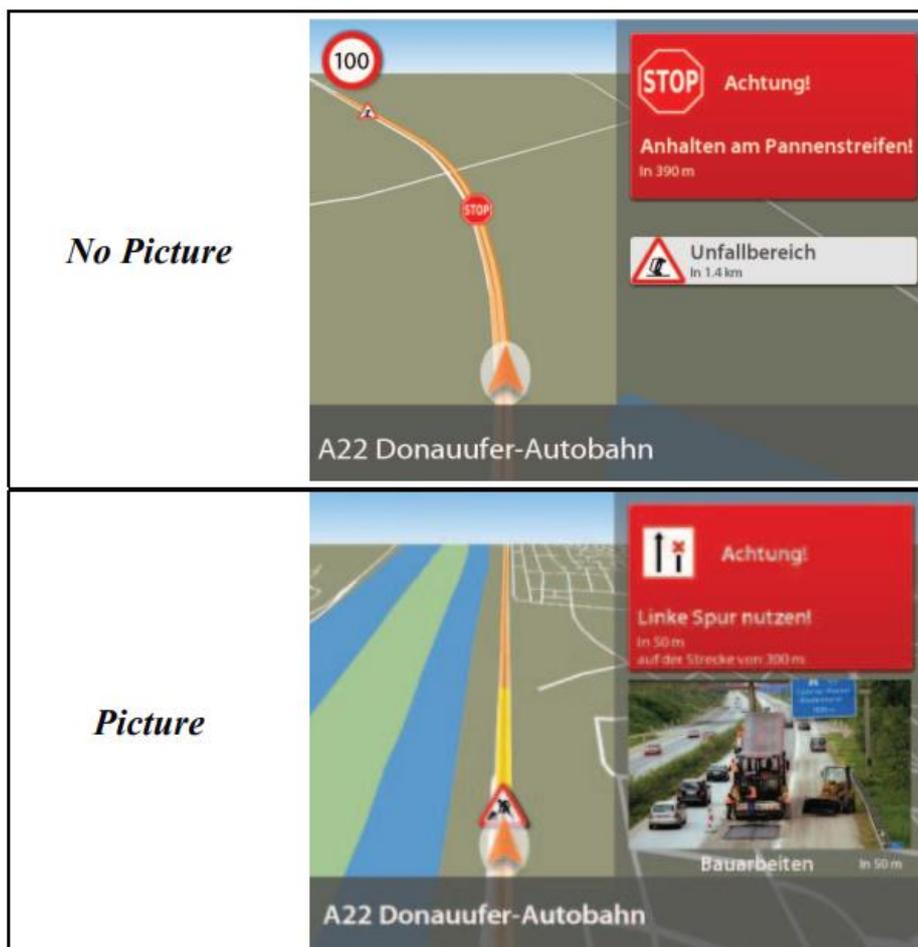


Figura 4 – GPS sem imagem de trânsito e GPS com imagem de trânsito

Fonte: Fröhlich, 2013

### 3. MATERIAIS E MÉTODOS

Este capítulo descreve todas as etapas que fizeram parte do desenvolvimento deste trabalho. Inicialmente são descritas as etapas de concepção do sistema e os materiais utilizados, em seguida a arquitetura que organiza e determina a hierarquia do sistema.

Por fim, apresentam-se os procedimentos da metodologia experimental e as métricas para se avaliar as condições de operação.

#### 3.1 LEVANTAMENTO DAS NECESSIDADES DO SISTEMA

Pelo fato da predominância empírica do trabalho, entende-se a necessidade de definir métricas a serem atendidas pelo sistema, com base em suas funcionalidades, a fim de proporcionar um dimensionamento adequado de hardware e software.

Das diversas variáveis proporcionadas pelo veículo, algumas podem ser extraídas diretamente do seu módulo de injeção, ou circuitos complementares, e outras podem ser gerados com uso de sensores externos, conforme o Quadro 2.

Quadro 2 – Possíveis fontes de dados do sistema

Funcionalidades	Fontes	
	Veículo	Sensores externos
Consumo de combustível		
Eventos de frenagem brusca		
Localização geográfica		
Estado do veículo		
Velocidade		

Em termos práticos, o trabalho foi roteirizado da seguinte forma:

1. Seleção de variáveis de tempo real;
2. Seleção de variáveis históricas;
3. Desenvolvimento de sistema embarcado;
4. Desenvolvimento de interface gráfica;
5. Avaliação do desempenho do sistema embarcado em coleta de dados;

As próximas seções descrevem as etapas supracitadas.

### **3.1.1 SELEÇÃO DE VARIÁVEIS DE TEMPO REAL**

Conforme destacado no quadro 1, diversas variáveis do veículo podem ser obtidas diretamente no circuito eletrônico do veículo. Para veículos que possuem a interface OBD2 essa obtenção de variáveis fica facilitada e padronizada. Diante à diversidade de variáveis oferecidas pela interface, algumas foram selecionadas de forma empírica a fim de se compor a parte do trabalho de monitoramento e exibição de dados em tempo real, sendo elas:

- Velocidade;
- Consumo de combustível;
- Eventos de frenagem brusca;
- Temperatura do motor (líquido de arrefecimento ou óleo).
- Estado do motor.

Essas variáveis contemplam os fatores em destaque no trabalho: consumo de combustível, segurança e manutenção.

### **3.1.2 SELEÇÃO DE VARIÁVEIS HISTÓRICAS**

Já de forma a se gerar um relatório em um banco de dados, outras variáveis foram selecionadas, com a adição das variáveis de aquisição em tempo real, como as coordenadas geográficas e o instante (data e hora) da obtenção dos dados.

Por meio desse armazenamento de dados históricos é possível preservar informações a longo prazo.

## **3.2 DESENVOLVIMENTO DE SISTEMA EMBARCADO**

A etapa de desenvolvimento do sistema embarcado envolve a integração entre as variáveis do veículo e um sistema de acompanhamento em tempo real e o banco de dados, de maneira segura e confiável. A estrutura de aquisição de dados do veículo foi dividida em duas camadas: sistema embarcado de baixo nível e sistema embarcado de alto nível.

O sistema de baixo nível lida com a aquisição direta de dados do veículo, e também com a obtenção de variáveis consequentes do deslocamento do veículo, como aceleração e posição geográfica. Já o sistema de alto nível armazena, processa e distribui os dados compartilhados pela de baixo nível.

Para testes que envolvem a coleta de dados disponibilizados diretamente pela infraestrutura do veículo, foi utilizado um motor Fiat Fire 1.0 alimentado por uma injeção eletrônica Magneti Marelli IAW59FB, com os dados sendo extraídos da porta OBD-II da injeção por meio de um scanner multiprotocolos ELM327, conforme detalhado na Seção 3.2.2.

### 3.2.1 SISTEMA EMBARCADO DE BAIXO NÍVEL

Para obtenção dos dados provenientes do ELM327, aquisição de dados de aceleração e localização, foi escolhido um microcontrolador ESP32-WROOM-32 (“ESP32-WROOM-32 Datasheet”, [s.d.]), conforme a figura 4.



Figura 5 – Placa de desenvolvimento com ESP32

Fonte: <https://www.amazon.in/Easy-Electronics-Development-Bluetooth-Consumption/dp/B07TYCFX5C>

Essa escolha foi baseada na disponibilidade de comunicação via Bluetooth de forma integrada, que permite o trabalho com o scanner ELM327 sem alterar suas características originais de fabricação e sem a necessidade de integração com outros módulos de comunicação via Bluetooth, o que reduz o custo e a complexidade do projeto.

Além disso, o ESP32 possui baixo custo e desempenho considerável, contando com um controlador WiFi, utilizado para o envio dos dados obtidos na rede MQTT em intervalos pré-definidos de tempo, para a interface de acompanhamento em tempo real e para o sistema embarcado de alto nível.

Já para os eventos de frenagem brusca, foi optado por priorizar um canal de envio de dados de desaceleração, a porta serial do ESP32, trabalhando com uma taxa de transmissão de 115200 baud rate, que disponibiliza ao sistema embarcado de alto nível a aceleração do veículo a todo instante. Caso a aceleração seja menor que  $-0,5g$ , é interpretada como um evento de frenagem brusca. No que diz respeito à obtenção de localização geográfica, é realizada a interface entre o ESP32 e um módulo de geolocalização NEO-6m via comunicação serial.

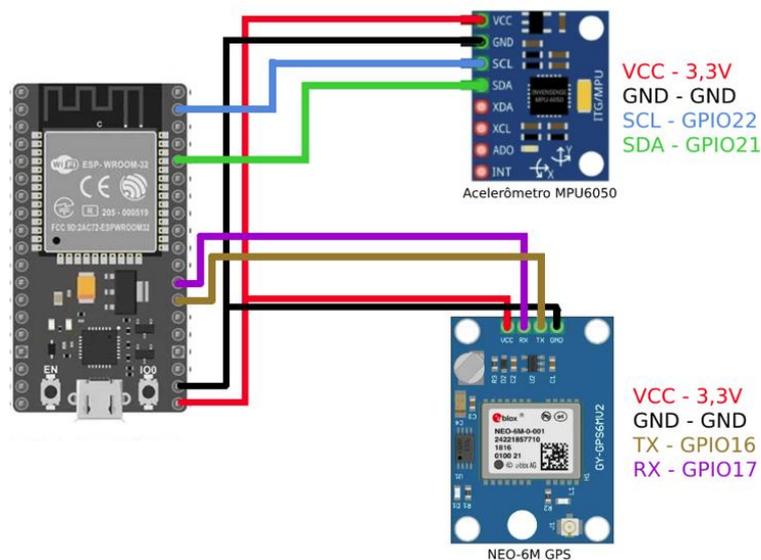


Figura 6 – Sistema embarcado de baixo nível

Fonte: Autoria própria

### 3.2.2 ELM 327

Foi utilizado no projeto o ODBII Scan (Figura 6), pois o mesmo tem suporte a todos protocolos padrão da interface OBD2, permitindo uma alta compatibilidade do sistema com diversos veículos, além do baixo custo. Este scanner utiliza um microcontrolador PIC18F2460 que simula o circuito integrado ELM327 (ELM Electronics), que por sua vez é projetado para traduzir os protocolos do OBDII realizando a leitura dos dados originados no veículo.



Figura 7 – Scanner genérico OBDII

Fonte: Autoria própria

O OBDII Scan compartilha os dados via Bluetooth ou UART, sendo que esse último caso implica em acessar certas vias não disponíveis no conector externo, mas presentes na placa de circuito impresso. Com intuito de manter a originalidade do scanner e preservar sua garantia, foi feita uma comunicação via Bluetooth com o ESP32, por meio da biblioteca ELMduino (PB2, 2022 ), concedida de forma gratuita e livre.

### 3.2.3 ESTIMAÇÃO DE CONSUMO DE COMBUSTÍVEL

Entre os PIDs existentes em um veículo, possivelmente pode haver a disponibilidade do PID 94 (SAE J1979), que retorna o consumo de combustível em litros por hora. Na ausência do mesmo, de acordo com Fontes (2014) é possível realizar o cálculo do consumo, com base nos seguintes dados: velocidade do veículo (PID 13) e a taxa de massa de ar (PID 11).

Porém, de acordo com Fontes (2014, apud Lightner 2005), apesar de nem todos os veículos possuírem um sensor MAF, é possível estimar a taxa de massa de ar utilizando outros dados fornecidos pelo veículo como pressão absoluta do coletor de admissão (MAP, obtido com PID 11) em Kpa, temperatura do ar de admissão (IAT, obtido com PID 15) em kelvin e RPM (PID 12), por meio do cálculo da variável sintética IMAP (2). Para tanto, é necessário saber o valor do volume do deslocamento do motor (VDM) em litros e a sua eficiência volumétrica (EV) em porcentagem.

$$IMAP = \frac{RPM \times MAP}{IAT/2} \quad (2)$$

De posse do valor de IMAP, é possível aproximar o MAF da seguinte forma:

$$MAF = \frac{\frac{IMAP}{60} \times EV \times VDM \times MM}{R} \quad (3)$$

onde MM é a massa molecular média do ar, com valor de 28,97 gramas por mol e R é uma constante representando 8,314 joules por kelvin por mol. (FONTES 2014, apud. LIGHTNER, 2011).

Com base nos cálculos propostos por Fontes (2014), o consumo de combustível em KM/l pode ser obtido através da equação 4, em que VSS é a velocidade do veículo em quilômetros por hora, AC é razão estequiométrica de ar/combustível,  $\mu_g$  é a densidade da gasolina em gramas por litro e MAF taxa de massa de ar em gramas por segundo.

$$\text{Consumo} = \frac{VSS}{3600} \times MAF^{-1} \times AC \times \mu_g \quad (4)$$

### 3.2.4 MÓDULO GPS NEO-6M

O módulo NEO-6M (Figura 7) traz o alto desempenho do mecanismo de posicionamento u-blox6. O u-blox6 é projetado com baixo consumo de energia e baixo custo em mente. O gerenciamento inteligente de energia é um avanço para aplicações de baixo consumo de energia. Esses receptores combinam um alto nível de capacidade de integração com opções de conectividade flexíveis em um pacote pequeno. Isso os torna perfeitamente adequados para produtos finais do mercado em grande escala com requisitos elevados de tamanho e custo. Os módulos são qualificados de acordo com a ISO 16750, que especifica condições ambientais e testes elétricos para equipamentos eletrônicos aplicados em veículos rodoviários (“NEO-6 series Versatile u-blox 6 GPS modules Product description”, [s.d.]).

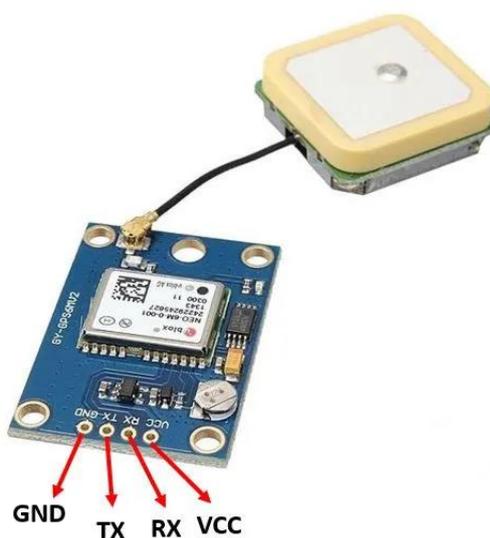


Figura 8 – Módulo GPS Neo-6m

Fonte: Autoria própria

### 3.2.5 MÓDULO MPU - 6050

O chip MPU-6050 (Figura 8) consiste em um processador, um sensor de giroscópio, um sensor de aceleração, um sensor de temperatura e um conversor analógico-digital (ADC) de 16 bits. A programação do dispositivo é realizada escrevendo nos registradores de configuração, e a comunicação é feita através de uma interface I<sup>2</sup>C no padrão 400kHz ou SPI até 20MHz.

Em relação ao acelerômetro, trata-se de um sensor capaz de detectar e medir a aceleração, incluindo a aceleração da gravidade e qualquer outra aceleração de movimento. O MPU-6050 é capaz de medir separadamente a aceleração em três eixos até 16g (156.896 m/s<sup>2</sup>) com quatro faixas de programação e precisão: +2g, +4g +6g e +16g em unidades não pertencente ao SI g refere-se ao valor da aceleração gravitacional (“MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.0 MPU-6000/MPU-6050 Register Map and Descriptions”, 2012).

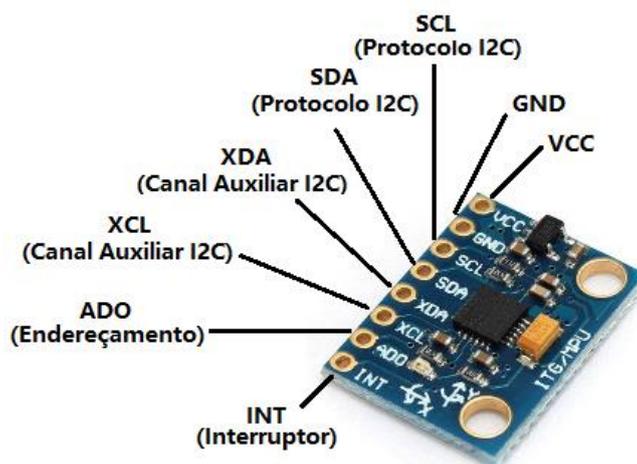


Figura 9 – Módulo MPU-6050

Fonte: Autoria própria

### 3.2.6 SISTEMA EMBARCADO DE ALTO NÍVEL

O sistema embarcado de alto nível tem como principais responsabilidades armazenamento de dados de forma remota via internet por conexão WiFi, compartilhada por um telemóvel por meio de rede 3G e 4G, execução de buffer circular para aquisição de imagens constantemente, e armazenamento das imagens em caso de frenagens bruscas, ou seja, eventos de força  $g$  negativa enviados pelo ESP32. Com base nesses pré-requisitos operacionais, foi escolhido o kit de desenvolvimento BeagleBone Black Rev C (“BeagleBoard.org - black”, [s.d.]), conforme a Figura 9. Ele é baseado no processador AM3358x que integra um ARM Cortex™-A8, rodando a 1GHz e disponibilizando vários periféricos, com destaque para a porta USB. Com um HUB, conectamos o ESP32 para envio de dados de aceleração e também um adaptador WiFi compacto, para a conexão com a Internet. A entrada de cartão SD, por sua vez, permite a expansão da memória de armazenamento.

No quesito armazenamento de dados, será utilizado um banco de dados não relacional. Todos os bancos de dados não-relacionais possuem características e requerimentos que os diferenciam dos bancos de dados relacionais convencionais. Essas características “os tornam capazes de manipular grandes volumes de dados não estruturados ou semiestruturados: Escalabilidade, Alta disponibilidade, esquema flexível e simples manipulação” (SILVA DE OLIVEIRA, 2014).

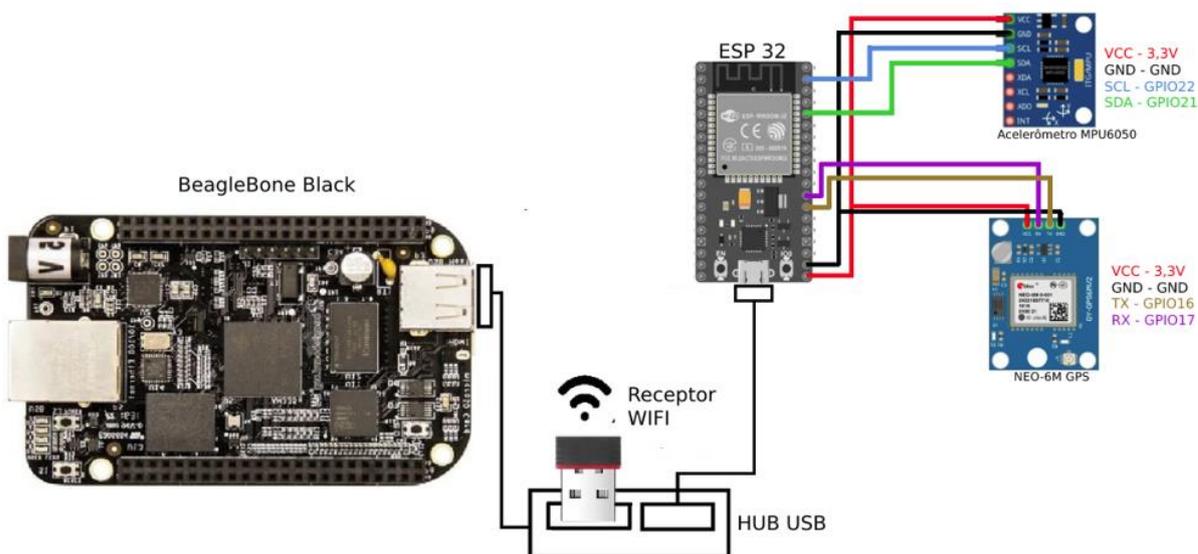


Figura 10 – Integração sistemas embarcados alto nível-baixo nível

Fonte: Autoria própria

### 3.2.7 DESENVOLVIMENTO DA APLICAÇÃO IOT

Com a integração do carro com sistema embarcado, pode-se prosseguir para a etapa de consumo dos dados remotamente, por meio de uma aplicação IoT. Será aplicada uma arquitetura *webhook* (gancho da web em tradução literal), que se define como um recurso que possibilita o envio de dados em tempo real entre dois sistemas ou aplicativos distintos, nesse caso, o sistema embarcado. Nesse sentido, o sistema se tornará mais dinâmico, permitindo o consumo dos dados fornecidos de forma remota, e por múltiplos usuários, além de permitir um acesso aos dados por múltiplas plataformas.

As tecnologias escolhidas para o desenvolvimento da aplicação de consumo e processamento dos dados foram o ambiente de execução Javascript Node.js e as linguagens TypeScript para o consumo de dados da ECU fornecidos pelo ESP32 e Python 3 para a funcionalidade de captura de imagens nos eventos de frenagem brusca. Já para o armazenamento remoto foi aplicado o banco de dados não-relacional RavenDB, por possibilitar seu uso de forma remota e local, além de ser compatível com a arquitetura de processamento da BeagleBone Black. Ambas as tecnologias são de código-aberto e tem uma forte comunidade de suporte consolidada.

### 3.3 AVALIAÇÃO DE DESEMPENHO DO SISTEMA EMBARCADO

Com a etapa de consumo de dados funcional, segue-se para a etapa de testes com aplicação em ambiente real, operando o sistema com veículo em movimento a fim de verificar as limitações de processamento e envio de dados para a aplicação Web, por meio de conexão móvel de internet disponível.

São realizados teste de comunicação de dados em ambientes rodoviário e urbano, e testes específicos de leitura de eventos, como de frenagem brusca em ambientes

### 3.4 TRANSMISSÃO DE DADOS

Para a comunicação entre o sistema embarcado de baixo nível e o *webhook* e o *dashboard*(Painel em tradução literal), será utilizada a comunicação por meio de protocolo MQTT, dividida entre um *broker* local para os dados a serem armazenados, e um *broker* remoto para exibição.

Na relação sistema embarcado de baixo nível e alto nível, serão utilizados, conforme a Figura 10, os seguintes tópicos: TCCvelocidade, TCCtemperatura, TCCcoordenadas, TCCconsumo, TCCrpm, TCCimagem, TCCcarro. Esses tópicos contêm mensagens para a velocidade instantânea do veículo, a temperatura do líquido de arrefecimento do motor, as coordenadas geográficas do veículo, o consumo de combustível instantâneo, as rotações por minuto do motor, as imagens em caso de frenagens bruscas e o nome do veículo, sendo todos acessados via broker local.

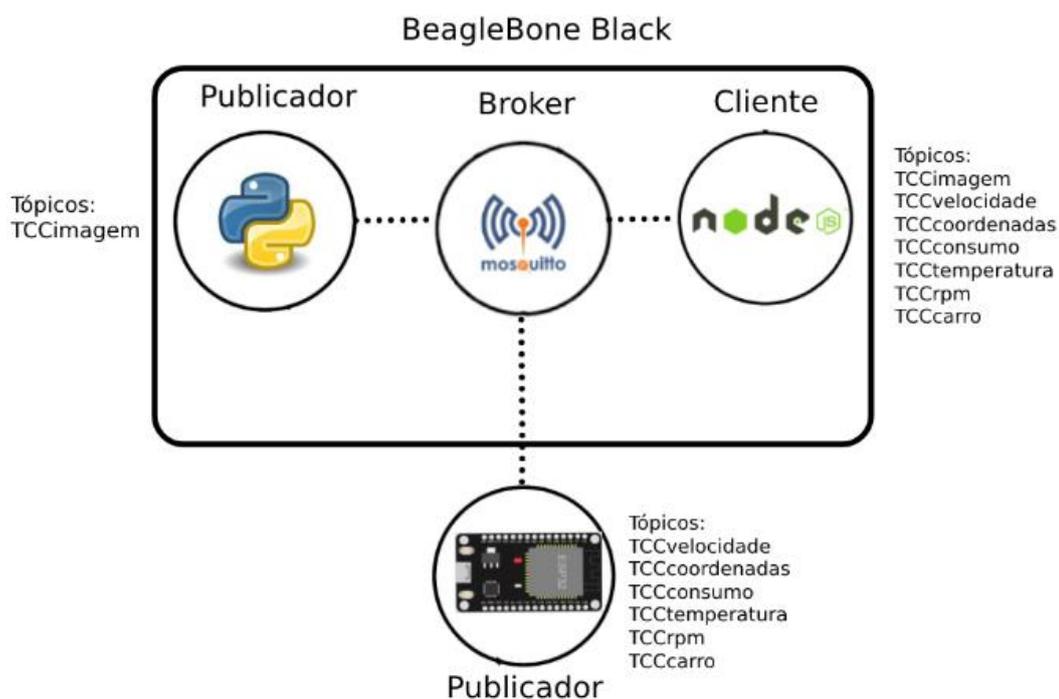


Figura 11 - Estrutura de comunicação com broker local

Fonte: Autoria própria

Já na relação do sistema embarcado de baixo nível e o *dashboard* de exibição de variáveis em tempo real, serão utilizados, conforme Figura 11, os seguintes tópicos: TCCvelocidade, TCCtemperatura, TCCconsumo, TCCrpm, TCCcarro, que informam os mesmos dados citados anteriormente. Desta vez, contudo, são acessados pelo broker remoto.

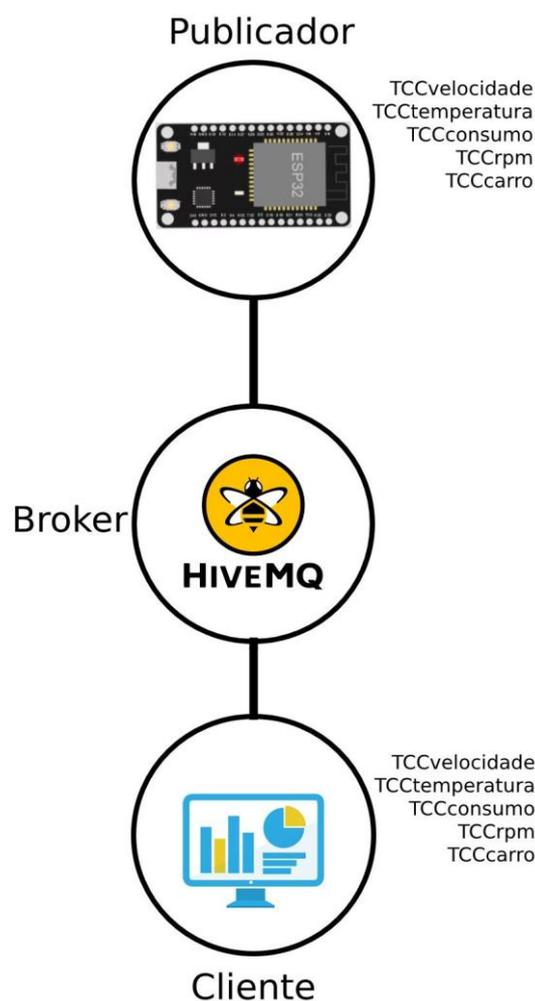


Figura 12 - Estrutura de comunicação com broker remoto

Fonte: Autoria própria

### 3.5 DESENVOLVIMENTO DA INTERFACE GRÁFICA COM OS USUÁRIOS

Com o sistema já funcional, faz-se necessário o desenvolvimento de uma interface gráfica para que o mesmo possa ser utilizado de forma facilitada e acessível. A parte gráfica, também chamada de *front-end*, foi desenvolvida no framework Angular, permitindo que o monitoramento possa ser feito de forma mais assertiva e remota por qualquer usuário que tenha conexão com a internet.

Além da ampla conexão remota, é notável o ganho de acessibilidade da interface tendo em vista que, por funcionar dentro de um navegador Web, é acessível a diversas plataformas presentes no mercado, como celulares, tablets, computadores, smart TVs etc, o que facilita o desenvolvimento de trabalhos futuros.

## 4. RESULTADOS

Os testes consistem em quatro etapas pré-definidas: simulação em ambiente urbano e rodoviário, testes de frenagem brusca e telemetria de dados vindos de uma ECU de um veículo. Abaixo explicaremos mais detalhadamente cada etapa do projeto e as alternativas para a solução dos problemas encontrados.

### 4.1 VALIDAÇÃO DE REDE 3G E 4G

O sistema em questão tem como objetivo adquirir dados de um automóvel e exibi-los, em tempo real, em um *dashboard* remoto, sendo eles: rpm, velocidade, consumo de combustível e temperatura do motor. Além disso, deve-se armazenar essas variáveis ao longo do tempo de forma remota, com a adição de localização, data do armazenamento, e registro de tempo de funcionamento, por meio de conexão com a internet. Em caso de instabilidade de conexão, deve-se salvar esses dados de forma local, na memória interna do sistema embarcado de alto-nível, baseado na BeagleBone Black.

A fim de explorar as consequências da instabilidade de rede, foram realizados testes com conexões provenientes de redes 3G e 4G. Isso foi feito tanto estaticamente, para os casos de visualização em tempo real, como em movimento, para os casos de armazenamento. Em ambas as abordagens, foram usados dados simulados a fim de reduzir a complexidade dos testes.

#### 4.1.1 TELEMETRIA

Para a validação do comportamento do sistema de aquisição de dados e exibição em tempo real, foram efetuadas verificações no intervalo de obtenção de dados aleatórios e simulados pelo *dashboard*, a fim de verificar as discrepâncias envolvidas em cada tipo de rede. O envio de dados para a aplicação de monitoramento é feito diretamente entre o ESP 32 e um broker remoto e gratuito, o *hiveMQ*. Na prática, para a realização desse teste, a interface de visualização interrompe o seu funcionamento após 5 minutos de seu início, e fornece os intervalos entre uma mensagem recebida e a anterior no console do navegador Web, em milissegundos (Figura 13).

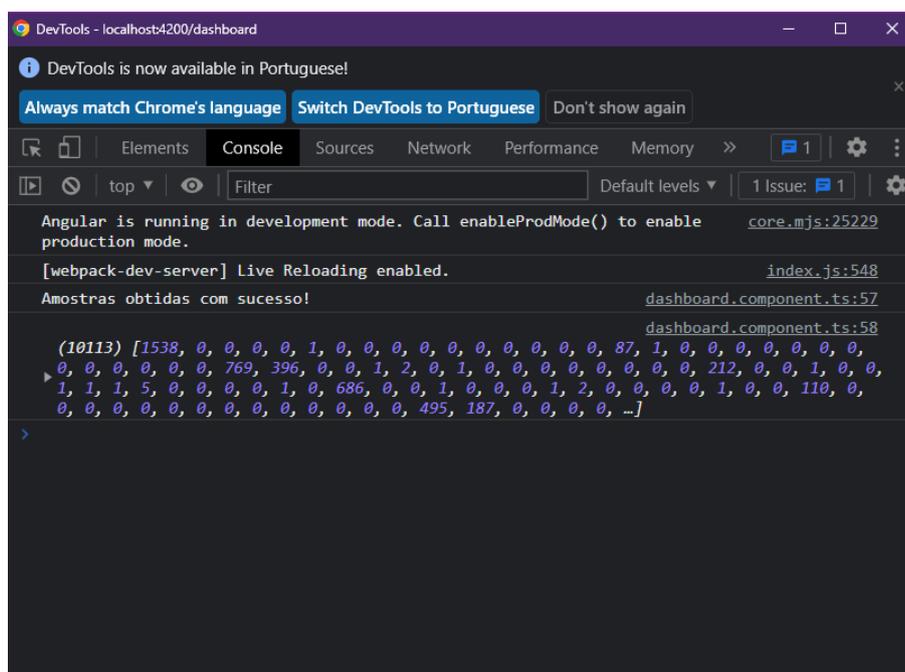


Figura 13 – Console do navegador

Fonte: Autoria própria

Em posse desses dados, tanto nos testes com rede 4G, quanto com 3G, realizamos o cálculo de média e desvio padrão dos intervalos obtidos. Conforme a Tabela 1, para uma amostra de 9560 intervalos obtida com conexão 3G, foi observada uma média entre os intervalos de mensagem de 31,33ms e um desvio padrão de 253,06 ms. Já para o teste com 4G, obtivemos uma amostra de 10481 intervalos, com média de tempo entre mensagens de 28,63 ms e desvio padrão de 119,56ms. Com base nesses dados obtidos, foi possível notar que, para a percepção humana, e para a disposição dos dados nos gráficos de forma contínua, ambas redes fornecem qualidade aceitável.

Tabela 1 - Intervalo de aquisição de mensagens

Tipo de conexão	Intervalos obtidos	Tempo médio(ms)	Desvio padrão (ms)
4G	10481	28,63	119,56
3G	9560	31,33	253,06

Fonte: Autoria própria

Outro ponto a se destacar é que o tempo médio entre mensagens do 4G foi somente 8,6% menor que a do 3G, porém o desvio padrão entre o tempo de mensagens foi 52,7% menor que no 3G, possibilitando dessa forma confirmar que a rede 4G oferece maior estabilidade na conexão ao longo do tempo.

Além disso, para trabalhos futuros, notamos a necessidade verificar o desvio padrão elevada se comparada à média para ambos casos, tendo em vista que outros fatores contribuem para a chegada da mensagem do sistema embarcado de baixo nível até o *dashboard* que não foram levadas em consideração, como o desempenho da rede de aquisição dos dados (*dashboard*), e a capacidade de demanda do broker remoto (hiveMQ).

#### 4.1.2 ARMAZENAMENTO DE DADOS

No que diz respeito a funcionalidade de armazenamento de dados, foi proposto que primeiramente ocorre a tentativa de armazenamento de dados de forma remota, por meio da conexão com a internet, e que somente em caso de falha o armazenamento ocorre de forma local. Este último cenário implica em uma aquisição posterior pelo usuário, por meio da interface de acesso ao banco de dados (Figura 14), que pode ser feita por qualquer computador que esteja conectado na mesma rede do sistema embarcado de alto nível.

Id	data	velocidade	temperatura	veiculo	local	consumo
EventoTelemetriaColle...	2022-06-30T15:39:25...	41	104	Carro Teste	xxxxxxxxxxxxxxxx	2
EventoTelemetriaColle...	2022-06-30T15:39:20...	43	107	Carro Teste	xxxxxxxxxxxxxxxx	2
EventoTelemetriaColle...	2022-06-30T15:39:15...	38	86	Carro Teste	xxxxxxxxxxxxxxxx	1
EventoTelemetriaColle...	2022-06-30T15:39:10...	57	107	Carro Teste	xxxxxxxxxxxxxxxx	1
EventoTelemetriaColle...	2022-06-30T15:39:05...	31	101	Carro Teste	xxxxxxxxxxxxxxxx	0
EventoTelemetriaColle...	2022-06-30T15:39:00...	49	105	Carro Teste	xxxxxxxxxxxxxxxx	1
EventoTelemetriaColle...	2022-06-30T15:38:54...	36	98	Carro Teste	xxxxxxxxxxxxxxxx	2
EventoTelemetriaColle...	2022-06-30T15:38:49...	40	115	Carro Teste	xxxxxxxxxxxxxxxx	0
EventoTelemetriaColle...	2022-06-30T15:38:44...	30	76	Carro Teste	xxxxxxxxxxxxxxxx	2
EventoTelemetriaColle...	2022-06-30T15:38:39...	59	113	Carro Teste	xxxxxxxxxxxxxxxx	2
EventoTelemetriaColle...	2022-06-30T15:38:34...	46	75	Carro Teste	xxxxxxxxxxxxxxxx	1
EventoTelemetriaColle...	2022-06-30T15:38:29...	36	109	Carro Teste	xxxxxxxxxxxxxxxx	2

Figura 14 – Interface gráfica do cliente RavenDB

Fonte: Autoria própria

Por meio dessa interface administrativa, é possível ver os dados salvos, filtrar com base nas colunas, e exportar os dados em formato CSV, que podem ser importados posteriormente para o banco de dados remoto ou serem utilizados conforme a necessidade do usuário.

### 4.1.3 TESTE URBANO

Para a validação do comportamento do sistema de armazenamento de dados em ambiente urbano, foi efetuado teste no volume de dados armazenados no banco de dados local e remoto, diante o deslocamento de um veículo em um determinado percurso (Figura 15) a fim de verificar a sensibilidade das redes 3G e 4G às diferenças de intensidade de sinal, as variações de velocidade do deslocamento do veículo e as intempéries ambientais.

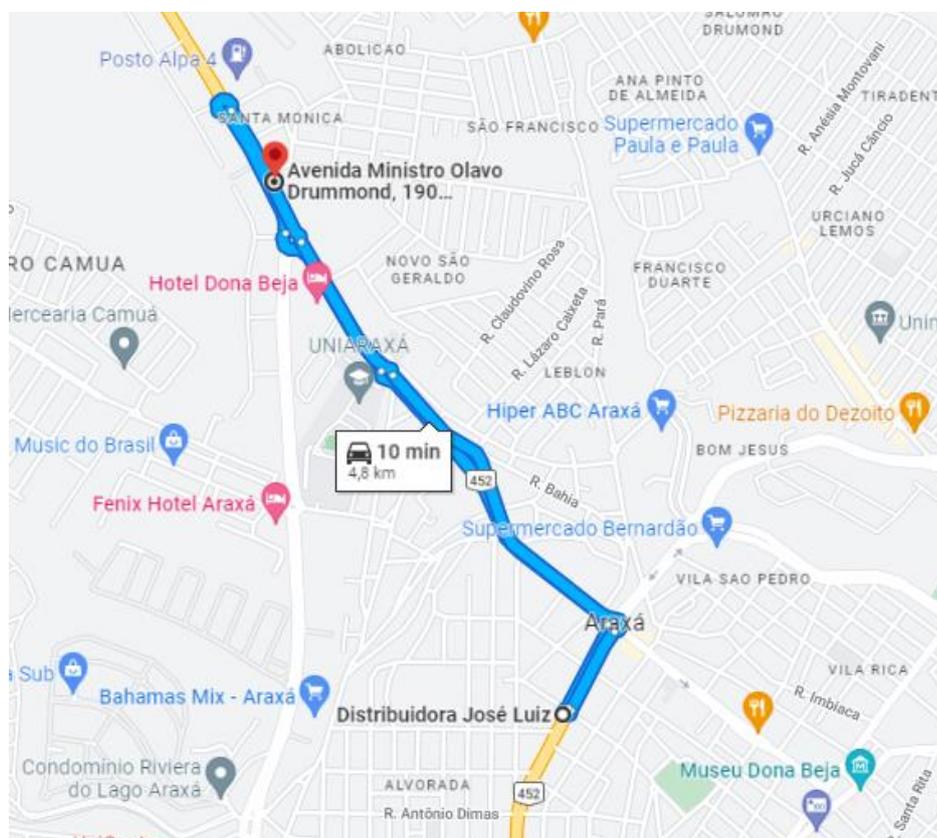


Figura 15 – Percurso urbano

Fonte: Autoria própria

No teste em questão, com armazenamento das variáveis no banco de dados a cada 5 segundos, foram obtidos 90 registros, para a conexão de internet via 3G, com todos os dados sendo armazenados de forma remota, ou seja, não ocorreram instabilidades na conexão que pudessem impactar o sistema.

Posteriormente, o mesmo teste foi realizado com conexão 4G em horário próximo, a fim de aproximar os cenários de ambos os testes, no que diz respeito ao volume de trânsito urbano. Foram obtidos 102 registros, para a conexão de internet via 4G, com todos os dados

sendo armazenados de forma remota. Novamente, não ocorreu instabilidade na conexão, validando a usabilidade de ambas as redes como provedoras de comunicação para o sistema.

#### 4.1.4 TESTE RODOVIÁRIO

No caso do ambiente rodoviário, foi feito um teste análogo, com o percurso ilustrado na Figura 16.

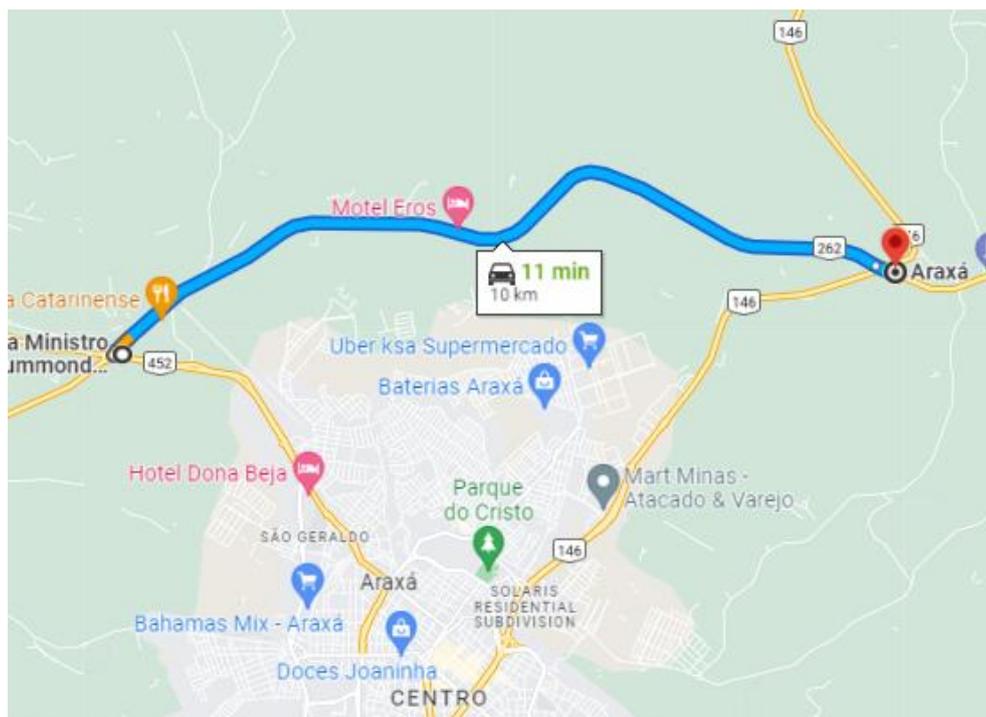


Figura 16 – Percurso rodoviário

Fonte: Autoria própria

No teste rodoviário, com o armazenamento das variáveis no banco de dados a cada 5 segundos, foram obtidos 86 registros, para a conexão de internet via 3G, com 74 sendo armazenados de forma remota e 12 de forma local, ou seja, nesse caso pôde-se observar que houve uma instabilidade na conexão ao longo do percurso.

Similarmente ao que foi feito no teste urbano, o teste rodoviário foi repetido na rede 4G em horário próximo, a fim de aproximar os cenários de ambos os testes, no que diz respeito ao volume de trânsito rodoviário. Foram obtidos 112 registros para a conexão de internet via 4G, com 102 sendo armazenados de forma remota e 10 de forma local, evidenciando

novamente uma pequena instabilidade ao longo do percurso. Portanto, podemos observar que com conexão via 3G houve uma instabilidade em 13,95% das tentativas de armazenamento, e no 4G, 8,9%, conforme (Figura 17).

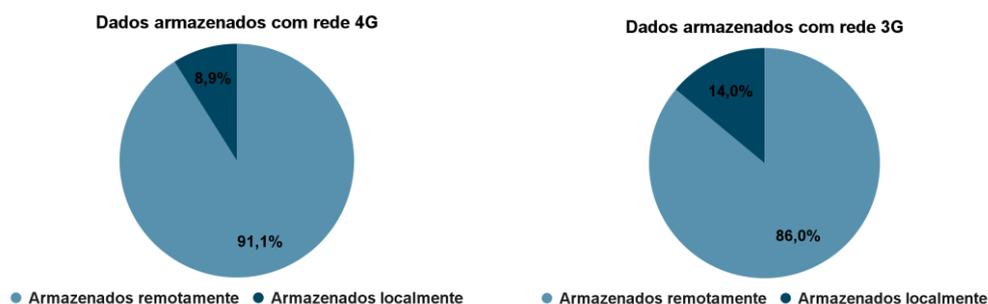


Figura 17 – Distribuição do armazenamento (local-remoto) nas redes 3G e 4G

Fonte: Autoria própria

Diante desses dados foi constatado que ambas as redes apresentam desempenho próximo no quesito estabilidade ao longo do percurso avaliado e, além disso, o sistema correspondeu bem à expectativa de armazenamento de dados em caso de presença ou ausência de conexão com a internet.

## 4.2 IDENTIFICAÇÃO DE FRENAGENS BRUSCAS

A identificação de frenagem brusca é feita através do envio de valores de aceleração obtidos pelo sistema embarcado de baixo nível, por meio de um acelerômetro, e enviadas para o sistema de alto nível via comunicação UART. O último possui um programa em Python responsável por armazenar imagens provenientes da câmera de um smartphone, que são enviadas ao programa via *sockets* TCP. Isso é possível graças a aplicativos que transformam o periférico em uma *IP Webcam*.

Foram realizados três testes com objetivo de validar a capacidade de aquisição de imagens assertivas em um evento de frenagem brusca: aplicação de frenagem com o carro se deslocando a 70 km/h, 50 km/h e 30 km/h. Para cada evento de frenagem, são capturadas 30 imagens localmente na BeagleBone Black, que são armazenadas em um buffer circular.

Para padronizar os testes, foi posicionado um indicador na lateral da via, que indica o ponto de frenagem (Figura 18).



Figura 18 – Sinalizador de ponto de frenagem

Fonte: Autoria própria

O sinalizador foi capturado com sucesso nas três velocidades, conforme a Figura 19 (teste a 70 km/h), a Figura 20 (teste a 50 km/h) e a Figura 21 (teste a 30 km/h). Em todos os casos, a imagem enviada é aquela que apresenta a maior proximidade do ponto alvo, dentre as 30 imagens presentes no buffer circular na ocasião da detecção da frenagem brusca.

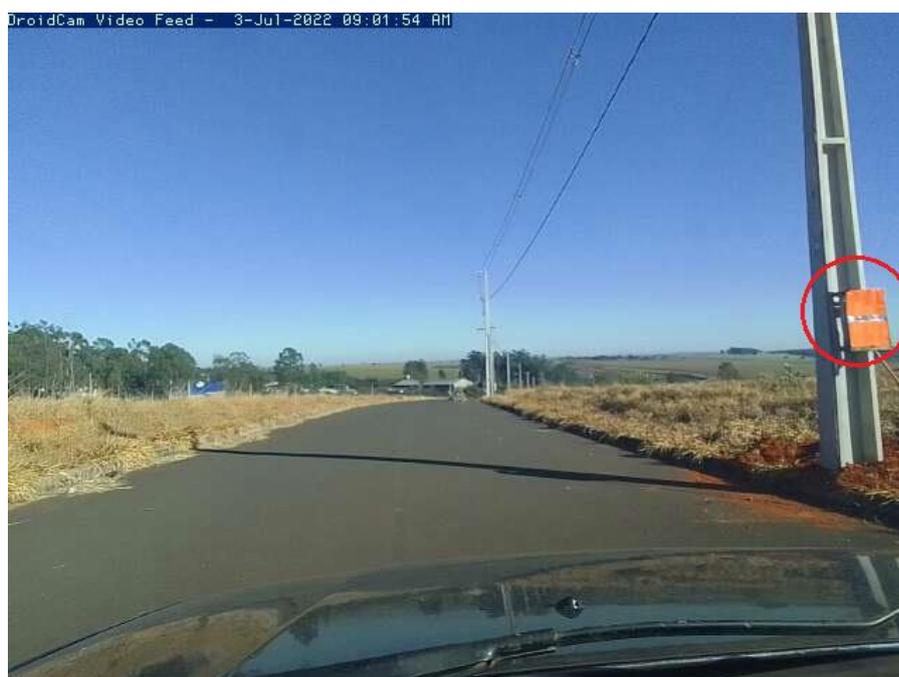


Figura 19 – Teste de frenagem brusca a 70 km/h

Fonte: Autoria própria



Figura 20 – Teste de frenagem brusca a 50 km/h

Fonte: Aatoria própria



Figura 21 – Teste de frenagem brusca a 30 km/h

Fonte: Aatoria própria

Portanto, pudemos observar que o sistema tem capacidade de adquirir imagens assertivas, no que diz respeito a identificação da possível causa de uma frenagem brusca, como a tentativa de travessia de um pedestre por exemplo, conforme a Tabela 2.

Tabela 2 – Relação de imagens capturadas

Velocidade de teste(km/h)	Imagens com alvo identificável	Imagens com alvo não identificável
70	23	7
50	21	9
30	22	8

Ademais, para integração do sistema com armazenamento remoto, seriam necessários mais testes a fim de identificar automaticamente qual dentre as 30 imagens poderiam ser mais conclusivas para serem enviadas para o banco de dados remoto, evitando o consumo de memória com imagens não conclusivas, como a da Figura 22.



Figura 22 – Imagem sem captura de sinalizador

Fonte: Autoria própria

### 4.3 AQUISIÇÃO DE DADOS REAIS

Conforme retratado anteriormente, a fim de facilitar os procedimentos de teste, e validar o desempenho da rede, os dados armazenados em banco de dados e exibidos em *dashboard* foram simulados. Por fim, para validar o sistema como funcional com base nos seus pré-requisitos, ele foi testado com todas as funcionalidades em conjunto, com fontes de dados reais.

Para a aquisição de dados, no que diz respeito às variáveis de ECU, foi utilizado um veículo Volkswagen Polo 1.6. Conforme identificamos no teste inicial, o veículo não possui um suporte a um PID específico de consumo de combustível (94), portanto o referido parâmetro foi estimado com base nas equações (2), (3) e (4). Os demais PIDs eram suportados normalmente pela ECU do veículo.

Em posse desses dados, foi percorrido um percurso pré-definido, ilustrado na figura 23.

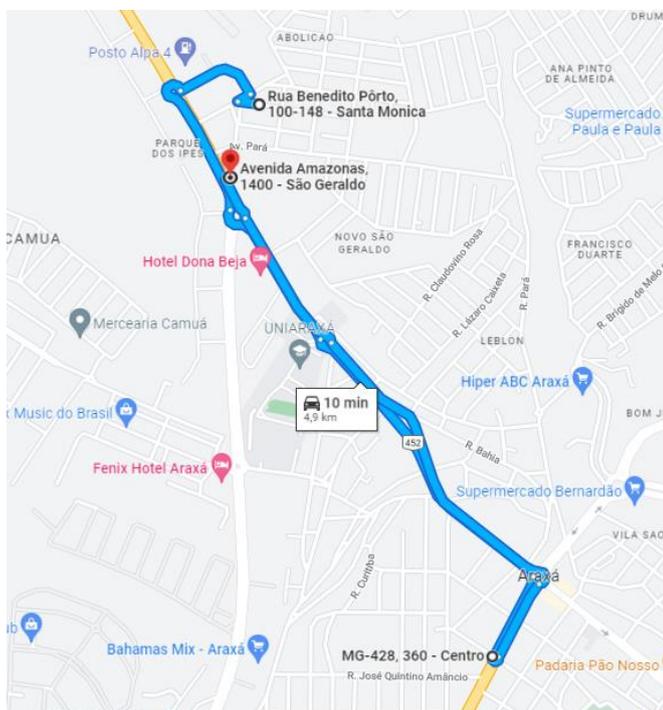


Figura 23 – Percurso de teste com dados reais

Fonte: Autoria própria

Nessa etapa, foram obtidos dados reais de geolocalização ao longo do percurso, os quais puderam ser extraídos do banco de dados para conferência. Por meio de uma ferramenta do Google Maps, as coordenadas foram inseridas em um mapa, a fim de estimar uma rota (Figura 24).

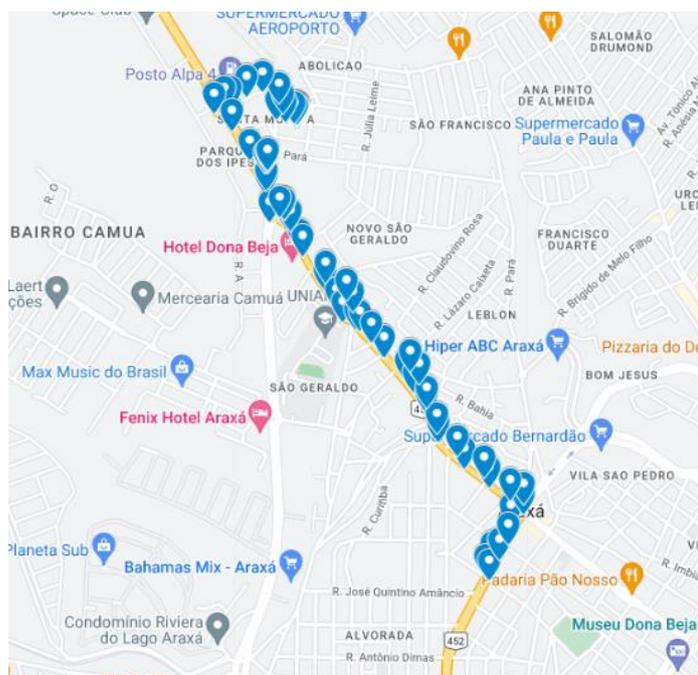


Figura 24 – Coordenadas armazenadas no percurso

Fonte: Autoria própria

Quanto aos dados referentes à ECU do veículo, toda a captura de dados ocorreu conforme o esperado ao longo do percurso, com dados condizentes com o painel do carro (“Dashboard de monitoramento automotivo em tempo real”, 2022), conforme a Figura 25 e a Figura 26.

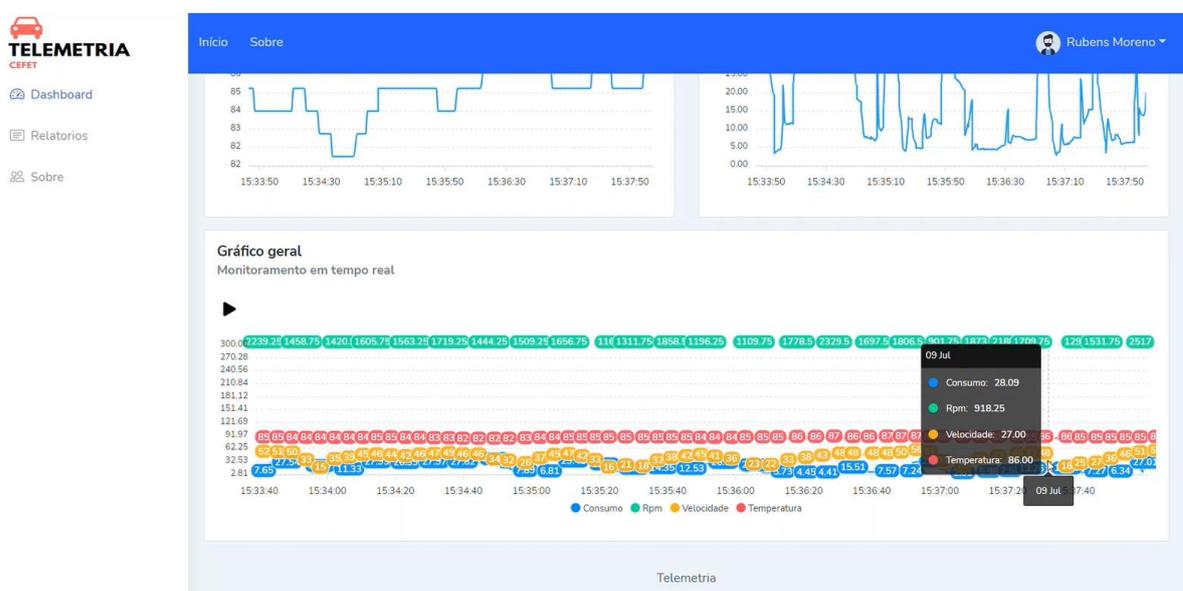


Figura 25 – Gráfico geral pausado

Fonte: Autoria própria

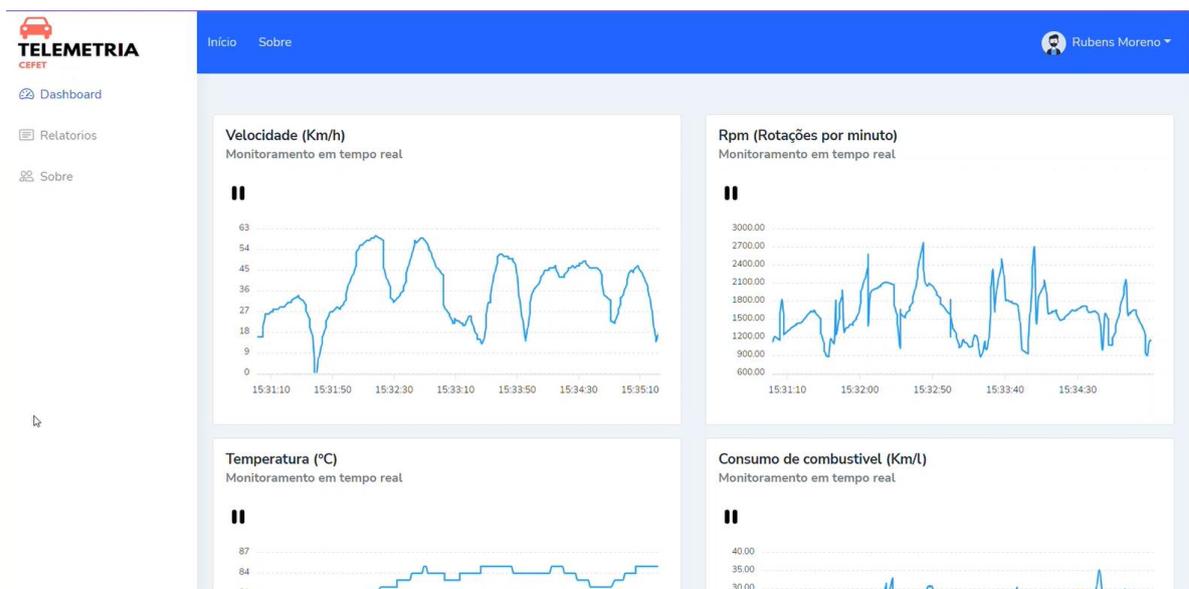


Figura 26 – Gráficos individuais

Fonte: Autoria própria

Quanto à parte de geolocalização, o módulo NEO-6M se mostrou ineficiente, tanto no tempo de conexão inicial com os satélites, quanto na capacidade de se manter conectado com o veículo em movimento. Portanto, como alternativa, foi utilizado o aplicativo gratuito de código-aberto OwnTracker, que tem como principal funcionalidade, fornecer dados do GPS do smartphone via MQTT e HTTP. Dessa forma, cadastramos o aplicativo no tópico de coordenadas, e obtivemos dados assertivos, conforme a relação de coordenadas capturadas no percurso (Figura 24).

## 5. CONCLUSÕES

Criar um protótipo de sistema embarcado integrado ao scanner OBD requer uma grande variedade de ferramentas e conhecimento como observado durante o desenvolvimento do trabalho. Portanto, o trabalho aqui apresentado pode ser considerado satisfatório tendo em vista que o objetivo de desenvolver um sistema embarcado de monitoramento automotivo em tempo real e validar sua funcionalidade foi executado com sucesso.

Analisando os resultados obtidos, podemos concluir que em alguns pontos obtivemos êxito de forma objetiva e clara, porém em outros encontramos algumas dificuldades devido à falta de conhecimento técnico sobre o assunto. Um exemplo disso foi a tentativa de realizar a comunicação com a porta de diagnósticos do veículo, já que não aprofundamos tanto aos seus protocolos durante o curso, com isso dedicamos a maior parte do tempo na busca de métodos para estabelecer a comunicação. Além disso, houveram contratempos quanto às partes de aquisição de consumo de combustível, localização geográfica e disponibilidade de veículos para testes.

Por fim, após o término do desenvolvimento, observamos algumas melhorias para dar continuidade ao projeto para novas aplicações na área, conforme segue:

- a) Implementar o recurso de emissão de relatórios direto na interface do *dashboard*, a fim tornar o sistema mais amigável ao usuário;
- b) Implementação de sincronização entre banco de dados local e remoto automaticamente;
- c) Identificação de imagens pertinentes com uso de redes neurais, a fim de reduzir o número de imagens não conclusivas armazenadas.

## REFERÊNCIAS

AMARASINGHE, M. et al. Cloud-based driver monitoring and vehicle diagnostic with OBD2 telematics. **2015 IEEE International Conference on Electro/Information Technology (EIT)**, maio 2015.

**Código de trânsito brasileiro.** Disponível em: [http://www.planalto.gov.br/ccivil\\_03/leis/19503compilado.htm](http://www.planalto.gov.br/ccivil_03/leis/19503compilado.htm).

Elm Electronics -**Circuits for the Hobbyist OBD to RS232 Interpreter.** [s.l: s.n.]. Disponível em: <https://www.elmelectronics.com/wp-content/uploads/2016/07/ELM327DS.pdf>.

FRÖHLICH, P. et al. **Live traffic camera content in the connected car: Experiences from motorway experiments.** Disponível em: <https://ieeexplore.ieee.org/document/6799826>. Acesso em: 21 jul. 2022.

GALVANI, Asdi et al. **Prototype of microcontroller-based odometer reading for early warning in the vehicle lubricants replacement,** 2015. 3rd International Conference on Information and Communication Technology (ICoICT)

GORLICK, Michael M. **The flight recorder,** 1991. Proceedings of the 1991 ACM/ONR workshop on Parallel and distributed debugging.

GUPTA, G. Sen, et al. **Electronic Instrumentation for Real-time Measurement of Fuel Consumption in EFI Gasoline Engines,** 2008. *IEEE Instrumentation and Measurement Technology Conference.*

HANS, Vaibhav et al. **An Approach to IoT based Car Parking and Reservation system on Cloud,** 2015. International Conference on Green Computing and Internet of Things (ICGCIoT)

MAO, H. et al. Decision-adjusted driver risk predictive models using kinematics information. **Accident Analysis & Prevention,** v. 156, p. 106088, jun. 2021.

MARTINS, I. R.; ZEM, J. L. Estudo dos protocolos de comunicação MQTT e COaP para aplicações machine-to-machine e Internet das coisas. **Revista Tecnológica da Fatec Americana,** v. 3, n. 1, p. 24p.24p., 2015.

MDN CONTRIBUTORS. **Uma visão geral do HTTP,** 2022. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>. Acesso em: 24 jan. 2022.

**MQTT: The Standard for IoT Messaging**, 2020. Disponível em: <https://mqtt.org/>. Acesso em: 24 jan. 2022.

OLIVEIRA, Samuel Silva. **Bancos de dados não-relacionais: um novo paradigma para armazenamento de dados em sistemas de ensino colaborativo**, 2016. Revista da Escola de Administração Pública do Amapá.

**OMS lança Década de Ação pela Segurança no Trânsito 2021-2030 - OPAS/OMS | Organização Pan-Americana da Saúde**. Disponível em: <https://www.paho.org/pt/noticias/28-10-2021-oms-lanca-decada-acao-pela-seguranca-no-transito-2021-2030>.

**RESOLUÇÃO 315/2002 DO CONAMA**. [s.l: s.n.]. Disponível em: [http://conama.mma.gov.br/?option=com\\_sisconama&task=arquivo.download&id=333](http://conama.mma.gov.br/?option=com_sisconama&task=arquivo.download&id=333). Acesso em: 9 jul. 2022.

**RESOLUÇÃO 354/2004 DO CONAMA**. [s.l: s.n.]. Disponível em: [http://conama.mma.gov.br/?option=com\\_sisconama&task=arquivo.download&id=447](http://conama.mma.gov.br/?option=com_sisconama&task=arquivo.download&id=447). Acesso em: 9 jul. 2022.

RIBEIRO, H. M. G. Dados CAN / OBD2 em tempo real de viaturas auto. **repositorio-aberto.up.pt**, 24 jul. 2015.

RUENGITTINUN, Somchoke et al. **IoT solution for bad habit of car security**, 2017. 10th International Conference on Ubi-media Computing and Workshops.

SÁNCHEZ CARRIZO, J. GITT. Simulador de una ECU y diagnóstico mediante CAN y OBD-II. **ruidera.uclm.es**, 2017.

SIM, A. X. A.; SITO HANG, B. OBD-II standard car engine diagnostic software development. **2014 International Conference on Data and Software Engineering (ICODSE)**, nov. 2014.

SIMONS-MORTON, B. G. et al. Trajectories of kinematic risky driving among novice teenagers. **Accident Analysis & Prevention**, v. 51, p. 27–32, mar. 2013.

VIJAY, E. Vargil, et al. **Electronic control unit for an adaptive cruise control system and engine management system in a vehicle using electronic fuel injection**, 2010. INTERACT-2010

WANG, Kunfeng et al. **An Embedded System for Vision-based Driving Environment Perception**, 2006. *2nd IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications*



---

Emitido em 08/08/2022

**CÓPIA DO TRABALHO Nº 110/2022 - DELMAX (11.57.05)**

**(Nº do Protocolo: NÃO PROTOCOLADO)**

*(Assinado digitalmente em 09/08/2022 19:30 )*

LEANDRO RESENDE MATTIOLI  
PROFESSOR ENS BASICO TECN TECNOLOGICO  
DELMAX (11.57.05)  
Matrícula: 2973183

Para verificar a autenticidade deste documento entre em <https://sig.cefetmg.br/documentos/> informando seu número:  
**110**, ano: **2022**, tipo: **CÓPIA DO TRABALHO**, data de emissão: **09/08/2022** e o código de verificação:  
**8a5d38549c**