



PROJETO E CONSTRUÇÃO DE UMA ESTUFA MONITORADA POR REDE
DE COMUNICAÇÃO LORA INTEGRADA A IEEE802.11

Yann Machado da Silva

Projeto de Graduação apresentado ao Curso de Engenharia de Controle e Automação da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientadores: Lilian Kawakami Carvalho
Claudio Miceli de Farias

Rio de Janeiro
Junho de 2022

PROJETO E CONSTRUÇÃO DE UMA ESTUFA MONITORADA POR REDE
DE COMUNICAÇÃO LORA INTEGRADA A IEEE802.11

Yann Machado da Silva

PROJETO DE GRADUAÇÃO SUBMETIDO AO CORPO DOCENTE DO
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO DA ESCOLA
POLITÉCNICA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO
PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU
DE ENGENHEIRO DE CONTROLE E AUTOMAÇÃO.

Examinado por:

Lilian K. Carvalho

Prof. Lilian Kawakami Carvalho, D.Sc.

Claudio Miceli de Farias

Prof. Claudio Miceli de Farias, D.Sc.

Marcos V. de Brito M.

Prof. Marcos Vicente de Brito Moreira, D.Sc.

Emanuele Nunes de Lima Figueiredo Jorge

Prof. Emanuele Nunes de Lima Figueiredo Jorge, M.Sc.

RIO DE JANEIRO, RJ – BRASIL

JUNHO DE 2022

Machado da Silva, Yann

Projeto e construção de uma estufa monitorada por rede de comunicação LORA integrada a IEEE802.11/Yann Machado da Silva – Rio de Janeiro: UFRJ/ Escola Politécnica, 2022.

XI, 74 p.: il.; 29, 7cm.

Orientadores: Lilian Kawakami Carvalho

Claudio Miceli de Farias

Projeto de Graduação – UFRJ/ Escola Politécnica/
Curso de Engenharia de Controle e Automação, 2022.

Referências Bibliográficas: p. 42 – 44.

1. Estufa.
 2. IoT.
 3. LoRa.
 4. Monitoramento.
 5. Controle.
- I. Kawakami Carvalho, Lilian *et al.* II. Universidade Federal do Rio de Janeiro, Escola Politécnica, Curso de Engenharia de Controle e Automação. III. Título.

Agradecimentos

A jornada para finalização do Trabalho de Conclusão de Curso e, consequentemente, da graduação foi longa e árdua e eu certamente não conseguiria ter feito isso sozinho. Por isso, dedico esse capítulo a todos que me fortaleceram ao longo dessa jornada e me fizeram chegar aonde cheguei.

Primeiramente, tenho que agradecer aos meus pais e meu irmão. Sem eles e sua influência no meu crescimento, eu não seria nem metade do ser humano que sou hoje. Devo tudo à eles por terem me ajudado de todas as formas possíveis para eu alcançar o que alcancei em toda minha vida.

Em seguida, gostaria de agradecer especialmente a minha namorada que entrou na minha vida pouco depois de eu iniciar na faculdade e permanece nela até agora. Sem você eu certamente não conseguiria suportar todo o trajeto que foi chegar até aqui e por isso, e por muitos outros motivos, te agradeço muito.

Não posso deixar faltar o agradecimento a todos os meus amigos. Primeiramente aos meus amigos do colégio, que estão na minha vida há anos e vão certamente permanecer por mais anos a vir, que estiveram presentes em muitos momentos importantes na minha vida mesmo antes de eu entrar na faculdade. Aos meus amigos da T20, dedico esse trabalho a todos vocês, que tanto me ajudaram ao longo de todos esses sofridos anos, os quais eu não conseguiria ter passado se não fosse vocês. E, por último, mas não menos importante, aos meus amigos da Equipe de Competição MinervaBots, que rapidamente se tornaram partes importantíssimas da minha vida.

Tenho muito a agradecer também às experiências que pude ter na faculdade, em especial à Equipe de Competição MinervaBots, que me fez crescer como profissional e como pessoa e abriu meus olhos para um mundo que eu ainda não tinha percebido.

E, por fim, gostaria de agradecer a todos os professores do curso de Engenharia de Controle e Automação, em especial aos professores e meus orientadores Cláudio Miceli e Lilian Kawakami, que me acompanharam ao longo de toda a trajetória do Projeto Final e me auxiliaram de todas as formas que puderam.

Meus mais sinceros agradecimentos a todos que fizeram parte dessa jornada!

Resumo do Projeto de Graduação apresentado à Escola Politécnica/ UFRJ como parte dos requisitos necessários para a obtenção do grau de Engenheiro de Controle e Automação.

PROJETO E CONSTRUÇÃO DE UMA ESTUFA MONITORADA POR REDE DE COMUNICAÇÃO LORA INTEGRADA A IEEE802.11

Yann Machado da Silva

Junho/2022

Orientadores: Lilian Kawakami Carvalho

Claudio Miceli de Farias

Curso: Engenharia de Controle e Automação

O projeto apresentado neste trabalho busca mostrar como aplicações básicas de tecnologias IoT em conjunto com algoritmos de controle simples podem melhorar a qualidade de plantações de horticultores dentro do nicho da agricultura urbana. Para alcançar esse objetivo, foi projetada e construída uma estufa inteligente, denominada ELI, que tem alguns de seus parâmetros monitorados, através dos protocolos de comunicação LoRa e WiFi, e controlados por controles aplicados localmente.

Abstract of Undergraduate Project presented to POLI/UFRJ as a partial fulfillment
of the requirements for the degree of Engineer.

Yann Machado da Silva

June/2022

Advisors: Lilian Kawakami Carvalho

Claudio Miceli de Farias

Course: Automation and Control Engineering

The project presented in this work showed how basic IoT technologies together with a simple control algorithm can improve the quality of crops inside the urban agriculture scope. To accomplish this goal, a smart greenhouse denominated ELI was designed and built, that had its parameters controlled by local controllers and monitored by the communications protocols LoRa and Wifi.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
1 Introdução	1
2 Conceitos Básicos	7
2.1 Sistemas Ciber-físicos	7
2.2 Ambientes inteligentes	8
2.3 Agricultura inteligente baseada em IoT	8
2.4 IoT - Internet das Coisas	9
2.4.1 RFID - Identificação por Rádio Frequência	10
2.4.2 Redes de sensores sem fio	10
2.5 LoRa e LoRaWAN	11
3 Trabalhos Relacionados	13
4 ELI - Estufa Lora Inteligente	16
4.1 Motivação e objetivo	16
4.2 Desenvolvimento	17
4.2.1 Monitoramento e Controle	18
4.2.1.1 Arquitetura física	18
4.2.1.2 Arquitetura Eletrônica	20
4.2.1.3 Lógica de funcionamento	23
4.2.2 Comunicação e Postagem	25
4.2.2.1 Estrutura Física	25
4.2.2.2 Arquitetura Eletrônica	26
4.2.2.3 Lógica de Funcionamento	27
4.3 Resultados	28
4.3.1 Distância de comunicação	37

5 Conclusão	39
5.1 Trabalhos Futuros	40
Referências Bibliográficas	42
A Projeto das estruturas da Estufa e do Módulo de Controle	45
B Projeto da Placa de Potência	49
C Projeto da Placa Lógica	51
D Código do Módulo de Controle	53
E Projeto da Placa do Receptor	60
F Código do Receptor	62
G Dados do experimento	69

Lista de Figuras

1.1	Diagrama expondo as possibilidades de aplicação das tecnologias IoT	2
1.2	Exemplo de IoT na agricultura	3
1.3	Horta Comunitária do Dirceu, em Teresina. Maior horta comunitária da América Latina	4
1.4	Diagrama simplificado do Projeto	5
2.1	Diagrama de funcionamento do protocolo LoRa e da rede LoRaWAN	11
4.1	Diagrama de funcionamento	17
4.2	Estrutura da Estufa	19
4.3	Estrutura do Módulo de Controle	20
4.4	Circuitos utilizados para a regulação das tensões	21
4.5	Placas Lógica (a esquerda) e de Potência (a direita)	22
4.6	Diagrama de blocos da lógica	24
4.7	Receptor	25
4.8	Placas utilizadas no projeto	26
4.9	Diagrama de funcionamento do bloco de Comunicação e Postagem	27
4.10	Gráfico das cargas da bateria pelo tempo	29
4.11	Dados da Temperatura do Ar	31
4.12	Dados da Umidade do Ar	32
4.13	Dados de umidade do ar e temperatura do ar cruzados	32
4.14	Dados da Umidade do Solo	33
4.15	Dados de umidade do solo e nível do recipiente cruzados	34
4.16	Resultado do experimento	35
4.17	Dados da temperatura do ar ao longo do tempo	36
4.18	Dados da umidade do ar ao longo do tempo	37
4.19	Dados da umidade do solo ao longo do tempo	37
4.20	Dados do sensor de nível ao longo do tempo	37
A.1	Modelo da tampa e vista superior detalhada	46
A.2	Vistas 2D	47
A.3	Modelo 3D	48

B.1	Esquemático da Placa de Potência	49
B.2	PCB da Placa de Potência	50
B.3	Modelo 3D da Placa de Potência	50
C.1	Esquemático da Placa Lógica	51
C.2	PCB da Placa Lógica	52
C.3	Modelo 3D da Placa Lógica	52
E.1	Esquemático da placa do Receptor	60
E.2	PCB da placa do Receptor	61
E.3	Modelo 3D da placa de Receptor	61

Lista de Tabelas

4.1 Carga das baterias 1 (Módulo de Controle) e 2 (Receptor) ao longo do tempo	29
G.1 Dados gerados no segundo experimento	74

Capítulo 1

Introdução

Indústria 4.0 é a nova Revolução Industrial na qual busca-se uma integração entre as operações de manufatura em geral e as tecnologias de comunicação e informação, formando os elementos conhecidos como Sistemas Ciber-físicos [1]. Com essa integração, esses sistemas conseguiram experienciar e aprender com o mundo a sua volta através de dados que receberiam de sensores variados, melhorando sua performance e, consequentemente, otimizando a produção de forma geral [2].

A Internet das Coisas (do inglês *Internet of Things*, mais popularmente conhecida como IoT) é o conjunto de tecnologias chave dentro do contexto de Indústria 4.0, uma vez que é utilizada como ferramenta para conseguir alcançar os Sistemas Ciber-físicos. Isso é feito através do uso de sistemas de sensoriamento que se comunicam através de protocolos de rede sem fio, que tornam possível realizar a integração de objetos variados à internet.

Com o passar do tempo e o desenvolvimento das tecnologias, os sensores e seus custos energéticos ficaram cada vez menores e mais eficientes. Isso fez com que o termo “Coisas” (traduzido do inglês *Things*, de *Internet of Things*) passasse a ser cada vez mais abrangente, ou seja, a utilização desse tipo de tecnologia se expandiu para diversas áreas de aplicabilidade, visto que se tornou possível integrar objetos que antes eram inviáveis à rede, como bem abordado em [3].

O resultado da utilização de tecnologias IoT é a geração de uma grande quantidade de dados que podem ser analisados e utilizados de diversas formas dependendo

da necessidade da aplicação. O diagrama da figura 1.1 mostra exemplos de setores e áreas nas quais a IoT já está presente.

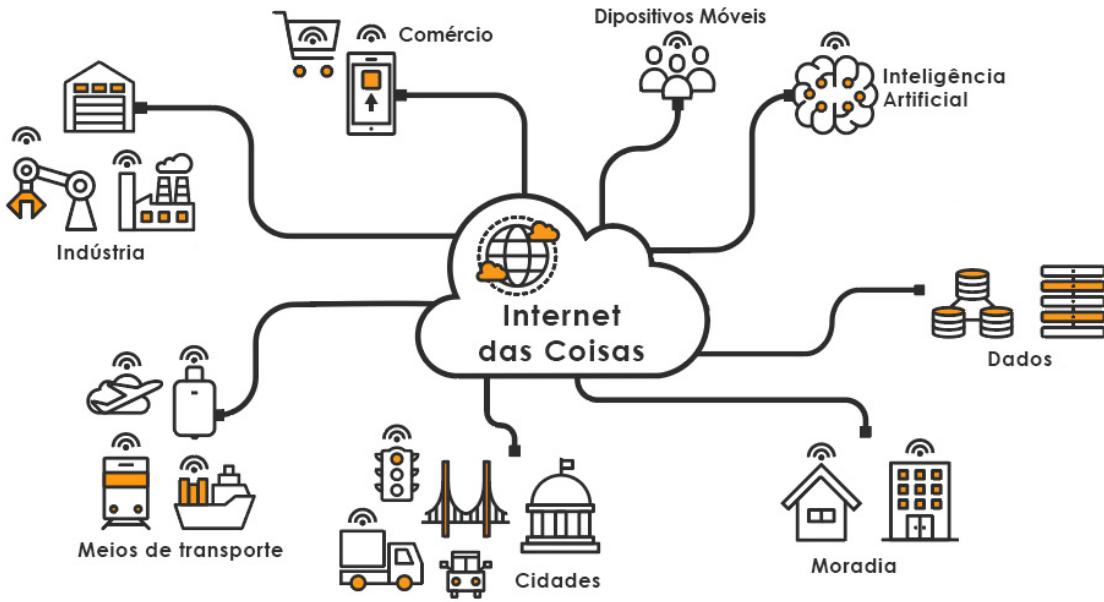


Figura 1.1: Diagrama expondo as possibilidades de aplicação das tecnologias IoT ¹

Dentre as áreas de aplicabilidade, a agricultura foi uma que se beneficiou bastante com o avanço das tecnologias IoT, já que é uma técnica que se beneficia muito com o monitoramento de parâmetros chaves, como umidade do solo/ar, temperatura, composição do solo, etc. A Agricultura de Precisão, por exemplo, é uma técnica que utiliza de informações e tecnologias para gerenciar as inconsistências de plantações, com o intuito de aumentar a rentabilidade, sustentabilidade e saúde do plantio [4]. As tomadas de decisão desse gerenciamento são baseadas em dados e com o progresso das tecnologias IoT, a quantidade e qualidade desses dados aumenta, uma vez que novos tipos de sensoreamentos são possíveis, tornando o processo decisório cada vez mais eficaz.

Um exemplo da aplicação dessa técnica foi apresentado em [5]. Em que foram utilizados, sensores para monitoramento de uma doença específica que ataca plantações de batata e tomate. Com os dados desses sensores, foi feita uma análise preditiva cujo resultado serviu para aconselhar os fazendeiros, de forma que conse-

¹Imagem editada pelo autor, original retirada de URL:<https://www.tibco.com/pt-br/reference-center/what-is-the-internet-of-things-iot>; Acesso em: 25/10/2021

guissem evitar ou conter o avanço da doença. Essa aplicação é um exemplo entre muitos das possibilidades das tecnologias IoT dentro do ramo da agricultura. A figura 1.2 mostra um diagrama simples de um sistema exemplo, no qual pode-se ver os sensores (ou nós) sendo representados pelos pontos, que enviam dados para os gateways (representados pelos símbolos de roteador) que, por sua vez, enviam os dados para nuvem, onde esses dados são tratados e postados em uma aplicação qualquer, que é representada no tablet.



Figura 1.2: Exemplo de IoT na agricultura²

Um nicho muito interessante dentro do ramo da agricultura e que tem muito espaço para aplicações de tecnologias IoT é a Agricultura Urbana. A Agricultura Urbana é definida como uma prática agrícola que ocorre dentro ou na periferia das áreas urbanas, sendo responsável por grande parte do abastecimento de alimentos frescos dos mercados das grandes cidades. Exemplos possíveis são pequenas fazendas urbanas, plantações em terraços, compartilhamento de terras, produção de alimentos em espaço público e outros [6]. Por causa do crescimento constante dos grandes centros urbanos, essa prática tem se valorizado cada vez mais, visto que traz benefícios valiosos. Alguns exemplos desses benefícios são: do ponto de vista ecológico, a presença de mais áreas verdes dentro do perímetro urbano que ajuda na

²Imagen editada pelo autor, original retirada de URL: <https://www.biz4intellia.com/blog/5-applications-of-iot-in-agriculture/>; Acesso em: 25/10/21

manutenção do microclima e ainda auxilia na redução da poluição do ar; do ponto de vista social, a prática ajuda na nutrição das famílias de renda mais baixa que se envolvem com ela, além de fornecer um retorno financeiro para essas famílias [7].

A figura 1.3 mostra um exemplo de Agricultura Urbana, uma horta comunitária em Teresina, no Amapá, que auxilia na renda de várias famílias, como foi verificado no estudo de caso [8].



Figura 1.3: Horta Comunitária do Dirceu, em Teresina. Maior horta comunitária da América Latina³

No entanto, como é exposto em [7] e [9], a Agricultura Urbana não tem só pontos positivos. Por estar localizada em áreas urbanas, essa técnica traz preocupações diferentes do que as abordadas na agricultura latifundiária. Os fatores de risco que precisam ser levados em consideração quando se discute Agricultura Urbana são: a poluição do ar alta demais no local em que alimentos são produzidos, poluição da rede hídrica que irriga as plantações, contaminações do solo na região de plantio não percebidas, entre outros. Tais fatores de risco quando não levados em consideração podem levar a sérios problemas de saúde tanto para os consumidores finais, quanto para os horticultores, que também consomem do seu próprio plantio. A utilização

³URL: <https://www.oitomeia.com.br/noticias/2017/09/20/horta-do-dirceu-garante-bem-estar-social-e-alimentacao-saudavel-para-moradores/>; Acesso em: 14/11/21

de tecnologias IoT conseguiria ajudar a prever esses possíveis problemas antes que eles se desenvolvessem e impactassem de forma contundente os envolvidos, além de facilitar bastante o cultivo para os horticultores.

O projeto proposto, desenvolvido e estudado neste trabalho é uma miniestufa denominada ELI (Estufa Lora Inteligente) que através de tecnologias IoT tem três parâmetros monitorados, são eles: umidade do ar e do solo e temperatura do ar. Além disso, algumas ações são automatizadas: abertura e fechamento da porta da estufa(para regular a saída e entrada de ar fresco) e irrigação. A ideia deste projeto é conseguir mostrar como as tecnologias IoT podem auxiliar nos problemas que a agricultura urbana apresenta, através de monitoramentos e atuações simples, e também mostrar como a vida dos horticultores e a eficácia das suas plantações podem ser melhoradas através do uso dessas tecnologias.

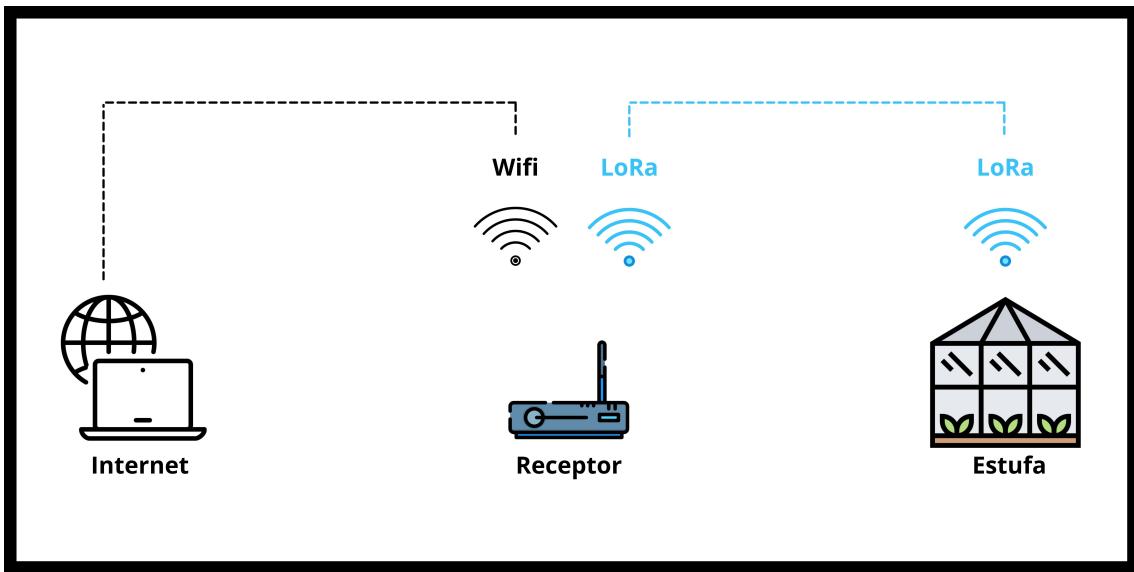


Figura 1.4: Diagrama simplificado do Projeto

O sistema foi desenvolvido utilizando o protocolo de comunicação LoRa que é um tipo de LPWAN (*Low Power Wide Area Network*) muito conhecido dentro do cenário IoT para aplicações que precisam de comunicações de longa distância, baixo custo energético e pouca necessidade de alta taxa de transferência de dados. A comunicação LoRa é feita entre dois sistemas eletrônicos que tem como componentes microcontroladores ESP32 que possuem o chip SX1276/SX1278 de comunicação LoRa, estando um deles na estufa e o outro deles no receptor. Esse receptor cumpre

o papel de Gateway, que é o sistema que conecta esses dados à internet. Para fazer a conexão do módulo receptor à internet foi escolhido o protocolo de comunicação WiFi. A figura 1.4 mostra um diagrama de funcionamento simplificado do projeto.

É importante evidenciar que o projeto desenvolvido neste trabalho não monitora todas as variáveis que poderiam ser exploradas dentro de um contexto de plantio. A motivação para o objeto de estudo ser uma miniestufa veio da vontade de simular um ambiente controlado no qual se quer fazer um plantio qualquer. É importante ressaltar que a tecnologia utilizada neste trabalho pode ser utilizada em outros casos, como, por exemplo, em hortas abertas e vasos de plantas. Além disso, os sensores utilizados devem ser escolhidos de acordo com a necessidade da aplicação.

Capítulo 2

Conceitos Básicos

Neste capítulo serão dadas explicações sobre temas importantes para o entendimento do trabalho proposto. Primeiramente, na seção 2.1, Sistemas Ciber-físicos são abordados para contextualizar o leitor a respeito do que eles são e por que eles são o objetivo final da Indústria 4.0. Em seguida, nas seções 2.2 e 2.3, será dada uma explicação, respectivamente, do que são ambientes inteligentes e agricultura inteligente, por que eles tem esse nome e qual o papel dos dois no contexto tecnológico atual. E, por fim, nas seções 2.4 e 2.5 serão abordados o que é IoT e quais os principais tipos tecnologias que a compõe e algumas informações mais detalhadas sobre o protocolo de comunicação LoRa.

2.1 Sistemas Ciber-físicos

Sistemas Ciber-físicos são sistemas físicos e construídos pelo homem cujo operação é monitorada, coordenada, controlada e integrada por um núcleo de computação e informação. Essa junção íntima entre o mundo cibernetico e físico é manifestada desde um mundo nano, até sistemas de grande escala em áreas vastas. Há uns anos atrás, a internet revolucionou como os humanos acessam, interagem e geram informação. Analogamente, os sistemas ciber-físicos vão transformar como os humanos interagem e controlam o mundo físico a sua volta [10]. Exemplos de aplicação de sistemas ciber-físicos incluem sistemas e aparelhos médicos, sistemas aeroespaci-

ais, rodovias inteligentes, sistemas de robótica, controle de processos, automação de fábricas, espaços inteligentes, entre outros [10].

Dentro de um contexto de Indústria 4.0, os sistemas ciber-físicos serão aqueles que possibilitarão as empresas a terem produções flexíveis e mais estratégicas, através de análises em tempo real de dados [1]. E o caminho para conseguir isso é construir máquinas que sejam capazes de aprender sozinhas com o ambiente a sua volta, através de dados de sensores [2].

2.2 Ambientes inteligentes

Um ambiente inteligente pode ser definido como um espaço que, equipado com diversos dispositivos, consegue monitorar e atuar sobre o ambiente, de acordo com as mudanças e os habitantes presentes no local, de forma que a vida das pessoas que o habitem se torne mais confortável [11].

Exemplos simples de coisas que formam um ambiente inteligente são: controle remoto de dispositivos como lâmpadas, cafeteiras e outros; a comunicação entre dispositivos, de forma que a percepção do ambiente consiga ser mais acurada e as experiências mais completas, por exemplo, a cafeteira ligar no momento em que o alarme tocar, ações correlacionadas que farão diferença na manhã de um usuário que sempre toma café ao acordar [11]. Além dessas, ambientes inteligentes podem ter aplicações mais complexas, todas buscando perceber melhor o ambiente para conseguir agir da melhor forma possível, tendo como objetivo deixar cada vez mais confortável e conectada a vida dos habitantes.

2.3 Agricultura inteligente baseada em IoT

Analogamente a explicação de ambientes inteligentes, a agricultura inteligente é aquela na qual parâmetros chaves são monitorados e as atuações necessárias são feitas sem necessidade de atuação humana, de forma que a vida do agricultor seja mais fácil e que a produção agrícola consiga ser a mais efetiva possível. Quanto

maior a quantidade de dados coletada e melhor a inteligência por trás dos sistemas implementados, melhores são as tomadas de decisão do processo.

2.4 IoT - Internet das Coisas

Não existe uma definição única para IoT que seja aceita por toda a comunidade [12]. Uma das possibilidades de definição, apresentada em [3], é a interconexão de sensores e atuadores que dão a possibilidade de compartilhar informações através de plataformas unificadas, desenvolvendo uma figura comum para aplicações inovadoras. Isso seria possível através de sensoreamentos ubíquos, análise de dados e monitoramento de informação, tudo isso unificado por um ambiente em nuvem. Outra possível definição, trazida em [12], é uma rede aberta e comprehensível de objetos inteligentes que tem a capacidade de se auto organizar, compartilhar informações, dados e recursos, reagindo e agindo diante situações e mudanças no ambiente. No entanto, o que todas as definições tem em comum, é a ideia de que a primeira versão da internet é aquela na qual dados eram criados por pessoas e a segunda, é aquela na qual dados são gerados por coisas [12].

Acredita-se que o nome IoT foi originado de uma apresentação que Kevin Ashton fez para a empresa P&G em 1999, como ele explica em [13]. Nessa mesma publicação, ele traz ideias sobre pontos muito interessantes, em 2009, de como a maioria dos dados que eram gerados na época eram produzidos majoritariamente por seres humanos. E vislumbra um futuro no qual o computador consegue experienciar o mundo por si próprio, adquirindo dados sem a necessidade de auxílio humano, o que possibilitaria melhores análises e, consequentemente, melhores tomadas de decisão.

Essa ideia trazida por Ashton vai ao encontro com a afirmação proposta em [12] e reafirma o cerne da ideia de IoT, que seria a mudança do núcleo de geração de dados saindo dos humanos e indo para as coisas, podendo “Coisas” ser qualquer tipo de objeto. Para possibilitar que esses objetos sejam ligados à rede, sistemas IoT se baseiam em tecnologias RFID (Identificação por Rádio Frequência) e redes de sensores sem fio, de forma que o ambiente possa estar plenamente conectado à

rede de forma “invisível” [3].

2.4.1 RFID - Identificação por Rádio Frequência

RFID foi um termo cunhado para tecnologias de radio frequência de curto alcance, para comunicar majoritariamente informações digitais, entre uma localidade estacionária e objetos em movimento ou dois objetos em movimento. A tecnologia é normalmente caracterizada por um componente mais complexo em uma ponta da conexão e um mais simples na outra. Os componentes mais simples são normalmente chamados de tags ou transponders e são pequenos e de baixo custo, em geral alocados em larga escala [14]. A vantagem desses tags é que eles podem funcionar indefinidamente, por que não precisam estar conectadas à energia, diferente dos receptores. A alimentação desses tags é feita pelo receptor da seguinte forma: ao fazer uma requisição, o receptor envia um sinal que é captado pela antena do tag, a energizando. Como consequência de ser alimentado, o tag envia uma resposta de volta ao receptor contendo seu ID [15].

2.4.2 Redes de sensores sem fio

A outra forma de comunicação IoT são as comunicações usando redes de sensores sem fio. As redes de sensores sem fio são compostas por vários nós de sensores que são alocados próximos ou dentro do fenômeno o qual se quer medir. A posição desses sensores não precisa ser predeterminada ou projetada, o que permite a alocação aleatória desses sensores em ambientes de difícil acesso. Além disso, isso faz com que essas redes tenham que ter protocolos e algoritmos que as façam ter capacidade de se auto-organizar. Outra diferença das redes de sensores é que os nós possuem um poder de processamento próprio (mesmo que pequeno), então ao invés deles enviarem o dado cru, eles enviam apenas o dado necessário e já pré-processado, agilizando o processo dos outros sistemas que fariam as fusões/processamentos dos dados [16].

2.5 LoRa e LoRaWAN

LoRa(*Long Range*) é uma técnica de modulação de espectro espalhado (*spread spectrum modulation*) desenvolvida pela Semtech, que é derivada da técnica CSS(*chirp spectrum modulation*) [17]. A comunicação LoRa é um exemplo de tecnologia LPWA(*Low Power Wide Area*) que são tecnologias que conseguem proporcionar comunicações de longa distância a preços mais baixo e baixo custo energético [18].

A comunicação LoRa define a camada física e a rede LoRaWAN define o protocolo de comunicação e a arquitetura do sistema. O protocolo LoRaWAN utiliza de uma topologia estrela, na qual os nós finais utilizam os gateways como receptores que transmitem os dados para a rede. Os nós finais não mandam mensagens para um gateway específico, ao invés disso eles enviam mensagem para todos os gateways próximos e todos eles enviam essas mensagens para a rede em nuvem. Essa rede tem a inteligência necessária para filtrar os possíveis pacotes repetidos, checar a segurança, enviar resposta de recebimento para os gateways e enviar esses dados para o servidor de aplicação certo [18].

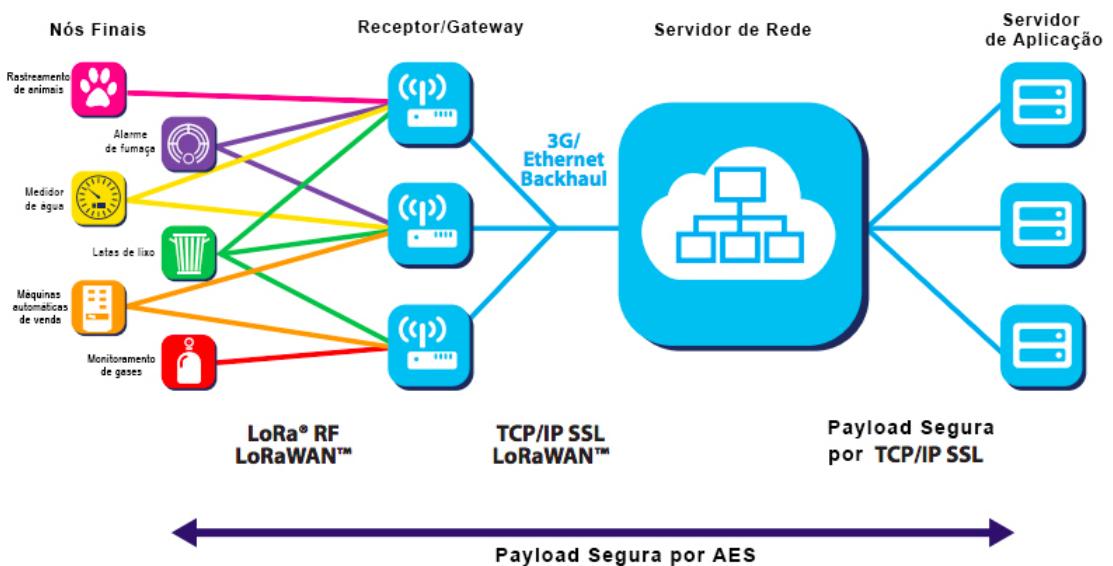


Figura 2.1: Diagrama de funcionamento do protocolo LoRa e da rede LoRaWAN¹

¹Imagem editada pelo autor, original retirada de URL: <https://medium.com/coinmonks/lpwans-lora-lorawan-and-the-internet-of-things-aed7d5975d5d>; Acessado em: 21/12/2021

A Figura 2.1 mostra um diagrama do fluxo de comunicações do protocolo LoRa e a rede LoRaWAN. Mais a esquerda é possível ver todos os nós finais que podem ser sensores de todos os tipos. Logo após, estão os gateways em contato direto com esses nós finais. Como foi mencionado, cada nó envia dado para mais de um gateway diferente. Esses gateways transmitem os dados para a nuvem através do protocolo TCP/IP e de lá esses dados são enviados para os servers de aplicação, ou seja, as aplicações finais.

Os nós finais da rede são divididos em 3 classes diferentes que separam de acordo com a necessidade energética e de latência do sistema. Os nós classe A são sensores alimentados por baterias, os nós de classe B são atuadores alimentados por baterias e os nós de classe C são atuadores ligados diretamente à rede de energia. Eles operam de forma a otimizar o tempo em que as janelas de recepção de transmissão dos nós ficam disponíveis, isto é, quanto menor a autonomia energética do dispositivo, menos disponível ele vai ficar para receber possíveis transmissões, sendo o contrário também válido. Os dispositivos classe A são os que tem a maior latência e a maior eficiência energética, os de classe B possuem uma latência menor que os A e uma autonomia energética também menor que os de classe A e, por fim, os de classe C possuem a menor eficiência energética e a menor latência do que os demais [18].

Capítulo 3

Trabalhos Relacionados

Neste capítulo serão discutidos brevemente sobre alguns trabalhos da literatura cujos temas são relacionados à utilização de IoT na agricultura, assunto principal do trabalho proposto.

CULMAN *et al.* [19] propuseram uma solução de monitoramento de plantações de palmeiras utilizando redes de sensores sem fio para aquisição de dados e um aplicativo web. De acordo com o artigo, o objetivo do trabalho era coletar e centralizar dados de umidade do solo e postar esses dados em uma interface web, de forma que conseguissem ser monitorados. Os autores deixam claro que a motivação do trabalho partiu da ausência de monitoramento em tempo real de dados de umidade do solo em áreas de plantações de palmeiras. Algumas das tecnologias utilizadas nesse trabalho são: plataforma Arduíno para desenvolvimento do código, módulos de radio XBee, que utilizam o protocolo de comunicação ZigBee, e redes de celular 3G e 4G. As tecnologias utilizadas nesse trabalho e a forma de implementação demonstram que a aplicação necessitava de autonomia energética e tecnologia de baixo custo.

A aplicação mencionada no parágrafo acima tem um caráter puramente de monitoramento, no entanto, como mencionado anteriormente, há inúmeras formas de utilizar os dados que são gerados através do uso de tecnologias IoT. Um exemplo diferente de utilização desses dados foi apresentado por ARABY *et al.* [5]. Nesse trabalho, os autores desenvolveram um sistema que recolhesse dados de plantações e então inseriram esses dados em um algoritmo de machine learning que dava uma

previsão de se a plantação estava infectada e, se sim, em quanto tempo a doença espalharia e quando seria inteligente o responsável começar a aplicar pesticidas para conter a doença. Essa previsão era apresentada junto com os dados em uma interface gráfica para o fazendeiro e o projeto tinha como objetivo otimizar a produção agrícola, através da redução das infecções e do uso desnecessário de pesticidas. As tecnologias principais utilizadas na aquisição de dados desse trabalho foram: NodeMCU (microcontrolador que recebia os dados dos sensores) e os protocolos de comunicação MQTT e WiFi para envio de dados entre os sensores, gateway e a internet. Nessa aplicação, a análise dos dados era feita diretamente no Gateway, já apresentando para o usuário final uma ação que precisava ser tomada.

Nos trabalhos CULMAN *et al.* [19] e ARABY *et al.* [5], mencionados anteriormente, as malhas de controle do sistema não foram fechadas, isto é, não houve nenhum tipo de atuação, apenas monitoramento. No trabalho de MUANGPRATHUB *et al.* [20] é proposto um sistema que utiliza redes de sensores sem fio para otimizar a irrigação de plantações. Como explicado pelos autores, o objetivo do trabalho era não só o monitoramento de parâmetros chaves para a plantação, como também a atuação. O projeto era composto de três principais componentes: a parte de hardware, que era responsável pela aquisição de dados e o controle local da irrigação, a parte web, que armazenava e analisava os dados e por último, a parte de aplicação para dispositivos móveis (ou app) que tinha por objetivo fornecer a possibilidade de irrigação manual do fazendeiro, caso ele preferisse fazê-la manualmente e não automaticamente. Através dos dados gerados pelos sensores, a aplicação fazia um controle da umidade do solo, buscando deixar sempre a umidade ótima para o crescimento da plantação. Além disso, os dados armazenados também eram usados para gerar modelos que previam os parâmetros ótimos para o crescimento das plantações.

Um outro trabalho que chamou atenção por ter malhas de controle fechadas e utilizar estufas como objeto de estudo foi o apresentado por PALLAVI *et al.* [21]. Nesse trabalho, os autores buscaram fazer um processo completo de monitoramento e controle de parâmetros chaves como: temperatura, umidade do solo, luz e concen-

tração de CO₂. A atuação foi focada na abertura/fechamento das portas e janelas da estufa e essas atuações eram baseadas nos valores dados pelos sensores e setpoint predefinidos pelos usuários. Para alcançar os objetivos propostos, a aplicação utilizou a tecnologia WiFi como protocolo principal para as comunicações. A utilização do protocolo WiFi, no entanto, reduz a área na qual a aplicação pode estar alocada.

Para conseguir contornar esse problema que é gerado quando se usa protocolos de curto alcance, a utilização de outros protocolos se torna necessária. O trabalho apresentado por GUTIÉRREZ *et al.* [22] traz um bom exemplo de aplicação que utiliza tecnologia LoRa, protocolo que é muito usado atualmente para sanar problemas de alcance. Nesse trabalho, os autores desenvolveram um sistema de monitoramento agrícola utilizando LoRa e LoRaWAN. Algumas das tecnologias utilizadas nesse trabalho foram: ESP32 WiFi LoRa, da Heltec; sensor DHT11 de temperatura e umidade; Raspberry PI3 B+; gateway HT-M01 Mini LoRa Gateway da Heltec. Como os autores deixam claro na conclusão, a maior vantagem desse sistema desenvolvido por eles com relação a outros sistemas agrícolas IoT é o grande alcance e a mobilidade que torna o sistema possível ter um gateway móvel, que lerá todos os sensores conforme passasse por eles e enviaria os dados para a nuvem.

Dentre os trabalhos exemplificados, a ELI seria uma junção do trabalho feito por GUTIÉRREZ *et al.* [22], que utilizou tecnologia LoRa para monitoramento e do trabalho desenvolvido por MUANGPRATHUB *et al.* [20], que faz um sistema de controle de irrigação. Nenhum dos trabalhos exemplificados uniu o monitoramento utilizando IoT com o controle local, como foi feito na ELI.

Capítulo 4

ELI - Estufa Lora Inteligente

Nesse capítulo será detalhado o desenvolvimento da ELI e seus resultados. Inicialmente, na seção 4.1, será apresentado a motivação e o objetivo do projeto. Em seguida, na seção 4.2 será detalhado seu desenvolvimento e funcionamento. Por fim, na seção 4.3, serão discutidos os resultados.

4.1 Motivação e objetivo

A motivação principal deste trabalho foi mostrar como a agricultura urbana pode se beneficiar com soluções simples de monitoramento e controle utilizando tecnologias IoT. Dessa forma, o objetivo do trabalho foi construir uma miniestufa inteligente que utiliza tecnologias IoT para conseguir ser feito um monitoramento e controle de forma remota. O sistema foi desenvolvido utilizando os protocolos de comunicação WiFi e LoRa, que funcionam em conjunto para conseguir levar os dados dos sensores alocados na estufa até uma página web, possibilitando o seu monitoramento. Além disso, a ELI foi projetada para ter autonomias hídrica e energética, de forma a reduzir o tempo humano utilizado no plantio.

4.2 Desenvolvimento

Os componentes escolhidos para o desenvolvimento da ELI foram: 2 ESP32 WiFi LoRa da Heltec, 1 sensor de temperatura e umidade DHT11, 1 sensor capacitivo de umidade do solo, 1 sensor de nível, 1 mini bomba de água 12V RS358, 1 driver de motor L298N, 1 relé, 1 motor DC 35V e 2 baterias LiPo 4S de 1300mAh. Um diagrama do funcionamento geral do projeto pode ser visto na figura 4.1, na qual pode se observar os módulos de Comunicação e Postagem (à esquerda) e de Monitoramento e Controle (à direita).

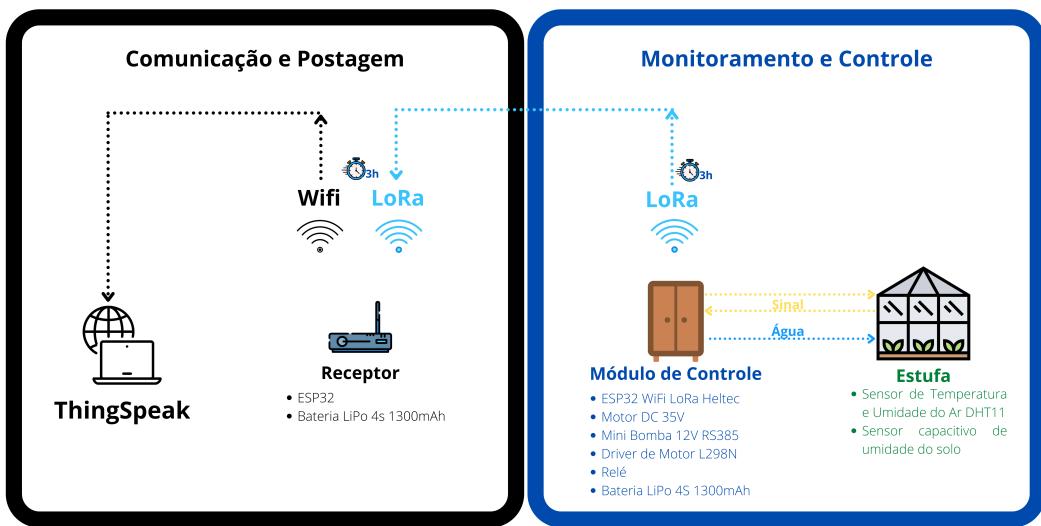


Figura 4.1: Diagrama de funcionamento

Como pode ser visto no módulo de Monitoramento e Controle, na Estufa ficam alocados dois dos sensores e no Módulo de Controle acoplado a Estufa ficam o ESP32, o motor, a bomba, o relé, o sensor de nível e uma das baterias, de forma que os dados possam ser aquisitados através dos sensores na Estufa, levados até o Módulo e enviados para o Receptor. O Módulo de Controle também é responsável por analisar os dados e verificar se alguma ação é necessária, ativando os atuadores quando preciso. O sistema de Monitoramento e Controle ativa a cada 3 horas, verifica a necessidade de irrigação do solo ou abertura da janela da estufa e, em seguida, envia os dados para o Receptor, e esse processo é repetido ao longo de 1 minuto. Já no módulo de Comunicação e Postagem, temos o Receptor no qual

estão alocados o segundo ESP32 e a segunda bateria e que tem funcionamento bem parecido com o do outro módulo. Ele ativa a cada 3 horas, recebe os dados que estão sendo enviados pelo bloco de Monitoramento e Controle e posta os dados na plataforma ThingSpeak também repetindo ao longo de 1 minuto. Ao fim desse minuto, os dois sistemas voltam a dormir.

A seguir será detalhado cada um dos grandes blocos de desenvolvimento do projeto: Monitoramento e Controle (seção 4.2.1) e Comunicação e Postagem (seção 4.2.2).

4.2.1 Monitoramento e Controle

Nesta seção é apresentada a arquitetura física, a arquitetura eletrônica e a lógica do projeto focada na aquisição de dados e controle da estufa.

4.2.1.1 Arquitetura física

Toda a arquitetura física do projeto foi feita de madeira sob medida. Como é possível observar na figura 4.2a a estufa é formada por quatro hastes de madeira fixadas em uma base paralelepipedal, cujo interior foi feito de tal forma que ela conseguisse ser preenchida de terra. O uso de hastes possibilitou que todas as laterais e tampa superior da estufa fossem vazadas para que fosse possível cobri-las com plástico. Além do plástico, as laterais e tampa da estufa também foram cobertas com tela sombrite, que auxilia na redução da incidência de sol diretamente na planta. Para evitar a deterioração da base de madeira pelo contato direto com a terra úmida, foi colocada uma peça de alumínio que cobrisse toda a base e servisse de receptáculo para a terra. No fundo dessa peça de alumínio foram postas pequenas pedras para auxiliar no escoamento de água e feitos furos pelos quais a água pudesse escorrer para a bandeja retirável que fica embaixo da estrutura, que foi pensada para que a água não caísse da estufa direto no chão, facilitando a alocação da estufa em diferentes ambientes. Na figura 4.2b é possível ver o produto final já em operação e no anexo A, é possível ver os desenhos do projeto com as medidas.



(a) Estufa com a estrutura finalizada



(b) Estufa em operação

Figura 4.2: Estrutura da Estufa

Para a alocar a eletrônica responsável pela aquisição, foi feito uma outra estrutura de madeira que fica anexada à estufa, chamada de Módulo de Controle na figura 4.1. Como pode ser visto na figura 4.3a esse receptáculo se assemelha a um armário, com 3 prateleiras e um grande compartimento para alocar o recipiente que vai armazenar água, feito de garrafa pet. A ideia é que a estufa com esse módulo anexado consiga ter autonomia hídrica e energética, tendo só a necessidade do responsável encher o recipiente e carregar a bateria quando necessário. Na figura 4.3b é possível ver o armário já com a eletrônica alocada e, também, no anexo A é possível ver o projeto com as medidas.



(a) Armário finalizado



(b) Armário já operante

Figura 4.3: Estrutura do Módulo de Controle

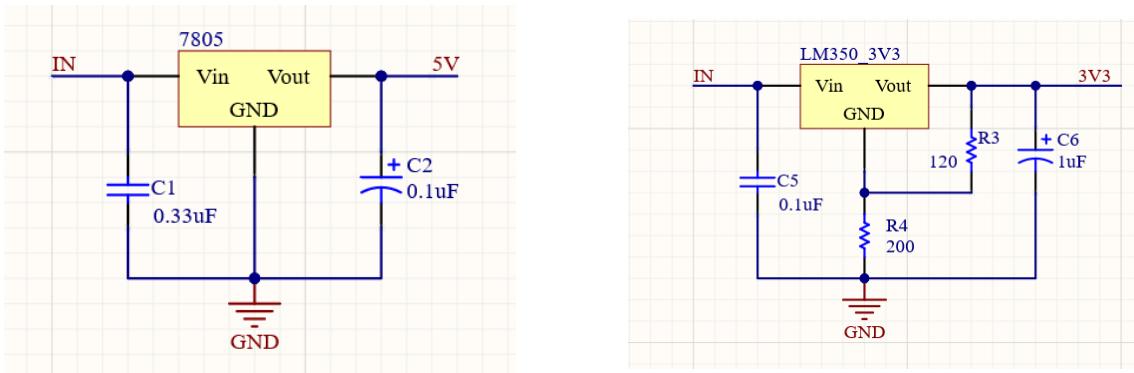
4.2.1.2 Arquitetura Eletrônica

Todas as placas de circuito foram projetadas utilizando o software Altium Designer e mandadas para serem impressas. Para reduzir a complexidade do projeto eletrônico dessa parte do projeto, ele foi dividido em duas placas diferentes. Sendo uma delas responsável pela alimentação de todo circuito (Placa de Potência) e uma das responsáveis pelo recebimento de sinais dos sensores e envio de sinais para os atuadores (Placa Lógica).

A Placa de Potência se fez necessária visto que cada componente do sistema precisava de um tipo de alimentação diferente. Ela foi projetada para receber a alimentação da bateria, regular a tensão e alimentar cada componente da forma necessária. Para conseguir essas regulações, foram utilizados dois tipos diferentes de regulador de tensão, o LM350 para regulações de 3.3V e 12V e um LM7805 para regulação de 5V, os três anexados aos seus circuitos auxiliares recomendados. No caso dos reguladores LM350, foi necessário utilizar resistores de valores específicos, visto que são reguladores de tensão variável que seguem a fórmula:

$$V_{OUT} = 1.25V + \left(1 + \frac{R1}{R2}\right) I_{ADJ} R2$$

sendo V_{OUT} a tensão de saída do regulador; I_{ADJ} a corrente de saída do pino de ajuste e R1 e R2 os resistores. As figuras 4.4a e 4.4b mostram, respectivamente, os circuitos auxiliares utilizados para os reguladores LM7805 e LM350 (que são recomendações tiradas do datasheet de cada um dos componentes).



(a) Circuito auxiliar utilizado para o LM7805

(b) Circuito utilizado para fazer a regulação ajustável de um dos LM350

Figura 4.4: Circuitos utilizados para a regulação das tensões

As alimentações para todos os componentes saem da Placa de Potência. Para a Placa Lógica, são usadas as alimentações de 5V e 3.3V, para a Mini Bomba de água, é utilizada a alimentação de 12V e para os motores é utilizada a tensão da bateria, sem nenhuma regulação. A placa foi alocada na base do Módulo de Controle, de forma que ficasse ao lado da bateria e o cabeamento saísse todo da base e subisse para se conectar aos outros componentes. No anexo B é possível ver o esquemático completo, PCB e modelo 3D da placa.

Já a Placa Lógica foi projetada para ser um shield para o ESP32, isto é, o ESP32 se conecta a essa placa e os dados são enviados/recebidos por ele. Esta placa possui os conectores de entrada para todos os sensores e conectores de saída para todos os atuadores.

As alimentações para cada sensor variam dependendo das necessidades deles, assim como a dos atuadores. As alimentações do Relé e do sensor de umidade do solo

são 5V. Os outros sensores e o sinal lógico do motor funcionam a 3.3V que também é a tensão utilizada para alimentar o ESP32. Essa placa foi alocada na prateleira do meio do Módulo de Controle, de forma que ela conseguisse se comunicar não só com todos os compartimentos com facilidade, mas também com a estufa, visto que essa placa é o componente principal do Módulo de Controle já que todos os componentes se ligam a ela.

O esquemático, PCB e modelo 3D da Placa Lógica podem ser vistos no anexo C. Na figura 4.5 é possível ver as duas placas Lógica e de Potência já soldadas.

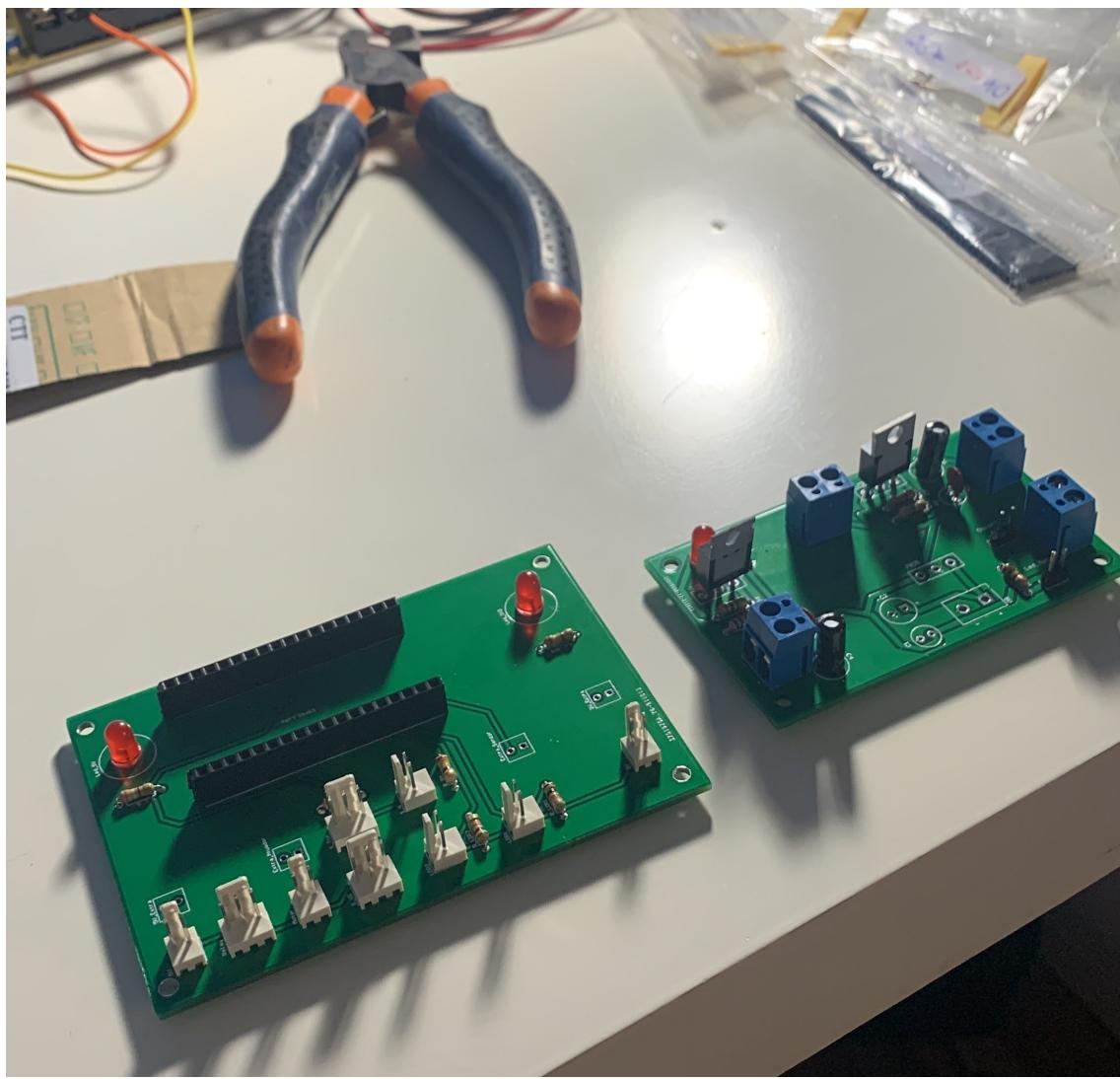


Figura 4.5: Placas Lógica (a esquerda) e de Potência (a direita)

4.2.1.3 Lógica de funcionamento

O responsável pelo funcionamento da lógica do projeto tanto no bloco de Monitoramento e Controle, quanto no de Comunicação e Postagem é o ESP32. O ESP32 é uma placa de desenvolvimento bem parecida com as placas Arduíno, isso é, é uma placa que já possui diversos chips integrados de forma que o usuário precise somente programar diretamente nela as funções que ele quer. O ESP32 da Heltec utilizado no projeto é uma versão que já possui o chip responsável pela comunicação LoRa.

A lógica pensada para o bloco de Monitoramento e Controle foi acionar o ESP32 a cada 3 horas, o deixando ligado por 1 minuto entre esses blocos de tempo. Nesse 1 minuto no qual o ESP32 está ligado, ele envia os dados dos sensores para o receptor, e também verifica os sensores para acionar os atuadores se necessário.

Nesse um minuto de funcionamento, todos os sensores são lidos e seus dados são enviados via LoRa para o receptor. O sensor de nível fica alocado no receptáculo que armazena a água e envia um sinal digital simples para o ESP32, avisando se o receptáculo está com nível baixo ou não. O sensor de umidade do solo envia um sinal analógico que varia de 2500 a 4095, sendo 2500 o mais úmido e 4095 o mais seco (determinado em testes empíricos). Para colocar o dado do sensor de uma forma mais amigável, foi feito um mapeamento de 0 a 100 no código, sendo 0 o menos úmido e 100 o mais úmido. E, por fim, o sensor de umidade e temperatura do ar possui uma biblioteca específica do próprio sensor, que já envia os dados da temperatura em graus Celsius e da umidade em porcentagem.

O controle é feito em duas frentes e nas duas foi escolhido um controle ON/OFF, uma vez que a natureza do sistema é tal que um controle mais complexo necessitaria de uma quantidade grande de dados para ser bem sintonizado e fazer diferença na aplicação. Uma das frentes é abrir ou fechar a janela superior da estufa dependendo da umidade do ar dentro dela. A ideia é conseguir diminuir a umidade do ar quando ela sobe demais, visto que naturalmente a evapotranspiração da planta dentro de um ambiente fechado tende a deixar o ambiente mais úmido. O set-point escolhido foi uma umidade de 95%, por que a umidade do ar na cidade do Rio de Janeiro, que

é onde o projeto foi desenvolvido, é alta na maior parte do tempo, então valores mais baixos não seriam facilmente alcançáveis. A outra frente de controle é a irrigação da plantação. O sensor de umidade do solo é lido e, se a umidade estiver abaixo de uma referência predeterminada, a mini bomba de água é acionada, irrigando o solo. O valor escolhido para ser a referência foi 65, ou seja, quando a umidade do solo estiver abaixo de 65, a irrigação é ativada. Como foi mencionado anteriormente, o valor desse sensor foi mapeado de 0 a 100. A determinação desse parâmetro foi feita a partir de testes com a estufa e vai ser explicada com mais detalhes na seção de resultados. A figura 4.6 mostra um diagrama de blocos do funcionamento desse módulo do projeto.

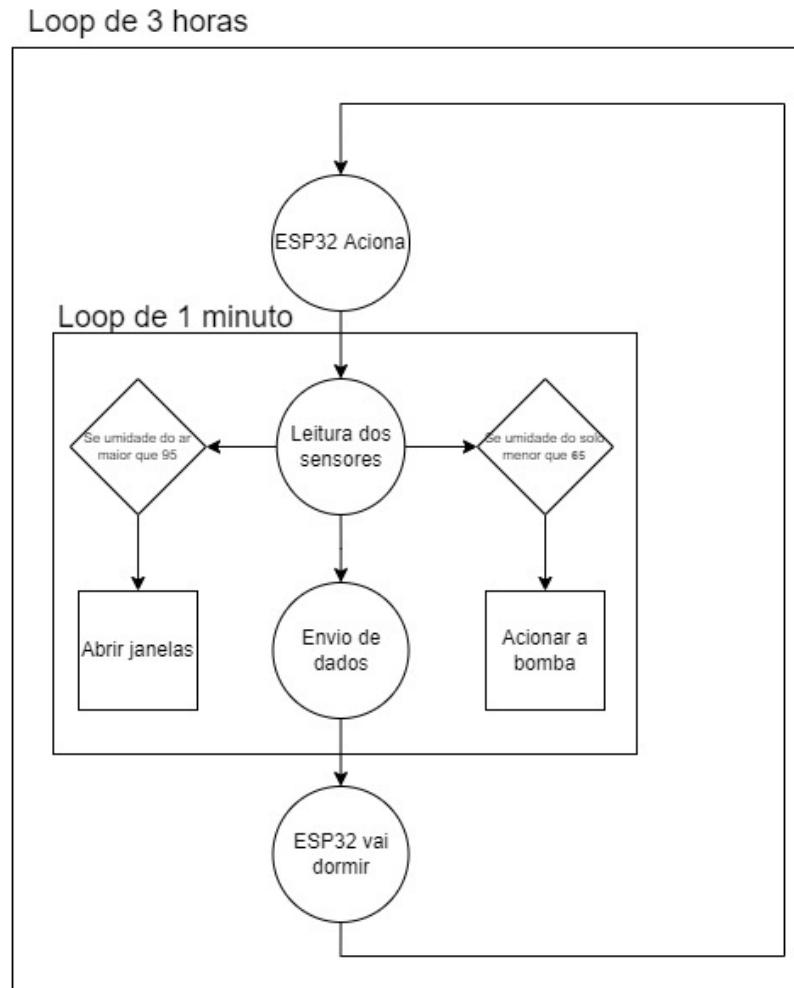


Figura 4.6: Diagrama de blocos da lógica

Todo o código feito para o projeto foi desenvolvido utilizando a plataforma Arduíno e pode ser encontrado no link: <https://github.com/Yannmsilva/>

Códigos-TCC. O código específico para o bloco de Monitoramento e Controle pode ser visto, também, no anexo D.

4.2.2 Comunicação e Postagem

Nesta seção, serão apresentados detalhes sobre o desenvolvimento do bloco responsável pela recepção dos dados e postagem na web, que contém o Receptor.

4.2.2.1 Estrutura Física

Para esse bloco, apenas uma estrutura mínima foi necessária para o Receptor, visto que é composto apenas por: uma bateria LiPo de 4S (ou 14,8V) e 1300mAh e um ESP32. Dessa forma, foi utilizado um recipiente de plástico genérico e feito alguns furos nele para conseguir colocar um botão, dois LEDs e uma antena. Na figura 4.7 é possível ver como ficou o Receptor.



(a) Receptor em funcionamento



(b) Parte interna do receptor

Figura 4.7: Receptor

4.2.2.2 Arquitetura Eletrônica

Como mencionado anteriormente, o único componente que o Receptor tem é o ESP32. No entanto, para conseguir utilizar o ESP32 com uma bateria de carga maior como a 4S utilizada no projeto, era necessário fazer uma regulação de tensão, já que o ESP32 funciona a 3.3V. Então, uma placa foi projetada especificamente para o Receptor.

Essa placa, assim como a Placa Lógica, é uma placa shield que serve como receptáculo para o ESP32. Nela, foi utilizado um outro LM350, para regular a tensão direta da bateria para 3,3V. Além disso, também foram colocados alguns LEDs na placa, para conseguir fazer monitoramentos, como o recebimento de dados e o estado do sistema. A figura 4.8 mostra as 3 placas completas, sendo a mais à esquerda a placa do Receptor, a do meio a Placa de Potência e a mais à direita a Placa Lógica. No anexo E é possível ver o esquemático completo, PCB e modelo 3D da placa do Receptor.



Figura 4.8: Placas utilizadas no projeto

4.2.2.3 Lógica de Funcionamento

A lógica de funcionamento do Receptor teve que ser moldada a partir da lógica do bloco de Monitoramento e Controle. Para conseguir que o Receptor conseguisse recolher os dados vindos do Módulo de Controle, o Receptor, ativa a cada 3 horas e passa 1 minuto ativo. Um diagrama de blocos do funcionamento desse bloco pode ser visto na figura 4.9.

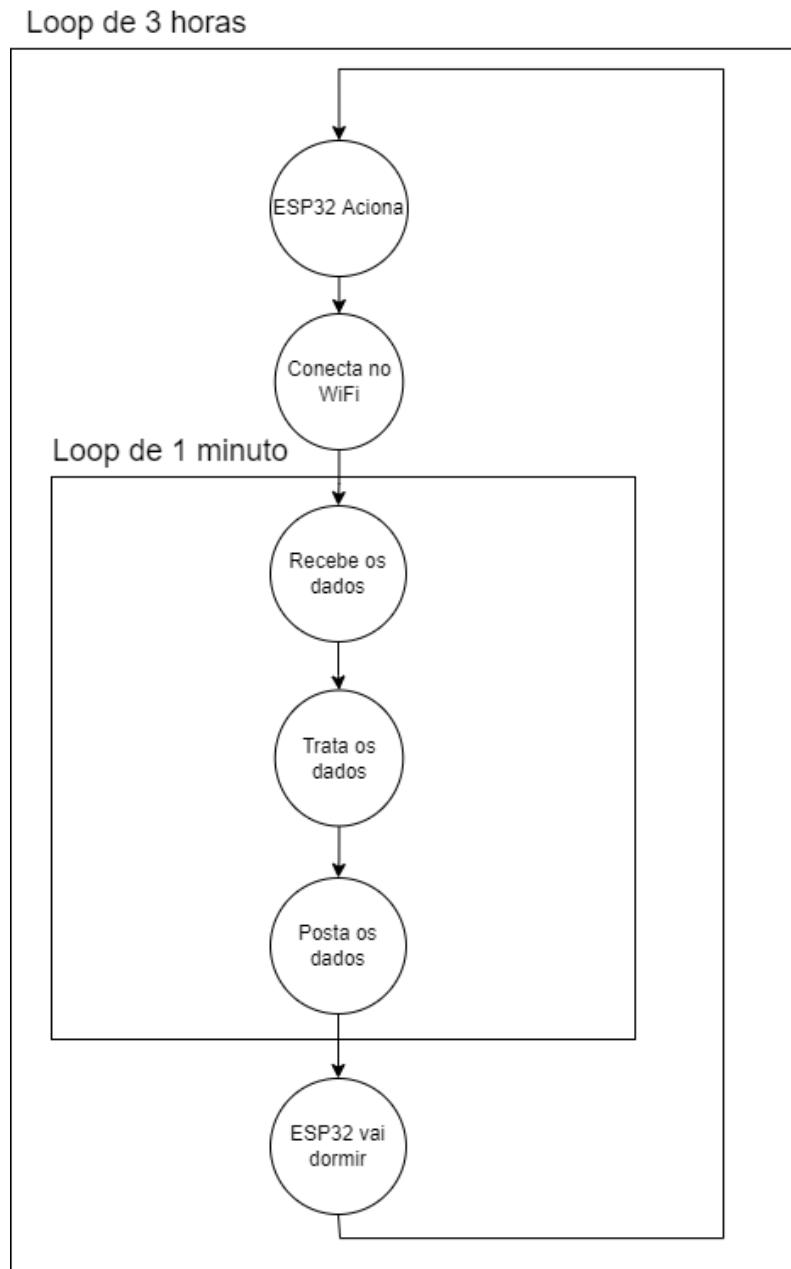


Figura 4.9: Diagrama de funcionamento do bloco de Comunicação e Postagem

Nesse 1 minuto ativo, o Receptor se conecta ao WiFi determinado no código,

recebe os dados via LoRa e os posta no site ThingSpeak. Para conseguir que os dados fossem postados de forma correta, foi necessário fazer um tratamento, uma vez que os dados chegam todos como tipo char e precisam ser postados como int. Dessa forma, os dados chegam, são tratados e seguem para a postagem.

A postagem é feita utilizando uma biblioteca pronta da plataforma ThingSpeak, que é uma plataforma aberta da MathWorks para projetos relacionados a IoT, na qual você consegue visualizar, analisar e armazenar dados.

4.3 Resultados

Nessa seção serão discutidos os resultados observados nos testes com o protótipo da ELI. Os testes tiveram duas etapas diferentes, sendo uma delas uma etapa preliminar, na qual se buscou verificar o funcionamento do sistema em um curto período de tempo e chegar a uma estimativa da autonomia energética do projeto e a outra uma etapa mais duradoura, na qual se buscou validar a estimativa feita no teste preliminar, testar a autonomia hídrica do projeto e verificar se a ELI conseguiria manter o cuidado de uma planta sem a ajuda humana, cumprindo com seu objetivo proposto.

Inicialmente, para verificar a autonomia energética do projeto, foi feita a medição da carga das duas baterias (Receptor e Módulo de Controle) ao longo do tempo enquanto o sistema estava ativo, para uma tomada de dados. Os resultados encontrados nesse teste podem ser encontrados na tabela 4.1 e no gráfico na figura 4.10, na qual a Bateria 1 representa a bateria do Módulo de Controle e a Bateria 2 a bateria do Receptor.

Hora	Carga Bat 1(V)	Carga Bat 2(V)
17:16	16.7	16.6
18:49	16.2	16.6
20:31	15.8	16.5
22:05	15.5	16.4
06:37	13.8	16.2

Tabela 4.1: Carga das baterias 1 (Módulo de Controle) e 2 (Receptor) ao longo do tempo

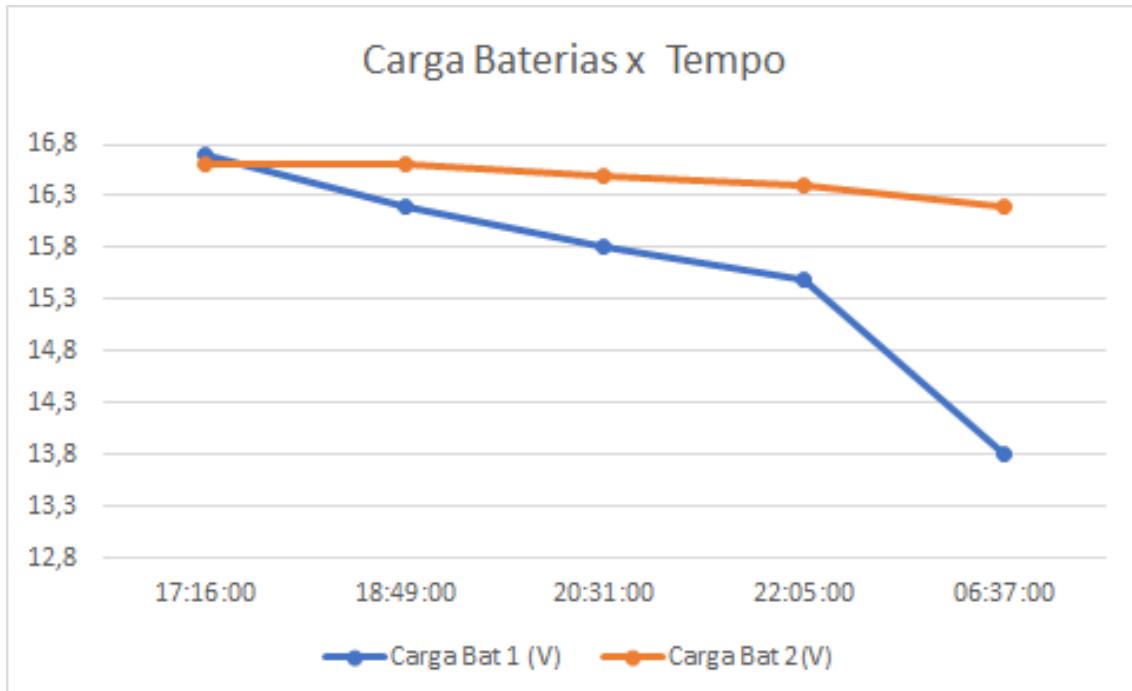


Figura 4.10: Gráfico das cargas da bateria pelo tempo

Para conseguir entender bem o que esse resultado significa, será explicado brevemente a estrutura de uma bateria LiPo. As baterias LiPo são formadas por 1 ou mais células de Lítio-Polímero ligadas em série e isso vem explícito no nome, uma bateria 1s tem apenas uma célula, 2s duas células e assim se segue. Cada célula tem tensão nominal de 3.7V, mas quando totalmente carregadas podem chegar a 4.2V e no processo de descarga não podem descer a menos de 3.2V, por risco de danificar a bateria. Outra característica importante da bateria, é a sua capacidade.

A capacidade é o quanto de corrente a bateria tem que fornecer em uma hora para descarregar totalmente e, no caso das baterias utilizadas nesse teste, esse valor é de 1300mAh ou 1.3Ah.

Dito isso, temos que o intervalo de tensões que as duas baterias 4s do projeto tem para funcionar de forma saudável seria de 16.8V até 12.8V, totalizando 4V. Como pode se observar, as verificações foram feitas em intervalos de aproximadamente 1 hora e meia, com exceção do último ponto, e através desses dados pôde-se estimar os seguintes tempos de funcionamento:

Módulo de Controle	15 horas
Receptor	6 dias

Tais resultados nos permite assumir, de forma indireta e aproximada, que o consumo energético do Módulo de Controle foi de 85mAh e do Receptor de 8,5mAh.

Com os resultados do teste de autonomia energética em mãos, foi tomada a decisão de utilizar uma bateria de carga um pouco maior para o segundo teste que seria mais duradouro, para evitar a necessidade de carregamento constante da bateria. Dessa forma, para o segundo e mais longo teste, se utilizou no Módulo de Controle uma bateria também LiPo 4s, mas de 3000mAh de carga. Usando a estimativa proposta como base, uma bateria com essa carga deveria funcionar por aproximadamente 35h sem precisar carregá-la, estimativa essa que foi confirmada nos testes que serão detalhados a seguir.

O segundo e principal teste com o protótipo consistia em deixar o sistema ligado tempo o suficiente para se verificar qualquer possível alteração na planta presente na estufa. A planta utilizada como objeto teste foi um pé de salsa e a ELI foi deixada funcionando por aproximadamente 9 dias consecutivos. Nesse período de tempo, foi necessário carregar a bateria do Módulo de controle 6 vezes e a do Receptor 1 vez, o que bate bem com a estimativa energética estipulada no teste preliminar.

Os resultados desse teste são mais facilmente visualizáveis através dos dados gerados. Como explicado anteriormente, os dados eram enviados em períodos fixos de 3 em 3 horas, então uma quantidade razoável de pontos foi gerada para cada um dos

parâmetros monitorados. A anexo G contém todos os dados crus gerados no experimento. Devido a quantidade de pontos seria complicado fazer uma análise usando a tabela como base, então para tornar a interpretação mais amigável e eficiente, serão utilizados os gráficos gerados dos dados expostos no anexo G.

Primeiramente, serão apresentados os parâmetros de temperatura e umidade do ar. Na figura 4.11 é possível verificar a variação de temperatura ao longo do tempo. O gráfico apresentado na figura 4.11a contém os dados crus tirados direto dos sensores e o gráfico na figura 4.11b os dados depois de uma análise feita para excluir os pontos outliers.

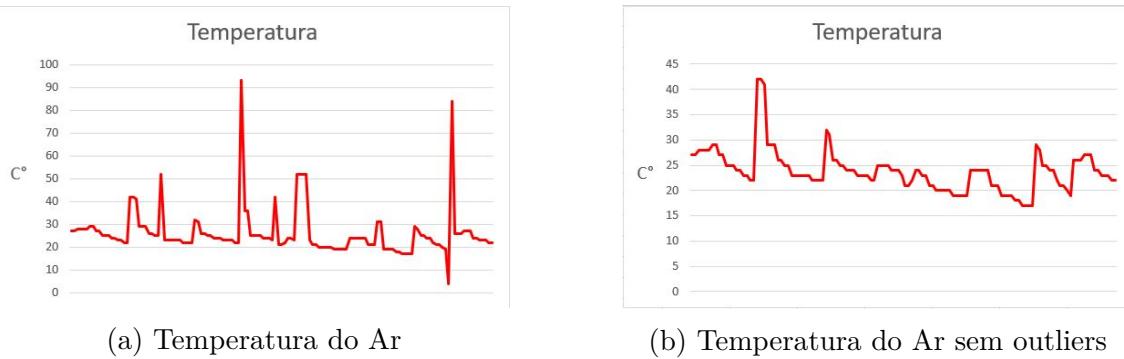
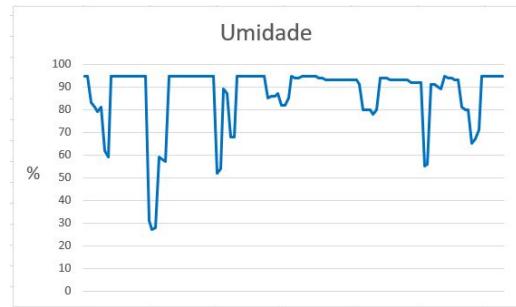


Figura 4.11: Dados da Temperatura do Ar

Essa análise para exclusão dos outliers foi feita por que foi observado que toda vez que o sistema era religado após o carregamento da bateria, os sensores davam um pico, gerando dados que não faziam sentido para a aplicação, como temperaturas acima de 90°C ou abaixo de 5°C. Possivelmente isso acontecia por causa do pico de tensão gerado no momento que se ligava o sistema. Como esse processo acabou acontecendo mais de uma vez dada a limitação da bateria que precisava ser carregada constantemente, fez sentido retirar esses dados fora do esperado para entender o comportamento do sistema de forma mais razoável. Para os dados de umidade do ar o mesmo procedimento foi feito, como pode ser observado na figura 4.12.



(a) Umidade do Ar



(b) Umidade do Ar sem outliers

Figura 4.12: Dados da Umidade do Ar

Analizando agora com os dois gráficos cruzados como pode ser visto na figura 4.13 é possível perceber uma tendência clara da umidade abaixar conforme a temperatura aumenta. Os períodos em que isso acontecia, por mais que os dados não consigam demonstrar isso, eram os períodos nos quais o sol incidia na estufa. Normalmente, o sol incidia na estufa sempre em um mesmo momento ao longo dos dias, fazendo com que a temperatura nesse período subisse e a umidade descesse. Esses momentos representam os picos da temperatura e os vales na umidade. É possível perceber também uma diminuição da temperatura média ao longo da semana dos testes, o que faz sentido se comparado com os dados de temperatura entre os dias 03/05/22 e 12/05/22 na cidade do Rio de Janeiro, como aponta o site <https://www.accuweather.com/pt/br/rio-de-janeiro/45449/may-weather/45449>.

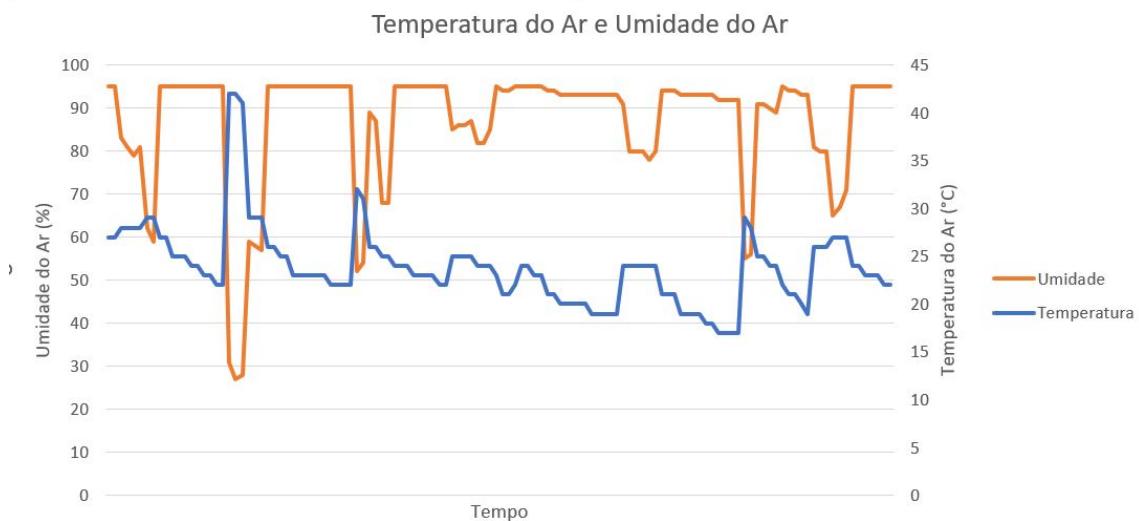


Figura 4.13: Dados de umidade do ar e temperatura do ar cruzados

Analizando agora apenas a umidade do ar, é possível perceber através dos gráficos 4.12 e 4.13 que o parâmetro escolhido inicialmente de 95% como parâmetro de controle para o fechamento/abertura da tampa da estufa não foi bom. Essa conclusão foi tirada por que, excluindo os momentos nos quais o sol incidia na estufa (que são marcados pelos vales) a tendência eram os valores estarem sempre 95% ou acima, dessa forma, a estufa esteve praticamente sempre aberta ao longo do teste, salvo os momentos em que aconteciam os vales, nos quais ela forçava o fechamento e logo depois abria. Depois de alguns dias de teste, para não danificar a estrutura, foi optado por deixar a estufa aberta sem o controle de abertura da tampa, uma vez que ele não se provou muito útil dada as circunstâncias do teste.

Para o caso dos dados da umidade do solo, também foi feito uma análise dos outliers, mas excepcionalmente para esse parâmetro os outliers foram importantes para entender o que acontecia ao longo do teste. A figura 4.14 mostra os dados obtidos.

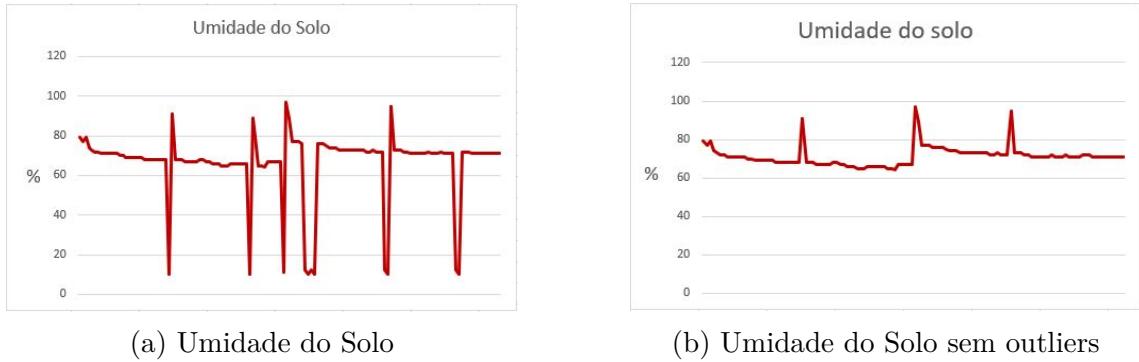


Figura 4.14: Dados da Umidade do Solo

A figura 4.14a mostra os dados sem estarem tratados e neles é possível observar os mesmos picos e vales que se observava nos dois outros parâmetros já analisados. No entanto, como esse parâmetro controlava a irrigação, no momento em que a ELI era ligada e havia um vale por causa do pico de tensão nos sensores, a umidade do solo ficava abaixo do setpoint determinado para a irrigação ser acionada, fazendo com que a bomba acionasse e o próximo dado coletado era sempre uma umidade bem maior, por que a bomba estava ligada no exato momento de aquisição. Esse comportamento pode ser observado em quase todos os vales. Olhando apenas para os dados tratados

vistos na figura 4.14b, é difícil de tirar essa conclusão, uma vez que eles excluíam os vales provenientes dos picos de tensão. Outra relação importante de se observar é quantas irrigações foram necessárias para o recipiente chegar em um nível baixo. A figura 4.15 mostra o gráfico de umidade do solo comparado com o gráfico de nível baixo no recipiente que contém água, é possível observar que em dado momento, o recipiente indica a necessidade de ser enchido, e se compararmos com os dados do sensor de umidade do solo, que contém dois vales seguidos de picos (o que indica uma irrigação do sistema), conseguimos concluir que são necessárias aproximadamente 2 irrigações para o sistema precisar de que o recipiente seja enchido.



Figura 4.15: Dados de umidade do solo e nível do recipiente cruzados

O controle de umidade do solo foi pautado na ideia de que a planta precisaria ser irrigada uma vez por dia. Não necessariamente esse é o certo para todo tipo de planta, mas era o ideal para espécie sendo observada no teste. Dessa forma, foi tentado ajustar o parâmetro certo para que a irrigação ocorresse uma vez ao dia, ao longo do teste. Inicialmente, com os dados do primeiro dia, o parâmetro foi setado para 65. No entanto, conforme o teste seguiu, foi percebido que o sensor de umidade do solo variava muito pouco, muitas vezes não chegando a descer de 65. Por isso, o valor foi ajustado para 70 e assim ficou até o fim do teste.

É importante evidenciar que todas as análises envolvem fatores externos que não

puderam ser controlados, como incidência do sol, clima do local onde a estufa estava, a espécie da planta utilizado como objeto de estudo, entre outros. Para o caso desse teste, por exemplo, a baixa variação do sensor de umidade do solo claramente se dava pela baixa incidência de sol na estufa, uma vez que o sol ajudaria a fazer com que o solo ficasse mais seco, assim como a umidade do ar. No entanto, como pode se verificar pelas fotos da figura, o sistema foi capaz de manter a salsa ao longo dos 9 dias de teste.



(a) Primeiro dia do experimento
03/05/22



(b) Dia seguinte ao último dia de experimento 13/05/22

Figura 4.16: Resultado do experimento

É possível observar uma breve diferença da figura 4.16a para a figura 4.16b se olharmos para o canto mais embaixo da figura, região na qual a densidade de salsa claramente diminuiu. Através da observação do experimento, foi possível perceber que isso se deu por que esse canto era o que menos havia incidência do sol, visto que ele incidia sobre a estufa exatamente pelo outro lado. Como esse era um dos parâmetros os quais não se tinha controle, pode se dizer que o parâmetro no qual se buscou o controle sucedeu em seu objetivo, era manter a planta irrigada.

Através dessa análise, fica evidente que o monitoramento de alguns outros parâ-

metros poderia ter ajudado a alcançar resultados melhores. O principal deles sendo a incidência do sol, que influenciou bastante nos resultados como pôde se observar. Outra melhoria que certamente traria melhores resultados seria utilizar mais de um sensor de umidade do solo. Por mais que o sensor tenha cumprido com seu papel, dada a área que ele estava, possivelmente a utilização de dois sensores condiria melhor com a real umidade do solo de toda a estufa.

Todos os dados utilizados nessa análise de resultados foram retirados da plataforma ThingSpeak. Os dados armazenados ao longo do tempo podiam ser extraídos da plataforma em um arquivo de formato .CSV, que foi aberto no programa Microsoft Excel para conseguir fazer os gráficos utilizados na análise. Além disso, com a plataforma ThingSpeak também foi possível fazer o monitoramento em tempo real do sistema, sempre sendo possível saber se postagem estava sendo feita do jeito correto. As fotos 4.17, 4.18, 4.19 e 4.20 abaixo mostram como é a interface de monitoramento para os 4 parâmetros trabalhados, nas quais à esquerda se observa um mostrador com o último dado recebido e a direita uma série histórica dos dados. Para o caso do sensor de nível, à esquerda se vê um LED que, quando aceso, significa que o recipiente de água está vazio, indicando a necessidade de enchê-lo.

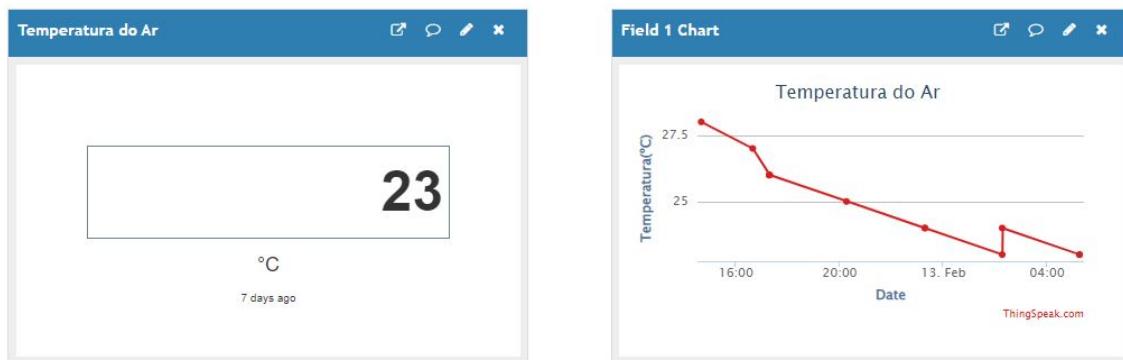


Figura 4.17: Dados da temperatura do ar ao longo do tempo

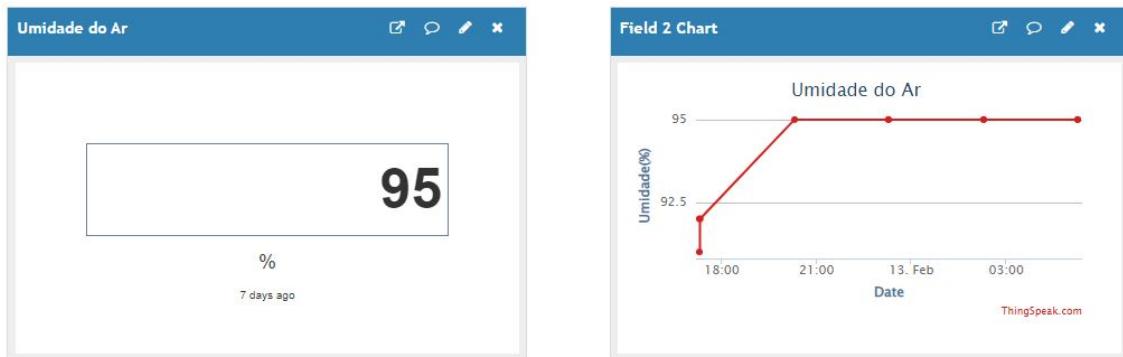


Figura 4.18: Dados da umidade do ar ao longo do tempo

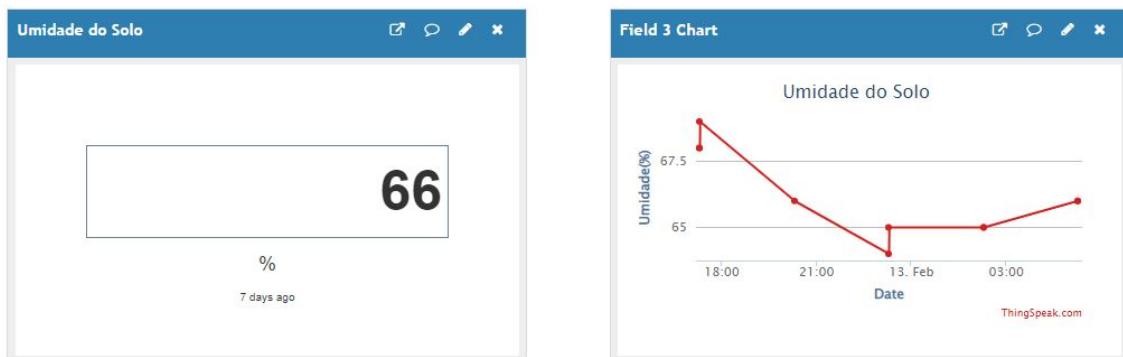


Figura 4.19: Dados da umidade do solo ao longo do tempo

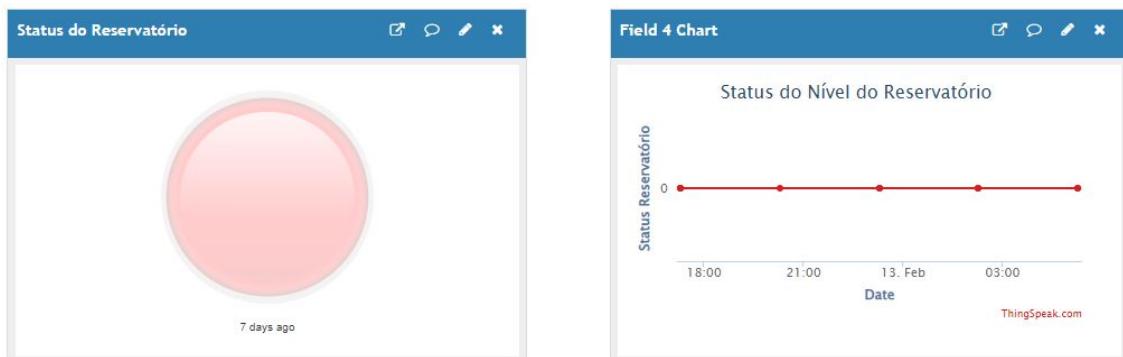


Figura 4.20: Dados do sensor de nível ao longo do tempo

4.3.1 Distância de comunicação

Por último, mas não menos importante, foi feita uma análise de alcance de comunicação do Módulo de Controle e Receptor. Esses trouxeram resultados condizentes, dado os componentes utilizados. O máximo de distância que se conseguiu fazer com que os dois ESP32 interagissem via protocolo LoRa foi aproximadamente 160m. De

acordo com a documentação do protocolo¹, em ambientes urbanos o alcance deveria ser entre 3 e 5km. No entanto, não é especificado no site o tipo de antena utilizada, nem as condições dos testes. Como é possível observar nos resultados do trabalho apresentador por [22], utilizando componentes bem parecidos aos utilizados nesse projeto, foi conseguido um alcance de aproximadamente 300m em terreno aberto, o que é de certa forma comparável aos 160m, já que o teste com a estufa não foi executado em campo aberto.

Alguns fatores podem auxiliar na melhora dos resultados encontrados. O primeiro deles é a utilização de antenas com ganho maior, que certamente resultariam em um alcance maior de comunicação. Outro fator que poderia ajudar é mudar a forma e localidade dos testes, os dois ESP32 estavam sempre no mesmo plano e em um ambiente que não era aberto, colocá-los em planos diferentes e em campo aberto provavelmente traria resultados positivamente diferentes. No entanto, é sempre importante evidenciar que a necessidade de maior alcance depende da aplicação, a utilização de antenas de maior ganho, por exemplo, reduziria a eficiência energética do projeto o que é um ponto a se levar em consideração ao tomar decisões de projeto. Trazendo exemplos da Agricultura Urbana, podemos dizer que para uma horta comunitária de bairro próxima da residência do horticultor ou plantações residenciais, o projeto cumpriria com o objetivo.

¹URL: <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>; Acesso em: 22/02/22

Capítulo 5

Conclusão

Indústria 4.0 é um dos temas mais importantes da atualidade por causa do seu impacto na vida das pessoas. A ideia de sistemas que sozinhos conseguem experienciar o mundo e tomar decisões estratégicas abrem portas para uma maior liberdade na dependência humana de geração de dados. As tecnologias que estão no cerne dessa mudança do núcleo de geração de dados dos humanos para as coisas são as tecnologias IoT.

Esse trabalho propôs a construção de uma miniestufa que, através de tecnologias IoT, conseguisse ser monitorada remotamente e controlada automaticamente. O trabalho teve como motivação principal mostrar como a Agricultura Urbana poderia se beneficiar com a aplicação de projetos simples de IoT, que aumentariam a eficiência e produtividade das plantações.

Foram utilizados no projeto dois ESP32, com os quais se construiu uma arquitetura de nó e receptor, sendo a estufa o nó, com sensores de umidade do solo e ar e temperatura do ar e o receptor o gateway, que recebia os dados vindos da estufa através do protocolo LoRa. Os dados recolhidos na estufa tinham duas funções principais: serem monitorados via uma plataforma web e serem entradas para o controle dos parâmetros localmente. O monitoramento foi feito a partir do envio via WiFi dos dados recebidos pelo Receptor para a web e o controle tinha duas frentes de ação, a irrigação que age diretamente na umidade do solo e a abertura/fechamento da janela da estufa, que age na umidade do ar dentro da estufa.

O objetivo proposto foi alcançado com sucesso e os resultados apresentados nos testes foram satisfatórios. A comunicação e postagem dos dados foi um sucesso e o controle dos parâmetros deu certo para a umidade do solo e não foi satisfatório para a umidade do ar, devido a fatores externos e o equipamento escolhido. Além disso, foi possível estabelecer uma autonomia energética satisfatória de 35 horas de funcionamento para a Estufa e de até 7 dias para o Receptor, utilizando as baterias LiPo 4s de 3000mAh no Módulo de Controle e 1300mAh no Receptor. A autonomia hídrica ficou por volta de 3 dias, dependendo da necessidade de irrigação do plantio. E o alcance de comunicação encontrado foi de até 160m, sendo plausível para algumas aplicações da Agricultura Urbana.

5.1 Trabalhos Futuros

Os testes mostraram que o projeto possui alguns pontos de melhoria que otimizariam o protótipo desenvolvido. O primeiro deles seria o aumento da autonomia energética. A mudança da bateria para uma bateria com maior carga aumentaria consideravelmente a autonomia do projeto. Um aumento para uma bateria LiPo de 14Ah de carga, por exemplo, aumentaria a autonomia para até 150h, se compararmos com os dados encontrados de 15h(para uma bateria de 1.3Ah) seria um aumento de 1000%. E essa mudança não precisaria se ater apenas a baterias LiPo, podendo ser utilizadas quaisquer bateria que encontrasse as necessidades do projeto.

Outra possível melhoria ainda dentro do aspecto energético seria o corte de alimentação de todos os sensores ligados ao ESP32, da forma como o projeto foi desenvolvido, apenas o ESP32 entra em modo de sleep, ficando os sensores ainda ligados a alimentação. Colocar esses sensores também para dormir ou cortar a alimentação deles certamente ajudaria no custo energético. Por fim, outra possibilidade seria também a utilização de uma fonte que recarregasse a bateria, como um painel fotovoltaico. Como a estufa é uma aplicação agrícola, a presença de sol é praticamente indispensável, então não seria estranho pensar em uma solução com energia solar que talvez tornasse a estufa 100% autônoma energeticamente falando.

A incidência do sol foi um fator que também influenciou bastante os resultados do experimento. Certamente o monitoramento desse parâmetro influenciaria positivamente no projeto, dando a possibilidade do agricultor saber se a planta estaria recebendo sol suficiente ou não. Uma forma simples de fazer isso seria utilizado fotoresistores(ou LDRs) para captar a iluminação no local. Outro parâmetro que ajudaria muito o usuário se fosse monitorado seria a tensão da bateria, o projeto não contou com nenhum circuito que medisse a tensão da bateria e isso resultou no trabalho de medir a tensão de tempos em tempos para garantir a funcionalidade contínua do protótipo. Caso a bateria também estivesse sendo monitorada via página web, esse trabalho seria reduzido a zero, sendo possível facilmente verificar a tensão e quase seria necessário parar o sistema para carregá-la.

No que diz respeito a situação hídrica da estufa, algumas melhorias também foram pensadas. A primeira dela, seria a utilização um recipiente mais otimizado para a água. O projeto foi desenvolvido usando uma garrafa pet, no entanto, para conseguir alocar o sensor de nível na sua base, não foi possível utilizar a totalidade do seu volume. Dessa forma, a utilização de um recipiente cuja geometria fosse tal que otimizasse o volume de água e facilitasse o anexo do sensor de nível certamente aumentaria a eficiência hídrica da estufa. Outra mudança que traria benefícios para estufa é a utilização de mais sensores de umidade do solo, como mencionado anteriormente. O projeto foi desenvolvido com apenas um, no entanto, dependendo da área de terra a qual o sistema esteja tentando cobrir, apenas um sensor pode não ser suficiente para refletir a verdadeira umidade de toda a área de plantio. Outro acréscimo interessante seria conseguir medir o volume de água sendo utilizado nas irrigações, de forma com que o sistema pudesse ser adaptado para diferentes espécies de planta que demandassem menos ou mais água.

Por fim, a última melhoria proposta para trabalhos futuros seria no alcance de comunicação. A utilização de antenas com ganhos maiores e um posicionamento mais estratégico do receptor com relação à estufa trará, com certeza, melhorias no alcance, que contribuirão para mais liberdade nas possíveis aplicações do projeto.

Referências Bibliográficas

- [1] DALENOGARE, L. S., BENITEZ, G. B., AYALA, N. F., et al. “The expected contribution of Industry 4.0 technologies for industrial performance”, *International Journal of Production Economics*, v. 204, pp. 383–394, 2018.
- [2] VAIDYA, S., AMBAD, P., BHOSLE, S. “Industry 4.0—a glimpse”, *Procedia manufacturing*, v. 20, pp. 233–238, 2018.
- [3] GUBBI, J., BUYYA, R., MARUSIC, S., et al. “Internet of Things (IoT): A vision, architectural elements, and future directions”, *Future generation computer systems*, v. 29, n. 7, pp. 1645–1660, 2013.
- [4] BANU, S. “Precision agriculture: tomorrow’s technology for today’s farmer”, *Journal of Food Processing & Technology*, v. 6, n. 8, pp. 1, 2015.
- [5] ARABY, A. A., ABD ELHAMEED, M. M., MAGDY, N. M., et al. “Smart iot monitoring system for agriculture with predictive analysis”. In: *2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pp. 1–4. IEEE, 2019.
- [6] TORNAGHI, C. “Critical geography of urban agriculture”, *Progress in Human Geography*, v. 38, n. 4, pp. 551–567, 2014.
- [7] ORSINI, F., KAHANE, R., NONO-WOMDIM, R., et al. “Urban agriculture in the developing world: a review”, *Agronomy for sustainable development*, v. 33, n. 4, pp. 695–720, 2013.
- [8] DO SOCORRO SILVA, M., FERREIRA, M. J., SILVA, G., et al. “AGRICULTURA URBANA: HORTA COMUNITÁRIA DO BAIRRO DIRCEU ARCOVERDE II EM TERESINA-PI—UM ESTUDO DE CASO”, *Agrarian Academy*, v. 3, n. 05, 2016.
- [9] ARMAR-KLEMESU, M., OTHERS. “Urban agriculture and food security, nutrition and health”, *Growing cities, growing food. Urban agriculture on the policy agenda*, pp. 99–118, 2000.

- [10] RAJKUMAR, R., LEE, I., SHA, L., et al. “Cyber-physical systems: the next computing revolution”. In: *Design automation conference*, pp. 731–736. IEEE, 2010.
- [11] COOK, D., DAS, S. K. *Smart environments: technology, protocols, and applications*, v. 43. John Wiley & Sons, 2004.
- [12] MADAKAM, S., LAKE, V., LAKE, V., et al. “Internet of Things (IoT): A literature review”, *Journal of Computer and Communications*, v. 3, n. 05, pp. 164, 2015.
- [13] ASHTON, K., OTHERS. “That internet of things thing”, *RFID journal*, v. 22, n. 7, pp. 97–114, 2009.
- [14] LANDT, J. “The history of RFID”, *IEEE potentials*, v. 24, n. 4, pp. 8–11, 2005.
- [15] WANT, R. “An introduction to RFID technology”, *IEEE pervasive computing*, v. 5, n. 1, pp. 25–33, 2006.
- [16] AKYILDIZ, I. F., SU, W., SANKARASUBRAMANIAM, Y., et al. “Wireless sensor networks: a survey”, *Computer networks*, v. 38, n. 4, pp. 393–422, 2002.
- [17] BOR, M., VIDLER, J. E., ROEDIG, U. “LoRa for the Internet of Things”, 2016.
- [18] SINHA, R. S., WEI, Y., HWANG, S.-H. “A survey on LPWA technology: LoRa and NB-IoT”, *Ict Express*, v. 3, n. 1, pp. 14–21, 2017.
- [19] CULMAN, M., PORTOCARRERO, J. M., GUERRERO, C. D., et al. “Palm-NET: An open-source wireless sensor network for oil palm plantations”. In: *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, pp. 783–788. IEEE, 2017.
- [20] MUANGPRATHUB, J., BOONNAM, N., KAJORNKASIRAT, S., et al. “IoT and agriculture data analysis for smart farm”, *Computers and electronics in agriculture*, v. 156, pp. 467–474, 2019.
- [21] PALLAVI, S., MALLAPUR, J. D., BENDIGERI, K. Y. “Remote sensing and controlling of greenhouse agriculture parameters based on IoT”. In: *2017 International Conference on Big Data, IoT and Data Science (BID)*, pp. 44–48. IEEE, 2017.

- [22] GUTIÉRREZ, S., MARTÍNEZ, I., VARONA, J., et al. “Smart mobile LoRa agriculture system based on internet of things”. In: *2019 IEEE 39th Central America and Panama Convention (CONCAPAN XXXIX)*, pp. 1–6. IEEE, 2019.

Apêndice A

Projeto das estruturas da Estufa e do Módulo de Controle

Esse anexo contém cópias dos projetos das estruturas da Estufa e do Módulo de Controle, contendo todas medidas e vistas.

Vista SUPERIOR

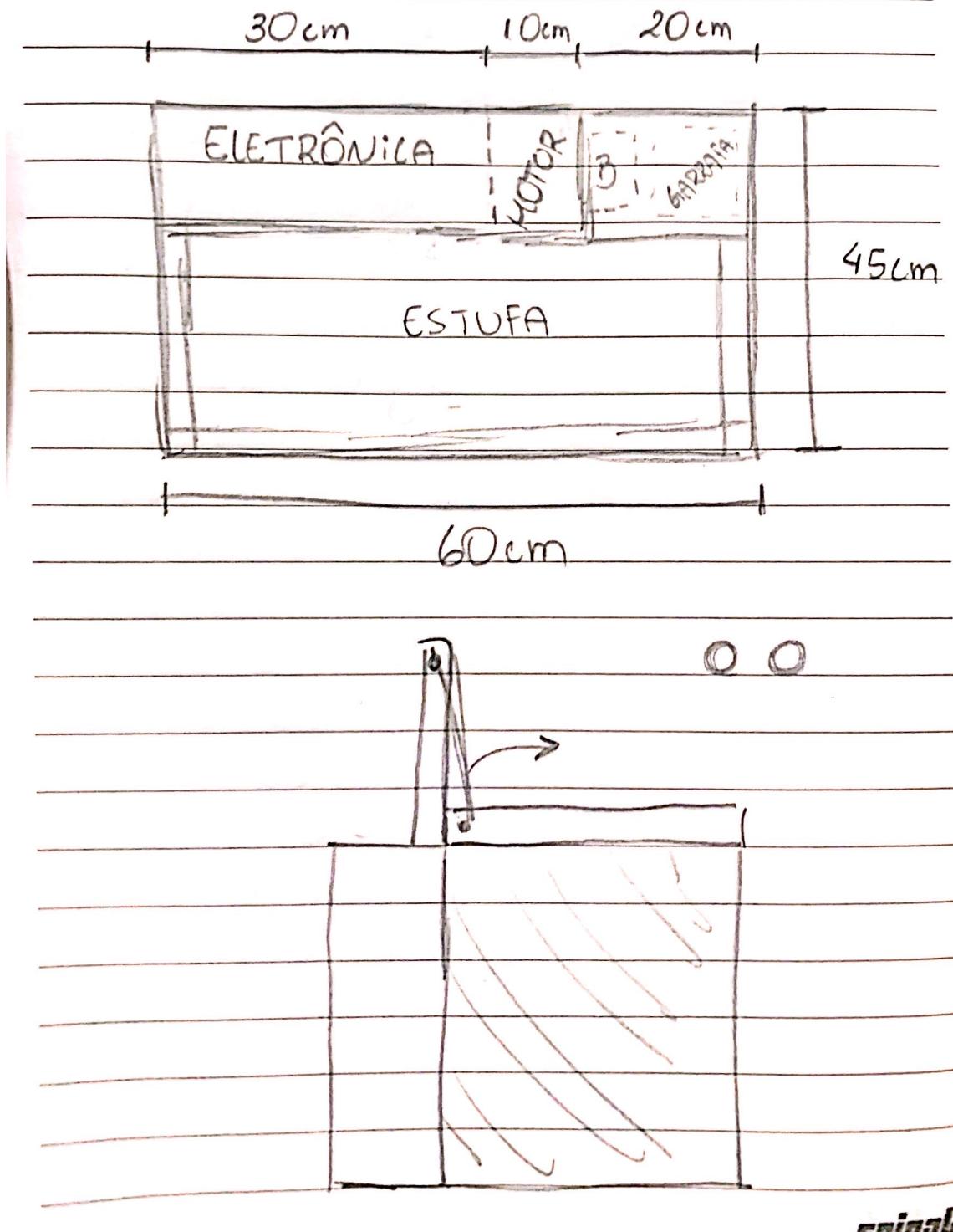


Figura A.1: Modelo da tampa e vista superior detalhada

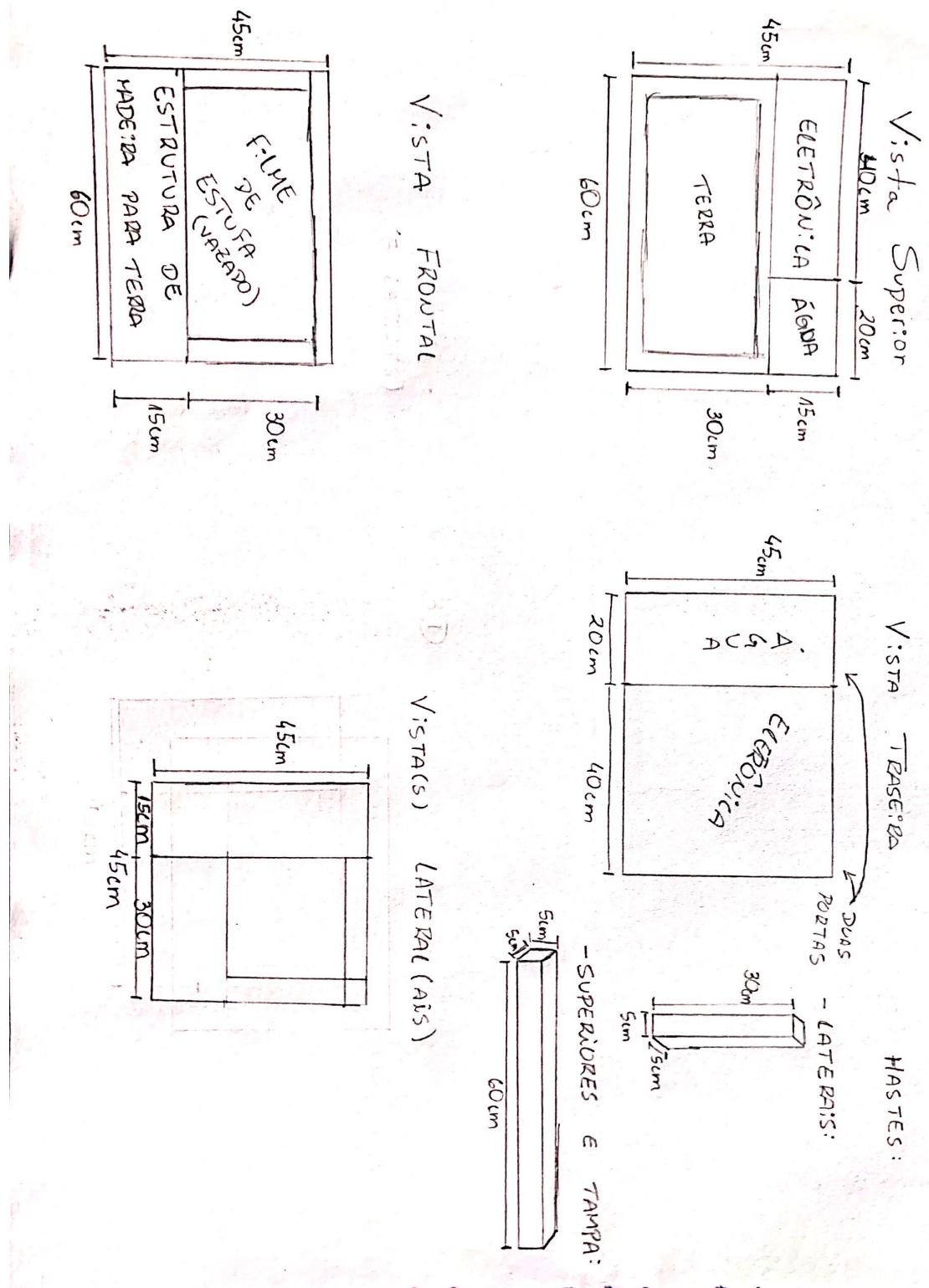


Figura A.2: Vistas 2D

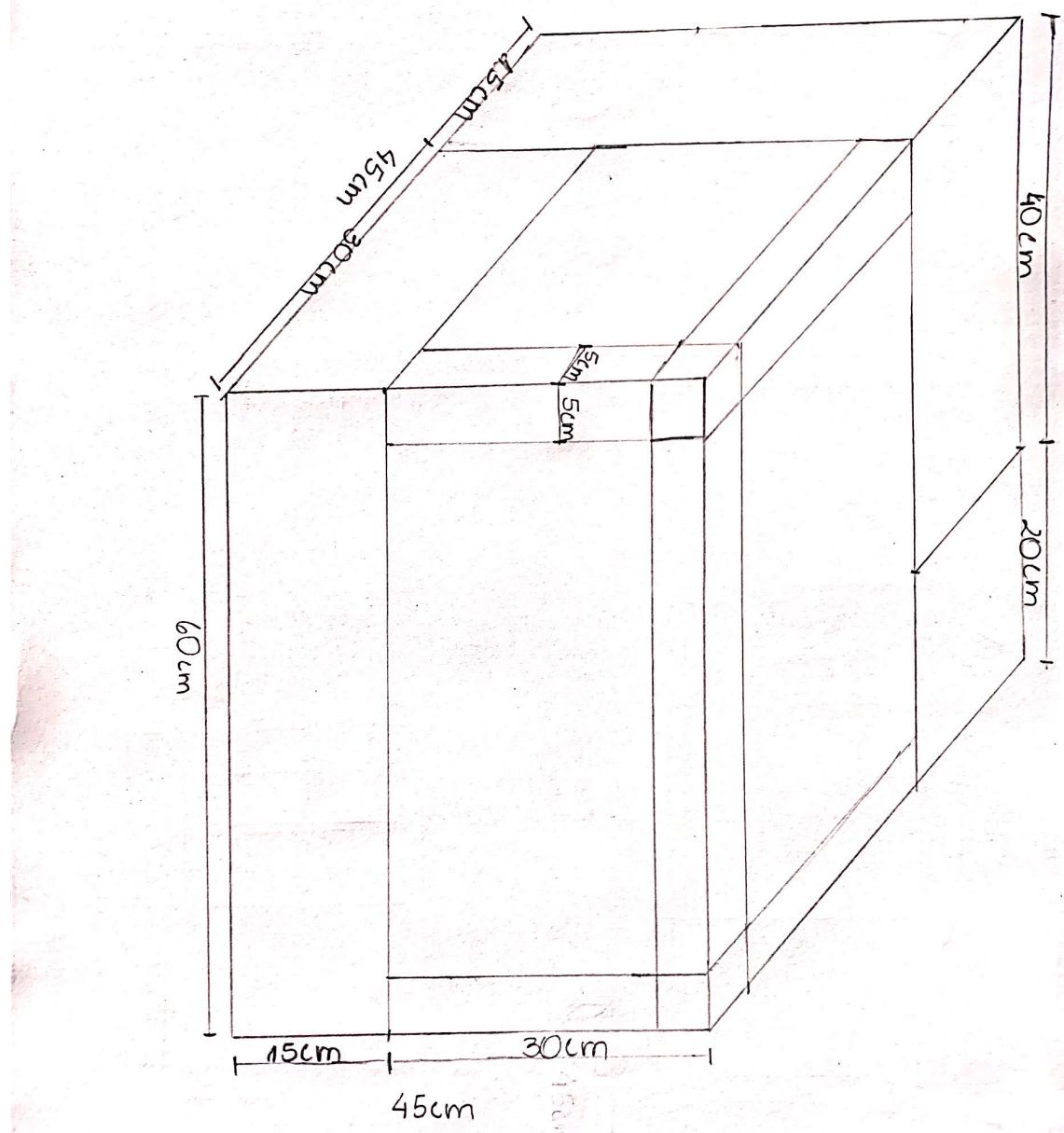


Figura A.3: Modelo 3D

Apêndice B

Projeto da Placa de Potência

Esse anexo contém o esquemático, PCB e modelo 3D da Placa de Potência.

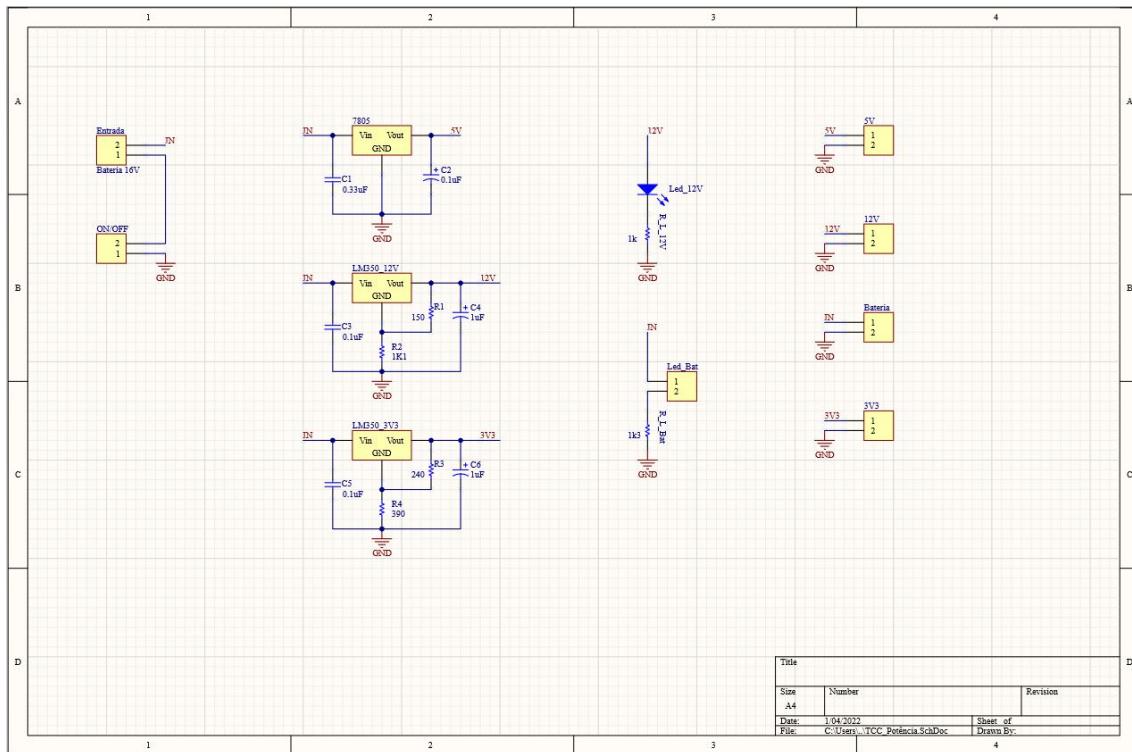


Figura B.1: Esquemático da Placa de Potência

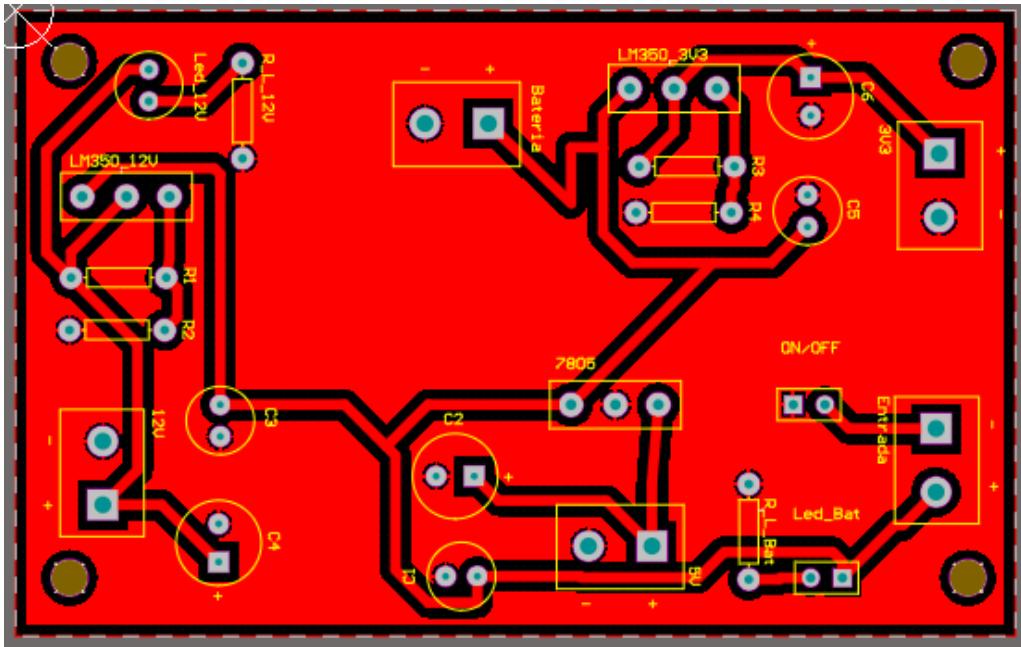


Figura B.2: PCB da Placa de Potência

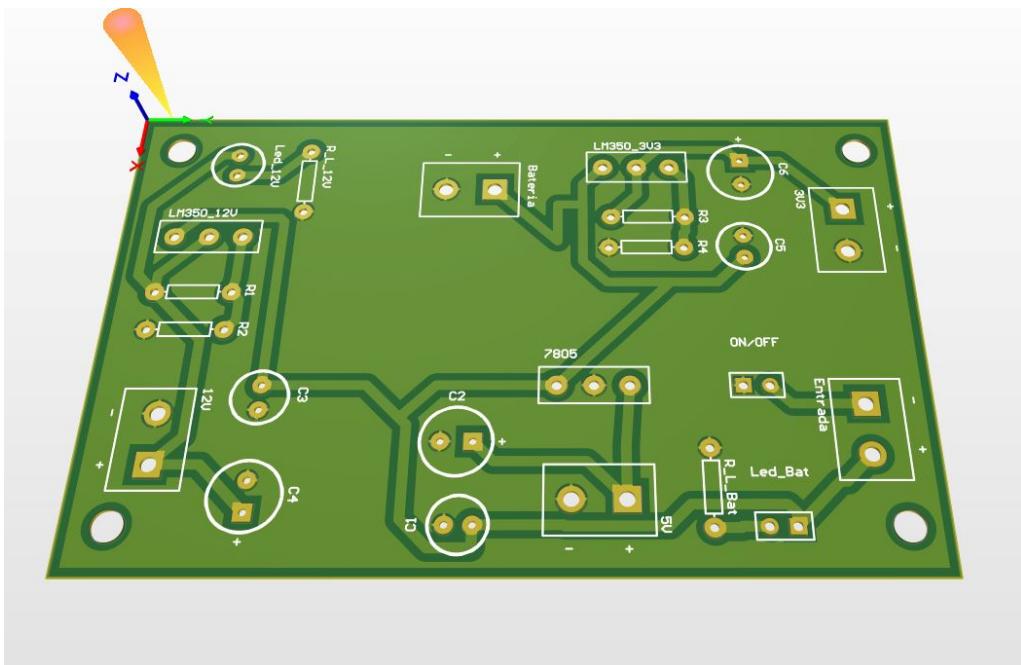


Figura B.3: Modelo 3D da Placa de Potência

Apêndice C

Projeto da Placa Lógica

Esse anexo contém o esquemático, PCB e modelo 3D da Placa Lógica.

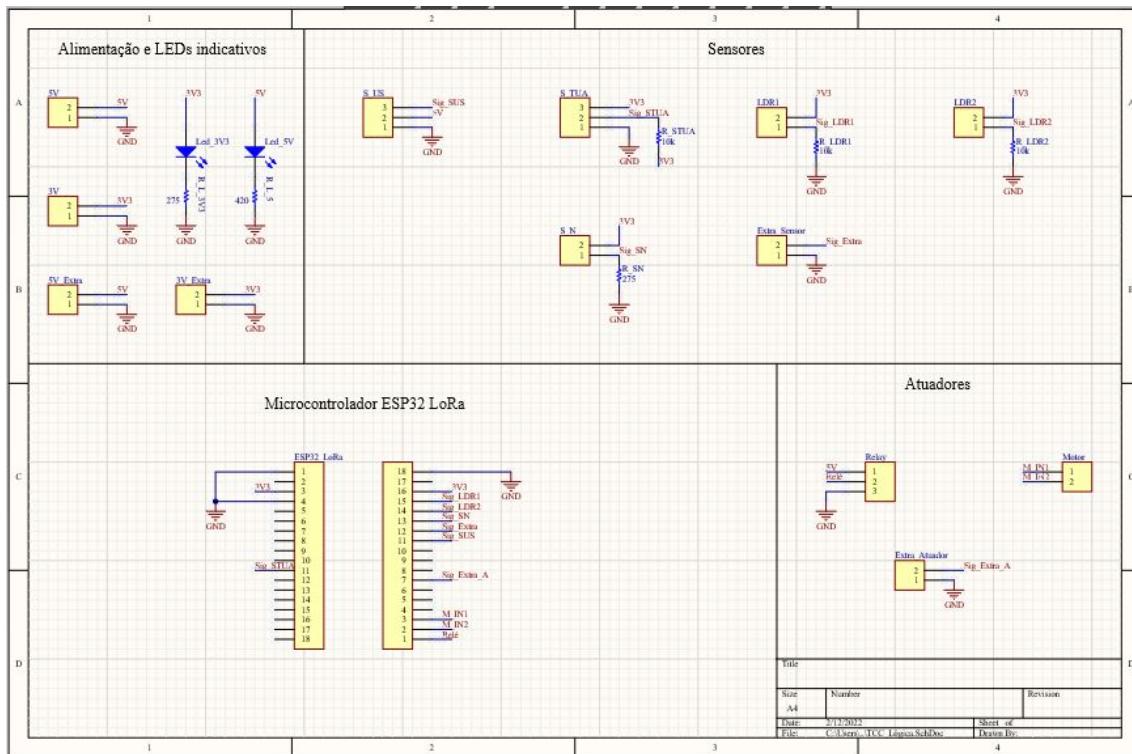


Figura C.1: Esquemático da Placa Lógica

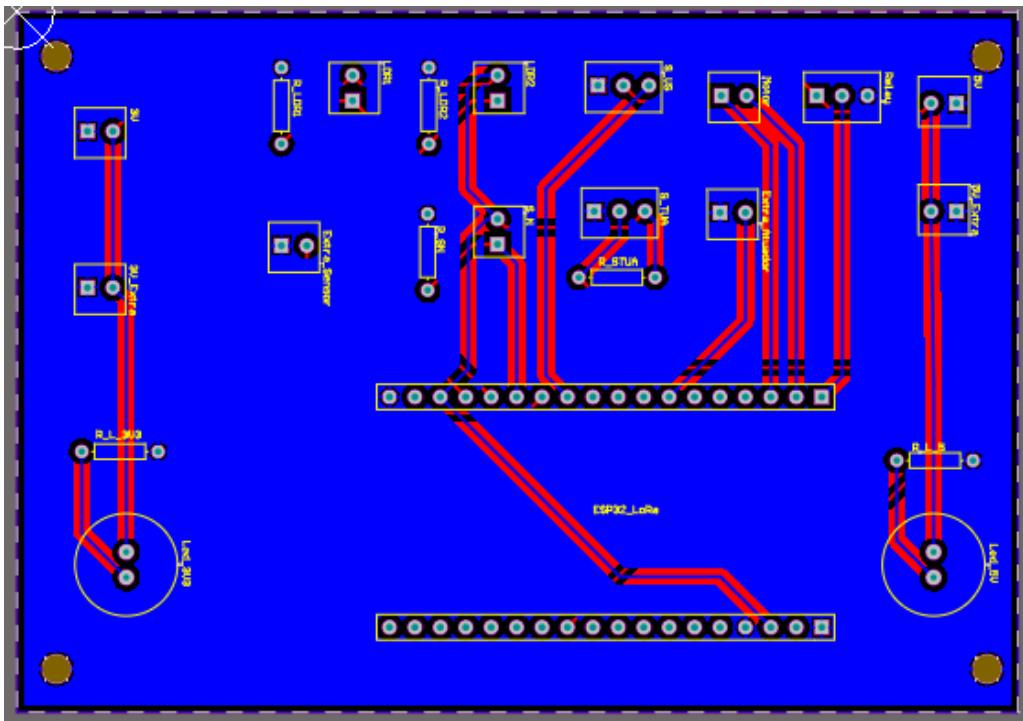


Figura C.2: PCB da Placa Lógica

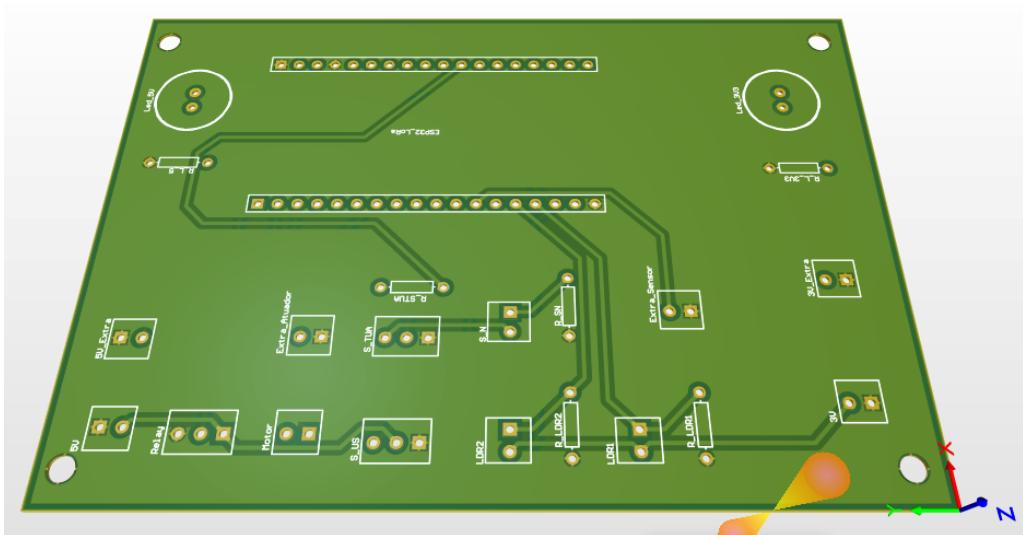


Figura C.3: Modelo 3D da Placa Lógica

Apêndice D

Código do Módulo de Controle

Esse anexo contém o código desenvolvido em Arduíno para o ESP32 alocado no Módulo de Controle.

```
#include "DHT.h"
#include "heltec.h"

#define BAND 433E6

// Sensores e Atuadores
// Configurações do sensor de Umidade e Temperatura do Ar
#define DHTPIN 0
#define DHTTYPE DHT11

int s_UmidS = 34;
int s_Nivel = 38;

int bomba = 21;
int motor_IN1 = 12;
int motor_IN2 = 13;
int motor_ENA = 25;
```

```

//Propriedades do PWM do motor
const int freq = 30000;
const int pwmChannel = 8;
const int resolution = 8;
#define dutyCycle 150

//Variáveis auxiliares
int aux = 0;
int UmidAr = 0;
int UmidSolo = 0;
int TempAr = 0;
int Nivel = 0;

//Inicialização do sensor de Temperatura e Umidade do Ar
DHT dht(DHTPIN, DHTTYPE);

//Parâmetros para setagem do timer
unsigned long previousMillis = 0;
unsigned long intervalo = 5000;
unsigned long tempoLigado = 60000;

//Parâmetros do deep sleep
/* Conversion factor for micro seconds to seconds */
#define uS_TO_S_FACTOR 1000000ULL
/* Time ESP32 will go to sleep (in seconds) */
#define TIME_TO_SLEEP 10770

```

```

RTC_DATA_ATTR int bootCount = 0;

void ativar_Bomba(){
    //Função específica para a ativação do Relé que vai acionar a bomba
    digitalWrite(bomba, HIGH);
    delay(5000);
    digitalWrite(bomba,LOW);
}

void girar_Motor(int sentido, int tempo){
    /*Função específica para a ativação do motor que vai subir a janela .
    A variável sentido indica qual o sentido de rotação e
    a variável tempo quanto tempo o motor vai ficar rodando.
    Sentido horário = 1, Sentido antihorário = -1
    Tempo de rotação em milisegundos.
    Parâmetro velocidade é referente ao dutycycle do PWM.
    Valores de 0 a 255.*/
    if (sentido == 1){
        digitalWrite(motor_IN1,HIGH);
        digitalWrite(motor_IN2,LOW);
        ledcWrite(pwmChannel, dutyCycle);
    }
    else if (sentido == -1){
        digitalWrite(motor_IN1,LOW);
        digitalWrite(motor_IN2,HIGH);
        ledcWrite(pwmChannel, dutyCycle);
    }

    delay(tempo);
}

```

```

digitalWrite (motor_IN1 ,LOW);
digitalWrite (motor_IN2 ,LOW);
}

float leitura_TemperaturaAr(){
    float t = dht.readTemperature();
    return t;
}

float leitura_UmidadeAr(){
    float h = dht.readHumidity();
    return h;
}

int leitura_UmidadeSolo(){
    int x = analogRead(s_UmidS);
    int z = map(x, 2500,4100,100,0);
    return z;
}

int leitura_NivelReservatorio(){
    int y = digitalRead(s_Nivel);
    return y;
}

void enviar_Dados(){

```

```
//Função responsável por enviar as leituras dos sensores pro roteador
```

```
Nivel = leitura_NivelReservatorio();  
UmidSolo = leitura_UmidadeSolo();  
UmidAr = leitura_UmidadeAr();  
TempAr = leitura_TemperaturaAr();  
  
LoRa.beginPacket();  
LoRa.setTxPower(14,RF_PA_CONFIG_PASELECT_PA_BOOST);  
LoRa.print(Nivel);  
//Serial.println(Nivel);  
LoRa.print(UmidSolo);  
//Serial.println(UmidSolo);  
LoRa.print(UmidAr);  
//Serial.println(UmidAr);  
LoRa.print(TempAr);  
//Serial.println(TempAr);  
LoRa.endPacket();  
}
```

```
void Controle(){  
  
int x = leitura_UmidadeSolo();  
int y = leitura_UmidadeAr();  
int z = leitura_NivelReservatorio();
```

```
if (x < 20 and z != 1){
```

```
    ativar_Bomba();
```

```

}

if  (y > 95){
    girar_Motor(-1,340);
    aux = 1;
} else if (y < 95 and aux == 1){
    girar_Motor(1, 200);
    aux = 0;
}
}

void setup() {
    // put your setup code here, to run once:
    dht.begin();
    Heltec.begin(false /*DisplayEnable Enable*/,
    true /*Heltec.LoRa Disable*/,
    true /*Serial Enable*/,
    true /*PABOOST Enable*/, BAND /*long BAND*/);

pinMode(s_UmidS, INPUT);
pinMode(s_Nivel, INPUT);

pinMode(bomba, OUTPUT);
pinMode(motor_IN1, OUTPUT);
pinMode(motor_IN2, OUTPUT);
pinMode(motor_ENA,OUTPUT);

ledcSetup(pwmChannel, freq , resolution);
}

```

```

ledcAttachPin(motor_ENA, pwmChannel);

esp_sleep_enable_timer_wakeup(TIME_TO_SLEEP * uS_TO_S_FACTOR);
delay(10);

while( millis() < tempoLigado){
    if ( millis() - previousMillis >= intervalo){
        previousMillis += intervalo;
        enviar_Dados();
        Controle();
    }
}
delay(50);

LoRa.end();
LoRa.sleep();
delay(100);

esp_deep_sleep_start();

}

void loop() {
    // put your main code here, to run repeatedly:
}

```

Apêndice E

Projeto da Placa do Receptor

Esse anexo contém o esquemático, PCB e modelo 3D da Placa do Receptor.

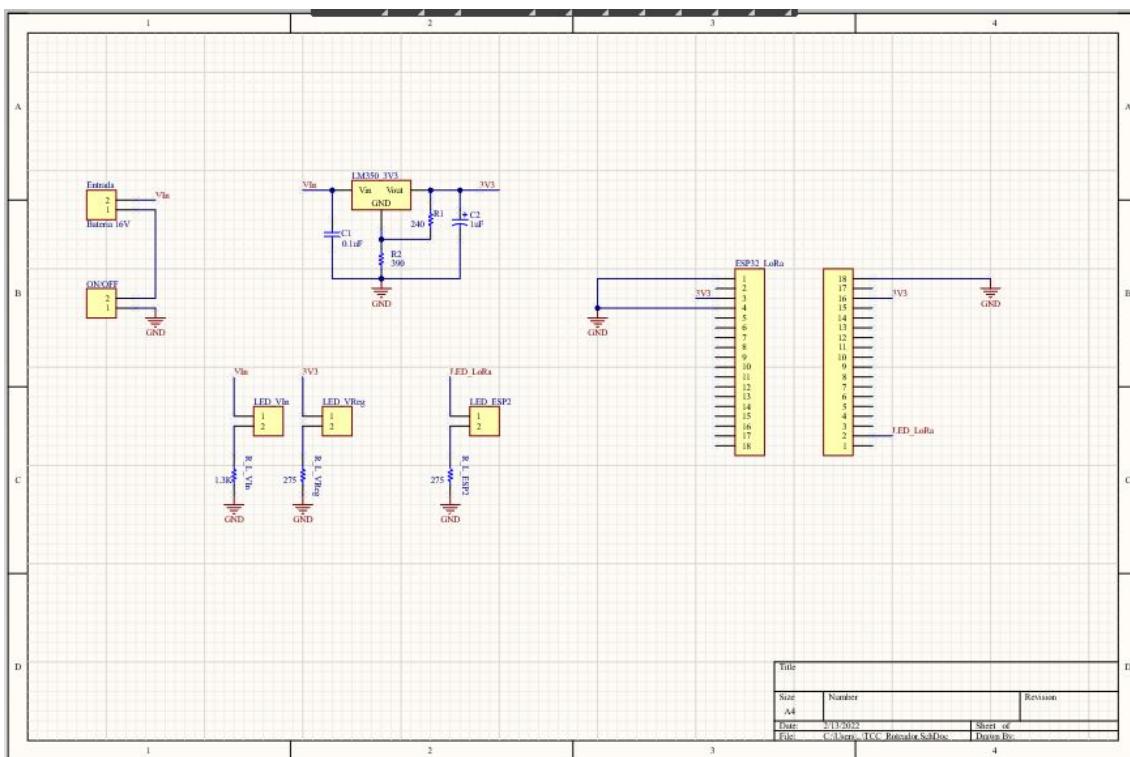


Figura E.1: Esquemático da placa do Receptor

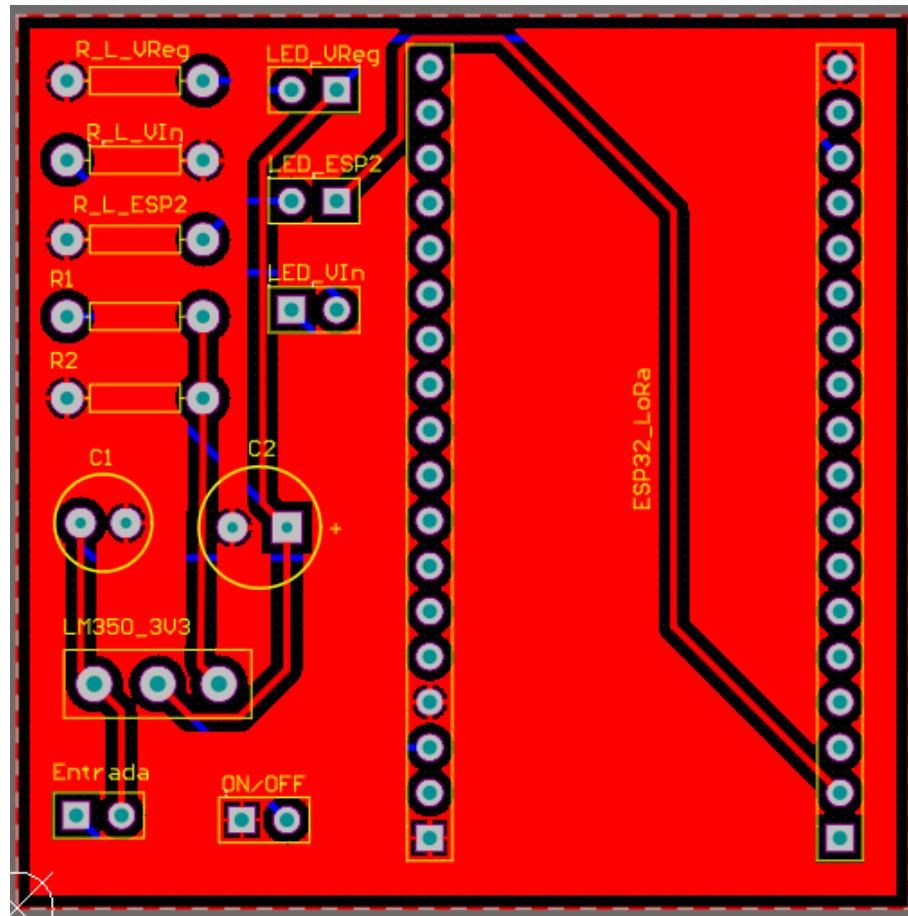


Figura E.2: PCB da placa do Receptor

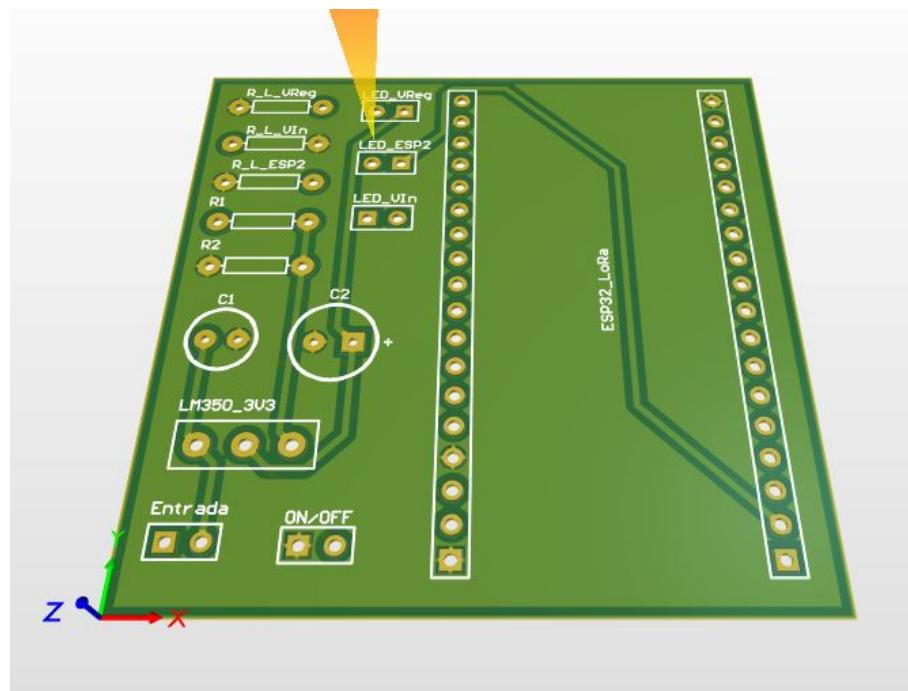


Figura E.3: Modelo 3D da placa de Receptor

Apêndice F

Código do Receptor

Esse anexo contém o código desenvolvido em Arduíno para o ESP32 alocado no Receptor.

```
/*
Check the new incoming messages ,
and print via serialin 115200 baud rate .
```

by Aaron.Lee from HelTec AutoMation , ChengDu , China

www.heltec.cn

this project also realess in GitHub:

https://github.com/Heltec-Aaron-Lee/WiFi_Kit_series

```
*/
```

```
#include "heltec.h"
#include <WiFi.h>
#include "secrets.h"
#include "ThingSpeak.h"
// Always include thingspeak header file
```

```

// after other header files and custom macros.

/* Conversion factor for micro seconds to seconds */
#define uS_TO_S_FACTOR 1000000ULL
/* Time ESP32 will go to sleep (in seconds) */
#define TIME_TO_SLEEP 10770

RTC_DATA_ATTR int bootCount = 0;

#define BAND      433E6 //you can set band here directly ,e.g. 868E6,915E6

char ssid [] = SECRET_SSID; // your network SSID (name)
char pass [] = SECRET_PASS; // your network password
int keyIndex = 0;           // your network key Index number
                           //((needed only for WEP))

WiFiClient client;

unsigned long myChannelNumber = SECRET_CH_ID;
const char * myWriteAPIKey = SECRET_WRITE_APIKEY;

char myData[7];
int number1;
int number2;
int number3;
int number4;

int LED_OUT = 13;

unsigned long previousMillis = 0;

```

```
unsigned long tempoLigado = 60000;

int dataSorting2(char myData[7]){

    char w = myData[3];
    char xy = myData[4];

    String fz = String(w) + String(xy);
    int f2 = fz.toInt();

    return f2;
}

int dataSorting4(char myData[7]){
    char x = myData[0];

    String fx = String(x);
    int f4 = fx.toInt();

    return f4;
}

int dataSorting3(char myData[7]){
    char y = myData[1];
    char z = myData[2];

    String fy = String(y) + String(z);
    int f3 = fy.toInt();
}
```

```

    return f3;

}

int dataSorting1(char myData[7]){
    char xz = myData[5];
    char xw = myData[6];

    String fw = String(xz) + String(xw);
    int f1 = fw.toInt();

    return f1;
}

void WiFiConnect(){
    if (WiFi.status() != WL_CONNECTED){
        Serial.print("Attempting to connect to SSID: ");
        Serial.println(SECRET_SSID);
        while(WiFi.status() != WL_CONNECTED){
            WiFi.begin(ssid, pass); // Connect to WPA/WPA2 network.
            //Change this line if using open or WEP network.
            Serial.print(".");
            delay(5000);
        }
        Serial.println("\nConnected .");
    }
}

void LoRaReceive(){

}

```

```

int packetSize = LoRa.parsePacket();

if (packetSize) {
    int i = 0;
    while (LoRa.available()) {
        //Serial.println((char)LoRa.read());
        myData[ i ] = (char)LoRa.read();
        i += 1;
    }

    number1 = dataSorting1(myData);
    //Serial.println(number1);
    number2 = dataSorting2(myData);
    //Serial.println(number2);
    number3 = dataSorting3(myData);
    //Serial.println(number3);
    number4 = dataSorting4(myData);
    //Serial.println(number4);

    ThingSpeak.setField(1, number1);
    ThingSpeak.setField(2, number2);
    ThingSpeak.setField(3, number3);
    ThingSpeak.setField(4, number4);

    int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
    if(x == 200){
        Serial.println("Channel update successful.");
    }
    else{

```

```

        Serial.println("Problem updating channel.
                        HTTP error code " + String(x));
    }

    digitalWrite(LED_OUT, HIGH);
    delay(1000);
    digitalWrite(LED_OUT, LOW);
}

void setup() {
    //WIFI Kit series V1 not support Vext control
    Heltec.begin(false /*DisplayEnable Enable*/,
                 true /*Heltec.LoRa Disable*/,
                 true /*Serial Enable*/,
                 true /*PABOOST Enable*/,
                 BAND /*long BAND*/);

    pinMode(LED_OUT, OUTPUT);

    ++bootCount;
    Serial.println("Boot number: " + String(bootCount));
    delay(10);

    Serial.println("Acordei!");
}

WiFi.mode(WIFI_STA);
ThingSpeak.begin(client); // Initialize ThingSpeak

```

```
while( millis() < tempoLigado){  
    WiFiConnect();  
    LoRaReceive();  
}  
  
esp_sleep_enable_timer_wakeup(TIME_TO_SLEEP * uS_TO_S_FACTOR);  
delay(10);  
  
LoRa.end();  
LoRa.sleep();  
delay(100);  
  
Serial.println("Indo dormir");  
esp_deep_sleep_start();  
}  
  
void loop() {  
}
```

Apêndice G

Dados do experimento

Esse apêndice contém os dados gerados no experimento utilizando o protótipo. Os parâmetros podem ser identificados pelos headers das colunas. Para tornar a visualização possível, os dados foram divididos em 6 tabelas que seguem os dados ao longo do tempo. As linhas com fonte vermelho representam os dados marcados como outliers na análise.

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
03/05 09:03	27	95	79	0
03/05 09:03	27	95	77	0
03/05 12:08	28	83	79	0
03/05 12:08	28	81	74	0
03/05 12:08	28	79	73	0
03/05 15:07	28	81	72	0
03/05 15:07	29	62	72	0
03/05 18:06	29	59	71	0
03/05 18:07	27	95	71	0
03/05 18:07	27	95	71	0

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
03/05 21:06	25	95	71	0
03/05 21:06	25	95	71	0
03/05 21:06	25	95	71	0
04/05 00:05	24	95	70	0
04/05 00:06	24	95	70	0
04/05 03:05	23	95	69	0
04/05 03:05	23	95	69	0
04/05 06:04	22	95	69	0
04/05 06:04	22	95	69	0
04/05 09:04	42	31	69	0
04/05 09:04	42	27	69	0
04/05 12:03	41	28	68	0
04/05 12:04	29	59	68	0
04/05 12:04	29	58	68	0
04/05 15:03	29	57	68	0
04/05 15:03	26	95	68	0
04/05 15:03	26	95	68	0
04/05 18:02	25	95	68	0
04/05 18:03	25	95	68	0
04/05 21:40	52	39	10	0
04/05 21:41	23	95	91	0
05/05 00:40	23	95	68	0
05/05 00:40	23	95	68	0
05/05 00:41	23	95	68	0

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
05/05 03:40	23	95	67	0
05/05 03:40	23	95	67	0
05/05 06:39	22	95	67	0
05/05 06:39	22	95	67	0
05/05 06:40	22	95	67	0
05/05 09:39	22	95	68	0
05/05 09:39	32	52	68	0
05/05 12:38	31	54	67	0
05/05 12:38	26	89	67	0
05/05 15:38	26	87	66	0
05/05 15:38	25	68	66	0
05/05 18:37	25	68	66	0
05/05 18:37	24	95	65	0
05/05 18:38	24	95	65	0
05/05 21:37	24	95	65	0
05/05 21:37	23	95	66	0
06/05 00:36	23	95	66	0
06/05 00:36	23	95	66	0
06/05 03:35	23	95	66	0
06/05 03:36	22	95	66	0
06/05 03:36	22	95	66	0
06/05 08:58	93	33	10	0
06/05 08:58	36	39	89	0

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
06/05 08:59	36	38	80	0
06/05 11:58	25	85	65	0
06/05 11:58	25	86	65	0
06/05 11:59	25	86	64	1
06/05 14:58	25	87	67	1
06/05 14:58	24	82	67	1
06/05 14:58	24	82	67	1
06/05 17:57	24	85	67	0
06/05 17:58	23	95	67	0
06/05 20:37	42	49	11	0
06/05 20:37	21	94	97	0
06/05 20:37	21	94	89	0
07/05 08:34	22	95	77	0
07/05 11:34	24	95	77	0
07/05 14:33	24	95	77	0
07/05 17:33	23	95	76	0
07/05 19:21	52	29	12	0
07/05 19:22	52	59	10	0
08/05 16:04	52	39	12	0
08/05 16:04	52	69	10	0
08/05 19:04	23	95	76	0
08/05 19:04	21	94	76	0
08/05 19:04	21	94	76	0
08/05 22:03	20	93	75	0
09/05 01:02	20	93	74	0
09/05 01:03	20	93	74	0
09/05 01:03	20	93	74	0
09/05 04:02	20	93	73	0

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
09/05 04:02	19	93	73	0
09/05 04:03	19	93	73	0
09/05 07:01	19	93	73	0
09/05 07:02	19	93	73	0
09/05 10:01	19	93	73	0
09/05 13:01	24	91	73	0
09/05 13:01	24	80	73	0
09/05 13:01	24	80	73	0
09/05 16:00	24	80	72	0
09/05 16:00	24	78	72	0
09/05 16:01	24	80	73	0
09/05 19:00	21	94	72	0
09/05 19:00	21	94	72	0
09/05 19:00	21	94	72	0
09/05 21:34	31	39	12	0
09/05 21:34	31	19	10	0
09/05 21:35	19	93	95	0
10/05 00:34	19	93	73	0
10/05 00:34	19	93	73	0
10/05 00:35	19	93	73	0
10/05 03:33	18	93	72	0
10/05 03:34	18	93	72	0
10/05 06:33	17	92	71	0
10/05 06:33	17	92	71	0
10/05 06:34	17	92	71	0
10/05 09:33	17	92	71	0

Data e Hora	Temperatura do Ar	Umidade do Ar	Umidade do Solo	Nível Baixo
10/05 09:33	29	55	71	0
10/05 12:32	28	56	71	0
10/05 12:32	25	91	72	0
10/05 12:33	25	91	71	0
10/05 15:32	24	90	71	0
10/05 15:32	24	89	71	0
10/05 18:31	22	95	72	0
10/05 21:31	21	94	71	0
10/05 21:31	21	94	71	0
11/05 00:30	20	93	71	0
11/05 03:30	19	93	71	0
11/05 09:13	4	53	12	0
11/05 09:13	84	52	10	0
11/05 12:13	26	81	72	0
11/05 12:13	26	80	72	0
11/05 12:13	26	80	72	0
11/05 15:12	27	65	71	0
11/05 15:13	27	67	71	0
11/05 15:13	27	71	71	0
11/05 18:12	24	95	71	0
11/05 18:12	24	95	71	0
11/05 21:11	23	95	71	0
11/05 21:12	23	95	71	0
12/05 03:10	23	95	71	0
12/05 06:10	22	95	71	0
12/05 06:10	22	95	71	0

Tabela G.1: Dados gerados no segundo experimento