



**INSTITUTO
FEDERAL**

Brasília

Instituto Federal de Brasília
Campus Taguatinga
Tecnologia em Automação Industrial

Carlos Eduardo Lopes Bezerra

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA AUTOMAÇÃO E
SUPERVISÃO REMOTA DE BOMBAS D'ÁGUA**

Brasília
2024

Carlos Eduardo Lopes Bezerra

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA AUTOMAÇÃO E
SUPERVISÃO REMOTA DE BOMBAS D'ÁGUA**

Monografia apresentada ao curso de Tecnologia em Automação Industrial do Instituto Federal de Brasília do *Campus* Taguatinga, como requisito parcial para a obtenção do título de Tecnólogo em Automação Industrial.

Orientador: Prof. Me. Sérgio Messias Cruz
Instituto Federal de Brasília

Brasília
2024

Carlos Eduardo Lopes Bezerra

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA AUTOMAÇÃO E
SUPERVISÃO REMOTA DE BOMBAS D'ÁGUA**

Monografia apresentada ao curso de Tecnologia em Automação Industrial do Instituto Federal de Brasília do *Campus* Taguatinga, como requisito parcial para a obtenção do título de Tecnólogo em Automação Industrial.

Aprovado em

BANCA EXAMINADORA

Prof. Me. Sérgio Messias Cruz
Instituto Federal de Brasília

Prof. Dr. Marcus Vinicius Costa Sá
Instituto Federal de Brasília

Prof. Dr. Marlon Marques Soudré
Instituto Federal de Brasília

Ficha de identificação da obra elaborada pelo bibliotecário
Marcelo José Rodrigues da Conceição (CRB1-2323)

Bezerra, Carlos Eduardo Lopes

B574d Desenvolvimento de um protótipo para automação e supervisão remota de bombas d'água e controle / Carlos Eduardo Lopes Bezerra. Brasília-DF, 2024.

77 f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Automação Industrial) Instituto Federal de Educação, Ciência e Tecnologia de Brasília. *Campus* Taguatinga, 2024.

Orientador: Sérgio Messias Cruz.

Inclui referências.

1. Automação industrial. 2. Controladores programáveis. 3. Programação (Computadores). 4. Arduino (Controlador programável). I. Título. II. Cruz, Sérgio Messias. III. Instituto Federal de Educação, Ciência e Tecnologia de Brasília.

CDU 681.5:004

Dedico este trabalho àqueles que, direta ou indiretamente, contribuíram para minha formação e crescimento ao longo dessa jornada acadêmica.

AGRADECIMENTOS

Agradeço a todos que me acompanharam e apoiaram durante essa jornada. À minha família, por sua paciência e incentivo incondicional; à pessoa especial que esteve sempre ao meu lado, oferecendo carinho e apoio inestimável; aos amigos e colegas, pelos momentos compartilhados de estudo e desafios; e aos professores e orientadores, por sua sabedoria e dedicação essenciais para este processo. Também sou profundamente grato ao servidor Alexandre Moraes, técnico em eletrônica do IFB *campus* Taguatinga, quem me repassou a demanda que inspirou o tema deste trabalho, confiando em mim para desenvolver essa ideia. Este trabalho é fruto não apenas do meu esforço, mas também da contribuição de todos que tornaram possível a realização desse objetivo.

*“O ontem é história, o amanhã é um mistério,
mas o hoje é uma dádiva. Por isso chama-se
presente.”*

— **Mestre Oogwey**

RESUMO

Com o avanço da tecnologia, a automação residencial se torna cada vez mais presente na sociedade, proporcionando alto nível de conforto, segurança, acessibilidade e até mesmo sustentabilidade. Contudo, os elevados custos dos sistemas comerciais se tornam uma barreira de entrada aos usuários, resultando em demandas não atendidas e levando-os a buscar alternativas mais econômicas. O objetivo deste projeto foi atender a demanda levantada por um condomínio que não dispõe de orçamento para alto investimento em automação e conta com um sistema de abastecimento de água por poço artesiano, onde é comum a danificação da bomba por falta de água ou por oscilação na rede de energia. O dispositivo desenvolvido desempenhou o papel de controlar a bomba d'água autonomamente ou manualmente, além de possibilitar supervisão remota, prevenindo defeitos e permitindo replicação em outros contextos.

@articleherculano2024desenvolvimento, title=Desenvolvimento de um circuito de controle de bomba d'água: Com ajuste dinâmico de velocidade., author=HERCULANO, Gabriel Gonçalves and SALVADOR, Thiago Camargo and SILVA, André dos Santos and SOUZA, Everton Pereira de and TOLEDO, Estevão Gabriel Silva, year=2024, publisher=228

<https://ric.cps.sp.gov.br/handle/123456789/22050>

<https://scholar.googleusercontent.com/scholar.bib?q=info:uF4Ut4EBIbQJ:scholar.google.com/output-1hl=pt-BR>

Palavras-chave: Automação Residencial. Bomba d'água. Dispositivo Autônomo.

ABSTRACT

With the advancement of technology, home automation is becoming increasingly present in society, providing a high level of comfort, security, accessibility, and even sustainability. However, the high costs of commercial systems create a barrier for users, resulting on unmet demands and prompting them to seek more affordable alternatives. This project aims to meet the needs identified by a condominium that lacks the budget for significant investment in automation and relies on a water supply system through an artesian well, where pump damage due to water shortage or power grid fluctuations is common. The device developed will control the water pump either autonomously or manually, in addition to allowing remote monitoring, preventing malfunctions, and enabling replication in other contexts.

Keywords: Home Automation. Water Pump. Autonomous device.

LISTA DE FIGURAS

Figura 2.1 – Controlador de nível NI35W	7
Figura 2.2 – Controlador Eletrônico TPC-58	8
Figura 2.3 – Diagrama Funcional do Microcontrolador ESP8266	10
Figura 3.1 – Módulo Sensor de Tensão ZMPT101B.	12
Figura 3.2 – Sensor de Corrente SCT013.	13
Figura 3.3 – Relé JQC-3F(T73).	14
Figura 3.4 – Vista Frontal do Display LCD com Módulo I2C.	14
Figura 3.5 – Vista Traseira do Display LCD com Módulo I2C.	15
Figura 3.6 – ESP-WROOM-32.	15
Figura 3.7 – Tela Inicial do Projeto na Plataforma Blynk.	16
Figura 3.8 – Sinais senoidais sem <i>offset</i> e com <i>offset</i>	18
Figura 3.9 – Circuito elétrico do sensor de corrente SCT-013	19
Figura 3.10–Circuito elétrico do relé.	20
Figura 3.11–Botões utilizados para acionamento manual do dispositivo.	21
Figura 3.12–Botões com o circuito de <i>pulldown</i> implementado.	22
Figura 3.13–Fluxograma do firmware do Dispositivo.	23
Figura 3.14–Esquema elétrico desenvolvido no Proteus para a versão 3.0 do dispositivo.	24
Figura 3.15–Bibliotecas utilizadas no firmware.	25
Figura 3.16–Interface do <i>Blynk</i> no aplicativo móvel.	27
Figura 4.1 – Protótipo, versão 1.0, montado em uma placa de fenolite perfurada.	28
Figura 4.2 – CD74HC4053E - Descrição das portas	29
Figura 4.3 – CD74HC4053E - Tabela Verdade	30
Figura 4.4 – Módulo Sensor de Tensão AC 127/220V.	31
Figura 4.5 – Esquema elétrico desenvolvido no proteus para a versão 2.0 do dispositivo.	33
Figura 4.6 – Modelo 3D da placa projetada para a versão 2.0 do dispositivo.	33
Figura 4.7 – Protótipo, versão 2.0, montado em uma <i>protoboard</i> para teste.	34
Figura 4.8 – Leitura do Multímetro no dispositivo versão 3.0.	35
Figura 4.9 – Leitura do Sensor de Tensão Calibrado no Dispositivo Versão 3.0.	35
Figura 4.10–ESP32 com descrição das portas.	36
Figura 4.11–Protótipo, versão 3.0, montado em uma <i>protoboard</i> para teste.	37
Figura 4.12–Modelo 3D da placa projetada para a versão 3.0 do dispositivo.	38
Figura 4.13–Trilhas do lado inferior do projeto da PCB	39
Figura 4.14–Impressão do lado superior do projeto da PCB	39
Figura 4.15–Sistema de simulação em bancada.	40
Figura 4.16–Leitura por multímetro e sensor da corrente consumida.	41
Figura 4.17–Gráfico de comparação entre os estados dos botões e a bomba d’água.	42

Figura 4.18–Gráfico de comparação entre a corrente e o estado da bomba d'água. . .	43
Figura 4.19–Gráfico de comparação entre a tensão da rede e a bomba d'água. . . .	44

LISTA DE ABREVIATURAS E SIGLAS

Auresid	Associação Brasileira de Automação Residencial e Predial
CLP	Controlador Lógico Programável
<i>EPROM</i>	<i>Erasable Programmable Read-Only Memory</i>
IFB	Instituto Federal de Brasília
<i>I2C</i>	<i>Inter-Integrated Circuit</i>
<i>IoT</i>	<i>Internet of Things</i>
<i>LCD</i>	<i>Liquid Crystal Display</i>
<i>LED</i>	<i>Light-Emitting Diode</i>
<i>PCB</i>	<i>Printed Circuit Board</i>
<i>PIC</i>	<i>Peripheral Interface Controller</i>
<i>RAM</i>	<i>Random Access memory</i>
<i>SCL</i>	<i>Serial Clock</i>
<i>SDL</i>	<i>serial Data</i>
<i>SoC</i>	<i>System-on-Chip</i>
TCC	Trabalho de Conclusão de Curso
<i>USB</i>	<i>Universal Serial Bus</i>
<i>wi-fi</i>	<i>Wireless Fidelity</i>

LISTA DE SÍMBOLOS

A	Ampere
F	Farad
V	Volt
Ω	Ohm
μ	Micro

SUMÁRIO

1	INTRODUÇÃO	1
2	FUNDAMENTAÇÃO TEÓRICA	3
2.1	Caracterização da Automação Residencial	3
2.2	Trabalhos Correlatos	4
2.2.1	<i>Protótipo de Automação Residencial Utilizando Uma Assis-</i> <i>tente de Voz</i>	4
2.2.2	<i>Uso do Microcontrolador Esp8266 para Automação Residencial</i>	4
2.2.3	<i>Automação Residencial - Sistema Integrado de Uma Residên-</i> <i>cia Inteligente Utilizando Controlador Lógico Programável . .</i>	5
2.2.4	<i>DESENVOLVIMENTO DE UM CIRCUITO DE CONTROLE</i> <i>DE BOMBA D'ÁGUA: com ajuste dinâmico de velocidade . .</i>	6
2.2.5	<i>COEL - NI35W Controlador de Nível</i>	6
2.2.6	<i>Thebe TPC-58</i>	7
2.3	Microcontroladores	9
3	METODOLOGIA	11
3.1	Materiais e Métodos	11
3.2	Protótipo Desenvolvido	12
3.2.1	<i>Sensor De Tensão</i>	12
3.2.2	<i>Sensor De Corrente SCT-013-000</i>	13
3.2.3	<i>Interruptor eletrônico - Relé</i>	14
3.2.4	<i>Display LCD com módulo I2C</i>	14
3.2.5	<i>ESP32</i>	15
3.2.6	<i>Componentes Auxiliares</i>	15
3.2.7	<i>Blynk</i>	16
3.3	Detalhes de Construção	16
3.4	Fluxograma do Firmware	23
3.5	Esquema Elétrico	24
3.6	Descrição do Firmware do Dispositivo	25
4	RESULTADOS E DISCUSSÃO	28
4.1	Protótipo versão 1.0	28
4.2	Protótipo versão 2.0	31
4.3	Protótipo versão 3.0	34
4.4	Testes Práticos	40

5	CONSIDERAÇÕES FINAIS	47
6	SUGESTÃO DE MELHORIAS FUTURAS	48
	REFERÊNCIAS	49
	APÊNDICE A – FIRMWARE	53

1 INTRODUÇÃO

A automação em geral passou a fazer parte da sociedade com mais ênfase em meados do século XVIII, na Inglaterra. Isso se deu com o advento dos primeiros dispositivos semi automáticos que foram desenvolvidos para transformar os sistemas de produção artesanal e agrário em industrial. Porém, os sistemas de fato automáticos começaram a surgir apenas no século XX (KUSTRO, 2017).

O termo ‘automação’ começou a se popularizar nos anos 50 e se remetia à movimentação automática de materiais (KUSTRO, 2017).

O objetivo principal da automação é reduzir a necessidade da intervenção humana em seus processos. A criação do moinho hidráulico para fornecimento de farinha no século X foi um dos primeiros dispositivos automáticos a serem criados, ainda que rústico, e era capaz de substituir o trabalho de 10 a 20 homens (GOEKING, 2010).

Com isso, podemos observar a presença de sistemas autônomos em várias aplicações distintas. Alguns exemplos possíveis nos tempos modernos são os automóveis Tesla que, segundo a empresa norte-americana de mesmo nome, possui um piloto automático baseado no processamento de imagem capturada por 8 câmeras e 12 sensores ultrassônicos a fim de identificar objetos sólidos e flexíveis a uma distância de até 250 metros. Outro exemplo é a inteligência artificial da rede social Instagram que, segundo a própria rede social, identifica autonomamente conteúdos que violam as diretrizes da rede e os removem antes mesmo de qualquer denúncia. E, por fim, temos o exemplo que vai nortear o presente trabalho: a automação de equipamentos residenciais. Para isso, a empresa brasileira de tecnologia WEG apresenta alguns tipos de automação, como: monitoramento da iluminação, sensores de vazamento de água e automação de cortinas e persianas. A aplicação dessa tecnologia é vasta e possui várias possibilidades de nichos. O sistema que a empresa produz com foco específico em equipamentos residenciais pode ser reconhecido como um sistema de automação residencial.

A automação residencial é apenas uma área da automação, dentre várias outras. Como o próprio nome evidencia, esta área tem como foco automatizar processos dentro de uma residência, e possui objetivos específicos que, ainda segundo a WEG, são: garantir conforto e segurança para o morador; facilitar a execução de tarefas repetitivas, garantindo mais praticidade ao usuário; e controlar gastos com energia e água, contribuindo para a sustentabilidade (WEG, 2020).

Com isso, é importante entender como essa tecnologia está sendo difundida pelo Brasil. Segundo a Associação Brasileira de Automação Residencial e Predial (Aureside), a área de automação residencial tem uma projeção de crescimento de 22% ao ano até 2025, evidenciando a procura das pessoas por uma residência automatizada (AURESIDE, 2020). Contudo, há um dificultador para a distribuição da automação residencial: o custo de

aquisição. De acordo com a empresa de automação residencial BASS, os profissionais da automação cobram a partir de R\$6 mil (seis mil reais) para automatizar uma residência em Brasília. E este valor pode ser maior que R\$30 mil (trinta mil reais) a depender de como será a complexidade da automação implementada. Há a ressalva de que esses valores não incluem o custo dos equipamentos necessários para a automação, podendo elevar ainda mais o custo efetivo total. Com isso, é fácil perceber que esse é um mercado em franco crescimento mas que possui valores altos para aquisição, ainda mais para pessoas com baixa capacidade de investimento.

Um exemplo real da necessidade da automação foi apresentado por um técnico do IFB (Instituto Federal de Brasília) Campus Taguatinga. Ele é morador de um condomínio localizado em uma área rural de Brasília e que não possui fornecimento de água da concessionária. Com isso, é necessário que a água para consumo seja retirada de um poço e enviada para uma caixa d'água a partir de uma bomba elétrica. Porém, é comum que o nível de água na caixa diminua ao ponto de obrigar que o consumo pelos moradores tenha que ser interrompido. Também é comum que o poço fique sem água por um período de tempo imprevisível e a bomba funcione à vazia, conseqüentemente sobrecarregando e danificando-a, levando prejuízo ao condomínio. E, por último, outro desafio recorrente ao condomínio é a oscilação de tensão da rede de energia, ficando distante dos 220V (duzentos e vinte volts) habituais, o que também danifica a bomba elétrica. E, para solucionar tais questões, o condomínio não dispõe de um orçamento para investimento em um dos sistemas de automação já disponíveis no mercado, encontrando a necessidade de buscar uma solução mais acessível assim como a desenvolvida neste projeto.

Dessa forma, tem-se como objetivo do presente trabalho desenvolver um dispositivo para automação residencial que seja capaz de monitorar o funcionamento da bomba d'água, afim de identificar falta de água no poço e oscilação na rede de energia. Para isso, criou-se os seguintes objetivos específicos: criar um sistema de monitoramento da corrente da bomba e da tensão da rede, criar sistema de acionamento da bomba e criar sistema de controle e supervisão remota. A partir disso, foi possível acionar automaticamente o protótipo apenas em contextos favoráveis com parâmetros pré definidos de tensão e corrente que, para este trabalho, foram mensurados em testes práticos, obtendo os seguintes valores: 7,2A para valor nominal de corrente, 5,6A para valor mínimo de corrente e 200V para valor mínimo de tensão.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta etapa do projeto, será abordado o conteúdo teórico acerca dos conceitos já mencionados. Da mesma forma, ocorrerá o embasamento da definição dos termos a partir de pesquisas realizadas em dissertações e estudos de pesquisadores da área da automação e afins.

2.1 Caracterização da Automação Residencial

Segundo Groover (2002), a automação pode ser definida como a combinação entre as aplicações mecânicas, eletrônicas e sistemas computadorizados para controlar e operar uma produção. Ainda segundo o autor, a automação pode ser dividida em duas categorias: automatização do sistema de fabricação e informatizar os sistema de apoio à fabricação.

Concordando com Groover (2002), o *DICTIONARY*, dicionário oficial da universidade norte americana Cambridge, diz que automação é o uso de máquinas e computadores que podem operar sem a necessidade de intervenção humana.

Já para a automação residencial em específico, Dias (2023) traz outros termos que se referem ao mesmo conceito como ‘casas inteligentes’, ‘casas conectadas’ e ‘domótica’. O último é o mais utilizado para referenciar essa área.

Ainda segundo Dias (2023), domótica é a junção de duas palavras: *domus*, que em latim significa casa, e robótica, que se refere aos sistemas robotizados. E, fundamentando o conceito, Plati (Apud at DIAS, 2023) diz que a domótica é uma nova experiência que possibilita uma grande conexão e controle dos aparelhos eletrônicos de uma casa e que esse controle pode ser feito remotamente, desde um *keyboard* até às palmas das mãos a partir de um dispositivo conectado a uma rede *wireless*. Por esse motivo a domótica é uma tecnologia que proporciona conforto, segurança, eficiência e controle sob os vários processos residenciais.

Para exemplificar, podemos trazer um breve resumo sobre o projeto realizado por SANTOS *et al.* (2017). Eles apresentaram, em seu trabalho de conclusão de curso, a ideia de desenvolver um dispositivo que auxilia pessoas com deficiência motora parcial a realizarem tarefas do cotidiano de sua residência por meio de dispositivos móveis. Com base nessas informações, foi proposto por eles desenvolver um aplicativo para dispositivos móveis com o sistema operacional *android*, desenvolver um circuito eletrônico e realizar a comunicação entre o aplicativo e o circuito eletrônico.

Dessa forma, o circuito seria utilizado para controlar aparelhos da necessidade da pessoa com deficiência, neste caso foram tomados como exemplo uma lâmpada e um ventilador. O aplicativo teria uma interface simples e com apenas alguns botões. E a comunicação Bluetooth seria feita por meio da linguagem C e um microcontrolador

PIC. Sendo assim, a pessoa pode controlar aparelhos eletrônicos sem a necessidade de se locomover. Portanto, podemos observar a partir do trabalho apresentado acima, que a domótica de fato possibilita conforto e segurança às pessoas que optam por seu uso.

2.2 Trabalhos Correlatos

Um dos objetivos do presente trabalho é realizar o monitoramento e o controle de uma bomba elétrica de forma remota. Para isso, estudos foram realizados com base em outros projetos relacionados a área da automação. Nesta seção, serão apresentados trabalhos que possuem relação com este projeto, possibilitando absorver ideias que podem ser utilizadas e melhoradas para desenvolver o dispositivo com o máximo de profundidade e inovação possível.

2.2.1 Protótipo de Automação Residencial Utilizando Uma Assistente de Voz

NETO (2018) trouxe em seu trabalho de conclusão de curso um projeto no qual tinha como objetivo controlar a abertura de uma porta e aferir a temperatura de um ambiente. Para isso, foi utilizado um eletroímã para abrir ou trancar a porta e um sensor de temperatura e umidade DHT11 para registrar os dados da temperatura. O microcontrolador utilizado para centralizar os comandos foi o Arduíno Ameba, uma plataforma aberta para programação. O software foi composto por uma série de plataformas online que tinham como eixo central a Amazon Web Services (AWS), são elas: A AWS Lambda, a AWS IoT (Amazon Web Services Internet of Things), a assistente virtual ‘Alexa’, o servidor HTTP da Thingspeak e a plataforma de desenvolvimento do firmware do protótipo baseado em linguagem C++, a IDE (Integrated Development Environment) oficial do Arduíno. O eixo central do sistema é a assistente virtual Alexa que, ao receber um comando, o interpreta e executa uma ação pré definida, que pode ser abrir a porta ou aferir a temperatura de um ambiente. A estrutura do projeto foi feita em uma maquete residencial de madeira MDF que comporta os servo motores, o sensor, o controlador e os demais componentes. Os pontos que puderam ser avaliados deste projeto e, por conseguinte aproveitados, são os seguintes: o conceito de utilizar um software intermediário para requisitar as informações do sensor e a utilização de um software em nuvem para enviar comandos à distância. O contraponto avaliado deste projeto é a quantidade de softwares da cadeia de comando que pode dificultar a replicação e diminuir a escalabilidade do dispositivo.

2.2.2 Uso do Microcontrolador Esp8266 para Automação Residencial

Neste projeto, Oliveira (2017a) trouxe um sistema de automação residencial controlado pelo microcontrolador Esp8266. Porém, diferentemente dos anteriores, o método de acionamento é baseado em um *web site* autoral do criador do projeto. O projeto teve

como foco viabilizar a automação de 6 lâmpadas com o controlador já mencionado e, para isso, foram utilizados 6 servo-motores para o acionamento eletrônico. Além do acionamento eletrônico/digital, foram implementados interruptores para o acionamento manual que funciona em paralelo com o acionamento digital devido ao esquema de ligação *threeway*. Para o controle digital, foi criada uma página *web* com o fim de centralizar os comandos. A escolha por uma página *web* foi feita pela maior facilidade de construção e implementação quando comparada à um aplicativo para *smartphones*. A comunicação da página *web* com o microcontrolador é feita por meio de um código IP gerado diretamente pelo Esp8266. Os códigos IPs são facilitadores para a comunicação entre várias ESPs caso o sistema de automação seja mais complexo e exija mais de uma placa, pois cada ESP tem um servidor único gerado por ela mesma e que pode ser acessado individualmente, por meio do IP, impedindo que haja confusão no acionamento de cada dispositivo. A comunicação a partir de uma página *web* pôde ser estudada e avaliada como possibilidade de implementação para este trabalho. Outro ponto do projeto acima que pode ser avaliado como opção de adaptação, é o acionamento por interruptores/botões físicos baseados em uma lógica *threeway*.

2.2.3 Automação Residencial - Sistema Integrado de Uma Residência Inteligente Utilizando Controlador Lógico Programável

Neste trabalho de conclusão de curso, Morais (2017) propõe o desenvolvimento de um sistema de automação residencial diferente dos referenciados anteriormente, baseado no controle por um Controlador Lógico Programável (CLP). O sistema possui uma grande quantidade de dispositivos a serem controlados, envolvendo sensores, botões e atuadores, tornando o processo mais complexo, englobando várias etapas do que é uma rotina residencial. Os dispositivos base utilizados no projeto são os seguintes: quatro lâmpadas externas, irrigadores de jardim, um portão de garagem e uma caixa d'água. Para atuar sobre os componentes e receber informações destes, foram utilizados alguns outros dispositivos. São eles: sensores de presença infravermelhos, relés fotoelétricos e botão de emergência para a iluminação externa. Motor de duas direções, dois sensores de proximidade indutivos, um sensor óptico tipo barreira, um controle remoto e um receptor de sinal para o portão da garagem. E seis sensores de nível para a caixa d'água. O funcionamento do sistema é baseado em um CLP, responsável por controlar todo o sistema. E, para centralizar o controle, foi implementado um sistema supervisor do software *FactoryTalk View Studio* para definir o horário da irrigação e o *setpoint* do nível da caixa d'água, além de outras informações adjacentes. A iluminação externa possui uma condicional para não operar durante o dia ou caso não identificar alguma presença nas áreas delimitadas pelo projetista. O portão da garagem tem seu funcionamento baseado em sensores que irão identificar se há alguma presença na linha do portão caso esteja aberto e se está na posição aberta ou fechada, informação fornecida pelos sensores de fim de curso. Um ponto que pode ser

aproveitado deste projeto é a ideia de implementar um sistema supervisor. Por ele é possível alterar parâmetros do sistema de forma simples e ter um controle completo do processo de forma centralizada e remota.

2.2.4 DESENVOLVIMENTO DE UM CIRCUITO DE CONTROLE DE BOMBA D'ÁGUA: com ajuste dinâmico de velocidade

HERCULANO *et al.* (2024) traz em seu trabalho de conclusão de curso o desenvolvimento de um circuito com o objetivo de regular a vazão de uma bomba d'água de acordo com o fluxo de água do sistema. Com isso, os autores esperam que o circuito garanta eficiência operacional da bomba, contribuindo com a sustentabilidade ao otimizar o uso dos recursos hídricos. Para isso, foi criado um circuito que possibilita identificar o nível mínimo e máximo de água do reservatório de prova. Além disso, o circuito também permite regular eletronicamente a velocidade de operação da bomba conectada a ele. Os componentes utilizados para atingir esses objetivos foram: Circuito integrado 4011, Relé, transistor, sensores de nível, *Dimmer Triac* e componentes auxiliares como resistores, *LEDs*, diodos e chave *on/off*. Estes componentes foram montados e soldados em uma placa de fenolite perfurada após passar por uma etapa de validação em uma *protoboard*. Para testar experimentalmente, os autores prepararam um sistema de simulação com um balde de 5 litros e mangueiras e conexões hidráulicas. HERCULANO *et al.* (2024) concluem que a eficácia do sistema foi provada por meio dos testes e que o circuito pode apresentar grande contribuição com o uso de recursos hídricos principalmente durante a variação dos horários de pico do uso de água, já que o circuito aciona a bomba com 50% da velocidade fora dos horários de pico e com 100% da velocidade durante o horário de pico. A partir deste projeto, pôde-se tomar como inspiração o uso de um controlador para identificar o nível de água e acionar a bomba de acordo com a variação do nível, bem como utilizar sensores de nível no reservatório para interromper o funcionamento caso o mesmo esteja completo.

2.2.5 COEL - NI35W Controlador de Nível

A empresa COEL (2024) apresenta o controlador de nível NI35W, um produto desenvolvido para garantir a proteção de bombas e que pode ser aplicado em reservatórios ou poços.

Figura 2.1 – Controlador de nível NI35W



Fonte: COEL (2024).

Segundo a empresa, o funcionamento desta solução é baseado em eletrodos instalados no reservatório/poço que, a depender do nível de água, ativa ou desativa o Relé em que está conectado à bomba. O produto dispõe de 3 tipos de eletrodos: Es (superior), Ei (inferior) e Er (referencial). Com o auxílio dos eletrodos Es e Ei é possível identificar os níveis máximos e mínimos de água e, com o auxílio do eletrodo Er, é possível obter a referência do reservatório, devendo ser instalado um pouco abaixo do nível mínimo ou até na carcaça do reservatório, caso este seja condutor de eletricidade. A partir desta solução, pôde-se aproveitar para este projeto a ideia de controlar a bomba baseando-se em um controle de níveis do poço e utilizando um componente relé.

2.2.6 Thebe TPC-58

A empresa Thebe (2024) traz como uma solução para o controle de fluxo de água em bombas o controlador eletrônico TPC-58.

Figura 2.2 – Controlador Eletrônico TPC-58



Fonte: Thebe (2024).

A empresa afirma que este controlador pode ligar ou desligar a bomba automaticamente bem como a proteger contra sobrecargas de tensão e trabalho a seco. A empresa Biosementes (2024), que também comercializa este equipamento, afirma estes mesmos pontos e acrescenta que o controlador pode identificar quando há fluxo de água no sistema afim de acionar a bomba apenas quando há demanda, como por exemplo o acionamento de um irrigador que demanda o fluxo de água e, após um intervalo de tempo, interrompe o fluxo novamente, possibilitando desligar a bomba. Pôde-se utilizar como inspiração deste produto a ideia de proteger a bomba contra sobrecargas de tensão e trabalho a seco.

A seguir, é apresentada uma tabela resumindo a finalidade, os materiais e os softwares utilizados no desenvolvimento dos trabalhos correlatos e produtos comerciais realizando um comparativo com os materiais utilizados no projeto desenvolvido neste trabalho.

Trabalho	Aplicação	Materiais	Softwares
Trab 2.2.1	Controle de uma porta e medição de temperatura	Eletrôímã, sensor DHT11 e Arduino Ameba	AWS Lambda, AWS IoT, Alexa e servidor da Thingspeak
Trab 2.2.2	Automação de 6 lâmpadas	6 lâmpadas, 6 servo motores, interruptores e o microcontrolador Esp8266	Página WEB e servidor local no microcontrolador
Trab 2.2.3	Automação residencial para uma casa com garagem e jardim	CLP, 4 Lâmpadas, irrigadores de planta, portão, caixa d'água, sensores infravermelhos, relés fotoelétricos, botão de emergência, motor, sensores de proximidade indutivos, sensor óptico, emissor e receptor infravermelho e 6 sensores de nível	<i>FactoryTalk View Studio</i>
Trab 2.2.4	Controle Autônomo de acionamento e velocidade de bombas	Circuito integrado 4011, Relé, transistor, sensores de nível, <i>Dimmer Triac</i> e placa de fenolite perfurada	Não se aplica
Trab 2.2.5	Controle Autônomo de bombas	Relé e Eletrodos	Não se aplica
Trab 2.2.6	Controle Autônomo de bombas	controlador eletrônico	Não se aplica
Este trabalho	Automação de uma bomba d'água	sensor SCT013, módulo sensor ZMPT101B, relés e microcontrolador ESP32	<i>Blynk</i>

Tabela 1 – Quadro Resumo dos Trabalhos Correlatos

2.3 Microcontroladores

Oliveira (2017b) explicita em seu livro que os microcontroladores tiveram incremento em seu uso no período que a Intel lançou seus microprocessadores de baixo custo

e suas concorrentes lançaram os microcontroladores, especialmente a Texas Instruments com o microcontrolador TMS 1000. Com isso, foi possível proceder com a substituição de tarefas manuais por sistemas automatizados. Ainda segundo o autor, esses dispositivos eram conhecidos como *SoC (System-on-Chip)* por carregar as funcionalidades no próprio componente.

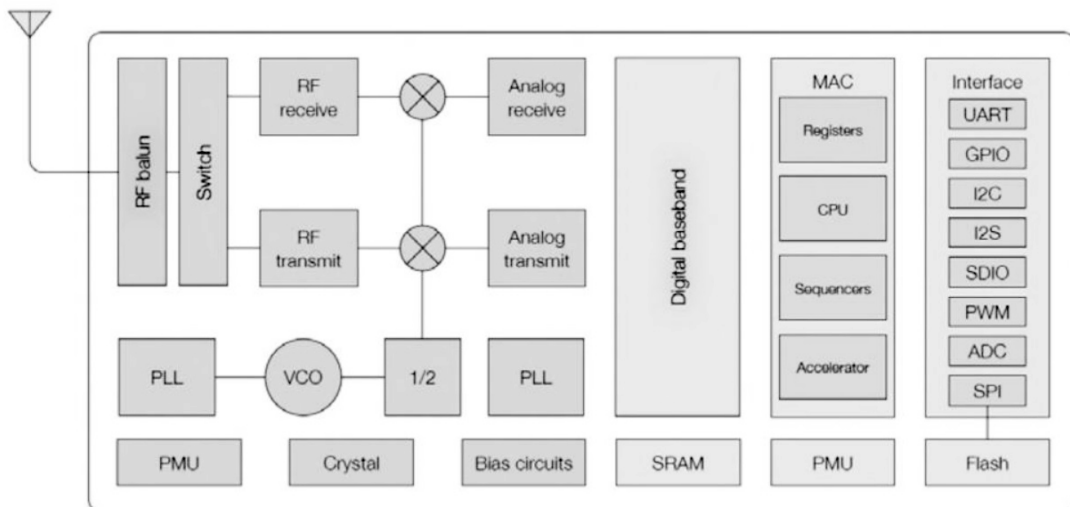
Oliveira (2017b) também diz que a diferença entre um microcontrolador e um microprocessador de uso geral, é que o microcontrolador geralmente é usado com dispositivos elétricos como botoeiras e relés, já os microprocessadores são comumente utilizados em computadores com uso de periféricos como teclado e *mouse*. E, com isso, os microcontroladores começaram inicialmente a ganhar projeção na indústria em diversas áreas de atuação.

Ainda de acordo com o autor, os microcontroladores inicialmente se limitavam a interfaces de entrada e saída e, a cada versão, foram incorporando novas funcionalidades como memória *RAM (Random Access memory)*, memória *EPROM (Erasable Programmable Read-Only Memory)*, interfaces de comunicação serial e *USB (Universal Serial Bus)* e, mais recentemente, as tecnologias de rede como *Ethernet, wi-fi (Wireless Fidelity)* e *bluetooth*.

Oliveira (2017b) traz como uma grande vantagem dos microcontroladores a sua capacidade de aplicação em diferentes contextos aliada com seu baixo consumo de energia, sendo suficiente apenas uma bateria para mantê-lo energizado por muito tempo.

A plataforma de controle a ser utilizada neste trabalho será a NodeMCU V1 que utiliza o microcontrolador ESP8266 da Espressif. Oliveira (2017b) apresenta em seu livro o seguinte diagrama funcional deste microcontrolador, o qual contém o conjunto completo de componentes presentes no circuito.

Figura 2.3 – Diagrama Funcional do Microcontrolador ESP8266



Fonte: Oliveira (2017b, p. 41).

3 METODOLOGIA

Tendo em vista que a natureza do projeto é em sua grande parte aplicada, o procedimento que será seguido para a obtenção dos resultados dos objetivos deste trabalho é o seguinte:

Primeiramente, foram pesquisados os conceitos de automação em geral e automação residencial em sites e fóruns que discutem o assunto mais profundamente.

Os próximos passos foram estudar as necessidades do projeto para identificar exatamente em quais pontos atuar. Posteriormente, foram estudadas como essas necessidades podem ser atendidas.

Com essa etapa realizada, avaliou-se ferramentas que poderiam ser utilizadas baseando-se em pesquisas e projetos semelhantes. Tais ferramentas deveriam abranger o controle e o monitoramento remoto do sistema. Após isso, foram estudados em artigos e sites especializados os sensores a serem utilizados para verificar a corrente da bomba e a tensão da rede de energia.

Após a definição das ferramentas e dos sensores, deu-se início ao desenvolvimento do circuito eletrônico responsável por realizar o controle da bomba elétrica. O desenvolvimento baseou-se em pesquisas prévias e testes práticos. Esse circuito deve ser capaz de identificar a corrente de operação da bomba em tempo real e verificar se está dentro dos limites pré definidos já citados e, se não estiver, o funcionamento deve ser interrompido. Paralelamente, esse circuito também deve ser capaz de enviar tais informações para um sistema supervisorio que possa ser acessado remotamente. Por último, o acionamento da bomba a partir deste circuito deve ser feito automática ou manualmente. O acionamento automático é baseado nas informações obtidas pelos sensores de corrente e tensão. E o acionamento manual é realizado por meio do sistema supervisorio e fisicamente a partir de botões.

Ao final do desenvolvimento, os circuitos devem ser integrados em um único microcontrolador para que seja desenvolvido um protótipo funcional em uma protoboard. O objetivo da criação deste protótipo é possibilitar uma futura instalação do dispositivo no condomínio que repassou a demanda com o fim de realizar testes de funcionamento e a validação do circuito em cenários reais.

O último passo deste trabalho é desenvolver o projeto de confecção de uma placa de circuito impresso para abarcar o sistema em um dispositivo que será mais seguro, mais eficiente e mais compacto que o protótipo.

3.1 Materiais e Métodos

Com vista às necessidades dos moradores do condomínio referenciadas anteriormente, foi desenvolvido um dispositivo de automação para coletar os dados da rede de

energia e da bomba com o objetivo de realizar o acionamento seguro deste equipamento e enviar tais dados para um sistema supervisório online. Esse sistema é autônomo e realiza o controle da bomba durante as 24 (vinte e quatro) horas do dia. Por meio do sistema supervisório, também é possível acionar a bomba manualmente à distância.

3.2 Protótipo Desenvolvido

Primeiramente, houve uma discussão para apurar o que seria necessário para a realização do projeto. Com isso, chegou-se à conclusão de que seria necessário um sensor de tensão, para verificar constantemente a tensão da rede, um sensor de corrente, para verificar a corrente de operação da bomba, um interruptor eletrônico, para acionar a bomba remotamente, e um microcontrolador, para realizar o controle do sistema.

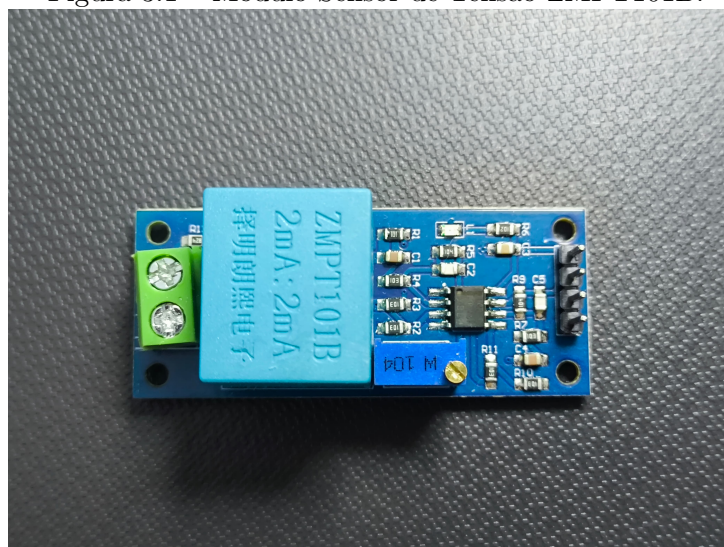
Um conjunto de opções passou por avaliação para cada componente com o objetivo de reunir informações sobre as possibilidades de escolha e decidir qual se encaixaria melhor nos requisitos do projeto.

Com base nas informações levantadas, foram selecionados os seguintes componentes com as respectivas justificativas:

3.2.1 Sensor De Tensão

A principal opção analisada, com base em projetos anteriores e na discussão recorrente entre especialistas da área, foi o módulo sensor de tensão AC 0-250V ZMPT101B. Este sensor foi escolhido por fornecer um sinal preciso e sensível a pequenas variações de tensão, além de dispôr de um módulo pré fabricado, o que garante facilidade na utilização.

Figura 3.1 – Módulo Sensor de Tensão ZMPT101B.



Fonte: Próprio Autor.

Este módulo é capaz de fornecer o valor digital equivalente à tensão medida na rede elétrica. A partir disso, é possível identificar variações da tensão sendo para mais

ou para menos dos 220V habituais da rede elétrica residencial. Como o projeto necessita, obrigatoriamente, identificar com precisão essas variações para o correto funcionamento, este sensor foi avaliado como o mais adequado.

A programação para o módulo sensor de tensão ZMPT101B torna-se relativamente mais complexa por exigir cálculos matemáticos que transformam o sinal senoidal de entrada em um valor digital que pode ser interpretado pelo microcontrolador. Porém, para facilitar este processo, optou-se pelo uso de uma biblioteca externa (descrita na Sec. 3.6) que realiza tais cálculos automaticamente e ainda disponibiliza métodos de calibração do sensor, o qual, para ser calibrado, basta comparar o valor resultante do sensor com uma leitura conhecida do sinal de entrada, como por exemplo a leitura de um multímetro. A vantagem da escolha da biblioteca é que a programação do sensor pode ser resumida em poucas linhas de código quando comparada à programação sem o uso da mesma.

3.2.2 *Sensor De Corrente SCT-013-000*

Este sensor foi escolhido por sua característica não intrusiva, ou seja, não é necessário romper o cabo do equipamento para inserí-lo no sensor, sendo suficiente apenas posicionar a pinça do sensor em um dos fios do equipamento a ser medido, o que simplifica a instalação. Este sensor também foi escolhido por realizar medições de até 100A (cem *ampères*), o que garante uma boa margem de segurança para o acionamento da bomba. Outro fator que influenciou a escolha deste sensor foi a sensibilidade, permitindo fornecer sinais com resolução de até 0,0122mA, calculada a partir da divisão entre a saída máxima de tensão do sensor (50mA) e a quantidade de níveis de leitura da ESP32 (4096 níveis para 12 bits).

Figura 3.2 – Sensor de Corrente SCT013.

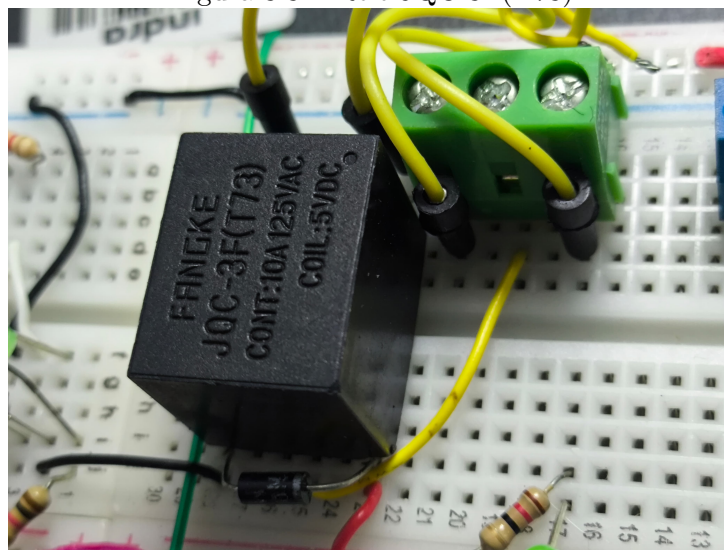


Fonte: Próprio Autor.

3.2.3 *Interruptor eletrônico - Relé*

O interruptor eletrônico foi escolhido para realizar o acionamento eletronicamente da bomba d'água, conectando-a ao interruptor. A partir disso, basta enviar um sinal de tensão ao relé para acioná-lo, o que irá inverter o estado da bomba de ligada para desligada ou vice-versa, a depender da configuração prévia do relé, que pode estar configurado em normalmente aberto ou normalmente fechado. Com este componente e o restante de seu circuito, é possível ainda garantir o isolamento entre o sistema de controle do dispositivo e a alta tensão da rede elétrica, que poderia danificar o microcontrolador em casos de sobrecargas de energia.

Figura 3.3 – Relé JQC-3F(T73).



Fonte: Próprio Autor.

3.2.4 *Display LCD com módulo I2C*

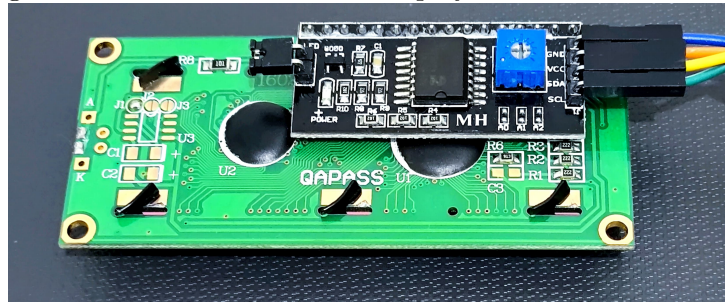
O display LCD foi escolhido para apresentar as informações do sistema localmente para o usuário de forma facilitada caso não seja possível acessar a plataforma online.

Figura 3.4 – Vista Frontal do Display LCD com Módulo I2C.



Fonte: Próprio Autor.

Figura 3.5 – Vista Traseira do Display LCD com Módulo I2C.



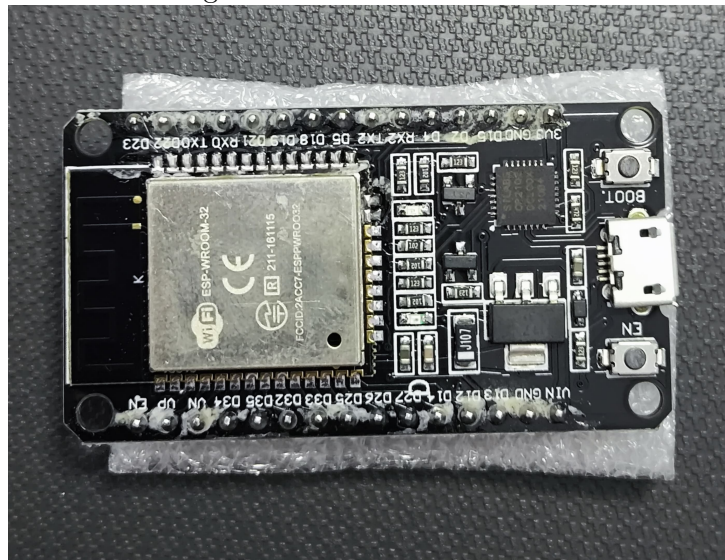
Fonte: Próprio Autor.

Este display em específico foi escolhido por possuir um módulo e PCF8574/74A com um protocolo de comunicação I2C (*Inter-Integrated Circuit*), o que possibilita consumir menos portas digitais do microcontrolador, assim como será explicado a seguir.

3.2.5 ESP32

A escolha deste microcontrolador se deu em função de possuir uma simples operação e ter apresentado diversas funcionalidades úteis. Como exemplo dessas funcionalidades, podemos citar sua característica de possuir *Wi-Fi* embutido à placa, possuir bluetooth, variados tipos de portas e também possuir bastante compatibilidade com outras plataformas, como o Arduíno, o que expande o acervo literário acerca de sua programação.

Figura 3.6 – ESP-WROOM-32.



Fonte: Próprio Autor.

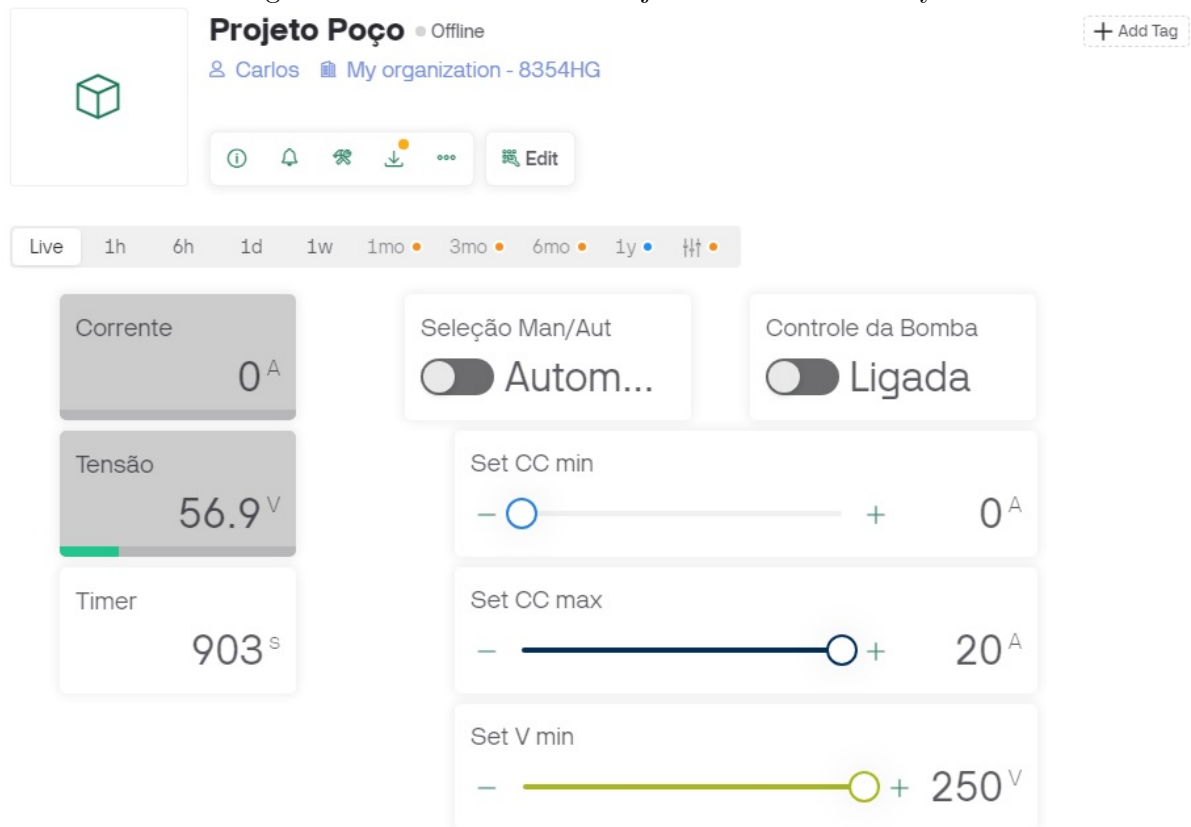
3.2.6 Componentes Auxiliares

Foram escolhidos componentes auxiliares como *LEDs*, resistores, capacitores, *protoboard* e cabos de acordo com a demanda do dispositivo e de outros componentes.

3.2.7 Blynk

Esta plataforma foi escolhida por oferecer diversos recursos a projetos voltados à Internet das Coisas - *Internet of Things* (IoT). Seu plano gratuito atendeu às necessidade do projeto oferecendo a quantidade suficiente de botões e *displays* para acionamento do sistema e leitura dos sensores e possibilitou a integração do dispositivo com a plataforma em nuvem disponibilizada pela empresa.

Figura 3.7 – Tela Inicial do Projeto na Plataforma Blynk.



Fonte: Próprio Autor.

O protótipo contém uma placa *proto-board* onde foram montados todos os componentes necessários. Contém também um microcontrolador ESP32, um relé, um sensor de corrente, um sensor de tensão, um capacitor, resistores, *LEDs*, transistor, opto-acoplador, botões, conectores e uma fonte de alimentação.

3.3 Detalhes de Construção

Após algumas modificações e melhorias, o projeto do dispositivo foi finalizado com a seguinte configuração, de acordo com os componentes apresentados anteriormente:

O dispositivo contém um sensor de corrente, um sensor de tensão, um relé, um display LCD com módulo I2C, dois botões e um microcontrolador, além dos componentes

auxiliares, como resistores, capacitores, transistores, opto-acopladores e bornes.

Para a validação experimental, o circuito completo foi montado em uma *proto-board* a fim de reproduzir fielmente o circuito previamente simulado no *software Proteus*. O uso da *proto-board* permitiu testar em um ambiente controlado, garantindo a flexibilidade necessária para realizar ajustes e modificações rápidas no *hardware*. Dessa forma, foi possível verificar se o comportamento do circuito físico correspondia ao que havia sido projetado e simulado. Os testes práticos abrangeram a análise detalhada do desempenho dos componentes, assegurando que as respostas fossem condizentes com as expectativas. Além disso, foram feitas medições de corrente e tensão em pontos críticos do circuito para confirmar o funcionamento adequado dos sensores e atuadores, observando possíveis variações ou perdas no sistema real em comparação ao modelo teórico, assim como descrito ao longo deste trabalho e principalmente no Capítulo 4.

Partindo para uma descrição detalhada dos componentes utilizados, primeiramente há o sensor de corrente SCT-013, representado na Figura 3.2. A finalidade deste sensor é identificar, em tempo real, a corrente de operação da bomba d'água. Essa leitura é necessária pois, quando há água no poço em que está alocada, a bomba realiza maior trabalho, exigindo mais corrente para movimentar a carga. E, quando o poço está parcialmente ou totalmente sem água, a bomba opera à vazia, exigindo menos corrente, porém com risco de superaquecimento por funcionamento sem carga.

Esse comportamento pode ser explicado pela seguinte equação:

$$P = I \times V \quad (3.1)$$

onde:

- P é a potência,
- I é a corrente,
- V é a tensão.

Ao variar a carga, varia-se também a potência gerada pelo trabalho realizado pela bomba. De acordo com a equação Equação (3.1), percebemos que a corrente é diretamente proporcional à potência, já que a tensão da rede elétrica tende a se manter constante em 220V. Por esse motivo é possível identificar variações do nível de água baseando-se na corrente elétrica.

A partir disso, um intervalo de corrente de operação aceitável da bomba é determinado com base no manual do fabricante ou por medições práticas e, em seguida, configurado no firmware do dispositivo ou no sistema supervisor. Para este projeto, este intervalo de corrente foi definido a partir de medições práticas do sistema em funcionamento com alto nível de água e com baixo nível de água, obtendo os seguintes valores: 5,6A quando está sem carga e 7,2A quando está com carga.

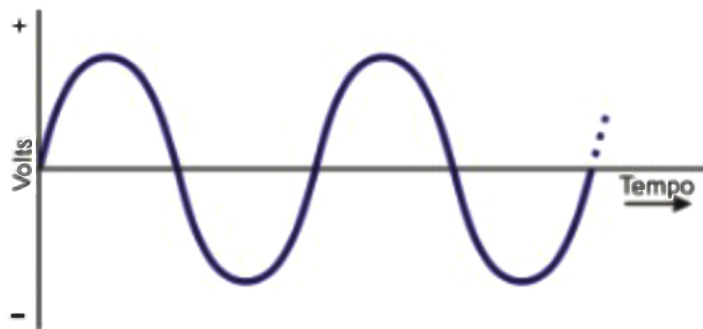
Partindo para a aplicação do sensor SCT013 no sistema, Demetras (2017) traz em seu artigo a forma correta da implementação. Segundo o autor, este sensor tem a vantagem

de não ser invasivo, ou seja, não obriga realizar um corte nos cabos a serem lidos, pois ele faz uso das propriedades magnéticas da corrente elétrica, bastando apenas fixar a presilha do sensor ao cabo.

Ainda de acordo com Demetras (2017), no momento em que há corrente elétrica no cabo de prova, uma corrente elétrica proporcional é induzida no sensor, enviando essa informação para o microcontrolador.

Porém, para que seja possível realizar a leitura do sensor com o microcontrolador, é necessário um circuito elétrico intermediário a fim de transformar a corrente em um sinal de tensão com um intervalo definido. Para isso, conforme descrito pelo autor, é adicionado um resistor de carga entre os terminais do sensor e um divisor de tensão formado por dois resistores de $10\text{k}\Omega$ para somar o valor do *offset* (deslocamento vertical do sinal senoidal) e alterar para $2,5\text{V}$. Dessa forma, a variação do sinal do sensor de corrente estará sempre entre 0 e 5V pois a amplitude do sinal é 5V e a oscilação é de $2,5\text{V}$, respeitado o intervalo de leitura do microcontrolador que também é de 0 a 5V .

Figura 3.8 – Sinais senoidais sem *offset* e com *offset*



Forma de onda da tensão no resistor de carga



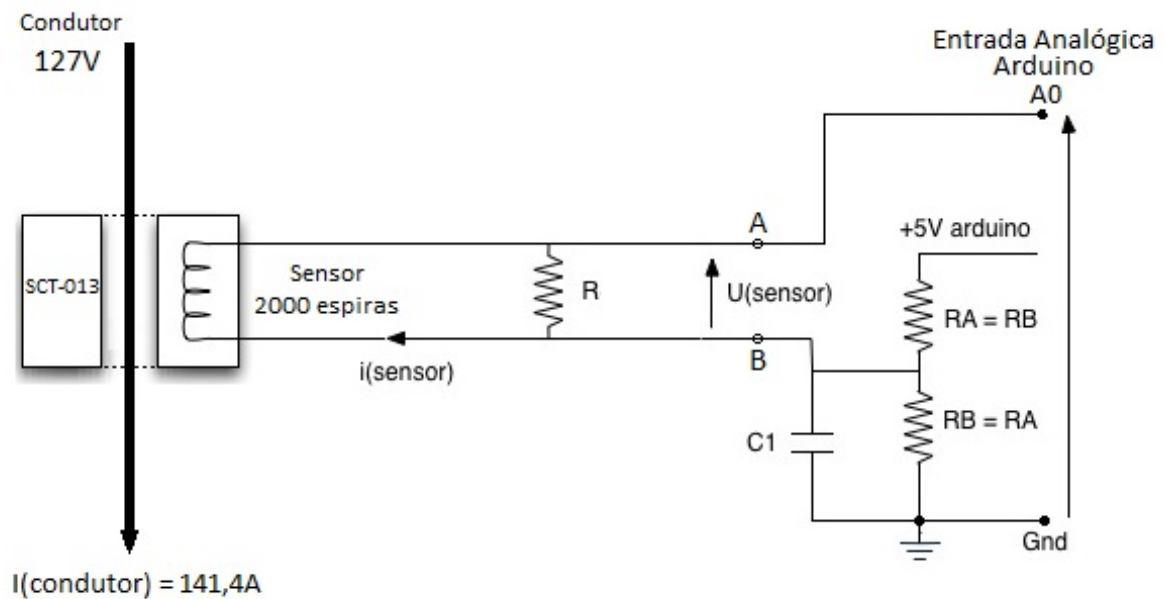
Forma de onda com *offset*

Fonte: (DEMETRAS, 2017)

Também foi adicionado um capacitor de $10\mu\text{F}$ para estabilizar o sinal. Sendo

assim, esse é o circuito elétrico final que foi utilizado para o sensor de corrente:

Figura 3.9 – Circuito elétrico do sensor de corrente SCT-013

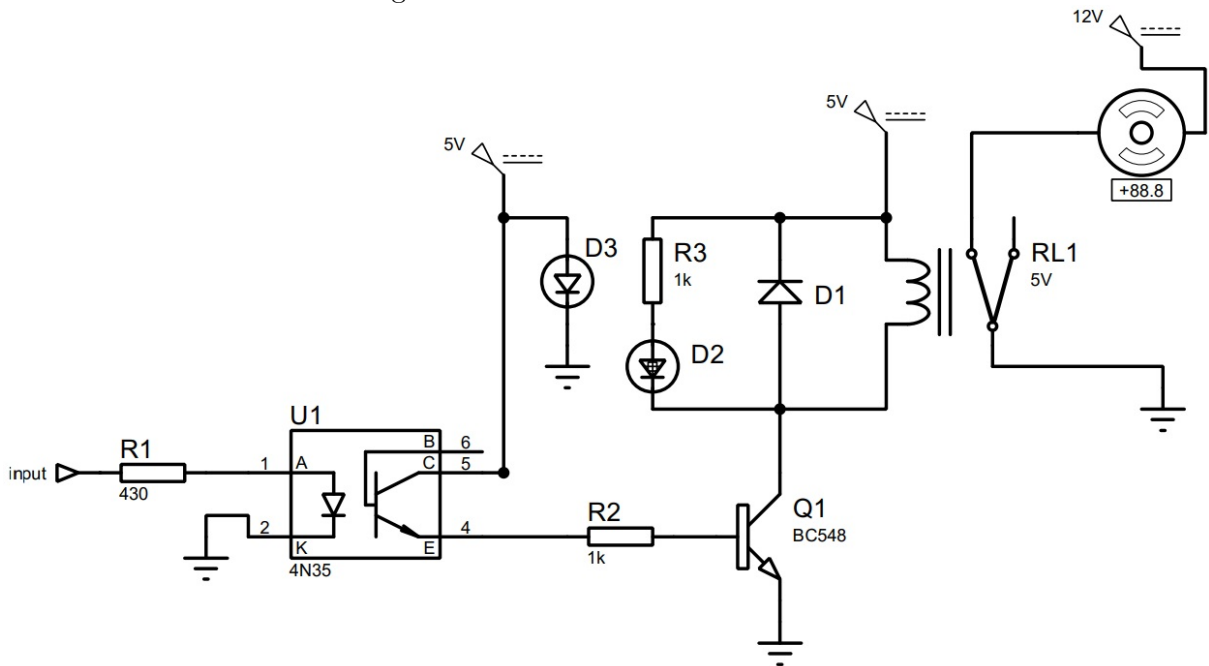


Fonte: (DEMETRAS, 2017)

Para medir a tensão, foi utilizado o módulo externo ZMPT101B apresentado na Fig. 3.1. Com o circuito eletrônico já pronto, basta realizar a conexão ao microcontrolador e fixar o sensor na placa do dispositivo. Esse sensor é responsável pela leitura em tempo real da rede elétrica onde a bomba está conectada, identificando oscilações e quedas de tensão que possam danificá-la. Em tais situações, o sistema atua desativando a bomba, prevenindo possíveis danos. O valor mínimo de tensão aceitável para a operação da bomba também pode ser definido diretamente no *firmware* do dispositivo ou no sistema supervisor. Para este trabalho, o valor mínimo aceitável de tensão, definido a partir de medições práticas, foi de 200V.

Para o relé, assim como o sensor de corrente, decidiu-se implementar o circuito elétrico diretamente na placa do dispositivo. O projeto deste circuito foi baseado no circuito proposto pela empresa (TECDICAS, 2019).

Figura 3.10 – Circuito elétrico do relé.



Fonte: Próprio Autor.

Neste esquema, o relé é acionado quando o transistor (Q1) também é acionado. Há um *LED* (D2) para sinalizar quando o relé está acionado e há um diodo (D1) para impedir que a corrente induzida pelo relé retorne para o circuito. Também há um *LED* (D3) para informar que o sistema do relé está alimentado com a tensão de 5V. E, para finalizar, há um opto-acoplador (U1) que irá receber o sinal do microcontrolador e acionar o transistor.

Segundo RASHID (2003), o opto-acoplador é utilizado principalmente para isolar sinais. E, ainda de acordo com o autor, o opto-acoplador é um componente que possui internamente um *LED* emissor de luz, o qual receberá a tensão de entrada, e um fototransistor que é ativado quando este *LED* emite luz, permitindo a passagem de tensão de uma fonte externa para a saída do componente. Esse fato garante o isolamento entre a rede elétrica de entrada e a rede elétrica de saída, já que as duas não estão em contato direto. Esse isolamento é essencial para que possíveis sobrecargas de energia sofridas pelo relé não afete o circuito do microcontrolador. Neste projeto serão utilizados dois relés com circuitos independentes, um principal e um de backup para o caso de ocorrer falha no primeiro relé.

Outro componente utilizado no dispositivo foi o display LCD (Fig. 3.4). Com ele é possível apresentar as informações diretamente no dispositivo, aumentando a operabilidade do mesmo.

Este display, especificamente, possui 16 (dezesesseis) colunas e 2 (duas) linhas, ou seja, é um display 16x2. Normalmente, o display LCD consome um grande número de portas digitais do microcontrolador. Barbacena e Fleury (1996) trazem em seu artigo que para um display 16x2, são necessários de 14 a 16 pinos, incluindo os pinos de alimentação

e da luz de *backlight*. Essa característica acaba sendo desvantajosa para o projeto pois muitas portas digitais seriam demandadas apenas para o display, podendo limitar o uso de outros componentes. Por esse motivo, optou-se pela utilização do módulo I2C.

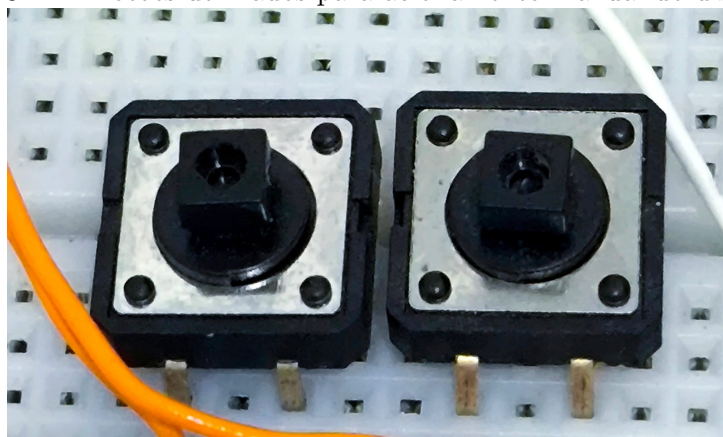
De acordo com Mankar *et al.* (2014), o I2C é um protocolo que permite uma fácil comunicação e sem perda de dados. Ainda segundo o autor, a vantagem desta interface de comunicação é utilizar apenas dois fios para comunicação e um para o aterramento. Os dois fios de comunicação ativa são compostos pelo *Serial Clock (SCL)* e o *Serial Data (SDL)*. Estes dois fios são bidirecionais e *half duplex*, ou seja, levam a informação entre os dispositivos conectados ao barramento. Mankar *et al.* (2014) diz ainda que a relação entre os dispositivos são de mestre-escravo, sendo o microcontrolador o mestre e o dispositivo com o módulo I2C o escravo.

O módulo e PCF8574/74A faz uso deste protocolo para seu funcionamento. Esta é a descrição do componente de acordo com o datasheet do fabricante: "*The PCF8574/74A provides general-purpose remote I/O expansion via the two-wire bidirectional I2C-bus (serial clock (SCL), serial data (SDA)).* (SEMICONDUCTORS, 2013)". Em um breve resumo, essa descrição diz que o módulo tem um uso geral por meio de barramento I2C bidirecional de dois fios.

Dessa forma, com a utilização do módulo I2C, a grande quantidade de portas demandadas pelo LCD pode ser substituída por apenas duas portas de sinais.

Também foram utilizados dois botões para realizar o controle manual do dispositivo.

Figura 3.11 – Botões utilizados para acionamento manual do dispositivo.



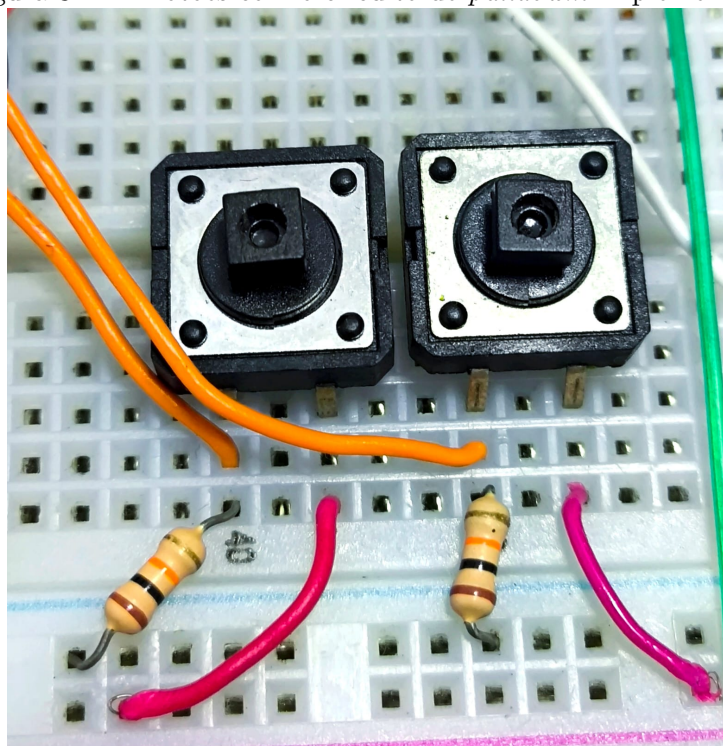
Fonte: Próprio Autor.

Porém, assim como descrito por Mariano (2021) em sua monografia, os contatos mecânicos do botão geram ruídos aleatórios fazendo com que a leitura oscile, em um intervalo de microssegundos, entre o nível alto e baixo, o que caracteriza o efeito *bounce*. Este efeito é prejudicial pelo fato de induzir o microcontrolador a interpretar um sinal que não reflete o estado real do botão.

Portanto, um dos métodos citados pelo autor para mitigar esse efeito é o método *pulldown*, que consiste em inserir uma tensão positiva em um terminal do botão e aterrar

o outro terminal passando por uma resistência, enquanto a leitura do sinal pelo microcontrolador é realizada no terminal aterrado do botão. Dessa forma, quando o botão não está pressionado, o valor lógico é 0 (zero) e, quando o botão está pressionado, o valor lógico é 1 (um).

Figura 3.12 – Botões com o circuito de *pulldown* implementado.



Fonte: Próprio Autor.

O fio vermelho está ligado à alimentação de 5V do circuito, enquanto o resistor está retornando para o aterramento. O fio laranja, conectado ao terminal negativo do botão, está conectado ao microcontrolador. Este foi o método escolhido para implementar os botões neste projetos sem a incidência do efeito *bounce*. Neste exemplo foram utilizados dois botões, porém, assim como citado anteriormente, serão dois interruptores eletrônicos a serem controlados, exigindo dois botões, além do botão de seleção entre o estado manual ou automático. Sendo assim, serão no total 3 botões.

Por fim, o último componente utilizado foi o software *Blynk*. De acordo com a própria empresa em seu site oficial Blynk Inc. (2024), a *Blynk* é uma plataforma em nuvem voltada para o desenvolvimento *low-code* de tecnologias *IoT* (*Internet of Things*) com foco na experiência do usuário, ou seja, exige pouca codificação para criar aplicações IoT com interfaces amigáveis aos usuários.

Esta ferramenta foi escolhida por fornecer a possibilidade de criar um *dashboard* completo e realizar a integração com o microcontrolador a partir de uma biblioteca, tornando o processo de criação de um sistema supervisorio mais simples.

No *Blynk*, é possível inserir botões, displays, entradas de valores, entre outros. Para este projeto, foram utilizados botões, medidores gráficos, rótulos e botões *sliders*. Como

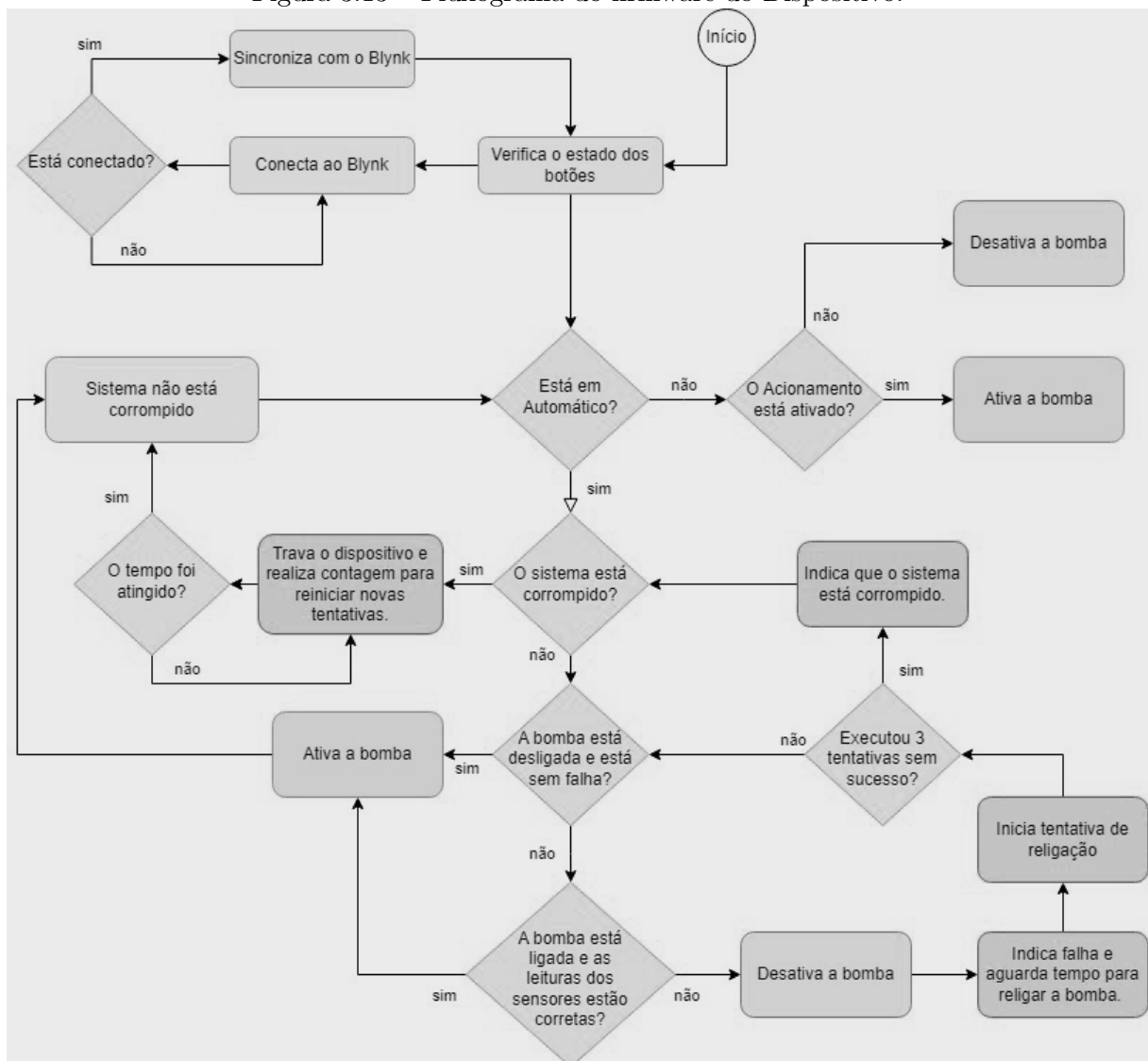
exemplo, há a Figura 3.7 que apresenta um *Web Dashboard* (utilizado em computadores) e há a Figura 3.16 que apresenta um *dashboard mobile* (utilizado em *smartphones*).

Com isso, estes são os componentes presentes no dispositivo deste projeto.

3.4 Fluxograma do Firmware

A figura a seguir representa a sequência lógica utilizada para a programação do dispositivo a partir de um fluxograma. Neste fluxo, é possível identificar os passos que o dispositivo segue desde o momento em que é energizado até o acionamento automático da bomba.

Figura 3.13 – Fluxograma do firmware do Dispositivo.

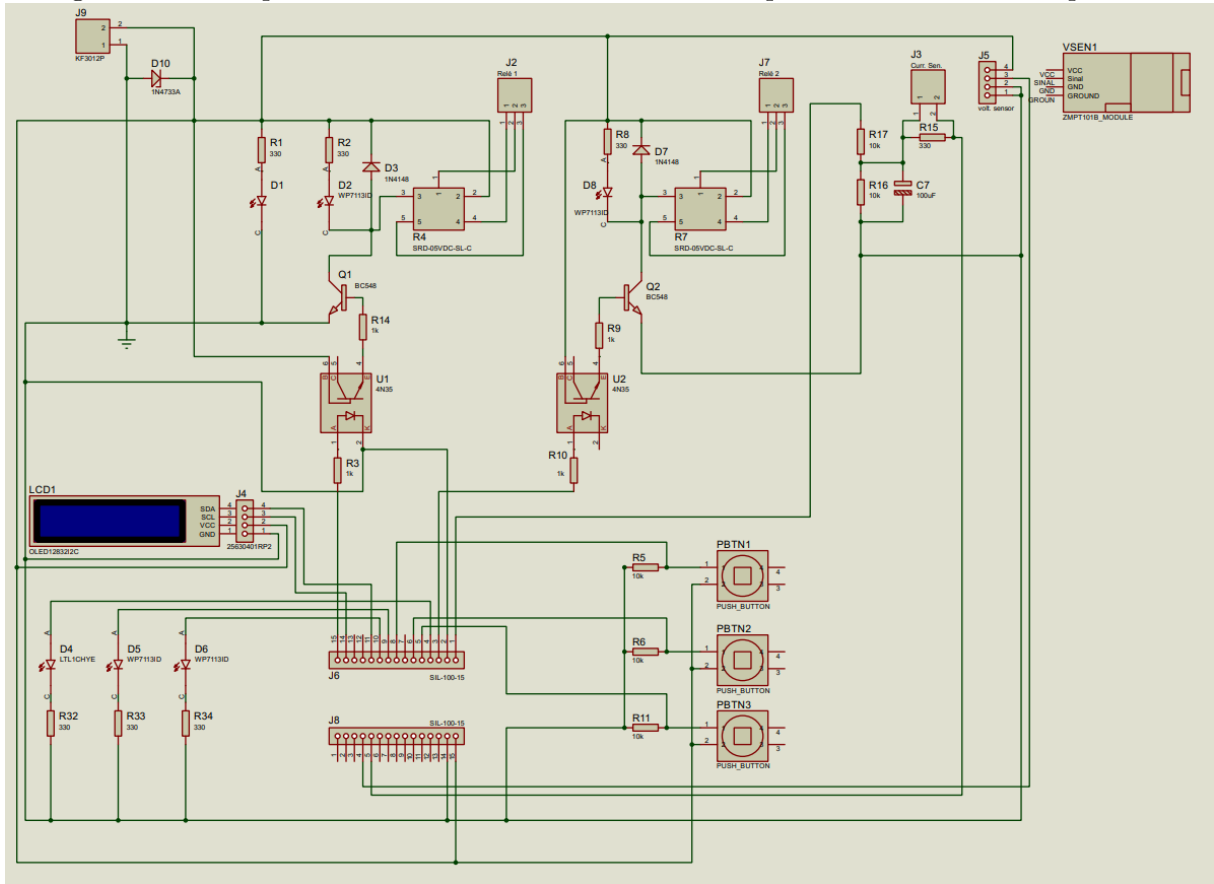


Fonte: Próprio Autor.

3.5 Esquema Elétrico

Conforme descrito na Sec. 3.3, este é o circuito elétrico para a versão 3 do dispositivo contendo todos os componentes citados:

Figura 3.14 – Esquema elétrico desenvolvido no Proteus para a versão 3.0 do dispositivo.



Fonte: Próprio Autor.

Vale ressaltar que o dispositivo foi projetado para ser compatível com as fontes de alimentação de 5V comumente utilizadas em *smartphones*. E, como forma de proteção, foi inserido um diodo zener (D10) que, de acordo com Technology (2024), pode ser utilizado para proteger circuitos contra sobrecargas de tensão e contra descargas eletrostáticas devido a sua característica de manter a tensão em um valor específico quando a tensão de entrada supera o valor de tensão *threshold*, mais conhecido como tensão de zener.

Além dos componentes já citados, podemos destacar também os três *LEDs* (*Light Emitter Diodes*) de avisos visuais (D4, D5 e D6). Com tais *LEDs* é possível repassar uma informação visual do *status* do dispositivo que abrange o funcionamento, se está operando ou se está em falha, e a conexão com o *Wi-Fi*, se está conectado ou não.

3.6 Descrição do Firmware do Dispositivo

O firmware do dispositivo foi construído totalmente na IDE (Integrated Development Environment) do Arduíno que, segundo o que a empresa diz em seu site oficial, é uma ferramenta *open-source* que torna fácil escrever código e fazer o *upload* para as placas.

Também foram utilizadas bibliotecas externas para a programação de alguns componentes por fornecerem as funcionalidades desejadas previamente desenvolvidas por outros autores.

As bibliotecas utilizadas são:

- **Blynk:**

Biblioteca oficial da plataforma Blynk. Está disponível diretamente no gerenciador de bibliotecas da IDE do Arduíno com o nome "*Blynk*" escrita por Volodymyr Shymanskyy.

- **LiquidCrystal_I2C**

É uma biblioteca derivada da biblioteca LiquidCrystal do Arduíno. De acordo com os criadores, essa biblioteca é quase 5 (cinco) vezes mais rápida que a biblioteca do Arduíno. Está disponível em um repositório público do *GitHub* (Hospedeiro de Códigos) chamado "New-LiquidCrystal" criado pelo usuário "fmalpartida".

- **EmonLib**

Biblioteca para monitoramento de eletricidade. Disponível também em um repositório público do *GitHub* chamado "EmonLib" criado pelo usuário "openenergymonitor".

- **ZMPT101B-arduino**

Biblioteca criada para interagir com o módulo sensor de tensão ZMPT101B. Baseada na biblioteca Arduíno para sensores de corrente ACS712 de Ruslan Koptev. Disponível também em um repositório público do *GitHub* chamado "ZMPT101B-arduino" criado pelo usuário "Abdurraziq".

A importação das bibliotecas é representada na seguinte figura:

Figura 3.15 – Bibliotecas utilizadas no firmware.

```
//=====
//Inclusão de Bibliotecas

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "EmonLib.h"
#include <ZMPT101B.h>
```

Fonte: Próprio Autor.

O código-fonte completo está presente no Apêndice A. Este código contém toda a lógica utilizada para o dispositivo de forma detalhada e documentada.

O objetivo principal do *firmware* é coletar os dados recebidos dos sensores e identificar o estado dos botões. A partir dos botoes, é possível selecionar o tipo de operação

do equipamento entre manual e automático e acionar os relés que, conseqüentemente, irá acionar a bomba.

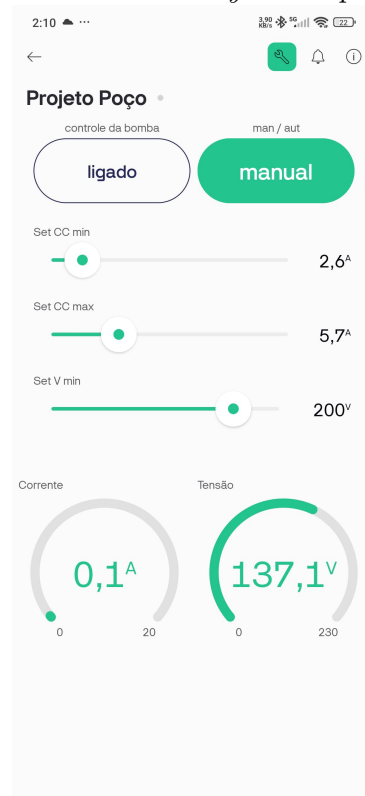
Existem dois tipos de botões, os botões físicos e os botões virtuais do *Blynk*. É possível controlar o dispositivo independentemente por qualquer uma das duas formas.

O controle manual, como o próprio nome sugere, permite ao usuário controlar o acionamento manualmente, ou seja, ativando ou desativando o diretamente o botão.

Já o controle automático irá realizar a leitura dos sensores corrente e tensão e verificar se está dentro do intervalo definido pelo usuário como o correto. Caso alguma leitura esteja fora do intervalo, o sistema irá autonomamente cortar a alimentação da bomba a partir do relé e aguardar um intervalo de tempo para realizar mais duas tentativas de reativação. Se, mesmo percorrido esse intervalo de espera o sistema não voltar à normalidade automaticamente, será informado que o sistema está corrompido e será necessária a intervenção humana para corrigir o problema. Caso essa intervenção não seja feita, o sistema realizará 3 novas tentativas de religação após um período maior de tempo para que o sistema não fique colapsado indefinidamente.

As informações dos sensores e dos botões estão disponíveis *online* na plataforma *Blynk* quando a mesma está sincronizada com o dispositivo por *Wi-Fi* assim como mostra a figura Fig. 3.7.

A plataforma *Blynk* também dispõe de um aplicativo móvel para *smartphones* chamado "*Blynk IoT*". Com isso, é possível acompanhar os dados do dispositivo tanto pelo computador quanto pelo celular.

Figura 3.16 – Interface do *Blynk* no aplicativo móvel.

Fonte: Próprio Autor.

A figura mostra que é possível definir, pela própria plataforma, os valores de corrente máxima e mínima e tensão mínima que serão respeitados quando o dispositivo estiver configurado como automático.

4 RESULTADOS E DISCUSSÃO

O fim deste trabalho é controlar uma bomba d'água elétrica autonomamente ou manualmente, de forma remota ou local. Este objetivo norteará os resultados aqui presentes.

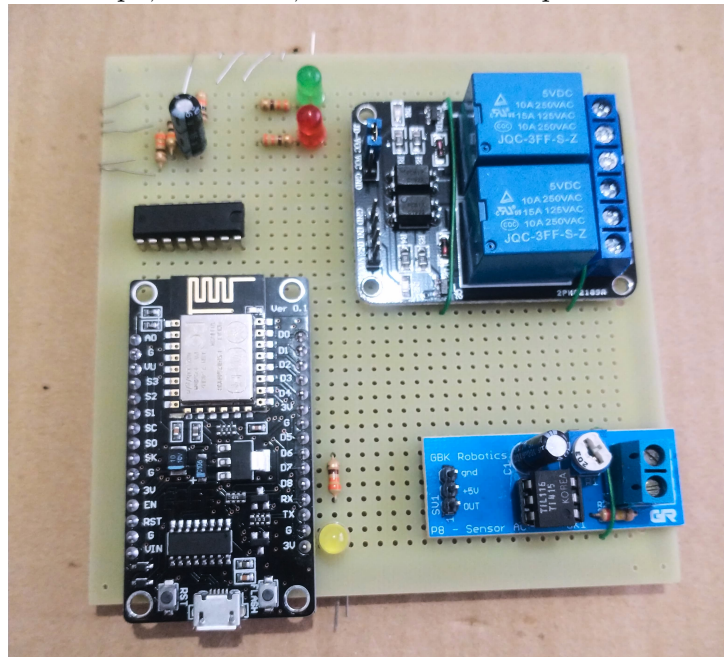
Partindo deste princípio, podemos detalhar o percurso até a versão atual do dispositivo construído, nomeado como versão 3.0.

O dispositivo passou por várias modificações antes de sua versão final. Serão apresentadas, cronologicamente, as principais versões com o fim de elucidar as alterações importantes e a contribuição das mesmas para a versão versão do projeto.

Primeiramente, foi desenvolvida uma versão, a 1.0, construída em uma placa de fenolite perfurada.

4.1 Protótipo versão 1.0

Figura 4.1 – Protótipo, versão 1.0, montado em uma placa de fenolite perfurada.



Fonte: Próprio Autor.

Esta placa de fenolite foi utilizada para alocar os componentes e soldá-los, com a conexões sendo feitas por meio de cabos. Nesta versão, foram utilizados os módulos relé de 2 vias e sensor de tensão baseado em opto-acoplador, o sensor de corrente, o multiplexador e o microcontrolador ESP8266.

O microcontrolador escolhido inicialmente foi o nodeMCU ESP8266. Esta plataforma foi escolhida por possui *Wi-Fi* embutido e por ter sido utilizada em projeto

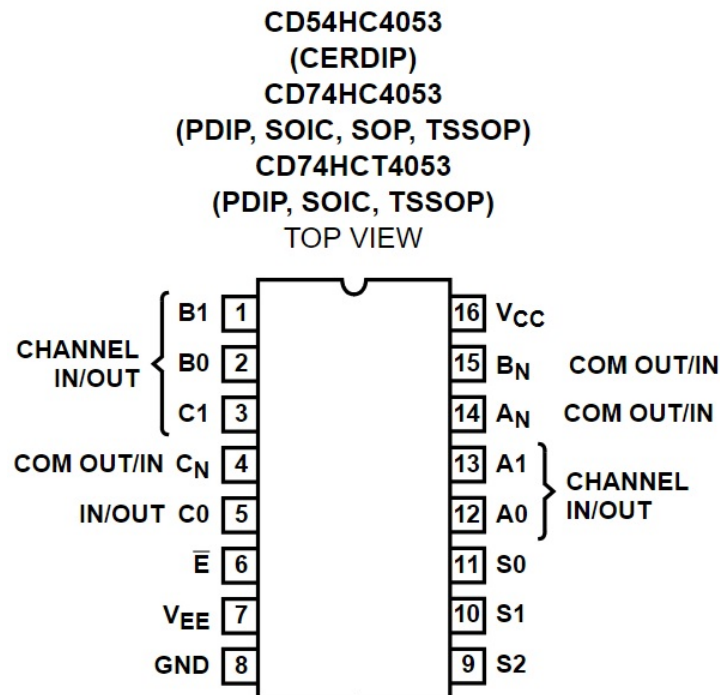
anteriores. Porém, a principal desvantagem deste microcontrolador é possuir apenas uma porta analógica, criando a necessidade do uso de um componente externo, o multiplexador CD74HC4053E, para a comutação entre os sinais analógicos dos dois sensores que deveriam ser recebidos na mesma porta analógica.

Nesta versão, optou-se também por um módulo eletrônico para cada componente, ou seja, uma placa de circuito eletrônico dedicada para cada componente que era fixada posteriormente no dispositivo, por exemplo, os sensores e os relés. Essa configuração não era a ideal por tornar o projeto dependente de outros fabricantes e pela ocupação de espaço dos módulos no circuito do dispositivo.

Outra característica importante desta versão do dispositivo, já citada anteriormente, é a utilização do multiplexador CD74HC4053E.

Este circuito integrado possibilita definir qual sinal de saída será lido entre dois sinais de entrada ativando os respectivos pinos de seleção. O CD74HC4053E, especificamente, possui três saídas e 6 entradas, 2 entradas para cada saída. Além disso, possui 3 pinos de seleção e os pinos de alimentação.

Figura 4.2 – CD74HC4053E - Descrição das portas



Fonte: Instruments (1997, p. 2).

Neste projeto, o multiplexador foi utilizado para realizar a comutação entre os sensores para a mesma porta do microcontrolador, visto que a placa controladora dispunha de apenas uma porta analógica. Para realizar este controle e selecionar qual sensor seria lido, foi escolhida a saída C e as entradas C0 e C1.

De acordo com a seguinte tabela verdade, podemos perceber que para ler a porta C0, bastava definir todos os valores zerados para S0, S1 e S2. E, para ler a porta S1,

bastava definir as portas S0 e S1 como zero e a porta S2 como 1.

Figura 4.3 – CD74HC4053E - Tabela Verdade

TRUTH TABLE
'HC4053, CD74HCT4053

INPUT STATES				"ON" CHANNELS
ENABLE	S ₀	S ₁	S ₂	
L	L	L	L	C0, B0, A0
L	H	L	L	C0, B0, A1
L	L	H	L	C0, B1, A0
L	H	H	L	C0, B1, A1
L	L	L	H	C1, B0, A0
L	H	L	H	C1, B0, A1
L	L	H	H	C1, B1, A0
L	H	H	H	C1, B1, A1
H	X	X	X	None

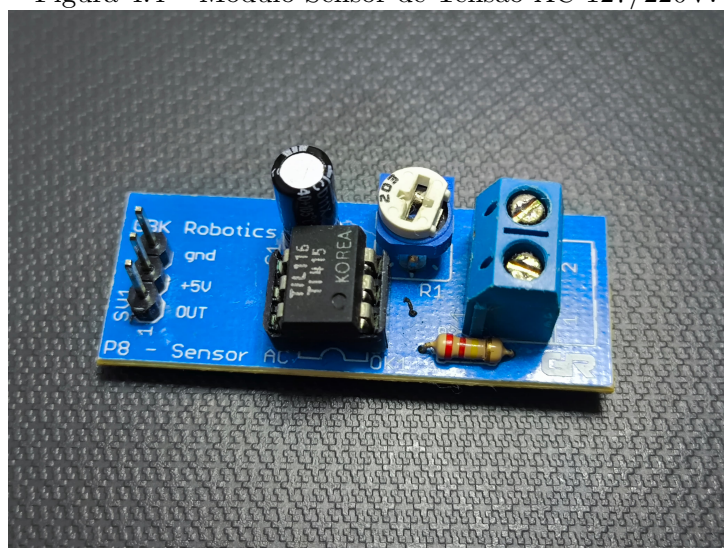
Fonte: Instruments (1997, p. 5).

Conforme descrito pelo *datasheet* (INSTRUMENTS, 1997), o multiplexador possui um tempo de comutação na escala de nano segundos, possibilitando um baixo tempo resposta para esta operação.

Contudo, este componente apresentou instabilidade de leituras, fornecendo valores de tensão mesmo quando nenhum sensor estava conectado. E, com os sensores conectados, apresentava variações que não aconteciam ao testar cada sensor individualmente. Não foi possível solucionar esse erro.

Já para o sensor de tensão, inicialmente, optou-se pelo módulo sensor de tensão AC 127/220V.

Figura 4.4 – Módulo Sensor de Tensão AC 127/220V.



Fonte: Próprio Autor.

Este sensor abrange a faixa de medição esperada para o projeto, 0 a 250V. E, apesar de ser um sensor invasivo, ou seja, exige corte no cabo a ser medido, ele apresenta um simples funcionamento. Basta conectar as pontas dos fios em seu *borne* de entrada e ajustar a sensibilidade no resistor variável até atingir uma boa medição. A programação também é simples, pois não exige biblioteca externa, bastando realizar a leitura do valor analógico convertido para digital que varia de 0 a 1023 e recalcular para a unidade de medida desejada, nesse caso para *Volts*.

O cálculo realizado para medição é o seguinte:

$$TensaoMedida = \frac{(ValorLido \times 220)}{1023} \quad (4.1)$$

Porém, posteriormente, identificou-se que este módulo é baseado no uso de um opto-acoplador, que, como já mencionado anteriormente na Sec. 3.3, é utilizado principalmente para isolar sinais.

Ou seja, a tensão da rede elétrica apenas ativará o opto-acoplador, informando que há uma tensão aplicada sobre o circuito. Com isso, este sensor é mais indicado para detecção de tensão e não para medição em um intervalo determinado, pois não há o tratamento adequado do sinal para fornecer uma medição precisa.

Com base nestes três pontos cruciais, a utilização de módulos, o sensor de tensão inadequado e o multiplexador, optou-se por alterar o projeto do dispositivo, dando surgimento para a versão 2.0.

4.2 Protótipo versão 2.0

Nesta próxima versão, decidiu-se pela substituição da placa de fenolite perfurada por uma placa de circuito impresso, a qual seria impressa em maquinário apropriado

fornecido pelo próprio IFB, o que será detalhado mais a frente. Além disso, também decidiu-se por substituir os módulos pela criação dos circuitos dos sensores na própria PCI (Placa de Circuito Impresso). Dessa forma, o módulo relé 2 vias foi substituído pelo circuito eletrônico de apenas um relé. O módulo sensor de tensão baseado em opto-acoplador foi substituído pelo circuito do sensor de tensão ZMPT101B. O sensor de corrente foi mantido inalterado. O multiplexador foi removido. E foi adicionado um *display LCD* para apresentar as informações no próprio dispositivo.

A justificativa para tais modificações foi a seguinte:

O circuito construído em uma placa de circuito impresso é menos suscetível a falhas de conexão e fixação, pois a posição de todos os componentes é previamente planejada com auxílio do *software* de desenho, assim como as trilhas de cobre. Isso garante segurança na soldagem e na conexão entre os componentes. Outro fator relevante é a estética que o dispositivo apresenta ao ser construído em uma placa de circuito impresso, sendo superior à estética de um circuito construído em uma placa de fenolite perfurada.

A substituição do módulo relé 2 vias pelo circuito do relé único não se motivou por mau funcionamento, e sim por outros dois motivos: utilização de um único relé e também por questão estética, já que o relé com seus componentes soldados diretamente na placa tem uma apresentação superior à um módulo externo fixado sobre o circuito.

Já a substituição entre os sensores de tensão foi por fins de funcionamento e aplicação. O sensor baseado em opto-acoplador não fornece a variação precisa que o ZMPT101B fornece, como já descrito anteriormente. E, aliado a isso, optou-se por também criar o circuito do sensor de tensão já soldado diretamente na placa do dispositivo, evitando o uso do módulo externo.

Como mencionado, não houve alteração no circuito do sensor de corrente, pois este já havia sido planejado para ter sua soldagem diretamente na placa do dispositivo.

O multiplexador continua sendo uma limitação para esta versão do dispositivo, o qual conta com um microcontrolador com apenas uma porta analógica. Dessa forma, para solucionar efetivamente esta limitação na versão 3.0, decidiu-se por substituir a placa microcontroladora para a ESP32, assim como descrito na Seção 4.3.

E, por fim, foi adicionado um display LCD para que os dados sejam apresentados diretamente pelo dispositivo, e não somente no sistema supervisorio *Blynk*. Essa decisão tem por objetivo incluir o usuário que não dispõe de acesso online ao sistema e necessita verificar as condições do dispositivo no local da instalação.

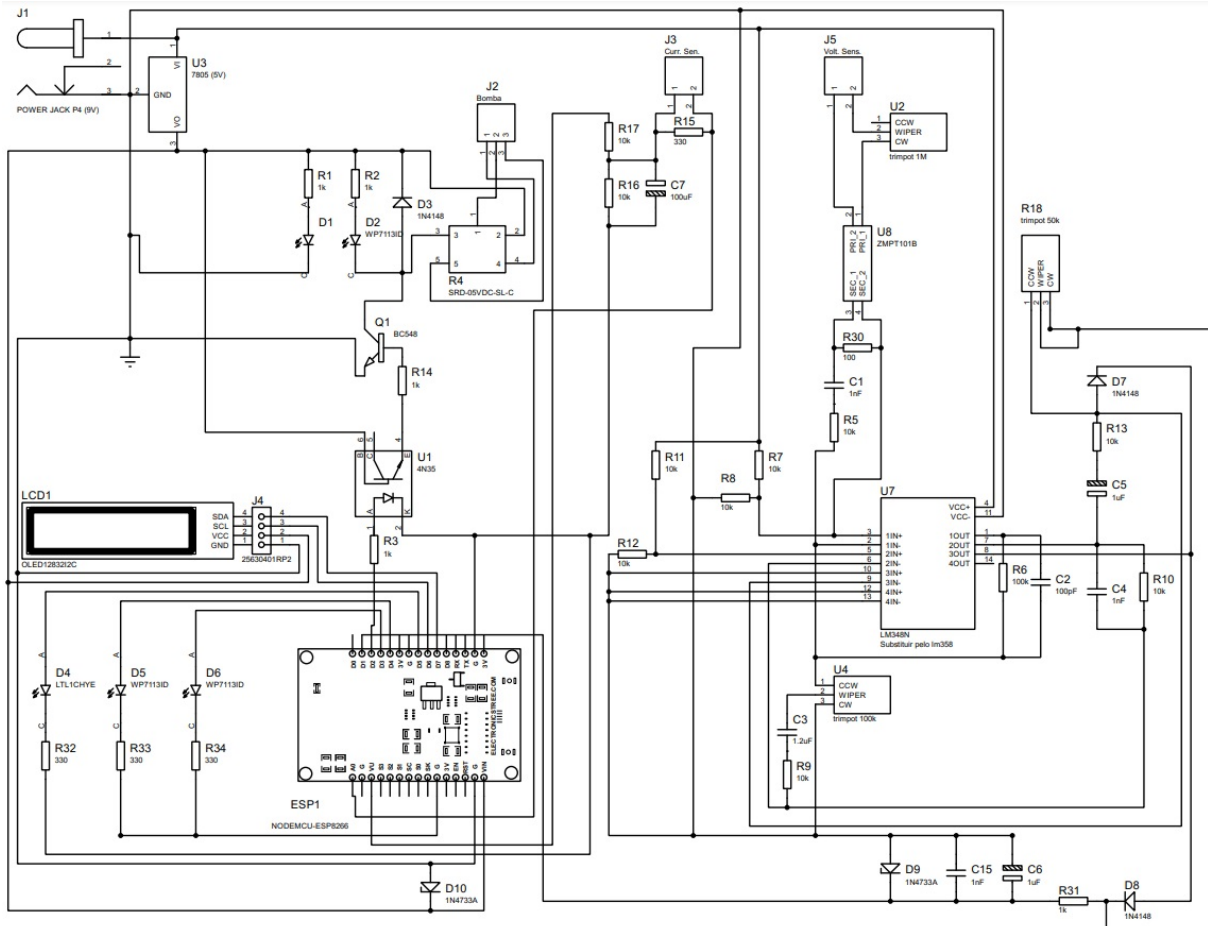
Para a implementação dessa adaptação, primeiro foi planejado o circuito eletrônico e o desenho da placa no *software* Proteus.

De acordo com o fabricante do *software*, o Proteus é uma ferramenta que combina esquemáticos elétricos com *layouts* de PCB para criar um conjunto de ferramentas integradas e fáceis de utilizar para a criação de *designs* profissionais de placas de circuito impresso (PCB's) e ainda fornece a opção de simular o circuito desenvolvido para correção

de erros (Labcenter, 2024).

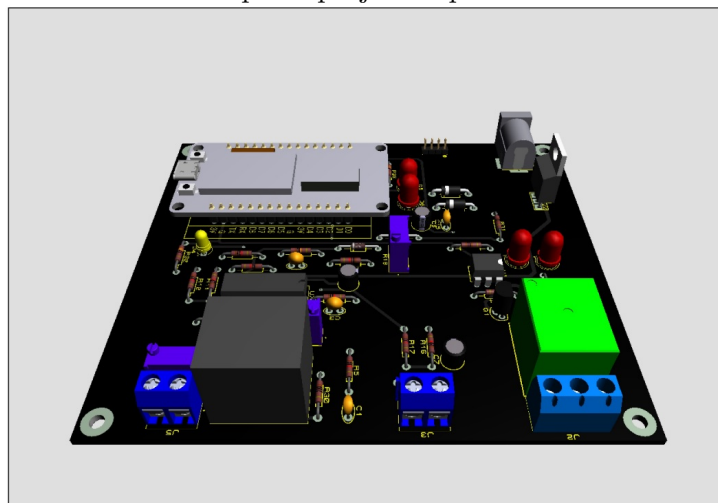
Com este software, foi possível criar e simular o circuito eletrônico proposto, além de criar o projeto da placa de circuito impresso.

Figura 4.5 – Esquema elétrico desenvolvido no proteus para a versão 2.0 do dispositivo.



Fonte: Próprio Autor.

Figura 4.6 – Modelo 3D da placa projetada para a versão 2.0 do dispositivo.

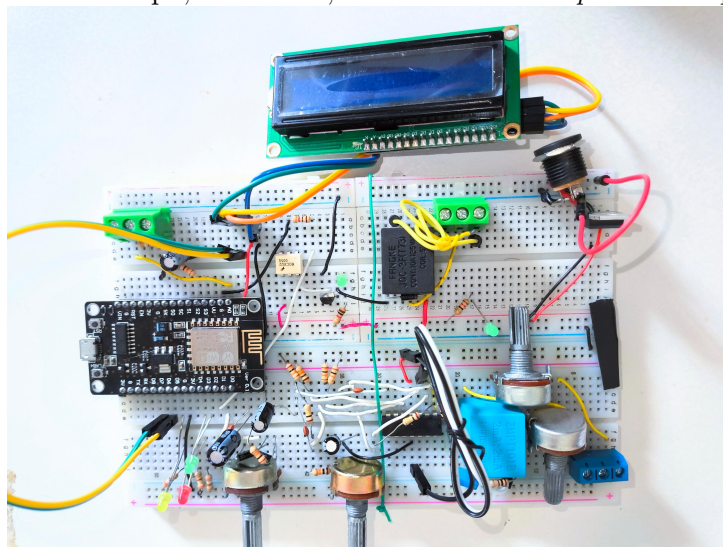


Fonte: Próprio Autor.

Nas simulações realizadas dentro do software, o circuito funcionou perfeitamente de acordo com o esperado, ou seja, os sensores forneciam os valores de tensão condizentes com a tensão ou corrente de entrada.

Para teste prático, o circuito eletrônico foi replicado em uma *protoboard*.

Figura 4.7 – Protótipo, versão 2.0, montado em uma *protoboard* para teste.



Fonte: Próprio Autor.

Durante os testes, verificou-se que quase todos os componentes funcionavam bem, incluindo a programação do microcontrolador. A única exceção foi o sensor de tensão. Este apresentava, quando não havia conexão com a rede elétrica residencial, valores de tensões variados em cada teste. O esperado para esse caso, é que o sinal fornecido pelo sensor se mantivesse no valor definido pelo *offset*, que é 2,5V. Porém, o sinal fornecido por muitas vezes era mantido em 3,2V, 6,3V e não atuava de acordo com o esperado. E aplicando a tensão da rede ao sensor, era esperado que o sinal fornecido pelo sensor variasse de acordo com a oscilação da tensão de entrada e retornasse um valor de tensão próximo a 4,5V quando a tensão medida na rede fosse 220V. Contudo, ao aplicar a tensão da rede elétrica, o sinal se manteve constante em 3,2V, evidenciado um erro no projeto ou na montagem do circuito. Não foi possível solucionar este erro e optou-se por modificar a implementação do sensor de tensão.

Com isso, a partir desta segunda versão e suas falhas, surgiu a terceira versão (3.0) e suas respectivas modificações.

4.3 Protótipo versão 3.0

O sensor de tensão embutido no circuito impresso do dispositivo foi substituído pelo módulo do sensor de tensão ZMPT101B e a placa controladora foi substituída por uma ESP32.

Dessa forma, há a garantia do funcionamento correto do sensor de tensão, possibilitando a leitura da tensão da rede elétrica. Constatou-se tal informação com base em um multímetro e o sensor de tensão calibrado aplicados à mesma rede afim de comparar os resultados.

Figura 4.8 – Leitura do Multímetro no dispositivo versão 3.0.



Fonte: Próprio Autor.

Figura 4.9 – Leitura do Sensor de Tensão Calibrado no Dispositivo Versão 3.0.



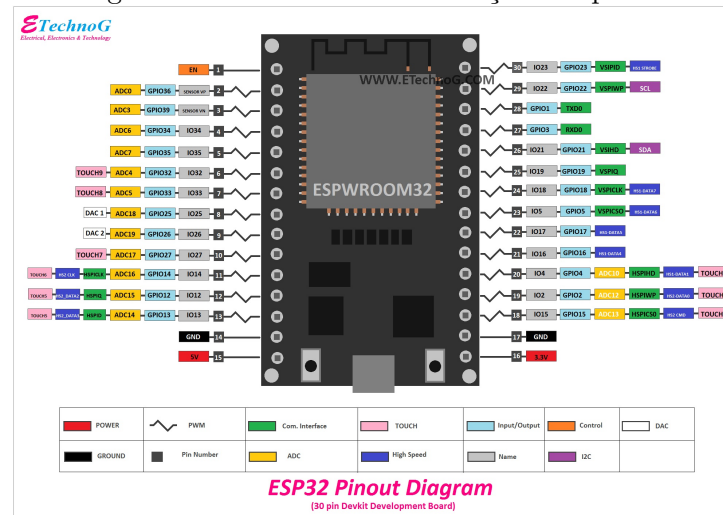
Fonte: Próprio Autor.

Pode-se observar que as duas leituras apresentam 230V (duzentos e trinta volts), confirmando que o sensor está calibrado e fornecendo uma leitura confiável.

A desvantagem da escolha do módulo do sensor de tensão é a necessidade de fixar à placa do dispositivo um módulo externo, considerando que o ideal é utilizar um circuito dedicado para o sensor de tensão embutido ao dispositivo.

A segunda alteração foi a substituição da placa controladora da ESP8266 para a ESP32. Esta segunda placa possui as mesmas vantagens da ESP8266 e ainda fornece uma maior quantidade de portas analógicas.

Figura 4.10 – ESP32 com descrição das portas.



Fonte: ETechnoG (2024)

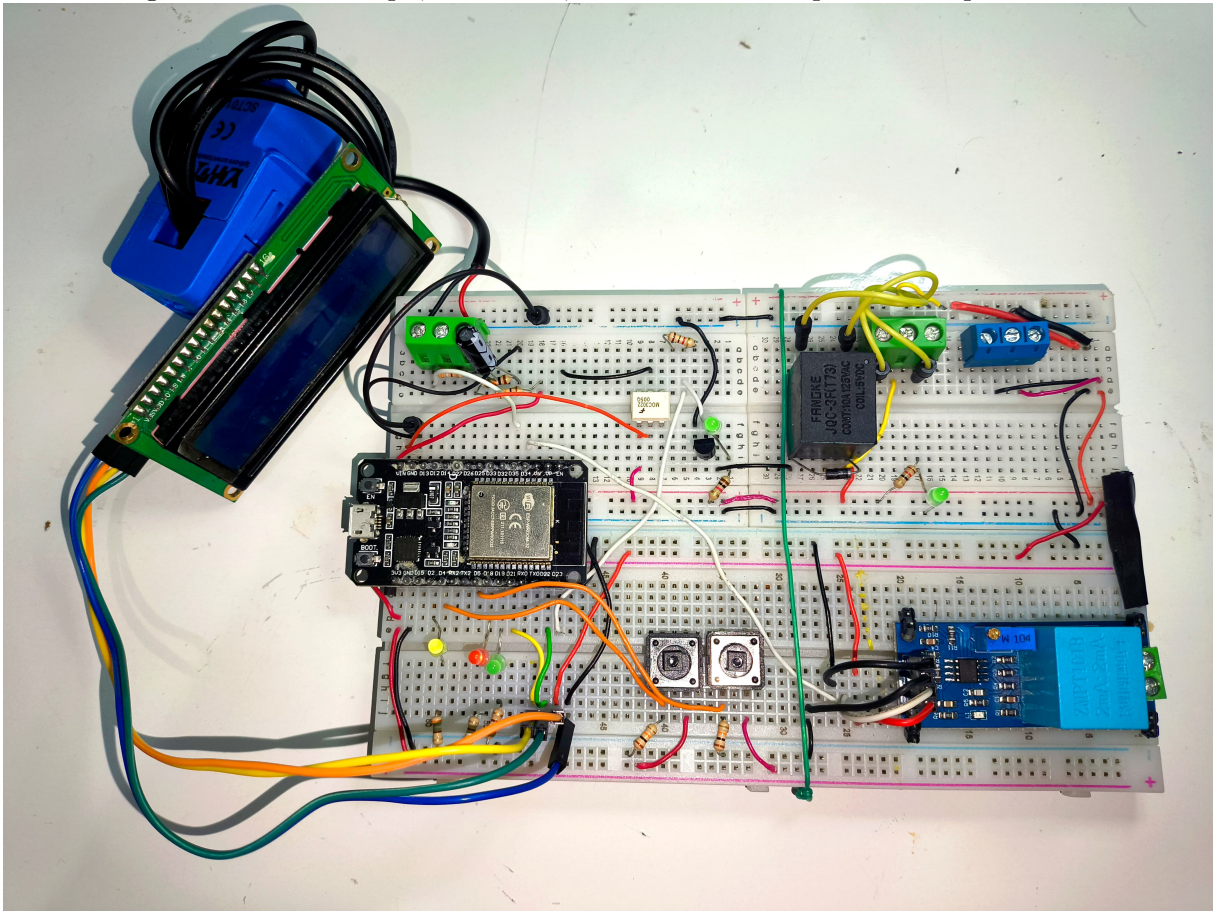
Com isso, é possível eliminar o multiplexador do circuito, eliminando também as falhas que este componente carregava. Sendo assim, os dois sensores podem ser lidos diretamente pelo controlador por meio de portas independentes.

Juntamente com essas correções, surgiram outras melhorias. Foram adicionados botões físicos para controle local do dispositivo, um segundo relé para o circuito final do dispositivo e a alimentação de 12V (doze *volts*) foi substituída por uma alimentação de 5V (cinco *volts*).

Já para os testes práticos desta versão, foi alterado o circuito da versão anterior na protoboard, retirando o circuito do sensor de tensão e alimentação e substituindo pelo módulo do sensor e pela nova fonte de tensão. Também foram inseridos os botões físicos. Neste primeiro teste prático montado na protoboard, foram adicionados apenas dois botões ao circuito, um para controle de seleção entre manual e automático e outro para controle do estado do relé. Na versão final planejada para o circuito, que será detalhada a seguir, há um terceiro botão para controlar o segundo relé que poderá ser acionado opcionalmente.

A disposição ficou da seguinte forma:

Figura 4.11 – Protótipo, versão 3.0, montado em uma protoboard para teste.

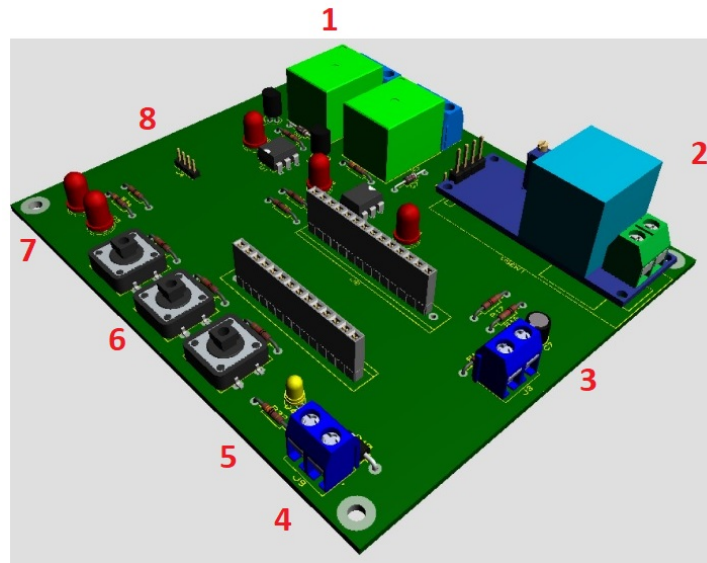


Fonte: Próprio Autor.

Neste circuito, podemos identificar todos os elementos citados dispostos em uma única protoboard, assim como os dois botões e o relé único.

A versão final projetada no esquemático da Figura 3.14, diferente do protótipo da Figura 4.11, conta com três botões e dois relés. Para melhor visualização do projeto final do dispositivo, foi criado o seguinte modelo 3D da placa de circuito impresso:

Figura 4.12 – Modelo 3D da placa projetada para a versão 3.0 do dispositivo.

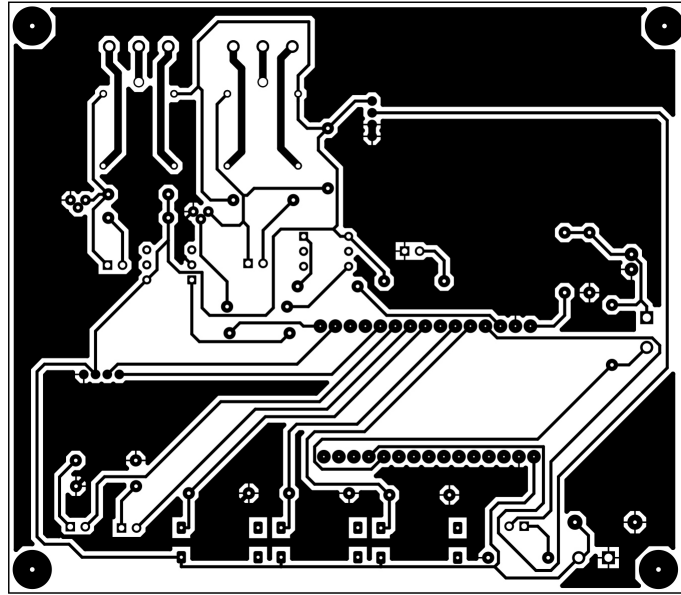


Fonte: Próprio Autor.

Neste modelo 3D podemos observar todos os componentes presentes no projeto do dispositivo. Os dois relés e seus circuitos (1), o módulo sensor de tensão ZMPT101B (2), o *borne* de entrada do sensor de corrente (3), borne de entrada da fonte de alimentação (4), *LED* de aviso da conexão com o *Wi-Fi* (5), conjunto de 3 botões para o controle manual do dispositivo (6), dupla de *LEDs* de aviso (verde e vermelho) para o status de funcionamento do dispositivo (7) e os pinos de entrada para o *display LCD* (8).

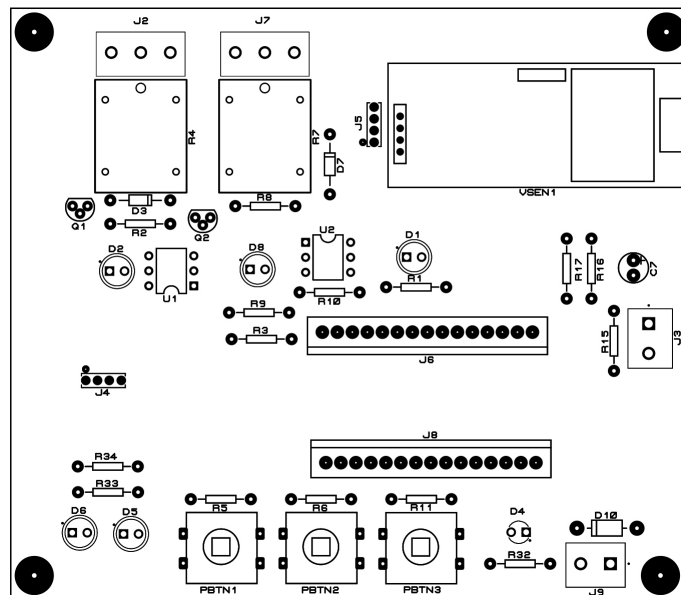
A partir do *software Proteus* que gerou modelo 3D, também é possível exportar os arquivos de impressão da Placa de Circuito Impresso (PCB), apresentados nas figuras abaixo. Com estes arquivos, pode-se confeccionar a PCB de diversas formas, sendo duas delas: utilizar maquinário específico como fresadoras ou máquinas de prototipagem de placas e encomendar de empresas especializadas em produzir placas personalizadas como a MICROPRESS.

Figura 4.13 – Trilhas do lado inferior do projeto da PCB



Fonte: Próprio Autor.

Figura 4.14 – Impressão do lado superior do projeto da PCB



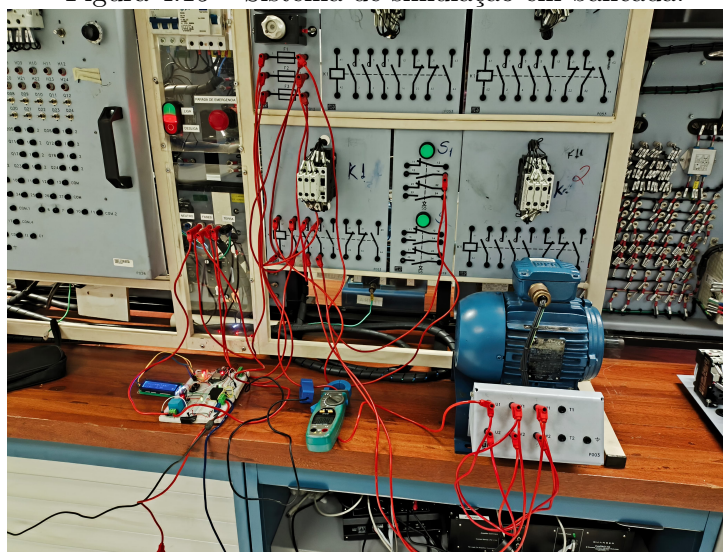
Fonte: Próprio Autor.

O IFB dispõe de maquinário de prototipagem de placa em um de seus laboratórios, mais especificamente a máquina LPKF ProtoMat que, segundo a empresa alemã LPKF, é uma máquina de prototipagem rápida e precisa de PCB de face simples ou dupla, projetada para uso diretamente em laboratório (LPKF, 2024). Houve uma tentativa de validar a confecção da PCB deste projeto com o auxílio da máquina supracitada, contudo alguns erros de operação impediram esta atividade. Um dos erros do maquinário estava relacionado à tensão do *spindle* (ferramenta que fixa a fresa). Como não foi possível solucionar o erro mencionado, esta validação deverá ser realizada em projetos futuros.

4.4 Testes Práticos

Para a validação do dispositivo, foi simulado em laboratório uma aplicação que refletisse o contexto real em que a placa será utilizada. Ou seja, foi montado em bancada o circuito de potência de um motor trifásico (simulando a bomba d'água) que é acionado por uma contatora. Essa disposição é a mesma que o circuito real utiliza atualmente, conforme informado pelo responsável técnico do condomínio que repassou a demanda.

Figura 4.15 – Sistema de simulação em bancada.



Fonte: Próprio Autor.

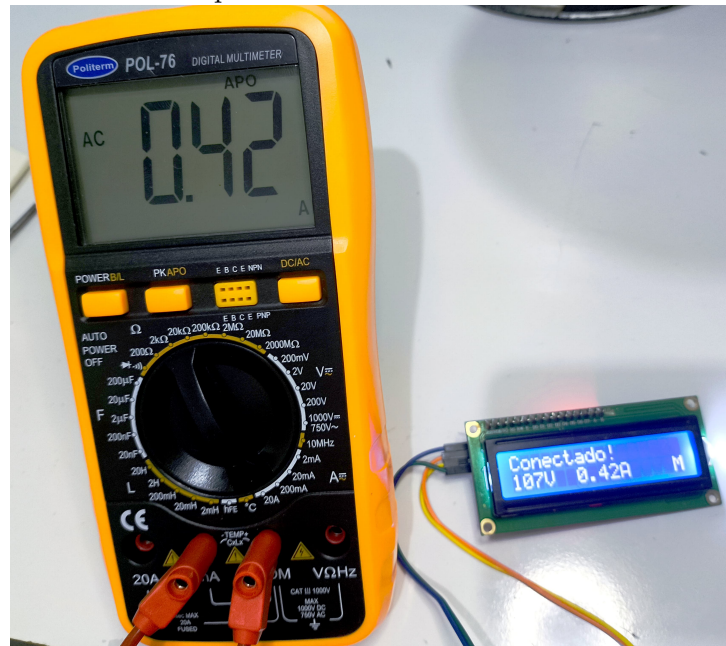
Dessa forma, o resultado esperado é que o dispositivo informe corretamente a tensão da rede e informe corretamente a corrente do motor. Além disso, quando está em automático, é esperado que o dispositivo faça a validação dos valores de tensão e corrente para acionar ou não o motor. E, quando em manual, é esperado que o dispositivo acione o motor de acordo com os botões físicos ou virtuais.

Para efeitos de comparação, foram utilizados um multímetro e um amperímetro calibrados para verificar se os valores informados pelo dispositivo refletem o sinal real de leitura.

Com tais informações à vista, pode-se dar início à validação do dispositivo.

De acordo com as seguintes imagens, é possível observar que a leitura dos valores de tensão pelo dispositivo é igual à leitura apresentada pelo multímetro, 230V. A Fig. 4.8 mostra o valor de referência da tensão e a Fig. 4.9 mostra o valor de tensão lido pelo dispositivo. E a figura a seguir mostra a comparação de leitura da corrente consumida pelo motor de simulação em um multímetro e no display do dispositivo, os quais apresentam o mesmo valor de 0,42A.

Figura 4.16 – Leitura por multímetro e sensor da corrente consumida.

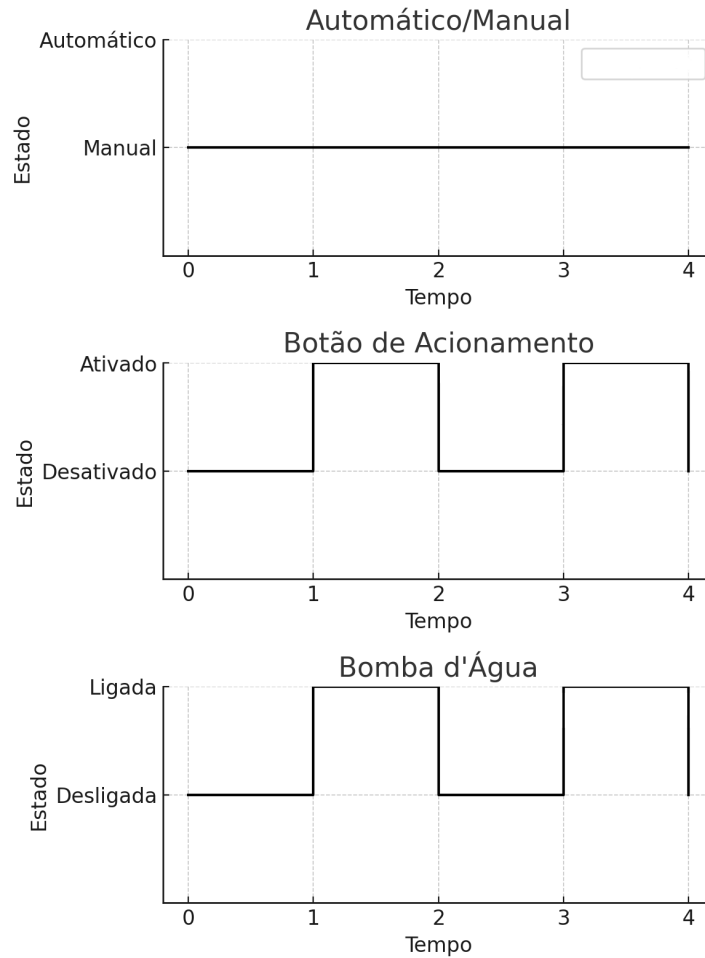


Fonte: Próprio Autor.

Além destes, foram realizados outros testes de verificação da leitura dos sensores, apresentando também valores similares ao comparar o valor informado pelo dispositivo e o valor informado pelo multímetro ou amperímetro. Com isso, pode-se dizer que a confiabilidade na leitura dos sensores, o primeiro objetivo do dispositivo, foi atingido com sucesso.

Outro ponto avaliado é o acionamento manual. Na inicialização, o dispositivo deve iniciar no modo manual e com o motor acionado. E, ao pressionar os botões físicos ou virtuais de acionamento, o motor deve ser desligado ou ligado. Esse comportamento foi observado na prática assim como era esperado de acordo com a seguinte figura. Sendo assim, o objetivo de acionar o dispositivo manualmente também foi concluído.

Figura 4.17 – Gráfico de comparação entre os estados dos botões e a bomba d'água.

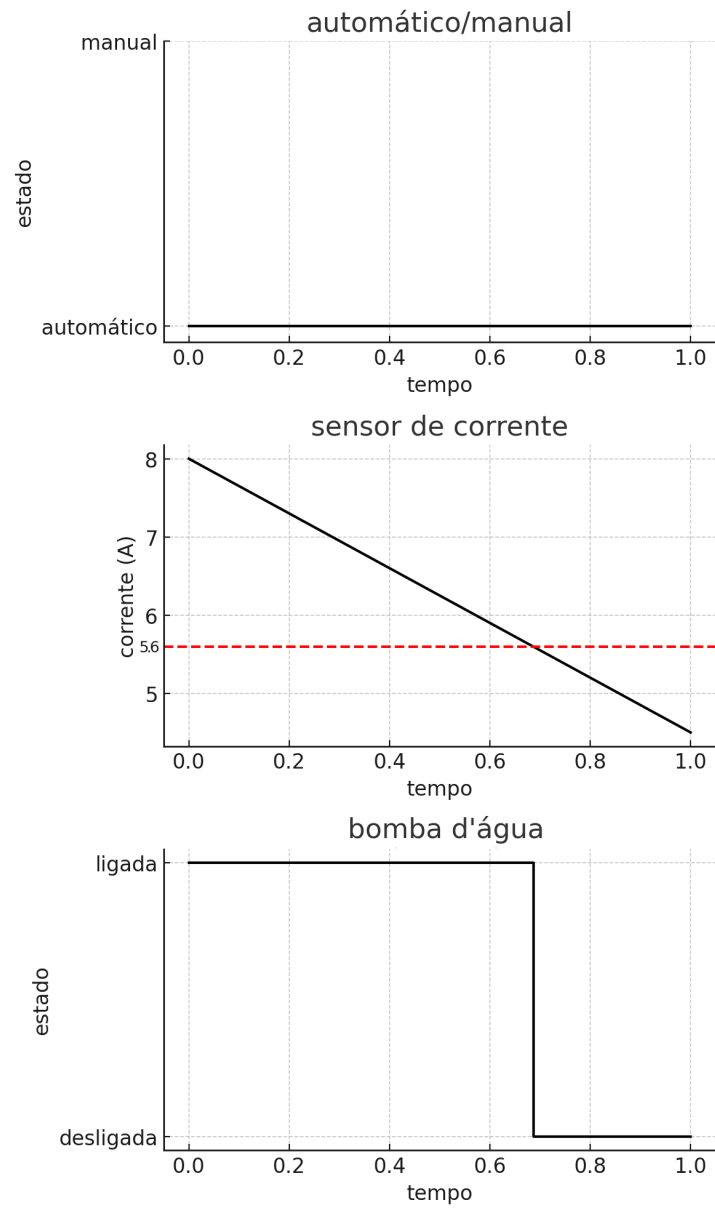


Fonte: Próprio Autor.

Na sequência, outro ponto avaliado foi o acionamento automático. Nesta configuração, o dispositivo inicia a validação das informações lidas e verifica se está dentro dos limites estabelecidos pelo usuário. Para este teste foram assumidos os seguintes limites: 5,6A para corrente mínima, 8A para corrente máxima e 200V para tensão mínima. Portanto, caso os valores lidos pelos sensores estejam todos dentro dos limites, o dispositivo mantém o motor ligado e ativa o *LED* de aviso de sistema funcionando (*LED* verde). Para testar este caso, foi inserida uma alimentação de 220V ao sistema, o que garante um contexto ideal para o funcionamento do dispositivo. Com isso, o comportamento de ativar o motor quando os valores lidos estão dentro do limite definido foi observado com sucesso, atingindo mais um objetivo.

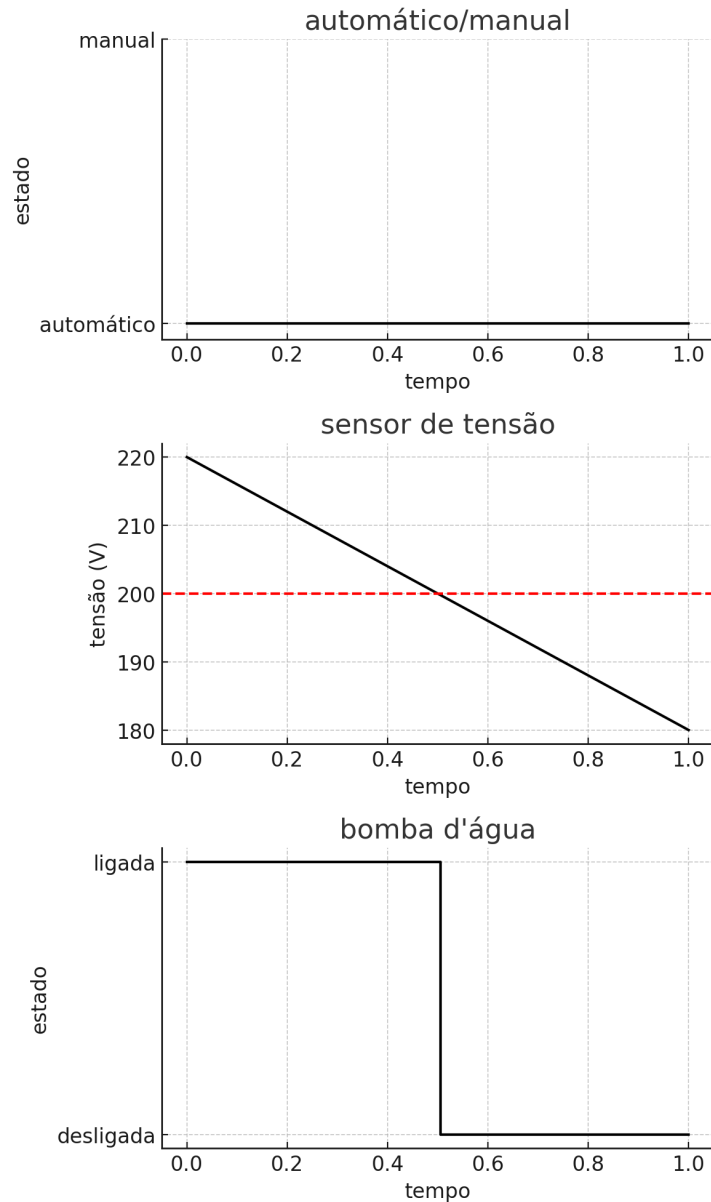
As seguintes figuras apresentam um comparativo entre o sinal de corrente e a bomba e o sinal de tensão e a bomba quando o sistema está definido como automático.

Figura 4.18 – Gráfico de comparação entre a corrente e o estado da bomba d'água.



Fonte: Próprio Autor.

Figura 4.19 – Gráfico de comparação entre a tensão da rede e a bomba d'água.



Fonte: Próprio Autor.

Assim como visto na Fig. 4.18 e na Fig. 4.19, caso pelo menos um dos valores lidos pelos sensores de tensão ou corrente estejam fora do intervalo definido pelo usuário, o dispositivo deve cortar imediatamente o acionamento da bomba, desativar o *LED* de aviso de sistema funcionando e ativar o *LED* de aviso de sistema em falha (*LED* vermelho). E, após identificar a falha, o dispositivo deve iniciar uma contagem de tempo para realizar mais uma tentativa de religação. Para testar este caso, duas ações foram tomadas. A primeira foi inserir uma alimentação de 220V e, durante a operação, variar essa alimentação para 190V, considerando que o limite mínimo aceitável é 200V e, depois que o dispositivo identificar a falha, retornar a alimentação para 220V, simulando uma oscilação na rede elétrica. E a segunda foi adicionar uma carga ao motor para que a corrente ultrapassasse o valor máximo permitido de 3A e, após o dispositivo identificar falha, retirar a carga do motor,

simulando uma sobrecarga temporária na bomba d'água. Portanto, considerando esses dois cenários de falha, o dispositivo apresentou o comportamento esperado, desativando a bomba e reativando após a contagem de tempo.

Outro cenário em que pode ocorrer a mesma falha, é o sistema identificar uma leitura fora do padrão e essa leitura se manter fora do padrão por um longo período de tempo. Neste caso, o dispositivo deve executar os mesmos passos do cenário anterior e realizar 3 tentativas de religação. Caso a falha não tenha sido corrigida automaticamente durante essas três tentativas, o sistema irá interromper o funcionamento e informar que o sistema está corrompido, piscando o LED de falha e adicionando uma mensagem de 'Sistema em Falha' ao *display LCD*. Caso não tenha intervenção humana para a correção da falha, o dispositivo irá aguardar um intervalo de tempo maior para realizar mais 3 tentativas de religação, e permanecer nesse ciclo até que a correção seja feita, manual ou automaticamente. Para testar este caso, a fonte de alimentação foi variada de 220V para 190V durante a operação e mantida nesse valor. Com isso, o dispositivo apresentou o comportamento esperado e apresentou a informação visual e textual que o sistema está corrompido. Dessa forma, o objetivo em que o dispositivo deve identificar falhas automaticamente também foi cumprido.

Por fim, resta um ponto a ser avaliado no dispositivo: a conexão com o *Wi-Fi*. Neste caso, quando o dispositivo está desconectado e tentando conectar à rede, o *LED* de aviso de conexão (*LED* amarelo), deve piscar. E, quando o dispositivo está conectado, o *LED* de aviso de conexão deve permanecer ligado e o dispositivo deve sincronizar os dados com o *Blynk*. Para testar este cenário, o dispositivo foi conectado a um roteador *Wi-Fi* que era ligado ou desligado afim de testar todos cenários. Com isso, foi observado que o dispositivo também apresentou o comportamento esperado para este ponto, cumprindo com mais este objetivo.

A plataforma de supervisão *Blynk* também passou por testes de validação. Em cada um dos cenários citados acima, é esperado que a plataforma apresente as informações dos sensores e permita o acionamento remoto do dispositivo. Também é esperado que seja possível definir os valores limites de corrente e tensão diretamente na plataforma. Durante os testes, foi constatado que a plataforma atua assim como o esperado. Por ela é possível controlar o acionamento do dispositivo e definir os valores limites de corrente e tensão e também é possível receber em tempo real a leitura dos sensores de tensão e corrente. A Fig. 3.7 e a Fig. 3.16 representam a plataforma durante a operação e as respectivas funcionalidades.

Com todos estes pontos avaliados, é possível afirmar que o dispositivo foi simulado e testado em vários cenários possíveis e está pronto para ser implementado em um contexto real, onde também deverá passar por um período de validação para que seja considerado 100% operacional.

Para finalizar a implementação do dispositivo, deve-se analisar o orçamento alocado

para o desenvolvimento e avaliar se o custo benefício é aplicável à usuários que buscam alternativas acessíveis aos sistemas comerciais. Para isso, foram criadas as seguintes tabelas afim de reunir o valor dos componentes utilizados bem como comparar com soluções comerciais já existentes.

Componente	Preço 1	Preço 2
SCT-013	R\$65,46 - ELETROGATE	R\$47,40 - ROBOCORE
ZMPT101B	R\$21,76 - ELETROGATE	R\$18,90 - ROBOCORE
Relé	R\$4,66 - ELETROGATE	R\$5,22 - ROBOCORE
Display LCD	R\$28,40 - ELETROGATE	R\$35,50 - HUINFINITO
ESP32	R\$44,55 - ROBOCORE	R\$83,90 - HUINFINITO
Total	R\$164,83	R\$190,92

Tabela 2 – Tabela de preços dos componentes utilizados

Os dois valores totais presentes na tabela apresentada não contemplam a confecção da PCB pois o valor desta varia com a forma de confecção. Considerando a utilização do maquinário disponibilizado pelo IFB para a confecção do dispositivo, seria necessário uma placa de fenolite de 10x10cm para a prototipagem, o que resulta em um custo de aproximadamente R\$15,00 por placa. Portanto, para este trabalho, foi considerado que o custo de confecção do dispositivo está entre R\$175,00 a R\$206,00.

Para avaliar o custo benefício do dispositivo deste projeto, podemos realizar uma comparação com as soluções apresentadas na Subseção 2.2.5 e na Subseção 2.2.6.

Solução	Custo	Funcionalidades
Dispositivo deste projeto	De R\$175 a R\$206	Controle autônomo da bomba, controle manual, supervisão remota, sistema de alertas
NI35W -COEL	R\$285,00 - AQUECECONTROL	Controle autônomo da bomba, sistema de alertas
TPC-58 -THEBE	R\$690,00 - THEBE	Controle autônomo da bomba, sistema de alertas

Tabela 3 – Tabela de comparação entre o dispositivo deste projeto e soluções comerciais

Tendo em vista que o dispositivo desenvolvido neste projeto ainda é um protótipo e o valor de custo final é aproximado, podemos observar que possui o menor valor de investimento entre as três soluções. O dispositivo deste projeto também oferece maiores opções de funcionalidades, o que incrementa a avaliação do custo benefício. Com isso, pode-se afirmar que este protótipo tem um bom custo benefício por oferecer ampla gama de funcionalidades a um custo relativamente baixo.

5 CONSIDERAÇÕES FINAIS

Este trabalho foi motivado na necessidade de um condomínio que possui sistema de abastecimento por poço artesiano e frequentemente sofre danos na bomba elétrica devido à falta de água no poço ou quedas de tensão na rede elétrica.

Com base nos testes práticos apresentados na Sec. 4.4, os resultados demonstraram que o sistema supervisorio *Blynk* e o dispositivo de monitoramento da bomba funcionaram corretamente em diversos cenários de ambiente controlado, o que valida a sua aplicação e permite implementação no contexto real do condomínio.

Portanto, conclui-se que o dispositivo desenvolvido foi validado com sucesso, possibilitando instalação no condomínio e iniciar a etapa de testes em campo. Futuras melhorias, como a confecção da PCB, permitirão sua plena implementação e ampliação para outros contextos.

6 SUGESTÃO DE MELHORIAS FUTURAS

Assim como descrito na Cap. 4, o dispositivo final passou por diversas modificações, resultando no protótipo apresentado. Porém, ainda há muitos pontos de melhorias que podem ser realizados afim de evoluir o projeto.

A primeira melhoria que pode ser citada é criar um circuito embarcado à placa principal para o sensor de tensão ZMPT101B, pois, na versão final do dispositivo desenvolvida neste projeto, utilizou-se um módulo externo. Para tal melhoria, pode-se utilizar o circuito da Fig. 4.5 já desenvolvido e descrito na Sec. 4.2. O circuito apresentado nessa seção foi simulado no *software* Proteus e estava em perfeito funcionamento nas simulações, porém, ao aplicar o circuito à protoboard e realizar os testes práticos, houve falhas. Portanto, uma opção é estudar e corrigir os erros que surgiram para que assim possa ser criado um circuito embutido à placa do dispositivo para o sensor de tensão. Outra opção para o mesmo fim é criar um novo circuito para o sensor de tensão livre de erros desde a origem.

A segunda melhoria a ser feita é desenvolver um circuito para o sensor de corrente a fim de garantir maior segurança elétrica e estabilidade no sinal, pois o circuito atual possui apenas um divisor de tensão e um capacitor, podendo ser suscetível a falhas em cenários de alto estresse.

A terceira melhoria disponível é confeccionar a placa de circuito impresso (PCB) que é de suma importância para aplicação real já que garante melhor fixação dos componentes e segurança na instalação. Para isso, pode-se utilizar o modelo projetado neste trabalho e desenvolvido com auxílio do *software* Proteus.

A quarta melhoria, assim com citada no Subseção 2.2.4, é implementar um sistema de medição do nível de água da caixa d'água do condomínio. Assim o controle da bomba também pode ser feito de acordo com a quantidade de água disponível no reservatório. Ou seja, caso esteja vazio, a bomba deve ser ligada para abastecer a caixa. Caso o reservatório esteja cheio, o funcionamento da bomba deve ser interrompido. E caso o nível da água fique abaixo da metade da capacidade total, o síndico deve ser notificado via plataforma *Blynk* ou *email* para estabelecer planos de contingência e economia de água com os moradores.

A quinta e última melhoria possível é incrementar funcionalidades ao *firmware* e à plataforma *Blynk* como: possibilitar edição da conexão *Wi-Fi* a partir de comunicação *Bluetooth* e criar sistema de notificação para informar aos usuários o status de funcionamento do sistema a partir de *smartphones*.

REFERÊNCIAS

- AQUECECONTROL. Rele de nível coel ni35w 110-220v (poço) com eletrodo. **Mercado Livre**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <https://produto.mercadolivre.com.br/MLB-3690247643-rele-de-nivel-coel-ni35w-110-220v-poco-com-eletrodo-_JM?attributes=VOLTAGE%3AMjRWIC0gMjQwVg%3D%3D&quantity=1>.
- AURESIDE. Automação residencial - riscos e oportunidades. **Aureside - Associação Brasileira de Automação Residencial e Predial**, 2020. Acesso em: 28 de agosto de 2024. Disponível em: <<http://www.aureside.org.br/noticias/automacao-residencial---riscos-e-oportunidades#:~:text=O%20crescimento%20geral%20do%20mercado,2025%20um%20percentual%20de%20178%25>>.
- BARBACENA, I. L.; FLEURY, C. A. Display lcd. **IFSC**, 1996. Acesso em: 14 de agosto de 2024.
- BASS. **Automação Residencial Preço: Saiba Quanto Custa Viver no Futuro**. Acesso em: 28 de agosto de 2024. Disponível em: <<https://bassautomacao.com.br/automacao-residencial-preco/>>.
- BIOSEMENTES. Controlador eletrônico automático liga e desliga bombas de água. **Biosementes**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.biosementes.com.br/loja/item/controlador-eletronico-automatlico-liga-e-desliga-bombas-de-agua>>.
- Blynk Inc. **Low-code IoT cloud platform with user experience at its core**. 2024. [Accessed: 25-August-2024]. Disponível em: <<https://blynk.io>>.
- COEL. Ni35w - controlador de nível. **COEL**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.coel.com.br/produto/ni35w-controlador-de-nivel/>>.
- DEMETRAS, E. **SCT-013 - Sensor de Corrente Alternada com Arduino**. 2017. <<https://portal.vidadesilicio.com.br/sct-013-sensor-de-corrente-alternada/>>. Acesso em: 14 de agosto de 2024.
- DIAS, N. Automação residencial: sua casa inteligente sem complicações. **Tuacasa**, 2023. Acesso em: 28 de agosto de 2024. Disponível em: <<https://www.tuacasa.com.br/automacao-residencial/5>>.
- DICTIONARY, C. **Automation**. Acesso em: 28 de agosto de 2024. Disponível em: <<https://dictionary.cambridge.org/pt/dicionario/ingles-portugues/automation>>.
- ELETROGATE. Display lcd 16x2 com backlight azul e i2c. **Eletrogate**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.eletrogate.com/display-lcd-16x2-i2c-backlight-azul>>.
- ELETROGATE. Módulo sensor de tensão ac 0 a 250v zmpt101b. **Eletrogate**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.eletrogate.com/modulo-sensor-de-tensao-ac-0-a-250v-voltmetro-zmpt101b>>.

ELETROGATE. Relé single 12v. **Eletrogate**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.eletrogate.com/rele-songle-12V>>.

ELETROGATE. Sensor de corrente não invasivo 100a sct-013. **Eletrogate**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.eletrogate.com/sensor-de-corrente-nao-invasivo-100a-sct-013>>.

ETechnoG. **ESP32 Pinout Diagram(30-pin Devkit)**. 2024. [Accessed: 25-August-2024]. Disponível em: <<https://www.etechnog.com/2021/12/esp32-pinout-diagram30-pin-devkit.html>>.

GOEKING, W. Da máquina a vapor aos softwares de automação. **O Setor Elétrico**, 2010. Acesso em: 28 de agosto de 2024. Disponível em: <<https://www.osestoreletrico.com.br/da-maquina-a-vapor-aos-softwares-de-automacao/>>.

GROOVER, M. P. **Automation, Production Systems, and Computer-Integrated Manufacturing**. Department of Industrial Engineering, Tsinghua University, 2002. Disponível em: <<https://www.amazon.com/Automation-Production-Systems-Computer-Integrated-Manufacturing/dp/9332572496>>. Acesso em: 28 de agosto de 2024.

HERCULANO, G. G.; SALVADOR, T. C.; SILVA, A. d. S.; SOUZA, E. P. d.; TOLEDO, E. G. S. **Desenvolvimento de um circuito de controle de bomba d'água: Com ajuste dinâmico de velocidade**. Trabalho de Conclusão de Curso. — CENTRO ESTADUAL PAULA SOUZA, SÃO PAULO, 2024. Acesso em: 19 de outubro de 2024. Disponível em: <https://ric.cps.sp.gov.br/bitstream/123456789/22050/2/cursotecnicoemeletrot%c3%a9cnica_2024_1_andredossantossilva_desenvolvimentodeumcircuitodecontroledebombadagua.pdf>.

HUINFINITO. Display cristal Líquido (lcd 16x02 - az/br) com módulo i2c. **HU Infinito**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.huinfinito.com.br/modulos/1182-display-cristal-liquido-lcd-16x02-azbr-com-modulo-i2c.html>>.

HUINFINITO. Módulo wifi nodemcu esp-32 + bluetooth. **HU Infinito**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.huinfinito.com.br/home/1411-modulo-wifi-nodemcu-esp-32-bluetooth.html>>.

INSTRUMENTS, T. **Datasheet: CD54/74HC4051, CD54/74HCT4051, CD54/74HC4052, CD54/74HCT4052, CD54/74HC4053, CD54/74HCT4053**. 1997. SCHS122I. Novembro 1997 - Revisado julho 2004. Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/191457/TI/CD74HC4053E.html>>. Acesso em: 09 ago. 2024.

KUSTRO, G. O que é automação industrial? **Automação Industrial**, 2017. Acesso em: 28 de agosto de 2024. Disponível em: <<https://www.automacaoindustrial.info/o-que-e-automacao-industrial/#:~:text=A%20automa%C3%A7%C3%A3o%20industrial%20come%C3%A7ou%20a,primeiros%20dispositivos%20simples%20e%20semiautom%C3%A1ticos>>.

Labcenter. **Proteus - PCB Design and Simulation Made Easy**. 2024. [Accessed: 26-August-2024]. Disponível em: <<https://www.labcenter.com/>>.

LPKF. **The Fastest Road to Success**. 2024. [Accessed: 26-August-2024]. Disponível em: <<https://en.lpkf.com/landing/rapid-pcb-prototyping/protomat-s103.htm>>.

MANKAR, J.; DARODE, C.; TRIVEDI, K.; KANOJE, M.; SHAHARE, P. Review of i2c protocol. **International Journal of Research in Advent Technology**, Citeseer, v. 2, n. 1, 2014. Acesso em: 14 de agosto de 2024.

MARIANO, J. M. **AUTOMAÇÃO DE DISPOSITIVO PARA MEDIÇÃO DE LIMITE DE LIQUIDEZ CONCHA CASAGRANDE COM SENSOR LASER**. Trabalho de Conclusão de Curso. — Universidade Tecnológica Federal do Paraná, Toledo, 2021. Acesso em: 14 de agosto de 2024. Disponível em: <https://www.researchgate.net/profile/Marcos-Bombacini/publication/351747395_AUTOMACAO_DE_DISPOSITIVO_PARA_MEDICAO_DE_LIMITE_DE_LIQUIDEZ_CONCHA_CASAGRANDE_COM_SENSOR_LASER/links/60a7769f92851ca9dcd3a184/AUTOMACAO-DE-DISPOSITIVO-PARA-MEDICAO-DE-LIMITE-DE-LIQUIDEZ-CONCHA-CAS.pdf>.

MICROPRESS. Micropress - placas de circuito impresso. **Micropress**, 2024. Acesso em: 19 de outubro de 2024. Disponível em: <<https://www.micropress.com.br/produtos/placas-de-circuito-impresso>>.

MORAIS, H. S. **Automação residencial : sistema integrado de uma residência inteligente utilizando controlador lógico programável**. Trabalho de Conclusão de Curso. — Universidade Federal de Ouro Preto, João Monlevade, 2017. Acesso em: 28 de agosto de 2024. Disponível em: <<https://www.monografias.ufop.br/handle/35400000/360>>.

NETO, L. D. **PROTÓTIPO DE AUTOMAÇÃO RESIDENCIAL UTILIZANDO UMA ASSISTENTE DE VOZ**. Trabalho de Conclusão de Curso. — Universidade Regional de Blumenau, Blumenau, 2018. Acesso em: 28 de agosto de 2024. Disponível em: <https://www.furb.br/dsc/arquivos/tccs/monografias/2018_1_leandro-dallarosa-neto_monografia.pdf>.

OLIVEIRA, R. R. **USO DO MICROCONTROLADOR ESP8266 PARA AUTOMAÇÃO RESIDENCIAL**. Trabalho de Conclusão de Curso. — Escola Politécnica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2017. Acesso em: 28 de agosto de 2024. Disponível em: <<http://repositorio.poli.ufrj.br/monografias/monopoli10019583.pdf>>.

OLIVEIRA, S. de. **Internet das coisas com ESP8266, Arduino e Raspberry PI**. Novatec Editora, 2017. Disponível em: <https://books.google.com.br/books?hl=pt-BR&lr=&id=E8gmDwAAQBAJ&oi=fnd&pg=PA17&dq=o+que+%C3%A9+microcontroller&ots=5RM7dt9Z3x&sig=mzsMxFwA_LD4jd8uYpXTd6cFAL8#v=onepage&q&f=false>. Acesso em: 19 de março de 2024.

RASHID, M. H. **Power Electronics: Circuits, Devices and Applications**. [S.l.]: Prentice Hall, 2003.

ROBOCORE. Esp32 - wifi + bluetooth. **Robocore**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.robocore.net/wifi/esp32-wifi-bluetooth>>.

ROBOCORE. Módulo sensor de tensão ac 0 a 250v zmpt101b. **Robocore**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.robocore.net/sensor-corrente-tensao/modulo-sensor-de-tensao-ac-zmpt101b>>.

ROBOCORE. Relé metaltex. **Robocore**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.robocore.net/atuator-rele/rele-metaltex>>.

ROBOCORE. Sensor de corrente não invasivo 100a sct-013. **Robocore**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.robocore.net/sensor-corrente-tensao/sensor-de-corrente-nao-invasivo-100a-sct-013>>.

SANTOS, C. C. D.; BIASO, I. C. D.; OLIVEIRA, S. D. **Quarto Adaptado para Pessoas com Deficiência Física Motora**. Trabalho de Conclusão de Curso. — FACULDADE DE TECNOLOGIA DE ITAQUERA, São Paulo, 2017. Acesso em: 28 de agosto de 2024. Disponível em: <<https://drive.google.com/file/d/1ionC0elE9uJXliF-3FelUDyOpNNMWsG8/view>>.

SEMICONDUCTORS, N. **PCF8574; PCF8574A**. 2013. Rev. 5. Disponível em: <https://www.arduinoecia.com.br/downloads/Datasheet_PCF8574_PCF8574A.pdf>. Acesso em: 15 ago. 2024.

TECDICAS. **Desenvolvendo um módulo relé 5V para sistemas embarcados**. 2019. <<https://tecdicas.com/desenvolvendo-um-modulo-rele-5v-para-sistemas-embarcados/>>. Acesso em: 14 de agosto de 2024.

TECHNOLOGY, E. Zener diode as voltage regulator and overvoltage protection. **Electrical Technology**, 2024. Acesso em: 21 de agosto de 2024. Disponível em: <<https://www.electricaltechnology.org/2020/05/zener-diode-as-voltage-regulator.html>>.

THEBE. Controlador eletrônico tpc-58/3,0cv mono (220 v) (cód. 1127.04.10661). **Thebe Bombas Hidráulicas**, 2024. Acesso em: 20 de outubro de 2024. Disponível em: <<https://www.lojathebe.com.br/produto/controlador-eletronico-tpc-58-3-0cv-mono-220v>>.

WEG. Automação residencial: o que é, como funciona e quais as vantagens. **WEG**, 2020. Acesso em: 28 de agosto de 2024. Disponível em: <<https://www.weg.net/tomadas/blog/tecnologia/automacao-residencial-o-que-e-como-funciona-e-quais-as-vantagens/>>.

APÊNDICE A – Firmware

```

1
2 // Inormacoes do Dispositivo fornecidas pelo Blynk
3 #define BLYNK_TEMPLATE_ID "TEMPLATE_BLYNK"
4 #define BLYNK_TEMPLATE_NAME "NOME_DO_PROJETO"
5 #define BLYNK_AUTH_TOKEN "TOKEN_DE_AUTENTICACAO"
6
7 // Comente esta linha para desabilitar os prints e economizar espaco
8 #define BLYNK_PRINT Serial
9
10 //=====
11 //Inclusao de Bibliotecas
12
13 #include <WiFi.h>
14 #include <BlynkSimpleEsp32.h>
15 #include <Wire.h>
16 #include <LiquidCrystal_I2C.h>
17 #include "EmonLib.h"
18 #include <ZMPT101B.h>
19
20 //=====
21 // Dados do WiFi
22 char ssid[] = "NOME_DO_WIFI";
23 char pass[] = "SENHA_DO_WIFI";
24
25 BlynkTimer timer;
26 //=====
27 //Mapeamento de hardware
28
29 #define Pin_Sensor_Tensao 34
30 #define pinSCT 35
31 #define led_conexao 2
32 #define led_bomba_falha 18
33 #define led_bomba_ok 19
34 #define relay_Pin 23
35 #define relay_Pin_Backup 15
36 #define SDA_Pin 21
37 #define SCL_Pin 22
38 #define selecao_btn_Pin 4
39 #define controle_btn_Pin 5
40 #define controle_btn_Pin_Backup 16
41
42 ZMPT101B SensorTensao(Pin_Sensor_Tensao, 50.0); // Inicia o sensor
    de tensao

```

```

43   EnergyMonitor SCT013;                               // Inicia o sensor
      de corrente
44
45   #define LCD_endr 0x27
      // Cria endereco I2C do LCD e define tamanho
46   LiquidCrystal_I2C lcd(LCD_endr, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
      // Inicia o LCD
47
48   //=====
49   //Definicao de Variaveis
50
51   //Armazena estado de conexao com o blynk
52   bool blynkConectado;
53
54   //Sensor de Corrente
55   int tensao = 220;
56   int potencia;
57   double Irms = 0;
58
59   //controle da corrente da bomba
60   float correnteMinima = 2.0;
61   float correnteMaxima = 3.5;
62
63   //Sensor de Tensao
64   float tensaoMedida = 0;
65
66   //controle da Tensao da Rede
67   int tensaoMinima = 200;
68
69   //Controle do Blynk
70   bool selecaoAutManValue = 1; // 0 -> automatica e 1 -> manual
71   bool releValue = 0;          // Recupera o valor do blynk para o
      Rele
72   bool releValue_Backup = 0;    // Recupera o valor do blynk
      para o Rele de backup
73
74   //Controle da bomba e de falha da bomba
75   bool acionarBomba = false;   //Variavel para controlar
      manualmente o rele
76   bool acionarBomba_Backup = false; //Variavel para controlar
      manualmente o rele de backup
77
78   bool inicializacaoBomba = 0; //desativada no inicio -
      verifica se foi iniciada
79   int tempoInicializacaoBomba = 2000; // dois segundos para iniciar
80
81   bool falhaBomba = 0;

```

```

82
83 unsigned long inicioTempoFalha = 0;
84 unsigned long intervaloCorrecaoErro = 10000; //10s
85 unsigned long inicioTempoCorrompido = 0;
86 unsigned long intervaloCorrecaoCorrompido = 20000; //20s
87
88 int tentativasInicializacao = 0; // armazena a quantidade de
      tentativas de inicializacao da bomba apos o erro
89 bool sistemaCorrompido = false; // diz se o sistema esta
      corrompido
90
91 //Controle de estado dos botoes fisicos
92 bool botaoSelecaoPressionado = false;
93 bool botaoControlePressionado = false;
94 bool botaoControlePressionado_Backup = false;
95
96 // Variaveis para o LCD
97 String LCDMensagem = "";
98 float LCDCorrente = 0;
99 int LCDTensao = 0;
100 String LCDselecaoAM = "M";
101
102 //=====
103 //Funcoes de controle do Blynk
104
105 BLYNK_WRITE(V0) {
106     correnteMinima = param.asFloat();
107 }
108
109 BLYNK_WRITE(V3) {
110     correnteMaxima = param.asFloat();
111 }
112
113 BLYNK_WRITE(V7) {
114     tensaoMinima = param.asInt();
115 }
116
117 BLYNK_WRITE(V6) {
118     selecaoAutManValue = param.asInt();
119     // Atualiza o estado do V5
120     if (selecaoAutManValue == 0) {
121         //Se automatico
122         Blynk.virtualWrite(V5, 0);
123         acionarBomba = false;
124         Blynk.virtualWrite(V8, 0);
125         acionarBomba_Backup = false;
126     }

```

```
127     }
128
129     BLYNK_WRITE(V5) {
130         //recebe o valor do blynk
131         releValue = param.asInt();
132
133         if (selecaoAutManValue == 1) {
134             //aciona o rele de acordo com o botao se esta em manual
135             acionarBomba = releValue;
136         }
137     }
138
139     BLYNK_WRITE(V8) {
140         //recebe o valor do blynk
141         releValue_Backup = param.asInt();
142
143         if (selecaoAutManValue == 1) {
144             //aciona o rele backup de acordo com o botao se esta em manual
145             acionarBomba_Backup = releValue_Backup;
146         }
147     }
148
149     //funcao chamada sempre que o dispositivo estiver conectado ao Blynk
150     .Cloud
151     BLYNK_CONNECTED() {
152         // Change Web Link Button message to "Congratulations!"
153         Blynk.setProperty(V3, "offImageUrl", "https://static-image.nyc3.
154             cdn.digitaloceanspaces.com/general/fte/congratulations.png");
155         Blynk.setProperty(V3, "onImageUrl", "https://static-image.nyc3.cdn
156             .digitaloceanspaces.com/general/fte/congratulations_pressed.png
157             ");
158         Blynk.setProperty(V3, "url", "https://docs.blynk.io/en/getting-
159             started/what-do-i-need-to-blynk/how-quickstart-device-was-made"
160             );
161     }
162
163     void setup() {
164
165         //Definindo portas
166         pinMode(relay_Pin, OUTPUT);
167         digitalWrite(relay_Pin, LOW);
168         pinMode(relay_Pin_Backup, OUTPUT);
169         digitalWrite(relay_Pin_Backup, LOW);
170
171         pinMode(Pin_Sensor_Tensao, INPUT);
172         SCT013.current(pinSCT, 1.4); // Current: input pin, calibration.
173         pinMode(led_conexao, OUTPUT);
```

```

168     pinMode(led_bomba_ok, OUTPUT);
169     pinMode(led_bomba_falha, OUTPUT);
170
171     pinMode(selecao_btn_Pin, INPUT_PULLDOWN);
172     pinMode(controle_btn_Pin, INPUT_PULLDOWN);
173     pinMode(controle_btn_Pin_Backup, INPUT_PULLDOWN);
174
175     //Configura interrupcoes para os botoes fisicos
176     attachInterrupt(selecao_btn_Pin, lerSelecaoBtnPressionado, RISING)
177         ;
178     attachInterrupt(controle_btn_Pin, lerControleBtnPressionado,
179         RISING); attachInterrupt(controle_btn_Pin_Backup,
180         lerControleBtnPressionado_Backup, RISING);
181
182     SensorTensao.setSensitivity(1050);
183
184     Wire.begin(SDA_Pin, SCL_Pin); // Inicializa a comunicacao I2C
185
186     LCDMensagem = "Iniciando...";
187     lcd.begin(16, 2);
188     lcd.setBacklight(HIGH);
189     lcd.setCursor(0, 0);
190     lcd.print(LCDMensagem);
191
192     Serial.begin(115200);
193
194     //Configura WiFi e Blynk
195     WiFi.begin(ssid, pass);
196     Blynk.config(BLYNK_AUTH_TOKEN);
197
198     timer.setInterval(300L, leituraSensores);
199 }
200
201 void loop() {
202     if (WiFi.status() == WL_CONNECTED) {
203         Blynk.run();
204         timer.run();
205
206         if (Blynk.connected()) {
207             if (blynkConectado == false) {
208                 //Sincroniza os valores do Blynk
209                 Blynk.syncVirtual(V0);
210                 Blynk.syncVirtual(V3);
211                 Blynk.syncVirtual(V7);
212                 Blynk.virtualWrite(V6, selecaoAutManValue);
213                 Blynk.virtualWrite(V5, acionarBomba);
214                 Blynk.virtualWrite(V8, acionarBomba_Backup);

```

```
212         blynkConectado = true;
213     }
214 }
215 //Acende o led se conectado ao Blynk e pisca se nao conectado
216 digitalWrite(led_conexao, HIGH);
217 } else {
218     leituraSensores();
219     blynkConectado = false;
220     piscarLed(led_conexao, 100);
221 }
222 }
223
224 void leituraSensores() {
225
226     //Verifica se algum botao foi pressionado
227     if (botaoSelecaoPressionado == true) {
228         botaoSelecaoPressionado = false;
229         selecaoAutManValue = !selecaoAutManValue;
230         acionarBomba = selecaoAutManValue == 0 ? false : acionarBomba;
231         //Define controle como desligado se botao
232         mudou pra automatico
233         acionarBomba_Backup = selecaoAutManValue == 0 ? false :
234         acionarBomba_Backup; //Define controle como desligado se
235         botao mudou pra automatico
236     if (blynkConectado == true) {
237         Blynk.virtualWrite(V6, selecaoAutManValue);
238     }
239 }
240
241 if (botaoControlePressionado == true) {
242     botaoControlePressionado = false; //Reinicia estado do botao
243     acionarBomba = !acionarBomba;
244     if (blynkConectado == true) {
245         Blynk.virtualWrite(V5, acionarBomba);
246     }
247 }
248
249 if (botaoControlePressionado_Backup == true) {
250     botaoControlePressionado_Backup = false; //Reinicia estado do
251     botao
252     acionarBomba_Backup = !acionarBomba_Backup;
253     if (blynkConectado == true) {
254         Blynk.virtualWrite(V5, acionarBomba_Backup);
255     }
256 }
257
258 //Atualiza estado do botao de selecao no display
```

```

254     LCDselecaoAM = selecaoAutManValue == 0 ? "A" : "M";
255
256     //Impressao dos resultados no display LCD
257     lcd.setBacklight(HIGH);
258     lcd.clear();
259     lcd.setCursor(0, 0);
260     lcd.print(LCDMensagem);
261     lcd.setCursor(0, 1);
262     lcd.print(LCDTensao);
263     lcd.print("V");
264     lcd.setCursor(6, 1);
265     lcd.print(String(LCDCorrente, 1)); //Mostra a variavel com apenas
        uma casa decimal
266     lcd.print("A");
267     lcd.setCursor(15, 1);
268     lcd.print(LCDselecaoAM);
269
270     //=====
271     //Funcionamento do Sensor de Corrente
272     Irms = SCT013.calcIrms(1480); // Calcula o valor da Corrente 1172
273
274     potencia = Irms * tensao; // Calcula o valor da Potencia
        Instantanea
275
276     LCDCorrente = Irms;
277
278     //=====
279     //Funcionamento do Sensor de Tensao
280     tensaoMedida = SensorTensao.getRmsVoltage(20);
281
282     LCDTensao = tensaoMedida;
283
284     //=====
285
286     //Tratamento dos dados para acionar a bomba
287     if (selecaoAutManValue == 1) { //Se manual
288         falhaBomba = 0;
289         sistemaCorrompido = 0;
290         inicializacaoBomba = 0;
291         tentativasInicializacao = 0;
292         digitalWrite(relay_Pin, acionarBomba);
293         digitalWrite(relay_Pin_Backup, acionarBomba_Backup);
294         digitalWrite(led_bomba_ok, HIGH);
295         digitalWrite(led_bomba_falha, HIGH);
296
297     } else if ((selecaoAutManValue == 0) && (sistemaCorrompido == 0))
        { //Se automatico e nao corrompido

```

```
298 //Sistema ainda nao foi ligado
299 if (inicializacaoBomba == 0 && falhaBomba == 0) {
300     digitalWrite(relay_Pin, LOW);
301     digitalWrite(relay_Pin_Backup, LOW);
302     delay(tempoInicializacaoBomba); //intervalo para ligar a
        bomba
303     inicializacaoBomba = 1;           //armazena status que a bomba
        foi ativada
304
305     //Sistema esta ligado e dentro dos parametros
306 } else if ((inicializacaoBomba == 1) && (Irms >= correnteMinima
        && Irms <= correnteMaxima && tensaoMedida >= tensaoMinima)) {
307     digitalWrite(relay_Pin, LOW);
308     digitalWrite(relay_Pin_Backup, LOW);
309     inicializacaoBomba = 1; //armazena status que a bomba foi
        ativada
310     tentativasInicializacao = 0;
311
312     //Sistema esta ligado e fora dos parametros
313 } else if ((inicializacaoBomba == 1) && (Irms <= correnteMinima
        || Irms >= correnteMaxima || tensaoMedida <= tensaoMinima)) {
314     digitalWrite(relay_Pin, HIGH);
315     digitalWrite(relay_Pin_Backup, HIGH);
316     falhaBomba = 1; //ocorreu a falha
317     inicioTempoFalha = millis();
318     inicializacaoBomba = 0; //armazena status que a bomba foi
        desativada
319 }
320
321 if ((falhaBomba == 1) && (sistemaCorrompido == 0)) {
322     unsigned long tempoAtual = millis();
323     unsigned long diferencaTempo = tempoAtual - inicioTempoFalha;
324
325     if (diferencaTempo >= intervaloCorrecaoErro) {
326         falhaBomba = 0;
327         tentativasInicializacao++;
328     }
329 }
330
331 if (tentativasInicializacao >= 3) {
332     //Inicia o tempo ao identificar erro e o sistema ainda nao
        estava corrompido
333     sistemaCorrompido = 1;
334     falhaBomba = 1;
335     inicioTempoCorrompido = millis();
336     Serial.println("Sistema Corrompido!");
337 }
```

```

338 } else if ((selecaoAutManValue == 0) && (sistemaCorrompido == 1))
339 {
340     unsigned long tempoAtual = millis();
341     unsigned long diferencaTempo = tempoAtual -
342         inicioTempoCorrompido;
343
344     if (diferencaTempo >= intervaloCorrecaoCorrompido) {
345         falhaBomba = 0;
346         sistemaCorrompido = 0;
347         tentativasInicializacao = 0;
348     }
349 }
350
351 //Mostra mensagem de conexao no display
352 LCDMensagem = blynkConectado == true ? "Conectado!" : "
353     Desconectado!";
354
355 if (falhaBomba == 0 && selecaoAutManValue == 0) {
356     Serial.println("Bomba OK!");
357     digitalWrite(led_bomba_ok, HIGH);
358     digitalWrite(led_bomba_falha, LOW);
359 } else if (falhaBomba == 1 && sistemaCorrompido != 1) {
360     Serial.println("Falha na bomba...");
361     digitalWrite(led_bomba_falha, HIGH);
362     digitalWrite(led_bomba_ok, LOW);
363 } else if (sistemaCorrompido == 1) {
364     Serial.println("Sistema Corrompido!!!!");
365     digitalWrite(led_bomba_ok, LOW);
366     piscarLed(led_bomba_falha, 500);
367     LCDMensagem = blynkConectado == true ? "Sistema em Falha" : "
368         Desc. de Corromp.";
369 }
370
371 //=====
372 //Impressao dos resultados no monitor Serial
373 Serial.print("Corrente = ");
374 Serial.print(Irms);
375 Serial.println(" A");
376 Serial.print("Potencia = ");
377 Serial.print(potencia);
378 Serial.println(" W");
379
380 Serial.print("Tensao = ");
381 Serial.print(tensaoMedida);
382 Serial.println(" V");
383
384 //Sincroniza se o blynk esta conectado

```

```
381     if (blynkConectado == true) {
382         Blynk.virtualWrite(V1, Irms);           //Envia a corrente para o
           aplicativo blynk
383         Blynk.virtualWrite(V2, tensaoMedida); //Envia a tensao para o
           aplicativo blynk
384     }
385 }
386
387 void piscarLed(int set_led, int set_intervalo) {
388     digitalWrite(set_led, HIGH);
389     delay(set_intervalo);
390     digitalWrite(set_led, LOW);
391     delay((set_intervalo / 10));
392 }
393
394 void lerSelecaoBtnPressionado() {
395     botaoSelecaoPressionado = true;
396 }
397
398 void lerControleBtnPressionado() {
399     botaoControlePressionado = true;
400 }
401
402 void lerControleBtnPressionado_Backup() {
403     botaoControlePressionado_Backup = true;
404 }
```

Listing A.1 – Firmware do dispositivo