

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA

ESTEVAN ROCHA PITHAN

Controle de Brassagem Automatizado de uma *Single Vessel* Utilizando Microcontrolador
Esp32 com o aplicativo Blynk

Uberlândia

2022

ESTEVAN ROCHA PITHAN

Controle de Brassagem Automatizado de uma *Single Vessel* Utilizando Microcontrolador
Esp32 com o aplicativo Blynk

Trabalho de Conclusão de Curso apresentado à
Faculdade Engenharia Elétrica da Universidade
Federal de Uberlândia como requisito parcial
para obtenção do título de bacharel em
Engenharia Elétrica.

Orientador: Gustavo Brito de Lima

Uberlândia

2022

ESTEVAN ROCHA PITHAN

Controle de Brassagem Automatizado de uma *Single Vessel* Utilizando Microcontrolador
Esp32 com o aplicativo Blynk

Trabalho de Conclusão de Curso apresentado à
Faculdade Engenharia Elétrica da Universidade
Federal de Uberlândia como requisito parcial
para obtenção do título de bacharel em
Engenharia Elétrica.

Uberlândia, data

Banca Examinadora:

Dr. Vítor Fonseca Barbosa (UFU)

Dr. Antônio de Oliveira Costa Neto (UFU)

Prof. Dr. Gustavo Brito de Lima (UFU) (Orientador)

Dedico este trabalho aos meus pais, pelo estímulo,
carinho e compreensão.

AGRADECIMENTOS

Aos meus pais Sérgio e Maria Aparecida, por todo o apoio e pela ajuda, que muito contribuíram para a realização deste trabalho e pela minha formação. A Rafaela, minha namorada, por todo carinho e paciência durante essa trajetória. Aos meus colegas de curso, com quem convivi intensamente durante os últimos anos, pelo companheirismo e pela troca de experiências que me permitiram crescer não só como pessoa, mas também como formando.

RESUMO

A produção de cerveja artesanal está em alta. Muitos apreciadores dessa bebida procuram estilos e produtos que tenham sabores e aromas diferentes das tradicionais cervejas padronizadas. A cerveja produzida nas casas dos *hobbistas* está cada vez mais atingindo patamares de excelência, devido ao estudo de cada processo de produção de modo independente. A melhoria e principalmente a simplificação de técnicas já utilizadas nas grandes cervejarias merece um destaque para a produção de cervejas diferenciadas. Uma das causas que torna possível a simplificação é a produção em menor escala. A finalidade desse estudo é aliar a melhoria da eficiência de algumas etapas do processo de fabricação de cerveja artesanal com o tempo gasto, automatizar o máximo possível esses mesmos processos. Esses objetivos foram alcançados controlando a potência aplicada ao líquido através de equipamentos eletroeletrônicos desenvolvidos ao longo do projeto. A Brassagem, também conhecida como etapa quente, teve sua eficiência e seu tempo total melhorados com o controle de potência na Mosturação e na Fervura do mosto cervejeiro. Foi utilizado o microcontrolador ESP32 com a plataforma de prototipagem *Dev Module* automatizando o monitoramento e controle de temperatura no tempo. Utilizando o aplicativo *Blynk*, para *Android*, como interface de usuário para controle das várias etapas que exigem diferentes parâmetros.

Palavras-chave: Eficiência de Brassagem; Automação; Equipamento cervejeiro.

ABSTRACT

Craft beer production is booming. Many connoisseurs of this drink are looking for styles and products that have flavors and scents that are different from traditional standardized beers. The beer produced in the homes of hobbyists is increasingly reaching levels of excellence, due to the study of each production process independently. The improvement and especially the simplification of techniques already used in large breweries deserves a mention for the production of differentiated beers. One of the causes that makes simplification possible is production on a smaller scale. The purpose of this study is to combine the improvement of the efficiency of some stages of the craft beer manufacturing process with the time spent, automating as much as possible these same processes. These objectives were achieved by controlling the power applied to the liquid through electro-electronic equipment developed throughout the project. Mashing, also known as the hot stage, had its efficiency and total time improved with the power control in the Mash and Boiling of the brewer's wort. The ESP32 microcontroller was used with the Dev Module prototyping platform, automating time and temperature monitoring and control. Using the Blynk app for Android as a user interface to control the various steps that require different parameters.

Keywords: Brewing Efficiency; Automation; Brewery equipment.

LISTA DE ILUSTRAÇÕES

Figura 1 - Grão de cevada em processo de germinação para o processo de malteação	15
Figura 2 - Malte de cevada moído pronto para a mosturação	16
Figura 3 - Adição de lúpulo no mosto fervente	20
Figura 4 - Esquema de uma tina durante a realização do whirlpool.....	21
Figura 5 - Processo de filtragem dos grãos utilizando um sistema BIAB.....	24
Figura 6 - Sistema Single Vessel de 100 litros.....	25
Figura 7 - Sistema Tri Bloco caseiro	26
Figura 8 - Tela inicial do <i>Brewfather</i>	28
Figura 9 - Modelos de poços termométricos.....	29
Figura 10 - Modelos de resistência elétrica utilizadas. Esquerda: 2500 watts. Direita: 3500 watts.....	31
Figura 11 - Peças de conexão de aço inox.....	31
Figura 12 - Projeto da panela de mostura	32
Figura 13 - Panela de mostura montada	32
Figura 14 - Panela de água montada	33
Figura 15 - Modelos de chiller de placas utilizado.	34
Figura 16 – Bomba Magnética MP-15R	35
Figura 17 - (a): Cesto de grãos utilizado. (b): Malha do cesto	36
Figura 18 - (a): Cesto de grãos dentro da panela, etapa da mosturação. (b): Cesto erguido para filtrar grãos do mosto e pronto para realizar a etapa de lavagem	36
Figura 19 - Esquema do sistema de controle de temperatura.....	38
Figura 20 - Plataforma de prototipagem NodeMCU-32S	39
Figura 21 - Interface do Arduino (IDE)	40
Figura 22 - Sensor DS18B20 aprova d'água.....	41
Figura 23 – Esquema de comunicação do <i>Blynk</i>	43
Figura 24 - (a): Tela para criação de um novo projeto. (b): <i>dashboard</i> do projeto.....	43
Figura 25 - (a): Tela para a escolha dos pinos digitais. (b): Tela para a escolha dos pinos virtuais.	45
Figura 26 - Telas de configuração do <i>Labeled Value</i>	45
Figura 27 - Telas de configuração do <i>Numeric Input</i>	46
Figura 28 - Telas de configuração do <i>Slider</i>	46
Figura 29 - Circuito detector de passagem por zero.....	49

Figura 30 - Comparação entre o sinal da rede (Amarelo) com a saída do circuito detector de passagem por zero (Azul).	50
Figura 31 - Circuito com microcontrolador.....	51
Figura 32 - Circuito com microcontrolador.....	52
Figura 33 – Comparação de uma senoide pura com uma senoide que teve um atraso de disparo.	54
Figura 34 – Comparação entre uma senoide pura e outras duas senoides com tempos de disparo diferentes	54
Figura 35 - Tela de edição de perfil de equipamento do <i>software Brewfather</i>	57
Figura 36 - Tela de configuração da receita.	59
Figura 37 - Maltes colocados no cesto para o começo da mosturação.....	61
Figura 38 - Sistema de recirculação durante a mosturação.	61
Figura 39 - Sistema de lavagem do bagaço de malte.	62
Figura 40 - Sistema de resfriamento do mosto junto com a decantação do mesmo.....	63
Figura 41 – Gráfico da variação da temperatura durante a Brassagem do primeiro teste.....	67
Figura 42 – Gráfico da variação da temperatura durante a Brassagem do segundo teste	69
Figura 43 – Gráfico da variação da temperatura durante a Brassagem do Terceiro teste.	70
Figura 44 – Gráfico da comparação das eficiências de mostura	71
Figura 45 – Gráfico da comparação das eficiências do equipamento durante cada teste.....	72
Figura 46 – Gráfico da comparação dos tempos de Brassagem de cada teste.	73

LISTA DE TABELAS

Tabela 1 - Faixas de temperatura para gelatinização do amido em grãos não maltados	17
Tabela 2 - Faixas de temperatura e pH relacionada a ação enzimática do mosto.	18
Tabela 3 - Dados configurados no <i>Brewfather</i>	57
Tabela 4 - Estatísticas vitais do guia BJCP	59
Tabela 5 - Valores previstos pelo <i>software</i> para o resultado da receita.....	60
Tabela 6 - Valores mensurados no primeiro teste.	66
Tabela 7 - Eficiências do primeiro teste.	66
Tabela 8 - Valores de tempo e temperaturas do primeiro teste	66
Tabela 9 - Valores mensurados no segundo teste.....	67
Tabela 10 - Eficiências do segundo teste.....	68
Tabela 11 - Valores de tempo e temperaturas do segundo teste.....	68
Tabela 12 - Valores mensurados no terceiro teste.	69
Tabela 13 - Eficiências do terceiro teste.....	70
Tabela 14 - Valores de tempo e temperaturas do terceiro teste.....	70
Tabela 15 - Comparação dos resultados dos testes.....	71

LISTA DE ABREVIATURAS E SIGLAS

BIAB	Brew In A Bag
OG	Original Gravity
IBU	Internacional Bitter Units
IDE	Arduino Integrated Development Environment
IoT	Internet Of Things
TCC	Trabalho De Conclusão De Curso
BJCP	Beer Judge Certification Program
FG	Final Gravity
ABV	Alcohol By Volume
EBC	European Brewing Convention
IBU	International Bitter Unit

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 Motivações	13
1.2 Objetivos do trabalho	13
1.3 Metodologia	13
2 PROCESSO CERVEJEIRO.....	15
2.1 Brassagem.....	15
2.2 Malteação.....	15
2.3 Moagem do grão	16
2.4 Mosturação	17
2.4.1 Gelatinização	17
2.4.2 Enzimas e rampas de temperatura	18
2.5 Clarificação do mosto, Filtração e Lavagem do bagaço.....	19
2.6 Cozimento	19
2.7 Decantação.....	20
2.8 Resfriamento	21
2.9 Fermentação e maturação.....	21
3 METODOS DE BRASSAGEM E EQUIPAMENTOS.....	22
3.1 Infusão	23
3.2 Decocção	23
3.3 Rampas de temperatura.....	23
4 EQUIPAMENTOS.....	23
4.1 BIAB.....	24
4.2 Single vessel	24
4.3 Tri Bloco	25
4.4 Eficiência da Brassagem.....	26
5 PROJETO DO EQUIPAMENTO SINGLE VESSEL	29
5.1 Poço termométrico.....	29
5.2 Resistência Elétrica.....	30
5.3 Conexões E Valvulas Hidraulicas De Aço Inox De ½”	31
5.4 Painel De Mostura	31
5.5 Painel De Água	33
5.6 Chiller De Placas.....	33
5.7 Bomba Elétrica.....	34

5.8	Cesto De Grãos.....	35
6	SISTEMA DE CONTROLE DE TEMPERATURA	38
6.1	ESP32.....	39
6.2	Arduino IDE.....	39
6.3	Sensor De Temperatura DS18B20	40
6.4	Blynk	42
6.4.1	Configuração dos widgets Labeled Value	45
6.4.2	Configuração do widget Numeric Input	46
6.4.3	Configuração do widget Slider	46
6.4.4	Interação Blynk e Esp32	47
6.5	Circuito liga e desliga Sem Controle De Potência.....	49
6.6	Circuito liga e desliga com Controle De Potência.....	49
6.6.1	Detector de passagem por zero	49
6.6.2	Microcontrolador	51
6.6.3	Circuito de potência	52
6.6.4	Firmware do controle de potência	52
7	METODOLOGIA.....	57
7.1	Configurações do equipamento.	57
7.2	Receita: American Lager	58
7.2.1	Maltes e adjuntos:	59
7.3	Etapas de funcionamento do equipamento.....	60
7.4.1	Do primeiro teste	64
7.4.2	Do segundo teste.....	65
7.4.3	Do terceiro teste.....	65
8	RESULTADOS E DISCUSSÃO	66
8.1	Resultado do primeiro teste	66
8.2	Resultados do segundo teste.....	67
8.3	Resultados do terceiro teste	69
8.4	Comparação dos resultados	71
9	CONCLUSÃO.....	74
	REFERÊNCIAS.....	75
10	APÊNDICE A – ESQUEMÁTICO DO CIRCUITO.....	77
11	ANEXO A – FIRMWARE DO ESP32.....	78

1 INTRODUÇÃO

Não se sabe exatamente quando a primeira cerveja da história da humanidade foi feita, mas existem registros que por volta de 6000 a.C, na época das construções das primeiras cidades, a produção de bebidas fermentadas a base de cereais já era bem estabelecida. Essas bebidas são as precursoras da cerveja, implicando o surgimento da cerveja pode ser colocado entre os milênios 6 e 9 a.C., já que este último marca o início da cultura de cereais (Morado, 2009).

Apesar de seu surgimento ser tão antigo, foi somente a partir da Idade Média que a produção de cerveja deixou de ser uma atividade familiar para se tornar um ofício. Durante essa época que podemos ver o surgimento de três das quatro grandes escolas cervejeiras, a escola Belga, Inglesa e Alemã. Os mosteiros tiveram grande importância no desenvolvimento e documentação de técnicas de produção, como, por exemplo, a conservação a frio da bebida. Durante a Renascença, a introdução de conceitos do capitalismo e a crescente urbanização afetaram as técnicas de produção, trazendo principalmente um aumento do volume produzido. Por volta do ano 1400, o lúpulo passou a ser mais amplamente utilizado, devido a suas propriedades conservativas. Durante os séculos seguintes, a produção de cerveja passou por modificações oriundas de diversas mudanças comportamentais e avanços tecnológicos (Morado, 2009).

Atualmente tem-se observado um relevante aumento no consumo de cervejas artesanais e caseiras, as quais se caracterizam por ser um produto fabricado em menor escala. Todo o processo de fabricação é acompanhado pelos produtores, prezando a qualidade da cerveja (Morado, 2009). Devido ao gosto apurado de um público de consumidores que procuram por uma cerveja com mais nuances de sabores, cores e aromas, o hobby de produção caseira de cerveja tem crescido em grandes proporções nos últimos anos, inclusive no Brasil, uma vez que, embora os processos industriais de fabricação de cerveja em larga escala sejam responsáveis por cerca de 98,6 % do mercado nacional, os produtos ofertados são muito semelhantes entre si (SEBRAE, 2015). O processo de produção caseira é relativamente mais simples. Equipamentos são improvisados, etapas do processo são eliminadas e um controle adequado das etapas é negligenciado.

A Brassagem (do francês “*Brassage*” – ato de produzir cerveja) é a etapa que exige tempo e recursos do cervejeiro, além de ter enorme influência no produto final (Morado, 2009). Assim, à medida que o usuário ganha experiência com o *hobby*, naturalmente busca aprimorar sua técnica e equipamento, de forma a produzir cervejas melhores com maior facilidade, muitas vezes através de melhorias no processo de Brassagem.

1.1 Motivações

A Brassagem do cervejeiro caseiro iniciante pode ser demorada e com uma eficiência baixa, podendo esses fatores desestimular quem gostaria de se envolver com essa cultura cervejeira. Equipamentos automatizados, fáceis de manusear e com altas eficiências de Brassagem são encontrados facilmente no mercado, mas com preços elevados, fazendo com que até mesmo os cervejeiros caseiros mais experientes sintam receio em investir neles.

1.2 Objetivos do trabalho

O objetivo inicial deste trabalho é entender as etapas de fabricação de cerveja artesanal, com ênfase na etapa de Brassagem. Compreender quais fatores influenciam nessa etapa do processo de fabricação de cerveja, o objeto de estudo serão as eficiências de mostura e do equipamento, além do tempo total de Brassagem de um sistema *Single Vessel* e sua abrangência para as demais etapas da produção de cerveja artesanal.

O objetivo final é o desenvolvimento e construção de um projeto automatizado de um sistema *Single Vessel* de baixo custo, adequado para todas as etapas quentes do processo de produção de cerveja, da Brassagem até o resfriamento, por meio de um aplicativo e um painel de controle que para o controle da temperatura, bomba e resistências elétricas utilizadas no projeto. Serão realizados testes para o levantamento dos valores das eficiências relacionadas à Brassagem e quanto ao tempo total desta etapa.

1.3 Metodologia

A realização de um projeto de um sistema *Single Vessel* exige, primeiramente, um estudo completo das etapas de produção de cerveja artesanal, com o foco principal na Brassagem, mais conhecida como etapa quente do processo de fabricação de cerveja. Com esse estudo espera-se também a possibilidade de fazer uma avaliação crítica dos resultados visando a possibilidade de realizar melhorias no projeto para obter melhor eficiência energética, na qualidade final do produto e no tempo. Uma melhoria nas etapas deve ser conseguida automatizando os processos.

Foi realizado um estudo de outros sistemas de produção de cerveja como BIAB, Herms e Tri-bloco para a retirada de ideias para melhorar a implementação do sistema *Single Vessel*.

Com o projeto definido começou-se a procura por materiais com preço acessível, algumas partes da estrutura foram adquiridas em ferros-velhos por preços mais acessíveis, em contrapartida foi necessária a polir e realizar a solda de algumas barras. Em paralelo com essa procura de matérias, o primeiro projeto de *hardware* junto com o firmware começaram a serem feitos e testados com lâmpadas incandescentes para simular a carga.

Com a estrutura e as painelas prontas deu-se início aos testes de vazamento e aquecimentos. Com todo o projeto pré-ajustado iniciou-se o primeiro teste já realizando a Brassagem.

2 PROCESSO CERVEJEIRO

O processo de produção de cerveja vem se sofisticando desde sua descoberta milhares de anos atrás, com o avanço da qualidade de insumos e equipamentos é possível produzir levas em casa apresentando uma qualidade comparável à de grandes cervejarias. Fazer cerveja em casa, também chamado de *Homebrew* pode ser dividido em duas etapas: Brassagem e fermentação.

2.1 Brassagem

Conhecida industrialmente como Brassagem, é a primeira etapa de produção de cerveja, na qual ocorre a produção do mosto cervejeiro, um líquido rico em açúcares fermentáveis e outros nutrientes para a levedura, tais como, cálcio, aminoácidos, que será fermentado para a produção da cerveja. Antes da produção do mosto, há a etapa de malteação da cevada, mas poucos cervejeiros caseiros a fazem, devido ao alto custo de equipamentos e demanda de espaço para a instalação dos mesmos, como alternativa os maltes podem ser comprados em lojas específicas de insumos cervejeiros, também chamadas de *Brew Shops* podendo até mesmo ser comprado já moído, por isso diz-se que a Brassagem é a primeira etapa da produção caseira. A produção do mosto pode ser dividida basicamente nas etapas de moagem do grão, mosturação, clarificação do mosto, cozimento, decantação e resfriamento.

2.2 Malteação

Maltar é o processo no qual o grão é umidificado para iniciar o processo de germinação. Quando a semente germina, ela ativa enzimas que começam a converter suas reservas de amido e proteínas em açúcares e aminoácidos, os quais serão utilizados no crescimento da planta. O objetivo de maltar um grão é liberar estas enzimas para sua utilização pelo cervejeiro. Uma vez que as enzimas começam a agir, o grão que começou a brotar é seco num forno, "desativando" as enzimas do malte, até que o cervejeiro esteja pronto para usá-lo. (PALMER, 2017)

Figura 1 - Grão de cevada em processo de germinação para o processo de malteação



Fonte: Homini Lúpulo (2019)

2.3 Moagem do grão

De acordo com Krottenhaler et al. (2009), a moagem do grão de malte é realizada por um processo mecânico e tem influência direta sobre os resultados de rendimento de sacarificação, velocidade de mostura e clarificação do mosto, além de afetar os resultados do produto acabado. O malte deve ser quebrado em partes menores para dar condições da ativação enzimática, durante a mostura, e permitir que essas enzimas atuem no amido, quebrando-o em açúcares menores.

Figura 2 - Malte de cevada moído pronto para a mosturação



Fonte: Cerveja Henrik Boden (2009)

2.4 Mosturação

Durante a mosturação, água, malte e adjuntos moídos são misturados para que o amido e proteínas presentes no malte sejam degradados através da ação de enzimas e estes produtos sejam solubilizados no mosto para que se obtenha o chamado extrato. Cada enzima atua com uma faixa de temperatura e pH ideal, cabendo ao cervejeiro criar a melhor situação possível de acordo com seu objetivo, escolhendo rampas de temperatura e selecionando qual enzima ele quer que atue. As rampas mais utilizadas são a gelatinização, parada proteica, sacarificação e desativação das enzimas.

2.4.1 Gelatinização

De acordo com PALMER (2017) a gelatinização é o processo de solubilizar amidos em água por calor ou por uma combinação de calor e ação enzimática. A maioria dos grãos não maltados armazenam seus amidos em grânulos que não são liberados só por infusão em água quente. No entanto dentro de um intervalo, e uma faixa específica de temperatura o amido vai gelatinizando e perde sua estrutura, facilitando assim a extração do amido e a ação enzimática. Cada grão possui sua faixa de temperatura para a gelatinização.

Tabela 1 - Faixas de temperatura para gelatinização do amido em grãos não maltados

Fonte de amido	Temp. de Gelatinização (°C)
Trigo sarraceno	61-65
Triticale	59-65
Aveia	52-64
Milho	62-77
Arroz	61-82
Painço	54-80
Cevada	60-62
Amaranto	59-70
Quínoa	57-64
Centeio	49-61
Trigo	52-66
Sorgo	69-75
Batata	56-71
Mandioca (polvilho)	63-80

Fonte: Homini Lupulo (2011)

2.4.2 *Enzimas e rampas de temperatura*

A atividade enzimática depende de alguns fatores como temperatura e pH (KUNZE, 1996). A atividade enzimática depende da temperatura do meio em que se encontra, à medida que a temperatura aumenta, a atividade aumenta, até atingir o ponto ótimo, onde a temperatura será a máxima possível, perto de 75°C, antes que a enzima sofra desnaturação. Cada enzima possui seu ponto ótimo de atuação e após esse ponto a degradação e inativação das enzimas torna-se cada vez maior assim que a temperatura do meio excede a temperatura máxima (KUNZE, 1996). A Tabela 2 demonstra os intervalos de temperatura nas quais as enzimas atuam durante a mosturação.

Tabela 2 - Faixas de temperatura e pH relacionada a ação enzimática do mosto.

Enzima	Temperatura ótima	pH ótimo	Função
Fitase	30-52°C	5,0-5,5	Abaixa pH da mostura
Debranching	35-45°C	5,0-5,8	Solubilização do amido
BetaGlucanase	35-45°C	4,5-5,5	Quebra dos glucano
Peptidase	45-55°C	4,6-5,3	Produz o Amino-Nitrogênio livre
Protease	45-55°C	4,6-5,3	Quebra de proteína grandes formadoras de turvação
Beta Amilase	55-65°C	5,0-5,5	Produz maltose
Alfa Amilase	68-72°C	5,3-5,7	Produz diversos açucares, incluindo a maltose

Fonte: PALMER (2017)

No processo de produção de cerveja caseira PALMER (2017) foca principalmente nas duas enzimas relacionadas a sacarificação do mosto, são elas a Beta Amilase e a Alfa Amilase. A sacarificação propriamente dita se inicia quando a enzima Alfa amilase continuamente degrada as cadeias de amilose e amilopectina em moléculas de dextrina contendo de 7 a 12 resíduos de glucose. Enquanto isso, a Beta amilase passa a atacar as pontas dessas novas cadeias menores, formando maltose (KUNZE, 1996). Após o final de mostura, espera-se que a sacarificação tenha ocorrido totalmente e que o amido tenha sido convertido em maltose e

dextrinas. É possível identificar se esse processo ocorreu totalmente através do teste do iodo, o qual é realizado com uma solução de 0,02M de iodo e iodeto de potássio. O teste se baseia no fato de a solução variar sua coloração quando em presença de iodo ou em presença de açúcares menores. Quando a solução não mudar de cor quando em contato com o mosto, é dito que a sacarificação ocorreu totalmente e a mosturação está encerrada (KROTTENHALER et al., 2009).

A temperatura de mostura que mais frequentemente se encontra na literatura está em torno de 67,2 °C. Isto é um compromisso entre as temperaturas mais favoráveis para as duas enzimas. A Alfa Amilase atua melhor entre 67,7 °C e 72,2 °C, enquanto que a Beta Amilase é desnaturada a essa temperatura. A Beta Amilase atua melhor entre 55 °C e 65,5 °C.

2.5 Clarificação do mosto, Filtração e Lavagem do bagaço

No final da etapa de mosturação, o mosto é formado por uma solução aquosa com substâncias dissolvidas, extrato, e não dissolvidas, chamada de bagaço. O bagaço é basicamente formado pela casca do malte e materiais insolúveis em água. O objetivo da clarificação do mosto é separar essa parte não solúvel deixando o mosto com a menor quantidade de sólidos sedimentáveis possível. Também extrair o máximo de extrato (mosto) do bagaço. A clarificação é realizada através do processo de filtração, no qual o elemento filtrante é o próprio bagaço podendo ser realizado de maneiras diferentes no contexto do *Homebrew*. No método BIAB utiliza-se um saco de pano para filtrar. No método do fundo falso utiliza-se um disco perfurado para separar o extrato do bagaço, sendo que o líquido é retirado da panela através de uma válvula e restando somente o bagaço na panela. O método do cesto de inox, mais utilizado em *Single Vessel*, é semelhante ao BIAB. O bagaço é retirado da panela através do elemento filtrante, nesse caso o cesto de inox. Muitos cervejeiros fazem essa etapa juntamente com a desativação das enzimas, onde a temperatura é elevada acima de 75°C provocando a desnaturação das enzimas e interrompendo a sacarificação. Ao final dessa etapa, ainda há resquícios de açúcares fermentáveis no bagaço, então é aplicada água aquecida no bagaço de modo que essa água esorra para o mosto. Esse procedimento é chamado de lavagem dos grãos.

2.6 Cozimento

Essa etapa consiste basicamente em ferver o mosto resultante da filtragem e pode levar de 60 a 120 minutos. Com a fervura ocorre a esterilização do mosto, eliminando contaminações

previamente ocorridas. Durante o processo ocorre a evaporação de substâncias indesejadas que podem causar sabores e aromas ruins, também evapora uma grande quantidade de água, acarretando um aumento da densidade inicial (OG – *Original Gravity*) do mosto, logo é possível atingir uma OG desejada aumentando o tempo ou a intensidade da fervura. Para expelir as substâncias indesejadas e também para evitar o boil-over (mosto sobe e vaza da panela) é aconselhado realizar a fervura com a tampa da panela aberta. Durante o processo de fervura ocorre também a caramelização de alguns açúcares (reação de Maillard) PALMER (2017). Após o início da fervura são adicionados os lúpulos de amargor, durante a fervura ocorre a isomerização dos alfa ácidos do lúpulo que definem o IBU (*Internacional Bitter Units*) da cerveja, medida que define o amargor da cerveja, quanto maior o IBU, maior a chance da cerveja ser amarga, dependendo da quantidade e dos tipos de açúcares extraídos. Para evitar a evaporação completa dos aromas, controla-se o tempo de fervura dos lúpulos aromáticos e especiarias para controlar o IBU da cerveja.

Figura 3 - Adição de lúpulo no mosto fervente

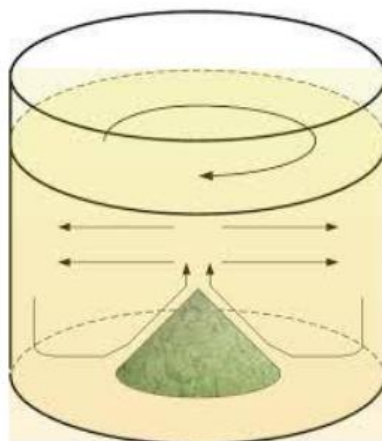


Fonte: Allbot (2021)

2.7 Decantação

O processo de cozimento forma um aglomerado de proteínas e restos de partículas que não são desejadas (*trub*). Para evitar que ocorra a transferência desse aglomerado para a etapa de fermentação, ao final da fervura pode ser realizado um movimento forte e circular (*whirlpool*) para decantar o *trub* no centro da panela, através de uma pequena bomba conectada à panela de fervura ou manualmente, como acontece na maioria das produções artesanais.

Figura 4 - Esquema de uma tina durante a realização do *whirlpool*



Fonte: Concerveja (2021)

2.8 Resfriamento

Essa etapa consiste em resfriar o mosto para diminuir e facilitar a decantação do material sólido (*trub*) proveniente da fervura, e para atingir uma temperatura de ação das leveduras. As leveduras em temperaturas altas, acima de 40 °C, não conseguem sobreviver. Quanto maior o tempo de resfriamento maior a probabilidade de contaminação do mosto. Para atingir a temperatura de inoculação da levedura no mosto e diminuir o tempo de um resfriamento natural existem equipamento e métodos de resfriamento, *chiller* de imersão, *chiller* de contra fluxo e *chiller* de placas.

2.9 Fermentação e maturação

Essa etapa consiste na fermentação dos açúcares pelas leveduras transformando-os em álcool e CO₂. Existem dois tipos básicos de fermentos: Ales e Lagers, e para cada um inúmeras variedades que proporcionam características específicas, em geral relacionadas ao aroma desejado na cerveja PALMER (2017). As fermentações Ale, de alta fermentação (*top*

fermented, receberam esse nome porque as leveduras agem no topo do mosto), trabalham a temperaturas de 15 °C a 23 °C e as fermentações Lager, de baixa fermentação (*bottom fermented*, recebem esse nome porque as leveduras afundam no tanque), trabalham a temperaturas de 8 °C a 15 °C e geralmente demoram mais que as Ales (Missão Sommelier, 2012). A fermentação se divide em três fases:

- *Lag time*: O fermento começa a se adaptar ao mosto e a capturar o oxigênio e aminoácidos presentes para poder iniciar a fase de crescimento.
- Crescimento exponencial: O fermento começa a consumir o açúcar do mosto e produzir álcool e CO₂, inicia também o crescimento exponencial da população do fermento.
- Fase estacionária: O crescimento do fermento desacelera e começa a sedimentar no fundo do recipiente usado como fermentador.

Para a etapa de fermentação normalmente é utilizado um *air lock*, que serve para liberar o CO₂ produzido bloqueando a entrada de quaisquer microorganismos e partículas.

Uma vez a fermentação esteja finalizada é necessário maturar a cerveja. A maturação consiste na consolidação dos sabores e aromas, fundamental para a cerveja atingir seu equilíbrio. A maturação complementa a etapa da fermentação e é geralmente feita em duas etapas: quente e fria. A etapa quente é a excitação extra das leveduras via temperatura (aumenta-se a temperatura) para que elas possam se alimentar de sub compostos da fermentação primária, além de evitar o aparecimento de diacetil principalmente nas cervejas do tipo *Lager*. Já a etapa fria a abaixa-se a temperatura e aos poucos as leveduras vão se decantando ao fundo do recipiente, com a temperatura mais baixa e ausência de “alimentos” para a levedura, o processo de fermentação vai se encerrando e o que sobrou flocula, acontecendo uma clarificação da cerveja. Durante a etapa de maturação também é possível a adição de novos ingredientes para aperfeiçoar o sabor e aroma desejado na cerveja. A técnica mais utilizada é o *dry-hopping*, que consiste na adição do lúpulo diretamente no maturador com o objetivo de intensificar o aroma. Muitas das vezes essa etapa é desconsiderada pelo cervejeiro caseiro e o envase ocorre diretamente após a fermentação

3 Métodos de Brassagem e equipamentos

A etapa de Brassagem permite manipular o processo visando criar um caráter desejável no mosto, resultado na cerveja esperada. As principais técnicas dessa etapa são: Infusão, Decocção e Rampas de temperatura.

3.1 Infusão

Este é o método mais simples da Brassagem e consiste em mesclar todo o malte moído com água quente, obtendo uma temperatura do mosto entre 65,5 °C e 70 °C, dependendo do estilo da cerveja (PALMER, 2017). A temperatura da água de infusão varia dependendo da relação de água e grãos usada para a Brassagem, em geral ela está um pouco acima da temperatura do mosto. Quase todo *software* utilizado para fabricar cerveja, como o *Beersmith* ou *Brewfather*, faz esse cálculo. Para manter uma temperatura estável é recomendado que a etapa de infusão seja realizada em um recipiente isolado termicamente. Neste método não há aquecimento após a mosturação ter começado.

3.2 Decocção

Na Decocção, ao invés de realizar infusões de água quente, parte do mosto é separada em outro recipiente, fervida e volta a ser adicionada ao mosto principal. Primeiramente a adição do mosto separado eleva a temperatura do mosto principal para o próximo descanso. O processo de fervura divide as moléculas de amido do grão não convertido, produz um maior grau de extração nos maltes, possibilitando um malte mais seco. Esse procedimento pode ser simples, duplo ou triplo em função do número de repetições. Um problema, que merece menção, nesse método é quanto a eficiência, uma vez que as enzimas presentes no mosto fervido são desnaturadas, afetam a sacarificação do mosto.

3.3 Rampas de temperatura

O método usa uma fonte de calor para esquentar o mosto em várias rampas de temperatura, cada faixa de temperatura é responsável pela atuação enzimática diferente. O processo possibilita personalizar a cerveja com mais facilidade. Esse método é o mais utilizado por cervejeiros caseiros no Brasil (AWADALLAK, 2020) e diferentes arranjos de equipamentos são comuns, dentre eles, temos o BIAB (*Brew in a Bag*), *Single Vessel* (evolução do BIAB) e o Tri Bloco. Para o projeto será adotado o sistema de *Single Vessel*.

4 Equipamentos

4.1 BIAB

Brew in a Bag (BIAB) é um sistema para produção de cerveja de grãos que teve origem na Austrália. O conceito por trás da "cerveja em um saco" (*Brew in a Bag*), é começar a fazer cerveja com grãos com o mínimo de equipamento adicional, ou menor tempo de preparo. Normalmente utiliza-se uma panela só para fazer toda a Brassagem. O método envolve o uso de um saco de grãos do tamanho da panela de fervura para misturar os grãos, seguido pela remoção do saco, deixando o mosto que é fervido com o restante dos ingredientes, como se faz com qualquer outro método de produção de cerveja. No método BIAB toda a água utilizada é colocada no começo da mostura, nesse método normalmente não há lavagem dos grãos. Embora menos eficiente do que os métodos tradicionais pode-se facilmente compensar utilizando maior quantidade de grãos.

Figura 5 - Processo de filtragem dos grãos utilizando um sistema BIAB.



Fonte: BrewBeer (2021)

4.2 Single vessel

Considerada uma evolução do BIAB, as *Single Vessel* possuem um sistema com um cesto de metal (Inox) no lugar do saco de pano, um sistema de recirculação do mosto, visando manter a uniformidade da temperatura em todo o recipiente, também vai clarificando o mosto durante o processo de mosturação. A eficiência desse sistema é mais alta do que a do BIAB, uma vez que há a lavagem dos grãos antes do início do cozimento do mosto. Em contrapartida o custo do equipamento é mais alto, uma vez que este possui mais acessórios do que o BIAB.

Figura 6 - Sistema Single Vessel de 100 litros.



Fonte: Heis Industrial (2021)

4.3 Tri Bloco

O sistema Tri Bloco contém três panelas, a mostura é feita na primeira panela, a filtração na segunda, e a última panela é destinada ao cozimento e decantação do mosto. Normalmente há uma quarta panela com água para lavar os grãos, mas ela considerada algo a parte. Esse sistema é considerado dentro do movimento *Homebrew* como o mais tradicional, uma vez que ele imita, de certo modo, o método de produção de cervejarias industriais que possuem diversas tinas. Industrialmente cada etapa é feita em diferentes tinas para brassagens simultâneas, o que aumenta consideravelmente o volume produzido. O rendimento do Tri Bloco tende a ser maior, pois sua lavagem dos grãos é mais eficiente e suas perdas de líquido são menores. Dos sistemas

citados este é o mais caro pelo número maior de panelas e por usar uma bomba hidráulica indispensável nesse sistema.

Figura 7 - Sistema Tri Bloco caseiro



Fonte: Concerveja (2021)

4.4 Eficiência da Brassagem

A eficiência de Brassagem é de suma importância para o processo de fabricação de cerveja artesanal. Esta eficiência sintetiza a capacidade de extração dos açúcares dos grãos e as perdas de volume do processo em relação ao seu potencial máximo. Uma alta eficiência no processo incorre em menores custos com insumos e um lote envasado maior. Os direcionadores dessa grandeza são os tipos e quantidades de grãos utilizados, densidade medida e volume medido no fermentador. A formulação parte da quantidade teórica máxima de pontos de densidade de acordo com as quantidades e tipos de malte usados visto que cada grão tem sua densidade potencial. Essa grandeza é definida como a densidade que pode ser obtida com uma libra de grãos em um galão americano de água (SMITH, 2008)

$$P_{máximo} = (SG_{potencialgrão} - 1) * 1000 * M_{grão}$$

Onde:

- $P_{\text{máximo}}$ = total de pontos de densidade potenciais
- SG potencial grão = densidade potencial do grão em pontos/libras
- $M_{\text{grão}}$ = massa do grão em libras

Dado um determinado volume alvo informado em galões americanos no fermentador, é calculada qual a densidade máxima em g/mL do volume.

$$OG_{\text{máxima}} = \frac{P_{\text{máximo}}}{V_{\text{mosto}}} + 1$$

Por fim, é utilizada uma proporção simples entre a $OG_{\text{máxima}}$ e V_{mosto} com a densidade medida (OG_{medida}) e volume medido (V_{medido}). Dessa forma, a eficiência de Brassagem é calculada.

$$E = \frac{(OG_{\text{medida}} - 1) * V_{\text{medido}}}{(OG_{\text{máxima}} - 1) * V_{\text{mosto}}}$$

É importante notar que a relação utilizada das grandezas em unidades de medidas americanas, libras para massa e galões americanos para volume. A conversão dessa unidade de massa/volume para o S.I. é feita através da multiplicação pela constante 0,119826. Na prática, os cálculos foram realizados através do *software* que já possui as conversões e cálculos internalizados, sendo necessário apenas fornecer os dados de volumes, receita e densidades obtidas.

4.5 BREWFATHER

O *Brewfather* é um *software* para a fabricação de cerveja podendo a produção ser caseira ou até mesmo industrial. Ele pode ser executado no navegador de um computador ou através de um aplicativo para dispositivos móveis.

Esse *software* oferece diversas facilidades para o cervejeiro, tendo já pré-cadastrado diversos ingredientes e realizando cálculos de rendimento, cor, concentração de álcool e amargor e demais características importantes para o planejamento de uma boa receita e avaliação do equipamento.

O *software* oferece suporte para comunicação com alguns sistemas de controle de Brassagem, podendo assim as receitas e etapas de produção serem exportadas para os equipamentos.

Figura 8 - Tela inicial do *Brewfather*

The screenshot displays the Brewfather mobile application interface. On the left is a dark sidebar menu with the 'Brewfather' logo at the top. Below the logo, it shows 'Monitores Ativos (1)' with a power icon and 'Wit Hibisco #1' with 'Fervura 00:00' and 'Fim da Fervura'. The menu items include: Receitas, Lotes, Estoque, Biblioteca, Perfis, Estilos, Ferramentas, Configurações, Conta (with a Premium badge and email 'estevanpithan@hotmail.com'), and a 'Brewfather' section with links for Sobre, Ajuda, Contato, Alterações, and Termos. At the bottom of the sidebar is a Brazilian flag icon and 'version 2.8.1'. The main content area is titled 'Receitas' and features a search bar 'Pesquisar receitas...'. Below the search bar, it shows '- Outros (15)' and a list of 15 recipes, each with a beer glass icon, a title, a description, and technical specifications (ABV, OG, FG, P, IBU). The recipes listed are: APA competição Estevan, American Lager Estevan, American Lager, Belgian Golden Strong Ale, Belgian golden versão 2, Bohemian Pilsner Estevan, Doppelbock, Dry stout Estevan, Lager - Trabalho de Conclusão de Curso Estevan Pithan, Lutra formatura Dede, NEIPA single Nelson sauvin, Pilsen Estevan Pithan, Weiss Estevan, Witbier Estevan, and Witbier 2 Estevan.

Fonte: Autor (2022)

5 PROJETO do EQUIPAMENTO SINGLE VESSEL

Utilizou-se os seguintes itens para a montagem do equipamento

- 1 Panela de mostura de 35 litros.
- 1 Panela para aquecer água de 20L.
- Cesto de grãos
- 2 resistências elétricas de baixa densidade.
- 2 poços termométricos.
- Conexões e válvulas de aço inox de ½”.
- 1 Bomba Magnética.
- 1 Chiller de 30 placas.
- Mangueiras de silicone

5.1 Poço termométrico

Poço termométrico é uma peça essencial para o equipamento, pois através dele é possível fazer a instalação do sensor de temperatura, evitando o contato direto do sensor com o líquido dentro da panela. Além de evitar o contato direto do sensor com o líquido, o poço com um comprimento específico, permite posicionar o sensor para monitorar a temperatura em um ponto desejável do líquido.

Para as duas panelas foram escolhidos os poços de 150 milímetros de comprimento, colocados lateralmente, permitindo que o sensor vá até o centro da panela.

Figura 9 - Modelos de poços termométricos.



Fonte: Mercado livre (2021)

5.2 Resistência Elétrica

Assim como os outros equipamentos o material mais indicado é o aço inox. A escolha do material da resistência elétrica é muito importante. As ligas que contém chumbo não podem ser utilizadas devido à alta toxicidade deste elemento.

A resistência elétrica pode ser de baixa ou alta densidade. As resistências de baixa densidade possuem uma área de contato maior, conseguindo dissipar a mesma potência de forma espalhada. Consequentemente, a superfície da resistência de baixa densidade fica menos quente que uma resistência de alta densidade. A resistência elétrica que possui contato com o mosto, panela *Single Vessel*, deve ser de baixa densidade, com isso é evitada a caramelização do mosto.

A escolha da potência correta é fundamental. Há um consenso, em fóruns de cervejeiros caseiros, que para se obter o melhor tempo de aquecimento, sem causar caramelização do mosto na *Single Vessel*, são necessários em torno de 1000 watts para cada 10 litros. Para o projeto foi escolhido uma resistência para desenvolver uma potência de 3500 watts para a panela de mostura. A resistência para desenvolver uma potência de 2500 watts foi melhor para a panela

de água. As duas painelas são feitas de aço inox, de baixa densidade e com tensão de alimentação igual a 220 volts.

Figura 10 - Modelos de resistência elétrica utilizadas. Esquerda: 2500 watts. Direita: 3500 watts.



Fonte: Mercado livre (2021)

5.3 Conexões E Valvulas Hidraulicas De Aço Inox De ½”

Todas as conexões hidráulicas são de aço inox. Para a extração do mosto ou da água das painelas, as conexões utilizadas na bomba de recirculação, no *chiller* de placas e nas válvulas que controlam a vazão da bomba.

Figura 11 - Peças de conexão de aço inox.



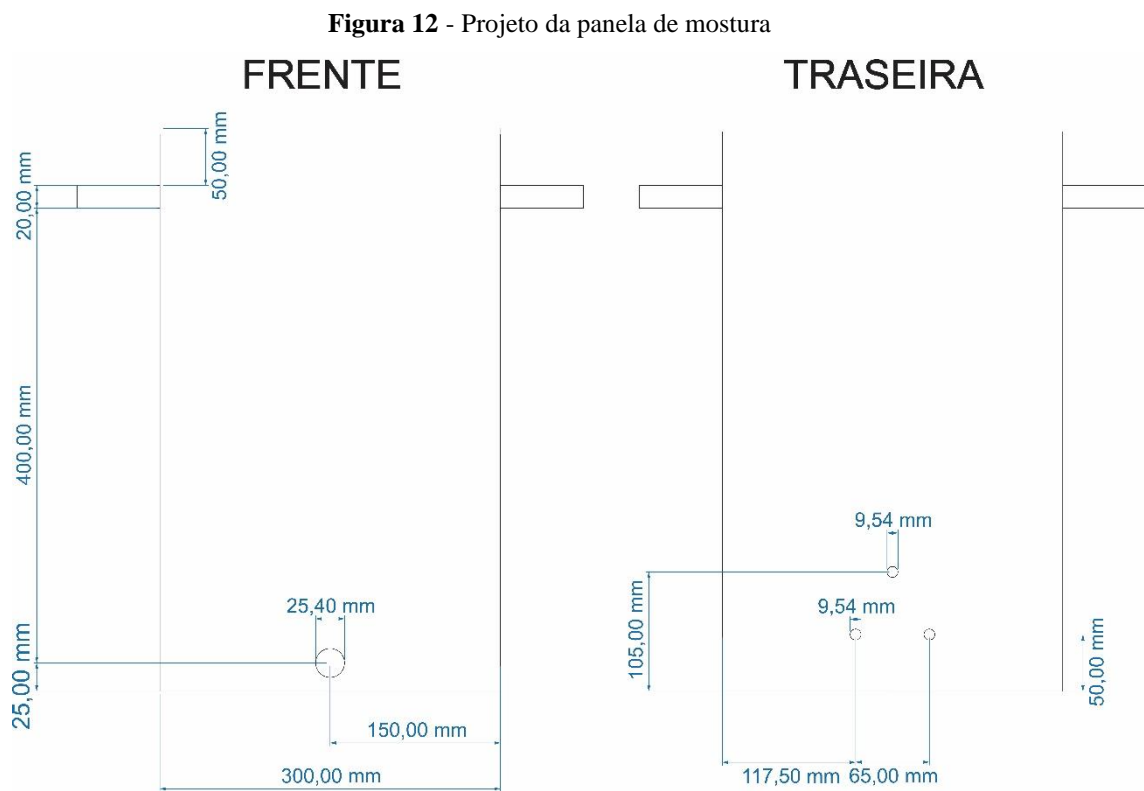
Fonte: Aços mota (2021)

5.4 Painela De Mostura

A panela utilizada para a panela de mostura do sistema Single Vessel nesse projeto foi feita de uma chapa de aço inox 304. A panela foi projetada para ter um volume de 35 litros e apresenta as medidas mostradas na Figura 12.

Na parte frontal foi feito um furo para que seja a saída de líquido da panela. Nesse furo foi adaptado uma conexão de $\frac{1}{2}$ ".

Na parte traseira foram feitos três furos, dois são para o encaixe da resistência e o terceiro para a instalação de um poço termométrico, onde será colocado o sensor de temperatura do sistema.



Fonte: Autor (2021)

O resultado final da panela com a resistência, poço termométrico e a conexão de $\frac{1}{2}$ " instalados ficou conforme a imagem abaixo.

Figura 13 - Panela de mostura montada



Fonte: Autor (2021)

5.5 Painel De Água

Para a panela de água de lavagem foi utilizada um modelo de 20 litros, feito de aço inox 201 vendido no mercado.

O mesmo procedimento de furação feito na panela de mostura também foi feito na panela de água. Foram feitos um furo na parte da frente para a instalação das conexões de ½” e três furos na parte de trás da panela sendo dois furos para a instalação da resistência elétrica e um furo para a instalação do poço termométrico. A montagem final da panela está representada da Figura 14.

Figura 14 - Painel de água montada



Fonte: Autor (2021)

5.6 Chiller De Placas

O ideal é que, após a fervura, o mosto seja resfriado, abaixo de 30 °C, o mais rápido possível. Pela eficiência no resfriamento, o *chiller* de placas é o dispositivo mais indicado para o processo, podendo ser encontrado com 10, 20, 30 e até 40 placas, atendendo a necessidade de cada projeto. Para este projeto, o *chiller* de 30 placas foi a melhor escolha.

Figura 15 - Modelos de chiller de placas utilizado.



Fonte: Mercado livre (2021)

5.7 Bomba Elétrica

Algumas etapas da produção de cerveja podem chegar a temperaturas elevadas, atingindo até a temperatura de ebulição da água. As bombas elétricas utilizadas para recirculação do mosto devem suportar essas altas temperaturas, além de serem adequadas para processos alimentícios.

Há vários modelos no mercado, as mais adequadas são as bombas com a cabeça em aço inox, porém os preços desses modelos são elevados. Como alternativa existem modelos de bomba elétrica que possuem a cabeça em nylon alimentício atóxico, reduzindo o custo da bomba em até 50 %. O modelo escolhido para o sistema da *Single Vessel* foi o Mp-15r da marca Mestre Brew

A seguir estão descritas algumas características listadas na placa da bomba:

- Bomba centrífuga com acionamento magnético
- Modelo: MP-15RM, Cabeça em Nylon Alimentício Atóxico. 1/2 - rosca BSP
- Temperatura máxima do líquido: 140 Graus Celsius
- Pressão: Baixa pressão

- Potência nominal: 10 watts
- Tensão: 220 volts 50 / 60 hertz
- Vazão máxima: 16 litros por minuto
- Vazão nominal: 8 litros por minuto
- Elevação máxima: 2,4 metros
- Elevação nominal: 1,5 metros

Figura 16 – Bomba Magnética MP-15R



Fonte: Cervejaria Inversa (2022)

5.8 Cesto De Grãos

A filtragem dos grãos é uma das etapas mais críticas de uma Brassagem, nessa etapa é onde os grãos são separados do líquido. Cada sistema realiza essa separação de uma maneira diferente. Para o sistema *Single Vessel* é utilizado um cesto em que as suas paredes e seu fundo são feitos com uma malha filtrante. O cesto escolhido para o projeto tem a capacidade de 7 quilogramas de malte molhado. A Figura 17 mostra a malha utilizada e a Figura 18 o posicionamento do cesto na panela.

Figura 17 - (a): Cesto de grãos utilizado. (b): Malha do cesto



Fonte: Autor (2021)

Ele irá ficar dentro da panela durante toda a mosturação e no fim do *Mash Out* esse cesto é erguido e preso a borda da panela para que o líquido esorra e que assim a lavagem dos grãos possa ser feita.

Figura 18 - (a): Cesto de grãos dentro da panela, etapa da mosturação. (b): Cesto erguido para filtrar grãos do mosto e pronto para realizar a etapa de lavagem

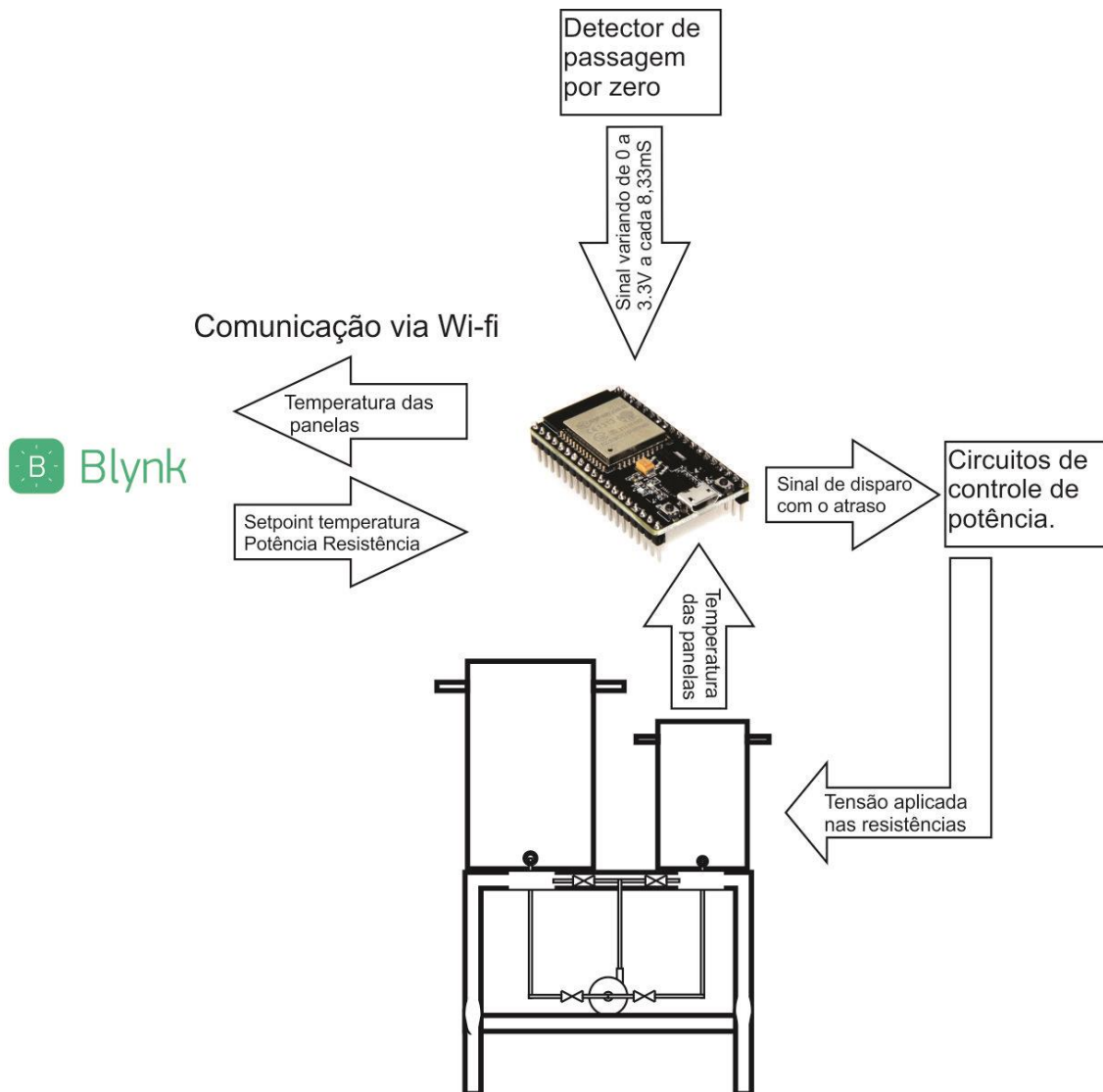


Fonte: Autor (2021)

6 SISTEMA DE CONTROLE DE TEMPERATURA

Durante o processo de mosturação é necessário controlar a temperatura da panela. Para isso foi criado um sistema de controle em malha fechada onde os parâmetros de *setpoint* de temperatura e potência aplicada à resistência elétrica são configurados através de um aplicativo para *Android* (*Blynk*). A leitura dos termômetros instalados nas panelas alimenta o microcontrolador e este envia esses valores para o *Blynk*. O controle é feito através de um sistema liga e desliga da alimentação das resistências para manter a temperatura em um valor desejado. Cada etapa do processo tem a sua temperatura e potência visando o melhor resultado possível. A Figura 19 mostra o esquema do sistema projetado.

Figura 19 - Esquema do sistema de controle de temperatura.



Fonte: Autor (2022)

6.1 ESP32

O ESP32 é um dispositivo muito utilizado na IoT (Internet das Coisas). A placa consiste de um microprocessador de baixo consumo de potência, dual core Tensilica Xtensa 32-bit LX6. A placa já inclui os circuitos para acesso à rede WiFi e ao protocolo Bluetooth v4.2. A memória flash integrada, complementa a arquitetura do ESP 32, possibilitando uma versatilidade muito grande ao microcontrolador.

Assim como outros microcontroladores populares como os Atmegas e PICs. O ESP32 possui também uma plataforma de prototipagem, no caso, o NodeMCU-32S. Uma de suas grandes vantagens é a possibilidade de ser programado utilizando a IDE (Ambiente de Desenvolvimento Integrado) do Arduíno.

Figura 20 - Plataforma de prototipagem NodeMCU-32S



Fonte: blogmasterwalkershop (2021)

6.2 Arduino IDE

O *Arduino Integrated Development Environment* - ou *Arduino Software* (IDE) - contém um editor de texto para escrever código, uma área de mensagem, um console de texto, uma barra de ferramentas com botões para funções comuns e uma série de menus. Ele se conecta ao *hardware* para fazer *upload* de programas e se comunicar com eles. Os programas escritos com o *software* Arduino (IDE) são chamados de *sketches*. Esses esboços são escritos no editor de texto e salvos com a extensão de arquivo *.ino*. A área de mensagem fornece *feedback* ao salvar e exportar e também exibe erros. Uma das grandes vantagens da utilização desse *software* é a variedade de bibliotecas e funções disponíveis e testadas.

Figura 21 - Interface do Arduino (IDE)



```

Controlador_Brassagem_Blynk | Arduino 1.8.15
Arquivo Editar Sketch Ferramentas Ajuda
Controlador_Brassagem_Blynk buzzer ctrl_Temp med_temp
#define BLYNK_PRINT Serial

/*BIBLIOTECAS*/
#include <WiFi.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

/*DECLARAÇÃO DOS PINOS*/
#define pinResPanela32L 13
#define pinResPanela20L 12
#define zeroPin 34
#define buzzer 5
#define oneWireTermometro 15

// //variaveis que indicam o núcleo
// static uint8_t taskCoreZero = 0;
// static uint8_t taskCoreOne = 1;
<
ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb), 921600, None em COM3

```

Fonte: Autor (2021)

6.3 Sensor De Temperatura DS18B20

Para a aferição das temperaturas das panelas foi escolhido o termômetro DS18B20. Segundo o datasheet do fabricante ele é um tipo de sensor com um range de trabalho de -55 °C a 125 °C. Também fornece leituras de temperatura convertidas em uma comunicação *One Wire* de 9 a 12 bits.

Esses valores mostram a temperatura de um determinado dispositivo. A comunicação *One Wire* deste sensor é feita através de um protocolo de barramento de fio único (e GND). Essa comunicação possibilita a conexão de mais de um sensor de mesmo modelo em um mesmo barramento, ou seja, vários sensores DS18B20 podem coexistir e funcionar simultaneamente através de uma única entrada digital do microcontrolador. A entrada escolhida nesse projeto foi a D15. Este sistema de comunicação mencionado pode ser utilizado apenas com este modelo de sensor de temperatura, uma vez que cada DS18B20 possui um código serial exclusivo de 64 bits que é configurado de fábrica e possibilita que o microcontrolador diferencie um sensor do outro de maneira individual, o que eleva consideravelmente as possibilidades de utilização e desenvolvimento de projetos.

Figura 22 - Sensor DS18B20 aprova d'água



Fonte: Mercado Livre (2021)

Para esse projeto foram escolhidas duas bibliotecas para auxiliar no uso desse sensor com a IDE do Arduino, são elas:

- OneWire.h
- DallasTemperature.h

Cada termômetro possui um endereço próprio. Há códigos e funções dentro dessas bibliotecas para ler esses endereços e assim utiliza-los. A configuração para a utilização dos termômetros está descrita no trecho de código abaixo.

```

OneWire oneWire(oneWireTermometro);
DallasTemperature sensors(&oneWire);
DeviceAddress sensorTempPanela35L = { 0x28, 0x19, 0x10, 0x79, 0x97,
0x20, 0x3, 0xE5 };
DeviceAddress sensorTempPanela20L = { 0x28, 0xEE, 0x1E, 0x79, 0x97,
0x20, 0x3, 0xB0 };

```

Para fazer a requisição das temperaturas e manda-las para o aplicativo *Blynk* foi criada uma função chamada “verifica_temp()” chamada por um timer.

Nessa função são requisitadas as temperaturas através do comando “sensors.requestTemperatures()” e após elas são armazenadas em duas variáveis globais. O envio de dados e finalmente a verificação do erro para o acionamento das resistências.

```

void verifica_temp() {
  sensors.requestTemperatures(); //Requisitando as temperaturas
  floatsensorTempPanela35L = sensors.getTempC(sensorTempPa-
nela35L); //Armazenando as temperaturas em variáveis globais
  floatsensorTempPanela20L = sensors.getTempC(sensorTempPanela20L);
  Blynk.virtualWrite(V0, sensors.getTempC(sensorTempPanela35L)); //En-
viando as temperaturas para o Blynk
  Blynk.virtualWrite(V3, sensors.getTempC(sensorTempPanela20L));

  if (floatsensorTempPanela35L < setTemp35L - 0.3 && map_potMax35L >
limMin) { //Fazendo a verificação
    flagRes35L = 1;
  } else {
    flagRes35L = 0;
  }
  if (floatsensorTempPanela20L < setTemp20L - 0.3 && map_potMax20L >
limMin) {
    flagRes20L = 1;
  } else {
    flagRes20L = 0;
  }
}

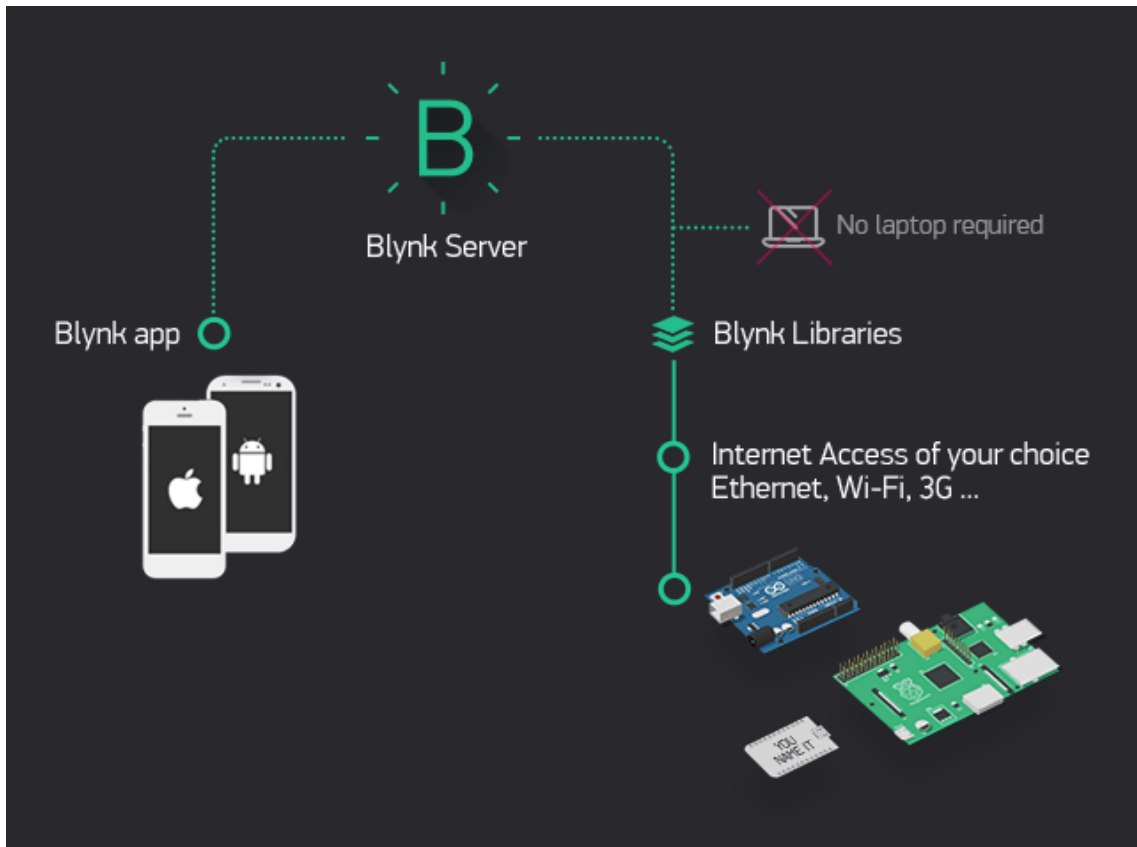
```

6.4 Blynk

O aplicativo Blynk está disponível para Android e iOS. O aplicativo permite ao usuário criar aplicações que enviam parâmetros e recebem dados de sensores através de um espaço próprio para cada projeto. O usuário pode inserir *widgets* que implementam funções de controle (como botões, *sliders* e chaves), notificação e leitura de dados do *hardware* (exibindo em

displays, gráficos e mapas). Toda comunicação entre o aplicativo e o *hardware* do usuário se dá através da *cloud Blynk*. O servidor da plataforma é responsável por transmitir, armazenar estados e dados de sensores lidos do aplicativo e do *hardware*, mesmo se o aplicativo estiver fechado.

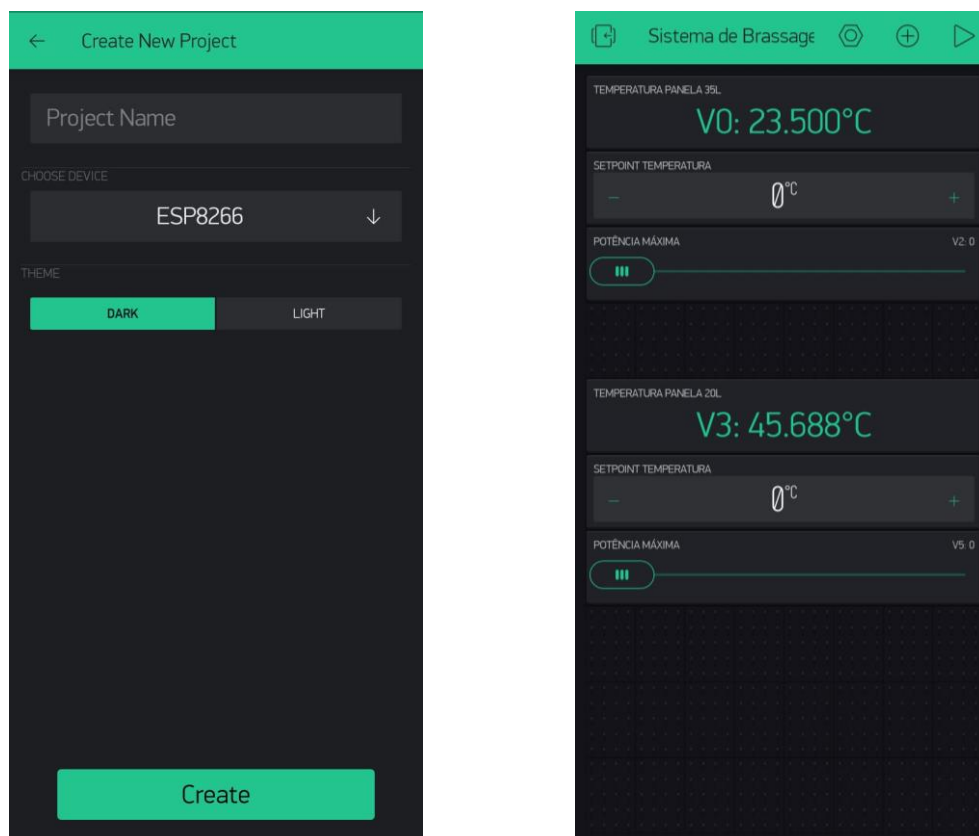
Figura 23 – Esquema de comunicação do *Blynk*.



Fonte: Blynk (2022)

Na tela inicial do aplicativo, o usuário pode ver todos os projetos feitos, ou criar um projeto novo. Ao escolher criar um novo projeto o usuário irá se deparar com a tela inicial de criação, nessa tela é possível escolher o nome do projeto e o microcontrolador que será utilizado, Figura 24 (a). Uma das grandes vantagens do Blynk é a compatibilidade com diversos microcontroladores diferentes, sendo uma ótima opção para a realização de projetos e protótipos de IoT. Após a escolha do nome e do microcontrolador a tela de montagem do projeto será aberta, também chamada de *dashboard*, Figura 24 (b).

Figura 24 - (a): Tela para criação de um novo projeto. (b): *dashboard* do projeto.



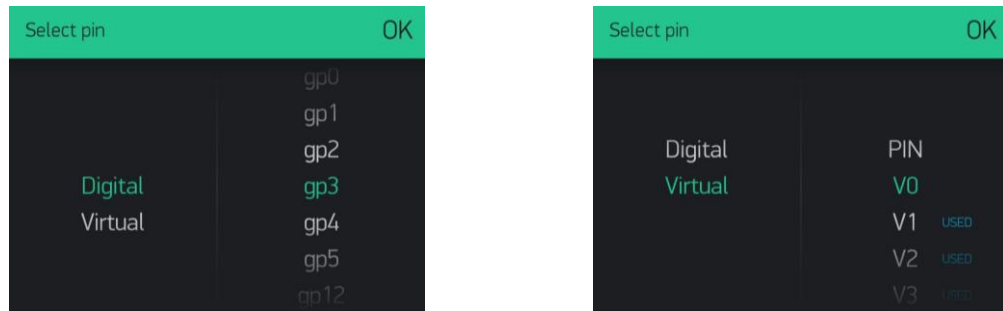
Fonte: Autor (2022)

Na *dashboard* é possível colocar *widjets* de diferentes tipos. *Widjets* são peças pré-projetadas da Interface Gráfica do Usuário. Cada *widjet* executa uma função específica de entrada/saída ao se comunicar com seu hardware ou usuário final. Há uma variedade grande para atender diferentes propostas de projetos, tendo opções em relação a conectividade, gráficos e até reprodução de vídeos para microcontroladores que possuem câmeras e muito mais.

Para o projeto da *single vessel* foram escolhidos três tipos diferentes o *Labeled Value* tem a função de exibir dados vindos de sensores ligados em entradas digitais do *hardware* ou pinos virtuais, o *Numeric Input* tem a função de entrada de valores numéricos a serem passados para o *hardware* e o *Slider* ele é similar a um potenciômetro, permitindo o usuário enviar um número entre o mínimo e o máximo estabelecido na configuração desse item.

Cada um dos *widjets* escolhidos deve ser vinculado a um pino como é chamado na tela de configuração. Esse pino é um nome que será utilizado para a comunicação com o hardware. O Blynk permite a escolha de pinos digitais e virtuais. Os pinos digitais estarão diretamente relacionados à uma entrada ou saída digital física do microcontrolador. Os pinos virtuais não possuem relação direta com nenhum pino físico do microcontrolador, podendo ser tratados como variáveis dentro do firmware. Para esse projeto, todos pinos dos *widjets* foram configurados como virtuais.

Figura 25 - (a): Tela para a escolha dos pinos digitais. (b): Tela para a escolha dos pinos virtuais.

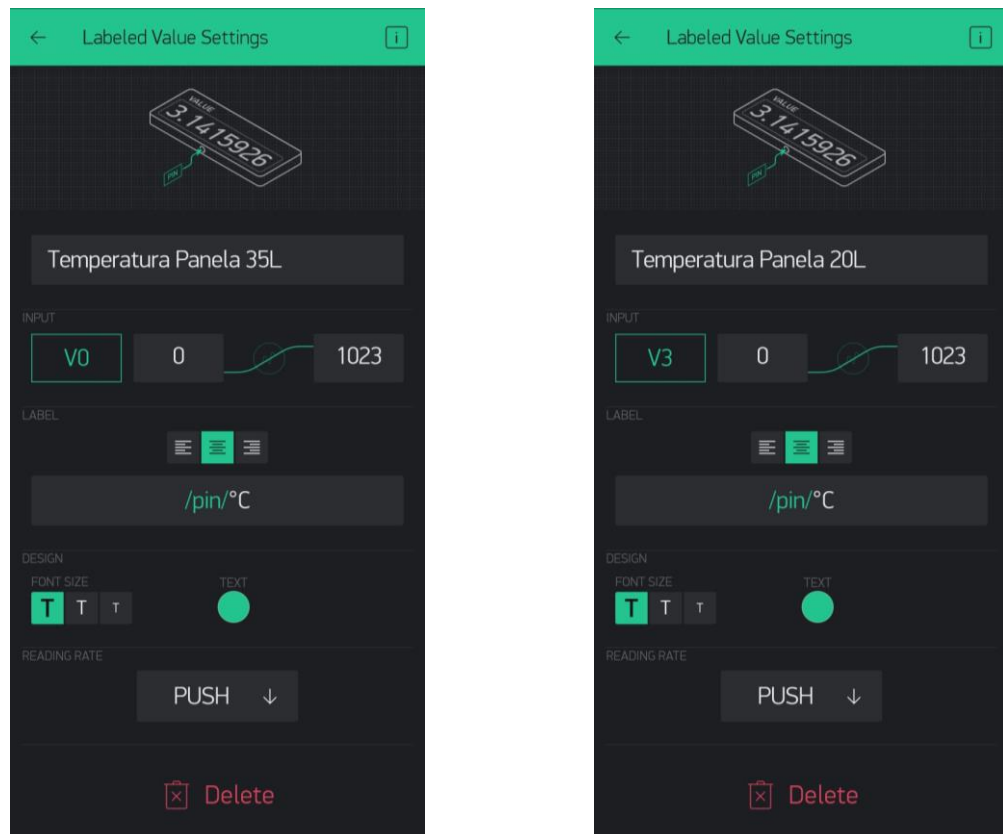


Fonte: Autor (2022)

6.4.1 Configuração dos widgets *Labeled Value*

Para esse *widget* foi dado o nome de “Temperatura Panela 35L” e “Temperatura Panela 20L” e os pinos virtuais utilizados foram o V0 e o V3 para as panelas de 35 litros e de 20 litros respectivamente. O campo de configuração *Reading Rate* foi configurado como *Push*, isso significa que o microcontrolador irá definir quando a atualização dos valores irá acontecer, Figura 26.

Figura 26 - Telas de configuração do *Labeled Value*

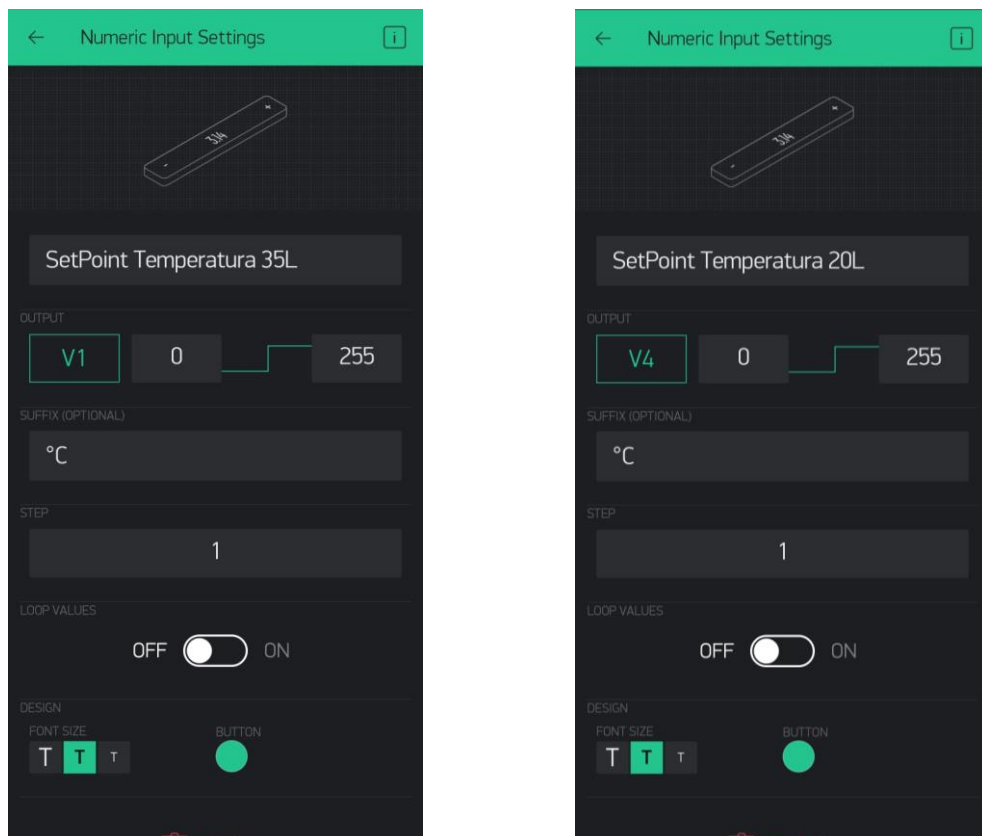


Fonte: Autor (2022)

6.4.2 Configuração do widget Numeric Input

Esse *widget* foi chamado de *SetPoint Temperatura 35L* e *SetPoint Temperatura 20L* e os pinos configurados foram V1 e o V4 para as painéis de 35 litros e de 20 litros respectivamente. Para a estética do aplicativo foi configurado um sufixo “°C” para aparecer na barra de digitação. Figura 27.

Figura 27 - Telas de configuração do *Numeric Input*

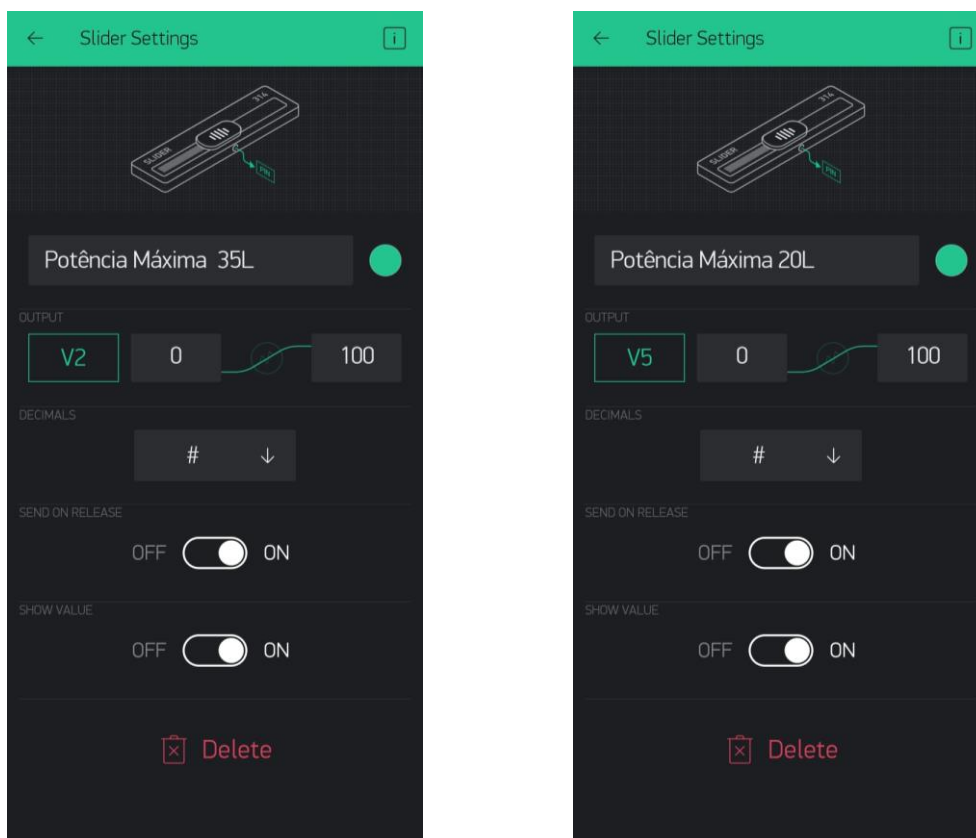


Fonte: Autor (2022)

6.4.3 Configuração do widget Slider

Os *Sliders* utilizados foram chamados de *Potência Máxima 35L* e *Potência Máxima 20L* e os pinos virtuais utilizados foram V2 e o V5 para as painéis de 35 litros e de 20 litros respectivamente. O valor mínimo e máximo configurados foram de 0 e 100, ou seja, esse *widget* irá enviar valores de 0 até 100 para o microcontrolador. Figura 28.

Figura 28 - Telas de configuração do *Slider*.



Fonte: Autor (2022)

6.4.4 Interação Blynk e Esp32

Para a comunicação do Blynk com o Esp32 via Wifi é preciso declarar bibliotecas específicas. Para esse projeto foram utilizadas as

- WiFi.h;
- WiFiClient.h;
- BlynkSimpleEsp32.h.

As bibliotecas “WiFi.h” e “WiFiClient.h” já são usuais para a utilização de conexão à rede com microcontroladores como o Arduino. A “BlynkSimpleEsp32.h” é uma biblioteca para comunicar o Blynk com o Esp32. Para realizar a conexão do Wifi e comunicar o Esp32 com o Blynk são necessários a declaração de três vetores char.

- char auth[] = "";
- char ssid[] = "";
- char pass[] = "".

Para a variável “char auth[]” é preciso atribuir o valor do *Token* dado pelo *Blynk* ao criar o projeto. Para “char ssid[]” é necessário inserir o nome da rede Wifi que o Esp32 irá se conectar. Já para o “char pass[]” é preciso atribuir o valor da senha da rede na qual o Esp32 será

conectado. Um ponto a ser observado é que o Esp32 e o dispositivo que o aplicativo Blynk está instalado não precisam estar conectados na mesma rede. Esses valores serão utilizados no comando `Blynk.begin(auth, ssid, pass)` feito no *setup* do código. Esse comando inicia um processo do Linux, conectando o microcontrolador aos servidores do Blynk.

Por fim o comando `Blynk.run()` deverá ser colocado no *loop* do código. Esse código iniciará uma função de bloqueio, ou seja, nada mais ocorrerá no *sketch*, até que todas as funções chamadas por ele sejam concluídas.

Para a interação com os *widgets* nesse projeto foram utilizadas duas funções da biblioteca do *Blynk*, são elas

- `BLYNK_WRITE()`;
- `Blynk.virtualWrite()`.

A função “`BLYNK_WRITE()`” é chamada toda vez que há uma atualização de algum pino virtual vindo do servidor, ou do aplicativo. Ela foi usada para receber os valores dos *widgets Numeric Input* e *Slider*. O trecho de código abaixo mostra como as funções foram utilizadas

```
//Recebimento de valores do Blynk, convertendo para float
BLYNK_WRITE(V1) {
  setTemp35L = param.asFloat();
}

BLYNK_WRITE(V2) {
  potMax35L = param.asFloat();
  map_potMax35L = map(potMax35L, 0, 100, limMin, limMax); //Transformando a base de 0 a 100 para 0 8333
  potMaxTime35L = 8333 - map_potMax35L - 10; //Achando o tempo desligado e compensando o delay de 10 uS
}

BLYNK_WRITE(V4) {
  setTemp20L = param.asFloat();
}

BLYNK_WRITE(V5) {
  potMax20L = param.asFloat();
  map_potMax20L = map(potMax20L, 0, 100, limMin, limMax); //Transformando a base de 0 a 100 para 0 8333
  potMaxTime20L = 8333 - map_potMax20L - 10; //Achando o tempo desligado e compensando o delay de 10 uS
}
```

Todos valores recebidos foram convertidos para o tipo de variável *float* através do comando “`param.asFloat()`”. Para os valores dos pinos virtuais referentes aos *Widgets Slider* além de serem convertidos para *float* foi necessário fazer uma conversão neles através da função

map. Para o usuário é mais conveniente escolher uma potência de 0% a 100%, mas para a lógica do código esse raciocínio não é válido, sendo necessária essa conversão.

A função “Blynk.virtualWrite()” foi utilizada para enviar os valores de temperatura lidos pelos termômetros para o aplicativo para serem exibidos nos *widgets Labeled Value*.

```
//Enviando as temperaturas para o Blynk
Blynk.virtualWrite(V0, sensors.getTempC(sensorTempPanela35L));
Blynk.virtualWrite(V3, sensors.getTempC(sensorTempPanela20L));
```

6.5 Circuito liga e desliga Sem Controle De Potência

O primeiro circuito projetado realiza um controle liga e desliga da resistência. Toda vez que a temperatura cai 0,3 °C do valor desejado, o microcontrolador ESP32 comuta o relé de estado sólido fazendo a resistência elétrica ligar e assim aquecendo o líquido no interior da panela.

6.6 Circuito liga e desliga com Controle De Potência

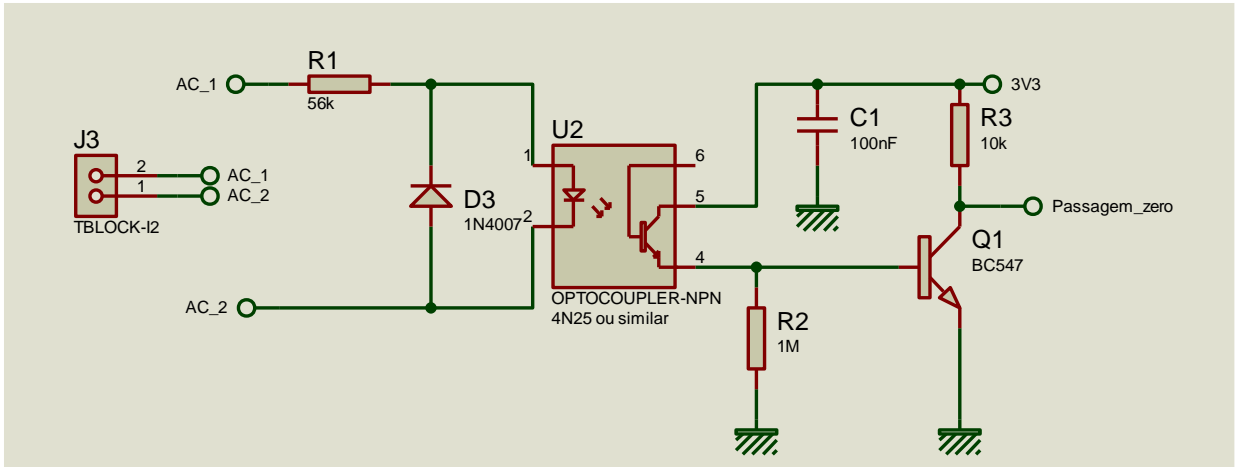
O segundo projeto de *hardware* também realiza um controle liga e desliga da resistência elétrica da panela, porém o circuito de acionamento através do relé foi retirado e em seu lugar foi inserido um circuito de controle de potência de corrente alternada. Este novo projeto de *hardware* de controle se divide em três partes.

6.6.1 Detector de passagem por zero

O primeiro circuito, alimentado pela rede elétrica, é um detector de passagem por zero. O resistor R1 causa uma queda de tensão para evitar a queima do diodo D3 e do foto acoplador U2 uma vez que, a rede é 220Vac.

Durante o semi-ciclo positivo o fotoacoplador U2 irá levar o transistor Q1 à condução. Devido a condução de Q1 o sinal “Passagem_zero” será zero. Durante o semi ciclo negativo o diodo D3 irá conduzir e o fotoacoplador não, logo, o transistor Q1 não irá conduzir fazendo com que o sinal “Passagem_zero” tenha um valor próximo a 3,3V, Figura 29.

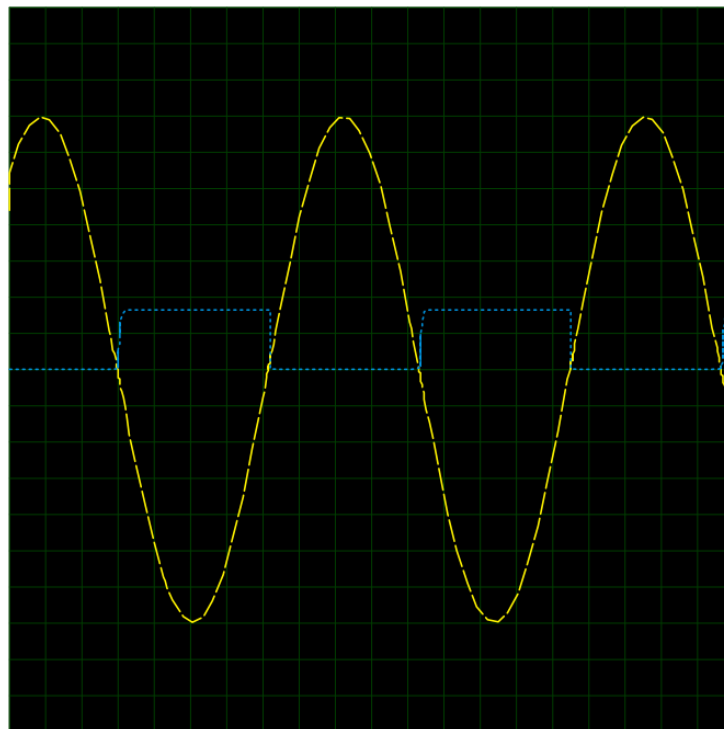
Figura 29 - Circuito detector de passagem por zero.



Fonte: Autor (2021)

A cada passagem da tensão da rede por zero, este circuito faz uma mudança de nível de tensão na entrada digital D34 do ESP32 disparando uma interrupção que irá fazer o controle das tensões através dos *Triacs*, Figura 32. A comparação entre o sinal da rede e o sinal de saída do detector de passagem por zero, Figura 30.

Figura 30 - Comparação entre o sinal da rede (Amarelo) com a saída do circuito detector de passagem por zero (Azul).

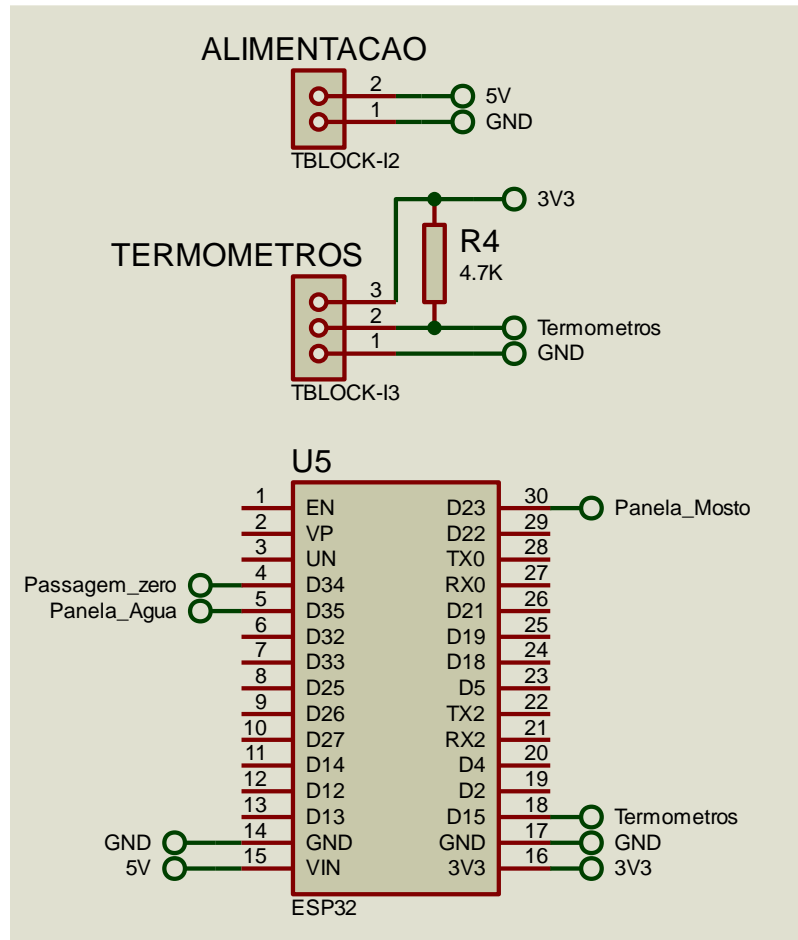


Fonte: Autor (2021)

6.6.2 Microcontrolador

A segunda parte do circuito é a parte responsável por fazer a comunicação com o aplicativo, receber as informações dos termômetros e realizar o controle da temperatura e das tensões a partir das entradas de controle, Figura 31.

Figura 31 - Circuito com microcontrolador.



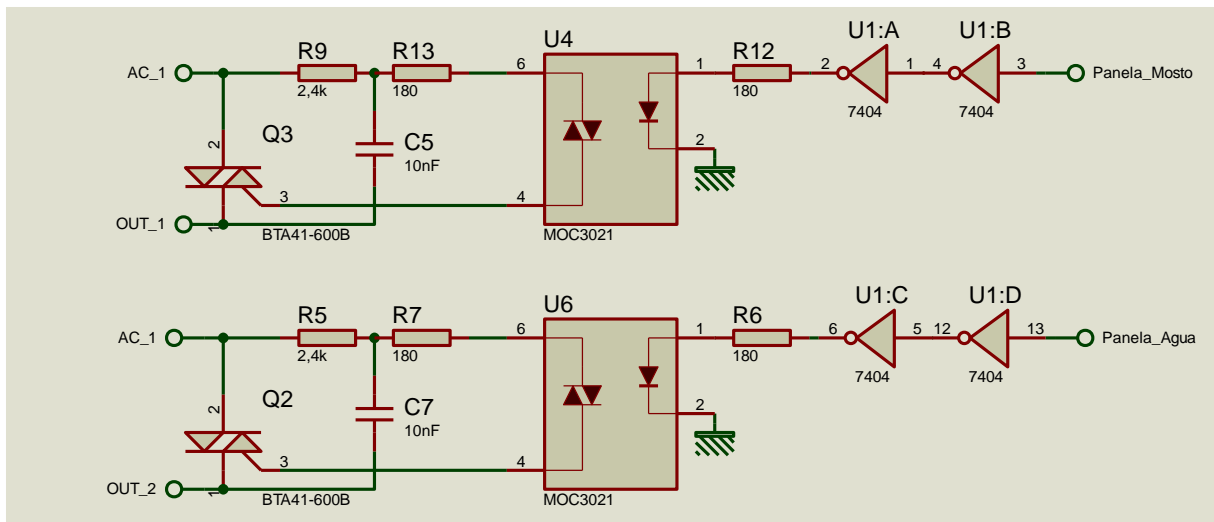
Fonte: Autor (2021)

O ESP32, de acordo com os sinais de controle do aplicativo e dos sensores de temperatura, envia o sinal com um atraso, retardando assim o disparo dos *Triacs* e fazendo com que parte da forma de onda da tensão de alimentação seja ceifada. Lembrando que, a cada passagem por zero os *Triacs* interrompem as alimentações das cargas. Com a senoide ceifada o valor RMS de tensão aplicado na carga diminui, conseqüentemente diminuindo a potência aplicada a carga.

6.6.3 Circuito de potência

O circuito de potência é constituído por um foto-diac MOC3021M que recebe o pulso do microcontrolador e assim dispara o *Triac* através de um divisor resistivo. Esse divisor resistivo é indicado pela própria fabricante *Onsemi*© em seu *datasheet*. A carga, neste caso a resistência elétrica, está conectada a um *Triac*. Na entrada do foto-diac foram colocados portas lógicas 7404, duas portas inversoras com o intuito de servirem como *buffer* digital protegendo as saídas do microcontrolador para que elas não sejam sobrecarregadas, Figura 32.

Figura 32 - Circuito com microcontrolador.



Fonte: Autor (2021)

6.6.4 Firmware do controle de potência

O controle de potência é feito após a interrupção causada pelo detector de passagem por zero chamar a função “passagem_zero()”. Primeiro essa função analisa quais resistências devem ser ligadas, através das variáveis *booleanas* “flagRes35L” e “flagRes20L”. Essas variáveis são alteradas na função “verifica_temp()”. Caso as variáveis sejam verdadeiras as painelas deverão ser ligadas. A função “passagem_zero()” faz a análise de quatro condições:

- As duas resistências desligadas;
- Somente a resistência da panela de 35 litros ligada;
- Somente a resistência da panela de 20 litros ligada;
- As duas painelas ligadas.

Uma maior ou menor potência aplicada na resistência determinará o tempo que a temperatura varia entre as etapas (rampas). O controle de potência é feito variando o momento em que o *Triac* conduzirá. Na rede brasileira é utilizada a frequência de 60 Hz, ou seja, o período de da senoide é de $1/60 = 16,66$ milissegundos. Cada semi-ciclo possui então 8,33 milissegundos ou 8333 microssegundos. O código atrasa a condução do *Triac* de acordo com a potência desejada. Quanto maior o atraso para a condução do mesmo menor será a tensão aplicada na resistência.

O tempo de atraso do disparo é feito pelo comando “*delayMicroseconds()*”. Um semi-ciclo completo está definido de 0 a 100%, esse valor recebido do aplicativo é convertido em uma nova escala, de 0 a 8333, através da função *map()*. Supondo que o valor enviado pelo aplicativo seja de 60% a função *map()* gera 4999,8, como esse é o tempo que a resistência ficará ligada é necessário subtrair de 8333, resultando em 3333,2. Devemos subtrair 10 microssegundos desse tempo, para o exemplo resultando em 3323,2, devido ao atraso da resposta do circuito foto-diac e *Triac*. O valor de 3323,2 parametrizará a função *delayMicroseconds()*.

```
BLYNK_WRITE(V2) {
  potMax35L = param.asFloat();
  map_potMax35L = map(potMax35L, 0, 100, limMin, limMax); //Transformando a base de 0 a 100 para 0 8333
  potMaxTime35L = 8333 - map_potMax35L - 10; //Achando o tempo desligado e compensando o delay de 10 uS
}

BLYNK_WRITE(V4) {
  setTemp20L = param.asFloat();
}

BLYNK_WRITE(V5) {
  potMax20L = param.asFloat();
  map_potMax20L = map(potMax20L, 0, 100, limMin, limMax); //Transformando a base de 0 a 100 para 0 8333
  potMaxTime20L = 8333 - map_potMax20L - 10; //Achando o tempo desligado e compensando o delay de 10 uS
}
```

A parte do código responsável por ligar e controlar a potência da panela de 35 litros:

```
if (flagRes35L == true && flagRes20L == false) { //Ligando somente a
panela de 35 litros
  delayMicroseconds(potMaxTime35L);
  digitalWrite(pinResPanela35L, HIGH);
  delayMicroseconds(10);
  digitalWrite(pinResPanela35L, LOW);
}
```

A parte do código responsável por ligar e controlar a potência da panela de 20 litros:

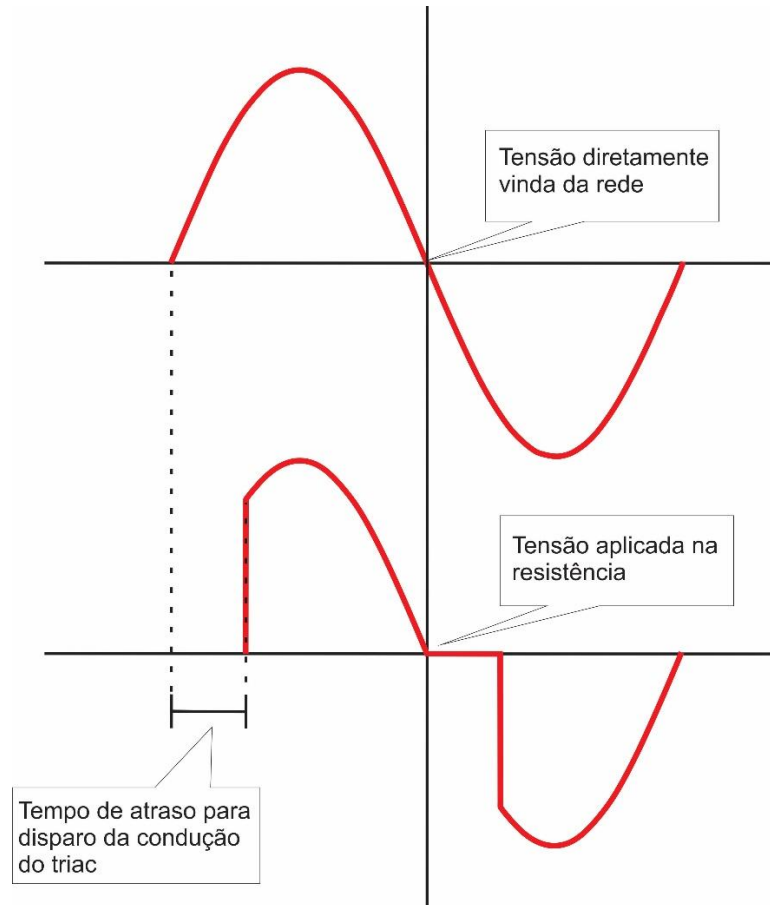

```

if (flagRes35L == false && flagRes20L == true) { //Ligando somente a
panela de 20 litros
  delayMicroseconds (potMaxTime20L);
  digitalWrite (pinResPanela20L, HIGH);
  delayMicroseconds (10);
  digitalWrite (pinResPanela20L, LOW);
}

```

A forma da senoide com um atraso, Figura 33.

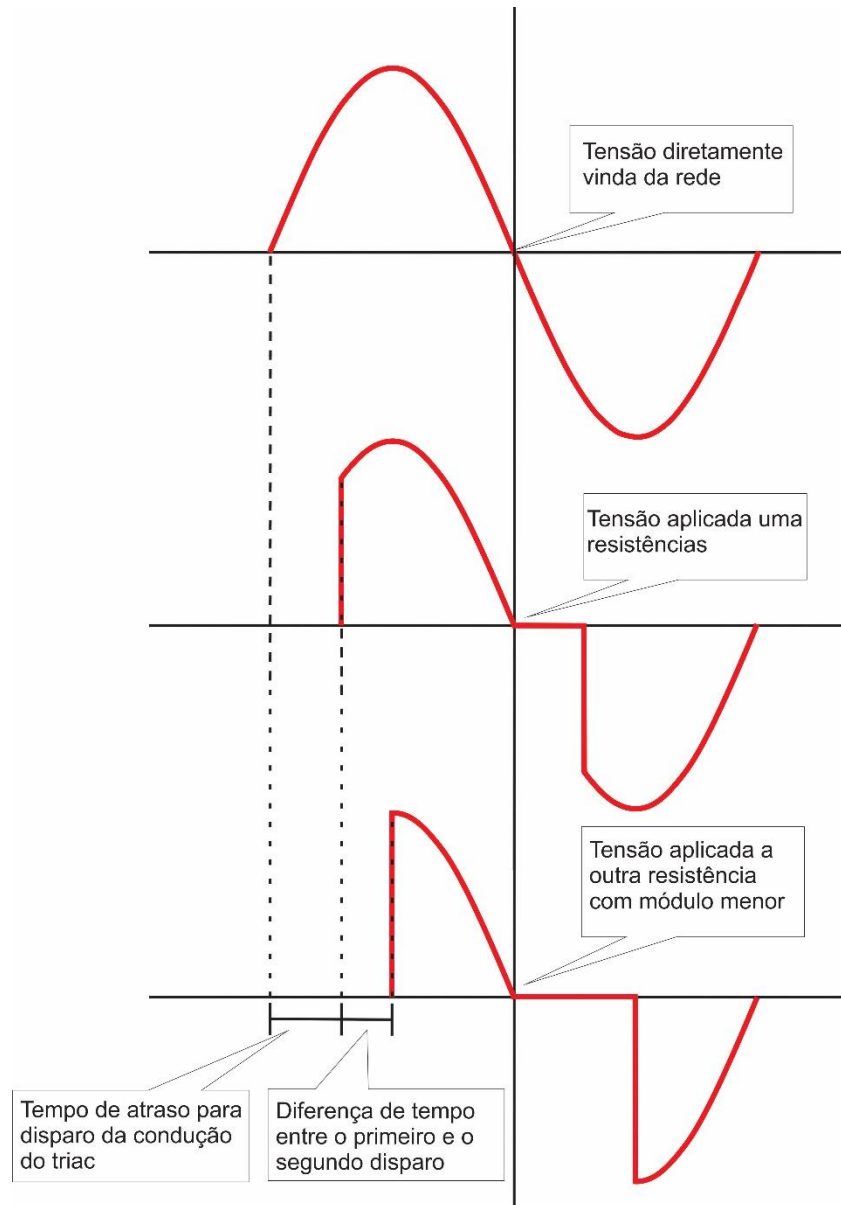
Figura 33 – Comparação de uma senoide pura com uma senoide que teve um atraso de disparo.



Fonte: Autor (2022)

Uma solução para realizar o controle de potência das duas resistências simultaneamente, caso as resistências tenham um tempo de disparo diferente. Após disparado o primeiro *Triac*, Figura 33, subtraindo o tempo de disparo do segundo *Triac* do tempo do primeiro, já levando em consideração o atraso de 10 microssegundos, com o tempo determinado, ou seja, a diferença entre os tempos de disparo vai ser usada para disparar o segundo *Triac*. Com isso garantimos que os *Triacs* sejam disparados no mesmo semi-ciclo, Figura 34.

Figura 34 – Comparação entre uma senoide pura e outras duas senoides com tempos de disparo diferentes



A parte do código responsável pela verificação de qual disparo deve ocorrer primeiro e o acionamento dos dois *Triacs* durante o mesmo semi-ciclo, está descrita abaixo:

```

//Caso as duas panelas precisam ser utilizadas ao mesmo tempo
if (flagRes35L == true && flagRes20L == true) {
//Caso o potência da panela de 35 litros seja menor
if (potMaxTime35L > potMaxTime20L) {
    delayMicroseconds (potMaxTime20L); //Deixo a resistência desli-
gada por potMaxTime20L microssegundos
    digitalWrite (pinResPanela20L, HIGH);
    delayMicroseconds (10); // tempo necessário apra o circuito do
triac perceber o pulso
    digitalWrite (pinResPanela20L, LOW);
    delayMicroseconds (potMaxTime35L - potMaxTime20L + 10);
    digitalWrite (pinResPanela35L, HIGH);
    delayMicroseconds (10);
    digitalWrite (pinResPanela35L, LOW);
}
{//Caso a potência da panela de 20 litros seja menor
if (potMaxTime20L > potMaxTime35L)
    delayMicroseconds (potMaxTime35L);
    digitalWrite (pinResPanela35L, HIGH);
    delayMicroseconds (10);
    digitalWrite (pinResPanela35L, LOW);
    delayMicroseconds (potMaxTime20L - potMaxTime35L + 10);
    digitalWrite (pinResPanela20L, HIGH);
    delayMicroseconds (10);
    digitalWrite (pinResPanela20L, LOW);
}
//Caso a potência das panelas sejam iguais
if (potMaxTime20L == potMaxTime35L) {
    delayMicroseconds (potMaxTime35L);
    digitalWrite (pinResPanela35L, HIGH);
    digitalWrite (pinResPanela20L, HIGH);
    delayMicroseconds (10);
    digitalWrite (pinResPanela35L, LOW);
    digitalWrite (pinResPanela20L, LOW);
}
}
}

```

7 METODOLOGIA

7.1 Configurações do equipamento.

O primeiro passo foi caracterizar o equipamento no *software Brewfather*, para isso foram parametrizados os dados necessários para a realização dos testes.

As características do equipamento, no *software*, foi a mesma para os três testes. O equipamento foi nomeado como *Single Vessel TCC* e estão descritas na Tabela 3 e representadas na Figura 35.

A formulação da receita, passo seguinte, foi a partir das características descritas no guia adotado. A inserção dos dados de densidade e volume pré e pós fervura, Tabela x2, permite o acompanhamento detalhado da evolução do lote pretendido.

Tabela 3 - Dados configurados no *Brewfather*

Dados inseridos no <i>Software Brewfather</i>	
Tempo de fervura	60 minutos
Volume do lote	25 litros
Evaporação	3 litros por hora
Perda trub /chiller	2 litros
Espaço morto na panela de mostura	0 litros
Perda panela de mostura	0 litros
Espaço morto na panela de lavagem	0 litros
Perda fermentador 5l	1 litro
Água no fermentador	0 litros
Eficiencia do equipamento	65 %

Fonte: Autor (2021)

Figura 35 - Tela de edição de perfil de equipamento do *software Brewfather*.

? ★ ✕

Editar Perfil de Equipamento

Nome Tempo de Fervura minutos

Single Vessel - TCC 60

Descrição

Volumes

Volume Desejado do Lote

Fermentador

Volume do Lote (Fermentador) L

25

Calc. vol. fervura

Volume Pré Fervura* L

31,13

Evaporação (9.6%) L/hora

3

Perda Trub/Chiller L

2

Espaço Morto na Painela de Mostura L

0

Perda na Painela de Mostura L

0

Espaço Morto na Painela de Lavagem L

0

Perda no Fermentador L

1

Água no Fermentador L

0

Volume Pós Fervura: 28.13 L
 Volume de Envase: 24 L
 *Volumes de pré e pós-fervura são **quente**
 (incluindo 4% de expansão)

Eficiência

Eficiência do Equipamento %

65

Eficiência da Mostura %

70,2

Calcular eficiência da mostura

Fonte: Autor (2021)

7.2 Receita: American Lager

Para todos os testes realizados foi definida uma receita, com a mesma quantidade de ingredientes, água e mesmas rampas de temperatura. O estilo escolhido foi o *1B-American Lager* (BJCP 2015).

De acordo com o guia BJCP 2015 uma *American Lager* tem as seguintes impressões gerais “Uma *Lager* muito clara, altamente carbonatada, de corpo leve e bem atenuada, com um perfil de sabor muito neutro e de baixo amargor.”

Seus ingredientes característicos são “cevada de duas ou seis fileiras, com uma porcentagem elevada de arroz ou milho como adjuntos.”

As estatísticas vitais são parâmetros estabelecidos pelos guias de cervejas para que se possa manter um padrão e distinção melhor entre os estilos. Os parâmetros estabelecidos para o estilo 1B-*American Lager* (BJCP 2015) estão definidos na Tabela 4.

Tabela 4 - Estatísticas vitais do guia BJCP

Faixas das características	
OG	1,040 - 1,050 kg/m ³
FG	1,004 - 1,010 kg/m ³
COR	4 - 15,7 EBC
AMARGOR	8 – 18 IBU
ABV	4,2-5,3 %

Fonte: BJCP (2015)

A partir desses dados e do equipamento já pré configurado foi definida a receita através do *software Brewfather*.

7.2.1 Maltes e adjuntos:

Foi definida para a receita o uso de 4 quilogramas de malte de cevada *Pilsner* da marca *Viking Malt* e o uso de 1 quilograma de flocos de milho. Foi definido 8 gramas de lúpulo do tipo Columbus da marca *Yakima Chiefs*. A levedura utilizada foi a *German lager Levteck*. Foi definida somente uma rampa de temperatura de 65 °C por 60 minutos e a fervura se´ra de 60 minutos com a inserção do lúpulo logo no começo da fervura. A Figura 36 mostra a tela de configuração da receita do *software Brewfather*. Os valores de OG, FG, Cor, Amargor e ABV são calculados automaticamente relacionando as configurações do equipamento com as quantidades de ingredientes da receita.

Figura 36 - Tela de configuração da receita.

Lager - Trabalho de Conclusão de Curso — 4.2%

Nome: Lager - Trabalho de Conclusão de Curso
 Autor: Estevan Pithan
 Tipo: Somente Gr...

Equipamento: Single...
 01B American Lager

Volume do Lote: 25 L
 Tempo de Fervura: 60 min
 Volume Pré Fervura: 31.13 L
 Eficiência do Equipamento: 65 %
 Eficiência da Mostura: 70.2 %
 Temperatura da Fervura: 97.1 °C

ABV	4.2%	5.3%	
OG	1.041	1.050	
FG	1.009	1.009	
EBC	3.9	6.1	7.9
IBU	12	18	
BU/GU	0.18	0.29	0.40

ADICIONAR IMAGEM

Fermentáveis (5 kg) OG + ADICIONAR

- 4 kg Pilsner Viking Malt (Grão 4 EBC) 80%
- 1 kg Corn Yellow, Flaked Briess (Grão 2.6 EBC) 20%

Densidade Pré Fervura: 1.037 / 9.3 °P
 Densidade Original: 1.041 / 10.3 °P
 Cor: 6.1 EBC

Lúpulos (8 g) IBU + ADICIONAR

- 8 g Columbus/Tomahawk/Zeus (CTZ) 17.2% (0.32 g/l, 12 IBU) 60 min Fervura

Hopstand: 5 min @ 80 °C
 Total de IBU (Tinseth): 12
 BU/GU: 0.29
 RBR: 0.29

Diversos + ADICIONAR Levedura CALC + ADICIONAR

- 1 pct Levteck TeckBrew 84 77% German Lager II
- 2.1 L starter (218 g DME / 266 g LME)
- 392 bilhões de células de levedura
- 1.56 milhões de células / ml / °P

Fonte: Autor (2021)

Considerando as configurações iniciais do equipamento a previsão do *Brewfather* para o mosto após pronto então mostrados na Tabela 5 abaixo

Tabela 5 - Valores previstos pelo *software* para o resultado da receita.

Faixas das características	
OG	1,041 kg/m ³
FG	1,009 kg/m ³
COR	6,1 EBC
AMARGOR	12 IBU
ABV	4,2 %

Fonte: Autor (2021)

7.3 Etapas de funcionamento do equipamento

Os testes executados neste trabalho tiveram em comum os procedimentos básicos descritos a seguir:

A panela de mostura foi enchida com 18 litros, na sequência a temperatura foi elevada para 30 °C. Para termos parâmetros de comparação, os testes devem ter a mesma temperatura

inicial. Em seguida a temperatura de *setpoint* foi alterada para 70 °C. Feito isso o cronômetro foi disparado com a finalidade de marcação do tempo da Brassagem.

Após a temperatura da água atingir 70 °C, os maltes foram colocados dentro do cesto e misturados com a água. Novamente foi ajustado o *setpoint* da temperatura para 65 °C.

Figura 37 - Maltes colocados no cesto para o começo da mosturação.

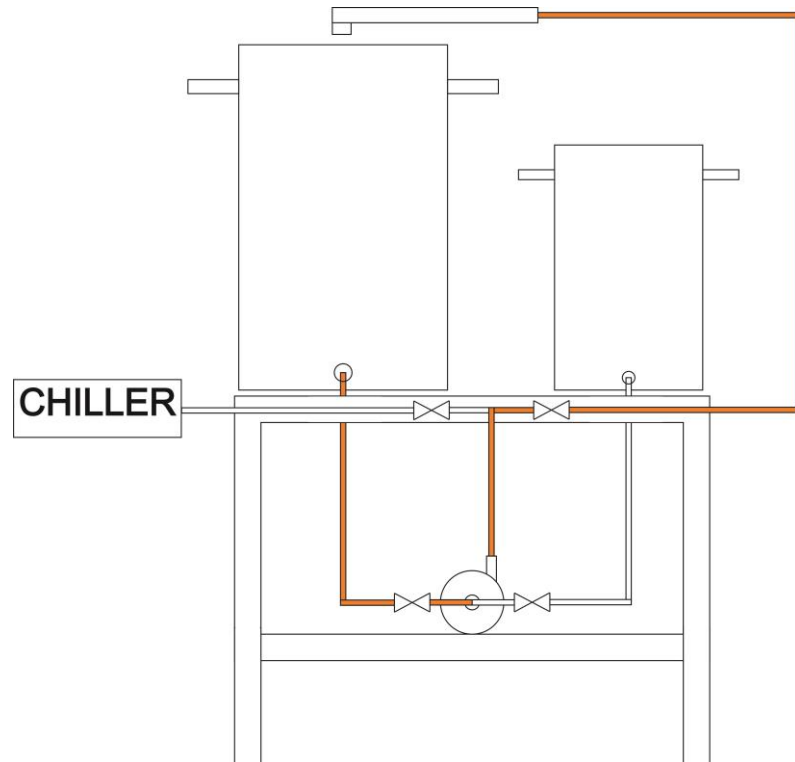


Fonte: Autor (2021)

A temperatura inicial da água é calculada pelo *software Brewfather*. O motivo do primeiro *setpoint* de temperatura ser mais alto que o segundo se deve ao fato dos maltes estarem em temperatura ambiente fazendo com que a temperatura da água diminua quando estes mesmos maltes são arriados.

A temperatura é mantida em 65 °C por 60 minutos, neste intervalo de tempo ocorre a recirculação do mosto, visando manter a temperatura uniforme para a conversão dos amidos em açúcares mais eficiente. A figura abaixo ilustra o fluxo do líquido no sistema.

Figura 38 - Sistema de recirculação durante a mosturação.

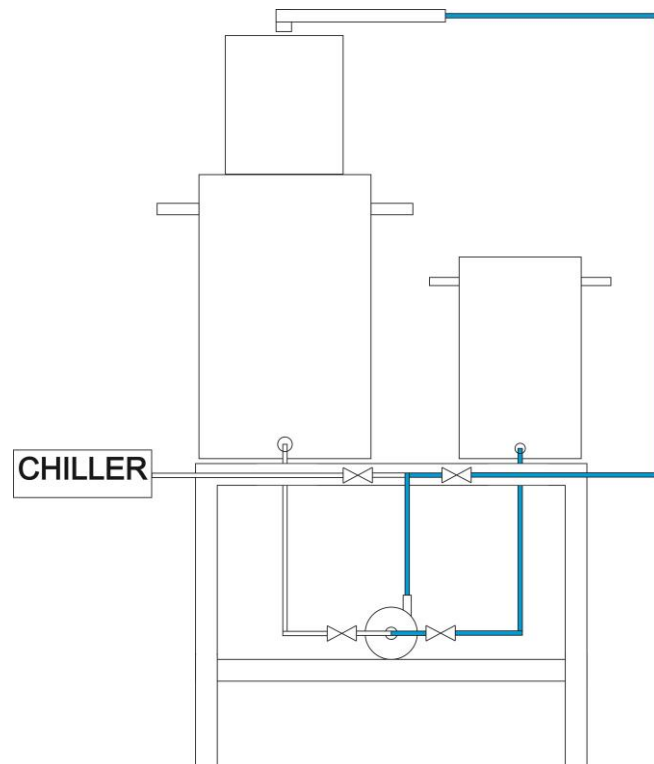


Fonte: Autor (2021)

Após os 60 minutos iniciais o *setpoint* é alterado para 78 °C. Quando a temperatura atinge os 78 °C, a etapa de *Mash Out* é iniciada e sua duração é de 10 minutos. Durante essa etapa a cama de grãos não foi mexida, para que a clarificação seja feita.

Finalizada a etapa de *Mash Out*, o cesto é erguido fazendo a separação do mosto e do bagaço de malte. A próxima etapa é a lavagem do bagaço de malte que começa com a água previamente aquecida à 78°C. Ao todo foram utilizados 18 litros de água. A Figura 39 ilustra o processo de lavagem.

Figura 39 - Sistema de lavagem do bagaço de malte.

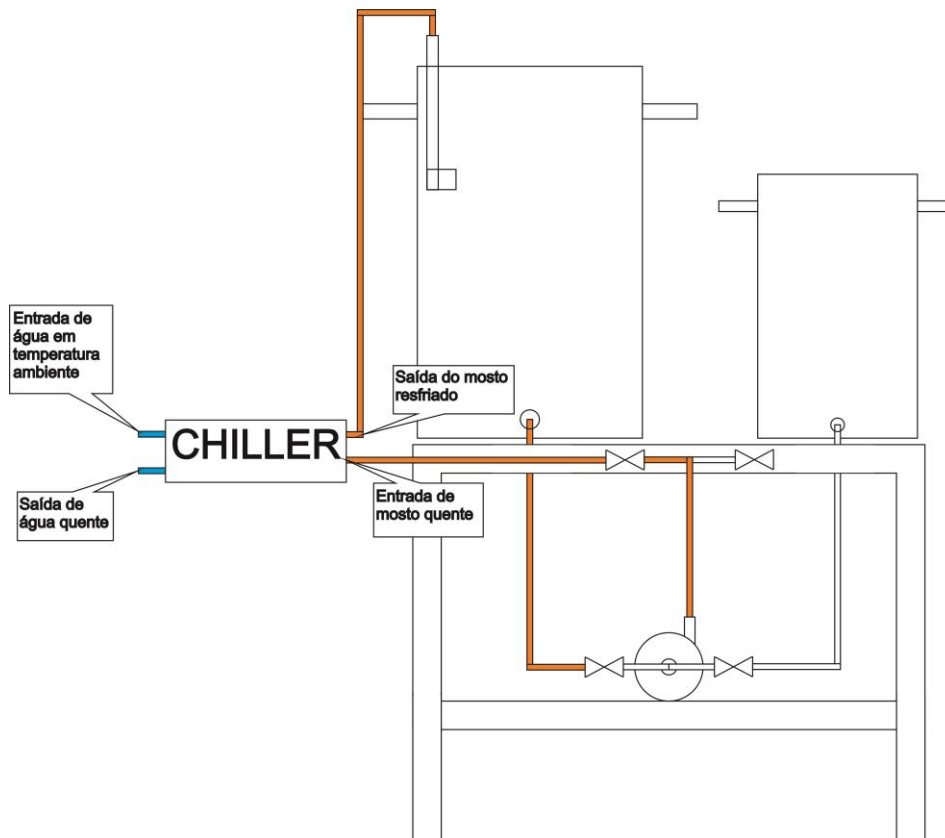


Fonte: Autor (2021)

Uma vez concluída a lavagem dos grãos, o *setpoint* da temperatura foi ajustado em 100 °C. Quando o mosto entrou em fervura, foi adicionado lúpulo e mantida a fervura por 60 minutos. Ao fim da fervura o cronômetro marcava o tempo total da Brassagem.

O tempo da etapa de resfriamento do mosto e de decantação do *trub* não foram contabilizados para o projeto, pois a velocidade de resfriamento depende da temperatura da água que é utilizada no *chiller* de placas e não foi possível manter um padrão em cada teste. A Figura 40 ilustra o funcionamento do sistema durante a etapa de resfriamento e decantação do *trub*.

Figura 40 - Sistema de resfriamento do mosto junto com a decantação do mesmo.



Fonte: Autor (2021)

Uma vez que o mosto esteja em uma temperatura menor do que 30 °C o mesmo foi transferido para o fermentador e em seguida encaminhado para o freezer para que se inicie a fermentação.

7.4 Testes realizados

Foram feitos ao todo três testes diferentes. Abaixo está uma descrição do que foi feito em cada um.

7.4.1 Do primeiro teste

O primeiro teste, foi realizado com a tensão nominal da rede, 220 volts, resultando em uma potência total de 3500 *Watts* gerados pela resistência elétrica, aplicada diretamente ao mosto. O sistema de liga e desliga automático, em função da temperatura, mas sem controle de potência.

7.4.2 *Do segundo teste*

O segundo teste consistiu na implementação de um circuito de controle de potência de corrente alternada. A implementação deste circuito tem como objetivo diminuir a tensão aplicada à resistência, sendo assim, reduzindo a potência gerada pela mesma em determinados momentos da Brassagem, nesse teste especificamente a tensão da rede foi reduzida em 50 % durante a etapa de fervura somente, uma vez que, no primeiro teste, ocorreram perdas de líquido durante essa etapa. As perdas de líquido foram causadas pela grande intensidade da fervura, ou seja, muitas borbulhas fazendo com que o líquido caíssem para fora da panela.

7.4.3 *Do terceiro teste*

O terceiro teste consistiu em avaliar o impacto no tempo total da Brassagem e na eficiência caso no início do projeto fosse utilizado uma resistência com menor potência, evitando os problemas de transbordamento causados no primeiro teste. Durante toda essa Brassagem foi aplicada somente 50 % da tensão da rede, resultando em uma potência quatro vezes menor do que a potência nominal.

8 RESULTADOS E DISCUSSÃO

8.1 Resultado do primeiro teste

Durante a fervura do primeiro teste ocorreram transbordamentos devido a alta intensidade de borbulhamento, resultando assim em um volume final menor. Os dados coletados e inseridos no *Brewfather* foram os especificados na Tabela 6.

Tabela 6 - Valores mensurados no primeiro teste.

Valores mensurados durante a Mostura do primeiro teste	
Volume da fervura	32 litros
Densidade Pré Fervura	1,037 kg/m ³
Densidade Pós Fervura	1,042 kg/m ³
Volume Pós Fervura	22 litros
Densidade Original	1,042 kg/m ³
Volume no Fermentador	20 litros

Fonte: Autor (2021)

Após os dados da tabela acima serem inseridos obteve-se os seguintes resultados do primeiro teste.

Tabela 7 - Eficiências do primeiro teste.

Resultados de eficiência de Mostura e do equipamento do primeiro teste	
Eficiência de Mostura	72,16 %
Eficiência do equipamento	52,98 %

Fonte: Autor (2021)

O tempo total da Brassagem até o fim da fervura foi de 159 minutos. A Tabela 8 mostra o tempo em cada etapa da Brassagem.

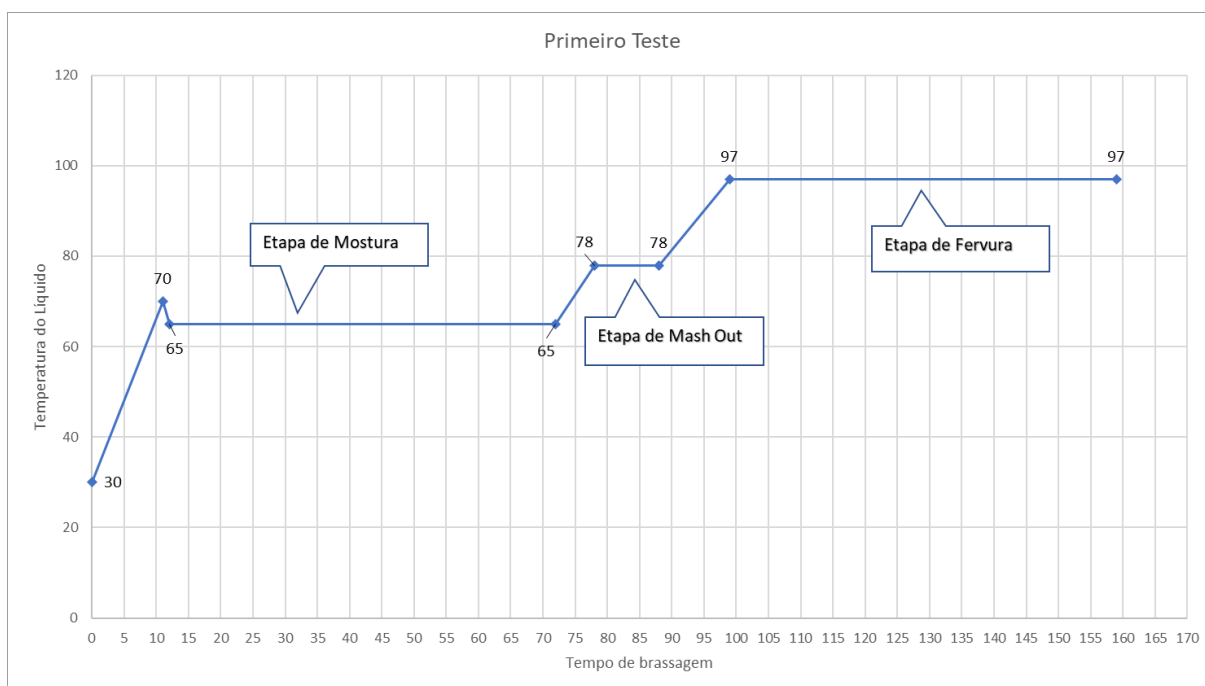
Tabela 8 - Valores de tempo e temperaturas do primeiro teste

Relação de tempo e temperatura do primeiro teste	
--	--

Etapa da Brassagem	Tempo de Brassagem	Temperatura
Início da Brassagem	0 minutos	30 °C
Inserção dos grãos	11 minutos	70 °C
Início da Mostura	12 minutos	65 °C
Fim da Mostura	72 minutos	65 °C
Início do <i>Mash Out</i>	78 minutos	78 °C
Fim do <i>Mash Out</i>	88 minutos	78 °C
Início da Fervura	99 minutos	97 °C
Fim da Fervura	159 minutos	97 °C

Fonte: Autor (2021)

Figura 41 – Grafico da variação da temperatura durante a Brassagem do primeiro teste



Fonte: Autor (2021)

8.2 Resultados do segundo teste

No segundo teste foi possível, devido ao circuito novo, fazer um controle de potência durante a fervura, aplicando 50 % da tensão nominal da rede e como esperado não ocorreram derramamentos, sendo toda a diferença de volume entre o pré e pós fervura.

Tabela 9 - Valores mensurados no segundo teste.

Valores mensurados durante a Mostura do segundo teste	
Volume da fervura	32 litros
Densidade Pré Fervura	1,038 kg/m ³
Densidade Pós Fervura	1,041 kg/m ³
Volume Pós Fervura	27,5 litros
Densidade Original	1,041 kg/m ³
Volume no Fermentador	25,5 litros

Fonte: Autor (2021)

Após os dados da tabela acima serem inseridos obteve-se os seguintes resultados do segundo teste.

Tabela 10 - Eficiências do segundo teste.

Resultados de eficiência de Mostura e do equipamento do segundo teste	
Eficiência de Mostura	74,11 %
Eficiência do equipamento	65,94 %

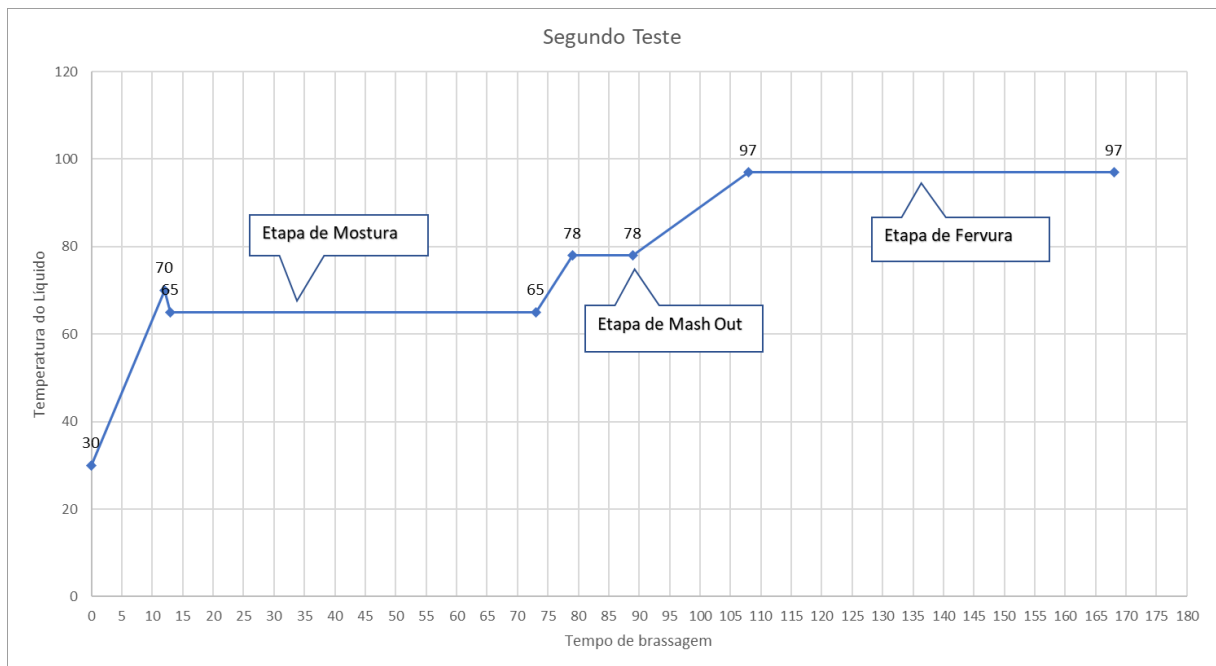
Fonte: Autor (2021)

O tempo total da Brassagem até o fim da fervura foi de 168 minutos. A Tabela 11 mostra o tempo em cada etapa da Brassagem.

Tabela 11 - Valores de tempo e temperaturas do segundo teste.

Relação de tempo e temperatura do primeiro teste		
Etapa da Brassagem	Tempo de Brassagem	Temperatura
Início da Brassagem	0 minutos	30 °C
Inserção dos grãos	12 minutos	70 °C
Início da Mostura	13 minutos	65 °C
Fim da Mostura	73 minutos	65 °C
Início do <i>Mash Out</i>	79 minutos	78 °C
Fim do <i>Mash Out</i>	89 minutos	78 °C
Início da Fervura	108 minutos	97 °C
Fim da Fervura	168 minutos	97 °C

Fonte: Autor (2021)

Figura 42 – Gráfico da variação da temperatura durante a Brassagem do segundo teste

Fonte: Autor (2021)

8.3 Resultados do terceiro teste

O terceiro teste foi realizado para avaliar a possibilidade de utilizar uma resistência elétrica de menor potência, a tensão aplicada à resistência foi 50 % da tensão nominal da rede, durante toda a Brassagem. Valores mensurados estão na Tabela 12.

Tabela 12 - Valores mensurados no terceiro teste.

Valores mensurados durante a Mostura do terceiro teste	
Volume da Fervura	32 litros
Densidade Pré Fervura	1,041 kg/m ³
Densidade Pós Fervura	1,044 kg/m ³
Volume Pós Fervura	27,5 litros
Densidade Original	1,044 kg/m ³
Volume no Fermentador	25,5 litros

Fonte: Autor (2021)

Após os dados da tabela acima serem inseridos obteve-se os seguintes resultados do terceiro teste.

Tabela 13 - Eficiências do terceiro teste.

Resultados de eficiência de Mostura e do equipamento do terceiro teste	
Eficiência de Mostura	81,91 %
Eficiência do equipamento	70,76%

Fonte: Autor (2021)

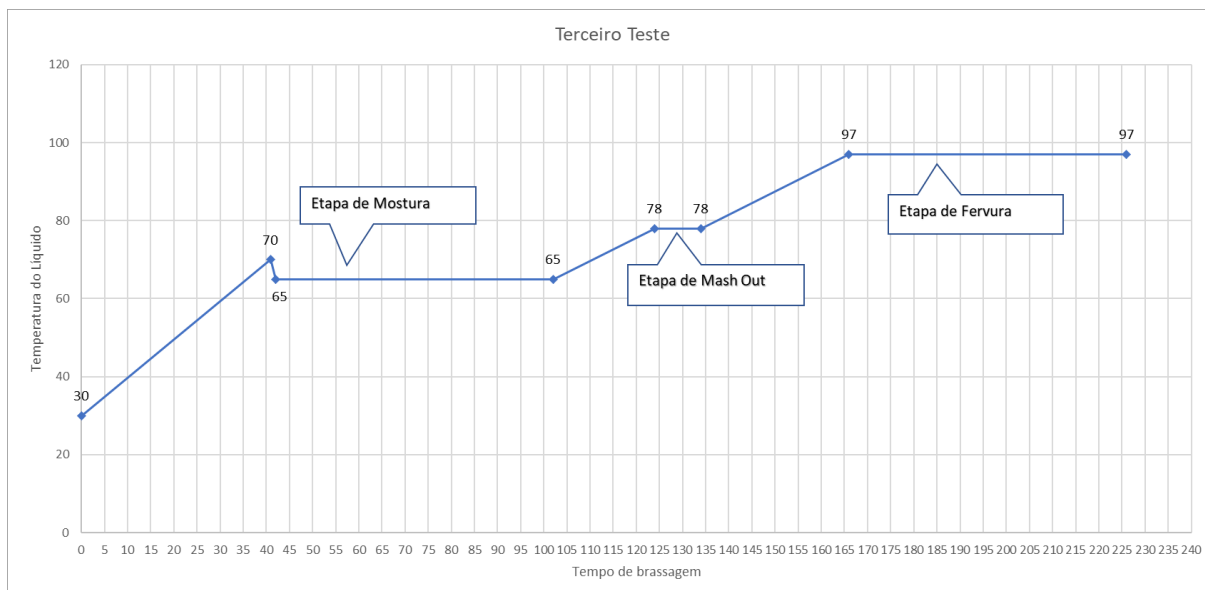
O tempo total da Brassagem até o fim da fervura foi de 226 minutos. A Tabela 14 mostra o tempo em cada etapa da Brassagem.

Tabela 14 - Valores de tempo e temperaturas do terceiro teste.

Relação de tempo e temperatura do terceiro teste		
Etapa da Brassagem	Tempo de Brassagem	Temperatura
Início da Brassagem	0 minutos	30 °C
Inserção dos grãos	41 minutos	70 °C
Início da Mostura	42 minutos	65 °C
Fim da Mostura	102 minutos	65 °C
Início do <i>Mash Out</i>	124 minutos	78 °C
Fim do <i>Mash Out</i>	134 minutos	78 °C
Início da Fervura	166 minutos	97 °C
Fim da Fervura	226 minutos	97 °C

Fonte: Autor (2021)

Figura 43 – Gráfico da variação da temperatura durante a Brassagem do Terceiro teste.



Fonte: Autor (2021)

8.4 Comparação dos resultados

A tabela abaixo compara diretamente os resultados obtidos nos três testes. A partir da tabela foram gerados os gráficos da Figura 44, Figura 45 e Figura 46.

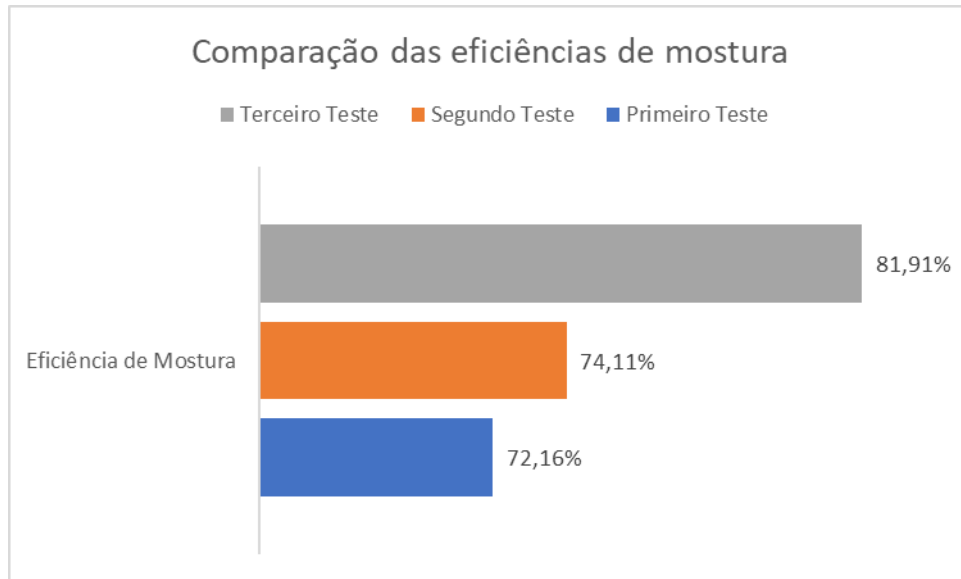
Tabela 15 - Comparação dos resultados dos testes.

Resultados de eficiência de Mostura e do equipamento do terceiro teste			
Testes realizados	Primeiro	Segundo	Terceiro
Eficiência de Mostura	72,16%	74,11%	81,91%
Eficiência do equipamento	52,98%	67,55%	70,76%
Tempo de Brassagem	159 minutos	168 minutos	226 minutos

Fonte: Autor (2021)

Pela Figura 44 é possível observar que os dois primeiros testes apresentaram eficiências próximas, com uma pequena diferença entre elas. Em contra partida o terceiro teste apresentou uma diferença adicional maior, conseguindo ter mais açúcares no começo da fervura.

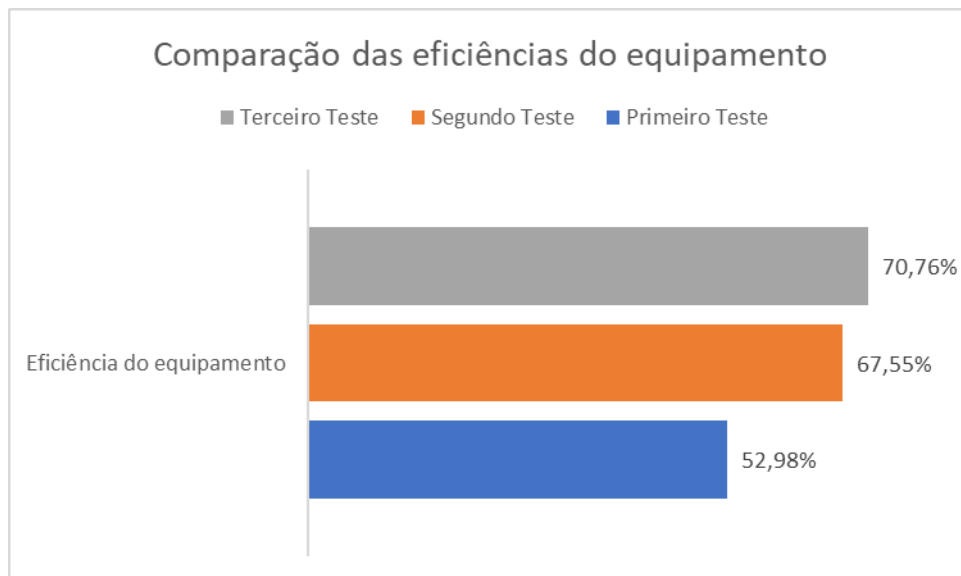
Figura 44 – Gráfico da comparação das eficiências de mostura



Fonte: Autor (2021)

Já pela Figura 45 é possível notar que a perda de líquido no primeiro teste teve um impacto significativo na eficiência do equipamento e que a implementação de um controle de potência teve um bom resultado. A melhora da eficiência do equipamento do primeiro para o segundo teste foi de 12,96 %.

Figura 45 – Gráfico da comparação das eficiências do equipamento durante cada teste



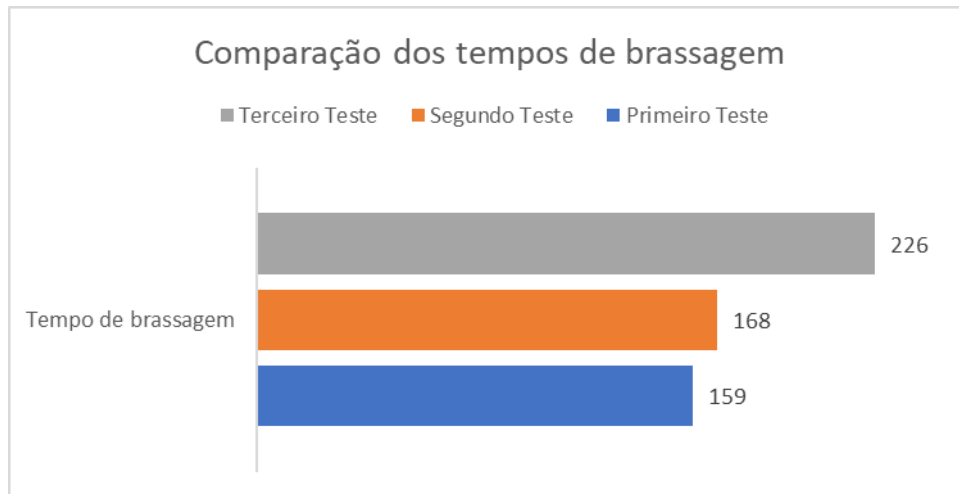
Fonte: Autor (2021)

Considerando a Figura 46 os tempos de Brassagem dos dois primeiros testes foram próximos, sendo a diferença de 9 minutos. Observou-se que ao reduzir a tensão da resistência

a fervura parou por alguns minutos, sendo isso, um dos motivos pela diferença de tempo entre os dois testes.

O terceiro teste, como esperado teve um tempo de Brassagem relativamente maior.

Figura 46 – Gráfico da comparação dos tempos de Brassagem de cada teste.



Fonte: Autor (2021)

9 CONCLUSÃO

No teste final foi comprovado ver um aumento principalmente na eficiência de mostura, acarretando também em um aumento da eficiência do equipamento em contrapartida o tempo total de Brassagem foi de 69 minutos a mais em relação ao segundo teste.

O aumento da eficiência de mostura, relacionado a quantidade de açúcares extraídos dos grãos, no terceiro teste foi um ganho adicional. A potência mais elevada na etapa de mostura durante os dois primeiros testes pode estar desnaturando as enzimas fazendo com que a conversão dos amidos em açúcares não esteja ocorrendo de forma completa.

Para futuros trabalhos será testado o procedimento de redução da potência durante a etapa de sacarificação e de fervura visando uma eficiência de mostura e de Brassagem próxima a do teste três. A potência aplicada no período das rampas de temperatura será mantida em 100 %, a mesma potência que foi utilizada no segundo teste, com isso diminuindo o tempo total de Brassagem.

Com os resultados obtidos vai ser possível automatizar todo o processo, cada etapa será inserida previamente no aplicativo não necessitando da intervenção do usuário para a alteração de potência e temperatura durante a Brassagem. A possibilidade de automatizar a inserção dos grãos assim como o levantamento do cesto para a lavagem do bagaço também será avaliada.

REFERÊNCIAS

MORADO, Ronaldo. Larousse da Cerveja. São Paulo: Larousse do Brasil, 2009.

SEBRAE - Serviço Brasileiro de Apoio às Micro e Pequenas Empresas – Cervejas Artesanais. Disponível em: <https://www.sebraeinteligenciasetorial.com.br/produtos/relatorios-de-inteligencia/cervejas-artesanais/55c4ad3614d0c01d007ffeae#download>. Acesso em 19 dez 2021

SEBRAE. Microcervejarias ganham espaço no mercado nacional. 2017. Disponível em: <<https://www.sebrae.com.br/sites/PortalSebrae/artigos/microcervejarias-ganham-espaco-no-mercado-nacional,fbe9be300704e410VgnVCM1000003b74010aRCRD>>. Acesso em 19 dez 2021

PALMER John. **How to Brew**. 3 Ed. Boulder, CO: Brewers Publications, 2006.

KROTTENTHALER, M.; BACK, W.; ZARKNOW.; **Handbook of Brewing: Processes, Technology, Markets**. Weinheim: WILEY-VCH Verlag GmbH & Co., 2009.

KUNZE, W.; **Technology Brewing and Malting**. 2. Ed, Berlim: VLB Berlin, 724p., 1996

BRIGGS, D. E.; BOULTON, C. A.; BROOKES, P. A.; STEVENS, R. **Malts, adjuncts and supplementary enzymes**. In: BRIGGS, D. E.; BOULTON, C. A.; BROOKES, P. A.; STEVENS, R. *Brewing: Science and Practice*. Cambridge: Woodhead Publishing, 2004b. cap. 2, p. 11-51. <https://doi.org/10.1201/9780203024195>

SMITH, Brad. **Brewhouse Efficiency for All Grain Beer Brewing**. 2008. Disponível em: <<http://beersmith.com/blog/2008/10/26/brewhouse-efficiency-for-all-grain-beer-brewing/>>. Acesso em: Dezembro. 2021.

AWADALLAK, Jamal Abd. Como é o perfil do cervejeiro caseiro Brasileiro - Resultados 2019. Youtube, 17 de nov. de 2020. Disponível em: <https://www.youtube.com/watch?v=CsZ_EjauZJI&ab_channel=BeerSchool> Acesso em 19 dez 2021

LARA, P. C. **Brassagem de cerveja artesanal: tudo o que você precisa saber**. Disponível em: <<https://www.hominilupulo.com.br/brassagem-avancada/>>. Acesso em: 18 dez. 2021.

Blynk. Disponível em: <<https://docs.blynk.cc/>>. Acesso em: 8 dez. 2021

ROSENTHAL, P. R. **O que é a maltagem e como funciona seu processo?** Disponível em: <<https://www.hominilupulo.com.br/maltagem/>>. Acesso em: 12 dez. 2021.

AHMED, Ashfaq. *Eletrônica de Potência*. São Paulo: Prentice Hall, 2000.

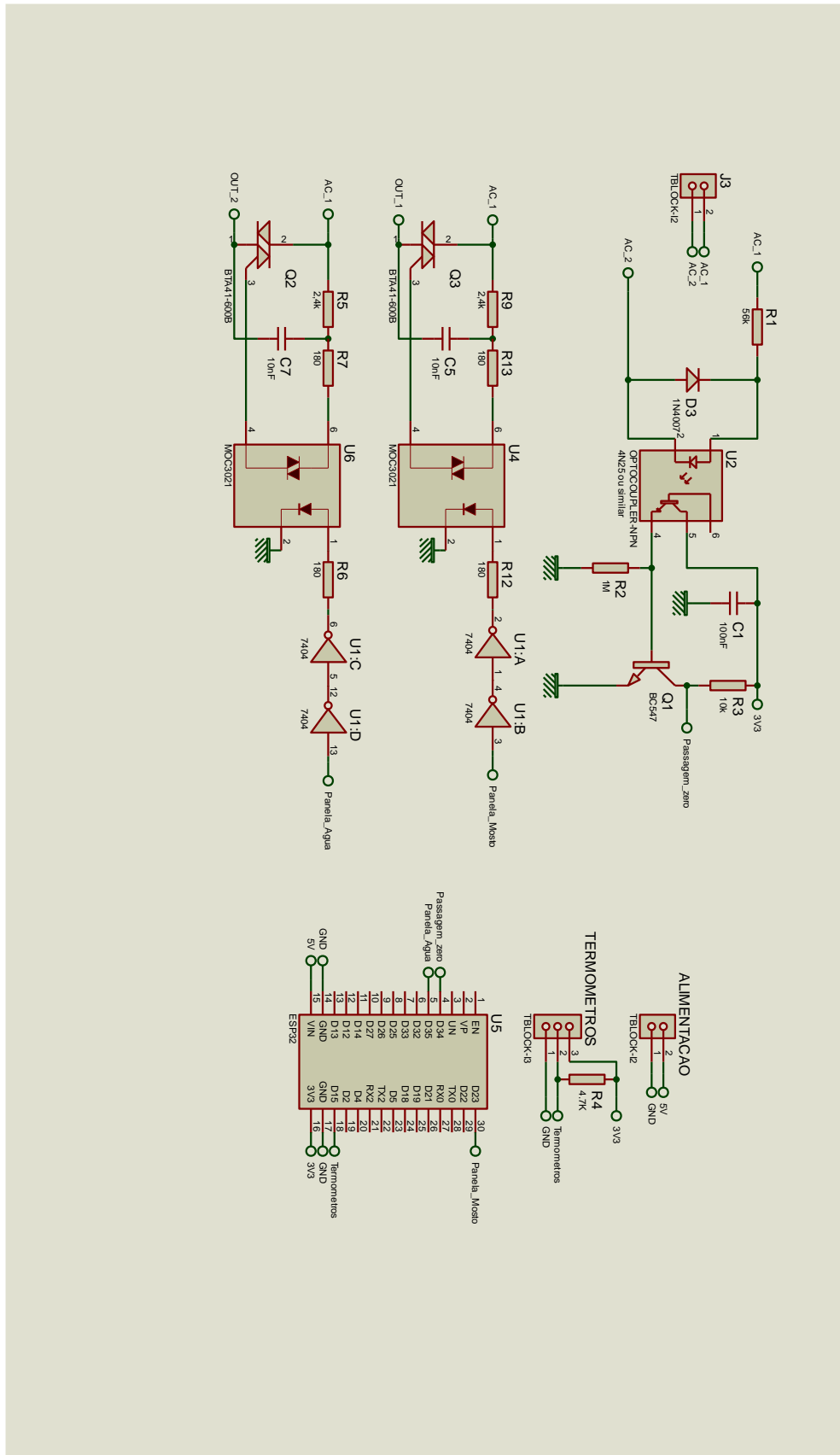
PERFIL, V. **Controle PID de Potência em Corrente Alternada - Arduino e TRIAC - Parte III****blogspot.com**Blogger, , 18 maio 2013. Disponível em: <http://automatobr.blogspot.com/2013/05/controle-de-potencia-em-corrente_18.html>. Acesso em: 14 nov. 2021

TECNOLOGIA, T. **Bomba Magnética Mp-15r Recirculação Cerveja Artesanal 110v**. Disponível em: <<https://www.cervejariainversa.com/equipamentos/bomba-magnetica-mp-15r-recirculacao-erveja-artesanal-110v>>. Acesso em: 14 nov. 2021.

BAU. **Controle de Potência AC com Arduino e Triac**. Disponível em: <<https://blog.baudaeletronica.com.br/potencia-ac-Arduino-Triac/>>. Acesso em: 14 nov. 2021.

6-Pin DIP Random-Phase Triac Driver Output Optocoupler (250/400 V Peak). Disponível em: <https://br.mouser.com/datasheet/2/308/1/MOC3023M_D-2316077.pdf>. Acesso em: 14 nov. 2021.

10 APÊNDICE A – ESQUEMÁTICO DO CIRCUITO



11 ANEXO A – FIRMWARE DO ESP32

```

#define BLYNK_PRINT Serial

/*BIBLIOTECAS*/
#include <OneWire.h>
#include <DallasTemperature.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

/*DECLARAÇÃO DOS PINOS*/
#define pinResPanela35L 13
#define pinResPanela20L 12
#define pinPaasagemZero 34
#define oneWireTermometro 15

/*VARIÁVEIS GLOBAIS*/
volatile int potMax35L = 8333;
volatile int map_potMax35L = 0;
volatile int potMaxTime35L = 0;
volatile int potMax20L = 8333;
volatile int map_potMax20L = 0;
volatile int potMaxTime20L = 0;
int limMin = 0;
int limMax = 8200;
float setTemp35L = 0;
float setTemp20L = 0;
bool flagRes35L = 0;
bool flagRes20L = 0;
float floatsensorTempPanela35L = 0;
float floatsensorTempPanela20L = 0;

/*VARIÁVEIS PARA BLYNK E WIFI*/
char auth[] = "oWbC3eLp3XWt9OttjNKfkLC-m7LrvAsp";
char ssid[] = "Cheiro verde";
char pass[] = "chicaamelia";

/*OUTRAS DECLARAÇÕES*/
OneWire oneWire(oneWireTermometro); //Definindo a porta onde estão os
termometros.
DallasTemperature sensors(&oneWire);
DeviceAddress sensorTempPanela35L = { 0x28, 0x19, 0x10, 0x79, 0x97, 0x20,
0x3, 0xE5 }; //Endereço dos termometros
DeviceAddress sensorTempPanela20L = { 0x28, 0xEE, 0x1E, 0x79, 0x97, 0x20,
0x3, 0xB0 };
BlynkTimer timer;

portMUX_TYPE mux = portMUX_INITIALIZER_UNLOCKED;

/*DECLARAÇÃO DAS FUNÇÕES UTILIZADAS*/
void passagem_zero();
void verifica_temp();

//Recebimento de valores do Blynk, convertendo para float
BLYNK_WRITE(V1) {
    setTemp35L = param.asFloat();
}

BLYNK_WRITE(V2) {

```

```

    potMax35L = param.asFloat();
    map_potMax35L = map(potMax35L, 0, 100, limMin, limMax); //Transformando a
base de 0 a 100 para 0 8333
    potMaxTime35L = 8333 - map_potMax35L - 10; //Achando o tempo desligado e
compensando o delay de 10 uS
}

BLYNK_WRITE(V4) {
    setTemp20L = param.asFloat();
}

BLYNK_WRITE(V5) {
    potMax20L = param.asFloat();
    map_potMax20L = map(potMax20L, 0, 100, limMin, limMax); //Transformando a
base de 0 a 100 para 0 8333
    potMaxTime20L = 8333 - map_potMax20L - 10; //Achando o tempo desligado e
compensando o delay de 10 uS
}

void setup() {
    //Comunicação
    Serial.begin(115200);
    Blynk.begin(auth, ssid, pass); //Configurando o Blynk
    sensors.begin();

    //Definição dos pinos
    pinMode(pinResPanela35L, OUTPUT);
    pinMode(pinResPanela20L, OUTPUT);
    pinMode(pinPaasagemZero, INPUT);

    //Interrupções
    attachInterrupt(digitalPinToInterrupt(pinPaasagemZero), passagem_zero,
CHANGE);
    timer.setInterval(100L, verifica_temp); // Timer para verificar a
temperatura
}

void loop() {
    Blynk.run();
    timer.run();
}

void passagem_zero() {
    if (flagRes35L == false) {
        digitalWrite(pinResPanela35L, LOW);
    }

    if (flagRes20L == false) {
        digitalWrite(pinResPanela20L, LOW);
    }

    if (flagRes35L == true && flagRes20L == true) { //Caso as duas painelas
precisam ser utilizadas ao mesmo tempo
        if (potMaxTime35L > potMaxTime20L) { //Caso o potência da panela de 35
litros seja menor
            delayMicroseconds(potMaxTime20L); //Deixo a resistência desligada por
potMaxTime20L microssegundos
            digitalWrite(pinResPanela20L, HIGH);
            delayMicroseconds(10); // tempo necessário apra o circuito do Triac
perceber o pulso
            digitalWrite(pinResPanela20L, LOW);

```

```

        delayMicroseconds (potMaxTime35L - potMaxTime20L + 10);
        digitalWrite (pinResPanela35L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela35L, LOW);
    }

    if (potMaxTime20L > potMaxTime35L) { //Caso a potência da panela de 20
litros seja menor
        delayMicroseconds (potMaxTime35L);
        digitalWrite (pinResPanela35L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela35L, LOW);
        delayMicroseconds (potMaxTime20L - potMaxTime35L + 10);
        digitalWrite (pinResPanela20L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela20L, LOW);
    }

    if (potMaxTime20L == potMaxTime35L) { //Caso a potência das panelas
sejam iguais
        delayMicroseconds (potMaxTime35L);
        digitalWrite (pinResPanela35L, HIGH);
        digitalWrite (pinResPanela20L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela35L, LOW);
        digitalWrite (pinResPanela20L, LOW);
    }
}

    if (flagRes35L == true && flagRes20L == false) { //Ligando somente a
panela de 35 litros
        delayMicroseconds (potMaxTime35L);
        digitalWrite (pinResPanela35L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela35L, LOW);
    }

    if (flagRes35L == false && flagRes20L == true) { //Ligando somente a
panela de 20 litros
        delayMicroseconds (potMaxTime20L);
        digitalWrite (pinResPanela20L, HIGH);
        delayMicroseconds (10);
        digitalWrite (pinResPanela20L, LOW);
    }
}

void verifica_temp() {
    sensors.requestTemperatures(); //Requisitando as temperaturas
    floatsensorTempPanela35L =
sensors.getTempC(sensorTempPanela35L); //Armazenando as temperaturas em
variáveis globais
    floatsensorTempPanela20L = sensors.getTempC(sensorTempPanela20L);
    Blynk.virtualWrite (V0, sensors.getTempC(sensorTempPanela35L)); //Enviando
as temperaturas para o Blynk
    Blynk.virtualWrite (V3, sensors.getTempC(sensorTempPanela20L));

    if (floatsensorTempPanela35L < setTemp35L - 0.3 && map_potMax35L >
limMin) { //Fazendo a verificação
        flagRes35L = 1;
    } else {

```

```
    flagRes35L = 0;
  }
  if (floatsensorTempPanela20L < setTemp20L - 0.3 && map_potMax20L >
limMin) {
    flagRes20L = 1;
  } else {
    flagRes20L = 0;
  }
}
```