

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

DEPARTAMENTO ACADÊMICO DE ELETRÔNICA

CURSO DE ENGENHARIA ELETRÔNICA

GUSTAVO MENDES GONÇALVES

JORGE LEANDRO FRANCISCO

**AUTOMAÇÃO DE PROCESSO DE BRASSAGEM PARA
FABRICAÇÃO DE CERVEJA ARTESANAL**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2019

GUSTAVO MENDES GONÇALVES

JORGE LEANDRO FRANCISCO

**AUTOMAÇÃO DE PROCESSO DE BRASSAGEM PARA
FABRICAÇÃO DE CERVEJA ARTESANAL**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Engenharia Industrial Elétrica com ênfase em Eletrônica e Telecomunicações do Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Orientador: Professor Doutor Paulo Roberto Brero de Campos.

CURITIBA

2019

GUSTAVO MENDES GONÇALVES

JORGE LEANDRO FRANCISCO

AUTOMAÇÃO DE PROCESSO DE BRASSAGEM PARA FABRICAÇÃO DE CERVEJA ARTESANAL

Este Trabalho de Conclusão de Curso de Graduação foi apresentado como requisito parcial para obtenção do título de Engenheiro Eletrônico, do curso de Engenharia Eletrônica do Departamento Acadêmico de Eletrônica (DAELN) outorgado pela Universidade Tecnológica Federal do Paraná (UTFPR). Os alunos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Curitiba, 27 de fevereiro de 2019.

Prof. Dr. Robinson Vida Noronha
Coordenador de Curso
Engenharia Eletrônica

Prof^a. Dr^a. Carmen Caroline Rasera
Responsável pelos Trabalhos de Conclusão de Curso
de Engenharia Eletrônica do DAELN

BANCA EXAMINADORA

Prof. Dr. Paulo Roberto Brero de Campos.
Universidade Tecnológica Federal do Paraná
Orientador

Prof.^a. Dr^a. Carmen Caroline Rasera.
Universidade Tecnológica Federal do Paraná

Prof. M.e. Luiz Fernando Copetti.
Universidade Tecnológica Federal do Paraná

DEDICATÓRIA

*Aos nossos pais, amigos, companheiros,
familiares e a todos aqueles que estiveram nos
apoiando nesta jornada.*

AGRADECIMENTOS

A equipe gostaria de agradecer a todos que colaboraram direta e indiretamente para a realização do presente projeto de conclusão de curso:

- Agradecemos ao professor Dr. Paulo Roberto Brero de Campos pelo apoio, a disponibilização de todos os materiais e equipamentos necessários para a realização do projeto. Assim como as orientações gerais.
- Agradecemos aos familiares e amigos pelo apoio constante em nossas atividades acadêmicas e profissionais, além de sempre acreditarem em nossos potenciais. Sem a expertise, principalmente dos senhores: Jeferson Wilson Mendes, Anderson Mendes Ramos, Fernando Mendes Gonçalves, em fabricação de cerveja artesanal, seria improvável a concepção deste projeto.
- Agradecemos aos servidores que trabalham no almoxarifado, sem a boa vontade deles em nos cederem os equipamentos, seria impossível a realização deste projeto.
- Agradecemos ao metalúrgico Alexandre Moczinski pela sua maestria, expertise, experiência em confeccionar o protótipo — Panela, base, mangueira, sparge arm —, sem isso seria impossível à concepção deste trabalho de conclusão de curso.

RESUMO

GONÇALVES, Gustavo M.; FRANCISCO, Jorge L. **Automação de Processo de Brassagem para a Fabricação de Cerveja Artesanal**. Trabalho de Conclusão de Curso, Graduação em Engenharia Industrial Elétrica – ênfase Eletrônica/Telecomunicações – DAELN – Departamento Acadêmico de Eletrônica – UTFPR – Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

O consumo de cerveja artesanal está ficando cada vez mais comum e popular, bem como seu processo de fabricação. Pensando nisso, este projeto foi concebido e idealizado a fim de criar um produto para facilitar o processo de brassagem — Procedimento chave para efetuação da cerveja artesanal —, permitindo assim, maior rapidez, limpeza e efetividade. A brassagem consiste em três subprocessos: Mosturação, Lavagem e fervura. Sendo assim, este trabalho de conclusão de curso tem o objetivo de conceber um protótipo — Panela, base, mangueiras, *sparge arm*, bomba d’água — com um *hardware* e *firmware* para realização desse produto.

O *hardware* terá uma interface amigável com usuário, permitindo que este possa executar o processo (mosturação, lavagem, fervura) de forma intuitiva, com uma curva de aprendizagem rápida. O *Firmware multi-thread* executa cada parte do processo de forma conjunta, homogênea e personalizável pelo usuário. Também realizará controle, em malha fechada, da temperatura do sistema, pois, na etapa de mosturação deverão ser respeitadas determinadas rampas de temperatura configuradas pelo cliente por meio da interface. O *firmware* também efetuará um controle de fluxo, em malha aberta, da bomba d’água que fará a circulação de temperatura do mosto — mistura açucarada destinada à fermentação —. O *firmware* que realiza a automação do sistema, aquisição e controle de temperatura do mosto foi desenvolvido em *linguagem em C* adaptado para Arduino.

Palavras-chave: Cerveja artesanal; Automação; Controle térmico.

ABSTRACT

GONÇALVES, Gustavo M.; FRANCISCO, Jorge L. **Automação de Processo de Brassagem para a Fabricação de Cerveja Artesanal**. Trabalho de Conclusão de Curso, Graduação em Engenharia Industrial Elétrica – ênfase Eletrônica/Telecomunicações – DAELN – Departamento Acadêmico de Eletrônica – UTFPR – Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

The consumption of craft beer is getting more and more common and popular as well as its manufacturing process. With this in mind, this project was conceived and idealized in order to create a product to facilitate the process of mashing - a key procedure for brewing artisanal beer -, thus allowing greater speed, cleanliness and effectiveness. The mashing consists of three subprocesses: Mosturation, Washing and boiling. In this way, this final paper has the objective of designing a prototype - Pan, base, hoses, sparge arm, water pump - with a hardware and firmware to achieve this product.

The hardware will have a user-friendly interface allowing it to execute the process (mosturation, washing, boiling) intuitively with a quick learning curve. The multi-threaded Firmware performs each part of the process together, homogeneously and customizable by the user. It will also perform a closed-loop control of the system temperature, since certain temperature ramps configured by the customer through the interface must be respected at the sampling stage. The firmware will also perform an open loop flow control of the water pump that will circulate the temperature of the wort - a sugary mixture intended for fermentation. The firmware that performs the system automation, acquisition and temperature control of the wort was developed in C language adapted for Arduino

Keywords: Craft Beer; Automation; Thermic Control.

LISTA DE FIGURAS

Figura 1- Esquemático com equipamentos para fabricação de cerveja artesanal.....	15
Figura 2 - Curva clássica de temperatura de uma mosturação	16
Figura 3 - Modelagem térmica do protótipo.....	17
Figura 4 - Fluxo de calor	18
Figura 5 - Diagrama de blocos	20
Figura 6 - Diagrama do Protótipo.....	21
Figura 7 - Protótipo construído para projeto	22
Figura 8 - Resistência de inox baixa densidade.....	23
Figura 9 - Bomba d'água, 12v 800l/h.....	23
Figura 10 - Solenoide ½ polegada, 12V	24
Figura 11 - Sparge arm	24
Figura 12 - Circuito Regulador de potência	25
Figura 13 - Curvas de sincronizações e gatilho	26
Figura 14 - Resistencia térmica, analogia com lei de Ohm	27
Figura 15 - Perdas em condução de um semiconductor	29
Figura 16 - Corrente máxima na carga	29
Figura 17 - Arduino Due	30
Figura 18 - Display LCD 20x4.....	32
Figura 19 - Esquemático de ligação do alto-falante	32
Figura 20 - Esquema de botões do teclado	33
Figura 21 - Fluxograma do laço principal do firmware.....	35
Figura 22 - Sinal para o buzzer.....	38
Figura 23 - Montagem do sistema	39

Figura 24 – Diagrama de blocos de um sistema em malha fechada PID	40
Figura 25 - Tensão aplicada à carga pelo controlador PID	42
Figura 26 - Tensão aplicada à carga pelo controlador PID	42
Figura 27 – Tensão aplicada à carga pelo controlador PID.....	43
Figura 28 – Tensão aplicada à carga pelo controlador PID.....	43
Figura 29 – Tensão aplicada à carga pelo controlador PID.....	44

LISTA DE TABELAS

Tabela 1 - Características principais do triac Bta41	27
Tabela 2- Características térmicas do triac Bta41	28
Tabela 3 - Características dinâmicas do triac Bta41	28
Tabela 4 - Tempo de conversão do termômetro em relação à resolução utilizada	31

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

DB – Data Bus.

ESC – Escape.

GND – Ground.

I/O – Input/Output.

LCD – Liquid Crystal Display.

PCI – Placa de Circuito Impresso.

PD – Proporcional Derivativo.

PID – Proporcional Integral Derivativo.

PWM – Pulse Width Modulation.

R/W – Read/Write.

SRAM – Static Random Access Memory.

VDD – Voltage Drain (Ground).

VSS – Voltage Source (Ground).

SUMÁRIO

1 INTRODUÇÃO.....	13
1.1 MOTIVAÇÃO.....	13
1.2 OBJETIVOS.....	14
1.2.1 Objetivo Geral	14
1.2.2 Objetivos Específicos	14
1.3 ORGANIZAÇÃO DO TRABALHO.....	14
2 REVISÃO BIBLIOGRÁFICA.....	15
2.1 PROCESSO DE FABRICAÇÃO DE CERVEJA ARTESANAL	15
2.1.1 Visão Geral.....	15
2.1.2 Método Single Vessel.....	16
2.1.3 Aquecimento do mosto.....	16
2.2 MODELAGEM TERMICA DO SISTEMA	17
2.2.1 Definição de Resistência térmica	17
2.2.2 Definição de Capacitância térmica	18
2.2.3 Função de Transferência em Malha aberta do Sistema Térmico.....	18
3. MATERIAIS E DESENVOLVIMENTO	21
3.1 Diagrama	21
3.2 ESTRUTURA.....	22
3.2.1 Panela	22
3.2.2 Resistência.....	23
3.2.3 Bomba d'água.....	23
3.2.4 Solenoides.....	24
3.2.5 Sparge arm.....	24
3.3 HARDWARE	25
3.3.1 Circuito Regulador de Potência.....	25

3.3.1.1 Esquemático	25
3.3.1.2 TCA785	25
3.3.1.3 TL071	26
3.3.1.4 BTA41	26
3.3.1.5 4N25	27
3.3.2 Dissipador de Calor e Cálculo da Resistência Térmica.....	27
3.3.3 Arduino.....	30
3.3.4 Sensor de Temperatura	30
3.3.5 LCD	31
3.3.6 Buzzer.....	32
3.2.7 Teclado	32
3.4 FIRMWARE.....	33
3.4.1 Loop principal	33
3.4.2 Display de LCD	36
3.4.3 Teclado	36
3.4.4 Timer	36
3.4.5 Sensor de temperatura	37
3.4.6 Buzzer.....	37
3.5 MONTAGEM.....	38
3.7 CONTROLE.....	39
3.7.1 PWM.....	39
3.7.2 PID.....	39
3.8 INTERFACE COM O USUÁRIO	40
4. RESULTADOS	41
5. CONCLUSÃO.....	45
REFERÊNCIAS	47

1 INTRODUÇÃO

Um dos fatores de maior importância no processo de fabricação de cerveja artesanal é o processo de brassagem, subdividido em três subprocessos: Mosturação, lavagem, fervura. Esse processo deve ser feito de forma minuciosa para garantir que cada etapa seja feita de forma rápida, limpa e homogênea. Portanto, determinadas variações de temperaturas (sinais do tipo rampas) deverão ser respeitadas, etapas deverão ser seguidas e o fluxo do mosto deverá ser ajustado.

Diante disso, para aumentar a confiabilidade e a agilidade das rampas de temperaturas, será utilizado um controle em malha fechada utilizando, para tal, o controlador PID. Os compensadores PID são utilizados no controle em malha fechada, onde, geralmente, não se conhecem os parâmetros do sistema em questão.

O presente trabalho apresenta o desenvolvimento de um produto que realiza a automação do processo de brassagem. O protótipo realiza, passo a passo, a execução dos subprocessos: mosturação, lavagem e fervura a fim de conceber ao usuário uma experiência satisfatória ao fabricar sua própria cerveja artesanal.

1.1 MOTIVAÇÃO

Devido ao grande crescimento tanto do consumo quanto da fabricação de cervejas artesanais, houve um aumento significativo no interesse das pessoas em como produzir tal bebida, o que acabou tornando essa prática um lazer, um *hobby* e até um investimento. Para que isso continue evoluindo e o *hobby* permaneça prazeroso, começa a existir uma necessidade de tornar todos esses processos mais práticos, intuitivos e acessíveis à população.

Honrando o papel da engenharia de tornar a vida, os procedimentos e o trabalho das pessoas mais fáceis, práticos e otimizados, decidiu-se criar um sistema capaz de proporcionar isso às pessoas.

O processo de brassagem, justamente, é o preferido por grande parte dos aficionados em fabricação de cerveja artesanal, por isso foi escolhido para ser o alvo deste projeto. A motivação é a de avançar um nível na qualidade dessa etapa, proporcionar uma personalização e uma praticidade que não se obteriam em um procedimento tradicional, para que cada artesão possa colocar sua marca e fazer seu produto do seu jeito de uma maneira fácil.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo deste trabalho é desenvolver um sistema de automação para facilitar e tornar mais agradável o processo de fabricação de cerveja artesanal. O sistema deve ter a capacidade de regular a temperatura de aproximadamente 50 litros de mosto de acordo com o que o cliente desejar e deverá ter um design de interação amigável para que seja facilmente programado e utilizado por qualquer tipo de usuário.

1.2.2 Objetivos Específicos

- Desenvolver e implementar o firmware do sistema. O projeto deve possibilitar o controle da temperatura do mosto com precisão de 0.5 graus centígrados, bem como controlar o fluxo, o tempo que a mistura permanece em cada temperatura e a potência a ser entregue na resistência de aquecimento.
- Desenvolver e montar o hardware do sistema. O sistema deverá trabalhar com tensões de 220V para a resistência de aquecimento; 12V para a bomba d'água; 5V para os periféricos e 3,3V para o microcontrolador. Deverá ser capaz de conter e ferver até 50 litros de mistura.
- Implementar um sistema de controle de temperatura que permita otimizar o tempo e realizar as rampas necessárias ao processo de brassagem da cerveja artesanal sem nunca ultrapassar o valor de temperatura desejado pelo usuário, bem como mantê-la constante quando for necessário ou requerido.

1.3 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está dividido em cinco capítulos. No Capítulo 2, é feita uma revisão bibliográfica sobre os processos de fabricação das cervejas artesanais e modelagem térmica. No Capítulo 3, encontram-se descritos os materiais, equipamentos e métodos adotados para o desenvolvimento, implementação e execução do trabalho, toda a parte da estrutura física do produto, a parte de hardware utilizado e a implementação do firmware para administrar todo o sistema. O Capítulo 4 apresenta os resultados experimentais e o Capítulo 5 apresentam as conclusões do trabalho com algumas sugestões para trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta a fundamentação teórica dos conceitos básicos necessários para esse projeto. São abordados os temas: Processos de fabricação de cerveja artesanal, método single vessel, modelagem térmica do sistema.

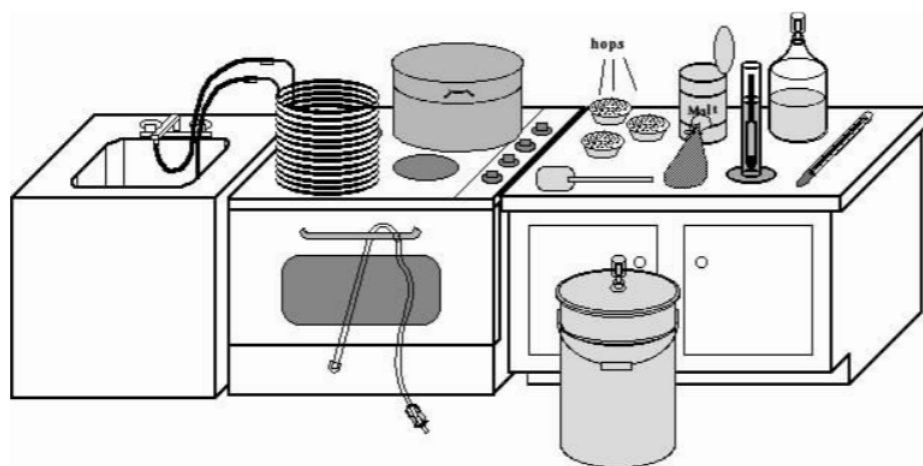
2.1 PROCESSO DE FABRICAÇÃO DE CERVEJA ARTESANAL

2.1.1 Visão Geral

Na figura 1 estão representados os espaços e utensílios comumente utilizados para o processo. A fabricação de cerveja artesanal resume-se em aproximadamente cinco partes:

- Malte parcialmente mergulhado em água quente, respeitando uma determinada curva de temperatura - Projeto de controle proposto neste trabalho.
- A Solução de açúcar maltado é fervida com lúpulo para tempera-la.
- A solução é resfriada e o fermento é adicionado, começando a fermentação.
- O fermento fermenta os açúcares, liberando, assim, dióxido de carbono e álcool etílico.
- Quando a fermentação principal é completada, a cerveja é engarrafada, uma quantidade pequena de açúcar maltado é adicionada a fim de aumentar a carbonatação.

Figura 1- Esquemático com equipamentos para fabricação de cerveja artesanal.



Fonte: How to brew, 2006.

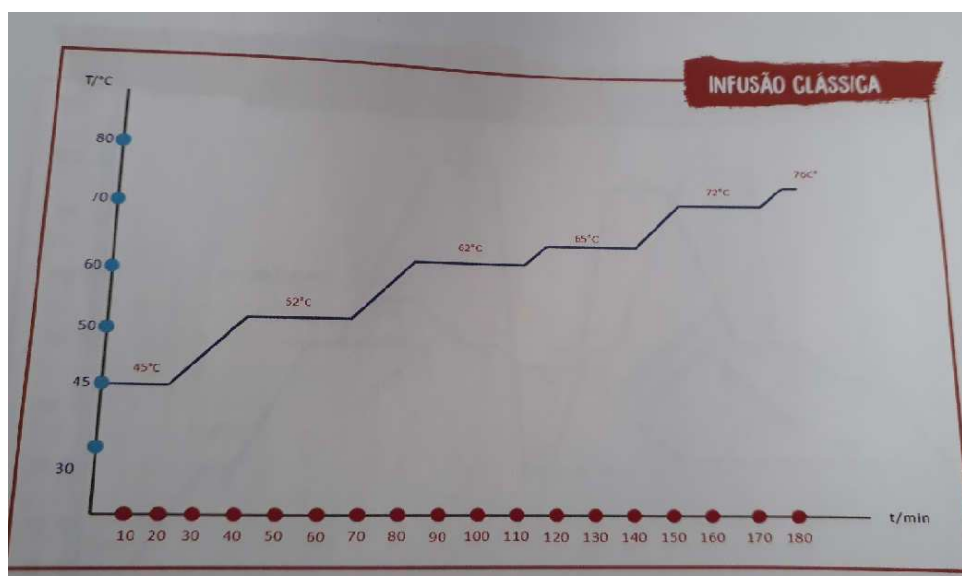
2.1.2 Método Single Vessel

Método utilizado para a automatização do processo de brassagem para fabricação de cerveja artesanal será o single vessel. Método que utiliza somente uma panela e fazendo a lavagem junto da mosturação.

2.1.3 Aquecimento do mosto

Na figura 2 é mostrada a curva clássica de temperatura de uma mosturação, com seus diversos patamares e o tempo necessário em cada um deles.

Figura 2 - Curva clássica de temperatura de uma mosturação.

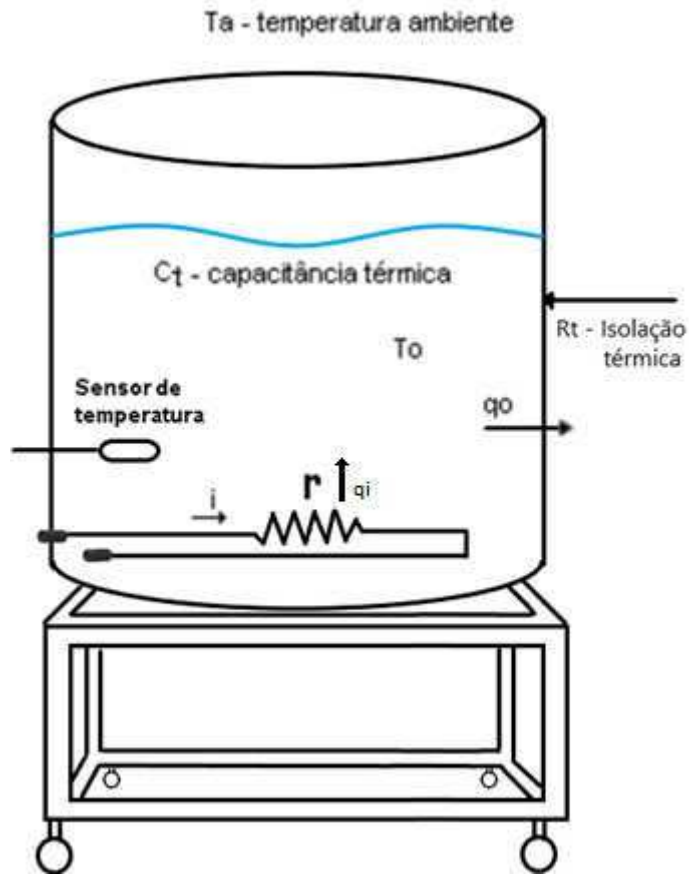


Fonte: MODENESI et al., 2012.

2.2 MODELAGEM TERMICA DO SISTEMA

Para a modelagem térmica do protótipo, usa-se a configuração mostrada na figura 3.

Figura 3 - Modelagem térmica do protótipo.



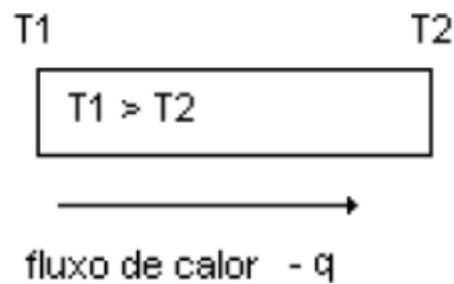
Fonte: How to brew, 2006.

Sendo que, r = resistência elétrica para aquecer a panela; a variável de entrada é o fluxo de calor (q_i) e a variável de saída é a temperatura do poço térmico T_o presente na panela.

2.2.1 Definição de Resistência térmica

Dado um material que seja condutor, como mostrado na figura 4, caso exista uma diferença de temperatura em dois pontos distintos, haverá um fluxo de calor indo da temperatura maior para a menor. A resistência a esse fluxo, recebe o nome de resistência térmica.

Figura 4 - Fluxo de calor.



Fonte: Notas de Aula, Prof. Paulo Roberto Brero.

A resistência térmica é definida, matematicamente, pela equação 1:

$$R_t = \frac{T_1 - T_2}{q} \quad (1)$$

No qual T = temperatura e q = fluxo de calor.

2.2.2 Definição de Capacitância térmica

A diferença entre o calor fornecido (q_i) e o calor perdido através das paredes (q_o) é igual ao calor dentro da panela.

O calor armazenado dentro da panela é proporcional à taxa de variação da temperatura na panela, em que a constante de proporcionalidade (C) é definida como a capacitância térmica do meio, como mostrado na equação 2.

$$q_i - q_o = C \frac{dT}{dt} \quad (2)$$

2.2.3 Função de Transferência em Malha aberta do Sistema Térmico

Reescrevendo a equação (1), tem-se a equação 3:

$$q_o = (T_o - T_a) / R_T \quad (3)$$

Substituindo (3) em (2):

$$q_i - (T_o - T_a) / R_T = C \frac{dT_o}{dt}$$

$$q_i - T_o / R_T + T_a / R_T = C \frac{dT_o}{dt}$$

$$R_t q_i - T_o + T_a = R_t C \frac{dT_o}{dt}$$

Dessa forma, a equação 4 fornece a relação dinâmica entre temperatura e fluxo de calor.

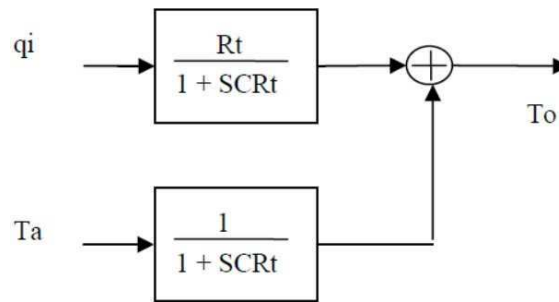
$$R_t q_i + T_a = R_t C \frac{dT_o}{dt} + T_o \quad (4)$$

Aplicando a transformada de Laplace na equação 4 obtém-se a equação 5.

$$R_t q_i + T_a = S R_t C T_o + T_o \quad (5)$$

Na equação (5), têm-se duas variáveis de entrada: a variável T_a e q_i e uma variável de saída, T_o . Que corresponde, respectivamente, à temperatura ambiente, potência dissipada no resistor e temperatura de saída do sensor de temperatura. Como é exemplificado na figura 5.

Figura 5 - Diagrama de blocos.



Fonte: Notas de Aula, Prof. Paulo Roberto Brero.

Nota-se que a temperatura ambiente, T_a , **tem o efeito de uma carga no sistema.**

Para este sistema, o ideal seria que houvesse apenas uma entrada, q_i , e uma saída, T_o . Da forma como mostrada na figura 4, não é possível escrever na forma de função de transferência devido ao termo T_a . Para resolver este impasse, a resistência térmica da panela (R_{te}) é definida, como mostrado pela equação 6:

$$R_{te} = \frac{(R_t q_i + T_a)}{q_i} \quad (6)$$

Substituindo (6) em (5):

$$R_{te} q_i = S R_t C T_o + T_o$$

Assim, é possível obter a função de transferência como mostrado na equação 7.

$$\frac{T_o}{q_i} = \frac{R_{te}}{1 + S C R_t} \quad (7)$$

Onde, $C R_t$ = constante de tempo da panela.

Nota-se que o sistema é de primeira ordem e lento, uma vez que a constante de tempo, $C R_t$, depende de várias grandezas físicas de respostas lentas, tais como: Capacitância térmica do ar, da água, calor específico da água. Fazendo, assim, com que o sistema fique lento.

3. MATERIAIS E DESENVOLVIMENTO

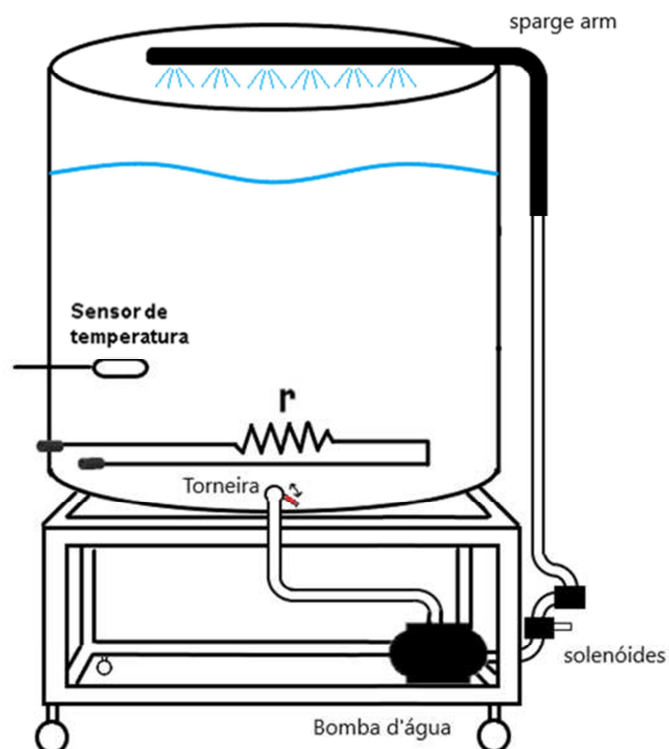
Como citado anteriormente, o presente trabalho é um produto que automatizará o processo de brassagem para fabricação de cerveja artesanal.

Nas próximas seções serão expostos detalhadamente os materiais e os procedimentos utilizados para a realização desse projeto.

3.1 Diagrama

Na figura 6 é mostrado o diagrama do protótipo do produto desenvolvido, que esquenta o mosto seguindo a rampa de temperatura pré-determinada. Contém dois solenóides que nunca estão acionados concomitantemente, e definem, dependendo de qual esteja acionado no momento, se o fluxo do mosto está em circulação interna, para uma variação homogênea de temperatura, ou se está no fim, exportando o mosto para o fermentador.

Figura 6 - Diagrama do Protótipo.



Fonte: Autoria Própria.

3.2 ESTRUTURA

3.2.1 Panela

Na figura 7 é mostrada a fotografia da panela de inox 304 com furos para torneira e para resistência.

Figura 7 - Protótipo construído para projeto.

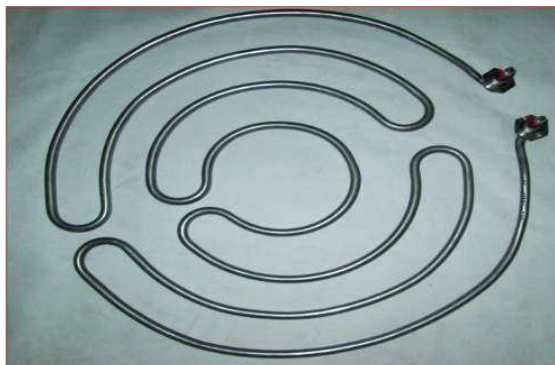


Fonte: Autoria Própria.

3.2.2 Resistência

Uma resistência de inox de, aproximadamente, 8,8 fi, que ligado ao 220V, produz uma potência dissipada, Q_i , de 5500W, como mostrado na figura 8.

Figura 8 - Resistência de inox baixa densidade



Fonte: Autoria Própria.

3.2.3 Bomba d'água

A bomba d'água (12V, 800l/h) para fazer a circulação do mosto e exportação, deste, para o fermentador, é mostrada na figura 9.

Figura 9 - Bomba d'água, 12v 800l/h.



Fonte: Internet, Mercado livre.

3.2.4 Solenoides

São utilizados dois solenoides (½ polegada, 12V) para controlar o fluxo do mosto, seja para circulação interna — mosturação, lavagem — seja na hora de exportar para o fermentador. Mostrado na figura 10.

Figura 10 - Solenoide ½ polegada, 12V.



Fonte: Internet, Mercado livre.

3.2.5 Sparge arm

O *sparge arm* é um braço de inox 304 que irriga o mosto, fazendo com que temperatura do mosto varie de forma homogênea, como mostrado na figura 11.

Figura 11 - Sparge arm.



Fonte: Autoria Própria.

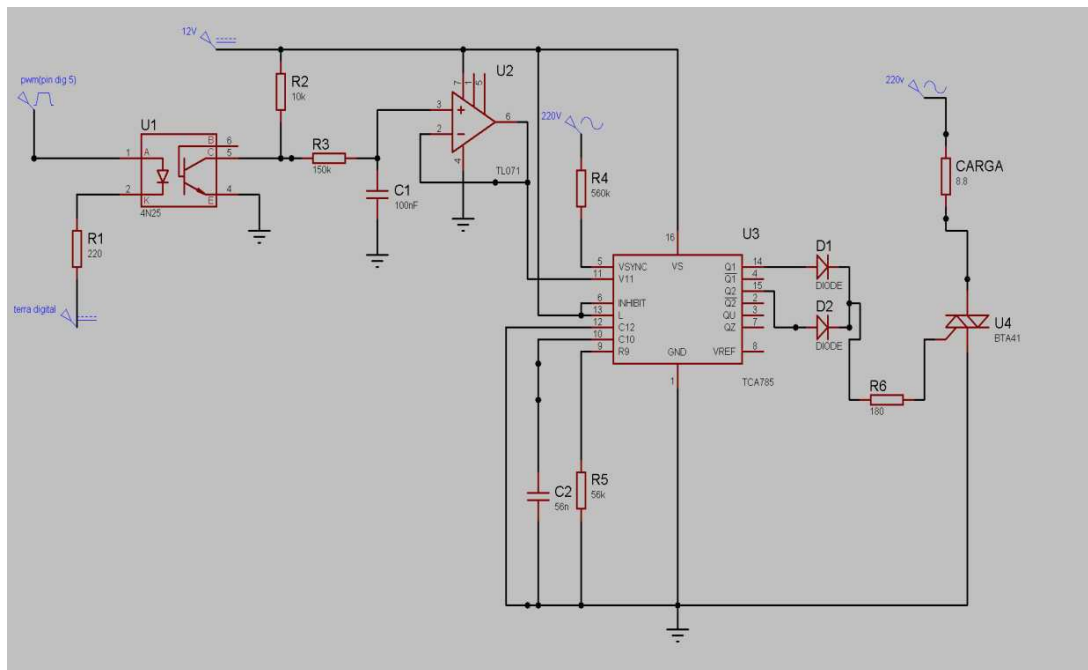
3.3 HARDWARE

3.3.1 Circuito Regulador de Potência.

3.3.1.1 Esquemático

O diagrama esquemático é mostrado na figura 12.

Figura 12 - Circuito Regulador de potência.



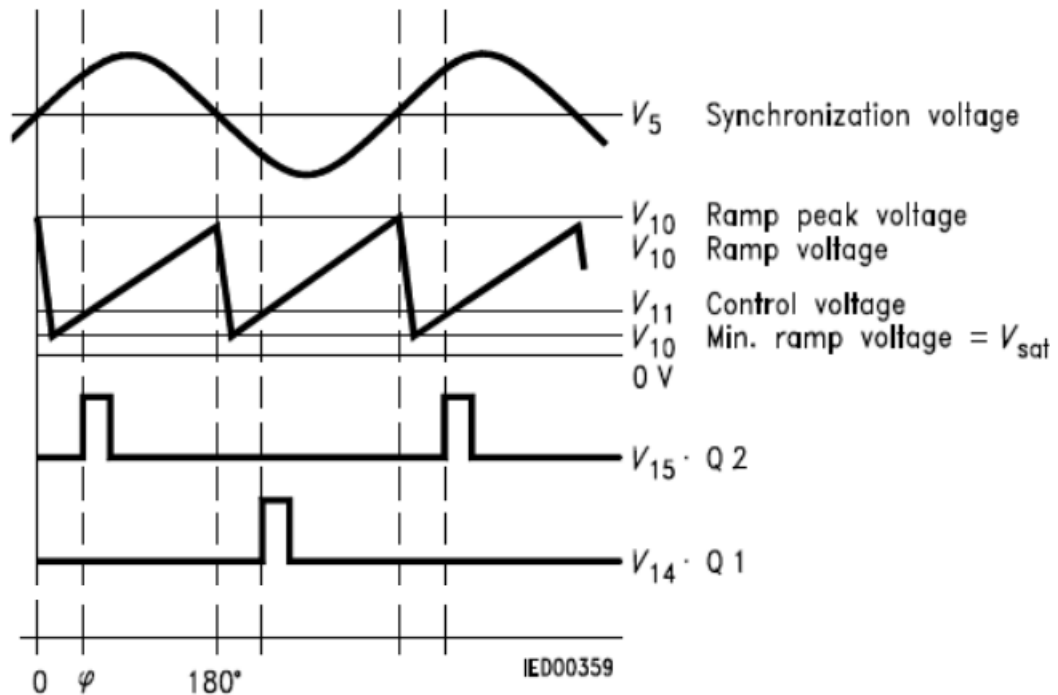
Fonte: Aatoria Própria.

É o circuito responsável por regular a potência na carga (resistência de inox mostrada anteriormente) e fazer a interface entre o microcontrolador e a carga, além de fazer a isolação galvânica do mesmo, ou seja, isolar o terra do microcontrolador da fase 220V da rede.

3.3.1.2 TCA785

TCA785 é um controlador de fase IC que se destina a controlar tiristores, triacs e transistores. Pode ser deslocado-tanto fase positiva quanto negativa— dentro de um ângulo de 0° a 180°.O CI funciona a partir da detecção da passagem por zero da senoide controlada, desta forma as curvas de sincronização são mostradas na figura 13:

Figura 13 - Curvas de sincronizações e gatilho.



Fonte: Datasheet TCA 785, SIEMENS.

O sinal de sincronização é obtido através de resistência de alta impedância (tensão v_5). Um detector de tensão zero avalia as passagens por zero do sinal, e as manda para um registrador de sincronização. Esse registrador controla uma rampa dente-de-serra (em V_{10}) sincronizada com a tensão da rede. Quando a tensão de controle (saída do Amplificador Operacional) excede a tensão da rampa V_{10} (ângulo de disparo), um pulso (*de 30ms de duração*) é acionado em V_{15} e V_{14} que dispara o triac tanto na fase positiva (V_{15}) quanto na fase negativa (V_{14}).

3.3.1.3 TL071

O TL071 é um Amplificador operacional na configuração de buffer para driver de corrente. Fornece corrente para o filtro RC que transforma em tensão média o PWM que sai do pino 5 do acoplador óptico (4n25).

3.3.1.4 BTA41

O BTA41 é um triac de alta potência indicado para uso geral de chaveamento AC. Foi escolhido, pois contempla as especificações do projeto de alta corrente (Triac que suporte 25A RMS), como mostra a tabela 1.

Tabela 1 - Características principais do triac Bta41

Symbol	Value	Unit
$I_{T(RMS)}$	40	A
V_{DRM}/V_{RRM}	600 and 800	V
$I_{GT} (Q_1)$	50	mA

Fonte: Datasheet BTA40, BTA41 and BTB41 Series, STANDARD.

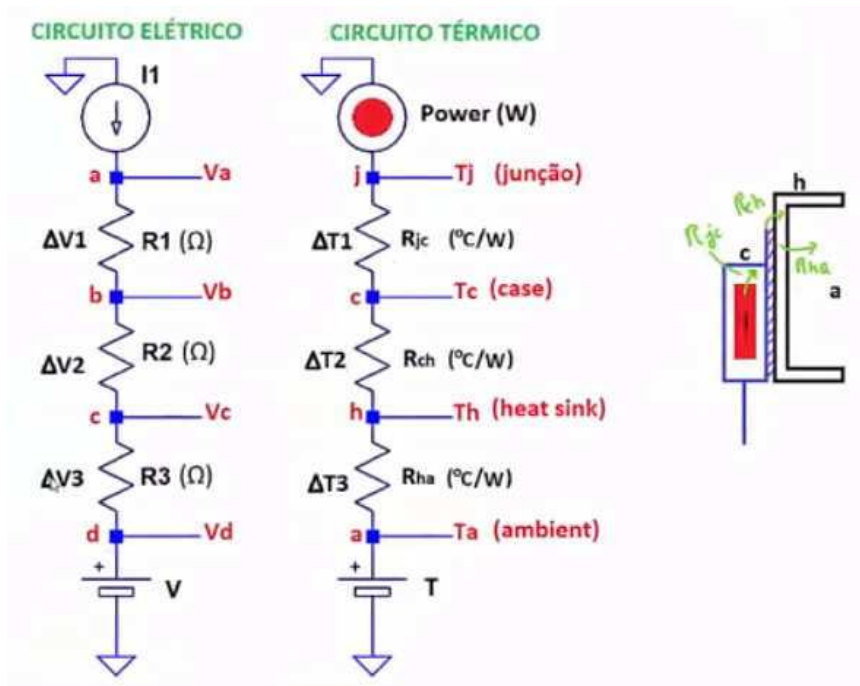
3.3.1.5 4N25

O 4N25 é um acoplador óptico que faz o isolamento galvânico entre o terra digital (Microcontrolador) e a fase 220V da rede.

3.3.2 Dissipador de Calor e Cálculo da Resistência Térmica

Para um projeto de alta potência como este. Projetar um dissipador de calor para o triac é essencial. Para tal, precisamos rever alguns conceitos, como: Resistência térmica e sua analogia com a lei de Ohm, como mostrado na figura 14.

Figura 14 - Resistência térmica, analogia com lei de Ohm.



Fonte: Notas de aula, Profa Fabiana Pottker.

$$V_a - V_d = (R_1 + R_2 + R_3) I \quad (8)$$

$$T_j - T_a = (r_{jc} + r_{ch} + r_{ha}) P \quad (9)$$

Ou seja, é possível calcular a resistência térmica do dissipador para o triac fazendo um cálculo análogo à lei de Ohm. Para isso, é importante determinar: Resistência dinâmica do triac, resistência térmica capsula-junção do triac, resistência térmica da mica e potência dissipada no triac.

Tanto a resistência em condução quanto a resistência junção-capsula são obtidos através da do datasheet do Bta41, como visto nas tabelas 2 e 3 abaixo.

Tabela 2- Características térmicas do triac Bta41

Symbol	Parameter		Value	Unit
$R_{th(j-c)}$	Junction to case (AC)	RD91 (Insulated) / TOP3	0.9	°C/W
		TOP3 Insulated	0.6	
$R_{th(j-a)}$	Junction to ambient	TOP3 / TOP3 Insulated	50	°C/W

Fonte: Datasheet BTA40, BTA41 and BTB41 Series, STANDARD.

Tabela 3 - Características dinâmicas do triac Bta41

Symbol	Test Conditions			Value	Unit	
V_T (2)	$I_{TM} = 60 \text{ A}$	$t_p = 380 \mu\text{s}$	$T_j = 25^\circ\text{C}$	MAX.	1.55	V
V_{T0} (2)	Threshold voltage		$T_j = 125^\circ\text{C}$	MAX.	0.85	V
R_d (2)	Dynamic resistance		$T_j = 125^\circ\text{C}$	MAX.	10	mΩ
I_{DRM} I_{RRM}	$V_{DRM} = V_{RRM}$		$T_j = 25^\circ\text{C}$	MAX.	5	μA
			$T_j = 125^\circ\text{C}$		5	mA

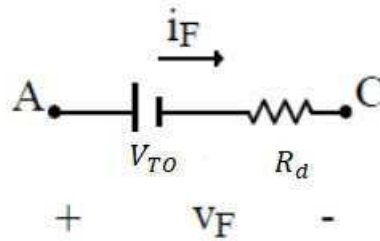
Fonte: Datasheet BTA40, BTA41 and BTB41 Series, STANDARD.

Ao pesquisar as tabelas 2 e 3, encontrou-se uma resistência dinâmica, R_d , de 10mfi e uma resistência junção-capsula, R_{jc} , de 0,6 °C/W.

A resistência térmica da folha de mica, R_{ch} , é dada pelo fabricante, como sendo de 0,4 °C/W.

Para o cálculo da potência dissipada no triac (BTA41), temos que recorrer ao circuito elétrico equivalente a um triac em condução. Como mostra a figura 15.

Figura 15 - Perdas em condução de um semiconductor.

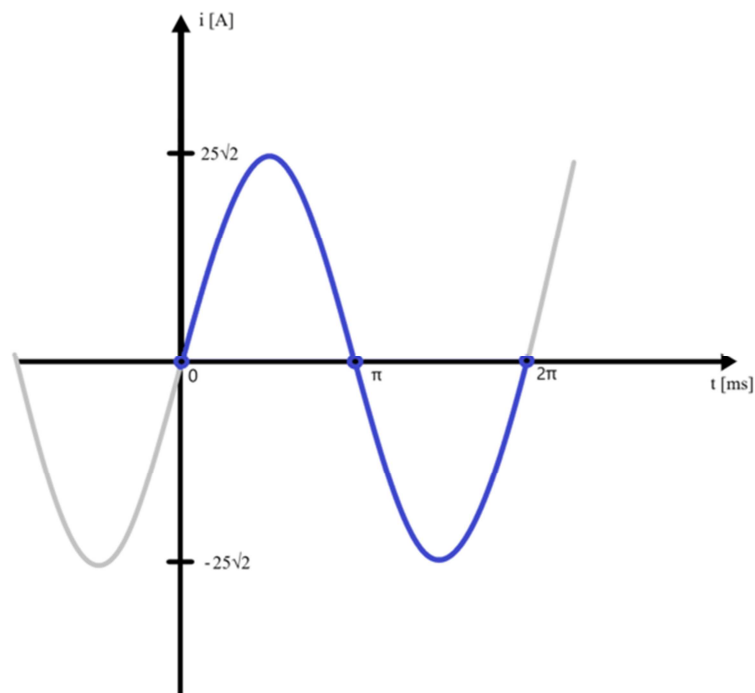


Fonte: Notas de aula, Profa Fabiana Pottker.

Em que V_{TO} (tensão de polarização do triac) para o BTA41 é 1,55V (Tabela 2). E a resistência dinâmica, R_d , de 10mfi.

A corrente I_f é a situação de maior potência na carga possível, como indica a figura 16

Figura 16 - Corrente máxima na carga.



Fonte: autoria própria.

A potência dissipada no triac é dada pela equação a seguir:

$$P_{\text{Triac}} = V_{TO} I_{T\text{Ned}} + R_d I_{Tef}^2 \quad (10)$$

Em que, $I_{T\text{Ned}}$ = corrente média no triac; I_{Tef} = corrente eficaz no triac.

Assim, tem-se:

$$P_{\text{Triac}} = 1,55 \times 0 + 10\text{mfi} \times 25^2$$

$$P_{\text{Triac}} = 6,28\text{W}$$

Agora tem-se todos os dados necessários, precisa-se definir uma temperatura ambiente, T_a , e uma temperatura de junção, T_j , desejadas. Lembrando que esta temperatura deve ser menor que 125°C (temperatura máxima de junção do Bta41).

Portanto foi escolhido: $T_a = 25^{\circ}\text{C}$ e $T_j = 80^{\circ}$

Com auxílio da equação (9), calcula-se r_{ha} (resistência térmica do dissipador).

$$80 - 25 = (0,6 + 0,4 + r_{ha}) \cdot 6,28$$

$$r_{ha} = 7,76^{\circ}\text{C/W}$$

Porem foi utilizado o dissipador comercial, HS 8620, que possui uma resistência térmica de $5,34^{\circ}\text{C/W}$, fazendo com que a temperatura de junção seja ainda mais baixa.

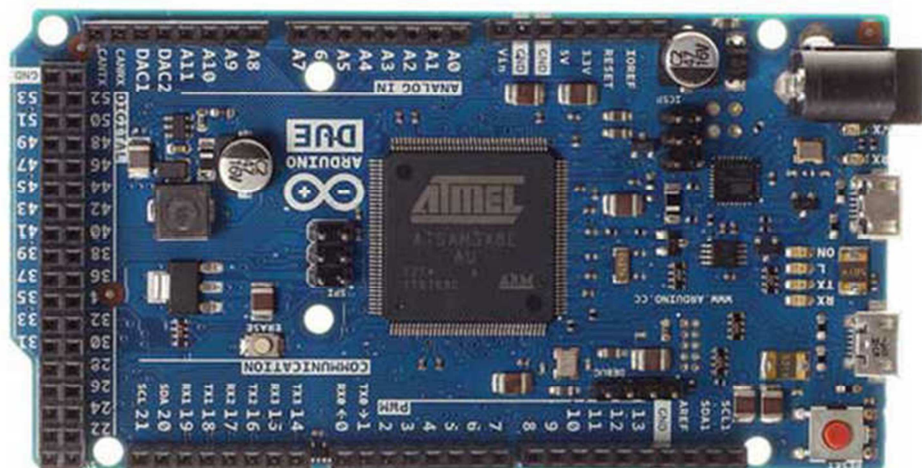
$$T_j - 25 = (0,6 + 0,4 + 5,34) \cdot 6,28$$

$$T_j = 64,82^{\circ}$$

3.3.3 Arduino

O Aduino Due, mostrado na figura 17 é uma placa baseada no microprocessador Atmel SAM3X8E ARM Cortex-M3 de 32bits. Opera com uma tensão de 3,3V, possui 54 pinos de I/O digital (12 dos quais suporta saída PWM), 12 pinos de entrada analógica, 2 pinos de conversor Digital-Analógico, 512kB de memória flash, 96kB de SRAM e frequência de clock de 84MHz.

Figura 17 - Arduino Due.



Fonte: arduino.cc.

3.3.4 Sensor de Temperatura

O sensor de temperatura utilizado é o DS18B20 da Dallas Semiconductor, ele é um termômetro de um fio, a prova d'água, com resolução de até 0.0625 graus, porém no

projeto está sendo utilizado com apenas 9 bits, o que comporta uma resolução de 0.5 graus centígrados.

As diferentes resoluções são de 0.5, 0.25, 0.125 e 0.0625 utilizando 9, 10, 11 e 12 bits respectivamente. O que varia de forma muito expressiva é o tempo de conversão, como indicado na tabela 4.

Tabela 4 - Tempo de conversão do termômetro em relação à resolução utilizada.

R1	R0	Thermometer Resolution	Max Conversion Time
0	0	9 bit	93.75 ms $(t_{conv}/8)$
0	1	10 bit	187.5 ms $(t_{conv}/4)$
1	0	11 bit	375 ms $(t_{conv}/2)$
1	1	12 bit	750 ms (t_{conv})

Fonte: Datasheet DS18B20 Dallas Semiconductor.

Foi escolhida a resolução de 9bits para a realização deste projeto, pois impactaria menos na temporização do firmware. Isto é, tem um menor tempo de conversão.

3.3.5 LCD

Como interface gráfica foi escolhido o módulo display LCD 2004A, mostrado na figura 18, que pode representar até 20 caracteres em cada uma de suas quatro linhas. É alimentado com 5V e se conecta com o microcontrolador por meio de 16 pinos (visíveis na figura 17), dos quais: VSS, VDD e V0 são para a alimentação (aterramento, alimentação 5V, e tensão de acionamento respectivamente); RS, R/W e Enable para o controle das instruções; DB0-DB7 para os dados; e A e K para a alimentação da luz de fundo (*backlight*) do display (alimentação e terra, respectivamente).

O display necessita uma alimentação de 5V para seu funcionamento, entretanto os sinais do microcontrolador possuem apenas 3,3V. Para resolver esse desencontro foi necessária a utilização de conversores que permitiram elevar para 5V o sinal proveniente do Arduino.

Figura 18 - Display LCD 20x4

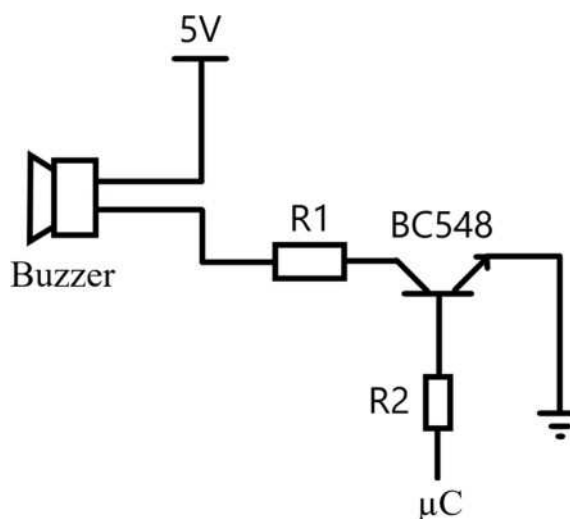


Fonte: Autoria própria.

3.3.6 Buzzer

Para este projeto foi utilizado um alto-falante conectado como indica a figura 19. O resistor R1 é o que regula a amplitude (volume) do som tocado pelo alto-falante.

Figura 19 - Esquemático de ligação do alto-falante.



Fonte: Autoria própria.

3.2.7 Teclado

Para compor a interface de usuário do produto foi utilizado um teclado 4x4, ilustrado na figura 20, para que seja possível navegar nos menus e configurar o sistema com o que for requerido. O teclado possui oito pinos (um para cada linha e um para cada coluna) que foram ligados diretamente no microcontrolador.

Figura 20 - Esquema de botões do teclado.

1	2	3	ESC
4	5	6	▲
7	8	9	▼
◀	0	▶	OK

Fonte: Autoria própria.

3.4 FIRMWARE

3.4.1 Loop principal

O laço principal do firmware consiste em uma grande máquina de estados que gerencia todas as funcionalidades do projeto. O programa inicia no primeiro menu, nele é possível: ajustar as configurações do procedimento de brassagem a ser realizado, temperatura, tempos de espera e número de rampas desejadas; fazer pequenos ajustes de som, como silenciar bipes e alarmes; e, para uma opção mais rápida e controlada, pode-se iniciar o processo de modo manual, ajustando cada temperatura, rampa e espera em tempo de execução. A figura 20 apresenta um fluxograma geral do funcionamento do laço principal.

O padrão de utilização do produto será em modo automático, onde o usuário poderá configurar o número de patamares, a temperatura na qual deseja que a mistura permaneça e o tempo no qual deseja que a mistura permaneça naquela temperatura. Logo após ter configurado todas essas opções é possível iniciar o processo automático.

Inicialmente será entregue potência à resistência de aquecimento e será acionada a bomba d'água para homogeneizar a temperatura. Por meio do sensor térmico o programa é capaz de medir a temperatura e controlar de forma eficaz o aquecimento. O quesito mais importante desta etapa é reservar o cuidado de nunca ultrapassar a temperatura desejada, pois além de poder comprometer a mistura não será possível reverter a temperatura rapidamente, pois não há como retirar o calor do sistema de forma eficaz.

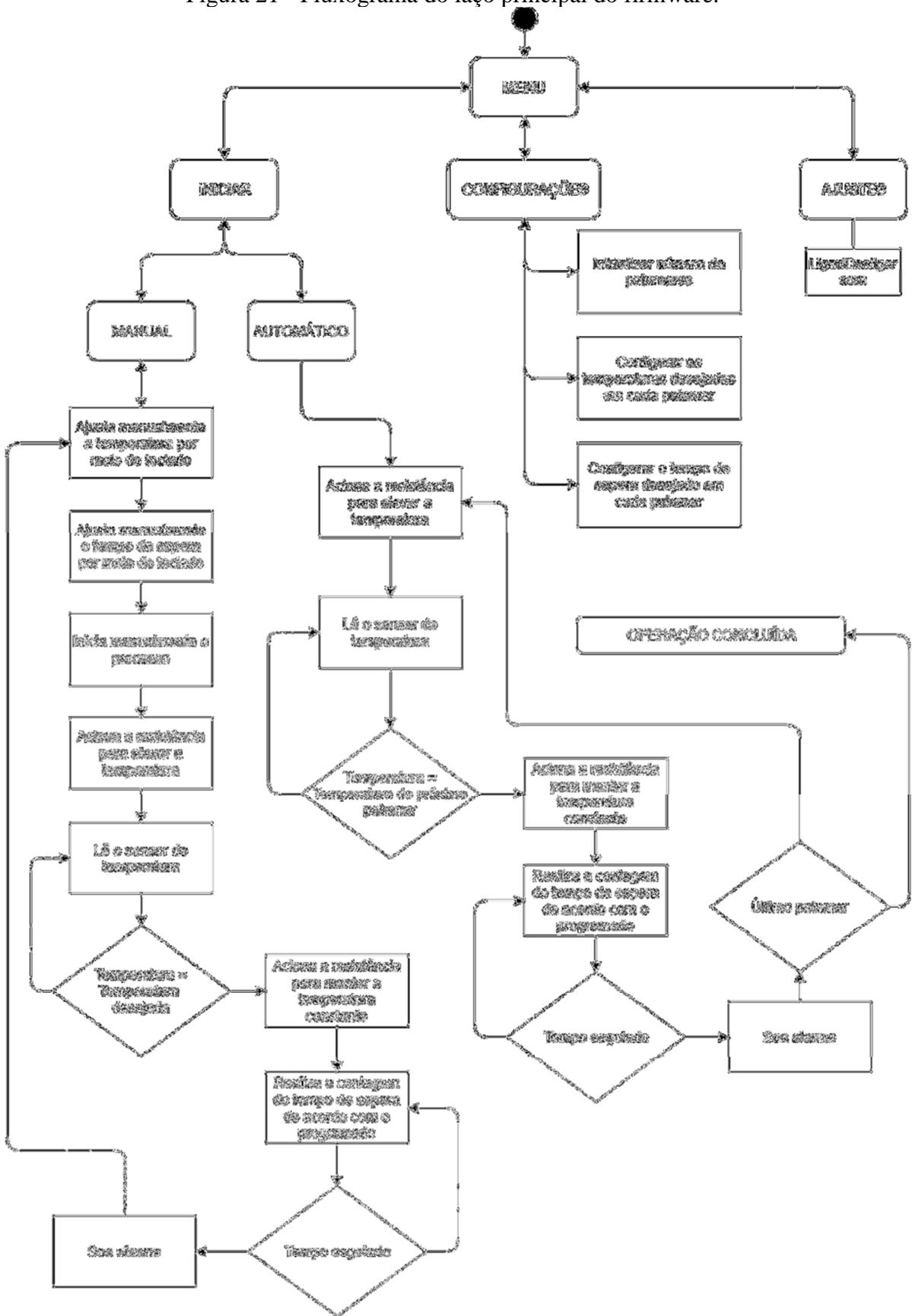
Quando a temperatura desejada for atingida o programa entra em estado de espera, um contador de tempo é mostrado no display, bem como a temperatura atual e o tempo que será esperado, para controle do usuário. Nesse momento a temperatura é mantida constante.

Assim que o tempo de espera desejado acabar, o sistema iniciará novamente o aquecimento da mistura na rampa seguinte até atingir o próximo patamar, repetindo assim o processo. O procedimento acaba quando terminar a espera do último patamar configurado, ou quando o usuário desejar interrompê-lo.

A qualquer momento nas etapas de aquecimento ou espera o usuário poderá pressionar a tecla ESC para “pausar” o programa. Feito isso, o sistema manterá a temperatura constante do momento em que foi pausado e qualquer cronômetro que estiver em execução será congelado. Ao apertar a tecla BACK “<” o usuário poderá cancelar todo o processo e volta ao menu principal, ao pressionar ESC novamente o programa retoma a execução de onde parou. Isso pode ser muito útil para acrescentar “subpatamares” de improviso no procedimento.

Na figura 21, é mostrado o fluxograma do laço principal do firmware.

Figura 21 - Fluxograma do laço principal do firmware.



Fonte: Autoria própria.

3.4.2 Display de LCD

Para a utilização deste display LCD em alto nível foi criada uma biblioteca com as funções necessárias. As principais e mais utilizadas funções criadas são: LCD_GOTOXY() e LCD_COUT(). A primeira é capaz de posicionar o cursor em qualquer uma das 80 posições do LCD, indicando qual linha e qual coluna se deseja escrever. A segunda é capaz de escrever uma cadeia de caracteres no display.

Com a junção dessas duas funções é possível escrever qualquer coisa em qualquer lugar da tela do dispositivo. Adicionalmente também foi criada uma biblioteca para gerenciar todas as telas do firmware, assim foi possível fazer os menus, a interface, os avisos e todas as etapas visuais do produto de maneira mais prática, modular e rápida.

3.4.3 Teclado

Para a utilização do teclado também foi criada uma biblioteca específica para tratar esse componente.

Para realizar a leitura do teclado e identificar corretamente qual tecla o usuário está pressionando é feita uma varredura arbitrando as colunas como entradas e as linhas como saídas. Por definição os pinos de I/O inicializam em nível lógico alto, portanto para a varredura ser bem-sucedida atribui-se valor lógico baixo a uma das linhas e logo se verifica o estado de cada coluna. Caso o valor lido de alguma coluna for um nível lógico baixo, significa que o usuário está pressionando a tecla posicionada exatamente naquela linha e coluna lidas.

Para evitar o efeito de *bouncing*¹ o firmware espera o usuário soltar a tecla para considerar que ela foi pressionada garantindo assim que a tecla foi de fato apertada e não houve enganos. Além disso, o firmware realiza uma espera de 100ms entre cada varredura para impedir os múltiplos pressionamentos gerados pelo *bouncing*.

3.4.4 Timer

Para a utilização dos timers do Arduino Due foi utilizada a biblioteca “DueTimer.h” criada por Ivan Seidel.

O Projeto utiliza três timers, todos eles disparam uma vez a cada segundo. O primeiro timer é o que gerencia toda a temporização do programa, a cada segundo ele verifica se há algum tempo sendo contado e realiza o incremento. Utilizado principalmente para contar o

¹ O efeito de *bouncing* consiste na tendência de dois contatos metálicos gerarem múltiplos sinais assim que forem conectados ou desconectados. Em teclados esse efeito é muito comum e costuma gerar vários pressionamentos de tecla, mesmo que o usuário as haja pressionado apenas uma vez.

tempo decorrido durante a espera em cada patamar de temperatura. O segundo timer utilizado é o do sensor de temperatura. A cada segundo ele realiza uma requisição ao sensor da temperatura que o mesmo está medindo e atualiza a variável de “temperatura atual”. O terceiro timer é utilizado para entregar a potência na resistência de aquecimento. A cada segundo ele verifica o valor da temperatura atual (por meio do sensor de temperatura) e realiza a equação a diferenças característica do controlador PD para calcular quanta potência deve deixar na carga. Esse sistema de potência também trabalha com uma máquina de estados, porém muito mais simples que a do laço principal. O estado da potência pode estar em “Idle”, “Esquentando” ou “Pause”. Em “Idle” o sistema entrega potência zero na carga, para os momentos em que não há nada inicializado e as configurações ainda estão sendo feitas. Em “Esquentando” ela realiza a equação a diferenças visando a temperatura desejada e entregando a potência adequada na carga. Em “Pause” também realiza a equação a diferenças, porém com o *setpoint* de temperatura (objetivo) sendo a mesma temperatura que estava no momento em que o sistema foi parado, mantendo assim a temperatura constante.

3.4.5 Sensor de temperatura

Para o controle do sensor de temperatura foram utilizadas as bibliotecas “DallasTemperature.h” juntamente com a “OneWire.h”.

Um timer que dispara a cada 1000ms (um segundo) realiza o procedimento de leitura do sensor. Assim que o firmware receber o valor lido, atualiza a temperatura atual do sistema.

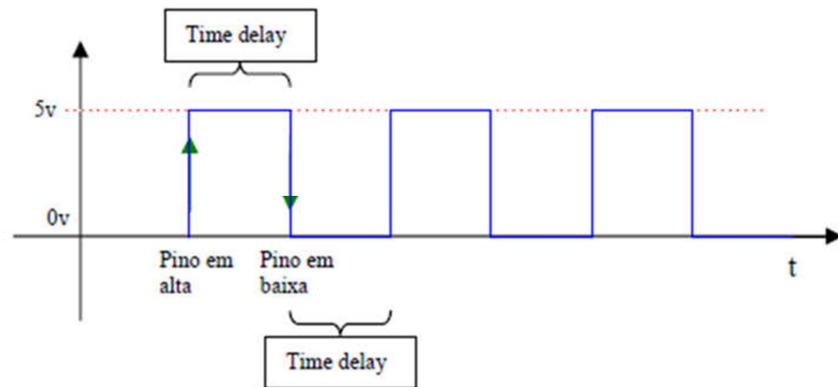
O programa utiliza o sensor com uma resolução de 9 bits, isso lhe dá uma precisão de 0.5 graus centígrados e não compromete muito a temporização do firmware, pois o tempo de leitura é de apenas 93.75ms.

3.4.6 Buzzer

Para este projeto foi criada uma biblioteca própria personalizada incluindo as funções necessárias para a utilização do alto-falante. Na figura 22 é mostrado o sinal do buzzer.

Os sons de alarme, bipes e avisos foram compostos utilizando sons de frequências de notas musicais, para que sejam mais agradáveis aos ouvidos do usuário. Para gerar notas musicais utilizando onda quadrada, sem a necessidade de um circuito oscilador adicional, foi preciso programar os intervalos de tempo com o High e Low comutando de um pino de I/O escolhido para essa finalidade.

Figura 22 - Sinal para o buzzer.



Fonte: Autoria própria.

O escopo deve ter a seguinte sequência:

Função nota:

```

Faça i 0 até duração:
    pinoA = HIGH;
    tempo_delay;
    pinoA = LOW;
    tempo_delay;

```

```

fim loop;

```

Fim função

Para ajustar a frequência de acordo com as notas musicais, deve-se geou-se uma tabela de frequências baseada em uma nota de referência (por exemplo a nota Lá 440Hz) escolher a escala musical e a oitava. Uma oitava é composta por 12 semitons.

Para gerar as frequências de todos os semitons se utilizou a fórmula da figura 19.

$$K = e^{\frac{\log(2)}{12}} \quad (11)$$

$$F(n) = \frac{T_0}{K^n} \quad (12)$$

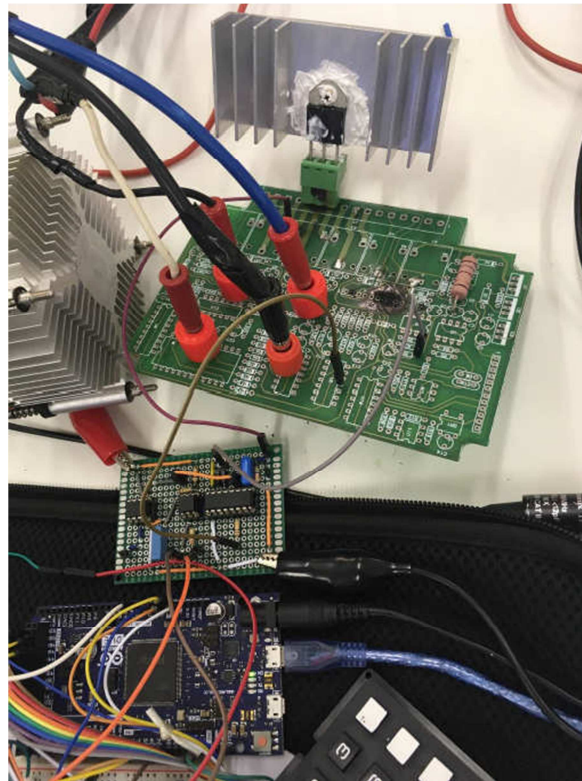
Com as frequências de todas as notas calculadas, foi possível compor os sons e bipes para a utilização no produto, que possui sons para indicar erros de utilização, confirmações, avisos, indica teclas pressionadas (som do teclado) e alarmes.

3.5 MONTAGEM

O sistema foi montado em duas placas perfuradas, uma para o circuito de baixa potência e outra para o de alta potência.

O TCA785, o amplificador operacional e o acoplador óptico foram montados na placa de baixa potência (no centro da figura 23). Já a placa de alta potência é composta pelo triac, com seu respectivo dissipador térmico (parte superior da figura 23), pelo resistor R6 (figura 11) e por quatro conectores nos quais são ligadas ambas as fases da rede elétrica (220V) e ambos os terminais da panela (resistência de aquecimento).

Figura 23 - Montagem do sistema.



Fonte: Autoria própria.

3.7 CONTROLE

3.7.1 PWM

PWM é técnica mais comumente utilizada para fazer controle de potência e o controle digital em malha fechada. Por ser uma ferramenta de fácil programação, ainda mais em conjunto com controlador PID, o PWM foi utilizado como a saída, $U[n]$, da equação a diferenças utilizada para implementar o PID utilizado.

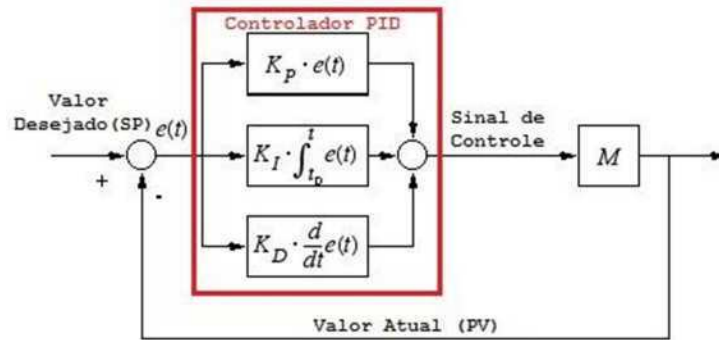
3.7.2 PID

O compensador PID que, no domínio do tempo, possui a seguinte equação:

$$U(t) = K_p e(t) + K_I \int_{t=0}^t e(t) dt + K_D \frac{de(t)}{dt} \quad (11)$$

Como é mostrado, também, no diagrama de blocos, na figura 24.

Figura 24 – Diagrama de blocos de um sistema em malha fechada PID.



Fonte: Notas de Aula, Prof. Carlos Raimundo Erig Lima.

Dado que a Planta M já foi equacionada anteriormente (seção 2.2), temos a equação (7):

$$\frac{T_o}{q_i} = \frac{R_{ie}}{1+SCR_t}$$

Que indica um sistema de primeira ordem, com um polo lento, uma vez que CR_t depende de várias grandezas físicas de respostas lentas, tais como: Capacitância térmica do ar, da água, calor específico da água. Sendo assim, é confortável usar aproximações matemáticas - para o uso do PID num sistema digital - mais simples. Como a aproximação retangular, contemplada na equação (12).

$$U[n] = K_p e[n] + K_d (e[n] - e[n-1]) + K_I (e[n] + e[n-1]) \quad (12)$$

Onde cada constante (K_p , K_d , K_I) foi escolhida arbitrariamente para então ser testada em laboratório e ajustada conforme a necessidade.

3.8 INTERFACE COM O USUÁRIO

A interface com o usuário foi feita na caixa de controle logo ao lado direito da panela. Nela é possível observar por meio do display LCD o estado e o funcionamento do sistema, bem como configurar, iniciar, pausar, retomar e ajustar os elementos do procedimento. Para a navegação entre os menus utilizou-se o teclado e suas teclas de direção. A tecla OK confirma a escolha ou avança uma tela, já a tecla ESC cancela uma escolha ou retrocede uma tela. A tecla ESC também é utilizada para pausar o procedimento a qualquer momento, nesse estado

o sistema mantém a temperatura constante do momento em que foi pausado. Ao pressionar ESC pela segunda vez o processo é retomado. Durante o estado pausado é possível também cancelar todo o procedimento e voltar ao menu principal apertando a tecla BACK (ou seta para a esquerda).

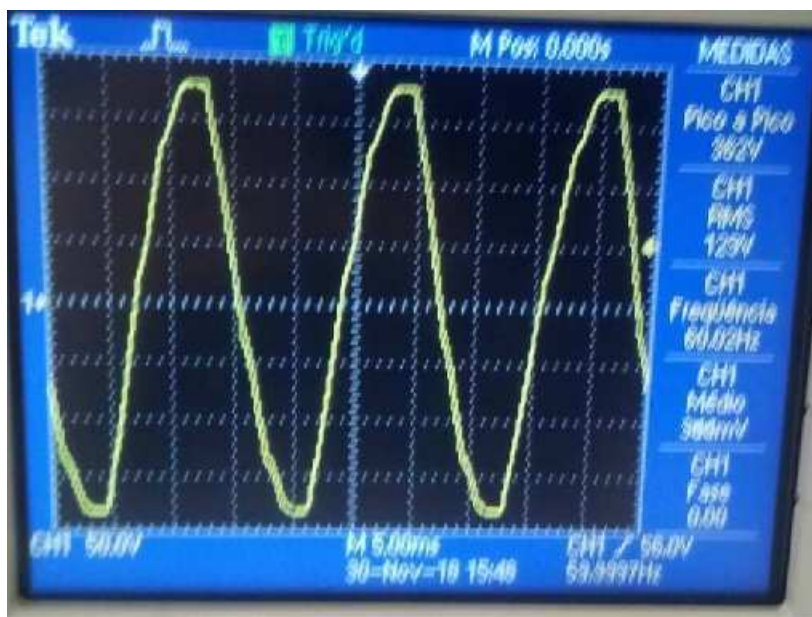
O produto também conta com diversos avisos sonoros para guiar o usuário sem a necessidade de que fique pendente observando o display. Existem basicamente três tipos de avisos sonoros: 1) do teclado, que realiza um bipe assim que uma tecla pressionada é reconhecida; 2) o de Alarmes, que indica a ocorrência de um evento importante, por exemplo ao atingir um patamar de temperatura desejado na rampa ou ao concluir algum processo; 3) os sons de Avisos, que indicam que algo foi bem ou mal sucedido, por exemplo algum erro ao configurar as temperaturas ou a confirmação de sucesso ao registrar o tempo do patamar. Todos esses avisos sonoros podem ser silenciados pelo usuário por meio do menu Ajustes no menu principal.

4. RESULTADOS

Neste capítulo são apresentados os resultados dos testes realizados que validam o funcionamento do protótipo desenvolvido. Com o intuito de verificar a funcionalidade do projeto para o controle de temperatura do mosto, foram realizados testes utilizando, apenas, água, já que ela possui um calor específico próximo a do mosto. Para a validação dos testes, foram capturadas as diversas tensões na carga (Resistencia de Inox) que foi aplicada pelo controlador PID—quanto acionado a atingir uma determinada temperatura.

A figura 25 mostra a tensão máxima sendo aplicada na carga.

Figura 25 - Tensão aplicada à carga pelo controlador PID.



Fonte: Autoria Própria.

Como no começo a temperatura atual está distante da desejada, fazendo com que o erro seja grande, o controlador aplica potência máxima na carga.

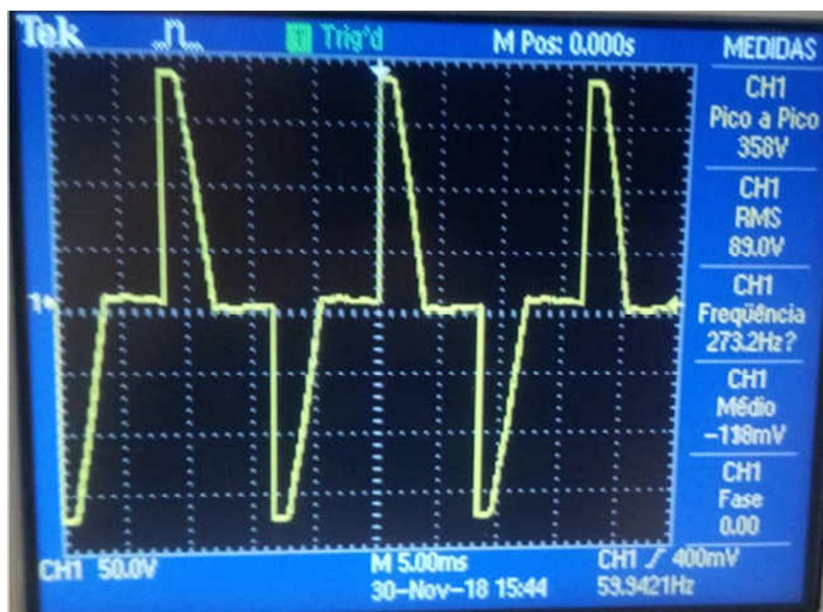
Mas conforme o erro vai diminuindo, menor será a tensão aplicada pelo controlador. E este padrão é seguido até a temperatura estabilizar-se, como mostra as figuras a seguir.

Figura 26 - Tensão aplicada à carga pelo controlador PID.



Fonte: Autoria Própria.

Figura 27 - Tensão aplicada à carga pelo controlador PID.



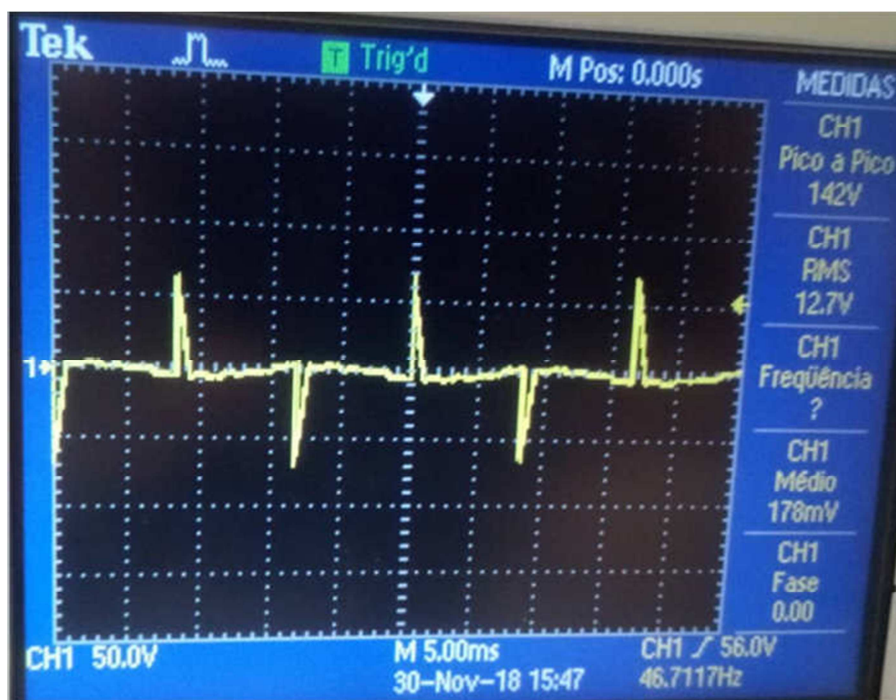
Fonte: Autoria Própria.

Figura 28 - Tensão aplicada à carga pelo controlador PID.



Fonte: Autoria Própria.

Figura 29 - Tensão aplicada à carga pelo controlador PID.



Fonte: Autoria Própria.

5. CONCLUSÃO

O objetivo principal deste Trabalho de Conclusão de Curso foi o desenvolvimento de um sistema que auxilie no processo de fabricação de cerveja artesanal. O sistema constitui-se de um carrinho com rodas, uma panela com uma resistência de aquecimento, uma segunda panela onde os grãos serão colocados, uma bomba d'água, dois solenoides e um braço aspersor para a condução da água e da mistura, bem como para a homogeneização da temperatura. Também consiste de um circuito eletrônico para regulação da potência e de uma placa Arduino com um microcontrolador para o controle e administração do sistema como um todo. Como a fabricação de cerveja artesanal é uma arte e cada fabricante tem seu estilo e preferências próprios, foi desenvolvido um firmware para suprir todas as necessidades do usuário quanto a personalização e improvisação deste processo.

A elaboração do projeto ocorreu conforme o previsto, embora, devido a alguns contratempos que encurtaram a margem disponível para a realização do produto, foi necessário reduzir ou abandonar algumas das funcionalidades que pretendia-se implementar, como por exemplo a implementação de conexões com dispositivos móveis para o usuário poder controlar as temperaturas e o sistema através de seu smartphone.

O circuito e a elaboração do sistema de potência geraram algumas surpresas, contudo foram rapidamente identificadas e resolvidas. O controle da potência na resistência de aquecimento ocorreu totalmente como planejado, bem como a implementação do firmware.

Com o sistema concluído foi possível esquentar e controlar a temperatura da panela de forma precisa indicando que é perfeitamente funcional e utilizável em um ambiente próprio para a fabricação de cerveja artesanal.

Com a conclusão desta versão do projeto percebe-se que há muito campo para trabalhos futuros, desde o aprimoramento das funcionalidades já implementadas, a confecção de PCBs próprias para o projeto para melhorar a estabilidade e otimizar espaço na caixa do sistema, bem como de criação de funcionalidades novas para tornar a experiência do usuário ainda mais prática e versátil. Também é possível ampliar o escopo do projeto e incluir também outros processos da fabricação, não apenas o de brassagem. Quanto à interface, também há muitos aprimoramentos que podem ser feitos, como o aplicativo de *smartphone* já comentado, a inclusão de um display gráfico, substituir o teclado por um display maior *touchscreen* por exemplo.

Tendo em vista o crescimento desta área de produção de cerveja e o interesse dos profissionais do setor, percebe-se uma oportunidade bastante expressiva da criação de uma versão comercial do projeto para atender essa demanda.

REFERÊNCIAS

CAMPOS, Paulo B. : **Notas de aula**, 2012. Disponível em: <<http://pessoal.utfpr.edu.br/brero>>. Acesso em: Dezembro 2018.

CRENSHAW, Jack W. : **Math toolkit for real time programming**, 2001.

HUGHES, Greg. : **Cerveja feita em casa**, 2014.

LIMA, Carlos E. : **Notas de aula**, 2015. Disponível em : < <http://pessoal.utfpr.edu.br/erig>>. Acesso em: Fevereiro 2019

POTTKER, Fabiana. : **Notas de aula**, 2014. Disponível em : < <http://pessoal.utfpr.edu.br/pottker> >. Acesso em: Novembro 2018

MEYRATH, Todd : **Multipurpose analog PID controller**, 2005.

NEVES, Felipe. : **Controlador PID digital, uma modelagem prática**,2014. Disponível em : <<https://www.embarcados.com.br/controlador-pid-digital-parte-2/>>. Acesso em: Fevereiro 2019.

PALMER, John J. : **How to brew**, 2006.

POLEY, Richard : **Introduction to digital control using digital signal processors**, 2003.

STARR, Gregory P. : **Introduction to applied digital control** , 2006.