



TITULADOR AUTOMÁTICO CONTROLADO POR ARDUINO® UNO

Lucas da Conceição Ferreira^{1*}
Luiz Fernando Almeida Silva^{1*}
Paulo Sérgio de Souza Santos^{1*}
Rafael Carvalho Souza^{1*}
Vinícius Bellini Batistelli^{1*}

Orientador: Prof. Dr. Fábio Rizzo de Aguiar
Coorientadora: Profa. Esp. Thais Taciano dos Santos

RESUMO

Atualmente um dos mais utilizados métodos da Química Analítica nas indústrias é a titulação ou volumetria, que se trata de uma técnica para descobrir a concentração de uma determinada substância em uma amostra. Porém, a titulação clássica, que consiste em identificar a alteração de cor da solução a partir do uso de um indicador, acaba não sendo tão acessível para alguns químicos com limitações físicas ou visuais. Por essa razão, este trabalho descreve o desenvolvimento de um equipamento de titulação automatizada de baixo custo utilizando uma bomba peristáltica, controlado por um Arduino® Uno que comanda e analisa o processo da titulação por meio de um sensor de cor. Foram realizados diversos testes para verificar a vazão da bomba peristáltica e posteriormente calibrar o sensor de cor para a titulação de neutralização utilizando o indicador fenolftaleína. Os resultados foram comparados com a titulação clássica, não apresentando diferenças estatisticamente significativas para o processo de titulação, com aplicação do teste t pareado a 95% de confiança. O método proposto obteve precisão confirmada pelos valores de desvio padrão relativos inferiores a 1% para padronização do titulante e 1,3% para titulação e pelos valores de erro relativo inferior a 3,4% para padronização do titulante e -0,7% para titulação.

Palavras-chave: Titulação. Automatização. Arduino® Uno.

ABSTRACT

One of the most used methods of Analytical Chemistry in industries today is titration or volumetry, which is a technique to discover the concentration of a given substance in a sample. However, the classic titration, which consists of identifying the change in color of the solution using an indicator, ends up not being so accessible for some chemists with physical or visual limitations. For this reason, this work describes the development of a low-cost automated titration device using a peristaltic pump, controlled by an Arduino® Uno that commands and analyzes the titration process through a color sensor. Several tests were carried out to verify the flow of the peristaltic pump and later calibrate the color sensor for the neutralization titration using the phenolphthalein indicator. The results were compared with the classic titration, showing no statistically significant differences for the titration process, with the application of the paired t test at 95% confidence. The proposed method achieved accuracy confirmed by relative standard deviation values lower than 1% for titrant standardization and 1.3% for titration, also for relative error values of less than 3.4% for standardization of the titrant and -0.7% for titration.

Keywords: Titration. Automation. Arduino® Uno.

¹ Curso Técnico em Química – ETEC Irmã Agostina
Av. Feliciano Correa s/n – Jardim Satélite - CEP 04815-240 - São Paulo – Brasil
* tituladorautomaticoetecia@gmail.com

1. INTRODUÇÃO

De maneira geral a titulação baseia-se na reação estequiométrica completa entre as espécies envolvidas, e em técnicas analíticas clássicas que são baseadas na identificação da alteração de cor do titulado, seja pela ação de um indicador, como fenolftaleína, ou excesso de titulante que possua coloração característica, como permanganato de potássio. O processo de titulação pode ser automatizado por meio de instrumentos e análises a partir de ferramentas de programação, evitando assim erros de análise, e proporcionando maior eficiência, repetibilidade, confiabilidade e exatidão aos dados obtidos. Existe uma diversa gama de modelos comerciais no mercado, automatizados por diferentes plataformas com diferentes métodos de detecção, sendo os mais comuns deles a potenciometria e a condutimetria (BATISTA, 2022).

O Arduino® Uno é uma espécie de computador, ou seja, é uma máquina que processa instruções e através da sua programação pode desempenhar diferentes tarefas para desenvolvimento de projetos inclusive na química, por exemplo, para automatizar processos de titulações.

O ponto de viragem (alteração da cor do indicador), pode ser um fator de difícil identificação para quem possui daltonismo, além dos vários erros oriundos da titulação manual, o que pode dificultar ou até mesmo inviabilizar o processo clássico de titulação por parte da população daltônica. Além disso, pessoas com dificuldade motoras podem ter problemas para realizar a titulação pelo método clássico pois há necessidade de controle da bureta e ao mesmo tempo realizar a agitação do Erlenmeyer. Além de todas as dificuldades, encontramos dentro de uma titulação manual, erros aleatórios do analista. (CARPENTER, 2021).

Perante essas dificuldades apontadas, o trabalho pode contribuir para as próximas turmas da ETEC Irmã Agostina, com um titulador automático de baixo custo, presumindo pessoas que apresentam certo grau de dificuldade em uma titulação manual. Assim o objetivo deste trabalho é construir um titulador automático controlado por Arduino® Uno.

2. METODOLOGIA

Para a realização do trabalho foram utilizados os seguintes reagentes, equipamentos e procedimentos.

2.1. REAGENTES

Nos testes foram utilizados solução de ácido clorídrico P.A., hidróxido de sódio P.A., biftalato de

potássio P.A, água destilada e o indicador fenolftaleína.

2.2. MATERIAIS E EQUIPAMENTOS

Os materiais e equipamentos utilizados na realização deste trabalho estão listados a seguir, seguidos por uma breve descrição das respectivas funcionalidades.

- Um Arduino® Uno: componente responsável por processar os dados originados do sensor de cor, teclado, *encoder* e, a partir da sua programação, transmitir sinais para a bomba peristáltica, a tela e o *buzzer*.

- Uma bomba peristáltica: possui a função de transferir o líquido do titulante para a solução titulada (BOMBA PERISTÁLTICA, 2015), ela é ligada e desligada indiretamente através do transistor TIP122 a partir do comando do Arduino® Uno. Em suas extremidades são conectadas as mangueiras de silicone e na ponta da mangueira onde fica a solução titulada foi colocada a ponteira de pipeta para possibilitar um gotejamento mais preciso.

- Um sensor de cor TCS3200: componente que capta a intensidade das três cores primárias (vermelho, verde e azul) e a intensidade da cor branca, ou seja, das três cores combinadas. Estes quatro sinais são transferidos ao Arduino® Uno de maneira independente e são convertidos em valores inversamente proporcionais a intensidade luminosa de cada cor (OLIVEIRA, [2020?]). Como o indicador de pH utilizado no estudo foi a fenolftaleína, que é incolor em meio ácido e possui cor rosa intensa em meio alcalino, o Arduino® Uno foi ajustado para interromper a injeção de titulante assim que fosse detectada uma variação de cor da solução titulada para um rosa claro.

- Um teclado matricial: com a função de permitir a interação entre o usuário e o titulador possibilitando a inserção de dados e determinação do momento de início e pausa do processo.

- Uma tela OLED 0,96": que permite ao usuário visualizar as opções e obter dados do processo de titulação.

- Um *buzzer* que permite um retorno sonoro ao usuário ao pressionar uma tecla ou quando ocorre o fim do processo de titulação.

- Um *encoder*: componente responsável por transformar o movimento mecânico do eixo do motor da bomba peristáltica em pulsos elétricos a serem interpretados pelo Arduino® Uno (O QUE É ENCODER, c2022).

- Uma ventoinha: empregada com intuito de substituir o agitador magnético. No seu centro foram colados dois ímãs com polos opostos de forma a interagir com a barra magnética presente

no interior do béquer que contém a solução titulada. Uma vez que o titulador é ligado, a ventoinha é acionada.

- Uma caneca plástica branca e uma tampa que propiciam um ambiente fechado para evitar que alterações na iluminação ou coloração do ambiente externo interfiram na análise da solução titulada, ao mesmo tempo, suas colorações brancas refletem a coloração da solução e aumentam a sensibilidade do sensor.

- Os demais itens incluem: duas mangueiras de silicone com diâmetro interno de 2 mm, uma ponteira de pipeta 1000 μL , uma caixa de acrílico 20x15x5 cm, uma barra magnética cilíndrica 5x15 mm, dois ímãs de neodímio, uma *Protoboard* 170 pontos, um transistor TIP122, um diodo 1N4148, um conector *Jack DC P4*, uma chave gangorra com dois terminais, um resistor 10 k Ω , dois resistores 220 Ω , dois capacitores 10 nF, fios para conexão e uma fonte de alimentação 9 V x 1 A, um copo quadrado de 50 mL.

Para os testes e calibrações foram utilizados uma balança analítica, duas buretas de 25 mL, um balão volumétrico de 250 mL, um balão volumétrico de 25 mL, um Erlenmeyer de 250 mL, dois béqueres de 100 mL, suporte universal, quatro pipetas Pasteur, um recipiente plástico com tampa de 1 L e um recipiente de vidro âmbar com tampa de 1 L.

2.3. PROCEDIMENTOS

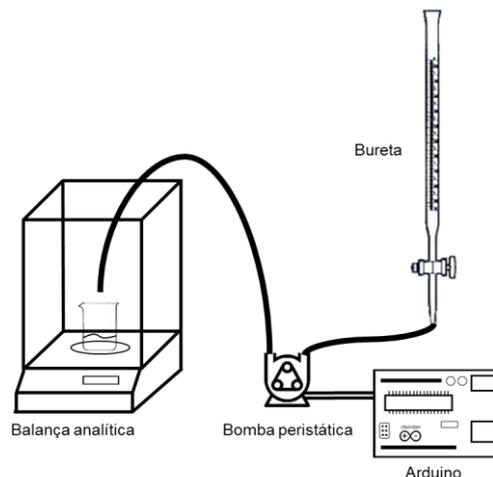
Para que fosse possível o gotejamento do titulante em quantidades conhecidas foram necessários processos prévios como a calibração das buretas, a calibração e teste de vazão da bomba peristáltica, a calibração do sensor de cor, a programação do Arduino® Uno e a preparação das soluções utilizadas (hidróxido de sódio 0,2 mol L⁻¹ e ácido clorídrico 0,1 mol L⁻¹) e padronização do titulante (hidróxido de sódio).

Para testar a vazão da bomba foram colocados 25 mL de água em uma bureta e a água foi bombeada e pesada gotejando-a para outro béquer previamente tarado numa balança analítica. A partir desta prática, foram feitos três ensaios e foi possível obter a vazão da bomba em intervalos de 2 segundos. Também foram feitos três ensaios para obter a vazão da bomba em função do número de pulsos medidos pelo sensor acoplado ao eixo do motor da bomba. A Fórmula 1 demonstra a correlação linear dos dados do volume em função dos pulsos, onde foi possível obter os dados de “V(titulante)”, que é o volume em mL e “p”, que é a quantidade de pulsos lidos através do sensor no eixo do motor, “a” é coeficiente angular e “b” é o coeficiente linear.

$$V(\text{titulante}) = a \cdot p + b \quad (\text{Fórmula 1})$$

O esquema do sistema para calibração da bomba encontra-se demonstrado na Figura 1.

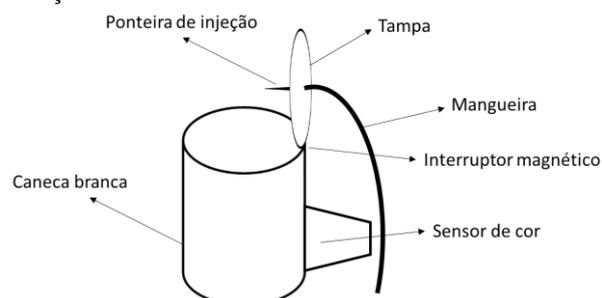
Figura 1: Esquema do sistema de calibração da bomba



Uma vez encerrados os testes de vazão para calibração da bomba, foram necessários testes para determinação do valor percentual da variação da coloração que seria utilizado para indicar o término da titulação.

Para evitar que interferências advindas do ambiente afetassem a detecção do sensor, foi necessário instalar o sensor e o recipiente para a solução titulada em um recipiente fechado de cor branca – para proporcionar uma melhor reflexão das cores para o sensor, conforme esquema da Figura 2.

Figura 2: Esquema do compartimento do béquer para titulação

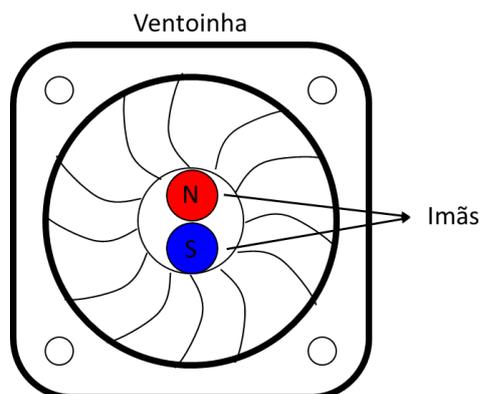


Implementou-se um interruptor magnético próximo a dobradiça da tampa para que, caso ela seja aberta durante a titulação, o processo seja pausado, evitando que a luminosidade do ambiente interfira no processo e que a solução titulante caia fora do recipiente com a solução titulada.

Para garantir a homogeneização da solução durante o processo de titulação e garantir uma cor

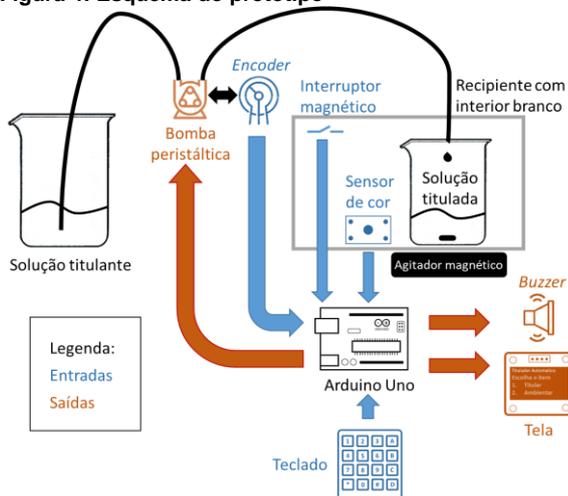
uniforme durante a leitura do sensor, foi implementado um agitador magnético composto por dois ímãs colados lado a lado, com polos opostos, em uma ventoinha conforme Figura 3. Este arranjo foi posicionado abaixo do recipiente onde encontrava-se a solução titulada onde foi colocada a barra magnética.

Figura 3: Esquema do agitador magnético



O protótipo foi montado conforme esquema demonstrado na Figura 4.

Figura 4: Esquema do protótipo



Por fim, foi necessária a criação de um código para que houvesse o controle da bomba peristáltica em função dos valores medidos pelo sensor de cor. Assim, usando-se o IDE oficial do Arduino® Uno, pode-se programar adequadamente a placa através das funções da linguagem C++. As variáveis, funções e estruturas utilizadas encontram-se nas Tabelas 1, 2 e 3, respectivamente.

Tabela 1: Códigos para programação em C++ – Variáveis

Nome	Função
<i>const</i>	Constante. Modifica o comportamento da variável, tornando uma variável "somente leitura", ou seja, o valor não pode ser alterado.
<i>byte</i>	Um <i>byte</i> armazena um número sem sinal de 8 <i>bits</i> , de 0 a 255.
<i>char</i>	Um tipo de dado usado para armazenar um valor de caractere.
<i>unsigned long</i>	Variáveis longas sem sinal são variáveis de tamanho estendido para armazenamento de números e armazenam 32 bits (4 bytes). Ao contrário dos <i>longs</i> padrões, os <i>longs</i> não armazenam números negativos, variando de 0 a 4.294.967.295.
<i>unsigned int</i>	<i>unsigned ints</i> (<i>unsigned integers</i>) são os mesmos que <i>ints</i> , pois armazenam um valor de 2 bytes. Em vez de armazenar números negativos, eles armazenam apenas valores positivos, produzindo um intervalo útil de 0 a 65.535.
<i>int</i>	Valores numéricos (inteiros)
<i>float</i>	Valores numéricos (decimais)
<i>String</i>	Valores de textos (caracteres)
<i>HIGH</i>	Nível lógico de uma entrada/saída como alto, ou seja, próximo a tensão de alimentação
<i>LOW</i>	Nível lógico de uma entrada/saída como baixo, ou seja, próximo a tensão 0V
<i>INPUT</i>	Define pino como entrada
<i>OUTPUT</i>	Define pino como saída

Fonte: Arduino.cc. (c2022)

Tabela 2: Códigos para programação em C++ – Funções

Nome	Função
<i>analogRead()</i>	Lê o valor de um pino analógico especificado. O Arduino® Uno permite uma resolução entre leituras de: 5 volts / 1024 unidades.

<code>analogWrite()</code>	Aciona uma onda PWM em um pino. Pode ser usada para variar o brilho de um LED ou acionar um motor a diversas velocidades.
<code>digitalRead()</code>	Lê o valor de um pino digital especificado, HIGH ou LOW.
<code>digitalWrite()</code>	Escreve um valor HIGH ou LOW em um pino digital.
<code>delay()</code>	Pausa o programa por uma quantidade especificada de tempo (em milissegundos). Cada segundo equivale a 1000 milissegundos.

Fonte: Arduino.cc. (c2022)

Tabela 3: Códigos para programação em C++ – Estruturas

Nome	Função
<code>void setup(){}</code>	Executada apenas uma vez quando o Arduino® Uno é inicializado.
<code>void loop(){}</code>	Fica em ciclo constante após a execução do "void setup".
<code>if(){}</code>	Verifica se a expressão entre parênteses é verdadeira e executa o que está dentro das chaves.
<code>else{}</code>	Utilizado após o "if", é chamado quando o "if" que o antecede tem retorno falso e executa o que está dentro das chaves.
<code>!=</code>	Diferente de
<code><</code>	Menor que
<code><=</code>	Menor que ou igual a
<code>=</code>	Igual a
<code>></code>	Maior que
<code>>=</code>	Maior que ou igual a
<code>while(){}</code>	Irá se repetir continuamente, e infinitamente, até a expressão dentro dos parênteses (), se torne falsa.
<code>goto</code>	Transfere o fluxo do programa para um ponto rotulado no programa
<code>*</code>	Multiplicação
<code>+</code>	Adição
<code>-</code>	Subtração
<code>/</code>	Divisão
<code>=</code>	Operador de atribuição
<code>!</code>	NÃO lógico
<code>&&</code>	E lógico

<code>//</code>	OU lógico
<code>#define</code> <i>(define)</i>	Permite ao programador dar um nome a um valor constante antes de o programa ser compilado.
<code>#include</code> <i>(include)</i>	É usada para incluir bibliotecas externas ao programa
<code>/* */</code> <i>(comentário em bloco)</i> ou <code>//</code> <i>(comentário)</i>	Comentários são textos no programa que são usadas para informar você e a outros a forma como o programa funciona. Eles são ignorados pelo compilador, e não fazem parte do arquivo gravado no chip, então não ocupam nenhum espaço na memória flash do microcontrolador.
<code>;</code>	Usado para encerrar um comando.
<code>{}</code>	Delimitam início e fim de um comando

Fonte: Arduino.cc. (c2022)

A partir do volume de solução titulante utilizada é possível obter a concentração da solução titulada caso os dados da concentração da solução titulante, o volume de solução titulada e a estequiometria da reação tenham sido informados pelo usuário. Supondo-se uma reação genérica expressa na Equação 1 em que "A" é a substância titulada e "B" é o titulante:

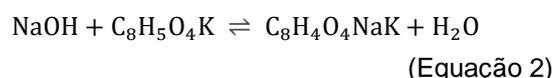


Desse modo, a relação em quantidade de matéria para que a reação se complete pode ser determinada e transformada para obter a concentração do titulado conforme Fórmula 2. Em que "c_A" é a concentração em mol L⁻¹ do titulado, "c_B" é a concentração em mol L⁻¹ do titulante, "a" é o coeficiente estequiométrico do titulado, "b" é o coeficiente estequiométrico do titulante, "V_A" é o volume em mL do titulado e "V_B" é o volume em mL do titulante.

$$c_A = \frac{a \cdot c_B \cdot V_B}{b \cdot V_A} \quad (\text{Fórmula 2})$$

A padronização do titulante foi realizada da forma clássica (com o uso da bureta e indicador) e com o titulador automático.

Para padronização, utilizou-se o biftalato de potássio cuja reação pode ser representada como descrito na Equação 2.



Pela Equação 2 observa-se que a cada 1 mol de NaOH é necessário 1 mol de biftalato de potássio, cuja massa molecular é 204,22 g mol⁻¹.

Foram pesados 0,1 g de biftalato de potássio, em uma balança analítica, e dissolvidos em aproximadamente 60 mL de água destilada, adicionando fenolftaleína como indicador. Em seguida, as amostras foram tituladas com hidróxido de sódio em triplicata.

Utilizou-se a Fórmula 3 para determinar a concentração padronizada da solução de NaOH, em que “c(NaOH)” é a concentração em mol L⁻¹ e “V(NaOH)” é o volume em L de NaOH e “m(Bif)” é a massa em g e “MM(Bif)” é a massa molecular do biftalato de potássio.

$$c(\text{NaOH}) = \frac{m(\text{Bif})}{MM(\text{Bif}) \cdot V(\text{NaOH})} \quad (\text{Fórmula 3})$$

A determinação da concentração de ácido clorídrico foi feita de duas formas paralelamente: uma de forma clássica e outra usando o titulador automático.

Para determinar a concentração do ácido clorídrico pelo método clássico, a bureta foi completada com 25 mL da solução de hidróxido de sódio padronizada, em seguida completou-se o balão volumétrico de 10 mL com a solução de ácido clorídrico, transferiu-se para o Erlenmeyer e adicionou-se 3 gotas de indicador fenolftaleína. Realizou-se o processo de titulação até obtenção da coloração rosa claro e anotou-se o volume de titulante utilizado. O cálculo para obter a concentração do ácido clorídrico foi realizado conforme Fórmula 4, em que “c(HCl)” é a concentração em mol L⁻¹ e “V(HCl)” é o volume em L de HCl e “c(NaOH)” é a concentração em mol L⁻¹ e “V(NaOH)” é o volume em L de NaOH.

$$c(\text{HCl}) = \frac{c(\text{NaOH}) \cdot V(\text{NaOH})}{V(\text{HCl})} \quad (\text{Fórmula 4})$$

O processo também foi realizado em triplicata e considerada a média dos valores obtidos.

Para determinar a concentração do ácido clorídrico através do titulador automático, foi colocado em um béquer aproximadamente 30 mL da solução de hidróxido de sódio padronizada e ambientou-se as mangueiras do titulador com esta solução. Em seguida completou-se o balão volumétrico de 10 mL com a solução de ácido clorídrico, transferiu-se para o béquer de 100 mL, completou-se com água até 60 mL (o suficiente para ultrapassar o sensor de cor), adicionaram-se 3 gotas de indicador fenolftaleína e a barra magnética. Este béquer foi levado para dentro do recipiente do titulador, foram inseridos os dados no

titulador e iniciou-se o processo de titulação. Ao término do processo foram anotados os dados de volume consumido e concentração.

O processo foi realizado em triplicata e considerada a média dos valores obtidos.

2.4. TRATAMENTO DE DADOS

A padronização e titulação foram feitas em triplicata nos métodos clássico e automático. A partir dos dados de concentração obtidos foram calculadas as médias, desvios padrão, desvios padrão relativos (DPR), teste t e teste Q com nível de confiança de 95%, e cálculos dos erros relativos entre as técnicas clássica e automática.

3. RESULTADOS E DISCUSSÕES

3.1. CALIBRAÇÃO DAS BURETAS

As buretas utilizadas nos processos de titulação manual e automática foram calibradas em triplicata pesando-se água destilada, ou seja, os volumes foram convertidos com base nas massas de água à 19 °C em intervalos de 5 mL, para que fosse possível calcular o volume real a cada intervalo dentro da faixa de 25 mL de cada bureta. Os resultados médios para cada intervalo podem ser verificados nas Tabelas 4 e 5.

Tabela 4: Dados dos ensaios de calibração da bureta utilizada para titulações manuais

Intervalo de volume (mL)	Volume real (mL)
0-5	4,9941
5-10	5,0210
10-15	4,9500
15-20	5,0756
20-25	4,9622

Tabela 5: Dados dos ensaios de calibração da bureta utilizada para titulações automáticas

Intervalo de volume (mL)	Volume real (mL)
0-5	5,0077
5-10	4,9889
10-15	4,9467
15-20	4,9697
20-25	4,9072

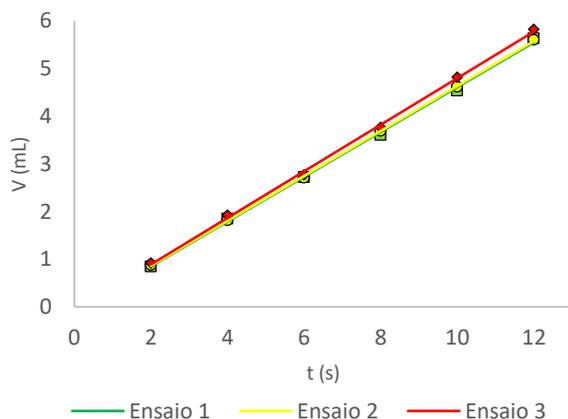
A calibração das buretas indicou os volumes reais para cada intervalo de volume do instrumento, que foram ligeiramente diferentes de 5 mL como indicado para cada intervalo. Esses valores foram

utilizados para converter os valores teóricos lidos nas buretas durante os ensaios de titulação.

3.2. CALIBRAÇÃO DA BOMBA

Para determinação do volume bombeado em função do tempo, foi montado o sistema demonstrado na Figura 1 e os dados obtidos foram expressos em forma de correlação linear, indicados na Figura 5.

Figura 5: Correlação linear do volume bombeado em função do tempo



Os dados de equação da reta e de coeficientes de correlação podem ser verificados na Tabela 6.

Tabela 6: Dados da equação da reta e coeficientes de correlação

Ensaio	R ²	Equação da reta
1	0,9987	$V = 0,4699 \cdot t + 0,0905$
2	0,9998	$V = 0,4709 \cdot t + 0,0784$
3	0,9991	$V = 0,4885 \cdot t + 0,0847$
Média	0,9992	$V = 0,4764 \cdot t + 0,0845$

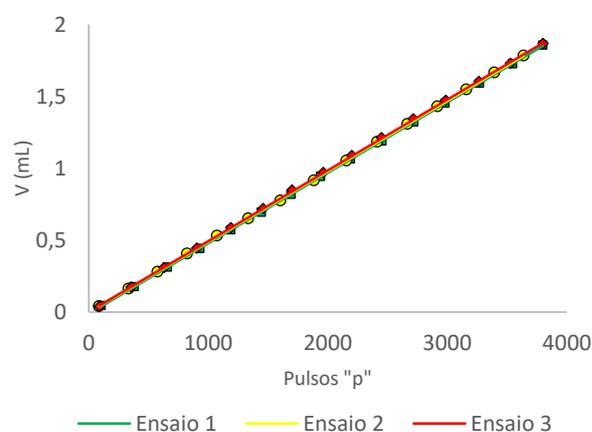
Os dados mostraram uma boa linearidade verificada pelo valor de R² médio próximo a 1, porém, após os ensaios, verificou-se que variações na tensão de alimentação da bomba provenientes da fonte ou mesmo interferências mecânicas poderiam afetar a velocidade do motor e, conseqüentemente, o volume bombeado. De forma a tornar o sistema mais preciso optou-se por utilizar um sistema retroalimentado incluindo um sensor de rotação denominado *encoder*. Este sensor foi acoplado ao eixo do motor da bomba e fornece doze pulsos a cada volta completa do eixo. A partir desses pulsos, o Arduino® Uno pôde ser

programado para determinar o volume bombeado a partir da quantidade de pulsos provenientes do motor.

Observou-se que o diâmetro de saída da mangueira da bomba afetava o tamanho da gota despejada. Para um sistema de titulação, onde o tamanho da gota é crucial para obter o ponto de viragem ideal, fez-se necessário reduzir o diâmetro do orifício de saída colocando uma ponteira de pipeta na extremidade de saída da mangueira.

Após as modificações, foi realizado o ensaio de calibração da bomba para o volume em função do número de pulsos “p” e foram obtidos os dados conforme gráfico da Figura 6.

Figura 6: Correlação linear do volume bombeado em função dos pulsos



Os dados de equação da reta e de coeficientes de correlação podem ser verificados na Tabela 7.

Tabela 7: Dados da equação da reta e coeficientes de correlação

Ensaio	R ²	Equação da reta
1	1,0000	$V = 490,11 \cdot 10^{-6} \cdot p - 11,50 \cdot 10^{-3}$
2	1,0000	$V = 490,07 \cdot 10^{-6} \cdot p - 2,00 \cdot 10^{-3}$
3	0,9999	$V = 492,09 \cdot 10^{-6} \cdot p + 1,07 \cdot 10^{-3}$
Média	1,0000	$V = 490,76 \cdot 10^{-6} \cdot p - 4,14 \cdot 10^{-3}$

Com base na equação da reta média o Arduino® Uno foi programado para calcular o volume bombeado através da Fórmula 5, em que “V(titulante)” é o volume em mL e “p” é a quantidade de pulsos lidos através do sensor no eixo do motor.

$$V(\text{titulante}) = 0,00049076 \cdot p - 0,00414$$

(Fórmula 5)

A excelente linearidade verificada pelo valor de R^2 médio equivalente a 1, junto a repetibilidade, observada pela proximidade dos valores dos coeficientes angulares e baixos valores dos coeficientes lineares, indicaram que a utilização do sensor de pulsos foi melhor em relação ao uso do tempo para realização dos cálculos de volume.

3.3. CALIBRAÇÃO DO SENSOR DE COR

Durante os ensaios verificou-se que, para o indicador fenolftaleína, o momento ideal para encerrar o processo de titulação ocorreu quando a maior variação de cor atingisse 13% em relação a cor detectada no início da titulação.

3.4. MONTAGEM DO PROTÓTIPO

Para que o usuário pudesse inserir dados, comandar as funções do titulador e obter informações implementou-se um teclado alfanumérico e uma tela OLED.

Os componentes foram montados numa caixa de acrílico transparente. As imagens do protótipo montado e do compartimento do bquer para o titulado podem ser verificadas nas Figuras 7, 8 e 9, respectivamente.

Figura 7: Foto do protótipo montado



Figura 8: Foto do protótipo montado



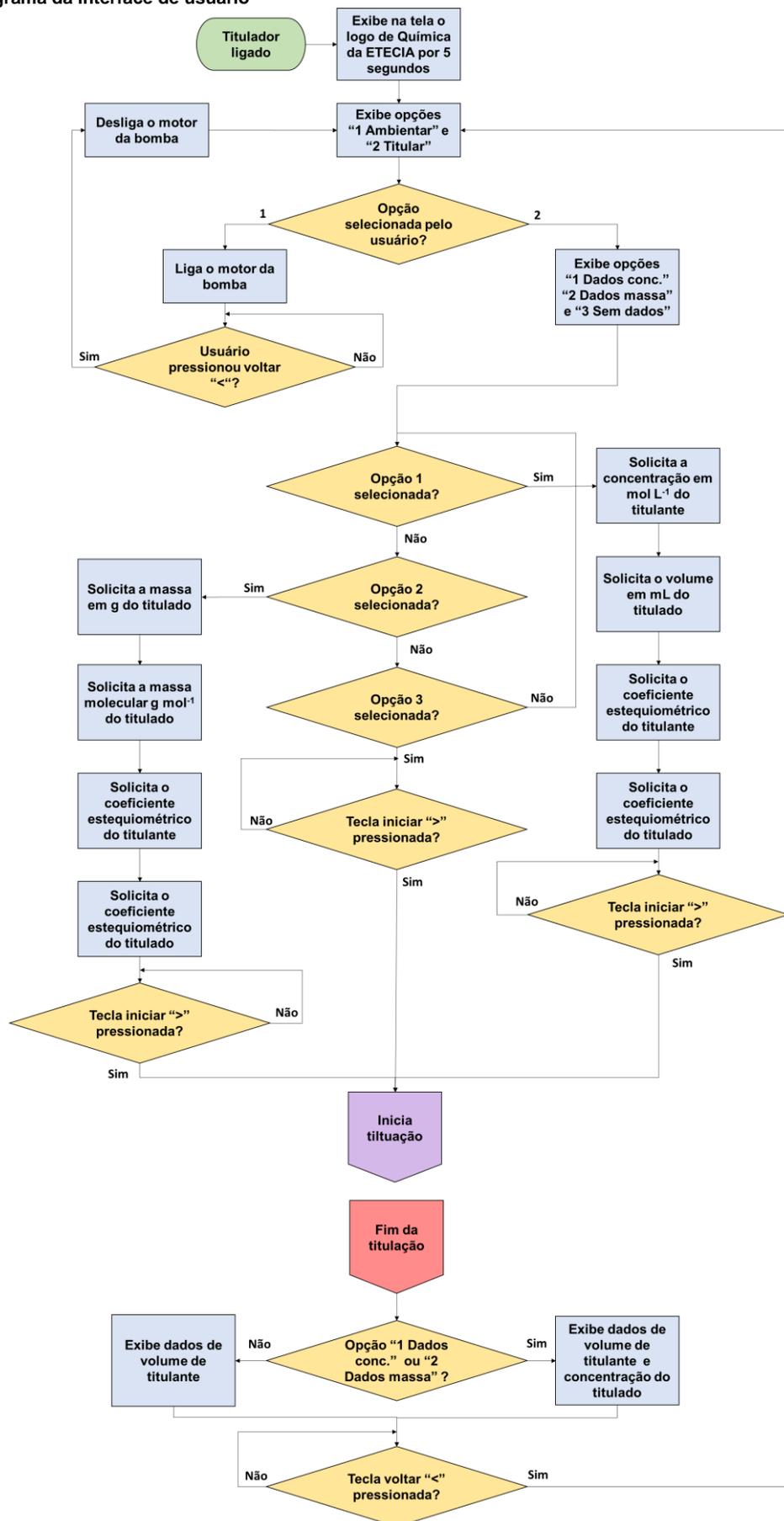
Figura 9: Compartimento do bquer para o titulado



3.5. PROGRAMAÇÃO DO ARDUINO® UNO

Para que o titulador faça a interface com o usuário foi criado um programa que interpreta os dados inseridos no teclado e exibe na tela as informações. O fluxograma que contém a lógica de programação para interface de usuário encontra-se na Figura 10.

Figura 10: Fluxograma da interface de usuário



Ao ligar o titulador a tela solicita ao usuário que selecione, a partir dos números do teclado, a função desejada, conforme Figura 11.

Figura 11: Tela inicial do programa



Ao clicar no número “1” o usuário entra no modo de ambientação das mangueiras e ao clicar no número “2” o usuário entra no modo de titulação. A ambientação faz com que o motor da bomba seja acionado continuamente para que preencha ou esvazie as mangueiras com a solução titulante. É imprescindível que, antes de iniciar o processo de titulação, as mangueiras estejam completamente cheias com a solução titulante.

Ao escolher a opção “Titular” o usuário pode escolher entre as opções mostradas na Figura 12.

Figura 12: Tela de opções para a titulação



A opção “1”, destinada para os ensaios de volumetria de neutralização, possibilita ao usuário inserir os dados da concentração em mol L⁻¹ do titulante, o volume em mL do titulado, coeficiente estequiométrico do titulante e coeficiente estequiométrico do titulado. A partir desses dados, e conhecendo o volume utilizado na titulação, o Arduino® Uno realiza o cálculo conforme Fórmula 2 para determinar a concentração em mol L⁻¹ do titulado e exibe-a ao final do processo de titulação junto com volume de titulante utilizado em mL.

A opção “2”, destinada para os ensaios de padronização, possibilita ao usuário inserir os dados da massa em g do titulante (“m_B”), massa molecular em g mol⁻¹ do titulante (“MM_B”), coeficiente estequiométrico do titulante (“x_B”) e coeficiente estequiométrico do titulado (“x_A”). A partir desses dados, e conhecendo o volume utilizado na titulação (“V_A”), o Arduino® Uno realiza

o cálculo conforme Fórmula 6 para determinar a concentração em mol L⁻¹ do titulado (c_A) e exibe-o ao final do processo de titulação junto com volume de titulado (V_A) utilizado em mL.

$$c_A = \frac{x_A \cdot m_B \cdot 1000}{x_B \cdot MM_B \cdot V_A} \quad (\text{Fórmula 6})$$

A opção “3” possibilita ao usuário prosseguir com a titulação sem inserir nenhum dado, neste caso é exibido apenas o volume utilizado na titulação em mL ao final.

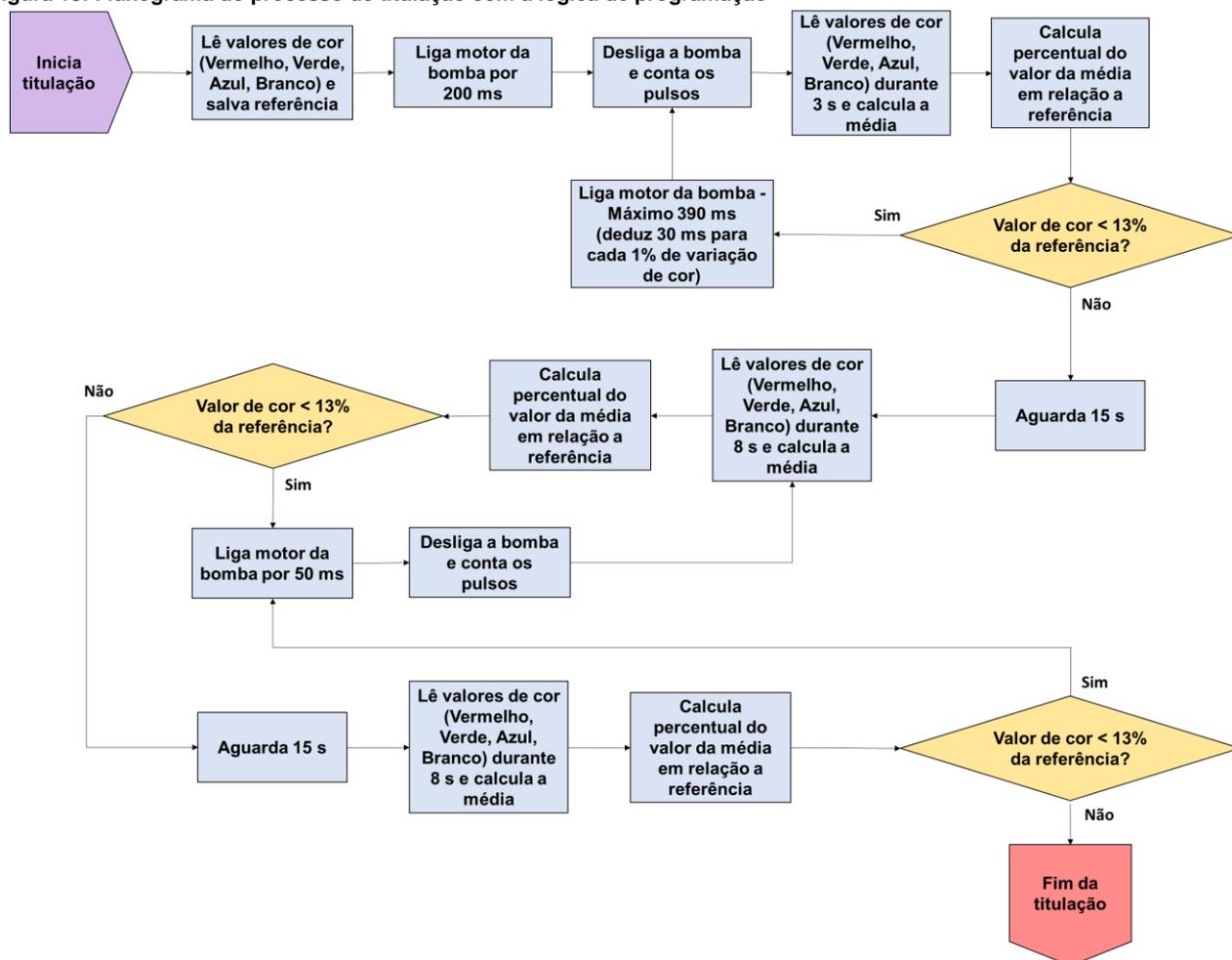
O fluxograma que contém a lógica de programação para o processo de titulação encontra-se na Figura 13.

Na primeira etapa, o motor é ligado brevemente para injetar um volume mínimo de solução e, em seguida, o sensor de cor faz a leitura durante 3 segundos, pois se o ponto de viragem estiver próximo logo no início, uma injeção muito alta poderia ultrapassar este ponto.

Na segunda etapa, entra-se num ciclo onde o acionamento do motor é feito baseado na variação de cor detectada até um limite de 390 milissegundos, descontando-se 30 milissegundos a cada 1% de variação. Quando a variação de cor ultrapassa os 13% entra-se na terceira etapa pois o ponto de viragem foi atingido ou está próximo. Nesta etapa aguarda-se 15 segundos e lê-se novamente o sensor, caso a variação de cor esteja inferior a 13% uma nova injeção é feita ligando-se a bomba por apenas 50 milissegundos a cada leitura de cor, dessa vez realizada durante 8 segundos, até que a variação seja igual ou superior a 13%. Na quarta e última etapa, aguarda-se 15 segundos para verificar se a variação de cor se mantém igual ou superior a 13%, em caso positivo, o processo é encerrado, em caso negativo, retorna-se à terceira etapa.

Para calcular o volume da solução utilizada no processo e exibir os dados na tela, o Arduino® Uno armazena em sua memória a somatória de todos os pulsos enviados pelo *encoder* desde o início do processo de titulação. Dessa forma, a partir da equação obtida no processo de calibração da bomba é possível relacionar o volume em função do número de pulsos conforme Fórmula 1, em que “V(titulante)” é o volume em mL e “p” é a quantidade de pulsos lidos através do sensor no eixo do motor.

Figura 13: Fluxograma do processo de titulação com a lógica de programação



3.6. TESTE DO VOLUME DA BURETA E ARDUINO

O erro relativo do volume de solução bombeada foi calculado utilizando como referência o volume da bureta calibrada comparando ao apresentado pelo Arduino® Uno em testes realizados para este fim, e seguem conforme Tabela 8.

Tabela 8: Erro relativo do volume apresentado pelo Arduino® Uno

Volume bureta (mL)	Volume Arduino® Uno (mL)	Erro relativo (%)
2,59	2,43	-6,18
2,60	2,46	-5,38
2,60	2,47	-5,00
2,69	2,59	-3,72
2,70	2,56	-5,37
2,71	2,54	-6,27

3,72	3,72	0,00
3,73	3,70	-0,80
3,79	3,64	-3,96
3,80	3,67	-3,42
3,90	3,70	-5,13
3,95	3,74	-5,32
7,60	7,21	-5,13
7,61	7,31	-3,94
7,67	7,17	-6,52
11,10	10,39	-6,40
11,45	10,77	-5,94

Erro médio ± s -4,62 ± 1,85 %

Estes testes demonstraram um erro muito pequeno do volume calculado pelo Arduino® Uno em relação a bureta.

3.7. PADRONIZAÇÃO

Para o processo de padronização da solução de NaOH foram realizadas titulações pelo método clássico avaliando os pontos de viragem pelo analista, com os pontos obtidos pelo Arduino® Uno, esses dados seguem conforme Tabelas 9 e 10 e Figura 14.

Tabela 9: Dados obtidos na padronização da solução de NaOH via método clássico

Massa do padrão (g)	Concentração (mol L ⁻¹)
0,1075	0,200
0,0983	0,192
0,1029	0,197
Média ± s	0,196 ± 0,004

Tabela 10: Dados obtidos na padronização de solução de NaOH via método automático

Massa do padrão (g)	Concentração (mol L ⁻¹)
0,1013	0,204
0,1063	0,201
0,1021	0,203
Média ± s	0,203 ± 0,002

Figura 14: Resultado da padronização de solução de NaOH via método automático



A coloração rosa claro atingida pelo equipamento nos ensaios de padronização demonstram que o ponto de viragem ideal foi atingido em todas elas.

3.8. TITULAÇÃO

Para o processo de titulação da solução titulada (HCl) foram realizadas titulações pelo método clássico avaliando os pontos de viragem obtidos pelo analista, com os pontos obtidos pelo Arduino® Uno, esses dados seguem conforme Tabelas 11 e 12 e Figura 15.

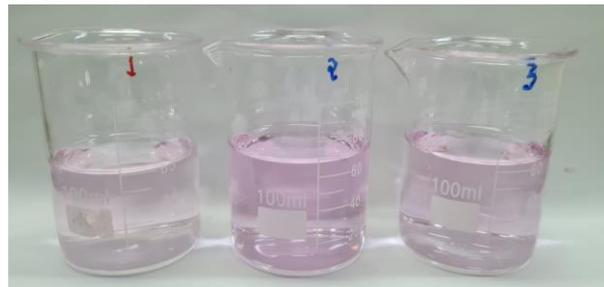
Tabela 11: Dados obtidos na titulação da solução HCl via método clássico

Volume de titulante (mL)	Concentração (mol L ⁻¹)
3,87	0,0758
3,82	0,0748
3,85	0,0754
Média ± s	0,0753 ± 0,0005

Tabela 12: Dados obtidos na titulação da solução HCl via método automático

Volume de titulante (mL)	Concentração (mol L ⁻¹)
3,74	0,0757
3,70	0,0749
3,64	0,0738
Média ± s	0,0748 ± 0,0010

Figura 15: Resultado da titulação da solução HCl via método automático



Tanto no método clássico quanto no automático apontaram um resultado por volta de 0,075 mol L⁻¹ para a concentração da solução de HCl que havia sido preparada para ser de 0,1 mol L⁻¹. Esta diferença pode ser explicada pelo fato do frasco do reagente já ter sido utilizado em outros experimentos na instituição de ensino e ser uma solução muito volátil, conseqüentemente é muito provável que a concentração não se encontrava dentro dos parâmetros do rótulo. Isso demonstra a importância do processo de titulação.

A coloração rosa claro atingida pelo equipamento nos ensaios de titulação demonstram que o ponto de viragem ideal foi atingido em todas elas.

Ao todo foram realizados dezessete ensaios, entre padronização e titulação, com o protótipo e em todos eles o tom de rosa atingido foi muito similar, demonstrando o funcionamento correto, porém os dados utilizados no artigo foram os três últimos de padronização e os três últimos de

titulação por terem sido considerados os mais relevantes.

4. CONSIDERAÇÕES FINAIS

É possível afirmar que conseguimos atingir com sucesso nossa principal tarefa de criar um Titulador Automático de baixo custo. O custo estimado do projeto foi de cerca de R\$ 350,00, enquanto tituladores automáticos comerciais são vendidos a cerca de R\$ 50.000,00.

A partir dos resultados obtidos e apresentados nos testes de padronização, o teste t pareado, utilizado para determinar se existem diferenças entre os valores medidos dos métodos clássico e automático, mostrou que os valores são estatisticamente diferentes, porém o teste Q não indica rejeição de valores para as titulações clássicas e automáticas. O erro relativo entre a técnica clássica e automática foi de +3,36% e o valor de desvio padrão relativo (DPR) foi de 0,91%, indicando uma boa precisão dos dados e que nos permite afirmar que o protótipo pode ser utilizado em ensaios de padronização.

Nos testes de titulação, o teste t-pareado mostrou que não há diferença estatística entre os valores. O erro relativo entre a técnica clássica e automática foi de -0,66% e o valor de desvio padrão relativo (DPR) foi de 1,29%, indicando uma boa precisão dos dados e que nos permite afirmar que o protótipo pode ser utilizado em ensaios de titulação.

Como sugestão para trabalhos futuros pode-se considerar a realização de ensaios de outros tipos de titulação e outros tipos de indicadores, com o levantamento do percentual de cor ideal para o ponto de viragem. Inclusão de uma tabela na memória do Arduino® Uno de indicador e percentual de cor, selecionado pelo usuário através do teclado antes de iniciar o processo de titulação.

AGRADECIMENTOS

Agradecemos a Deus por possibilitar estarmos aqui e nos dar forças para continuarmos.

A nossos familiares por nos apoiar e incentivar a todo momento.

Ao nosso orientador Professor Doutor Fábio Rizzo de Aguiar e coorientadora Thais Taciano por nos ajudarem e apoiarem na execução do projeto e na elaboração do artigo.

Aos Professores Doutor Alexandre Barros, Doutor Klauss Engelmann, e Márcia da Silva, que nos incentivaram, deram sugestões e nos motivaram durante a execução do projeto.

A todos os Professores da ETEC Irmã Agostina que repartiram seus conhecimentos, experiências, nos inspiram e nos fazem acreditar em nossos potenciais e sonhos.

APÊNDICES

APÊNDICE A – Diagrama elétrico

APÊNDICE B – Código da programação

REFERÊNCIAS

BATISTA, C. Titulação. **Toda Matéria**, c2022. Disponível em: <<https://www.todamateria.com.br/titulacao/>>. Acesso em: 15 nov. 2022.

BOMBA PERISTÁLTICA - COMO FUNCIONA? **MaxLabor**, 2015. Disponível em: <<https://www.maxlabor.com.br/blog/bomba-peristaltica-como-funciona>>. Acesso em: 28 de nov. 2022.

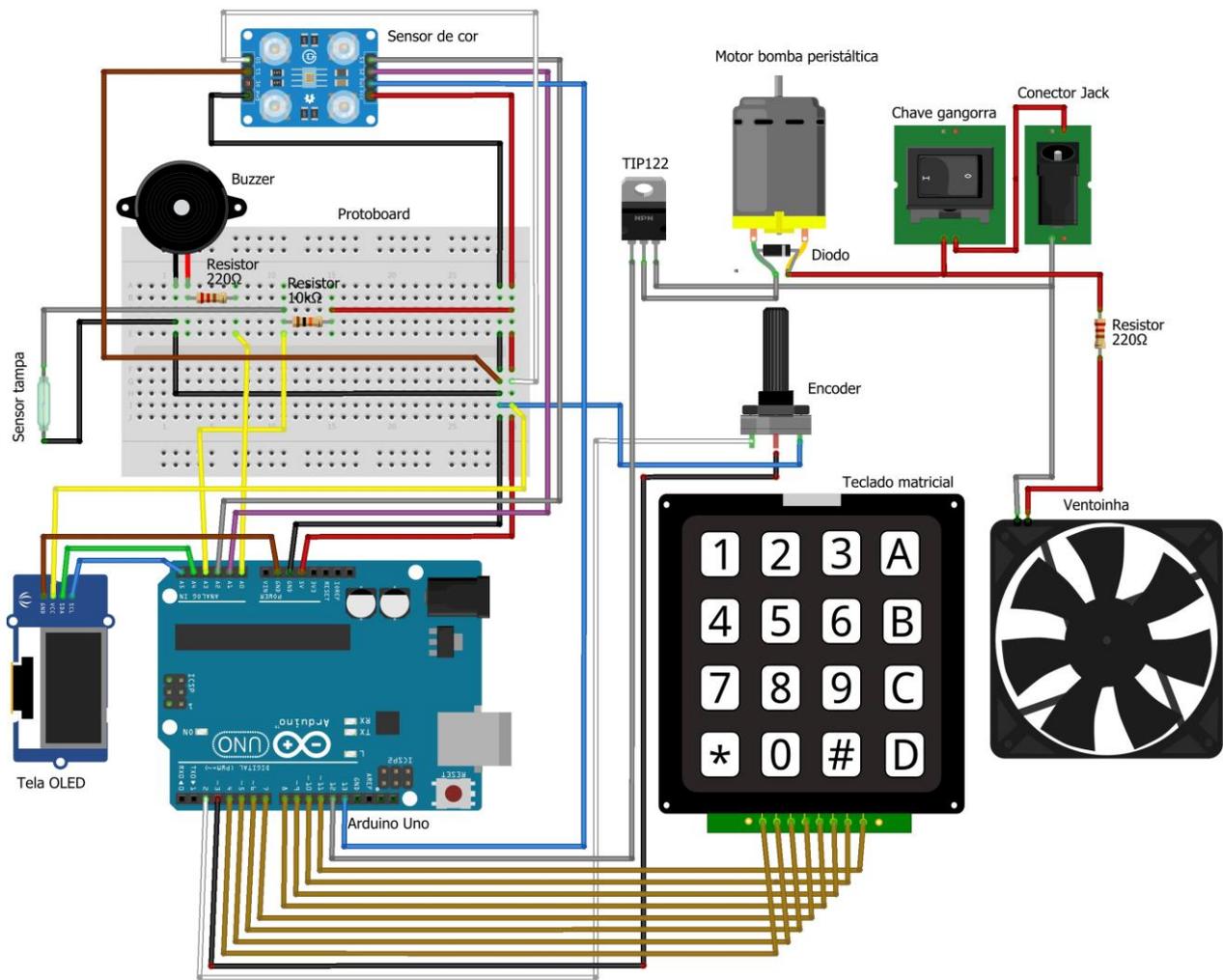
CARPENTER, E. M. Erros em experimento de titulação. **eHow Brasil**, 2021. Disponível em: <https://www.ehow.com.br/erros-experimentos-titulacao-info_66934/>. Acesso em: 16 de nov. 2022.

DOCUMENTO DE REFERÊNCIA DE LINGUAGEM DO ARDUINO. **Arduino.cc**, 2022. Disponível em: <<https://www.arduino.cc/reference/pt/>>. Acesso em: 28 de nov. 2022.

O QUE É ENCODER? **Dynapar**, c2022. Disponível em: <<https://www.dynaparencoders.com.br/blog/o-que-e-encoder/>>. Acesso em: 28 de nov. 2022.

OLIVEIRA, E. Como usar com Arduino – Sensor de Cor TCS230 (TCS3200). **Blog MasterWalker**, [2020?]. Disponível em: <<https://blogmasterwalkershop.com.br/arduino/como-usar-com-arduino-sensor-de-cor-tcs230-tcs3200>>. Acesso em: 28 de nov. 2022.

APÊNDICE A – Diagrama elétrico




```

//int encoderLastState;
int a;
int g;
int modo;
float tempoLeitura;
int tempobuzzer;

unsigned int valorVermelho = 0;
unsigned int valorVerde = 0;
unsigned int valorAzul = 0;
unsigned int valorBranco = 0;

String concentracao;
String voltitulado;
String esteqtitulante;
String esteqtitulado;
float Concentracao;
String massa;
String MM;

void setup() {

    // Inicializa Display e mostra o logo
    lcd.begin();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.draw(logo, 120, 48);
    delay(5000);
    inicioLCD();
    g=0;
    tempobuzzer=200;

    concentracao = "";
    voltitulado = "";
    esteqtitulante = "";
    esteqtitulado = "";
    massa = "";
    MM = "";

    // Declara o pino do motor como saída:
    pinMode(motorPin, OUTPUT);

    // Declara os pinos de seleção de cor como saída:
    pinMode(S2, OUTPUT);
    pinMode(S3, OUTPUT);

    // Declara o pino do sensor de cor como entrada:
    pinMode(sensorOut, INPUT);
    // Declara os pinos do encoder como entrada:
    pinMode(encoder0PinA, INPUT_PULLUP);
    pinMode(encoder0PinB, INPUT_PULLUP);
    // Declara o pino do sensor da tampa como entrada.
    pinMode(sensortampa,INPUT);

    pinMode(BUZZER,OUTPUT); // define o pino do buzzer como saída.

    attachInterrupt(0, contarpulso, CHANGE); // encoder pin on interrupt 0 - pin 2

```

```

// Inicializa comunicação serial
Serial.begin(9600);

}
void loop() {

início:

percentualRef = 13;
percentualMax = 0;
motorOncounter = 0;
referenciaVermelho =0;
referenciaAzul =0;
referenciaVerde =0;
referenciaBranco =0;
percentualVermelho=0;
percentualAzul=0;
percentualVerde=0;
percentualBranco=0;
tempoLeitura=3000;

char leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado

if (leitura_teclas == '1') { // Se a tecla 1 foi pressionada
  buzzer();
  ambientando:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  lcd.println("Ambientando...");
  lcd.println("Pressione < p/");
  lcd.println("parar e voltar");
  delay (400);
  digitalWrite(motorPin, HIGH);
  while (leitura_teclas != '<'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();goto ambientando;}
  }
  buzzer();
  digitalWrite(motorPin, LOW);
  inicioLCD();
  delay (400);
  goto início;
}

if (leitura_teclas == '2') { // Se a tecla 2 foi pressionada
  buzzer();
  delay (400);
  //t = 1;
  leitura_teclas = "";

dados:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  //lcd.println("Escolha o item");
  lcd.println("1 Dados conc.");
  lcd.println("2 Dados massa");
  lcd.println("3 Sem dados");
}
}

```

```

modo = 0;

while (leitura_teclas != '1' && leitura_teclas != '2' && leitura_teclas != '3'){
  leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
  if (leitura_teclas == '<') {
    buzzer();delay (200);inicioLCD();goto inicio;
  }
}

if (leitura_teclas == '1') { // Se a tecla 1 foi pressionada
  modo = 1;
  buzzer();
  delay (400);
  //t = 1;
  leitura_teclas = "";

concentracaotitulante:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  lcd.println("Conc. (mol/L)");
  lcd.println("do titular:");
  concentracao="";
  //Serial.println(leitura_teclas);
  while (leitura_teclas != '>'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<') {
      buzzer();delay (200);concentracao="";Serial.println(concentracao);inicioLCD();goto dados;
    }
    if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.'){
      buzzer();delay (400);
      Serial.println(leitura_teclas);
      concentracao.concat(leitura_teclas);
      Serial.println(concentracao);
      limpaLCD();
      lcd.setFontSize(FONT_SIZE_MEDIUM);
      lcd.println("Conc. (mol/L)");
      lcd.println("do titular:");
      lcd.print(concentracao);
    }
  }
  leitura_teclas = "";buzzer();delay (400);

volumetitulado:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  lcd.println("Volume (mL)");
  lcd.println("do titulado:");
  voltitulado="";
  while (leitura_teclas != '>'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<') {
      buzzer();delay (400);voltitulado="";limpaLCD();goto concentraotitulante;
    }

    if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.'){
      buzzer();delay (400);

```

```

    voltitulado.concat(leitura_teclas);
    limpaLCD();
    lcd.setFontSize(FONT_SIZE_MEDIUM);
    lcd.println("Volume (mL)");
    lcd.println("do titulado:");
    lcd.print(voltitulado);
  }
}
leitura_teclas = "";buzzer();delay (400);

}

if (leitura_teclas == '2') { // Se a tecla 2 foi pressionada
  modo = 2;
  buzzer();
  delay (400);
  //t = 1;
  leitura_teclas = "";

massatitulante:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  lcd.println("Massa (g)");
  lcd.println("do titulado:");
  massa="";
  //Serial.println(leitura_teclas);
  while (leitura_teclas != '>'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<') {
      buzzer();delay (200);massa="";Serial.println(massa);inicioLCD();goto dados;
    }
    if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.') {
      buzzer();delay (400);
      Serial.println(leitura_teclas);
      massa.concat(leitura_teclas);
      Serial.println(massa);
      limpaLCD();
      lcd.setFontSize(FONT_SIZE_MEDIUM);
      lcd.println("Massa (g)");
      lcd.println("do titulado:");
      lcd.print(massa);
    }
  }
  leitura_teclas = "";buzzer();delay (400);

MMtitulado:
  limpaLCD();
  lcd.setFontSize(FONT_SIZE_MEDIUM);
  lcd.println("Massa molar");
  lcd.println("do titulado:");
  MM="";
  while (leitura_teclas != '>'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<') {
      buzzer();delay (400);MM="";limpaLCD();goto massatitulante;
    }
  }
}

```

```

    if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.') {
        buzzer();delay (400);
        MM.concat(leitura_teclas);
        limpaLCD();
        lcd.setFontSize(FONT_SIZE_MEDIUM);
        lcd.println("Massa molar");
        lcd.println("do titulado:");
        lcd.print(MM);
    }
}
leitura_teclas = "";buzzer();delay (400);
}

```

```

if (leitura_teclas == '3') { // Se a tecla 3 foi pressionada
    modo = 3;
    buzzer();
    delay (400);
    //t = 1;
    leitura_teclas = "";
    goto iniciartitulacao;
}

```

estequiometriaTitulante:

```

limpaLCD();
lcd.setFontSize(FONT_SIZE_MEDIUM);
lcd.println("Coef. esteq.");
lcd.println("do titulante:");
lcd.print(esteqtitulante);
esteqtitulante="";
while (leitura_teclas != '>'){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<') {
        buzzer();delay (400);esteqtitulante="";limpaLCD();goto dados;
    }
    if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.') {
        buzzer();delay (400);
        esteqtitulante.concat(leitura_teclas);
        limpaLCD();
        lcd.setFontSize(FONT_SIZE_MEDIUM);
        lcd.println("Coef. esteq.");
        lcd.println("do titulante:");
        lcd.print(esteqtitulante);
    }
}
leitura_teclas = "";buzzer();delay (400);

```

estequiometriaTitulado:

```

limpaLCD();
lcd.setFontSize(FONT_SIZE_MEDIUM);
lcd.println("Coef. esteq.");
lcd.println("do titulado:");
lcd.print(esteqtitulado);
esteqtitulado="";

```

```

while (leitura_teclas != '>'){
  leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
  if (leitura_teclas == '<') {
    buzzer();delay (400);esteqtitulado="";limpaLCD();goto dados;
  }
  if (leitura_teclas == '1' || leitura_teclas == '2' || leitura_teclas == '3' || leitura_teclas == '4' || leitura_teclas ==
'5' || leitura_teclas == '6' || leitura_teclas == '7' || leitura_teclas == '8' || leitura_teclas == '9' || leitura_teclas == '0'
|| leitura_teclas == '.'){
    buzzer();delay (400);
    esteqtitulado.concat(leitura_teclas);
    limpaLCD();
    lcd.setFontSize(FONT_SIZE_MEDIUM);
    lcd.println("Coef. esteq.");
    lcd.println("do titulado.");
    lcd.print(esteqtitulado);
  }
}
leitura_teclas = "";buzzer();delay (400);

```

iniciartitulacao:

```

limpaLCD();
lcd.setFontSize(FONT_SIZE_MEDIUM);
lcd.println("Pressione >");
lcd.println("para iniciar");
while (leitura_teclas != '>'){
  leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
  if (leitura_teclas == '<') {
    buzzer();delay (400);limpaLCD();goto dados;
  }
}
leitura_teclas = "";buzzer();delay (400);

```

```

limpaLCD();
lcd.setFontSize(FONT_SIZE_MEDIUM);
lcd.println("Titulando...");
lcd.println("");
lcd.println("< para pausar");

```

```

leitura();if (g==1){goto fim;}
referenciaVermelho = abs(mediaVermelho);
referenciaAzul = abs(mediaAzul);
referenciaVerde = abs(mediaVerde);
referenciaBranco = abs(mediaBranco);

```

```

startTime = millis();
while (millis() - startTime < 200){
  leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
  if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}
  //if (leitura_teclas == '<') {buzzer();goto fim;}
  if (analogRead(sensor Tampa) > 500) {buzzer();tampaaberta();}

  digitalWrite(motorPin, HIGH);
}

digitalWrite(motorPin, LOW);

```

```

Serial.print("Pulsos: ");
Serial.println(motorOncounter);

leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}
//if (leitura_teclas == '<') {buzzer();goto fim;}
if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();}
leitura();if (g==1){goto fim;}

while (percentualMax < percentualRef){

leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}
//if (leitura_teclas == '<') {buzzer();goto fim;}
if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();}
percentualVermelho = abs(referenciaVermelho-mediaVermelho)*100/mediaVermelho;
percentualAzul = abs(referenciaAzul-mediaAzul)*100/mediaAzul;
percentualVerde = abs(referenciaVerde-mediaVerde)*100/mediaVerde;
percentualBranco = abs(referenciaBranco-mediaBranco)*100/mediaBranco;
percentualMax =
max(max(max(max(max(max(percentualVermelho,percentualAzul),max(percentualVermelho,percentualVerde
)),max(percentualVermelho,percentualBranco)),max(percentualAzul,percentualVerde)),max(percentualAzul,p
ercentualBranco)),max(percentualVerde,percentualBranco));
Serial.print("PercentualMax1: ");
Serial.print(percentualMax);
Serial.println("%");
Serial.print("PercentualR = ");
Serial.print(percentualVermelho);
Serial.println("%");
Serial.print("ReferênciaV = ");
Serial.println(referenciaVermelho);
Serial.print("PercentualG = ");
Serial.print(percentualVerde);
Serial.println("%");
Serial.print("ReferênciaG = ");
Serial.println(referenciaVerde);
Serial.print("PercentualB = ");
Serial.print(percentualAzul);
Serial.println("%");
Serial.print("ReferênciaB = ");
Serial.println(referenciaAzul);
Serial.print("PercentualW = ");
Serial.print(percentualBranco);
Serial.println("%");
Serial.print("ReferênciaW = ");
Serial.println(referenciaBranco);

startTime = millis();
while (millis() - startTime < min(30*((percentualRef+1)-min(percentualMax,percentualRef)),390)){

leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}
//if (leitura_teclas == '<') {buzzer();goto fim;}
if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();}

digitalWrite(motorPin, HIGH);
}
digitalWrite(motorPin, LOW);
Serial.print("Pulsos: ");
Serial.println(motorOncounter);

```

```

leitura(); if (g==1){goto fim;}

if (percentualMax >= percentualRef) {

    delay (15000);
    leitura();if (g==1){goto fim;}
    percentualVermelho = abs(referenciaVermelho-mediaVermelho)*100/mediaVermelho;
    percentualAzul = abs(referenciaAzul-mediaAzul)*100/mediaAzul;
    percentualVerde = abs(referenciaVerde-mediaVerde)*100/mediaVerde;
    percentualBranco = abs(referenciaBranco-mediaBranco)*100/mediaBranco;
    percentualMax =
max(max(max(max(max(percentualVermelho,percentualAzul),max(percentualVermelho,percentualVerde
)),max(percentualVermelho,percentualBranco)),max(percentualAzul,percentualVerde)),max(percentualAzul,p
ercentualBranco)),max(percentualVerde,percentualBranco));
    Serial.print("PercentualR = ");
    Serial.print(percentualVermelho);
    Serial.println("%");
    Serial.print("ReferênciaV = ");
    Serial.println(referenciaVermelho);
    Serial.print("PercentualG = ");
    Serial.print(percentualVerde);
    Serial.println("%");
    Serial.print("ReferênciaG = ");
    Serial.println(referenciaVerde);
    Serial.print("PercentualB = ");
    Serial.print(percentualAzul);
    Serial.println("%");
    Serial.print("ReferênciaB = ");
    Serial.println(referenciaAzul);
    Serial.print("PercentualW = ");
    Serial.print(percentualBranco);
    Serial.println("%");
    Serial.print("ReferênciaW = ");
    Serial.println(referenciaBranco);

pontofinal:
while (percentualMax < percentualRef){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}
    //if (leitura_teclas == '<') {buzzer();goto fim;}
    if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();}
    Serial.println("Entrou na segunda etapa");

    tempoLeitura=8000;

startTime = millis();
while (millis() - startTime < 50){
    leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    if (leitura_teclas == '<' ) {buzzer();pausa();} if (g == 1){goto fim;}    if (analogRead(sensortampa) > 500)
{buzzer();tampaaberta();}
    //if (leitura_teclas == '<') {buzzer();goto fim;}
    if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();}

    digitalWrite(motorPin, HIGH);
}
digitalWrite(motorPin, LOW);

```

```
Serial.print("Pulsos: ");
Serial.println(motorOncounter);
```

```
leitura(); if (g==1){goto fim;}
percentualVermelho = abs(referenciaVermelho-mediaVermelho)*100/mediaVermelho;
percentualAzul = abs(referenciaAzul-mediaAzul)*100/mediaAzul;
percentualVerde = abs(referenciaVerde-mediaVerde)*100/mediaVerde;
percentualBranco = abs(referenciaBranco-mediaBranco)*100/mediaBranco;
percentualMax =
max(max(max(max(max(max(percentualVermelho,percentualAzul),max(percentualVermelho,percentualVerde
)),max(percentualVermelho,percentualBranco)),max(percentualAzul,percentualVerde)),max(percentualAzul,p
ercentualBranco)),max(percentualVerde,percentualBranco));
Serial.print("PercentualR = ");
Serial.print(percentualVermelho);
Serial.println("%");
Serial.print("ReferênciaV = ");
Serial.println(referenciaVermelho);
Serial.print("PercentualG = ");
Serial.print(percentualVerde);
Serial.println("%");
Serial.print("ReferênciaG = ");
Serial.println(referenciaVerde);
Serial.print("PercentualB = ");
Serial.print(percentualAzul);
Serial.println("%");
Serial.print("ReferênciaB = ");
Serial.println(referenciaAzul);
Serial.print("PercentualW = ");
Serial.print(percentualBranco);
Serial.println("%");
Serial.print("ReferênciaW = ");
Serial.println(referenciaBranco);
}
```

```
delay (15000);
leitura(); if (g==1){goto fim;}
percentualVermelho = abs(referenciaVermelho-mediaVermelho)*100/mediaVermelho;
percentualAzul = abs(referenciaAzul-mediaAzul)*100/mediaAzul;
percentualVerde = abs(referenciaVerde-mediaVerde)*100/mediaVerde;
percentualBranco = abs(referenciaBranco-mediaBranco)*100/mediaBranco;
percentualMax =
max(max(max(max(max(max(percentualVermelho,percentualAzul),max(percentualVermelho,percentualVerde
)),max(percentualVermelho,percentualBranco)),max(percentualAzul,percentualVerde)),max(percentualAzul,p
ercentualBranco)),max(percentualVerde,percentualBranco));
if (percentualMax < percentualRef){
goto pontofinal;
}
}
}
```

fim:

```
digitalWrite(motorPin, LOW);
tempobuzzer=400; buzzer(); buzzer(); buzzer();tempobuzzer=200;
Serial.println("Fim da titulação");
//Volume = (motorOncounter - 1508.7)/5942.7; //Gráfico 2
//Volume = (-5105.4 + sqrt(pow(5105.4,2)-(4*187.28 * (1999.7-motorOncounter))))/(2*187.28); //Gráfico 3
//Volume = (motorOncounter + 38.81)/3800.6; //Gráfico 2
//Volume = (motorOncounter - 287.70)/23487;
```

```
Volume = 0.000490755 * motorOncounter - 0.00414505;
```

```
Serial.print("Volume: ");
```

```
Serial.print(Volume);
```

```
Serial.println(" mL");
```

```
limpaLCD();
```

```
lcd.setFontSize(FONT_SIZE_SMALL);
```

```
lcd.println("Volume:");
```

```
lcd.setFontSize(FONT_SIZE_MEDIUM);
```

```
lcd.print(Volume);
```

```
lcd.println(" mL");
```

```
g=0;
```

```
if (modo==1){
```

```
Concentracao = (esteqtitulado.toFloat()* concentracao.toFloat()* Volume)/(esteqtitulante.toFloat()*voltitulado.toFloat());
```

```
lcd.setFontSize(FONT_SIZE_SMALL);
```

```
lcd.println("Concentracao:");
```

```
lcd.setFontSize(FONT_SIZE_MEDIUM);
```

```
lcd.print(Concentracao,5);
```

```
lcd.println(" mol/L");
```

```
}
```

```
if (modo==2){
```

```
Concentracao = (esteqtitulante.toFloat() * massa.toFloat())/(MM.toFloat() * esteqtitulado.toFloat() * Volume/1000);
```

```
lcd.setFontSize(FONT_SIZE_SMALL);
```

```
lcd.println("Concentracao:");
```

```
lcd.setFontSize(FONT_SIZE_MEDIUM);
```

```
lcd.print(Concentracao,5);
```

```
lcd.println(" mol/L");
```

```
}
```

```
if (modo==3){
```

```
lcd.setFontSize(FONT_SIZE_SMALL);
```

```
lcd.println("Pulsos:");
```

```
lcd.setFontSize(FONT_SIZE_MEDIUM);
```

```
lcd.println(motorOncounter);
```

```
}
```

```
while (leitura_teclas != '<'){
```

```
leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
```

```
}
```

```
buzzer();inicioLCD();g=0;concentracao="";voltitulado="";esteqtitulante="";esteqtitulado="";massa="";MM="";goto inicio;
```

```
}
```

```
}
```

```
void leitura(){
```

```
char leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
```

```
startTime = millis();
```

```
i=1;
```

```
somaVermelho=0;
```

```
somaAzul=0;
```

```
somaVerde=0;
```

```
somaBranco=0;
```

```

while(millis() - startTime < tempoLeitura){ // Faz leituras durante Tempo de leitura
  leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
  if (leitura_teclas == '<') {buzzer();pausa();leitura();}
  //if (leitura_teclas == '<') {buzzer();g=1;}
  if (analogRead(sensortampa) > 500) {buzzer();tampaaberta();leitura();}

  //Vermelho
  digitalWrite(S2, LOW);
  digitalWrite(S3, LOW);
  valorVermelho = pulseIn(sensorOut, digitalRead(sensorOut) == HIGH ? LOW : HIGH);

  //Sem filtro
  digitalWrite(S2, HIGH);
  valorBranco = pulseIn(sensorOut, digitalRead(sensorOut) == HIGH ? LOW : HIGH);

  //Azul
  digitalWrite(S2, LOW);
  digitalWrite(S3, HIGH);
  valorAzul = pulseIn(sensorOut, digitalRead(sensorOut) == HIGH ? LOW : HIGH);

  //Verde
  digitalWrite(S2, HIGH);
  valorVerde = pulseIn(sensorOut, digitalRead(sensorOut) == HIGH ? LOW : HIGH);

  somaVermelho = somaVermelho + valorVermelho;
  somaAzul = somaAzul + valorAzul;
  somaVerde = somaVerde + valorVerde;
  somaBranco = somaBranco + valorBranco;
  i++;
}

//delay(3000);
Serial.print("Leituras = ");
Serial.println(i);
mediaVermelho = abs(somaVermelho)/i;
mediaAzul = abs(somaAzul)/i;
mediaVerde = abs(somaVerde)/i;
mediaBranco = abs(somaBranco)/i;
Serial.print("MédiaR = ");
Serial.println(mediaVermelho);
Serial.print("MédiaG = ");
Serial.println(mediaVerde);
Serial.print("MédiaB = ");
Serial.println(mediaAzul);
Serial.print("MédiaW = ");
Serial.println(mediaBranco);
}

void inicioLCD(){
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.setFontSize(FONT_SIZE_SMALL);
  lcd.println("Titulador Automatico");
  lcd.println("");
}

```

```

    lcd.setFontSize(FONT_SIZE_MEDIUM);
    lcd.println("Escolha o item");
    lcd.println("1 Ambientar");
    lcd.println("2 Titular");
}

void limpaLCD(){
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.setFontSize(FONT_SIZE_SMALL);
    lcd.println("Titulador Automatico");
    lcd.println("");
}

void pausa(){
    digitalWrite(motorPin, LOW);
    char leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    delay (400);
    limpaLCD();
    lcd.setFontSize(FONT_SIZE_MEDIUM);
    //lcd.println("Pausado");
    lcd.print(motorOncounter);
    lcd.println(" Pulsos");
    lcd.println("> p/ continuar");
    lcd.println("< p/ cancelar ");
    g=0;

    while (leitura_teclas != '>' && leitura_teclas != '<'){
        leitura_teclas = teclado_personalizado.getKey(); // Realiza leitura do teclado
    }
    buzzer();
    limpaLCD();
    if (leitura_teclas == '>'){
        lcd.setFontSize(FONT_SIZE_MEDIUM);
        lcd.println("Titulando...");
        lcd.println("");
        lcd.println("< para pausar");
    }else{
        g=1;
    }
}

}

void buzzer(){
    digitalWrite(BUZZER, HIGH); //liga o buzzer
    delay(tempobuzzer); //fica com ele ligado durante 200 milissegundos
    digitalWrite(BUZZER, LOW); //desliga o buzzer
    delay(tempobuzzer);
}

void tampaaberta(){
    digitalWrite(motorPin, LOW);
    buzzer();
    limpaLCD();
    lcd.setFontSize(FONT_SIZE_MEDIUM);
    lcd.println("Tampa aberta");
    while (analogRead(sensortampa) > 500){}
    buzzer();
}

```

```
limpaLCD();  
lcd.setFontSize(FONT_SIZE_MEDIUM);  
lcd.println("Titulando...");  
lcd.println("");  
lcd.println("< para pausar");  
}  
  
void contarpulso(){  
    if (digitalRead(encoder0PinA) == digitalRead(encoder0PinB)) {  
        motorOncounter++;  
    } else {  
        motorOncounter--;  
    }  
}
```