

**CENTRO PAULA SOUZA
FATEC SANTO ANDRÉ
Tecnologia em Eletrônica Automotiva**

**Jean Merino Choma
Wesley dos Santos Souza**

INJEÇÃO ELETRÔNICA COM RASPBERRY PI

**Santo André
2022**

Jean Merino Choma
Wesley dos Santos Souza

INJEÇÃO ELETRÔNICA COM RASPBERRY PI

Trabalho de Conclusão de Curso apresentado ao Curso de Tecnologia em Eletrônica Automotiva da FATEC Santo André, como requisito parcial para obtenção do título de Tecnólogo em Eletrônica Automotiva.

Orientador: Prof. Dr. Edson Caoru Kitani

Santo André
2022

FICHA CATALOGRÁFICA

C454i

Choma, Jean Merino

Injeção eletrônica com Raspberry PI / Jean Merino Choma,
Wesley dos Santos Souza. - Santo André, 2022. – 54f: il.

Trabalho de Conclusão de Curso – FATEC Santo André.
Curso de Tecnologia em Eletrônica Automotiva, 2022.

Orientador: Prof. Dr. Edson Caoru Kitani

1. Eletrônica. 2. Injeção eletrônica. 3. Automóveis. 4. Sensores.
5. Tecnologia. 6. Microcontrolador. 7. Desenvolvimento. 8.
Raspberry PI. 9. PIC18F4550. 10. Motores de combustão
interna. I. Souza, Wesley dos Santos. II. Injeção eletrônica com
Raspberry PI.

629.2

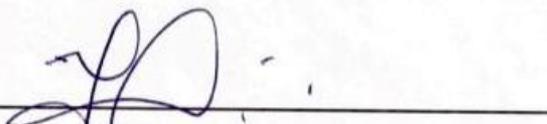
LISTA DE PRESENÇA

Santo André, 14 de dezembro de 2022.

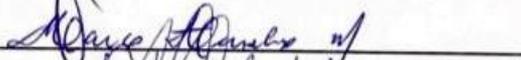
LISTA DE PRESENÇA REFERENTE À APRESENTAÇÃO DO
TRABALHO DE CONCLUSÃO DE CURSO COM O TEMA:
“INJEÇÃO ELETRÔNICA COM RASPBERRY PI” DOS ALUNOS
DO 6º SEMESTRE DESTA U.E.

BANCA

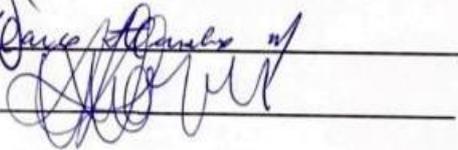
PRESIDENTE:
PROFº EDSON CAORU KITANI



MEMBROS:
PROF. MARCO AURÉLIO FRÓES



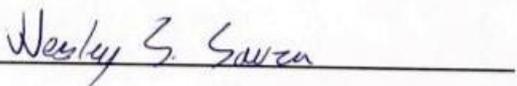
PROF. WESLEY MEDEIROS TORRES

**ALUNO:**

JEAN MERINO CHOMA



WESLEY DOS SANTOS SOUZA



DEDICATÓRIA

É com muito orgulho e felicidade que dedico este trabalho primeiramente a Deus que nos deu a sabedoria e o poder de questionar, também dedico aos meus pais que ainda são responsáveis por me proporcionar a melhor base que eles podem oferecer, dedico para minhas irmãs que estiveram me apoiando como também aos meus avós. O desenvolvimento do trabalho é de especial dedicação para cada pessoa aqui mencionada por ter seu papel importante na minha evolução e desenvolvimento incessante.

AGRADECIMENTOS

Este trabalho só foi possível com a ajuda das melhores pessoas presentes em minha vida. Agradeço a minha família por apoiar o meu grande interesse na área. Cada conquista que eu tenho, sei o quanto vocês me ajudaram. Agradeço aos meus amigos por compreender a minha ausência e por me ajudarem nesse trabalho. Agradeço a FATEC Santo André, uma instituição de ensino excelente, que, no qual me ensinou muito. Agradeço aos professores e funcionários da FATEC pela paciência e dedicação, principalmente ao meu orientador. Obrigado a todos que estiveram comigo nesse trajeto.

EPÍGRAFE

“A maioria dos homens e mulheres, por nascimento ou natureza, não tem meios de avançar na riqueza ou poder, mas todos têm a capacidade de avançar no conhecimento.”

(PITÁGORAS)

RESUMO

O presente trabalho projeta uma injeção eletrônica automotiva e mostra os procedimentos e assuntos abordados para realização de testes práticos. A injeção eletrônica é construída com Raspberry Pi 3 B junto ao PIC 18F4550, uma versão com *interface* e interação que traz facilidade ao usuário para possíveis edições de calibração e leitura dos sensores comportados no veículo. Cabe ressaltar que a importância do projeto que tem código aberto, pode ser consultada por discentes e docentes no qual complementa os assuntos abordados em aulas ministradas junto com práticas que envolvem motores de combustão interna, mostrando ao referido consultor os *softwares*, *hardwares* e cálculos utilizados e dar liberdade de acesso para editá-lo e testá-lo na prática os estudos aprendidos. O conteúdo que aqui está, tem resultados obtidos em testes práticos, junto com docentes, com gráficos e imagens de fácil interpretação, uma vez que, pode ser entendido futuramente como material didático fundamental para cursos de Eletrônica Automotiva e similares.

Palavras-chave: Injeção eletrônica. Raspberry Pi. Raspberry Pi 3 B. PIC 18F4550. Motores de combustão interna.

ABSTRACT

The present current work design projects an automotive electronic injection and shows the procedures and subjects addressed for conducting practical tests. The electronic injection is built with Raspberry Pi next to the PIC 18F4550, a version with interface and interaction that brings ease to the user for possible calibration and reading editions of the sensors behaved in the vehicle. It is worth mentioning that the importance of the project that has open source, can be consulted by students and teachers in which complements the subjects addressed in classes administered along with practices involving internal combustion engines, showing the said consultant programs, hardware and calculations used and give freedom to edit it and test it in practice the studies learned. The content that is here, has results obtained in practical tests, together with teachers, with graphics and images of easy interpretation, since it can be understood in the future as a fundamental didactic material for automotive electronics courses and the like.

Keywords: *Electronic injection. Raspberry Pi. PIC 18F4550. Internal combustion engines.*

LISTA DE FIGURAS

FIGURA 1 - MEGASQUIRT-II (MC9S12C64)	19
FIGURA 2 - MEGASQUIRT-I: (68HC908).....	19
FIGURA 3 - PLACA PRINCIPAL MEGASQUIRT DIY	20
FIGURA 4 - SOFTWARE TUNERSTUDIOMS.....	21
FIGURA 5 - FT600 SFI.....	21
FIGURA 6 - PLACA SPEEDUINO V0.4.....	23
FIGURA 7 - ARDUINO MEGA 2560 REV3	23
FIGURA 8 – DIAGRAMA DA ECU DE INJEÇÃO ELETRÔNICA.....	26
FIGURA 9 - PLACA DE ENSAIO DE CIRCUITOS SIMULADORES E CONDICIONADORES COM A VISTA SUPERIOR.....	27
FIGURA 10 - PLACA DE ENSAIO DE CIRCUITOS SIMULADORES E CONDICIONADORES COM VISTA FRONTAL	27
FIGURA 11 - CIRCUITO CONDICIONADOR PIP	28
FIGURA 12 - ESTRUTURA ELÉTRICA INTERNA DO CI LM339.....	29
FIGURA 13 - CIRCUITO CONDICIONADOR DO SENSOR MAP	30
FIGURA 14 - CIRCUITO CONDICIONADOR DO SENSOR DE VELOCIDADE	30
FIGURA 15 - CIRCUITO CONDICIONADOR LINHA 15	31
FIGURA 16 - CIRCUITO CONDICIONADOR TPS	31
FIGURA 17 - ESTRUTURA ELÉTRICA INTERNA DO CI LM358.....	32
FIGURA 18 - CIRCUITO CONDICIONADOR DO SENSOR ECT	32
FIGURA 19 - CIRCUITO CONDICIONADOR DO SENSOR ACT	33
FIGURA 20 - CIRCUITO CONDICIONADOR TENSÃO DA BATERIA	33
FIGURA 21 - CONECTOR C60	34
FIGURA 22 - PINAGEM C60	35
FIGURA 23 - CIRCUITO <i>DRIVE</i> DE MARCHA LENTA.....	37
FIGURA 24 - DIAGRAMA EM BLOCOS CI L298	38
FIGURA 25 - CIRCUITO <i>DRIVE</i> DA VÁLVULA INJETORA.....	38
FIGURA 26 - CIRCUITO <i>DRIVE</i> DO CANISTER.....	39
FIGURA 27 - CIRCUITO <i>DRIVE</i> DA BOMBA DE COMBUSTÍVEL.....	39
FIGURA 28 - CIRCUITO <i>DRIVE</i> BOBINA DE IGNIÇÃO	40
FIGURA 29 - PINAGEM DO PIC18F4550.....	41
FIGURA 30 - PINAGEM E CONEXÕES DO RASPBERRY PI.....	42

FIGURA 31 - FLUXOGRAMA DA LEITURA DO SENSOR ACT.....	43
FIGURA 32 - FLUXOGRAMA DE LEITURA DO SENSOR TPS	44
FIGURA 33 - FLUXOGRAMA DO SENSOR ECT.....	45
FIGURA 34 - FLUXOGRAMA DE LEITURA DA TENSÃO DE BATERIA	45
FIGURA 35 - FLUXOGRAMA DO PROGRAMA PRINCIPAL.....	46
FIGURA 36 - FLUXOGRAMA ESCALONADOR	47
FIGURA 37 - FLUXOGRAMA DA COMUNICAÇÃO SPI	48
FIGURA 38 - <i>INTERFACE</i> DO MAPA DO TEMPO DE INJEÇÃO DE COMBUSTÍVEL	49
FIGURA 39 - <i>INTERFACE</i> DOS GRÁFICOS DOS SENSORES	50
FIGURA 40 - TESTE DA ECU COM OS SINAIS SIMULADOS DO VEÍCULO	52

LISTA DE ABREVIATURAS

ACT	<i>Air Charge Temperature</i>
Amp-Op	Amplificador Operacional
CETESB	Companhia de Tecnologia de Saneamento Ambiental
CI	Circuito Integrado
DIP	<i>Dual In-line Package</i>
DIY	<i>Do It Yourself</i>
ECT	<i>Engine Coolant Temperature</i>
ECU	<i>Eletronic Control Unit</i>
EFI	<i>Eletronic Fuel Injection</i>
HDMI	<i>High-Definition Multimedia Interface</i>
Hz	Hertz
IDE	<i>Integrated Development Environment</i>
MAP	<i>Manifold Absolute Pressure</i>
MCI	Motor de Combustão Interna
MISO	<i>Master In Slave Out</i>
MOSFET	<i>Metal Oxide Semiconductor Field Effect Transistor</i>
MOSI	<i>Master Out Slave In</i>
NTC	<i>Negative Temperature Coefficient</i>
PCI	Placa de Circuito Impresso
PIP	<i>Pick up Ignition Profile</i>
PROCONVE	Programa de Controle da Poluição do Ar por Veículos Automotores
RAM	<i>Random-Access Memory</i>
SO	Sistema Operacional
SPI	<i>Serial Peripheral Interface</i>
SS	<i>Slave Select</i>
TCC	Trabalho de Conclusão de Curso
TPS	<i>Throttle Position Sensor</i>
V	Volts

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Motivação	16
1.2	Objetivo	16
1.3	Resultados Esperados	17
2	REVISÃO BIBLIOGRÁFICA	18
2.1	MegaSquirt	18
2.1.1	<i>Hardware</i>	18
2.1.2	<i>Software</i>	20
2.2	Fuel Tech	21
2.3	Speeduino	22
3	METODOLOGIA DO PROJETO	25
3.1	Teoria Sobre o Projeto	25
3.2	Descrição do Projeto	33
3.2.1	<i>Hardware</i> do Projeto	36
3.2.2	<i>Software</i> do Projeto no PIC	43
3.2.3	<i>Software</i> do Projeto no Raspberry Pi	49
3.3	Detalhes Finais e Comentários do Projeto	50
4	TESTES E ANÁLISE DOS RESULTADOS	51
5	CONCLUSÃO	53
5.1	Propostas Futuras	54
	REFERÊNCIAS	55

1 INTRODUÇÃO

O desenvolvimento da humanidade tem princípios de realizar tarefas por meios próprios na utilização do corpo como espécie de ferramenta para sobrevivência, em geral, foi percebendo a necessidade de criar ferramentas por questão de conforto e tempo. O ato de criar, faz com que os humanos se diferenciam e se destacam sobre os outros seres vivos, sendo assim, obtendo o poder de criar ferramentas dando o início as invenções mais importantes para história que atualmente utilizamos, mesmo depois de séculos. Uma invenção que se destaca, no qual incentivou a realização deste projeto, é o motor de combustão interna (MCI), “uma máquina que transforma energia química, obtido pela queima do combustível, em trabalho mecânico” (BRUNETTI, 2012, p. 27).

As atividades no dia a dia tornavam-se mais versáteis com a utilização de máquinas no qual possuem características predominantes, sendo elas a repetição constante de força que executa trabalhos desde pequenos até grandes dimensões. Vantagens para a evolução da humanidade são diversas, utilizando máquinas onde operam em mineração, transporte, gerador de energia elétrica, construção civil, agricultura etc. como também há desvantagem principalmente para o meio ambiente. Acontece que para o motor funcionar, a queima de combustível se transforma em gases poluentes e não poluentes. Os gases poluentes contribuem para o aquecimento global e por outras questões ambientais que se não for controlado o limite de emissão, as projeções de poluição agravarão e por consequência, ocorrerá a degradação do meio em que vivemos que dificulta a vida no planeta. Com base nos dados do Departamento Nacional de Trânsito (2011), percebemos um aumento significativo de 121% de veículos no período de 2001 até 2011, entretanto, o aumento ainda existe até hoje e por isso é importante controlar emissões resultadas dos MCI, sendo evidenciado que eles contribuem em 22% para o aquecimento global.

Diante do crescimento, a Companhia de Tecnologia Saneamento Ambiental (CETESB) desenvolveu o Programa de Controle da Poluição do Ar por Veículos Automotores (PROCONVE) na década de 80, que padroniza com fases opondo o nível de emissões de gases que os veículos novos precisam emitir (JUNIOR, 2006, p.71). As fábricas então, tiveram que adequar e modificar seus veículos para se

enquadrar com os números de emissões permitidos pelo PROCONVE, que teve uma linha de acontecimentos para chegar nos níveis de hoje. Segundo Altese (2008), os gases admitidos nos MCI até a década de 80, se fazia o uso de carburador, e com o tempo não conseguia acompanhar os níveis de emissões exigidos pelo PROCONVE. A indústria então, teve que modificar o sistema carburado para injeção eletrônica analógica em 1988 que por sua vez passou a ser injeção eletrônica digital em 1991 por não ter componentes importantes para diagnóstico e devolver informações da queima de combustível para a injeção eletrônica.

Um ponto importante no qual é abordado neste projeto, é desenvolver uma injeção eletrônica com recursos atuais de fácil calibração do tempo de injeção e avanço da ignição.

1.1 Motivação

O método que se aplica nas aulas ministradas aos discentes nos cursos de tecnologia, tem grande valor tanto nas aulas teóricas quanto nas aulas práticas, tudo isso acontece pelo nível de capacitação dos docentes no qual envolve alta experiência no mercado a ser passado. Com isso, as aulas práticas devem comprovar as ideias das aulas teóricas fazendo com que o discente acredite que funciona a tecnologia na qual esteja estudando, e ter mais confiança quando estiver trabalhando no mercado de trabalho.

O motivo desse projeto então, é complementar as aulas práticas do curso de Tecnologia em Eletrônica Automotiva, disponibilizando uma Unidade de Controle Eletrônica (ECU – Electronic Control Unit) do motor no qual tenha características específicas ao desenvolvimento do discente. A ECU é encarregada de gerenciar os sensores e atuadores do motor como qualquer outra ECU de fábrica, o que a diferencia das outras é sua interação com o usuário, mostrando na tela em tempo real, os sinais lidos dos sensores e disponibilizando alguns mapas para o usuário editar os tempos de injeção e avanço da ignição.

Outro motivo também, é a facilidade de acesso aos valores de exibição e edição, mostrando que o modo de usabilidade não atrapalhe no desenvolvimento da aula, que isso seja o menor dos problemas quando for acionar o motor. Então, esse projeto servirá de conteúdo didático para as aulas de MCI, um projeto de código aberto onde é disponibilizado todos os *softwares* e circuitos confeccionados na ECU, desenvolvido para complementar os métodos passados no curso de tecnologia.

1.2 Objetivo

Esse trabalho tem como objetivo desenvolver um sistema de injeção eletrônica e fornecer os respectivos *softwares* para o *hardware* da injeção. Em seguida, o sistema desenvolvido irá controlar um sistema de injeção de combustível monoponto no veículo com MCI de quatro cilindros. O veículo é o GOL 1.0i 1995 da *Volkswagen*, e será controlado por um Raspberry Pi 3 B como uma ECU de injeção de combustível eletrônica (EFI - *Electronic Fuel Injection*). Esse projeto é de uso didático para as

universidades que estudam o conceito de injeção eletrônica, colocando em prática o aprendizado teórico.

1.3 Resultados Esperados

Espera-se desenvolver o *hardware* e *software* de uma nova ECU de EFI que substitua a original do veículo, mantendo o motor funcionando e fazendo testes no veículo com carga de torque e potência no dinamômetro da Fatec Santo André. Com o resultado dos testes será possível uma comparação com os mesmos testes usando a ECU de EFI original.

2 REVISÃO BIBLIOGRÁFICA

O desenvolvimento do projeto se baseia em projetos *open source*, “quando as pessoas dizem que o *software* é “*free*” ou “*open source*”, elas querem dizer que um usuário pode obter uma cópia sem nenhum custo e depois estudar legalmente seu código-fonte, modificá-lo e distribuí-lo a outros - também gratuitamente” (VON HIPPEL, 2001, p.84). Contudo, o trabalho sendo *open source* garante disponibilizar os códigos para serem usados, modificados e compartilhado sem quaisquer meios lucrativos.

Existem trabalhos que foram feitos para substituir um sistema de injeção eletrônica de fábrica dos veículos e irão servir como base para este trabalho.

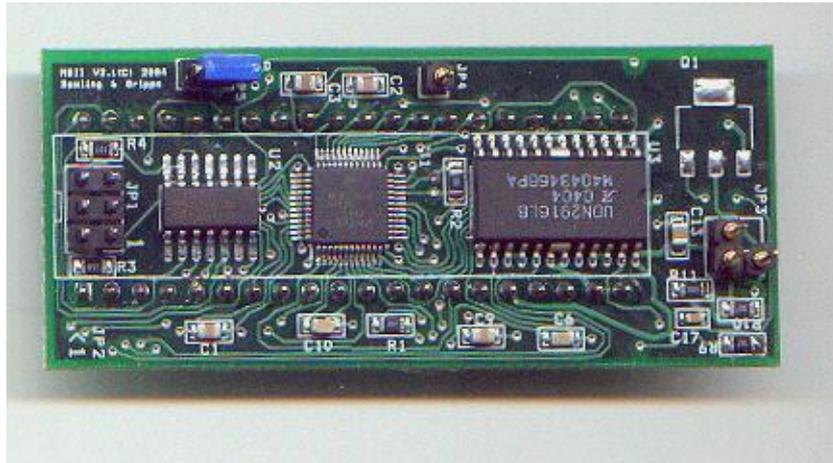
2.1 MegaSquirt

O projeto *MegaSquirt* é um controlador de injeção eletrônica criado em 1996, onde seus criadores, Bruce Bowling e Al Grippo, dois entusiastas sobre o assunto que foram integrantes do fórum criado por Jhon Gwynne, universitário no estado de *Ohio*, decidiram se juntar para criar um *hardware* para melhorar o desempenho do Camaro 1973, cujo dono do veículo era o Al Grippo, que disponibilizou para ser o carro teste do projeto. Com tudo, por trás do objetivo que o projeto entrega, o motivo se tornou o principal interesse aos usuários. Não havia documentação específica de como funcionava uma injeção eletrônica naquele tempo, então, o projeto se baseia no conceito *open source* com o método faça você mesmo (*do it yourself* - DIY). O intuito de fornecer os códigos e circuitos aos usuários, possibilita que o usuário, proprietário do veículo, faça e desenvolva sua injeção eletrônica através do material disponibilizado pelos criadores e está disponível o fórum na internet para auxiliar e não deixar os usuários com dúvidas, assim então, fazendo com que o projeto funcione como o esperado. (MEGASQUIRT, 2022)

2.1.1 Hardware

O *MegaSquirt* trabalha com duas versões de processadores, o *MegaSquirt-I* e o *MegaSquirt-II*, ilustrado na figura 1.

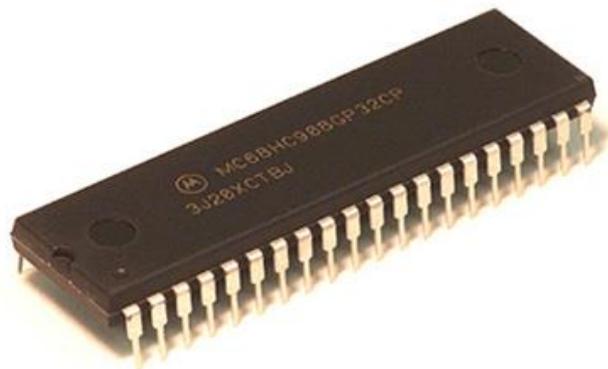
Figura 1 - MegaSquirt-II (MC9S12C64)



Fonte: Extraído de (MEGAMANUAL, 2022)
<http://www.megamanual.com/ms2/MS-11dcard.jpg>

O *MegaSquirt-II* é um processador atualizado e mais poderoso que o *MegaSquirt-I*, ilustrado na figura 2.

Figura 2 - MegaSquirt-I: (68HC908)

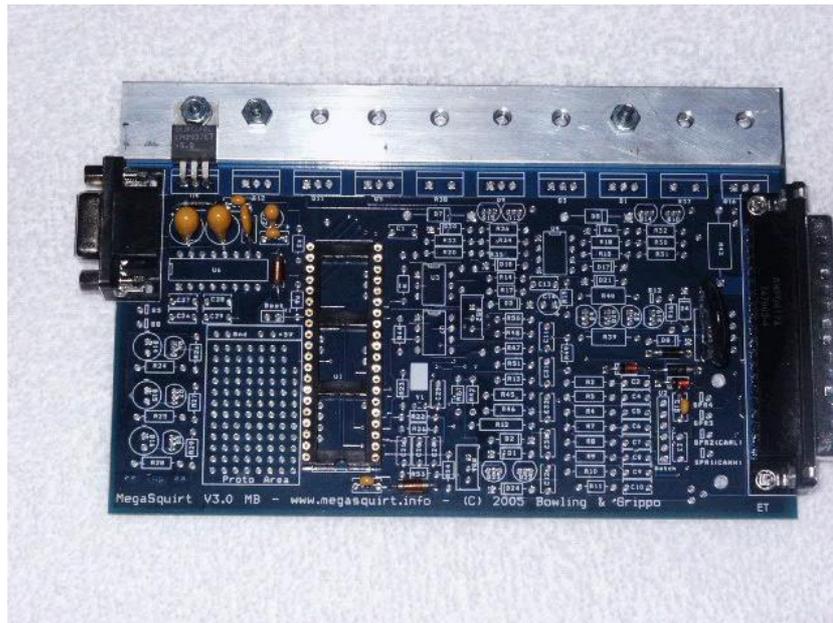


Fonte: Extraído de (MEGAMANUAL, 2022)
<http://www.megamanual.com/v22manual/CPU.jpg>

Esses processadores são fixados na placa de circuito impresso (PCI) principal da MegaSquirt, ilustrado na figura 3, que está pronta para ser soldada com os componentes disponíveis com o conjunto. O *hardware* é apenas o controlador de EFI,

significa que o usuário deve ter todos os sensores e atuadores do veículo já disponíveis para conectar ao *MegaSquirt*.

Figura 3 - Placa principal MegaSquirt DIY

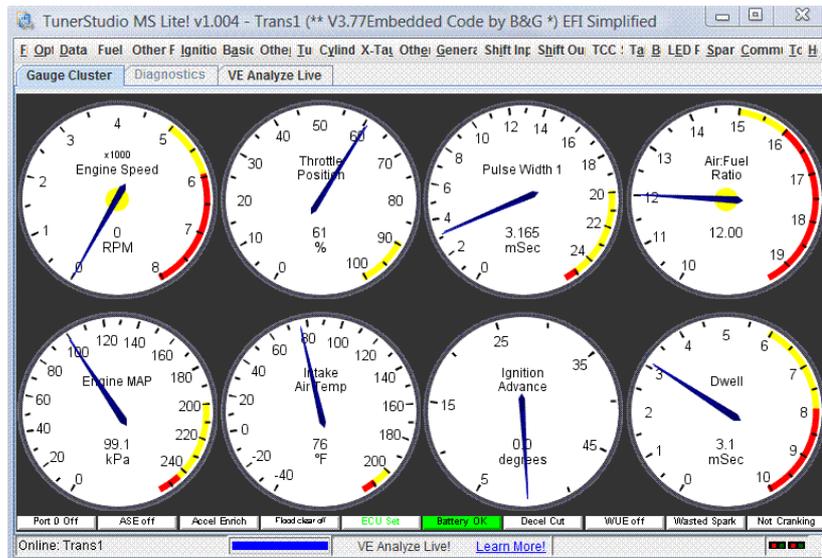


Fonte: Extraído de (MEGAMANUAL, 2022)
http://www.megamanual.com/ms2/heat_sink2.JPG

2.1.2 Software

O *software* é carregado no processador compatível pela versão e está disponível gratuitamente com atualizações. Além do *software* para gerenciar os sensores e atuadores do veículo, há também os *softwares* de interação ao usuário como o TunerStudioMS, que será necessário um computador com sistema operacional (OS) para visualizar o comportamento dos parâmetros do motor como rotação, velocidade, temperatura, entre outros. O *software* também possibilita ao usuário fazer o ajuste do tempo de injeção em cada regime de trabalho do motor, tudo isso através de uma comunicação serial à cabo conectado ao veículo.

Figura 4 - Software TunerStudioMS



Fonte: Extraído de (MEGAMANUAL, 2022)
http://www.megamanual.com/TS_front.GIF

2.2 Fuel Tech

Anderson Dick, criou o projeto de injeção eletrônica como trabalho de conclusão de curso (TCC) que fundou sua empresa registrada em 2003, nomeando-a de Fuel Tech, uma empresa brasileira que virou referência mundial. O dispositivo é compacto e busca por inovação incessante, o *drive* conta com uma tela interativa e sensível ao toque que possibilita a visualização de valores lidos dos sensores do veículo como mostra a figura 5, também disponibiliza o acesso para edição dos parâmetros como por exemplo, os tempos de injeção e avanço da ignição.

Figura 5 - FT600 SFI



Fonte: Extraído de (FUELTECH, 2022)
<https://fueltech.com.br/products/ft600-sfi-injecao-eletronica>

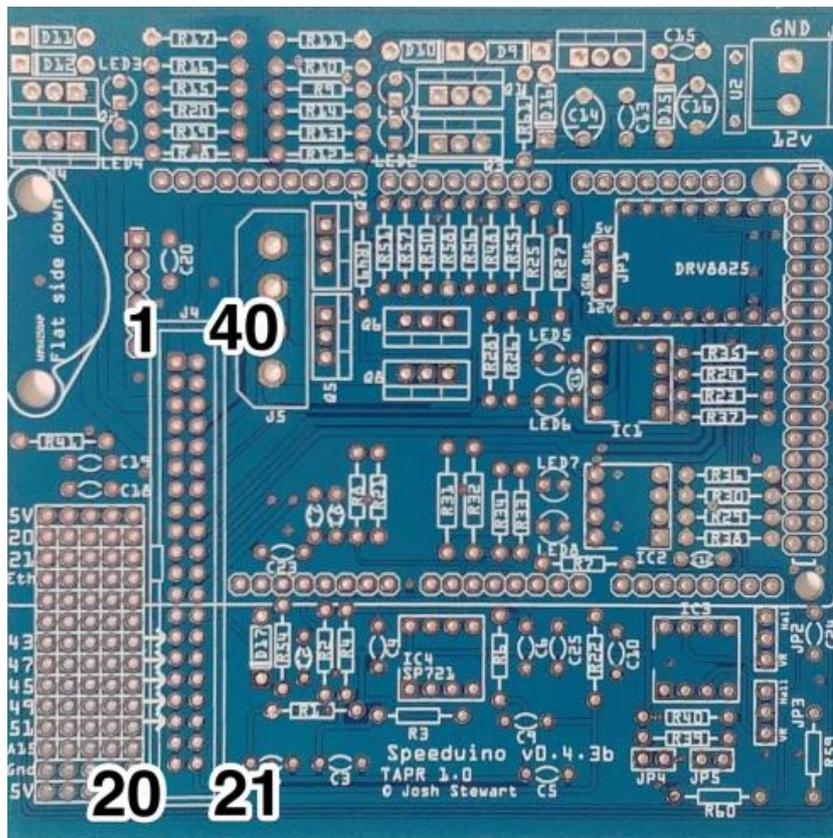
A utilização da Fuel Tech é principalmente voltada para carros de competição, tal categoria incluso no mercado de alta performance automotiva. O objetivo do projeto é obter mais potência no veículo, modificando poucas peças no motor, como injetores, embreagens, fluido de freios por exemplo, para que possa trabalhar com a injeção eletrônica com mais precisão. Através das modificações feitas por Anderson Dick, em um Corvette 6.2 cm³ V8 modelo 2020, o projeto está sendo referência no mundo por conta dos 1350 cv de torque alcançados nas rodas do veículo sendo conhecido como o Corvette mais rápido e potente do mundo.

A Fuel Tech visa crescer junto com o avanço e pré-requisitos estabelecidos pela tecnologia. Contudo, um dos principais avanços que a Fuel Tech quer alcançar, é desenvolver o gerenciamento eletrônico para veículos com motores elétricos até 2024, cujo veículo é tendência de crescimento, está sendo desenvolvido pelas montadoras para diminuir os poluentes emitidos por motores a combustão. O projeto tem como resultado converter motor a combustão em elétrico, modificando peças do veículo e substituindo equipamentos. (GAZ, 2022)

2.3 Speeduino

Baseado em um projeto *Open Source*, *Speeduino* também foi desenvolvido para melhorar o desempenho do motor de combustão interna tendo seu funcionamento similar ao *MegaSquirt*, onde possui uma placa de circuito impresso como mostra na figura 6, o processador e a programação disponível na Internet junto com os programas executáveis para computador que tem toda a parte gráfica e interativa.

Figura 6 - Placa Speeduino v0.4

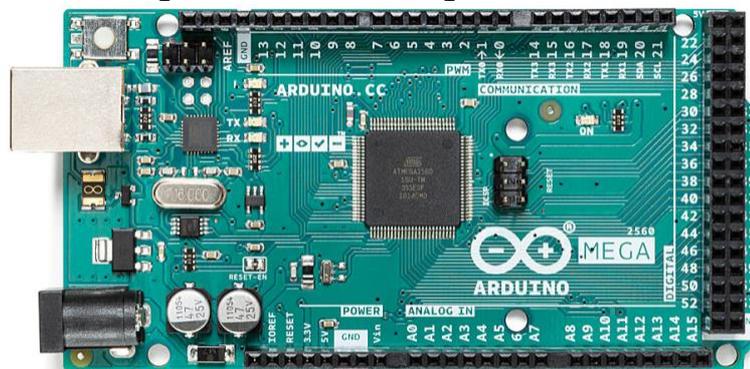


Fonte: Extraído de (SPEEDUINO MANUAL, 2022)

https://speeduino.com/Speeduino_manual.pdf

O *Arduino Mega 2560*, como ilustra a figura 7, é encaixado sobre a placa *Speeduino*, onde a placa é responsável por tratar os sinais lidos pelos sensores, deixando em condições para o *Arduino Mega* entender, processar os sinais, calcular o tempo de injeção e devolver esse sinal para a placa novamente, porém, agora usa a parte dos *drives* para atuar os injetores de combustível e as bobinas de ignição.

Figura 7 - Arduino Mega 2560 Rev3



Fonte: Extraído de (ARDUINO, 2022)

<https://store.arduino.cc/products/arduino-mega-2560-rev3>

O projeto conta com o método DIY, isso faz com que o usuário aprimore e se aprofunde mais nos conceitos de funcionamento do motor, no qual o usuário tenha conhecimento básico em elétrica para fazer as devidas conexões de sensores e atuadores, como também tenha conhecimento básico em eletrônica para soldar os componentes na PCI. Outro fator que desafia o usuário é ter um conhecimento intermediário em programação, já que a programação é total editável para cada finalidade.

O *Speeduino* foi desenvolvido pela empresa *Arduino* no qual foi fundada pelo Massimo Banzi em 2005. Um projeto no qual complementa como material didático e de apoio em cursos que envolvam motores de combustão interna.

3 METODOLOGIA DO PROJETO

O funcionamento da injeção eletrônica programável desenvolvida neste trabalho, é dividido em processos no qual são apresentados desde *softwares*, programações feitas para o microcontrolador e para o Raspberry, até o esquema de montagem da ECU, ilustrado através de diagramas em blocos. O projeto utiliza o microcontrolador PIC18F4550 para ler os sinais analógicos no qual o Raspberry Pi não possui o conversor analógico para digital. O PIC18F4550 fica responsável pela leitura dos sensores e controle dos atuadores enquanto o Raspberry Pi 3 B faz toda a parte gráfica da *interface* de interação com o usuário, calculando e retornando os valores de controle para microcontrolador.

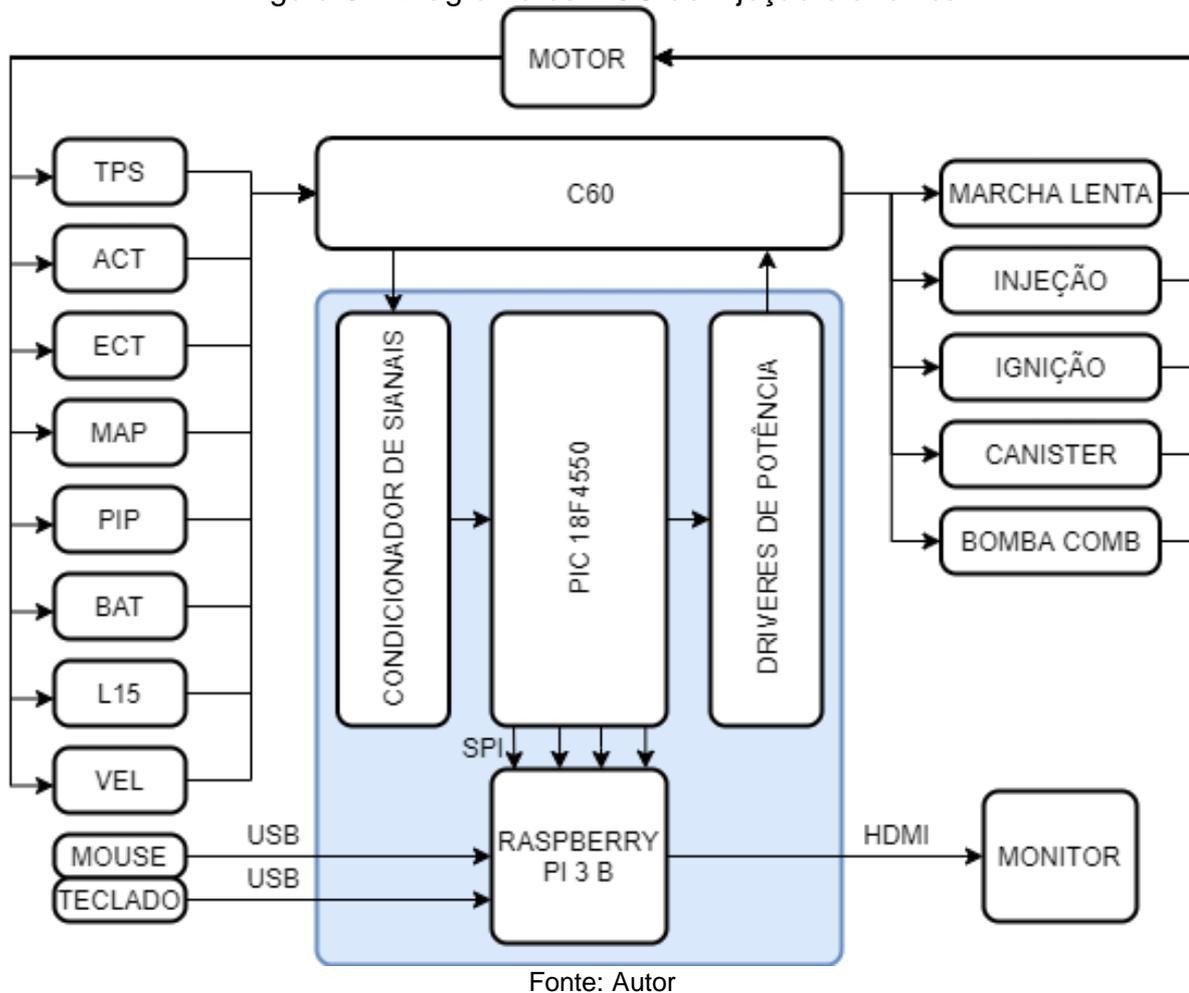
3.1 Teoria Sobre o Projeto

De acordo com Alves e Santos (2019) o conceito do funcionamento do MCI, possibilita desenvolver o gerenciamento do motor. O uso da injeção eletrônica é exigido para os veículos de fabricação atuais, estando dentro das normas do PROCONVE, que limita as emissões de gases do veículo. Desta forma, surgiu o sistema eletrônico, o qual atendeu as exigências, garantindo precisão e eficácia ao sistema.

Para o funcionamento do motor é necessário o gerenciamento entre sensores e atuadores junto com seus processadores, ilustrado na figura 8. O motor de combustão interna que possui fenômenos físicos como: temperatura do motor (ECT – *Engine Coolant Temperature*), temperatura do ar de admissão (ACT – *Air Charge Temperature*), posição da válvula borboleta (TPS – *Throttle Position Sensor*), pressão de admissão (MAP – *Manifold Absolute Pressure*), rotação do motor (PIP – *Pick up Ignition Profile*), nível de bateria e a velocidade do veículo. Todos esses fenômenos físicos são convertidos em sinais elétricos e tratados pelos circuitos condicionadores para o sistema eletrônico, podendo ser sinais digitais ou analógicos. Esses sinais que são medidos no motor a todo instante é a base para informar o conjunto de processadores sendo: a) o microcontrolador da família 18F modelo 4550 da *Microchip* e b) microcomputador da Raspberry Pi 3 B, juntos, a função é receber os sinais medidos no motor e emitir os pulsos do tempo de injeção, avanço de ignição, canister, bomba de combustível e a marcha lenta, conforme as condições de funcionamento do

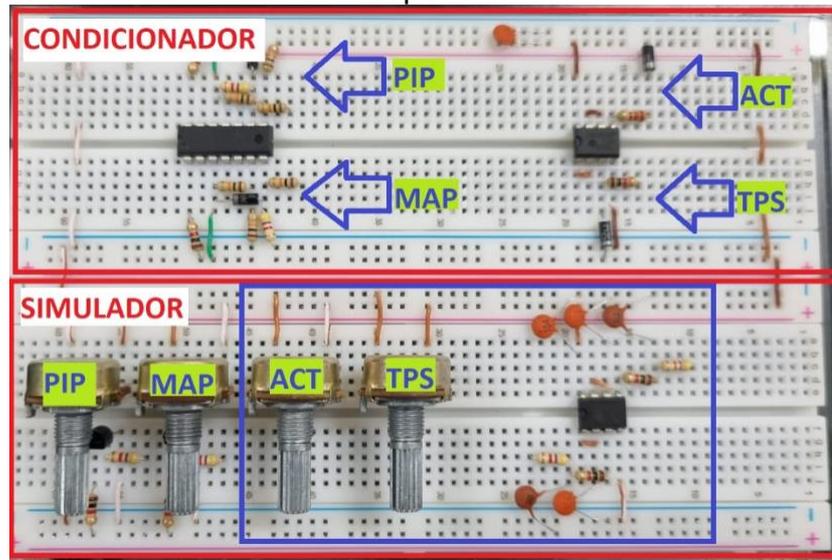
motor, onde os sinais antes de chegarem nos atuadores são passados no sistema de *drive* para amplificar os sinais e serem compatíveis para cada atuador.

Figura 8 – Diagrama da ECU de injeção eletrônica



O desenvolvimento do projeto utiliza ferramentas computacionais para fazer o *layout* da ECU, programação do microcontrolador e programação do processador. A figura 9, ilustra a simulação dos sinais dos sensores e os circuitos condicionadores tratando os sinais que foram feitos fisicamente na placa de ensaio, testando todos os componentes e comportamento dos sinais. Na parte superior da figura, estão presentes os circuitos condicionadores e na parte inferior os circuitos simuladores dos sinais dos sensores.

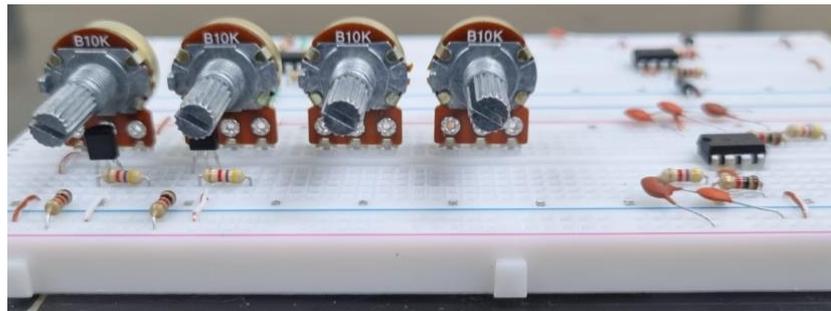
Figura 9 - Placa de ensaio de circuitos simuladores e condicionadores com a vista superior



Fonte: Autor

Os mesmos circuitos também estão ilustrados na figura 10 de modo que agora com a vista da parte frontal e a posição dos potenciômetros simulando os sinais de PIP, MAP, ACT e TPS.

Figura 10 - Placa de ensaio de circuitos simuladores e condicionadores com vista frontal



Fonte: Autor

O programa KiCad tem a função de fazer o *layout* da placa de circuito impresso com os circuitos condicionadores, circuitos de *drives* e processadores. O esquema elétrico de cada circuito que engloba a ECU está ilustrado em figuras, iniciando com o esquema elétrico do sensor de PIP.

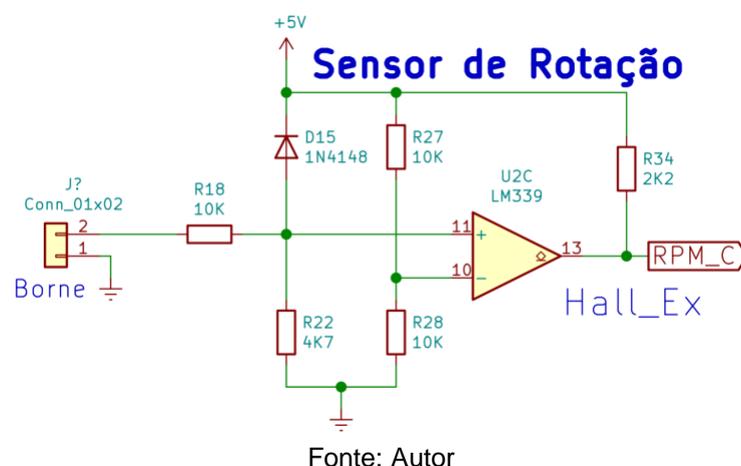
O sensor de rotação passa pelo condicionador de sinal exclusivo para sensores digitais, aqueles sinais que desligam e ligam no tempo. O sinal sempre vai ser 0, a tensão de referência 0 Volts (V) da fonte, ou 1, a tensão máxima da fonte. O sinal

digital não pode estar variando a tensão entre 0 e 1, no entanto, o sensor pode estar apresentando defeito porque estas são características de sensores analógicos.

O circuito condicionador do sinal de PIP, um sinal que é resultado da movimentação do campo magnético, precisa ser convertido na faixa de tensão operacional do microcontrolador. O circuito condicionador que ilustra a figura 11, há o conjunto de resistor e diodo grampeando o sinal para operar na faixa de 0 e 5 V com um resistor de *pull down* e assim o sinal é ligado em uma das entradas do CI LM339. Na outra entrada do CI tem o conjunto de resistores no modo divisor de tensão, sendo assim, o valor na entrada do CI vai ser a metade do sinal da fonte tendo um valor fixo de 2,5 V, já que a fonte gera 5 V.

Em seguida, os sinais são comparados pelo CI LM339 testando se o sinal vindo do sensor é maior que 2,5 V, caso a resposta seja verdadeira a saída se torna um circuito aberto que não gera tensão e não possui referência. Logo, caso a resposta seja falsa, a saída chaveia para o negativo da fonte, jogando 0 V na saída. Entretanto, quando a resposta for verdadeira, existe o resistor de *pull up* para jogar a tensão da fonte na saída fazendo assim, o circuito gerar um sinal de 0 volts e 5 volts.

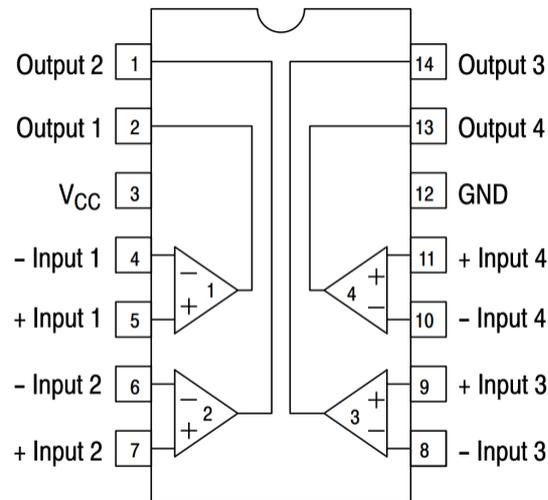
Figura 11 - Circuito condicionador PIP



O Circuito Integrado (CI) LM339 é construído com uma linha dupla de pinos (DIP – Dual In-line Package) de 14 pinos composto por quatro comparadores de baixo consumo como ilustra a figura 12 com a estrutura elétrica interna do CI. O CI permite que as entradas não consumam corrente do sensor por ter entradas de alta

impedância, e como a saída é sempre aberta quando a comparação entre a entrada positiva for maior que a entrada negativa, esse sinal pode controlar os circuitos com qualquer tensão através do resistor de *pull up*.

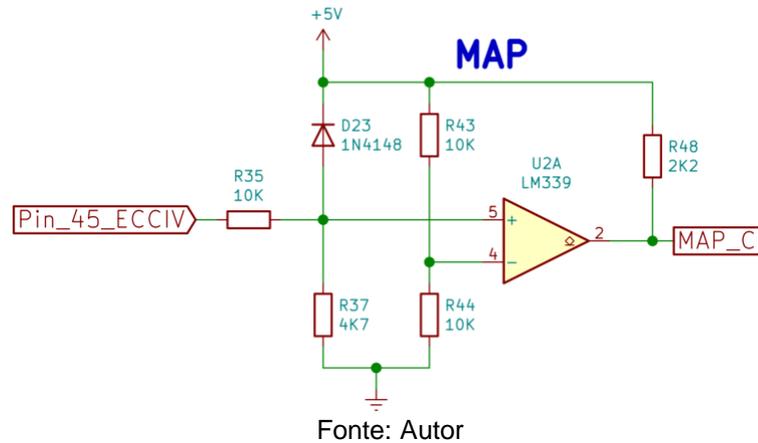
Figura 12 - Estrutura elétrica interna do CI LM339



Fonte: Extraído de (LM339 ONSEMI DATASHEET, 2022)
<https://www.onsemi.com/pdf/datasheet/lm339-d.pdf>

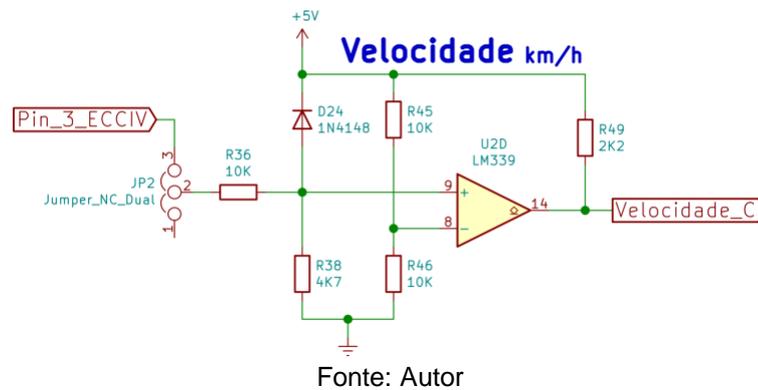
Este CI, além de condicionar o sinal vindo do sensor de PIP, compacta mais três circuitos, sendo o sensor MAP, sensor de velocidade e sensor da linha 15. Estes circuitos têm a mesma base de funcionamento que o de PIP, ou seja, grampeamento evitando tensão maior que 5 V, um divisor de tensão de 2,5 V para ser a base da comparação e um resistor *pull up* na saída. O circuito condicionador do sensor MAP está ilustrado na figura 13.

Figura 13 - Circuito condicionador do sensor MAP



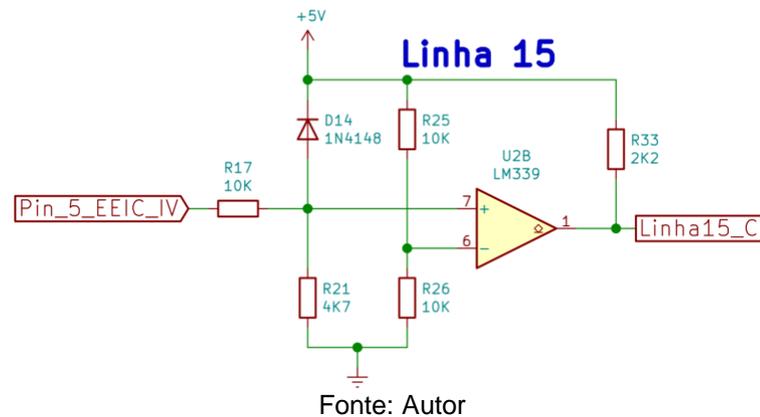
O circuito condicionador do sensor de velocidade está ilustrado na figura 14. Este sensor se localiza o mais próximo às rodas para captar o movimento do veículo. É construído através de um conjunto de indutor e ferro que ao ser induzido pela roda dentada do eixo rotativo, irá transformar em tensão.

Figura 14 - Circuito condicionador do sensor de velocidade



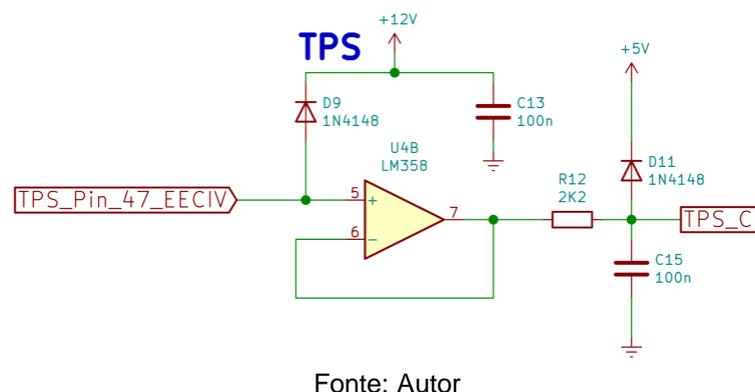
O circuito condicionador do sensor da linha 15, está ilustrado na figura 15. A linha 15 é responsável por informar se a ignição está acionada e alimentando o circuito em todo o veículo.

Figura 15 - Circuito condicionador linha 15



Para os sensores analógicos, o tratamento dos sinais utiliza outros circuitos condicionadores, o circuito condicionador do sensor de TPS ilustrado na figura 16 há um diodo no modo grampeador que evita surtos de tensão na entrada do circuito, impedindo a queima do CI LM358. Em frente há um conjunto de resistor e diodo no modo grampeador, evitando que o sinal passe de 5 V no pino de saída para o próximo circuito. Os capacitores, estão montados para filtrar tanto aos ruídos da fonte como também estabilizar o sinal de saída. É importante ressaltar que a tensão vinda do sensor é de sinal analógico que varia no tempo com amplitude de até 12 V.

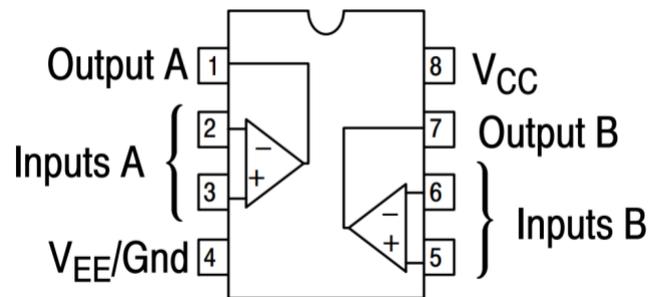
Figura 16 - Circuito condicionador TPS



O CI tipo DIP de 8 pinos que tem um duplo Amplificador Operacional (Amp-Ops) de baixo consumo, é a parte principal do circuito que está configurado com seguidor de tensão, ou seja, a tensão que entra é a mesma que sai do CI, porém, a impedância do pino de saída é muito baixa, enquanto o pino de entrada é extremamente elevado, evitando o consumo da corrente vindo do sensor. A vantagem de usar o LM358 é que

ele possui dois Amp-Ops fazendo com que ajude a compactar dois condicionadores em apenas um CI, a qual possui a estrutura de seu circuito interno ilustrada na figura 17.

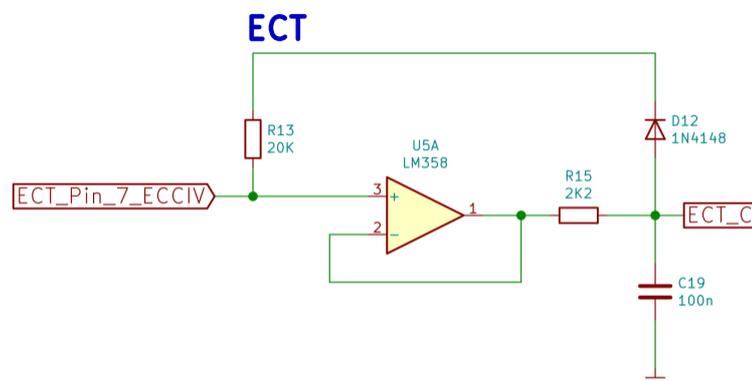
Figura 17 - Estrutura elétrica interna do CI LM358



Fonte: Extraído de (LM358 ONSEMI DATASHEET, 2022)
<https://www.onsemi.com/pdf/datasheet/lm358-d.pdf>

O circuito condicionador do sensor ECT está ilustrado na figura 18, o funcionamento é o mesmo que o circuito que trata o sinal vindo do sensor TPS. O sensor ECT é um sistema resistivo de coeficiente de temperatura negativo (NTC – *Negative Temperature Coeficient*) que diminui a resistência conforme eleva a temperatura. Neste caso, o sensor precisa se referenciar com um resistor *pull up* situado na entrada do CI.

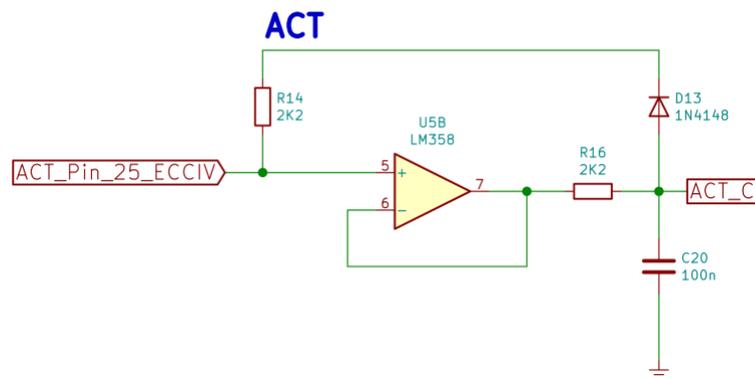
Figura 18 - Circuito condicionador do sensor ECT



Fonte: Autor

O circuito condicionador do sensor de ACT que está ilustrado na figura 19, tem o mesmo funcionamento do circuito condicionador do sensor ECT.

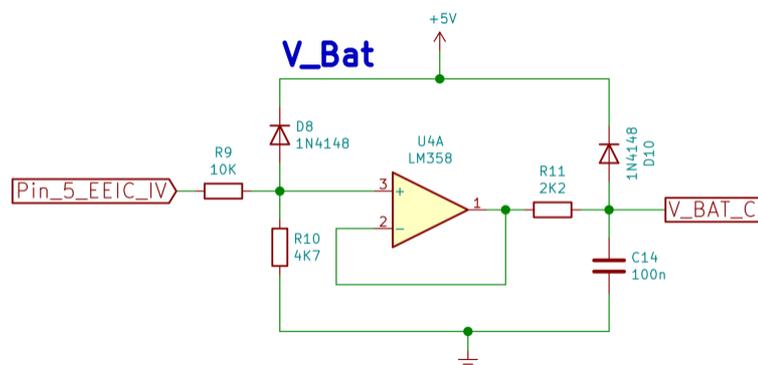
Figura 19 - Circuito condicionador do sensor ACT



Fonte: Autor

Os circuitos que condicionam os sensores analógicos são tratados com o mesmo CI, isso compacta e melhora o posicionamento dos circuitos na ECU. O circuito condicionador do sinal vindo da bateria, ilustrado na figura 20, também utiliza o mesmo CI, totalizando dois CI's para quatro sensores diferentes.

Figura 20 - Circuito condicionador tensão da bateria



Fonte: Autor

3.2 Descrição do Projeto

O GOL 1.0i 1995 Volkswagen, tem um motor aspirado com 8 válvulas e uma única válvula injetora de combustível. Contudo, a parte elétrica do motor se unifica em um conector que chamamos de C60, demonstrado na figura 21. Nele, estão todos os sensores e atuadores que a ECU precisa para ler e controlar respectivamente.

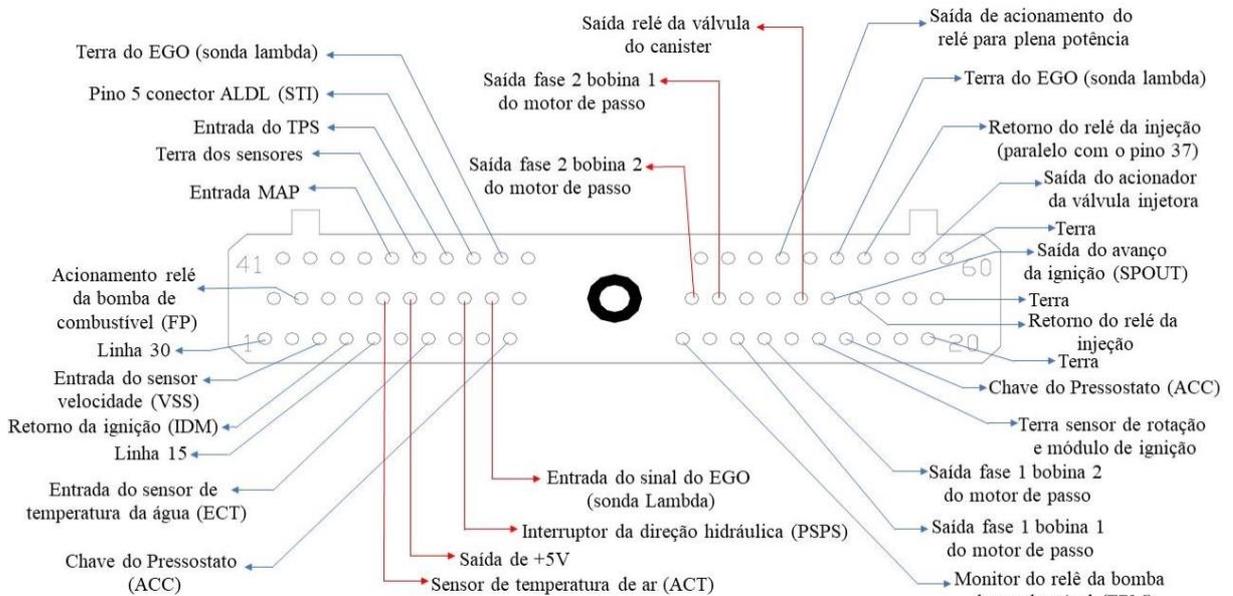
Figura 21 - Conector C60



Fonte: Extraído de (PAHLS PERFORMANCE, 2022)
<https://pidashs.com/products/ford-60-pin-pcm-connector>

A identificação dos pinos do conector C60 está ilustrado na figura 22. Os pinos estão conectados para cada sensor e atuador no chicote de fios do GOL e os pinos de alimentação para a ECU de EFI.

Figura 22 - Pinagem C60



Fonte: Estudo de ECUs de EFI "open source" do mercado brasileiro em substituição a EEC-IV da Ford

O tratamento dos sinais vindos do conector C60, passam pelos circuitos condicionadores que transformam e filtram os sinais dentro da faixa de trabalho do microcontrolador que, para sinais digitais e analógicos precisam estar dentro de 0 a 5 V de tensão, caso exceda o valor, tanto o pino pode vir a queimar quanto o microcontrolador poderá apresentar curto-circuito e parar de funcionar completamente. Deste modo, é importante um tratamento dos sinais antes de chegarem nos pinos do microcontrolador.

Quando o microcontrolador fizer uma varredura dos valores lidos de todos os sensores em um tempo de amostragem constante, os valores são armazenados em sua memória de acesso aleatório (RAM – Random Access Memory) e transmitidos via comunicação de *interface* Periférica Serial (SPI – *Serial Peripheral Interface*) para o Raspberry Pi. Logo, o Raspberry Pi recebe os valores e calcula através de fórmulas ou mapas, o tempo de injeção e o ângulo do avanço de ignição que o motor precisa controlar a partir das características de funcionamento naquele instante (PIP, TPS, ACT, ECT, MAP). Em seguida, o Raspberry Pi retorna com os valores para o microcontrolador que aciona a válvula injetora e a bobina de ignição brevemente. Os sinais de que o microcontrolador tem a função de controlar, também precisam ser tratados e amplificados para cada atuador em específico, neste caso, o tratamento se chama *drive*.

Diante disso, há dois microprocessadores trabalhando ao mesmo tempo sendo: 1) microprocessador embutido no microcontrolador PIC18F4550 e 2) microprocessador embutido no microcomputador Raspberry Pi. Juntos, trabalham em harmonia e solucionam problemas. O microcontrolador PIC18F4550 tem periféricos que o Raspberry Pi não possui como o conversor analógico para digital, o qual lê sensores de sinais que variam no tempo. O Raspberry Pi também possui vantagens sobre o microcontrolador, como desenvolver uma *interface* de interação com o usuário. Com isso, o microcontrolador tem a função de obter os valores de todos os sensores, enquanto o Raspberry Pi processa os dados e retorna o tempo de injeção e ângulo do avanço de ignição.

3.2.1 Hardware do Projeto

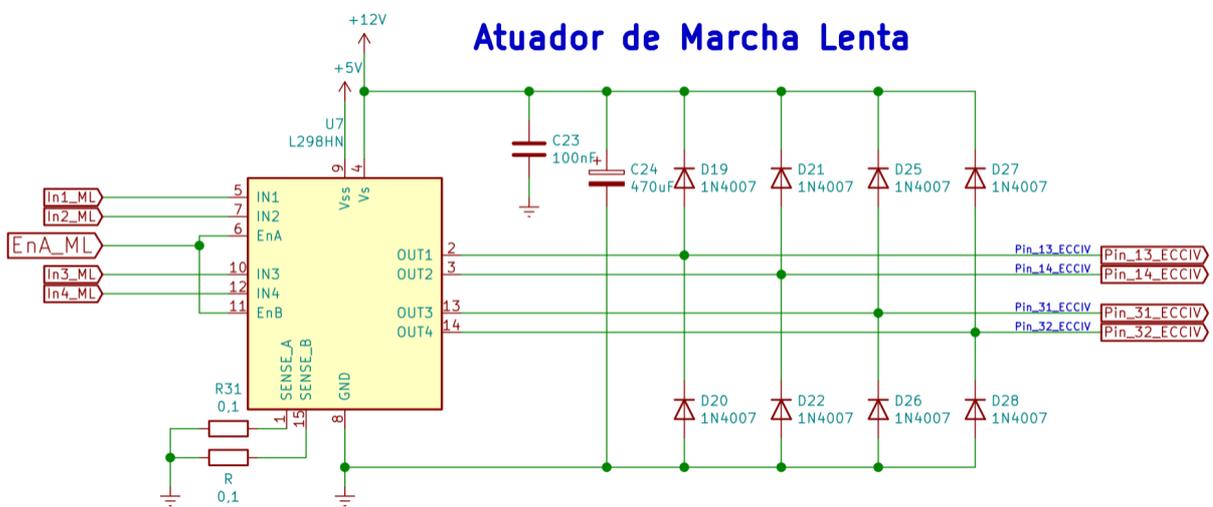
O *Hardware* do projeto, também conhecido como ECU, possui condicionadores, *drive* e processadores, detalhados abaixo.

3.2.1.1 Drive

Os circuitos condicionadores recebem os sinais direto dos sensores que por sua vez são sinais de tensão superior ao que o microcontrolador suporta, ou ainda, sinais de que contém ruídos que prejudicam os pinos de entradas do microcontrolador. Por diversos fatores é importante o sinal ser tratado e preparado em condições no qual o microcontrolador possa interpretar sem ocasionar nenhum dano a ele. Enquanto, os *drives* são circuitos que amplificam os sinais vindos do microcontrolador quando na intenção de controlar um *hardware* de potência, sua função é garantir que a tensão e a corrente que sai do pino do microcontrolador de baixa potência controle uma bobina de 12 V, por exemplo, que exige uma potência maior. Portanto, se faz o uso de *drive* para amplificar os sinais de acordo com cada atuador existente no motor. Além disso, os *drives* também tem como função desacoplar o circuito de comando com o circuito de potência. Neste caso, os circuitos de potência vão controlar os atuadores de marcha lenta, válvula injetora, canister, bomba de combustível e as bobinas de ignição.

O circuito de marcha lenta, ilustrado na figura 23, controla um motor de passo, o qual possui duas bobinas que precisam desligar ou inverter a polaridade a cada ângulo de rotação. O sinal vindo do microcontrolador está conectado ao CI L298 que vai amplificar e inverter a polaridade do sinal com potência suficiente para o motor de passo. Na saída do CI estão conectados os pinos do motor de passo e um conjunto de diodos, os quais são colocados de forma paralela em cada pino no modo de proteção contra eletromotriz, uma característica das bobinas que dissipam a energia do campo magnético em forma de alta corrente quando são desligadas.

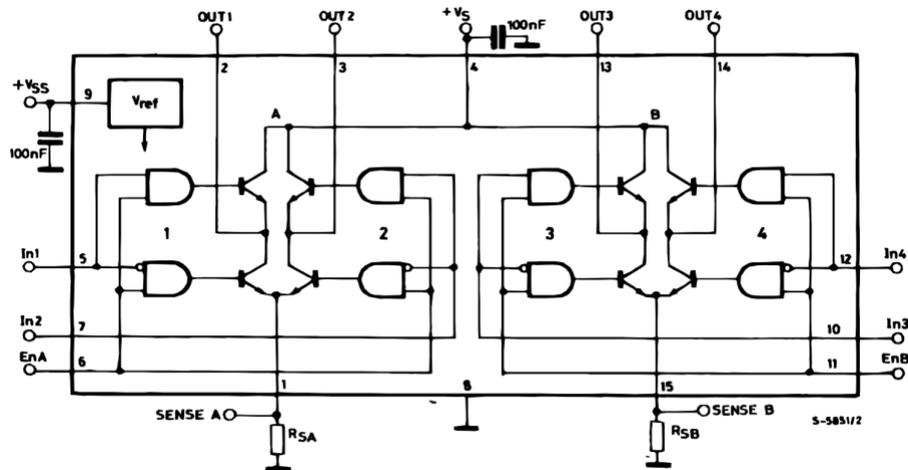
Figura 23 - Circuito *drive* de marcha lenta



Fonte: Autor

O CI é construído a partir de duas pontes que é conhecido como ponte H pelo formato de construção dos transistores como ilustra o diagrama em blocos na figura 24.

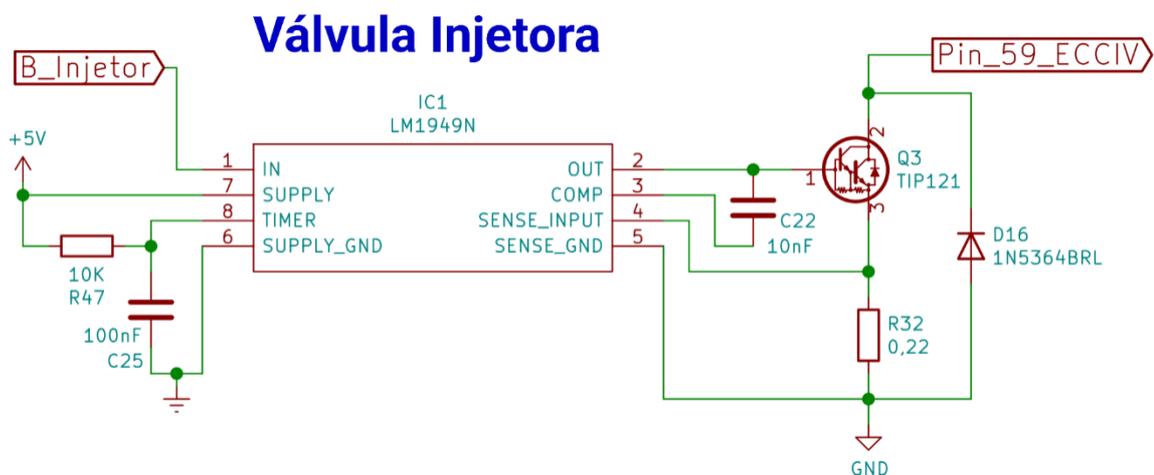
Figura 24 - Diagrama em blocos CI L298



Fonte: Extraído de (L298 ST, 2022)

<https://www.st.com/en/motor-drivers/l298.html>

O circuito que controla a corrente da válvula injetora de combustível, ilustrado na figura 25, precisa do sinal ativo do microcontrolador com o tempo de abertura da válvula somado com o tempo que a válvula precisa estar aberta e considerar o tempo em que a válvula está fechando, tudo em um único sinal.

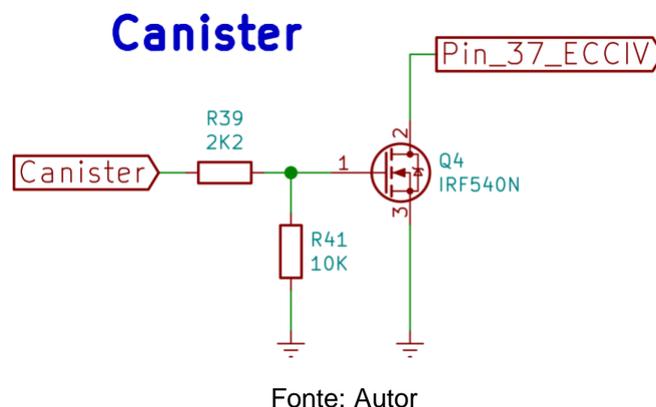
Figura 25 - Circuito *drive* da válvula injetora

Fonte: Autor

Em seguida, esse pulso é conectado na entrada do CI LM1949, específico para acionar válvulas injetoras no qual fornece corrente de pico de até 4 vezes a corrente normal da válvula, isso faz com que vença as forças de atrito e mola interna da válvula possibilitando a abertura com eficiência.

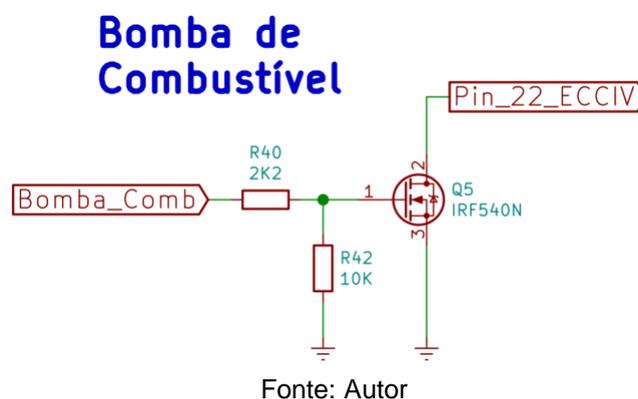
O circuito para controlar o canister, ilustrado na figura 26, possui um Transistor de Efeito de Campo Metal Oxido Semicondutor (MOSFET – *Metal Oxide Semiconductor Field Effect Transistor*) três pinos do tipo N IRF540. O funcionamento e a função do transistor é chavear o circuito para o polo negativo, pois, o circuito já está sendo alimentado pela fonte e precisa suportar a potência calculada que passa pelo transistor. No sinal de comando no pino 1, o sinal que entra passa por um grupo de resistor montados como divisores de tensão, sendo assim, essa tensão será o parâmetro básico que indica o ganho no qual deve controlar a carga.

Figura 26 - Circuito *drive* do Canister



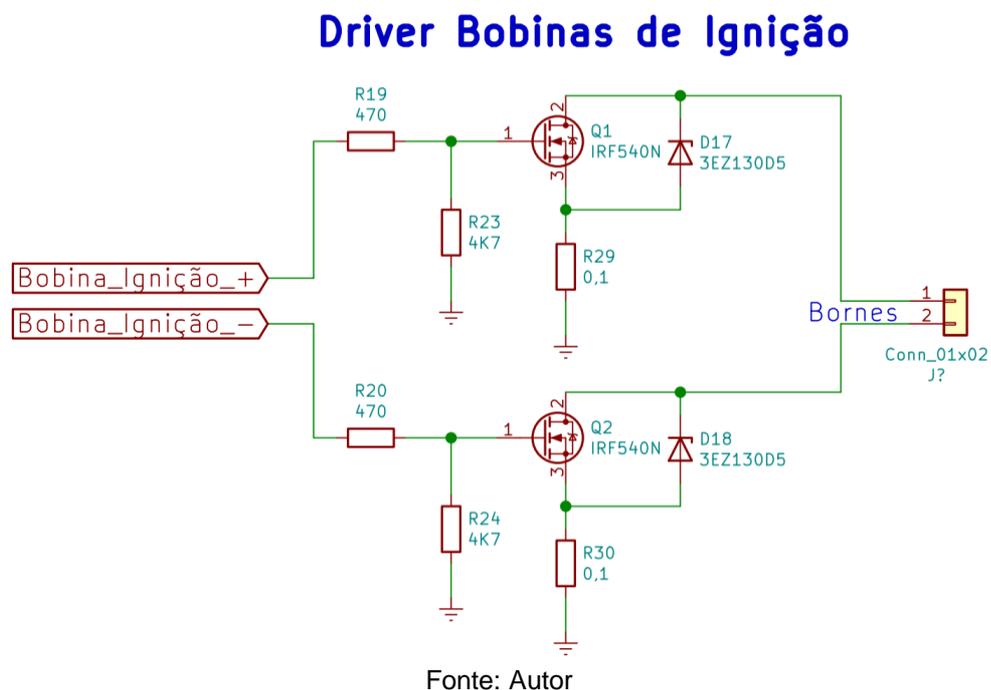
O circuito que controla a bomba de combustível, ilustrado na figura 27, tem o mesmo funcionamento do circuito que controla o canister. Possui um MOSFET IRF540 do tipo N que dá ganho ao sinal de comando para controlar o circuito de potência da bomba.

Figura 27 - Circuito *drive* da bomba de combustível



O circuito da bobina de ignição, ilustrado na figura 28, não se diferencia dos circuitos de canister e bomba de combustível, pois, apresenta os mesmos componentes com as mesmas características. A única diferença é que possui um diodo em paralelo com o transistor para limitar a corrente contra a eletromotriz das bobinas ao serem desligadas.

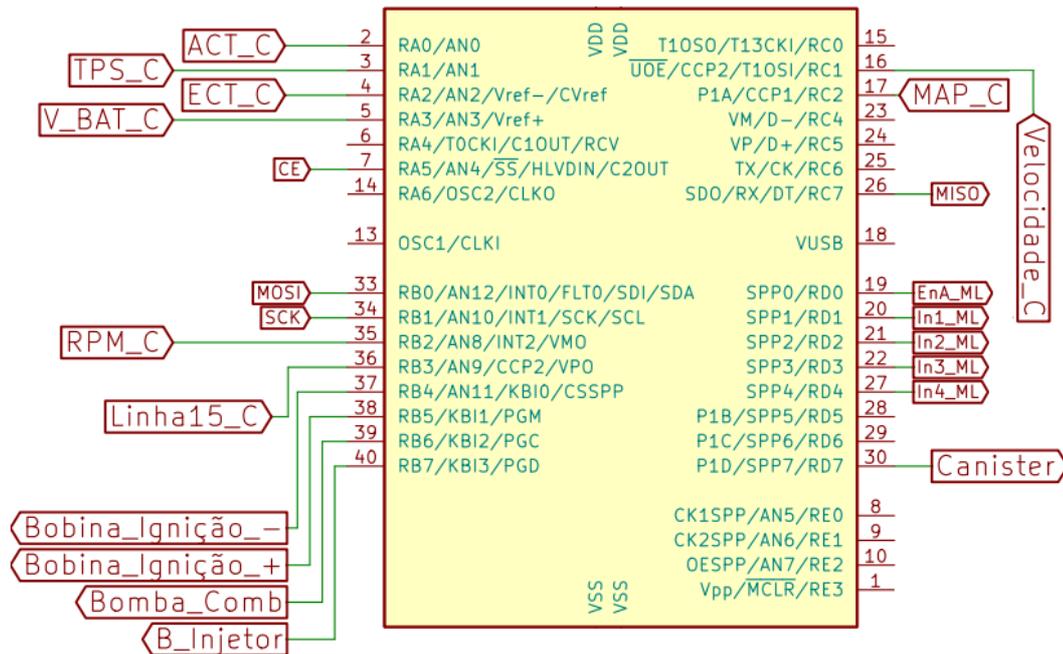
Figura 28 - Circuito *drive* bobina de ignição



3.2.1.2 Microcontrolador PIC18F4550

Os sinais que saem e retornam ao motor, tais como os sensores e atuadores, se juntam ao conector C60. Em seguida, passam pelos circuitos condicionadores e *drive*, que se conectam diretamente ao PIC18F4550, componente que faz parte da central de controle. Os locais em que serão conectados cada sinal que o microcontrolador precisa gerenciar, e os pinos de comunicação de SPI entre o PIC18F4550 e o microcomputador Raspberry Pi 3 B, estão ilustrados na figura 29.

Figura 29 - Pinagem do PIC18F4550



Fonte: Autor

O microcontrolador PIC18F4550 é um DIP de 40 pinos, desenvolvido pela empresa Microchip com o objetivo de ser um CI programável com a vantagem de fazer o controle de diversos projetos, sendo assim, um único CI que pode ser substituído por diversos circuitos ao mesmo tempo em um projeto. Isso compacta e traz confiabilidade na questão de montagem, solucionando problemas como o mau contato. Este microcontrolador faz a leitura dos sinais analógicos e digitais, de modo que o consumo de processamento seja baixo, sendo essa uma das vantagens de sua utilização. Em seguida, esses valores lidos serão enviados via comunicação SPI para o Raspberry Pi, retornando via SPI os sinais para controle dos atuadores que o PIC vai gerenciar.

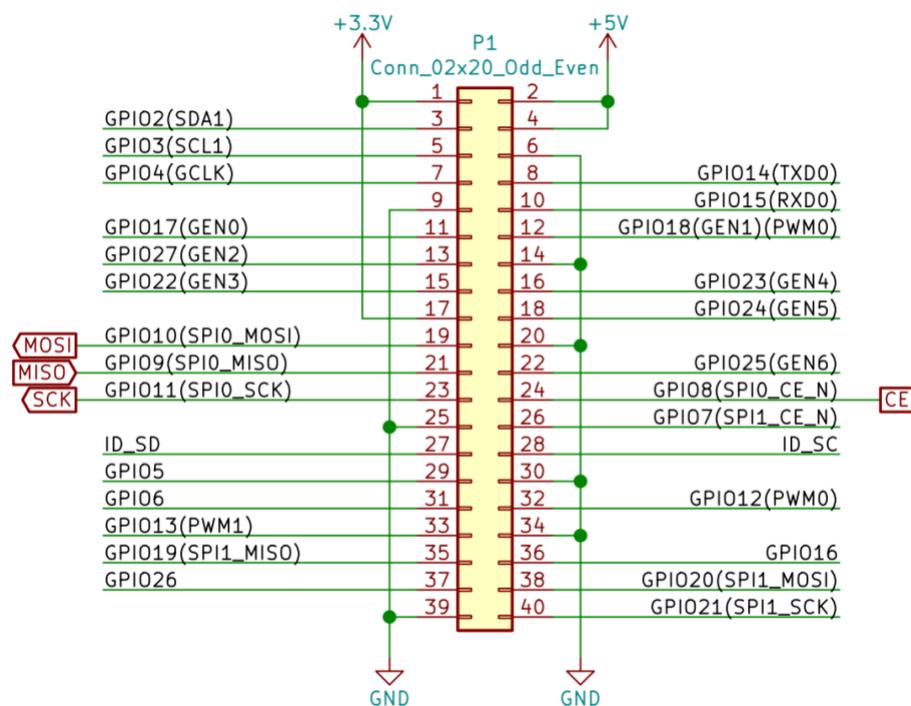
A conexão dos pinos de comunicação SPI do PIC18F4550 é composta por 4 pinos na *interface*: 1) sinal de *clock* no qual faz toda a sincronização da comunicação, pino 34; 2) mestre envia e escravo recebe (*master out slave in* - MOSI), o qual identifica que o mestre está enviando o pacote de dados, pino 33; 3) mestre recebe e escravo envia (*master in slave out* - MISO) o qual identifica que o escravo está enviando o pacote de dados, pino 26; 4) seleção de escravo (*slave select* - SS) que seleciona qual o escravo que o mestre quer se comunicar, pino 7.

3.2.1.3 Raspberry Pi 3 B V1.2

O Raspberry Pi 3 B v1.2 é um microcomputador que é instalado o Sistema Operacional (SO) *LINUX*, com nível de processamento elevado que faz operações aritméticas em frações de segundos. A responsabilidade de fazer os cálculos necessários dos atuadores é transferida para o Raspberry Pi, tendo em vista que o microcontrolador PIC18F4550 apresenta lentidão, tornando assim o processo mais versátil e eficaz. Além disso, o Raspberry Pi ajusta o tempo de injeção e avanço do ângulo de ignição em tempo real, conectando um teclado e *mouse bluetooth* junto com um monitor de conexão de *Interface Multimédia de Alta-Definição (HDMI – High-Definition Multimedia Interface)* para visualizar a *interface*.

A conexão dos pinos de comunicação SPI do Raspberry Pi está ilustrada na figura 30. São 4 pinos que compõe a *interface*: 1) sinal de *clock*, pino 23; 2) MOSI, pino 19; 3) MISO, pino 21; 4) SS, pino 24. Neste caso o mestre é o Raspberry Pi e o escravo é o PIC18F4550.

Figura 30 - Pinagem e conexões SPI do Raspberry Pi



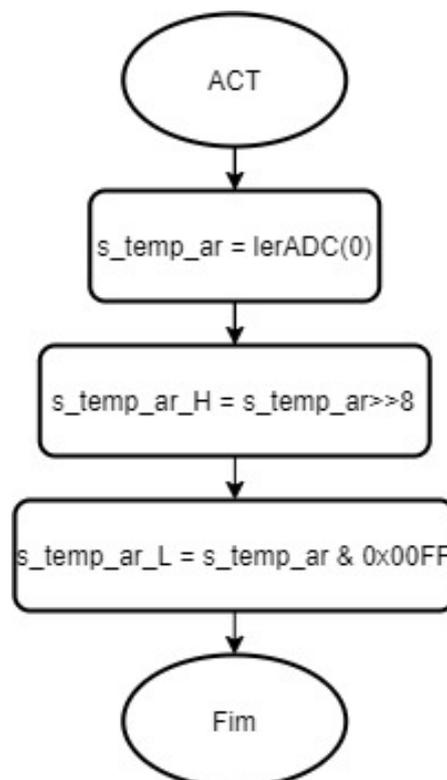
Fonte: Autor

3.2.2 Software do Projeto no PIC

A escolha da linguagem de programação do *software* do projeto, foi feita a partir da funcionalidade e métodos de operação, tendo a linguagem C, aquela com melhores resultados, que possui confiabilidade e seu código ocupa pouca memória. A linguagem C é versátil por exigir um alto nível de conhecimento sobre programação, que trabalha junto ao compilador que converte a linguagem C em linguagem hexadecimal, a linguagem de máquina.

O fluxograma que faz a leitura do sensor ACT, está ilustrado na figura 31. Através da entrada analógica zero, o PIC18F4550 armazena o valor lido em uma variável de 16 *bits*, depois divide o valor em duas partes, o *byte* mais significativo tem o nome de *high* e o menos significativo tem o nome *low*.

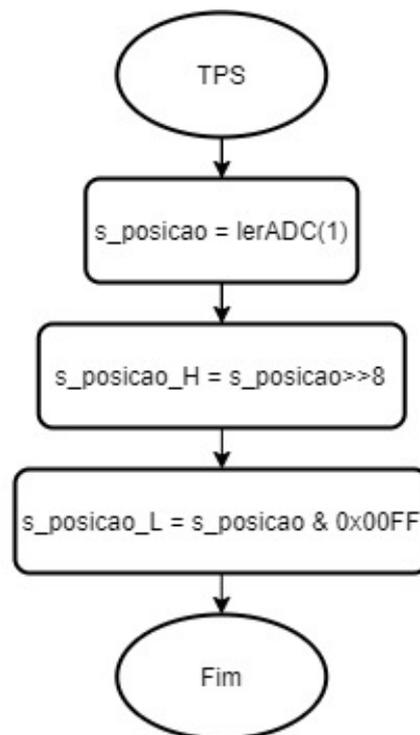
Figura 31 - Fluxograma da leitura do sensor ACT



Fonte: Autor

O processo de leitura do sensor de TPS, ilustrado na figura 32, é o mesmo do ACT e assim segue o mesmo princípio de funcionamento para leitura de sensores analógicos

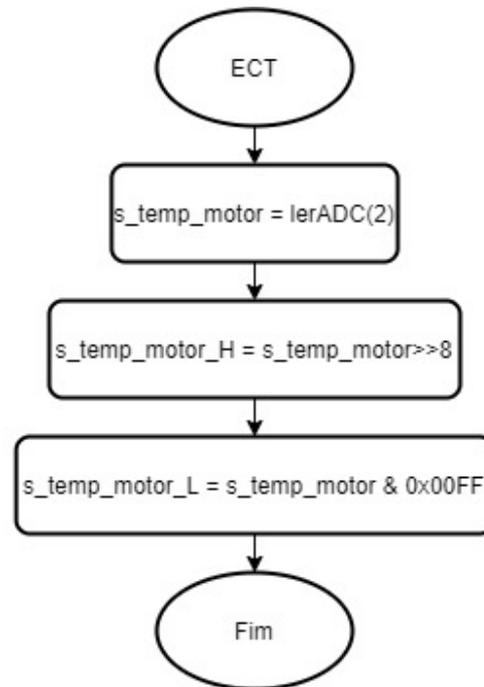
Figura 32 - Fluxograma de leitura do sensor TPS



Fonte: Autor

O fluxograma da leitura do sinal do sensor ECT, está ilustrado na figura 33. Como a leitura é de um sensor analógico, o fluxograma segue a mesma lógica de armazenar a leitura em uma variável e dividir a mesma em dois *bytes*.

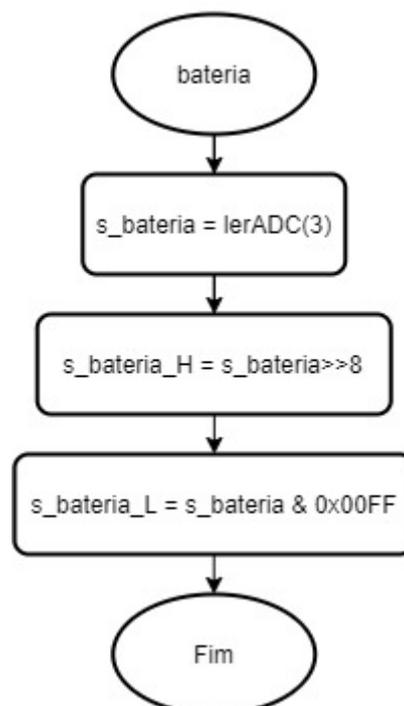
Figura 33 - Fluxograma do sensor ECT



Fonte: Autor

O fluxograma que faz a leitura da tensão de bateria que está ilustrado na figura 34. A base da lógica é a mesma da leitura para sensores analógicos.

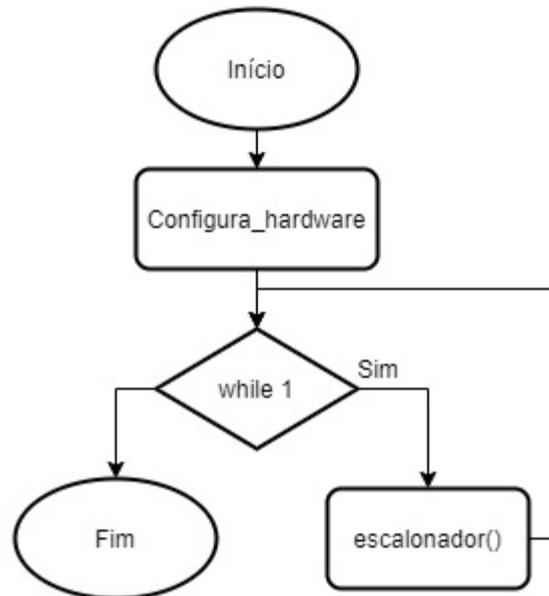
Figura 34 - Fluxograma de leitura da tensão de bateria



Fonte: Autor

O programa principal, ilustrado na figura 35, inicializa todos o sistema do *hardware* e depois entra no loop infinito sempre chamando a função do escalonador.

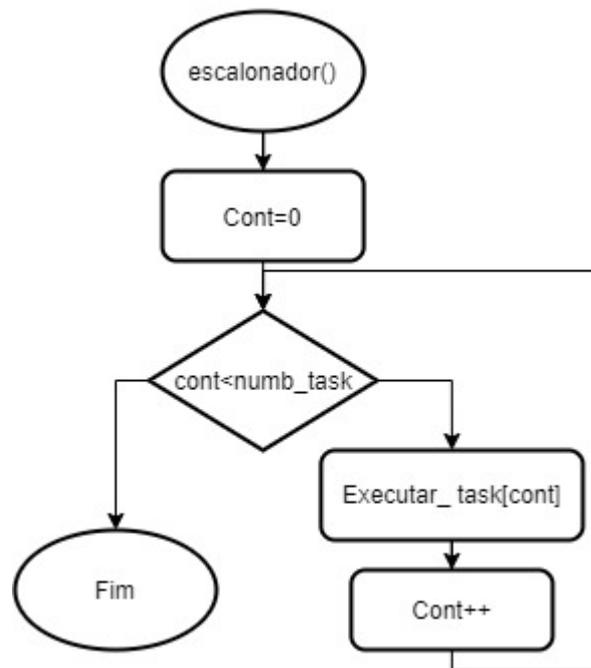
Figura 35 - Fluxograma do programa principal



Fonte: Autor

O escalonador funciona para organizar as subfunções de modo que cada função tenha um espaço de tempo determinado de ação. Quando o tempo da função acabar, o escalonador irá para a próxima função e assim sucessivamente. O fluxograma do escalonador está ilustrado na figura 36.

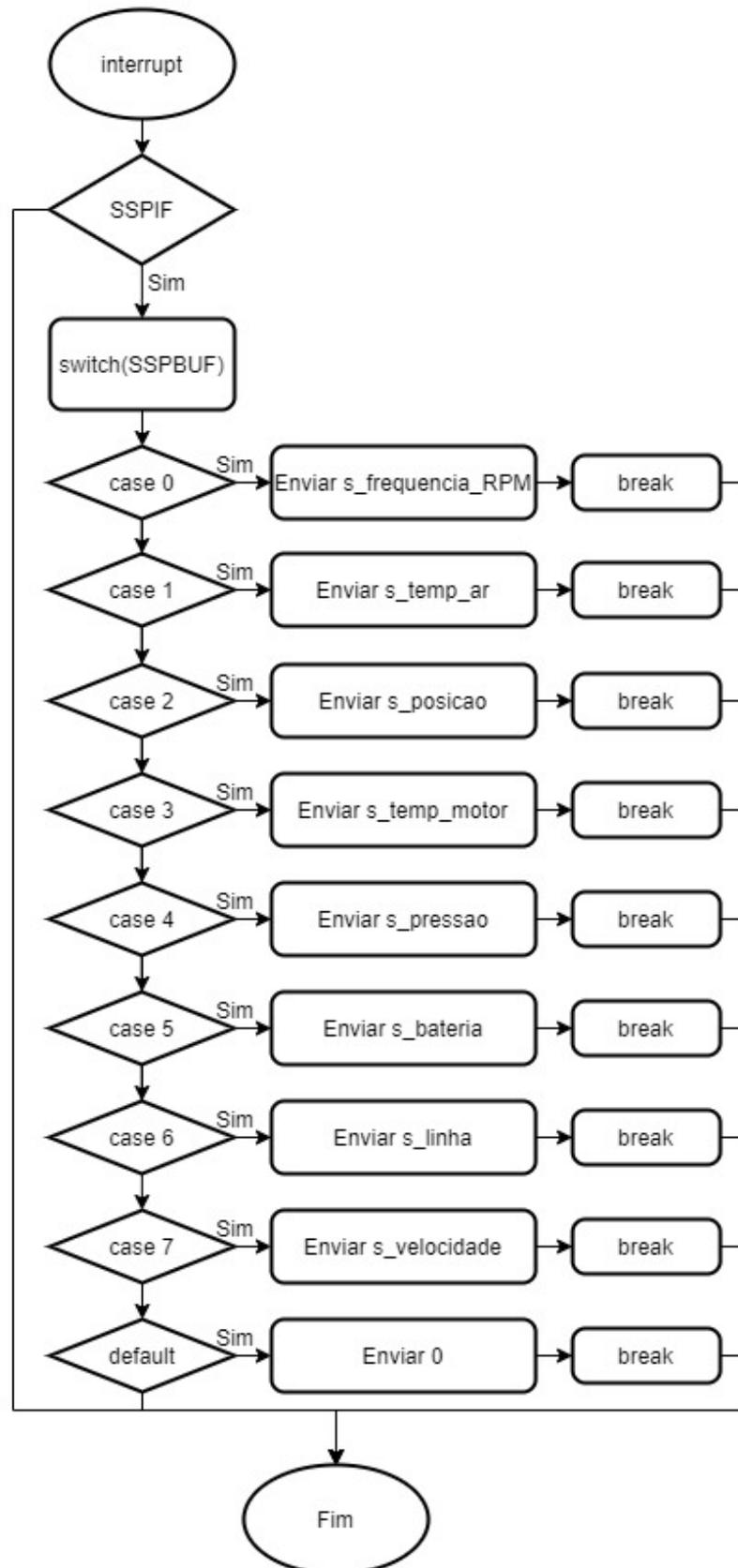
Figura 36 - Fluxograma escalonador



Fonte: Autor

O fluxograma de como os dados são enviados via SPI para o Raspberry Pi, está ilustrado na figura 37. Quando o Raspberry Pi manda o primeiro pacote ao PIC18F4550, acontece uma interrupção que lê os dados do barramento e retorna com os valores dos sensores analógicos e digitais.

Figura 37 - Fluxograma da comunicação SPI



Fonte: Autor

3.2.3 Software do Projeto no Raspberry Pi

No microcomputador Raspberry Pi é instalado um SO Raspbian, um *software* livre baseado em Debian e adaptado para o Raspberry Pi. O SO permite gerenciar arquivos, abrir programas, navegar na rede. Desta forma, o Raspberry literalmente tornou-se um computador com dimensões reduzidas comparável a de um cartão de crédito. Nele, é possível fazer programas usando ambientes de desenvolvimento integrado (IDE – Integrated Development Environment), usamos a IDE Thonny que é específico para desenvolver programas em linguagem de programação Python.

O *software* desenvolvido possui três características: 1) comunicação SPI; 2) conversão dos valores recebidos do microcontrolador, cálculo do tempo de ignição incluindo todo o controle de atuadores; 3) *interface*, que envolve a visualização e interação do usuário. A *interface* do mapa do tempo de injeção de combustível, está ilustrado na figura 38, que pode ser editável em tempo real.

Figura 38 - *Interface* do mapa do tempo de injeção de combustível

TPS/RPM	1000	1500	2000	2500	3000
10	2,5	2,2	1,9	1,7	1,6
20	3,5	3,1	2,4	2,2	2,0
40	5,5	5,1	3,6	2,9	2,5
80	5,8	5,5	4,8	4,2	3,7
100	6,2	5,9	5,3	4,9	4,4

gravar

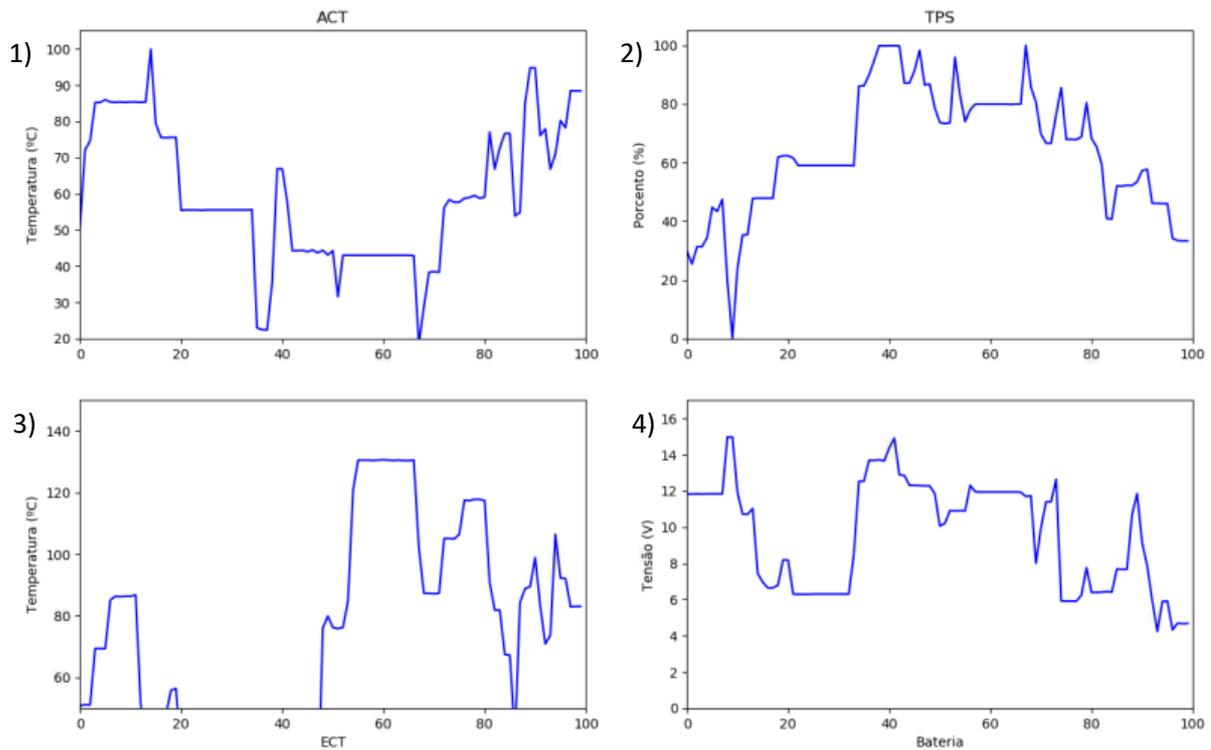
Quit

Fonte: Autor

A *interface* é uma janela onde pode ser exibido qualquer informação, também pode ter botões e caixas de texto para o usuário interagir do jeito que for necessário. Isso é possível através da biblioteca Tkinter, uma ferramenta que contém diversos recursos para adicionar *widgets* como botões, textos, figuras etc. Ao mostrar os gráficos dentro

da janela, foi utilizada a biblioteca Matplotlib, a qual traça diversos tipos de gráficos exigidos para a aplicação desejada. No projeto, foi utilizado um gráfico simples de eixo 'x' e 'y', tempo e tensão respectivamente para apresentar os valores lidos dos sensores analógicos, ilustrados na figura 39.

Figura 39 - Interface dos gráficos dos sensores



Fonte: Autor

Os gráficos demonstram os comportamentos de diferentes sinais: 1) sensor de temperatura do ar de admissão; 2) sensor TPS; 3) sensor de temperatura do líquido do arrefecimento do motor. 4) tensão da bateria;

3.3 Detalhes Finais e Comentários do Projeto

Não foi possível realizar os testes do VOLKSWAGEN GOL monoponto 1.0 1995 da praça técnica da FATEC Santo André, pela indisponibilidade do veículo no qual os testes somente foram realizados em ambientes simulados.

Os gráficos foram gerados separados do programa principal pelo motivo do código desenvolvido não deixar rodar o mapa junto com os gráficos.

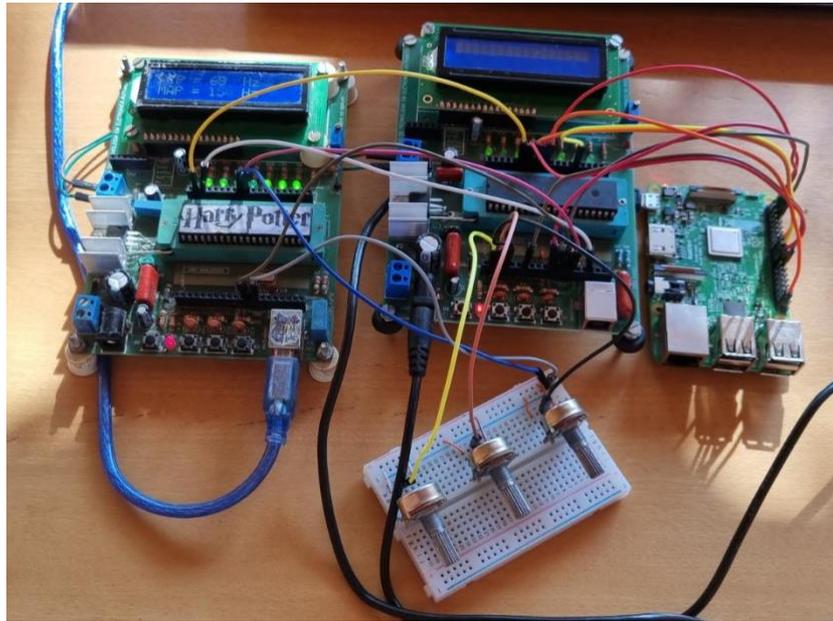
A parte dos cálculos do tempo de injeção e avanço do ângulo de ignição e outros tipos de controle não foram desenvolvidos por dar foco no funcionamento da comunicação SPI e desenvolvimento da parte gráfica.

Foi pensado inicialmente em desenvolver o trabalho com o Raspberry Pi Zero W, porém, o processamento foi insuficiente trocando pelo Raspberry Pi 3 B. O Raspberry Pi Zero 2 W também entrou em pauta que seria uma solução para o processamento, porém, houve a falta de peça no mercado, impossibilitando a compra do componente.

4 TESTES E ANÁLISE DOS RESULTADOS

Não foi possível realizar os testes no veículo pela indisponibilidade do Gol monoponto 1.0 1995. Esta é a razão da simulação dos sensores analógicos e digitais emitidos por outro PIC18F4550, uma simulação errônea no sentido de os sinais serem uma aproximação dos sinais reais do veículo, que possui faixa de operação e limites de frequências. O projeto pronto, está ilustrado na figura 40. Na placa a esquerda da figura, temos o módulo que simula os sinais do veículo. Nele é possível alterar os valores de rotação do motor e a frequência do sensor MAP através dos botões fixados na placa. Ao centro da imagem, o microcontrolador que está recebendo os valores dos sensores do motor: RPM e MAP vindos da placa de simulação; TPS, ACT e ECT vindos dos potenciômetros na parte inferior da figura. À direita, o Raspberry Pi está se comunicando com o microcontrolador para fazer os cálculos que exige um processamento maior.

Figura 40 - Teste da ECU com os sinais simulados do veículo



Fonte: Autor

O teste para verificar se os valores dos sensores lidos do simulador estão coerentes, a figura 41 ilustra, uma tabela com alguns valores gerados de frequência ao lado esquerdo e como resultado da leitura temos os valores ao lado direito da tabela. De acordo com os testes, há uma tolerância irrelevante no resultado dos valores, como no sinal de PIP gerado na frequência de 180 Hertz (Hz) e teve como resultado da leitura um valor de 186 Hz.

Figura 41 – Valores simulados dos sensores PIP e MAP

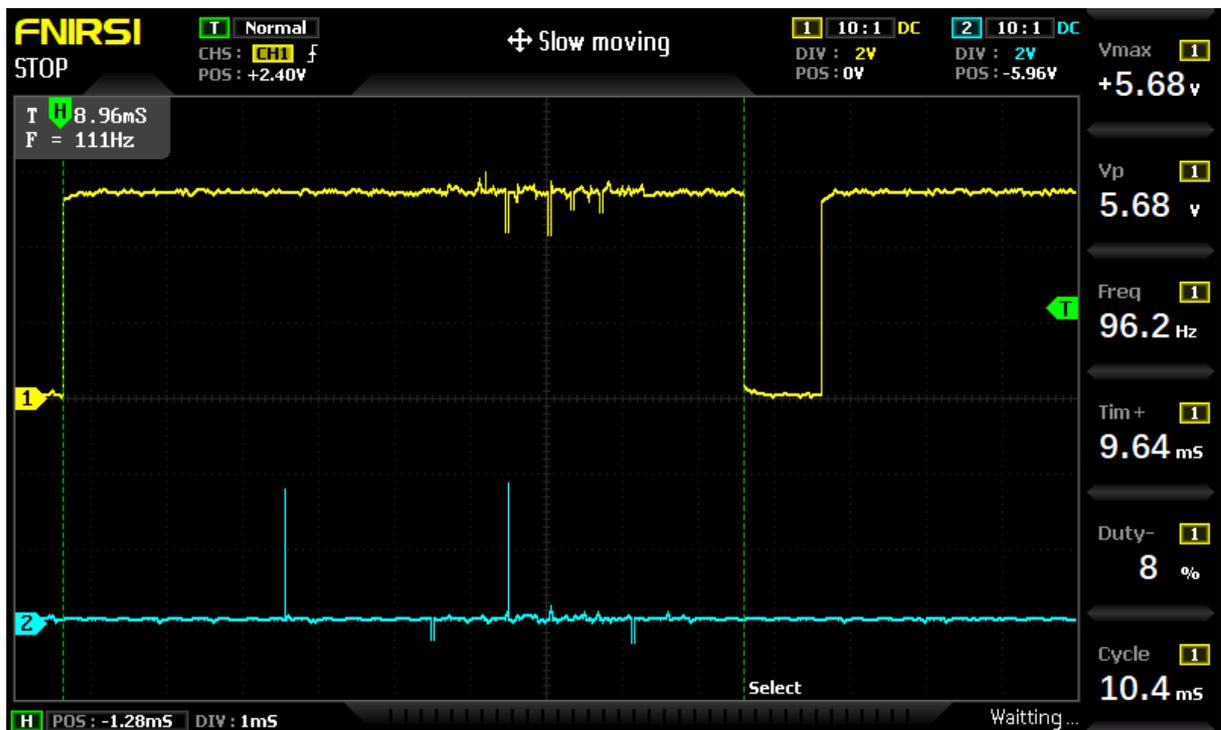
SIMULADOR		ECU	
PIP (Hz)	MAP (Hz)	PIP (Hz)	MAP (Hz)
20	30	20	30
40	60	40	61
70	80	70	83
100	100	101	104
120	110	121	114
150	120	153	126
180	130	186	136
200	150	200	158

Fonte: Autor

O tempo de processamento da comunicação SPI, que está ilustrado na figura 42, consome resulta em 8,96 milissegundos. Para este valor, a comunicação está

configurada com o clock em 1 MHz, com a transmissão de 12 *byte* de dados. Logo, é importante considerar no projeto final a latência entre de tempo entre o PIC enviar e receber os dados do Raspberry, o que pode ser um problema no controle do veículo. Como não forão realizados testes práticos no veículo não foi possível avaliar o impacto desse atraso.

Figura 42 – Tempo de transmissão da comunicação SPI



5 CONCLUSÃO

Observando o que foi proposto no início do projeto, pela ausência física do veículo, o projeto só foi testado simuladamente, cujos resultados que estão discutidos no capítulo 4, demonstram a capacidade do projeto ser testado no veículo em outra ocasião. Os testes demonstram que a parte do *software* é funcional e teria condições de fazer o veículo ligar, e a parte da leitura dos sinais dos sensores demonstra que o *software* consegue capturar os valores com uma certa precisão, e consegue ler sinais analógicos e digitais. O projeto conta com a comunicação SPI que obteve um ótimo funcionamento, com uma frequência de comunicação elevada. Conta também com o *software de interface* e interação muito bem apresentados.

5.1 Propostas Futuras

Para que valide a ideia da ECU de injeção eletrônica programável, é preciso:

- Desenvolver o *hardware* da ECU;
- Conectar no veículo;
- Avaliar o impacto do delay de comunicação no SPI durante o controle do motor em altas rotações. Estudar alternativas de trocas mais rápidas de dados.
- Fazer os devidos testes de funcionamento do motor.
- Completar o *software* colocando o avanço da ignição, o tempo de injeção, e apresentar os gráficos na mesma janela que o mapa da injeção.
- Implementar comunicação CAN
- Implementar comunicação OBD II

REFERÊNCIAS

ALVES, Matheus Damasceno; ANTOS, Barbara Pereira. **SISTEMA DE INJEÇÃO MONOPONTO UTILIZANDO MYRIO E LABVIEW**. Santo André, 2019. 10 p.

BRUNETTI, Franco Brunetti. **Motores de Combustão Interna**: volume 1. São Paulo: Bluncher, 2012.

DENATRAN, frota 2011

FUELTECH. Documento eletrônico. Disponível em <https://fueltech.com.br/pages/sobre-nos>>. Acesso em 03 Nov 2022.

GAZ. Documento eletrônico. Disponível em <https://www.gaz.com.br/santa-cruzense-conta-fez-um-sonho-se-tornar-uma-empresa/>>. Acesso em 03 nov 2022.

HURTADO, Diego Kellermann; Souza, Alfeu de Arruda. **A Evolução do Sistema de Injeção de Combustível em Motores Ciclo Otto**: uma análise crítica desde suas implicações no meio ambiente à regulamentação legal no sistema normativo pátrio. Santa Catarina: GPDS, ISSN 1981-3694

JUNIOR, Waldir Nunes dos Santo Junior. **Um Estudo Sobre o Impacto Ambiental dos Combustíveis Automotivos na Qualidade do Ar na RMSP – Região Metropolitana de São Paulo Período de 2001 a 2005**. São Paulo: 2006.

MEGA manual. Documento eletrônico. Disponível em <http://www.megamanual.com/MSFAQ.htm>>. Acesso em 03 Nov 2022.

MEGASQUIRT. Documento eletrônico. Disponível em <https://megasquirt.info/history/>>. Acesso em 03 nov 2022.

PORTAL arauto. Documento eletrônico. Disponível em <https://www.portalarauto.com.br/Pages/199154/conheca-a-historia-do-santa-cruzense-criador-do-corvette-mais-rapido-do-mundo>>. Acesso em 03 nov 2022.

VON HIPPEL, Eric. Learning from open-source software. **MIT Sloan management review**, v. 42, n. 4, p. 84, 2001.