

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E  
AUTOMAÇÃO

Giuliano Francisco Fioreze Cattani

**CONTRIBUIÇÃO À AUTOMAÇÃO DO ENSAIO DE ELEVAÇÃO DE  
TEMPERATURA DE TRANSFORMADORES DE DISTRIBUIÇÃO  
IMERSOS EM LÍQUIDO ISOLANTE**

Santa Maria, RS  
2023

Giuliano Francisco Fioreze Cattani

**CONTRIBUIÇÃO À AUTOMAÇÃO DO ENSAIO DE ELEVAÇÃO DE TEMPERATURA DE TRANSFORMADORES DE DISTRIBUIÇÃO IMERSOS EM LÍQUIDO ISOLANTE**

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Engenheiro de Controle e Automação**.

Orientador: Prof. Rafael Concatto Beltrame

Santa Maria, RS  
2023

**Giuliano Francisco Fioreze Cattani**

**CONTRIBUIÇÃO À AUTOMAÇÃO DO ENSAIO DE ELEVAÇÃO DE TEMPERATURA DE TRANSFORMADORES DE DISTRIBUIÇÃO IMERSOS EM LÍQUIDO ISOLANTE**

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Engenheiro de Controle e Automação**.

**Aprovado em 12 de julho de 2023:**

---

**Rafael Concatto Beltrame, Dr. (UFSM)**  
**(Presidente/Orientador)**

---

**Vitor Cristiano Bender, Dr. (UFSM)**

---

**Frederico Menine Schaf, Dr. (UFSM)**

Santa Maria, RS  
2023

## AGRADECIMENTOS

Inicialmente, gostaria de agradecer meus pais, Sílvia e Maximiliano, e meu irmão, Nicolás.

Também gostaria de agradecer meus amigos de longa data: Dieisson Vieira, Giovana Marion, Luis Henrique Medeiros, Mateus Piccinin, Matheus Lovato e Vinicius Denardin.

Durante a graduação tive o prazer de conhecer pessoas que, além de colegas de sala de aula, também viraram companheiros de estudos e amigos do peito. Obrigado, principalmente, Bruno Facco, Fernando Neves, Guilherme Basso e Theisson Becker. Demais colegas e professores, também não esqueci de vocês.

Um agradecimento especial, indispensável, a todos colegas e professores que conheci no INRIMT. Muito obrigado Ana Paula Marchesan, Élen Brodt, Manuella Farias, Rafael Beltrame, Rodrigo de Oliveira e Vitor Bender.

Gostaria de agradecer novamente meu professor orientador, Rafael Beltrame, por todo período de aprendizado. Obrigado pela paciência, por acreditar no meu trabalho e por todo apoio durante o desenvolvimento.

Um agradecimento especial à Yasmin. Não existem palavras pra descrever tua importância durante todo meu período de graduação. Obrigado.

## RESUMO

# CONTRIBUIÇÃO À AUTOMAÇÃO DO ENSAIO DE ELEVAÇÃO DE TEMPERATURA DE TRANSFORMADORES DE DISTRIBUIÇÃO IMERSOS EM LÍQUIDO ISOLANTE

AUTOR: Giuliano Francisco Fioreze Cattani

Orientador: Rafael Concatto Beltrame

No ano de 2001 publicou-se a Lei de Eficiência Energética, a qual estabeleceu regulamentações para a eficiência dos equipamentos elétricos. Então criou-se o Programa Brasileiro de Etiquetagem (PBE), o qual deu origem à Etiqueta Nacional de Conservação de Energia (ENCE). Nesse contexto, a portaria INMETRO N° 510 estabelece especificações para o programa de avaliação da conformidade para transformadores de distribuição em líquido isolante. Dentre os ensaios mandatórios a serem realizados, há o ensaio de elevação de temperatura, o qual envolve a medição da temperatura no topo do óleo do transformador, enquanto o transformador é submetido às suas perdas nominais. O ensaio é finalizado quando o gradiente de temperatura é inferior a 1 °C no intervalo de 3 horas, o que geralmente requer entre 10 e 12 horas de ensaio. Assim, buscando automatizar o ensaio de elevação de temperatura, este trabalho propõe o desenvolvimento de um circuito eletrônico de instrumentação para monitorar os sinais elétricos e térmicos provenientes do ensaio, incluindo um sistema supervisor e uma página Web. Foi realizada uma revisão bibliográfica abordando os tópicos contemplados na execução deste projeto. Dentre eles, pode-se citar instrumentação eletrônica, protocolos de comunicação, desenvolvimento Web e os equipamentos de medição presentes no Laboratório de Média Tensão do Instituto de Redes Inteligentes (INRIMT). O projeto da placa de circuito impresso (PCI) foi dividido em duas etapas principais. A primeira etapa consiste em realizar o projeto do circuito eletrônico de instrumentação e, em seguida, construí-lo. Após o teste e validação do circuito projetado, teve início a segunda etapa do projeto, que consiste no desenvolvimento do software, composto por um sistema supervisor e uma página Web. O sistema supervisor é responsável por receber os dados da PCI e, após a interpretação dos dados recebidos, realizar o controle automático da fonte emuladora de tensão. Esta, por sua vez, é controlada através do protocolo de comunicação Modbus. Por fim, os dados relevantes para monitoramento do ensaio de elevação de temperatura são selecionados e enviados, através de uma requisição, para a página Web. Desse modo, pode-se implementar o controle automático da tensão da fonte de alimentação bem como possibilitar ao metrologista o monitoramento remoto do ensaio, auxiliando, assim, na automação laboratorial.

**Palavras-chave:** Ensaio de elevação de temperatura. Circuito eletrônico de instrumentação. Sistema supervisor.

## ABSTRACT

### CONTRIBUTION TO THE AUTOMATION OF THE TRANSFORMERS IMMERSED IN LIQUID INSULATION TEMPERATURE RISE TEST

AUTHOR: Giuliano Francisco Fioreze Cattani

ADVISOR: Rafael Concatto Beltrame

In the year of 2001, the Brazilian Government published the Law of Energy Efficiency, which established regulations for the efficiency in electrical equipment. On that occasion, the Brazilian Government also created the Program of Labeling, which originated the National Label of Energy Conservation (ENCE, in Portuguese). In that context, the National Institute of Metrology, Quality and Technology (INMETRO, in Portuguese), released the ordinance number 510, establishing the specifications for the program of compliance evaluation for distribution transformers immersed in liquid insulation. Among the mandatory tests that should be performed, there is the temperature rise test (i.e. Heat Run), in which is made the measurement in the top of the oil of the transformer, while the equipment is submitted to its load losses. The metrologist can end the test when the variation of temperature is lower than 1 °C during 3 hours, what usually demands about 10 to 12 hours of testing. Therefore, in search to automate the temperature rise test, in this Final Paper is proposed the development of an instrumentation electronic circuit to monitor the electrical and thermal signals originated by the test, including a monitoring system and a web page. The research contains a literature review about the topics covered in the development of that project. Between said topics, there are the electronic instrumentation, communication protocols, web development and the measurement tools inside the Medium Voltage Laboratory of the Smart Grid Institute, which is located in the Federal University of Santa Maria. The project of the printed circuit board (PCB) has two main phases. The first one is about the designing of the instrumentation electronic circuit and its assembly. After test and validation of the designed circuit, the second phase started, that consisted in the software development, formed by the monitoring system and web page. The monitoring system is responsible for receive the data of the PCB and, after the analysis of said data, it does the automatic control of the power supply through the Modbus communication protocol. At last, the metrologist, or the assistant, can, by request, select and send the relevant data of the monitoring of the temperature rise test to the web page. That way, it is possible to implement the automatic control of the power supply, as well the remote supervision of the test by the metrologist, fact that contributes to the automation of the laboratory.

**Keywords:** Temperature rise test. Instrumentation electronic circuit. Monitoring system.

## LISTA DE FIGURAS

Figura 1 – Modelo da ENCE para transformadores novos. ....	16
Figura 2 – Esquema elétrico simplificado da aplicação do circuito eletrônico de instrumentação no ensaio de elevação de temperatura, com o sistema proposto destacado. ....	19
Figura 3 – Configuração básica de um amplificador operacional. ....	23
Figura 4 – Amplificador Não-inversor. ....	23
Figura 5 – Amplificador Inversor. ....	24
Figura 6 – Resposta em frequência de um filtro passa-baixa ideal. ....	25
Figura 7 – Esquema elétrico de um filtro passa-baixa. ....	25
Figura 8 – Demonstração de como ocorre o efeito de <i>boucing</i> . ....	27
Figura 9 – Aplicação de um filtro RC para corrigir efeito de <i>boucing</i> . ....	27
Figura 10 – Modelo simplificado de um instrumento. ....	28
Figura 11 – Curva de resistência dos materiais Níquel e Platina normalizada para a resistência a 25 °C ....	30
Figura 12 – Termorresistência Pt100. ....	30
Figura 13 – Esquema de funcionamento do transmissor de temperatura PR 5333A. ...	31
Figura 14 – Dispositivo Arduino Nano. ....	32
Figura 15 – Equipamento JMAN MONITEK 9451. ....	33
Figura 16 – Diagrama de blocos mostrando os estágios de uma fonte de alimentação. ....	34
Figura 17 – Representação elétrica de um transformador. ....	35
Figura 18 – Retificador de onda completa em ponte. ....	35
Figura 19 – Sinal de saída resultante de um retificador de onda completa em ponte. .	36
Figura 20 – Filtro de entrada utilizando capacitor ....	36
Figura 21 – Funcionamento de capacitor para filtragem de sinal. (a) Sinal retificado. (b) Sinal filtrado ....	36
Figura 22 – Diagrama de blocos de um regulador de tensão. ....	37
Figura 23 – Fonte CA trifásica Supplier® modelo FCATQ 1000-60-40-PFC55450. ....	38
Figura 24 – Representação do modelo OSI e suas camadas. ....	39
Figura 25 – Representação das camadas do modelo ISO/OSI no protocolo Modbus ..	43
Figura 26 – Representação da ADU e PDU dentro de uma mensagem Modbus. ....	43
Figura 27 – Representação de uma mensagem Modbus sem erro ....	45
Figura 28 – Representação de uma mensagem Modbus com erro ....	45
Figura 29 – Demonstração da ADU do Modbus TCP/IP quando comparado com a ADU do Modbus tradicional ....	47
Figura 30 – Demonstração visual da proposta de automação. ....	51
Figura 31 – Diagrama representativo da medição de temperatura do equipamento JMAN	

MONITEK 9451 (a) Sistema de medição padrão. (b) Proposta de medição inserindo um resistor shunt em série na medição padrão. ....	55
Figura 32 – Circuito de medição de temperatura do equipamento JMAN MONITEK 9451. ....	55
Figura 33 – Conexões elétricas para medição de temperatura .....	56
Figura 34 – Conector circular Lumberg. (a) Conector Macho modelo SV30. (b) Conector Fêmea modelo KFV30. ....	56
Figura 35 – Conexões elétricas para medição de temperatura e leitura dos relés do equipamento de medição JMAN MONITEK 9451. ....	57
Figura 36 – Conexões elétricas para acionamento dos relés do equipamento de medição JMAN MONITEK 9451. ....	58
Figura 37 – Apresentação gráfica do condicionamento de sinal do sensor de temperatura .....	59
Figura 38 – Circuito de condicionamento de sinal do sensor de temperatura .....	61
Figura 39 – Circuito de condicionamento de sinal dos relés do equipamento MONITEK 9451. ....	61
Figura 40 – Corrente versus temperatura do sensor de temperatura. ....	63
Figura 41 – Circuito elétrico para a fonte de tensão CC. ....	65
Figura 42 – Construção do frame Modbus para leitura de dados através da função Modbus de código 03. ....	66
Figura 43 – Construção do frame Modbus para escrita de dados através da função Modbus de código 16. ....	66
Figura 44 – <i>Pinout</i> da placa de desenvolvimento Arduino Nano .....	70
Figura 45 – Estrutura de um pacote de dados enviado da PCI para o sistema supervisorio. ....	71
Figura 46 – Visualização do Sistema Supervisorio .....	72
Figura 47 – Página simulada para validar requisições HTTP .....	76
Figura 48 – Configuração da simulação utilizando Postman .....	76
Figura 49 – Demonstração do funcionamento da função <i>filter_var</i> .....	79
Figura 50 – Simulação do circuito de condicionamento do sinal de corrente. ....	86
Figura 51 – Resultados obtidos via simulação nos sinais de entrada do circuito de instrumentação do sensor de temperatura. ....	87
Figura 52 – Resultados obtidos via simulação nos sinais de saída do circuito de instrumentação do sensor de temperatura. ....	88
Figura 53 – Simulação da resposta em frequência do circuito de instrumentação do sensor de temperatura. ....	89
Figura 54 – Esquemático dos circuitos conversão corrente-tensão e condicionamento do sinal de temperatura. ....	91
Figura 55 – Esquemático contendo o circuito projetado para <i>debouncing</i> dos relés. ..	92



Figura 56 – Esquemático contendo as conexões do microcontrolador Arduino Nano e LED para sinalizar operação da PCI. ....	93
Figura 57 – Esquemático contendo o circuito da fonte CC para alimentação do AmpOp. ....	93
Figura 58 – Vista 2D da PCI projetada no <i>software</i> CAD de confecção, contendo todos os circuitos projetados. ....	94
Figura 59 – <i>Top Layer</i> da PCI projetada no <i>software</i> CAD de confecção de placas. ..	95
Figura 60 – Esquemático do experimento realizado em laboratório. ....	96
Figura 61 – Gráfico relacionando medição da entrada analógica A0 do microcontrolador Arduino Nano com a medição de corrente em mA. ....	97
Figura 62 – Gráfico relacionando medição da entrada analógica A0 do microcontrolador Arduino Nano com a medição de corrente em mA. ....	98
Figura 63 – Microcontrolador ESP32. ....	99
Figura 64 – Representação dos componentes utilizados para emular equipamentos presentes no INRIMT. ....	99
Figura 65 – Configuração do circuito de emulação para realização de testes. ....	100

## LISTA DE QUADROS

Quadro 1 – Especificações do transmissor de temperatura PR 5333A .....	31
Quadro 2 – Especificações da placa Arduino Nano .....	32
Quadro 3 – Especificações da fonte emuladora Supplier® modelo FCATQ 1000-60-40- PFC55450. ....	38
Quadro 4 – Modelo de dados Modbus .....	45
Quadro 5 – Acesso aos dados Modbus .....	46
Quadro 6 – Valores medidos pelo Pt100 a partir da variação de temperatura. ....	63
Quadro 7 – Funções da fonte Supplier para utilização do protocolo Modbus .....	67
Quadro 8 – Funções da fonte Supplier para leitura das medidas utilizando o protocolo Modbus .....	68
Quadro 9 – Exemplo de tabela para visualização dos dados medidos .....	75
Quadro 10 – Valores configurados para simulação das requisições .....	77
Quadro 11 – Demonstração dos valores filtrados pela função filter_var do PHP. ....	80
Quadro 12 – Simulação dos sinais medidos pela PCI utilizando fonte de corrente. ...	96

## LISTA DE SÍMBOLOS

<i>ABNT</i>	Associação Brasileira de Normas Técnicas
<i>ADU</i>	Application Data Unit
<i>AMPOP</i>	Amplificador Operacional
<i>API</i>	Application Programming Interfaces
<i>CA</i>	Corrente Alternada
<i>CC</i>	Corrente Contínua
<i>CLR</i>	Common Language Runtime
<i>CLP</i>	Controladores Lógicos Programáveis
<i>DBMS</i>	Database Management System
<i>DDC</i>	Direct Digital Control
<i>ENCE</i>	Etiqueta Nacional de Conservação de Energia
<i>HTTP</i>	HyperText Transfer Protocol
<i>IEEE</i>	Institute of Electrical and Electronics Engineers
<i>INMETRO</i>	Instituto Nacional de Metrologia, Qualidade e Tecnologia
<i>INRI</i>	Instituto de Redes Inteligentes
<i>INRIMT</i>	Laboratório de Ensaios em Média Tensão do INRI
<i>ISO</i>	International Standards Organization
<i>MBAP</i>	Application Protocol Header
<i>PBE</i>	Programa Brasileiro de Etiquetagem
<i>PCI</i>	Placa de Circuito Impresso
<i>PDU</i>	Protocol Data Unit
<i>PHP</i>	Hypertext Preprocessor
<i>RTD</i>	Resistance Temperature Detector
<i>RX</i>	Receiver Line

<i>SCADA</i>	Supervisory Control and Data Acquisition
<i>SGML</i>	Standard Generalized Markup Language
<i>SQL</i>	Structured Query Language
<i>TCP/IP</i>	Transmission Control Protocol and Internet Protocol
<i>TC</i>	Transformador de Corrente
<i>TP</i>	Transformador de Potência
<i>TX</i>	Transmitter Line
<i>UCS</i>	Unidade de Condicionamento de Sinais
<i>UDP</i>	User Datagram Protocol
<i>URI</i>	Uniform Resource Identifier
<i>URL</i>	Uniform Resource Locator
<i>URN</i>	Uniform Resource Name
<i>OSI</i>	Open Systems Interconnection
<i>RM – OSI</i>	Reference Model for Open Systems Interconnection
<i>W3C</i>	World Wide Web Consortium
<i>WWW</i>	World Wide Web
<i>UART</i>	Universal Asynchronous Receiver/Transmitter
<i>ANSI</i>	American National Standard Institute
<i>XML</i>	Extensible Markup Language

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>15</b>
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b> .....	<b>20</b>
2.1	O ENSAIO DE ELEVAÇÃO DE TEMPERATURA: ABNT NBR 5356-2 .....	20
<b>2.1.1</b>	<b>Ensaio de elevação de temperatura para regime permanente pelo método de curto-circuito</b> .....	<b>20</b>
<b>2.1.2</b>	<b>Determinação da temperatura média do enrolamento</b> .....	<b>21</b>
2.2	AMPLIFICADORES OPERACIONAIS .....	22
<b>2.2.1</b>	<b>Amplificador Não-Inversor</b> .....	<b>23</b>
<b>2.2.2</b>	<b>Amplificador Inversor</b> .....	<b>24</b>
2.3	FILTROS ANALÓGICOS .....	24
2.4	CIRCUITO DE DEBOUNCING .....	27
2.5	INSTRUMENTAÇÃO .....	28
<b>2.5.1</b>	<b>Medição de temperatura</b> .....	<b>29</b>
<b>2.5.2</b>	<b>RTD de Platina PT100</b> .....	<b>30</b>
<b>2.5.3</b>	<b>Transmissor de temperatura programável PR 5333A</b> .....	<b>31</b>
2.6	AQUISIÇÃO DE DADOS .....	31
2.7	MEDIDOR DE GRANDEZAS ELÉTRICAS JMAN MONITEK 9451 .....	32
2.8	FONTE CC DE ALIMENTAÇÃO .....	33
<b>2.8.1</b>	<b>Transformador</b> .....	<b>34</b>
<b>2.8.2</b>	<b>Retificador de onda completa e filtro</b> .....	<b>35</b>
<b>2.8.3</b>	<b>Regulador de tensão</b> .....	<b>37</b>
2.9	FONTE CA DE ALIMENTAÇÃO .....	37
2.10	EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL E DAS REDES DE COMUNICAÇÃO DE DADOS .....	39
<b>2.10.1</b>	<b>Modelo de referência ISO/OSI</b> .....	<b>40</b>
2.10.1.1	Camada Física .....	40
2.10.1.2	Camada de Enlace de Dados .....	40
2.10.1.3	Camada de Rede .....	41
2.10.1.4	Camada de Transporte .....	41
2.10.1.5	Camada de Sessão .....	41
2.10.1.6	Camada de Apresentação .....	41
2.10.1.7	Camada de Aplicação .....	42
2.11	PROTOCOLO MODBUS .....	42
<b>2.11.1</b>	<b>Modbus <i>Application Data Unit</i> - ADU</b> .....	<b>43</b>
<b>2.11.2</b>	<b>Transações no protocolo Modbus</b> .....	<b>44</b>
<b>2.11.3</b>	<b>Modelo de dados Modbus e Códigos de Funções</b> .....	<b>45</b>

2.12	MODBUS TCP/IP .....	46
2.13	PADRÃO IEEE 745 .....	48
2.14	DESENVOLVIMENTO <i>WEB</i> .....	48
<b>2.14.1</b>	<b>Páginas estáticas e dinâmicas</b> .....	<b>49</b>
<b>2.14.2</b>	<b>Desenvolvimento de aplicações com PHP</b> .....	<b>49</b>
<b>2.14.3</b>	<b>Banco de Dados</b> .....	<b>50</b>
<b>2.14.4</b>	<b>XML</b> .....	<b>50</b>
2.15	CRIAÇÃO DO SISTEMA SUPERVISÓRIO.....	51
<b>2.15.1</b>	<b>Sistema Supervisório</b> .....	<b>52</b>
<b>2.15.2</b>	<b>Banco de Dados</b> .....	<b>52</b>
<b>2.15.3</b>	<b>Banco de Dados - Aplicativo <i>Web</i></b> .....	<b>53</b>
2.16	CONCLUSÕES PARCIAIS .....	53
<b>3</b>	<b>SOLUÇÃO PROPOSTA</b> .....	<b>54</b>
3.1	DESENVOLVIMENTO DO CIRCUITO DE INSTRUMENTAÇÃO .....	54
<b>3.1.1</b>	<b>Método de medição</b> .....	<b>54</b>
3.1.1.1	Circuito de medição de temperatura – JMAN MONITEK 9451 .....	55
3.1.1.2	Conectores de medição .....	56
3.1.1.3	Medição dos Relés.....	57
<b>3.1.2</b>	<b>Instrumentação</b> .....	<b>58</b>
3.1.2.1	Projeto do circuito de instrumentação do sensor de temperatura.....	58
3.1.2.2	Projeto do resistor shunt – conversão corrente-tensão .....	59
3.1.2.3	Projeto do filtro passa-baixas – atenuação de ruídos de alta frequência.....	59
3.1.2.4	Projeto do circuito amplificador – ganho e proteção .....	60
3.1.2.5	Projeto do circuito de leitura dos relés.....	61
<b>3.1.3</b>	<b>Modelagem dos sensores de temperatura Pt100</b> .....	<b>62</b>
<b>3.1.4</b>	<b>Fonte de tensão CC</b> .....	<b>64</b>
3.1.4.1	Determinação da carga.....	64
3.2	DESENVOLVIMENTO DO SISTEMA SUPERVISÓRIO E PÁGINA <i>WEB</i> .....	65
<b>3.2.1</b>	<b>Fonte Emuladora de Tensão Comunicação <i>Ethernet</i> através de Modbus TCP/IP</b> .....	<b>65</b>
<b>3.2.2</b>	<b>Algoritmo para microcontrolador - Comunicação do circuito de instrumentação com o sistema supervisório</b> .....	<b>69</b>
3.2.2.1	Desenvolvimento da comunicação de dados.....	69
<b>3.2.3</b>	<b>Desenvolvimento do Sistema Supervisório</b> .....	<b>71</b>
<b>3.2.4</b>	<b>Página <i>Web</i> – Comunicação do ensaio de elevação de temperatura com o cliente</b> .....	<b>75</b>
3.2.4.1	Validação de requisições com o Postman .....	75
3.2.4.2	Programação no lado do servidor – Armazenamento das Informações e Requisições <i>Web</i> .....	77

3.2.4.3	Programação no lado do cliente – construindo a <i>interface</i> .....	82
3.3	CONCLUSÕES PARCIAIS .....	84
<b>4</b>	<b>RESULTADOS EXPERIMENTAIS .....</b>	<b>85</b>
4.1	SIMULAÇÃO DO CIRCUITO DE INSTRUMENTAÇÃO DO SENSOR DE TEM- PERATURA.....	85
4.1.1	<b>Validação do condicionamento do sinal de corrente .....</b>	<b>85</b>
4.1.2	<b>Validação do filtro passa-baixas de atenuação dos ruídos e ganho do sistema.....</b>	<b>89</b>
4.2	PROJETO DA PLACA DE CIRCUITO IMPRESSO .....	90
4.2.1	<b>Esquemático dos circuitos conversão corrente-tensão e condiciona- mento do sinal de temperatura .....</b>	<b>90</b>
4.2.2	<b>Esquemático do circuito de <i>Debouncing</i> .....</b>	<b>91</b>
4.2.3	<b>Esquemático do microcontrolador Arduino Nano e LED .....</b>	<b>92</b>
4.2.4	<b>Esquemático do circuito da fonte CC .....</b>	<b>93</b>
4.2.5	<b>Apresentação da PCI finalizada.....</b>	<b>94</b>
4.2.6	<b>Análise da operação da PCI .....</b>	<b>95</b>
4.2.6.1	Simulação das medições na PCI.....	96
4.2.7	<b>Validação do Sistema Supervisório e Página <i>Web</i>.....</b>	<b>98</b>
4.3	CONCLUSÕES PARCIAIS .....	102
<b>5</b>	<b>CONCLUSÃO .....</b>	<b>104</b>
5.1	DIFICULDADES ENFRENTADAS .....	104
5.2	PUBLICAÇÕES VINCULADAS .....	104
5.3	PROPOSTA DE CONTINUIDADE .....	105
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>106</b>

## 1 INTRODUÇÃO

Os transformadores de distribuição são os equipamentos mais numerosos em uma rede de distribuição de energia elétrica. Apesar de apresentarem eficiência energética relativamente alta (normalmente superior a 98%), esses equipamentos são responsáveis por aproximadamente um terço das perdas do sistema elétrico (Agência Nacional de Energia Elétrica, 2023). Desse modo, o projeto de transformadores de elevada eficiência é mandatório para maximizar a eficiência do sistema elétrico (HERNANDEZ; ARJONA, 2010). Além disso, pesquisas sobre perdas em transformadores apontam um grande potencial a ser explorado com o desenvolvimento de novas tecnologias, como a utilização de novos materiais, ou até mesmo a elaboração de novos documentos que incentivem maior eficiência em transformadores de distribuição (CARLEN *et al.*, 2010). Nesse sentido, a pesquisa e o desenvolvimento de equipamentos mais eficientes devem ser prioritários, pois representam uma importante estratégia para reduzir os impactos ambientais, considerando que a eficiência energética contribui diretamente para a redução do consumo de energia elétrica.

Nesse contexto, no ano de 2001 foi publicada a Lei de Eficiência Energética, responsável por estabelecer níveis máximos de consumo específico de energia elétrica, ou mínimos de eficiência energética, visando a alocação eficiente de recursos energéticos e a preservação do meio ambiente (BRASIL, 2001). A fim de atender às atribuições desta lei, foram criados programas que visam a eficiência energética, onde está presente o mecanismo de etiquetagem, desenvolvido no âmbito do Programa Brasileiro de Etiquetagem (PBE). Dentro deste mecanismo de etiquetagem, há a Etiqueta Nacional de Conservação de Energia (ENCE), representada na Figura 1, que tem como objetivo indicar que os transformadores de distribuição em líquido isolante estão em conformidade com as normas vigentes que tratam tanto de eficiência quanto de confiabilidade.



Figura 1 – Modelo da ENCE para transformadores novos.

<b>Energia</b> (Elétrica)	<b>TRANSFORMADOR EM LÍQUIDO ISOLANTE PARA REDE DE DISTRIBUIÇÃO</b>
Fornecedor	ABCDEF
Modelo	XYZ(Logo)
Tipo	ABC 1 2 3
Potência (kVA)	ABC 1 2 3
Classe de Tensão (kV)	ABC 1 2 3
<b>Perdas máximas (tap nominal)</b>	
- Vazio (W)	<input type="text"/>
- Totais (W)	<input type="text"/>
Relação de Transformação	<input type="text"/>
<b>Perdas máximas (tap crítico)</b>	
- Vazio (W)	<input type="text"/>
- Totais (W)	<input type="text"/>
Relação de Transformação	<input type="text"/>
<b>NBI (kV)</b>	<input type="text"/>
 <p>Etiqueta Nacional de Conservação de Energia - ENCE Transformadores de Distribuição em Líquido Isolante Relatório ____ / ____ - XXX</p> <p><b>INMETRO</b></p> <p>IMPORTANTE: FICA PROIBIDA A REMOÇÃO DESTA ETIQUETA DO EQUIPAMENTO.</p>	

Fonte: (INMETRO, 2016)

Com a finalidade de atender critérios de qualidade, foi elaborada a Portaria INMETRO N° 378 (INMETRO, 2010), publicada no Diário Oficial da União no dia 1° de outubro de 2010. Esse documento estabelece especificações para o programa de avaliação da conformidade para transformadores de distribuição em líquido isolante. Publicada pelo Ministério do Desenvolvimento, Indústria e Comércio Exterior e aprovada pelo INMETRO, essa Portaria estabelece a etiquetagem voluntária para transformadores novos e reconicionados.

Publicado no Diário Oficial da União no dia 22 de março de 2013, a Portaria N° 104 (MME, 2013) descreve a regulamentação específica de transformadores de distribuição em líquido isolante. Esse documento alega que somente fabricantes e importadores que atendam ao PBE Transformadores, autorizados pelo INMETRO, estão liberados para a comercialização de transformadores de distribuição no Brasil, a partir de janeiro de 2015. Ou seja, tornou-se compulsório o processo de etiquetagem. A publicação desse documento foi indispensável para o processo de etiquetagem, pois nele estão descritos os indicadores de eficiência energética e procedimentos de ensaios, dados ausentes na Portaria INMETRO N° 378/2010. Além disso, o Programa de Metas definiu que, a partir de 2015, novos níveis de perdas em vazio e totais máximos seriam estabelecidos a cada quatro anos.

Essas regulamentações foram impactantes para o processo de etiquetagem, que até então era voluntário. Com a intenção de facilitar este processo, publicou-se a Portaria INMETRO N° 510 (INMETRO, 2016), que aprova ajustes à Portaria N° 378/2010, alterando

passos importantes do processo de concessão da etiquetagem, flexibilizando o processo de análise do nível de perdas dos transformadores de distribuição em líquido isolante.

O Programa de Metas teve grande impacto na comercialização de transformadores. Segundo (MIRANDA, 2019), vivencia-se o cenário de alta demanda de fabricantes de transformadores *versus* a carência de laboratórios acreditados pelo INMETRO para a realização destes ensaios. Existem no Brasil quatro instalações com este perfil: o Centro de Pesquisa em Energia Elétrica (CEPEL) da Eletrobrás, em Nova Iguaçu, RJ, o Instituto de Energia e Ambiente (IEE) da Universidade de São Paulo (USP), em São Paulo, SP, o Laboratório de Alta Tensão dos Institutos LACTEC, em Curitiba, PR, e o Laboratório de Alta Tensão da Universidade Federal de Itajubá (LAT-EFEI), em Itajubá, MG. Dentre esses laboratórios, há apenas um na região sul do país, e nenhum situado no Rio Grande do Sul. Possuir um laboratório acreditado pela INMETRO na UFSM seria economicamente interessante para a região. Diante dessas circunstâncias, o Laboratório de Ensaios em Média Tensão do Instituto de Redes Inteligentes (INRIMT) está trabalhando no sentido de obter sua acreditação. Com a finalidade de garantir eficácia, confiabilidade e segurança na execução dos ensaios, propôs-se a automação do Laboratório de Ensaios em Média Tensão (FARIAS *et al.*, 2020).

A norma ABNT NBR 5356 (ABNT, 2010a) é responsável por abordar os ensaios realizados em transformadores. Atualmente, o INRIMT possui em seu escopo sete dos ensaios presentes nessa norma, sendo eles:

- Medição de resistência dos enrolamentos;
- Medição de relação de transformação (TTR);
- Medição de impedância de curto-circuito e perdas em carga;
- Medição de perdas a vazio e corrente de excitação;
- Ensaio de tensão suportável a frequência industrial;
- Ensaio de tensão induzida;
- Ensaio de elevação de temperatura;

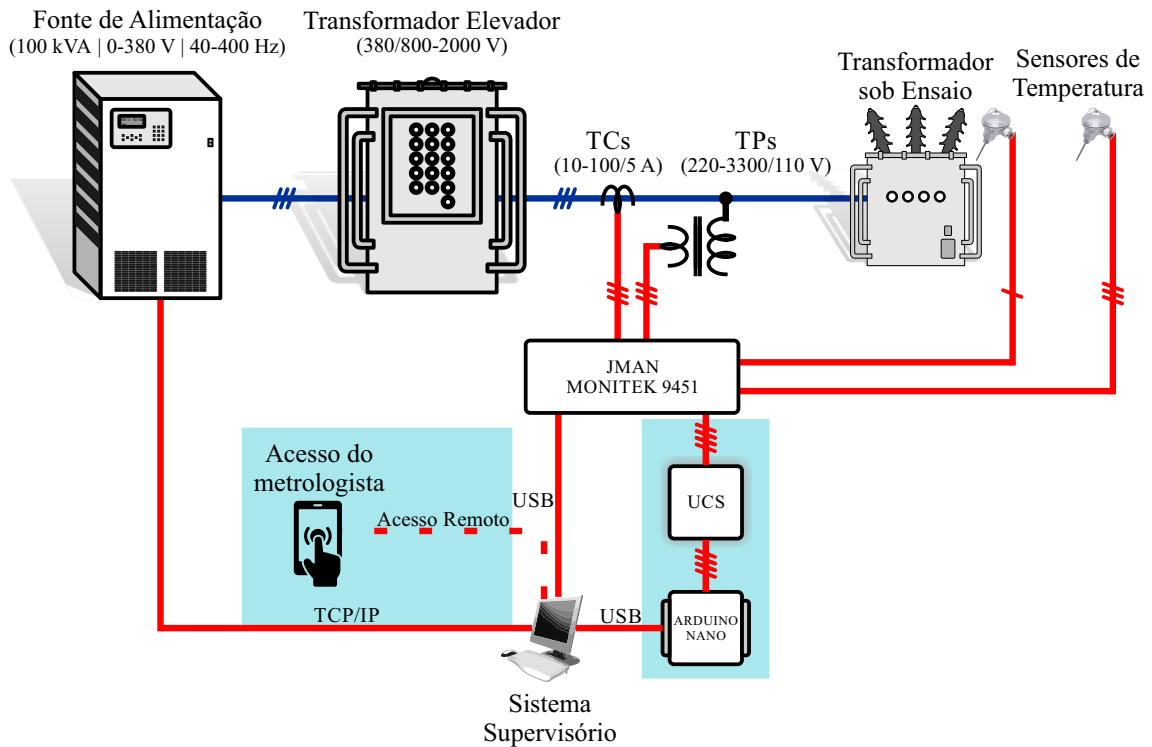
Dentre todos os ensaios realizados em transformadores de distribuição, o ensaio de elevação de temperatura se destaca por sua maior complexidade (ABNT, 2010b). Esse ensaio é dividido em duas fases: medição da temperatura no topo do óleo e medição da temperatura média nos enrolamentos. Para executar a medição da temperatura no topo do óleo do transformador, instala-se um sensor dentro de seu tanque, enquanto o transformador é submetido a uma tensão de ensaio no lado de alta tensão, com a baixa tensão em curto-circuito, que garante que a potência ativa medida seja igual às perdas totais calculadas (soma das perdas nos condutores, perdas no núcleo e perdas adicionais)

corrigidas para a temperatura de referência. Mede-se também a temperatura em pelo menos três pontos no entorno do transformador, cujo objetivo é determinar a temperatura ambiente média para, então, calcular a elevação de temperatura do óleo. O ensaio é finalizado quando o gradiente de temperatura não se altera mais do que  $1^{\circ}\text{C}$  durante um período de três horas. O instrumento de medição empregado pelo INRIMT no ensaio de elevação de temperatura, JMAN MONITEK 9451, é um equipamento digital que permite a medição de tensões, correntes, potências e temperaturas instantaneamente, além de criar um histórico dos dados e servir de plataforma para supervisão do ensaio.

Salienta-se que o ensaio de elevação de temperatura se caracteriza por sua longa duração (usualmente, superior a 10h), pois o transformador sob ensaio demora a entrar em equilíbrio térmico com o ambiente. Além disso, as variáveis são registradas e exibidas, pelo próprio *software* que acompanha o JMAN MONITEK 9451. Assim, o metrologista é obrigado a monitorar constantemente as variáveis elétricas e térmicas do ensaio a fim de garantir a correta excitação do transformador, bem como a inobservância de falhas no equipamento sob ensaio.

A Figura 2 apresenta um esquema elétrico simplificado da execução do ensaio de elevação de temperatura, bem como do circuito proposto destacado em azul. Durante a execução do ensaio, é responsabilidade do metrologista fazer o controle da potência aplicada ao transformador sob ensaio através do ajuste da tensão da fonte de alimentação. Assim, surge uma oportunidade de aplicar conceitos de automação a fim de tornar o ensaio menos desgastante ao metrologista, ou mesmo eliminar a necessidade de sua presença ao longo de toda a duração do ensaio.

Figura 2 – Esquema elétrico simplificado da aplicação do circuito eletrônico de instrumentação no ensaio de elevação de temperatura, com o sistema proposto destacado.



Fonte: Elaborado pelo autor.

O instrumento de medição JMAN MONITEK 9451 disponibiliza dois relés para controle da tensão de ensaio, além de utilizar termorresistências PT100 para leitura das temperaturas (três ambientes e uma no topo do óleo). A presença desses elementos possibilita a construção de um circuito eletrônico de instrumentação que faça a leitura dessas variáveis (estado dos relés e níveis de temperatura), e que essas leituras possam ser adquiridas por um conversor analógico-digital (Arduino Nano) e, por consequência, um sistema supervisor obtenha essas informações. Desse modo, pode-se implementar o controle automático da tensão da fonte de alimentação bem como possibilitar ao metrologista o monitoramento remoto do ensaio.

Portanto, a proposta de automação consiste em aprimorar a integração entre os equipamentos já existentes no laboratório, adicionando funcionalidades que contribuam na execução do ensaio de elevação de temperatura, sem alteração significativa dos procedimentos de ensaio implantados.

Este documento é dividido em diferentes capítulos, cada um com sua devida finalidade para compreensão do projeto proposto. No Capítulo 2 consta a revisão bibliográfica, seção que apresenta a teoria envolvendo o circuito eletrônico de instrumentação. Por sua vez, o Capítulo 3 é constituído pelo desenvolvimento do projeto. Finalmente, no Capítulo 4 são apresentados os resultados obtidos e, por fim, a conclusão no Capítulo 5.

## 2 REVISÃO BIBLIOGRÁFICA

A Revisão Bibliográfica apresenta uma revisão teórica dos principais elementos envolvidos no projeto de automação do ensaio de elevação de temperatura de transformadores de distribuição imersos em líquido isolante. Inicia com a apresentação da norma ABNT NBR 5356-2, que regula o ensaio de elevação de temperatura. Em seguida, são abordados tópicos como amplificadores operacionais, filtros analógicos e circuitos de *debouncing*, que compõem a instrumentação necessária para a obtenção de dados precisos do ensaio.

Também são apresentados equipamentos específicos, como o medidor de grandezas elétricas JMAN Monitek 9451, as fontes de alimentação CC e CA, além de uma breve descrição sobre a evolução da automação industrial e das redes de comunicação de dados. São descritos protocolos de comunicação como o Modbus e o padrão IEEE 745, relevantes para a implementação da automação do ensaio.

Por fim, são abordados tópicos relacionados ao desenvolvimento *Web* e à criação do sistema supervisor, como a aquisição de dados. O capítulo Revisão Bibliográfica, portanto, fornece uma visão geral dos elementos teóricos e práticos que compõem o projeto de automação do ensaio de elevação de temperatura.

### 2.1 O ENSAIO DE ELEVAÇÃO DE TEMPERATURA: ABNT NBR 5356-2

O ensaio de elevação de temperatura é descrito na norma ABNT NBR 5356-2, onde documentam-se as diferentes formas de realizar esse ensaio. O método utilizado pelo IN-RIMT é o de curto-circuito, usualmente utilizado nas fábricas para ensaiar transformadores de distribuição imersos em líquido isolante.

#### 2.1.1 Ensaio de elevação de temperatura para regime permanente pelo método de curto-circuito

A execução deste ensaio não submete o transformador à tensão nominal e corrente nominal simultaneamente, mas sim às perdas totais calculadas. Estas perdas são obtidas a partir da determinação das perdas em carga à temperatura de referência e das perdas a vazio.

Esse método se caracteriza por dividir o ensaio em duas partes principais. A primeira parte tem como objetivo medir a elevação de temperatura no topo do óleo em regime permanente com dissipação das perdas totais. A segunda parte determina a elevação de temperatura média dos enrolamentos à corrente nominal com a elevação de temperatura

do topo do óleo determinada previamente.

A seguir é apresentado o método de ensaio:

- a) Aplicação das perdas totais máximas: conforme explicado anteriormente, essa etapa do ensaio tem como objetivo estabelecer a elevação de temperatura do óleo sobre o meio de resfriamento. Primeiro é determinada a elevação de temperatura do topo do óleo enquanto o transformador é submetido a uma tensão de ensaio tal que a potência ativa medida seja igual às perdas totais do transformador. É importante ressaltar que as perdas totais são obtidas a partir de soma das perdas em vazio e das perdas em carga. Durante a execução do ensaio é feito o monitoramento da temperatura do óleo e também do ambiente. Essa etapa é finalizada quando a taxa de elevação de temperatura no topo do óleo não seja superior a 1 °C durante um período de três horas.
- b) Aplicação da corrente nominal: Essa etapa tem como objetivo determinar a elevação de temperatura média dos enrolamentos sobre óleo, à corrente nominal. Após a primeira etapa deve-se aplicar imediatamente corrente de ensaio igual ao valor nominal para a derivação de maior corrente. O ensaio deve continuar dessa forma por uma hora enquanto é feito o monitoramento das temperaturas do óleo e do meio de resfriamento. Após esse período é feita a medição da resistência dos enrolamentos a partir da desconexão rápida da alimentação e do curto-circuito. Essa medição tem como objetivo determinar a elevação da temperatura dos enrolamentos. Na derivação máxima de corrente, o valor da elevação de temperatura do óleo a ser utilizado deve corresponder às perdas totais calculadas dessa derivação, a fim de determinar a elevação de temperatura dos enrolamentos.

### 2.1.2 Determinação da temperatura média do enrolamento

A temperatura do topo do óleo é obtida ao inserir sensores de temperatura em contato com o óleo no interior do transformador. A temperatura média dos enrolamentos é obtida a partir da medição da resistência dos enrolamentos. A relação entre o valor da resistência  $R_{QUENTE}$  a uma temperatura  $T_{QUENTE}$  (°C) e  $R_{FRIO}$  a  $T_{FRIO}$  é dada pela Equação (2.1). Como resultado, a temperatura média dos enrolamentos é obtida ao isolar  $T_{QUENTE}$ .

$$\frac{R_{QUENTE}}{R_{FRIO}} = \frac{k + T_{QUENTE}}{k + T_{FRIO}} \quad (2.1)$$

Onde:

- a)  $R_{FRIO}$  é a resistência obtida através do ensaio de medição de resistência dos enrolamentos, descrito na norma ABNT NBR 5356-1;

- b)  $R_{QUENTE}$  é a resistência estimada ao final do período de aplicação da corrente nominal (resistência a quente);
- c)  $k$  é a constante de temperatura do material empregado na construção do enrolamento. Alumínio 225 e cobre 235.
- d)  $T_{FRIO}$  é a temperatura onde  $R_{FRIO}$  foi medida.

Por fim, obtém-se a equação (2.2) que define a elevação de temperatura do enrolamento sob medição.

$$\Delta T = T_{QUENTE} - T_{AMB} \quad (2.2)$$

Onde:

- a)  $\Delta T$  é definida como a variação de temperatura do enrolamento sob ensaio;
- b)  $T_{AMB}$  é a temperatura externa ao redor do transformador sob ensaio.

A norma define que o valor de  $R_{QUENTE}$  deve ser determinado no instante imediato antes do corte de alimentação do ensaio. Para que isso seja possível, logo após desligar a fonte alimentação e desconectar o curto-circuito, deve ser conectado um miliohmímetro aos terminais de cada enrolamento a ser medido. Naturalmente, a resistência dos enrolamentos varia com o tempo, pois os enrolamentos devem esfriar. Dessa forma, a fim de obter resultados precisos, essa medida deve ser feita num intervalo de tempo suficiente para permitir a extrapolação para o instante de desenergização.

Para obter um  $\Delta T$  real é necessário aumentar o valor de temperatura do mesmo valor da queda de temperatura do óleo, a partir do valor exato definido na primeira parte do procedimento. Esse valor de temperatura será chamado de  $T_{CORRIG}$ . Como resultado, a equação final para obter a elevação de temperatura nos enrolamentos sob medida é obtida por (2.3):

$$\Delta T_{CORRIG} = \Delta T + T_{CORRIG} \quad (2.3)$$

## 2.2 AMPLIFICADORES OPERACIONAIS

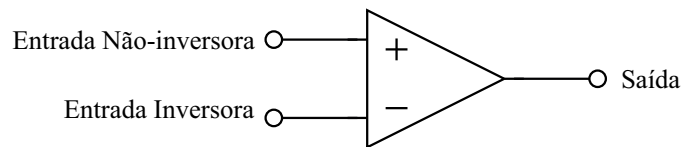
Um amplificador operacional é definido como um amplificador multiestágio com entrada diferencial, cujas características se aproximam das de um amplificador ideal (PER-TENCE, 2015). Conhecer as características ideais dos amplificadores operacionais é fundamental para compreender seu funcionamento, são elas:

- Resistência de entrada infinita;
- Resistência de saída nula;

- Ganho de tensão infinito;
- Resposta de frequência infinita;
- Insensibilidade à temperatura.

São muitas as aplicações dos amplificadores operacionais (AmpOps), como sistemas eletrônicos de controle industrial, instrumentação industrial, instrumentação nuclear, instrumentação médica, computadores analógicos, equipamentos de telecomunicações, equipamentos de áudio, aquisição de dados, dentre outros (PERTENCE, 2015). A Figura 3 mostra a configuração básica de um AmpOp.

Figura 3 – Configuração básica de um amplificador operacional.

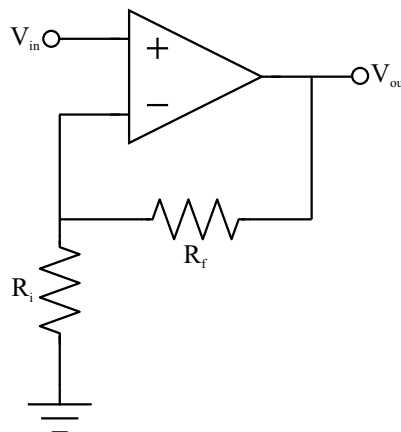


Fonte: Elaborado pelo autor.

### 2.2.1 Amplificador Não-Inversor

A configuração não-inversora é apresentada na Figura 4. De acordo com Boylestad e Nashelsky (1984), essa configuração apresenta alta impedância de entrada.

Figura 4 – Amplificador Não-inversor.



Fonte: Elaborado pelo autor.

O principal objetivo do projetista ao utilizar um amplificador operacional é a determinação do ganho de tensão do circuito. Nessa configuração, o ganho de tensão  $A_v$  é



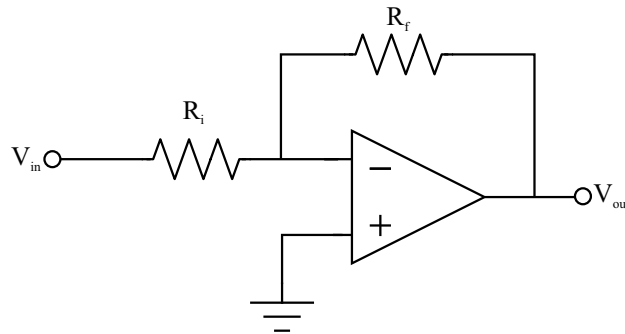
calculado a partir do divisor de tensão formado através do resistor de entrada e o de realimentação (FARIAS, 2019), conforme (2.4):

$$A_v = \frac{V_{out}}{V_{in}} = 1 + \frac{R_f}{R_i} \quad (2.4)$$

## 2.2.2 Amplificador Inversor

A configuração inversora, Figura 5, recebe esse nome porque o sinal estará 180° defasado em relação ao sinal de entrada (PERTENCE, 2015).

Figura 5 – Amplificador Inversor.



Fonte: Elaborado pelo autor.

A saída é obtida pela multiplicação da entrada por um ganho fixo ou constante, definido pelo resistor de entrada  $R_i$  e pelo resistor de realimentação  $R_f$  (BOYLESTAD; NASHELSKY, 1984), conforme (2.5).

$$A_v = \frac{V_{out}}{V_{in}} = -\frac{R_f}{R_i} \quad (2.5)$$

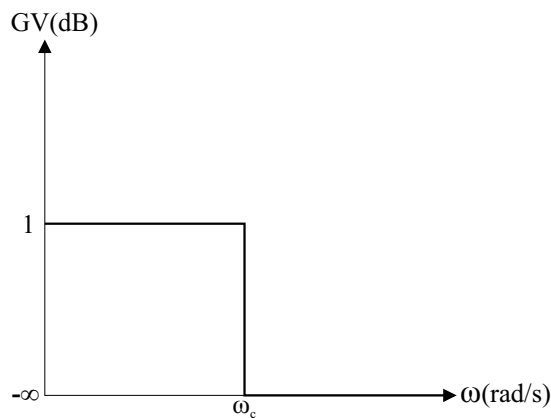
## 2.3 FILTROS ANALÓGICOS

Segundo Pertence (2015), um filtro elétrico é um quadripolo capaz de atenuar determinadas frequências de espectro do sinal de entrada e permitir a passagem das demais. Existem diferentes formas de implementação de filtros, podendo ser ativos, passivos ou digitais. Existem quatro categorias de filtros mais comumente conhecidos, sendo eles: passa-altas (PA), passa-baixas (PB), passa-faixa (PF) e rejeita-faixa (RF) (FARIAS, 2019).

Por questão de objetividade, nessa seção será abordado o filtro passivo do tipo passa-baixas, o qual está empregado no trabalho.

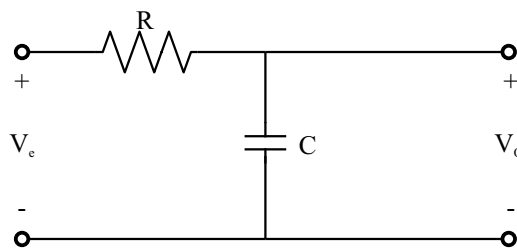
Quando um projeto de filtro passivo é executado, a frequência de corte ( $f_c$ ) é a variável responsável por limitar a frequência passante. No caso do filtro passa-baixas, só é permitido a passagem de frequências abaixo da  $f_c$ , sendo as frequências superiores atenuadas (PERTENCE, 2015). Seu comportamento ideal é representado na Figura 6, e seu esquema elétrico na Figura 7.

Figura 6 – Resposta em frequência de um filtro passa-baixa ideal.



Fonte: Adaptado de Mussoi (2004).

Figura 7 – Esquema elétrico de um filtro passa-baixa.



Fonte: Adaptado de Mussoi (2004).

Para sinais de frequências abaixo da frequência do corte do filtro, o ganho é unitário. Ou seja, o módulo do sinal de entrada é igual ao de saída. Enquanto isso, para frequências acima da frequência de corte o ganho é zero, ou seja, o módulo do sinal de saída é atenuado até zero (MUSSOI, 2004).

A tensão de saída  $V_o$  é obtida a partir da Equação (2.6):

$$V_o = \frac{X_c}{R + X_c} \cdot V_e \quad (2.6)$$

$$V_o = \frac{\frac{1}{j\omega C}}{R + \frac{1}{j\omega C}} \cdot V_e \quad (2.7)$$

Ao fatorar a Equação (2.7) acima, obtém-se a função de transferência de um filtro passa-baixas segundo a Equação (2.8).

$$H(\omega) = \frac{1}{1 + j\omega RC} = \frac{V_o}{V_e} \quad (2.8)$$

Deve-se observar que a função de transferência do filtro passa-baixas é um número complexo. O ganho de tensão é obtido pelo módulo dessa função na forma polar, enquanto que a fase corresponde ao ângulo dessa mesma função de transferência. Como resultado, o ganho de tensão é representado pela Equação (2.9), e a Equação (2.10) representa a fase.

$$G = \frac{1}{\sqrt{1 + (\omega RC)^2}} \quad (2.9)$$

$$\alpha = -\arctan(\omega RC) \quad (2.10)$$

A frequência de corte é a frequência em que a tensão de saída é aproximadamente 70,7% da tensão de entrada, ou seja, a frequência que provoca um ganho de -3 dB (MUS-SOI, 2004). Obtém-se a frequência de corte pela Equação (2.11).

$$f_c = \frac{1}{RC} \quad (2.11)$$

Aplicando (2.11) em (2.10), conforme (2.12), obtém-se a frequência de corte (2.13).

$$\alpha = -\arctan\left(\frac{1}{RC} \cdot RC\right) \quad (2.12)$$

$$\alpha = -45^\circ \quad (2.13)$$

Ter conhecimento do comportamento das curvas de resposta da frequência do filtro Passa-Baixa RC é essencial para realização do seu projeto. Assim, para ganho e fase, respectivamente:

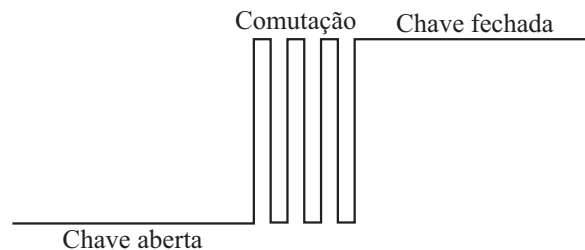
- $\omega = 0 \Rightarrow G = 1$  e  $\alpha = 0$ ;
- $\omega = \omega_c \Rightarrow G = 0,707$  e  $\alpha = -45^\circ$ ;
- $\omega \rightarrow \infty \Rightarrow G = 0$  e  $\alpha = -90^\circ$ .

## 2.4 CIRCUITO DE DEBOUNCING

Relés de acionamento e controle são comumente utilizados em circuitos de automação. Esses dispositivos são chaves mecânicas ON/OFF.

Quando uma chave mecânica fecha seu contato, por alguns instantes ocorre uma oscilação mecânica que pode ser interpretada pelo circuito como várias comutações, ao invés de uma só. A Figura 8 representa como essa oscilação pode comprometer a interpretação de um circuito. Esse fenômeno é comumente chamado de *bouncing* ou repique.

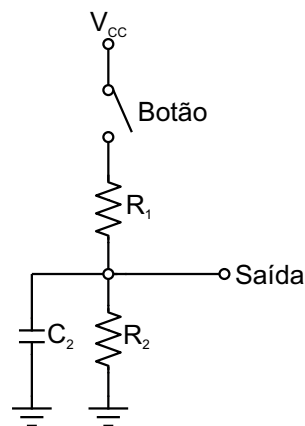
Figura 8 – Demonstração de como ocorre o efeito de *bouncing*.



Fonte: Adaptado de Christoffersen (2015)

Esse fenômeno é corrigido através da implementação de um circuito *debouncing*. Há diferentes maneiras de implementar esse tipo de circuito e a mais comum é filtrar as oscilações através da associação de resistores e capacitores. Ou seja, um filtro passa-baixas. Uma aplicação de um circuito *debouncing* é mostrado na Figura 9.

Figura 9 – Aplicação de um filtro RC para corrigir efeito de *bouncing*.



Fonte: Elaborado pelo autor

Os resistores  $R_1$  e  $R_2$  e o capacitor  $C_2$  são utilizados para definir as constantes de tempo em que o filtro opera. Com a Equação (2.11) obtém-se as constantes de tempo  $\tau_1$  e  $\tau_2$  para operação do circuito:

$$\tau_1 = R_1 \cdot C_2 \quad (2.14)$$

$$\tau_2 = R_2 \cdot C_2 \quad (2.15)$$

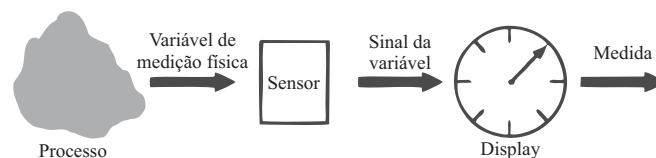
É importante ressaltar que  $R_2 \gg R_1$  é fundamental para operação adequada do sistema.

## 2.5 INSTRUMENTAÇÃO

A instrumentação pode ser definida como a ciência que estuda, desenvolve e aplica instrumentos de medição e controle de processos. É empregada desde processos simples, como o controle de temperatura em uma residência, até no controle de processos críticos, como reatores nucleares (FRANCHI, 2018).

De uma forma simplificada, um sensor é um dispositivo que transforma uma variável física de interesse (pressão, temperatura, etc.) em um formato passível de medição por um instrumento. A Figura 10 descreve o modelo simplificado de um instrumento típico (FRANCHI, 2018).

Figura 10 – Modelo simplificado de um instrumento.



Fonte: Adaptado de Franchi (2018)

É possível representar o funcionamento de um sistema de instrumentação através de três elementos fundamentais: sensores, processamento de sinais e representação dos dados (BOLTON, 2015). Os sensores estão diretamente conectados com o sistema a ser medido e se responsabilizam por viabilizar sua medição. Por exemplo, quando há a necessidade monitorar a temperatura de um processo, existem diferentes formas de realizar essa medição, como através da utilização de termopares ou termorresistências. O termopar fornece em sua saída um pequeno sinal de tensão, enquanto a termorresistência resulta em uma variação de resistência ôhmica. O processamento do sinal recebe a medição do sensor e condiciona o mesmo para controle do processo, por exemplo. No caso da termorresistência, é necessário transformar essa variação de resistência em uma grandeza elétrica, como tensão, para poder ser interpretada por um microcontrolador, por exemplo.

### 2.5.1 Medição de temperatura

Apesar do grande número de medições existentes na indústria, pode-se considerar a temperatura como uma das variáveis mais medidas em processos industriais, pois ela é uma variável crítica para indústrias químicas, petroquímica, de polímeros, de plásticos, alimentícia, entre outras (FRANCHI, 2018).

A expansão ou contração de sólidos, líquidos ou gases, a mudança na resistência elétrica de condutores e semicondutores, diferença de potencial termoeletrônicos e a mudança na corrente através da junção de diodos semicondutores e transistores são exemplos de propriedades que mudam quando a temperatura muda e podem ser usados como base para sensores de temperatura (BOLTON, 2015). O presente projeto utiliza a termorresistência para medição de temperatura.

Visto que a resistência elétrica de certos materiais varia com a temperatura, esses materiais podem ser utilizados na confecção de sensores para medir tal grandeza (AGUIRRE, 2013). Esse comportamento pode ser representado a partir da Equação (2.16).

$$R_T = R_0(1 + \alpha \cdot T) \quad (2.16)$$

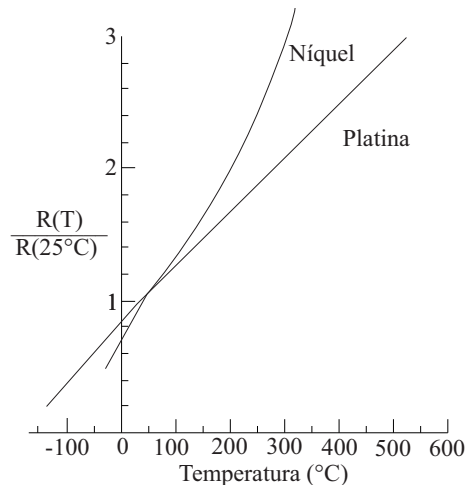
Onde:

- a)  $R_T$  é a resistência quando a temperatura é igual a T;
- b)  $R_0$  é a resistência a 0°C;
- c)  $\alpha$  é a constante do metal.

Quanto aos sistemas de medição de temperatura que utilizam termorresistências, deve-se notar que, geralmente, não se deseja medir a resistência. O mais comum é medir alguma grandeza diretamente relacionada a ela (AGUIRRE, 2013), como tensão ou corrente elétrica.

À medida que se movem através de um material, os elétrons colidem com átomos estacionários ou moléculas do material. Quando uma energia térmica está presente no material, ela faz com que os átomos vibrem, e os elétrons de condução tendem a colidir mais com os átomos em vibração. Isso impede o movimento dos elétrons e absorve um pouco de sua energia, isto é, o material exibe uma maior resistência à passagem de corrente elétrica. Assim, a resistência elétrica é função da vibração dos átomos e, dessa forma, da temperatura. À medida que a temperatura aumenta, os átomos vibram com amplitude e frequência maiores, o que causa mais colisões com elétrons e impede o escoamento da corrente elétrica. A partir disso, pode-se afirmar que a resistência elétrica aumenta com a temperatura (FRANCHI, 2018). A Figura 11 demonstra o efeito de aumento da resistência com o aumento da temperatura para a platina e o níquel.

Figura 11 – Curva de resistência dos materiais Níquel e Platina normalizada para a resistência a 25 °C



Fonte: Adaptado de Franchi (2018)

Ou seja, esse sensor, também chamado de RTD (*Resistance Temperatura Detector*), opera baseado no fato da resistência do material variar de acordo com a temperatura.

### 2.5.2 RTD de Platina PT100

O termorresistor de platina PT100, Figura 12, é um sensor muito aplicado na indústria devido à sua estabilidade, faixa de medição e precisão. Recebe esse nome pois apresenta uma resistência de 100  $\Omega$  em 0°C.

Figura 12 – Termorresistência Pt100.

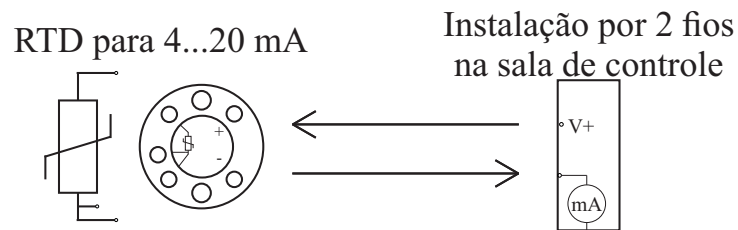


Fonte: Adaptado de Dewesoft (2021)

### 2.5.3 Transmissor de temperatura programável PR 5333A

Transmissores de temperatura são dispositivos que convertem sinais de vários tipos de sensores, como termorresistência e termopares, em sinais de saída padronizados. O Transmissor de Temperatura Programável PR5333A é utilizado para converter sinais do sensor PT100 e fornecer em sua saída um sinal de corrente entre 4 a 20mA. A Figura 13 representa o funcionamento desse equipamento, enquanto o Quadro 1 mostra suas características.

Figura 13 – Esquema de funcionamento do transmissor de temperatura PR 5333A.



Fonte: Adaptado de PR Eletronics (s.d.)

Quadro 1 – Especificações do transmissor de temperatura PR 5333A

Descrição	PR 5333A
Range de sinal	[4 a 20] mA
Range de sinal mínimo	16 mA
Tempo de atualização	135 ms
Resistência de carga	$\leq (V_{\text{alimentação}} - 8,0) / 0,023 \Omega$
Estabilidade de carga	$< \pm 0,01\%$ de span / 100 $\Omega$
Tensão de alimentação	[8,0 a 35] V
Sinal / Ruído	Min. 60 dB

Fonte: (PR Eletronics, s.d.)

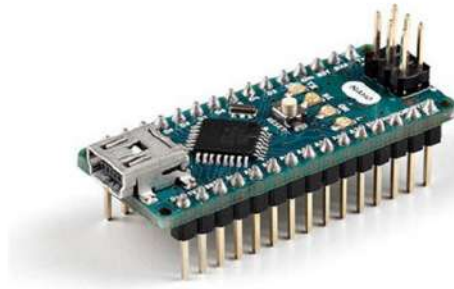
A respeito das condições ambientais, esse transmissor de temperatura opera entre -45 °C e 85 °C.

## 2.6 AQUISIÇÃO DE DADOS

O Arduino Nano, Figura 14, emprega um microcontrolador ATmega328. Suas especificações estão presentes no Quadro 2.



Figura 14 – Dispositivo Arduino Nano.



Fonte: (Arduino, s.d.)

Quadro 2 – Especificações da placa Arduino Nano

Descrição	Valor
Microcontrolador	ATmega328
Tensão de alimentação	5 V
Frequência de <i>clock</i>	16 MHz
Entradas analógicas	8
Corrente CC por entrada	40 mA
Tensão de alimentação	7-12 V
Entradas I/O digitais	22
Saídas PWM	6
Corrente consumida	19 mA

Um conversor analógico-digital (ADC) é um circuito elétrico que converte um sinal elétrico de tensão analógico em um sinal digital. Essa conversão pode ser realizada diretamente pela maioria dos microcontroladores, pois possuem conversores internos (JR.; SILVA, 2015). Por sua vez, o Arduino Nano possui um conversor ADC com resolução de 10 bits com *range* de 0 a 5V. O ADC será empregado neste projeto, em conjunto com o circuito de condicionamento de sinais, para realizar a leitura do sensor de temperatura.

## 2.7 MEDIDOR DE GRANDEZAS ELÉTRICAS JMAN MONITEK 9451

O instrumento de medição JMAN MONITEK9451<sup>®</sup> é um aparelho digital que permite a medição de tensão, corrente e temperatura instantâneos. A Figura 15 mostra o equipamento em questão. Através destas grandezas, o *software* TRANS4, vinculado a

este equipamento, calcula os valores desejados conforme o ensaio executado. Esse equipamento foi desenvolvido para a realização dos ensaios de perdas a vazio e corrente de excitação, impedância de curto-circuito e perdas em carga e ensaio de elevação de temperatura (MIRANDA, 2019) de transformadores elétricos.

Figura 15 – Equipamento JMAN MONITEK 9451.



Fonte: Foto do autor.

Como esse equipamento tem como objetivo realizar o ensaio de elevação de temperatura, sua construção apresenta alguns circuitos que possibilitam a automação desse ensaio. Segundo Vicente (2006), é implementado nesse equipamento um circuito eletrônico que permite ligar e desligar dois relés, do tipo normalmente aberto, que atuam no sentido de comandar um varistor de tensão para aumentar ou diminuir a tensão da fonte primária de energia, mantendo constante o valor da potência ativa de perdas aplicada ao transformador. Ainda sobre esse ensaio, para determinar a estabilização de temperatura são utilizados quatro sensores de temperatura do tipo PT100, um instalado no topo do óleo e três medindo a temperatura do ar ambiente, conforme descrito na norma ABNT NBR 5356-2 (VICENTE, 2006).

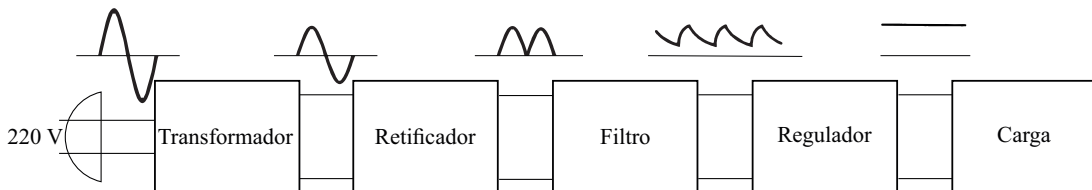
## 2.8 FONTE CC DE ALIMENTAÇÃO

Fontes de alimentação são circuitos com a finalidade de alimentar circuitos eletroeletrônicos com níveis de tensão regulados. Esses circuitos são compostos por transformador, filtros, retificador e reguladores de tensão.

De acordo com Boylestad e Nashelsky (1984), a tensão CA, alimenta um transformador cuja função é reduzi-la para o nível de tensão CC desejado na saída. Um retificador

a diodo fornece, então, uma tensão retificada de onda completa, sendo inicialmente filtrada por um filtro básico a capacitor para produzir uma tensão CC. Essa tensão CC resultante usualmente tem alguma ondulação (*ripple*). Um circuito regulador pode utilizar essa entrada CC para produzir uma tensão CC que não apenas tem muito menos ondulação, mas também mantém constante o valor na saída, mesmo para variações na tensão CC de entrada ou mudanças no valor da carga conectada à tensão CC de saída. A Figura 16 mostra os estágios de uma fonte de alimentação:

Figura 16 – Diagrama de blocos mostrando os estágios de uma fonte de alimentação.



Fonte: Adaptado de Boylestad e Nashelsky (1984)

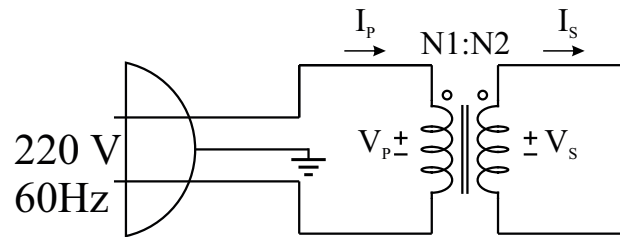
### 2.8.1 Transformador

No Brasil, dependendo da concessionária de energia elétrica, a tensão entregue na residência pode ser de 127 V ou 220 V. Um transformador é constituído basicamente por um núcleo e duas ou mais bobinas. O núcleo do transformador é feito de material ferromagnético, enquanto as bobinas são compostas por N espiras, isoladas eletricamente entre si.

A passagem de uma corrente elétrica alternada em qualquer uma das bobinas de um transformador resulta em fluxo magnético variável em seu núcleo, causando o surgimento de uma força eletromotriz induzida na outra bobina. Esse efeito foi descoberto por Michael Faraday, dando origem à lei de indução de Faraday, que descreve o efeito da tensão elétrica induzida nas bobinas dos transformadores em razão das mudanças no fluxo de campo magnético.

Transformadores podem ser utilizados de diversas formas, como elevador, rebaixador ou isolador de tensão. Quando utilizado como rebaixador, o transformador abaixa a tensão da linha para um nível seguro, mais adequado para o uso com diodos, transistores e outros dispositivos semicondutores (MALVINO, 2015). A representação elétrica de um transformador é mostrada na Figura 17.

Figura 17 – Representação elétrica de um transformador.



Fonte: Elaborado pelo autor.

A relação de espiras de um transformador com sua corrente e tensão é dada pela equação (2.17).

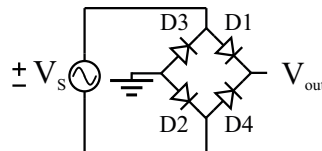
$$\frac{N_1}{N_2} = \frac{V_P}{V_S} = \frac{I_P}{I_S} \quad (2.17)$$

### 2.8.2 Retificador de onda completa e filtro

A fonte CA produz uma tensão senoidal. Para construir uma fonte CC, deve-se transformar esse sinal em contínuo. Para isso, utiliza-se arranjos de diodos. É possível arranjar diodos para compor retificadores de meia onda, onda completa com *tap* central e onda completa em ponte. Nessa seção, será apresentado o retificador de onda completa em ponte.

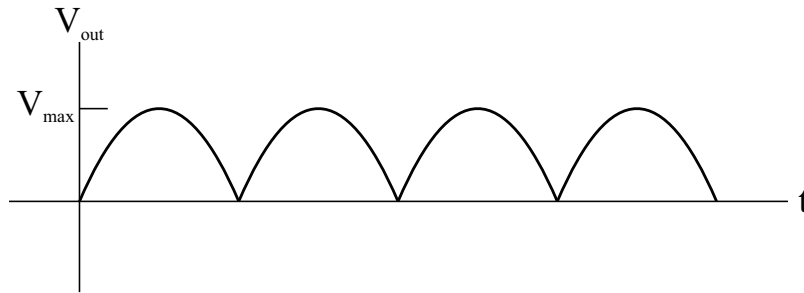
A Figura 18 mostra diodos arranjos como retificador de onda completa em ponte. Os diodos D1 e D2 conduzem durante o semiciclo positivo, e D3 e D4 conduzem durante o semiciclo negativo. Para que a tensão CC de saída da fonte seja mais estável, um circuito de filtro torna-se necessário. Como resultado, a corrente retificada na carga circula durante os dois semiciclos. O arranjo de diodos é mostrado na Figura 18, e a onda retificada na Figura 19.

Figura 18 – Retificador de onda completa em ponte.



Fonte: Elaborado pelo autor.

Figura 19 – Sinal de saída resultante de um retificador de onda completa em ponte.



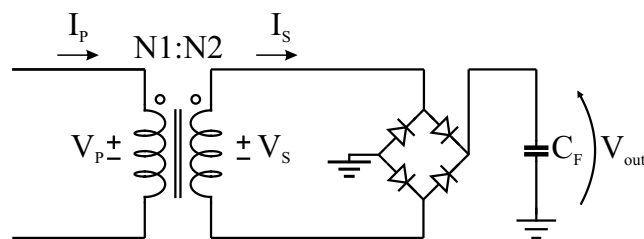
Fonte: Elaborado pelo autor.

Nesse caso,  $V_{max}$  seria a tensão secundária do transformador, obtida através de (2.18), considerando as quedas geradas pelos diodos da ponte retificadora, conforme Figura 18.

$$V_{max} = \max(V_S) - 1,4 \quad (2.18)$$

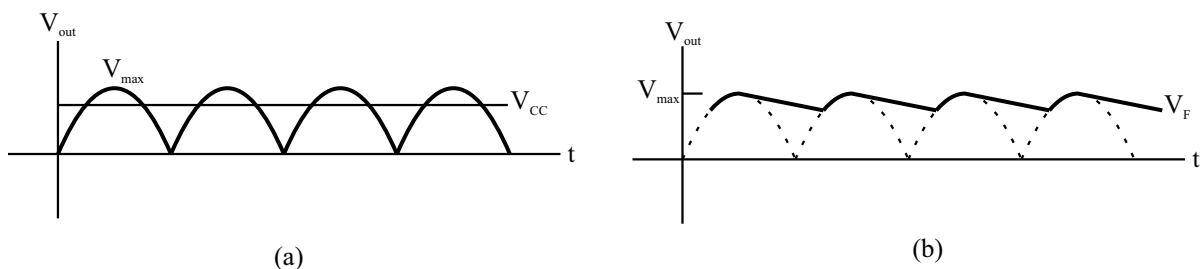
O filtro de entrada com capacitor produz uma tensão CC de saída igual ao valor de pico da tensão retificada. Este tipo de filtro é muito usado nas fontes de alimentação (MALVINO, 2015). A Figura 20 apresenta a configuração de um filtro utilizando capacitor, enquanto a Figura 21 mostra o resultado obtido com a saída filtrada.

Figura 20 – Filtro de entrada utilizando capacitor



Fonte: Elaborado pelo autor.

Figura 21 – Funcionamento de capacitor para filtragem de sinal. (a) Sinal retificado. (b) Sinal filtrado



Fonte: Elaborado pelo autor.

Apesar de a tensão retificada não ser uma tensão filtrada, ela contém uma componente CC e uma componente de ondulação (*ripple*) (BOYLESTAD; NASHELSKY, 1984).

Um sinal retificado de onda completa tem uma componente CC obtido pela Equação (2.19):

$$V_{CC} = \frac{2}{\pi} \cdot V_{max} \quad (2.19)$$

A forma da ondulação do filtro com capacitor é dado por (2.20).

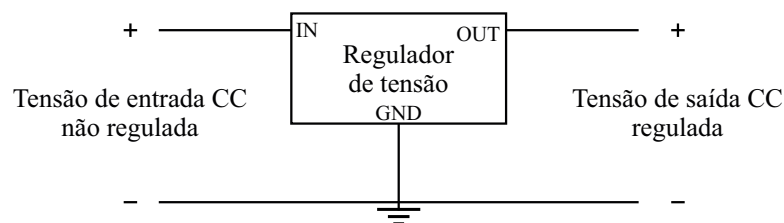
$$V_F = \frac{I_{CC}}{f \cdot C_F} \quad (2.20)$$

### 2.8.3 Regulador de tensão

Reguladores de tensão compreendem uma classe de circuitos integrados largamente utilizados. Os CIs reguladores contêm os circuitos de fonte de referência, o amplificador comparador, o dispositivo de controle e a proteção contra sobrecarga, tudo em uma única pastilha (BOYLESTAD; NASHELKY, 1984). Reguladores de tensão são amplamente utilizados para regular um sinal de tensão CC para um valor fixo.

Quando um regulador de tensão é utilizado, deve-se atentar em algumas particularidades. A tensão de desligamento de um CI regulador é definida como a mínima tensão de margem de operação necessária para a regulação. Por exemplo, CIs reguladores padrão têm uma tensão de desligamento de 2 V a 3 V. Isso significa que a tensão de entrada tem que ser pelo menos de 2 V a 3 V maior que a tensão de saída regulada para o *chip* regular conforme especificação (MALVINO, 2015). A Figura 22 mostra a representação em blocos de um regulador de tensão.

Figura 22 – Diagrama de blocos de um regulador de tensão.



Fonte: Adaptado de Boylestad e Nashelsky (1984).

## 2.9 FONTE CA DE ALIMENTAÇÃO

O ensaio de elevação de temperatura é alimentado por uma fonte emuladora de rede CA trifásica. O equipamento, desenvolvido pela Supplier<sup>®</sup>, é o modelo FCATQ 1000-60-40-PFC55450, Figura 23, e suas especificações estão no Quadro 3.

Figura 23 – Fonte CA trifásica Supplier® modelo FCATQ 1000-60-40-PFC55450.



Fonte: (SUPPLIER, s.d.a)

Quadro 3 – Especificações da fonte emuladora Supplier® modelo FCATQ 1000-60-40-PFC55450.

Descrição	Valor
Tensão nominal de alimentação	380 V $\pm$ 10%
Corrente nominal de entrada	179 A
Frequência de alimentação	50/60 Hz
Fator de potência sobre carga nominal	1,00
Potência nominal de saída	100 kVA
Tensão de saída	0 a 600 V (tensão de linha)
Corrente nominal de saída	152 A @ 220 V (fase)
	96,6 A @ 345 V (fase)
Proteção contra sobrecorrente de saída	$\leq$ 215 A (pico) instantâneo
Frequência de saída	40 a 400 Hz

Fonte: (SUPPLIER, s.d.a)

Conforme descrito no manual do equipamento (SUPPLIER, s.d.), a fonte é equipada com uma porta de comunicação *Ethernet*, através do qual podem ser operada e monitorada remotamente. Através de uma interface virtual em plataforma PC, poder-se-á alterar os valores (*preset*) de diversos parâmetros da fonte, obter os valores atualmente ajustados, as condições de operação atuais do equipamento (*status*), e também realizar a leitura das medições realizadas pela fonte. O protocolo utilizado para leitura e envio de dados é o MODBUS TCP/IP, onde todas as transições são realizadas via porta 502.

## 2.10 EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL E DAS REDES DE COMUNICAÇÃO DE DADOS

A utilização de computadores digitais como controladores digitais aconteceu pela primeira vez no começo de 1960. O termo *direct digital control* (DDC) foi utilizado para enfatizar que determinado processo era controlado diretamente por um computador. Nesse mesmo período surgiram os Controladores Lógicos Programáveis (CLPs), substituindo as aplicações convencionais baseadas na atuação de relés. Como consequência, novas tecnologias foram desenvolvidas para máquinas de produção e processos discretos. Com o uso mais difundido de computadores digitais e das tecnologias associadas, as redes de comunicação industrial passaram a ser desenvolvidas ou convertidas para transmissão digital. Nesse mesmo período surgiram as primeiras redes de comunicação, quando se conectaram os computadores utilizados nos processos de automação (DJIEV, s.d.).

Em meados de 1970, o primeiro computador para controle de sistemas foi anunciado por Honeywell como um sistema de controle hierárquico com um grande número de microcontroladores. Esse novo sistema espalhou-se pelas diversas indústrias de automação, ainda sem um padrão de utilização. Ou seja, cada gerenciamento empresarial desenvolvia sua própria forma de implementação. Conforme os sistemas de automação industrial foram se desenvolvendo e a quantidade de dispositivos de automação cresceu, tornou-se muito importante para a automação industrial fornecer padrões que tornem possível interconectar dispositivos de automação através de um padrão. Então, diferentes órgãos internacionais começaram a buscar uma padronização para redes. Conhecidos como *Open Systems Interconnection* (OSI), estes modelos permitiram que qualquer par de dispositivos de automação se comunique de forma confiável, independentemente do fabricante (DJIEV, s.d.). A Figura 24 apresenta, resumidamente, as camadas desse modelo, seu nome e sua função.

Figura 24 – Representação do modelo OSI e suas camadas.

Camada	Modelo ISO/OSI	Função
1	Aplicação	Interface com aplicativos
2	Apresentação	Formatos/Criptografia
3	Sessão	Controle de sessão entre aplicativos
4	Transporte	Conexão entre <i>hosts</i> /Portas
5	Rede	Endereço lógico/Roteadores
6	Enlace de dados	Endereço físico/Pontes e <i>switches</i>
7	Física	<i>Hardware</i> /Sinal elétrico/ <i>bits</i>

Fonte: Elaborado pelo autor.

Portanto, redes de Comunicação Industrial tem um papel importante para as indústrias. As redes são fundamentais para que uma indústria, seja ela de pequeno, médio ou grande porte, consiga ter interoperabilidade entre todos os seus setores, desde o chão de



fábrica até a diretoria da empresa, oferecendo informações precisas, em tempo real e com segurança na integração entre redes corporativas e redes de automação industrial, que serão fundamentais para o gerenciamento e controle do processo produtivo (NOGUEIRA, 2019).

### **2.10.1 Modelo de referência ISO/OSI**

À medida que cresceu a conexão de computadores industriais através de redes de comunicação, surgiu a necessidade de criar uma padronização. Iniciou-se, então, através da ISO (*International Standards Organization*), uma reunião de esforços no sentido de definir uma proposta de arquitetura normalizada para as redes de comunicação. Dada a grande diversidade dos equipamentos e das soluções existentes no que diz respeito à comunicação, o resultado deste trabalho foi de fato a padronização de um modelo (denominado Modelo de Referência) sobre o qual deveriam ser baseadas as arquiteturas de redes de comunicação, de forma a permitir a interconexão de equipamentos heterogêneos, tornando transparente ao usuário a forma como esta interconexão fosse implementada (STEMMER, 2001).

Baseada nesta filosofia, a proposta, definida numa série de documentos produzidos por aquela organização, foi denominada Modelo de Referência para a Interconexão de Sistemas Abertos ou RM-OSI (*Reference Model for Open Systems Interconnection*) (STEMMER, 2001).

#### **2.10.1.1 Camada Física**

De acordo com Tanenbaum (2011), a camada física recebe os quadros pela camada de transmissão de dados e os transforma em sinais compatíveis com o meio no qual os dados devem ser transmitidos. É nesta camada que se dá, por exemplo, a transferência de *bytes*.

#### **2.10.1.2 Camada de Enlace de Dados**

A camada de enlace de dados tem como objetivo transformar um canal de transmissão normal em uma linha que pareça livre de erros de transmissão. Para fazer isso, a camada de enlace mascara os erros reais, de modo que a camada de rede não os veja. Isso é executado fazendo com que o transmissor divida os dados de entrada em quadros de dados e transmita os quadros sequencialmente. Se o serviço for confiável, o receptor

confirmará a recepção correta de cada quadro, enviando de volta um quadro de confirmação (TANENBAUM, 2011).

#### 2.10.1.3 Camada de Rede

A camada de rede controla a operação da sub-rede. Uma questão fundamental de projeto é determinar a maneira como os pacotes são roteados da origem até o destino. As rotas podem se basear em tabelas estáticas, "amarradas" à rede e raramente alteradas, ou frequentemente podem ser atualizadas de forma automática, para evitar componentes defeituosos. Elas também podem ser determinadas no início de cada conversaç o; por exemplo, uma sess o de terminal, como um *login* em uma m quina remota (TANENBAUM, 2011).

#### 2.10.1.4 Camada de Transporte

Essa camada gerencia a transfer ncia dos dados e das informa es, executado pelo protocolo utilizado. A camada de transporte tamb m determina que tipo de servi o deve ser fornecido   camada de sess o e, por fim, aos usu rios da rede (TANENBAUM, 2011).

#### 2.10.1.5 Camada de Sess o

Tanenbaum (2011) diz que a camada de sess o permite que os usu rios em diferentes m quinas estabeleam sess es de comunica o entre eles. Uma sess o oferece diversos servi os, inclusive o controle de di logo (mantendo o controle de quem deve transmitir em cada momento), o gerenciamento de *tokens* (impedindo que duas partes tentem executar a mesma opera o cr tica em simult neo) e a sincroniza o (realizando a verifica o peri dica de longas transmiss es para permitir que elas continuem a partir do ponto em que estavam ao ocorrer uma falha e a subsequente recupera o).

#### 2.10.1.6 Camada de Apresenta o

Diferente das camadas mais baixas, que se preocupam principalmente com a movimentaa o de *bits*, a camada de apresenta o est  relacionada   sintaxe e   sem ntica das informa es transmitidas. A camada de apresenta o gerencia essas estruturas de

dados abstratas e permite a definição e o intercâmbio de estruturas de dados de nível mais alto (por exemplo, registros bancários) (TANENBAUM, 2011).

#### 2.10.1.7 Camada de Aplicação

A camada de aplicação contém uma série de protocolos comumente necessários para os usuários. Um protocolo de aplicação amplamente utilizado é o HTTP (*HyperText Transfer Protocol*), que constitui a base da *World Wide Web*. Quando um navegador deseja uma página *Web*, ele envia o nome da página desejada ao servidor que hospeda a página, utilizando o HTTP. O servidor, então, transmite a página ao navegador. Outros protocolos de aplicação são usados para transferências de arquivos, correio eletrônico e transmissão de notícias pela rede.

### 2.11 PROTOCOLO MODBUS

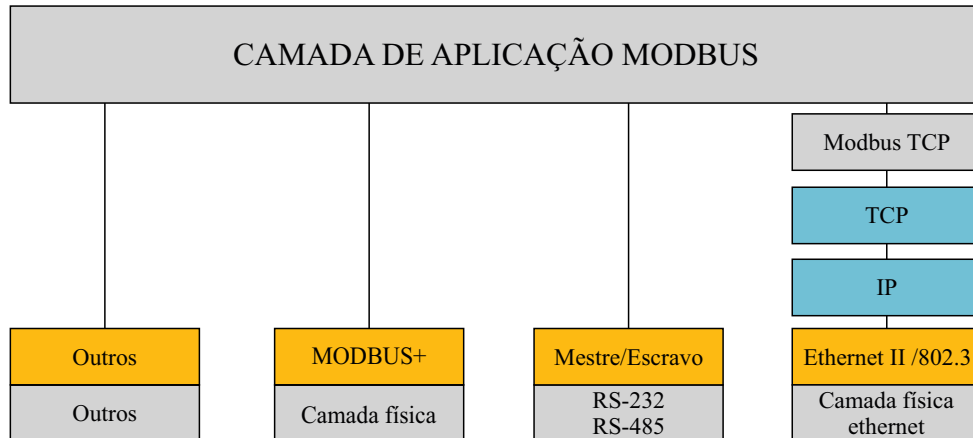
O Modbus é um protocolo industrial desenvolvido em 1979 para possibilitar a comunicação entre dispositivos de automação. Originalmente implementado como um protocolo de nível de aplicação destinado a transferir dados por uma camada serial, o Modbus foi ampliado para incluir implementações em comunicações seriais, TCP/IP e UDP (*User Datagram Protocol*) (National Instruments, 2021). Por definição, Modbus é um protocolo de mensagens da camada de aplicação, posicionado no nível 7 do modelo OSI, que fornece comunicação cliente/servidor entre dispositivos conectados em diferentes tipos de barramentos ou redes (MODBUS, 2023).

Ele permite comunicações eficientes entre dispositivos interconectados com base em uma metodologia de solicitação/resposta. Ele pode ser usado por dispositivos extremamente simples, como sensores ou motores, para se comunicar com um computador mais complexo, que pode ler medições e realizar análises e controle. O suporte a um protocolo de comunicação em um dispositivo simples requer que a geração, transmissão e recebimento da mensagem exijam pouca sobrecarga de processamento. Como o Modbus é um protocolo da camada 7, ele pode operar independentemente dos protocolos de rede subjacentes e permitiu que o Modbus fosse facilmente adaptado às arquiteturas de redes seriais e roteáveis (CARCANO, 2013). Conforme apresentado na Figura 25, o Modbus pode ser implementado de diversas formas:

1. TCP/IP sobre *Ethernet*;
2. Transmissão serial assíncrona (RS-232, RS-485, fibra óptica, radiofrequência, etc);

### 3. Modbus PLUS, que proporciona passagem de *tokens* em alta velocidade;

Figura 25 – Representação das camadas do modelo ISO/OSI no protocolo Modbus

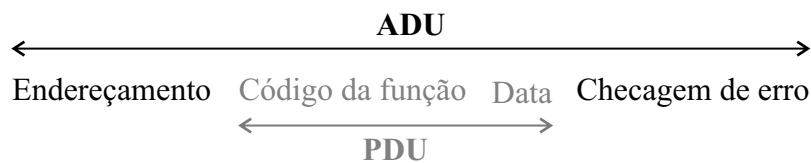


Fonte: Adaptado de Modbus (2023)

#### 2.11.1 Modbus *Application Data Unit* - ADU

A especificação do protocolo Modbus define o formato de uma mensagem como uma ADU. A ADU é composta a partir dos campos apresentados na Figura 26. No entanto, o formato da ADU pode variar dependendo da interface de comunicação usada. Por exemplo, a ADU para TCP/IP inclui um campo adicional para o endereço IP do dispositivo escravo.

Figura 26 – Representação da ADU e PDU dentro de uma mensagem Modbus.



Fonte: Adaptado de Modbus (2023)

O *Protocol Data Unit* (PDU) é a parte que contém as informações essenciais para o protocolo Modbus e é independente das camadas de comunicação subjacentes. Os campos adicionais, endereçamento e checagem de erro, estão relacionados ao protocolo de comunicação usado pelas camadas abaixo do modelo OSI.

- Endereçamento: identificador do escravo (1 *byte*);
- Data: define o tipo de ação requerida (1 *byte*);

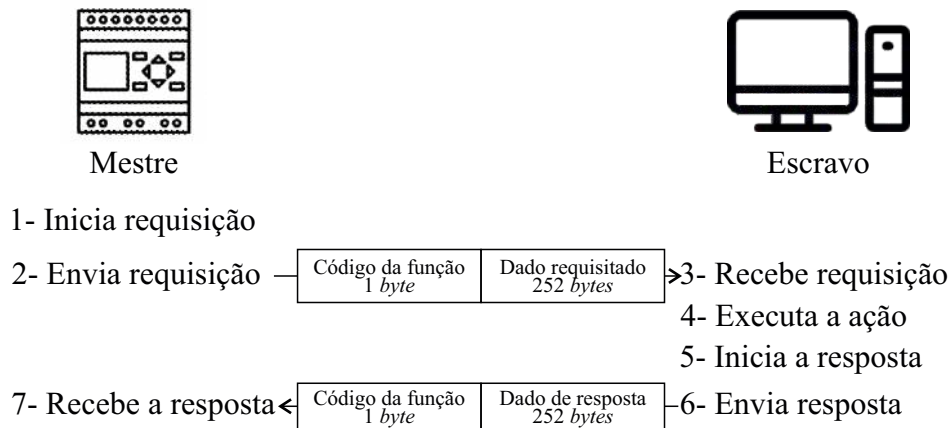
- Código de função: contém informações adicionais usadas para realizar a ação definida pelo código de função. Pode incluir itens como endereços discretos e de registro, a quantidade de itens a serem manipulados e a contagem de *bytes* de dados reais no campo (252 *bytes*);
- Checagem de erro: a verificação de redundância cíclica é uma função de controle de erro.

Em sua estrutura, a PDU é constituída por um código de função seguido por um conjunto de dados associado. As dimensões e o conteúdo desses dados são definidos pelo código de função, mas toda a PDU (código de função e dados) não pode ultrapassar 253 *bytes*. Cada código de função tem um comportamento específico, que os escravos têm flexibilidade para implementar com base no comportamento desejado da aplicação. A especificação de PDU define os conceitos básicos para o acesso e manipulação dos dados; entretanto, um escravo pode tratar os dados de uma maneira não definida explicitamente na especificação (National Instruments, 2021).

### 2.11.2 Transações no protocolo Modbus

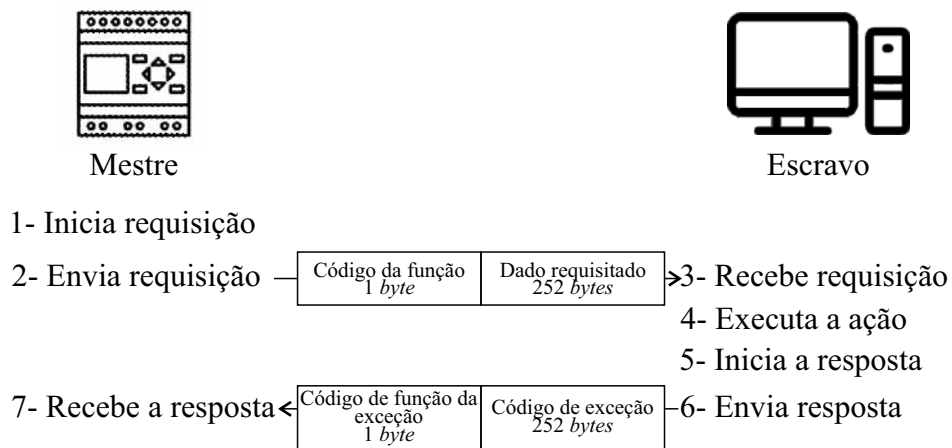
Conforme apresentado, o protocolo Modbus segue uma arquitetura mestre/escravo (cliente/servidor para Modbus TCP/IP), onde um mestre solicitará dados do escravo. O mestre também pode solicitar que o servidor execute algumas ações. Apenas o mestre pode iniciar transações ou consultas. Os demais dispositivos, escravos, respondem fornecendo os dados solicitados ao mestre ou realizando a ação solicitada na consulta. Existem dois tipos de transações: as que ocorrem sem erros (Figura 27) e as que ocorrem erros (Figura 28) (CARCANO, 2013). Já foi dito que a PDU do Modbus é composta por um campo de função e outro de dados, possuindo 1 e 252 *bytes*, respectivamente. Porém, quando ocorre um erro na troca de mensagens, o escravo (ou servidor) retorna um quadro *Exception Code*. Sua estrutura também é mostrada na Figura 28.

Figura 27 – Representação de uma mensagem Modbus sem erro



Fonte: Adaptado de Carcano (2013).

Figura 28 – Representação de uma mensagem Modbus com erro



Fonte: Adaptado de Carcano (2013).

### 2.11.3 Modelo de dados Modbus e Códigos de Funções

As transações Modbus sempre executam um conjunto de ações lendo ou gravando em um conjunto de quatro tipos de dados (CARCANO, 2013). O Quadro 4 descreve os quatro formatos de dados, bem como suas funcionalidades. O Quadro 5 mostra como é feito o acesso aos dados Modbus.

Quadro 4 – Modelo de dados Modbus

Referência	Nome da função	Tipo do objeto	Ação
0xxxx	<i>Read/Write Discrete Outputs/Coils</i>	Único <i>bit</i> (1 <i>bit</i> ). Apenas leitura	Fornecido por um sistema <i>ON/OFF</i>
1xxxx	<i>Read Discrete Inputs</i>	Único <i>bit</i> (16 <i>bits</i> ). Leitura de <i>status</i> e escrita	Alterado por uma aplicação
3xxxx	<i>Read Input Registers</i>	Apenas leitura (1 <i>bit</i> ). Apenas leitura	Fornecido por um sistema <i>ON/OFF</i>
4xxxx	<i>Read/Write Output/Holding Registers</i>	Leitura e escrita (16 <i>bits</i> ).	Alterado por uma aplicação

Quadro 5 – Acesso aos dados Modbus

Descrição				Código da Função	
				Código	Sub Código
Acesso ao dado	Bit Access	Input Registers Físicos	Read Discrete Inputs	2	
		Bits Internos ou Boils físicas	Read Coils	1	
			Write Single Coil	5	
			Write Multiple Coils	15	
	16 Bits Access	Input Registers físicos	Read Input Register	4	
		Registers internos ou registers de saídas	Read Holding Registers	3	
			Write Single Register	6	
			Read/Write Multiple Registers	16	
Diagnósticos			Ler Exception Status	7	
			Diagnóstico	8	00-18, 20
			Reportar ID do servidor	17	
			Ler ID do Dispositivo	43	14

De acordo com Carcano (2013), a especificação Modbus original incluía dois modos de codificação possíveis:

- RTU: é a implementação mais comum, usando codificação binária e verificação de erro CRC (*Cyclic Redundancy Check*), que é uma técnica matemática para garantir a integridade dos dados transmitidos. O CRC calcula um valor de verificação que é anexado aos dados, permitindo que o receptor verifique se ocorreram erros durante a transmissão;
- ASCII: é menos eficiente, pois apesar de utilizar a codificação ASCII, utiliza *bits* em excesso, a verificação de erros LRC (*Longitudinal Redundancy Check*) também é menos eficaz. O LRC calcula um valor de verificação com base na soma de todos os caracteres transmitidos. No entanto, ele não é tão robusto quanto o CRC, pois não consegue detectar certos tipos de erros, como substituições de *bits*.

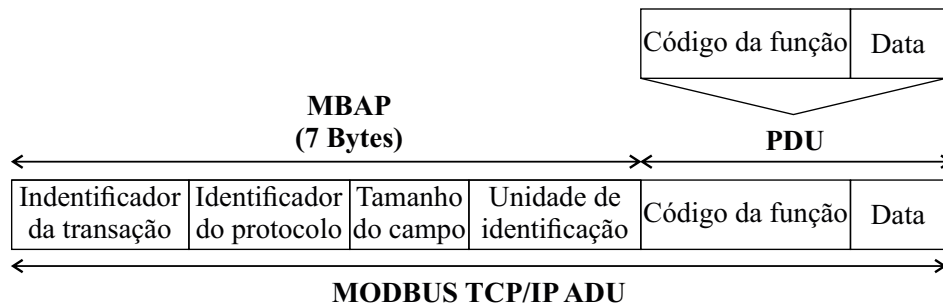
Os dois modos são incompatíveis, portanto, um dispositivo configurado para o modo ASCII não pode se comunicar com outro usando RTU. Modbus/TCP é um desenvolvimento muito mais recente, criado para permitir que o protocolo Modbus ASCII/RTU seja transportado apenas por redes baseadas em TCP/IP. Dessa forma, essa configuração incorpora mensagens Modbus em quadros TCP/IP.

## 2.12 MODBUS TCP/IP

É possível classificar o Modbus TCP/IP como um protocolo Modbus RTU aplicado à *Ethernet*. TCP/IP refere-se ao *Transmission Control Protocol* e *Internet Protocol*, que fornece o meio de transmissão para suas mensagens. O TCP/IP permite que blocos de dados binários sejam trocados entre computadores. É também um padrão mundial que serve de base para a *World Wide Web* (WWW). A principal função do TCP é garantir que

todos os segmentos de dados sejam recebidos corretamente, enquanto o IP garante que os pacotes sejam endereçados e roteados corretamente. Observe que a combinação TCP/IP é meramente um protocolo de transporte e de rede e não define o que os dados significam ou como os dados devem ser interpretados (este é o trabalho do protocolo de aplicação, Modbus neste caso). Portanto, em resumo, Modbus TCP/IP usa TCP/IP e *Ethernet* para transportar os dados da estrutura de mensagens Modbus entre dispositivos compatíveis. Ou seja, o Modbus TCP/IP combina uma rede física (*Ethernet*), com um padrão de rede e um método padrão de representação de dados (protocolo de aplicação). Essencialmente, sua mensagem é simplesmente uma comunicação Modbus encapsulada em um invólucro *Ethernet* e TCP/IP (ACROMAG, s.d.). A Figura 29 apresenta a construção da ADU do Modbus TCP/IP a partir da ADU do Modbus tradicional (ACROMAG, s.d.).

Figura 29 – Demonstração da ADU do Modbus TCP/IP quando comparado com a ADU do Modbus tradicional



Fonte: Adaptado de Acromag (s.d.).

Conforme representado, a PDU (*Protocol Data Unit*) é a mesma, mas há um cabeçalho adicional chamado cabeçalho MBAP (*Application Protocol Header*). Ele contém alguns campos úteis para o protocolo TCP/IP.

- Identificador de transição: identifica uma transação Modbus Solicitação/Resposta. Inicializado pelo mestre e recopiado pelo escravo a partir da solicitação recebida. Possui 2 *bytes*;
- Identificador de protocolo: este campo é sempre 0 para serviços Modbus e outros valores são reservados para extensões futuras. É iniciado pelo mestre. Possui 2 *bytes*;
- Tamanho: este campo é uma contagem de *bytes* dos campos restantes e inclui o *byte* do identificador da unidade, o *byte* do código de função e os campos de dados. Possui 2 *bytes*;
- Identificador de unidade: identificação de um escravo remoto. Possui 1 *byte*.



### 2.13 PADRÃO IEEE 745

De acordo com (VIANA, s. d.), o padrão IEEE 754, recomendado pelos institutos ANSI (*American National Standard Institute*) e IEEE (*Institute of Electrical and Eletronic Engineers*), refere-se às normas a serem seguidas pelos fabricantes de computadores e construtores de compiladores de linguagens científicas, ou de bibliotecas de funções matemáticas, na utilização da aritmética binária para números de ponto flutuante. As recomendações são relativas ao armazenamento de dados numéricos, métodos de arredondamento, tratamento de casos de *underflow* e *overflow*, formas de realização das quatro operações aritméticas básicas e implementação de funções nas linguagens de programação.

Conforme Chipkin (2022), o protocolo Modbus foi projetado baseado em dispositivos com um comprimento de registro de 16 *bits*. Consequentemente, conforme a evolução dos dispositivos, foram necessárias alterações para que o Modbus também suportasse dados de 32 *bits*. Esta implementação estabeleceu-se em usar dois registros consecutivos de 16 *bits* para representar 32 *bits* de dados ou essencialmente 4 *bytes* de dados. Desta forma, é a partir destes 4 *bytes* de dados que os dados de ponto flutuante de precisão única podem ser codificados em uma mensagem Modbus. Vale ressaltar que o Modbus em si não define um tipo de dados de ponto flutuante, mas é amplamente aceito que ele implemente dados de ponto flutuante de 32 *bits* usando o padrão IEEE 754.

### 2.14 DESENVOLVIMENTO WEB

Desenvolvimento *Web* refere-se, em geral, às tarefas associadas ao desenvolvimento de *sites* para hospedagem via intranet ou Internet. O processo de desenvolvimento *Web* inclui *Web design*, desenvolvimento de conteúdo *Web*, *scripts* do lado cliente/servidor e configuração de segurança de rede, entre outras tarefas. Ou seja, o desenvolvimento engloba todas as ações, atualizações e operações necessárias para construir, manter e gerenciar um *site* para garantir que seu desempenho, experiência do usuário e velocidade sejam ideais (TECHOPEDIA, 2022).

O desenvolvimento *Web* possui a seguinte hierarquia:

- Programação do lado do cliente;
- Programação no lado do servidor;
- Banco de dados.

Navegadores comunicam-se com *Web servers* utilizando o *HyperText Transfer Protocol* (HTTP). Por exemplo, quando acessa-se uma página na *Web*, envia-se uma solicitação HTTP (*HTTP request*) para o servidor de destino. A solicitação inclui uma URL

que identifica o recurso afetado, um método que define a ação necessária e pode incluir informações adicionais codificadas em parâmetros de URL, como dados POST (dados enviados pelo método HTTP POST) (MDN, 2022c). Os servidores da *Web* esperam por mensagens de solicitação do cliente, processam-nas quando chegam e respondem ao navegador *Web* com uma mensagem HTTP *response*. A resposta contém uma linha de *status* indicando se a solicitação foi bem-sucedida ou não.

Essa seção tem como objetivo mostrar a definição dos elementos presentes no desenvolvimento de sistemas, bem como apresentar as tecnologias utilizadas no desenvolvimento deste projeto.

### 2.14.1 Páginas estáticas e dinâmicas

Páginas *Web* podem ser classificadas como páginas estáticas ou dinâmicas.

Conforme MDN (2022c), um *site* estático é aquele que retorna o mesmo conteúdo embutido em código do servidor sempre que um determinado recurso é solicitado.

Um *site* dinâmico é aquele em que parte do conteúdo da resposta é gerado dinamicamente, apenas quando necessário. Em um *site* dinâmico, as páginas HTML são normalmente criadas inserindo dados de um banco de dados em espaços reservados em modelos HTML (MDN, 2022c).

A maior parte do código para oferecer suporte a um *site* dinâmico deve ser executado no servidor. A criação desse código é denominado *server-side programming* (ou, às vezes, *back-end scripting*). De forma análoga às páginas estáticas, os navegadores enviam solicitações HTTP para o servidor, então o servidor processa as solicitações e retorna as respostas HTTP apropriadas.

O código do lado do servidor pode ser escrito algumas linguagens de programação como PHP, por exemplo. O código do lado do servidor tem acesso total ao sistema operacional do servidor e o desenvolvedor pode escolher qual linguagem de programação (e versão específica) deseja usar.

### 2.14.2 Desenvolvimento de aplicações com PHP

Segundo Brown *et al.* (s.d.), o PHP (*Hypertext Preprocessor*), é uma linguagem de *script open source* de uso geral utilizada e adequada para o desenvolvimento *Web*, ainda possibilitando ser embutida no HTML. Trata-se de uma linguagem de *script* criada para comunicações do lado do servidor. Uma linguagem de *script* serve para automatizar a execução de tarefas num ambiente de tempo de execução especial. Conseqüentemente, ela consegue lidar com várias funções junto do servidor, como coletar formulários de da-

dos, gerenciar arquivos do servidor, modificar bases de dados, dentre outros.

### 2.14.3 Banco de Dados

Um banco de dados é uma coleção organizada de informações estruturadas, normalmente armazenadas eletronicamente em um sistema de computador. Um banco de dados é geralmente controlado por um sistema de gerenciamento de banco de dados (DBMS). Juntos, os dados e o DBMS, juntamente com os aplicativos associados a eles, são chamados de sistema de banco de dados, geralmente abreviados para apenas banco de dado (ORACLE, 2022).

Os dados nos tipos mais comuns de bancos de dados em operação são atualmente modelados em linhas e colunas em uma série de tabelas para tornar o processamento e a consulta eficiente. Os dados podem ser facilmente acessados, gerenciados, modificados, atualizados, controlados e organizados. A maioria dos bancos de dados usa a linguagem de consulta estruturada (SQL) para escrever e consultar dados. De acordo com Oracle (2022), SQL é uma linguagem de programação usada por quase todos os bancos de dados relacionais para consultar, manipular e definir dados e fornecer controle de acesso. O SQL foi desenvolvido pela primeira vez na IBM nos anos 1970, o que levou à implementação do padrão SQL ANSI.

### 2.14.4 XML

De acordo com Junior (s. d.), o XML (*Extensible Markup Language*) é linguagem de marcação de dados (*meta-markup language*) que provê um formato para descrever dados estruturados. Isso facilita declarações mais precisas do conteúdo e resultados mais significativos de busca através de múltiplas plataformas. O XML também vai permitir o surgimento de uma nova geração de aplicações de manipulação e visualização de dados via Internet.

O XML provê uma representação estruturada dos dados que mostrou ser amplamente implementável e fácil de ser desenvolvida.

Implementações industriais na linguagem SGML (*Standard Generalized Markup Language*) mostraram a qualidade intrínseca e a força industrial do formato estruturado em árvore dos documentos XML. O XML é um subconjunto do SGML, o qual é otimizado para distribuição através da *Web*, e é definido pelo *Word Wide Web Consortium (W3C)*, assegurando que os dados estruturados serão uniformes e independentes de aplicações e fornecedores.

O XML provê um padrão que pode codificar o conteúdo, as semânticas e as esque-

matizações para uma grande variedade de aplicações, desde simples até as mais complexas, dentre elas:

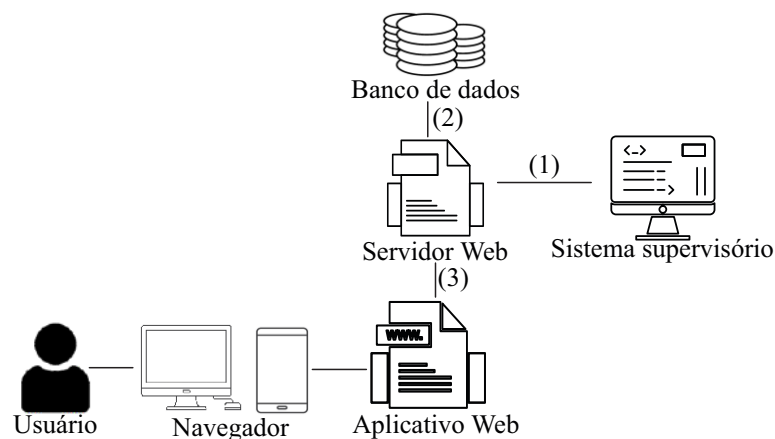
- Documento simplificado;
- Registro estruturado, visando ordem em seus objetos;
- Representação gráfica, como, por exemplo, interface com aplicações de usuário;
- Utilização de entidades e tipos de esquemas padronizados.

Ainda é observado por Junior (s. d.) que uma característica importante é que, uma vez recebido o dado pelo cliente, tal dado pode ser manipulado, editado e visualizado sem a necessidade de relacionar o servidor. Dessa forma, reduzindo-se a necessidade de computação e também a requisição de banda passante para as comunicações entre cliente e servidor.

## 2.15 CRIAÇÃO DO SISTEMA SUPERVISÓRIO

A *interface* de conexão entre o circuito de instrumentação e o processo de ensaio de elevação de temperatura é feita através da criação de um aplicativo. Este procura monitorar e controlar o ensaio realizado, possibilitando a supervisão remota do mesmo. O aplicativo é composto por um sistema supervisório que opera junto do *software* de supervisão da fonte emuladora de tensão, mostrando suas medições sem prejudicar seu funcionamento. Os dados mostrados no sistema supervisório são armazenados em um banco de dados, visando disponibilizar as informações em uma página *Web*. A Figura 30 demonstra visualmente a proposta de automação descrita. A partir dessa página o metrologista consegue monitorar remotamente as variáveis relevantes para o ensaio de elevação de temperatura.

Figura 30 – Demonstração visual da proposta de automação.



Fonte: Elaborado pelo autor.

### 2.15.1 Sistema Supervisório

De acordo com ELIPSE (2019), os sistemas supervisórios permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou de uma instalação física. Tais informações são coletadas por equipamentos de aquisição de dados e, em seguida, manipuladas, analisadas, armazenadas e, posteriormente, apresentadas ao usuário. Estes sistemas também são chamados SCADA (*Supervisory Control and Data Acquisition*).

Neste trabalho, o sistema supervisório é construído utilizando a linguagem C#. Em seu escopo devem conter:

- Plataforma para comunicação e supervisão da fonte emuladora de tensão;
- Plataforma para supervisão dos dados obtidos pelo circuito eletrônico de instrumentação;
- Comunicação com o servidor *Web* para enviar os dados para o banco de dados remoto.

O C# é uma linguagem de programação moderna, orientada a objeto e fortemente tipada. Essa linguagem permite que os desenvolvedores criem muitas categorias de aplicativos seguros e robustos executados no .NET. Em seu núcleo, O C# é uma linguagem orientada a objeto (WAGNER, 2021). Já o .NET é uma plataforma de desenvolvimento gratuita e de *software* livre para a criação de muitas categorias de aplicativos, como aplicativos móveis, aplicativos *Web*, móveis, área de trabalho, IoT, dentre outros. Programas C# são executados no .NET, um sistema de execução virtual chamado CLR (*Common Language Runtime*). Incluso no *framework* .NET, há o *Windows Forms*, uma estrutura para criação de aplicativos. Ele fornece formas de criar aplicativos com base nas ferramentas fornecidas pelo *Visual Studio*.

### 2.15.2 Banco de Dados

O *servidor Web* age como uma do conexão do banco de dados com o aplicativo *Web* e o sistema supervisório.

A consulta no banco de dados é realizada por *queries*, que pode ser definida como um pedido de informação, como solicitações ou requisições. Na linguagem de banco de dados, essas consultas são feitas por inserções, atualizações, seleções e exclusões. Ou seja, são comandos que, quando executados, agem na tabela criada. Cada linguagem de consulta retorna diferentes categorias de dados, conforme a função utilizada. Por exemplo, com o SQL os programas interagem com um banco de dados relacional e geram informações organizadas em linhas e colunas (CLEMENTE, 2022).

### 2.15.3 Banco de Dados - Aplicativo *Web*

Uma vez que os dados são amostrados pelo sistema supervisorio, eles são armazenados em um banco de dados para que possam ser acessados posteriormente por meio de um aplicativo *Web*. Isso permite que os dados sejam analisados e interpretados.

O aplicativo *Web* é a página acessada pelo metrologista onde são amostrados os dados do ensaio. Esses dados são obtidos através do banco de dados remoto apresentado anteriormente.

Ou seja, o aplicativo *Web* se trata de uma página para amostrar as medições feitas nos processos anteriores. É uma página escrita em PHP, utilizando elementos HTML (Linguagem de Marcação de Hipertexto). O PHP trata-se de uma linguagem de *script* criada para comunicações do lado do servidor. Uma linguagem de *script* serve para automatizar a execução de tarefas num ambiente de tempo de execução especial. Consequentemente, ela consegue lidar com várias funções junto do servidor, como coletar formulários de dados, gerenciar arquivos do servidor, modificar bases de dados, dentre outros.

Segundo MDN (2022b), HTML não é uma linguagem de programação, mas sim uma linguagem de marcação utilizada para definir a estrutura do seu conteúdo. HTML consiste em uma série de elementos, onde são delimitados ou agrupados diferentes partes do conteúdo para aparecer na página estruturada pelo usuário.

## 2.16 CONCLUSÕES PARCIAIS

Neste capítulo foi realizada uma revisão bibliográfica a respeito da base teórica necessária para o desenvolvimento do presente trabalho. Inicialmente, foi apresentado o ensaio de elevação de temperatura e seu método utilizado para testes. Em seguida, teve início o estudo de conceitos fundamentais para construção da placa de circuito impresso: amplificadores operacionais, filtros analógicos, instrumentação e aquisição de dados. Também foram apresentados os equipamentos responsáveis pela execução do ensaio de elevação de temperatura: o equipamento JMAN MONITEK 9451 e a fonte emuladora FCATQ 1000-60-40-PFC55450. Por fim, foram apresentados conceitos a respeito do sistema supervisorio proposto e também a criação de uma página *Web*.

### 3 SOLUÇÃO PROPOSTA

No presente capítulo, será apresentada a solução proposta para a automação do ensaio de elevação de temperatura em transformadores de distribuição. Com o intuito de alcançar esse objetivo, foram desenvolvidos três tópicos principais: o projeto do circuito de instrumentação de sinais, a construção do sistema supervisor e a criação da página *Web*. Cada um desses tópicos será detalhado, demonstrando as soluções adotadas e os resultados obtidos. Além disso, será realizada uma discussão sobre a integração entre esses elementos, evidenciando como eles se complementam para tornar a automação do ensaio possível.

#### 3.1 DESENVOLVIMENTO DO CIRCUITO DE INSTRUMENTAÇÃO

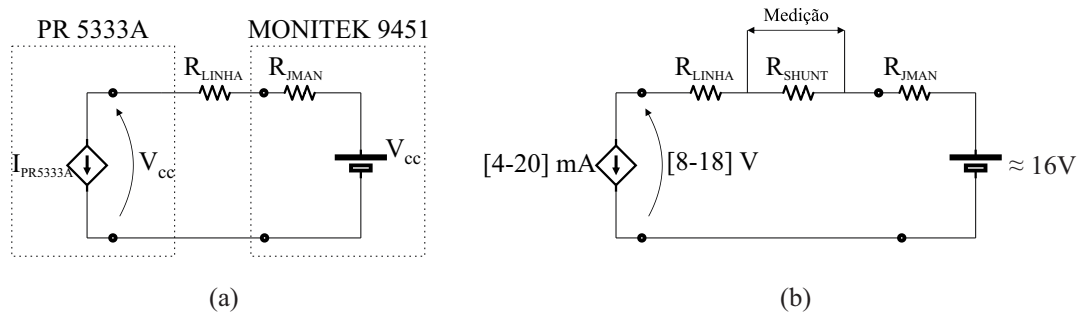
Esta seção tem como objetivo apresentar o processo de construção do circuito de instrumentação, desde o dimensionamento dos componentes até a validação do circuito por meio de simulações. Também será apresentado o projeto da placa de circuito impresso referente ao circuito proposto, mostrando todos os passos necessários para a sua construção. Com essas informações, será possível compreender de forma completa e detalhada o processo de construção do circuito de instrumentação utilizado no ensaio.

##### 3.1.1 Método de medição

O ensaio de elevação de temperatura é realizado pelo equipamento JMAN MONITEK 9451. Ou seja, esse equipamento integra todos elementos de medições, inclusive o sensor de temperatura Pt100. O presente trabalho não tem como objetivo substituir o procedimento de ensaio já existente, mas sim aprimorá-lo. Portanto, é fundamental que, ao inserir uma nova medida no sensor de temperatura, não ocorra uma inserção de erros no circuito de medição original.

Para garantir essa condição, foram realizadas medidas nas linhas de conexão do sensor de temperatura. A Figura 31 (a) mostra como é feita a medição pelo equipamento JMAN MONITEK 9451. Foi considerado uma carga  $R_{LINHA}$  que representa as perdas na linha de conexão do sensor. Após a medição, foram obtidos os resultados apresentados na Figura 31 (b). Ao analisar os resultados, foi possível concluir que a melhor forma de medir a temperatura do Pt100 é inserir um resistor *shunt* em série com a medição do JMAN MONITEK 9451. Essa condição garante que a nova medição não resulte em erros na execução do ensaio de elevação de temperatura.

Figura 31 – Diagrama representativo da medição de temperatura do equipamento JMAN MONITEK 9451 (a) Sistema de medição padrão. (b) Proposta de medição inserindo um resistor shunt em série na medição padrão.

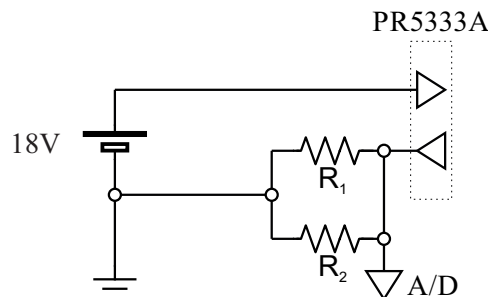


Fonte: Elaborado pelo autor

### 3.1.1.1 Circuito de medição de temperatura – JMAN MONITEK 9451

Antes de proceder com o projeto de instrumentação do sensor de temperatura, é necessário compreender o circuito de medição de temperatura do equipamento de medição JMAN MONITEK 9451. O circuito que representa a medição é mostrado na Figura 32.

Figura 32 – Circuito de medição de temperatura do equipamento JMAN MONITEK 9451.

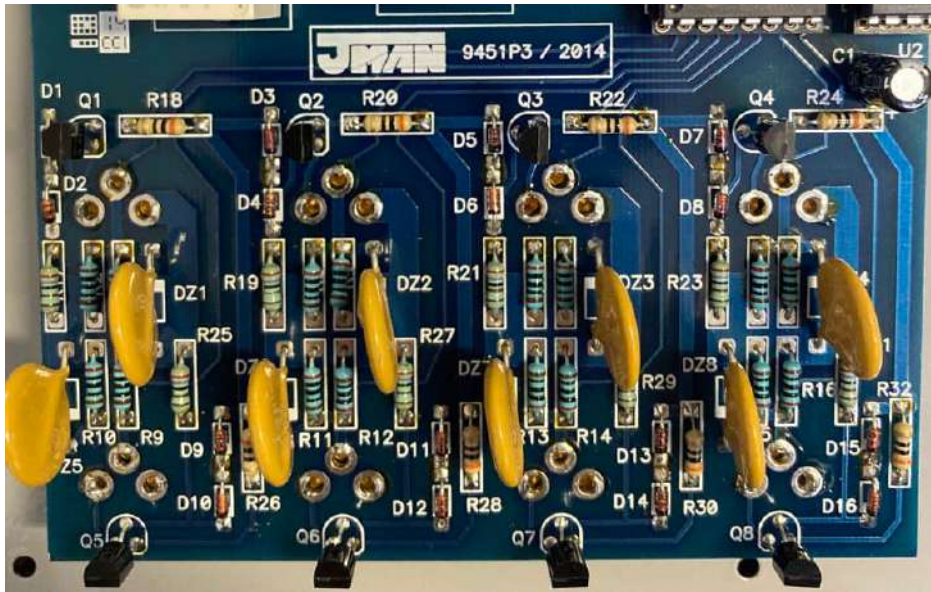


Fonte: Elaborado pelo autor

A medição é feita através de 2 pinos, como mostrado na Figura 32. Em sua composição, o valor de 18V é alcançado através do transistor PNP modelo BC327, operando como uma fonte regulada. Já o valor utilizado para medição é obtido através da associação em paralelo dos resistores *shunt*  $R_1$  e  $R_2$ , ambos no valor de  $200 \Omega$  com tolerância de 1%. Esses resistores foram associados em paralelo para garantir maior precisão na medição. Assim, evita-se valores elevados de dissipação de potência, pois o calor pode ocasionar erros de medição. Como resultado, obtém-se  $100 \Omega$ . O resultado da medição é enviado para o conversor analógico digital MCP3208. A Figura 33 mostra a placa de medição para medição de temperatura do JMAN MONITEK 9451.



Figura 33 – Conexões elétricas para medição de temperatura



Fonte: Elaborado pelo autor

### 3.1.1.2 Conectores de medição

A ligação entre os Pt100 e o JMAN MONITEK 9451 é feita por um conector de 3 vias, modelo Lumberg, Figura 34. Dentre suas características, possui temperatura de operação entre  $-40^{\circ}\text{C}$  até  $85^{\circ}\text{C}$ .

Figura 34 – Conector circular Lumberg. (a) Conector Macho modelo SV30. (b) Conector Fêmea modelo KVF30.



(a)

(b)

Fonte: Adaptado de Lumberg (2021)

### 3.1.1.3 Medição dos Relés

Conforme apresentado em sua Tese de Doutorado, onde desenvolveu os equipamentos de medição JMAN MONITEK, é abordado sobre a utilização dos relés para automação do ensaio de elevação de temperatura:

[...] inicialmente comanda o fechamento do relé do varivolt motorizado, fazendo a tensão subir até que a potência ativada lida no JMAN MONITEK 9443, seja igual à potência das perdas totais calculadas anteriormente e cadastrada no programa. Se a potência ativa for maior que a desejada, será comandado o fechamento de outro relé que fará a tensão diminuir. Este processo, garante que o valor das perdas totais fique constante ao longo de todo ensaio. (VICENTE, 2006).

O equipamento JMAN MONITEK 9443 é a versão antiga do JMAN MONITEK 9451 (vide Figura 35), porém a configuração dos relés continua a mesma. Vale observar que, inicialmente, os relés foram projetados para controlar um varivolt motorizado para controle do nível de tensão do ensaio de elevação de temperatura. Porém, nada impede que esses relés sejam utilizados para controlar o nível de tensão aplicando outros métodos.

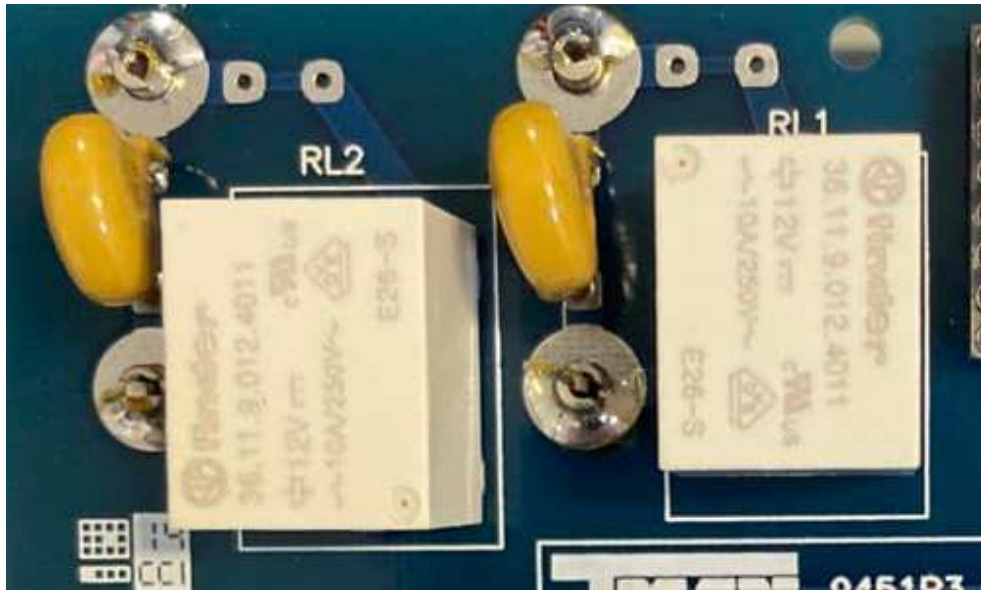
Figura 35 – Conexões elétricas para medição de temperatura e leitura dos relés do equipamento de medição JMAN MONITEK 9451.



Fonte: Elaborado pelo autor

O relé utilizado no circuito é do modelo *Finder* 36.11.9.012.4011. A Figura 36 mostra a conexão dos relés no equipamento JMAN MONITEK 9451.

Figura 36 – Conexões elétricas para acionamento dos relés do equipamento de medição JMAN MONITEK 9451.



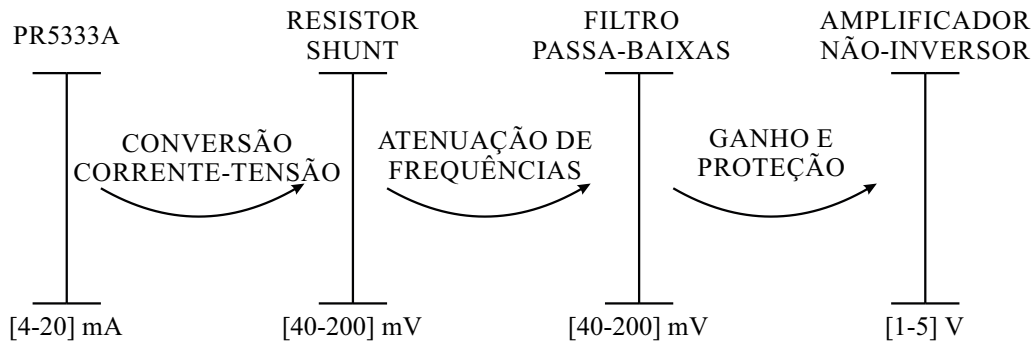
Fonte: Elaborado pelo autor

### 3.1.2 Instrumentação

#### 3.1.2.1 Projeto do circuito de instrumentação do sensor de temperatura

O condicionamento do sinal proveniente do sensor de temperatura teve início ao adicionar um resistor *shunt* em série com a medição do equipamento JMAN MONITEK 9451, mostrado na Figura 31 (b). Esse primeiro estágio foi nomeado como conversão corrente-tensão. O estágio seguinte compõe a filtragem do sinal, atenuando ruídos presentes no circuito de medição ao aplicar um filtro passa-baixas. Esses ruídos podem ser produzidos pela fonte emuladora de tensão. O próximo estágio é responsável pela amplificação do sinal, condicionando-o de acordo com as especificações da entrada analógica do microcontrolador Arduino Nano<sup>®</sup>. Além disso, esse estágio também é responsável por proteger o microcontrolador contra sobretensões. Esses estágios são representados por diagramas de blocos na Figura 37.

Figura 37 – Apresentação gráfica do condicionamento de sinal do sensor de temperatura



Fonte: Elaborado pelo autor

### 3.1.2.2 Projeto do resistor shunt – conversão corrente-tensão

Na subseção 3.1.2.1 foi abordado a importância de escolher um resistor adequado para a conversão corrente-tensão. Como o resistor *shunt* utilizado pelo JMAN MONITEK 9451 é de  $100 \Omega$ , então para o projeto é adequado escolher um resistor *shunt* de valor menor, como  $10 \Omega$ . Assim, obtém-se a queda de tensão ao aplicar a Equação (3.1).

$$V_1 = I_{PR5333A} \cdot R_{SHUNT} \quad (3.1)$$

O resultado é obtido ao substituir os valores na Equação (3.2), resultando em (3.3).

$$V_1 = ([4 \text{ } 20]mA) \cdot (10 \Omega) \quad (3.2)$$

$$V_2 = [40 \text{ } 200]mV \quad (3.3)$$

### 3.1.2.3 Projeto do filtro passa-baixas – atenuação de ruídos de alta frequência

A fim de atenuar ruídos no sistema de medição, o filtro passa-baixas é projetado. No projeto proposto, o principal elemento causador de ruídos é a fonte emuladora de tensão.

A medição dos ruídos foi feita utilizando um osciloscópio modelo *Tektronix* THS3014. Como resultado foram medidos ruídos na frequência de 6,2 kHz na saída da fonte. Para o dimensionamento do projeto, julgou-se adequado projetar o filtro 2 décadas abaixo do valor medido. Dessa forma, foi adotada uma frequência de corte no valor de 62 Hz. O filtro

passa-baixas é projeto de acordo com (2.11), apresentado na subseção 2.3.

$$f_c = \frac{1}{2 \cdot \pi \cdot R_{FILTRO} \cdot C_{FILTRO}} \quad (3.4)$$

Portanto, adotando um capacitor  $C_{FILTRO}$  no valor de  $0,56 \mu F$ , conforme Equação (3.5), tem-se o resultado em (3.6).

$$62Hz = \frac{1}{2 \cdot \pi \cdot R_{FILTRO} \cdot 0,56\mu F} \quad (3.5)$$

$$R_{FILTRO} = 4,7k\Omega \quad (3.6)$$

### 3.1.2.4 Projeto do circuito amplificador – ganho e proteção

O circuito de instrumentação para o sensor de temperatura é apresentado na Figura 38. Para amplificar do sinal foi utilizado um amplificador *rail-to-rail*, modelo OPA192. Essa escolha foi feita para proteger o microcontrolador. A configuração de AmpOp aplicada foi a não-inversora, devido à alta-impedância de entrada. Essa característica é interessante para o projeto desenvolvido, pois garante que o fluxo de corrente na entrada do circuito passe completamente pelo resistor *shunt*, garantindo uma conversão satisfatória. Conforme apresentado em (2.4), apresentado na subseção 2.2.1, o ganho do circuito é dado pela Equação (3.7).

$$A_v = \frac{V_o}{V_1} = 1 + \frac{R_f}{R_1} \quad (3.7)$$

Considerando a relação entre os valores de entrada e saída, obtém-se o ganho mostrado na Equação (3.8) para um amplificador não-inversor.

$$A_v = \frac{5}{200m} = 25 \quad (3.8)$$

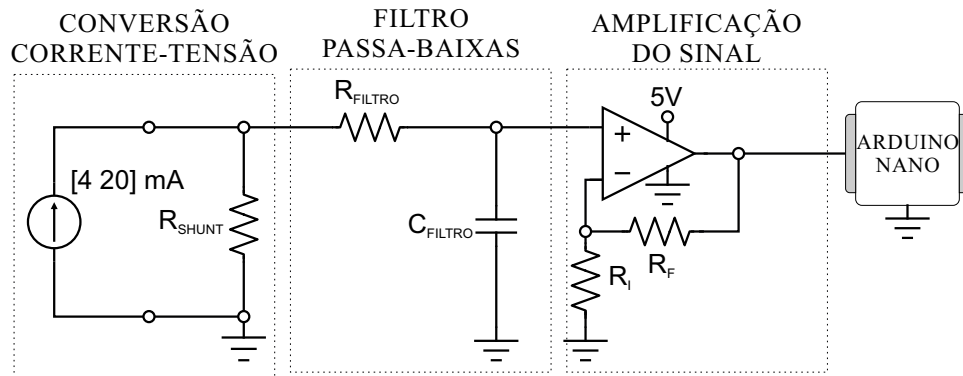
Logo, escolhendo um resistor  $R_f$  no valor de  $470 k\Omega$ , vide Equação (3.9), obtém-se  $R_1$  em (3.10):

$$25 = 1 + \frac{470k}{R_1} \quad (3.9)$$

$$R_1 = 20k\Omega \quad (3.10)$$

Finalmente, a Figura 38 representa o projeto do amplificador operacional.

Figura 38 – Circuito de condicionamento de sinal do sensor de temperatura

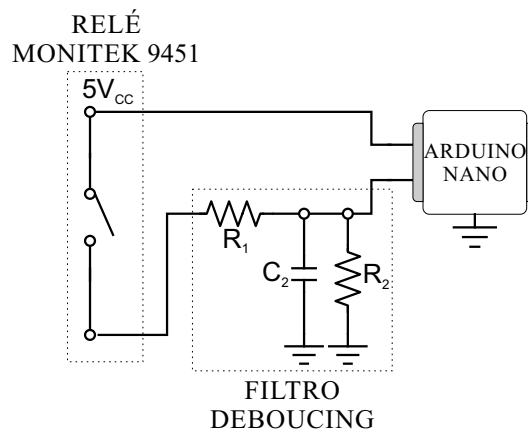


Fonte: Elaborado pelo autor

### 3.1.2.5 Projeto do circuito de leitura dos relés

Para a proposta de automação apresentada, as informações obtidas através das comutações dos relés serão enviadas para as entradas digitais do microcontrolador Arduino Nano®, como mostra Figura 39. Este dispositivo possui em sua configuração entradas digitais de até 5 V. Dessa forma, o circuito de condicionamento de sinal deve ser projetado de acordo as especificações do dispositivo de aquisição de dados.

Figura 39 – Circuito de condicionamento de sinal dos relés do equipamento MONITEK 9451.



Fonte: Elaborado pelo autor

As constantes de tempo do filtro *debouncing* são calculadas através das Equações (2.14) e (2.15). Para o projeto será considerado  $\tau_1 = 10ms$  e  $\tau_2 = 100ms$ , enquanto o capacitor  $C_2 = 0,56\mu F$ . A definição de  $R_1$  é apresentada nas Equações (3.11), (3.12) e

(3.13).

$$R_1 = \frac{\tau_1}{C_2} \quad (3.11)$$

$$R_1 = \frac{10 \times 10^{-3}}{0,56 \times 10^{-6}} \quad (3.12)$$

$$R_1 = 17,857 \times 10^3 \Omega \approx 18 \text{ k}\Omega \quad (3.13)$$

Por sua vez, o cálculo de  $R_2$  é mostrado em (3.14), (3.15) e (3.16).

$$R_2 = \frac{\tau_2}{C_2} \quad (3.14)$$

$$R_2 = \frac{100 \times 10^{-3}}{0,56 \times 10^{-6}} \quad (3.15)$$

$$R_2 = 178,571 \times 10^3 \Omega \approx 180 \text{ k}\Omega \quad (3.16)$$

### 3.1.3 Modelagem dos sensores de temperatura Pt100

O sensor de temperatura Pt100 opera junto de um transmissor de temperatura, modelo PR5333A, da *PR Electronics*<sup>®</sup>. Este componente é responsável por condicionar a leitura do sensor de temperatura no *range* de 4 a 20 mA. Porém, para ler e interpretar com exatidão esse sinal, é necessário obter uma expressão matemática que represente seu comportamento conforme a variação de temperatura medida. Dessa forma, é necessário submeter esse sensor a diferentes valores de temperatura, a fim de analisar seu comportamento. No final desse processo, será obtido, uma equação que representa o funcionamento do sensor. Conhecendo o comportamento do transmissor, finalmente é possível realizar implementar a instrumentação.

A corrente foi medida com o auxílio de um multímetro digital, instalado nos terminais do transmissor de temperatura. Os resultados obtidos são apresentados no Quadro 6, onde são mostrados as medições de corrente no sensor de temperatura Pt100.

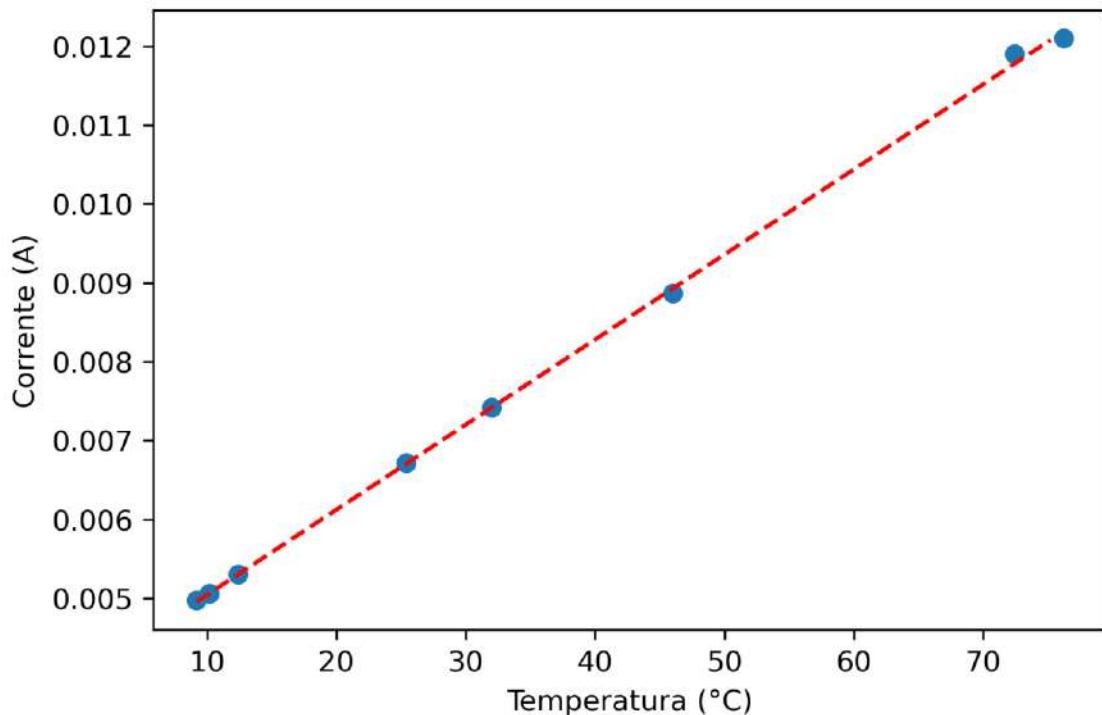
Quadro 6 – Valores medidos pelo Pt100 a partir da variação de temperatura.

Temperatura (°C)	Corrente (mA)
9,2	4,97
10,2	5,06
12,4	5,30
25,4	6,72
32,0	7,42
46,0	8,87
72,4	11,19
76,2	11,21

Fonte: Elaborado pelo autor

Os resultados da medição foram plotados no gráfico da Figura 40. O eixo horizontal representa os valores de temperatura, enquanto o eixo vertical a corrente medida.

Figura 40 – Corrente versus temperatura do sensor de temperatura.



Fonte: Elaborado pelo autor.

Como resultado, obteve-se a seguinte Equação (3.17) que representa o comportamento da variação de corrente em função da temperatura.

$$I(t) = 107,45 \times 10^{-6}t + 3,9925 \times 10^{-3} \quad (3.17)$$



O coeficiente  $R^2$ , representado pela Equação (3.19) é uma medida da qualidade de ajuste de uma linha reta a um conjunto de dados. Ele varia de 0 a 1, sendo 0 o pior ajuste possível e 1 o melhor ajuste possível. Um valor de  $R^2$  alto indica que a linha reta ajusta-se bem aos dados, enquanto um valor de  $R^2$  baixo indica que a linha reta não ajusta-se bem aos dados.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3.18)$$

Onde:

- $y_i$  é o valor real da variável dependente para o  $i$ -ésimo ponto de dados.
- $\hat{y}_i$  é o valor estimado da variável dependente para o  $i$ -ésimo ponto de dados.
- $\bar{y}$  é a média da variável dependente.
- $n$  é o número de pontos de dados.

A partir dos dados medidos, obtém-se em (3.19) o coeficiente  $R^2$ .

$$R^2 = 0,9996 \quad (3.19)$$

Como resultado, conclui-se que a reta obtida ajusta-se bem aos dados medidos pelo sensor de temperatura Pt100.

### 3.1.4 Fonte de tensão CC

O AmpOp *rail-to-rail* utilizado na instrumentação do sinal de corrente, conforme o projeto, estabelece uma  $V_{CC}$  de 5 V em seus terminais. Para satisfazer essa condição, é construída uma fonte de tensão CC. Seu projeto é apresentado a seguir.

#### 3.1.4.1 Determinação da carga

A primeira etapa para projetar uma fonte é conhecer a carga a ser alimentada. Neste projeto, a fonte é utilizada apenas para energizar o AmpOp. Consultando o *datasheet* desde componente, verifica-se que o modelo OPA192 consome aproximadamente 10 mA. Ainda é necessário garantir uma margem de segurança de 50% para o projeto, conforme (3.20).

$$I_{Carga} = 10 \text{ mA} \cdot 1,50 = 15 \text{ mA} \quad (3.20)$$

Conforme apresentado na sessão anterior, uma fonte de tensão é normalmente composta por um transformador, retificador, filtro e regulador.

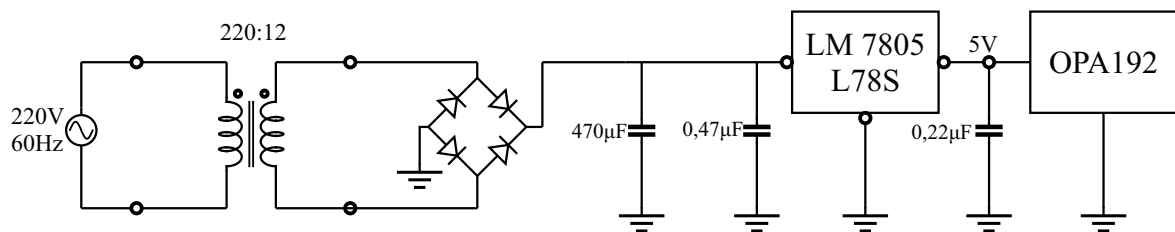
Neste projeto é utilizado um transformador comercial com as seguintes especificações.

- $V_P = 220/127 V$
- $V_S = 12 + 12 V$
- $I_S = 70 mA$

A conversão do sinal CA para CC é feita por um retificador de onda completa, junto de um filtro capacitivo no valor de  $470 \mu F$ .

O sinal é regulado por um CI regulador em série modelo LM7805. Qualquer regulador na série 78XX tem uma tensão de desligamento de 2 V a 3 V, dependendo da tensão de saída. Isso significa que a tensão de entrada tem que ser pelo menos de 2 V a 3 V maior que a tensão de saída. Segundo o *datasheet* deste componente, o modelo LM7805 regula a tensão de saída  $V_O$  para 5V. O fabricante também recomenda utilizar dois capacitores junto do CI, a fim de obter um sinal com maior estabilidade. Como resultado, obtém-se o circuito apresentando da Figura 41:

Figura 41 – Circuito elétrico para a fonte de tensão CC.



Fonte: Elaborado pelo autor.

## 3.2 DESENVOLVIMENTO DO SISTEMA SUPERVISÓRIO E PÁGINA WEB

### 3.2.1 Fonte Emuladora de Tensão Comunicação *Ethernet* através de Modbus TCP/IP

A fonte de alimentação FCATQ 10000-60-40 PFC 55450 é equipada com uma porta de comunicação *Ethernet*, através da qual pode ser operada e monitorada remotamente. Através de qualquer interface virtual, é possível alterar os valores (*preset*) de diversos parâmetros da fonte, obter os valores atualmente ajustados, as condições de operação

atuais do equipamento (*status*), e também realizar a leitura das medições realizadas pela fonte (SUPPLIER, s.d.b).

Segundo Supplier (s.d.b), o equipamento trabalha no modo escravo, ou seja, somente responde a comandos enviados pelo mestre, não iniciando uma comunicação. Assim, por operar utilizando o protocolo Modbus TCP/IP, toda comunicação entre o computador e o equipamento é feita no modo mestre-escravo, através de requisições. O equipamento trabalha no modo escravo, aguardando que sejam enviadas requisições.

As fontes Supplier utilizam as funções de leitura (*Read Multiple Registers* FC 03) e escrita (*Write Multiple Registers* FC 16). É válido ressaltar que, a fim de explicar o funcionamento dos comandos da fonte, a Supplier adaptou a representação de um *frame* Modbus. Por exemplo, o Modbus dedica dois *bytes* para os campos de endereço principal. Ao invés de mostrar esses dois *bytes*, substituiu-se por "ID" e "COMANDO". Porém, o *frame* Modbus permanece intacto. As Figuras 42 e 43 mostram a criação de um quadro Modbus TCP/IP para ler e escrever múltiplos registradores, respectivamente.

Figura 42 – Construção do frame Modbus para leitura de dados através da função Modbus de código 03.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Código da Função	ID	COMANDO	Dado 1 (DH)	Dado 2 (DL)
03	X	X	X	X

(a)

Byte 1	Byte 2	Byte 3	Byte n
Código da Função	Contagem de bytes	Dado 1 (MSB)	Dado 2 (LSB)
03	X	X	X

(b)

Fonte: Elaborado pelo autor.

Figura 43 – Construção do frame Modbus para escrita de dados através da função Modbus de código 16.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	
Código da Função	ID	COMANDO	Quantidade de endereços		Quantidade de dados		Dado 1 (DH)	Dado 2 (DL)
16	X	X	X	X	X	0x0n	X	X

(a)

Byte 1	Byte 2	Byte 3	Byte 4	Byte n...
Código da Função	Contagem de bytes	Valor endereço 1 (MSB)	Valor endereço 2 (LSB)	Quant. n de endereços
16	X	X	X	X

(b)

Fonte: Elaborado pelo autor.

O Quadro 7 mostra os valores referência para construção dos quadros Modbus da fonte emuladora de tensão Supplier.

Quadro 7 – Funções da fonte Supplier para utilização do protocolo Modbus

Descrição	ID	COM	DH	DL	FC	Função
Escrever valor tensão	0	205	DH	DL	16	Escreve o valor de tensão para as 3 fases
	1	205	DH	DL	16	Escreve o valor de tensão para a fase U
	2	205	DH	DL	16	Escreve o valor de tensão para a fase V
	3	205	DH	DL	16	Escreve o valor de tensão para a fase W
Ler os valores das correntes e potencias	X	212	X	X	03	Ver Quadro 8

Onde:

- ID: Código de Identificação da função. Trata-se do endereço inicial do registrador utilizado no protocolo Modbus, e está representado no *byte* 2;
- COM: Código de Comando da fonte. É o outro campo que compõe o endereço do registrador, juntamente com o ID. Esse código está representado no *byte* 3;
- DH: Dado para leitura/escrita. Refere-se ao *byte* mais significativo do dado que será lido ou escrito no protocolo Modbus;
- DL: Dado para leitura/escrita. Refere-se ao *byte* menos significativo do dado que será lido ou escrito no protocolo Modbus;
- FC: Código de função Modbus. Define a função que será executada no protocolo Modbus;
- X: De acordo com Supplier (s.d.b), esse valor é indiferente (pode ser utilizado 0 (zero)).

Quadro 8 – Funções da fonte Supplier para leitura das medidas utilizando o protocolo Modbus

COM	Leitura	Byte de retorno	Descrição
212	Corrente U	1	<i>Byte</i> mais significativo (MSB) da corrente medida na fase U.
		2	<i>Byte</i> menos significativo (LSB) da corrente medida na fase U.
	Corrente V	3	<i>Byte</i> mais significativo (MSB) da corrente medida na fase V.
		4	<i>Byte</i> menos significativo (LSB) da corrente medida na fase V.
	Corrente W	5	<i>Byte</i> mais significativo (MSB) da corrente medida na fase W.
		6	<i>Byte</i> menos significativo (LSB) da corrente medida na fase W.
	Tensão Eficaz U	7	<i>Byte</i> mais significativo (MSB) da tensão eficaz medida na fase U.
		8	<i>Byte</i> menos significativo (LSB) da tensão eficaz medida na fase U.
	Tensão Eficaz V	9	<i>Byte</i> mais significativo (MSB) da tensão eficaz medida na fase V.
		10	<i>Byte</i> menos significativo (LSB) da tensão eficaz medida na fase V.
	Tensão Eficaz W	11	<i>Byte</i> mais significativo (MSB) da tensão eficaz medida na fase W.
		12	<i>Byte</i> menos significativo (LSB) da tensão eficaz medida na fase W.
	Potência Ativa U	13	<i>Byte</i> mais significativo (MSB) da potência ativa medida na fase U.
		14	<i>Byte</i> menos significativo (LSB) da potência ativa medida na fase U.
	Potência Ativa V	15	<i>Byte</i> mais significativo (MSB) da potência ativa medida na fase V.
		16	<i>Byte</i> menos significativo (LSB) da potência ativa medida na fase V.
	Potência Ativa W	17	<i>Byte</i> mais significativo (MSB) da potência ativa medida na fase W.
		18	<i>Byte</i> menos significativo (LSB) da potência ativa medida na fase W.

É válido ressaltar que, no protocolo de comunicação Modbus, a leitura é representada por dois *bytes*, um *byte* mais significativo (MSB) e um *byte* menos significativo (LSB). Ou seja, para cada leitura é necessário fazer a conversão desses *bytes* para ponto flutuante, conforme apresentado na seção a seguir.

### **3.2.2 Algoritmo para microcontrolador - Comunicação do circuito de instrumentação com o sistema supervisor**

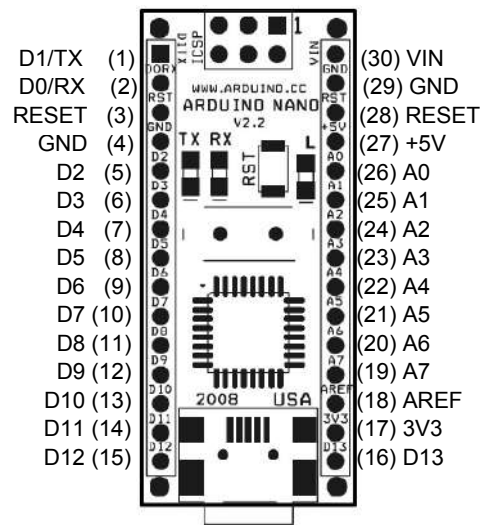
A comunicação com microcontroladores é a base para interações de sistemas embarcados, como, por exemplo, sistemas supervisórios. Ela permite que o microcontrolador receba informações de sensores. Durante a etapa de planejamento do protótipo, para realizar a comunicação do ensaio de elevação de temperatura com o sistema supervisor, foi necessário escolher um microcontrolador que atendia a todas especificações de projeto. Dessa forma, optou-se pelo microcontrolador Arduino Nano.

O Arduino Nano oferece diversas opções de comunicação serial, como UART, SPI e I2C. Assim, optou-se pela utilização da interface de comunicação UART, que permite transmitir e receber dados entre dispositivos por meio das linhas TX e RX. Esta interface possui um sistema de controle de fluxo de dados para uma comunicação eficiente e sem erros. Para utilizar a comunicação serial com o microcontrolador, é necessário configurar parâmetros como a taxa de transmissão (*baud rate*), o número de *bits* de dados e o formato de paridade, além de garantir a sincronização entre o transmissor e o receptor.

#### **3.2.2.1 Desenvolvimento da comunicação de dados**

Para o desenvolvimento de projetos com o Arduino, utilizou-se o próprio ambiente de programação fornecido pelo fabricante, o qual permite aos usuários escrever, carregar e depurar os códigos diretamente na placa de desenvolvimento. A Figura 44 mostra os pinos de entrada e saída do Arduino Nano utilizado.

Figura 44 – Pinout da placa de desenvolvimento Arduino Nano



Fonte: Adaptado de (ARDUINO, 2008).

Para o desenvolvimento do projeto é necessário utilizar entradas analógicas e entradas digitais. Portanto, foram utilizadas as seguintes entradas do Arduino:

- Pino 26 - A0: Entrada analógica utilizada para receber o sinal de corrente medido pela termorresistência Pt100;
- Pino 5 - D2: Entrada digital utilizada para receber sinal do relé controlado pelo equipamento JMAN MONITEK 9451;
- Pino 6 - D3: Entrada digital utilizada para receber sinal do relé controlado pelo equipamento JMAN MONITEK 9451;
- Pino 7 - D4: Entrada digital auxiliar reservada para projetos futuros;
- Pino 8 - D5: Saída digital utilizada para LED de sinalização.

Além de definir as entradas utilizadas para transmissão dos dados, também é necessário especificar como os dados são transmitidos. Conforme subseção 2.10.1, existe uma padronização para comunicação dos dados. Portanto, a seguir são mostradas as particularidades da comunicação entre a PCI e o sistema supervisor. Ao invés de utilizar um protocolo de comunicação já existente para envio dos dados, criou-se um protocolo próprio. Para isto, organizou-se os dados a fim de garantir que as variáveis recebidas pela PCI sejam transmitidas no mesmo pacote para o sistema supervisor. Além disso, também é necessário garantir que nenhum pacote será perdido durante o processo. Para isso, os

dados são escritos na porta serial do Arduino e, por padrão, dados são convertidos para ASCII. Ou seja, cada caractere escrito possui 1 *byte*. A Figura 45 mostra a estrutura de um pacote de dados.

Figura 45 – Estrutura de um pacote de dados enviado da PCI para o sistema supervisorio.

\_/01.15/

<i>Byte</i>	<i>Caractere</i>
1	-
2	/
3	0
4	1
5	.
6	1
7	5
8	/

Fonte: Elaborado pelo autor.

Nesse quadro é observado:

- *Byte* 1: Delimitador inicial. Informa início de um pacote;
- *Bytes* 2 e 8: Delimitador inicial e final de envio dados, respectivamente. Informa início e final da transmissão de dados;
- *Byte* 3: Estado do relé para aumentar nível de tensão do ensaio;
- *Byte* 4: Estado do relé para diminuir nível de tensão do ensaio;
- *Byte* 5: Divisor de dados. Informa término das informações dos relés e início da informação de medição de temperatura;
- *Bytes* 6 e 7: Medição de temperatura.

Diante do modelo OSI, essas são as configurações do protocolo de comunicação:

- Camada física: Baseada em uma conexão serial com *baudrate* de 9600 bps, largura de *bit* de 8 *bits*, *bits* de paridade de 1 bit e de bits de parada 1 *bit*;
- Camada de enlace de dados: protocolo UART.

### 3.2.3 Desenvolvimento do Sistema Supervisorio

O sistema supervisorio foi desenvolvido através da plataforma de desenvolvimento .NET, sendo este um conjunto de ferramentas, bibliotecas e linguagens de programação que permitem a criação de aplicativos para diversas plataformas, como *Windows*, *Web* e

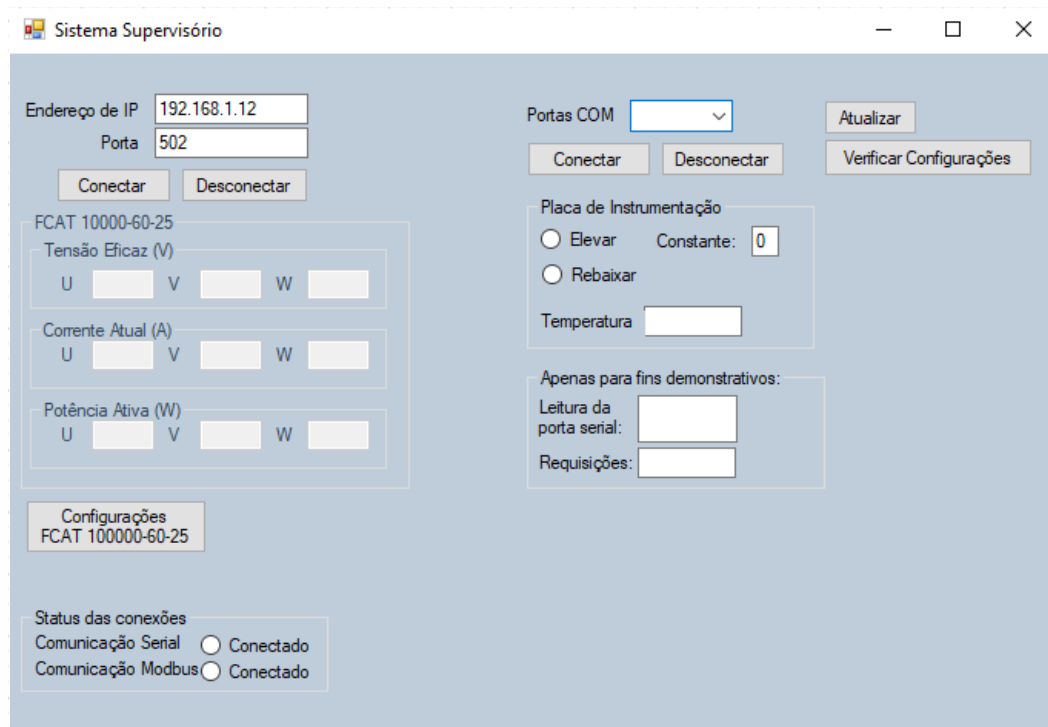


dispositivos móveis. Antes de iniciar o desenvolvimento do sistema supervisorio é fundamental definir sua finalidade. Os itens abaixo listam os objetivos da criação desse sistema supervisorio:

- Apresentar uma plataforma para supervisionar as variáveis do ensaio de elevação de temperatura;
- Manter conexão com a fonte emuladora de tensão;
- Manter conexão com o circuito de instrumentação desenvolvido;
- Enviar as medições em tempo real para a página *Web*.

A partir disso, criou-se o seguinte *layout* para o sistema supervisorio, apresentado na Figura 46. Na parte esquerda do sistema supervisorio vê-se a comunicação com a fonte emuladora de tensão. Esses dados são obtidos através do protocolo Modbus, sendo obtidos em tempo real. Já na parte da direita vê-se conexão com o circuito eletrônico de instrumentação projetado, contendo as informações obtidas pelo aparelho de medição JMAN MONITEK 9451.

Figura 46 – Visualização do Sistema Supervisorio



Fonte: Elaborado pelo autor.

Em paralelo com essas funcionalidades apresentadas está o envio das requisições para a página *Web*. Após o sistema receber os dados da fonte emuladora de tensão e do circuito de instrumentação, os dados são tratados para o envio através do protocolo HTTP.

O código abaixo, escrito em C#, representa o envio da requisição após o tratamento de todos os dados.

```

/// <summary>
/// Envia uma requisição assíncrona para um servidor local, contendo
/// medições de tensão, corrente, potência e temperatura.
/// </summary>
/// <param name="medida">Um array contendo as medições.</param>
/// <returns>A tarefa que representa a operação assíncrona.</returns>
private async Task EscreverURLAsync(ushort[] medida)
{
    // Convertendo as medições em strings
    string TensaoU = String.Join("\r\n", ConverterBytes(medida[0]));
    string TensaoV = String.Join("\r\n", ConverterBytes(medida[1]));
    string TensaoW = String.Join("\r\n", ConverterBytes(medida[2]));

    string CorrenteU = String.Join("\r\n",
        ConverterBytes(medida[3]));
    string CorrenteV = String.Join("\r\n",
        ConverterBytes(medida[4]));
    string CorrenteW = String.Join("\r\n",
        ConverterBytes(medida[5]));

    string PotU = String.Join("\r\n", ConverterBytes(medida[6]));
    string PotV = String.Join("\r\n", ConverterBytes(medida[7]));
    string PotW = String.Join("\r\n", ConverterBytes(medida[8]));

    string Temp = String.Join("\r\n", ConverterBytes(medida[9]));
    string Senha = "senha-definida-pelo-usuario";

    using (HttpClient cliente = new HttpClient())
    {
        // Construindo os dados da requisição no formato apropriado
        var dados_requisicao = new FormUrlEncodedContent(new[]
        {
            new KeyValuePair<string, string>("TensaoU", TensaoU),
            new KeyValuePair<string, string>("TensaoV", TensaoV),
            new KeyValuePair<string, string>("TensaoW", TensaoW),

            new KeyValuePair<string, string>("CorrenteU", CorrenteU),
            new KeyValuePair<string, string>("CorrenteV", CorrenteV),
            new KeyValuePair<string, string>("CorrenteW", CorrenteW),
        })
    }
}

```

```

        new KeyValuePair<string, string>("PotU", PotU),
        new KeyValuePair<string, string>("PotV", PotV),
        new KeyValuePair<string, string>("PotW", PotW),

        new KeyValuePair<string, string>("Temp", Temp),
        new KeyValuePair<string, string>("Senha", Senha),
    });

    // Enviando a requisição assíncrona
    var resposta = await
        cliente.PostAsync("http://localhost/tcc/",
            dados_requisicao);
    }
}

```

No contexto do protocolo Modbus, a conversão de dados é fundamental para manipulação e transmissão dos dados. O método "ConverterBytes", apresentado anteriormente, desempenha um papel relevante nesse processo. Essa sequência de operações tem como objetivo reconstruir o valor de ponto flutuante original a partir dos *bytes* extraídos.

```

/// <summary>
/// Converte um valor ushort em bytes para um valor float.
/// </summary>
/// <param name="registers">Valor ushort a ser convertido.</param>
/// <returns>Valor float resultante da conversão.</returns>
public static float ConvertBytesToFloat(ushort registers)
{
    // Converte o valor 'registers' em um array de bytes
    byte[] bytes = BitConverter.GetBytes(registers);

    // Combina os bytes para obter o valor float
    // Multiplica o byte de índice 1 por 256 para mover seus bits
    // para a esquerda
    // e adiciona o byte de índice 0 para obter o valor final
    // Em seguida, divide o valor resultante por 130f
    float floatValue = ((bytes[1] * 256) + bytes[0]) / 130f;

    // Retorna o valor float resultante
    return floatValue;
}

```

### 3.2.4 Página Web – Comunicação do ensaio de elevação de temperatura com o cliente

Uma página Web para monitoramento de variáveis é uma aplicação Web que permite visualizar e monitorar em tempo real as variações de uma ou mais variáveis, como temperatura, pressão, umidade, dentre outras. Ela pode ser acessada através de um navegador Web e possui uma interface gráfica intuitiva e fácil de usar, com recursos como alertas, notificações e relatórios.

A página desenvolvida tem como objetivo mostrar ao usuário informações pertinentes sobre o ensaio de elevação de temperatura, sendo elas:

- Tensões da fonte Supplier;
- Correntes da fonte Supplier;
- Potências da fonte Supplier;
- Temperatura no topo do óleo do transformador sob ensaio.

O Quadro 9 demonstra visualmente como seriam mostrados os dados de medição na página Web desenvolvida.

Quadro 9 – Exemplo de tabela para visualização dos dados medidos

Fases	Tensão (V)	Corrente (A)	Potência (W)	Temperatura (°C)
Fase 1	Vx	Ix	Px	Temperatura do topo do óleo
Fase 2	Vy	Iy	Py	
Fase 3	Vz	Iz	Pz	

Essas informações são recebidas através do sistema supervisorio, através de *requests*, explicadas na seção anterior.

#### 3.2.4.1 Validação de requisições com o Postman

O Postman é uma ferramenta que permite aos usuários testar APIs (*Application Programming Interfaces*). Dentre suas funcionalidades, dispõe uma série de ferramentas para simular e validar requisições HTTP. Portanto, o Postman será utilizado para validar a construção das requisições que enviarão informações à página Web.

Para fins demonstrativos, foi desenvolvida localmente uma página para enviar as requisições, conforme apresentado na Figura 47.

Figura 47 – Página simulada para validar requisições HTTP

FCAT 10000-60-25

Tensão Eficaz (V)

U 2218 V 2218 W 2218

Corrente Ativa (A)

U 2220 V 2220 W 2220

Potência Ativa (W)

U 2217 V 2217 W 2217

Placa de Instrumentação

Temperatura do topo do óleo

Temperatura 24.8

Fonte: Elaborado pelo autor.

Nesta imagem, destacam-se alguns campos:

- Tensão: Diferença de potencial medida em cada fase da fonte Supplier;
- Corrente: Corrente medida em cada fase da fonte Supplier;
- Potência: Potência dissipada em cada fase da fonte Supplier.

De acordo com os campos configurados acima, deve-se inserir os respectivos campos no Postman. A simulação é apresentada na Figura 48.

Figura 48 – Configuração da simulação utilizando Postman

POST http://localhost/TCC/Index.php?Tensa...

Params Auth Headers (7) Body Pre-req. Tests Settings

Query Params

	Key	Value	...	Bulk Edit
<input checked="" type="checkbox"/>	TensaoU	220		
<input checked="" type="checkbox"/>	TensaoV	220		
<input checked="" type="checkbox"/>	TensaoW	220		
<input checked="" type="checkbox"/>	CorrenteU	10		
<input checked="" type="checkbox"/>	CorrenteV	10		
<input checked="" type="checkbox"/>	CorrenteW	10		
<input checked="" type="checkbox"/>	PotU	2200		
<input checked="" type="checkbox"/>	PotV	2200		
<input checked="" type="checkbox"/>	PotW	2200		
<input checked="" type="checkbox"/>	Temp	26,12		
	Key	Value		

Fonte: Elaborado pelo autor.

Os dados inseridos na tabela são apresentados no Quadro 10.

Quadro 10 – Valores configurados para simulação das requisições

Campo	Tensão (V)	Corrente (A)	Potência (W)
Bobina 1	12	15	18
Bobina 2	13	16	19
Bobina 3	14	17	20

Após configurar a requisição *Web*, é necessário construir o código que recebe esses dados, conforme seção a seguir.

### 3.2.4.2 Programação no lado do servidor – Armazenamento das Informações e Requisições *Web*

Conforme apresentado anteriormente, uma requisição *Web* é uma solicitação que é enviada pelo cliente (geralmente um navegador *Web*) para um servidor *Web* com o objetivo de obter algum tipo de informação ou recurso. Existem vários tipos diferentes de requisições *Web*, incluindo *GET*, *POST*, *PUT* e *DELETE*, cada uma com uma finalidade específica. As requisições *Web* são um dos elementos fundamentais da tecnologia da internet e são amplamente utilizadas em muitas aplicações da *Web*.

Em PHP, a requisições *Web* são acessadas através da variável `$_REQUEST`. Essa variável é um *array* associativo contendo informações de *GET*, *POST* E *COOKIE*. Como prática de programação, é recomendado sempre tratar as requisições recebidas, pois as mesmas podem conter dados maliciosos. Esse tratamento é feito através de filtros, como demonstrado a seguir.

```
<?php
/**
 * Recebe e processa as informações de tensão.
 *
 * @param int $tensaoU Tensão U a ser processada.
 * @param int $tensaoV Tensão V a ser processada.
 * @param int $tensaoW Tensão W a ser processada.
 * @return void
 */
$tensaoU = filter_var($_REQUEST['TensaoU'], FILTER_VALIDATE_INT);
$tensaoV = filter_var($_REQUEST['TensaoV'], FILTER_VALIDATE_INT);
$tensaoW = filter_var($_REQUEST['TensaoW'], FILTER_VALIDATE_INT);

/**
```

```

* Recebe e processa as informações de corrente.
*
* @param int $correnteU Corrente U a ser processada.
* @param int $correnteV Corrente V a ser processada.
* @param int $correnteW Corrente W a ser processada.
*/
$correnteU = filter_var($_REQUEST['CorrenteU'], FILTER_VALIDATE_INT);
$correnteV = filter_var($_REQUEST['CorrenteV'], FILTER_VALIDATE_INT);
$correnteW = filter_var($_REQUEST['CorrenteW'], FILTER_VALIDATE_INT);

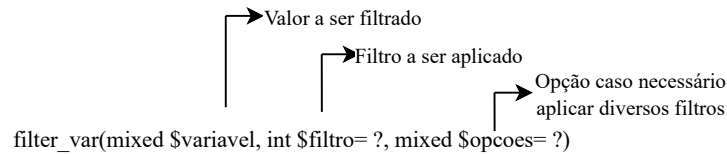
/**
* Recebe e processa as informações de potência.
*
* @param int $potU Potência U a ser processada.
* @param int $potV Potência V a ser processada.
* @param int $potW Potência W a ser processada.
*/
$potU = filter_var($_REQUEST['PotU'], FILTER_VALIDATE_INT);
$potV = filter_var($_REQUEST['PotV'], FILTER_VALIDATE_INT);
$potW = filter_var($_REQUEST['PotW'], FILTER_VALIDATE_INT);

/**
* Recebe e processa as informações da temperatura no topo do óleo.
*
* @param float $Temp Temperatura a ser processada.
*/
$Temp = filter_var($_REQUEST['Temp'], FILTER_VALIDATE_FLOAT);

```

Para fins demonstrativos, a Figura 49 apresenta a configuração da função *filter\_var*, que é utilizada para filtrar e validar dados, garantindo sua conformidade com critérios pre-definidos. Ela permite aplicar filtros embutidos ou criar filtros personalizados para atender requisitos específicos de validação. Ao receber um valor, a função retorna "false" se a validação falhar ou o valor filtrado em caso de sucesso. Por exemplo, se utilizado um filtro do tipo *FILTER\_VALIDATE\_INT*, a função verifica o tipo da variável passada e só retorna o valor se ele for do tipo inteiro.

Figura 49 – Demonstração do funcionamento da função *filter\_var*



Fonte: Elaborado pelo autor.

O código abaixo demonstra uma utilização do filtro, mostrando diferentes valores na entrada e o resultado filtrado na saída. Por sua vez, o Quadro 11 mostra a relação de entrada e saída dos valores.

```
<?php

// Valores de entrada
$entrada1 = 'texto';      // Valor do tipo string
$entrada2 = 2023;        // Valor do tipo inteiro
$entrada3 = 3.1415;      // Valor do tipo ponto flutuante
$entrada4 = '2023';      // Valor do tipo string contendo um número

// Filtrando os valores de entrada

// Saída: false
$teste_string = filter_var($entrada1, FILTER_VALIDATE_INT);

// Saída: 2023
$teste_inteiro = filter_var($entrada2, FILTER_VALIDATE_INT);

// Saída: false
$teste_float = filter_var($entrada3, FILTER_VALIDATE_INT);

// Saída: 2023
$teste_string_numero = filter_var($entrada4, FILTER_VALIDATE_INT);

?>
```



Quadro 11 – Demonstração dos valores filtrados pela função `filter_var` do PHP.

Entrada	Saída
"texto"	<i>false</i>
2023	2023
3,1415	<i>false</i>
"2023"	2023

Originalmente, decidiu-se que todas variáveis seriam armazenadas num banco de dados, como o *MySQL*. Porém, como são poucos dados para registrar, optou-se por implementar arquivos do tipo XML para armazenamento. Por definição, arquivos XML são arquivos de texto que são usados para armazenar e transmitir dados estruturados, usados para armazenar praticamente qualquer tipo de informação, desde dados simples até dados mais complexos.

Como demonstração, foi adicionado abaixo um código que representa um aluno matriculado estruturado em XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<alunos>
  <aluno>
    <nome>Giuliano Francisco Fioreze Cattani</nome>
    <matricula>2016520158</matricula>
    <email>giuliano.cattani@acad.ufsm.br</email>
  </aluno>
</alunos>
```

Na página desenvolvida, para demonstrar a medição em cada bobina do transformador, as medições serão armazenadas em XML com a seguinte configuração:

```
<?xml version="1.0" encoding="UTF-8"?>
<Fonte>
  <fases descrição="fase1">
    <Tensão>V1</Tensão>
    <Corrente>I1</Corrente>
    <Potência>P1</Potência>
  </fases>
  <fases descrição="fase2">
    <Tensão>V2</Tensão>
    <Corrente>I2</Corrente>
    <Potência>P2</Potência>
  </fases>
  <fases descrição="fase3">
    <Tensão>V3</Tensão>
```

```

        <Corrente>I3</Corrente>
        <Potência>P3</Potência>
    </fases>
    <Temperatura>26</Temperatura>
</Fonte>

```

O documento XML é gerado a partir do seguinte código:

```

/**
 * Função para criar um elemento do DOM com um texto
 *
 * @param DOMDocument $dom O objeto DOMDocument
 * @param string $tagName O nome da tag do elemento
 * @param string $text O texto a ser inserido no elemento
 * @return DOMELEMENT O elemento do DOM criado
 */
function criaElementoTexto($dom, $tagName, $text) {
    $element = $dom->createElement($tagName);
    $element->appendChild($dom->createTextNode($text));
    return $element;
}

/**
 * Função para criar um nó de fases com os valores específicos de
    cada fase
 *
 * @param DOMDocument $dom O objeto DOMDocument
 * @param string $fase A descrição da fase
 * @param int $tensao O valor da tensão
 * @param int $corrente O valor da corrente
 * @param int $potencia O valor da potência
 * @return DOMELEMENT O nó de fases criado
 */
function criaNodoFases($dom, $fase, $tensao, $corrente, $potencia) {
    $fasesNode = $dom->createElement('fases');
    $attrFase = $dom->createAttribute('descrição');
    $attrFase->value = $fase;
    $fasesNode->appendChild($attrFase);
    $fasesNode->appendChild(criaElementoTexto($dom, 'Tensão',
        $tensao));
    $fasesNode->appendChild(criaElementoTexto($dom, 'Corrente',
        $corrente));
    $fasesNode->appendChild(criaElementoTexto($dom, 'Potência',

```

```

        $potencia));
    return $fasesNode;
}

$dom = new DOMDocument('1.0', 'UTF-8');
$arquivoXML = 'medidasXML.xml';

$fonteNode = $dom->createElement('Fonte');
$fonteNode->appendChild(criaNodoFases($dom, 'fase1', $tensaoU,
    $correnteU, $potU));
$fonteNode->appendChild(criaNodoFases($dom, 'fase2', $tensaoV,
    $correnteV, $potV));
$fonteNode->appendChild(criaNodoFases($dom, 'fase3', $tensaoW,
    $correnteW, $potW));
$fonteNode->appendChild(criaElementoTexto($dom, 'Temperatura',
    $Temp));
$dom->appendChild($fonteNode);

$dom->save($arquivoXML);

```

Ao final do tratamento de todos os dados fundamentais para a página *Web*, finalmente é possível prosseguir para a construção visual da página *Web*.

### 3.2.4.3 Programação no lado do cliente – construindo a *interface*

Apesar de já apresentar visualmente na Figura 47 como serão mostrados os dados medidos no ensaio, ainda é necessário explicar sua construção. Portanto, essa seção tem como objetivo explicar a construção da programação no lado do cliente, também conhecida como *front-end*. Por questão de objetividade, o foco estará no tratamento dos dados recebidos pelo servidor.

A medição do ensaio de elevação de temperatura constantemente verifica o estado do ensaio e realiza medições. Diante disto, é fundamental criar uma página que atualiza periodicamente os dados enviados pelo servidor. Ou seja, a página construída deve ser dinâmica. Por definição, uma página dinâmica é uma página da *Web* gerada em tempo real pelo servidor quando é acessada pelo usuário. Ou seja, o conteúdo da página pode ser alterado de acordo com as informações que são fornecidas pelo usuário.

Para tornar a página *Web* dinâmica, utiliza-se *HTML*, *CSS* e *JavaScript*. Para lidar com a linguagem de programação JavaScript, faz-se uso de *AJAX (Asynchronous JavaScript and XML)*. Em poucas palavras, é o uso do objeto XMLHttpRequest para se comunicar com os *scripts* do lado do servidor. Ele pode enviar bem como receber informações em

uma variedade de formatos, incluindo JSON, XML, HTML, e até mesmo arquivos de texto. Porém a característica mais atraente do AJAX, é a sua natureza "assíncrona", o que significa que ele pode fazer tudo isso sem a necessidade de atualizar a página. Isso permite atualizar partes de uma página com base em eventos do usuário (MDN, 2022a). O código a seguir mostra a aplicação de AJAX para receber os dados do servidor e mostrar para o cliente de forma dinâmica.

```
<table id="medidas">
  <tr>
    <th>Tensão</th>
    <th>Corrente</th>
    <th>Potência</th>
  </tr>
</table>

<script>
  $(document).ready(function () {
    $.ajax({
      type: 'GET',
      url: 'medidasXML.xml',
      dataType: 'xml',
      success:
        function (xml) {
          let textoHTML = '';
          $(xml).find('fases').each(function () {
            textoHTML +=
              '<tr><td>' +
                $(this).find('Tensão').text() +
              '</td><td>' +
                $(this).find('Corrente').text() +
              '</td><td>' +
                $(this).find('Potência').text() +
              '</td></tr>';
          });
          $('#medidas').append(textoHTML);
          setInterval('location.reload()', 7000);
        }
    });
  });
</script>
```

### 3.3 CONCLUSÕES PARCIAIS

Neste capítulo apresentou-se o desenvolvimento da solução para o presente trabalho. Inicialmente foi mostrado o método de medição adotado, o qual tem como objetivo garantir que não sejam inseridos erros na medição original do equipamento JMAN MONITEK 9451. Em seguida foi apresentado o projeto do circuito eletrônico de instrumentação, bem como o método utilizado para se comunicar com o sistema supervisor. Este, por sua vez, possui comunicação com a PCI, a fonte emuladora de tensão e a página *Web*. Essa comunicação é feita através de diferentes protocolos de comunicação. Por fim, mostrou-se a proposta da página *Web*.

## 4 RESULTADOS EXPERIMENTAIS

Os circuitos de instrumentação propostos foram testados e validados com o auxílio do *software LTSPICE*. A importância de utilizar um software de simulação para validação do projeto está na forma que o *software* analisa o comportamento do circuito, pois, quando realizam-se os cálculos para escolher os componentes a serem utilizados, consideram-se todos elementos como modelos ideais. Por sua vez, o *software* de simulação utiliza componentes próximos do seu modelo real, e como resultado, obtém-se uma simulação próxima do resultado obtidos na prática. Portanto, resultados experimentais são fundamentais na construção de um projeto, pois seus resultados são necessários para validar um projeto feito.

### 4.1 SIMULAÇÃO DO CIRCUITO DE INSTRUMENTAÇÃO DO SENSOR DE TEMPERATURA

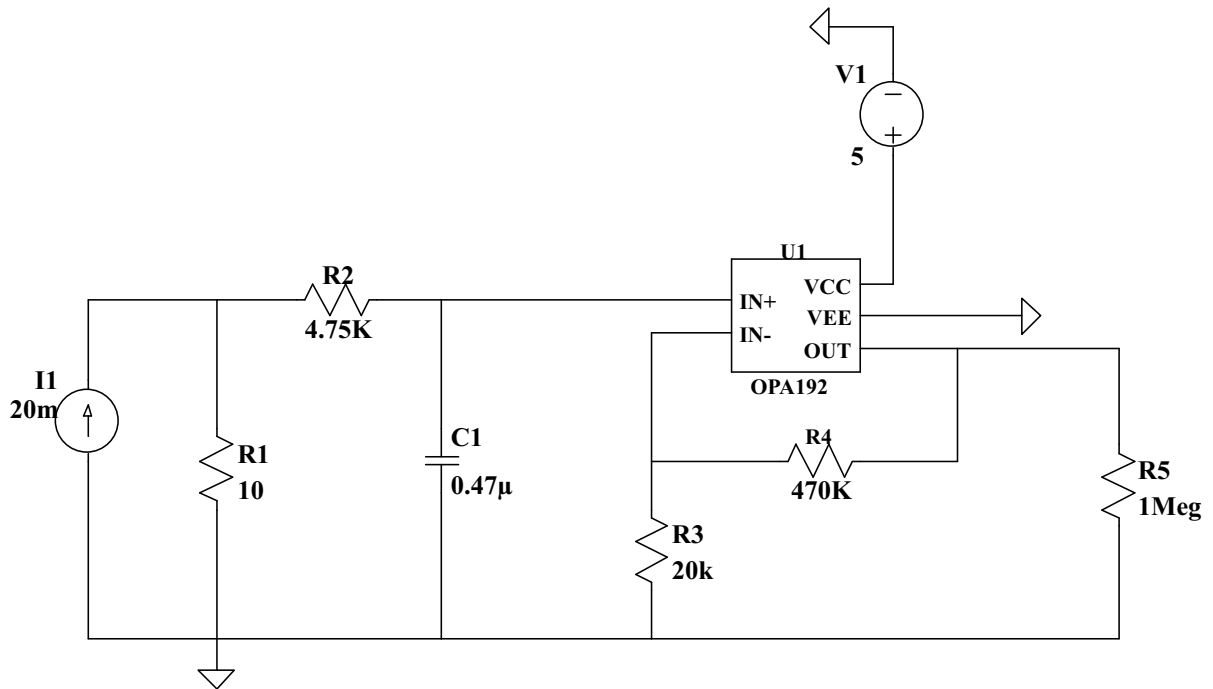
A validação do circuito de instrumentação proposto é composta por duas etapas:

1. Verificar o condicionamento dos sinais de corrente, garantindo a proteção do dispositivo de aquisição de dados;
2. Verificar atenuação do filtro passa-baixa, garantindo sinal sem ruídos.

#### 4.1.1 Validação do condicionamento do sinal de corrente

A simulação é construída a partir do projeto de instrumentação calculado anteriormente. Em seguida, são selecionados os modelos dos componentes eletrônicos que compõem o circuito de instrumentação do sensor de temperatura. Componentes como o amplificador operacional *rail-to-rail* apresentam várias particularidades, portanto, é importante utilizar no *software* de simulação o modelo aproximado do seu funcionamento real. O mesmo acontece para os demais componentes, como os resistores e os capacitores. A simulação construída é apresentada na Figura 50.

Figura 50 – Simulação do circuito de condicionamento do sinal de corrente.

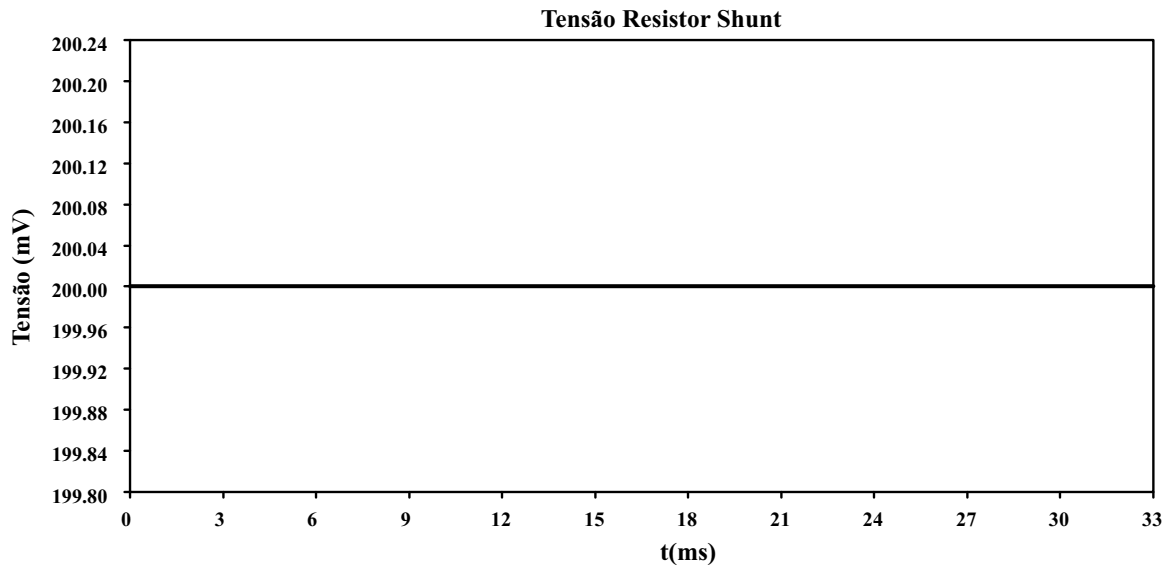


Fonte: Elaborado pelo autor.

O resistor  $R_1$  representa o resistor *shunt* no valor de  $10\ \Omega$ , responsável pela conversão corrente-tensão. O resistor  $R_2$  e capacitor  $C_1$  são os elementos que constituem o filtro RC, encarregado de atenuar ruídos prejudiciais para a medição. O elemento U1 é o amplificador operacional OPA192, modelo aproximado fornecido da própria fabricante *Texas Instruments*. Esse AmpOp compõe a configuração não-inversora do amplificador, junto do resistor  $R_3$  e  $R_4$ , além da fonte  $V_1$ . Na saída do circuito, foi escolhido um resistor  $R_5$  com uma alta impedância, cuja finalidade é simular o dispositivo de aquisição de dados. Os resultados das simulações são apresentados em seguida.

O sinal de entrada, Figura 51, mostra a medição onde é feita a conversão de corrente para tensão. Certificar que esse elemento do circuito está operando de acordo com o projeto é essencial para o funcionamento das etapas seguintes. Conforme mostrado na Figura 51, a simulação é feita considerando um sinal de 20 mA na entrada do circuito.

Figura 51 – Resultados obtidos via simulação nos sinais de entrada do circuito de instrumentação do sensor de temperatura.



Fonte: Elaborado pelo autor.

Pela teoria, nesse caso o resistor *shunt* projetado deve transformar esse sinal de corrente em 200 mV através da Equação (4.1).

$$V_{SHUNT} = I_{PT100} \cdot R_{SHUNT} \quad (4.1)$$

Após a substituição (4.2), obtém-se o resultado em (4.3).

$$V_{SHUNT} = 20mA \cdot 10\Omega \quad (4.2)$$

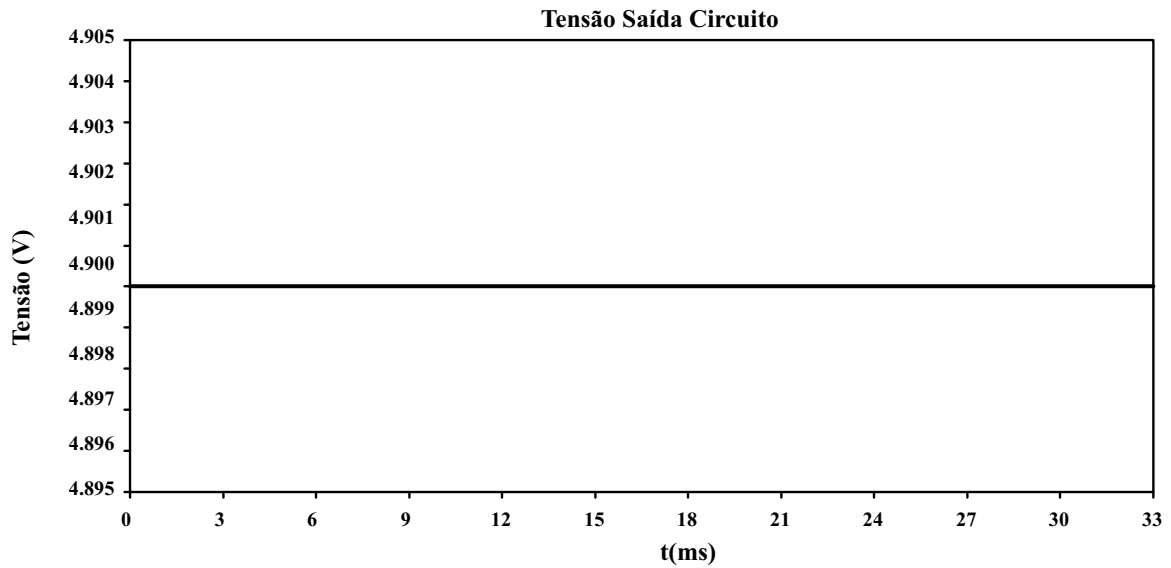
$$V_{SHUNT} = 200mV \quad (4.3)$$

Analisando os resultados obtidos, concluímos que os resultados obtidos via simulação estão de acordo com o projeto executado.

Por sua vez, a Figura 52 mostra o sinal medido na saída do circuito, após condicionamento. O AmpOp OPA192 tem função de condicionar o sinal de entrada para o valor de projetado, assim como proteger sobretensões nocivas para o microcontrolador.



Figura 52 – Resultados obtidos via simulação nos sinais de saída do circuito de instrumentação do sensor de temperatura.



Fonte: Elaborado pelo autor.

Conforme o projeto, esse AmpOp deve ter um ganho de 25. Ou seja, considerando a tensão de entrada  $V_{in}$  em (4.4) e a tensão de saída em  $V_{out}$  (4.5):

$$V_{in} = 200mV \quad (4.4)$$

$$V_{out} = 5V \quad (4.5)$$

Através da Equação (4.6), substituímos os valores na Equação (4.7) e obtemos o resultado em (4.8):

$$V_{out} = V_{in} \cdot \left(1 + \frac{R_f}{R_1}\right) \quad (4.6)$$

$$V_{out} = 200mV \cdot \left(1 + \frac{470K\Omega}{20K\Omega}\right) \quad (4.7)$$

$$V_{out} = 4,90V \approx 5V \quad (4.8)$$

Esse valor aproximado é consequência da escolha dos resistores comerciais, pois dificilmente os valores projetados são comercializados, dessa forma são escolhidos sempre valores aproximados ao projeto.

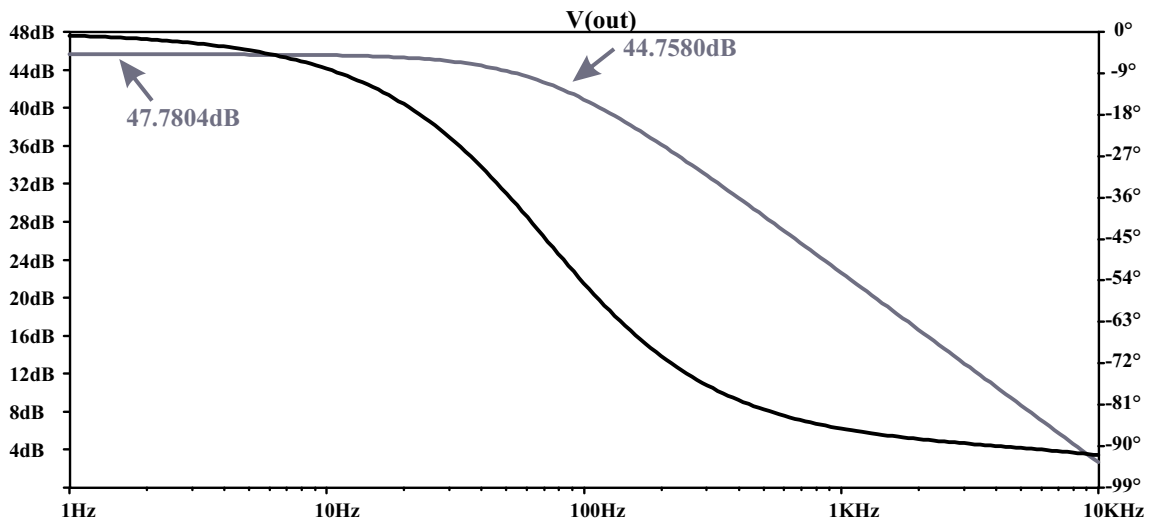
Os resultados são apresentados na Figura 52. É possível concluir então que a simulação está de acordo com os resultados teóricos, validando o projeto executado.

#### 4.1.2 Validação do filtro passa-baixas de atenuação dos ruídos e ganho do sistema

Validar o funcionamento do filtro passa-baixas é importante para a atenuação de ruídos prejudiciais para a medição. A comprovação é feita ao analisar a resposta em frequência do circuito de instrumentação. Ao analisar a resposta em frequência, também são observadas demais informações relevantes para o projeto, como o ganho do circuito. No final desses testes, o circuito de instrumentação projetado é finalmente validado.

O comportamento do filtro passa-baixas é observado a partir do Diagrama de Bode. O ganho unitário do circuito é obtido na frequência zero, conforme apresentado no Capítulo 2. Por sua vez, o ganho do filtro passa-baixas é obtido na frequência de corte projetada. Ou seja, na fase  $-45^\circ$ , o ganho do circuito é de 44,7580dB, enquanto na frequência zero, o ganho é de 47,7804dB. Ambos valores de ganho são evidenciados na Figura 53.

Figura 53 – Simulação da resposta em frequência do circuito de instrumentação do sensor de temperatura.



Fonte: Elaborado pelo autor.

O ganho em tensão é obtido através da Equação (4.9).

$$G_{dB} = 20 \cdot \log_{10} \left( \frac{V_o}{V_{in}} \right) \quad (4.9)$$

Após substituir os valores em (4.10), obtemos o resultado em (4.11).

$$20 \cdot \log_{10} \left( \frac{V_o}{V_{in}} \right) = 47,7804dB \quad (4.10)$$

$$\frac{V_o}{V_{in}} = 244,91 \quad (4.11)$$

Portanto, o ganho do circuito de instrumentação resultou em 244,91. Pela teoria de projeto, o ganho seria de 250. Porém, devido à aproximação de valores dos componentes escolhidos, o valor variou.

## 4.2 PROJETO DA PLACA DE CIRCUITO IMPRESSO

Após a elaboração e validação dos circuitos propostos, utilizando *softwares de simulação*, finalmente é possível projetar a placa de circuito impresso (PCI). Esta placa é o elemento que integra todos os circuitos apresentados nesse projeto.

Placas de circuito impresso são usualmente placas de fenolite ou fibra de vidro cobreadas, sendo algumas de dupla face (*bottom* e *top* cobreados) e outras de face simples (apenas *bottom* cobreado), variando de acordo com a complexidade dos circuitos (FARIAS, 2019).

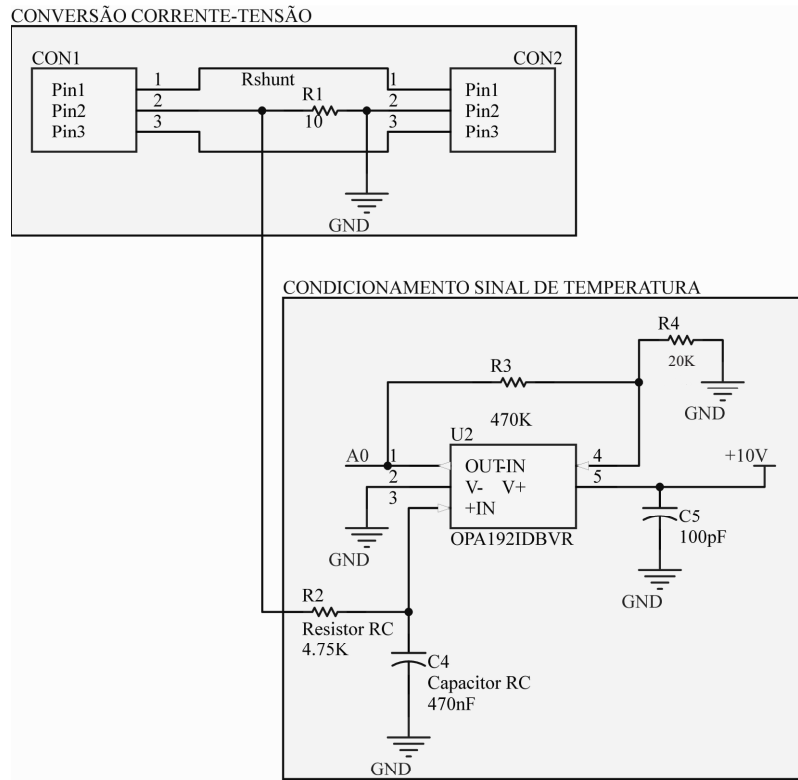
A confecção da PCI foi feita através de um *software* CAD para confecção de placas. O *software* oferece uma biblioteca de componentes eletrônicos para aplicação, como resistores, capacitores e conectores. Ainda é possível importar os componentes a partir de seus fabricantes, contendo suas características reais. Cada componentes possui seu *footprint*, contendo suas dimensões para inserir na PCI.

A primeira etapa na confecção da PCI é desenhar seu esquemático. Ou seja, inserir no *software* os circuitos desenvolvidos. O esquemático será apresentado em diferentes figuras para facilitar a visualização.

### 4.2.1 Esquemático dos circuitos conversão corrente-tensão e condicionamento do sinal de temperatura

A Figura 54 apresenta os circuitos de conversão corrente-tensão e condicionamento do sinal de temperatura. Os componentes CON1 e CON2 representam os conectores Lumberg, utilizados pelo MONITEK JMAN 9451. Conforme mostrado na seção 3, um resistor *shunt* é posicionado entre os conectores, realizando a conversão corrente-tensão. Em seguida, esse sinal é recebido pelo condicionamento do sinal de temperatura. O AmpOp OPA192IDBVR é conectado a entrada analógica A0 do microcontrolador Arduino Nano.

Figura 54 – Esquemático dos circuitos conversão corrente-tensão e condicionamento do sinal de temperatura.

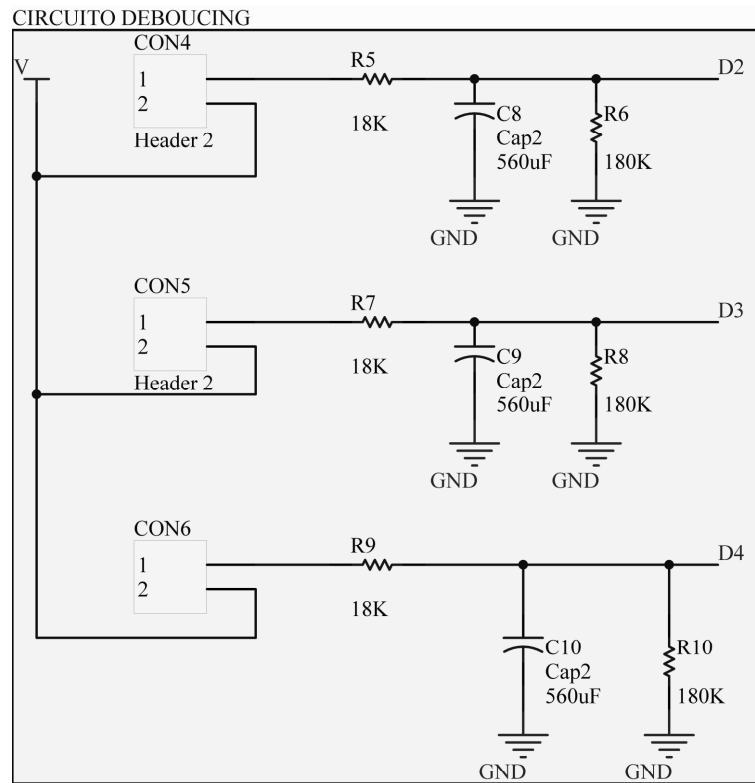


Fonte: Elaborado pelo autor.

#### 4.2.2 Esquemático do circuito de *Debouncing*

Os circuitos anti-repique (*debouncing*) para leitura dos relés presentes no MONITEK JMAN 9451 são mostrados na Figura 55. Os componentes CON4 e CON5 são os conectores para leitura desses relés. Um conector auxiliar CON6 foi adicionado para leitura de um detector de fumaça, ferramenta importante para monitoramento remoto do ensaio. Esses sinais são recebidos pelas entradas digitais D2, D3 e D4 do Arduino Nano.

Figura 55 – Esquemático contendo o circuito projetado para *debouncing* dos relés.

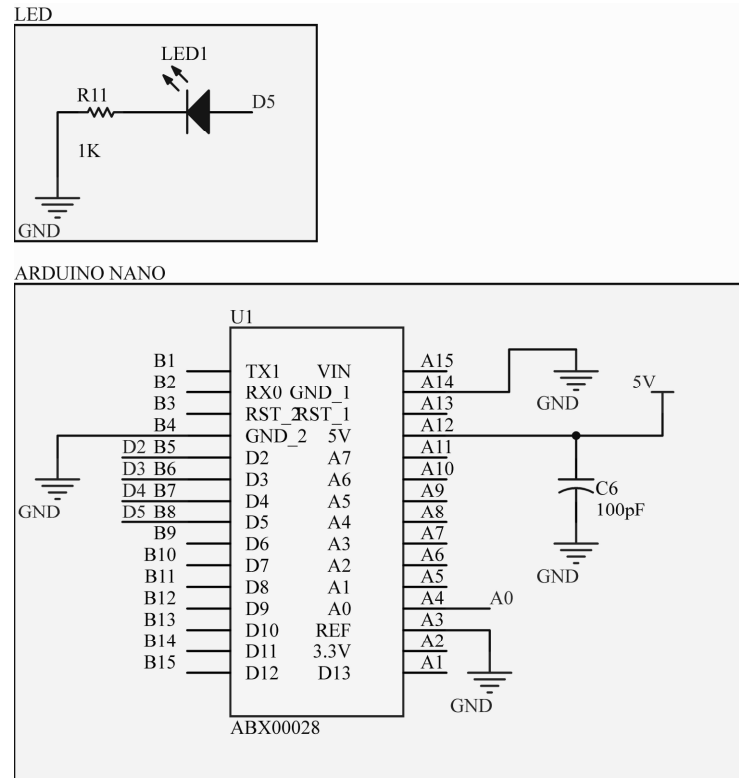


Fonte: Elaborado pelo autor.

#### 4.2.3 Esquemático do microcontrolador Arduino Nano e LED

A Figura 56 mostra o esquemático do microcontrolador Arduino Nano e um LED com a finalidade de sinalizar o funcionamento da PCI. Observa-se que é utilizada a porta A12 do microcontrolador para alimentação dos circuitos do circuito de debouncing dos relés apresentados acima.

Figura 56 – Esquemático contendo as conexões do microcontrolador Arduino Nano e LED para sinalizar operação da PCI.

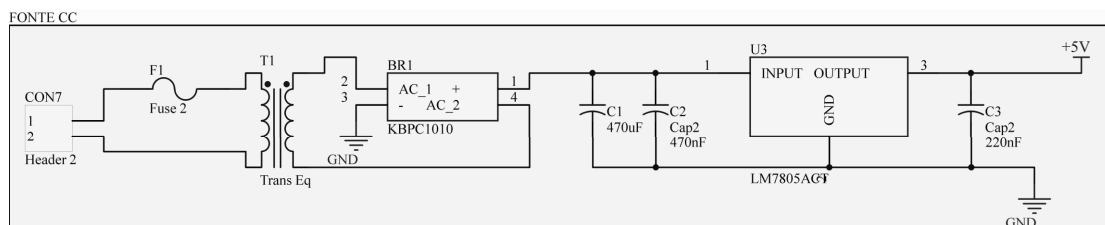


Fonte: Elaborado pelo autor.

#### 4.2.4 Esquemático do circuito da fonte CC

A Figura 57 mostra o esquemático da fonte CC para alimentação do AmpOp, apresentado na seção 3. Observa-se que é através da CON7 que é feita a conexão com a rede 220V.

Figura 57 – Esquemático contendo o circuito da fonte CC para alimentação do AmpOp.

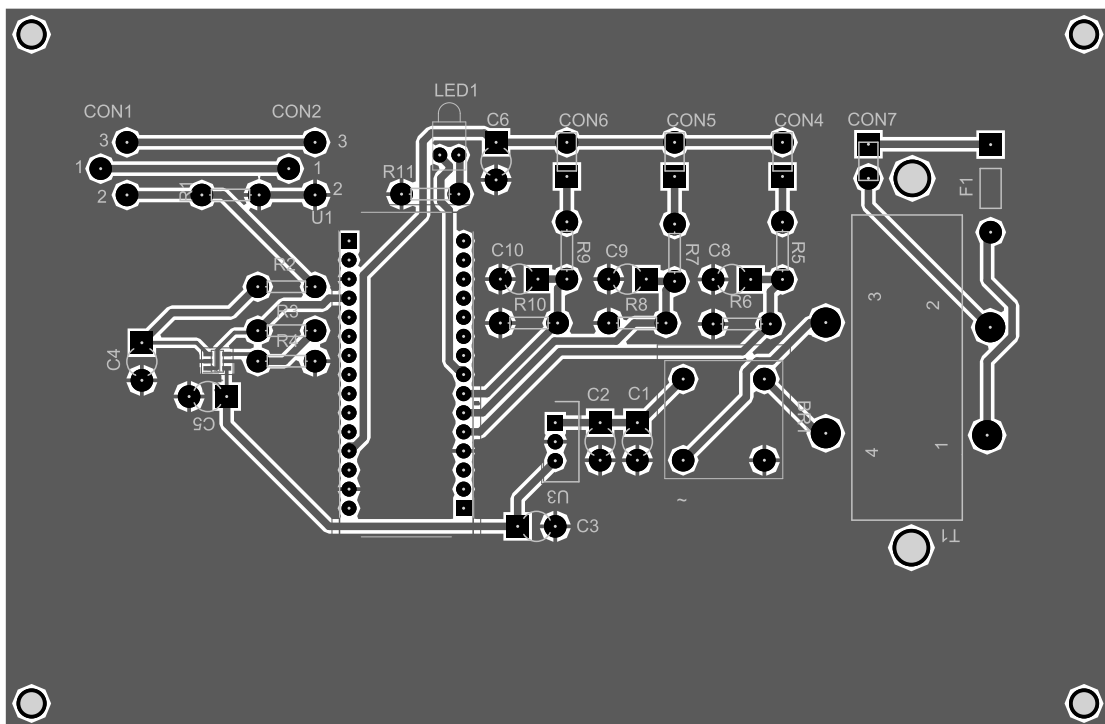


Fonte: Elaborado pelo autor.

#### 4.2.5 Apresentação da PCI finalizada

Após finalizar o esquemático, a próxima etapa é planejar a construção da PCI. Aqui os componentes são distribuídos com a finalidade de facilitar a medição quando o protótipo for implementado. A Figura 58 mostra a distribuição dos componentes na placa, bem como suas trilhas. Todas suas trilhas foram dispostas na parte de baixo da placa (*Bottom Layer*), assim como o AmpOp OPA192, por ser um componente SMD. Os componentes foram posicionados na parte da cima da placa (*Top Layer*).

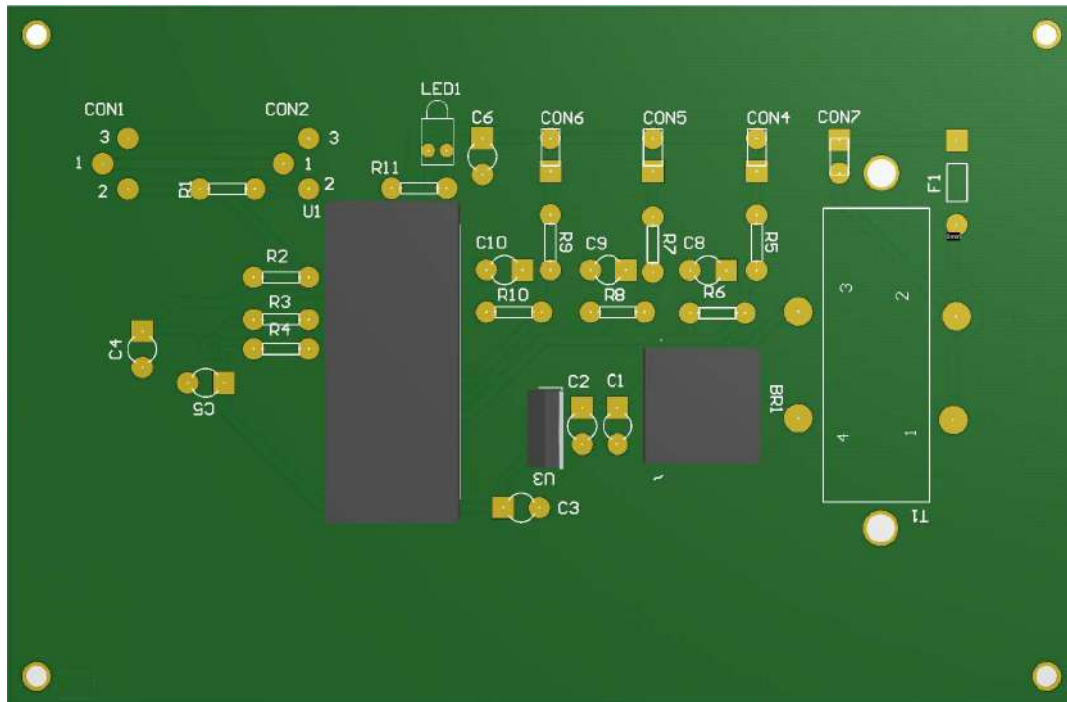
Figura 58 – Vista 2D da PCI projetada no *software* CAD de confecção, contendo todos os circuitos projetados.



Fonte: Elaborado pelo autor.

A Figura 59 mostra a vista superior da PCI. A dimensão resultante da PCI foi de 97x148 mm.

Figura 59 – Top Layer da PCI projetada no software CAD de confecção de placas.



Fonte: Elaborado pelo autor.

#### 4.2.6 Análise da operação da PCI

Este capítulo apresentará os testes realizados para validar a correta operação da placa de circuito impresso (PCI). Entre os testes realizados, estão o teste de continuidade, isolamento, grandeza elétrica e aterramento, que são fundamentais para garantir o correto funcionamento da PCI e identificar possíveis falhas antes da utilização em aplicações práticas.

Conforme descrito previamente, o objetivo da PCI é obter as medidas realizadas pelo equipamento JMAN MONITEK 9451, responsável por realizar as medições do ensaio de elevação de temperatura em transformadores de distribuição. É relevante destacar que a medição realizada pela PCI é caracterizada como uma medição invasiva, o que pode interferir na precisão da medição principal. Por se tratar de uma medição invasiva, essa seção tem como objetivo relatar experimentos práticos com o intuito de validar o circuito de instrumentação projetado.



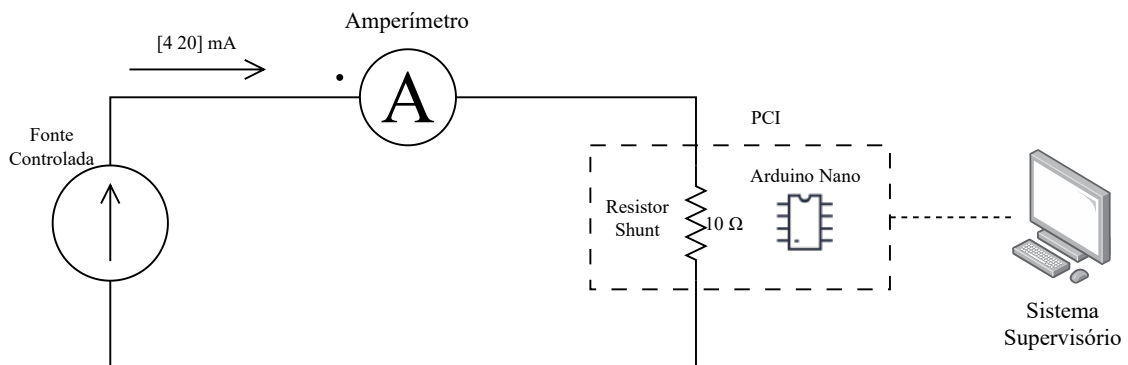
#### 4.2.6.1 Simulação das medições na PCI

Utilizando uma fonte de corrente controlada e um multímetro, foram simulados e medidos os sinais. Os resultados são apresentados no Quadro 12 a seguir, enquanto a Figura 60 mostra um esquemático representando como foram medidos os sinais.

Quadro 12 – Simulação dos sinais medidos pela PCI utilizando fonte de corrente.

Fonte Controlada (mA)	Arduino Nano (Leitura serial A0)
-10,02	9
-5,06	10
0	20
2,08	101
4,05	211
5,98	317
8,06	432
10,02	540
12,1	655
14,08	765
14,95	810
16,09	875
18,05	984
21,08	1023
24,95	1023
30,15	1023

Figura 60 – Esquemático do experimento realizado em laboratório.

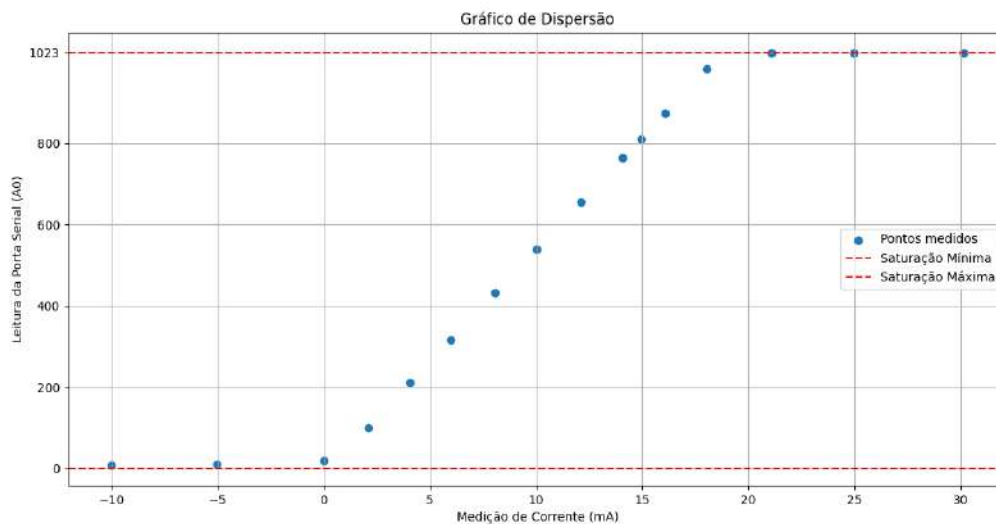


Fonte: Elaborado pelo autor.

Além de ser fundamental para validação da PCI construída, os dados obtidos também foram utilizados para construir diferentes gráficos com o intuito de comprovar que a medição está em conformidade com o projeto. Primeiramente tem-se a Figura 61, representando um gráfico de dispersão dos valores medidos. O destaque principal desse gráfico é demonstrado por meio de duas setas, que apontam para os pontos de saturação das medições. De acordo com a descrição do projeto, foram realizadas leituras de sinais

variando de 4 a 20 mA, porém, para fins de validação, foram medidos também diferentes pontos. Em outras palavras, quando o sinal de entrada era de 4 mA, o microcontrolador registrava uma leitura de 0, enquanto que 20 mA correspondia ao limite máximo da escala de leitura. Considerando que o microcontrolador possui uma precisão de 10 *bits*, o valor máximo da escala seria 1023 (resolução do conversor AD de 10 *bits*). Conforme representado pelo gráfico, pode-se constatar que as condições estabelecidas pelo projeto foram atendidas, uma vez que qualquer valor acima ou abaixo dos limites especificados resulta em saturação na leitura do microcontrolador.

Figura 61 – Gráfico relacionando medição da entrada analógica A0 do microcontrolador Arduino Nano com a medição de corrente em mA.

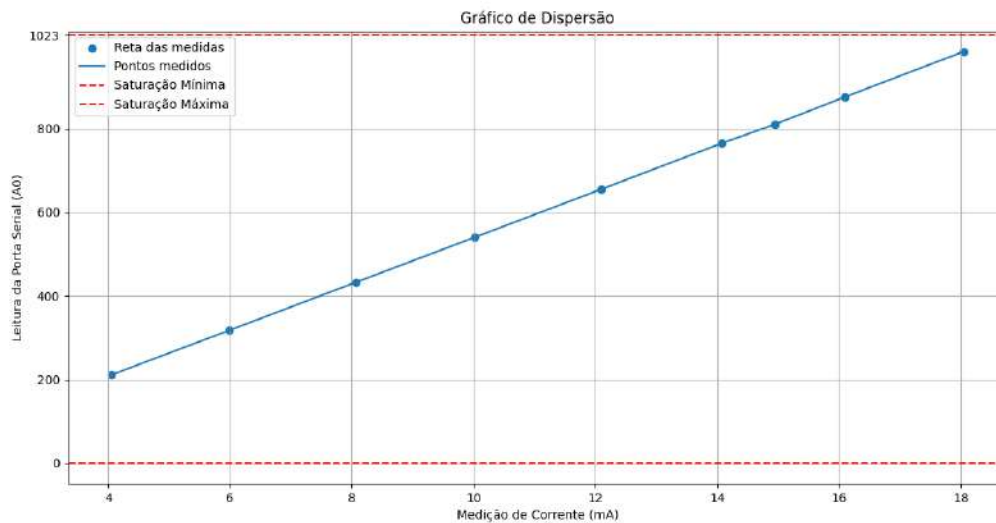


Fonte: Elaborado pelo autor.

Por sua vez, analisa-se na Figura 62 um gráfico linear que mostra exclusivamente os pontos medidos dentro da faixa de 4 a 20 mA. A partir desses dados, é possível obter a Equação (4.12) que representa o padrão de comportamento que relaciona a medição do PT100 com a leitura do microcontrolador.

$$y = 55,17x - 12,741 \quad (4.12)$$

Figura 62 – Gráfico relacionando medição da entrada analógica A0 do microcontrolador Arduino Nano com a medição de corrente em mA.



Fonte: Elaborado pelo autor.

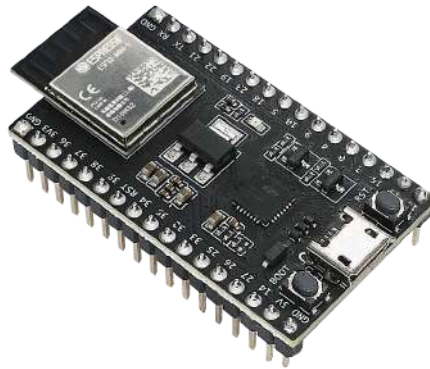
Com isto, finalmente é possível concluir que a PCI desenvolvida está em conformidade com o projeto executado. A próxima seção tem como propósito mostrar apresentar o funcionamento do sistema supervisorio e da página *Web*.

#### 4.2.7 Validação do Sistema Supervisorio e Página *Web*

Para fins experimentais, toda validação do sistema supervisorio e página *Web* foram feitas a partir da emulação de elementos do ensaio. Durante o desenvolvimento do presente trabalho, o laboratório de média tensão esteve participando do processo de acreditação do INMETRO. Devido a este motivo, os equipamentos estavam inacessíveis. Porém, para não impedir o desenvolvimento do trabalho, foram emulados os instrumentos do laboratório, conforme apresentado a seguir.

Conforme definido no projeto, a fonte emuladora de tensão seria controlada por meio do protocolo de comunicação Modbus. Para atender a essa exigência, optou-se por utilizar microcontroladores capazes de suportar o protocolo Modbus. Nesse contexto, o microcontrolador escolhido foi o ESP32, que oferece recursos integrados, tais como conectividade Wi-Fi/Bluetooth, suporte a pilhas TCP/IP e interface UART para comunicação serial. Essas características permitem que o ESP32, apresentado na Figura 63, seja empregado como emulador da fonte emuladora de tensão.

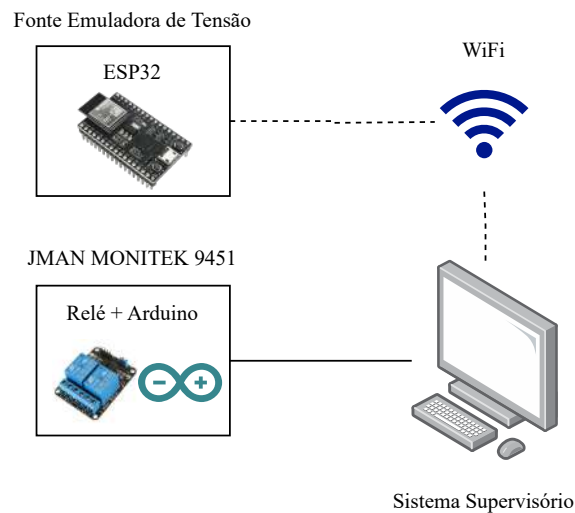
Figura 63 – Microcontrolador ESP32.



Fonte: Adaptado de (ESPRESSIF, 2023).

De forma análoga, também foi emulado o funcionamento do equipamento JMAN MONITEK 9451. Para tal, montou-se em uma *proto-board* um circuito com dois botões com a finalidade de representar o comando dos relés que controlam o varivolt motorizado. Por questões de objetividade, a Figura 64 mostra o diagrama representativo das emulações responsáveis para validação do projeto proposto.

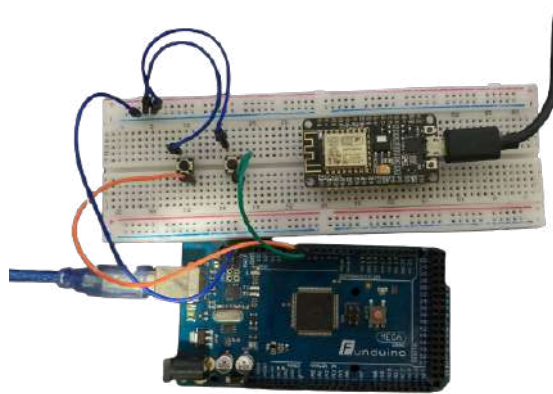
Figura 64 – Representação dos componentes utilizados para emular equipamentos presentes no INRIMT.



Fonte: Elaborado pelo autor.

O circuito para emula a operação do equipamento JMAN MONITEK 9451 e a fonte emuladora de tensão é mostrado na Figura 65.

Figura 65 – Configuração do circuito de emulação para realização de testes.



Fonte: Elaborado pelo autor.

A implementação protocolo de comunicação Modbus no microcontrolador ESP32 foi feita através da linguagem de programação C++. O código abaixo mostra, de forma resumida, a configuração do programa responsável por emular o funcionamento da fonte emuladora de tensão FCATQ 1000- 60-40-PFC55450.

```

ModbusIP mb;

// Constantes dos registradores Modbus
const uint16_t SENSOR_HREG = 0;    // Endereço do primeiro
    registrador
const uint16_t valorInicial = 0;    // Valor inicial dos registradores
const uint16_t quantRegist = 10;    // Número de registradores

// Variáveis de dados
uint16_t tensaoRand1, correnteRand1, potRand1;
uint16_t tensaoRand2, correnteRand2, potRand2;
uint16_t tensaoRand3, correnteRand3, potRand3;
uint16_t tempAmb;
uint16_t valorLido;
long ts;

void setup()
{
    Serial.begin(115200);

    // Conectar à rede Wi-Fi
    WiFi.begin("login_rede", "senha_rede");

```

```

while (WiFi.status() != WL_CONNECTED)
{
    delay(500);
    Serial.print(".");
}

Serial.println("");
Serial.println("WiFi_conectado");
Serial.println("Endereço_IP:_");
Serial.println(WiFi.localIP());

// Inicializar o servidor Modbus e adicionar os registradores
mb.server();
mb.addHreg(SENSOR_HREG, valorInicial, quantRegist);

ts = millis();
}

void loop()
{
    // Lidar com a comunicação Modbus
    mb.task();

    // Ler o valor atual do registrador Modbus: Controle da fonte
    // Constante passada pelo supervisor para incrementar o
    // decrementar a tensão da fonte
    // Seria o valor a ser escrito
    valorLido = mb.Hreg(SENSOR_HREG);

    // Emula os valores de tensão lidos pela fonte emuladora e
    // soma/diminui
    tensaoRand1 = random(210, 220) + valorLido;
    tensaoRand2 = random(210, 220) + valorLido;
    tensaoRand3 = random(210, 220) + valorLido;

    // Emula os valores de corrente lidos pela fonte emuladora
    correnteRand1 = random(5, 10);
    correnteRand2 = random(5, 10);
    correnteRand3 = random(5, 10);

    // Emula os valores de potência lidos pela fonte emuladora
    potRand1 = random(2150, 2200);

```

```
potRand2 = random(2150, 2200);
potRand3 = random(2150, 2200);

// Atualizar os registradores Modbus com os novos valores
mb.Hreg(1, tensaoRand1);
mb.Hreg(2, tensaoRand2);
mb.Hreg(3, tensaoRand3);

mb.Hreg(4, correnteRand1);
mb.Hreg(5, correnteRand2);
mb.Hreg(6, correnteRand3);

mb.Hreg(7, potRand1);
mb.Hreg(8, potRand2);
mb.Hreg(9, potRand3);

}
```

A construção desse código tornou possível estabelecer conexão com o sistema supervisor e realizar a leitura e escrita de registradores Modbus.

A validação do sistema supervisor se provou satisfatória ao término desse experimento, pois, através de emulações, comprovou-se o funcionamento de todas suas funcionalidades, conforme resumido nos itens a seguir:

- Comunicação serial com a PCI: emulação feita com microcontrolador Arduino;
- Comunicação Modbus com a fonte emuladora de tensão: emulação feita com o microcontrolador ESP32;
- Comunicação com a página *Web*: emulação feita ao hospedar localmente uma página *Web*.

### 4.3 CONCLUSÕES PARCIAIS

Este capítulo descreve o processo de validação da solução proposta. Inicialmente, o circuito de instrumentação foi validado utilizando *softwares* de simulação. Em seguida, a PCI foi construída com o auxílio de *softwares* CAD. Finalmente, o protótipo foi validado em laboratório, utilizando uma fonte reguladora para alimentar o circuito com diferentes valores. Essa etapa foi importante para garantir que o circuito opera conforme projetado e está protegido.

Devido a problemas enfrentados no decorrer do trabalho, o acesso aos equipamentos foi limitado. Dessa forma, para garantir o funcionamento do sistema supervisório e da página *Web*, esses equipamentos foram emulados. A conclusão deste capítulo, através dos resultados apresentados, comprova o funcionamento da proposta de automação.



## 5 CONCLUSÃO

Conforme ressaltado no projeto, o processo do ensaio de elevação de temperatura é exaustivo e, devido à sua longa duração, pode induzir a falhas humanas. Os equipamentos utilizados neste ensaio oferecem meios de automatizar o ensaio. A fonte emuladora de tensão, por exemplo, possui interface para conexão remota, através do protocolo Modbus. O equipamento responsável pelas medições do ensaio de elevação de temperatura, JMAN MONITEK 9451, pode controlar, de forma automática, a fonte emuladora de tensão. Dessa forma, o projeto proposto tem como objetivo apresentar uma solução para essa dificuldade enfrentada no processo de ensaios em transformadores de distribuição. A proposta de automação é feita através de integração dos equipamentos envolvidos no ensaio de elevação de temperatura.

Durante a execução do projeto, foi feita uma revisão na bibliografia envolvendo ensaios em transformadores de distribuição. Além disso, o desenvolvimento do projeto possibilitou a integração de diversos conhecimentos obtidos durante o processo de graduação. Foram utilizados conceitos de máquinas elétricas, eletrônica de potência, redes industriais, sistemas de supervisão e desenvolvimento *Web*. Além disso, diversas noções teóricas foram aprofundadas, bem como novos conceitos introduzidos.

Como resultado do estudo, obteve-se uma PCI que, integrado a um sistema supervisor, realiza as medições dos sinais de temperatura, controlando a fonte emuladora de tensão automaticamente, conforme a demanda do ensaio.

### 5.1 DIFICULDADES ENFRENTADAS

A principal dificuldade enfrentada durante a execução do projeto foi em relação a disponibilidade dos equipamentos presentes no INRIMT. Isso ocorreu devido à participação do laboratório no processo de acreditação da INMETRO. Porém, para não atrasar o desenvolvimento do projeto, os equipamentos foram emulados, conforme apresentado no Capítulo 4. Dessa forma, mesmo de forma remota, foi possível obter resultados satisfatórios e finalizar a proposta de automação.

### 5.2 PUBLICAÇÕES VINCULADAS

O desenvolvimento deste Trabalho de Conclusão de Curso resultou na divulgação de trabalhos na Jornada Acadêmica Integrada (JAI) da UFSM:

CATTANI, G. F. F.; BENDER, V. C.; BELTRAME, R. C. **Sistema de monitoramento de grandezas elétricas e térmicas para ensaio de elevação de temperatura de transformadores de distribuição.** In: Anais 35° Jornada Acadêmica Integrada. 2020.

CATTANI, G. F. F.; BELTRAME, R. C.; MARCHESAN, T. B. **Projeto de um circuito eletrônico para automação do ensaio de elevação de temperatura de transformadores de distribuição.** In: Anais 36° Jornada Acadêmica Integrada. 2021.

### 5.3 PROPOSTA DE CONTINUIDADE

O presente trabalho abordou um problema enfrentado pelo INRIMT ao executar o ensaio de elevação de temperatura em transformadores de distribuição. Os resultados obtidos foram satisfatórios, solucionando assim o problema descrito. No entanto, ainda há tópicos que permitem melhorias.

- Implementar o protocolo Modbus para comunicação da PCI com o sistema supervi-sório, padronizando assim a comunicação dos equipamentos utilizados no ensaio;
- Substituir XML por um banco de dados para armazenamento dos dados;
- Histórico de medidas mostradas na página *Web*;
- Possibilidade de interromper o ensaio através da página *Web*, caso detectada alguma anormalidade.

## REFERÊNCIAS

ACROMAG. **Introduction to MODBUS TCP/IP**. s.d. Acessado em 20 de novembro de 2021. Disponível em: <<http://www.supplier.ind.br/produto/fontes-ca-trifasicas/8/fonte-ca-trifasica/213>>.

AGUIRRE, L. A. **Fundamentos de Instrumentação**. 1. ed. [S.l.]: Pearson, 2013. ISBN 9788581431833.

Agência Nacional de Energia Elétrica. **Perdas no sistema elétrico brasileiro**. 2023. Disponível em: <<https://www.aneel.gov.br/perdas-no-sistema-eletrico-brasileiro>>.

ARDUINO. **Arduino Nano – User Manual**. 2008. Acessado em 24 de Outubro de 2022. Disponível em: <<https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf>>.

Arduino. **ARDUINO NANO**. s.d. Acessado em 30 de julho de 2021. Disponível em: <<https://store.arduino.cc/usa/arduino-nano>>.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 5356**: Transformadores de potência — requisitos. Rio de Janeiro, 2010. 31 p. Acessado em 31 de março de 2021.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 5356-2**: Transformadores de potência. parte 2: Aquecimento. Rio de Janeiro, 2010. 31 p. Acessado em 31 de março de 2021.

BOLTON, W. **Instrumentation and Control Systems**. 2. ed. Amsterdam: Newnes, 2015. ISBN 9780081006214.

BOYLESTAD, R. L.; NASHELKY, L. **Dispositivos eletrônicos e teoria de circuitos**. São Paulo: Prentice-Hall do Brasil, 1984. v. 6. ISBN 8587918222.

BRASIL. Lei nº 10.295, de 17 de outubro de 2001. dispõe sobre a política nacional de conservação e uso racional de energia e dá outras providências. p. 1, 2001.

BROWN, D. *et al.* **Manual PHP**. s.d. Acessado em 15 de janeiro de 2022. Disponível em: <<https://www.php.net/manual/en/preface.php>>.

CARCANO, A. **Advanced security aspects on Industrial Control Network**. 2013. Tese (Doutorado) — Università degli Studi dell'Insubria, Itália, 2013.

CARLEN, M. *et al.* Ultra high efficiency distribution transformers. In: **IEEE PES T&D 2010**. [S.l.]: IEEE, 2010. p. 1–7. ISBN 9781424465477.

CHIPKIN. **HOW REAL (FLOATING POINT) AND 32-BIT DATA IS ENCODED IN MODBUS RTU MESSAGES**. 2022. Acessado em 13 de julho de 2022. Disponível em: <<https://www.techopedia.com/definition/23889/web-development>>.

CHRISTOFFERSEN, J. **Switch Bounce and How to Deal with It**. 2015. Acessado em 15 de março de 2021. Disponível em: <<https://www.allaboutcircuits.com/technical-articles/switch-bounce-how-to-deal-with-it/>>.

CLEMENTE, M. **O que é e como usar uma Query**. 2022. Acessado em 03 de fevereiro de 2022. Disponível em: <<https://rockcontent.com/br/blog/query/>>.

DEWESOFT. **Como medir a temperatura com sensores RTD**. 2021. Acessado em 23 de julho de 2021. Disponível em: <<https://dewesoft.com/br/aquisicao-de-dados/medir-temperatura-com-rtd-sensores>>.

DJIEV, S. **Industrial Networks for Communication and Control**. s.d. Acessado em 14 de novembro de 2021. Disponível em: <[https://data.kemt.fei.tuke.sk/SK\\_rozhrania/en/industrial%20networks.pdf](https://data.kemt.fei.tuke.sk/SK_rozhrania/en/industrial%20networks.pdf)>.

ESPRESSIF. **ESP32 - DevKitM**. 2023. Acessado em 10 de Março de 2023. Disponível em: <<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/user-guide-devkitm-1.html>>.

FARIAS, M. L. **Unidade de Condicionamento de Sinais para Laboratório de Ensaios em Transformadores de Distribuição**. 2019. Monografia (Trabalho de Conclusão de Curso) Curso de Graduação em Engenharia Elétrica, Universidade Federal de Santa Maria, Santa Maria, 2019.

FARIAS, M. L. *et al.* Unidade de condicionamento de sinais para laboratório de ensaios em transformadores de distribuição. In: **Anais do Congresso Brasileiro de Automática 2020**. [S.l.]: sbabra, 2020. v. 2.

FRANCHI, C. M. **Instrumentação de Processos Industriais - Princípios e Aplicações**. 1. ed. [S.l.]: Érica, 2018. ISBN 9788536512174.

HERNANDEZ, C.; ARJONA, M. A. Design of an efficient distribution transformer based on an expert system and FE. In: **The XIX International Conference on Electrical Machines - ICEM 2010**. [S.l.]: IEEE, 2010. p. 1–5.

INSTITUTO NACIONAL DE METROLOGIA, NORMALIZAÇÃO E QUALIDADE INDUSTRIAL. Ministério do Desenvolvimento, Indústria e Comércio Exterior. Portaria nº 510, de 07 de novembro de 2016. Ajustes e Esclarecimentos à Portaria Inmetro n.º 378/2010. p. 15, 2016.

INSTITUTO NACIONAL DE METROLOGIA, NORMALIZAÇÃO E QUALIDADE INDUSTRIAL. Ministério do Desenvolvimento, Indústria e Comércio Exterior. Portaria nº 378, de 28 de setembro de 2010. Aprova os Requisitos de Avaliação da Conformidade para Transformadores de Distribuição em Líquido Isolante. p. 21, 2010.

JR., S. L. S.; SILVA, R. A. **Automação e instrumentação industrial com Arduino: teoria e projetos**. [S.l.]: Érica, 2015.

JUNIOR, M. B. F. **Noções de XML**. s. d. Acessado em 12 de julho de 2022. Disponível em: <[https://www.gta.ufrj.br/grad/00\\_1/miguel/index.html](https://www.gta.ufrj.br/grad/00_1/miguel/index.html)>.

LUMBERG. **Circular connectors with threaded joint M16 acc. to IEC 61076-2-106, IP40/IP68**. [S.l.], 2021. Acessado em 5 de maio de 2021. Disponível em: <<https://www.lumberg.com/en/products/product/SV>>.

MALVINO, D. B. A. **Electronic Principles**. [S.l.]: MCGRAW HILL BOOK CO, 2015. ISBN 0073373885.

MDN. **AJAX**. 2022. Acessado em 13 de dezembro de 2022. Disponível em: <[https://developer.mozilla.org/pt-BR/docs/Web/Guide/AJAX/Getting\\_Started](https://developer.mozilla.org/pt-BR/docs/Web/Guide/AJAX/Getting_Started)>.

MDN. **Elementos HTML**. 2022. Acessado em 15 de janeiro de 2022. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/HTML/Element>>.

MDN. **Introdução ao lado servidor**. 2022. Acessado em 15 de janeiro de 2022. Disponível em: <[https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/First\\_steps/Introduction](https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/First_steps/Introduction)>.

MINISTÉRIO DE MINAS E ENERGIA. Ministério de Minas e Energia. Portaria interministerial nº 104, de 22 de março de 2013. Aprova a Regulamentação Específica de Transformadores de Distribuição em Líquido Isolante e o seu respectivo Programa de Metas na forma constante dos Anexos I e II à presente Portaria. p. 7, 2013.

MIRANDA, R. O. de. **Contribuição à Instalação de um Laboratório de Ensaios em Transformadores de Distribuição no Instituto de Redes Inteligentes**. 2019. Dissertação (Mestrado) — Universidade Federal de Santa Maria, Santa Maria, 2019.

MODBUS. **Modbus Application Protocol V1.1b3**. [S.l.], 2023. Acessado em 12 de junho de 2023. Disponível em: <[https://modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b3.pdf](https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf)>.

MUSSOI, F. L. R. **Respostas em Frequência: filtros passivos**. 2004. Centro Federal de Educação Tecnológica de Santa Catarina, Gerência Educacional de Eletrônica. Acessado em 30 de março de 2021. Disponível em: <<https://intranet.ctism.ufsm.br/gsec/Apostilas/filtropassivo.pdf>>.

National Instruments. **O protocolo Modbus em detalhes**. 2021. Acessado em 12 de novembro de 2021. Disponível em: <<https://www.ni.com/pt-br/innovations/white-papers/14/the-modbus-protocol-in-depth.html>>.

NOGUEIRA, T. A. **Redes de Comunicação para Sistemas de Automação Industrial**. 2019. Monografia de Graduação em Engenharia de Controle e Automação, Universidade Federal de Ouro Preto, Minas Gerais, 2019.

ORACLE. **Banco de dados definido**. 2022. Acessado em 15 de janeiro de 2022. Disponível em: <<https://www.oracle.com/br/database/what-is-database/>>.

PERTEENCE, A. **Amplificadores Operacionais e Filtros Ativos-8**. 8. ed. [S.l.]: Bookman, 2015.

PR Eletronics. **2-wire programmable transmitter - 5333A**. [S.l.], s.d. Acessado em 31 de março de 2021. Disponível em: <[https://technosupply.com.br/documentos/129\\_5333A-Datasheet.pdf](https://technosupply.com.br/documentos/129_5333A-Datasheet.pdf)>.

STEMMER, M. R. **Das 5331- Sistemas Distribuídos e Redes de Computadores para Controle e Automação Industrial**. 2001. Acessado em 13 de novembro de 2021. Disponível em: <[http://alvarestech.com/temp/simprebal/Relatorios\\_Tecnicos-Publicacoes-Dissertacoes/docs/cursos/Aula6-Apostila-Sistemas\\_Distribuidos\\_E\\_Redres\\_De\\_Computadores\\_Para\\_Controlre.pdf](http://alvarestech.com/temp/simprebal/Relatorios_Tecnicos-Publicacoes-Dissertacoes/docs/cursos/Aula6-Apostila-Sistemas_Distribuidos_E_Redres_De_Computadores_Para_Controlre.pdf)>.

SUPPLIER. **Fonte CA Trifásica**. s.d. Acessado em 31 de março de 2021. Disponível em: <<http://www.supplier.ind.br/produto/fontes-ca-trifasicas/8/fonte-ca-trifasica/213>>.

SUPPLIER. **MANUAL DA COMUNICAÇÃO ETHERNET MODBUS TCP**. s.d. Acessado em 31 de março de 2021. Disponível em: <<http://www.supplier.ind.br/produto/fontes-ca-trifasicas/8/fonte-ca-trifasica/213>>.

SUPPLIER. **Manual da Comunicação Ethernet – Modbus TCP**. [S.l.], s.d. Acessado em 05 de abril de 2021.

TANENBAUM, A. S. **Redes de Computadores**. 5. ed. [S.l.]: Pearson, 2011. ISBN 857605924X.

TECHOPEDIA. **What is Web Development?** 2022. Acessado em 21 de agosto de 2022. Disponível em: <<https://www.techopedia.com/definition/23889/web-development>>.

VIANA, G. V. R. **Padrão IEEE 754 para Aritmética Binária de Ponto Flutuante**. s. d. Acessado em 04 de junho de 2022. Disponível em: <<https://www.lia.ufc.br/~valdisio/download/ieee.pdf>>.

VICENTE, J. M. E. **Uma Contribuição à Automação de Ensaios em Transformadores de Potência em Média Tensão**. 2006. Tese (Doutorado) — Universidade Federal de Itajubá, Itajubá, 2006.

WAGNER, B. **Um tour pela linguagem C#**. 2021. Acessado em 30 de novembro de 2021. Disponível em: <<https://docs.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp/>>.