

Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Engenharia Eletrônica

**Desenvolvimento de protótipo de volante para  
veículo tipo Formula SAE integrado com  
central de injeção eletrônica**

Autor: Davi Dantas Lopes

Orientador: Prof. Me. Rafael Rodrigues da Silva

Coorientador: Prof. Dr. Evandro Leonardo Silva Teixeira

Brasília, DF

2023



Davi Dantas Lopes

**Desenvolvimento de protótipo de volante para veículo  
tipo Formula SAE integrado com central de injeção  
eletrônica**

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica .

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Me. Rafael Rodrigues da Silva

Coorientador: Prof. Dr. Evandro Leonardo Silva Teixeira

Brasília, DF

2023

---

Davi Dantas Lopes

Desenvolvimento de protótipo de volante para veículo tipo Formula SAE integrado com central de injeção eletrônica/ Davi Dantas Lopes. – Brasília, DF, 2023-70 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Me. Rafael Rodrigues da Silva

Coorientador: Prof. Dr. Evandro Leonardo Silva Teixeira

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA , 2023.

1. CAN, Formula SAE, ESP32, ESP-NOW. I. Prof. Me. Rafael Rodrigues da Silva. II. Prof. Dr. Evandro Leonardo Silva Teixeira. III. Universidade de Brasília. IV. Faculdade UnB Gama. VI. Desenvolvimento de protótipo de volante para veículo tipo Formula SAE integrado com central de injeção eletrônica

CDU 02:141:005.6

---

Davi Dantas Lopes

# **Desenvolvimento de protótipo de volante para veículo tipo Formula SAE integrado com central de injeção eletrônica**

Monografia submetida ao curso de graduação em Engenharia Eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletrônica .

Trabalho aprovado. Brasília, DF, 20 de dezembro de 2023:

---

**Prof. Me. Rafael Rodrigues da Silva**  
Orientador

---

**Prof. Dr. Evandro Leonardo Silva  
Teixeira**  
Coorientador

---

**Prof. Dr. Guillermo Alvarez Bestard**  
Convidado 1

---

**Prof. Dr. Bruno Luiz Pereira**  
Convidado 2

Brasília, DF  
2023

# Agradecimentos

Gostaria de expressar meus sinceros agradecimentos às Marias da minha vida, minhas mães, Lucia e Helena, que enfrentaram inúmeras adversidades para garantir que eu pudesse apresentar este trabalho. Apesar dos desafios velados que enfrentamos, a perseverança delas foi fundamental. Aos meus irmãos, que trilharam este caminho comigo, compartilhando não apenas as dificuldades, mas também os momentos de alegria. Às vezes, questiono o propósito de tudo isso, mas percebo que é para poder compartilhar com vocês as conquistas, obrigado Anne e Everaudo.

Um agradecimento especial à minha namorada Amanda, que esteve ao meu lado durante todo o curso, mostrando-me que sou capaz. Ter uma mulher forte como apoio faz toda a diferença, não posso ser menos para alguém que é tudo para mim. Expresso minha gratidão às figuras paternas e suas famílias presentes em minha vida, querido Tio Sérgio e Tia Neide, Tio Rodrigo e Tia Gabriela, Tio Reinaldo e Tia Janaía, Renata e Seu Naim. Cada um deles contribuiu para moldar meus gostos e *hobbies* através de atos de solidariedade e ensinamentos. Agradeço também a família que me acolheu e deu espaço para realizar estudos e também este trabalho, muito obrigado Milson e Suely.

Não posso esquecer dos amigos que encontrei ao longo dessa jornada. Agradecimentos aos amigos da vida João Victor, Gabriel Súdre e Pedro Daia e também aos do primeiro semestre, em especial o Vilela, Aldo e Bernardo, aos companheiros da UnBaja, especialmente a Catarina Claudino, Victor Martins e Lucas Torres, e aos amigos da Apuama Racing, em especial o João Pedro, Mateus Ramalho, Bruno Fazzini e Rafael Marques. Cada um de vocês deixou uma marca significativa em minha vida.

Meus agradecimentos também se estendem aos meus professores de base Olegário Neto, Flávia Santos, Carlos Bairy, Lorena e Marly e também aos orientadores deste trabalho Me. Rafael Rodrigues e Dr. Evandro Teixeira, cujo suporte e conhecimento foram fundamentais para a construção e desenvolvimento do projeto. A todos, meu muito obrigado por fazerem parte desta trajetória e contribuírem para o meu crescimento pessoal e acadêmico.

*“Desista de me fazer desistir!”*

*(Naruto Uzumaki)*

*“Nenhum esforço faz sentido, se você não acredita em si mesmo”*

*(Maito Gai)*

# Resumo

A equipe de competição Apuama Racing é uma das muitas equipes universitárias que participam da competição Formula SAE Brasil, que tem como objetivo desafiar estudantes de engenharia a projetar e construir um carro de corrida tipo Fórmula, com alta performance e eficiência energética. Um dos principais componentes do carro de corrida é o volante, que não só permite ao motorista controlar o veículo, mas também fornece informações cruciais sobre o desempenho do carro. É por isso que a equipe decidiu desenvolver um volante instrumentalizado que possui comunicação com a central de injeção eletrônica do veículo FuelTech modelo FT600. O projeto se baseia então no desenvolvimento do protótipo de duas centrais, uma estabelece a troca de mensagens por meio do protocolo CAN com o módulo de injeção e a outra fica responsável pelo controle do display e interface do volante.

**Palavras-chaves:** CAN, Formula SAE, ESP32, ESP-NOW.

# Abstract

The Apuama Racing team is one of many university teams participating in the Formula SAE Brazil competition, which aims to challenge engineering students to design and build a high-performance, energy-efficient Formula-type racing car. One of the main components of the race car is the steering wheel, which not only allows the driver to control the vehicle, but also provides crucial information about the car's performance. This is why the team decided to develop an instrumented steering wheel that has communication with the electronic injection center of the FuelTech vehicle model FT600. The project is based on the development of a prototype of two central offices, one establishing the exchange of messages through CAN protocol with the module, and the other responsible for controlling the steering wheel display and interface.

**Key-words:** CAN, Formula SAE, ESP32, ESP-NOW, FuelTech.

# Lista de ilustrações

Figura 1 – AF22 durante a realização de provas na competição Formula SAE 2022.	15
Figura 2 – Vista do piloto - AF22.	16
Figura 3 – Painel de controle McLaren MP4/6.	21
Figura 4 – Volante Mercedes Benz F1 2015.	22
Figura 5 – Exemplo de MAPA 2D.	24
Figura 6 – Exemplo de MAPA 3D.	24
Figura 7 – FuelTech600	25
Figura 8 – Arquitetura ESP32	27
Figura 9 – Conexão entre os dispositivos Master-Slave de uma rede SPI.	28
Figura 10 – Conexão entre os dispositivos nas comunicação UART.	29
Figura 11 – Conexão entre os dispositivos nas comunicação I <sup>2</sup> C.	29
Figura 12 – Arquitetura Centralizada e Distribuída.	31
Figura 13 – Relação entre o modelo OSI e os padrões ISO 11898 (CAN) e ISO 11519 (LIN).	32
Figura 14 – Diagrama de Nível Lógico do Barramento CAN.	32
Figura 15 – Método de arbitragem para priorização de acesso no barramento.	33
Figura 16 – Campos de um <i>frames</i> de dados CAN 2.0A.	34
Figura 17 – Descrição de cada campo de um <i>frames</i> CAN 2.0A.	34
Figura 18 – Comparativo de latência dos principais protocolos de comunicação do ESP32.	37
Figura 19 – Comparativo de alcance dos principais protocolos de comunicação do ESP32.	37
Figura 20 – Microcontrolador ESP32.	39
Figura 21 – Conexão das centrais com a Injeção	40
Figura 22 – Arquitetura da central Sniffer.	40
Figura 23 – Módulo de comunicação CAN MCP2515.	41
Figura 24 – Arquitetura da central Volante.	41
Figura 25 – Display Nextion NX4827P043.	42
Figura 26 – Circuito divisor de tensão e diagrama da chave seletora.	43
Figura 27 – Acelerômetro e giroscópio MPU6050.	44
Figura 28 – Esquemático dos circuitos de cada central	45
Figura 29 – Placas de circuito impresso desenvolvidas utilizando o <i>software EasyEDA</i> .	45
Figura 30 – <i>frames</i> extraídos do barramento CAN-D FT600 utilizando o MCP2515	46
Figura 31 – Estrutura do campo de identificação FTCAN 2.0	47
Figura 32 – Estrutura de <i>frames</i> do protocolo FTCAN 2.0.	47
Figura 33 – Campo de dados do tipo <i>Standart</i> .	47

Figura 34 – Campo de dados do tipo <i>Segmented</i> . . . . .	48
Figura 35 – Mensagens fixas de transmissão. . . . .	48
Figura 36 – Exemplo de estratificação de um <i>frames</i> de broadcast FT CAN 2.0. . . . .	48
Figura 37 – Exemplo de estratificação de <i>frames</i> segmentados. . . . .	49
Figura 38 – Tabela de <i>MeasureID</i> . . . . .	49
Figura 39 – Interface de configuração do <i>Switch Panel</i> no <i>FT Manager</i> . . . . .	50
Figura 40 – Dispositivo <i>SwitchPanel</i> 8 Mini. . . . .	50
Figura 41 – Interface de desenvolvimento <i>Nextion Editor</i> . . . . .	52
Figura 42 – Interface do piloto. . . . .	53
Figura 43 – Fluxograma do firmware da central Sniffer. . . . .	53
Figura 44 – Fluxograma do firmware da central Volante. . . . .	55
Figura 45 – Placas fabricadas: Sniffer(direita) e Volante(esquerda). . . . .	57
Figura 46 – Sniffer montado junto com os componentes do sistema. . . . .	57
Figura 47 – Consumo central volante. . . . .	58
Figura 48 – Autonomia central volante. . . . .	59
Figura 49 – Mapeamento dos botões no volante. . . . .	59
Figura 50 – Interface do usuário para a exibição de dados e gráficos de telemetria. . . . .	60
Figura 51 – Telas de alertas para o piloto. . . . .	61
Figura 52 – Montagem e instalação dos componentes da central no invólucro. . . . .	61
Figura 53 – Posicionamento final da central no volante. . . . .	62
Figura 54 – Decodificação de <i>frames</i> FT CAN2.0. . . . .	62
Figura 55 – Dados exibidos na FT600 e na central Volante. . . . .	63
Figura 56 – Amostragem gráfica dos valores de RPM e TPS. . . . .	63
Figura 57 – FT600 reconhece a presença do <i>Switch Panel</i> . . . . .	64
Figura 58 – Mapeamento dos botões do <i>Switch Panel</i> . . . . .	64
Figura 59 – Exibição dos dados simulados no volante e menu de simulação FT Ma- nager. . . . .	65
Figura 60 – Conexão entre LoRa e49 e Raspberry Pi. . . . .	65
Figura 61 – Dados recebidos pelo LoRa e49. . . . .	65

# Lista de tabelas

Tabela 1 – <i>frames</i> de envio do ESP-NOW . . . . .	36
Tabela 2 – Modos de operação LoRa EBYTE E49. . . . .	38
Tabela 3 – Posição da chave seletora com sua saída em tensão. . . . .	42
Tabela 4 – Tabela de consumo dos componentes do volante. . . . .	45
Tabela 5 – Utilização de periféricos do ESP32. . . . .	57
Tabela 6 – Mapeamento das funções dos botões. . . . .	60

# Lista de abreviaturas e siglas

CAN	Controller Area Network
ECU	Eletronic Control Unit
BMS	Battery Management System
UART	Universal Asynchronous Receiver/Transmitter
SAE	Society of Automotive Engineers
LCD	Liquid Crystal Display
OLED	Organic Light-Emitting Diodes
TCP	Transmission Control Protocol
FIA	Fédération Internationale de l'Automobile
RPM	Rotações Por Minuto
BLE	Bluetooth Low Energy
IoT	Internet of Things
RTOS	Real Time Operational System

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
1.1	Objetivos	17
1.2	Organização do trabalho	17
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>18</b>
2.1	Estado da Arte	18
2.2	Formula SAE	19
2.3	Volantes para veículos de competição.	20
2.4	Central de Injeção Eletrônica	22
2.4.1	FuelTech600	24
2.5	Sistema Embarcado	26
2.5.1	Microcontrolador	26
2.5.1.1	SPI	28
2.5.1.2	UART	29
2.5.1.3	I <sup>2</sup> C	29
2.6	Rede CAN	29
2.6.1	Conceito	30
2.6.2	Funcionamento	31
2.6.2.1	Camada Física	31
2.6.2.2	Camada de Enlace	33
2.6.2.3	Detecccão de erros	33
2.7	ESP-NOW	35
2.8	Long Range	37
<b>3</b>	<b>METODOLOGIA</b>	<b>39</b>
3.1	Arquitetura Eletroeletrônica	39
3.2	Aquisição de dados FT CAN2.0	46
3.3	Escrita FT CAN2.0	48
3.4	Transmissão ESP-Now	50
3.5	Display TFT Nextion	51
3.6	Firmware	52
<b>4</b>	<b>RESULTADOS</b>	<b>56</b>
4.1	Central Sniffer	56
4.2	Volante	58
4.3	Integração	62

<b>5</b>	<b>CONCLUSÃO</b> . . . . .	<b>66</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>68</b>

# 1 Introdução

Coordenada pela SAE (*Society of Automotive Engineers*) e destinada a estudantes, a Formula SAE tem como objetivo projetar e construir um veículo de corrida de alto desempenho. O evento ocorre em vários países ao redor do mundo, incluindo Brasil, Estados Unidos, Alemanha e Japão. Os estudantes do ensino superior envolvidos trabalham em equipes multidisciplinares que incluem integrantes dos cursos de engenharia automotiva, mecânica, elétrica, eletrônica, software e outras áreas relevantes. Durante o processo de desenvolvimento do veículo, a equipe deve seguir um conjunto rigoroso de regras estabelecidas pela competição que abrangem desde a construção do chassi até a escolha dos motores e sistemas de freios e suspensão.

Após finalizada a construção do protótipo do veículo, as equipes participam de uma série de eventos, incluindo testes de desempenho dinâmico, avaliações de segurança, testes de eficiência de combustível e apresentações técnicas para um painel de juízes. O objetivo final da competição Formula SAE é fornecer aos estudantes uma oportunidade única para aplicarem seus conhecimentos em um ambiente prático e desafiador, enquanto desenvolvem habilidades técnicas valiosas como trabalho em equipe e solução de problemas complexos. Além disso, a competição oferece uma oportunidade para os estudantes se conectarem com outros profissionais da área durante a realização das provas. Esse contato também proporciona um ambiente propício para troca de ideias, discussões técnicas e *feedback* construtivo, que podem contribuir significativamente para o crescimento e desenvolvimento dos estudantes como profissionais da área.

Outro aspecto importante da competição Formula SAE é a ênfase na inovação e no pensamento criativo. Os estudantes são encorajados a pensar de maneira divergente e criar soluções criativas para os desafios técnicos que enfrentam durante o processo de construção do carro. Neste contexto, a equipe da Universidade de Brasília, Apuama Racing, fundada em 2004 pelos integrantes dos cursos de engenharias da UnB, participa da etapa nacional Brasil da competição. O modelo atual da equipe Apuama Racing é o Formula SAE AF22, Figura 1, décimo protótipo desenvolvido pela equipe onde diversos subsistemas veiculares, incluindo sistemas de suspensão, aerodinâmica, elétrico e direção foram projetados e fabricados pelos integrantes. O protótipo possui um conjunto de potência composto por um motor CB600F, encontrado em motos modelo Honda CB600 Hornet. O gerenciamento eletrônico do motor é realizado pela central de injeção eletrônica FuelTech FT600, da empresa brasileira FuelTEch. O sistema de direção é composto por rodas, caixa de direção, coluna de direção e um volante laminado em fibra de vidro. É importante ressaltar que o sistema de direção não possui integração com a eletrônica do carro como demonstra a Figura 2, o que significa que não é possível ao condutor do

veículo visualizar informações em tempo real diretamente no volante dos valores dos sensores presentes no veículo como por exemplo sensor de velocidade e sensor de temperatura do bloco do motor. Além de não conseguir visualizar tais informações, o piloto também não tem a opção de alterar parâmetros de calibração seja do motor ou outros subsistemas uma vez que não há uma interface entre volante e a central de injeção eletrônica.



Figura 1 – AF22 durante a realização de provas na competição Formula SAE 2022.

Fonte: Autor.

A equipe tem como um dos objetivos para o próximo modelo do projeto o desenvolvimento de um volante instrumentado que seja capaz de realizar ajustes na central de controle do motor e de exibir dados do carro, proporcionando ao piloto um entendimento mais claro dos indicadores de segurança e desempenho do veículo. Essa abordagem é amplamente utilizada nas categorias profissionais de corridas automotivas, como a *Formula 1* e a *Formula Indy*. Nestes casos, o volante instrumentado pode incluir um display *LCD* que mostra informações importantes, como velocidade, rotação do motor, temperatura dos pneus e pressão do óleo. Os botões estrategicamente posicionados no volante permitem ao piloto ajustar parâmetros cruciais, como o controle de tração, a distribuição de freio e a potência do motor. Essa interface direta e personalizada no volante oferece ao piloto um acesso rápido e intuitivo às informações e controles necessários durante a corrida, permitindo-lhe tomar decisões rápidas e ajustar o desempenho do veículo de acordo com as necessidades de cada momento.

Ante o exposto, a implementação de uma solução semelhante pode trazer diversos benefícios para um veículo Formula SAE, como melhoria no desempenho através de ajustes em tempo real, comunicação eficiente entre sistemas veiculares, telemetria aprimorada para análise de dados e decisões estratégicas, além de oferecer maior personalização e controle ao piloto, otimizando a experiência de condução e a performance em provas. No que diz respeito aos componentes desta solução, destacam-se a injeção eletrônica



Figura 2 – Vista do piloto - AF22.

Fonte: Autor.

utilizada pela equipe, o microcontrolador ESP32 como núcleo central das centrais a serem desenvolvidas, módulos de comunicação CAN (Controller Area Network) para a extração dos dados de sensores e uma tela de exibição das informações para o piloto. A injeção eletrônica desempenha um papel crucial no gerenciamento do motor, controlando a mistura ar-combustível e a ignição para garantir um desempenho eficiente. Através da programação de diferentes mapas de injeção e estratégias de funcionamento, é possível adaptar o motor para diferentes situações, como alta performance ou economia de combustível. O módulo de comunicação CAN atua como um ponto de coleta para os dados gerados pelos vários sensores espalhados pelo veículo. Esses dados são transmitidos através do barramento CAN pela unidade de injeção eletrônica em um formato proprietário do fabricante, possibilitando a exibição de informações relevantes no volante e a realização de ajustes em tempo real. O microcontrolador ESP32 atua como o núcleo central do sistema, realizando a integração e processamento dos dados dos sensores, a comunicação com os módulos CAN e a interação com o volante instrumentado. Além disso, ele pode ser programado para enviar dados em tempo real para os boxes utilizando um transmissor de longa distância, permitindo que a equipe monitore e analise o desempenho do veículo durante testes e competições.

Essa abordagem completa e integrada proporcionará aos membros da equipe uma vantagem competitiva, permitindo que eles otimizem o desempenho do carro, realizem ajustes em tempo real e consigam visualizar diversas informações importantes de monitoramento do estado do veículo.

## 1.1 Objetivos

O objetivo geral deste trabalho é desenvolver um protótipo de volante para um veículo do tipo Formula SAE que se comunique com a central de injeção eletrônica do carro e com a equipe presente no box. Os objetivos específicos do trabalho são:

- Desenvolver um algoritmo para leitura do protocolo FT CAN 2.0;
- Desenvolver uma central capaz de adquirir dados e escrever sinais na rede CAN;
- Desenvolver central de painel para exibição de dados no volante;
- Desenvolver sistema de telemetria para para transmissão dos dados coletados para equipe nos boxes.

## 1.2 Organização do trabalho

Este trabalho será dividido em capítulos, cada um com um tema principal e objetivos específicos que serão abordados ao início de cada capítulo. Espera-se que os capítulos tenham a seguinte distribuição:

- Capítulo 2: Neste capítulo, abordaremos os conceitos teóricos essenciais relacionados aos temas deste trabalho. Apresentaremos uma base teórica abrangente para elucidar os procedimentos e aplicações envolvidos neste projeto. Discutiremos os fundamentos teóricos e forneceremos equações matemáticas relevantes. Os principais tópicos explorados neste capítulo incluem: sistemas de transmissão digitais, sistemas embarcados e introdução à rede CAN.
- Capítulo 3: Neste capítulo, será apresentada a metodologia utilizada, evidenciando os métodos e as ferramentas.
- Capítulo 4: Neste capítulo, serão apresentados e discutidos os resultados obtidos.
- Capítulo 5: Neste capítulo, será apresentada a conclusão sobre este trabalho.

## 2 Fundamentação Teórica

Este capítulo tem como objetivo apresentar os principais estudos, teorias e aplicações consolidadas sobre temas e tecnologias relacionadas à implementação de um volante com exibição de informações, que permita a extração de dados da injeção eletrônica de um veículo de corrida. Para isso, serão abordados conceitos sobre sistemas de aquisição de dados, especialmente aqueles que utilizam a rede CAN, que é amplamente utilizada em sistemas automotivos. Também serão discutidos tópicos relevantes para o desenvolvimento de um volante inteligente, como sensores, microcontroladores, sistemas embarcados e técnicas de processamento de dados.

### 2.1 Estado da Arte

Diversos trabalhos têm focado na integração de interfaces entre o piloto e o sistemas do veículo por meio do volante. Essas tecnologias desempenham um papel crucial no avanço dos volantes automotivos, criando ambientes que podem proporcionar ao piloto informações fundamentais do estado do veículo, bem como permitir alterações no comportamento do veículo por meio do acesso a parâmetros que podem ser alterados via interface.

O estudo de [PIMPLE\(2018\)](#) tem como objetivo o desenvolvimento de um sistema de aquisição de dados de uma rede CAN utilizando o módulo eletrônico MCP2515 em conjunto com um Arduino Uno para extração de informações de veículos. Utilizou-se uma central Vector CANoe 9.0 para simular a troca de mensagens que seria realizada com uma central eletrônica automotiva e o software CANCase XL Log para a geração de dados. As mensagens recebidas do dispositivo podem ser utilizadas como indicadores de performance ou sinalizadores de falhas permitindo assim realizar um diagnóstico. Como resultados o estudo conseguiu realizar a leitura em tempo real de informações pelo protocolo CAN.

Abordando o estudo de [ATANASOVSKI\(2021\)](#), sistemas de medição e aquisição de dados são amplamente utilizados em aplicações distintas como controle de processos, medição de qualidade e gestão de tráfego. Desta forma, o armazenamento de dados localmente é um problema comum entre as aplicações, sendo assim o autor sugere a transmissão dos dados por uma comunicação sem fio. O protocolo ESP-NOW desenvolvido pela Espressif como um protocolo de baixa latência pode resolver problemas de transmissão de dados em nós de sensores em ambientes onde existe uma escassez de rede local sem fio. O design robusto, baixo consumo energético ampla disponibilidade de pinos de controle e variedade de protocolos de comunicação, são características ideias que tornam o microcontrolador ESP32 uma opção ideal para sistemas de aquisição de dados em tempo

real.

De acordo com [ELVANES\(2022\)](#), à medida que as tecnologias avançam, novos desenvolvimentos são feitos no design de veículos para competição de Fórmula SAE, há cada vez mais dados e informações para que os engenheiros possam ajustar seus veículos e alcançarem um melhor desempenho. No trabalho realizado, o autor desenvolve um projeto de central eletrônica para um volante instrumentado sem fio baseado na comunicação *Bluetooth Low Energy* para evitar retrabalho na parte mecânica de acoplamento do volante. Como resultado o estudo obteve uma central eletrônica com um display OLED de 20x4 caracteres controladores pelo microcontrolador MDBT42Q-512KV2 que possui integrado o módulo de conexão BLE, alimentada por um banco de baterias de polímero lítio Philips18650, desenvolvida para a instalação na estrutura de direção do veículo e que possui integração com a unidade de controle do motor.

Com o objetivo de estabelecer um sistema de aquisição e transmissão de dados de um veículo de competição elétrico, [CASADEVALL\(2016\)](#) desenvolveu uma central que realiza a leitura das informações no barramento CAN que transitam entre as centrais instaladas no veículo. Para esse fim, foi selecionado o microcontrolador PIC32MX795F512H como o dispositivo central do sistema, responsável por gerenciar o módulo de comunicação sem fio *WifiFly* RN131C, que utiliza a interface UART para receber mensagens e as envia utilizando o protocolo TCP. Além disso, utilizando o *Android Studio*, foi desenvolvida uma aplicação *Android* que exibe as informações em tempo real por meio de gráficos em um dispositivo móvel, permitindo uma análise mais eficiente e tomada de decisões rápidas durante os testes.

Neste sentido diversos autores têm focado no desenvolvimento de interfaces para monitoramento e atuação em centrais eletrônicas. Estas pesquisas são fundamentais para avançar no campo da telemetria automotiva, buscando soluções que proporcionem informação em tempo real e permitam ajustes dinâmicos para otimizar o desempenho e a segurança dos veículos.

## 2.2 Formula SAE

A Formula SAE (FSAE) é uma competição de engenharia estudantil que desafia equipes universitárias a projetar, construir e competir com veículos de corrida de alta performance nas categorias com propulsão a combustão e elétrica. O objetivo principal da FSAE é proporcionar aos estudantes a oportunidade de aplicar seus conhecimentos teóricos em um ambiente prático, desenvolvendo habilidades técnicas, trabalho em equipe e gerenciamento de projetos ([BRASIL, 2023](#)).

Durante a competição FSAE, as equipes são avaliadas em diversas categorias, incluindo design do veículo, desempenho em pista, eficiência de combustível e apresentação

de projeto. Os carros devem ser projetados levando em consideração critérios de segurança, performance e inovação tecnológica, além de atenderem o regulamento estabelecido pela organização.

Os carros projetados para a competição Formula SAE no Brasil devem atender às rigorosas especificações de motorização, conforme estabelecido pelo regulamento. Os motores devem ser do tipo pistão com ciclo primário de quatro tempos e ter um deslocamento combinado total igual ou inferior a 710 centímetros cúbicos por ciclo [INTERNATIONAL\(2022\)](#), que são comumente encontrados em motocicletas. Os motores podem ser modificados dentro das restrições das regras, o que permite aos estudantes aplicar conhecimentos de engenharia para otimizar a performance dentro das limitações estabelecidas.

O chassi dos veículos Formula SAE é projetado seguindo normas rigorosas estabelecidas pelo regulamento da competição. O regulamento especifica critérios detalhados sobre a estrutura do chassi, incluindo sua segurança, dimensões, materiais permitidos e aspectos de design. Esses requisitos garantem a integridade estrutural do chassi, priorizando a segurança do piloto. Os materiais escolhidos para a construção, como ligas de alumínio ou fibra de carbono, são selecionados para equilibrar leveza e resistência de acordo com o orçamento disponível. Além disso, o chassi deve ser capaz de acomodar todos os sistemas mecânicos e elétricos do veículo, mantendo uma distribuição de peso otimizada para desempenho eficiente.

A competição FSAE proporciona aos estudantes uma experiência única no desenvolvimento de veículos de alto desempenho, colocando-os em um ambiente semelhante ao da indústria automotiva. A participação nessa competição promove o aprendizado prático, estimula a criatividade e incentiva a busca por soluções inovadoras. Além disso, a FSAE oferece uma oportunidade valiosa para *networking* com profissionais da indústria e outras equipes universitárias de todo o país.

## 2.3 Volantes para veículos de competição.

Segundo [BROWN\(2020\)](#), nos últimos 20 anos, o volante automotivo do esporte a motor sofreu diversas alterações, o que antes era apenas um instrumento para controle do veículo evoluiu para uma interface complexa com múltiplos controles utilizados para melhoria de performance. Ainda segundo o autor, no campeonato de Fórmula 1 de 1991, o carro vencedor possuía 4 opções de controles disponíveis no painel para interação do piloto com o veículo por meio de botões, como demonstra a Figura 3, esse número aumentou para 9 em 1998 e 20 em 2001. Desde então, a quantidade média de controles, incluindo chaves seletoras e botões, variou entre 14 e 25, refletindo a crescente complexidade e a importância da interface do volante na interação do piloto com o veículo.

O formato do volante também passou por mudanças, deixou de ter o formato



Figura 3 – Painel de controle McLaren MP4/6.

Fonte: (M.P.H., 2018)

convencional circular e adotou-se o um formato retangular para permitir uma visão aprimorada da pista por cima do volante uma vez que a posição de assento ser baixa na categoria. Nota-se que nos volantes modernos, a quantidade de controles não reflete diretamente na quantidade de funcionalidades disponíveis, é possível que certos comandos sejam realizados por controles individuais e outros por combinações de controles como o acionamento de dois botões diferentes, por exemplo. Além disso, o display de exibição, que é regulamentado pela FIA (Fédération Internationale de l'Automobile), tem a capacidade de exibir até 100 páginas de informações variadas e relevantes relacionadas ao veículo. Essas páginas podem incluir dados sobre o desempenho do motor, a telemetria em tempo real, informações sobre a temperatura do motor, pressão dos pneus, dados de combustível, tempos de volta, entre outros aspectos importantes para o monitoramento e análise do veículo durante a corrida.

Essa complexidade dos volantes pode ser justificada com base nos benefícios que trouxe ao permitir a troca de configuração de gerenciamento do motor e extrair o máximo desempenho de acordo com a natureza da competição. Mais da metade dos controles em um volante de um carro de Fórmula 1 moderno é dedicada à otimização de desempenho. A Figura 4 demonstra o volante utilizado na categoria, três diferentes tipos de interfaces de controle podem ser notadas na Figura: doze botões, três encoders rotativos e três encoders de roda de polegar. No centro está instalado um display de cristal líquido e LEDs que indicam a rotação por minuto do motor. Nota-se o uso intensivo de adesivos coloridos para identificar a funcionalidade de cada controle, com a maioria dessas funções influenciando diretamente no desempenho do veículo.

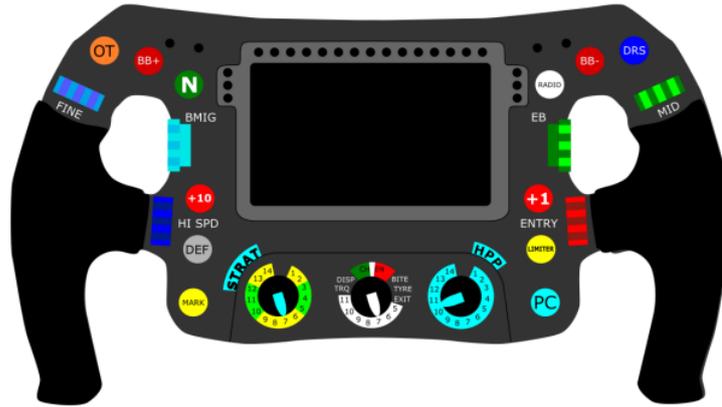


Figura 4 – Volante Mercedes Benz F1 2015.

Fonte: (BROWN, 2020)

Ao longo dos anos, os volantes automotivos esportivos evoluíram significativamente, tornando-se centros de controle cada vez mais complexos com a adição de controles avançados e tecnologias de exibição de informações, eles se tornaram peças essenciais na busca por um desempenho automotivo aprimorado. Uma das áreas cruciais em que essas informações se baseiam é a injeção eletrônica do motor, um sistema sofisticado que gerencia o fornecimento de combustível e o desempenho do motor.

Através dos controles integrados ao volante, os pilotos podem realizar ajustes nos parâmetros da injeção eletrônica, como a troca de mapa ou estratégia de gerenciamento do motor. Essa interação direta é fundamental para a otimização da performance do veículo, permitindo modificações imediatas e estratégicas sem a necessidade de intervenção externa ou paradas para configuração, mantendo o foco do piloto na condução.

## 2.4 Central de Injeção Eletrônica

Conforme mencionado por [NASCIMENTO\(2019\)](#), o sistema de injeção eletrônica realiza o controle da mistura ar-combustível em um motor de combustão interna, bem como o ajuste do ponto de ignição de acordo com as condições operacionais do motor. Esse sistema é supervisionado pela Unidade de Controle eletrônico do Motor (ECM), que atua como uma central recebendo informações dos sensores, processando os dados recebidos e acionando os atuadores mecânicos e elétricos de acordo com a programação de funcionamento configurada.

Os sensores desempenham um papel crucial no fornecimento de dados de entrada para a ECM [NASCIMENTO\(2019\)](#), sendo os principais sensores encontrados no motor de um veículo são: sensores de temperatura (como temperatura do líquido de arrefecimento do motor, temperatura ambiente e temperatura do coletor de admissão), sensor de fluxo de ar, sensor de pressão de admissão, sensor de posição do corpo de borboleta, sensor de

posição do pedal do acelerador, sensor de detonação, sensor de rotação do motor, sensor de fase de comando de válvulas e sonda lambda. Esses dispositivos eletrônicos convertem parâmetros físicos do funcionamento do motor, como pressão, temperatura e rotação, em sinais elétricos que são interpretados pela unidade de controle e mantêm a precisão de controle do sistema.

Por outro lado os atuadores recebem os sinais processados pela ECM e atuam de forma a manter o funcionamento adequado do motor em diversas condições. Eles são responsáveis por regular a relação combustível-ar na admissão por meio das válvulas injetoras, gerenciar o momento de ignição por meio das bobinas de ignição e velas, e controlar o atuador de marcha lenta para manter o funcionamento do motor quando está em marcha mínima. Esses componentes são controlados pelos sinais enviados pela unidade de controle, garantindo o funcionamento ideal do motor (NASCIMENTO, 2019).

O adequado desempenho dos sensores e atuadores é crucial para alcançar um rendimento térmico ideal do motor, assegurando alta eficiência, redução nas emissões de gases nocivos, economia de combustível e uma operação suave. Os parâmetros como o tempo de injeção de combustível e o avanço da ignição são armazenados na ECU em forma de "mapas", disponíveis em versões 2D e 3D. A escolha entre mapas 2D e 3D é determinada pelas exigências específicas do motor e pela complexidade do sistema de injeção eletrônica, bem como pela competência do calibrador para ajustar cada tipo de mapa, que são visualizados em tabelas ou matrizes.

- 2D: São mapas mais simples e bidimensionais. Consistem em uma tabela que relaciona dois parâmetros, geralmente a rotação do motor (RPM) e a carga do motor (geralmente representada pela pressão no coletor de admissão). Cada célula da tabela contém um valor correspondente, como o tempo de injeção de combustível ou o avanço de ignição, para as combinações específicas de RPM e carga. Esses mapas são mais fáceis de serem programados e oferecem uma visão simplificada do comportamento do motor em diferentes regimes de operação.
- 3D: Os mapas 3D são mais complexos e tridimensionais. Eles incorporam uma terceira dimensão além do RPM e carga, que pode ser a temperatura do motor, a pressão de admissão ou outros parâmetros relevantes. Assim, a tabela é estendida para incluir valores adicionais para cada combinação de RPM, carga e terceiro parâmetro. Essa representação mais detalhada permite ajustes mais precisos e refinados do tempo de injeção e do avanço de ignição, levando em consideração as variações das condições de funcionamento do motor.

No contexto da injeção eletrônica automotiva, um avanço notável é representado pela injeção comercial eletrônica FuelTech600 (FT600), oferecendo recursos avançados e

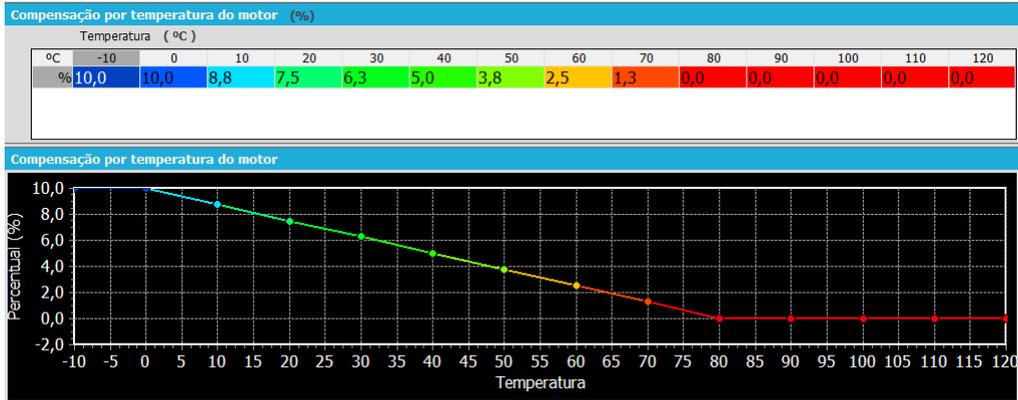


Figura 5 – Exemplo de MAPA 2D.

Fonte: Autor.

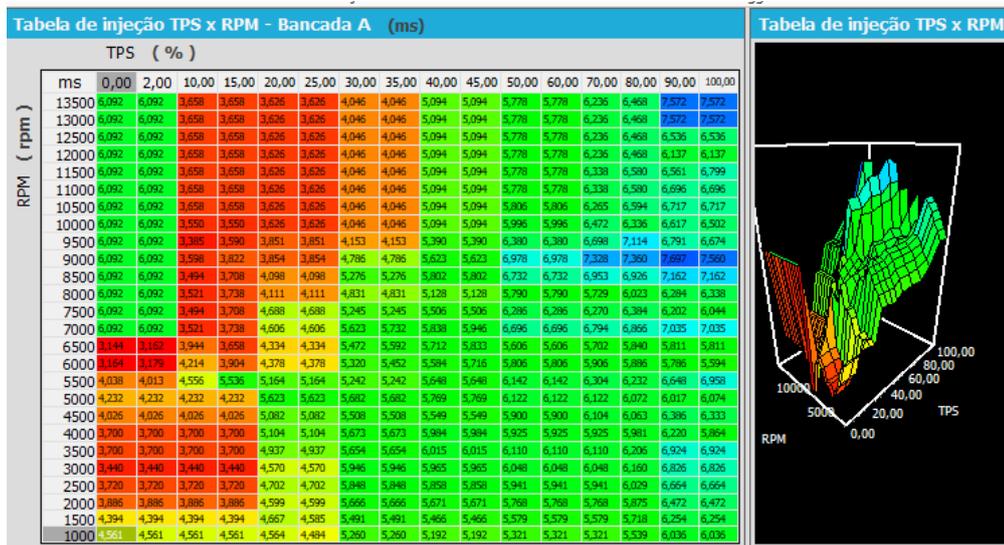


Figura 6 – Exemplo de MAPA 3D.

Fonte: Autor.

capacidades aprimoradas de ajuste e otimização. Desenvolvida para oferecer uma gama abrangente de ajustes finos e opções de otimização, a FT600 permite que engenheiros e pilotos possam monitorar e controlar diversos parâmetros do motor em tempo real maximizando a potência, a eficiência e a confiabilidade do veículo nas pistas.

### 2.4.1 FuelTech600

A FT600 é um modelo avançado de injeção eletrônica programável que oferece uma variedade de recursos e funcionalidades. Seu design inclui uma tela sensível ao toque para uma interação intuitiva com o sistema, bem como conectores de 34 pinos que permitem a conexão de sensores e atuadores. Além disso, o dispositivo possui LEDs que fornecem uma exibição gráfica das rotações do motor, proporcionando aos pilotos uma forma imediata de monitorar a faixa de operação do motor e garantir que estejam aproveitando as rotações

de forma eficiente durante a condução. Esses LEDs são estrategicamente posicionados no volante, oferecendo ao piloto uma representação visual imediata das rotações do motor, facilitando a tomada de decisões durante as corridas e permitindo ajustes precisos de acordo com as condições da pista e a estratégia de corrida(FUELTECH, 2016a).



Figura 7 – FuelTech600

Fonte: (FUELTECH, 2016a)

A injeção eletrônica oferece suporte ao controle de injeção e ignição sequencial em motores de 1 a 12 cilindros. Ela permite o uso de mapas 2D ou 3D para ajustes precisos em diferentes condições de operação. Além das funções básicas, como partida a frio e controle em marcha lenta, o módulo também oferece recursos avançados, como controle de tração e limitador de velocidade. Essas funcionalidades adicionais permitem um controle mais refinado do veículo, proporcionando maior segurança e desempenho.

Além disso, a FT600 está equipada com sensores embutidos, como acelerômetro e giroscópio, que fornecem informações valiosas sobre o comportamento do veículo. Esses sensores adicionais permitem uma análise mais detalhada da dinâmica do veículo, auxiliando na identificação de problemas. As informações coletadas de sensores da podem ser armazenadas na memória interna da central que pode ser configurada para guardar os dados em diferentes taxas por um período contínuo de aproximadamente 3 horas.

A conexão entre as injeções eletrônicas e os sistemas embarcados tem sido um ponto crucial na evolução dos veículos modernos. Os sistemas embarcados desempenham um papel fundamental na integração e no gerenciamento dos componentes eletrônicos complexos presentes nos motores e nos veículos em geral. Através da combinação dessas tecnologias, é possível obter um controle mais preciso e eficiente do desempenho do motor, otimizando a combustão, a eficiência e a confiabilidade. Além disso, os sistemas embarcados permitem o armazenamento e a análise de dados importantes, fornecendo informações valiosas para o desenvolvimento contínuo das injeções eletrônicas.

## 2.5 Sistema Embarcado

Um sistema embarcado pode ser definido como um conjunto de estruturas computacionais projetadas e programadas para desempenhar uma tarefa específica. Ao contrário dos microcomputadores convencionais, que são utilizados de acordo com as necessidades individuais dos usuários, os sistemas embarcados possuem finalidades definidas e são otimizados para atender a requisitos específicos de desempenho, eficiência e confiabilidade. Esses sistemas são projetados para funcionar de forma autônoma, executando tarefas específicas de maneira eficiente e confiável, muitas vezes com recursos limitados de processamento e armazenamento (CHASE, 2007). Os sistemas embarcados estão cada vez mais presentes em diferentes cenários de aplicação, constituem sistemas domésticos e industriais e desempenham tarefas distintas que podem envolver aquisição de dados, controles de sistemas e interação com o ambiente em que estão inseridos.

Os sistemas embarcados geralmente são compostos por três principais componentes:

- *Hardware*: É a parte física do sistema embarcado, que inclui componentes eletrônicos como processadores, memória, periféricos de entrada/saída, interfaces de comunicação digital e outros dispositivos específicos necessários para a funcionalidade do sistema
- *Software* embarcado: Refere-se aos programas e algoritmos desenvolvidos para executar as funções desejadas pelo sistema embarcado. O software é projetado para ser executado no hardware específico do sistema e pode incluir sistemas operacionais em tempo real, drivers de dispositivo, firmware e aplicativos específicos.
- *Firmware*: É o software de baixo nível armazenado em memória não volátil que fornece as instruções básicas para inicializar o sistema embarcado. O firmware geralmente controla o processo de inicialização, configuração do hardware e fornece as interfaces entre o software e o hardware.

### 2.5.1 Microcontrolador

Combinando em um único chip processador, memória e periféricos integrados, microcontroladores são sistemas embarcados compactos versáteis e eficientes que costumam executar as instruções implementadas em *software* de maneira sequencial. São capazes também de se comunicar com outros microcontroladores utilizando variados protocolos de comunicação digital como UART e SPI.

O ESP32 é um microcontrolador com uma arquitetura de 32-bits baseado no processador *Tensilica Xtensa LX6 dual-core* com um *clock* de até 240 MHz e um extenso leque

de periféricos integrados, conforme a Figura 8 que demonstra a arquitetura do ESP32 ilustrando os periféricos e principais componentes a central de processamento e módulos de comunicação sem fio, que é bastante utilizado para atividades de desenvolvimento. Sua principal característica em relação a outros modelos de microcontroladores é a conectividade que ele oferece pois pode suportar protocolos de comunicação tais como Wi-Fi, *Bluetooth Classic* e *Bluetooth Low Energy* e o ESP-NOW (ESPRESSIF, 2023). O dispositivo possui uma memória *Flash* integrada que permite a gravação de dados de sistema ou de configurações, que podem ser salvas durante a operação do microcontrolador e se mantêm armazenadas mesmo após reinício do sistema. Outra ferramenta do ESP32 é a possibilidade de gerenciar armazenamentos de memória SD diretamente sem a necessidade de módulos de controle.

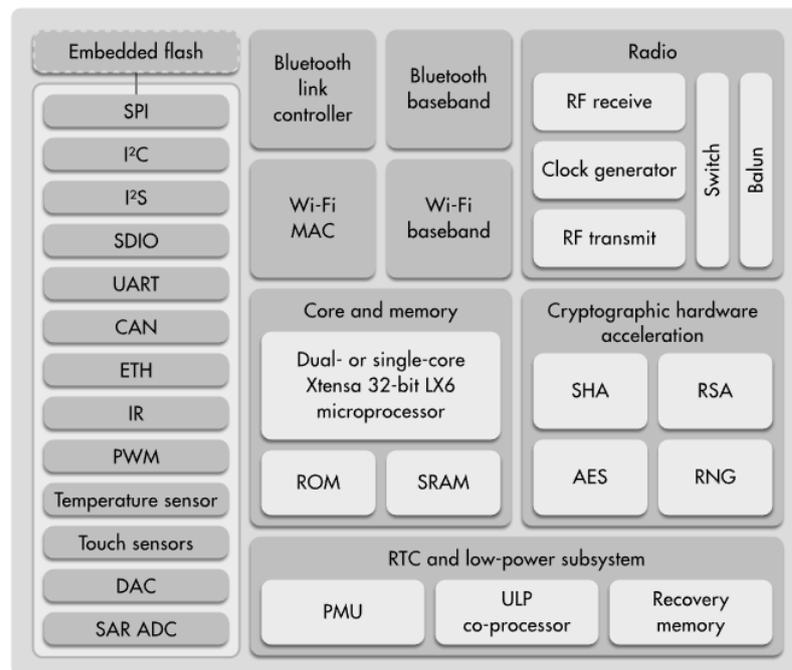


Figura 8 – Arquitetura ESP32

Fonte: (ESPRESSIF, 2023)

Desta forma, o ESP32 é um poderoso microcontrolador, com ampla capacidade de conectividades e interfaces de comunicação. Sua combinação de alto desempenho e baixo consumo de energia o torna uma escolha ideal para uma variedade de aplicações de sistemas embarcados, desde projetos de IoT (Internet of Things) até dispositivos portáteis. No desenvolvimento do projeto com o microcontrolador, a escolha dos protocolos de comunicação é fundamental para a eficácia e a flexibilidade do sistema. Neste contexto, serão explorados os protocolos SPI, UART e I2C, que representam as principais interfaces de comunicação utilizadas. Cada protocolo tem características distintas que o tornam adequado para diferentes tipos de tarefas e periféricos no projeto.

### 2.5.1.1 SPI

A comunicação serial SPI é um protocolo rápido baseado na troca de mensagens entre dispositivos que possuem um papel definido, podendo ser o de mestre ou escravo (TIER, 2019). A comunicação se caracteriza como *Full-Duplex*, pode realizar o envio de dados enquanto recebe informações de outra unidade. O protocolo suporta a conexão entre múltiplos equipamentos onde o dispositivo de destino da mensagem é selecionado através do pino SS (*Slave Select*), mantendo em nível lógico baixo.

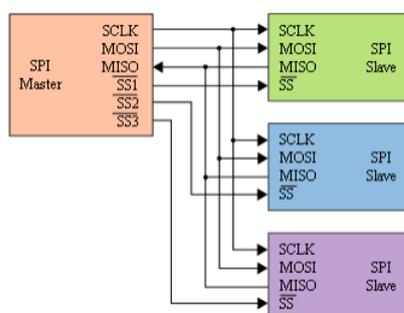


Figura 9 – Conexão entre os dispositivos Master-Slave de uma rede SPI.

Fonte: (TIER, 2019)

### 2.5.1.2 UART

O protocolo UART (Universal Asynchronous Receiver/Transmitter) é um método de comunicação serial amplamente utilizado em microcontroladores. Sua implementação é eficiente, necessitando apenas de uma linha para transmissão (pino TX) e outra para recepção (pino RX), como ilustrado na Figura 10. O protocolo UART possui parâmetros de paridade que garantem a integridade dos sinais transmitidos. Esta definição de paridade assegura o reconhecimento do início de uma transmissão e a duração exata de cada *bit*, essencial para a comunicação eficiente e precisa entre dispositivos.

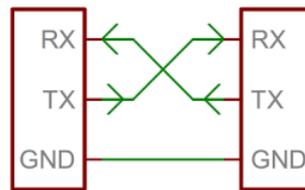


Figura 10 – Conexão entre os dispositivos nas comunicação UART.

Fonte: (TIER, 2019)

### 2.5.1.3 I<sup>2</sup>C

O protocolo de comunicação I<sup>2</sup>C é uma técnica serial que facilita a interação entre dispositivos distintos por meio de um barramento compartilhado, sendo uma criação da Philips Semiconductors, com uso disseminado em uma vasta gama de dispositivos eletrônicos. Sua principal vantagem é permitir que múltiplos dispositivos se conectem ao mesmo barramento para comunicação e troca de dados de forma ágil e simplificada (EMBARCADOS, 2023). No modelo I<sup>2</sup>C, um dispositivo opera como mestre, enviando solicitações de dados para dispositivos escravos, que respondem com os dados requisitados, usando endereços únicos para identificação precisa dos destinatários. Este protocolo é valorizado por sua simplicidade, flexibilidade e eficiência, constituindo um componente crucial em muitos sistemas eletrônicos contemporâneos.

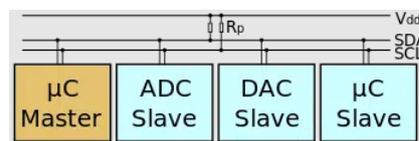


Figura 11 – Conexão entre os dispositivos nas comunicação I<sup>2</sup>C.

Fonte: (EMBARCADOS, 2023)

## 2.6 Rede CAN

Neste capítulo será explorado o funcionamento do protocolo CAN incluindo sua arquitetura, camadas de comunicação, quadros de mensagens e mecanismos de detecção e

correção de erros. Com uma compreensão mais profunda do funcionamento do protocolo CAN, será possível integrar sistemas que possam receber mensagens utilizando rede CAN.

### 2.6.1 Conceito

A eletrônica embarcada trouxe transformações significativas na estrutura de um automóvel na indústria automotiva. De modo geral, os componentes de um veículo podem ser separados em três categorias: componentes mecânicos, componentes eletroeletrônicos e componentes de *software*. Uma segunda classificação pode ser feita, agora em relação a categoria dos eletroeletrônicos, temos:

- Sensores e transdutores automotivos
- Atuadores
- Unidade Eletrônica de Controle

O avanço da utilização da eletrônica embarcada em veículos se deu devido a melhorias de consumo de combustível, menor emissão de poluentes, demandas de segurança veículo, fornecimento de entretenimento entre outras (DUARTE; FIGUEIREDO, 2022). Os primeiros modelos de sistemas eletroeletrônicos para veículos eram centralizados, unificando todas as tarefas em uma única central, porém posteriormente esses sistemas se tornaram distribuídos (Figura 12). O motivo da transição e priorização do sistema distribuído se deu pela complexidade envolvida no grande número de sensores e atuadores que a central eletrônica precisa gerenciar. As principais vantagens são a redução do cabeamento elétrico do veículo, redução da possibilidade de falhas e modularização do sistema. Devido a este formato, é necessário que haja uma rede de comunicação entre as centrais, para que possam transmitir e receber dados necessários para sua operação.

As informações são trocadas entre as Unidades de Controle Eletrônico (ECUs) por meio de protocolos de comunicação. Eles visam criar diretrizes para o projeto e implementação de redes automotivas. Existem vários protocolos, incluindo CAN, *FlexRay*, LIN e *Ethernet*. Esses protocolos atuam como um canal para a transmissão e recepção de dados. Cada protocolo possui recursos técnicos únicos que definem e gerenciam sua utilização.

O protocolo CAN (*Controller Area Network*) foi desenvolvido na década de 80 pela Robert Bosch GmbH, empresa Alemã, pensando inicialmente em instituir a comunicação de sistemas de ônibus e caminhões, porém acabou sendo aplicado em outros campos como em máquinas agrícolas, robótica, máquinas industriais, entre outros (SILVA, 2015).

O padrão de comunicação CAN consiste em um sinal serial assíncrono. A sincronia é designada entre os módulos no início do envio de cada mensagem que é transmitida em um período de tempo regular conhecido. As mensagens são direcionadas por meio de

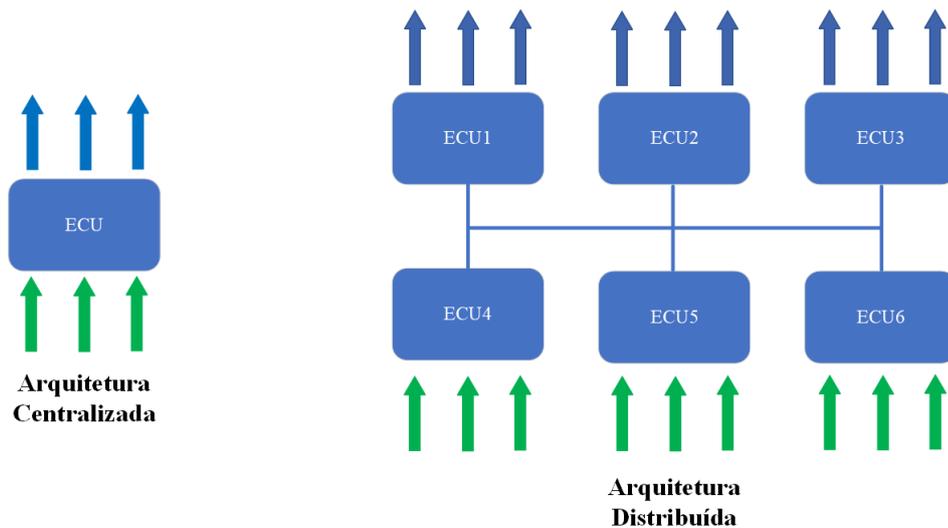


Figura 12 – Arquitetura Centralizada e Distribuída.

Fonte: (SILVA, 2015)

um par trançado de fios, chamados de barramento CAN, em inglês *CAN bus*, sendo um deles denominado de *CAN HIGH* e o segundo de *CAN LOW*. Em cada extremidade do barramento é adicionada uma resistência por padrão de 120 Ohms para evitar reflexão do sinal devido a diferença de impedância dos dispositivos do barramento.

Por ter sido difundido em diferentes aplicações, o protocolo passou por um processo de padronização para que houvesse um modelo comum de comunicação entre os dispositivos de uma rede genérica. A ISO (*International Organization for Standardization*), criou então o modelo OSI (*Open Systems Interconnection*), dividindo o protocolo em sete camadas como demonstra a imagem 13, cada uma com uma função pré-definida, sendo apenas as camadas físicas e de enlace voláteis para customização do projetista de acordo com a aplicação.

## 2.6.2 Funcionamento

No protocolo CAN as mensagens enviadas no barramento recebem o nome de *frames*, que podem ter duas variantes, as de identificação *Standard* com tamanho de 11 *bits*, e o formato *Extended* com identificador constituído de 29 *bits*. Nos dois casos o campo de dados na mensagem possuem o mesmo tamanho de 64 *bits* (8 *bytes*) (SILVA, 2015).

### 2.6.2.1 Camada Física

Sendo o meio no qual a informação é enviada entre os dispositivos no formato de valores binários, 0 ou 1, a camada física é constituída por componentes, como o par de fios trançados (*CAN HIGH* e *CAN LOW*), conectores e resistores de terminação, e parâmetros, como o nível de tensão, sincronização dos módulos eletrônicos, codificação

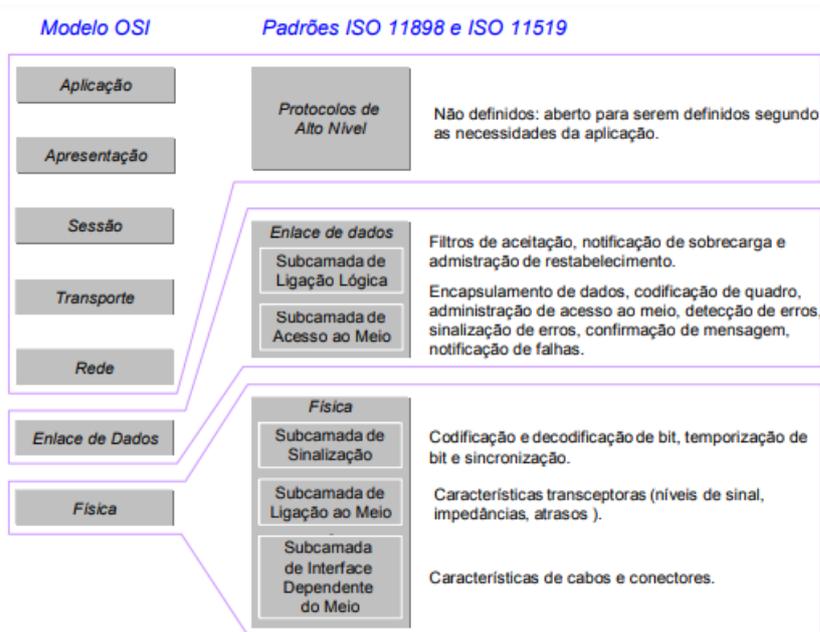


Figura 13 – Relação entre o modelo OSI e os padrões ISO 11898 (CAN) e ISO 11519 (LIN).

Fonte: (SILVA, 2015)

e decodificação dos *bits*. No protocolo CAN os níveis lógicos são constituídos não pela diferença de tensão entre o sinal e o valor de referência, mas sim entre a diferença de tensão entre os dois condutores do barramento. Quando a diferença de tensão entre os fios do barramento CAN atinge 2 Volts, isso indica que um *bit* dominante está sendo transmitido. Por outro lado, quando essa diferença de tensão é de 0 Volts, significa que um *bit* recessivo está sendo transmitido.

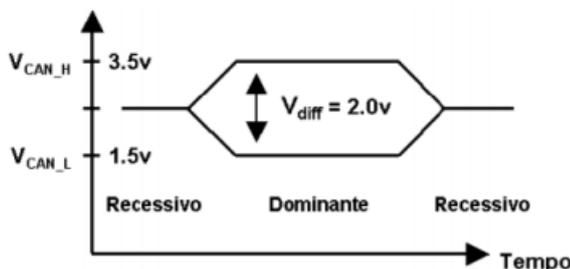


Figura 14 – Diagrama de Nível Lógico do Barramento CAN.

Fonte: (SILVA, 2015)

Com o objetivo de minimizar erros de transmissão, o protocolo CAN utiliza a técnica de *bit stuffing*, que consiste na adição um bit diferente sempre que ocorre uma sequência de cinco *bit* consecutivos de mesmo nível logico na mensagem (o modulo receptor da mensagem retira este *bit* de maneira a manter a mensagem original), assim *bits* de dados transmitidos pelo barramento CAN são agrupados em blocos de 6 *bits* com características

previsíveis. As ECUs utilizam esses blocos para fins de resincronização e verificação de erros de transmissão.

### 2.6.2.2 Camada de Enlace

Na camada de enlace, são realizados processos como o encapsulamento e desencapsulamento de mensagens, detecção e indicação de erros, além da arbitragem das mensagens. Para essa arbitragem, o protocolo CAN utiliza o método baseado no CSMA/CD (*Carrier Sense Multiple Access with Collision Detect*), que permite que apenas uma mensagem seja transmitida no barramento de cada vez. Caso mais de um módulo tente enviar uma mensagem ao mesmo tempo, o *bit* dominante sobrepõe o bit recessivo e o módulo que transmitir o *bit* dominante prevalece, enquanto o outro entra em modo de espera. Esse processo continua até que haja apenas um dispositivo transmitindo no barramento. Essa metodologia é ilustrada na Figura 15, onde é mostrado um exemplo de colisão de mensagens entre dois módulos, sendo que o primeiro módulo prevalece e continua a mensagem enquanto o segundo espera.

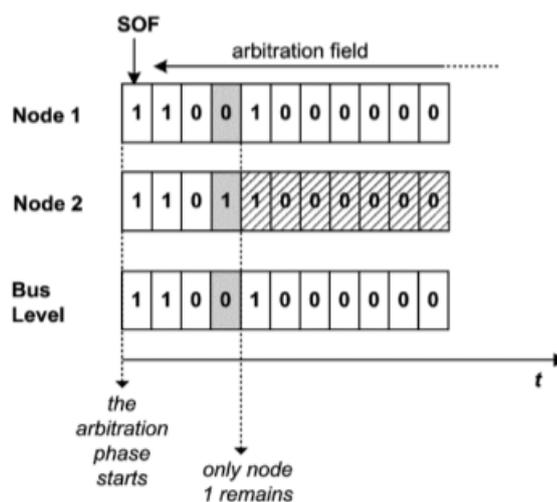


Figura 15 – Método de arbitragem para priorização de acesso no barramento.

Fonte: (SILVA, 2015)

A Figura 16 ilustra a constituição de um *frames* padrão CAN2.0A, nomeando cada um do sete campos que tem suas funcionalidades definidas na Figura 17.

### 2.6.2.3 Detecção de erros

O protocolo CAN possui cinco métodos para detectar erros de transmissão ou elaboração de mensagem, sendo dois deles a nível de *bit* (monitoramento de *bit* e *bit stuffing*) e três a nível de mensagem (*Cyclic Redundancy Check*, checagem de *frames* e *Acknowledge*). Se houver identificação de algum erro, as ECUs podem executar diagnósticos específicos

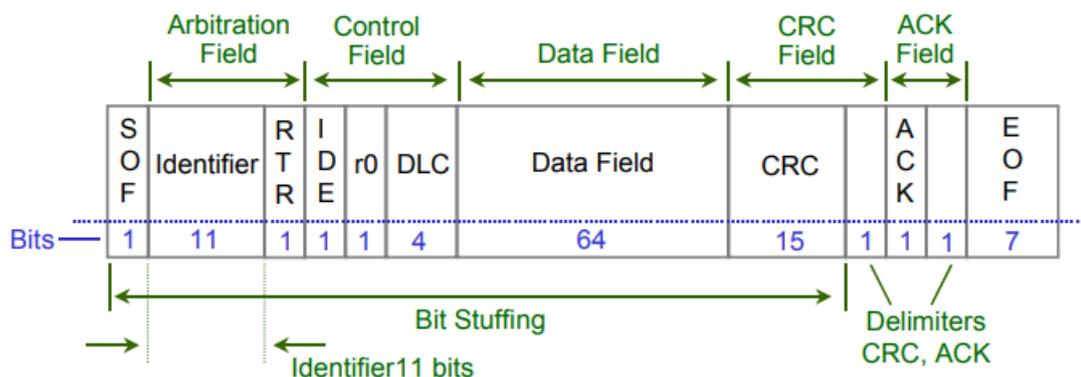


Figura 16 – Campos de um frames de dados CAN 2.0A.

Fonte: (DUARTE; FIGUEIREDO, 2022)

Campo	Subcampo	Funcionalidade
SOF	-	Indicar o início de um <i>frame</i> através de um <i>bit</i> dominante
Arbitration Field Campo de Arbitragem	Identifier	Indicar o <i>ID</i> da mensagem, definindo a prioridade e contribuindo para o endereçamento de mensagens
	RTR Remote Transmission Request	Indicar se o <i>frame</i> é de dados ou de requisição (0 - Dados; 1 - Requisição)
Control Field Campo de Controle	IDE Identifier Extended Bit	Indicar se o <i>frame</i> é do formato padrão ou estendido (0 - Padrão; 1 - Estendido)
	r0	<i>Bit</i> reservado para aplicações futuras não definidas
	DLC Data Length Code	Indicar o número de <i>Bytes</i> presentes no campo de dados
Data Field Campo de Dados	-	Trazer em sua estrutura as informações encapsuladas que se deseja transmitir. Organizando os <i>Bytes</i> do mais significativo para o menos significativo
CRC Field Campo CRC	CRC Cyclic Redundancy Check	Detectar erros utilizando a verificação de redundância cíclica
	<i>Bit</i> recessivo	Indicar o fim do CRC
ACK Field Campo ACK	ACK Acknowledge	<i>Bit</i> recessivo utilizado para a confirmação de recebimento do <i>frame</i> por outro módulo
	<i>Bit</i> recessivo	Indicar o fim do ACK
EOF End of Frame	-	Fim do <i>frame</i> - conjunto de 7 <i>Bits</i> recessivos que informam o fim do <i>frame</i>

Figura 17 – Descrição de cada campo de um frames CAN 2.0A.

Fonte: (DUARTE; FIGUEIREDO, 2022).

para cada tipo de erro detectado (DUARTE; FIGUEIREDO, 2022). Essas medidas permitem uma maior confiabilidade na transmissão de dados pelo barramento CAN. Além do *bit stuffing*, que já foi explicado, os procedimentos funcionam da seguinte maneira:

- Monitoramento de *bit*: Ao enviar um *bit* dominante no barramento CAN, a ECU de envio realiza a leitura do barramento para confirmar a escrita. Em caso de divergência é enviado um *frames* de erro.

- *Cyclic Redundancy Check* (CRC): Baseado em um polinômio predefinido a ECU calcula um valor enviado junto ao *frames*, que é definido em função dos *bits* enviados na mensagem. A ECU que recebe a mensagem realiza o mesmo procedimento e verifica se o valor está de acordo com o recebido;
- Checagem de *frames*: As ECUs que recebem uma mensagem CAN verificam *bits* que possuem valores fixos seguindo o padrão utilizado, a exemplo o *bit* de início de mensagem que sempre possui o valor dominante no protocolo.
- ACK: Após receber um *frames* CAN a ECU receptora preenche o campo ACK com valor dominante e retorna uma mensagem de resposta a ECU emissora. Caso esse campo esteja com o valor recessivo, a mensagem pode ter sido corrompida ou nenhum dispositivo recebeu a informação.

A utilização do protocolo CAN proporciona uma forma eficiente e confiável de comunicação entre as diversas centrais de um veículo. Para ampliar ainda mais as possibilidades de troca de dados dentro de um veículo, a integração do barramento CAN com sistemas secundários para o funcionamento do veículo, como um volante, o protocolo ESP-NOW oferece uma solução complementar que pode ser realizada pelo ESP32, que tem suporte ao protocolo, transmitindo mensagens de forma rápida e sem fio.

## 2.7 ESP-NOW

ESP-NOW é um protocolo de comunicação sem fio desenvolvido pela Espressif Systems que permite a transmissão de dados de baixa potência, baixa latência e alta taxa de transferência entre microcontroladores ESP8266 e ESP32 (ESPRESSIF, 2020). O protocolo foi projetado para comunicação simples ponto a ponto, sem a necessidade de uma rede WiFi ou conexão com a internet. O ESP-NOW opera na banda de 2,4 GHz em uma taxa de transmissão de 1 Mbps e usa um protocolo de comunicação proprietário, tornando-o compatível apenas com outros dispositivos habilitados para ESP-NOW (ESPRESSIF, 2016).

O funcionamento do ESP-NOW envolve dois papéis: o transmissor e o receptor. Para estabelecer a comunicação, o transmissor deve primeiro se registrar com o endereço MAC, que um identificador único para dispositivos de comunicação sem fio gerado pelo fabricante do hardware, do receptor. Uma vez registrado, o transmissor pode enviar dados para o receptor usando o protocolo ESP-NOW. Os dados são divididos em pacotes, cada um dos quais inclui um cabeçalho e uma carga útil. O cabeçalho contém informações sobre o pacote, como seu comprimento, identificação da versão do protocolo e a carga útil contém de fato a mensagem a ser enviada, podendo ter um comprimento máximo de 250 *bytes* (ESPRESSIF, 2016).

Element ID	Length	Organization Identifier	Type	Version	Body
1 byte	1 byte	3 bytes	1 byte	1 byte	0 250 bytes

Tabela 1 – *frames* de envio do ESP-NOW

Fonte: (ESPRESSIF, 2016)

- *ID Element*: Possui o valor fixo de 221 e identifica o fornecedor;
- *Length*: Tamanho total do Organization Identifier, Type, Version e Body;
- *Organization Identifier*: Valor dos três primeiros *bytes* do endereço MAC;
- *Type*: Possui o valor 4, indicando ESP-NOW;
- *Version*: Define a versão do protocolo;
- *Body*: Possui a mensagem da transmissão

O protocolo ESP-NOW opera por meio de uma topologia de comunicação em estrela, onde cada dispositivo transmissor se comunica diretamente com o dispositivo receptor. Essa abordagem traz uma série de vantagens, incluindo uma comunicação de baixa latência, pois os dados são transmitidos de forma direta, sem a necessidade de encaminhamento por uma rede intermediária. A eliminação desse processo de roteamento aumenta a eficiência e reduz a latência, tornando o ESP-NOW ideal para aplicações que exigem uma resposta rápida e em tempo real.

No entanto, é importante notar que, devido à sua topologia em estrela, a comunicação por meio do ESP-NOW é restrita aos dispositivos previamente registrados na rede. Isso garante a segurança e a integridade dos dados transmitidos, pois apenas dispositivos autorizados podem participar da comunicação. Essa limitação é particularmente benéfica em cenários em que a confidencialidade dos dados é essencial, como em sistemas de segurança ou na troca de informações sensíveis.

Para fornecer um nível adicional de segurança, o ESP-NOW suporta recursos de criptografia e autenticação de dados. A criptografia é utilizada para codificar os dados transmitidos, tornando-os ilegíveis para terceiros não autorizados. Além disso, a autenticação garante que apenas os dispositivos confiáveis possam participar da comunicação, prevenindo ataques de dispositivos não autorizados.

Em termos de alcance e latência, o ESP-NOW no ESP32 possui vantagem apresentando uma latência próxima de 1 ms para o envio de 1 *byte* enquanto o protocolo WiFi leva 3,3 ms e o Bluetooth com 6 ms (ERIDANIE; ROCHIM; CESARA, 2021). Como exibido no gráfico da Figura 18, para diferentes tamanhos de pacotes enviados para os protocolos, o ESP-NOW é o que apresenta menor tempo de latência. ESP-NOW também possui resultados superiores tratando-se de alcance, cerca do dobro de alcance em relação

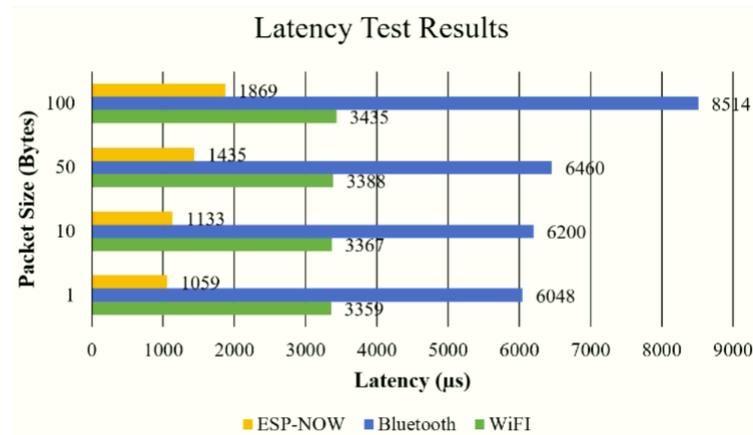


Figura 18 – Comparativo de latência dos principais protocolos de comunicação do ESP32.

Fonte: (ERIDANIE; ROCHIM; CESARA, 2021)

ao WiFi, apresentando um alcance de 185 metros utilizando a antena embutida. Pode-se observar também em metros a distância que o ESP32 possui ao utilizar Bluetooth para estabelecer uma comunicação sem fio na Figura 19.

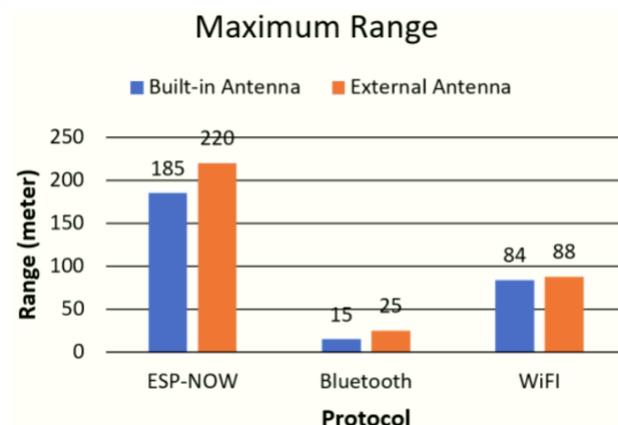


Figura 19 – Comparativo de alcance dos principais protocolos de comunicação do ESP32.

Fonte: (ERIDANIE; ROCHIM; CESARA, 2021)

Essas características de baixa latência, segurança e confiabilidade tornam o protocolo ESP-NOW uma escolha poderosa para a transmissão de dados provenientes do barramento CAN. A integração da ESP32 com o ESP-NOW possibilita a transferência rápida e segura dessas informações para outros dispositivos, ampliando as possibilidades de conectividade e aplicação em sistemas embarcados.

## 2.8 Long Range

Para a implementação da comunicação entre os sistemas de leitura dos sinais CAN do veículo e a equipe nos boxes, foi selecionado o módulo EBYTE e49, um módulo LoRa (Long Range) de transmissão de longo alcance e baixo consumo energético. O módulo

possui um alcance de até 2,5 quilômetros (EBYTE, 2019), e trabalha em uma frequência de 410 MHz até 510 MHz. O dispositivo apresenta 4 modos distintos de operação que podem ser selecionados de acordo com a configuração dos pinos M1 e M0. O modo de *Transmission Mode* pode ser utilizado para realizar a transmissão de mensagens seriais recebidas pelo pino RX. As mensagens recebidas pelo módulo são transmitidas pelo pino TX. Esse modo utiliza então uma interface UART para a comunicação com o microcontrolador. Os parâmetros da comunicação serial como a taxa de transmissão e demais configurações podem ser ajustados por meio do modo de configuração *Setting Mode*. Os modos de operação do dispositivo estão descritos na Tabela 2.

Modo	M1	M0	Descrição
0	0	0	UART e canal de transmissão aberto, o módulo transmite de acordo com as configurações.
1	0	1	UART e canal de transmissão desativado, o módulo transmite a intensidade do canal atual a cada 100 ms.
2	1	0	Canal de transmissão desativado e a interface UART opera para receber os parâmetros de funcionamento.
3	1	1	UART e canal de transmissão desativado, o módulo entra em modo <i>Stand By</i> .

Tabela 2 – Modos de operação LoRa EBYTE E49.

Fonte: (ESPRESSIF, 2016)

Os conhecimentos adquiridos neste capítulo, que incluem métodos de comunicação e o funcionamento das injeções eletrônicas, juntamente com os periféricos de comunicação do microcontrolador ESP32 e os meios de transmissão de dados entre as centrais, constituem uma base sólida para fundamentar as etapas do próximo capítulo. Por fim, o protocolo ESP-NOW e as tecnologias LoRa são explorados, ressaltando suas aplicações em sistemas de telemetria veicular. Este conhecimento fornece a base necessária para avançar para a metodologia do projeto, abrangendo os procedimentos realizados e as ferramentas usadas para alcançar os objetivos do trabalho.

## 3 Metodologia

Neste capítulo, serão abordados os métodos e recursos empregados para o desenvolvimento deste trabalho, divididos nas seguintes etapas: arquitetura eletroeletrônica, modelagem mecânica e testes e validações. São apresentadas as montagens, o desenvolvimento dos sistemas e os componentes utilizados

### 3.1 Arquitetura Eletroeletrônica

Para o desenvolvimento da arquitetura eletrônica, que visa extrair informações da injeção eletrônica e possibilitar a comunicação entre duas centrais desenvolvidas para serem instaladas no veículo- o módulo de comunicação CAN, chamado de Sniffer, e a central de exibição de dados, denominada Volante - foram empregados componentes específicos. Esses incluem o microcontrolador ESP32 (conforme a imagem 20), módulo de controle CAN, display, sensor GPS, sensor acelerômetro, baterias para alimentação do circuito no volante, e um módulo de controle de carga. A configuração dessas conexões é ilustrada na Figura 21, detalhando a estrutura e o funcionamento integrado do sistema. Para a construção das centrais foram utilizados os seguintes componentes: microcontrolador ESP32, módulos de controle CAN, display, sensor GPS, acelerômetro e baterias para alimentação do circuito instalado no volante e um módulo de controle de carga. A conexão das centrais está representada na Figura 21.



Figura 20 – Microcontrolador ESP32.

Fonte: (EMBARCADOS, 2018).

A instalação da central Sniffer, constituída do ESP32, um módulo MCP2515, módulo GPS VK16E e o transmissor LoRa e49, no chicote elétrico do carro (Figura 22), requer a adição de um circuito regulador de tensão utilizando o CI (Circuito Integrado) LM2596S-5.0, uma vez que a principal fonte energética do veículo possui a tensão de 12V. O CI é um regulador do tipo *Step-Down* com capacidade de corrente de até 3A INSTRU-

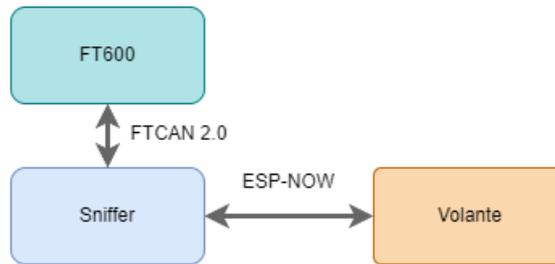


Figura 21 – Conexão das centrais com a Injeção

Fonte: Autor.

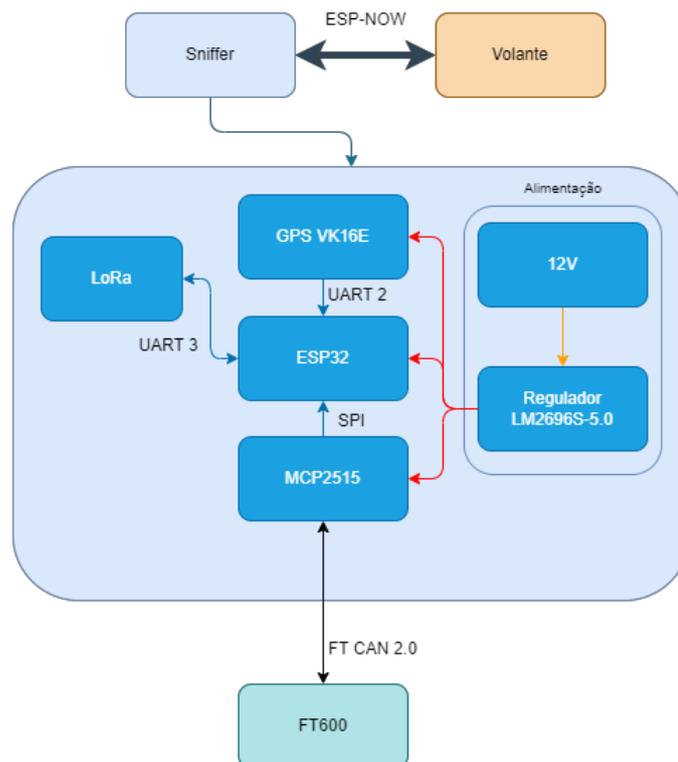


Figura 22 – Arquitetura da central Sniffer.

Fonte: Autor.

MENTS(2013) e reduz a tensão de maneira ativa, sendo mais eficiente do que reguladores lineares comuns.

Com o objetivo de incrementar a funcionalidade do sistema, a equipe solicitou a inclusão de um módulo GPS para fornecer dados de geolocalização do veículo durante as provas. O módulo VK16E foi selecionado por sua alta sensibilidade e capacidade de início rápido, garantindo precisão e confiabilidade na aquisição de coordenadas em tempo real (CHIPSKEY, 2013). Sua integração ao sistema ocorre através da interface UART, facilitando a comunicação com o microcontrolador principal e permitindo que a equipe tenha acesso contínuo às informações de posicionamento.

A arquitetura da segunda central eletrônica desenvolvida, denominada central volante, foi projetada para ser embutida no próprio volante. Esta integração impõe desafios



Figura 23 – Módulo de comunicação CAN MCP2515.

Fonte: (DUARTE; FIGUEIREDO, 2022).

relacionados ao espaço físico limitado e à necessidade de soluções para o fornecimento de cabos que não sejam comprometidos pelo movimento de direção. A estrutura e disposição dos componentes desta central, com suas particularidades e soluções técnicas, são detalhadas na Figura 24. O display utilizado é o TFT Nextion NX4827P043 de 4,3 polegadas e é diretamente conectado ao pinos de comunicação serial UART do ESP32 (NEXTION, 2022). Este modelo possui uma tela sensível ao toque e suporte ao uso de figuras sem fundo na montagem das interfaces, tendo uma maior liberdade para customização das telas que serão exibidas.

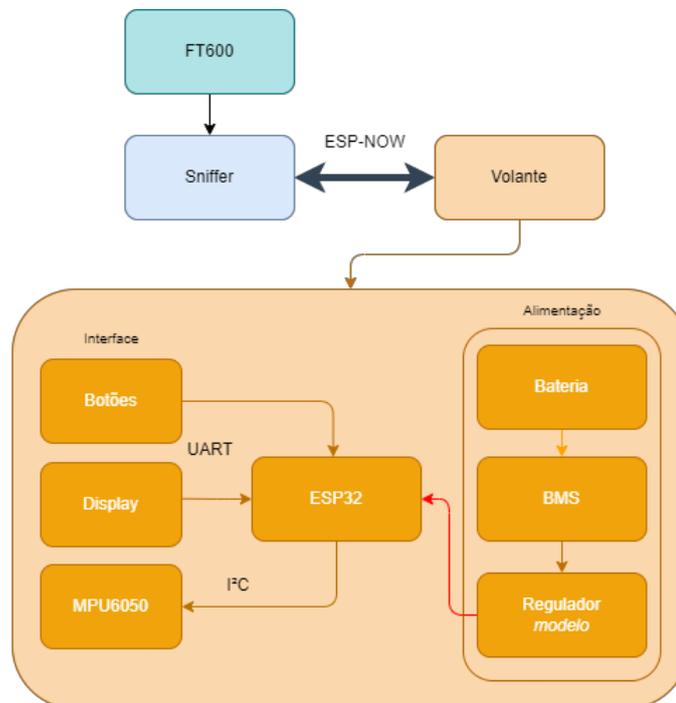


Figura 24 – Arquitetura da central Volante.

Fonte: Autor.

Os botões utilizados para o envio de comandos são conectados aos pinos de entradas do microcontrolador e utilizam resistores de 10k Ohms para realizar o *PullDown* do sinal. Cada uma das três chaves seletoras possuem quatro posições distintas, em que um dos pinos de cada posição está sempre conectado ao pino comum. Essas chaves são empre-



Figura 25 – Display Nextion NX4827P043.

Fonte: (NEXTION, 2022).

gadas para selecionar um parâmetro específico antes de enviar um comando ou mensagem para outra central. A fim de minimizar o número de pinos de entrada do microcontrolador empregado na leitura das chaves, optou-se por utilizar um circuito com divisor de tensão (Figura 26), conectado exclusivamente a uma porta de leitura analógica. Os valores dos resistores foram determinados a partir da equação do divisor resistivo de tensão 3.1, de maneira que respeitassem a Tabela 3.

$$V_{out} = V_{in} \times \frac{R2}{R1 + R2} \quad (3.1)$$

Nessa equação,  $V_{out}$  representa a tensão de saída,  $V_{in}$  é a tensão de entrada,  $R1$  é o valor do resistor  $R1$  e  $R2$  é o valor do resistor  $R2$ .

Posição	Vin(V)	R1(Ohms)	R2(Ohms)	Vout(V)
1	3	-	-	3
2	3	4,7k	10k	2,2
3	3	10k	4,7k	1,2
4	0	-	-	0

Tabela 3 – Posição da chave seletora com sua saída em tensão.

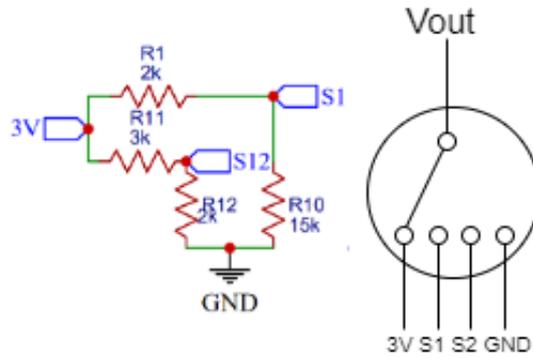


Figura 26 – Circuito divisor de tensão e diagrama da chave seletora.

Fonte: Autor.

Para aferir a posição de giro do volante, o acelerômetro MPU6050 foi integrado à central do volante. O acelerômetro do MPU6050 mede as acelerações nos três eixos espaciais (X, Y e Z) (INC., 2013). Quando a MPU6050 está em repouso e devidamente nivelada, a componente da aceleração gravitacional é diretamente medida pelo acelerômetro na direção do eixo Z, que é perpendicular ao plano de referência. Assumindo que o acelerômetro está idealmente calibrado, a magnitude da aceleração gravitacional ( $A_g$ ) é igual a  $1g$  ( $9,81 \text{ m/s}^2$ ), considerando a aceleração da gravidade padrão na superfície da Terra.

Utilizando trigonometria, podemos calcular o ângulo de inclinação  $\theta_{acel}$  do plano em relação à direção da gravidade a partir das leituras do acelerômetro nos eixos X, Y e Z:

$$\theta_{acel} = \arctan \left( \frac{A_{eixo}}{\sqrt{A_X^2 + A_Y^2 + A_Z^2}} \right) \quad (3.2)$$

Onde:

- $\theta_{acel}$  é o ângulo de inclinação estimado pelo acelerômetro no eixo em questão.
- $A_X, A_Y, A_Z$  são as leituras do acelerômetro nos eixos X, Y e Z, respectivamente.
- $A_{eixo}$  é o valor do acelerômetro no eixo de interesse (X, Y ou Z).

A alimentação do sistema é proveniente de um banco de baterias configurado com duas pilhas conectadas em série, resultando em uma tensão total de  $8,4 \text{ V}$  quando carregadas. Para adequar essa tensão aos requisitos de operação, é utilizado um regulador de tensão AMS1117-5V, que regula a tensão para um valor constante de  $5 \text{ V}$  (CORP, 2005). Esse regulador assegura a estabilidade da tensão de alimentação para os componentes eletrônicos. Para garantir o gerenciamento adequado das baterias, incluindo carga, sobrecarga, descarga e proteção contra curto-circuito, é empregado um módulo BMS (*Battery*

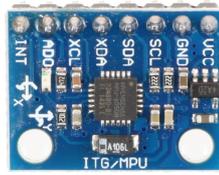


Figura 27 – Acelerômetro e giroscópio MPU6050.

Fonte: [RYNDACKCOMPONENTES](#).

*Management System*) HX-2S-D01. Esse módulo é capaz de monitorar o estado das baterias e oferece funcionalidades de controle e proteção. Ele é projetado para fornecer uma corrente máxima de até 8A [MEGASATURNV\(2017\)](#), atendendo às demandas de energia do sistema de forma segura e eficiente. BMS HX-2S-D01 desempenha um papel fundamental na manutenção da saúde das baterias, garantindo a otimização do seu desempenho e prolongando sua vida útil. Além disso, o módulo é responsável por prevenir situações de sobrecarga, descarga excessiva e curto-circuito, evitando danos às baterias e mantendo a integridade do sistema como um todo.

$$I = \frac{P}{V} \quad (3.3)$$

Nessa equação,  $I$  representa o valor de corrente,  $V$  é a tensão de alimentação do dispositivo,  $P$  é o valor da potência consumida.

Com essa estrutura de bateria é possível estimar a autonomia energética do sistema e validar seu uso durante a competição, que conta por exemplo com uma prova de enduro onde o carro precisa percorrer a maior distância em 20 minutos. O consumo teórico de corrente dos componentes exibidos na Tabela 4. Para a aferição do consumo do ESP32 levou-se em consideração o momento de consumo máximo de potência do controlador, sendo este o momento de atuação do controlador para envio de mensagens ESP-NOW [ERIDANIE; ROCHIM; CESARA\(2021\)](#), e foi feita a conversão do valor para amperes utilizando a equação 3.3. O valor de consumo para o display é referente à sua configuração de brilho máximo fornecido na ficha técnica do componente ([NEXTION, 2022](#)). Dessa forma, a equação 3.4 é capaz de fornecer uma estimativa de tempo de funcionamento da central, levando em consideração a capacidade das baterias, que é de 2200 mAh ([BAK, 2011](#)). Com base nessa capacidade, é possível estimar um tempo de duração aproximado de cerca de 2 horas antes que as baterias precisem ser recarregadas ou substituídas.

$$T = \frac{\text{Capacidade da Bateria}}{\text{Corrente de Consumo}} \quad (3.4)$$

Onde  $T$  é o tempo de duração estimado em horas, a *Capacidade da Bateria* é medida em mAh e a *Corrente de Consumo* em mA.

Componente	Consumo(mA)
Nextion TFT	370
ESP32	520
Total	890

Tabela 4 – Tabela de consumo dos componentes do volante.

O esquemático da Figura 28 desenvolvido utilizando o *software EasyEDA* ilustra as conexões eletrônicas dos módulos com o ESP32 bem como os circuitos auxiliares para que os reguladores tensão funcionem corretamente e o circuito de condicionamento de sinal dos botões.

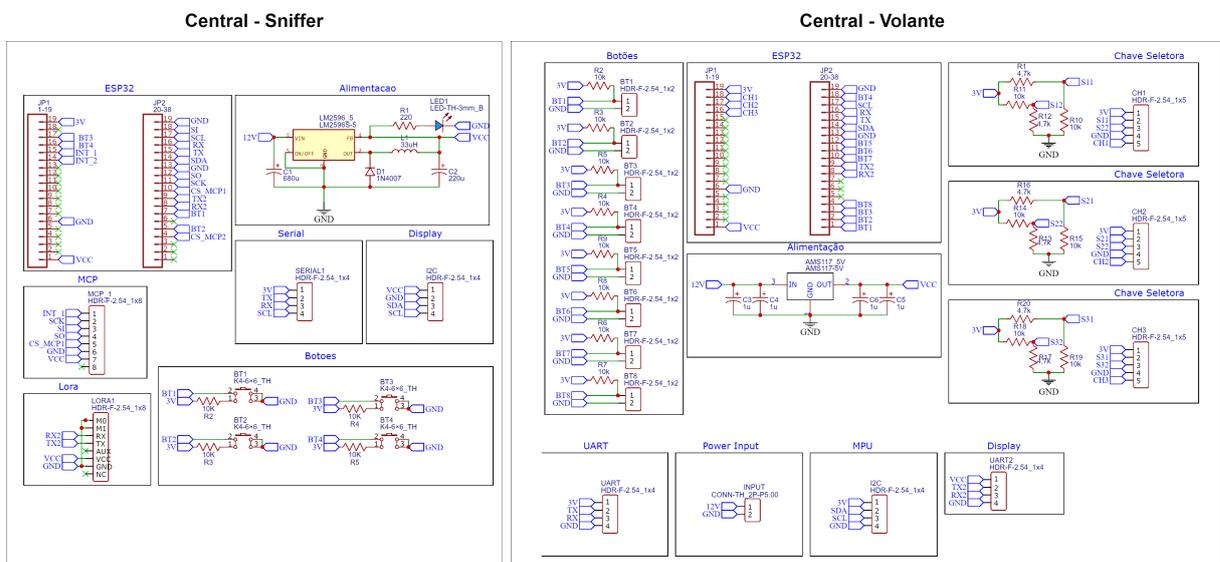


Figura 28 – Esquemático dos circuitos de cada central

Fonte: Autor.

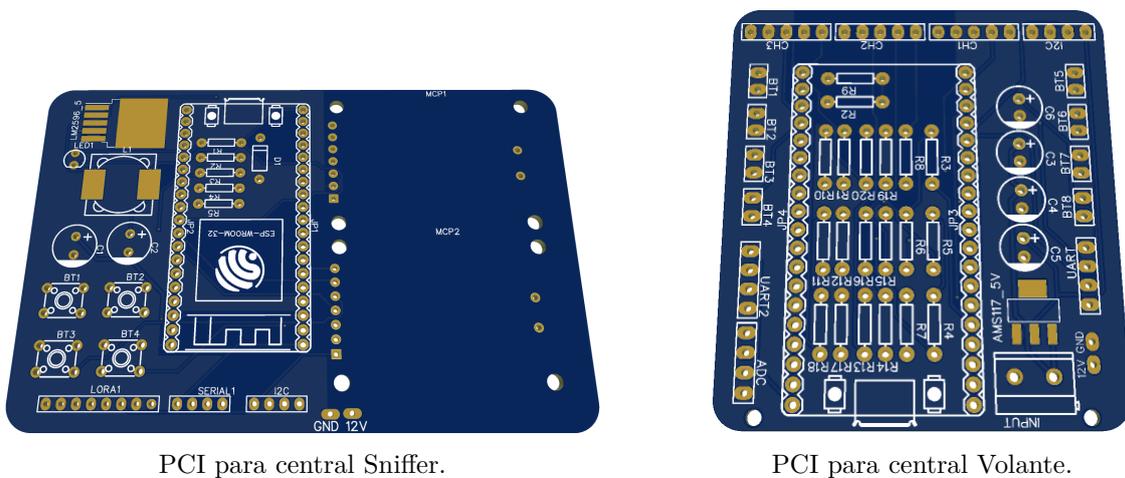


Figura 29 – Placas de circuito impresso desenvolvidas utilizando o *software EasyEDA*.

Os dois circuitos foram montados em protoboard para a realização dos testes e validações de funcionamento utilizando uma fonte de bancada para fornecimento de energia.

Após os testes foi possível produzir as placas de circuito impresso (PCI), que estão ilustradas na Figura 29, seguindo as etapas de projeto: esquemático, layout da PCI, geração dos arquivos Gerber e envio desses arquivos para fabricação. Desta maneira, com o hardware estabelecido foi possível prosseguir as atividades de desenvolvimento do software das duas centrais para a implementação dos requisitos funcionais do protótipo.

## 3.2 Aquisição de dados FT CAN2.0

Para a obtenção das informações registradas pela Unidade de Controle Eletrônico responsável pelo controle do motor, foi adotado o módulo eletrônico MCP2515, que oferece uma interface de comunicação digital SPI. O módulo foi configurado para uma taxa de transmissão de 1000 kbps, que foi definida a partir da taxa de transmissão do fabricante da FT600, e ao ser conectado no barramento CAN B FUELTECH(2016b) foi possível extrair *frames* com o uso de exemplos de leitura da biblioteca do controlador como consta na Figura 30 (FOWLER, 2013). Nota-se que todos os *frames* foram do formato de identificador estendido de 29 *bits* em conformidade com o que o fabricante apresenta em documentação(FUELTECH, 2022).

```
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xA7
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xA7
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAB
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAB
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAB
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAB
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAB
Extended ID: 0x120003FF DLC: 4 Data: 0x00 0x24 0x0D 0xAF
```

Figura 30 – *frames* extraídos do barramento CAN-D FT600 utilizando o MCP2515

Fonte: Autor.

Para a interpretação dos *frames* recebidos, foi empregado o manual do protocolo da ECU, mais especificamente o FT CAN 2.0, como referência (FUELTECH, 2022). O formato padrão de comunicação CAN adotado é o CAN 2.0B. O manual aborda a lógica de identificação de diferentes dispositivos FuelTech no mesmo barramento, o significado de cada mensagem recebida e seu nível de prioridade (Figura 31). O campo de identificação pode ser dividido em 3 partes:

- *ProductID*: Identifica o dispositivo que está enviando a mensagem assim como qual a prioridade que ele possui no barramento;
- *DataFieldID*: identifica o formato da mensagem;
- *MessageID*: identifica a natureza da mensagem, podendo ser um comando, leitura de sensor, configuração de sistemas.

IDENTIFICATION (29 bits)		
Bits 28 to 14 (15 bits)	Bits 13 to 11 (3 bits)	Bits 10 to 0 (11 bits)
<i>ProductID</i>	<i>DataFieldID</i>	<i>MessageID</i>

Figura 31 – Estrutura do campo de identificação FTCAN 2.0

Fonte: (FUELTECH, 2022).

Tratando-se do campo de dados da mensagem CAN (Figura 32), o protocolo pode dividir os *frames* recebidos em dois grupos:

CAN FRAME	
29 bits	0 to 8 bytes
IDENTIFICATION	DATA FIELD

Figura 32 – Estrutura de *frames* do protocolo FTCAN 2.0.

Fonte: (FUELTECH, 2022).

- *Simple Packet*: Toda a informação é enviada em um único *frames* - primeiro byte do *frames* possui o valor 0xFF (Figura 33);
- *Segmented Packet*: A informação é dividida em múltiplos *frames*, o primeiro *byte* de cada *frames* possui o valor da ordem de cada segmento, a exemplo o primeiro inicia com 0x00 e o segundo com 0x01 (Figura 34). O tamanho máximo de um pacote segmentado em *bytes* é de 1776 *bytes*.

DATA FIELD (1 to 8 bytes)							
0	1	2	3	4	5	6	7
0xFF	PAYLOAD						

Figura 33 – Campo de dados do tipo *Standart*.

Fonte: (FUELTECH, 2022).

O fabricante realiza a transmissão de mensagens no tipo broadcast de informações fixas que constam na Figura 35 (FUELTECH, 2022). Essas informações são dados de sensores chaves enviados em uma taxa de 100 Hz no barramento no formato *Simple Packet* e carrega 4 sinais em cada *frames*. Desta forma, ao identificar o *frames* por meio do MessageID, basta realizar a atribuição do valor do sensor de acordo com a posição de cada *byte* como feito na Figura 36.

No caso dos *frames* segmentados o fabricante traz a estruturação de como realizar a identificação de cada sinal que é composto de uma palavra de 4 *bytes* sendo os dois primeiros o identificador do sinal e o demais o valor. A Figura 37 demonstra como estratificar um conjunto de mensagens segmentadas compostas por 5 sinais e serviu como base para a implementação do algoritmo de leitura dos *frames* pelo microcontrolador.

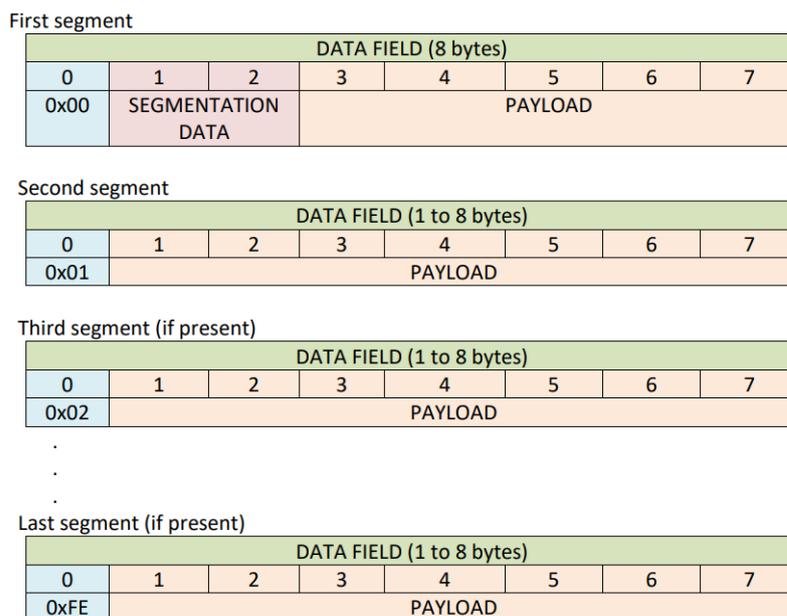


Figura 34 – Campo de dados do tipo *Segmented*.

Fonte: (FUELTECH, 2022).

		Bytes							
ID (FT500)	ID (FT600/550/450)	1	2	3	4	5	6	7	8
0x14000600	0x14080600	TPS		MAP		Air Temperature		Engine Temperature	
0x14000601	0x14080601	Oil Pressure		Fuel Pressure		Water Pressure		Gear	
0x14000602	0x14080602	Exhaust O2		RPM		Oil Temperature		Pit Limit	
0x14000603	0x14080603	Wheel Speed FR		Wheel Speed FL		Wheel Speed RR		Wheel Speed RL	
0x14000604	0x14080604	Traction Ctrl - Slip		Traction Ctrl - Retard		Traction Ctrl - Cut		Heading	
0x14000605	0x14080605	Shock Sensor FR		Shock Sensor FL		Shock Sensor RR		Shock Sensor RL	
0x14000606	0x14080606	G-force (accel)		G-force (lateral)		Yaw-rate (frontal)		Yaw-rate (lateral)	
0x14000607	0x14080607	Lambda Correction		Fuel Flow Total					

Figura 35 – Mensagens fixas de transmissão.

Fonte: (FUELTECH, 2022).

MessageID	Measure 1		Measure 2		Measure 3		Measure 4	
	Measure ID	Description	Measure ID	Description	Measure ID	Description	Measure ID	Description
0x600	0x0002	TPS	0x0004	MAP	0x0006	Air temperature	0x0008	Engine temperature

Figura 36 – Exemplo de estratificação de um *frames* de broadcast FTCAN 2.0.

Fonte: (FUELTECH, 2022).

Os *MeasureID* de cada sinal transmitido pelos dispositivos FuelTech estão relacionados em uma tabela no manual do protocolo (Figura 38) com a descrição, a unidade do sinal, um multiplicador e a taxa de transmissão(FUELTECH, 2022).

### 3.3 Escrita FT CAN2.0

Para estabelecer a conexão de maneira adequada entre a central Sniffer e a FT600, é crucial que a injeção esteja configurada para receber sinais de outros dispositivos, essa configuração pode ser realizada por meio do software *FT Manager*, na seção "Comunicação

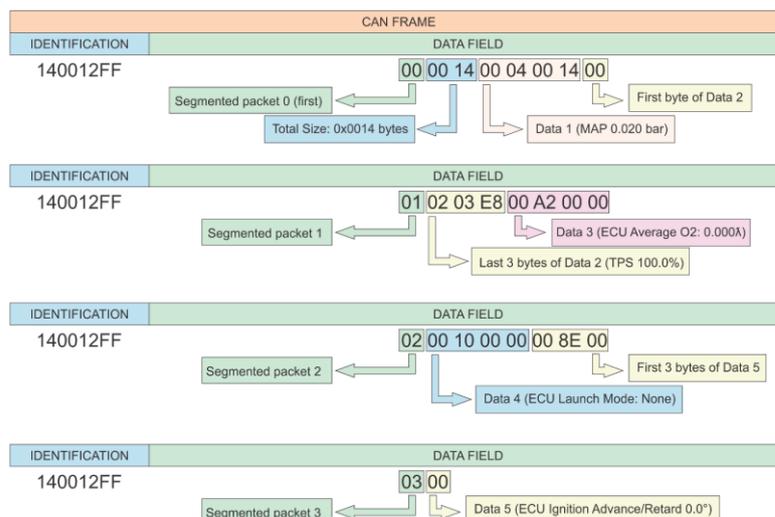


Figura 37 – Exemplo de estratificação de frames segmentados.

Fonte: (FUELTECH, 2022).

MeasureID	DataID	Description	Unity	Multiplier	Broadcast source (rate*)
0x0000	0x0000	Unknown	-	-	-
0x0002	0x0001	TPS	%	0.1	PowerFT ECU 100Hz
0x0004	0x0002	MAP	Bar	0.001	PowerFT ECU 100Hz
0x0006	0x0003	Air temperature	°C	0.1	PowerFT ECU 10Hz
0x0008	0x0004	Engine temperature	°C	0.1	PowerFT ECU 10Hz
0x000A	0x0005	Oil pressure	Bar	0.001	PowerFT ECU 100Hz
0x000C	0x0006	Fuel pressure	Bar	0.001	PowerFT ECU 100Hz
0x000E	0x0007	Water pressure	Bar	0.001	PowerFT ECU 100Hz
0x0010	0x0008	ECU Launch Mode (2-Step, 3-Step, Burnout, Burnout + Spool)	-	Note 1	PowerFT ECU 100Hz
0x0012	0x0009	ECU Battery voltage	Volts	0.01	PowerFT ECU 100Hz
0x0014	0x000A	Traction speed	Km/h	1	PowerFT ECU 100Hz Gear Controller 100Hz
0x0016	0x000B	Drag speed	Km/h	1	PowerFT ECU 100Hz Gear Controller 100Hz

Figura 38 – Tabela de MeasureID.

Fonte: (FUELTECH, 2022).

CAN" (Figura 39). Para assegurar a comunicação eficaz com a central de injeção eletrônica FT600, foi implementada uma simulação do dispositivo *Switch Panel* (Figura 40), um componente utilizado para enviar comandos e parâmetros pelo barramento CAN por meios de botões do próprio fabricante da injeção. Para o envio de cada mensagem, é necessário realizar a configuração do ID específico do dispositivo e dos bytes do frames, de acordo com os botões acionados.

Além disso, para manter uma comunicação ativa e contínua, é essencial que a central transmita periodicamente o estado dos botões, com um intervalo de 50ms à 250ms. Uma *task* pode ser configurada no microcontrolador para o envio contínuo dos frames, garantindo a continuidade da comunicação entre a central Sniffer e a FT600. Para efetuar a escrita de frames com identificador estendido utilizando o MCP2515, é necessário realizar uma operação lógica OR (OU) entre o valor do identificador e o valor 0x800000. Essa operação é um requisito específico do controlador MCP2515.

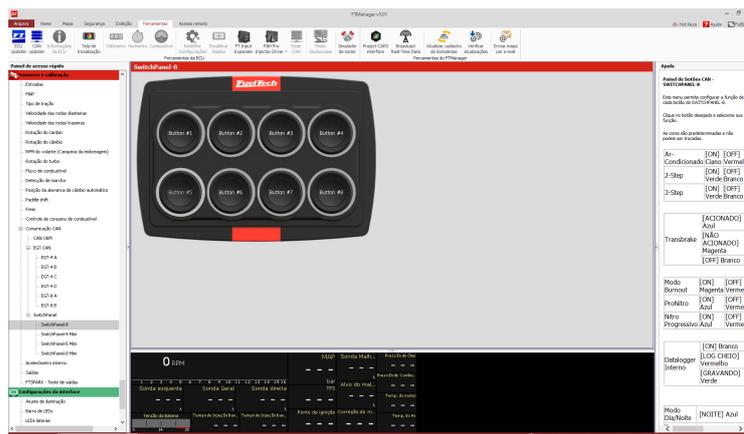


Figura 39 – Interface de configuração do *Switch Panel* no *FT Manager*.

Fonte: Autor



Figura 40 – Dispositivo *SwitchPanel 8 Mini*.

Fonte: (FUELTECH, 2023)

### 3.4 Transmissão ESP-Now

Após a classificação de cada sinal, a central acoplada à injeção eletrônica desempenha o papel de enviar os dados para o controlador no volante. Dentre as diversas opções de transmissão de dados disponíveis, foi escolhida uma modalidade que exigisse a menor modificação possível no chicote elétrico já existente no veículo. Entre as opções embutidas no microcontrolador, o protocolo ESP-NOW foi selecionado devido à sua baixa latência e facilidade de implementação. Para configurar o protocolo ESP-NOW, é necessário definir o endereço MAC do dispositivo de destino da mensagem, especificar o modo de funcionamento e os dados que serão enviados. Os valores selecionados para transmissão foram escolhidos de acordo com as necessidades da equipe e são declarados em uma estrutura de objetos que será enviada pelo protocolo (3.1), a mesma estrutura deve ser declarada no receptor de maneira que cada sinal seja do mesmo formato declarado na estrutura de

envio. frames

Listing 3.1 – Estrutura de envio ESP-NOW

```
//===== ESPNOW SEND =====
uint8_t broadcastAddress [] = {0x58, 0xBF, 0x25, 0x33, 0x53, 0x48};
// Endereco de broadcast - Display

/*Estrutura de envio com declaracao de cada variavel*/
typedef struct struct_message {
    int espRPM; #Rotacao por minuto
    int espTPS; #Pedal de aceleracao
    int espMAP; #Sensor MAP
    int espAirTemp; # Temperatura do Ar
    int espEngineTemp; #Temperatura do motor
    int espBatteryVoltage; #Tensao da bateria
    int espOilPressure; #Pressao de oleo
    int espFuelPressure; #Pressao de combustivel
    int espExhaustO2; #Sonda Lambda
    int espGear; #Marcha
    int espECUSpeed; #Velocidade do ve culo
    int espBrakePressure; #Pressao na linha de freio
} struct_message;
struct_message FTData;
esp_now_peer_info_t peerInfo;
```

No ESP32 da central Volante, as mesmas configurações foram realizadas para que fosse possível receber as mensagens utilizando o protocolo e também enviar mensagens relacionadas aos comandos provenientes dos botões. Isso permite uma comunicação bi-direcional entre o Sniffer e o controlador no volante, facilitando o controle e a interação entre os sistemas.

## 3.5 Display TFT Nextion

Um display TFT Nextion de 4,3” com resolução de 480×272 pixels foi selecionado como componente de exibição dos dados no volante. O dispositivo possui uma interface serial UART para a comunicação com o microcontrolador, desta forma com o auxílio da biblioteca de controle do display (ITEAD, 2015) é possível realizar a partir do ESP32 o envio dos valores de cada variável a ser exibida na tela. A estrutura gráfica da interface foi feita utilizando o software do fabricante *Nextion Editor* 41, onde se posicionou os elementos gráficos que constituíram a tela e assim gerou-se um arquivo de configuração do display. Este arquivo carrega todas as ilustrações que serão exibidas bem como suas posições, bastando ao microcontrolador realizar o controle de exibição.

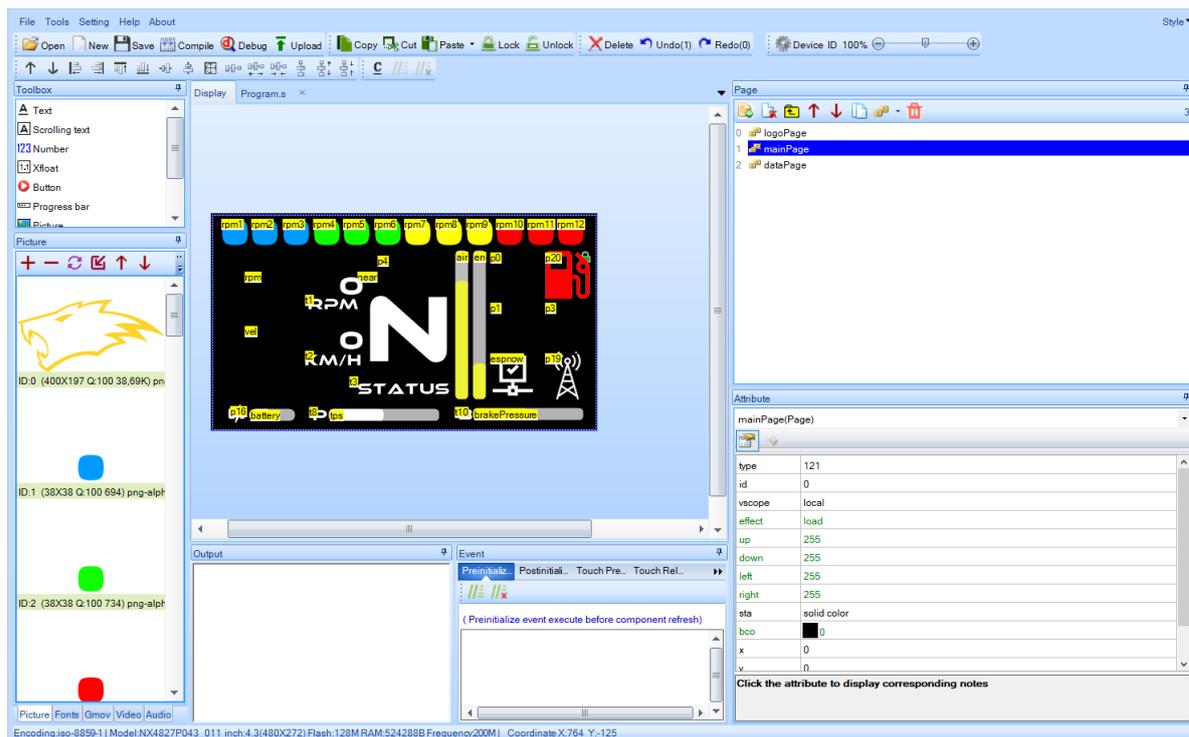


Figura 41 – Interface de desenvolvimento *Nextion Editor*.

Fonte: Autor.

A tela contém a interface do piloto, dispondo de gráficos e valores referentes a dados chaves de desempenho como velocidade linear do veículo, giros por minuto do virabrequim e posição da marcha atual. Demais dados secundários também são ilustrados como posição do pedal de aceleração e temperatura do motor, são dados que podem auxiliar a equipe a identificar falhas ou prevenir danos à unidade de potência. A interface conta com ícones que indicam estados importantes como baixo nível de combustível, conexão com a central de leitura CAN e comunicação com boxes como demonstra a Figura 42.

### 3.6 Firmware

A implementação de um projeto de *firmware* otimizado e que possua e menor latência no gerenciamento das diversas funções das centrais pode ser feita utilizando a biblioteca FreeRTOS, um sistema operacional em tempo real (RTOS) para o microcontrolador ESP32 (ESPRESSIF, 2018). Esta biblioteca facilita a execução concorrente de múltiplas tarefas, conhecidas como *tasks*. Cada *task* é cuidadosamente projetada para funcionar de forma autônoma, otimizando o compartilhamento e a gestão dos recursos do sistema de maneira eficaz.

O emprego do RTOS possibilita uma gestão de fluxo de operações mais estruturada e organizada. Com isso, a execução de uma operação não bloqueia ou atrasa indevidamente as demais, permitindo que o sistema responda prontamente a eventos em tempo

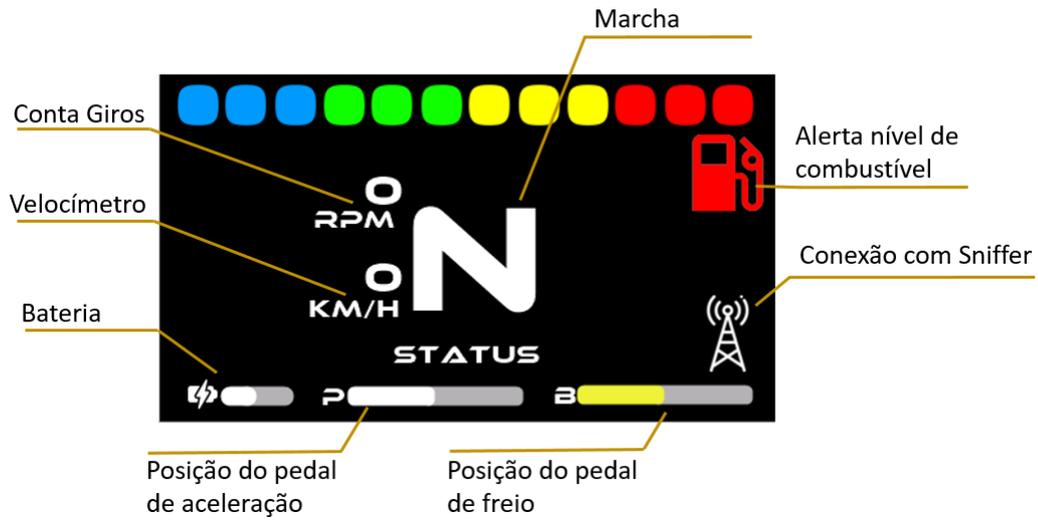


Figura 42 – Interface do piloto.

Fonte: Autor.

real. A divisão do programa em *tasks* distintas, cada uma com sua própria prioridade e função específica, melhora significativamente a confiabilidade e a eficiência do sistema, tornando-o mais escalável e fácil de manter. Desta maneira foram criadas *tasks* para cada funcionalidade da central Sniffer, como demonstra o diagrama da Figura 43.

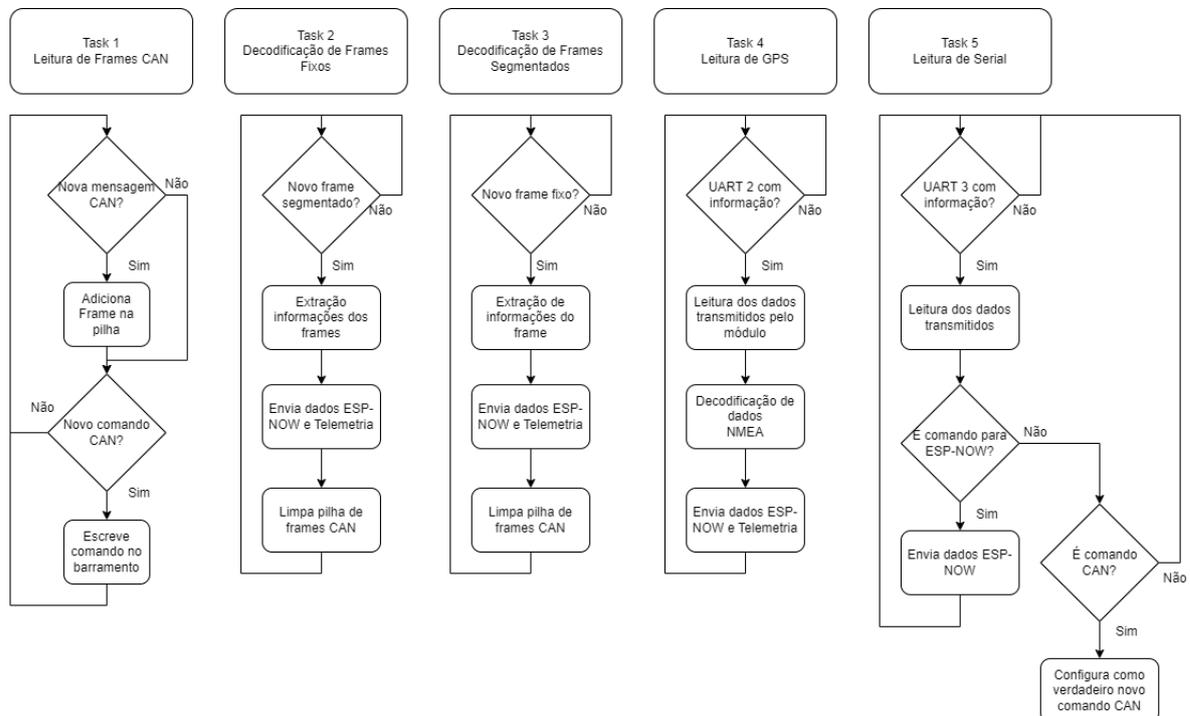


Figura 43 – Fluxograma do firmware da central Sniffer.

Fonte: Autor

A “Task 1 - Leitura *frames* CAN” está configurada para leitura contínua de *frames* CAN, servindo como o ponto inicial para a coleta de dados do barramento. Quando uma nova mensagem CAN é identificada, o pino INT do MCP2515 muda para nível

lógico baixo, ela é prontamente adicionada à pilha para processamento subsequente. Outra responsabilidade atribuída a esta *task* é verificar periodicamente, a cada 250 ms, se o semáforo está liberado, permitindo assim o envio do estado dos botões. Esse procedimento mantém a conexão ativa que simula o *Switch Panel*, interagindo com a injeção eletrônica. Além disso, a *task* monitora a recepção de novos comandos provenientes da central do Volante, que devem ser escritos no barramento CAN.

A “Task 2 - Decodificação de *frames* Fixos” e “Task 3 - Decodificação de *frames* Segmentados” são especializadas na decodificação de *frames* CAN. A primeira lida com *frames* fixos, decodificando seguindo a estrutura fornecida pelo fabricante para esse tipo de *frames*, Figura 35. A Task 3 decodifica os *frames* segmentados, passando por duplas de *frames* armazenadas na pilha para a extração dos sinais de acordo com MeasureID detectado. Ambas as *tasks*, realizam a transmissão via ESP-NOW e telemetria, proporcionando uma integração contínua dos dados CAN com sistemas de monitoramento remoto. A “Task 4 - Leitura de GPS” gerencia a leitura de dados de GPS através da UART 2, decodificando os protocolos NMEA para extração de informações de localização, enquanto a “Task 5 - Leitura de Serial”, aguarda dados e de telemetria ou para serem transmitidos via ESP-NOW a central do Volante, ou serem enviados no barramento CAN para a FT600.

A construção do *firmware* da central Volante também pode ser baseada na utilização das funções de RTOS. Nesta central três *tasks* foram configuradas como demonstra o diagrama 44. A “Task 1 - Leitura dos botões” é responsável pela leitura de botões, utilizando um *loop* contínuo para monitorar as interações do usuário. Quando um botão é pressionado, uma função de *callback* é invocada para identificar o botão acionado e determinar o comando correspondente a ser executado. Essa abordagem assegura que cada ação do usuário seja capturada de forma precisa e que as respostas do sistema ocorram imediatamente e em tempo real. A “Task 2 - Controle Display” gerencia o controle do display, aguardando a chegada de novas mensagens ESP-NOW ou comandos de troca de tela. Quando uma mensagem é recebida, a *tasks* é responsável por enviar essas informações através da interface UART para o dispositivo. A “Task 3 - Comunicação ESP-NOW” lida com a comunicação ESP-NOW, verificando a presença de novos comandos que devem ser enviados para a central Sniffer. Se algo novo é detectado, a *task* procede para o envio valor que deve ser escrito no barramento CAN, de acordo com o comando acionado. Juntas, as *tasks* trabalham de forma sincronizada para garantir a interatividade do usuário com o sistema e a comunicação efetiva entre os componentes do firmware.

Através da análise detalhada dos requisitos e especificações do projeto, foi possível realizar testes iniciais para o desenvolvimento do volante e a integração com a ECU. No próximo tópico, serão apresentados os resultados dessas implementações, incluindo a extração de *frames* e escrita na central de injeção FT600.

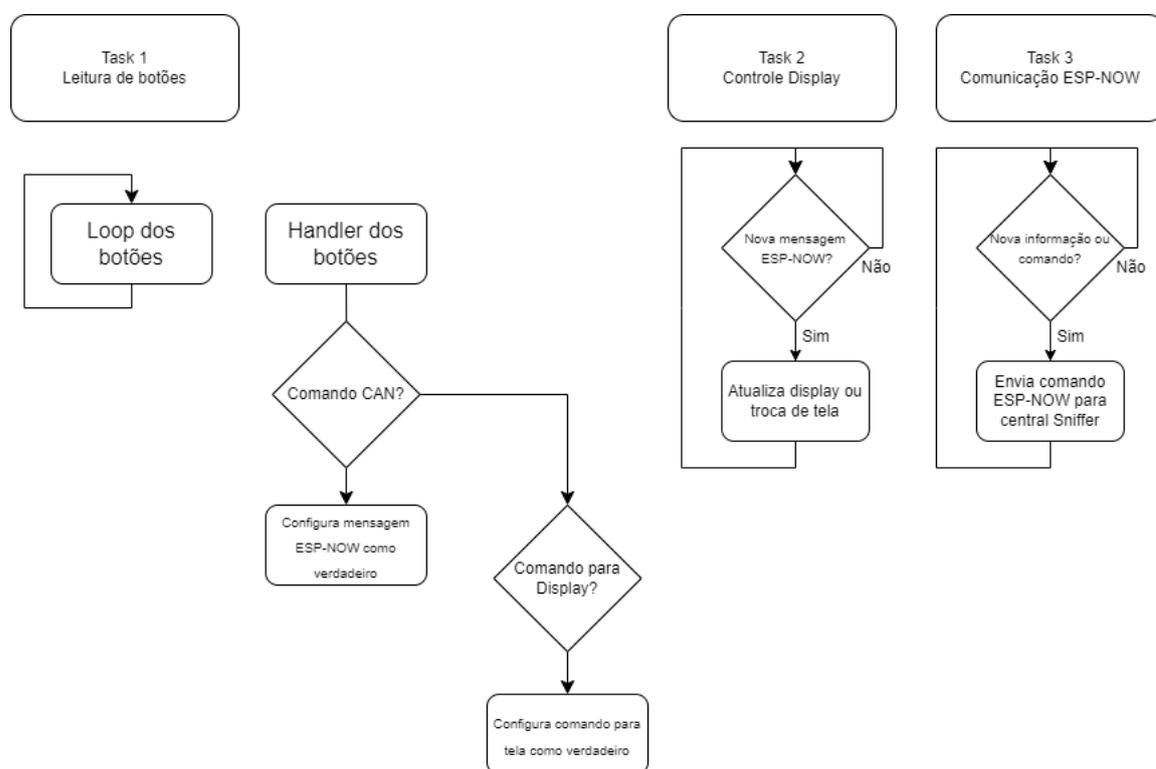


Figura 44 – Fluxograma do firmware da central Volante.

Fonte: Autor

## 4 Resultados

Este capítulo dedica-se à apresentação dos resultados alcançados na implementação do sistema proposto. O foco principal foi o desenvolvimento de um fluxograma eficaz para a extração e transmissão de dados utilizando a tecnologia ESP-NOW, com especial atenção à otimização da comunicação e ao processamento dos dados provenientes da FT600. Os resultados indicam a eficiência do algoritmo desenvolvido na detecção e separação de *frames* individuais, bem como na escrita de mensagens no barramento da FT600, simulando um *Switch Panel*. Além disso, a integração do sistema com os boxes por meio da telemetria permitiu a comunicação em tempo real e o monitoramento à distância do desempenho e dos dados do veículo, o que representa um avanço significativo na capacidade de gestão e análise durante as provas.

### 4.1 Central Sniffer

Nesta seção, apresentaremos os resultados referentes à central Sniffer, detalhando os componentes e a funcionalidade do sistema. Discutiremos a implementação e configuração dos periféricos assim como a integração do hardware que resultou na funcionalidade otimizada do sistema.

Um dos principais motivos da utilização do ESP32 como controlador principal de cada central é justamente por sua variedade de interfaces e periféricos eletrônicos. Conforme proposto no capítulo anterior, fez-se uso dos periféricos UART e SPI para estabelecer a comunicação essencial para o funcionamento da central Sniffer. A implementação do hardware foi organizada para priorizar a eficiência e a confiabilidade do sistema, conforme relacionado na Tabela 5. A interface serial UART 1 foi dedicada à depuração do sistema, fornecendo um meio direto para o monitoramento de atividades. A UART 2 do microcontrolador foi definida para conexão com o módulo LoRa E32 para o sistema de telemetria, enquanto a serial UART 3 se integrou ao módulo de GPS VK16E, responsável por fornecer dados de geolocalização para o sistema. Além disso, a comunicação SPI com o módulo MCP2515 permitiu uma interação efetiva com o barramento CAN, lendo os *frames* e escrevendo comandos CAN no barramento, destacando a capacidade do ESP32 de atuar como um ponto central de coleta e processamento de dados, fundamental para o sniffer e suas funcionalidades avançadas.

Desta forma, foram fabricadas as placas de circuito impresso do diagrama 29, e os componentes instalados como demonstra a Figura 45, permitindo desenvolver a programação da central no *hardware* final do protótipo.

Periférico	Uso
Serial UART 1	Depuração do sistema
Serial UART 2	LoRa E32 - Telemetria
Serial UART 3	GPS VK16E
SPI	MCP2515 - Módulo CAN

Tabela 5 – Utilização de periféricos do ESP32.

Fonte: Autor.



Figura 45 – Placas fabricadas: Sniffer(direita) e Volante(esquerda).

Fonte: Autor

Para o acoplamento da central no veículo um invólucro foi modelado utilizando a ferramenta Dassault Systems CATIA, levando em consideração os componentes da central. O invólucro foi fabricado utilizando uma impressora 3D, e pode ser observado na Figura 46, junto com os componentes que integram a central. A central conta com apenas um conector de quatro vias para conexão com o chicote elétrico, sendo dois para alimentação e o outros dois são para o barramento CAN.

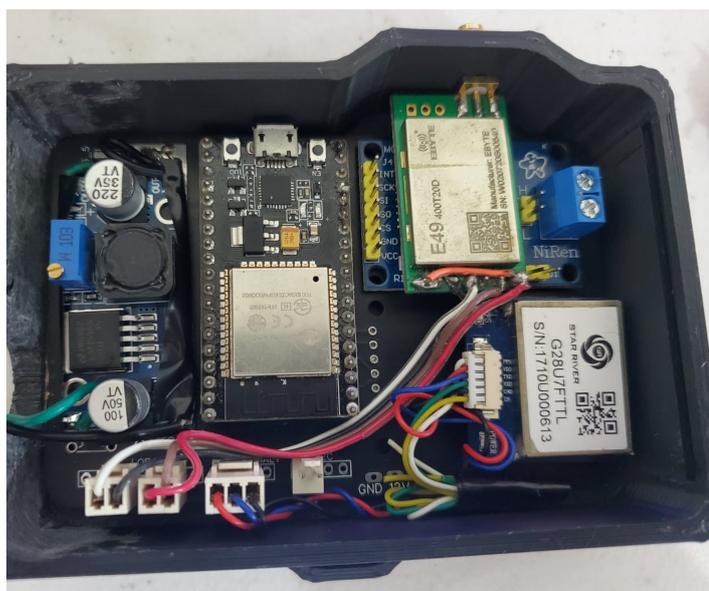


Figura 46 – Sniffer montado junto com os componentes do sistema.

Fonte: Autor

## 4.2 Volante

O projeto da central Volante foi concebido com uma arquitetura que emprega a interface UART para comunicação com o display e I<sup>2</sup>C para leitura dos dados do acelerômetro MPU6050, enquanto aproveita o hardware integrado do microcontrolador para estabelecer a conexão via protocolo ESP-NOW. Os botões são conectados diretamente aos pinos de entrada do microcontrolador, garantindo uma leitura imediata e precisa de suas ativações. Além disso, a leitura analógica das chaves seletoras é realizada de maneira direta, conforme proposto no capítulo anterior. Esta configuração assegura uma interação simplificada e eficaz entre os componentes da central.

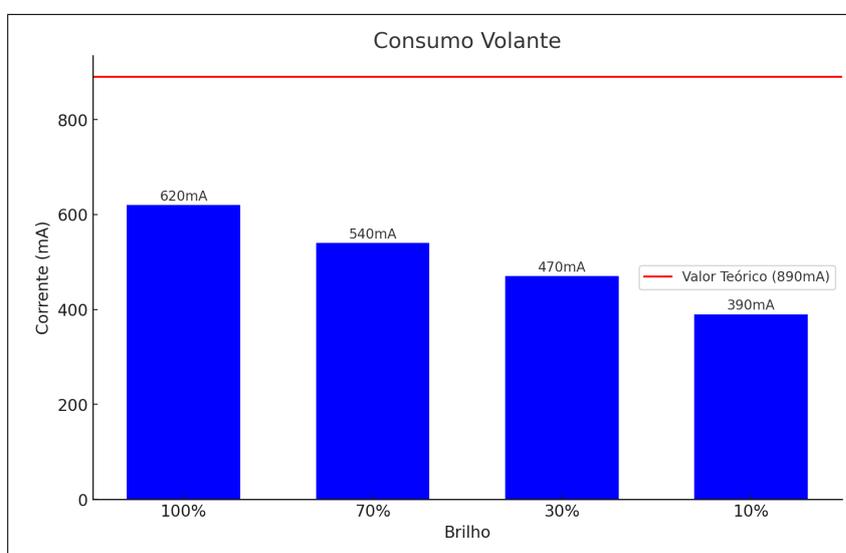


Figura 47 – Consumo central volante.

Fonte: Autor

A alimentação da central foi construída de acordo com o proposto e é eficiente em fornecer energias para o microcontrolador e o display. O consumo real da central durante a troca de mensagens utilizando ESP-NOW foi medido variando a intensidade da tela e está demonstrado na Figura 47. Neste mesmo cenário e mantendo a configuração de intensidade de brilho do display em 70%, foi medido a tensão da bateria instalada na central do volante e os resultados obtidos estão no gráfico 48, que contém os limites de tensão de carga completa e descarga do banco de bateria. O conjunto excedeu 2 horas de fornecimento de energia e pode ser considerado um sistema superdimensionado levando em consideração apenas a duração da principal prova da competição, o Enduro. Um sistema com essa autonomia proporciona mais segurança a equipe para realizar o acompanhamento do sistema eletrônico na preparação para as provas.

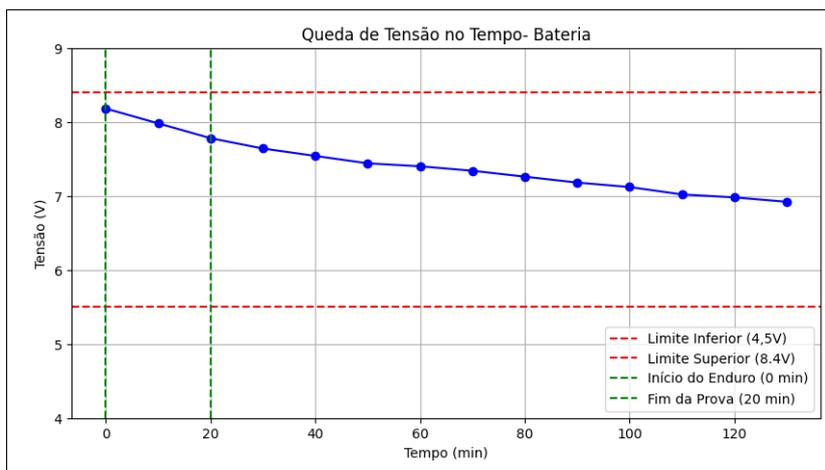


Figura 48 – Autonomia central volante.

Fonte: Autor

A customização dos comandos na central de injeção é feita levando em conta tanto as funcionalidades disponíveis no modelo específico de injeção quanto as necessidades particulares da equipe. Os comandos escolhidos para implementação estão relacionados na Tabela 6. A alocação de cada comando a um botão específico no volante foi decidida em colaboração direta com o piloto, como demonstra a Figura 49, assegurando que a disposição dos controles seja intuitiva e eficiente.



Figura 49 – Mapeamento dos botões no volante.

Fonte: Autor

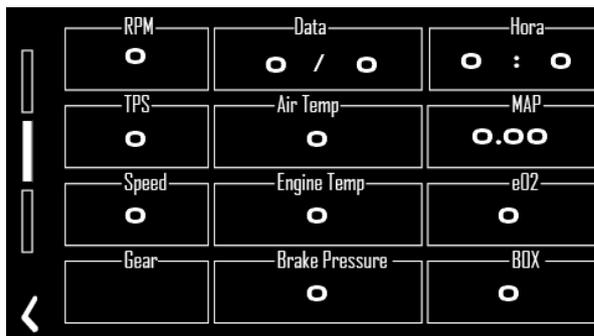
O display Nextion oferece a flexibilidade de alternar entre diversas telas programáveis, possibilitando a personalização do painel de controle conforme as necessidades da equipe. Para este projeto, foram desenvolvidas três telas em colaboração com os membros da equipe, cada uma destacando informações vitais e formatos de exibição específicos. Além da interface principal ilustrada na Figura 42, há uma tela dedicada à representação gráfica de variáveis críticas como a posição do pedal e a rotação do virabrequim, uma

Botão	Funcionalidade
1	Pit Limit
2	2-Step
3	Confirmar Mapa
4	Controle de Tela
5	Zerar ângulo MPU6050
6	Alternar
Chave	
1	Escolha de Mapa
2	Alterar Intensidade do brilho da tela
3	Controle de Tração

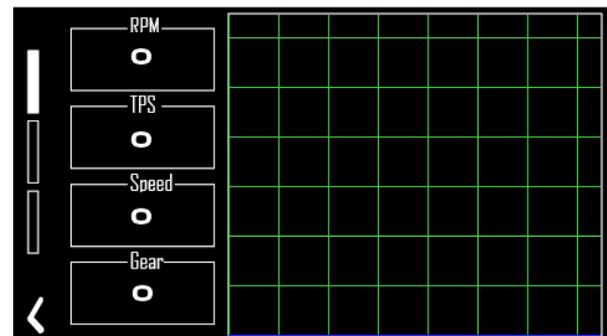
Tabela 6 – Mapeamento das funções dos botões.

Fonte: Autor.

terceira tela apresenta uma visão detalhada dos dados dos sensores em formato de tabela e uma página reservada para o registro de mensagens recebidas via ESP-NOW. A três telas adicionais estão ilustradas na Figura 50 e a transição entre as telas é realizada pelo botão 1 ou por um simples toque na tela sensível ao toque.



Tela de exibição dos valores numéricos recebidos.



Tela de exibição do gráfico do RPM e posição do pedal.



Tela de exibição do log do ESP-NOW e Telemetria.

Figura 50 – Interface do usuário para a exibição de dados e gráficos de telemetria.

Fonte: Autor

Telas adicionais foram concebidas para funcionar como interfaces de alerta e comu-

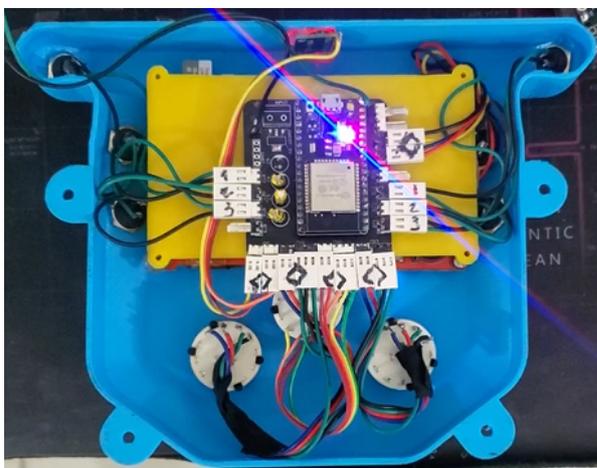
nicação, permitindo que mensagens importantes da equipe sejam transmitidas ao piloto de forma clara e visual. Estas telas são programadas para aparecer por alguns segundos, garantindo que as informações sejam destacadas sem distrair o piloto de maneira clara. As cinco telas de comunicação desenvolvidas estão ilustradas na Figura 51.



Figura 51 – Telas de alertas para o piloto.

Fonte: Autor

Desta forma, um invólucro foi desenvolvido com o intuito de acomodar os componentes de forma segura e ergonômica no volante. Este suporte foi projetado para garantir não só a integridade dos elementos eletrônicos durante as provas, mas também para proporcionar facilidade de acesso e operação pelo piloto. Na Figura 52, é possível observar o posicionamento interno dos componentes eletrônicos e a conexão entre a placa de circuito fabricada e as chaves e botões. Na Figura 53 é possível ver o posicionamento final da central no volante utilizado pela equipe, demonstrando que o projeto comporta os componentes sem a necessidade de fabricação de um novo volante.



Posicionamento dos componentes da central.



Vista frente da central Volante.

Figura 52 – Montagem e instalação dos componentes da central no invólucro.

Fonte: Autor



Figura 53 – Posicionamento final da central no volante.

Fonte: Autor

### 4.3 Integração

Com a finalização da montagem dos protótipos, foi possível realizar a validação do algoritmo de extração de sinais dos *frames* CAN e da comunicação ESP-NOW implementada entre as centrais. No processo de extração dos *frames* CAN, a unidade central Sniffer foi integrada ao sistema de injeção eletrônica por meio do barramento CAN. Após processar os dados, os sinais decodificados foram visíveis na saída serial de log da central, como mostrado na Figura 54. Para validar, utilizou-se a função de simulação de motor do software FT Manager, que permite estabelecer valores específicos para sinais como RPM e TPS. Esses valores simulados foram transmitidos pelo barramento CAN da unidade central, assegurando a acurácia na leitura dos dados simulados.

```

ID: 0x940810FF Data: 0x0 0x0 0xC 0x0 0x84 0x0 0x0 0x0 Measure ID 0x84 : RPM
{"RPM":0.00} Valor : 0
ID: 0x940811FF Data: 0x0 0x0 0x38 0x0 0x4 0x0 0x0 0x0 Measure ID 0x04: MAP
{"MAP":0.00} Valor : 0
ID: 0x940811FF Data: 0x1 0x2 0x0 0x0 0x0 0xA 0x2 0x9E Measure ID 0x0A : Oil Pressure
{"Oil Pressure":0.67} Valor : 670/1000 = 0.67

```

Figura 54 – Decodificação de *frames* FT CAN2.0.

Fonte: Autor

Após extrair com sucesso sinais dos *frames*, estes foram enviados via ESP-NOW e exibidos no display da central Volante como demonstra a Figura 55. Para certificação do funcionamento correto da tela de gráfico, também utilizou-se a seção de simulação e foram enviados valores que foram então exibidos no gráfico da segunda tela, funcionalidade que pode por exemplo auxiliar o piloto e a equipa e realizar diagnósticos. Para o envio de comandos, utilizou-se a seção de Rede CAN do FT Manager, onde é possível observar os dispositivos conectados no barramento da FT600. Com o sistema conectado o ícone indica a presença de um *Switch Panel* de 8 botões, e quando um botão é pressionado um indicador de acionamento fica na coloração verde, indicando que a central recebeu

o comando do dispositivo, como demonstra a Figura 57. Desta forma bastou apenas a o mapeamento de qual o botão do *Switch Panel* ficou relacionado os comandos selecionados pela equipe, como demonstra a Figura 58.



Figura 55 – Dados exibidos na FT600 e na central Volante.

Fonte: Autor

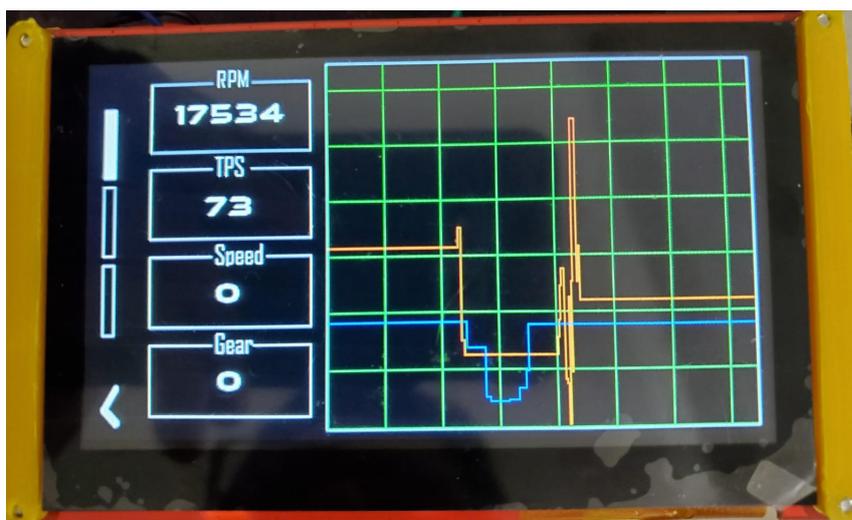


Figura 56 – Amostragem gráfica dos valores de RPM e TPS.

Fonte: Autor

No processo de validação da comunicação entre a unidade Sniffer e a equipe, empregou-se o módulo LoRa acoplado à interface UART de um computador Raspberry Pi 4. Foi desenvolvido um código específico para facilitar a leitura e a exibição dos dados recebidos, garantindo a transmissão eficiente e a integridade dos sinais. A Figura 59 ilustra os resultados obtidos, exibindo a interface de usuário que detalha os dados simulados no volante, bem como o menu de simulação do FT Manager. Para validação do sistema de telemetria entre o veículo e a equipe, utilizou-se o módulo LoRa conectado na interface UART de um computador Raspberry Pi 4, como demonstrado na imagem 60. Um código de leitura direta da interface foi desenvolvido utilizando a linguagem de programação

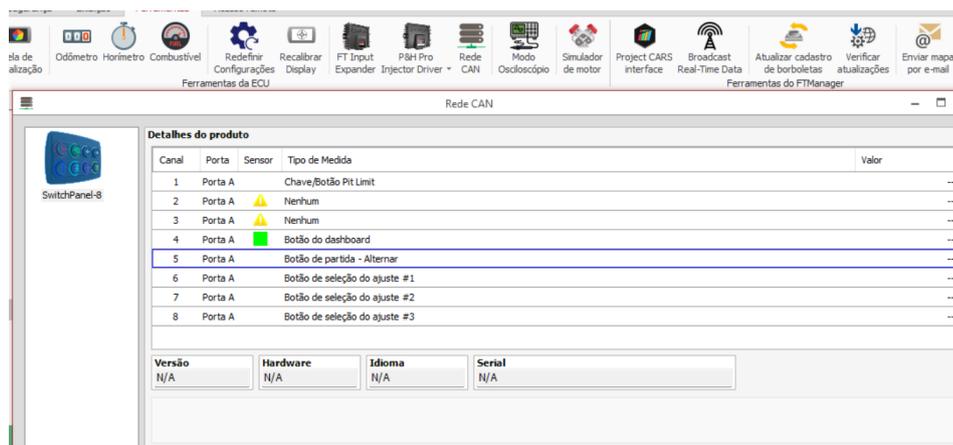


Figura 57 – FT600 reconhece a presença do *Switch Panel*.

Fonte: Autor

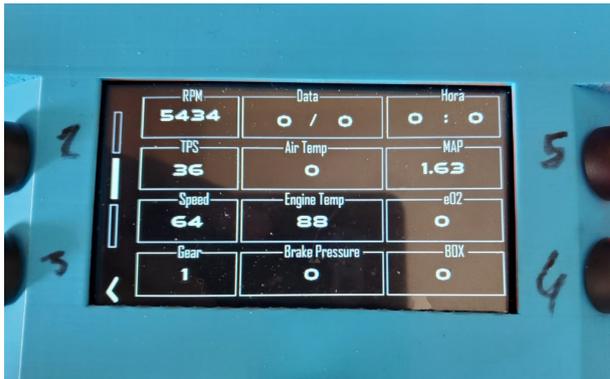


Figura 58 – Mapeamento dos botões do *Switch Panel*.

Fonte: Autor

Python para exibição dos dados recebidos, que na Figura 61 contém os valores de latitude e longitude, MAP, RPM, temperatura do ar, pressão de óleo e temperatura do motor.

Os resultados alcançados neste trabalho não apenas atestam a viabilidade técnica da abordagem adotada, mas também estabelecem uma base sólida para aplicações futuras em telemetria automotiva. Demonstrou-se que a combinação da extração de dados dos *frames* CAN com a transmissão eficaz via ESP-NOW resulta em um sistema confiável. A sinergia entre hardware e software, evidenciada pelos dados simulados exibidos com precisão e os comandos acionados, executados com sucesso, reflete o potencial de sistema para melhorar a experiência e segurança em competições de Formula SAE.



Tela de dados com valores simulados.



Menu de simulação FT Manager.

Figura 59 – Exibição dos dados simulados no volante e menu de simulação FT Manager.

Fonte: Autor

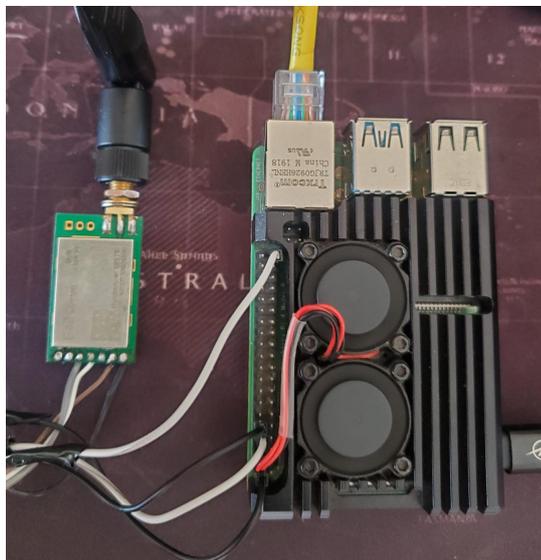


Figura 60 – Conexão entre LoRa e49 e Raspberry Pi.

Fonte: Autor

```

Shell
>>> %Run telemetria.py
Serial port opened: /dev/ttyS0
Connected with result code 0
{"Lat":-15.792318,"Lon":-47.855606,"Speed":0.81}
Failed to read from Serial: 'utf-8' codec can't decode byte 0x8a in position 0: invalid start byte
{"Lat":-15.792328,"Lon":-47.855606,"Speed":0.65}
{"MAP":65.41}
{"Oil Pressure":19.97}
{"Drag Speed":0.00}
{"Fuel Pressure":9.99}
{"TPS":100.00,"MAP":65.41,"AirTemp":136,"EngineTemp":133}
{"Air Temp":6536.80}
{"RPM":0.00}
{"Lat":-15.792334,"Lon":-47.855602,"Speed":0.94}
{"MAP":65.41}
{"Oil Pressure":19.97}
{"Drag Speed":0.00}
{"Fuel Pressure":9.99}
{"TPS":100.00,"MAP":65.41,"AirTemp":136,"EngineTemp":133}
{"RPM":0.00}
{"Air Temp":6536.80}
    
```

Figura 61 – Dados recebidos pelo LoRa e49.

Fonte: Autor

## 5 Conclusão

O presente trabalho aborda de forma detalhada a implementação e o desenvolvimento de um sistema embarcado utilizando tecnologias avançadas, como o protocolo CAN e o protocolo ESP-NOW. Ao longo deste estudo, foram explorados conceitos teóricos e práticos relacionados aos volantes automotivos esportivos, injeção eletrônica, comunicação CAN e a integração do protocolo ESP-NOW. Foi evidenciado o papel crucial dos sistemas embarcados na indústria automotiva, fornecendo soluções avançadas e inovadoras para o controle e o monitoramento de veículos de alto desempenho. Através da revisão teórica, foi possível compreender a evolução e a complexidade dessas tecnologias, bem como os benefícios que proporcionam em termos de desempenho, eficiência e segurança.

A implementação da extração de dado dos *frames* CAN e envio utilizando o protocolo ESP-NOW demonstrou resultados promissores e abriu caminho para futuras melhorias e otimizações. Esses resultados preliminares destacam o potencial do protocolo ESP-NOW como uma solução de comunicação sem fio confiável para a transmissão de dados provenientes do barramento CAN.

A implementação do display no volante, além de fornecer informações vitais como RPM, temperatura e pressão, integrou também funções de comando através de chaves e botões. Essa característica enriquece a interação do piloto com o veículo, permitindo ajustes rápidos e eficazes durante a condução. A combinação de visualização de dados e controles interativos transforma o volante em uma central de comando eficiente, essencial para a performance dos veículos Formula SAE.

A integração do sistema de telemetria, ainda em fase inicial, já revela um potencial notável. Este elemento possibilita a transmissão em tempo real de dados do veículo para a equipe nos boxes, viabilizando uma análise detalhada e decisões ágeis. Esta abordagem promissora não só visa aprimorar a eficiência na comunicação durante as corridas, mas também abre o caminho para avanços consideráveis em estratégias e performance. Com a equipe bem informada sobre as condições atuais do veículo, é possível oferecer suporte mais assertivo ao piloto, potencializando tanto a segurança quanto o desempenho nas competições.

Além destes resultados, o trabalho possui diversas áreas de melhoria como a validação do alcance do sistema de telemetria, adição de sensoriamento da tensão da bateria do Volante, expansão de protocolos CAN suportados pela FT600, adição de registros dos *frames* lidos localmente e utilização de modos de economia de energia para o volante. São melhorias que podem tornar o sistema ainda mais robusto e funcional para a aplicação pretendida.

Este trabalho representa um passo importante para o time Apuama Racing, trazendo inovações tecnológicas que prometem aprimorar significativamente a estratégia e o desempenho na pista. A aplicação dessas tecnologias não só oferece uma vantagem competitiva, mas também estabelece uma base sólida para futuras inovações. A equipe, agora equipada com ferramentas mais avançadas, está em uma posição melhor para tomar decisões informadas e otimizar o desempenho do veículo, refletindo o compromisso contínuo com a excelência e a evolução no automobilismo esportivo estudantil.

# Referências

- ATANASOVSKI, K. Esp-now communication protocol with esp32. Faculty of Technical Sciences - Bitola, 2021. Citado na página 18.
- BAK. *B18650CA*. 2011. Disponível em: <<https://www.recell.ro/Files/B18650CA.pdf>>. Acesso em: 10 dec. 2022. Citado na página 44.
- BRASIL, S. *Formulas SAE*. 2023. Disponível em: <<https://saebrasil.org.br/programas-estudantis/formula-sae-brasil/>>. Acesso em: 18 jun. 2023. Citado na página 19.
- BROWN, N. A. S. e. K. M. A. R. J. W. H. The evolution of steering wheel design in motorsport. University of Southampton, 2020. Citado 2 vezes nas páginas 20 e 22.
- CASADEVALL, A. G. Telemetry for etseib motorsport formula student cat09e. Universitat Politècnica de Catalunya, 2016. Citado na página 19.
- CHASE, O. *Sistemas Embarcados*. 2007. Disponível em: <<http://www.lyfreitas.com.br/ant/pdf/Embarcados.pdf>>. Acesso em: 15 jun. 2023. Citado na página 26.
- CHIPSKEY. *VK16E*. 2013. Disponível em: <<https://ardubotics.eu/en/communication-control/1091-vk16e-gmouse-gps-module-sirf3-chip-ceramic-antenna-ttl-signal-9600-baud.html>>. Acesso em: 09 out. 2022. Citado na página 40.
- CORP, M. eletronic. *LM1117-5V*. 2005. Disponível em: <<https://www.alldatasheet.com/datasheet-pdf/pdf/141839/CYSTEKEC/LM1117-5.0.html>>. Acesso em: 09 jun. 2023. Citado na página 43.
- DUARTE, J. V. C.; FIGUEIREDO, P. H. L. Desenvolvimento de arquitetura eletroeletrônica para veículo de pequena escala. Universidade de Brasília, 2022. Citado 3 vezes nas páginas 30, 34 e 41.
- EBYTE. *E49-400T20D User Manual*. 2019. Disponível em: <<https://www.ebyte.com/en/product-view-news.html?id=921>>. Acesso em: 18 jun. 2023. Citado na página 38.
- ELVANES, A. Electronics development of a formula type steering wheel. Metropolia University of Applied Sciences, 2022. Citado na página 19.
- EMBARCADOS. *Ambiente ESP32 no Windows*. 2018. Disponível em: <<https://embarcados.com.br/ambiente-esp32-no-windows/>>. Acesso em: 15 jun. 2023. Citado na página 39.
- EMBARCADOS. *Comunicação I2C*. 2023. Disponível em: <<https://embarcados.com.br/comunicacao-i2c/>>. Acesso em: 15 nov. 2023. Citado na página 29.
- ERIDANIE, D.; ROCHIM, A. F.; CESARA, F. N. Comparative performance study of esp-now, wifi, bluetooth protocols based on range, transmission speed, latency, energy usage and barrier resistance. Diponegoro University, 2021. Citado 3 vezes nas páginas 36, 37 e 44.

- ESPRESSIF. *ESP-IDF Programming Guide - ESP-NOW API*. 2016. Disponível em: <[https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp\\_now.html](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_now.html)>. Acesso em: 10 jun. 2023. Citado 3 vezes nas páginas 35, 36 e 38.
- ESPRESSIF. *FreeRTOS Overview*. 2018. Disponível em: <<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/freertos.html>>. Acesso em: 9 dec. 2023. Citado na página 52.
- ESPRESSIF. *ESP-NOW*. 2020. Disponível em: <<https://www.espressif.com/en/solutions/low-power-solutions/esp-now>>. Acesso em: 10 jun. 2023. Citado na página 35.
- ESPRESSIF. *ESP32 Series Datasheet*. 2023. Disponível em: <[https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)>. Acesso em: 10 jun. 2023. Citado na página 27.
- FOWLER, C. *MCP CAN LIB*. 2013. Disponível em: <[https://github.com/coryjfowler/MCP\\_CAN\\_lib](https://github.com/coryjfowler/MCP_CAN_lib)>. Acesso em: 09 jun. 2023. Citado na página 46.
- FUELTECH. *FT600*. 2016. Disponível em: <[https://fueltech.com.br/products/ft600-sfi-injecao-eletronica?gclid=CjwKCAjw-vmkBhBMEiwAlrMeF1sn7yyDMTbP4JPQMnimvLnV\\_auCVs3s1CwANz\\_2XsnlAHUu8APqIBoCPncQAvD\\_BwE](https://fueltech.com.br/products/ft600-sfi-injecao-eletronica?gclid=CjwKCAjw-vmkBhBMEiwAlrMeF1sn7yyDMTbP4JPQMnimvLnV_auCVs3s1CwANz_2XsnlAHUu8APqIBoCPncQAvD_BwE)>. Acesso em: 19 jan. 2023. Citado na página 25.
- FUELTECH. *FT600 FT 550 FT550 LITE FT450 Manual do Usuário*. 2016. Disponível em: <[https://ftmanager.fueltech.com.br/downloads/manual/Portugues/FT450\\_FT550\\_FT600.pdf?\\_gl=1\\*1xacreb\\*\\_ga\\_YGKB920TCN\\*MTY4OTAzOTgyMy4xLjEuMTY4OTAzOTgyMy42MC4wLjA.>](https://ftmanager.fueltech.com.br/downloads/manual/Portugues/FT450_FT550_FT600.pdf?_gl=1*1xacreb*_ga_YGKB920TCN*MTY4OTAzOTgyMy4xLjEuMTY4OTAzOTgyMy42MC4wLjA.>)>. Acesso em: 13 mai. 2023. Citado na página 46.
- FUELTECH. *FTCAN 2.0 Protocol*. 2022. Disponível em: <[https://files.fueltech.net/manuals/Protocol\\_FTSCAN20\\_Public\\_R022.pdf](https://files.fueltech.net/manuals/Protocol_FTSCAN20_Public_R022.pdf)>. Acesso em: 14 jun. 2023. Citado 4 vezes nas páginas 46, 47, 48 e 49.
- FUELTECH. *SwitchPanel 8 Mini*. 2023. Disponível em: <<https://fueltech.com.br/products/switchpanel-8-mini>>. Acesso em: 15 jun. 2023. Citado na página 50.
- INC., I. *MPU6050 Product Specification*. 2013. Disponível em: <<https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>>. Acesso em: 20 nov. 2023. Citado na página 43.
- INSTRUMENTS, T. *LM2596S-5.0*. 2013. Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/520535/TI1/LM2596S-5.0.html>>. Acesso em: 09 jun. 2023. Citado na página 40.
- INTERNATIONAL, S. *Formula SAE® Rules 2023*. 2022. Disponível em: <<https://fsaebrasil.net/>>. Acesso em: 24 ago. 2023. Citado na página 20.
- ITEAD. *Arduino Nextion Library*. 2015. Disponível em: <[https://github.com/itead/ITEADLIB\\_Arduino\\_Nextion](https://github.com/itead/ITEADLIB_Arduino_Nextion)>. Acesso em: 03 jun. 2023. Citado na página 51.

- MEGASATURNV. *HX-2S-D01*. 2017. Disponível em: <[https://github.com/megasaturnv/HX-2S-JH20-JH10-reverse-engineered\\_aka-2S-10A-Balance-Li-ion-Protection-PCB](https://github.com/megasaturnv/HX-2S-JH20-JH10-reverse-engineered_aka-2S-10A-Balance-Li-ion-Protection-PCB)>. Acesso em: 09 jun. 2023. Citado na página 44.
- M.P.H., I. *The MP4/6 at Goodwood FOS*. 2018. Disponível em: <<https://iberianmph.com/2018/01/16/mclaren-vintage/>>. Acesso em: 9 dec. 2023. Citado na página 21.
- NASCIMENTO, A. M. do. Projeto de melhoria de sistema de injeção eletrônica do motor de combustão interna bmw m50tub25 com uso de unidade controladora programável e conversão para etanol. UNIVERSIDADE DO ESTADO DO AMAZONAS, 2019. Citado 2 vezes nas páginas 22 e 23.
- NEXTION. *NX4827P043-011C*. 2022. Disponível em: <<https://nextion.tech/datasheets/nx4827p043-011c/>>. Acesso em: 09 jun. 2023. Citado 3 vezes nas páginas 41, 42 e 44.
- PIMPLE, P. Sniffing the automotive can bus for real-time data-logging and real time diagnostics display. University of Michigan-Dearborn, 2018. Citado na página 18.
- RYNDACKCOMPONENTES. *Módulo Acelerômetro e Giroscópio MPU6050*. 20? Disponível em: <[https://www.ryndackcomponentes.com.br/media/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/i/m/image\\_2524.jpg](https://www.ryndackcomponentes.com.br/media/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/i/m/image_2524.jpg)>. Acesso em: 9 dec. 2023. Citado na página 44.
- SILVA, R. R. da. Modelagem e análise de redes automotivas em ambiente virtual. Universidade de Brasília, 2015. Citado 4 vezes nas páginas 30, 31, 32 e 33.
- TIER, E. G. *DESENVOLVIMENTO DE SISTEMAS DE MONITORAMENTO EMPREGANDO TECNOLOGIA LORA EM UM PROTÓTIPO TIPO FORMULA SAE*. 2019. Disponível em: <[https://repositorio.ufsm.br/bitstream/handle/1/26881/Tier\\_Eduardo\\_Guasso\\_2019\\_TCC.pdf?sequence=1&isAllowed=y](https://repositorio.ufsm.br/bitstream/handle/1/26881/Tier_Eduardo_Guasso_2019_TCC.pdf?sequence=1&isAllowed=y)>. Acesso em: 18 jun. 2023. Citado 2 vezes nas páginas 28 e 29.