



PROJETO FINAL DE GRADUAÇÃO

**PROPOSTA PARA CONTROLE DE TEMPERATURA
PARA A PRODUÇÃO ARTESANAL DA CERVEJA**

Douglas Vasconcelos Cardoso Alves

Brasília, Fevereiro de 2019

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

PROJETO FINAL DE GRADUAÇÃO
**PROPOSTA PARA CONTROLE DE TEMPERATURA
PARA A PRODUÇÃO ARTESANAL DA CERVEJA**

Douglas Vasconcelos Cardoso Alves

*Projeto Final de Graduação submetido ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Engenheiro de Redes de Comunicação.*

Banca Examinadora

Prof. Dr. Ricardo Zelenovsky, UnB _____
Orientador

Prof^a. Dra. Cláudia Jacy Barenco Abbas, UnB _____

Prof. Dr. Georges Daniel Amvame Nze, UnB _____

FICHA CATALOGRÁFICA

ALVES, DOUGLAS VASCONCELOS CARDOSO

PROPOSTA PARA CONTROLE DE TEMPERATURA PARA A PRODUÇÃO ARTESANAL DA CERVEJA [Distrito Federal] 2019.

xvi, 42 p., 210 x 297 mm (ENE/FT/UnB, Engenheiro, Engenharia Elétrica, 2019).

Projeto Final de Graduação - Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Controle de temperatura

2. Arduino

3. PID

4. Cerveja Artesanal

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

ALVES, D. V. C. (2019). *PROPOSTA PARA CONTROLE DE TEMPERATURA PARA A PRODUÇÃO ARTESANAL DA CERVEJA*. Projeto Final de Graduação, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 42 p.

CESSÃO DE DIREITOS

AUTORES: Douglas Vasconcelos Cardoso Alves,

TÍTULO: PROPOSTA PARA CONTROLE DE TEMPERATURA PARA A PRODUÇÃO ARTESANAL DA CERVEJA.

GRAU: Engenheiro de Redes de Comunicação ANO: 2019

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Projeto Final de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Os autores reservam outros direitos de publicação e nenhuma parte desse Projeto Final de Graduação pode ser reproduzida sem autorização por escrito dos autores.

Douglas Vasconcelos Cardoso Alves
Depto. de Engenharia Elétrica (ENE) - FT
Universidade de Brasília (UnB)
Campus Darcy Ribeiro
CEP 70919-970 - Brasília - DF - Brasil

DEDICATÓRIA

*Aos meus pais:
Vasconcelos e Darilene*

Douglas Vasconcelos Cardoso Alves

AGRADECIMENTOS

A Deus.

Ao professor Ricardo Zelenovsky, pelos ensinamentos e orientação.

A todos os professores que contribuíram para a minha formação.

Aos meus pais, pelo apoio incondicional.

Douglas Vasconcelos Cardoso Alves

RESUMO

O processo da produção da cerveja é composto por resfriamento e aquecimento em diferentes faixas de temperatura. A qualidade do produto final está sensivelmente associada à utilização adequada dessas faixas. Diferentemente das grandes indústrias cervejeiras, as microcervejarias, muitas vezes caracterizadas pela produção por *hobby*, não detêm de instrumental adequado para o controle de temperatura, procedendo muitas vezes de forma manual. Diante disso, são propostos sistemas de controle de temperatura para resfriamento e aquecimento baseados nos controles liga/desliga, proporcional e PID (Proporcional, Integral e Derivativo) com o emprego de aquecedor por Efeito Joule e refrigerador.

ABSTRACT

The beer production process consists of cooling and heating in different temperature ranges. The quality of the final product is closely associated with the proper use of these bands. Unlike the large breweries, microbreweries, often characterized by hobby production, do not have adequate instruments for temperature control, often proceeding manually. Therefore, temperature control systems are proposed for cooling and heating based on proportional, proportional and proportional PID controls (Proportional, Integral and Derivative) with the use of a Joule effect heater and refrigerator.

SUMÁRIO

1	Introdução	1
1.1	AMBIENTAÇÃO	1
1.2	MOTIVAÇÃO	3
1.3	DESCRIÇÃO.....	4
1.4	OBJETIVO	5
2	Materiais	6
2.1	REFRIGERADOR.....	6
2.2	ARDUINO MEGA 2560	8
2.3	SENSOR DE TEMPERATURA DS18B20.....	9
2.4	EBULIDOR ELÉTRICO.....	9
2.5	DISPOSITIVO DE RELÉ DE ESTADO SÓLIDO	10
3	Ensaio com Refrigerador	12
4	Ensaio de Controle de Temperatura - Aquecimento	24
5	Conclusão	34
	REFERÊNCIAS BIBLIOGRÁFICAS	36
	Apêndices	38

LISTA DE FIGURAS

1.1	Os 15 maiores produtores de cerveja do mundo (em quilolitros por ano). Fonte: [4]	1
1.2	Processo produtivo da cerveja. Adaptado de [8].	3
1.3	Diagrama do sistema de controle de temperatura para resfriamento.	4
1.4	Diagrama do sistema de controle de temperatura para aquecimento.	5
2.1	Parte interna do refrigerador.	6
2.2	Ciclo de refrigeração por compressão de vapor. Fonte: [11].	7
2.3	Componentes principais do refrigerador. Fonte: [12].	7
2.4	Arduino Mega 2560. Fonte: [14].	8
2.5	Sensor de temperatura DS18B20. Fonte: [16].	10
2.6	Ebulidor elétrico (Mergulhão). Fonte: [18].	10
2.7	Dispositivo de Relé de Estado Sólido.	11
2.8	Circuito do Dispositivo de Relé de Estado Sólido.	11
3.1	Leituras de temperatura.	12
3.2	Representação gráfica das temperaturas coletadas na <i>Posição 1</i>	13
3.3	Representação gráfica das temperaturas coletadas na <i>Posição 2</i>	13
3.4	Leitura de temperatura na água. Recipiente Plástico. <i>Posição 1</i>	15
3.5	Leitura de temperatura na água. Recipiente Alumínio. <i>Posição 1</i>	15
3.6	Leitura de temperatura na água. Recipiente Plástico. <i>Posição 2</i>	16
3.7	Leitura de temperatura na água. Recipiente Alumínio. <i>Posição 2</i>	16
3.8	Representação gráfica das temperaturas da água. Recipiente plástico. <i>Posi- ção 1</i>	17
3.9	Representação gráfica das temperaturas da água. Recipiente de alumínio. <i>Posição 1</i>	17
3.10	Representação gráfica das temperaturas da água. Recipiente plástico. <i>Posi- ção 2</i>	18
3.11	Representação gráfica das temperaturas da água. Recipiente de alumínio. <i>Posição 2</i>	18
3.12	Diagrama do sistema de controle de temperatura para resfriamento.	19
3.13	Resfriamento da água a 5°C. Controle liga/desliga. Recipiente de plástico.	20
3.14	Resfriamento da água a 5°C. Controle liga/desliga. Recipiente de plástico.	21
3.15	Resultado para o resfriamento da água a 5°C. Controle liga/desliga. Recipi- ente de plástico.	22
3.16	Resultado para o resfriamento da água a 5°C. Controle liga/desliga. Recipi- ente de alumínio.	22
3.17	Histerese em torno de 5°C. Controle liga/desliga. Recipiente de plástico.	22

3.18	Histerese em torno de 5°C. Controle liga/desliga. Recipiente de alumínio.	23
3.19	Aquecimento no refrigerador após resfriamento. Recipiente de plástico.	23
3.20	Aquecimento no refrigerador após resfriamento. Recipiente de alumínio.	23
4.1	Diagrama do sistema de controle de temperatura para aquecimento.	24
4.2	Erro estacionário. Fonte: [19].	25
4.3	Diagrama PID para o sistema de aquecimento.	26
4.4	Sistema de controle de temperatura para aquecimento	26
4.5	Controle proporcional. $K_p = 2,5$ $K_i = 0$ $K_d = 0$	27
4.6	Controle proporcional. $K_p = 5$ $K_i = 0$ $K_d = 0$	27
4.7	Controle proporcional. $K_p = 10$ $K_i = 0$ $K_d = 0$	28
4.8	Controle proporcional. $K_p = 6$ $K_i = 0$ $K_d = 0$	28
4.9	Controle proporcional mais integral. $K_p = 5$ $K_i = 2,5$ $K_d = 0$	29
4.10	Controle proporcional mais integral. $K_p = 5$ $K_i = 10$ $K_d = 0$	29
4.11	Controle proporcional mais integral. $K_p = 5$ $K_i = 100, 500, 1000$ e 5000 $K_d = 0$	30
4.12	Controle proporcional mais integral. $K_p = 5$ $K_i = 500$ e 700 $K_d = 0$	30
4.13	Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 10$ e 100	31
4.14	Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 2,5$ e 3	31
4.15	Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 3$ e $3,5$	32
4.16	Melhores controles: Controle Proporcional e Controle PID.	32
4.17	Tempo de funcionamento do aquecedor elétrico a cada intervalo de 30 segundos. Controlador Proporcional.	33
4.18	Tempo de funcionamento do aquecedor elétrico a cada intervalo de 30 segundos. Controlador PID.	33

LISTA DE TABELAS

- 3.1 Coeficientes de decaimento para as posições e tipos de material do recipiente. 15

1 INTRODUÇÃO

1.1 AMBIENTAÇÃO

A produção de cerveja é uma das mais importantes indústrias alimentares tradicionais [1]. Dentro desse cenário, destacam-se cinco cervejarias que juntas possuem mais de 50% do mercado mundial de cerveja: AB Inbev, que produz cerca de 20% da cerveja do mundo, a SABMiller, a Heineken, a Carlsberg e a China Resources Breweries [2]. Em relação à produção por país, consta no Anuário 2015 da Associação Brasileira da Indústria da Cerveja (CervBrasil), que segundo pesquisa feita pela Kirin Beer University, em 2014, os três principais maiores produtores mundiais de cerveja são, na ordem: China, EUA e Brasil [3].

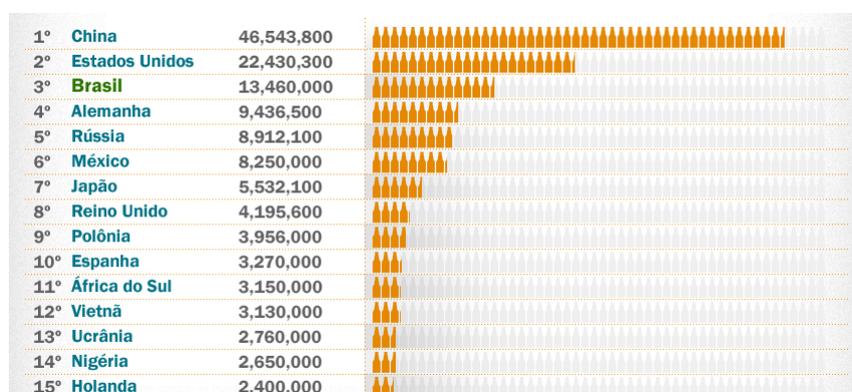


Figura 1.1: Os 15 maiores produtores de cerveja do mundo (em quilolitros por ano). Fonte: [4]

Distante desse mercado de grande volume de produção, encontram-se as microcervejarias, caracterizadas pela produção em baixa quantidade, sendo em alguns casos por *hobby*. Geralmente, produzem cervejas com características diferenciadas das cervejas tradicionais produzidas pelas grandes empresas.

Um estudo realizado pela MindMiners - *startup* brasileira especializada em pesquisas digitais - em parceria com a A.T. Kearney - empresa de consultoria empresarial norte-americana - concluiu que as cervejas artesanais têm ganhado espaço em relação às cervejas tradicionais, resultado de mudanças nos hábitos dos consumidores de cerveja [5].

Segundo o Sebrae (Serviço Brasileiro de Apoio às Micro e Pequenas Empresas), o conhecimento a fundo do processo produtivo é importante para quem quer atuar no ramo, pois a qualidade e a criatividade na elaboração das receitas são essenciais para proporcionar experiências sensoriais diferenciadas [6]. O processo produtivo, que possui a água, o malte e o lúpulo como elementos principais, segue o processo tradicional dividido nas seguintes fases: brassagem, fermentação, maturação, filtração, acabamento, envasamento e pasteurização.

A brassagem é a fase inicial na qual os elementos principais da cerveja são misturados e dissolvidos em água para formarem o líquido base, chamado mosto. Resumidamente, essa fase é composta pelas seguintes etapas [7]:

- Moagem do malte e dos demais ingredientes ocasionando a liberação do amido;
- Mistura com água;
- Aquecimento para auxiliar na dissolução;
- Formação de açúcar pela atuação das enzimas do malte no amido. Visando a não inativação das enzimas, a temperatura máxima é de 72 °C;
- Filtragem visando a separação das cascas do malte e dos demais ingredientes. Após essa etapa, obtém-se o mosto;
- Acréscimo do lúpulo;
- Dissolução do lúpulo e esterilização por meio da fervura do mosto;
- Resfriamento para deixar mistura em condições adequadas para a fermentação. A etapa é realizada em trocadores de calor entre 9 a 15 °C, acompanhada de aeração.

A fermentação é a principal fase para se determinar o sabor da cerveja. Para que se tenha um bom resultado, o controle da temperatura, geralmente na faixa de 10 °C a 25 °C, necessita ser preciso. Nessa fase, o fermento (levedura) é acrescido ao mosto para transformar o açúcar presente nele em álcool e gás carbônico. Essa mistura é mantida nos fermentadores, que são grandes tanques, para que a fermentação aconteça [7].

Após o fim da fermentação, a cerveja é resfriada a 0 °C. Através de sedimentação, grande parte da levedura é separada e há o início da maturação que pode levar de 6 a 30 dias. Nela acontece a chamada fecundação secundária, em que o açúcar restante é consumido pelo fermento residual, que também elimina substâncias indesejáveis resultantes da fermentação. A cerveja é mantida em tanques maturadores a temperaturas abaixo de 0°C. Com o final dessa fase, a cerveja está praticamente pronta com aroma e sabor estabelecidos [7].

A fase de filtração tem como objetivo deixar a cerveja com aspecto cristalino. Para isso, partículas em suspensão, principalmente do fermento, são retiradas. Essa fase não altera a composição e o sabor da cerveja [7].

Na fase de acabamento, a cerveja é acrescida de estabilizantes e antioxidantes para aumentar o seu prazo de validade. Posteriormente, acontece a fase de envasamento da cerveja em latas ou garrafas, seguida da fase de pasteurização, em que a cerveja é aquecida a 60 - 70 °C e resfriada. A pasteurização tem como objetivo eliminar micro-organismos que podem alterar a qualidade da cerveja [7].

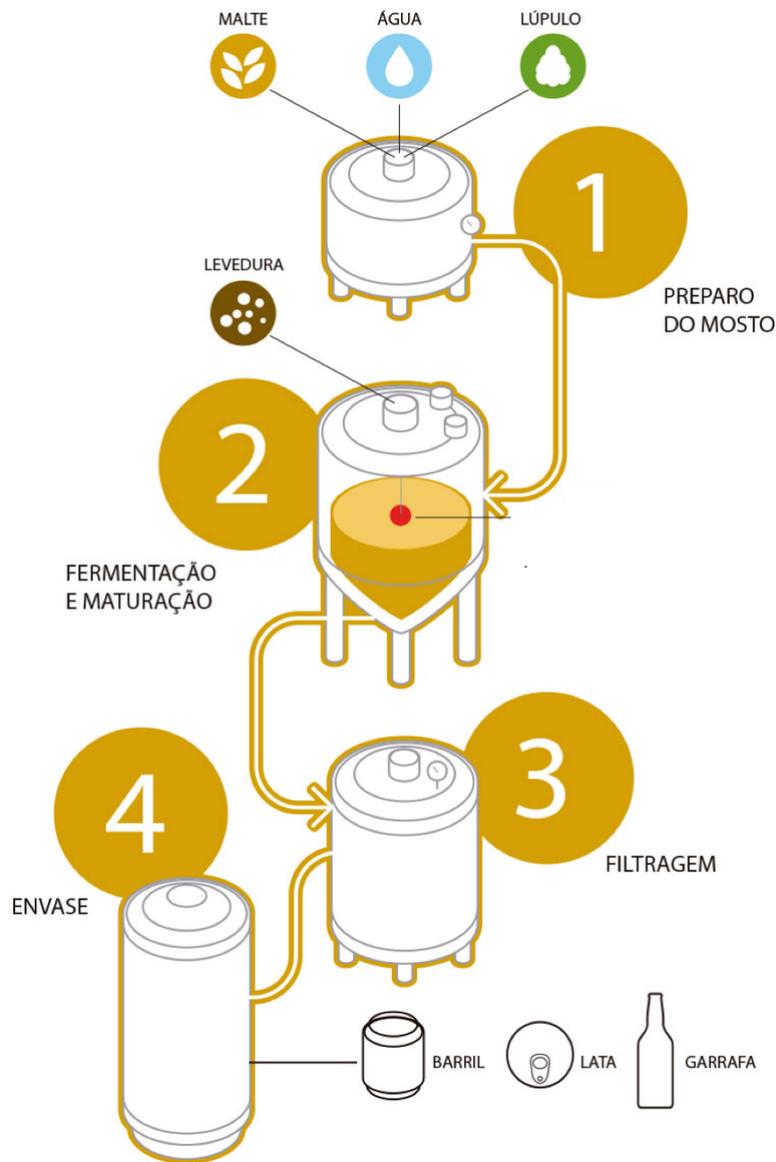


Figura 1.2: Processo produtivo da cerveja. Adaptado de [8].

1.2 MOTIVAÇÃO

O processo de fabricação da cerveja está fortemente relacionado ao controle de temperatura. Com exceção das fases de filtração, acabamento e envasamento, todas as demais fases exigem ações de aquecimento ou resfriamento. Ademais, a qualidade do produto final está sensivelmente associada ao uso de temperaturas ideais.

Por outro lado, um dos principais problemas das microcervejarias é a qualidade instável da cerveja, devido ao baixo nível de automação, tecnologia e equipamentos utilizados [1], sendo adotados meios manuais em muitos casos.

1.3 DESCRIÇÃO

Tendo em vista a problemática do controle de temperatura na produção de cerveja em microcervejarias, este trabalho tem como tema o controle de temperatura no aquecimento e resfriamento da água. No contexto deste trabalho, a água foi utilizada em substituição ao mosto e à cerveja, já que ela possui calor específico e densidade semelhantes ao do mosto e é, obviamente, de maior facilidade de obtenção.

Para o processo de aquecimento e resfriamento, propõem-se sistemas de baixa complexidade e compostos por dispositivos presentes no cotidiano - excetuando-se o Arduino Mega 2560 e o relé de estado sólido.

O sistema de controle de temperatura para o resfriamento tem como base a refrigeração por compressão de vapor e o controle liga/desliga. O sistema é mostrado na Fig. 1.3. O controle da temperatura da água presente no refrigerador é realizado pelo Arduino Mega que atua como um controlador liga/desliga, utilizando como base a temperatura de momento do líquido. Sua ação de resposta pode ser descrita da forma: quando a temperatura está acima de um valor pré-ajustado, ele liga o refrigerador; quando a temperatura está abaixo de um valor pré-ajustado, o refrigerador é desligado. Além de exercer o papel de controlador, o Arduino armazena registros de leituras de temperatura realizadas pelo termômetro digital DS18B20 para a análise do sistema.

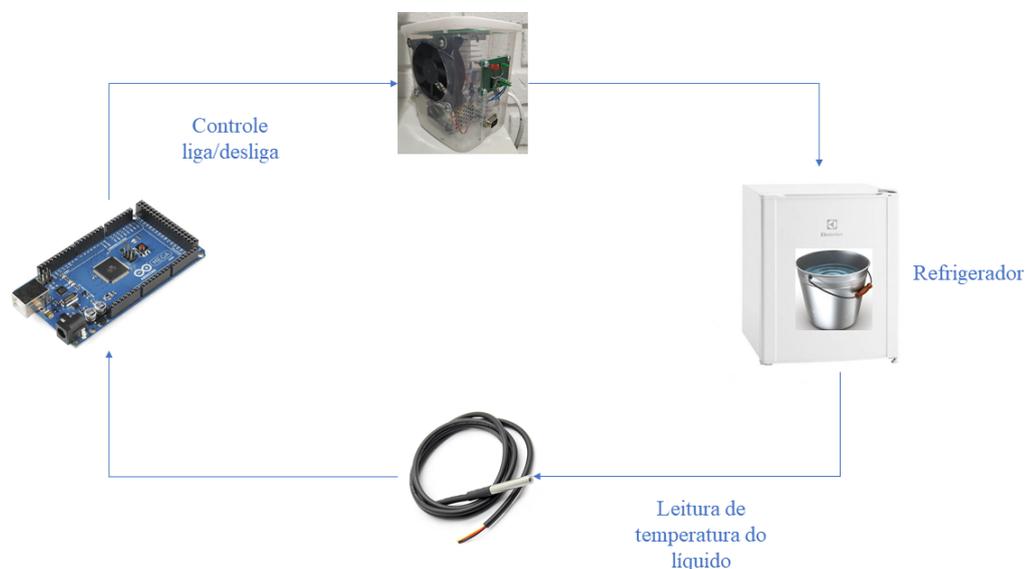


Figura 1.3: Diagrama do sistema de controle de temperatura para resfriamento.

O sistema de controle de temperatura para aquecimento é mostrado na Fig. 1.4. Assim como no sistema de controle de temperatura para resfriamento, nele o controle da temperatura da água é exercido pelo Arduino, no entanto, o tipo de controle usado é diferente. Para o aquecimento, são usados os controles proporcional, proporcional mais integral e PID. A

ação de tais controles é pautada no valor de deslocamento da variável medida, neste caso a temperatura, em relação ao valor desejado.

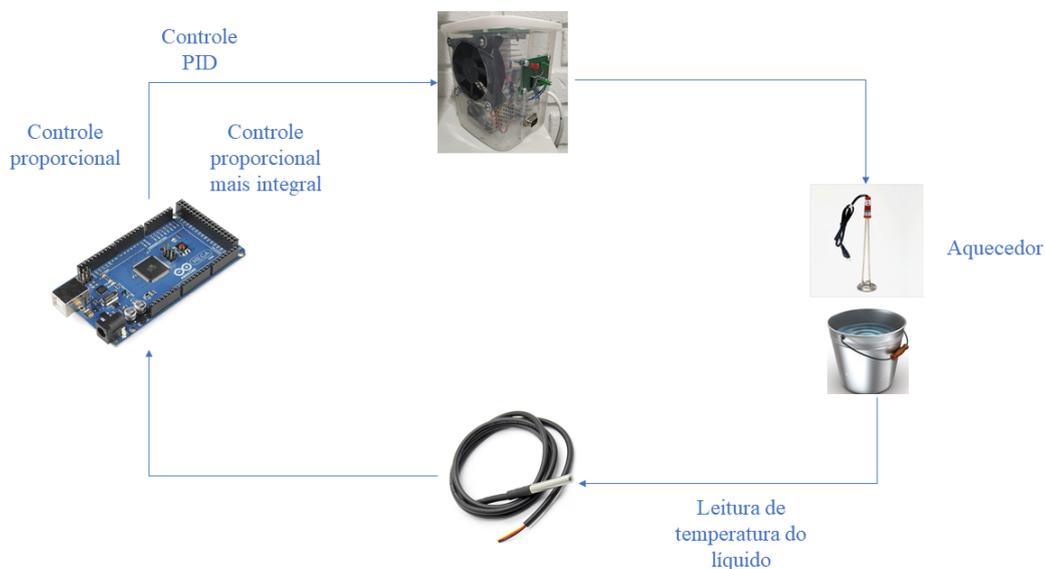


Figura 1.4: Diagrama do sistema de controle de temperatura para aquecimento.

Em ambos os sistemas, é utilizado um dispositivo, cujo principal componente é o relé de estado sólido, que tem como função atuar como interface entre o Arduino Mega e o dispositivo que realiza o resfriamento ou aquecimento, o refrigerador e o ebulidor elétrico, respectivamente.

1.4 OBJETIVO

Este trabalho propõe sistemas de controle de temperatura de baixa complexidade para o processo produtivo da cerveja. Utilizando-se de controles liga/desliga, proporcional e PID (Proporcional, Integral e Derivativo), pretende-se atingir e manter a temperatura da água em determinados valores, tanto em ensaios envolvendo resfriamento como aquecimento.

2 MATERIAIS

2.1 REFRIGERADOR

O refrigerador, utilizado nos ensaios envolvendo refrigeração, é o Frigobar RE80 da marca Electrolux. Possuindo as dimensões aproximadas de 64 x 49,5 x 54 cm, ele conta com um capacidade líquida de 79 litros. Em seu interior, há um congelador, um compartimento *cold drink* (gaveta de degelo), uma prateleira interna removível, prateleira porta latas e prateleira removível para garrafas, como mostra a Fig. 2.1. Como gás refrigerante é utilizado o R134a (Hidrofluorcarbono HFC). As pressões de alta e baixa são 1456 KPa e 122 KPa (198 psi e 4,5 psi), respectivamente. Possui três faixas de funcionamento: mínima, média e máxima. As informações técnicas das faixas de funcionamento, bem como, a capacidade de resfriamento, que consiste na taxa de remoção de calor do sistema refrigerado, não são informadas no manual do eletrodoméstico.



Figura 2.1: Parte interna do refrigerador.

O papel do refrigerador é transferir calor de uma região de temperatura mais baixa para uma região de temperatura mais alta. Ele trabalha em ciclo utilizando um fluido de trabalho chamado de refrigerante. O fluido refrigerante tem como característica evaporar a baixas pressões e condensar a altas pressões [9]. Quanto ao ciclo, chamado de ciclo de refrigeração, destacam-se o ciclo de refrigeração por compressão de vapor e o ciclo de refrigeração a gás. No ciclo de refrigeração por compressão de vapor, o refrigerante é transformado em vapor e condensado de forma alternada. Já no ciclo de refrigeração a gás, o refrigerante sempre está na fase gasosa [10]. O ciclo de refrigeração por compressão de vapor é o mais utilizado e está presente no refrigerador utilizado nos ensaios.

O ciclo de refrigeração por vapor é composto por quatro componentes principais: um compressor, um condensador, uma válvula de expansão (tubo capilar) e um evaporador. No refrigerador, o compressor está localizado na parte inferior traseira, o condensador é representado pelas serpentinhas traseiras, a válvula de expansão na parte superior traseira, e o evaporador é representado pelos tubos do compartimento do congelador, conforme mostrado na Fig. 2.3 .

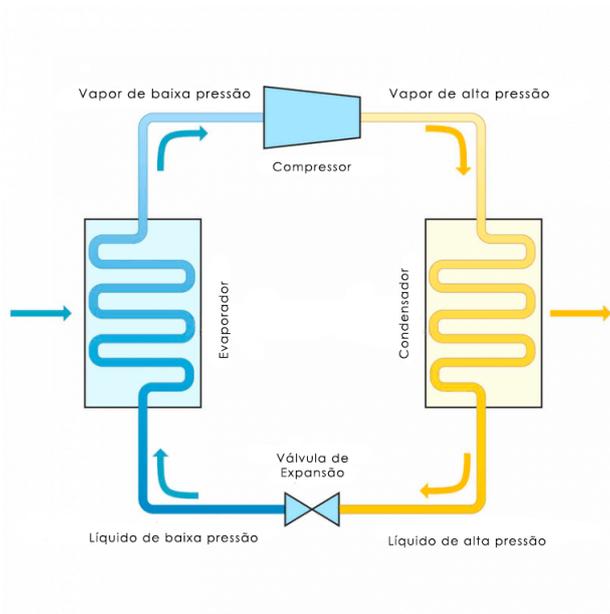


Figura 2.2: Ciclo de refrigeração por compressão de vapor. Fonte: [11].

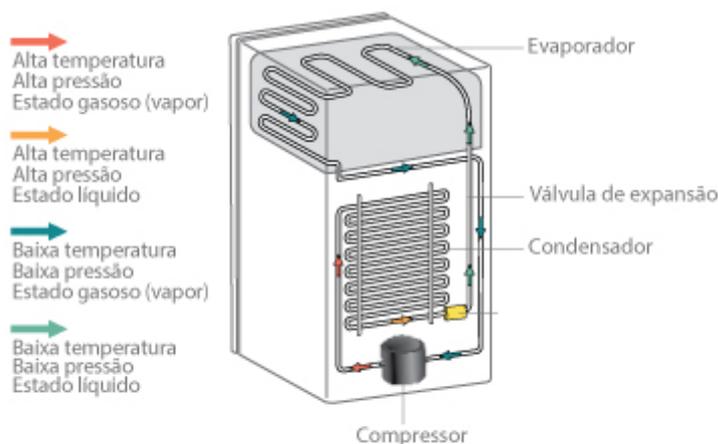


Figura 2.3: Componentes principais do refrigerador. Fonte: [12].

O compressor atua comprimindo o refrigerante, na forma de vapor, resultando no aumento da pressão e da temperatura do fluido, e posteriormente, bombeando-o para o condensador. Ao passo que o vapor se move pelo condensador, ele libera calor para o ambiente e condensa. Em seguida, é introduzido no tubo capilar, que cria uma resistência ao seu

movimento (estrangulamento), fazendo com que ele mude de um líquido aquecido a alta pressão para um líquido resfriado a baixa pressão, a caminho do evaporador. No evaporador, região de baixa pressão, o refrigerante evapora, ao absorver calor da parte interna do refrigerador. Na saída do evaporador, o fluido é sugado pelo compressor, onde é iniciado o ciclo novamente. Com isso, o refrigerante age retirando calor de dentro do refrigerador (pelo evaporador), passando-o para o ambiente externo (pelo condensador) [9].

2.2 ARDUINO MEGA 2560

O Arduino Mega 2560 é uma placa microcontroladora baseada no microcontrolador ATmega2560 (Fig. 2.4). Ele possui 54 pinos digitais, operando a 5 volts, que podem ser usados como entrada e saída. Cada um deles possui um resistor de *pull-up* interno de 20-50 k Ω e podem fornecer ou receber 20 mA como requisito recomendado para operação. Do total de 54 pinos, 15 podem ser usados para PWM (Modulação por largura de pulso) e alguns possuem funções especializadas, como receber e transmitir dados seriais, interrupção externa, LED, SPI (*Serial Peripheral Interface*) e TWI (*Two Wire Interface*) - essas duas últimas relacionadas à parte de comunicação. Além dos pinos digitais, a placa possui 16 entradas analógicas com resolução de 10 bits [13].

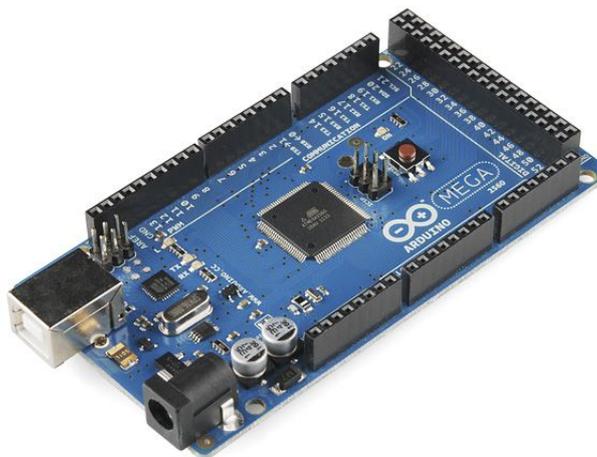


Figura 2.4: Arduino Mega 2560. Fonte: [14].

Na parte de comunicação, o Arduino Mega 2560 tem quatro UARTs (*Universal Asynchronous Receiver-Transmitter*) para comunicação serial TTL e suporta comunicação TWI e SPI.

Quanto à alimentação, ela pode ser realizada via conexão USB ou uma fonte de alimentação externa que opera de 6 a 20 volts, contudo, recomenda-se um intervalo de operação de 7 a 12 volts [13].

O tamanho da placa é de 5,3 x 10,2 x 1 cm, estando nela um cristal oscilador de 16 MHz, uma conexão USB, um *power jack*, um soquete de comunicação ICSP (*In-Circuit*

Serial Programming), um botão de *reset*, além das UARTS, pinos digitais e seriais [13].

Estão presentes no Mega 2560 três tipos de memória: Flash, SRAM e EEPROM. A memória Flash, destinada ao armazenamento do código do usuário, possui 256 kbytes de tamanho, sendo que 8 kbytes são pertencentes ao *bootloader*. A memória SRAM (memória de acesso aleatório), local para as variáveis criadas quando o código é executado, tem 8 kbytes de tamanho. Já a memória EEPROM, que é destinada ao armazenamento de dados a longo prazo, tem como tamanho 4 kbytes [13]. Destaca-se que as memórias Flash e EEPROM, diferentemente da memória SRAM, não são voláteis, ou seja, as informações permanecem nelas após a placa ser desligada da fonte de energia.

Os códigos para o Arduino Mega 2560 podem ser escritos e passados para a placa, com bastante praticidade, através da *open-source* Arduino Software (IDE).

Neste trabalho, nos ensaios realizados, o Arduino Mega possui papel central na execução dos algoritmos para controle da temperatura, na coordenação do funcionamento do refrigerador, do ebulidor elétrico (mergulhão), por meio do dispositivo de relé de estado sólido, além de armazenar leituras de temperatura coletadas na sua memória EEPROM.

2.3 SENSOR DE TEMPERATURA DS18B20

Nos ensaios realizados, a coleta de leituras de temperatura é realizada por meio do sensor de temperatura DS18B20, que é um termômetro digital produzido pela Dallas Instruments. Ele fornece temperatura em graus Celsius numa faixa de medição de temperatura que varia de $-55\text{ }^{\circ}\text{C}$ a $125\text{ }^{\circ}\text{C}$, com uma precisão de $\pm 0,5\text{ }^{\circ}\text{C}$ entre $-10\text{ }^{\circ}\text{C}$ e $85\text{ }^{\circ}\text{C}$. O DS18B20 se comunica com a placa Arduino por meio do protocolo de comunicação *1-Wire*, que requer apenas uma linha de dados [15].

Para utilização na água, o sensor DS18B20 é selado numa cápsula metálica, como mostra a Fig. 2.5.

2.4 EBULIDOR ELÉTRICO

Nos ensaios envolvendo aquecimento, é utilizado como aquecedor um ebulidor elétrico metálico, também conhecido como mergulhão, de potência 2 kW, cujo comprimento é de 36 cm (Fig.2.6). O funcionamento do ebulidor é baseado no efeito Joule. Ao ser ligado a uma tomada da rede elétrica, a corrente elétrica passa por sua resistência elétrica sendo transformada em energia térmica. Basicamente, os elétrons da corrente elétrica transferem parte de sua energia cinética (energia relacionada ao movimento) ao se colidirem com os átomos da resistência elétrica. Isso faz com o que o átomo aumente a sua temperatura, devido ao aumento do estado de agitação advindo da colisão, ocasionando consequentemente

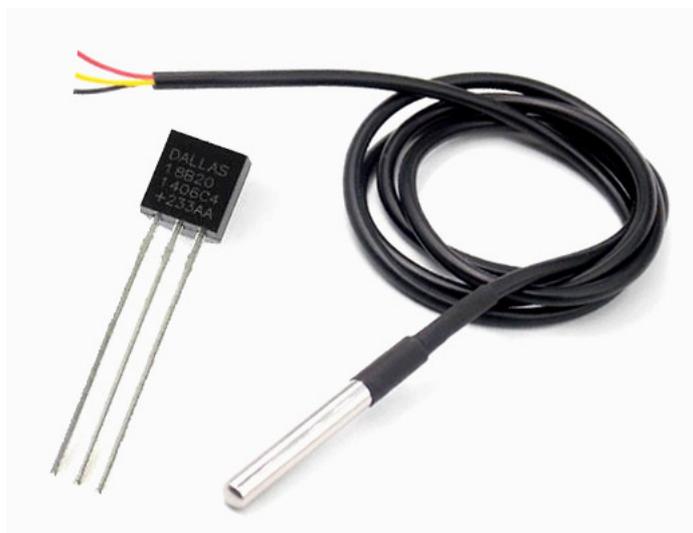


Figura 2.5: Sensor de temperatura DS18B20. Fonte: [16].

o aumento da temperatura do ebulidor [17].



Figura 2.6: Ebulidor elétrico (Mergulhão). Fonte: [18].

2.5 DISPOSITIVO DE RELÉ DE ESTADO SÓLIDO

Para o controle do refrigerador e do ebulidor elétrico, ambos cargas AC, a partir de sinais DC do Arduino Mega 2560, foi construído o dispositivo da Fig. 2.7 que tem como elemento principal um relé de estado sólido.

O dispositivo possui três conexões: uma para o Arduino Mega, uma para equipamentos e uma para a rede elétrica. O sinal de comando proveniente do Arduino Mega no terminais de entrada do dispositivo determina o acionamento do equipamento. Resumidamente, tal sinal de comando é conduzido ao relé de estado sólido, que ao recebê-lo, conduz e permite

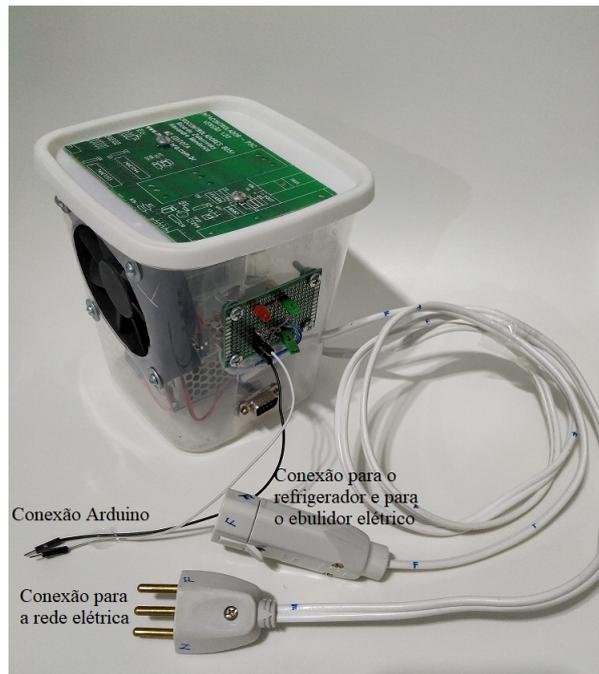


Figura 2.7: Dispositivo de Relé de Estado Sólido.

que o equipamento seja alimentado pela rede elétrica a 220V. Tendo em vista o tempo de duração dos ensaios realizados, foi inserida uma ventoinha para o dissipação de calor, cuja alimentação é gerada pelo retificador de 5V. Quando o relé está acionado, o LED é aceso.

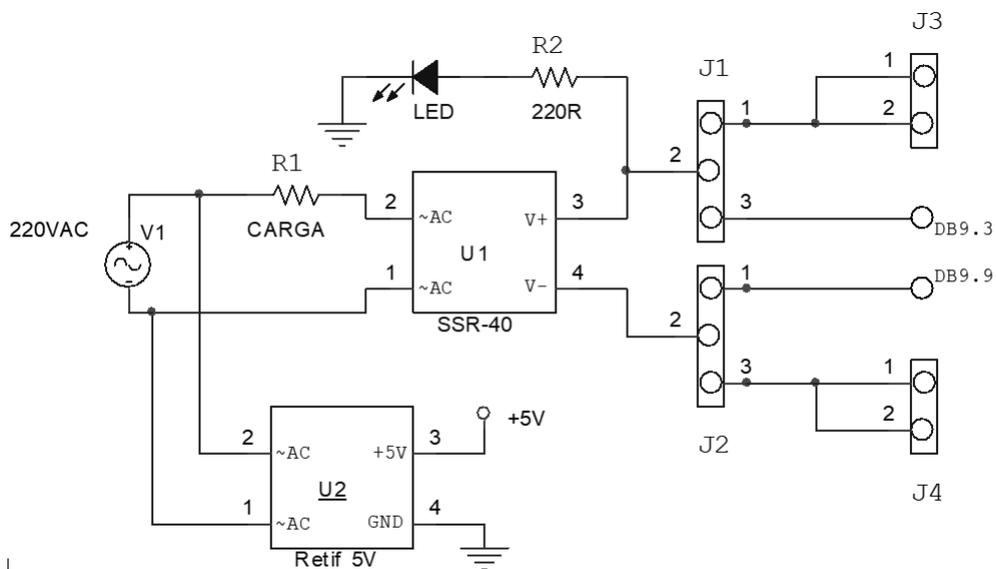
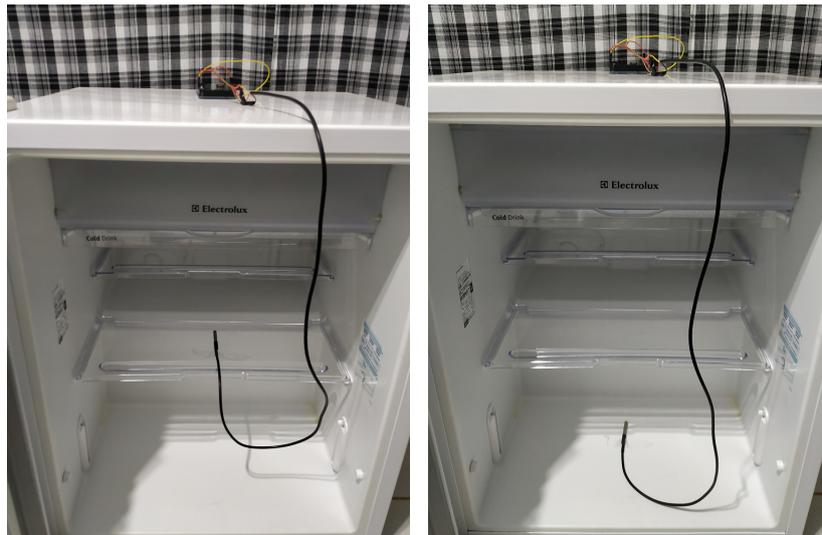


Figura 2.8: Circuito do Dispositivo de Relé de Estado Sólido.

3 ENSAIOS COM REFRIGERADOR

O primeiro conjunto de ensaios teve como objetivo examinar a diferença de variação de temperatura com o tempo em dois pontos internos do refrigerador: prateleira removível, localizada no centro do eletrodoméstico, e a parte inferior. A escolha de tais posições foi devida as suas dimensões, tendo em vista serem as únicas que possibilitavam a inserção dos recipientes utilizados nos ensaios posteriores. Neste trabalho, a prateleira removível será referenciada como *Posição 1* e a parte inferior, como *Posição 2*, e o refrigerador será utilizado na faixa de funcionamento máxima.

Para a realização do ensaio, o termômetro digital DS18B20 foi colocado em cada um dos dois locais, sem qualquer invólucro, pelo período de 6h, conforme mostrado nas Figs. (3.1(a)) e (3.1(b)). Durante esse intervalo de tempo, realizaram-se leituras da temperatura interna do refrigerador com a porta fechada, que eram armazenadas sequencialmente na memória EEPROM do Arduino MEGA 2560, sendo uma leitura a cada 30 s. O código utilizado se encontra no Apêndice 1.



(a) Leitura de temperatura. *Posição 1*. (b) Leitura de temperatura. *Posição 2*.

Figura 3.1: Leituras de temperatura.

As Figuras (3.2) e (3.3) mostram a representação gráfica dos dados de temperatura coletados na *Posição 1* e na *Posição 2*, respectivamente. Nota-se em ambos os gráficos, por volta dos primeiros 50 minutos, uma tendência de queda linear dos valores. Considerando os valores de temperatura desse intervalo de tempo, foram realizadas regressões lineares que resultaram nas seguintes equações de reta de regressão para os citados gráficos, na sequência:

$$T_p = -0,47t + 27,60 \quad (3.1)$$

$$T_p = -0,36t + 24,22 \quad (3.2)$$

Em que T_p representa a temperatura e t representa o tempo.

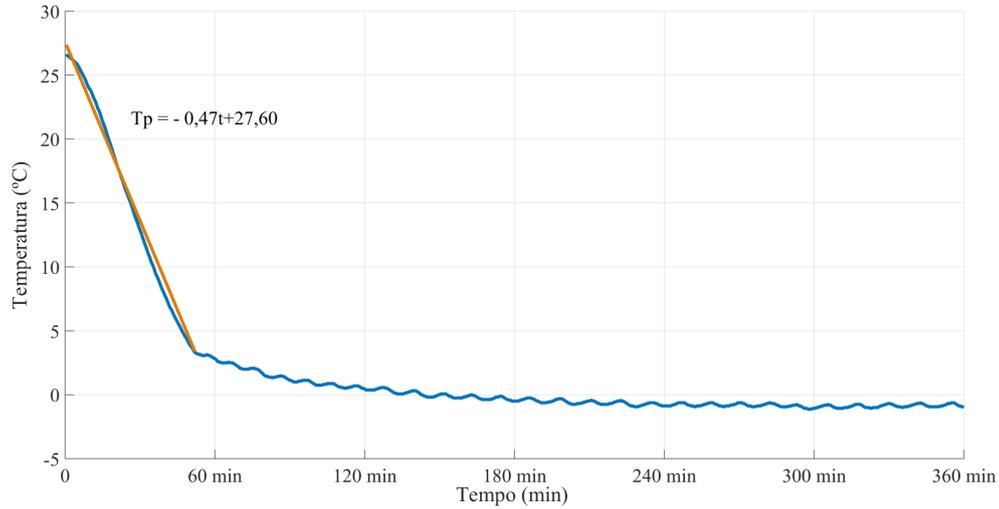


Figura 3.2: Representação gráfica das temperaturas coletadas na *Posição 1*.

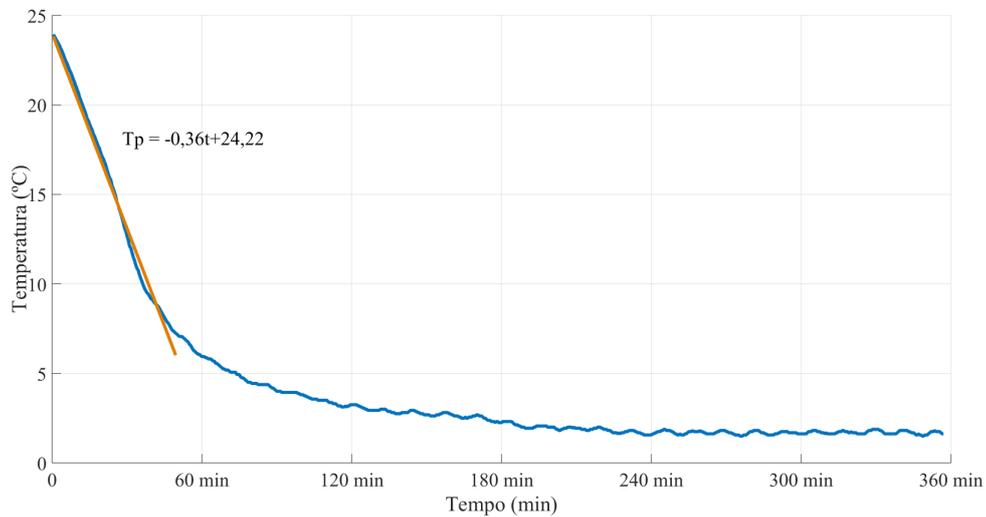


Figura 3.3: Representação gráfica das temperaturas coletadas na *Posição 2*.

Tendo em vista que os ensaios objetivavam o exame da variação da temperatura com o tempo, a temperatura inicial, representada pelos coeficientes lineares das Eq.(3.1) e Eq.(3.2), não é relevante. Por outro lado, os coeficientes angulares das equações, que representam a taxa média de variação, têm papel importante para a análise.

Por meio da comparação dos coeficientes angulares das Eq.(3.1) e Eq.(3.2), verifica-se que a *Posição 1* possui uma maior rapidez na queda de temperatura do que a *Posição 2*. Ademais, observa-se que a partir de 240 min, os valores de temperatura se estabilizam num comportamento de subida e descida em torno de um valor mediano. Para a *Posição 1* esse valor é de $-0,8472$ °C, atingindo valores máximos de subida e descida de $-0,625$ ° C e $-1,125$ °C, respectivamente, resultando num *ripple* de $0,5$ °C. Já para a *Posição 2*, o valor é de $1,6939$ °C, atingindo valores máximos de subida e descida de $1,875$ ° C e $1,5$ °C, respectivamente, resultando num *ripple* de $0,375$ °C.

Tais resultados são decorrentes do fato da *Posição 1* estar mais próxima do evaporador do refrigerador do que a *Posição 2*. É no evaporador, que geralmente está localizado na parte superior dos refrigeradores, que o refrigerante do eletrodoméstico retira calor, tornando a parte de cima mais fria em relação as demais.

O segundo conjunto de ensaios teve como objetivo examinar a variação da temperatura com o tempo em 1 litro de água contida num recipiente no refrigerador. A utilização da água em substituição ao mosto é devida a sua facilidade de obtenção comparada a ele, e por não implicar em prejuízos aos resultados, tendo em vista que ambas as substâncias possuem valores de calor específico e densidade aproximados. O calor específico da água é de $1,0$ cal/g.°C e do mosto, $0,93$ cal/g.°C. Ela possui densidade de 997 kg/m³, em condições ambiente, enquanto ele possui densidade relativa de $1,048$ antes da fermentação, e entre $1,005$ e $1,015$ após a fermentação. Além disso, a água representa por volta de 90% da composição em massa da cerveja [7]. Para se verificar a influência do material do recipiente, as leituras de temperatura foram feitas em vasilhames de plástico (polipropileno) e de alumínio.

Os recipientes de plástico e de alumínio foram dispostos no refrigerador conforme as Figs. (3.4), (3.5), (3.6) e (3.7). Para cada caso, foram realizadas, por meio do termômetro digital DS18B20, leituras de temperatura a cada 1 min, armazenadas sequencialmente na memória EEPROM do Arduino MEGA 2560. O tempo de duração dos ensaios foi de 12h.

Os valores de temperatura coletados estão representados graficamente nas Figs. (3.8), (3.9), (3.10) e (3.11). Nota-se uma tendência de decaimento exponencial nos resultados. Dessa forma, realizaram-se ajustes exponenciais que resultaram nas seguintes equações para os gráficos, respectivamente:

$$Tp = 31,80e^{(-0,0057t)} \quad (3.3)$$

$$Tp = 30,62e^{(-0,0065t)} \quad (3.4)$$

$$Tp = 24,73e^{(-0,0032t)} \quad (3.5)$$



Figura 3.4: Leitura de temperatura na água. Recipiente Plástico. *Posição 1*.



Figura 3.5: Leitura de temperatura na água. Recipiente Alumínio. *Posição 1*.

$$T_p = 25,36e^{(-0,0037t)} \quad (3.6)$$

Observa-se nas equações logo acima que os coeficientes de decaimento são maiores para as medições realizadas na *Posição 1* em relação às realizadas na *Posição 2*. Quanto ao tipo de material do recipiente, o alumínio apresentou maiores coeficientes de decaimento do que o plástico (Tab. 3.1).

	Plástico	Alumínio
<i>Posição 1</i>	0,0057	0,0065
<i>Posição 2</i>	0,0032	0,0037

Tabela 3.1: Coeficientes de decaimento para as posições e tipos de material do recipiente.



Figura 3.6: Leitura de temperatura na água. Recipiente Plástico. *Posição 2.*



Figura 3.7: Leitura de temperatura na água. Recipiente Alumínio. *Posição 2.*

Tais resultados são decorrentes do fato da *Posição 1* estar mais próxima do evaporador, região mais fria do refrigerador, do que a *Posição 2*, e do fato do alumínio ter uma maior capacidade de conduzir calor, ou seja, possui maior condutividade térmica, do que o plástico (polipropileno). As condutividades térmicas do alumínio e do polipropileno são de 237 W/m.K e 0,25 W/m.K, respectivamente. Portanto, a água no recipiente metálico resfriará mais rápida, pois as paredes do recipiente possibilitam maior condução de calor.

Partindo dos resultados anteriores, o terceiro conjunto de ensaios teve como objetivo a construção de um sistema que resfriasse e mantivesse um litro de água em 5°C, por meio da utilização de refrigeração a vapor. Para isso, o Arduino Mega 2560 passou a exercer o papel de controlador liga/desliga, controlando o funcionamento do refrigerador através do acionamento do dispositivo de relé de estado sólido. A ação do Arduino, baseada na leitura da temperatura da água fornecida pelo sensor de temperatura DS18B20, pode ser resumida

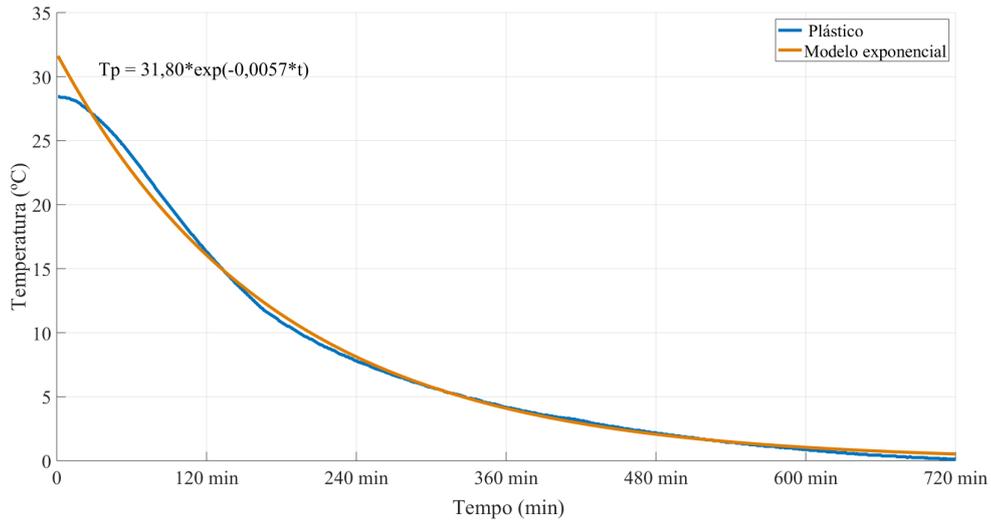


Figura 3.8: Representação gráfica das temperaturas da água. Recipiente plástico. *Posição 1*.

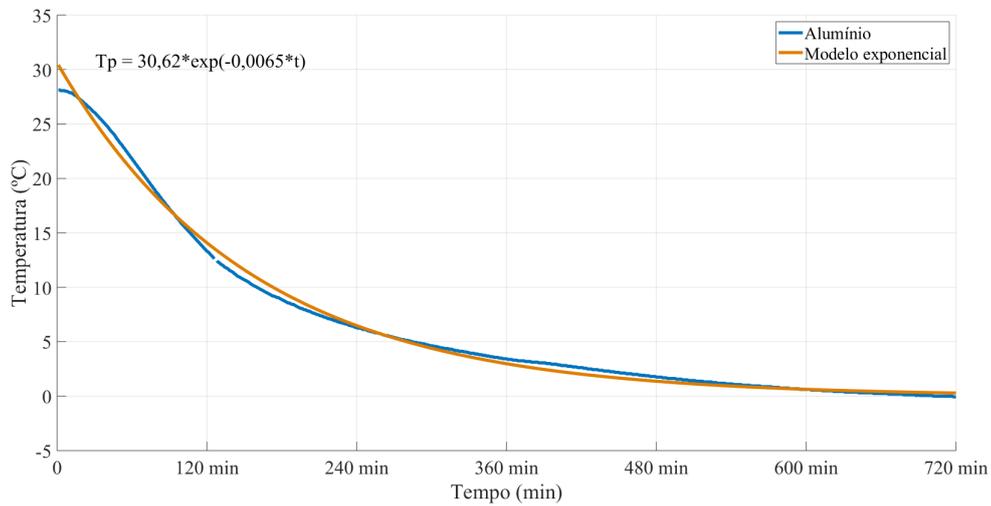


Figura 3.9: Representação gráfica das temperaturas da água. Recipiente de alumínio. *Posição 1*.

da seguinte forma:

- Se a temperatura da água estiver acima de um valor pré-ajustado, o refrigerador será ligado.
- Se a temperatura da água estiver abaixo ou for igual a um valor pré-ajustado, o refrigerador será desligado.

Utilizando o código presente no Apêndice 2 no Arduino Mega 2560, para torná-lo um controlador liga/desliga, procedeu-se aos ensaios que duraram 12h cada. A *Posição 1* do refrigerador foi escolhida tendo em vista o seu melhor desempenho no tocante ao resfriamento, como demonstrado em ensaios anteriores. Foram utilizados recipientes de plástico e de alumínio.

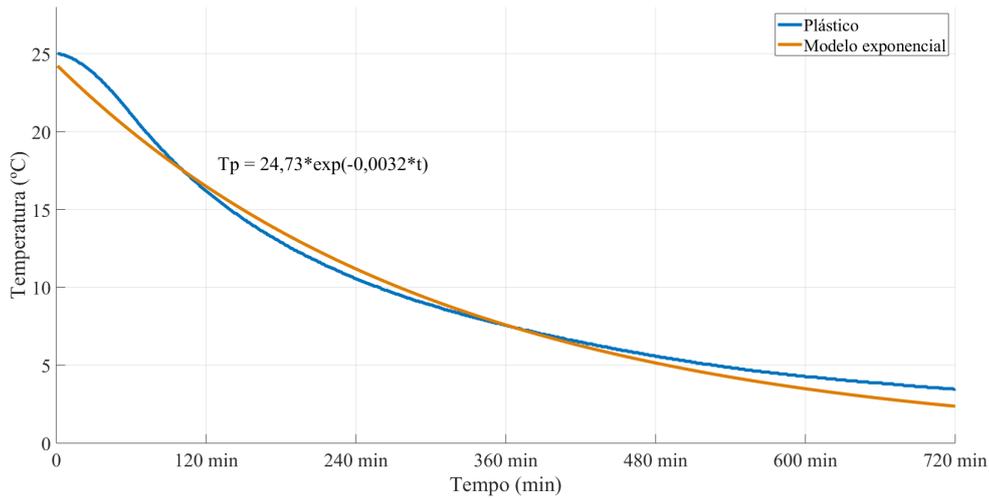


Figura 3.10: Representação gráfica das temperaturas da água. Recipiente plástico. *Posição 2.*

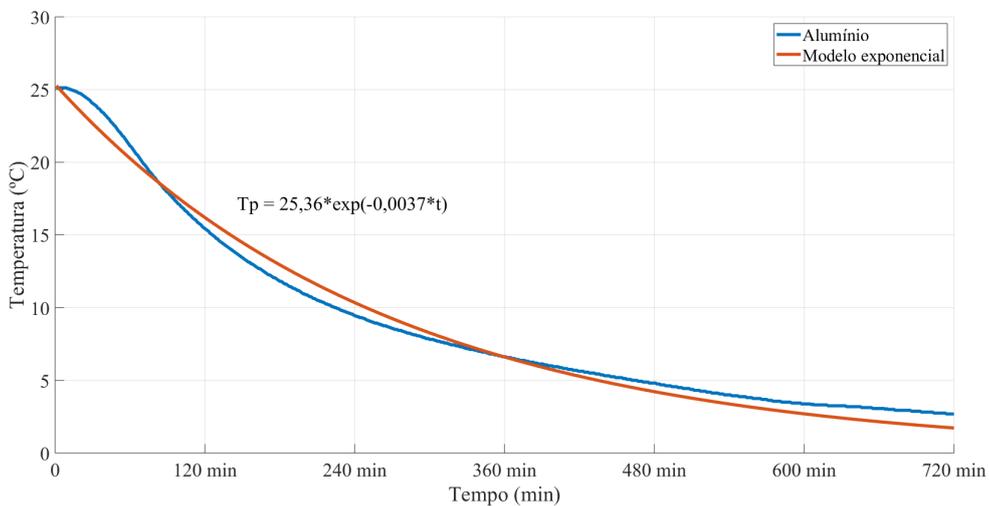


Figura 3.11: Representação gráfica das temperaturas da água. Recipiente de alumínio. *Posição 2.*

As Figuras 3.15 e 3.16 mostram a representação gráfica dos resultados obtidos. As equações associadas aos ajustes exponenciais dos gráficos, até o momento de atingimento da temperatura desejada, são:

$$Tp = 27,05e^{(-0,0053t)} \quad (3.7)$$

$$Tp = 26,00e^{(-0,0069t)} \quad (3.8)$$

Em ambos os casos, a temperatura da água foi diminuída para 5°C e mantida por volta desse valor de forma satisfatória. Para o ensaio realizado com o recipiente plástico, obteve-se oscilações máximas de 0,125°C e 0,375°C, acima e abaixo do valor de referência, respec-

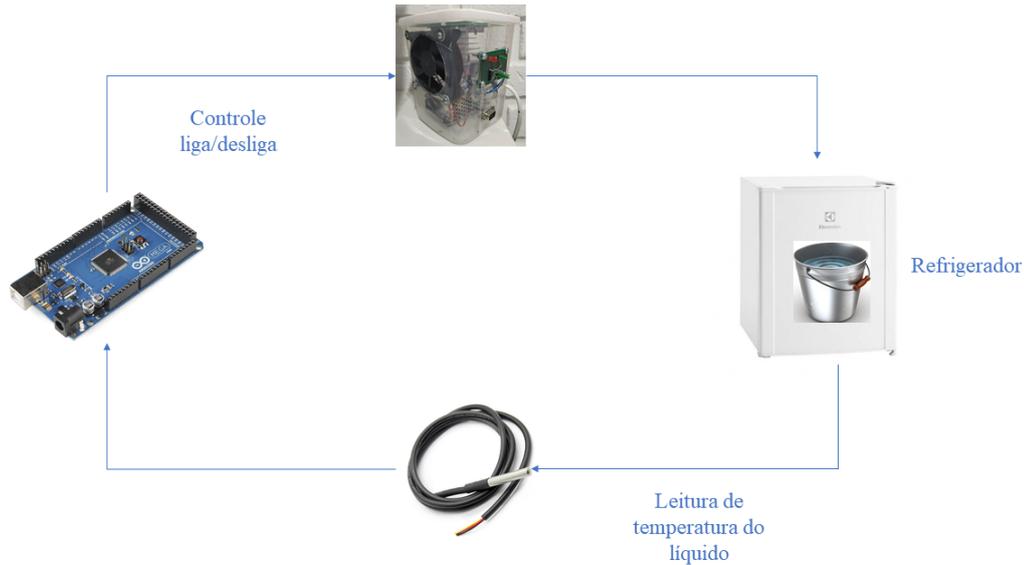


Figura 3.12: Diagrama do sistema de controle de temperatura para resfriamento.

tivamente, com um valor médio de $4,90^{\circ}\text{C}$, após o atingimento da temperatura desejada. Já para o ensaio realizado com o recipiente de alumínio, as oscilações máximas foram de $0,125^{\circ}\text{C}$ e $0,250^{\circ}\text{C}$, acima e abaixo do valor de referência, respectivamente, com um valor médio de $4,94^{\circ}\text{C}$, após o alcance da temperatura alvo.

Nota-se também que, no ensaio realizado com o recipiente de alumínio, o atingimento da temperatura desejada acontece mais rapidamente. Nele isso acontece após $4\text{h}11\text{min}$ do início do ensaio, enquanto que no ensaio realizado com o recipiente de plástico acontece após $05\text{h}31\text{min}$. Tal resultado se explica pelo fato do alumínio ter uma condutividade térmica maior do que o plástico (polipropileno), e pode ser percebido pelo maior coeficiente de decaimento na Eq. 3.10 em comparação à Eq. 3.9.

Nas duas situações, observa-se grande ocorrência de comandos liga/desliga após o atingimento da temperatura de 5°C . No ensaio em que foi utilizado o recipiente de alumínio, a frequência foi maior. Tal comportamento pode ser prejudicial ao compressor do refrigerador/motor do refrigerador. Com o intuito de minimizar a incidência de tais comandos, foi estabelecida uma faixa de $1,0^{\circ}\text{C}$ em torno do valor de 5°C ($4,5^{\circ}\text{C} - 5,5^{\circ}\text{C}$), visando diminuir o número de oscilações em torno dele. O código utilizado para os ensaios, que duraram 12h , está no Apêndice 3.

Os resultados obtidos se encontram na Fig. 3.17 e na Fig. 3.18. As equações associadas aos ajustes exponenciais dos gráficos, antes do atingimento do início da histerese, em $4,5^{\circ}\text{C}$, são para os resultados mostrados nas duas figuras, respectivamente:

$$Tp = 25,39e^{(-0,0053t)} \quad (3.9)$$



Figura 3.13: Resfriamento da água a 5°C. Controle liga/desliga. Recipiente de plástico.

$$Tp = 25,74e^{(-0,0067t)} \quad (3.10)$$

Observa-se através das Fig. 3.17 e Fig. 3.18 a redução significativa de ações de liga e desliga. Com a utilização da histerese de 0,5°C em torno de 5°C, obteve-se como temperatura média 4,70 °C, quando foi utilizado o recipiente de plástico, e 4,83 °C, quando foi utilizado o recipiente de alumínio. Tais resultados médios foram próximos dos valores médios obtidos quando não utilizada a faixa de 1°C: 4,90 °C e 4,94 °C.

Com o intuito de analisar o comportamento de aquecimento de um litro d'água no refrigerador após sofrer resfriamento, foram realizados ensaios na *Posição 1* com os recipientes de plástico e de alumínio por 13h30. Por meio da utilização do código contido no Apêndice 4 no Arduino, após a água chegar aos 5°C o refrigerador foi desligado, com a intenção de que a temperatura da água aumentasse com o passar do tempo. Os resultados obtidos estão mostrados graficamente nas Fig. 3.19 e 3.20.

Nota-se uma tendência de crescimento exponencial na fase de aquecimento. Dessa forma, realizam-se ajustes exponenciais nos gráficos que resultaram nas seguintes equações para o recipiente de plástico e de alumínio, respectivamente:

$$Tp = 1,81e^{(0,0021t)} \quad (3.11)$$

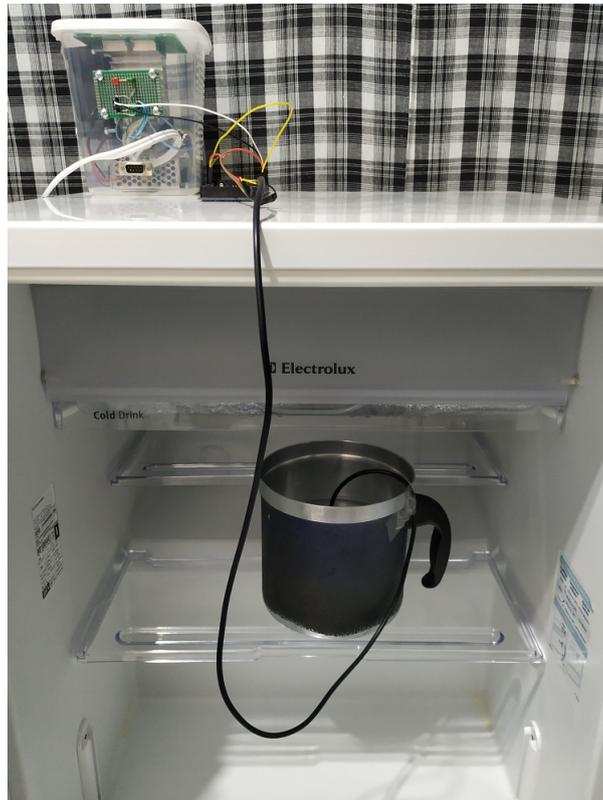


Figura 3.14: Resfriamento da água a 5°C. Controle liga/desliga. Recipiente de plástico.

$$Tp = 1,77e^{(0,0027t)} \quad (3.12)$$

Apesar dos coeficientes de crescimento exponencial obtidos para os dois ensaios serem próximos, houve grande diferença no resultado final. No fim intervalo de 13h30, o ensaio realizado com o uso de recipiente de alumínio conseguiu alcançar a temperatura de 15,44°C após o resfriamento a 5°C, que aconteceu em 4h25 do início do ensaio. Por outro lado, com o uso do recipiente de plástico, a temperatura alcançada foi de 9,62°C após o atingimento de 5°C, que aconteceu em 06h17.

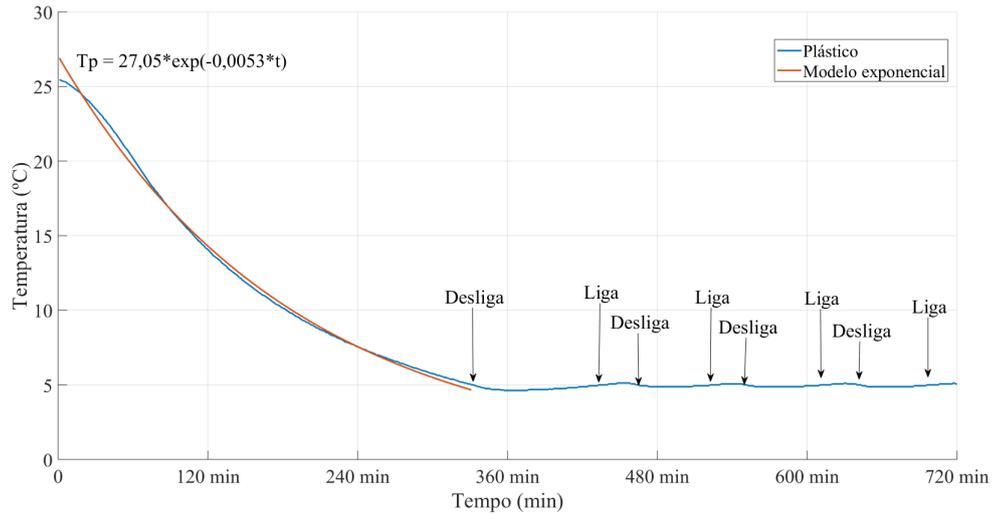


Figura 3.15: Resultado para o resfriamento da água a 5°C. Controle liga/desliga. Recipiente de plástico.

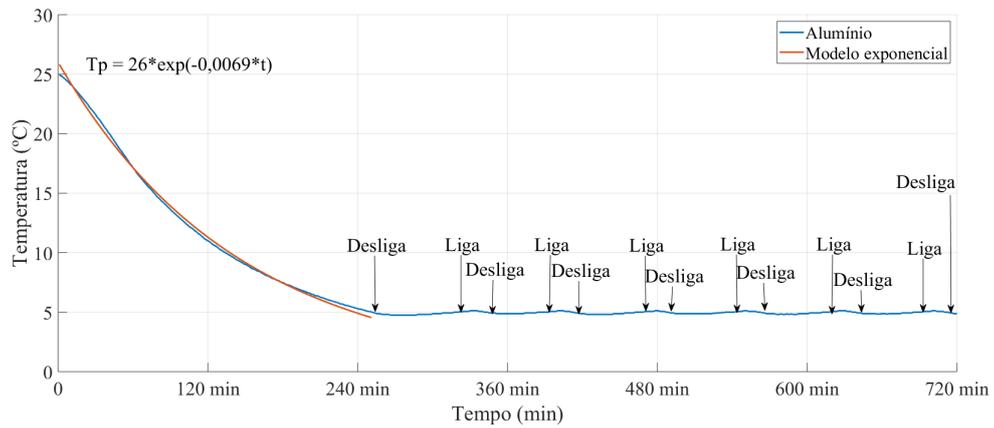


Figura 3.16: Resultado para o resfriamento da água a 5°C. Controle liga/desliga. Recipiente de alumínio.

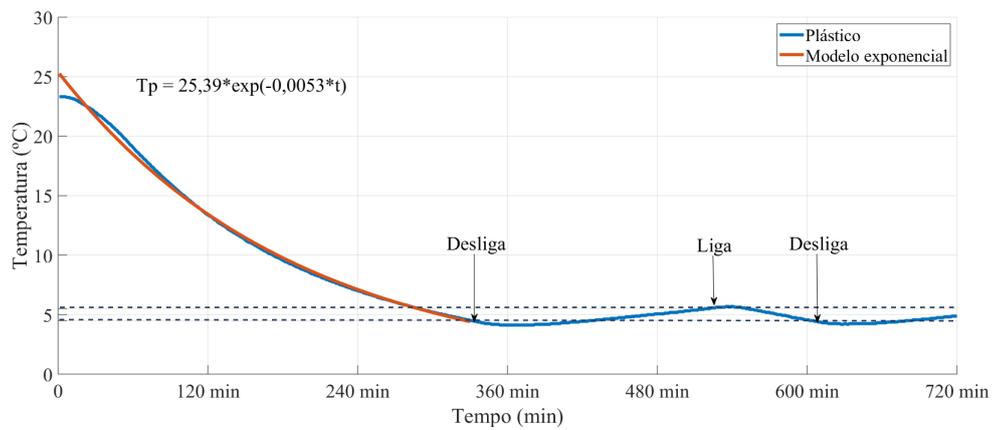


Figura 3.17: Histerese em torno de 5°C. Controle liga/desliga. Recipiente de plástico.

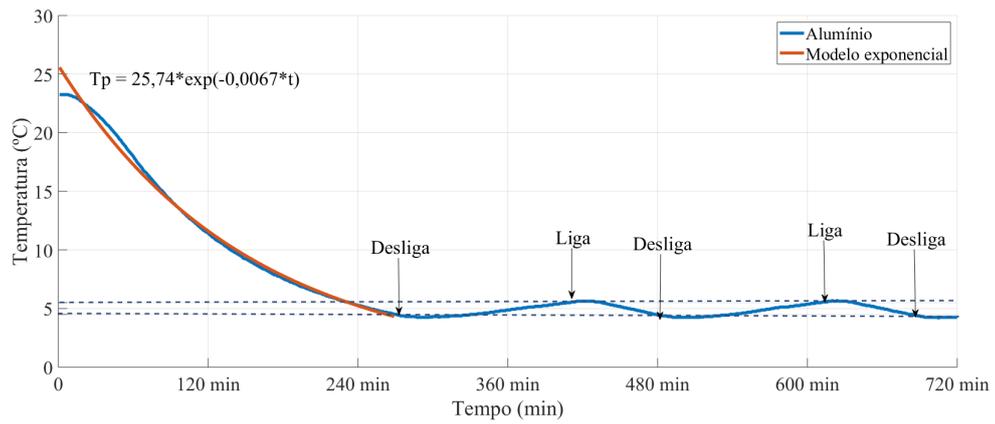


Figura 3.18: Histerese em torno de 5°C. Controle liga/desliga. Recipiente de alumínio.

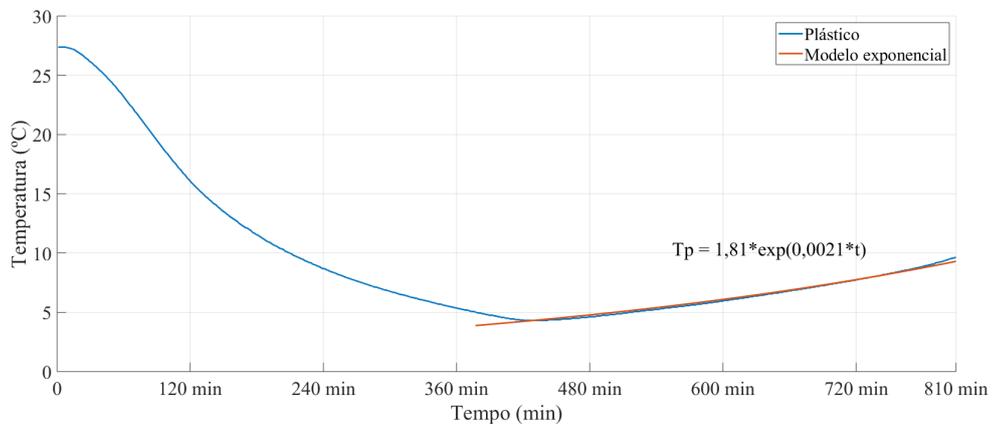


Figura 3.19: Aquecimento no refrigerador após resfriamento. Recipiente de plástico.

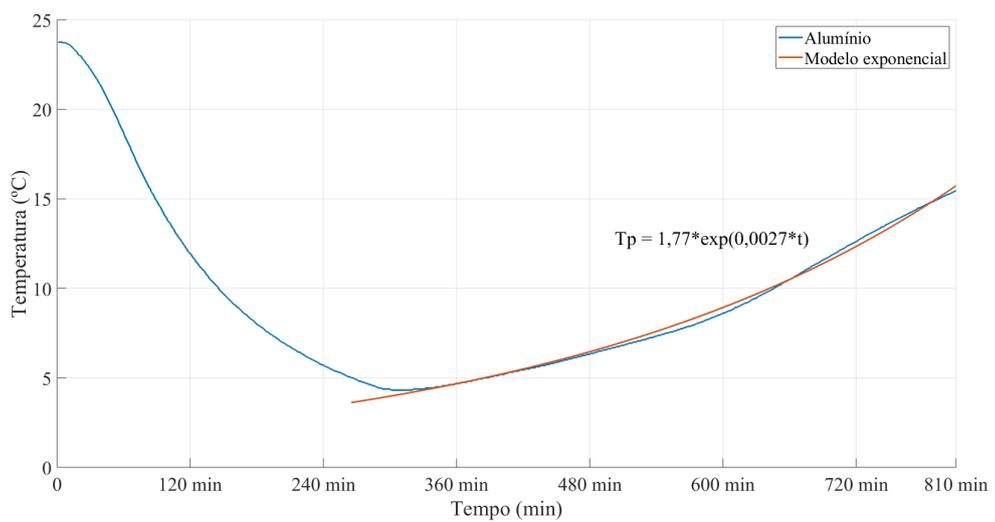


Figura 3.20: Aquecimento no refrigerador após resfriamento. Recipiente de alumínio.

4 ENSAIOS DE CONTROLE DE TEMPERATURA - AQUECIMENTO

Os ensaios de aquecimento tiveram como objetivo a construção de um sistema que permitisse o alcance e a manutenção da temperatura de 5 litros de água em 60°C num intervalo de 15 minutos. Para isso, foram utilizados um ebulidor elétrico acionado por um dispositivo de relé de estado sólido, comandado por um Arduino Mega 2560, que exerceu o papel de controlador proporcional, controlador proporcional-integral (PI) e controlador proporcional-integral-derivativo (PID). Um sensor de temperatura DS18B20 fornecia as medidas de temperatura da água para o Arduino.

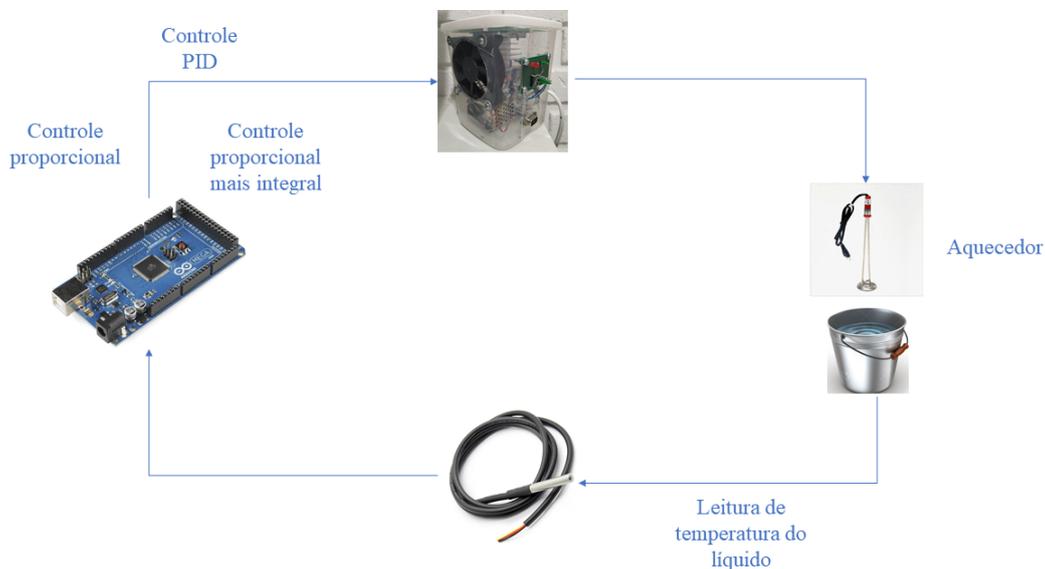


Figura 4.1: Diagrama do sistema de controle de temperatura para aquecimento.

No controle proporcional, o valor de saída do controlador (VS) é proporcional à diferença entre o valor desejado (valor pré-ajustado, *Setpoint*) e o valor da variável que é controlada no processo (VC), no caso deste trabalho, a temperatura. Essa diferença é chamada de erro ($E = Setpoint - VC$). Assim, o valor de saída é ajustado de acordo com as alterações sofridas no sistema. Contudo, o controle proporcional resulta num sinal de desvio chamado erro estacionário ou *offset*, como mostra a Fig. 4.2.

Para contornar o erro estacionário, o controle proporcional é utilizado juntamente com o controle integral, formando o controle proporcional mais integral (PI). O PI tem como característica adicionar à resposta proporcional uma resposta integral, ou seja, em intervalos recorrentes, o valor de saída será corrigido com o acréscimo do erro. A atuação integral é proporcional à amplitude e duração do erro.

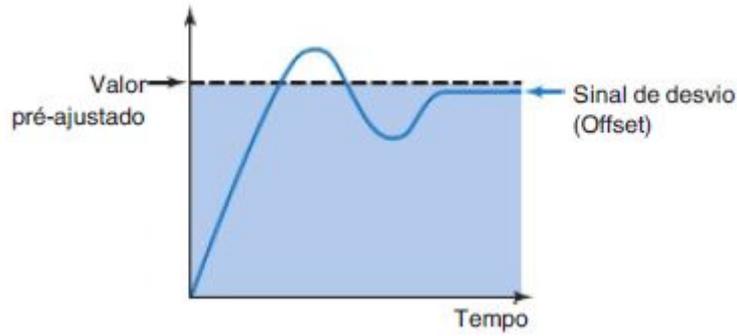


Figura 4.2: Erro estacionário. Fonte: [19].

Um outro controle comumente utilizado com o controle proporcional, é o controle derivativo. O controle derivativo tem como função amenizar as consequências de variações na variável que é controlada no processo, evitando que o valor de saída do controlador aumente ou diminua com muita rapidez. O controle proporcional utilizado juntamente com o controle derivativo é chamado de controle proporcional mais derivativo (PD) e atua somando a variação do erro à resposta proporcional, em intervalos recorrentes. A atuação derivativa é proporcional à velocidade de variação do erro.

Destaca-se que os controles integral e derivativo não podem ser usados dissociados de um controle proporcional.

A combinação dos controles proporcional, integral e derivativo resulta no controle PID. Com isso, tem-se um controle básico, representado pelo controle proporcional, acrescido de correção do erro estacionário, representada pelo controle integral, e amenização de variações abruptas, representada pelo controle derivativo. O controle PID é representado matematicamente pela seguinte expressão:

$$VS = K_p E(t) + K_i \int E(t) dt + K_d \frac{\partial E(t)}{\partial t} \quad (4.1)$$

Em que K_p , K_i , K_d , $E(t)$ são os ganhos das parcelas proporcional, integral e derivativa e o erro, respectivamente.

A determinação dos ganhos das parcelas proporcional, integral e derivativo pode ser realizada através de diversas técnicas. Neste trabalho, a determinação deles será feita de forma manual por meio de método de ensaio e erro.

Em todos os ensaios realizados neste capítulo, o valor de saída do controlador é representado pelo tempo de aplicação de energia no ebulidor elétrico dentro de um intervalo de 30 segundos, a variável controlada no processo é a temperatura da água, e o valor desejado (valor pré-ajustado ou *Setpoint*) é a temperatura de 60°C (Fig. 4.3). As durações dos ensaios foram de 15 minutos.

O primeiro ensaio objetivou a determinação do ganho da parcela proporcional num

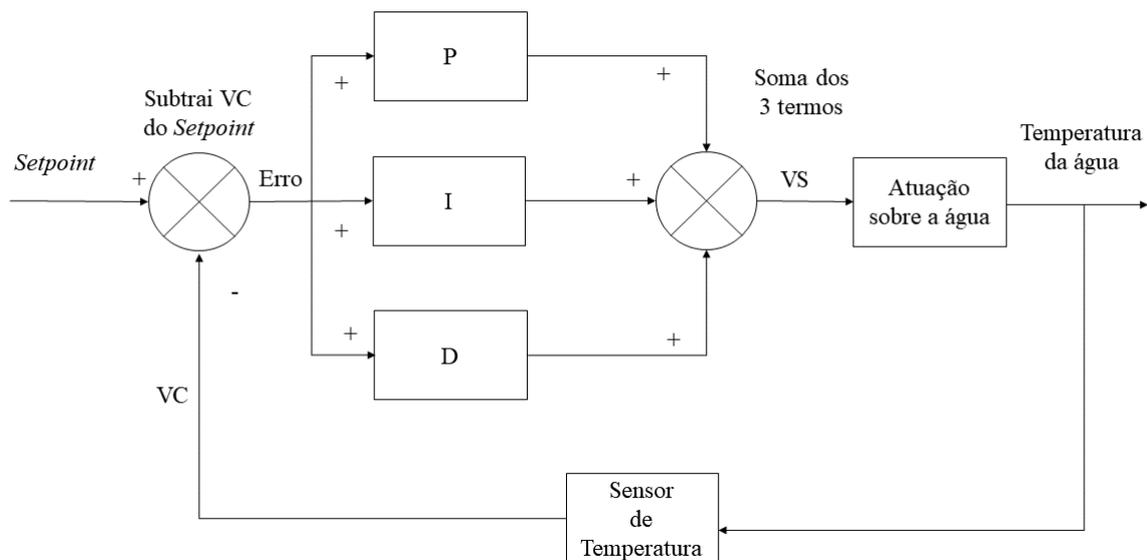


Figura 4.3: Digrama PID para o sistema de aquecimento.

sistema para elevar e manter a temperatura de cinco litros d'água a 60°C, mostrado na Fig. 4.4. Para isso, foi utilizado um controlador proporcional - implementado no Arduino Mega 2560 por meio do código que se encontra no Apêndice 5 - ligado ao dispositivo de relé de estado sólido que por sua vez estava conectado ao ebullidor elétrico. O comportamento de controle proporcional foi determinado ao se manter os valores de ganho integral e derivativo nulos no código.



Figura 4.4: Sistema de controle de temperatura para aquecimento

Iniciou-se escolhendo o valor de $K_p = 2,5$. Os valores obtidos para a variável controlada, a temperatura, estão representados graficamente na Fig. 4.5. Nota-se nela a presença de

erro estacionário de valor em torno de 58,56 °C a partir dos 10 minutos.

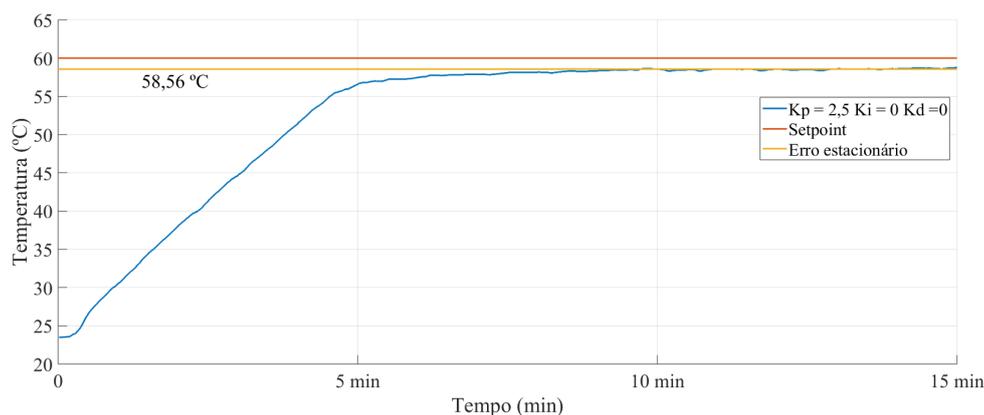


Figura 4.5: Controle proporcional. $K_p = 2,5$ $K_i = 0$ $K_d = 0$.

Na tentativa de se encontrar um melhor valor de K_p que resultasse no controle desejado, o experimento foi repetido para o valor de $K_p = 5$. Os valores obtidos para a variável controlada estão representados graficamente na Fig. 4.6. Observa-se a presença de erro estacionário de valor em torno de 59,31 °C.

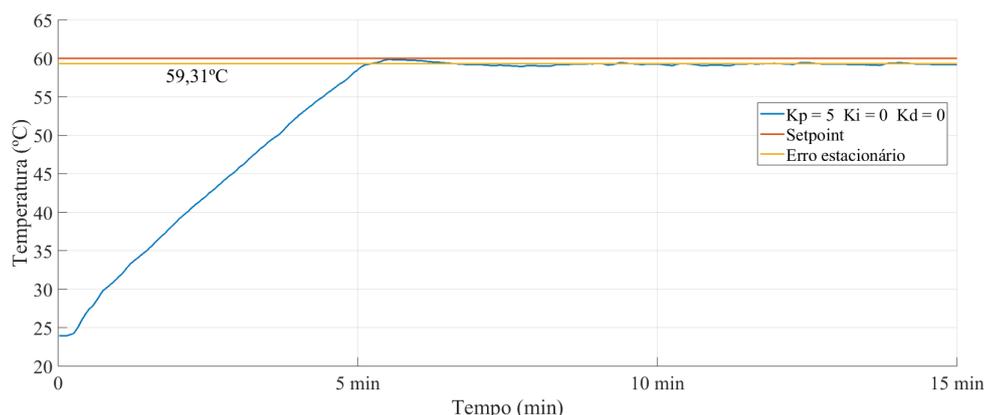


Figura 4.6: Controle proporcional. $K_p = 5$ $K_i = 0$ $K_d = 0$.

O valor de $K_p = 5$ apresentou uma melhor resposta em comparação ao valor de $K_p = 2,5$. Ainda buscando um melhor ajuste, o valor do ganho proporcional foi aumentado para 10. O resultado está representado graficamente na Fig.4.7. Nota-se a presença de uma ultrapassagem de 1,38 °C em relação ao valor do *Setpoint* e a ausência de valor estacionário, que foi substituído por pequenas oscilações. Assim, $K_p = 10$ resultou num pior ajuste em comparação ao $K_p = 5$.

Comparando-se os resultados para $K_p = 10$ e $K_p = 5$, nota-se que o melhor ajuste para K_p se encontra em torno de 5. Ademais, tendo em vista os resultados obtidos para $K_p = 2,5$, entende-se que o valor também terá que ser maior do que 5, pois para valores menores se obtém um maior valor para o erro estacionário. Assim, foi realizado um novo experimento, utilizando $K_p = 6$. Os valores obtidos para a variável controlada estão representados

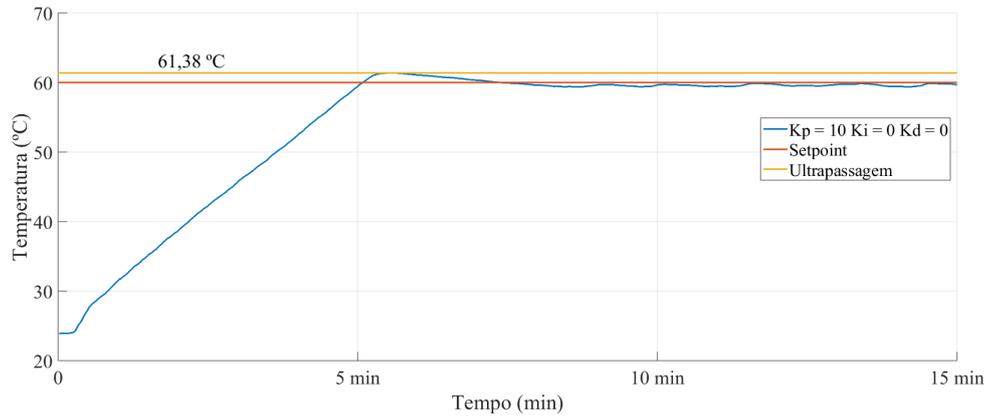


Figura 4.7: Controle proporcional. $K_p = 10$ $K_i = 0$ $K_d = 0$.

graficamente na Fig. 4.8 Observa-se a presença de um erro estacionário em torno de 59,44 °C, juntamente com uma ultrapassagem de 1,25 °C em relação ao *Setpoint*.

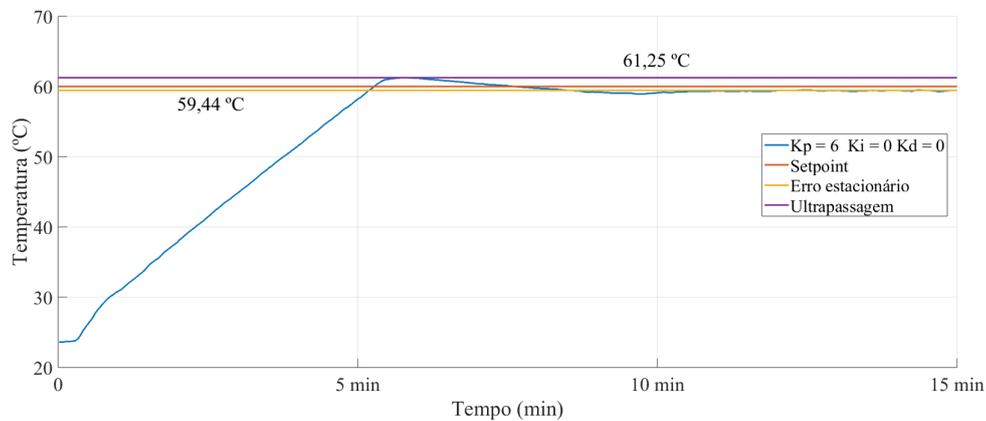


Figura 4.8: Controle proporcional. $K_p = 6$ $K_i = 0$ $K_d = 0$.

Com base nos quatros resultados obtidos, conclui-se que, dentre os valores ensaiados, $K_p = 5$ é o mais adequado para o sistema.

Tendo em vista a presença do erro estacionário associado ao melhor ajuste para o controle proporcional ($K_p = 5$), foram realizados ensaios com o objetivo de eliminá-lo. Para isso, incluiu-se o controle integral ao controle proporcional, formando um controle proporcional mais integral (PI). O PI foi implementado no Arduino Mega 2560 por meio do código que se encontra no Apêndice 5, colocando o valor de $K_p = 5$ e mantendo o valo de K_d nulo. Em seguida se procedeu à determinação do ganho da parcela integral, o K_i .

Primeiramente, foi escolhido o valor de $K_i = 2,5$. Assim o controle proporcional mais integral ficou configurado com $K_p = 5$, $K_i = 2,5$ e $K_d = 0$. O resultado obtido está representado graficamente na Fig. 4.9 e mostrou-se ser insatisfatório. Nota-se uma ultrapassagem de 5,81 °C associada a um erro estacionário, superior ao *Setpoint*, em torno de 62,56 °C.

Posteriormente, o valor de K_i foi aumentado para 10. O resultado encontra-se na Fig.

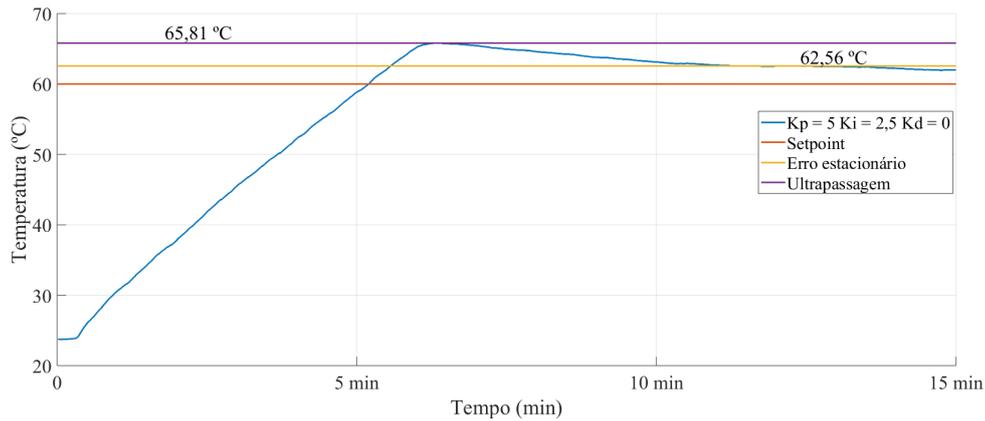


Figura 4.9: Controle proporcional mais integral. $K_p = 5$ $K_i = 2,5$ $K_d = 0$.

4.10. Observa-se uma alteração em relação ao erro estacionário, que deixou de existir. Entretanto, uma acentuada ultrapassagem de $65,75\text{ }^\circ\text{C}$ se manteve presente. Levando-se em consideração a melhora com o aumento do ganho integral, foram realizados ensaios com $K_i = 100, 500, 1000$ e 5000 , cujas representações gráficas são mostradas na Fig. 4.11.

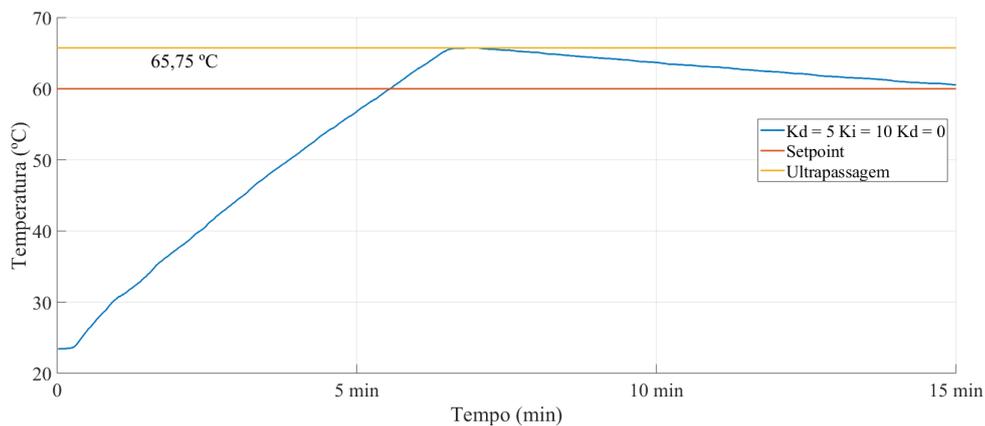


Figura 4.10: Controle proporcional mais integral. $K_p = 5$ $K_i = 10$ $K_d = 0$.

Com base na Fig. 4.11, verifica-se que os melhores ajustes, que perpassaram pelo *Setpoint* por mais tempo, foram para $K_i = 500$ e 1000 , sendo o primeiro o melhor. Tendo isso em vista, realizou-se um ensaio com o valor intermediário de $K_i = 700$. O gráfico do resultado comparado com o gráfico para $K_i = 500$ é apresentado na Fig. 4.12. Nota-se que $K_i = 500$ se manteve como melhor ajuste para o ganho integral dentro do intervalo de tempo de análise, não obstante a ultrapassagem de $4,19\text{ }^\circ\text{C}$.

Comparado ao controle proporcional ($K_p = 5, K_i = 0, K_d = 0$), o controle proporcional mais integral ($K_p = 5, K_i = 500, K_d = 0$) resultou num pior controle devido, em grande parte, à ultrapassagem em relação ao *Setpoint*. Assim, com o intuito de de aproximar o controle proporcional mais integral ao controle desejado, que elevasse e mantivesse a temperatura da água em 60°C foi introduzido o controle derivativo, formando o controle PID. A atuação do derivativo busca diminuir ou eliminar a ultrapassagem, bem como

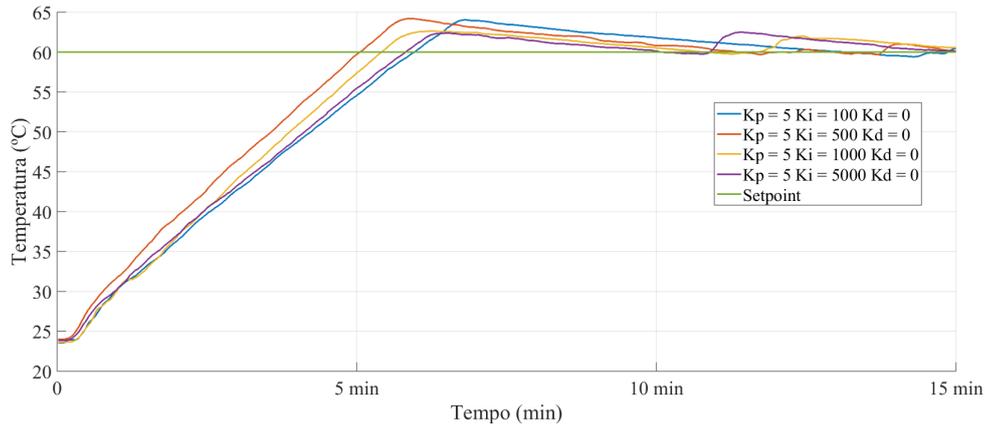


Figura 4.11: Controle proporcional mais integral. $K_p = 5$ $K_i = 100, 500, 1000$ e 5000 $K_d = 0$.

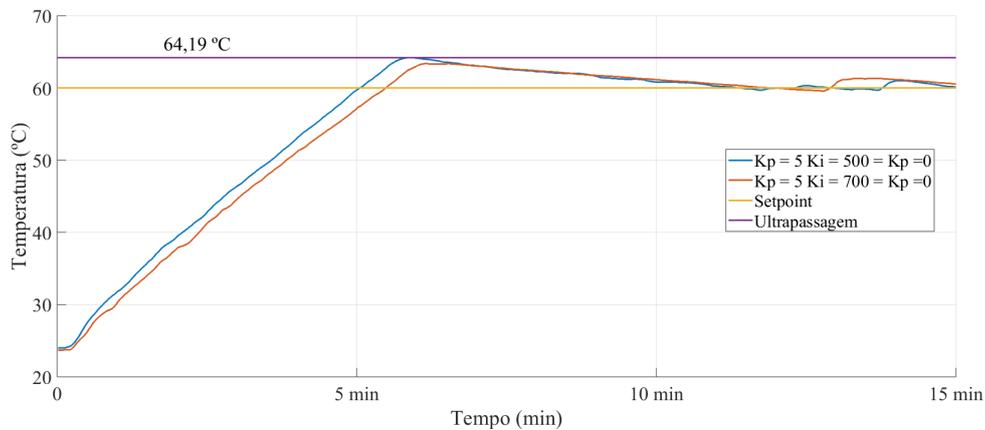


Figura 4.12: Controle proporcional mais integral. $K_p = 5$ $K_i = 500$ e 700 $K_d = 0$.

oscilações.

Implementou-se o controlador PID no Arduino Mega 2560 por meio do código que se encontra no Apêndice 5, colocando os valores de $K_p = 5$ e $K_i = 500$.

Inicialmente, escolheu-se o resultado de 10 para o ganho derivativo, seguido de 100. O resultado está graficamente representado na Fig.4.13. Nota-se que ambos os gráficos mostram ajustes ruins para os dois ganhos, sendo $K_d = 100$ o pior.

Considerando-se os resultados mostrados na Fig.4.13, entende-se que os melhores ajustes advêm de ganhos derivativos menores do que 10. Assim, realizou-se um ensaio com $K_d = 5$, no qual verificou-se que um melhor ajuste seria com um valor menor ainda de ganho. Com isso, foram realizados ensaios com $K_d = 3$ e $K_d = 2,5$. Os resultados de tais ensaios se encontram na Fig 4.14. Observa-se que entre os três ensaios, o melhor ajuste foi para $K_d = 3$ e o pior para $K_d = 2,5$.

Na tentativa de melhorar ainda mais o resultado foi realizado o ensaio para $K_d = 3,5$. Comparando-o com o ensaio para $K_d = 3$, verifica-se que o último resultou num melhor ajuste de controle para o controlador PID possuindo ligeiras oscilações com valor máximo de

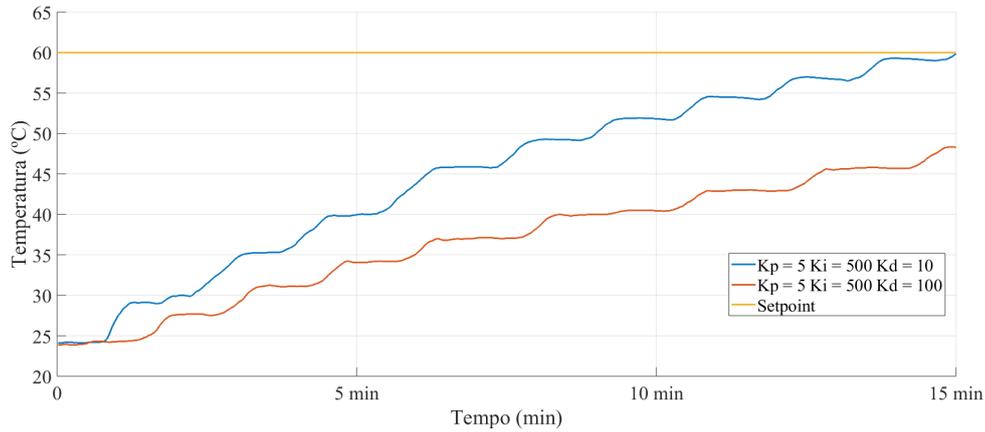


Figura 4.13: Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 10$ e 100 .

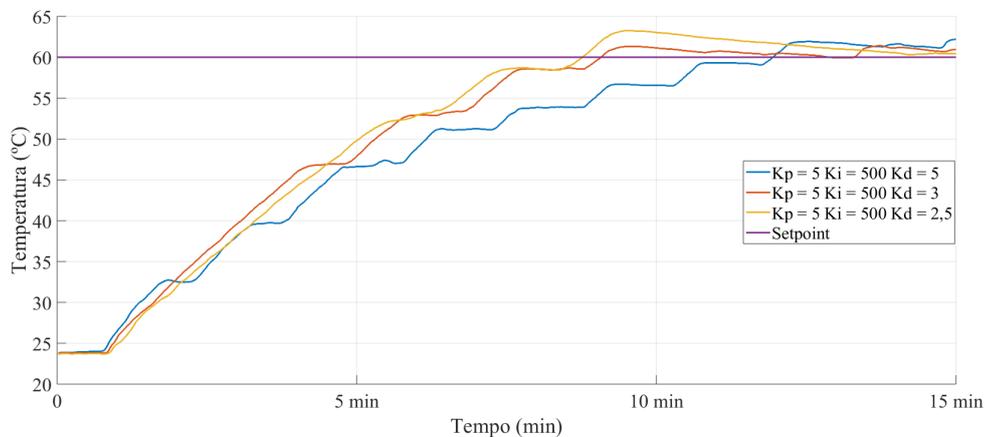


Figura 4.14: Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 2,5$ e 3 .

0,94 °C de ultrapassagem, em relação ao *Setpoint*, e com valor médio de 60,39 °C, conforme mostrado na Fig. 4.15.

Dos resultados obtidos, verifica-se que os controles que mais se aproximaram do controle desejado para o sistema foram o controle proporcional ($K_p = 5$ $K_i = 0$ $K_d = 0$) e o controle PID ($K_p = 5$ $K_i = 500$ $K_d = 3,5$) Fig.4.16 .

Nota-se que o controle PID atinge a temperatura alvo, o *Setpoint*, em 8,75 minutos, enquanto o controle proporcional se aproxima dela (59,88°C), não a atingindo, em 5,50 minutos. O controlador proporcional mantém a temperatura em torno de 59,31°C, uma diferença de 0,69 °C abaixo da temperatura desejada, e o controlador PID mantém uma temperatura média de 0,39°C acima, com a presença de pequenas oscilações.

Por fim, as Figuras 4.17 e 4.18 mostram o tempo de funcionamento do aquecedor elétrico a cada intervalo de 30s dentro do período de 15 minutos no uso do controlador proporcional e do controlador PID. Como mencionado anteriormente, o valor de saída do controlador corresponde ao tempo de aplicação de energia no aquecedor elétrico dentro de um intervalo de 30 segundos. Como pode ser observado nas figuras, no controle PID o aquecedor é menos

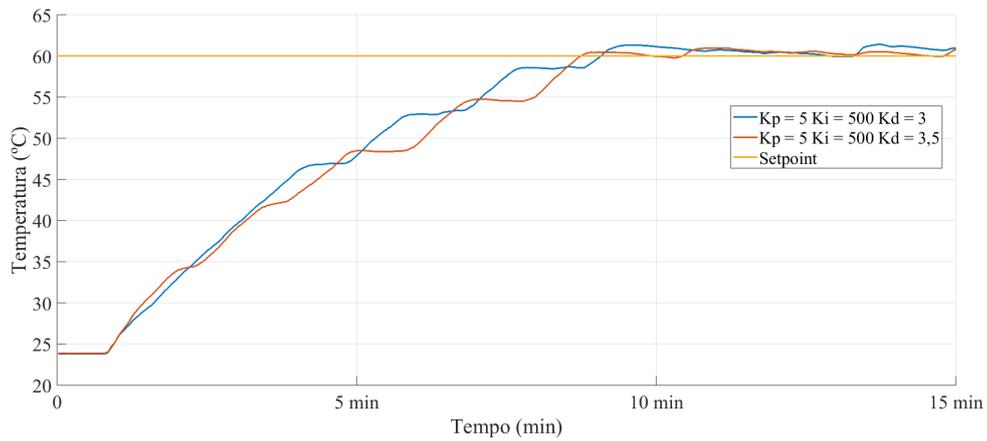


Figura 4.15: Controle PID. $K_p = 5$ $K_i = 500$ $K_d = 3$ e $3,5$.

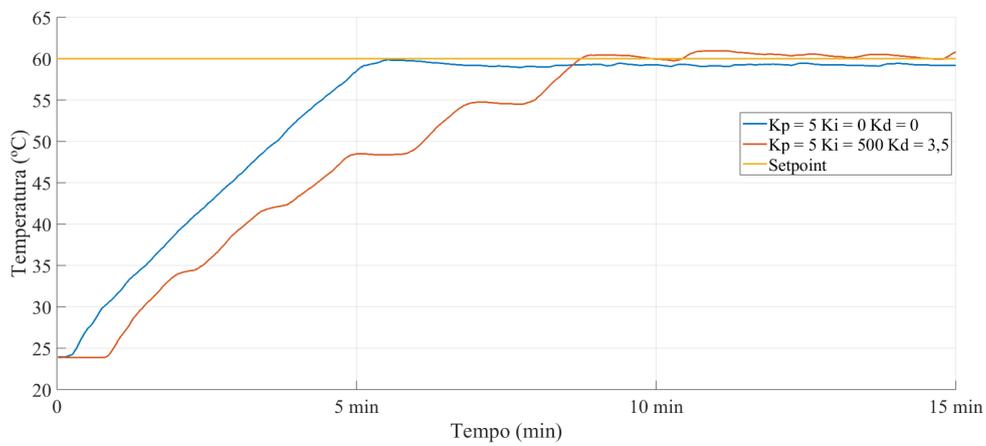


Figura 4.16: Melhores controles: Controle Proporcional e Controle PID.

atuante do que no controle proporcional. Isso se reflete no crescimento menos acentuado de sua curva de aquecimento, conforme mostrado na Fig. 4.16.

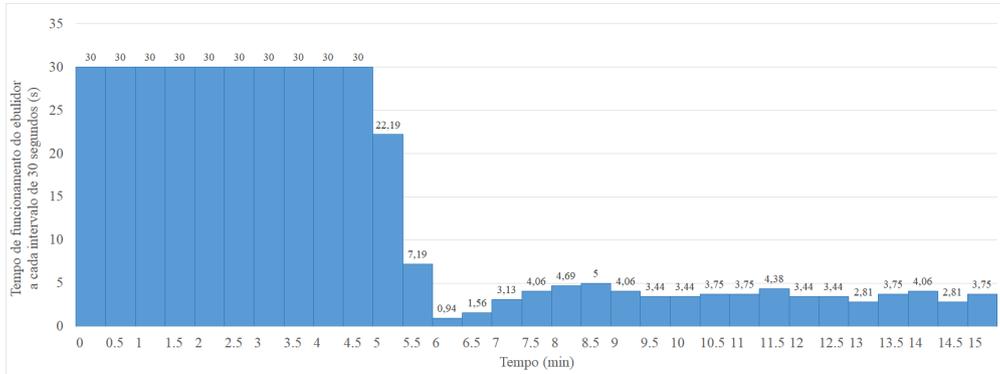


Figura 4.17: Tempo de funcionamento do ebulidor elétrico a cada intervalo de 30 segundos. Controlador Proporcional.

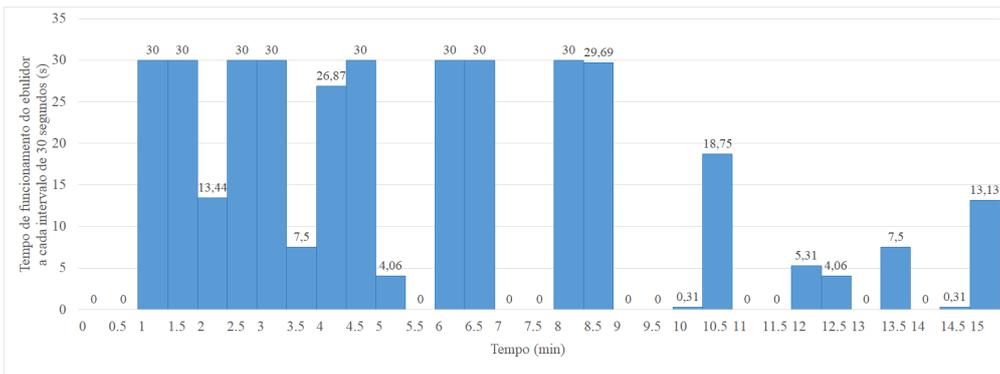


Figura 4.18: Tempo de funcionamento do ebulidor elétrico a cada intervalo de 30 segundos. Controlador PID.

5 CONCLUSÃO

O processo produtivo da cerveja é caracterizado por processos de resfriamento e aquecimento em diferentes faixas de temperatura. A qualidade do produto final está sensivelmente associada à utilização precisa delas. Diferentemente das grandes indústrias cervejeiras, as microcervejarias, muitas vezes caracterizadas pela produção por *hobby*, não detêm de instrumental adequado para o controle de temperatura, procedendo muitas vezes de forma manual.

Diante disso, o trabalho objetivou a construção de sistemas de controle de resfriamento e aquecimento de baixa complexidade. O sistema de controle de resfriamento, baseado na refrigeração por compressão de vapor, foi composto por um refrigerador, um dispositivo de relé de estado sólido, um sensor de temperatura DS18B20 e um recipiente. O sistema de controle de aquecimento foi composto por um ebulidor elétrico, um dispositivo de relé de estado sólido, um sensor de temperatura DS1820 e um recipiente. Ambos os sistemas foram controlados por um Arduino Mega 2560 que exerceu o papel de controlador liga/desliga, controlador proporcional, controlador proporcional mais integral e controlador PID, e armazenador das temperaturas colidas pelo sensor de temperatura.

O sistema de controle de resfriamento foi fundamentado no controle liga/desliga. Antes da sua implementação, foram realizados ensaios visando a melhor forma de uso do refrigerador. Foram analisados a variação de temperatura em dois pontos internos distintos do eletrodoméstico, bem como qual melhor tipo de material de recipiente a ser utilizado.

Para a construção do sistema de controle de aquecimento foram analisados os controles proporcional, proporcional mais integral e PID. A utilização dos dois últimos foi motivada pela procura de melhorias no controle proporcional. Os controles proporcional e PID foram os que apresentaram melhores resultados.

Nos ensaios realizados, o sistema de resfriamento visou o resfriamento e manutenção de 1 litro de água a 5°C. Para tal, obteve-se resultados satisfatórios com valores médios de temperatura de 4,905°C e 4,937°C, no caso da utilização de recipiente plástico e de alumínio, respectivamente. Também foram observadas oscilações máximas de 0,125°C e 0,375°C, acima e abaixo do valor de referência para o primeiro recipiente, e de 0,125°C e 0,250°C para o segundo. Todavia, foi constatado que o motor/compressor do refrigerador foi utilizado de forma que pode provoca danos a ele no médio e longo prazo. Quanto ao tempo para atingimento da temperatura, quando foi usado o recipiente de plástico foi de 05h31min, e com o recipiente de alumínio de 4h11min.

O sistema de aquecimento teve como objetivo o aquecimento e manutenção de 5 litros de água a 60°C. Realizaram-se ensaios com o intuito de encontrar os melhores valores de ganho proporcional, integral e derivativo a serem utilizados nos controles proporcional,

proporcional mais integral e PID. Os controles proporcional e PID, os melhores para o sistema, trouxeram resultados satisfatórios. O primeiro manteve a temperatura da água a $0,69^{\circ}\text{C}$ abaixo da temperatura desejada, e o segundo, $0,39^{\circ}\text{C}$ acima. O controle proporcional atinge o temperatura de 60°C em 5,5 minutos e o controle PID, em 8,75 minutos.

A determinação dos ganhos foi feita de forma manual e individual, por meio de ensaios e erros. Um melhor determinação dos valores se daria com a utilização de inteligência artificial no controlador que reajustaria os parâmetros conforme preciso.

Não obstante a aproximação de características físicas com a água, sugerem-se ensaios utilizando o mosto e a própria cerveja, para maior compatibilidade com a realidade. Dentro desse enfoque, também sugerem-se as simulações das rampas de temperatura presentes no processo produtivo, bem como ensaios envolvendo maiores quantidades de líquido com equipamentos de refrigeração e aquecimento de maiores capacidades.

REFERÊNCIAS BIBLIOGRÁFICAS

- 1 XIAODONG, Z.; JIE, Z.; KE, L. Design and implementation of control system for beer fermentation process based on simatic plc. *27th Chinese Control and Decision Conference (CCDC)*, v. 1, p. 5653–5656, 2015.
- 2 BARBOSA, D. *As 5 maiores cervejarias do mundo*. Disponível em: <<https://exame.abril.com.br/negocios/as-5-maiores-cervejarias-do-mundo/>>.
- 3 CERVBRAIL. *Anuário 2015*. Disponível em: <http://cervbrasil.org.br/arquivos/ANUARIO_CB_2015_WEB.pdf>.
- 4 GLOBO.COM. *Os povos que mais bebem cerveja no mundo*. Disponível em: <<http://g1.globo.com/especial-publicitario/somos-todos-ervejeiros/infografico/os-povos-que-mais-bebem-cerveja-no-mundo.html>>.
- 5 BEERART. *Um estudo sobre o consumo de cerveja no Brasil*. Disponível em: <<https://revistabeerart.com/news/estudo-consumo-cerveja-brasil>>.
- 6 SEBRAE. *Microcervejaria*. Disponível em: <www.sebrae.com.br/sites/PortalSebrae/ideias/como-montar-uma-microcervejaria,8f387a51b9105410VgnVCM1000003b74010aRCRD>.
- 7 ROSA, N.; AFONSO, J. A química da cerveja. *Química Nova Escola*, v. 37, p. 98–105, 2015.
- 8 LIMA, R. *Sistema Controlado para resfriamento de mosto de cerveja*. Disponível em: <<https://rodrigolima91.wordpress.com/sistema-controlado-para-resfriamento-de-mosto-de-cerveja/>>.
- 9 EMBRACO. *Pressões de trabalho dos gases refrigerantes nos sistemas domésticos*. Disponível em: <<http://refrigerationclub.com/pt-br/pressoes-de-trabalho-dos-gases-refrigerantes-nos-sistemas-domesticos/>>.
- 10 ÇPOREL, Y.; BOLES, M. *Termodinâmica*. [S.l.]: Bookmann Editora, 2013.
- 11 OSCARAS. *Como funciona o Ar Condicionado?* Disponível em: <<https://oscarasdoarcondicionado.com.br/como-funciona-o-ar-condicionado/>>.
- 12 EMBRACO. *Termodinâmica: Entenda o que é esse ramo da ciência e as suas aplicações na refrigeração*. Disponível em: <<https://refrigerationclub.com/pt-br/termodinamica-entenda-o-que-e-esse-ramo-da-ciencia-e-as-suas-aplicacoes-na-refrigeracao/>>.
- 13 ARDUINO. *Arduino Mega 2560 Rev3*. Disponível em: <<https://store.arduino.cc/usa/arduino-mega-2560-rev3>>.
- 14 FILIPEFLOP. *Placa MEGA 2560 R3*. Disponível em: <<https://www.filipeflop.com/produto/placa-mega-2560-r3-cabo-usb-para-arduino/>>.
- 15 MAXIMIINTEGRATED. *Datasheet DS18B20*. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>.
- 16 COMPONENTS101. *DS18B20 Temperature Sensor*. Disponível em: <<https://components101.com/sensors/ds18b20-temperature-sensor>>.
- 17 ESTEVÃO, V. *Efeito Joule*. Disponível em: <<https://www.efeitojoule.com/2008/09/autor-professor-vanks-estevao.html>>.

18 PANELASECIA. *Ebulidor/Mergulhão*. Disponível em: <<https://www.panelasecia.com.br/85-ebulidormergulhao-reforcado-aluminio-cherubino.html>>.

19 PETRUZELLA, F. D. *Controladores lógicos programáveis*. [S.l.]: AMGH Editora Ltda., 2014.

APÊNDICES

```
// Medição da temperatura do frigobar com um sensor DS18B20 por 6h
// Douglas Vasconcelos C. Alves

#include <OneWire.h>
#include <DallasTemperature.h>
#include <EEPROM.h>

#define TRUE 1

OneWire pino(5);
DallasTemperature barramento(&pino);
DeviceAddress sensor;

void setup() {

  Serial.begin(9600);
  barramento.begin();
  barramento.getAddress(sensor, 0);

  //Inicializar o TC1
  TCCR1A = 0; //WGM11=WGM10=0
  //Modo CTC e CLK/1024
  TCCR1B = (1<<WGM12)|(1<<CS12)|(1<<CS10);
  OCR1A = 15624;

  pinMode(2, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
}

void loop() {

  uint8_t seg = 0; //Contador
  float temperatura;
  int armazena_temp = digitalRead(2);
  int mostra_temp = digitalRead(4);
  int endereco = 0;
  float f = 0.00f; //Variável para armazenar os dados lidos da EEPROM

  //Ler temperatura e grava na EEPROM
  if (armazena_temp == LOW){

    Serial.println("Ler temperatura e grava na EEPROM: ");

    while(TRUE){
      while ((TIFR1&(1<<OCF1A))==0); //Esperar OCF1A = 1
      TIFR1 = (1<<OCF1A); //Apagar OCF1A
      seg++;
      if (seg == 30){
        seg = 0;
        barramento.requestTemperatures();
        temperatura = barramento.getTempC(sensor);

        EEPROM.put(endereco, temperatura);
        endereco += sizeof (float); //Move o endereço para o próximo byte depois do armazenamento da temperatura
      }
    }
  }

  //Mostra as temperaturas
  if (mostra_temp == LOW){

    Serial.println("Mostra as temperaturas armazenadas na EEPROM: ");

    while (endereco < 2880){
      EEPROM.get(endereco,f);
      Serial.println(f,3);
      endereco += sizeof (float);
    }
  }
}
```

Apêndice 1: Código para coleta de dados de temperatura a cada 30 segundos

```

//Douglas Vasconcelos C. Alves

#include <OneWire.h>
#include <DallasTemperature.h>
#include <EEPROM.h>

#define TRUE 1

OneWire pino(5);
DallasTemperature barramento(&pino);
DeviceAddress sensor;

void setup() {

  Serial.begin(9600);
  barramento.begin();
  barramento.getAddress(sensor, 0);

  //Inicializar o TC1
  TCCR1A = 0; //WGM11-WGM10=0
  //Modo CTC e CLK/1024
  TCCR1B = (1<<WGM12)|(1<<CS12)|(1<<CS10);
  OCR1A = 15624;

  pinMode(2, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  pinMode(7,OUTPUT);
}

void loop() {

  uint8_t seg = 0; //Contador
  float temperatura;
  int armazena_temp = digitalRead(2);
  int mostra_temp = digitalRead(4);
  int endereco = 0;
  float f = 0.00f; //Variável para armazenar os dados lidos da EEPROM

  //Ler temperatura e grava na EEPROM
  if (armazena_temp == LOW){

    Serial.println("Ler temperatura e grava na EEPROM: ");

    while(TRUE){
      while ((TIFR1&(1<<OCF1A))==0); //Esperar OCF1A = 1
      TIFR1 = (1<<OCF1A); //Apagar OCF1A
      seg++;
      if (seg == 60){
        seg = 0;
        barramento.requestTemperatures();
        temperatura = barramento.getTempC(sensor);
        EEPROM.put(endereco, temperatura);
        endereco += sizeof (float); //Move o endereço para o próximo byte depois do armazenamento da temperatura
        if(temperatura>5){
          digitalWrite(7,HIGH);
        }
        else{
          digitalWrite(7,LOW);
        }
      }
    }
  }
  //Mostra as temperaturas
  if (mostra_temp == LOW){

    Serial.println("Mostra as temperaturas armazenadas na EEPROM: ");

    while (endereco < 2880){
      EEPROM.get(endereco,f);
      Serial.println(f,3);
      endereco += sizeof (float);
    }
  }
}

```

Apêndice 2: Código para controle liga/desliga.

```

//Douglas Vasconcelos C. Alves
#include <OneWire.h>
#include <DallasTemperature.h>
#include <EEPROM.h>

#define TRUE 1

OneWire pino(5);
DallasTemperature barramento(&pino);
DeviceAddress sensor;

int histerese;
void setup() {

  Serial.begin(9600);
  barramento.begin();
  barramento.getAddress(sensor, 0);
  //Inicializar o TC1
  TCCR1A = 0; //WGM11=WGM10=0
  //Modo CTC e CLK/1024
  TCCR1B = (1<<WGM12)|(1<<CS12)|(1<<CS10);
  OCR1A = 15624;
  pinMode(2, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  pinMode(7, OUTPUT);
  histerese = 0; //Histerese desligada
}

void loop() {

  uint8_t seg = 0; //Contador
  float temperatura;
  int armazena_temp = digitalRead(2);
  int mostra_temp = digitalRead(4);
  int endereco = 0;
  float f = 0.00f; //Variável para armazenar os dados lidos da EEPROM

  //Ler temperatura e grava na EEPROM

  if (armazena_temp == LOW){

    Serial.println("Ler temperatura e grava na EEPROM: ");
    while(TRUE){
      while ((TIFR1&(1<<OCF1A))==0); //Esperar OCF1A = 1
      TIFR1 = (1<<OCF1A); //Apagar OCF1A
      seg++;
      if (seg == 60){
        seg = 0;
        barramento.requestTemperatures();
        temperatura = barramento.getTempC(sensor);
        EEPROM.put(endereco, temperatura);
        endereco += sizeof (float); //Move o endereço para o próximo byte depois do armazenamento da temperatura
        if((temperatura>4.5) && (histerese==0)){
          digitalWrite(7,HIGH);
        }
        if(temperatura<4.5){
          digitalWrite(7,LOW);
          histerese = 1; //Histerese ligada
        }
        if(temperatura>5.5){
          digitalWrite(7,HIGH);
        }
      }
    }
  }

  //Mostra as temperaturas
  if (mostra_temp == LOW){
    Serial.println("Mostra as temperaturas armazenadas na EEPROM: ");
    while (endereco < 2880){
      EEPROM.get(endereco, f);
      Serial.println(f,3);
      endereco += sizeof (float);
    }
  }
}

```

Apêndice 3: Código para controle liga/desliga - Histerese em torno de 5 graus Celsius.

```

//Douglas Vasconcelos C. Alves

#include <OneWire.h>
#include <DallasTemperature.h>
#include <EEPROM.h>

#define TRUE 1

OneWire pino(5);
DallasTemperature barramento(&pino);
DeviceAddress sensor;

int flag;
void setup() {

  Serial.begin(9600);
  barramento.begin();
  barramento.getAddress(sensor, 0);
  //Inicializar o TC1
  TCCR1A = 0; //WGM11=WGM10=0
  //Modo CTC e CLK/1024
  TCCR1B = (1<<WGM12)|(1<<CS12)|(1<<CS10);
  OCR1A = 15624;
  pinMode(2, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  pinMode(7, OUTPUT);
  flag = 1;
}

void loop() {

  uint8_t seg = 0; //Contador
  float temperatura;
  int armazena_temp = digitalRead(2);
  int mostra_temp = digitalRead(4);
  int endereco = 0;
  float f = 0.00f; //Variável para armazenar os dados lidos da EEPROM
  //Ler temperatura e grava na EEPROM
  if (armazena_temp == LOW){

    Serial.println("Ler temperatura e grava na EEPROM: ");

    while(TRUE){
      while ((TIFR1&(1<<OCF1A))==0); //Esperar OCF1A = 1
      TIFR1 = (1<<OCF1A); //Apagar OCF1A
      seg++;
      if (seg == 60){
        seg = 0;
        barramento.requestTemperatures();
        temperatura = barramento.getTempC(sensor);
        Serial.println(temperatura);
        Serial.println(endereco);
        EEPROM.put(endereco, temperatura);
        endereco += sizeof (float); //Move o endereço para o próximo byte depois do armazenamento da temperatura
        if ((temperatura>4.5) && (flag==1)){
          digitalWrite(7,HIGH); }
        else{
          digitalWrite(7,LOW);
          flag = 0;
        }
      }
    }
  }

  //Mostra as temperaturas

  if (mostra_temp == LOW){

    Serial.println("Mostra as temperaturas armazenadas na EEPROM: ");

    while (endereco < 3240){
      EEPROM.get(endereco,f);
      Serial.println(f,3);
      endereco += sizeof (float);
    }
  }
}

```

Apêndice 4: Código para recomposição da temperatura da água no refrigerador após resfriamento.

```

#include <PID_v1.h>
#include <OneWire.h>
#include <DallasTemperature.h>

#define aquecedor 7 // Pino associado ao acionamento do mergulhão

double Setpoint, Input, Output;
double Kp=5, Ki=500, Kd=3; // Parâmetros PID
float temperatura;
uint8_t seg = 0;

PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);
OneWire pino(5);
DallasTemperature barramento(&pino);
DeviceAddress sensor;

void setup() {
  Serial.begin(9600);
  barramento.begin();
  barramento.getAddress(sensor,0);
  //Inicializar o TC1
  TCCR1A = 0; //WGM11=WGM10=0
  //Modo CTC e CLK/1024
  TCCR1B = (1<<WGM12)|(1<<CS12)|(1<<CS10);
  OCR1A = 15624;

  Setpoint = 60.0; // Set point PID
  myPID.SetOutputLimits(0, 30); // PID varia entre 0 ao equivalente de 30s para o PID
  //myPID.SetSampleTime(janela);
  myPID.SetMode(AUTOMATIC);

  pinMode(aquecedor, OUTPUT);
  digitalWrite(aquecedor, LOW);
  Serial.print(" ");
  Serial.print("Tp.");
  Serial.print("\t");
  Serial.print("\t");
  Serial.print("\t");
  Serial.print("\t");
  Serial.println("Output");
  medicoes();
}

void loop() {

while ((TIFR1&(1<<OCF1A))==0); //Esperar OCF1A = 1
TIFR1 = (1<<OCF1A); //Apagar OCF1A
seg++;

barramento.requestTemperatures();
temperatura = barramento.getTempC(sensor);
Serial.print(" ");
Serial.print(temperatura);
Serial.print("\t");
Serial.print(seg);
Serial.print("\t");
Serial.println(Output);

if (seg <= Output){
  digitalWrite(aquecedor, HIGH);
}
else{
  digitalWrite(aquecedor, LOW);
}
if (seg == 30){ //Atingindo 30s
  seg = 0;
  medicoes();
}
}

void medicoes (){
barramento.requestTemperatures();
temperatura = barramento.getTempC(sensor);
Input = (double)temperatura;
myPID.Compute();
}

```

Apêndice 5: Código dos controle proporcional, integral, derivativo, proporcional mais integral e PID.