



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA DE AUTOMAÇÃO E SISTEMAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Adrian Willian Frasson

Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor

Florianópolis
2025

Adrian Willian Frasson

Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor

Relatório final da disciplina DAS5511 (Projeto de Fim de Curso) como Trabalho de Conclusão do Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Catarina em Florianópolis.

Orientador: Marcelo De Lellis Costa de Oliveira, Dr.
Supervisor: David Ciro Taborda, Dr.

Florianópolis
2025

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Frasson, Adrian Willian

Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor / Adrian Willian Frasson ; orientador, Marcelo De Lellis Costa de Oliveira, coorientador, David Ciro Tabora, 2025.

90 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia de Controle e Automação, Florianópolis, 2025.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Sistema termodinâmico. 3. Transferência de calor. 4. Fluxo de calor. 5. Simulação. I. Oliveira, Marcelo De Lellis Costa de. II. Tabora, David Ciro. III. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. IV. Título.

Adrian Willian Frasson

Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor

Esta monografia foi julgada no contexto da disciplina DAS5511 (Projeto de Fim de Curso) e aprovada em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação

Florianópolis, 25 de fevereiro de 2025.

Prof. Marcelo De Lellis Costa de Oliveira, Dr.
Coordenador do Curso

Banca Examinadora:



Documento assinado digitalmente
Marcelo de Lellis Costa de Oliveira
Data: 10/03/2025 05:23:15-0300
CPF: ***.180.730-**
Verifique as assinaturas em <https://v.ufsc.br>

Prof. Marcelo De Lellis Costa de Oliveira, Dr.
Orientador
UFSC/CTC/EAS

David Giro Taborda, Dr.
Supervisor
REMA/UFSC

Eng. Diogo Ortiz Machado, Dr.
Avaliador
IFRS

Prof. Hector Bessa Silveira, Dr.
Presidente da Banca
UFSC/CTC/EAS

Este trabalho é dedicado aos meus queridos pais.

AGRADECIMENTOS

Em primeiro lugar, agradeço aos meus pais, por todo o suporte fornecido sem medir esforços, sendo meu porto seguro durante todos esses anos.

À minha companheira, pela compreensão e carinho durante essa jornada, me apoiando e guiando nos momentos de incerteza.

Ao meu supervisor, pela paciência e pelos ensinamentos transmitidos, sem o seu apoio, este trabalho não existiria.

Ao meu orientador, por acreditar no meu potencial desde o início e sempre estar disponível para tirar dúvidas, orientando-me com sabedoria e confiança.

A todos os colegas do laboratório REMA, com os quais tive o prazer de trabalhar e compartilhar experiências.

Por fim, a todas as pessoas que passaram pela minha vida, seja de forma breve ou constante, e que, com suas palavras, gestos e ensinamentos, de alguma forma, impactaram meu crescimento e amadurecimento pessoal e profissional.

*"But there's no sense crying over every mistake.
You just keep on trying till you run out of cake."
(GLaDOS, 2007)*

DECLARAÇÃO DE PUBLICIDADE

Florianópolis, 25 de fevereiro de 2025.

Na condição de representante do REMA (UFSC) na qual o presente trabalho foi realizado, declaro não haver ressalvas quanto ao aspecto de sigilo ou propriedade intelectual sobre as informações contidas neste documento, que impeçam a sua publicação por parte da Universidade Federal de Santa Catarina (UFSC) para acesso pelo público em geral, incluindo a sua disponibilização *online* no Repositório Institucional da Biblioteca Universitária da UFSC. Além disso, declaro ciência de que o autor, na condição de estudante da UFSC, é obrigado a depositar este documento, por se tratar de um Trabalho de Conclusão de Curso, no referido Repositório Institucional, em atendimento à Resolução Normativa n° 126/2019/CUn.

Por estar de acordo com esses termos, subscrevo-me abaixo.



Documento assinado digitalmente

David Ciro Taborda

Data: 10/03/2025 08:29:14-0300

CPF: ***.599.908-**

Verifique as assinaturas em <https://v.ufsc.br>

David Ciro Taborda, Dr.
REMA/UFSC

RESUMO

Na indústria do petróleo, vazamentos acidentais de óleo e derivados ao longo da cadeia de produção, transporte, refino e distribuição representam graves ameaças à saúde humana e ao equilíbrio ambiental, além de acarretarem altos custos para as petrolíferas na recuperação de áreas contaminadas. Uma abordagem promissora para mitigar a contaminação é o *Natural Source Zone Depletion* (NSZD), um processo de longo prazo impulsionado pela biodegradação natural de hidrocarbonetos, que tem demonstrado eficácia comparável a técnicas de remediação ativa, porém com custos significativamente menores. Esse avanço incentivou o desenvolvimento do Sistema Termodinâmico de Monitoramento de Fontes (STMF), uma abordagem inovadora e de baixo custo para o monitoramento em tempo real do processo de NSZD em áreas contaminadas. Utilizando sondas subterrâneas de fluxo de calor integradas a modelos matemáticos, o sistema permite estimar as taxas de depleção do NSZD por meio do balanço termodinâmico do solo. No entanto, a estimativa exata do fluxo de calor no centro da sonda é limitada por questões físicas e instrumentais inerentes à própria sonda, como a distância finita entre os pontos de medição no cálculo do gradiente de temperatura e a resolução limitada dos sensores de temperatura, além de não considerar as propriedades térmicas específicas de cada área analisada, comprometendo os resultados. Para contornar os problemas mencionados, este trabalho propõe o desenvolvimento de um Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor, capaz de caracterizar localmente as sondas utilizadas em STMFs. Esse sistema possibilita a determinação dos parâmetros térmicos do solo no local de interesse, fornecendo uma representação digital detalhada do cenário experimental por meio de simulações numéricas otimizadas experimentalmente. Isso permite o estudo de modelos calibrados de sondas de fluxo de calor, contribuindo para a melhoria das medições. O sistema é composto por um volume de controle isolado termicamente do ambiente, com a capacidade de gerar fluxos de calor controlados no meio. Além disso, inclui um gabinete de controle para o gerenciamento do sistema e coleta dados experimentais, bem como um *software* de interface gráfica para a configuração dos parâmetros dos experimentos. Os resultados obtidos por testes de validação confirmaram o funcionamento do sistema desenvolvido, evidenciando que o projeto atingiu com êxito um mínimo produto viável funcional.

Palavras-chave: Sistema termodinâmico. Transferência de calor. Fluxo de calor. Simulação. Otimização

ABSTRACT

In the oil industry, accidental spills of oil and its derivatives throughout the production, transportation, refining, and distribution chain pose serious threats to human health and environmental balance, as well as result in high costs for oil companies in the recovery of contaminated areas. A promising approach to mitigate contamination is Natural Source Zone Depletion (NSZD), a long-term process driven by the natural biodegradation of hydrocarbons, which has shown effectiveness comparable to active remediation techniques, but at significantly lower costs. This advancement encouraged the development of the Thermodynamic Source Monitoring System (STMF), an innovative and low-cost approach for real-time monitoring of the NSZD process in contaminated areas. By using underground heat flow probes integrated with mathematical models, the system allows for the estimation of NSZD rates through the thermodynamic balance of the soil. However, the exact estimation of the heat flux at the center of the probe is limited by physical and instrumental constraints inherent to the probe itself, such as the finite distance between measurement points in the calculation of the temperature gradient and the limited resolution of temperature sensors. Additionally, it does not take into account the specific thermal properties of each analyzed area, compromising the results. To overcome these issues, this work proposes the development of a Thermodynamic System for the Characterization of Heat Flux Probes, capable of locally characterizing the probes used in STMFs. This system enables the determination of the thermal parameters of the soil at the site of interest, providing a detailed digital representation of the experimental scenario through numerically optimized simulations. This allows the study of calibrated models of heat flux probes, contributing to improved measurements. The system consists of a control volume thermally insulated from the environment, with the capability to generate controlled heat fluxes in the medium. Additionally, it includes a control cabinet for system management and experimental data collection, as well as a graphical interface software for configuring experiment parameters. The results obtained from validation tests confirmed the proper functioning of the developed system, demonstrating that the project successfully achieved a minimum viable functional product.

Keywords: Thermodynamic system. Heat transfer. Heat flow. Simulation. Optimization.

LISTA DE FIGURAS

Figura 1 – Modelo conceitual - Biodegradação do LNAPL por NSZD.	17
Figura 2 – Exemplo de sonda de fluxo de calor utilizada em STMFs.	18
Figura 3 – Núcleo Ressacada de Pesquisas em Meio Ambiente - REMA (UFSC).	21
Figura 4 – Termistor Semitec NTC 103AT-11.	26
Figura 5 – Divisor de tensão com termistor NTC.	27
Figura 6 – Multiplexador 4x1.	28
Figura 7 – Esquemático simplificado de um amplificador de instrumentação.	29
Figura 8 – Estrutura multiprocesso do Electron.	31
Figura 9 – Esquemático do volume de controle.	34
Figura 10 – Revestimento da madeira com película térmica.	35
Figura 11 – Suporte interno do termistor.	37
Figura 12 – Construção do volume de controle.	38
Figura 13 – Fonte chaveada AC/DC 220V/12V 30A.	39
Figura 14 – Módulo regulador de tensão DC/DC LM2596.	39
Figura 15 – Módulo multiplexador CD74HC4067.	40
Figura 16 – Módulo amplificador de instrumentação INA333.	41
Figura 17 – Módulo relé 2 canais.	43
Figura 18 – Sensor de corrente por efeito Hall ACS712-20A.	43
Figura 19 – Visualização dos caminhos da fresa na placa de multiplexação.	44
Figura 20 – Fresadora CNC utilizada para fabricar as PCBs.	45
Figura 21 – Placas cobreadas com as trilhas e furos fresados.	45
Figura 22 – Microcontrolador ESP32 - LILYGO T8 V1.8.	46
Figura 23 – Componentes e PCBs a serem fixados no painel elétrico	47
Figura 24 – Chave Gangorra liga-desliga KCD1.	48
Figura 25 – Curva ADC x tensão do microcontrolador.	49
Figura 26 – Procedimento de calibração dos termistores.	50
Figura 27 – Curva tensão x temperatura dos termistores.	50
Figura 28 – Estados de funcionamento do sistema.	51
Figura 29 – Interface gráfica: Página de controle.	53
Figura 30 – Interface gráfica: Página de monitoramento.	54
Figura 31 – Tratamento da amostra de solo.	55
Figura 32 – Dados experimentais obtidos para um fluxo de calor vertical.	57
Figura 33 – Dados experimentais obtidos para um fluxo de calor horizontal.	58
Figura 34 – Fracionamento do volume de controle em pequenos volumes finitos.	59
Figura 35 – Dados experimentais para o fluxo de calor vertical após aplicação do filtro.	60

Figura 36 – Comparação entre as curvas obtidas via simulação otimizada e as curvas experimental para o fluxo de calor vertical.	62
Figura 37 – Comparação entre o gradiente e o fluxo de calor vertical calculado a partir dos dados experimentais e dos dados simulados no centro da sonda.	63
Figura 38 – Circuito de multiplexação: Conexões.	70
Figura 39 – Circuito de multiplexação: <i>Layout</i>	71
Figura 40 – Circuito de amplificação: Conexões.	72
Figura 41 – Circuito de amplificação: <i>Layout</i>	73
Figura 42 – Circuito de acionamento: Conexões.	74
Figura 43 – Circuito de acionamento: <i>Layout</i>	75

LISTA DE TABELAS

Tabela 1 – Condutividade térmica de alguns materiais.	34
Tabela 2 – Comparação entre os parâmetros térmicos do solo obtidos via otimização e valores médios encontrados na literatura.	61

LISTA DE ABREVIATURAS E SIGLAS

ABS	Acrilonitrila Butadieno Estireno
ADC	<i>Analog-to-Digital Converter</i>
CAD	<i>Computer-Aided Design</i>
CENPES	Centro de Pesquisas, Desenvolvimento e Inovação Leopoldo Américo Miguez de Mello
CNC	Controle Numérico Computadorizado
CONAMA	Conselho Nacional do Meio Ambiente
EPS	<i>Expanded Polystyrene</i>
IPC	<i>Inter-Process Communication</i>
LNAPL	<i>Light Non-Aqueous Phase Liquid</i>
MVP	<i>Minimum Viable Product</i>
NSZD	<i>Natural Source Zone Depletion</i>
PCB	<i>Printed Circuit Board</i>
REMA	Núcleo Ressacada de Pesquisas em Meio Ambiente
STL	<i>Standard Tessellation Language</i>
STMF	Sistema Termodinâmico de Monitoramento de Fontes
UFSC	Universidade Federal de Santa Catarina

SUMÁRIO

1	INTRODUÇÃO	16
1.1	PROBLEMÁTICA	17
1.2	SOLUÇÃO	19
1.3	OBJETIVOS	19
2	O LABORATÓRIO	21
2.1	REMA	21
3	FUNDAMENTAÇÃO TEÓRICA	23
3.1	CAPACIDADE TÉRMICA	23
3.2	CONDUTIVIDADE TÉRMICA	24
3.3	FLUXO DE CALOR	24
3.4	VOLUME DE CONTROLE	25
3.5	COMPONENTES ELETRÔNICOS PERTINENTES	26
3.5.1	Termistor	26
3.5.2	Multiplexador	28
3.5.3	Amplificador de instrumentação	28
3.6	FREERTOS	30
3.7	INTERFACE GRÁFICA DE USUÁRIO	30
3.7.1	Electron	31
4	DESENVOLVIMENTO	33
4.1	VOLUME DE CONTROLE	33
4.1.1	Isolamento térmico	34
4.1.2	Dimensionamento das resistências de aquecimento	35
4.1.3	Modelagem CAD e impressão 3D	37
4.1.4	Construção	38
4.2	GABINETE DE CONTROLE	38
4.2.1	Alimentação do sistema	39
4.2.2	Placa de multiplexação	40
4.2.3	Placa de amplificação	41
4.2.4	Placa de acionamento	42
4.2.5	Elaboração das PCBs	44
4.2.6	Microcontrolador	46
4.2.7	Elaboração do painel elétrico	47
4.2.8	Calibração dos sensores	48
4.2.9	Programação do sistema	51
4.3	INTERFACE GRÁFICA DE USUÁRIO	52
4.3.1	Página de controle	53
4.3.2	Página de monitoramento	54

5	RESULTADOS E AVALIAÇÃO	55
5.1	PREPARAÇÃO DA AMOSTRA	55
5.2	EXPERIMENTO	56
5.3	SIMULAÇÃO E OTIMIZAÇÃO	59
6	CONCLUSÃO	65
	REFERÊNCIAS	66
	APÊNDICE A – CIRCUITO DE MULTIPLEXAÇÃO	70
	APÊNDICE B – CIRCUITO DE AMPLIFICAÇÃO	72
	APÊNDICE C – CIRCUITO DE ACIONAMENTO	74
	APÊNDICE D – CÓDIGO DO MICROCONTROLADOR	76
	APÊNDICE E – ALGORITMO DE SIMULAÇÃO DO FLUXO DE CA- LOR	82
	APÊNDICE F – ALGORITMO DE OTIMIZAÇÃO	87
	ANEXO A – AT THERMISTOR - TEMPERATURA X RESISTÊNCIA	88
	ANEXO B – ESP32 LILYGO T8 V1.8 PINMAP	89

1 INTRODUÇÃO

Na indústria do petróleo, vazamentos acidentais de óleo, derivados e biocombustíveis em áreas de exploração e produção, refinarias, dutos, terminais de combustíveis e postos de serviço constituem ameaças significativas à saúde humana e ao equilíbrio ambiental. Em solo, os impactos desses vazamentos são diversos, prejudicando a fauna e flora do local, causando a morte de organismos essenciais para a manutenção da fertilidade, desestabilizando o equilíbrio ecológico e impedindo a absorção de água e nutrientes pelas plantas (EUZEBIO; SILVEIRA RANGEL; MARQUES, 2019).

Em casos de vazamentos em ambientes terrestres, a recuperação inicial, muitas vezes limitada, não elimina completamente os contaminantes. Por isso, em função da presença de contaminação remanescente, a resolução 420/2009 do Conselho Nacional do Meio Ambiente (CONAMA) estabelece que as áreas impactadas passem a ser classificadas como áreas contaminadas, demandando a implementação de medidas como investigação ambiental, análise de riscos, aplicação de técnicas de remediação e monitoramento contínuo, até que sejam alcançados os critérios de recuperação ambiental estabelecidos pelos órgãos competentes (CONAMA, 2009).

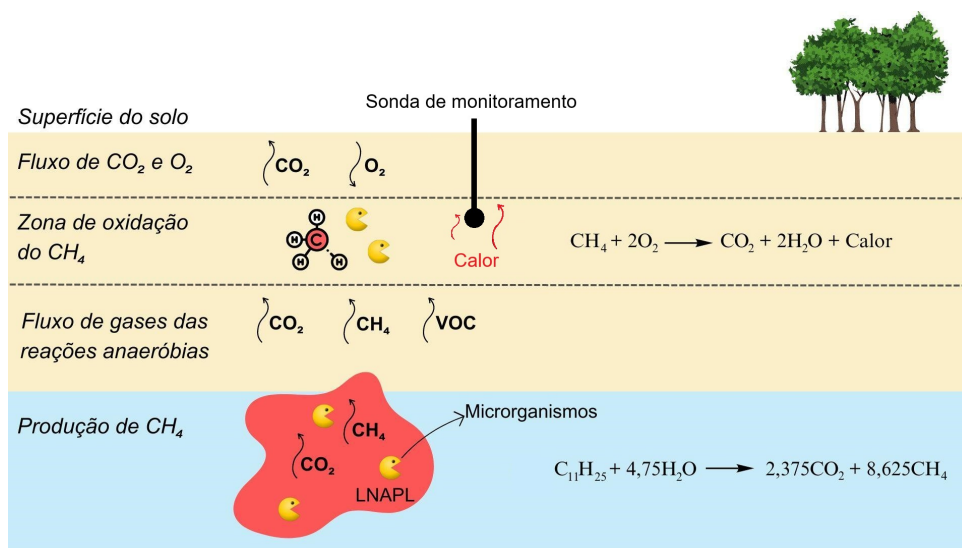
No entanto, observa-se na prática que a recuperação de áreas contaminadas pela indústria do petróleo pode levar anos ou até décadas. Isso se deve às rigorosas exigências de baixas concentrações aceitáveis de contaminantes no ambiente, à elevada complexidade técnica para a extração dos contaminantes e aplicação de substâncias remediadoras, além dos custos significativos e da longa duração das campanhas de monitoramento ambiental necessárias para validar a eficácia das ações.

Um dos maiores desafios nessa recuperação está nas limitações tecnológicas para a remoção de hidrocarbonetos de petróleo em fase livre que se comportam como *Light Non-Aqueous Phase Liquid* (LNAPL) no meio poroso. Esses compostos, ao se infiltrarem no solo e frequentemente alcançarem as águas subterrâneas, tornam-se fontes de contaminação secundária contínua, liberando substâncias tóxicas, mesmo em concentrações extremamente baixas, o que dificulta ainda mais as estratégias de remediação (SCHNEIDER *et al.*, 2020).

Dada essa problemática, uma abordagem conhecida como *Natural Source Zone Depletion* (NSZD) tem recebido cada vez mais atenção e aplicação em escala internacional, destacando-se como uma alternativa eficiente às técnicas ativas de remediação, especialmente em cenários de longo prazo. Estudos indicam que a NSZD é capaz de alcançar resultados semelhantes a outras metodologias, aproveitando processos naturais nas áreas contaminadas. Como ilustrado na Figura 1, nessas regiões, as condições geoquímicas favorecem a biodegradação de hidrocarbonetos do petróleo por arqueas metanogênicas, que metabolizam esses compostos e liberam metano. O metano, por sua vez, se difunde pelo solo até a zona vadosa, onde é oxidado por

bactérias metanotróficas na presença de oxigênio. Durante esse processos, ocorre a liberação de energia na forma de calor, o que provoca um aumento na temperatura do solo na área impactada (GARG *et al.*, 2017).

Figura 1 – Modelo conceitual - Biodegradação do LNAPL por NSZD.



Fonte: Adaptado de REMA (2024).

Diversas técnicas podem ser empregadas para demonstrar e quantificar as taxas de NSZD, sendo uma das mais promissoras, a estimativa baseada no balanço termodinâmico do solo (SWEENEY; RIRIE, 2014). Esse método requer uma descrição detalhada do balanço termodinâmico na região da fonte de contaminação, de modo que, por meio da medição do fluxo de calor, seja possível inferir a cinética de degradação dos contaminantes, levando em consideração tanto a massa inicial quanto a distribuição do LNAPL no solo.

Nesse contexto, foi desenvolvido o Sistema Termodinâmico de Monitoramento de Fontes (STMF), fruto de uma parceria entre o laboratório REMA/UFSC e o centro de pesquisa CENPES/PETROBRAS. Esse sistema inovador possibilita o monitoramento em tempo real e de baixo custo das áreas contaminadas, utilizando sondas subterâneas de fluxo de calor para realizar o balanço termodinâmico do solo, integrado a um modelo matemático que permite estimar as taxas de depleção de NSZD (REMA, 2024).

1.1 PROBLEMÁTICA

O STMF passou, ao longo de seu desenvolvimento, por testes rigorosos e aprimoramentos contínuos em todos os seus componentes, resultando em avanços significativos na sua funcionalidade. Mais recentemente, foi introduzido ao projeto um

novo modelo de sonda de fluxo de calor, ilustrado na Figura 2 e que será objeto de estudo ao longo deste trabalho.

Figura 2 – Exemplo de sonda de fluxo de calor utilizada em STMFs.



Fonte: REMA (2024).

Esse modelo de sonda não mede diretamente o fluxo de calor, mas fornece uma estimativa do vetor gradiente de temperatura em seu centro, calculado a partir da diferença de temperatura registrada pelos sensores dispostos a uma distância finita entre si e arranjados em direções cardeais na sonda. Para determinar o fluxo de calor efetivo, assume-se que o coeficiente de condutividade térmica do meio seja previamente conhecido, permitindo o cálculo do fluxo de calor através da lei de Fourier, dada pelo produto entre o gradiente de temperatura e o coeficiente de condutividade térmica do meio.

Entretanto, estimar o gradiente instantâneo em um centroide por meio da medição simultânea da temperatura em vários pontos apresenta desafios técnicos. Isso ocorre porque o cálculo do gradiente pressupõe uma distância infinitesimal entre os pontos de medição, algo irrealizável na prática devido às limitações de resolução dos sensores de temperatura. Por outro lado, a distância finita entre os pontos de medição traz como consequência a não correspondência entre medidas simultâneas de temperatura entre si, já que as perturbações térmicas se propagam pelo meio a uma velocidade limitada, definida pela difusividade térmica do material.

Outra complicação relevante é que nem sempre são conhecidos os valores dos parâmetros térmicos do solo, como o coeficiente de condutividade térmica e a capacidade térmica volumétrica, sendo necessário utilizar valores médios aproximados. Esses parâmetros são essenciais para o cálculo do fluxo de calor e para a inserção no modelo matemático que determina as taxas de depleção de NSZD. No entanto, essas propriedades podem variar significativamente entre diferentes tipos de solo,

influenciadas por fatores como densidade, teor de umidade e composição mineralógica. Essa variabilidade dificulta a padronização e compromete os resultados, mostrando-se necessário considerar as propriedades específicas de cada área contaminada

1.2 SOLUÇÃO

Buscando contornar os erros provenientes do cálculo do gradiente de temperatura em uma distância finita e da desconsideração dos parâmetros térmicos específicos de cada área analisada, surgiu a ideia de estabelecer uma relação direta entre o histórico de medições de temperatura em curto prazo e o fluxo de calor instantâneo no centro da sonda. Essa abordagem busca aumentar a qualidade das medições de fluxo de calor sem introduzir complexidade adicional à estrutura da sonda e a instrumentação. Além disso, tal metodologia permite a criação de funções de transformação adaptáveis a diferentes geometrias de sondas, construídas com variados materiais e componentes eletrônicos.

Para isso, inicialmente é necessário a criação de um volume de controle onde a sonda será inserida e o volume preenchido com uma amostra do solo de interesse. Esse volume deve permitir a geração controlada de cenários de propagação de calor, sob condições de contorno conhecidas. Em seguida, esses cenários são simulados computacionalmente, otimizando os parâmetros térmicos do solo até que os resultados da simulação estejam consistentes com os dados experimentais. Após alcançar essa consistência, torna-se possível estimar o fluxo de calor em qualquer ponto dentro do volume, inclusive no centro da sonda. Essas informações podem ser utilizadas futuramente para estabelecer relações diretas entre os fluxos simulados e as medições de temperatura instantâneas ou históricas realizadas pela sonda. O produto final desse processo é um modelo digital da sonda, que pode ser empregado para aprimorar as estimativas de fluxo de calor em campo, com base no histórico das temperaturas coletadas. Adicionalmente, a otimização dos parâmetros térmicos do solo permite determinar os coeficientes de condutividade térmica e capacidade térmica volumétrica específicos do solo de interesse.

1.3 OBJETIVOS

O objetivo geral deste trabalho é desenvolver um Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor, um ambiente de testes controlados para estudos de caracterização de sondas de STMFs e das propriedades térmicas do solo.

Os objetivos específicos a serem alcançados com o desenvolvimento deste projeto são:

- Projetar e construir um volume de controle termicamente isolado do ambiente, capaz de criar cenários controlados de propagação de calor no meio;

- Elaborar um gabinete de controle para monitoramento e controle do sistema, condicionamento de sinais e armazenamento de dados experimentais;
- Desenvolver uma interface gráfica de usuário integrada para monitoramento em tempo real dos experimentos e configuração de seus parâmetros;
- Realizar um teste prático utilizando uma amostra de solo de interesse para validar o funcionamento do sistema construído;
- Implementar um algoritmo de simulação e uma técnica de otimização para obter uma representação numérica do cenário experimental.

Ao término deste trabalho, busca-se alcançar um *Minimum Viable Product* (MVP) para aplicação em ambientes laboratoriais, um sistema destinado ao estudo das propriedades térmicas do solo e dos sinais de temperatura das sondas de fluxo de calor em situações com respostas conhecidas ou modeláveis computacionalmente. Esse MVP proporcionará dados que servirão como base para a criação e otimização futura de modelos mais eficientes de sondas.

ORGANIZAÇÃO DO DOCUMENTO

Este documento está dividido em um total de 6 capítulos. Além da introdução apresentada neste primeiro capítulo, os demais conteúdos estão organizados da seguinte maneira:

- **Capítulo 2:** Este segundo capítulo aborda o local de trabalho do autor, destacando sua estrutura, equipe, parcerias e soluções para mitigação de impactos da indústria P&G.;
- **Capítulo 3:** Este capítulo apresenta os fundamentos teóricos essenciais para o trabalho, abrangendo conceitos de transferência de calor, instrumentação eletrônica e tecnologias de *software* utilizadas no projeto;
- **Capítulo 4:** Este capítulo aborda o desenvolvimento dos três componentes principais do sistema: o volume de controle, o gabinete de controle e a interface gráfica de usuário, detalhando suas funções e interações no projeto;
- **Capítulo 5:** O quinto capítulo apresenta a validação do sistema desenvolvido por meio de ensaios com uma amostra de solo. Também será realizado simulações dos cenários experimentais e obtenção do fluxo de calor da sonda via simulação otimizada;
- **Capítulo 6:** No sexto e último capítulo, são apresentadas as conclusões do trabalho, com destaque para os resultados alcançados, as possibilidades de aprimoramento e as sugestões para estudos futuros.

2 O LABORATÓRIO

Este capítulo apresenta brevemente o laboratório onde foi desenvolvido este trabalho, abordando seu surgimento, parcerias, pesquisas e soluções para a mitigação de impactos ambientais e para a recuperação de áreas contaminadas.

2.1 REMA

O Núcleo Ressacada de Pesquisas em Meio Ambiente (REMA), situado na Fazenda Experimental da Universidade Federal de Santa Catarina (UFSC), em Florianópolis, destaca-se como um centro de excelência em pesquisa e inovação tecnológica voltado para o desenvolvimento de soluções ambientais. Inaugurado em 2014, Figura 3, o REMA foi projetado com uma estrutura moderna, ocupando 1.300 metros quadrados, com laboratórios equipados para atender às demandas complexas de pesquisas na área de meio ambiente (NOTÍCIAS DA UFSC, 2014). Essa infraestrutura avançada permite a realização de experimentos de longa duração, fundamentais para o estudo detalhado de áreas contaminadas e para o desenvolvimento de tecnologias eficazes no setor industrial.

Figura 3 – Núcleo Ressacada de Pesquisas em Meio Ambiente - REMA (UFSC).



Fonte: Notícias da UFSC (2016).

Ao longo de sua trajetória, o REMA formou uma equipe multidisciplinar composta por profissionais das engenharias Ambiental, Química, de Controle e Automação, Agrônoma e Biotecnologia. Essa diversidade de especializações tem sido essencial para a abordagem abrangente de problemas ambientais, garantindo o desenvolvimento

de soluções inovadoras e eficientes. A interação entre pesquisadores experientes e alunos da graduação e pós-graduação é outro ponto forte do laboratório, que já capacitou mais de 300 estudantes, contribuindo diretamente para a formação de profissionais altamente qualificados.

A parceria com o Centro de Pesquisas, Desenvolvimento e Inovação Leopoldo Américo Miguez de Mello (CENPES) da Petrobras é um marco na história do REMA. Esse relacionamento de mais de 25 anos viabilizou o desenvolvimento de ferramentas inovadoras, como modelos matemáticos para previsão de cenários de contaminação e *softwares* de análise de riscos ambientais (NOTÍCIAS DA UFSC, 2024). Essas inovações têm contribuído para a mitigação de impactos ambientais e para a recuperação de áreas contaminadas, fortalecendo o compromisso do laboratório com a proteção da biodiversidade e a preservação ambiental.

O laboratório tem se destacado no cenário nacional e internacional por seu trabalho pioneiro na prevenção, gerenciamento e remediação de áreas contaminadas por hidrocarbonetos e biocombustíveis. Tecnologias como o STMF, desenvolvidas no REMA, têm sido amplamente reconhecidas por sua eficácia e menor custo, promovendo a sustentabilidade ambiental e agregando valor à cadeia produtiva do setor energético (SINOVA, 2024). Protocolos elaborados a partir de estudos conduzidos no núcleo foram adotados por órgãos reguladores e empresas, consolidando o REMA como referência técnica e científica no Brasil e no mundo.

Com uma atuação que vai além do âmbito acadêmico, o REMA desempenha um papel crucial no avanço tecnológico do setor de petróleo, gás natural e biocombustíveis. Seus projetos aliam rigor técnico e científico a uma visão prática, gerando impacto positivo tanto para a indústria quanto para a sociedade. Essa trajetória de conquistas e inovação consolida o REMA como um exemplo de como a ciência e a tecnologia podem ser usadas para enfrentar os desafios ambientais e construir um futuro mais sustentável.

3 FUNDAMENTAÇÃO TEÓRICA

Este capítulo discute os fundamentos necessários para a compreensão do trabalho, abrangendo conceitos físicos relacionados à transferência de calor, como propriedades térmicas, fluxo de calor e volume de controle. Além disso, aborda componentes eletrônicos de instrumentação relevantes para o projeto, como termistores, multiplexadores e amplificadores de instrumentação. Também são apresentados o *kernel* FreeRTOS, utilizado para a programação do sistema, e outras tecnologias de *software*, incluindo o desenvolvimento de interfaces gráficas de usuário e o uso de *frameworks* específicos.

3.1 CAPACIDADE TÉRMICA

A capacidade térmica é uma propriedade física que quantifica a quantidade de calor necessária para alterar a temperatura de um corpo em uma unidade de medida (HALLIDAY; RESNICK; WALKER, 2013). Essa grandeza está relacionada à capacidade do material de armazenar energia térmica e é influenciada pela composição e estrutura do objeto. Materiais com alta capacidade térmica exigem mais calor para sofrer variações de temperatura, enquanto aqueles com baixa capacidade térmica aquecem ou resfriam mais rapidamente. A capacidade térmica, denotada por C (J/K), é o limite

$$C = \lim_{\Delta T \rightarrow 0} \frac{\Delta Q}{\Delta T}, \quad (1)$$

onde ΔQ é a quantidade de energia necessária a ser adicionada ao objeto de forma a aumentar sua temperatura em ΔT .

A capacidade térmica volumétrica, por sua vez, é uma extensão do conceito de capacidade térmica, considerando o volume do material. Ela é definida como a quantidade de calor necessária para alterar a temperatura de uma unidade de volume de uma substância por uma unidade de temperatura. É definida por C_V (J/(m³K)), conforme:

$$C_V = \frac{C}{V}. \quad (2)$$

A capacidade térmica é uma propriedade fundamental no estudo de solos contaminados, pois desempenha um papel crucial na modelagem da depleção de contaminantes. Nos experimentos realizados, essa propriedade será determinada juntamente com o coeficiente de condutividade térmica, que será apresentado na próxima seção.

3.2 CONDUTIVIDADE TÉRMICA

A condutividade térmica, geralmente representada por k e de unidade (W/(mK)), é uma propriedade física dos materiais que indica sua capacidade de conduzir calor (ÇENGEL; BOLES, 2013). Em termos físicos, a condutividade térmica está relacionada à maneira como as partículas de um material, sejam átomos, moléculas ou elétrons, interagem para transferir energia térmica. Materiais como metais possuem alta condutividade térmica, devido à alta mobilidade de seus elétrons, enquanto materiais como plásticos e madeiras têm baixa condutividade térmica, atuando como isolantes.

A condutividade térmica, além de ser uma das propriedades térmicas do solo investigadas nos experimentos, desempenha um papel crucial na determinação de fluxos de calor, que será abordado com maior detalhe na próxima seção.

3.3 FLUXO DE CALOR

O fluxo de calor corresponde à quantidade de energia térmica que atravessa uma superfície por unidade de tempo. Trata-se de uma grandeza vetorial, pois possui direção e magnitude, sendo sua direção sempre perpendicular à superfície de transferência e orientada do ponto de maior para o de menor temperatura. Conforme (BERGMAN; LAVINE, 2020), a densidade de fluxo de calor, que representa o fluxo térmico por unidade de área, é descrita pela lei de Fourier para condução térmica, a qual estabelece que o fluxo de calor é proporcional ao gradiente de temperatura, dado por

$$\vec{q} = -k\nabla T, \quad (3)$$

onde $\vec{q}$ é a densidade de fluxo de calor (W/m²), k é a condutividade térmica do meio (W/(mK)), sendo que o sinal negativo indica que o calor é transferido da região quente para a fria e ∇T o vetor gradiente de temperatura (K/m), calculado por

$$\nabla T = \left(\lim_{\Delta x \rightarrow 0} \hat{x} \frac{\Delta T_x}{\Delta x}, \lim_{\Delta y \rightarrow 0} \hat{y} \frac{\Delta T_y}{\Delta y}, \lim_{\Delta z \rightarrow 0} \hat{z} \frac{\Delta T_z}{\Delta z} \right), \quad (4)$$

tal que ΔT_x é a variação de temperatura observada entre dois pontos separados por uma distância Δx na direção $\hat{x}$, e assim para as demais direções.

É importante observar que, ao calcular o gradiente de temperatura, a utilização de uma distância não infinitesimal pode introduzir erros. Isso ocorre porque o gradiente de temperatura, por definição, representa a taxa de variação da temperatura em uma distância infinitesimal, ou seja, em um ponto específico. Em regiões onde o gradiente de temperatura varia rapidamente, o uso de uma distância não infinitesimal pode levar à subestimação ou superestimação do valor real do gradiente, o que impacta diretamente no cálculo do fluxo de calor. Portanto, quanto maior a distância considerada entre os

dois pontos de medição, maior é o erro na aproximação do gradiente. Por outro lado, na prática, ao reduzir a distância entre os pontos de medida, aumenta-se a resolução necessária para distinguir temperaturas muito próximas e aumenta também o potencial de introduzir erros pelo aumento relativo de materiais alheios que compõem a sonda de fluxo de calor em proporção à quantidade de solo que rodeia os termômetros.

3.4 VOLUME DE CONTROLE

O conceito de volume de controle se refere a uma região específica do espaço onde se analisa o comportamento de sistemas físicos, com o objetivo de entender como propriedades como energia, massa e quantidade de movimento variam dentro dessa região. Em termos simples, o volume de controle pode ser entendido como uma caixa imaginária utilizada para estudar processos físicos ou fluxos de energia.

O balanço de energia em um volume de controle envolve a aplicação da primeira lei da termodinâmica, que estabelece que a variação da energia dentro de um volume de controle é igual à energia que entra ou sai, além das transformações de energia internas (como a geração ou consumo), conforme:

$$\Delta E_{VC} = E_{Entra} - E_{Sai} + E_{Gerada} - E_{Consumida}. \quad (5)$$

Ainda, como essa lei deve ser válida para qualquer instante de tempo, pode-se escrevê-la em termos de taxa,

$$\dot{E}_{VC} = \dot{E}_{Entra/Sai} + \dot{E}_{Gerada/Consumida}. \quad (6)$$

Para compreender as interações entre o volume de controle e o ambiente externo, é fundamental introduzir o conceito de condições de contorno. Essas condições definem como as variáveis físicas, como temperatura, velocidade ou pressão, se comportam nas fronteiras do volume analisado, permitindo calcular os fluxos de massa, energia ou momento que entram ou saem do sistema. Existem diferentes tipos de condições de contorno, dependendo do fenômeno físico em análise.

No contexto da transferência de calor, por exemplo, as condições de contorno de Dirichlet especificam diretamente o valor da temperatura nas fronteiras do volume de controle, enquanto as condições de Neumann definem a derivada da temperatura em relação à direção normal à superfície, ou seja, o gradiente de temperatura, que está diretamente relacionado com o fluxo de calor através da lei de Fourier:

$$\nabla T = -\frac{\vec{q}}{k}. \quad (7)$$

Assimilar esses conceitos é essencial para compreender o presente trabalho, uma vez que o sistema termodinâmico desenvolvido é, em essência, um volume de controle. Nele, a geração controlada de fluxos de calor nas superfícies do volume

permite estabelecer condições de contorno conhecidas, possibilitando a simulação computacional do cenário e a análise detalhada dos fenômenos que ocorrem no interior do sistema.

3.5 COMPONENTES ELETRÔNICOS PERTINENTES

Nesta seção serão abordados alguns dos componentes eletrônicos utilizados no desenvolvimento do projeto, detalhando seu funcionamento e destacando a importância de cada um para o sucesso do trabalho.

3.5.1 Termistor

Termistores são dispositivos semicondutores cuja resistência elétrica varia com a temperatura, sendo amplamente utilizados em aplicações de medição e controle térmico. Existem dois tipos principais de termistores: os do tipo NTC (Coeficiente de Temperatura Negativo) e PTC (Coeficiente de Temperatura Positivo). Os NTC apresentam uma diminuição da resistência elétrica à medida que a temperatura aumenta, sendo comumente utilizados em circuitos de proteção contra sobrecarga e sensores de temperatura. Em contraste, os termistores PTC aumentam sua resistência com o aumento da temperatura, sendo úteis em aplicações de proteção contra curto-circuitos e controle de corrente (SCHULTZ, 2015).

A sonda de fluxo de calor apresentada foi projetada com base em um modelo específico de termistor NTC. Por conveniência, optou-se por utilizar neste projeto o mesmo modelo de termistor nos demais sensores de temperatura distribuídos no volume de controle. O modelo em questão é o 103AT-11, Figura 4, fabricado pela Semitec, que se destaca pela ponta vulcanizada, garantindo um alto nível de proteção contra a entrada de água e umidade, características essenciais para aplicações no interior do solo.

Figura 4 – Termistor Semitec NTC 103AT-11.



Fonte: DigiKey (2024).

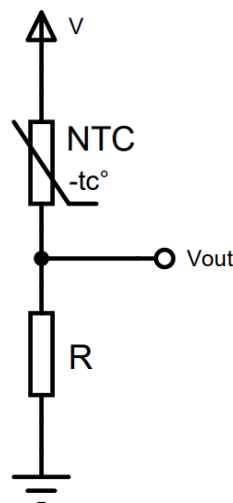
Para relacionar a resistência do termistor com a temperatura, pode-se utilizar a equação do parâmetro beta (β), derivada da equação de *Steinhart–Hart*, que fornece uma aproximação eficiente e amplamente utilizada:

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \ln \frac{R_{NTC}}{R_0}. \quad (8)$$

Nesta equação, T representa a temperatura em Kelvin, T_0 é a temperatura nominal (geralmente 25 °C, equivalente a 298,15 K), β é um coeficiente empírico fornecido pelo fabricante, R_0 corresponde à resistência nominal do termistor a 25 °C, e R_{NTC} é a resistência atual medida no termistor.

Uma configuração muito utilizada para a medição de temperatura com termistores é a implementação do componente em um divisor de tensão, conforme ilustrado na Figura 5.

Figura 5 – Divisor de tensão com termistor NTC.



Fonte: do autor.

Nesta configuração, a tensão de saída V_{out} do divisor é diretamente relacionada à resistência do termistor, R_{NTC} . A partir da medição de V_{out} , é possível calcular de forma simples o valor da resistência do termistor, sendo

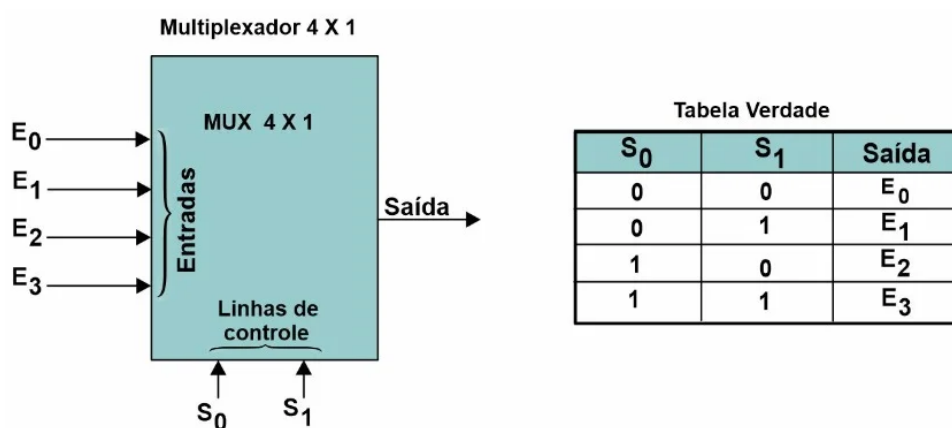
$$R_{NTC} = R \frac{V - V_{out}}{V_{out}}. \quad (9)$$

Embora a configuração de divisor de tensão para o termistor tenha sido utilizada neste trabalho, a aplicação da equação do parâmetro beta não proporcionou resultados satisfatórios. Diante disso, foi necessário desenvolver uma metodologia de calibração dos sensores, com o objetivo de estabelecer uma curva de relação direta entre a temperatura e a tensão medida. Essa abordagem será explicada em detalhes no capítulo de Desenvolvimento.

3.5.2 Multiplexador

Um multiplexador (MUX) é um dispositivo que seleciona um de vários sinais de entrada e o direciona para uma saída com base em sinais de controle. Ele funciona como um "interruptor inteligente", permitindo que múltiplos sinais compartilhem uma única linha de transmissão, otimizando recursos e simplificando o design de circuitos. Em circuitos analógicos, multiplexadores são frequentemente usados para realizar a comutação entre diferentes fontes de sinal, como em sistemas de medição ou em instrumentos eletrônicos. Os multiplexadores desempenham um papel importante na redução de complexidade e no aumento da eficiência de circuitos, permitindo que componentes sejam usados de maneira mais eficiente em sistemas analógicos. A quantidade de entradas de seleção necessárias em um multiplexador depende do número de entradas de dados. Por exemplo, na Figura 6, um multiplexador com 2^n entradas de dados requer n entradas de seleção (MAINI, 2007).

Figura 6 – Multiplexador 4x1.



Fonte: saber360 (2024).

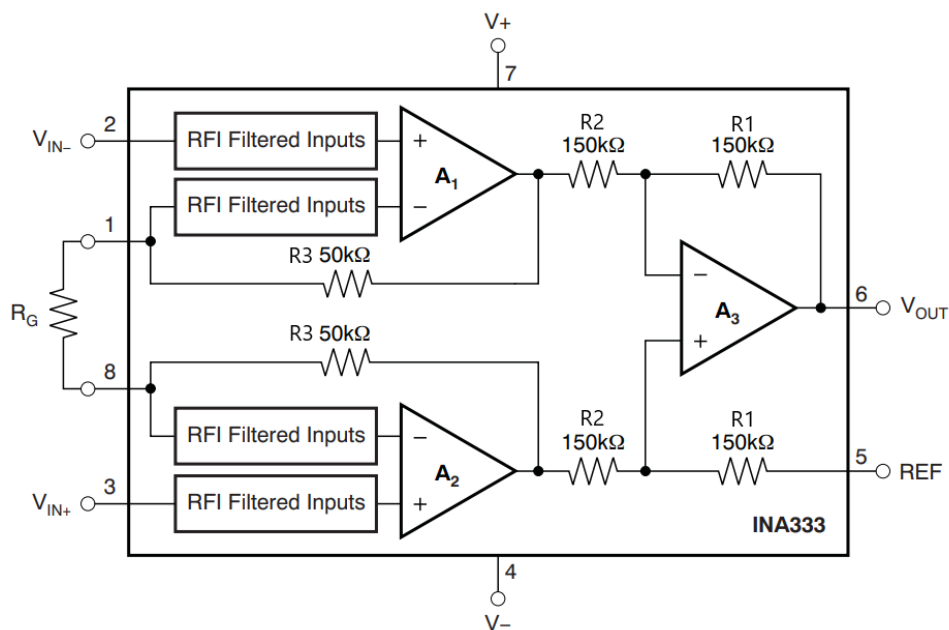
Neste projeto, considerando o grande número de sensores de temperatura envolvidos, a utilização de multiplexadores se mostrou uma solução conveniente e eficiente. Eles permitem a comutação entre os sinais dos diversos termistores, o que, por sua vez, reduz a demanda de canais no microcontrolador. Esse componente não apenas otimiza o uso dos recursos do microcontrolador, mas também simplifica os circuitos, tornando o sistema mais compacto e econômico, sem comprometer o monitoramento das temperaturas.

3.5.3 Amplificador de instrumentação

O amplificador de instrumentação (INA) é um circuito amplificador diferencial projetado para medir pequenos sinais. Geralmente, ele utiliza três amplificadores operacionais: dois configurados como seguidores de tensão em cada uma das entradas

de sinal, o que proporciona alta impedância de entrada e preserva a integridade do sinal original, e um terceiro que realiza a amplificação diferencial propriamente dita (SEDRA; SMITH, 2014). Além disso, alguns INAs possuem uma entrada de referência (REF), que permite aplicar um *offset* à saída, facilitando o ajuste do nível do sinal em aplicações específicas, como adequar o sinal a uma faixa aceitável para as entradas *Analog-to-Digital Converter* (ADC) de microcontroladores. O esquemático de um amplificador de instrumentação utilizado neste projeto, INA333, é apresentado na Figura 7.

Figura 7 – Esquemático simplificado de um amplificador de instrumentação.



Fonte: Adaptado de Texas Instruments (2015).

A configuração de amplificador de instrumentação apresentada na Figura 7 não utiliza dois amplificadores em modo seguidor de tensão, em vez disso, utiliza dois amplificadores não inversores, o que adiciona um estágio de ganho extra ao sinal, dado pela seguinte equação:

$$V_{OUT} = \frac{R_1}{R_2} \left(1 + \frac{2R_3}{R_G} \right) (V_{IN+} - V_{IN-}) + REF. \quad (10)$$

Na prática, o ganho do amplificador de instrumentação é frequentemente ajustado apenas pela seleção do valor da resistência R_G , enquanto as demais resistências são previamente definidas pelo fabricante durante a produção do circuito integrado (CI). Essa abordagem simplifica o ajuste do ganho, que, neste exemplo, permite calcular o sinal de saída como:

$$V_{OUT} = \left(1 + \frac{100k\Omega}{R_G}\right) (V_{IN+} - V_{IN-}) + REF. \quad (11)$$

Neste projeto, amplificadores de instrumentação foram utilizados para amplificar os sinais analógicos provenientes dos termistores, com o objetivo de melhorar a sensibilidade do sistema. Essa amplificação permite que o sinal aproveite plenamente a faixa de operação dos conversores ADC do microcontrolador, aprimorando a resolução das medições. Como resultado, o sistema é capaz de detectar pequenas variações de temperatura, essencial para os objetivos propostos.

3.6 FREERTOS

O FreeRTOS é um *kernel* de sistema operacional em tempo real (RTOS) amplamente utilizado em sistemas embarcados. Ele foi projetado para ser leve e eficiente, sendo ideal para dispositivos com recursos limitados, como microcontroladores. O *kernel* FreeRTOS gerencia a execução de tarefas concorrentes, oferecendo funcionalidades essenciais para sistemas que demandam previsibilidade e desempenho em tempo real.

Uma das principais funcionalidades do FreeRTOS é seu agendador de tarefas, que utiliza diferentes algoritmos de escalonamento, como prioridade fixa, para determinar qual tarefa deve ser executada a cada momento, garantindo que tarefas críticas sejam atendidas dentro de prazos rigorosos. Além disso, o FreeRTOS oferece suporte à comunicação e sincronização entre tarefas, disponibilizando mecanismos como filas, semáforos e mutexes, que facilitam a troca de dados e a coordenação de recursos compartilhados, evitando problemas como condições de corrida e bloqueios, comuns em sistemas multitarefa. Outro aspecto importante é o gerenciamento de tempo, que permite configurar atrasos e temporizadores para as tarefas, essencial para aplicações que demandam controle preciso de eventos baseados no tempo (FREERTOS, 2024).

No desenvolvimento deste projeto, o FreeRTOS foi utilizado no microcontrolador do sistema para gerenciar as tarefas executadas paralelamente, possibilitando uma programação concorrente e a eficiente divisão do trabalho. Além disso, os *software timers* fornecidos pelo *kernel* permitem a temporização precisa do acionamento do mecanismo de geração de calor dentro do volume de controle, garantindo o controle adequado dos experimentos.

3.7 INTERFACE GRÁFICA DE USUÁRIO

Uma interface gráfica de usuário (GUI, do inglês *Graphical User Interface*) é um tipo de interface que permite a interação entre o usuário e sistemas computacionais por meio de elementos visuais, como janelas, ícones, botões e menus. As GUIs têm como objetivo simplificar o uso de *softwares*, tornando-os mais acessíveis a pessoas sem

conhecimentos técnicos avançados, pois substituem comandos de texto por interações visuais e intuitivas. Além disso, essas interfaces desempenham um papel fundamental em aplicações modernas, possibilitando a execução de tarefas de maneira eficiente e agradável ao usuário.

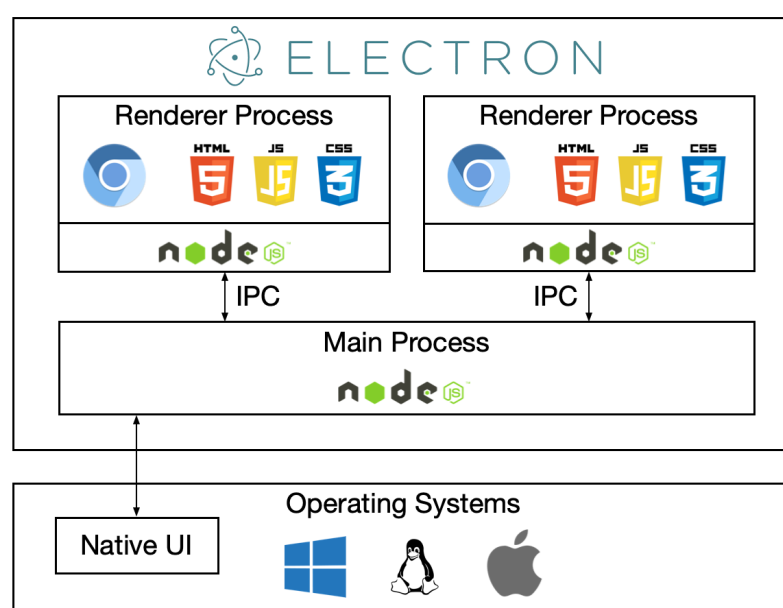
Para a implementação da interface gráfica de usuário deste projeto será empregado o *framework* Electron. A seção seguinte apresentará detalhes sobre a utilização e as funcionalidades dessa tecnologia.

3.7.1 Electron

O Electron é um *framework* de código aberto que permite a criação de aplicações de interface gráfica de usuário em ambientes de *desktop* multiplataforma utilizando tecnologias *web* como HTML, CSS e JavaScript. Desenvolvido e mantido pela OpenJS Foundation, o Electron combina o poder do Chromium para renderização de páginas e do Node.js para funcionalidades de *back-end*, permitindo que desenvolvedores criem aplicativos para Windows, macOS e Linux (ELECTRONJS, 2024). Sua principal vantagem é a possibilidade de usar um único código base para todas as plataformas, o que facilita o desenvolvimento e manutenção de *software*, sem a necessidade de escrever código específico para cada sistema operacional. Ele é amplamente utilizado para criar aplicativos de produtividade, alguns exemplos são o Visual Studio Code e o Slack.

Uma aplicação Electron segue um modelo multiprocesso, Figura 8, dividindo a execução entre dois processos principais: o *Main Process* e o *Renderer Process*.

Figura 8 – Estrutura multiprocesso do Electron.



Fonte: cnblogs (2024).

O *Main Process* é o responsável pela execução do código JavaScript principal, além de gerenciar a janela da aplicação e interagir com a plataforma subjacente, como o sistema operacional. Já o *Renderer Process* é responsável por renderizar a interface gráfica, com cada janela ou aba da aplicação rodando um processo renderizador separado. Esses processos se comunicam entre si por meio de *Inter-Process Communication* (IPC), que permite enviar e receber mensagens entre o *Main Process* e o *Renderer Process* de maneira assíncrona, garantindo uma separação clara das responsabilidades e maior estabilidade. No *Renderer Process*, pode-se utilizar qualquer *framework* ou biblioteca *web* moderna, como React, Vue.js, Angular, entre outros, para construir a interface gráfica da aplicação, tornando o desenvolvimento ainda mais flexível e acessível para desenvolvedores familiarizados com essas tecnologias.

Integrado à aplicação em Electron, várias outras bibliotecas JavaScript foram utilizadas, sendo elas:

- **React:** Uma biblioteca *front-end* de código aberto usada para construir interfaces de usuário, especialmente em aplicações *web* de página única. Ela permite a criação de componentes reutilizáveis, facilitando a gestão de estados e a renderização eficiente da interface. Seu principal objetivo é melhorar a performance e a experiência do usuário, tornando o desenvolvimento mais ágil e modular;
- **Recharts:** Uma biblioteca baseada em React, utilizada para criar gráficos interativos e responsivos de forma simples e eficiente. Ela oferece uma ampla variedade de tipos de gráficos, como linhas, barras e setores, permitindo a visualização clara de dados. Seu principal objetivo é facilitar a integração de gráficos complexos em interfaces *web* com facilidade e personalização;
- **Node SerialPort:** Uma biblioteca usada para estabelecer comunicação com dispositivos através de portas seriais, como sensores e microcontroladores. Ela facilita a leitura e escrita de dados entre o computador e dispositivos externos, sendo essencial em aplicações que exigem interação com hardware. É utilizada neste projeto para comunicar com o microcontrolador do sistema via interface gráfica.

4 DESENVOLVIMENTO

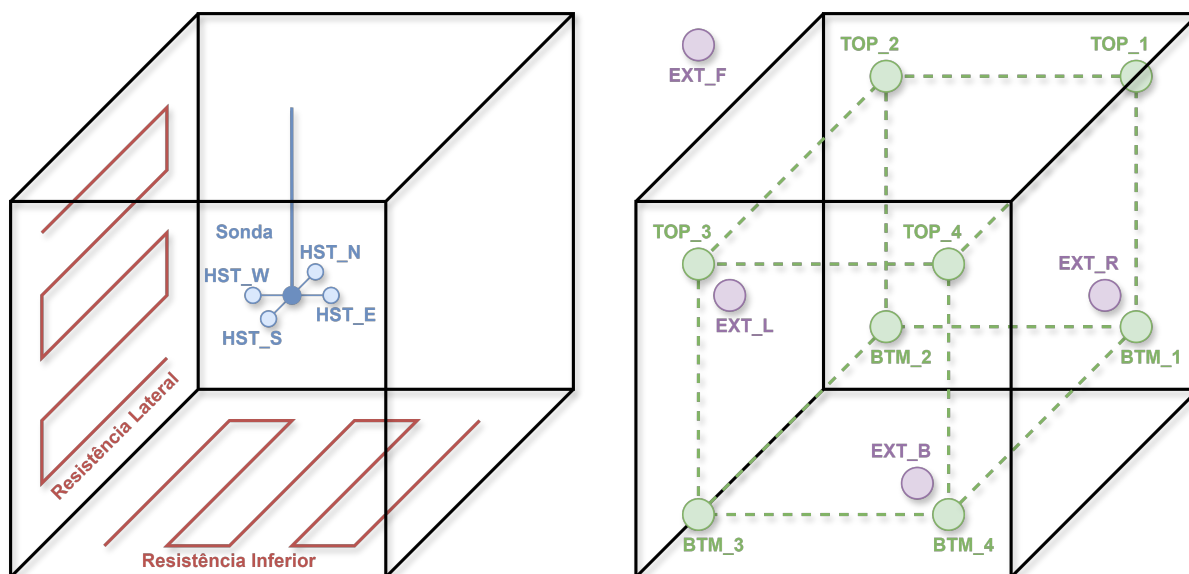
Este capítulo aborda o desenvolvimento e a construção dos principais componentes do sistema, divididos em três seções principais. A primeira explora o volume de controle, detalhando o isolamento térmico, o dimensionamento das resistências de aquecimento, o uso de modelagem *Computer-Aided Design* (CAD) e impressão 3D, além da etapa de construção. A segunda seção trata do gabinete de controle, englobando desde a alimentação do sistema até a elaboração das placas eletrônicas, a integração do microcontrolador, a montagem do painel elétrico e a calibração dos sensores, finalizando com a programação do sistema. Por fim, a terceira seção apresenta a interface gráfica de usuário, destacando sua importância na simplificação da interação com o sistema.

4.1 VOLUME DE CONTROLE

O volume de controle será estruturado como uma caixa cúbica. Resistências de aquecimento serão fixadas nas paredes internas da lateral esquerda e da base inferior, com acionamento individual e controlado. A dissipação de potência dessas resistências permitirá a propagação de calor nas direções horizontal, a partir da parede lateral esquerda, e vertical, a partir da base inferior, reproduzindo diferentes condições de contorno térmicas dentro do volume.

O volume de controle também será equipado com um suporte para suspensão da sonda de fluxo de calor exatamente em seu centro que, por sua vez, inclui termômetros orientados nas direções cardeais (HST_N, HST_S, HST_E e HST_W). Adicionalmente, oito termômetros internos estarão distribuídos nos vértices de um cubo imaginário localizado dentro do volume de controle (TOP_1, TOP_2, TOP_3, TOP_4, BTM_1, BTM_2, BTM_3 e BTM_4). Complementando o sistema, quatro termômetros externos serão instalados: três nas superfícies externas do volume de controle (EXT_L, EXT_B e EXT_R) e um afastado do volume (EXT_F) para registrar a temperatura ambiente. A comparação entre eles irá proporcionar dados adicionais, permitindo corrigir na simulação fluxos parasitas de calor entrando ou saindo do volume. Um esquemático do que foi proposto é apresentado na Figura 9.

Figura 9 – Esquemático do volume de controle.



Fonte: do autor.

4.1.1 Isolamento térmico

Para melhorar o desempenho do volume de controle, foi imprescindível que sua construção priorizasse o isolamento térmico em relação ao ambiente externo. Para isso, foi realizada uma etapa de estudos sobre materiais isolantes que combinassem baixa condutividade térmica, facilidade de manuseio e viabilidade econômica, Tabela 1. O material selecionado para as paredes do volume foi escolhido com o objetivo de minimizar perdas de calor para o exterior, assegurando que a energia gerada pelas resistências se propagasse predominantemente no interior do volume de controle, otimizando os experimentos.

Tabela 1 – Condutividade térmica de alguns materiais.

Material	Condutividade Térmica [W/(mK)]
Cobre	385 - 400
Alumínio	205 - 235
Aço Carbono	43 - 58
Vidro Comum	0,78 - 1,0
Concreto	0,8 - 1,4
Borracha	0,13 - 0,20
Madeira (Pinho)	0,12 - 0,14
Lã de Vidro	0,035 - 0,040

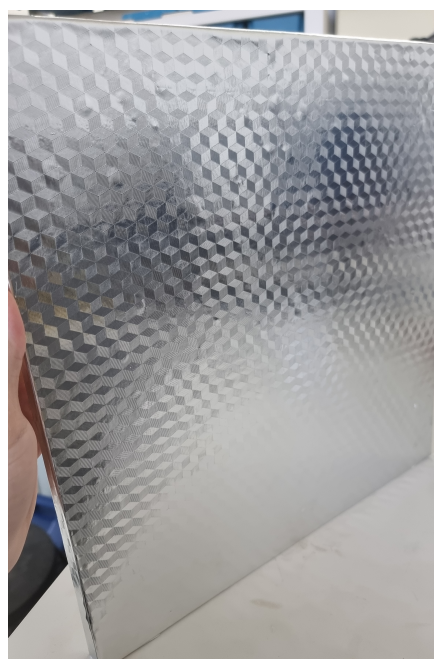
Fonte: Dewitt *et al.* (2006).

A madeira maciça foi escolhida como material principal para a construção do volume de controle, pois além de atender ao requisito de baixa condutividade térmica, apresenta facilidade de manuseio e foi economicamente viável para este projeto. Para melhorar ainda mais o desempenho térmico, a madeira foi revestida com uma película térmica impermeável e refletiva, a fim de minimizar as perdas radiativas do meio, Figura 10. Por fim, as paredes externas foram recobertas com *Expanded Polystyrene* (EPS), conhecido por sua excelente capacidade de isolamento térmico, proporcionando um controle ainda melhor das condições térmicas internas do volume de controle.

Figura 10 – Revestimento da madeira com película térmica.



(a) Madeira bruta



(b) Madeira revestida

Fonte: do autor.

4.1.2 Dimensionamento das resistências de aquecimento

As resistências de aquecimento precisam ser determinadas com antecedência para um correto dimensionamento do volume de controle e seleção adequada dos componentes eletrônicos.

Para isso, considera-se uma região de influência ao redor do elemento de aquecimento, com área A_c e comprimento L_h , preenchida por uma amostra de solo com capacidade térmica volumétrica C_V . A temperatura nessa região deve ser elevada em um valor ΔT dentro de um intervalo de tempo Δt . Dado que C_V é a quantidade de energia Q necessária para causar uma variação de temperatura ΔT a uma unidade de volume de solo, tem-se que

$$C_V = \frac{Q}{A_c L_h \Delta T}, \quad (12)$$

de forma que a potência da resistência de aquecimento pode ser dada por

$$P = \frac{Q}{\Delta t}. \quad (13)$$

Desse modo, substituindo a Equação (13) na Equação (12) e que $2 \text{ MJ}/(\text{m}^3\text{°C})$ é um valor típico para C_V , $L_h = 0,03 \text{ m}$, o volume de controle tem lado de $0,3 \text{ m}$ e portanto $A_c = 0,09 \text{ m}^2$, um tempo de aquecimento $\Delta t = 10 \text{ min}$ e uma variação de temperatura média $\Delta T = 10 \text{ °C}$, a potência necessária que deve ser dissipada pelas resistências é

$$P = \frac{C_V A_c L_h \Delta T}{\Delta t} = 90 \text{ W}. \quad (14)$$

Portanto, considerando que as resistências estão sob uma tensão de $V = 12\text{V}$, o valor de cada resistência de aquecimento R é determinado como

$$R = \frac{V^2}{P} = 1,6 \text{ } \Omega. \quad (15)$$

Utilizando um filamento de nicromo, uma liga metálica composta de níquel e cromo com resistividade de $1,3 \text{ } \Omega/\text{m}$, o comprimento necessário para atingir a resistência desejada é de aproximadamente $1,23 \text{ m}$.

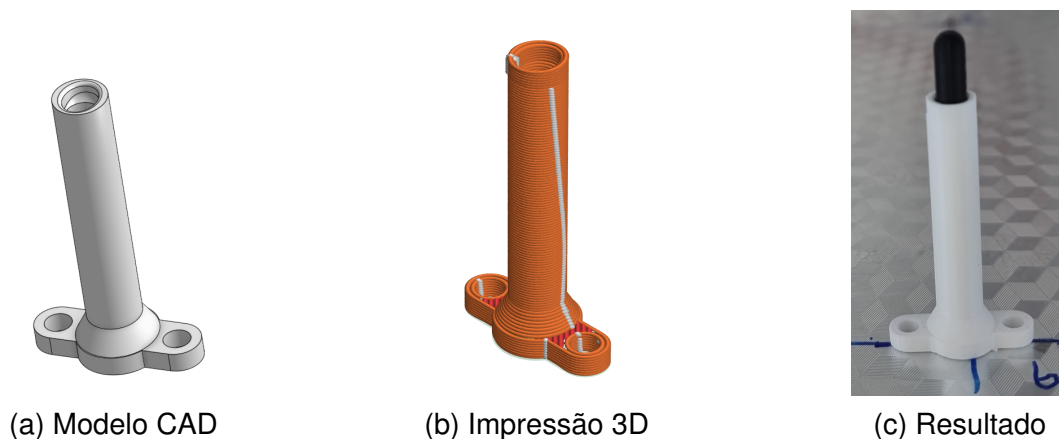
Perceba que mensurar o valor de resistência dessa ordem de grandeza apresenta desafios significativos, especialmente quando realizada a dois fios, devido às resistências adicionais introduzidas pelas conexões e cabos, que podem distorcer a leitura real. Para melhorar a qualidade da medição, foi aplicada a técnica de Kelvin de 4 fios, utilizando uma fonte de corrente de bancada e um voltímetro. Além disso, como a resistividade dos materiais condutores tende a aumentar com a elevação da temperatura, a resistência utilizada no experimento pode sofrer variações significativas em função das altas temperaturas alcançadas, impactando a potência dissipada. Dado que essa potência pode não ser fixa ao longo do tempo, sensores de corrente serão empregados para monitoramento contínuo, permitindo que a energia entregue ao sistema seja estimada pela integração dos valores de corrente no tempo, considerando uma tensão de alimentação fixa.

4.1.3 Modelagem CAD e impressão 3D

Após a concepção do esquemático do volume de controle e dimensionamento de componentes, avançou-se para a criação do modelo CAD, permitindo uma visualização mais detalhada do projeto. Utilizando o *software* SolidWorks, cada componente do sistema foi desenhado individualmente e posteriormente integrado para formar a montagem completa. Por questões de sigilo, a modelagem CAD realizada, assim como outras partes do projeto, encontram-se no documento de apêndice sigiloso, que não é de acesso público. Portanto, a vista explodida do desenho juntamente com uma lista dos principais elementos do volume de controle é apresentado no apêndice sigilo A, enquanto imagens renderizadas de alta qualidade, geradas pelo SolidWorks Visualize podem ser visualizadas no apêndice sigilo B.

Alguns componentes do volume de controle, como o suporte interno do termistor, Figura 11, foram fabricados por impressão 3D, utilizando o material Acrilonitrila Butadieno Estireno (ABS), que fornece uma boa resistência mecânica e baixa condutividade térmica. O modelo CAD desses elementos foi exportado no formato *Standard Tessellation Language* (STL), compatível com o *software* de impressão 3D Creality Print, responsável por interpretar o arquivo e gerar o G-code para a impressora 3D.

Figura 11 – Suporte interno do termistor.

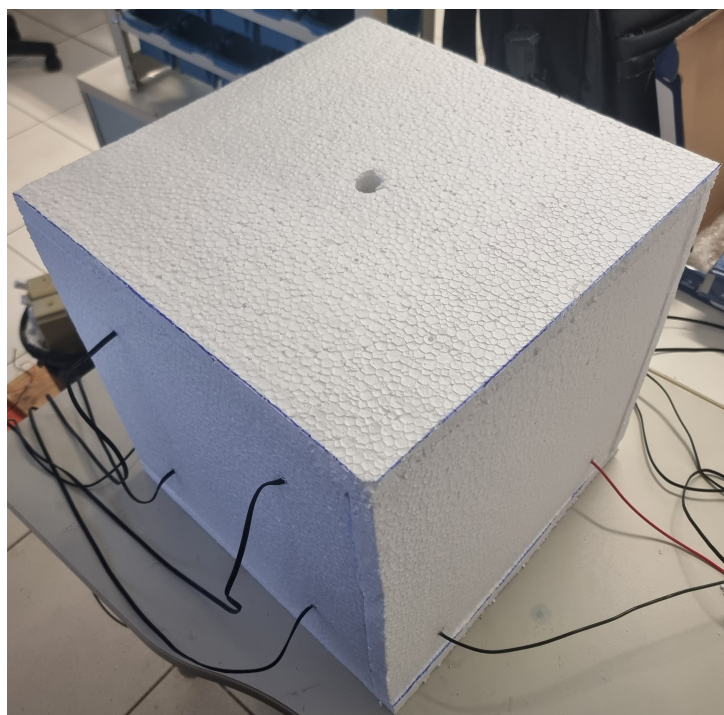


Fonte: do autor.

4.1.4 Construção

Com a modelagem 3D do projeto finalizada e os materiais previamente definidos, iniciou-se a construção do volume de controle. O volume foi projetado na forma de um cubo, com paredes de 1,5 cm de espessura e 30 cm de comprimento internamente, resultando em uma capacidade aproximada de 27.000 cm^3 , desconsiderando o espaço ocupado pelos elementos internos. Para assegurar a impermeabilidade do sistema, as arestas do cubo foram seladas com silicone, minimizando as perdas de qualquer umidade presente na amostra de solo. A construção finalizada é apresentada na Figura 12 e descrita em maior detalhe no apêndice sigiloso C.

Figura 12 – Construção do volume de controle.



Fonte: do autor.

4.2 GABINETE DE CONTROLE

Esta seção descreve o desenvolvimento do gabinete de controle, projetado para integrar de maneira compacta todos os componentes eletrônicos do sistema. Este gabinete é responsável pela gestão de energia, tratamento de sinais, armazenamento de dados e comunicação com a interface gráfica do usuário. Construído em uma caixa de montagem feita de plástico resistente e impermeável, o gabinete protege os componentes contra poeira e os organiza de forma eficiente em um painel elétrico,

garantindo segurança e funcionalidade no sistema. As subseções seguintes tratarão em detalhes de todos os elementos que compõem o gabinete de controle.

4.2.1 Alimentação do sistema

Todo o sistema será alimentado diretamente pela rede elétrica, garantindo uma fonte de energia confiável e contínua. O gabinete de controle conta com um cabo de alimentação AC, composto por fase, neutro e terra, equipado com um conector padrão para tomadas. Este cabo está ligado a uma fonte chaveada, Figura 13, responsável por converter a tensão alternada de 220V da rede elétrica em uma tensão contínua de 12V. A fonte utilizada é capaz de fornecer até 30A, garantindo energia suficiente para alimentar as resistências de aquecimento e os demais componentes eletrônicos integrados ao gabinete de controle.

Figura 13 – Fonte chaveada AC/DC 220V/12V 30A.



Fonte: do autor.

Alguns componentes eletrônicos do sistema demandam tensões menores, como 5V e 3,3V, sendo necessário a utilização de um conversor de tensão *step-down* para adaptar a tensão de 12V fornecida pela fonte chaveada. Por conta disso, foi empregado o conversor *Buck* LM2596, Figura 14, que possui capacidade de aceitar tensões de entrada entre 4,5V a 40V, reduzindo-as para valores ajustáveis entre 1,2V a 37V, com uma corrente de saída máxima de 3A (TEXAS INSTRUMENTS, 2023). No projeto, foram utilizados dois desses conversores, disponibilizando tensões reguladas de 5V e 3,3V no gabinete de controle.

Figura 14 – Módulo regulador de tensão DC/DC LM2596.



Fonte: ROBOCORE (2024).

4.2.2 Placa de multiplexação

A placa de multiplexação é responsável pelo primeiro estágio de condicionamento dos sinais de temperatura, permitindo que o sistema contorne a limitação de entradas ADC do microcontrolador. Como este não possui canais suficientes para realizar a leitura simultânea de todos os termistores, a utilização de multiplexadores torna-se indispensável, permitindo a expansão virtual dos canais disponíveis e possibilitando a seleção e leitura sequencial de cada sensor de forma otimizada.

Ao todo o sistema possui 16 termistores, que, para facilitar o entendimento, foram divididos em dois grupos: o primeiro grupo inclui os 4 termistores localizados na sonda de fluxo de calor, enquanto o segundo grupo compreende os 12 termistores dispostos dentro e fora do volume de controle, como já descrito anteriormente. A placa é equipada com dois multiplexadores de 16 canais, Figura 15, cada um designado para lidar com um grupo específico de termistores. Cada multiplexador é responsável por selecionar apenas um termistor de seu respectivo grupo por vez, conectando-o a uma configuração de divisor de tensão que permite medir o sinal gerado pelo termistor selecionado. O sinal de tensão resultante do divisor de tensão é então direcionado à saída da placa, de onde segue para o estágio de amplificação.

Figura 15 – Módulo multiplexador CD74HC4067.



Fonte: saravati (2024).

Em suma, a placa de multiplexação desempenha a função de reduzir os 16 sinais provenientes dos termistores para apenas 2. A seleção dos termistores nos multiplexadores é realizada por meio de quatro portas digitais, S0, S1, S2 e S3, controladas pelo microcontrolador, permitindo a escolha sequencial de cada sensor. Vale destacar que, embora um único multiplexador de 16 canais fosse suficiente para acomodar todos os termistores, a separação dos sinais em dois grupos se mostrou vantajosa, dado que os sinais provenientes da sonda de fluxo de calor necessitam de um estágio de amplificação diferente daqueles originados no volume de controle. Essa abordagem permite um tratamento mais adequado e eficiente dos sinais desde a etapa inicial.

Todo o circuito da placa de multiplexação foi realizado no *software* Proteus, apresentado no apêndice A, assim como seu *layout* para *Printed Circuit Board* (PCB).

4.2.3 Placa de amplificação

A placa de amplificação é responsável pelo segundo e último estágio de condicionamento dos sinais de temperatura no sistema. Os dois sinais provenientes da placa de multiplexação possuem amplitudes relativamente pequenas em comparação à faixa de entrada do ADC do microcontrolador, o que limita o uso eficaz dos níveis de quantização disponíveis. Ao amplificar esses sinais, é possível aproximá-los do intervalo máximo de medição do ADC, maximizando a utilização dos níveis de quantização disponíveis. Embora isso não aumente a resolução física do ADC, que permanece fixa, melhora significativamente a representação digital dos sinais amplificados, tornando pequenas variações nos valores de entrada mais perceptíveis e proporcionando maior sensibilidade na análise dos dados.

Para amplificar os dois sinais de temperatura provenientes da placa de multiplexação, foram utilizados dois amplificadores de instrumentação INA333, Figura 16, alimentados de forma assimétrica com 3,3 V. Cada amplificador foi configurado para aplicar um ganho específico, ajustado de acordo com a faixa de temperatura desejada para monitoramento.

Figura 16 – Módulo amplificador de instrumentação INA333.



Fonte: easytronics (2024b).

Para a sonda de fluxo de calor, o intervalo de monitoramento de temperatura adotado é de $[15, 30]^\circ\text{C}$, um intervalo padrão comumente utilizado nas sondas do STMF. Conforme a tabela apresentada no anexo A, o termistor 103AT-11 apresenta uma variação em sua resistência dentro desse intervalo de temperatura de $[14,69, 8,313] \text{ k}\Omega$. Assim, levando em consideração esse intervalo de resistência do termistor, a tensão de saída do divisor de tensão formado pelos termistores da sonda assume o seguinte intervalo

$$V_{OUT,1} = \left[3,3 \frac{16\text{k}}{16\text{k} + 14,69\text{k}}, 3,3 \frac{16\text{k}}{16\text{k} + 8,313\text{k}} \right] \text{ V} \approx [1,70, 2,20] \text{ V}. \quad (16)$$

Perceba que o intervalo de tensão assumido por $V_{OUT,1}$ pode ser amplificado de maneira a se aproximar do intervalo de leitura mais linear do ADC do microcontrolador,

que está compreendido entre $[0,3, 3]$ V. Para alcançar a adaptação necessária, recorre-se à equação Equação (11) de amplificação. Nessa equação, o resistor R_G pode ser ajustado por meio de um potenciômetro integrado ao próprio módulo INA333, permitindo a definição de ganhos arbitrários. Além disso, a tensão V_{IN-} pode ser configurada utilizando *trimpots* instalados na placa, projetados especificamente para cada amplificador. Por fim, o valor de REF é fixado em 1,65 V, obtido a partir de um divisor de tensão alimentado com 3,3 V e resistências idênticas. Isso permite alcançar a seguinte equação de amplificação para o sinal dos sensores de temperatura da sonda:

$$V_{OUT,1,amp} = 5,4(V_{OUT,1} - 1,95) + 1,65, \quad (17)$$

o que resulta em um intervalo de tensão de aproximadamente $[0,3, 3]$ V, aproveitando melhor a resolução do ADC.

O mesmo pode ser feito para os sinais de temperatura do volume de controle. Contudo, neste caso, o intervalo de monitoramento de temperatura é maior, variando de $[10, 60]$ °C. Essa amplitude é justificada pelo fato de alguns termistores estarem expostos ao ambiente externo, enquanto outros estão próximos às resistências de aquecimento, podendo atingir temperaturas mais elevadas. Nesse intervalo, o termistor apresenta uma variação de resistência mais ampla, de $[17,96, 3,02]$ k Ω , resultando em um sinal de saída do divisor de tensão de

$$V_{OUT,2} = \left[3,3 \frac{16k}{16k + 17,96k}, \quad 3,3 \frac{16k}{16k + 3,02k} \right] V \approx [1,60, 2,80] V. \quad (18)$$

Amplificando o intervalo assumido por $V_{OUT,2}$, chega-se na seguinte amplificação de sinal

$$V_{OUT,2,amp} = 2,25(V_{OUT,2} - 2,2) + 1,65, \quad (19)$$

resultando novamente em um intervalo de tensão mais adequado para leitura pelo ADC, de $[0,3, 3]$ V.

O circuito da placa de amplificação é apresentada em maior detalhes no apêndice B, assim como seu *layout* para PCB.

4.2.4 Placa de acionamento

Por fim, a placa de acionamento foi projetada para centralizar e simplificar o controle das resistências de aquecimento, permitindo tanto o acionamento eficiente quanto o monitoramento da corrente elétrica dessas resistências, organizando os sinais e tensões comuns em um único local.

O acionamento controlado e individual das resistências de aquecimento, tanto lateral quanto inferior, do volume de controle é realizado por meio de um módulo relé de dois canais, Figura 17. Esses relés são ativados por sinais de 3,3 V provenientes do microcontrolador, que, ao energizar a chave, conectam as resistências a uma tensão de 12 V fornecida pela fonte chaveada. Por questões de segurança, os relés são configurados no modo normalmente aberto, garantindo que permaneçam desligados em caso de falha no microcontrolador, prevenindo acionamentos indesejados.

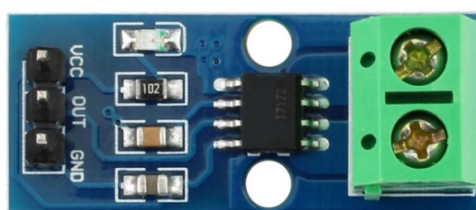
Figura 17 – Módulo relé 2 canais.



Fonte: eletrogate (2024).

O monitoramento da corrente das resistências de aquecimento é realizado por dois sensores de corrente baseados no efeito Hall, Figura 18, capazes de suportar até 20 amperes.

Figura 18 – Sensor de corrente por efeito Hall ACS712-20A.



Fonte: easytronics (2024a).

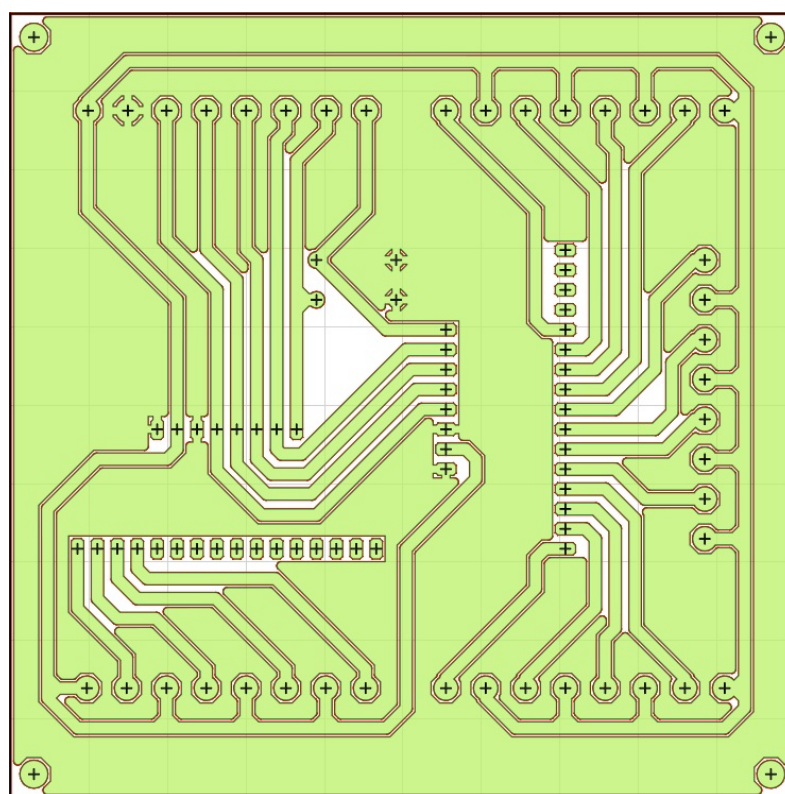
Esses sensores requerem alimentação de 5 V para funcionamento correto e fornecem uma saída linear que varia de 0 V, quando a corrente é 0 A, até 5 V, correspondente a 20 A. No entanto, como o ADC do microcontrolador utilizado aceita uma tensão máxima de 3,3 V, o sinal desses sensores precisa ser condicionado. Para isso, é empregado dois divisores de tensão, um para cada sensor, utilizando resistências de 3,3 k Ω e 2,2 k Ω , ajustando a faixa do sinal para [0, 3] V, compatível com o ADC.

No apêndice C, encontra-se uma descrição detalhada do circuito da placa de acionamento, incluindo o *layout* completo para a fabricação da PCB.

4.2.5 Elaboração das PCBs

Com o projeto do circuito e o *layout* das placas de multiplexação, amplificação e acionamento finalizados, iniciou-se a etapa de confecção das PCBs. O método escolhido foi a fresagem por Controle Numérico Computadorizado (CNC), uma técnica precisa e amplamente utilizada para prototipagem de placas eletrônicas. Os *layouts*, desenvolvidos no *software* Proteus, foram exportados no formato Gerber, padrão da indústria para transferência de designs de PCB. Esses arquivos foram posteriormente lidos pelo *software* Universal Gcode Sender, que interpreta o g-code e os envia à fresadora CNC. Na Figura 19 a seguir, é apresentada uma visualização do caminho da fresa no cobre inferior e dos furos.

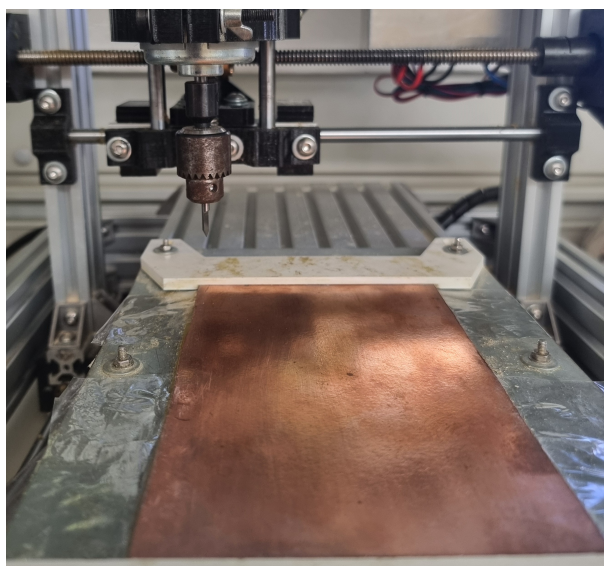
Figura 19 – Visualização dos caminhos da fresa na placa de multiplexação.



Fonte: do autor.

As PCBs foram produzidas utilizando placas com uma camada de cobre em uma das faces. Utilizado uma fresadora CNC disponível no laboratório, Figura 20, a face cobreada foi fresada com uma broca para gravura de 0,1mm para confecção das trilhas e uma broca de 1mm para os furos de componentes.

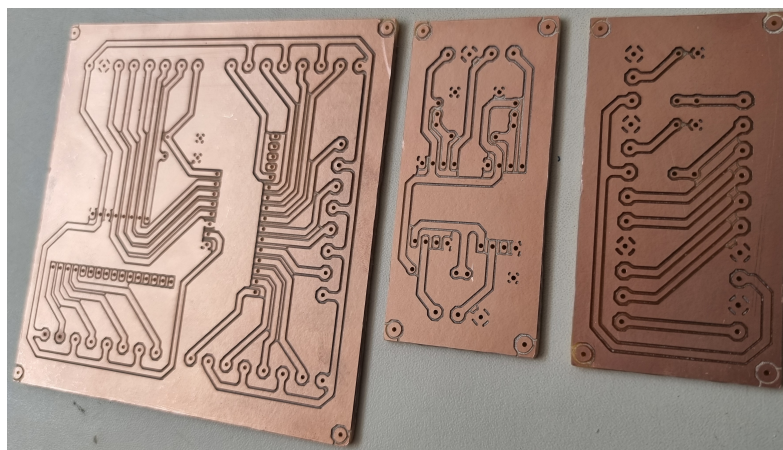
Figura 20 – Fresadora CNC utilizada para fabricar as PCBs.



Fonte: do autor.

O resultado do fresamento das três placas projetadas (multiplexação, amplificação e acionamento) pode ser observado na Figura 21. Em seguida, todos os componentes foram devidamente soldados em suas respectivas placas, concluindo a etapa de elaboração das PCBs.

Figura 21 – Placas cobreadas com as trilhas e furos fresados.

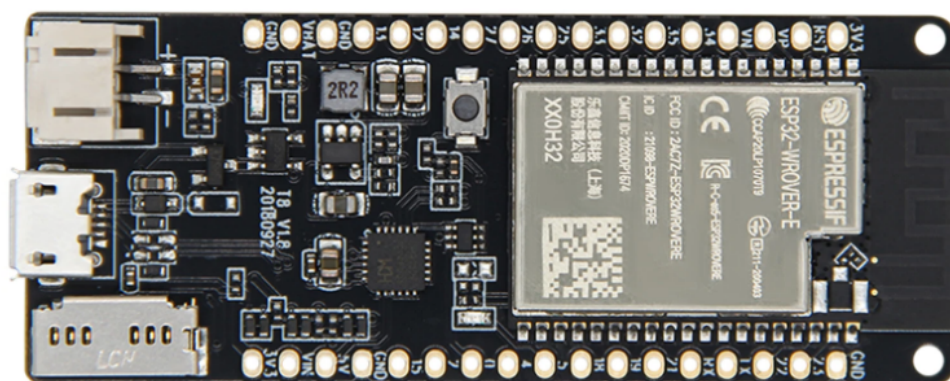


Fonte: do autor.

4.2.6 Microcontrolador

O controle e monitoramento dos sinais do gabinete de controle são realizados pelo microcontrolador ESP32, modelo LILYGO T8 V1.8, Figura 22. Esse módulo é baseado no chip ESP32-WROVER-E, desenvolvido pela Espressif Systems.

Figura 22 – Microcontrolador ESP32 - LILYGO T8 V1.8.



Fonte: lilygo (2024).

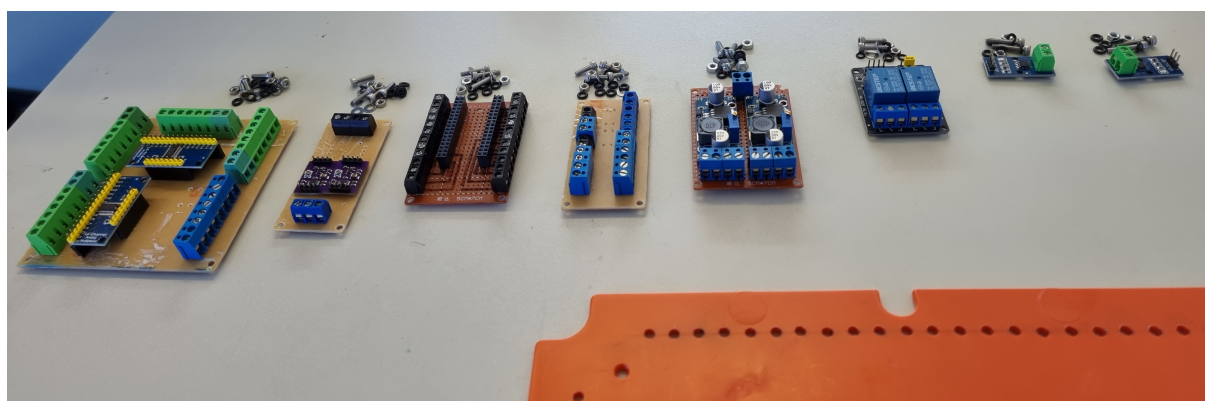
O ESP32 é um microcontrolador de alto desempenho que se destaca por seus dois núcleos de processamento e suporte ao kernel FreeRTOS, sendo essencial para a implementação baseada em programação concorrente utilizada neste projeto. Esse componente opera com uma tensão de 3,3V e oferece uma ampla gama de funcionalidades, incluindo portas GPIO para sinais lógicos de acionamento, conversores ADC para leituras analógicas e diversas interfaces de comunicação, como a comunicação USB serial utilizada para se comunicar com computadores externos. Esse modelo de microcontrolador utilizado foi particularmente vantajoso neste projeto devido à integração de um módulo para leitura de cartões de memória SD, o que permitiu uma expansão significativa da capacidade de armazenamento e possibilitando o registro de grandes volumes de dados experimentais. O anexo B mapeia e apresenta detalhadamente as portas disponíveis, assim como a integração com o módulo de cartão SD.

Esse microcontrolador utilizado no gabinete de controle exerce funções essenciais ao executar toda a rotina de controle e monitoramento do sistema. Durante os experimentos, ele é responsável pelo acionamento individual das resistências de aquecimento e pela leitura dos sinais de corrente associados, além de realizar a seleção dos canais nos multiplexadores e a leitura e armazenamento dos sinais de temperatura. Adicionalmente, interage com computadores externos via interface USB, enviando dados experimentais e recebendo comandos.

4.2.7 Elaboração do painel elétrico

Com todos os componentes definidos e as PCBs devidamente produzidas, iniciou-se a montagem do painel elétrico do gabinete de controle, Figura 23. Os componentes e as placas foram estrategicamente organizados e fixados no painel, visando otimizar a disposição interna, garantir acessibilidade para eventuais manutenções e promover a organização geral do sistema.

Figura 23 – Componentes e PCBs a serem fixados no painel elétrico



Fonte: do autor.

As fiações que conectam os componentes entre si foram cuidadosamente passadas por trás do painel elétrico, utilizando furos específicos para garantir a organização e a estética do gabinete. Além disso, para facilitar a identificação e garantir a clareza nas conexões, os fios seguiram um padrão de cores bem definido:

- **Vermelho:** Alimentação de 5V;
- **Alaranjado:** Alimentação de 3,3V;
- **Preto:** GND;
- **Branco:** Sinais lógicos (0V ou 3,3V);
- **Amarelo:** Sinais analógicos, que variam de 0V a 3,3V.

Para garantir a segurança do sistema em caso de falhas elétricas, foram adicionadas três chaves de acionamento manual ao painel elétrico, Figura 24, permitindo controlar a energia de certos componentes de forma independente. A primeira e a segunda chave estão em série com a alimentação das resistências de aquecimento, uma para a resistência inferior e outra para a resistência lateral, possibilitando o corte de energia de cada uma delas individualmente. A terceira chave, por sua vez, controla

Figura 24 – Chave Gangorra liga-desliga KCD1.



Fonte: HR2 Eletronica (2024).

a alimentação do microcontrolador, oferecendo um meio adicional de desligamento do sistema em situações de emergência ou manutenção.

O apêndice sigiloso D deste trabalho apresenta figuras detalhadas do painel elétrico elaborado.

4.2.8 Calibração dos sensores

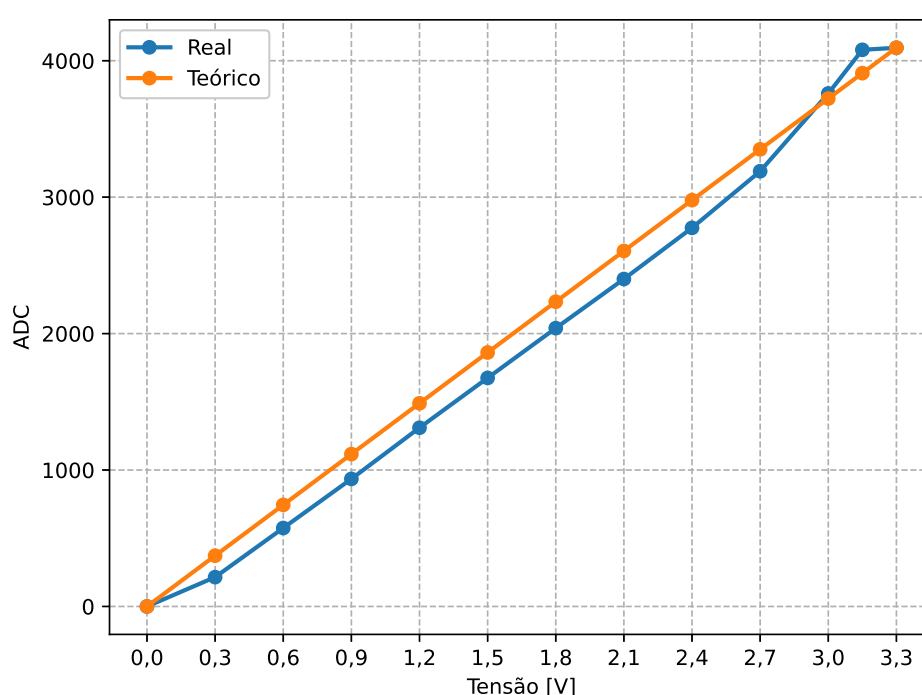
No capítulo 3 deste trabalho foi abordado uma forma de medir a temperatura utilizando termistores, através da equação do parâmetro beta, que relaciona temperatura com a resistência do termistor, calculada a partir da tensão de saída do divisor de tensão formado com o termistor.

Contudo, mensurar com exatidão o valor do sinal de tensão do divisor para determinar a temperatura é uma tarefa complexa, devido às diversas distorções que o sinal sofre até chegar à entrada analógica (ADC) do microcontrolador. Elementos intermediários como fios, bornes, multiplexadores, amplificadores e trilhas de PCBs introduzem alterações no sinal que reduzem sua qualidade, como quedas de tensão adicionais decorrentes da resistência dos condutores, ruído por contatos imperfeitos ou resistências parasitas. Além disso, as próprias entradas analógicas do microcontrolador apresentam erros intrínsecos, como *offset* e não linearidade, que também impactam a exatidão da medida.

Para mitigar os problemas relacionados à distorção do sinal, a metodologia de calibração adotada considerou o caminho completo do sinal no sistema, estabelecendo uma curva que relaciona diretamente a temperatura com a tensão medida pelo microcontrolador. Além disso, foi realizada a calibração do ADC do microcontrolador, definindo uma escala ajustada que associa cada valor de ADC à sua respectiva tensão.

Utilizando uma fonte de bancada e um voltímetro para validação, foram aplicadas tensões conhecidas à entrada analógica do microcontrolador, e os respectivos valores de ADC foram registrados. A Figura 25 apresenta a comparação entre a curva teórica, que assume um comportamento perfeitamente linear, e a curva real, obtida pelos valores medidos. Observa-se que, na prática, a curva real exibe um claro desvio de *offset* e uma leve não linearidade. Para corrigir essas discrepâncias, as medições calibradas são obtidas por meio da interpolação entre os pontos da curva real e não da teórica.

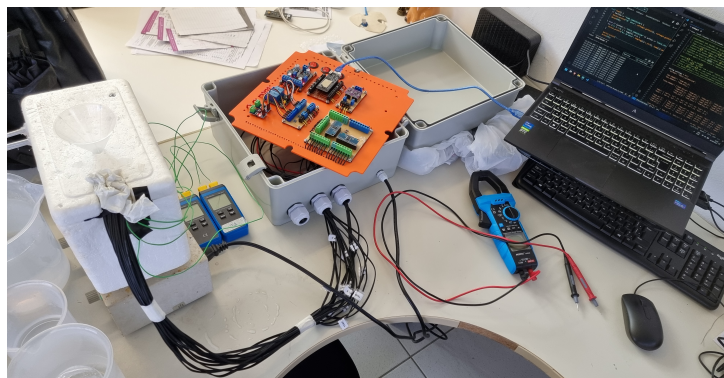
Figura 25 – Curva ADC x tensão do microcontrolador.



Fonte: do autor.

Para a calibração dos sinais de temperatura, todos os termistores foram colocados em um banho térmico, juntamente com termômetros de referência baseados em termopares, com resolução de 0,01 °C, Figura 26. Em seguida, foi realizado um varrido de temperatura no intervalo de 10 °C a 60 °C, com incrementos de aproximadamente 2 °C. Para cada temperatura registrada no banho térmico, os valores de tensão medidos pelo microcontrolador, já ajustados com a escala ADC calibrada, foram anotados, estabelecendo uma relação direta entre temperatura e tensão.

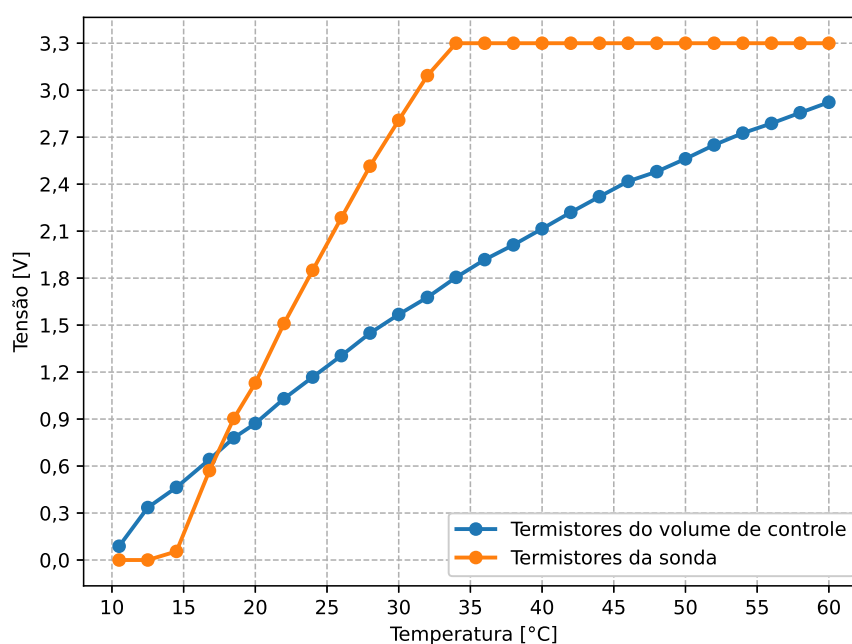
Figura 26 – Procedimento de calibração dos termistores.



Fonte: do autor.

A Figura 27 apresenta as curvas obtidas para os sinais dos termistores da sonda e do volume de controle. É importante lembrar que o intervalo de medição dos termistores da sonda é limitado a $[15, 30]$ °C, sendo saturado fora desse intervalo. É interessante observar nas curvas que, apesar das tentativas de condicionar os sinais a uma faixa ideal de $[0,3, 3,0]$ V, pequenas distorções nos sinais pelos problemas já comentados são inevitáveis. Por essa razão, a calibração é necessária, onde novamente foi realizada a interpolação entre os pontos das curvas dos respectivos sinais, relacionando diretamente a temperatura e a tensão medida no microcontrolador.

Figura 27 – Curva tensão x temperatura dos termistores.



Fonte: do autor.

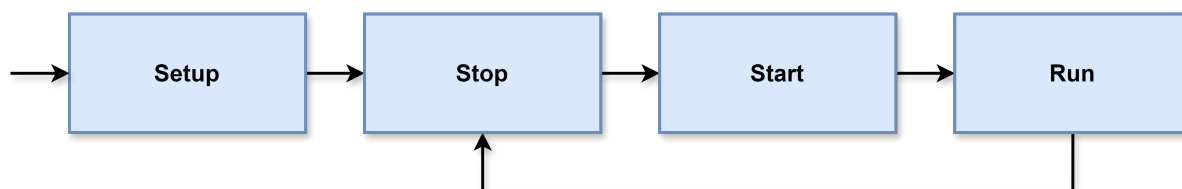
A mesma técnica de calibração foi aplicada aos sensores de corrente das resistências. Para cada sensor, uma corrente conhecida foi aplicada utilizando uma fonte de corrente de bancada, sendo validada por meio de um amperímetro. A partir desses testes, foi possível levantar curvas que estabelecem uma relação direta entre a corrente aplicada e a tensão medida pelo microcontrolador, permitindo a calibração adequada dos sinais de corrente no sistema.

4.2.9 Programação do sistema

O funcionamento do sistema foi projetado com base em programação concorrente, utilizando o *kernel* FreeRTOS para gerenciar as tarefas. Nesse modelo, o trabalho é dividido em unidades chamadas tarefas, que são executadas de forma independente e simultânea. No total, foram criadas duas tarefas principais para gerenciar todo o sistema, as quais são executadas paralelamente pelo microcontrolador.

Antes de abordar as tarefas, é fundamental compreender os estados operacionais que o sistema pode assumir. De maneira simplificada, o sistema opera em quatro estados distintos, conforme ilustrado no fluxograma da Figura 28.

Figura 28 – Estados de funcionamento do sistema.



Fonte: do autor.

O primeiro estado é o **Setup**, dedicado à inicialização do sistema, no qual são configuradas a interface de comunicação, o módulo de armazenamento, o vínculo das portas do microcontrolador com seus respectivos sinais, as escalas calibradas do ADC e dos sensores e também as duas tarefas a serem executadas, deixando o sistema preparado para funcionamento. Após essa etapa, o sistema transita para o estado **Stop**, indicando que nenhum experimento está sendo executado no momento. Por meio de comandos, o sistema pode avançar para o estado **Start**, um estágio intermediário que sinaliza que um experimento está prestes a começar. Finalmente, o sistema entra no estado **Run**, no qual o experimento é efetivamente executado. Durante o estado **Run**, também é possível, por meio de comandos, retornar ao estado **Stop**, encerrando a execução do experimento.

A primeira tarefa independe do estado do sistema, sendo executada da mesma forma nos estados de **Stop**, **Start** e **Run**. Ela é responsável por monitorar continua-

mente a interface serial, recebendo e executando comandos de um computador externo. Esses comandos incluem iniciar ou parar o experimento, momento em que a tarefa altera o estado do sistema conforme necessário ou atualiza parâmetros específicos do experimento. Os parâmetros que podem ser modificados incluem a escolha da resistência a ser acionada (lateral ou inferior), o tempo de acionamento das resistências, o período de amostragem dos sensores de temperatura e até mesmo o nome do arquivo onde os dados experimentais serão armazenados. Além disso, essa tarefa também é encarregada de enviar, caso solicitado, os status atuais do sistema para o computador externo, proporcionando a comunicação em tempo real durante a execução do experimento.

A segunda tarefa do sistema desempenha funções específicas, conforme o estado atual do sistema. No estado **Stop**, a tarefa realiza o monitoramento periódico de todos os sensores, de acordo com o tempo de amostragem configurado. Embora os dados sejam enviados ao computador externo via interface serial, eles não são armazenados. Quando o sistema entra no estado **Start**, a tarefa é responsável por acionar a resistência selecionada e configurar um *software timer*, gerenciado pelo *kernel* FreeRTOS, para desligar as resistências assim que o tempo de acionamento configurado for precisamente alcançado. Após concluir essas configurações, a tarefa altera o estado do sistema para **Run**. Nesse último estado, os dados monitorados pela tarefa são enviados ao computador externo e também armazenados em um arquivo na memória do cartão SD, viabilizando uma forma prática de recuperar as informações geradas durante o experimento.

Para simplificar a programação do sistema, foram utilizadas duas bibliotecas específicas. A primeira, denominada "RincProbe.h", oferece classes para o gerenciamento genérico de sensores. Essa biblioteca permite a inserção de escalas calibradas personalizadas, realiza interpolação de valores com base na escala e fornece métodos para calcular valores médios a partir de múltiplas medições. A segunda biblioteca, "StorageCard.h", é responsável pelo gerenciamento do módulo de armazenamento. Ela disponibiliza métodos práticos para criar, excluir e adicionar valores em arquivos armazenados no cartão SD, facilitando o registro e a recuperação dos dados do experimento. O código elaborado para o sistema é apresentado no apêndice D.

4.3 INTERFACE GRÁFICA DE USUÁRIO

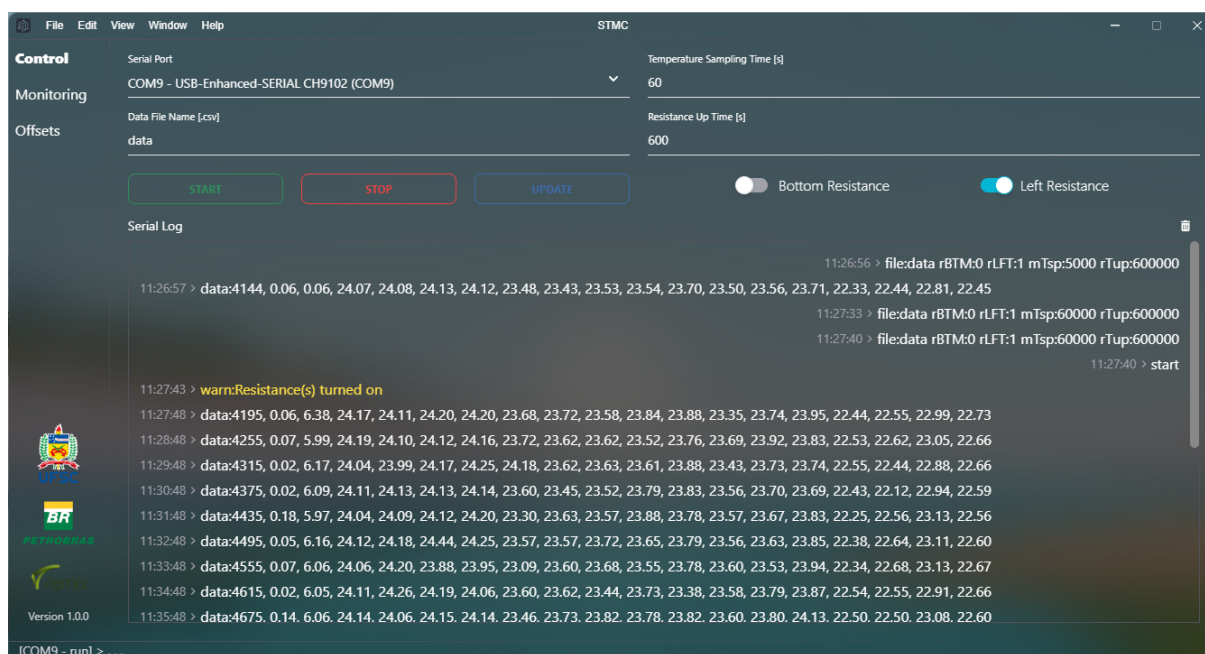
Para facilitar a interação com o sistema, foi desenvolvido um *software* com interface gráfica que permite tanto o monitoramento quanto o controle do sistema. Esse *software* foi criado utilizando o *framework* Electron, possibilitando sua execução em computadores com sistemas operacionais diversos, como Windows, macOS e Linux. A comunicação entre o computador e o sistema ocorre exclusivamente por meio de mensagens enviadas pela interface serial USB. A interface gráfica abstrai esses

comandos em elementos visuais intuitivos, como botões, campos de texto, caixas de seleção e gráficos. Ela é organizada em duas páginas principais, voltadas para as funções de controle e monitoramento, que serão detalhadas nas subseções seguintes.

4.3.1 Página de controle

A página de controle da interface gráfica, ilustrada na Figura 29, é dedicada aos comandos necessários para operar o sistema. Inicialmente, é preciso selecionar a porta USB conectada ao sistema por meio de uma caixa de seleção, estabelecendo a comunicação entre o computador e o microcontrolador. Os botões *START* e *STOP* permitem enviar comandos para alterar o estado do sistema, iniciando ou encerrando o experimento. Além disso, parâmetros do experimento, como o nome do arquivo de armazenamento, o período de amostragem dos sensores, o tempo de acionamento das resistências e a escolha da resistência a ser ativada, podem ser configurados nesta página. Essas configurações são enviadas ao sistema através do botão *UPDATE*, que transmite os parâmetros atualizados. Complementando a funcionalidade, um *log* exibe o registro completo da comunicação entre o sistema e o computador, incluindo mensagens de aviso, como o acionamento das resistências, e erros que possam surgir durante o experimento, como falhas no armazenamento dos dados.

Figura 29 – Interface gráfica: Página de controle.

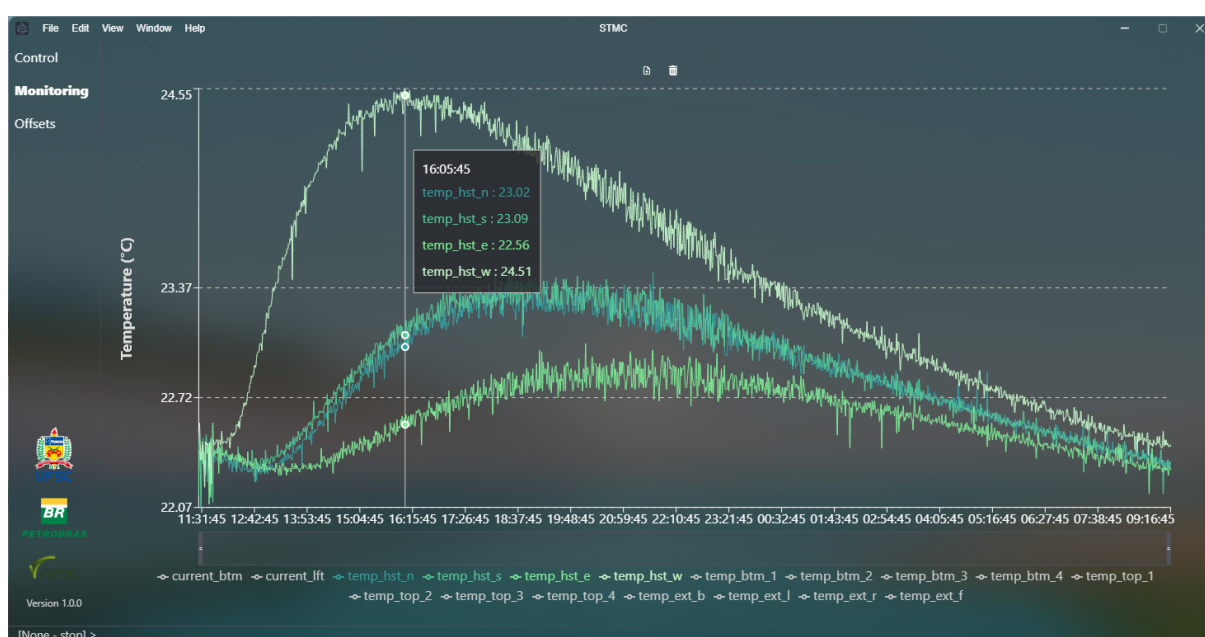


Fonte: do autor.

4.3.2 Página de monitoramento

A página de monitoramento da interface gráfica, ilustrada na Figura 30, permite acompanhar em tempo real os dados gerados por cada sensor do sistema. É possível selecionar múltiplos sensores simultaneamente, cujos valores são exibidos em um gráfico contínuo, cobrindo o período desde o início do experimento até o momento atual. Além disso, para facilitar a análise posterior, todos os dados de monitoramento podem ser exportados para um arquivo no computador por meio de um botão de exportação disponível na página.

Figura 30 – Interface gráfica: Página de monitoramento.



Fonte: do autor.

5 RESULTADOS E AVALIAÇÃO

Este capítulo aborda um experimento de validação destinado a verificar o desempenho do sistema desenvolvido. Para isso, será preparada uma amostra de solo para a realização de diferentes ensaios de propagação de calor, testando sua capacidade de medição em tais cenários. Adicionalmente, será apresentado a implementação de uma simulação numérica, cujos parâmetros serão otimizados até alcançar consistência com os resultados observados experimentalmente. Por fim, será discutido o processo de obtenção do fluxo de calor no centro da sonda por meio da simulação otimizada, destacando a aplicabilidade do sistema desenvolvido.

5.1 PREPARAÇÃO DA AMOSTRA

Para o experimento de validação do sistema, será utilizada uma amostra de solo coletada em uma área onde opera uma refinaria, local monitorado por sondas de fluxo de calor do STMF devido à contaminação de petróleo ou seus derivados. Antes de ser empregada no experimento, essa amostra será submetida a um preparo adequado, incluindo tratamentos específicos, a fim de garantir que esteja nas condições apropriadas para a realização dos testes.

Inicialmente, foi realizada a moagem do solo, desagregando torrões para assegurar uma distribuição uniforme dentro do volume de controle, evitando espaços vazios que poderiam interferir no fluxo de calor. Em seguida, o solo foi peneirado para remover elementos indesejáveis incluídos durante a coleta. Além disso, foram realizados procedimentos para estimar a porosidade e a quantidade de água na amostra, parâmetros essenciais para analisar o fluxo de calor em diferentes níveis de saturação hídrica, uma vez que essas propriedades impactam diretamente os parâmetros térmicos do solo. A Figura 31 ilustra a amostra de solo após esses processos.

Figura 31 – Tratamento da amostra de solo.



(a) Amostra antes do tratamento



(b) Amostra depois do tratamento

Fonte: do autor.

5.2 EXPERIMENTO

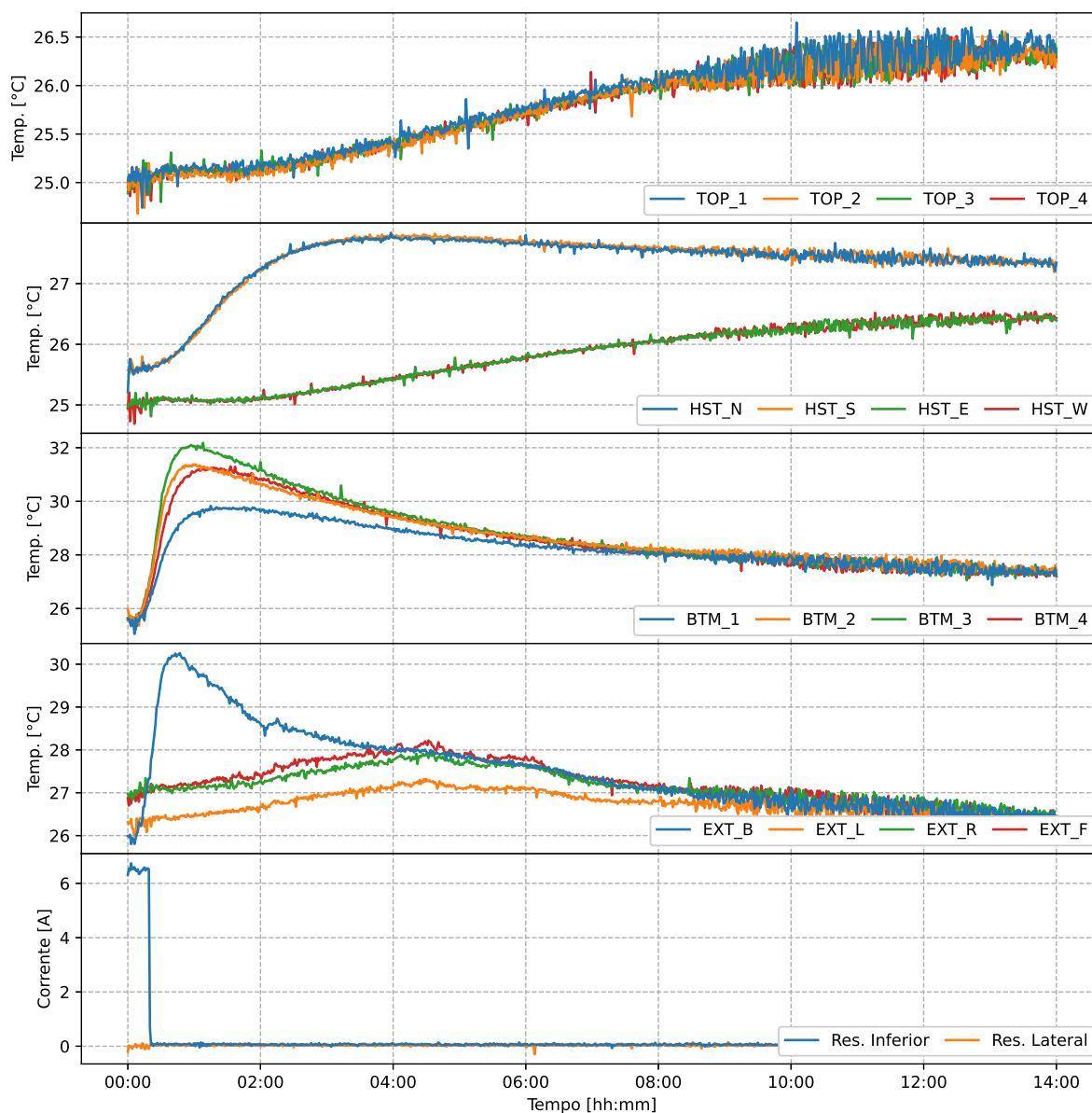
No experimento de validação serão realizados dois ensaios distintos, ambos baseados na aplicação de pulsos de calor no volume de controle em tempos bem definidos. Esses pulsos de calor geram fluxos térmicos no solo, resultantes da propagação do calor ao longo do volume de controle. No primeiro ensaio, o fluxo de calor será direcionado verticalmente, utilizando a resistência inferior como fonte de aquecimento. Já no segundo ensaio, será aplicado um fluxo de calor horizontal, empregando a resistência lateral para promover a propagação térmica nessa direção

Primeiramente, a sonda de fluxo de calor é posicionada no centro do volume de controle construído. Em seguida, o volume de controle é preenchido cuidadosamente com a amostra de solo preparada, controlando a variação da altura correspondente à massa aplicada para garantir densidade e compactação uniformes em todo o volume. Após o preenchimento, o volume de controle é tampado, e as temperaturas de monitoramento são acompanhadas até que o sistema atinja a termalização, assegurando um equilíbrio térmico antes do início do experimento.

Em ambos os ensaios, foi aplicado um pulso de calor com duração de 20 minutos, utilizando um tempo de amostragem dos sensores configurado para 60 segundos. Cada experimento teve uma duração total de pouco mais de 14 horas, permitindo observar a propagação do calor ao longo do tempo no volume de controle. Os dados experimentais obtidos são apresentados nas Figuras 32 e 33, correspondendo ao fluxo de calor vertical e horizontal, respectivamente.

No primeiro ensaio, com o fluxo de calor vertical, capturou-se com clareza a propagação do calor ao longo do tempo no volume de controle. A propagação térmica foi inicialmente identificada pelos sensores localizados mais próximos à fonte de calor, BTM_1, BTM_2, BTM_3 e BTM_4, que registraram grandes picos de temperatura no mesmo plano de propagação. À medida que o calor se propagou, observou-se um leve pico de temperatura nos sensores HST_N e HST_S, também no mesmo plano. Posteriormente, o calor alcançou os sensores HST_W e HST_E e, por fim, os sensores mais afastados da fonte, TOP_1, TOP_2, TOP_3 e TOP_4. Os picos de temperatura tornaram-se progressivamente mais fracos ao longo do volume devido à termalização do sistema. Além disso, o termômetro externo EXT_B, localizado na superfície externa do volume de controle, registrou uma significativa variação térmica, enquanto a temperatura ambiente (EXT_F) se manteve estável, sinalizando a transferência de uma parte do calor para fora do volume de controle, através da parede próxima do aquecedor.

Figura 32 – Dados experimentais obtidos para um fluxo de calor vertical.

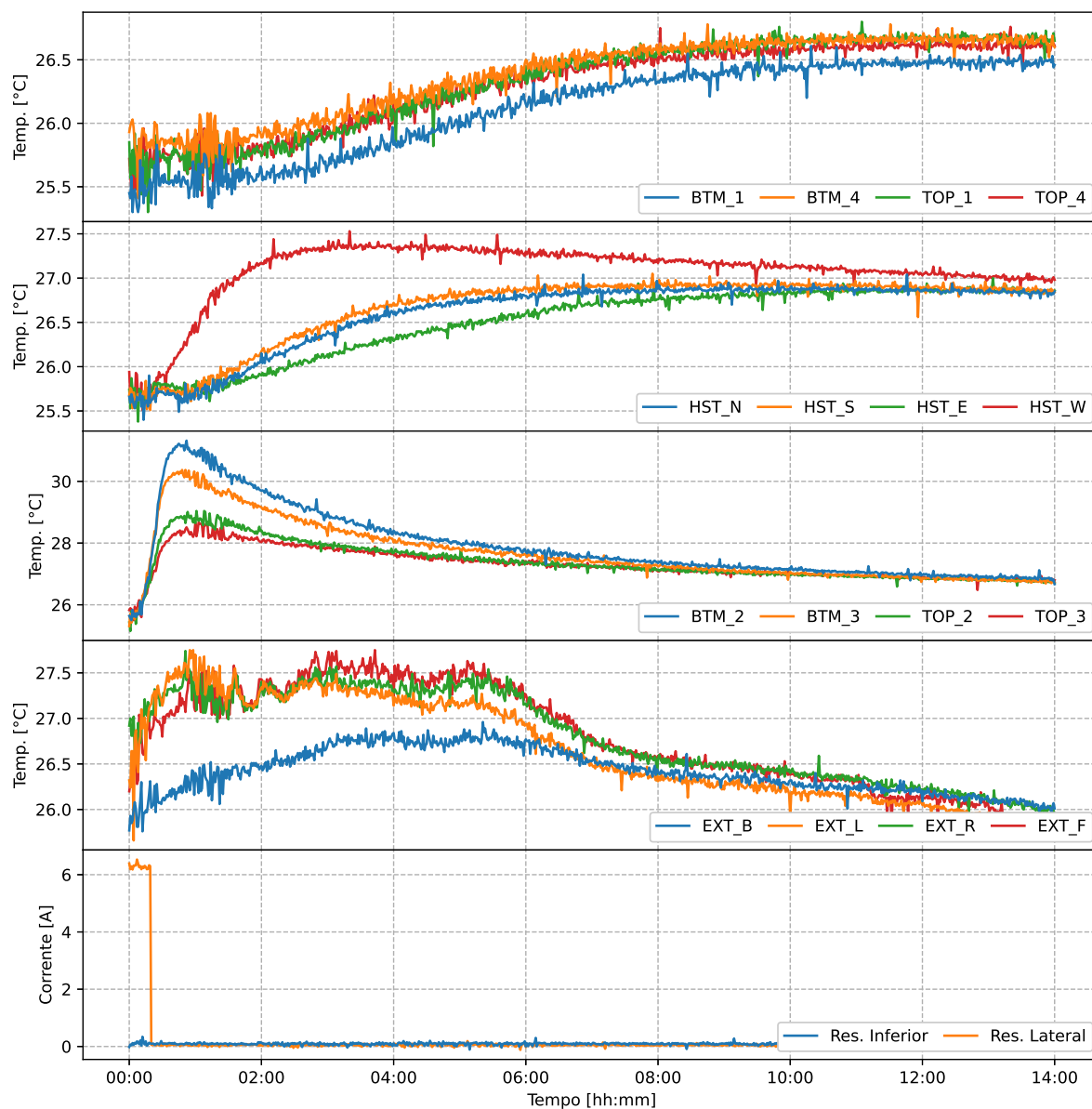


Fonte: do autor.

No segundo ensaio, com um fluxo de calor horizontal, também foi possível observar a propagação térmica no volume de controle. Inicialmente, os sensores mais próximos à fonte de calor lateral, BTM_2, BTM_3, TOP_2 e TOP_3, registraram picos de temperatura significativos. Em seguida, o calor propagou-se para os sensores da sonda, que produziram três curvas distintas devido às diferentes distâncias em relação à fonte. Primeiramente, o sensor HST_W detectou o calor, seguido pelos sensores HST_N e HST_S, e, por fim, pelo sensor HST_E. Posteriormente, os sensores mais distantes da fonte, BTM_1, BTM_4, TOP_1 e TOP_4, registraram o calor em menor intensidade. Diferentemente do primeiro ensaio, desta vez, não foram observadas

variações expressivas nos sensores externos, indicando menor interação térmica com o ambiente.

Figura 33 – Dados experimentais obtidos para um fluxo de calor horizontal.



Fonte: do autor.

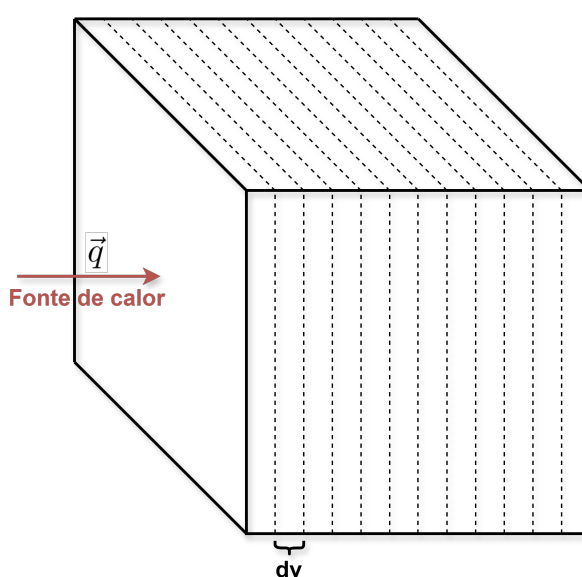
5.3 SIMULAÇÃO E OTIMIZAÇÃO

Nesta última seção será apresentada uma simulação computacional da propagação unidimensional do calor, definindo um cenário com condições de contorno semelhantes às do volume de controle utilizado no experimento. O principal objetivo desta simulação é otimizar os parâmetros térmicos do solo até alcançar uma consistência entre as curvas simuladas e as obtidas via experimento.

É importante destacar que a solução numérica utilizada para o simulador não foi desenvolvida pelo autor deste trabalho, mas sim elaborada pelo laboratório de pesquisa onde o autor esteve inserido, sendo essa de acesso restrito e não disponibilizada publicamente. O documento de referência, que orientou a implementação do simulador, está devidamente citado no Anexo A do documento sigiloso, com os devidos créditos.

Em síntese, a solução adotada se baseia no método dos volumes finitos para a resolução numérica da equação do calor, no qual o volume de controle é subdividido em pequenos elementos de volume na direção da propagação do calor, conforme ilustrado na Figura 34. A entrada de calor no volume de controle é determinada pela condição de contorno de Neumann, definida pela derivada da temperatura normal à fronteira do volume, diretamente relacionada com a lei de Fourier. Consequentemente, a propagação do calor dentro do volume ocorre por condução térmica através das superfícies que delimitam os elementos de volume. Neste trabalho, o simulador foi implementado utilizando essa solução numérica na linguagem de programação Python, a qual está detalhada no apêndice E.

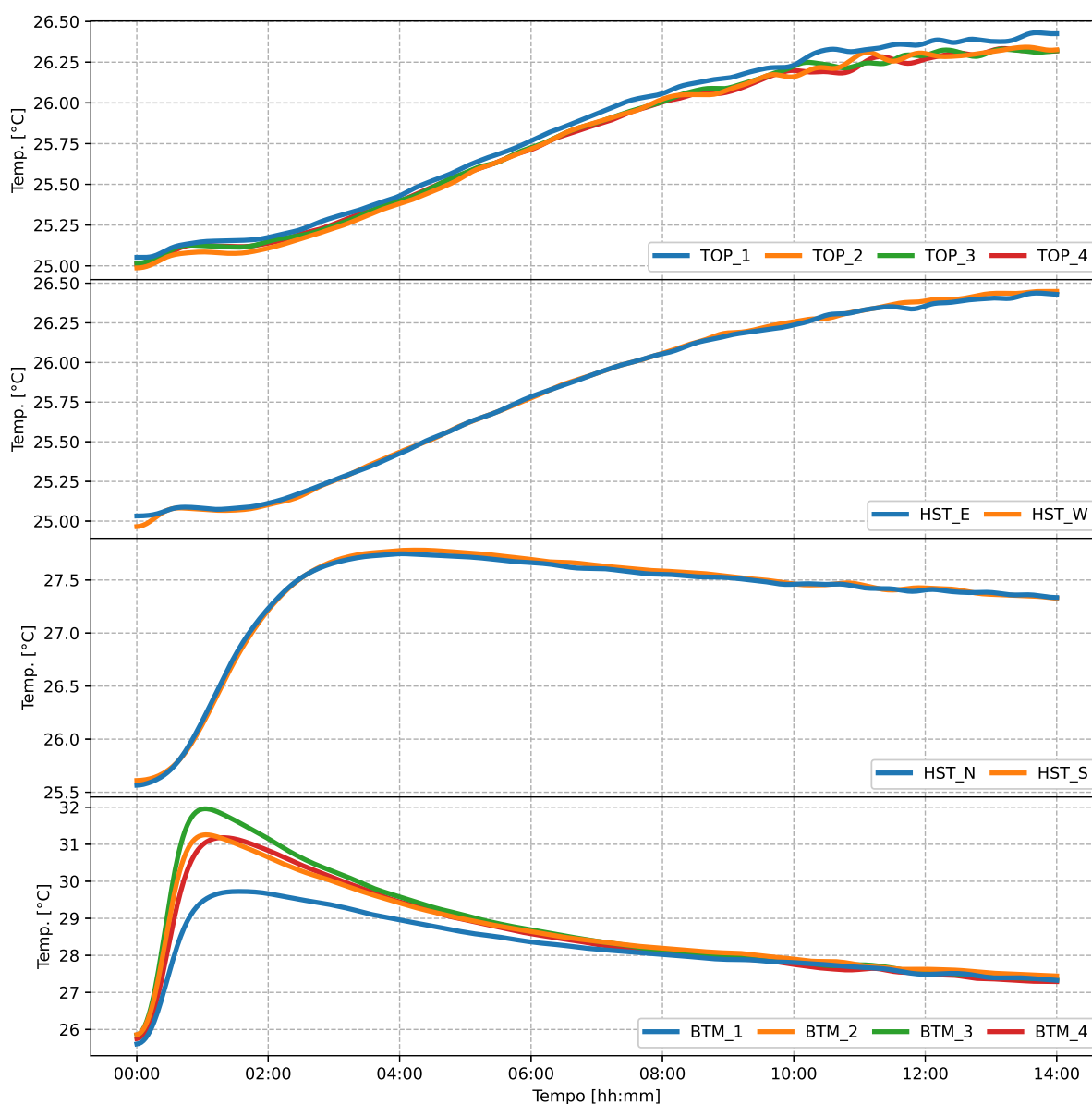
Figura 34 – Fracionamento do volume de controle em pequenos volumes finitos.



Fonte: do autor.

As curvas experimentais utilizadas para a otimização são provenientes do primeiro ensaio, no qual foi aplicado um fluxo de calor vertical. Com o objetivo de reduzir o aspecto ruidoso das medições e facilitar o processo de otimização, foi aplicado um filtro passa-baixas do tipo gaussiano a cada uma das curvas, suavizando as variações indesejadas nos dados. O resultado do tratamento dessas curvas é apresentado na Figura 35.

Figura 35 – Dados experimentais para o fluxo de calor vertical após aplicação do filtro.



Fonte: do autor.

Via simulação, são gerados sinteticamente os sinais de perfil de temperatura correspondentes às posições dos termômetros ao longo do volume de controle, obtidos pelo monitoramento dos valores de temperatura de cada elemento de volume associado às respectivas localizações dos termômetros.

Ao fim do experimento, as curvas experimentais de temperatura são comparadas diretamente com as curvas simuladas. Um algoritmo de otimização é empregado, de forma a encontrar os parâmetros térmicos do solo utilizados na simulação, como a condutividade térmica e a capacidade térmica volumétrica, buscando minimizar a diferença entre as curvas simulada e experimental.

O algoritmo adotado para a otimização dos parâmetros térmicos do solo foi o método dos mínimos quadrados, onde a função custo a ser minimizada é:

$$\begin{aligned} \arg \min_{k, C_V} \quad & \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^n \left(T_{exp,ij} - T_{sim,ij}(k, C_V) \right)^2 \\ \text{s.t.} \quad & 0,25 \leq k \leq 3, \\ & 1 \times 10^6 \leq C_V \leq 3 \times 10^6, \end{aligned} \quad (20)$$

sendo que k e C_V representam, respectivamente, a condutividade térmica e a capacidade térmica volumétrica, enquanto T_{exp} e T_{sim} são matrizes que contêm os dados de temperatura das curvas experimental e simulada. O índice i refere-se ao conjunto de dados de cada curva de temperatura, e j , ao respectivo elemento. Na otimização, foi considerado restrições nos parâmetros térmicos, limitando-os a uma faixa de valores realistas, comumente encontrados na literatura. A implementação do algoritmo de otimização pode ser consultada no Apêndice F, escrito na linguagem de programação Python.

A Tabela 2 apresenta os parâmetros térmicos do solo determinados por meio da otimização, com base nos dados experimentais do fluxo de calor vertical do primeiro ensaio. Observa-se que os valores obtidos estão em conformidade com a faixa de valores médios reportados na literatura.

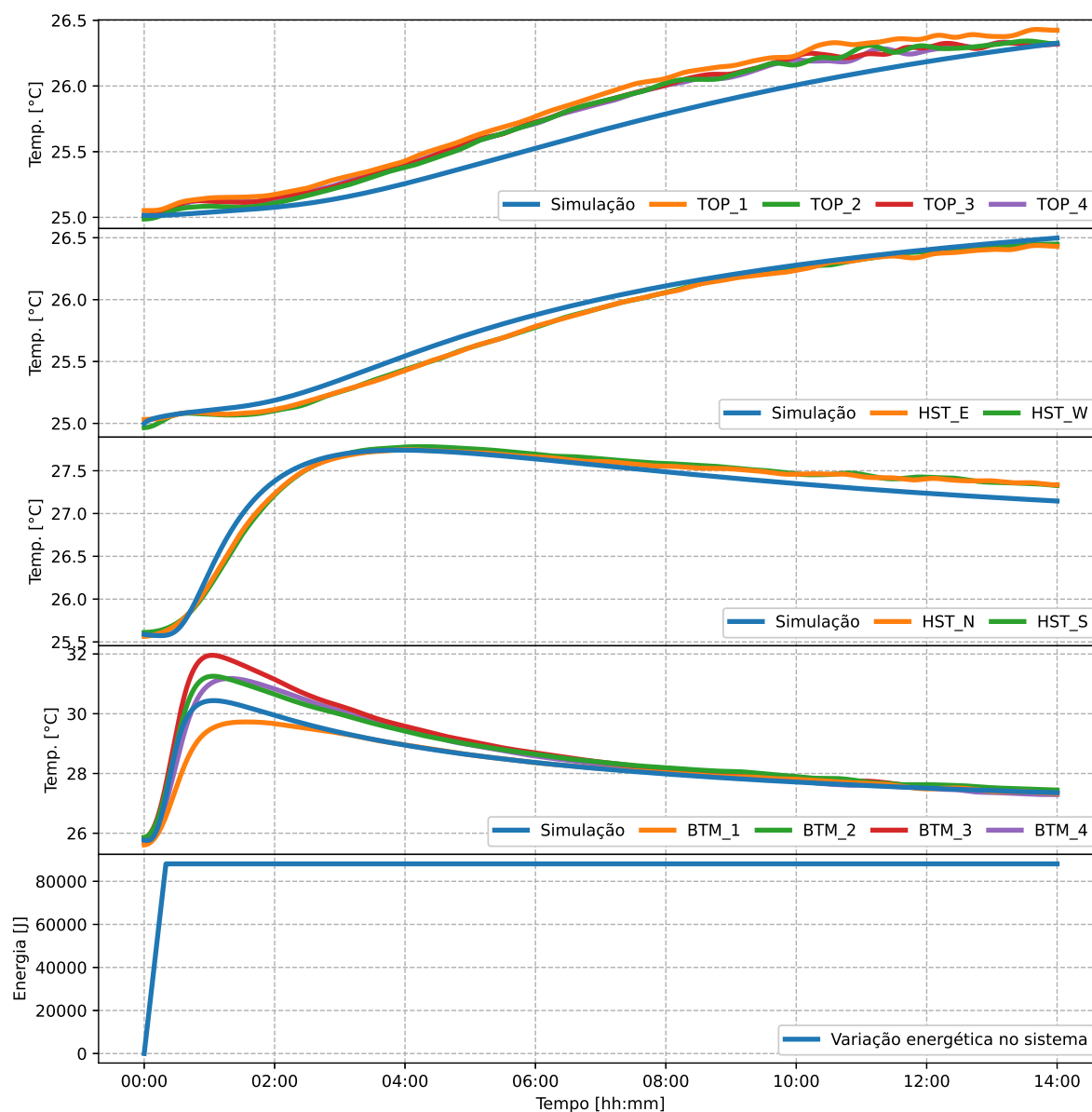
Tabela 2 – Comparação entre os parâmetros térmicos do solo obtidos via otimização e valores médios encontrados na literatura.

Parâmetro	Valor encontrado	Valor de referência	Unidade
Condutividade térmica (k)	0,7	0,25 - 2,9	W/(mK)
Capacidade térmica volumétrica (C_V)	$2,2 \times 10^6$	$1,92 \times 10^6$	J/(m ³ K)

Fonte: Adaptado de Duarte (2004).

As curvas simuladas e otimizadas são exibidas na Figura 36, juntamente com as curvas experimentais, para efeito de comparação.

Figura 36 – Comparação entre as curvas obtidas via simulação otimizada e as curvas experimental para o fluxo de calor vertical.

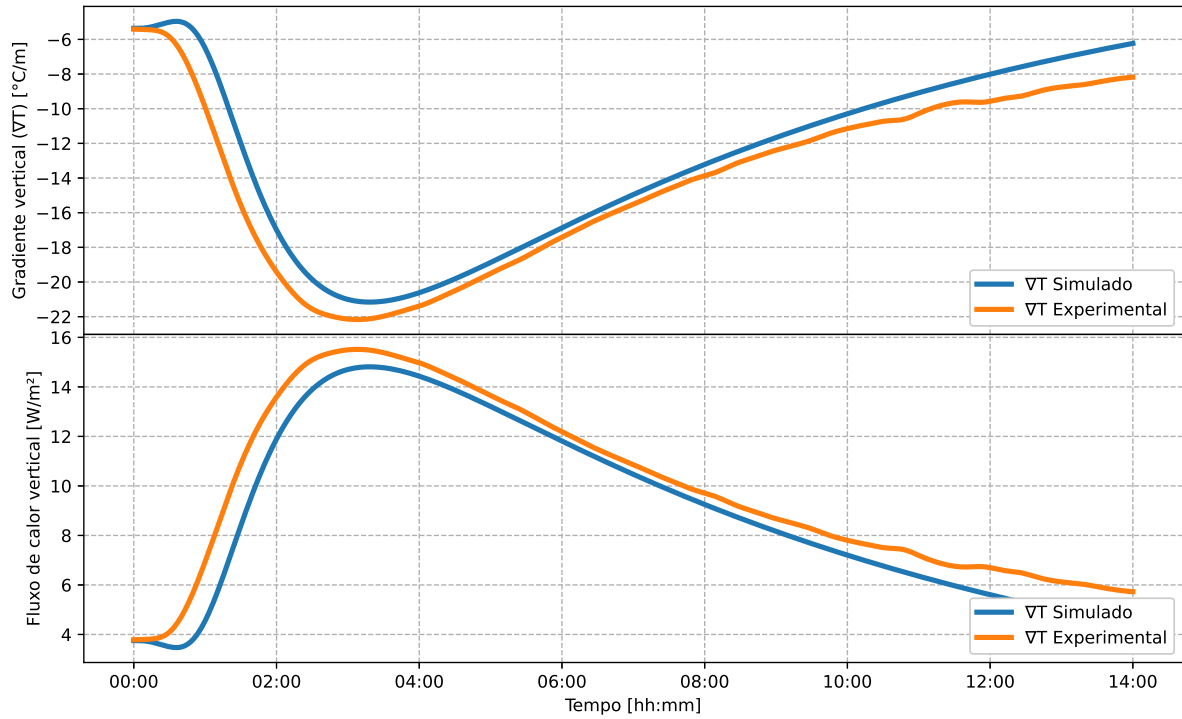


Fonte: do autor.

Uma vez concluída a etapa de otimização, com os parâmetros térmicos ajustados e as curvas simuladas apresentando consistência em relação às curvas experimentais, procede-se à determinação do gradiente de temperatura no centro da sonda por meio do simulador, comparando-o com o gradiente real calculado a partir dos sensores instalados na sonda. O gradiente de temperatura vertical real é obtido considerando a diferença entre a média dos sinais de temperatura dos sensores HST_W e HST_E e

a média dos sensores HST_N e HST_S, os quais estão separados verticalmente por uma distância de 0,109 m na sonda. Por outro lado, a simulação calcula o gradiente de temperatura vertical a partir da diferença de temperatura entre elementos de volume localizados no centro da sonda, separados por uma distância significativamente menor, de 0,01 m, permitindo uma estimativa mais acurada, já que o cálculo do gradiente deve considerar uma distância infinitesimal. A Figura 37 ilustra a comparação entre os gradientes verticais calculados por essas duas abordagens, além de apresentar o fluxo de calor correspondente, obtido utilizando a condutividade térmica do solo determinada durante o processo de otimização.

Figura 37 – Comparação entre o gradiente e o fluxo de calor vertical calculado a partir dos dados experimentais e dos dados simulados no centro da sonda.



Fonte: do autor.

Para uma análise mais quantitativa das duas abordagens, pode-se realizar uma comparação numérica entre as curvas de gradiente, utilizando o cálculo do erro absoluto percentual médio (MAPE) entre o gradiente real obtido pela sonda e o gradiente simulado no centro da sonda, conforme a expressão:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{\nabla T_{exp,i} - \nabla T_{sim,i}}{\nabla T_{exp,i}} \right| = 10,02\%, \quad (21)$$

sendo que ∇T_{exp} representa o gradiente experimental, enquanto ∇T_{sim} corresponde ao gradiente simulado. Cabe destacar que a diferença observada entre as curvas de

gradiente pode ser influenciada não apenas pela consideração de uma menor distância no cálculo do gradiente simulado, mas também por erros inerentes à simulação, que não consegue reproduzir integralmente o cenário experimental.

Todavia, em relação ao fluxo de calor, é importante ressaltar que a diferença entre os fluxos de calor calculados pelas duas abordagens pode ser ainda mais acentuada na prática, considerando que o parâmetro de condutividade térmica do solo, necessário para o cálculo do fluxo de calor, não é conhecido com exatidão quando a sonda é aplicada em campo. Essa incerteza contribui para uma estimativa menos acurada no cálculo do fluxo de calor da sonda real. Em contrapartida, na abordagem por simulação, os parâmetros térmicos do solo de interesse são encontrados durante a etapa de otimização, o que proporciona uma estimativa mais confiável.

Ressalta-se também que o presente estudo conduzido nesta seção teve como foco os dados do fluxo de calor vertical obtidos durante o primeiro ensaio experimental, sendo necessário repetir a análise para incluir os dados referentes ao fluxo de calor horizontal no solo. Dessa forma, a partir dessas análises, busca-se estabelecer relações unívocas entre os fluxos simulados e as medidas de temperatura cardinais, sejam elas instantâneas ou históricas, registradas pelos sensores da sonda. Assim, a modelagem da sonda de fluxo de calor visa desenvolver funções multiparamétricas capazes de processar as temperaturas medidas nos vértices da sonda e retornar o valor do fluxo de calor simulado e otimizado no centro da sonda, resultando em um modelo calibrado e confiável para a estimativa do fluxo de calor.

As etapas relacionadas à modelagem da sonda de fluxo de calor, no entanto, não serão abordadas neste documento devido à necessidade de um período mais extenso de estudo e desenvolvimento. Essas atividades permanecem como uma perspectiva para trabalhos futuros, a serem conduzidos no laboratório de pesquisa responsável pelo projeto.

6 CONCLUSÃO

Esta monografia apresentou o desenvolvimento de um Sistema Termodinâmico para Caracterização de Sondas de Fluxo de Calor, um ambiente de testes controlados para o estudo de sinais de temperatura de sondas de fluxo de calor de STMFs. O sistema desenvolvido viabiliza a criação e validação de novos conceitos de sonda de fluxo de calor e o estudo de modelos calibrados, superando certas limitações físicas e instrumentais da sonda, como a distância finita entre os pontos de medição no cálculo do gradiente de temperatura e a resolução limitada dos sensores de temperatura, além de determinar os parâmetros térmicos do solo do local de interesse, atendendo à necessidade de considerar as propriedades térmicas específicas de cada área analisada.

Os objetivos traçados no início deste projeto foram alcançados de forma satisfatória. Durante o desenvolvimento, foi projetado e construído um volume de controle para o sistema, com um bom isolamento térmico do ambiente. Além disso, foi elaborado um gabinete de controle completo, capaz de gerenciar e monitorar o sistema, acompanhado de um *software* de interface gráfica para configuração dos experimentos pelo usuário. Um algoritmo de simulação também foi apresentado, e, por meio de otimizações, alcançou-se consistência com o cenário experimental. O experimento de validação confirmou o funcionamento do sistema, evidenciando que o projeto alcançou com êxito um MVP funcional.

Em relação a possíveis aprimoramentos, destaca-se a importância de melhorias na isolamento térmica do volume de controle. Durante um dos ensaios realizados, foi observada uma variação nos termômetros externos ao longo do experimento, sugerindo a presença de fluxos de calor parasitas entrando ou saindo do volume de controle. Conferir uma isolamento térmica mais eficiente pode mitigar essa questão. Outra abordagem complementar seria o aprimoramento do simulador. Atualmente, a simulação apresentada neste trabalho considera como condições de contorno apenas o fluxo de calor proveniente das resistências de aquecimento. Incorporar esses fluxos de calor parasitas ao simulador o aperfeiçoaria, melhorando a representação dos cenários experimentais.

Para trabalhos futuros, considerando que os resultados obtidos neste estudo demonstram a possibilidade de determinar o fluxo de calor no centro da sonda por meio de uma simulação experimentalmente otimizada, sugere-se a investigação de modelos que relacionem diretamente as temperaturas medidas pela sonda com o fluxo de calor simulado e otimizado. Esse avanço permitiria aprimorar a estimativa do fluxo de calor em campo, tanto para condições atuais quanto históricas, utilizando os dados de temperatura previamente coletados pela sonda.

REFERÊNCIAS

BERGMAN, Theodore L.; LAVINE, Adrienne S. **Fundamentals of Heat and Mass Transfer**. 8. ed. [S.l.]: Wiley, 2020. P. 4. ISBN 978-1119722489.

ÇENGEL, Yunus A.; BOLES, Michael A. **Termodinâmica**. 7. ed. [S.l.]: AMGH, 2013. P. 92. ISBN 978-8580552010.

CNBLOGS. **Electron main process and rendering process**. Disponível em: <https://www.cnblogs.com/pengjiali/p/16732090.html>. Acesso em: 18 nov. 2024.

CONAMA. **Resolução nº 420, de 28 de dezembro de 2009**. [S.l.: s.n.], 2009. Dispõe sobre os critérios e valores orientadores de qualidade do solo quanto à presença de substâncias químicas. Disponível em: <https://cetesb.sp.gov.br/areas-contaminadas/wp-content/uploads/sites/17/2017/09/resolucao-conama-420-2009-gerenciamento-de-acrs.pdf>.

DEWITT, David P.; BERGMAN, Theodore L.; LAVINE, Adrienne S.; INCROPERA, Frank P. **Fundamentals of Heat and Mass Transfer**. 6. ed. [S.l.]: John Wiley & Sons, 2006. P. 935–940. ISBN 978-0471457282.

DIGIKEY. **103AT-11**. 2024. Disponível em: <https://www.digikey.pt/pt/products/detail/semitec-usa-corp/103AT-11/16578928>. Acesso em: 18 nov. 2024.

DUARTE, Anna Paula Lougon. **Avaliação de Propriedades Termo-Hidráulicas de Solos Requeridas na Aplicação da Técnica de Dessorção Térmica**. 2004. Tese de Doutorado – Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio. Disponível em: <https://www.maxwell.vrac.puc-rio.br/colecao.php?strSecao=resultado&nrSeq=4794&idi=1>.

EASYTRONICS. **ACS712-20A Sensor de Corrente 20A Bidirecional | ACS712-20A**. 2024a. Disponível em: <https://www.easytronics.com.br/acs712-20a-sensor-de-corrente-20a-bidirecional-acrs712-20a>. Acesso em: 17 nov. 2024.

EASYTRONICS. **Módulo Amplificador de Instrumentação CJMCU-333 | INA333**. 2024b. Disponível em: <https://www.easytronics.com.br/modulo-amplificador-de-instrumentacao-cjmcu-333>. Acesso em: 17 nov. 2024.

ELECTRONJS. **What is Electron?** Disponível em:

<https://www.electronjs.org/docs/latest>. Acesso em: 18 nov. 2024.

ELETROGATE. **Módulo Relé 2 Canais 3V 10A com Borne KRE para ESP32.** 2024.

Disponível em: <https://www.eletrogate.com/modulo-rele-2-canais-3v-10a-com-borne-kre-para-esp32>. Acesso em: 17 nov. 2024.

EUZEBIO, Camilla Szerman; SILVEIRA RANGEL, Giovanna da;

MARQUES, Rejane Côrrea. OIL SPILLS AND ITS ENVIRONMENTAL AND HUMAN HEALTH IMPACTS. **Revista Brasileira De Ciências Ambientais**, v. 52, p. 79–98,

2019. DOI: <https://doi.org/10.5327/Z2176-947820190472>.

FREERTOS. **What is FreeRTOS?** Disponível em:

<https://www.freertos.org/Why-FreeRTOS/What-is-FreeRTOS>. Acesso em: 18 nov. 2024.

GARG, Sanjay; NEWELL, Charles J.; KULKARNI, Poonam R.; KING, David C.;

ADAMSON, David T.; RENNO, Maria Irianni; SALE, Tom. Overview of Natural Source Zone Depletion: Processes, Controlling Factors, and Composition Change.

Groundwater Monitoring & Remediation, v. 37, p. 62–81, 2017. DOI:

<https://doi.org/10.1111/gwmr.12219>.

HALLIDAY, David; RESNICK, Robert; WALKER, Jearl. **Fundamentals of Physics**. 10. ed. [S.l.]: Wiley, 2013. P. 524. ISBN 978-1118230718.

HR2 ELETRONICA. **Botão Led Vermelho Liga Desliga Chave Gangorra.** 2024.

Disponível em: <https://www.hr2.net.br/produtos/botao-led-vermelho-liga-desliga-chave-gangorra/>. Acesso em: 17 nov. 2024.

LILYGO. **ESP32 T8.** 2024. Disponível em: <https://lilygo.cc/products/t8-c3>.

Acesso em: 17 nov. 2024.

MAINI, Anil K. **Digital Electronics: Principles, Devices and Applications**. 1. ed.

[S.l.]: Wiley, 2007. P. 269. ISBN 978-0470032145.

MOUSER ELECTRONICS. **AT THERMISTOR**. [S.l.: s.n.], 2024. Disponível em:

https://www.mouser.com/datasheet/2/362/semitec_atthermistor-1202913.pdf.

Acesso em: 17 nov. 2024.

NOTÍCIAS DA UFSC. **Núcleo Ressacada de Pesquisa em Meio Ambiente é selecionado para receber recursos da Embrapii.** 2016. Disponível em: <https://noticias.ufsc.br/2016/01/nucleo-ressacada-de-pesquisa-em-meio-ambienteto-e-selecionado-para-receber-recursos-da-embrapii/>. Acesso em: 16 nov. 2024.

NOTÍCIAS DA UFSC. **Projeto da UFSC com a Petrobras pretende reduzir danos de vazamento de combustível.** 2024. Disponível em: <https://noticias.ufsc.br/2024/01/projeto-da-ufsc-com-a-petrobras-pretende-reduzir-danos-de-vazamentos-de-combustiveis/>. Acesso em: 16 nov. 2024.

NOTÍCIAS DA UFSC. **UFSC inaugura Núcleo Ressacada de Pesquisas em Meio Ambiente.** 2014. Disponível em: <https://noticias.ufsc.br/2014/04/ufsc-inaugura-nucleo-ressacada-de-pesquisas-em-meio-ambiente/>. Acesso em: 16 nov. 2024.

REMA. **STMF.** 2024. Disponível em: <https://scbn.com.br/stmf/>. Acesso em: 18 nov. 2024.

ROBOCORE. **Módulo Regulador de Tensão Ajustável LM2596 3A.** 2024. Disponível em: https://www.robocore.net/regulador-de-tensao/modulo-regulador-de-tensao-ajustavel-lm2596-3a?gad_source=1. Acesso em: 17 nov. 2024.

SABER360. **Multiplexador Digital.** Disponível em: <https://www.saber360.com.br/multiplexadores-e-dmux>. Acesso em: 18 nov. 2024.

SARAVATI. **Módulo Multiplexador 16 Canais Analógico Digital CD74HC4067.** 2024. Disponível em: <https://www.saravati.com.br/modulo-multiplexador-16-canais-analogico-digital-cd74hc4067.html>. Acesso em: 17 nov. 2024.

SCHNEIDER, Marcio Roberto; GIACHINI, Admir José;
BAESSA, Marcus Paulus Martins; SOUZA FILHO, André Moreira de;
SORIANO, Adriana Ururahy. Monitoramento e modelagem da temperatura como estratégia de baixo custo para demonstrar a atenuação natural de vazamentos onshore. **Rio Oil & Gas Expo And Conference**, 2020. ISSN 2525-7579. Disponível em: <https://biblioteca.ibp.org.br/scripts/bnmapi.exe?router=upload/33634>.

SCHULTZ, Mitchel. **Grob's Basic Electronics**. 12. ed. [S.l.]: McGraw Hill, 2015. P. 58. ISBN 978-0073373874.

SEDRA, Adel S.; SMITH, Kenneth C. **Microelectronic Circuits**. 7. ed. [S.l.]: Oxford University Press, 2014. P. 82. ISBN 978-0199339136.

SINOVA. **Pesquisa do Núcleo Ressacada de Pesquisas em Meio Ambiente da UFSC, em parceria com a PETROBRAS, é reconhecida com Prêmio Inventor Petrobras 2024**. 2024. Disponível em:

<https://sinova.ufsc.br/2024/05/08/pesquisa-do-nucleo-ressacada-de-pesquisas-em-meio-ambiente-da-ufsc-em-parceria-com-a-petrobras-e-reconhecida-com-premio-inventor-petrobras-2024/>. Acesso em: 16 nov. 2024.

SWEENEY, Robert E.; RIRIE, G. Todd. Temperature as a Tool to Evaluate Aerobic Biodegradation in Hydrocarbon Contaminated Soil. **Groundwater Monitoring & Remediation**, v. 34, p. 41–50, 2014. DOI: <https://doi.org/10.1111/gwmr.12064>.

TEXAS INSTRUMENTS. **INA333 Micro-Power (50uA), Zero-Drift, Rail-to-Rail Out Instrumentation Amplifier**. [S.l.: s.n.], 2015. Disponível em:

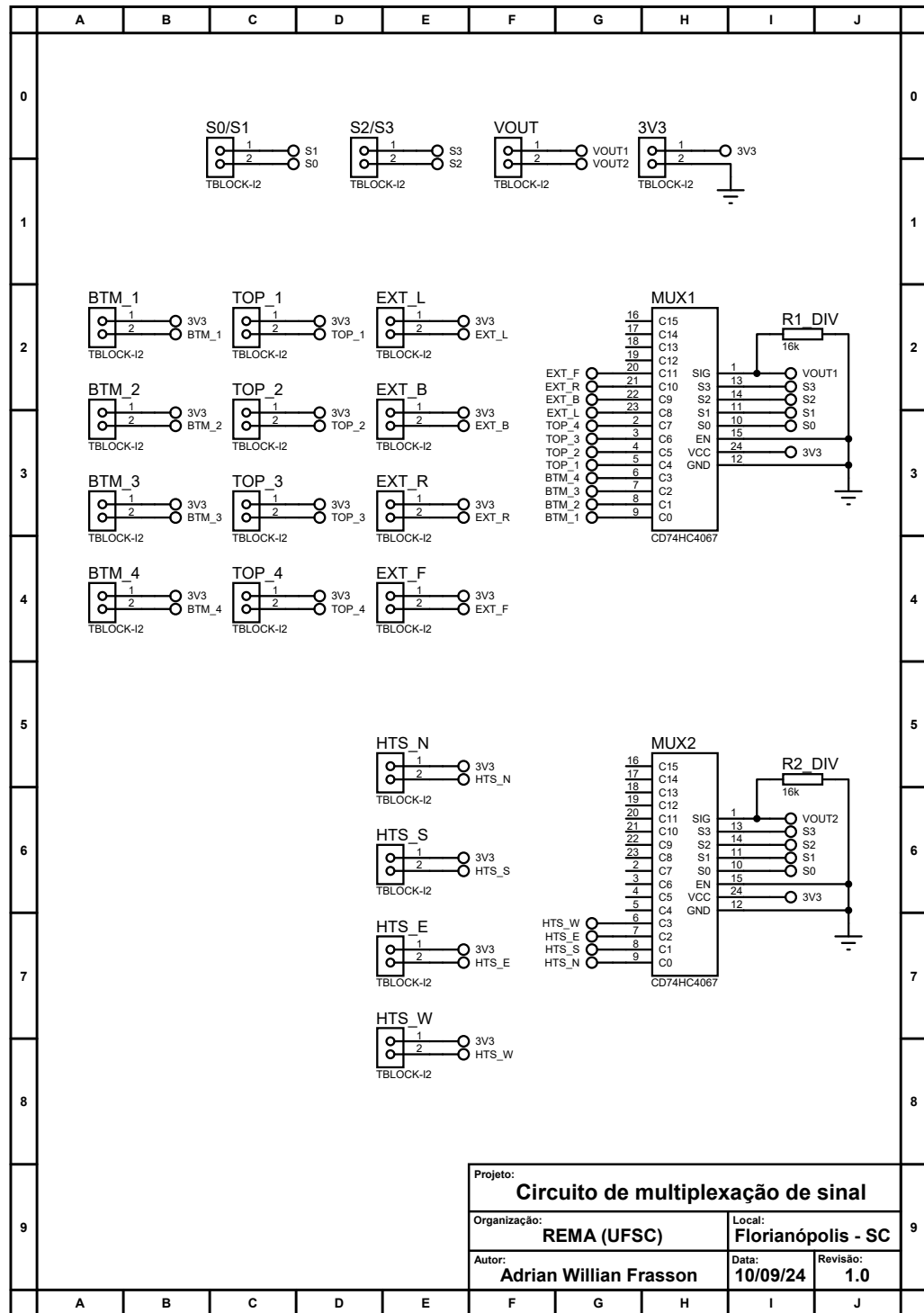
<https://www.ti.com/lit/ds/symlink/ina333.pdf?ts=1735042858907>. Acesso em: 17 nov. 2024.

TEXAS INSTRUMENTS. **LM2596 SIMPLE SWITCHER® Power Converter 150-kHz 3-A Step-Down Voltage Regulator**. [S.l.: s.n.], 2023. Disponível em:

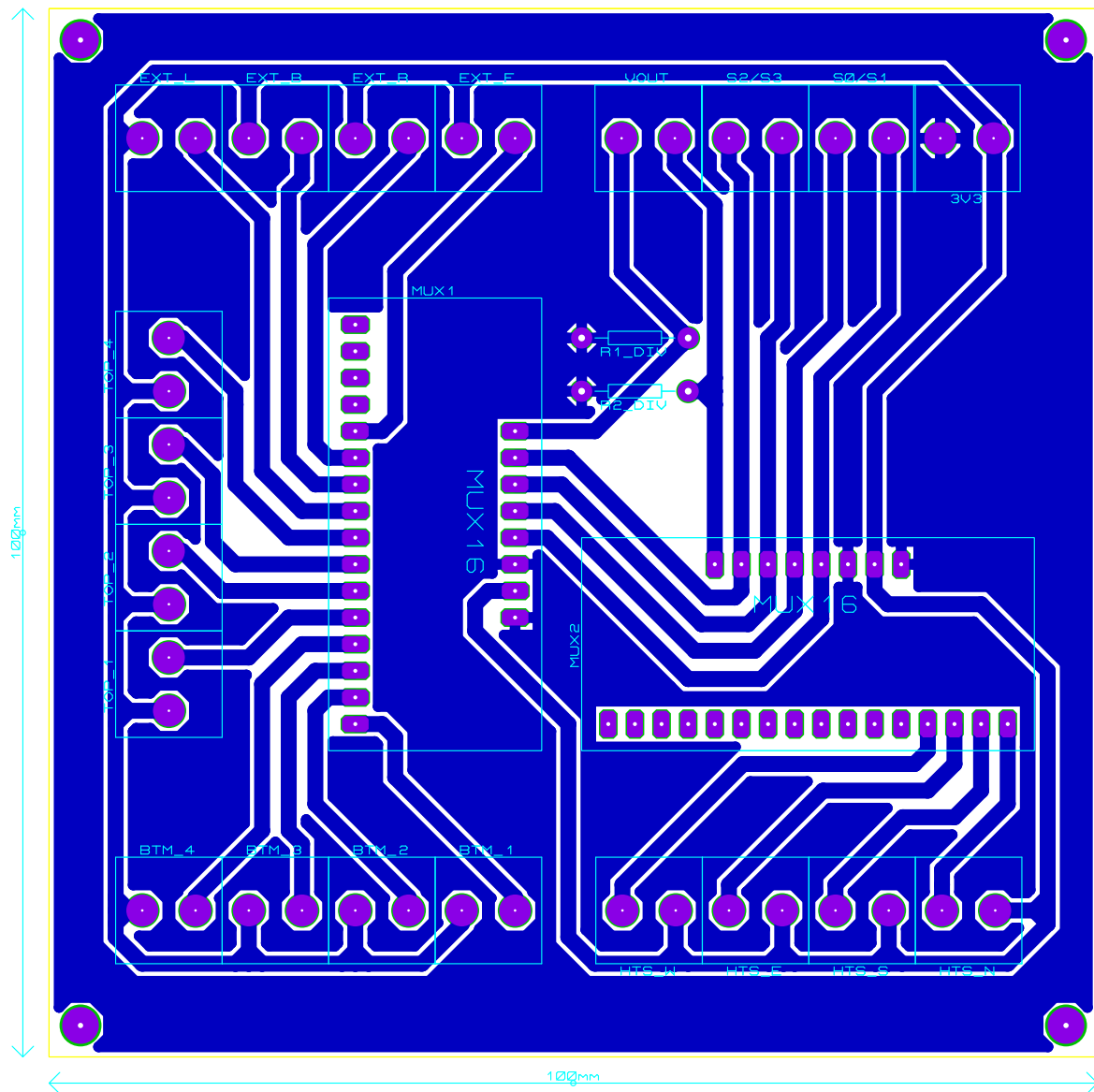
<https://www.ti.com/lit/ds/symlink/lm2596.pdf>. Acesso em: 17 nov. 2024.

APÊNDICE A – CIRCUITO DE MULTIPLEXAÇÃO

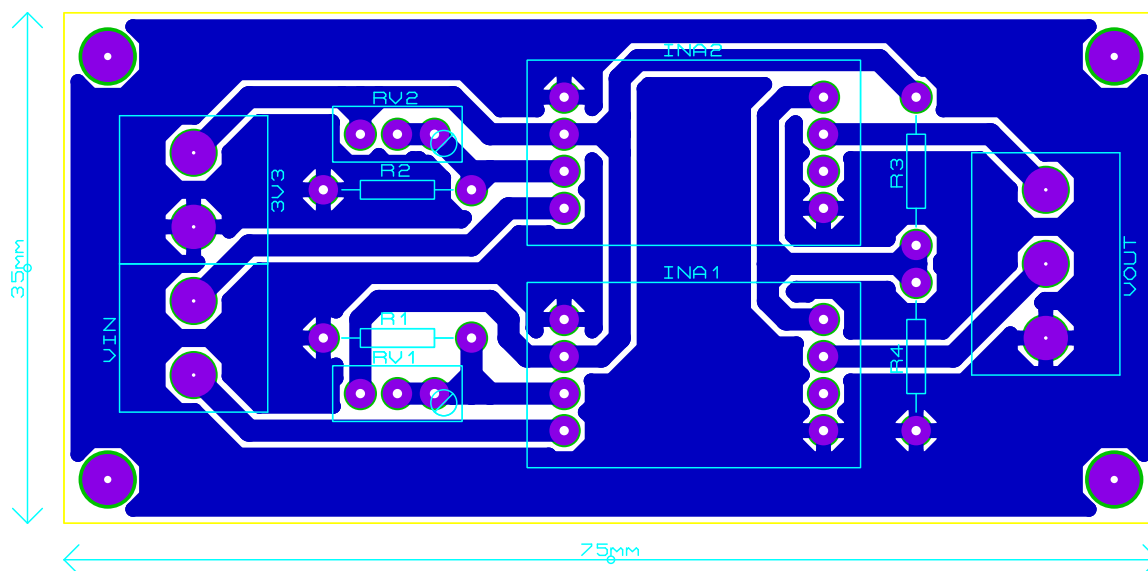
Figura 38 – Circuito de multiplexação: Conexões.



Fonte: do autor.

Figura 39 – Circuito de multiplexação: *Layout*.

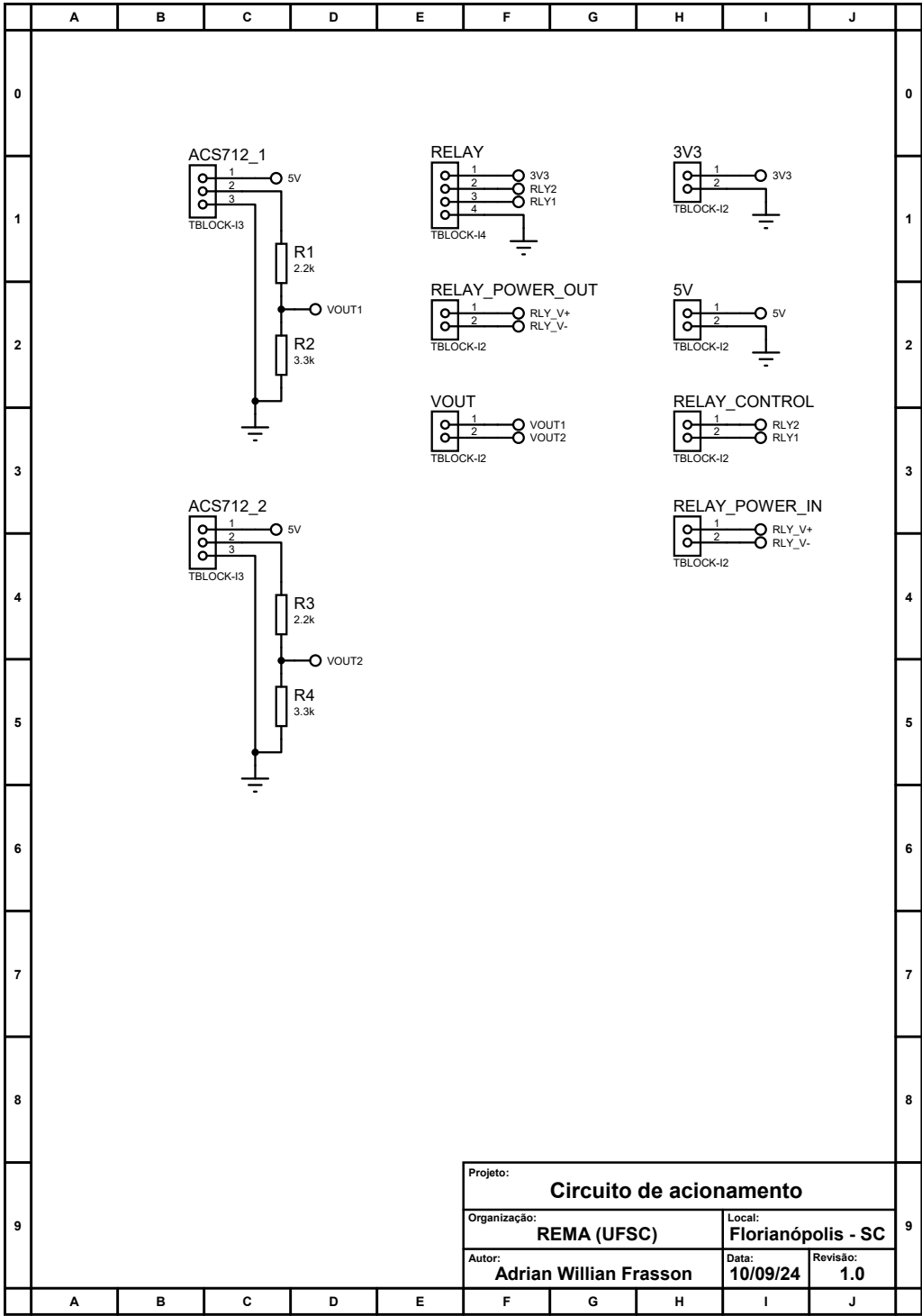
Fonte: do autor.

Figura 41 – Circuito de amplificação: *Layout*.

Fonte: do autor.

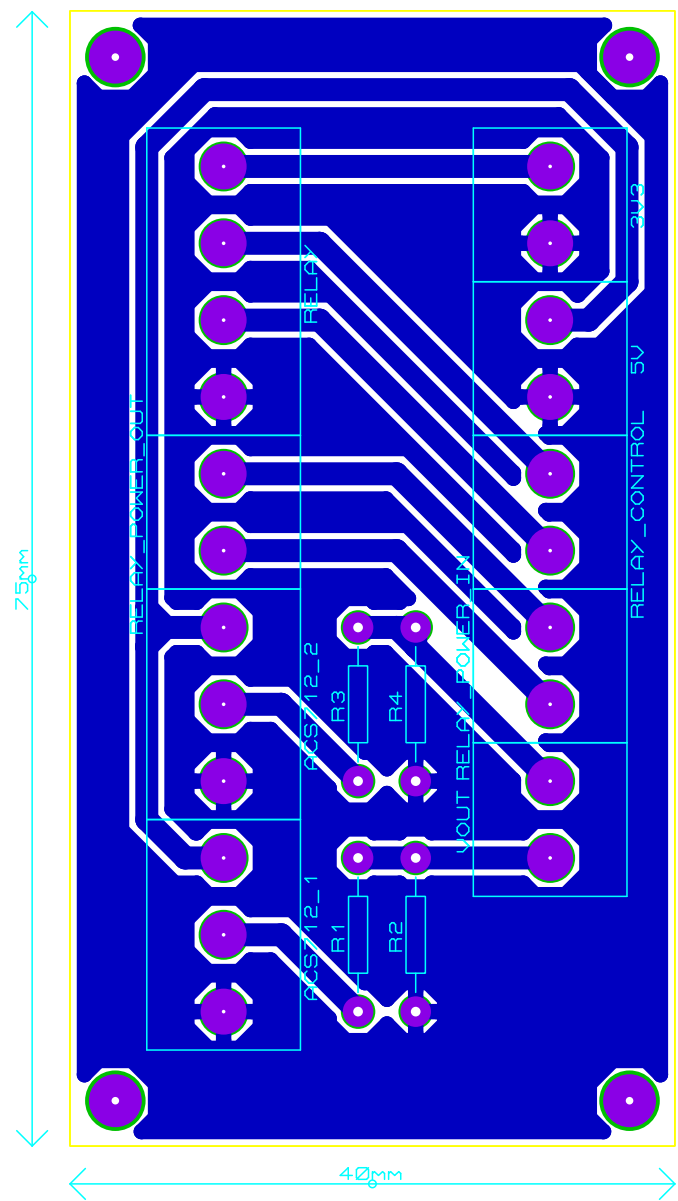
APÊNDICE C – CIRCUITO DE ACIONAMENTO

Figura 42 – Circuito de acionamento: Conexões.



Fonte: do autor.

Figura 43 – Circuito de acionamento: *Layout*.



Fonte: do autor.


```

1 // Libs
2 #include "RincProbe.h" // Classe para criar um objeto de medição.
3 #include "StorageCard.h" // Classe para tratar o armazenamento SD.
4
5 // Pins
6 // Pinos dos sinais dos multiplexadores da sonda e dos termistores de referência.
7 #define PIN_SIG_HST 34
8 #define PIN_SIG_REF 32
9
10 // Pinos para seleção de sinal dos multiplexadores.
11 #define PIN_MUX_S0 5
12 #define PIN_MUX_S1 21
13 #define PIN_MUX_S2 22
14 #define PIN_MUX_S3 23
15
16 #define PIN_AMP_BT_M 35 // Pino de leitura de corrente da resistência inferior.
17 #define PIN_AMP_LFT 33 // Pino de leitura de corrente da resistência lateral.
18 #define PIN_RLY_BT_M 27 // Pino de acionamento do relé da resistência inferior.
19 #define PIN_RLY_LFT 26 // Pino de acionamento do relé da resistência lateral.
20
21 // Pinos para comunicação com o módulo de armazenamento.
22 #define PIN_SD_MISO 2
23 #define PIN_SD_MOSI 15
24 #define PIN_SD_SCK 14
25 #define PIN_SD_CS 13
26
27 // ADC Scale
28 int adc_scale_count = 13;
29
30 // Escala calibrada tensão/ADC da entrada analógica do microcontrolador.
31 int adc_scale_value[] = { 0, 215, 575, 935, 1310, 1675, 2040,
32 2400, 2775, 3190, 3760, 4080, 4095 };
33 float adc_scale_volt[] = { 0.00, 0.30, 0.60, 0.90, 1.20, 1.50, 1.80,
34 2.10, 2.40, 2.70, 3.00, 3.15, 3.30 };
35
36 adc_scale adc_scale_custom = { adc_scale_count, adc_scale_volt, adc_scale_value, 1.57, 1.0 };
37
38 // Temperature Scale
39 int temp_scale_count = 28;
40
41 // Escala calibrada temperatura/tensão dos termistores da sonda.
42 float temp_hst_scale_value[] = { 10.5, 12.5, 14.5, 16.8, 18.5, 20.0, 22.0, 24.0, 26.0, 28.0,
43 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 42.0, 44.0, 46.0, 48.0,
44 50.0, 52.0, 54.0, 56.0, 58.0, 60.0, 62.0, 64.0 };
45 float temp_hst_scale_volt[] = { 0.000, 0.000, 0.055, 0.571, 0.904, 1.130, 1.510, 1.850,
46 2.185, 2.515, 2.808, 3.093, 3.300, 3.300, 3.300, 3.300,
47 3.300, 3.300, 3.300, 3.300, 3.300, 3.300, 3.300, 3.300,
48 3.300, 3.300, 3.300, 3.300 };
49
50 // Escala calibrada temperatura/tensão dos termistores de referência.
51 float temp_ref_scale_value[] = { 10.5, 12.5, 14.5, 16.8, 18.5, 20.0, 22.0, 24.0, 26.0, 28.0,
52 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 42.0, 44.0, 46.0, 48.0,
53 50.0, 52.0, 54.0, 56.0, 58.0, 60.0, 62.0, 64.0 };
54 float temp_ref_scale_volt[] = { 0.088, 0.335, 0.464, 0.642, 0.780, 0.872, 1.030, 1.168,
55 1.305, 1.449, 1.568, 1.677, 1.805, 1.918, 2.012, 2.115,
56 2.220, 2.320, 2.418, 2.480, 2.562, 2.650, 2.726, 2.788,
57 2.856, 2.923, 2.993, 3.050 };
58
59 // Current Scale
60 int current_scale_count = 11;
61
62 // Escala calibrada corrente/tensão do sensor de corrente da resistência inferior.
63 float current_btm_scale_value[] = { 0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0 };
64 float current_btm_scale_volt[] = { 1.515, 1.545, 1.574, 1.603, 1.632, 1.662,
65 1.691, 1.720, 1.750, 1.779, 1.809 };

```

```

66
67 // Escala calibrada corrente/tensão do sensor de corrente da resistência lateral.
68 float current_lft_scale_value[] = { 0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0 };
69 float current_lft_scale_volt[] = { 1.489, 1.519, 1.549, 1.579, 1.609, 1.639,
70                                   1.670, 1.700, 1.730, 1.760, 1.790 };
71
72 // Measurement Objects
73 // Objetos para lidar com os multiplexadores.
74 RincMux mux_hst;
75 RincMux mux_ref;
76
77 // Objetos de medição para as correntes.
78 RincProbe current_btm;
79 RincProbe current_lft;
80
81 // Objetos de medição para os termistores da sonda.
82 RincProbe temp_hst_n;
83 RincProbe temp_hst_s;
84 RincProbe temp_hst_e;
85 RincProbe temp_hst_w;
86
87 // Objetos de medição para os termistores de referência.
88 RincProbe temp_btm_1;
89 RincProbe temp_btm_2;
90 RincProbe temp_btm_3;
91 RincProbe temp_btm_4;
92 RincProbe temp_top_1;
93 RincProbe temp_top_2;
94 RincProbe temp_top_3;
95 RincProbe temp_top_4;
96 RincProbe temp_ext_b;
97 RincProbe temp_ext_l;
98 RincProbe temp_ext_r;
99 RincProbe temp_ext_f;
100
101 // SD Card Object
102 StorageCard storage_card(PIN_SD_MISO, PIN_SD_MOSI, PIN_SD_SCK, PIN_SD_CS);
103
104 // Tasks Handler
105 // Handlers para lidar com tasks e timers do FreeRTOS.
106 TaskHandle_t handler_task = NULL;
107 TimerHandle_t handler_timer = NULL;
108
109 // Process Parameters
110 // O sistema pode estar em apenas 3 estados de funcionamento.
111 enum status { PROCESS_START, // Experimento prestes a começar.
112              PROCESS_RUN,    // Experimento em execução.
113              PROCESS_STOP }; // Experimento parado.
114
115 status process_status = PROCESS_STOP; // Estado inicial do sistema.
116
117 String file_name = "data"; // Nome do arquivo de medições.
118 bool resistance_btm = false; // Resistência inferior on/off.
119 bool resistance_lft = true; // Resistência lateral on/off.
120 int resistance_Tu = 600000; // Tempo de resistências ligadas [s].
121 int measure_Ts = 60000; // Tempo de amostragem [s].
122 int64_t timestamp = 0; // Timestamp desde o boot do esp32 [ms].
123
124 //
125 void setup() {
126     Serial.begin(115200); // Inicializa a comunicação serial.
127     storage_card.begin(); // Inicializa o módulo de armazenamento.
128
129     // Configura os multiplexadores dos sinais da sonda e dos sinais de referência.
130     mux_hst.begin(PIN_SIG_HST, PIN_MUX_S3, PIN_MUX_S2, PIN_MUX_S1, PIN_MUX_S0);
131     mux_ref.begin(PIN_SIG_REF, PIN_MUX_S3, PIN_MUX_S2, PIN_MUX_S1, PIN_MUX_S0);
132

```

```
133 // Vincula os objetos de medição de corrente com seus respectivos pinos.
134 current_btm.begin(PIN_AMP_BTM);
135 current_lft.begin(PIN_AMP_LFT);
136
137 // Vincula os objetos de medição de temperatura
138 // com seus respectivos multiplexadores e canais.
139 temp_hst_n.begin(0, &mux_hst);
140 temp_hst_s.begin(1, &mux_hst);
141 temp_hst_e.begin(2, &mux_hst);
142 temp_hst_w.begin(3, &mux_hst);
143 temp_btm_1.begin(0, &mux_ref);
144 temp_btm_2.begin(1, &mux_ref);
145 temp_btm_3.begin(2, &mux_ref);
146 temp_btm_4.begin(3, &mux_ref);
147 temp_top_1.begin(4, &mux_ref);
148 temp_top_2.begin(5, &mux_ref);
149 temp_top_3.begin(6, &mux_ref);
150 temp_top_4.begin(7, &mux_ref);
151 temp_ext_b.begin(8, &mux_ref);
152 temp_ext_l.begin(9, &mux_ref);
153 temp_ext_r.begin(10, &mux_ref);
154 temp_ext_f.begin(11, &mux_ref);
155
156 // Vincula a escala calibrada de tensão.
157 current_btm.setCustomADC(&adc_scale_custom);
158 current_lft.setCustomADC(&adc_scale_custom);
159 temp_hst_n.setCustomADC(&adc_scale_custom);
160 temp_hst_s.setCustomADC(&adc_scale_custom);
161 temp_hst_e.setCustomADC(&adc_scale_custom);
162 temp_hst_w.setCustomADC(&adc_scale_custom);
163 temp_btm_1.setCustomADC(&adc_scale_custom);
164 temp_btm_2.setCustomADC(&adc_scale_custom);
165 temp_btm_3.setCustomADC(&adc_scale_custom);
166 temp_btm_4.setCustomADC(&adc_scale_custom);
167 temp_top_1.setCustomADC(&adc_scale_custom);
168 temp_top_2.setCustomADC(&adc_scale_custom);
169 temp_top_3.setCustomADC(&adc_scale_custom);
170 temp_top_4.setCustomADC(&adc_scale_custom);
171 temp_ext_b.setCustomADC(&adc_scale_custom);
172 temp_ext_l.setCustomADC(&adc_scale_custom);
173 temp_ext_r.setCustomADC(&adc_scale_custom);
174 temp_ext_f.setCustomADC(&adc_scale_custom);
175
176 // Vincula as escalas calibradas de corrente.
177 current_btm.setConversion(current_btm_scale_value, current_btm_scale_volt, current_scale_count);
178 current_lft.setConversion(current_lft_scale_value, current_lft_scale_volt, current_scale_count);
179
180 // Vincula as escalas calibradas de temperatura.
181 temp_hst_n.setConversion(temp_hst_scale_value, temp_hst_scale_volt, temp_scale_count);
182 temp_hst_s.setConversion(temp_hst_scale_value, temp_hst_scale_volt, temp_scale_count);
183 temp_hst_e.setConversion(temp_hst_scale_value, temp_hst_scale_volt, temp_scale_count);
184 temp_hst_w.setConversion(temp_hst_scale_value, temp_hst_scale_volt, temp_scale_count);
185 temp_btm_1.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
186 temp_btm_2.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
187 temp_btm_3.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
188 temp_btm_4.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
189 temp_top_1.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
190 temp_top_2.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
191 temp_top_3.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
192 temp_top_4.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
193 temp_ext_b.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
194 temp_ext_l.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
195 temp_ext_r.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
196 temp_ext_f.setConversion(temp_ref_scale_value, temp_ref_scale_volt, temp_scale_count);
197
198 // Ajuste fino individual de cada objeto de medição.
199 current_btm.calibrate(0.0, -0.05, 6.6, 6.5);
```

```

200 current_lft.calibrate(0.0, 0.18, 6.2, 6.9);
201 temp_hst_n.calibrate(18.0, 18.055, 28.0, 27.811);
202 temp_hst_s.calibrate(18.0, 18.013, 28.0, 27.772);
203 temp_hst_e.calibrate(18.0, 18.021, 28.0, 27.768);
204 temp_hst_w.calibrate(18.0, 18.121, 28.0, 27.849);
205 temp_btm_1.calibrate(25.0, 24.905, 45.0, 44.756);
206 temp_btm_2.calibrate(25.0, 24.777, 45.0, 44.341);
207 temp_btm_3.calibrate(25.0, 24.951, 45.0, 44.699);
208 temp_btm_4.calibrate(25.0, 24.798, 45.0, 44.560);
209 temp_top_1.calibrate(25.0, 24.780, 45.0, 44.571);
210 temp_top_2.calibrate(25.0, 24.953, 45.0, 44.598);
211 temp_top_3.calibrate(25.0, 24.930, 45.0, 44.762);
212 temp_top_4.calibrate(25.0, 24.868, 45.0, 44.669);
213 temp_ext_b.calibrate(25.0, 24.900, 45.0, 44.751);
214 temp_ext_l.calibrate(25.0, 24.843, 45.0, 44.633);
215 temp_ext_r.calibrate(25.0, 24.927, 45.0, 44.746);
216 temp_ext_f.calibrate(25.0, 24.948, 45.0, 44.691);
217
218 // Configura os pinos dos relés e os desliga inicialmente.
219 pinMode(PIN_RLY_BTM, OUTPUT);
220 pinMode(PIN_RLY_LFT, OUTPUT);
221 digitalWrite(PIN_RLY_BTM, LOW);
222 digitalWrite(PIN_RLY_LFT, LOW);
223
224 // Cria a tarefa de medição.
225 xTaskCreate(taskMeasurement, "TempsMonitoring", 8192, NULL, 1, &handler_task);
226 }
227
228 // -----
229 // Tarefa principal, apenas verifica se existe dados na serial.
230 void loop() {
231     vTaskDelay(pdMS_TO_TICKS(500));
232 }
233
234 // Recebe comandos via serial.
235 void serialEvent() {
236     if (Serial.available()) {
237         String cmd = Serial.readStringUntil('\n');
238         commandHandler(cmd);
239     }
240 }
241
242 // Utils -----
243 // Pega o tempo desdo boot do microcontrolador.
244 int64_t getStartTime() {
245     return esp_timer_get_time() / 1000;
246 }
247
248 // Converte string para int64
249 int64_t stringToInt64(String str) {
250     const char *cStr = str.c_str();
251
252     return strtoll(cStr, NULL, 10);
253 }
254
255 // Trata diferentes comandos recebidos via serial.
256 void commandHandler(String cmd) {
257     if (cmd.startsWith("start")) {
258         timestamp = stringToInt64(cmd.substring(6)) - getStartTime();
259         startProcess();
260
261     } else if (cmd.startsWith("stop")) {
262         stopProcess();
263
264     } else if (cmd.startsWith("sync")) {
265         String sync_msg = String(process_status) + ", " + file_name + ", "
266             + String(resistance_btm) + ", " + String(resistance_lft) + ", "

```

```

267         + String(measure_Ts) + ", " + String(resistance_Tu);
268
269     Serial.println("sync:" + sync_msg);
270
271     } else if (cmd.startsWith("file")) {
272         file_name = cmd.substring(5);
273
274     } else if (cmd.startsWith("rBTM")) {
275         resistance_btm = cmd.substring(5).equals("1");
276
277     } else if (cmd.startsWith("rLFT")) {
278         resistance_lft = cmd.substring(5).equals("1");
279
280     } else if (cmd.startsWith("mTsp")) {
281         measure_Ts = cmd.substring(5).toInt();
282
283     } else if (cmd.startsWith("rTup")) {
284         resistance_Tu = cmd.substring(5).toInt();
285     }
286 }
287
288 // Inicia o experimento e cria um novo arquivo de medições.
289 void startProcess() {
290     if (process_status == PROCESS_STOP) {
291         storage_card.deleteFile(file_name);
292         storage_card.appendFile(file_name, "time, current_btm, current_lft, temp_hst_n, temp_hst_s,
293                                         temp_hst_e, temp_hst_w, temp_btm_1, temp_btm_2, temp_btm_3,
294                                         temp_btm_4,temp_top_1, temp_top_2, temp_top_3, temp_top_4,
295                                         temp_ext_b, temp_ext_l, temp_ext_r, temp_ext_f");
296
297         process_status = PROCESS_START;
298     } else {
299         Serial.println("erro:Process already running");
300     }
301 }
302
303 // Finaliza o experimento e garante que os relés estão desligados.
304 void stopProcess() {
305     process_status = PROCESS_STOP;
306
307     digitalWrite(PIN_RLY_BTM, LOW);
308     digitalWrite(PIN_RLY_LFT, LOW);
309
310     if (handler_timer != NULL) {
311         if (xTimerIsTimerActive(handler_timer) != pdFALSE) {
312             xTimerDelete(handler_timer, 0);
313         }
314     }
315 }
316
317 // Tasks
318 // Callback do timer para desligar os relés após o tempo configurado.
319 void callBackTurnOffResistances(TimerHandle_t xTimer) {
320     digitalWrite(PIN_RLY_BTM, LOW);
321     digitalWrite(PIN_RLY_LFT, LOW);
322     Serial.println("warn:Resistance(s) turned off");
323 }
324
325 // Tarefa para realizar as medições do experimento e armazenar os dados.
326 void taskMeasurement(void *params) {
327     TickType_t elapsed_time = xTaskGetTickCount();
328
329     for (;;) {
330         // Se o experimento está prestes a começar,
331         // cria o timer das resistências e muda o
332         // estado do processo para PROCESS_RUN.
333         if (process_status == PROCESS_START) {

```

```

334     handler_timer = xTimerCreate("TurnOffResistances", pdMS_TO_TICKS(resistance_Tu),
335                                 pdFALSE, (void *)0, callBackTurnOffResistances);
336
337     // Aciona a respectiva resistência de aquecimento.
338     digitalWrite(PIN_RLY_BT, resistance_btm);
339     digitalWrite(PIN_RLY_LFT, resistance_lft);
340     process_status = PROCESS_RUN;
341     Serial.println("warn:Resistance(s) turned on");
342
343     // Inicializa o timer.
344     if (xTimerStart(handler_timer, 0) != pdPASS) {
345         digitalWrite(PIN_RLY_BT, LOW);
346         digitalWrite(PIN_RLY_LFT, LOW);
347
348         Serial.println("err:Failed to start timer");
349         process_status = PROCESS_STOP;
350     }
351 }
352
353 // Realiza as medições.
354 float current_btm_value = current_btm.measure();
355 float current_lft_value = current_lft.measure();
356 float temp_hst_n_value = temp_hst_n.measure();
357 float temp_hst_s_value = temp_hst_s.measure();
358 float temp_hst_e_value = temp_hst_e.measure();
359 float temp_hst_w_value = temp_hst_w.measure();
360 float temp_btm_1_value = temp_btm_1.measure();
361 float temp_btm_2_value = temp_btm_2.measure();
362 float temp_btm_3_value = temp_btm_3.measure();
363 float temp_btm_4_value = temp_btm_4.measure();
364 float temp_top_1_value = temp_top_1.measure();
365 float temp_top_2_value = temp_top_2.measure();
366 float temp_top_3_value = temp_top_3.measure();
367 float temp_top_4_value = temp_top_4.measure();
368 float temp_ext_b_value = temp_ext_b.measure();
369 float temp_ext_l_value = temp_ext_l.measure();
370 float temp_ext_r_value = temp_ext_r.measure();
371 float temp_ext_f_value = temp_ext_f.measure();
372
373 // Cria uma "linha" medição em formato CSV.
374 String measuring_row = String(timestamp + getStartTime()) + ", "
375     + String(current_btm_value) + ", " + String(current_lft_value) + ", "
376     + String(temp_hst_n_value) + ", " + String(temp_hst_s_value) + ", "
377     + String(temp_hst_e_value) + ", " + String(temp_hst_w_value) + ", "
378     + String(temp_btm_1_value) + ", " + String(temp_btm_2_value) + ", "
379     + String(temp_btm_3_value) + ", " + String(temp_btm_4_value) + ", "
380     + String(temp_top_1_value) + ", " + String(temp_top_2_value) + ", "
381     + String(temp_top_3_value) + ", " + String(temp_top_4_value) + ", "
382     + String(temp_ext_b_value) + ", " + String(temp_ext_l_value) + ", "
383     + String(temp_ext_r_value) + ", " + String(temp_ext_f_value);
384
385 // Envia as medições via serial para a interface gráfica do usuário.
386 Serial.println(String("data:" + measuring_row));
387
388 // Armazena as medições.
389 if (process_status == PROCESS_RUN) {
390     storage_card.appendFile(file_name, measuring_row);
391 }
392
393 // Aguarda até o próximo tempo de amostragem.
394 vTaskDelayUntil(&elapsed_time, pdMS_TO_TICKS(measure_Ts));
395 }
396 }
397

```

APÊNDICE E – ALGORITMO DE SIMULAÇÃO DO FLUXO DE CALOR

solver\lib\heat_model.py

```

1 import numpy as np
2 import math
3 import copy
4 import heat_solver_MMQ
5 import heat_solver_LS
6
7
8 # -----
9 # Executa a simulação.
10 def run(params, settings, solver="LS"):
11     # Cria cópias para não alterar o objeto original.
12     params = copy.deepcopy(params)
13     settings = copy.deepcopy(settings)
14
15     # Converte as temperaturas para kelvin.
16     settings["T0"] = temperature_c2k(settings["T0"])
17
18     # Cria o modelo para simulação.
19     sln = _sln_setup(params, settings)
20
21     # Inicializa o vetor do tempo.
22     time = np.array([sln["t"]])
23
24     # Inicializa o vetor de monitoramento.
25     signals = np.array([sln["T"][sln["idxRecord"]]])
26
27     # Inicializa o vetor de gradiente da sonda.
28     gradient = np.array([_calc_probe_gradient(sln, settings)])
29
30     # Inicializa o vetor de energia do sistema.
31     energy = np.array([_calc_energy(sln)])
32
33     # Guarda a energia inicial do sistema.
34     energy_initial = energy[0]
35
36     # Seleciona o solver para o passo de simulação.
37     if solver == "LS":
38         evolve = heat_solver_LS.evolve
39     else:
40         evolve = heat_solver_MMQ.evolve
41
42     # Inicia a simulação.
43     while sln["t"] < sln["tf"]:
44         evolve(sln)
45
46         time = np.append(time, sln["t"])
47         energy = np.append(energy, _calc_energy(sln))
48         gradient = np.append(gradient, _calc_probe_gradient(sln, settings))
49
50         signals = np.vstack((signals, sln["T"][sln["idxRecord"]]))
51
52     # Remove a energia inicial.
53     energy -= energy_initial
54
55     # Retorna as temperaturas para celcius.
56     signals_C = temperature_k2c(signals)
57     T_C = temperature_k2c(sln["T"])
58
59     return time, signals_C, gradient, sln["x"], T_C, energy
60
61 # Cria o modelo para simulação.
62 def _sln_setup(params, settings):
63     sln = {
64         "t": 0,
65         # Tempo inicial da simulação [s].

```

```

66     "k": params["k"],      # Condutividade térmica do solo [W/mK].
67     "Cv": params["Cv"],   # Capacidade térmica volumétrica do solo [J/m3K].
68     "DX": params["DX"],   # Tamanho do domínio [m].
69     "dt": settings["dt"], # Passo de simulação [s].
70     "tf": settings["tf"], # Tempo final da simulação [s].
71
72     # Índices dos volumes para monitorar
73     "idxRecord": np.array([], dtype=np.int8),
74     "qLParams": {"Q0": params["QSRC"],      # Potência da fonte [W/m2]
75                  "duration": params["TSRC"], # Duração da fonte [s]
76                  },
77     "qRParams": {"Q0": 0, # Sem perdas de energia.
78                  "duration": 0,
79                  },
80 }
81
82 sln["alpha"] = sln["k"] / sln["Cv"]
83 sln["dim"] = math.ceil(params["DX"] / settings["dx"])
84 sln["dx"] = sln["DX"] / sln["dim"]
85 sln["lamb"] = 0.5 * sln["alpha"] * sln["dt"] / (sln["dx"] ** 2)
86 sln["x"] = np.array([(0.5 + i) * sln["dx"] for i in range(sln["dim"])])
87
88 # Determina os índices dos volumes que devem ser monitorados.
89 for i in settings["xRecord"]:
90     for j in range(sln["dim"] - 1):
91         if (i > j * sln["dx"] and i ≤ (j+1) * sln["dx"]):
92             sln["idxRecord"] = np.append(sln["idxRecord"], j)
93
94 sln["T"] = _initial_condition(settings["T0"], sln["idxRecord"], sln["x"])
95
96 return sln
97
98
99 # Calcula a energia atual dentro do sistema.
100 def _calc_energy(sln):
101     #  $K * m * (J/m^3 K) * m^2 \Rightarrow J$ 
102     return np.sum(sln["T"]) * sln["dx"] * sln["Cv"] * sln["DX"]**2
103
104
105 # Calcula o gradiente no centro da sonda.
106 def _calc_probe_gradient(sln, settings):
107     dx = settings["dx"]
108     dT = sln["T"][15] - sln["T"][14]
109
110     return dT / dx
111
112
113 # Calcula a temperatura inicial para cada elemento de volume.
114 def _initial_condition(T0, idxRecord, x):
115     initial_T = np.zeros(len(x))
116
117     initial_T[0:idxRecord[0]] = T0[0]
118     initial_T[idxRecord[-1]:] = T0[-1]
119
120     for i in range(len(idxRecord) - 1):
121         initial_T_i = np.linspace(T0[i], T0[i+1], idxRecord[i+1]-idxRecord[i]+1)
122         initial_T[idxRecord[i]:idxRecord[i+1] + 1] = initial_T_i
123
124     return initial_T
125
126 # Converte de celsius para kelvin.
127 def temperature_c2k(T):
128     return T + 273.15
129
130 # Converte de kelvin para celsius.
131 def temperature_k2c(T):
132     return T - 273.15

```


solver\lib\heat_solver_MMQ.py

```

1 import numpy as np
2 from scipy.optimize import least_squares
3
4
5 # -----
6 def evolve(sln):
7     result = least_squares(_residual, sln["T"], jac="2-point", kwargs=sln)
8
9     if (result.success):
10         sln["T"] = result.x
11         sln["t"] += sln["dt"]
12         return True
13
14     return False
15
16
17 def _residual(Tnew, **sln):
18     T = np.array(sln["T"])
19     Tnew = np.array(Tnew)
20
21     lamb = sln["lamb"]
22     dx_k = sln["dx"] / sln["k"]
23
24     boundary_flux_L0 = lamb * _boundary_flux(sln["t"], sln["qLParams"]) * dx_k
25     boundary_flux_L1 = lamb * _boundary_flux(sln["t"] + sln["dt"], sln["qLParams"]) * dx_k
26     boundary_flux_R0 = lamb * _boundary_flux(sln["t"], sln["qRParams"]) * dx_k
27     boundary_flux_R1 = lamb * _boundary_flux(sln["t"] + sln["dt"], sln["qRParams"]) * dx_k
28
29     F0 = np.zeros(len(Tnew))
30     F1 = np.zeros(len(Tnew))
31
32     F0[0] = lamb * (T[1] - T[0]) + boundary_flux_L0
33     F1[0] = lamb * (Tnew[1] - Tnew[0]) + boundary_flux_L1
34
35     F0[1:-1] = lamb * (T[2:] - 2 * T[1:-1] + T[:-2])
36     F1[1:-1] = lamb * (Tnew[2:] - 2 * Tnew[1:-1] + Tnew[:-2])
37
38     F0[-1] = lamb * (T[-2] - T[-1]) - boundary_flux_R0
39     F1[-1] = lamb * (Tnew[-2] - Tnew[-1]) - boundary_flux_R1
40
41     return T - Tnew + F0 + F1
42
43
44 def _boundary_flux(t, qParams):
45     return qParams["Q0"] if t < qParams["duration"] else 0
46

```

solver\lib\heat_solver_LS.py

```

1 import numpy as np
2
3
4 # -----
5 def evolve(sln):
6     # →  $A^- * T_{new} = A^+ * T_{old} + C$ 
7     # →  $A * x = B$ 
8
9     #  $A = A^-$ 
10    #  $x = T_{new}$ 
11    #  $B = A^+ * T_{old} + C$ 
12
13    # -----
14    T = sln["T"].reshape(-1, 1)
15
16    A0, A1 = _calc_matrix_A(sln)
17    C = _calc_vector_C(sln)
18
19    A = A0
20    B = np.dot(A1, T) + C
21
22    # Solução do sistema linear  $Ax=B$ 
23    x = np.linalg.solve(A, B)
24
25    sln["T"] = x.ravel()
26    sln["t"] += sln["dt"]
27
28    return True
29
30
31 def _calc_matrix_A(sln):
32     lamb = sln["lamb"]
33     dt = sln["dt"]
34
35     B = _calc_matrix_B(sln)
36
37     var = np.dot(lamb * dt / 2, B)
38     I = np.eye(B.shape[0])
39
40     #  $A_{\pm} = I \pm (\lambda * \delta t / 2) * B$ 
41     A0 = I - var #  $A^-$ 
42     A1 = I + var #  $A^+$ 
43
44     return A0, A1
45
46
47 def _calc_matrix_B(sln):
48     n = len(sln["T"])
49
50     b0 = -1
51     bn = -1
52
53     B = np.zeros((n, n))
54
55     B[np.arange(0, n-1), np.arange(1, n)] = 1.0
56     B[np.arange(0, n), np.arange(0, n)] = np.array([b0] + [-2] * (n - 2) + [bn])
57     B[np.arange(1, n), np.arange(0, n-1)] = 1.0
58
59     return B
60
61
62 def _calc_vector_C(sln):
63     n = len(sln["T"])
64
65     lamb = sln["lamb"]

```

```
66     dt = sln["dt"]
67     dx = sln["dx"]
68     k = sln["k"]
69
70     boundary_flux_L0 = -_boundary_flux(sln["t"], sln["qLParams"]) / k
71     boundary_flux_L1 = -_boundary_flux(sln["t"] + sln["dt"], sln["qLParams"]) / k
72     boundary_flux_R0 = -_boundary_flux(sln["t"], sln["qRParams"]) / k
73     boundary_flux_R1 = -_boundary_flux(sln["t"] + sln["dt"], sln["qRParams"]) / k
74
75     var = (dt * dx / 2) * lamb
76
77     c0 = -var * (boundary_flux_L0 + boundary_flux_L1)
78     cn = var * (boundary_flux_R0 + boundary_flux_R1)
79
80     return np.vstack((c0, np.zeros((n - 2, 1)), cn))
81
82
83 def _boundary_flux(t, qParams):
84     return qParams["Q0"] if t < qParams["duration"] else 0
85
```

APÊNDICE F – ALGORITMO DE OTIMIZAÇÃO

solver\soil_optimizer.py

```

1  import sys
2  import numpy as np
3  from scipy.optimize import least_squares
4
5  sys.path.append("./")
6  sys.path.append("./lib")
7
8  from lib import utils
9
10 # -----
11 # Dados experimentais.
12 FILE_DATA = "./data/data_mmg.csv"
13
14 # Chute inicial.
15 INITIAL_GUESS = [
16     1.0, # k = Condutividade térmica [W/(mK)].
17     2.0e6, # Cv = Capacidade térmica volumétrica [J/(m3K)].
18 ]
19
20 # Restrições nos parâmetros.
21 CONSTRAINTS = ([0.25, 1e6], [3, 3e6])
22
23 # Parâmetros do sistema.
24 MODEL_PARAMS = {
25     "DX": 0.3, # Comprimento do domínio [m].
26     "QSRC": 816.1, # Fluxo de potência médio da fonte [W/m2] → 6.5*11.3/0.09.
27     "TSRC": 20*60, # Duração da fonte [s].
28 }
29
30 # Configurações da simulação.
31 SETTINGS = {
32     "dx": 0.01, # Comprimento do elemento volumétrico [m].
33     "dt": 1, # Passo de simulação [s].
34     "tf": 14*60*60, # Tempo de simulação [s].
35
36     # Salvar o histórico do perfil de temperatura nas posições dos sensores [m].
37     "xRecord": np.array([signal[0] for signal in utils.SIGNALS]),
38 }
39
40 # -----
41 def main():
42     exp = utils.load_data(FILE_DATA)
43
44     # Determina as temperaturas iniciais (T0) para a simulação das curvas.
45     SETTINGS["T0"] = utils.get_T0(exp["T"])
46
47     result = least_squares(residual, INITIAL_GUESS, args=(exp,), bounds=CONSTRAINTS)
48     minimizer = result.x
49     print(minimizer)
50
51 # Função residual de otimização.
52 def residual(values, exp):
53     PARAMS = MODEL_PARAMS.copy()
54     PARAMS["k"], PARAMS["Cv"] = values
55
56     # Executa a simulação e obtém as curvas simulada.
57     sim = utils.run_simulation(PARAMS, SETTINGS, "MMQ")
58
59     # Calcula a diferença entre as respectivas curvas experimentais e simuladas.
60     erro = exp["T"] - sim["T"]
61
62     return erro
63
64 if __name__ == "__main__":
65     main()

```

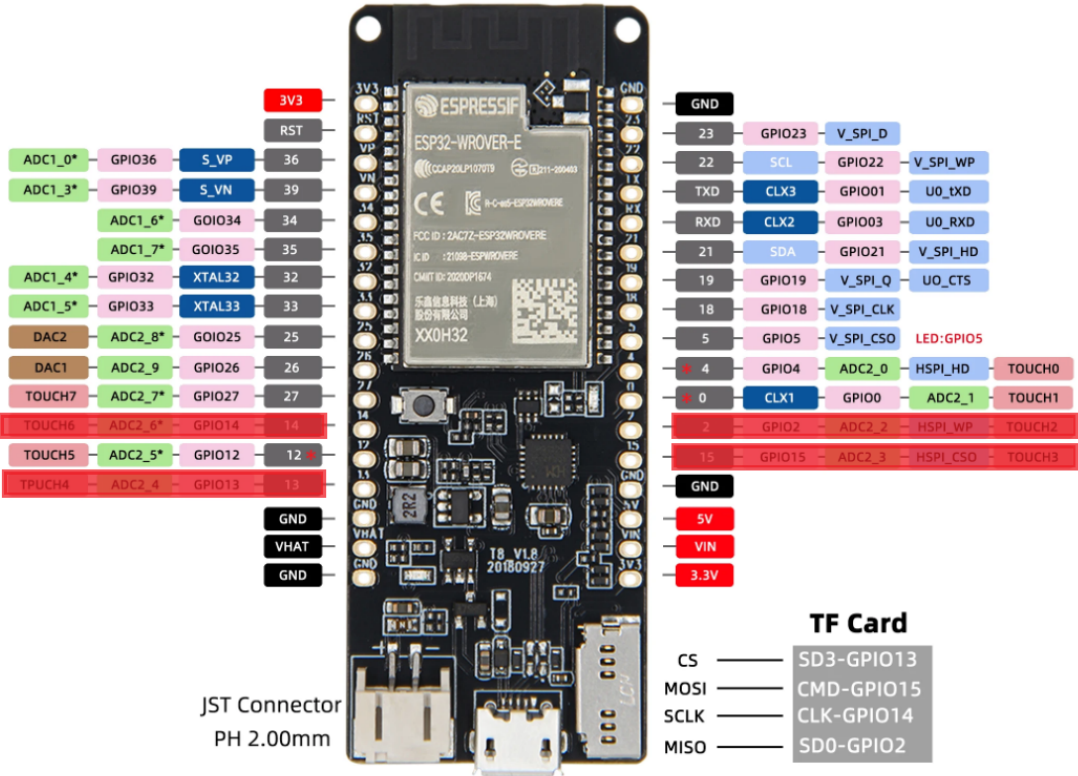
ANEXO A – AT THERMISTOR - TEMPERATURA X RESISTÊNCIA

Resistance - Temperature																		
Temperature		Type							Temperature		Type							
(°C)		102AT	202AT	502AT	103AT	203AT	503AT	104AT	(°C)		102AT	202AT	502AT	103AT	203AT	503AT	104AT	
-50	24.46	55.66	154.6	329.5	1253	3168	11473		35	0.7229	1.424	3.508	6.940	13.06	32.48	60.94		
-45	18.68	42.17	116.5	247.7	890.5	2257	7781		40	0.6189	1.211	2.961	5.827	10.65	26.43	48.10		
-40	14.43	32.34	88.91	188.5	642.0	1632	5366		45	0.5316	1.033	2.509	4.911	8.716	21.59	38.13		
-35	11.23	24.96	68.19	144.1	465.8	1186	3728		50	0.4587	0.8854	2.137	4.160	7.181	17.75	30.44		
-30	8.834	19.48	52.87	111.3	342.5	872.8	2629		55	0.3967	0.7620	1.826	3.536	5.941	14.64	24.42		
-25	6.998	15.29	41.21	86.43	253.6	646.3	1864		60	0.3446	0.6587	1.567	3.020	4.943	12.15	19.72		
-20	5.594	12.11	32.44	67.77	190.0	484.3	1340		65	0.3000	0.5713	1.350	2.588	4.127	10.13	15.99		
-15	4.501	9.655	25.66	53.41	143.2	364.6	969.0		70	0.2622	0.4975	1.168	2.228	3.464	8.482	13.05		
-10	3.651	7.763	20.48	42.47	109.1	277.5	709.5		75	0.2285	0.4343	1.014	1.924	2.916	7.129	10.68		
-5	2.979	6.277	16.43	33.90	83.75	212.3	523.3		80	0.1999	0.3807	0.8835	1.668	2.468	6.022	8.796		
0	2.449	5.114	13.29	27.28	64.88	164.0	390.3		85	0.1751	0.3346	0.7722	1.451	2.096	5.105	7.271		
5	2.024	4.188	10.80	22.05	50.53	127.5	292.5		90	0.1536	0.2949	0.6771	1.266	1.788	4.345	6.041		
10	1.684	3.454	8.840	17.96	39.71	99.99	221.5		95			0.5961	1.108	1.530	3.712	5.037		
15	1.408	2.862	7.267	14.69	31.36	78.77	168.6		100			0.5265	0.9731	1.315	3.185	4.220		
20	1.184	2.387	6.013	12.09	24.96	62.56	129.5		105			0.4654	0.8572	1.134	2.741	3.546		
25	1.000	2.000	5.000	10.00	20.00	50.00	100.0		110			0.4128	0.7576	0.9807	2.369	2.994		
30	0.8486	1.684	4.179	8.313	16.12	40.20	77.81											
Unit(kΩ)																		

Unit(kΩ)

Fonte: Mouser Electronics (2024).

ANEXO B – ESP32 LILYGO T8 V1.8 PINMAP



LILYGO T8 V1.8 PinMAP

Fonte: lilygo (2024).