

UNIVERSIDADE FEDERAL DO PARANÁ

ALEXANDRE GABRIEL MOHR

**DESENVOLVIMENTO DE SISTEMA PARA VERIFICAÇÃO DO APARELHO DE
LUBRIFICAÇÃO EM VEÍCULOS PESADOS**

CURITIBA

2019

ALEXANDRE GABRIEL MOHR

**DESENVOLVIMENTO DE SISTEMA PARA VERIFICAÇÃO DO APARELHO DE
LUBRIFICAÇÃO EM VEÍCULOS PESADOS**

Trabalho de conclusão de curso apresentado ao Curso de Engenharia Elétrica com Ênfase em Sistemas Eletrônicos Embarcados da Universidade Federal do Paraná como requisito à obtenção do título do grau de bacharelado em Engenharia Elétrica.

Orientador: Prof. Dr. Eduardo Kazumi Yamakawa

CURITIBA

2019

TERMO DE APROVAÇÃO

ALEXANDRE GABRIEL MOHR

DESENVOLVIMENTO DE SISTEMA PARA VERIFICAÇÃO DO APARELHO DE LUBRIFICAÇÃO EM VEÍCULOS PESADOS

Trabalho de conclusão de curso apresentado como requisito parcial para à obtenção do grau de Engenheiro Eletricista com Ênfase em Sistemas Eletrônicos Embarcados, do Setor de Tecnologia da Universidade Federal do Paraná, pela seguinte banca examinadora:

Prof. Dr. Eduardo Kazumi Yamakawa

Departamento de Engenharia Elétrica, UFPR

Prof. Dr. Armando Heilmann

Departamento de Engenharia Elétrica, UFPR

M.Sc. Vithor Bernardo Nypwipwy

Departamento de Engenharia Elétrica, UFPR

Curitiba, 6 de dezembro de 2019.

AGRADECIMENTOS

Em primeiro lugar, agradecer a minha família que sempre esteve presente e forneceu apoio essencial durante toda a extensão desta longa caminhada, servindo como refugio nos momentos turbulentos. Agradecer aos meus amigos, sempre apoiando e trilhando o caminho junto a mim, alegrando e deixando mais leve a rotina do dia a dia.

Não esquecer de meus colegas de trabalho, que durante todo este tempo no decorrer do curso foram compreensivos com relação as atividades realizadas na faculdade, sempre prontos para ajudar em qualquer necessidade. Por fim, agradecer a todos os professores do departamento que fizeram a diferença na minha formação profissional e pessoal, dedicando-se a ministrar as matérias com empenho e didática, sempre buscando o maior aproveitamento dos alunos na disciplina.

Fazer, todos os dias, as mesmas coisas e esperar resultados diferentes é a maior prova de insanidade.

Albert Einstein

LISTA DE ILUSTRAÇÕES

FIGURA 2.1 – Arquitetura interna do microcontrolador ESP8266.	16
FIGURA 2.2 – ESP8266 NodeMCU.	17
FIGURA 2.3 – Relação de pinos para o ESP8266 NodeMCU V3	18
FIGURA 2.4 – Procedimento genérico de método científico.	19
FIGURA 2.5 – Funcionamento de um medidor analógico de temperatura	20
FIGURA 2.6 – Funcionamento de um medidor digital de temperatura	21
FIGURA 2.7 – Sinais analógicos e digitais no tempo	22
FIGURA 2.8 – Comparação da variação de resistência de termistores NTC e PTC	24
FIGURA 2.9 – Equação e circuito elétrico que representa o comportamento do sensor R300.	25
FIGURA 2.10 – Circuito exemplo e curva do sensor R300.	25
FIGURA 2.11 – Sensores de pressão automotivos capacitivos e curva de res- posta.	26
FIGURA 2.12 – Componentes da válvula solenoide.	26
FIGURA 2.13 – Primeiro motor a combustão 4 tempos criado por Otto	27
FIGURA 2.14 – Fluxo de funcionamento de motores	28
FIGURA 2.15 – Ciclo de combustão em um motor a diesel	29
FIGURA 2.16 – Sensores e atuadores utilizados no controle de motores a com- bustão interna.	30
FIGURA 2.17 – Sistema de arrefecimento por líquido.	31
FIGURA 2.18 – Sistema de lubrificação sob pressão	32
FIGURA 2.19 – Temperatura do pistão em motores a combustão	33
FIGURA 3.1 – Sistema de óleo do motor.	35
FIGURA 3.2 – Fluxo de funcionamento geral do sistema.	36
FIGURA 3.3 – Procedimento de teste de motores.	37
FIGURA 3.4 – Curva de resposta do sensor.	38
FIGURA 3.5 – Resposta dos sensores na partida do motor.	39
FIGURA 3.6 – Resposta dos sensores com o acionamento da válvula.	40

FIGURA 3.7 – Filtro RC primeira ordem.	41
FIGURA 3.8 – Resposta do filtro RC de primeira ordem $\omega_c = 10Hz$	41
FIGURA 3.9 – Resposta dos sensores na partida do motor filtrados.	42
FIGURA 3.10 – Resposta dos sensores com o acionamento da válvula filtrados.	42
FIGURA 3.11 – Comparação do sinal na entrada e na saída do filtro RC.	43
FIGURA 3.12 – Fluxograma do primeiro laço do software.	44
FIGURA 3.13 – Fluxograma do segundo laço do software.	45
FIGURA 3.14 – Fluxograma do terceiro laço do software.	46
FIGURA 3.15 – Divisor de tensão resistivo utilizado nas entradas digitais.	47
FIGURA 3.16 – Módulo Relé 5V 2 Canais	48
FIGURA 3.17 – Primeiro esquemático do dispositivo	49
FIGURA 4.1 – Circuito em placa de prototipagem	50
FIGURA 4.2 – Gráficos gerados pela API <i>ThingSpeak</i>	52
FIGURA 4.3 – Esquemático da placa principal	53
FIGURA 4.4 – Esquemático da placa LED	54
FIGURA 4.5 – Layout da placa principal e LED	54
FIGURA 4.6 – Vista das placas impressas	55
FIGURA 4.7 – Vista 3D da placa principal	55
FIGURA 4.8 – Vista 3D da placa LED	56
FIGURA 4.9 – Caixa plástica universal para circuitos elétricos	56
FIGURA 5.1 – Relação de materiais utilizados no projeto	58

LISTA DE ABREVIATURAS E SIGLAS

IoT	<i>Internet of Things</i>
CLP	Controlador Lógico Programável
PCJ	<i>Piston Cooling Jet</i>
CNI	Confederação Nacional da Indústria
LED	Diodo emissor de luz
USB	<i>Universal Serial Bus</i>
SoC	<i>System On Chip</i>
PMBOK	<i>Project Management Body of Knowledge</i>
API	<i>Application Programming Interface</i>
CPU	Unidade Central de Processamento
RAM	<i>Random Access Memory</i>
GPIO	<i>General Purpose Input/Output</i>
NTC	<i>Negative Temperature Coefficient</i>
PTC	<i>Positive Temperature Coefficient</i>
ECU	<i>Electronic Control Units</i>

SUMÁRIO

1	INTRODUÇÃO	10
1.1	JUSTIFICATIVA	11
1.2	OBJETIVOS	12
1.2.1	Objetivo Geral	12
1.2.2	Objetivos Específicos	12
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	SISTEMAS ELETRÔNICOS EMBARCADOS	15
2.1.1	MICROCONTROLADORES	15
2.1.1.1	ESP8266 NodeMCU	17
2.1.1.2	PLATFORMIO	18
2.2	INSTRUMENTAÇÃO ELETRÔNICA	19
2.2.1	Sensores	21
2.2.1.1	Sensores de temperatura	23
2.2.1.2	Sensores de pressão automotivos	25
2.3	VÁLVULAS SOLENOIDES	26
2.4	MOTORES A COMBUSTÃO INTERNA	27
2.4.1	Motores movidos a combustão Diesel	28
2.4.2	Sistemas eletrônicos automotivos	29
2.4.3	Sistema de lubrificação e arrefecimento	30
2.4.4	Sistema de <i>Piston Cooling Jet</i>	32
3	DESENVOLVIMENTO	34
3.1	ANÁLISE DO PROBLEMA A SER SOLUCIONADO	34
3.2	CIRCUITO DE ÓLEO DO MOTOR	35
3.2.1	Diagrama funcional do sistema	36
3.3	ANÁLISE DA RESPOSTA DOS SENSORES E ATUADORES	38
3.3.1	Primeiro teste de motor instrumentado	39
3.3.2	Segundo teste de motor instrumentado	41
3.4	PROGRAMAÇÃO DO MICROCONTROLADOR	43

	9
3.5 ESQUEMÁTICO DO CIRCUITO ELÉTRICO	47
4 RESULTADOS	50
4.1 INTEGRAÇÃO DAS FUNÇÕES DO MICROCONTROLADOR	50
4.1.1 Teste do comportamento esperado do microcontrolador	51
4.2 PLACA DE CIRCUITO IMPRESSO DO DISPOSITIVO	53
5 CONCLUSÃO	57
5.0.1 Trabalhos futuros	58
REFERÊNCIAS	60
6 APÊNDICE A	61
6.1 CÓDIGO DO MICROCONTROLADOR	61

1 INTRODUÇÃO

Nos últimos anos a sociedade vem vivenciado um crescimento acelerado da indústria de bens e consumo, e junto a este fator, a indústria automotiva de veículos pesados teve sua demanda elevada na mesma proporção da necessidade de transporte e escoamento da produção. Segundo a série histórica Associação Nacional dos Fabricantes de Veículos Automotores, a fabricação de veículos pesados mais que dobrou dos anos 2000 para 2019, isso gerou a necessidade de as empresas serem capazes de entregar cada vez mais veículos, para atingir maiores níveis de produção são utilizadas técnicas de produção rápidas, econômicas e eficientes.

As técnicas para acelerar o processo de produção datam de muito antes, popularizado pelo Fordismo, termo criado e implementado por Henry Ford em 1914, baseava-se na utilização da mão de obra ociosa em mão de obra útil, através da divisão do processo produtivo em pequenas tarefas organizadas em uma linha de montagem de fluxo contínuo. Este modelo foi melhorado pelos subsequentes e atuais conceitos de produção enxuta, baseado no sistema consolidado pela *Toyota Motor Company* em 1960. O Toyotismo impõe alguns conceitos diferentes do anterior, como maior eficiência do processo com o detalhamento minucioso de cada operação a ser realizada e a implementação de verificações e validações durante o processo produtivo. (WOMACK; JONES, 2004)

O Toyotismo define que os erros ocorridos durante a produção não são aleatórios e sim previsíveis e solucionáveis, porem qualquer método de melhoria ou solução deve ser testado e comprovado antes de ser implementado. Com processos tão ágeis, e uma necessidade constante de produção, é extremamente custoso lidar com problemas que podem acarretar mudanças na linha de produção, deste modo a solução para qualquer problema deve ser assertiva e bem planejada. (WOMACK; JONES, 2004)

Defeitos e problemas de qualidade detectados no produto, muitas vezes, exigem que novos processos ou testes sejam implementados em linhas de montagem, e isso demanda a utilização das mais atuais definições da indústria, como por exemplo, os conceitos de Indústria 4.0 e *Internet of Things* (IoT). Com isso os dispositivos eletrônicos ganham espaço na indústria, proporcionando soluções eficientes e robustas, transferindo a tarefa manual do operador para verificações utilizando microcontrola-

dores, sensores e atuadores. Testes automatizados vem para garantir a qualidade e funcionamento de todos os componentes do produto e são realizados em diversos pontos da cadeia produtiva, sendo eles cruciais para determinar que todas as funcionalidades do produto serão válidas e duradouras conforme o projetado.

Testes de funcionalidade se tornam cruciais no caso dos sistemas envolvidos no funcionamento de motores a combustão interna, devido à grande quantidade de energia envolvida durante o processo da combustão, o mal funcionamento pode acarretar danos irreversíveis e irreparáveis. Portanto é necessário haver teste rigorosos de teste de funcionalidade dos componentes do motor como sistema de alimentação de combustível, arrefecimento, alimentação elétrica e lubrificação, que devem ser verificados durante o funcionamento simulado do conjunto completo. Para manter a velocidade de escoamento da linha de produção, tais testes devem ser realizados de maneira rápida e precisa, permitindo que os motores sejam validados e liberados para a montagem final no veículo. Com os requisitos de confiabilidade e agilidade, os sistemas eletrônicos de teste são amplamente utilizados nesta etapa da produção.

Em suma este projeto visa fornecer uma ferramenta para utilização em linhas de produção para o teste de sensores de pressão do óleo do motor utilizados em veículos automotores peados, onde tais componentes são de vital importância para o funcionamento e conservação dos motores a combustão, já que é ele garante o funcionamento do sistema de lubrificação que impede o sobreaquecimento de peças mecânicas causado pelo atrito, evitando danos severos e irreparáveis ao conjunto.

1.1 JUSTIFICATIVA

Incluir novos processos de verificação em linhas de montagem podem ser feitos de duas maneiras. Primeiro, criando um processo em que o próprio operador verifica o item e avalia se está de acordo ou não com os requisitos e especificações, esta opção vem em conjunto com o risco de falha humana por parte do operador, o qual vai realizar repetidamente a mesma rotina e pode vir a permitir que um produto com falha seja liberado. Outra opção é utilizar de sistemas e processos eletrônicos, os quais são livres do erro humano após seu desenvolvimento, porem para incluir um novo sistema eletrônico existem custos elevados em um primeiro momento, como a compra de equipamentos ou reprogramação de Controladores Lógicos Programáveis (CLP).

Este projeto tem como foco o fornecimento de um sistema eletrônico que verifique condições de funcionamento do motor a combustão, utilizando um microcontrolador de baixo custo, e forneça ao final do teste uma confirmação ao CLP da fábrica, economizando os custos de reprogramação no caso da utilização somente de um CLP como controlador e evitando que erros humanos sejam cometidos caso o processo de verificação fosse realizado manualmente pelo operador. Atualmente o tempo despendido com correção de erros detectados nas linhas de produção é de aproximadamente 25% do tempo da produção, logo a economia de tempo gerada pelos processos automatizados é refletida diretamente na redução do custo da montagem. (WOMACK; JONES, 2004)

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo principal é o desenvolvimento de um sistema embarcado que auxilia e automatiza o teste de funcionamento do sistema de lubrificação do motor, realizado após o processo de montagem. O dispositivo vai receber os sinais de pressão do óleo, acionamento da válvula solenoide de circulação do óleo, acionamento da solenoide do motor de partida e, a partir disto, analisar se o funcionamento é o esperado para o sistema, retornando para o CLP da fábrica se o sistema de lubrificação do motor está funcionando corretamente.

1.2.2 Objetivos Específicos

- Análise do comportamento do sinal do sensor de pressão do óleo do motor e do acionamento da válvula solenoide que controla o fluxo de óleo do sistema de *Piston Cooling Jet* (PCJ);
- Escolha da melhor opção de microcontrolador para o projeto, que seja de custo compatível e tenha entradas e saídas (Analógicas e Digitais) suficientes para o funcionamento do dispositivo;
- Criação do software de controle do dispositivo e comunicação, a ser desenvolvido em linguagem C e posteriormente embarcado no microcontrolador;
- Integração do microcontrolador com os sinais de sensores e atuadores, realizando o correto dimensionamento das tensões e correntes do circuito;

- Comunicação via *WiFi* com *API web* para visualização dos dados;
- Validação do sistema utilizando o sinal do sensor e atuador simulados, transmitindo o resultado do teste para o microcontrolador, validando o funcionamento do dispositivo;
- Analisar o custo do produto frente ao custo das falhas reportadas em campo;
- Revisão teórica dos tópicos abordados e do desenvolvimento do produto.

2 FUNDAMENTAÇÃO TEÓRICA

A busca por soluções alinhadas aos conceitos de eficiência e agilidade é cada vez mais procurada pelas empresas, segundo a Confederação Nacional da Indústria (CNI), 48% das empresas industriais do País utilizam pelo menos uma tecnologia digital, com isso observa-se que as oportunidades de implementação de conceitos de automação são relevantes. A motivação que leva as industriais a buscar por soluções digitais são principalmente para reduzir custos operacionais (54%) e aumentar a produtividade (50%). (ESTADÃO, 2016)

Para que a solução desenvolvida seja adequada para o problema e implementável, deve-se analisar a origem do problema detectado e qual a melhor maneira de implementar uma solução digital para o problema, para isso é necessário entender como o objeto a ser analisado e suas possíveis soluções funcionam. Segundo o guia *Project Management Body of Knowledge (PMBOK)* – Um Guia do Conhecimento em Gerenciamento de Projetos, um projeto é definido como "um esforço temporário empreendido na criação de um produto, serviço ou resultado único. Sua natureza temporária indica que ele tem início e término bem definidos e que o término é atingido quando seus objetivos são alcançados ou quando o projeto é encerrado". O ciclo de vida de um projeto é definido pelas quatro fases: (PINAZZA, 2017)

- Iniciar o projeto;
- Organização e preparação;
- Execução do trabalho, e;
- Encerramento do projeto.

Dentro da etapa de organização e preparação e a fim de estabelecer os conceitos necessários para o desenvolvimento da solução proposta neste trabalho, a fundamentação teórica fornece a base de conhecimento para que seja possível estudar a melhor forma de amadurecer o projeto e definir as metodologias a serem implementadas durante o desenvolvimento, de modo que o capítulo está dividido nas seguintes seções.

A seção 2.1 trata conceitos de sistemas eletrônicos embarcados e como podem ser utilizados para automação industrial, bem como a linguagem de programação a

ser utilizada no desenvolvimento do sistema. A seção 2.2 aborda sobre a instrumentação eletrônica e o comportamento dos sensores e atuadores envolvidos no projeto, fornecendo a base para determinar como tratar os sinais oriundos destes componentes. Por fim a seção 2.3 trata dos conceitos relativos ao funcionamento de motores a combustão interna e tópicos importantes a serem compreendidos para determinar o funcionamento esperado do sistema.

2.1 SISTEMAS ELETRÔNICOS EMBARCADOS

Um sistema embarcado por basicamente é uma combinação de hardware e software, que pode conter outras peças externas mecânicas e móveis, designadas para realizar uma função específica. A propriedade principal que distingue os sistemas embarcados é a certeza de que software não confiável jamais será executado nele. Não se pode baixar novas aplicações no forno micro-ondas, seu software está armazenado na memória interna do aparelho. Essa característica, alinhada ao seu custo cada vez mais reduzido, torna a tecnologia de sistemas eletrônicos embarcados cada vez mais presente e difundida no mercado. (TANENBAUM, 2009)

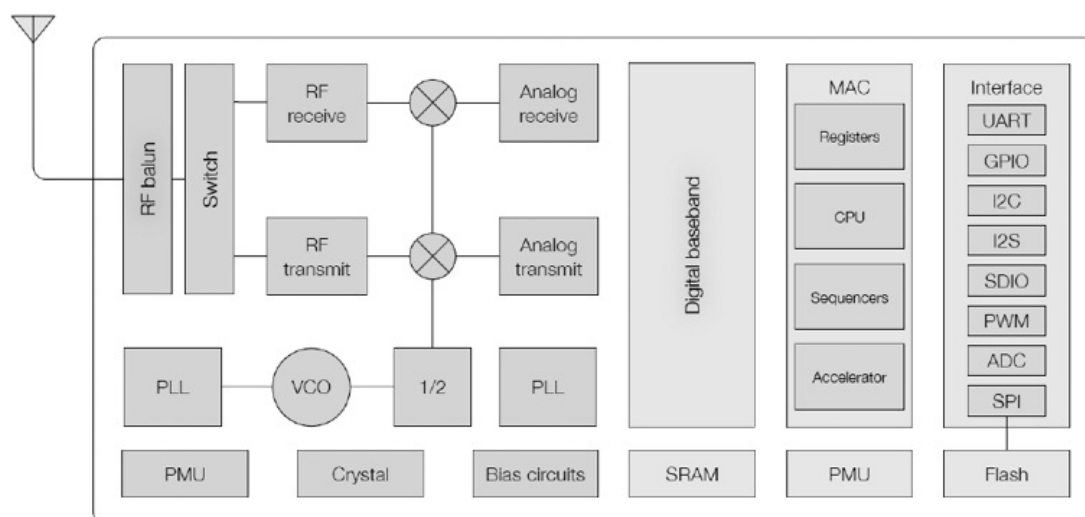
Alguns exemplos de sistemas embarcados incluem geladeiras, televisões, relógios, automóveis, máquinas industriais e sistemas de segurança. Na indústria, a eletrônica embarcada encontra um forte aliado, a atual revolução da indústria 4.0 que vem fomentando ainda mais o desenvolvimento de sistemas que possam auxiliar na automação de tarefas, comunicação entre equipamentos e processos dentro do ambiente de manufatura. Estes sistemas recebem informações de sensores e, através de atuadores, conseguem controlar o processo, se comunicando com outros sistemas e em alguns casos interagindo com o usuário, permitindo que o processo seja o mais eficiente possível.

2.1.1 MICROCONTROLADORES

Por volta de 1970 as produtoras de microprocessadores *Intel* e *Texas Instruments* começaram uma disputa para dominar o mercado de componentes para sistemas eletrônicos embarcados, os quais vinham sendo empregados para automatizar tarefas que antes eram realizadas manualmente. Com o avanço do uso em aplicações de automação cada vez mais complexas, os microcontroladores passaram a integrar cada

vez mais funções e se transformam em componentes chamados de *System-on-Chip* (SoC) ou placas de desenvolvimento, embarcando não só o microcontrolador responsável pelo processamento dos dados, mas também interfaces de entrada e saída para comunicação, como *Universal Serial Bus* (USB), *Ethernet* e *WiFi*, memória RAM e circuito de *clock* integrado. Um microcontrolador com todas essas funcionalidades integradas é capaz de atender a um número de aplicações sem precedentes, e um exemplo de como estes componentes se tornaram complexos está ilustrado na figura 2.1, mostrando que há um conjunto completo de componentes para processamento (CPU + memória), interfaces de entrada e saída para diversos tipos de protocolos e aplicações, e comunicação. (OLIVEIRA, 2017)

FIGURA 2.1 – Arquitetura interna do microcontrolador ESP8266.



Fonte: (OLIVEIRA, 2017)

Com o aumento da utilização e demanda, os preços das placas integradas foram reduzindo, e os circuitos de baixo custo passaram a ser viáveis para a implementação. Como mencionado anteriormente, a disputa pelo mercado de placas de desenvolvimento gera uma grande quantidade de opções quando se trata destes dispositivos, sendo de fundamental importância a comparação das opções e suas respectivas funcionalidades para determinar qual o melhor custo benefício para o projeto.

Dentre as opções que temos disponíveis no Brasil, algumas delas são o Arduino Uno, MSP430 LaunchPad, ESP8266 e Raspberry Pi3. A comparação das funcionalidades de cada placa e suas limitações é um dos pontos mais importantes a se considerar na avaliação das placas, parâmetros como sua capacidade de processamento, me-

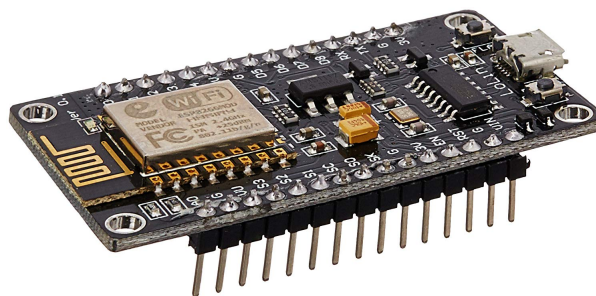
moria, quantidade de entradas e saídas digitais e analógicas, devem ser analisados para verificar se são compatíveis com as necessidades do projeto, o que pode afetar diretamente no custo e complexidade do mesmo.

2.1.1.1 ESP8266 NodeMCU

O microcontrolador ESP8266, produzido pela fabricante chinesa *Espressif*, é um microcontrolador de 32 bits que inclui um núcleo microprocessado Tensilica L106, que funciona na frequência-padrão de 80 MHz, podendo chegar a 160 MHz. O processamento da pilha de protocolos WiFi usa 20% da capacidade de processamento desse processador. Com isso, 80% dessa capacidade pode ser utilizada em aplicações do usuário. A tensão nominal de funcionamento é de 3,3 volts e possui 17 interfaces chamadas de *General Purpose Input Output* (GPIO), que podem ser configuradas como entradas ou saídas digitais. Tem também quatro interfaces de saída pulso modulado e uma entrada analógica com 10 bits de precisão e faixa de 0 a 3,3 volts. (OLIVEIRA, 2017)

O módulo mais popular é o módulo chamado NodeMCU, que agrega com ele uma plataforma de desenvolvimento em linguagem Lua. Essa linguagem, desenvolvida no Brasil por professores da PUC-Rio, é interpretada e tem uma sintaxe de fácil aprendizagem e integração. A figura 2.2 mostra o módulo NodeMCU, o qual possui tamanho reduzido e facilita sua aplicação em locais onde não há muito espaço para alocar o circuito. Seu design facilita sua prototipação e além da conexão USB já integrada, que auxilia na programação do módulo. (OLIVEIRA, 2017)

FIGURA 2.2 – ESP8266 NodeMCU.

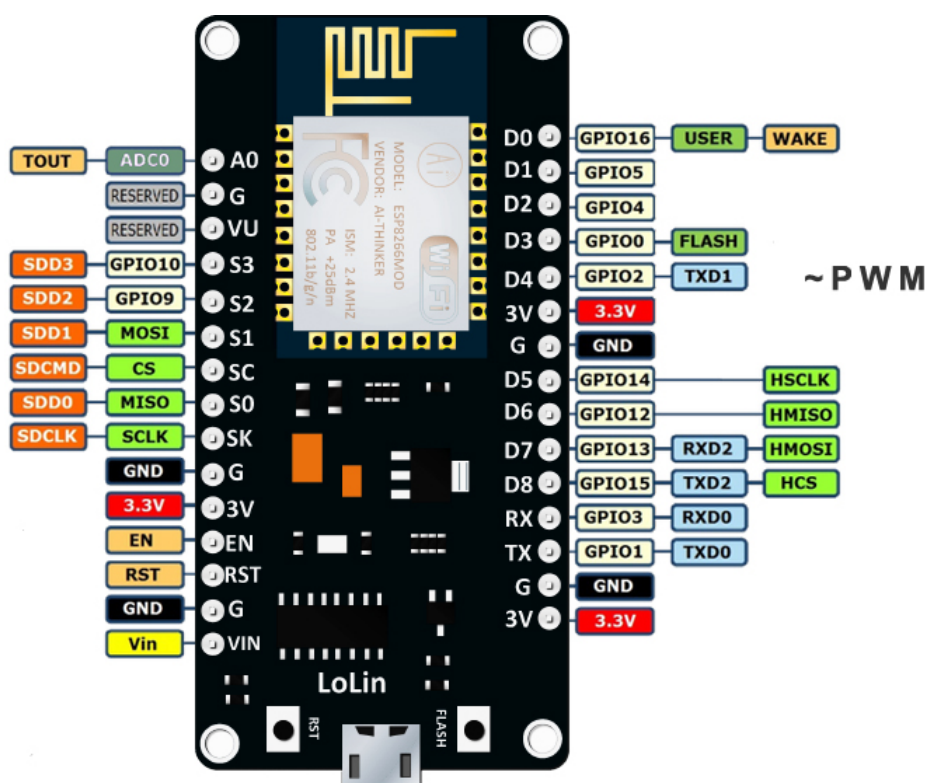


Fonte: AMAZON.COM (2019)

O modelo mais recente do ESP8266 NodeMCU é o V3, como já citado, sua interface fornece diversos tipos de entradas e saídas, analógicas, digitais e de pulso

modulado. A relação das funções e pinos relacionados está ilustrado na figura 2.3, material fundamental para a utilização desta plataforma de sistema embarcado, fornecendo a referência para a integração do hardware com os parâmetros de software que vão realizar a leitura e escrita dos sinais nas portas de entrada e saída do dispositivo.

FIGURA 2.3 – Relação de pinos para o ESP8266 NodeMCU V3



Fonte: TEACHMEMICRO.COM (2019)

2.1.1.2 PLATFORMIO

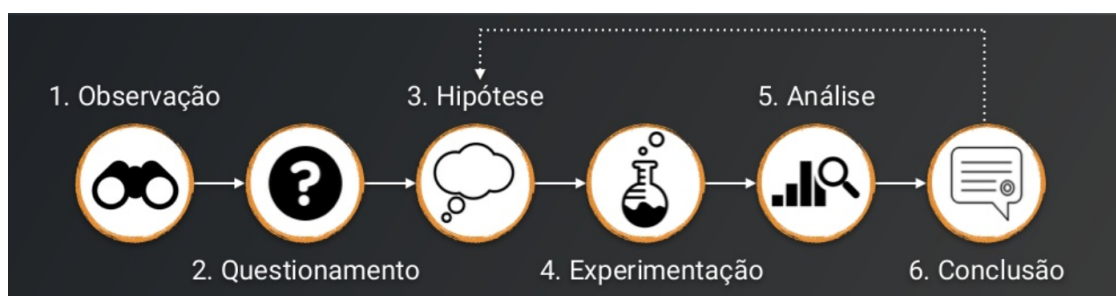
Para facilitar a programação de uma grande variedade de microcontroladores, foi criada a plataforma de programação de sistema embarcados *PlatformIO*, mantida através de desenvolvimento *Open Source* que permite a utilização da plataforma sem custos. Atualmente a plataforma fornece suporte a mais de 650 placas de desenvolvimento, compilando um código em C/C++ que pode ser executado nos circuitos, além de diversos exemplos e bibliotecas de implementação que podem ser utilizadas para facilitar a chamada de funções e ferramentas da placa.

2.2 INSTRUMENTAÇÃO ELETRÔNICA

A instrumentação pode ser definida como a medição de qualquer grandeza em qualquer processo, ou seja, a capacidade de mensurar grandezas físicas inerentes a atividade que se deseja analisar. A instrumentação está presente por exemplo, em uma instalação elétrica. na simples medida da tensão elétrica de uma residência (220 V ou 110 V), no controle do sistema que está gerando a tensão, por exemplo, na medição da velocidade da turbina de geração hidráulica ou na medição da temperatura dos reatores de energia térmica. (BALBINOT; BRUSAMARELLO, 2000)

A importância da Instrumentação poderia ser resumida em uma frase: "A medição é a base do processo experimental.", seja em um processo que deve ser controlado, seja em pesquisa ou em uma linha de produção em uma indústria, o processo da medição de grandezas físicas é fundamental e necessário. Antes de iniciar o procedimento de instrumentação o experimentalista deve antes conhecer o processo e suas incertezas, a fim de poder analisar corretamente os resultados provenientes da instrumentação do objeto a ser analisado, para isso pode-se utilizar o método genérico de análise científica ilustrado na figura 2.4. (BALBINOT; BRUSAMARELLO, 2000)

FIGURA 2.4 – Procedimento genérico de método científico



Fonte: BLOGDACIENCIA.COM (2019)

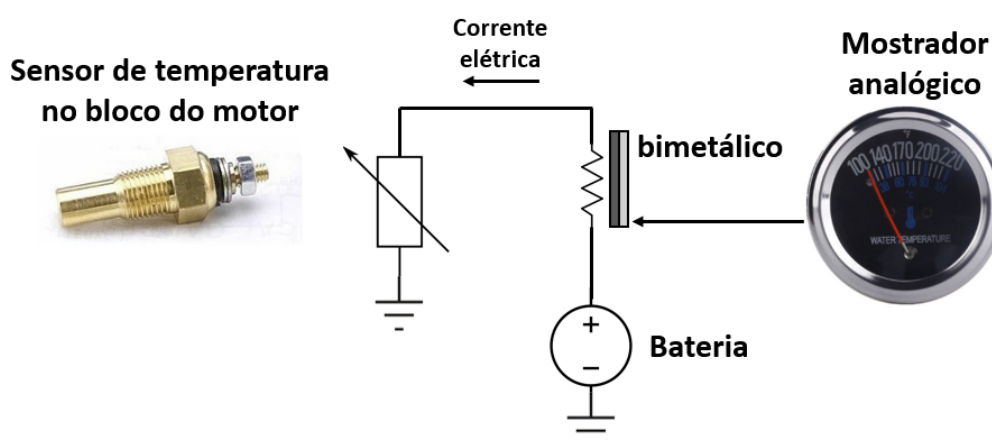
Portanto, a instrumentação é fundamental para que haja o controle de qualquer processo de maneira eletrônica, sendo ela a responsável por fornecer a alimentação de entrada de informações ao sistema. Com ela é possível criar sistemas de controle e automação industriais que recebem como entrada medições de grandezas físicas através de sensores e, a partir disso, determinar se os dados recebidos são condizentes com o esperado para o funcionamento do processo.

Para que haja a medição de uma determinada grandeza, são utilizados sensores e dispositivos de medição, este último responsável por interpretar a resposta do

sensor e transformá-la em escala de medida. Por exemplo, para a medição de sensores de temperatura resistivos, são utilizados dispositivos de medição capazes de converter a variação da resistência elétrica do sensor em informação da temperatura em graus Celsius, Fahrenheit ou Kelvin, sendo o processo de conversão feito de maneira digital ou analógica. Este tipo de medidor é muito comum em veículos de passeio, fornecendo a informação de temperatura do veículo ao condutor e até mesmo alertando o mesmo sobre situações de alta temperatura do motor.

Nos carros mais antigos, a conversão analógica do sinal gerado pelo sensor de temperatura do motor, era realizada através de um relógio analógico ligado diretamente ao sensor. O ponteiro do mostrador é ligado através de uma lamina bimetálica que varia seu tamanho de acordo com a temperatura gerada pela corrente que passa através da resistência elétrica do relógio, o qual é proporcional a variação da resistência elétrica do sensor, o circuito é ilustrado na figura 2.5.

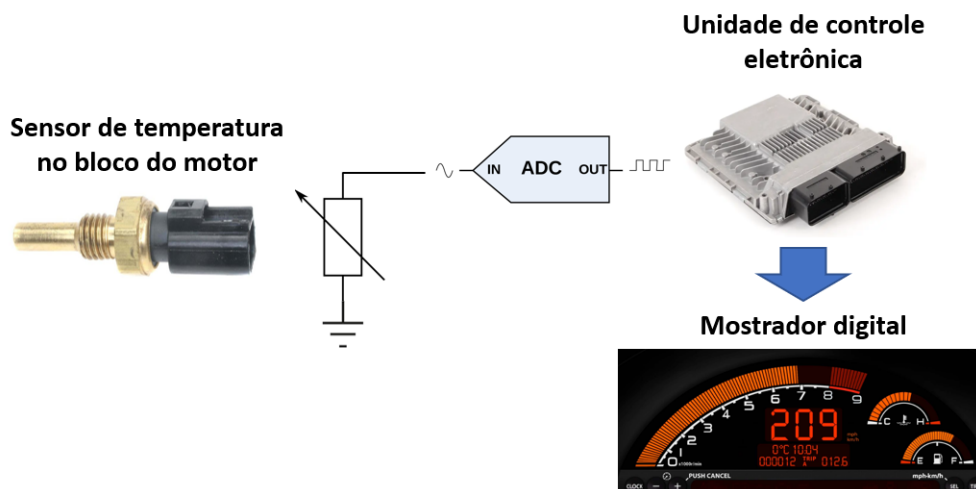
FIGURA 2.5 – Funcionamento de um medidor analógico de temperatura



Fonte: O AUTOR (2019)

Com a disponibilidade de circuitos de conversão analógico para digitais, o medidor de temperatura passou a ser mostrado digitalmente no painel do veículo. Agora o sinal de variação da resistência do sensor é medido pela unidade de controle e transformado em um valor digital, que pode ser transmitido ao painel digital do veículo, conforme a figura 2.6. Essa nova maneira de obtenção dos dados permite que o valor medido no sensor de temperatura seja agora disponibilizado para diversos processos ao mesmo tempo, tendo em vista que agora é dado como uma variável do sistema de controle e não mais um sinal elétrico diretamente ligado ao mostrador.

FIGURA 2.6 – Funcionamento de um medidor digital de temperatura



Fonte: O AUTOR (2019)

A forma de escolher a maneira correta para medir e mostrar uma determinada grandeza depende dos requisitos de quão preciso deve ser o valor lido e qual a resolução necessária na medição, por exemplo, para o controle de temperatura de um veículo não são importantes variações de temperatura na escala de duas casas decimais, por outro lado, para um sistema de controle de fermentação de bebidas este grau de resolução e precisão do valor pode se fazer necessário.

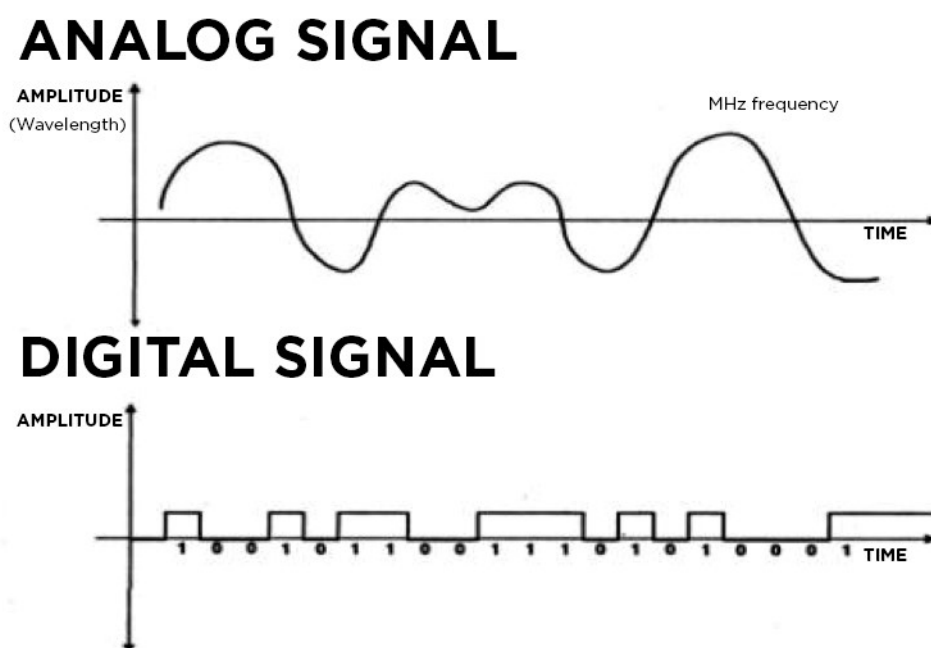
2.2.1 Sensores

Sensor é por definição um dispositivo sensível a alguma forma de energia do ambiente que pode ser luminosa, cinética ou relacionando informações sobre uma grandeza que precisa ser medida como temperatura, pressão, velocidade, corrente elétrica ou aceleração. (THOMAZINI; ALBUQUERQUE, 2005) Eles estão cada vez mais presentes nas atividades do dia a dia, através dos dispositivos como celulares e relógios inteligentes, que estão constantemente utilizando as medições de como temperatura, altitude, inclinação, luminosidade e velocidade para determinar seu modo de funcionamento ou interagir de maneira mais fiel com o usuário.

O sinal de saída fornecido pelo sensor pode ser de dois tipos, o primeiro analógico, onde o valor de saída pode assumir qualquer valor dentro de sua faixa de operação ao longo do tempo, usualmente esse tipo de saída é utilizado para medir grandezas que também se comportam podendo assumir qualquer valor dentro

do tempo de medição, algumas delas sendo luminosidade, pressão, temperatura, vazão, umidade e velocidade. Já os sensores digitais podem assumir dois valores ao longo do tempo, sendo interpretados como zero ou um, esses tipos são comumente utilizados para determinar eventos transitório, como encoders e passagens de objetos. A diferença entre essas duas saídas pode ser observada na figura 2.7. (THOMAZINI; ALBUQUERQUE, 2005)

FIGURA 2.7 – Sinais analógicos e digitais no tempo



Fonte: KSIXMOBILE.COM (2019)

Sensores podem também serem encontrados na forma de transdutores, que é o nome do dispositivo completo que é composto pelo sensor e por um circuito eletrônico responsável por converter a excitação gerada no sensor pela grandeza física em sinais elétricos.

Cada sensor possui características elétricas diferentes que definem como será a resposta em relação aos estímulos externos. Estes requisitos devem ser cuidadosamente analisados para decidir qual a melhor opção de sensor a se utilizar, tendo em vista que refletem diretamente na qualidade e preço do produto. São eles:

- Tipo de saída: digital ou analógico, conforme supracitado;
- Sensibilidade: relação entre a variação da grandeza física na entrada e o sinal elétrico gerado na saída;

- Exatidão: relação entre o sinal medido e o erro induzido pelo transdutor na medida real;
- Precisão: grau de repetibilidade do valor medido por um sensor ao longo do tempo;
- Linearidade: forma da curva do sinal de saída do sensor, sendo uma saída linear representada por uma reta ou comportamento não lineares ao longo da escala de medida;
- Alcance (range): faixa de valores de entrada e saída que o transdutor foi projetado para operar;
- Estabilidade: relaciona a flutuação do sinal de saída do sensor;
- Velocidade de resposta: tempo decorrido para que as variações aplicadas na entrada do sensor são refletidas no sinal de saída;
- Preço: uma composição de todos os requisitos anteriores, onde quanto mais rigorosos os requisitos, maior o preço para a fabricação do sensor ou transdutor.

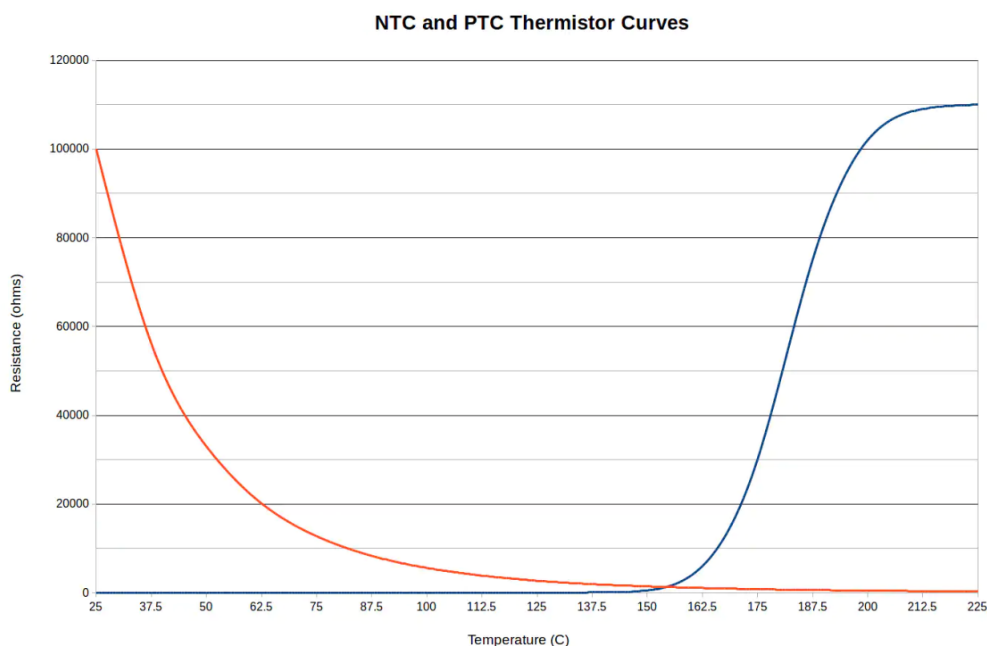
2.2.1.1 Sensores de temperatura

Amplamente utilizados na indústria, veículos e eletrodomésticos, os sensores de temperatura elétricos podem ser realizados utilizando técnicas diferentes, como termistores, termopares, termoresistores e pares metálicos. O uso de cada tipo de sensor depende diretamente dos requisitos do sinal de temperatura a ser medido, a grande maioria funciona com a variação de sua resistência elétrica de acordo com a variação da temperatura, sendo bem sensíveis a pequenas variações.

Um exemplo da curva formada por este tipo de sensor são os termistores, que estão ilustrados na figura 2.8, a qual representa um sensor do tipo NTC (*Negative Temperature Coefficient*), que varia sua resistência de forma inversa ao aumento da temperatura e um sensor do tipo PTC (*Positive Temperature Coefficient*), que varia sua resistência de maneira direta a variação da temperatura. Este tipo de sensor é construído utilizando elementos resistivos como manganês, níquel, cobalto e cobre, revestido por cerâmica ou polímeros. Um ponto importante a atentar com relação ao uso deste tipo de sensor é o aquecimento gerado pela própria potência dissipada pelo componente, que tende a esquentar com o aumento da corrente conduzida pela queda da resistência. Termistores não possuem suas curvas de resposta padronizadas

entre os fabricantes, o que pode dificultar a sua implementação, principalmente em casos onde a troca das pontas de medição são frequentes pelos diferentes níveis de temperatura máxima de operação. (THOMAZINI; ALBUQUERQUE, 2005)

FIGURA 2.8 – Comparação da variação de resistência de termistores NTC e PTC



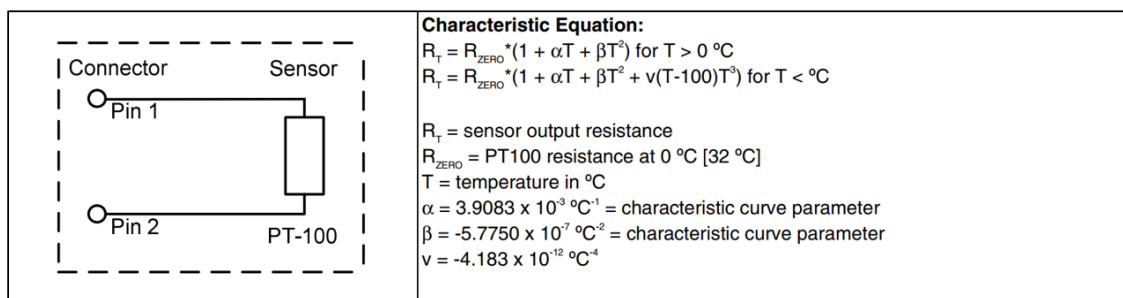
Fonte: MOUSER ELECTRONICS, INC. (2019)

Uma alternativa bem próxima aos termistores são as termorresistências, que tem o funcionamento bem parecido, porém com valores padronizados na indústria o que facilita sua utilização. Amplamente utilizado na indústria automotiva para a medição das temperaturas de fluidos do motor, as termorresistências tem diversas opções de valores de alcance e ambientes de aplicação. Seu valor é padronizado através da resistência nominal a 0° C, um exemplo que é amplamente utilizado na indústria por oferecer grande estabilidade e precisão são as termorresistências Pt-100, feitas a partir de platina e com resistência nominal de 100 Ohm a 0° C. Uma outra grande vantagem na utilização destes sensores é que eles permitem a medição da temperatura a grandes distancias entre o sensor e o medidor, porém seus preços costumam ser mais elevados do que os tipos termistores. As desvantagens ficam novamente com o auto aquecimento gerado pela sua utilização e pelo atraso na medição até que todo o corpo do sensor (bulbo metálico externo) seja aquecido. (THOMAZINI; ALBUQUERQUE, 2005)

Um exemplo deles são os da série R300 da empresa *Honeywell*, que foram projetos para aplicação em veículos na medição de gases do escape ou dentro do

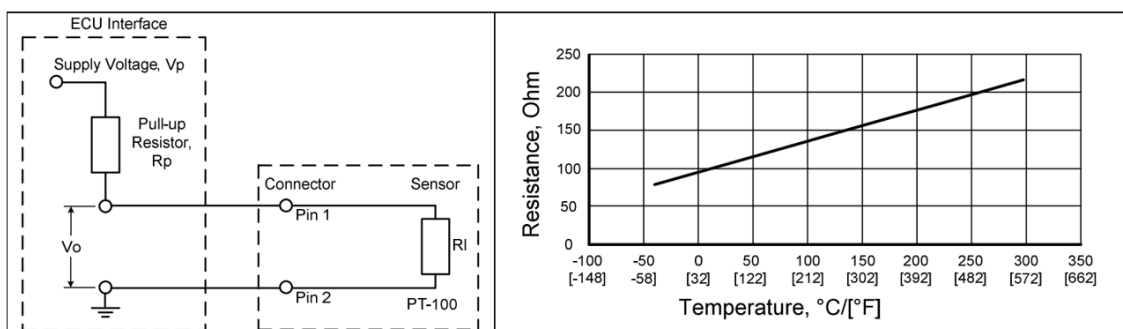
cilindro, sua temperatura de operação varia de -40°C a 275°C, e resistência nominal a 0°C de 100 Ohm. A fórmula de funcionamento elétrico do sensor e sua curva resultante está ilustrado nas figuras 2.9 e 2.10. (HONEYWELL, 2008)

FIGURA 2.9 – Equação e circuito elétrico que representa o comportamento do sensor R300



Fonte: (HONEYWELL, 2008)

FIGURA 2.10 – Circuito exemplo e curva do sensor R300

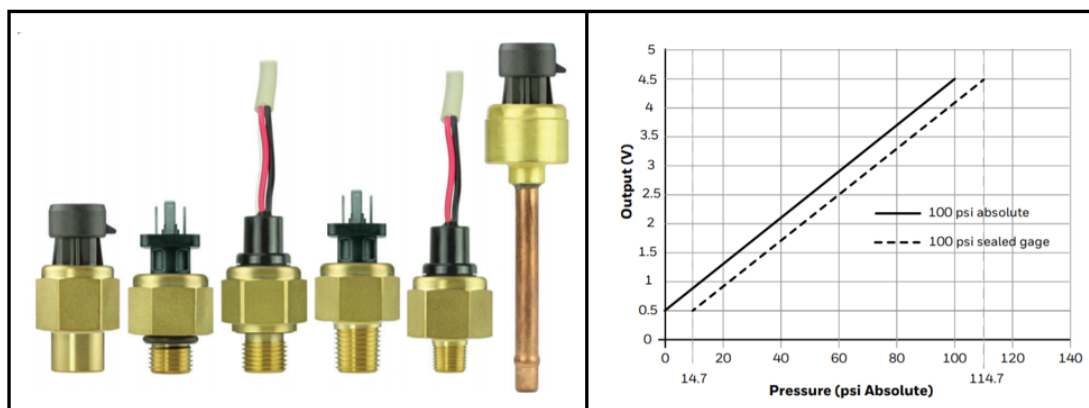


Fonte: (HONEYWELL, 2008)

2.2.1.2 Sensores de pressão automotivos

Sensores de pressão são utilizados na indústria automotiva para monitorar a pressão do ar, óleo e combustível, fornecendo o feedback necessário para o controle do motor e evitar danos ao conjunto. Um dos sensores utilizados é do tipo capacitivo, onde um dos lados do capacitor tem sua distância variada conforme o aumento da pressão exercida na entrada de pressão do sensor. Essa variação da capacitância do sensor, gera um sinal de saída de tensão variável diretamente proporcional a variação da pressão. Sua ligação é feita através de três cabos, um para alimentação do sensor, outro para a referência do circuito e o terceiro fornecendo o sinal de tensão de saída do sensor. Alguns exemplos de sensores automotivos de pressão, e sua curva de resposta, podem ser vistos na figura 2.11.

FIGURA 2.11 – Sensores de pressão automotivos capacitivos e curva de resposta

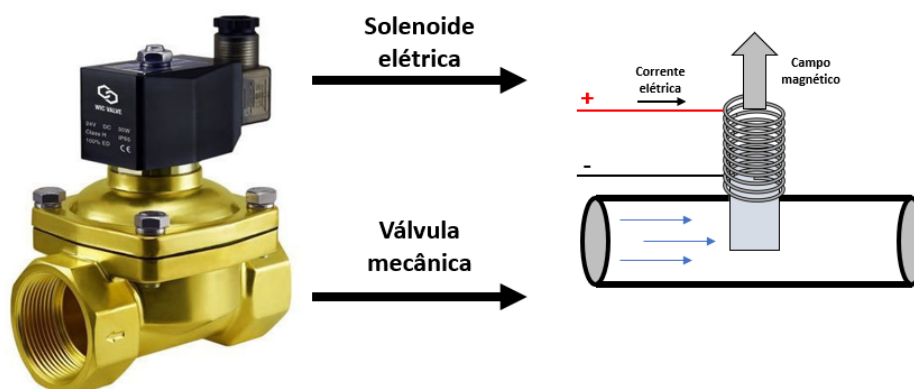


Fonte: (HONEYWELL, 2008)

2.3 VÁLVULAS SOLENOIDES

Para controlar eletronicamente o fluxo de fluidos ou gases em sistemas são utilizadas válvulas solenoides que permitem o controle através do acionamento de uma bobina elétrica que move um embolo e libera a passagem do fluxo. Seu funcionamento se dá através do campo magnético criado pela energização da bobina, que move o embolo na direção do campo magnético até que a corrente elétrica seja novamente reduzida, e uma mola interna pressiona novamente o embolo para sua posição inicial. O funcionamento e componentes que compõem uma válvula solenoide de corrente contínua está ilustrado na figura 2.12.

FIGURA 2.12 – Componentes da válvula solenoide

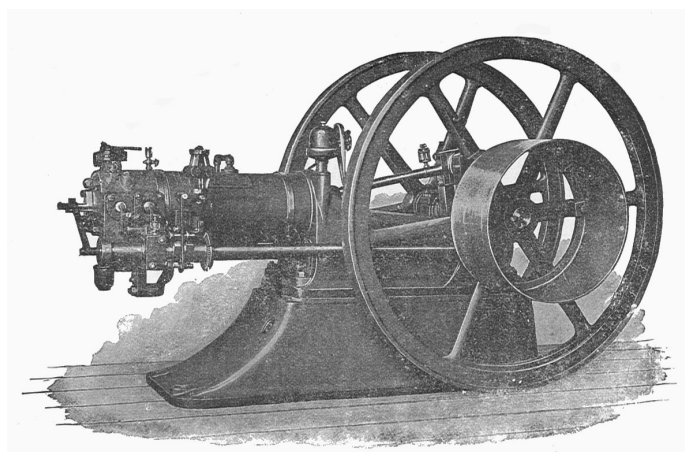


Fonte: O AUTOR (2019)

2.4 MOTORES A COMBUSTÃO INTERNA

Motores são por definição máquinas destinadas a converter qualquer forma de energia (térmica, elétrica, hidráulica, etc.) em energia mecânica. No caso dos motores de combustão interna, há transformação de energia térmica (queima de combustível) em energia mecânica. (MAHLE, 2016) Seu primeiro modelo, ilustrado na figura 2.13, foi concebido por Nikolaus August Otto, sendo o primeiro motor a combustão interna 4 tempos. Ele foi apresentado durante uma exposição em Paris em 1876, seu funcionamento ficou conhecido como ciclo Otto. (SACK, 2019)

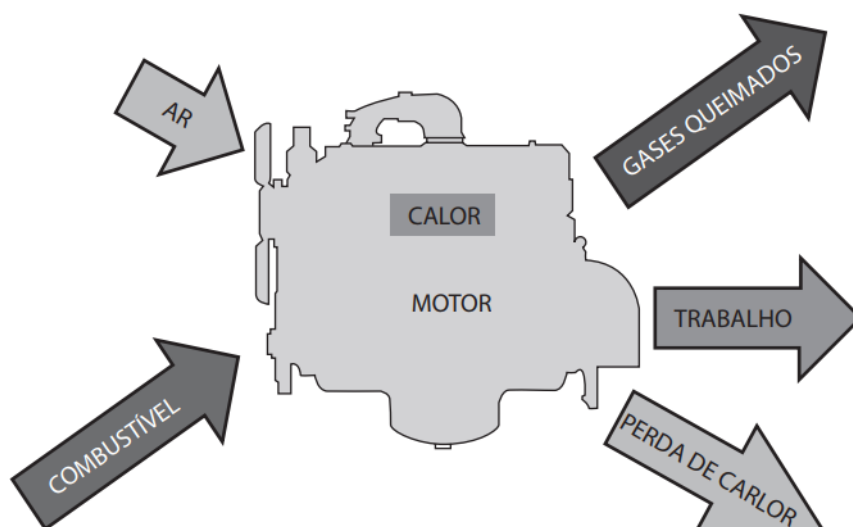
FIGURA 2.13 – Primeiro motor a combustão 4 tempos criado por Otto



Fonte: (HAWKINS, 1903)

Os ciclos de conversão térmica, tal qual o motor criado por Otto, são realizados utilizando o que é chamado de fluido ativo, o qual é utilizado na combustão interna para transformação em energia térmica, como por exemplo a gasolina, etanol ou óleo diesel. Cada fluido possui diferentes modos de combustão, ocorrendo por centelhamento ou por compressão, o que resulta em diferentes respostas térmicas do sistema. Além do fluido ativo, é necessário que o sistema seja alimentado com ar, para que haja a combustão. O resultado deste processo são os gases queimados, trabalho que é transferido para o eixo girante e a perda de calor com o aquecimento do sistema como um todo, o fluxo de substâncias e grandezas envolvidas está ilustrado na figura 2.14. (BRUNETTI, 2012)

FIGURA 2.14 – Fluxo de funcionamento de motores



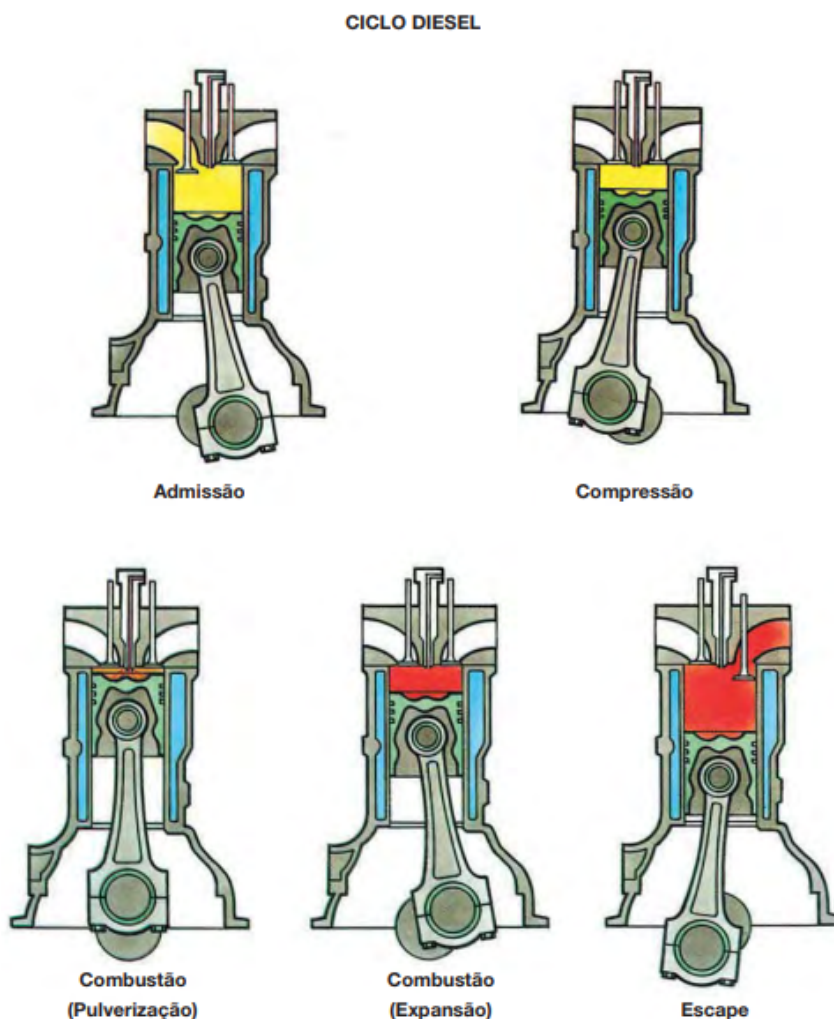
Fonte: (BRUNETTI, 2012)

Diversos tipos de veículos automotores têm como propulsores os motores a combustão interna, projetados para diferentes tipos de aplicação como veículos de passeio, transporte de pessoal e transporte de cargas, o que influencia diretamente na quantidade de potência necessária e, conseqüentemente, na quantidade de calor gerado pelo sistema.

2.4.1 Motores movidos a combustão Diesel

Uma das possíveis opções de gerar energia através da combustão é com a utilização do óleo diesel, substrato do petróleo, e para que haja a combustão é utilizado o processo de compressão do ar presente no cilindro do motor até que a temperatura se eleve ao ponto de comburir o óleo diesel injetado na câmara. Este processo leva a liberação de energia que impulsiona o pistão para baixo e essa energia é transferida para um eixo chamado de virabrequim, o processo de combustão é ilustrado na figura 2.15, onde primeiro há a admissão do ar para a câmara de combustão, o ar é comprimido pela força produzida por outro cilindro, o combustível é pulverizado na câmara logo antes de atingir o pico de temperatura onde ocorre a explosão e por fim o pistão é empurrado para baixo com a força da expansão dos gases, e este processo se repete em quanto houver ar e combustível sendo fornecido ao sistema. (MAHLE, 2016)

FIGURA 2.15 – Ciclo de combustão em um motor a diesel



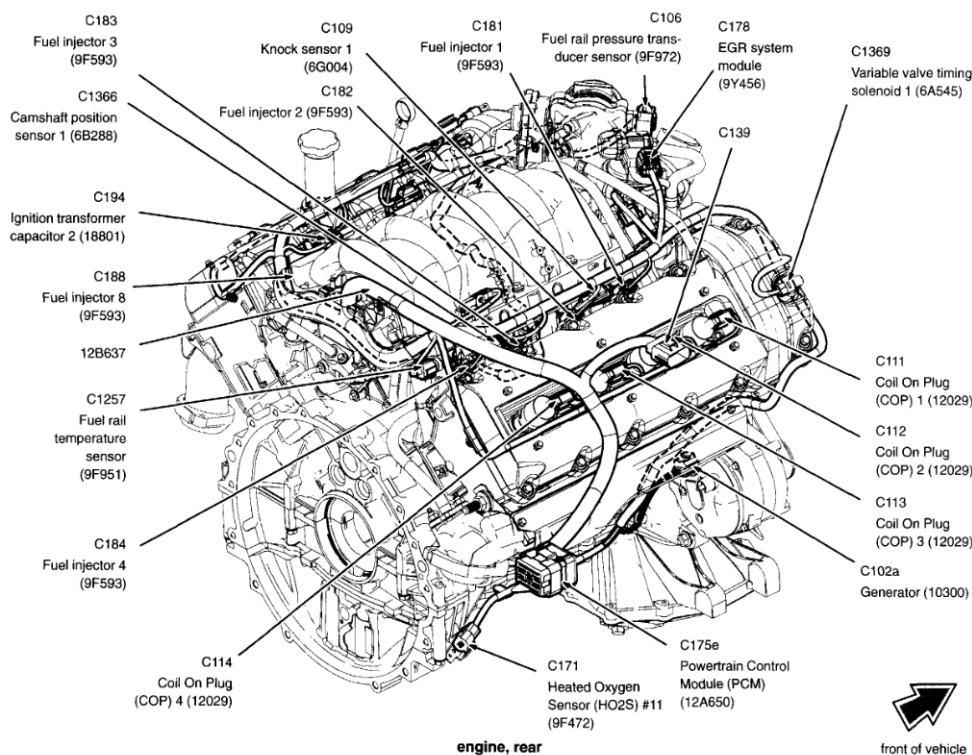
Fonte: (MAHLE, 2016)

2.4.2 Sistemas eletrônicos automotivos

Com a necessidade de obter cada vez mais eficiência, menor consumo de combustível com menor emissão de poluentes, os motores vem recebendo uma grande carga de componentes eletrônicos para realizar o seu controle de maneira mais detalhada. São utilizados diversos sensores eletrônicos para medir o comportamento do motor como, sensores de pressão, sensores de velocidade, sensores de temperatura e sensores de fluxo. A partir a leitura destes sensores é realizado o processamento dos dados por *Electronic Control Units* (ECU), dentro delas a mais popular é a unidade de injeção eletrônica, que permitiu o controle eletrônico da quantidade de combustível e ar injetada na câmara de combustão. A ECU realiza o controle do motor através de atuadores eletrônicos, como válvulas solenoides e reles acionadores. Uma representa-

ção da quantidade de sensores e atuadores presentes nos motores está na figura 2.16. (BRUNETTI, 2012)

FIGURA 2.16 – Sensores e atuadores utilizados no controle de motores a combustão interna



Fonte: ENGINE-SENSORS.COM (2019)

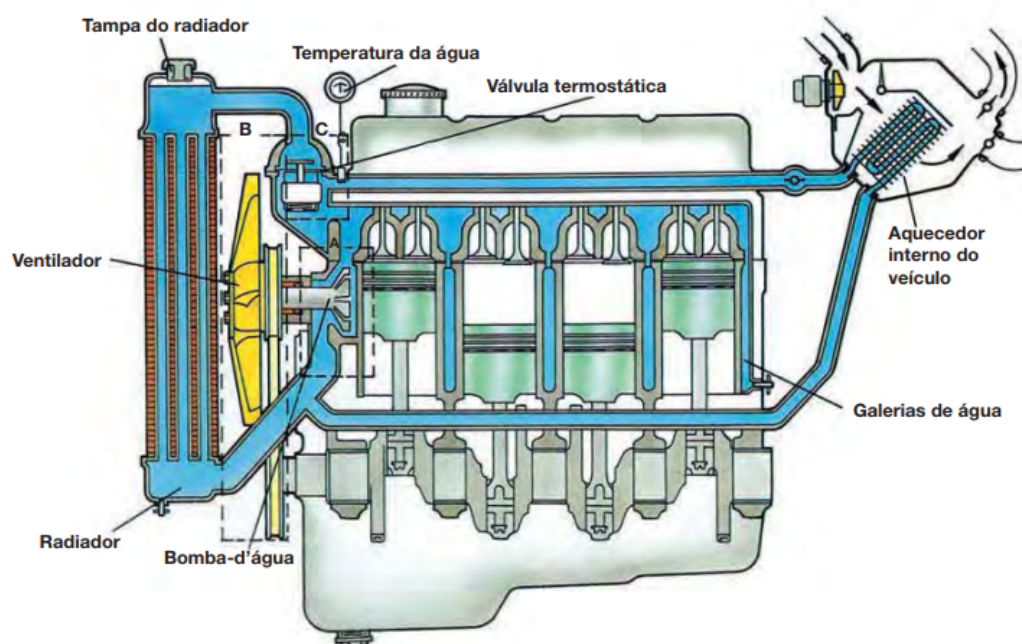
Além de auxiliar na eficiência dos motores, outra função fundamental realizada pelos sensores é de providenciar segurança na operação dos motores, alertando a ECU sobre condições adversas de funcionamento, detectando aumentos de temperatura e pressão, podendo evitar que o motor sofra algum dano pela operação incorreta. Sensores de temperatura do sistema de arrefecimento podem por exemplo alertar a ECU sobre a alta temperatura do sistema a qual pode limitar a potência gerada pelo motor e impedir que esse venha a fundir pelo excesso de calor gerado na combustão, alertando o motorista de que o veículo deve ser conduzido a uma oficina especializada.

2.4.3 Sistema de lubrificação e arrefecimento

Os motores a diesel são, em sua maioria, aplicados a veículos pesados como vans e caminhões, já que este motor fornece uma grande quantidade de torque para o eixo. A sua utilização para estas aplicações demanda que o motor forneça muita energia, que por consequência gera uma grande quantidade de calor, que deve ser dissipada

para não causar sobreaquecimento dos componentes. O sistema de arrefecimento é muito importante porque ele tem por finalidade absorver o excesso de calor gerado durante o funcionamento do motor, mantendo a temperatura dele dentro dos limites especificados em seu projeto. Há dois sistemas básicos: arrefecimento a ar ou líquido, este último ilustrado na figura 2.17. (MAHLE, 2016)

FIGURA 2.17 – Sistema de arrefecimento por líquido



Fonte: (MAHLE, 2016)

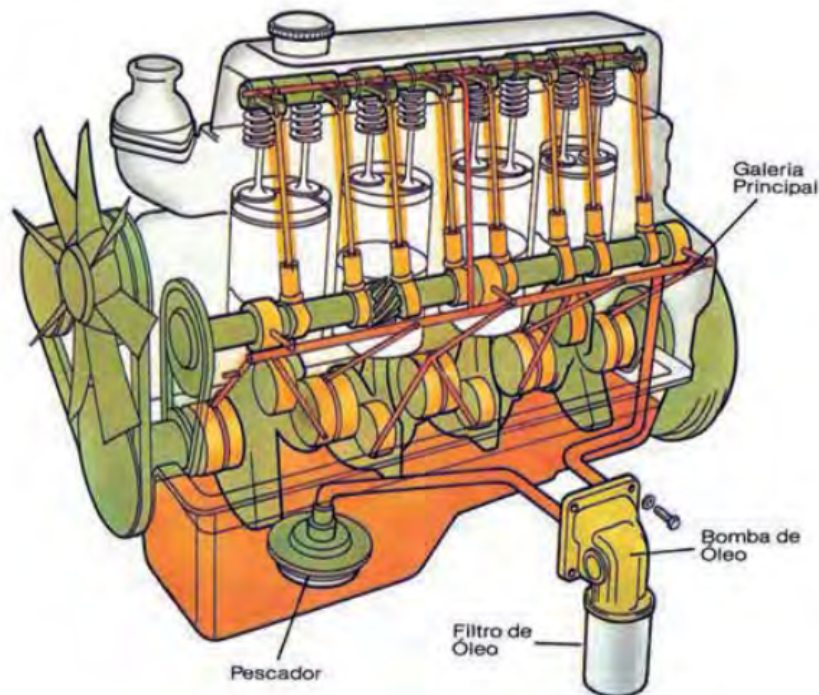
O calor absorvido pelo líquido de arrefecimento que circula pelo motor é trocado com o ambiente através do radiador, o qual possui ventilação forçada por um ventilador. Este fluxo se repete durante todo o funcionamento do motor para garantir que a temperatura não suba mais do que os limites térmicos dos componentes.

Porem há uma limitação do arrefecimento por líquidos nas partes móveis do motor, e, portanto, estas peças devem ser arrefecidas pelo sistema de lubrificação. O sistema de lubrificação de um motor é composto por diversos componentes que fazem circular o óleo no sistema, controlam a pressão do mesmo e fazem a sua filtragem, de maneira que haja uma lubrificação adequada em todas as áreas de atrito sob as diversas condições de operação. (MAHLE, 2016)

Em motores a diesel esse sistema possui uma circulação forçada sob pressão, como ilustrado na figura 2.18. O fluxo é controlado por válvulas de pressão mecânicas

e solenoides elétricas que determinam a direção do fluxo de óleo e, por consequência, quais sistemas serão arrefecidos. O mal funcionamento desse sistema vai acarretar sobreaquecimento dos componentes, chegando ao limite de fundir as peças metálicas do motor e causar um dano irreversível. (BRUNETTI, 2012)

FIGURA 2.18 – Sistema de lubrificação sob pressão

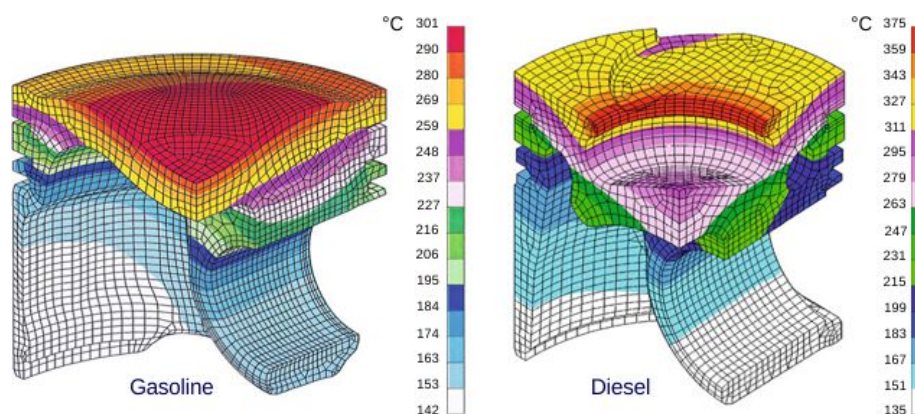


Fonte: (MAHLE, 2016)

2.4.4 Sistema de *Piston Cooling Jet*

Em torno de 6 a 8% da temperatura gerada pela combustão é transferida para o pistão durante a explosão, o que gera um aquecimento acelerado do componente. A figura 2.19 mostra os níveis de temperatura atingidos por pistões de motores a combustão durante a operação.

FIGURA 2.19 – Temperatura do pistão em motores a combustão



Fonte: (DIESELNET.COM, 2017)

Para auxiliar no resfriamento dos pistões do motor é utilizado o sistema chamado de *Piston Cooling Jet* (PCJ), o qual é composto por um injetor posicionado logo abaixo do pistão que libera um jato de óleo direcionado para a parte inferior do cilindro. Esse sistema permite que maiores relações de compressão e combustão sejam atingidas e por consequência maior potência utilizando o mesmo motor. O sistema é controlado através de válvulas elétricas e mecânicas que garantem que o fluxo de óleo normal do motor seja desviado para o sistema de PCJ, este controle é fundamental para que não haja grande acréscimo no consumo de combustível ao manter o sistema todo tempo acionado, ou que haja resfriamento excessivo do pistão, o que pode também comprometer a reação de combustão interna.

3 DESENVOLVIMENTO

Tendo a fundamentação teórica sido consolidada, neste capítulo o foco será a descrição do problema e todos os passos realizados para a solução, apresentando as etapas, especificações e verificações do sistema proposto. Para entender como foi concebido o objetivo deste trabalho, devemos antes compreender o problema que originou a necessidade de se construir uma solução eletrônica e automática para verificação do funcionamento do sistema de lubrificação de um motor de veículo pesado.

3.1 ANÁLISE DO PROBLEMA A SER SOLUCIONADO

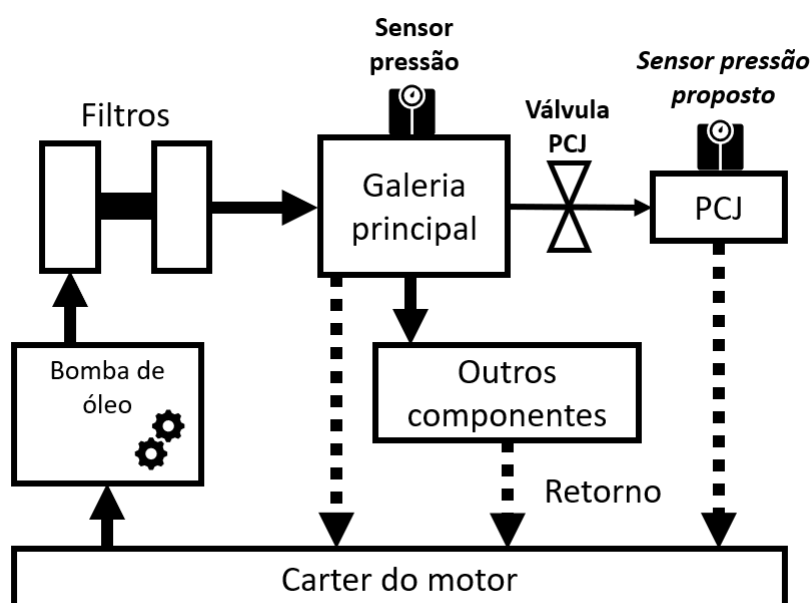
Um grande erro cometido no desenvolvimento de produtos é não analisar corretamente o problema a ser solucionado, sem claras definições de requisitos mínimos e objetivos a serem atingidos. Portanto, a primeira etapa do trabalho é realizar a análise do problema que será resolvido e, somente após identificado os autores da falha, trabalhar em uma possível solução. O problema que o trabalho se propõem a resolver foi derivado de problemas de campo e na fábrica de motores que estão sofrendo com alta temperatura e consequentemente a fusão dos componentes metálicos, o que causa a inutilização do sistema e com um custo de reparo altíssimo, em torno de R\$ 200.000,00 por veículo falhado.

Com isso, a primeira análise é dos fatores que poderiam estar levando a este sobreaquecimento anormal do sistema em alguns veículos, de aproximadamente 2% da frota. Após estudos das possíveis causas de falha de componentes do sistema de arrefecimento do motor, foi constatado que algumas válvulas mecânicas e elétricas do sistema de Piston Cooling Jet estavam danificadas e não liberavam o fluxo de óleo para o dispositivo de arrefecimento dos pistões. Definidos os autores da falha, podemos definir os componentes que devem ser utilizados para elaborar uma solução de verificação das válvulas de mecânicas e elétricas do sistema de maneira automatizada.

3.2 CIRCUITO DE ÓLEO DO MOTOR

Para determinar quais válvulas estão com defeito de funcionamento devemos primeiro analisar o comportamento esperado do sistema e o posicionamento das válvulas e sensores. O sistema é composto por diversas galerias de óleo que distribuem o fluido pelo motor, para isolar o sistema de PCJ, podemos analisar somente a galeria principal de óleo e as galerias do PCJ, indicadas na figura 3.1.

FIGURA 3.1 – Sistema de óleo do motor.



Fonte: O AUTOR (2019)

Podemos verificar que o sistema atual possui somente um sensor de pressão na galeria principal, o qual consegue proteger o motor em caso de falta ou excesso de óleo, que causa um aumento geral na pressão do sistema. Porém, o sistema que atua o PCJ possui somente uma válvula, que quando eletricamente acionada pela controladora do motor, não possui qualquer sinal de feedback indicando que o sistema de óleo realmente foi acionado e que as solenoides e reguladores de pressão mecânicos estão funcionando como deveriam.

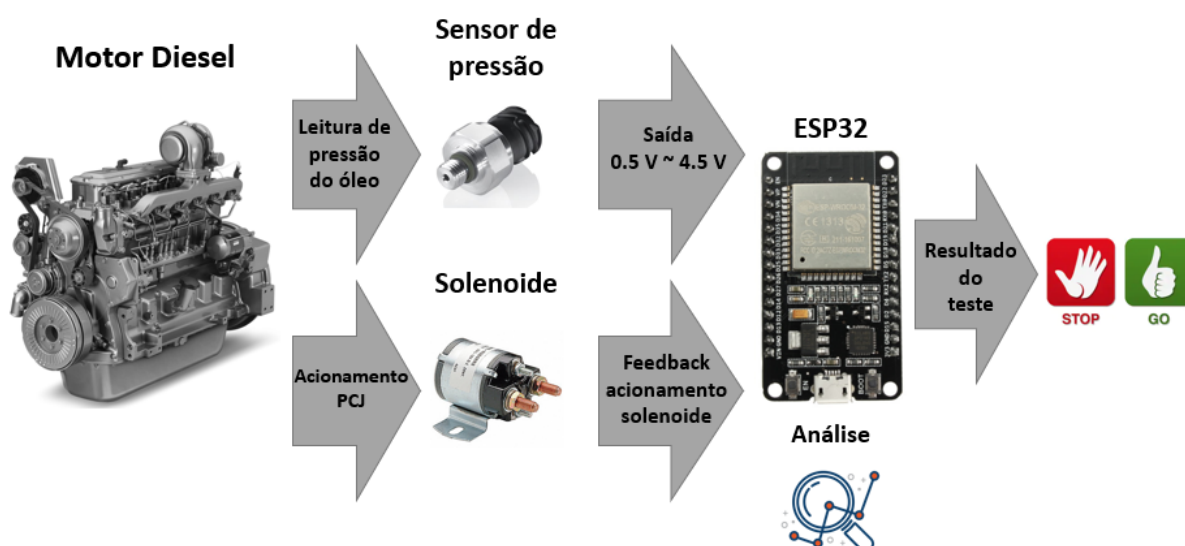
Juntando a análise da causa raiz, que é o não funcionamento correto dos reguladores de pressão e da válvula solenoide, com a falta de um sinal de feedback para o motor de que o sistema está com defeito temos como resultado a falha reportada em campo. Logo a solução para que o sistema possa ser validado quanto a seu funcionamento é a utilização de um sensor de pressão adicional na galeria do sistema

de PCJ, como indicado na figura 3.1, que forneça um sinal de feedback do acionamento do sistema.

3.2.1 Diagrama funcional do sistema

Entendendo como funciona o sistema de refrigeração e todo o caminho percorrido pelo óleo que é injetado nos pistões, podemos definir o funcionamento geral do sistema proposto. Na figura 3.2 podemos ver como devem ser as entradas e saídas e quais componentes serão utilizados no desenvolvimento da solução proposta.

FIGURA 3.2 – Fluxo de funcionamento geral do sistema.



Fonte: O AUTOR (2019)

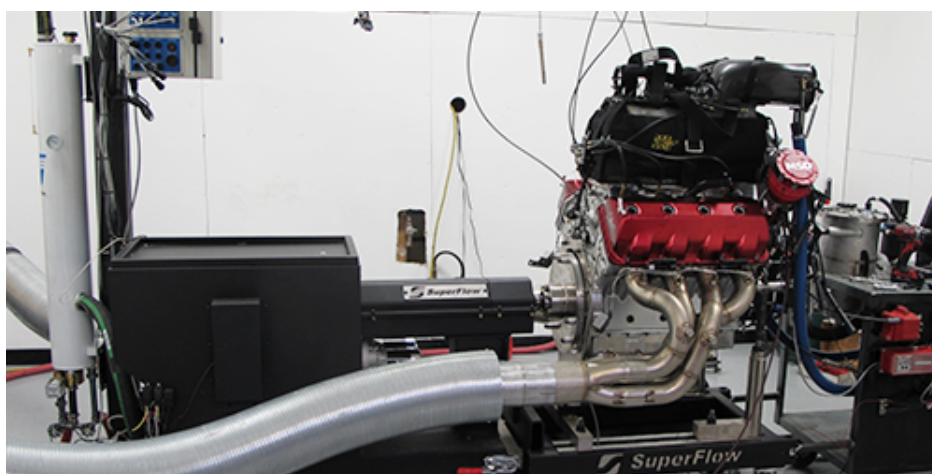
Em linhas gerais, o sistema irá realizar a leitura de pressão através de um sensor de pressão localizado na carcaça do filtro de óleo, onde será possível extrair a pressão atual do sistema de PCJ, e cruzar essas leituras de pressão com os comandos de acionamento da válvula solenoide pela unidade de controle do motor. Todas estas leituras serão obtidas durante um teste de funcionalidade do motor executado ao final da linha de montagem, realizando ciclos padrão de teste e com comportamento conhecido. Com isso, será possível determinar durante a realização do teste se os componentes elétricos e mecânicos do sistema estão funcionando corretamente, evitando que motores com peças defeituosas sejam liberados e montados no veículo.

Além dos sinais descritos, serão utilizados os sinais: indicação de posição

da chave, ligada nos níveis lógicos 0V ou 24V que indica a condição da chave do veículo na posição "LIGADA" e sinal indicador da solenoide de acionamento do motor de partida, que fornece um feedback ao sistema de quando o motor de partida foi acionado e, por consequência, o teste foi iniciado. Com isso o sistema consegue detectar de maneira automática quando ele foi ligado ao veículo e quando se inicia o teste, sem que o operador do procedimento de teste tenha que realizar nenhuma tarefa adicional. Para garantir que o sistema esteja funcionando antes do início do teste teremos duas condições, a primeira de que o ESP32 esteja realmente ligado a alimentação (proveniente do mesmo chicote elétrico que liga o motor) e a segunda de que há uma tensão mínima sendo detectada no sensor de pressão, indicando que o operador não esqueceu de conectar os cabos do dispositivo de teste antes do início do mesmo.

As verificações e sinais citados acima são requisitos de extrema importância ao projeto, tendo em vista que o custo hora do funcionamento destes laboratórios que realizam os testes são de custo elevado, logo há de se garantir que o sistema não inicie o teste sem que todos os componentes estejam conectados e comunicando de maneira correta. Um exemplo de como são estes locais de teste pode ser visto na figura 3.3, onde o motor é instalado após a produção e testado suas capacidades de produção de torque e rotação conforme o projetado.

FIGURA 3.3 – Procedimento de teste de motores.

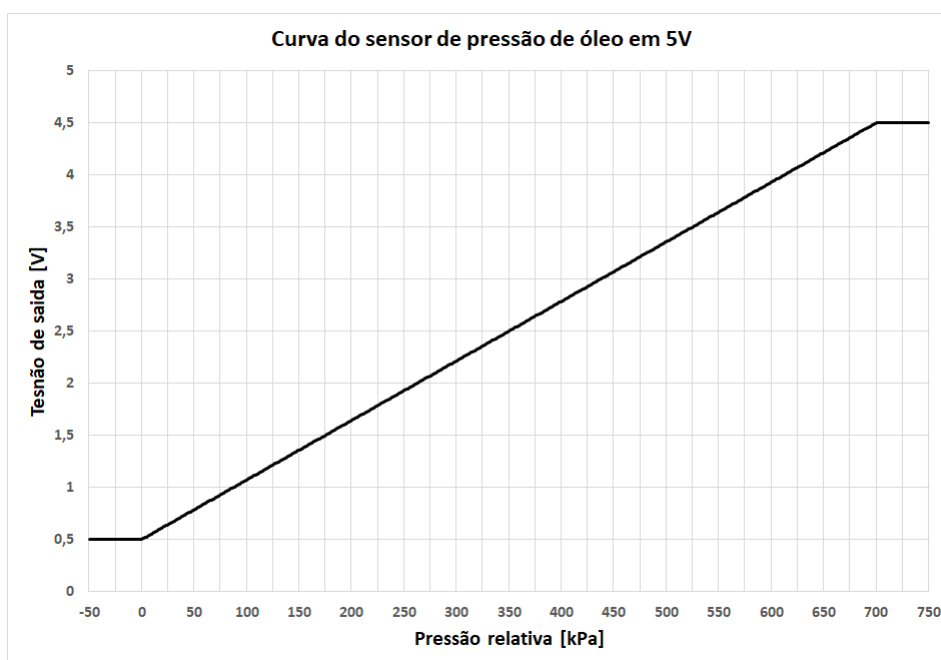


Fonte: SUPERFLOW.COM (2019)

3.3 ANÁLISE DA RESPOSTA DOS SENSORES E ATUADORES

Para que seja possível desenvolver o software de leitura do sinal de pressão da galeria PCJ, devemos antes compreender o comportamento esperado do sinal ao longo do ciclo de testes que serão realizados. Para isso, um motor foi instrumentado com um sensor de pressão na galeria PCJ e cabos conectados ao sensor da galeria principal, galeria PCJ e solenoide ligados a um osciloscópio. Os pontos mais importantes que deveriam ser analisados são: a resposta em tensão do sensor quando a solenoide está desligada ou ligada, a pressão máxima medida na galeria PCJ e sua relação com a pressão da galeria principal, tempo de resposta do sensor PCJ quando ocorre o acionamento da solenoide e, por último, a presença de sinais ruidosos que podem estar incluídos ao sinal enviado pelo sensor. A curva de resposta em tensão pela variação de pressão foi retirada do manual do componente e está ilustrada na figura 3.4.

FIGURA 3.4 – Curva de resposta do sensor.



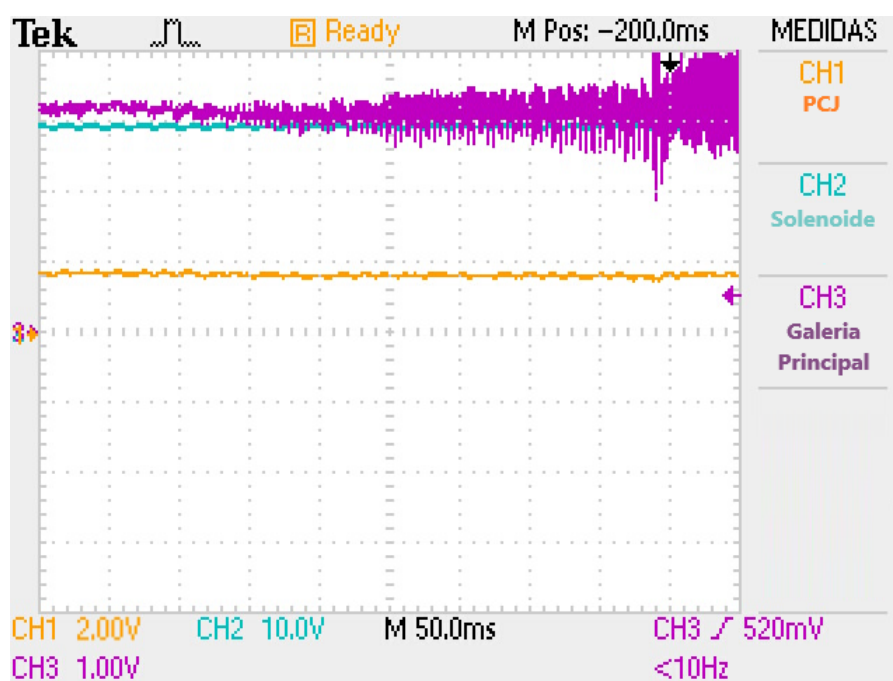
Fonte: O AUTOR (2019)

Observa-se que antes de 0 kPa e depois de 700kPa o sensor retorna um nível de tensão constante, isso é para que a controladora do motor consiga identificar que o sensor está conectado e que não é um curto circuito ou circuito aberto no ramo de ligação deste sensor.

3.3.1 Primeiro teste de motor instrumentado

O teste realizado no motor foi elaborado com base nas condições de funcionamento da válvula PCJ, com objetivo de obter diversos sinais dos sensores e atuadores, aumentando a chance de detectar a falha de operação do sistema. A primeira parte do teste foi analisar a resposta dos sensores e atuadores durante a partida do motor, que pode ser visto na figura 3.5.

FIGURA 3.5 – Resposta dos sensores na partida do motor.



Fonte: O AUTOR (2019)

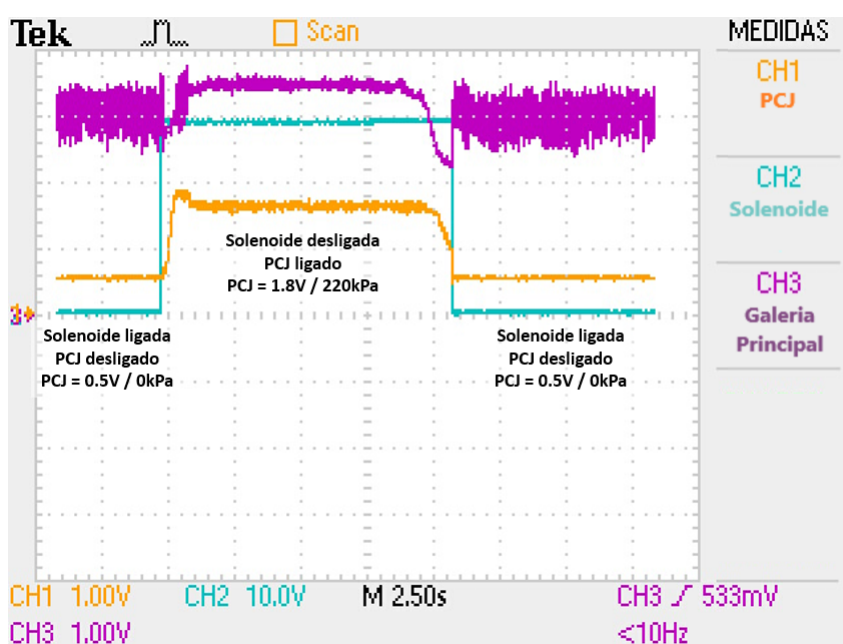
O teste na partida mostrou que os sinais de pressão e da válvula demoram cerca de 2 segundos para se estabilizar e que há uma grande quantidade de ruído vinda do sensor da galeria principal, com amplitude de até 1V quando a pressão da galeria se encontra em torno de 3.1V ou 450 kPa (observados a partir da escala de 1V por nível no osciloscópio). Já o sinal do PCJ estava em torno de 1.5V (escala de 2V por nível) ou 175kPa e possui uma componente de ruído de baixa amplitude e comportamento senoidal, com aproximadamente 0.1 V de amplitude e 50 Hz de frequência.

Por fim o sinal da válvula se comporta como o esperado, um sinal de 28.5V a 30V, indicando que ela está desligada, o que libera o circuito de óleo do PCJ (lógica inversa de acionamento, já que o componente utiliza a estratégia de acionamento *low side driver*). Cabe ressaltar que o ruído induzido no sinal da galeria principal,

proveniente do sinal de alta frequência enviado aos injetores de combustível, não é notado no sinal da galeria PCJ que está isolado da placa que controla o motor e injetores, além do fato de cabos blindados terem sido utilizados para conduzir o sinal dos sensores, reduzindo a chance de absorver interferências do ambiente.

A próxima etapa analisa o comportamento do sensor PCJ com o acionamento da válvula, tempo de resposta do sinal e se há algum comportamento não esperado. A ponteira de tensão do sinal foi ajusta de 2V por nível para 1V por nível para melhor representar o comportamento, bem como o tempo da amostra foi elevado para analisar o comportamento ao longo do teste. A figura 3.6 mostra o sinal obtido e os pontos de acionamento da válvula e valores de pressão retirados da curva do sensor.

FIGURA 3.6 – Resposta dos sensores com o acionamento da válvula.



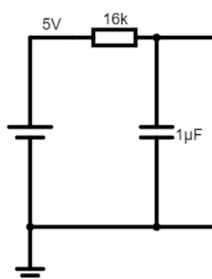
Fonte: O AUTOR (2019)

Pelos gráfico obtidos dos sinais elétricos, podemos verificar que o comportamento dos sinais elétrico está ocorrendo como esperado, com a variação de tensão do sensor e estabilização do sinal em 1.5 segundos após o acionamento da válvula solenoide (gerado pela resposta das válvulas mecânicas que controlam a pressão do sistema de PCJ e pelo tempo de pressurização das galerias), com nenhuma resposta do sensor fora da faixa de 0.5V até 2V, estes são valores importantes para a programação do software de verificação.

3.3.2 Segundo teste de motor instrumentado

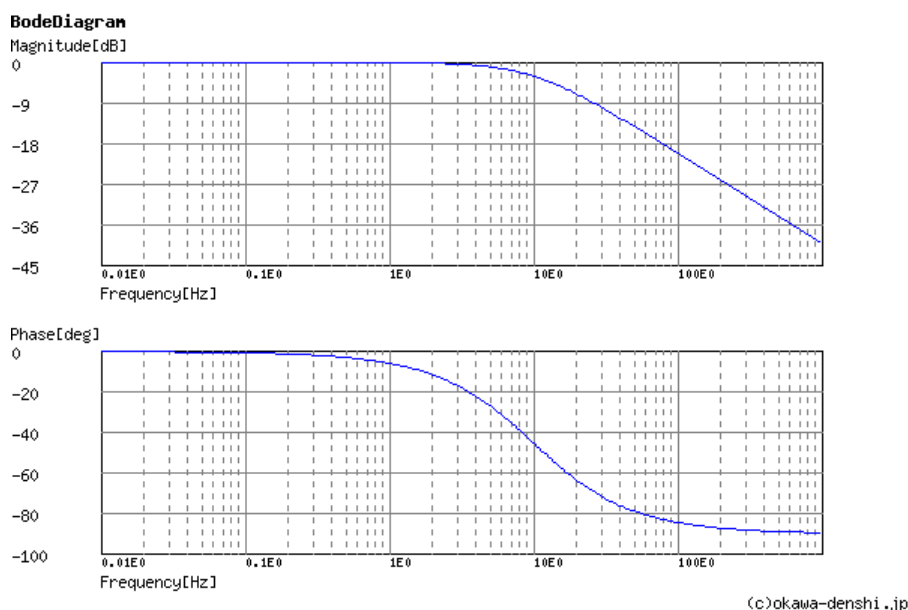
Para garantir que os resultados obtidos no primeiro teste refletem a realidade de outras amostras de motores, foi novamente instrumentado outro motor com os mesmos sensores. Porém, desta vez foi utilizado um filtro de ruídos para remover o sinal de tensão que foi obtido durante o primeiro teste. Como a resposta do sensor de pressão tem sua variação relativamente lenta, foi projetado um filtro de 1ª ordem que tem frequência de corte a partir de 10 Hz, possuindo uma atenuação relevante na casa dos 50 Hz para as primeiras harmônicas do sinal de ruído. O circuito do filtro utilizado está ilustrado na figura 3.7 bem como o a resposta em frequência do sistema na figura 3.8.

FIGURA 3.7 – Filtro RC primeira ordem.



Fonte: O AUTOR (2019)

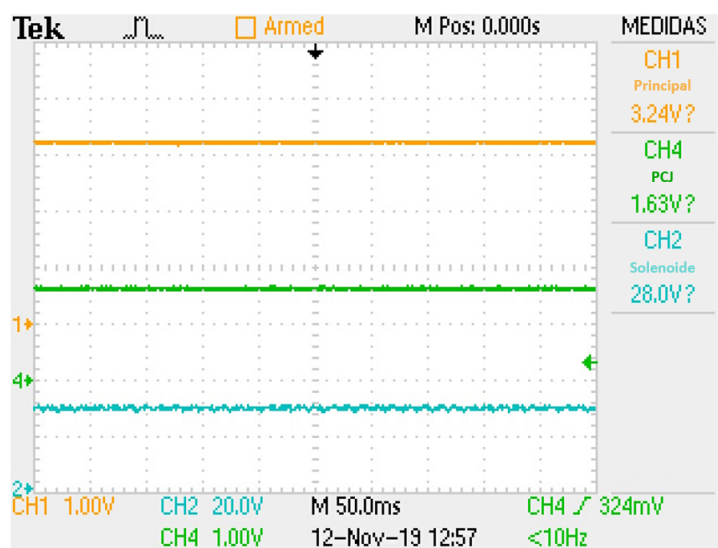
FIGURA 3.8 – Resposta do filtro RC de primeira ordem $\omega_c = 10\text{Hz}$.



Fonte: O AUTOR (2019)

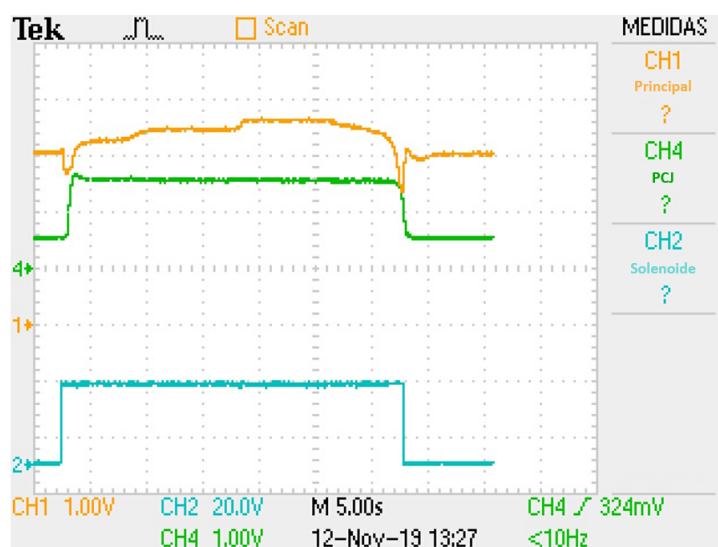
Novamente foram analisadas as respostas dos sensores durante a partida do motor e durante o processo de acionamento da solenoide. As curvas obtidas estão nas figuras 3.9 e 3.10, onde podemos verificar que, com a atuação do filtro, os ruídos indesejados foram removidos dos sinais dos sensores, deixando mais confiável o valor de leitura. O sinal do segundo teste é compatível com o observado no primeiro, logo é mais seguro afirmar que os valores obtidos durante os dois testes podem ser replicados para todos os motores do mesmo modelo instrumentados com o sensor de pressão do PCJ.

FIGURA 3.9 – Resposta dos sensores na partida do motor filtrados.



Fonte: O AUTOR (2019)

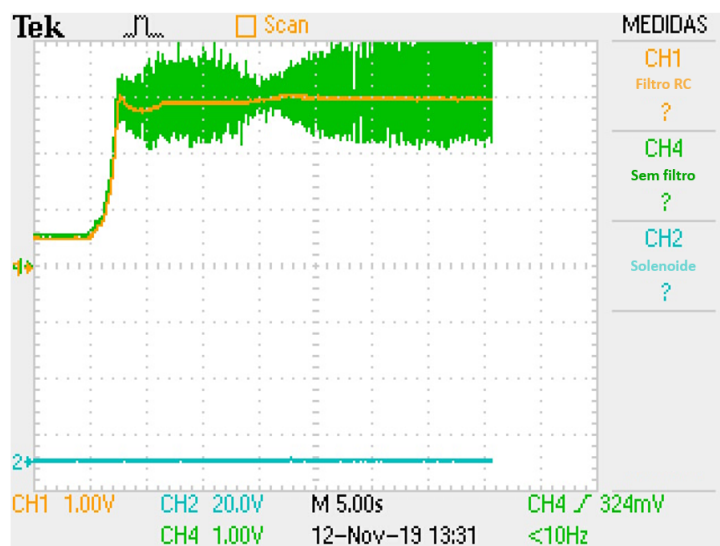
FIGURA 3.10 – Resposta dos sensores com o acionamento da válvula filtrados.



Fonte: O AUTOR (2019)

Para ilustrar o funcionamento do filtro RC foi realizado a medição do sinal antes e depois do filtro, utilizando o sinal da galeria principal. Podemos observar na figura 2.4 como o sinal de ruído é atenuado pelo filtro, o que comprova que o filtro RC apesar de simples, é bem eficiente para remover ruídos do sinal.

FIGURA 3.11 – Comparação do sinal na entrada e na saída do filtro RC.



Fonte: O AUTOR (2019)

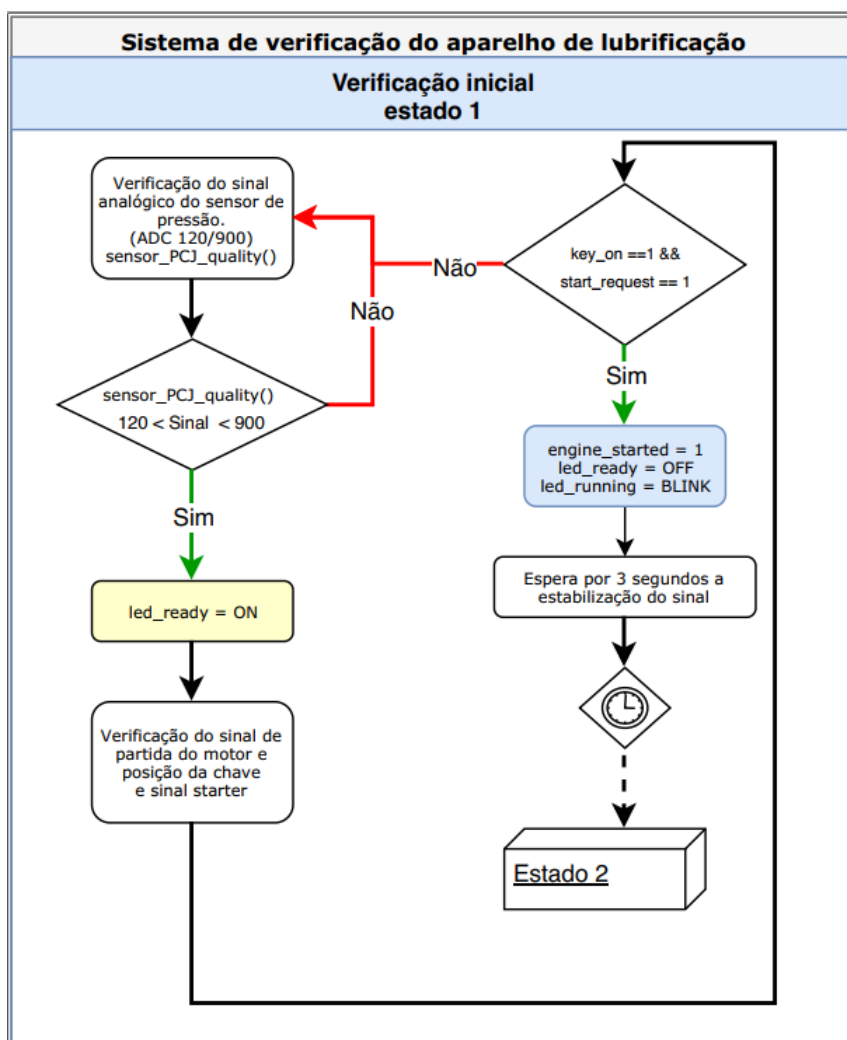
Com os testes realizados podemos agora inferir o comportamento dos sensores e atuadores a serem monitorados pelo software, com os limites e tempos de repostas necessários.

3.4 PROGRAMAÇÃO DO MICROCONTROLADOR

O desenvolvimento do software consiste em conferir o comportamento de cada componente durante a realização do teste. Para que esse fluxo de funcionamento possa ser desenvolvido, um fluxograma do software foi elaborado, facilitando a compreensão e implementação das funções e laços do programa, que foi dividido em três laços principais: verificação de início do teste, realização e monitorando do teste e envio dos resultados. A primeira parte está na figura 3.12, que consiste em basicamente verificar o sinal do sensor de PCJ e determinar se ele está realmente conectado ao motor e ao chicote elétrico. Isso é possível devido aos valores constantes do sensor quando está em operação normal, entre 0.5V e 4.5V. Após esse sinal ser confirmado o software espera o sinal de chave ligada e o de partida do motor, com isso é possível afirmar que

o teste foi realmente iniciado e espera-se 3 segundos pela estabilização dos sinais do motor, que não são relevantes para o teste durante o início da partida.

FIGURA 3.12 – Fluxograma do primeiro laço do software.



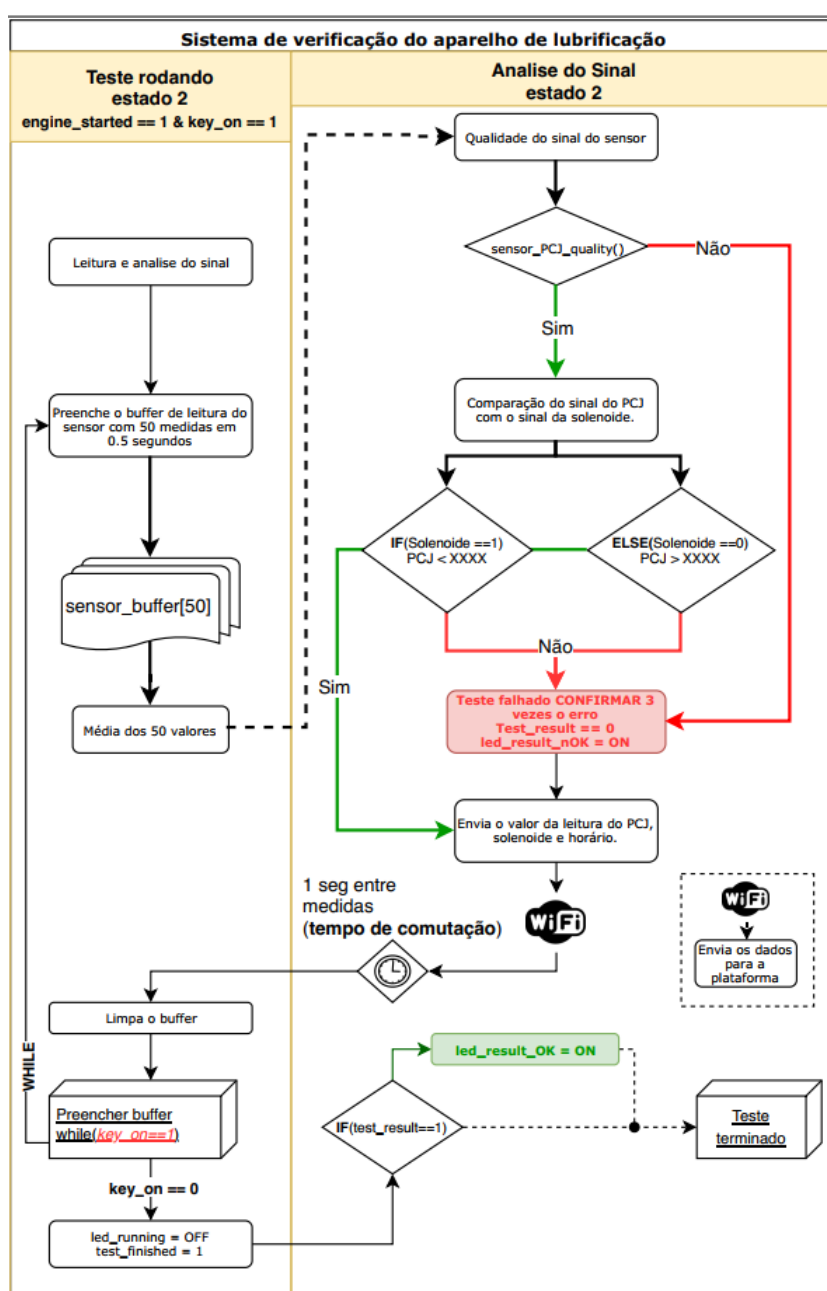
Fonte: O AUTOR (2019)

Com o início do teste o software deve verificar a condição da válvula e do sensor de pressão, determinando se o valor de pressão lido está de acordo com o valor esperado pela lógica do sistema, este valor do sensor é obtido dentre uma média de 50 amostras retiradas em 0.5 segundos, evitando que sinais fora do normal sejam interpretados de maneira errada pelo algoritmo. Logo, se a solenoide está com o sinal alto, o PCJ deve estar com a pressão alta, com o contrário sendo também válido. Este fluxo está na figura 3.13, que mostra o looping de verificação do sinal da solenoide e PCJ. Cabe ressaltar que toda vez que a leitura é realizada a qualidade do sinal é comparada, para que seja possível detectar que o teste falhou por conta de uma

desconexão do sensor.

Após a leitura do sinal e comparações, esses valores são enviados a uma API na nuvem que irá mostrar um gráfico dos valores lidos e após isso o buffer é limpo e o laço se repete até que o sinal de chave ligada do motor seja desligado, indicando que o teste terminou. Durante a realização do teste as luzes do dispositivo e reles ligados ao CLP são chaveados conforme o resultado das comparações, as quais precisam falhar ao mínimo 3 vezes para confirmar que os componentes do teste estão falhados.

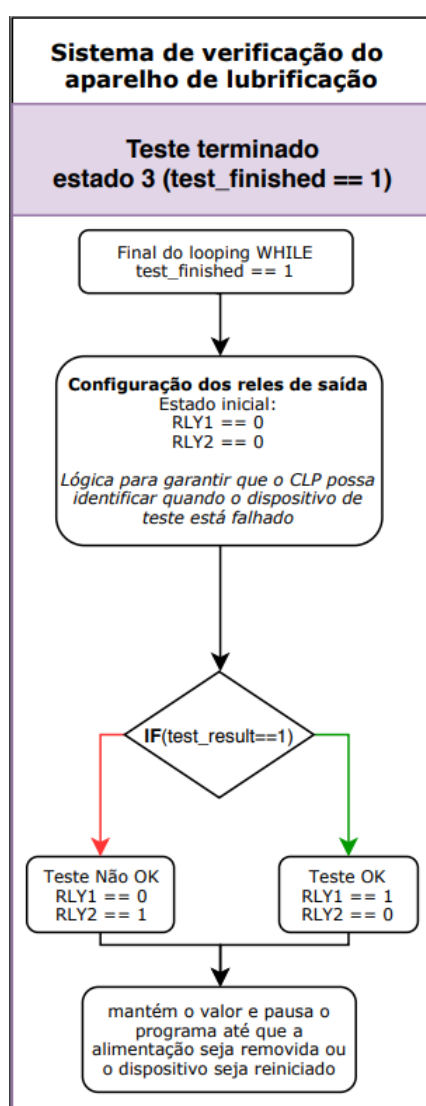
FIGURA 3.13 – Fluxograma do segundo laço do software.



Fonte: O AUTOR (2019)

Por fim o último laço do programa, ilustrado na figura 3.14, é responsável somente por segurar os valores obtidos no teste até que o CLP realize a leitura dos reles, que acontece logo após o término do ciclo e desligamento do motor. Com isso o dispositivo pode ser desligado e removido, iniciando novamente o ciclo do algoritmo quando for realimentado com a tensão de entrada de 5V.

FIGURA 3.14 – Fluxograma do terceiro laço do software.



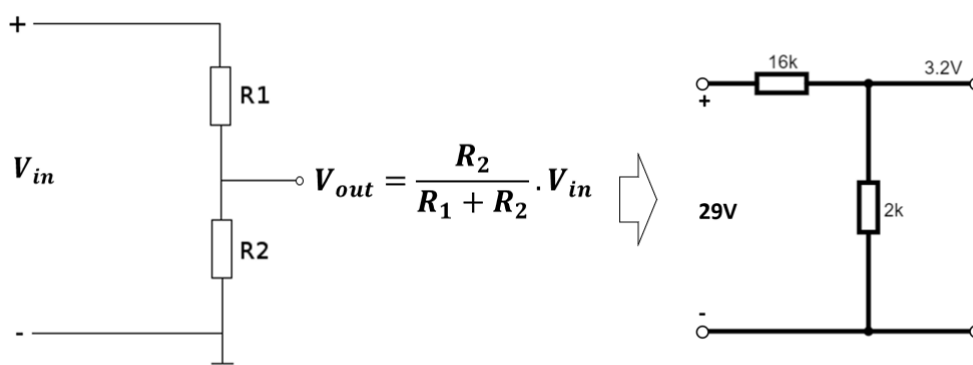
Fonte: O AUTOR (2019)

Com o fluxo do algoritmo elaborado, foi iniciado a programação em linguagem C do microcontrolador, anexada na seção 6.1 do Apêndice A. O próximo passo é elaborar um esquemático elétrico do sistema, para que seja possível verificar se todas as entradas e saídas são compatíveis com o microcontrolador escolhido e planejar os componentes que devem ser adquiridos para a confecção da placa de circuito impresso.

3.5 ESQUEMÁTICO DO CIRCUITO ELÉTRICO

A elaboração do esquemático do circuito elétrico foi realizado a partir da análise do comportamento de cada sinal a ser medido. Primeiramente os sinais de entrada digitais deveriam ter sua tensão ajustada, já que o sinal vindo do motor tem o nível de tensão em 29V e as portas de entrada digital do módulo ESP32 são projetadas para receber sinais de até 3.3V em nível lógico alto e alta impedância de entrada, portanto o sinal pode ser reduzido através de um divisor de tensão resistivo. Os divisores de tensão resistivos são circuitos simples compostos por dois resistores em série que fornecem uma tensão reduzida no ramo que se encontra entre eles, uma solução barata e eficiente, desde que o sinal não precise ser regulado. O exemplo deste tipo de circuito e os valores comerciais utilizados no projeto, estão ilustrados na figura 3.15. Com isso todos os sinais de chave ligada, partida e acionamento da solenoide, são regulados para as tensões suportadas pelo microcontrolador e possíveis de serem processados sem danos aos componentes.

FIGURA 3.15 – Divisor de tensão resistivo utilizado nas entradas digitais.

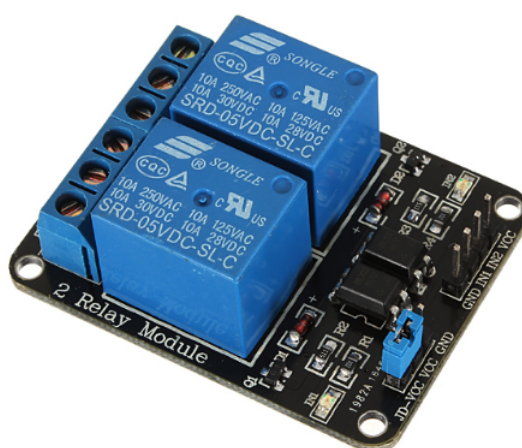


Fonte: O AUTOR (2019)

O próximo sinal que foi analisado foram as saídas digitais, que devem fornecer níveis de tensão para o acionamento dos reles e dos LEDs de indicação de funcionamento do dispositivo. De acordo com o manual do fabricante da placa, cada porta de uso geral GPIO pode suportar até 20mA de corrente, portanto os dispositivos ligados a estas portas não podem exceder esse limite com o risco de dano ao microcontrolador. A corrente de alimentação dos LEDs é facilmente controlada pela adição de resistores em série com o circuito, limitando a corrente a 10mA com a adição de um resistor de

150 Ohms. Para o acionamento dos reles, visando a simplificação do circuito e maior facilidade em repor componentes, foi escolhido a utilização de um módulo de dois canais como o da figura 3.16, que já possui todo o circuito de acionamento utilizando transistores e opto-acopladores, excluindo a complexidade de projetar este tipo de circuito.

FIGURA 3.16 – Módulo Relé 5V 2 Canais



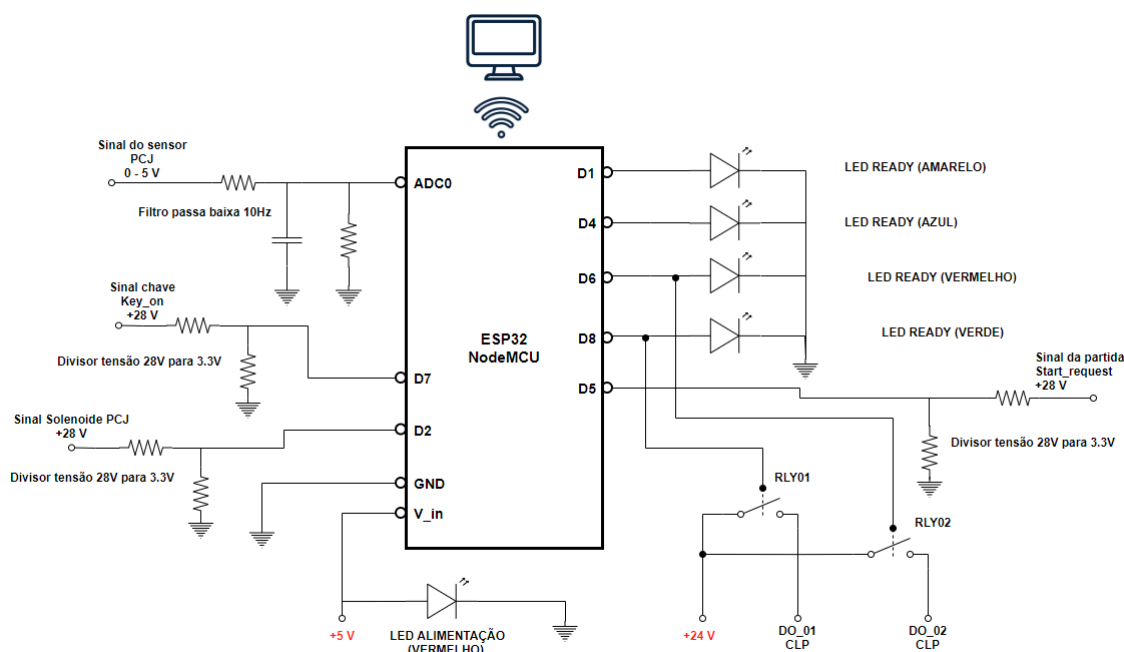
Fonte: FLIFLOP.COM (2019)

Este módulo possui corrente de operação entre 15mA e 20mA, portanto está entre o suportado pela placa. Porém o nível de tensão comum a ser utilizado para o acionamento do módulo é de 5V, diferente do fornecido pelo microcontrolador que é de 3.3V. Uma possível solução é realizar a ligação do módulo de maneira que os reles sejam acionados por uma fonte de 5V, a mesma que alimenta nosso microcontrolador, e a polarização dos transistores do módulo seja realizado utilizando uma tensão de 3.3V das saídas digitais do ESP32. O módulo aciona os reles através de um nível de tensão lógico baixo, portanto quando o LED e o acionamento do rele pode ser realizado utilizando a mesma saída digital, já que nenhum dos dois dispositivos vai estar acionado ao mesmo tempo.

Para a entrada analógica do circuito, que irá realizar a leitura do sensor de pressão do PCJ, já foi exposto a necessidade de utilizar um filtro passa baixa de primeira ordem, ilustrado na figura 3.7. Contudo o filtro ainda fornece uma saída de tensão entre 0V e 5V, o que não é compatível com a entrada do conversor analógico digital do módulo ESP32. Novamente a solução encontrada foi a utilização de um divisor resistivo após o circuito de supressão de ruídos, baixando a tensão de entrada do

conversor para níveis entre 0V e 3.3V que podem ser lidos e convertidos a valores de 0 a 1024 interpretados pelo microcontrolador. A partir das análises feitas foi elaborado um esquemático base para servir de início a produção da placa protótipo e posteriormente a placa de circuito impresso, já que com ele é possível saber todas as ligações que devem ser realizadas no ESP32 e suas respectivas portas de entradas e saídas, este esquemático está na figura 3.17.

FIGURA 3.17 – Primeiro esquemático do dispositivo



Fonte: O AUTOR (2019)

Havia uma preocupação inicial com relação as tensões utilizadas na alimentação do ESP32, porem a tensão de 5V que irá alimentar o dispositivo vem diretamente da controladora do motor, a qual possui diversos circuito de proteção e regulação da tensão. Esta regulação é essencial e também confiável, já que alimenta todos os sensores do motor que são regulados em 5V. Adicionalmente as proteções da controladora do motor, o próprio módulo ESP32 possui um regulador de tensão NCP1117 na entrada de alimentação V_{in} , que segundo o manual de especificações do componente, tem a capacidade de regular valores de tensão entrada de 4.75V até 10V e 800mA de corrente máxima.

4 RESULTADOS

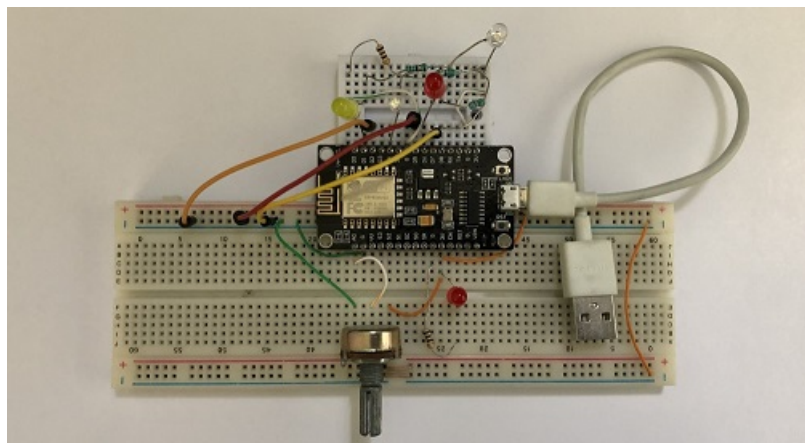
Através do desenvolvimento e da fundamentação realizados, se torna possível a integração dos conceitos em um sistema completo, os resultados foram obtidos seguem descritos nesta seção dedicada a exposição dos resultados obtidos.

Começando pela integração do fluxo de software com as entradas no hardware, com a elaboração do circuito em placa de prototipação, essencial para realizar os testes de funcionamento do sistema antes da elaborar uma placa de circuito impresso. Após testar os casos de uso e sinais na placa de prototipação, foi elaborado o diagrama elétrico final do sistema e o design da placa de circuito impresso, tornando possível a fabricação de um primeiro protótipo.

4.1 INTEGRAÇÃO DAS FUNÇÕES DO MICROCONTROLADOR

Após desenvolver o software com base no fluxo apresentado na seção de desenvolvimento, foi utilizada uma placa de prototipação para montar o primeiro circuito com base na análise dos sinais necessários. Para simular a atuação do sensor de pressão foi utilizado uma resistência variável de 1k Ohms, para que fosse estimulados valores de leitura no conversor analógico digital do ESP32. Já os sinais digitais de entrada foram também aplicados diretamente as respectivas portas utilizando a saída de 3.3V do módulo, facilitando esta primeira fase do processo de prototipagem. A figura 4.1 ilustra como ficou a montagem do primeiro protótipo.

FIGURA 4.1 – Circuito em placa de prototipagem



Fonte: O AUTOR (2019)

A programação do microcontrolador é realizada através da interface USB do módulo, onde é possível carregar o código em C criado e executá-lo. Durante esta fase foi verificado que nem todas as portas que estão relacionadas como GPIO podem facilmente serem utilizadas, já que durante o processo de inicialização da placa elas tem um papel específico no processo, e somente depois de iniciada podem ser configuradas como entradas ou saídas. Por exemplo, as portas RX e TX do módulo, quando usadas como entradas e saídas, não permitem que o módulo seja programado, já que estas portas realizam a comunicação entre o computador e o microcontrolador.

A placa de prototipagem foi essencial durante o processo de desenvolvimento do software, permitindo que cada nova função fosse implementada no módulo e testada quanto ao seu comportamento, processo que muitas vezes não é eficiente quando o software é verificado de maneira isolada.

4.1.1 Teste do comportamento esperado do microcontrolador

Com uma primeira versão do software já desenvolvido, foram realizados alguns testes para certificar que o sistema não possui nenhum comportamento estranho quanto a seu funcionamento. Para isso o teste foi separado em três partes, sendo a primeira relativo ao laço de verificar inicial do sistema, o segundo no laço de verificação de parâmetros do sensor e configuração dos resultados do teste e o último laço que mantém o resultado na saída do sistema.

Na primeira parte o sistema precisa estar inicialmente com os valores corretos de leitura no sensor corretos, indicando que o sensor está de fato conectado, e para isso foi testado se o sistema não iniciava mesmo com o resistor variável desconectado ou valores muito baixos e altos na leitura do ADC, que devem estar entre 120 e 900, conforme a curva do sensor na figura 3.4. Outro caso testado foi para quando a leitura do sensor está correta e a chave está desligada, tentando iniciar o sistema com o sinal de partida, o que não deve ocorrer.

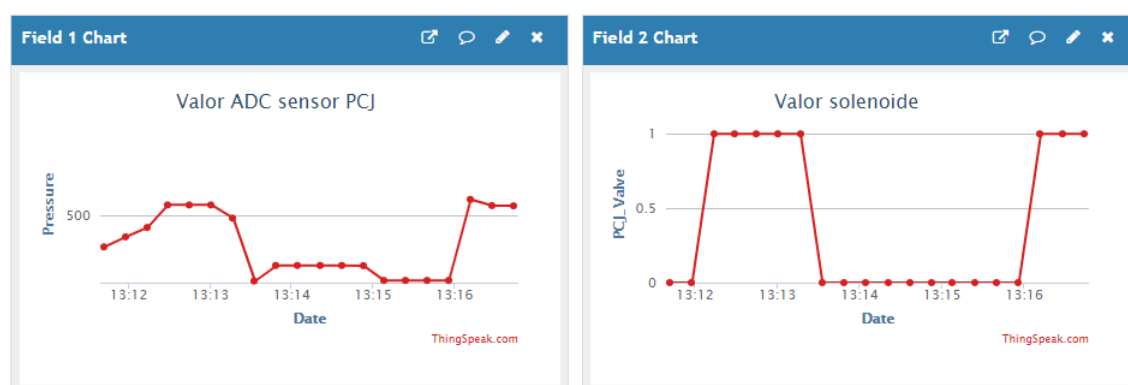
Por fim a condição correta de acionamento do sistema foi testada, com os valores do sensor e chave ligada, passando o sistema para a fase de análise do teste de motor. Todo o processo de verificação dos sinais iniciais é indicado de maneira visual, com o LED amarelo piscando em quanto a placa está alimentada e sem um sinal válido do sensor de pressão, passando a ligada quando o sinal está correto. O sinal de

chave ligado é indicado quando o LED azul acende e o teste foi iniciado quando o LED azul começa a piscar.

Durante a análise do teste o módulo ESP32 deve comparar os valores de tensão do sensor com o valor presente na entrada digital da solenoide, verificando se o valor está coerente com a lógica de acionamento do sistema. Esse processo se repete em quanto o sinal de chave está ligado, não havendo nenhuma relação com o sinal de partida do motor que já foi processado na etapa anterior. Foi testado se o sistema acusa falha quando o resistor variável era removido ou quando os valores estavam fora da faixa de operação, bem como todas as combinações lógicas entre o sinal da válvula e os sinais de pressão. Todo o processo é indicado de maneira visual pelos LED que indica o teste rodando, o qual deve piscar durante o processo de teste, e os LEDs vermelho e verde indicam o resultado do teste até o momento.

A partir de três resultados não coerentes com a lógica de funcionamento esperado das válvulas e da pressão do sistema, o teste entra em modo de falha, não retornando novamente a condição inicial que é a do sistema com correto funcionamento. Durante todo o processo do teste, os valores do sensor e da solenoide são enviados via WiFi a plataforma API na nuvem *ThingSpeak*, onde podemos através de um navegador web visualizar o comportamento dos sinais em tempo real. Porém a plataforma API possui uma limitação para usuários fora do plano pago, o que limita o envio de dados a cada 15 segundos para o servidor, dificultando a visualização dos dados no gráfico, a figura 4.2 mostra como são mostrados os gráficos na plataforma e que são atualizados a cada 15 segundos, formando um gráfico parecido com o observado no osciloscópio.

FIGURA 4.2 – Gráficos gerados pela API *ThingSpeak*



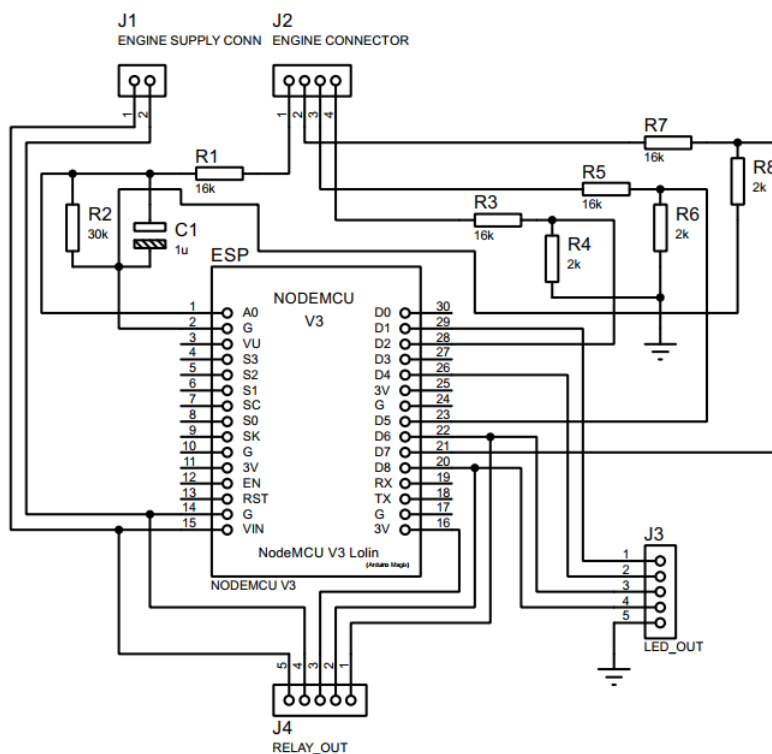
Fonte: O AUTOR (2019)

Por fim, a última parte do algoritmo é responsável somente por replicar o resultado do teste para os reles, enviando os sinais lógicos de 24V ao CLP. Nesta etapa o teste era bem simples, verificando se mesmo alterando todos os sinais do módulo, ele não alterava o estado dos LEDs. Após realizar todos os testes e obter resultados satisfatórios, o algoritmo está confiável para ser utilizado, e a placa de circuito impresso pode ser projetada utilizando o mesmo circuito montado na placa de prototipagem.

4.2 PLACA DE CIRCUITO IMPRESSO DO DISPOSITIVO

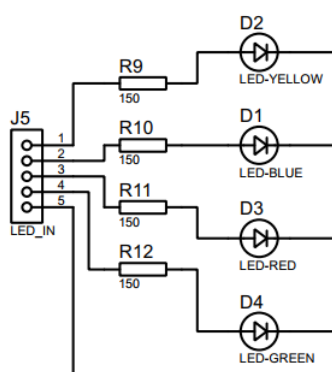
Após o processo de prototipagem do circuito e testes de software, foi desenhado o esquemático da placa de circuito impresso, implementando todas as mudanças necessárias e circuitos de conversão de níveis de tensão mencionados na seção de desenvolvimento. O esquemático desenvolvido estão nas figuras 4.3 e 4.4, os circuitos foram divididos em dois esquemáticos, um para a placa principal com todo os sinais do motor e do módulo e outro com os LEDs, isso facilita a fixação dos LEDs na caixa que irá acomodar o circuito.

FIGURA 4.3 – Esquemático da placa principal



Fonte: O AUTOR (2019)

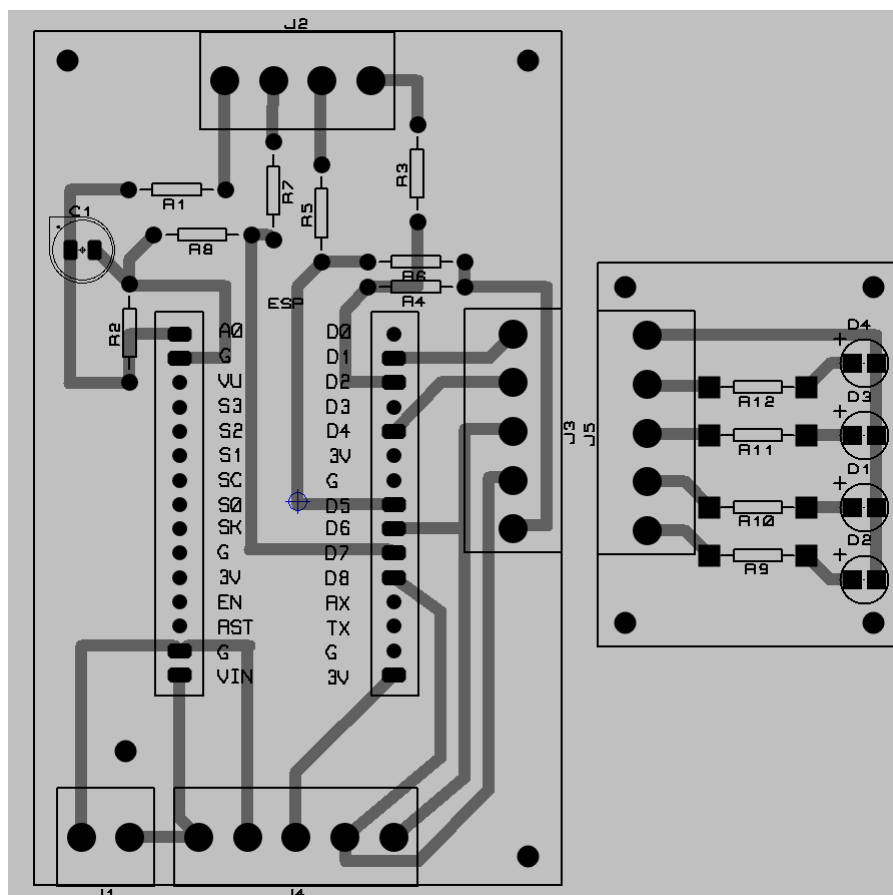
FIGURA 4.4 – Esquemático da placa LED



Fonte: O AUTOR (2019)

Após a realização do esquemático, foi realizado a elaboração da placa de circuito impresso, posicionando os componentes e trilhas relacionados no circuito. O *layout* final das placas está na figura 4.5, que pode ser utilizado para impressão e confecção das placas do sistema de verificação proposto.

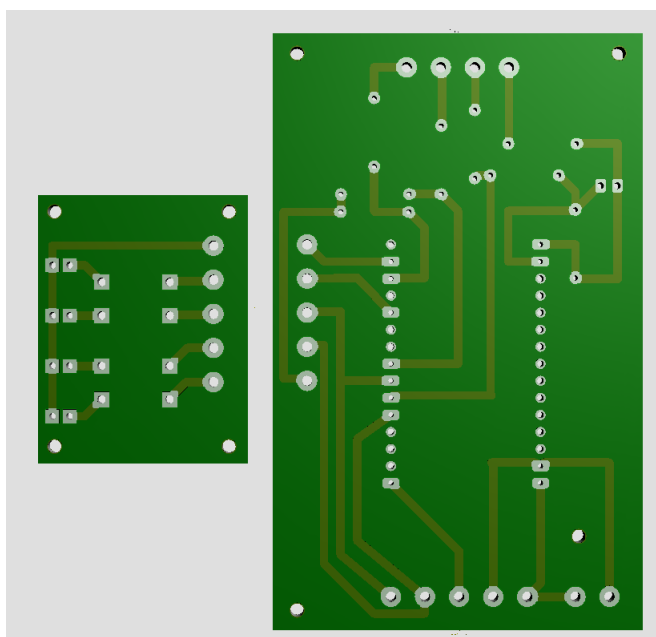
FIGURA 4.5 – Layout da placa principal e LED



Fonte: O AUTOR (2019)

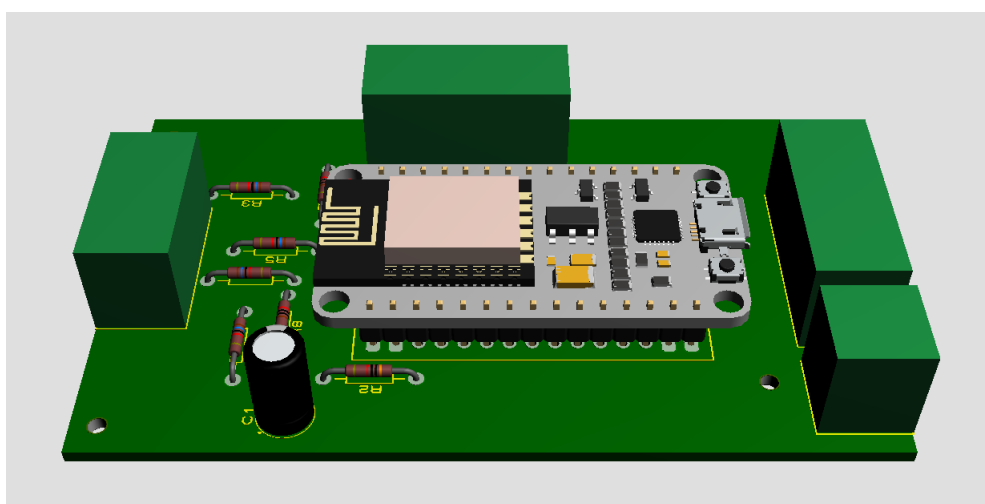
Agora é possível fabricar uma placa de protótipo em placa de fenolite, utilizando qualquer um dos processos disponíveis para elaboração de circuitos impressos e obter como resultado a figura 4.6, soldar os componentes na placa e possuir uma primeira amostra do produto, que pode ser novamente testada e validada. A aparência das placas pode ser vista nas figuras 4.7 e 4.8.

FIGURA 4.6 – Vista das placas impressas



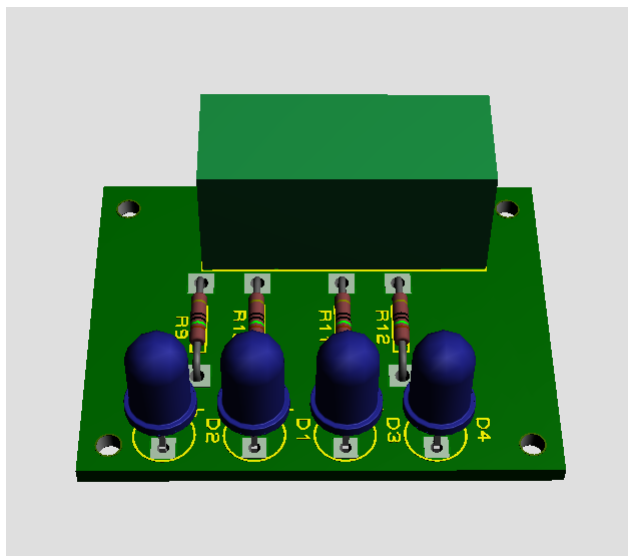
Fonte: O AUTOR (2019)

FIGURA 4.7 – Vista 3D da placa principal



Fonte: O AUTOR (2019)

FIGURA 4.8 – Vista 3D da placa LED



Fonte: O AUTOR (2019)

Para acomodar o circuito foi decidido utilizar uma caixa plástica universal para circuitos elétricos, facilmente encontrada em lojas de materiais eletrônicos. Assim, caso ocorra dano a caixa do produto, ela pode ser rapidamente substituída. A caixa utilizada está ilustrada na figura 4.9, necessitando somente de alguns furos adicionais para acomodar e fixar as placas principal e LED do sistema.

FIGURA 4.9 – Caixa plástica universal para circuitos elétricos



Fonte: O AUTOR (2019)

5 CONCLUSÃO

Através de toda a fundamentação teórica e desenvolvimento do projeto, foram encontrados diversos problemas que não estavam sendo considerados quando na concepção do trabalho. Tais problemas foram sendo solucionados ao longo do processo de estudo e experimentação, ajudando a consolidar os conceitos aprendidos ao longo do curso. Analisando os resultados obtidos no projeto, é interessante ressaltar que foram utilizados diversos conceitos de matérias diferentes cursadas ao longo da graduação, o que foi fundamental para a conclusão do trabalho e resolução dos problemas enfrentados no decorrer do caminho.

Em relação ao produto, ele cumpre todos os requisitos de funcionalidade propostos no início do projeto, funcionando como um dispositivo confiável para a validação do sistema de lubrificação em veículos pesados, além de possuir grande flexibilidade para utilização dos componentes e possibilidade de produzir diversas unidades. A proposta de um sistema que não precisasse de intervenção humana durante o processo foi entregue, já que o operador da sala de testes precisa somente conectar o sensor ao sistema de teste e iniciar a sequência normal de validação do motor, ao mesmo tempo que o dispositivo realiza as leituras e comparações dos sinais recebidos do motor. Ao final do teste o operador irá receber somente o resultado na tela do computador que está conectado ao CLP, podendo remover o motor do processo comum de fabricação e encaminhando ele para que seja solucionado a falha no sistema de lubrificação.

Um dos pontos que no início era considerado, foi a utilização de uma API web que permitisse a visualização dos dados em tempo real, porem ao decorrer do desenvolvimento foi descoberto que este tipo de API é paga para a utilização com taxas menores do que uma amostra a cada 15 segundos, contudo o objetivo do projeto era transmitir um sinal de tensão em nível lógico para o controlador CLP da fábrica, sendo a API somente um meio auxiliar no processo de análise do sinal.

A parte de validação do software e hardware desenvolvidos foi fundamental para comprovar o funcionamento do produto, junto com a prototipação do mesmo. Isso fornece maior confiança para seguir em frente com o projeto e incrementar a maturidade do produto. Os próximos passos incluem realizar a impressão de uma placa

de circuito impresso por uma empresa especializada, recebendo uma placa que tenha vida útil muito maior do que as fabricadas em processos caseiros.

O custo necessário para a fabricação do produto é em torno de R\$116,00 para compra dos componentes e impressão das placas, que é irrelevante perto ao custo estimado das falhas em campo de R\$200.000,00 por veículo, além do fato de eliminar a chance do problema ocorrer com o veículo sendo utilizado por um cliente e todo o tempo de reparo necessário para a troca do motor. O preço de cada componente pode ser conferido na figura 5.1, que foi extraída do software utilizado no desenho e confecção dos esquemáticos e *layouts*.

FIGURA 5.1 – Relação de materiais utilizados no projeto

1 Capacitores			
Quantity	References	Value	Unit Cost
1	C1	1u	R\$0,10
Sub-totals:			R\$0,10
12 Resistores			
Quantity	References	Value	Unit Cost
4	R1,R3,R5,R7	16k	R\$0,08
1	R2	30k	R\$0,08
3	R4,R6,R8	2k	R\$0,08
4	R9-R12	150	R\$0,08
Sub-totals:			R\$0,96
4 Diodos			
Quantity	References	Value	Unit Cost
1	D1	LED-BLUE	R\$0,25
1	D2	LED-YELLOW	R\$0,25
1	D3	LED-RED	R\$0,25
1	D4	LED-GREEN	R\$0,25
Sub-totals:			R\$1,00
8 Geral			
Quantity	References	Value	Unit Cost
1	CAIXA UNIVERSAL1	BOX	R\$20,00
1	ESP	NODEMCU V3	R\$38,00
1	J1	ENGINE SUPPLY CONN	R\$0,90
1	J2	ENGINE CONNECTOR	R\$1,20
1	J3	LED_OUT	R\$1,50
1	J4	RELAY_OUT	R\$1,50
1	J5	LED_IN	R\$1,50
1	PCB1	IMPRESSAO	R\$50,00
Sub-totals:			R\$114,60
Totals:			R\$116,66

Fonte: O AUTOR (2019)

5.0.1 Trabalhos futuros

Para a apresentação do produto foi fabricado um protótipo inicial, portanto podemos planejar os trabalhos futuros de melhoria do projeto, já que o conceito foi provado como funcional, é possível implementar novas melhorias no produto e fabricar placas a partir de indústrias especializadas.

- Fabricação das placas em circuito impresso em indústria especializada, aumentando a confiabilidade e vida útil dos componentes do sistema.
- Realizar o projeto de um case específico para o produto, incluindo pontos de fixação para que ele possa ser colocado dentro das salas de teste do motor e fixado de maneira firme, ficando longe de áreas quentes do motor;
- Caso a análise do sinal em tempo real seja uma necessidade, pode ser implementado um servidor local no próprio módulo ESP32, onde é possível armazenar uma página em HTML que irá mostrar os resultados de pressão e acionamento da válvula em tempo real. Essa página pode ser acessada por qualquer dispositivo conectado a mesma rede WiFi do ESP32;

REFERÊNCIAS

- BALBINOT, A.; BRUSAMARELLO, V. J. *Instrumentação E Fundamentos de Medidas. Volume 2*. [S.l.]: Grupo Gen-LTC, 2000. Citado na página 19.
- BRUNETTI, F. Motores de combustão interna. *São Paulo: Blucher*, v. 1, 2012. Citado 4 vezes nas páginas 27, 28, 30 e 32.
- DIESELNET.COM. *Piston Cooling*. 2017. Disponível em: <https://www.dieselnets.com/tech/combustion_piston-cool.php>. Acesso em: 16 de ago. de 2019. Citado na página 33.
- ESTADÃO. *SÓ 48% DAS INDÚSTRIAS BRASILEIRAS SÃO '4.0', DIZ CNI*. 2016. Disponível em: <<https://revistapegn.globo.com/Empreendedorismo/noticia/2016/05/so-48-das-industrias-brasileiras-sao-40-diz-cni.html>>. Acesso em: 13 de ago. de 2019. Citado na página 14.
- HAWKINS, N. *New catechism of the steam engine, with chapters on gas, oil and hot air engines*. [S.l.]: T. Audel, 1903. Citado na página 27.
- HONEYWELL. *R300 Series Temperature Sensor Datasheet*. 2008. Disponível em: <<https://sensing.honeywell.com/007443-2-en.pdf>>. Acesso em: 29 de ago. de 2019. Citado 2 vezes nas páginas 25 e 26.
- MAHLE. *CURSO MAHLE METAL LEVE MOTORES DE COMBUSTÃO INTERNA*. [S.l.: s.n.], 2016. Citado 5 vezes nas páginas 27, 28, 29, 31 e 32.
- OLIVEIRA, S. de. *Internet das coisas com ESP8266, Arduino e Raspberry PI*. [S.l.]: Novatec Editora, 2017. Citado 2 vezes nas páginas 16 e 17.
- PINAZZA, M. *GESTÃO DE PROJETOS: As fases do ciclo de vida de um projeto, a função do gestor de projetos e a importância do gerenciamento*. 2017. Disponível em: <<http://movimentoimpactoglobal.com.br/fases-do-projeto/>>. Acesso em: 13 de ago. de 2019. Citado na página 14.
- SACK, H. *Nikolaus Otto and the Four Stroke Engine*. 2019. Disponível em: <<http://scih.org/nikolaus-otto-four-stroke-engine/>>. Acesso em: 16 de ago. de 2019. Citado na página 27.
- TANENBAUM, A. S. *Sistemas operacionais modernos*. [S.l.]: Prentice-Hall, 2009. v. 3. Citado na página 15.
- THOMAZINI, D.; ALBUQUERQUE, P. U. B. d. Sensores industriais: fundamentos e aplicações. *São Paulo*, v. 3, p. 32, 2005. Citado 3 vezes nas páginas 21, 22 e 24.
- WOMACK, J. P.; JONES, D. T. *A máquina que mudou o mundo*. [S.l.]: Gulf Professional Publishing, 2004. Citado 2 vezes nas páginas 10 e 12.

6 APÊNDICE A

6.1 CÓDIGO DO MICROCONTROLADOR

```
#include <ESP8266WiFi.h>

#define led_ready 5
#define led_running 2
#define led_result_ok 15
#define led_result_Nok 12
#define key_on_DI 13
#define start_request_DI 14
#define solenoid_PCJ_DI 4

#define SSID_REDE "SSID_REDE"
#define SENHA_REDE "senhaWiFi"
#define INTERVALO_ENVIO_THINGSPEAK 15000
// intervalo ThingSpeak (em ms)

//API and WiFi settings
char EnderecoAPIThingSpeak[] = "api.thingspeak.com";
String ChaveEscritaThingSpeak = "IYYOIXZF7RPSQWFI";
long lastConnectionTime;
WiFiClient client;

const int analogPin = A0;
// ESP8266 Analog Pin ADC0 = A0

int state = 1;
int cont_failures = 0;
int cont_key_off = 0;
bool test_final_result = true;

void setup() {
```

```

pinMode(led_ready, OUTPUT);
pinMode(led_running, OUTPUT);
pinMode(led_result_ok, OUTPUT);
pinMode(led_result_Nok, OUTPUT);
pinMode(key_on_DI, INPUT);
pinMode(start_request_DI, INPUT);
pinMode(solenoid_PCJ_DI, INPUT);
Serial.begin(115200);
lastConnectionTime = 0;
FazConexaoWiFi();
digitalWrite(led_ready, HIGH);
delay(500);
digitalWrite(led_ready, LOW);
digitalWrite(led_running, HIGH);
delay(500);
digitalWrite(led_running, LOW);
digitalWrite(led_result_Nok, HIGH);
delay(500);
digitalWrite(led_result_Nok, LOW);
digitalWrite(led_result_ok, HIGH);
delay(500);
digitalWrite(led_result_ok, LOW);
}

// the main loop function
void loop() {

  switch(state){
    //INITIAL CHECK
    //(SENSOR QUALITY && KEY ON && START SIGNAL REQUEST)
    case 1:
      if(sensor_PCJ_quality()!=0){
        digitalWrite(led_ready, HIGH);

```

```

    if(key_signal_on() == true){
        digitalWrite(led_running, HIGH);
        if(start_signal_on() == true){
            digitalWrite(led_ready, LOW);
            state = 2;
        } //test started
    }

    else digitalWrite(led_running, LOW);
}

else{
    digitalWrite(led_running, LOW);
    digitalWrite(led_ready, HIGH);
    delay(250);
    digitalWrite(led_ready, LOW);
}

break;

//TEST RUNNING UNTIL KEY SIGNAL GOES OFF

case 2:
    if(key_signal_on()==true){
        int value = sensor_PCJ_quality();
        running_led_blink();
        if(value!=0){
            bool solenoid_value = solenoid_signal_on();
            if((solenoid_value==true && value>400)||
                (solenoid_value==false && value<400)){
                if(test_final_result==true) digitalWrite
                    (led_result_ok, HIGH);
                cont_failures=0;
            }
        }
        else{
            cont_failures = cont_failures+1;
            if(cont_failures>3){
                test_final_result=false;
            }
        }
    }
}

```



```

        digitalWrite(led_result_Nok, HIGH);
        digitalWrite(led_result_ok, LOW);
        serial_text_test_fail();
        //state = 3;
        //break;
    }
    delay(250);
}
}
else{
    cont_failures = cont_failures+1;
    if(cont_failures>3){
        test_final_result=false;
        digitalWrite(led_result_Nok, HIGH);
        digitalWrite(led_result_ok, LOW);
        serial_text_test_fail();
        //state = 3; //test finished
        //break;
    }
    delay(250);
}

//envia dados ThingsSpeak
if(!client.connected() &&
    (millis() - lastConnectionTime >
    INTERVALO_ENVIO_THINGSPEAK))
{
    int solenoid_status = 0;
    if(solenoid_signal_on()==true) solenoid_status=1;
    char sensor_PCJ_string[11];
    sprintf(sensor_PCJ_string,
    "field1=%i&field2=%i",sensor_PCJ_quality(),
    solenoid_status);

```

```

        EnviaInformacoesThingspeak(sensor_PCJ_string);
        client.stop();
    }
}

else{
    cont_key_off = cont_key_off+1;
    if(cont_key_off>4){
        serial_text_test_finished();
        state = 3; //test finished
        break;
    }
    delay(250);
}

break;

//FINAL LOOPING TO HOLD CLP RELAYS SIGNAL AND TEST RESULT
case 3:
    Serial.println("*****");
    Serial.println("TEST HAS FINISHED WAITING RESET");
    Serial.println("*****");
    delay(2000);

    break;

//ANY CASE OF ERROR ON VAR STATE
default:
    Serial.println("*****");
    Serial.println("ERROR OCURRED START ON CASE 1");
    Serial.println("*****");
    state = 1;

}

}

```

```

//*****FUNCTIONS*****

//Read and check if the sensor signal is good
int sensor_PCJ_quality(){
    int sensorValue = sensor_PCJ_read();
    if(sensorValue>100 & sensorValue<900){
        Serial.print("Sensor ADC: ");
        Serial.println(sensorValue);
        return sensorValue;
    }
    else{
        Serial.println("Sensor malfunction");
        return 0;
    }
}

//read 100 samples of analog sensor PCJ in half second
int sensor_PCJ_read(){
    int i;
    int sensor_buffer=0;
    int sensor_out=0;
    for (i=0; i<50; i++){
        sensor_buffer = sensor_buffer + analogRead(analogPin);
        delay(5);
    }
    sensor_out = sensor_buffer/50;
    return sensor_out;
}

//check if the key signal is ON
bool key_signal_on(){
    int val = 0;
    val = digitalRead(key_on_DI);
    if(val == 1){

```

```

        Serial.println("KEY_ON");
        return true;
    }
    else{
        Serial.println("KEY_OFF");
        return false;
    }
}

//check if the starter signal is ON
bool start_signal_on(){

    int val = 0;
    val = digitalRead(start_request_DI);
    if(val == 1){
        Serial.println("*****");
        Serial.println("START_REQUESTED");
        Serial.println("*****");
        return true;
    }
    else{
        Serial.println("START_OFF");
        return false;
    }
}

bool solenoid_signal_on(){

    int val = 0;
    val = digitalRead(solenoid_PCJ_DI);
    if(val == 1){
        Serial.println("SOLENOID_OFF");
        return false;
    }
}

```

```

    else{
        Serial.println("SOLENOID_ON");
        return true;
    }
}

void running_led_blink(){
    digitalWrite(led_running, HIGH);
    delay(250);
    digitalWrite(led_running, LOW);
}

void serial_text_test_fail(){
    Serial.println("*****");
    Serial.println("TEST FAIL");
    Serial.println("*****");
}

void serial_text_test_finished(){
    Serial.println("*****");
    Serial.println("TEST FINISHED");
    Serial.println("*****");
}

void FazConexaoWiFi(void)
{
    client.stop();
    Serial.println("Conectando-se à rede WiFi...");
    Serial.println();
    delay(1000);
    WiFi.begin(SSID_REDE, SENHA_REDE);
    while (WiFi.status() != WL_CONNECTED)
    {

```

```

        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi conectado com sucesso!");
    Serial.println("IP obtido: ");
    Serial.println(WiFi.localIP());
    delay(1000);
}

void EnviaInformacoesThingspeak(String StringDados)
{
    if (client.connect(EnderecoAPIThingSpeak, 80))
    {
        //faz a requisição HTTP ao ThingSpeak
        client.print("POST /update HTTP/1.1\n");
        client.print("Host: api.thingspeak.com\n");
        client.print("Connection: close\n");
        client.print
        ("X-THINGSPEAKAPIKEY: "+ChaveEscritaThingSpeak+"\n");
        client.print
        ("Content-Type: application/x-www-form-urlencoded\n");
        client.print("Content-Length: ");
        client.print(StringDados.length());
        client.print("\n\n");
        client.print(StringDados);
        lastConnectionTime = millis();
        Serial.println("TeamSpeak data SENT");
    }
}

```