

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA QUÍMICA

**DESENVOLVIMENTO DO PROTÓTIPO: CERVEJARIA ARTESANAL MÓVEL
AUTOMATIZADA - CAMA**

Bryan Jonathan do Nascimento

Trabalho de Graduação apresentado ao
Departamento de Engenharia Química da
Universidade Federal de São Carlos

Orientadora: **Profa. Dra. Fernanda Perpétua Casciotori**

BANCA EXAMINADORA

Trabalho de Graduação apresentado no dia 16 de novembro de 2021 perante a seguinte banca examinadora:

Orientadora: Profa. Dra. Fernanda Perpétua Casciotori, DEQ/UFSCar

Convidado: Erick Willian Nascimento, UNIP

Professora da Disciplina: José Maria Corrêa Bueno, DEQ/UFSCar

AGRADECIMENTOS

Agradeço primeiramente a Deus, por estar sempre comigo em todos os momentos da minha trajetória profissional e pessoal.

A Cintia Martins, minha namorada a qual a 10 anos atrás escolheu unir forças comigo em busca de um futuro melhor. Não há palavras que permita descrever o quanto você foi importante em todos esses anos.

A minha família, da qual recebi imenso apoio e confiança a qual é sempre determinante em todas as escolhas que fiz durante o meu caminho.

A minha Vó, a pessoa a qual pude aprender uma das melhores qualidades que uma pessoa pode ter, humildade.

A minha mãe, da qual herdei muitas qualidades, dentre elas o valor de um trabalho bem executado.

Aos meus irmãos simplesmente por existirem, e assim, me fazerem pensar que um futuro melhor para mim, pode significar um futuro melhor para eles.

Aos meus colegas de graduação, por fazerem os momentos fáceis ou difíceis da graduação, mais engraçados.

A minha tutora do PET, orientadora de estágio e orientadora de TG, prof. Fernanda Casciotori pela disponibilidade e mentoria.

A todos os meus colegas de trabalho que tive durante o estágio, que me mostraram como é a vida após universidade.

Agradeço ao PET e a todos os Petianos que me permitiram vivenciar experiências e projetos incríveis que inclusive tem como produto esse trabalho. Cabe aqui salientar que esse projeto, Cervejaria Artesanal Móvel Automatizada – CAMA, foi desenvolvido no grupo alimentos do PET e que sem vocês, Petianos, isso não seria possível.

À UFSCar por me propiciar um ensino de excelência o qual mudou minha vida.

RESUMO

A cerveja é uma bebida mundialmente consumida, produzida a partir de uma combinação de grãos que gera um mosto açucarado que, por sua vez, passa posteriormente por fermentação, resultando na bebida final. Existem variações quanto ao método de produção, o que origina vários diferentes tipos de cerveja. Porém, de um modo geral, a legislação no Brasil define que: “Cerveja é a bebida resultante da fermentação, a partir da levedura cervejeira, do mosto de cevada malteada ou de extrato de malte...”. Nos últimos anos, houve um aumento do número de micro cervejarias no Brasil, surgindo assim a oportunidade de associar o aprendizado de processos, característico dos cursos de Engenharia, com soluções mais baratas para o sistema de controle eletrônico da planta industrial. Diante do exposto, o objetivo do presente trabalho é apresentar um protótipo que combina o processo de produção de uma mini cervejaria com tecnologias eletrônicas acessíveis e de baixo custo, buscando assim automação e controle da planta cervejeira, sendo passível de replicação em micro cervejarias artesanais. O hardware foi desenvolvido utilizando um pequeno computador de placa única o Raspberry Pi, um módulo de relês mecânicos disponível comercialmente, um controlador de relês de estado sólido produzido artesanalmente e outros acessórios como válvulas e bombas. A programação foi desenvolvida em uma plataforma baseada em fluxo, Node-RED, criada pela IBM, com base em JavaScript. Dentre as funcionalidades do protótipo denominado cervejaria artesanal móvel automatizada (CAMA), incluem-se abertura e fechamento automático de válvulas, acionamento automático de resistências elétricas e atuação automática de eletrobombas. Desta forma, os processos de aquecimento e bombeamento dos fluidos entre os diferentes tanques pelos quais passa o mosto cervejeiro pode se dar de forma bem controlada e automatizada, contribuindo para a qualidade da bebida e para a otimização da produção, mesmo nas pequenas cervejarias. O centro de controle e o software desenvolvido atendeu aos seus objetivos, realizando uma rampa de temperatura na parte de aquecimento do processo juntamente com uma recirculação e mantendo uma temperatura de fermentação na parte de refrigeração.

Palavras-chave: cerveja, Raspberry Pi, controle, mini cervejaria, automação de processos, cerveja artesanal.

ABSTRACT

Beer is a drink consumed worldwide, based on a combination of grains that generate a sugary must which, in turn, undergoes fermentation, prepared in the final drink. There are variations in the method of production, which gives rise to different types of beer. However, in general, the legislation in Brazil defines that: “Beer is the beverage resulting from fermentation, from brewer's yeast, malted barley must or malt extract...”. In recent years, there has been an increase in the number of micro breweries in Brazil, giving rise to the opportunity to associate the process process, characteristic of engineering courses, with cheaper solutions for the electronic control system of the industrial plant. Given the above, the objective of this work is to present a prototype that combines the production process of a mini brewery with low-cost and returned technologies, thus seeking automation and control of the brewery, being capable of replication in micro craft breweries. The hardware was developed using a small single board computer the Raspberry Pi, a commercially available mechanical relay module, a handcrafted solid state relay driver and other accessories such as valves and pumps. The programming was developed on a stream-based platform, Node-RED, created by IBM, based on JavaScript. Among the functionalities of the prototype called automated mobile craft brewery (CAMA), it includes automatic opening and closing of valves, automatic activation of electrical resistances and automatic activation of electric pumps. In this way, the processes of heating and pumping fluids between the different tanks through which the brewer's must passes can take place in a well-controlled and automated way, contributing to the quality of the drink and to the optimization of production, even in small breweries. The control center and the developed software met their goals, performing a temperature ramp in the heating part of the process together with a recirculation and maintaining a fermentation temperature in the refrigeration part.

Keywords: beer, Raspberry Pi, control, mini brewery, process automation, craft beer.

SUMÁRIO

Banca Examinadora	i
Agradecimentos	ii
Resumo	iii
Abstract	iv
Lista de Figuras	v
1-INTRODUÇÃO	1
2-FUNDAMENTOS TEÓRICOS	4
2.1-PRODUÇÃO DE CERVEJA.....	4
2.1.1-Matéria-prima	4
2.1.2-Processo de produção	5
2.1.2.1-Brassagem.....	7
2.1.2.2-Fermentação.....	13
2.1.2.3-Maturação, acabamento e envase.....	19
2.2-HARDWARE.....	21
2.2.1-Raspberry Pi	21
2.2.2-Relés mecânicos e sólidos	23
2.2.3-Sensor de temperatura	24
2.3-NODE-RED.....	25
3-A CENTRAL DE SOFTWARE	29
4-O SOFTWARE	33
4.1-VISÃO GERAL.....	33
4.2-FLUXO.....	37
4.3-MOSTURA.....	38
4.4-TANQUE DE LAVAGEM E TANQUE DE FERVURA.....	51
4.5-RESFRIAMENTO.....	55
4.6-FERMENTAÇÃO E MATURAÇÃO.....	57
5-RESULTADOS E DISCUSSÕES	60
5.1-CONTROLE DE REFRIGERAÇÃO.....	60

5.2-CONTROLE DE AQUECIMENTO.....	63
5.3-CONTROLE DE RESFRIAMENTO.....	67
6-CONSIDERAÇÕES FINAIS.....	67
7-REFERÊNCIAS.....	70
8-APÊNDICE.....	73
8.1-PROGRAMAÇÃO EXPORTADA DO NODE-RED.....	73
8.2-LICENÇA HIGHCHARTS.....	97

LISTA DE FIGURAS

Figura 1.1.	Número de registro de estabelecimento por ano.	1
Figura 1.2.	Distribuição do número de cervejarias por município no Brasil a partir de círculos Proporcionais.	2
Figura 2.1.	<i>Humulus lupulus</i> planta de lúpulo (fêmea) com estróbilos.	5
Figura 2.2.	Visão geral do processo de fermentação para uma cervejaria.	6
Figura 2.3.	Efeitos da temperatura durante os intervalos da mostura.	8
Figura 2.4.	Atividade da alfa e beta-amilase: tempo e temperatura.	8
Figura 2.5.	Efeitos da temperatura sobre o mosto.	9
Figura 2.6.	Estruturas alvo de algumas enzimas.	10
Figura 2.7.	Mudança da viscosidade do mosto durante a degradação do amido.	10
Figura 2.8.	Temperatura ótima e intervalos de ótimo pH das enzimas.	11
Figura 2.9.	Tanque para mosturação, filtragem e lavagem.	12
Figura 2.10.	Valores típicos de temperatura, concentração de sólidos e leveduras em suspensão durante a fermentação de uma Ale.	14
Figura 2.11.	Valores típicos de concentração de carboidratos durante a fermentação.	15
Figura 2.12.	Densímetro flutuante.	17
Figura 2.13.	Tubo em U com atuador piezoelétrico.	18
Figura 2.14.	Fermentador típico de produção caseira de cerveja.	18
Figura 2.15.	Intervalos típicos de temperatura para fermentação e maturação.	19
Figura 2.16.	Raspberry Pi Modelo 3 B+.	21
Figura 2.17.	Diagrama de pinos do Raspberry Pi Modelo 3 B+.	22
Figura 2.18.	Relé mecânico tipo NA/NF.	24
Figura 2.19.	Relé convencional (a) e relé de estado sólido (b).	24
Figura 2.20.	Esquema de um opta-acoplador em conjunto com o TRIAC.	25
Figura 2.21.	Formas de onda de uma corrente alternada controlada.	26
Figura 2.22.	Interface de programação Node-RED.	27
Figura 2.23.	Exemplo de interface do painel Node-RED.	28
Figura 3.1.	Central de controle visão geral externa.	29
Figura 3.2.	Central de controle visão geral interna.	29

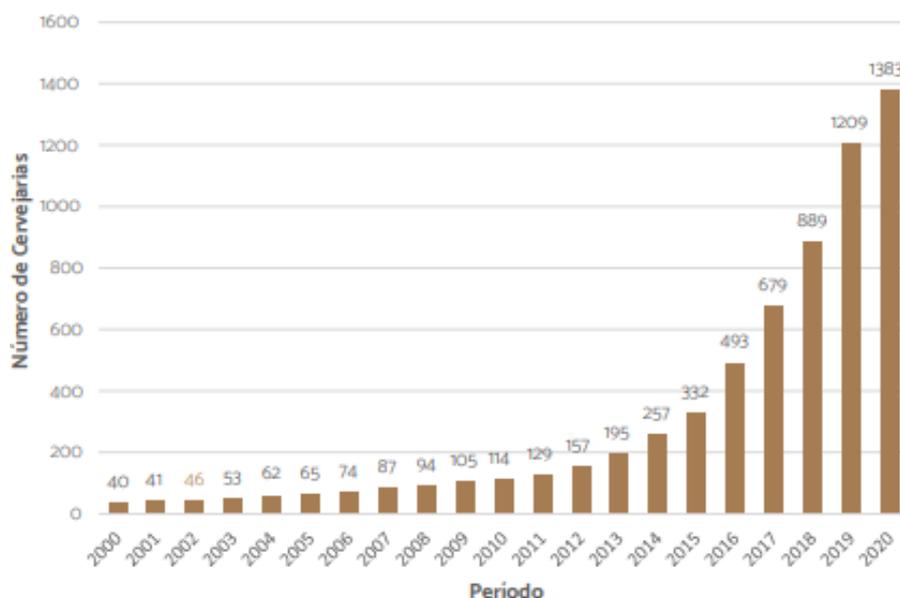
Figura 3.3.	Central de Controle visão interna.	30
Figura 3.4.	Central de Controle visão interna tampa.	32
Figura 4.1.	Variáveis importantes e controle criado para CAMA.	33
Figura 4.2.	Fluxos criados para controle da CAMA.	34
Figura 4.3.	Blocos de controle criados para operação da CAMA.	35
Figura 4.4.	Programação do Fluxo.	37
Figura 4.5.	Painel de operação do Fluxo.	38
Figura 4.6.	Programação da Mostura.	38
Figura 4.7.	Painel de operação da Mostura.	39
Figura 4.8.	Programação função F1.	40
Figura 4.9.	Programação do gráfico	42
Figura 4.10.	Notificação de temperatura inserida inválida.	44
Figura 4.11.	Notificação de temperatura selecionada.	44
Figura 4.12.	Notificação de reset realizado.	45
Figura 4.13.	Programação função A1.	46
Figura 4.14.	Janela de confirmação de resistência submersa.	47
Figura 4.15.	Programação função A2.	47
Figura 4.16.	Programação função Validação.	48
Figura 4.17.	Programação função Reiniciar.	49
Figura 4.18.	Programação função Deleta Dados.	49
Figura 4.19.	Programação função Controle Potência.	50
Figura 4.20.	Programação da Lavagem.	51
Figura 4.21.	Programação da Fervura.	51
Figura 4.22.	Função F1 para um sensor de temperatura.	53
Figura 4.23.	Programação do Resfriamento.	55
Figura 4.24.	Programação da função On/Off do resfriamento.	56
Figura 4.25.	Programação da Fermentação/Maturação.	57
Figura 4.26.	Janela de confirmação de refrigerador em potência máxima.	58
Figura 4.27.	Programação da função On/Off do fermentador.	59
Figura 5.1.	Esquema para controle de refrigeração.	60
Figura 5.2.	Resultados do controle de refrigeração.	61
Figura 5.3.	Esquema para o teste de aquecimento.	63
Figura 5.4.	Disposição dos sensores no teste de aquecimento.	64

Figura 5.5.	Recirculação efetuada no teste de aquecimento.	64
Figura 5.6.	Rampa de temperatura objetivo.	65
Figura 5.7.	Resultados do controle de aquecimento.	65
Figura 5.8.	Resultados do aquecimento topo e fundo.	66
Figura 5.9.	Resultados do controle de resfriamento.	67

1 INTRODUÇÃO

Nos últimos anos o número de cervejarias no Brasil tem aumentado significativamente, o número de registros de estabelecimentos em 2020 é cerca de 4 vezes maior do que em 2015 (CERVBRASIL, 2020).

Figura 1.1. Número de registro de estabelecimento por ano.



Fonte: CERVBRASIL, 2020.

Só em 2020, 174 novas cervejarias foram registradas, sendo que todas as unidades da federação possuem no mínimo uma cervejaria. Porém, vale destacar que existe uma alta concentração desses estabelecimentos na região sul e sudeste, representando 85,6% dos estabelecimentos em 2020 (CERVBRASIL, 2020).

Figura 1.2. Distribuição do número de cervejarias por município no Brasil a partir de círculos Proporcionais.



Fonte: CERVBASIL, 2020.

Utilizando o histórico dos últimos 10 anos se projeta uma taxa de crescimento de 26,6% ao ano, o que seria equivalente a termos 2418 registros de estabelecimentos em 2022 (CERVBASIL, 2020).

Em 2016, o setor empregava 2,2 milhões de pessoas ao longo da cadeia produtiva e foi responsável por 1,6% PIB (CERVBASIL, 2016).

No cenário mundial, em 2020, o Brasil foi o terceiro maior produtor de cerveja, representando cerca 8,4% da produção mundial, e o segundo maior da América, representando cerca 24,7% da produção americana, produzindo no total cerca de 15,19 milhões de metros cúbicos de cerveja (BARTH-HAAS, 2020).

Em relação ao ano anterior, 2019, a produção brasileira aumentou 4,9% (BARTH-HAAS, 2020).

Com a evolução tecnológica, as indústrias têm forte demanda de melhorias não apenas de seus processos de produção, mas de todo o ambiente produtivo. O desenvolvimento da tecnologia da informação (TI) está fazendo com que novas indústria passem por uma transformação se aproximando do conceito de Indústria 4.0 que é chamada também de *Industrial Internet of Things*. (CHENG et al., 2015)

Essas novas tecnologias possibilitam reduzir custos, fornecer soluções eficazes e integrar as comunicações entre clientes, fornecedores e fabricantes (URBIKAIN et al., 2016).

A Indústria 4.0 já é uma meta de alguns governos como o da Alemanha que possui uma rede mundial de indústrias a qual tem o nome de “Platform Industrie 4.0”, o projeto promove a transição da indústria. (PLATTFORM, 2021).

Nesse contexto de demanda por soluções de modernização industrial e ascensão de novas cervejarias, algumas tecnologias acessíveis e de baixo custo que podem ser implementadas em conjunto nas novas cervejarias como o pequeno e portátil computador de placa única Raspberry Pi e o software de código aberto baseado em fluxo Node-RED.

Alguns trabalhos foram desenvolvidos com o uso dessas ferramentas, pode-se citar como exemplo o uso do Node-RED no ambiente industrial para o monitoramento de uma máquina (FERENCZ, 2019), a conclusão desse trabalho mostra que a prototipagem de aplicativos IoT em uma indústria ocorre de forma rápida e oferece a modificação da interface do software de acordo com a necessidade. Essa característica mantém o sistema de monitoramento da indústria de forma competitiva, adaptando-o conforme as necessidades do processo, mesmo que esse mude com o tempo.

Outro trabalho que se pode citar foi o desenvolvimento de um controle automático e remoto para um laboratório industrial (DOMÍNGUEZ et al., 2020), o trabalho mostrou ser possível desenvolver uma aplicação onde nos alunos configuram um PID em uma malha de controle de uma planta industrial laboratorial da Universidade de León.

A proposta do presente trabalho é integrar a tecnologia IoT através do Node-RED em conjunto com o Raspberry Pi em uma mini produção de cerveja de forma que seja possível automatizar parte do processo de produção. O trabalho visa proporcionar uma alternativa de baixo custo a inserção de micro cervejarias nas chamadas Indústrias 4.0, apresentando um sistema inicial que pode ser personalizado sob demanda para o tipo de planta produtiva.

2 FUNDAMENTOS TEÓRICOS

2.1 PRODUÇÃO DE CERVEJA

Os primeiros campos de cultivo de grão surgiram na Ásia por volta de 9000 a.C., acredita-se que a cevada e o trigo foram os plantios escolhidos devido aos seus aspectos de sabor e germinação. Na característica nutricional a cerveja se aproxima do pão, pois ambos são produzidos a partir de grãos, podendo então a cerveja ser chamada de “pão líquido” (MORADO).

Como explicitado na introdução deste trabalho, apenas bebidas fermentadas com um limite mínimo malte de cevada pode ser consideradas cerveja, diferenciando-se de outras bebidas como hidromel, sidra, vinho, saquê, os quais têm a maior parte de seu malte provenientes do mel, maçã ou pera, uva e arroz, respectivamente. A Lei da Pureza (Reinheitsgebot) alemã é ainda mais rigorosa classificando a cerveja como uma bebida oriunda apenas do malte de cevada, lúpulo, levedura e água. (MORADO, 2011)

2.1.1 Matérias-primas

Das matérias-primas utilizadas na cerveja, 90% corresponde a água, podendo o processo consumir até 12 volumes de água para 1 volume de cerveja, sendo a qualidade da água utilizada imprescindível para o resultado final da produção.

O malte utilizado na produção passa por um processo de maltagem que consiste na modificação dos grãos a partir de um processo enzimático para a produção de açúcares que serão utilizados na fermentação (FARBER,2019). Entende-se malte o resultado da maltagem da cevada, demais grãos devem ser acompanhados da especificação, como por exemplo, malte de arroz.

O lúpulo ou *Humulus lupulus* é uma planta trepadeira que produz pequenos cones, como os da Figura 2.1, dos quais se tem interesse nas resinas que por sua vez conferem o aroma, e o amargor que equilibra o doce do malte, contribuindo para o estilo da cerveja. Além disso, o lúpulo tem propriedades antibacterianas e estabilizadora da formação de espuma que contribui para a armazenagem e processo de produção, respectivamente. (FARBER, 2019).

Figura 2.1. *Humulus lupulus* planta de lúpulo (fêmea) com estróbilos.



Fonte: FARBER, 2019.

As leveduras mais utilizadas na produção de cerveja são as *Saccharomyces cerevisiae* para produção de cervejas estilo Ale e *Saccharomyces pastorianus* na produção de cervejas estilo Lager, elas são inoculadas na etapa de fermentação bioquímica. Até o século XIX, não se tinha compreensão de todos os mecanismos envolvidos no processo de fermentação, foi a partir da publicação de trabalhos, como por exemplo, os de Louis Pasteur (1822-1895), que houve um avanço científico significativo na área (MORADO, 2011).

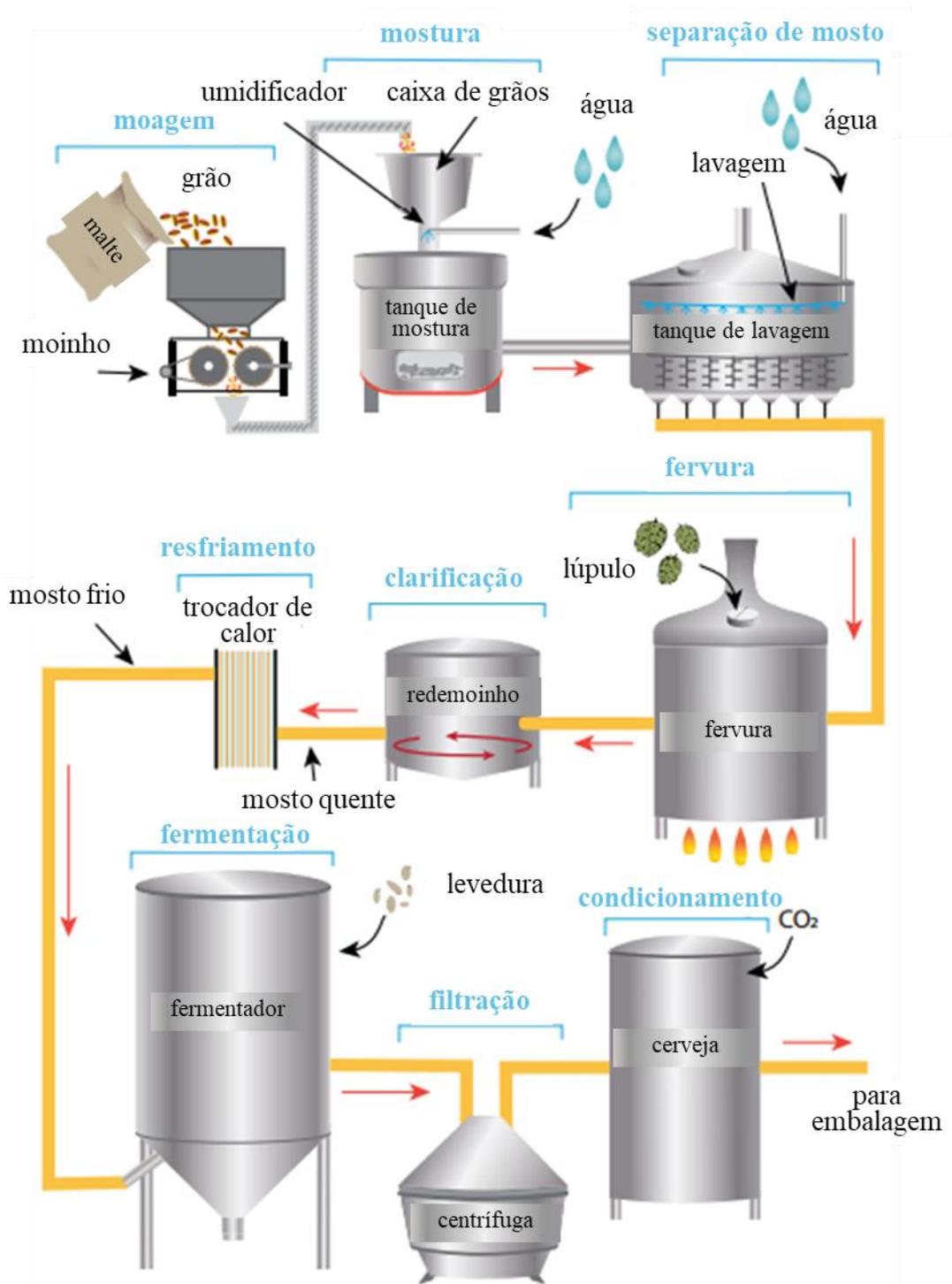
Essas matérias primas são as principais na produção da cerveja, porém, outras substâncias e adjuntos podem ser necessários para a melhoria do processo ou obtenção de características distintas (MORADO, 2011).

2.1.2 Processo de Produção

O processo de produção de cerveja foi aprimorado ao longo do tempo, plantas industriais mais antigas se aproveitavam da gravidade para realizar o transporte do líquido entre as operações de produção, dando um aspecto mais vertical as indústrias cervejeiras. O processo completo engloba as etapas: maltagem, brassagem, fermentação, maturação, acabamento e envase (MORADO, 2011).

A primeira etapa da produção é a maltagem, porém esse processo costuma acontecer em uma indústria separada que é especializada nesse procedimento.

Figura 2.2. Visão geral do processo de fermentação para uma cervejaria.



Fonte: FARBER, 2019 (traduzida).

2.1.2.1 Brassagem

A etapa de Brassagem tem o objetivo de extrair o amido contido no malte produzindo um líquido com açúcares, proteínas e outras substâncias. Esse líquido é denominado mosto e necessita de três subprocessos. A moagem, Figura 2.2 – moagem, que deve ser feita logo no início do processo e serve para expor o endosperma do grão a ação enzimática do processo. A qualidade da moagem tem consequências diretas na filtração. A mostura, Figura 2.2 – mostura, é a adição de água aos grãos e a manutenção dessa mistura a certas temperaturas por determinados períodos de tempo, essa técnica é conhecida como “rampa de temperatura” e serve obter efeitos específicos. Por exemplo, nos intervalos de temperatura (MORADO, 2011):

- 40°C à 45°C - Ocorre a ativação enzimática, a solubilização do amido, a entrada de enzimas na solução e a ação da betaglucanase que ajudará na solubilização do amido.
- 50°C à 55°C - Ocorre a quebra das proteínas gerando peptídeos e aminoácidos, esse processo é importante já que os grãos estão rodeados por proteínas e a quebra dessas proteínas expõem o amido para o pico da ação enzimática que ocorre no próximo intervalo de temperatura. Essa etapa pode ser usada para obtenção de diferentes brilho e espuma da cerveja.
- 60°C à 72°C - Nessa etapa, conhecida como sacarificação, são ativadas duas principais enzimas a alfa-amilase e a beta-amilase, cada qual tem temperaturas ótimas em pontos diferentes dentro desse intervalo e proporcionam características diferentes ao corpo da cerveja.
- 76°C à 78°C - Ocorre a inativação enzimática, fazendo com que aspectos obtidos nas etapas anteriores não sejam modificados durante os demais processos.

Diferentes etapas podem ser encontradas em FARBER, 2019, as quais são reproduzidas na Figura 2.3:

Figura 2.3. Efeitos da temperatura durante os intervalos da mostura.

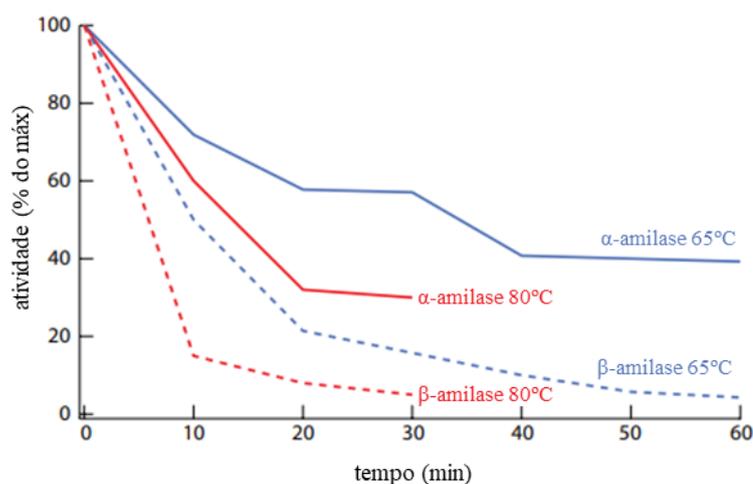
Temperatura	Etapa (efeito)
35–40 °C (95–104 °F)	Descanso ácido (liberação de ácido ferúlico da hemicelulose; liberação de fosfato via fitase)
45–50 °C (113–122 °F)	Descanso proteico (hidrólise de proteínas; aumentar de ANL; diminuir a estabilidade da espuma)
45–55 °C (113–131 °F)	Descanso β -glucanase (hidrólise glucanos; diminuir viscosidade)
60–65 °C (140–149 °F)	Descanso produção de maltose (temperatura ótima β -amilase)
66–70 °C (150–158 °F)	Descanso sacarificação (promove atividade de α -amilase)
76–78 °C (168–172 °F)	Mash-out (diminuir viscosidade, desativar enzimas)

Fonte: FARBER, 2019 (traduzida).

Cabe salientar que não existe uma receita única de rampa de temperatura e cada produtor desenvolve-a para obter as características desejadas. (MORADO, 2011)

A rampa de temperatura, na grande maioria das produções sempre ocorre da temperatura mais baixa para a mais alta devido a uma melhor atividade das enzimas em temperaturas mais baixas, sendo também, outra justificativa a impossibilidade da reativação de uma enzima que foi inativada devido a altas temperaturas. A Figura 2.4 mostra o efeito da temperatura na atividade das enzimas:

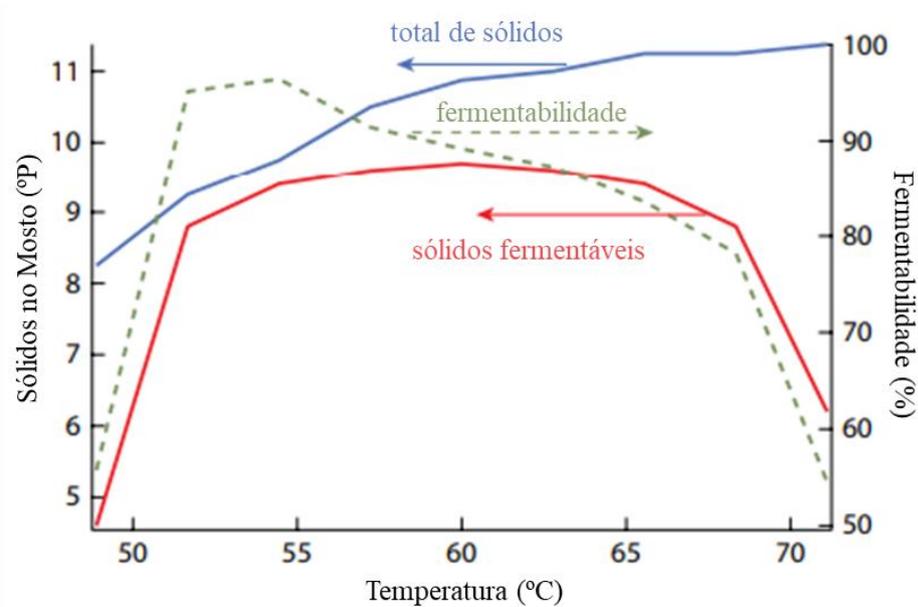
Figura 2.4. Atividade da alfa e beta-amilase: tempo e temperatura.



Fonte: FARBER, 2019 (traduzida).

A Figura 2.5 ilustra os efeitos das temperaturas sobre as características do mosto, cabe salientar que esse efeito ainda depende do tipo de malte, rampa de temperatura e da capacidade da levedura. É possível observar que conforme se aumenta a temperatura na mosturação se obtém mais sólidos, porém, a partir da temperatura de 60°C muito desses sólidos não são fermentáveis, diminuindo a fermentabilidade do mosto.

Figura 2.5. Efeitos da temperatura sobre o mosto.



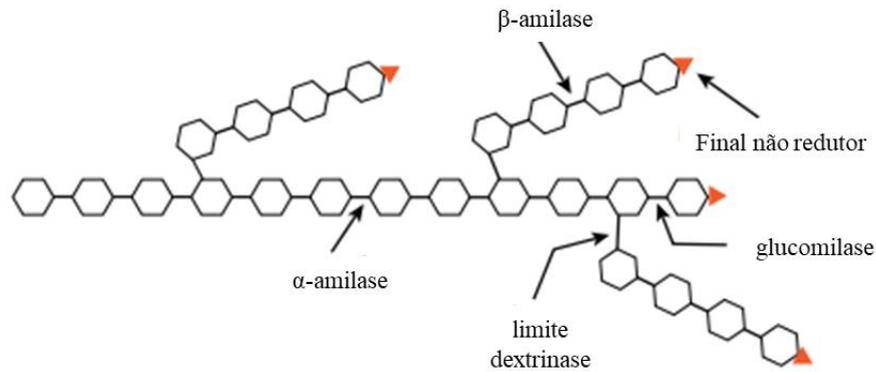
Fonte: FARBER, 2019 (traduzida).

O processo de degradação do amido pode ainda ser separado em três fases, que estão incluídas nas faixas de temperaturas apresentadas anteriormente, as etapas são:

- Gelatinização: onde as moléculas de água ficam entre os grânulos de cristalino do amido substituindo as ligações de hidrogênio o que faz com que os grânulos inchem e estourem liberando grandes moléculas de amido e aumentando a viscosidade do mosto.
- Liquefação: ocorre uma diminuição da viscosidade principalmente pela quebra do amido na ligação glicosídica alfa (1→4) pela alfa-amilase.
- Sacarificação: quando ocorre a hidrólise completa do amido com a atividade combinada da alfa-amilase que como dito age nas ligações alfa (1→4) o que gera uma nova extremidade redutora sobre a qual age a beta-amilase.

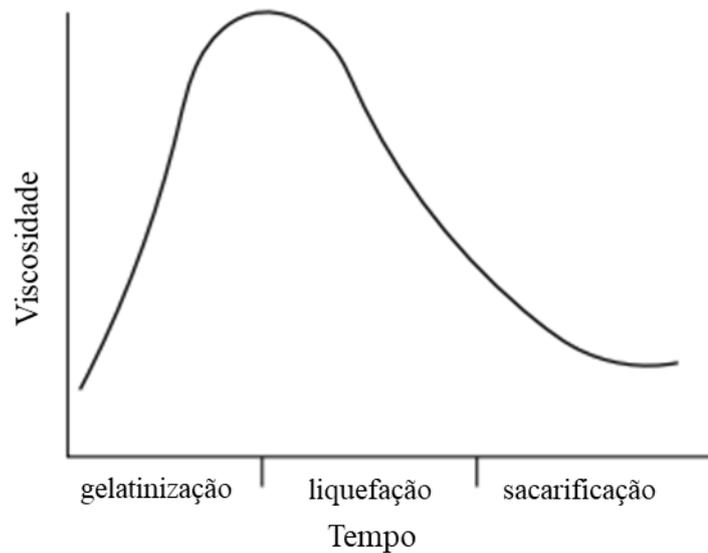
A Figura 2.6 ilustra as estruturas alvo de algumas enzimas incluindo a alfa-amilase e beta-amilase, a Figura 2.7 ilustra o perfil da viscosidade do mosto.

Figura 2.6. Estruturas alvo de algumas enzimas.



Fonte: FARBER, 2019 (traduzida).

Figura 2.7. Mudança da viscosidade do mosto durante a degradação do amido.



Fonte: FARBER, 2019 (traduzida).

Além da temperatura, outra variável importante para a produção de cerveja é o pH da mistura, isso pois as reações enzimáticas são diretamente afetadas pelo pH. Na Figura 2.8 é

possível ver as faixas ótimas de temperatura e pH, destacando que para alfa-amilase e beta-amilase que são as principais enzimas da Brassagem o pH está entre 5.4 e 5.8.

Figura 2.8. Temperatura ótima e intervalos de ótimo pH das enzimas.

Enzima	Temperatura Ótima	PH Ótimo
β -Glucanases	40–60 °C (104–140 °F)	4.5–4.8
Proteases	45–50 °C (113–122 °F)	3.9–7.2
Lipoxigenase	45–55 °C (95–104 °F)	6.5–7.0
Fosfatase	50–53 °C (95–104 °F)	5.0
Limit destrinase	55–65 °C (95–104 °F)	5.1
β -amilase	60–65 °C (140–149 °F)	5.4–5.5
α -amilase	70–74 °C (158–165 °F)	5.6–5.8

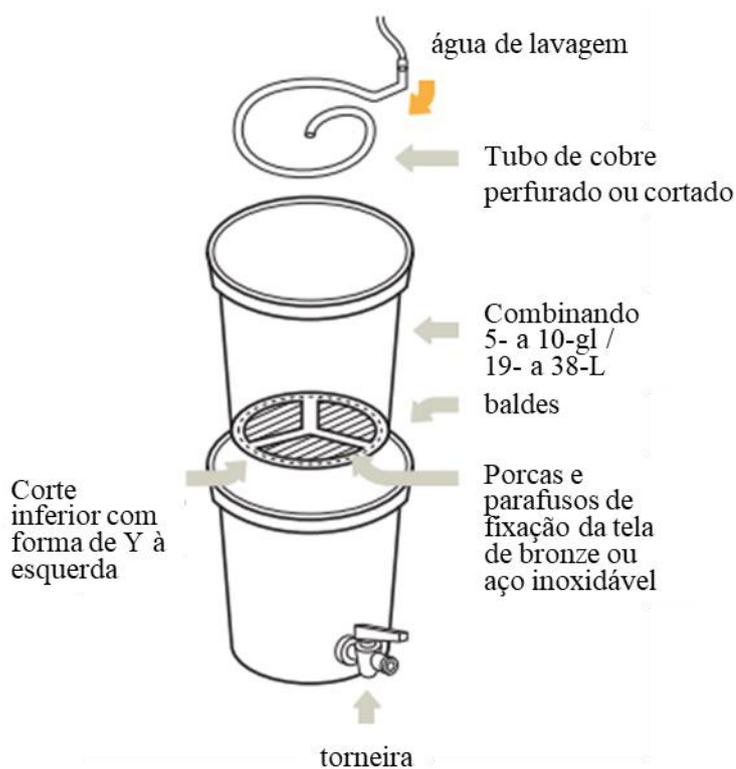
Fonte: FARBER, 2019 (traduzida).

Após a mostura ocorre a filtração do mosto, Figura 2.2 – separação de mosto, que consiste em separar o bagaço do mosto líquido. Ela pode ser executada de duas formas, com filtros de placas ou com tina de clarificação, nesta última, utiliza-se um fundo falso com furos, em que a palha dos grãos forma um elemento filtrante natural. Neste caso, a moagem deve preservar essa casca dos grãos. Uma etapa opcional é a lavagem, Figura 2.2 – sparging, que serve para se recuperar o máximo de extrato dos grãos. Após a filtragem que dá origem ao chamado mosto primário, adiciona-se água, em uma temperatura ideal e com vazão controlada aos grãos ainda húmidos, gerando assim, um mosto secundário o qual pode ser misturado ao primário (MORADO, 2011).

A finalização da lavagem ocorre quando, o mosto resultante atinge o pH excede 6 ou a gravidade específica cai abaixo de 1,005, a partir disso a concentração de fenólicos e minerais extraídos do grão é excessiva, compostos esses que prejudicam a qualidade da cerveja (FARBER, 2019).

No caso da produção caseira em pequena escala, a maioria dos cervejeiros costumam fazer a mosturação, filtragem e lavagem, no mesmo recipiente, podendo se utilizar equipamentos filtrantes e técnicas de lavagem distintas. A Figura 2.9 ilustra um equipamento onde é possível fazer as três operações, utilizando um dispersor e um fundo falso. Neste caso, uma recirculação, também chamada de Vorlaufing, é utilizada para filtragem e posteriormente, após a retirada do líquido primário, se faz a lavagem (MOSHER, 2015).

Figura 2.9. Tanque para mosturação, filtragem e lavagem.



Fonte: MOSHER, 2015 (traduzida).

Para a finalização da Brassagem se realiza a fervura do mosto, Figura 2.2 – fervura, que serve para estabilizar o mosto para que seja usado na fermentação. Nessa etapa ocorre a esterilização do mosto, eliminando micro-organismos que podem ser concorrentes das leveduras na etapa de fermentação, também ocorre a evaporação de aromas indesejáveis, os aldeídos, que lembram legumes cozidos e, por fim, a caramelização de açúcares que contribuem para a cor da cerveja. Na Fervura também é realizada a lupulagem, onde se adiciona o lúpulo para obtenção de amargor e aroma desejáveis (MORADO, 2011).

Durante a fervura ocorre a aglutinação de proteínas que formam uma espécie de massa chamada trub, que assim como resíduos de lúpulo, precisam ser separados para a clarificação do mosto e para evitar problemas nos processos seguintes de produção. Essa clarificação, Figura 2.2 – clarificação, geralmente é feita por meio de um redemoinho, também chamado de whirlpool, que concentra esses resíduos no meio do tanque.

Finalizando a Brassagem, ocorre o resfriamento, Figura 2.2 – resfriamento, que precisa ocorrer de forma rápida para evitar contaminação e formação de aromas (MORADO, 2011),

os equipamentos mais comuns utilizados nessa etapa são o trocador de calor de placas e o resfriador por imersão, este último consiste em uma serpentina o qual é imerso no mosto e circula o líquido refrigerante, normalmente água, por dentro.

2.1.2.2 Fermentação

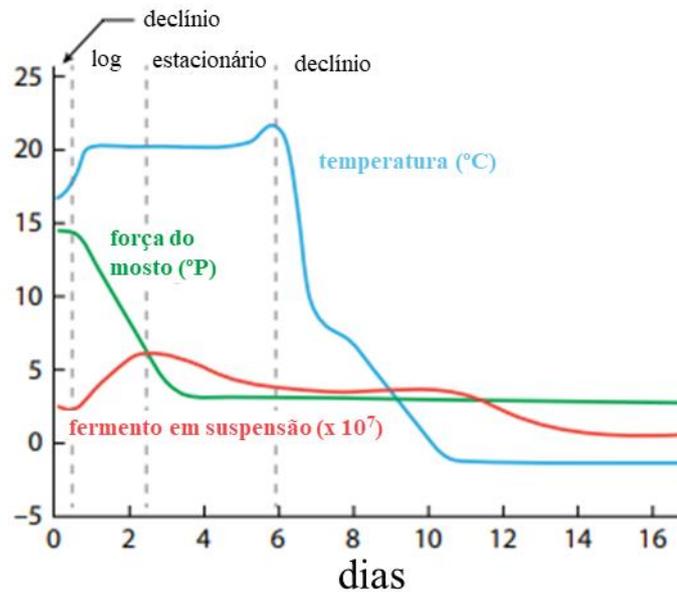
A fermentação, Figura 2.2 – fermentação, consiste na conversão de açúcares do mosto que são em maior parte maltose e glicose, em álcool e gás carbônico. É desejável que essa reação seja lenta e ocorra por um período longo para que a conversão da reação seja maior. Durante o processo se gera dióxido de carbono que ajuda a arrastar algumas substâncias indesejáveis, como compostos de enxofre (MORADO, 2011).

No final da fermentação a levedura flocula podendo: sedimentar, tratando-se da produção de uma cerveja estilo “Lager”, onde a fermentação ocorre 9°C e 15°C; ou flutuar, tratando-se da produção de uma cerveja estilo “Ale”, onde a fermentação ocorre entre 15°C e 25°C (MORADO, 2011).

Existem quatro fases pela qual a população de leveduras passa durante a fermentação: fase de latência, Figura 2.10 - lag, onde o número de leveduras não se altera, fase exponencial, Figura 2.10 - log, em que ocorre um rápido crescimento populacional das leveduras, a fase estacionária, Figura 2.10 - estacionário, em que a taxa de divisão celular é igual a de mortes, e por fim, a fase de declínio, Figura 2.10 – decline, em que a taxa de morte prevalece (FARBER,2019)..

Na produção cervejeira, tipicamente, mede-se apenas a concentração das leveduras em suspensão, a Figura 2.10 ilustra essa medida juntamente com a temperatura e a concentração de sólidos dissolvidos, por todas as fases da fermentação.

Figura 2.10. Valores típicos de temperatura, concentração de sólidos e leveduras em suspensão durante a fermentação de uma Ale.



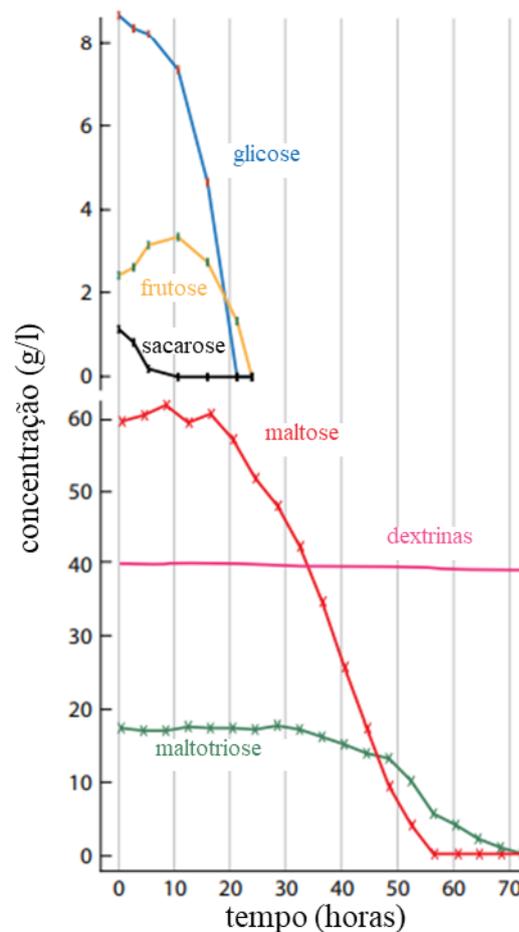
Fonte: FARBER, 2019 (traduzida).

O mosto é normalmente composto por minerais, vitaminas e aminoácidos necessários para as leveduras, porém, o zinco do mosto pode ser insuficiente sendo algumas vezes adicionado ou vendido como aditivo de leveduras comerciais, isso ocorre, pois, parte do zinco é retirado juntamente com o trub ao final do processo de fervura (FARBER,2019).

Além desses compostos, após o resfriamento, Figura 2.2 - resfriamento, a oxigenação do mosto na concentração de 8-15 ppm é necessária para o crescimento populacional das leveduras, no caso de o mosto ter elevada concentração de sólidos pode exigir doses de oxigênio durante o processo de fermentação para um maior crescimento celular. Para atingir esse intervalo de concentração de oxigênio, normalmente, cervejeiros caseiros agitam e respingam o mosto obtendo uma aeração resultante que pode chegar a 3 ppm, abaixo do ideal. Outros cervejeiros utilizam oxigênio puro e pedras difusoras, mas nesse caso, deve-se ter cuidado para não extrapolar a oxigenação. Uma alternativa para o oxigênio puro pode ser a bomba de aquário com a pedra difusora, podendo atingir até 8 ppm (MOSHER, 2015).

Durante a fase de crescimento se inicia a fermentação, convertendo açúcares, que termina na fase estacionária, a Figura 2.11 ilustra as concentrações dos açúcares durante a fermentação (FARBER, 2019).

Figura 2.11. Valores típicos de concentração de carboidratos durante a fermentação.



Fonte: FARBER, 2019 (traduzida).

A quantidade de leveduras adicionadas por volume de cerveja é chamada de taxa de inoculação ou taxa pitch. A dosagem excessiva leva a um menor tempo de latência e fermentação, mas empobrece o corpo da cerveja, levando a uma espuma de baixa qualidade, por outro lado, leveduras insuficientes podem levar a taxas de fermentação lentas e, portanto, taxa de assimilação de açúcares inadequada. Dado isso, a determinação da quantidade de células utilizadas na fermentação é crucial para o resultado da cerveja, existindo várias técnicas para determinar a quantidade de células na pasta de fermento, de forma geral, o padrão industrial é adicionar $1,5 \times 10^6$ células/mL/°P para Lager e $1,0 \times 10^6$ células/mL/°P para Ale. Nesse contexto, deve-se sempre levar em conta somente as células viáveis (FARBER, 2019).

Durante a fermentação ocorre a formação de uma espuma, conhecida como krausen, devido a liberação de dióxido de carbono, por isso é necessário reservar 70% de espaço no

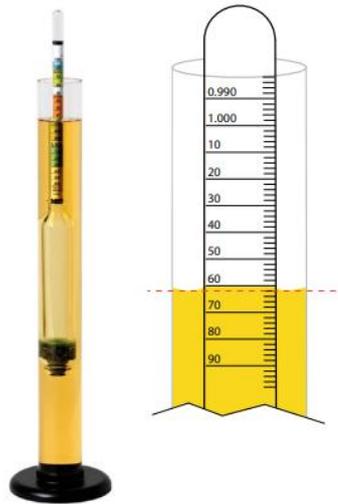
fermentador para essa espuma. Essa espuma pode ser usada para indicar o estágio da fermentação, já que atinge seu volume máximo no pico de atividade de fermentação e colapsa no final do processo deixando um anel conhecido como brandhefe, na parede do fermentador. Esse anel ao cair na cerveja pode gerar um sabor adstringente. Buscando evitar a formação do anel pode ser usado antiespumantes. Alguns desses antiespumantes tem base em óleos vegetais que além de evitar o krausen melhoram a saúde das leveduras exigindo menos oxigenação no mosto, eliminando a exigência do espaço para espuma no fermentador (FARBER, 2019).

Existem diversas maneiras de se medir o progresso da fermentação, um modo é obtendo a gravidade específica ou índice de refração e com isso, realizando relações e simplificações, estime-se o teor de sólidos que é o balizador para a determinação do progresso da fermentação (FARBER, 2019).

Duas medidas importantes são a gravidade específica original e a final (real), OG e FG respectivamente. Uma vez que, amostrada diariamente a gravidade específica, se a mesma permanece inalterada por 3 dias, considera-se então essa a FG (FARBER, 2019).

Para obtenção da densidade o instrumento mais comum é o densímetro flutuante que fornece uma precisão na casa dos milésimos, sendo um processo de medição offline. A Figura 2.12 ilustra uma a obtenção da gravidade específica de 1,062 g/ml utilizando o densímetro flutuante.

Figura 2.12. Densímetro flutuante.

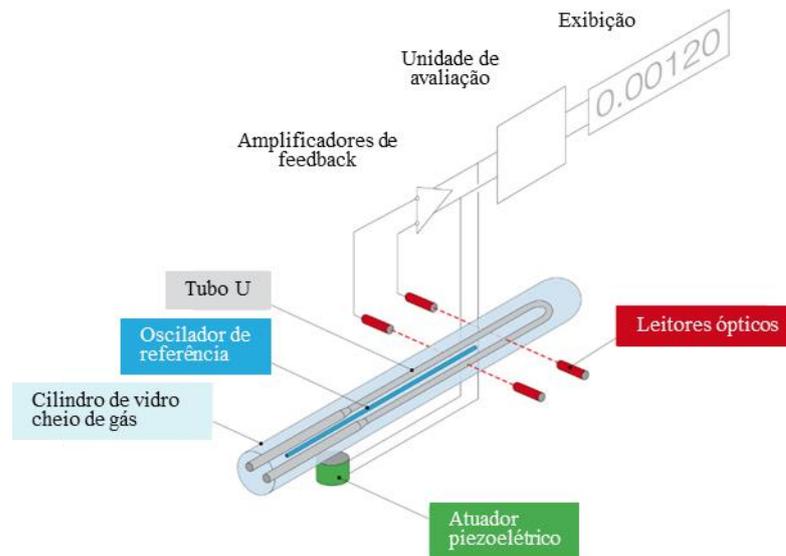


Fonte: FARBER, 2019.

Alternativamente para medição online, pode-se utilizar um densímetro que consiste em um tubo U oscilante. Neste caso o tubo em formato de U é preenchido com o líquido e em seguida vibrado produzindo uma oscilação ao longo do tubo. O período da oscilação chamado de τ é medido e relacionada a gravidade específica pela Equação 2.1 onde A e B são constantes do oscilador, a Figura 2.13 ilustra esse tipo de aparelho. Apesar das medidas de OG e FG serem utilizadas para estimar a concentração de álcool na cerveja, o cálculo ainda é complicado pois nem todo sólido foi convertido em álcool, mas também foi utilizado para o crescimento populacional e na produção de gás carbônico (FARBER, 2019).

$$\rho = A \cdot \tau^2 - B \quad (2.1)$$

Figura 2.13. Tubo em U com atuador piezoelétrico.



Fonte: OSCILLATING, 2020 (traduzida).

O fermentador tipo cilíndrico cônico encamisado é o fermentador mais popular utilizado na indústria e possui diversos acessos e válvulas, entre elas o airlock, que permite a saída do dióxido de carbono produzido e ao mesmo tempo impede a entrada de ar, isso é importante pois o ar pode conter contaminantes que prejudiquem a fermentação. Na produção caseira, normalmente, utiliza-se um recipiente atóxico simples como o da Figura 2.14, nesse caso o airlock também pode servir para identificar o final da reação de fermentação quando o mesmo para de borbulhar.

Figura 2.14. Fermentador típico de produção caseira de cerveja.



Fonte: CERVEJADACASA, 2021

2.1.2.3 Maturação, acabamento e envase

Cabe ressaltar um dos livros utilizados para essa introdução teórica, FARBER de 2019, agrupa a maturação e o acabamento da cerveja, incluindo a carbonatação, em uma etapa chamada de condicionamento, que é dividida em condicionamento quente, onde são reabsorvidos o acetaldeído e diacetil pelas leveduras ao final da fermentação e condicionamento frio que é introduzido nesta seção.

A maturação pode ser considerada o refinamento da cerveja. Ela ocorre após a retirada das leveduras e em temperaturas tipicamente inferiores à da fermentação, podendo ser adicionado aditivos como casca de frutas e até mesmo lúpulo, esse último é uma técnica conhecida como dry hopping. Nessa etapa ocorre também uma carbonatação natural através de resíduos de levedura e extrato sólido que ainda restam após a fermentação, sendo a temperatura e o tempo dessa etapa, importantes para o resultado final (MORADO, 2011).

As temperaturas típicas da fermentação e maturação podem ser divididas em 3 fases: a inoculação, que ocorre mais fria que a fermentação para aclimatar as leveduras; a fermentação primária, que dura de 4 a 10 dias para Ale, e 5 a 14 dias, para Lager; e a maturação que dura de duas a seis semanas. Esses estágios de temperatura são ilustrados na Figura 2.15.

Figura 2.15. Intervalos típicos de temperatura para fermentação e maturação.

Estágio	Ale		Lager	
	Mínimo	Máximo	Mínimo	Máximo
Na inoculação	15 °C (58 °F)	22 °C (72 °F)	6 °C (42 °F)	11 °C (52 °F)
Na fermentação primária	16 °C (60 °F)	25 °C (77 °F)	8 °C (48 °F)	14 °C (57 °F)
Condicionamento frio	-1 °C (30 °F)	6 °C (43 °F)	-1 °C (30 °F)	6 °C (43 °F)

Fonte: FARBER, 2019 (traduzida).

Antes da maturação a cerveja pode ser chamada de cerveja verde. A maturação geralmente ocorre no mesmo tanque da fermentação após a retirada das leveduras. Esse

processo contribui também para prolongar a validade da cerveja, melhorar a sua estabilidade, clarificar e dar brilho. Itens que fazem parte do acabamento da cerveja (FARBER, 2019).

A clarificação ocorre durante a maturação através da sedimentação de partículas sólidas. A velocidade da sedimentação, v , é governada pela Equação 2.2 onde α é a aceleração promovida, via gravidade ou aceleração centrífuga, ρ_p e ρ_b são as densidades da partícula e da cerveja, respectivamente; η é a viscosidade e r o raio da partícula. Uma segunda opção para a clarificação é a utilização de filtros (FARBER, 2019).

$$v = \frac{2r^2(\rho_p - \rho_b)\alpha}{9\eta} \quad (2.2)$$

No caso de sedimentação por gravidade, após a fermentação a cerveja é resfriada até próximo da temperatura de congelamento o que faz as leveduras se aglomerarem e sedimentarem no recipiente, que caso seja cônico facilita sua retirada. A capacidade de floculação da levedura é fundamental para a retirada das leveduras e clarificação da cerveja. Existe a possibilidade do uso de resinas que contribuem para a coagulação das leveduras, como Isinglass, mas que podem não ser aceitos por consumidores veganos (FARBER, 2019).

Em geral, no caso da produção caseira após a fermentação primária ou apenas fermentação, para separar as leveduras e iniciar a maturação se faz o transporte da cerveja para outro recipiente através de um sifão.

A carbonatação pode ser medida em volume de dióxido de carbono a 0°C e 1 atm por litro de cerveja, o que dá 1,96g de dióxido de carbono por litro de cerveja. A carbonatação pode ser realizada de modo forçado através de um cilindro de gás carbônico, ideal para cervejas que passam pelo processo de pasteurização. Já na produção caseira, se utiliza uma solução de açúcares, xarope, ou de malte seco que são adicionados na cerveja acabada. Após isso se engarrafa a cerveja, assim quando esses açúcares são fermentados por leveduras ainda existentes no líquido resultante da maturação, gera-se CO₂ e, conseqüentemente, gaseifica-se a cerveja. Essa técnica é chamada de priming (MOSHER, 2015) ou refermentação (MORADO, 2011).

O envase da cerveja pode ser feito na garrafa, barril ou lata.

2.2 HARDWARE

O hardware do equipamento é composto de um computador de placa única Raspberry Pi, um conjunto de relés mecânicos, uma placa de relés sólidos desenvolvida artesanalmente para o projeto, sensores de temperatura, fontes e conexões.

2.2.1 Raspberry Pi

A Raspberry Pi Foudantion é a empresa que desenvolve o Raspberry Pi e tem o objetivo de promover a educação, principalmente no campo da informática.

O Raspberry Pi é um computador do tamanho de um cartão de crédito que tem a capacidade de executar a maior parte das tarefas de um computador convencional, como: navegar na internet, utilizar monitores e periféricos, criar planilhas, processar texto e linguagens de programação como Python; e ainda traz a capacidade de interagir com componentes externos através de portas GPIO (WHAT, 2021).

Figura 2.16. Raspberry Pi Modelo 3 B+.



Fonte: RASPBERRY, 2021a.

O modelo utilizado no projeto é o Raspberry Pi 3 B+, Figura 2.16, com as seguintes especificações:

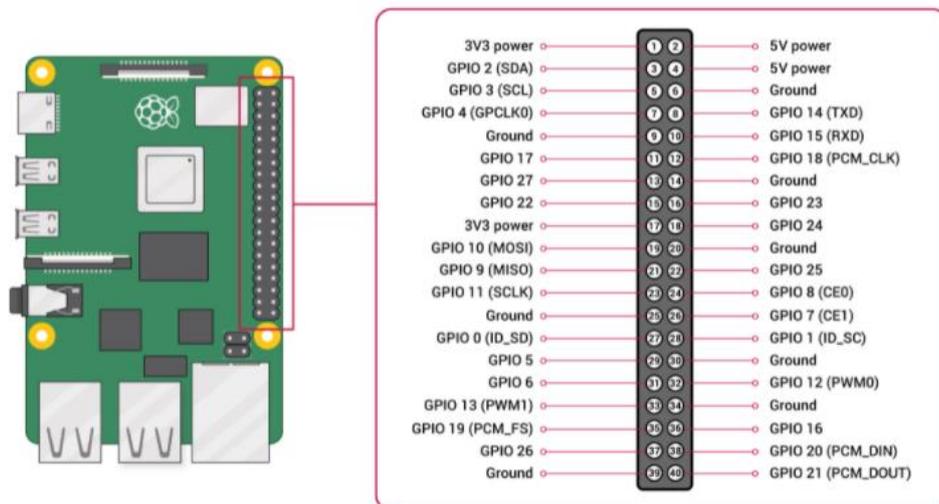
- Broadcom BCM2837B0, Cortex-A53 (ARMv8) SoC de 64 bits a 1,4 GHz
- 1GB LPDDR2 SDRAM
- 2,4 GHz e 5 GHz IEEE 802.11.b / g / n / ac LAN sem fio, Bluetooth 4.2, BLE
- Gigabit Ethernet sobre USB 2.0 (taxa de transferência máxima de 300 Mbps)
- Cabeçalho GPIO estendido de 40 pinos

- HDMI de tamanho normal
- 4 portas USB 2.0
- Porta de câmera CSI para conectar uma câmera Raspberry Pi
- Porta de tela DSI para conectar uma tela de toque Raspberry Pi
- Saída estéreo de 4 pólos e porta de vídeo composto
- Porta Micro SD para carregar seu sistema operacional e armazenar dados
- Entrada de alimentação 5V / 2,5A DC

O sistema operacional instalado no Raspberry é o Raspbian, que é uma variante do Debian, que já vem com uma série de ferramentas (RASPBerry, 2021a).

O diagrama de pinos do Raspberry Pi é dado na Figura 2.17, contendo um total de 40 pinos. Entre esses pinos existem as portas tipo GPIO (General Purpose Input/Output) que servem para receber e enviar sinais elétricos, podendo executar uma variedade de funções, inclusive a leitura de sensores (RASPBerry, 2021b).

Figura 2.17. Diagrama de pinos do Raspberry Pi Modelo 3 B+.



Fonte: RASPBerry, 2021b.

Para o propósito desse trabalho não será apresentado como instalar o sistema operacional, nem os passos para o uso do Raspberri Pi. Vários tutoriais podem ser

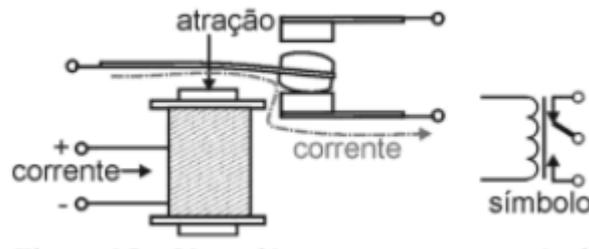
encontrados na internet, inclusive no próprio site do Raspberry Pi Foundation: <https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started>.

2.2.2 Relés mecânicos e sólidos

Relé mecânico é um comutador eletromecânico na qual uma bobina, quando percorrida por uma corrente elétrica, cria um campo magnético que atua sobre uma haste fazendo o contato móvel encostar no contato fixo, fechando o circuito. O mesmo pode ocorrer de forma inversa fazendo o circuito controlado ser aberto com a energização da bobina, no primeiro caso temos um relé normal aberto, NA, e no segundo relé normal fechado, NF. Ainda existe a possibilidade de o relé combinar as duas ações, sendo um NA/NF, a Figura 2.18 ilustra um relé desse tipo (BRAGA, 2012).

A capacidade de corrente que um relé pode suportar depende da sua construção e de seu dimensionamento. O tempo de operação, ou seja, tempo para fechar ou abrir contatos está na casa dos milissegundos sendo a vida útil entre 250 mil a 30 milhões de operações, dependendo da intensidade da corrente controlada (BRAGA, 2012).

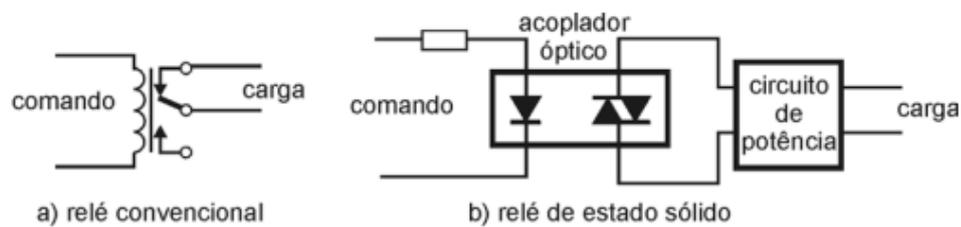
Figura 2.18. Relé mecânico tipo NA/NF.



Fonte: BRAGA, 2012.

Já os relés de estado sólido têm as mesmas funções do relé mecânico, comutar circuitos de potência a partir de sinais elétricos, porém o seu acionamento depende de um acoplador óptico onde um emissor de luz consegue acionar um dispositivo sensível à luz. A Figura 2.19, ilustra os dois tipos de relé, convencional e sólido.

Figura 2.19. Relé convencional (a) e relé de estado sólido (b).

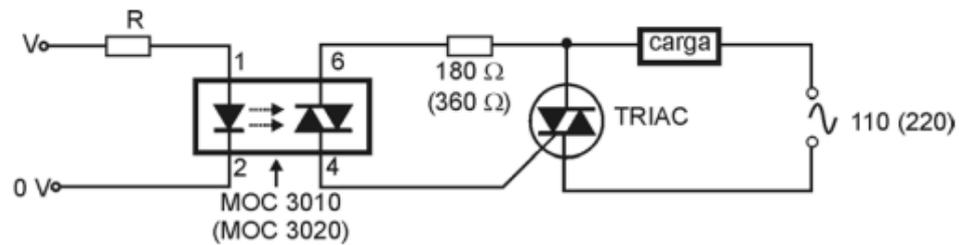


Fonte: BRAGA, 2012.

Os relés de estado sólido apresentam uma velocidade de comutação muito superior aos mecânicos, o que permite controlar até mesmo o disparo entre a onda senoidal da corrente alternada que chega da rede, porém, se trata de um sistema mais sensível e que há aquecimento quando utilizado para controle de altas correntes (BRAGA, 2012).

O opta-acoplador ou acoplador óptico pode ser ainda combinado com um TRIAC que é um dispositivo que comuta circuitos de corrente alternada mais intensas a partir de um sinal elétrico fornecido. Nesse caso, o opta-acoplador serve para acionar o TRIAC e também para isolar o sistema de alta potência dos demais componentes já que seu acionamento é óptico (BRAGA, 2012).

Figura 2.20. Esquema de um opta-acoplador em conjunto com o TRIAC.



Fonte: BRAGA, 2012.

2.2.3 Sensor de temperatura

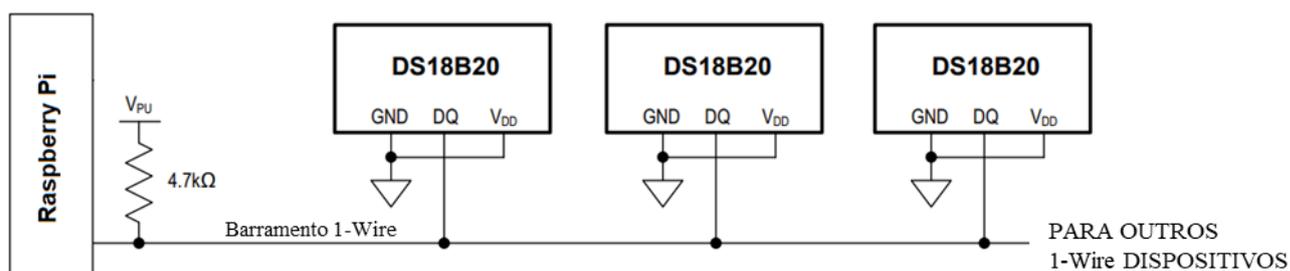
Um dos sensores de temperatura existente no mercado e de fácil aquisição é o DS18B20, ele pode ser encontrado encapsulado em um pequeno tubo podendo ser utilizado em contato com líquidos.

O DS18B20 é capaz de fornecer medidas entre -55°C e $+125^{\circ}\text{C}$, com uma acurácia de $\pm 0.5^{\circ}\text{C}$, e resolução de até $0,0625^{\circ}\text{C}$ (MAXIM INTEGRATED, 2019).

Esse sensor possui a tecnologia 1-Wire que permite o envio de informação sobre as medidas de temperatura utilizando apenas um fio na qual a comunicação pode prover 9-bit a 12-bit de informação.

Essa tecnologia permite que vários sensores DS18B20 possam ser operados utilizando apenas uma porta GPIO do Raspberry, podendo ainda ser conectado no modo parasita, em que o sensor utiliza apenas dois tipos conexões, o terra e o data (MAXIM INTEGRATED, 2019).

Figura 2.21. Formas de onda de uma corrente alternada controlada.



Fonte: Adaptado de MAXIM INTEGRATED, 2019 (traduzida).

2.3 NODE-RED

O Node-RED é uma ferramenta de programação utilizada para conectar Hardware, API's e serviços online. O seu editor é utilizado através de um navegador de internet e sua estrutura é composta de nós e fluxos. O Node-RED é executado com base no Node.js, que utiliza Javascript, sua lógica de programação é orientada a eventos e sem bloqueios (NODE-RED, 2021a).

A programação é estilo visual, onde os nós tem uma finalidade bem definida, podendo receber alguns dados, executar algo com eles e passar uma resposta adiante. Dessa forma, é possível criar um programa sem precisar entender linhas de códigos individuais (NODE-RED, 2021b).

O objeto que é inserido ou passado por um nó é chamado de mensagem. Dentro do nó a mensagem recebida pode ser acessada usando a sigla "msg". Além disso, esse objeto tem alguns parâmetros padrão na maioria dos casos, como o "msg.payload" que seria o conteúdo da mensagem e o "msg.topic" que seria um rótulo para a mensagem (WORKING, 2021b).

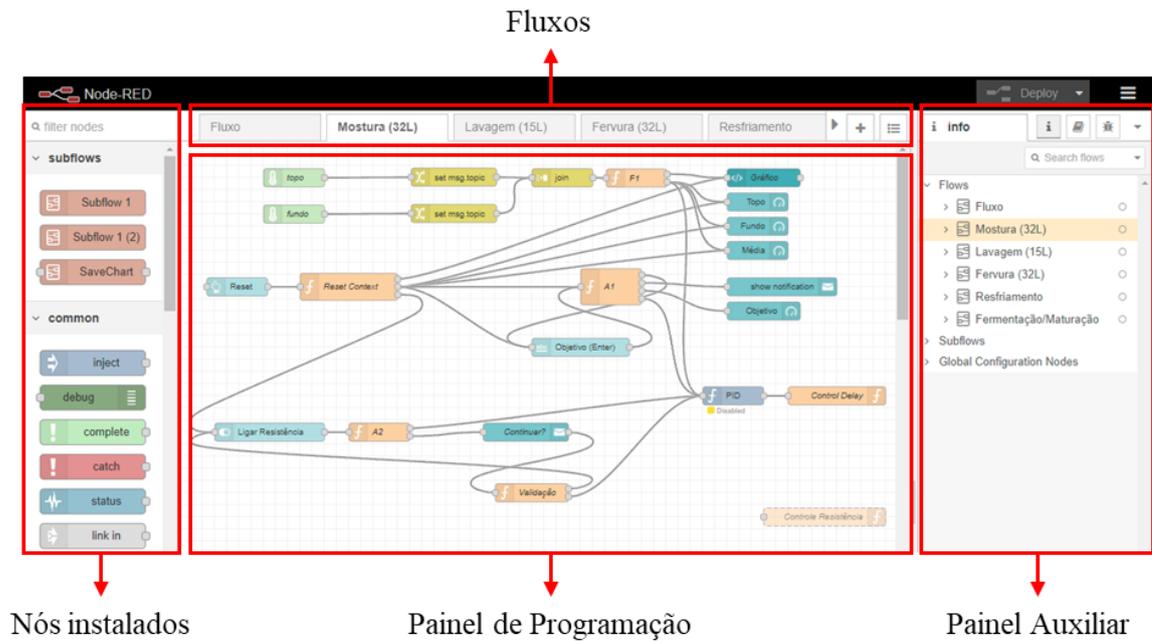
As variáveis definidas durante a programação podem ser definidas em três contextos diferentes (WORKING, 2021a):

- Nó: visível somente no nó em que foi definida.
- Fluxo: visível em todos os nós do fluxo em que foi definida.
- Global: visível em todos os nós.

Os nós e fluxos podem ser desenvolvidos e disponibilizados pela comunidade na forma de biblioteca de nós.

A Figura 2.22 ilustra a interface de programação do Node-RED, os fluxos são exibidos na parte central superior em forma de guias, os nós disponíveis para uso na esquerda e na direita o painel auxiliar, que fornece informações, debugs e outros itens.

Figura 2.22. Interface de programação Node-RED.



Fonte: Produção própria, 2021.

Uma das principais bibliotecas de nós do Node-RED é a node-red-dashboard, que possui uma série de nós para a criação de painéis de dados e operações.

Figura 2.23. Exemplo de interface do painel Node-RED.



Fonte: Produção própria, 2021.

3 A CENTRAL DE CONTROLE

A central de controle funciona em conjunto com o software desenvolvido. Uma visão geral externa e interna são mostradas nas Figuras 3.1 e 3.2.

Figura 3.1. Central de controle visão geral externa.

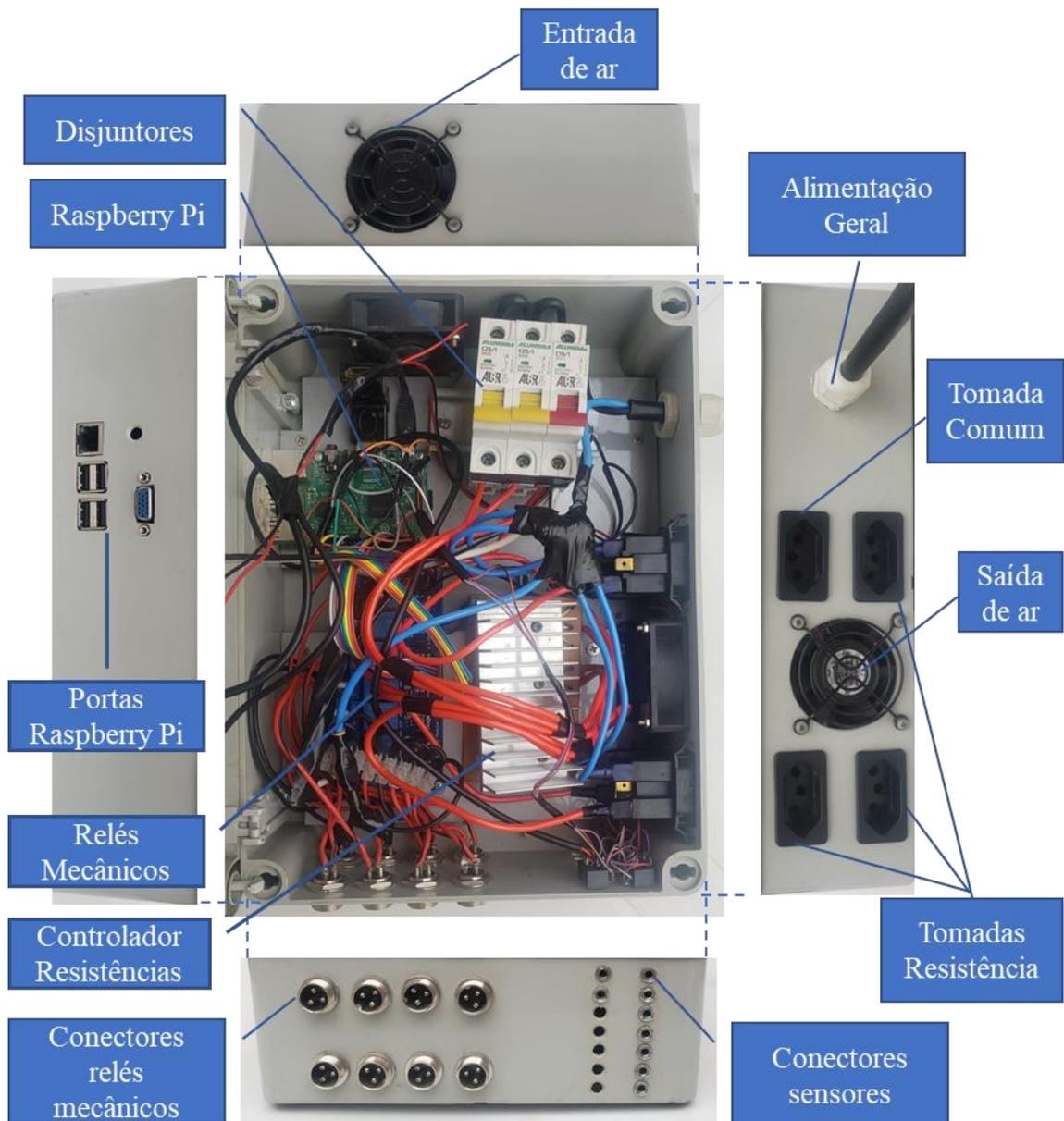


Figura 3.2. Central de controle visão geral interna.



A central é composta de alguns componentes eletrônicos e conectores que são mostrados nas Figura 3.3 e Figura 3.4:

Figura 3.3. Central de Controle visão interna.



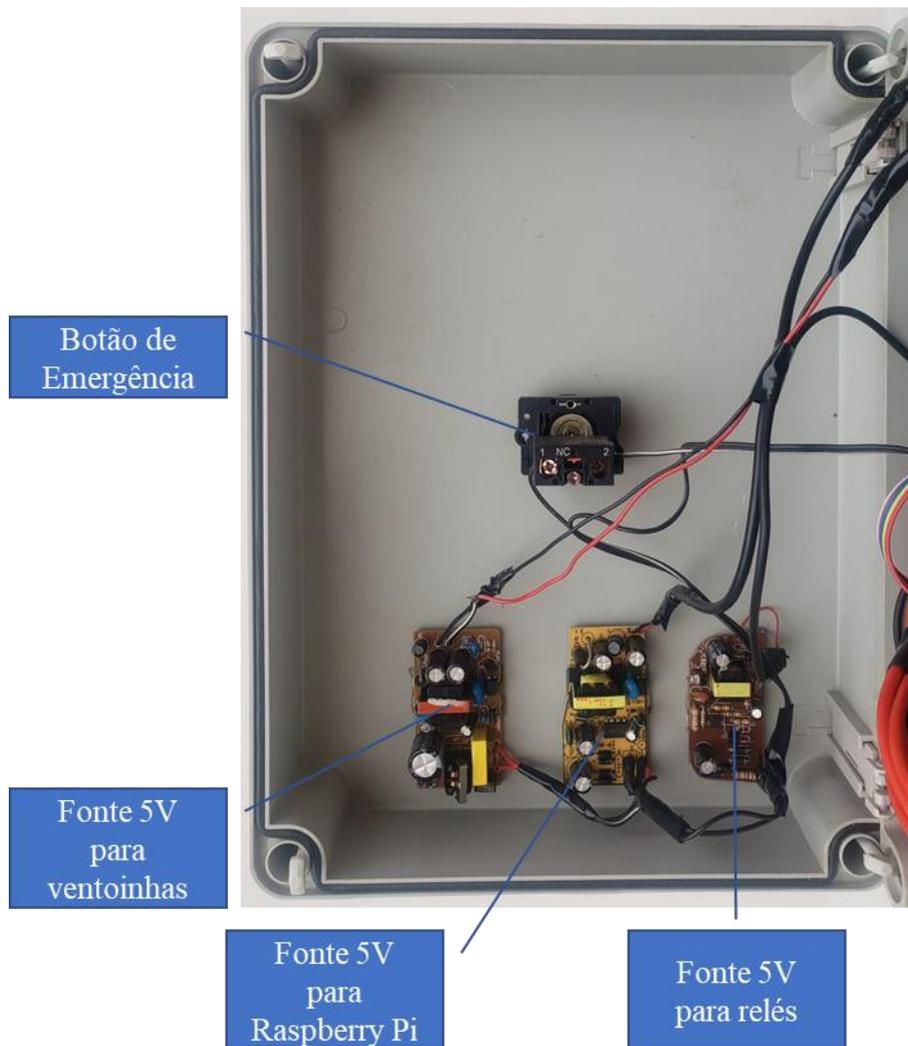
Internamente na caixa de controle são dispostos:

- Raspberry Pi: realiza toda a lógica da programação gerando a interface usuário máquina.
- Relés Mecânicos: realiza o controle liga/desliga de válvulas, bombas e do refrigerador.
- Controlador de Resistência: desenvolvida artesanalmente para o projeto, o componente é responsável pelo controle da potência fornecida às resistências através de relés de estado sólido.
- Disjuntores: para segurança no caso de uma sobrecarga no sistema.

- Saída de ar e Entrada de ar: possuem duas ventoinhas que criam uma corrente de ar par resfriamento dos componentes, principalmente do Controlador de Resistências.
- Conectores Relés Mecânicos: são conectores tipo Mike 3 pinos, que realizam a interface relé mecânico e componente controlado.
- Conectores Sensores: são conectores tipo P2, que foi a solução encontrada na quando se confeccionou da caixa de controle. Esse tipo de conector fornece 3 vias que são suficientes para sensores de temperatura. Cabe ressaltar que um sensor só pode ser conectado ou desconectado quando o sistema está desligado para evitar a queima das portas GPIO do Raspberry Pi.
- Tomadas Resistência: tomada de três vias para a conexão das resistências com a placa controladora. É uma boa prática de operação apenas conectar as resistências quando as mesmas estiverem submersas, mesmo que o controle não esteja ligado.
- Portas Raspberry Pi: utilizadas para conectar periféricos como mouse, teclado, monitor, adaptador, entre outros periféricos comuns a computadores.
- Tomada Comum: utilizada algumas vezes para ligar o monitor.
- Alimentação Geral: alimenta todo o sistema, incluindo as resistências. Cabe ressaltar que para operação é recomendado que a rede elétrica tenha a capacidade de fornecer no mínimo 30 A de corrente e que se evite de ligar mais de uma resistência ao mesmo tempo.

Ainda na parte interna, tem-se na tampa as fontes de alimentação de corrente contínua e o botão de emergência que serve para ligar e desligar essas fontes, Figura 3.4:

Figura 3.4. Central de Controle visão interna tampa.

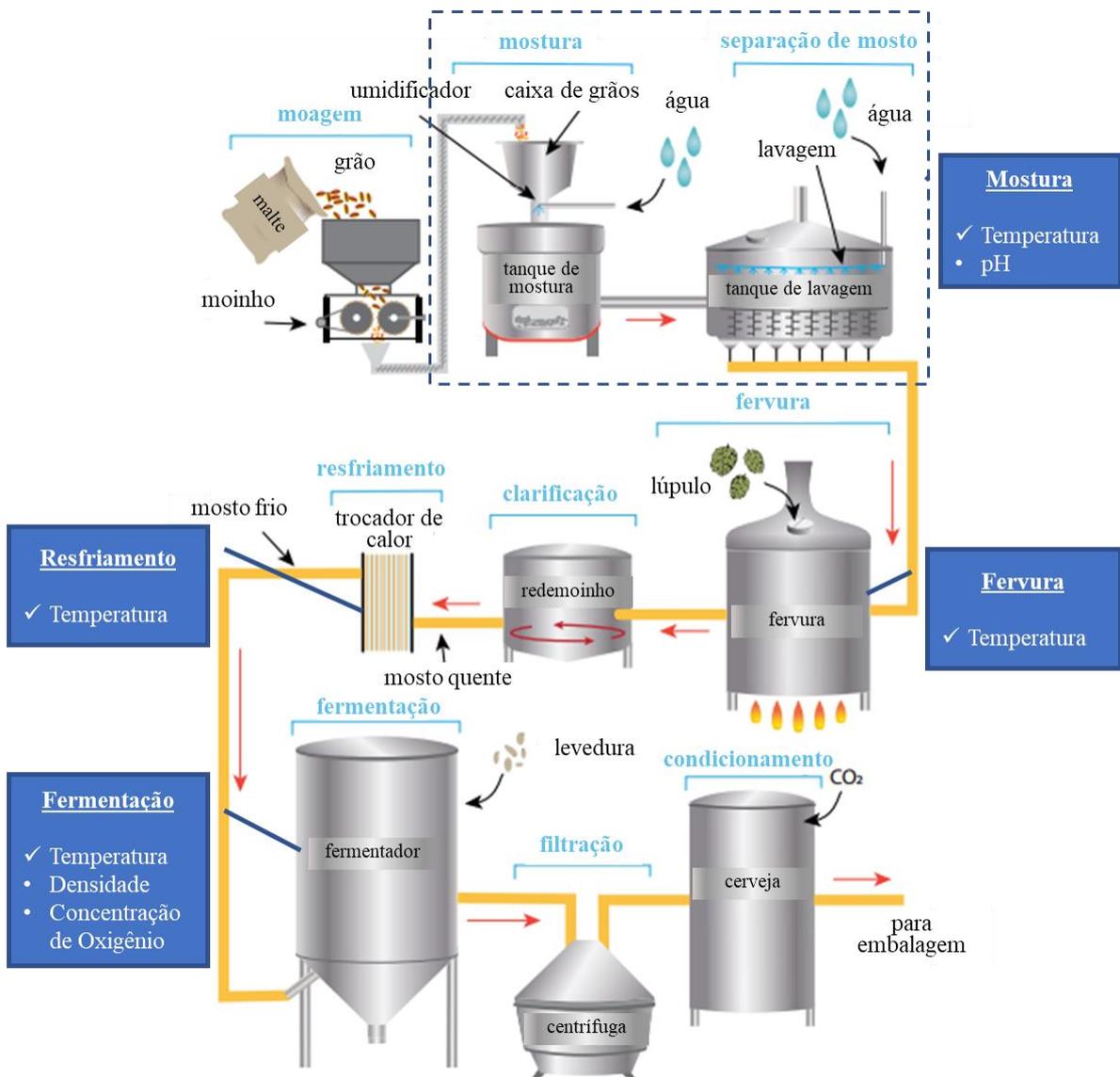


4 O SOFTWARE

4.1 VISÃO GERAL

Na interface de programação do software, criou-se 5 fluxos um para cada operação correspondente. Na figura 4.1 se associa as variáveis mais importantes do processo de produção com um controle, ainda em fase de desenvolvimento, para uma planta cervejeira. Lembrando que no presente trabalho foi abordado apenas o controle de temperatura.

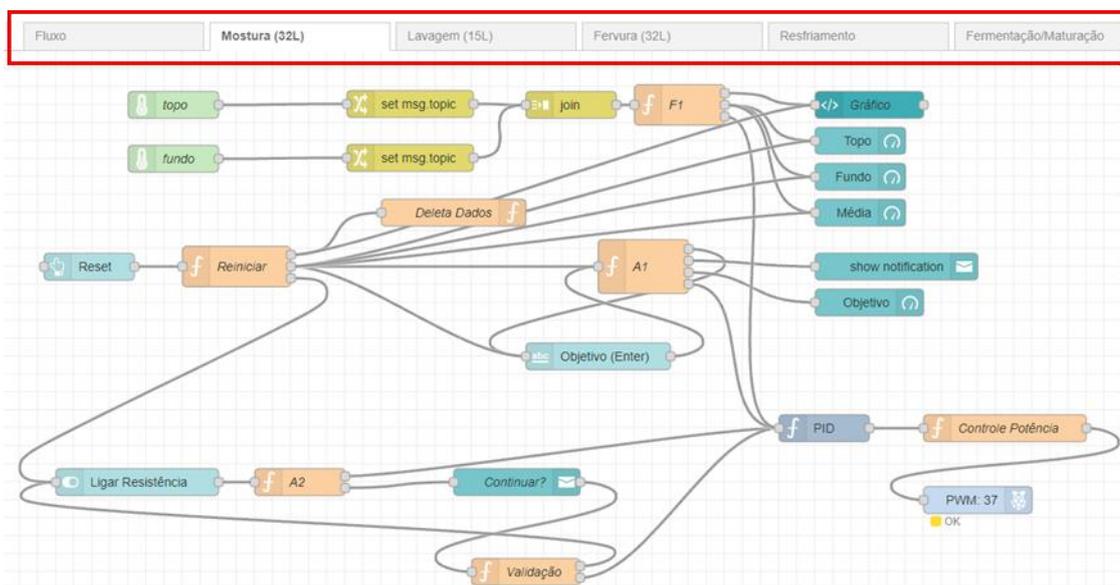
Figura 4.1. Variáveis importantes e controle criado para CAMA.



Fonte: FARBER, 2019 (traduzida e adaptada).

O fluxo “Fluxo” lida com o controle de válvulas e bombas, “Mostura (32L)” com a operação de mosturação e o tanque de mostura, “Lavagem (15L)” com o tanque de lavagem, “Fervura (32L)” com o tanque e o processo de fervura, o “Resfriamento” com o trocador de calor e a “Fermentação/Maturação” com o respectivo processo de fermentação e maturação.

Figura 4.2. Fluxos criados para controle da CAMA.



Na interface de controle, criou-se 5 blocos de controle cada qual com seus itens correspondendo ao fluxo da interface de programação.

Foi ainda necessário o uso de uma biblioteca externa ao Node-RED, file-system (FS, 2021). Essa biblioteca é utilizada para manipulação de arquivos no Raspberry Pi. Para utiliza-la é preciso seguir o procedimento de alteração das configurações do Node-RED (EXAMPLE, 2021).

Figura 4.3. Blocos de controle criados para operação da C.A.M.A.

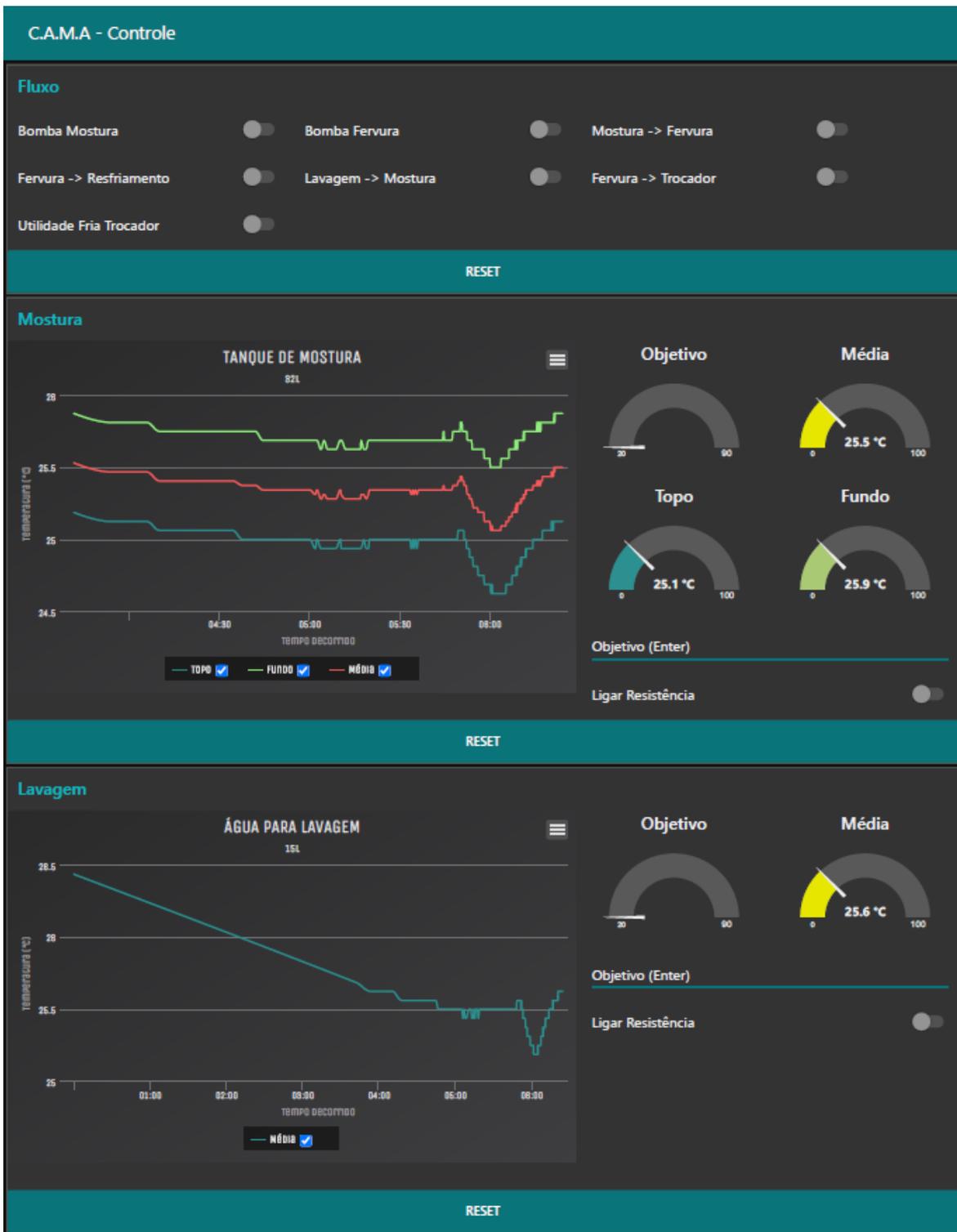
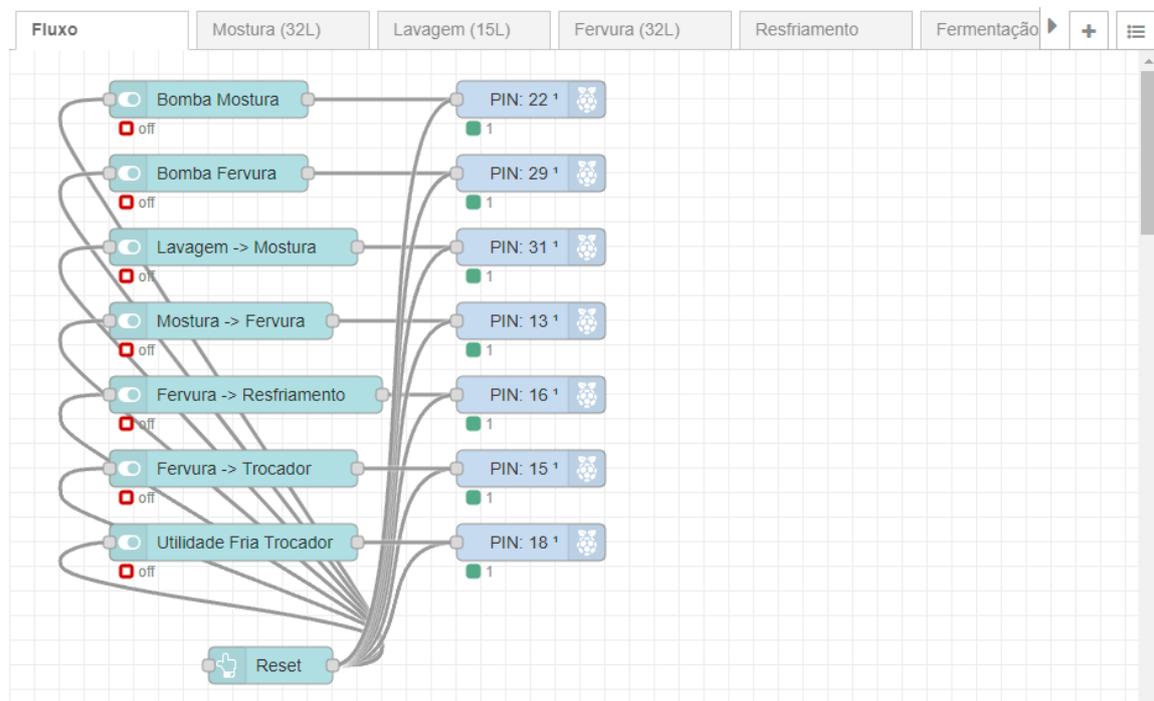


Figura 4.3. Blocos de controle criados para operação da CAMA (continuação).



4.2 FLUXO

Figura 4.4. Programação do Fluxo.



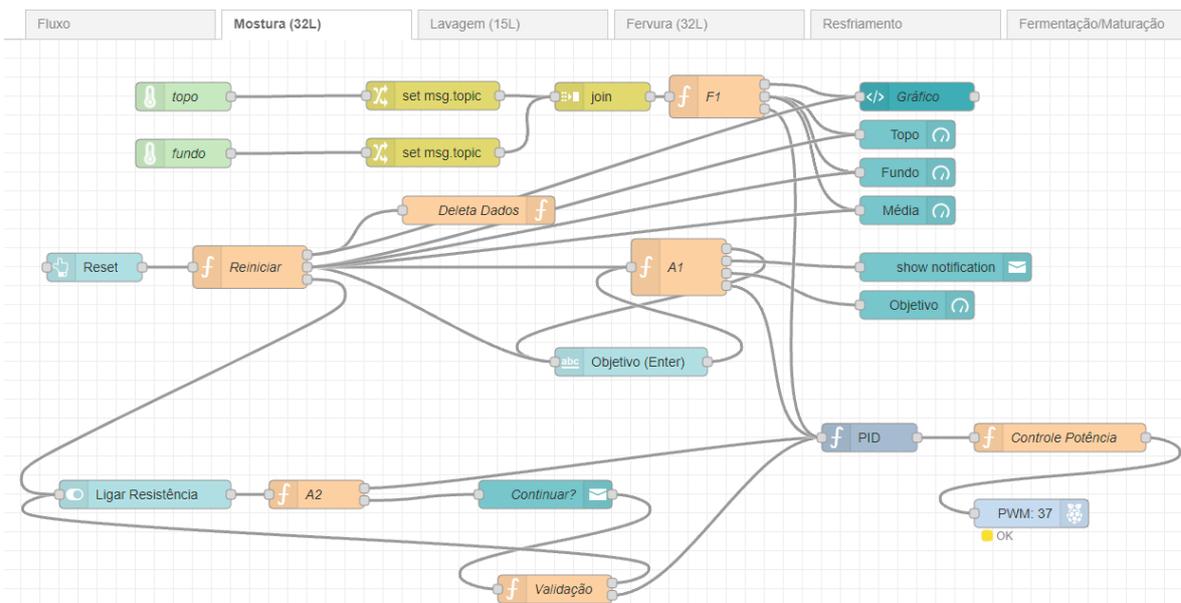
Esse fluxo tem o objetivo de controlar a direção das correntes manipulando válvulas e bombas, porém, ainda não foi implementado controladores de válvulas na planta, estando a programação e a interface de controle preparada para a implantação. Atualmente, o controle é manual necessitando pleno conhecimento do processo por parte do operador para evitar erros operacionais, como: transbordo dos tanques, queima das resistências por não estarem submersa e queima das bombas. Os controles do tipo On/Off, “Bomba Mostura” e “Bomba Fervura”, acionam relé mecânicos fornecendo energia para as bombas que empurram o líquido pela parte superior dos tanques de Mostura e Fervura respectivamente. No caso da operação de resfriamento, o mosto trafega pelo trocador de placas por ação da gravidade e a utilidade fria pela pressão da rede de fornecimento.

Figura 4.5. Painel de operação do Fluxo.



4.3 MOSTURA

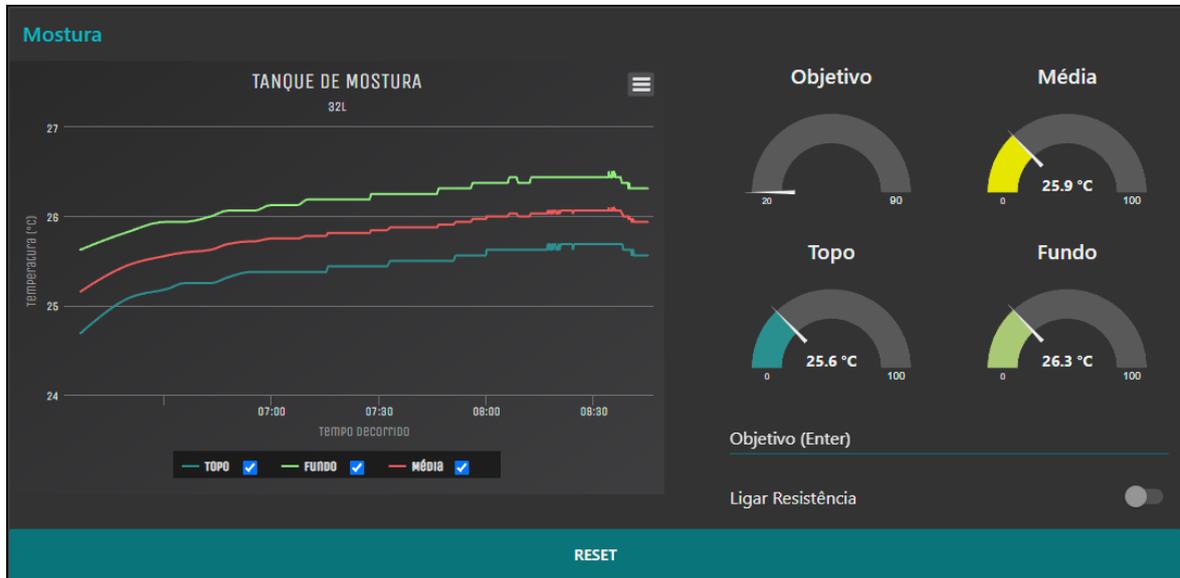
Figura 4.6. Programação da Mostura.



O controle do tanque de mostura utiliza a temperatura média de dois sensores diferentes, um que está localizado acima da cama de grãos e outro logo abaixo do fundo falso. O fluxo começa obtendo a temperatura dos dois sensores nos nós “topo” e “fundo” que passa essa informação através da propriedade `msg.payload`. Posteriormente, atribui-se o parâmetro `msg.topic` como “fundo” ou “topo” nos nós “set msg.topic”. Essas mensagens com os parâmetros definidos são inseridas no nó “join” que aguarda a chegada de duas msg com `msg.topic` diferentes ou 3 segundos, esse tratamento é necessário uma vez que os sensores obtém medidas em tempos ligeiramente diferentes.

Com essas informações de temperatura, executa-se o nó “F1” que tem o objetivo de preparar o msg.payload que será exibido no gráfico e nos medidores do painel de controle.

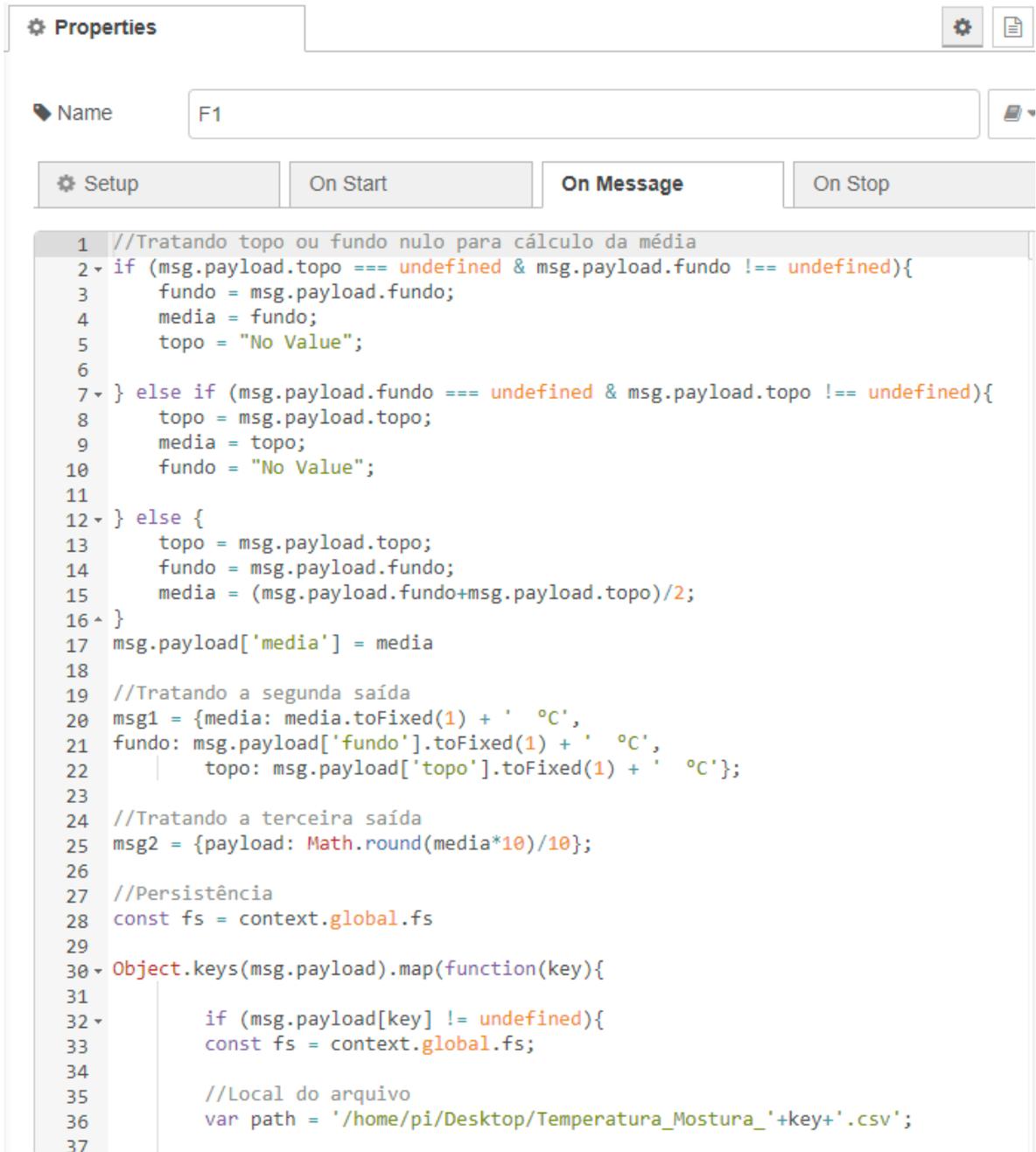
Figura 4.7. Painel de operação da Mostura.



Para isso esse nó realiza algumas funções:

- Cálculo da média das temperaturas, de forma que se alguma das temperaturas vier nula, a média será apenas o valor da temperatura que não foi nula. Isso para no caso de haver um impedimento de algum sensor, como por exemplo, um problema de conexão ou um desencontro no recebimento das informações dos sensores com mais de 3 segundos, que é o limite máximo do nó “join”. Esse tratamento é importante já que a média das temperaturas é utilizada para controle de temperatura no nó “PID”.
- Cálculo do tempo decorrido desde o primeiro ponto do gráfico.
- A persistência das informações de temperatura e tempo, gerando 3 arquivos, um para cada medida: fundo, topo e média. Os arquivos são do tipo “csv”, criados na área de trabalho do Raspberry Pi. Desse modo caso haja um desligamento acidental do sistema, as informações são preservadas. Como já mencionado no item 4.1, se utiliza ainda uma biblioteca externa, chamada de “File System” para manipular arquivos no Raspberry Pi. Como mostra a Figura 4.8.

Figura 4.8. Programação função F1.



```
1 //Tratando topo ou fundo nulo para cálculo da média
2 if (msg.payload.topo === undefined & msg.payload.fundo !== undefined){
3     fundo = msg.payload.fundo;
4     media = fundo;
5     topo = "No Value";
6
7 } else if (msg.payload.fundo === undefined & msg.payload.topo !== undefined){
8     topo = msg.payload.topo;
9     media = topo;
10    fundo = "No Value";
11
12 } else {
13     topo = msg.payload.topo;
14     fundo = msg.payload.fundo;
15     media = (msg.payload.fundo+msg.payload.topo)/2;
16 }
17 msg.payload['media'] = media
18
19 //Tratando a segunda saída
20 msg1 = {media: media.toFixed(1) + ' °C',
21 fundo: msg.payload['fundo'].toFixed(1) + ' °C',
22 | topo: msg.payload['topo'].toFixed(1) + ' °C'};
23
24 //Tratando a terceira saída
25 msg2 = {payload: Math.round(media*10)/10};
26
27 //Persistência
28 const fs = context.global.fs
29
30 Object.keys(msg.payload).map(function(key){
31
32     if (msg.payload[key] !== undefined){
33         const fs = context.global.fs;
34
35         //Local do arquivo
36         var path = '/home/pi/Desktop/Temperatura_Mostura_'+key+'.csv';
37
```

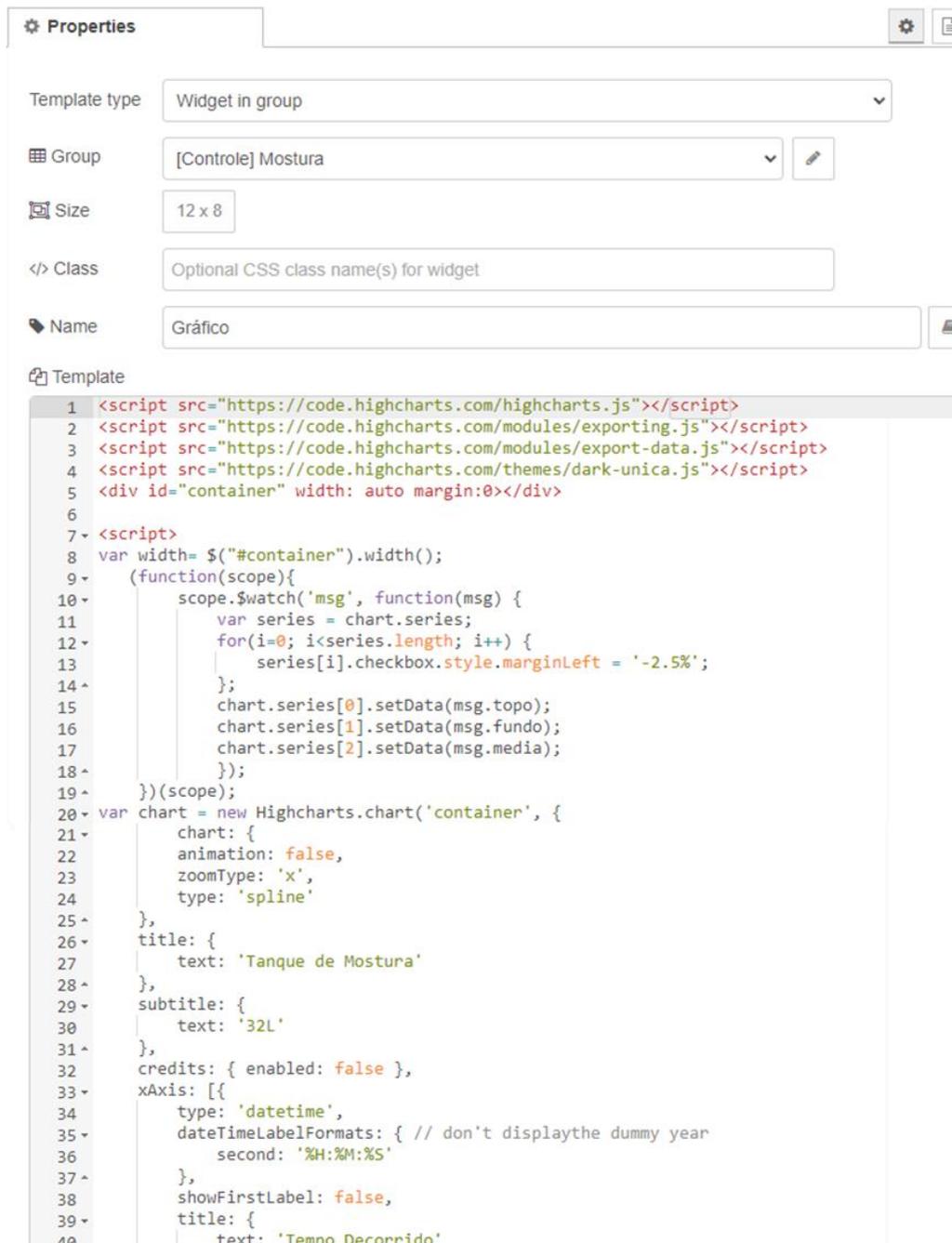
Figura 4.8. Programação função F1 (continuação).

```
38 //Cria um arquivo em branco caso não exista
39 if (!fs.existsSync(path)) {
40     fs.appendFileSync(path, Date.now()+", "+msg.payload[key])
41 }
42 else{
43     //Insere dados se o arquivo existe
44     fs.appendFileSync(path, "\n"+Date.now()+", "+msg.payload[key]);
45     //Lê o arquivo
46     data = fs.readFileSync(path, 'utf-8');
47     //Convert CSV em Array de Números
48     rows = data.split("\n");
49
50     msg[key] = rows.map(function (row, index)
51     {
52         //Se é a primeira medida
53         //guarda o primeiro valor
54         if (index == 0)
55         {
56             return row.split(",").map(function(item, index, arr)
57             {
58                 if (index === 0){
59                     first = Number(item);
60                     item = 0;
61                 }
62                 else{
63                     item = Number(item);
64                 }
65             }
66             return item
67         }
68     }
69     )
70     //não é a primeira medida subtrai o primeiro valor
71     //para obter a duração
72     else{
73         return row.split(",").map(function(item, index, arr){
74             if (index === 0){
75                 item = Number(item) - first;
76             }
77             else{item = Number(item)}
78             return item;
79         }
80     }
81     }
82     }
83     }
84     )
85     }
86     }
87     }
88 )
89
90 return [msg, msg1, msg2];
91
```

Enabled

Os nós “Fundo”, “Topo” e “Média” exibem os medidores de temperatura. Já o nó “Gráfico” exibe um gráfico que utiliza uma biblioteca multiplataforma chamada HighCharts (HIGHCHARTS, 2021). Essa biblioteca foi escolhida devido a gama de opções de controle, capacidade de exibição da biblioteca e a possibilidade de realizar download dos gráficos. Sua programação é realizada em Javascript com HTML. Sua licença foi obtida apenas para uso nesse trabalho, sem fins comerciais, e está no apêndice 8.2.

Figura 4.9. Programação do gráfico.



The image shows a screenshot of an IDE's Properties and Template sections for a widget named "Gráfico".

Properties:

- Template type: Widget in group
- Group: [Controle] Mostura
- Size: 12 x 8
- Class: Optional CSS class name(s) for widget
- Name: Gráfico

Template:

```
1 <script src="https://code.highcharts.com/highcharts.js"></script>
2 <script src="https://code.highcharts.com/modules/exporting.js"></script>
3 <script src="https://code.highcharts.com/modules/export-data.js"></script>
4 <script src="https://code.highcharts.com/themes/dark-unica.js"></script>
5 <div id="container" width: auto margin:0></div>
6
7 <script>
8 var width= $("#container").width();
9 (function(scope){
10     scope.$watch('msg', function(msg) {
11         var series = chart.series;
12         for(i=0; i<series.length; i++) {
13             series[i].checkbox.style.marginLeft = '-2.5%';
14         };
15         chart.series[0].setData(msg.topo);
16         chart.series[1].setData(msg.fundo);
17         chart.series[2].setData(msg.media);
18     });
19 })(scope);
20 var chart = new Highcharts.chart('container', {
21     chart: {
22         animation: false,
23         zoomType: 'x',
24         type: 'spline'
25     },
26     title: {
27         text: 'Tanque de Mostura'
28     },
29     subtitle: {
30         text: '32L'
31     },
32     credits: { enabled: false },
33     xAxis: [{
34         type: 'datetime',
35         dateTimeLabelFormats: { // don't display the dummy year
36             second: '%H:%M:%S'
37         },
38         showFirstLabel: false,
39         title: {
40             text: 'Tempo Decorrido'
```

Figura 4.9. Programação do gráfico (continuação).

```
41 ^    }
42 ^  }],
43 ^  yAxis: {
44 ^    title: {
45 ^      text: 'Temperatura (°C)'
46 ^    },
47 ^  },
48 ^  tooltip: {
49 ^    headerFormat: '<b>{point.x:Tempo decorrido: %H:%M:%S}</b></br>',
50 ^    pointFormat: '{series.name}:<b style="text-align:center;">{point.y} °C</b></br>'
51 ^    valueDecimals:3,
52 ^    shared: true,
53 ^    useHTML: true
54 ^  },
55 ^  plotOptions: {
56 ^    spline: {
57 ^      marker: {
58 ^        enabled: false
59 ^      }
60 ^    },
61 ^    series: {
62 ^      turboThreshold: '0',
63 ^      showCheckbox: true,
64 ^      events: {
65 ^        checkboxClick: function (event) {
66 ^          if (event.checked) {
67 ^            this.show();
68 ^          } else {
69 ^            this.hide();
70 ^          }
71 ^        },
72 ^        legendItemClick: function() {
73 ^          return false;
74 ^        }
75 ^      }
76 ^    }
77 ^  },
78 ^  series: [{name:'Topo ',selected: true},{name:'Fundo', selected: true},{name:'Média',
79 ^    }]);
80 ^ </script>
```

Pass through messages from input.
 Add output messages to stored state.
 Reload last value on refresh.

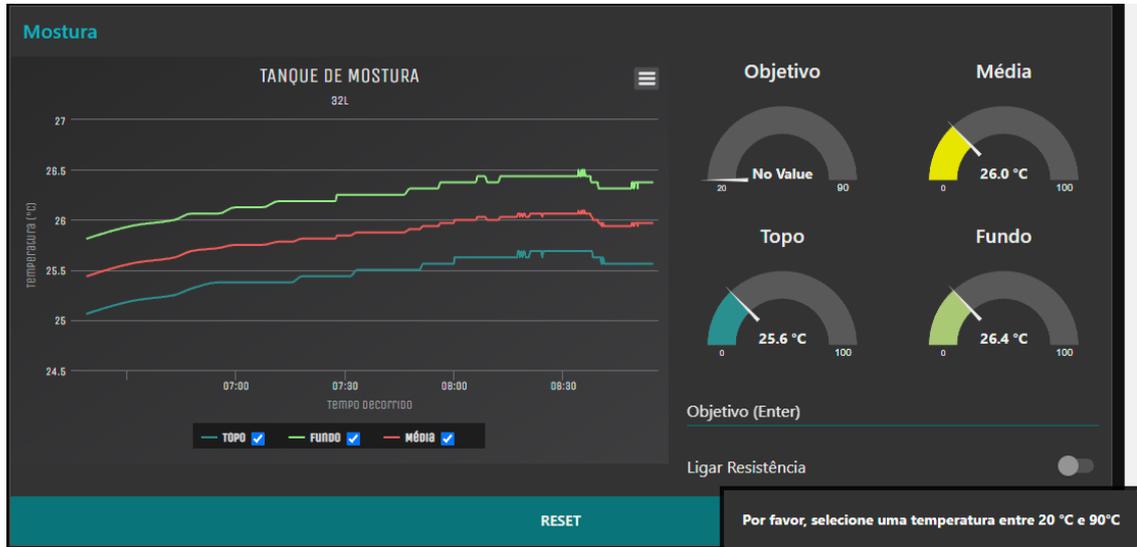
Enabled

O nó “Objetivo (Enter)” serve para configurar o setpoint do controlador de temperatura, essa entrada ocorre no painel de controle através de um campo de texto numérico e passa por um tratamento no nó “A1” que realiza os seguintes procedimentos:

- Quando o objetivo de temperatura inserido não está no intervalo válido, no caso entre 20°C e 90°C: manda uma notificação que será exibido no painel de controle através do nó “show notification” informando para selecionar uma temperatura válida, indica

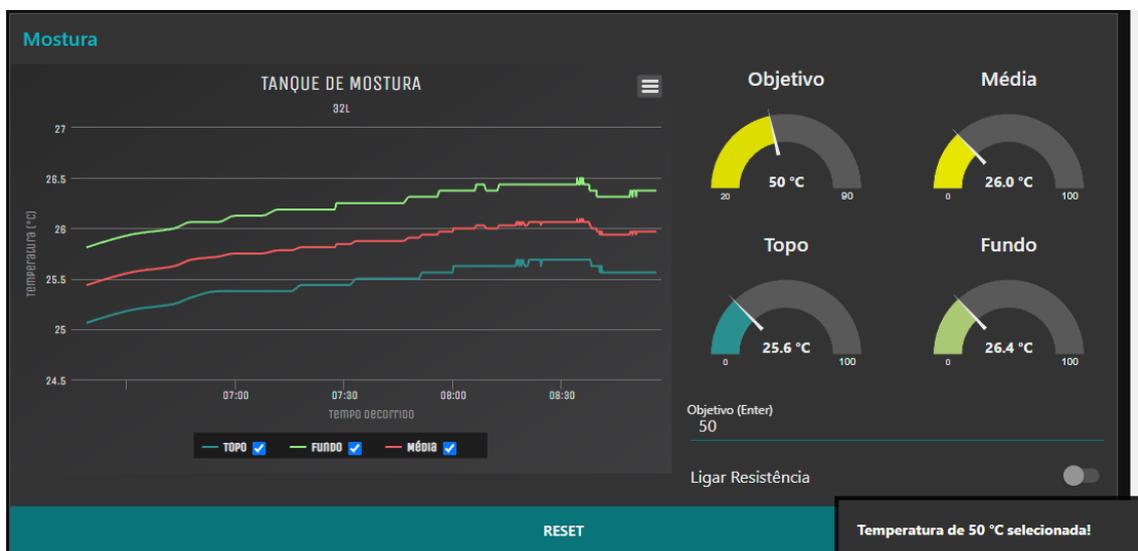
“No Value” no medidor “Objetivo”, limpa o controle “Objetivo (Enter)” e configura para 0 o setpoint do PID.

Figura 4.10. Notificação de temperatura inserida inválida.



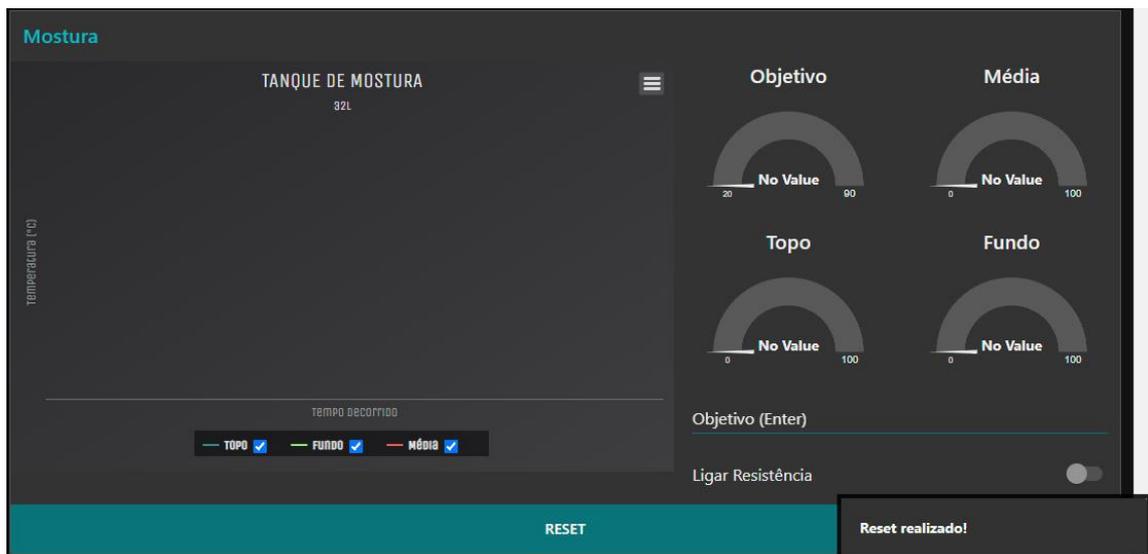
- Quando o objetivo de temperatura inserido está no intervalo válido: passa essa entrada para o valor para o setpoint do nó “PID” e para o medidor “Objetivo”, exibe uma notificação através do “show notification” informando que a temperatura foi selecionada.

Figura 4.11. Notificação de temperatura selecionada.



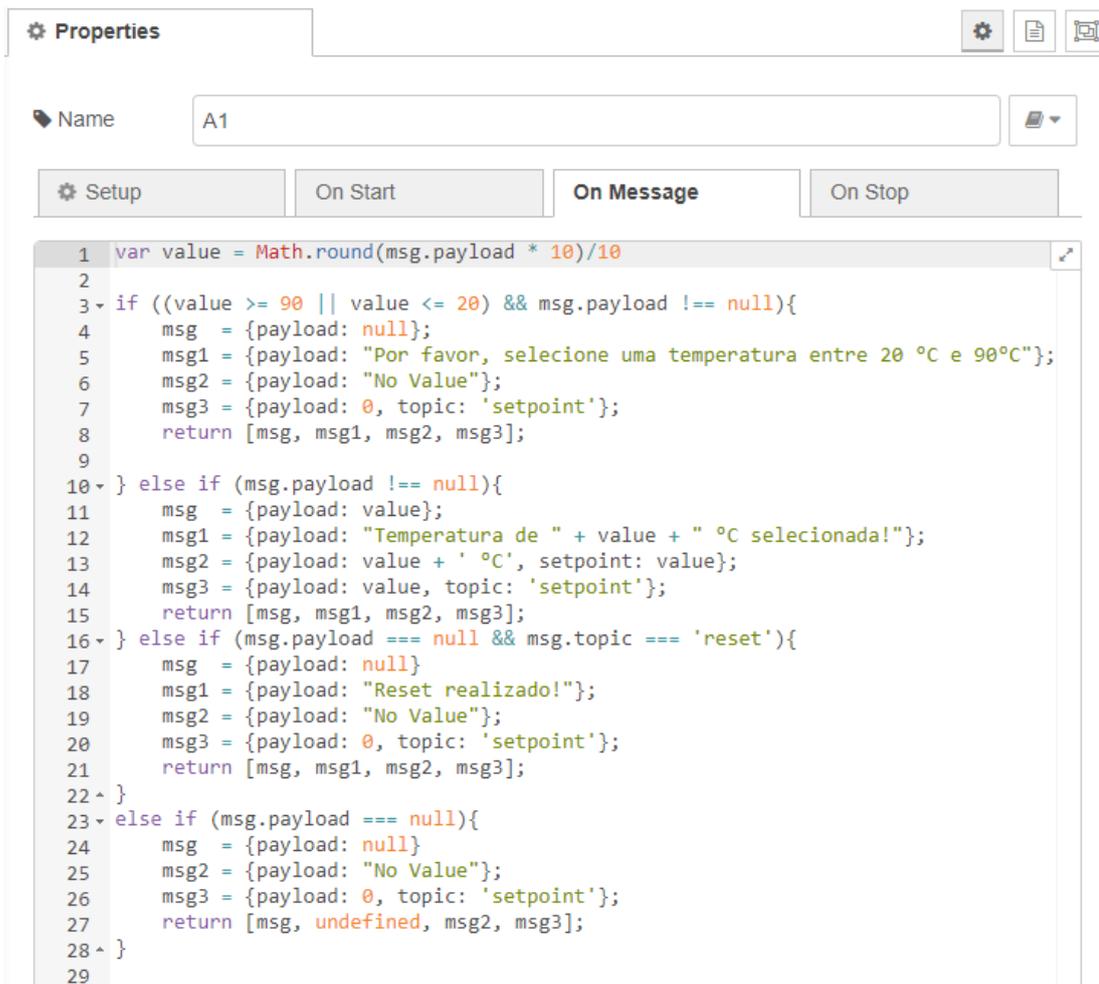
- Quando a mensagem de entrada for nula e a origem da mensagem for o acionamento do botão “Reset”: realiza um reset do medidor “Objetivo” exibindo a medida “No Value”, configura o setpoint do PID para 0 que é o valor padrão, e exibe uma notificação de que o reset foi realizado.

Figura 4.12. Notificação de reset realizado.



- Quando a mensagem de entrada for nula e a origem da mensagem for pela inserção de um valor em branco pelo usuário, no controle “Objetivo (Enter)”: o medidor “Objetivo” exibe a medida “No Value” e configura o setpoint do PID para 0.

Figura 4.13. Programação função A1.

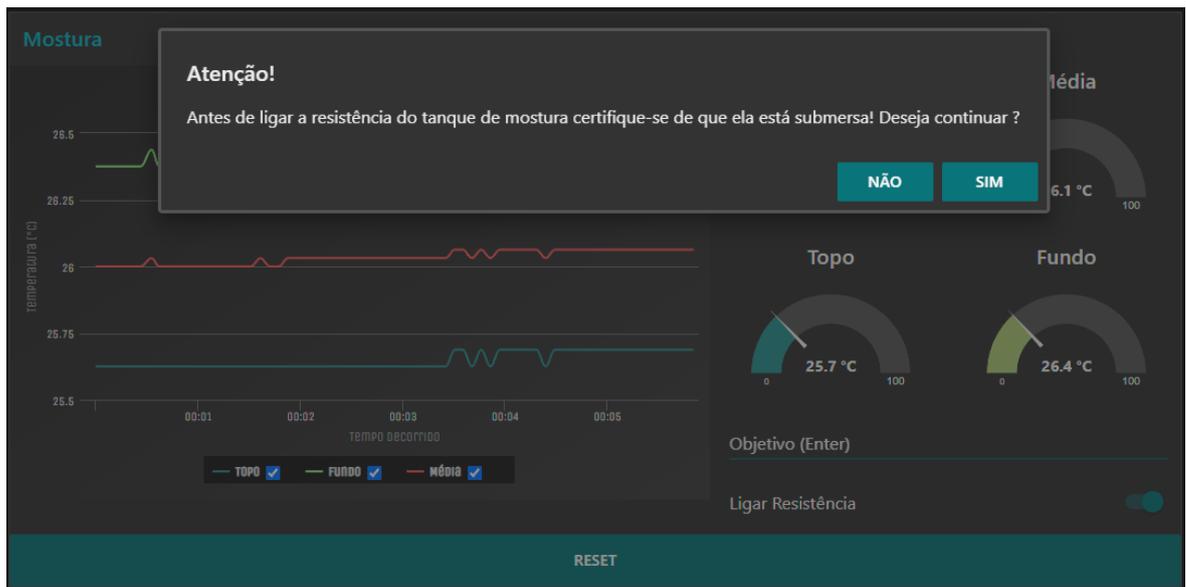


```
1 var value = Math.round(msg.payload * 10)/10
2
3 if ((value >= 90 || value <= 20) && msg.payload !== null){
4   msg = {payload: null};
5   msg1 = {payload: "Por favor, selecione uma temperatura entre 20 °C e 90°C"};
6   msg2 = {payload: "No Value"};
7   msg3 = {payload: 0, topic: 'setpoint'};
8   return [msg, msg1, msg2, msg3];
9
10 } else if (msg.payload !== null){
11   msg = {payload: value};
12   msg1 = {payload: "Temperatura de " + value + " °C selecionada!"};
13   msg2 = {payload: value + ' °C', setpoint: value};
14   msg3 = {payload: value, topic: 'setpoint'};
15   return [msg, msg1, msg2, msg3];
16 } else if (msg.payload === null && msg.topic === 'reset'){
17   msg = {payload: null}
18   msg1 = {payload: "Reset realizado!"};
19   msg2 = {payload: "No Value"};
20   msg3 = {payload: 0, topic: 'setpoint'};
21   return [msg, msg1, msg2, msg3];
22 }
23 else if (msg.payload === null){
24   msg = {payload: null}
25   msg2 = {payload: "No Value"};
26   msg3 = {payload: 0, topic: 'setpoint'};
27   return [msg, undefined, msg2, msg3];
28 }
29
```

O nó “Ligar Resistência” é exibido na forma de botão On/Off no painel de controle, a ação do botão passa por um tratamento no nó “A2” o qual tem as seguintes funções:

- Quando a mudança do controle for de Off para On: Passa uma mensagem para o nó “Continuar?” que irá exibir um diálogo do tipo OK/Cancel no painel de controle perguntando se a resistência a qual irá ligar está submersa, evitando o erro operacional de ligar a resistência a seco e consequentemente a queimar a resistência.

Figura 4.14. Janela de confirmação de resistência submersa.



- Quando a mudança do controle for de On para Off ou quando a mensagens de entrada do nó “A2” for “false”: Manda uma mensagem para desativar o PID e muda a variável do fluxo que guarda o estado da resistência, para “desligada”. Neste caso a mensagem “false” poder vir do botão Reset ou do nó Validação, para ambas origens o efeito a lógica é suficiente.

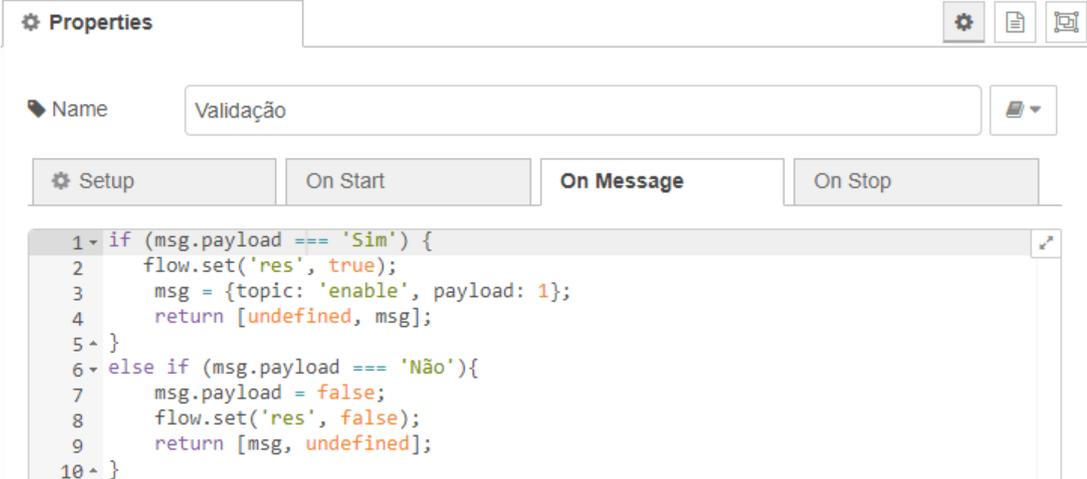
Figura 4.15. Programação função A2.

```
Properties
Name: A2
On Message:
1 if (msg.payload === true) {
2   msg.topic = 'Atenção!';
3   msg.payload = 'Antes de ligar a resistência do tanque de mostura'+
4     'certifique-se de que ela está submersa! \n Deseja continuar?';
5   return [undefined, msg];
6 }else if (msg.payload === false){
7   flow.set('res', false);
8   msg = {topic: 'enable', payload: 0};
9   return [msg, undefined];
10 }
```

O nó “Validação” serve para validar a resposta do usuário quando questionado se a resistência está submersa e se deseja continuar, essa mensagem é exibida para o usuário no nó “Continuar?”.

- Caso a resposta seja “sim”, então o nó manda uma mensagem para ativar o PID e muda a variável do fluxo que guarda o estado da resistência para “ligada”.
- Caso a resposta seja não, o nó muda a variável do fluxo que guarda o estado da resistência para “desligada”, manda uma mensagem retornando o controle “Ligar Resistência” para a posição Off, o que em sequência aciona o nó “A1” e executa a lógica já descrita nos parágrafos anteriores.

Figura 4.16. Programação função Validação.



```
1 if (msg.payload === 'Sim') {
2   flow.set('res', true);
3   msg = {topic: 'enable', payload: 1};
4   return [undefined, msg];
5 }
6 else if (msg.payload === 'Não'){
7   msg.payload = false;
8   flow.set('res', false);
9   return [msg, undefined];
10 }
```

O nó “Reset” exibe um controle do tipo botão no painel de controle, quando acionado retorna para os valores padrão todo o painel do bloco correspondente, incluindo os gráficos e o estado “desligado” para resistência. Para isso, utiliza-se o nó “Reiniciar” para auxiliar nos diferentes tipos de mensagens a depender do controle ou variável que se reinicia e o nó “Deleta Dados” para excluir os arquivos “csv” criados.

Figura 4.17. Programação função Reiniciar.

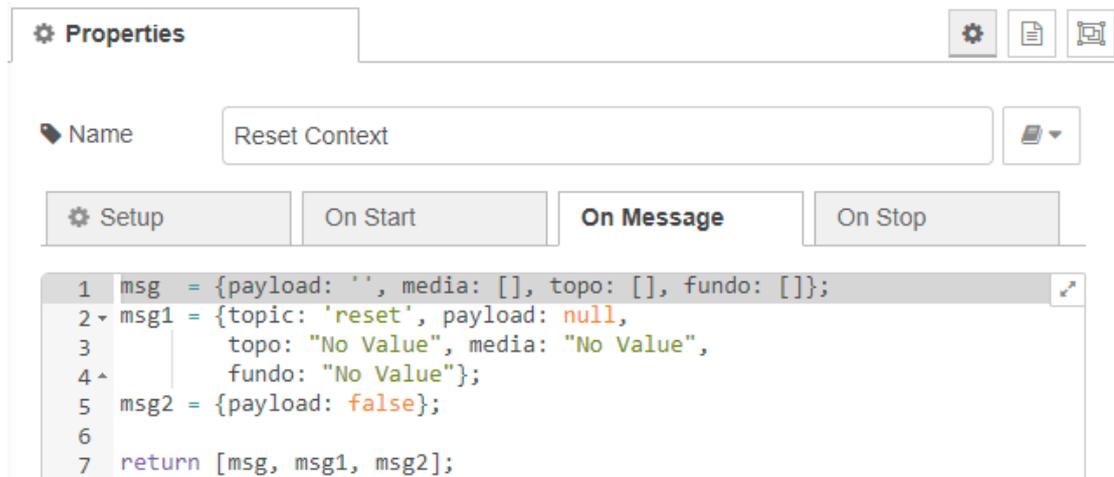
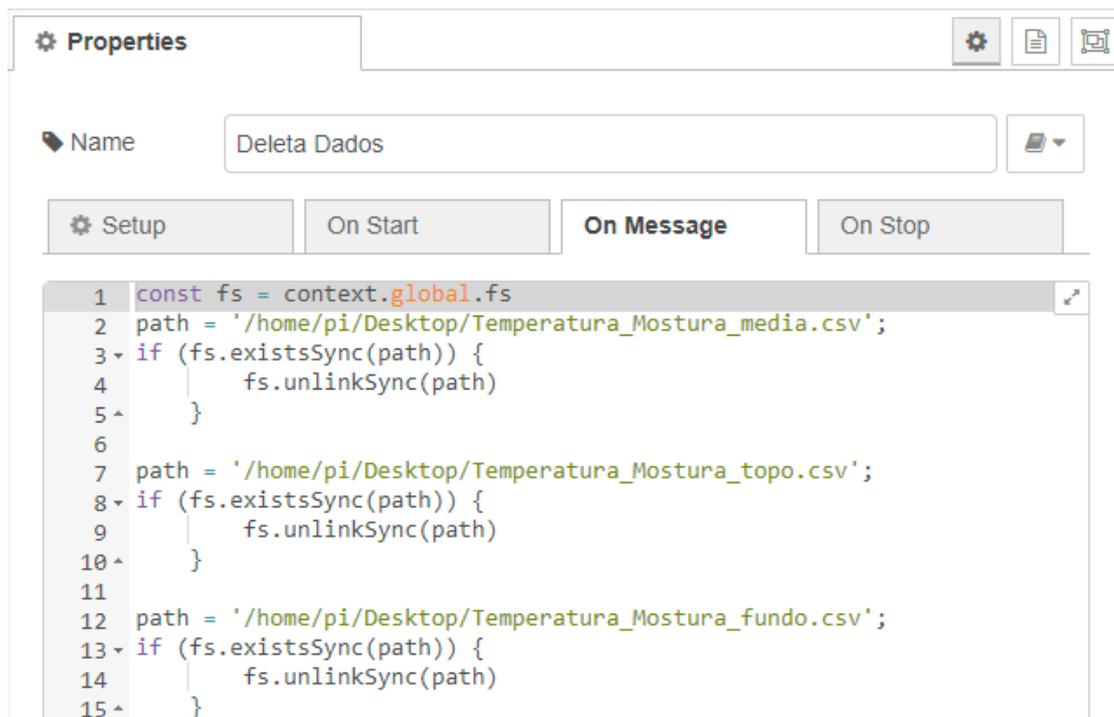


Figura 4.18. Programação função Deleta Dados.

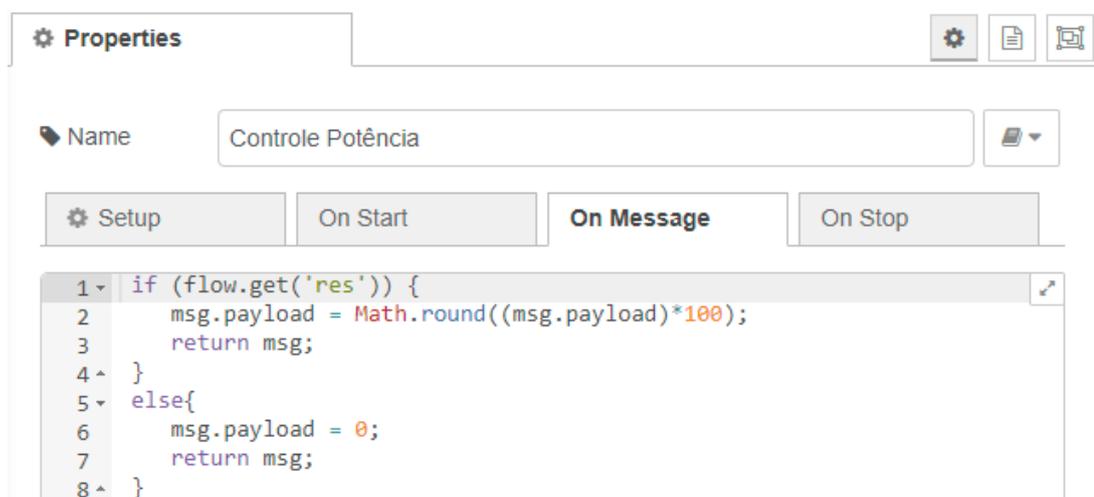


O nó “PID” recebe a média das temperaturas, o setpoint e o estado de ligado ou desligado de outros nós e utiliza essas informações para realizar um controle no ângulo de disparo do relé de estado sólido. Os parâmetros do PID foram configurados através de tentativa e erro e este presente trabalho de graduação não se ateu ao rigor do cálculo desses

parâmetros, registro das tentativas, nem ao algoritmo da biblioteca utilizada, apenas ao efeito do controle desejado.

A resposta do nó “PID” é um número decimal no intervalo de 1 a 0 enviado na msg.payload. Essa resposta é utilizada no nó “Controle Potência” multiplicado por 100 e arredondando para o inteiro mais próximo caso a variável que guarda o estado da resistência seja verdadeira, ou seja a resistência deve ser ligada. No caso contrário apenas retorna 0:

Figura 4.19. Programação função Controle Potência.



A frequência da corrente alternada no Brasil é de 60 Hz, isso significa que cada semi-ciclo, ou seja, intervalos de passagem pelo zero, conhecido como “zero crossing”, tem duração de 8333 microssegundos, porém devido a latências inerentes a plataforma de programação e ao próprio sistema operacional do Raspberry Pi, optou-se por limitar o atraso em no máximo 6500 microssegundos beneficiando a estabilidade do controle.

Por fim, o nó “PWM” executa uma modulação por largura de pulso utilizando uma frequência de 1 Hz, ou seja, de um ciclo por segundo. A largura do pulso é controlada pela saída do nó “Controle de Potência”. Por exemplo, se a saída do nó “Controle de Potência” for 70, então o nó “PWM” vai manter a resistência ligada durante 70% do ciclo de 1 segundo. Controlando dessa forma, a potência fornecida para a resistência.

4.4 TANQUE DE LAVAGEM E TANQUE DE FERVURA

Figura 4.20. Programação da Lavagem.

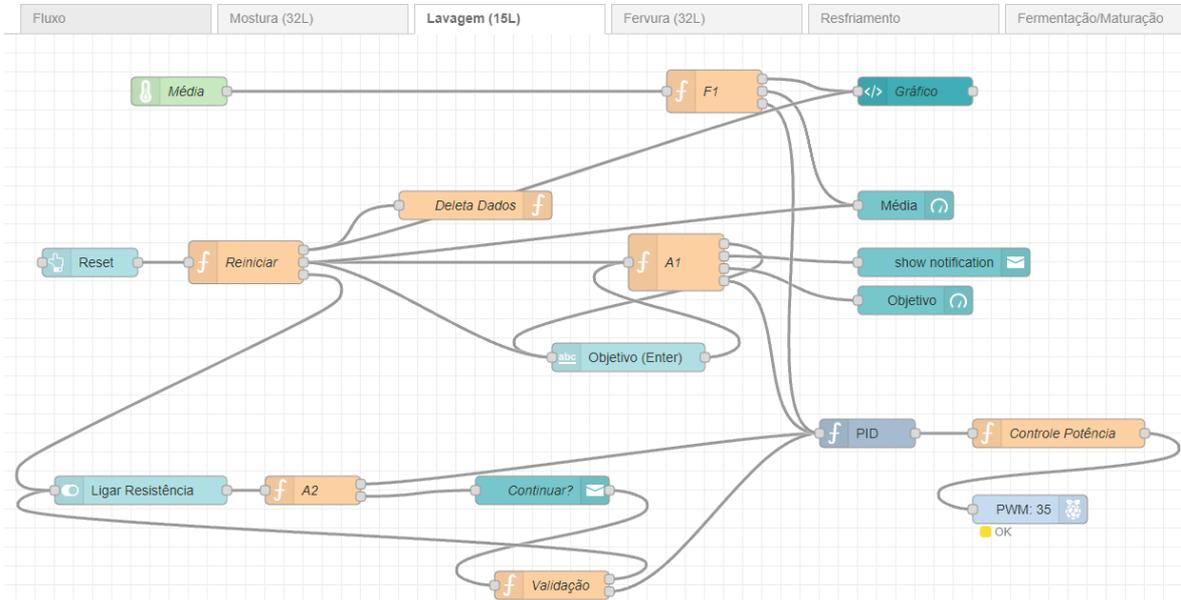
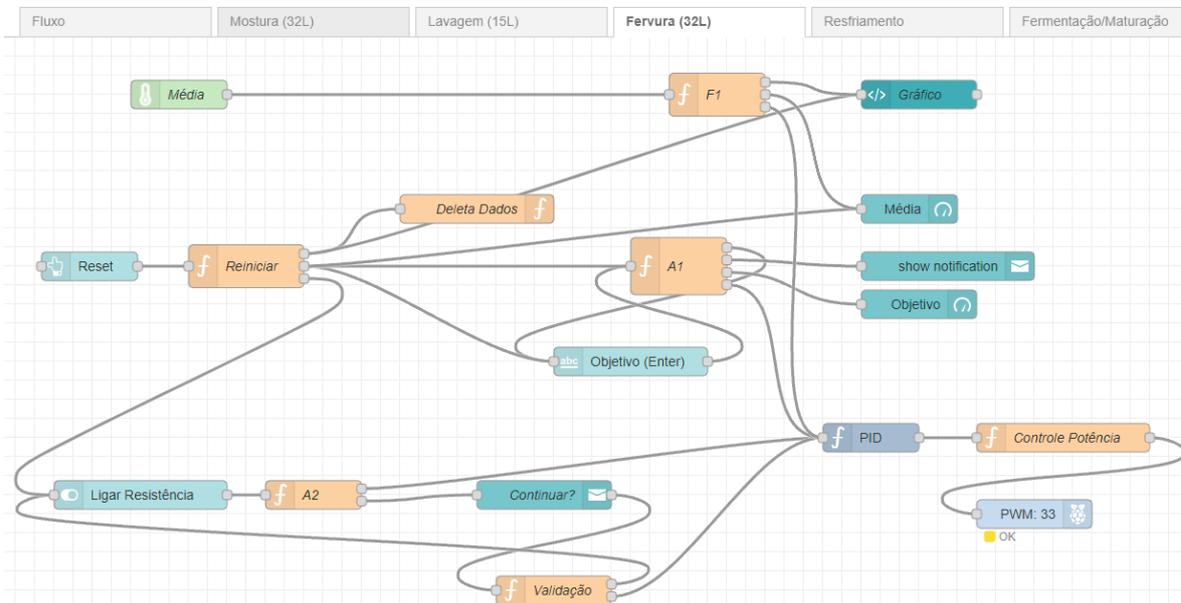


Figura 4.21. Programação da Fervura.



A programação dos tanques de lavagem e fervura são na maior parte iguais ao do tanque de mostura. As diferenças são:

- Esses tanques utilizam apenas um sensor de temperatura, portanto, não é necessário o tratamento para do desencontro de informações de sensores, conseqüentemente, a temperatura média do tanque é dada por esse único sensor.
- A função F1 salva apenas um arquivo já que se tem apenas um sensor de temperatura.
- O pino do Raspberry Pi que controla o relé de estado sólido também é diferente para que a resistência controlada seja a dos tanques correspondentes
- Os objetos do painel de controle assim como as variáveis HTML utilizadas no gráfico precisam ser diferentes para não haver conflito de informações no painel de controle.
- As mensagens e notificações para o usuário são personalizadas para o tanque que se opera.

Com relação as variáveis do fluxo, não é preciso alterar seus nomes pois elas são acessíveis apenas no contexto do fluxo, dessa forma, a maior parte da programação e da lógica é idêntica ao tanque de mostura.

Figura 4.22. Função F1 para um sensor de temperatura.



The image shows a screenshot of a Node-RED function node configuration. The node is named 'F1'. The 'On Message' tab is selected, showing the following JavaScript code:

```
1 media = msg.payload
2 //Tratando a segunda saída
3 msg1 = {media: media.toFixed(1) + ' °C'};
4 //Tratando a terceira saída
5 msg2 = {payload: Math.round(media*10)/10};
6
7 const fs = context.global.fs
8
9 if (media != undefined){
10   const fs = context.global.fs;
11   //Local do arquivo
12   path = '/home/pi/Desktop/Temperatura_Lavagem.csv';
13
14   //Cria um arquivo em branco caso não exista
15   if (!fs.existsSync(path)) {
16     fs.appendFileSync(path, Date.now()+","+media)
17   }
18   else{
19     //Insere dados se o arquivo existe
20     fs.appendFileSync(path, "\n"+Date.now()+","+media);
21     //Lê o arquivo
22     data = fs.readFileSync(path, 'utf-8');
23     //Convert CSV em Array de Números
24     rows = data.split("\n");
25
26     msg.media = rows.map(function (row, index)
27     {
28       //Se é a primeira medida
29       //guarda o primeiro valor
30       if (index == 0)
31       {
32         return row.split(",").map(function(item, index, arr)
33         {
34           if (index === 0){
35             first = Number(item);
36             item = 0;
```

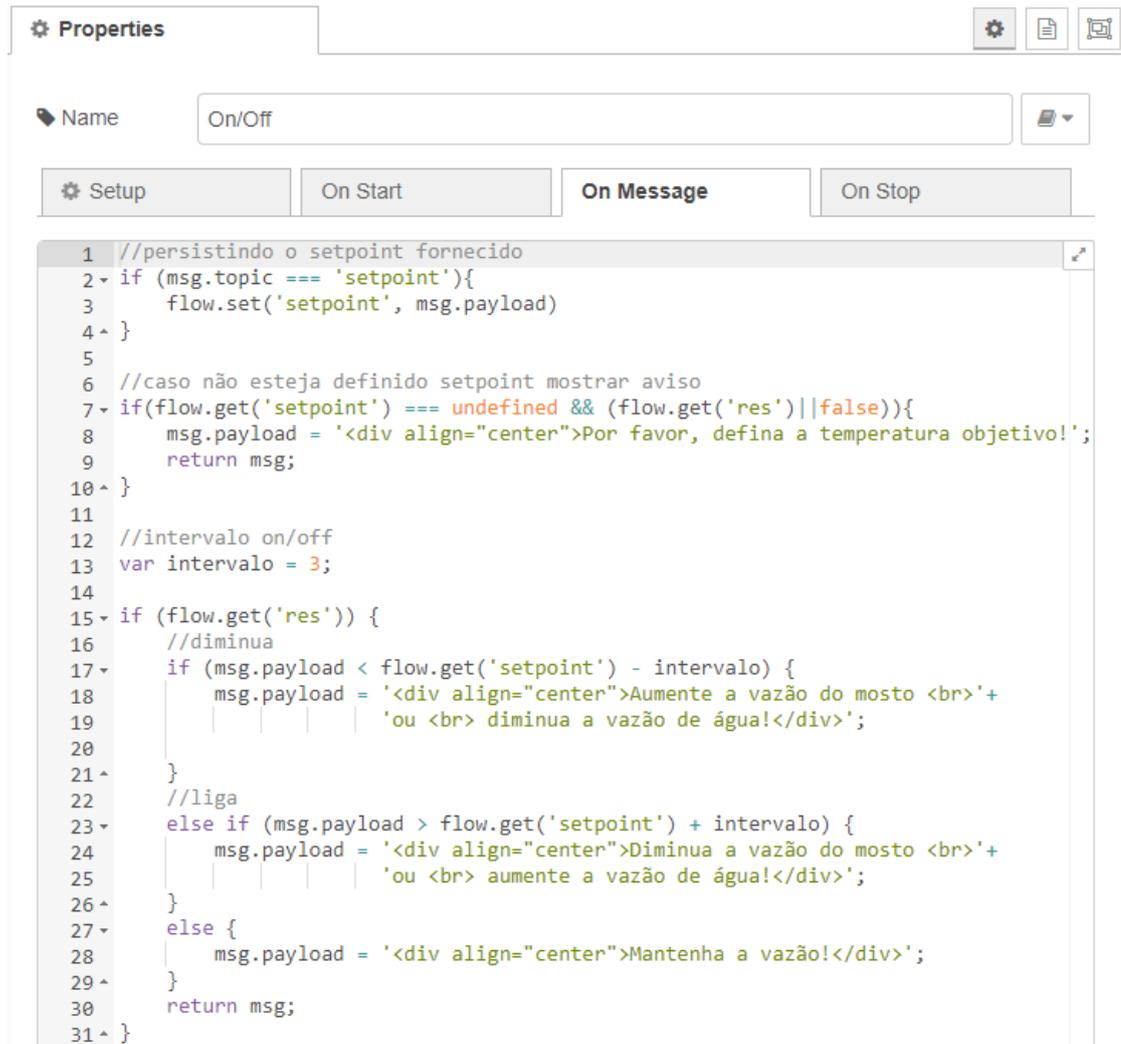
Figura 4.22. Função F1 para um sensor de temperatura (continuação).

```
37 ^      }
38 v      else{
39      |         item = Number(item);
40      |
41 ^      |     }
42      |     return item
43 ^      |     }
44 ^      |
45 ^      |     }
46      |     //não é a primeira medida subtrai o primeir valor
47      |     //para obter a duração
48 v      |     else{
49 v      |         return row.split(",").map(function(item, index, arr){
50 v      |             |     if (index === 0){
51      |             |         |     item = Number(item) - first;
52 ^      |             |         |     }
53      |             |         |     else{item = Number(item)}
54      |             |         |     return item;
55 ^      |             |         |     }
56 ^      |             |         }
57 ^      |             |     }
58      |             }
59 ^      |         }
60 ^      |     )
61 ^      | }
62 ^      }
63      return [msg, msg1, msg2];
64
```

Enabled

o setpoint não foi definido corretamente fazendo com que não haja mensagem de controle e seja exibido uma notificação para usuário pedindo a definição da temperatura objetivo.

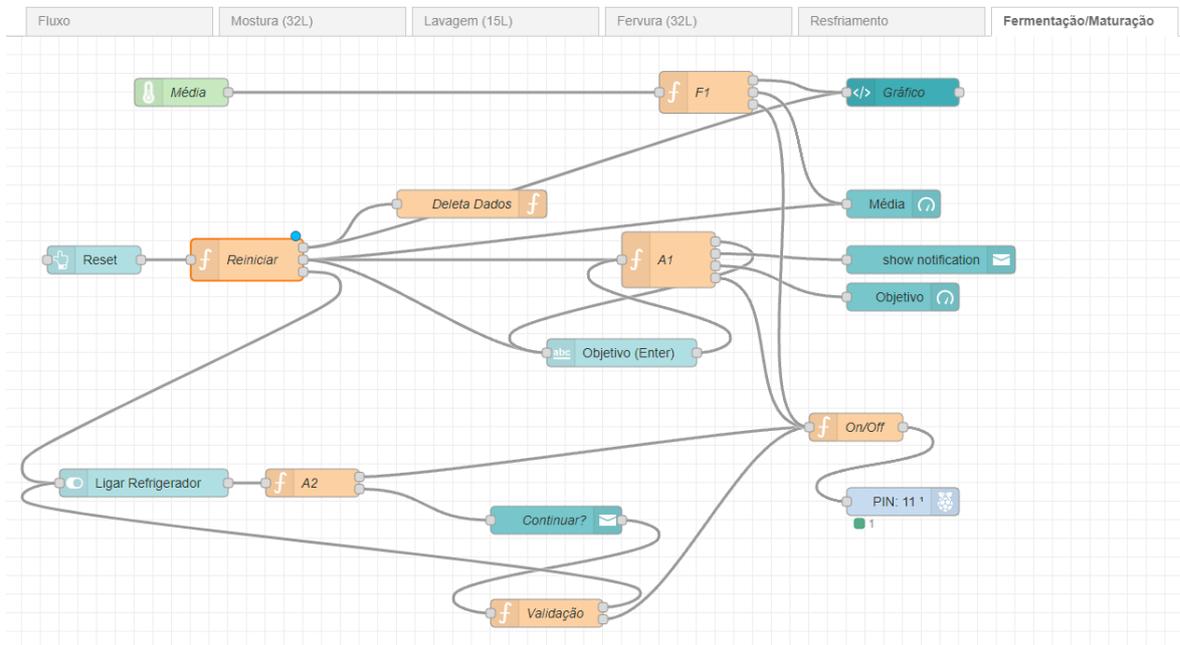
Figura 4.24. Programação da função On/Off do resfriamento.



```
1 //persistindo o setpoint fornecido
2 if (msg.topic === 'setpoint'){
3     flow.set('setpoint', msg.payload)
4 }
5
6 //caso não esteja definido setpoint mostrar aviso
7 if(flow.get('setpoint') === undefined && (flow.get('res')||false)){
8     msg.payload = '<div align="center">Por favor, defina a temperatura objetivo!';
9     return msg;
10 }
11
12 //intervalo on/off
13 var intervalo = 3;
14
15 if (flow.get('res')) {
16     //diminua
17     if (msg.payload < flow.get('setpoint') - intervalo) {
18         msg.payload = '<div align="center">Aumente a vazão do mosto <br>'+
19             '| | | | | 'ou <br> diminua a vazão de água!</div>';
20     }
21 }
22 //liga
23 else if (msg.payload > flow.get('setpoint') + intervalo) {
24     msg.payload = '<div align="center">Diminua a vazão do mosto <br>'+
25         '| | | | | 'ou <br> aumente a vazão de água!</div>';
26 }
27 else {
28     msg.payload = '<div align="center">Mantenha a vazão!</div>';
29 }
30 return msg;
31 }
```

4.6 FERMENTAÇÃO E MATURAÇÃO

Figura 4.25. Programação da Fermentação/Maturação.



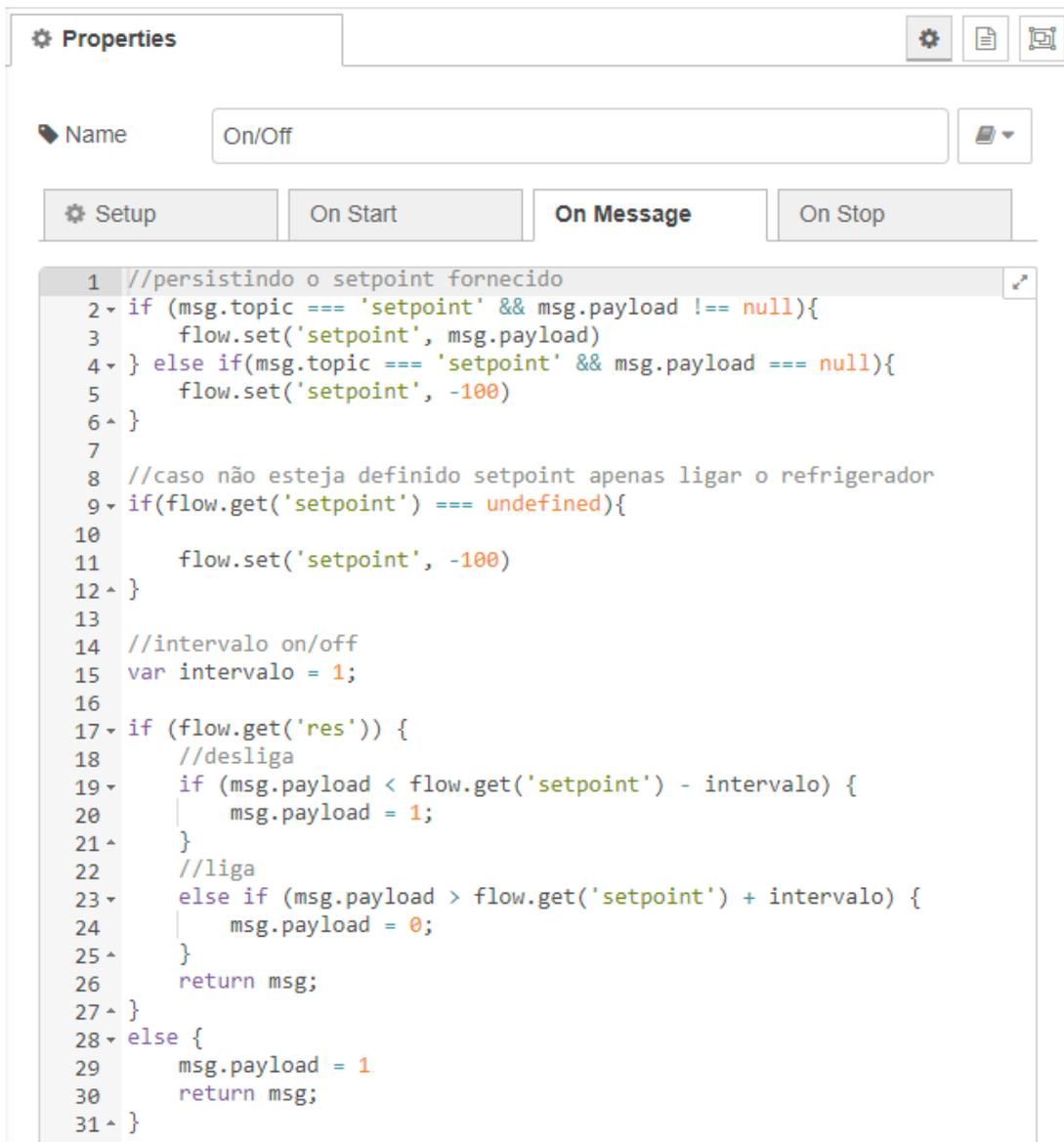
No processo de fermentação e maturação, o objetivo é controlar a temperatura do refrigerador por meio de um relé mecânico comuta a energia elétrica fornecida. Sendo a programação semelhantes aos demais fluxos apresentados. O intervalo de temperatura objetivo válido está entre 0°C e 30°C. A mensagem exibida pelo nó “Continuar?” é personalizada para instruir ao usuário a configurar o termostato do refrigerador em potência máxima, evitando que o controle do refrigerador impeça o controle do Raspberry Pi.

Figura 4.26. Janela de confirmação de refrigerador em potência máxima.



O controle é do tipo On/Off com uma histerese de 1°C, sendo o acionamento do relé mecânico realizado pelo nó “Controle Refrigerador”. Nesse fluxo em específico, quando não se é determinado um setpoint e o nó “Ligar Refrigerador” é alterado para a posição On, o refrigerador é acionado normalmente sem qualquer tipo de controle por parte do programa.

Figura 4.27. Programação da função On/Off do fermentador.



The screenshot shows a software configuration window titled "Properties" for a function named "On/Off". The "On Message" tab is active, displaying the following JavaScript code:

```
1 //persistindo o setpoint fornecido
2 if (msg.topic === 'setpoint' && msg.payload !== null){
3     flow.set('setpoint', msg.payload)
4 } else if(msg.topic === 'setpoint' && msg.payload === null){
5     flow.set('setpoint', -100)
6 }
7
8 //caso não esteja definido setpoint apenas ligar o refrigerador
9 if(flow.get('setpoint') === undefined){
10
11     flow.set('setpoint', -100)
12 }
13
14 //intervalo on/off
15 var intervalo = 1;
16
17 if (flow.get('res')) {
18     //desliga
19     if (msg.payload < flow.get('setpoint') - intervalo) {
20         msg.payload = 1;
21     }
22     //liga
23     else if (msg.payload > flow.get('setpoint') + intervalo) {
24         msg.payload = 0;
25     }
26     return msg;
27 }
28 else {
29     msg.payload = 1
30     return msg;
31 }
```

5 RESULTADOS E DISCUSSÕES

Foram realizados três testes para verificar a funcionalidade do projeto como um todo. Cabe ressaltar que os testes visam verificar se o sistema de controle responde como o desejado, sendo dispensado o rigor de se utilizar os mesmos equipamentos que as cervejarias usam. Uma vez se provando possível o controle, ajustes de parâmetros seriam necessários para a CAMA atuar no mesmo tipo de finalidade utilizando equipamentos diferentes.

5.1 CONTROLE DE REFRIGERAÇÃO

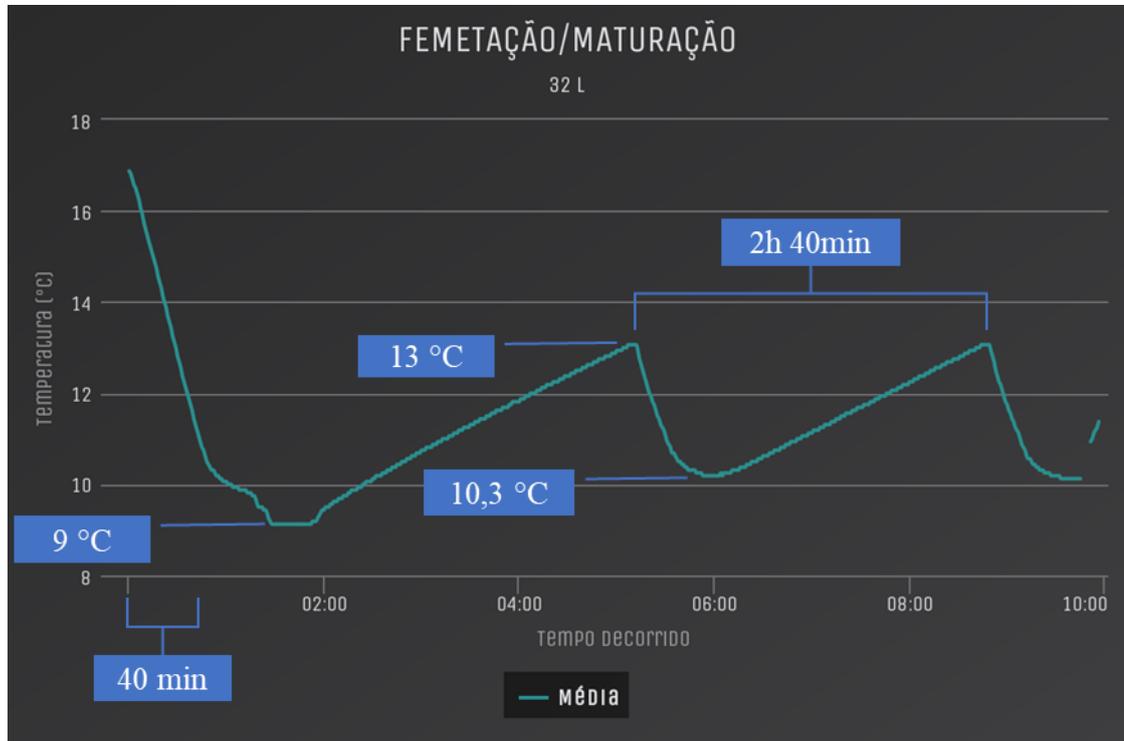
O primeiro teste foi o controle do refrigerador, para isso se usou uma garrafa com aproximadamente de com um litro de água e o controle foi realizado em um mini refrigerador, conforme ilustra a Figura 5.1.

Figura 5.1. Esquema para controle de refrigeração.



O resultado obtido está apresentado na Figura 5.2:

Figura 5.2. Resultados do controle de refrigeração.



O experimento durou cerca de 10 horas e ocorreu durante o período noturno.

Como pode ser observado, partiu-se de uma temperatura de aproximadamente 17 °C com objetivo de manter a 12 °C no processo de fermentação. Para isso, quando o sistema atinge a temperatura de 11 °C ele desliga o refrigerador, então ao perder calor e atingir 13 °C ele liga novamente.

Houve uma temperatura mínima atingida de aproximadamente 9 °C logo no início do controle, 1 hora e 40 minutos. Isso ocorreu pois o refrigerador estava desligado no início do experimento a uma temperatura bem mais elevada que a objetivo, dessa forma o sistema ficou ligado por mais tempo para refrigerar, aproximadamente 40 minutos. Quando o controle foi desligado o ambiente refrigerado foi se tornando mais homogêneo, gerando essa temperatura mínima. Esse aspecto observado pode ser corrigido refrigerando o sistema a uma temperatura próxima do objetivo antes do controle.

A partir de 2 horas e 30 minutos, o controle atinge um comportamento padrão, com picos de 13 °C e mínimos de 10,3 °C, com um ciclo de 2 horas e 40 minutos. Ou seja, quando o controle desliga o refrigerador a temperatura diminui mais 0,7 °C. Isso poderia ser corrigido ajustando o controle para desligar a uma temperatura mais alta, por exemplo 11,7 °C.

A taxa de resfriamento é maior que a de aquecimento devido ao isolamento do refrigerador. Isso não chega a ser um problema, dado que o real objetivo é um intervalo de temperaturas ao redor de uma temperatura ideal.

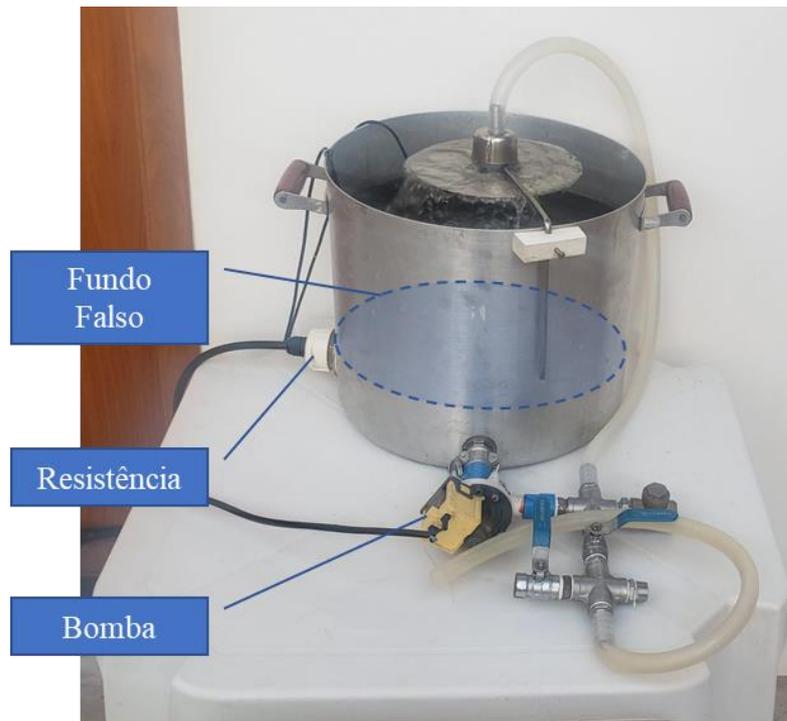
Cabe salientar que a diminuição do intervalo objetivo gera um custo de desgaste do sistema. Quanto menor o intervalo que se deseja, mais será preciso desligar e ligar o refrigerador podendo diminuir sua vida útil.

Pode-se dizer que o controle do refrigerador funciona e que ajustes são necessários a depender do controle desejado.

5.2 CONTROLE DE AQUECIMENTO

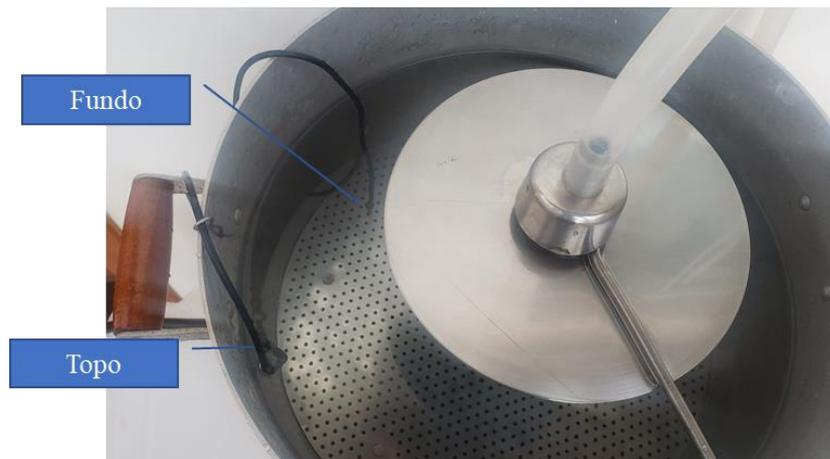
Para o teste de aquecimento se utilizou um tanque de alumínio com uma resistência interna de 3000 W a qual fica submersa do líquido e abaixo do fundo falso, Figura 5.3.

Figura 5.3. Esquema para o teste de aquecimento.



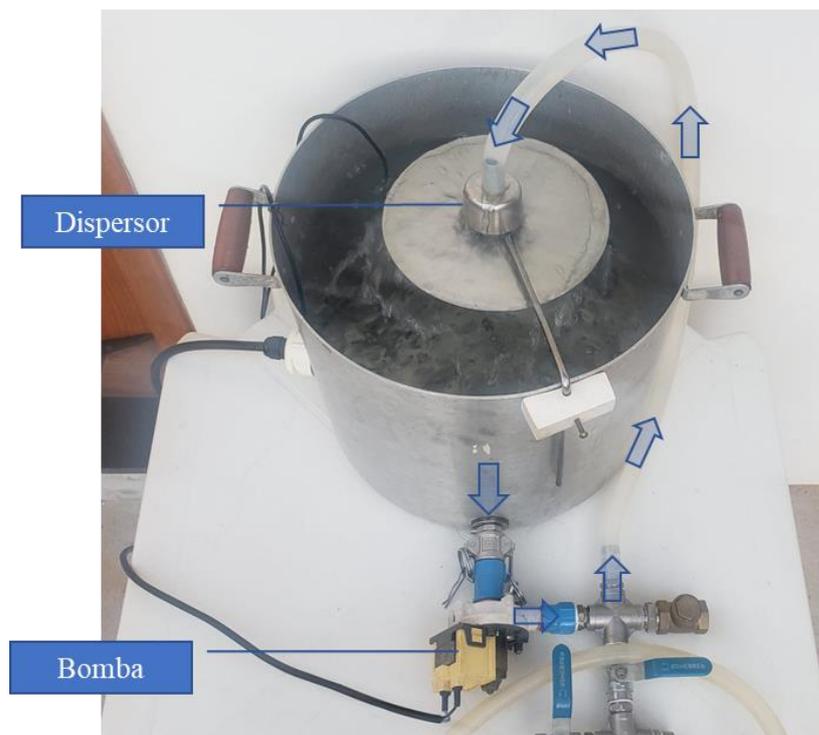
No tanque foram colocados aproximadamente 30L de água e foi disposto um sensor abaixo do fundo falso, chamado de “Fundo” e outro acima, chamado de “Topo” conforme Figura 5.4.

Figura 5.4. Disposição dos sensores no teste de aquecimento.



A operação de aquecimento foi realizada juntamente com a recirculação, Figura 5.5. Isso pode parecer contra intuitivo dado que se espera uma convecção natural no tanque já que o aquecimento ocorre no fundo, porém o fundo falso prejudica essa convecção e na operação real de produção os grãos também vão prejudicar essa convecção, sendo então a recirculação necessária para maior homogeneidade do tanque.

Figura 5.5. Recirculação efetuada no teste de aquecimento.



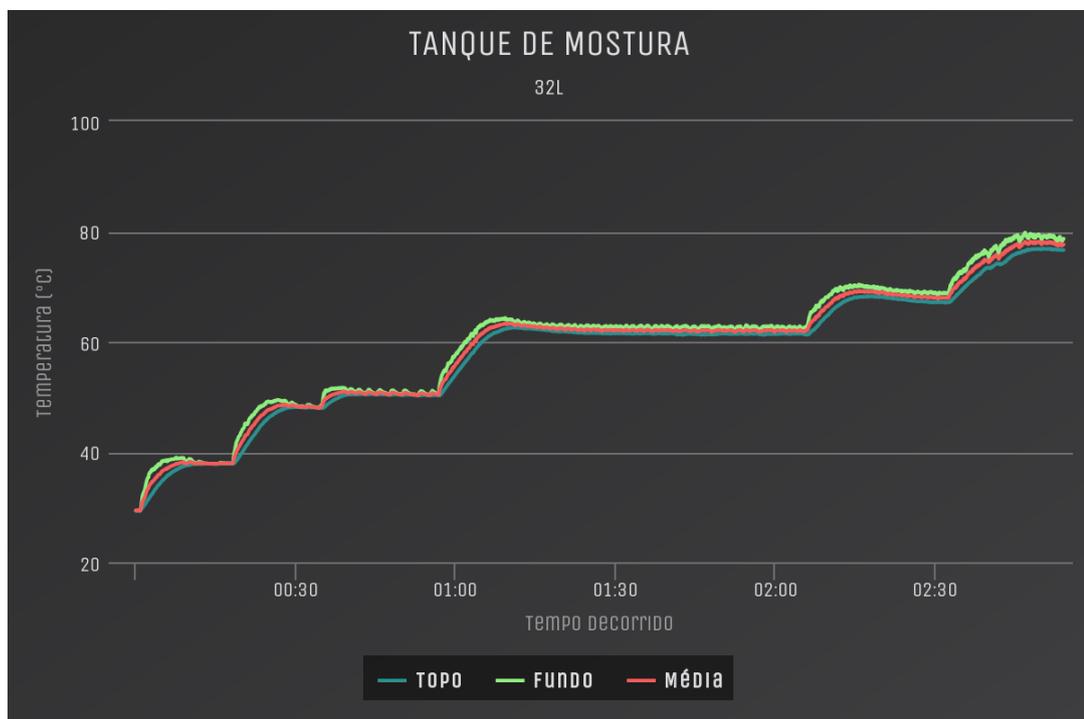
Para o teste, utilizou-se uma rampa objetivo ilustrada na Figura 5.6, esses valores foram obtidos com base na Figura 2.3.

Figura 5.6. Rampa de temperatura objetivo.

Temperatura (°C)	Tempo (min)
38	10
48	10
50	20
62	60
68	20
77	10

O resultado obtido está apresentado na Figura 5.7:

Figura 5.7. Resultados do controle de aquecimento.

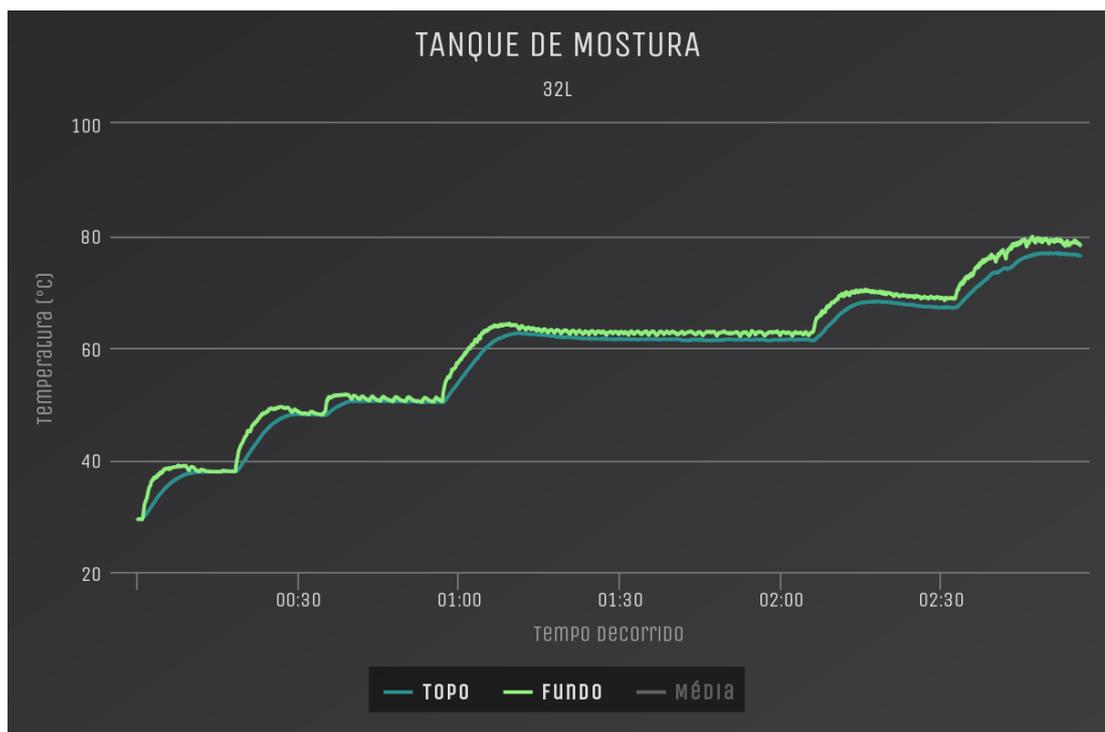


Como se pode observar, a rampa de temperatura corresponde ao esperado. O tempo em cada estágio foi contado a partir do momento em que se atinge a temperatura desejada pela primeira vez.

Observa-se que no início do experimento a taxa de subida de temperatura é maior e os ruídos são menores. Isso é esperado, pois no início, se está a temperaturas menores sendo o fenômeno de perda de calor para o ambiente menor tendo taxas maiores de subida de temperatura e maior estabilidade do sistema.

A diferença entre as temperaturas “Topo” e “Fundo” ocorre principalmente na transição de um estágio para o outro, Figura 5.8. O que não é um problema já que a máxima diferença observada é de 4 °C que se reduz ao decorrer do estágio.

Figura 5.8. Resultados do aquecimento topo e fundo.



Pode-se dizer que o controle do aquecimento funciona e que se pode fazer ajustes no PID a depender do equipamento que se aplica.

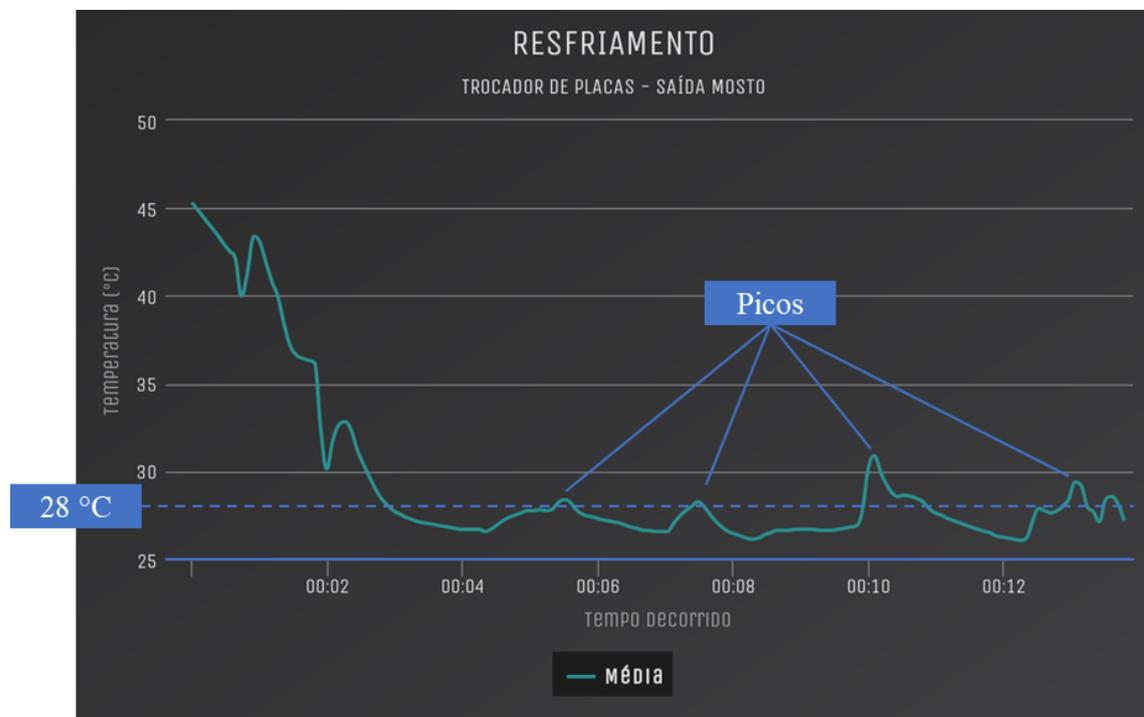
5.3 CONTROLE DO RESFRIAMENTO

No experimento do resfriamento se utilizou a operação manual da válvula na tentativa de simular um atuador automático, dessa forma o controle indica se deve aumentar ou diminuir a vazão das correntes no trocador dado a temperatura de saída do mosto.

A temperatura ideal, recomendada para inoculação do fermento é de 25 °C. Com isso se utilizou um intervalo de 25 °C ± 3 °C, partindo de um líquido a 63°C.

A Figura 5.9 ilustra a temperatura de saída da corrente quente:

Figura 5.9. Resultados do controle de resfriamento.



Do início do experimento até aproximadamente 3 minutos há um período de entendimento do sistema e regulagem da vazão da água de resfriamento obtida da rede. Após esse período foi necessário aplicar a vazão máxima de água de resfriamento, para se obter uma vazão razoável de corrente resfriada. Cabe ressaltar que o volume de água de resfriamento utilizado foi considerável, o que contribuiu para se optar por manter a temperatura de saída na parte superior do intervalo, realizando alguns “picos”, a partir do aumento da vazão da corrente quente, para diminuir o tempo de operação.

Dado o resultado apresentado. No aspecto de controle, o software é capaz de indicar qual a ação a ser tomada. Porém o líquido de resfriamento a ser utilizado precisa estar a uma menor temperatura do que a fornecida pela rede para que haja um menor volume gasto. Com isso também se evita que se opere com a válvula quase fechada, fornecendo uma vazão muito baixa de corrente quente, fazendo com que a operação seja mais lenta.

6 CONSIDERAÇÕES FINAIS

Diante dos resultados se conclui que o centro de controle e o software desenvolvido atendeu aos seus objetivos de controle de temperatura tanto no processo de refrigeração quanto no processo de aquecimento. No processo de resfriamento o controle também foi realizado, mas o processo precisa ser reajustado, implementando controladores de válvula automáticos e o uso de um líquido de resfriamento a temperaturas mais baixas.

Outros sensores como medidores de vazão, pH, densidade e atuadores automáticos de válvula são sugestões de implementação ao presente trabalho. Sugere-se também a busca por uma alternativa a placa de controle de temperatura desenvolvida para o projeto, já que este componente é único e uma falha pode inviabilizar o controle. Uma alternativa seria os relés de estado sólido encontrados no mercado.

7 REFERÊNCIAS

CERVBRASIL. Anuário 2020. Cervbrasil, 2021. Disponível em: <
http://www.cervbrasil.org.br/novo_site/wp-content/uploads/2021/04/anuariocerveja2.pdf>.
Acesso em: 15 de setembro 2021.

CERVBRASIL. Anuário 2015. Cervbrasil, 2021. Disponível em:
<http://www.cervbrasil.org.br/novo_site/anuarios/ANUARIO_CB_2015_WEB.pdf>.
Acesso em: 15 de setembro 2021.

BARTH-HAAS. BarthHaas Report Hops 2020/2021. Barthhaas, 2021. Disponível em:
<https://www.barthhaas.com/fileadmin/user_upload/kampagnen/barthhaas_bericht/BarthHaas_Report_Hops_2020_21.pdf>. Acesso em: 15 de setembro 2021.

BARTH-HAAS. BarthHaas Report Hops 2019/2020. Barthhaas, 2020. Disponível em: <
https://www.barthhaas.com/fileadmin/user_upload/downloads/barth-berichte-broschueren/barth-berichte/englisch/2010-2020/barthhaas_report_2020_en.pdf>. Acesso em: 15 de setembro 2021.

CHENG C.; GUELFIRAT, T.; MESSINGER, C.; SCHMITT, J.; SCHNELTE, M.; WEBER, P. Semantic Degrees for Industrie 4.0 Engineering: Deciding on the Degree of Semantic Formalization to Select Appropriate Technologies. In Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering (pp. 1010–1013). Association for Computing Machinery.

URBIKAIN, G.; ALVAREZ, A.; LÓPEZ DE LACALLE, L.N.; ARSUAGA, M.; ALONSO, M.A.; VEIGA, F. A reliable turning process by the early use of a deep simulation model at several manufacturing stages. Preprints, [s.l.], 2016.

PLATTFORM. Plattform Industrie 4.0, 2021. Home. Disponível em: <
<https://www.plattform-i40.de/IP/Navigation/EN/Home/home.html>> Acesso: 15 de outubro e 2021.

FERENCZ, Katalin; DOMOKOS, József. Using Node-RED platform in an industrial environment. XXXV. Jubileumi Kandó Konferencia, Budapest, p. 52-63, 2019.

DOMÍNGUEZ, Manuel et al. Development of a Remote Industrial Laboratory for Automatic Control based on Node-RED. IFAC-PapersOnLine, v. 53, n. 2, p. 17210-17215, 2020.

MORADO, R. Larousse da Cerveja. Larousse do Brasil. 1 ed, São Paulo, 2011.

FARBER, Matthew; BARTH, Roger. Mastering brewing science: Quality and production. John Wiley & Sons, 2019.

MOSHER, Randy. Mastering Homebrew: the complete guide to brewing delicious beer. Chronicle Books, 2015.

OSCILLATING u-tube. Wikipedia, 2020. Disponível em: <
https://en.wikipedia.org/wiki/Oscillating_U-tube/>. Acesso: 16 de outubro de 2021.

CERVEJADACASA. Cerveja da Casa, 2021. Balde Fermentador 30L com Torneira, Airlock e Anel de Silicone. Disponível em: <
<https://www.cervejadacasa.com/equipamentos/fermentacao-maturacao/baldes/balde-fermentador-30l-com-torneira-airlock-e-anel-de-silicone>>. Acesso: 16 de outubro e 2021.

WHAT is a Raspberry Pi. Raspberrypi, 2021. Disponível em: <
<https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>>. Acesso: 27 de setembro de 2021.

RASPBERRY Pi 3 Model B+. Raspberrypi, 2021a. Disponível em: <
<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>>. Acesso: 27 de setembro de 2021.

RASPBERRY Pi Documentation. Raspberrypi, 2021b. Disponível em: <
<https://www.raspberrypi.org/documentation/computers/os.html>>. Acesso: 27 de setembro de 2021.

BRAGA, Newton C. Relés: Circuitos e aplicações. Editora Newton C. Braga, 2012.

AHMED, Ashfaq. Eletrônica de potência. Pearson Education do Brasil, 2008.

MAXIM INTEGRATED. DS18B20 Programmable Resolution 1-Wire Digital Thermometer. 2019.

NODE-RED. Node-RED, 2021a. Home. Disponível em: <<https://nodered.org/>>. Acesso: 09 de outubro e 2021.

NODE-RED. Node-RED, 2021b. About. Disponível em: <<https://nodered.org/>>. Acesso: 09 de outubro e 2021.

HIGHCHARTS. HighCharts, 2021. Home. Disponível em: <
<https://www.highcharts.com/>> Acesso: 13 de outubro e 2021.

WORKING with context. Node-RED, 2021a. Disponível em:
<<https://nodered.org/docs/user-guide/context/>>. Acesso: 14 de setembro de 2021.

WORKING with messages. Node-RED, 2021b. Disponível em: <
<https://nodered.org/docs/user-guide/messages/>> Acesso: 14 de setembro de 2021.

EXAMPLE - using an external javascript in a function node. Node-RED, 2021. Disponível
em: < <https://flows.nodered.org/flow/195773d3b493d81c9bf012f64da02ea3>> Acesso: 18
de outubro de 2021.

FS. Npmjs, 2021. Disponível em: <<https://www.npmjs.com/package/fs>> Acesso: 18 de
outubro de 2021.

8 APÊNDICE

8.1 PROGRAMAÇÃO EXPORTADA DO NODE-RED

```
[{"id":"29e0e22c.ba14ae","type":"tab","label":"Fluxo","disabled":false},{ "id":"7999b7cf.b7c508","type":"tab","label":"Mostura (32L)","disabled":false,"info":""},{ "id":"d880af12.1b24f","type":"tab","label":"Lavagem (15L)","disabled":false,"info":""},{ "id":"31c8b922.7219c6","type":"tab","label":"Fervura (32L)","disabled":false,"info":""},{ "id":"fc31302a.f6743","type":"tab","label":"Resfriamento","disabled":false,"info":""},{ "id":"c5dc1810.861398","type":"tab","label":"Fermentação/Maturação","disabled":false,"info":""},{ "id":"c96122d3.3f04c","type":"ui_base","theme":{"name":"theme-dark","lightTheme":{"default":"#0094CE","baseColor":"#0094CE","baseFont":"Helvetica Neue","edited":true,"reset":false},"darkTheme":{"default":"#097479","baseColor":"#097479","baseFont":"-apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica Neue,sans-serif","edited":true,"reset":false},"customTheme":{"name":"Untitled Theme 1","default":"#4B7930","baseColor":"#4B7930","baseFont":"Helvetica Neue"},"themeState":{"base-color":{"default":"#097479","value":"#097479","edited":false},"page-titlebar-background-color":{"value":"#097479","edited":false},"page-background-color":{"value":"#111111","edited":false},"page-sidebar-background-color":{"value":"#333333","edited":false},"group-text-color":{"value":"#0eb8c0","edited":false},"group-border-color":{"value":"#555555","edited":false},"group-background-color":{"value":"#333333","edited":false},"widget-text-color":{"value":"#eeeeee","edited":false},"widget-background-color":{"value":"#097479","edited":false},"widget-border-color":{"value":"#333333","edited":false},"base-font":{"value":"-apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Oxygen-Sans,Ubuntu,Cantarell,Helvetica Neue,sans-serif"}}, "angularTheme":{"primary":"indigo","accents":"blue","warn":"red","background":"grey"}}, "site":{"name":"C.A.M.A - Controle"},"hideToolbar":"false","allowSwipe":"false","dateFormat":"DD/MM/YYYY","sizes":{"sx":48,"sy":48,"gx":6,"gy":6,"cx":6,"cy":6,"px":0,"py":0},"allowTempTheme":"fal
```

se"}}, {"id": "953a3e33.6f214", "type": "ui_group", "name": "Fluxo", "tab": "b6ead543.23e548", "order": 1, "disp": true, "width": "20", "collapse": false}, {"id": "452cced9.b0753", "type": "ui_group", "name": "Mostura", "tab": "b6ead543.23e548", "order": 2, "disp": true, "width": "20", "collapse": false}, {"id": "975bdb91.989a98", "type": "ui_group", "name": "Lavagem", "tab": "b6ead543.23e548", "order": 3, "disp": true, "width": "20", "collapse": false}, {"id": "801d0619.8aad68", "type": "ui_group", "name": "Fermentação /
Maturação", "tab": "b6ead543.23e548", "order": 6, "disp": true, "width": "20", "collapse": false}, {"id": "8927d5b.74d4128", "type": "ui_group", "name": "Fervura", "tab": "b6ead543.23e548", "order": 4, "disp": true, "width": "20", "collapse": false}, {"id": "16108a45.798416", "type": "ui_group", "name": "Resfriamento", "tab": "b6ead543.23e548", "order": 5, "disp": true, "width": "20", "collapse": false}, {"id": "f9e4442f8bcc22f1", "type": "ui_spacer", "name": "spacer", "group": "953a3e33.6f214", "order": 4, "width": 2, "height": 1}, {"id": "1852b37ffd5f865a", "type": "ui_spacer", "name": "spacer", "group": "953a3e33.6f214", "order": 8, "width": 2, "height": 1}, {"id": "2aec8fc93053574a", "type": "ui_spacer", "name": "spacer", "group": "953a3e33.6f214", "order": 10, "width": 14, "height": 1}, {"id": "96064c3d3b345b71", "type": "ui_spacer", "name": "spacer", "group": "975bdb91.989a98", "order": 6, "width": 8, "height": 1}, {"id": "e55a431d447f6d43", "type": "ui_spacer", "name": "spacer", "group": "975bdb91.989a98", "order": 7, "width": 8, "height": 1}, {"id": "f2dc13250634a2c3", "type": "ui_spacer", "name": "spacer", "group": "975bdb91.989a98", "order": 8, "width": 8, "height": 1}, {"id": "675cecfdd5541195", "type": "ui_spacer", "name": "spacer", "group": "8927d5b.74d4128", "order": 6, "width": 8, "height": 1}, {"id": "b38662e8c5e7d475", "type": "ui_spacer", "name": "spacer", "group": "8927d5b.74d4128", "order": 7, "width": 8, "height": 1}, {"id": "67242528c7f364a4", "type": "ui_spacer", "name": "spacer", "group": "8927d5b.74d4128", "order": 8, "width": 8, "height": 1}, {"id": "9c20a461d494d4ff", "type": "ui_spacer", "name": "spacer", "group": "16108a45.798416", "order": 7, "width": 8, "height": 1}, {"id": "4447a144d281e9ae", "type": "ui_spacer", "name": "spacer", "group": "16108a45.798416", "order": 8, "width": 8, "height": 1}, {"id": "d21cac160f2e4884", "type": "ui_spacer", "name": "spacer", "group": "801d0619.8aad68", "order": 6, "width": 8, "height": 1}, {"id": "c899de73ef7fea0b", "type": "ui_spacer", "name": "spacer", "group": "801d0619.8aad68", "order": 7, "width": 8, "height": 1}, {"id": "95a060c7843c2de1", "type": "ui_spacer", "name": "spacer", "group": "801d0619.8aad68", "order": 8, "width": 8, "height": 1}, {"id": "b6ead543.23e548", "type": "ui_tab", "name": "C.A.M.A
-
Controle", "icon": "computer", "order": 1, "disabled": false, "hidden": false}, {"id": "131c002.be685", "type": "ui_switch", "z": "29e0e22c.ba14ae", "name": "", "label": "Bomba
Mostura", "tooltip": "", "group": "953a3e33.6f214", "order": 1, "width": 6, "height": 1, "passthru":


```

"x":150,"y":480,"wires":[["9356709a.c45c5"]],{"id":"c5df4c38.b4c3a","type":"ui_gauge",
,"z":"7999b7cf.b7c508","name":"","group":"452cced9.b0753","order":3,"width":4,"height":3,"gtype":"gage","title":"Média","label":"","format":"{{msg.media}}","min":0,"max":100,"colors":["#00b500","#e6e600","#ca3838"],"seg1":"25","seg2":"70","x":950,"y":180,"wires":[]},{"id":"f855acf9.58a74","type":"ui_gauge","z":"7999b7cf.b7c508","name":"","group":"452cced9.b0753","order":2,"width":4,"height":3,"gtype":"gage","title":"Objetivo","label":"","format":"{{msg.payload}}","min":20,"max":90,"colors":["#ffffae","#dddd00","#ab4a4a"],"seg1":"40","seg2":"70","x":960,"y":280,"wires":[]},{"id":"26286a38.1effa6","type":"ui_toast","z":"7999b7cf.b7c508","position":"dialog","displayTime":3,"highlight":"","sendall":true,"outputs":1,"ok":"Sim","cancel":"Não","raw":false,"topic":"","name":"Continuar?","x":570,"y":480,"wires":[["cc47cef2.a6d5b"]],{"id":"9356709a.c45c5","type":"function","z":"7999b7cf.b7c508","name":"A2","func":"if (msg.payload === true) {\nmsg.topic = 'Atenção!';\n msg.payload = 'Antes de ligar a resistência do tanque de mostura certifique-se de que ela está submersa! \\\n Deseja continuar ?';\n return [undefined, msg];\n}else if (msg.payload === false){\n flow.set('res', false);\n msg = {topic: 'enable', payload: 0};\n return [msg, undefined];\n}","outputs":2,"noerr":0,"initialize":"","finalize":"","x":330,"y":480,"wires":[["96338031.46704"],["26286a38.1effa6"]],{"id":"cc47cef2.a6d5b","type":"function","z":"7999b7cf.b7c508","name":"Validação","func":"if (msg.payload === 'Sim') {\nflow.set('res', true);\n msg = {topic: 'enable', payload: 1};\n return [undefined, msg];\n}\nelse if (msg.payload === 'Não'){ \n msg.payload = false;\n flow.set('res', false);\n return [msg, undefined];\n}\n\n","outputs":2,"noerr":0,"initialize":"","finalize":"","x":580,"y":580,"wires":[["4122dcf5.4829d4"],["96338031.46704"]],{"id":"f5ba8b55.607a78","type":"ui_button","z":"29e0e22c.ba14ae","name":"","group":"953a3e33.6f214","order":11,"width":20,"height":1,"passthru":false,"label":"Reset","tooltip":"","color":"","bgcolor":"","icon":"","payload":1,"payloadType":"num","topic":"reset","x":230,"y":500,"wires":[["131c002.be685","e140ac71.208cc","c8bafe51.cf094","91b0c2bc.d2ca2","738665e.7d08f9c","4da98108.0076f","407c3fc9.18db7","38761508.c5d6ba","9d79bec8.09817","70d0a1c9.9709","81a4639a.e03d6","b50ec7db.443448","c0c12183.280fc","36862c58.a96a44"]],{"id":"83234576.3914d8","type":"ui_gauge","z":"7999b7cf.b7c508","name":"","group":"452cced9.b0753","order":4,"width":4,"height":3,"gtype":"gage","title":"Topo","label":"","format":"{{msg.topo}}","min":0,"max":100,"colors":["#46c85c","#2b908f","#ab4a4a"],"seg1":"25","seg2":"70","x":950,"y":100,"wires":[]},{"id":"57590844.2aa108","type":"ui_gauge","z":"7999b7c

```

```
f.b7c508", "name": "", "group": "452cced9.b0753", "order": 5, "width": 4, "height": 3, "gtype": "g
age", "title": "Fundo", "label": "", "format": "{ {msg.fundo} }", "min": 0, "max": "100", "colors": ["
#00b500", "#aac975", "#ab4a4a"], "seg1": "25", "seg2": "70", "x": 950, "y": 140, "wires": []}, {"id
": "8f3ea0b6.f9d0b", "type": "function", "z": "c5dc1810.861398", "name": "On/Off", "func": "//p
ersistindo o setpoint fornecido\nif (msg.topic === 'setpoint' && msg.payload !== null){\n
flow.set('setpoint', msg.payload)\n} else if(msg.topic === 'setpoint' && msg.payload ===
null){\n  flow.set('setpoint', -100)\n}\n\n//caso não esteja definido setpoint apenas ligar o
refrigerador\nif(flow.get('setpoint') === undefined){\n\n  flow.set('setpoint', -
100)\n}\n\n//intervalo on/off\nvar intervalo = 1;\n\nif (flow.get('res')) {\n  //desliga\n  if
(msg.payload < flow.get('setpoint') - intervalo) {\n    msg.payload = 1;\n  }\n  //liga\n
else if (msg.payload > flow.get('setpoint') + intervalo) {\n    msg.payload = 0;\n  }\n
return  msg;\n}\nelse  {\n\n  msg.payload  =  1\n\n  return
msg;\n}", "outputs": 1, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 910, "y": 420, "wires"
: [{"2fe8b8453c1acf2d"}]}, {"id": "8aacd062.f144f", "type": "join", "z": "7999b7cf.b7c508", "n
ame": "", "mode": "custom", "build": "object", "property": "payload", "propertyType": "msg", "k
ey": "topic", "joiner": "\n", "joinerType": "str", "accumulate": false, "timeout": "3", "count": "2", "
reduceRight": false, "reduceExp": "", "reduceInit": "", "reduceInitType": "", "reduceFixup": "", "
x": 630, "y": 60, "wires": [{"75c46bfe09c21a66"}]}, {"id": "6cff43b9.93bd1c", "type": "ui_templ
ate", "z": "7999b7cf.b7c508", "group": "452cced9.b0753", "name": "Gráfico", "order": 1, "width
": "12", "height": 8, "format": "<script>//httpStatic:
/home/pi/.node-
red/static*/></script>\n<script
src=/Highcharts-
9.2.2/code/highcharts.js"></script>\n<script
src=/Highcharts-
9.2.2/code/modules/exporting.js"></script>\n<script
src=/Highcharts-
9.2.2/code/modules/export-data.js"></script>\n<script
src=/Highcharts-
9.2.2/code/themes/dark-unica.js"></script>\n<div
id="container"
width:auto
margin:0></div>\n
\n<script>\nvar
width=
$("#container").width();\n
(function(scope){\n
scope.$watch('msg', function(msg) {\n
var series =
chart.series;\n
for(i=0; i<series.length; i++) {\n
series[i].checkbox.style.marginLeft
= '-2.5%';\n
});\n
chart.series[0].setData(msg.topo);\n
chart.series[1].setData(msg.fundo);\n
chart.series[2].setData(msg.media);\n
});\n
})(scope);\nvar chart = new
Highcharts.chart('container', {\n
chart: {\n
animation: false,\n
zoomType: 'x',\n
type: 'spline'\n
},\n
title: {\n
text: 'Tanque de Mostura'\n
},\n
subtitle: {\n
text:
'32L'\n
},\n
credits: { enabled: false },\n
xAxis: [{\n
type: 'datetime',\n

```

```

dateTimeLabelFormats: { // don't display the dummy year\n      \tsecond: '%H:%M:%S'\n
},\n      showFirstLabel: false,\n      title: {\n          text: 'Tempo Decorrido'\n      }\n
}],\n      yAxis: {\n          title: {\n              text: 'Temperatura (°C)'\n          },\n          tooltip: {\n              headerFormat: '<b>{point.x:Tempo decorrido: %H:%M:%S}</b></br>'\n              pointFormat: '{series.name}:<b style=\"text-align:center;\">{point.y} °C</b></br>'\n              valueDecimals:3,\n              shared: true,\n              useHTML: true\n          },\n          plotOptions: {\n              spline: {\n                  marker: {\n                      enabled: false\n                  }\n              },\n              series: {\n                  turboThreshold: '0',\n                  showCheckbox: true,\n                  events: {\n                      checkboxClick: function (event) {\n                          if (event.checked) {\n                              this.show();\n                          } else {\n                              this.hide();\n                          }\n                      },\n                      legendItemClick: function() {\n                          return false;\n                      }\n                  }\n              },\n              series: [{name:'Topo ',selected: true},{name:'Fundo', selected: true},{name:'Média', selected: true}]\n          }\n      });\n</script>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":false,"templateScope":"local","x":960,"y":60,"wires":[[]]},{"id":"e403b88e.9cfcb8","type":"function","z":"7999b7cf.b7c508","name":"A1","func":"var value = Math.round(msg.payload * 10)/10\n\nif ((value >= 90 || value <= 20) && msg.payload !== null){\n  msg = {payload: null};\n  msg1 = {payload: \"Por favor, selecione uma temperatura entre 20 °C e 90°C\"};\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\nelse if (msg.payload !== null){\n  msg = {payload: value};\n  msg1 = {payload: \"Temperatura de \" + value + \" °C selecionada!\"};\n  msg2 = {payload: value + ' °C', setpoint: value};\n  msg3 = {payload: value, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\nelse if (msg.payload === null && msg.topic === 'reset'){ \n  msg = {payload: null}\n  msg1 = {payload: \"Reset realizado!\"};\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\nelse if (msg.payload === null){\n  msg = {payload: null}\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, undefined, msg2, msg3];\n}\n\n","outputs":4,"noerr":0,"initialize":"","finalize":"","libs":[],"x":710,"y":240,"wires":[["a63f0a5d.e12b48"],["68fbe8a.5fcb118"],["f855acf9.58a74"],["96338031.46704"]],{"id":"a63f0a5d.e12b48","type":"ui_text_input","z":"7999b7cf.b7c508","name":"","label":"Objetivo (Enter)","tooltip":"Selecione uma temperatura entre 20°C e 90°C","group":"452ccd9.b0753","order":6,"width":8,"height":1,"passthru":false,"mode":"number","delay":"0","topic":"topic","x":660,"y":340,"wires":[["e403b88e.9cfcb8"]]},{"id":"68fbe8a.5fcb118","type":"ui_toast","z":"7999b7cf.b7c508","position":"bottom

```

```

right", "displayTime": "3", "highlight": "", "sendall": true, "outputs": 0, "ok": "OK", "cancel": "", "raw": true, "topic": "", "name": "", "x": 990, "y": 240, "wires": [], {"id": "96338031.46704", "type": "PID", "z": "7999b7cf.b7c508", "name": "", "setpoint": "0", "pb": "5.0", "ti": "300", "td": 0, "integral_default": "0", "smooth_factor": 3, "max_interval": 600, "enable": "0", "disabled_op": 0, "x": 910, "y": 420, "wires": [{"03d0d522e8bbbeac"}]}, {"id": "6ecbe348.c7d3cc", "type": "ui_button", "z": "d880af12.1b24f", "name": "", "group": "975bdb91.989a98", "order": 9, "width": 0, "height": 0, "passthru": false, "label": "Reset", "tooltip": "", "color": "", "bgcolor": "", "icon": "", "payload": "null", "payloadType": "str", "topic": "", "x": 96.8968505859375, "y": 240.18256378173828, "wires": [{"43214368fa8e6305"}]}, {"id": "730482c9.f4c56c", "type": "ui_switch", "z": "d880af12.1b24f", "name": "", "label": "Ligar Resistência", "tooltip": "", "group": "975bdb91.989a98", "order": 5, "width": 8, "height": 1, "passthru": true, "decouple": "false", "topic": "", "style": "", "onvalue": "true", "onvalueType": "bool", "onicon": "", "oncolor": "", "offvalue": "false", "offvalueType": "bool", "officon": "", "offcolor": "", "x": 150, "y": 480, "wires": [{"3fd1c369.d3e6dc"}]}, {"id": "5a49ca8f.b30214", "type": "ui_gauge", "z": "d880af12.1b24f", "name": "", "group": "975bdb91.989a98", "order": 3, "width": 4, "height": 3, "gtype": "gage", "title": "Média", "label": "", "format": "{{ msg.media }}", "min": 0, "max": 100, "colors": ["#00b500", "#e6e600", "#ca3838"], "seg1": "25", "seg2": "70", "x": 950, "y": 180, "wires": [], {"id": "b699d93f.f271b8", "type": "ui_gauge", "z": "d880af12.1b24f", "name": "", "group": "975bdb91.989a98", "order": 2, "width": 4, "height": 3, "gtype": "gage", "title": "Objetivo", "label": "", "format": "{{ msg.payload }}", "min": 20, "max": 90, "colors": ["#ffffae", "#ddd00", "#ab4a4a"], "seg1": "40", "seg2": "70", "x": 960, "y": 280, "wires": [], {"id": "3dc183a2.e60b1c", "type": "ui_toast", "z": "d880af12.1b24f", "position": "dialog", "displayTime": "3", "highlight": "", "sendall": true, "outputs": 1, "ok": "Sim", "cancel": "Não", "raw": false, "topic": "", "name": "Continuar?", "x": 570, "y": 480, "wires": [{"4e15b098.bdccf"}]}, {"id": "3fd1c369.d3e6dc", "type": "function", "z": "d880af12.1b24f", "name": "A2", "func": "if (msg.payload === true) {\n  msg.topic = 'Atenção!';\n  msg.payload = 'Antes de ligar a resistência do tanque de mostura certifique-se de que ela está submersa! \\n Deseja continuar?';\n  return [undefined, msg];\n} else if (msg.payload === false) {\n  flow.set('res', false);\n  msg = {topic: 'enable', payload: 0};\n  return [msg, undefined];\n}\n\n", "outputs": 2, "noerr": 0, "initialize": "", "finalize": "", "x": 330, "y": 480, "wires": [{"9602ec52.6caf"}, {"3dc183a2.e60b1c"}]}, {"id": "4e15b098.bdccf", "type": "function", "z": "d880af12.1b24f", "name": "Validação", "func": "if (msg.payload === 'Sim') {\n  flow.set('res', true);\n  msg = {topic: 'enable', payload: 1};\n  return [undefined, msg];\n} else if (msg.payload === 'Não') {\n  msg.payload = false;\n}

```

```

flow.set('res',false);\n                                return [msg,
undefined];\n)\n\n","outputs":2,"noerr":0,"initialize":"","finalize":"","x":580,"y":580,"wire
s":[[{"730482c9.f4c56c"},{"9602ec52.6caf"}]],{"id":"6dfefdbe.8f3c24","type":"ui_template
","z":"d880af12.1b24f","group":"975bdb91.989a98","name":"Gráfico","order":1,"width":1
2,"height":8,"format":"<script>//httpStatic: '/home/pi/.node-red/static*/></script>\n<script
src=\"/Highcharts-9.2.2/code/highcharts.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/modules/exporting.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/modules/export-data.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/themes/dark-unica.js\"></script>\n<div id=\"container2\" width: auto
margin:0></div>\n \n<script>\nvar width2 = $('#container2').width();\n
(function(scope){\n scope.$watch('msg', function(msg) {\n var series2 =
chart2.series;\n for(i=0; i<series2.length; i++) {\n
series2[i].checkbox.style.marginLeft = '-2.5%';\n });\n
chart2.series[0].setData(msg.media);\n });\n })(scope);\nvar chart2 = new
Highcharts.chart('container2', {\n chart: {\n animation: false,\n zoomType: 'x',\n
type: 'spline'\n },\n title: {\n text: 'Água para Lavagem'\n },\n subtitle: {\n text:
'15L'\n },\n credits: { enabled: false },\n xAxis: [{\n type: 'datetime',\n
dateTimeLabelFormats: { // don't display the dummy year\n \tsecond: '%H:%M:%S'\n
},\n showFirstLabel: false,\n title: {\n text: 'Tempo Decorrido'\n }\n
}],\n yAxis: {\n title: {\n text: 'Temperatura (°C)'\n },\n },\n tooltip: {\n
headerFormat:'<b>{point.x:Tempo decorrido: %H:%M:%S}</b><br>',\n pointFormat:
'{series.name}:<b style=\"text-align:center;\">{point.y} °C</b><br>',\n
valueDecimals:3,\n shared: true,\n useHTML: true\n },\n plotOptions: {\n
spline: {\n marker: {\n enabled: false\n }\n },\n series: {\n
turboThreshold: '0',\n showCheckbox: true,\n events: {\n checkboxClick:
function (event) {\n if (event.checked) {\n this.show();\n }
else {\n this.hide();\n }\n },\n legendItemClick:
function() {\n return false;\n }\n }\n },\n series:
[{\n name:'Média',\n selected: true\n }]\n
});\n</script>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":false,"t
emplateScope":"local","x":960,"y":60,"wires":[]]],{"id":"2925819a.f5e19e","type":"funct
ion","z":"d880af12.1b24f","name":"A1","func":"var value = Math.round(msg.payload *
10)/10\n\nif ((value >= 90 || value <= 20) && msg.payload !== null){\n msg = {payload:
null};\n msg1 = {payload: \"Por favor, selecione uma temperatura entre 20 °C e 90°C\"};\n

```

```

msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg,
msg1, msg2, msg3];\n  \n} else if (msg.payload !== null){\n  msg = {payload: value};\n
msg1 = {payload: \"Temperatura de \" + value + \" °C selecionada!\"};\n  msg2 = {payload:
value + ' °C', setpoint: value};\n  msg3 = {payload: value, topic: 'setpoint'};\n  return [msg,
msg1, msg2, msg3];\n} else if (msg.payload === null && msg.topic === 'reset'){
msg = {payload: null}\n  msg1 = {payload: \"Reset realizado!\"};\n  msg2 = {payload: \"No
Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, msg1, msg2,
msg3];\n}\nelse if (msg.payload === null){\n  msg = {payload: null}\n  msg2 = {payload:
\"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, undefined, msg2,
msg3];\n}\n\n,\"outputs\":4,\"noerr\":0,\"initialize\":\"\",\"finalize\":\"\",\"x\":710,\"y\":240,\"wires\":[[\"7
caee071.6ecd8\"],[\"14c0e683.829a79\"],[\"b699d93f.f271b8\"],[\"9602ec52.6caf\"]],{\"id\":\"7c
aee071.6ecd8\",\"type\":\"ui_text_input\",\"z\":\"d880af12.1b24f\",\"name\":\"\",\"label\":\"Objetivo
(Enter)\",\"tooltip\":\"Selecione uma temperatura entre 20°C e
90°C\",\"group\":\"975bdb91.989a98\",\"order\":4,\"width\":8,\"height\":1,\"passthru\":false,\"mode
\":\"number\",\"delay\":\"0\",\"topic\":\"topic\",\"x\":660,\"y\":340,\"wires\":[[\"2925819a.f5e19e\"]],{
\"id\":\"14c0e683.829a79\",\"type\":\"ui_toast\",\"z\":\"d880af12.1b24f\",\"position\":\"bottom
right\",\"displayTime\":\"3\",\"highlight\":\"\",\"sendall\":true,\"outputs\":0,\"ok\":\"OK\",\"cancel\":\"\",
\"raw\":true,\"topic\":\"\",\"name\":\"\",\"x\":990,\"y\":240,\"wires\":[]},{\"id\":\"9602ec52.6caf\",\"type\":\"P
ID\",\"z\":\"d880af12.1b24f\",\"name\":\"\",\"setpoint\":\"0\",\"pb\":\"5.0\",\"ti\":\"300\",\"td\":0,\"integral_d
efault\":\"0\",\"smooth_factor\":3,\"max_interval\":600,\"enable\":\"0\",\"disabled_op\":0,\"x\":910,\"y
\":420,\"wires\":[[\"a21c298f51851d76\"]],{\"id\":\"49afa6be.d35828\",\"type\":\"ui_button\",\"z\":\"
31c8b922.7219c6\",\"name\":\"\",\"group\":\"8927d5b.74d4128\",\"order\":9,\"width\":0,\"height\":0,
\"passthru\":false,\"label\":\"Reset\",\"tooltip\":\"\",\"color\":\"\",\"bgcolor\":\"\",\"icon\":\"\",\"payload\":\"n
ull\",\"payloadType\":\"str\",\"topic\":\"\",\"x\":96.8968505859375,\"y\":240.18256378173828,\"wir
es\":[[\"cf8dece7970850aa\"]],{\"id\":\"151a6dc8.c323b2\",\"type\":\"ui_switch\",\"z\":\"31c8b922.
7219c6\",\"name\":\"\",\"label\":\"Ligar
Resistência\",\"tooltip\":\"\",\"group\":\"8927d5b.74d4128\",\"order\":5,\"width\":8,\"height\":1,\"passt
hru\":true,\"decouple\":\"false\",\"topic\":\"\",\"style\":\"\",\"onvalue\":\"true\",\"onvalueType\":\"bool\",
\"onicon\":\"\",\"oncolor\":\"\",\"offvalue\":\"false\",\"offvalueType\":\"bool\",\"officon\":\"\",\"offcolor\":\"
\",\"x\":150,\"y\":480,\"wires\":[[\"ec201a.012ccfe8\"]],{\"id\":\"a108bdfb.d9a8f\",\"type\":\"ui_gauge
\",\"z\":\"31c8b922.7219c6\",\"name\":\"\",\"group\":\"8927d5b.74d4128\",\"order\":3,\"width\":4,\"hei
ght\":3,\"gtype\":\"gage\",\"title\":\"Média\",\"label\":\"\",\"format\":\"{ msg.media }\",\"min\":0,\"max
\":\"100\",\"colors\":[\"#00b500\",\"#e6e600\",\"#ca3838\"],\"seg1\":\"25\",\"seg2\":\"70\",\"x\":950,\"y\":18
0,\"wires\":[]},{\"id\":\"37afdd08.42fe52\",\"type\":\"ui_gauge\",\"z\":\"31c8b922.7219c6\",\"name\":\"

```

```

", "group": "8927d5b.74d4128", "order": 2, "width": 4, "height": 3, "gtype": "gage", "title": "Objetivo", "label": "", "format": "{ {msg.payload} }", "min": "20", "max": "90", "colors": [ "#ffffae", "#ddd000", "#ab4a4a" ], "seg1": "40", "seg2": "70", "x": 960, "y": 280, "wires": [ ], { "id": "3bc2f75b.8ff6f8", "type": "ui_toast", "z": "31c8b922.7219c6", "position": "dialog", "displayTime": "3", "highlight": "", "sendall": true, "outputs": 1, "ok": "Sim", "cancel": "N\u00e3o", "raw": false, "topic": "", "name": "Continuar?", "x": 570, "y": 480, "wires": [ [ "803d6b76.a705d8" ] ] }, { "id": "ec201a.012ccfe8", "type": "function", "z": "31c8b922.7219c6", "name": "A2", "func": "if (msg.payload === true) {\n  msg.topic = 'Aten\u00e7\u00e3o!';\n  msg.payload = 'Antes de ligar a resist\u00eancia do tanque de fervura certifique-se de que ela est\u00e1 submersa! \\n Deseja continuar ?';\n  return [undefined, msg];\n} else if (msg.payload === false & flow.get('res') === true) {\n  flow.set('res', false);\n  msg = {topic: 'enable', payload: 0};\n  return [msg, undefined];\n}\n} else if (msg.payload === false & flow.get('res') === false) {\n  //retorno da valida\u00e7\u00e3o\n  //ignorar, n\u00e3o retornar\n  msg = {topic: 'enable', payload: 0};\n  return [msg, undefined];\n}\n\n", "outputs": 2, "noerr": 0, "initialize": "", "finalize": "", "x": 330, "y": 480, "wires": [ [ "33524f05.eeaf2" ], [ "3bc2f75b.8ff6f8" ] ] }, { "id": "803d6b76.a705d8", "type": "function", "z": "31c8b922.7219c6", "name": "Valida\u00e7\u00e3o", "func": "if (msg.payload === 'Sim') {\n  flow.set('res', true);\n  msg = {topic: 'enable', payload: 1};\n  return [undefined, msg];\n}\n} else if (msg.payload === 'N\u00e3o') {\n  msg.payload = false;\n  flow.set('res', false);\n  return [msg, undefined];\n}\n\n", "outputs": 2, "noerr": 0, "initialize": "", "finalize": "", "x": 580, "y": 580, "wires": [ [ "151a6dc8.c323b2" ], [ "33524f05.eeaf2" ] ] }, { "id": "d87546c6.8df2a8", "type": "function", "z": "31c8b922.7219c6", "name": "A1", "func": "var value = Math.round(msg.payload * 10)/10\n\nif ((value >= 90 || value <= 20) && msg.payload !== null) {\n  msg = {payload: null};\n  msg1 = {payload: \"Por favor, selecione uma temperatura entre 20 \u00b0C e 90\u00b0C\"};\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\n} else if (msg.payload !== null) {\n  msg = {payload: value};\n  msg1 = {payload: \"Temperatura de \" + value + \" \u00b0C selecionada!\"};\n  msg2 = {payload: value + ' \u00b0C', setpoint: value};\n  msg3 = {payload: value, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\n} else if (msg.payload === null && msg.topic === 'reset') {\n  msg = {payload: null};\n  msg1 = {payload: \"Reset realizado!\"};\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, msg1, msg2, msg3];\n}\n\n} else if (msg.payload === null) {\n  msg = {payload: null};\n  msg2 = {payload: \"No Value\"};\n  msg3 = {payload: 0, topic: 'setpoint'};\n  return [msg, undefined, msg2, msg3];\n}\n\n", "outputs": 4, "noerr": 0, "initialize": "", "finalize": "", "x": 710, "y": 240, "wires": [ [ "6"

```

```

3e0f0dd.9b48e"],["c93fa0e5.8b56b"],["37afdd08.42fe52"],["33524f05.eeaf2"]]],{"id":"63
e0f0dd.9b48e","type":"ui_text_input","z":"31c8b922.7219c6","name":"","label":"Objetivo
(Enter)","tooltip":"Seleccione una temperatura entre 20°C e
90°C","group":"8927d5b.74d4128","order":4,"width":8,"height":1,"passthru":false,"mode"
:"number","delay":"0","topic":"topic","x":660,"y":340,"wires":[["d87546c6.8df2a8"]]],{"i
d":"c93fa0e5.8b56b","type":"ui_toast","z":"31c8b922.7219c6","position":"bottom
right","displayTime":"3","highlight":"","sendall":true,"outputs":0,"ok":"OK","cancel":"","r
aw":true,"topic":"","name":"","x":990,"y":240,"wires":[]},{ "id":"33524f05.eeaf2","type":"
PID","z":"31c8b922.7219c6","name":"","setpoint":"0","pb":"5.0","ti":"300","td":0,"integra
l_default":"0","smooth_factor":3,"max_interval":600,"enable":"0","disabled_op":0,"x":910
,"y":420,"wires":[["fd8dbca3a1c25cfd"]]],{"id":"e7c51b68.6a4688","type":"ui_template",
"z":"31c8b922.7219c6","group":"8927d5b.74d4128","name":"Gráfico","order":1,"width":
12,"height":8,"format":"<script>/httpStatic:
red/static*/></script>\n<script
9.2.2/code/highcharts.js\"></script>\n<script
9.2.2/code/modules/exporting.js\"></script>\n<script
9.2.2/code/modules/export-data.js\"></script>\n<script
9.2.2/code/themes/dark-unica.js\"></script>\n<div id=\"container3\" width: auto
margin:0></div>\n \n<script>\nvar width3 = $(\"#container3\").width();\n
(function(scope){\n scope.$watch('msg', function(msg) {\n var series3 =
chart3.series;\n for(i=0; i<series3.length; i++) {\n
series3[i].checkbox.style.marginLeft = '-2.5%';\n
});\n chart3.series[0].setData(msg.media);\n
})(scope);\nvar chart3 = new
Highcharts.chart('container3', {\n chart: {\n animation: false,\n zoomType: 'x',\n
type: 'spline'\n },\n title: {\n text: 'Tanque de Fervura'\n },\n subtitle: {\n text:
'32L'\n },\n credits: { enabled: false },\n xAxis: [{\n type: 'datetime',\n
dateTimeLabelFormats: { // don't display the dummy year\n \tsecond: '%H:%M:%S'\n
},\n showFirstLabel: false,\n title: {\n text: 'Tempo Decorrido'\n
}],\n yAxis: {\n title: {\n text: 'Temperatura (°C)'\n },\n },\n tooltip: {\n
headerFormat:'<b>{point.x:Tempo decorrido: %H:%M:%S}</b><br>',\n pointFormat:
'{series.name}:<b style=\"text-align:center;\">{point.y} °C</b><br>',\n
valueDecimals:3,\n shared: true,\n useHTML: true\n },\n plotOptions: {\n
spline: {\n marker: {\n enabled: false\n }
},\n series: {\n
turboThreshold: '0',\n showCheckbox: true,\n events: {\n checkboxClick:

```

```

function (event) {\n
    if (event.checked) {\n
        this.show();\n
    }
else {\n
    this.hide();\n
    },\n
    legendItemClick:
function() {\n
    return false;\n
    },\n
    series:
[{\n
    name:'Média',
    selected:
true}],\n
});\n</script>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":false,"t
emplateScope":"local","x":960,"y":60,"wires":[]],{"id":"527b8daa.aede94","type":"funct
ion","z":"c5dc1810.861398","name":"Reiniciar","func":"flow.set('media',null);\nflow.set('
old',null);\n\nmsg = {payload: "", media: []};\nmsg1 = {topic: 'reset', payload: null, topo:
\n'No Value\n', media: \n'No Value\n', fundo: \n'No Value\n'};\nmsg2 = {payload:
false};\n\nreturn
[msg,
msg1,
msg2];","outputs":3,"noerr":0,"initialize":"","finalize":"","libs":[],"x":260,"y":240,"wires":
[["36486e36c566bfc8","a84f3936.38a9c8"],["74d80454.8f544c","dbf13284.aef91","eceac
10e.a944f"],["62d6aa19.e03874"]],{"id":"3c0c9714.28a998","type":"ui_button","z":"c5d
c1810.861398","name":"","group":"801d0619.8aad68","order":9,"width":0,"height":0,"pas
sthu":false,"label":"Reset","tooltip":"","color":"","bgcolor":"","icon":"","payload":"null",
payloadType":"str","topic":"","x":96.8968505859375,"y":240.18256378173828,"wires":[["
527b8daa.aede94"]],{"id":"62d6aa19.e03874","type":"ui_switch","z":"c5dc1810.861398
","name":"","label":"Ligar
Refrigerador","tooltip":"","group":"801d0619.8aad68","order":5,"width":8,"height":1,"pas
sthu":true,"decouple":"false","topic":"","style":"","onvalue":"true","onvalueType":"bool",
"onicon":"","oncolor":"","offvalue":"false","offvalueType":"bool","officon":"","offcolor":
","x":150,"y":480,"wires":[["e5d4a459.2bb798"]],{"id":"74d80454.8f544c","type":"ui_ga
uge","z":"c5dc1810.861398","name":"","group":"801d0619.8aad68","order":3,"width":4,"
height":3,"gtype":"gage","title":"Média","label":"","format":"{{msg.media}}","min":0,"m
ax":30,"colors":["#00b500","#e6e600","#ca3838"],"seg1":25,"seg2":70,"x":950,"y":
180,"wires":[]},{"id":"c33821aa.a9d1d","type":"ui_gauge","z":"c5dc1810.861398","name
":"","group":"801d0619.8aad68","order":2,"width":4,"height":3,"gtype":"gage","title":"Ob
jetivo","label":"","format":"{{msg.payload}}","min":0,"max":30,"colors":["#ffffae","#
dddd00","#ab4a4a"],"seg1":40,"seg2":70,"x":960,"y":280,"wires":[]},{"id":"a64277b.f
e4e588","type":"ui_toast","z":"c5dc1810.861398","position":"dialog","displayTime":3,"
highlight":"","sendall":true,"outputs":1,"ok":"Sim","cancel":"Não","raw":false,"topic":"","
name":"Continuar?","x":590,"y":520,"wires":[["19c24fed.cbb4"]],{"id":"e5d4a459.2bb79
8","type":"function","z":"c5dc1810.861398","name":"A2","func":"if (msg.payload ===
true) {\n
msg.topic = 'Atenção!';\n
msg.payload = 'Para o controle do refrigerador o

```

```

mantenha na potência máxima! \n Deseja continuar ?;\n return [undefined, msg];\n} else
if (msg.payload === false & flow.get('res') === true){\n flow.set('res',false);\n msg =
{topic: 'enable', payload: 0};\n return [msg, undefined];\n}\nelse if (msg.payload === false
& flow.get('res') === false){\n //retorno da validação\n //ignorar, não retornar\n msg =
{topic: 'enable', payload: 0};\n return [msg,
undefined];\n}\n\n", "outputs":2, "noerr":0, "initialize":"","finalize":"","libs":[, "x":330, "y":48
0, "wires":[[{"8f3ea0b6.f9d0b"}, {"a64277b.fe4e588"}]], {"id": "19c24fed.cbb4", "type": "funct
ion", "z": "c5dc1810.861398", "name": "Validação", "func": "if (msg.payload === 'Sim') {\n
flow.set('res',true);\n msg = {topic: 'enable', payload: 1};\n return [undefined,
msg];\n}\nelse if (msg.payload === 'Não'){
msg.payload = false;\n flow.set('res',false);\n return [msg,
undefined];\n}\n\n", "outputs":2, "noerr":0, "initialize":"","finalize":"","libs":[, "x":580, "y":6
20, "wires":[[{"62d6aa19.e03874"}, {"8f3ea0b6.f9d0b"}]], {"id": "dbf13284.aef91", "type": "fu
nction", "z": "c5dc1810.861398", "name": "A1", "func": "var value = Math.round(msg.payload
* 10)/10\n\nif ((value >= 30 || value <= 0) && msg.payload !== null){\n msg = {payload:
null};\n msg1 = {payload: \"Por favor, selecione uma temperatura entre 20 °C e 90°C\"};\n
msg2 = {payload: \"No Value\"};\n return [msg, msg1, msg2, undefined];\n } else if
(msg.payload !== null){\n msg = {payload: value};\n msg1 = {payload: \"Temperatura
de \" + value + \" °C selecionada!\"};\n msg2 = {payload: value + ' °C', setpoint: value};\n
msg3 = {payload: value, topic: 'setpoint'};\n return [msg, msg1, msg2, msg3];\n } else if
(msg.payload === null && msg.topic === 'reset'){
msg = {payload: null}\n msg1 =
{payload: \"Reset realizado!\"};\n msg2 = {payload: \"No Value\"};\n return [msg, msg1,
msg2,
undefined];\n}\n\n", "outputs":4, "noerr":0, "initialize":"","finalize":"","x":710, "y":240, "wires"
: [[{"eceac10e.a944f"}, {"de559dd2.be963"}, {"c33821aa.a9d1d"}, {"8f3ea0b6.f9d0b"}]], {"id":
"eceac10e.a944f", "type": "ui_text_input", "z": "c5dc1810.861398", "name": "", "label": "Objeti
vo (Enter)", "tooltip": "Selecione uma temperatura entre 0°C e
30°C", "group": "801d0619.8aad68", "order":4, "width":8, "height":1, "passthru":false, "mode"
:"number", "delay": "0", "topic": "topic", "x":660, "y":340, "wires": [{"dbf13284.aef91"}]}, {"id"
:"de559dd2.be963", "type": "ui_toast", "z": "c5dc1810.861398", "position": "bottom
right", "displayTime": "3", "highlight": "", "sendall": true, "outputs":0, "ok": "OK", "cancel": "", "r
aw": true, "topic": "", "name": "", "x":990, "y":240, "wires": []}, {"id": "a84f3936.38a9c8", "type"
:"ui_template", "z": "c5dc1810.861398", "group": "801d0619.8aad68", "name": "Gráfico", "ord
er":1, "width":12, "height":8, "format": "<script>//httpStatic:
/home/pi/.node-

```

```

red/static'*/></script>\n<script src=\"/Highcharts-
9.2.2/code/highcharts.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/modules/exporting.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/modules/export-data.js\"></script>\n<script src=\"/Highcharts-
9.2.2/code/themes/dark-unica.js\"></script>\n<div id=\"container5\" width: auto
margin:0></div>\n \n<script>\nvar width4 = $(\"#container5\").width();\n
(function(scope){\n scope.$watch('msg', function(msg) {\n var series5 =
chart5.series;\n for(i=0; i<series5.length; i++) {\n
series5[i].checkbox.style.marginLeft = '-2.5%';\n });\n
chart5.series[0].setData(msg.media);\n });\n })(scope);\nvar chart5 = new
Highcharts.chart('container5', {\n chart: {\n animation: false,\n zoomType: 'x',\n
type: 'spline'\n },\n title: {\n text: 'Femetação/Maturação'\n },\n subtitle: {\n
text: '32 L'\n },\n credits: { enabled: false },\n xAxis: [{\n type: 'datetime',\n
dateTimeLabelFormats: { // don't display the dummy year\n \tsecond: '%H:%M:%S'\n
},\n showFirstLabel: false,\n title: {\n text: 'Tempo Decorrido'\n }\n
}],\n yAxis: {\n title: {\n text: 'Temperatura (°C)'\n },\n },\n tooltip: {\n
headerFormat:'<b>{point.x:Tempo Decorrido: %H:%M:%S}</b></br>',\n pointFormat:
'{series.name}:<b style=\"text-align:center;\">{point.y} °C</b></br>',\n
valueDecimals:3,\n shared: true,\n useHTML: true\n },\n plotOptions: {\n
spline: {\n marker: {\n enabled: false\n }\n },\n series: {\n
turboThreshold: '0',\n showCheckbox: true,\n events: {\n checkboxClick:
function (event) {\n if (event.checked) {\n this.show();\n }
else {\n this.hide();\n }\n },\n legendItemClick:
function() {\n return false;\n }\n }\n },\n series:
[{\n name:'Média',\n selected: true\n }]\n
});\n</script>\",\"storeOutMessages\":true,\"fwdInMessages\":true,\"resendOnRefresh\":true,\"te
mplateScope\":\"local\", \"x\":960,\"y\":60,\"wires\":[[ ]],{\n \"id\":\"b69a8181.49a2\", \"type\":\"function
\", \"z\":\"fc31302a.f6743\", \"name\":\"On/Off\", \"func\":\"//persistindo o setpoint fornecido\nif
(msg.topic === 'setpoint'){ \n flow.set('setpoint', msg.payload)\n }\n\n//caso não esteja
definido setpoint mostrar aviso\nif(flow.get('setpoint') === undefined &&
(flow.get('res')||false)){ \n msg.payload = '<div align=\"center\">Por favor, defina a
temperatura objetivo!';\n return msg; \n }\n\n//intervalo on/off\nvar intervalo = 3;\n\nif
(flow.get('res')) {\n //diminua\n if (msg.payload < flow.get('setpoint') - intervalo) {\n
msg.payload = '<div align=\"center\">Aumente a vazão do mosto <br>'+\n 'ou

```

```

<br> diminua a vazão de água!</div>;\n      \n } \n //liga\n else if (msg.payload >
flow.get('setpoint') + intervalo) {\n      msg.payload = '<div align=\"center\">Diminua a
vazão do mosto <br>'+\n      'ou <br> aumente a vazão de água!</div>;\n      }\n
else {\n      msg.payload = '<div align=\"center\">Mantenha a vazão!</div>;\n      }\n return
msg;\n}","outputs":1,"noerr":0,"initialize":"","finalize":"","x":910,"y":420,"wires":[[["313a
c5ec.55af1a"]]],{"id":"5cff5e0d.ea979","type":"ui_button","z":"fc31302a.f6743","name":"
","group":"16108a45.798416","order":9,"width":0,"height":0,"passthru":false,"label":"Res
et","tooltip":"","color":"","bgcolor":"","icon":"","payload":"null","payloadType":"str","top
ic":"","x":96.8968505859375,"y":240.18256378173828,"wires":[[["8985e279906e575d"]]]
,{"id":"4cc829fa.956408","type":"ui_switch","z":"fc31302a.f6743","name":"","label":"Lig
ar
Resfriamento","tooltip":"","group":"16108a45.798416","order":5,"width":8,"height":1,"pa
ssthru":true,"decouple":"false","topic":"","style":"","onvalue":"true","onvalueType":"bool"
,"onicon":"","oncolor":"","offvalue":"false","offvalueType":"bool","officon":"","offcolor":
":"","x":150,"y":480,"wires":[[["9a4f9f47.0cef6"]]],{"id":"47a818af.f72688","type":"ui_gau
ge","z":"fc31302a.f6743","name":"","group":"16108a45.798416","order":3,"width":4,"hei
ght":3,"gtype":"gage","title":"Média","label":"","format":"{{ msg.media }}","min":0,"max"
:"100","colors":["#00b500","#e6e600","#ca3838"],"seg1":"25","seg2":"70","x":950,"y":18
0,"wires":[]},{ "id":"bb8c7b62.1d12a8","type":"ui_gauge","z":"fc31302a.f6743","name":"","
","group":"16108a45.798416","order":2,"width":4,"height":3,"gtype":"gage","title":"Objeti
vo","label":"","format":"{{ msg.payload }}","min":0,"max":30,"colors":["#ffffae","#dd
dd00","#ab4a4a"],"seg1":"40","seg2":"70","x":960,"y":280,"wires":[]},{ "id":"28de72fd.c9
63fe","type":"ui_toast","z":"fc31302a.f6743","position":"dialog","displayTime":3,"highl
ight":"","sendall":true,"outputs":1,"ok":"Sim","cancel":"Não","raw":true,"topic":"","name"
:"Continuar?","x":590,"y":520,"wires":[[["f1eb0c87.03859"]]],{"id":"9a4f9f47.0cef6","typ
e":"function","z":"fc31302a.f6743","name":"A2","func":"if (msg.payload === true) {\n
msg.topic = 'Atenção!';\n  msg.payload = \"1. Ligue cuidadosamente a corrente fria do
trocador na vazão adequada.<br>\"+\n      \"2. Ligue a corrente quente e siga o controle
do painel!<br>\"+\n      \"Deseja continuar?\";\n  return [undefined, msg];\n} else if
(msg.payload === false & flow.get('res') === true){\n  flow.set('res',false);\n  msg =
{topic: 'enable', payload: \"\"};\n  return [msg, undefined];\n}\nelse if (msg.payload ===
false & flow.get('res') === false){\n  //retorno da validação\n  //ignorar, não retornar\n
msg = {topic: 'enable', payload: \"\"};\n      return [msg,
undefined];\n}\n}","outputs":2,"noerr":0,"initialize":"","finalize":"","x":330,"y":480,"wires"

```

```

: [{"id": "f1eb0c87.03859", "type": "function", "z": "fc31302a.f6743", "name": "Validação", "func": "if (msg.payload === 'Sim') {\n flow.set('res',true);\n msg = {topic: 'enable', payload: 1};\n return [undefined, msg];\n}\nelse if (msg.payload === 'Não'){ \n msg.payload = false;\n flow.set('res',false);\n return [msg, undefined];\n}\n\n", "outputs": 2, "noerr": 0, "initialize": "", "finalize": "", "x": 580, "y": 620, "wires": [{"id": "513133ec.ab1adc", "type": "function", "z": "fc31302a.f6743", "name": "A1", "func": "var value = Math.round(msg.payload * 10)/10\n\nif ((value >= 30 || value <= 20) && msg.payload !== null){\n msg = {payload: null};\n msg1 = {payload: \"Por favor, selecione uma temperatura entre 20 °C e 90°C\"};\n msg2 = {payload: \"No Value\"};\n return [msg, msg1, msg2, undefined];\n\n} else if (msg.payload !== null){\n msg = {payload: value};\n msg1 = {payload: \"Temperatura de \" + value + \" °C selecionada!\"};\n msg2 = {payload: value + ' °C', setpoint: value};\n msg3 = {payload: value, topic: 'setpoint'};\n return [msg, msg1, msg2, msg3];\n} else if (msg.payload === null && msg.topic === 'reset'){ \n msg = {payload: null}\n msg1 = {payload: \"Reset realizado!\"};\n msg2 = {payload: \"No Value\"};\n msg3 = {payload: undefined, topic: 'setpoint'};\n return [msg, msg1, msg2, msg3];\n} else if (msg.payload === null){\n msg = {payload: null}\n msg1 = {payload: \"\"};\n msg2 = {payload: \"No Value\"};\n msg3 = {payload: undefined, topic: 'setpoint'};\n return [msg, undefined, msg2, msg3];\n}\n\n", "outputs": 4, "noerr": 0, "initialize": "", "finalize": "", "x": 710, "y": 240, "wires": [{"id": "286968a0.af8828", "type": "ui_text_input", "z": "fc31302a.f6743", "name": "", "label": "Objetivo (Enter)", "tooltip": "Selecione uma temperatura entre 0°C e 30°C", "group": "16108a45.798416", "order": 4, "width": 8, "height": 1, "passthru": false, "mode": "number", "delay": "0", "topic": "topic", "x": 660, "y": 340, "wires": [{"id": "3bcde085.ef40c", "type": "ui_toast", "z": "fc31302a.f6743", "position": "bottom right", "displayTime": "3", "highlight": "", "sendall": true, "outputs": 0, "ok": "OK", "cancel": "", "raw": true, "topic": "", "name": "", "x": 990, "y": 240, "wires": []}, {"id": "ff3433d9.223", "type": "ui_template", "z": "fc31302a.f6743", "group": "16108a45.798416", "name": "Gráfico", "order": 1, "width": 12, "height": 8, "format": "<script>/httpStatic: /home/pi/.node-red/static*/></script>\n<script src=/Highcharts-9.2.2/code/highcharts.js></script>\n<script src=/Highcharts-9.2.2/code/modules/exporting.js></script>\n<script src=/Highcharts-"}

```

```

9.2.2/code/modules/export-data.js"></script>\n<script src="/Highcharts-
9.2.2/code/themes/dark-unica.js"></script>\n<div id="container4" width: auto
margin:0></div>\n \n<script>\nvar width4 = $("#container4").width();\n
(function(scope){\n scope.$watch('msg', function(msg) {\n var series4 =
chart4.series;\n for(i=0; i<series4.length; i++) {\n
series4[i].checkbox.style.marginLeft = '-2.5%';\n });\n
chart4.series[0].setData(msg.media);\n });\n })(scope);\nvar chart4 = new
Highcharts.chart('container4', {\n chart: {\n animation: false,\n zoomType: 'x',\n
type: 'spline'\n },\n title: {\n text: 'Resfriamento'\n },\n subtitle: {\n text:
'Trocador de Placas - Saída Mosto'\n },\n credits: { enabled: false },\n xAxis: [{\n
type: 'datetime',\n dateTimeLabelFormats: { // don't display the dummy year\n
\tsecond: '%H:%M:%S'\n },\n showFirstLabel: false,\n title: {\n text:
'Tempo Decorrido'\n } }],\n yAxis: {\n title: {\n text: 'Temperatura
(°C)'\n },\n },\n tooltip: {\n headerFormat: '<b>{point.x:Tempo decorrido:
%H:%M:%S}</b><br>',\n pointFormat: '{series.name}:<b style="text-align:center;">{point.y} °C</b><br>',\n valueDecimals:3,\n shared: true,\n
useHTML: true\n },\n plotOptions: {\n spline: {\n marker: {\n enabled:
false\n } }},\n series: {\n turboThreshold: '0',\n showCheckbox:
true,\n events: {\n checkboxClick: function (event) {\n if
(event.checked) {\n this.show();\n } else {\n this.hide();\n
}\n },\n legendItemClick: function() {\n return false;\n
}\n },\n series: [{name:'Média', selected: true}]\n
});\n</script>","storeOutMessages":true,"fwdInMessages":true,"resendOnRefresh":false,"t
emplateScope":"local","x":960,"y":60,"wires":[]],{"id":"313ac5ec.55af1a","type":"ui_t
ext","z":"fc31302a.f6743","group":"16108a45.798416","order":6,"width":8,"height":1,"nam
e":"","label":"Controle de Vazão","format":"{ msg.payload }","layout":"row-
spread","x":950,"y":500,"wires":[]],{"id":"4da98108.0076f","type":"ui_switch","z":"29e0
e22c.ba14ae","name":"","label":"Utilidade Fria Trocador","tooltip":"","group":"953a3e33.6f214","order":9,"width":6,"height":1,"passthru
":true,"decouple":"false","topic":"Utilisade Fria do Trocador","style":"","onvalue":"0","onvalueType":"num","onicon":"","oncolor":"","offval
ue":"1","offvalueType":"num","officon":"","offcolor":"","x":200,"y":400,"wires":["38761
508.c5d6ba"]],{"id":"407c3fc9.18db7","type":"ui_switch","z":"29e0e22c.ba14ae","name"
:"","label":"Fervura ->

```

Trocador", "tooltip": "", "group": "953a3e33.6f214", "order": 7, "width": 6, "height": 1, "passthru": true, "decouple": "false", "topic": "Fervura" ->
Trocador", "style": "", "onvalue": "0", "onvalueType": "num", "onicon": "", "oncolor": "", "offvalue": "1", "offvalueType": "num", "officon": "", "offcolor": "", "x": 200, "y": 340, "wires": [{"id": "9d79bec8.09817"}], {"id": "36862c58.a96a44", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "22", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 40, "wires": []}, {"id": "c0c12183.280fc", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "29", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 100, "wires": []}, {"id": "b50ec7db.443448", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "31", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 160, "wires": []}, {"id": "81a4639a.e03d6", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "13", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 220, "wires": []}, {"id": "70d0a1c9.9709", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "16", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 280, "wires": []}, {"id": "9d79bec8.09817", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "15", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 340, "wires": []}, {"id": "38761508.c5d6ba", "type": "rpi-gpio
out", "z": "29e0e22c.ba14ae", "name": "", "pin": "18", "set": true, "level": "1", "freq": "", "out": "out", "x": 440, "y": 400, "wires": []}, {"id": "9da8b3c.c403a5", "type": "ds18b20", "z": "7999b7cf.b7c508", "name": "topo", "sensorid": "28-031590210dff", "timer": "0.09", "x": 190, "y": 60, "wires": [{"id": "1798ea1e.175b26"}], {"id": "a7a34ec3.f6659", "type": "ds18b20", "z": "7999b7cf.b7c508", "name": "funдо", "sensorid": "28-011564ce4eff", "timer": "0.09", "x": 190, "y": 120, "wires": [{"id": "ac099f53.99753"}], {"id": "28fb050a.976c4a", "type": "ds18b20", "z": "d880af12.1b24f", "name": "Média", "sensorid": "not found.", "timer": "0.09", "x": 190, "y": 60, "wires": [{"id": "6daa07c95efe1442"}], {"id": "c9f2faeb.ef8dc8", "type": "ds18b20", "z": "31c8b922.7219c6", "name": "Média", "sensorid": "28-011564ce4eff", "timer": "0.09", "x": 190, "y": 60, "wires": [{"id": "89202d673752bf29"}], {"id": "8500bf07.271de", "type": "ds18b20", "z": "fc31302a.f6743", "name": "Média", "sensorid": "28-031590210dff", "timer": "0.09", "x": 190, "y": 60, "wires": [{"id": "453dfb30b4d1ac47"}], {"id": "f0efe180.8218c", "type": "ds18b20", "z": "c5dc1810.861398", "name": "Média", "sensorid": "not found.", "timer": "1", "x": 190, "y": 60, "wires": [{"id": "8829c7b9f2ede258"}], {"id": "1798ea1e.175b26", "type": "change", "z": "7999b7cf.b7c508", "name": "", "rules": [{"t": "set", "p": "topic", "pt": "msg", "to": "topo", "tot": "str"}], "action": "", "property": "", "from": "", "to": "", "reg": false, "x": 451.9881286621094, "y": 58.9959774017334, "wires": [{"id": "8aacd062.f144f"}], {"id": "ac099f53

```

.99753", "type": "change", "z": "7999b7cf.b7c508", "name": "", "rules": [{"t": "set", "p": "topic", "pt": "msg", "to": "fundo", "tot": "str"}], "action": "", "property": "", "from": "", "to": "", "reg": false, "x": 451.9881286621094, "y": 118.9959774017334, "wires": [{"8aacd062.f144f"}], {"id": "2fe8b8453c1acf2d", "type": "rpi-gpio
out", "z": "c5dc1810.861398", "name": "", "pin": "11", "set": true, "level": "1", "freq": "", "out": "out", "x": 960, "y": 500, "wires": []}, {"id": "36486e36c566bfc8", "type": "function", "z": "c5dc1810.861398", "name": "Deleta Dados", "func": "const fs = context.global.fs\npath = '/home/pi/Desktop/Temperatura_Fermentador.csv';\nif (fs.existsSync(path)) {\nfs.unlinkSync(path)\n}\n", "outputs": 0, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 500, "y": 180, "wires": []}, {"id": "8985e279906e575d", "type": "function", "z": "fc31302a.f6743", "name": "Reiniciar", "func": "flow.set('media', null);\nflow.set('old', null);\n\nmsg = {payload: ", media: []};\nmsg1 = {topic: 'reset', payload: null, topo: \"No Value\", media: \"No Value\", fundo: \"No Value\"};\nmsg2 = {payload: false};\n\nreturn [msg, msg1, msg2];", "outputs": 3, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 260, "y": 240, "wires": [{"83a4d83a91d623a4", "ff3433d9.223"}, {"47a818af.f72688", "513133ec.ab1adc", "286968a0.af8828"}, {"4cc829fa.956408"}], {"id": "83a4d83a91d623a4", "type": "function", "z": "fc31302a.f6743", "name": "Deleta Dados", "func": "const fs = context.global.fs\npath = '/home/pi/Desktop/Temperatura_Resfriamento.csv';\nif (fs.existsSync(path)) {\nfs.unlinkSync(path)\n}\n", "outputs": 0, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 500, "y": 180, "wires": []}, {"id": "cf8dece7970850aa", "type": "function", "z": "31c8b922.7219c6", "name": "Reiniciar", "func": "flow.set('media', null);\nflow.set('old', null);\n\nmsg = {payload: ", media: []};\nmsg1 = {topic: 'reset', payload: null, topo: \"No Value\", media: \"No Value\", fundo: \"No Value\"};\nmsg2 = {payload: false};\n\nreturn [msg, msg1, msg2];", "outputs": 3, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 260, "y": 240, "wires": [{"b11f3b51da83ebc6", "e7c51b68.6a4688"}, {"63e0f0dd.9b48e", "d87546c6.8df2a8", "a108bdfb.d9a8f"}, {"151a6dc8.c323b2"}], {"id": "b11f3b51da83ebc6", "type": "function", "z": "31c8b922.7219c6", "name": "Deleta Dados", "func": "const fs = context.global.fs\npath = '/home/pi/Desktop/Temperatura_Fervura.csv';\nif (fs.existsSync(path)) {\nfs.unlinkSync(path)\n}\n", "outputs": 0, "noerr": 0, "initialize": "", "finalize": "", "libs": [], "x": 500, "y": 180, "wires": []}, {"id": "6daa07c95efe1442", "type": "function", "z": "d880af12.1b24f", "name": "F1", "func": "media = msg.payload\n//Tratando a segunda saída\nmsg1 = {media: media.toFixed(1) + '

```



```

topo = \"No Value\";\n  \n} else if (msg.payload.fundo === undefined & msg.payload.topo
!== undefined){\n  topo = msg.payload.topo;\n  media = topo;\n  fundo = \"No Value\";\n
\n} else {\n  topo = msg.payload.topo;\n  fundo = msg.payload.fundo;\n  media =
(msg.payload.fundo+msg.payload.topo)/2;\n}\nmsg.payload['media']
=
media\n\n//Tratando a segunda saída\nmsg1 = {media: media.toFixed(1) + ' °C',\nfundo:
msg.payload['fundo'].toFixed(1) + ' °C',\n  topo: msg.payload['topo'].toFixed(1) + '
°C'};\n\n//Tratando a terceira saída\nmsg2 = {payload:
Math.round(media*10)/10};\n\n//Persistência\nconst fs =
context.global.fs\n\nObject.keys(msg.payload).map(function(key){\n  \n  if
(msg.payload[key] !== undefined){\n    const fs = context.global.fs;\n    \n    //Local
do arquivo\n    var path = '/home/pi/Desktop/Temperatura_Mostura_'+key+'.csv';\n    \n
//Cria um arquivo em branco caso não exista\n    if (!fs.existsSync(path)) {\n
fs.appendFileSync(path, Date.now()+\"\\n\\n\"+msg.payload[key])\n    }\n    else{\n
//Insere dados se o arquivo existe\n
fs.appendFileSync(path, \"\\n\\n\"+Date.now()+\"\\n\\n\"+msg.payload[key]);\n    //Lê o
arquivo\n    data = fs.readFileSync(path,'utf-8');\n    //Convert CSV em Array de
Números\n    rows = data.split(\"\\n\\n\");\n    \n    msg[key] = rows.map(function
(row, index) \n    {\n    //Se é a primeira medida\n    //guarda o primeiro
valor\n    if (index == 0)\n    {\n    return
row.split(\"\\n\").map(function(item, index, arr)\n    {\n    if (index ===
0){\n    first = Number(item);\n    item = 0;\n    }\n
else{\n    item = Number(item);\n    \n    }\n
return item\n    }\n    }\n    }\n    //não é a primeira medida
subtrai o primeiro valor\n    //para obter a duração\n    else{\n    return
row.split(\"\\n\").map(function(item, index, arr){\n    if (index === 0){\n
item = Number(item) - first;\n    }\n    else{item = Number(item)}\n
return item;\n    }\n    }\n    }\n    \n    }\n    }\n    }\n
\n    }\n    }\n    }\n\nreturn [msg, msg1,
msg2];\n\n,\"outputs\":3,\"noerr\":0,\"initialize\":\"\",\"finalize\":\"\",\"libs\":[],\"x\":750,\"y\":60,\"wires\"
:[[\"6cff43b9.93bd1c\"],[\"83234576.3914d8\",\"57590844.2aa108\",\"c5df4c38.b4c3a\"],[\"9633
8031.46704\"]]}, {\"id\":\"1ce7b9f9d7c17bbd\",\"type\":\"function\",\"z\":\"7999b7cf.b7c508\",\"nam
e\":\"Reiniciar\",\"func\":\"msg = {payload: '', media: [], topo: [], fundo: []};\nmsg1 = {topic:
'reset', payload: null, \n  topo: \"No Value\", media: \"No Value\", \n  fundo: \"No
Value\"};\nmsg2 = {payload: false};\n\nreturn [msg, msg1,

```


8.2 LICENÇA HIGHCHARTS



Highsoft Non-Commercial License Statement

This email constitutes as your license statement.

Date of issue:

License holder:

Bryan Nascimento
Universidade Federal de São Carlos

This license is valid for:

Personal / Student Use usage of the following product(s):
Highcharts.

This software is released under [Creative Commons Attribution-NonCommercial 3.0](#)

Download

Download the library from highcharts.com/download. No further activation or license key is required.

fyi:

[Click here for Newsletter](#)

We highly recommend that you subscribe to our newsletter. You will receive approximately one email a month with tips and tricks on how to get the most out of your charting products.