

UNIVERSIDADE FEDERAL DE ALFENAS

**GABRIEL MAGALHÃES BORGES
PEDRO PAULO CANESCHI MASILI SANTOS
RAFAEL MANSI GUARINELO**

**DESENVOLVIMENTO DE UM CONTROLADOR PID PARA TEMPERATURA EM
RECIPIENTES VOLTADOS A EXPERIMENTOS DIDÁTICOS**

**POÇOS DE CALDAS/MG
2023**

**GABRIEL MAGALHÃES BORGES
PEDRO PAULO CANESCHI MASILI SANTOS
RAFAEL MANSI GUARINELO**

**DESENVOLVIMENTO DE UM CONTROLADOR PID PARA TEMPERATURA EM
RECIPIENTES VOLTADOS A EXPERIMENTOS DIDÁTICOS**

Produto de Conclusão PIEPEX apresentada como parte dos requisitos para obtenção do título de Bacharel em Interdisciplinar em Ciência e tecnologia pela Universidade Federal de Alfenas.

Orientador: Prof. Dr. Laos Alexandre Hirano
Coorientador: Prof. Dr. Marlus Pinheiro
Rolemberg

**POÇOS DE CALDAS/MG
2023**

Sistema de Bibliotecas da Universidade Federal de Alfenas
Biblioteca Campus Poços de Caldas

Magalhães, Gabriel Borges.

DESENVOLVIMENTO DE UM CONTROLADOR PID PARA
TEMPERATURA EM RECIPIENTES VOLTADOS A EXPERIMENTOS
DIDÁTICOS / Gabriel Borges Magalhães, Pedro Paulo Caneschi Masili
Santos, Rafael Mansi Guarinelo. - Poços de Caldas, MG, 2023.

36 f. : il. -

Orientador(a): Laos Alexandre Hirano.

Trabalho de Conclusão de Curso (Graduação em Bacharelado
Interdisciplinar em Ciência e Tecnologia) - Universidade Federal de
Alfenas, Poços de Caldas, MG, 2023.

Bibliografia.

1. Controlador de temperatura PID. 2. Método de Ziegler-Nichols. 3.
Controlador de temperatura on-off zona morta. 4. Calibração de sensor de
temperatura. 5. Desenvolvimento de um experimento didático. I. Santos,
Pedro Paulo Caneschi Masili. II. Guarinelo, Rafael Mansi. III. Alexandre
Hirano, Laos, orient. IV. Título.

Ficha gerada automaticamente com dados fornecidos pelo autor.

**GABRIEL MAGALHÃES BORGES
PEDRO PAULO CANESCHI MASILI SANTOS
RAFAEL MANSI GUARINELO**

**DESENVOLVIMENTO DE UM CONTROLADOR PID PARA TEMPERATURA EM
RECIPIENTES VOLTADOS A EXPERIMENTOS DIDÁTICOS**

O(A) Presidente da banca examinadora abaixo
assina a aprovação da apresentada como parte dos
requisitos para obtenção do título de Bacharel em
Interdisciplinar em Ciência e tecnologia pela
Universidade Federal de Alfenas.

Aprovada em: 22 de julho de 2023

Prof. Dr. Flávio Augusto Dias Oliveira
Universidade Federal de Alfenas

Assinatura:

Prof^a. Dr^a. Jaqueline Costa Martins
Universidade Federal de Alfenas

Assinatura:

Prof. Dr. Marcos Vinicius Rodrigues
Universidade Federal de Alfenas

Assinatura:

AGRADECIMENTOS

Gostaríamos de agradecer inicialmente aos nossos pais, que sempre nos deram suporte e condições para chegarmos até aqui.

Agradecemos aos nossos orientadores, Doutor Laos e Doutor Marlus, pela paciência, oportunidade e assistência na realização deste trabalho.

Agradecemos também aos técnicos de laboratório Logan e Ícaro que sempre estiveram à disposição.

Agradecemos a LAPSE – Laboratório de Prototipagem de Sistemas Embarcados que proporcionou oportunidades exclusivas.

Agradecemos aos nossos amigos, que estiveram por perto dando forças.

RESUMO

Os controladores PID possuem grande aplicação na engenharia e, por essa razão, são amplamente estudados em aulas práticas e experimentais. Em um sistema de controle de processos, a intenção é buscar e ajustar a variável de controle com base em um sinal de erro. A partir de uma demanda para aprimorar os experimentos com controladores PID nas aulas práticas dos cursos de engenharia da Universidade Federal de Alfenas, este trabalho teve como objetivo o desenvolvimento de um controlador PID de temperatura de baixo custo, programável e com experimentos de curta duração, voltado para o estudo destes sistemas. O aparato foi programado em uma placa da Arduino® para enviar as medidas de temperatura diretamente para o Excel® de forma simultânea à medição, onde as mesmas foram realizadas por termopares, amplificadas e convertidas por um módulo Max6675. A fonte de calor, resistência de 350 Watts, foi controlada a partir do Arduino® por intermédio de um relé de estado sólido de 2 Amperes. O modo de controle PID foi sintonizado através do primeiro método de Ziegler-Nichols, que teve como base uma planta de um controlador *on-off* zona morta. A faixa de controle em 40°C para o controlador PI e PID, respectivamente, teve uma diminuição de $1.17 \pm 0.11^\circ\text{C}$ e $1.67 \pm 0.11^\circ\text{C}$ em comparação ao *on-off* zona morta. Diante do exposto, foi possível concluir o desenvolvimento do aparato científico, no qual há possibilidades de incluir diferentes programações para meios de controle da temperatura, adaptação e incrementação com mais dispositivos e interface com planilha Excel. Destacando o custo de R\$127,00 e um tempo médio de duração experimental de trinta minutos, de modo a atrelar perfeitamente com a proposta inicial do trabalho.

Palavras-chave: Controlador de temperatura PID. Método Ziegler-Nichols. Controlador de temperatura *on-off* zona morta. Calibração sensor de temperatura.

ABSTRACT

PID controllers have a wide range of applications in engineering and are extensively studied in practical and experimental classes. In a process control system, the intention is to seek and adjust the control variable based on an error signal. With the aim of improving the experiments with PID controllers in practical classes of engineering courses at the Federal University of Alfenas, this work aimed to develop a low-cost, programmable PID temperature controller for short-duration experiments to acquire and analyze temperature data over time. The apparatus was programmed using an Arduino® to directly send temperature measurements to Excel simultaneously with the measurement. The measurements were performed using thermocouples, amplified, and converted by a Max6675 module. The 350-watt heat source was controlled by the Arduino® through a 2-ampere solid-state relay. The PID control method used for tuning was the Ziegler-Nichols first method, based on a plant with a dead-zone on-off controller. Based on the findings of this work, it was possible to conclude the development of a scientific apparatus that allows for different temperature control programming, adaptation, and expansion with additional devices and interface with Excel spreadsheets. It is worth highlighting its low cost of R\$127.00 and an average experimental duration of thirty minutes, perfectly aligned with the initial proposal of the work.

Keywords: PID temperature controller, Ziegler-Nichols method, on-off temperature controller with dead-zone, temperature sensor calibration.

LISTA DE FIGURAS

Figura 1 - Arquitetura básica de um controlador de temperatura.....	1
Figura 2 - Esquema genérico de controlador em malha fechada.....	3
Figura 3 - Modelo de resposta de um sistema on-off	4
Figura 4 - Curva característica de um controlador on-off com zona morta.	5
Figura 5 - Curva característica de um sistema PID.....	6
Figura 6 - Esquema genérico de um controlador PID.	7
Figura 7 - Representação visual das variáveis T e L em curva de resposta em forma de S.	9
Figura 8 - Arduino® UNO.....	10
Figura 9 - Esquemática da montagem do circuito para calibração.	12
Figura 10 - Esquemática da montagem do circuito, controlador on-off zona morta.....	15
Figura 11 - Gráfico da curva de calibração do aparo, em conjunto a equação da reta da tendência linear e o valor do χ^2 reduzido.....	18
Figura 12 - Aquecimento em triplicata de temperatura em função do tempo, do controlador on-off com zona morta de 0.5°C para as temperaturas de setpoint.	19
Figura 13 - Aquecimento em triplicata de temperatura em função do tempo, do controlador PI para as temperaturas de setpoint.....	21
Figura 14 - Aquecimento em triplicata de temperatura em função do tempo, do controlador PI para as temperaturas de setpoint.....	22
Figura 15 - Dados do aquecimento dos controles on-off zona morta, PI e PID em 40°C e 50°C de setpoint.	24
Figura 16 - Gráfico da perturbação nos sistemas de controle PI e PID nas temperaturas de setpoint de 40°C, 60°C e 80°C.	26
Figura 17 - Curso da LAPSE de introdução a sistemas embarcados (2023).....	28
Figura 18 - Estudantes e organizadores do curso de sistemas embarcados da LAPSE (2023).....	29
Figura 19 - Protótipo do controlador de temperatura	30

LISTA DE TABELAS

Tabela 1 - Regra de sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta.	9
Tabela 2 - Materiais e custos.	11
Tabela 3 - Sistematização das ligações entre a conexão Max6675 - Arduino®.	12
Tabela 4 - Sistematização das ligações entre a conexão módulo IC2 - Arduino®.	12
Tabela 5 - Análise da viabilidade dos sensores de temperatura.	17
Tabela 6 - Valores médios de temperatura máxima e mínima, faixa de controle e tempo médio de chaveamento do controlador <i>on-off</i>	19
Tabela 7 - Valores de T, L, Kp, Ki e Kd para cada temperatura de <i>setpoint</i>	20
Tabela 8 - Valores médios do controle PI para as temperaturas máximas e mínimas, faixa de temperatura do controle e temperatura máxima do <i>overshoot</i>	21
Tabela 9 - Valores médios do controle PID para as temperaturas máximas e mínimas, faixa de controle e <i>overshoot</i>	23
Tabela 10 - Valores de tempo de ajuste e temperatura mínima durante o ensaio de perturbação.	26

SUMÁRIO

1. INTRODUÇÃO	1
2. OBJETIVOS	2
2.1. Objetivo principal	2
2.2. Objetivos específicos	2
3. REVISÃO BIBLIOGRÁFICA	2
3.1. Controladores	2
3.2. Controlador <i>ON-OFF</i>	3
3.3. Controlador PID	5
3.3.1. Regra de sintonia de Ziegler-Nichols para controladores PID	8
3.3.2. Obtenção das Variáveis T e L para o primeiro método	8
3.3.3. Regra e sintonia de Ziegler-Nichols primeiro método	9
3.4. Microcontroladores	9
3.5. Arduíno	10
4. METODOLOGIA	11
4.1. Materiais	11
4.2. Escolha do sensor de temperatura	11
4.3. Calibração do termopar tipo-K	11
4.3.1. Ligações e montagem do circuito	11
4.3.2. Procedimento experimental	13
4.4. Controlador <i>on-off</i> zona morta	14
4.4.1. Ligações e montagem do circuito	14
4.4.2. Procedimento experimental	15
4.5. Controlador PI e PID	16
4.5.1. Ligações e montagem do circuito	16
4.5.2. Procedimento Experimental	16
4.6. Ensaio de perturbação	17
5. RESULTADOS E DISCUSSÃO	17
5.1. Escolha do sensor	17
5.2. Calibração do Termopar Tipo K	18
5.3. Controlador <i>on-off</i> com zona morta	18
5.4. Cálculo da sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta	20
5.5. Controlador PI	20
5.6. Controlador PID	22
5.7. Controlador <i>on-off</i> zona morta, em comparação ao controlador PI e PID	23
5.8. Comparação entre controlador PI e PID	25

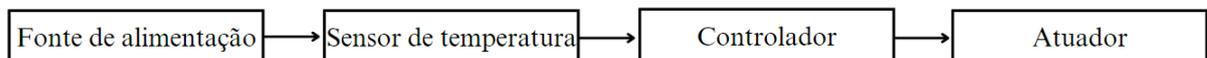
5.9. Perturbação dos sistemas de controle PI e PID.....	26
6. CONCLUSÃO.....	27
7. PRODUTO PIEPEX.....	28
8. REFERÊNCIAS.....	31
APÊNDICE.....	33
APÊNDICE A – Programação da calibração do termopar.....	33
APÊNDICE B – Programação do controlador on-off com zona morta.....	34
APÊNDICE C – Programação do controlador PID.....	35

1. INTRODUÇÃO

O controle de um processo é feito a partir da interação ordenada de suas variáveis, mantendo-as o mais próximo possível dos valores ideais (GARCIA, 2021). Para um engenheiro, é indispensável saber quanto, como e quando mudar o valor de uma variável para obter uma melhor resposta do sistema. Os controladores podem atuar sobre diferentes variáveis, como velocidade, pressão, temperatura, movimento, nível, vazão, entre outras. A título de exemplo, no trabalho de LIMA et al. (2019) foi explorado um protótipo para o monitoramento da temperatura para o armazenamento de vacinas, ou no estudo de MATHEUS (2019), que descreve o processo de controle de temperatura durante o aquecimento do mosto para uma cervejaria artesanal.

OGATA (2014) ressalta a importância do controle automático no campo da engenharia e da ciência. É um componente intrínseco em sistemas de veículos espaciais, sistemas robóticos, sistemas de manufatura e quaisquer operações industriais que envolvam o controle de temperatura, pressão, umidade, viscosidade, vazão etc. De acordo com LUGÃO (2015), os sistemas de controle de temperatura são compostos pela fonte de alimentação, sensores, utilizados para medir com precisão a temperatura, controladores (digitais ou analógicos) e atuadores, que vão aquecer ou resfriar dependendo da ação transmitida pelos controladores com o *setpoint* definido. A Figura 1 apresenta a arquitetura básica de um controlador de temperatura.

Figura 1 - Arquitetura básica de um controlador de temperatura.



Fonte: Autoral

Aulas teóricas e práticas sobre controladores, especialmente de temperatura, são indispensáveis devido à sua importância tecnológica. Os laboratórios de ensino de engenharia do Instituto de Ciência e Tecnologia (ICT) da Universidade Federal de Alfenas (Unifal-MG) não dispõem de uma plataforma para ensaios de controladores no momento em que este texto foi escrito. Por essas razões, este trabalho teve como objetivo o desenvolvimento de um protótipo de uma plataforma de baixo custo que seja programável e de experimentação de curta duração ou duração compatível com uma aula prática típica de um curso de engenharia.

2. OBJETIVOS

2.1. Objetivo principal

Desenvolvimento do protótipo de baixo custo que seja programável e de experimentação de curta duração ou duração compatível com uma aula prática típica de um curso de engenharia.

2.2. Objetivos específicos

- Seleção de sensor de temperatura;
- Calibração do sensor de temperatura escolhido;
- Desenvolvimento de um controlador *on-off* com zona morta;
- Desenvolvimento de um controlador PI;
- Desenvolvimento de um controlador PID;
- Comparação de comportamento entre as diferentes arquiteturas de controlador.

3. REVISÃO BIBLIOGRÁFICA

3.1. Controladores

COLOGNI (2008) define um sistema de controle como um dispositivo em que uma quantidade medida é utilizada para modificar o comportamento do sistema por meio de computação e atuação. Ele enxerga a realimentação como uma ferramenta para gerenciamento de incertezas, sejam elas paramétricas, de condições de funcionamento, de limites práticos ou advindas de alteração de sinais externos (ruídos, perturbação) não controlados.

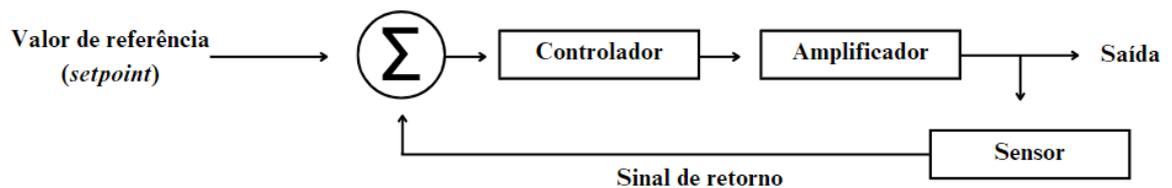
Um sistema que estabeleça relação de comparação entre a saída e a entrada de referência, utilizando a diferença como meio de controle, é definida, para OGATA (2014), como sistema de controle com realimentação. Um exemplo disto é o controle de temperatura de um ambiente, que, ao medir a temperatura real do ambiente, compara com a temperatura de referência (desejada) ativando ou desativando um equipamento de resfriamento ou aquecimento, de modo

que assegure que a temperatura ambiente permaneça em um nível confortável independentemente das condições exteriores (OGATA, 2014).

O controle apresentado na Figura 2 é do tipo malha fechada, que, de acordo com FACCIN (2004), trata-se da situação do sistema de controle em que a ação de controle depende da saída. Ele também define outra situação que pode ocorrer, em malha aberta, na qual a ação de controle é independente da saída.

Todo esse controle acontece por realimentação, em que o procedimento através do qual o sinal de saída de um sistema (ou circuito) é transferido para a entrada deste mesmo sistema, com o objetivo de diminuir, amplificar ou controlar a saída deste. (FACCIN, 2004).

Figura 2 - Esquema genérico de controlador em malha fechada

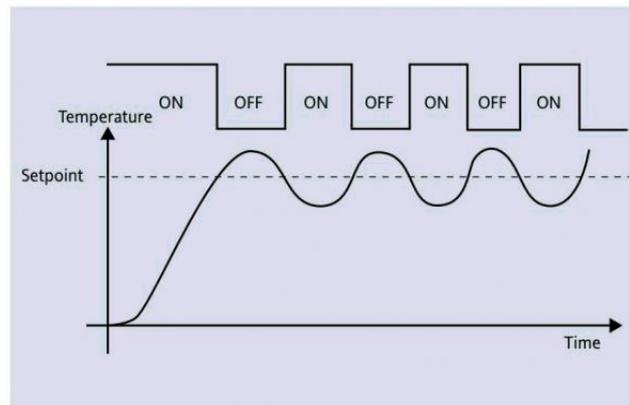


Fonte: Citysystem, (2016).

3.2. Controlador *ON-OFF*

O controle *on-off* ou liga/desliga é o método mais simples de controle, em que a saída do controlador assume apenas dois estados: ligado/desligado (1/0) e altera sempre que a variável controlada cruza o valor desejado. Sempre que a variável real é menor que a variável desejada (*setpoint*), o controle liga. A variável sendo maior, ele desliga (MATHEUS, 2019). A curva característica deste sistema é apresentada na Figura 3.

Figura 3 - Modelo de resposta de um sistema *on-off*



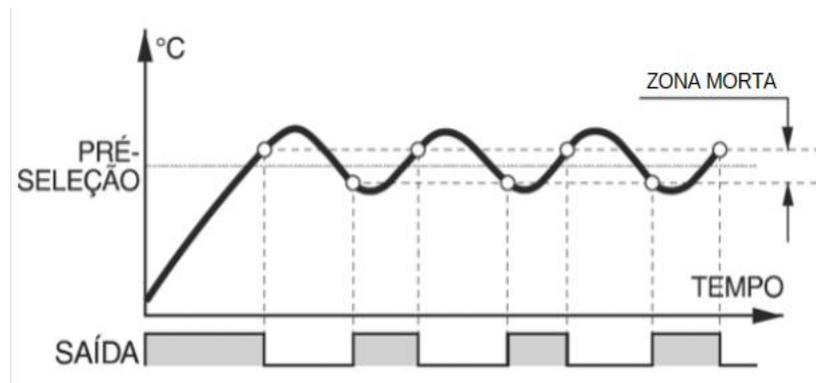
Fonte: Citysystem, (2016).

Este modo descontínuo não é apropriado para cenários que carecem maior precisão, uma vez que qualquer diferença entre a variável desejada e o valor real provocará uma resposta com potência máxima. É indicado para controle de sistemas simples com dinâmicas lentas, como o controle de temperatura de um ar-condicionado (FACCIN, 2004).

A Figura 3 representa o comportamento de um sistema no controle *on-off*, em que a temperatura alterna ao longo do *setpoint*, devido à ação extrema apresentada pelo controlador que utiliza toda sua capacidade e excede o valor estipulado.

A zona morta de um sistema, também conhecida como banda morta, no seu sentido mais geral, é a faixa de valores da entrada correspondente a um mesmo valor de saída (AGUIRRE, 2013). Este parâmetro é calculado no algoritmo sendo representado por um adendo ou subtração ao valor de referência pré-definido. Ou seja, em uma temperatura de referência definida a 60 °C, pode se ter como banda morta a faixa de +0.5 e -0.5 graus Celsius para acionamento e desligamento do sistema. A Figura 4 representa a curva característica de um controlador *on-off* com zona morta, em que a linha pontilhada representa a faixa de atuação da zona morta do sistema.

Figura 4 - Curva característica de um controlador *on-off* com zona morta.



Fonte: 1 Adaptada de FUENTES (2005).

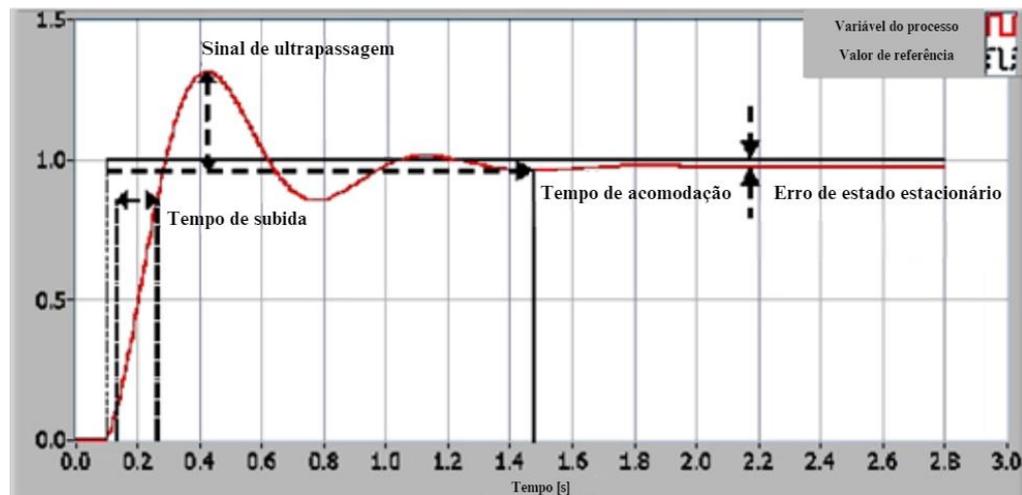
3.3. Controlador PID

Não há dúvidas que, em processos industriais, o algoritmo de controle mais usado continua sendo o PID (proporcional, integral e derivativo). Isso se deve à sua fácil implementação, versatilidade e resultados satisfatórios. Estes parâmetros podem ser utilizados individualmente ou em conjunto, conforme necessidade do sistema em questão. (MATHEUS, 2019).

O controle PID tem como objetivo minimizar ou eliminar as oscilações e fazer com que a temperatura se estabilize no alvo pré-definido (*setpoint*). Isso é feito a partir do monitoramento da aproximação da temperatura limite e, sistematicamente, diminui a potência a ser entregue ao sistema, de forma a não ultrapassá-la. Isto é possível a partir da diferença entre a temperatura atual e a temperatura limite, que é chamada de “erro”, e à medida que se aproxima do alvo, o erro diminui. (FERRANDO et.al, 2017).

A curva característica de um controlador PID é apresentada na Figura 5. Nela, é possível identificar o *rise time*, que é o tempo de subida da temperatura até atingir o *setpoint*, o *overshoot*, que é a ultrapassagem do *setpoint*, o *settling time*, que refere-se ao tempo de acomodação da curva, ou seja, tempo para estabilizar na temperatura de *setpoint*, e por fim, o *steady-state error*, ou erro de estado estacionário, que é um regime em que a temperatura do sistema fica muito próxima do *setpoint*, ou seja, é calculada a partir da diferença da variável do processo e o *setpoint*.

Figura 5 - Curva característica de um sistema PID.



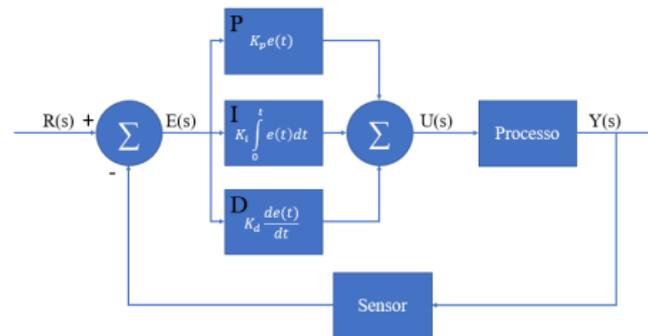
Fonte: Adaptado de The PID Controller & Theory Explained – NI (2023).

A ação Proporcional faz com que o sistema reaja ao erro presente, possibilitando uma ação imediata diante de variações ou perturbações; a ação Integral elimina erros em regime permanente - em longo prazo - do tipo salto ou degrau, e a ação Derivativa antecipa o comportamento do processo (ALPI, L. et al, 2019).

A seleção de parâmetros do controlador que garanta determinada especificação de desempenho é chamada de sintonia do controlador e pode ser feita através do ajuste dos valores de K_p , K_i e K_d sugerida pela regra de Ziegler-Nichols, que fornece estimativas dos valores dos parâmetros e proporcionam um ponto de partida na sintonia (OGATA, 2014).

Dentro de um processo genérico em controlador PID, podemos observar o valor de *setpoint*, $R(s)$, sinal de erro, $E(s)$, somatório do PID que vai para o atuador do sistema, $U(s)$ e o valor da variável controlada do processo, $Y(s)$ (ALVES, 2023). A Figura 6 contém um processo genérico de um controlador PID.

Figura 6 - Esquema genérico de um controlador PID.



Fonte: ALVES (2023).

O controlador proporcional pode atuar de forma unitária, em que sua ação depende apenas da diferença entre o valor do *setpoint* e o valor da variável de processo, denominada erro. O ganho proporcional (K_p), determina então a relação entre a resposta de saída e o sinal de erro, que está ligado à velocidade de resposta do sistema, no qual seu aumento ocasiona uma velocidade de resposta maior (ALVES,2023).

Uma característica importante desta ação de controle, é a existência de um erro residual permanente sempre que ocorre uma alteração de carga, e o sistema que se pretende controlar seja do tipo 0. O erro estacionário que é dependente de K_p e da carga, pode ser minimizado por um aumento de K_p . No entanto deve-se notar que o aumento deste parâmetro conduz a um aumento do tempo de estabelecimento e eventualmente até à instabilidade. Conclui-se assim que este tipo de controlador só pode ser usado quando o ganho proporcional é suficientemente elevado para reduzir o erro estacionário a um nível aceitável, ou quando não são previsíveis alterações frequentes da carga (LOURENÇO, 1997).

A ação integral assegura que a saída do processo esteja de acordo com o *setpoint*, ou seja, com erro nulo. Ela atua fornecendo uma saída do controlador que é proporcional à integral do erro acumulado. (ALVES, 2023)

O modo integrativo responde ao valor integral do erro, ou seja, responde de forma mais rápida e precisa do que o controle proporcional, que atua somente no valor do erro. Com o aumento do parâmetro integrativo, o *setpoint* é atingido mais rapidamente, porém, este tipo de situação apresenta uma oscilação mais agressiva antes da estabilização, sendo necessário que se diminua o ganho integral a fim de se obter uma resposta menos oscilatória (MATHEUS, 2019).

A componente integral elimina o erro estacionário de posição, independentemente do sistema que se pretende controlar, mas aumenta o tempo de estabelecimento e piora a estabilidade relativa (LOURENÇO, 1997).

Já o modo derivativo apresenta ação de controle proporcional à taxa de variação do sinal de erro, ou seja, a sua derivada no tempo. Ele estimula a tendência de aumento ou diminuição do erro futuro. Assim, é capaz de aumentar a velocidade de correção do processo, pois atua de forma antecipatória quando são detectadas variações no sinal de erro. Devido a isto, a ação derivativa é bastante sensível a erros de alta frequência, como ruídos de processos e mudanças no valor de referência da variável controlada (FACCIN, 2004).

3.3.1. Regra de sintonia de Ziegler-Nichols para controladores PID

Segundo OGATA, 2014:

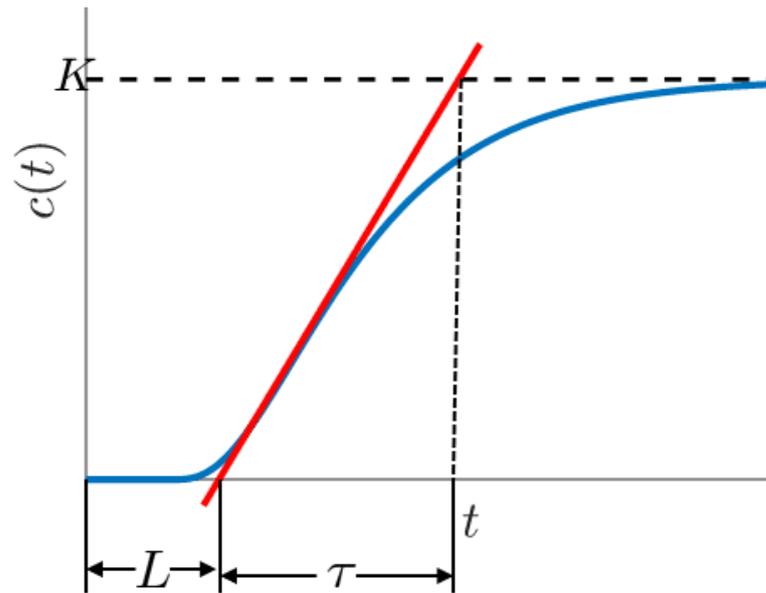
“Ziegler e Nichols propuseram regras para a determinação de valores do ganho proporcional K_p , do tempo integral T_i e do tempo derivativo T_d , baseadas nas características da resposta transitória de dada planta. Essa determinação dos parâmetros dos controladores PID ou de sintonia dos controladores PID pode ser feita por engenheiros de campo, por meio de experimentos com a planta.”

3.3.2. Obtenção das Variáveis T e L para o primeiro método

Conforme OGATA, 2014:

“A curva com o formato em S pode ser caracterizada por duas constantes, o atraso L e a constante de tempo T . O atraso e a constante de tempo são determinados desenhando-se uma linha tangente no ponto de inflexão da curva com o formato em S e determinando-se a intersecção da linha tangente com o eixo dos tempos e a linha $c(t) = K$ ”, como mostra a Figura 7.

Figura 7 - Representação visual das variáveis T e L em curva de resposta em forma de S.



Fonte: OGATA (2014).

3.3.3. Regra e sintonia de Ziegler-Nichols primeiro método

A regra de sintonia de Ziegler-Nichols, baseada na resposta ao degrau da planta (primeiro método) está representada na Tabela 1, na qual realiza uma sintonização grosseira.

Tabela 1 - Regra de sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta.

Controlador	K_p	K_i	K_d
P	T/L	0	0
PI	$0.9 T/L$	$0.3/L$	0
PID	$1.2 T/L$	$0.5/L$	$0.5L$

3.4. Microcontroladores

CHASE, et.al. (2007) classifica os microcontroladores como pequenos sistemas computacionais bastante poderosos que englobam em um único chip: interfaces de entrada/saída digitais e analógicas, periféricos importantes como a memória RAM, memória FLASH, interfaces de comunicação serial, conversores analógicos/digitais e temporizadores/contadores.

São capazes de executar e armazenar os programas escritos para eles, assim como a capacidade de absorver mais funções com o incremento de periféricos, como comunicação USB (CHASE et. al., 2007). Hoje, existem várias versões de placas microcontroladas e aquisições de dados no mercado com preços populares, mas sem dúvida, a mais conhecida é a Arduino® (MARTINAZZO, 2014).

3.5. Arduíno

MCROBERTS (2018) define Arduino® (Figura 8) como um pequeno computador que você pode programar para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele. É uma plataforma embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de hardware e software. Pode ser conectado a LEDs, botões, interruptores, sensores de temperatura, de pressão, de vazão, ou qualquer outro dispositivo que emita dados ou possa ser controlado.

Figura 8 - Arduino® UNO



Fonte: Autoral.

Este microcontrolador AVR da Atmel é capaz de fazer leitura simultânea de dezenas de sensores, tanto digitais quanto analógicos. Inclui um editor de código de fonte livre, código aberto de linguagem C/C++ e, via cabo USB, é capaz de receber uma programação de um computador que controla a saída de um sistema. A IDE (Ambiente Integrado de Desenvolvimento, em português) do Arduino® é escrita em Java, ou seja, é portátil para diversos sistemas operacionais (MARTINAZZO, 2014).

A sua aplicação vai desde a segurança até a automação de processos eletromecânicos. No campo didático, tem grande aplicabilidade, pois qualquer fenômeno físico detectável por sensor pode ser lido pelo sistema.

4. METODOLOGIA

4.1. Materiais

A Tabela 2 apresenta os materiais usados e seus respectivos valores.

Tabela 2 - Materiais e custos (custos pesquisados em 15/07/2023).

Materiais	Custo
Arduino UNO	R\$ 26,00
Módulo Max6675	R\$ 15,00
Termopar tipo-K	R\$ 15,00
Display LCD 0x16 e Módulo IC2	R\$ 38,00
Relé de estado sólido 2A	R\$ 13,00
Resistencia 350W-110V	R\$ 20,00
Termopar calibrado IMPAC HD 2307.0 RTD ThermoMeter	-
Total	R\$ 127,00

4.2. Escolha do sensor de temperatura

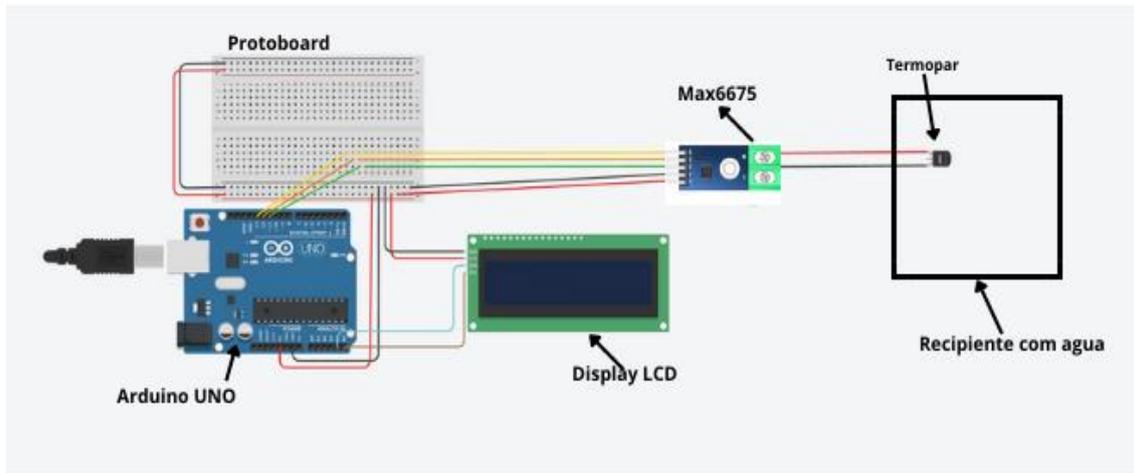
Avaliaram-se os medidores de temperatura: coluna de álcool, termistor e termopar. Os critérios adotados para a seleção foram: estabilidade, tempo de resposta, linearidade, possibilidade de comunicação com Arduino® e custo médio.

4.3. Calibração do termopar tipo-K

4.3.1. Ligações e montagem do circuito

Antes do desenvolvimento do trabalho, conferiu-se o funcionamento de todos os componentes e cabos a serem utilizados com o auxílio de um multímetro. A montagem do circuito foi feita como apresentada na Figura 9. O termopar tipo-K foi ligado ao módulo Max6675 seguindo os pólos positivos e negativos indicados nos componentes. A Tabela 3 apresenta estas conexões.

Figura 9 - Esquemática da montagem do circuito para calibração.



Fonte: Autoral.

Tabela 3 - Sistematização das ligações entre a conexão Max6675 - Arduino®.

Conexão	Max6675	Arduino UNO
Ligação 1	GND	GND
Ligação 2	VCC	Polo positivo 5V
Ligação 3	SO	Porta digital, pino 11
Ligação 4	CS	Porta digital, pino 12
Ligação 5	SCK	Porta digital, pino 13

Após a montagem, com a finalidade de não utilizar a comunicação serial e Arduino®-Excel, foi adicionado um display LCD soldado ao módulo IC2, que funcionou como um encurtador para o circuito, com isso a comunicação do LCD mais o módulo IC2 com Arduino® é feita conectando os polos negativos e positivos do módulo IC2 aos respectivos do Arduino®, em conjunto com a saída SDA do módulo IC2 na porta analógica A4, e saída SCL do módulo IC2 à porta analógica A5, como demonstra a Tabela 4.

Tabela 4 - Sistematização das ligações entre a conexão módulo IC2 - Arduino®.

Conexão	Módulo IC2	Arduino UNO
Ligação 1	GND	GND
Ligação 2	VCC	Polo positivo 5V
Ligação 3	SDA	Porta analógica A4
Ligação 4	SCL	Porta analógica A5

Deste modo, com os circuitos prontos e carregado com a programação de nome "Programação da calibração do termopar " alocada no apêndice A, pode-se utilizar o Arduino® em conjunto aos instrumentos citados, como um termômetro descalibrado.

4.3.2. Procedimento experimental

Uma calibração indireta foi realizada através da comparação entre as temperaturas medidas pelo aparato e pelo termopar IMPAC adotado como referência.

As duas sondas foram posicionadas a uma distância não maior que 2 centímetros dentro do recipiente, preenchido com água a temperaturas controladas por um banho termostaticado.

Nesta calibração, para cada temperatura das 10 adotadas entre aproximadamente 10°C e a temperatura de ebulição da água, coletaram-se 5 valores de temperatura em intervalos de 15 segundos usando o termopar de referência e o aparato, após o equilíbrio térmico no banho termostaticado.

As incertezas das medidas obtidas no termopar de referência foram determinadas através do desvio padrão amostral em conjunto com a incerteza instrumental, referente à sua carta de calibração, respectivamente nas Equações 1 e 2. E o desvio padrão das medidas do aparato foram obtidas através da Equação 1.

$$\sigma_{desvio}^2 = \sum_{i=1}^N \frac{(x_i - \bar{X})^2}{N-1} \quad (1)$$

$$\sigma^2 = \sigma_{desvio}^2 + \sigma_{instrumental}^2 \quad (2)$$

Estes pontos foram ajustados a uma reta $y = ax + b$ pelo método dos mínimos quadrados em conjunto ao teste de qualidade do χ^2 reduzido. As Equações 3 e 4 apresentam a função χ^2 e χ^2 reduzido que foi adotada para avaliar a qualidade deste ajuste no procedimento de calibração.

$$\chi^2 = \sum_{i=1}^N \frac{(y_i - f(x_i))^2}{\sigma_i^2} \quad (4)$$

$$\chi_{red}^2 = \frac{\chi^2}{\vartheta} \quad (5)$$

4.4. Controlador *on-off* zona morta

4.4.1. Ligações e montagem do circuito

A montagem do circuito foi feita como apresentada na Figura 10, em que o termopar tipo-K foi ligado à placa amplificadora Max6675, e as conexões da placa Max6675 com o Arduino®, conforme a Tabela 3.

Para a montagem do ebulidor ao relé de estado sólido, como demonstra a Figura 9, foi necessário ligar um dos fios do ebulidor ao interruptor que mediará a conexão desse mesmo fio à fonte de energia, enquanto o outro fio do ebulidor será ligado à porta positiva do relé. Para fechar o circuito, ligou-se a porta negativa do relé ao polo negativo da fonte de energia, dessa maneira, mesmo que a programação esteja acontecendo e mandando o ebulidor esquentar, pode-se decidir se o mesmo recebe ou não energia a partir do interruptor.

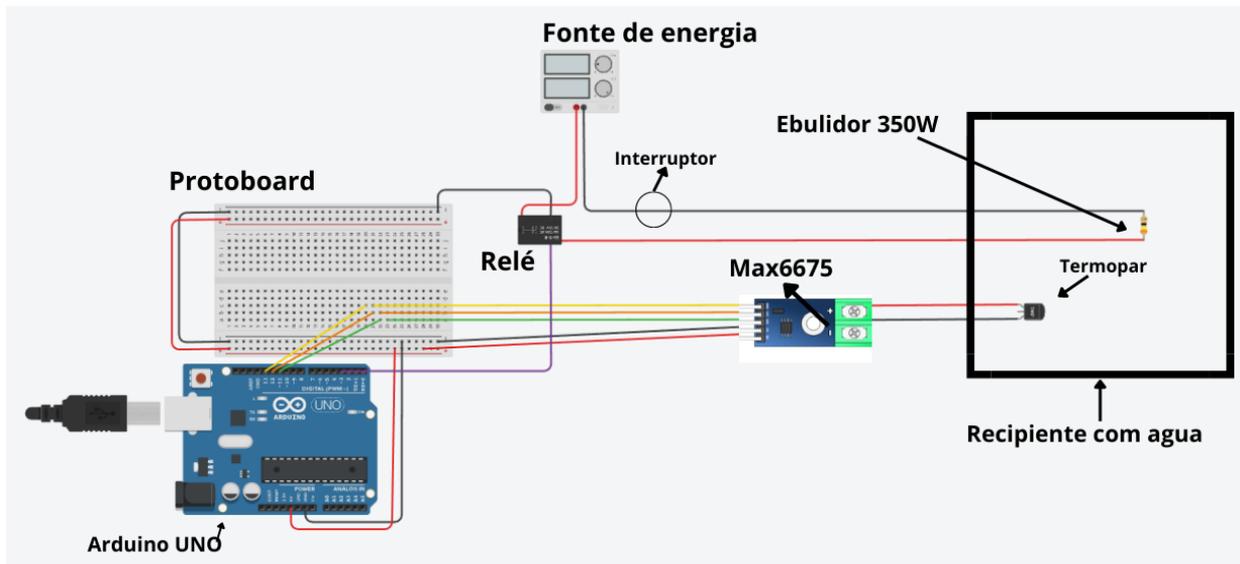
As conexões do relé de estado sólido ao Arduino® foram feitas conforme a Tabela 3.

Tabela 3 - Sistematização das ligações entre a conexão relé de estado sólido - Arduino®.

Conexão	Relé de estado	Arduino UNO
Ligação 1	Polo negativo	GND
Ligação 2	Polo positivo	Polo positivo 5V
Ligação 3	Polo mediador	Porta digital, pino 3 (PWM)

Dessa maneira, ligou-se o Arduino® à porta USB do computador, e, utilizando a interface Arduino IDE foi possível compilar a programação referente ao controlador de temperatura *on-off* zona morta, apêndice B, para o Arduino® UNO.

Figura 10 - Esquematização da montagem do circuito, controlador *on-off* zona morta.



Fonte: Autoral.

4.4.2. Procedimento experimental

Os testes e obtenção de dados foram realizados com cinco diferentes temperaturas de *setpoint*: 40°C, 50°C, 60°C, 70°C e 80°C. Todos foram realizados com a mesma quantidade de água e em triplicata, com o propósito de registrar as relações entre as temperaturas e os tempos dos momentos de aquecimento e resfriamento do sistema.

Para iniciar o procedimento experimental, um recipiente de polímero rígido foi utilizado, preenchido com 400 ml de água. A temperatura média inicial da água durante os experimentos foi de 19°C. Em seguida, uma tampa feita de isopor que continha o termopar e o aquecedor unidos, foi colocada no recipiente para reduzir a transferência de calor com o ambiente externo do sistema. O aquecedor foi posicionado no fundo do copo, enquanto o termopar foi colocado no meio do copo, de modo a evitar interferências da corrente de convecção térmica que subiria do aquecedor.

Ao enviar a programação referente ao controlador de temperatura *on-off* com zona morta para o Arduino®, configurou-se uma temperatura de *setpoint* para cada uma das cinco temperaturas, mantendo o parâmetro de limite superior e inferior para a zona morta, sendo estes de $\pm 0.5^\circ\text{C}$ em relação à temperatura de *setpoint*, descritas na programação no Apêndice B.

A obtenção dos dados foi realizada por meio da interface Excel (2023), em que se habilitou a opção de streaming de dados para se conectar ao Arduino® UNO para a leitura, armazenamento e tratamento dos dados obtidos.

Foi feito o registro de pelo menos quatro chaveamentos, sendo eles do momento em que a temperatura atinge o limite inferior e o controlador ativa 100% da potência da resistência, até a temperatura do limite superior na qual o controlador interrompe o aquecimento.

4.5. Controlador PI e PID

4.5.1. Ligações e montagem do circuito

As ligações e montagem do circuito foram de acordo com o padrão descrito na metodologia, Tópico 3, mais especificamente no Subtópico 3.3.1.

4.5.2. Procedimento Experimental

Após a obtenção das triplicatas de pelo menos quatro chaveamentos para as temperaturas de *setpoint*, foi utilizado o primeiro método de Ziegler-Nichols para calcular os coeficientes T e L conforme a Figura 7, que foram necessários para obter os coeficientes progressivo, derivativo e integrado do método de controle PID.

Com os fatores calculados pelo método de Ziegler-Nichols para os controles PI e PID conforme a Tabela 1, colocou-se em funcionamento os sistemas com as mesmas cinco temperaturas de *setpoint*, 40°C, 50°C, 60°C, 70°C e 80°C. Em todos os aquecimentos, foi medido o mesmo volume de água, com aproximadamente 400 ml de água à $\pm 18^\circ\text{C}$, para identificar as relações entre as temperaturas máximas e mínimas, faixa de controle, sinal de *overshoot* e as diferenças entre os métodos de controle.

Para iniciar o procedimento experimental, foi utilizado o mesmo sistema e dimensões do controlador *on-off*, recipiente de polímero, tampa de isopor com o termopar e resistência nas mesmas posições. Com isso, o sistema estava montado e completo para o início da obtenção de dados experimentais.

Utilizando a programação referente ao controlador de temperatura PI e PID, apêndice C, realizou-se triplicata dos controles para cada temperatura de *setpoint* e seus respectivos

coeficientes, K_p e K_i para o controlador PI e, K_p , K_i e K_d para o controlador PID. A obtenção de dados foi realizada por meio da interface Excel.

4.6. Ensaio de perturbação

Após a obtenção dos dados para as temperaturas de *setpoint*, foi realizado um experimento para analisar a resposta do sistema de controle a um distúrbio. Utilizou-se o mesmo aparato das obtenções dos dados PI e PID, porém trocando o recipiente de polímero por um recipiente isolante, de modo a minimizar as correntes de condução e convecção do sistema para o meio, e adicionando uma quantidade de 10% de água a 0°C, em relação ao volume total de água aquecida.

Usou-se a interface Excel conectada ao Arduino® para analisar a resposta ao distúrbio. O procedimento seguiu um padrão: após o primeiro registro de temperatura igual ou maior à de *setpoint*, cronometrou-se 3 minutos para garantir que o sistema estava estabilizado, e então, 10% de água à 0°C do volume do recipiente foi adicionado, sempre buscando adicionar todo este volume junto e em um curto período.

Mesmo que o sistema tenha se estabilizado com menos de três minutos após o distúrbio, aguardou-se esse tempo para que os ensaios de perturbação seguissem o mesmo padrão.

5. RESULTADOS E DISCUSSÃO

5.1. Escolha do sensor

A Tabela 5 representa a forma da escolha do sensor a ser utilizado no aparato, partindo dos princípios de estabilidade, tempo de resposta, linearidade, comunicação com Arduino® e custo médio, de forma que as opções partiriam dos sensores: coluna de álcool, termistor e termopar.

Tabela 5 - Análise da viabilidade dos sensores de temperatura.

Sensor	Estabilidade	Tempo de resposta	Linearidade	Comunicação Arduino	Custo médio
Coluna	Bom	Lento	Linear	Não	Baixo
Termistor	Bom	Rápido	Não linear	Sim	Baixo
Termopar	Bom	Rápido	Linear	Sim	Baixo

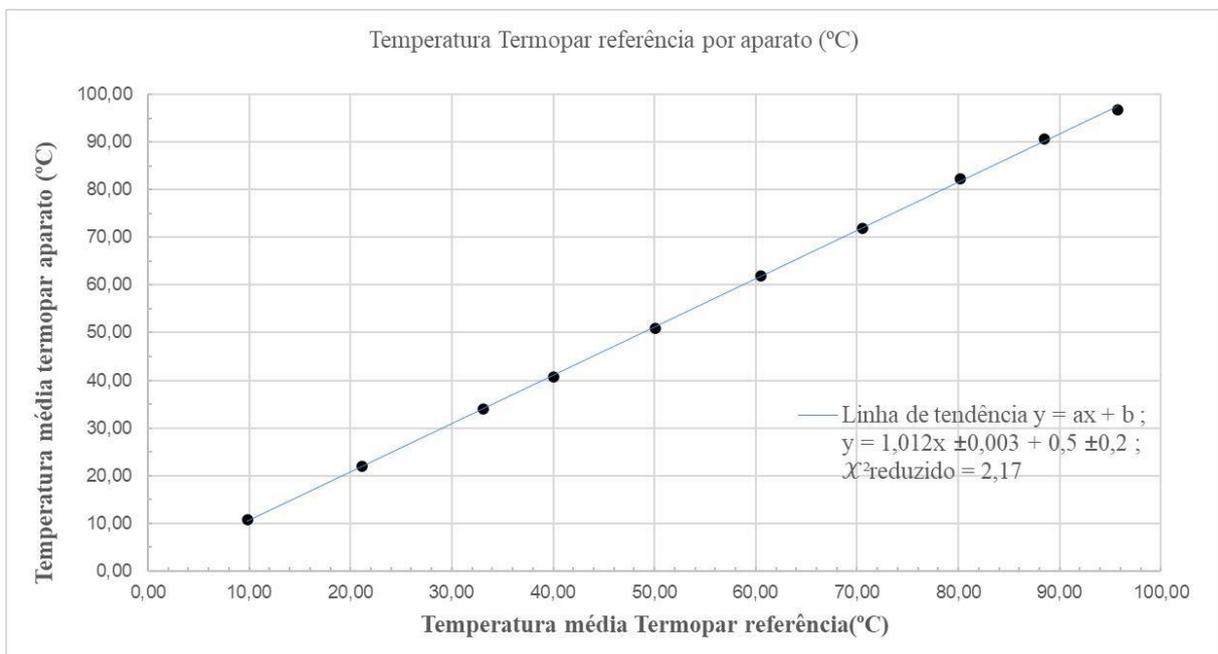
Concluiu-se que o termopar foi a melhor opção, com baixo custo e que tem seu comportamento de forma linear, com possibilidade de comunicação com o Arduino®.

5.2. Calibração do Termopar Tipo K

Apresenta-se na Figura 11 a curva de calibração do sensor descalibrado, em que os dados obtidos pelo experimento de calibração, dispostos em um gráfico se apresentam conforme uma linha de tendência linear, como o esperado, e com um coeficiente angular no valor de 1.012 ± 0.003 , ou seja, 1.2% de erro no qual foi o coeficiente utilizado para a calibração do sensor. Ademais, os dados da incerteza propagada para o eixo y, apresentaram valores ínfimos para serem representados e notados no gráfico, de modo que por opção, não foram apresentados.

Para a determinação da qualidade do ajuste, foram utilizadas as Equações 4 e 5, que resultaram em um valor de 2.17 para o coeficiente de determinação χ^2 reduzido, valor este congruente às temperaturas medidas pelo aparato, levando em consideração o termopar de referência IMPAC.

Figura 11 - Gráfico da curva de calibração do aparato, em conjunto a equação da reta da tendência linear e o valor do χ^2 reduzido.

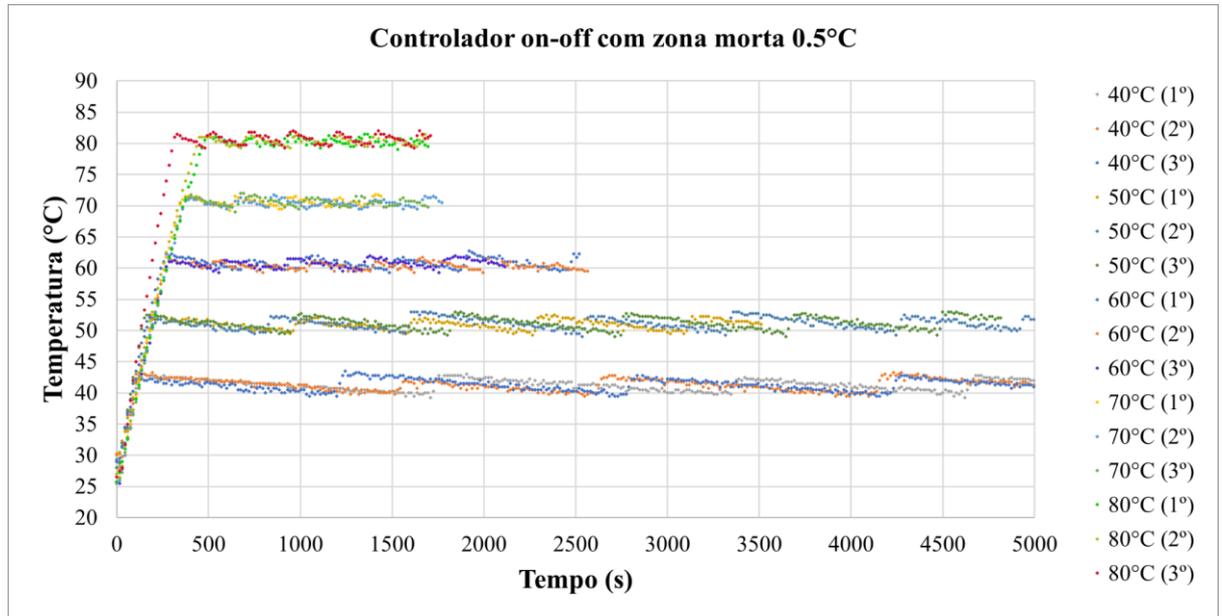


Fonte: Autoral.

5.3. Controlador *on-off* com zona morta

Os dados de cada temperatura de *setpoint* no controlador *on-off* com zona morta, possibilitaram construir os gráficos de temperatura em função do tempo, representados na Figura 12.

Figura 12 - Aquecimento em triplicata de temperatura em função do tempo, do controlador *on-off* com zona morta de 0.5°C para as temperaturas de *setpoint*.



Fonte: Autoral.

Nota-se que a resposta do sistema foi rápida e ultrapassou o limite superior da zona morta, ou seja, a temperatura máxima do sistema foi muito elevada. O controle *on-off* gera uma saída de 100% da potência à resistência ao registrar uma temperatura próxima ao limite inferior e, ao atingir o limite superior de temperatura o controlador gera uma saída de 0%. Mas devido a localização do sensor, existe um determinado tempo para que ele registre a variação de temperatura, devido ao calor que foi gerado ser dissipado pelo sistema até o sensor, ocorrendo uma variação de temperatura mesmo após o desligamento da resistência.

Com o propósito de analisar o comportamento do controlador, apresenta-se a Tabela 6 com informações das temperaturas máximas e mínimas médias, faixa de temperatura e tempo médio de chaveamento.

Tabela 6 - Valores médios de temperatura máxima e mínima, faixa de controle e tempo médio de chaveamento do controlador *on-off*.

Setup	on-off			
	Temperatura máxima média [°C]	Temperatura mínima [°C]	Faixa de controle [°C]	Tempo médio de chaveamento [s]
40°C	43.42 ± 0.11	39.25 ± 0.11	4.17 ± 0.11	1413.89
50°C	53.00 ± 0.11	49.00 ± 0.11	4.00 ± 0.11	792.5
60°C	62.25 ± 0.27	59.08 ± 0.27	3.17 ± 0.27	341.5
70°C	71.92 ± 0.27	69.00 ± 0.27	2.92 ± 0.27	272.5
80°C	81.83 ± 0.31	78.83 ± 0.31	3.00 ± 0.31	227.5

Os valores médios de temperatura máxima, mínima, faixa de controle e tempo de chaveamento foram retirados das curvas de aquecimento, no qual explicita um controlador com grandes faixas de controle. Como exemplo, no *setpoint* de 40°C obteve-se uma faixa de controle de $4.17 \pm 0.11^\circ\text{C}$, representando aproximadamente 10.5% em relação ao *setpoint*. Em temperaturas de *setpoint* elevadas, o controlador operou em menores faixas de controle.

Enfim, é notório que o controle *on-off* não opera de forma estável próximo a temperatura de *setpoint*, por conta dos longos chaveamentos, como representado na curva característica do controlador presente na revisão bibliográfica, Figura 3.

5.4. Cálculo da sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta

Utilizando os gráficos e dados dos experimentos do controlador de temperatura *on-off* zona morta foi possível analisar e determinar as variáveis T e L conforme a Figura 6, a partir da média das triplicatas, calculando os fatores Kp e Ki e Kd para os controladores PI e PID, segundo a Tabela 7.

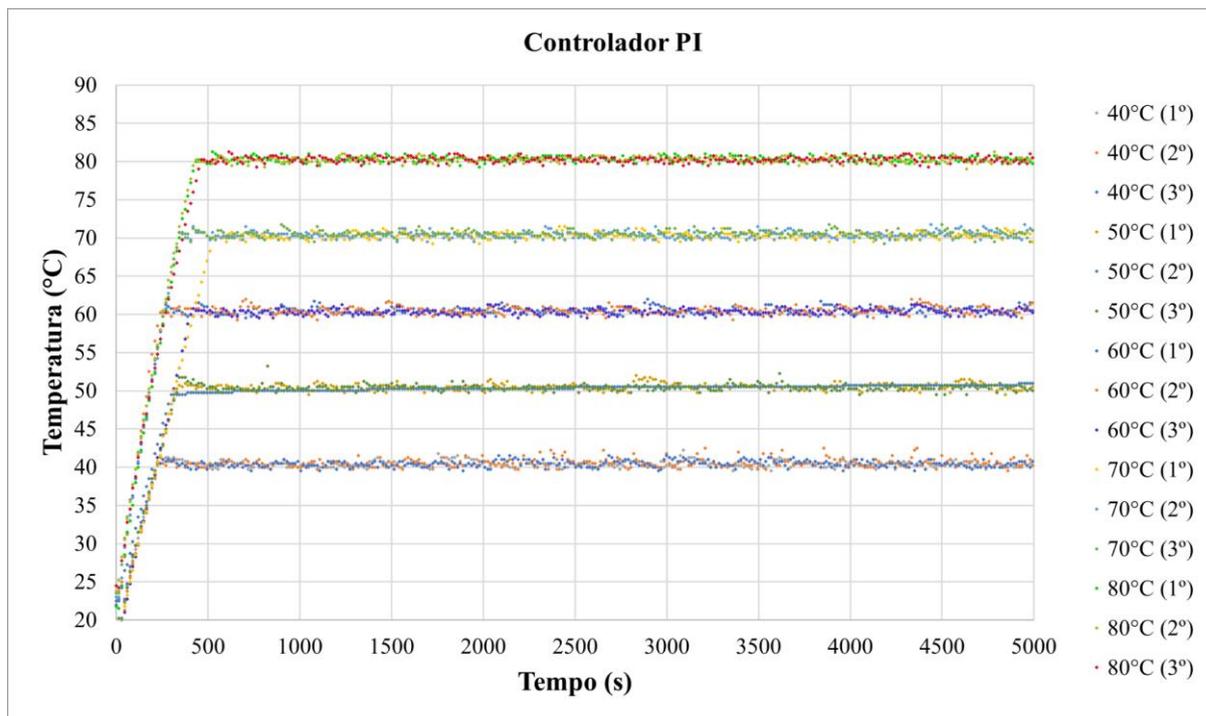
Tabela 7 - Valores de T, L, Kp, Ki e Kd para cada temperatura de *setpoint*.

Setpoint	T	L	PI			PID		
			Kp	Ki	Kd	Kp	Ki	Kd
40°C	61.6 s	20 s	2.772	205.33	0	3.696	40	10
50°C	126.6 s	16.6 s	6.863	422	0	9.152	32.3	9.3
60°C	228.3 s	20 s	10.273	761	0	13.698	40	10
70°C	336.6 s	10 s	30.294	1122	0	40.392	20	5
80°C	395 s	10 s	35.55	1316.6	0	47.4	20	5

5.5. Controlador PI

Introduzindo os fatores de ganho proporcional e integrativo na programação do controlador PI, foi possível registrar os dados da Figura 13, e calcular as médias das temperaturas máximas e mínimas, a faixa de controle e o sinal do *overshoot*, representados na Tabela 8.

Figura 13 - Aquecimento em triplicata de temperatura em função do tempo, do controlador PI para as temperaturas de *setpoint*.



Fonte: Autoral.

Tabela 8 - Valores médios do controle PI para as temperaturas máximas e mínimas, faixa de temperatura do controle e temperatura máxima do *overshoot*.

Setup	PI			
	Temperatura máxima média [°C]	Temperatura mínima [°C]	Faixa de controle [°C]	Overshoot [°C]
40°C	42.33 ± 0.11	39.33 ± 0.11	3.00 ± 0.11	1.17 ± 0.11
50°C	52.50 ± 0.11	49.33 ± 0.11	3.17 ± 0.11	1.75 ± 0.11
60°C	62.17 ± 0.27	59.25 ± 0.27	2.92 ± 0.27	1.00 ± 0.27
70°C	71.75 ± 0.27	69.10 ± 0.27	2.65 ± 0.27	0.75 ± 0.27
80°C	81.42 ± 0.31	79.17 ± 0.31	2.25 ± 0.31	0.50 ± 0.31

Os valores da Tabela 8 representam uma faixa de controle máxima de valor $3.00 \pm 0.11^\circ\text{C}$, ou seja, aproximadamente 7.5% da temperatura de *setpoint* 40°C . O sinal de *overshoot* não representou o maior pico do aquecimento.

Ademais, experimentos geraram curvas de aquecimentos da Figura 13, de comportamento mais estável em relação aos chaveamentos provocados pelo *on-off* zona morta, ao utilizar de uma porcentagem para o ganho do resistor.

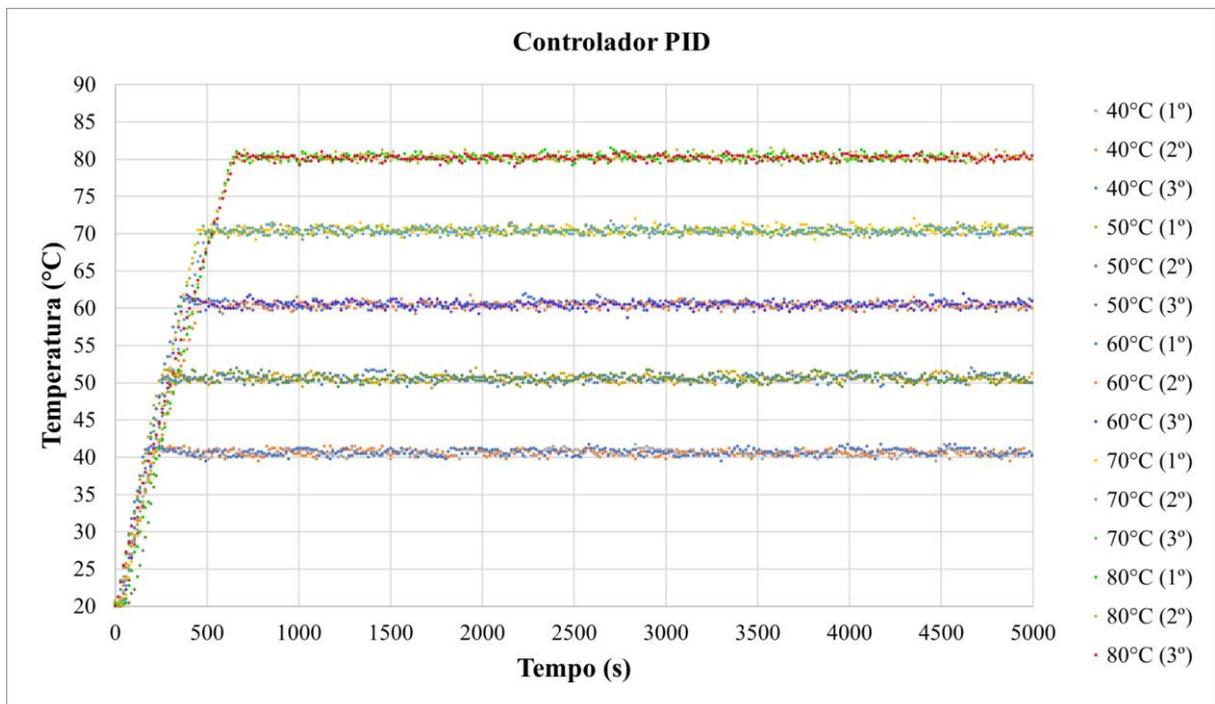
O comportamento das curvas se adequou ao controle PI, demonstrando inicialmente um sinal de *overshoot* em seguida de um tempo de acomodação, podendo ou não apresentar um erro de estado estacionário. Pode ser considerado um erro sistemático do sistema de controle

em relação aos valores dos fatores atribuídos, tornando a curva característica congruente com o referencial teórico presente na Figura 4.

5.6. Controlador PID

Utilizando os fatores K_p , K_i e K_d calculados para o sistema de controle PID, foram obtidos os dados de temperatura em função do tempo, dispostos na Figura 14. Foram utilizados para calcular as médias das temperaturas máximas e mínimas, a faixa de temperatura do controle e o sinal do *overshoot*, representados na Tabela 9.

Figura 14 - Aquecimento em triplicata de temperatura em função do tempo, do controlador PI para as temperaturas de *setpoint*.



Fonte: Autoral.

Tabela 9 - Valores médios do controle PID para as temperaturas máximas e mínimas, faixa de controle e *overshoot*.

PID				
Setup	Temperatura máxima média [°C]	Temperatura mínima [°C]	Faixa de controle [°C]	Overshoot [°C]
40°C	41.92 ± 0.11	39.42 ± 0.11	2.5 ± 0.11	1.58 ± 0.11
50°C	52.17 ± 0.11	49.08 ± 0.11	3.10 ± 0.11	1.33 ± 0.11
60°C	62.00 ± 0.27	59.00 ± 0.27	3.00 ± 0.27	1.42 ± 0.27
70°C	72.00 ± 0.27	69.17 ± 0.27	2.83 ± 0.27	1.17 ± 0.27
80°C	81.25 ± 0.31	79.00 ± 0.31	2.25 ± 0.31	0.67 ± 0.31

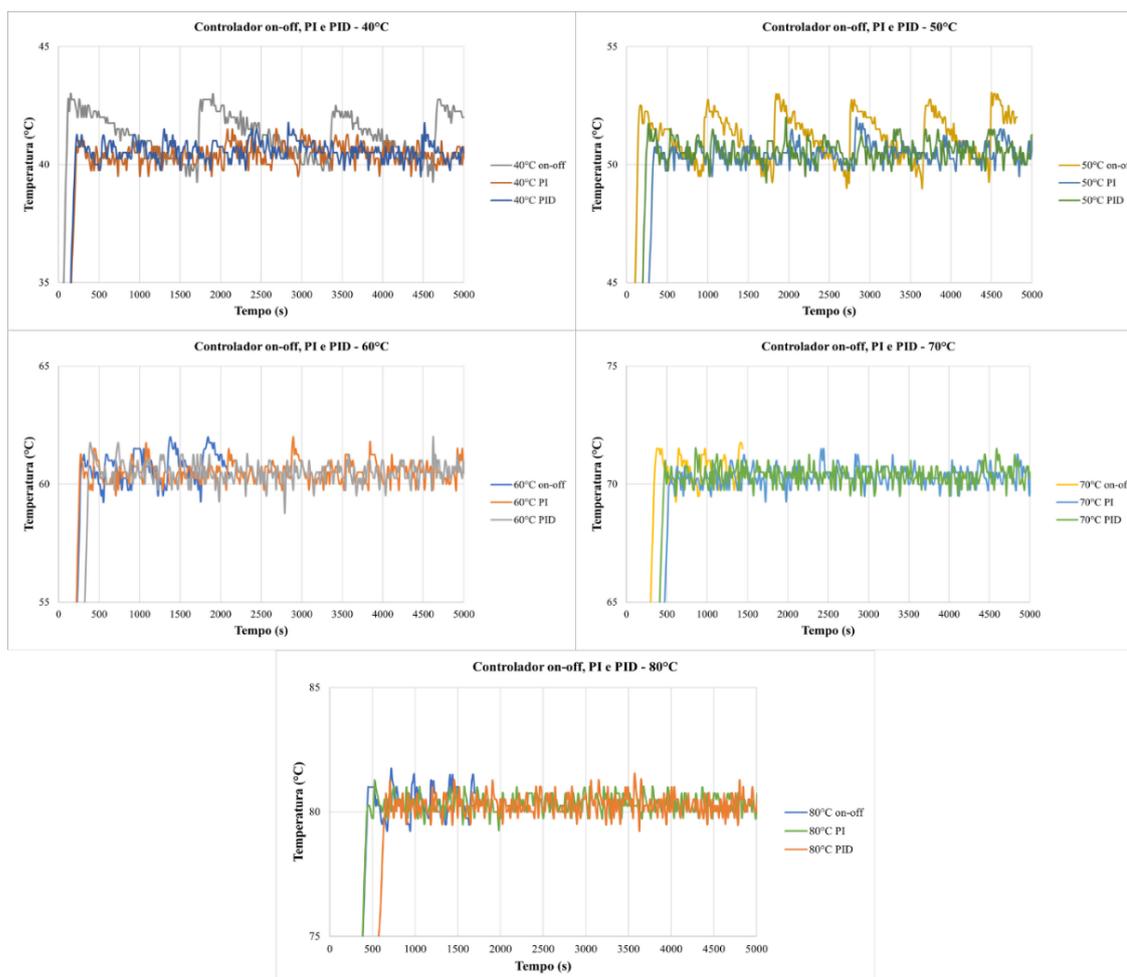
Os dados apresentados na Tabela 9 indicam uma faixa de controle máxima de $2.50 \pm 0.11^\circ\text{C}$, cerca de 6.25% da temperatura de referência de 40°C . O sinal de *overshoot* não correspondeu ao pico mais elevado do aquecimento.

Durante o controle PID, as curvas de aquecimento apresentaram um comportamento congruente ao referencial, presente na Figura 5 da revisão bibliográfica, em que retratou primeiramente o sinal de *overshoot* com a sequência do tempo de acomodação, podendo ou não conter um erro de estado estacionário em decorrência dos fatores atribuídos.

5.7. Controlador *on-off* zona morta, em comparação ao controlador PI e PID

Visto que as triplicatas apresentaram comportamentos sem discrepâncias expressivas, uma curva de aquecimento obtida por cada estratégia de controle para as temperaturas de *setpoint*, estão representadas na Figura 15 para fins de comparação.

Figura 15 - Dados do aquecimento dos controles *on-off* zona morta, PI e PID em 40°C e 50°C de *setpoint*.



Fonte: Autoral.

Notou-se pelas Tabelas 7, 8 e 9 e pela Figura 15 que para menores temperaturas, os sistemas de controle PI e PID demonstram menores flutuações após atingir o regime estacionário em relação ao comportamento observado no controlador *on-off*. A faixa de controle em 40°C para o PI e PID, respectivamente, teve uma diminuição de $1.17 \pm 0.11^\circ\text{C}$ e $1.67 \pm 0.11^\circ\text{C}$ em comparação ao *on-off* zona morta, ou seja, aproximadamente 3% e 4% de diferença em relação ao *setpoint*.

A menor diferença presente entre os controladores PID e *on-off* foi indicada na temperatura de 70°C com um valor de $0.09 \pm 0.27^\circ\text{C}$, aproximadamente 0.13% em relação ao *setpoint*, enquanto nos modos de controle PI e *on-off*, foi no *setpoint* de 60°C, apresentando um valor de $0.25 \pm 0.27^\circ\text{C}$, aproximadamente 0.4%. Isto indica que para altas temperaturas os máximos e

mínimos dos controladores foram relativamente próximos, enquanto para menores temperaturas foram distantes.

Ao contrapor os resultados dos diferentes modos de controle, pode-se observar que as curvas de aquecimento PI e PID em todas as temperaturas operaram com menores flutuações que o *on-off* zona morta, pois este é controlado por repetidos chaveamentos contendo um ápice e um ponto extremo de declínio, enquanto os controladores PI e PID, após passar pelo sinal de *overshoot*, entram em um estado de correção no qual tendem sempre à temperatura de *setpoint*.

Conclui-se que o método de controle PI e PID empregado nos experimentos realizados, demonstraram menores flutuações durante o regime estacionário em comparação com o *on-off* zona morta em todas as faixas de temperatura, mesmo que em altas temperaturas os valores máximos e mínimos estivessem próximos.

5.8. Comparação entre controlador PI e PID

Analisando as Tabelas 8 e 9, pode-se verificar que as temperaturas máximas, mínimas, faixa de controle e sinal de *overshoot* dos sistemas PI e PID, resultaram em valores próximos, no qual a diferença mais significativa está presente na faixa de controle de 40°C, em que o controle PI apresentou um valor de $3.0 \pm 0.11^\circ\text{C}$ e o PID resultou em $2.5 \pm 0.11^\circ\text{C}$, ou seja, a diferença entre ambos foi de 1.25% em relação ao *setpoint*.

Apesar dos valores próximos, o sistema de controle PI desenvolveu um sinal de *overshoot* levemente menor que o PID, de modo que a maior diferença foi no experimento de 60°C no valor de $0.42 \pm 0.27^\circ\text{C}$, ou seja 0.7% em relação ao *setpoint*.

Contudo, o controlador PID resultou em temperaturas máximas ligeiramente menores que o PI, como exemplo, a maior diferença obtida ocorreu durante o experimento de 40°C no valor de $0.41 \pm 0.27^\circ\text{C}$, ou seja, aproximadamente 1% em relação ao *setpoint*.

Outrossim, ao analisar visualmente a Figura 15, é possível determinar que o controlador PI e PID obtiveram curvas semelhantes, em que o estado estacionário demonstrou flutuações análogas.

Pode-se concluir então que os métodos de controle PI e PID, sob as condições citadas durante a metodologia, obtiveram resultados muito próximos sobre os sinais de *overshoot*, temperatura máxima e mínima, faixa de controle e o erro do estado estacionário.

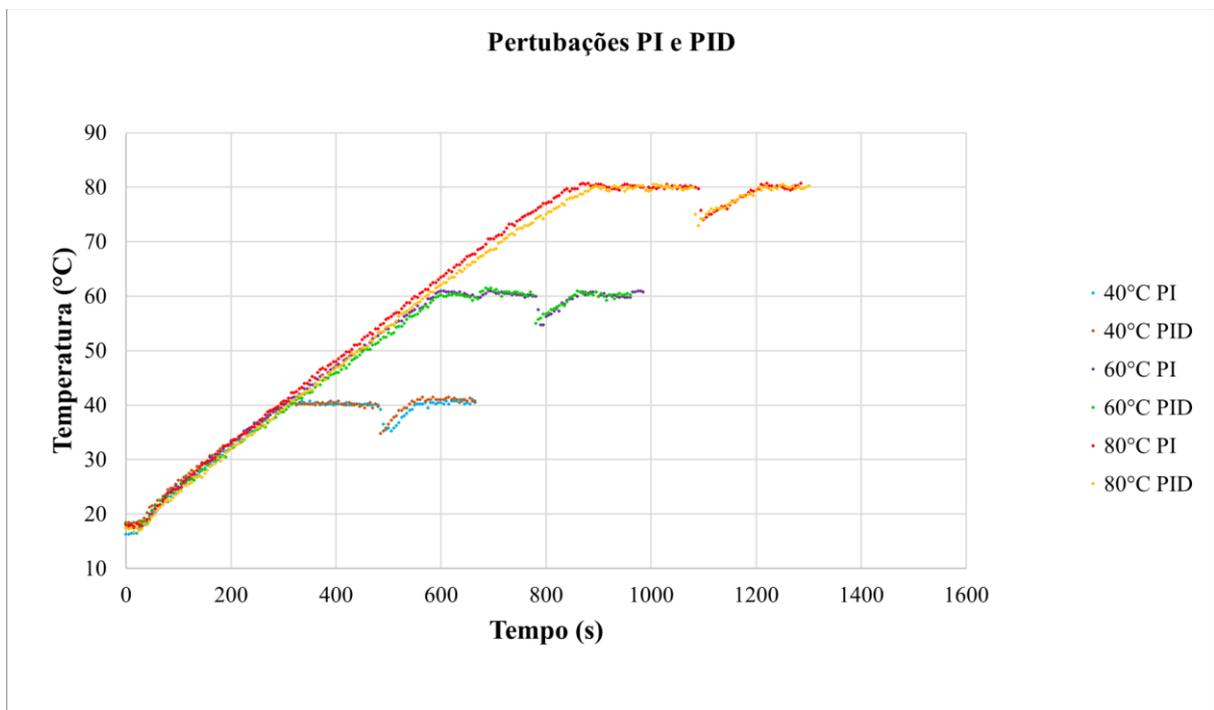
Não foi possível determinar com convicção qual o método de controle melhor aplicado durante os ensaios de aquecimento realizados, de modo que as pequenas flutuações já eram esperadas e demonstraram um controle de temperatura satisfatório.

5.9. Perturbação dos sistemas de controle PI e PID

Para concretizar o estudo de caso de um controlador de temperatura PI e PID, foi realizado um ensaio de perturbação, como descrito no procedimento experimental do mesmo.

A fim de comparação, foram dispostos os dados de temperatura por tempo de 3 ensaios de perturbação nas temperaturas de *setpoint* de 40°C, 60°C e 80°C, como na Figura 16, e os valores do tempo de acomodação e temperatura mínima na Tabela 10.

Figura 16 - Gráfico da perturbação nos sistemas de controle PI e PID nas temperaturas de *setpoint* de 40°C, 60°C e 80°C.



Fonte: Autoral.

Tabela 10 - Valores de tempo de ajuste e temperatura mínima durante o ensaio de perturbação.

Setpoint	PI [s]		PID [s]	
	Temperatura após perturbação [°C]	Tempo de acomodação [s]	Temperatura após perturbação [°C]	Tempo de acomodação [s]
40°C	35,5	81	35,66	61
60°C	55,08	72	54,91	71
80°C	73,66	118	73,5	108

Conforme os dados apresentados, os ensaios de perturbação se mantiveram congruentes de acordo com a seção de discussão da comparação dos controladores PI e PID. As temperaturas

mínimas foram análogas e os tempos médios de acomodação foram próximos, sendo a maior disparidade durante o *setpoint* de 40°C com 20 segundos de diferença.

Desse modo, foi possível determinar que os controladores agiram de forma semelhante durante o ensaio de perturbação realizado.

6. CONCLUSÃO

O protótipo foi desenvolvido com um valor de R\$ 127,00, em que é possível a sincronização com diferentes softwares de planilhas, tornando-o compatível com uma aula prática típica de um curso de engenharia sobre o fator de obtenção de dados, diferente de outros controladores da mesma faixa de valor.

A característica programável do protótipo foi alcançada com sucesso, possibilitando a realização de experimentações e ajustes de parâmetros conforme necessário. A programabilidade é um aspecto fundamental, pois permite explorar diferentes configurações e cenários, contribuindo para a compreensão dos conceitos abordados na disciplina de forma prática e interativa.

Outra vantagem significativa do protótipo desenvolvido é a sua capacidade de realizar experimentos de curta duração. Com um tempo estimado de 30 minutos por experimento, ele se adequa perfeitamente às limitações de tempo comuns em uma aula prática. Essa característica permite que os estudantes realizem múltiplos experimentos ao longo de uma única sessão, aprofundando sua compreensão e permitindo a análise de diferentes cenários e resultados.

O protótipo, portanto, pode ser aplicado em aulas práticas de engenharia, tendo como algumas sugestões iniciais, estudos sobre temas como ensaios de calibração de termopar, em que os alunos podem aprender sobre os princípios de funcionamento do sensor, além de compensar erros e realizar testes de precisão e linearidade em diferentes temperaturas.

Ademais, podem elaborar controladores *on-off* com ou sem zona morta, de modo que, os experimentos podem envolver o controle de um sistema simples, como um sistema de aquecimento ou resfriamento para observar o comportamento, analisar sua eficiência, estabilidade e comparar com outras arquiteturas.

Além da clássica estratégia de sintonização de controladores conhecida como Ziegler-Nichols, podem ser exploradas outras técnicas e algoritmos de sintonização, como o método de Cohen-Coon e o método de resposta ao degrau.

7. PRODUTO PIEPEX

O desenvolvimento do produto Piepex teve início durante o curso de introdução de sistemas embarcados oferecido pela LAPSE, laboratório de prototipagem de sistemas embarcados. Tevesse 80 inscritos dos quais 25 completaram a programação das aulas práticas e teóricas.

A Figura 17 e 18 retrata o acontecimento das aulas práticas do curso de introdução de sistemas embarcados.

Figura 17 - Curso da LAPSE de introdução a sistemas embarcados (2023)



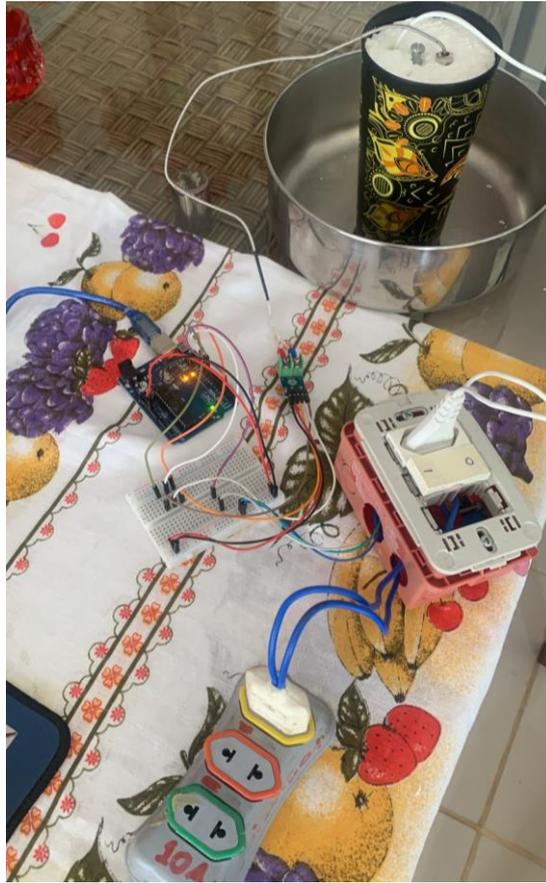
Figura 18 - Estudantes e organizadores do curso de sistemas embarcados da LAPSE (2023)



O protótipo foi realizado a partir de uma carência apresentada pela UNIFAL, na qual não dispõe de um material didático para o estudo de controladores, até o presente momento. Desse modo, o protótipo desenvolvido atende as necessidades do campus e de uma aula prática.

A Figura 19 representa o protótipo desenvolvido.

Figura 19 - Protótipo do controlador de temperatura



Contudo é necessário que seja realizado um curso de capacitação aos técnicos sobre o aparato desenvolvido.

8. REFERÊNCIAS

AGUIRRE, Luis Antonio. Fundamentos de instrumentação. Luis Antonio Aguirre, 2013.

Disponível em:

https://www.academia.edu/43835276/Fundamentos_de_Instrumenta%C3%A7%C3%A3o_AGUIRRE_2013. Acesso em: 03 jul 2023.

ALPI, L. et al. Controle PID: rompendo a barreira do tempo. 2019. Disponível em:

<https://www.novus.com.br/blog/artigo-controle-pid-rompendo-a-barreira-do-tempo/>. Acesso em: 05 jul 2023.

ALVES, Max Matias. Implementação de sistema pi para controle de temperatura utilizando o método da frequência de ziegler-nichols. Trabalho de Conclusão de Curso. Brasil. 2023.

Disponível em: <https://repository.ufrpe.br/handle/123456789/4613>. Acesso em: xx jul 2023.

CHASE, Otavio; ALMEIDA, F. Sistemas embarcados. Mídia Eletrônica. 2007. Disponível em:

http://www.maxpezzin.com.br/aulas/6_EAC_Sistemas_Embarcados/1_SE_Introducao.pdf.

Acesso em: 04 jul 2023.

COLOGNI, Mario Andrei, et al. Estudo e avaliação de metodologias de auto-sintonia de controladores pid visando uma implementação em controlador industrial. 2008. Disponível em:

<https://repositorio.ufsc.br/bitstream/handle/123456789/91562/261764.pdf?sequence=1&isAllowed=y>. Acesso em: 03 jul 2023.

FACCIN, Flávio. Abordagem inovadora no projeto de controladores PID. 2004. Disponível em:

<https://www.lume.ufrgs.br/bitstream/handle/10183/12544/000626885.pdf?sequence=1&isAllowed=y>. Acesso em: 05 jul 2023.

FERRANDO, Daniel Fonseca Corradini; ARAÚJO, John Welvins Barros de; KAKUNO, Edson Massayuki. Controle de temperatura PID com arduino. 2017. Disponível em:

<http://www.repositorio.jesuita.org.br/bitstream/handle/UNISINOS/8331/6873-10258-1-DR.pdf?sequence=1&isAllowed=y>. Acesso em: 08 jul 2023.

FUENTES, Rodrigo Cardoso. APOSTILA DE AUTOMAÇÃO INDUSTRIAL. 2005.

Disponível em: http://w3.ufsm.br/fuentes/index_arquivos/CA02.pdf. Acesso em: 01 jul 2023.

GARCIA, Claudio. Controle de processos industriais: estratégias convencionais. Editora

Blucher, 2021. Disponível em: <https://books.google.com.br/books?hl=pt->

<BR&lr=&id=fyxDwAAQBAJ&oi=fnd&pg=PA21&dq=controle+processos&ots=SwZe9xfT>

[80&sig=lCecb36ywTJSb4IT0PDCWfzv8_0&redir_esc=y#v=onepage&q=controle%20processos&f=false](#). Acesso em: 02 jul 2023.

LIMA, Michelle Silva et al. Controle de temperatura com arduíno. Revista Mythos, v. 12, n. 2, 2019. Disponível em:

<https://www.periodicos.unis.edu.br/index.php/mythos/article/view/308>. Acesso em: 06 jul 2023.

LOURENÇO, João. Sintonia de controladores PID. Escola superior de tecnologia, 1997.

Disponível em: https://scholar.google.pt/scholar?hl=pt-BR&as_sdt=0%2C5&q=LOUREN%C3%87O%2C+Jo%C3%A3o.+Sintonia+de+controladores+PID&btnG=. Acesso em: 05 jul 2023.

LUGÃO, João Ricardo Lhullier. Controlador de temperatura para célula de medição de propriedades de líquidos por ultrassom. 2015. Disponível em:

<https://repositorio.unesp.br/handle/11449/136279>. Acesso em: 04 jul 2023.

MARTINAZZO, Claodomir Antonio et al. Arduino: Uma tecnologia no ensino de física.

Revista Perspecfiva, 2014. Disponível em:

https://eadcampus.spo.ifsp.edu.br/pluginfile.php/938271/mod_resource/content/1/143_430.pdf. Acesso em: 04 jul 2023.

MATHEUS, Lucas das Neves. Sintonia de um controlador PID para sistema de aquecimento em batelada de um mini-tanque de uma microcervejaria. 2019. Disponível em:

<https://repositorio.animaeducacao.com.br/bitstream/ANIMA/4083/4/SINTONIA%20PID.pdf>.

Acesso em: 03 jul 2023.

MCROBERTS, Michael. Arduino básico. Novatec Editora, 2018. Disponível em:

https://edisciplinas.usp.br/pluginfile.php/4287597/mod_resource/content/2/Ardu%C3%ADno%20B%C3%A1sico%20-%20Michael%20McRoberts.pdf. Acesso em: 02 jul 2023.

OGATA, Katsuhiko; SEVERO, Bernardo. Engenharia de controle moderno. 5ª Edição. São Paulo: Pearson Education do Brasil, 2014. Disponível em:

https://www.academia.edu/45620848/ENGENHARIA_DE_CONTROLE_MODERNO_5a_E_DI%C3%87%C3%83O_OGATA. Acesso em: 09 jul 2023.

APÊNDICE

APÊNDICE A – Programação da calibração do termopar

```
//Inclusão de bibliotecas
#include "max6675.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
//definicão
#define endereco 0x27 // Endereços comuns: 0x27, 0x3F
#define colunas 16
#define linhas 2

int SO = 11;
int CS = 12;
int CLK = 13;
MAX6675 sensor(CLK, CS, SO); // ou MAX6675 sensor(SO, CS, CLK);

LiquidCrystal_I2C lcd(endereco, colunas, linhas);

void setup()
{
  Serial.begin(9600);
  lcd.init(); // INICIA A COMUNICAÇÃO COM O DISPLAY
  lcd.backlight(); // LIGA A ILUMINAÇÃO DO DISPLAY
  lcd.clear(); // LIMPA O DISPLAY
  lcd.print("Graus C = ");
  lcd.print(sensor.readCelsius());
  delay(5000);
}
void loop()
```

APÊNDICE B – Programação do controlador on-off com zona morta

```
#include "max6675.h"
#include <Wire.h>
#define RBQ 3 //Rabo quente

float Tsetup, Ls, Li;
int SO = 11;
int CS = 12;
int CLK = 13;
MAX6675 sensor(CLK, CS, SO); //ou MAX6675 sensor(SO, CS, CLK);

void setup() {
  Serial.begin(9600);
  pinMode(RBQ,OUTPUT);
  digitalWrite(RBQ,HIGH);
}

void loop() {
  Tsetup = X;
  Ls = Tsetup +0.5;
  Li = Tsetup -0.5;
  if (( sensor.readCelsius() < Li) ){
    digitalWrite(RBQ,LOW);
  }
  else if( sensor.readCelsius() > Ls){
    digitalWrite(RBQ,HIGH);
  }
  Serial.println(sensor.readCelsius());
  delay(5000);
}
```

APÊNDICE C – Programação do controlador PID

```

#include <PID_v1.h>
#include <max6675.h>

// Pinos da placa MAX6675
int thermoDO = 11;
int thermoCS = 12;
int thermoCLK = 13;

// Pino do relé de estado sólido
int relayPin = 5;

// Parâmetros PID
double Setpoint = X; // Temperatura de setup desejada em °C
double Kp = X; // Ganho proporcional
double Ki = X; // Ganho integral
double Kd = X; // Ganho derivativo

// Variáveis PID
double Input, Output;
PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);

// Objeto MAX6675
MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);

void setup()
{
  // Inicialização do Arduino
  pinMode(relayPin, OUTPUT);
  digitalWrite(relayPin, LOW);

  // Configuração da comunicação serial
  Serial.begin(9600);

  // Inicialização do PID
  myPID.SetMode(AUTOMATIC);
  myPID.SetOutputLimits(0, 255); // Limites de saída para o relé

```

```
}  
  
void loop()  
{  
  // Leitura da temperatura  
  Input = thermocouple.readCelsius();  
  
  // Cálculo do controle PID  
  myPID.Compute();  
  
  // Controle do relé  
  if (Input < Setpoint) {  
    digitalWrite(relayPin, HIGH); // Liga o relé  
  } else {  
    digitalWrite(relayPin, LOW); // Desliga o relé  
  }  
  
  // Saída de informações  
  Serial.print(Input);  
  Serial.print("\t");  
  Serial.println(Output);  
  
  delay(5000); // Intervalo de leitura e controle  
}
```