



**Tiago Henrique Bortoli**

**DESENVOLVIMENTO DE UM SISTEMA REMOTO PARA AQUISIÇÃO DE DADOS  
DE VEICULOS EM COMPETIÇÕES AUTOMOBILISTICAS**

Horizontina-RS

2022

**Tiago Henrique Bortoli**

**DESENVOLVIMENTO DE UM SISTEMA REMOTO PARA AQUISIÇÃO DE DADOS  
DE VEICULOS EM COMPETIÇÕES AUTOMOBILISTICAS**

Trabalho Final de Curso apresentado como requisito parcial para a o Trabalho Final de Curso na Engenharia de Controle e Automação da Faculdade Horizontina, sob a orientação do Prof. Douglas de Castro Karnikowski

Horizontina-RS

2022

Gratidão pelos meus pais, sua presença e amor incondicional na minha vida sempre. Esta monografia é a prova de que os esforços deles pela minha educação não foram em vão e valeram a pena.

“Se A é o sucesso, então A é igual a X mais Y mais Z. O trabalho é X; Y é o lazer; e Z é manter a boca fechada.” (Albert Einstein)

## RESUMO

Para auxiliar a tomada de decisão competições automobilísticas utilizam um sistema de telemetria para captação de dados tanto para tempo real quanto para análise futura. Tendo isso em mente foi desenvolvido um sistema de telemetria para aquisição de dados, em tempo real e para futuras referencias, de veículos *off Road* durante as corridas para um melhor aproveitamento de tempo e ter um melhor desempenho em competições, com a utilização de sensores para a aquisição do nível de combustível, temperatura da CVT e RPM (rotação por minuto) do motor, com a utilização de microcontrolador e tecnologia LoRa, resultando em uma aquisição de dados relevantes para veículos do tipo *off-Road*.

Palavra-chave: Telemetria, LoRa, Baja, Off-Road

## LISTA DE FIGURAS

Figura 1 - Esquema de um microcontrolador	14
Figura 2 - ECUs em um automóvel	15
Figura 3 - Microcontrolador STM32	16
Figura 4 - Arquitetura Cortex-M3	17
Figura 5 - Placas Arduino	19
Figura 6 - Arduino IDE	20
Figura 7 - Banda vs distância dos tipos de rede	21
Figura 8 - Representação da arquitetura da rede LoRaWAN	22
Figura 9 - Representação dos modos de transmissão	23
Figura 10 - Ambiente de desenvolvimento Proteus	25
Figura 11 - Método de desenvolvimento do projeto	26
Figura 12 - Fluxo de funcionamento do sistema	26
Figura 13 - Local de instalação do arquivo Arduino_STM32	28
Figura 14 - Local para inclusão da biblioteca	29
Figura 15 - Seleção do microcontrolados	29
Figura 16 - Diagrama elétrico	31
Figura 17 - Portas Chip LoRa	31
Figura 18 - PCB	32
Figura 19 - Modelo 3D da PCB com componentes	33
Figura 20 - Modelo 3D da PCB sem componentes	33
Figura 21 - Modelo 3D da PCB lado de traz	34
Figura 22 - Placa com componentes soldados	34
Figura 24 - Receptor	35

Figura 25 - Localização dos componentes no veículo	35
Figura 26 - Saída dos dados adquiridos	26

## **SUMÁRIO**

<b>1.</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1	TEMA	11
1.2	DELIMITAÇÃO DO TEMA	11
1.3	PROBLEMA DE PESQUISA	11
1.4	HIPÓTESES	12
1.5	OBJETIVOS	12
1.5.1	Objetivo geral	12
1.5.2	Objetivos específicos	12
1.6	JUSTIFICATIVA	12
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>14</b>
2.1	BAJA SAE BRASIL	14
2.2	SISTEMAS EMBARCADOS	14
2.2.1	Modulo eletrônico	16
2.2.2	Microcontrolador Stm32	17
2.2.3	Cortex-M3	18
2.2.4	Arduino	19
2.2.5	Ambiente de Desenvolvimento Integrado	20
2.3	REDE DE COMUNICAÇÃO	20
2.3.1	Tipos de transmissão	21
2.3.1.1	Rede LoRa	22
2.3.2	Modos de transmissão	23
2.4	TELEMETRIA	23
2.5	INTERNET DAS COISAS	24
2.6	PROGRAMA ELETRONICO PARA DESENVOLVIMENTO DE CIRCUITOS	24
2.6.1	PROTEUS	24
<b>3</b>	<b>METODOLOGIA</b>	<b>25</b>

<b>4</b>	<b>RECURSOS NECESSÁRIOS</b>	<b>27</b>
<b>5</b>	<b>APRESENTAÇÃO E ANÁLISE DOS RESULTADOS</b>	<b>28</b>
5.1	AMBIENTES DE DESENVOLVIMENTO	28
5.2	PREPARAÇÃO DA IDE ARDUINO	28
5.2	MODELAGEM DA ESQUEMÁTICA	30
<b>5.2.1</b>	<b>COMPARAÇÃO</b>	<b>30</b>
<b>5.2.2</b>	<b>MICROCONTROLADOR</b>	<b>30</b>
<b>5.2.3</b>	<b>LoRa</b>	<b>31</b>
<b>5.2.4</b>	<b>SENSORES</b>	<b>32</b>
5.3	MODELAGEM DA PLACA DE CIRCUITO IMPRESSO	32
5.4	Montagem	34
5.5	Testes	36
<b>6</b>	<b>Conclusão</b>	<b>37</b>
	<b>SUGESTÃO DE MELHORIA E TRABALHOS FUTUROS</b>	<b>38</b>
	<b>ANEXOS</b>	<b>39</b>

## 1. INTRODUÇÃO

O desenvolvimento tecnológico deu acesso a informações de forma remota, facilitando a aquisição de dados seja ele em tempo real ou não, reduzindo a necessidade de pessoas no local da coleta deles ampliando a eficiência da análise de informações e diminuindo o tempo de tomada de decisão.

Nos últimos anos especialmente em veículos Baja SAE, viu-se a necessidade de transmissão remota de dados instantâneos, que possibilitasse o monitoramento e a tomada de decisões em tempo real. Para isso, definiu-se o método de transmissão chamado telemetria, cujo funcionamento é baseado numa central de captação de informações, que recebe dados vindos de dispositivos remotos (VENTURA, LÓPEZ & GOROSO, 2013).

A telemetria veicular é um recurso que tem por finalidade contribuir com a coleta de informação sobre o veículo tais como a velocidade, a distância percorrida, o consumo médio de combustível, a temperatura do sistema mecânico etc. Com base nesses dados são tomadas as decisões e elaboradas estratégias para tornar tanto veículo quanto a pilotagem mais eficiente.

A aquisição de dados relevantes é importante tanto quanto a telemetria, por isso os dados adquiridos para o presente projeto são a aquisição dos dados de temperatura da CVT, rotação do motor e consumo de combustível.

Dados são informações sobre a história determinado objeto em determinado momento, portanto pode-se deduzir futuras ocorrências com base em informações passadas, aumentando assim a probabilidade da diminuição de falhas críticas sobre o objeto nas mesmas situações e descartando a necessidade de superdimensionamento.

Nesse estudo, é mostrado os passos para a implementação da telemetria com a tecnologia LoRa em conjunto com o microcontrolador de arquitetura ARM de modo que se optou pela capacidade e velocidade superior em comparação às do Arduino, permitindo assim futuros desenvolvimentos em relação ao sistema de aquisição, como acréscimo de outros sensores.

O atual projeto tem como finalidade apresentar as melhoras em empregar a arquitetura ARM tanto quanto a importância da transmissão de dados em tempo real e armazenamento das informações para análise futura a fim de promover possíveis melhorias no projeto do veículo Baja da equipe Sinuelo FAHOR.

## 1.1 TEMA

Utilização de um microcontrolador de baixo custo para coleta de dados e transmissão das informações através de radiofrequência.

## 1.2 DELIMITAÇÃO DO TEMA

Desenvolver um sistema para aquisição de informações para veículos *off-Road* com a utilização de um microcontrolador com base na arquitetura ARM, e transmitir os dados através da tecnologia LoRa.

## 1.3 PROBLEMA DE PESQUISA

O problema se caracteriza pela falta de informação sobre o carro na pista para a equipe no *pit stop*, além do armazenamento das referidas informações para futuras consultas, permitindo assim o acompanhamento para manutenção e situações adversas do funcionamento do veículo para tratamento preventivo, diminuindo assim a necessidade de manutenção e de problemas relacionados a defeitos mecânicos.

Um dos problemas que se encontra em projetos e para carros de estilo *off Road* é a falta de robustez em equipamentos eletrônicos, portanto busca-se resistência e capacidade de suportar testes de longa duração em terrenos de difícil acesso além de suportar um alto estresse juntamente com o veículo.

Em certos aspectos a Aquisição de dados se vê crucial para competições no estilo *off Road*, levando em conta que é uma necessidade do piloto saber as informações sobre o carro como por exemplo nível de combustível, temperatura do motor. Porém esses dados além de importante para o piloto são importantes para a equipe criar estratégias para futuras competições.

Observando o exposto, o problema de pesquisa tem por finalidade responder a seguinte pergunta: Como a aplicação de sistemas de aquisição de informações com microcontrolador ARM e transmissão através da tecnologia LoRa no veículo Baja auxiliam para seu melhor desempenho tanto durante a competição quanto para futuras competições?

## 1.4 HIPÓTESES

Hipótese 1: a utilização de arquitetura ARM para aquisição tem maior estabilidade.

Hipótese 2: a rede de transmissão LoRa impacta o desempenho do sistema no quesito de processamento.

## 1.5 OBJETIVOS

### 1.5.1 Objetivo geral

Desenvolver um sistema capaz de adquirir informações sobre o veículo e transmiti-las em tempo real ao mesmo período em que as arquiva para estudos posteriores.

### 1.5.2 Objetivos específicos

- I. Criar um sistema de telemetria para transmitir e receber os dados de maneira segura e acertiva;
- II. Desenvolver um sistema de medição e coleta de dados que garanta a confiabilidade
- III. Realizar testes do protótipo em ambiente real, a fim de comprovar seu funcionamento;

## 1.6 JUSTIFICATIVA

O presente estudo pode além de contribuir com a dirigibilidade e melhorar a performance do veículo Baja SAE em competições de sua categoria, colaborar com conhecimento sobre a comunicação de dados por radiofrequência com a tecnologia LoRa. Além de apresentar uma oportunidade de aplicação de telemetria na transmissão de dados de forma remota.

O intuito principal da telemetria é agilizar e melhorar os objetivos de uma aplicação, com a análise e tomadas de decisões feitas de forma remota, tendo assim

o controle da aplicação de qualquer lugar e não somente de onde se localiza a aplicação (SICHONANY et al., 2012).

Da mesma forma família de processadores Cortex-M3, que são os processadores utilizados no microcontrolador STM32, é ideal para produtos de microcontroladores. Isso inclui microcontroladores de baixo custo com tamanhos pequenos de memória e microcontroladores de alto desempenho com altas velocidades de operação (DA SILVA, 2020). Sendo assim a utilização de outro microcontrolador que não seja o Arduino torna-se viável.

Com a utilização da tecnologia apresentada, Cortex-M3, apoia a ideia de confiabilidade além de durabilidade provando que a utilização apresenta as características necessárias para o desenvolvimento do projeto além de apresentar aspectos positivos tanto para a área automotiva quanto para a de comunicação de dados.

Os aspectos que mais justificam a escolha do Cortex-M3 são a eficiência energética muito alta, baixa latência de interrupção, alta eficiência do processador, combinado as instruções Thumb-2 para manipulação de campo de bits j a escolha da tecnologia LoRa se baseia no baixo custo, além de baixo consumo energético, enquanto redes como ZigBee e WiFi tem alcance muito menor e custos de implementação e consumo mais altos.

Perante o exposto, o projeto justifica se pela avaliação da viabilidade da utilização de novas tecnologias que proporcionam um melhoramento do desempenho relacionado a baixo custo e credibilidade.

## 2 REFERENCIAL TEÓRICO

### 2.1 BAJA SAE BRASIL

O programa Baja SAE BRASIL é um desafio lançado aos estudantes de Engenharia para a aplicação prática dos conhecimentos obtidos na sala de aula. Ao participar do Baja SAE, o aluno participa em um caso real de desenvolvimento de um veículo *off Road*, desde sua concepção ao projeto detalhado, construção e testes. Os alunos que desejam participar devem formar equipes que representarão a Instituição de Ensino Superior à qual estão ligados. Estas equipes são desafiadas anualmente a participar da competição, que reúne os estudantes e promove a avaliação comparativa dos projetos, a competição nacional recebe o nome de Competição Baja SAE BRASIL e as competições regionais são nomeadas como Etapa Sul, Sudeste e Nordeste. As equipes vencedoras do Baja SAE BRASIL terão a oportunidade de viajar e mostrar seu projeto a profissionais estrangeiros na competição internacional, nos Estados Unidos (SAE BRASIL, 2021)

A Equipe de Baja, SINUELO FAHOR, foi criada em abril de 2006, através da iniciativa do Professor Daniel Werner Zacher, membro da diretoria da SAE Brasil – Seção Porto Alegre na qual estimulou acadêmicos a montar uma equipe para participar das competições a nível regional e nacional. A equipe é formada por acadêmicos voluntários dos cursos de engenharias da FAHOR. O projeto é dividido em núcleo de negócios e núcleo técnico. O núcleo de negócios é composto pelo gerenciamento do projeto, vendas e marketing e planejamento orçamentário. Já o núcleo técnico abrange as áreas de *frame/body*, suspensão e direção, freios, *powertrain*, design e conforto, eletroeletrônica e validação física e numérica (EQUIPE SINUELO, 2021).

### 2.2 SISTEMAS EMBARCADOS

Segundo Dos Reis (2015) um sistema embarcado é um sistema de computador (hardware + software) que possui uma função dedicada, geralmente

operando dentro de um outro sistema, elétrico, mecânico, hidráulico. Sistemas embarcados são usados para controlar muitos tipos de sistemas e equipamentos atualmente, além de encontrar aplicações muito variadas, desde a área médica até comunicação e entretenimento.

A utilização da eletrônica em automóveis tem trazido novas aplicações e opcionais que são oferecidas aos consumidores. A cada ano essas novidades na eletrônica aplicado em veículos automotores tem superado os quesitos como desempenho, confiabilidade e conforto. Em alguns casos inclusive, a utilização da eletrônica facilitou a montagem de alguns componentes em seus produtos. Porém, nem tudo ligado a eletrônica ainda é um assunto simples de se discutir, pois muito ainda há de se difundir tanto nos processos de fabricação como na forma de melhorar a vida do cliente final (DOS ANJOS, 2011)

De acordo com Dos Reis (2015) computadores embutidos, às vezes chamados de microcontroladores, podem ser considerados dispositivos compostos por um processador dedicado com finalidade de executar uma aplicação específica, diferentemente de computadores de propósito geral. Para tal, esses dispositivos apresentam, em menor quantidade, os mesmos elementos de um computador reunidos em um único chip.

O microcontrolador pode ser visto como uma caixa preta, em que as entradas são fornecidas ao sistema, processadas, e por fim geram uma ação, representada pelas saídas. Nesse caso, entradas e saídas são representadas como: sensores, atuadores, dispositivos de comunicação, dispositivos de interface gráfica, entre outros, como demonstrado na Figura 1 (DOS REIS, 2015)

Figura 1 - Esquema de um microcontrolador



Fonte: Fabio dos Reis, 2015.

### 2.2.1 Módulo eletrônico

Segundo Guimarães (2007) os módulos eletrônicos são os dispositivos responsáveis pela leitura das entradas, acionamento das saídas e pelo gerenciamento do funcionamento dos protocolos de comunicação utilizados nos veículos.

Figura 2 - ECUs em um automóvel



Fonte: Leonardo Contesini, 2017.

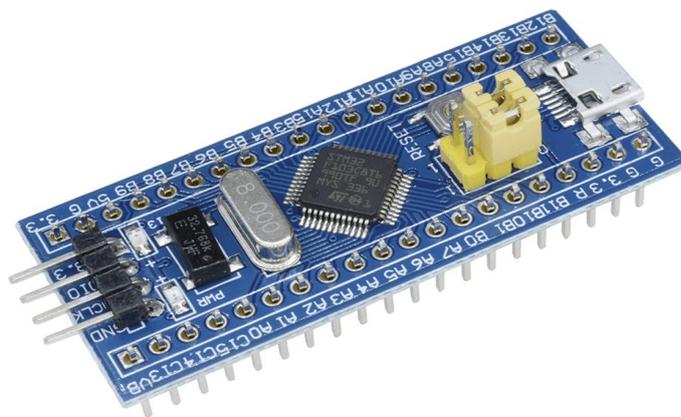
Segundo Contesini (2017) os módulos eletrônicos automotivos são comumente chamados de ECU significa *electronic control unit*, ou unidade de controle eletrônico. Atualmente os carros modernos podem ter até 70 ECUs diferentes como demonstrado na Figura 2, com tantas interligadas, o nome “ECU”, na aplicação automotiva, caiu em desuso e foi substituído por siglas mais específicas, a ECU do motor, por exemplo, hoje é mais conhecida como “ECM” (*Engine Control Module* ou “módulo de controle do motor”).

### 2.2.2 Microcontrolador Stm32

Como o nome STM32 sugere são dispositivos de 32 bits e são consideravelmente mais poderosos que os dispositivos de 8 bits. O F103 é um ramo (F1+03) da plataforma STM32. Esta subdivisão define a CPU e os recursos periféricos do dispositivo. Por último, o sufixo C8T6 define ainda mais os recursos do dispositivo, como a capacidade de memória e velocidade do *clock* do dispositivo (DA SILVA, 2020)

Levando em conta o tamanho da placa e a quantidade de entradas e saídas como demonstrado na figura 3 aumenta a capacidade para futuras adições de sensores. Os equipamentos baseados em ARM (Advanced RISK Machine), no geral, são conhecidos por otimizarem as respostas, possibilitando a reação em tempo real (COMERCIAL, 2020).

Figura 3 - Microcontrolador STM32

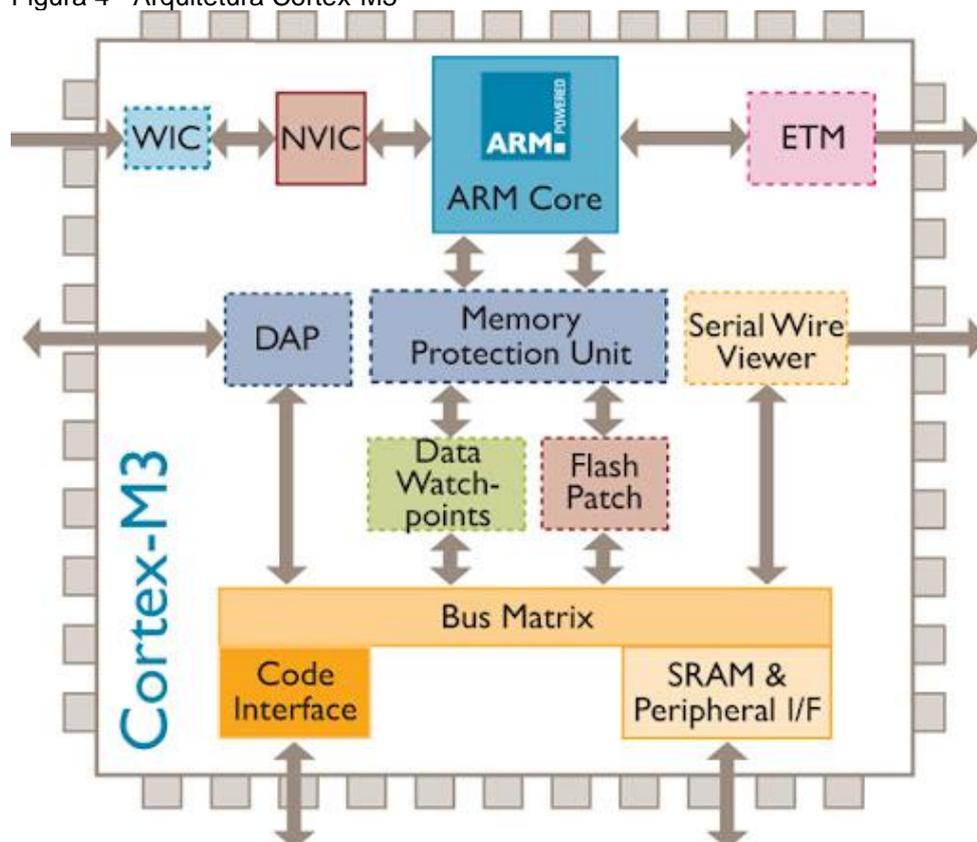


### 2.2.3 Cortex-M3

Segundo Da Silva (2020) os processadores Cortex-M3 e Cortex-M4 são baseados na arquitetura ARMv7-M. Ambos são processadores de alto desempenho projetados para microcontroladores. Nesse caso o processador ARM (Advanced RISC Machines Ltd) sendo mais adequado por permitir que o processamento seja realizado com menos instruções em relação a outros processadores como da família ATmega, como mostra a figura 4 que apresenta sua arquitetura interna.

Uma Instruction Set Architecture (ISA) é parte do modelo abstrato de um computador. Ele define como o software controla a CPU. A família Arm ISA permite que os desenvolvedores escrevam software e firmware em conformidade com as especificações Arm, com a certeza de que qualquer processador Arm irá executá-lo da mesma maneira. Esta é a base da promessa de portabilidade e compatibilidade do Arm, subjacente ao ecossistema do Arm. (ARM DEVELOPER LOGO, 2020)

Figura 4 - Arquitetura Cortex-M3



Fonte: Embedded Insights, 2012.

Os processadores Cortex-M3 possuem um endereçamento de 32 bits, suportando assim 4 GB de espaço em memória, suporte ao modo de suspensão e vários recursos para baixo consumo de energia, um controlador de interrupção que

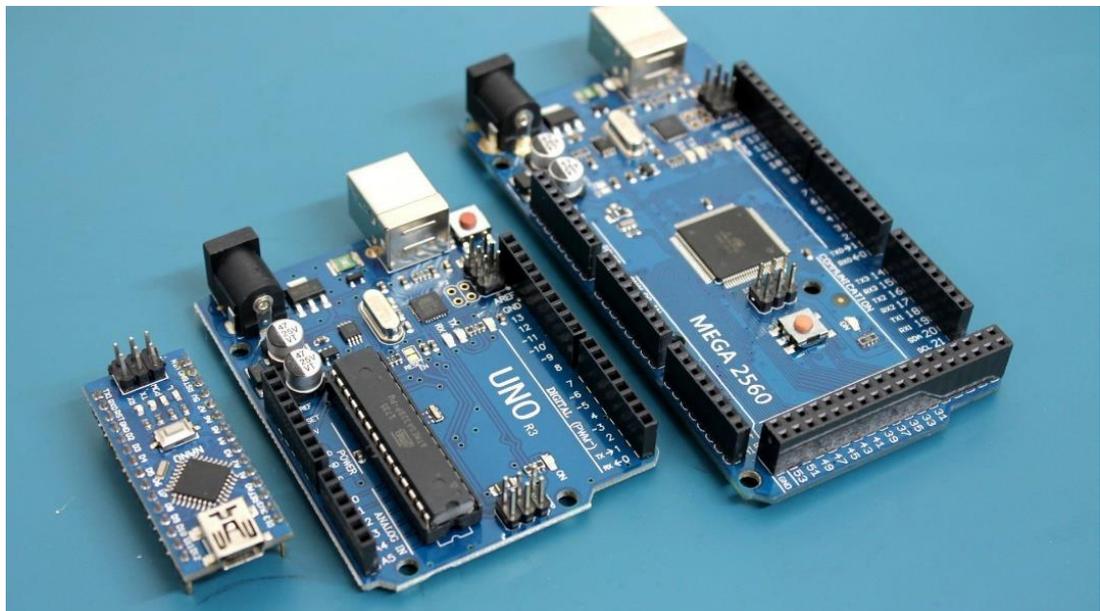
suporta até 240 solicitações de interrupção e de 8 a 256 níveis de prioridade de interrupção, além da opção de ser usado em projetos de único processador ou multiprocessador (DA SILVA, 2020)

#### 2.2.4 Arduino

O Arduino é outro tipo de microcontrolador utilizado para prototipagem e aprendizado, a plataforma Arduino possui vários modelos, porém alguns são mais conhecidos como por exemplo o Arduino Uno, Arduino Mega e o Arduino Mini, demonstrado na figura 5.

Arduino Uno é a placa mais conhecida e mais utilizada ela simples, o Arduino Mini é difere da versão Uno pela ausência da conexão USB própria que a Uno possui enquanto o Arduino Mega é a maior placa da família Arduino o qual é recomendado para projetos mais complexos.

Figura 5 – Placas Arduino



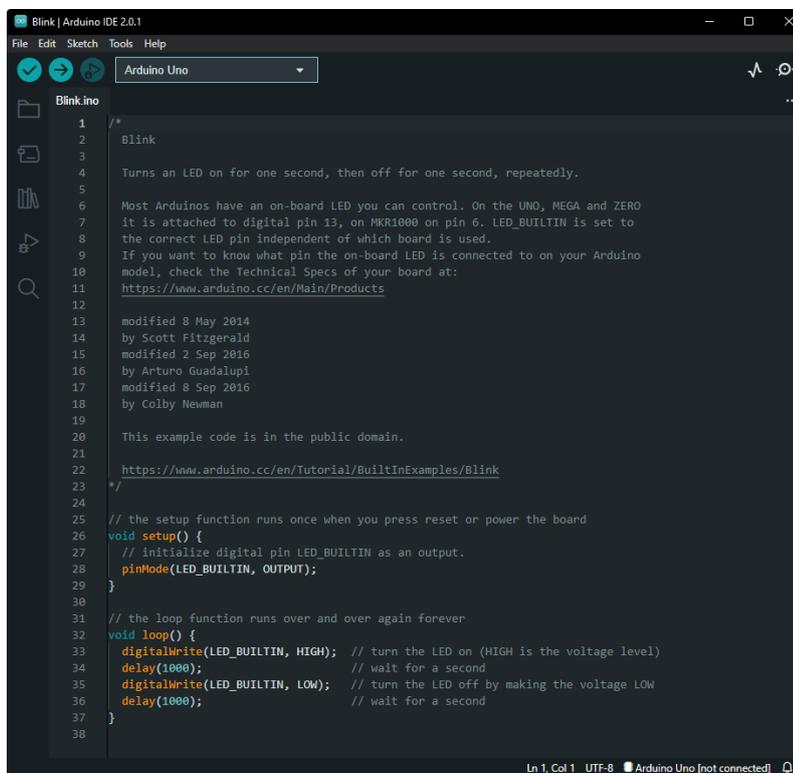
Fonte: Matheus Gebert Straub, 2017.

Cada um dos microcontroladores possui um processador diferente o Arduino mega possui o ATmega2560 enquanto o Arduino Uno possui o ATmega328 e por último o Arduino mini possui o mesmo do Arduino Uno, porém na versão SMD.

## 2.2.5 Ambiente de Desenvolvimento Integrado

Segundo Lima (2021) IDE (ambiente de desenvolvimento integrado) é um software para o desenvolvimento para editar o código de desenvolvimento em uma única interface gráfica, uma IDE tem o intuito de facilitar o processo de desenvolvimento de software.

Figura 6 – Arduino IDE



```

1  /*
2  3  Blink
4  5  Turns an LED on for one second, then off for one second, repeatedly.
6  7  Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
7  8  it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
8  9  the correct LED pin independent of which board is used.
9  10 If you want to know what pin the on-board LED is connected to on your Arduino
10 11 model, check the Technical Specs of your board at:
11 12 https://www.arduino.cc/en/Main/Products
12 13
13 14 modified 8 May 2014
14 15 by Scott Fitzgerald
15 16 modified 2 Sep 2016
16 17 by Arturo Guadalupi
17 18 modified 8 Sep 2016
18 19 by Colby Newman
19 20
20 21 This example code is in the public domain.
21 22
22 23 https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
23 24 */
24 25 // the setup function runs once when you press reset or power the board
25 26 void setup() {
26 27   // initialize digital pin LED_BUILTIN as an output.
27 28   pinMode(LED_BUILTIN, OUTPUT);
28 29 }
29 30
30 31 // the loop function runs over and over again forever
31 32 void loop() {
32 33   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
33 34   delay(1000); // wait for a second
34 35   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
35 36   delay(1000); // wait for a second
36 37 }
37 38

```

Fonte: Autor, 2022.

Existem várias IDEs no mercado algumas mais conhecidas e utilizadas do que outras inclusive o Arduino possui uma IDE própria para o desenvolvimento dos códigos, porém o Arduino IDE é focado em linguagem C, como demonstrado na figura 6.

## 2.3 REDE DE COMUNICAÇÃO

Segundo Neumann et al. (2021) a comunicação de dados é o processo de transferência de informações entre dispositivos. Para que ocorra a transferência de

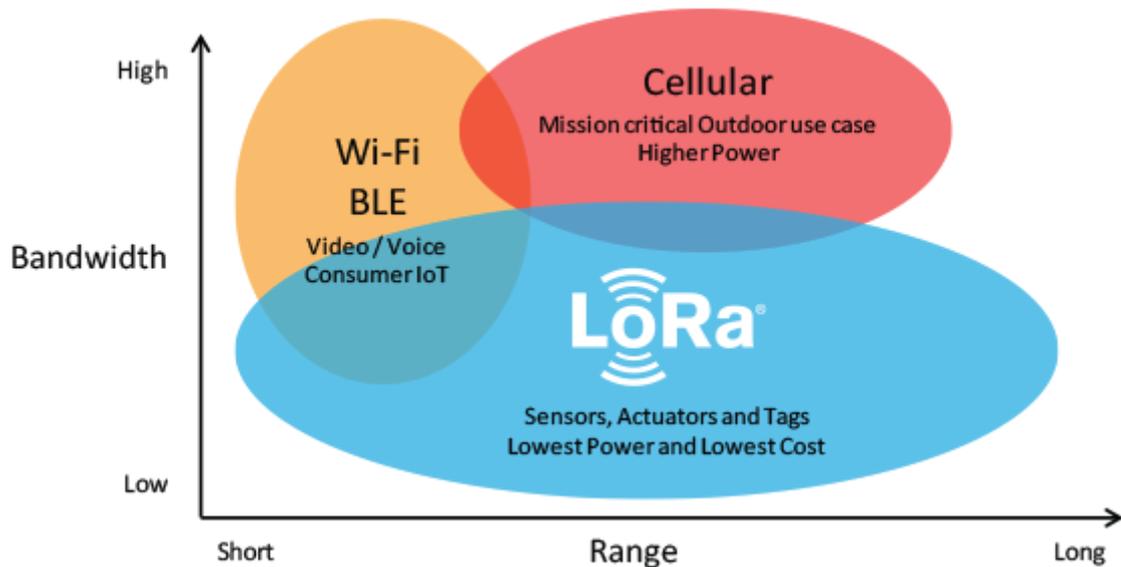
dados requer um meio de transporte entre os dispositivos que desejam se comunicar, como cabos de fibra ótica, fio de cobre e sinais sem fio.

Diferentes métodos e padrões de comunicação sem fio foram desenvolvidos em todo o mundo, com base em vários requisitos comerciais, como aplicação específica e faixa de transmissão. Essas tecnologias podem ser classificadas em quatro categorias individuais: Rede de área pessoal sem fio (WPAN), Rede de área local sem fio (WLAN), Rede de área metropolitana sem fio (WMAN), Rede de área ampla sem fio (WWAN). (NEWARK, 2021)

### 2.3.1 Tipos de transmissão

As principais tecnologias de transmissão sem fio são RFID, NFC, Bluetooth, ZigBee, Wi-Fi, celular, SigFox e LoRa. Elas possuem suas principais diferenças no alcance e complexidade de implementação (NEWARK, 2021), como apresentado na figura 7.

Figura 7 – Banda vs distância dos tipos de rede



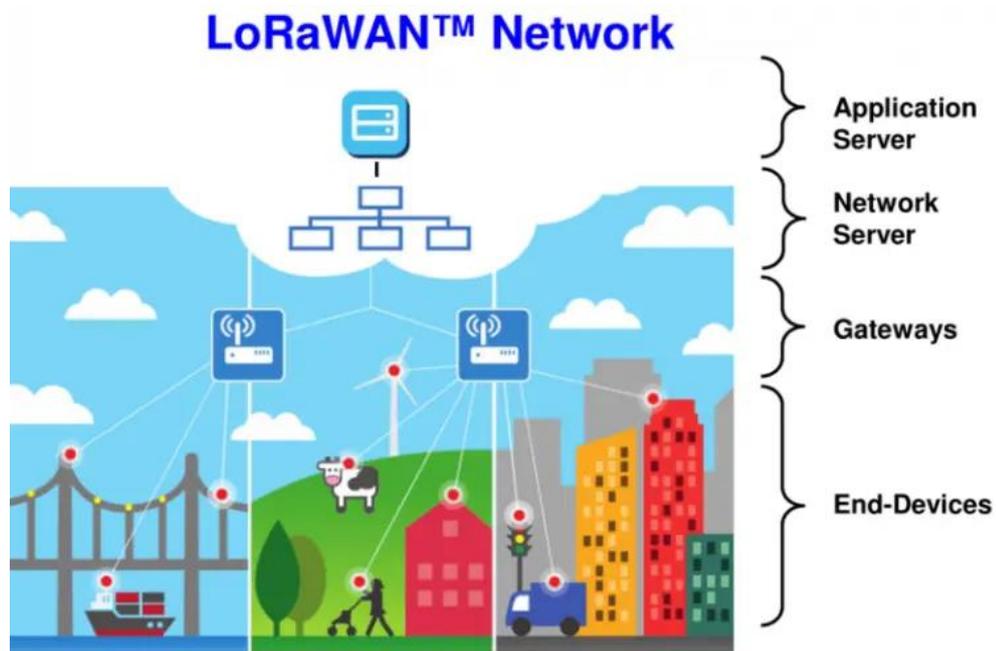
Fonte: Autor, 2022.

A melhor maneira de transmissão é por Wi-Fi (CANSSI et al., t. al. (2017). Essa forma não deixa de ser por radiofrequência, pois seu sinal é transferido por frequência. Porém, o que distingue, é que o Wi-Fi manda e recebe dados (Half-Duplex), já a radiofrequência apenas manda ou recebe dados (Simplex).

### 2.3.1.1 Rede LoRa

LoRa é uma tecnologia sem fio desenvolvida para permitir que comunicações de baixa taxa de dados sejam feitas em longas distâncias por sensores e atuadores para aplicações M2M e Internet das Coisas. Uma rede LoRa pode ser organizada para fornecer cobertura semelhante à de uma rede celular. Os aplicativos da tecnologia sem fio LoRa incluem medição inteligente, rastreamento de estoque, máquina de venda automática, dados e monitoramento, indústria automotiva, aplicativos de serviços públicos onde relatórios e controle de dados podem ser necessários, como exemplificado na figura 8. (NEWARK, 2021)

Figura 8 - Representação da arquitetura da rede LoRaWAN



Fonte: Vidal Pereira da Silva Junior, 2016.

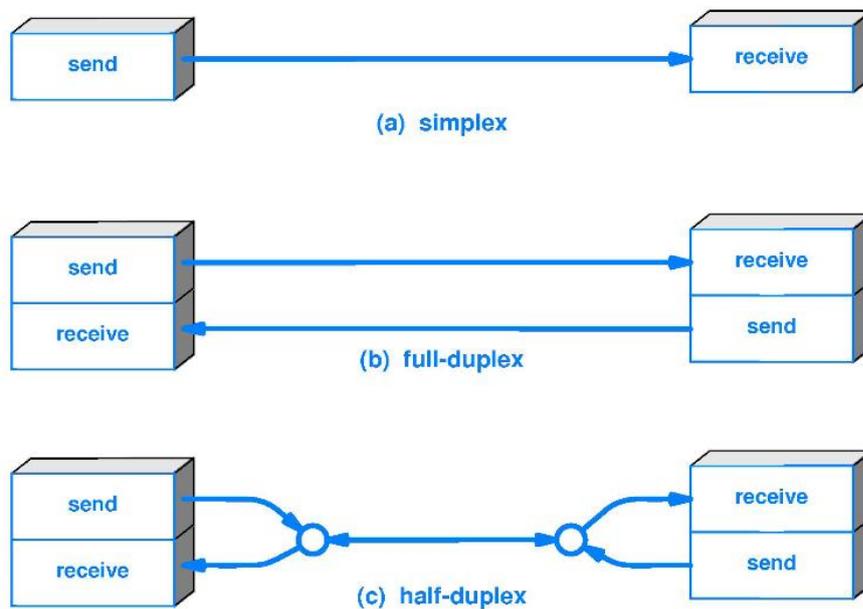
A tecnologia LoRa usa radio frequência o que permite a comunicação a longas distancias, em áreas urbanas 4 Km a até 12 km de alcance em áreas rurais com baixo consumo de energia. LoRaWAN é o nome dado ao protocolo que define a

arquitetura do sistema bem como os parâmetros de comunicação usando a tecnologia LoRa esse protocolo implementa os detalhes de funcionamento, segurança, qualidade do serviço, ajustes de potência visando maximizar a duração da bateria dos módulos, e os tipos de aplicações tanto do lado do módulo quanto do servidor (JUNIOR, 2016)

### 2.3.2 Modos de transmissão

Existem três modos de transmissão simplex, Half-duplex e full-duplex. O modo de transmissão descreve a direção do fluxo do sinal entre dois dispositivos conectados. A principal diferença é que, em um simplex a comunicação é unidirecional, já no Half-duplex a comunicação é bidirecional, mas o canal é usado alternadamente pelo dispositivo conectado. Por outro lado, no full-duplex a comunicação é bidirecional e o canal é usado pelo dispositivo simultaneamente, exemplificado na figura 9. (NEUMANN et al., 2021)

Figura 9 - Representação dos modos de transmissão



Fonte: Black Box, 2021.

## 2.4 TELEMETRIA

Segundo Gomes (2019) telemetria significa a arte de medir coisas. Dentro da indústria, o termo é utilizado como uma tecnologia que permite a medição remota e a

comunicação de informações entre sistemas, através de dispositivos de comunicação sem fio, como ondas de rádio ou sinais de satélite.

A telemetria, um sistema de monitoramento e medição à distância, permite a cientistas e engenheiros monitorar as características físicas como temperatura, velocidade, aceleração e pressão em mísseis ou veículos espaciais controlados remotamente. (NEUMANN et al., 2021)

## 2.5 INTERNET DAS COISAS

Segundo De Moraes et al. (2018) o termo IoT, do inglês Internet of Things e que é traduzido como Internet das Coisas, é utilizado para transmitir a ideia de que a internet pode estar presente em todas as coisas. O conceito de IoT é baseado na ideia de fusão do mundo real com o mundo digital, fazendo com que os indivíduos estejam em constante comunicação e interação com outras pessoas e objetos.

A IoT é uma nova onda tecnológica que veio complementar a internet tradicional, oferecendo acesso de baixa velocidade para elementos simples de nosso dia a dia, ampliando assim a utilização da internet para algo além da pesquisa ou diversão, passando também a desenvolver e ampliar negócios e oportunidades. (LEITE; MARTINS; URSINI, 2017)

## 2.6 PROGRAMA ELETRONICO PARA DESENVOLVIMENTO DE CIRCUITOS

Segundo Gianfagna (2022) EDA (Electronic Design Automation), é um segmento de mercado que consiste em software, hardware e serviços com o objetivo coletivo de auxiliar no desenvolvimento de dispositivos semicondutores ou chips.

Existem muitos softwares EDA no mercado atualmente com alguns tendo maior destaque que outros, portanto tendo maior documentação e sendo de fácil utilização que outros.

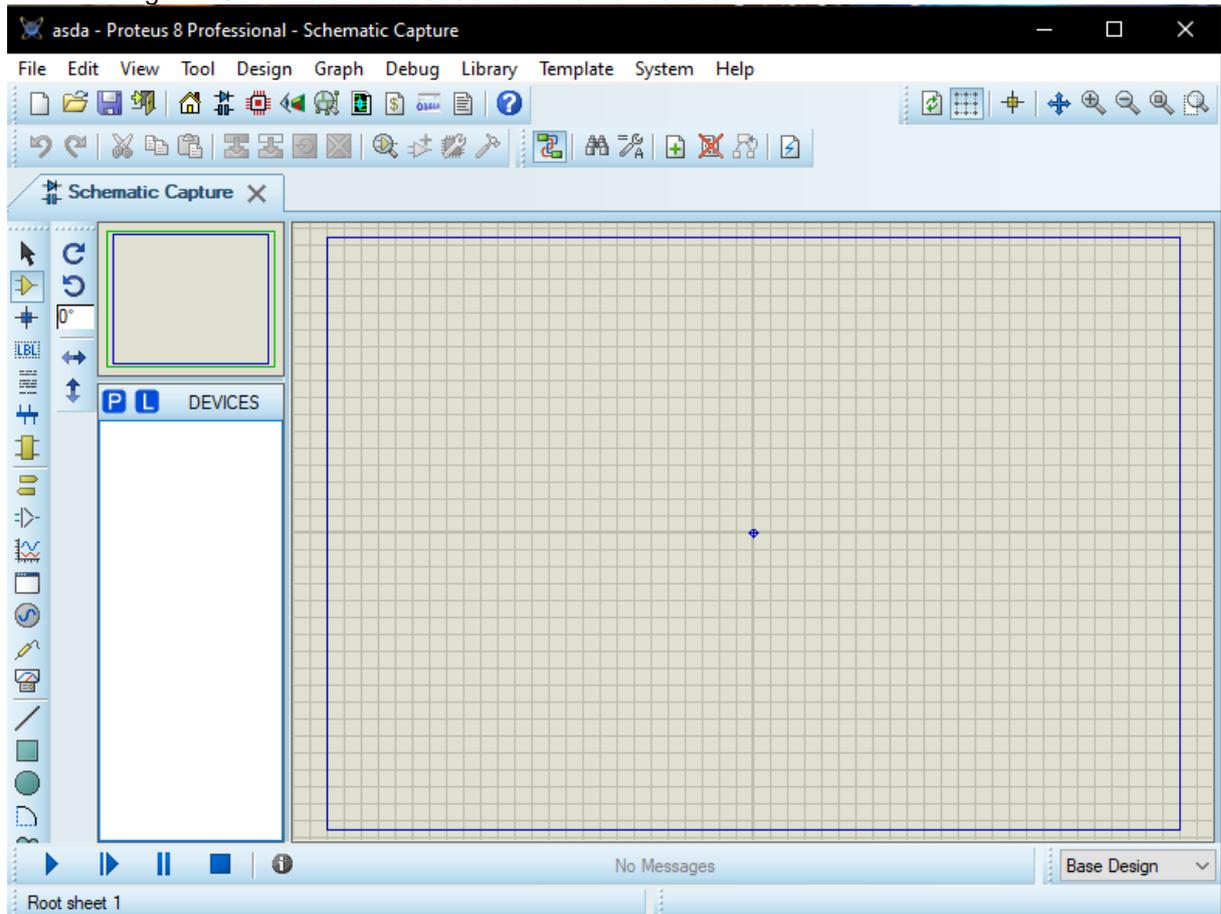
### 2.6.1 PROTEUS

De acordo com Lima e Furtado (2011), Proteus é uma ferramenta EDA, desenvolvido pela Labcenter Electronics, que permite criar a esquemática de projetos,

elaborar PCBs (placas de circuito impresso), visualização 3D do projeto, simulações e várias outras ferramentas

Segundo Zarth (2021) o Proteus apresenta uma grande variedade de componentes e possui fidelidade em suas simulações, podendo assim ser testado sem a necessidade de prototipar o sistema ou manufaturá-lo.

Figura 10 – Ambiente de desenvolvimento Proteus



Fonte: Autor, 2022.

### 3 METODOLOGIA

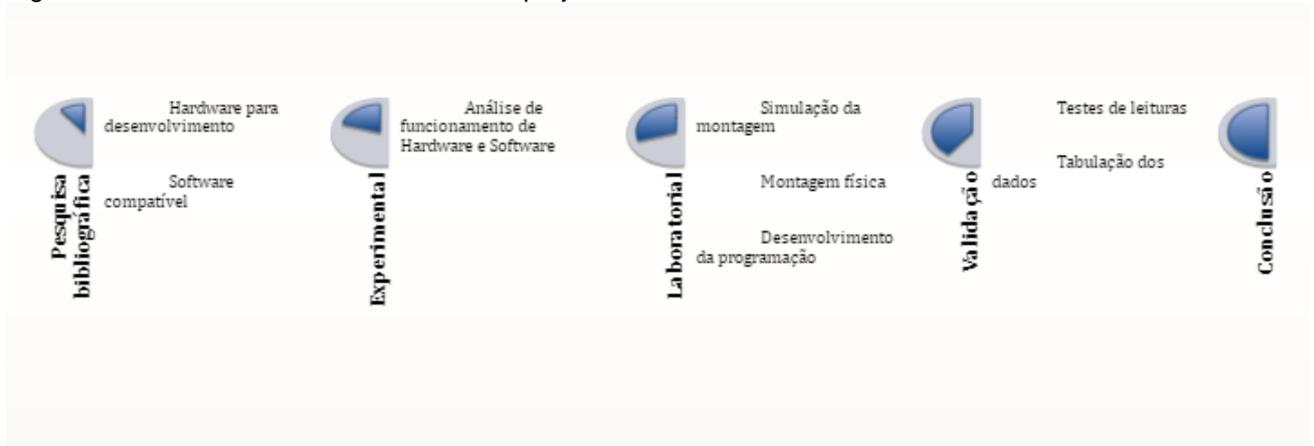
A pesquisa a que se propõe o estudo foi desenvolvida no veículo Baja, da Faculdade Horizontina, localizado no município de Horizontina-RS no Noroeste do Estado do Rio Grande do Sul, nos anos de 2021 e 2022.

Para se conseguir atingir o objetivo proposto, métodos, procedimentos e técnicas de pesquisa foram utilizados. Assim, para uma maior compreensão do estudo estes métodos e técnicas foram explicitados a seguir.

O método de abordagem utilizado no presente trabalho foi o exploratório o qual é demonstrado na Figura 12. Os procedimentos seguidos para a realização da

pesquisa foram a pesquisa bibliográfica, pois se consultou todos os tipos de obras escritas sobre os assuntos abordados adquirindo dessa forma maior conhecimento sobre os mesmos; o experimental por meio de testes de formulações de testes a serem feitos; o laboratorial onde utilizou-se de instrumentos laboratoriais para a manipulação das variáveis pesquisadas e a pesquisa estatística onde após a coleta de dados quantitativos estes foram conduzidos a manipulação estatística gerando médias.

Figura 11 - Método de desenvolvimento do projeto



Fonte: Autor, 2021.

Para realização do trabalho, foi usado o microcontrolador STM32 e Módulo LoRa E32. A implementação foi feita no veículo base da equipe Sinuelo da FAHOR. O protótipo foi desenvolvido de acordo com o fluxograma da Figura 13; onde os dados foram coletados através de 3 tipos de sensores a leitura. Os tratamentos dos dados foram feitos pelo microcontrolador STM32 que também está no controle de um módulo E32 o qual por sua vez foi responsável pela Transmissão para outro módulo LoRa Ligado ao outro STM32, demonstrado na figura 12.

Figura 12 - Fluxo de funcionamento do sistema



Fonte: Autor, 2021.

O responsável pela decodificação dos dados foi responsável também pelo armazenamento deles em um cartão micro SD além de apresentar os dados em tempo real através da serial.

Para análise de funcionamento do microcontrolador e do módulo LoRa foram realizados testes de bancada para análises comportamentais de ambos, após compreender o seu funcionamento foram iniciadas simulações em software para o desenvolvimento de uma placa central, para ser instalada no veículo, para a conexão de todos os componentes, para recepção de dados foi desenvolvido outra placa é com conexão para o computador, entrada para cartão micro SD na qual possui uma bateria integrada para o caso de não haver energia no local além de não precisar necessariamente do uso de um computador em tempo integral conectado ao sistema de recepção.

Com relação à montagem foi seguida especificações nas diretrizes da competição Baja SAE da parte elétrica, para tanto os testes feitos em bancadas foram replicados no carro, como posterior implementação para testes e aquisição de dados dos quais foram feitas tabelas para análises e futura conclusão da implementação do sistema.

O código desenvolvido para o funcionamento do projeto foi feito em um compilador para códigos em C desenvolvido para a plataforma Arduino, após testes e conclusões de funcionamento do código foi desenvolvido um novo código específico para o microcontrolador STM32 em um compilador específico com maior possibilidade de controle sobre o microcontrolador.

Os testes foram realizados em ambientes controlados além de serem realizados em pistas com diferentes terrenos podendo assim resultar em uma gama diferente de respostas aumentando a confiabilidade e diminuindo imprevistos em implementações futuras do sistema. Para análise foram utilizados gráficos e cálculos de média para confirmação da utilização do sistema de telemetria em veículos *off Road*.

#### **4 RECURSOS NECESSÁRIOS**

Nesta seção são listados os principais materiais e equipamentos utilizados no desenvolvimento do protótipo e durante a fase de testes deste projeto científico.

- a) Módulos LoRa – E32;
- b) Microcontrolador STM32f103t6c8;

- c) Softwares de programação em linguagem C;
- d) Software de Eletroeletrônica e Simulação;
- e) Manuais de instruções dos componentes adquiridos;

## 5 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Nessa seção são abordados os resultados do projeto e planejamentos discutidos anteriormente, mostrando o desenvolvimento e a validação além dos resultados obtidos através das simulações e testes aplicados.

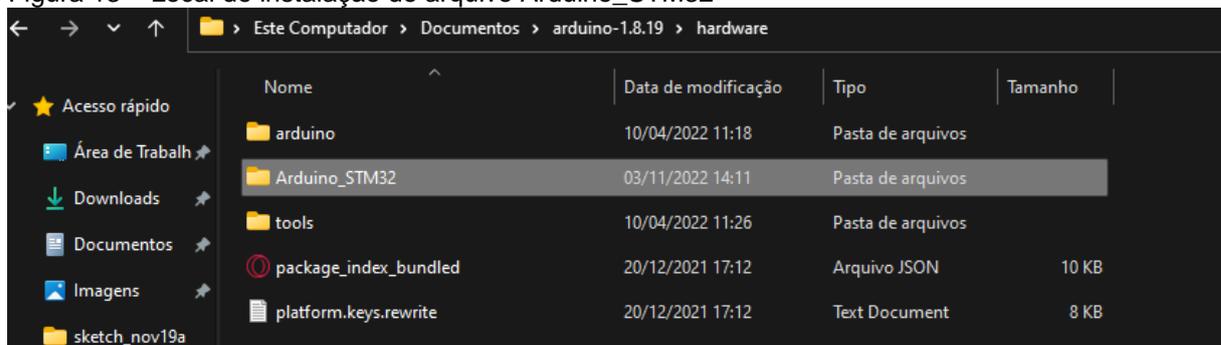
### 5.1 AMBIENTES DE DESENVOLVIMENTO

Para o desenvolvimento do projeto inicialmente foi necessário o preparo do ambiente de desenvolvimento dentro do Proteus, criando um projeto novo e em branco, o próximo passo foi pegar todos os componentes necessários para criar o esquemático e desenvolver a PCB. Além do Proteus foi necessário a utilização da IDE Arduino para o desenvolvimento do código utilizado para o funcionamento do protótipo.

### 5.2 PREPARAÇÃO DA IDE ARDUINO

Como a IDE Arduino é desenvolvida para suas placas não possui por padrão as placas STM32 sendo assim necessário a configurar a IDE para o STM32. Acessando o GitHub e procurando pelo arquivo Arduino\_STM32, foi possível baixar um arquivo compactado que deverá ser descompactado dentro da pasta hardware da IDE Arduino, possibilitando assim a utilização dos microcontroladores da família STM32, exemplificado na figura 13.

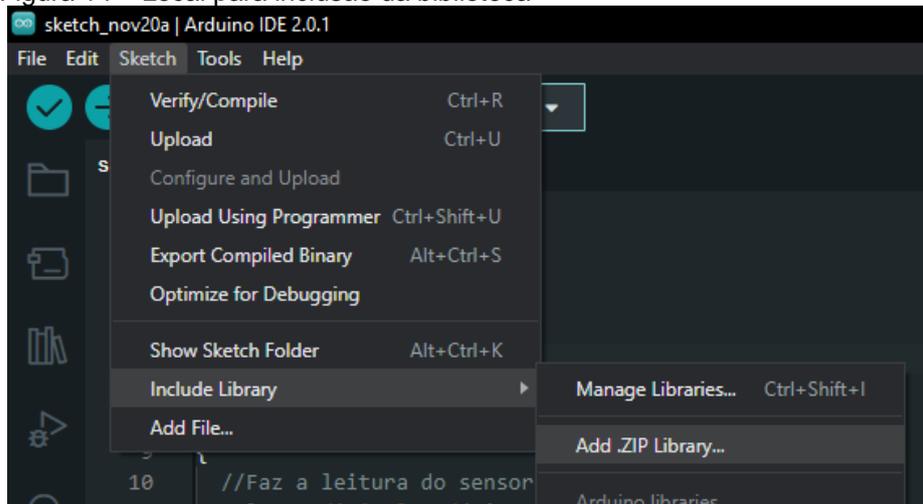
Figura 13 – Local de instalação do arquivo Arduino\_STM32



Fonte: Autor, 2021.

Também foi necessário baixar e instalar a biblioteca Thermistor para o funcionamento do sensor de temperatura, a qual também pode ser encontrada no GitHub, porém possui um método um pouco diferente para a instalação que deve ser feita pela própria IDE Arduino onde serra necessário somente seguir o caminho mostrado na figura seguinte e selecionar o arquivo e confirmar.

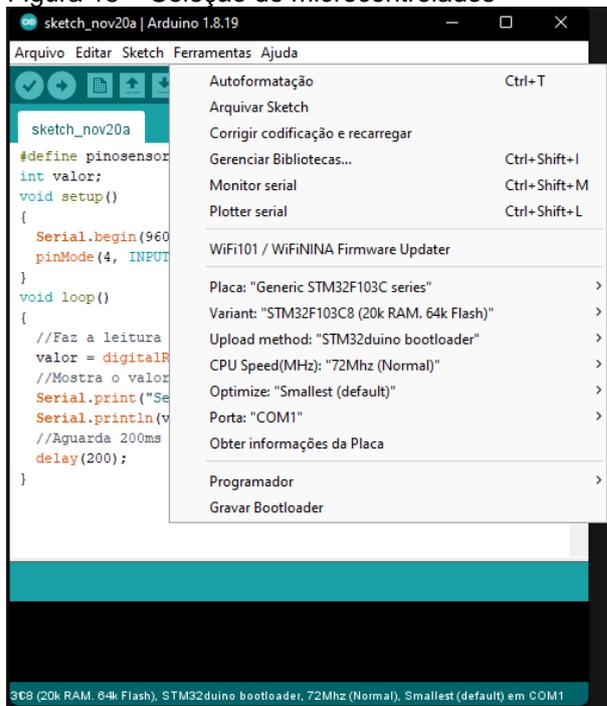
Figura 14 – Local para inclusão da biblioteca



Fonte: Autor, 2021.

Após todos os arquivos instalados é necessário selecionar a placa desejada para ser possível passar o código para o microcontrolador desejado como exemplificado na figura 15.

Figura 15 – Seleção do microcontrolados



Fonte: Autor, 2021.

## 5.2 MODELAGEM DA ESQUEMÁTICA

Na escolha dos componentes para o desenvolvimento da PCB foi pensado em como agregar todos os sensores necessários, o microcontrolador e o LoRa em uma única placa, por isso foram utilizados tipos de conectores diferentes.

### 5.2.1 COMPARAÇÃO

Para a escolha do microcontrolador utilizado foi feita uma comparação das especificações entre o STM32F103 e os mais populares da família Arduino, optando assim o microcontrolador com as melhores velocidades de processamento, mostrado no quadro 1.

Quadro 1 – Comparação entre microcontroladores

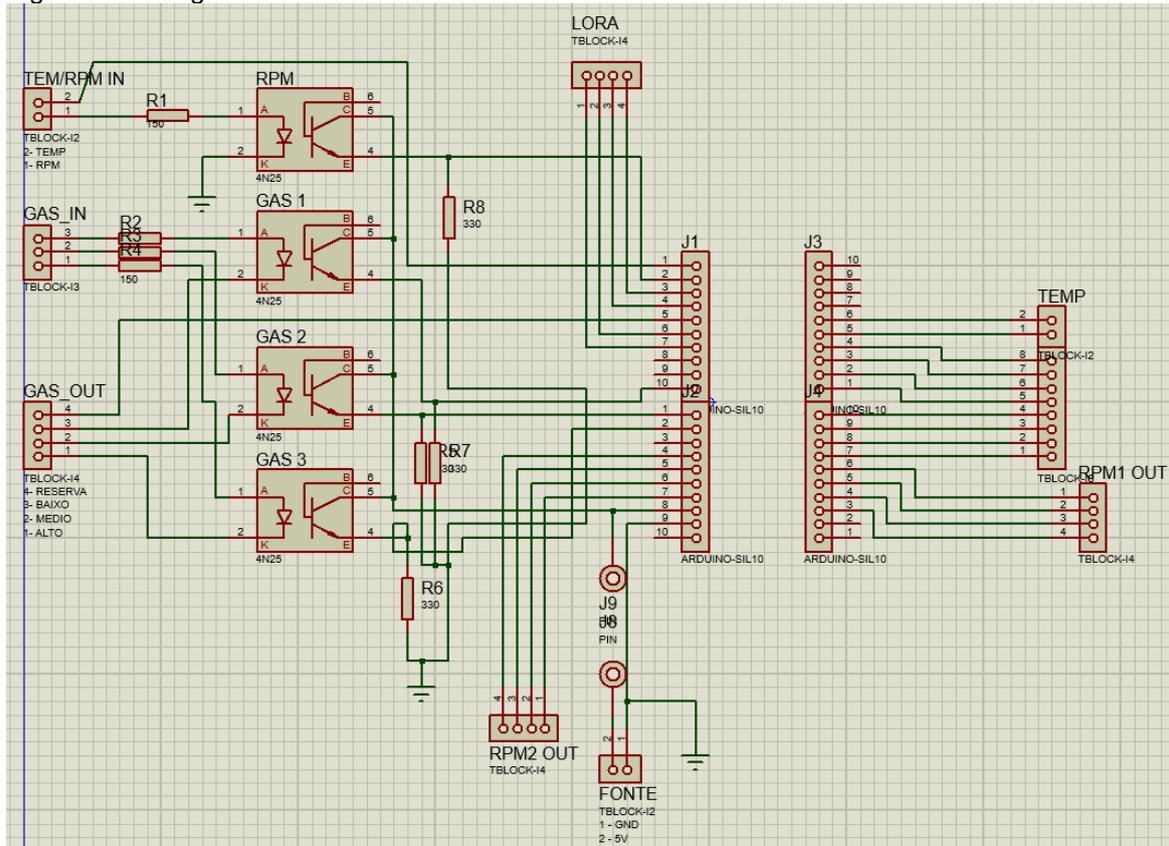
	<b>STM32F103</b>	<b>ARDUINO UNO/MINI</b>	<b>Arduino Mega</b>
<b>Velocidade CPU</b>	72MHz	16MHz	16MHz
<b>Memória Flash</b>	64KB	32KB	256KB
<b>ADC</b>	12-bit, 10 canais	10-bit, 8 canais	10-bit, 16 canais

Fonte: Autor, 2022

### 5.2.2 MICROCONTROLADOR

A modelagem esquemática iniciou-se pelo chip STM32F103 pois ele é o microcontrolador do circuito e comanda todos os demais componentes. Os pinos G (GND) e 3.3 (POWER) são responsáveis pela energia da placa já os pinos PB13, PA12, PA15, PB3 e PB4 são responsáveis pelos sensores digitais o pino PA12 é responsável pelo sensor analógico enquanto os pinos PB14, PB15, PA9 e PA10 são responsáveis por fazer a conexão com o LoRa, como mostrado na figura 16.

Figura 16 – Diagrama elétrico



Fonte: Autor, 2022.

### 5.2.3 LoRA

Partindo para o chip Lora ele possui 7 pinos que vão conectados a placa central cada pino é responsável por uma função como mostrado na figura 17.

Figura 27 – Portas Chip LoRa

Pino LoRa	Nome do Pino	Direção do Pino	Descrição
1	M0	Entrada	Utilizado para configurar modo de operação
2	M1	Entrada	Utilizado para configurar modo de operação
3	RXD	Entrada	Receptor UART de dados, conectado ao TX do microcontrolador
4	TXD	Saída	Transmissor UART de dados, conectado ao RX do microcontrolador
5	AUX	Saída	Indicação do estado do módulo
6	VCC	Entrada	Fonte de energia 3,3V ou 5V
7	GND	Entrada	Comum, ground

Fonte: Filipeflop, 2022.

O modelo utilizado o E32-TTL-100 é um modulo transiever com a tecnologia LoRa “spread spectrum”, com potencial de transmissão a longas distâncias o E32 traz

vantagens como estabilidade de potência e melhor performance no quesito anti-interferências.

As conexões M0 e M1 do LoRa E32 foram conectados diretamente ao negativo, para que o sistema opere em seu modo normal, o único pino não conectado foi o pino AUX por não haver necessidade.

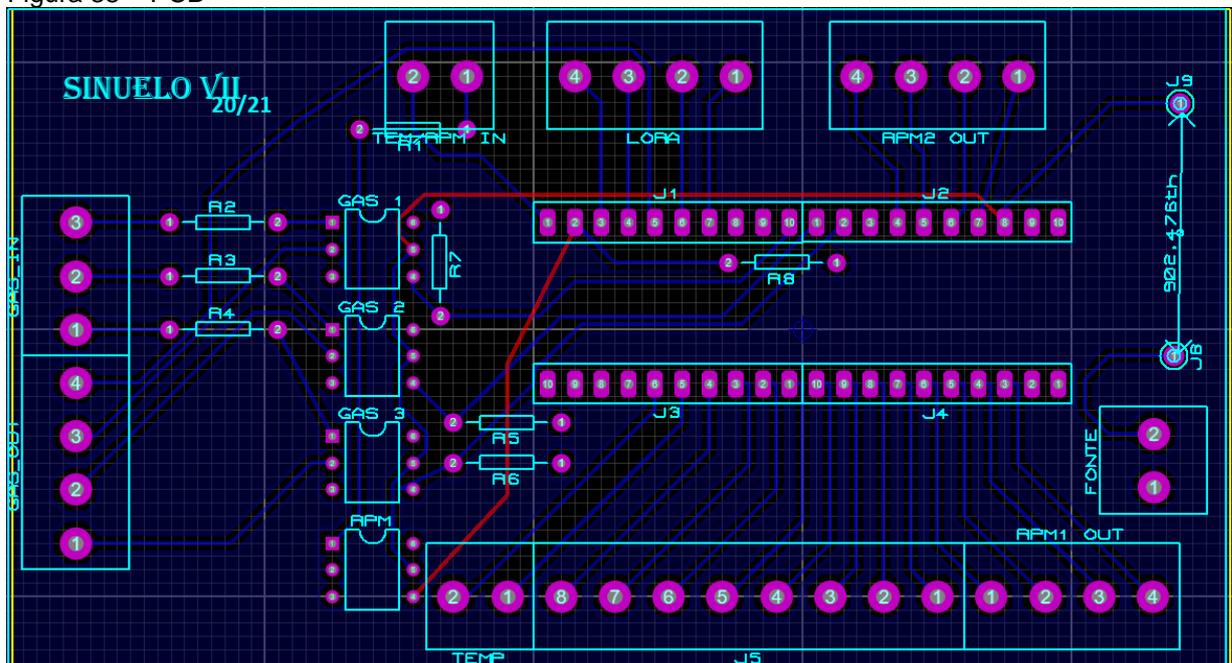
#### 5.2.4 SENSORES

São usados 3 diferentes tipos de sensores, sensores de combustível do tipo indutivo que usam portas digitais, um sensor para o rpm também porta digital e por último sensor de temperatura que utiliza uma porta analógica.

### 5.3 MODELAGEM DA PLACA DE CIRCUITO IMPRESSO

Partindo do esquemático e utilizando o mesmo programa foi feita a PCB como mostrado na figura 18, pensando em ser alimentado com 5 volts enquanto os sensores seriam alimentados de forma externa por uma bateria de 12 volts.

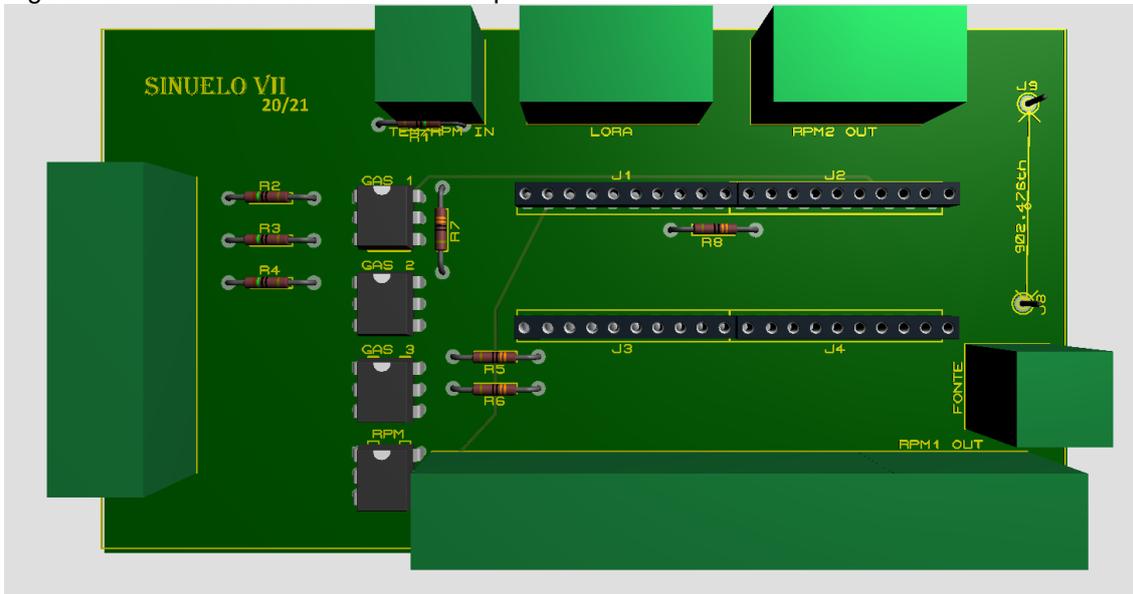
Figura 38 – PCB



Fonte: Autor, 2022.

O *Proteus* conta com uma ferramenta de modelagem 3D, mostrado na figura 20 a qual só funciona se os componentes utilizados tiverem os modelos na biblioteca, esse Modelo 3D é gerado a partir da PCB podendo assim ser utilizado em outros programas.

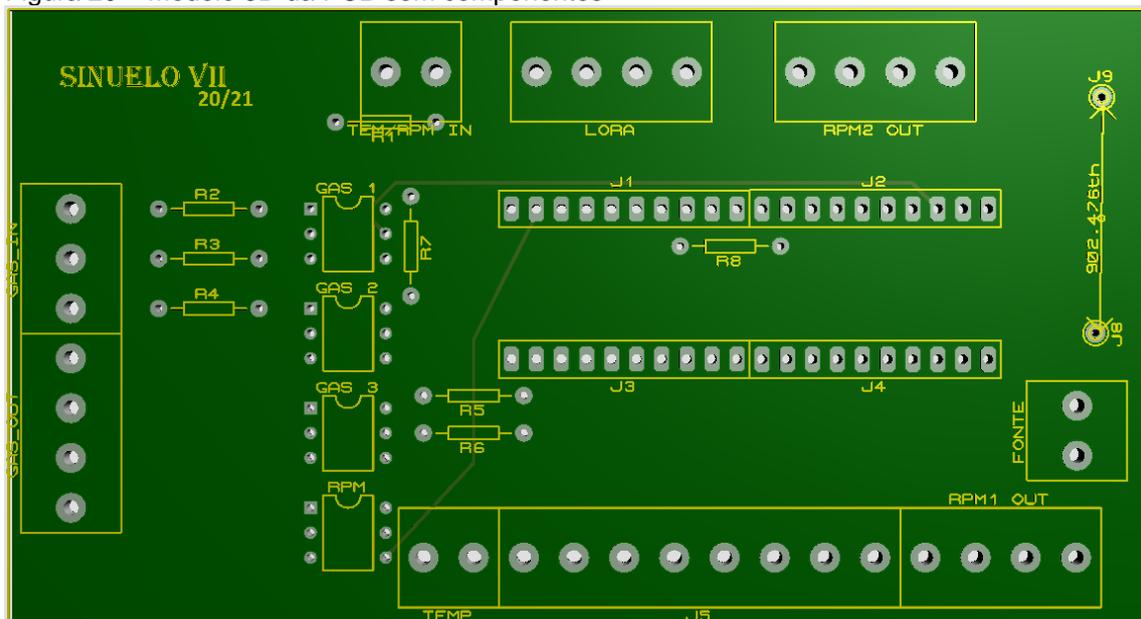
Figura 19 – Modelo 3D da PCB com componentes



Fonte: Autor, 2022.

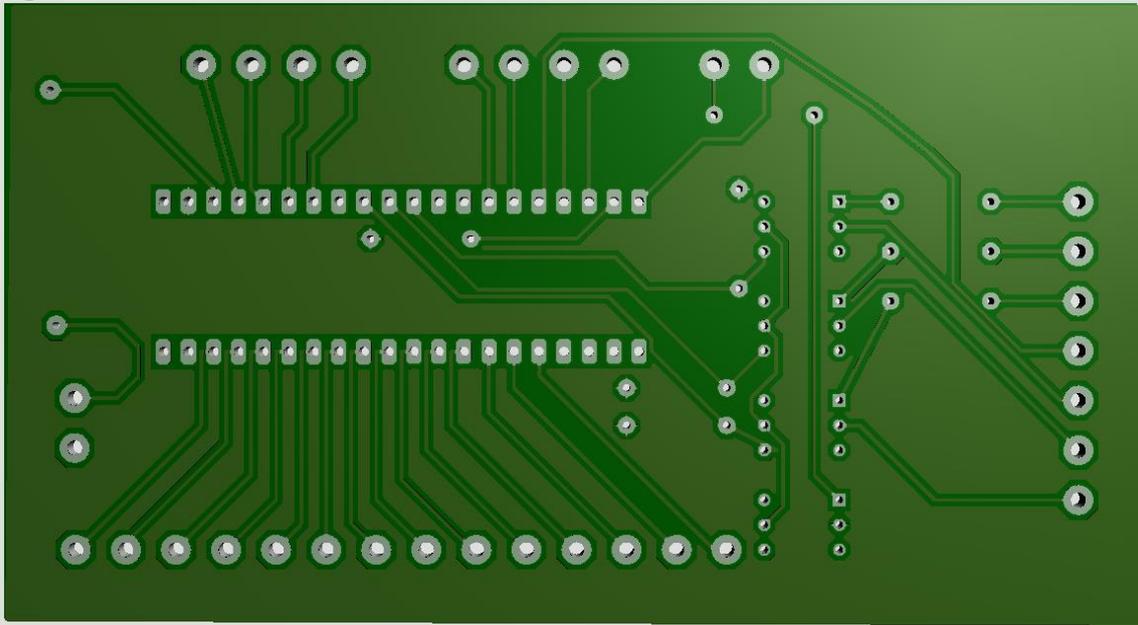
A ferramenta do modelo 3D conta com funções para diferentes tipos de visualização como demonstrado na figura 20 que apresenta o modelo sem os componentes instalados na placa, e a figura 21 que apresenta a parte de traz da placa com as trilhas.

Figura 20 – Modelo 3D da PCB sem componentes



Fonte: Autor, 2022.

Figura 21 – Modelo 3D da PCB lado de traz

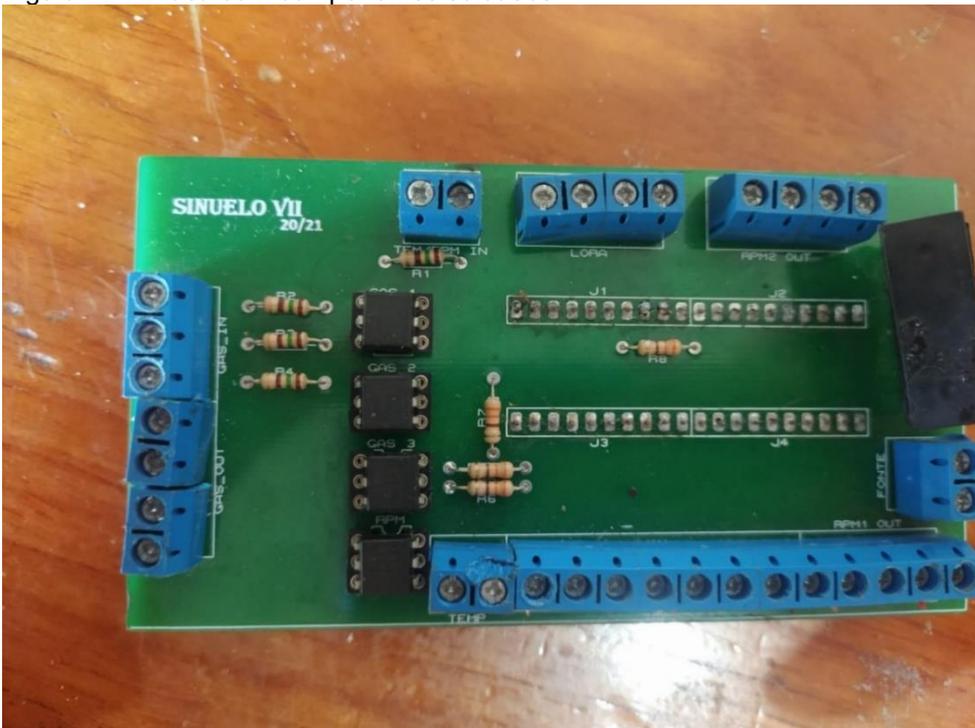


Fonte: Autor, 2022.

#### 5.4 MONTAGEM

A montagem do sistema se iniciou com a soldagem dos componentes na placa seguido de verificação de continuidade das trilhas utilizando um multímetro, também foi feita a montagem do receptor em protoboard como mostrado nas figuras 22 e 23.

Figura 22 – Placa com componentes soldados



Fonte: Autor, 2022.

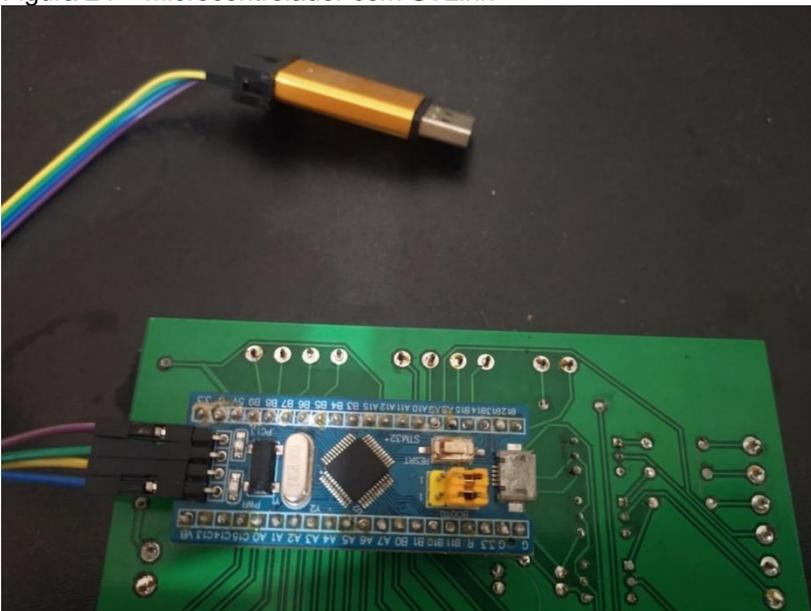
Figura 23 – Receptor



Fonte: Autor, 2022.

Após a verificação do hardware foi iniciado o desenvolvimento do código a ser utilizado no sistema tanto para o local em que seria adquirido os dados quanto para o local que iria receber esses dados. Após o código pronto foi utilizado o programador STLink, ou seja, componente responsável por passar o código do computador para o microcontrolador.

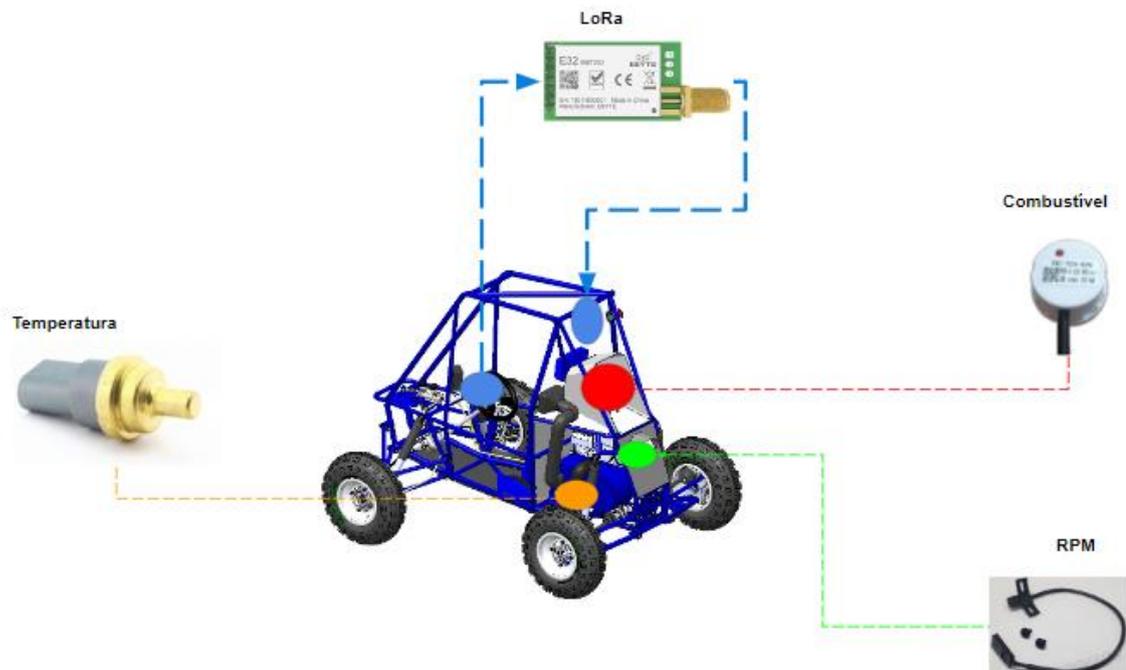
Figura 24 – Microcontrolador com STLink



Fonte: Autor, 2022.

Com essas etapas prontas foi necessário pensar em um local para ser instalado no veículo, o microcontrolador e o módulo LoRa foram instalados embaixo do capô do veículo com a antena para a transmissão sendo instalada em cima do veículo para ter um melhor sinal durante as corridas.

Figura 25 – Localização dos componentes no veículo



Fonte: Autor, 2022.

## 5.5 TESTES

Os testes foram feitos com o intuito de validar o sistema, eles foram realizados de dois modos diferentes, no primeiro foi feito em bancada para verificar o funcionamento utilizando fontes de bancada e geradores de frequência para simular o que se teria de sinais enviados pelo sensor do RPM do veículo.

A segunda parte foi feita no veículo para comprovação da funcionalidade e validação da implementação do sistema no veículo, como demonstrado na figura 26 onde as saídas de nível que apresentarem o valor 1 estão lendo um valor positivo para o combustível e se apresentarem o valor 0 então mostra que não tem combustível, o valor do RPM deve ser multiplicado por 100 e a temperatura está sendo apresentada em graus celsius.

Figura 26 – Saída dos dados adquiridos

```
nivel 1: 1
nivel 2: 1
nivel 3: 1
RPM: 45
Temperatura: 20C
nivel 1: 1
nivel 2: 1
nivel 3: 1
RPM: 47
Temperatura: 21C
nivel 1: 1
nivel 2: 1
nivel 3: 1
RPM: 46
Temperatura: 21C
nivel 1: 1
nivel 2: 1
nivel 3: 1
RPM: 45
```

Fonte: Autor, 2022.

## 6 CONCLUSÃO

O estudo apresentado buscou desenvolver uma solução de telemetria para a aquisição de dados em veículos de estilo *off Road* que tem como objetivo participar em competições.

Foi observado que o sistema se comporta de maneira satisfatória, com as finalidades do protótipo deste projeto funcionando com confiabilidade e estabilidade objetivadas.

Em relação a placa ela apresentou problemas de funcionamento inviabilizando o uso dela para o projeto, sem o uso da placa foi feita as ligações diretamente no microcontrolador possibilitando a aquisição, envio e gravação dos dados adquiridos durante o uso do veículo

Em conclusão, a maioria dos objetivos foram atingidos, sendo alguns com maior maestria que outros. Com esse estudo mostrou-se que a utilização de telemetria é possível e pode apresentar ganhos em competições futuras, contudo ainda há espaço para melhorias futuras.

## **SUGESTÃO DE MELHORIA E TRABALHOS FUTUROS**

Como sugestão de melhoria cita-se a criação de uma nova placa para agregação dos componentes pensando na utilização de um microcontrolador que já possua um meio de transmissão integrado.

Para trabalhos futuros sugere-se a implementação de um sistema de alimentação de energia elétrica renovável junto do sistema de alimentação interno, como por exemplo o uso de um mini painel solar, além de um sistema backup de armazenamento de dados diretamente no veículo para casos em que a comunicação for perdida.

## REFERÊNCIAS

INTRODUÇÃO aos Sistemas Embarcados (Embedded Systems). In: DOS REIS, Fábio. Introdução aos Sistemas Embarcados (Embedded Systems). [S. l.], 24 abr. 2015. Disponível em: <http://www.bosontreinamentos.com.br/eletronica/eletronica-geral/introducao-aos-sistemas-embarcados/>. Acesso em: 5 nov. 2021.

DA SILVA, Ismael Lopes. Introdução ao microcontrolador ARM Cortex M3. In: Introdução ao microcontrolador ARM Cortex M3. [S. l.]: Ismael Lopes da Silva, 23 jul. 2020. Disponível em: <https://www.embarcados.com.br/introducao-ao-microcontrolador-arm-cortex-m3/>. Acesso em: 26 out. 2021.

TEIXEIRA, Clever Marcos. Copyspider Anti-plágio: História. [S. l.], 2013. Disponível em: <http://www.copyspider.com.br>. Acesso em: 10 jul. 2017.

MICROCONTROLADOR: SAIBA O QUE É E COMO USAR ESTE DISPOSITIVO: MICROCONTROLADORES BASEADOS EM ARQUITETURA ARM. [S. l.], 9 abr. 2020. Disponível em: <https://blog.multcomercial.com.br/microcontrolador-saiba-o-que-e-e-como-usar/>. Acesso em: 12 nov. 2021.

GUIMARÃES, Alexandre de Almeida. Módulos Eletrônicos. In: ELETRÔNICA Embarcada Automotiva. [S. l.: s. n.], 2007.

CONTESINI, Leonardo. Afinal o que é ECU? Como elas funcionam?. [S. l.], 7 abr. 2017. Disponível em: <https://flatout.com.br/afinal-o-que-e-ecu-como-elas-funcionam/>. Acesso em: 12 nov. 2021.

NEUMANN, Fabiano Berlinck et al. PRINCÍPIOS DE COMUNICAÇÃO DE DADOS. [S. l.: s. n.], 2021.

NEWARK, LATeRe Representante. Conheça as principais Tecnologias de comunicação sem fio. [S. l.], 8 jul. 2021. Disponível em:

<https://www.embarcados.com.br/tecnologias-de-comunicacao-sem-fio/#LoRa>.  
Acesso em: 15 nov. 2021.

CANSSI, Anderson; SCHEUNEMANN, Bernardo; DOS SANTOS, Cristiano R. Comparação entre transmissores para sistema de telemetria do Baja Sinuelo FAHOR. Disponível em: <>. Acesso em 16 nov. 2021.

GOMES, Pedro César Tebaldi. TELEMETRIA: O QUE É E COMO FUNCIONA?. [S. l.], 5 dez. 2019. Disponível em: <https://www.opsservices.com.br/telemetria/>. Acesso em: 16 nov. 2021.

DE MORAIS, Zabelly Soares et al. INTRODUÇÃO A BIG DATA E INTERNET DAS COISAS (IOT). [S. l.: s. n.], 2018.

ARM DEVELOPER LOGO. Architectures: Instruction Sets. [S. l.], 2020. Disponível em: <https://developer.arm.com/architectures/instruction-sets>. Acesso em: 27 nov. 2021.

ZARTH, Mateus. PROGRAMA ELETRONICO PARA DESINE DE CIRCUITOS. In: ZARTH, Mateus. MINIFICAÇÃO EM INTERNET DAS COISAS PARA CAPTAÇÃO E ENVIO AUTÔNOMO DE DADOS EM HIDRÔMETROS ANALÓGICOS. 2022. Trabalho de final de curso (Graduação) - FAHOR, [S. l.], 2021. Disponível em: <https://fahor.com.br/publicacoes2/23-tfcs/tfcs-engenharia-de-controle-automacao/3063-tfcs-de-automacao-outros?ano=2022>. Acesso em: 31 out. 2022.

WHAT is Electronic Design Automation?. [S. l.], 2022. Disponível em: <https://www.synopsys.com/glossary/what-is-electronic-design-automation.html>. Acesso em: 31 out. 2022.

PRIMEIROS passos comunicação LoRa com Arduino. [S. l.], 25 maio 2018. Disponível em: <https://www.filipeflop.com/blog/primeiros-passos-lora-com-arduino/>. Acesso em: 13 ago. 2022.

DOS ANJOS, Eduardo G. P. A Evolução da Eletrônica Embarcada na Indústria Automobilística Brasileira. Disponível: <<https://maua.br/files/monografias/a-evolucao->

da-eletronica-embarcada-na-industria-automobilistica-brasileira.pdf> . Acesso em 29 jun 2021.

SANTOS, Bruno P.; SILVA, Lucas A. M.; CELES, Clayson S. F. S.; NETO, João B. B.; PERES, Bruna S.; VIEIRA, Marcos A. M.; VIEIRA, Luiz F. M.; GOUSSEVSKAIA, Olga N.; LOUREIRO, Antonio A. F. Internet das Coisas: da Teoria à Prática. Disponível em: <<https://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-dascoisas.pdf>>. Acesso em 21 ago. 2021

Equipe Sinuelo. Equipe Sinuelo Fahor: A Equipe. Disponível em: <<http://sinuelo.fahor.com.br/home>>. Acesso em 26 ago 2021.

VENTURA, Jocélio Barbosa; LÓPEZ, Leonardo Juan Ramírez; GOROSO, Daniel Gustavo. Aplicação de sistema multicanal na detecção de quedas para idosos. Disponível em: <<https://periodicos.ufrn.br/reb/article/download/3550/3377>>. Acesso em 25 ago. 2019.

LIMA, C.B FURTADO, A.O. Avaliação de software educativo para educação profissional: uma abordagem cognitiva. II ENINED – Encontro Nacional de Informática e Educação. São Paulo, 2011.

TEIXEIRA, G. B.; ALMEIDA, J. V. P. de. REDE LORA® E PROTOCOLO LORAWAN® APLICADOS NA AGRICULTURA DE PRECISÃO NO BRASIL [artigo científico]. Disponível em: <[http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/8436/1/PG\\_COELE\\_2017\\_2\\_01.pdf](http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/8436/1/PG_COELE_2017_2_01.pdf)>. Acesso em: 18 maio. 2021.

## ANEXOS

### CÓDIGO DO RECEPTOR

```
#include <SoftwareSerial.h>

SoftwareSerial loraSerial(Porta TX, Porta RX);

void setup() {
  Serial.begin(9600);
  loraSerial.begin(9600);
}

void loop() {
  if(loraSerial.available() > 1){
    String input = loraSerial.readString();
    Serial.println(input);
  }
  delay(60000);
}
```

### CÓDIGO DO EMISSOR

```
#include <Thermistor.h>
#include <SoftwareSerial.h>

#define GAS1 Porta 1
#define GAS2 Porta 2
#define GAS3 Porta 3
#define RPM Porta 4
int rpm;
unsigned long Time;
```

```
Thermistor temp(Porta ANALOGICA);

SoftwareSerial loraSerial(Porta TX, Porta RX);

String turnOn = "on";
String turnOff = "off";

void setup() {

    pinMode(Porta 1, INPUT);
    pinMode(Porta 2, INPUT);
    pinMode(Porta 3, INPUT);

    pinMode(Porta 4, INPUT);

    pinMode(Porta ANALOGICA, INPUT);

    Serial.begin(9600);
    loraSerial.begin(9600);
}

void loop() {
    int GAS1State = digitalRead(GAS1);
    int GAS2State = digitalRead(GAS2);
    int GAS3State = digitalRead(GAS3);
    loraSerial.print("nivel 1: ");
    loraSerial.println(GAS1State);
    loraSerial.print("nivel 2: ");
    loraSerial.println(GAS2State);
    loraSerial.print("nivel 3: ");
    loraSerial.println(GAS3State);
    if(digitalRead(RPM)==HIGH){
        rpm = rpm + 1;
        Time = millis();
    }
}
```

```
}while(Time < 60000);  
  
loraSerial.print("RPM: ");  
loraSerial.println(rpm);  
  
int temperatura = temp.getTemp();  
loraSerial.print("Temperatura: ");  
loraSerial.print(temperatura);  
loraSerial.println("C");  
  
delay(60000);  
}
```