



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
DEPARTAMENTO DE AUTOMAÇÃO E SISTEMAS  
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Mathias Dutra Nichele

**myTapp Fast - Auto serviço de chope automatizado**

Florianópolis  
2023

Mathias Dutra Nichele

**myTapp Fast - Auto serviço de chope automatizado**

Relatório final da disciplina DAS5511 (Projeto de Fim de Curso) como Trabalho de Conclusão do Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Catarina em Florianópolis.

Orientador: Prof. Carlos Barros Montez

Supervisor: João Paulo Bodanese

Florianópolis

2023

Mathias Dutra Nichele

**myTapp Fast - Auto serviço de chope automatizado**

Esta monografia foi julgada no contexto da disciplina DAS5511 (Projeto de Fim de Curso) e aprovada em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação

Florianópolis, 10 de Julho de 2023.

---

Prof. Hector Bessa Silveira, Dr.  
Coordenador do Curso

**Banca Examinadora:**

---

Prof. Carlos Barros Montez, Dr.  
Orientador  
UFSC/CTC/DAS

---

João Paulo Bodanese  
Supervisor  
myTapp

---

Publio Macedo  
Avaliador  
UFSC/CTC/DAS

---

Prof. Eduardo Camponogara, Dr.  
Presidente da Banca  
UFSC/CTC/DAS

## **AGRADECIMENTOS**

Aos meus familiares e amigos, gostaria de expressar meu sincero agradecimento por estarem sempre presentes em minha vida, me apoiando e me encorajando a nunca desistir. Seu suporte foi fundamental para o meu sucesso e sou imensamente grato por isso.

A todos que fizeram parte dessa jornada, quero agradecer de coração por terem contribuído de alguma forma em minha vida. Cada um de vocês deixou sua marca e sou grato por ter tido a oportunidade de aprender e crescer ao lado de pessoas tão especiais.

Mais uma vez, obrigado a todos vocês por me ajudarem a alcançar meus objetivos. Estou verdadeiramente feliz em compartilhar essa conquista com cada um de vocês. Sua presença e apoio foram inestimáveis e sou grato por tê-los ao meu lado nessa jornada.

*"You're either the one that creates the automation or you're getting automated."  
(Tom Preston-Werner, 2014)*

## DECLARAÇÃO DE PUBLICIDADE

Florianópolis, 10 de Julho de 2023.

Na condição de representante da myTapp na qual o presente trabalho foi realizado, declaro não haver ressalvas quanto ao aspecto de sigilo ou propriedade intelectual sobre as informações contidas neste documento, que impeçam a sua publicação por parte da Universidade Federal de Santa Catarina (UFSC) para acesso pelo público em geral, incluindo a sua disponibilização *online* no Repositório Institucional da Biblioteca Universitária da UFSC. Além disso, declaro ciência de que o autor, na condição de estudante da UFSC, é obrigado a depositar este documento, por se tratar de um Trabalho de Conclusão de Curso, no referido Repositório Institucional, em atendimento à Resolução Normativa n° 126/2019/CUn.

Por estar de acordo com esses termos, subscrevo-me abaixo.



Documento assinado digitalmente

JOAO PAULO BODANESE

Data: 03/07/2023 16:00:40-0300

Verifique em <https://validar.itl.gov.br>

---

João Paulo Bodanese

myTapp

## RESUMO

Este trabalho apresenta um sistema automatizado de serviço de chope para eventos, visando melhorar a eficiência no atendimento, reduzir custos operacionais e garantir a segurança do serviço. A metodologia adotada envolveu o projeto mecânico e eletrônico, seleção de componentes adequados como motores de passo e sensores, programação do microcontrolador e realização de testes para validar o funcionamento do sistema. Espera-se obter um sistema automático de auto serviço de cerveja para eventos, oferecendo uma experiência diferenciada aos clientes. O trabalho está estruturado em torno da problemática do atendimento eficiente, justificando a necessidade desse sistema. O aluno desempenhou um papel ativo na execução das etapas e contribuiu individualmente para o desenvolvimento do projeto.

**Palavras-chave:** Automação de chope. Automação para eventos. Eficiência no atendimento.

## **ABSTRACT**

This work presents an automated beer service system for events, aiming to improve efficiency in service, reduce operational costs, and ensure service safety. The methodology adopted involved mechanical and electronic design, selection of suitable components such as stepper motors and sensors, microcontroller programming, and conducting tests to validate the system's operation. The expected outcome is an automatic self-service beer system for events, offering a differentiated experience to customers. The work is structured around the problem of efficient service, justifying the need for this system. The student played an active role in carrying out the stages and made individual contributions to the project development.

**Keywords:** Beer automation. Automation for events. Efficiency in service.



## LISTA DE FIGURAS

Figura 1 – Torneira italiana . . . . .	21
Figura 2 – Torneira belga . . . . .	21
Figura 3 – Ativando PWM via DTS. . . . .	32
Figura 4 – Código para configuração do PWM. . . . .	32
Figura 5 – Código para configuração do ADC. . . . .	32
Figura 6 – Código para configuração de saída do GPIO. . . . .	33
Figura 7 – Código para configuração de entrada do GPIO. . . . .	33
Figura 8 – Montagem de protótipo em chopeira. . . . .	35
Figura 9 – Montagem de protótipo em um análogo a câmara fria. . . . .	36
Figura 10 – Montagem do produto na Oktoberfest. . . . .	37
Figura 11 – Representação da velocidade por distancia. . . . .	38
Figura 12 – Tela de repouso. . . . .	39
Figura 13 – Tela servindo. . . . .	39
Figura 14 – Tela de aviso: "Acabou o barril". . . . .	40
Figura 15 – Fluxo para gerar consumo em cartão netPDV. . . . .	41
Figura 16 – Primeira versão da estrutura mecânica para acoplamento. . . . .	43
Figura 17 – Versão revisada da estrutura mecânica para acoplamento. . . . .	44
Figura 18 – Primeiro protótipo funcional. . . . .	45
Figura 19 – Estrutura com chave de FDC e apoio para os sensores. . . . .	47
Figura 20 – Estrutura final com case. . . . .	48
Figura 21 – Modelo de estrutura para melhorar funcionamento do atuador linear. . . . .	52
Figura 22 – Protótipo da estrutura para o atuador linear. . . . .	53
Figura 23 – Datasheet do motor linear. . . . .	66

## LISTA DE ABREVIATURAS E SIGLAS

PFC	<i>Projeto de Fim de Curso</i>
TAP	<i>Torneira com o auto serviço</i>
CMS	<i>Content Management System</i>
PDV	<i>Pontos de Venda</i>
PWM	<i>Pulse Width Modulation</i>
ADC	<i>Analog-to-digital converter</i>
DTS	<i>Device Tree Source</i>
DTB	<i>Device Tree Binary</i>
FDC	<i>Fim de Curso</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
<b>2</b>	<b>A EMPRESA</b>	<b>14</b>
<b>3</b>	<b>ASPECTOS CONCEITUAIS</b>	<b>16</b>
3.1	DIFERENÇA PARA O AUTO SERVIÇO	16
3.2	ATUADOR LINEAR	16
3.3	MOTOR DE PASSO	17
3.4	ACOPLAMENTO MECÂNICO	18
3.5	CONTROLE DO MOTOR	19
3.6	SERVO MOTOR	20
3.7	TORNEIRAS	21
3.8	SERVIÇO DE CERVEJA	22
3.9	POUPA CHOPE	23
3.10	APLICAÇÃO	24
3.11	DEVICE TREE SOURCE (DTS) E DEVICE TREE BINARY (DTB)	24
<b>4</b>	<b>REQUISITOS E DESCRIÇÃO CONCEITUAL DA SOLUÇÃO PROPOSTA</b>	<b>26</b>
4.1	REQUISITOS GERAIS, FUNCIONAIS E NÃO FUNCIONAIS	26
4.2	DESCRIÇÃO CONCEITUAL DO FUNCIONAMENTO	26
4.3	ARQUITETURA E COMPONENTES ENVOLVIDOS	27
<b>5</b>	<b>DESENVOLVIMENTO</b>	<b>29</b>
5.1	VERSÃO 1.0	29
<b>5.1.1</b>	<b>Validações com Arduino</b>	<b>29</b>
<b>5.1.2</b>	<b>Validações com Toradex</b>	<b>31</b>
<b>5.1.3</b>	<b>Programa e lógica de ativação</b>	<b>35</b>
<b>5.1.4</b>	<b>Interface gráfica</b>	<b>38</b>
<b>5.1.5</b>	<b>Integração com o cashless</b>	<b>39</b>
5.2	VERSÃO 2.0	40
<b>5.2.1</b>	<b>Seleção do motor de passo e projeto estrutura mecânica</b>	<b>42</b>
<b>5.2.2</b>	<b>Programa e lógica de ativação</b>	<b>46</b>
<b>5.2.3</b>	<b>Estrutura mecânica para atuador linear</b>	<b>51</b>
<b>5.2.4</b>	<b>Problema da espuma</b>	<b>51</b>
<b>6</b>	<b>IMPLEMENTAÇÃO</b>	<b>54</b>
6.1	VERSÕES	54
6.2	HARDWARE	55
6.3	SOFTWARE	55
6.4	INTEGRAÇÕES E COMUNICAÇÃO	55
<b>7</b>	<b>ANÁLISE DOS RESULTADOS</b>	<b>57</b>

<b>8</b>	<b>CONCLUSÃO . . . . .</b>	<b>61</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>62</b>
	<b>APÊNDICE A – DEFINIÇÃO DOS STRUCTS . . . . .</b>	<b>64</b>
	<b>APÊNDICE B – FUNÇÃO PARA RELAÇÃO DA ESPUMA . . . . .</b>	<b>65</b>
	<b>ANEXO A – DATASHEET DO ATUADOR LINEAR . . . . .</b>	<b>66</b>

## 1 INTRODUÇÃO

Este trabalho apresenta o desenvolvimento de um sistema automatizado de serviço de chope para eventos, com o objetivo de melhorar a eficiência no atendimento, reduzir custos operacionais e garantir a segurança do serviço. A empresa myTapp, pioneira em auto serviço no Brasil, tem buscado novas oportunidades de expansão diante da crescente competição no mercado. Nesse contexto, identificou-se um grande espaço para a implementação de um sistema automatizado de serviço de chope específico para eventos.

A adoção de um sistema automatizado traz diversas vantagens, incluindo a capacidade de atender um maior número de clientes em menos tempo, proporcionando uma experiência mais satisfatória para o público. Além disso, a automatização permite a redução de custos operacionais ao dispensar a necessidade de funcionários para servir cerveja, representando uma economia significativa para as empresas organizadoras de eventos. A segurança também é beneficiada, pois é possível controlar com precisão a quantidade de cerveja servida em cada copo, garantindo que os clientes sejam atendidos de forma adequada.

O propósito deste Projeto de Fim de Curso (PFC) consiste em desenvolver o produto final. Para alcançar esse objetivo, será criado um modelo inicial simplificado, com o intuito de validar o modelo de negócios. Após a validação, será concebida uma versão aprimorada e altamente escalável do sistema.

No primeiro modelo, foi utilizado um atuador linear elétrico para controlar o movimento do mecanismo responsável pelo serviço de cerveja. Esse atuador linear proporciona um movimento linear preciso e controlado.

Na versão mais avançada do sistema, será utilizado um motor de passo em conjunto com uma estrutura mecânica adequada. O motor de passo permite um controle mais refinado do movimento, possibilitando um serviço de chope ainda mais preciso e eficiente.

Além disso, sensores serão integrados ao sistema para detectar a quantidade de cerveja servida em cada copo. Esses sensores garantirão que a quantidade correta seja despejada, proporcionando uma experiência consistente aos usuários.

Para alcançar esse objetivo geral, foram estabelecidos objetivos específicos, tais como projetar o sistema mecânico, incluindo o mecanismo de servir cerveja; desenvolver a parte eletrônica do sistema, abrangendo o controle do motor e leitura dos sensores; programar o microcontrolador responsável pelo controle do sistema e assegurar seu correto funcionamento; realizar testes para validar o desempenho do sistema, considerando diferentes tipos de cerveja e condições de operação.

Ao final deste projeto, espera-se obter um sistema automático de auto serviço de cerveja que possa ser amplamente utilizado em eventos, integrando-se aos sistemas

de ponto de venda (PDV) existentes e oferecendo uma experiência diferenciada aos clientes.

A estrutura deste documento está organizada da seguinte forma:

- **Capítulo 2:** será apresentada uma descrição da empresa, seus processos, layouts, problemas identificados e indicadores relevantes.
- **Capítulo 3:** abordará os aspectos conceituais, incluindo teorias, modelos e técnicas utilizados no desenvolvimento da solução proposta.
- **Capítulo 4:** serão estabelecidos os requisitos gerais e funcionais do sistema, juntamente com uma descrição conceitual do projeto, abrangendo fluxos de informação e materiais e estratégias de integração de sistemas.
- **Capítulo 5:** irá tratar sobre o processo de desenvolvimento em si, assim como as soluções alcançadas.
- **Capítulo 6:** fornecerá uma descrição detalhada do projeto, incluindo hardware, software e implementação.
- **Capítulo 7:** os resultados serão analisados criticamente, considerando a efetividade do sistema em relação aos objetivos estabelecidos.
- **Capítulo 8:** apresentará as conclusões gerais do trabalho, resumindo o que foi realizado, as limitações encontradas e as sugestões para trabalhos futuros.

A metodologia adotada neste projeto será discutida ao longo deste documento, detalhando as técnicas e procedimentos utilizados em cada fase do desenvolvimento.

Espera-se que este trabalho proporcione uma solução inovadora e eficiente para o serviço de chope em eventos, atendendo às necessidades do mercado e agregando valor à empresa myTapp.

## 2 A EMPRESA

A myTapp é a primeira empresa de auto serviço de chope do Brasil, fundada em 2015 na cidade de Florianópolis. A empresa opera no modelo de comodato, no qual os equipamentos são de propriedade da empresa, mas são locados para os contratantes.

O sistema myTapp é composto por duas partes principais: a TAP e o Content Management System (CMS). A TAP é um dispositivo embarcado que é instalado nas torneiras. Essa tecnologia é responsável pela leitura dos sensores, medindo a quantidade de chope servido, realizando a leitura do cartão do cliente e permitindo a comunicação com o CMS. Por sua vez, o CMS é um servidor que armazena todas as informações relevantes do contratante, como cadastro de clientes, histórico de consumo, informações sobre as cervejas disponíveis e registros das TAPS instaladas. A TAP é composta por uma placa de circuitos conectada a um dispositivo embarcado chamado Toradex, produzido por uma empresa suíça.

A myTapp, além de alugar os equipamentos, é responsável pelos serviços de contratação de técnicos para a montagem dos equipamentos, acompanhamento da implantação inicial e suporte contínuo após a instalação. Essa abordagem garante que os contratantes tenham uma experiência tranquila e segura ao utilizar o sistema myTapp.

Atualmente, a myTapp atende a uma base de +360 clientes em 6 países, o que demonstra a expansão e a aceitação do sistema em diferentes mercados.

No mercado altamente competitivo de auto serviço de chope, a myTapp destaca-se como uma empresa líder, enfrentando uma concorrência significativa de outros provedores. O diferencial da myTapp reside na qualidade superior do seu produto, impulsionada pelo uso de uma placa própria com um microcontrolador Toradex, que garante um desempenho excepcional e confiável. Com base nessa reputação estabelecida, a myTapp reconheceu o potencial do ramo de eventos, um mercado considerado novo e inexplorado. Essa expansão estratégica para o setor de eventos oferece oportunidades significativas de crescimento e expansão para a empresa, permitindo que ela atinja novos públicos e aumente sua presença no mercado.

A myTapp, reconhecendo a problemática enfrentada na implementação do auto serviço tradicional em eventos, buscou desenvolver uma solução inovadora para lidar com os desafios específicos desse contexto. Ao permitir o acesso direto da torneira aos consumidores, o produto tradicional da myTapp enfrentava dificuldades relacionadas à falta de fiscalização adequada e à alta quantidade de pessoas presentes nos eventos.

Uma das principais questões observadas era o fato de que as pessoas, muitas vezes, tentavam servir chope com a solenoide da torneira fechada, o que resultava na entrada de ar na linha de chope e na formação excessiva de espuma nas próximas servidas. Além disso, a falta de conhecimento sobre o manuseio correto de uma torneira

de chope por parte dos clientes também contribuía para problemas durante o serviço.

Consciente desses desafios, a myTapp buscou desenvolver uma solução que oferecesse um serviço mais seguro, eficiente e de melhor qualidade para eventos. A proposta foi criar um sistema fechado e controlado por motor, reduzindo a interação direta dos clientes com a torneira. Com esse sistema automatizado, a myTapp garante um serviço mais confiável, evitando situações indesejadas causadas por erros de manuseio ou falta de conhecimento dos clientes.

Dessa forma, a myTapp conseguiu superar as limitações e oferecer uma solução inovadora para o segmento de eventos, proporcionando uma experiência de auto serviço de chope mais segura, eficiente e livre de problemas. A abordagem diferenciada da myTapp permitiu que ela se destacasse no mercado, conquistando a confiança dos organizadores de eventos e abrindo portas para novas oportunidades de negócios.

Diante desses desafios, a myTapp propõe uma solução inovadora ao criar um sistema fechado e controlado por motor. Esse sistema automatizado reduz a interação direta do cliente com a torneira, oferecendo um serviço mais seguro e evitando situações indesejadas causadas pela falta de conhecimento ou pela manipulação incorreta da torneira.

Ao abordar essa problemática, o projeto de desenvolvimento do sistema busca garantir um serviço mais eficiente, seguro e com melhor qualidade, superando os desafios enfrentados na implementação do auto serviço tradicional em eventos.



### 3 ASPECTOS CONCEITUAIS

No contexto do desenvolvimento do sistema, é fundamental compreender os aspectos conceituais relacionados ao funcionamento dos elementos envolvidos. Neste capítulo, discutiremos as teorias, modelos e técnicas utilizados na concepção da solução proposta.

#### 3.1 DIFERENÇA PARA O AUTO SERVIÇO

O sistema de auto serviço de chope convencional tem como objetivo proporcionar maior liberdade ao cliente, permitindo que ele experimente pequenas quantidades de cerveja enquanto paga apenas pelo que consumiu. Essa abordagem reduz as perdas para o estabelecimento e diminui a necessidade de funcionários para atender os clientes. No auto serviço convencional, os créditos do usuário são descontados posteriormente, de acordo com o que foi servido.

Por outro lado, o sistema de auto serviço de chope automatizado busca oferecer uma experiência ágil, em que uma quantidade fixa de chope é servida de forma rápida e eficiente. Nesse caso, os créditos podem ser descontados antes do serviço, o que abre possibilidades adicionais de integração com sistemas de pagamento. Além disso, no auto serviço automatizado, o cliente não precisa ter conhecimentos sobre como operar uma torneira de chope, pois o sistema cuida de todo o processo.

Essa diferença fundamental entre os dois sistemas reflete a abordagem de cada um em relação à experiência do cliente. Enquanto o auto serviço convencional oferece uma maior liberdade de experimentação e pagamento proporcional, o auto serviço automatizado prioriza a agilidade e a facilidade de uso, proporcionando uma experiência mais fluida e rápida para o cliente.

#### 3.2 ATUADOR LINEAR

Os atuadores lineares são dispositivos mecânicos que transformam o movimento rotacional em um movimento linear. No sistema foram utilizados atuadores lineares elétricos. Esses atuadores são compostos por um motor elétrico acoplado a um sistema de transmissão que converte o movimento rotacional em um movimento linear.

Os atuadores lineares apresentam várias vantagens, incluindo controle preciso do deslocamento linear, alta repetibilidade, capacidade de operar em velocidades variáveis e facilidade de integração com sistemas de controle eletrônico (TURNER *et al.*, 2006). Além disso, o atuador linear utilizado no sistema possui um feedback de posição por meio de um potenciômetro, o que garante uma operação mais segura.

Essas características tornam os atuadores lineares adequados para aplicações

que exigem posicionamento preciso e controle de movimento linear. No caso do sistema, os atuadores lineares desempenham um papel importante no controle do movimento dos componentes, contribuindo para a dosagem precisa da cerveja e da espuma durante o serviço.

### 3.3 MOTOR DE PASSO

Os motores de passo são dispositivos amplamente utilizados em aplicações que exigem controle preciso de posicionamento e movimento (TEJWANI; SOLANKI, 2010). Eles são compostos por bobinas e um conjunto de ímãs permanentes, que interagem para gerar o movimento. O princípio de funcionamento baseia-se na aplicação de correntes elétricas controladas nas bobinas do motor.

Quando uma corrente elétrica passa por uma bobina, um campo magnético é criado, interagindo com o campo magnético do conjunto de ímãs. Essa interação faz com que o rotor do motor se alinhe com a posição magnética mais forte, gerando um passo de movimento. Ao enviar pulsos elétricos controlados para as bobinas do motor, é possível realizar o movimento de passo em incrementos fixos.

No entanto, o movimento em passos completos pode resultar em menor precisão e suavidade. Assim pode ser implementada a técnica de microstepping. O microstepping permite dividir cada passo em subdivisões menores, proporcionando um controle mais preciso do motor de passo. Isso é alcançado por meio do ajuste das correntes elétricas nas bobinas em níveis intermediários entre os níveis máximos e mínimos. Dessa forma, o motor é capaz de executar movimentos suaves, reduzindo vibrações e melhorando a precisão do posicionamento.

No microstepping, a corrente nas bobinas é controlada de forma senoidal ou em formas de onda específicas para obter o movimento suave. Essa técnica permite que o motor alcance posições intermediárias entre os passos completos, proporcionando uma resolução mais alta e uma melhor resposta dinâmica.

No contexto do sistema, o microstepping é utilizado para aumentar a precisão do posicionamento do motor de passo que controla a torneira. Isso é crucial para garantir a dosagem precisa da cerveja, evitando desperdícios.

Quando o motor de passo recebe um comando para se mover, ele tenta realizar o movimento correspondente ao número de passos determinado. No entanto, em algumas situações, pode ocorrer o chamado "pulo de passo" (step skipping), que é quando o motor não consegue exercer força suficiente para completar o movimento desejado.

Esse problema pode ocorrer devido a diversos fatores, como carga excessiva, torque insuficiente ou frequência de pulsos muito alta. Quando o motor encontra dificuldades em superar essas situações, ocorre o pulo de passo, que causa uma perda de referência entre a posição real do motor e a posição que o programa acredita que

o motor está. Isso ocorre porque o motor é controlado em malha aberta, ou seja, o programa não possui um feedback direto sobre o posicionamento real do motor.

O resultado dessa perda de referência é uma falta de sincronia entre o sistema de controle e o motor de passo. O programa continua enviando comandos para o motor se mover uma determinada quantidade de passos, mas o motor não consegue executar corretamente esse movimento. Isso pode levar a erros de posicionamento e comprometer a precisão do sistema de serviço de chope automatizado.

Portanto, é fundamental compreender e enfrentar os desafios associados ao pulo de passo, implementando estratégias adequadas de controle e compensação para garantir a operação precisa e confiável do sistema.

### 3.4 ACOPLAMENTO MECÂNICO

Para transformar o movimento de rotação do motor de passo em uma translação angular capaz de controlar a torneira, é utilizado um acoplamento de engrenagens. Esse acoplamento consiste em duas engrenagens: uma fixada no eixo do motor de passo e outra presa à torneira. Por meio de uma estrutura mecânica adequada, as engrenagens garantem o contato e o curso necessário para a transmissão do movimento.

No entanto, nos atuadores lineares utilizados no sistema, não é necessário um acoplamento via engrenagens. O próprio atuador linear é projetado de forma que o movimento de rotação do motor seja convertido em movimento linear sem a necessidade de engrenagens externas. Essa integração interna permite uma conexão direta do atuador linear com a torneira, simplificando o sistema mecânico e reduzindo a possibilidade de desgaste e perdas de energia associadas ao uso de engrenagens.

A escolha da relação de engrenagens para o motor de passo é um aspecto crucial no projeto do sistema, pois diferentes relações oferecem vantagens e desvantagens específicas. Por exemplo, ao optar por uma relação de engrenagens com uma grande redução, é possível obter um maior torque de saída, o que é importante para superar possíveis resistências durante o movimento. No entanto, essa escolha pode resultar em uma diminuição da velocidade, afetando o tempo necessário para o serviço de chope. Por outro lado, ao utilizar uma relação de engrenagens com uma menor redução, é possível alcançar uma velocidade mais alta, mas com um torque reduzido. No caso desse sistema, como o curso é pequeno, foi optado por uma relação com grande redução.

Além da consideração da relação de engrenagens, é essencial avaliar o impacto do atrito e da eficiência na transmissão do movimento por meio das engrenagens. Durante o contato entre os dentes das engrenagens, ocorre o atrito, o qual pode resultar em perdas de energia, redução da eficiência geral do sistema, além de desgaste das peças. Portanto, é necessário utilizar materiais adequados para minimizar essas perdas.

Outro aspecto relevante é o controle de ruídos e vibrações gerados pela transmissão por engrenagens. Essas vibrações indesejadas podem impactar negativamente a experiência do usuário durante o serviço de chope, tornando-o desconfortável. Portanto, é necessário realizar análises e testes para identificar e minimizar esses ruídos e vibrações.

Ao projetar o acoplamento mecânico com engrenagens para o sistema, é fundamental considerar todas essas variáveis e realizar testes práticos para determinar a relação de engrenagens mais adequada. Esses testes podem envolver a avaliação do desempenho do sistema em diferentes cenários e condições de uso, bem como a análise de aspectos como precisão, eficiência energética, durabilidade e segurança.

### 3.5 CONTROLE DO MOTOR

No caso dos atuadores lineares utilizados no sistema, o controle da velocidade do motor é realizado diretamente por meio da técnica de Pulse Width Modulation (PWM), ou Modulação por Largura de Pulso. Nesse método, o Duty Cycle do sinal PWM é utilizado para definir a velocidade de atuação do motor.

Esses atuadores lineares elétricos do sistema são controlados por meio de uma ponte H. Uma ponte H é um circuito eletrônico utilizado para controlar a direção e a velocidade de um motor DC. Ela permite a inversão do sentido de rotação do motor, bem como a variação da velocidade de acordo com o sinal de controle aplicado.

A ponte H desempenha um papel crucial no controle do movimento do motor, permitindo a inversão da direção com base no estado do sinal de controle enviado pelo pino GPIO.

Quando o pino GPIO está desligado, o sinal PWM é aplicado a um dos lados da ponte H, polarizando a ponte H. Nessa configuração, o Duty Cycle do sinal PWM afeta a velocidade de forma proporcional, ou seja, quanto maior o Duty Cycle, maior a velocidade do motor.

Por outro lado, quando o pino GPIO está ligado, a ponte H é polarizada pelo outro lado. Nessa situação, o Duty Cycle do sinal PWM afeta a velocidade de forma inversamente proporcional, ou seja, quanto menor o Duty Cycle, maior a velocidade do motor.

Essa solução utilizando a ponte H em conjunto com o sinal PWM e o pino GPIO permite controlar tanto a velocidade quanto a direção do atuador linear de forma flexível e precisa, usando somente um dos pinos PWM disponíveis no Toradex. Com essa abordagem, é possível obter um controle adequado e confiável do motor, garantindo um funcionamento correto do sistema.

No entanto, ao utilizar um sistema operacional Linux para controlar o motor de passo, surgem desafios relacionados à precisão e constância do controle em altas frequências. O Linux, embora seja uma plataforma robusta, não é um sistema em

tempo real, o que resulta em um agendamento não determinístico de tarefas. Essa característica dificulta o controle do motor de passo com pulsos individuais em espaçamentos constantes.

A natureza não determinística do agendamento de tarefas no Linux impede a garantia de uma temporização precisa dos pulsos enviados ao motor de passo. Isso significa que não é possível controlar o motor com uma sequência de pulsos de espaçamento fixo, o que poderia comprometer a precisão e a consistência do movimento.

Diante dessa limitação, é adotada uma abordagem inteligente para superar o problema. A solução consiste em empregar o uso de um PWM para controlar o movimento do motor de passo. Nessa abordagem, o tempo em que o sinal PWM permanece ligado é calculado de forma aproximada para determinar o número de passos que o motor deve se mover. Essa determinação leva em consideração a frequência do sinal PWM, ou seja, a quantidade de pulsos ou degraus de subida que o sinal deve enviar em um determinado período de tempo.

Ao utilizar o PWM como método de controle do motor de passo, é possível alcançar uma aproximação satisfatória do movimento desejado. No entanto, vale ressaltar que essa abordagem é uma aproximação, uma vez que o sistema Linux não garante a precisão necessária para um controle rigoroso em tempo real. Portanto, embora o sistema seja capaz de executar o movimento do motor de passo de forma adequada para aplicação em questão, é importante estar ciente das limitações inerentes ao uso do sistema operacional Linux.

Essa abordagem adaptativa e inteligente permite contornar as limitações do sistema operacional e alcançar um controle satisfatório do motor de passo. Com isso, é possível assegurar o funcionamento adequado do sistema, garantindo posicionamento do motor para se servir o chope.

### 3.6 SERVO MOTOR

Um servo motor é um dispositivo amplamente utilizado na área de automação e controle de movimento. É um tipo especial de motor que combina um motor de corrente contínua com um sistema de feedback de posição, geralmente por meio de um potenciômetro, para controlar com precisão a posição angular do eixo de saída.

O servo motor é projetado para responder a sinais de controle, permitindo o posicionamento preciso em uma determinada posição desejada. Ele oferece um alto grau de precisão e controle, o que o torna ideal para uma variedade de aplicações que exigem movimento controlado e repetitivo, como robótica, automação industrial, sistemas de posicionamento, modelismo, entre outros.

A principal característica de um servo motor é sua capacidade de manter a posição desejada, mesmo diante de forças externas ou perturbações. Isso é possível

graças ao sistema de feedback incorporado, que compara a posição real do eixo com a posição de referência e ajusta o sinal de controle para corrigir qualquer desvio. Dessa forma, o servo motor pode compensar alterações no ambiente ou cargas aplicadas, garantindo uma operação estável e precisa.

Além disso, os servo motores geralmente possuem uma faixa de rotação limitada, que pode variar de 90 a 180 graus, dependendo do modelo. Eles também são capazes de operar em velocidades variáveis, permitindo um controle fino do movimento.

No contexto da eletrônica, o servo motor é frequentemente controlado por um sinal de PWM (Pulse Width Modulation), onde a largura do pulso determina a posição desejada. Com base nessa informação, o servo motor ajusta sua posição até alcançar o ângulo correto.

### 3.7 TORNEIRAS

No sistema é utilizado um tipo específico de torneira: a torneira italiana, como mostrado na figura 1. Essa torneira possui uma característica distintiva em relação ao fluxo de líquido que é servido. Quando acionada, a torneira italiana permite o fluxo de cerveja para frente e a formação de espuma para trás, proporcionando um serviço adequado e preciso.

O sistema myTapp foi projetado levando em consideração essa característica da torneira italiana, garantindo um controle preciso do fluxo de cerveja e espuma durante o processo de serviço. Essa escolha é fundamentada na compatibilidade entre a estrutura mecânica do sistema e o acoplamento de engrenagens utilizado.

É importante ressaltar que existem outros tipos de torneiras disponíveis no mercado, como a torneira belga, mostrada na figura 2. Essa torneira apresenta uma peculiaridade diferente em relação ao fluxo de líquido. Ao ser parcialmente aberta, a torneira belga permite o serviço de espuma, e ao ser completamente aberta, permite o serviço de cerveja. Essa variação no fluxo é um aspecto específico do design da torneira belga (XAVIER, 2022).



Figura 1 – Torneira italiana



Figura 2 – Torneira belga

Fonte: Arquivo myTapp (2022).

Além disso, é relevante mencionar que diferentes fabricantes de torneiras podem oferecer produtos com qualidades distintas. Algumas torneiras podem ser mais "pesadas" para servir, exigindo um esforço maior por parte do usuário. Essa variação na qualidade e no funcionamento das torneiras é importante considerar ao escolher os componentes do sistema, garantindo a melhor experiência para os usuários finais.

### 3.8 SERVIÇO DE CERVEJA

O processo de servir cerveja envolve alguns aspectos importantes que contribuem para a qualidade da bebida e a satisfação dos consumidores. Dentre esses aspectos, destacam-se a presença de gás na cerveja e o nível de espuma. Compreender esses elementos é fundamental para garantir uma experiência agradável aos apreciadores da bebida.

A presença de gás na cerveja, geralmente dióxido de carbono (CO<sub>2</sub>), é responsável por conferir características como efervescência e frescor à bebida. Esse gás é produzido durante o processo de fermentação, no qual o açúcar presente nos ingredientes da cerveja é convertido em álcool e CO<sub>2</sub>. A quantidade de gás presente na cerveja pode variar de acordo com o estilo e a preferência do fabricante, resultando em diferentes níveis de carbonatação. A carbonatação adequada contribui para uma sensação refrescante e uma experiência sensorial mais agradável ao paladar.

Além disso, a espuma é um elemento essencial no serviço de cerveja, pois desempenha papéis importantes tanto na estética quanto no sabor. A formação correta da espuma na cerveja proporciona uma apresentação visual atrativa, com a cor e a textura adequadas. Além disso, a espuma retém os aromas da cerveja, liberando-os gradualmente para intensificar a percepção dos sabores. A espuma também atua como uma barreira protetora, ajudando a preservar a carbonatação da cerveja e evitar a oxidação prematura. O nível de espuma ideal varia de acordo com o estilo da cerveja, mas geralmente é desejável uma espuma densa e persistente, que não se dissipe rapidamente.

Para garantir o serviço adequado da cerveja, é importante considerar a temperatura ideal de consumo, o tipo de copo utilizado e a técnica correta de servir. A temperatura influencia diretamente a percepção dos sabores e aromas da cerveja, sendo recomendado seguir as recomendações do fabricante ou as diretrizes do estilo em questão. O tipo de copo utilizado também pode influenciar na experiência, uma vez que diferentes formatos favorecem a liberação dos aromas e a formação da espuma. Já a técnica de servir, incluindo o ângulo de inclinação do copo e o controle do fluxo de cerveja, contribui para a formação adequada da espuma e evita o desperdício de líquido.

O ângulo do copo no momento do serviço de cerveja é um fator importante a ser considerado. Geralmente, recomenda-se inclinar o copo em um ângulo de aproxima-

damente 45 graus e direcionar o fluxo da cerveja para a parede do copo. Essa técnica auxilia na formação adequada de espuma, pois permite que a cerveja escorra suavemente pelas laterais do copo, facilitando a criação de uma espuma densa e uniforme. À medida que o copo vai sendo preenchido, é possível ir ajustando gradualmente o ângulo para uma posição mais vertical, evitando o derramamento e mantendo a espuma na proporção adequada.

Outro fator a ser considerado é o impacto do tempo de inatividade da torneira de chope. Quando uma torneira fica sem servir cerveja por um período prolongado, especialmente em dias quentes, a temperatura do sistema interno da torneira pode aumentar. Isso ocorre devido ao contato do líquido residual com partes metálicas da torneira, como o barril ou os componentes internos. Esse aumento de temperatura pode resultar em uma maior formação de espuma ao servir a primeira cerveja após o período de inatividade. Portanto, é recomendado realizar uma sangria inicial, servindo um pequeno volume de cerveja antes de servir aos clientes, a fim de eliminar o excesso de espuma e garantir que o serviço seja adequado.

Portanto, ao desenvolver um sistema automatizado de serviço de chope, é fundamental considerar esses aspectos para garantir a qualidade da bebida e proporcionar uma experiência satisfatória aos consumidores. O controle preciso do fluxo de cerveja e a correta dosagem do gás são elementos-chave para alcançar o equilíbrio desejado entre sabor, carbonatação e formação de espuma, resultando em uma cerveja de qualidade que atenda às expectativas dos apreciadores.

### 3.9 POUPA CHOPE

Um poupa chope é um dispositivo utilizado em sistemas de distribuição de chope para evitar a entrada de ar na linha de chope. Ele desempenha um papel crucial na preservação da qualidade e do sabor da bebida. O ar é considerado prejudicial para o chope, pois pode levar à oxidação do líquido, resultando em uma deterioração do sabor e na perda das características desejadas.

O funcionamento do poupa chope é bastante simples. Ele é instalado na linha de chope, entre o barril e a torneira, e possui uma câmara de ar que atua como um reservatório. Quando o chope é servido, o poupa chope é preenchido com líquido, criando uma barreira entre o chope e o ar externo. Dessa forma, o ar é impedido de entrar na linha de chope, evitando a oxidação e garantindo a qualidade do produto.

É fundamental evitar a entrada de ar na linha de chope, pois isso afeta diretamente a experiência do consumidor. O ar pode alterar o sabor e o aroma da bebida, resultando em um chope sem frescor e com características indesejadas. Além disso, a presença de ar na linha de chope pode causar problemas de carbonatação, resultando em uma bebida com excesso de espuma ou com baixa carbonatação.

Portanto, o uso de um poupa chope é essencial para manter a integridade do



chope, preservando seu sabor, aroma e textura. Ao evitar a entrada de ar na linha de chope, podemos oferecer aos consumidores uma experiência agradável e garantir a excelência do chope servido.

### 3.10 APLICAÇÃO

No sistema o backend foi desenvolvido com base em uma aplicação já existente chamada go10x, que foi desenvolvida pelo aluno durante o estágio obrigatório. Essa aplicação proporcionou uma base sólida para a implementação do sistema, fornecendo recursos e funcionalidades necessárias para o controle e gerenciamento das operações.

Quanto ao frontend, ele foi desenvolvido em cima de uma aplicação já existente chamada sdl-view, também desenvolvida pelo aluno durante o estágio obrigatório. Essa aplicação, utilizando a biblioteca SDL2, permitiu a criação da interface gráfica que interage com os usuários do sistema de serviço de chope automatizado.

Essas escolhas foram feitas levando em consideração a familiaridade do aluno com as tecnologias e as características específicas das aplicações. Além disso, essas ferramentas foram selecionadas por oferecerem recursos adequados para a criação de uma interface gráfica intuitiva e de fácil utilização.

Dessa forma, a combinação do backend desenvolvido com base na aplicação go10x e do frontend baseado na aplicação sdl-view proporcionou uma integração eficiente e funcional entre os componentes do sistema.

### 3.11 DEVICE TREE SOURCE (DTS) E DEVICE TREE BINARY (DTB)

A DTS é uma representação estruturada em formato de texto utilizada pelo Linux para descrever a configuração de hardware de um sistema embarcado. Essa descrição inclui informações sobre os dispositivos periféricos, suas características, conexões e atributos específicos.

Juntamente com a DTS, é gerado o DTB, que é uma versão compilada e binária da DTS. O DTB contém as informações do hardware de forma otimizada para o kernel do Linux interpretar durante a inicialização do sistema.

A utilização da DTS e do DTB no Linux é especialmente importante em sistemas embarcados, nos quais o hardware pode variar consideravelmente de uma placa para outra. Ao invés de ter drivers de dispositivo específicos para cada placa, a DTS permite uma abstração do hardware, descrevendo-o de maneira genérica e permitindo que o kernel do Linux se adapte às diferentes configurações de hardware.

A DTS é escrita em uma linguagem de descrição de hardware específica, baseada na linguagem C, com uma sintaxe bem definida. Ela descreve a hierarquia de

dispositivos e suas propriedades, como endereços de memória, interrupções, registros de configuração e outros detalhes relevantes para o funcionamento dos dispositivos.

Através da DTS, é possível definir e configurar os dispositivos do sistema, permitindo que o kernel Linux os reconheça e os configure corretamente durante a inicialização. Essa abordagem proporciona uma maior flexibilidade e portabilidade dos sistemas embarcados, simplificando o suporte a diferentes placas e variantes de hardware no ambiente Linux.

## 4 REQUISITOS E DESCRIÇÃO CONCEITUAL DA SOLUÇÃO PROPOSTA

### 4.1 REQUISITOS GERAIS, FUNCIONAIS E NÃO FUNCIONAIS

A solução proposta visa atender a uma série de requisitos para o desenvolvimento do sistema. Esses requisitos incluem:

- Capacidade de servir copos de chope de forma autônoma e repetida, garantindo rapidez e eficiência no atendimento.
- Adaptabilidade a diferentes sistemas de refrigeração com torneiras, como câmaras frias e chopeiras, proporcionando versatilidade de aplicação.
- Facilidade de montagem e desmontagem do sistema, permitindo uma instalação rápida e simplificada.
- Compactação do sistema para otimização do espaço e facilidade de transporte, quando necessário.
- Robustez do sistema, garantindo durabilidade e resistência a diferentes condições de operação.
- Design atraente, proporcionando uma experiência visual agradável aos usuários do sistema.
- Facilidade de configuração e ajuste, permitindo que pessoas com pouca experiência técnica possam realizar as regulagens necessárias.

### 4.2 DESCRIÇÃO CONCEITUAL DO FUNCIONAMENTO

Para garantir o funcionamento correto do sistema, é essencial que a linha de chope tenha os componentes em uma ordem específica. A sequência correta é a seguinte: primeiro, o barril de chope, seguido pelo poupa chope, o sensor de fluxo, a solenoide, caso exista, a chopeira e por fim, a torneira onde o chope será servido.

O funcionamento proposto do sistema é baseado na interação entre o cliente, o sistema de pagamento e controle de créditos, e a torneira de chope. O cliente utiliza um cartão ou QR Code, que é validado para verificar a disponibilidade de créditos para o serviço em uso. Caso o cartão seja validado com sucesso e o crédito seja descontado, o sistema inicia o processo de servir o chope.

Para servir o chope, o motor se move para frente, abrindo a solenoide da torneira para permitir a passagem do líquido. A quantidade de cerveja servida é monitorada através do uso de um sensor de fluxo. Quando a quantidade desejada é atingida, o motor se move para trás, servindo espuma da mesma forma. Após esse processo, a

posição do motor é ajustada para uma posição central neutra, evitando vazamentos, e a válvula solenoide é fechada.

Caso o barril acabe antes da finalização do serviço, o sistema verifica essa situação ao constatar que o sensor de fluxo não está contando mais a passagem de líquido enquanto está na posição de servir. Nesse caso, o sistema exibirá uma mensagem de erro e solicitará a intervenção de um atendente para resolver a situação.

Essa abordagem visa oferecer uma experiência autônoma e eficiente aos clientes, garantindo que eles possam desfrutar do chope pelo qual pagaram.

Com isso, a solução proposta atende aos requisitos estabelecidos, oferecendo uma solução robusta, de fácil montagem e desmontagem, compacta e com um design atraente. Além disso, o sistema é configurável e pode ser ajustado por pessoas com pouca expertise técnica, proporcionando facilidade de uso e manutenção.

No próximo tópico, serão apresentados os detalhes da arquitetura do sistema, incluindo os componentes envolvidos e os fluxos de informações e materiais.

### 4.3 ARQUITETURA E COMPONENTES ENVOLVIDOS

A arquitetura do sistema é composta por vários componentes que trabalham em conjunto para oferecer o serviço de chope automatizado. O principal dispositivo é a TAP, que consiste em uma placa com tela, um microcontrolador e um leitor NFC/QR Code. Essa TAP é responsável pela interação com o cliente, leitura de cartões e QR Codes, e controle do processo de servir o chope.

Além da TAP, o sistema conta com o motor e a estrutura mecânica, incluindo sensores de presença responsáveis por capturar o posicionamento do copo embaixo da torneira. Esses componentes são responsáveis por abrir a torneira de chope de acordo com as instruções da TAP. Além disso, temos a solenoide, que controla a abertura do fluxo de chope, e o sensor de vazão, que monitora a quantidade de líquido servido.

A integração do sistema de pagamento é realizada de duas formas. Para o serviço normal da myTapp, o cliente utiliza o cartão, que é validado pelo CMS para verificar a disponibilidade de créditos. Caso haja créditos suficientes, o CMS libera o serviço. Para a opção de QR Code, o sistema integra-se com um parceiro externo responsável pela validação do QR Code, verificando sua autenticidade e outras informações relevantes.

Os componentes do sistema estão conectados à TAP por meio de cabos, permitindo o processamento dos sinais dos sensores e a atuação dos atuadores. O motor de passo é controlado por um driver específico, enquanto o PWM utilizado para o controle é nativo do Toradex.

A comunicação com o sistema de pagamento ocorre por meio de APIs, permitindo a validação dos cartões e a liberação do serviço. Essa comunicação é feita pelo

CMS, que desempenha um papel importante na integração do sistema, armazenando informações sobre as cervejas, registros de consumo e configurações das torneiras.

A comunicação entre o CMS e o backend da TAP é realizada por meio de uma API, enquanto a comunicação entre o backend e a interface gráfica da TAP ocorre por meio de sockets, utilizando JSONs como formato de dados.

Para facilitar a montagem em massa, a estrutura mecânica foi projetada para ser impressa em 3D, permitindo uma fabricação mais rápida e eficiente.

Essa arquitetura do sistema atende aos requisitos gerais e funcionais estabelecidos, proporcionando um serviço de chope automatizado de forma rápida, confiável e eficiente.

## 5 DESENVOLVIMENTO

O processo de desenvolvimento foi conduzido principalmente pelo aluno, com a orientação e apoio de seu supervisor, um engenheiro de computação. Também contou-se com a participação do CEO da empresa, um engenheiro elétrico e a ajuda de um terceirizado, engenheiro mecânico.

A ideia para este projeto surgiu a partir da experiência da myTapp ao tentar implementar o sistema de auto serviço tradicional em um evento. Durante essa tentativa, foram identificadas diversas dificuldades, como a ocorrência frequente de pessoas tentando se servir com a solenoide da torneira fechada, resultando na entrada de ar na linha de chope. Além disso, constatou-se que muitos participantes do evento não tinham conhecimento adequado sobre o funcionamento correto de uma torneira de chope, o que resultava em atrasos e problemas no serviço.

Diante dessas dificuldades, surgiu a ideia de automatizar o processo de serviço de chope, reduzindo a interação direta dos usuários com a torneira. Essa solução visava minimizar os erros cometidos pelos participantes e garantir um serviço mais eficiente e livre de problemas. Com a automação, a myTapp buscou criar um sistema que oferecesse uma experiência de auto serviço de chope mais segura, simples e intuitiva para os usuários.

A partir dessa necessidade identificada, o projeto evoluiu para o desenvolvimento de um sistema completo de serviço de chope automatizado para eventos. Essa abordagem inovadora permitiu que a myTapp superasse as limitações encontradas no modelo tradicional de auto serviço e oferecesse uma solução mais eficaz e satisfatória tanto para os organizadores de eventos quanto para os participantes.

### 5.1 VERSÃO 1.0

#### 5.1.1 Validações com Arduino

A versão 1.0 do projeto foi concebida com o objetivo de criar uma solução de implementação simples e rápida. A ideia inicial foi utilizar um atuador linear como meio de empurrar a torneira, e para isso foram realizadas as primeiras validações utilizando um Arduino e um mini atuador linear (FIT080X... , s.d.) com um curso de 1cm. Além disso, foi utilizada uma ponte H baseada no L293D (TEXAS INSTRUMENTS, s.d.) para controlar o atuador.

No entanto, durante os testes iniciais, percebeu-se que o curso do atuador não seria viável para posicionar adequadamente a torneira nas posições de servir cerveja e espuma. Além disso, o atuador mais simples não possuía um controle fino de posição, o que limitava suas possibilidades de funcionamento. Ele suportava apenas dois modos: completamente retraído ou completamente exposto. Essa limitação inviabilizava

sua utilização, uma vez que mesmo com a solenoide fechada, ainda poderia haver vazamentos se a torneira não estivesse completamente fechada.

Com relação à escolha da ponte H, o shield utilizado necessitava de um padrão de comunicação específico para definir o endereço dos comandos antes de ativar o motor. Embora fosse uma opção simples de implementar em um Arduino com bibliotecas prontas, sabíamos que seria difícil reproduzir esse padrão de comunicação em um sistema com Linux.

Diante dessas limitações, iniciamos a busca por opções de atuadores que tivessem um curso maior e algum tipo de realimentação de posição. Após avaliar diferentes modelos, especialmente na China, solicitamos algumas amostras e decidimos utilizar o atuador PLARIU de 24V com curso de 5cm (23).

A escolha de um modelo com 24V foi motivada pela intenção de reduzir a corrente utilizada no motor, o que poderia potencialmente aumentar a vida útil da ponte H. Para acionar esse atuador, foi necessário substituir a ponte H baseada no L293D, uma vez que ela possui um limite de tensão de 6V. Encontramos facilmente um substituto baseado no L298 (SPARKFUN ELECTRONICS, s.d.), que suporta tensões de até 35V.

Para detectar a posição do copo e garantir o correto acionamento do mecanismo de serviço, era necessário utilizar sensores adequados. Inicialmente, consideramos a possibilidade de utilizar sensores de presença ou ultrassônicos com um alcance mais amplo. No entanto, levando em conta a eventual implementação do sistema em ambientes mais abertos onde haveria maior chance de um falso positivo, optamos por utilizar sensores infravermelhos, que são mais direcionais.

Durante nossa busca por sensores adequados, levamos em consideração a necessidade de um sensor que fosse facilmente instalado, sem a necessidade de componentes adicionais. Foi então que encontramos o sensor infravermelho E18-D80NK-N (ETT CO., LTD., s.d.), que atendeu aos nossos requisitos. Além de atender às nossas necessidades, esse sensor também possui a vantagem de permitir a regulação da distância de detecção e requer apenas um GPIO para realizar a leitura do sinal.

Utilizando dois pinos de PWM para controlar a velocidade e o sentido do motor, e um pino ADC para realizar a leitura da posição, pudemos realizar testes mais abrangentes com o produto. Durante esses testes, identificamos alguns problemas, como o alto ruído na leitura da posição e o fato de que o motor continuava se movendo por uma pequena distância mesmo quando o sinal de controle era desligado. Esse problema pode ser atribuído às características intrínsecas de um indutor, que gera um pico de corrente no mesmo sentido quando o sinal de controle do motor é desativado devido à variação abrupta de tensão. A combinação desses dois problemas dificultava o posicionamento preciso do motor na posição neutra.

Para solucionar esse problema, implementamos uma função que reduzia gra-

dualmente a velocidade do motor à medida que ele se aproximava de sua posição de destino, diminuindo a taxa de variação. Isso permitiu um movimento mais suave e preciso, evitando que o motor ultrapassasse a posição desejada. Além disso, para a leitura da posição, implementamos uma média das últimas cinco leituras, o que ajudou a reduzir o efeito de possíveis picos de ruído que pudessem ocorrer durante a leitura. Essa abordagem proporcionou uma leitura mais estável e confiável da posição do motor.

Dessa forma, o próximo passo seria a validação dos componentes utilizando o microcontrolador Toradex, que é usado nas placas da myTapp.

### 5.1.2 Validações com Toradex

De acordo com o datasheet do Toradex IMX6ULL (TORADEX, s.d.), o microcontrolador utilizado oferece um total de 196 pinos de Entrada e Saída disponíveis. Para operar o atuador linear, controlando sua velocidade e direção, e também para fazer a leitura de sua posição e de dois sensores, seriam necessários 2 pinos com suporte a PWM, 1 pino com suporte a ADC e 2 pinos GPIO. No entanto, considerando a possibilidade de escalar o produto e utilizar 2 motores por dispositivo, seriam necessários o dobro desses recursos, ou seja, 4 pinos PWM, 2 pinos ADC e 4 pinos GPIO.

Além disso, foi considerada a utilização de 2 sensores por motor, visando garantir uma maior precisão na identificação do copo.

Ao verificar os pinos disponíveis no Toradex, constatamos que alguns dos PWM já estavam sendo utilizados por outros componentes da placa, restando apenas 2 pinos com suporte PWM.

Diante dessa limitação, buscamos uma solução viável para controlar cada motor utilizando apenas um pino PWM e um pino GPIO. Como haviam mais GPIOs disponíveis no Toradex. Então, após validar esse tipo de acionamento, separamos 6 pinos GPIO, 2 pinos ADC e 2 pinos PWM para uso nesse projeto. Dessa forma, conseguimos contornar a restrição de recursos e seguir conforme o planejado.

Para utilizar os pinos da placa, é necessário primeiro configurar seus modos de operação no DTS. Essa configuração é realizada através da modificação dos arquivos DTS base da placa, como mostrado na figura 3. Esses DTS modificados são então compilados em arquivos DTB, que podem ser usados para reconfigurar a placa, garantindo o funcionamento esperado dos pinos.

Mesmo após a configuração dos pinos, é necessário acioná-los para que tenham a saída correta. No caso do PWM, é preciso enviar um byte para o arquivo "export" a fim de que o Linux crie o sistema de arquivos para as configurações necessárias. Em seguida, é possível definir o período do sinal em nanossegundos, ajustar o duty cycle também em nanossegundos e, por fim, ativar o sinal. Um exemplo de implementação em Go, utilizado na aplicação, pode ser visualizado na figura 4.



Figura 3 – Ativando PWM via DTS.

```
/* PWM <D> */
&pwm7 {
-   status = "disabled";
+   status = "okay";
};
```

Figura 4 – Código para configuração do PWM.

```
// Setup PWM with default period of 1MHZ
func (motor *Motor) PWMsetup(pwm string) {
    err := os.WriteFile("/sys/class/pwm/pwmchip"+pwm+"/export", []byte("0"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }

    err = os.WriteFile("/sys/class/pwm/pwmchip"+pwm+"/pwm0/period", []byte("100000"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }

    err = os.WriteFile("/sys/class/pwm/pwmchip"+pwm+"/pwm0/duty_cycle", []byte("500000"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }

    err = os.WriteFile("/sys/class/pwm/pwmchip"+pwm+"/pwm0/enable", []byte("1"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
}
```

Na implementação do ADC, é necessário apenas configurar a frequência de leitura dos dados em nanossegundos. Por meio de experimentos, observamos que o valor de 7541 ns proporcionou uma leitura com menos ruído no microcontrolador. Esse valor foi utilizado na aplicação, conforme ilustrado na figura 5.

Figura 5 – Código para configuração do ADC.

```
// Setup sampling frequency for ADC (ideal = 7541)
func (motor *Motor) ADCsetup() {
    err := os.WriteFile("/sys/bus/iio/devices/iio:device0/in_voltage_sampling_frequency", []byte("7541"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
}
```

Também é necessário configurar os pinos GPIO, tanto como saída (definindo a direção como "OUT" e o valor de saída no arquivo "value") quanto como entrada (definindo a direção como "IN" e realizando a leitura do valor por meio do arquivo "value"). Ambas as implementações em Go podem ser visualizadas nas figuras 6 e 7.

Figura 6 – Código para configuração de saída do GPIO.

```
// Setup a GPIO as output, signaling high
func (motor *Motor) SetGPIOout(gpio string) {
    err := os.WriteFile("/sys/class/gpio/export", []byte(gpio), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
    err = os.WriteFile("/sys/class/gpio/gpio"+gpio+"/direction", []byte("out"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
    err = os.WriteFile("/sys/class/gpio/gpio"+gpio+"/value", []byte("1"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
}
```

Figura 7 – Código para configuração de entrada do GPIO.

```
// Setup a GPIO as input
func (motor *Motor) SetGPIOin(gpio string) {
    err := os.WriteFile("/sys/class/gpio/export", []byte(gpio), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
    err = os.WriteFile("/sys/class/gpio/gpio"+gpio+"/direction", []byte("in"), 0644)
    if err != nil {
        fmt.Printf("Erro: %s", err.Error())
    }
}
```

Após a realização dessas configurações finais, validamos o funcionamento de todos os pinos necessários para acionar corretamente os atuadores e sensores no Toradex.

Em seguida, tentamos ativar o atuador implementando uma lógica semelhante à utilizada no Arduino, porém, não obtivemos sucesso. Foi nesse momento que identificamos um problema: o nível lógico alto dos GPIOs do Toradex é de 3.3V, enquanto o shield de ponte H que estávamos utilizando necessitava de 5V para ser ativado.

Diante dessa incompatibilidade, precisamos encontrar um componente que atendesse às nossas necessidades.

Após uma pesquisa, encontramos um shield baseado no BTS7960 (BTS7960... , s.d.), que supria as nossas exigências de tensão de ativação do motor. Esse novo shield foi selecionado como substituto adequado para o nosso sistema.

Após adquirirmos o novo shield, pudemos verificar o funcionamento dos componentes em conjunto com o Toradex. Com essa validação concluída, demos início ao processo de desenvolvimento da lógica no código.

Após a validação de todos os componentes, realizamos uma série de testes internos para avaliar o desempenho do sistema. Esses testes foram conduzidos tanto em chopeiras, onde o atuador seria fixado na parte superior e acionado por uma alavanca, conforme ilustrado na figura 8, quanto em uma geladeira adaptada com barris de chope, simulando uma câmara fria, como mostrado na figura 9.

O protótipo do sistema acoplado à chopeira nos permitiu verificar a funcionalidade do produto em um ambiente mais próximo ao cenário real de uso. Com base nos resultados obtidos, confirmamos que o sistema estava operando de maneira adequada e atendendo aos requisitos estabelecidos. Essa etapa foi crucial para garantir que o sistema fosse capaz de realizar o serviço de chope de forma eficiente e confiável.

Além disso, realizamos testes de estresse ao instalar o sistema em uma geladeira adaptada com barris de chope. Essa configuração simulava as condições de uma câmara fria, similar ao ambiente de operação da empresa. Esses testes nos permitiram avaliar o desempenho do sistema em condições mais desafiadoras, como variações de temperatura e umidade. Os resultados obtidos nessas condições extremas foram analisados para garantir a robustez e a estabilidade do sistema em diferentes cenários de uso.

Apesar do sucesso dos testes, acreditamos que essa solução não seria implementada devido à necessidade de realizar furos adicionais na estrutura. Assim, ficou como uma melhoria futura buscar alternativas que não comprometessem a integridade da câmara fria.

Após várias semanas de testes internos e aperfeiçoamentos, o produto teve sua primeira atuação em campo durante a Oktoberfest 2022. Nessa ocasião, foram feitas algumas adições estéticas e uma modificação no posicionamento do sensor infravermelho. Originalmente, o sensor estava localizado atrás da torneira, mas decidimos movê-lo para a lateral do sistema, como pode ser observado na Figura 10. Essa alteração proporcionou uma melhor detecção de copos e contribuiu para a eficiência do serviço de chope.

Figura 8 – Montagem de protótipo em chopeira.



### 5.1.3 Programa e lógica de ativação

No código em Go, é importante criar structs para manter a organização do código. Esses structs são usados para criar tipos que contêm outras variáveis. Para essa implementação específica, foram criados três structs: Motor, Sensor e Converter que são apresentados no apêndice A.

Esses structs também são utilizados para definir as hierarquias dos objetos, permitindo identificar que cada motor possui 2 sensores e um leitor ADC, além de especificar quais variáveis de controle pertencem a cada componente. Essa estrutura organizada facilita o gerenciamento e a compreensão do código, proporcionando uma melhor estruturação e clareza nas relações entre os elementos do sistema.



Figura 9 – Montagem de protótipo em um análogo a câmara fria.



Com a integração da lógica do atuador no código principal, foram realizados refinamentos em seu funcionamento. Alguns dos aprimoramentos implementados incluem:

1. Modo de Testes: Foi implementado um modo de testes que permite que o motor execute repetidamente o movimento de servir, simulando o atendimento a várias pessoas. Esse modo de teste tem como objetivo medir a robustez do sistema, testando sua capacidade de realizar operações contínuas de forma confiável.
2. Função de Calibração: Foi adicionada uma função de calibração que permite que o usuário meça os valores mínimo e máximo do potenciômetro no motor. Essa calibração proporciona uma referência mais precisa para o controle do motor, permitindo um ajuste mais preciso da posição e velocidade.

Figura 10 – Montagem do produto na Oktoberfest.



Além das funcionalidades mencionadas anteriormente, outras funções importantes foram implementadas para aumentar a robustez e confiabilidade do sistema. Essas funções têm como objetivo garantir um desempenho consistente e estável do atuador. Algumas das principais funcionalidades adicionais são:

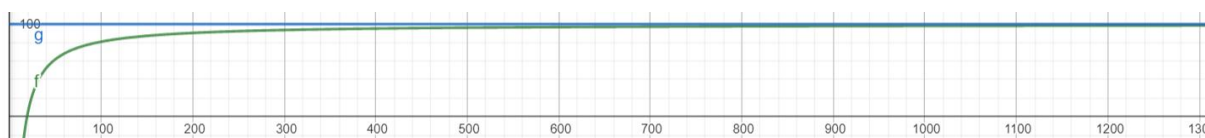
1. **SensorContinuousRead:** Esta função é executada continuamente em segundo plano e é responsável por fazer a leitura dos sensores para identificar a presença de um copo. Para evitar falsos positivos, o sensor considera uma mudança de estado apenas quando faz 5 leituras seguidas com a mesma informação. Isso aumenta significativamente a robustez do sistema, garantindo uma detecção confiável dos copos.
2. **GoNeutral:** Essa função tem a finalidade de posicionar o motor na posição neutra, onde a torneira está fechada. Primeiramente, ela realiza um cálculo com base

em uma regra de três para determinar o valor de tensão lido pelo potenciômetro que representa a posição neutra. Essa regra de três é calculada com base nas informações de distância configuradas durante a instalação. Com o valor de referência calculado e conhecendo a posição atual do motor, a função determina a direção em que o motor deve se mover. Para isso, a velocidade do motor, controlada pelo duty cycle do PWM, é ajustada com base na distância até o alvo. A função utiliza uma equação que proporciona uma diminuição rápida da velocidade quando o motor está próximo do destino, reduzindo a interferência e garantindo uma parada suave quando a posição alvo é alcançada. Essa função é fundamental para manter o atuador na posição neutra com alta precisão.

A equação utilizada para calcular a velocidade do motor enquanto se move em direção à posição neutra é representada pela Equação 1 e ilustrada graficamente na Figura 11. Nessa equação, o eixo X representa o módulo da distância até o objetivo, enquanto o eixo Y representa o duty cycle do motor, com um valor máximo de 100. Observa-se que a equação possui uma interferência mínima para valores de X distantes de zero, mas resulta em uma diminuição drástica da velocidade quando se aproxima do destino.

$$\frac{-1}{\text{diff}} * 1900 + 100 \quad (1)$$

Figura 11 – Representação da velocidade por distancia.



#### 5.1.4 Interface gráfica

A interface gráfica do sistema utiliza a mesma aplicação desenvolvida durante o estágio obrigatório pelo aluno. Essa aplicação é escrita em C++ e utiliza a biblioteca SDL2 para a geração das telas. Para criar as telas de eventos, uma nova classe foi desenvolvida com funções específicas para esse modo. O backend do sistema se comunica com essa aplicação por meio de sockets, enviando comandos em formato JSON que representam a tela desejada, juntamente com as informações necessárias para desenhar essa tela. Essa comunicação permite que o backend atualize dinamicamente a interface gráfica de acordo com o estado atual do sistema, proporcionando uma experiência interativa e informativa para o usuário. Algumas das telas podem ser observadas nas figuras 12, 13 e 14.

Figura 12 – Tela de repouso.

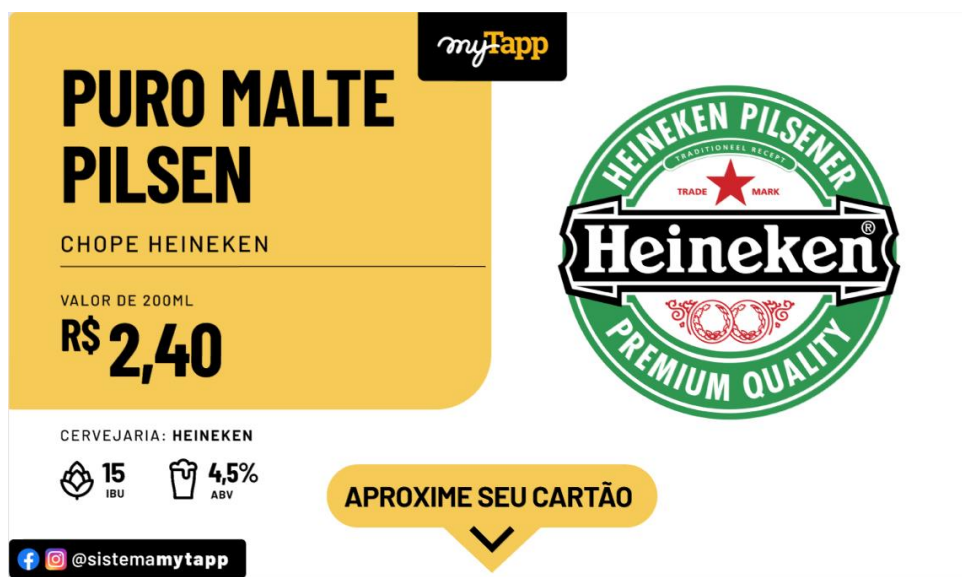
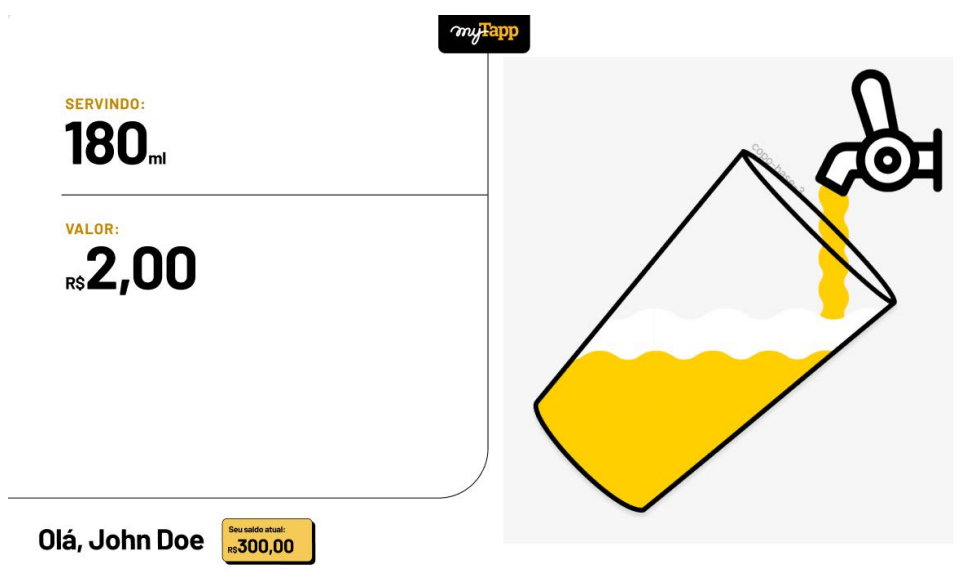


Figura 13 – Tela servindo.



### 5.1.5 Integração com o cashless

Uma das motivações para o desenvolvimento deste projeto foi possibilitar a integração com a NetPDV, uma empresa de soluções cashless presente nos principais eventos do Brasil. A NetPDV possui sua própria solução em que todas as informações, incluindo créditos, são armazenadas em cartões protegidos por camadas de criptografia. Para viabilizar a integração entre a myTapp e a solução da NetPDV, foi necessário desenvolver um software adicional com o único propósito de realizar a leitura e escrita nos cartões.



Figura 14 – Tela de aviso: "Acabou o barril".



O processo para gerar um consumo no sistema da NetPDV é descrito na Figura 15. Basicamente, envolve a leitura de blocos específicos do cartão, o envio desses dados para o servidor da NetPDV por meio de uma API e o recebimento de blocos atualizados para sobrescrever no cartão, efetuando assim o desconto do saldo correspondente.

## 5.2 VERSÃO 2.0

Após o sistema ter sido utilizado em diversos eventos e com as observações feitas, discutidas mais amplamente no capítulo de discussão de resultados, chegamos à conclusão de que não era viável continuar utilizando um atuador linear para movimentar a torneira. Diante dessa necessidade de busca por uma nova solução, consideramos duas opções: o servo motor e o motor de passo.

Inicialmente, focamos nas validações com o servo motor, uma vez que suas características pareciam perfeitas para serem incorporadas ao sistema. O servo motor proporciona um movimento de rotação com alta confiabilidade de controle de posição, o que permitiria acoplá-lo diretamente ao eixo da torneira por meio de uma alavanca.

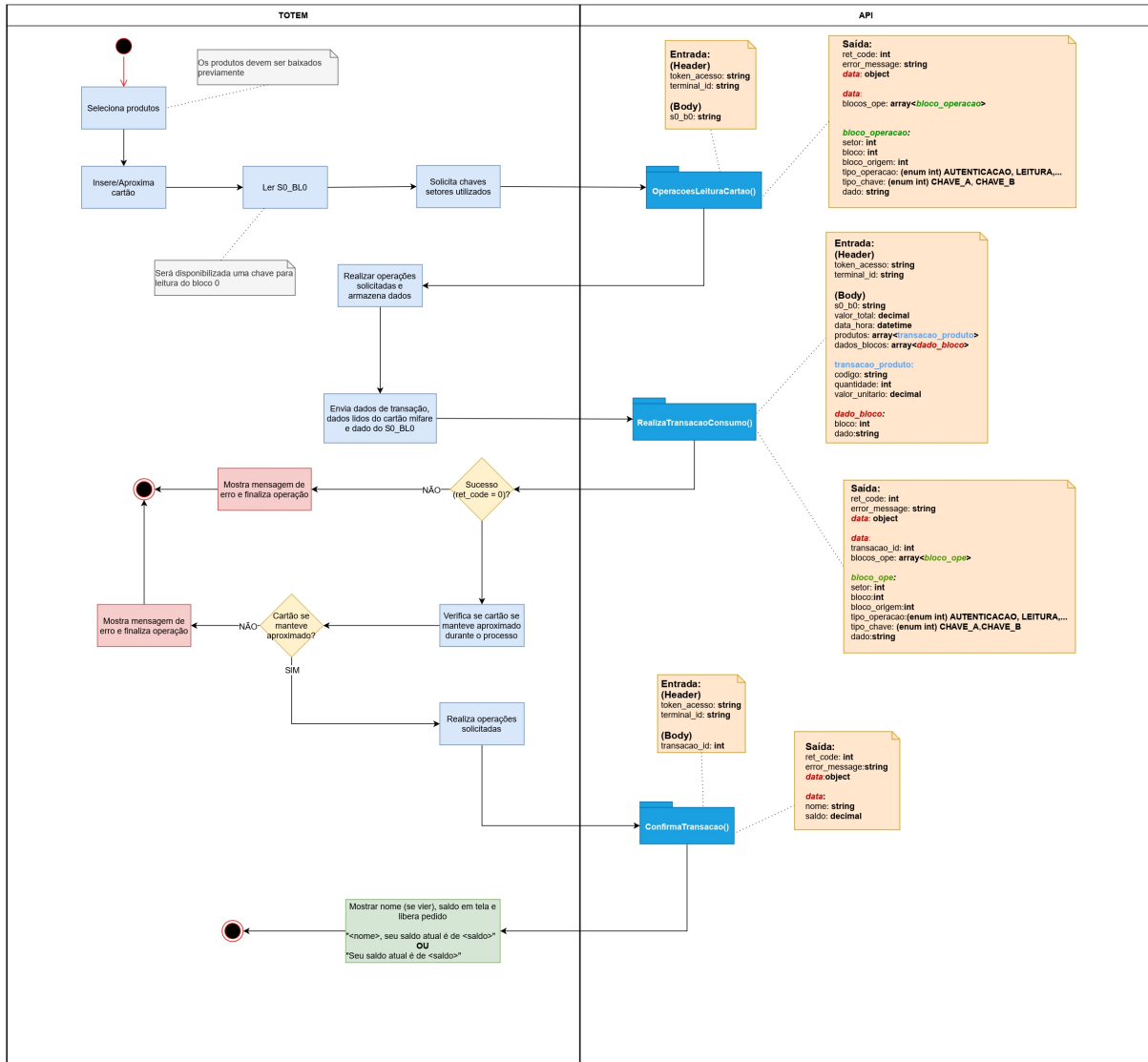
No entanto, nos deparamos com um problema: a maioria dos servos disponíveis no mercado era do tipo micro servo, que não possuíam força suficiente para movimentar a torneira. Embora existissem servos maiores, capazes de exercer a força necessária, o preço desses componentes era demasiadamente elevado, tornando o produto final inviável comercialmente.

A outra opção, o motor de passo, também apresentava suas vantagens devido ao seu funcionamento baseado no controle por passos. Acreditávamos que poderia-

Figura 15 – Fluxo para gerar consumo em cartão netPDV.

## Fluxo de transação de consumo entre Totem parceiro e API netPDV

v1.02.000



Fonte: Arquivo myTapp (2022).

mos alcançar um nível elevado de precisão utilizando esse tipo de motor. No entanto, tínhamos ciência de que o motor de passo exigiria um acoplamento mecânico, o que demandaria a participação de um profissional mais qualificado na área para sua execução.

Após realizar alguns testes internos e considerando nossa maior preocupação com a confiabilidade, bem como nossa experiência com o desgaste do controle de posição por potenciômetro, optamos por utilizar o motor de passo com acoplamento mecânico como solução para a versão 2.0 do sistema. Essa escolha nos proporcionou um nível de precisão elevado e uma maior confiabilidade no controle do movimento da

torneira.

### 5.2.1 Seleção do motor de passo e projeto estrutura mecânica

Após definir a solução a ser utilizada, iniciamos o projeto para o sistema. Para isso, contratamos um profissional terceirizado para desenvolver o projeto mecânico e nos auxiliar na escolha do modelo de motor de passo a ser utilizado.

Primeiramente, realizamos a modelagem 3D da torneira acoplada a uma chopeira, com o objetivo de permitir que o sistema pudesse ser fixado de forma segura diretamente na torneira, independentemente de estar acoplada a uma chopeira ou a uma câmara fria, por exemplo.

Com essa premissa em mente, foi recomendado fazer o acoplamento mecânico do motor de passo com a torneira utilizando engrenagens. Essa abordagem nos permitiria controlar a força necessária por meio da relação de engrenagens, solucionando nossa preocupação de que o motor de passo não teria força suficiente para movimentar a torneira.

Assim, foi projetada a primeira versão da estrutura de acoplamento, conforme ilustrado na figura 16. Essa versão consistia em uma estrutura que poderia ser presa à torneira por meio do uso de duas flanges, que por sua vez eram conectadas à estrutura de acoplamento. Essa estrutura era composta por várias peças fixadas entre si por parafusos, incluindo a parte com o motor de passo com uma engrenagem em seu eixo, ligada a outra engrenagem presa à torneira. Dessa forma, ocorria o acoplamento entre ambas. Na engrenagem presa à torneira, eram incluídos dois limitadores nas extremidades, que impediam que a engrenagem do motor de passo saísse do acoplamento. Esses limitadores eram fixados à engrenagem maior por meio de parafusos.

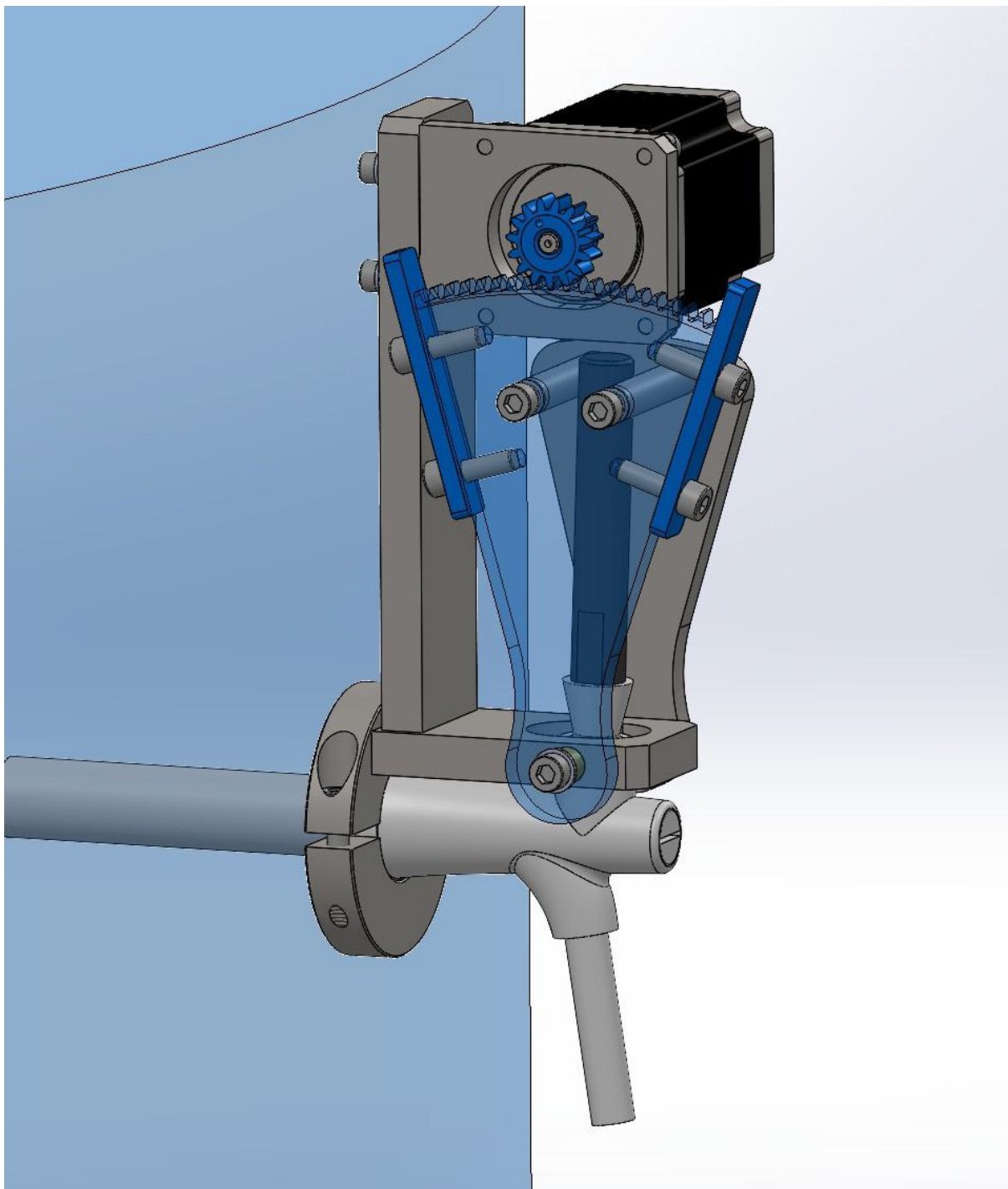
Após a apresentação dessa versão, identificamos algumas questões relacionadas à sua construção. Primeiramente, havia a preocupação de que os limitadores não desempenhassem efetivamente seu papel e pudessem ser destruídos caso o motor tentasse sair do curso previsto. Portanto, decidimos modificar o projeto para que a engrenagem maior não tivesse limitadores presos a ela.

Além disso, consideramos a ideia de fabricar o máximo possível dessa estrutura por meio da impressão 3D, a fim de facilitar sua produção. Outra questão levantada foi a quantidade de partes montadas na estrutura, o que ia contra a ideia de um sistema de montagem rápida. Por isso, decidimos que a maior parte da estrutura deveria ser uma única peça.

Com essas alterações em mente, foi criada uma versão revisada, conforme mostrado na figura 17. Essa versão também inclui um pequeno pino na parte traseira, com o objetivo de criar mais um ponto de apoio em qualquer estrutura em que esteja montado.

Com o projeto em mãos, imprimimos as peças e montamos o primeiro protótipo,

Figura 16 – Primeira versão da estrutura mecânica para acoplamento.

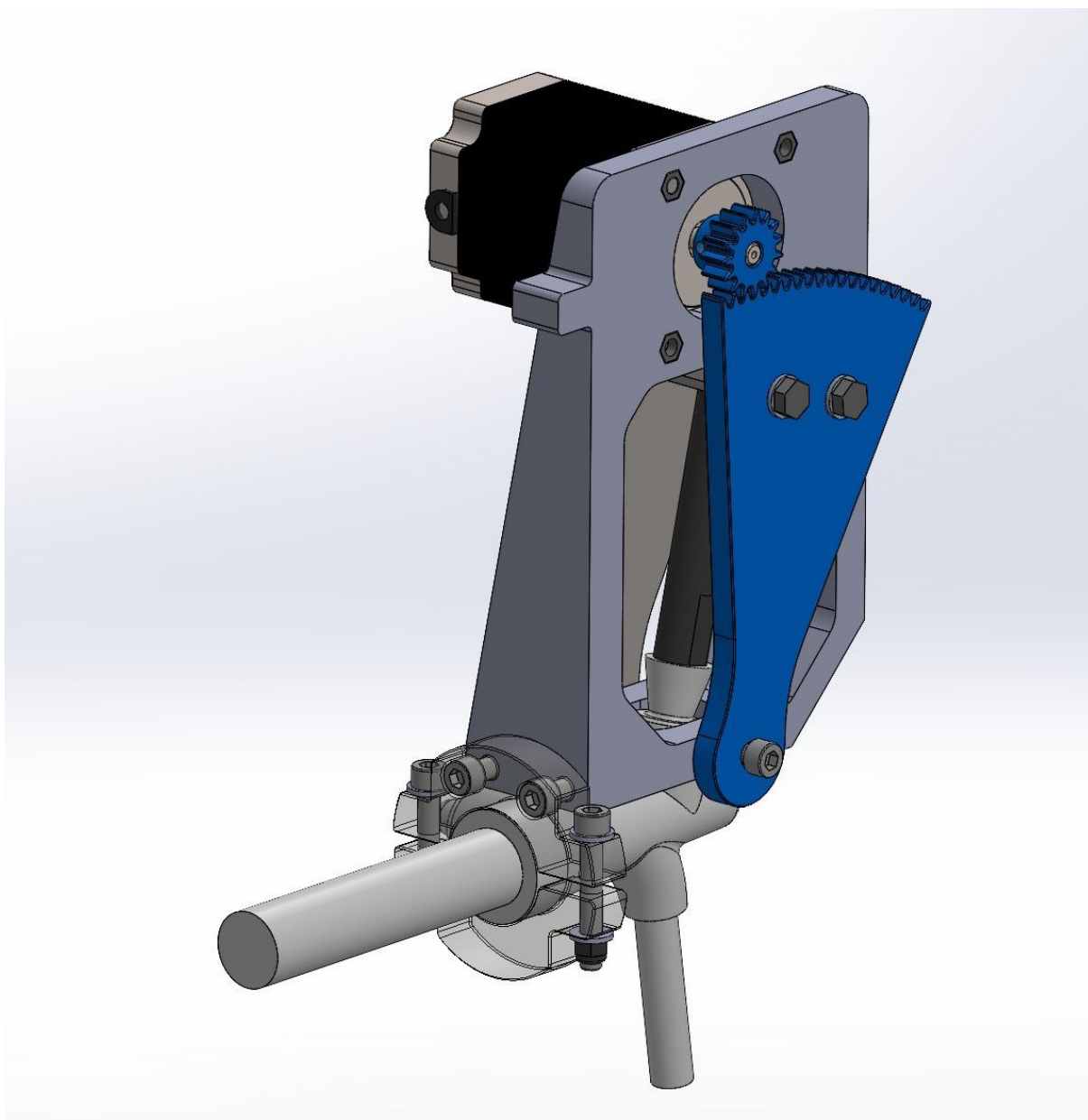


conforme mostrado na figura 18. Durante os testes realizados com esse protótipo, identificamos algumas questões que precisavam ser abordadas.

O primeiro problema observado foi o tamanho do protótipo. Tínhamos em mente um produto mais compacto, visando facilitar o transporte e a instalação em diferentes ambientes.

Além disso, notamos que havia um espaço considerável entre os parafusos que

Figura 17 – Versão revisada da estrutura mecânica para acoplamento.



seguravam a alavanca da torneira, criando uma folga indesejada. Isso comprometia a precisão do movimento e a estabilidade do sistema.

Esses problemas identificados nos ajudaram a refinar o projeto e realizar as melhorias necessárias para a próxima versão do protótipo. Buscamos otimizar o tamanho do dispositivo, reduzir a folga identificada, e aprimorar a ergonomia e a usabilidade geral do sistema.

Após realizar todas as validações necessárias com a estrutura, ficou evidente a necessidade de implementar algum tipo de realimentação de posição. Em torneiras mais rígidas, o motor não conseguia executar completamente seu movimento de recuo, resultando em pulos de passo e perda de referência. Além disso, percebemos que

Figura 18 – Primeiro protótipo funcional.



não havia espaço suficiente na estrutura para acomodar os sensores infravermelhos responsáveis pela identificação do copo. Para manter a ideia de um sistema que

pudesse ser totalmente acoplado à torneira, decidimos incluir um suporte para os sensores na parte frontal da estrutura.

Após testar o sistema com um motor NEMA-23, decidimos explorar outras opções para a estrutura. Durante os testes, constatamos que um motor de tamanho menor também seria adequado para o funcionamento do sistema. Dessa forma, optamos por utilizar um motor NEMA-17, que possui dimensões mais compactas e pode oferecer um desempenho satisfatório para as necessidades do projeto. Essa mudança nos permitiu reduzir o tamanho e o peso da estrutura, tornando-a mais eficiente e econômica.

Quanto à realimentação de posição, consideramos que a adição de um encoder ao projeto seria custosa demais. Portanto, optamos por incorporar chaves de fim de curso (FDC) nas extremidades do curso da torneira. Essas chaves permitem verificar sempre que a torneira atinge um dos extremos do movimento, proporcionando o reset da referência e reduzindo o impacto dos pulos de passo no sistema. A estrutura, com todas essas modificações, pode ser visualizada na figura 19.

Com a versão mais confiável da estrutura em mãos, iniciamos os testes de estresse. Durante esses testes, a estrutura em si apresentou um resultado excelente. No entanto, a parte que não se saiu tão bem foi a engrenagem acoplada diretamente ao eixo do motor de passo. Essa engrenagem era fixada ao eixo por meio de um único parafuso, que acabava espanando a rosca da engrenagem com o tempo, resultando em sua perda de fixação. Para solucionar esse problema, adicionamos mais dois furos na engrenagem, totalizando três, a fim de proporcionar uma fixação mais robusta e distribuir melhor as forças ao redor da engrenagem.

Além disso, optamos por imprimir a engrenagem utilizando um filamento 3D mais resistente, conhecido como Tritan (FILAMENTO... , s.d.), amplamente utilizado para peças mecânicas. Essas modificações resultaram na resolução completa dos problemas relacionados à estrutura mecânica.

Após definirmos a forma final da estrutura, projetamos um case impresso em 3D para proteger os mecanismos do sistema. Esse case inclui painéis de acrílico ao redor, permitindo que o usuário visualize o processo de serviço de chope. Uma imagem do case pode ser observada na figura 20. Essa adição proporciona segurança aos componentes internos do sistema, ao mesmo tempo em que oferece uma experiência visual ao usuário durante o serviço de chope.

### 5.2.2 Programa e lógica de ativação

A implementação do motor de passo trouxe algumas alterações significativas no código, principalmente no processo de calibração e no controle dos movimentos da torneira.

No novo código, a calibração não envolve mais a leitura de um ADC, como



Figura 19 – Estrutura com chave de FDC e apoio para os sensores.

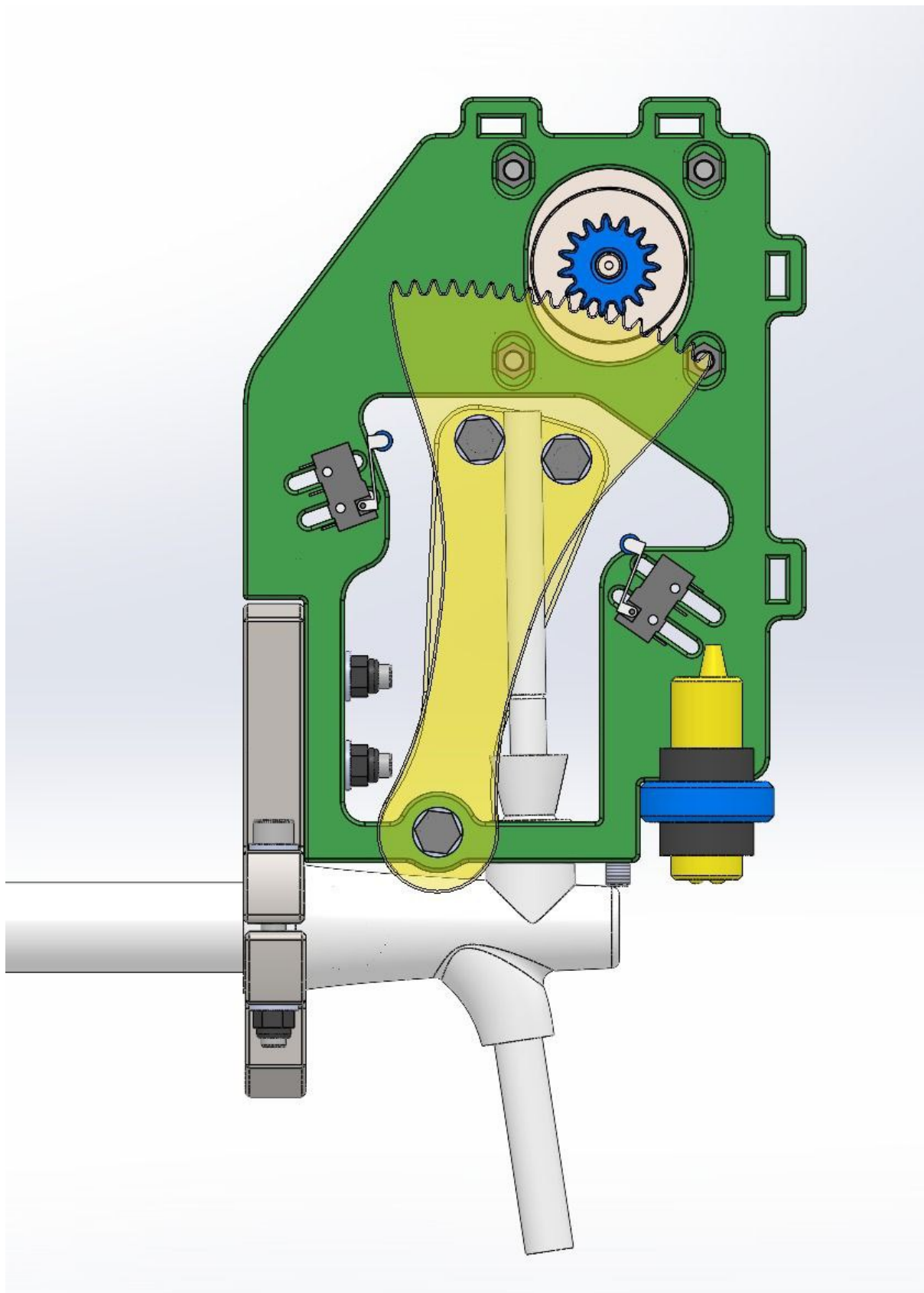




Figura 20 – Estrutura final com case.



anteriormente mencionado. Em vez disso, utiliza-se um processo baseado em tempo para determinar a posição dos fins de curso da torneira. Isso é feito movendo-se a torneira para trás até que ela alcance o fim de curso traseiro ou até que um tempo pré-determinado tenha decorrido. Essa abordagem garante que a torneira tenha atingido a posição final e proporciona uma referência confiável para os movimentos subsequentes.

Após a confirmação da posição traseira, inicia-se um temporizador e a torneira é movida para a frente até que atinja o fim de curso frontal. É importante ressaltar

que, para o correto funcionamento do sistema, é necessário que pelo menos o fim de curso frontal esteja sendo acionado. O fim de curso traseiro é opcional, dependendo do design e requisitos específicos do sistema.

Quando o fim de curso frontal é acionado, o temporizador é interrompido e o tempo decorrido é registrado. Com base nesse tempo, utilizando uma regra de três, é possível calcular aproximadamente quantos passos foram dados desde uma extremidade até a outra. Esse cálculo fornece informações cruciais sobre o tamanho completo do curso da torneira, permitindo ajustes precisos no funcionamento do motor de passo.

As funções responsáveis pelo movimento do motor passaram por alterações significativas. Agora, existem três funções principais: GoBeer, GoFoam e GoNeutral. Cada uma delas é responsável por controlar o movimento do motor de acordo com o estado desejado.

1. A função GoBeer() controla o movimento do motor para servir cerveja. Antes de iniciar o movimento, ela verifica o estado atual do motor. Se não estiver parado, o motor é parado e aguarda até que o movimento seja interrompido completamente. Em seguida, o tempo de execução é calculado com base no período e no número de passos do motor. Dependendo da posição atual do motor, o tempo de execução é ajustado. A função define a posição como "beer" e o estado como "forward"(avanço), e então inicia um loop. O loop é executado continuamente até que uma das seguintes condições seja atendida: (1) o sensor de FDC é acionado e ainda não foi lido, (2) o motor é parado ou (3) o tempo decorrido excede o tempo de execução mais 10% do tempo de execução. Quando uma dessas condições é atendida, o movimento é interrompido, o estado é definido como "stop" e o sensor de FDC é marcado como lido.
2. A função GoFoam() controla o movimento do motor para servir espuma. Assim como a função GoBeer(), ela verifica o estado atual do motor e o para, se necessário. O tempo de execução é calculado com base no número de passos e no período do motor, e é ajustado dependendo da posição atual do motor. A função define a posição como "foam" e o estado como "backward"(reverso), e então inicia um loop. O loop é executado até que uma das seguintes condições seja atendida: (1) o sensor de FDC é acionado e ainda não foi lido, (2) o motor é parado ou (3) o tempo decorrido excede o tempo de execução mais 10% do tempo de execução. Quando uma dessas condições é atendida, o movimento é interrompido, o estado é definido como "stop" e o sensor de FDC é marcado como lido.
3. A função GoNeutral() controla o movimento do motor para a posição neutra. Novamente, o estado do motor é verificado e ele é parado, se necessário. Dependendo da posição atual do motor, o movimento é definido como reverso ("beer-> "neu-

tral") ou avanço ("foam-> "neutral"). O tempo de execução é calculado com base no período e no número de passos do motor, e um loop é iniciado. O loop é executado até que uma das seguintes condições seja atendida: (1) o motor é parado ou (2) o tempo decorrido excede o tempo de execução. Quando uma dessas condições é atendida, a posição é definida como "neutral", o estado é definido como "stop" e o movimento é interrompido.

No código implementado, existem algumas condições adicionais nos loops de movimento do motor que merecem destaque. Além das condições que param o motor quando ele atinge um dos fins de curso, foram adicionadas duas condições extras. A primeira ocorre quando o estado do motor é alterado para "stop". Essa condição permite que outras partes do código garantam que o motor possa ser interrompido a qualquer momento, evitando problemas de movimentos simultâneos indesejados.

Além disso, foi incluída uma condição que interrompe o movimento caso o tempo de movimentação calculado seja excedido em mais de 10% e a chave de fim de curso não tenha sido acionada. Essa condição serve como uma garantia adicional para evitar que o motor continue movendo-se indefinidamente caso ocorra algum problema com as chaves de fim de curso ou qualquer outro imprevisto que impeça o acionamento adequado delas.

Essas condições extras foram implementadas com o objetivo de aprimorar a segurança e a confiabilidade do sistema. Elas fornecem uma camada adicional de proteção, garantindo que o motor possa ser interrompido de forma rápida e segura em diversas situações, evitando comportamentos indesejados ou possíveis danos ao sistema.

Dentro da lógica de leitura das chaves de fim de curso, há uma variável adicional chamada "wasRead". Essa variável desempenha um papel importante ao garantir que uma chave ativada por outro movimento não seja lida novamente. A cada movimento completo, a variável "wasRead" é alterada para verdadeiro, indicando que a chave foi lida e registrada. Por outro lado, sempre que a chave de fim de curso muda de estado, a variável "wasRead" é alterada para falso, permitindo que seja lida novamente quando necessário.

Essa abordagem garante que as chaves de fim de curso sejam adequadamente monitoradas e evita problemas de leituras duplicadas ou incorretas. A variável "wasRead" atua como um mecanismo de controle confiável para garantir a precisão na detecção do estado das chaves de fim de curso durante o funcionamento do sistema.

Dessa forma, o código foi projetado para levar em consideração não apenas o acionamento das chaves de fim de curso, mas também a necessidade de controle e segurança adicionais ao movimento do motor. Essas condições extras ajudam a assegurar um funcionamento mais confiável e evitam possíveis problemas decorrentes de situações imprevistas durante o processo de servir cerveja.

Também consideramos a frequência dos sinais de passo ao escolher os parâmetros para o motor de passo. Com base no gráfico de rendimento (DATASHEET... , s.d.), procuramos encontrar um valor que oferecesse um torque alto, mas ao mesmo tempo permitisse que o motor completasse o curso completo da torneira em poucos segundos. Observando o gráfico e realizando testes, determinamos que uma frequência de 150 MHz seria apropriada, o que corresponde a uma velocidade de rotação de 50 RPM. Levando em conta o uso de micropasso de 1/4, cada volta completa do motor requer 800 passos. Esses parâmetros foram selecionados para garantir um desempenho adequado e um movimento suave da torneira durante o serviço de chope.

### 5.2.3 Estrutura mecânica para atuador linear

Aproveitando a presença de alguém com experiência em projetos mecânicos, e considerando que a versão do sistema com o motor de passo é melhor do que a do atuador linear, a empresa ainda possui um grande estoque de kits de atuador linear e ponte H, adquiridos para eventos anteriores. Com o intuito de aproveitar esses recursos e permitir a utilização dos atuadores lineares, se necessário, desenvolvemos uma alavanca simples para aumentar o curso efetivo do atuador linear, mesmo em chopeiras mais altas. O projeto desse modelo de alavanca pode ser observado na Figura 21.

Após a conclusão do projeto, adquirimos um protótipo, que pode ser visto na Figura 22. Esse protótipo passou por testes e validações internas, e constatamos que ele cumpre a função pretendida, permitindo um curso maior do atuador linear e adequando-se às necessidades das chopeiras mais altas. Dessa forma, garantimos que os kits de atuador linear já adquiridos possam ser utilizados de maneira eficiente e aproveitando ao máximo os recursos disponíveis.

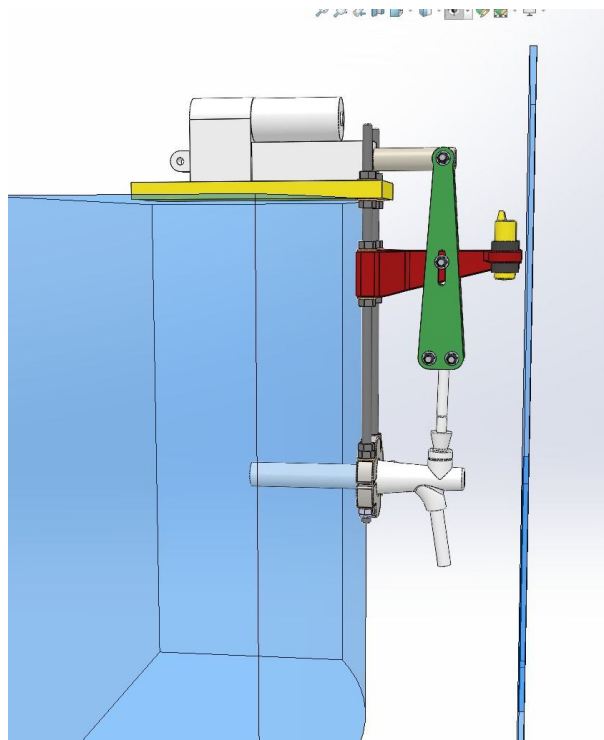
### 5.2.4 Problema da espuma

Também abordamos outro problema que afetava a qualidade do serviço de chope, que consistia em grandes intervalos entre as servidas, resultando no aquecimento da torneira e na geração excessiva de espuma durante o serviço, até que a torneira esfriasse novamente.

Para resolver esse problema, consideramos algumas soluções, sendo a mais comum a prática de sangria, em que a torneira serviria uma pequena quantidade de cerveja caso houvesse alguns minutos de inatividade, a fim de manter a temperatura adequada da torneira e, conseqüentemente, a qualidade do chope.

No entanto, percebemos que essa solução não seria a ideal, pois os clientes poderiam interpretar erroneamente o desperdício de chope, sem ter conhecimento técnico sobre a questão. Além disso, existia o risco de servir chope quando alguém estivesse se preparando para servir, o que poderia gerar confusão.

Figura 21 – Modelo de estrutura para melhorar funcionamento do atuador linear.



Diante disso, optamos por uma abordagem diferente. Implementamos um cálculo do tempo entre cada servida e, se o intervalo entre as servidas fosse inferior a 5 minutos, a quantidade de espuma e cerveja servidas permaneceria a mesma. No entanto, após esse intervalo, um percentual dos mililitros de espuma seria misturado à cerveja, até um limite de 30 minutos. Durante esse período, a torneira não se movimentaria para a posição de espuma, servindo apenas cerveja, uma vez que a espuma seria gerada pela alta temperatura da torneira.

Essa implementação foi realizada em Go, no código do sistema, e pode ser observada no Apêndice B. Essa solução contribuiu significativamente para melhorar a qualidade do serviço de chope ao evitar a geração excessiva de espuma e garantir uma temperatura adequada da torneira.

Figura 22 – Protótipo da estrutura para o atuador linear.



## 6 IMPLEMENTAÇÃO

### 6.1 VERSÕES

A primeira versão do sistema utilizava um motor linear elétrico para atuar na torneira. Esse atuador era equipado com um potenciômetro que permitia a leitura da posição do motor. Essa leitura era realizada por meio de um conversor analógico-digital (ADC). Devido às variações entre os motores, foi necessário implementar um processo de calibração. Esse processo consistia em acionar o atuador por meio da ponte H, movendo-o em uma direção e depois na outra. Após cada movimentação, era feita a leitura do nível de tensão no ADC, permitindo obter os valores mínimo e máximo do ADC. Com base nesses valores, era possível realizar uma simples regra de três para calcular a posição do atuador em qualquer ponto, considerando a linearidade do potenciômetro.

O atuador linear da primeira versão tinha um curso de 5 cm e, por padrão, considerava-se a posição 0 cm como a de servir espuma e a posição 5 cm como a de servir chope. A posição neutra era então a única que podia ser alterada. No entanto, esse modelo apresentava algumas limitações. Em choperias muito altas, o curso total da torneira era reduzido em relação ao curso do atuador, devido ao tamanho do braço de alavanca. Além disso, a primeira versão não podia ser instalada em câmaras frias, uma vez que seria necessário perfurar a câmara para acomodar o motor.

Essas limitações foram identificadas durante os testes e eventos realizados, levando à necessidade de uma nova abordagem na segunda versão do sistema.

A segunda versão do sistema utilizou um motor de passo acoplado à torneira por meio de um acoplamento mecânico de engrenagens. Essa configuração permitiu que o motor de passo fosse conectado diretamente à torneira, eliminando os problemas causados por estruturas altas. O acoplamento mecânico garantiu um movimento preciso e confiável da torneira, proporcionando um melhor desempenho geral do sistema.

Devido à ausência de realimentação de posição no motor de passo, foram incorporadas chaves de fim de curso reguláveis na parte frontal e traseira da estrutura. Essas chaves foram posicionadas de forma a permitir a regulação nos limites máximo e mínimo da torneira. Durante o processo de calibração, o motor de passo move a torneira para trás até atingir a chave de fim de curso traseira e, em seguida, move a torneira para a frente até atingir a chave de fim de curso frontal. Com base nesse percurso, é possível calcular a quantidade de passos entre os limites, possibilitando o controle preciso da posição da torneira em qualquer ponto desejado.

Essa abordagem de calibração permite que o sistema se adapte a diferentes tipos e tamanhos de torneiras, garantindo um funcionamento adequado independentemente das características específicas de cada choperia. Além disso, a utilização de chaves de fim de curso reguláveis proporciona flexibilidade para ajustes e correções

necessárias ao longo do tempo de uso do sistema.

Essa segunda versão do sistema demonstrou uma maior eficiência e confiabilidade no serviço de chope, superando os problemas encontrados na primeira versão.

## 6.2 HARDWARE

O hardware do sistema é composto pelo dispositivo chamado "TAP", que consiste em uma placa equipada com tela, um microcontrolador e um leitor NFC/QR Code. Além disso, temos o motor e a estrutura mecânica responsáveis por abrir a torneira, juntamente com os sensores de presença que capturam o posicionamento do copo. Também são utilizados a solenoide, responsável pelo controle do fluxo de chope, e o sensor de vazão, que monitora a quantidade de líquido servido.

A implementação do projeto foi realizada utilizando o microcontrolador Toradex Colibri iMX6ULL, que possui 512MB de memória. Esse microcontrolador oferece a capacidade de processamento necessária para executar o sistema de forma eficiente e confiável.

## 6.3 SOFTWARE

O software do sistema foi desenvolvido utilizando diferentes tecnologias e linguagens de programação. O backend foi escrito em Go, uma linguagem conhecida por sua eficiência e escalabilidade. Já o frontend, responsável pela interface gráfica, foi desenvolvido em C++ utilizando a biblioteca SDL2. Ambos os componentes do software são executados em um sistema operacional Linux. Além disso, foram implementados programas auxiliares em C++ para a leitura e escrita de cartões cashless, assim como para a integração com parceiros externos.

A interface gráfica do sistema é apresentada por meio da tela do dispositivo TAP. Foram projetadas telas com o objetivo de fornecer informações claras e concisas aos clientes, facilitando a compreensão do processo de serviço do chope.

O ambiente de desenvolvimento utilizado para a criação do projeto foi o Visual Studio Code (VSCode), uma plataforma popular e robusta que proporciona uma experiência de desenvolvimento eficiente.

## 6.4 INTEGRAÇÕES E COMUNICAÇÃO

A comunicação entre os componentes do sistema é realizada por meio de diferentes mecanismos e protocolos. O sensor de fluxo gera interrupções que são lidas por um programa em Bash, que as interpreta e as envia para o backend. O dispositivo de leitura de cartões cashless, baseado no PN532, se comunica com a placa por meio de uma conexão serial UART. Da mesma forma, o leitor de QR Code também utiliza



uma conexão serial UART. Outros sensores mais simples geram sinais de GPIO, que são lidos diretamente pelo backend.

A integração do sistema de pagamento é realizada por meio de parcerias com fornecedores externos. Para o serviço de QR Code e cashless, utiliza-se a máquina SmartPOS do parceiro, onde ocorre a recarga do cartão ou a geração do QR Code. As informações são armazenadas no sistema do parceiro, permitindo a validação por meio de APIs. Assim todo o controle financeiro fica por parte do parceiro.

## 7 ANÁLISE DOS RESULTADOS

Durante o primeiro teste real do sistema na Oktoberfest de 2022 em Blumenau, onde foram operadas 8 torneiras ao longo dos 19 dias de festa, servindo mais de 10 mil litros de chope, foram observados alguns pontos importantes. O produto em si foi bem recebido pelos usuários e as pessoas gostaram da ideia de um sistema automatizado de serviço de chope. No entanto, em termos técnicos, enfrentamos alguns desafios.

Um dos problemas identificados foi relacionado à fixação do atuador linear na chopeira. Nas chopeiras utilizadas na Oktoberfest, que eram mais altas, havia uma distância considerável entre o atuador e a torneira. Isso resultava em um curso insuficiente do atuador de 5 cm para acionar completamente a torneira, devido ao tamanho da alavanca. Como resultado, a torneira nem sempre conseguia servir a espuma de forma adequada. Esse foi um ponto crítico que exigiu atenção e uma solução adequada para garantir o funcionamento correto do sistema em diferentes tipos de chopeiras.

Uma observação relevante durante o teste foi o desgaste prematuro do potenciômetro utilizado como leitor de posição do atuador. Com alguns dias de uso contínuo, o potenciômetro começava a apresentar um sinal extremamente ruidoso durante o movimento do motor. Isso comprometia a confiabilidade das leituras enquanto o motor estava em movimento, e em alguns casos, mesmo quando o motor estava parado, o potenciômetro fornecia leituras incorretas. Diante desse problema e considerando o custo envolvido na substituição frequente dos potenciômetros, ficou evidente que essa não seria uma opção viável para escalar o produto.

Além disso, constatou-se a necessidade de facilitar o acesso às funções da aplicação, como calibração e ajustes. No momento, essas atividades exigiam o acesso à TAP por meio do protocolo SSH com uma VPN, o que criava uma barreira técnica e limitava a praticidade e a agilidade na realização dessas tarefas.

Essa limitação no acesso às funcionalidades da aplicação dificultava o processo de calibração e ajustes necessários para garantir o correto funcionamento do sistema. Era necessário estabelecer uma conexão VPN e utilizar o protocolo SSH, o que demandava conhecimentos específicos e adicionava uma camada de complexidade ao processo.

Considerando a importância de uma operação eficiente e uma manutenção adequada do sistema, é essencial encontrar soluções que simplifiquem o acesso às funções da aplicação. Isso permitirá que os operadores possam realizar calibrações e ajustes de forma mais ágil e fácil, otimizando o tempo e garantindo uma experiência mais satisfatória no uso do sistema automatizado.

Outro ponto notado durante a análise dos resultados foi a constatação de que as telas do frontend continham uma quantidade excessiva de informações. Isso acabava

sobrecarregando os usuários e muitas vezes levava à falta de leitura e compreensão adequadas das informações apresentadas. Como resultado, a utilização do sistema apresentava algumas dificuldades.

Diante dessa observação, foi decidido que era necessário redesenhar as telas do frontend, adotando uma abordagem mais simples e intuitiva. O objetivo era proporcionar uma interface mais amigável, na qual as informações essenciais fossem apresentadas de forma clara e direta, facilitando a navegação e a compreensão por parte dos usuários.

Esses resultados e observações obtidos durante o teste real fornecem uma base sólida para a análise e aprimoramento do sistema. É necessário encontrar soluções eficientes para superar os desafios identificados, como a distância entre o atuador e a torneira e o desgaste prematuro do potenciômetro. Essas questões devem ser consideradas no desenvolvimento da versão mais robusta e escalável do sistema, a fim de garantir um desempenho confiável e uma experiência satisfatória para os usuários.

Durante o segundo evento, a Mondial de la Bière no Rio de Janeiro, foram servidos 5.800 litros de chope ao longo de 5 dias, por meio de 48 torneiras. Todos os problemas identificados na Oktoberfest foram abordados e aprimorados para este evento específico. As chopeiras utilizadas eram mais baixas, o que melhorou o funcionamento do atuador linear. Além disso, as telas do frontend foram redesenhadas e o acesso às funcionalidades da aplicação foi facilitado. Vale ressaltar que a duração do evento foi menor, o que tornou o problema de desgaste do potenciômetro menos relevante.

No entanto, o principal desafio enfrentado durante a Mondial de la Bière foi a dificuldade na montagem do sistema. Como estávamos lidando com a montagem de 48 torneiras, demandou-se um tempo significativo para a conclusão do processo. Em eventos com estruturas móveis, como era o caso, essa situação se tornou impraticável, uma vez que a estrutura foi entregue muito próximo ao início do evento, deixando pouco tempo disponível para configurar e garantir o pleno funcionamento de todo o sistema.

Outro ponto que chamou a atenção durante o evento foi a interferência dos clientes nas válvulas de vazão das torneiras. Com um maior número de torneiras e menor supervisão, era comum que os clientes tentassem ajustar a vazão, o que prejudicava a qualidade do serviço. É importante ressaltar que a regulagem adequada das válvulas de vazão é essencial para garantir a qualidade do serviço prestado.

Após a participação bem-sucedida no evento Mondial de la Bière, surgiu a oportunidade de instalar o sistema no Estádio do Benfica, em Portugal. Foram instaladas 8 torneiras que operam durante os intervalos dos jogos, em períodos de aproximadamente 15 a 20 minutos. Desde janeiro, o sistema já atendeu a 8 jogos, servindo uma

média de 100 litros de chope por intervalo.

A implementação do sistema no Estádio do Benfica representa uma aplicação em um ambiente diferente dos eventos anteriores. Os intervalos dos jogos proporcionam uma demanda mais intensa e concentrada em um curto período de tempo. Isso exige um desempenho eficiente e confiável do sistema para atender à demanda dos torcedores durante os intervalos.

Além disso, a utilização do sistema no Estádio do Benfica tem proporcionado benefícios significativos. O sistema permite uma operação mais ágil e eficiente durante os intervalos dos jogos, reduzindo o tempo de espera dos torcedores para serem servidos. Além disso, a automatização do serviço de chope proporciona um controle mais preciso e padronizado, garantindo a qualidade do chope servido aos torcedores.

Após a realização de diversos eventos e análises dos resultados obtidos, chegou-se à conclusão de que seria necessário remodelar o produto para solucionar principalmente os problemas relacionados à escalabilidade, custos, manutenção e montagem. Dessa forma, foi desenvolvida a segunda versão do sistema, que utilizava um motor de passo.

A nova versão do sistema foi testada na feira da Fispal, em São Paulo, com 3 torneiras operando ao longo de 4 dias e servindo um total de 600 litros de chope. Durante os primeiros dois dias da feira, foram observados alguns problemas relacionados ao uso de novos tipos de torneiras, que eram mais pesadas e geravam perda de referência. O motor de passo não conseguia acionar corretamente as chaves de fim de curso localizadas na parte traseira da estrutura mecânica.

Além disso, uma implementação do código que movia o motor em segundo plano resultava em situações em que um movimento sobrescrevia o outro, impedindo que o motor se movesse corretamente. No entanto, esses problemas foram solucionados ao integrar o movimento do motor no fluxo principal do código e alterar a lógica de calibração, eliminando a necessidade das chaves de fim de curso na parte traseira.

Após a resolução dessas questões nos últimos dois dias da feira, o sistema demonstrou um desempenho excelente. Os problemas iniciais foram superados e o sistema operou de forma eficiente e confiável, atendendo às demandas de serviço de chope com sucesso.

Essa experiência na feira da Fispal foi fundamental para validar as melhorias implementadas na segunda versão do sistema. Os resultados obtidos reforçaram a eficácia das soluções adotadas, demonstrando que as modificações realizadas foram bem-sucedidas em resolver os desafios encontrados anteriormente.

A utilização desse sistema na empresa abre diversas portas e oportunidades, permitindo a expansão do negócio para uma área com poucos concorrentes. Além disso, proporciona aos contratantes uma solução que reduz as filas nos serviços de chope, garantindo maior segurança e confiabilidade.

Os testes realizados seguiram uma abordagem incremental, abrangendo diferentes aspectos do sistema, desde a seleção do tipo de motor até a validação da estrutura mecânica. Após cada etapa, foram implementadas melhorias e ajustes com base nos resultados obtidos.

Durante a avaliação do sistema, foram considerados indicadores como o tempo de montagem, o tempo de serviço e a robustez do sistema. Esses indicadores eram essenciais para avaliar o desempenho do sistema de acordo com os requisitos funcionais estabelecidos no projeto.

A implementação do sistema trouxe benefícios significativos em termos de tempo e eficiência no serviço de chope. O sistema foi capaz de atender às demandas de forma mais rápida e eficiente em comparação com a solução anterior.

Considerando os requisitos funcionais e não funcionais especificados e os objetivos estabelecidos, pode-se afirmar que o sistema foi concluído com sucesso. Ele atendeu às expectativas, proporcionando benefícios significativos em termos de desempenho, eficiência e confiabilidade.

## 8 CONCLUSÃO

Em conclusão o aluno foi capaz de cumprir todos os requisitos funcionais propostos, resultando em um produto funcional que atendeu às expectativas estabelecidas.

No entanto, foram identificadas algumas limitações ao longo do processo. Uma delas está relacionada à dificuldade em lidar com torneiras mais pesadas, o que afeta o movimento adequado do sistema e pode comprometer o fluxo de espuma durante o serviço. Além disso, o sensor de fluxo pode apresentar dificuldades em detectar vazões muito baixas, o que impacta a precisão das medições.

Apesar das limitações, o sistema trouxe diversos benefícios. A agilidade e praticidade do serviço foram pontos positivos, permitindo uma experiência mais eficiente para os clientes. Além disso, o sistema despertou atenção e interesse por sua inovação no setor.

Em relação ao impacto na empresa, a utilização desse sistema oferece oportunidades de expansão de negócios em um mercado com poucos competidores. Além disso, ele proporciona uma solução que reduz filas, aumenta a segurança e a confiabilidade do serviço de chope.

Para avaliar os resultados, foram considerados indicadores como tempo de montagem, tempo de serviço e robustez do sistema, uma vez que esses eram requisitos fundamentais para o projeto.

Visando trabalhos futuros, sugere-se a substituição do motor de passo por um motor CC simples, incorporando chaves de fim de curso para garantir o funcionamento adequado do sistema. Essa alteração traria maior simplicidade e robustez ao sistema, solucionando algumas das limitações identificadas.

Além disso, seria interessante explorar soluções para lidar de forma mais eficiente com torneiras mais pesadas, melhorar a detecção de vazões baixas pelo sensor de fluxo e aprimorar a precisão das medições.

Em resumo, o projeto do sistema automatizado de serviço de chope alcançou resultados positivos e atendeu aos requisitos propostos. As limitações identificadas apontam para possibilidades de melhorias futuras, tornando o sistema ainda mais robusto e eficiente. A implementação desse sistema representa um avanço significativo na indústria de serviços de chope, oferecendo uma solução moderna e atraente para o mercado.

## REFERÊNCIAS

BTS7960 Datasheet. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://image.dfrobot.com/image/data/DRI0018/BTS7960.pdf>.

DATASHEET motor NEMA-17. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://techmakers.fbittstatic.net//media/Datasheet%5C%20Motor%5C%20de%5C%20Passo%5C%20Nema%5C%2017%5C%201.1Kgf.cm%5C%20P-N-AK17-1.1F6LN1.8.pdf?origem=MediaCenter/Datasheet%5C%20Motor%5C%20de%5C%20Passo%5C%20Nema%5C%2017%5C%201.1Kgf.cm%5C%20P-N-AK17-1.1F6LN1.8.pdf>.

ETT CO., LTD. **E18-D80NK IR Sensor Switch**. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://datasheetspdf.com/pdf/1311838/ETT/E18-D80NK-N/1>.

FILAMENTO Tritan: Conheça o Novo filamento. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://3dlab.com.br/filamento-tritan/>.

FIT080X Product Overview. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://www.mouser.com/pdfDocs/Productoverview-DFRobot-FIT080x.pdf>.

SPARKFUN ELECTRONICS. **L298 Dual Full-Bridge Driver**. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: [https://www.sparkfun.com/datasheets/Robotics/L298\\_H\\_Bridge.pdf](https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf).

TEJWANI, Ravi; SOLANKI, Chetan S. 360 sun tracking with automated cleaning system for solar PV modules. *In: IEEE. 2010 35th IEEE Photovoltaic Specialists Conference*. [S.l.: s.n.], 2010. P. 002895–002898.

TEXAS INSTRUMENTS. **L293 Quadruple Half-H Drivers**. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://www.ti.com/lit/ds/symlink/l293.pdf>.

TORADEX. **Colibri ARM SoM iMX6ULL Datasheet**. [S.l.: s.n.]. Acessado em: junho de 2023. Disponível em: <https://docs.toradex.com/104446-colibri-arm-som-imx6ull-datasheet.pdf>.

TURNER, AJ; RAMSAY, Keith; CLARK, Richard Edward; HOWE, D. **Development of high force electromechanical linear actuator for shift-by-wire automated manual transmissions**. [S.l.], 2006.

---

XAVIER, Raquel. **Torneira de chopp italiana ou belga: qual escolher?** [S.l.: s.n.], 2022. Acesso em: 30 jun. 2023. Disponível em:  
<https://www.mytapp.com.br/blog/torneira-de-chopp-italiana-ou-belga/>.



## APÊNDICE A – DEFINIÇÃO DOS STRUCTS

```
type Motor struct {
    channel      chan string
    target       chan int
    cupSensor    Sensor
    otherCupSensor Sensor
    lastState    bool
    inPosition   bool
    id           string
    adc          Converter
    pwm          string
    gpio         string
    state        string
    pos          string
    log          *logger.Logger
    info         *event
}
```

```
type Sensor struct {
    state  bool
    current bool
    gpio  string
}
```

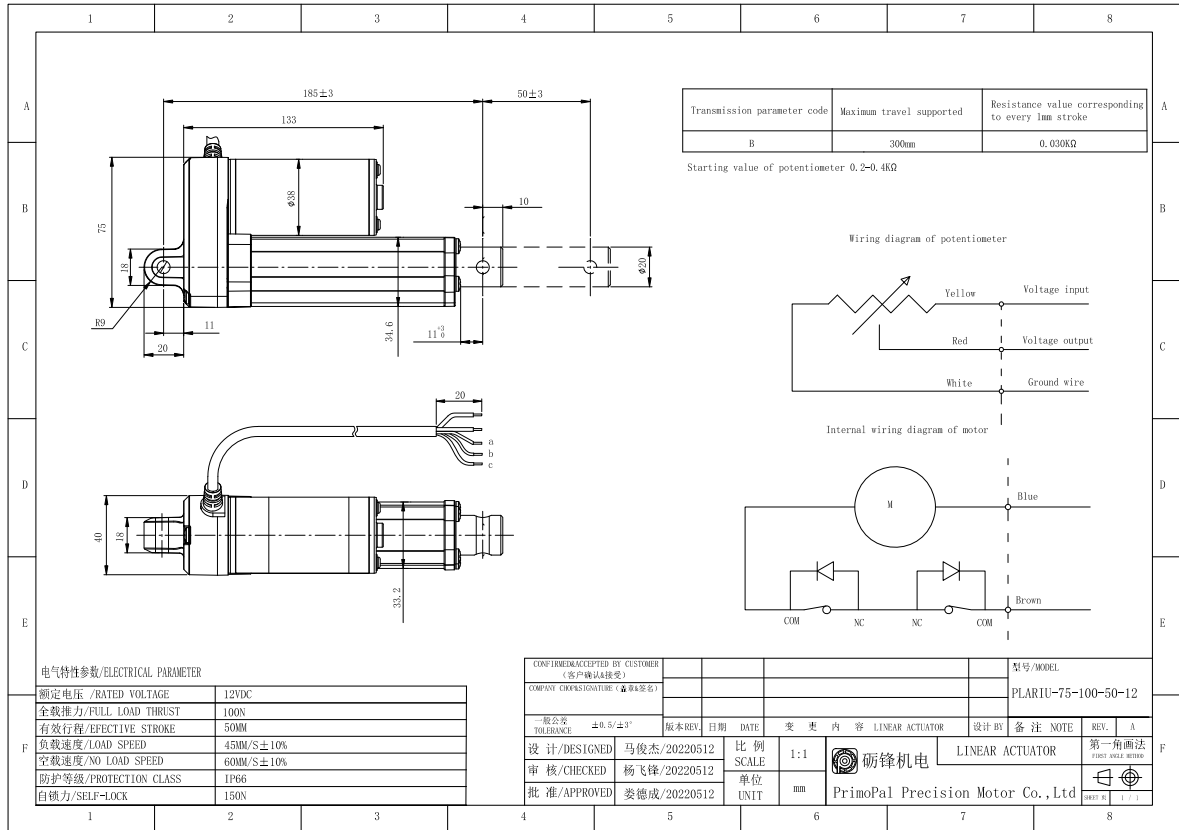
```
type Converter struct {
    id string
    max float32 // actually int
    min float32 // actually int
}
```

## APÊNDICE B – FUNÇÃO PARA RELAÇÃO DA ESPUMA

```
func (tap *Tap) getProportion(timeSince time.Duration) int {  
    if timeSince <= time.Duration(5)*time.Minute {  
        return 0  
    } else if timeSince >= time.Duration(30)*time.Minute {  
        return tap.event.foamVolume  
    } else {  
        timeDiff := timeSince.Minutes() - 5  
        return int(float64(tap.event.foamVolume) * (timeDiff / 25))  
    }  
}
```

# ANEXO A – DATASHEET DO ATUADOR LINEAR

Figura 23 – Datasheet do motor linear.



Fonte: Arquivo myTapp (2022).