



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE ESTUDOS COSTEIROS  
CAMPUS UNIVERSITÁRIO DE BRAGANÇA  
FACULDADE DE CIÊNCIAS BIOLÓGICAS

KELVIN MARLON BARBOSA LACERDA

**DESENVOLVIMENTO DE PROTÓTIPOS COM ARDUINO PARA  
MONITORAMENTO DE PARÂMETROS DA ÁGUA NO CULTIVO DE  
*Macrobrachium amazonicum* (Heller, 1862)**

Bragança

2026

UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE ESTUDOS COSTEIROS  
CAMPUS UNIVERSITÁRIO DE BRAGANÇA  
FACULDADE DE CIÊNCIAS BIOLÓGICAS

KELVIN MARLON BARBOSA LACERDA

**DESENVOLVIMENTO DE PROTÓTIPOS COM ARDUINO PARA  
MONITORAMENTO DE PARÂMETROS DA ÁGUA NO CULTIVO DE  
*Macrobrachium amazonicum* (Heller, 1862)**

Trabalho de Conclusão de Curso apresentado à Faculdade de Ciências Biológicas do Instituto de Estudos Costeiros da Universidade Federal do Pará, como requisito parcial para obtenção do título de Licenciado Pleno em Ciências Biológicas.

Orientador: Prof. Dr. Carlos Murilo Tenório Maciel.

Bragança

2026

KELVIN MARLON BARBOSA LACERDA

**DESENVOLVIMENTO DE PROTÓTIPOS COM ARDUINO PARA  
MONITORAMENTO DE PARÂMETROS DA ÁGUA NO CULTIVO DE  
*Macrobrachium amazonicum* (Heller, 1862)**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Licenciado” e aprovado em sua forma final pelo Curso de Ciências Biológicas, obtendo conceito \_\_\_\_\_.

Bragança, \_\_\_\_\_ de fevereiro de 2026.

\_\_\_\_\_  
Prof. Dra. Rosigleyse Correa de Sousa Félix.  
Coordenadora do curso

**Banca Examinadora:**

\_\_\_\_\_  
Prof. Dr. Carlos Murilo Tenório Maciel.  
Orientador  
Universidade Federal do Pará

\_\_\_\_\_  
Prof. Dr. Colin Robert Beasley.  
Universidade Federal do Pará

\_\_\_\_\_  
Prof. Msc. Bruna Ramalho Maciel.  
Universidade Federal do Pará

Dedico este trabalho aos meus pais, Maria Vanderleia e José Eufrásio, e aos meus melhores amigos, João Vitor, Laysa de Fátima, Dayane Cunha e Vilsiane Costa, que sempre estiveram ao meu lado, oferecendo apoio incondicional, incentivo e confiança em todos os momentos da minha trajetória, sendo a base de tudo que conquistei até aqui.

## AGRADECIMENTOS

Agradeço primeiramente a Deus por me guiar e fortalecer durante essa caminhada.

Aos meus orientadores, que considero meus pais científicos, Prof. Dr. Carlos Murilo Tenório Maciel e Profa. Dra. Cristiana Ramalho Maciel, por terem me recebido em seus laboratórios, acreditado no meu potencial e me acompanhado de forma constante ao longo deste trabalho, oferecendo orientação, conhecimento, apoio e incentivo fundamentais para minha formação científica e pessoal.

A todos os integrantes dos laboratórios LAQUA e BioDATTA, pelo convívio, colaboração e pelo ambiente de aprendizado que contribuíram diretamente para o desenvolvimento deste projeto e para minha formação acadêmica.

Aos meus pais americanos, Dallas e Rosângela Flickinger, que me acolheram com carinho e generosidade, proporcionando-me apoio e fazendo com que eu me sentisse em casa mesmo estando longe do meu país.

À minha turma, a Bio20, pelos momentos compartilhados, aprendizados, desafios e companheirismo ao longo dessa jornada.

À professora Lilia Rodrigues, que me incentivava desde cedo e teve papel fundamental em minha formação, contribuindo significativamente para que eu chegasse até este momento.

Aos professores, técnicos e demais funcionários da Universidade Federal do Pará, em especial do Campus Bragança, pelo suporte, ensinamentos e contribuição para minha formação acadêmica.

Agradeço também à ferramenta de inteligência artificial ChatGPT (OpenAI), pelo apoio na revisão textual, organização estrutural e esclarecimento de dúvidas técnicas relacionadas à programação.

Por fim, agradeço a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho e para minha formação como biólogo e pesquisador.

“Ensinar não é transferir conhecimento  
mas criar as possibilidades para a sua  
própria produção ou a sua construção.”

— Paulo Freire

## RESUMO

O camarão de água doce *Macrobrachium amazonicum* (Heller, 1862) possui elevada relevância produtiva, social e econômica na Região Amazônica, sendo amplamente utilizado tanto na pesca artesanal quanto em sistemas de cultivo. Apesar dos avanços científicos relacionados ao desenvolvimento de técnicas de manejo e pacotes tecnológicos para a espécie, a adoção de ferramentas de monitoramento ambiental ainda é limitada pelo alto custo, complexidade e baixa acessibilidade dos equipamentos comerciais disponíveis. Nesse contexto, este trabalho teve como objetivo desenvolver e avaliar protótipos de baixo custo, baseados na plataforma Arduino, destinados ao monitoramento de parâmetros físico-químicos da água em sistemas de cultivo de *M. amazonicum*. Foram desenvolvidos dispositivos para medição de pH, temperatura, vazão, condutividade elétrica (estimada por TDS) e turbidez. Os protótipos foram testados em ambiente laboratorial, sob condições associadas à larvicultura, sendo avaliados quanto à precisão, estabilidade, repetibilidade e aplicabilidade prática por meio da comparação com equipamentos comerciais de referência. O sensor de pH passou por calibração formal utilizando soluções tampão padrão (pH 4,0; 7,0 e 10,0), enquanto o sensor de temperatura foi verificado por comparação direta com instrumentos laboratoriais, assegurando confiabilidade às medições. Também foi realizada análise estimativa de custos com base nos componentes utilizados, evidenciando a viabilidade técnica e econômica da replicação dos sistemas. Os protótipos de pH e temperatura apresentaram desempenho compatível com equipamentos comerciais, indicando viabilidade para monitoramento contínuo. O sensor de vazão apresentou desempenho satisfatório em fluxo contínuo, mas foi descontinuado devido à interferência de bolhas de ar na aplicação pretendida. Os sensores de turbidez e TDS mostraram limitações de precisão e sensibilidade, demandando ajustes e calibração mais refinada. A utilização de impressão 3D, associada à implementação de alimentação por bateria recarregável e sistema experimental de registro automatizado de dados, ampliou a portabilidade e o potencial de aplicação dos dispositivos. Conclui-se que os protótipos desenvolvidos constituem alternativa acessível e tecnicamente promissora para o monitoramento ambiental em sistemas aquícolas de pequeno porte.

**Palavras-chave:** Aquicultura; Camarão-da-Amazônia; *hardware*; qualidade da água; sensores; *software*.

## ABSTRACT

The freshwater shrimp *Macrobrachium amazonicum* (Heller, 1862) has high productive, social, and economic relevance in the Amazon Region, being widely used in both artisanal fisheries and aquaculture systems. Despite scientific advances in cultivation techniques and technological packages for the species, the adoption of environmental monitoring tools remains limited due to the high cost, complexity, and low accessibility of commercial equipment. In this context, the present study aimed to develop and evaluate low-cost prototypes based on the Arduino platform for monitoring physicochemical water parameters in *M. amazonicum* cultivation systems. Devices were developed to measure pH, temperature, water flow rate, electrical conductivity (estimated by TDS), and turbidity. The prototypes were tested in a laboratory environment under conditions associated with larviculture and evaluated regarding accuracy, stability, repeatability, and practical applicability through comparison with commercial reference equipment. The pH sensor underwent formal calibration using standard buffer solutions (pH 4.0, 7.0, and 10.0), while the temperature sensor was verified through direct comparison with laboratory instruments, ensuring measurement reliability. An estimated cost analysis was also conducted based on the electronic components and materials used, demonstrating the technical and economic feasibility of system replication. The pH and temperature prototypes showed performance compatible with commercial equipment, indicating feasibility for continuous environmental monitoring. The flow sensor performed satisfactorily under continuous flow conditions but was discontinued due to air bubble interference in the intended application. The turbidity and TDS sensors presented limitations in precision and sensitivity, requiring further refinement and calibration. The use of 3D printing, combined with the implementation of rechargeable battery power supply and an experimental automated data logging system, enhanced device portability and application potential. The developed prototypes represent an accessible and technically promising alternative for environmental monitoring in small-scale aquaculture systems.

**Keywords:** Amazon-river-prawn; Aquaculture; *hardware*; *software*; sensors; water quality.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Tanque em sistema de recirculação utilizando <i>airlift</i> . .....	7
Figura 2 – Soldas realizadas para conectar a bateria/interruptor à placa Arduino. ....	10
Figura 3 – comparação estrutural da placa Arduino antes e depois das alterações. ...	12
Figura 4 – Protótipos em fase inicial contendo sensores de pH, temperatura, turbidez, vazão e condutividade montados em bancada. ....	14
Figura 5 – Protótipos finalizados para monitoramento de pH e temperatura, respectivamente, acondicionados em invólucros produzidos por impressão 3D de ABS. ....	15
Figura 6 – Itens produzidos por impressão 3D para uso laboratorial. ....	21
Figura N.1 – Protótipo inicial de temperatura montado em protoboard com display LCD convencional. ....	43
Figura N.2 – Protótipo inicial de pH montado em protoboard com display LCD I2C. ....	43
Figura T.1 – Vista frontal da disposição do circuito do protótipo de pH semi-acondicionado no invólucro. ....	50
Figura T.2 – Vista frontal do circuito do protótipo de pH acondicionado no invólucro sem tampa. ....	51
Figura T.3 - Vista superior do protótipo final, contemplando o LCD, interruptor e o conector da sonda de pH (ou saída do sensor DS18B20 no caso do protótipo de temperatura) ....	51
Figura T.4 – Vista lateral do protótipo final, contemplando os parafusos que mantém o invólucro fechado. ....	52
Figura T.5 – Vista inferior do protótipo final, contemplando a entrada USB-C para carregamento. ....	52
Figura T.6 – Vista traseira do protótipo final, contemplando pés antiderrapantes/antichoque. ....	53

**LISTA DE TABELAS**

Tabela 1 - Estimativa de custos dos protótipos desenvolvidos. ....	19
Tabela 2 – Comparativo de alguns protótipos com produtos comerciais.....	20
Tabela P.1 – Características do Arduino Uno R3 e do Arduino Nano V3.....	45
Tabela Q.1 – Detalhamento dos custos estimados dos componentes dos protótipos.	46

**LISTA DE ABREVIATURAS E SIGLAS**

3D – Tridimensional  
ABNT – Associação Brasileira de Normas Técnicas  
ABS – Acrilonitrila Butadieno Estireno  
DC – Corrente contínua  
DS18B20 – Sensor digital de temperatura  
I2C – Inter-Integrated Circuit (protocolo de comunicação)  
IoT – Internet das Coisas  
KE – Klipper Extended  
LCD – Display de Cristal Líquido  
NTU – Nephelometric Turbidity Unit  
pH – Potencial hidrogeniônico  
PLA – Ácido Polilático  
R3 – Revision 3  
TDS – Total Dissolved Solids  
UFPA – Universidade Federal do Pará  
USB – Universal Serial Bus  
V3 – Version 3

**LISTA DE SÍMBOLOS**

°C – Grau Celsius  
A – Ampère  
cm – Centímetro  
l – Litro  
mA – Miliampère  
min – Minuto  
ml – Mililitro  
mm – Milímetro  
 $\mu$ S – Microsiemens  
V – Volt  
 $\Omega$  – Ohm

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>16</b>
<b>2</b>	<b>OBJETIVOS.....</b>	<b>18</b>
2.1	Objetivo Geral.....	18
2.2	Objetivos Específicos .....	18
<b>3</b>	<b>REVISÃO DE LITERATURA.....</b>	<b>18</b>
3.1	Importância biológica, social e econômica de <i>Macrobrachium amazonicum</i> .....	18
3.2	Qualidade da água na carcinicultura .....	19
3.3	Parâmetros recomendados para <i>Macrobrachium amazonicum</i> .....	20
<b>4</b>	<b>MATERIAIS E MÉTODOS .....</b>	<b>21</b>
4.1	Local de realização.....	21
4.2	Sistema de Recirculação e filtração .....	21
4.3	Determinação da vazão .....	22
4.4	Plataforma de desenvolvimento .....	22
4.5	Parâmetros da água .....	23
4.6	Montagem dos protótipos.....	23
4.7	Montagem elétrica dos protótipos de monitoramento.....	23
<b>4.7.1</b>	<b>Conexões dos LCDs utilizados.....</b>	<b>24</b>
<b>4.7.2</b>	<b>Montagem do sensor de temperatura .....</b>	<b>25</b>
<b>4.7.3</b>	<b>Montagem do sensor de condutividade elétrica (TDS).....</b>	<b>25</b>
<b>4.7.4</b>	<b>Montagem do sensor de vazão.....</b>	<b>25</b>
<b>4.7.5</b>	<b>Montagem do sensor de pH.....</b>	<b>26</b>
<b>4.7.6</b>	<b>Montagem do sensor de turbidez.....</b>	<b>26</b>
4.8	Mudança estrutural para acondicionamento .....	26
4.9	Calibração e validação dos sensores .....	27
4.10	Validação experimental .....	27
4.11	Análise de custos .....	27

<b>5</b>	<b>RESULTADOS E DISCUSSÃO</b> .....	<b>28</b>
5.1	Protótipos em fase inicial de desenvolvimento .....	28
5.2	Protótipos finais.....	29
5.3	Evolução e otimização dos protótipos.....	29
5.4	Protótipo de pH .....	30
5.5	Protótipo de temperatura.....	30
5.6	Protótipo de vazão.....	31
5.7	Protótipos de turbidez e condutividade elétrica (TDS) .....	31
5.8	Viabilidade técnica e econômica .....	31
5.9	Aplicações complementares da impressão 3D no laboratório.....	34
<b>6</b>	<b>CONCLUSÃO</b> .....	<b>35</b>
<b>7</b>	<b>PERSPECTIVAS FUTURAS</b> .....	<b>37</b>
<b>8</b>	<b>REFERENCIAL BIBLIOGRÁFICO</b> .....	<b>39</b>
<b>9</b>	<b>APÊNDICE A – Lista de componentes do protótipo de pH (versão final)</b> .....	<b>41</b>
<b>10</b>	<b>APÊNDICE B – Lista de componentes do protótipo de temperatura (versão final)</b>	<b>41</b>
<b>11</b>	<b>APÊNDICE C – Lista de componentes do protótipo de vazão (fase inicial)</b> .....	<b>41</b>
<b>12</b>	<b>APÊNDICE D – Lista de componentes do protótipo de turbidez (fase inicial)</b> .	<b>41</b>
<b>13</b>	<b>APÊNDICE E – Lista de componentes do protótipo de condutividade elétrica – TDS (fase inicial)</b> .....	<b>42</b>
<b>14</b>	<b>APÊNDICE F – Código-fonte para calibração inicial do protótipo de pH</b> .....	<b>42</b>
<b>15</b>	<b>APÊNDICE G – Código-fonte do protótipo de pH (versão inicial em protoboard com LCD convencional)</b> .....	<b>43</b>
<b>16</b>	<b>APÊNDICE H – Código-fonte do protótipo de pH (versão final com LCD I2C e alimentação por bateria)</b> .....	<b>45</b>
<b>17</b>	<b>APÊNDICE I – Código-fonte do protótipo de temperatura (versão inicial em protoboard com LCD convencional)</b> .....	<b>47</b>
<b>18</b>	<b>APÊNDICE J – Código-fonte do protótipo de temperatura (versão final com LCD I2C e alimentação por bateria)</b> .....	<b>49</b>
<b>19</b>	<b>APÊNDICE K – Código-fonte do protótipo de vazão (fase inicial)</b> .....	<b>51</b>
<b>20</b>	<b>APÊNDICE L – Código-fonte do protótipo de turbidez (fase inicial)</b> .....	<b>53</b>

21	APÊNDICE M – Código-fonte do protótipo de condutividade elétrica – TDS (fase inicial) .....	54
22	APÊNDICE N – Evolução estrutural dos protótipos de pH e temperatura .....	56
23	APÊNDICE O – Implementação do sistema de alimentação por bateria .....	57
24	APÊNDICE P – Proposta conceitual de substituição do Arduino Uno R3 pelo Arduino Nano V3.....	58
25	APÊNDICE Q – Detalhamento dos custos por componente dos protótipos.....	59
26	APÊNDICE R – Código-fonte do sistema de registro de dados com módulo RTC e armazenamento em cartão SD.....	60
27	APÊNDICE S – Código-fonte do sistema de registro e transmissão remota de dados com módulos RTC, cartão SD e comunicação WiFi .....	62
28	APÊNDICE T – Registros fotográficos complementares dos protótipos finais .	64

## 1 INTRODUÇÃO

O Camarão-da-Amazônia (*Macrobrachium amazonicum* Heller, 1862) é amplamente registrado em atividades de pesca artesanal e em sistemas de cultivo na Região Amazônica, apresentando relevância econômica, social e cultural em diferentes contextos locais (MACIEL; VALENTI, 2009). A exploração dessa espécie contribui para a segurança alimentar e para a geração de renda de populações ribeirinhas, sobretudo por meio da pesca artesanal. No entanto, o aumento do esforço pesqueiro em determinadas regiões tem sido associado à redução de estoques naturais, o que impõe limitações à expansão sustentável da atividade extrativista (MORAES-VALENTI; VALENTI, 2010).

Nesse contexto, a aquicultura tem sido apontada como uma alternativa para a produção controlada de *M. amazonicum*, com potencial para reduzir a pressão sobre os estoques naturais e ampliar a disponibilidade do produto de forma mais previsível (VALENTI et al., 2018). Ao longo das últimas décadas, diversos estudos contribuíram para o avanço do conhecimento sobre o cultivo da espécie, abordando aspectos relacionados à reprodução, larvicultura, engorda e manejo de sistemas aquícolas (CARVALHO, 2004; CHONG-CARRILLO et al., 2015). Apesar desses avanços, parte significativa desse conhecimento permanece restrita ao meio acadêmico, o que dificulta sua aplicação prática, especialmente por produtores de pequena e média escala e em sistemas experimentais com limitações orçamentárias e estruturais.

A qualidade da água é reconhecida como um dos fatores centrais para o sucesso da carcinicultura, uma vez que influencia diretamente o crescimento, a sobrevivência e a sanidade dos organismos cultivados (YODER, 2008). Parâmetros físico-químicos como temperatura, potencial hidrogeniônico (pH), oxigênio dissolvido, vazão, turbidez e condutividade elétrica requerem monitoramento frequente, pois variações fora das faixas recomendadas podem induzir estresse fisiológico, reduzir o desempenho zootécnico e aumentar a suscetibilidade a doenças (MORAES-VALENTI; VALENTI, 2010). Em ambientes experimentais e de produção controlada, o acompanhamento sistemático desses parâmetros é essencial para garantir condições adequadas de cultivo, reprodutibilidade experimental e confiabilidade dos dados obtidos.

Tradicionalmente, o monitoramento desses parâmetros depende de equipamentos comerciais de custo elevado, muitas vezes importados, cuja aquisição e manutenção podem representar entraves financeiros para laboratórios acadêmicos e produtores de menor escala.

Além disso, muitos desses equipamentos não permitem customização, integração com outros sistemas ou adaptação a necessidades específicas de pesquisa e manejo.

A adoção de tecnologias de monitoramento ambiental tem sido indicada como uma estratégia eficiente para otimizar o manejo aquícola e subsidiar a tomada de decisão, contribuindo para maior controle dos sistemas produtivos e potencial redução de custos operacionais (COLEZEA et al., 2018). Nesse cenário, o avanço de ferramentas associadas à Internet das Coisas (IoT) ampliou as possibilidades de monitoramento contínuo, permitindo a integração de sensores, sistemas de aquisição de dados e plataformas computacionais capazes de registrar, armazenar e disponibilizar informações em tempo real. Plataformas de prototipagem eletrônica, como o Arduino, destacam-se nesse contexto por apresentarem arquitetura aberta, baixo custo relativo e elevada flexibilidade para adaptação a diferentes aplicações experimentais.

Segundo a Oracle, a IoT descreve uma rede de objetos físicos incorporados a sensores e softwares com o objetivo de conectar e trocar dados com outros dispositivos e sistemas por meio da internet (ORACLE, 2024). Complementando essas soluções tecnológicas, a impressão 3D surge como uma estratégia eficiente para a construção de invólucros, suportes e acessórios para os protótipos. Essa tecnologia permite fabricar peças personalizadas de forma rápida, econômica e com boa precisão, possibilitando ajustes de projeto iterativos durante o desenvolvimento. Além disso, a utilização de diferentes materiais termoplásticos, como PLA e ABS, oferece flexibilidade quanto à resistência mecânica, durabilidade e adequação a diferentes condições de operação, ampliando a adaptabilidade e funcionalidade dos dispositivos desenvolvidos.

Dessa forma, o desenvolvimento de protótipos acessíveis para o monitoramento da qualidade da água em sistemas de cultivo representa uma alternativa promissora para ampliar o uso de tecnologias de monitoramento em contextos experimentais e de pequeno porte. Considerando *M. amazonicum* como modelo de aplicação, este trabalho propõe o desenvolvimento e a avaliação de protótipos de baixo custo baseados em microcontroladores, voltados ao monitoramento de parâmetros ambientais relevantes. Os dispositivos desenvolvidos são avaliados por meio da comparação com equipamentos comerciais de referência, visando verificar sua precisão, estabilidade, limitações operacionais e aplicabilidade como ferramentas complementares no manejo e no acompanhamento da qualidade da água em sistemas aquícolas experimentais e produtivos.

## 2 OBJETIVOS

### 2.1 Objetivo Geral

Desenvolver e avaliar protótipos de equipamentos de baixo custo para o monitoramento da qualidade da água em sistemas de cultivo do Camarão-da-Amazônia (*M. amazonicum*).

### 2.2 Objetivos Específicos

- a) Desenvolver e construir protótipos baseados em Arduino para monitoramento de parâmetros físico-químicos da água: pH, temperatura, vazão, turbidez e sólidos totais dissolvidos (TDS).
- b) Validar o desempenho dos protótipos em ambiente laboratorial;
- c) Estimar os custos de produção e avaliar a viabilidade técnica e econômica dos equipamentos;
- d) Analisar o potencial de aplicação dos protótipos em sistemas de cultivo em escala produtiva

## 3 REVISÃO DE LITERATURA

### 3.1 Importância biológica, social e econômica de *Macrobrachium amazonicum*

O *M. amazonicum* é uma espécie nativa amplamente distribuída na bacia amazônica e em outros sistemas hidrográficos da América do Sul. Trata-se de um crustáceo decápode de grande plasticidade ecológica, capaz de se adaptar a diferentes condições ambientais, incluindo variações de salinidade, temperatura e disponibilidade de alimento (MACIEL; VALENTI, 2009).

Na Região Amazônica, a espécie possui elevada relevância socioeconômica, sendo explorada principalmente por meio da pesca artesanal e do cultivo em pequena escala. O consumo do Camarão-da-Amazônia está fortemente associado à cultura alimentar local, contribuindo diretamente para a segurança alimentar e geração de renda de comunidades ribeirinhas (MARQUES et al., 2020). Entretanto, o aumento da demanda e a pressão sobre os estoques naturais tornam imprescindível o fortalecimento da aquicultura como alternativa sustentável.

Nas últimas décadas, esforços científicos resultaram na consolidação de um pacote tecnológico para o cultivo de *M. amazonicum*, abrangendo reprodução, larvicultura, engorda e

manejo de sistemas de recirculação (MORAES-VALENTI; VALENTI, 2010; VALENTI et al., 2018). Apesar desses avanços, a adoção prática dessas tecnologias ainda é limitada, sobretudo devido ao custo elevado de equipamentos e à dificuldade de acesso a ferramentas de monitoramento ambiental.

### 3.2 Qualidade da água na carcinicultura

A qualidade da água é um dos fatores mais críticos para o sucesso da carcinicultura, influenciando diretamente o crescimento, a sobrevivência, a conversão alimentar e o bem-estar dos organismos cultivados. Parâmetros como temperatura, pH, oxigênio dissolvido, condutividade elétrica, turbidez e vazão de água devem ser mantidos dentro de faixas adequadas para evitar estresse fisiológico e mortalidade (YODER, 2008; BOYD, 2020).

A temperatura afeta diretamente os processos metabólicos, a taxa de crescimento e o consumo alimentar dos camarões, uma vez que crustáceos são organismos ectotérmicos e dependem da temperatura ambiental para regulação fisiológica (BOYD, 2020; MORAES-RIODADES et al., 2018). Variações bruscas ou valores fora da faixa ideal podem comprometer o desenvolvimento, alterar o ciclo de muda e reduzir a eficiência produtiva.

O pH influencia a toxicidade de compostos nitrogenados, especialmente da amônia não ionizada ( $\text{NH}_3$ ), além de interferir no equilíbrio osmótico dos organismos aquáticos (BOYD, 2020; KUBITZA, 2011). Dessa forma, é considerado um parâmetro essencial de monitoramento contínuo em sistemas aquícolas.

A vazão de água está diretamente relacionada à renovação do meio, à remoção de metabólitos e à manutenção da estabilidade físico-química da água, principalmente em sistemas de recirculação (TIMMONS; EBELING, 2010). Já a turbidez e a condutividade elétrica fornecem informações importantes sobre a presença de partículas em suspensão e a concentração de íons dissolvidos, respectivamente, podendo indicar alterações no ambiente de cultivo (BOYD, 2020).

Tradicionalmente, o monitoramento desses parâmetros é realizado por meio de análises pontuais utilizando equipamentos comerciais ou kits químicos descartáveis, o que pode elevar os custos operacionais e limitar sua aplicação em sistemas de pequena escala (KUBITZA, 2011). Nesse contexto, o desenvolvimento de sensores de baixo custo e sistemas automatizados representa alternativa promissora para ampliar o controle ambiental na carcinicultura

### 3.3 Parâmetros recomendados para *Macrobrachium amazonicum*

No cultivo de *M. amazonicum*, a manutenção adequada dos parâmetros físico-químicos é determinante para o desempenho zootécnico da espécie, influenciando diretamente crescimento, sobrevivência e eficiência alimentar (EMBRAPA, 2016; LIMA; BASTOS, 2019).

O potencial hidrogeniônico (pH) exerce influência direta sobre a fisiologia do camarão, afetando a toxicidade da amônia e o equilíbrio ácido-base (BOYD, 2020). Em estudo conduzido pela Embrapa Amapá, avaliando juvenis de *M. amazonicum* em sistema fechado de recirculação, os valores de pH variaram entre 6,0 e 8,5 ao longo de 60 dias, sem impactos negativos sobre crescimento e sobrevivência (EMBRAPA, 2016).

Na larvicultura comercial, recomenda-se que o pH permaneça preferencialmente entre 7,0 e 8,0, faixa considerada adequada para o desenvolvimento larval e estabilidade fisiológica (MEDEIROS JUNIOR, 2024).

A temperatura regula diretamente o metabolismo e a taxa de crescimento dos crustáceos (BOYD, 2020). Temperaturas próximas a 28 °C favorecem o crescimento e a sobrevivência de *M. amazonicum* (MORAES-RIODADES et al., 2018). Valores inferiores a 25 °C reduzem a atividade metabólica e o consumo alimentar, enquanto temperaturas superiores a 32–35 °C elevam o estresse fisiológico e podem aumentar a mortalidade.

A condutividade elétrica representa a concentração de íons dissolvidos e está relacionada ao equilíbrio osmótico dos organismos aquáticos (BOYD, 2020). No experimento descrito por EMBRAPA (2016), os valores variaram entre 79 e 475  $\mu\text{S}/\text{cm}$ , mantendo bom desempenho produtivo.

A turbidez está associada à presença de partículas suspensas que podem afetar trocas gasosas e a qualidade sanitária da água (TIMMONS; EBELING, 2010). No sistema fechado avaliado por EMBRAPA (2016), a turbidez variou entre 0 e 9 NTU, sendo considerada adequada para juvenis da espécie.

## 4 MATERIAIS E MÉTODOS

### 4.1 Local de realização

O estudo foi conduzido no Laboratório de Aquicultura (LAQUA) e Laboratório de Bioinformática, Desenvolvimento, Aplicação e Transferência de Tecnologia para Aquicultura (BioDATTA), vinculado ao Instituto de Estudos Costeiros da Universidade Federal do Pará (UFPA), Campus Bragança, onde são desenvolvidas atividades de pesquisa relacionadas à carcinicultura e à inovação tecnológica aplicada à aquicultura.

### 4.2 Sistema de Recirculação e filtração

No laboratório, o sistema de circulação de água utilizado funciona por recirculação, com dois movimentos principais: a água sai do tanque por gravidade e retorna por meio de um sistema do tipo *airlift*.

Figura 1 – Tanque em sistema de recirculação utilizando *airlift*.



Fonte: Fotografia de autoria própria (2026).

Primeiramente, a água escoar do tanque de cultivo para o filtro biológico devido à diferença de altura entre eles. Ou seja, como o tanque está em nível mais alto, a água desce naturalmente até o filtro, sem necessidade de bomba nessa etapa. No filtro biológico ocorre o tratamento da água, principalmente a transformação de compostos nitrogenados que podem ser tóxicos aos camarões.

O retorno da água tratada ao tanque acontece por meio de um sistema conhecido como *airlift*. Esse sistema consiste basicamente em um tubo (cano PVC) vertical instalado dentro do filtro, no qual é inserida uma mangueira ligada a um compressor de ar. Quando o ar é liberado na parte inferior interna do tubo, ele forma bolhas que sobem naturalmente até a superfície. À medida que essas bolhas sobem, elas “puxam” a água junto, fazendo com que a água também se desloque para cima dentro do tubo. Dessa forma, o ar funciona como um mecanismo de bombeamento, impulsionando a água de volta para o tanque.

Esse tipo de sistema apresenta algumas vantagens importantes: não possui partes mecânicas submersas que possam entupir facilmente, consome pouca energia e ainda contribui para a oxigenação da água durante o processo de recirculação.

#### 4.3 Determinação da vazão

A vazão avaliada neste trabalho corresponde ao volume de água que retorna do filtro biológico para o tanque por meio do sistema *airlift*. Para medir essa vazão, foi utilizado o método volumétrico direto: o volume de água que sai pelo tubo de retorno é coletado durante 10 segundos em um recipiente graduado. Em seguida, o valor é convertido para litros por minuto ( $L \cdot \text{min}^{-1}$ ), permitindo verificar se o fluxo de recirculação é adequado para o funcionamento do sistema.

Essa metodologia possibilita um controle simples e eficiente do desempenho hidráulico, garantindo que a água esteja circulando em quantidade suficiente para manter a qualidade ambiental adequada ao cultivo.

#### 4.4 Plataforma de desenvolvimento

Para o desenvolvimento dos protótipos foi utilizada a plataforma Arduino Uno R3, selecionada devido à sua arquitetura aberta, baixo custo, ampla disponibilidade no mercado e compatibilidade com diversos sensores comerciais. A programação foi realizada em linguagem C++, utilizando o software Arduino IDE.

Em um computador externo, utilizando o software Arduino IDE, o código-fonte é inicialmente compilado pelo próprio software com a finalidade de verificar erros de sintaxe e inconsistências lógicas. Após a validação, o programa foi transferido para a placa Arduino Uno R3 por meio de conexão USB, sendo gravado na memória do microcontrolador. Esse procedimento permitiu que o dispositivo operasse de forma autônoma, executando continuamente as rotinas de leitura, processamento e transmissão dos dados coletados pelos sensores.

#### 4.5 Parâmetros da água

Foram desenvolvidos protótipos para o monitoramento dos seguintes parâmetros:

- a) pH da água
- b) Temperatura da água (°C)
- c) Vazão de água (L/min)
- d) Turbidez (NTU)
- e) Condutividade elétrica (estimada por TDS) ( $\mu\text{S}/\text{cm}$ )

Cada sistema foi composto por um microcontrolador Arduino Uno R3, sensores específicos para cada parâmetro, display LCD para visualização em tempo real e componentes auxiliares, como interruptores, módulos de carregamento e fontes de alimentação (bateria).

#### 4.6 Montagem dos protótipos

Os circuitos eletrônicos foram inicialmente montados em protoboard, permitindo a realização de testes preliminares de funcionamento, verificação de leituras, identificação de ruídos elétricos e execução dos ajustes necessários. Essa etapa foi fundamental para validação lógica dos códigos, calibração inicial dos sensores e definição da arquitetura elétrica mais adequada para cada protótipo.

Após a validação experimental, os protótipos foram montados em sua configuração final, utilizando display LCD com interface I2C e sistema de alimentação por bateria recarregável, associado a módulo carregador com entrada USB-C.

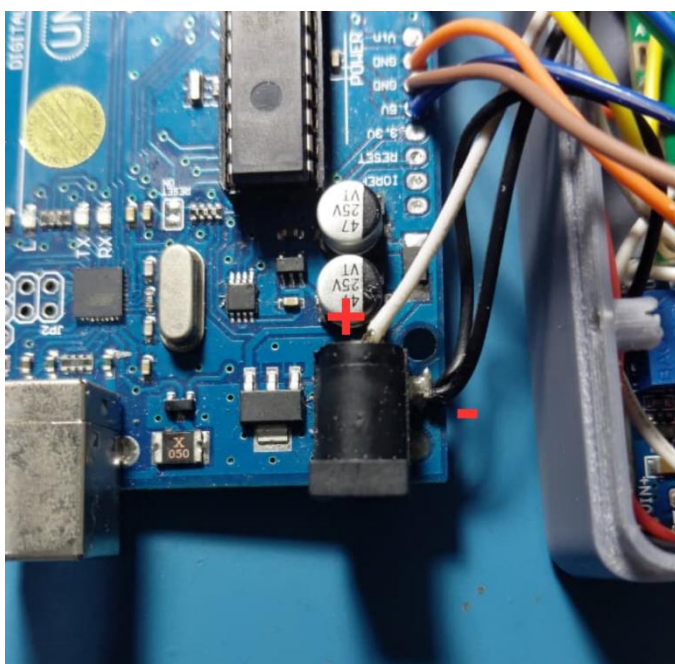
#### 4.7 Montagem elétrica dos protótipos de monitoramento

A montagem elétrica dos protótipos foi realizada com base em documentação técnica especializada e tutoriais consolidados para cada sensor empregado, incluindo o sensor de condutividade Gravity: Analog TDS Sensor / Meter for Arduino (SEN0244) da DFRobot, o

sensor de temperatura DS18B20 descrito pela RoboCore, e os sensores de vazão, pH e turbidez apresentados em tutoriais técnicos elaborados por Matheus Gebert Straub.

O sistema foi alimentado por bateria recarregável de corrente contínua, conectada a módulo carregador dedicado, permitindo recarga externa controlada. A saída do módulo carregador foi conectada a bateria, ajustado para fornecer 9V estáveis. A bateria e interruptor, por sua vez, foram conectadas de forma direta a dois pontos de solda no conector de energia do Arduino, como mostra a figura a seguir:

Figura 2 – Soldas realizadas para conectar a bateria/interruptor à placa Arduino.



Fonte: Fotografia de autoria própria (2026).

#### 4.7.1 Conexões dos LCDs utilizados

O display LCD convencional 16×2 (aplicado a versão inicial dos protótipos), baseado no controlador HD44780, foi conectado à placa Arduino Uno R3 utilizando interface paralela em modo de 4 bits. As conexões foram realizadas da seguinte forma: o pino 1 (VSS) foi conectado ao GND da placa; o pino 2 (VDD) ao 5 V; e o pino 3 (VO), responsável pelo contraste, foi conectado ao terminal central de um potenciômetro de 10 kΩ, cujas extremidades foram ligadas ao 5 V e ao GND. O pino 4 (RS – Register Select) foi conectado ao pino digital 8 do Arduino; o pino 5 (RW – Read/Write) foi diretamente conectado ao GND, mantendo o display em modo de escrita; e o pino 6 (E – Enable) foi conectado ao pino digital 7. Os pinos de dados 11 (D4), 12 (D5), 13 (D6) e 14 (D7) foram conectados, respectivamente, aos pinos digitais 6, 5, 4 e 3 do Arduino. Os pinos 15 (A – ânodo do backlight) e 16 (K – cátodo do backlight) foram conectados

ao 5 V (com resistor limitador de corrente de aproximadamente 220  $\Omega$ ) e ao GND, respectivamente.

No caso do display LCD 16×2 com interface I2C (aplicado às versões finais), foi utilizado módulo adaptador acoplado à parte posterior do display, responsável por converter o protocolo paralelo para comunicação serial I2C. As conexões foram simplificadas para quatro terminais: o pino GND do módulo foi conectado ao GND do Arduino; o pino VCC foi conectado ao 5 V; o pino SDA (Serial Data) foi conectado ao pino A4 da placa; e o pino SCL (Serial Clock) foi conectado ao pino A5. Internamente, o módulo I2C gerencia os sinais RS, E e D4–D7 do display, dispensando a ligação individual desses pinos ao microcontrolador.

#### **4.7.2 Montagem do sensor de temperatura**

O sensor digital DS18B20 foi utilizado para monitoramento da temperatura da água, operando por meio do protocolo OneWire.

O terminal VCC (fio vermelho) do sensor foi conectado ao pino 5V da placa Arduino, e o terminal GND (fio preto) ao pino GND. O pino de dados (fio amarelo) foi conectado ao pino digital 3 da placa. Entre o pino de dados e o terminal 5V foi instalado resistor de 10k $\Omega$ , atuando como pull-up para estabilização do barramento de comunicação.

#### **4.7.3 Montagem do sensor de condutividade elétrica (TDS)**

O sensor Gravity: Analog TDS Sensor / Meter for Arduino (SEN0244) foi empregado para estimativa da concentração de sólidos dissolvidos totais na água.

O módulo transmissor do sensor foi alimentado com 5V e GND provenientes da placa Arduino. A saída analógica do módulo foi conectada ao pino A1, permitindo leitura via conversor analógico-digital interno de 10 bits.

#### **4.7.4 Montagem do sensor de vazão**

O sensor de fluxo de água do tipo efeito Hall foi utilizado para mensuração da vazão volumétrica.

O terminal VCC (fio vermelho) foi conectado ao pino 5V, e o terminal GND (fio preto) ao GND da placa Arduino. O fio de saída de pulsos (fio amarelo) foi conectado ao pino digital 2, configurado como interrupção externa (Interrupt 0). A contagem de pulsos foi realizada via rotina de interrupção no microcontrolador, permitindo o cálculo da vazão em litros por minuto conforme constante de calibração do fabricante.

#### 4.7.5 Montagem do sensor de pH

O a saída analógica (pino Po) do módulo conversor foi conectado ao pino analógico A0 da placa Arduino. O terminal VCC foi alimentado com 5V e o GND conectado ao pino GND da placa. A saída analógica do módulo foi utilizada para leitura da tensão correspondente ao potencial hidrogeniônico da solução.

#### 4.7.6 Montagem do sensor de turbidez

O sensor de turbidez foi conectado ao módulo conversor sendo os fios vermelho, azul e amarelo do sensor conectados aos pinos 1, 2 e 3, respectivamente, do módulo conversor.

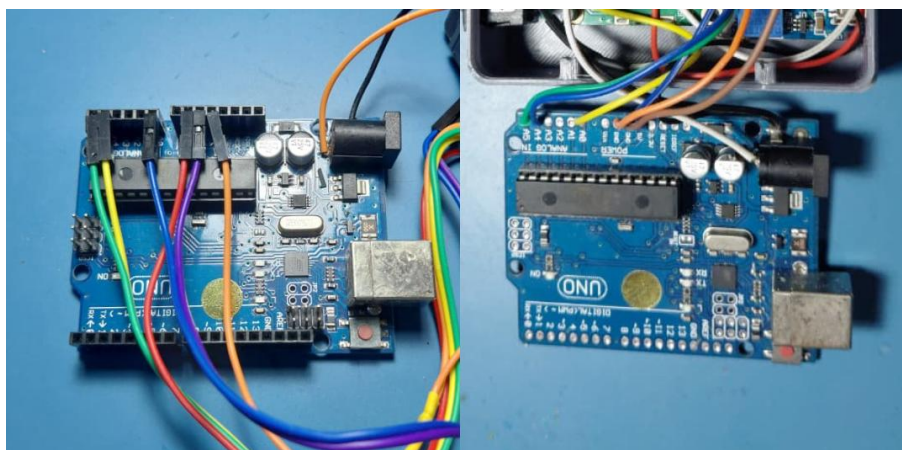
O pino A do módulo conversor foi conectado ao pino analógico A2 da placa Arduino, sendo alimentado com V conectado ao 5V da placa e G ao GND da placa. A leitura da tensão analógica foi convertida em unidades de turbidez (NTU) por meio de equação polinomial recomendada na documentação técnica do fabricante.

Para reduzir ruídos nas leituras, foi aplicada média de múltiplas amostras consecutivas no software de aquisição.

#### 4.8 Mudança estrutural para acondicionamento

Para o adequado encaixe na case impressa, foram realizadas alterações estruturais na placa Arduino. Os conectores laterais (headers) foram removidos e as conexões passaram de ser por encaixe para ser por solda direta na placa, reduzindo o volume físico da placa e permitindo melhor organização interna dos componentes, conforme ilustrado na Figura 3.

Figura 3 – comparação estrutural da placa Arduino antes e depois das alterações.



Fonte: Fotografia de autoria própria (2026).

#### 4.9 Calibração e validação dos sensores

A calibração e a verificação dos sensores utilizados nos protótipos foram realizadas conforme recomendações técnicas dos fabricantes e metodologias descritas na literatura, respeitando as particularidades de cada parâmetro monitorado.

O sensor de pH passou por calibração formal utilizando soluções tampão padrão de pH 4,0, pH 7,0 e pH 10,0, garantindo maior precisão e confiabilidade das leituras.

O sensor de temperatura DS18B20 foi submetido a um processo de verificação por comparação direta com um termômetro de mercúrio e um termômetro digital acoplado a multímetro, ambos utilizados como referência no laboratório, visando avaliar a concordância entre as leituras obtidas.

O sensor de vazão foi verificado por meio da passagem controlada de volumes conhecidos de água, relacionando-se o número de pulsos gerados pelo sensor ao volume real medido, permitindo uma estimativa do desempenho do dispositivo.

Os sensores de turbidez e de condutividade elétrica (TDS) não passaram por calibração formal, sendo avaliados apenas de forma preliminar, por comparação com sensores de laboratório disponíveis, com o objetivo de verificar seu comportamento geral e limitações operacionais.

#### 4.10 Validação experimental

A validação experimental dos protótipos consistiu na comparação das leituras obtidas pelos dispositivos desenvolvidos com equipamentos comerciais utilizados rotineiramente no laboratório. Foram avaliados aspectos como precisão, estabilidade e repetibilidade das medições, em condições controladas que simulam o ambiente de cultivo do Camarão-da-Amazônia.

Os resultados obtidos a partir dessa etapa permitiram avaliar o desempenho dos protótipos como sistemas integrados de monitoramento, fornecendo subsídios para a análise de sua aplicabilidade prática, discutida na seção de Resultados e Discussão.

#### 4.11 Análise de custos

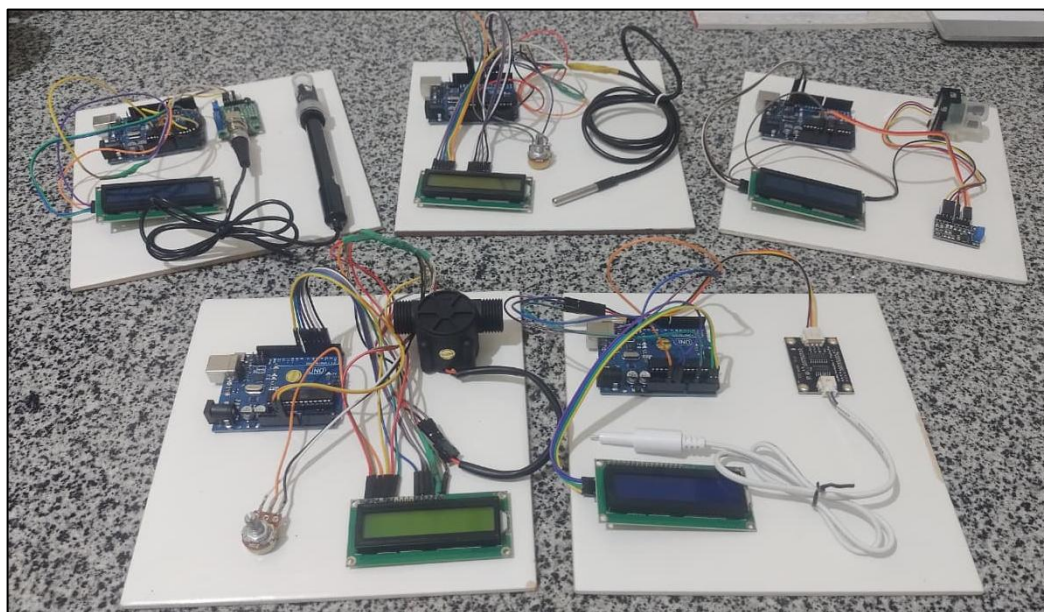
Os custos de produção dos protótipos foram estimados com base nos valores de mercado dos componentes eletrônicos, sensores, materiais de impressão 3D e acessórios utilizados. Essa

análise permitiu avaliar a viabilidade técnica e econômica da replicação dos equipamentos em sistemas de cultivo de pequena e média escala.

## 5 RESULTADOS E DISCUSSÃO

### 5.1 Protótipos em fase inicial de desenvolvimento

Figura 4 – Protótipos em fase inicial contendo sensores de pH, temperatura, turbidez, vazão e condutividade montados em bancada.



Fonte: Fotografias de autoria própria (2024).

## 5.2 Protótipos finais

Figura 5 – Protótipos finalizados para monitoramento de pH e temperatura, respectivamente, acondicionados em invólucros produzidos por impressão 3D de ABS.



Fonte: Fotografias de autoria própria (2026).

## 5.3 Evolução e otimização dos protótipos

Durante o desenvolvimento dos protótipos, foram realizadas modificações estruturais comuns a todos os dispositivos, com o objetivo de otimizar a montagem, reduzir a complexidade dos circuitos e aumentar a portabilidade dos equipamentos.

Na versão inicial, os protótipos utilizavam display LCD convencional com backlight verde, o que demandava a utilização de múltiplos cabos, resistores e potenciômetro para ajuste de contraste, totalizando aproximadamente 12 cabos advindos do display interconectados apenas para o funcionamento dele. Essa configuração aumentava a complexidade da montagem e a suscetibilidade a falhas de contato.

A adoção do display LCD com interface I2C e backlight azul representou um avanço significativo, reduzindo o número de conexões necessárias de 12 para apenas 4 cabos, além de eliminar a necessidade de componentes auxiliares. Essa modificação resultou em maior estabilidade das leituras, organização do circuito, facilidade de replicação dos protótipos, facilidade de visualização do display e aparência mais moderna.

Posteriormente, os protótipos passaram a incorporar sistema de alimentação por bateria recarregável, associado a módulo de conversão de tensão (step-up) com entrada USB-C para carregamento, permitindo operação autônoma e ampliando o potencial de uso em ambientes externos e em sistemas de cultivo.

Durante o desenvolvimento dos protótipos, também foram realizadas alterações nos invólucros impressos em 3D, passando de PLA para ABS. Essa mudança buscou aumentar a resistência mecânica, a durabilidade e a estabilidade térmica dos equipamentos, melhorando a robustez dos dispositivos e sua adaptabilidade a diferentes condições de cultivo.

Por fim, foi elaborada uma proposta de otimização adicional, ainda não implementada fisicamente, que consiste na substituição da placa Arduino Uno R3 pela Arduino Nano V3. Essa modificação, apresentada em nível conceitual, visa reduzir dimensões, consumo energético e custos, mantendo a funcionalidade dos sistemas desenvolvidos.

Os detalhes técnicos da evolução estrutural dos protótipos de pH e temperatura, da implementação do sistema de alimentação por bateria e da proposta conceitual de substituição da plataforma Arduino encontram-se descritos nos Apêndices N, O e P.

Documentação fotográfica adicional, incluindo diferentes ângulos e detalhamento estrutural dos dispositivos, está disponível no Apêndice T.

#### 5.4 Protótipo de pH

O protótipo desenvolvido para o monitoramento do pH apresentou leituras estáveis e confiáveis após a calibração, com variação máxima de  $\pm 0,1$  unidades de pH ( $\pm 0,2$  unidades de pH em soluções ácidas como pH 4,00) em comparação ao equipamento comercial de referência. A estabilidade das medições indica que o dispositivo é adequado para o acompanhamento contínuo desse parâmetro em sistemas de cultivo. A descrição dos componentes eletrônicos utilizados e o código de programação correspondente ao protótipo de pH, em suas versões inicial e final, encontram-se apresentados nos Apêndices A, G e H, respectivamente.

#### 5.5 Protótipo de temperatura

O sensor de temperatura DS18B20 a prova d'água apresentou elevada precisão, com variação máxima de  $\pm 0,1$  °C quando comparado aos termômetros utilizados como referência. Além da leitura quase instantânea, o sistema permitiu o registro das temperaturas mínima e máxima, fornecendo informações relevantes para o manejo térmico do cultivo. A descrição dos componentes eletrônicos utilizados e o código de programação correspondente ao protótipo de temperatura, em suas versões inicial e final, encontram-se apresentados nos Apêndices B, I e J, respectivamente.

### 5.6 Protótipo de vazão

O protótipo de vazão apresentou desempenho satisfatório, com variações aproximadas de  $\pm 20$  ml a cada 5l em testes com volumes controlados em fluxo laminar não-turbulento. O monitoramento contínuo da vazão é fundamental para garantir a renovação da água e a remoção de metabólitos, especialmente em sistemas de recirculação. Porém, o protótipo foi descontinuado, visto que a aplicação na qual ele seria inserido teria a passagem de bolhas de ar junto da passagem de água (devido ao *airlift*), o que tornaria a leitura imprecisa e inviabilizaria o uso de protótipo. A descrição dos componentes eletrônicos utilizados e o código de programação correspondente ao protótipo de vazão em sua versão inicial encontram-se apresentados nos Apêndices C e K, respectivamente.

### 5.7 Protótipos de turbidez e condutividade elétrica (TDS)

Os protótipos de turbidez e TDS apresentaram limitações quanto à precisão e sensibilidade. No sensor de turbidez, observou-se baixa diferenciação entre amostras com características distintas, enquanto o sensor de TDS apresentou dificuldades na conversão confiável dos valores de condutividade elétrica, visto que em nosso laboratório utilizamos água com variações de salinidade, e isso parece ter interferido no funcionamento do sensor, depois constatado que tem aplicação voltada especificamente a água doce. Mesmo ao trocar o sensor e a placa Arduino, as leituras continuavam instáveis, apresentando possível falha no código-fonte. A descrição dos componentes eletrônicos utilizados e do código de programação correspondentes ao protótipo de turbidez, em sua versão inicial, encontram-se apresentados nos Apêndices D e L, enquanto os referentes ao protótipo de condutividade elétrica (TDS) estão apresentados nos Apêndices E e M, respectivamente.

### 5.8 Viabilidade técnica e econômica

A análise de custos demonstrou que os protótipos desenvolvidos apresentam valores significativamente inferiores quando comparados a equipamentos comerciais disponíveis no mercado, mantendo desempenho compatível com as exigências de aplicações experimentais e laboratoriais. Esse resultado indica viabilidade técnica para monitoramento contínuo de parâmetros físico-químicos em sistemas aquícolas e experimentais.

Entretanto, a viabilidade econômica não depende exclusivamente do custo reduzido de fabricação, mas também da existência de demanda de mercado. Nesse contexto, identificam-se dois segmentos potenciais: (i) instituições de pesquisa acadêmica e aplicada voltadas a

sistemas aquáticos e (ii) empreendimentos de cultivo comercial, especialmente na carcinicultura. No âmbito acadêmico, há demanda por soluções de baixo custo que possibilitem experimentação, replicabilidade e customização dos sistemas de monitoramento, sobretudo em laboratórios com orçamento limitado. Já no setor produtivo, conforme discutido na Introdução, o monitoramento contínuo da qualidade da água constitui ferramenta estratégica para prevenção de perdas, otimização de manejo e aumento da produtividade, embora o alto custo de equipamentos comerciais ainda represente uma barreira para pequenos e médios produtores.

Adicionalmente, deve-se considerar que a adoção em larga escala dificilmente ocorreria por meio da autoconstrução dos dispositivos, uma vez que a maioria dos usuários potenciais não possui conhecimento técnico em eletrônica, programação ou manutenção de sistemas embarcados. Dessa forma, observa-se potencial para a oferta da tecnologia na forma de produto finalizado, acompanhado de serviços de instalação, calibração e manutenção periódica. Nesse modelo, incidiriam custos adicionais relacionados a mão de obra especializada, garantia, suporte técnico, tributos e logística de distribuição, os quais impactariam o preço final. Ainda assim, a estrutura simplificada e o baixo custo dos componentes sugerem que, mesmo com tais acréscimos, o valor final poderia permanecer competitivo em relação às soluções comerciais consolidadas.

Além disso, a natureza modular dos sistemas e o uso de componentes eletrônicos de ampla disponibilidade comercial favorecem a manutenção simplificada, permitindo substituições rápidas e de baixo custo, sem necessidade de assistência técnica especializada.

Portanto, os resultados indicam não apenas viabilidade técnica, mas também potencial econômico condicionado à estruturação de um modelo de negócio adequado, capaz de atender tanto ao setor acadêmico quanto ao produtivo.

Tabela 1 - Estimativa de custos dos protótipos desenvolvidos.

<b>Protótipo</b>	<b>Versão</b>	<b>Componentes</b>	<b>Status</b>	<b>Custo estimado (R\$)</b>
<b>pH</b>	Final	Arduino*, sensor pH, LCD I2C, bateria, módulo carregador, interruptor, conector de bateria, case impresso em 3D, cabos de conexão.	Funcional	394,00
<b>Temperatura</b>	Final	Arduino*, sensor DS18B20, LCD I2C, resistor 10k $\Omega$ , bateria, módulo carregador, interruptor, conector de bateria, case impresso em 3D, cabos de conexão.	Funcional	193,20
<b>Vazão</b>	Inicial	Arduino*, sensor de fluxo, LCD, potenciômetro 10k $\Omega$ , resistor 330 $\Omega$ , protoboard, cabos.	Descontinuado**	155,20
<b>Turbidez</b>	Inicial	Arduino*, sensor turbidez, LCD, potenciômetro 10k $\Omega$ , resistor 330 $\Omega$ , protoboard, cabos.	Descontinuado**	329,20
<b>Condutividade (TDS)</b>	Inicial	Arduino*, sensor TDS, potenciômetro 10k $\Omega$ , resistor 330 $\Omega$ , protoboard, cabos.	Descontinuado**	244,20

\*Todos os protótipos utilizam Arduino Uno R3

\*\*Os protótipos de vazão, turbidez e condutividade elétrica apresentaram custos relativamente baixos na fase inicial; contudo, limitações técnicas relacionadas à precisão e estabilidade dos sensores inviabilizaram sua continuidade para aplicação prática, resultando na sua descontinuação, por isso apresentam menos componentes na tabela, em comparação com os protótipos de pH e temperatura.

Fonte: Elaboração própria (2026).

Os valores apresentados na Tabela 1 correspondem a estimativas globais por protótipo.

O detalhamento dos custos por componente, com valores unitários aproximados, encontra-se descrito no Apêndice Q.

Tabela 2 – Comparativo de alguns protótipos com produtos comerciais.

<b>Parâmetro</b>	<b>Custo (R\$)</b>	<b>Produto Comercial (Exemplo)</b>	<b>Custo Comercial (R\$)</b>	<b>Diferença de Custo</b>
<b>pH e Temperatura*</b>	587,20	Medidor de pH/temperatura comercial Extech EXTPH300	~4.000,00	581%
<b>Turbidez</b>	329,20	Turbidímetro portátil profissional Hanna (HI 93102)	~8.000,00	2.330%

\*Considera-se o os protótipos individuais somados.

Fonte: Elaboração própria, com base em valores médios de mercado (2026).

Conforme demonstrado, os protótipos desenvolvidos apresentaram custo significativamente inferior aos equipamentos comerciais, com diferenças que variam de 581% a 2.330%. O sistema de pH e temperatura mostrou-se funcional e viável frente ao medidor profissional Extech EXTPH300, ainda que este ofereça maior robustez, precisão certificada e recursos avançados de calibração.

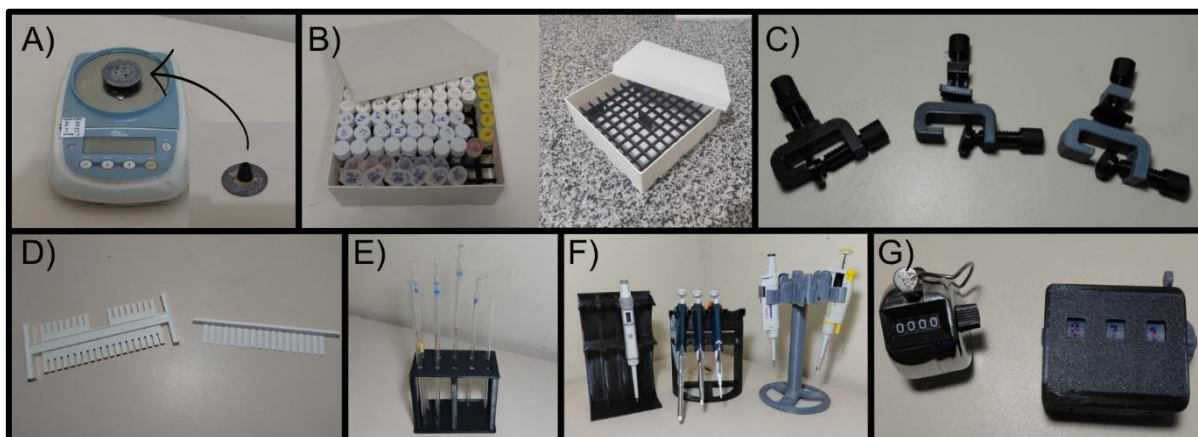
Por outro lado, o protótipo de turbidez não atingiu funcionamento satisfatório, sendo descontinuado durante o desenvolvimento. Mesmo assim, a comparação com o turbidímetro portátil profissional Hanna HI 93102 evidencia a grande diferença de custo entre soluções experimentais e equipamentos consolidados no mercado, reforçando a complexidade técnica envolvida na medição precisa desse parâmetro e a necessidade de aprimoramentos futuros.

### 5.9 Aplicações complementares da impressão 3D no laboratório

Além da confecção dos invólucros e suportes dos protótipos desenvolvidos, a utilização da impressora 3D permitiu a produção de outros itens funcionais destinados ao laboratório, como suportes, organizadores e acessórios personalizados. Para alguns itens, foram feitos testes onde optou-se pelo uso do ABS em substituição ao PLA, devido à maior resistência mecânica, durabilidade e estabilidade dimensional desse material, características importantes para equipamentos que podem ser manipulados com frequência ou utilizados em condições variadas de temperatura e umidade. Esses materiais contribuíram para a otimização do espaço,

organização dos equipamentos e redução de custos com a aquisição de materiais comerciais, ampliando os benefícios do uso da tecnologia para além do escopo inicial do projeto.

Figura 6 – Itens produzidos por impressão 3D para uso laboratorial.



Fonte: Fotografias de autoria própria; modelos 3D de autoria própria (A e D) e de terceiros (B, C, E, F e G), obtidos em repositórios públicos online (2026).

A Figura 6 reúne os principais componentes confeccionados por manufatura aditiva para utilização no laboratório: A) peça de reposição para balança de precisão; B) caixa para microtubos; C) grampos para fixação das mangueiras de aeração nos tanques de cultivo; D) pente para gel de agarose; E) suporte para termômetro de mercúrio e pipetas de vidro; F) suporte para diversos modelos de pipeta; G) contador manual.

## 6 CONCLUSÃO

O desenvolvimento e a validação de protótipos de baixo custo baseados na plataforma Arduino demonstraram ser uma alternativa tecnicamente viável e cientificamente consistente para o monitoramento de parâmetros físico-químicos da água no cultivo do *M. amazonicum*. Os resultados obtidos evidenciam que a utilização de tecnologias acessíveis pode contribuir significativamente para a melhoria do manejo aquícola, especialmente em contextos de produção de pequena e média escala.

Ressalta-se que a validação dos dispositivos foi conduzida em ambiente laboratorial real, sob condições associadas à larvicultura de *M. amazonicum*, utilizando equipamentos comerciais de referência disponíveis nos laboratórios LAQUA e BioDATTA. Esse procedimento conferiu maior robustez à análise comparativa e à avaliação da aplicabilidade prática dos protótipos desenvolvidos.

Os protótipos destinados à medição de pH e temperatura apresentaram elevada precisão e estabilidade, com valores compatíveis aos registrados por equipamentos comerciais utilizados como referência. Esses parâmetros são reconhecidamente críticos para o metabolismo, crescimento e sobrevivência de *M. amazonicum*, reforçando a relevância prática dos dispositivos desenvolvidos para o acompanhamento contínuo das condições ambientais de cultivo.

O protótipo de vazão demonstrou desempenho satisfatório, permitindo o monitoramento da renovação hídrica, fator essencial para a manutenção da qualidade da água em sistemas de recirculação. Porém, em usos no qual a vazão de água não é contínua, a passagem de ar pode influenciar na leitura.

Por outro lado, os protótipos de turbidez e condutividade elétrica (TDS) apresentaram limitações quanto à precisão e sensibilidade, evidenciando desafios inerentes ao uso de sensores de baixo custo para parâmetros mais complexos. Tais limitações não invalidam o estudo, mas ressaltam a importância de calibrações mais robustas e do aprimoramento dos componentes empregados, configurando-se como resultados relevantes do ponto de vista científico e tecnológico.

A estimativa detalhada dos custos por componente evidenciou que os sistemas desenvolvidos apresentam redução de investimento em comparação a equipamentos comerciais equivalentes, mantendo desempenho adequado para aplicações experimentais, o que reforça sua viabilidade econômica. Dessa forma, a adoção dessas tecnologias pode ampliar o acesso ao monitoramento ambiental, favorecendo a sustentabilidade da carcinicultura amazônica, além de outras formas de cultivo, e fortalecendo a transferência de tecnologia do meio acadêmico para o setor produtivo.

Ainda vale destacar a facilidade de manutenção dos protótipos desenvolvidos, uma vez que seus componentes são amplamente disponíveis no mercado nacional, apresentam baixo custo unitário e podem ser facilmente substituídos em caso de falha. Diferentemente de equipamentos comerciais de arquitetura fechada, os sistemas desenvolvidos permitem diagnóstico simples, reparo local e substituição individual de módulos, aumentando sua vida útil e reduzindo custos operacionais a longo prazo.

A adoção da impressão 3D, com a escolha do ABS para os invólucros dos protótipos, demonstrou-se estratégica para o desenvolvimento de equipamentos robustos, duráveis e de fácil manuseio, contribuindo não apenas para a funcionalidade dos dispositivos, mas também para a otimização do espaço e a redução de custos laboratoriais.

O presente estudo evidencia então que a integração entre conhecimento biológico aplicado, eletrônica embarcada e manufatura aditiva pode gerar soluções tecnológicas acessíveis e adaptáveis às realidades regionais. Mais do que o desenvolvimento de dispositivos específicos, o trabalho demonstra a viabilidade de construção de infraestrutura tecnológica própria dentro do ambiente acadêmico, fortalecendo a autonomia científica e a transferência de tecnologia para o setor produtivo.

## **7 PERSPECTIVAS FUTURAS**

Como perspectivas para trabalhos futuros, recomenda-se o aprimoramento dos protótipos de turbidez e condutividade elétrica, que serão mais estudados e trabalhados para que se valide sua funcionalidade, bem como a implementação de protocolos de calibração mais detalhados e específicos para ambientes aquícolas e a consideração de substituição dos sensores utilizados, caso necessário, ainda visando o baixo custo.

Destaca-se também a substituição potencial da placa Arduino Uno R3 pela Arduino Nano V3, visando à miniaturização dos protótipos, à redução do consumo energético com consequente aumento da autonomia, e à maior integração dos sistemas desenvolvidos com aplicações portáteis, mantendo-se as funcionalidades de monitoramento propostas neste estudo.

Além disso, foi desenvolvido, em nível experimental, um sistema de registro automatizado dos dados obtidos pelos sensores, utilizando um módulo de relógio de tempo real (RTC – Real Time Clock) e um módulo de armazenamento em cartão de memória SD. Essa implementação permite o salvamento contínuo das medições em arquivos digitais contendo o valor do parâmetro monitorado, bem como a data e o horário exatos (hora e minuto) de cada leitura. Esse recurso possibilita a criação de um histórico detalhado das condições ambientais, permitindo análises retrospectivas, identificação de padrões e correlação com eventos ocorridos no sistema de cultivo.

Como evolução dessa proposta, também foi desenvolvido um protótipo conceitual que integra, além do módulo RTC e do módulo de armazenamento em cartão SD, um módulo de comunicação sem fio (WiFi), permitindo simultaneamente o armazenamento local dos dados e sua transmissão remota em tempo real. Essa abordagem representa um passo importante para a implementação de sistemas baseados em Internet das Coisas (IoT), possibilitando o monitoramento remoto, o acesso às informações por meio de aplicações web ou dispositivos móveis e a criação de plataformas centralizadas de gerenciamento ambiental.

Embora a estrutura física e o sistema embarcado para coleta, registro e transmissão dos dados tenham sido implementados e validados em nível funcional, a etapa de desenvolvimento da infraestrutura digital completa, incluindo servidor, interface web ou aplicativo dedicado para recepção, armazenamento e visualização dos dados, ainda se encontra em fase de planejamento. A implementação futura dessa infraestrutura permitirá a consolidação de um sistema integrado de monitoramento remoto, ampliando significativamente o potencial de aplicação dos protótipos desenvolvidos.

Essa evolução tecnológica contribuirá para a automatização do monitoramento ambiental, redução da necessidade de intervenção manual e melhoria na tomada de decisões no manejo aquícola, fortalecendo a aplicabilidade prática e o potencial de transferência tecnológica das soluções desenvolvidas. Os códigos-fonte responsáveis pelo registro e transmissão dos dados foram desenvolvidos pelo autor, demonstrando a viabilidade técnica da integração entre sensores ambientais, armazenamento digital e comunicação sem fio em sistemas de baixo custo. Tais códigos estão disponíveis nos apêndices R e S.

Adicionalmente, a validação dos protótipos em ambientes produtivos reais, como viveiros escavados e sistemas de recirculação em escala comercial permitirá avaliar o desempenho dos equipamentos sob condições ambientais variáveis, ampliando sua confiabilidade e aplicabilidade prática.

Por fim, a expansão do sistema para o monitoramento de outros parâmetros relevantes à aquicultura, como oxigênio dissolvido e compostos nitrogenados pode contribuir para o desenvolvimento de uma plataforma integrada de monitoramento ambiental, fortalecendo a sustentabilidade e a eficiência da produção de *M. amazonicum*.

## 8 REFERENCIAL BIBLIOGRÁFICO

BOYD, C. E. *Water Quality: An Introduction*. 3. ed. Cham: Springer, 2020.

CARVALHO, H. A. *Larvicultura do camarão-da-amazônia *Macrobrachium amazonicum**. 2004. Dissertação (Mestrado em Aquicultura) – Universidade Estadual Paulista, Jaboticabal, 2004.

CHONG-CARRILLO, O. et al. Larval culture of the Amazon river prawn *Macrobrachium amazonicum*: a review. *Latin American Journal of Aquatic Research*, v. 43, n. 5, p. 819–831, 2015.

COLEZEA, M. et al. Internet of Things: challenges and solutions for agriculture. *International Journal of Advanced Computer Science and Applications*, v. 9, n. 6, p. 19–27, 2018.

CREALITY. *Creality Print*. Versão 7.0. Software de fatiamento para impressão 3D. Shenzhen: Creality, 2023.

DFROBOT. *Gravity: Analog TDS Sensor / Meter for Arduino (SEN0244)*. DFRobot Wiki. Disponível em:

[https://wiki.dfrobot.com/Gravity\\_Analog\\_TDS\\_Sensor\\_Meter\\_For\\_Arduino\\_SKU\\_SEN0244](https://wiki.dfrobot.com/Gravity_Analog_TDS_Sensor_Meter_For_Arduino_SKU_SEN0244). Acesso em: 20 jan. 2026.

EMBRAPA AMAPÁ. Qualidade de água no cultivo de camarão-da-Amazônia (*Macrobrachium amazonicum*) em sistema fechado de recirculação. Macapá: Embrapa Amapá, 2016.

KUBITZA, F. *Qualidade da água no cultivo de peixes e camarões*. Jundiaí: Acqua Supre, 2011.

LIMA, J. D. F.; BASTOS, A. M. Qualidade de água e produtividade de camarão e de alface em sistema de aquaponia com *Macrobrachium amazonicum*. Macapá: Embrapa Amapá, 2019. (Boletim de Pesquisa e Desenvolvimento).

MACIEL, C. R.; VALENTI, W. C. Biology, fisheries, and aquaculture of the Amazon River prawn *Macrobrachium amazonicum*: a review. *Nauplius*, v. 17, n. 2, p. 61–79, 2009.

MARQUES, A. C. et al. Importância socioeconômica do camarão-da-amazônia na Região Norte do Brasil. *Boletim do Instituto de Pesca*, v. 46, n. 1, p. 1–12, 2020.

MEDEIROS JUNIOR, E. F. Qualidade da água na larvicultura comercial do camarão de água doce *Macrobrachium amazonicum*. Curitiba: Editora Científica Digital, 2024.

MORAES-RIODADES, P. M. C. et al. Effect of increase in temperature on survival and growth of *Macrobrachium amazonicum*. *Aquatic Living Resources*, v. 31, 2018.

MORAES-VALENTI, P. M. C.; VALENTI, W. C. Culture of the Amazon River prawn *Macrobrachium amazonicum*. In: NEW, M. B. et al. (eds.). *Freshwater prawns: biology and farming*. Oxford: Wiley-Blackwell, 2010. p. 485–501.

ORACLE. O que é Internet das Coisas (IoT)?. Disponível em: <https://www.oracle.com/br/internet-of-things/what-is-iot/>. Acesso em: 10 jan. 2025.

ROBOCORE. *Primeiros Passos com o DS18B20 (Arduino)*. RoboCore, 23 set. 2019. Disponível em: <https://www.robocore.net/tutoriais/primeiros-passos-ds18b20-arduino>. Acesso em: 23 jan. 2026.

STRAUB, Matheus Gebert. *Sensor de Fluxo de Água para Arduino 1–30 L/min*. UsinaInfo, 15 abr. 2016. Disponível em: <https://www.usinainfo.com.br/blog/sensor-de-fluxo-de-agua-para-arduino-1-30-lmin/>. Acesso em: 23 jan. 2026.

STRAUB, Matheus Gebert. *Sensor de pH Arduino: como calibrar e configurar*. UsinaInfo, 23 maio 2022. Disponível em: <https://www.usinainfo.com.br/blog/sensor-de-ph-arduino-como-calibrar-e-configurar/>. Acesso em: 20 jan. 2026.

STRAUB, Matheus Gebert. *Sensor de Turbidez – Projeto de Leitura da Qualidade da Água*. UsinaInfo, 12 fev. 2020. Disponível em: <https://www.usinainfo.com.br/blog/sensor-de-turbidez-projeto-de-leitura-da-qualidade-da-agua/>. Acesso em: 20 jan. 2026.

TIMMONS, M. B.; EBELING, J. M. *Recirculating Aquaculture*. 2. ed. Cayuga Aqua Ventures, 2010.

VALENTI, W. C. et al. Aquaculture in Brazil: past, present and future. *Aquaculture Reports*, v. 8, p. 1–14, 2018.

YODER, M. *Aquaculture water quality management*. Ames: Wiley-Blackwell, 2008.

**9 APÊNDICE A – Lista de componentes do protótipo de pH (versão final)**

- Bateria 9V Recarregável 450mAh – Flex
- Cabos (jumpers) para conexões
- Conector Clip de Bateria 9V – Vertical
- Display LCD 16x2 I2C com Fundo Azul
- Interruptor Liga/Desliga tipo Gangorra 10A/120
- Módulo step-up Mt3608 com entrada USB-C
- Placa Arduino Uno R3
- Sensor de pH Arduino + Módulo de Leitura BNC PH4502C

**10 APÊNDICE B – Lista de componentes do protótipo de temperatura (versão final)**

- Bateria 9V Recarregável 450mAh – Flex
- Cabos (jumpers) para conexões
- Conector Clip de Bateria 9V – Vertical
- Display LCD 16x2 I2C com Fundo Azul
- Interruptor Liga/Desliga tipo Gangorra 10A/120
- Módulo step-up Mt3608 com entrada USB-C
- Placa Arduino Uno R3
- Resistor 10k $\Omega$  (resistor de pull-up na linha de dados do sensor DS18B20)
- Sensor de Temperatura DS18B20 à Prova D'Água

**11 APÊNDICE C – Lista de componentes do protótipo de vazão (fase inicial)**

- Cabos (jumpers) para conexões
- Display LCD 16x2 com fundo verde
- Placa Arduino Uno R3
- Potenciômetro linear 10K com eixo estriado
- Resistor 330  $\Omega$ , 1/4 W
- Sensor de Fluxo de Água YF-S201 G1/2 1-30 l/min

**12 APÊNDICE D – Lista de componentes do protótipo de turbidez (fase inicial)**

- Cabos (jumpers) para conexões
- Display LCD 16x2 com fundo verde

- Placa Arduino Uno R3
- Potenciômetro linear 10K com eixo estriado
- Resistor 330  $\Omega$ , 1/4 W
- Sensor de Turbidez ST100 + Módulo de Leitura – ORIGINAL

### 13 APÊNDICE E – Lista de componentes do protótipo de condutividade elétrica – TDS (fase inicial)

- Cabos (jumpers) para conexões
- Display LCD 16x2 com fundo verde
- Medidor de TDS Sensor de Condutividade da Água Analógico TDS Meter V1.0
- Placa Arduino Uno R3
- Potenciômetro linear 10K com eixo estriado
- Resistor 330  $\Omega$ , 1/4 W

### 14 APÊNDICE F – Código-fonte para calibração inicial do protótipo de pH

O presente apêndice apresenta o código-fonte desenvolvido em plataforma Arduino, utilizado exclusivamente durante a etapa de montagem do protótipo de pH, com a finalidade de realizar a calibração inicial do módulo sensor em soluções tampão de referência.

```
// Código adaptado de:
// "Sensor de pH Arduino: como calibrar e configurar"
// por Matheus Gebert Straub
// UsinaInfo (23/05/2022)
// Disponível em: https://www.usinainfo.com.br/blog/sensor-de-ph-arduino-como-calibrar-e-configurar/
// Acesso em: 20 jan. 2026

int buf[10];

void setup() {
  Serial.begin(9600);
  Serial.println("Kelvin-calibragem pH");
  delay(500);
}

void loop() {
  for (int i = 0; i < 10; i++) { // 11 amostras
    buf[i] = analogRead(A0); // Ler o sensor PH
    delay(10);
  }
}
```

```

int valorMedio = 0;
for (int i = 2; i < 8; i++) { // Realiza o valor médio utilizando 6 amostras
    valorMedio += buf[i];
}

float tensao = (valorMedio * 5.0) / 1024.0 / 6;
Serial.println(tensao);
delay(250);
}

```

## 15 APÊNDICE G – Código-fonte do protótipo de pH (versão inicial em protoboard com LCD convencional)

Apresenta o código de programação utilizado na primeira versão do protótipo de pH, implementada em protoboard e com display LCD convencional, empregado nos testes iniciais de leitura e funcionamento do sensor.

```

// Código adaptado de:
// "Sensor de pH Arduino: como calibrar e configurar"
// por Matheus Gebert Straub
// UsinaInfo (23/05/2022)
// Disponível em: https://www.usinainfo.com.br/blog/sensor-de-ph-arduino-como-calibrar-e-configurar/
// Acesso em: 20 jan. 2026

//Calibrando com tampão pH 7 e 4:
// - Insira o valor obtido coma sonda no tampão de pH 7 em "calibracao_ph7",
utilize ponto para separação no ligar da vírgula.

// - Insira o valor obtido coma sonda no tampão de pH 4 em "calibracao_ph4",
utilize ponto para separação no ligar da vírgula.

// - Pronto, basta utilizar o sensor em soluções neutras e acídas.

// Calibrando com tampão pH 7 e 10:
// - Insira o valor obtido coma sonda no tampão de pH 7 em "calibracao_ph7",
utilize ponto para separação no ligar da vírgula.

// - Insira o valor obtido coma sonda no tampão de pH 10 em "calibracao_ph10",
utilize ponto para separação no ligar da vírgula.

// - Mude o valor da variavel "UTILIZAR_PH_10" de "false" para "true".

// - Pronto, basta utilizar o sensor em soluções neutras e alcalinas.

#include <LiquidCrystal.h>

// RS, E, D4, D5, D6, D7
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

float calibracao_ph4 = 3.15; // Tensão em pH 4
float calibracao_ph7 = 2.62; // Tensão em pH 7

```

```

float calibracao_ph10 = 2.12; // Tensão em pH 10

#define UTILIZAR_PH_10 true // true = pH 7 e 10 | false = pH 7 e 4

float m;
float b;
int buf[10];

void setup() {
  Serial.begin(9600);

  if (calibracao_ph7 == 0 && calibracao_ph4 == 0 && calibracao_ph10 == 0) {
    Serial.println("Erro - Insira os valores de calibração!");
    while (1);
  }

  if (!UTILIZAR_PH_10 && calibracao_ph4 == 0) {
    Serial.println("Erro - Falta calibração pH 4.");
    while (1);
  }

  // Inicializa LCD
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("Kelvin-BioDATTA");
  lcd.setCursor(0, 1);
  lcd.print("pHmetro");
  delay(2000);
  lcd.clear();

  // Cálculo da reta
  if (UTILIZAR_PH_10) {
    m = (7.0 - 10.0) / (calibracao_ph7 - calibracao_ph10);
    b = 10.0 - m * calibracao_ph10;
  } else {
    m = (4.0 - 7.0) / (calibracao_ph4 - calibracao_ph7);
    b = 7.0 - m * calibracao_ph7;
  }
}

void loop() {
  // Coleta de amostras
  for (int i = 0; i < 10; i++) {
    buf[i] = analogRead(A0);
    delay(10);
  }

  // Ordenação
  for (int i = 0; i < 9; i++) {
    for (int j = i + 1; j < 10; j++) {
      if (buf[i] > buf[j]) {
        int temp = buf[i];
        buf[i] = buf[j];
        buf[j] = temp;
      }
    }
  }
}

```

```

int valorMedio = 0;
for (int i = 2; i < 8; i++) {
    valorMedio += buf[i];
}

float tensao = (valorMedio * 5.0) / 1024.0 / 6.0;
float ph = m * tensao + b;

Serial.println(ph);

lcd.setCursor(0, 0);
lcd.print("Valor pH:    ");
lcd.setCursor(10, 0);
lcd.print(ph, 1);

delay(1000);
}

```

## 16 APÊNDICE H – Código-fonte do protótipo de pH (versão final com LCD I2C e alimentação por bateria)

Este apêndice contém o código de programação correspondente à versão final do protótipo de pH, adaptado para uso de display LCD com interface I2C e sistema de alimentação por bateria, refletindo as otimizações realizadas ao longo do desenvolvimento do projeto.

```

// Código adaptado de:
// "Sensor de pH Arduino: como calibrar e configurar"
// por Matheus Gebert Straub
// UsinaInfo (23/05/2022)
// Disponível em: https://www.usinainfo.com.br/blog/sensor-de-ph-arduino-como-calibrar-e-configurar/
// Acesso em: 20 jan. 2026

//Calibrando com tampão pH 7 e 4:
// - Insira o valor obtido coma sonda no tampão de pH 7 em "calibracao_ph7",
utilize ponto para separação no lugar da vírgula.

// - Insira o valor obtido coma sonda no tampão de pH 4 em "calibracao_ph4",
utilize ponto para separação no lugar da vírgula.

// - Pronto, basta utilizar o sensor em soluções neutras e acidas.

// Calibrando com tampão pH 7 e 10:
// - Insira o valor obtido coma sonda no tampão de pH 7 em "calibracao_ph7",
utilize ponto para separação no lugar da vírgula.

// - Insira o valor obtido coma sonda no tampão de pH 10 em "calibracao_ph10",
utilize ponto para separação no lugar da vírgula.

// - Mude o valor da variavel "UTILIZAR_PH_10" de "false" para "true".

// - Pronto, basta utilizar o sensor em soluções neutras e alcalinas.

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

```

```

LiquidCrystal_I2C lcd(0x27, 16, 2);

float calibracao_ph4 = 3.15; // Tensão obtida em solução de calibração pH 4
float calibracao_ph7 = 2.62; // Tensão obtida em solução de calibração pH 7
float calibracao_ph10 = 2.12; // Tensão obtida em solução de calibração pH 10

#define UTILIZAR_PH_10 true // Habilita calibração entre pH 7 e 10,
//                             caso contrário utiliza pH 7 e 4.
float m;
float b;
int buf[10];

void setup() {
  Serial.begin(9600);

  if (calibracao_ph7 == 0 && calibracao_ph4 == 0 && calibracao_ph10 == 0) {
    delay(500);
    Serial.println();
    Serial.println("Erro - Necessário colocar os valores de calibração no
código!");
    while (1)
      ;
  }

  if (!UTILIZAR_PH_10 && calibracao_ph4 == 0 && calibracao_ph10 != 0 &&
calibracao_ph7 != 0) {
    delay(500);
    Serial.println();
    Serial.println("Erro - Você não marcou a opção UTILIZAR_PH_10 como true para
utilizar pH 7 e 10.");
    while (1)
      ;
  }

  lcd.init(); // Inicializa o display
  lcd.begin(16, 2); // Inicializa todos os caracteres
  lcd.backlight(); // Inicializa o backlight
  lcd.setCursor(0, 0); // Posiciona o cursor na posição
  lcd.print("Kelvin-BioDATTA"); // Escreve no monitor serial
  lcd.setCursor(0, 1); // Posiciona o cursor na posição
  lcd.print("pHmetro"); // Escreve no monitor serial
  delay(2000);
  lcd.clear(); // Limpa o conteúdo do display

  if (UTILIZAR_PH_10) {
    m = (7.0 - 10.0) / (calibracao_ph7 - calibracao_ph10);
    b = 10.0 - m * calibracao_ph10;
  } else {
    m = (4.0 - 7.0) / (calibracao_ph4 - calibracao_ph7);
    b = 7.0 - m * calibracao_ph7;
  }
}

void loop() {
  for (int i = 0; i < 10; i++) { // 11 amostras

```

```

    buf[i] = analogRead(A0);    // Ler o sensor PH
    delay(10);
}

for (int i = 0; i < 9; i++) { // Ordena em ordem crescente
  for (int j = i + 1; j < 10; j++) {
    if (buf[i] > buf[j]) {
      int temp = buf[i];
      buf[i] = buf[j];
      buf[j] = temp;
    }
  }
}

int valorMedio = 0;
for (int i = 2; i < 8; i++) { // Realiza o valor médio utilizando 6 amostras
  valorMedio += buf[i];
}

float tensao = (valorMedio * 5.0) / 1024.0 / 6; // Realiza a média e \
                                                transforma o valor \
                                                analógico em volt

float ph = m * tensao + b; // Converte para pH
Serial.println(ph);

lcd.setCursor(0, 0);    // Posiciona o cursor no display
lcd.print("Valor pH: "); // Escreve no display
lcd.setCursor(11, 0);  // Posiciona o cursor no display
lcd.print(ph, 1);      // Escreve o pH com uma casa decimal
delay(1000);          // Aguarda para próxima leitura
}

```

## 17 APÊNDICE I – Código-fonte do protótipo de temperatura (versão inicial em protoboard com LCD convencional)

Inclui o código utilizado na fase inicial de desenvolvimento do protótipo de monitoramento de temperatura da água, montado em protoboard e utilizando display LCD convencional, empregado para validação preliminar do sensor DS18B20.

```

// Código adaptado de:
// "Primeiros Passos com o DS18B20 (Arduino)"
// por RoboCore
// RoboCore (23/09/2019)
// Disponível em: https://www.robocore.net/tutoriais/primeiros-passos-ds18b20-arduino
// Acesso em: 23 jan. 2026

#include <OneWire.h>
#include <DallasTemperature.h>
#include <LiquidCrystal_I2C.h>

// Porta do pino de sinal do DS18B20
#define ONE_WIRE_BUS 3

// Define uma instancia do oneWire para comunicacao com o sensor

```

```

OneWire oneWire(ONE_WIRE_BUS);

// Armazena temperaturas minima e maxima
float tempMin = 999;
float tempMax = 0;

DallasTemperature sensors(&oneWire);
DeviceAddress sensor1;

// Inicializa o LCD
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup(void)
{
  lcd.init();
  lcd.begin(16, 2);           // Inicializa todos os caracteres
  lcd.backlight();
  lcd.setCursor(0, 0);      // Posiciona o cursor na posição
  lcd.print("Kelvin-BioDATTA"); // Escreve no monitor serial
  lcd.setCursor(0, 1);      // Posiciona o cursor na posição
  lcd.print("Termometro"); // Escreve no monitor serial
  delay(2000);
  lcd.clear();              // Limpa o conteúdo do display

  Serial.begin(9600);
  sensors.begin();
  // Localiza e mostra enderecos dos sensores
  Serial.println("Localizando sensores DS18B20...");
  Serial.print("Foram encontrados ");
  Serial.print(sensors.getDeviceCount(), DEC);
  Serial.println(" sensores.");
  if (!sensors.getAddress(sensor1, 0))
    Serial.println("Sensores nao encontrados !");
  // Mostra o endereco do sensor encontrado no barramento
  Serial.print("Endereco sensor: ");
  mostra_endereco_sensor(sensor1);
  Serial.println();
  Serial.println();
  lcd.begin(16, 2);
}

void mostra_endereco_sensor(DeviceAddress deviceAddress)
{
  for (uint8_t i = 0; i < 8; i++)
  {
    // Adiciona zeros se necessário
    if (deviceAddress[i] < 16) Serial.print("0");
    Serial.print(deviceAddress[i], HEX);
  }
}

void loop()
{
  // Le a informacao do sensor
  sensors.requestTemperatures();
  float tempC = sensors.getTempC(sensor1);
  // Atualiza temperaturas minima e maxima

```

```

if (tempC < tempMin)
{
  tempMin = tempC;
}
if (tempC > tempMax)
{
  tempMax = tempC;
}
// Mostra dados no serial monitor
Serial.print("Temp C: ");
Serial.print(tempC);
Serial.print(" Min : ");
Serial.print(tempMin);
Serial.print(" Max : ");
Serial.println(tempMax);

// Mostra dados no LCD
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Temp.:      ");
//Simbolo grau
lcd.write(223);
lcd.print("C");
lcd.setCursor(7,0);
lcd.print(tempC);
lcd.setCursor(0,1);
lcd.print("Min: ");
lcd.setCursor(3,1);
lcd.print(tempMin,1);
lcd.setCursor(8,1);
lcd.print("Max: ");
lcd.setCursor(11,1);
lcd.print(tempMax,1);
delay(3000);
}

```

## 18 APÊNDICE J – Código-fonte do protótipo de temperatura (versão final com LCD

### I2C e alimentação por bateria)

Apresenta o código correspondente à versão final do protótipo de monitoramento de temperatura, com adaptações para display LCD I2C e alimentação por bateria, garantindo maior portabilidade, organização e estabilidade do sistema.

```

// Código adaptado de:
// "Primeiros Passos com o DS18B20 (Arduino)"
// por RoboCore
// RoboCore (23/09/2019)
// Disponível em: https://www.robocore.net/tutoriais/primeiros-passos-ds18b20-arduino
// Acesso em: 23 jan. 2026

#include <OneWire.h>
#include <DallasTemperature.h>
#include <LiquidCrystal_I2C.h>

```

```

#define ONE_WIRE_BUS 3

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
DeviceAddress sensor1;

LiquidCrystal_I2C lcd(0x27, 16, 2);

float tempMin = 999.0;
float tempMax = -999.0;

void setup() {
  Serial.begin(9600);

  // Inicializa LCD I2C
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Kelvin-BioDATTA");
  lcd.setCursor(0, 1);
  lcd.print("Termometro");
  delay(2000);
  lcd.clear();

  // Inicializa sensor
  sensors.begin();

  Serial.println("Localizando sensores DS18B20...");
  Serial.print("Sensores encontrados: ");
  Serial.println(sensors.getDeviceCount());

  if (!sensors.getAddress(sensor1, 0)) {
    Serial.println("Erro: sensor DS18B20 nao encontrado!");
  } else {
    Serial.print("Endereco do sensor: ");
    mostra_endereco_sensor(sensor1);
    Serial.println();
  }
}

void loop() {
  sensors.requestTemperatures();
  float tempC = sensors.getTempC(sensor1);

  // Atualiza min e max
  if (tempC < tempMin) tempMin = tempC;
  if (tempC > tempMax) tempMax = tempC;

  // Serial Monitor
  Serial.print("Temp C: ");
  Serial.print(tempC);
  Serial.print(" | Min: ");
  Serial.print(tempMin);
  Serial.print(" | Max: ");
  Serial.println(tempMax);

  // LCD
  lcd.clear();

```

```

    lcd.setCursor(0, 0);
    lcd.print("Temp: ");
    lcd.print(tempC, 1);
    lcd.write(223); // símbolo °
    lcd.print("C");

    lcd.setCursor(0, 1);
    lcd.print("Min:");
    lcd.print(tempMin, 1);
    lcd.setCursor(9, 1);
    lcd.print("Max:");
    lcd.print(tempMax, 1);

    delay(3000);
}

void mostra_endereco_sensor(DeviceAddress deviceAddress) {
    for (uint8_t i = 0; i < 8; i++) {
        if (deviceAddress[i] < 16) Serial.print("0");
        Serial.print(deviceAddress[i], HEX);
    }
}

```

## 19 APÊNDICE K – Código-fonte do protótipo de vazão (fase inicial)

Este apêndice reúne o código desenvolvido para o protótipo de monitoramento de vazão de água durante a fase inicial do projeto, o qual não foi levado à versão final devido às limitações técnicas observadas nos testes preliminares.

```

// Código adaptado de:
// "Sensor de Fluxo de Água para Arduino 1-30 L/min"
// por Matheus Gebert Straub
// UsinaInfo (15/04/2016)
// Disponível em: https://www.usinainfo.com.br/blog/sensor-de-fluxo-de-agua-para-arduino-1-30-lmin/
// Acesso em: 23 jan. 2026

#include <LiquidCrystal.h>

float vazao; //Variável para armazenar o valor em L/min
float media = 0; //Variável para fazer a média
int contaPulso; //Variável para a quantidade de pulsos
int i = 0; //Variável para segundos
int Min = 00; //Variável para minutos
float Litros = 0; //Variável para Quantidade de agua
float MiliLitros = 0; //Variavel para Conversão

LiquidCrystal lcd(8, 7, 6, 5, 4, 3);

void setup() {
    Serial.begin(9600);

    lcd.begin(16, 2);
    lcd.setCursor( 0, 0);
    lcd.print("Kelvin-BioDATTA");
    lcd.setCursor( 0, 1);

```

```

lcd.print("Sensor de fluxo");
delay(3000);
lcd.clear();

pinMode(2, INPUT);
attachInterrupt(0, inculso, RISING); //Configura o pino 2(Interrupção 0)
interrupção
}

void loop () {
  contaPulso = 0;//Zera a variável
  sei(); //Habilita interrupção
  delay (1000); //Aguarda 1 segundo
  cli(); //Desabilita interrupção

  vazao = contaPulso / 5.5; //Converte para L/min
  media = media + vazao; //Soma a vazão para o calculo da media
  i++;
  lcd.setCursor(0, 0);
  lcd.print(vazao); //Escreve no display o valor da vazão
  lcd.print(" L/min "); //Escreve L/min
  lcd.setCursor(0, 1);
  lcd.print(Min);
  lcd.print(":"); //Escreve :
  lcd.print(i); //Escreve a contagem i (segundos)
  lcd.print("Min "); //Escreve :
  Mililitros = vazao / 60;
  Litros = Litros + Mililitros;
  lcd.print(Litros);
  lcd.print("L ");

  // Neste conjunto de linhas fizemos a média das leituras obtidas a cada 1 minuto
  if (i == 59) {
    Min++;
    lcd.print(Min);

    if (Min >= 60) {
      Min = 0;
    }
    media = media / 60; //faz a média
    Serial.print("\nMedia por minuto = "); //Imprime a frase Media por minuto =
    Serial.print(media); //Imprime o valor da media
    Serial.println(" L/min - "); //Imprime L/min
    media = 0; //Zera a variável media para uma nova contagem
    i = 0; //Zera a variável i para uma nova contagem
  }
}

void inculso () {
  contaPulso++; //Incrementa a variável de pulsos
}

```

## 20 APÊNDICE L – Código-fonte do protótipo de turbidez (fase inicial)

Apresenta o código de programação elaborado para o protótipo de monitoramento de turbidez da água na fase inicial do projeto, desenvolvido em protoboard e não implementado em versão final.

```
// Código adaptado de:
// "Sensor de Turbidez - Projeto de Leitura da Qualidade da Água"
// por Matheus Gebert Straub
// UsinaInfo (12/02/2020)
// Disponível em: https://www.usinainfo.com.br/blog/sensor-de-turbidez-projeto-de-leitura-da-qualidade-da-agua/
// Acesso em: 20 jan. 2026

#include <LiquidCrystal.h>

// Inicializa o Display LCD 16x2 (RS, E, D4, D5, D6, D7)
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

// Define o pino de Leitura do Sensor
int SensorTurbidez = A0;

// Inicia as variáveis
int i;
float voltagem;
float NTU;

void setup() {
  // Inicia o display LCD
  lcd.begin(16, 2);
}

void loop() {
  // Inicia a leitura da voltagem em 0
  voltagem = 0;

  // Realiza a soma dos "i" valores de voltagem
  for (i = 0; i < 800; i++) {
    voltagem += ((float)analogRead(SensorTurbidez) / 1023) * 5;
  }

  // Realiza a média entre os valores lidos na função for acima
  voltagem = voltagem / 800;
  voltagem = ArredondarPara(voltagem, 1);

  // Se Voltagem menor que 2.5 fixa o valor de NTU
  if (voltagem < 2.5) {
    NTU = 3000;
  }

  else if (voltagem > 4.2) {
    NTU = 0;
    voltagem = 4.2;
  }

  // Senão calcula o valor de NTU através da fórmula
  else {
```

```

    NTU = -1120.4 * square(voltagem) + 5742.3 * voltagem - 4353.8;
}

// Imprime as informações na tela do LCD
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Leitura: ");
lcd.print(voltagem);
lcd.print(" V");

lcd.setCursor(3, 1);
lcd.print(NTU);
lcd.print(" NTU");

delay(10);
}

// Sistema de arredondamento para leitura
float ArredondarPara(float ValorEntrada, int CasaDecimal) {
    float multiplicador = powf(10.0f, CasaDecimal);
    ValorEntrada = roundf(ValorEntrada * multiplicador) / multiplicador;
    return ValorEntrada;
}

```

## 21 APÊNDICE M – Código-fonte do protótipo de condutividade elétrica – TDS (fase inicial)

Este apêndice contém o código desenvolvido para o protótipo inicial de monitoramento de condutividade elétrica (TDS), utilizado apenas em testes preliminares e posteriormente descontinuado.

```

// Código adaptado de:
// "Gravity: Analog TDS Sensor / Meter for Arduino (SEN0244)"
// por DFRobot
// DFRobot Wiki
// Disponível em:
https://wiki.dfrobot.com/Gravity\_\_Analog\_TDS\_Sensor\_\_\_Meter\_For\_Arduino\_SKU\_\_SEN0244
// Acesso em: 20 jan. 2026

#include <LiquidCrystal.h>

// Inicializa o LCD convencional (RS, E, D4, D5, D6, D7)
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define TdsSensorPin A1
#define VREF 5.0 // Tensão de referência analógica (V)
#define SCOUNT 30 // Soma do ponto amostral

int analogBuffer[SCOUNT];
int analogBufferTemp[SCOUNT];
int analogBufferIndex = 0, copyIndex = 0;
float averageVoltage = 0, tdsValue = 0, temperature = 25;

void setup()
{

```

```

Serial.begin(9600);
pinMode(TdsSensorPin, INPUT);

// Inicializa o LCD
lcd.begin(16, 2);
lcd.setCursor(0, 0);
lcd.print("BioDATTA-Kelvin");
lcd.setCursor(0, 1);
lcd.print("Condutividade");
delay(100);
lcd.clear();
}

void loop()
{
  static unsigned long analogSampleTimepoint = millis();
  if (millis() - analogSampleTimepoint > 400)
  {
    analogSampleTimepoint = millis();
    analogBuffer[analogBufferIndex] = analogRead(TdsSensorPin);
    analogBufferIndex++;
    if (analogBufferIndex == SCOUNT)
      analogBufferIndex = 0;
  }

  static unsigned long printTimepoint = millis();
  if (millis() - printTimepoint > 8000)
  {
    printTimepoint = millis();

    for (copyIndex = 0; copyIndex < SCOUNT; copyIndex++)
      analogBufferTemp[copyIndex] = analogBuffer[copyIndex];

    averageVoltage = getMedianNum(analogBufferTemp, SCOUNT) * (float)VREF /
1024.0;

    float compensationCoefficient = 1.0 + 0.02 * (temperature - 25.0);
    float compensationVoltage = averageVoltage / compensationCoefficient;

    tdsValue = (133.42 * compensationVoltage * compensationVoltage *
compensationVoltage
      - 255.86 * compensationVoltage * compensationVoltage
      + 857.39 * compensationVoltage) * 0.5;

    Serial.print("TDS Value:");
    Serial.print(tdsValue, 0);
    Serial.println("ppm");
  }

  lcd.setCursor(0, 0);
  lcd.print("Valor TDS: ");
  lcd.setCursor(11, 0);
  lcd.print(tdsValue, 1);
  delay(1000);
}

int getMedianNum(int bArray[], int iFilterLen)
{

```

```

int bTab[iFilterLen];
for (byte i = 0; i < iFilterLen; i++)
    bTab[i] = bArray[i];

int i, j, bTemp;
for (j = 0; j < iFilterLen - 1; j++)
{
    for (i = 0; i < iFilterLen - j - 1; i++)
    {
        if (bTab[i] > bTab[i + 1])
        {
            bTemp = bTab[i];
            bTab[i] = bTab[i + 1];
            bTab[i + 1] = bTemp;
        }
    }
}

if ((iFilterLen & 1) > 0)
    bTemp = bTab[(iFilterLen - 1) / 2];
else
    bTemp = (bTab[iFilterLen / 2] + bTab[iFilterLen / 2 - 1]) / 2;

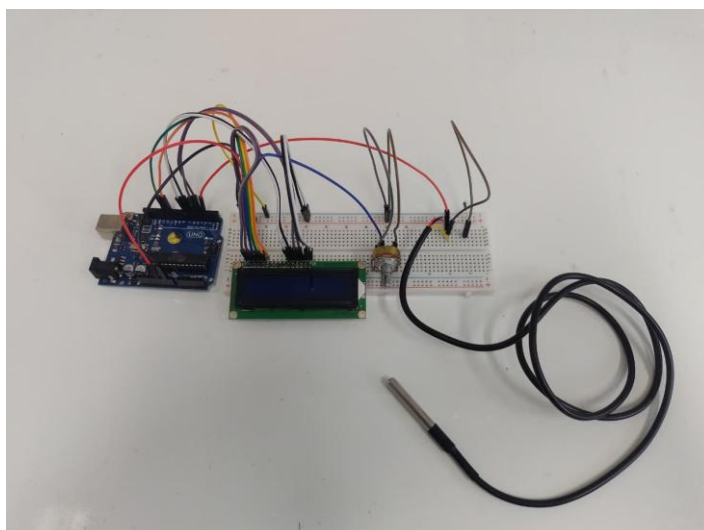
return bTemp;
}

```

## 22 APÊNDICE N – Evolução estrutural dos protótipos de pH e temperatura

Na versão inicial, os protótipos de pH e temperatura foram montados em protoboard, utilizando display LCD convencional com backlight verde. Essa configuração exigia o uso de potenciômetro para ajuste de contraste, resistores auxiliares e um elevado número de conexões elétricas, totalizando aproximadamente 12 cabos apenas entre o microcontrolador e o display, o que aumentava a complexidade da montagem e a suscetibilidade a falhas de contato.

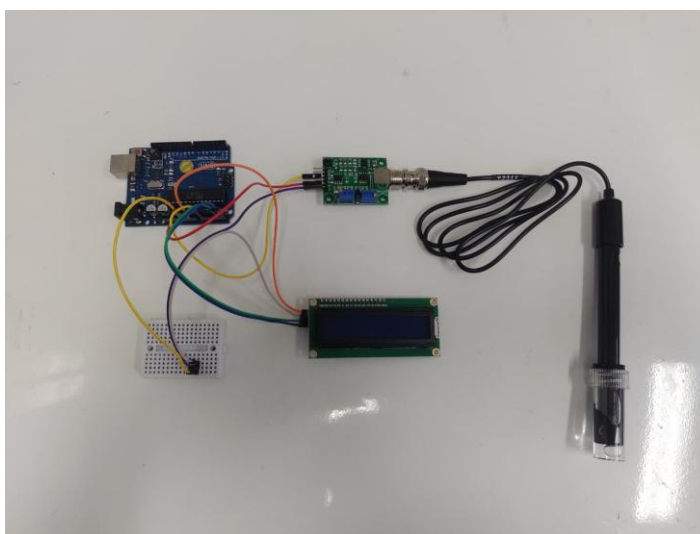
Figura N.1 – Protótipo inicial de temperatura montado em protoboard com display LCD convencional.



Fonte: Fotografia de autoria própria (2024).

Com o objetivo de reduzir a complexidade do sistema e melhorar a confiabilidade elétrica, os protótipos passaram a utilizar displays LCD com chip I2C integrado. Essa modificação permitiu a redução do número de conexões entre o display e o microcontrolador de aproximadamente 12 para apenas quatro cabos, eliminando a necessidade de potenciômetros e resistores externos, além de simplificar significativamente o processo de montagem.

Figura N.2 – Protótipo inicial de pH montado em protoboard com display LCD I2C.



Fonte: Fotografia de autoria própria (2024).

Na etapa final, os protótipos funcionais de pH e temperatura foram acondicionados em caixas protetoras confeccionadas por impressora 3D, promovendo melhor organização interna dos componentes e maior robustez estrutural. Essa configuração reduz a exposição dos circuitos a umidade, impactos mecânicos e manuseio inadequado, tornando os dispositivos mais adequados para aplicação em ambientes laboratoriais e de campo.

## 23 APÊNDICE O – Implementação do sistema de alimentação por bateria

A alimentação inicial dos protótipos era realizada exclusivamente por fonte externa ou conexão USB, o que limitava sua aplicação em ambientes sem acesso contínuo à rede elétrica. Visando ampliar a portabilidade e a aplicabilidade em campo, foi implementado um sistema de alimentação por bateria nos protótipos finais:

- Bateria (Ni-Mh 9V 450mAh)
- Conexão com o microcontrolador
- Interruptor liga/desliga tipo gangorra

- Módulo de carregamento com entrada USB-C (módulo step-up MT-3608)

O sistema de alimentação por bateria foi composto por uma bateria recarregável e um módulo de carregamento, que é um módulo conversor de tensão do tipo step-up, responsável por elevar e estabilizar a tensão, permitindo o uso de carregadores convencionais compatíveis com a maioria dos dispositivos celulares atuais.

A adoção do sistema de alimentação por bateria confere maior autonomia e flexibilidade aos protótipos, permitindo sua utilização em ambientes onde o acesso à energia elétrica é limitado ou inexistente.

Apesar das vantagens, a autonomia do sistema depende da capacidade da bateria e do consumo energético dos sensores e do display, sendo necessária, em trabalhos futuros, a otimização do consumo energético e a avaliação da duração da carga em condições reais de uso.

## 24 APÊNDICE P – Proposta conceitual de substituição do Arduino Uno R3 pelo Arduino Nano V3

Durante o desenvolvimento dos protótipos de pH e temperatura, foi utilizada a plataforma Arduino Uno R3, amplamente difundida no meio acadêmico, de fácil programação e elevada compatibilidade com sensores e módulos. Contudo, ao longo da evolução estrutural dos protótipos, identificou-se que o tamanho físico da placa e o consumo de espaço interno poderiam ser otimizados visando maior portabilidade e integração dos sistemas.

Tabela P.1 – Características do Arduino Uno R3 e do Arduino Nano V3

Característica	Arduino Uno R3	Arduino Nano V3
Característica	<b>Arduino Uno R3</b>	<b>Arduino Nano V3</b>
<b>Microcontrolador</b>	ATmega328P	ATmega328P
<b>Entradas digitais</b>	14 (6 com PWM)	14 (6 com PWM)
<b>Entradas analógicas</b>	6	8
<b>Memória Flash</b>	32 KB (0,5 KB usada pelo bootloader)	32 KB (2 KB usada pelo bootloader)
<b>Dimensões (mm)</b>	68,6 × 53,4	45 × 18
<b>Área aproximada</b>	~3663 mm <sup>2</sup>	~810 mm <sup>2</sup>
<b>Peso aproximado</b>	25 g	7 g
<b>Método de montagem</b>	Placa independente	Montagem em protoboard ou soldada
<b>Indicação principal</b>	Prototipagem em bancada	Sistemas embarcados compactos

Fonte: Elaboração própria (2025).

A substituição do Arduino Uno R3 pelo Arduino Nano V3 apresenta vantagens potenciais, especialmente no que se refere à miniaturização do sistema. O formato reduzido do Arduino Nano permite melhor aproveitamento do espaço interno das caixas de proteção, facilitando a integração com sensores, módulos de alimentação por bateria e displays, além de contribuir para a redução do peso total do dispositivo.

Considerando que os protótipos foram desenvolvidos utilizando bibliotecas e estruturas compatíveis com o microcontrolador ATmega328P, a migração para o Arduino Nano V3 não exigiria alterações significativas nos códigos de programação, representando apenas ajustes físicos e de disposição dos componentes.

Ressalta-se que a substituição da plataforma Arduino Uno R3 pelo Arduino Nano V3 não foi implementada no escopo deste trabalho, permanecendo como uma proposta conceitual. Aspectos como dissipação térmica, consumo energético e robustez em condições reais de campo devem ser avaliados em estudos futuros.

## 25 APÊNDICE Q – Detalhamento dos custos por componente dos protótipos

Este apêndice apresenta os componentes utilizados no desenvolvimento dos protótipos, bem como seus respectivos valores unitários aproximados, não correspondendo à quantidade total necessária para a implementação conjunta de todos os protótipos.

Tabela Q.1 – Detalhamento dos custos estimados dos componentes dos protótipos

<b>Módulo</b>	<b>Componente</b>	<b>Unidade</b>	<b>Valor estimado (R\$)</b>
<b>Processamento e Controle</b>	Placa Arduino Uno R3 + cabo USB	1	90,00
<b>Sensoriamento</b>	Sensor de pH com módulo BNC PH4502C	1	214,00
<b>Sensoriamento</b>	Sensor de temperatura DS18B20 à prova d'água	1	13,00
<b>Sensoriamento</b>	Sensor de fluxo de água YF-S201 G1/2	1	31,00
<b>Sensoriamento</b>	Sensor de turbidez ST100 com módulo de leitura	1	205,00
<b>Sensoriamento</b>	Sensor de TDS (condutividade da água)	1	120,00
<b>Interface com o Usuário</b>	Display LCD 16×2 I2C (fundo azul)	1	18,00
<b>Interface com o Usuário</b>	Display LCD 16×2 (fundo verde)	1	13,00
<b>Interface com o Usuário</b>	Potenciômetro linear 10kΩ	1	4,00

<b>Interface com o Usuário</b>	Interruptor liga/desliga tipo gangorra	1	2,00
<b>Alimentação</b>	Bateria 9V recarregável 450mAh	1	45,00
<b>Alimentação</b>	Conector clip para bateria 9V	1	2,00
<b>Alimentação</b>	Módulo step-up MT3608 (USB-C)	1	16,00
<b>Componentes Passivos</b>	Resistor 10kΩ, 1/4 W	1	0,20
<b>Componentes Passivos</b>	Resistor 330Ω, 1/4 W	1	0,20
<b>Estrutura Mecânica</b>	Case impressa em 3D em PLA*	1	4,00
<b>Estrutura Mecânica</b>	Case impressa em 3D em ABS*	1	6,00
<b>Conexões e Montagem</b>	Cabos (jumpers)	10	5,00
<b>Testes</b>	Protoboard 830 pontos para montagem de projetos	1	12,00

\*O custo estimado para a fabricação da case impressa em 3D foi calculado com base na simulação de fatiamento realizada no software Creality Print (versão 7.0), que fornece estimativas valor baseado no material utilizado. Fonte: Elaboração própria, com base em valores médios de mercado (2026).

## 26 APÊNDICE R – Código-fonte do sistema de registro de dados com módulo RTC e armazenamento em cartão SD

Este apêndice apresenta o código de programação desenvolvido para implementação de um sistema de registro automatizado de dados ambientais utilizando um módulo de relógio de tempo real (RTC – Real Time Clock) e um módulo de armazenamento em cartão de memória SD.

```
// Código desenvolvido por:
// Kelvin Marlon Barbosa Lacerda
// Universidade Federal do Pará (UFPA)
// Trabalho de Conclusão de Curso - 2026
// Título: Desenvolvimento de protótipos com Arduino para monitoramento
// de parâmetros da água no cultivo de Macrobrachium amazonicum (Heller, 1862)
//
// Descrição:
// Sistema de registro automatizado de dados utilizando módulo RTC e cartão SD,
// permitindo o armazenamento de medições com data e horário.
//
// Data de desenvolvimento: 2025

#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include <RTClib.h>

RTC_DS3231 rtc; // Utilize RTC_DS1307 se estiver usando o módulo DS1307
File myFile;
```

```

void setup() {
  // Inicialização Serial
  Serial.begin(9600);

  // Inicialização do módulo RTC
  if (!rtc.begin()) {
    Serial.println("RTC não encontrado!");
    while (1);
  }

  if (rtc.lostPower()) {
    Serial.println("RTC perdeu a energia, definindo a hora...");
    // Defina a hora correta aqui
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }

  // Inicialização do módulo SD Card
  if (!SD.begin(10)) {
    Serial.println("Falha na inicialização do SD!");
    while (1);
  }

  // Cria ou abre o arquivo para armazenar os dados
  myFile = SD.open("datalog.txt", FILE_WRITE);
  if (!myFile) {
    Serial.println("Erro ao abrir o arquivo!");
    while (1);
  }
}

void loop() {
  DateTime now = rtc.now();

  // Cria uma string com os dados de data e hora
  String dataString = String(now.year()) + "/" +
    String(now.month()) + "/" +
    String(now.day()) + " " +
    String(now.hour()) + ":" +
    String(now.minute()) + ":" +
    String(now.second());

  // Escreve a string no arquivo
  myFile.println(dataString);
  myFile.flush();

  // Exibe a string no monitor serial
  Serial.println(dataString);

  // Espera 1 minuto (60000 milissegundos)
  delay(60000);
}

```

## 27 APÊNDICE S – Código-fonte do sistema de registro e transmissão remota de dados com módulos RTC, cartão SD e comunicação WiFi

Este apêndice contém o código de programação desenvolvido para implementação de um sistema integrado de registro e transmissão remota de dados, utilizando módulo de relógio de tempo real (RTC), módulo de armazenamento em cartão SD e módulo de comunicação sem fio (WiFi).

```
// Código desenvolvido por:
// Kelvin Marlon Barbosa Lacerda
// Universidade Federal do Pará (UFPA)
// Trabalho de Conclusão de Curso - 2026
// Título: Desenvolvimento de protótipos com Arduino para monitoramento
// de parâmetros da água no cultivo de Macrobrachium amazonicum (Heller, 1862)
//
// Descrição:
// Sistema de registro e transmissão remota de dados utilizando módulo RTC,
// cartão SD e comunicação WiFi, permitindo armazenamento local e envio
// remoto das medições.
//
// Data de desenvolvimento: 2025

#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include <RTClib.h>
#include <SoftwareSerial.h>

RTC_DS3231 rtc;
File myFile;
SoftwareSerial espSerial(2, 3); // RX, TX

void setup() {
  // Inicialização Serial
  Serial.begin(9600);
  espSerial.begin(115200);

  // Inicialização do módulo RTC
  if (!rtc.begin()) {
    Serial.println("RTC não encontrado!");
    while (1);
  }

  if (rtc.lostPower()) {
    Serial.println("RTC perdeu a energia, definindo a hora...");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }

  // Inicialização do módulo SD Card
  if (!SD.begin(10)) {
    Serial.println("Falha na inicialização do SD!");
    while (1);
  }

  // Cria ou abre o arquivo para armazenar os dados
  myFile = SD.open("datalog.txt", FILE_WRITE);
```

```

if (!myFile) {
    Serial.println("Erro ao abrir o arquivo!");
    while (1);
}

// Configuração inicial do ESP8266 (exemplo: modo cliente)
espSerial.println("AT+CWMODE=1"); // Configura como cliente
delay(2000);
espSerial.println("AT+CWJAP=\"SSID\", \"PASSWORD\""); // Conectar ao Wi-Fi
delay(5000);
}

void loop() {
    DateTime now = rtc.now();

    // Cria uma string com os dados de data e hora
    String dataString = String(now.year()) + "/" +
        String(now.month()) + "/" +
        String(now.day()) + " " +
        String(now.hour()) + ":" +
        String(now.minute()) + ":" +
        String(now.second());

    // Escreve a string no arquivo
    myFile.println(dataString);
    myFile.flush();

    // Envia dados para o monitor serial
    Serial.println(dataString);

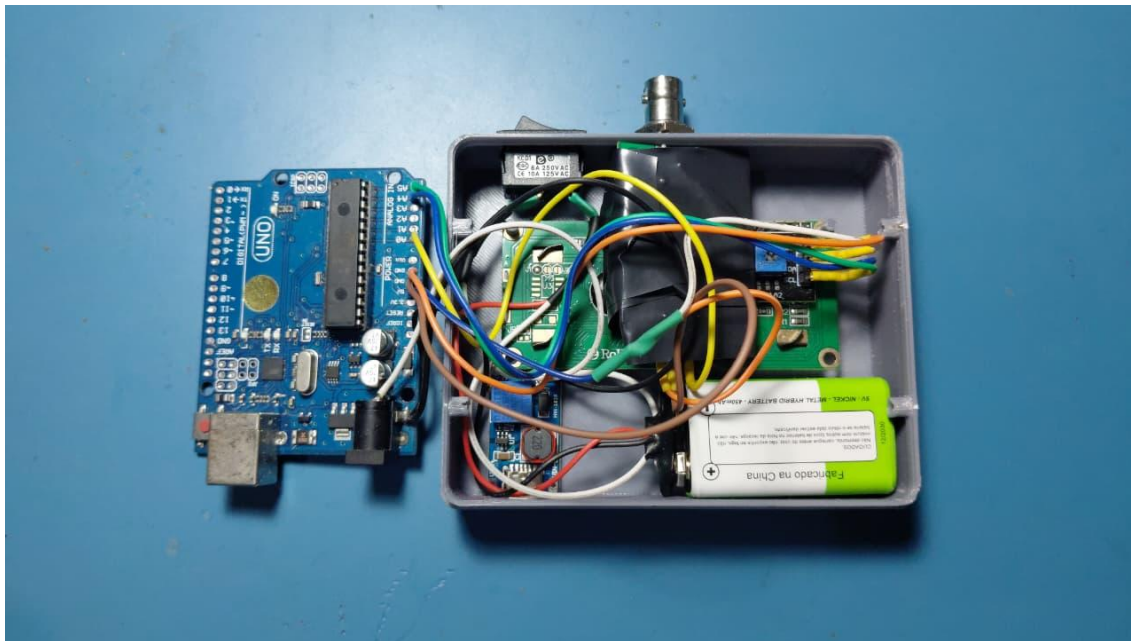
    // Envia dados para o computador via Wi-Fi
    espSerial.println("AT+CIPSTART=\"TCP\", \"192.168.1.100\", 80"); // Conectar ao IP
do servidor
    delay(2000);
    espSerial.print("AT+CIPSEND=");
    espSerial.println(dataString.length());
    delay(1000);
    espSerial.println(dataString); // Enviar dados
    delay(2000);
    espSerial.println("AT+CIPCLOSE");

    // Espera 1 minuto
    delay(60000);
}

```

## 28 APÊNDICE T – Registros fotográficos complementares dos protótipos finais

Figura T.1 – Vista frontal da disposição do circuito do protótipo de pH semi-acondicionado no invólucro.



Fonte: Fotografia de autoria própria (2026).

O módulo conversor do sensor de pH está envolto em fita isolante para evitar contato acidental com algum componente do LCD, por estarem próximos.

Figura T.2 – Vista frontal do circuito do protótipo de pH acondicionado no invólucro sem tampa.



Fonte: Fotografia de autoria própria (2026).

Figura T.3 - Vista superior do protótipo final, contemplando o LCD, interruptor e o conector da sonda de pH (ou saída do sensor DS18B20 no caso do protótipo de temperatura)



Fonte: Fotografia de autoria própria (2026).

Figura T.4 – Vista lateral do protótipo final, contemplando os parafusos que mantêm o invólucro fechado.



Fonte: Fotografia de autoria própria (2026).

Figura T.5 – Vista inferior do protótipo final, contemplando a entrada USB-C para carregamento.



Fonte: Fotografia de autoria própria (2026).

Figura T.6 – Vista traseira do protótipo final, contemplando pés antiderrapantes/antichoque.



Fonte: Fotografia de autoria própria (2026).