

# IMPLEMENTAÇÃO DE SISTEMA PI PARA CONTROLE DE TEMPERATURA UTILIZANDO O MÉTODO DA FREQUÊNCIA DE ZIEGLER-NICHOLS

## IMPLEMENTATION OF PI SYSTEM FOR TEMPERATURE CONTROL USING THE ZIEGLER-NICHOLS FREQUENCY METHOD

Max Matias Alves<sup>1</sup>  
Elias Marques Ferreira de Oliveira<sup>2</sup>

### RESUMO

Os controladores de processos industriais são cada vez mais necessários, e os mais utilizados são os controladores *on-off* e o PID (proporcional-integral-derivativo), que são amplamente utilizados em processos industriais. A proposta geral deste artigo é implementar um protótipo de um sistema de automação de controle de temperatura com PID, revisando conceitos e métodos de sintonia a fim de comparar a eficácia com o controlador *on-off*. O controle do tipo *on-off*, possui duas posições fixas, ligado ou desligado, já o PID corresponde com relação entre a variável lida e o *setpoint*. Existem vários métodos de sintonia desse tipo de controlador incluindo métodos empíricos até os mais refinados; mas geralmente envolve três etapas: perturbar o processo, analisar e quantificar a resposta do sistema ao distúrbio e, em seguida, sintonizar os parâmetros do controlador. O método da frequência proposto por Ziegler-Nichols consiste em determinar o ganho crítico e o período crítico para obter os parâmetros de sintonia. O aparato para os experimentos foi constituído por uma caixa térmica para simular o ambiente, com um sensor de temperatura em seu interior e uma lâmpada incandescente para gerar distúrbio da variável, uma ventoinha para insuflar ar no ambiente, e o sistema embarcado contendo um Arduino e uma placa de circuito elétrico. Aplicando o método proposto para os experimentos, obtivemos os resultados para os controles P, PI, PID e o *on-off*, ao analisar esses resultados, verificou-se que para tais condições, o melhor controlador foi o do tipo PI.

**Palavras-chave:** PID; *on-off*; Ziegler-Nichols; controlador.

### ABSTRACT

Industrial process controllers are increasingly needed, and the most used are the on-off controllers and the PID (proportional-integral-derivative), which are widely used in industrial processes. The general purpose of this article is to implement a prototype of a temperature control automation system with PID, reviewing tuning concepts and methods in order to compare the effectiveness with the on-off controller. The on-off type control has two fixed positions, on or off, since the PID corresponds to the relationship between the read variable and the setpoint. There are several methods of tuning this type of controller including empirical methods to the most refined; but that usually involves three steps: disturbing the process, analyzing and quantifying the system's response to the disturbance, and then tuning the controller parameters. The frequency method proposed by Ziegler-Nichols consists of determining the critical gain and the critical period to obtain the tuning parameters. The apparatus for the experiments consisted of a thermal box to simulate the environment, with a temperature sensor inside and an incandescent lamp to generate disturbance of the variable, a fan to blow air into the environment, and the embedded system containing an Arduino and an

---

<sup>1</sup> Bacharelado em Engenharia Mecânica pela Universidade Federal Rural de Pernambuco - Unidade Acadêmica do Cabo de Santo Agostinho, 2023.

<sup>2</sup> Doutor na Universidade Federal Rural de Pernambuco - Unidade Acadêmica do Cabo de Santo Agostinho, 2018.

electrical circuit board. Applying the proposed method to the experiments, we obtained the results for the P, PI, PID and on-off controls. When analyzing these results, it was verified that for such conditions, the best controller was the PI type.

**Keywords:** PID, *on-off*; Ziegler-Nichols; controller.

## INTRODUÇÃO

Nas últimas décadas pode-se observar que cada vez mais a implementação de novas máquinas e computadores aumentaram significativamente em todos os processos, especialmente nos meios industriais. Em meados do século passado, esse fato foi comprovado a partir do início da implementação de equipamentos voltados para o controle dos processos (MIYAGI, 1996). O principal objetivo de utilizar esses dispositivos para controle e automação industrial é otimizar a produção, ter uma maior produtividade e flexibilidade dos sistemas. Essas máquinas eletrônicas e computadores dedicados para o controle e automação industrial são denominados controladores industriais.

Uma das ferramentas mais conhecidas para controle é a ação de controle Proporcional-Integral-Derivativo. Um controlador PID é capaz de uma estabilidade de controle excepcional quando ajustado e usado adequadamente. Em Rubaai *et al* (2008) é citado que 96% dos controladores utilizados são do tipo PID. Mais de 90% das malhas de controle encontradas em processos industriais operam com controladores PI/PID atingindo uma larga faixa de aplicações: controle de processos, *drivers* para motores, indústria automobilística, controladores de voo, pilotos automáticos, instrumentação, entre outros (ÅSTRÖM; HÄGGLUND, 1995). Essa ação de controle está presente em praticamente todo processo de automação que precise de controle de malha fechada e necessite de ajuste de parâmetros a fim de que a mesma funcione de forma adequada.

Para entender a vasta aplicabilidade do PID, além de processos industriais já citados, ele também tem sido estudado para aumentar o fator de potência da geração de energia eólica, a partir de um controle de otimização dos parâmetros PID, de forma on-line (LI, LI, 2020).

O controlador PID, com o passar dos anos passou por evoluções, sendo desenvolvidas diversas variações. Sendo algumas delas: *Firefly*-PID, controlador PID baseado em ensino e aprendizagem; controlador PID de otimização por enxame de partículas; e um dos mais recentes é o controlador DO-PID (PID de otimização *dandelion*), que recentemente foi testado com sucesso para um controle eficiente para velocidade de veículos elétricos, com propósito de diminuir o erro entre os sinais de referência e de saída (BAIDYA, DHOPTÉ, BHATTACHARJEE, 2023).

Outro também recente é o controlador PID de ordem fracionária (HSU, 2019), que tem sido estudado na estabilização do efeito das correntes e fluxos magnéticos gerados pelos parâmetros dos PID em motores de indução, com o objetivo de reduzir a vibração e o ruído (HSU, 2019).

Existem inúmeras técnicas de ajuste de parâmetros PID (OGATA, 2011) (ZIEGLER NICHOLS, 1942), que vão desde técnicas mais robustas de ajuste, como análises de diagrama de estabilidade, até técnicas empíricas de resposta a salto ou método das frequências de Ziegler-Nichols ou até mesmo simplesmente tentativa e erro, que, apesar de não oferecer a melhor acurácia, é a uma das mais utilizada. Mesmo com seu grande uso, e possuindo um vasto histórico e bibliografia, o ajuste de controladores PID ainda é uma área ativa de pesquisa, tanto acadêmica quanto industrial (CONG; LIANG, 2009).

Diante desse contexto, neste trabalho propõe-se implementar um protótipo de um sistema de automação de controle de temperatura com lógica PID, buscando revisar os conceitos de sistemas de automação e controle de temperatura com foco em lógica PID; estudar a funcionalidade de instrumentos de controle (sensores, transdutores, controladores, atuadores); projetar o encapsulamento e lógica de funcionamento do sistema; construir o protótipo; atestar a funcionabilidade e eficácia do protótipo comparando o controle *on-off* com o PID pelo método de Ziegler-Nichols.

## FUNDAMENTAÇÃO TEÓRICA

### Controladores automáticos

Um controlador automático compara o valor real de saída de um processo, por meio de um sensor com o valor do *setpoint*, que é o valor da variável de processo que se deseja alcançar. Assim determina-se o desvio chamado de erro, e produz-se um sinal de controle, comumente referido como *output*, que tem o propósito de reduzir esse erro a zero ou a um valor pequeno. Esse sinal de *output* alimenta um atuador, como um motor elétrico, hidráulico, ou uma válvula (OGATA, 2011).

Basicamente temos dois tipos de controladores automáticos, são eles: controlador *on-off* e o PID.

### Controlador on-off

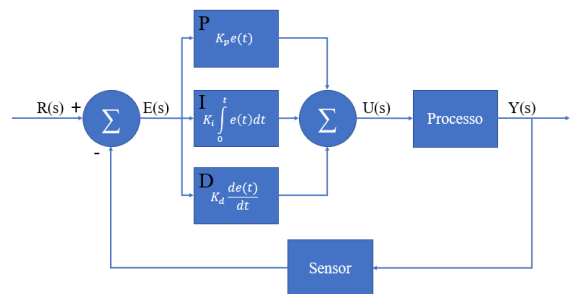
O controlador *on-off* é um sistema de controle de duas posições, cujo sinal de *output* fornece ao atuador somente duas posições fixas, que são, ligado ou desligado. Esse é um controle relativamente simples e de fácil implementação. Por essa razão, é bastante utilizado em sistemas de controle domésticos e industriais quando não é necessário um controle com alta precisão da variável a ser controlada dentro de um processo (OGATA, 2011).

### Controlador PID

O controlador Proporcional-Integral-Derivativo (PID) é de longe o algoritmo de controle mais comum. A maioria dos processos com controle de realimentação são controlados por esse algoritmo ou por pequenas variações dele. Ele é implementado de muitas formas diferentes, como um controlador autônomo ou como parte de um pacote de Controle Digital Direto, do inglês *Direct Digital Control* (DDC) um sistema de controle de processo distribuído hierárquico (ÅSTRÖM; HÄGGLUND, 1995). Ele atua com ações de controle proporcional (P), integrativo (I) e derivativo (D).

Um controlador PID visa a modificação de variáveis do sistema, do qual se quer um determinado controle modificando a relação entre entrada e saída de um processo, fazendo leitura da variável controlada, por meio de um sensor, e mudando seu valor por meio de atuador que manipula essa variável quando se há um erro determinado pela diferença entre o *setpoint* e o valor da variável controlada (ARAÚJO, 2007). No diagrama de bloco do controlador PID dentro de um processo genérico da figura 1, podemos observar os sinais:  $R(s)$ , valor de *setpoint*, depois  $E(s)$ , sinal de erro; posteriormente  $U(s)$ , somatório do PID que vai para o atuador do sistema; e  $Y(s)$ , valor da variável controlada do processo.

Figura 1 – Diagrama de blocos do controlador PID



Fonte: Adaptado de ÅSTRÖM; HÄGGLUND, 1995.

O controlador proporcional pode atuar de forma unitária. Comumente, é

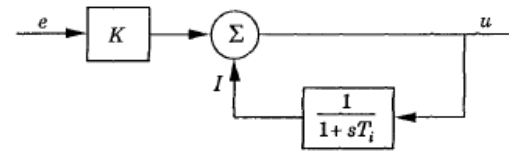
aplicado em processos simples que não demandam de um controle rigoroso objetivando corrigir o sistema de controle. A sua ação depende apenas da diferença entre o valor do *setpoint* e o valor da variável de processo, denominado de erro. Então o ganho proporcional ( $K_p$ ) determina então a relação entre a resposta de saída e o sinal de erro. Tomando como exemplo um erro de magnitude de 10, um ganho proporcional de 2 irá produzir uma resposta proporcional de 20.

No geral, o ganho proporcional está ligado à velocidade de resposta do sistema, no qual seu aumento ocasiona uma velocidade de resposta maior. Se for aplicado um ganho mais elevado a variável de processo passará a oscilar e deixar o sistema instável (NI, 2022). Entretanto, quando o processo está em regime permanente, o controle proporcional apresenta um erro residual (*offset*), ou seja, o erro não é completamente anulado.

A principal função da ação integral é assegurar que, em regime permanente, a saída do processo esteja de acordo com o *setpoint*, ou seja, com erro nulo (ÅSTRÖM; HÄGGLUND,1995). Ela atua fornecendo uma saída do controlador proporcional a integral do erro acumulado, que por muitas vezes causa uma reação lenta. No entanto, o controle integral, isoladamente, gera instabilidade ao processo em que o controlador é saturado, e não o conduz para o erro nulo, também chamado de *wind-up* (GOODWIN; GRAEBE; SALGADO, 2000). Por esse motivo, raramente é visto um controlador integral puro.

Originalmente, a ação integral era conhecida como "reinicialização automática", a qual tinha a função de realizar um *reset* automático. Esse sinal era inserido na realimentação e filtrado a partir da saída. Em seguida, ele era enviado para o ponto de soma do controlador para a ação proporcional, permitindo que o erro residual fosse redefinido até que se tornasse nulo. (ÅSTRÖM; HÄGGLUND,1995). É possível ver isso no diagrama de blocos da figura 2 a seguir.

Figura 2 – Ação proporcional como *reset* automático



Fonte: ÅSTRÖM; HÄGGLUND,1995

O controle derivativo é proporcional a taxa de variação do erro de controle e não atua a erros constantes. Sua ação faz com que a resposta de saída diminua se a variável de processo estiver aumentando rapidamente. Frequentemente, sua dependência em relação à tendência de erro o faz ser chamado de modo preditivo. (GOODWIN; GRAEBE; SALGADO, 2000). Aumentar o ganho derivativo ( $K_d$ ) acarretará ao sistema de controle reagir com mais agressividade às mudanças no erro. Logo, aumenta-se a velocidade de resposta geral do sistema. Por isso, comumente, os sistemas de controle reais usam um ganho derivativo pequeno, pois a resposta derivativa é bastante sensível ao ruído no sinal da variável do processo (tanto do sensor ou se a taxa de variação for muito lenta), tornando um sistema instável (NI, 2022).

Então, o controlador PID, que possui os três termos recém citados, relaciona as características de cada uma das ações, cujo controle proporcional atua para alcançar o *setpoint* minimizando o erro, a ação integral promove o erro nulo ao chegar no regime permanente e por sua vez, a ação de controle derivativo antecipa a resposta do sistema, aumenta a velocidade e a estabilidade do controle.

O controle PID de forma didática pode ser descrito como na equação 1:

$$u(t) = K_p(e(t) + \frac{1}{T_i} \int_0^t e(t)dt + \dots + T_d \frac{de(t)}{dt}) \quad (1)$$

Também podemos representá-lo pela função de transferência de acordo com a equação 2:

$$G(s) = K_p \left( 1 + \frac{1}{sT_i} + sT_d \right) \quad (2)$$

Porém nas atividades práticas, a forma amplamente utilizada é de acordo com a equação (3), por meio da qual há um melhor desempenho devido aos seus parâmetros serem de fácil percepção e de serem atribuídos aos processos utilizando parâmetros de ganho  $K_p$ ,  $K_i$  e  $K_d$ , que são respectivamente o ganho proporcional, integral e derivativo.

$$G(s) = K_p + \frac{K_i}{s} + sK_d \quad (3)$$

Os ganhos  $K_i$  e  $K_d$  estão relacionados de acordo com as equações (4) e (5).

$$K_i = \frac{K_p}{T_i} \quad (4)$$

$$K_d = K_p T_d \quad (5)$$

Também é possível combinar os controladores em PI (proporcional-integral) e PD (proporcional-derivativo), além de se poder usar apenas o controlador P. O controle PI soma da ação proporcional à ação integral, gerando um controle que atua para eliminar o *offset* quando o processo se encontra em regime permanente. Já o controle PD age a fim de reduzir o tempo de resposta e de aumentar a estabilidade do sistema.

Para obter esses controladores modificados, basta somente atribuir o valor zero ao ganho da ação que não deseja. Por exemplo: no controlador P, atribui-se  $K_i = K_d = 0$ ; já no caso de um controlador PI, considerar o valor de  $K_d = 0$ ; e caso queira obter um controlador PD, aplicar o valor de  $K_i = 0$ .

## Sintonia do controlador PID

Segundo Vandoren (2006), o princípio da sintonia de um controlador PID é identificar como a dinâmica do processo se comporta quanto a ação de controle e, se baseando no desempenho, definir os ganhos necessários para eliminar os erros. Existem vários métodos de sintonia de controladores PID, por exemplo análises de diagrama de estabilidade, métodos empíricos, como o método da resposta ao degrau de Cohen-Coon ou o método das frequências de Ziegler-Nichols, e até mesmo a abordagem de tentativa e erro. Mas independentemente da metodologia utilizada, há em comum três passos para identificar a dinâmica do processo: submeter o processo a perturbações; analisar e quantificar a resposta do sistema ao distúrbio; e sintonizar os parâmetros do controlador (JOHNSON & MORADI, 2005).

Seguindo o método da resposta em frequência proposto por Ziegler-Nichols, exemplificado por ÅSTRÖM e HÄGGLUND (1995), para obter os parâmetros de um controlador PID com sistema em malha fechada, primeiramente é determinado o ganho crítico ( $K_u$ ) obtido aumentando o ganho proporcional gradativamente até o valor de ganho ser capaz de gerar uma resposta oscilatória de amplitude constante e, conseqüentemente é determinado o período crítico ( $T_u$ ), que é o período da oscilação estável. De posse dos valores de  $K_u$  e  $T_u$ , pode-se obter os parâmetros para o controlador pela tabela 1 proposta por Ziegler e Nichols baseada no critério de desempenho a razão de decaimento de  $\frac{1}{4}$ . Esse método foi projetado a fim de fornecer boas respostas para distúrbios dos processos, obtidos por inúmeras simulações em sistemas diferentes. Por esse motivo, frequentemente, faz-se necessário realizar um ajuste fino nos ganhos do controlador (ÅSTRÖM e HÄGGLUND 1995).

Tabela 1 – Parâmetros de controlador PID do método da frequência por Ziegler Nichols

Controlador	$K_p$	$T_i$	$T_d$
P	$0,5K_u$		
PI	$0,4K_u$	$0,8T_u$	
PID	$0,6K_u$	$0,5T_u$	$0,125T_u$

Fonte: ÅSTRÖM; HÄGGLUND,1995

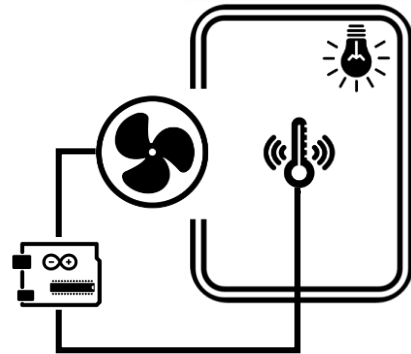
A relação dos fatores da tabela 1 pode ser calculada usando os parâmetros de sintonia  $K_p$ ,  $T_i$  e  $T_d$  para obter os valores de ganho  $K_i$  e  $K_d$  como demonstrado nas equações 4 e 5 já citadas.

Dessa forma, obtêm-se os ganhos proporcional, integral e derivativo ( $K_p$ ,  $K_i$  e  $K_d$ ), para serem implementados no sistema PID.

## MATERIAIS E MÉTODOS

Para desenvolver o experimento deste trabalho, foi implementado um aparato simulando um ambiente contendo uma lâmpada incandescente de 25W ligada diretamente à tomada elétrica a fim de gerar o distúrbio de temperatura no mesmo; um sensor de temperatura, um motor de corrente contínua com uma hélice acoplada em seu eixo como atuador e como dispositivo de controle; um sistema embarcado constituído por um Arduino e uma placa de circuito elétrico construída para abarcar os dispositivos e componentes necessários para seu funcionamento; uma fonte de alimentação externa para fornecer energia para o atuador, como observa-se nas figuras 3 e 4 com o arranjo experimental e a imagem do aparato.

Figura 3 – Arranjo experimental.



Fonte: O autor, 2023.

Figura 4 – Aparato do experimento.



Fonte: O autor, 2023.

### Ambiente

Para simular o ambiente a ser controlado foi utilizado uma caixa térmica de poliestireno (EPS) sem a tampa superior, e o sensor foi instalado no centro da parte inferior. Para receber o dispositivo atuador, foi aberto um orifício em uma das laterais, assim tendo uma área de entrada e saída para o fluxo de ar.

### Sensor de temperatura

No experimento o sensor de temperatura utilizado no experimento foi o DS18B20. Esse é um sensor de temperatura digital que dentro do seu invólucro possui circuitos de leitura de temperatura, conversor analógico/digital, memória para transmitir os valores lidos no formato digital. Ele é passível de inúmeras aplicações, como para controle termostático, sistemas industriais e até mesmo residenciais. Existem diferentes encapsulamentos com diferentes configurações de pinos do DS18B20. Neste trabalho, foi utilizado o modelo com

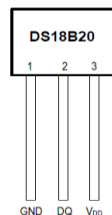
capsula metálica blindada (a prova d'água), como pode-se ver na figura 5, possuindo três terminais, sendo o terminal 1 o GND, o terminal 2 para transmitir os dados e o terceiro terminal é a alimentação. Esse sensor é semelhante ao de encapsulamento TO-92, o qual é idêntico a um transistor conforme observa-se na figura 6 (MAXIM INTEGRATED PRODUCTS, 2019).

Figura 5 – Sensor DS18B20 blindado.



Fonte: O autor, 2023.

Figura 6 – Modelo sensor DS18B20 em encapsulamento TO-92.



Fonte: MAXIM INTEGRATED PRODUCTS, 2019.

O DS18B20 possui escala graduada em graus Celsius, com *range* de temperatura de  $-55$  a  $125^{\circ}\text{C}$  e uma precisão de  $\pm 0,5^{\circ}\text{C}$  quando trabalhado com temperaturas entre  $-10$  e  $85^{\circ}\text{C}$ , com resolução programável de 9 a 12 bits correspondentes a incrementos de temperatura de  $0,5^{\circ}\text{C}$  a  $0,0625^{\circ}\text{C}$ . Sua alimentação funciona de 3 a 5,5V com corrente de 1,5mA, e necessita de um resistor de pull-up. Cada dispositivo tem um código serial exclusivo de 64 bits, e funciona com a tecnologia *1-Wire*<sup>®</sup> (MAXIM INTEGRATED PRODUCTS, 2019). O *1-Wire*<sup>®</sup> - marca registrada da Maxim Integrated Products - é um protocolo de comunicação serial que usa uma única linha de dados com a referência de um GND. Essa comunicação digital se dá

entre uma unidade computacional atuando como mestre e dispositivos de mesmo protocolo como sensores e adaptadores atuando como escravos, ou seja, gerenciado pelo mestre (MAXIM INTEGRATED PRODUCTS, 2013).

### Sistema embarcado

O sistema embarcado contém placa Arduino, placa de circuito elétrica montada conforme a necessidade dos instrumentos utilizados, uma fonte de 7,5V e 1A, como observa-se na figura 7. Também foi utilizado um computador para realizar a programação do Arduino e as leituras do experimento.

Figura 7 – Sistema embarcado.



Fonte: O autor, 2023.

### Arduino

Arduino é uma plataforma de hardware baseada em microcontroladores conectados a periféricos que permite uma programação de fácil linguagem proporcionando interação com o ambiente e atuadores. Há diversas versões de Arduino disponíveis construídos com diferentes periféricos e suas características de pinagem, capacidade de processamento, memória e outros. Algumas das versões mais conhecidas são o Arduino Uno, Mega 2560, Leonardo, pertencentes a família clássica, tem a família Nano, com as versões 33 IoT, RP2040 e outros, como também a família MKR, com modelos 1000 WiFi, FOX 1200 e vários outros (ARDUINO CC, 2023). A versão Arduino utilizada foi o Mega 2560.

Segundo Stevan e Silva (2015), a placa do Arduino MEGA 2560 é constituída pelo microcontrolador ATMEL ATMEGA2560 de 8 bits, memória de 256KB e cristal oscilador de 16MHz. Ela



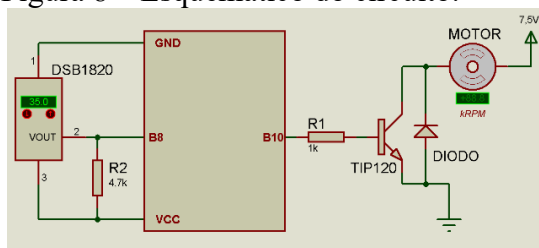
dispõe de dezesseis entradas analógicas, cinquenta e quatro pinos de entradas e saídas digitais, das quais doze podem ser utilizadas como saídas PWM (modulação por largura de pulso), e quatro portas de comunicação serial. A tensão de operação é de 5V, sua alimentação é via porta serial USB ou conexão específica que suporta uma tensão de 7 a 12V.

Para programar e interagir com placas de Arduino, tem-se a ferramenta Arduino IDE, disponibilizada gratuitamente pela própria plataforma. Esse *software* utiliza uma linguagem de programação baseada em C (STEVAN; SILVA, 2015).

### Placa de circuito

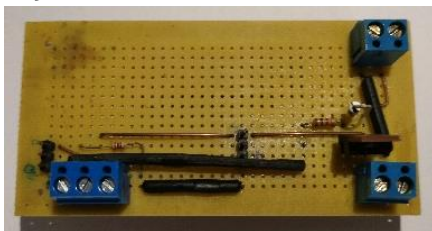
O circuito elétrico para receber o sensor e o atuador foi construído em uma placa padrão tipo ilha, contendo componentes como resistores, transistor, diodo e bornes para conexão dos fios elétricos dos dispositivos periféricos. Nas figuras 8 e 9 encontram-se o esquemático do circuito e a placa de circuito construída respectivamente.

Figura 8 – Esquemático do circuito.



Fonte: O autor, 2023.

Figura 9 – Placa de circuito construída.



Fonte: O autor, 2023.

No circuito do sensor, um resistor de *pull-up* é utilizado entre seu terminal de

saída e o VCC, para garantir que a entrada seja lida como alta pelo controlador. Já no circuito de acionamento do motor, um transistor é usado como chave eletrônica, para permitir que a energia da fonte externa seja drenada e o motor seja acionado com a tensão e a corrente necessárias. Além disso, um resistor é adicionado entre a porta de saída do Arduino e a base do transistor, a fim de limitar a corrente que flui entre eles e garantir que a placa não seja danificada pelo excesso de corrente.

### Dispositivo atuador

O dispositivo em questão é constituído por um motor elétrico, que aciona uma hélice metálica formando uma ventoinha, apresentado na figura 10, operando como um atuador eletromecânico. Esse dispositivo é capaz de converter energia elétrica em energia mecânica de rotação, aumentando a pressão do ar e gerando um fluxo de massa de ar. A hélice e o motor foram acoplados em um duto de PVC, funcionando como suporte e direcionador do fluxo de ar, conforme ilustrado na figura 11. Esse dispositivo foi utilizado como um ventilador durante o experimento, insuflando ar no ambiente.

Figura 10 – Atuador motor-hélice.



Fonte: O autor, 2023.

Figura 11 – Conjunto do atuador no tubo PVC.



Fonte: O autor, 2023.

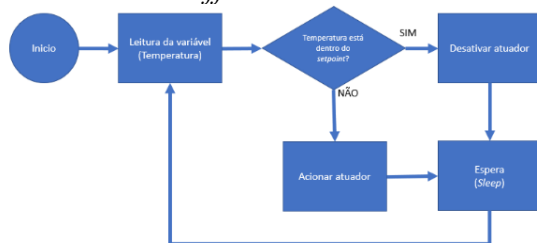


O motor elétrico utilizado funciona com alimentação DC (corrente contínua), tensão de 7,5V e corrente de 1A. Esse tipo de motor permite a aplicação de PWM para controlar sua velocidade, a qual foi modulado pelo Arduino utilizando o controlador PID. Essa técnica de controle de velocidade é fundamental para regular o fluxo de ar gerado pelo dispositivo de forma precisa e eficiente.

### Execução do experimento

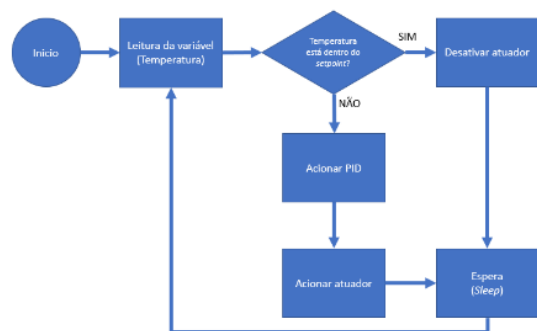
Para preparar a execução dos experimentos, foram criados dois fluxogramas dos algoritmos a serem elaborados, sendo um para o controle *on-off* e um para o controle PID, conforme as figuras 12 e 13. Esses fluxogramas serviram de base para a criação da programação de ambos os controles e assim escrevê-las de tal maneira que correspondesse ao propósito do experimento.

Figura 12 – Fluxograma da programação do controle *on-off*.



Fonte: O autor, 2023.

Figura 13 – Fluxograma da programação do controle PID.



Fonte: O autor, 2023.

Os algoritmos para os experimentos foram desenvolvidos por meio da IDE Arduino, com auxílio das

bibliotecas para utilização do sensor de temperatura DS18B20, o 1-Wire®, disponíveis na plataforma online do Arduino, onde existem também trabalhos e experimentos utilizando suas diversas placas controladoras. Foram escritos de acordo com as necessidades dos processos e da saída de monitoramento de dados para que assim fossem trabalhados com leituras de intervalo de 1s (um segundo). Os dados foram retirados do monitor serial da IDE Arduino foram: tempo, temperatura do sensor, e valor do sinal de *output*. Após cada rodada de teste, essas informações coletadas foram migradas e tratadas por um programa de planilha eletrônica, para obtenção de valores e gráficos a serem analisados.

Durante os testes, a temperatura do ambiente externo estava entre 29 e 29,5°C. E estando a lâmpada incandescente ligada no interior do ambiente simulado, a temperatura inicial foi de 37°C constante em cada experimento. O *setpoint* configurado foi de 35°C tanto para os controles PID quanto *on-off*, ou seja, o experimento foi executado em resfriamento do ambiente simulado.

Para escrever o algoritmo do PID, foi utilizado a biblioteca PID, disponível na plataforma Arduino. Ela trabalha com variáveis definidas de entrada (*input*), saída (*output*), *setpoint* e os ganhos  $K_p$ ,  $K_i$  e  $K_d$ . A entrada é o valor obtido pelo sensor, a saída é o sinal emitido pelo controle PID após computar seus cálculos, e os demais dados são definidos pelo usuário.

Para implementar o controlador PID conforme o método da frequência de Ziegler-Nichols, incrementou-se valores de  $K_p$  no algoritmo a cada ciclo de testes de 500s, até ser obtido o valor de  $K_u$  e conseqüentemente  $T_u$ . De posse desses valores, calculou-se os valores de  $K_p$ ,  $T_i$  e  $T_d$  de acordo com a tabela 1, em seguida calculou-se os ganhos  $K_i$  e  $K_d$ , com as equações 4 e 5. Posteriormente os ganhos foram aplicados às variações do controle, sendo eles o P, PI e o PID.

Já a construção do algoritmo para o controle *on-off* foi trabalhada com funções

condicionais do Arduino, para que se tomasse forma a funcionalidade do sistema, e fosse possível executar os experimentos. O *setpoint* também foi configurado para 35°C. E foi implementado uma histerese de meio grau celsius tanto para mais quanto para menos no acionamento do atuador, a fim de que não houvesse um descontrole no acionamento do motor.

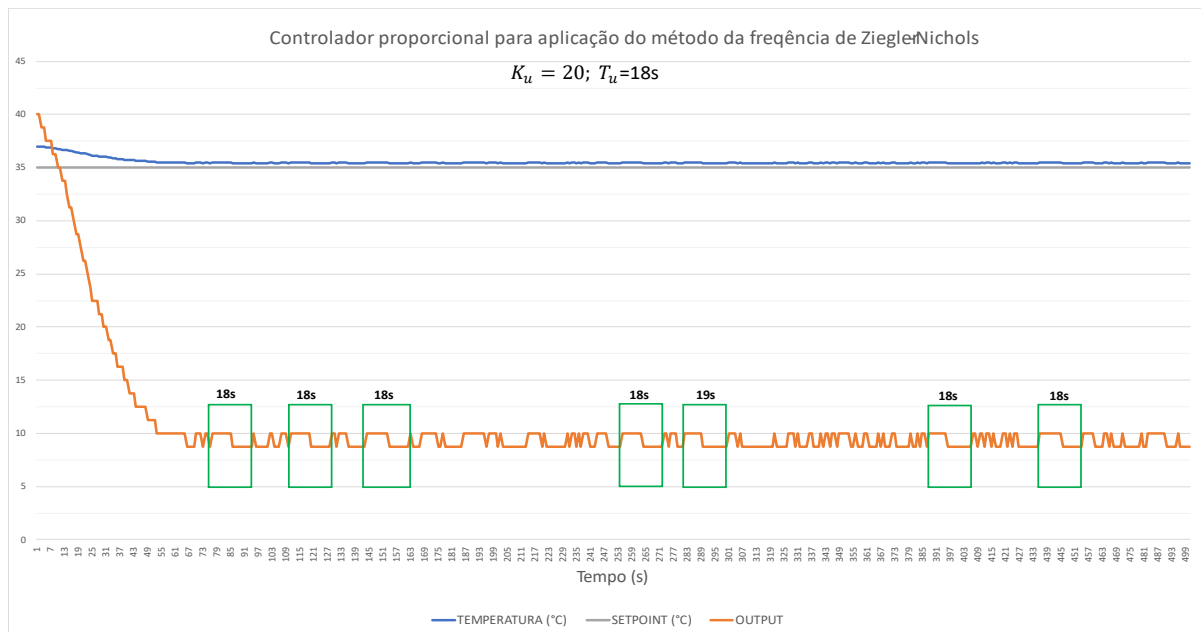
Com as aplicações e testes executados, partiu-se para a fase da análise de seus comportamentos e assim atestar qual o tipo de controle é mais eficiente para controlar a temperatura com as condições do ambiente simulado deste experimento. Dentre eles os controladores P, PI, PID ou *on-off*.

## RESULTADOS E DISCUSSÕES

Com os resultados obtidos dos testes iniciais com incrementos do ganho  $K_p$ , encontramos um  $K_u = 20$ . Esse valor foi selecionado, por ser o menor valor de  $K_p$  aplicado onde o sinal de *output* se estabilizou e se comportou com um padrão de oscilação. É importante salientar que como a temperatura é uma variável de resposta lenta, não se conseguiu observar consideráveis oscilações, mesmo quando os valores de  $K_p$  foram elevados.

É possível constatar na figura 14 o gráfico do experimento com o  $K_p = 20$ , e os padrões de oscilação na curva com períodos de 18s. Com isso, tomamos então que  $K_u = 20$  e  $T_u = 18$ .

Figura 14 – Gráfico do controlador proporcional para aplicação do método da frequência de Ziegler-Nichols



Fonte: elaborado pelo autor

De posse desses dados, calculamos  $K_p$ ,  $T_i$  e  $T_d$ , de acordo com a tabela 1, como podemos observar seus valores na seguinte tabela 2.

Tabela 2 – Parâmetros de controlador PID do método da frequência por Ziegler-Nichols.

Controlador	$K_p$	$T_i$	$T_d$
P	10		
PI	8	14,4	
PID	12	9	2,25

Fonte: O autor, 2023.

E logo após, temos a tabela 3 com os ganhos de  $K_p$ ,  $K_i$  e  $K_d$  segundo as equações 4 e 5.

Tabela 3 – Ganhos do controlador PID pelo método da frequência por Ziegler-Nichols.

Controlador	$K_p$	$K_i$	$K_d$
P	10		
PI	8	0,55	
PID	12	1,33	27

Fonte: O autor, 2023.

De posse desses valores de ganhos, foram executados o teste de cada controlador – P, PI e PID – obtivemos os dados e seus gráficos plotados como podemos ver na figura 15.

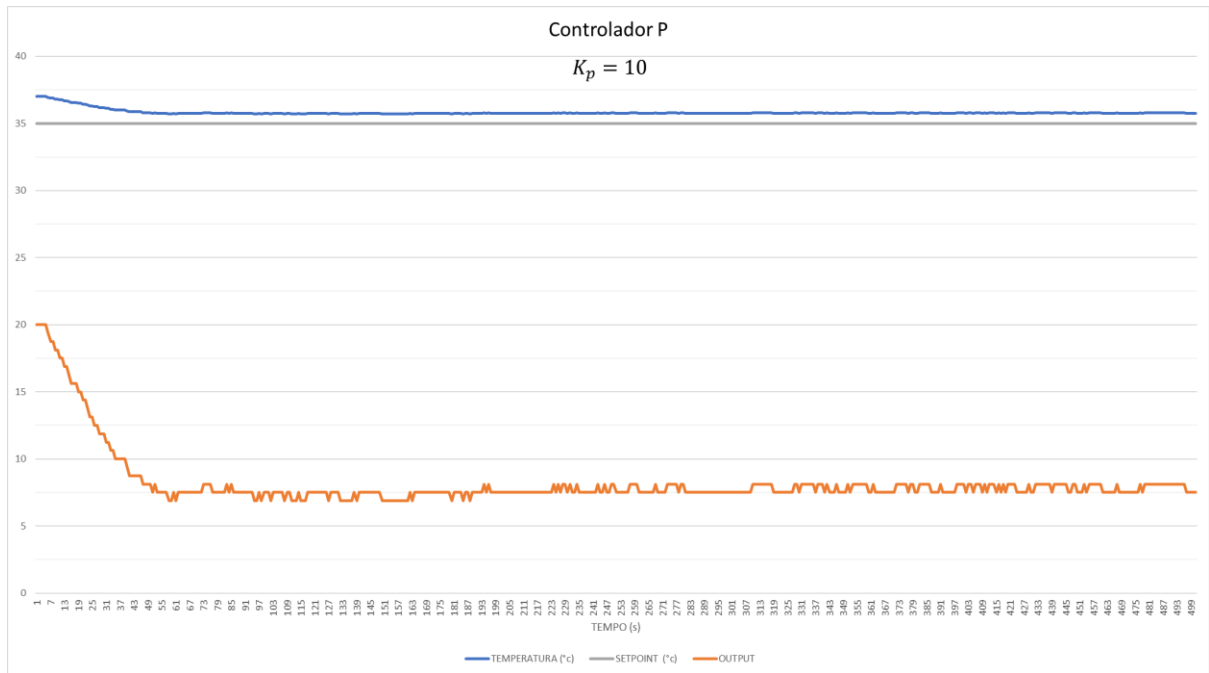
Na figura 15 (a), temos o gráfico do controlador P com  $K_p = 10$ , podemos observar o erro de *offset* entre 0,75 e 0,81°C, ou seja, a temperatura no ambiente simulado ficou estabilizada entre 35,75 e 35,81°C, não alcançando em nenhum momento no valor do *setpoint*. Fazendo jus ao que já foi explanado neste artigo, no qual o controlador P normalmente não é capaz de zerar o erro no regime permanente. Também é possível constatar o sinal do *output* se comportando inversamente proporcional ao erro.

É possível observar o gráfico do controle do tipo PI na figura 15 (b), com

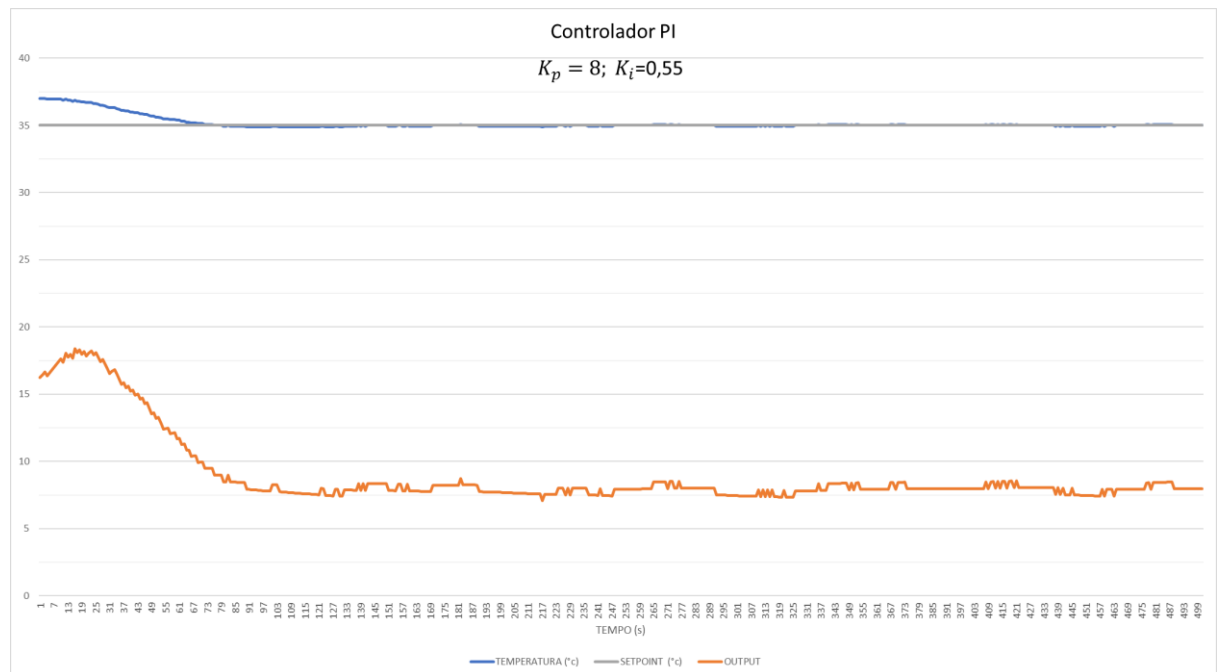
seus ganhos de  $K_p = 8$  e  $K_i = 0,55$ . Pode-se constatar que a temperatura lida pelo sensor atingiu o *setpoint*, a primeira vez em 75 segundos, após temos um pequeno *overshoot* na temperatura de até 34,88°C e voltamos ao *setpoint* 62s, ou seja, no tempo total de 137s, e segue estável com temperatura entre 34,94, 35 e 35,06°C, com o *output* respondendo de maneira análoga às pequenas oscilações a fim de manter o erro igual a zero, isto é, manter no *setpoint*.

Já o controlador PID, com seus ganhos configurados em  $K_p = 12$ ,  $K_i = 1,33$  e  $K_d = 27$ , também conseguiu estabilizar no *setpoint*, como podemos notar na figura 15 (c). A temperatura de 35°C foi atingida em 78s, e teve um *overshoot* de 34,63°C com período mais longo, atingindo o *setpoint* novamente em 231s, ou seja, 153s depois, ficando estabilizado entre 34,94, 35 e 35,06°C como no controle PI. Entretanto, o sinal de *output* nesse controlador, se comportou de forma bastante brusca, gerando elevados picos de sinais com variações de 16 unidades e até mesmo de mais de 30 unidades, diante de pequenas variações na temperatura, o que reitera a literatura quanto ao controle derivativo não operar bem quando há ruídos no sistema.

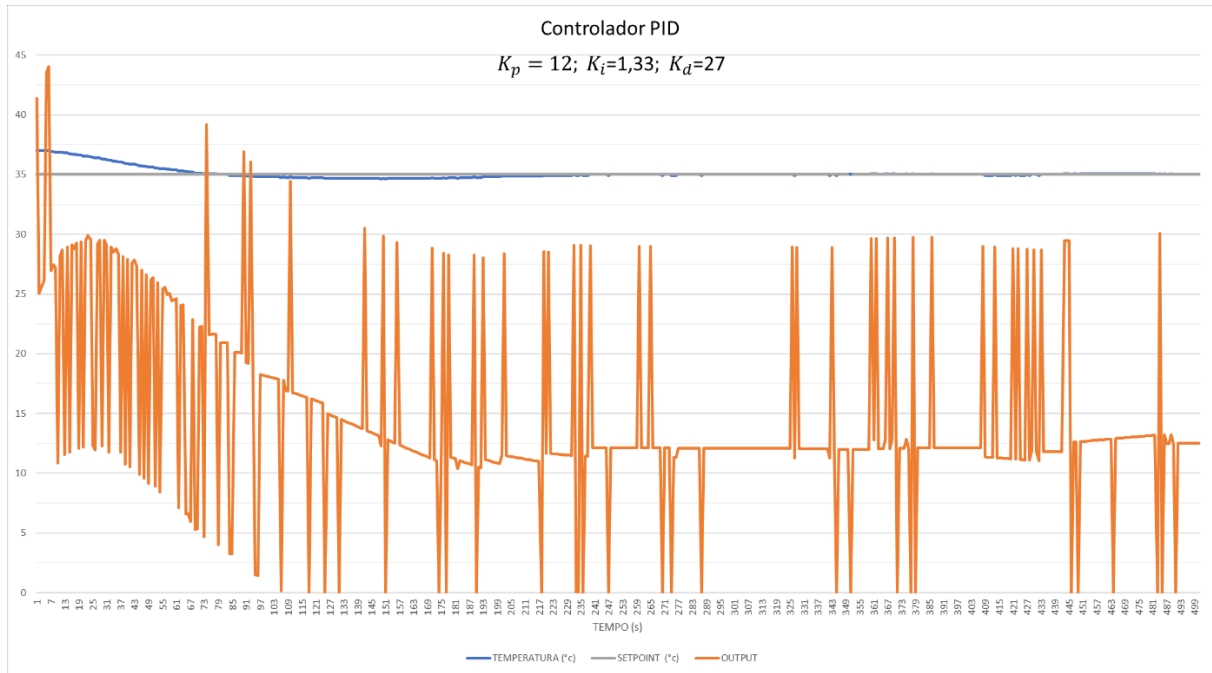
Figura 15 – Gráficos da implementação do método da frequência de Ziegler-Nichols com  $K_u = 20$  e  $T_u = 18$ , para os controles (a) P; (b) PI; e (c) PID.  
(a)



(b)



(c)

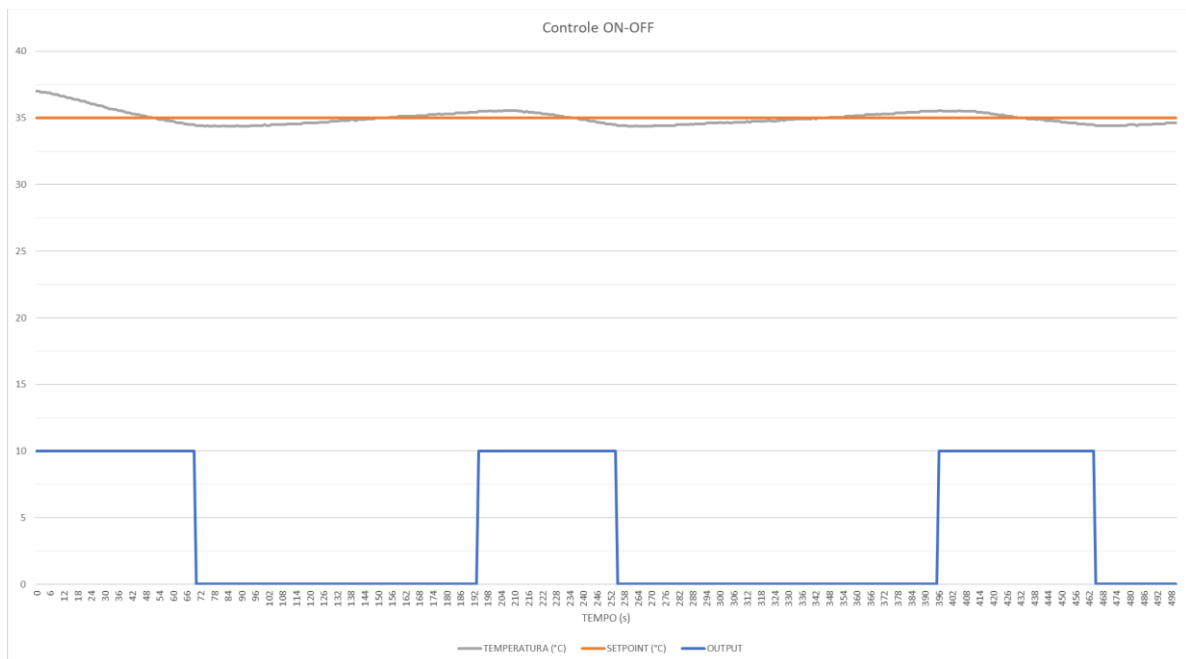


Fonte: O autor, 2023.

Após a implementação do controle PID, seguiu-se com o experimento do

controle *on-off* obtendo o resultado abaixo vide a figura 16, para completar a análise.

Figura 16 – Gráficos do controle *on-off*



Fonte: O autor, 2023.

Analisando os resultados agora citados, podemos chegar em alguns pontos, quanto a performance dos tipos de controladores.

O controlador P não foi capaz de zerar o erro, mantendo um erro de *offset* entre 0,75 e 0,85°C acima do *setpoint*. Logo para situações que necessitam de uma melhor assertividade de controle, não é indicado.

Já no caso do controle PI, obteve-se um bom desempenho dentro da situação do experimento, havendo uma estabilização com o erro muito próximo de zero, apresentando, em boa parte do experimento, a temperatura do ambiente equivalente ao *setpoint* definido.

No controlador PID, também se obteve a variável controlada em torno dos 35°C do *setpoint*. Porém se comparado ao controle PI, mostrou-se menos eficiente. Visto que o PID teve um intervalo de tempo maior até atingir o regime permanente, além do mais, seu sinal de *output* se comportou de forma descontínua com oscilações de amplitude muito elevadas, em alguns momentos de 0 a 36 unidades ou vice-versa, diante das pequenas variações da temperatura.

Já o caso do controle *on-off* correspondeu conforme descrito no algoritmo, mantendo corretamente a temperatura no intervalo da histerese definida.

## CONCLUSÕES

Em suma, este experimento teve como objetivo aplicar o controlador PID ao controle de temperatura em um ambiente simulado, utilizando como ferramenta o método de sintonia de Ziegler-Nichols e comparando sua eficácia com o controlador *on-off*. Para tal, foram executados vários testes para definição de seus parâmetros a fim da obtenção dos valores dos ganhos proporcional, integral e derivativo, e assim, posteriormente a implementação de cada controlador P, PI e PID. Também

apresentando um teste com o controlador *on-off* para assim fazer as análises pertinentes.

Diante dos resultados obtidos nos testes de cada controlador pelo método trabalhado, pode-se afirmar que o controle PI é o mais eficaz para essa aplicação. Sua estabilização em torno do *setpoint* foi mais rápida e obteve um baixo *overshoot*. Também em termo do comportamento, seu sinal de *output* mostrou-se mais regular.

Comparando ao controlador *on-off*, tem-se que sua eficácia é menor. pelo fato de que a temperatura não permanecer estabilizada no *setpoint*. Tratando de processos em que pequenas variações da temperatura não for tolerada, esse se torna um controle inapropriado.

Apesar de o controle PID também ter alcançado a estabilização da variável em torno do *setpoint*, seu intervalo de tempo até atingir tal ponto foi de quase 100s a mais comparado com o PI, e obteve um maior valor de *overshoot*. Além do seu sinal de *output* se comportar de forma bastante brusca e descontínua com as pequenas variações da temperatura lidas pelo sensor, mesmo estando em torno do *setpoint*, ou seja, encarando-as como ruídos.

Como proposição de estudos para trabalhos futuros pode-se considerar analisar a eficiência energética ou conservação do motor em relação ao comportamento do PID. Além disso, pode-se abordar temas como aplicações comerciais, como por exemplo para indústrias químicas ou produção de microprocessadores, que são ambientes de necessidade de alto controle; estudos com ênfase em conforto térmico, tanto para humanos quanto animais; o estudo do impacto da variação da temperatura no ambiente e também pode ser estudada a aplicabilidade dos controladores PID para processos de aquecimento do ambiente, ou até mesmo resfriamento e aquecimento.

## AGRADECIMENTOS

A Deus por ter me concedido a saúde e a determinação necessárias para não desanimar durante a realização deste trabalho. Gostaria também de agradecer aos meus pais, Eleriane e Max, e à minha noiva,

Luanna, que me incentivaram nos momentos difíceis, juntamente com os amigos e familiares que me apoiaram ao longo de toda essa jornada. E, por fim, quero expressar minha gratidão aos meus professores amigos, em especial ao meu orientador Elias, por ter desempenhado com dedicação e amizade sua função.

## REFERÊNCIAS

ARAÚJO, F. M. U. **Sistemas de controle**. 2007. Natal, 2007. 101 p. Disponível em: <https://www.dca.ufrn.br/~meneghet/FTP/Control/scv20071.pdf>. Acesso em: 14 de jan. de 2023.

ARDUINO Hardware, 2022. [S. I.]: **Arduino**. Disponível em: <https://www.arduino.cc/en/hardware>. Acesso em: 10 de abr. de 2023.

ÅSTRÖM, K. J.; HÄGGLUND, T. **PID controllers: theory, design and tuning**, 2 ed. North Caroline: Instrument Society of America, 1995.

BAIDYA, D.; DHOPTÉ, S.; BHATTACHARJEE, M. Sensing system assisted novel PID controller for efficient speed control of dc motors in electric vehicles. **IEEE Sensors Letters**, v. 7, n. 1, p. 1-4, jan. 2023, Disponível em: <https://ieeexplore.ieee.org/document/10007051>. Acesso em: 10 abr. 2023.

CONG, S. LIANG, Y. PID - like neural network nonlinear adaptive control for uncertain multivariable motion control systems. **IEEE Transactions On Industrial Electronic**, v 56, p. 3872-3879. 2009. Disponível em: <https://ieeexplore.ieee.org/document/4812095>. Acesso em: 26 fev. 2023.

MAXIM INTEGRATED PRODUCTS. **DS18B20: Programmable Resolution 1-Wire Digital Thermometer**. [S. I.]: 2019. Disponível em:

<https://www.analog.com/media/en/technical-documentation/data-sheets/DS18B20.pdf>. Acesso em: 09 de jan. de 2023.

GOODWIN, G. C.; GRAEBE, S. F.; SALGADO, M. E. **Control System Desing**. Valparaíso, 2000. 912 p. Disponível em: <http://caaelotel.elo.utfsm.cl/home/wp-content/uploads/Control-System-Design-SalgadoGoodwinGraebe.pdf>. Acesso em: 15 de jan de 2023.

HSU, C. Fractional Order PID Control for Reduction of Vibration and Noise on Induction Motor. **IEEE Transactions On Magnetics**, v. 55, n. 11, p. 1-7, nov. 2019. Disponível em: 10.1109/TMAG.2019.2933176. Acesso em: 11 abr. 2023.

JOHNSON, M. A.; MORADI, M. H. **PID Control: new identification and design methods**. London: Springer-Verlag London Limited, 2005. Disponível em: <https://link.springer.com/book/10.1007/1-84628-148-2>. Acesso em: 20 de jan. de 2023.

LI, J.; LI, W. On-Line PID Parameters Optimization Control for Wind Power Generation System Based on Genetic Algorithm. **IEEE Access**, v. 8, p. 137094-137100, 2020. Disponível em: 10.1109/ACCESS.2020.3009240. Acesso em: 11 abr. 2023.



MIYAGI, P. E. **Controle programável: fundamentos do controle de sistemas a eventos discretos.** São Paulo: Edgard Blücher, 1996. 194p.

NATIONAL INSTRUMENTS. **The PID Controller & Theory Explained.** [S. I]: 2022. Disponível em: <https://www.ni.com/pt-br/shop/labview/pid-theory-explained.html>. Acesso em: 14 de jan. de 2023.

OGATA, K. **Discrete-time Control Systems.** 2 ed. São Paulo: Pearson Prentice Hall, 2011.

MAXIM INTEGRATED PRODUCTS. **Overview of 1-Wire Technology and Its Use.** [S. I]: 2013. Disponível em: <https://www.analog.com/media/en/technical-documentation/tech-articles/guide-to-1-wire-communication--maxim-integrated.pdf>. Acesso em: 16 de jan. de 2023.

RUBAAI, A.; SITIRICHE, M.J.C.; OFOLI, A. R. Design and Implementation of parallel fuzzy PID controller for high-performance brushless motor drivers: an integrated environment for rapid control prototyping. **IEEE Transactions on**

**Industry Applications**, v 44, p. 1090-1098. 2008. Disponível em: <https://ieeexplore.ieee.org/document/4578808>. Acesso em: 26 fev. 2023.  
STEVAN, S. L. J.; SILVA, R. A. **Automação e instrumentação com Arduino: teoria e projetos**, 1 ed. São Paulo: Érica, 2015

THE PID Controller & Theory Explained. [S. I]: NATIONAL INSTRUMENTS. 06 jun. 2022. Disponível em: <https://www.ni.com/pt-br/shop/labview/pid-theory-explained.html>. Acesso em: 23 de jan. de 2023.

VANDOREN, V. Auto-Tuning Control Using Ziegler-Nichols. **Control Engineering**, 1 out. 2006. Disponível em: <https://www.controleng.com/articles/auto-tuning-control-using-ziegler-nichols/>. Acesso em: 16 de jan. de 2023.

ZIEGLER, J. B.; NICHOLS, N. B. Optimum settings for automatic controls Transactions. **ASME**, 65p. 759-768. 1942. Disponível em: [https://davidr.no/iiv3017/papers/Ziegler\\_Nichols\\_%201942.pdf](https://davidr.no/iiv3017/papers/Ziegler_Nichols_%201942.pdf). Acesso em: 26 fev. 2023.