



**UNIVERSIDADE FEDERAL DO VALE DO SÃO FRANCISCO
CURSO DE GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

Gustavo Marques de Souza Santos

**DESENVOLVIMENTO DE UM SISTEMA
EMBARCADO DISTRIBUÍDO PARA VEÍCULOS
OFF-ROAD**

Juazeiro - BA

2019

Gustavo Marques de Souza Santos

**DESENVOLVIMENTO DE UM SISTEMA
EMBARCADO DISTRIBUÍDO PARA VEÍCULOS
OFF-ROAD**

Trabalho apresentado a Universidade Federal do Vale do São Francisco - UNIVASF, Campus Juazeiro, como requisito da obtenção do título de Bacharel em Engenharia da Computação.

Orientador: Jadsonlee da Silva Sá

Juazeiro - BA

2019

Santos, Gustavo Marques de Souza
Desenvolvimento de um sistema embarcado distribuído para
S237d veículos off-road / Gustavo Marques de Souza Santos. - Juazeiro-BA,
2019.
xvii, 152 f. : il. ; 29 cm.

Trabalho de Conclusão de Curso (Graduação em Engenharia da
Computação) - Universidade Federal do Vale do São Francisco, Campus
Juazeiro-BA, 2019.

Orientador: Prof. Dr. Jadsonlee da Silva Sá.

1. Eletrônica automotiva. 2. Protocolo CAN. 3. Análise dos tempos
de respostas. I. Título. II. Sá, Jadsonlee da Silva. III. Universidade
Federal do Vale do São Francisco.

CDD 629.895

**UNIVERSIDADE FEDERAL DO VALE DO SÃO FRANCISCO
CURSO DE GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

FOLHA DE APROVAÇÃO

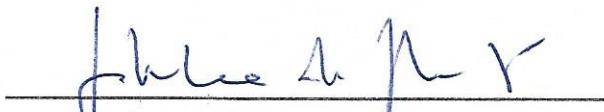
Gustavo Marques de Souza Santos

**DESENVOLVIMENTO DE UM SISTEMA
EMBARCADO DISTRIBUÍDO PARA VEÍCULOS
OFF-ROAD**

Trabalho apresentado a Universidade Federal do Vale do São Francisco - UNIVASF, Campus Juazeiro, como requisito da obtenção do título de Bacharel em Engenharia da Computação.

Aprovado em: 22 de Agosto de 2019

Banca Examinadora


Jadsonlee da Silva Sá, Doutor

UNIVASF

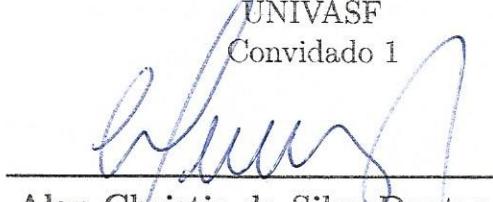
Orientador


Juracy Emanuel Magalhães da Franca,

Mestre

UNIVASF

Convidado 1


Alan Christie da Silva Dantas, Doutor

UNIVASF

Convidado 2

Dedico este trabalho aos meus pais.

Agradecimentos

Minha graduação, e consequentemente este trabalho, é o resultado do incentivo de muitas pessoas, fica aqui o meu muito obrigado!

Agradeço aos meus pais, Darlene Marques e José Marcos, pelo apoio incondicional, sempre acreditando e confiando em mim.

A minha irmã, Yasmin Marques, pela companhia nos vários anos morando juntos, pela confiança, pelo cuidado e por sempre acreditar que eu passaria nas matérias (mesmo eu não acreditando). E ao meu irmão, Kauê Victor, pelos muitos momentos de descontração e parceria.

Aos meus amigos e colegas de curso, pelos momentos alegres e desafiadores que passamos juntos. Em especial a Leonardo Cavalcante, Johnathan Alves, Esron Dtamar e Gabriel Gomes - pela amizade! e compreensão nos trabalhos em grupos.

Ao meu orientador, professor Jadsonlee, pela dedicação, paciência, disponibilidade e principalmente pelas oportunidades em muitos trabalhos extracurriculares. (Agradeço também pelos diversos chaveiros produzidos durante anos de pesquisa juntos).

Ao professor Jorge Cavalcanti, pela oportunidade de participar de projeto extra curricular com atividades desafiadoras.

Aos meus professores, pela fonte inesgotável de inspiração. Deixo o meu agradecimento ao Professor Brauliro Leal, por sempre me motivar.

Aos meus amigos da equipe Baajatinga pela confiança, ajuda, disponibilidade e compartilhar da alegria de ver a eletrônica funcionar. Em especial a Mateus Tamarindo, pelo projeto inicial e Enack Victor pela amizade, disponibilidade e ajuda imensurável.

“The child is grown, the dream is gone”
Trecho da música Comfortably numb, Pink Floyd

Resumo

A indústria automotiva tem exigido cada vez mais dos sistemas eletrônicos, buscando aumentar conforto, manutenibilidade e segurança nos veículos. Este trabalho apresenta um sistema embarcado distribuído para uso em veículos *off-road*. O sistema veicular desenvolvido consiste de sensores e módulos eletrônicos operando em tempo real e mantendo o motorista informado sobre a dinâmica do veículo. Para tanto, os módulos compartilham informações por meio de um barramento CAN. Com intenção de garantir o funcionamento do sistema embarcado, mesmo em situações críticas, é apresentada uma análise das restrições temporais e os tempos de respostas das mensagens enviadas pelos módulos eletrônicos. O veículo conta ainda com um módulo de telemetria capaz de transmitir a situação do veículo por meio de uma infraestrutura para uma unidade móvel. A unidade móvel facilita a manutenção preventiva do veículo. Após desenvolvido, o projeto foi testado em um veículo de competição do tipo Baja e os resultados são apresentados ao final do trabalho. Com base nos resultados apresentados, o sistema funciona adequadamente.

Palavras-chaves: eletrônica automotiva, protocolo CAN, análise dos tempos de respostas

Abstract

The automotive industry has been increasingly demanding of electronics systems, looking to improve comfort, maintenance and safety in cars. This project has an objective which is to project a distributed embedded system that can be used in off-road vehicles. The system consists of sensors and electronic modules, working in real time and keeping the driver aware about vehicle dynamic. For that, all electronic modules share information using CAN bus. To ensure a good behaviour of the embedded system, even under critical situation, an analysis of the time constraints and response times of messages sent by the electronic modules is presented. The vehicle also has a telematics module which is able to send all vehicle conditions using infrastructure to unity mobile. The unity mobile will improve vehicle preventive maintenance. In the end, the project was tested on a Baja type racing vehicle and the results are presented at the end of this paper, and based on the results presented, the system works properly.

Key-words: automotive electronic, CAN protocol, response time analysis

Lista de ilustrações

Figura 1 – Esquema do modelo OSI.	27
Figura 2 – Esquema das topologias básicas.	27
Figura 3 – Principais redes automotivas.	30
Figura 4 – Funcionamento do executivo cíclico em relação ao tempo.	34
Figura 5 – Modelo de rede CAN com barramento e nós conectados.	35
Figura 6 – Estados do barramento no decorrer do tempo.	36
Figura 7 – Formato de uma mensagem de dados do protocolo CAN.	36
Figura 8 – Formato do campo de arbitração.	37
Figura 9 – Formato do campo de controle.	37
Figura 10 – Formato do campo de checksum (CRC) e reconhecimento (ACK).	38
Figura 11 – Processo de arbitração de uma rede CAN.	40
Figura 12 – Diagrama de bloco de um nó conectado ao barramento CAN.	41
Figura 13 – Diagrama de blocos do sistema desenvolvido.	45
Figura 14 – Diagrama de blocos de um módulo eletrônico.	46
Figura 15 – Vista em perspectiva do projeto da placa de processamento e comunicação (PPC).	47
Figura 16 – Vista explodida do projeto do módulo eletrônico da estrutura.	48
Figura 17 – Esquema de instalação dos sensores capacitivos e os gráficos dos respectivos sinais.	49
Figura 18 – Vista explodida do módulo motor.	50
Figura 19 – Descrição dos ciclos de funcionamento do motor (ciclo Otto).	51
Figura 20 – Análise do sinal produzido pelo sistema de ignição.	51
Figura 21 – Vista explodida do projeto do módulo eletrônico de localização.	54
Figura 22 – Esquema de instalação do sensor indutivo na roda.	54
Figura 23 – Gráficos com sinal do sensor e da PCS.	55
Figura 24 – Vista explodida do projeto do painel de instrumentos.	57
Figura 25 – Vista frontal da modelagem do painel de instrumentos.	58
Figura 26 – Posição do painel em relação ao volante e motorista.	60
Figura 27 – Diagrama em blocos do módulo telemetria.	61
Figura 28 – Vista explodida do projeto do módulo da telemetria.	62
Figura 29 – Diagrama em blocos do subsistema telemetria.	63
Figura 30 – Modelagem da unidade móvel.	64
Figura 31 – Foto do automóvel utilizado para o teste da eletrônica.	66
Figura 32 – Foto de uma PPC com os seus componentes numerados.	67
Figura 33 – Foto das placas condicionadoras de sinais (PCS) e de algumas placas de processamento e comunicação(PPC).	68

Figura 34 – Montagem e instalação do módulo.	70
Figura 35 – Fluxograma simplificado do algoritmo do módulo.	71
Figura 36 – Fluxograma simplificado da tarefa <i>task battery</i>	72
Figura 37 – Fluxograma simplificado da tarefa <i>task CAN</i>	73
Figura 38 – Fluxograma simplificado da tarefa <i>task FUEL</i>	74
Figura 39 – Ativação das tarefas do módulo no decorrer do tempo.	75
Figura 40 – Tensão da bateria durante os testes do veículo.	75
Figura 41 – Situação do tanque de combustível durante os testes, com três momentos destacados.	76
Figura 42 – Montagem e instalação do módulo.	77
Figura 43 – Fluxograma simplificado da tarefa <i>task CAN</i>	78
Figura 44 – Fluxograma simplificado da tarefa <i>task Engine</i>	79
Figura 45 – Fluxograma simplificado da tarefa <i>task Temp</i>	80
Figura 46 – Fluxograma simplificado da tarefa <i>task EEPROM</i>	81
Figura 47 – Ativação das tarefas do módulo no decorrer do tempo.	82
Figura 48 – Análise do atraso das tarefas.	83
Figura 49 – Resultados do módulo do motor.	84
Figura 50 – Análise do horímetro em relação ao funcionamento do motor e do tempo.	85
Figura 51 – Análise do horímetro em relação aos contadores de minutos e horas.	86
Figura 52 – Montagem e instalação do módulo.	87
Figura 53 – Fluxograma simplificado da tarefa <i>task CAN</i>	88
Figura 54 – Fluxograma simplificado da tarefa <i>task speed</i>	89
Figura 55 – Sensor de nível crítico do fluído de freio.	90
Figura 56 – Fluxograma simplificado da tarefa <i>task brake</i>	91
Figura 57 – Ativação das tarefas do módulo no decorrer do tempo.	93
Figura 58 – Velocidade do automóvel durante os testes.	94
Figura 59 – Estado do reservatório (freios) durante testes.	95
Figura 60 – Fluxo de dados simplificado do painel de instrumentos.	95
Figura 61 – Fluxograma simplificado da tarefa <i>task speed</i>	96
Figura 62 – Fluxograma simplificado da tarefa <i>task fuel</i>	97
Figura 63 – Fluxograma simplificado da tarefa <i>task warnings</i>	98
Figura 64 – Fluxograma simplificado da tarefa <i>task rpm</i>	100
Figura 65 – Fluxograma simplificado da tarefa <i>task hourMeter</i>	101
Figura 66 – Fotografias do painel de instrumentos.	103
Figura 67 – Ativação das tarefas do módulo no decorrer do tempo.	104
Figura 68 – Fotografias do módulo da telemetria.	107
Figura 69 – Fluxograma simplificado da configuração da PPC.	108
Figura 70 – Fluxograma simplificado das interrupções do módulo.	109
Figura 71 – Diagrama simplificado dos estados do transmissor.	111

Lista de tabelas

Tabela 1 – Classificação das redes automotivas.	32
Tabela 2 – Configuração do sistema utilizado no exemplo	34
Tabela 3 – Tamanho permitido do barramento de acordo com a taxa de transmissão.	35
Tabela 4 – Relação entre tamanho da mensagem e a configuração do DLC.	38
Tabela 5 – Tarefas do módulo da estrutura.	49
Tabela 6 – Tarefas do módulo do motor.	53
Tabela 7 – Tarefas do módulo da localização.	56
Tabela 8 – Defeitos informados pelo painel.	59
Tabela 9 – Advertências informadas pelo painel.	59
Tabela 10 – Tarefas do módulo do painel.	59
Tabela 11 – Protocolo de comunicação.	63
Tabela 12 – Resumo da mensagem CAN do módulo.	70
Tabela 13 – Resumo da mensagem CAN do módulo.	77
Tabela 14 – Resumo da mensagem CAN do módulo.	87
Tabela 15 – Consumo elétrico no pior caso.	92
Tabela 16 – Funcionamento do conta-giros do painel.	102
Tabela 17 – Estrutura utilizada para salvar no cartão.	109
Tabela 18 – Relação entre ID e o módulo.	109
Tabela 19 – Primeiras mensagens salvas na memória do módulo.	112
Tabela 20 – Estrutura utilizada junto aos dados das mensagens.	113
Tabela 21 – Resumo da configuração da rede CAN por módulo.	119
Tabela 22 – Resumo do escalonamento das mensagens.	119
Tabela 23 – Relação entre rotação motora, led acesso e leds ativos.	143
Tabela 24 – Relação entre o número exibido e os bits ativos.	145
Tabela 25 – Relação entre os indicadores do lado esquerdo do painel e a porta utilizada para sua ativação.	146
Tabela 26 – Relação entre os indicadores do lado direito do painel e a porta utilizada para sua ativação.	146

Lista de abreviaturas e siglas

ACK	Acknowledge
ADC	Analog-to-Digital Converter
CAN	Controller Area Network
CRC	Cyclic Redundancy Checksum
CSMA/CA	Carrier-Sense Multiple Access/Collision Avoid
CSMA/CD	Carrier-Sense Multiple Access/Collision Detection
DHCP	Dynamic Host Configuration Protocol
DLC	Data Length Content
EOF	End of Frame
GPS	Global Positioning System
HMI	Human-machine interface
I2C	Inter-Integrated Circuit
ISO	International Standardization Organization
LIN	Local Interconnect Network
ME	Módulo da Estrutura
MM	Módulo do Motor
MP	Módulo do Painel
MT	Módulo de Telemetria
ML	Módulo de localização
MOST	Media Oriented Systems Transport
NRZ	Non Return to Zero
NTC	Negative Temperature Coefficient
OICA	International Organization of Motor Vehicle Manufacturers

OSI	Open Systems Interconnections
PCS	Placa para Condicionamento de Sinais
PPC	Placa para Processamento/Comunicação
PTC	Positive Temperature Coefficient
RATBSB	Regulamento Administrativo e Técnico Baja SAE Brasil
SAE	Sociedade de Engenheiros da Mobilidade
SOF	Start of Frame
SPI	Serial Peripheral Interface
TDMA	Time Division Multiple Access
UART	Universal Asynchronous Receiver-transmitter
UTC	Universal Time Coordinated

Sumário

Lista de ilustrações	9
Lista de tabelas	12
1 INTRODUÇÃO	18
1.1 Justificativa	18
1.2 Objetivos gerais	19
1.3 Objetivos específicos	19
1.4 Trabalho relacionado	20
1.5 Organização do trabalho	21
2 REFERENCIAL TEÓRICO	23
2.1 Eletrônica automotiva	23
2.1.1 Introdução	23
2.1.2 Domínios	23
2.1.3 Sensores	24
2.1.3.1 Sensor de nível/volume	24
2.1.3.2 Sensor de temperatura	25
2.1.3.3 Sensor de velocidade	25
2.1.3.4 Sensor de rotação	26
2.2 Conceitos e definições básicas de rede	26
2.2.1 Modelo de referência	26
2.2.2 Topologia	27
2.2.3 Método de endereçamento	28
2.2.4 Método de acesso ao meio	28
2.2.5 Controle de acesso ao meio	29
2.2.6 Redes automotivas	29
2.3 Sistema de tempo real	32
2.4 Rede de comunicação CAN	33
2.4.1 Camada física	34
2.4.2 Protocolo CAN	36
2.4.2.1 Mensagem de dados	36
2.4.2.2 Mecanismo de inserção de <i>bit</i>	39
2.4.2.3 Controle de acesso ao meio	39
2.4.2.4 Método de endereçamento	40
2.4.2.5 Detecção de erro	40

2.4.3	Nó CAN	41
2.5	Análise do tempo de resposta	42
3	SISTEMA IMPLEMENTADO	45
3.1	Arquitetura do sistema implementado	45
3.2	Sistema eletrônico veicular	46
3.2.1	Módulo estrutura	47
3.2.2	Módulo motor	50
3.2.3	Módulo localização	53
3.2.4	Módulo painel	56
3.2.5	Módulo telemetria	60
3.3	Subsistema Telemetria	62
3.4	Projeto e análise dos tempos de respostas do sistema	64
4	RESULTADOS OBTIDOS - MÓDULOS EMBARCADOS	66
4.1	Eletrônica veicular	66
4.1.1	Placa de processamento e comunicação e placa para condicionamento de sinal	67
4.1.2	Módulo da estrutura	69
4.1.2.1	Firmware	69
4.1.2.2	Resultados obtidos	71
4.1.2.3	Conclusão	74
4.1.3	Módulo do motor	74
4.1.3.1	Firmware	77
4.1.3.2	Resultados obtidos	80
4.1.3.3	Conclusão	84
4.1.4	Módulo de localização	85
4.1.4.1	Firmware	86
4.1.4.2	Dimensionamento elétrico	90
4.1.4.3	Resultados obtidos	90
4.1.4.4	Conclusão	93
4.1.5	Módulo do painel	93
4.1.5.1	Firmware	94
4.1.5.2	Resultados obtidos	102
4.1.5.3	Conclusão	104
5	RESULTADOS OBTIDOS - TELEMETRIA E ANÁLISE DOS TEMPOS DE RESPOSTA	106
5.1	Módulo de telemetria	106
5.1.1	Firmware	106
5.1.2	Telemetria	110

5.1.3	Resultados obtidos	112
5.1.4	Conclusão	114
5.2	Unidade móvel de telemetria	114
5.2.1	Conclusão	118
5.3	Análise da rede CAN	118
5.3.1	Análise do tempo de resposta	119
5.3.2	Resultados obtidos	120
6	CONCLUSÃO	122
6.1	Sugestões para trabalhos futuros	123
 REFERÊNCIAS		125
 ANEXO A – PLACA DE PROCESSAMENTO E COMUNICAÇÃO		129
 ANEXO B – MÓDULO DA ESTRUTURA		133
 ANEXO C – MÓDULO DO MOTOR		136
 ANEXO D – MÓDULO DE LOCALIZAÇÃO		139
 ANEXO E – MÓDULO DO PAINEL DE INSTRUMENTOS		142
 ANEXO F – MÓDULO DA TELEMETRIA		148

1 Introdução

A eletrônica automotiva tem assumido um papel importante na indústria automotiva. De acordo com Queisser (2006), os veículos atuais não são operacionais sem os sistemas eletrônicos. Além dos sistemas essenciais ao funcionamento, certas tecnologias só foram possíveis devido aos avanços da eletrônica, tais como, freios anti-blocantes, controle de velocidade automático, injeção eletrônica. Todo esse aparato tecnológico tem elevado a qualidade dos veículos.

O componente básico da eletrônica veicular é o módulo eletrônico, chamado também de ECU (*Electronic Control Unit*). O módulo eletrônico é responsável por realizar leituras de sensores, trocar informações com outros módulos, controlar atuadores, entre outras atividades. Diversos exemplos de veículos contendo centenas de dispositivos eletrônicos é dado por Navet e Simonot-Lion (2009).

O elevado número de módulos e de sensores passou a ser um problema em razão do custo, do peso e da manutenção do sistema. Com a intenção de compartilhar informações entre os diversos módulos e consequentemente diminuir o número de fios, adotou-se o uso de rede de comunicação. Embora um veículo conte com outras redes de comunicação, o protocolo CAN (*Controller Area Network*) foi projetada especificamente para a indústria automotiva e tem se estabelecido como protocolo de rede padrão segundo Bosch et al. (2007), Navet e Simonot-Lion (2009) e Emmelmann, Bochow e Kellum (2010).

Outra maneira de compartilhar informações é por meio da telemetria. Esse sistema é utilizado principalmente em competições. Durante uma competição os engenheiros podem acompanhar o desempenho mesmo com o veículo em movimento, como ocorre nos carros de fórmula 1.

Neste trabalho foi projetado e desenvolvido um sistema eletrônico automotivo para veículos do tipo fora de estrada (*off-road*). O sistema proposto consiste de uma eletrônica veicular usando a rede CAN e uma telemetria. O projeto da eletrônica veicular é uma remodelagem do projeto de pesquisa de Gonzaga (2018), projeto utilizado no veículo BJT05 da equipe Baajatinga.

1.1 Justificativa

A eletrônica veicular, sucessivamente, tem auxiliado os motoristas em tomadas de decisões mais eficientes. Uma vez que os sistemas eletrônicos são capazes de monitorar o veículo, e por meio de alertas informar ao motorista caso seja detectado alguma anomalia. Diversos aspectos de um veículo podem ser aprimorados por meio desses sistemas, tais

como: conforto, segurança e manutenibilidade.

No contexto de competição, as tomadas de decisões mais rápidas e eficientes traduzem-se em melhores resultados para a equipe, uma vez que reduz o tempo para detecção e resolução dos problemas. Além de tomadas de decisões mais rápidas, os sistemas eletrônicos auxiliam o motorista durante as provas, com informações que permitem acompanhar a dinâmica veicular e os limites seguros de operação.

Com intuito de melhorar os aspectos citados e, ao mesmo tempo, auxiliar a equipe Baajatinga, propomos e desenvolvemos um sistema veicular com informações disponibilizadas ao motorista por meio de painel digital atualizado em tempo real.

Por fim, com intuito de facilitar a manutenibilidade e acompanhando as tendências da indústria, um sistema de telemetria foi desenvolvido. Esse sistema é capaz de informar a situação do veículo aos responsáveis pela manutenção. As informações são passadas de maneira clara e de fácil acesso, sem necessidade de manusear nenhum fio ou ferramenta.

1.2 Objetivos gerais

Projetar, desenvolver e implantar uma eletrônica veicular capaz de mensurar as principais grandezas veiculares e informar ao motorista por meio de painel digital atualizado em tempo real.

Com intuito de garantir o funcionamento do sistema dado as restrições temporais, realizar um estudo da escalonabilidade do sistema e da rede CAN.

Projetar, desenvolver e implantar um sistema de aquisição de dados telemétrico, onde por meio de uma unidade móvel seja possível saber as condições do automóvel.

Por fim, adquirir e analisar os dados de um veículo em condições reais de uso, utilizando os sistemas desenvolvidos.

1.3 Objetivos específicos

- Desenvolver módulos eletrônicos específicos para cada domínio da eletrônica veicular, com capacidade de leitura de sensor e comunicação;
- Desenvolver um painel eletrônico de fácil visualização e com tacômetro, velocímetro, nível de combustível e luzes de advertência;
- Desenvolver uma unidade móvel com capacidade de comunicação com o veículo por meio de rede sem fio;

- Por meio do veículo de competição BJT06, verificar e analisar o funcionamento do sistema desenvolvido em situação de uso real;
- Verificar o cumprimento dos *deadlines* das tarefas projetadas levando em consideração o *firmware* desenvolvido;
- Verificar o cumprimento dos *deadlines* das mensagens, levando em consideração o pior tempo de resposta de uma mensagem na rede CAN.

1.4 Trabalho relacionado

O sistema proposto é uma remodelagem do projeto de pesquisa de Gonzaga (2018). O projeto de Gonzaga (2018) serviu como base, contudo, todo o projeto foi revisto e refatorado.

O projeto de pesquisa de Gonzaga (2018) era composto por quatro módulos, conectados por meio de barramento CAN e capaz de medir velocidade, rotação motora e a temperatura de óleo do motor.

Os principais pontos de melhorias foram:

- Módulo motor:
 - Novo método para aquisição da rotação com base no sinal da vela de ignição;
 - Desenvolvimento do horímetro;
 - Mudança no ponto de fixação com instalação mais próxima dos pontos de medições. O módulo foi acoplado ao motor, facilitando a retirada do motor em caso de manutenção;
 - Desenvolvimento do sistema para medir a temperatura do fluído de arrefecimento da transmissão.
- Módulo estrutura:
 - Desenvolvimento do sistema para medir o nível de combustível;
 - Mudança no ponto de fixação com redução do número de peças para fixação e consequentemente com redução do peso.
- Módulo localização:
 - Esse módulo é novo, além de medir velocidade conta com um porta fusível e centro de distribuição de energia;
 - Mudança no ponto de leitura da velocidade, seguindo as recomendações da literatura adotada;

- Mudança no ponto de fixação com redução do número de peças para fixação e consequentemente com redução do peso.
- Módulo Painel:
 - Desenvolvimento de um novo painel, mas mantendo um design semelhante ao projeto anterior. A posição e o design foram mantidos devido aos diversos *feedbacks* positivos;
 - Adoção de padrão ISO nas indicações e símbolos do painel.
- Telemetria:
 - Criação de uma unidade móvel com alimentação e display próprios;
 - Criação de *firmware* para os transmissores, de maneira a livrar os módulos da responsabilidade pela rede;
 - Criação de *firmware* para o módulo aproveitando as novas capacidades do transmissor;
 - Adoção de memória não volátil, no módulo, para armazenar o funcionamento do veículo, evitando a perda de dados caso a telemetria desconecte.

Além de adotar em todos os módulos proteção contra entrada de água (prensa-cabo), e adoção de um sistema de tempo real (RTOS) por meio de executivo cíclico. Por fim, o desenvolvimento do sistema de diagnóstico dos módulos.

1.5 Organização do trabalho

Este trabalho é composto por mais quatro Capítulos, sendo referencial teórico, o sistema implementado, os resultados obtidos e a conclusão. O anexo contém outros seis Capítulos, contendo os detalhes dos módulos a nível de *hardware*, descritos de forma bastante detalhada.

O Capítulo dois, referencial teórico, apresenta a teoria necessária para o desenvolvimento desse trabalho. Nesse Capítulo é apresentado a eletrônica automotiva e os seus domínios, os principais sensores e as definições de rede com foco nas redes automotivas, conceitos acerca dos sistemas de tempo real e os detalhes de funcionamento da rede CAN.

O Capítulo três, sistema implementado, apresenta o sistema de maneira gradual. Inicialmente é exposto uma visão macro, explicando à arquitetura e como os diversos sistemas interagem. Seguido com a explicação aprofundada da eletrônica embarcada, onde cada módulo é tratado separadamente. A telemetria é apresentada na sequência. O Capítulo é finalizado com uma análise do pior tempo de resposta do sistema proposto.

Nos Capítulos quatro e cinco são feitas análises dos resultados obtidos com os sistemas, ambos os Capítulos avaliam os dados obtidos de cada módulo proposto. O Capítulo quatro, resultados obtidos - módulos embarcados, apresenta os módulos embarcados no veículo. O Capítulo cinco, resultados obtidos - telemetria e análise dos tempos de resposta, contém os resultados do sistema de telemetria e por último uma análise detalhada da rede CAN.

O Capítulo seis, conclusão, encerra o desenvolvimento do trabalho com uma conclusão sobre o funcionamento do sistema completo, sobre todos os módulos, e traz algumas sugestões de trabalhos futuros.

Outros seis Capítulos contendo os detalhes elétricos dos respectivos módulos são apresentados nos anexos. Esses Capítulos foram organizando conforme descrito a seguir:

1. Anexo A - Apresenta os detalhes dos circuitos utilizados para condicionamento dos sinais e o diagrama elétrico da placa de processamento e comunicação (PPC).
2. Anexo B - Apresenta os condicionadores de sinais da placa de condicionamento de sinal (PCS) do módulo da estrutura. O anexo contém modelo do acoplamento, diagrama elétrico e descrição dos condicionadores.
3. Anexo C - Apresenta os condicionadores de sinais da PCS do módulo do motor, contendo modelo do acoplamento, diagrama elétrico e descrição dos condicionadores.
4. Anexo D - Apresenta os condicionadores de sinais da PCS do módulo de localização, contendo modelo do acoplamento, diagrama elétrico e descrição dos condicionadores.
5. Anexo E - Apresenta os condicionadores da PCS do módulo do Painel de instrumentos. O anexo apresenta os mesmos itens dos Anexo C e D.
6. Anexo F - Segue a mesma estrutura dos Anexos B, C, D e E. Apresenta os condicionadores de sinais do módulo da telemetria.

2 Referencial Teórico

2.1 Eletrônica automotiva

2.1.1 Introdução

A eletrônica automotiva normalmente é classificada por domínios, segundo Sangiovanni-Vincentelli e Natale (2007). Cada domínio apresenta requisitos próprios como: tempo de computação, escala de tempo, confiabilidade e flexibilidade. Além das funções por domínio, o setor automotivo tem sensores mais indicados para algumas atividades. As seções seguintes apresentam os detalhes dos domínios e dos sensores com foco na indústria automotiva.

2.1.2 Domínios

Os domínios da eletrônica veicular apresentados por Navet e Simonot-Lion (2009), são: *Powertrain*, chassis, corpo (*body*), interface homem máquina (HMI - *Human-machine interface*) e telemetria.

O domínio *powertrain* é formado pelos sistemas que são relacionados com a propulsão longitudinal do veículo, o que inclui motor e transmissão (NAVET; SIMONOT-LION, 2009). A rotação do motor é mensurada e enviada por um módulo desse domínio (*powertrain*) para o painel eletrônico que faz parte de outro domínio.

Segundo Navet e Simonot-Lion (2009) os sistemas que controlam a interação com a estrada, como é o caso das rodas e suspensão, formam o grupo do chassis ou domínio do chassis. Os sistemas do chassis devem garantir conforto e segurança ao motorista e aos passageiros. Em relação aos requisitos de tempo, Bosch et al. (2007) destaca a importância do tempo de resposta para garantir o comportamento em tempo real do sistema eletrônico do chassi.

Os sistemas responsáveis pelas funções do usuário e conforto são classificados como domínio do corpo (KESKIN, 2009). Por exemplo; controle de climatização (ar condicionado), portas, cintos e retrovisores. Esse domínio não apresenta requisitos críticos de tempo de resposta e de transferência.

Todas as interfaces de interação do motorista ou dos passageiros são classificados no domínio HMI. Exemplos de aplicações do domínio HMI são: painel eletrônico com velocidade do veículo, nível de óleo, controle do rádio. De acordo com Navet e Simonot-Lion

(2009), as interfaces desse domínio devem ser simples e intuitivas, e não deve distrair o motorista.

2.1.3 Sensores

Segundo Morris e Langari (2012), o sensor é o primeiro elemento de qualquer sistema de medição. Sensores são dispositivos que quando expostos a um fenômeno físico produzem um sinal proporcional (BISHOP, 2002). Embora existam muitas opções de sensores no mercado neste trabalho utiliza-se apenas o básico de um sistema automotivo.

Quando o sinal produzido pelo sensor é de um domínio diferente do mensurado trata-se como transdutor. Por exemplo, o sinal elétrico gerado devido a leitura da temperatura de um objeto ou ambiente. O domínio do sensor é o térmico, a saída é do domínio elétrico, logo houve uma conversão de energia. Os sinais gerados por transdutores podem ser analógicos ou digitais.

Sinais analógicos são acoplados à unidade de processamento (microcontrolador) por meio de conversores analógicos digitais (ADC - *Analogic Digital Converter*). Sensores cujos sinais são analógicos produzem um sinal contínuo proporcional ao fenômeno mensurado.

Sinais digitais podem ser acoplados diretamente a unidade de processamento, normalmente utilizam protocolo de comunicação. O sinal gerado proporcional ao fenômeno mensurado é discreto.

2.1.3.1 Sensor de nível/volume

Os sensores de nível ou volume são utilizados na indústria para mensurar a quantidade de conteúdo dentro de um tanque sem a necessidade de olhar o seu interior (BOYES, 2009), como é o caso do tanque de combustível ou do reservatório de fluído de freio.

Os sensores de nível podem ser divididos de acordo com a faixa de leitura em pontuais ou contínuos. Os sensores contínuos são capazes de detectar pequenas variações, por exemplo, sensor de pressão. Os sensores pontuais detectam a presença de líquido no ponto de medição, por exemplo, boia de nível e sensor capacitivo.

Os sensores de pressão usam o princípio físico que a pressão hidrostática de um líquido é diretamente proporcional a profundidade do líquido (MORRIS; LANGARI, 2012). A Equação 2.1 define o nível de um líquido de mesma densidade em um tanque aberto, sendo:

$$h = P/(\rho_*g) \quad (2.1)$$

onde,

$$\begin{aligned} h &: \text{Nível do líquido [m]} \\ P &: \text{Pressão [Pa]} \\ \rho &: \text{Densidade do líquido [kg/m}^3\text{]} \\ g &: \text{Aceleração da gravidade [m/s}^2\text{].} \end{aligned}$$

Os sensores capacitivos usam o princípio físico que a capacidade entre duas placas condutoras depende da constante dielétrica do material. Logo a presença ou ausência de líquido altera essa constante (BOYES, 2009). Uma vez que a distância também altera o valor da capacidade, esse tipo de sensor deve estar fixo e com a distância constante entre o ponto de medição e o sensor.

Outros sensores de nível/volume podem ser encontrados em Boyes (2009) ou Morris e Langari (2012).

2.1.3.2 Sensor de temperatura

De acordo com Queisser (2006), embora existam outros tipos, os termistores (NTC - *Negative Temperature Coeficient*) são amplamente adotados na indústria automotiva. Não existe um padrão automotivo desses sensores. Pode-se encontrar sensores tipo NTC com uma variedade grande de fios, conectores com ou sem cabo, curvas característica.

Os termistores são dispositivos que alteram a resistência interna de acordo com a temperatura. A relação entre temperatura e resistência pode ser positiva, caso do PTC, ou negativa, caso do NTC. Os termistores do tipo NTC apresentam coeficiente negativo de temperatura, ou seja, a resistência decresce conforme a temperatura aumenta. Em contrapartida o PTC (*Positive Temperature Coeficient*) tem o coeficiente positivo e a resistência cresce ao passo que a temperatura aumenta.

Outra maneira de mensurar a temperatura é por meio de dispositivos semicondutores. De acordo com Morris e Langari (2012), esses dispositivos possuem baixo custo e comparado ao termistor, apresenta melhor linearidade. A desvantagem desse tipo é a pequena faixa de leitura.

2.1.3.3 Sensor de velocidade

A velocidade de translação não pode ser mensurada diretamente, segundo Morris e Langari (2012). A velocidade de translação pode ser mensurada por meio de integração de um acelerômetro ou por meio de sensor de rotação.

2.1.3.4 Sensor de rotação

O dispositivo utilizado para medir rotação é chamado de tacômetro. Os tacômetros podem ser analógicos ou digitais. Os tacômetros digitais são mais precisos que os analógicos segundo Morris e Langari (2012).

Os tacômetros digitais utilizam diferentes tipos de sensores. De acordo com Queisser (2006) duas categorias principais de sensores são utilizados, sendo: ativo ou passivo. Os sensores ativos requerem fonte de alimentação para funcionar, por exemplo: sensores de efeito Hall, enquanto os sensores passivos não, por exemplo, sensores indutivos.

Os sensores indutivos são amplamente utilizados pela indústria automotiva (MORRIS; LANGARI, 2012).

2.2 Conceitos e definições básicas de rede

Com o aumento do número de sensores o sistema de comunicação automotivo se tornou um problema. Fatores como custo, peso e complexidade devido ao uso de fios e conectores motivaram a adoção de redes de comunicação (NAVET; SIMONOT-LION, 2009).

Ainda segundo Navet e Simonot-Lion (2009), em 1998 a BMW reduziu 15 kg de massa em um modelo quatro portas com adoção de redes locais (em inglês *Local Area Network*).

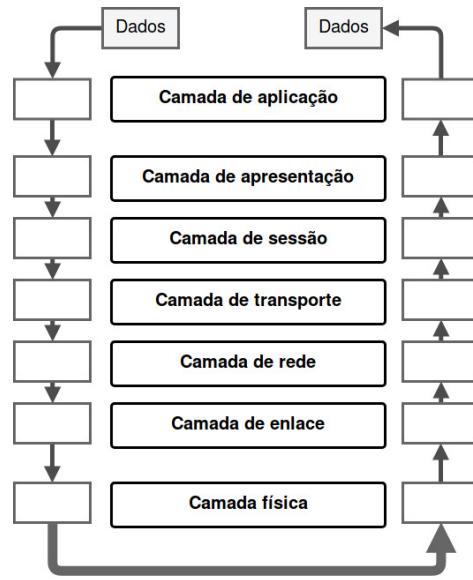
Devido a complexidade envolvendo a comunicação entre dispositivos, nesta e nas seções subsequentes serão apresentados os detalhes envolvendo a troca de mensagem, os mecanismos e os princípios.

2.2.1 Modelo de referência

De acordo com Morimoto (2008), o modelo OSI (*Open Systems Interconnections*) proposto pela ISO (*International Standardization Organization*) é um modelo teórico e serve para explicar diversos aspectos do funcionamento da rede. Além de explicar diversos aspectos, o modelo OSI permite a comparação entre os diferentes protocolos (BOSCH et al., 2007).

Nesse modelo um dado sistema de comunicação pode ser representado por camadas, de maneira que cada camada contém propriedades e tarefas bem definidas. O modelo OSI contém sete camadas, conforme representando na Figura 1. Dada uma camada, as propriedades da camada logo abaixo são disponibilizadas e podem ser acessadas por meio de interface bem definida.

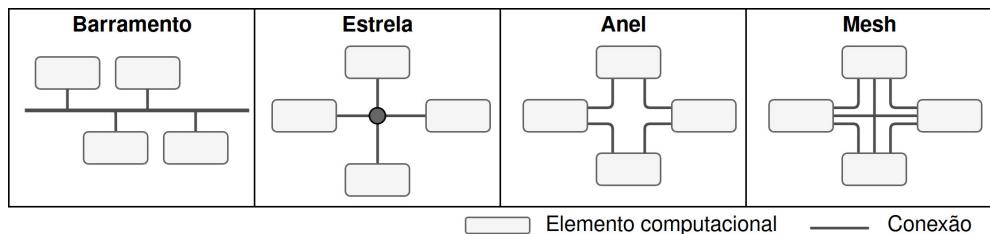
Embora esse modelo seja representado por sete camadas, algumas camadas do modelo OSI não são relevantes para sistemas de automação e controle, de acordo com Sá (2007).

Figura 1 – Esquema do modelo OSI.

Fonte: Baseado em Bosch et al. (2007).

2.2.2 Topologia

A maneira como os elementos computacionais são interligados influencia no funcionamento do sistema. De acordo com Bosch et al. (2007), as principais topologias de redes são baseadas nessas 4 (quatro) topologias básicas: Barramento, Estrela, Anel e Mesh. Conforme estão representadas na Figura 2.

Figura 2 – Esquema das topologias básicas.

Fonte: Baseado em Bosch et al. (2007).

Barramento

Na topologia barramento, todos os elementos computacionais estão conectados a um cabo principal, chamado de barramento. A falha de um nó (elemento computacional) não impede que os demais continuem se comunicando. Contudo a quebra do barramento desabilita a rede (BOSCH et al., 2007).

Estrela

Nessa topologia, todos os elementos computacionais estão conectados ao nó central. O nó central pode ser classificado em ativo ou passivo dependendo se esse nó atua ou não no sinal (MÁRIO, 2002).

Em caso de falha em um nó, a rede continua funcionando normalmente. Contudo, em caso de falha do nó central, o sistema para de funcionar.

Anel

A topologia anel é formada com todos os nós conectados em série e o último nó conectado ao primeiro, formando um círculo ou anel. De acordo com Bosch et al. (2007), essa topologia pode ser classificada em unidirecional ou bidirecional dependendo do número de anéis.

Ao ser utilizada em redes unidirecionais a falha de um nó compromete o funcionamento de toda rede, e mesmo em redes bidirecionais o isolamento de falha é difícil.

Mesh

Na topologia Mesh cada nó é conectado a um ou mais nós da rede (BOSCH et al., 2007). Dessa maneira, existem diversos caminhos para o mesmo destino.

Essa topologia apresenta alta resiliência, em caso de falha em um nó a mensagem pode ser entregue utilizando outro nó como intermediário.

2.2.3 Método de endereçamento

De acordo com Sá (2007), os protocolos podem ser diferenciados de acordo a orientação, podendo ser orientado ao nó ou orientado a mensagem. A principal diferença entre eles é a maneira com a qual uma mensagem é endereçada.

Em protocolos orientados a mensagem cada mensagem carrega consigo um identificador único que caracteriza a mensagem. O receptor escolhe as mensagens que irá ler.

Os protocolos orientados a transmissão cada mensagem carrega consigo o identificador do nó, ou grupo de nós, receptor. Nesse tipo de mensagem, o transmissor é quem decide quem irá ler a mensagem.

2.2.4 Método de acesso ao meio

De acordo com Navet e Simonot-Lion (2009), os principais paradigmas de comunicação em sistemas automotivos são: *Event-Triggered* e *Time-Triggered*.

Os sistemas de comunicação que acessam o meio de transmissão de forma aleatória seguem o paradigma *Event-Triggered*, de acordo com Sá (2007). Em sistemas *Event-*

Triggered as mensagens são enviadas assim que o evento ocorre (BOSCH et al., 2007).

Os sistemas que acessam o meio de transmissão de forma previsível seguem o paradigma *Time-Triggered* (SÁ, 2007). Em sistemas *Time-Triggered*, os nós tem permissão para transmitir durante um intervalo de tempo. Esse intervalo de tempo é definido durante o planejamento da rede (BOSCH et al., 2007).

2.2.5 Controle de acesso ao meio

Em sistemas *Event-Triggered*, onde os nós podem acessar o meio a qualquer momento, existe a possibilidade de colisões. Uma colisão acontece quando dois, ou mais, nós tentam acessar o meio de transmissão ao mesmo tempo. Os protocolos de comunicação devem definir políticas de acesso ao meio a fim de evitar as colisões (NAVET; SIMONOT-LION, 2009).

Uma política utilizada para controle consiste em analisar o canal de comunicação antes de iniciar a transmissão. O mecanismo que implementa essa política é chamado de CSMA (*Carrier-Sense Multiple Access*). Existem variações do CSMA de acordo com Kurose e Ross (2017).

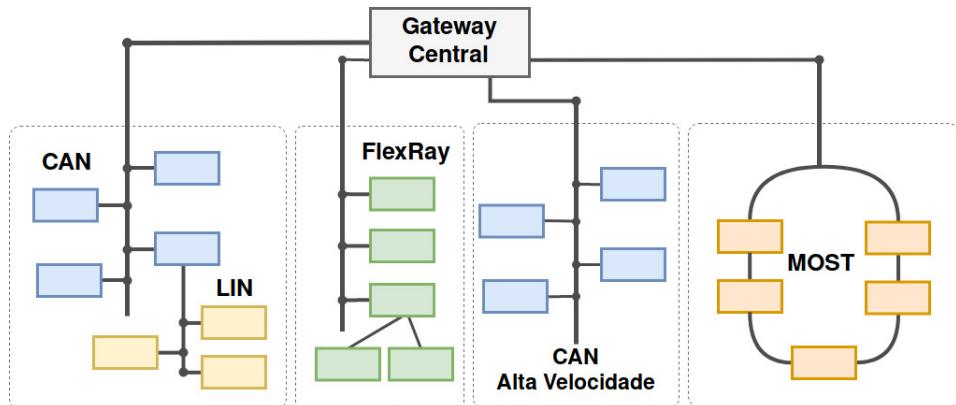
A Ethernet é um exemplo de protocolo de comunicação que utiliza a política CSMA com detecção de colisão (CSMA/CD - *Carrier-sense multiple access with collision detection*). Na rede Ethernet, sempre que uma colisão é detectada a transmissão é interrompida. Após um tempo aleatório, uma nova tentativa de transmissão é feita.

O protocolo CAN (*Controller Area Network*) é um exemplo que utiliza a política CSMA sem detecção de colisão (CSMA/CA - *Carrier-sense multiple access with collision avoidance*). No protocolo CAN, o nó deve esperar o fim da transmissão, caso o barramento esteja ocupado. Cada nó negocia o uso do barramento obedecendo prioridades.

2.2.6 Redes automotivas

Para atender aos diferentes requisitos, um veículo conta com diversos tipos de redes de comunicação, conforme representado na Figura 3. O modelo XC90 da Volvo é um exemplo que utiliza uma rede similar ao da Figura 3. De acordo com Navet e Simonot-Lion (2009), esse modelo da Volvo conta com barramento LIN, barramento MOST e barramentos CAN de alta e baixa velocidade.

Cada rede possui características próprias com aplicações indicadas. De acordo com Bosch et al. (2007), além do custo os critérios técnicos devem ser analisado para escolha da rede. Os principais critérios técnicos apresentado por Bosch et al. (2007) são; velocidade, imunidade a interferência eletromagnética, número de nós e capacidade de funcionar em tempo real.

Figura 3 – Principais redes automotivas.

Fonte: Baseado em Diarra (2013).

LIN

Local Interconnect Network (LIN) é um protocolo de comunicação serial de baixo custo, utilizado principalmente para comunicações que não exigem altas taxas de transferência (INSTRUMENTS, 2018). Esse protocolo foi desenvolvido por um consórcio formado por vários fabricantes de automóveis e tem ampla utilização nos veículos em controle de atuadores e leitura de sensores.

Em relação ao modelo OSI, a rede LIN utiliza apenas três camadas, sendo: camada física, camada de enlace e camada de aplicação. (DIARRA, 2013)

De acordo com o consórcio, a versão mais atualizada do protocolo LIN é a 2.2A. Esse protocolo está disponível por meio da ISO 17897 (*International Organization for Standardization*). Nessa rede, o acesso ao meio é feito por meio do princípio mestre-escravo de maneira que o barramento deve conter um mestre. A camada física da rede LIN é composta por apenas um fio com taxa de transmissão limitada até 20 kbit/s em razão da emissão eletromagnética (ZURAWSKI, 2014).

MOST

Media Oriented Systems Transport (MOST) é uma rede de alta velocidade para uso multimídia. Essa rede foi proposta devido a necessidade de taxa de transferência cada vez mais elevada no interior do veículo. Sistemas que contam com vídeo e som em alta definição e sistema de som com conectividade com celulares são exemplos de aplicações automotivas que as redes automotivas não atendiam devido a baixa velocidade do barramento.

Existem três versões da rede MOST. A primeira versão, MOST25, com velocidade de até 25 Mbit/s. A segunda versão, MOST50, com velocidade até 50 Mbit/s e a última versão, MOST150, com velocidade até 150 Mbit/s (BOSCH et al., 2007).

A rede MOST básica utiliza topologia anel, com as mensagens sendo transmitidas de um nó para o próximo. Outra topologia utilizada é a estrela, com adição de uma central. Utilizar a topologia estrela, nessa rede, tem a vantagem de permitir adição/remoção de nós sem interferir sobre os demais, melhorando a confiabilidade (GRZEMBA, 2012).

FlexRay

A necessidade de redes mais seguras e confiáveis, que possam atender aos requisitos das aplicações *X-by-wire* (aplicações controladas eletronicamente, sem redundância mecânica), levou ao desenvolvimento do sistema de comunicação FlexRay. De acordo com Zurawski (2014), FlexRay é um sistema de comunicação de alto desempenho, determinístico e escalável para aplicações automotivas.

O protocolo FlexRay segue o paradigma *time-triggered*, e utiliza TDMA (*time division multiple access*) para controle do barramento. Opera com velocidade de 10Mbit/s na configuração com redundância, ou com 20Mbit/s sem redundância (BOSCH et al., 2007). Os nós podem ser conectados em diferentes topologias, aceitando: estrela, barramento ou mesmo topologias híbridas (SHAW; JACKMAN, 2008).

CAN

Controller Area Network (CAN) é um protocolo de comunicação utilizado principalmente na indústria automotiva. O protocolo CAN pode ser utilizado em sistemas de tempo real, por meio da ISO 11898-2, ou em redes de baixa velocidade por meio da ISO 11898-3. Esse protocolo foi desenvolvido pela Bosch em meados dos anos 80.

A rede CAN funciona sobre o paradigma *event-triggered* com a política de acesso ao meio sem detecção de colisão. As colisões são evitadas por meio do processo de arbitragem, onde cada mensagem carrega consigo um identificador que além de identificar o conteúdo, determina a prioridade da mensagem. A rede CAN pode atingir taxa de transmissão de até 1 Mbit/s, conta com diversos mecanismos de detecção de erros além de confinamento de falhas. Todas essas características dão suporte para o uso desse protocolo em sistemas de tempo real, no caso automotivo os domínios do *powertrain* e *chassis*.

Com intuito de organizar as redes automotivas, em 1994, a Sociedade de Engenheiros da Mobilidade (SAE) classificou os sistemas de comunicação automotivo conforme a Tabela 1.

Tabela 1 – Classificação das redes automotivas.

Classe A		
Velocidade	Baixa (até 10 Kbit/s)	
Aplicações	Sensores e atuadores (Por exemplo: Sensor de chuva)	
Representantes	LIN e TTP/A	
Classe B		
Velocidade	Baixa (10 Kbit/s até 125 Kbit/s)	
Aplicações	Compartilhar informações entre unidades eletrônicas	
Representantes	CAN <i>low speed</i> e J1850	
Classe C		
Velocidade	Alta (125 Kbit/s até 1 Mbit/s)	
Aplicações	Sistemas que exigem resposta em tempo real (Por exemplo: Motor e chassis)	
Representante	CAN <i>high speed</i>	
Classe D		
Velocidade	Alta (acima de 1 Mbit/s)	
Aplicações	Sistemas multimídia (Por exemplo: Câmera de ré)	
Representantes	MOST, TTP/C e FlexRay	

Fonte: Baseado em Bosch et al. (2007)

2.3 Sistema de tempo real

De acordo com Bosch et al. (2007), os domínios do *powertrain* e *chassis* são sistemas com demandas de tempo real. Sistemas que exigem demandas de tempo real são sistemas onde cada resposta correta depende além da exatidão lógica, do instante em que ocorre. Segundo Lee, Leung e Son (2007), a principal diferença entre tarefas de tempo real e as demais tarefas está no fato que tarefas de tempo real devem finalizar até o *deadline*, ou seja, possuem um tempo máximo permitido para executar.

Os sistemas de tempo real podem ser classificados de acordo com o estouro do *deadline*. Os sistemas em que essa perda, do *deadline*, não são catastróficas são classificados como *soft real-time*. Caso a perca de um *deadline* possa levar a uma falha catastrófica, são classificados como *hard real-time* (BOSCH et al., 2007).

Segundo Sá (2016), os sistemas de tempo real são modelados por um conjunto de tarefas e mensagens que usam recursos limitados e compartilhados. A principal diferença entre mensagem e tarefa está no recurso compartilhado. Nesta seção serão tratados os detalhes das tarefas em tempo real, os detalhes das trocas de mensagens serão tratadas nas seções seguintes.

Cada tarefa é composta por um sequência de código, ativada de acordo com a necessidade e que compartilha os recursos da unidade de processamento. As tarefas são

ditas periódicas caso suas ativações ocorram sempre com o mesmo intervalo de tempo, chamado período. Caso a tarefa não tenha um período, é tratada como aperiódica. Para tarefas que possuem um tempo mínimo entre as ativações, caso particular das tarefas aperiódicas, trata-se por tarefa esporádica. (SÁ, 2016).

Com intuito de limitar o escopo desse trabalho, trataremos apenas de tarefas periódicas. Cada tarefa periódica τ_i é formada por uma tupla $\langle C_i, D_i, T_i \rangle$, onde C_i é o tempo de computação, D_i o seu *deadline* e o T_i é o período.

Segundo Sá (2016), a unidade de processamento pode ser compartilhado por um conjunto de tarefas \mathcal{T} , sendo $\mathcal{T} = \tau_1, \tau_2, \tau_3 \dots \tau_i$, logo o processador deve decidir qual tarefa ativar e qual tarefa terá ativação postergada. Esse processo de selecionar a tarefa é chamado de escalonamento.

Uma das maneiras mais comum, segundo Lee, Leung e Son (2007), de escalar um conjunto \mathcal{T} de tarefas em uma unidade processamento com único processador consiste de uma linha do tempo dividida em espaços de tempo bem definido. Cada tarefa τ_i tem um ou mais espaço de tempo dentro do hiperperíodo. O ciclo é repetido a cada hiperperíodo, de maneira que esse modelo também é chamado de executivo cíclico.

A escolha do hiperperíodo e do ciclo menor pode ser feita através do mínimo múltiplo comum (mmc) e do máximo divisor comum (mdc), respectivamente. A verificação se o espaço de tempo e o hiperperíodo é realizável é feito por meio da Equação 2.2.

$$\forall j = 1 \dots N, \sum_{i=1}^N h(i, j) C_i < T_{min} \quad (2.2)$$

Onde:

T_{min} : Valor do ciclo menor

C_i : Tempo de computação da tarefa τ_i

E a função h é binária sendo definida de acordo com a Equação 2.3.

$$h(i, j) = \begin{cases} 1 & \text{se a tarefa } \tau_i \text{ está no espaço de tempo } j \\ 0 & \text{se a tarefa } \tau_i \text{ não está no espaço de tempo } j \end{cases} \quad (2.3)$$

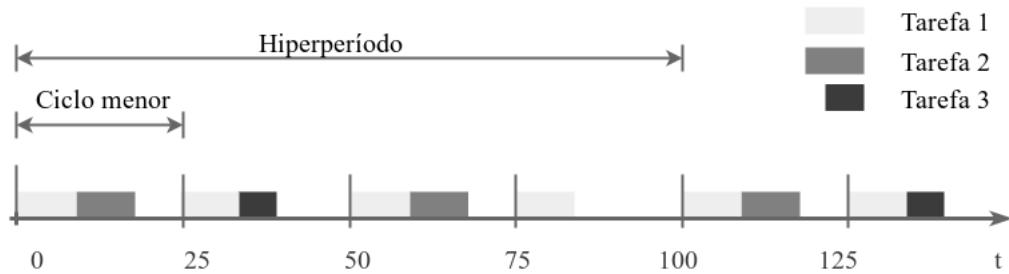
Considere um conjunto de tarefas $\mathcal{T} = \tau_1, \tau_2, \tau_3$ descrito na Tabela 2. Utilizando o modelo de divisão no tempo o hiperperíodo será então de 100 ms, com o menor período de 25 ms. O funcionamento desse modelo está representando na Figura 4.

2.4 Rede de comunicação CAN

A rede de comunicação CAN (*Controller Area Network*) foi desenvolvido durante a década de 80 pela empresa Bosch. Embora seja utilizada em outras indústrias, a rede CAN se

Tabela 2 – Configuração do sistema utilizado no exemplo

Tarefa τ_i	C (ms)	T (ms)
1	10	25
2	10	50
3	5	100

Figura 4 – Funcionamento do executivo cíclico em relação ao tempo.

Fonte: Baseado em Lee, Leung e Son (2007).

tornou padrão na indústria automotiva, segundo Bosch et al. (2007), Navet e Simonot-Lion (2009) e Emmelmann, Bochow e Kellum (2010).

A rede CAN está definida na ISO (*International Standardization Organization*) 11898, sendo sub dividida em três partes. A ISO 11898-1 contendo as definições de parte da camada física e a camada de enlace de dados. A ISO 11898-2 contendo as definições para redes de alta velocidade e a ISO 11898-3 para redes de baixa velocidade (SÁ, 2016). Conforme Natale et al. (2012), o padrão de camada física mais adotado é o definido pela ISO 11898-2.

Segundo Sá (2016), a camada de aplicação pode, opcionalmente, ser especificada e cita como exemplo o CAL (*CAN Application Layer*)/CANopen.

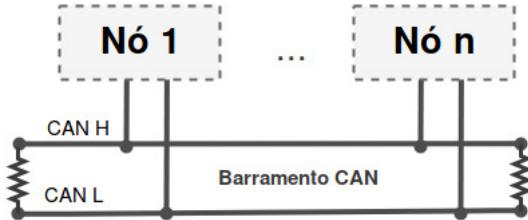
2.4.1 Camada física

O meio de transmissão pode variar de acordo com a norma adotada. A rede CAN de alta velocidade, definida pela ISO 11898-2, o meio de transmissão consiste de um barramento, tratado como barramento CAN, composto por um par traçado de fios tratados como CAN L (*CAN Low*) e CAN H (*CAN High*), conforme representado na Figura 5.

Na rede CAN de baixa velocidade pode-se utilizar apenas um fio para o barramento, desde que todos os nós utilizem a mesma referência de tensão (GND). De acordo com Bosch et al. (2007), esse tipo de rede é mais vulnerável a interferência eletromagnética.

Embora outras topologias sejam suportadas, a topologia barramento é a mais

Figura 5 – Modelo de rede CAN com barramento e nós conectados.



Fonte: Baseado em Sá (2007).

utilizada (SÁ, 2007). Em redes de alta velocidade o barramento deve conter resistências de 120Ω em ambas as pontas do barramento, conforme representando na Figura 5. Essas resistências evitam a leitura de mensagens falsas devido a reflexão do sinal (NATALE et al., 2012).

A taxa de transmissão permitida de uma rede CAN depende do tamanho do barramento uma vez que o tempo para uma mensagem atingir o último nó varia. Conforme apresentado na Tabela 3, para redes de 1 Mbits/s o tamanho máximo permitido é de 40 metros (NAVET; SIMONOT-LION, 2009). A Tabela 3 contém ainda o comprimento máximo do barramento para os valores típicos de taxa de transmissão.

Tabela 3 – Tamanho permitido do barramento de acordo com a taxa de transmissão.

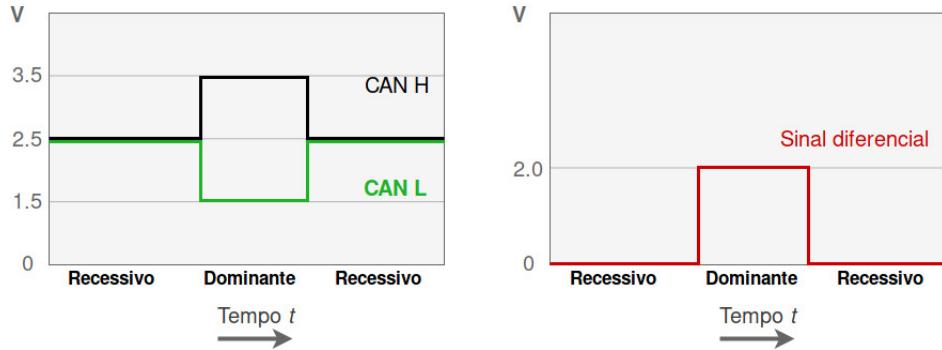
Taxa de transmissão (Kbits/s)	Tamanho do barramento (m)
1000	40
500	110
250	280
125	620
100	790

Fonte: Baseado em Sá (2007)

Na rede CAN dois estados são utilizados para representar os *bits* transmitidos, sendo: recessivo e dominante. Para redes definidas pela ISO 11898-2 os estados são representados por um sinal diferencial, conforme representando na Figura 6. Esse sinal é resultado da diferença de potencial entre as linhas de transmissão, CAN High e a CAN Low.

O estado recessivo é composto por ambas as linhas, CAN L e CAN H, com tensão igual a 2.5 V. Enquanto o estado dominante, a linha CAN H tem tensão igual a 3.5 V e a linha CAN L igual a 1.5 V (BOSCH et al., 2007).

A transmissão da mensagem é feita sobre o barramento usando codificação *Non Return to Zero* (NRZ). Com essa codificação o sinal permanece o mesmo sem necessidade

Figura 6 – Estados do barramento no decorrer do tempo.

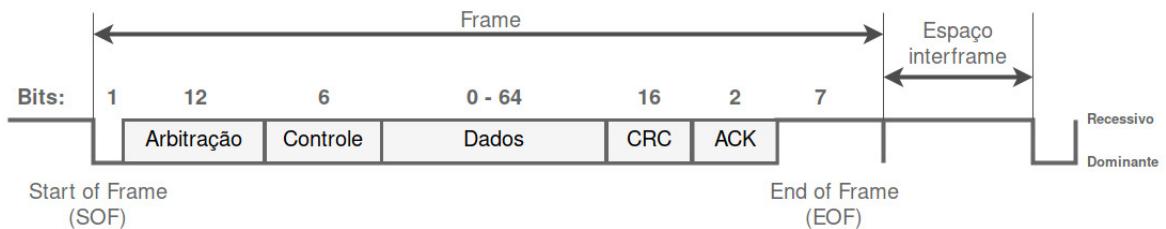
Fonte: Baseado em Bosch et al. (2007).

de retornar ao zero entre dois estados iguais (BOSCH et al., 2007).

2.4.2 Protocolo CAN

No protocolo CAN as mensagens são classificadas de acordo com sua função e conteúdo (NATALE et al., 2012). No total são 4 (quatro) tipos de mensagens diferentes, sendo: mensagem de dados, mensagem de requisição remota de dados, mensagem de erro, mensagem de sobrecarga.

A mensagem de dados é usada para transferir informações entre o nó transmissor e o(s) nó(s) receptor(es). A mensagem de erro é utilizada quando um nó deseja informar aos demais a detecção de uma anomalia, ou erro, na rede. Os outros tipos (requisição de dados e sobrecarga) não serão utilizados nesse trabalho.

Figura 7 – Formato de uma mensagem de dados do protocolo CAN.

Fonte: Baseado em Natale et al. (2012).

2.4.2.1 Mensagem de dados

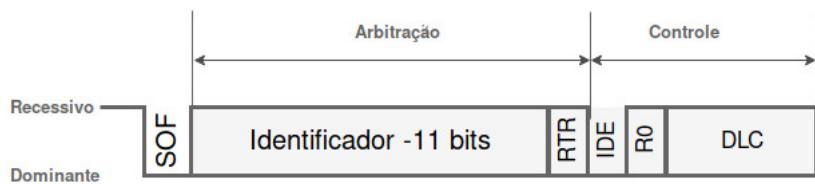
Conforme representado na Figura 7, a mensagem de dados é composta pelos seguintes campos: *Start of frame* (SOF), Campo de arbitragem, Campo de controle, Campo de dados,

campo de *Checksum* (CRC), campo de *Acknowledge* (ACK), *End of frame* (EOF) e o espaço *interframe* ou espaço entre *frames*.

O campo de *Start of frame*, ou SOF, tem duração de 1 *bit*, sempre no estado dominante, e inicia a transmissão. O campo SOF sempre é precedido por *bits* recessivos, em razão do espaço entre mensagens. Também é função do SOF sincronizar os nós na rede.

O campo de arbitragem pode ser de 12 *bits*, caso adotado identificadores no formato padrão, ou pode ser adotado o padrão estendido e esse campo será de 32 *bits*. De acordo com Sá (2007), o formato padrão é mais utilizado por apresentar um fator de utilização melhor.

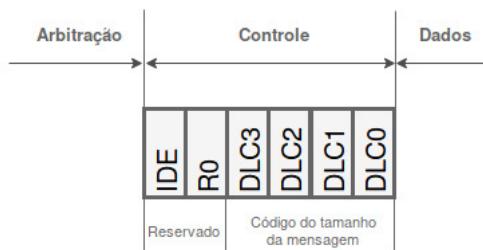
Figura 8 – Formato do campo de arbitragem.



Fonte: Baseado em Natale et al. (2012).

O campo de arbitragem, em seu formato padrão, consiste de 11 (onze) *bits* para o identificador e 1 (um) *bit* para distinção do tipo de mensagem (mensagem de dados ou requisição remota).

Figura 9 – Formato do campo de controle.



Fonte: Baseado em Natale et al. (2012).

O campo de controle, conforme representando na Figura 9, é composto por 6 (seis) *bits*, sendo reservado os 2 (dois) primeiros. De acordo com Natale et al. (2012), o *bit* IDE é dominante no formato padrão e recessivo no formato estendido. Os 4 (quatro) *bits* restante, *Data Length Content* (DLC), definem o tamanho da mensagem, a Tabela 4 apresenta todas as configurações possíveis dos *bits* DLC com seus respectivos tamanhos em bytes.

Tabela 4 – Relação entre tamanho da mensagem e a configuração do DLC.

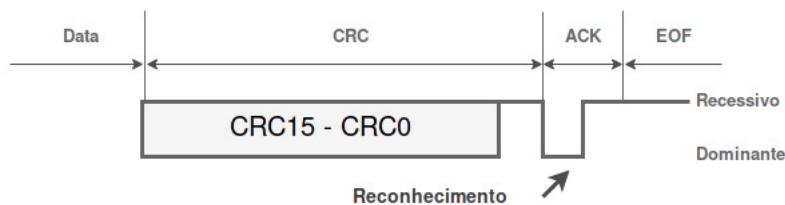
Tamanho (Bytes)	DLC3	DLC2	DLC1	DLC0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	X (0 ou 1)	x	x

Fonte: Baseado em Sá (2007).

O campo de *checksum*, ou campo CRC (*Cyclic Redundancy Checksum*), consiste de 15 *bits* contendo o código CRC, seguido de 1 *bit* recessivo. O código CRC possibilita a detecção de interferência na mensagem (BOSCH et al., 2007).

O campo de *Acknowledge* (ACK), ou campo de reconhecimento, consiste de dois *bits* e diferente dos outros campos esse não é enviado pelo transmissor (BOSCH et al., 2007). O campo de ACK sempre é deixado em recessivo pelo transmissor, sendo o primeiro *bit* sobreescrito pelos nós receptores após conferência do código CRC, confirmando o recebimento da mensagem.

Ambos os campos, *checksum* e reconhecimento, estão representados na Figura 10.

Figura 10 – Formato do campo de checksum (CRC) e reconhecimento (ACK).

Fonte: Baseado em Natale et al. (2012).

O campo EOF (*End of Frame*) consiste de 7 (sete) *bits* consecutivos de mesmo valor (recessivo). O espaço entre *frames* é utilizado para separar duas mensagens de dados, ou requisição de dados, seguidas. Esse espaço consiste de 3 *bits* consecutivos de valor recessivo.

Os campos SOF, arbitração, controle, dados e código CRC estão sujeitos ao mecanismo de inserção de *bits*. Os demais campos (delimitador CRC, ACK e EOF)

utilizam o formato definido.

2.4.2.2 Mecanismo de inserção de *bit*

Como consequência da codificação *Non Return to Zero* (NRZ) o barramento pode ficar sem nenhuma mudança de sinal durante um longo período. Uma vez que a mudança de estado é utilizado na sincronização da rede, longos períodos de mesmo sinal possibilitaria a perca de sincronização da rede. Para manter a rede sincronizada, o mecanismo de inserção de *bit* acrescenta *bits* durante a transmissão.

Esse mecanismo insere um *bit* de estado oposto a cada 5 *bits* com mesmo estado, garantindo assim uma troca de estado e consequentemente a sincronização da rede.

2.4.2.3 Controle de acesso ao meio

Durante o funcionamento da rede, quando um nó deseja utilizar o meio segue-se um protocolo bem definido. O acesso é controlado por meio do protocolo de arbitração. Esse protocolo impede a preempção do meio e define regras de acesso para evitar colisões.

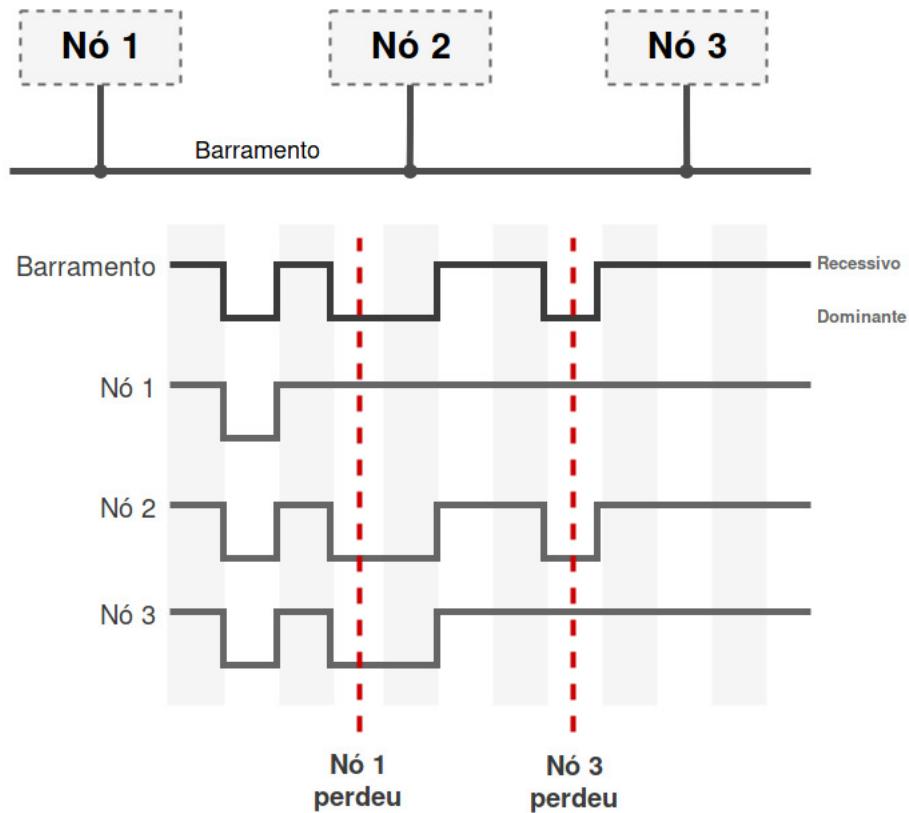
O processo de acesso ao meio inicia com a detecção do barramento livre. O nó que deseja enviar uma mensagem de dados inicia a transmissão por meio de um campo SOF, etapa onde todos os nós são sincronizados. Nesse ponto da transmissão cada nó que deseja enviar uma mensagem inicia a competição para ganhar acesso ao barramento (NATALE et al., 2012).

A disputa pelo barramento se dá através do envio do campo de arbitração. A rede CAN utiliza o método CSMA/CA (*Carrier-Sense Multiple Access with Collision Avoidance*) para controle de acesso, ou seja, cada nó envia o código de arbitração para o barramento e ao mesmo tempo analisa o estado atual. O envio do código de arbitração é feito *bit* a *bit*, caso ocorra uma colisão durante esse processo resolve-se aplicando o operador lógico *AND*. O nó perde a disputa se durante o processo enviar um *bit* recessivo, mas o barramento permanece em dominante. Ao final do processo de arbitração, o nó que enviar o código sem sofrer nenhuma mudança de seu sinal vence o processo e tem acesso ao meio, os demais nós ficam no modo escuta.

Devido ao fato de cada nó possuir um identificador único, o processo de arbitração é determinístico e baseado em prioridade (NATALE et al., 2012). O menor identificador sempre vence os demais, ou seja, tem alta prioridade.

Um exemplo de processo de arbitração está representado na Figura 11. De acordo com a Figura, o nó 1 no quarto *bit* envia um *bit* recessivo contudo o nó 2 envia um *bit* dominante, logo o nó 1 perde e entra em modo de escuta. Na sequência o nó três perde o processo, logo o único nó que não sofreu nenhuma interferência foi o nó 2. Ao final do processo o nó 2 poderá enviar a mensagem.

Figura 11 – Processo de arbitragão de uma rede CAN.



Fonte: Baseado em Bosch et al. (2007).

2.4.2.4 Método de endereçamento

A rede CAN é uma rede orientada a mensagem. Dessa maneira cada mensagem carrega consigo um identificador próprio que caracteriza o seu conteúdo (por exemplo: velocidade do veículo). Todos os nós recebem a mesma mensagem (*broadcast*), cada nó fica encarregado de selecionar quais são úteis, por meio de filtros e máscaras, e automaticamente descarta as demais.

De acordo com Bosch et al. (2007), esse método tem a vantagem dos nós operarem completamente independentes. Caso um nó precise de uma informação que já está disponível no barramento, ou um novo nó precise ser integrado ao sistema, tudo que precisa ser feito é atualizar a lista de recepção do novo nó.

2.4.2.5 Detecção de erro

Para garantir a entrega correta das mensagens a rede CAN implementa alguns mecanismos para detecção de erros, sendo: verificação de *Checksum*, verificação de formato, verificação de reconhecimento (ACK), verificação de *bit*, verificação da regra de inserção de *bits*.

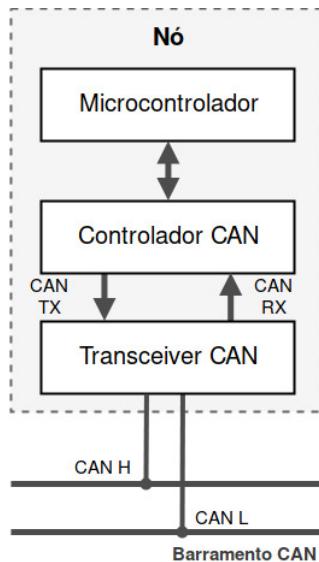
Qualquer nó da rede ao detectar uma anomalia no barramento transmite uma mensagem de erro. Essa mensagem consiste de 6 (seis) *bits* dominantes consecutivos. Uma mensagem de erro informa a todos os nós da rede a ocorrência de um erro. O transmissor interrompe o envio e entra na disputa pelo barramento para reenviar a mensagem. De acordo com Navet e Simonot-Lion (2009), o reenvio de uma mensagem pode levar a perca do *deadline* devido ao atraso.

A sucessiva ocorrência de erros, ou mesmo falhas de *hardware*, poderiam levar o bloqueio da comunicação. Dessa forma, o protocolo CAN implementa mecanismos de confinamento de erros. Esse mecanismo protege a rede contra interrupções frequentes. Os detalhes do funcionamento podem ser encontrados em: Sá (2007) e Natale et al. (2012).

2.4.3 Nô CAN

O componente básico para conectar um módulo eletrônico ao barramento CAN é o dispositivo de rede, composto por um controlador e um *transceiver*. A unidade de processamento (microcontrolador) é conectado ao barramento CAN por meio deste dispositivo. Um nó CAN básico é formado então por um microcontrolador, um controlador e um *transceiver*, conforme representando na Figura 12.

Figura 12 – Diagrama de bloco de um nó conectado ao barramento CAN.



Fonte: Baseado em Bosch et al. (2007).

De acordo com Sá (2007) o controlador CAN pode ser classificado de acordo com a integração dos componentes de um nó (unidade de processamento e *transceiver*). O controlador pode ser *stand-alone*, quando é completamente separado dos demais componentes,

pode ser integrado, quando é integrado ao microcontrolador mas separado do *transceiver* e pode ser *single-chip*, quando todos os componentes estão integrados em um único *chip*.

O controlador mais comum é o integrado ao microcontrolador de acordo com Natale et al. (2012) e segundo Bosch et al. (2007) essa arquitetura (microcontrolador com controlador CAN) tem melhor custo benefício.

O *transceiver* é responsável por converter o sinal da unidade de processamento (microcontrolador), normalmente 5 V, no sinal definido pela norma CAN adotada.

2.5 Análise do tempo de resposta

O tempo de resposta de uma instância k de uma mensagem m é dado pela Equação 2.4.

$$R_m(k) = J_m + W_m(k) - kT_m + C_m \quad (2.4)$$

Onde:

$J_m(k)$: Tempo de resposta no pior caso;

$W_m(k)$: Tempo de espera no buffer;

$C_m(k)$: Tempo de transmissão no pior caso.

Cada mensagem consiste de uma tupla formada por $\langle P_m, T_m, J_m, C_m, D_m \rangle$ sendo P_m a prioridade da mensagem, T_m o período, J_m o atraso de resposta (*Jitter*), C_m o tempo de transmissão no pior caso e D_m o requisito de tempo (*deadline*).

É de conhecimento do projetista a prioridade de cada mensagem P_m , o período T_m e o requisito de tempo D_m . O atraso de resposta J_m pode ser mensurado.

O tempo de transmissão C_m consiste do número de *bits* vezes o tempo de transmissão de cada *bit*. O tempo de *bit* (τ_{bit}) é definido de acordo com a taxa de transmissão escolhida.

O número de *bits* de um pacote CAN, conforme detalhado na seção 2.4.2, consiste de v *bits* de controle (34 *bits* no formato padrão), um pacote de dados b_m entre [0,8] *bytes* e outros 13 *bits*. O tamanho de um pacote CAN pode então ser definido de acordo com a Equação 2.5.

$$c_m = v + 13 + 8 * b_m \quad (2.5)$$

Os *bits* de controle v e de dados estão expostos ao mecanismo de inserção de *bits*. O número de *bits* inserido no pior caso é dado pela Equação 2.6. O operador $\left\lfloor \frac{a}{b} \right\rfloor$ retorna o maior inteiro menor ou igual à a/b .

$$ns_n = \left\lfloor \frac{n-1}{4} \right\rfloor \quad (2.6)$$

O número total de *bits*, pacotes mais os *bits* inseridos, é dado de acordo com a Equação 2.7.

$$c_m = v + 13 + 8 * b_m + \left\lfloor \frac{v + 8 * b_m - 1}{4} \right\rfloor \quad (2.7)$$

O pacote no formato padrão tem 34 *bits* de controle, logo o número total de *bits* no pior caso de inserção é dado de acordo com a Equação 2.8.

$$c_m = 47 + 8 * b_m + \left\lfloor \frac{34 + 8 * b_m - 1}{4} \right\rfloor \quad (2.8)$$

Por fim, o tempo de transmissão C_m é dado de acordo com a Equação 2.9.

$$\begin{aligned} C_m &= cm * \tau_{bit}; \\ C_m &= (47 + 8 * b_m + \left\lfloor \frac{34 + 8 * b_m - 1}{4} \right\rfloor) * \tau_{bit} \end{aligned} \quad (2.9)$$

O atraso de enfileiramento W_m é determinado pela Equação 2.10.

$$W_m = B_m + I_m \quad (2.10)$$

Sendo:

$$\begin{aligned} B_m &: \text{Atraso de bloqueio;} \\ I_m &: \text{Atraso de interferência.} \end{aligned}$$

O atraso de bloqueio B_m consiste do tempo em que a mensagem m espera devido ao barramento estar ocupado com outra mensagem de prioridade inferior. O pior caso de bloqueio é determinado de acordo com a Equação 2.11, em que lp é o conjunto de mensagens com prioridade inferior ao da mensagem m .

$$B_m = \max_{k \in lp(m)}(C_k) \quad (2.11)$$

O atraso por interferência I_m consiste do tempo que a mensagem espera devido ao uso do barramento por mensagens com maior prioridade. Para determinar o atraso por interferência primeiro devemos analisar o período ocupado de cada mensagem.

O maior período ocupado com nível de prioridade Pm é determinado pela Equação 2.12. O operador $\lceil \frac{a}{b} \rceil$ retorna o menor inteiro maior ou igual à a/b .

$$t_m^{n+1} = B_m + \sum_{\forall k \in h_{ep}(m)} \left\lceil \frac{t_m^n + J_k}{T_k} \right\rceil * C_k \quad (2.12)$$

onde:

$h_{ep}(m)$ é o conjunto de mensagens com prioridade maior ou igual à mensagem m

A relação recursiva da Equação 2.12 começa com $t_m^0 = C_m$ e termina quando $t_m^{n+1} = t_m^n$. Essa relação de recorrência irá convergir se o fator de utilização U_m for menor ou igual a um, para mensagens com prioridade maior ou igual m, conforme a Equação 2.13.

$$U_m = \sum_{\forall k \in h_{ep}(m)} \left(\frac{C_k}{T_k} \right) \leq 1 \quad (2.13)$$

O número total de instâncias disponíveis para transmissão dentro do período ocupado é dado pela Equação 2.14.

$$Q_m = \left\lceil \frac{t_m + J_m}{T_m} \right\rceil \quad (2.14)$$

O atraso de enfileiramento da instância q da mensagem m é dada de acordo com a Equação 2.15.

$$W_m^{n+1}(q) = B_m + qC_m + \sum_{\forall k \in hp(m)} \left\lceil \frac{W_m^n + J_k + \tau_{bit}}{T_k} \right\rceil * C_k \quad (2.15)$$

onde:

$hp(m)$ Conjunto de mensagens com prioridade maior que a mensagem m;

τ_{bit} Tempo de transferência do bit.

A relação de recorrência começa com $W_m^0 = B_m + qC_m$, e termina quando $W_m^{n+1} = W_m^n$ ou quando $J_m + W_m^{n+1}(q) - qT_m + C_m > D_m$, respectivamente; onde o atraso é calculado, ou quando a mensagem não é escalonável.

Por fim, o tempo de resposta no pior caso é dado pela Equação 2.16.

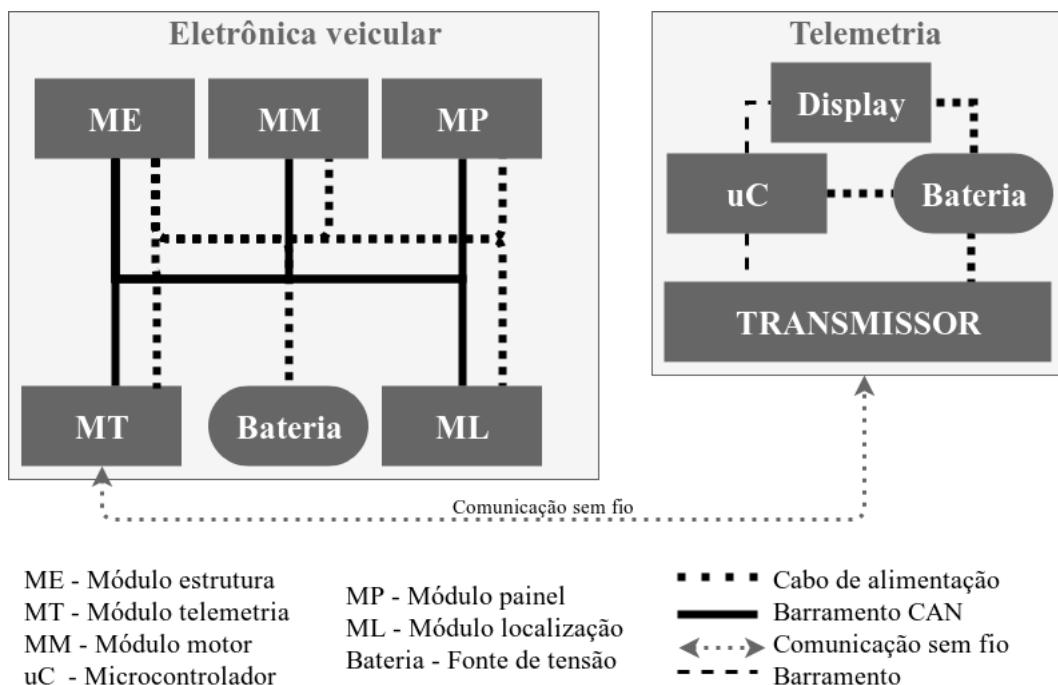
$$R_m = \max_{q=0 \dots Q_m-1} (R_m(q)) \quad (2.16)$$

3 Sistema Implementado

3.1 Arquitetura do sistema implementado

O sistema modelado e desenvolvido, representado na Figura 13, consiste de dois subsistemas chamados de eletrônica veicular e telemetria. A eletrônica veicular é um sistema embarcado distribuído que é composto por cinco módulos, ou nós, que se comunicam via um barramento CAN (*Controller Area Network*) com dois fios. Todos os nós são alimentados por uma única bateria de 12 V e 7 Ah. A telemetria é um sistema embarcado e consiste de uma unidade móvel com transmissor alimentada por uma bateria de 12 V e 7 Ah. Os dois subsistemas se comunicam via uma rede ponto a ponto 802.11/WiFi.

Figura 13 – Diagrama de blocos do sistema desenvolvido.



Fonte: O Autor.

Nas seções seguintes serão apresentados de um modo mais detalhado os componentes de hardware e software, e o funcionamento de cada subsistema. Além disso, serão apresentados o método para projetar e analisar o sistema considerando o comportamento temporal.

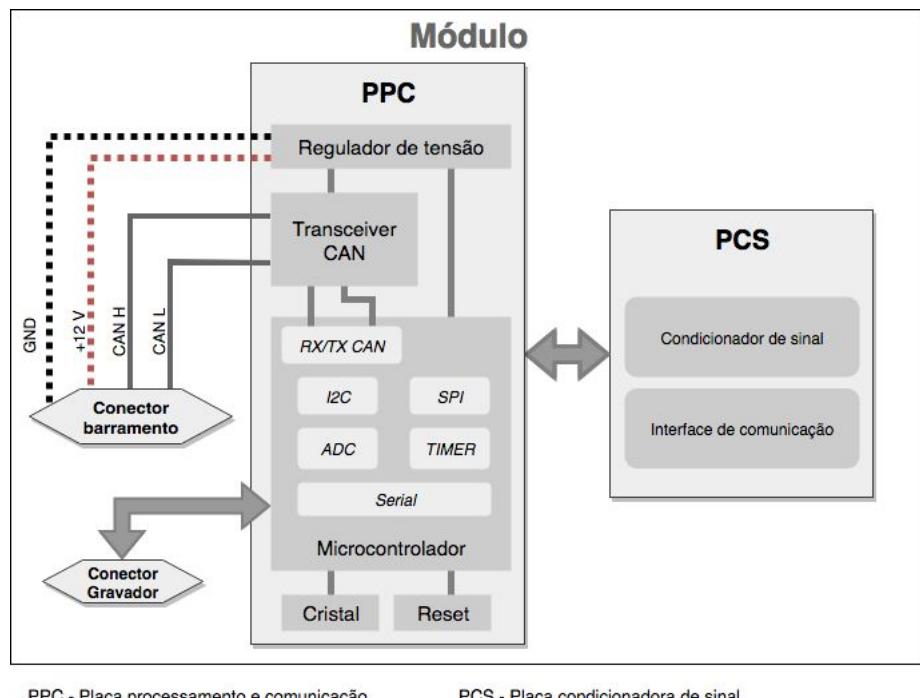
3.2 Sistema eletrônico veicular

Com base nos domínios apresentados por Navet e Simonot-Lion (2009), a eletrônica veicular conta com cinco módulos que comunicam-se entre si. Esses módulos foram instalados no veículo de acordo com a localização dos pontos de medição das grandezas físicas e suas funções.

A comunicação entre módulos é feita por meio de barramento CAN. O barramento CAN é o sistema de comunicação padrão no setor automobilístico de acordo com Bosch et al. (2007). Para mais detalhes sobre o barramento consultar a seção 2.4.

Cada módulo é formado por duas placas, sendo uma placa para processamento/comunicação (PPC) e outra para condicionamento de sinais (PCS), conforme representado na Figura 14. Além das placas, cada módulo conta com um *firmware* específico carregado na PPC.

Figura 14 – Diagrama de blocos de um módulo eletrônico.



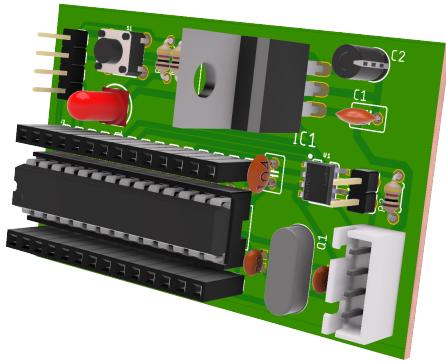
PPC - Placa processamento e comunicação

PCS - Placa condicionadora de sinal

Fonte: O Autor.

A PPC, representada na modelagem da Figura 15, consiste de um circuito para regulação de tensão, um microcontrolador PIC18F2580 (INC, 2009) e seus circuitos *reset* e cristal e um transceptor CAN MCP2551 (INC, 2007). Além disso, há um conector para o barramento com quatro vias (+12 V, GND, CAN H e CAN L) e um conector gravador, usado para carregar o *firmware* no microcontrolador via a placa PICKit 2 (INC, 2008). Todos os componentes são conectados de acordo com o diagrama do Anexo A.

Figura 15 – Vista em perspectiva do projeto da placa de processamento e comunicação (PPC).



Fonte: O Autor.

Todos os microcontroladores dos módulos (PPCs) foram programados em linguagem C usando o compilador CCS (SERVICES, 2018). Conforme apresentado na Figura 14, o microcontrolador dispõe de recursos/protocolos disponíveis para uso. O *firmware* utiliza esses recursos de acordo com a necessidade da PCS ou da função do módulo.

O *firmware* utiliza o sistema operacional de tempo real disponibilizado pela CCS (HILL, 2018). Esse sistema consiste de um executivo cíclico com capacidade de comunicação entre tarefas, capacidade de ativação e desativação de tarefas e gerenciamento dos recursos compartilhados.

A PCS consiste de circuitos eletrônicos usados para condicionar sinais de entrada provenientes de sensores e sinais de saídas fornecidos a dispositivos de visualização/sinalização tais como, display e LEDs (*Light Emitting Diode*). Cada módulo do sistema eletrônico veicular possui uma PCS específica.

Nas seções seguintes serão apresentados os detalhes dos cinco módulos do sistema eletrônico veicular.

3.2.1 Módulo estrutura

O módulo estrutura (ME) é responsável pela aferição do domínio do chassis por meio das seguintes grandezas: nível da carga da bateria e nível do combustível do veículo. Conforme representado na vista explodida da Figura 16, esse módulo conta com o conjunto PCS e PPC além de uma caixa de proteção com conector.

A aferição do nível da carga da bateria é realizada via leitura da tensão de saída da bateria (+12 V e GND). De acordo com manual técnico (UNICOBA, 2018), a bateria

Figura 16 – Vista explodida do projeto do módulo eletrônico da estrutura.



Fonte: O Autor.

está descarregada quando a tensão for igual a 11,8 V.

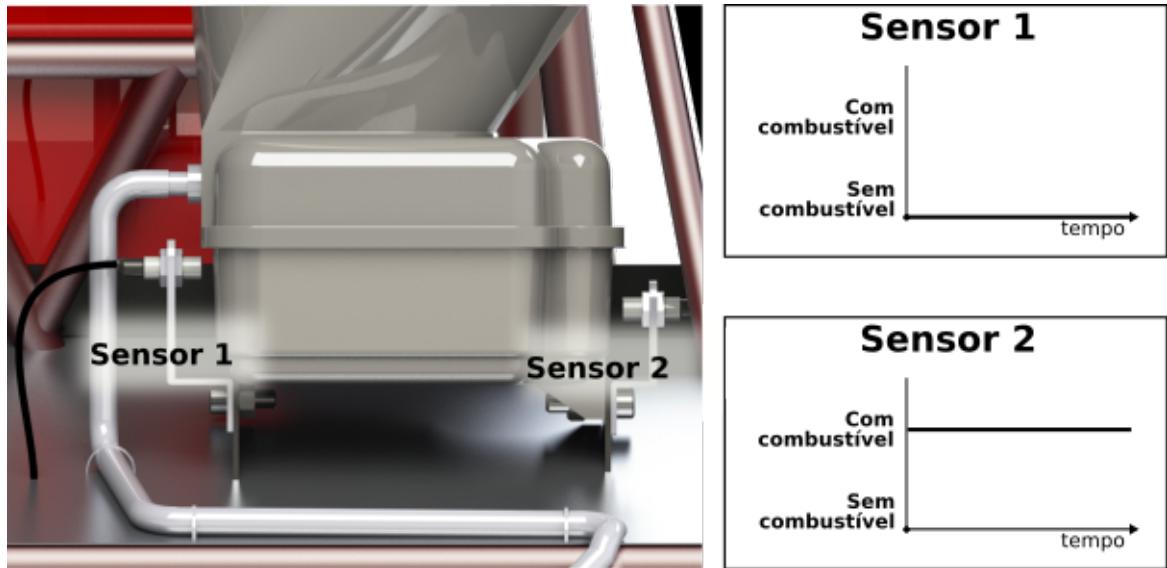
O sinal de carga da bateria é reduzido linearmente para níveis de tensão aceitos pela PPC por meio de um circuito divisor de tensão. A leitura é feita por meio do ADC (*Analog-to-digital converter*) disponibilizado pelo PIC18F2580 da PPC. O PIC18F disponibiliza conversores com precisão de 10 bits, de maneira que a menor variação captada pelo conversor é de 4 mV uma vez que a maior tensão (Vref+) é igual a 5 V e a menor tensão (Vref-) é igual a 0 V.

O nível do combustível é mensurado por meio de sensores capacitivos CR18-8DP (SERIES, 2008). Dois sensores são utilizados de maneira a mensurar o nível de combustível acima da metade do reservatório, abaixo da metade e na reserva. Os sensores capacitivos são fixados na estrutura do veículo com a face sensora apontada para o tanque de combustível, conforme representado na Figura 17.

Esse sensor possui três fios: alimentação, terra e sinal. A tensão de alimentação pode variar entre 6 V até 36 V. O sinal, representando por meio de gráfico na Figura 17, varia de acordo com a presença ou não de combustível. Com as oscilações normais do veículo, o combustível move-se dentro do reservatório. Devido ao movimento do combustível, um filtro via *firmware* é aplicado ao sinal de maneira a evitar leituras erradas dos sensores.

De acordo com os gráficos da Figura 17 o nível de combustível está abaixo do sensor 1, uma vez que o sinal está em sem combustível, e acima do sensor 2 dado que o sinal indica a presença de combustível.

Figura 17 – Esquema de instalação dos sensores capacitivos e os gráficos dos respectivos sinais.



Fonte: O Autor.

O *firmware* é composto por três tarefas periódicas, sendo uma tarefa para aquisição da carga da bateria (τ_{ME3}), uma tarefa para aquisição do nível de combustível (τ_{ME2}) e uma tarefa responsável por enviar as leituras por meio da rede CAN (τ_{ME1}). As tarefas são definidas de acordo com a Tabela 5.

Tabela 5 – Tarefas do módulo da estrutura.

Tarefa	C	T
τ_{ME1}	0,1	500
τ_{ME2}	0,1	500
τ_{ME3}	0,1	1000

Fonte: O Autor.

Onde:

C : Tempo de computação da tarefa [ms];

T : Período de ativação da tarefa [ms].

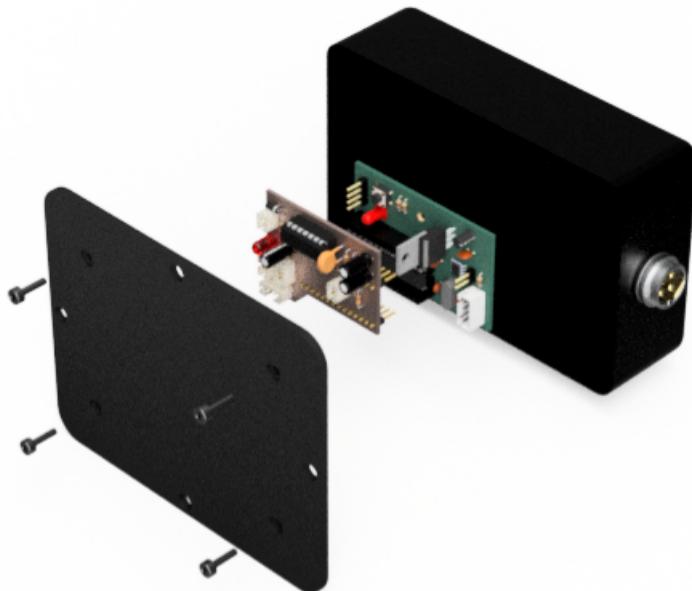
Sabendo que $C_{\tau_{ME1}} + C_{\tau_{ME2}} + C_{\tau_{ME3}} = 0,3$ [ms] e $T_{min} = 500$ [ms], o módulo tem espaço de tempo realizável, uma vez que $\sum_{i=1}^3 C_{\tau_{ME_i}} < T_{min}$.

Os detalhes elétricos contendo o esquemático com conexões e circuitos, bem como a montagem das placas encontram-se no Anexo B.

3.2.2 Módulo motor

O módulo motor (MM) mensura as seguintes grandezas: rotação do motor por minuto, temperatura do fluido de arrefecimento do motor, temperatura do fluido de arrefecimento da transmissão e o tempo de funcionamento do motor (horímetro). Conforme representado na vista explodida da Figura 18, esse módulo conta com o conjunto PCS e PPC além de uma caixa de proteção com conector.

Figura 18 – Vista explodida do módulo motor.



Fonte: O Autor.

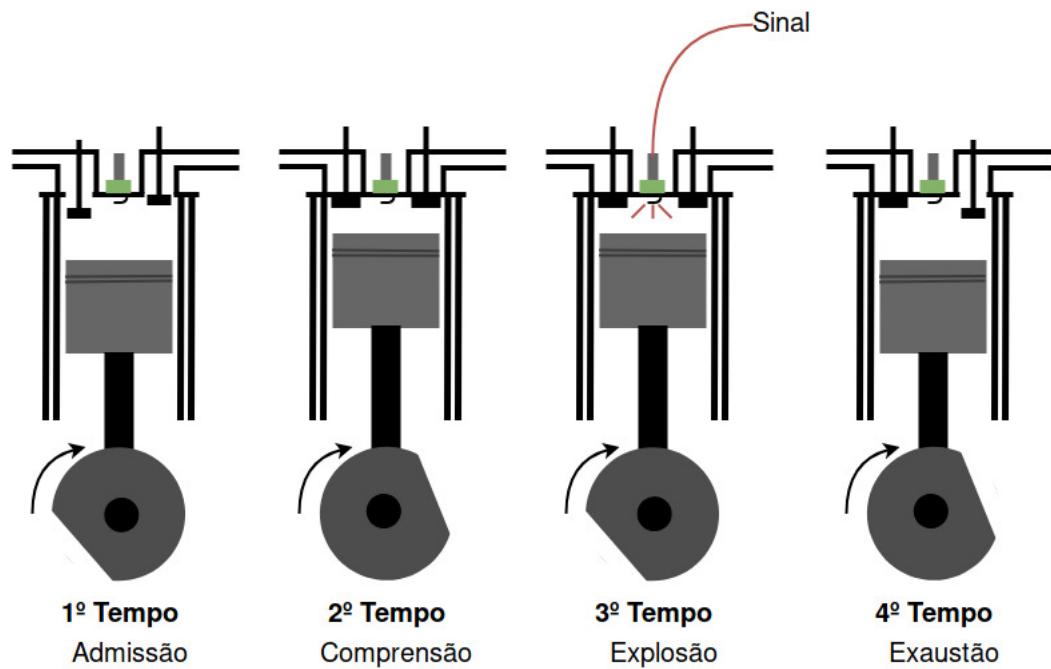
O motor utilizado no veículo é do tipo Briggs & Stratton da série 19. Esse motor é monocilíndrico de 4 (quatro) tempos com potência de 10 HP. De acordo com o fabricante esse motor utiliza o sistema de ignição *Magnetron* (*Magnetron Electronic Ignition*). Com esse sistema de ignição a vela dispara uma faísca no ciclo de explosão.¹

A rotação motora é determinada por meio do sinal produzido pelo sistema de ignição, representado na etapa de explosão da Figura 19. Esse sinal é periódico com frequência proporcional a rotação do volante do motor (eixo motor). O sinal produzido pelo sistema de ignição do motor está disposto na Figura 20.

Uma vez que o sinal é proporcional a rotação motora, podemos então encontrar a rotação em minutos.

¹ Dependendo do sistema de ignição outra centelha pode ser disparada na etapa de exaustão, sistemas com *wasted spark*.

Figura 19 – Descrição dos ciclos de funcionamento do motor (ciclo Otto).



Fonte: O Autor.

Figura 20 – Análise do sinal produzido pelo sistema de ignição.



Fonte: O Autor.

Considere:

f_{si} : Frequência do sistema de ignição [Hz];

rpm : Rotação motor em minutos [RPM].

Sabendo que:

$$1 \text{ [Hz]} = 60 \text{ [RPM]} \quad (3.1)$$

Então:

$$rpm = f_{si} * 60 \quad (3.2)$$

De acordo com a equação 3.2 e a frequência do sinal da Figura 20, sabemos que o motor está funcionando a $30.3 * 60 \approx 1800 RPM$.

O sinal produzido (Figura 20) pode variar de +15 V até -20 V. Uma vez que a PPC não suporta essa faixa de operação, esse sinal deve ser condicionado. Para tanto, a PCS conta com um conversor de frequência em tensão. Por meio desse conversor, o sinal periódico é convertido em um sinal DC com tensão proporcional a sua frequência, ou seja, com tensão proporcional a rotação motora.

Para medir a temperatura do fluído de arrefecimento do motor e do fluído de arrefecimento da transmissão são utilizados sensores de temperatura DS18B20 (PRODUCTS, 2008). O sensor utilizado no motor foi fixado na tampa de óleo do motor, o sensor da transmissão foi fixado no interior da proteção de transmissão.

O DS18B20 (PRODUCTS, 2008) dispõe de três fios, sendo dois fios para alimentação e um para comunicação. Esse sensor funciona a 5 V, tem precisão programável de até 12 bits e realiza leitura de temperaturas entre -55°C até 125°C. A comunicação é feita por meio de protocolo One-Wire. O protocolo One-Wire (MANIYAR, 2008) tem como características principais: *clock* próprio por dispositivo e sincronização feita pelo mestre, níveis de tensão padrão CMOS/TTL e comunicação *half duplex*. A PCS é responsável por fornecer tensão para alimentação e uma interface de comunicação para a PPC.

O tempo de funcionamento do motor é mensurado por meio de temporizador. A PPC disponibiliza temporizadores por meio do PIC18F2580 (INC, 2009). O temporizador é disparado ou parado de acordo com a rotação do motor. O tempo total é contabilizado e armazenado em memória não volátil.

O *firmware* conta com cinco tarefas periódicas. Uma tarefa responsável pela aquisição da rotação motora (τ_{MM1}), uma tarefa encarregada por medir o tempo de funcionamento do motor (τ_{MM2}), uma tarefa responsável pela aquisição da temperatura do óleo do motor (τ_{MM3}), uma tarefa responsável pela aquisição da temperatura da transmissão (τ_{MM4}) e por fim uma tarefa para enviar as leituras por meio da rede CAN (τ_{MM5}). As tarefas são definidas de acordo com a Tabela 6.

Tabela 6 – Tarefas do módulo do motor.

Tarefa	C	T
τ_{MM1}	0,1	100
τ_{MM2}	0,1	1000
τ_{MM3}	2	500
τ_{MM4}	2	500
τ_{MM5}	0,1	100

Fonte: O Autor.

Onde:

C : Tempo de computação da tarefa [ms];

T : Período de ativação da tarefa [ms].

Sabendo que $C_{\tau_{MM1}} + C_{\tau_{MM2}} + C_{\tau_{MM3}} + C_{\tau_{MM4}} + C_{\tau_{MM5}} = 4,3 \text{ [ms]}$ e $T_{min} = 100 \text{ [ms]}$, o módulo tem espaço de tempo realizável, uma vez que $\sum_{i=1}^5 C_{\tau_{MMi}} < T_{min}$.

O esquemático elétrico detalhado com todas as conexões e circuitos utilizados, bem como a montagem das placas encontram-se no Anexo C.

3.2.3 Módulo localização

O módulo localização mensura a velocidade do veículo e o nível de fluído de freio. Conforme representando na vista explodida da Figura 21 esse módulo ainda contém dispositivos para proteção da eletrônica, além dos conjunto PPC e PCS.

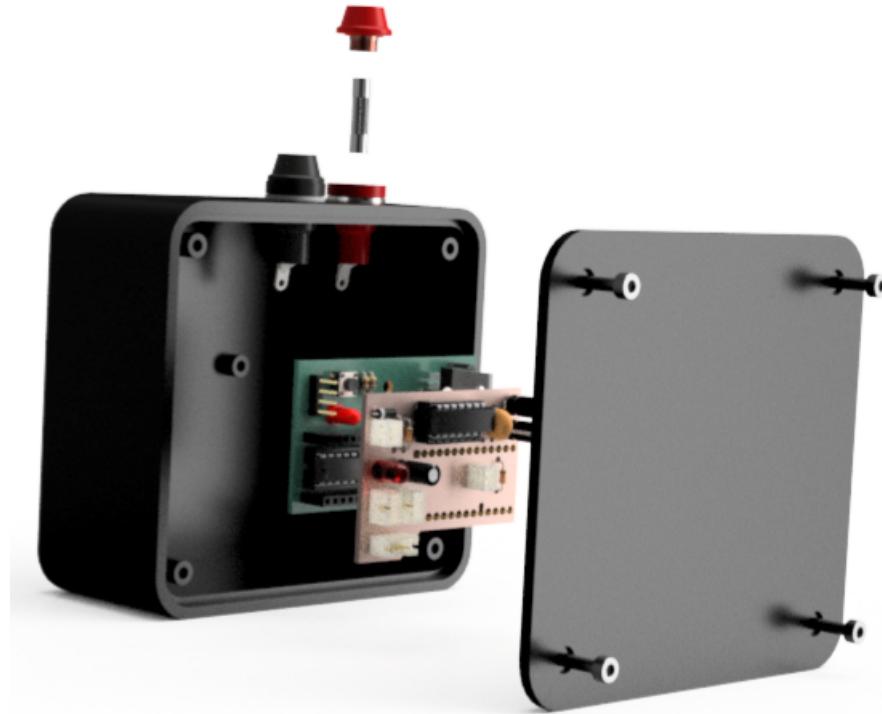
De acordo com Santini (2012), o dispositivo de proteção é um item necessário em circuitos automotivos. Ainda de acordo com Santini (2012), caso um condutor (fio) seja exposto a uma carga que não suporta, haverá sobreaquecimento com a possibilidade de incêndio.

Em razão da proximidade com a bateria, o módulo localização conta com dispositivos de proteção contra sobrecorrente. Conforme representado na vista explodida da Figura 21, o módulo conta com dois porta fusíveis.

O fusível é o dispositivo mais comumente utilizado para proteger os circuitos, segundo Santini (2012) e Hollembæk (2014). Logo, foi adota para proteção o fusível de vidro.

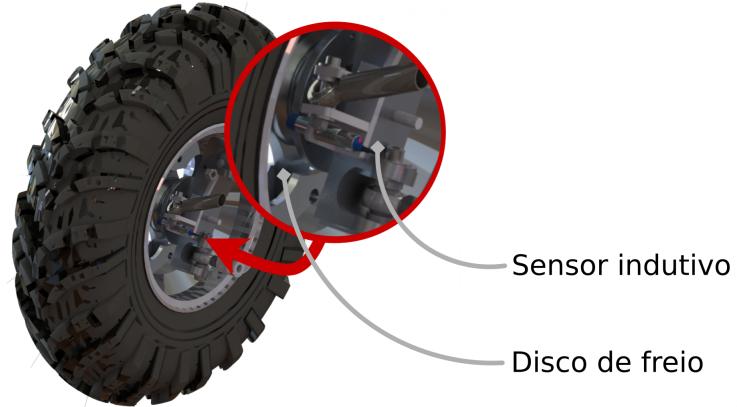
A velocidade do veículo é aferida por meio de um sensor indutivo LJ12A3E (CO., 2008). Esse sensor possui três fios: alimentação, terra (GND) e sinal; a tensão de funcionamento pode variar de 6 V até 36 V e foi fixado na roda dianteira do veículo, na altura do disco de freio conforme representado na Figura 22.

Figura 21 – Vista explodida do projeto do módulo eletrônico de localização.



Fonte: O Autor.

Figura 22 – Esquema de instalação do sensor indutivo na roda.

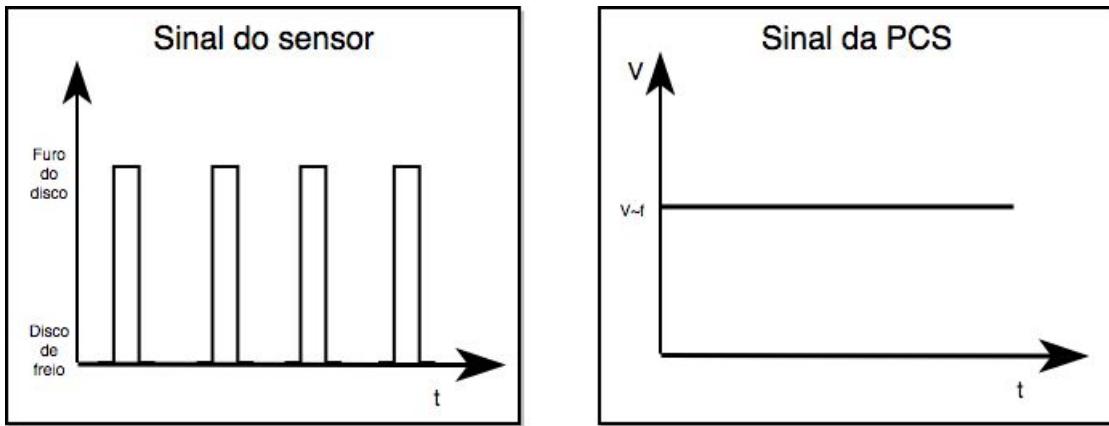


Fonte: O Autor.

Devido ao seu funcionamento, o sensor indutivo foi instalado de modo que os furos do disco de freio passem em frente a face sensora. A medida que o pneu gira, o sensor indutivo gera um pulso para cada ocorrência de furo, conforme representado na Figura 23. Para um disco de freio com furos simétricos, o sinal do sensor é periódico com frequência

proporcional a frequência angular do pneu. Esse sinal periódico é convertido na PCS do módulo em um sinal DC cuja tensão é proporcional a frequência angular do pneu, conforme representado na Figura 23. De acordo com Nicolazzi, Rosa e Leal (2012), em uma roda não motriz a velocidade tangencial é dada pela Equação 3.3.

Figura 23 – Gráficos com sinal do sensor e da PCS.



Fonte: O Autor.

$$v_t = r_d w_r \quad (3.3)$$

sendo,

v_t : Velocidade tangencial do pneu [m s^{-1}];

r_d : Raio dinâmico do pneu [m];

w_r : Velocidade angular [rad s^{-1}].

A relação entre velocidade angular e frequência angular, que é mensurada pelo sensor, segundo Nicolazzi, Rosa e Leal (2012) é dada pela Equação 3.4.

$$w_r = \frac{\pi n_r}{30} \quad (3.4)$$

em que,

ω_r : Velocidade angular [rad s^{-1}];

n_r : Frequência angular [rpm].

Ainda de acordo com Nicolazzi, Rosa e Leal (2012), o diâmetro do pneu varia de acordo a velocidade. A relação entre diâmetro estático e dinâmico pode ser aproximado segundo a Equação 3.5.

$$\begin{aligned} r_e &= 0,47D \\ r_d &= 1,02r_e \end{aligned} \quad (3.5)$$

onde,

$$\begin{aligned} r_e &: Raio \text{ estático [m]}; \\ r_d &: Raio \text{ dinâmico [m]}; \\ D &: Diâmetro \text{ externo do pneu [m]}. \end{aligned}$$

O nível do fluído de freio é mensurado nesse módulo devido a proximidade ao ponto de medição e é realizado por meio de uma chave de nível. O sensor utilizado é o mesmo do veículo modelo Pálio da marca FIAT. Esse sensor funciona como uma chave de maneira que o circuito fica aberto para fluído acima do recomendado e fechado para abaixo do recomendado.

Esse módulo conta com três tarefas periódicas, sendo uma tarefa responsável por determinar a velocidade do veículo (τ_{ML1}), uma tarefa responsável pelo envio das leituras por meio da rede CAN (τ_{ML2}) e por fim uma tarefa que avalia a situação do fluído de freio (τ_{ML3}). As tarefas são definidas de acordo com a Tabela 7.

Tabela 7 – Tarefas do módulo da localização.

Tarefa	C	T
τ_{ML1}	0,1	100
τ_{ML2}	0,1	100
τ_{ML3}	0,1	500

Fonte: O Autor.

Onde:

$$\begin{aligned} C &: \text{Tempo de computação da tarefa [ms]}; \\ T &: \text{Período de ativação da tarefa [ms]}. \end{aligned}$$

Sabendo que $C_{\tau_{ML1}} + C_{\tau_{ML2}} + C_{\tau_{ML3}} = 0,3 \text{ [ms]}$ e $T_{min} = 100 \text{ [ms]}$, o módulo tem espaço de tempo realizável, uma vez que $\sum_{i=1}^3 C_{\tau_{MLi}} < T_{min}$.

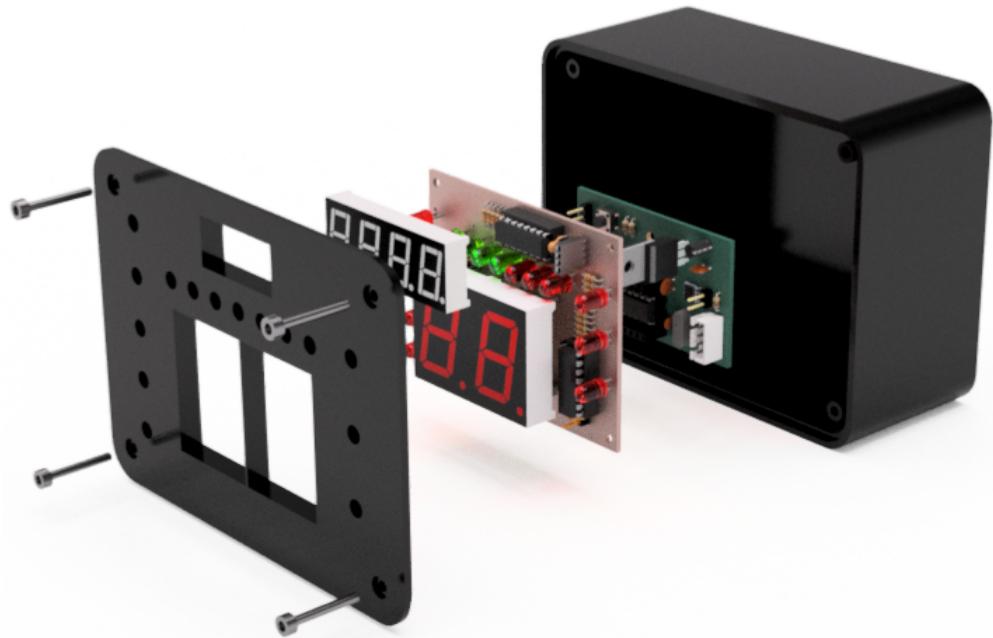
Maiores detalhes desse módulo, tais como esquemático elétrico e montagem, encontram-se no Anexo D.

3.2.4 Módulo painel

As informações que exigem reação do motorista devem ser apresentadas no painel de instrumentos. Logo esse módulo é responsável por exibir as informações dinâmicas e de monitoramento ao motorista. Para tanto, o painel conta com *displays* e LEDs dispostos de acordo com a Figura 25.

Para o acoplamento dos *displays* e LEDs, o módulo do painel conta com uma caixa. Essa caixa contém aberturas projetadas de acordo com a posição dos dispositivos de interação com o motorista. A modelagem do painel, bem como as placas PCS e PPC, estão representados na vista explodida da Figura 24.

Figura 24 – Vista explodida do projeto do painel de instrumentos.



Fonte: O Autor.

Conforme esquematizado na Figura 25, as informações exibidas são: horímetro, velocímetro, tacômetro, nível de combustível e luzes de advertência/diagnóstico. Essas informações são adquiridas dos outros módulos por meio do barramento CAN.

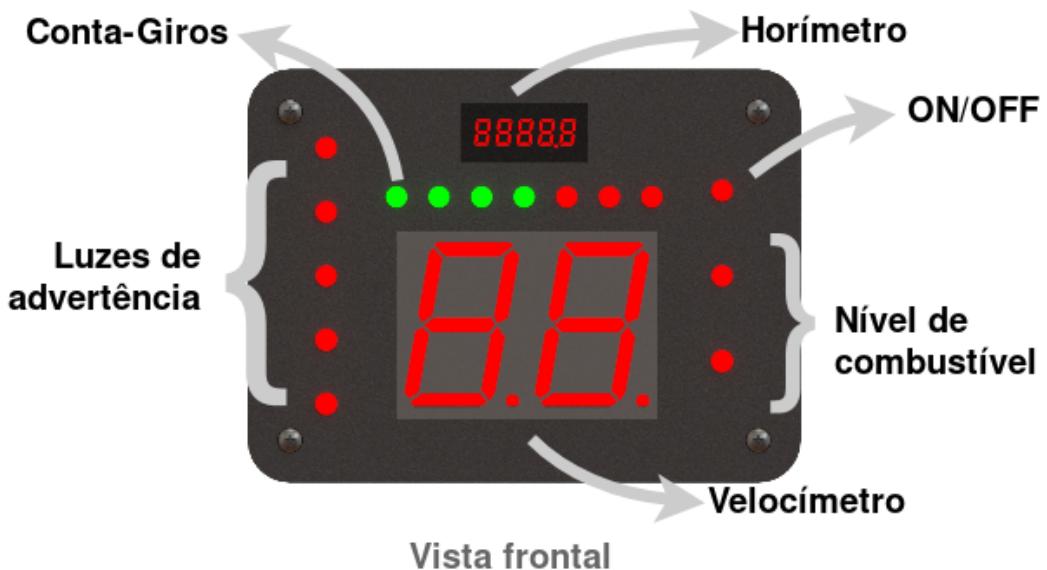
O tempo de funcionamento do motor é mostrado em quatro *displays* de sete segmentos. Os *displays* são controlados por meio do *driver* TM1637 (ELECTRONICS,). Para funcionar, o *driver* junto com o display de quatro dígitos são alimentados com +5 V e a comunicação é feita utilizando o protocolo de comunicação próprio do *driver*.

Para exibir a velocidade do veículo são utilizados dois *displays* de sete segmentos. Os *displays* são controlados pela PPC por meio de registradores de deslocamento (*shift registers*) 74HC595 (INSTRUMENTS, 2015) e são alimentados com tensão de +5 V. A comunicação da PPC com o 74HC595 é feita por meio das saídas digitais.

A rotação do motor, nível do combustível e as luzes indicadoras² são LEDs controlados diretamente pela PPC. O painel conta com um mecanismo para verificar o

² O manual do proprietário do PÁLIO modelo 2018 da FIAT chama de LUZES-ESPIA.

Figura 25 – Vista frontal da modelagem do painel de instrumentos.



Fonte: O Autor.

funcionamento dessas luzes. Esse mecanismo de verificação acende todos os LEDs do painel logo após ligar a eletrônica.

O funcionamento do tacômetro é semelhante ao utilizado em veículos esportivos (AMERICA, 2018) ou de competição (MCLAREN®, 2018), onde a medida que a rotação do motor aumenta o número de luzes acesas aumenta proporcionalmente. O medidor de combustível é semelhante ao tacômetro, contudo a medida é o nível de combustível.

O sistema de advertência e diagnóstico acende as luzes de aviso e luzes indicadoras, respectivamente, no quadro de instrumentos. As luzes de advertência e de diagnóstico são as mesmas fisicamente, a diferença é o tipo do alerta emitido. Cada luz tem associada um símbolo e uma cor definida de acordo com o padrão ISO (*International Organization for Standardization*) 2575 (STANDARDIZATION, 2000).

O sistema de diagnóstico ativa uma, ou mais, luzes de aviso caso seja detectado a desconexão de módulo ou sensor. Dessa maneira, caso uma luz do painel de instrumentos permaneça acesa após o veículo entrar em funcionamento, significa que o sistema de diagnóstico detectou um defeito. Os defeitos detectados e o seu respectivo alerta são definidos na Tabela 8.

O sistema de advertência ativa o alerta, caso seja detectado uma situação que pode se tornar grave em relação ao veículo. O alerta do sistema de advertências consiste em piscar a luz indicadora relacionada ao problema. As situações de risco são definidas de acordo na Tabela 9.

Tabela 8 – Defeitos informados pelo painel.

Luz acessa (de cima para baixo)	Defeito encontrado
Primeira	Módulo de telemetria com problema.
Segunda	Módulo de localização com problema.
Terceira	Módulo da estrutura com problema.
Quarta e quinta	Módulo do motor com problema.
Quarta	Sensor de temperatura da motor com problema.
Quinta	Sensor de temperatura da transmissão com problema.

Fonte: O Autor.**Tabela 9** – Advertências informadas pelo painel.

Alerta	Valor de risco
Óleo do motor	> 120 °C
Temperatura da transmissão	> 70 °C
Fluído de freio	Abaixo do indicado
Tensão da bateria	< 12,0 V

Fonte: O Autor.

O *firmware* do painel é composto por cinco tarefas, cada tarefa fica responsável por atualizar parte das informações disponibilizadas ao motorista. Dessa maneira, uma tarefa atualiza a velocidade (τ_{MP1}), uma tarefa atualiza o tacômetro (τ_{MP2}), uma tarefa atualiza o medidor de combustível (τ_{MP3}), uma tarefa atualiza o horímetro (τ_{MP4}), por fim uma tarefa para controlar as luzes de advertências (τ_{MP5}). As tarefas são definidas de acordo com a Tabela 10.

Tabela 10 – Tarefas do módulo do painel.

Tarefa	C	T
τ_{MP1}	0,1	100
τ_{MP2}	0,1	100
τ_{MP3}	0,1	500
τ_{MP4}	0,1	500
τ_{MP5}	0,1	400

Fonte: O Autor.

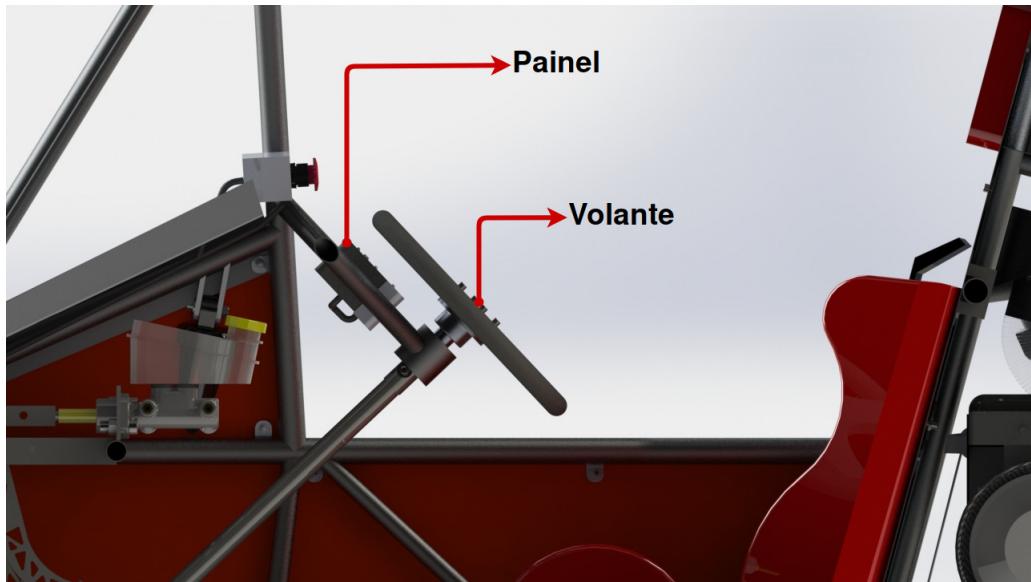
Onde:

 C : Tempo de computação da tarefa [ms]; T : Período de ativação da tarefa [ms].

Sabendo que $C_{\tau_{MP1}} + C_{\tau_{MP2}} + C_{\tau_{MP3}} + C_{\tau_{MP4}} + C_{\tau_{MP5}} = 0,5 \text{ [ms]}$ e $T_{min} = 100 \text{ [ms]}$, o módulo tem espaço de tempo realizável, uma vez que $\sum_{i=1}^5 C_{\tau_{MP_i}} < T_{min}$.

O painel foi instalado logo atrás do volante sem obstruir a visão do motorista e de fácil acesso, conforme representado na Figura 26.

Figura 26 – Posição do painel em relação ao volante e motorista.



Fonte: O Autor.

Maiores detalhes desse módulo, particularmente o esquemático elétrico, encontra-se no Anexo E.

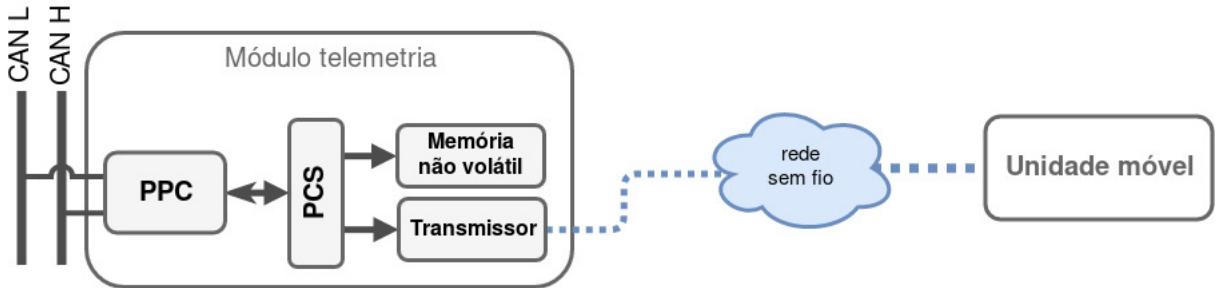
3.2.5 Módulo telemetria

Conforme Navet e Simonot-Lion (2009), os componentes que permitem troca de informação entre o veículo e o mundo são do domínio da telemática. Como exemplos, são citados, rádio e sistema de navegação. Ainda de acordo com Navet e Simonot-Lion (2009), os sistemas telemáticos apresentam requisitos diferentes dos outros módulos eletrônicos. A telemetria deve apresentar alta capacidade de transferência de dados.

O módulo de telemetria embarcado é um *gateway* entre a rede CAN veicular e os sistemas fora do carro, conforme representado na Figura 27. O módulo também mantém um histórico do funcionamento do veículo, armazenando todas as mensagens da rede CAN em memória não volátil.

Esse módulo conta com um transmissor ESP8266 (ESPRESSIF, 2018) conectado a unidade móvel por rede sem fio padrão IEEE 802.11. O ESP8266 suporta protocolos de comunicação 802.11 b/g/n, funciona com tensão de alimentação de 3.3 V e tem a pilha de

Figura 27 – Diagrama em blocos do módulo telemetria.



Fonte: O Autor.

protocolos TCP/IP. Esse transmissor também disponibiliza as interfaces de comunicação UART, I2C e SPI.

O ESP8266 é responsável pela conexão com o mundo exterior ao veículo, para isso conta com um *firmware* próprio. A PPC trata as informações da rede CAN e repassa ao ESP8266 apenas o que será transmitido. A comunicação entre PPC e ESP8266 é feita por meio da interface UART.

Para memória não volátil o módulo utiliza um cartão microSD. O cartão conta com capacidade de 8 GB. Para acessar a memória é utilizado o *driver* disponibilizado pelo compilador CCS (SERVICES, 2018).

O transmissor é acoplado a PPC por meio da PCS, conforme representado na vista explodida da Figura 28. Além de acoplar o transmissor, a PCS disponibiliza uma interface para realizar alteração no *firmware* do transmissor.

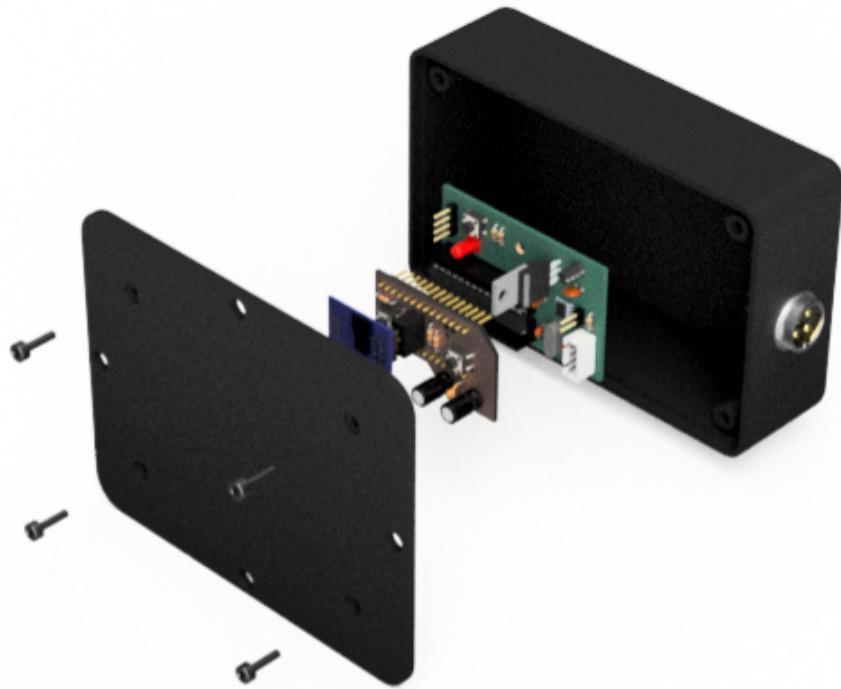
Para enviar as informações para o transmissor, o *firmware* da PPC dispõe de uma função, ativada periodicamente. A PPC recebe todas as mensagens da rede veicular, contudo, envia apenas o nível de combustível, tensão da bateria e temperatura do óleo do motor para o transmissor.

Para salvar as mensagens da rede veicular no cartão de memória, a PPC conta com uma interrupção, ou função, que é ativada sempre que o módulo recebe uma mensagem válida. Além de armazenar as informações da mensagem CAN, o *firmware* armazena também um indexador baseado no tempo que a mensagem foi recebida.

Por fim, o conjunto das placas é protegido por uma caixa, que dispõe de conector, conforme na vista explodida da Figura 28.

Maiores detalhes desse módulo, como por exemplo esquemático elétrico e montagem, encontra-se no Anexo F.

Figura 28 – Vista explodida do projeto do módulo da telemetria.



Fonte: O Autor.

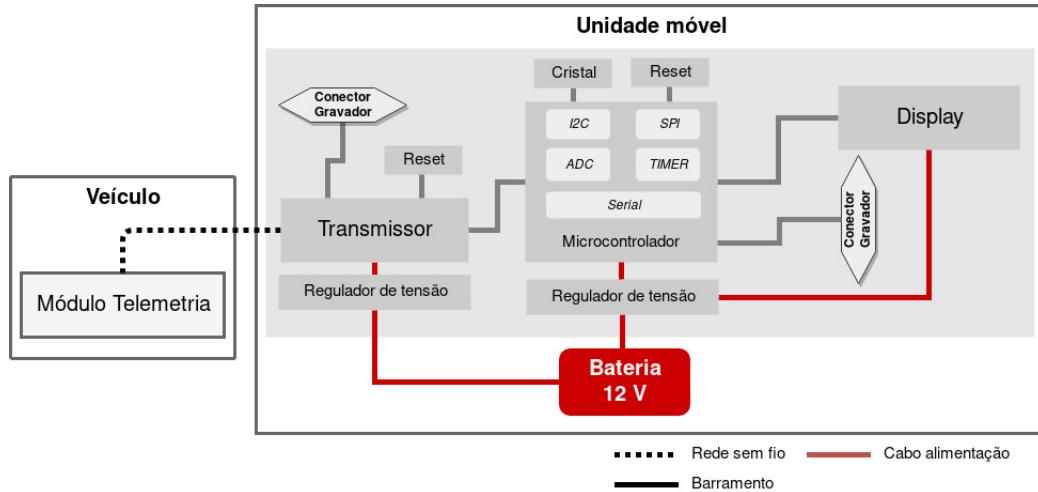
3.3 Subsistema Telemetria

O subsistema telemetria é composto de duas partes. Uma parte fixada no veículo, tratada por módulo de telemetria e detalhado na seção 3.2.5. Outra parte móvel, detalhada a seguir e chamada de unidade móvel. Ambas as unidades estão representadas na Figura 29.

A unidade móvel é responsável por prover a infraestrutura necessária para conexão com o veículo e por disponibilizar aos usuários informações sobre o veículo com o mesmo em funcionamento. Para tanto conta com transmissor e seus circuitos de funcionamento, microcontrolador e seus circuitos e um *display LCD*. Os detalhes da unidade móvel estão representados no diagrama de blocos da Figura 29.

Para prover a infraestrutura, a unidade móvel conta com um transmissor ESP8266 (ESPRESSIF, 2018) com regulador de tensão e *interface* para atualização do *firmware*.

O transmissor é configurado como ponto de acesso e servidor, de maneira que o mesmo administra a rede distribuindo endereços válidos para os usuários e impedindo acessos não autorizados por meio de autenticação. Para distribuir os endereços, o ESP8266 conta com o protocolo DHCP (*Dynamic Host Configuration Protocol*). A verificação de cada usuário é feita por meio de senha utilizando protocolo de segurança WPA2 (*Wi-Fi Protected Access II*).

Figura 29 – Diagrama em blocos do subsistema telemetria.

Fonte: O Autor.

O transmissor conta com um *firmware* próprio, que gerencia a infraestrutura. O *firmware* é responsável por detectar a conexão e desconexão do veículo, recebimento de mensagem do veículo e as tarefas envolvidas na autenticação e disponibilidade de endereço válido.

Para tratar as informações do transmissor e exibi-las em um *display*, a unidade móvel conta com um PIC18F2580 (INC, 2009). Além do microcontrolador, a unidade conta com os circuitos básicos de funcionamento. Um protocolo foi desenvolvido para permitir a comunicação entre o transmissor e o microcontrolador. Esse protocolo define quatro comandos básicos, todos dispostos na Tabela 11.

Para receber os comandos definidos no protocolo, o microcontrolador utiliza a interface UART. Com base nos comandos, o microcontrolador atualiza o *display*.

Tabela 11 – Protocolo de comunicação.

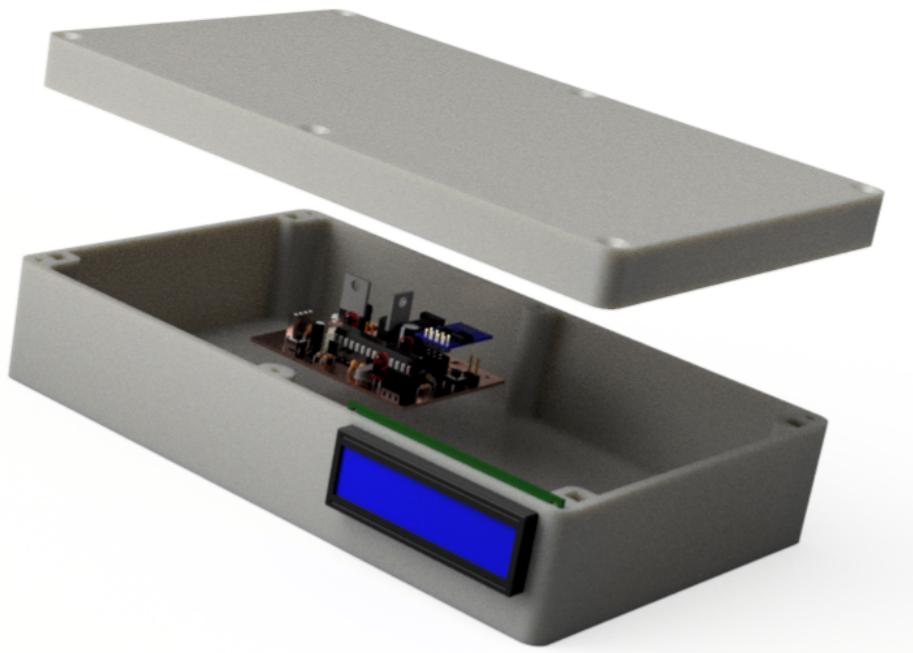
Comando	Descrição
S	O servidor do transmissor está ativo.
C	O veículo está conectado a unidade móvel.
D	O veículo está desconectado da unidade móvel.
V	O veículo está enviando dados.

Fonte: O Autor.

Para atualizar o *display*, o *firmware* utiliza o *driver* disponibilizado pelo compilador. O *display* da unidade é um LCD de 16x2 e informa o usuário sobre a situação do veículo.

A unidade móvel, por fim, foi condicionada dentro de uma caixa, conforme a modelagem da Figura 30.

Figura 30 – Modelagem da unidade móvel.



Fonte: O Autor.

3.4 Projeto e análise dos tempos de respostas do sistema

A eletrônica veicular foi analisada de acordo com os piores tempos de respostas das mensagens CAN proposta na seção 2.5 e os resultados serão discutidos no próximo capítulo. Por meio desta análise é possível determinar a menor taxa de transmissão respeitando os requisitos temporais das mensagens.

Para realizar a análise proposta a eletrônica veicular utiliza a política de atribuição de prioridade de *deadline* monotônico e não foi levado em consideração possíveis erros de transmissão, que acarretaria no aumento do tempo de transmissão por causa dos mecanismos de detecção e reenvio.

A eletrônica veicular conta com apenas uma mensagem aperiódica, logo, também foi desconsiderada da análise esse tipo de mensagem.

A análise consiste da verificação dos piores tempos de respostas das mensagens enviadas por meio dos módulos eletrônicos e a comparação com o *deadline* de cada mensagem. Se a mensagem pode ser entregue no pior cenário da rede CAN, logo a rede

será capaz de operar sobre as condições projetadas. Para realizar essa análise um conjunto de informações foram mensuradas de acordo com o módulo.

Embora a eletrônica veicular proposta na seção 3.2 seja composta por cinco módulos, apenas os módulos do motor (MM), de localização (ML) e de estrutura (ME) enviam mensagens periódicas no barramento CAN.

4 Resultados obtidos - módulos embarcados

Neste capítulo serão apresentados os resultados obtidos com a eletrônica veicular, o que engloba os módulos: da estrutura, do motor, de localização e o painel. O sistema de telemetria será abordado no próximo capítulo, juntamente com a análise da rede veicular.

4.1 Eletrônica veicular

Foram realizados testes com o sistema funcionando em um veículo de competição, para tanto, a eletrônica foi instalada no veículo da equipe Baajatinga, modelo BJT06. O automóvel utilizado está representado na Figura 31.

Figura 31 – Foto do automóvel utilizado para o teste da eletrônica.



Fonte: Equipe Baajatinga.

Os módulos eletrônicos foram instalados no veículo de acordo com a proximidade do ponto de medição. Para conectar os módulos o veículo conta com um barramento, tratado por chicote elétrico. O barramento é composto por dois pares traçados, totalizando 4 vias, além de contar com proteção eletrostática.

A eletrônica veicular foi utilizada durante a 25^a Competição Baja SAE Brasil. Todos os dados experimentais foram adquiridos durante a competição, em um ambiente que simula as situações reais dos veículos *off-road*.

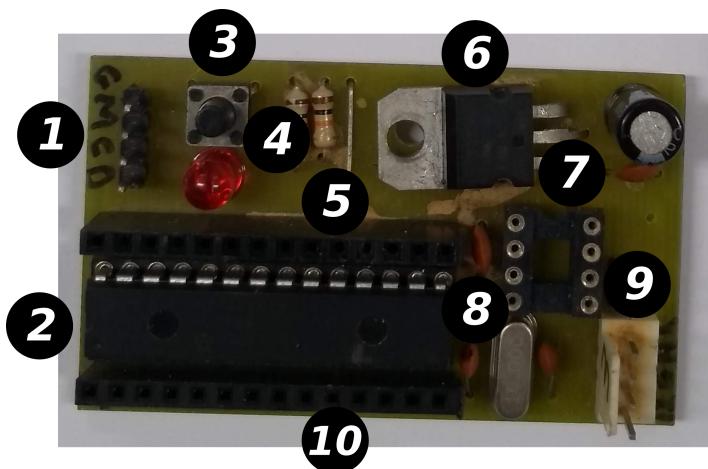
A seguir serão apresentados os módulos e todos os detalhes de seu funcionamento.

4.1.1 Placa de processamento e comunicação e placa para condicionamento de sinal

Todos as placas foram feitas em placas de circuito impresso (PCI) utilizando a prototipadora mecânica disponibilizada pelo Colegiado de Computação (CECOMP) da Universidade Federal do Vale do São Francisco. Os resultados obtidos estão a seguir:

Placa de processamento e comunicação - PPC

Figura 32 – Foto de uma PPC com os seus componentes numerados.



Fonte: O Autor.

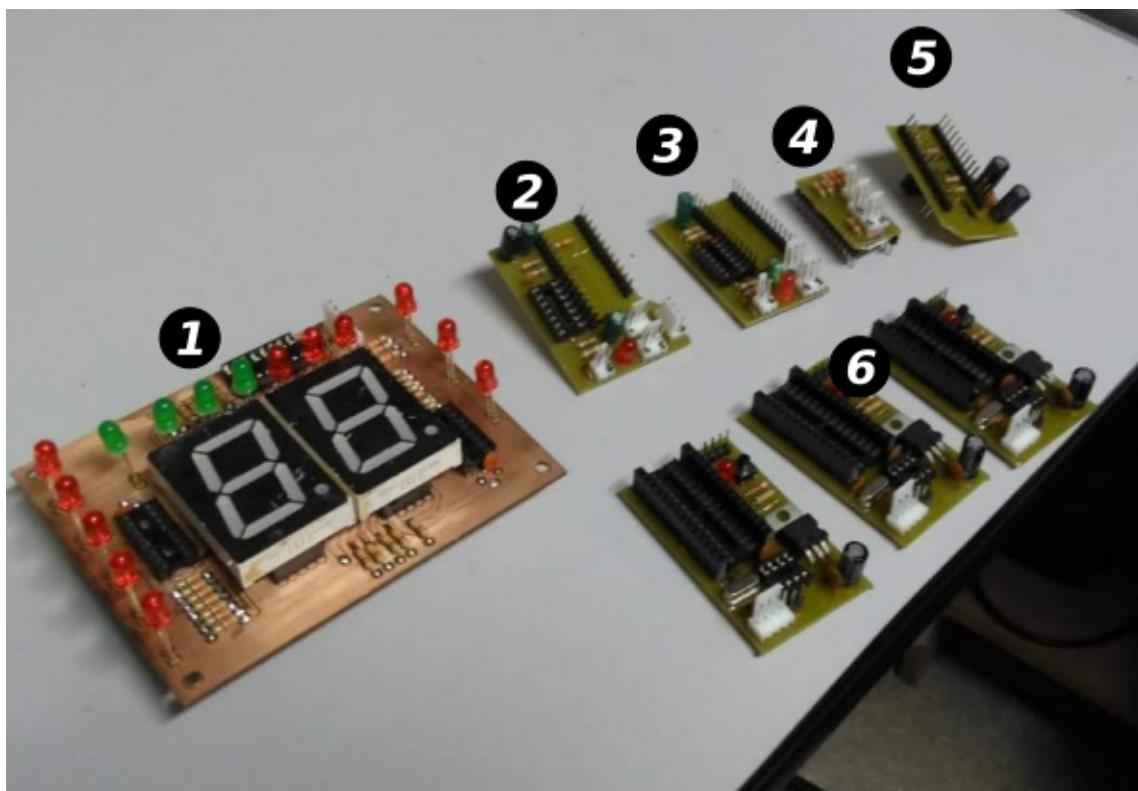
A placa de processamento ficou conforme a Figura 32. Ainda em relação a Figura 32, os componentes numerados na foto são:

1. Conector do gravador;
2. PIC18F2580;
3. Botão Reset;
4. LED;
5. Barra de conectores;

6. Regulador de tensão;
7. Socket para o *transceiver* CAN MCP2551;
8. Cristal de 10 MHz;
9. Conector do barramento;
10. Barra de conectores.

Placa para condicionamento de sinal - PCS

Figura 33 – Foto das placas condicionadoras de sinais (PCS) e de algumas placas de processamento e comunicação(PPC).



Fonte: O Autor.

As placas de condicionamento ficaram de acordo com a Figura 33. Em relação a Figura 33, considere:

1. PCS do painel de instrumentos;a
2. PCS do módulo de localização;
3. PCS do módulo do motor;

4. PCS do módulo da estrutura;
5. PCS do módulo da Telemetria;
6. PPC;

4.1.2 Módulo da estrutura

O módulo da estrutura é responsável por analisar o nível da bateria e o nível de combustível, dessa forma, esse módulo foi instalado próximo ao tanque de combustível.

As placas PCS e PPC foram condicionadas dentro do módulo, conforme representando na Figura 34a e 34b. Por fim, o módulo foi instalado na estrutura do veículo e conectado ao barramento do carro, conforme representado na Figura 34c.

Os sensores de combustível foram instalados com a face sensora próxima a parede do tanque de combustível do veículo. O sensor mais alto, em relação a estrutura do carro, detecta o tanque cheio de combustível. O segundo sensor detecta o nível de combustível na metade ou na reserva. Ambos os sensores estão representados na Figura 34d.

4.1.2.1 Firmware

Um fluxograma simplificado do funcionamento do módulo está representando na Figura 35. O código do módulo é dividido em interrupções e as tarefas periódicas.

Inicialmente o código configura a PPC, ativando as portas digitais, configurando o controlador CAN, configurando os conversores A/D e por fim ativando as interrupções. Uma vez que a PPC está configurada, o módulo entra no *main loop*.

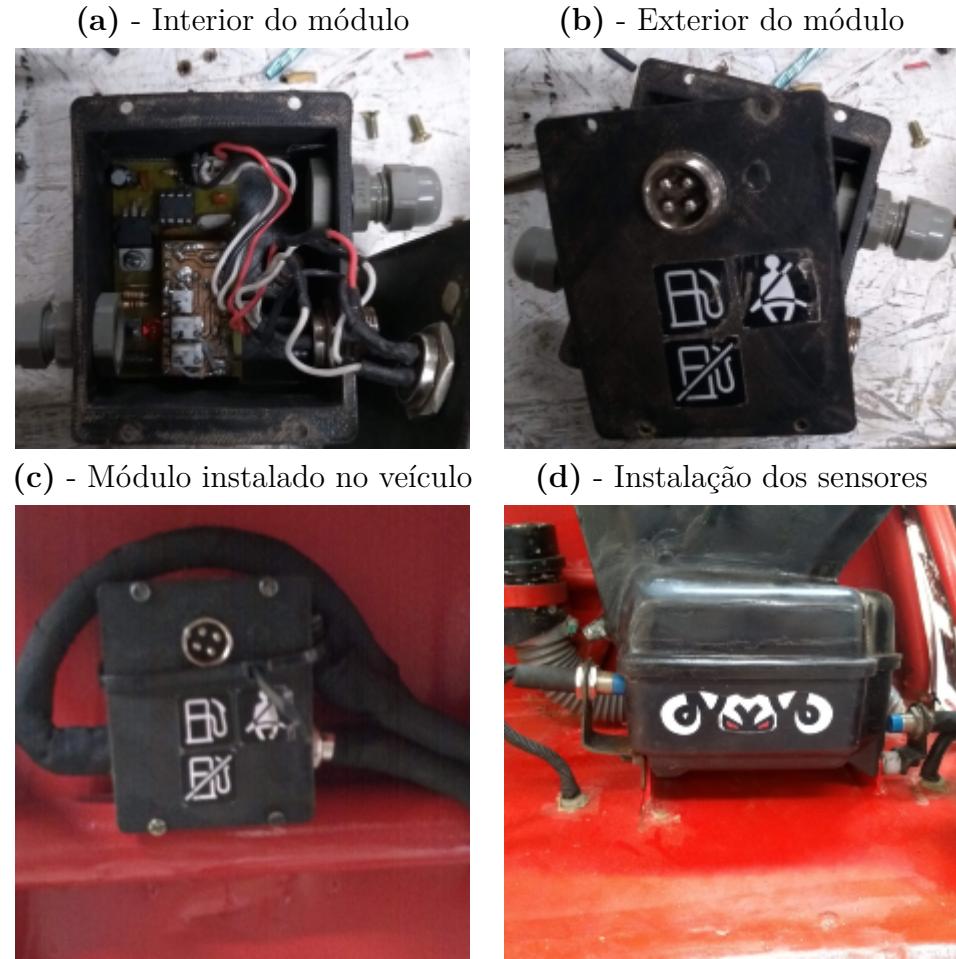
Dentro do *main loop*, o algoritmo alterna entre as tarefas baseado no período de ativação e ativa as interrupções de acordo com o disparo dos eventos.

O código da função CAN_ERROR apenas incrementa uma variável, dessa forma é possível acompanhar o número de erros da rede CAN no módulo.

Os códigos das tarefas são definidas de acordo com os fluxogramas das Figuras 36, 37 e 38.

Depois de tratar as interrupções ou após executar uma tarefa, o microcontrolador fica ocioso esperando o tempo correto para ativar a próxima tarefa ou o disparo de uma interrupção.

A tarefa *task_battery* realiza 10 (dez) leituras do conversor A/D na porta (AN0), com intervalo de 500 µs entre as leituras. Após as leituras, o algoritmo tira a média e converte a média para tensão [V]. Por fim o algoritmo multiplica a leitura por 10, melhorando a precisão da medida pois reduz a perda de informação quando retirar a parte decimal. A tarefa *task_battery* está representada em forma de fluxograma na Figura 36.

Figura 34 – Montagem e instalação do módulo.

Fonte: O Autor.

Conforme representado no fluxograma da Figura 37, a tarefa *task_CAN* prepara a mensagem em acordo com o ID do módulo; prepara os *buffers* de transmissão e por fim, ativa o envio. A mensagem CAN desse módulo é formada de acordo com a Tabela 12.

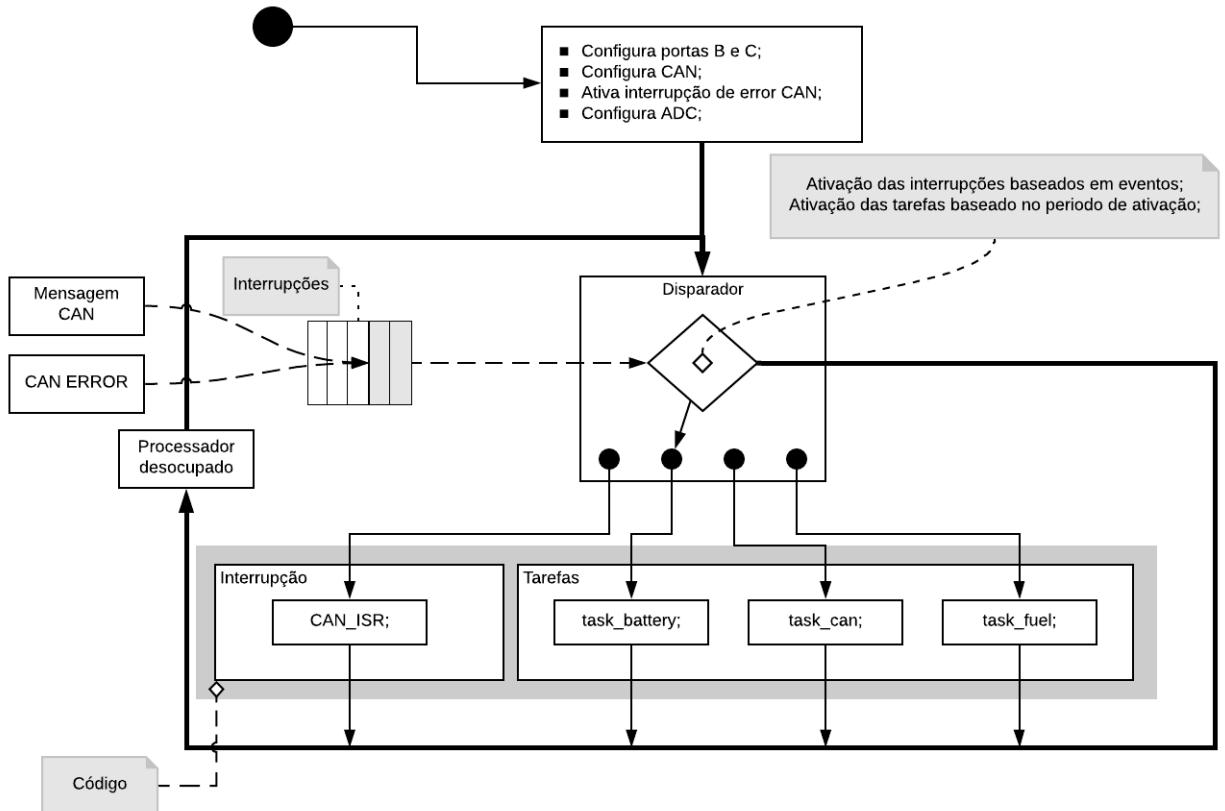
Tabela 12 – Resumo da mensagem CAN do módulo.

CAN ID	Mensagem							
	1	2	3	4	5	6	7	8
0x1CE0	Carga da bateria	Sensor 1	Sensor 2	Erro(s)				

Fonte: O Autor.

A tarefa *task_fuel*, por meio das portas digitais, verifica o estado dos sensores. Com base no valor lido do sensor o algoritmo incrementa ou decremente uma variável de estado do sensor. Essa variável evita que o módulo informe leituras erradas durante o movimento do combustível dentro do reservatório, uma vez que para o módulo detectar uma mudança no nível de combustível os sensores devem permanecer no mesmo estado durante, pelo

Figura 35 – Fluxograma simplificado do algoritmo do módulo.



Fonte: Adaptado de Samek (2008).

menos, 25 segundos. A tarefa *task_fuel* está representada em forma de fluxograma na Figura 38.

4.1.2.2 Resultados obtidos

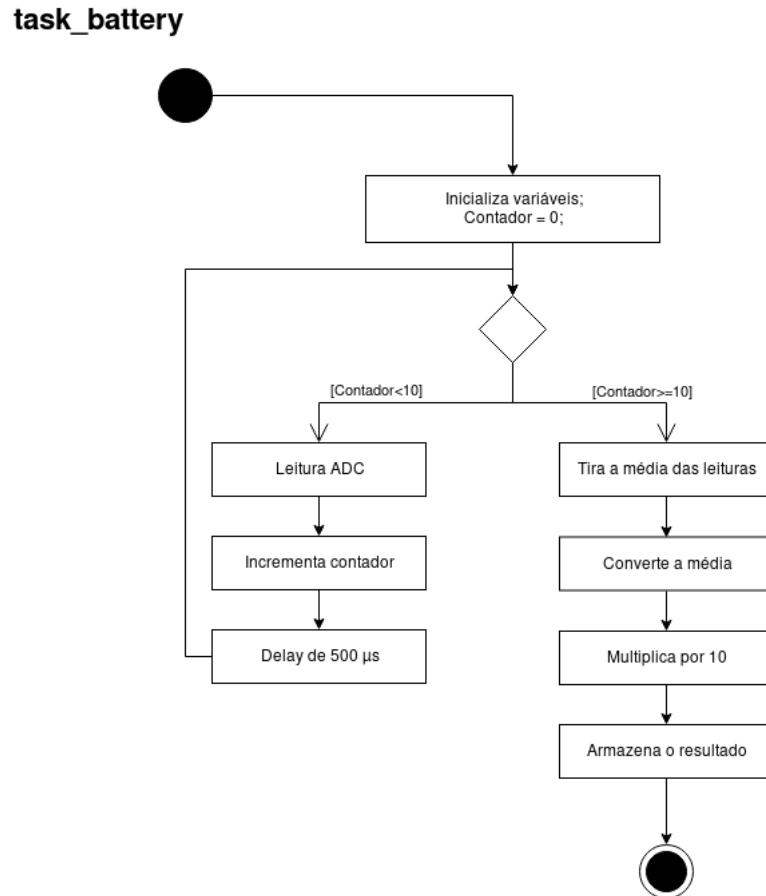
Com o módulo montado e com o *firmware* gravado no microcontrolador, os resultados a seguir foram obtidos.

Ativação das tarefas

O módulo foi conectado ao Matlab, por meio da porta serial, para verificação da ativação das tarefas. Embora com poucas mudanças, o algoritmo do módulo foi alterado. A principal modificação foi que o módulo passou a emitir uma mensagem, pela porta serial, quando uma tarefa era ativada.

A mensagem emitida pelo módulo contém a tarefa ativada e o instante da ativação. Para saber o instante de cada ativação, a PPC utilizou o TIMER1, disponível no microcontrolador, configurado com período de 0,8 µs e considerando o momento que o módulo é

Figura 36 – Fluxograma simplificado da tarefa *task battery*.



Fonte: O Autor.

liberado para execução, logo após soltar o botão de *reset*, como o instante inicial.

Com base nas mensagens emitidas, o Matlab era capaz de verificar o período de ativação de cada tarefa. Conforme representado no gráfico da Figura 39, o módulo ativa as tarefas de acordo com o sistema projetado.

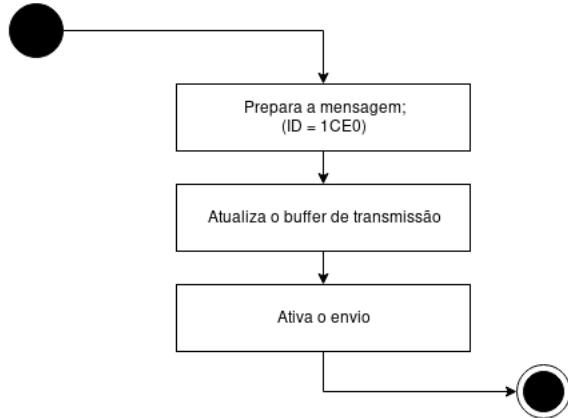
Ainda de acordo com a Figura 39, as tarefas são todas periódicas, onde cada tarefa conta com seu próprio instante de ativação. De acordo com o quadro maior, a tarefa τ_{ME2} é a primeira a ser ativada.

De acordo com os quadros menores é possível observar o período de ativação das tarefas, onde:

- Segundo o quadro τ_{ME1} : a primeira ativação ocorre próximo 1,5 s, a segunda ativação próximo de 2 s, a terceira ativação próximo de 2,5 s.
- Segundo o quadro τ_{ME2} : a primeira ativação ocorre próximo 1,3 s, a segunda ativação próximo de 1,8 s, a terceira ativação próximo de 2,3 s.

Figura 37 – Fluxograma simplificado da tarefa *task CAN*.

task_CAN



Fonte: O Autor.

- Segundo o quadro τ_{ME3} : a primeira ativação ocorre próximo 1,7 s, a segunda ativação próximo de 2,7 s, a terceira ativação próximo de 3,7 s.

Dessa forma, as tarefas τ_{ME1} e τ_{ME2} contam com períodos de 500 ms, a tarefa τ_{ME3} conta com período de 1 s.

Resultados experimentais

O módulo foi capaz de medir a tensão da bateria e o nível de combustível, os resultados das medições estão dispostas nos gráficos das Figuras 40 e 41.

Conforme representado no gráfico da Figura 40, a bateria não entrou em nível crítico. Com base no mesmo gráfico, a tensão da bateria ficou com média de 12,3 [V]. Uma vez que a eletrônica embarcada funcionou durante toda a competição, esse resultado era esperado.

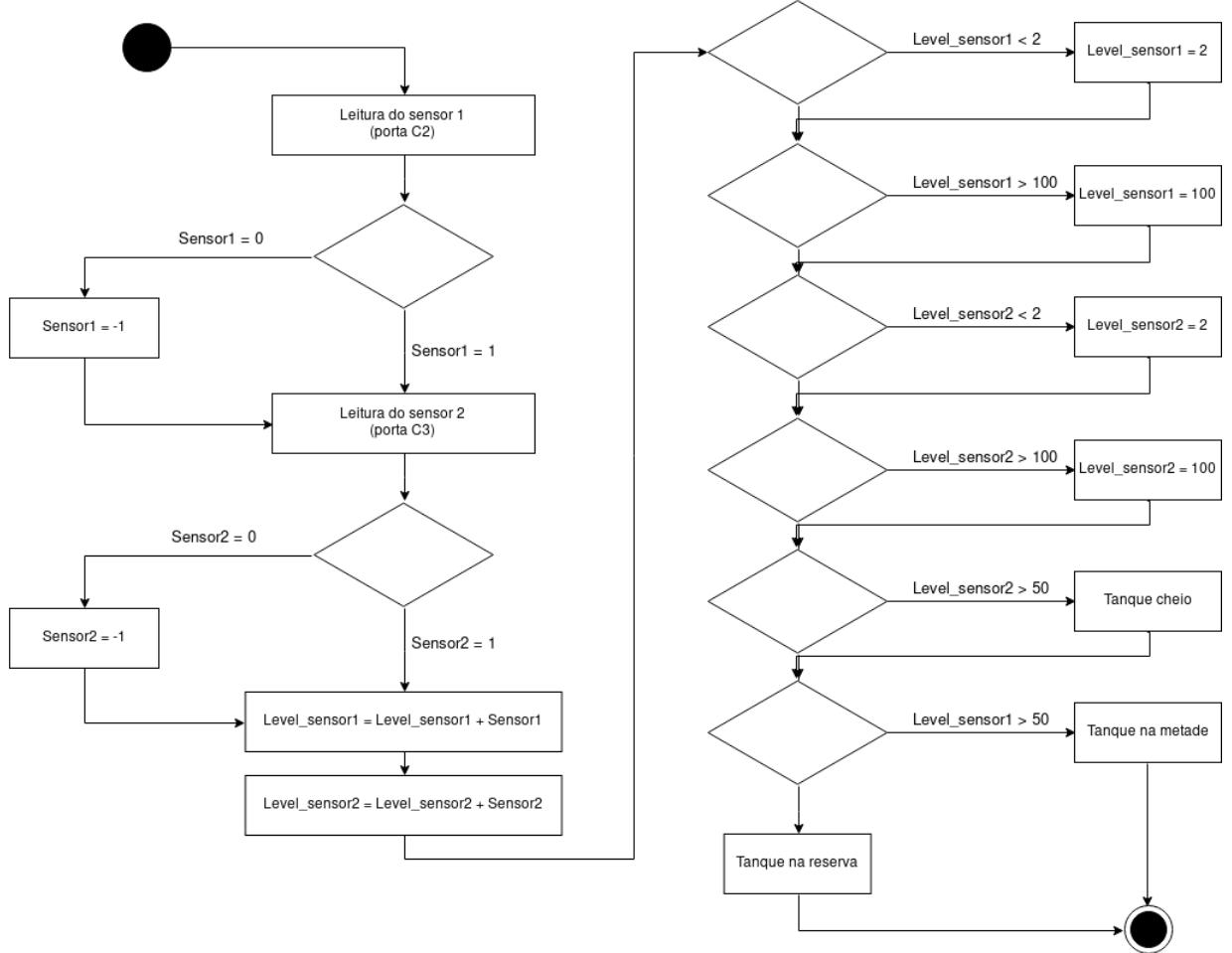
O tanque de combustível conta com 2 (dois) sensores, posicionados de forma que seja possível saber quando o reservatório está cheio, na metade ou na reserva. O tanque está cheio quando ambos os sensores estão com nível lógico 1 (um), conforme representado no instante 1(um) da Figura 41. O tanque está na metade quando o sensor superior (sensor 2) está em nível lógico 0 (zero), contudo o sensor inferior está nível lógico 1 (um). O veículo está com meio tanque de combustível no instante 3 (três) da Figura 41.

A situação 2 da Figura 41 indica que o veículo estava perto de entrar na reserva. Uma vez que o sensor superior (sensor 2) está a maior parte do tempo indicando sem combustível, e ao mesmo tempo, o estado do sensor inferior (sensor 1) está oscilando.

Após os instantes 2 da Figura 41, o veículo foi reabastecido.

Figura 38 – Fluxograma simplificado da tarefa task FUEL.

task_FUEL



Fonte: O Autor.

4.1.2.3 Conclusão

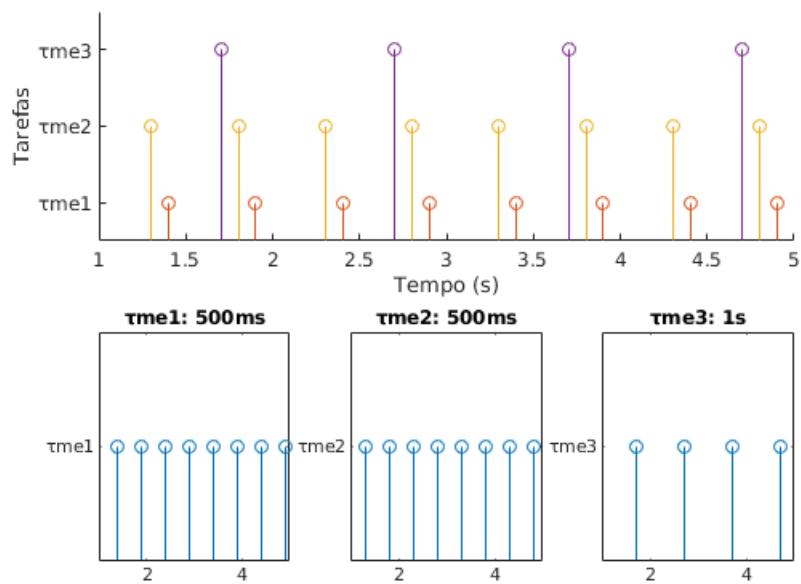
Com base nos resultados obtidos, o módulo da estrutura cumpre com os objetivos, sendo capaz de medir a carga da bateria e analisar o nível de combustível do automóvel. Todas as tarefas do módulo são escalonadas por meio de um *firmware* baseado em executivo cíclico e cumprem os seus respectivos períodos.

4.1.3 Módulo do motor

O módulo do motor é responsável pelo tacômetro, pela análise da temperatura do fluído de arrefecimento do motor, da temperatura do fluído de arrefecimento da transmissão e do tempo de funcionamento do motor (horímetro).

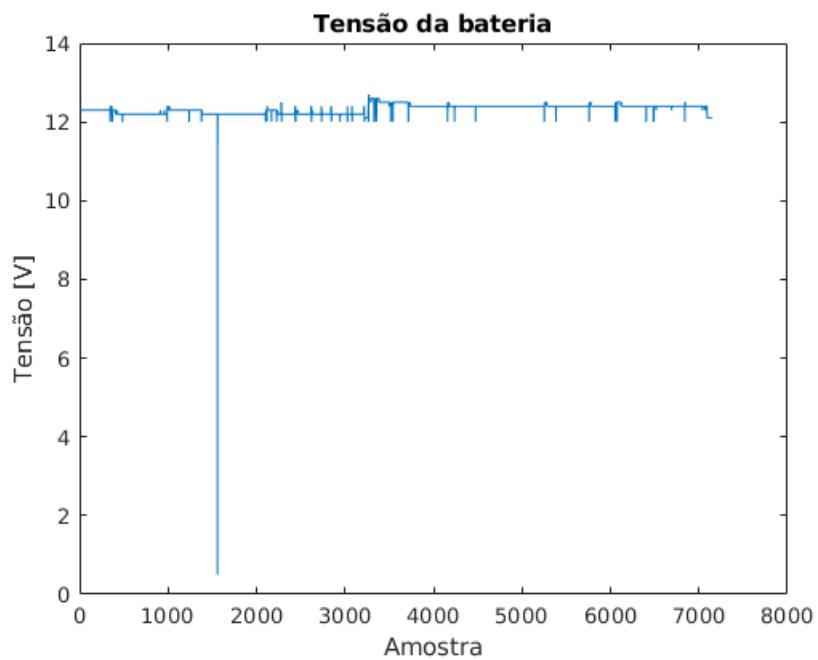
As placas PPC e PCS, Figura 42a, foram condicionadas dentro da caixa do módulo.

Figura 39 – Ativação das tarefas do módulo no decorrer do tempo.



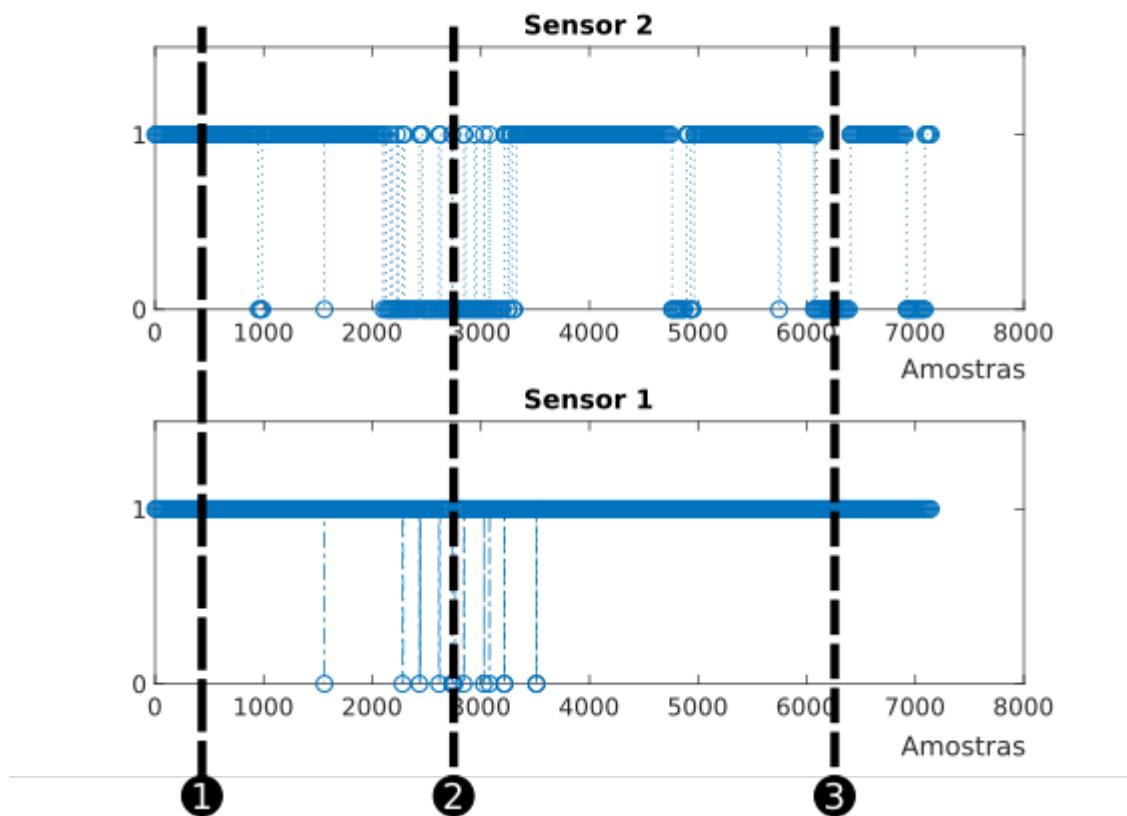
Fonte: O Autor.

Figura 40 – Tensão da bateria durante os testes do veículo.



Fonte: O Autor.

Figura 41 – Situação do tanque de combustível durante os testes, com três momentos destacados.



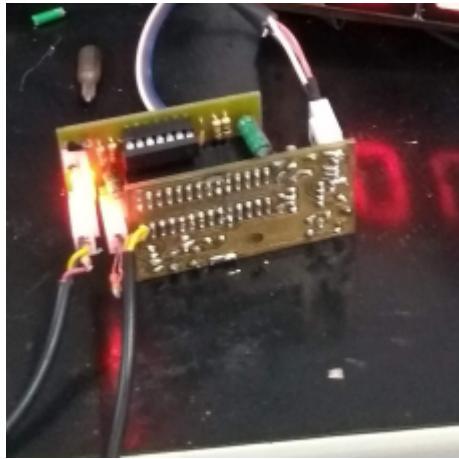
Fonte: O Autor.

O módulo, caixa com as placas, foi então instalado no automóvel, conforme representando na Figura 42b.

A instalação do módulo levou em consideração diversos aspectos mecânicos, principalmente em relação a facilidade de retirada e a proximidade com as partes quentes do veículo.

Em relação a facilidade de retirada, o módulo foi instalado de maneira que é possível remover o motor com a eletrônica junto, em caso de manutenção do motor. Além de ser possível retirar apenas o módulo eletrônico, sem a necessidade de retirar o motor, em caso de manutenção da eletrônica. Para isso, o módulo fica preso a uma proteção do motor que é removível.

Devido a proximidade com partes quentes do motor, os fios dos sensores foram protegidos com fita apropriada e o módulo foi instalado em local de boa ventilação.

Figura 42 – Montagem e instalação do módulo.(a) - Módulo funcionando
(Interior do módulo)(b) - Instalação do módulo
(Exterior do módulo)**Fonte:** O Autor.

4.1.3.1 Firmware

O algoritmo desse módulo segue a mesma estrutura do módulo da estrutura, seção 4.1.2.1, a diferença está nas tarefas que o módulo executa. O módulo do motor conta com 5 (cinco) tarefas.

A tarefa *task engine* cuida das atividades relacionada a rotação motora, a tarefa *task temp* cuida das atividades relacionadas com as temperaturas de transmissão e do motor, a *task EEPROM* cuida do horímetro e, por fim, a tarefa *task CAN* cuida do envio periódico das mensagens.

Os códigos das tarefas são definidos conforme representado nos fluxogramas das Figuras 43, 44, 45 e 46. A seguir serão apresentados os detalhes das tarefas.

A tarefa *task CAN*, conforme representado no fluxograma da Figura 43, configura o ID da mensagem, prepara os *buffers* de transmissão e depois ativa o envio. A mensagem desse módulo é formada de acordo com a Tabela 13.

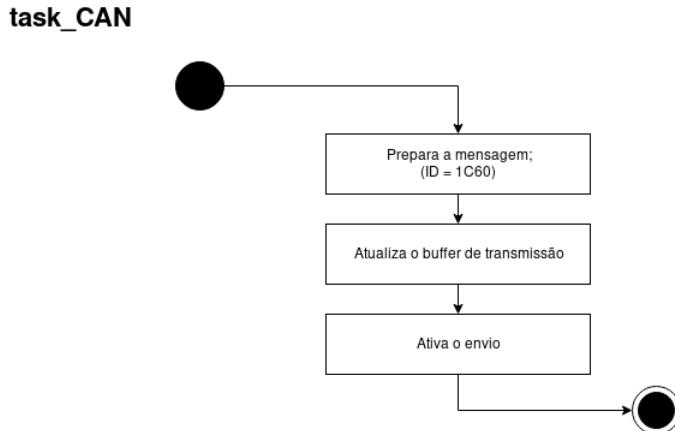
Tabela 13 – Resumo da mensagem CAN do módulo.

CAN ID	Mensagem							
	1	2	3	4	5	6	7	8
0x1C60	RPM HIGH	RPM LOW	Motor Horas	Motor Minutos	Motor Segundos	Motor Temp	CVT Temp	CAN Erro

Fonte: O Autor.

A tarefa *task engine* configura a porta do conversor A/D e depois faz duas leituras. Com as leituras, tira a média das medidas e converte o valor em tensão [V]. O valor de

Figura 43 – Fluxograma simplificado da tarefa *task CAN*.



Fonte: O Autor.

tensão é relacionado com a frequência de entrada de acordo com a Equação C.1 (Anexo C).

Para encontrar a relação entre a frequência de entrada e a rotação motora contamos com um tacômetro comercial. Antes de instalar o módulo, regulamos esse valor com o tacômetro.

O algoritmo da tarefa *task engine* está representando por meio de fluxograma na Figura 44.

O fluxograma da Figura 45 representa 2 (duas) tarefas, sendo a tarefa para medir a temperatura da transmissão e a tarefa para medir a temperatura do óleo do motor. Cada uma dessas tarefas cuida de um sensor de temperatura DS18B20. Como o sensor utilizado é o mesmo, ou seja, todo o protocolo de comunicação é idêntico. A diferença entre as tarefas está na porta utilizada, o sensor do motor utiliza a porta C3, enquanto o da transmissão a porta C2.

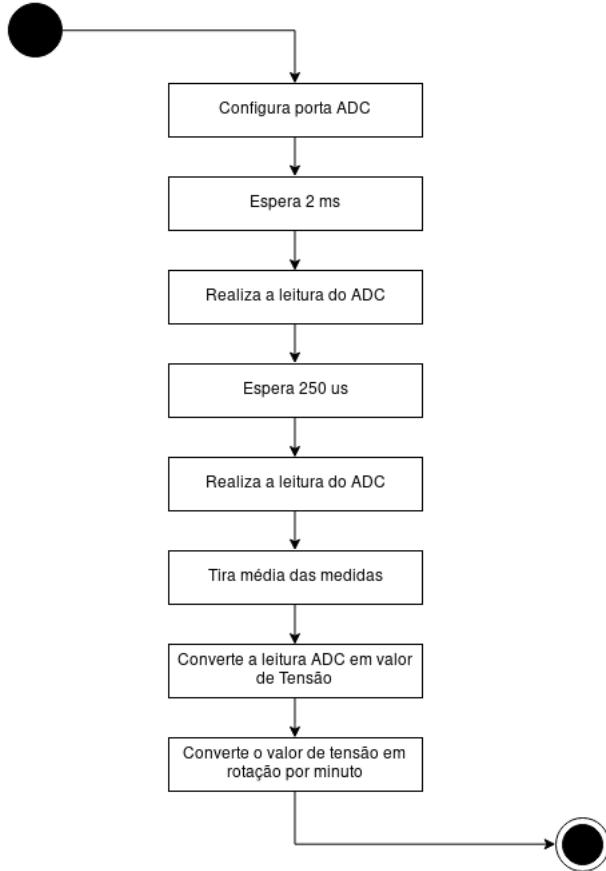
De acordo com o fluxograma da Figura 45, existem dois ramos para execução. A execução de cada ramo depende da variável Leitura, que funciona como uma *flag* de ativação. O estado inicial dessa variável é zero e pode assumir os seguintes estados:

- Leitura = 0: A primeira ativação da tarefa executa esse ramo. Durante a execução desse ramo, a *flag* é ativada.
- Leitura = 1: Durante a execução desse ramo, a *flag* é zerada.

O primeiro passo com a *flag* desativada (Leitura=0) é configurar a porta do sensor. Com a porta selecionada, a sequência de comandos para requisitar que o sensor DS18B20

Figura 44 – Fluxograma simplificado da tarefa *task Engine*.

task_ENGINE



Fonte: O Autor.

realize a leitura da temperatura é enviada. Uma vez que a leitura do sensor pode levar até 750ms, a tarefa ativa a *flag* de leitura e retorna o controle para o escalonador de tarefas.

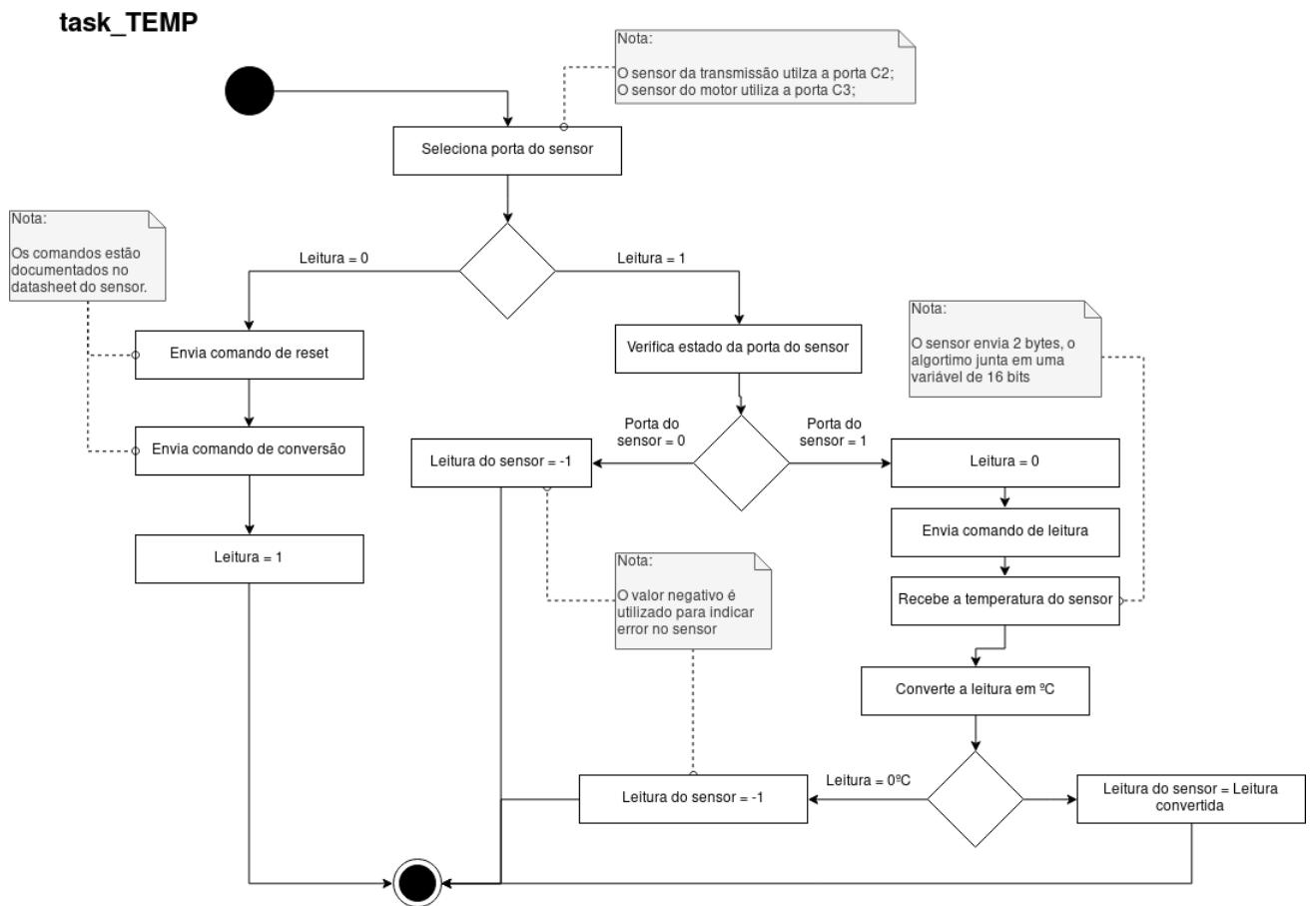
Na ativação seguinte, a tarefa entra no ramo com a *flag* ativada (Leitura=1). Nesse ramo, a primeira atividade é verificar se o sensor já terminou a leitura e a medida está disponível, caso a medida não esteja pronta a tarefa retorna o controle para o escalonador de tarefas com uma indicação de erro. Caso a porta do sensor esteja em nível alto, a *flag* de leitura é zerada e os comandos para receber os dados é enviado ao sensor.

Se a leitura do sensor for igual a zero significa que o sensor não enviou nenhum *byte*, um indicador de erro é ativado. Caso a leitura seja superior a zero, o valor é armazenado para envio.

Sabendo que a leitura é feita a cada 2 (duas) ativações e a tarefa é ativada a cada 500 ms, o intervalo de cada sensor é de 1 s.

A tarefa *task EEPROM*, representada por meio de fluxograma na Figura 46,

Figura 45 – Fluxograma simplificado da tarefa task Temp.



Fonte: O Autor.

implementa um relógio simples. A cada ativação a tarefa incrementa o valor dos segundos caso o motor esteja funcionando, e com base nesse valor atualiza as demais. Todos os valores são salvos na memória não volátil da PPC, uma memória EEPROM.

A memória EEPROM do PIC18F2580 suporta um milhão (1.000.000) ciclos de escritas, dessa forma, o horímetro tem capacidade de medir 277 horas de funcionamento, além de armazenar essa medida por mais de 40 (quarenta) anos.

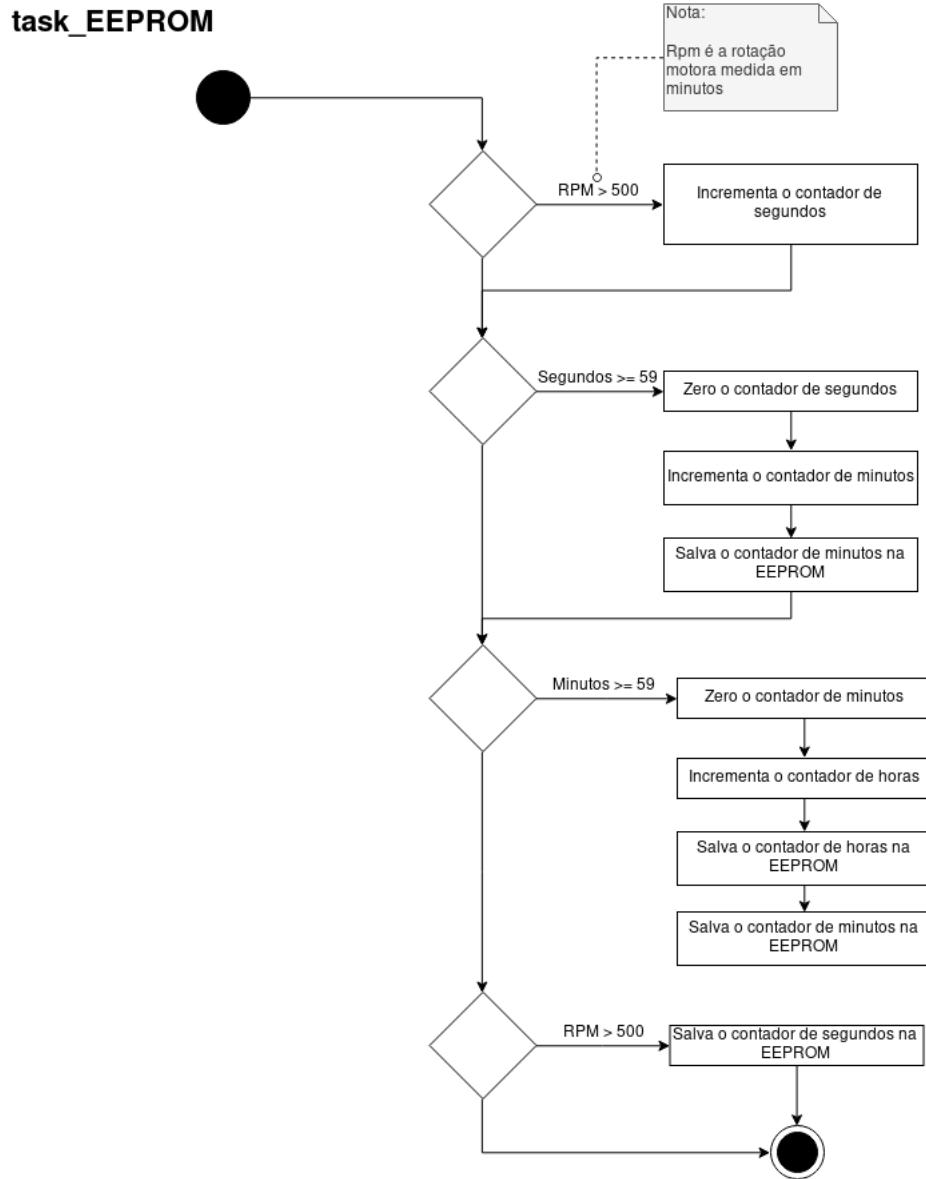
4.1.3.2 Resultados obtidos

Com o módulo montado e com o *firmware* gravado no microcontrolador, os resultados a seguir foram obtidos.

Ativação das tarefas

O procedimento para verificar os períodos das tarefas foi o mesmo do módulo da estrutura, detalhado na seção 4.1.2.2. De maneira resumida: o algoritmo desse módulo foi

Figura 46 – Fluxograma simplificado da tarefa task EEPROM.



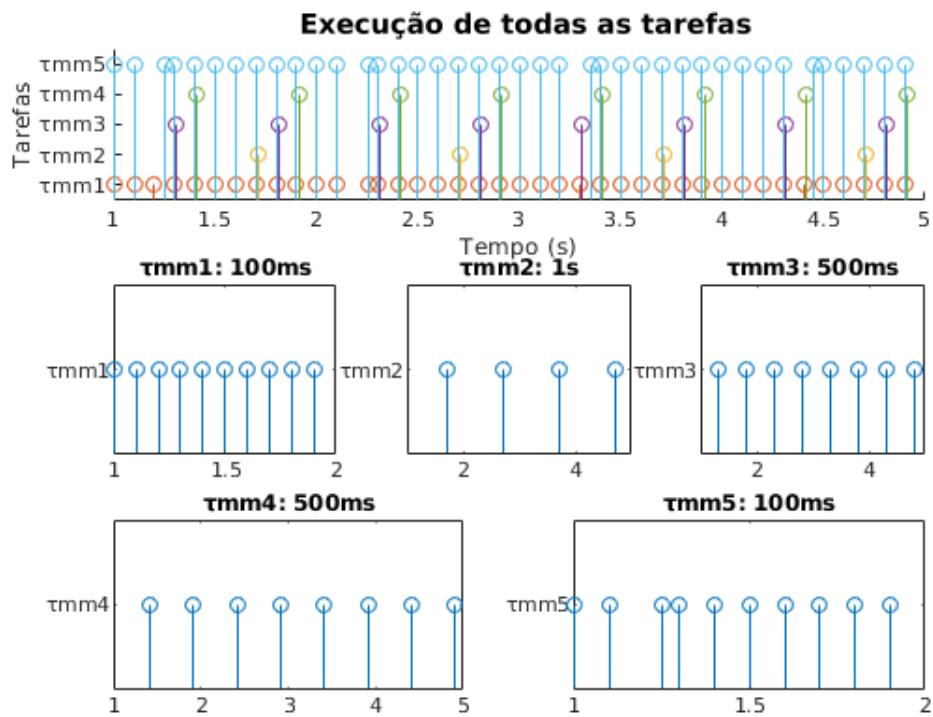
Fonte: O Autor.

alterado, adicionando um contador, e conectado ao Matlab por meio da porta serial.

A sequência de ativação das tarefas fica conforme representado no gráfico da Figura 47. Ainda de acordo com esse gráfico: as tarefas são periódicas, com os períodos em conformidade com o projeto. O quadro maior apresenta a ativação de todas as tarefas, enquanto os quadros menores apresentam os instantes de ativação por tarefa.

As tarefas τ_{MM1} e τ_{MM5} tem o período de ativação em média de 100 ms, contudo, entre o período 2 e 2,5 s (Figura 47) essas tarefas atrasaram suas respectivas ativações. A tarefa τ_{MM5} tem esse atraso bem evidente no quadro maior da Figura 47. De acordo com

Figura 47 – Ativação das tarefas do módulo no decorrer do tempo.



Fonte: O Autor.

o gráfico da Figura 48 as tarefas τ_{MM1} e τ_{MM5} tem um atraso superior ao seu período de ativação.

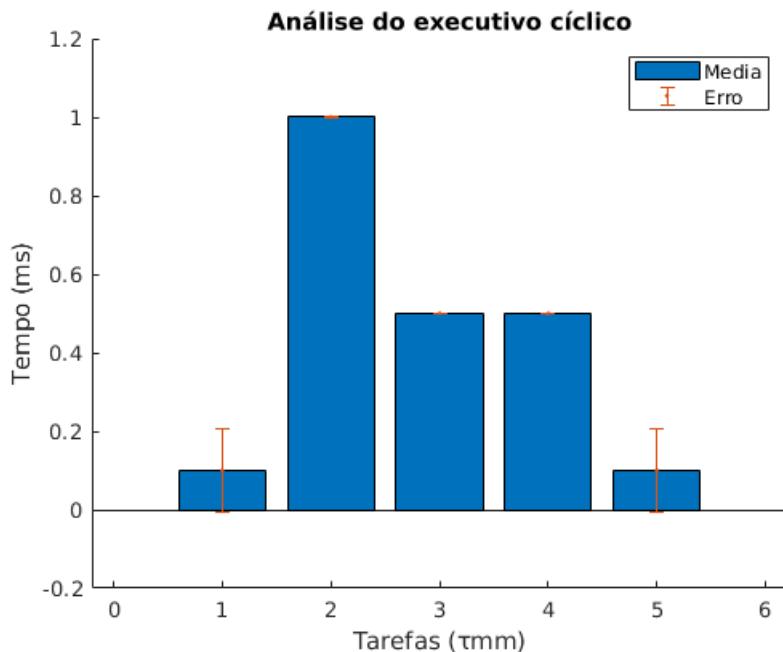
Resultados experimentais

O módulo foi capaz de medir a rotação motora por minuto, a temperatura do fluido de arrefecimento da transmissão e medir o tempo de funcionamento do motor. Devido a diversos aspectos relacionados ao regulamento da competição, o sensor de temperatura instalado no motor foi retirado.

Todos os resultados do módulo do motor estão dispostos nos gráficos da Figura 49. O funcionamento do tacômetro e da temperatura da transmissão são triviais, o gráfico apresenta as medidas diretamente.

O horímetro apresenta diversas características importantes e que serão abordadas a seguir.

Todas essas informações foram salvas pelo módulo da telemetria. Esse módulo, da telemetria, conta com um relógio com período de 0,8 μ s. Sempre que uma mensagem válida chega ao módulo da telemetria, além de salvar os dados da mensagem, o módulo salva o instante que essa mensagem chegou. A base de tempo da Figura 50 é baseado nesse

Figura 48 – Análise do atraso das tarefas.

Fonte: O Autor.

relógio.

Diferente do horímetro, essa base de tempo sempre é zerada quando a eletrônica é desligada.

Para análise do horímetro, considere inicialmente que o contador de segundos está zerado, conforme representado no instante 1 da Figura 50. Sabendo que o motor está ligado, o contador de segundos é incrementado até atingir 60 s^1 , conforme representado no instante 2. Após esse instante (2), o contador de segundos é zerado, contabilizando 1 minuto. Por fim, no instante 3 o motor está desligado e o contador de segundos fica constante.

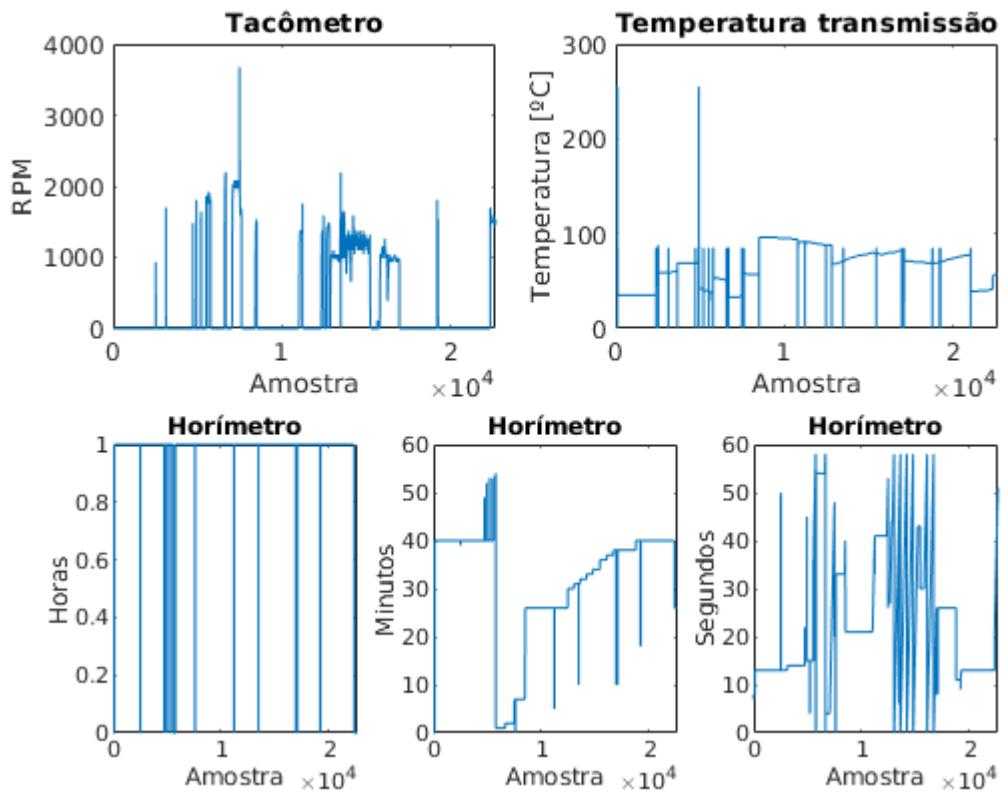
Utilizando a base de tempo disponibilizada pelo módulo da telemetria, a diferença de tempo entre instante 1 e o instante 2 é igual à 58,88 segundos, ou seja, um erro de 1,12 s.

O contador de minutos é incrementado à medida que o contador de segundos completa 60 ciclos². Situação representada na Figura 51, entre os instantes 1 e 2.

Ainda com relação a Figura 51, o contador de segundos apresenta uma quebra do ciclo. De acordo com o gráfico, o horímetro incrementou o contador de minuto antes de

¹ O contador inicia em zero e conta até 59, totalizando 60 segundos.

² O contador inicia em zero e conta até 59, totalizando 60 ciclos.

Figura 49 – Resultados do módulo do motor.

Fonte: O Autor.

completar 60 segundos, indicando a ocorrência de um erro.

4.1.3.3 Conclusão

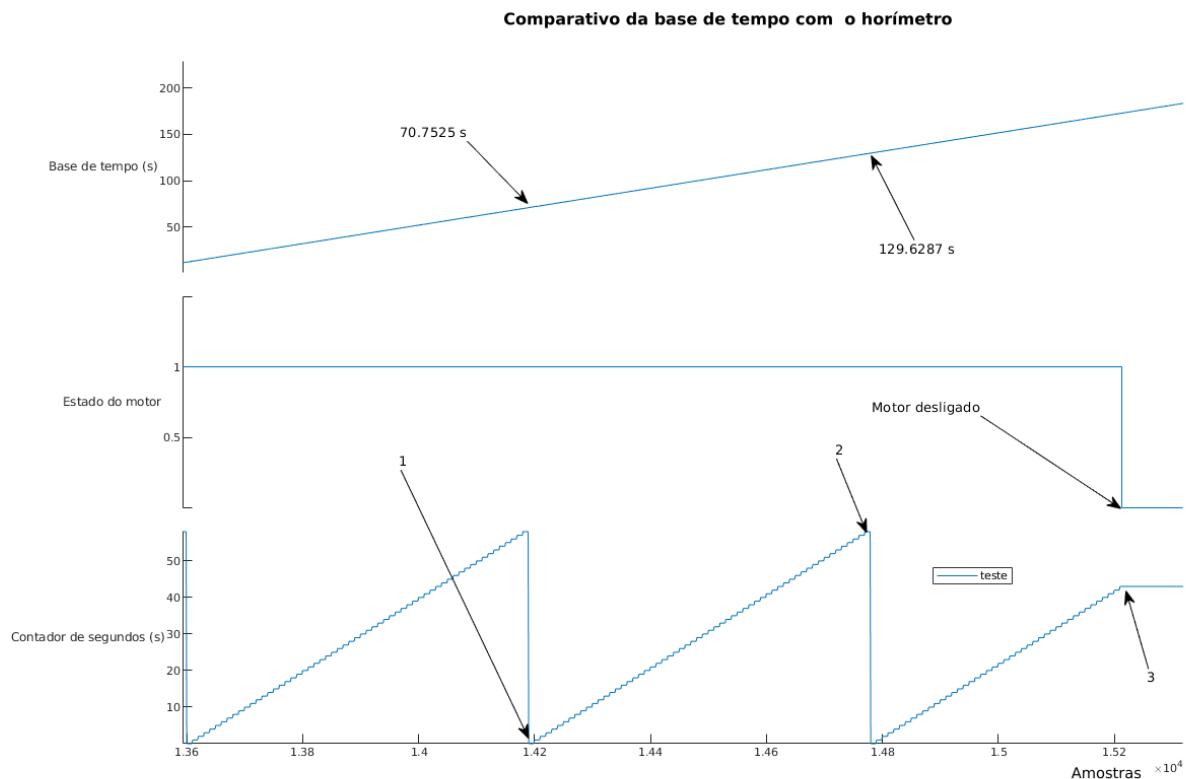
Embora o módulo do motor atenda aos objetivos propostos, sendo capaz de medir a rotação motora, a temperatura da transmissão e o tempo de funcionamento, o módulo precisa de melhorias.

Por se tratar de um sistema de tempo real não crítico, a perca do *deadline* das tarefas não ocasiona nenhum risco ao funcionamento do sistema.

O sistema para aquisição da rotação motora, descrito no Anexo C, precisa ser aperfeiçoado, principalmente por necessitar de atualização de *firmware* para alterar o fator da relação entre frequência de entrada e a rotação. Uma potencial solução seria a inclusão de um acoplador ótico, com intuito de proteger a PCS dos sinais provenientes do motor, além de padronizar o sinal de saída.

Esse sistema, para aquisição da rotação motora, quando exposto aos diversos tipos de situações na competição apresentou pouca confiabilidade. Como resultado, o motor

Figura 50 – Análise do horímetro em relação ao funcionamento do motor e do tempo.



Fonte: O Autor.

pode funcionar sem que o tempo seja contabilizado, ou seja, o horímetro passa a indicar o tempo incorreto mesmo que a sua tarefa esteja funcionando corretamente.

O sistema para aquisição de temperatura da transmissão funcionou corretamente durante toda a competição. No decorrer de uma prova, o módulo detectou o superaquecimento da transmissão e prontamente avisou ao motorista.

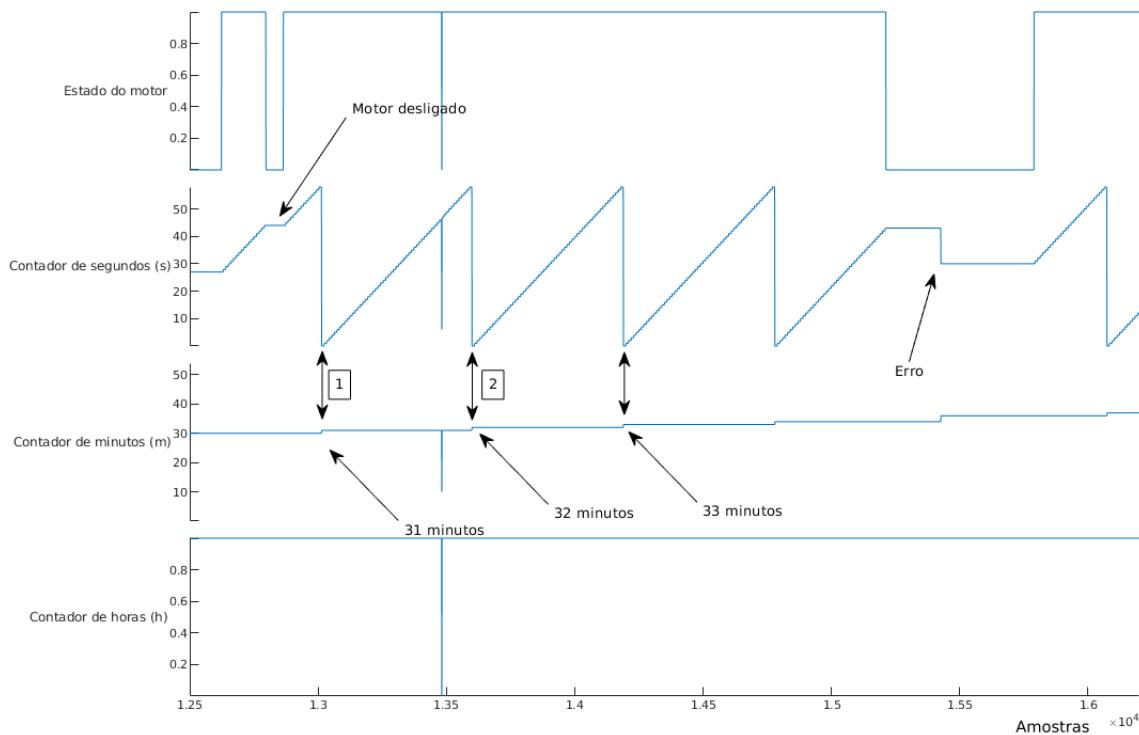
Embora nenhum resultado tenha sido obtido com o sensor de temperatura do motor durante a competição, o módulo está pronto para realizar essa atividade.

4.1.4 Módulo de localização

O módulo de localização é responsável por detectar a velocidade do veículo e analisar a situação do fluido de freio. Além disso, o módulo conta com dispositivos para proteção da eletrônica.

O módulo foi instalado na frente do automóvel, logo abaixo do capô. O conjunto, PCS e PPC, foi condicionado dentro do módulo, de acordo com a Figura 52a. O módulo foi fixado próximo da bateria, conforme representado na Figura 52b.

Figura 51 – Análise do horímetro em relação aos contadores de minutos e horas.



Fonte: O Autor.

Para proteção da eletrônica, adotou-se o uso de fusíveis de vidro. O porta fusível e o fusível utilizados estão destacados na fotografia da Figura 52c.

Por último, o sensor foi fixado na roda dianteira com a face sensora voltada para o disco de freio, conforme pode ser identificado na Figura 52d.

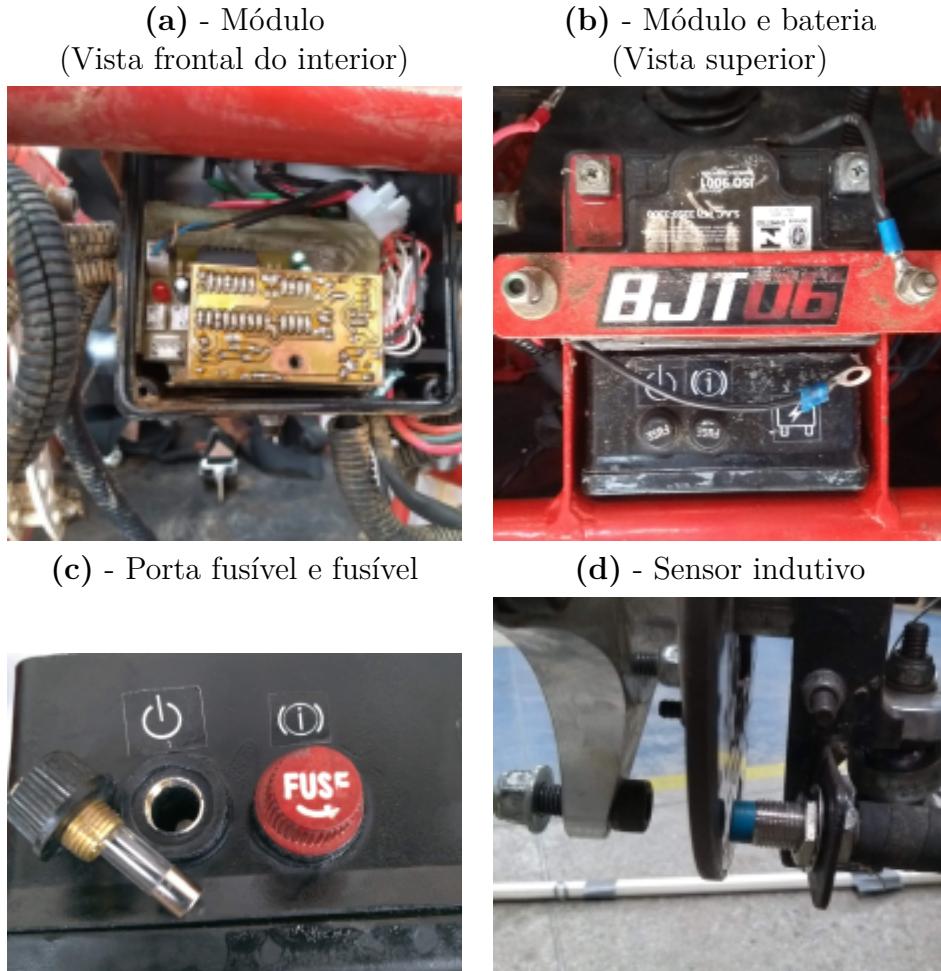
4.1.4.1 Firmware

O módulo de localização conta com 3 (três) tarefas e são escalonadas de acordo com os períodos de ativação. Os detalhes do escalonamento estão descritos na seção 4.1.2.1, módulo da estrutura.

Os códigos das tarefas são definidos conforme representado nos fluxogramas das Figuras 53, 54, 56. Os detalhes das tarefas serão descritos a seguir.

A tarefa *task CAN*, representada no fluxograma da Figura 53, configura o ID da mensagem, prepara os *buffers* de transmissão e depois ativa o envio. A Tabela 14 dispõe de todos os campos da mensagem CAN desse módulo.

A tarefa *task Speed*, exposta no fluxograma da Figura 54, realiza 10 (dez) leituras

Figura 52 – Montagem e instalação do módulo.**Fonte:** O Autor.**Tabela 14** – Resumo da mensagem CAN do módulo.

CAN ID	Mensagem							
	1	2	3	4	5	6	7	8
0x1C20	Velocidade (Km/H)	Estado do fluído de freio	CAN Erro					

Fonte: O Autor.

do conversor A/D com intervalos de 50 µs. Após as leituras, a tarefa tira uma média de todas as leituras e realiza uma série de conversões, descritas a seguir.

A média das leituras, feitas pelo conversor, é dada pela Equação 4.1.

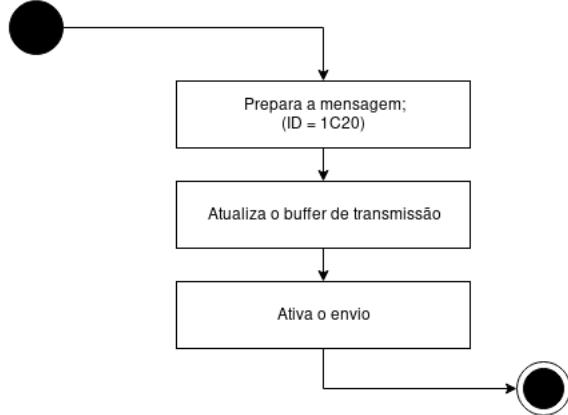
$$adc0 = adc0 / 10 \quad (4.1)$$

sendo,

$adc0$: Leituras do ADC (AN0).

Figura 53 – Fluxograma simplificado da tarefa *task CAN*.

task_CAN



Fonte: O Autor.

Em seguida, converte-se o valor da média em tensão, segundo a Equação 4.2.

$$voltage = adc0 * (0.00488758553) \quad (4.2)$$

onde,

voltage : Valor de tensão da porta AN0 [V].

A terceira etapa, converte-se o valor de tensão em frequência de acordo com a Equação 4.3.

$$frequency = voltage / (0.00756) \quad (4.3)$$

em que,

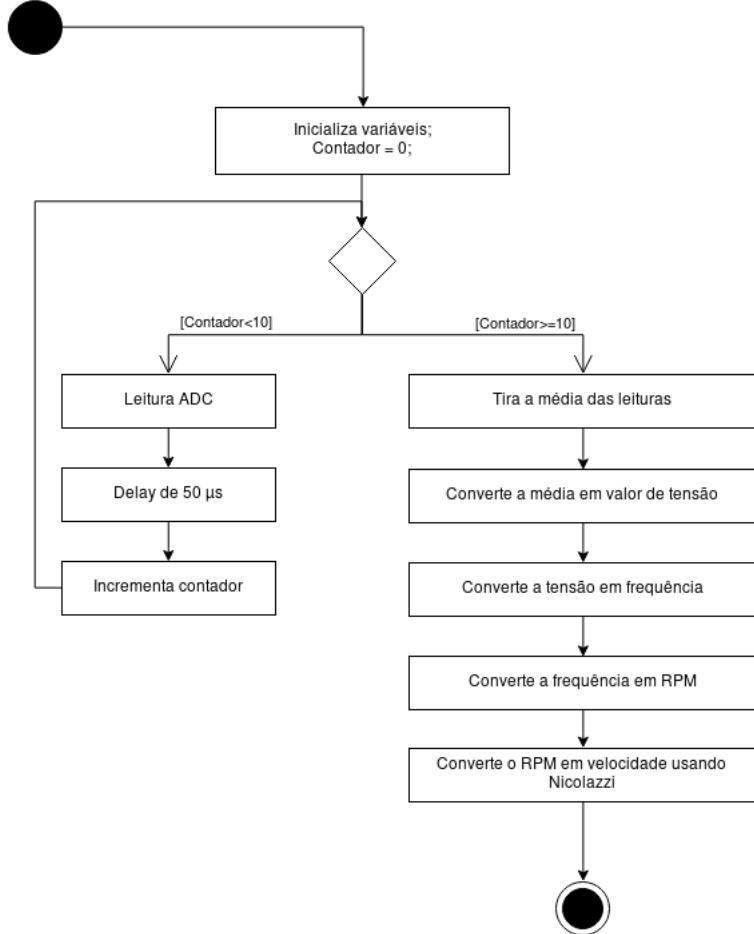
frequency : Valor da fréquencia do sinal do sensor [Hz].

Logo após, obtém-se o valor da velocidade angular, dada em rotações por minuto, por meio da frequência, usando a Equação 4.4. Essa Equação (4.4) considera um disco com 4 (quatro) furos.

$$speed_rpm = frequency * (60/4) \quad (4.4)$$

considere então,

speed_rpm : Valor da velocidade angular da roda [RPM]

Figura 54 – Fluxograma simplificado da tarefa task speed.**task_SPEED****Fonte:** O Autor.

A velocidade, dada em quilômetro por hora, é obtida através da Equação 4.5.

$$vt_nicolazzi = rd * wr * 3,6 \quad (4.5)$$

visto que,

$$\begin{aligned}
 wr &= 0,104719 * speed_rpm; \\
 re &= 0,47 * 0,0254 * 23; \\
 rd &= 1,02 * re.
 \end{aligned}$$

A tarefa *task brake* verifica o estado do sensor, Figura 55, por meio da porta digital C1. Baseado no valor lido do sensor, a tarefa incrementa ou decrementa uma variável de estado. Essa variável, de estado do sensor, tem por finalidade evitar que durante o movimento do fluido dentro do reservatório um alarme falso seja disparado.

Figura 55 – Sensor de nível crítico do fluído de freio.



Fonte: O Autor.

A variável de estado tem um intervalo de operação, assim, o tempo máximo para detectar a ocorrência do problema é conhecido. O módulo leva 2,5 s para detectar a mudança de estado do sensor.

4.1.4.2 Dimensionamento elétrico

Para a escolha do fusível, levantou-se o consumo elétrico do sistema no pior caso. Os dados estão agrupados na Tabela 15. Uma vez que o fusível protege todos os módulos, o consumo será dado por $\sum(\text{Consumo por módulo})$. Logo, o consumo do sistema é de $20e - 3 + 404e - 3 + 206e - 3 + 9e - 3 + 304e - 3 \approx 1 A$, sendo assim, o menor valor para o fusível será de 1 A.

Além do fusível, o barramento do veículo também deve suportar o consumo do sistema elétrico. No modelo utilizado para testes, o barramento é formado majoritariamente por cabo AFD de 4 vias, que suporta tensão de até 300 V. O sensor de freio e a conexão bateria com os fusíveis e o barramento, ambos utilizam cabos Flexicom que suportam tensões de 450 V.

4.1.4.3 Resultados obtidos

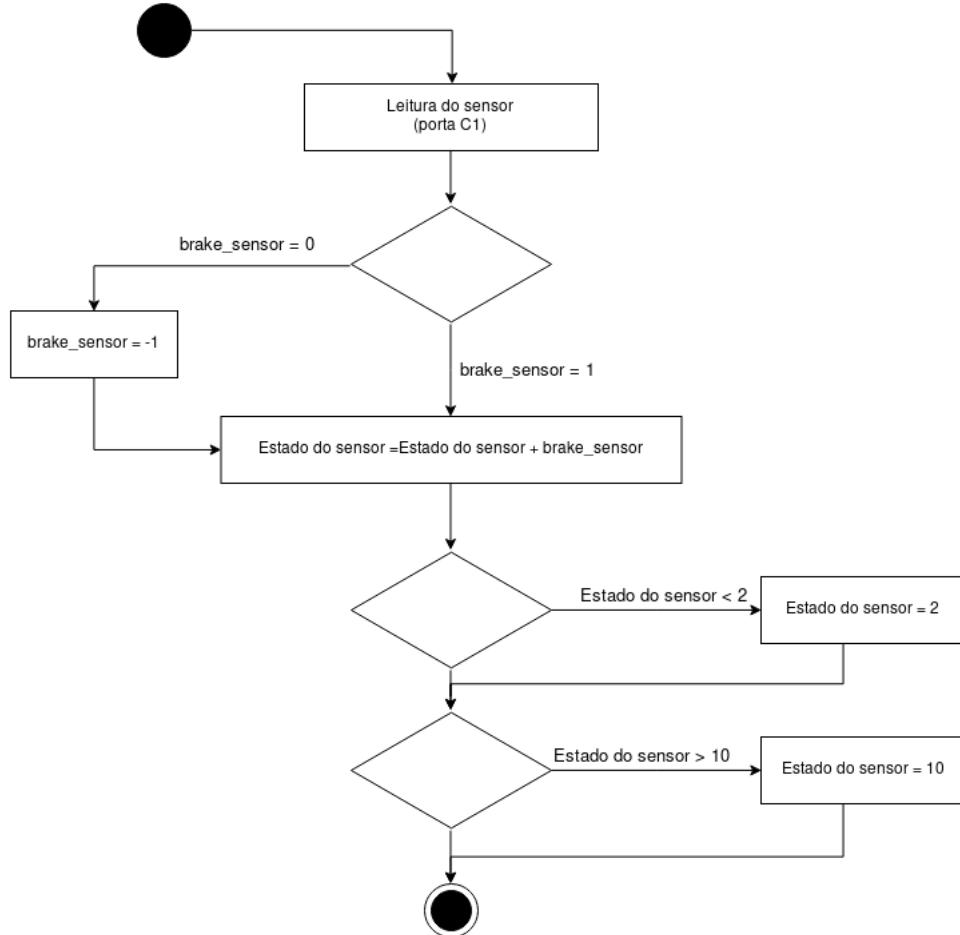
Com o módulo montado e com o *firmware* gravado no microcontrolador, os resultados a seguir foram obtidos.

Ativação das tarefas

Conforme descrito na seção 4.1.2.2, os períodos das tarefas foram mensurados por meio do TIMER1 da própria PPC. Um *firmware* específico foi utilizado, com intuito de

Figura 56 – Fluxograma simplificado da tarefa *task brake*.

task_BRAKE



Fonte: O Autor.

acoplar o módulo ao Matlab. Com base nessa estrutura, os dados a seguir foram obtidos.

A sequência de ativação das tarefas fica conforme representado nos gráficos da Figura 57. Em relação aos gráficos, nota-se que as tarefas são periódicas com períodos conforme projetado.

Resultados experimentais

O módulo foi capaz de mensurar a velocidade do automóvel durante todo o período de avaliação, além de analisar o nível do fluido de freio. A massa de dados do veículo foi filtrada, retirando os valores incompatíveis.

Em relação ao velocímetro, foram removidos os picos de velocidades. Esses picos de velocidade ocorrem ao energizar a eletrônica. Como se trata de valores pontuais, a próxima medida, 100 ms depois, atualiza o Painel com a informação correta e o motorista

Tabela 15 – Consumo elétrico no pior caso.

Módulo	Item	Consumo	Observação
Painel		20 mA	
	Placa base - PPC	4 mA	PIC18F+MCP+LED
	Acoplamento painel - PCS	16 mA	LEDS+DISPLAYs
Estrutura		404 mA	
	Placa base - PPC	4 mA	
	Sensor capacitivo 1	200 mA	Datasheet - Modelo: CR18-8DP
	Sensor capacitivo 2	200 mA	Datasheet - Modelo: CR18-8DP
Localização		206 mA	
	Placa base - PPC	4 mA	
	Acoplamento - PCS	2 mA	
	Sensor indutivo	200 mA	Datasheet - Modelo: Npn Lj12a3-4-z
Motor		9 mA	
	Placa base	4 mA	
	Acoplamento	3 mA	
	Sensor Temperatura 1	1 mA	Datasheet - Modelo: DS18B20
	Sensor Temperatura 2	1 mA	Datasheet - Modelo: DS18B20
Telemetria/ Diagnóstico		304 mA	
	Placa base - PPC	4 mA	
	Transmissor	300 mA	Datasheet - Modelo: ESP8266:01

Fonte: O Autor.

não percebe, contudo, a medida é armazenada.

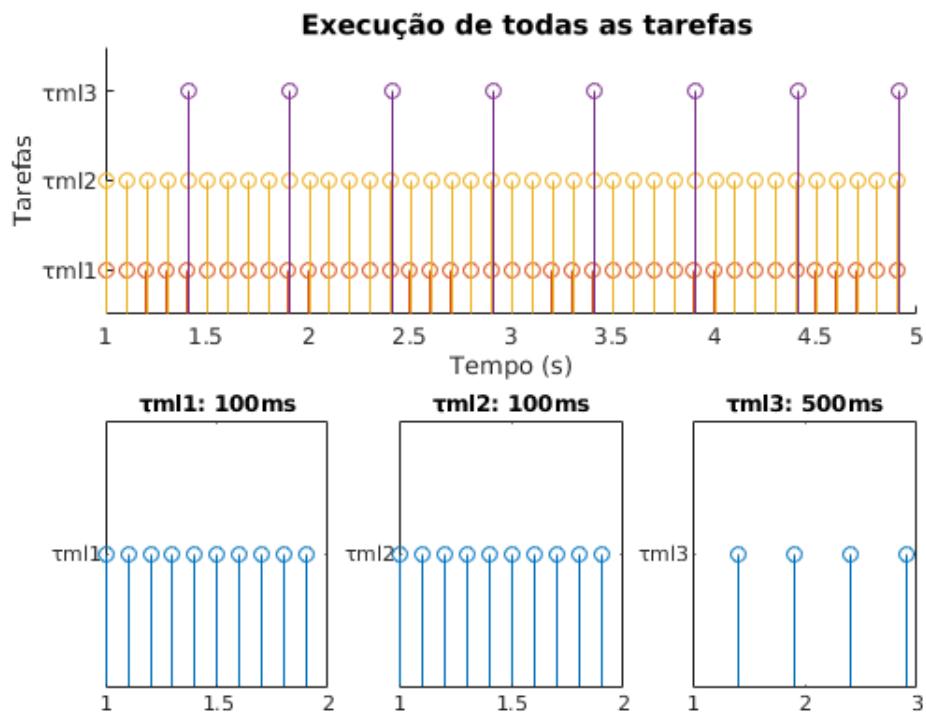
Uma maneira de reduzir esses picos de velocidades é por meio de filtro no módulo da telemetria. Outra forma, mais indicada, seria adoção de um atraso na inicialização do módulo de localização. Ambas são viáveis de serem implementadas, necessitando apenas de modificação do *firmware*.

Uma solução via *hardware* implicaria na construção de uma nova placa de condicionamento, o que torna inviável.

Conforme destacado no gráfico de velocidade máxima da Figura 58, o veículo alcançou 35 km h^{-1} . A critério de comparação, segundo o resultado da prova de velocidade da mesma competição, 25^a Competição Baja SAE BRASIL, o veículo atingiu 33.58 km h^{-1} em 100 m de pista.

Quanto ao nível do fluído de freio, os dados foram filtrados levando em consideração que o sensor só pode assumir dois estados. Os dados sem tratamento apresentaram dois valores incompatíveis, conforme representado no gráfico dados brutos da Figura 59.

Figura 57 – Ativação das tarefas do módulo no decorrer do tempo.



Fonte: O Autor.

De acordo com o gráfico dos dados filtrados, Figura 59, o reservatório ficou a maior parte do tempo com o nível adequado ao funcionamento, entretanto, o reservatório entrou em nível crítico 2 (duas) vezes, voltando ao estado normal logo após.

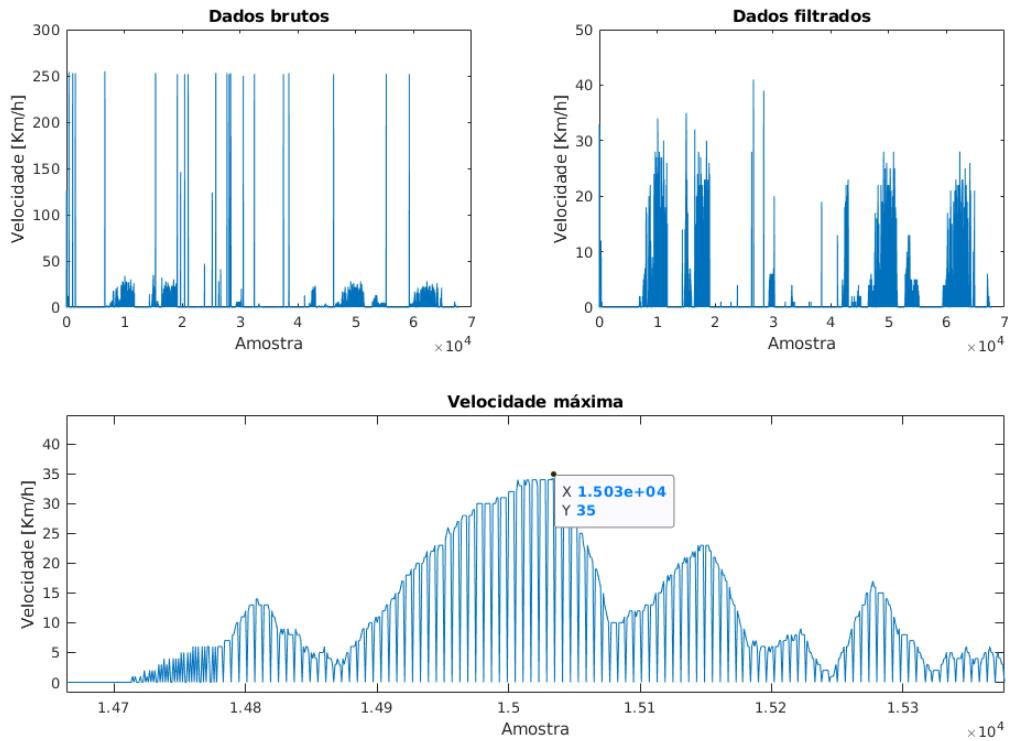
4.1.4.4 Conclusão

O módulo cumpre com os objetivos, sendo capaz de medir a velocidade e analisar o reservatório do fluido de freio. Os resultados de ambos, sensor de fluido e velocímetro, estão coerentes com os testes.

Todas as tarefas seguem o projetado, sendo periódicas e com os períodos de ativações de acordo.

4.1.5 Módulo do painel

O painel de instrumento é o encarregado de exibir as informações do veículo ao motorista. O módulo foi instalado atrás do volante, conforme projetado, sendo de fácil acesso.

Figura 58 – Velocidade do automóvel durante os testes.

Fonte: O Autor.

4.1.5.1 Firmware

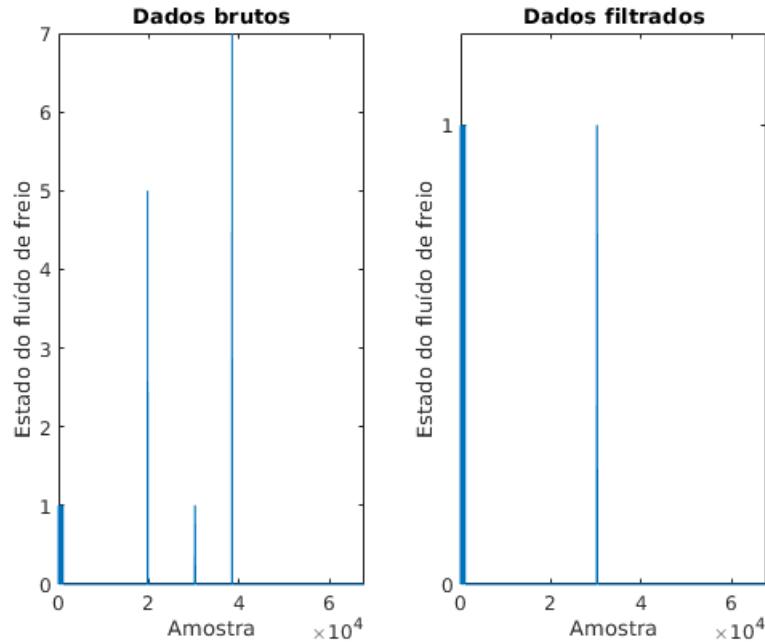
O painel de instrumentos conta com 5 (cinco) tarefas e são escalonadas de acordo com períodos de ativação. O escalonador de tarefas funciona da mesma maneira que o do módulo da estrutura, descrito na seção 4.1.2.1.

Esse módulo não utiliza nenhum sensor, todas as informações são recebidas via rede veicular. O *firmware* conta com uma função, ativada por meio de interrupção, que atualiza as variáveis de ambiente com as informações mais recentes. Conforme representado no diagrama da Figura 60, o módulo conta com 4 (quatro) filtros ativados.

Cada módulo conta com um ID de identificação, que caracteriza a mensagem. A estrutura da mensagem, e o CAN-ID, de cada módulo estão descritos nas Tabelas 12, 13 e 14. Por meio dessa identificação, CAN ID, é possível identificar a composição da mensagem e assim atualizar as variáveis do ambiente de execução.

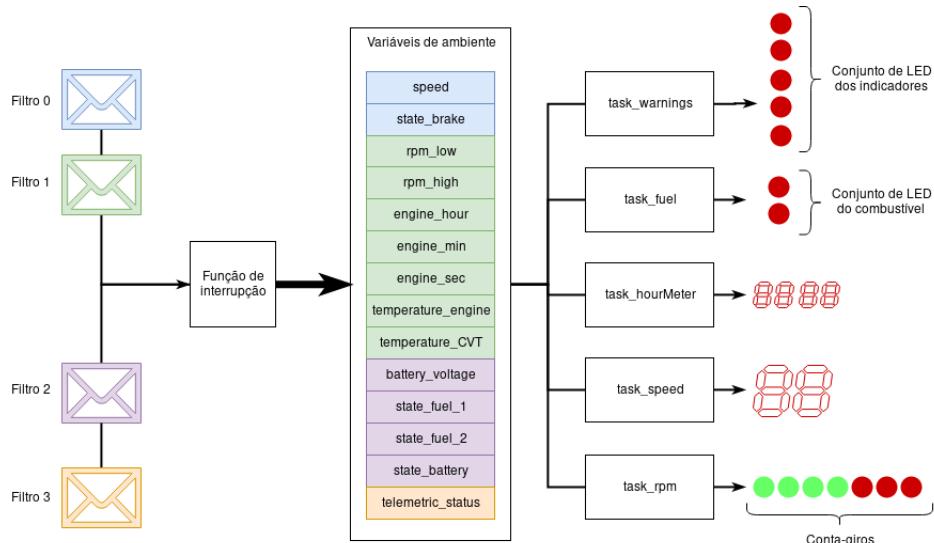
Cada tarefa fica então responsável por atualizar uma parte do painel de instrumentos com base nas variáveis de ambiente, conforme representado na Figura 60. Os códigos das tarefas são definidos de acordo com os fluxogramas das Figuras 61, 62, 63, 64 e 65. A

Figura 59 – Estado do reservatório (freios) durante testes.



Fonte: O Autor.

Figura 60 – Fluxo de dados simplificado do painel de instrumentos.



Fonte: O Autor.

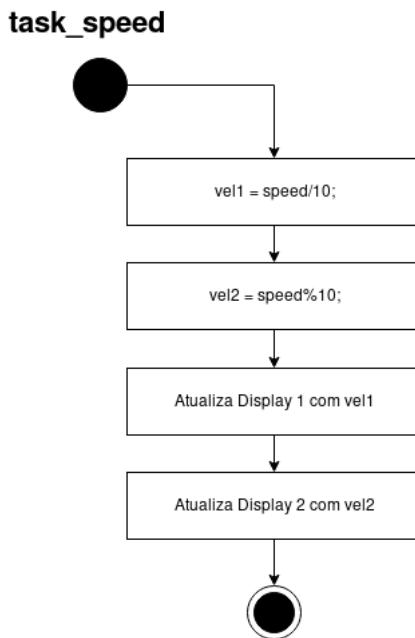
seguir serão tratados os detalhes das tarefas.

A tarefa *task speed* apenas separa a velocidade em duas casas, dezenas e unidades. Envia para o primeiro *display* o valor da dezena, e para o segundo *display* o valor da

unidade.

O fluxograma simplificado da tarefa *task speed* está representado na Figura 61. A variável *speed*, utilizada na tarefa, pertence ao ambiente de execução, conforme a Figura 60.

Figura 61 – Fluxograma simplificado da tarefa *task speed*.



Fonte: O Autor.

A tarefa *task fuel* compara os estados dos sensores, caso ambos os sensores estiverem ativos, indica que o reservatório está cheio; Caso apenas o sensor 1 estiver ativo, significa que o reservatório está na metade; por fim, se nenhum sensor estiver ativo, o tanque está na reserva. Para cada nível de combustível o painel de instrumento ativa os LEDs de acordo.

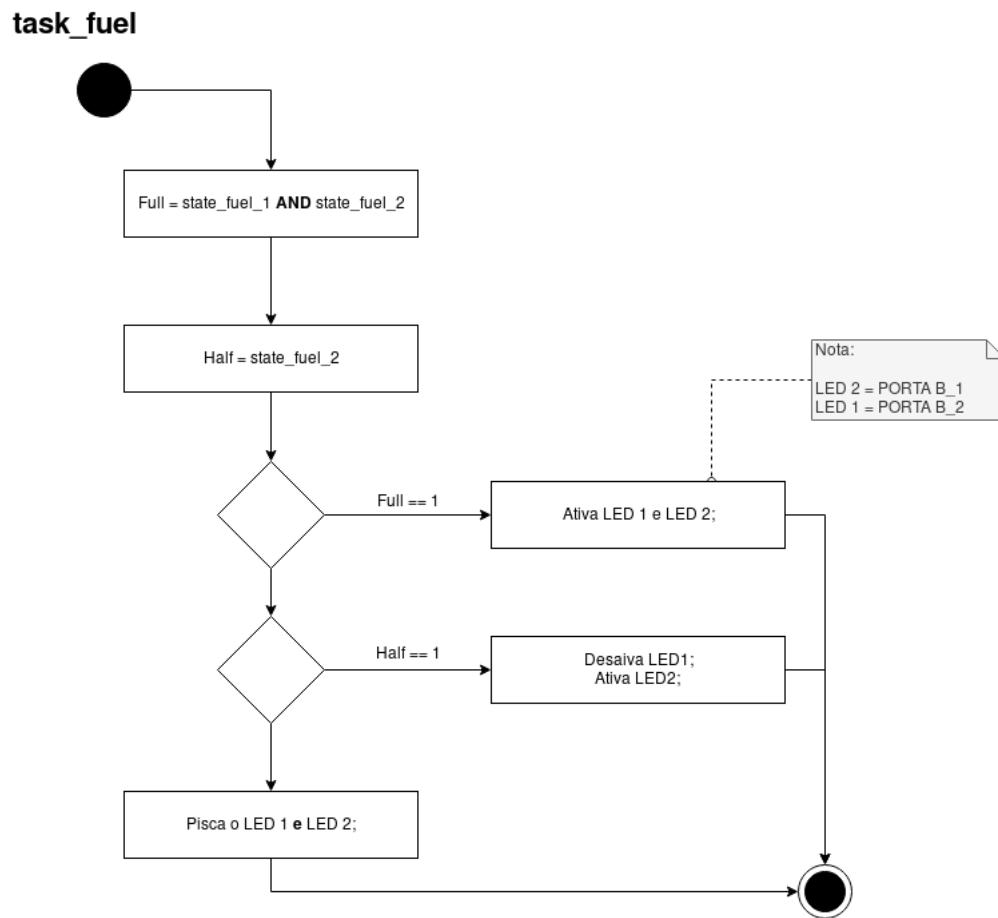
A tarefa *task fuel* está representada por meio de fluxograma na Figura 62.

O sistema de advertência e diagnóstico está implementado na tarefa *task warnings*, funciona de acordo com o fluxograma da Figura 63. Cada módulo conta com os seus parâmetros de testes. A seguir serão detalhados os alertas e os parâmetros escolhidos.

O indicador da telemetria, primeiro LED do painel de cima para baixo, pode assumir os seguintes estados:

- LED Ativo: Significa que o módulo da telemetria não inicializou, ou não está conectado;
- LED Desativado: Significa que o módulo está conectado e inicializou o funcionamento;

Figura 62 – Fluxograma simplificado da tarefa *task fuel*.



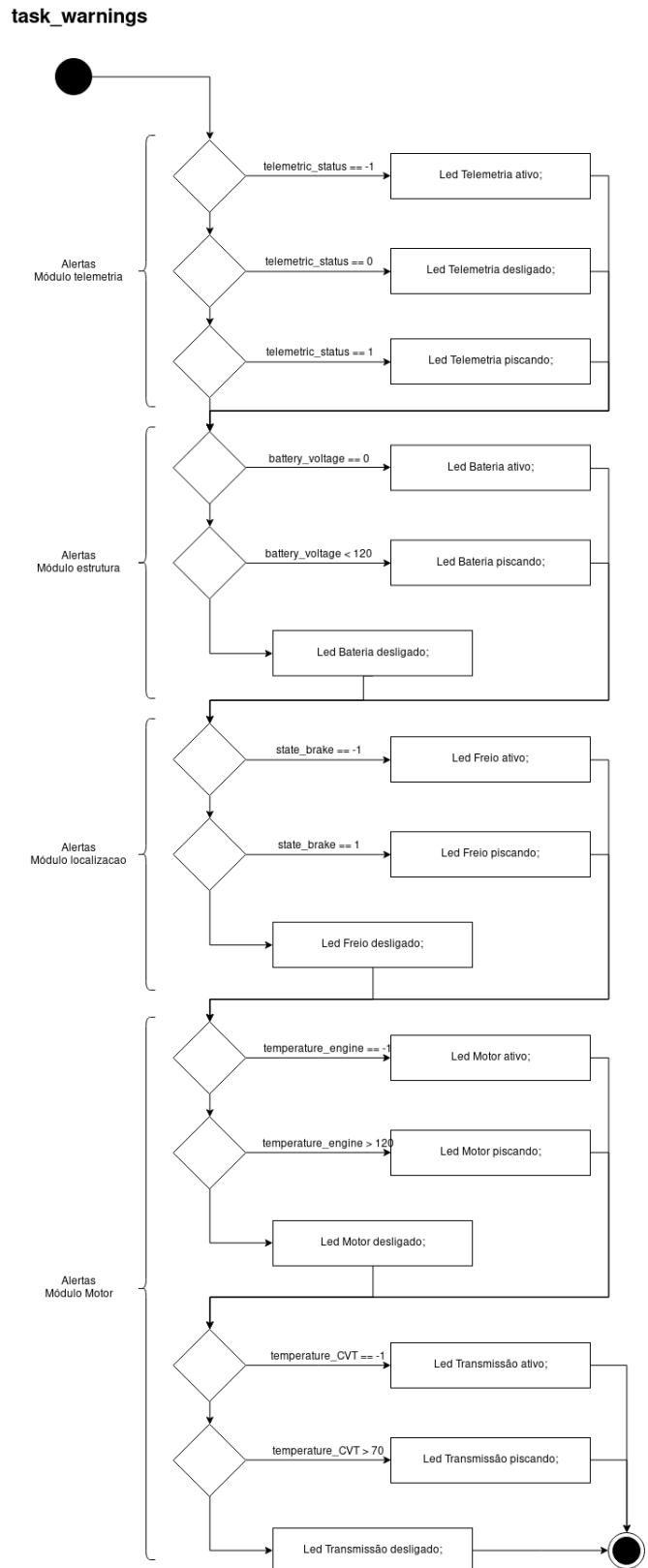
Fonte: O Autor.

- LED Piscando: Significa que o módulo da telemetria está com problemas. (Por Ex: sem cartão de memória, ou cartão solto).

O indicador da bateria, segundo indicador de cima para baixo, é baseado na carga da bateria (*battery voltage*), mensurada pelo módulo da estrutura. Esse indicador pode assumir três estados, sendo:

- LED Ativo: Significa que o módulo da estrutura não inicializou, ou não está conectado;
- LED Desativado: Significa que o módulo da estrutura está conectado, funcionando e a bateria está carregada;
- LED Piscando: Significa que a bateria está no nível crítico de carga.

O terceiro indicador, de cima para baixo, é o do fluído de freio. Esse indicador é baseado no módulo de localização, e pode assumir três modos, sendo:

Figura 63 – Fluxograma simplificado da tarefa *task warnings*.**Fonte:** O Autor.

- LED Ativo: Significa que o módulo não inicializou, ou não está conectado;
- LED Desativado: Significa que o módulo está conectado, funcionando e o reservatório está no nível adequado;
- LED Piscando: Significa que o reservatório está no nível crítico.

O quarto e quinto indicador, de cima para baixo, são para temperatura alta. Ambos os indicadores são baseados no módulo do motor. Esses indicadores podem assumir os seguintes estados:

- Apenas LED Temp. Motor Ativo: Significa que o sensor de temperatura do motor está com problemas, ou desconectado;
- Apenas LED Temp. Transmissão Ativo: Significa que o sensor de temperatura da transmissão está com problemas, ou desconectado;
- LEDs Ativos (ambos): Significa que o módulo está com problemas, ou ambos os sensores estão desconectados;
- LEDs Desativado (ambos): Significa que o módulo está conectado, funcionando e os sensores estão funcionando;
- LED Piscando: Significa que a temperatura do respectivo indicador está acima do permitido, onde:
 - Temperatura do motor acima de 120 °C;
 - Temperatura da transmissão acima de 70 °C.

Uma vez que o conta-giros do painel é composto por 7 (sete) LEDs, que acedem de acordo com a rotação motora, a tarefa *task rpm* relaciona o valor da rotação com os número de luzes acesas.

Cada faixa de rotação motora está relacionada conforme o fluxograma da Figura 64. A relação entre RPM e luzes acessas do conta-giros está resumido na Tabela 16.

A última tarefa, *task hourMeter*, Figura 65, é responsável por administrar o *display* do horímetro. A informação apresentada pelo *display* depende da situação do veículo.

Com o motor funcionando, o horímetro não ajuda o motorista, logo, o *display* passa a exibir o tacômetro.

Com o motor desligado, o *display* pode exibir um conjunto de informações, dependendo do contexto. Caso os indicadores de bateria e temperatura alta encontrem-se ativos, o *display* exibe o valor da medida, relacionada ao indicador, a cada 1 segundo. Caso não, o *display* exibe o tempo de funcionamento do motor.

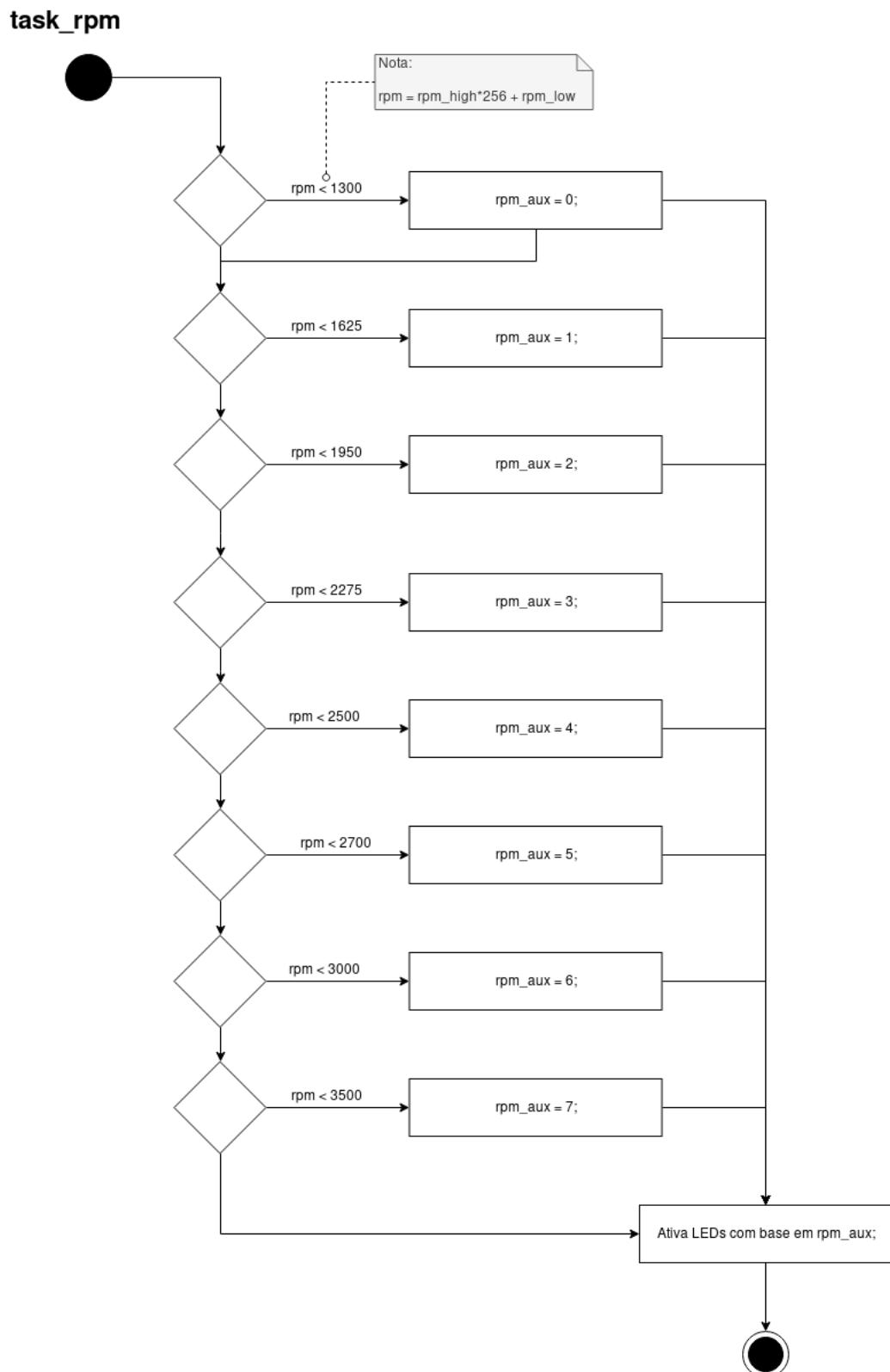
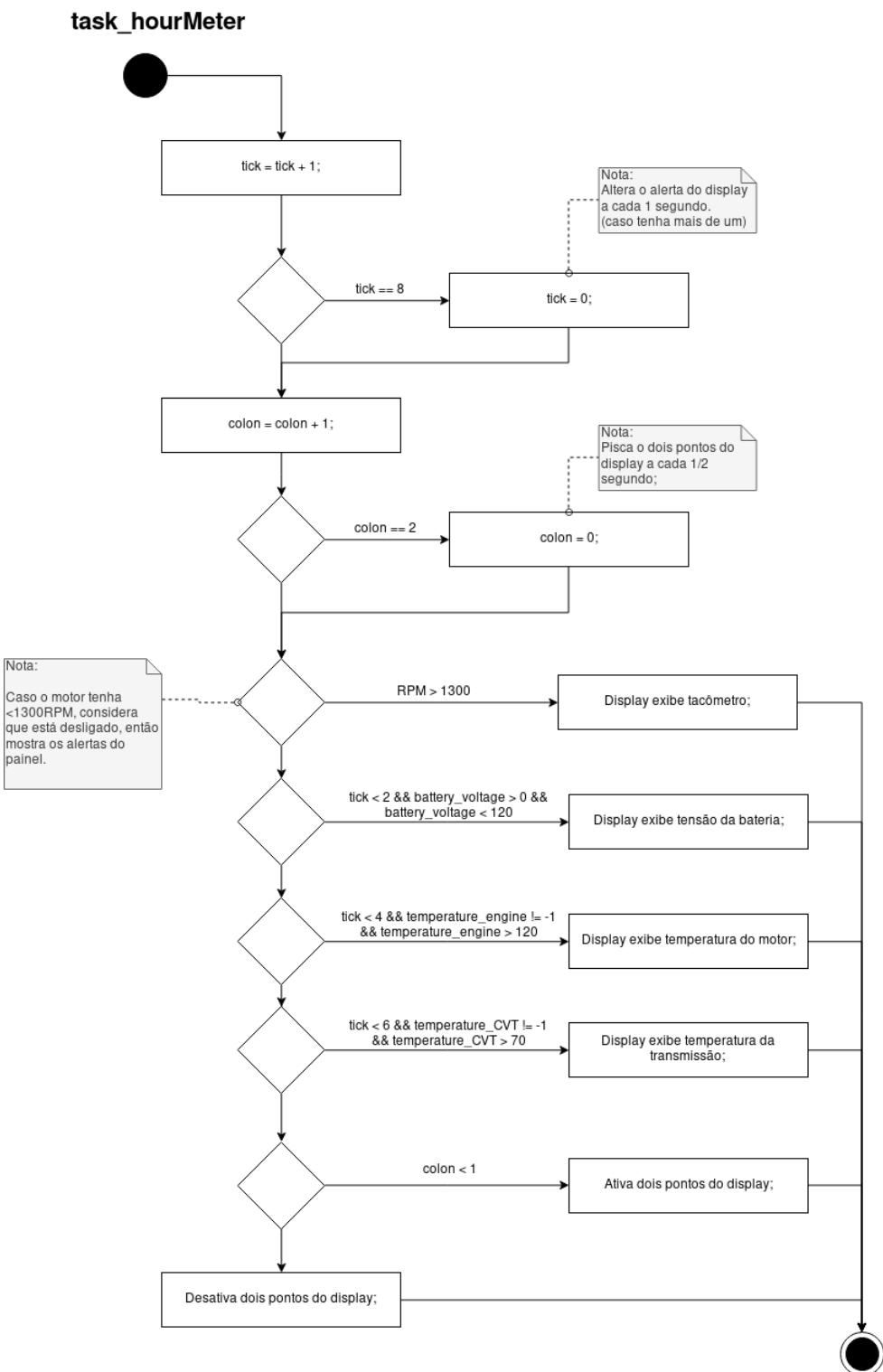
Figura 64 – Fluxograma simplificado da tarefa *task rpm*.**Fonte:** O Autor.

Figura 65 – Fluxograma simplificado da tarefa *task hourMeter*.



Fonte: O Autor.

Tabela 16 – Funcionamento do conta-giros do painel.

RPM	Número de leds ativos (<i>rpm_aux</i>)	Leds no Painel
<1300	0	○ ○ ○ ○ ○ ○ ○ ○
<1625	1	● ○ ○ ○ ○ ○ ○ ○
<1950	2	● ● ○ ○ ○ ○ ○ ○
<2275	3	● ● ● ○ ○ ○ ○ ○ ○
<2500	4	● ● ● ● ○ ○ ○ ○ ○
<2700	5	● ● ● ● ● ○ ○ ○ ○
<3000	6	● ● ● ● ● ● ○ ○ ○
<3500	7	● ● ● ● ● ● ● ○ ○ ○

Fonte: O Autor.

4.1.5.2 Resultados obtidos

O painel está de acordo com o projeto, contando com LEDs e *displays*. Cada indicador conta com um símbolo associado, facilitando na identificação dos alertas. Além disso, o painel conta com uma identificação do carro³. Essa identificação tem valor estético, não implicando em funcionalidade no painel.

Com relação a Figura 66a, o painel de instrumento está conectado a rede veicular, uma vez que está energizado. O módulo de telemetria está conectado a rede, por causa do primeiro indicador apagado. Os módulos: localização, estrutura e motor estão desconectados da rede veicular, posto que os indicadores estão acesos.

Essa fotografia, Figura 66a, foi obtida durante a montagem da eletrônica e por esse motivo o volante não está instalado.

Já em relação a fotografia da Figura 66b, fotografia obtida durante a competição com toda eletrônica instalada⁴, tem-se que:

- Todos os módulos estão conectados, dado que os indicadores estão apagados;
- O sensor de temperatura do motor está desconectado, visto que o indicador de sensor está aceso;
- O veículo está com meio tanque de combustível;
- O horímetro marca 01:01, ou seja, 1 hora e 1 minuto de funcionamento;
- O carro está parado (00 km h^{-1}) com o motor desligado (0 RPM).

³ Durante a competição, todos os carros são identificados pelo número.

⁴ Conforme descrito no módulo do motor, o sensor de temperatura do motor foi retirado. O painel está indicando.

Figura 66 – Fotografias do painel de instrumentos.(a) - Painel
sem volante.(b) - Painel
com volante.**Fonte:** O Autor.

O procedimento para aquisição das ativações das tarefas foi o mesmo descrito na seção 4.1.2.1. As tarefas são periódicas, contudo, os períodos são diferentes do projeto inicial.

Um erro de projeto permitiu que a PPC compartilhasse a porta de comunicação do *display* do horímetro com a rede veicular. Em resumo, o *transceiver* utiliza a porta RB3 (CAN_RX) e o *display* menor também utiliza a porta RB3. Os detalhes elétricos estão descritos no Anexo B e Anexo E.

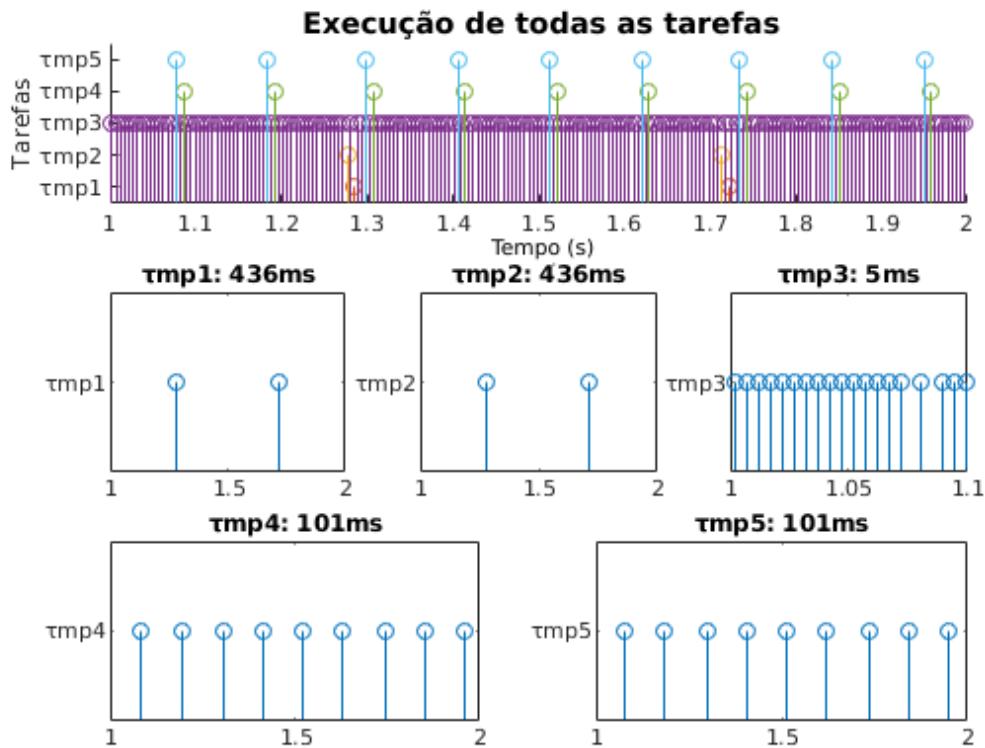
Para contornar o problema, por meio de atualização de *firmware*, o período de ativação da tarefa responsável pelo *display* foi alterado. O novo período de ativação adotado, 5 ms, solucionou o problema. Como resultado dessa alteração, as demais tarefas perderam os seus *deadlines*.

Conforme representado no quadro maior da Figura 67, as tarefas tem ativação periódica. Em relação aos quadros menores, as tarefas τ_{MP1} e τ_{MP2} tem período médio igual a 436 ms. Essas tarefas, τ_{MP1} e τ_{MP2} , foram projetadas inicialmente para 500 ms, mas à ativação antecipada não implica em problemas. A única diferença será na velocidade que o indicador alterna os estados (pisca), o indicador irá operar com 2,29 Hz ao invés de 2 Hz do projeto.

As tarefas τ_{MP4} e τ_{MP5} , conforme os resultados da Figura 67, apresentam, em média, períodos de 101 ms. A princípio, essas tarefas foram definidas com períodos de 100 ms. Esse atraso, 1 ms, não prejudica o funcionamento do painel uma vez que não interfere na percepção do motorista.

Por fim, a tarefa τ_{MP3} tem ativação das tarefas, em média, igual a 5 ms. Esse

Figura 67 – Ativação das tarefas do módulo no decorrer do tempo.



Fonte: O Autor.

período de ativação está superdimensionado. Para solucionar esse problema o diagrama elétrico deve ser revisto. Uma vez que a PCS dispõe das portas A (AN0 até AN4), basta trocar a comunicação do *display* menor para qualquer porta disponível.

4.1.5.3 Conclusão

O painel de instrumentos funciona corretamente, mesmo com as alterações. O módulo além de apresentar ao motorista as informações da dinâmica veicular, ajuda na manutenção, por meio dos indicadores de situações que podem se tornar graves.

Durante os testes, por exemplo, o motorista foi capaz de identificar, por meio do indicador de temperatura alta, que a transmissão do carro estava super aquecendo.

Além de atender aos requisitos propostos, o painel mostrou-se também robusto. Por se tratar de um veículo *off-road*, os módulos eletrônicos devem ser impermeáveis e resistentes. O painel de instrumentos, durante os testes, foi exposto a chuva e a lama; em ambos os casos, não apresentou problemas.

A porta utilizada pelo *display* menor pode ser trocada sem muitas dificuldades.

Essa atualização permite reduzir o uso da PPC, alterando o período de ativação da tarefa responsável pelo *display*.

5 Resultados obtidos - telemetria e análise dos tempos de resposta.

Neste capítulo serão apresentados os resultados obtidos com a telemetria e à análise do tempo de resposta da rede veicular. O sistema de telemetria é formado por 2 (duas) partes principais, sendo a unidade móvel e o módulo eletrônico, ambos serão descritos a seguir.

Os dados analisadas a seguir são da mesma base de dados do capítulo anterior e foram adquiridos durante a 25^a Competição Baja SAE Brasil.

5.1 Módulo de telemetria

O módulo de telemetria, Figura 68a, é responsável pela comunicação com a unidade móvel, seção 5.2, e por armazenar na memória não volátil todas as mensagens da rede veicular. A conexão com a unidade móvel é feita por meio do transmissor ESP8266. Os detalhes do transmissor estão descritos na seção 5.1.2. A seguir será detalhado o *firmware* e o armazenamento do histórico veicular.

O conjunto de placas, PPC e PCS, além do transmissor e o leitor de cartão, foram condicionados dentro do módulo, conforme a Figura 68a¹. Posteriormente, o módulo eletrônico foi instalado na parte superior do automóvel, próximo ao teto, conforme a Figura 68b.

5.1.1 Firmware

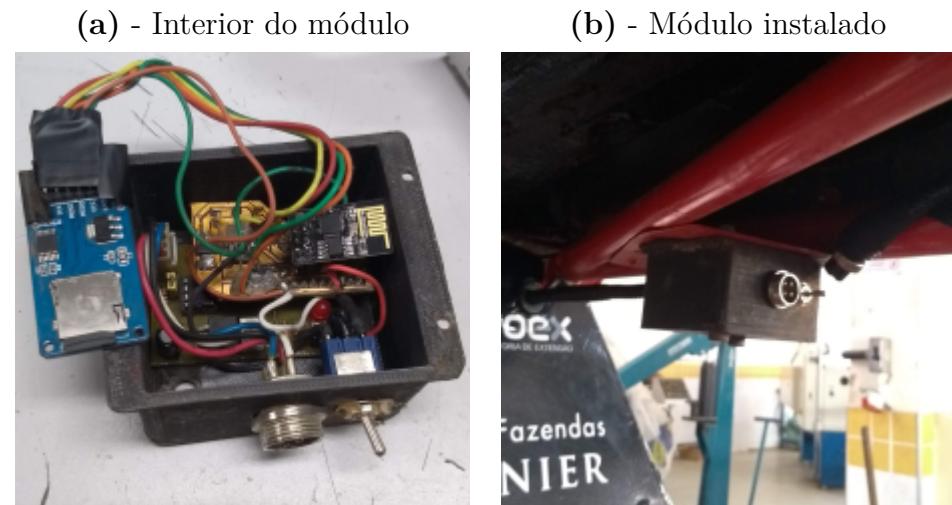
O *firmware* desse módulo, diferente dos demais, é orientado a eventos. Inicialmente o módulo configura a PPC, ativando as interrupções e configurando os periféricos. Essa tarefa, configurar a PPC, é feita de acordo com o fluxograma da Figura 69.

O módulo, a princípio, configura a rede CAN. Em relação a rede CAN, ativa os filtros das mensagens e ajusta a velocidade de transmissão. Após, configura o periférico TIMER1 para utilizar o período (tick) de 0,8 µs. Uma vez que esse *timer* é de 16 bits, o estouro ocorre a cada 52,4288 ms².

Para verificar o SD card, durante a configuração, o *firmware* utiliza uma *flag*. Caso ocorra um erro durante a inicialização do cartão de memória, o módulo ativa a *flag*

¹ O leitor de cartão está fora do módulo apenas para mostrar os detalhes das placas.

² $((2^{16}) - 1) * (0,8 * (10^{-6})) = 0,052428 = 52,428\text{ms}$.

Figura 68 – Fotografias do módulo da telemetria.

Fonte: O Autor.

(sync_status=1) e envia esse valor em uma mensagem CAN. Essa mensagem tem por finalidade ativar o indicador do painel de instrumentos.

Para não sobrescrever a memória, o módulo conta com um contador de mensagem. Após enviar a mensagem com a situação do módulo, o *firmware* recupera o número de mensagens armazenadas no cartão de memória. Com todos os periféricos configurados, o módulo pode funcionar de duas formas, sendo: diagnóstico ou armazenamento. A escolha entre ou diagnóstico ou armazenamento é feita por meio de um interruptor.

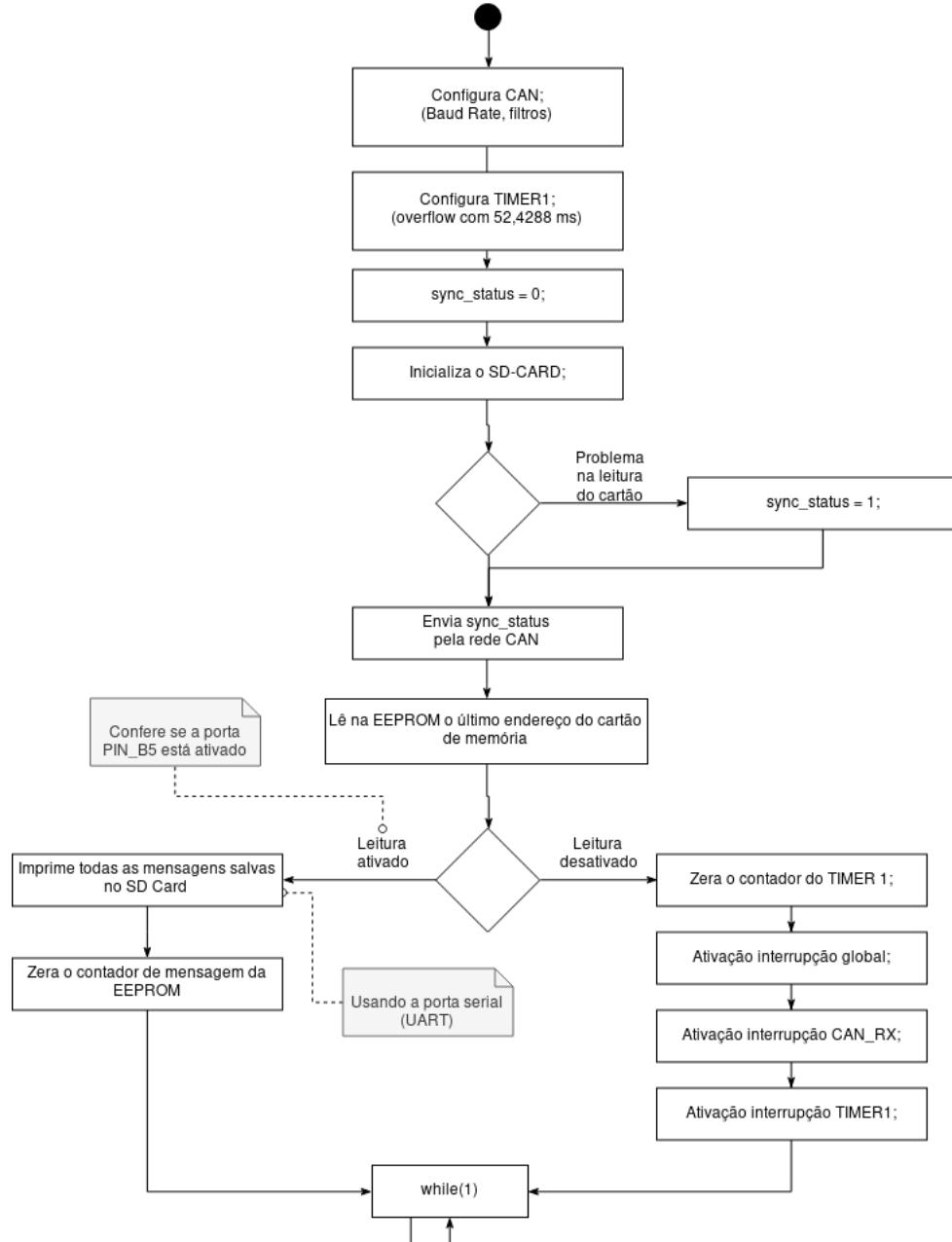
Caso o interruptor esteja na posição de diagnóstico, ou seja, para leitura do cartão de memória, o código entra no ramo da esquerda (Leitura ativada), Figura 69, e o módulo imprime via porta serial todas as mensagens armazenadas no cartão de memória. Após imprimir todas as mensagens armazenadas, o módulo zera o contador e bloqueia o módulo.

Caso o interruptor do módulo esteja na posição de armazenamento, o código entra no ramo da direita (Leitura desativado) da Figura 69. Nesse ramo de execução, o código inicialmente zera o TIMER1 e ativa as interrupções. Após ativar as interrupções, o módulo fica bloqueado esperando a ocorrência dos eventos.

Conforme detalhado na Figura 69, o código ativa a interrupção do *timer* 1 e do recebimento de mensagem CAN. Essas funções foram implementadas de acordo com o fluxograma da Figura 70.

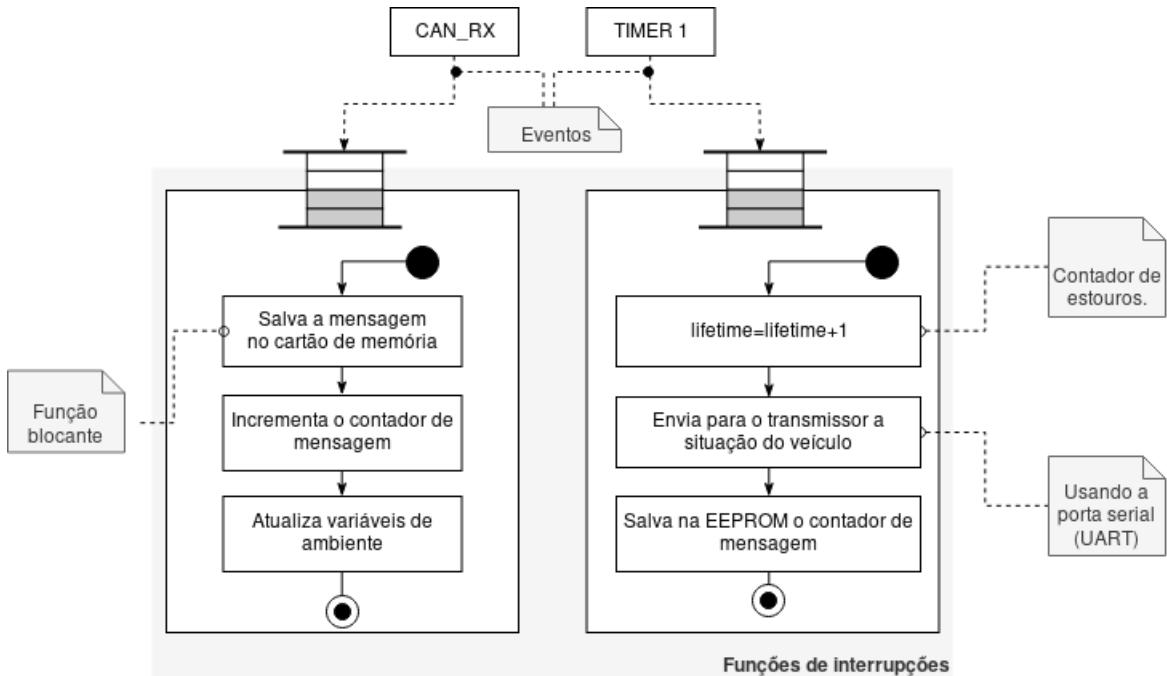
Esses eventos, Figura 70, podem ser classificados em externo ou interno. O *timer* é um evento interno ao módulo e ocorre a cada estouro. Conforme configurado inicialmente, o estouro do *timer* ocorre a cada 52,4288 ms.

A cada interrupção do *timer*, o módulo envia para o transmissor a situação do carro e salva na memória não volátil da PPC o número de mensagens recebidas.

Figura 69 – Fluxograma simplificado da configuração da PPC.**Fonte:** O Autor.

A PPC envia para o transmissor, por meio da porta serial, as informações do veículo. A situação do veículo consiste da tensão da bateria, do nível do tanque de combustível e da temperatura do óleo do motor.

O recebimento de mensagem válida é um evento externo, uma vez que depende dos outros módulos. A cada mensagem recebida, o módulo salva a mensagem no cartão. Para salvar a mensagem no cartão de memória, o módulo utiliza uma estrutura de dados.

Figura 70 – Fluxograma simplificado das interrupções do módulo.**Fonte:** O Autor.

A estrutura utilizada é composta por 15 variáveis, de 8 *bits* cada. Essas variáveis são organizadas de acordo com a Tabela 17. Segundo essa tabela, além de salvar a mensagem recebida com a identificação, o módulo armazena o instante que foi recebida.

Tabela 17 – Estrutura utilizada para salvar no cartão.

N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ID	Tempo				Mensagem CAN				Controle						

Fonte: O Autor.

O primeiro campo, ID, serve para relacionar a mensagem recebida com o módulo que a enviou. Visto que cada mensagem carrega consigo um identificador único, basta aplicar um filtro na recepção da mensagem. As mensagens e os seus respectivos IDs são relacionadas de acordo com a Tabela 18.

Tabela 18 – Relação entre ID e o módulo.

ID	Identificador CAN	Módulo
0	0x1C20	Módulo de Localização
1	0x1C60	Módulo do Motor
2	0x1CE0	Módulo da Estrutura

Fonte: O Autor.

Para salvar o instante que a mensagem chegou ao módulo, a estrutura conta o campo tempo. Esse campo é composto por 5 variáveis, as 3 (três) primeiras salvam o contador de interrupções (*lifetime*) e as 2 (duas) seguintes, o instante atual do *timer*.

A mensagem CAN salva os dados da mensagem recebida. Esses dados representam o funcionamento do carro.

Com intuito de tornar a programação mais simples, a estrutura tem tamanho fixo, ou seja, mesmo que a mensagem CAN contenha apenas 3 bytes de dados, a estrutura utilizada é a mesma.

Por fim, a estrutura contém o campo de controle. Esse campo adiciona uma constante de valor igual a 0xFE³. A principal utilidade da constante de controle é garantir qual é a primeira variável, durante o processo de leitura.

5.1.2 Telemetria

A telemetria é composta por três partes principais, sendo a placa de comunicação, o transmissor e a unidade móvel. Esses três componentes se relacionam de acordo com a Figura 71, onde a PPC já foi descrita na seção anterior, a unidade móvel será tratada na seção 5.2. Por fim, o funcionamento do transmissor será detalhado a seguir.

O módulo conta com um transmissor ESP8266, modelo ESP-01. O transmissor funciona como uma ponte, conectando o automóvel a unidade móvel. Para tal, o ESP conta com um *firmware* próprio, baseado na versão FreeRTOS disponibilizado pelo fabricante.

Adotar um *firmware* com FreeRTOS permitiu que o próprio ESP administrasse todo o processo de conexão, assim, não é necessário tratar de eventos não periódicos na PPC. Situações como conexão e desconexão são tratados diretamente no código do transmissor. O transmissor funciona conforme detalhado no diagrama de estados da Figura 71.

A PPC e a unidade móvel possuem seus próprios estados de funcionamento e estão representados de forma simplificada na Figura 71.

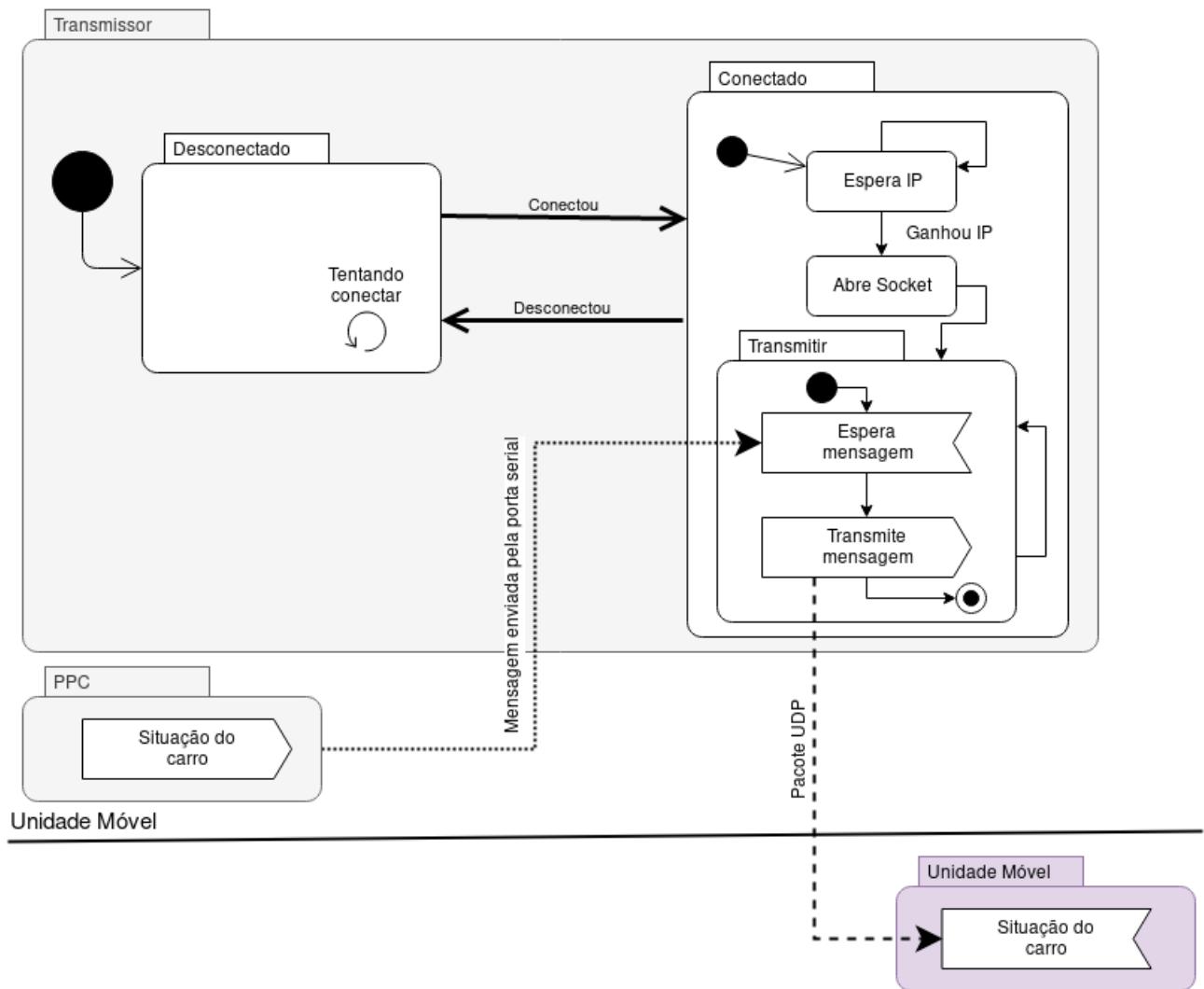
Ainda em relação a Figura 71, o transmissor tem dois estados principais, sendo, quando o transmissor está conectado, ou não, a unidade móvel. Quando o transmissor está desconectado, a única atividade do transmissor é procurar pela rede e caso encontre, conectar a rede.

Quando o transmissor conecta-se à unidade móvel, ambos negociam o endereço do transmissor por meio do protocolo *DHCP*. Após obter um endereço válido, o ESP tenta abrir um *socket* de comunicação com a unidade móvel.

³ $0xFE_{16} = 254_{10}$.

Figura 71 – Diagrama simplificado dos estados do transmissor.

Módulo eletrônico



Fonte: O Autor.

Com um endereço válido e com o *socket* aberto, o transmissor tem condições de comunicar com a unidade móvel. Nesse estado, todas as mensagens recebidas pela porta serial são empacotadas e enviadas para a unidade móvel.

Embora o ESP8266 suporte ambos os protocolos, UDP (*User Datagram Protocol*) e TCP (*Transmission Control Protocol*), a telemetria utiliza o protocolo UDP, principalmente, devido a capacidade de enviar um datagrama após o outro, sem necessariamente obter uma resposta da unidade móvel.

5.1.3 Resultados obtidos

Os resultados da telemetria serão analisados na seção seguinte, sobre a unidade móvel. Nesta seção serão analisados os resultados do sistema para armazenar o histórico veicular.

Para recuperar o histórico veicular, primeiro conectamos um computador ao módulo, usando um adaptador USB-SERIAL, por meio da porta serial disponibilizada na PCS. No computador, abrimos um programa para leitura dos dados. Por fim, colocamos o interruptor na opção de diagnóstico e ligamos a eletrônica do automóvel.

O módulo foi capaz de armazenar o histórico veicular. Para esse trabalho, por exemplo, foram analisadas mais de 60000 mensagens CAN, todas disponibilizadas por esse módulo.

Uma vez que fica inviável detalhar todas as mensagens, a seguir será analisado um conjunto com apenas as 10 (dez) primeiras. A Tabela 19 contém as primeiras mensagens armazenadas na memória do módulo e conforme descrito na seção 5.1.1, seguem a estrutura definida.

Tabela 19 – Primeiras mensagens salvas na memória do módulo.

N	Mensagem salva
1	0 0 0 1 60 17 0 0 0 0 0 0 0 0 254
2	0 0 0 1 183 18 33 0 0 0 0 0 0 0 254
3	0 0 0 2 58 238 3 0 0 0 0 0 0 0 254
4	0 0 0 4 131 214 0 0 0 0 0 0 0 0 254
5	0 0 0 6 205 190 0 0 0 0 0 0 0 0 254
6	0 0 0 8 22 167 0 0 0 0 0 0 0 0 254
7	2 0 0 10 83 133 120 1 1 0 0 0 0 0 254
8	0 0 0 10 94 143 0 0 0 0 0 0 0 0 254
9	0 0 0 12 167 119 0 0 0 0 0 0 0 0 254
10	0 0 0 14 242 95 0 0 0 0 0 0 0 0 254
<i>n</i>	...

Fonte: O Autor.

O formato da Tabela 19 dificulta a interpretação dos dados, contudo, a partir do campo de controle, podemos construir a Tabela 20. Por meio da segunda tabela podemos analisar o estado do veículo, uma vez que sabemos o identificador da mensagem, campo ID, e a mensagem, campo Mensagem CAN.

Dessa forma, sabendo o identificador e a mensagem, basta relacionar a estrutura definida no módulo com a mensagem CAN salva. Por exemplo:

- De acordo com a Tabela 12 a primeira variável do módulo da estrutura (ID=2) é a tensão da bateria multiplicado por 10 (dez), logo, a bateria estava com tensão igual a 12 V;

- De acordo com a Tabela 13 a primeira variável do módulo do motor (ID=1) é a rotação motora, logo, o carro estava desligado (0 RPM);
- De acordo com a Tabela 14 a primeira variável do módulo da localização (ID=0) é a velocidade, logo, o carro estava parado (0 km/h).

Tabela 20 – Estrutura utilizada junto aos dados das mensagens.

Mensagem salva														Controle	
N	ID	Tempo						Mensagem CAN						Controle	
1	0	0	0	1	60	17	0	0	0	0	0	0	0	254	
...															
7	2	0	0	10	83	133	120	1	1	0	0	0	0	254	
...															
36	1	0	0	54	137	218	0	0	0	0	7	255	255	0	254
...															

Fonte: O Autor.

O campo Tempo é composto por 5 (cinco) variáveis, onde: as 3 (três) primeiras compõem o contador de interrupções; as 2 (duas) últimas compõe o instante atual do *timer*. O valor do contador de interrupções e o instante do *timer* são recuperados de acordo com as Equações 5.1 e 5.2.

$$\text{contador} = \text{Tempo1} * 65536 + \text{Tempo2} * 256 + \text{Tempo3} \quad (5.1)$$

$$\text{atual} = \text{Tempo5} * 256 + \text{Tempo4} \quad (5.2)$$

em que,

contador : Total de interrupções causadas pelo Timer1;

atual : O instante atual do Timer1;

Tempo1 : Segunda variável da estrutura de dados;

...

Tempo5 : Sexta variável da estrutura de dados.

Sabendo que o *tick* do timer1 está configurado para 0,8 µs e por consequência, o estouro ocorre a cada 52,4288 ms. O instante que a mensagem chegou ao módulo pode ser recuperado por meio da Equação 5.3.

$$\text{instante} = \text{contador} * 52,4288 * 10^{-3} + \text{atual} * 0,8 * 10^{-6} \quad (5.3)$$

onde,

instante : Instante em que a mensagem chegou ao módulo [s].

Dessa forma é possível recuperar, além do estado do veículo, o instante em que cada mensagem chegou ao módulo.

5.1.4 Conclusão

De acordo com os resultados obtidos, o sistema de armazenamento e diagnóstico do módulo funcionou satisfatoriamente.

Alguns pontos do sistema de armazenamento podem ser aprimorados. Essas melhorias são propostas baseadas nos resultados, ou, como melhorar esses resultados.

O campo Tempo do sistema atual não é contínuo, uma vez que a eletrônica é desligada o sistema zera. Uma solução seria salvar na EEPROM, da PPC, o último valor do campo tempo. Essa alteração impediria o reinício do tempo cada vez que a eletrônica é desligada-ligada.

Uma vez que o sistema é capaz de armazenar o histórico do automóvel, cruzar esses dados com os acontecimentos dependem de um relógio sincronizado. Uma melhoria seria adoção de um relógio RTC (Relógio de tempo real).

5.2 Unidade móvel de telemetria

O sistema de telemetria é composto por 3 (três) componentes principais, sendo: o módulo eletrônico, formado pela PPC e o transmissor, e a unidade móvel. O módulo eletrônico já foi descrito em seções anteriores. Nesta seção será detalhado o funcionamento da unidade móvel.

No sistema de telemetria, a unidade móvel tem o papel de gerenciar a infraestrutura de rede, receber as informações do automóvel e exibir ao usuário. Conforme projetado, a unidade conta com um transmissor ESP8266, um *display* 16x2 e um microcontrolador PIC18F2580. Todos esses componentes são condicionados dentro de uma caixa, Figura 72a.

Para a unidade gerenciar a rede e receber as informações do automóvel, o transmissor conta com um *firmware* baseado em FreeRTOS. O *firmware* desenvolvido está representado por meio de diagrama de estados na Figura 73. O código ainda implementa a interface projetada, também representada no diagrama.

Figura 72 – Fotografias da unidade móvel.

(a) - Unidade desligada



(b) - Unidade ligada



(c) - Unidade conectada ao veículo



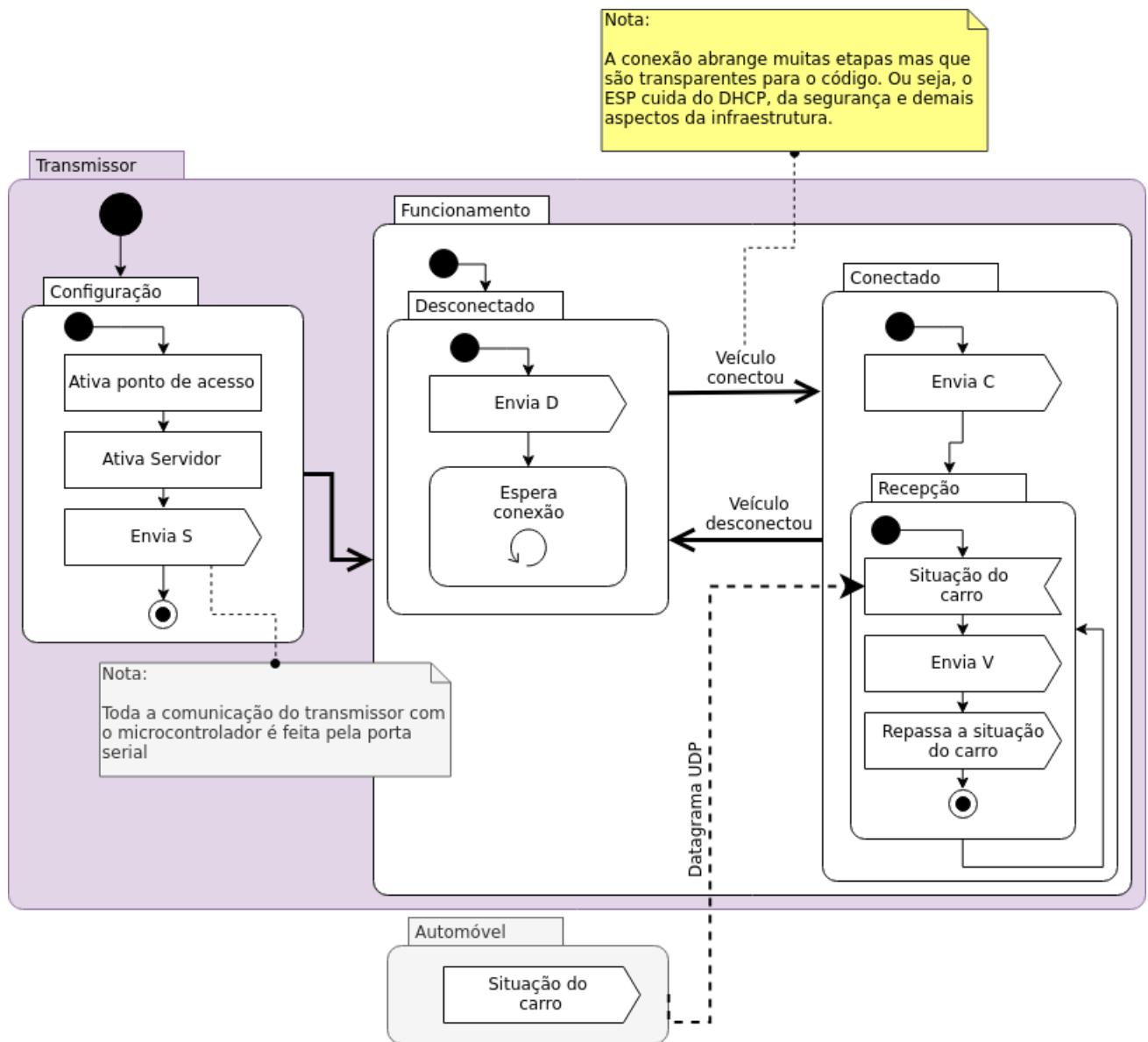
Fonte: O Autor.

Inicialmente o transmissor tem suas funções ativadas, sendo o ponto de acesso e o servidor (UDP). Após o servidor ser configurado, o transmissor informa ao microcontrolador conforme definido na interface. A unidade móvel se encontra nesse estado na Figura 72b.

Com a configuração concluída, o transmissor entra em funcionamento. A princípio os transmissores estão desconectados, uma vez que o ponto de acesso estava desativado. Nesse estado, desconectado, o transmissor informa ao microcontrolador seguindo a interface projetada e fica esperando pela conexão.

Uma vez que o automóvel se conectou a rede, o transmissor informa ao microcontrolador e espera pelo datagrama UDP contendo a situação do carro. Nesse estado, recepção, toda vez que um datagrama é recebido pelo transmissor, os seguintes passos são feitos: Primeiro, informa ao microcontrolador que uma mensagem chegou. Em seguida repassa a situação do carro ao microcontrolador.

Figura 73 – Diagrama simplificado dos estados do transmissor.



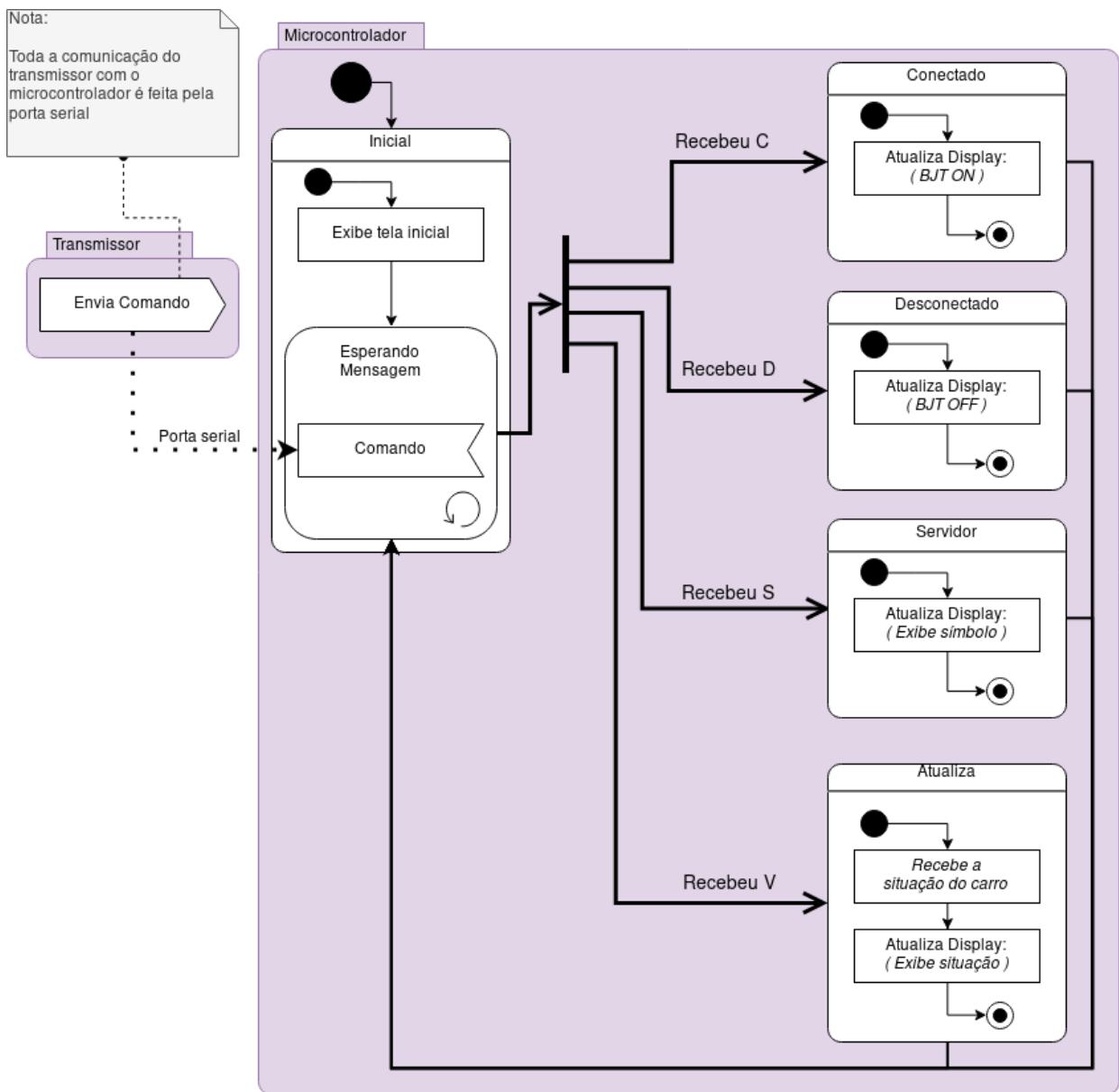
Fonte: O Autor.

Na Figura 72c, a unidade móvel está conectada ao veículo, conforme informa a primeira linha do *display*. A segunda linha informa a situação do carro. De acordo com a segunda linha, o veículo está com o nível de combustível na reserva (EMPT), a bateria está com 12.0 V e o sensor de temperatura do motor está com problemas⁴.

A comunicação entre o transmissor e o microcontrolador, baseada na interface projetada, é feita por meio da porta serial. O microcontrolador implementa o protocolo de acordo com o diagrama de estados da Figura 74.

⁴ Conforme descrito no módulo do motor, o sensor foi retirado.

Figura 74 – Diagrama simplificado dos estados do microcontrolador.



Fonte: O Autor.

Inicialmente o microcontrolador exibe no *display* uma mensagem informando que o módulo está ativo e espera os comandos do transmissor. Sempre que recebe um comando do transmissor, o microcontrolador atualiza parte do *display*. Conforme projetado, a interface consiste de 4 (quatro) tipos de mensagens.

Quando o microcontrolador recebe o comando C ou o comando D, o *display* passa a informar se o carro está conectado ou não. O comando S informa que o servidor está funcionando, no *display* é exibido um símbolo de conexão.

O comando V informa que um datagrama chegou ao transmissor contendo a situação

do automóvel. O microcontrolador primeiro recebe os dados do veículo, depois atualiza o *display* com essas informações.

5.2.1 Conclusão

A unidade móvel é capaz de se comunicar com o veículo e informar ao usuário a situação do mesmo, contudo, foram descobertas algumas limitações durante os testes, ou seja, durante a competição.

Durante os testes, a transmissão funcionava corretamente dentro do boxe da equipe. Uma vez o carro em movimento, dentro da pista, não havia troca de informação com a unidade móvel.

O problema anterior indicou a falta de tratamento dos erros. O projeto atual não conta com nenhum indicador, mensagem ou qualquer informação dos problemas que ocorrem, todo o projeto considera o perfeito funcionamento.

O tratamento de erros pode ser feito via *software*, sendo assim, basta uma atualização do *firmware* desenvolvido com adição do código para tratamento de erros.

A existência do tratamento não resolve a falta de comunicação, entretanto, o conjunto de indicadores levaria a causa raiz do problema, e por consequência, a solução mais efetiva. Atualmente não existe nenhuma informação que indique qual o impedimento da comunicação.

Uma melhoria ao sistema de telemetria seria adoção de transmissores com antenas externas, o modelo ESP-05, baseado no ESP8266, permite essa atualização sem muitas alterações do projeto.

5.3 Análise da rede CAN

Nesta seção serão apresentados os resultados obtidos com à aplicação da análise do tempo de resposta no pior tempo, apresentado na seção 2.5, com os dados do sistema proposto e implementado. Para tanto, inicialmente serão apresentados os principais pontos da rede veicular. Por fim, será feita uma análise da rede veicular com os dados do módulo da telemetria.

O sistema projetado consiste de 5 (cinco) módulos, sendo que apenas 4 (quatro) transmitem mensagens pelo barramento. O uso da rede veicular, pelos módulos, pode ser resumida de acordo com a Tabela 21.

Ainda em relação a Tabela 21 e conforme já descrito na seção 3.2.5, o módulo da telemetria envia uma mensagem não periódica. Uma vez que a mensagem é aperiódica, ela não será considerada nas análises a seguir.

Tabela 21 – Resumo da configuração da rede CAN por módulo.

Módulo	CAN ID	Tamanho (bytes)	Período (ms)
Localização	0x1C20	3	100
Telemetria	0x1C40	1	x
Motor	0x1C60	8	100
Estrutura	0x1CE0	4	500

Fonte: O Autor.

Entre os módulos com mensagens periódicas, o módulo de localização tem a maior prioridade de acesso ao meio, seguido pelo módulo do motor e por último o módulo da estrutura com a menor prioridade.

5.3.1 Análise do tempo de resposta

Para analisar se a rede veicular atende aos requisitos temporais, foi desenvolvido uma aplicação em Python para solucionar as equações apresentadas na seção 2.5. A seguir serão apresentados as entradas utilizadas para a solução das equações.

Com base na Tabela 21 podemos obter, de cada mensagem, a prioridade (m), o tamanho (b_m) e o período (T_m).

O atraso de resposta (J_m), ou *Jitter*, foi estimado a partir do código. Uma vez que cada módulo utiliza sempre a mesma tarefa para o envio da mensagem, podemos considerar o atraso como constante.

Uma vez que a mensagem deve ser enviada antes da próxima ativação da tarefa, caso contrário, será sobreescrita. O *deadline* pode ser aproximado, sabendo que $D_m \leq T_m + J_m$.

Em sistemas que exigem resposta em tempo real, ou Classe C, a velocidade da rede pode variar entre 125 Kbit/s até 1 MBit/s.

Tabela 22 – Resumo do escalonamento das mensagens.

ID CAN	m	b_m (bytes)	J_m (μ s)	T_m (ms)	D_m (ms)	R_m (μ s) 1 MBit/s	R_m (μ s) 250 Kbit/s	R_m (μ s) 125 Kbit/s
0x1C20	1	3	7	100	\approx 100	227	887	1767
0x1C60	2	8	15	100	\approx 100	330	1275	2535
0x1CE0	3	4	9	500	\approx 500	324	1269	2529

Fonte: O Autor.

Por fim, todas as entradas, bem como o pior tempo das mensagens, estão dispostas na Tabela 22.

Embora essa análise não seja completa, uma vez que desconsidera a mensagem aperiódica do módulo da telemetria, as mensagens de erros e os processos de retransmissão das mensagens, o sistema é escalonável mesmo para a menor taxa de transmissão, de 125 Kbit/s. Em todos os cenários, o sistema atende os requisitos temporais, visto que $R_m < D_m$.

Uma vez que a rede é escalonável, adotou-se para o sistema eletrônico a velocidade de 1 MBit/s. Com base nessa configuração, 1 MBit/s, os resultados a seguir foram obtidos.

5.3.2 Resultados obtidos

Com base no histórico veicular, a rede foi capaz de entregar 67730 mensagens válidas, distribuídas de acordo com o histograma da Figura 75a.

O módulo da localização entregou aproximadamente⁵ cinco vezes mais mensagens que o módulo da estrutura, resultado esperado, uma vez que o período de envio do módulo da estrutura é cinco vezes maior. O módulo da localização tem período de 0,1 s e o módulo da estrutura 0,5 s.

O módulo do motor enviou menos mensagens válidas que o módulo da localização, embora ambos os módulos tenham o mesmo período de ativação. Esse diferença ocorreu pois o módulo do motor foi desconectado durante a competição.

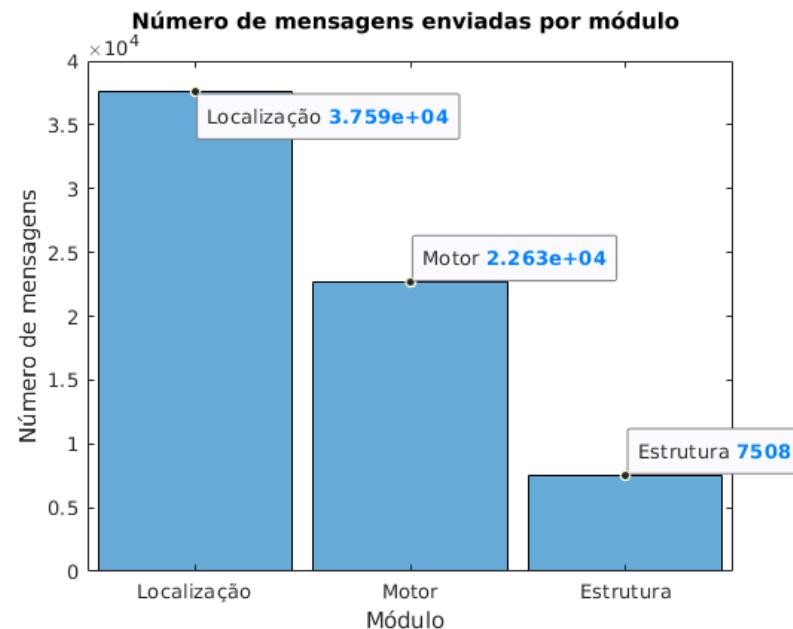
Conforme representando no gráfico da Figura 75b, as mensagens foram entregues com períodos médios iguais a: $0,1020 \pm 0,0293\text{s}$ para o módulo da localização, $0,1001 \pm 0,0060\text{s}$ para o módulo do motor e $0,5008 \pm 0,0492\text{s}$ para o módulo da estrutura.

Desta forma, o sistema funciona como projetado atendendo aos requisitos temporais.

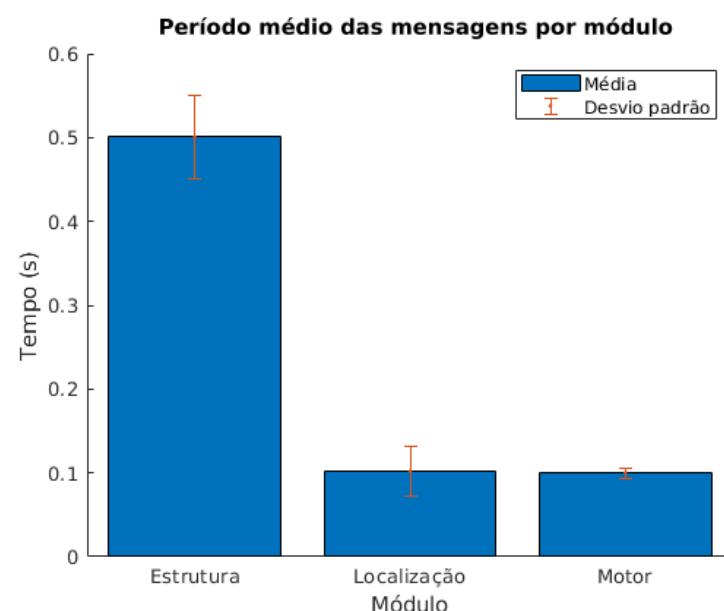
⁵ A diferença foi de 51 mensagens, que representa 5,1 segundos.

Figura 75 – Análise da rede CAN.

(a) - Número de mensagem por módulo.



(b) - Período médio das mensagens entregues.

**Fonte:** O Autor.

6 Conclusão

Esse trabalho se mostrou um grande desafio, uma vez que o seu projeto e desenvolvimento contemplou não somente as principais áreas de atuação de um engenheiro da computação, mas permitiu uma vivência em um ambiente multidisciplinar. A integração da eletrônica veicular ao automóvel da equipe Baajatinga, modelo BJT06, viabilizou que testes em condições reais de uso fossem realizados.

Essa integração, eletrônica ao veículo, permitiu a descoberta de diversos pontos de melhorias. As possíveis correções ou melhorias foram tratadas no decorrer do texto. Algumas das soluções foram adotadas durante o desenvolvimento, e em resumo, o projeto cumpre com os objetivos, uma vez que:

1. O módulo da estrutura é capaz de medir a carga da bateria e analisar o nível de combustível do automóvel;
2. O módulo do motor é capaz de medir a rotação motora, a temperatura da transmissão e do fluído de arrefecimento do motor (óleo) e o tempo de funcionamento;
3. O módulo de localização é capaz de medir a velocidade e analisar o reservatório do fluído de freio;
4. O painel de instrumentos é capaz de apresentar ao motorista as informações da dinâmica veicular e os indicadores das situações que podem se tornar graves, além de ser atualizado em tempo real;
5. O sistema de armazenamento e diagnóstico do módulo de telemetria funcionou satisfatoriamente. E a unidade móvel é capaz de se comunicar com o veículo por meio do módulo da telemetria;
6. Todos os módulos escalonam as tarefas por meio de *firmware* baseado em executivo cíclico.

Pode-se concluir que a eletrônica veicular atendeu aos requisitos, sendo capaz de medir, informar ao motorista e armazenar o histórico de uso do veículo. Além de atender aos requisitos projetados, o sistema funcionou em situações adversas as quais um veículo *off-road* é submetido. Durante os testes, o veículo, em especial a eletrônica, foi exposto a forte chuva e vento, além de lama.

Embora a unidade móvel tenha funcionado parcialmente durante os testes, o projeto atende aos requisitos propostos, uma vez que a unidade é capaz de se comunicar com o veículo por meio de comunicação sem fio.

Além do projeto elétrico, esse trabalho apresentou os resultados obtidos por módulo. Esses resultados representam o histórico veicular, permitindo que em carros de competição, caso da equipe Baajatinga, seja possível manter um histórico da evolução e de análise da dinâmica veicular.

Em relação a rede veicular, o projeto de análise da rede CAN garante, mesmo na pior situação, que a rede será capaz de entregar as mensagens dentro do limite projetado. Essa garantia é importante, posto que a eletrônica veicular é um sistema de tempo real. O funcionamento correto do painel eletrônico está diretamente ligado a capacidade de cumprimento das restrições temporais, o motorista deve ser informado da situação atual do veículo.

O histórico veicular, obtido por meio da eletrônica, confirmou a confiabilidade da rede veicular. As mensagens foram entregues segundo os prazos projetados.

Mesmo fora do escopo desse trabalho, o resultado da apresentação do projeto na 25^a Competição Baja SAE BRASIL (PETROBRAS) confirma a competitividade do sistema elétrico, com qualidade comparável a equipes de todo o país. Este projeto, e os seus resultados, conferiu a equipe Baajatinga o posto de sétimo (7º) melhor projeto elétrico da competição.

6.1 Sugestões para trabalhos futuros

Diversos pontos de melhorias foram apresentadas, de forma pontual, ao decorrer do texto. Em relação ao sistema completo, sugere-se:

- Realizar um estudo sobre a tolerância de falhas e implementar nos módulos:
 - Mecanismo para detecção de falhas;
 - Mecanismo para tratamento das falhas;
 - Adoção de CRC (*Cyclic redundancy check*) nos dados salvos pelos módulos.
- Aprimorar o sistema de diagnóstico, por meio:
 - Do desenvolvimento de um protocolo de comunicação com o módulo;
 - Da padronização do conector, para facilitar a conexão do computador com o módulo (Por exemplo: conector USB);
 - Da capacidade de enviar o histórico veicular por meio da telemetria;
 - Da adoção de um relógio sincronizado (RTC).
- Implementar no sistema elétrico a capacidade de recarregar a bateria:
 - Pode-se adotar recarga solar;

- Pode-se desenvolver uma plataforma externa (fonte), que se conecta ao barramento (chicote) e recarrega a bateria.
- Adoção de faróis de LED com controle automático, onde:
 - Ativação/Desativação automática ao anoitecer;
 - Desativação automática em caso de bateria em nível crítico.
- Aprimorar o painel com adoção de *display* LCD.

Referências

- AMERICA, L. © 2018 BMW of N. **BMW M Performance Electronic Steering Wheel for M Sport Equipped Vehicles.** [S.l.], 2018. Disponível em: <https://shop.bmw.com/bmw-us-responsive/en_US/p/-/bmw-m-performance-electronic-steering-wheel-for-m-sport-equipped-vehicles/2230189/>. Acesso em: 15 jul. 2018. Citado na página 58.
- BISHOP, R. H. **The Mechatronics Handbook, -2 Volume Set.** [S.l.]: CRC press, 2002. Citado na página 24.
- BOSCH, R. et al. **Automotive electrics, automotive electronics.** 5th. ed. [S.l.]: Springer vieweg, 2007. 85, 92, 71, 72, 73, 78, 79 p. Citado 17 vezes nas páginas 18, 23, 26, 27, 28, 29, 30, 31, 32, 34, 35, 36, 38, 40, 41, 42 e 46.
- BOYES, W. **Instrumentation reference book.** [S.l.]: Butterworth-Heinemann, 2009. Citado 2 vezes nas páginas 24 e 25.
- CO., L. Y. H. E. **Cylinder Inductive Proximity Switch Series.** [S.l.], 2008. Disponível em: <<https://opencircuit.nl/ProductInfo/1000064/LJ12A3.pdf>>. Acesso em: 10 jul. 2018. Citado na página 53.
- DIARRA, R. B. G. A. **OSI Layers in Automotive Networks.** [S.l.], 2013. Disponível em: <<http://www.ieee802.org/1/files/public/docs2013/new-tsn-diarra-osi-layers-in-automotive-networks-0313-v01.pdf>>. Acesso em: 23 ago. 2018. Citado na página 30.
- ELECTRONICS, T. M. **LED Drive Control Special Circuit.** [S.l.]. Disponível em: <www.titanmec.com/index.php/product/download/id/302.html>. Acesso em: 15 jul. 2018. Citado 2 vezes nas páginas 57 e 147.
- EMMELMANN, M.; BOCHOW, B.; KELLUM, C. **Vehicular networking: Automotive applications and beyond.** [S.l.]: John Wiley & Sons, 2010. v. 2. Citado 2 vezes nas páginas 18 e 34.
- ESPRESSIF. **ESP8266EX - Datasheet.** [S.l.], 2018. Disponível em: <https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf>. Acesso em: 24 jul. 2018. Citado 2 vezes nas páginas 60 e 62.
- GONZAGA, J. d. S. S. M. T. Desenvolvimento de um sistema de telemetria embarcado para veículos off-road utilizando as redes can e ieee 802.11. UNIVASF/PIVIC, 2018. Citado 2 vezes nas páginas 18 e 20.
- GRZEMBA, I. A. **MOST: the automotive multimedia network.** [S.l.]: Franzis Verlag, 2012. Citado na página 31.
- HILL, J. **Real Time Operating System (RTOS) Documentation V2.1.** [S.l.], 2018. Disponível em: <<http://silanus.fr/sin/formationISN/Robotique/Logiciels/CCS/Data%20Sheets/RTOS.pdf>>. Acesso em: 17 set. 2018. Citado na página 47.

- HOLLEMBEAK, B. **Today's Technician: Automotive Electricity and Electronics, Classroom and Shop Manual Pack.** [S.l.]: Cengage Learning, 2014. 70, 71 p. Citado na página 53.
- INC, M. T. **MCP2551.** [S.l.], 2007. Citado na página 46.
- INC, M. T. **PICKIT™ 2 PROGRAMMER-TO-GO USER GUIDE.** [S.l.], 2008. Disponível em: <<http://ww1.microchip.com/downloads/en/DeviceDoc/PICkit%202%20Programmer-To-Go%20User%20Guide%20b.pdf>>. Acesso em: 10 jul. 2018. Citado na página 46.
- INC, M. T. **PIC18F2480/2580/4480/4580 Data Sheet.** [S.l.], 2009. Citado 4 vezes nas páginas 46, 52, 63 e 142.
- INSTRUMENTS, T. **LM1117 800-mA Low-Dropout Linear Regulator.** [S.l.]. Disponível em: <<http://www.ti.com/lit/ds/symlink/lm1117.pdf>>. Acesso em: 15 abr. 2019. Citado na página 148.
- INSTRUMENTS, T. **LM340,LM340Aand LM7805 Family Wide Vin 1.5-A Fixed VoltageRegulators.** [S.l.]. Disponível em: <<http://www.ti.com/lit/ds/symlink/lm7800.pdf>>. Acesso em: 15 abr. 2019. Citado na página 129.
- INSTRUMENTS, T. **SNx4HC5958-Bit Shift Registers With 3-State Output Registers.** [S.l.], 2015. Disponível em: <<http://www.ti.com/lit/ds/symlink/sn74hc595.pdf>>. Acesso em: 15 jul. 2018. Citado 2 vezes nas páginas 57 e 143.
- INSTRUMENTS, T. **LIN Protocol and Physical Layer Requirements.** [S.l.], 2018. Disponível em: <[http://www.ti.com/lit/an/slra383.pdf](http://www.ti.com/lit/an/slla383/slra383.pdf)>. Acesso em: 23 ago. 2018. Citado na página 30.
- KESKIN, U. In-vehicle communication networks: a literature survey. **Computer Science Report**, v. 10, 2009. Citado na página 23.
- KUROSE, J. F.; ROSS, K. W. **Computer networking: a top-down approach: seventh edition.** 7th. ed. [S.l.]: Pearson Higher Ed, 2017. 487,488 p. Citado na página 29.
- LEE, I.; LEUNG, J. Y.; SON, S. H. **Handbook of real-time and embedded systems.** [S.l.]: CRC Press, 2007. Citado 3 vezes nas páginas 32, 33 e 34.
- MANIYAR, S. **1-Wire® Communication with PIC® Microcontroller.** [S.l.], 2008. Disponível em: <<http://ww1.microchip.com/downloads/en/AppNotes/01199a.pdf>>. Acesso em: 10 jul. 2018. Citado na página 52.
- MÁRIO, D. Tecnologia de redes de comunicação e computadores. **Rio de Janeiro: Axcel Books**, p. 150, 152, 153, 2002. Citado na página 28.
- MCLAREN©. **A guide to the F1 steering wheel.** [S.l.], 2018. Disponível em: <<https://www.mclaren.com/formula1/2017/malaysian-grand-prix/f1-steering-wheel-guide-3333333/>>. Acesso em: 15 jul. 2018. Citado na página 58.
- MORIMOTO, C. E. Redes, guia prático. **Porto Alegre: Sul Editores**, v. 4, p. M857r, 2008. Citado na página 26.

- MORRIS, A. S.; LANGARI, R. **Measurement and instrumentation: theory and application.** [S.l.]: Academic Press, 2012. Citado 3 vezes nas páginas 24, 25 e 26.
- NATALE, M. D. et al. **Understanding and using the controller area network communication protocol: theory and practice.** [S.l.]: Springer Science & Business Media, 2012. Citado 8 vezes nas páginas 34, 35, 36, 37, 38, 39, 41 e 42.
- NAVET, N.; SIMONOT-LION, F. **Automotive embedded systems handbook.** [S.l.]: CRC press, 2009. 1-5, 1-9, 4-5, 4-2 p. Citado 11 vezes nas páginas 18, 23, 24, 26, 28, 29, 34, 35, 41, 46 e 60.
- NICOLAZZI, L. C.; ROSA, E. da; LEAL, L. d. C. M. Uma introdução à modelagem quase-estática de automóveis. **Publicação interna GRANTE, Departamento de Engenharia Mecânica. Universidade Federal de Santa Catarina, Florianópolis,** p. 122, 123, 124, 126, 2012. Citado na página 55.
- PRODUCTS, I. M. I. **Programmable Resolution 1-Wire Digital Thermometer.** Rev 4. [S.l.], 2008. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>. Acesso em: 10 jul. 2018. Citado na página 52.
- QUEISSER, H.-J. **Sensors for automotive applications.** [S.l.]: John Wiley & Sons, 2006. v. 4. Citado 3 vezes nas páginas 18, 25 e 26.
- SÁ, J. da S. **Projeto e Análise de um Sistema de Aquisição de Dados Distribuído para Aplicações em Tempo Real.** 12, 13, 14 p. Dissertação (Mestrado) — Universidade Federal de Campina Grande, 2007. Citado 7 vezes nas páginas 26, 28, 29, 35, 37, 38 e 41.
- SÁ, J. da S. **Contribuição à Modelagem de Sistemas de Controle via Rede.** 50 p. Tese (Doutorado) — Universidade Federal de Campina Grande, 2016. Citado 3 vezes nas páginas 32, 33 e 34.
- SAMEK, M. **Practical UML statecharts in C/C++: event-driven programming for embedded systems.** [S.l.]: CRC Press, 2008. Citado na página 71.
- SANGIOVANNI-VINCENTELLI, A.; NATALE, M. D. Embedded system design for automotive applications. **Computer, IEEE,** v. 40, n. 10, 2007. Citado na página 23.
- SANTINI, A. **Automotive Electricity & Electronics.** [S.l.]: Cengage Learning, 2012. 29, 30 p. Citado na página 53.
- SERIES, C. **Eletric Capacitive Type.** [S.l.], 2008. Disponível em: <http://www.farnell.com/datasheets/1533393.pdf?_ga=2.244602885.1492393658.1530580321-1798801827.1530580321>. Acesso em: 10 jul. 2018. Citado na página 48.
- SERVICES, I. C. C. **CCS C Compiler Manual.** [S.l.], 2018. Disponível em: <https://www.ccsinfo.com/downloads/ccs_c_manual.pdf>. Acesso em: 10 jul. 2018. Citado 2 vezes nas páginas 47 e 61.
- SHAW, R.; JACKMAN, B. An introduction to flexray as an industrial network. In: **IEEE. Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on.** [S.l.], 2008. p. 1849–1854. Citado na página 31.

STANDARDIZATION, I. O. for. **Road vehicles-Symbols for controls, indicators, and tell-tales.** [S.l.]: International Standards Organization Geneva, Switzerland, 2000. Citado na página 58.

UNICOBA, D. da. **Manual Técnico - UNIPOWER - Série UP.** [S.l.], 2018. Disponível em: <<https://www.robocore.net/upload/ManualTecnicoBateriaUnipower.pdf>>. Acesso em: 10 jul. 2018. Citado na página 47.

ZURAWSKI, R. **Industrial communication technology handbook.** [S.l.]: CRC Press, 2014. Citado 2 vezes nas páginas 30 e 31.

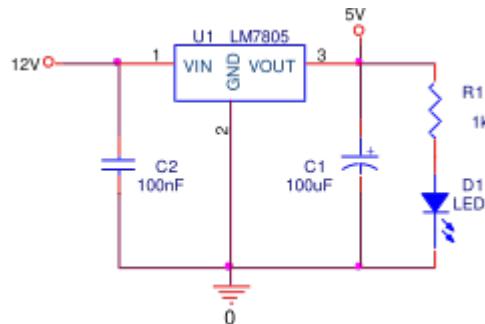
ANEXO A – Placa de processamento e comunicação

A placa de processamento e comunicação contém diversos circuitos que auxiliam no funcionamento dos módulos. A seguir serão destacados os modelos, os circuitos condicionadores de sinal, e as suas características, de todos os componentes da placa de processamento e comunicação.

Os veículos, por padrão, utilizam baterias com tensão de 12 V. Uma vez que o microcontrolador, o *transceiver* e os sensores de temperatura, funcionam com 5 V, utiliza-se um LM7805 (INSTRUMENTS, b) para regular a tensão. Na Figura 76 está representado o diagrama do circuito elétrico utilizado para regular a tensão da PPC.

Os capacitores C2 e C1 são utilizados para filtragem e desacoplamento DC (*Direct Current*). O led D1 é utilizado para indicar se a placa está energizada.

Figura 76 – Diagrama elétrico do regulador de tensão da PPC.



Fonte: O Autor.

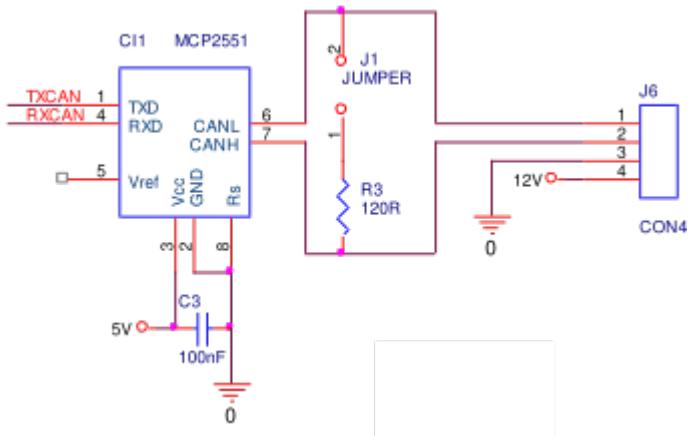
O diagrama elétrico utilizado para acoplar os módulos ao barramento CAN, bem como o diagrama elétrico do *transceiver* CAN, estão representados na Figura 77.

Os módulos são conectados ao barramento por meio de um conector composto por quatro vias (+12V, 0V, CAN H e CAN L). O conector J6 faz o roteamento desses sinais.

Para conectar o módulo ao barramento CAN, a PPC conta com um *transceiver* CAN MCP2551 da Microchip. O circuito ainda conta com resistências para evitar a reflexão do sinal, ativado por meio de um *jumper*. Esse resistor é ativado apenas nos módulos terminais.

Conforme detalhado na Figura 78, a PPC conta com um microcontrolador PIC18F2580

Figura 77 – Diagrama elétrico do *transceiver* CAN da PPC.



Fonte: O Autor.

da Microchip, com os circuitos para o botão de *reset*, interface para atualização de *firmware* e o circuito do oscilador. Além de fornecer acesso as portas do microcontrolador, os conectores disponibilizam tensão de 5 V, 12 V e o GND.

O botão de *reset* SW1, ao ser pressionar, reinicia o PIC18F2580 zerando a tensão no pino 1 do PIC. Ao ser solto, o botão abre o circuito e a tensão do pino 1 volta para 5V, liberando o microcontrolador para execução.

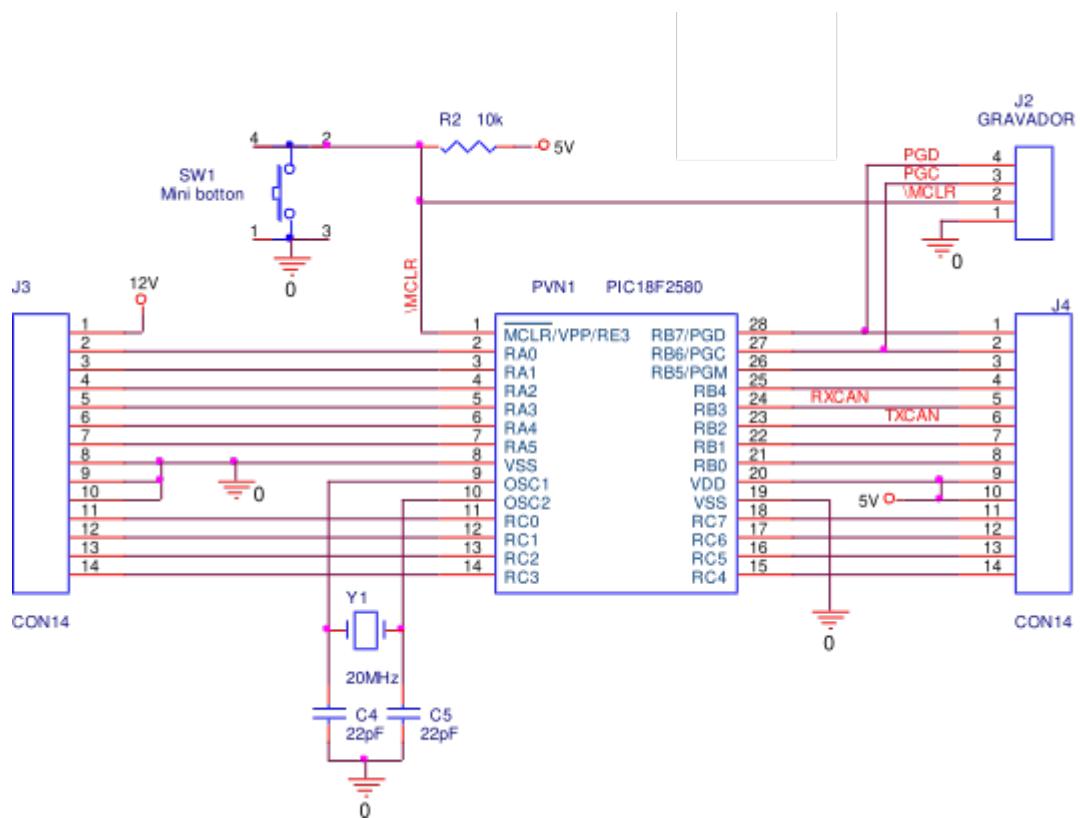
A interface de atualização do *firmware* conta com um conector, com acesso aos pinos 28 e 27 do microcontrolador. Esse circuito é o disponibilizado pelo próprio *datasheet* do PIC.

O microcontrolador conta com um cristal externo Y1, de 10 MHz. Para o bom funcionamento do cristal, os capacitores C4 e C5 são ligados ao cristal.

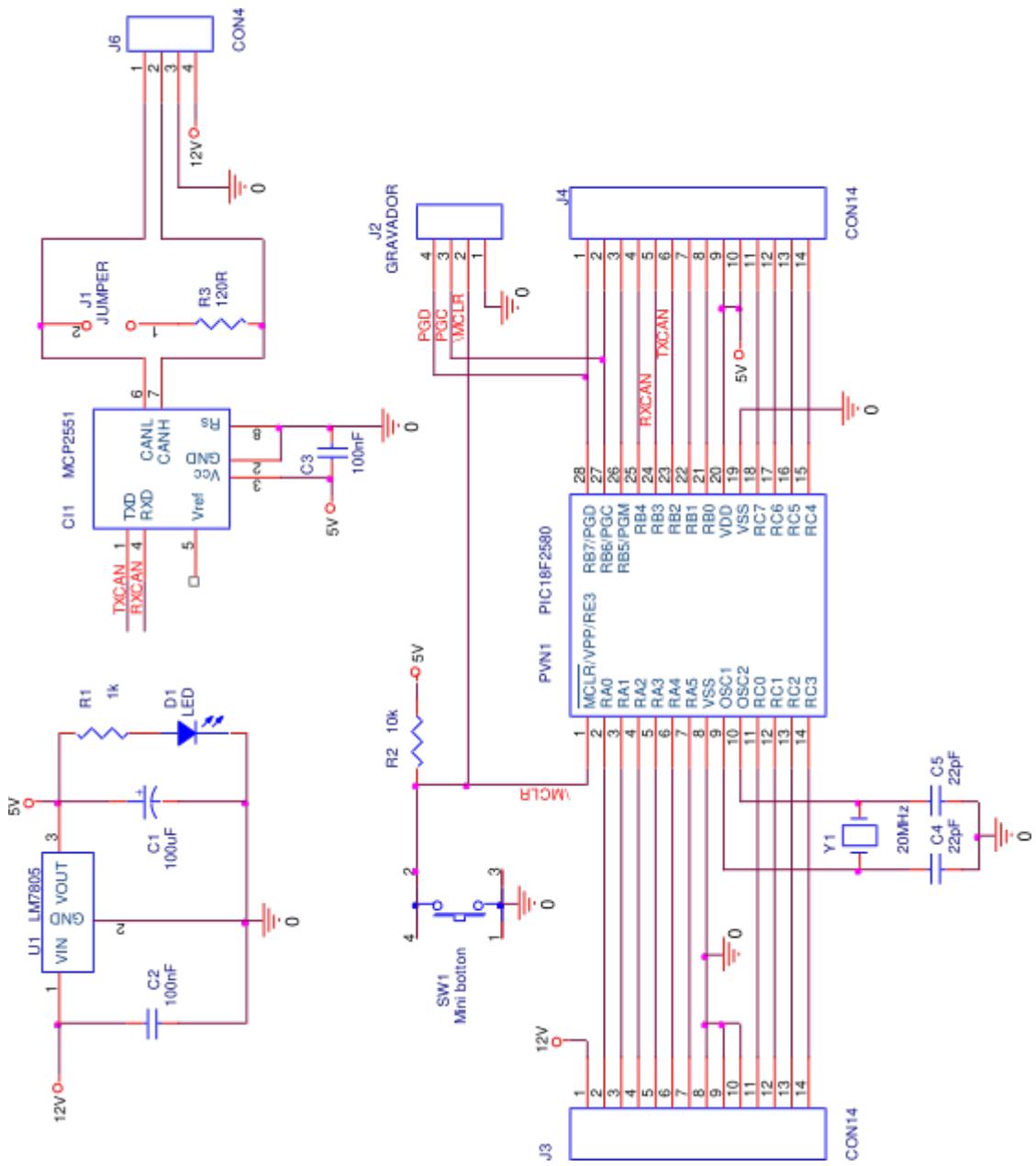
O conjunto de conector J3 e J4 permitem que as placas condicionadoras de sinais sejam acopladas a PPC. Esses conectores além de permitir acesso as portas A,B e C do microcontrolador, permite acesso a 12V por meio do pino 1 do conector J3, a 0V (GND) por meio dos pinos 8,9 e 10 do mesmo conector e a 5V por meio dos pinos 9 e 10 do conector J4.

O diagrama elétrico completo da placa de processamento e comunicação está detalhado na Figura 79.

Figura 78 – Diagrama elétrico do microcontrolador e seus circuitos básicos.



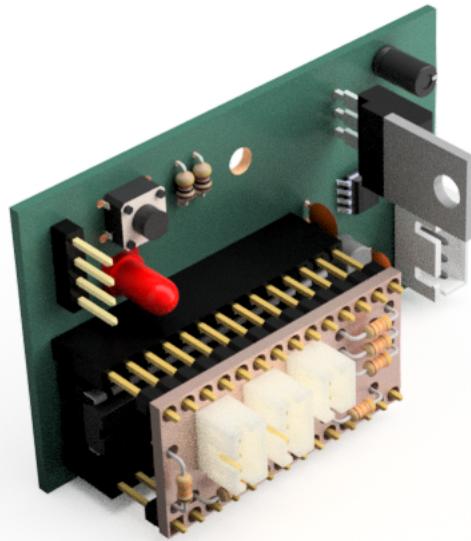
Fonte: O Autor.

Figura 79 – Diagrama elétrico completo da PPC.**Fonte:** O Autor.

ANEXO B – Módulo da estrutura

O módulo da estrutura consiste de uma placa de processamento e comunicação (PPC) e um acoplamento responsável pelo condicionamento dos sinais dos sensores utilizados. A PPC já foi descrita no Anexo A, os detalhes da placa para condicionamento de sinais (PCS) está descrito a seguir. Conforme representado na Figura 80, a PCS é conectada diretamente na PPC, formando o módulo.

Figura 80 – Modelagem das placas do módulo.



Fonte: O Autor.

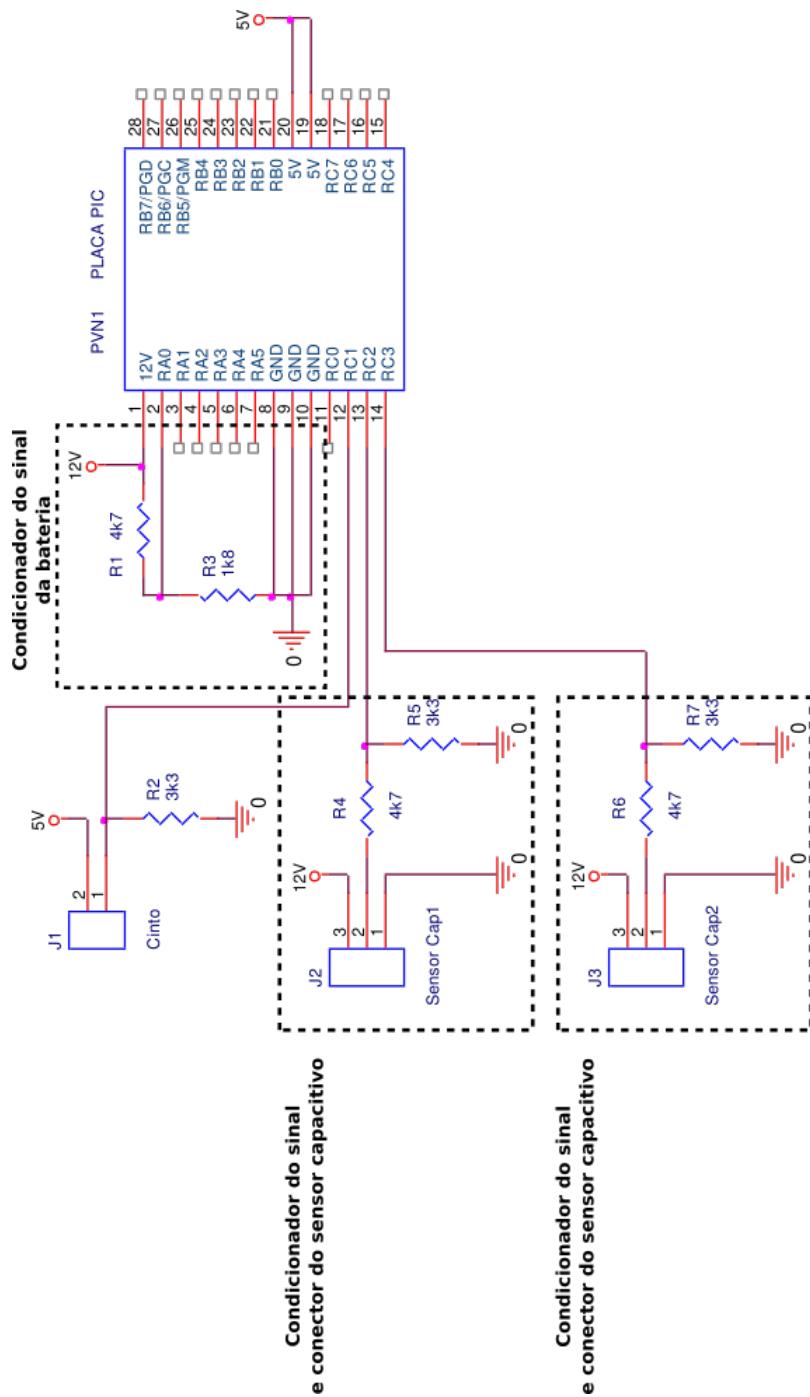
Uma vez que a tensão de operação da bateria (12V) é superior ao aceito pela PPC (5V), o sinal da bateria é reduzido por meio de um condicionador de sinal. O condicionador consiste de um divisor de tensão, que reduz linearmente o sinal de tensão da bateria para valores que a PPC aceita.

O sinal da bateria, depois de condicionado, é lido pelo conversor A/D da porta A0 (pino 2). O PIC18F2580 conta com oito conversores de sinal com capacidade de conversão de 10 bits cada, contudo, a PPC só disponibiliza apenas os seis primeiros.

Os sensores capacitivos são alimentados com 12 V, o sinal do sensor também é superior ao aceito pela PPC, de forma que condicionadores de sinais são utilizados. Os condicionadores dos sinais consistem de divisores de tensão, projetados de maneira

que a tensão máxima do sensor não ultrapasse o limite da PPC. Ambos os conectores e condicionadores de sinais dos sensores estão representados no diagrama da Figura 81. O sinal condicionado de cada sensor capacitivo é lido pela PPC nas entradas digitais RC2 e RC3.

O diagrama elétrico da placa de acoplamento do módulo da estrutura está detalhado na Figura 81. A montagem do módulo da estrutura é feito de acordo com a Figura 80.

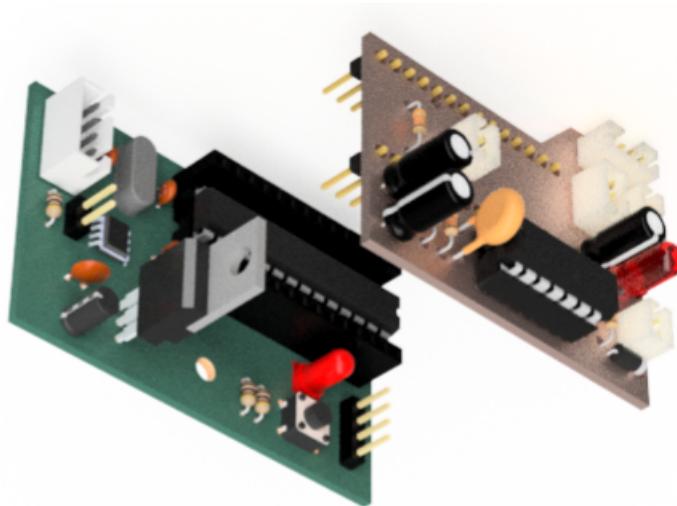
Figura 81 – Esquema elétrico da PCS do módulo.

Fonte: O Autor.

ANEXO C – Módulo do motor

O módulo do motor é composto por uma placa de processamento e comunicação (PPC) e uma placa para condicionamento de sinal (PCS). Essas placas são montadas de acordo com a Figura 82. A PPC está descrita no Anexo A, a seguir serão tratados os detalhes da PCS do motor.

Figura 82 – Modelagem das placas do módulo com a sua respectiva montagem.



Fonte: O Autor.

Para aquisição da rotação motora o módulo utiliza um conversor de frequência em tensão. O sinal elétrico, periódico, da vela de ignição é utilizado como frequência de entrada, sendo a saída do CI uma tensão proporcional ao valor da entrada. O conversor utilizado é um LM2917N configurado de acordo com o diagrama da Figura 83. O diodo D1 (1N4448) protege o CI contra a descarga elétrica da vela. O LED (LED1) indica quando a placa está energizada.

A leitura da tensão de saída é feita pelo conversor A/D da PPC. A PPC, por meio do PIC18F2580, conta com conversores de 10 bits.

A equação que relaciona a tensão de saída (V_{out}) com a frequência de entrada (F_{in}) é dada de acordo com a Equação C.1.

$$V_{out} = f_{in} * V_{cc} * R1 * C1 * K; \quad (\text{C.1})$$

onde

V_{out} : Tensão de saída [V];

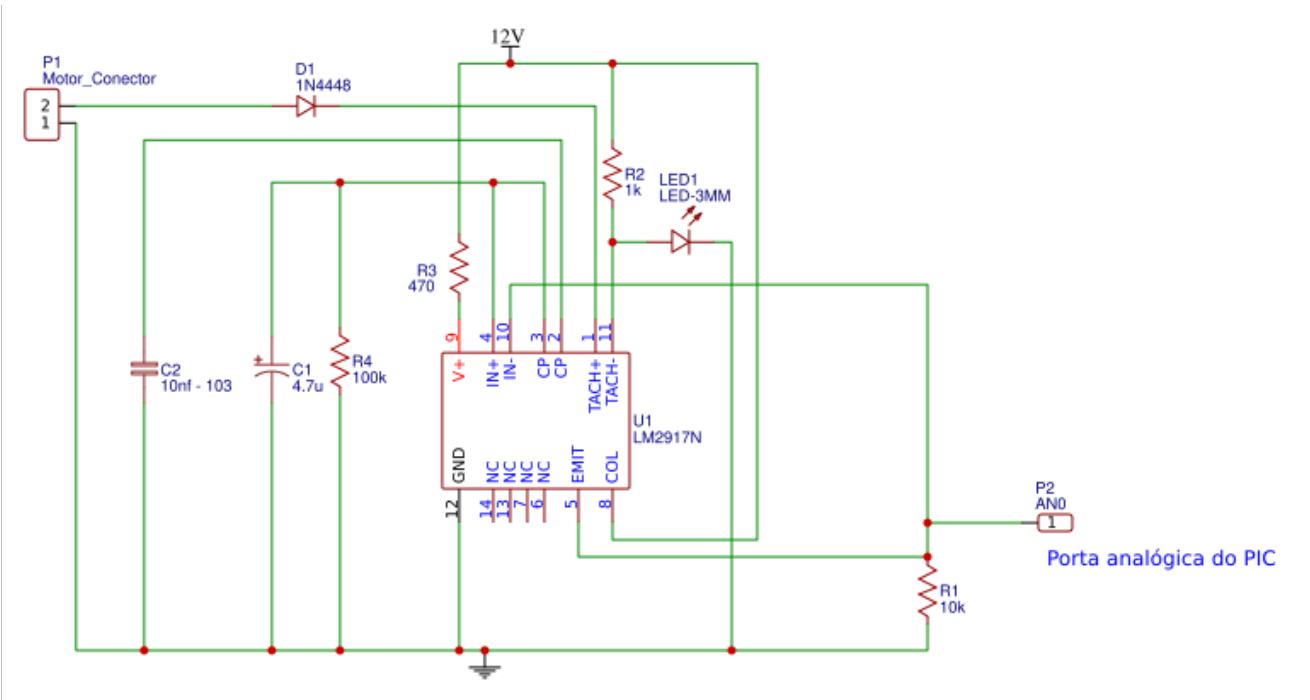
F_{in} : Frequência de entrada [Hz];

$R1$: Resistor conectado ao pino 3 [Ω];

$C1$: Capacitor conectado ao pino 2 [F];

K : Constante para ajuste da leitura.

Figura 83 – Diagrama elétrico do conversor de frequência em tensão.

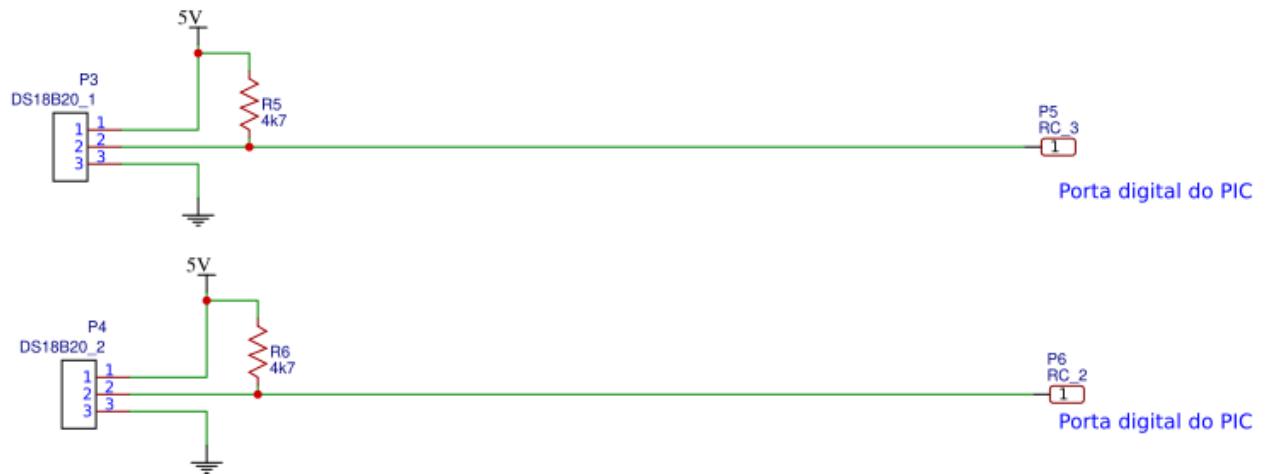


Fonte: O Autor.

Para comunicar com os sensores de temperatura DS18B20, são necessários resistores de *pull up* de valor igual a $4k7$ [Ω]. O diagrama da Figura 84 apresenta o diagrama elétrico dos conectores dos sensores de temperatura.

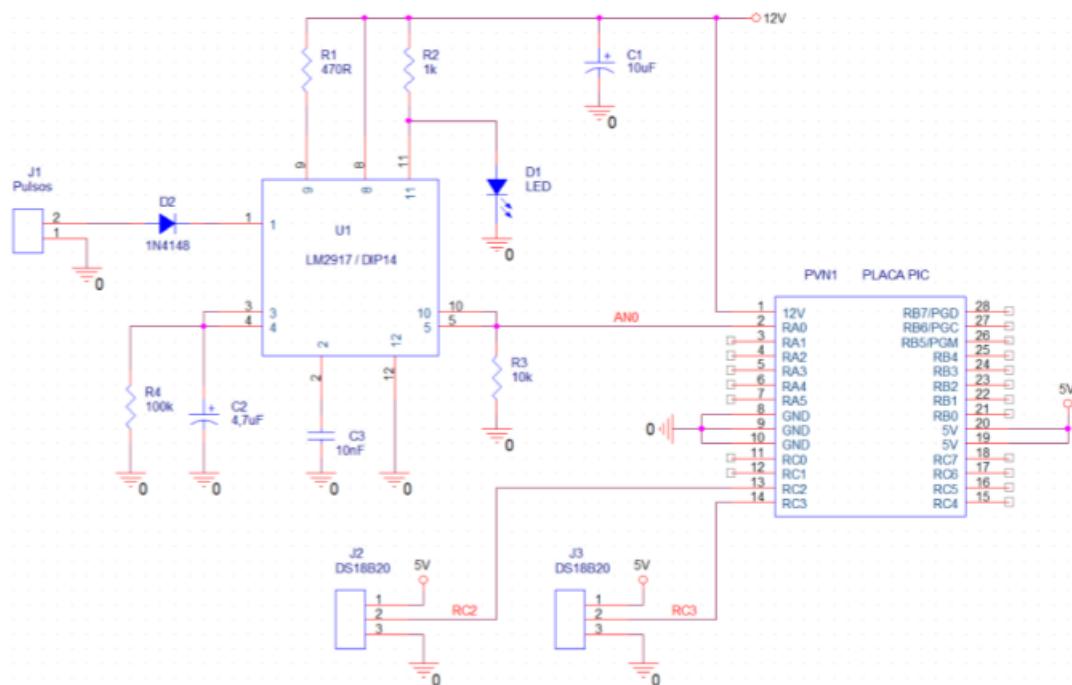
Por fim, o diagrama elétrico completo do módulo fica de acordo com a Figura 85.

Figura 84 – Diagrama elétrico dos conectores dos sensores de temperatura.



Fonte: O Autor.

Figura 85 – Diagrama elétrico do módulo.

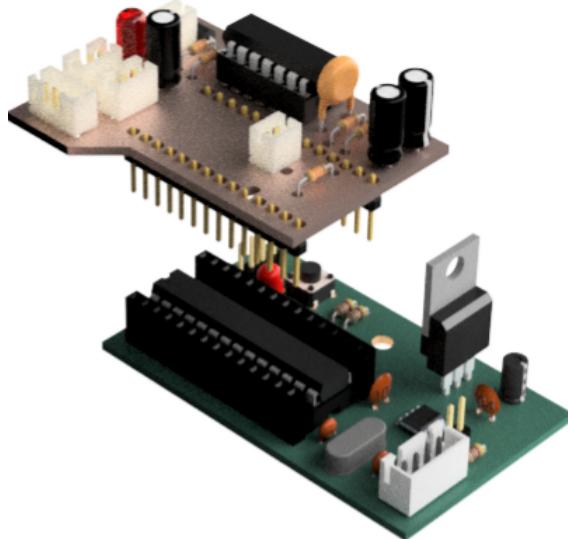


Fonte: O Autor.

ANEXO D – Módulo de localização

O módulo de localização é formado por uma placa de processamento e comunicação (PPC), discutida no Anexo A, e um acoplamento para condicionamento de sinais (PCS). Os detalhes da placa condicionadora serão descritas a seguir. A placa condicionadora de sinal é conectada a placa de processamento e comunicação por meio das barras de pinos, conforme representado na Figura 86.

Figura 86 – Modelagem das placas, assim como sua respectiva montagem.

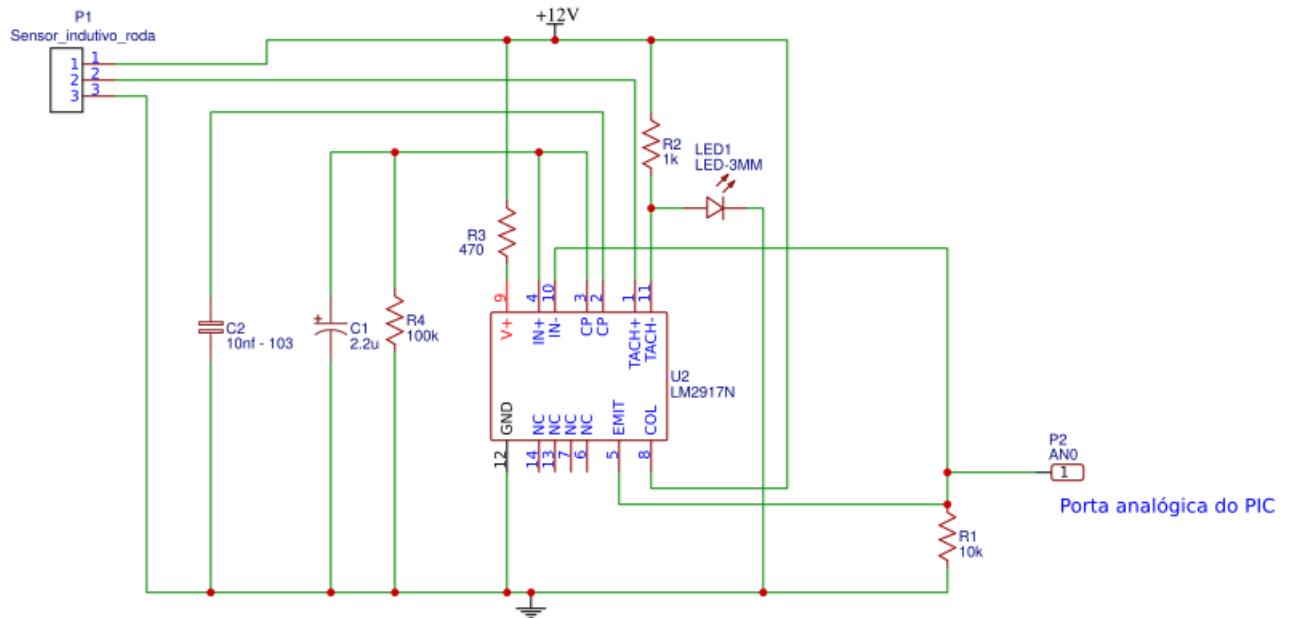


Fonte: O Autor.

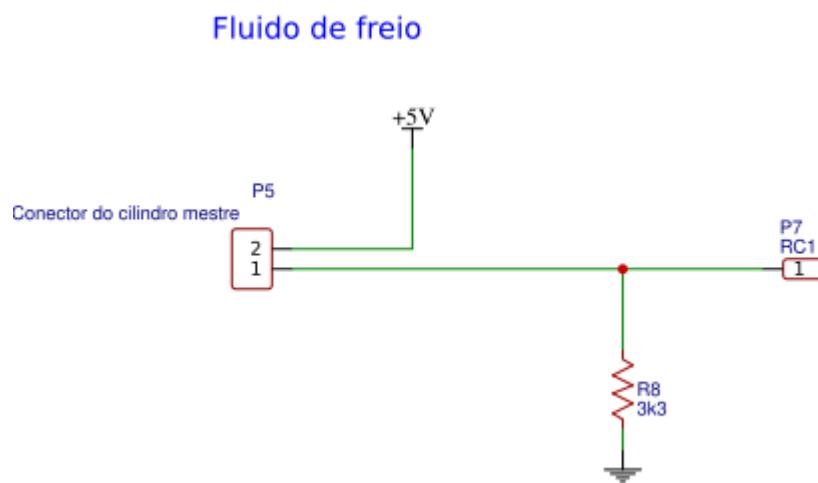
Para aquisição da velocidade, o módulo utiliza um conversor de frequência em tensão (LM2917N) similar ao utilizado no módulo do motor. Todos os detalhes do funcionamento do conversor, bem como a equação de conversão, estão descritos no Anexo C. O CI LM2917N é configurado de acordo com o diagrama elétrico da Figura 87.

O LED1 indica quando a placa está energizada. A tensão de saída é lida pela PPC por meio da porta AN0, onde a frequência de entrada é obtida do sensor por meio do conector P1. O conector P1 também disponibiliza +12V e 0V (GND), para alimentar o sensor indutivo.

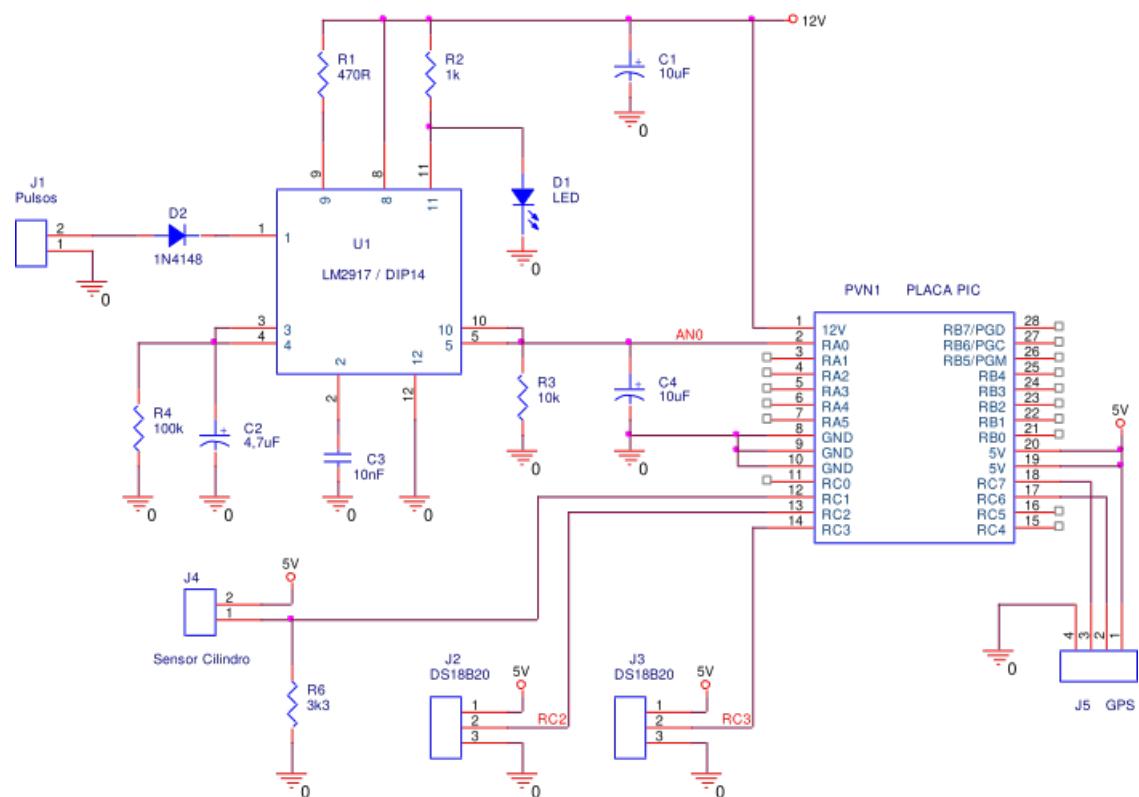
Para avaliar o nível do fluído de freio a PCS conta com o circuito representado na Figura 88. O sinal do sensor, nesse caso a boia de nível, é obtido por meio do conector P5. Quando o sensor está com o circuito aberto, a tensão da porta RC1 da PPC será igual a 0V (GND). Caso o sensor feche o circuito, a tensão da mesma porta fica em 5V, ou nível

Figura 87 – Diagrama elétrico do CI LM2917N do módulo de localização.**Fonte:** O Autor.

lógico 1. O resistor R8 é um resistor de *pull-down* e evita que o sinal da porta RC1 fique flutuando, quando o sensor está com circuito aberto.

Figura 88 – Diagrama elétrico do CI LM2917N do módulo de localização.**Fonte:** O Autor.

O diagrama elétrico, completo, do acoplamento fica de acordo com a Figura 89. O circuito conta ainda com os conectores J2, J3 e J5, que não foram utilizados.

Figura 89 – Diagrama elétrico do módulo.**Fonte:** O Autor.

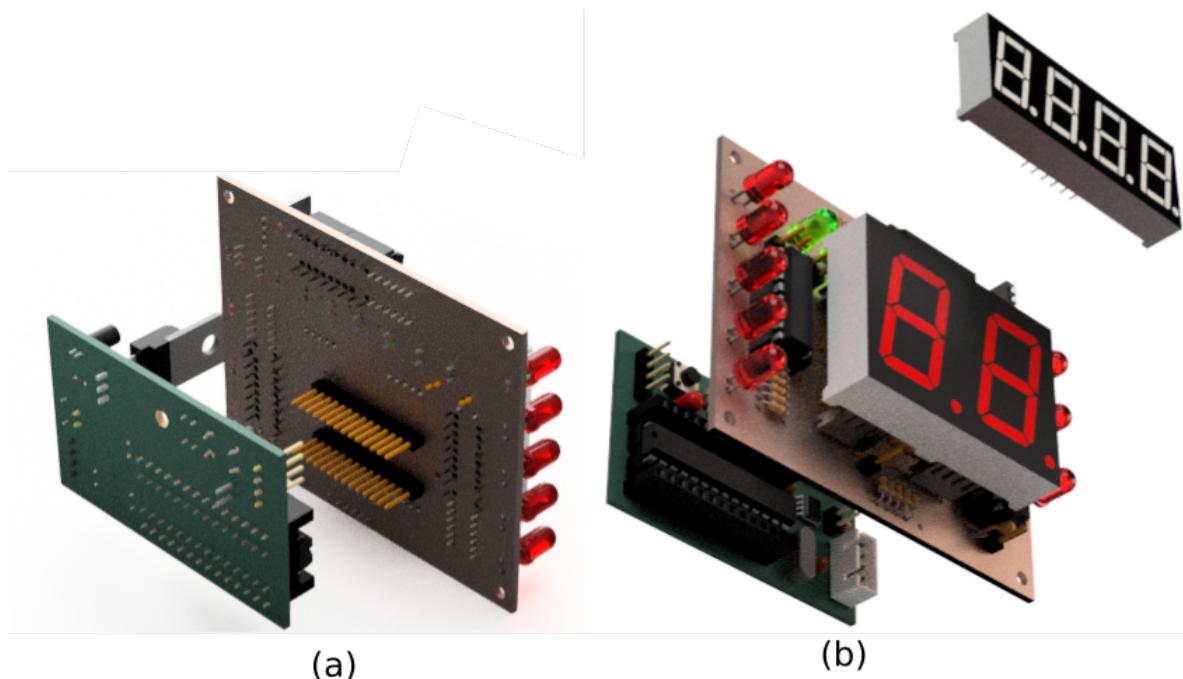
ANEXO E – Módulo do Painel de instrumentos

O módulo do painel de instrumentos, assim como todos os módulos, contém uma placa de processamento e comunicação (PPC) e uma placa condicionadora de sinal (PCS). Uma vez que os detalhes da PPC já foram tratados no Anexo A, a seguir será detalhado a PCS.

A PCS é conectada à PPC por meio de duas barras de pinos, conforme representando na Figura 90 (a). Ainda em relação a PCS, os LEDS são fixados diretamente na placa e os displays são acoplados por meio de *sockets*.

Embora o PIC18F2580 (INC, 2009) conte com 25 portas para comunicação digital, a PPC só permite o uso de 18 dessas portas. Esse número, de portas, é insuficiente para o controle dos LEDS do painel. Para contornar o limite de portas disponíveis pela PPC, a PCS utiliza três registradores de deslocamento para controlar o conta-giros e o *display* do velocímetro, o *display* do horímetro conta com um *driver* próprio.

Figura 90 – Modelagens do painel de instrumentos. (a) Vista em perspectiva traseira.
(b) Vista em perspectiva frontal.



Fonte: O Autor.

O conta-giros é composto por 7 LEDS, todos os leds são controlados por meio do

CI 74HC595. Conforme representado no diagrama elétrico da Figura 91: o CI é alimentado com +5V; Todos os LEDS utilizam resistor de 330 [Ω] para limitar a corrente.

O controle do CI 74HC595 (INSTRUMENTS, 2015) é feito por meio das portas: *SH_CP* (*shift register clock input*) no pino 11, *DS* (*serial data input*) no pino 14 e o *ST_CP* (*storage register clock input*) no pino 12.

As funções dos pinos 13 e 10, *reset* e ativação das portas de saída, respectivamente, não são controladas pela PPC.

Para enviar um byte de informação, ou seja, ativar os LEDs do conta-giro, a PPC primeiro coloca em nível baixo (0V) na porta RB5 (pino 11), coloca o bit desejado na porta RB7 (pino 14), coloca a porta RB5 em nível alto (5 V). Após repetir o procedimento anterior 8 vezes, a PPC envia um pulso na porta 12.

Para acender os LEDs do conta giro, a PPC classifica a rotação motora e depois envia um byte de informação para o registrador. A relação entre rotação motora, número de LEDs acesso e o byte enviado para o registrador é dado de acordo com a Tabela 23.

Os leds são ativados da esquerda para direita, logo, quando o primeiro led é ativado significa que o primeiro led mais a esquerda, e assim na sequência.

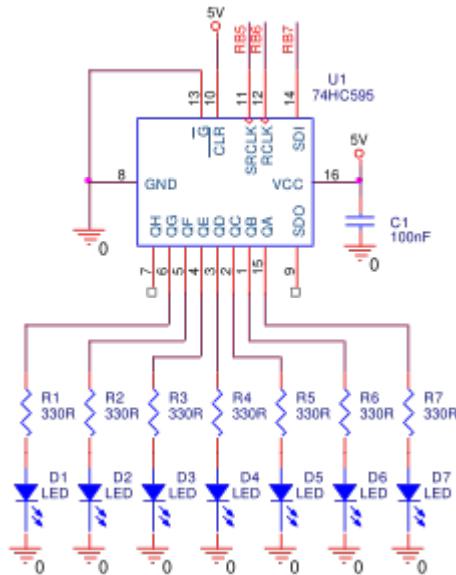
Tabela 23 – Relação entre rotação motora, led acesso e leds ativos.

Rotação motora	Leds acessos	Portas ativas (QH QG QF QE QD QC QB QA)
<1300	0	0,0,0,0,0,0,0,0
<1625	1	0,0,0,0,0,0,0,1
<1950	2	0,0,0,0,0,0,1,1
<2275	3	0,0,0,0,0,1,1,1
<2500	4	0,0,0,0,1,1,1,1
<2700	5	0,0,0,1,1,1,1,1
<3000	6	0,0,1,1,1,1,1,1
<3500	7	0,1,1,1,1,1,1,1

O velocímetro é composto por dois *displays*, para controlar ambos os *displays* a PCS conta com outros dois deslocadores 74HC595, conforme representando no diagrama da Figura 92.

Ambos os *displays* utilizados são de 7 segmentos com ânodo comum. O ponto do *display* não tem utilidade, logo está desativado e a PPC não controla. Com o conjunto é possível exibir o intervalo de 0 até 99.

Para exibir a velocidade, a PPC primeiro separa os números por meio das equações

Figura 91 – Diagrama elétrico do conta-giros do painel de instrumentos.

Fonte: O Autor.

E.1.

$$\begin{aligned} V_1 &= V / 10 \\ V_2 &= V \% 10 \end{aligned} \tag{E.1}$$

em que:

V : Velocidade do veículo. [km/h]

V_1 : Primeiro dígito da velocidade.

V_2 : Segundo dígito da velocidade.

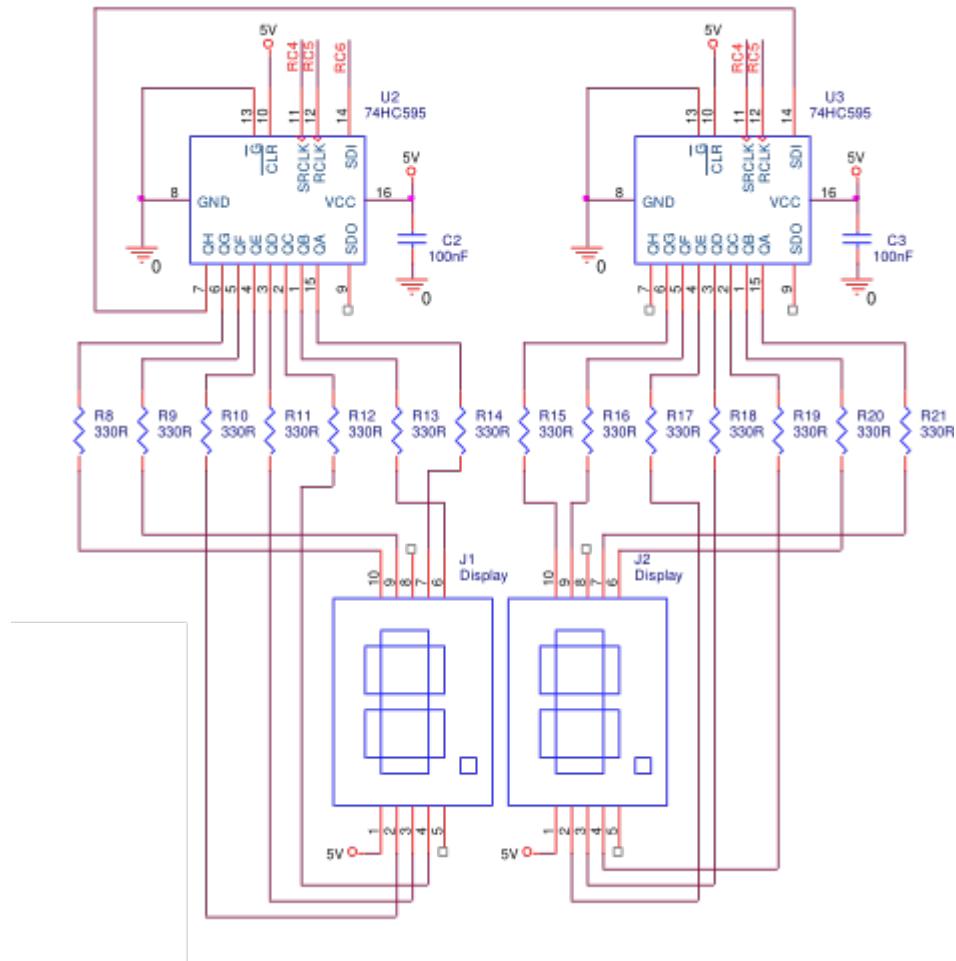
Uma vez que a PPC tem a informação que cada *display* deve exibir, o segundo passo é converter esse valor numérico nos segmentos ativos de cada *display*. A PPC utiliza a Tabela 24 para relacionar o valor de V_1 e V_2 com as respectivas portas ativas dos deslocadores.

Por último, a PPC envia para o registrador de deslocamento os bits de V_2 e na sequência os bits de V_1 . O procedimento para enviar é o mesmo do conta giros, apenas com as portas utilizadas diferentes.

Os leds do sistema de advertência e diagnósticos são controlados diretamente pela PPC, conforme representando no diagrama elétrico da Figura 93. Todos os LEDs contam com um resistor de 330 $[\Omega]$ para limitar a corrente.

Tabela 24 – Relação entre o número exibido e os bits ativos.

Número	Portas ativas								
	(QH	QG	QF	QE	QD	QC	QB	QA)	
0	1	1	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	1	
2	1	0	1	0	0	1	0	0	
3	1	0	1	1	0	0	0	0	
4	1	0	0	1	1	0	0	1	
5	1	0	0	1	0	0	1	0	
6	1	0	0	0	0	0	1	0	
7	1	1	1	1	1	0	0	0	
8	1	0	0	0	0	0	0	0	
9	1	0	0	1	0	0	0	0	

Figura 92 – Diagrama elétrico do velocímetro do painel de instrumentos.**Fonte:** O Autor.

Para ativar um LED, a PPC deve configurar a porta do led em nível lógico 1 ou seja, um nível de tensão de 5 [V] na porta.

A relação entre a porta utilizada pela PPC com o indicador de alerta/diagnóstico é definido de acordo com as Tabelas 25 e 26, onde a contagem dos leds é feita de cima para baixo.

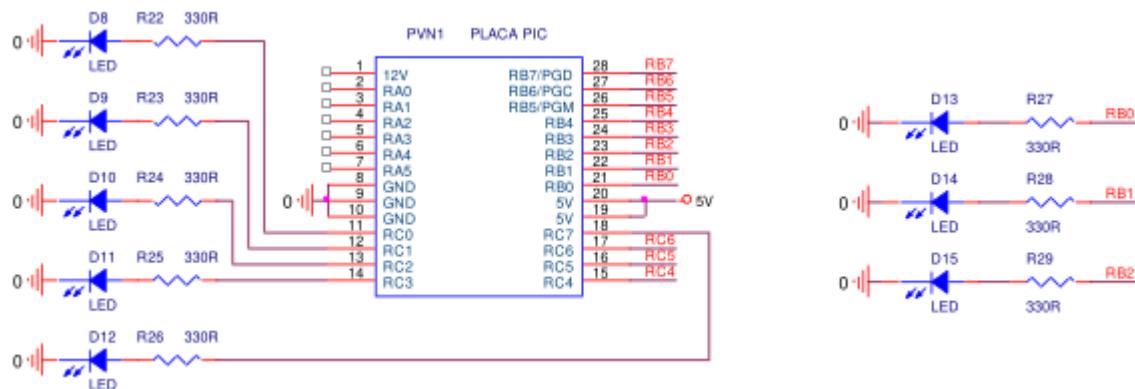
Tabela 25 – Relação entre os indicadores do lado esquerdo do painel e a porta utilizada para sua ativação.

Indicador	Porta do PIC (Pino)	Função
1º	RC0(11)	Alerta da telemetria
2º	RC1(12)	Nível de bateria
3º	RC2(13)	Nível do fluído de freio
4º	RC3(14)	Temperatura do óleo
4º	RC7(18)	Temperatura da transmissão

Tabela 26 – Relação entre os indicadores do lado direito do painel e a porta utilizada para sua ativação.

Indicador	Porta do PIC (Pino)	Função
1º	RB0(21)	Estado da eletrônica
2º	RB1(22)	Nível de combustível alto
3º	RB2(23)	Nível de combustível baixo

Figura 93 – Diagrama elétrico do sistema de alerta e diagnóstico do painel de instrumentos.



Fonte: O Autor.

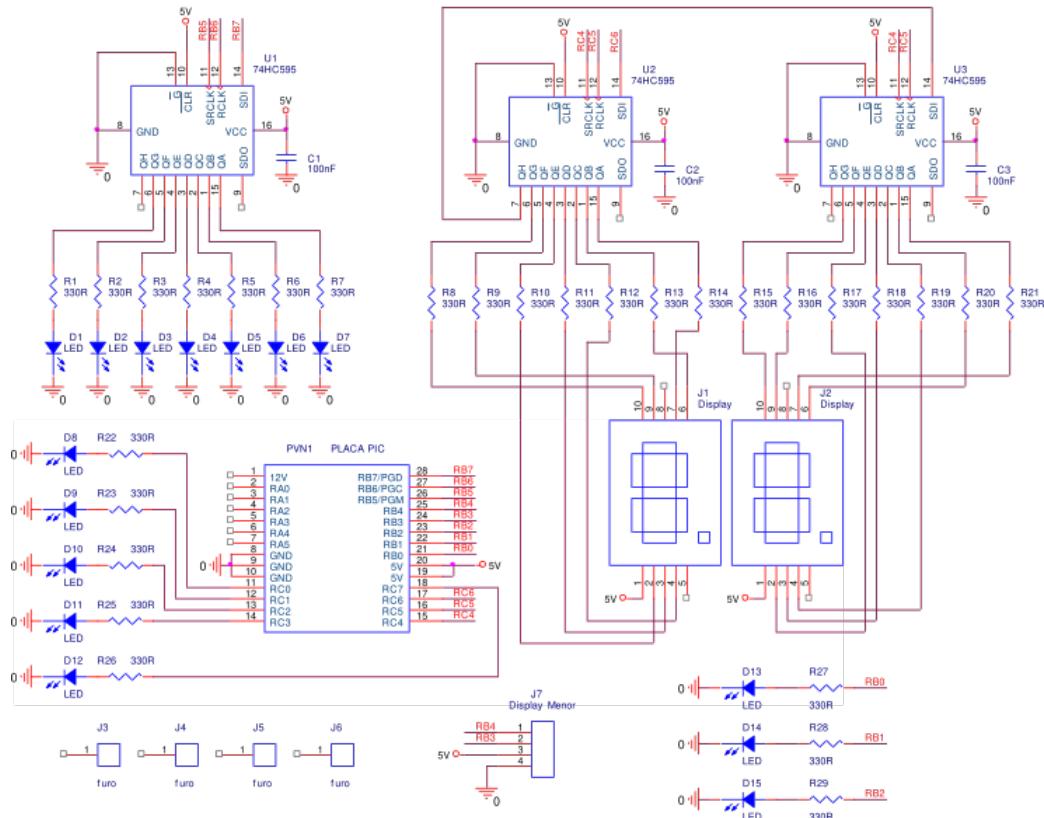
Os quatro displays de 7 segmentos utilizados pelo horímetro são acoplados por meio de um conector de quatro vias, conforme representando no diagrama da Figura 94.

O conector J7, além de conectar as portas RB4 e RB3, também energiza o *display* com +5V e GND.

O *display* do horímetro conta com o *driver* TM1637 (ELECTRONICS,). Dessa forma, a PPC implementa o protocolo de comunicação desse *driver*.

O diagrama elétrico completo do acoplamento está detalhado na Figura 94.

Figura 94 – Esquema elétrico da PCS do painel de instrumentos.



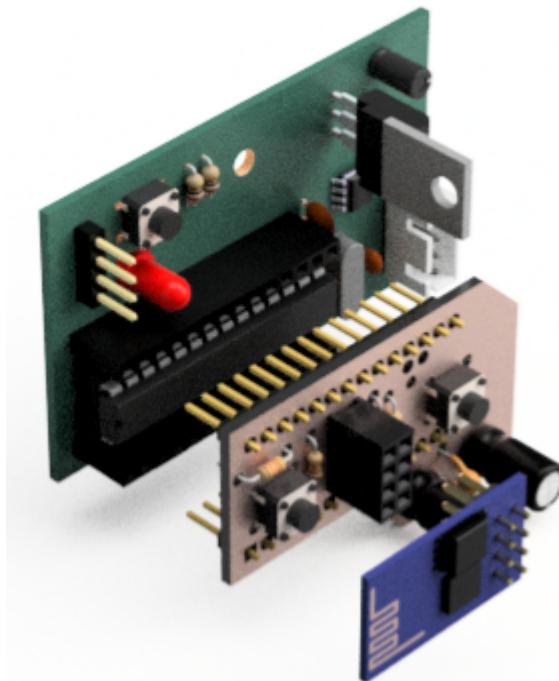
Fonte: O Autor.

ANEXO F – Módulo da Telemetria

O módulo de Telemetria é formado por uma placa de processamento e comunicação (PPC), detalhada no Anexo A, uma placa condicionadora de sinal (PCS), detalhada a seguir e um transmissor ESP-01, também detalhado a seguir.

O transmissor é acoplado na PCS por meio de um *socket*. A PCS, é conectada na PPC por meio de duas barras de pinos. Ambos os conectores estão representados na modelagem do módulo da Figura 95.

Figura 95 – Esquema elétrico da PCS. (a) Diagrama esquemático. (b) Desenho da placa. (c) Montagem do módulo.

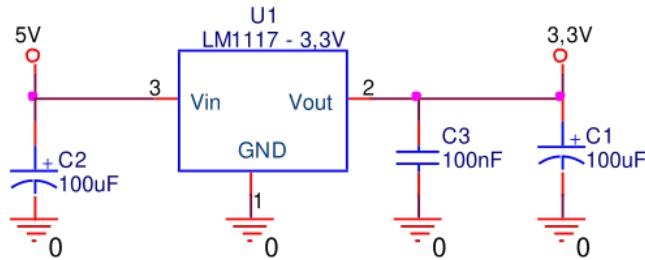


Fonte: O Autor.

Em consequência da tensão de operação do transmissor ser 3.3 [V], a PCS conta com um regulador de tensão LM1117 (INSTRUMENTS, a). Conforme representado no diagrama elétrico da Figura 96, o regulador é alimentado com 5 [V] no pino 3 (Vin), o pino 1 é conectado ao GND (0 [V]) e o pino 2 (Vout) alimenta o transmissor com a tensão necessária.

Ainda de acordo com o diagrama da Figura 96, o regulador LM1117 conta com os capacitores C1, C2 e C3 para filtragem e desacoplamento DC do sinal, melhorando assim a estabilidade da alimentação.

Figura 96 – Diagrama elétrico do regulador de tensão do transmissor.



Fonte: O Autor.

O transmissor é conectado a PPC de acordo com o diagrama da Figura 97. O pino 1 do transmissor, TXD, é conectado a porta RX da comunicação serial do PIC, pino 18. O pino 5, RST, conta com um circuito de *reset*. O pino 6 do transmissor, GPIO0, conta um botão utilizado para entrar no modo de gravação de *firmware*.

O connector J1 permite realizar a atualização do *firmware* do transmissor, e também funciona como uma porta de escuta durante a execução. Essa porta de escuta permite acompanhar toda comunicação entre o transmissor e a PPC, permitindo realizar *debug*, caso necessário.

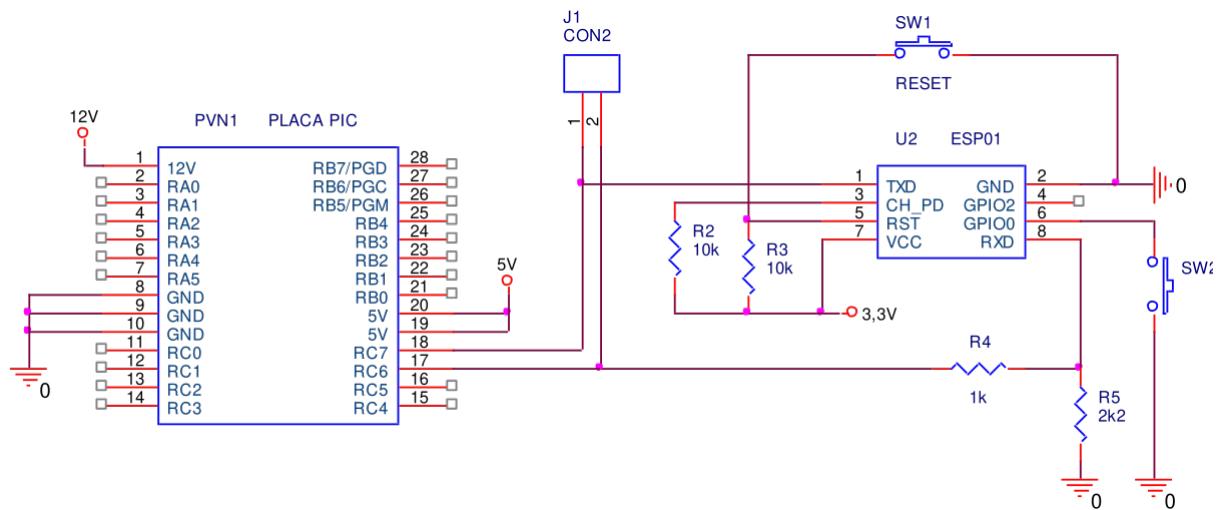
A tensão de operação da PPC, ou seja do PIC18F, é de 5 [V], no entanto, a tensão de operação do transmissor é de 3.3 [V]. Logo, para proteger o transmissor, o pino 8, RXD, conta com um divisor de tensão. O divisor é composto pelos resistores R4 e R5, no diagrama elétrico da Figura 97.

O diagrama elétrico completo da PCS está detalhado na Figura 98.

O cartão microSD conta com um circuito próprio condicionado em um módulo. Esse circuito é responsável por converter a tensão de comunicação entre a PPC e o cartão microSD. Para tanto, conta com um CI para *level shifter* e regulador de tensão. O *level shifter* realiza a conversão de tensão da comunicação automaticamente, logo a PPC pode comunicar com o microSD diretamente.

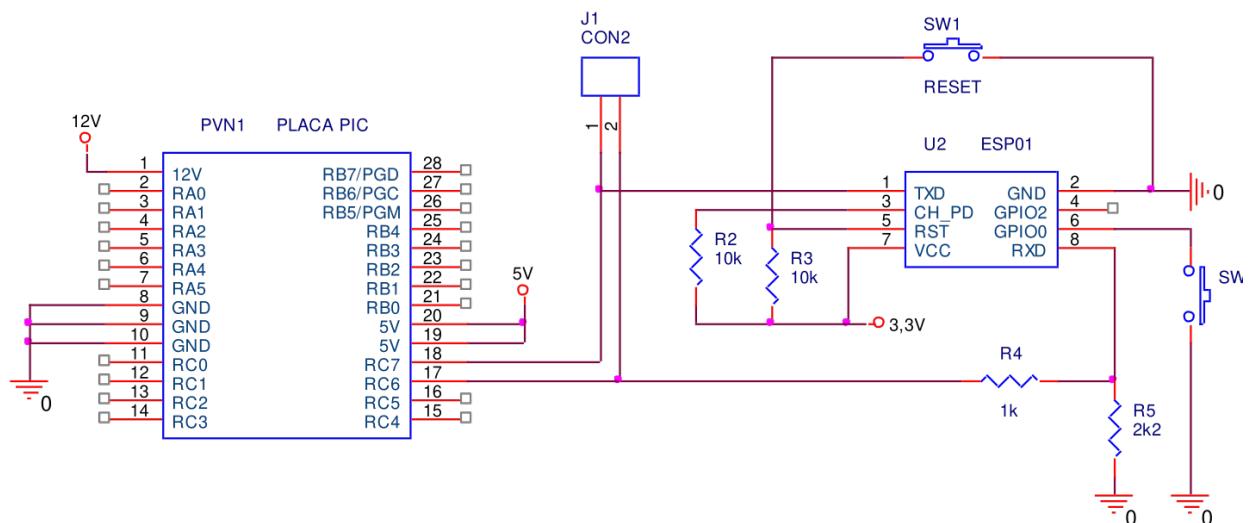
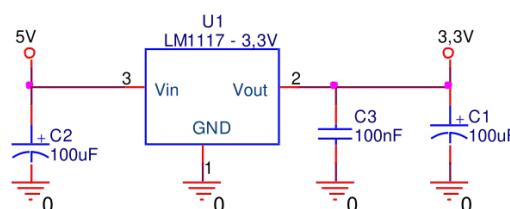
O condicionador do microSD é acoplado a PCS diretamente na barra de pinos, de forma que a PCS não utiliza nenhum conector, conforme representado na Figura 99.

Figura 97 – Diagrama elétrico da conexão do transmissor ESP01.



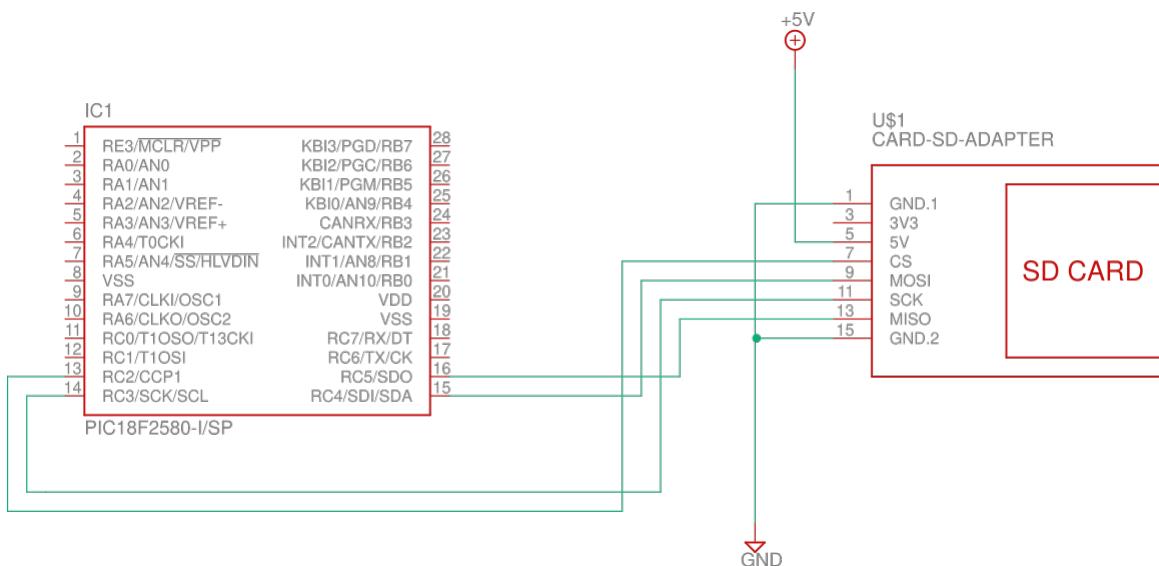
Fonte: O Autor.

Figura 98 – Esquema elétrico da PCS.



Fonte: O Autor.

Figura 99 – Esquema elétrico da conexão do adaptador de cartão microSD.



Fonte: O Autor.