

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Escola de Engenharia
Programa de Pós-Graduação em Engenharia Elétrica

Wuigor Ivens Siqueira Bine

**mannaWUIG: Um *Gateway* de canal
duplo com tecnologia LoRa para
redes de pequeno e médio porte**

Belo Horizonte

2025

Wuigor Ivens Siqueira Bine

mannaWUIG: Um *Gateway* de canal duplo com tecnologia LoRa para redes de pequeno e médio porte

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do Título de Doutor em Engenharia Elétrica.

Orientadora: Linnyer Beatrys Ruiz Aylon

Belo Horizonte
2025

B612m

Bine, Wuigor Ivens Siqueira.

MannaWUIG [recurso eletrônico] : um gateway de canal duplo com tecnologia LoRa para redes de pequeno e médio porte / Wuigor Ivens Siqueira Bine. - 2025.

1 recurso online (129 f. : il., color.) : pdf.

Orientadora: Linnyer Beatrys Ruiz Aylon.

Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia.

Anexos: f. 119-129.

Bibliografia: f. 110-118.

1. Engenharia elétrica - Teses. 2. Internet das coisas - Teses. 3. Hardware - Teses. 4. Redes de sensores sem fio - Teses. I. Ruiz, Linnyer Beatrys. II. Universidade Federal de Minas Gerais. Escola de Engenharia. III. Título.

CDU: 621.3(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS

Escola de Engenharia

Colegiado do Programa de Pós-Graduação em Engenharia Elétrica

FOLHA DE APROVAÇÃO

"Mannawuig: Um Gateway de Canal Duplo Com Tecnologia LORA para Redes de Pequeno e Médio Porte"

Wuigor Ivens Siqueira Bine

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor em Engenharia Elétrica.

Aprovada em 09 de outubro de 2025.

Por:

Profa. Dr. Linnyer Beatrys Ruiz Aylon
DI (UFMG) - Orientadora

Prof. Dr. Antonio Alfredo Ferreira Loureiro
DCC (UFMG)

Prof. Dr. Eduardo Coelho Cerqueira
FECT (UFPA)

Prof. Dr. André Augusto Mariano
DELT (UFPR)

Prof. Dr. Elyson Ádan Nunes Carvalho
DEL (UFS)



Documento assinado eletronicamente por **Linnyer Beatrys Ruiz Aylon, Usuário Externo**, em 28/10/2025, às 14:28, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Andre Augusto Mariano, Usuário Externo**, em 30/10/2025, às 07:36, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Eduardo Coelho Cerqueira, Usuário Externo**, em 03/11/2025, às 14:01, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Elyson Ádan Nunes Carvalho, Usuário Externo**, em 04/11/2025, às 11:36, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Antonio Alfredo Ferreira Loureiro, Professor(a)**, em 08/11/2025, às 07:08, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4625606** e o código CRC **D45B4580**.

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus por me guiar nesta etapa da vida.

À Bianca, pela paciência, pelo apoio e pelo incentivo constantes durante o desenvolvimento deste trabalho, além da confiança no meu potencial. Sou profundamente grato pelo auxílio prestado na fase de desenvolvimento e experimentos, sem o qual esta pesquisa não teria sido viável.

Em especial à Lailla, expresso profunda gratidão por dedicar seu tempo e oferecer apoio em todas as fases deste estudo, incentivando-me constantemente e contribuindo com conselhos valiosos.

Agradeço também ao Alisson pelas críticas construtivas, sugestões e correções, cujo apoio foi igualmente essencial.

Aos meus pais, por estarem sempre ao meu lado, apoiando-me e ajudando-me a trilhar o melhor caminho.

A minha orientadora, Prof^a. Dr^a Linnyer Beatrys Ruiz Aylon, expresso meu profundo agradecimento pelas valiosas sugestões, críticas construtivas e conselhos ao longo deste trabalho. Sua orientação constante, aliada à motivação e ao incentivo, foram fundamentais para minha inserção no universo da pesquisa e para o desenvolvimento dos projetos que resultaram nesta tese.

Ao Ecosystema Manna, pelo apoio fundamental na aquisição de componentes e na disponibilização dos laboratórios necessários ao desenvolvimento deste trabalho, bem como por proporcionar a oportunidade única de vivenciar essa experiência.

"A mente que se abre a uma nova ideia jamais volta ao seu tamanho original."

— Albert Einstein

RESUMO

A utilização da Internet das Coisas (IoT) com tecnologias de redes de longa distância de baixa potência (LPWANs), está proporcionando avanços significativos para diferentes setores, como cidades inteligentes, medição inteligente, agricultura de precisão, indústria 4.0, entre outros cenários, auxiliando na tomada de decisões de soluções/aplicações mais eficientes. A tecnologia LoRa se mostra promissora para redes LPWANs e tem sido foco de estudos e aplicações, empregada em conjunto com o protocolo de comunicação LoRaWAN. Redes que optam por utilizar LoRaWAN possuem as características de *upstream* de acesso aleatório, que em redes com muitos dispositivos promovem colisões de pacotes, congestionamentos, latência e maior consumo de bateria por parte dos dispositivos. Uma solução para esse problema é a utilização de *gateways* com capacidade de receber dados simultâneos em diferentes canais, contudo possuem um elevado custo de aquisição, consumindo boa parte do orçamento para redes privadas de pequeno e médio porte. Este trabalho propõe o projeto e desenvolvimento de um *gateway* inteligente para aplicações IoT em redes de pequeno e médio porte, visando a criação de tecnologia nacional nesta área.

Palavras-chave: *Gateway*, LoRa, LoRaWAN, IoT, RSSF, hardware, múltiplos-canais.

ABSTRACT

The use of the Internet of Things (IoT) with low-power wide area network technologies (LPWANs) is providing significant advances for different sectors, such as smart cities, smart metering, precision agriculture, industry 4.0, among other scenarios, helping decision making for more efficient solutions/applications. LoRa is a promising technology for LPWAN networks and has been the focus of studies and applications used in simultaneousness with the LoRaWAN communication protocol. Networks that choose to use LoRaWAN have the characteristics of random access upstream, which in networks with many devices promote packet collisions, congestion, latency, and higher battery consumption by the devices. A solution to this problem is the use of gateways capable of receiving simultaneous data on different channels. However, they have a high acquisition cost, consuming a good part of the budget for small and medium-sized private networks. This work proposes an intelligent gateway design and development for IoT applications in small and medium-sized networks, aiming for a national technology creation in this area.

Keywords: Gateway, LoRa, LoRaWAN, IoT, RSSF, hardware, multiple-channels.

LISTA DE ILUSTRAÇÕES

Figura 1 – Número total de conexões de dispositivos (incluindo não IoT).	20
Figura 2 – Mercado de IoT empresarial entre 2019 à 2027.	20
Figura 3 – Principais fornecedores de MCU/MPU sendo utilizados em território brasileiro.	31
Figura 4 – <i>Rank</i> global das tecnologias utilizadas em conexões LPWAN.	35
Figura 5 – Estrutura do pacote do transceptor LoRa.	38
Figura 6 – Arquitetura de rede do LoRaWAN.	39
Figura 7 – Frequências para o plano de canal AU915-928.	41
Figura 8 – IoT <i>network</i>	42
Figura 9 – Ciclo da IoT.	50
Figura 10 – Aplicação de <i>gateway</i> de canal duplo e longo alcance, exemplificada com medidores inteligentes. A arquitetura mostra a separação lógica dos dispositivos em dois grupos de comunicação (Canal 1 e Canal 2), permitindo recepção simultânea, redução de colisões e aumento da eficiência em cenários com alta densidade de dispositivos.	52
Figura 11 – mannaHIOT: Plataforma de <i>hardware</i> escolhida (a), modelo do sistema (b) e visão geral do aplicativo móvel (c).	54
Figura 12 – Aplicações utilizando <i>smart gateway</i> de longo alcance em ambientes não urbanos.	56
Figura 13 – Modelo de referência para construção de nós sensores.	61
Figura 14 – Placa mannaWUI.	62
Figura 15 – A arquitetura convencional IoT (a) e arquitetura de rede LoRaWAN (b).	63
Figura 16 – Modelo para um <i>gateway</i> de canal duplo.	64
Figura 17 – Composição do <i>hardware</i>	69
Figura 18 – Circuito projetado para o ESP32-S3FH4R2.	70
Figura 19 – Representação de (a) um filtro L, (b) associação de dois filtros e (c) resultado de uma rede de correspondência do tipo π	71
Figura 20 – Circuito para memória <i>flash</i> e RAM.	73
Figura 21 – Conectores para entrada de energia.	74
Figura 22 – Reguladores de tensão.	75

Figura 23 – Conversor USB-SERIAL.	75
Figura 24 – Circuito para entrar no modo " <i>download boot</i> " automaticamente.	77
Figura 25 – Circuito proposto para o transceptor SX1262.	78
Figura 26 – Detalhamento dos filtros e rede de correspondência para o amplificador PA e LNA.	79
Figura 27 – Exemplificação do circuito equivalente do amplificador PA e conexões da rede de correspondência e filtragem de polarização.	80
Figura 28 – Rede de correspondência RX, circuito "balun-like" proposto pela notas de aplicação.	82
Figura 29 – Circuito para camada física <i>Ethernet</i> utilizando W5500.	83
Figura 30 – Circuito para isolamento da linha de transmissão da camada física <i>Ethernet</i>	84
Figura 31 – Etapas de produção fornecidos por empresas de manufatura de placas de circuito impresso.	85
Figura 32 – Placa finalizada do gateway de canal duplo.	85
Figura 33 – Simulação <i>gateway</i> de canal único.	92
Figura 34 – Simulação <i>gateway</i> de canal duplo.	92
Figura 35 – Mapa onde os dispositivos foram instalados, distribuídos entre os blocos B1 e B2.	95
Figura 36 – Nós sensores usados para testes e o <i>gateway</i> de canal duplo.	95
Figura 37 – PDR segmentado por SF para o primeiro cenário.	98
Figura 38 – PDR segmentado por SF para o segundo cenário.	99
Figura 39 – PDR segmentado por SF para o terceiro cenário.	100
Figura 40 – Gráficos de interação entre o tipo de <i>gateway</i> , canal único (GW1) e canal duplo (GW2), e o fator de espalhamento (SF7 a SF12) sobre a taxa de perda de pacotes (%) nos três cenários experimentais: (a) Cenário 1, (b) Cenário 2 e (c) Cenário 3.	102
Figura 41 – Consumo de corrente durante a transmissão para os fatores de espalhamento SF7 a SF12.	105

LISTA DE TABELAS

Tabela 1 – Resumo comparativo das principais plataformas.	28
Tabela 2 – Resumo comparativo de microprocessadores comerciais.	30
Tabela 3 – Tecnologias de transmissão e suas principais características.	34
Tabela 4 – Sumarização dos estudos sobre tecnologias LoRa e otimizações em redes de comunicação.	46
Tabela 5 – Sumarização dos trabalhos relacionados a aplicações utilizando LoRa. .	48
Tabela 6 – Requisitos do projeto.	66
Tabela 7 – Sumarização dos pinos verificados na inicialização do ESP32-S3FH4R2 (<i>Strapping Pins</i>).	72
Tabela 8 – Booting Mode ESP32-S3.	76
Tabela 9 – Tabela verdade do circuito de autoprogramação.	77
Tabela 10 – Custo para construção de um <i>gateway</i>	86
Tabela 11 – Comparação dos valores de <i>gateways</i> comerciais.	87
Tabela 12 – Parâmetros de simulação.	91
Tabela 13 – Média do CAD.	103
Tabela 14 – Principais placas de prototipagem eletrônica	120
Tabela 15 – Principais sistemas operacionais de tempo real (RTOS)	123

LISTA DE ABREVIATURAS E SIGLAS

A/D	Analógico-Digital
ARM	Advanced RISC Machine
BLE	Bluetooth Low Energy
BW	Bandwidth
CAN	Controller Area Network
CDN	Content Delivery Network
CMMPP	Coupled Markov Modulated Poisson Processes
CPU	Central Processing Unit
CR	Error Correction Rate
CRC	Cyclic Redundancy Check
D/A	Digital-Analógico
DTR	Data Terminal Ready
DMA	Direct Memory Access
DSP	Digital Signal Processor
EC	Edge Computing
EPROM	Erasable programmable read-only memory
FC	Fog Computing
FPGA	Field Programmable Gate Array
HIoT	Healthcare Internet of Things

I ² C	Inter-Integrated Circuit
IIoT	The Industrial Internet of Things
IoE	Internet of Everything
IoT	Internet of Things
ISM	Industrial Scientific and Medical
LNA	Low-Noise Amplifier
LPWAN	Low Power Wide Area Network
LTE	Long Term Evolution
MAC	Media Access Control
MCU	Microcontroller
MPU	Microprocessor
PA	Power Amplifier
PAN	Personal Area Network
PCB	Printed Circuit Board
PHY	Physical Layer
PWM	Pulse Width Modulation
QoS	Quality of Service
RS	Taxa de Símbolos
RISC	Reduced Instruction Set Computer
RTC	Real Time Clock
SELV	Separated Extra Low Voltage
SF	Spreading Factor
SNR	Signal-to-NoiseRatio
SPI	Instant Payment System
TDMA	Time Division Multiple Access
TNV	Telecommunication Network Voltage

TTN	The Thing Network
USB	Universal Serial Bus
UTI	Unidade de Terapia Intensiva
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WWAN	Wireless Wide Area Network

SUMÁRIO

1	Introdução	17
1.1	Motivação	19
1.2	Objetivos	23
1.3	Contribuições e originalidade da pesquisa	24
1.4	Publicação	25
1.5	Organização do trabalho	25
2	Internet das Coisas	26
2.1	Plataformas de Prototipagem em IoT	27
2.2	Principais componentes dos elementos de IoT	28
2.2.1	Unidade de Processamento	28
2.2.2	Unidade de Sensoriamento	32
2.2.3	Unidade de Energia	32
2.2.4	Unidade de Comunicação	33
2.2.5	Sistemas Operacionais de Tempo Real (RTOS)	36
2.3	Sinopse da tecnologia LoRa	36
2.4	Sinopse do protocolo LoRaWAN	39
2.5	Abordagens Computacionais para IoT: Nuvem, Neblina e Borda	42
2.6	Considerações finais	43
3	Trabalhos relacionados	45
3.1	Trabalhos relacionados a rede LoRa/LoRaWAN	45
3.2	Trabalhos relacionados a aplicações utilizando LoRa	47
3.3	Construto de aplicações IoT	50
3.3.1	<i>Gateway</i> de canal duplo em cidades inteligentes	51
3.3.1.1	Medição inteligente	52
3.3.1.2	Semáforo Inteligente	53
3.3.1.3	Internet das Coisas em Saúde (HIoT)	53
3.3.1.4	Internet Industrial das Coisas (IIoT)	55
3.3.2	<i>Gateway</i> de canal duplo na agricultura inteligente	55
3.3.2.1	Fazenda inteligente de animais	56
3.3.2.2	Monitoramento ambiental	57
3.4	Considerações finais	58
4	Modelos de <i>hardware</i> para IoT	59
4.1	A família mannaWUI	60
4.2	Um modelo para o desenvolvimento de um <i>gateway</i> de canal duplo	62
4.3	Considerações finais	67

5	mannaWUIG: Decisões de projeto	68
5.1	Projeto de hardware	68
5.1.1	Unidade de processamento do mannaWUIG	69
5.1.2	Unidade de energia do mannaWUIG	73
5.1.3	Unidade USB do mannaWUIG	75
5.1.4	Unidade de comunicação do mannaWUIG	78
5.1.4.1	Unidade LoRa	78
5.1.4.2	Unidade Ethernet	83
5.2	Manufatura das placas de circuito impresso	84
5.3	Aquisição dos componentes e valores envolvidos	85
5.4	Conclusão e análise de execução do projeto	87
6	mannaWUIG: <i>Gateway</i> LoRa de canal duplo	88
6.1	Adaptação do protocolo LoRaWAN para o <i>gateway</i> de canal duplo	89
6.2	Simulação do <i>gateway</i> de canal duplo	91
6.3	Estudo de caso	93
6.3.1	Cenário 1	95
6.3.2	Cenário 2	96
6.3.3	Cenário 3	96
6.3.4	Resultados	97
7	Conclusão	107
	REFERÊNCIAS	110
	Anexo A Plataformas de IoT para prototipagem eletrônica	119
	Anexo B Principais sistemas operacionais de tempo real (RTOS)	122
	Anexo C Projeto do <i>hardware</i> proposto para o <i>Gateway</i> LoRa de canal duplo	124

Capítulo 1

Introdução

A evolução da tecnologia de dispositivos inteligentes sem fio vem promovendo a popularização da Internet das Coisas (do inglês, *Internet of Things* - IoT), tecnologia que propõe interconectar milhões de objetos físicos à Internet em diferentes aplicações (NGUYEN et al., 2021). Esses objetos se tornam elementos da rede, também chamados de nós, cuja configuração de *hardware* e *software* é dependente da aplicação.

Os elementos da IoT são equipados basicamente com sensores, conversores, atuadores, microcontroladores e/ou microprocessadores, memória, dispositivo de comunicação e fontes de energia, além dos algoritmos (PHAM; RAHIM; COUSIN, 2016). A definição dos componentes de *hardware* e *software* dos nós é de relevante importância para se obter um objeto inteligente autônomo com eficiência no processamento, na comunicação e no consumo de energia.

A configuração básica de uma IoT pode considerar dois tipos de elemento de rede: os nós comuns e os *gateways*. O processamento dos dados coletados pelos sensores dos nós comuns pode ser realizado localmente (no próprio nó) e/ou pelo *gateway*, sendo que neste último o serviço será chamado de computação na borda (*edge computing*) (QIU et al., 2020). O dado também poderá ser enviado para uma plataforma em nuvem (*cloud platform*) para processamento (DONNO; TANGE; DRAGONI, 2019).

Entre a computação na borda e a computação em nuvem pode estar implementada a computação em neblina (*fog computing*). Dentre as plataformas comerciais mais utilizadas para o serviço em nuvem é possível destacar: Amazon AWS¹, Thinger.io², Blynk³,

¹ A Amazon *Web Services* oferece serviços de computação em nuvem escaláveis. Mais informações em: <https://aws.amazon.com>

² Thinger.io é uma plataforma de código aberto criada para simplificar o desenvolvimento de produtos IoT. Mais informações em: <https://thinger.io>

³ A Blynk é uma empresa de tecnologia que desenvolve infraestrutura para a IoT. Mais informações em: <https://blynk.io>

Thingspeak⁴, NETPIE⁵, The Things Network⁶, entre outras. Adotar soluções em nuvem, neblina ou na borda é uma decisão de projeto que deve considerar a latência do processamento dos dados, a velocidade de resposta, o melhor gerenciamento de tráfego de rede e a conformidade com os requisitos de segurança e privacidade.

Ainda sobre a estrutura de uma IoT, diferentes tecnologias de comunicação podem ser adotadas no desenvolvimento das aplicações. Considerando a conexão dos objetos inteligentes, os sistemas têm adotado tecnologias de comunicação tais como Wi-Fi e *bluetooth*. Estudos conduzidos pela academia e indústria dedicam esforços significativos no desenvolvimento de redes de longas distâncias e baixa potência, mais conhecidas pela sigla LPWAN (*Low-power Wide-area Network*) (RAZA; KULKARNI; SOORIYABANDARA, 2017; IKPEHAI et al., 2019; RUOTSALAINEN; ZHANG; GREBENIUK, 2020; JIANG et al., 2021) e nesse contexto, surgiram tecnologias tais como Zigbee, Sigfox, NB-IoT e LoRa.

Dentre as tecnologias citadas, a LoRa conquistou destaque na comunidade acadêmica (ALI et al., 2019; POLONELLI et al., 2019; ISLAM; RAY; PASANDIDEH, 2020). Esta tecnologia usa transceptores comerciais certificados de baixo custo e operam em frequência sub-GHz não licenciada (POLONELLI et al., 2019), isto é, opera em frequências ISM (*Industrial, Scientific and Medical*), em que cada região/país utiliza uma faixa dedicada na banda sub-GHz (KHUTSOANE et al., 2020).

Redes que implementam LoRa geralmente empregam o protocolo LoRaWAN projetado especificamente para esta tecnologia. Essas redes apresentam a característica de *upstream* de acesso aleatório, com base no protocolo ALOHA, o que pode gerar uma explosão no número de transmissões simultâneas, levando a congestionamentos, consequentemente, aumentando o número de colisões e degradação do desempenho (CHASSERAT; ACCETTURA; BERTHOU, 2020; ASTERIOU et al., 2022; ELDEEB; SHEHAB; ALVES, 2022).

Neste contexto, surgem iniciativas de estudos que mostram a necessidade de melhorar as condições na camada de controle de acesso ao meio (MAC), como nos trabalhos apresentados em Shayo, Mafole e Mwambela (2020), Chinchilla-Romero et al. (2021), Asteriou et al. (2022). Contudo, estes trabalhos fazem o estudo de desempenho das modificações e melhorias no protocolo LoRaWAN por meio de simulações numéricas. Outros estudos também propõem contribuições (POLONELLI et al., 2019; YAPAR; TUGCU; ERMIS, 2019; ABDELFADEEL et al., 2020), contudo, são trabalhos teóricos ou realizados por simulação que assumem a utilização de um *gateway*, de alto desempenho e com foco em

⁴ O ThingSpeak é um serviço de plataforma de análise de IoT que permite visualizar e analisar fluxos de dados ao vivo na nuvem. Mais informações em: <https://thingspeak.com>

⁵ A NETPIE destina-se a atender as necessidades dos usuários de dispositivos IoT comerciais, especialmente os do setor industrial. Mais informações em: <https://netpie.io>

⁶ O Things Stack é um servidor de rede LoRaWAN que é o componente crítico para qualquer solução LoRaWAN. Mais informações em: <https://www.thethingsnetwork.org>

redes LoRaWAN públicas, tendo a capacidade para atender diversos canais simultâneos.

Outros estudos voltados para aplicações consideram a definição de redes privadas (PASOLINI et al., 2018; AKHTER et al., 2019; FROIZ-MÍGUEZ et al., 2020; NUGRAHA; ASHADI; EFENDI, 2024), empregando *gateways* com *hardware* equivalente aos demais elementos da rede ou, ainda, *gateways* de canal único. Essa abordagem impõe limitações à capacidade da rede e eleva a probabilidade de colisões de pacotes, especialmente quando múltiplos dispositivos estão em operação simultaneamente ou quando há a presença de redes coexistentes.

Neste contexto, o presente trabalho tem como objetivo discutir e apresentar um modelo de *hardware* para *gateways* que operam com tecnologias de longa distância. As principais contribuições deste estudo são: (i) a apresentação de um modelo de *hardware* para *gateways* de canal duplo utilizando tecnologia LoRa de baixo custo; (ii) a investigação de estratégias para aprimorar a escalabilidade das redes LoRa por meio da CAD (*Channel Activity Detection*) no modelo de *hardware* proposto; e (iii) a discussão dos principais desafios e oportunidades para aplicações de IoT em ambientes urbanos e rurais.

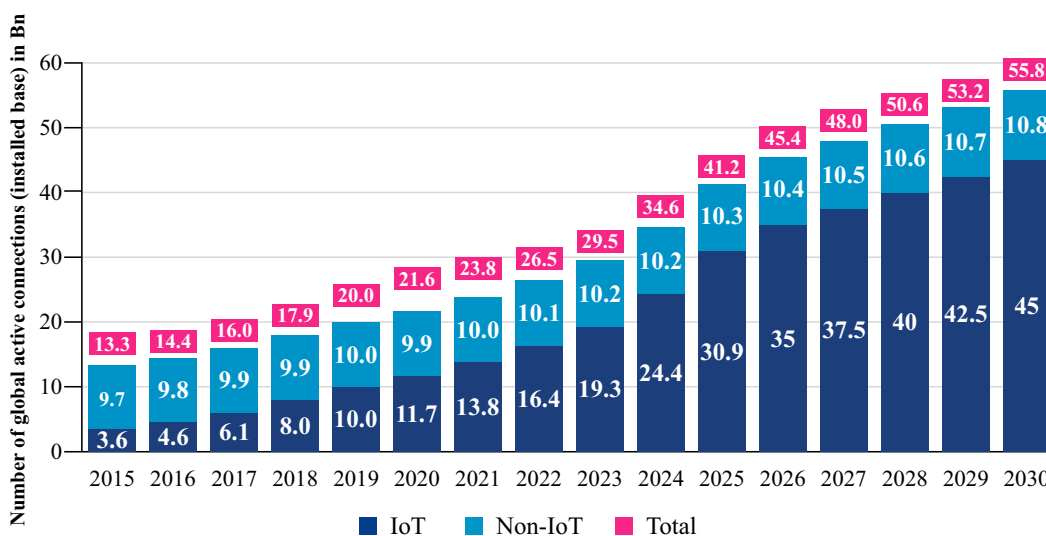
1.1 Motivação

Soluções baseadas em IoT têm sido desenvolvidas para praticamente todos os setores da sociedade, configurando-se como ferramentas relevantes tanto para apoiar políticas públicas quanto para melhorar a qualidade de vida dos cidadãos (ERGUN et al., 2022). Além disso, a IoT apresenta grande potencial como recurso pedagógico, uma vez que, por seu caráter disruptivo e aplicação transversal em diversas áreas, pode ser explorada de forma eficaz no ensino-aprendizagem de conceitos de computação e eletrônica.

A IoT consolidou-se como uma das áreas de pesquisa com maior crescimento na última década. De acordo com levantamento realizado pela IoT Analytics⁷, o número total de dispositivos conectados à Internet, pertencentes ou não ao paradigma da IoT, vem aumentando de forma acelerada. As projeções para 2030 indicam a existência de 55,8 bilhões de dispositivos conectados, dos quais 45 bilhões classificados como dispositivos IoT (SINHA, 2024). A Figura 1 ilustra essa evolução, evidenciando o crescimento exponencial do número de dispositivos IoT em comparação aos demais.

⁷ IoT Analytics é uma empresa líder global em *insights* de mercado e inteligência estratégica para IoT, Inteligência Artificial (IA), Computação em Nuvem, Computação em Borda e Indústria 4.0. Mais informações em <www.iot-analytics.com>

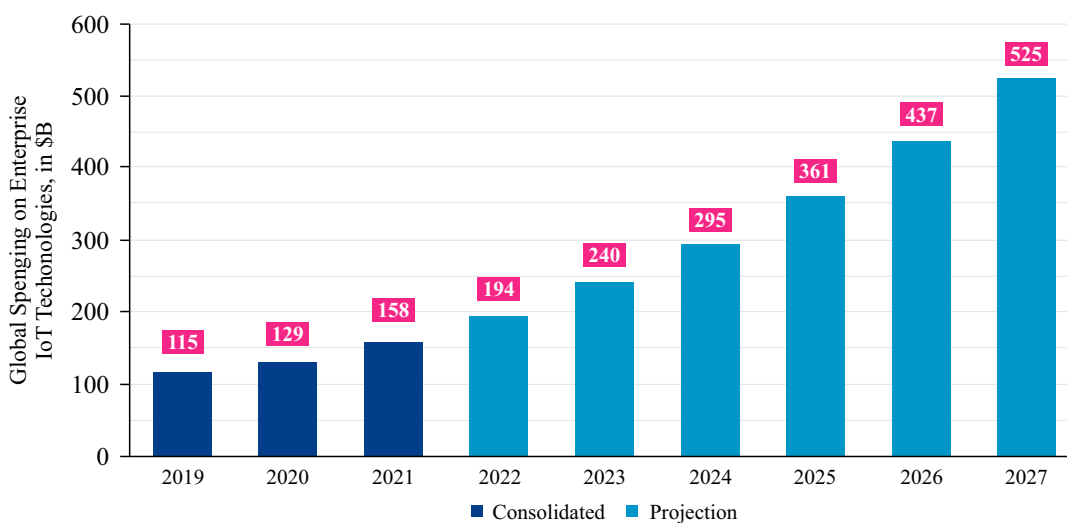
Figura 1 – Número total de conexões de dispositivos (incluindo não IoT).



Fonte: Adaptado de Sinha (2024).

A IoT Analytics realizou um levantamento do valor investido pelo setor empresarial em tecnologia IoT para melhorar processos e prover serviços. O gráfico da Figura 2, apresenta um crescimento de 37% entre os anos de 2019 a 2021 no volume de investimento em tecnologia, totalizando um valor de 158 bilhões de dólares em 2021, considerando as áreas de desenvolvimento de *software*, *hardware*, serviços e segurança em IoT. O levantamento foi enriquecido com um estudo que resultou em projeções futuras com um aumento significativo, chegando a um volume financeiro de 525 bilhões de dólares projetados para o ano de 2027 (WEGNER, 2022).

Figura 2 – Mercado de IoT empresarial entre 2019 à 2027.



Fonte: Adaptado de Wegner (2022).

No Brasil, segundo a Statista⁸, estima-se que existirão mais de 724 milhões de dispositivos conectados até 2027 e que a IoT irá gerar 1,9 a 2,6 milhões de novos postos de trabalhos, acrescentando ao PIB nacional o valor de 122 bilhões de reais.

Diante desse cenário de crescimento expressivo, o governo brasileiro instituiu o Decreto n° 9.854, de 25 de junho de 2019⁹, instituiu o Plano Nacional de IoT, resultado de uma iniciativa conjunta do Ministério da Ciência, Tecnologia e Inovações (MCTI), do Ministério da Economia, do Banco Nacional de Desenvolvimento Econômico e Social (BNDES) e da sociedade civil (empresas, academia, agências de fomento e outros órgãos). A partir desse marco, foram estruturadas diversas câmaras temáticas para impulsionar a transformação digital. Destacam-se a Câmara da Indústria 4.0, criada em abril de 2019, dedicada à incorporação de tecnologias como IoT, inteligência artificial, automação e computação em borda, neblina e nuvem no setor produtivo, e a Câmara Agro 4.0, instituída em agosto de 2019, fruto de cooperação entre o MCTI e o Ministério da Agricultura, Pecuária e Abastecimento (MAPA), com foco em inovações no agronegócio. No âmbito da Estratégia Nacional de IoT, também foram criadas as câmaras Saúde 4.0, Cidades Inteligentes e Turismo 4.0, todas com forte potencial de transformação habilitado pela IoT.

Além disso, o Projeto de Lei Complementar n° 249/2020¹⁰ instituiu o marco legal das *startups* e do empreendedorismo inovador, estabelecendo condições regulatórias favoráveis ao fortalecimento do ecossistema de inovação. Esse conjunto de políticas e estratégias evidencia a necessidade de domínio tecnológico nacional em projeto e desenvolvimento de elementos de *hardware* e *software* para IoT, que se configuram como diferenciais competitivos para o país.

O Ecossistema Manna¹¹, uma das maiores teias de pesquisadores do Paraná e do Brasil, tem desenvolvido pesquisa, ensino e extensão inovadora com o tema IoT e suas variações. Em seus estudos e em sua rotina de desenvolvimento, o Manna não encontrou no mercado ou na literatura uma proposta de *gateway* que usasse canal duplo para dirimir os problemas decorrentes do estado atual de desenvolvimento da tecnologia LoRa e do protocolo LoRaWAN considerando o uso em redes de pequeno e médio porte e custo acessível.

Essa lacuna está diretamente relacionada a principal limitação das redes Lo-

⁸ Statista é uma empresa alemã especializada em dados de mercado e consumidores. De acordo com a empresa, sua plataforma contém mais de 1.000.000 de estatísticas sobre mais de 80.000 tópicos de mais de 22.500 fontes e 170 setores diferentes. Mais informações em: www.statista.com

⁹ DECRETO N° 9.854, de 25 de junho de 2019 - Institui o Plano Nacional de IoT e dispõe sobre a Câmara de Gestão e Acompanhamento do Desenvolvimento de Sistemas de Comunicação Máquina a Máquina e IoT. Mais informações em: http://www.planalto.gov.br/ccivil_03/_ato2019-2022/2019/decreto/D9854.htm

¹⁰ PLP 249/2020 - Esta Lei Complementar institui o marco legal das *startups* e do empreendedorismo inovador. Disponível em: <https://bit.ly/3sq7r4M>

¹¹ www.manna.team

RaWAN, que se deve ao protocolo de acesso ao meio utilizado, o ALOHA. Esse protocolo permite que os dispositivos transmitam pacotes sem realizar uma escuta prévia do canal, o que pode levar a elevadas taxas de colisão em cenários com alta densidade de nós. Embora seja amplamente adotado devido à sua simplicidade de implementação e ao baixo consumo de energia, o modelo ALOHA apresenta baixa eficiência espectral e escalabilidade limitada à medida que o número de dispositivos conectados cresce significativamente (POLONELLI et al., 2019; XUEWEN; KAIHUA, 2023; YU et al., 2020).

Com o objetivo de mitigar tais limitações, a literatura tem proposto diferentes estratégias, entre as quais se destaca o uso do mecanismo de CAD. Esse recurso possibilita que os dispositivos verifiquem o estado do canal antes de iniciar a transmissão, atuando como uma forma de escuta do meio físico. Sua integração a protocolos de acesso múltiplo tem se mostrado eficaz na redução de colisões, no aumento da eficiência da rede e na economia de energia dos dispositivos finais, uma vez que evita retransmissões desnecessárias (ZHANG et al., 2024).

Muitos esforços na academia tem sido dedicados à melhoria da eficiência das redes LoRaWAN. Entre as abordagens mais estudadas, destaca-se a otimização das variantes do protocolo ALOHA, que visa aumentar a eficiência do uso do canal e reduzir as colisões, especialmente em redes densamente povoadas (POLONELLI et al., 2019; TRIANTAFYLLOU; ZORBAS; SARIGIANNIDIS, 2022; KHAN et al., 2022). O CAD tem se destacado como uma estratégia eficaz, permitindo que os dispositivos monitorem o estado do canal antes da transmissão, reduzindo significativamente o número de colisões e melhorando o desempenho global da rede (SHAO; MUTA, 2022; GAMAGE et al., 2023; ZHANG et al., 2024). Complementarmente, o agendamento de recursos de canal surge como uma solução para alocar, de forma controlada, os recursos de comunicação, prevenindo interferências e garantindo transmissões estáveis mesmo em cenários de elevada densidade (JIANG et al., 2021; AFHAMISIS; PALATTELLA, 2022; ASTERIOU et al., 2022).

Apesar dos avanços nas pesquisas voltadas para a otimização do protocolo ALOHA, implementação de CAD e agendamento de recursos de canal, muitas aplicações ainda recorrem ao uso de *gateways* com *hardware* idêntico aos dos nós sensores, como visto nos trabalhos de Pasolini et al. (2018), Akhter et al. (2019), Froiz-Míguez et al. (2020), Nugraha, Ashadi e Efendi (2024). Isso ocorre, principalmente, devido ao alto custo de aquisição de *gateways* com maior capacidade. Nessas soluções, os *gateways* costumam integrar microcontroladores (MCUs) como unidade central de controle, responsáveis pela gestão da rede e armazenamento de dados, além de fornecerem acesso remoto (CHEN; LIN; LIU, 2018). No entanto, seu design de radiofrequência (RF) geralmente inclui transceptores com recursos mais limitados, o que compromete o desempenho em comparação aos *gateways* de maior porte e capacidade.

Diante da crescente demanda por maior eficiência e escalabilidade nas redes

LoRaWAN, a implementação de mecanismos CAD torna-se essencial tanto nos *gateways* quanto nos dispositivos finais. O CAD desempenha um papel importante na otimização do acesso ao canal, atuando como uma etapa inicial para a adoção de protocolos mais avançados, como o CSMA (*Carrier Sense Multiple Access*), que melhora a eficiência no uso do espectro ao reduzir as colisões durante a transmissão de dados (VANGELISTA; CALVAGNO, 2024).

Embora a literatura apresente diversas estratégias para otimizar a comunicação em redes densas baseadas em LoRa, incluindo alocação adaptativa de canais (YU et al., 2020), protocolos de acesso ao meio como CSMA e TDMA (*Time Division Multiple Access*) (NUGRAHA; ASHADI; EFENDI, 2024) e mecanismos de prevenção de colisões (GAILLARD; PHAM, 2023), ainda persistem limitações fundamentais. Primeiramente, muitos estudos partem do pressuposto do uso de *gateways* de alto desempenho, capazes de executar algoritmos complexos, o que se mostra inviável em cenários de baixo custo. Em segundo lugar, uma parcela significativa da literatura concentra-se em arquiteturas de canal único (LE et al., 2021; HEUSSE; CAILLOUET; DUDA, 2023; ZORBAS, 2024), o que limita a recepção simultânea de dados.

Em contraste, o *gateway* proposto por este trabalho é concebido utilizando *hardware* de baixo custo e oferece suporte a processamento local por meio de um microprocessador com capacidades de executar algoritmos de IA leves, tornando-o adequado para ambientes restritos. O modelo proposto supera essa limitação ao incorporar múltiplos transceptores LoRa, possibilitando a recepção concorrente em canais distintos, aumentando a vazão e reduzindo colisões de pacotes.

Este trabalho de tese é uma contribuição ao desenvolvimento de uma família de elementos de IoT no escopo dos projetos financiados do Ecossistema Manna. O projeto de *hardware* para IoT é complexo e a confecção de protótipos configura-se desafiador, em especial no Brasil onde não estão disponíveis comercialmente todos os componentes e dispositivos eletrônicos necessários à montagem do protótipo e onde são poucas as empresas que aceitam encomendas para fabricação de pequenas quantidades de placas.

Além da contribuição para o avanço do estado da arte em IoT, esta tese também contribui com as políticas de capacitação de talentos em IoT tão necessários para a formação de capital intelectual e para responder aos desafios da alta demanda de talentos em TICs no país.

1.2 Objetivos

O objetivo geral do trabalho de tese é estudar o uso da tecnologia LoRa e do protocolo LoRaWAN em IoT, identificando as questões em aberto e propondo soluções em termos de *hardware* que melhorem o desempenho da rede.

Visando atender o objetivo geral, foram determinados os seguintes objetivos específicos:

- Estudar as plataformas de IoT disponíveis, ao que concerne aos seu *hardware*, identificando funcionalidades, dispositivos e componentes apropriados para o projeto e implementação de um elemento de rede;
- Desenvolver um modelo de *hardware* para a concepção de elementos de rede IoT considerando o projeto de um *gateway* de canal duplo de baixo custo para redes de pequeno e médio porte;
- Desenvolver/projetar um protótipo do *gateway* de canal duplo, usando múltiplas camadas físicas como prova de conceitos e *testbed* para outros desenvolvedores;
- Realizar um estudo de caso do *gateway* proposto em uma aplicação, tendo como foco ambientes urbanos.

1.3 Contribuições e originalidade da pesquisa

Algumas abordagens apresentadas por pesquisadores buscam realizar análises de transmissão de dados utilizando tecnologia LoRa, comparando seu alcance e consumo de energia. Outras apresentam métodos de serviço de sincronização para o protocolo LoRaWAN de acesso aleatório, ou alternativas que utilizam algoritmos para alocar *slots* de acesso agendado dos dispositivos da rede baseado em TDMA (*Time Division Multiple Access*). Também são mostradas algumas propostas que buscam utilizar protocolos de comunicação alternativos ou proprietários. Outros apresentam as vantagens de utilizar LoRaWAN em aplicações que necessitam baixo consumo de energia dos dispositivos ingressados à rede.

Esta pesquisa propõe um projeto original para a construção de um *gateway* de canal duplo baseado em tecnologia LoRa para redes privadas de pequeno e médio porte de baixo custo, considerando a utilização do protocolo LoRaWAN ou proprietário. Dentro deste contexto, as contribuições citadas abaixo também são obtidas.

- Estudo das principais aplicações da IoT em diferentes cenários, apresentando um construto de IoT;
- Desenvolvimento de um do modelo de *hardware* proposto para construção de um *gateway* de canal duplo com mecanismo individual para CAD;
- Ampliação da família mannaWUI de dispositivos IoT com a concepção e implementação do *gateway* mannaWUIG;

1.4 Publicação

Esta seção apresenta a publicação produzida no decorrer desta pesquisa, a saber:

- BINE, W. I. S.; AYLON L. B. R. **Design of a Low-Cost Gateway with LoRa Technology Serving Multiple Devices**. *Sensors*, v. 25, n. 16, p. 4948, 2025.

1.5 Organização do trabalho

O Capítulo 2 aborda os conceitos da IoT, introduz as plataformas comerciais mais comuns e apresenta uma sinopse para a tecnologia LoRa e para o protocolo LoRaWAN, além de uma abordagem de computação em nuvem, neblina e borda.

O Capítulo 3 apresenta os trabalhos relacionados a esta pesquisa, organizados em três seções. A primeira seção aborda estudos voltados à rede LoRa. A segunda contempla trabalhos relacionados a aplicações que utilizam LoRa, com o objetivo de investigar os *hardwares* empregados, e, por fim, é apresentado um construto de aplicações em IoT, juntamente com aplicações que podem beneficiar ao utilizar o *gateway* proposto neste trabalho.

O Capítulo 4 apresenta os modelos propostos para o projeto de elementos de rede em IoT, abrangendo tanto os nós *manifold* quanto o *gateway* de canal duplo, que constitui o foco central desta tese.

No Capítulo 5 são apresentadas as decisões e como o *gateway* de canal duplo foi projetado. A composição do *hardware* proposto é organizada em cinco subsistemas, sendo, unidade de energia, unidade USB, unidade de processamento, unidade de RF LoRa e unidade Ethernet, onde cada uma é detalhada.

O Capítulo 6 apresenta o estudo de caso realizado em três cenários distintos e os resultados obtidos utilizando o *gateway* proposto.

Por fim, o Capítulo 7 apresenta as considerações finais, bem como as limitações do estudo e as perspectivas para trabalhos futuros.

Capítulo 2

Internet das Coisas

Este capítulo constitui uma contribuição para a área de IoT, especialmente para pesquisadores e desenvolvedores interessados em criar aplicações. Nele são apresentados os resultados do estudo sobre as principais placas e plataformas comerciais disponíveis para prototipagem.

Os objetos (ou coisas) da IoT podem incluir equipamentos, veículos, dispositivos pessoais, eletrodomésticos, plantas, árvores, residências, edifícios, implementos agrícolas, máquinas, robôs, drones, entre outros. A conectividade desses elementos possibilita o monitoramento dos próprios objetos ou do ambiente em que estão inseridos, além de permitir o controle de funcionalidades e a execução de ações (MAHMOOD, 2019; RAYES; SALAM, 2019). O projeto de cenários em IoT depende diretamente da aplicação, assim como o desenho de *hardware* e *software* de seus elementos de rede, quais sejam: nós comuns e nós de borda ou *gateways*.

A Seção 2.1 introduz as plataformas de IoT mais comuns disponíveis até o momento. A Seção 2.2 apresenta o resumo do estudo dos componentes de *hardware* da unidade de processamento, unidade de sensoriamento, unidade de energia e unidade de comunicação e introduz os sistemas operacionais mais comuns em IoT. A Seção 2.3 introduz a tecnologia LoRa e a Seção 2.4 traz a visão geral do protocolo LoRaWAN. A Seção 2.5 introduz a computação de borda, neblina e nuvem. Por fim, a Seção 2.6 apresenta as considerações finais.

2.1 Plataformas de Prototipagem em IoT

Entre as plataformas de prototipagem eletrônica mais conhecidas e amplamente utilizadas destacam-se: Arduino¹, ESP32², Raspberry Pi³, Banana Pi⁴, STM32L4 Discovery Kit IoT Node, STM32L0 Discovery Kit LoRa⁵ e Microchip RN2903 LoRa Mote Board⁶, entre outras.

Essas plataformas são, em sua maioria, de propósito geral, com *hardware* pouco flexível em relação à customização de seus principais componentes. Além disso, as soluções desenvolvidas com base nelas não são passíveis de patenteamento. Ressalta-se ainda que tais plataformas são produzidas no exterior, exigindo processos de importação para sua aquisição no Brasil.

A Tabela 1 apresenta um resumo das principais plataformas de prototipagem comerciais utilizadas em IoT, destacando características de *hardware*, tais como tipo de microcontrolador ou microprocessador, opções de conectividade, suporte a áudio e vídeo e presença de sensores integrados à placa. A versão detalhada dessa tabela, com informações completas de cada plataforma, encontra-se no Anexo A.

Importante notar que na última linha da Tabela 1 esta listada a última do mannaWUI. A placa também podem ser considerada uma contribuição no escopo deste projeto, uma vez que toda a experiência de projeto e fabricação foram parte das primeiras atividades definidas no trabalho de doutoramento.

A Codebender⁷ realizou uma pesquisa, por meio de coleta de dados de forma anônima, a respeito de quais placas de prototipagem de IoT seus mais de 40 mil usuários utilizam. O resultado apresenta cerca de 55% dos usuários utilizam o Arduino UNO para seus projetos (SPARKFUN; CODEBENDER, 2015). Apesar da popularidade do Arduino Uno essa realidade está mudando porque as aplicações de IoT estão requerendo *design* miniaturizados e capacidade de comunicação sem fio. Assim, alternativas ao microcontrolador utilizado pelo Arduino estão ficando populares como será apresentado na Subseção

¹ Arduino é uma plataforma de prototipagem eletrônica de hardware de código aberto. Mais informações em: www.arduino.cc

² ESP32 é um SoC com transmissor Wi-Fi e microprocessador de arquitetura Xtensa LX7 desenvolvido pela Espressif. Mais informações em: www.espressif.com

³ Raspberry Pi é uma série de computadores de placa única de baixo custo. Mais informações em: www.raspberrypi.org

⁴ Banana Pi é uma série de computadores de placa única que utilizam processadores da Realtek. Mais informações em: www.banana-pi.org

⁵ STM32L4 Discovery Kit IoT Node é uma placa de prototipagem projetada para aplicações IoT de baixo consumo. Mais informações em: <https://www.st.com/en/evaluation-tools/b-l072z-lrwan1.html>

⁶ Microchip RN2903 LoRa Mote Board é um dispositivo baseado no modem RN2903 com tecnologia LoRa, permitindo conexão USB a um computador host para envio de comandos. Mais informações em: <https://bit.ly/3mXXHNW>

⁷ A Codebender desenvolve um serviço de codificação na nuvem, apresentando um editor web da versão Arduino IDE para que você possa codificar, armazenar e gerenciar seus esboços do Arduino na nuvem. Mais informações em: www.codebender.cc

2.2.1. As placas Arduino Uno WIFI REV2, Arduino NANO 33 IoT, Arduino MKR FOX 1200 e Arduino MKRWAN 1300 já apresentam algum tipo de conectividade

Tabela 1 – Resumo comparativo das principais plataformas.

Plataformas	MCU/MPU	Conectividade				Áudio/Vídeo		Sensores Integrados	Comercial
		Wi-Fi	Bluetooth	LoRa	Outros	Câmera	Microfone		
Arduino Uno	ATmega328P	Não	Não	Não	Não	Não	Não	Não	Sim
Arduino Nano 33 IoT	ATSAMD21G18	Sim	Sim	Não	Não	Não	Não	Sim	Sim
NodeMCU	ESP8266	Sim	Não	Não	Não	Não	Não	Não	Sim
ESP32 Devkit	ESP32-D0WDQ6-V3	Sim	Sim	Não	Não	Não	Não	Sim	Sim
ESP32 + LoRa	ESP32-D0WDQ6-V3	Sim	Sim	Sim	Não	Não	Não	Sim	Sim
Micro:bit	nRF52833	Não	Sim	Não	Não	Não	Sim	Sim	Sim
Raspberry Pi Pico	RP2040	Não	Não	Não	Não	Não	Não	Não	Sim
Raspberry Pi Zero	BCM2835	Sim	Sim	Não	Não	Não	Não	Não	Sim
Raspberry Pi 4	BCM2711	Sim	Sim	Não	Não	Sim	Não	Não	Sim
Banana Pi M5	S905X3	Sim	Não	Não	Não	Sim	Sim	Não	Sim
Lemaker Guitar	Actions S500	Sim	Não	Não	Não	Sim	Sim	Não	Sim
B-L475E-IOT01A Discovery kit for IoT	STM32L475VGT6	Sim	Sim	Não	NFC	Não	Sim	Sim	Sim
B-L462E-CELL1 Cellular IoT Discovery kit	STM32L462REY6TR	Não	Não	Não	LTE Cat.M1/ NB-IoT	Não	Não	Sim	Sim
STM32L562E-DK Discovery kit	STM32L562QE16QU	Não	Sim	Não	Não	Não	Sim	Sim	Sim
NVIDIA Jetson Nano	NVIDIA Maxwell e ARM Cortex-A57	Não	Não	Não	Ethernet	Sim	Sim	Não	Sim
mannawUI	STM32F407VGT6	Sim	Sim	Sim	Sub-GHz FSK	Não	Não	Sim	Sim

Fonte: Própria (2025).

2.2 Principais componentes dos elementos de IoT

Nas seções seguintes são apresentados os principais componentes de um dispositivo IoT, sendo organizado em quatro unidades de *hardware*, sendo elas: unidade de processamento (Seção 2.2.1), unidade de sensoriamento (Seção 2.2.2), unidade de energia (Seção 2.2.3) e unidade de comunicação (Seção 2.2.4). Os desenvolvedores de soluções em IoT podem fazer uso do estudo para tomarem conhecimento das características de *hardware* e suas indicações de uso.

2.2.1 Unidade de Processamento

O microprocessador (do inglês, microprocessor - MPU), é a unidade de *hardware* usada para executar o *software* embarcado (Gliwa, 2021). Os primeiros MPU de 8 bits apareceram na indústria no final da década de 1970, o Intel 4004, seguidos pelos famosos Intel 8080 e ou Zilog Z80 (CASS, 2018).

Inicialmente os MPUs fabricados eram capazes de executar instruções pré-estabelecidas para operar os registradores, como incremento e decremento, adição, *reset*, entre outros. Neste período tinha-se a necessidade de se ter uma quantidade significativa de componentes externos como memória ROM (*Read-Only Memory*), memória RAM (*Random Access Memory*) e outros circuitos para realizar a construção dos periféricos necessários, como conversores A/D (Analogicos para Digital), comunicação no padrão RS-232, entre diversas

outras. Com o avanço da eletrônica e sua miniaturização, os engenheiros e *designers* de circuito integrado, passaram a projetar os MPUs embutindo os periféricos no mesmo encapsulamento, iniciando a era dos microcontroladores.

Atualmente, os MPUs são fabricados com unidades de processamento de vídeo e conectividade (como exemplo, modem 5G) no mesmo *chip*, diminuindo a quantidade de componentes externos necessários para determinadas aplicações. Os dispositivos que utilizam MPU possuem maior capacidade de processamento e são amplamente utilizados para construção de microcomputadores, também chamados de computador de placa única (do inglês, *Single Board Computer* - SBC), devido a esta capacidade de processamento, as memórias RAM e FLASH e outros periféricos são feitos por circuitos externos e não necessariamente estão presente no mesmo *chip*. Na atualidade os microprocessadores são amplamente utilizados em sistemas embarcados. Tais sistemas microprocessados possuem a vantagem de seus componentes serem extremamente pequenos, possibilitando adaptar ao *design* de produtos compactos (MEYER-BAESE, 2021).

No Brasil há restrições em questões de aquisição de microprocessadores. Empresas como a Qualcomm⁸, não realizam a venda no varejo de seus microprocessadores, dificultando a aquisição de pequenas quantidades para o desenvolvimento de protótipos que utilizam os mesmos. Deste modo, a Tabela 2 apresenta um comparativo entre as famílias de microprocessadores para aquisição por importação em sites internacionais sendo, STM32MP1, i.MX 7ULP, i.MX 7Solo, i.MX 7Dual, i.MX 8, SAM5D, SAM9, Allwinner D1 e Allwinner H618.

O *design* de dispositivos IoT, é dependente da aplicação, capacidade de processamento e custo do projeto. Assim, o emprego de microcontroladores passa a ser uma alternativa de baixo custo e com redução de componentes externos, antes necessários pelo MPU.

Os microcontroladores (do inglês *microcontroller* - MCU), estão presentes em uma gama enorme de dispositivos em nosso dia-a-dia, como exemplo, controle remoto, microondas, máquina de lavar, forno elétrico, fogões, portões eletrônicos, em equipamentos mais robustos como controladores industriais, interface IHM (Interface Homem Máquina), entre outros. Com a popularização da IoT, esses MCUs passaram a prover conectividade para os dispositivos que operam em aplicações já existentes, por meio da utilização de transceptores que promovem a transmissão de dados sem fio em diferentes tipos de tecnologia.

⁸ Qualcomm empresa produtora de *chips* com tecnologias líderes do mercado como processadores, modem 5G, *bluetooth*, Wi-Fi, *chips* para inteligência artificial. Mais informações em: <https://www.qualcomm.com>

Tabela 2 – Resumo comparativo de microprocessadores comerciais.

Processador	CPU	Video Processing	OpenGL	2D/3D Acceleration	Ethernet	PCIe	Display	OS
STM32MP157	Dual Cortex-A7 cores @ 800 MHz + Cortex-M4 core @ 209 MHz	Não	2.0	Sim	Sim	Não	MIPI-DSI, LCD-TFT	Android, Linux, FreeRTOS
STM32MP153	Dual Cortex-A7 cores up to @ 800 MHz + Cortex-M4 core @ 209 MHz	Não	Não	Não	Sim	Não	LCD-TFT	FreeRTOS
STM32MP151	Single Cortex-A7 core @ 800 MHz + Cortex-M4 core @ 209 MHz	Não	Não	Não	Sim	Não	LCD-TFT	FreeRTOS
i.MX 7ULP	Single Cortex-A7 @800 MHz + Cortex-M4	Não	2.1	Sim	Não	Não	MIPI DSI	Android, Linux, FreeRTOS
i.MX 7Solo	Single Cortex-A7 @800 MHz + Cortex-M4 @200 MHz	Não	Não	Não	Sim	Não	MIPI DSI	FreeRTOS
i.MX 7Dual	Dual Cortex-A7 @ up to 1.2 GHz + Cortex-M4 @200 MHz	Não	Não	Não	Sim	Sim	MIPI DSI	FreeRTOS
i.MX 8	Cortex-A72, Cortex-A53, Cortex-M4F Cortex-M7, DSP NPU GPU	4K (h.265, h.264) 1080p@60fps (h.265, h.264, VP9, VP8)	3.1	Sim	Sim	Sim	2x LVDS 2x MIPI DSI 1x HDMI	Android, Linux, FreeRTOS
SAMA5D2	Arm Cortex-A5 @ 500 MHz	Não	Não	Não	Sim	Não	Graphic LCD	Linux, FreeRTOS
SAMA5D43/ SAMA5D44	Arm Cortex-A5 @ 600 MHz	720p@30fps	Não	Não	Sim	Não	Graphic LCD	Linux, FreeRTOS
SAM9	ARM926EJ @ up to 600 MHz	Não	Não	Não	Sim	Não	Graphic LCD	Linux, FreeRTOS
SAM9M10/M11	ARM926EJ-S @ 400 MHz	Não	Não	2D Graphics	Sim	Não	Graphic LCD	Linux, FreeRTOS
Allwinner D1	XuanTie C906 RISC-V CPU	4K@30fps (H.265)1080p@60fps (H.265)	Não	2D Graphics	Sim	Não	RGB LCD, LVDS, MIPI DSI, HDMI	Linux, FreeRTOS
Allwinner H618	Quad-core Arm Cortex-A53@1.5GHz Arm Mali-G31 MP2 GPU	VP9-10 Profile-2 (6Kp30) HEVC MP-10 (H.265) H.264 MVC@1080p60	Sim	Sim	Sim	Não	RGB LCD, LVDS, MIPI DSI, HDMI	Android, Linux, FreeRTOS

Fonte: Própria (2025).

Os MCUs são formados por uma unidade central de processamento (do inglês, *Central Processing Unit - CPU*), que consiste em um microprocessador, memórias SRAM (*Static random-access memory*), EPROM (*Erasable programmable read-only memory*) e/ou FLASH, além de uma gama de periféricos, como temporizadores, contadores, comparadores analógicos, conversores A/D (Analogico-Digital), D/A - (Digital-Analógico), PWM (*Pulse Width Modulation*) e interfaces de comunicação, sendo USB (*Universal Serial Bus*), USART (*Universal Synchronous Asynchronous Receiver Transmitter*), I²C (*Inter-Integrated Circuit*), CAN (*Controller Area Network*), SPI (*Serial Peripheral Interface*) (JIMÉNEZ; PALOMERA; COUVERTIER, 2014).

Um dos microcontroladores mais comuns e utilizados até a atualidade é o ATmega328P (ver Subseção 2.1), pertencente à família AVR, anteriormente fabricado pela Atmel que, em 2016, foi adquirida pela empresa Microchip. O AVR é um microcontrolador de 8 bits e arquitetura RISC (*Reduced Instruction Set Computer*). Ele se tornou muito popular por meio da plataforma de prototipagem rápida e *open-source* (código aberto) conhecida como Arduino, devido ao seu baixo custo financeiro e por ter uma programação acessível para iniciantes, possuindo uma comunidade ativa que fornece sempre novas bibliotecas para novos periféricos.

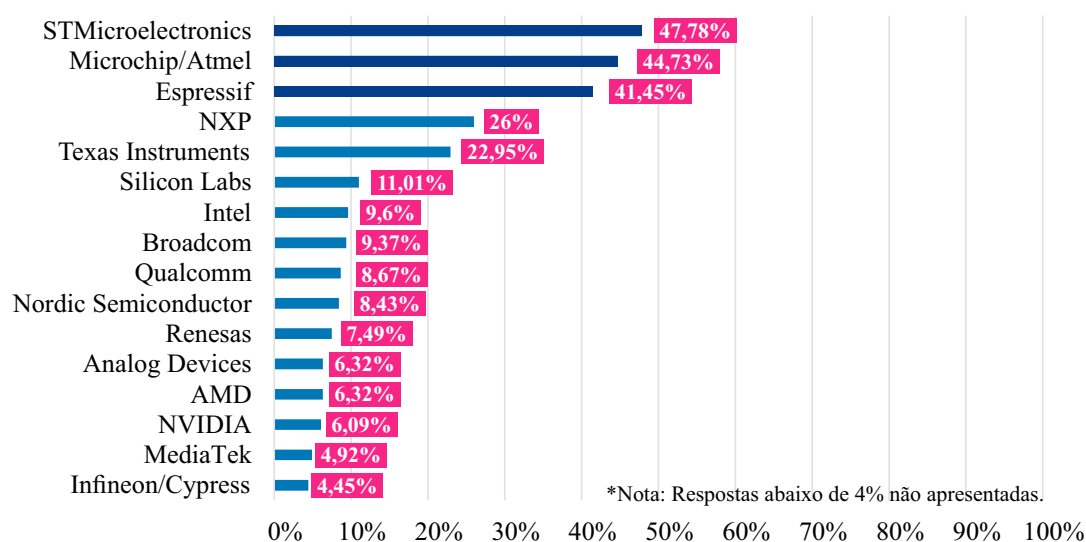
Outras famílias de microcontroladores também são populares, como o caso da

família de microcontroladores PIC16, PIC18, PIC24, dsPIC, PIC32, SAM, STM8, STM32, LPC800, LPC5500, i.MX RT, Nuvoton Series, entre diversos outros fabricantes. Os microcontroladores de 8 bits são muito utilizados em sistemas embarcados devido ao seu baixo custo. No entanto, tem-se restrições em processamentos de sinais rápidos, como monitoramento e controle de motores que atuam em alta velocidade, entre outras aplicações que necessitam de um processamento de sinais rápidos.

Os MCUs baseados em arquitetura ARM (*Advanced RISC Machine*) de 32 bits se destacam pelo fato de, no geral, serem equipados com DSP (*Digital Signal Processor*), tendo um microprocessador dedicado para sinais digitais e FPU (*Float Point Unit*), uma unidade dedicada para executar operações matemáticas em ponto flutuante, tornando possível o processamento de sinais em altas frequências. Outro ponto é o uso do DMA (*Direct Memory Access*) que permite a CPU do MCU continuar processando dados enquanto realiza outras tarefas, como por exemplo uma comunicação SPI ou I²C.

Um levantamento realizado pelo Embarcados⁹ apresenta quais fornecedores de MCU/MPU estão sendo mais utilizados pela indústria e academia brasileira. A Figura 3 destaca os três primeiros lugares, sendo os fornecedores, STMicroelectronics com 47,75%, Microchip/Atmel com 44,73% e Espressif 41,45%. O levantamento também aponta que 56,74% das aplicações desenvolvidas, estão utilizando comunicação sem fio Wi-Fi. Esse ponto mostra que o uso de MCU fornecidos pela Espressif vem ganhando espaço, por disponibilizar um transceptor Wi-Fi e *bluetooth* no mesmo *chip*.

Figura 3 – Principais fornecedores de MCU/MPU sendo utilizados em território brasileiro.



Fonte: Adaptado de Embarcados (2021).

O paradigma da IoT pretende prover conexão a Internet para diferentes objetos

⁹ O Portal Embarcados tem como foco inspirar qualidade e inovação tecnológica, disseminando o conhecimento da área de sistemas embarcados. Mais informações em: www.embarcados.com.br

e aplicações que, antes, já utilizavam algum circuito eletrônico ou não. A tendência no uso de MCU que já encapsulam um transceptor de comunicação sem fio estão passando a ganhar destaque, por permitir construção de placas menores, associado ao baixo custo de aquisição de componentes externos.

2.2.2 Unidade de Sensoriamento

A IoT, como um paradigma que estende a Internet ao mundo físico, necessita de dispositivos para coletar dados e atuar sobre o ambiente. Sensores e atuadores desempenham um papel importante no paradigma da IoT, pois captam os dados do mundo físico e os transformam para digital, ou vice-versa (SEHRAWAT; GILL, 2019). São componentes de *hardware* capazes de realizar leituras de variáveis do ambiente ou realizar alguma ação sobre o mesmo.

Com os dados coletados é possível realizar tomadas de decisões que refletem no acionamento de atuadores. Os atuadores são dispositivos que emitem estímulos visuais ou realizam operações mecânicas como: servo motor, *displays*, LEDs, entre outros. A integração de diversos sensores e atuadores nos dispositivos, aliados à capacidade de comunicação, produz sistemas complexos, com grande potencial de automação e autonomia.

Essa tese não considera o estudo sobre sensores e atuadores, pois cada aplicação exigirá um sensor específico para realizar o monitoramento ou determinado atuador para executar uma ação, mas considera a possibilidade de utilizá-los durante testes do *gateway* proposto.

2.2.3 Unidade de Energia

Os sistemas de *hardware* e de *software* de IoT devem ser projetados para serem eficientes em energia, pressupondo que o dispositivo pode ser instalado em locais sem infraestrutura da rede elétrica. Em geral, as plataformas comerciais de elementos de IoT usam baterias como unidades de energia, mas existem estudos para se usar outras fontes, tais como a coleta de energia (ILLIAS et al., 2020). Ao se projetar tais unidades, é necessário considerar o projeto de conversores DC-DC e reguladores de tensão.

Os conversores são necessários porque os níveis de tensão provisionados por fontes de correntes contínuas, tipicamente possuem valores elevados do que os necessários para os circuitos que compõem o *hardware* do dispositivos IoT. De maneira simples, fontes que fornecem saídas USB, ou as próprias portas USBs disponibilizadas por *desktop* e *notebooks*, por padrão, operam com níveis de tensão de 5 V e a maioria dos componentes do *hardware* operam 3,3 V. Para tornar esse fornecimento de energia apropriado, os elementos precisam de conversores de corrente contínua (DC-DC) ou reguladores lineares.

Os conversores DC-DC são comuns na conversão de níveis de tensão e em alguns

casos são necessários poucos componentes externos, destacando a necessidade de um indutor que em alguns casos pode ocupar um espaço considerável no *design* da placa. No passado não era considerado bons candidatos para sistemas de baixa potência, devido à sobrecarga gerada pelo controle do conversor, aumentando o consumo de corrente, resultando no descarregamento antecipado em sistemas que utilizam baterias (CHEN; FAYED, 2015; ELHEBEARY; YANG, 2020).

Uma alternativa é usar os reguladores de tensão. Os LDO (*Low-dropout*) são reguladores lineares que fornecem uma saída regulada em uma tensão fixa ou ajustável, além de serem componentes pequenos e não requerem indutores robustos, apenas um conjunto de capacitores de desacoplamento. Isso possibilita sua utilização em diversos dispositivos IoT por serem compactos.

Os dispositivos IoT que utilizam microcontroladores de baixa potência começam a operar a partir de 1,8 V. Tensões em milivolts são usadas em circuitos ou sensores específicos. Assim, os reguladores LDO são amplamente utilizados para fornecer tensão ao dispositivo de forma limpa e com baixo ruído em aplicações portáteis e alimentadas por bateria (LIU et al., 2019).

A corrente de inatividade I_Q (*Quiescent Current*) é um parâmetro importante para que o circuito tenha uma eficiência energética. Os dispositivos de IoT, em sua maioria, realizam medições em espaços de tempos predefinidos. No instante de tempo em que não ocorrem medições nem transmissão de dados, o dispositivo está em modo *standby* ou *sleep*, onde o consumo é na escala de μA . Caso o consumo do LDO seja maior que o consumo mínimo do circuito, tem-se um desperdício de energia.

Alguns tipos de LDOs oferecem a possibilidade de desativar a tensão de entrada por meio de um transistor FET interno ao CI, cortando a tensão no pino de saída. Assim, o LDO passa a ter uma corrente de estado de *standby*, em uma escala menor que a corrente I_Q . Alguns circuitos podem fazer uma otimização para que o sistema desligue e ligue apenas em momentos desejados, aumentando consideravelmente o tempo de vida útil da bateria (KENNEDY, 2016).

2.2.4 Unidade de Comunicação

Com o desenvolvimento de aplicações cada vez mais inteligentes que utilizam recursos oriundos de dispositivos IoT, surgindo diferentes tipos de tecnologias para se comunicar com esses dispositivos (nós sensores). Cada tecnologia possui suas características específicas aplicadas em redes com necessidades diferentes, como exemplo, redes que necessitam de alto fluxo de dados possuem alcance de transmissão limitados, logo redes de longo alcance não operam com altas taxas de dados. Os principais tipos de redes sem fio utilizadas são:

- WLAN (*Wireless Local Area Network*): utilizam o meio de troca de dados sem fio como a tecnologia Wi-Fi e são comumente encontradas em residências, empresas, lojas, restaurantes e espaços públicos.
- WWAN (*Wireless Wide Area Network*): as redes WWAN são amplamente utilizadas por operadores de telefonia, para cobrir uma região de determinada cidade.
- LPWAN (*Low Power Wide Area Network*): são redes formadas por tecnologias de transmissão de dados de longas distâncias, onde o dispositivo tem como requisito baixo consumo energético para realizar a troca de dados.
- PAN (*Personal Area Network*): utilizadas em tecnologias de comunicação que transmitem dados a curtos espaços, como o *bluetooth*.

A Tabela 3 apresenta um resumo das principais tecnologias de transmissão e suas características. O Wi-Fi, 5G, LTE e BLE são as tecnologias que possuem taxas de dados alta e são comumente utilizadas em redes domésticas para transmissão de mídia, como vídeo e áudio. As tecnologias ZigBee, EC-GSM, NB-IoT, Sigfox e LoRa possuem menores taxas de dados, mas também estão presente no nosso cotidiano.

Tabela 3 – Tecnologias de transmissão e suas principais características.

Redes	Modulação	Espectro	Largura de Banda	Taxa de dados (max.)
Wi-Fi (802.11a/b/g/n/ac/ax)	BPSK, CCK, QPSK, COFDM, MQAM, 256QAM, 1024QAM	2.4, 5, 6 GHz	20, 40, 80+80, 160 MHz	2 Mbps - 7 Gbps
Bluetooth Low Energy (BLE)	GFSK	2.4 GHz	1 MHz	1 Mbps
ZigBee	DSSS	2.4 GHz	2 MHz	250 kbps
LTE-M (Cat-M1)	QPSK, 16QAM, 64QAM, 256QAM	700 ou 1.800 MHz	1.4MHz	1 Mbps
EC-GSM	GMSK, 8PSK	Bandas GSM	200 KHz	240 kbps
4G/5G/6G	OFDM, CP-OFDM, DFT-S-OFDM, QPSK 16QAM, 64QAM, 256QAM	6 a 90 GHz	50, 400 MHz	20 Mbps - 100 Gbps
NB-IoT	QPSK, 16QAM	450 MHz a 3.5 GHz	180 KHz	85/150 kbps
Sigfox	BPSK, GFSK	868, 902 MHz	192 kHz	100 bps
LoRaWAN	CSS	433, 868, 951 MHz	125, 250, 500 kHz	50 kbps

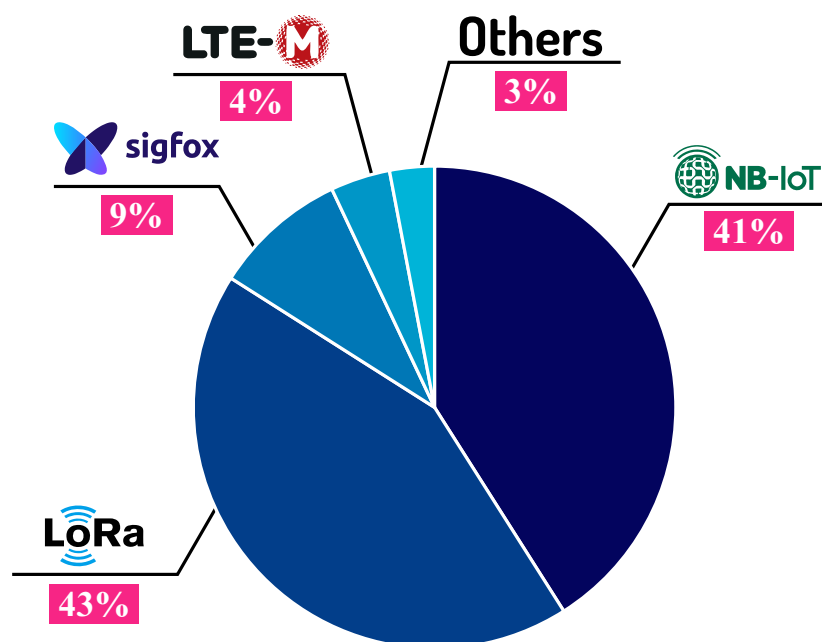
Fonte: Própria (2025).

Dentre as tecnologias, a mais utilizada atualmente para transmissão sem fio em dispositivos é o Wi-Fi, seguido pelo 4G/5G para dispositivos móveis, com capacidade de transmitir altas taxas de dados, em contrapartida, seu consumo energético é relativamente alto e possui um alcance de transmissão limitado. Diversas aplicações que utilizam dispositivos IoT, como apresentados nos trabalhos de Hlaing et al. (2017), Barai, Biswas

e Sau (2017) e Vedaiei et al. (2020) utilizam Wi-Fi para transmissão de dados, porém quando falamos de dispositivos instalados em ambientes de difícil acesso e hostis, outras tecnologias proporcionam maiores vantagens, como baixo consumo de energia e maior alcance comparado ao Wi-Fi e 4G/5G.

Um levantamento realizado pela IoT Analytics apresenta um *rank* global das conexões LPWAN e as principais tecnologias utilizadas, destacando a tecnologia LoRa com 43% de utilização, como visto na Figura 4, seguido por NB-IoT, Sigfox e LTE-M. Embora a tecnologia LTE-M era uma grande aposta para o setor, até a atualidade não desempenham um papel significativo no mercado global (SINHA, 2024). Um levantamento realizado pela Embarcados do cenário brasileiro em relação a aplicações que utilizam comunicação sem fio, mostra que a tecnologia LoRa está presente em cerca de 30,87% das aplicações (EMBARCADOS, 2021).

Figura 4 – *Rank* global das tecnologias utilizadas em conexões LPWAN.



Fonte: Adaptado de Sinha (2024).

Como o uso da tecnologia LoRa vem crescendo a cada ano, e com o aumento da quantidade de dispositivos IoT utilizados nas aplicações, novos desafios surgem para prover conexões para essa demanda crescente de dispositivos conectados em redes de longo alcance. Este trabalho tem como o foco o uso da tecnologia LoRa para o estudo e desenvolvimento de um *gateway* de canal duplo com capacidade de atender múltiplos dispositivos simultâneos.

2.2.5 Sistemas Operacionais de Tempo Real (RTOS)

O sistema operacional de tempo real (do inglês, *Real Time Operating System* - RTOS), é uma camada de *software* para a plataforma de *hardware* que auxilia no desenvolvimento de aplicações e ensino da IoT. Segundo Taivalsaari e Mikkonen (2018), é comum que grande parte das aplicações que realizam a coleta de uma informação ou uma simples atuação no ambiente não necessitem de um sistema operacional, portanto são No-OS (*No Operating System*). Por outro lado, para que os dispositivos suportem um conjunto maior de sensores e atuadores e códigos mais complexos, orientados a eventos e com escalonamento de processos, a um baixo custo de memória RAM (dezenas de kilobytes), a implementação de um RTOS é necessária. Dispositivos com restrição de recursos fazem uso do RTOS, uma vez que oferece suporte ao gerenciamento de recursos, gerenciamento de dados e segurança (SWAMY; KOTA, 2020).

A linguagem de programação *C/C++* é comum aos sistemas embarcados e tem uma curva de aprendizado maior em comparação com linguagens como Python e NodeJS, por exemplo. Essa diferença de aprendizado da linguagem dificulta o processo de ensino e desenvolvimento de aplicações IoT. Plataformas de *hardware* populares como Arduino, Raspberry Pi e Beaglebone adereçam esse problema permitindo a programação em linguagens próprias ou já existentes como Python, MicroPython (um conjunto reduzido de instruções Python), entre outras linguagens modernas. Há estudos que aplicam essas linguagens modernas ao ensino de robótica (KHAMPHROO et al., 2017b; KHAMPHROO et al., 2017a), mostrando que a utilização do RTOS será fundamental para implementar este conceito. A Tabela 15 mostra uma comparação entre os principais RTOS e está disponível no Anexo B.

2.3 Sinopse da tecnologia LoRa

Na literatura, a tecnologia LoRa tem sido considerada em diferentes aplicações urbanas, como cidades inteligentes e em cenários não urbanos, entre elas o agronegócio. Muitos trabalhos passaram a analisar o desempenho da tecnologia LoRa (AUGUSTIN et al., 2016; NOLAN; GUIBENE; KELLY, 2016; YANG et al., 2018; WANG et al., 2020; LOPEZ et al., 2021), com foco em realizar comparações com as demais tecnologias em termos de consumo de energia e taxa de transferência de dados. Comparações realizadas por esses trabalhos apontam para as vantagens da tecnologia LoRa associada ao seu protocolo de comunicação LoRaWAN para o baixo consumo e capacidade de atingir longas distâncias, porém com taxa de dados relativamente baixas, mas que atendem grande parte das aplicações de monitoramento. Apesar do sucesso inicial, existem questões em aberto em termos de escalabilidade, desempenho e tráfego máximo suportado em redes LoRaWAN massivas (POLONELLI et al., 2019).

A tecnologia LoRa é uma técnica de espalhamento espectral que utiliza pulsos *chirp* (do inglês, *Chirp Spread Spectrum* - CSS), sua camada física (PHY) foi desenvolvida pela Cycleo com objetivo de alcançar comunicação de longo alcance e baixo consumo (CITONI et al., 2019). Nos últimos anos essa tecnologia de transmissão de dados é uma das principais para utilização de redes de longa distância e para comunicação com dispositivos em ambientes hostis (EDWARD et al., 2019). Atualmente a empresa Cycleo foi adquirida pela Semtech¹⁰ que patenteou a camada física, e hoje é a principal fabricante de *chips* transceptores com essa tecnologia.

A tecnologia LoRa opera na banda ISM (*Industrial Scientific and Medical*) sub-GHz que em cada país/região tem faixas específicas. A Agência Nacional de Telecomunicações (Anatel) atribuiu no Brasil a faixa de frequência entre 902 a 928 MHz¹¹. O método de modulação CSS utilizado pelo LoRa nas faixas sub-GHz garante alta robustez contra ruídos originados por interferências de propagação de caminhos múltiplos (POLAK; MILOS, 2020).

A camada PHY é caracterizada por três principais parâmetros, sendo eles, SF (*Spreading Factor*), BW (*Bandwidth*) e CR (*Error Correction Rate*). A configuração desses parâmetros define a taxa de dados que os dados são transmitidos, podendo proporcionar um maior alcance de transmissão e estão relacionados ao consumo de energia do transceptor.

- **SF - Spreading factor:** está fortemente relacionado ao tempo necessário para realizar a transmissão de um pacote de dados. Quanto maior o SF, mais *chirps* são necessários para codificar um único bit na modulação. Valores de SF mais altos aumentam a relação sinal ruído (SNR).
- **BW - Bandwidth:** São três larguras de bandas predominantes utilizadas para transmissão, 125, 250 e 500 KHz. Essa faixa de frequência escolhida será a qual o *chirp* LoRa se espalha. Quanto mais alta a largura de banda, maior será a taxa de dados dos pacotes, conseqüentemente, o alcance de transmissão diminui.
- **CR - Error Correction Rate:** O valor definido em CR será igual ao número de bits adicionados ao cabeçalho do pacote para executar técnicas de correção de erros. Quando definidos valores altos para a taxa de codificação, sua robustez à interferência aumenta significativamente, contudo o comprimento do pacote aumenta com o tempo de transmissão e o consumo de energia (CITONI et al., 2019; CATTANI; BOANO; RÖMER, 2017).

¹⁰ Semtech Corporation é fornecedora de semicondutores analógicos e de sinais misto de alto desempenho e algoritmos avançados. Mais informações em: www.semtech.com

¹¹ Atribuição de faixas de frequências no brasil - ANATEL. Disponível em www.bit.ly/2VbwHOs último acesso: 02/04/2021.

O SF pode ser definido com valores entre 6 e 12, definindo a taxa de símbolos a serem enviados. Segundo a folha de dados¹² do transceptor SX1276, quanto mais alto for o SF, maior será a sensibilidade do receptor, aumentando o alcance de comunicação. Em contrapartida, maior será o tempo de envio dos dados. A taxa de símbolos está relacionada com largura de banda utilizada e pode ser calculada a partir da Equação 2.1.

$$R_S = \frac{BW}{2^{SF}} \quad (2.1)$$

Na qual BW é a largura de banda escolhida, SF é o fator de propagação e R_S é a taxa de símbolos.

A Figura 5 apresenta a estrutura padrão do pacote de transmissão de um transceptor LoRa que pode ser separado em preâmbulo, cabeçalho e *payload*. O preâmbulo e cabeçalho são configuráveis e por meio deles é possível deixar explícito as configurações necessárias que o pacote está enviando para ser detectável pelo lado do receptor, também é possível já conhecer essas informações e não ser necessário realizar o seu envio. A seguir é detalhado as três principais características da estrutura do pacote.

Figura 5 – Estrutura do pacote do transceptor LoRa.



Fonte: Própria (2025).

- **Preâmbulo:** A primeira parte da estrutura é utilizada para sincronizar o receptor com o fluxo de dados que está sendo transmitido, por uma sequência de símbolos. Para a recepção intensiva são necessários mais símbolos para reduzir o ciclo de trabalho no receptor, assim economizando energia. A quantidade de símbolos a ser transmitida é programável entre $6 + 4$ à $65535 + 4$.
- **Cabeçalho:** O cabeçalho pode ser configurável de forma explícita ou implícita. A forma explícita transmite as configurações de codificação, tamanho do *payload* e a presença do CRC, porém o tempo de transmissão será relativamente maior comparado ao modo implícito que considera que essas configurações sejam fixas não sendo necessário realizar seu envio.

¹² SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver - Semtech. Disponível em: www.semtech.com/products/wireless-rf/lora-transceivers/sx1276. Último acesso: 02/04/2021.

- **Payload:** O *payload* é onde os dados desejados são enviados codificados com base na taxa de erro contido no cabeçalho explícito ou na taxa conhecida quando usado no modo implícito. O comprimento do *payload* pode ser configurável e um CRC opcional pode ser adicionado ao final. A quantidade de bytes transmitidas influencia no tempo de transmissão, com SF escolhido para operação.

O tempo total de transmissão dessa estrutura é variável e pode ser definida com base na Equação 2.2, considerando o tempo necessário de transmissão dos símbolos do preâmbulo ($T_{preamble}$) e o tempo de transmissão do *payload* ($T_{payload}$).

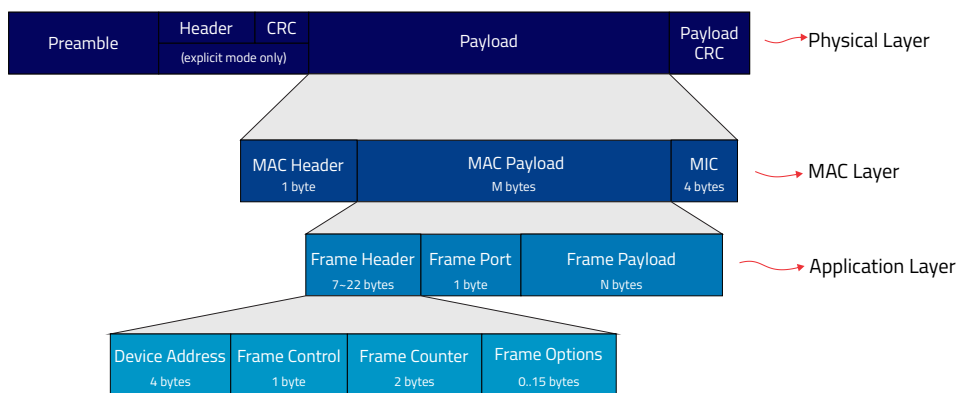
$$T_{packet} = T_{preamble} + T_{payload} \tag{2.2}$$

2.4 Sinopse do protocolo LoRaWAN

O protocolo LoRaWAN é aplicado na camada superior à camada física, denominada MAC (*Media Access Control*), foi construído pensando nas características da tecnologia LoRa, sendo fornecidas por LoRa Alliance¹³. A utilização deste protocolo vem ganhando atenção devido às suas características, que o tornam particularmente adequado para IoT (CITONI et al., 2019).

O protocolo LoRaWAN apresenta limite de comprimento de *payload* que variam conforme a configuração do SF e BW escolhidos e também considera se os comandos MAC devem ser transmitidos no MAC *Payload* (MARAIS; MALEKIAN; ABU-MAHFOUZ, 2019; LoRa Alliance, 2020). Na Figura 6 é apresentada a estrutura do protocolo LoRaWAN que é transmitida dentro do *Payload* da camada física.

Figura 6 – Arquitetura de rede do LoRaWAN.



Fonte: Própria (2025).

¹³ LoRa Alliance é uma associação aberta sem fins lucrativos que se tornou uma das maiores alianças no setor de tecnologia, promovendo e impulsionando o protocolo LoRaWAN. Mais informações em: www.lora-alliance.org

Observando a Figura 6 a camada MAC é subdividida em *MAC Header*, *MAC Payload* e *MIC (Message Integrity Code)*. O *MAC Header* é utilizado para definir a versão do protocolo seguido pelo tipo de mensagem, usado para gerenciamento. O *MAC Payload* é utilizado para enviar os dados desejados e mensagens de solicitação de junção a rede ou de aceitação. O *MIC* é utilizado para evitar a falsificação de mensagens e utiliza o *MAC Header* e *MAC Payload* para calcular seu valor.

A camada de aplicação é organizada em três partes, sendo elas: *Frame Header*, *Frame Port* e *Frame Payload*. O *Frame Header* pode conter de 7 a 22 bytes, nele é onde são adicionados o endereço do dispositivo e outros dados para controle. O *Frame Port* é determinado dependendo do tipo de aplicação, em geral, pode-se utilizar entre 1 à 223, sendo a 224 reservada para testes (LoRa Alliance, 2020). Por último, o *Frame Payload* carrega os dados da aplicação e são criptografados com uma chave de sessão do aplicativo e geralmente essa criptografia é baseada em algoritmo AES 128.

Como observado, o tamanho do *payload* disponível para envio de dados pode ter seu tamanho bem reduzido quando utilizado todas as funções disponíveis do protocolo. O comprimento máximo do *payload* para transmissão de dados sem considerar os comandos MAC, são 51, 115 ou 242 bytes a depender da largura de banda e SF utilizados (MARAIS; MALEKIAN; ABU-MAHFOUZ, 2019).

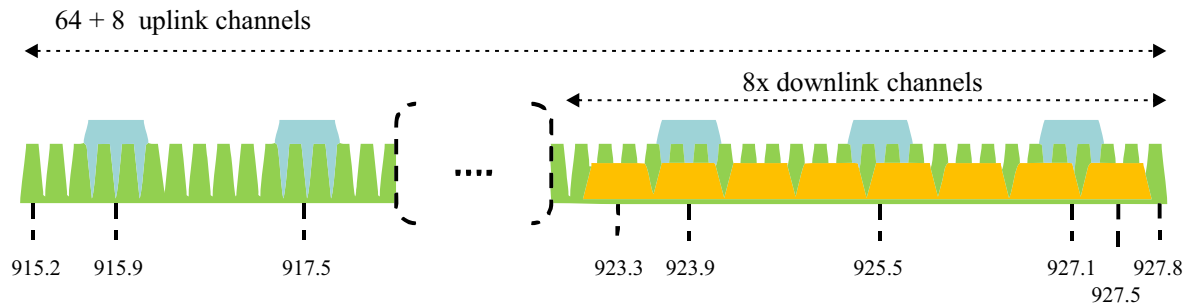
A rede que utilizada LoRaWAN possui topologia estrela e todos os nós sensores suportam comunicações bidirecionais, mas geralmente possui o foco no *uplink* (LAVRIC; PETRARIU, 2018). Assim as redes LoRaWAN são formadas por três principais elementos sendo eles, (i) nós sensores, capazes de coletar dados ou acionar atuadores; (ii) *gateways*, possui a finalidade de prover conectividade entre os nós sensores, atuando como ponte para o caminho dos dados entre os dispositivos e serviços em nuvem; e (iii) servidor de rede, que realiza o gerenciamento da rede, rejeitando pacotes corrompidos e realizando o agendamento de mensagens a serem transmitidas a dispositivos específicos (CITONI et al., 2019).

Segundo LoRa Alliance (2020), a partir da versão 1.0.2, houve uma divisão dos equipamentos terminais, considerados nós, em classes do tipo A, B e C, onde a classe A é considerada sensores, com capacidade de receber dados em janelas de tempo pré-determinado, logo após realizarem uma transmissão de dados, assim essa classe também se caracteriza como comunicação bi-direcional. Os terminais de classe B são considerados atuadores e também possuem comunicação bi-direcional, com janelas de recepção agendadas. E por fim, a classe C é considerada qualquer dispositivo que está sempre apto a receber dados do *gateway* ou transmitir dados para o mesmo.

A normativa RP002-1.0.2 LoRaWAN *Regional Parameters* fornecida pela LoRa Alliance (2020), apresenta os planos de canais aceitos pelo território brasileiro, sendo o US902-928 e AU915-928 que trabalham entre as faixas de 902,3 à 928 MHz. O plano mais

recomendado e utilizado é o AU915-928 apresentado na Figura 7, possuindo as seguintes características:

Figura 7 – Frequências para o plano de canal AU915-928.



Fonte: Adaptado de LoRa Alliance (2020).

- 64 canais de *uplink* de 125kHz, começando em 915,2MHz, separados por 200kHz e fator de espalhamento de espectro entre SF7 à SF12;
- 8 canais de *uplink* de 500kHz, começando em 915,9MHz, separados por 1600kHz, utilizando fator de espalhamento de espectro SF8;
- 8 canais de *downlink* de 500kHz, começando em 923,3MHz, separados por 600kHz e fator de espalhamento de espectro entre SF7 à SF12.

Sumarizando tem-se um total de 72 canais para *uplink* e 8 canais de *downlink*, considerando uma ampla quantidade de canais para *uplink*, os canais passam a ficar congestionados em uma rede TTN pública, considerando milhares de dispositivos conectados, isso faz com que aumente a latência para envio dos dados, devido a colisões que ocasiona a necessidade de retransmissão do dados, também é possível ocorrer a perda de mensagens enviadas quando diversos dispositivos disputam um canal.

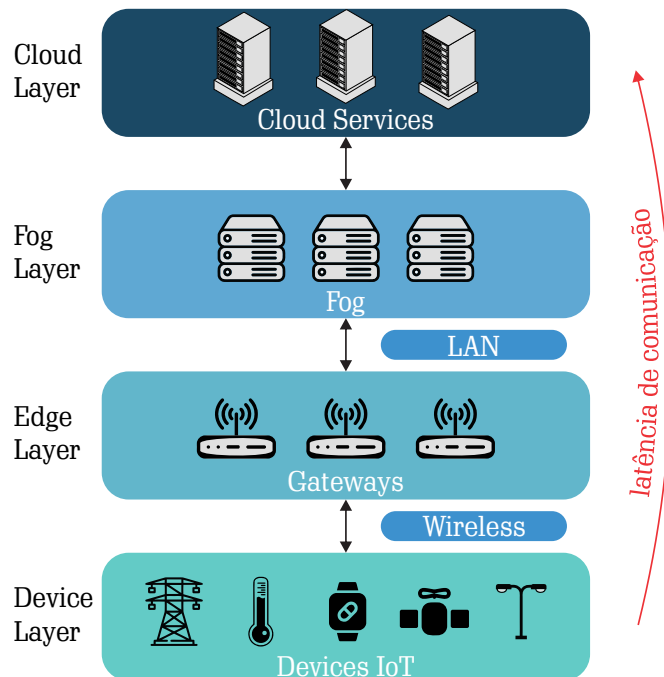
Entre as possíveis soluções para esse problema, destaca-se a utilização de um *gateway* com capacidade de receber múltiplos canais simultaneamente. A Semtech desenvolveu uma família de *chips* com tecnologia LoRa específica para construção de *gateway* com capacidade de receber até 8 canais simultâneos. Porém, a aplicação desses *gateways* em cenário real pode se tornar inviável devido ao custo do hardware, considerando que muitas aplicações implementem redes privadas de pequeno ou médio porte. Dessa forma, este trabalho propõe um modelo de hardware para desenvolvimento de um *gateway* de canal duplo, aumentando a capacidade do *gateway* em receber dados em redes de pequeno e médio porte, que utilizam o protocolo LoRaWAN ou proprietário, proporcionando integração a servidores de rede já existentes e também melhorando o desempenho de redes privadas.

2.5 Abordagens Computacionais para IoT: Nuvem, Neblina e Borda

Nos cenários de IoT, o volume de dados coletados, processados e armazenados apresenta tendência de crescimento exponencial (GOUDARZI et al., 2021). Nesse contexto, uma decisão de projeto fundamental refere-se à definição do local em que esses dados serão processados, armazenados e posteriormente acessados.

A Figura 8 ilustra, de forma simplificada, uma arquitetura organizada em quatro camadas, de acordo com o local de processamento, armazenamento e acesso aos dados: *Device Layer* (computação e/ou armazenamento no próprio nó IoT), *Edge Layer* (computação e/ou armazenamento na borda *Edge Computing*), *Fog Layer* (computação e/ou armazenamento em neblina *Fog Computing*, FC) e *Cloud Layer* (computação e/ou armazenamento em nuvem *Cloud Computing*). A figura também indica a tendência de aumento da latência à medida que se opta por estruturas de processamento e armazenamento mais distantes dos dispositivos finais.

Figura 8 – IoT network.



Fonte: Própria (2025).

O paradigma que envolve o termo borda consiste em dispositivos de rede que estão a um salto da fonte de geração de dados, geralmente os *gateways*. O paradigma de névoa ou neblina refere-se a computação executada entre os dispositivos que promovem a conexão intermediária entre os equipamentos tais como *gateways* e o serviço em nuvem provisionado pelos *data centers* (KANSAL; KUMAR; VERMA, 2022).

Para Ergun et al. (2022), a arquitetura do *Edge Computing* o processamento pode ser executado tanto no *gateways* como nos dispositivos e nós sensores na borda da rede, utilizando para isso *hardware* robusto o suficiente para realizar a computação. De acordo com Losada et al. (2021), a diferença entre o processamento na borda e em neblina é sutil, pois a literatura não apresenta uma definição de consenso que estabeleça a diferença entre os termos.

Segundo Tuvakov e Park (2018), a utilização crescente de dispositivos (nó sensores) estão proporcionando um aumento drástico nos dados coletados e transmitidos para nuvem, esse tráfego de dados pode tornar-se um problema, devido à sobrecargas geradas pelos nós conectados. Uma solução para atender à demanda crescente de tráfego de dados e redução da latência de forma a atender aos requisitos de certas aplicações é implementar o processamento total ou parcial dos dados ainda na borda da rede (LIU et al., 2020).

O surgimento do termo computação de borda, no final da década de 1990, foi promovido pelo surgimento da *Content Delivery Network* - CDN) introduzida por Akamai para melhorar o desempenho de sites e a velocidade de resposta (CHEN; LIU, 2021). O uso do conceito em estruturas IoT tem o objetivo de melhorar o desempenho dos sistemas de computação em nuvem realizando o processamento de dados na borda da rede (SHI et al., 2016; YUAN; LI, 2018). Contribuindo para reduzir o tempo de resposta, dirimir os problemas com a quantidade exponencial de dados e aspectos de segurança, em especial, privacidade, pois não haverá envio de dados brutos para a nuvem (SHI et al., 2016).

2.6 Considerações finais

Este capítulo apresentou uma visão abrangente da IoT, abordando desde a definição de seus elementos básicos até as tecnologias de comunicação, plataformas de prototipagem e paradigmas computacionais que a sustentam. Discutiu-se que os objetos conectados podem assumir diversas formas, de dispositivos pessoais a máquinas industriais, e que sua integração em rede possibilita tanto o monitoramento quanto o controle de ambientes físicos.

Foram analisadas as principais plataformas de prototipagem em IoT, evidenciando suas características de *hardware*, limitações de customização e restrições relacionadas à produção e aquisição no Brasil. Em seguida, exploraram-se os componentes fundamentais de um dispositivo IoT, sendo eles, unidade de processamento, sensoriamento, energia e comunicação. Também foram discutidas as tecnologias de transmissão sem fio mais relevantes, com ênfase nas soluções LPWAN, em particular o LoRa e o protocolo LoRaWAN, reconhecidos por sua capacidade de longo alcance e baixo consumo energético.

O capítulo consolidou os fundamentos necessários para compreender a arquitetura de sistemas IoT e os principais desafios relacionados à escalabilidade, eficiência energética e

gestão de dados. Esses elementos servirão de base, nos próximos capítulos, para a proposta e discussão de um modelo de *gateway* de canal duplo baseado em LoRa, concebido para ampliar a capacidade de redes privadas e contribuir para a evolução de soluções de IoT em diferentes contextos de aplicação.

Capítulo 3

Trabalhos relacionados

Neste capítulo são apresentados os principais trabalhos encontrados na literatura relacionados a redes de longa distância baseadas na tecnologia LoRa, nos quais são analisados e comparados o seu desempenho em diferentes cenários. Os trabalhos selecionados têm como foco o estudo e o desenvolvimento de melhorias em *gateways* que operam com LoRa, destacando-se que grande parte da comunidade acadêmica concentra seus esforços em propor aperfeiçoamentos na camada MAC, apoiada pelo protocolo LoRaWAN, bem como em realizar análises do uso da tecnologia em aplicações reais.

A Seção 3.1 apresenta os trabalhos voltados à melhoria do desempenho das redes LoRa, por meio da proposição de serviços e algoritmos para a sincronização dos dispositivos com o *gateway* e a alocação dinâmica de canais. A Seção 3.2 aborda os estudos relacionados à aplicação da tecnologia LoRa em diferentes áreas. Já a Seção 3.3 introduz o construto de aplicações IoT, juntamente com exemplos de aplicações que se beneficiam da proposta deste trabalho. Por fim, a Seção 3.4 apresenta as considerações finais do capítulo.

3.1 Trabalhos relacionados a rede LoRa/LoRaWAN

As redes de longo alcance e baixo consumo de energia, como as baseadas na tecnologia LoRaWAN, apresentam desafios crescentes em termos de escalabilidade e eficiência na comunicação. Soluções atuais, baseadas no protocolo ALOHA puro, enfrentam limitações significativas em cenários com elevado número de dispositivos, o que resulta em colisões frequentes de pacotes e perda de dados (SINHA; REDDY; PARK, 2024; HASSIJA et al., 2019). Além disso, a ausência de um método eficaz de detecção de atividade do canal compromete ainda mais o desempenho dessas redes, especialmente em cenários de alta densidade de dispositivos (NUGRAHA; ASHADI; EFENDI, 2024).

Tecnologias promissoras, como a introdução de múltiplos canais de recepção e a implementação de algoritmos de CAD, podem contribuir para a mitigação dos problemas de escalabilidade e melhorar o desempenho da rede. No entanto, essas abordagens ainda

não foram amplamente exploradas em redes LoRaWAN de longa distância e ambientes dinâmicos, justificando a necessidade de uma investigação mais aprofundada (SINHA; REDDY; PARK, 2024; HASSIJA et al., 2019; XU et al., 2019).

Para abordar essas questões, múltiplas iniciativas têm sido apresentadas na literatura. A Tabela 4 sumariza e compara os estudos existentes que propõem soluções para redes densas utilizando a tecnologia LoRa, com objetivo de otimizar o protocolo ALOHA, utilização de CSMA e TDMA para melhorar seu desempenho e eficiência.

Tabela 4 – Sumarização dos estudos sobre tecnologias LoRa e otimizações em redes de comunicação.

Ref.	Hardware ou Software	Múltiplos Canais	Objetivo	Método/ Abordagem	Benefícios/ Resultados
(MORABITO et al., 2019)	Software	Sim	Melhorar gateways IoT em termos de escalabilidade, latência e uso de recursos.	Uso de microserviços e virtualização leve com Docker.	Processamento distribuído e modular na borda da rede (<i>edge computing</i>).
(YU et al., 2020)	Software	Sim	Otimizar a alocação adaptativa de canais em redes LoRa.	Alocação adaptativa de múltiplos canais com diferentes larguras de banda.	Redução de interferências, retransmissões e economia de energia.
(LE et al., 2021)	Hardware	Não	Propor um gateway multiprotocolo para IoT.	Integração de ZigBee e LoRa.	Interoperabilidade entre diferentes protocolos e otimização do desempenho.
(HERRERÍA-ALONSO et al., 2022)	Software	Sim	Melhorar a escalabilidade de redes LoRaWAN.	Novo mecanismo de controle de acesso ao meio chamado <i>Longest First Slotted CSMA (LFSCSMA)</i> .	Otimização da utilização do canal e redução de colisões.
(GAILLARD; PHAM, 2023)	Software	Sim	Prevenir colisões em redes densas.	Propõe um método <i>Collision Avoidance by Neighbor Listening (CANL)</i> que utiliza a abordagem de escuta entre nós vizinhos.	Monitoramento de transmissões em andamento e melhoria na confiabilidade da comunicação.
(HEUSSE; CAILLOUET; DUDA, 2023)	Software	Não	Avaliar o desempenho do ALOHA não escalonado.	Análise do fenômeno de captura e colisões múltiplas.	Melhor eficiência em cenários de colisões múltiplas.
(NUGRAHA; ASHADI; EFENDI, 2024)	Software	Sim	Avaliar algoritmos de prevenção de colisões.	Comparação entre TDMA e CSMA.	O TDMA apresenta um desempenho superior na prevenção de colisões. O CSMA mostrou maior flexibilidade e escalabilidade.
(ZORBAS, 2024)	Software	Não	Melhorar a detecção de atividade de canal (CAD).	Aprimorar a escalabilidade das redes LoRa ao melhorar a detecção de atividade no canal.	Redução de colisões e melhor utilização do espectro.
(VANGELISTA; CALVAGNO, 2024)	Software	Não	Desenvolver protocolo com <i>downlink</i> assíncrono.	CSMA/CA com esquema assíncrono para <i>downlink</i> .	Redução de latência e aumento da eficiência da rede.
(ZHANG et al., 2024)	Software	Sim	Seleção otimizada de canais em redes LoRaWAN.	Uso de aprendizado de máquina para a seleção otimizada de canais.	Redução de interferências e maior eficiência na comunicação.

Diversos trabalhos, incluindo aqueles que exploram o TDMA e o CSMA, focam na prevenção de colisões, um desafio significativo em redes densas. Abordagens que envolvem escuta de nós vizinhos, (GAILLARD; PHAM, 2023), ou o aperfeiçoamento dos protocolos CSMA, (HERRERÍA-ALONSO et al., 2022; VANGELISTA; CALVAGNO, 2024; ZHANG et al., 2024), têm buscado minimizar a probabilidade de colisões, aumentando a confiabilidade da comunicação.

Como apresentado na Tabela 4, diferentes trabalhos têm se concentrado em soluções baseadas em *software*, visando aprimorar a comunicação e a escalabilidade das redes por meio de algoritmos e protocolos avançados. Essas abordagens refletem a relevância das otimizações em nível de *software* para lidar com os desafios impostos pela densidade das redes LoRa. No entanto, muitos dispositivos IoT apresentam limitações de *hardware* e

diversas aplicações possuem restrições de custo, o que torna necessário o desenvolvimento de redes menores e mais econômicas.

A literatura evidencia que, em várias soluções propostas, tanto os dispositivos finais quanto os *gateways* são frequentemente implementados utilizando o mesmo *hardware* (YASCARIBAY et al., 2022; MUKHIA; JAYARATHNA; LERTSINSRUBTAVEE, 2023; NUGRAHA; ASHADI; EFENDI, 2024). Além disso, trabalhos que focam em otimizações de *software* frequentemente abordam redes de canal único (LE et al., 2021; HEUSSE; CAILLOUET; DUDA, 2023; ZHANG et al., 2024; VANGELISTA; CALVAGNO, 2024). Nesses casos, é comum que se considere a aplicação dos algoritmos em redes equipadas com *gateways* robustos, no entanto, essa não é sempre a realidade como visto em soluções que utilizam *hardware* idênticos.

Diversos estudos têm explorado o uso de múltiplos canais como estratégia para reduzir interferências e aumentar a capacidade da rede. A utilização adaptativa desses canais mostra-se especialmente eficaz em cenários de alta densidade, seja por meio da aplicação de técnicas de aprendizado de máquina para seleção dinâmica de canais (SINHA; REDDY; PARK, 2024), seja pela adoção de algoritmos de alocação de canais (YU et al., 2020; NUGRAHA; ASHADI; EFENDI, 2024).

As soluções propostas para melhorar o desempenho da rede LoRa (Tabela 4) são frequentemente avaliadas em ambientes de simulação e assumem a utilização de *gateways* robustos para implantação em aplicações reais, o que nem sempre é viável em contextos práticos devido a custos. Nesse sentido, a adoção de um modelo de *gateway* de canal duplo, como o proposto neste trabalho, surge como uma alternativa para redes de pequeno e médio porte.

3.2 Trabalhos relacionados a aplicações utilizando LoRa

Os trabalhos resumidos na Tabela 5 têm como objetivo identificar quais dispositivos são empregados na implementação, testes e análises de aplicações em diferentes ambientes que empregam a tecnologia LoRa. A tabela apresenta uma categorização que inclui o tipo de aplicação, o *hardware* utilizado nos dispositivos, o *hardware* do *gateway* e uma breve descrição de cada estudo.

Em Pasolini et al. (2018), é demonstrado uma proposta para *smart city* voltado a edifícios inteligentes usando tecnologia LoRa. O sistema visa realizar o monitoramento de salas de um edifício usando sensores para obter informações de temperatura, umidade, iluminação e CO_2 . Os dispositivos instalados em cada sala realizam medição a cada cinco minutos e posteriormente transmitido para uma plataforma em nuvem. O autor apresenta

como resultado que a aplicação da tecnologia em ambientes urbanos denso, a cobertura máxima da rede é de 1 a 2 km, podendo ter sua distância reduzida quando o *gateway* é colocado em altitudes mais baixas.

Tabela 5 – Sumarização dos trabalhos relacionados a aplicações utilizando LoRa.

Trabalho	Categoria	Hardware Endpoint	Hardware Gateway	Descrição
Pasolini et al. (2018)	Edifícios Inteligentes	EMB-LR1272-EVK, EMB-LR1272	EMB-LR1272-EVK, EMB-LR1272	O trabalho propõe uma rede utilizando tecnologia LoRa com foco no monitoramento de edifícios inteligentes.
Lozano et al. (2018)	Coleta Inteligente	ATSAML21E18B, SX1276	ATSAML21E18B, SX1276	É proposto a utilização de nós sensores com tecnologia LoRa para monitorar contêiner de lixo pela cidade.
Akhter et al. (2019)	Cidades Inteligentes	ATmega328P, RFM95W	The Things Gateway	Uma rede para contagem de pedestres é empregada utilizando LoRaWAN para transmissão dos dados.
Yu et al. (2019)	Monitoramento da qualidade de água	ATmega32U4, RFM95W	LG01-N <i>Single Channel</i>	O monitoramento de qualidade de água é proposto utilizando dispositivos de baixo consumo e tecnologia LoRa.
Khutsoane et al. (2020)	Indústrias Inteligentes	RN2483	MultiConnect Conduit da Multitech	Um estudo da confiabilidade do link de uma rede LoRaWAN é realizado em um ambiente industrial.
Trigo et al. (2020)	Internet das Coisas Médicas	ATmega32u4, RN2483, PN532	The Things Gateway	Rastreamento de pacientes em um complexo hospitalar de vários edifícios.
Froiz et al. (2020)	Agricultura Inteligente	STM32L1, SX1276	RAK7258	Projeto e validação de um sistema de irrigação inteligente IoT com base em nós de sensores LoRa e LoRaWAN
Neto et al., (2020)	Agricultura Inteligente	ESP32, SX1276	Raspberry Pi, SX1301	Um estudo sobre medições do solo com sensor multiprofundidade para irrigação baseado em IoT e LoRaWAN.
Silveira et al., (2021)	Agricultura Inteligente	ESP32, SX1276	ESP32, SX1276	Um sistema de comutação de energia baseado em LoRaWAN de baixo custo para reduzir o consumo de água.
Raj et al., (2019)	Agricultura Inteligente	ATmega328P, SX1278	ESP8266, SX1278	Monitoramento ambiental de fazendas utilizando tecnologia LoRa.

O trabalho apresentado por Lozano et al. (2018), apresenta uma solução para coleta inteligente de resíduos, utilizando nós sensores e LoRaWAN. O autor apresenta um protótipo de nó sensor sem fio de baixo consumo, responsável por obter medidas de peso, volume de preenchimento e temperatura de contêineres de lixo espalhado pela cidade. O hardware para o protótipo do nó sensor é uma placa que possui como núcleo um ATSAML21E18B, com um transceptor LoRa SX1276, célula de carga e sensor ultrassônico. O hardware do *gateway* é o mesmo do nó sensor, utilizado nove desses dispositivos como *gateway* para atender a aplicação.

Akhter et al. (2019), expõe seu trabalho voltado a cidades inteligentes que realizam a contagem de pedestres utilizando nós sensores baseados em tecnologia LoRa. O dispositivo proposto utiliza uma placa Arduino Uno com um *shield* LoRa RFM95W e sensores de presença infravermelho, temperatura, umidade, pressão e CO_2 . O concentrador utilizado

na rede proposta é um *The Things Gateway*, um *gateway* encontrado comercialmente.

O monitoramento da qualidade da água empregando sensores de baixo consumo e transmissão de longo alcance com LoRaWAN é apresentado por Yu et al. (2019). O nó sensor é equipado com um ATmega32U4 e um *shield* LoRa RFM95W. Um *gateway* comercial é utilizado para receber os dados enviados, empregando o LG01-N de canal único, considerando seu baixo custo.

A aplicação de redes LoRaWAN também são testados em ambientes industriais como o trabalho apresentado por Khutsoane et al. (2020), realizando um estudo de confiabilidade do link e consumo de energia. O dispositivo é composto por um módulo RN2483 e o *gateway* utilizado é o *MultiConnect Conduit* da Multitech.

Em Trigo et al. (2020), é exposto um sistema de rastreamento de pacientes em uma Unidade de Terapia Intensiva (UTI) utilizando LoRaWAN, é apontado a necessidade de identificar a localização dos pacientes em um complexo hospitalar de vários edifícios conectados. O sistema é equipado com o *The Things Gateway* e os nós sensores utilizam um Arduino Leonardo que contém um ATmega32U4 com um módulo LoRa RN2483.

Esforços para análise de aplicações utilizando LoRaWAN na agricultura também são frequentemente encontrados, como o exposto por Froiz-Míguez et al. (2020), propondo um sistema de irrigação inteligente utilizando LoRaWAN. O sistema utiliza como dispositivo final um STM32L1 com um transceptor SX1276 e como *gateway* é utilizado um RAK5205. Outros trabalhos com a mesma temática, de fazenda inteligentes e agricultura de precisão, seguem o mesmo objetivo de analisar a confiabilidade de utilizar LoRaWAN (RAJ et al., 2019; NETO et al., 2020; SILVEIRA et al., 2021).

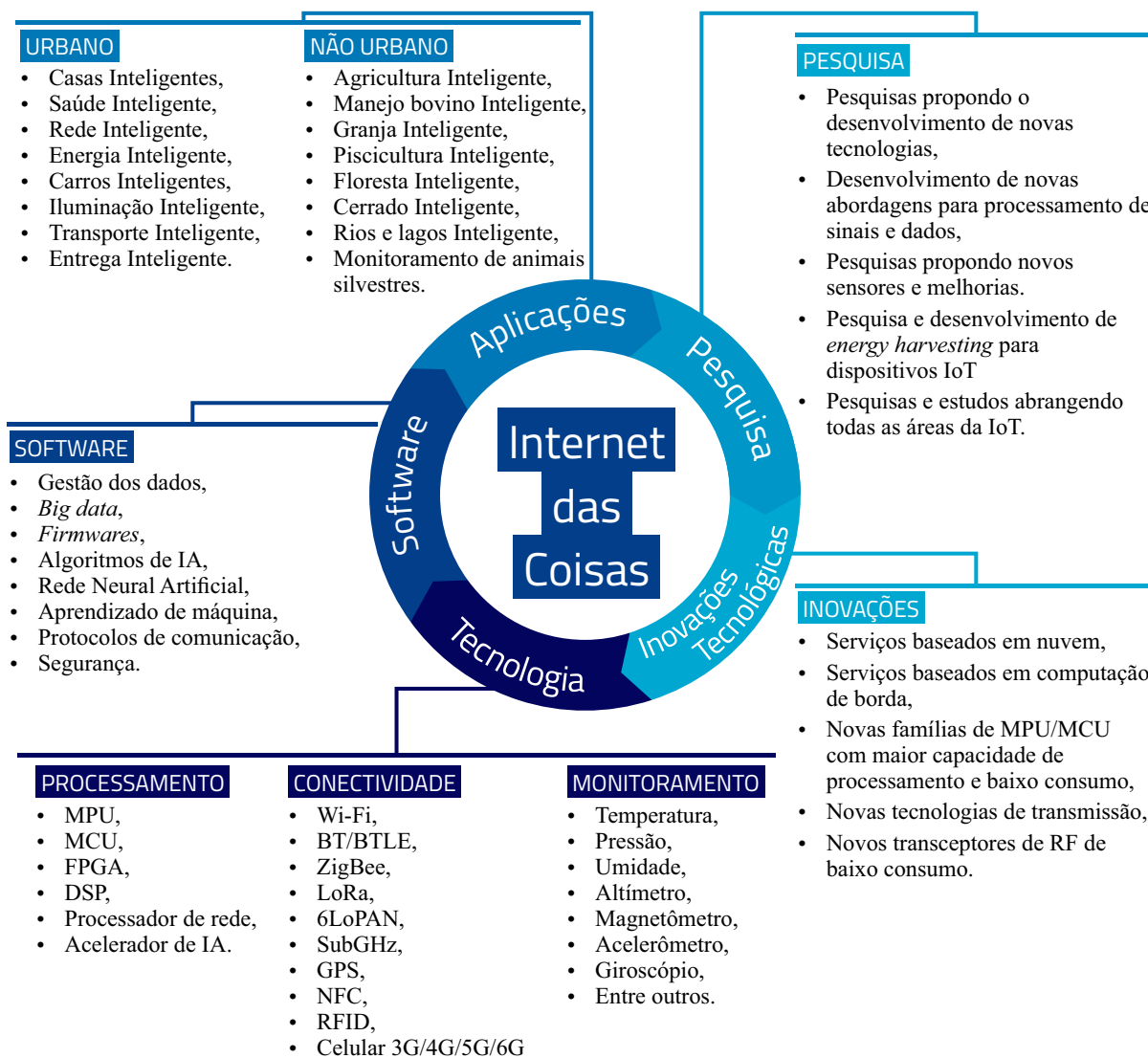
Observa-se que a maioria dos trabalhos analisados emprega *gateways* cujo *hardware* é equivalente ao dos nós sensores, restringindo-se, em geral, à operação em canal único. Embora essa estratégia seja viável em termos de custo e simplicidade, ela limita a capacidade da rede e aumenta a probabilidade de colisões de pacotes, sobretudo em cenários de alta densidade de dispositivos ou em ambientes com múltiplas redes LoRa coexistindo. Essa constatação converge com a discussão apresentada na Seção 3.1, de que muitos trabalhos que propõem melhorias no desempenho de redes LoRa não consideram que, na prática, devido ao elevado custo dos *gateways* multicanais robustos, opta-se pelo uso de soluções de menor custo e capacidade reduzida.

Nesse contexto, o *gateway* de canal duplo proposto neste trabalho mostra-se uma solução promissora para ampliar a escalabilidade e a robustez das redes, sem perder a viabilidade econômica necessária em aplicações privadas de pequeno e médio porte.

3.3 Construto de aplicações IoT

A análise dos trabalhos relacionados permitiu compreender que a IoT se estrutura em um ciclo contínuo e incremental, no qual a demanda por novas soluções impulsiona a geração de tecnologias, arquiteturas, *softwares* e inovações. Quando aplicadas em diferentes cenários, essas soluções produzem resultados que, por sua vez, instigam novos desafios e questões de pesquisa. A partir dessa compreensão, foi elaborado um construto de aplicações IoT, representado na Figura 9, que sintetiza esse processo como uma forma de compreender a IoT a partir das observações e experiências. O esquema ilustra o ciclo de pesquisa, inovação, surgimento de novas tecnologias e aplicações, evidenciando sua retroalimentação contínua e a conseqüente geração de novas demandas de investigação e inovação.

Figura 9 – Ciclo da IoT.



Fonte: Própria (2025).

Considerando contribuir com o ciclo de IoT, este trabalho, além de realizar um estudo detalhado dos componentes de *hardware* dos elementos de IoT, propõe o projeto e desenvolvimento de um *gateway* de canal duplo com tecnologia LoRa e baixo custo. As seções seguintes destacam os aspectos positivos da utilização de *gateways* de canal duplo em alguns cenários e aplicações.

3.3.1 *Gateway* de canal duplo em cidades inteligentes

O crescimento populacional nos grandes centros urbanos tem intensificado problemas sociais e econômicos, como congestionamentos devido ao aumento do tráfego veicular, sobrecarga no transporte público, poluição ambiental e sonora, gestão ineficiente de resíduos sólidos, falhas no saneamento básico e desafios relacionados à segurança pública. Nesse contexto, a IoT surge como uma aliada estratégica para o desenvolvimento de soluções tecnológicas que promovam cidades mais inteligentes e sustentáveis, contribuindo para a melhoria da qualidade de vida da população.

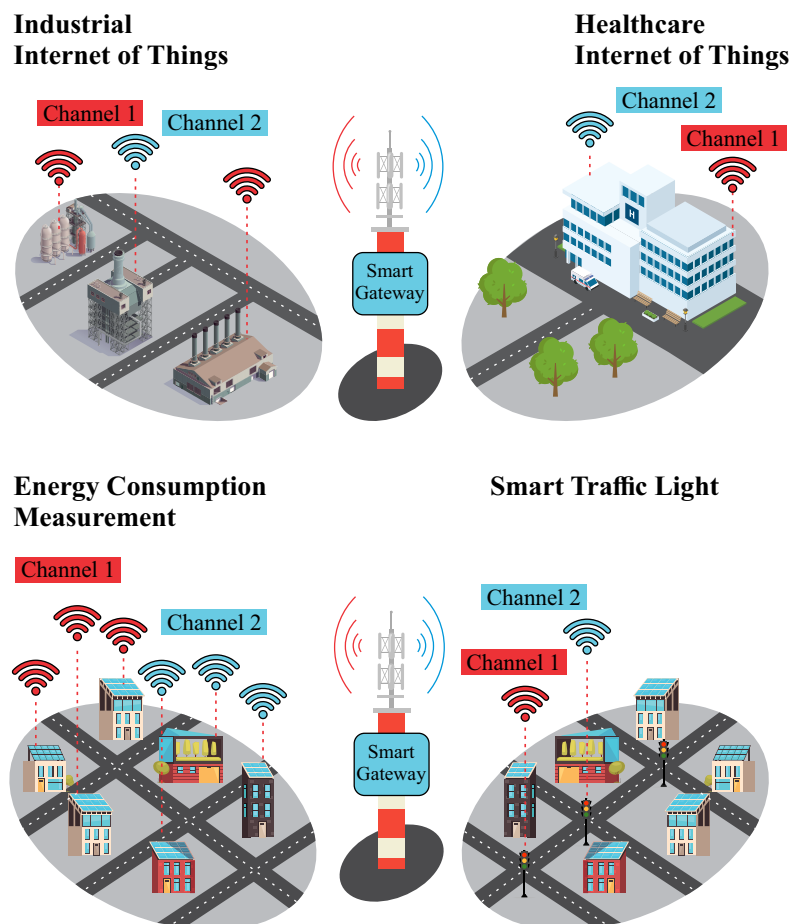
Governos e instituições têm investido na implementação de soluções baseadas em IoT para viabilizar a chamada cidade inteligente (do inglês, *smart cities*). No entanto, a crescente densidade de dispositivos conectados impõe desafios significativos à infraestrutura de comunicação, especialmente no que se refere à escalabilidade, eficiência energética e confiabilidade na entrega de dados (HASSIJA et al., 2019). Em redes LPWAN, como as baseadas em LoRaWAN, essas limitações tornam-se ainda mais críticas. Embora existam *gateways* comerciais de grande porte com capacidade para recepção simultânea em múltiplos canais, seu elevado custo ainda é uma barreira inicial para implantação de aplicações com restrições orçamentárias.

Os trabalhos apresentados por Mishra et al. (2020), Sohail et al. (2019), Chaudhari e Bhole (2018), apresentam soluções de IoT para coleta inteligente de resíduos sólidos utilizando sensores para monitoramento em grandes centros urbanos. O monitoramento da qualidade de ar também está presente em *smart cities* como apresentado em Choudhary et al. (2020), Kumar, Kumari e Gupta (2020), Cheng et al. (2019), considerando o ar como métrica para o bem-estar da população. Os estudos realizados por Kong et al. (2022), Wang et al. (2022), Li e Zheng (2020) dedicam esforços em estudos de mobilidade veicular e transportes inteligentes para *smart cities*.

A utilização de *gateways* de canal duplo surge como uma solução intermediária e economicamente viável, oferecendo ganhos significativos de desempenho em relação aos *gateways* de canal único, ao mesmo tempo em que mantém custos substancialmente inferiores aos modelos comerciais mais robustos. Essa abordagem permite ampliar a capacidade da rede, reduzindo colisões e melhorando a taxa de entrega de pacotes, viabilizando a implementação de sistemas IoT escaláveis e eficientes no ambiente urbano. Exemplos de aplicações com o uso do *gateway* de canal duplo são mostradas na Figura 10 e descritas a

seguir.

Figura 10 – Aplicação de *gateway* de canal duplo e longo alcance, exemplificada com medidores inteligentes. A arquitetura mostra a separação lógica dos dispositivos em dois grupos de comunicação (Canal 1 e Canal 2), permitindo recepção simultânea, redução de colisões e aumento da eficiência em cenários com alta densidade de dispositivos.



Fonte: Própria (2025).

3.3.1.1 Medição inteligente

As aplicações que necessitam realizar monitoramento e gerenciamento de consumo podem implantar a medição inteligente para coletar tais métricas, sendo os empregos mais comuns o de consumo de eletricidade, água e gás, mas no cenário industrial outras possibilidades surgem como a medição de óleos, sementes, grãos e farinhas em tanques de armazenagem (HASSIJA et al., 2019).

A Figura 10 ilustra o uso de medidores inteligentes como múltiplos nós conectados à rede, os quais enviam informações de consumo para as concessionárias de energia. Nesse cenário, a utilização de *gateways* de canal duplo viabiliza a comunicação simultânea com diversos dispositivos, mesmo em regiões com alta densidade de nós sensores. Essa capacidade permite, por exemplo, segmentar a infraestrutura urbana em zonas ou bairros,

com cada grupo de medidores operando em canais distintos, o que contribui para a redução de colisões, otimização do uso do espectro e menor custo de infraestrutura.

Além disso, o *gateway* de canal duplo oferece maior capacidade de recepção em tempo real, possibilitando à concessionária monitorar padrões de consumo. Em centros urbanos também são encontrados muitos pontos sensíveis, como áreas contendo fábricas e indústrias que trabalham com produtos químicos e também usinas de energia nuclear, que oferecem risco aos habitantes, realizar o monitoramento desses ambientes é imprescindível (HASSIJA et al., 2019).

3.3.1.2 Semáforo Inteligente

A utilização de *smart gateway* em vias urbanas têm um papel importante para melhorar o tráfego de veículos, tornando as vias mais rápidas, por meio do monitoramento do fluxo de veículos e controle de semáforos, atuando junto com a aplicação de inteligência artificial para direcionar o fluxo em casos de acidentes ou até mesmo liberar vias para a passagem de veículos especiais como ambulâncias e viaturas policiais.

Apesar de aparentar ser uma aplicação simplista, a utilização de semáforos inteligentes precisa ser estável e funcional, a dessincronização de um deles por falha de comunicação pode promover congestionamentos e dificuldade de locomoção nas vias de grandes centros urbanos.

Considerando ambientes urbanos densamente edificadas ou regiões com alta concentração de vegetação ao redor dos semáforos, torna-se necessário o uso de múltiplos *gateways* para garantir cobertura adequada e comunicação confiável entre os dispositivos. Nesse cenário, a adoção de *gateways* de canal duplo representa uma alternativa com custo de implementação reduzido em comparação aos *gateways* comerciais de maior porte.

3.3.1.3 Internet das Coisas em Saúde (HIoT)

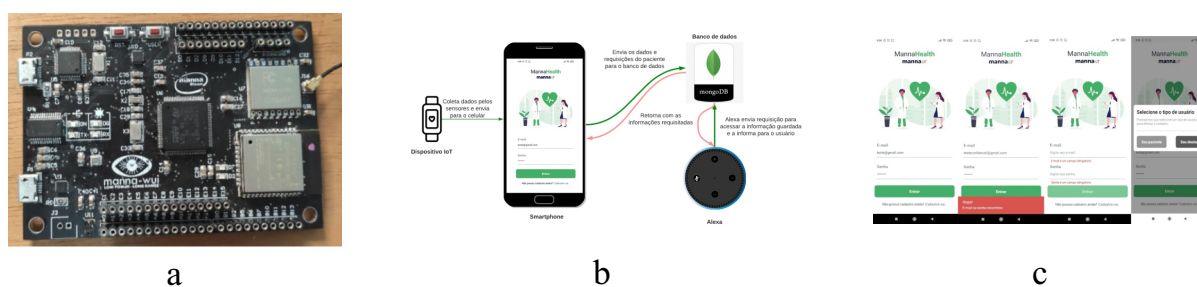
A transmissão confiável de dados de saúde é essencial para o bom funcionamento de redes médicas baseadas em dispositivos conectados, como *wearables*, sensores de sinais vitais e sistemas de monitoramento remoto (NGUYEN et al., 2021). Em ambientes hospitalares, onde múltiplos pacientes são monitorados simultaneamente, a capacidade de resposta da rede torna-se crítica, especialmente em situações que demandam decisões rápidas e intervenções imediatas.

A utilização de *gateways* de canal duplo neste contexto, oferece benefícios relevantes, considerando hospitais ou unidades de saúde com elevada densidade de dispositivos conectados. Essa arquitetura ajuda a reduzir o risco de colisões, aumentando a confiabilidade da comunicação entre sensores e sistemas clínicos. Além disso, a combinação do *gateway* de canal duplo com capacidades de computação na borda possibilita a execução

de algoritmos de aprendizado de máquina para análise local de dados, contribuindo para o diagnóstico precoce, prevenção de doenças e suporte a programas de reabilitação física, com respostas mais rápidas e inteligentes (NGUYEN et al., 2021).

No contexto deste trabalho, foi desenvolvido um projeto de iniciação científica na área da saúde, denominado MannaHIOT, uma aplicação IoT voltada ao monitoramento da saúde no cotidiano (CIZESKI, 2022). O resultado consistiu em um protótipo de dispositivo vestível capaz de coletar dados relacionados à saúde de pacientes idosos. Esses dados podem ser acessados de diferentes maneiras, de acordo com as permissões de usuário e os filtros selecionados. A plataforma de *hardware* escolhida para o desenvolvimento está apresentada na Figura 11 (a), correspondente à placa de prototipagem eletrônica mannaWUI; o modelo do sistema é ilustrado na Figura 11 (b); e a visão geral do aplicativo pode ser observada na Figura 11 (c).

Figura 11 – mannaHIOT: Plataforma de *hardware* escolhida (a), modelo do sistema (b) e visão geral do aplicativo móvel (c).



Fonte: Adaptado de Cizeski (2022).

O protótipo foi desenvolvido utilizando um microcontrolador integrado a acelerômetro e giroscópio. A plataforma estabelece comunicação com o celular do paciente por meio de *Bluetooth* e realiza o envio dos dados ao banco de dados via Wi-Fi. A aplicação móvel foi implementada com o *framework* React Native, em linguagem TypeScript, para o *front-end*, enquanto o *back-end* foi desenvolvido em Node.js utilizando o *framework* LoopBack 4, com armazenamento de dados em um banco MongoDB. Além disso, um dispositivo Amazon Echo Dot foi integrado ao sistema, permitindo a execução de comandos por meio da assistente virtual Alexa.

Embora o protótipo tenha sido concebido inicialmente com comunicação via *Bluetooth* e Wi-Fi, sua arquitetura pode ser expandida para contemplar tecnologias de longa distância e baixo consumo, como o LoRa. Essa adaptação permitiria maior alcance de monitoramento, especialmente em cenários com cobertura de rede limitada. Além disso, a integração com um *gateway* de canal duplo, conforme proposto neste trabalho, possibilitaria a conexão simultânea de múltiplos dispositivos vestíveis, ampliando a escalabilidade do sistema.

3.3.1.4 Internet Industrial das Coisas (IIoT)

Com inúmeras possibilidades que surgem com o paradigma de IoT, uma delas é seu emprego em ambientes industriais. Com aumento da capacidade de sensores de realizarem processamento, porém ainda com determinadas limitações, e novas tecnologias de comunicação, geram um grande crescimento de dispositivos IIoT.

A demanda por dispositivos sem fio em IIoT tem crescido consideravelmente, impulsionada pela necessidade de soluções de *hardware* mais robustas, com menor suscetibilidade a ruídos eletromagnéticos e maior confiabilidade operacional, como promovido pela tecnologia LoRa (NGUYEN et al., 2021).

Em plantas industriais de grande porte, é comum a presença de diversos dispositivos conectados voltados à manutenção preditiva e ao monitoramento contínuo dos processos produtivos. A integração do *gateway* de canal duplo com técnicas de inteligência artificial contribui significativamente para a redução de custos operacionais e do tempo total de produção.

A detecção precoce de falhas em máquinas permite à indústria evitar interrupções inesperadas, que geralmente resultam em perdas financeiras expressivas. Para viabilizar esse diagnóstico, são empregados múltiplos nós sensores distribuídos em equipamentos críticos, responsáveis pela coleta de dados que permitem avaliar, em tempo real, a condição e a vida útil de motores e outros componentes industriais essenciais (TANGE et al., 2020).

Além da elevada densidade de dispositivos dedicados ao monitoramento, sistemas de rastreabilidade também são amplamente empregados em ambientes industriais para o controle de ativos, insumos e produtos que utilizam tecnologia LoRa (TANGE et al., 2020). Nesses cenários, a utilização de *gateways* de canal duplo permite a segmentação eficiente dos dispositivos em múltiplos canais de comunicação melhorando a escalabilidade da rede. Essa abordagem contribui para a redução de colisões, sem demandar investimentos elevados em infraestrutura.

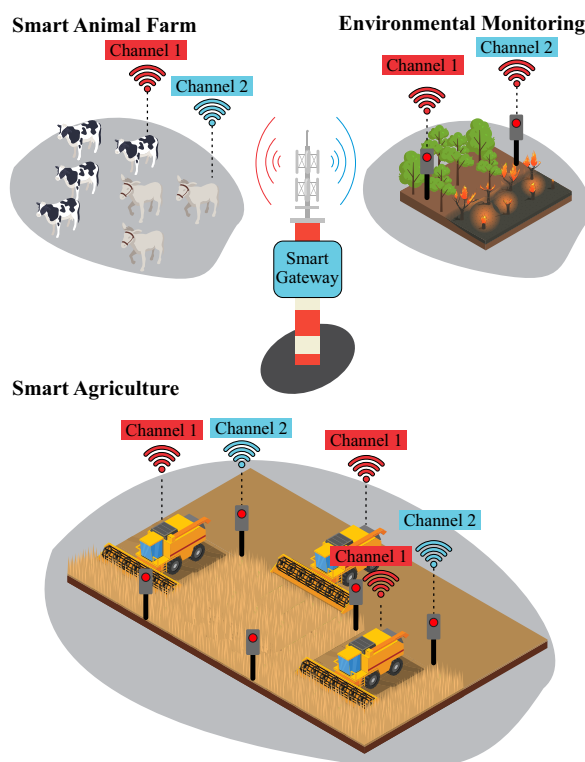
3.3.2 Gateway de canal duplo na agricultura inteligente

A utilização de tecnologias na agricultura expande rapidamente nos últimos anos, onde aplicações de técnicas de IA são úteis para otimizar os processos agrícolas (NGUYEN et al., 2021). Grandes áreas de cultivos tendem a ter diversos sensores coletando informações do ambiente, muitas dessas aplicações processam suas informações na nuvem, podendo levar um tempo significativo para obtenção de respostas. Outro ponto crítico é a cobertura do sinal de operadoras em ambientes não urbanos, havendo dificuldade para realizar a troca de dados ou mesmo não tendo cobertura.

A Figura 12 ilustra algumas das aplicações que se beneficiam da adoção de *gateways* de canal duplo, os quais, além de viabilizar a transmissão de dados em longas distâncias,

apresenta-se como uma solução eficaz para os desafios de conectividade e escalabilidade enfrentados nesse setor. Ao permitir a recepção simultânea de dados em múltiplos canais, essa arquitetura promove uma comunicação mais robusta e confiável entre os sensores distribuídos no campo e os sistema de controle. Essa capacidade é especialmente relevante em regiões remotas, onde a infraestrutura de comunicação convencional é limitada ou inexistente, exigindo soluções que operem de forma autônoma e resiliente às condições adversas do ambiente rural.

Figura 12 – Aplicações utilizando *smart gateway* de longo alcance em ambientes não urbanos.



Fonte: Própria (2025).

3.3.2.1 Fazenda inteligente de animais

O monitoramento do solo para detectar distúrbios em suas propriedades, acelera a recuperação do pasto, auxiliando no manejo de animais em grandes fazendas. A detecção de doenças com coleta de dados dos animais auxilia na prevenção de doenças, zelando pelo bem estar animal.

A utilização de *gateways* de canal duplo, associada à aplicação de técnicas de IA, viabiliza a execução de algoritmos de aprendizado de máquina diretamente na borda da rede. Essa abordagem permite respostas rápidas para a tomada de decisões críticas, especialmente em ambientes rurais ou remotos, onde frequentemente há ausência de cobertura de redes móveis ou conexões instáveis. A capacidade de processar e reagir localmente aos dados

coletados reduz a dependência de infraestrutura centralizada, aumentando a autonomia e a eficiência operacional das aplicações agrícolas.

O custo de implementação de um *gateway* de canal duplo de longo alcance é relativamente reduzido, o que o torna uma alternativa economicamente viável para ambientes distribuídos, como áreas agrícolas extensas. Essa solução permite a redução da quantidade de repetidores necessários, uma vez que o *gateway* pode ser configurado para operar em arquiteturas em malha (*mesh*), ampliando a cobertura e a confiabilidade da comunicação. Em comparação com outras tecnologias que demandam um número significativo de repetidores para garantir a propagação do sinal, o *gateway* de canal duplo se destaca por sua eficiência tanto em termos operacionais quanto de custo.

3.3.2.2 Monitoramento ambiental

O monitoramento e a detecção de incêndios florestais, o acompanhamento da fauna e flora, a identificação de deslizamentos de terra em áreas próximas a rodovias e zonas residenciais, bem como o monitoramento da poluição e a detecção de eventos sísmicos, como terremotos e tsunamis, são exemplos representativos de aplicações de monitoramento ambiental baseadas em IoT (HASSIJA et al., 2019).

Essas aplicações exigem um alto nível de confiabilidade e segurança na comunicação, uma vez que falhas ou vulnerabilidades podem resultar em consequências graves. A geração de falsos positivos, como alarmes indevidos de terremotos, pode acarretar prejuízos logísticos e financeiros consideráveis para o poder público. Por outro lado, a não detecção de eventos críticos pode comprometer vidas humanas e causar perdas significativas de patrimônio (HASSIJA et al., 2019).

Dada a exposição dos sensores a condições extremas, especialmente em cenários de desastres naturais, como incêndios florestais, é comum que esses dispositivos sejam projetados com recursos computacionais limitados, priorizando robustez e eficiência energética. No entanto, mesmo com limitações de *hardware*, esses dispositivos são equipados com sensores de alta precisão e mecanismos que garantem a integridade e a segurança dos dados coletados, assegurando a confiabilidade das informações transmitidas à rede.

Nesse contexto, a adoção de *gateways* de canal duplo com capacidade de comunicação de longo alcance constitui em uma arquitetura não apenas viabiliza a recepção simultânea de transmissões de múltiplos sensores, mas também permite a execução de tarefas computacionais na borda da rede, reduzindo significativamente a latência e aumentando a resiliência do sistema em situações críticas. Ao combinar recepção paralela e processamento local, os *gateways* de canal duplo contribuem para uma resposta mais rápida e eficiente a eventos ambientais extremos, assegurando maior robustez às aplicações de monitoramento distribuído.

3.4 Considerações finais

Com base nos estudos analisados nas Tabelas 4 e 5, é possível identificar quatro principais desafios recorrentes nas redes baseadas em LoRaWAN, especialmente em cenários com alta densidade de dispositivos:

- **Escalabilidade e eficiência em redes densas:** A crescente quantidade de dispositivos IoT impõe desafios de escalabilidade às redes LoRaWAN, exigindo mecanismos mais robustos de acesso ao meio e gerenciamento de recursos.
- **Limitações do protocolo ALOHA:** O uso do protocolo ALOHA puro, sem coordenação entre os dispositivos, resulta em altas taxas de colisões e perda de pacotes, comprometendo a eficiência da comunicação.
- **Ausência de mecanismos eficazes de detecção de canal:** Em implementações de baixo custo, é comum a utilização de *gateways* com hardware idêntico ao dos dispositivos finais. Mesmo com a adoção do mecanismo CAD no *gateway*, limitações de hardware podem comprometer o desempenho da rede. Por exemplo, durante a execução do CAD para envio de um pacote de resposta a um dispositivo, o *gateway* pode perder transmissões simultâneas provenientes de outros nós, uma vez que não possui capacidade de recepção paralela em múltiplos canais ou instâncias.
- **Necessidade de *gateways* multi-canal de baixo custo:** A maioria das soluções de *gateway* de alto desempenho é baseada em hardware especializado, com custo elevado. Em contrapartida, redes menores ou aplicações econômicas frequentemente recorrem ao uso de *gateways* com hardware semelhante ao dos nós sensores, o que limita significativamente a capacidade de recepção simultânea e a robustez da rede.

Esses desafios demandam por soluções mais acessíveis e eficientes, como a proposta apresentada neste trabalho, que visa combinar múltiplos canais de recepção e mecanismos CAD em um *gateway* de baixo custo, contribuindo para o avanço das redes LoRaWAN em diferentes contextos de aplicação.

Capítulo 4

Modelos de *hardware* para IoT

O desenvolvimento de *software* para aplicações em IoT apresenta particularidades, pois depende diretamente tanto da aplicação quanto do *hardware* em que será executado. Na literatura, observa-se que grande parte dos algoritmos propostos para IoT são concebidos e avaliados em ambiente de simulação, sem considerar, em muitos casos, as restrições e características específicas do *hardware* nos cenários analisados.

Por sua vez, o projeto de *hardware* para IoT constitui uma tarefa complexa, uma vez que exige, entre outros fatores, conhecimento especializado acerca dos componentes eletrônicos, além da disponibilidade destes no mercado ou, alternativamente, do desenvolvimento de novos dispositivos de microeletrônica, tais como sensores, conversores, antenas, microcontroladores, microprocessadores e memórias.

No contexto brasileiro, os dispositivos e componentes selecionados para um projeto de *hardware* nem sempre estão disponíveis comercialmente, o que torna necessária a importação. Esse processo, geralmente realizado por meio do programa Importa Fácil¹, implica a incidência de tributos, a concorrência pelo uso de cotas do CNPq, além de envolver significativa burocracia institucional² e longos prazos de espera.

Superado o desafio da aquisição dos componentes, a execução do projeto requer um laboratório especializado, equipado com recursos que vão além de computadores e servidores. Os protótipos são desenvolvidos em ambientes que demandam não apenas a aquisição e manutenção de equipamentos específicos, mas também instalações adequadas para o pleno funcionamento do inventário, além de insumos que variam dos mais simples, como solda, até os mais sofisticados. Dessa forma, o desenvolvimento em *hardware*, além de depender dos componentes e do conhecimento técnico, exige espaço físico, infraestrutura apropriada e recursos financeiros.

¹ CNPq. Importação de bens destinados à pesquisa. Disponível em: <<https://www.gov.br/pt-br/servicos/obter-credenciamento-de-pessoa-fisica-para-importacao-de-bens-destinados-a-pesquisa>>

² Parque Tecnológico da UFRJ. Custo Brasil: Burocracia e importação para ciência. Disponível em: <<https://www.parque.ufrj.br/custo-brasil-burocracia-e-importacao-para-ciencia/>>

Este capítulo está estruturado em duas partes. A primeira apresenta uma visão geral da família mannaWUI de elementos de IoT, destacando um dos dispositivos desenvolvidos durante o período de doutoramento. Nessa seção, também é introduzido o modelo de *hardware* adotado no projeto do mannaWUI para o desenvolvimento de nós sensores, constituindo uma das contribuições deste trabalho. A segunda parte descreve o modelo de *hardware* proposto para a construção de um *gateway* de canal duplo com tecnologia LoRa, voltado ao atendimento de redes LoRaWAN de pequeno e médio porte.

4.1 A família mannaWUI

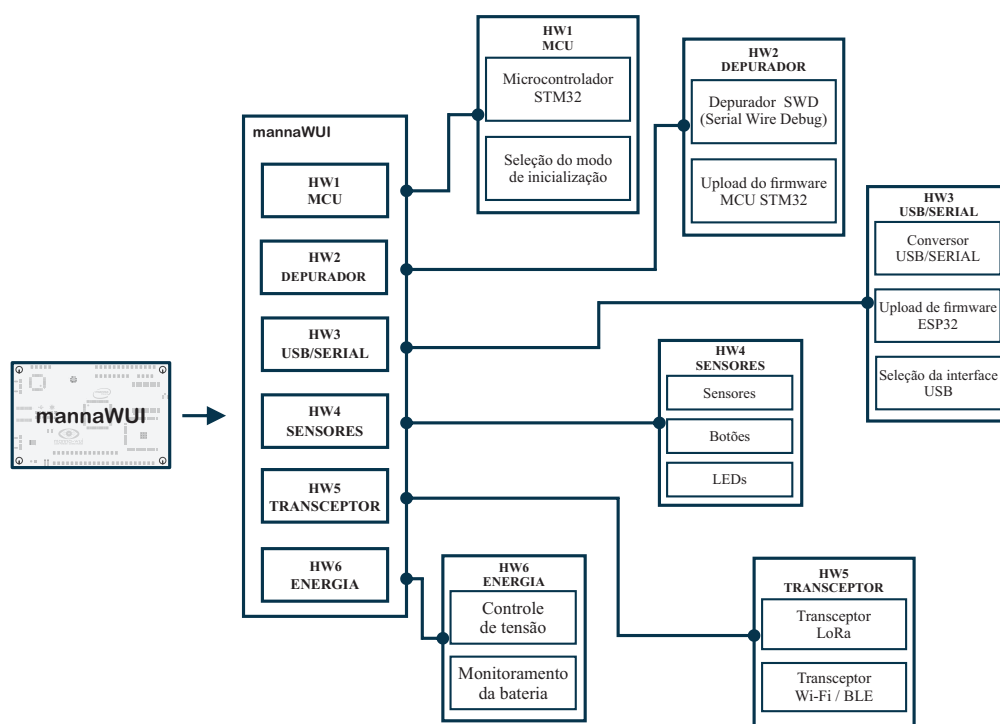
Um dos desafios de pesquisa do Ecossistema Manna consiste em estudar, projetar e confeccionar artefatos voltados à atração, formação e capacitação rápida de pessoas em TICs, bem como disponibilizar *hardware* que possa ser empregado por outros pesquisadores, *startups* e empresas. No âmbito da primeira fase deste doutoramento, foram desenvolvidos elementos de rede para IoT, o mannaWUI e novas versões aprimoradas, cuja concepção inicial havia sido idealizada em trabalhos anteriores.

O modelo de desenvolvimento dos nós é apresentado na Figura 13. Esse modelo tem por objetivo orientar o projeto e a implementação de *hardware* para nó sensores de IoT, oferecendo uma estrutura organizacional que auxilia tanto nas decisões de projeto quanto na organização do trabalho. Assim, contribui para ganhos de desempenho e redução do tempo de desenvolvimento quando comparado ao processo iniciado sem referências prévias ou exemplos práticos. Uma das características centrais desse modelo é a possibilidade de utilização de blocos de Propriedade Intelectual (do inglês, *Intellectual Property* (IP)) em cada um dos módulos (ou subsistemas) definidos.

A família de placas mannaWUI representa uma contribuição significativa, apresentando diferenciais em relação às plataformas disponíveis comercialmente, conforme discutido na Seção 2.1. O projeto destaca-se sobretudo pela sua versatilidade, possibilitando a aplicação em diferentes cenários.

O mannaWUI foi projetado com a capacidade de desligar subsistemas não utilizados, como, por exemplo, o transceptor Wi-Fi, que pode ser completamente desativado em aplicações que empregam exclusivamente a tecnologia LoRa. Dessa forma, evita-se o consumo energético em modo *standby/sleep*, recurso ausente nas principais placas comercializadas (BINE; RUIZ, 2020).

Figura 13 – Modelo de referência para construção de nós sensores.



Fonte: Bine (2021).

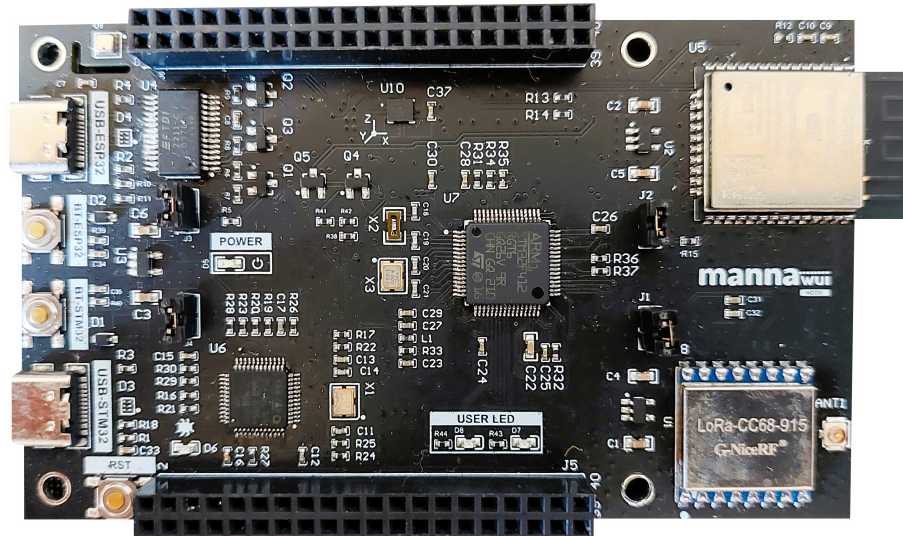
O projeto de *hardware* do mannaWUI suporta a utilização de dois tipos de microcontroladores da STMicroelectronics, um voltado ao baixo consumo e outro de maior desempenho, indicado para aplicações que requerem maior capacidade de processamento. Independentemente do microcontrolador selecionado, o elemento de *hardware* mantém compatibilidade, não havendo necessidade de alteração no *layout* da PCB (*Printed Circuit Boards*).

Outro diferencial do projeto é a inclusão de um sistema de monitoramento da descarga da bateria, implementado por meio da medição do fluxo de corrente e do nível de tensão. Além disso, a plataforma é disponibilizada como *open-hardware*, ampliando sua acessibilidade para a comunidade científica e tecnológica.

Nos anos de 2020, 2021 e 2022, durante a pandemia de COVID-19, diversos componentes originalmente previstos para o projeto do mannaWUI ficaram indisponíveis no mercado ou sofreram aumentos expressivos de preço. Esse cenário tornou necessária a reformulação do projeto, com substituição de componentes e simplificação de funcionalidades. Após sucessivas alterações, foi possível definir um novo conjunto de componentes adequados, dando origem a novas versões mannaWUI. Todos os detalhes referentes a esses projetos encontram-se registrados em documento específico.

Na Figura 14 apresenta a imagem da versão mais recente do mannaWUI, a qual será empregada como nó sensor na execução dos experimentos desenvolvidos neste trabalho.

Figura 14 – Placa mannaWUI.



Fonte: Bine (2021).

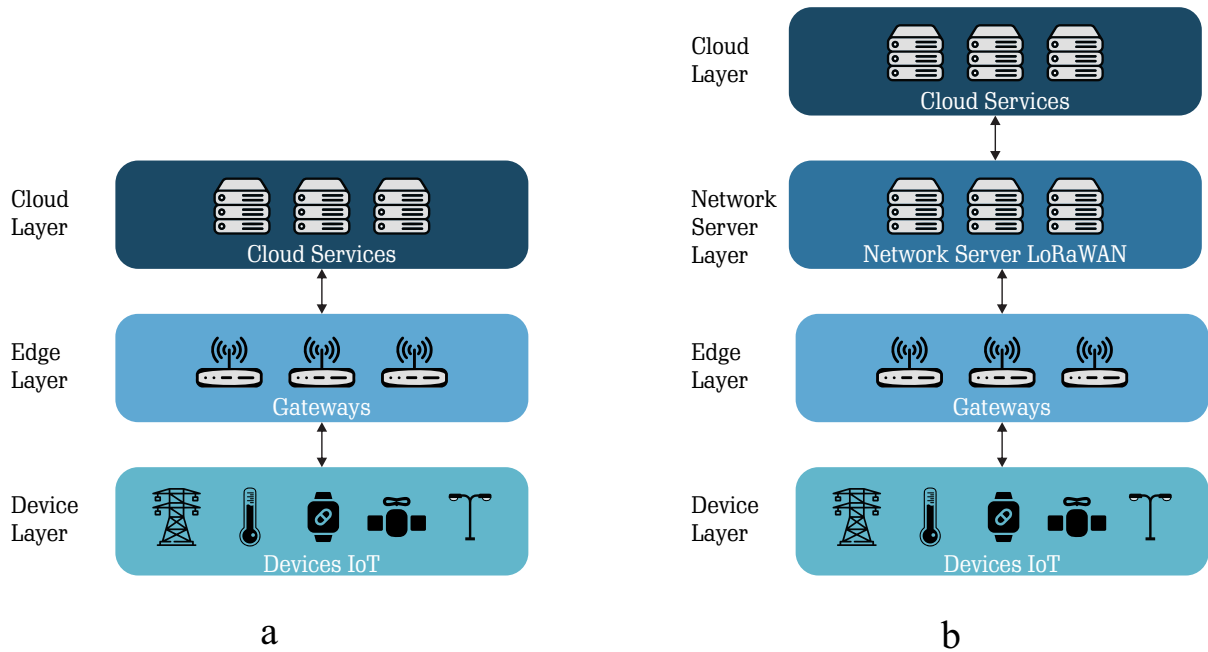
Considerando o trabalho desenvolvido até o momento, bem como os experimentos realizados com a tecnologia LoRa nas placas mannaWUI, e tendo em vista as limitações identificadas no uso do protocolo LoRaWAN, este trabalho propõe o desenvolvimento de um *gateway*, denominado mannaWUIG. Esse dispositivo representa uma contribuição relevante tanto para aplicações em IoT quanto para a academia de IoT do Ecossistema Manna. A próxima seção apresenta o modelo proposto para o projeto de *hardware* do referido *gateway*.

4.2 Um modelo para o desenvolvimento de um *gateway* de canal duplo

Diante dos desafios impostos pelo crescente número de dispositivos conectados em redes IoT e da necessidade de melhorar a eficiência da comunicação, muitos esforços têm sido direcionados ao desenvolvimento de soluções que otimizem o uso do canal e reduzam as colisões (MORABITO et al., 2019; GAILLARD; PHAM, 2023).

A Figura 15(a) apresenta uma arquitetura convencional de comunicação em sistemas IoT. Nela, os dispositivos responsáveis pela coleta de dados e/ou pela execução de ações estão organizados na primeira camada, denominada *Device Layer*.

Figura 15 – A arquitetura convencional IoT (a) e arquitetura de rede LoRaWAN (b).



Fonte: Própria (2025).

Para que os dispositivos realizem a troca de dados utilizando diferentes tipos de tecnologias, faz-se necessária a presença de um *gateway*, que atua como camada de borda, sendo responsável também pela transmissão das informações para serviços em nuvem, onde os dados são processados.

Na arquitetura LoRaWAN, acrescenta-se uma camada adicional, conhecida como servidor de rede (Figura 15(b)). Essa camada tem a função de gerenciar toda a rede LoRaWAN, verificando se o dispositivo transmissor está devidamente registrado e encaminhando os dados para as camadas superiores. Além disso, é responsável por ajustar parâmetros de transmissão dos dispositivos conectados à rede, como, por exemplo, o fator de espalhamento.

No design de um *gateway* de canal único, o transceptor encarregado das transferências de dados é capaz de se comunicar com apenas um dispositivo por vez. Esse método de transmissão mostra-se limitado para aplicações em ambientes com redes LoRa coexistentes, nas quais há a necessidade de recepção de dados provenientes de múltiplos dispositivos de forma simultânea.

Os *gateways* baseados em LoRaWAN, que utilizam amplamente o protocolo ALOHA em conjunto com mecanismos de CAD, têm sido explorados como forma de aumentar a escalabilidade dessas redes. Ademais, métodos de agendamento de recursos, como o TDMA, vêm sendo empregados para organizar o tráfego e evitar interferências, contribuindo para um desempenho mais eficiente, algo que se torna mais desafiador em

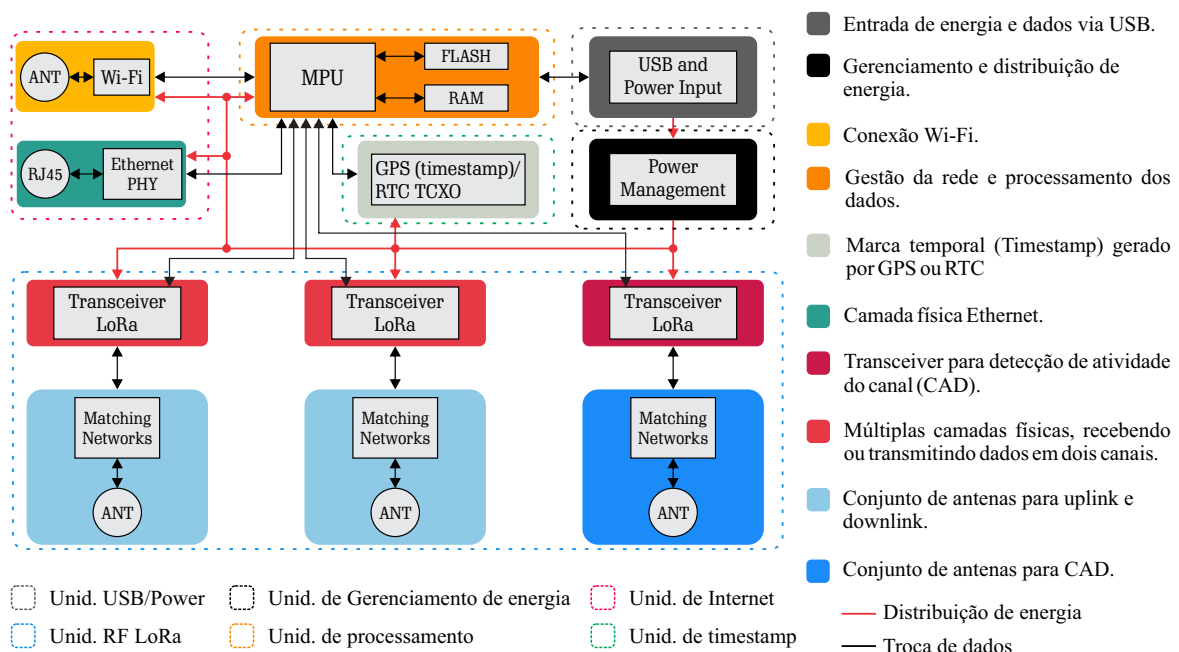
cenários com redes coexistentes (VANGELISTA; CALVAGNO, 2024).

Uma solução promissora para enfrentar essas limitações é a implementação de *gateways* de canal duplo, que permitem a comunicação simultânea em dois canais distintos, por meio do uso de transceptores LoRa. Esses dispositivos representam uma alternativa de menor custo em comparação aos *gateways* de 8 canais, cujo elevado preço constitui uma barreira para redes IoT de pequeno e médio porte.

Embora os *gateways* de canal duplo apresentem restrições em cenários de alta densidade de dispositivos, eles se mostram eficientes no suporte a múltiplos nós em redes de menor escala. Adicionalmente, a integração do mecanismo CAD possibilita a verificação da ocupação do canal antes da transmissão de dados de retorno aos dispositivos, reduzindo colisões e otimizando o uso do espectro.

A proposta do *gateway* de canal duplo é ilustrada na Figura 16, sendo implementada na *Edge Layer* (Figura 15). Esses *gateways* são projetados para oferecer conexões simultâneas a múltiplos dispositivos, além de dispor da capacidade de realizar o pré-processamento dos dados, o que contribui para o aumento da eficiência da rede.

Figura 16 – Modelo para um *gateway* de canal duplo.



Fonte: Própria (2025).

Diversos trabalhos, como os de Yascaribay et al. (2022), Mukhia, Jayarathna e Lertsinsrubtavee (2023), Nugraha, Ashadi e Efendi (2024), empregam *gateways* convencionais, nos quais o transceptor opera de forma limitada, atendendo os dispositivos de maneira sequencial. Em cenários com elevada densidade de nós, essa abordagem tradicional pode

comprometer a eficiência da rede, evidenciando a necessidade de soluções mais avançadas, como o *gateway* de canal duplo, que amplia a capacidade de comunicação e possibilita o gerenciamento de múltiplos dispositivos.

Neste contexto, observamos uma lacuna quando se trata da implementação de soluções de baixo custo para transmissão de longa distância com tecnologia LoRa, especialmente em redes de pequeno e médio porte. O modelo visto na Figura 16, propõe na unidade de RF, a utilização de três transceptores de tecnologia LoRa (como exemplo: SX127x ou SX126x), sendo dois para operar como receptores em canais distintos e um específico para realizar o mecanismo de CAD e transmissão.

Ao propor um conjunto de antenas e um transceptor dedicado exclusivamente para o processo de detecção de atividade no canal, assegura-se que os dois canais principais permaneçam disponíveis para a recepção de dados. Simultaneamente, o canal dedicado ao CAD monitora a ocupação do espectro, assegurando que, quando for necessário transmitir de volta para um dispositivo da rede (por exemplo, durante a janela de recepção dos dispositivos de classe A), a transmissão ocorra sem interferências. Isso evita colisões e garante que o *gateway* não perca informações de outros dispositivos que possam estar transmitindo no momento em que o retorno seria enviado, preservando assim a integridade da comunicação e o desempenho da rede.

A unidade de processamento apresentada na Figura 16, é concebida por um microprocessador, memória FLASH e RAM, com o propósito de viabilizar o processamento, gestão da rede e armazenamento de dados, assim o microprocessador é relativamente mais robusto que os microcontroladores encontrados na grande maioria dos dispositivos IoT.

O envio de dados para terceira ou quarta camada (Figura 15 (b)) é habilitado por meio da unidade de *Ethernet*, vista na Figura 16, provendo conexão com a Internet podendo ser por tecnologias sem fio (Wi-Fi), 3G/4G/5G (telefonia móvel) ou via cabo (*Ethernet*).

A unidade *timestamp* gera uma marca temporal precisa usando GPS ou um oscilador de cristal com compensação de temperatura (RTC+TCXO), garantindo a sincronização adequada, se necessário, entre os dispositivos, mesmo sem conexão com a Internet.

A unidade de gerenciamento de energia é responsável por regular a tensão para as demais unidades, garantindo uma gestão eficiente do consumo energético do *gateway*. A entrada de energia pode ser fornecida por meio de um conector USB, um plug de alimentação dedicado ou via PoE (*Power over Ethernet*).

Cabe ressaltar que a unidade de gestão da rede e processamento de dados, ao disponibilizar serviços de nuvem na borda, possibilita atender aos requisitos de latência de aplicativos, melhorar a escalabilidade e a eficiência energética (CHEN; LIN; LIU, 2018). Além disso, essa abordagem oferece a vantagem de fornecer serviços escaláveis para

aplicações IoT tolerantes a atrasos, que utilizam tecnologia LoRa.

Com o intuito de garantir que as decisões de projeto e os requisitos técnicos do sistema estejam alinhadas, a Tabela 6 apresenta uma consolidação dos principais requisitos considerados no desenvolvimento do modelo proposto ilustrado na Figura 16, organizados nas categorias de *hardware* e *software*. Cada requisito está diretamente associado às necessidades operacionais do sistema, considerando aspectos como custo, capacidade de recepção simultânea, escalabilidade e robustez em ambientes com interferência de redes coexistentes.

Tabela 6 – Requisitos do projeto.

Categoria	Requisitos	Justificativa	Elemento proposto para o modelo
<i>Hardware</i>	Vários canais de recepção simultâneos.	Para reduzir colisões e aumentar a escalabilidade em ambientes densos networks.	Múltiplos transceptores LoRa (SX126x).
<i>Hardware</i>	Capacidade de executar CAD sem comprometer a recepção de pacotes.	Redução de colisões durante transmissões de <i>gateway</i> para nó sensor.	Transceptor LoRa dedicado a CAD (SX126x).
<i>Hardware</i>	Sincronização de tempo sem acesso à Internet	Para habilitar a ordenação de pacotes e o controle da janela de recepção.	GPS (MAX-M10), RTC + TCXO (DS3231).
<i>Hardware</i>	Unidade de processamento e gerenciamento de rede.	Processamento local para gerenciamento da rede e tomada de decisão.	ESP32-S3 (MPU), com suporte para TensorFlow Lite e ESP-DL.
<i>Hardware</i>	Memória expansível.	Suporte para expansão do armazenamento local de dados.	Memória FLASH e RAM externa.
<i>Hardware</i>	Interface de rede com e sem fio.	Flexibilidade para conectividade via Ethernet ou Wi-Fi.	Ethernet PHY e Wi-Fi.
<i>Software</i>	Compatibilidade com sistemas operacionais de tempo real.	Gerenciamento eficiente de tarefas concorrentes, controle de interrupções e priorização de processos no <i>gateway</i> .	Microcontrolador ESP32-S3, com suporte nativo ao FreeRTOS de baixo custo.
<i>Software</i>	Suporte para <i>frameworks</i> de IA embarcada.	Execução local de modelos de IA leves para análise em tempo real e tomada de decisão autônoma.	Unidade de aceleração vetorial do ESP32-S3 para suporte a bibliotecas de IA (TensorFlow Lite).

Os principais requisitos destacados incluem a capacidade de recepção simultânea em múltiplos canais e a execução do CAD sem comprometer a escuta contínua dos transceptores dedicados à recepção. Para atender a esses requisitos, o modelo proposto adota múltiplos transceptores da família SX126x, sendo dois alocados para recepção em canais distintos e um terceiro destinado exclusivamente à realização do CAD.

Para assegurar a marcação temporal precisa e o agendamento de mensagens, propõe-se o uso de um módulo GPS em conjunto com RTC+TCXO, especialmente em situações em que a conexão com a Internet esteja indisponível. A unidade de processamento deve possuir capacidade suficiente para gerenciar a rede de forma eficiente e executar algoritmos leves de inteligência artificial, atendendo aos requisitos de baixo custo.

Em suma, o *gateway* de canal duplo proposto oferece uma solução econômica para redes IoT de pequeno e médio porte, abordando diretamente as limitações de escalabilidade e custo das soluções convencionais, como os *gateways* de grande porte, bem como as restrições dos *gateways* de canal único. Com a integração de transceptores dedicados à detecção de atividade no canal e a capacidade de operar simultaneamente em múltiplos canais, o modelo otimiza o uso do espectro, minimiza colisões e melhora a eficiência geral da rede.

4.3 Considerações finais

A IoT, em um primeiro momento, pode parecer simples de compreender, mas seu desenvolvimento e implementação estão longe de ser triviais. O número de elementos de rede pode crescer exponencialmente, o volume de dados trocados tende a aumentar de forma expressiva, novas tecnologias podem surgir, as características do ambiente de operação podem se modificar, os esquemas de computação e comunicação podem ser distribuídos entre borda, neblina e nuvem, a recarga ou substituição das fontes de energia pode se tornar inviável, elementos podem sair de serviço por diferentes razões, os nós podem apresentar mobilidade, e as questões de segurança ainda permanecem em aberto. Independentemente da aplicação ou da arquitetura de IoT considerada, para assegurar bom funcionamento e desempenho é imprescindível um projeto consistente de *hardware* e *software*. Neste capítulo, foi apresentado o modelo de referência para auxiliar projetistas no desenvolvimento de elementos de *hardware* para IoT, bem como o modelo original para o projeto do *gateway* de canal duplo. A contribuição da proposição desses modelos reside na redução do tempo despendido por outros projetistas ao explorar diferentes plataformas, no aumento da segurança quanto à escolha de componentes previamente testados e na maior clareza no momento de definir o esquemático do *hardware*.

Capítulo 5

mannaWUIG: Decisões de projeto

Neste capítulo são apresentadas as decisões de projeto do *smart gateway* de canal duplo, chamado de mannaWUIG. A composição do *hardware* está organizada em subsistemas (Seção 5.1). A unidade de energia é detalhada na Seção 5.1.2, apresentando os conectores de entrada de energia e como é realizada a regulação de tensão para as demais unidades. Na sequência é apresentado a unidade USB, responsável por permitir atualização do *firmware* (Seção 5.1.3). O microcontrolador utilizado para *gateway* é apresentado na unidade de processamento (Seção 5.1.1). O projeto RF utilizando os transceptores LoRa são apresentados na Seção 5.1.4. Por fim, é apresentado o projeto para a camada física Ethernet (Seção 5.1.4.2).

5.1 Projeto de hardware

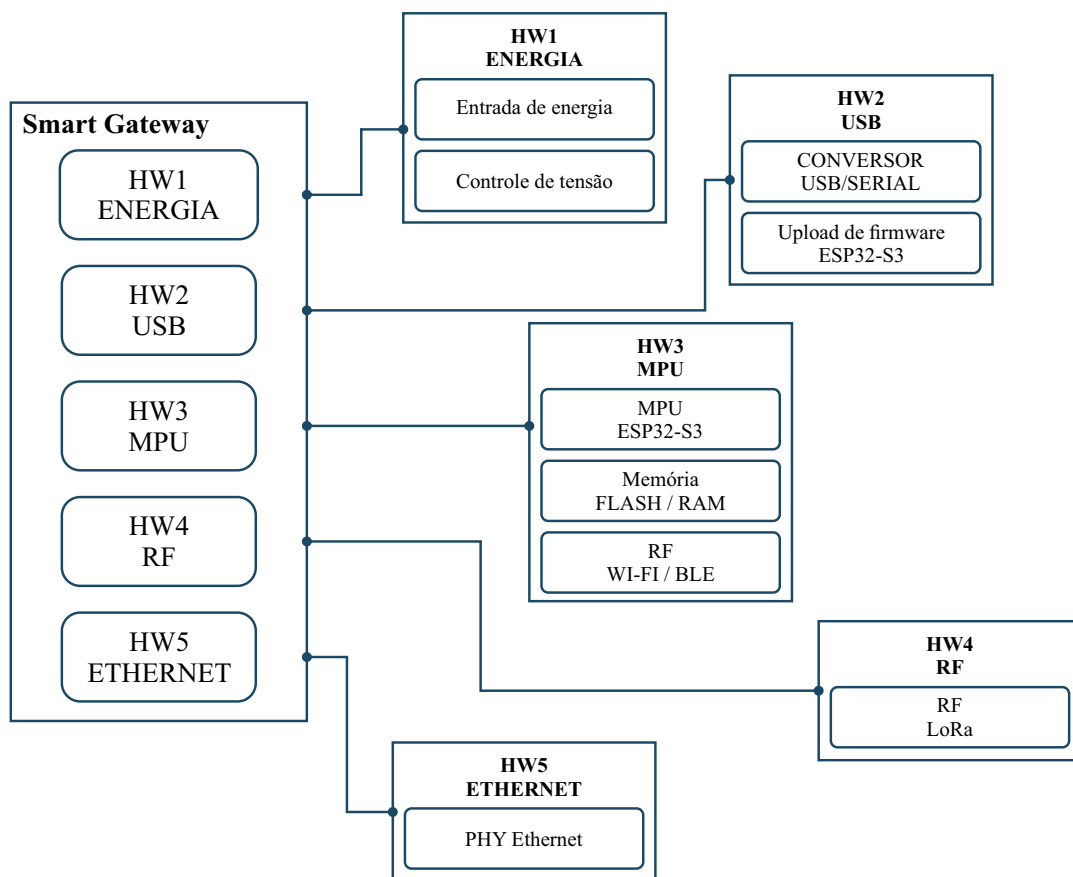
A composição do *hardware* proposto para o *smart gateway* foi organizado em 5 subsistemas apresentados na Figura 17. Para elaboração da proposta foi considerado componentes que possam ser adquiridos com maior facilidade, considerando a necessidade de importação, também seus valores para obter um *hardware* que equilibre desempenho e baixo custo. Na sequência é apresenta um resumo da funcionalidade de cada subsistema.

- **HW1 - Unidade de energia:** Esta unidade apresenta o projeto do circuito de controle de tensão para os subsistemas subsequentes, apresentando as entradas de energia e nível de tensão máximo de alimentação.
- **HW2 - Unidade USB:** A presente unidade possibilita realizar atualização do *firmware* do *gateway*, assim como visualizar *logs* gerados pelo sistema para monitoramento do funcionamento durante o desenvolvimento.
- **HW3 - Unidade de processamento:** Com capacidade de executar algoritmos leves de IA, essa unidade é responsável por todo processamento do *gateway*, bem

como realizar o controle dos transceptores e executar serviços de servidor de rede, quando não utilizado um externo.

- **HW4 - Unidade de RF LoRa:** Essa unidade tem como funcionalidade transmitir e receber dados utilizando a tecnologia LoRa.
- **HW5 - Unidade Ethernet:** Para se conectar a Internet, essa unidade disponibiliza uma camada física Ethernet para prover conectividade com serviços em nuvem.

Figura 17 – Composição do *hardware*.



Fonte: Própria (2025).

5.1.1 Unidade de processamento do mannaWUIG

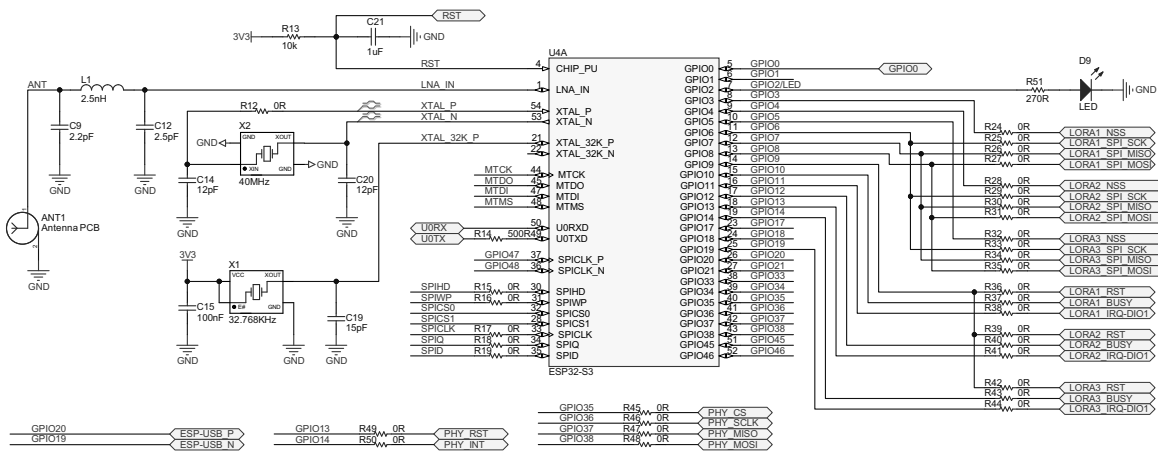
A unidade de processamento é responsável por todo funcionamento do dispositivo. O MCU escolhido para essa função foi o ESP32-S3FH4R2 por ter suporte adicional para instruções vetoriais, fornecendo aceleração para algoritmos leves de inteligência artificial na execução de redes neurais e processamento de sinais.

O ESP32-S3FH4R2 é um microcontrolador baseado no núcleo Xtensa LX7 *dual-core*, capaz de operar a frequências de até 240 MHz. O dispositivo integra conectividade Wi-Fi de 2,4 GHz (802.11 b/g/n) e *Bluetooth* 5 Low Energy (LE). Além de dispor de memória

flash e RAM internas, o ESP32-S3FH4R2 também suporta a utilização de memórias externas do tipo *flash* e RAM por meio de interface SPI de alta velocidade. Adicionalmente, o microcontrolador oferece um conjunto abrangente de periféricos, incluindo interfaces SPI, I²C, I²S, USART, USB, PWM, ADC, entre outros.

A Figura 18 mostra os dois cristais osciladores externos que permitem ter um sinal de *clock* mais preciso e menos suscetível a deformação do sinal, devido à variação de temperatura. Para gerar o *clock* principal foi utilizado um cristal de 40 MHz e um segundo de 32.768 KHz para o RTC (*Real Time Clock*). O circuito PLL (*Phase Locked Loop*) interno ao *chip* é responsável por permitir atingir a frequência de operação de 240 MHz, em momentos de maior processamento.

Figura 18 – Circuito projetado para o ESP32-S3FH4R2.

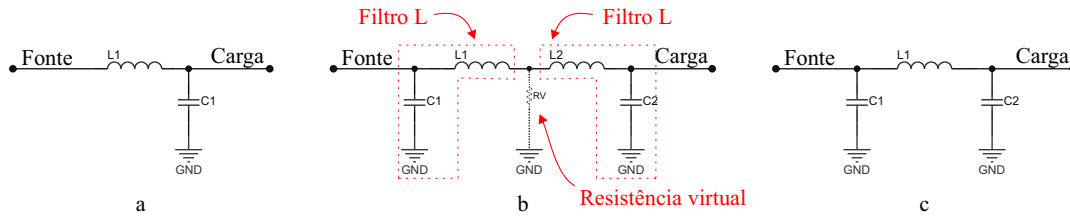


Fonte: Própria (2025).

Como mencionado, o ESP32-S3FH4R2 integra conectividade Wi-Fi/BLE e sua saída de sinal de RF possui uma impedância característica de $(30 + 10j)\Omega$. Segundo a folha de dados do *chip*, o fabricante recomenda a utilização de uma rede de correspondência do tipo π para fazer o casamento de impedância com uma antena de 50Ω .

O cálculo para rede de correspondência do tipo π tem como ponto de partida um filtro do tipo L, representada na Figura 19 (a). A rede π (Figura 19 (c)) considera que dois circuitos do tipo L estejam conectados inversamente Figura 19 (b). O cálculo para encontrar o fator Q segue o mesmo princípio para um filtro do tipo L, sendo apresentado na Equação 5.1.

Figura 19 – Representação de (a) um filtro L, (b) associação de dois filtros e (c) resultado de uma rede de correspondência do tipo π .



Fonte: Própria (2025).

$$Q = \sqrt{\frac{R_L}{R_S} - 1} \quad (5.1)$$

Para encontrar os valores do indutor e capacitor é realizado o cálculo a partir da associação dos dois filtros L (Figura 19 (b)), é considerado um resistor virtual (R_V) de baixa resistência entre os indutores, a reatância capacitiva em paralelo é calculada pela Equação 5.2 e a reatância do indutor em série é calculada pela Equação 5.3. Com os resultados calcula-se o valor do indutor e capacitor.

$$X_{paralelo} = \frac{R_{alto}}{Q} \quad (5.2)$$

$$X_{serie} = QR_{baixo} \quad (5.3)$$

Onde, $X_{paralelo}$ é a reatância capacitiva; X_{serie} é a reatância indutiva; R_{alto} é a resistência de maior valor entre a R_V e a resistência do lado que está realizando o cálculo; R_{baixo} é a resistência de menor valor entre a R_V e a resistência do lado que está realizando o cálculo.

Para o cálculo é considerado uma antena de 50Ω . Este processo é realizado para os dois lados e ao final é associado os dois indutores. Considerando que a impedância de saída é do *chip* é complexa, aconselha-se utilizar o Smith Chart¹ com um simulador de RF como Keysight ADS² para combinar a carga da antena e saída do sinal. Ajustes nos valores calculados podem ser necessários devido à influência da PCB sobre o circuito, assim é aconselhável medir a potência de saída para realizar ajustes.

O ESP32-S3FH4R2 possui quatro pinos chamados de *strapping pins*, sendo eles GPIO0, GPIO3, GPIO45 e GPIO46. Esses pinos são responsáveis por definir o modo de

¹ Smith Chart é considerado uma ferramenta para visualizar a impedância de uma linha de transmissão e sistema de antena em função da frequência, sendo amplamente utilizado para realizar casamento de impedâncias em sistemas de RF simulados e em medições em tempo real utilizando um analisador de rede vetorial (VNA).

² Keysight ADS fornece software de simulação e projeto de circuitos. Mais informações em: <https://www.keysight.com>

operação do MCU sendo verificados sempre que o MCU inicializa ou reinicia. A Tabela 7, mostra as possibilidades de configurações dos pinos e que alguns possuem *pull-up/pull-down* interno ao *chip* para definir seu estado padrão.

Tabela 7 – Sumarização dos pinos verificados na inicialização do ESP32-S3FH4R2 (*Strapping Pins*).

VDD SPI Voltage			
Pin	Default	3.3V	1.8V
GPIO45	Pull-down	0	1
Boot Mode			
Pin	Default	SPI Boot	Download Boot
GPIO0	Pull-up	1	0
GPIO46	Pull-down	Don't care	0
JTAG Signal Selection			
Pin	Default	JTAG signal from pins	JTAG signal from USB
GPIO3	-	0	1

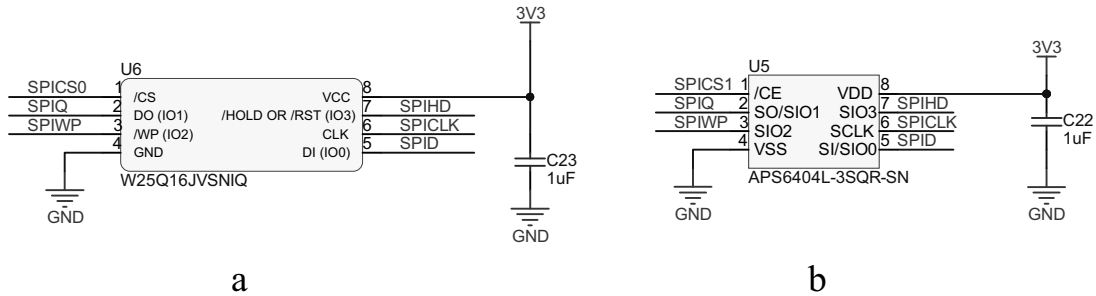
Fonte: Adaptado de Espressif Systems (2022a).

No projeto foram utilizadas memórias externas que operam com nível de tensão de 3.3V, assim não foi utilizado o regulador de tensão interno do MCU para se obter tensão de 1.8V, logo o pino GPIO45 sempre inicializa com nível lógico baixo. O modo de *boot* ficou configurado como padrão para sempre inicializar no modo *SPI Boot*, um resistor de *pull-up* foi adicionado ao pino GPIO0 externamente para garantir nível lógico alto. A alteração do modo de inicialização é feito automaticamente para "*Download Boot*" conforme visto na Seção 5.1.3.

O pino GPIO3 é responsável por direcionar os sinais do JTAG para o periférico USB interno ou deixar disponível os sinais diretamente nos pinos do *chip*. Assim foi adicionado um resistor de *pull-up* para direcionar os sinais para o periférico USB interno, deste modo é possível realizar depuração do MCU sem necessidade de um depurador externo ao *chip*.

O ESP32-S3FH4R2 tem suporte para memória externa *Flash* e RAM via comunicação SPI, podendo ser mapeados no espaço de memória de instruções da CPU e no espaço de memória de dados, também fornece criptografia de hardware baseada em XTS-AES para proteger os dados do usuário em ambas memórias externas. A memória *flash* utilizada foi a W25Q16JVSNIQ (fabricado pela Winbond³), como mostrada na Figura 20 (a), com capacidade de 2 MB, podendo ser posteriormente substituído para valores maiores como 8 MB ou 16 MB.

³ A Winbond é uma fabricante chinesa que produz semicondutores com foco principalmente em DRAMs, SRAMs e Flash. Mais informações em: <https://www.winbond.com>

Figura 20 – Circuito para memória *flash* e RAM.

Fonte: Própria (2025).

Segundo a Espressif Systems (2022a), a família do MCU ESP32 suportam várias marcas de *chips* de memória RAM, mas o *framework* fornecido pelo fabricante para desenvolvimento do *software*, o ESP-IDF⁴, somente fornece suporte para memórias ESP-PSRAM32 e ESP-PSRAM64 também fabricadas pela Espressif. Contudo, existe uma dificuldade de se encontrar essas memórias a venda no varejo no mercado nacional e internacional, assim optou-se por utilizar o *chip* APS6404L-3SQR (fabricado pela AP Memory⁵), como mostrado na Figura 20 (b), com capacidade de 8 MB, como o *chip* tem as mesmas características do ESP-PSRAM64, possui uma maior facilidade de integração com o ESP-IDF.

O circuito requer componentes externos mínimos, assim os capacitores C23 e C22 (Figura 20) foram adicionados para desacoplamento. Os resistores R15, 16, R17, R18 e R19 (Figura 18) são adicionados nos pinos de comunicação SPI com objetivo principal de reduzir a interferência de RF, o valor de resistência pode ser alterado para diminuir a corrente de condução, ajustar o tempo e diminuir ruídos externos. Quando o circuito dispõe de blindagem metálica aterrada, a utilização desses resistores torna-se opcional, embora ainda seja recomendada (Espressif Systems, 2022b).

5.1.2 Unidade de energia do mannaWUIG

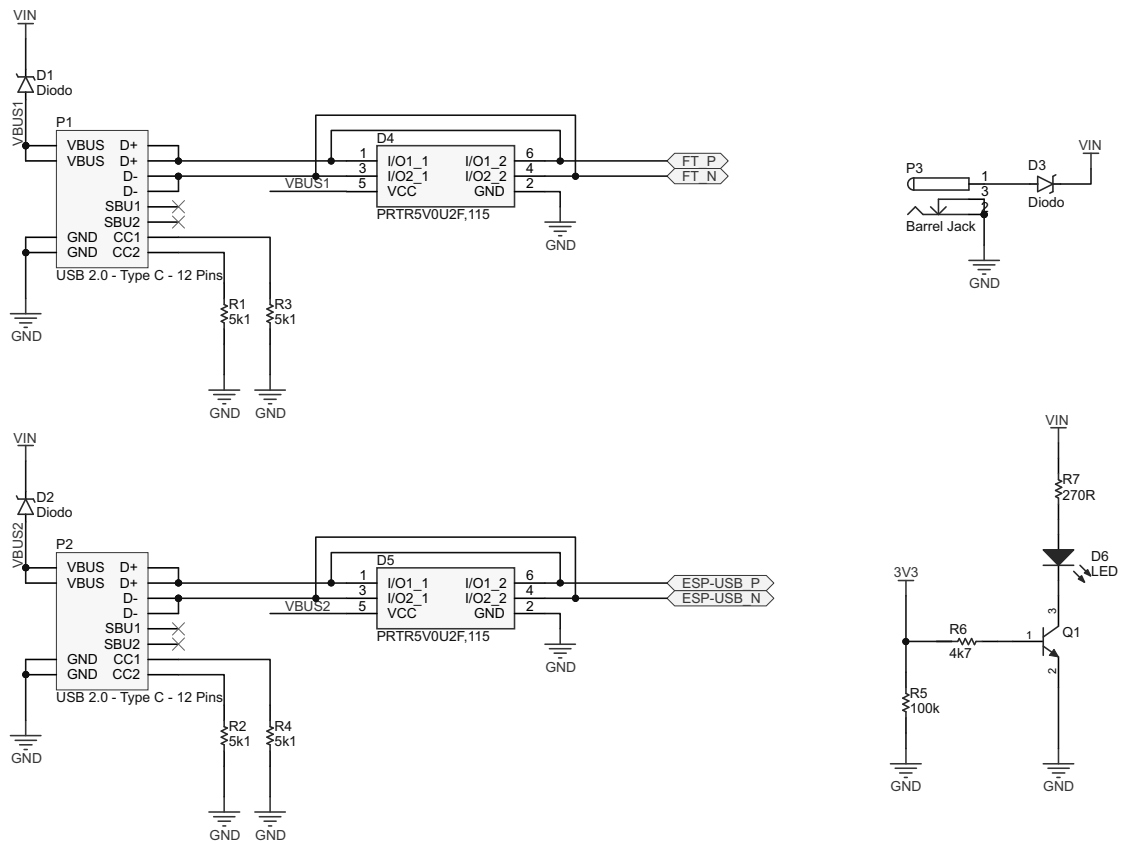
Esta unidade é responsável por realizar o controle da entrada de energia do *gateway*. O dispositivo possui dois conectores USB tipo C (P1 e P2) como apresentado na Figura 21, que ao ser ligados fornecem energia para todos os sistemas do dispositivo. A placa também possui um conector para plug P4, ficando disponível sua alimentação por meio de uma fonte de tensão de 5V.

⁴ O ESP-IDF é um framework desenvolvido para auxiliar no desenvolvimento de aplicações que utilizam o chips ESP32, ESP32-S2, ESP32-S3, ESP32-C2 e ESP32-C3. Mais informações disponíveis em: <https://github.com/espressif/esp-idf>

⁵ AP memory fabricante de memórias PSRAM e DRAM de baixo consumo, com foco no mercado de IoT. Mais informações em: www.apmemory.com

Considerando a possibilidade de utilização simultânea das duas portas USB e o conector P4, foram colocados os diodos D1, D2 e D3 para evitar possíveis curtos-circuitos. A placa também contempla um LED (D6) fornecendo um *feedback* visual de que o dispositivo está energizado.

Figura 21 – Conectores para entrada de energia.

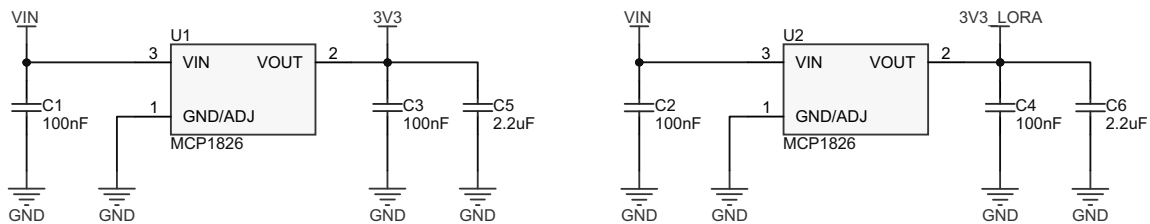


Fonte: Própria (2025).

Para manter o nível de tensão adequado, é utilizado o MCP1826 (fabricado pela MICROCHIP⁶), um regulador de tensão que mantém o nível de saída fixo em 3,3V e pode fornecer uma corrente de saída de até 1A, apresentado na Figura 22. O MCP1826 foi escolhido devido sua baixa corrente de inatividade chegando a 120 μ A e baixa tensão de *dropout* sendo tipicamente de 250mV. Os Capacitores C1, C2, C3, C4, C5 e C6 são utilizados como filtros para manter um nível de tensão estável.

⁶ Microchip Technology Inc. é líder em soluções de microcontrolador, sinal misto, analógico e Flash-IP. Mais informações em: <https://www.microchip.com/>

Figura 22 – Reguladores de tensão.

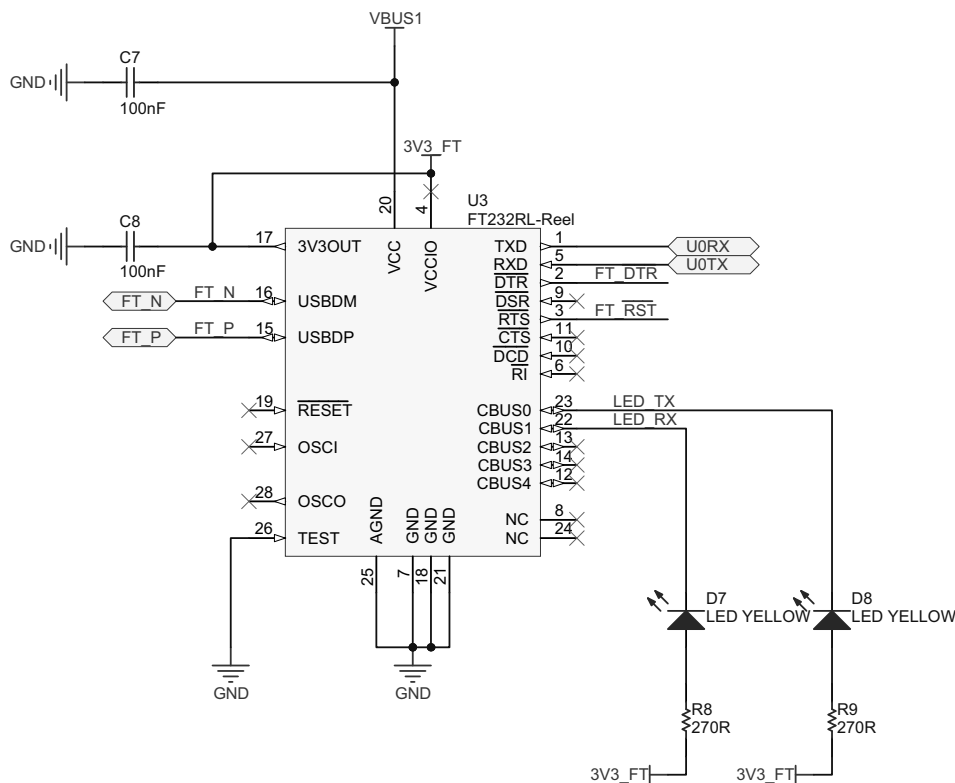


Fonte: Própria (2025).

5.1.3 Unidade USB do *mannaWUIG*

Um conversor USB-SERIAL é utilizado com o propósito de visualizar *logs* do dispositivo e permitir realizar *upload* de atualizações de *firmware* para a unidade de processamento. O FT232RL (fabricado pela FTDI Chip⁷) foi utilizado para realizar a conversão dos dados, seu circuito necessita de poucos componentes externos como visto na Figura 23.

Figura 23 – Conversor USB-SERIAL.



Fonte: Própria (2025).

⁷ FTDI - Future Technology Devices International desenvolve soluções inovadoras em chips com foco em conversores USB/Serrial. Mais informações em: <https://www.ftdichip.com/>

O FT232RL foi escolhido por ser um conversor de *chip* único que realiza a conversão de dados do padrão USB para o padrão RS-232 serial assíncrona. O protocolo USB é tratado inteiramente no *chip*. Tem a capacidade de transferir 256 bytes e receber 128 bytes. Sua tecnologia permite utilizar taxas de transferência de dados entre 300 *baud* até 3 *Mbaud*. Outro ponto de decisão de escolha para adicionar o conversor no *design* foi por possuir geração própria de relógio (*clock*) interno ao *chip*, não havendo a necessidade de adicionar um cristal externo.

No *design* também foram adicionados os LEDs D7 e D8 para obter um *feedback* visual de quando está ocorrendo transferência de dados entre os pinos RX e TX respectivamente. Os capacitores C7 e C8 são responsáveis por suprir picos de tensão provenientes da conexão à porta USB.

Para possibilitar realizar o *upload* do *firmware* para unidade de processamento é necessário colocar o MCU no modo de *download* durante sua inicialização. Esse processo é realizado por meio da leitura do nível de tensão de um pino específico do microcontrolador e esse processo sempre é realizado toda vez que o dispositivo é ligado ou reiniciado.

Os pinos verificados no momento da inicialização referente ao modo de *boot* são o GPIO0 e GPIO46, o estado padrão dos pinos é apresentado na Tabela 8, os demais pinos verificados na inicialização são detalhados na Seção 5.1.1. São dois modos de inicialização possíveis: "SPI *Boot*" e "Download *Boot*". Cada pino é conectado a um *pull-up/pull-down* interno, caso o pino não esteja sendo usado externamente, deste modo o *chip* inicializa com níveis padrão definidos pelo fabricante.

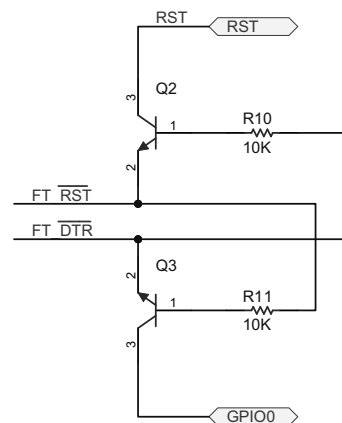
Tabela 8 – Booting Mode ESP32-S3.

Pin	Default	SPI Boot	Download Boot
GPIO0	Pull-up	1	0
GPIO46	Pull-down	Don't care	0

Fonte: Espressif Systems (2022a).

Analisando a Tabela 8, o MCU sempre inicializa no modo "SPI *Boot*", executando o código do *firmware* gravado na memória *flash*. Para se estabelecer o modo de "Download *Boot*", para enviar um novo *firmware*, os pinos GPIO0 e GPIO46 devem estar em nível baixo "0". Assim, foi feito um circuito, apresentado na Figura 24, para que o nível lógico dos pinos se altere automaticamente no momento de transferência do novo *firmware*, alterando o modo de operação do MCU para *Download Boot*.

Figura 24 – Circuito para entrar no modo "download boot" automaticamente.



Fonte: Própria (2025).

O software que realiza *upload* do *firmware* utiliza dois pinos de controle do padrão RS-232, sendo eles o pino RST (*Reset*) e DTR (*Data Terminal Ready*). O pino RST faz com que o MCU seja reiniciado para poder ser alterado para o modo *Download Boot*. No padrão RS-232, o pino DTR é usado para informar que está pronto para receber, iniciar ou continuar uma comunicação de dados, mas neste caso ele é usado para alterar o estado do pino GPIO0. Assim, após a alteração do estado do GPIO0 o MCU inicializa no modo *Download Boot* e os dados são transferidos pelos pinos RX e TX.

O circuito projetado (Figura 24) utiliza dois transistores do tipo NPN como chave. Os transistores recebem os níveis dos pinos RST e DTR para alterar o estado do nível dos pinos no MCU. A Tabela 9 apresenta os estados possíveis nos pinos DTR e RST e seus respectivos resultados nos pinos do MCU.

Tabela 9 – Tabela verdade do circuito de autoprogramação.

DTR	RST	MCU/RST	MCU/GPIO0
0	0	1	1
0	1	1	0
1	0	0	1
1	1	1	1

Fonte: Própria (2025).

O software utilizado para realizar esse processo de *upload* do *firmware* para o MCU está disponível no próprio site do fabricante do *chip*, é importante ressaltar que a utilização de outro software para *upload* do *firmware* pode não utilizar os pinos de controle DTR e RST mencionados, neste caso é necessário alterar o estado dos pinos manualmente.

5.1.4 Unidade de comunicação do mannaWUIG

5.1.4.1 Unidade LoRa

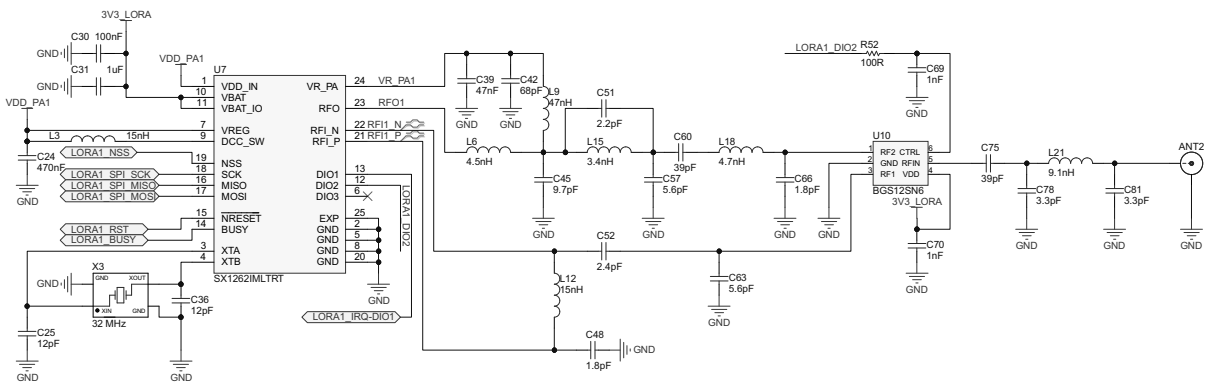
O projeto tem como objetivo aprimorar o desempenho de redes baseadas na tecnologia LoRa, ampliando sua capacidade de atendimento simultâneo a um maior número de dispositivos. Para isso, propõe-se a utilização de dois transceptores SX1262, fabricados pela Semtech, destinados a prover conectividade em dois canais distintos, possibilitando a troca de dados com múltiplos dispositivos de forma concorrente. Adicionalmente, um terceiro transceptor é empregado com a função de executar o mecanismo de CAD e realizar a transmissão de respostas aos dispositivos, garantindo que os dois canais principais permaneçam continuamente em modo de recepção.

O SX1262 é um transceptor LoRa que opera na frequência sub-GHz, projetado para atender especificações do protocolo LoRaWAN (ver Seção 2.4) e suportar protocolos proprietários. O *chip* também possui capacidade de realizar comunicação utilizado a modulação (G)FSK.

O circuito para o transceptor SX1262 é apresentado na Figura 25, no projeto foi considerado a faixa de frequência de operação de 902 a 928 MHz (frequência atribuída pela ANATEL para utilização no Brasil, ver Seção 2.3), o circuito de filtros e cálculos de impedância foram realizados conforme as especificações e orientações realizadas pelo fabricante.

Os cálculos realizados nesta seção tem como base um valor inicial para os componentes, ao realizar a montagem na PCB, podem surgir incompatibilidade de impedância devido ao processo de fabricação da mesma, sendo necessário realizar ajustes de valores dos componentes para se obter o melhor desempenho e potência de saída desejada.

Figura 25 – Circuito proposto para o transceptor SX1262.



Fonte: Própria (2025).

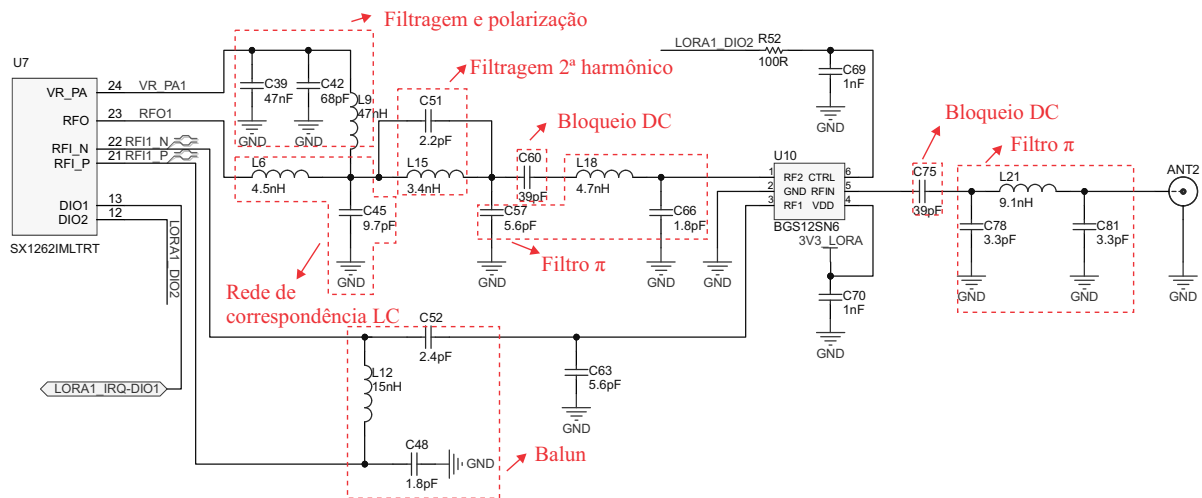
O SX1262 possui internamente um circuito PA (*Power Amplifier*) para se obter

uma potência de transmissão de até +22 dBm. Para fornecer energia ao amplificador pode ser utilizado um conversor DC-DC interno ao *chip* ou pela utilização de um regulador de tensão. No projeto optou-se pela utilização do conversor DC-DC para fornecer energia ao núcleo do *chip* e para o amplificador PA. Essa configuração faz com que o conversor DC-DC opere com 90% de eficiência, reduzindo drasticamente o consumo (Semtech, 2017).

Assim, um indutor de 15nH é adicionado entre os pinos VREG e DCC_SW e um capacitor de 470nF é adicionado entre o pino VREG e GND para o funcionamento do conversor. Um cristal oscilador de 32 MHz é utilizado para gerar a frequência de referência para o SX1262, como apresentado na Figura 25.

Seguindo orientações de *design* da Semtech (2014), é apresentado na Figura 26 os filtros utilizados na saída do amplificador PA. Para que o sinal de RF não retorne para a fonte CC um componente de alta impedância é utilizado para garantir que o sinal não retorne (circuito em destaque "filtragem e polarização" na Figura 26). Assim é utilizado um RF chokes, também conhecidos como indutores de polarização, com finalidade de bloquear ou suprimir sinais de alta frequência.

Figura 26 – Detalhamento dos filtros e rede de correspondência para o amplificador PA e LNA.



Fonte: Própria (2025).

Os indutores de polarização são calculados considerando que tenham uma impedância de 10 vezes a impedância de correspondência de entrada, um valor de 47 nH para frequências acima de 800 MHz é suficiente para bloquear o retorno da energia, considerando que a alta impedância do indutor não seja ideal é utilizado capacitores de *bypass* para absorver a energia vazada com valores tipicamente de 47 nF e 68 pF (Semtech, 2018).

Utilizando as impedâncias fornecida pela folha de dados do *chip* de $11.7 + j4.8\Omega$ para uma potência de saída de 14 dBm, é considerado uma rede de correspondência

LC (circuito em destaque "rede de correspondência LC" na Figura 26) para obter uma impedância em que a potência de saída necessária é alcançada com o menor consumo de corrente.

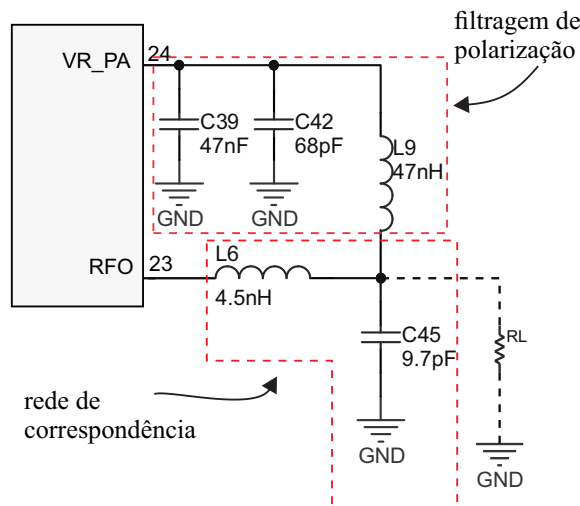
Para rede de correspondência LC é calculado o fator Q (Equação 5.1) e também o valor para o indutor correspondente (Equação 5.4) e capacitor (Equação 5.5). Para carga consideramos uma antena com impedância de 50Ω .

$$L_6 = \frac{1}{2\pi f} \times \left(\frac{50Q}{Q^2 + 1} + X_c \right) \quad (5.4)$$

$$C_{45} = \left| \frac{1}{2\pi f} \times \frac{\sqrt{\frac{R_s}{R_L}(1 + Q^2)} - 1 + Q}{R_s(1 + Q^2)} \right| \quad (5.5)$$

A Figura 27 apresenta o circuito equivalente para etapa anterior e ligação da filtragem de polarização, que impede que o sinal de RF retorna a fonte CC. Importante ressaltar que a implementação do circuito na PCB pode ocorrer em ajuste nos valores dos componentes devido à impedância da PCB e os valores dos componentes são arredondados para valores comerciais.

Figura 27 – Exemplificação do circuito equivalente do amplificador PA e conexões da rede de correspondência e filtragem de polarização.



Fonte: Própria (2025).

Para filtrar os harmônicos de segunda ordem gerados pelo sinal, considerando $f = 915MHz$, o segundo harmônico é $H2 = 1.83GHz$, é utilizado um filtro LC (circuito em destaque "filtragem 2° harmônico" na Figura 25). Como regra geral, a indutância de L_{15} (Figura 25) é selecionada como 3/4 da indutância utilizada pela rede de correspondência

calculada anteriormente, assim calculamos a valor do capacitor (Equação 5.6) para o filtro (Semtech, 2018).

$$C_{51} = \frac{1}{(2\pi H2)^2 \times L2} \quad (5.6)$$

Devido a efeitos parasitas causados pela PCB a potência de saída pode não ser a desejada, para isso é realizado ajuste nos valores de L_{15} e C_{51} , a potência também pode ser ajustada adicionando um capacitor (C_{57}) após o filtro do segundo harmônico para ajustar a incompatibilidade entre a rede de correspondência de L_6 e C_{45} com o filtro.

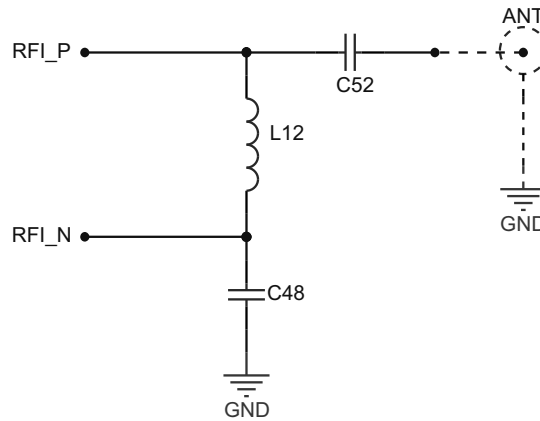
Posteriormente é adicionado um filtro π passa baixa para reduzir harmônicos de terceira e quarta ordem. Para utilizar a mesma antena para transmitir e receber é utilizado o BGS12SN6, fabricado pela Infineon⁸, que é um *switch* de RF. Para se obter melhor desempenho em suas especificações é recomendado adicionar um capacitor para bloquear o sinal DC na entrada e saída do sinal, assim é adicionado os capacitores C_{60} e C_{75} . Na saída do *switch* é utilizado uma rede de correspondência tipo π para casar com a impedância da antena de 50Ω como apresentado na Figura 26.

Um amplificador de baixo ruído (LNA) é utilizado para receber o sinal da antena e como o receptor possui uma entrada diferencial é projetado um circuito para converter o sinal que chega pela antena em um sinal diferencial. Esse processo é comumente realizado utilizando um balun que converte um sinal balanceado para um sinal desbalanceado.

Considerando as orientações de *design* para o *chip* SX1262 é recomendado a construção de um circuito *balun-like*, apresentado na Figura 28, utilizando um indutor e dois capacitores (Semtech, 2018). Essa proposta é uma alternativa a utilização de balun comerciais que possuem quatro ou seis elementos com alto ESR (*Equivalent Series Resistance*) que diminuem o desempenho da recepção.

⁸ Infineon - Soluções em MCUs, sensores, ICs automotivos e de gerenciamento de energia, memórias, USB, *Bluetooth*, WiFi e drivers de LED. Mais informações em: <https://www.infineon.com>

Figura 28 – Rede de correspondência RX, circuito "balun-like" proposto pela notas de aplicação.



Fonte: Adaptado de (Semtech, 2018).

Para essa etapa foi considerado a nota de aplicação "*LNA and PA Impedance Matching Techniques*"⁹ e também as orientações de *design*. Conhecendo a impedância de entrada do LNA, vista anteriormente, podemos calcular o valor da indutância (Equação 5.7) e dos capacitores (Equação 5.8 e 5.9). Os valores para R e X são respectivamente a impedância de entrada do amplificador LNA.

$$L_{12} = \frac{1}{2\pi f \sqrt{\frac{1}{R^2+X^2} - \left(\frac{R}{R^2+X^2}\right)^2}} \quad (5.7)$$

$$C_{52} = \frac{1}{2\pi f \times 50 \sqrt{\frac{1}{50} \left(\frac{R^2+X^2}{R}\right) - 1}} \quad (5.8)$$

$$C_{48} = \frac{2 \sqrt{\frac{1}{50} \left(\frac{R}{R^2+X^2}\right) - \left(\frac{R}{R^2+X^2}\right)^2}}{2\pi f} \quad (5.9)$$

Finalizando esta etapa é acionado um capacitor (C_{63}) antes do *switch* com objetivo de reduzir a quantidade de energia harmônica (Semtech, 2018). Assim foi projetado o circuito para realizar comunicação com tecnologia LoRa, como é proposto a utilização de três SX1262 o mesmo processo é repetido para o outro transceptor. É relevante ressaltar que o *design* da PCB e processo de fabricação causam alteração nos valores projetados, assim os valores finais devem ser ajustados realizando teste de bancada e verificando a potência de saída do transmissor.

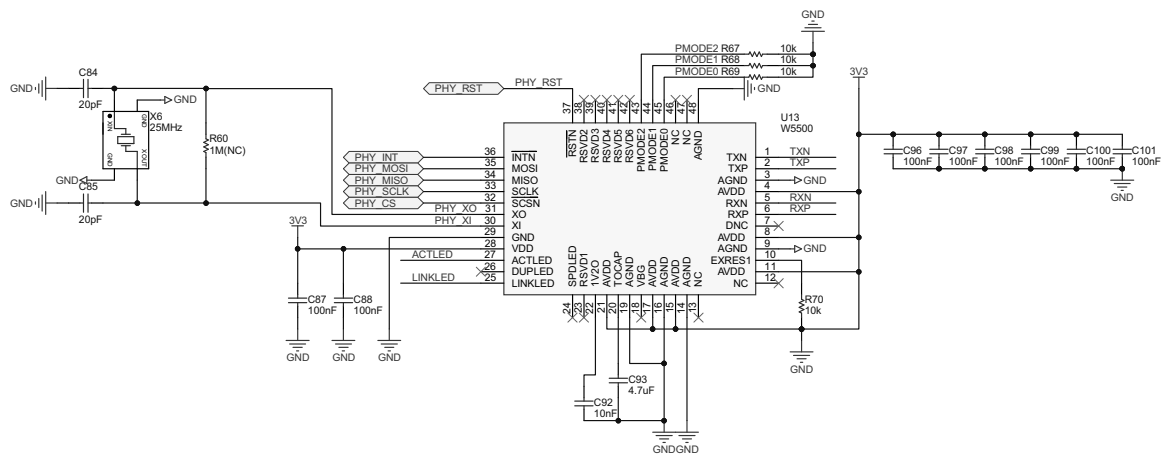
⁹ AN1200.16 Application Note: LNA and PA Impedance Matching Techniques for SX12xx - SEMTECH. Disponível em: <https://www.semtech.com/products/wireless-rf/frequency-shift-keying-fsk/sx1232#documentation>

5.1.4.2 Unidade Ethernet

Esta unidade possui uma camada física Ethernet possibilitando conectar o dispositivo a Internet via cabo não ficando dependente de conexão Wi-Fi como visto na Seção 5.1.1. O ESP32-S3 não possui interface para o padrão MII, RMII ou RGMII, por este motivo foi escolhido o *chip* W5500 que pode ser facilmente integrado ao dispositivo utilizando comunicação SPI, além de possuir um baixo custo.

A Figura 29 apresenta o circuito para o W5500 que utiliza um cristal oscilador de 25 MHz e os capacitores de desacoplamento 87 e 88 e também do C96 a 101. O *chip* possui um regulador de tensão interno que necessita de capacitores de 10 nF externo, assim foi adicionado o C92, assim como o C93 para estabilizar os sinais internos ao *chip*.

Figura 29 – Circuito para camada física *Ethernet* utilizando W5500.

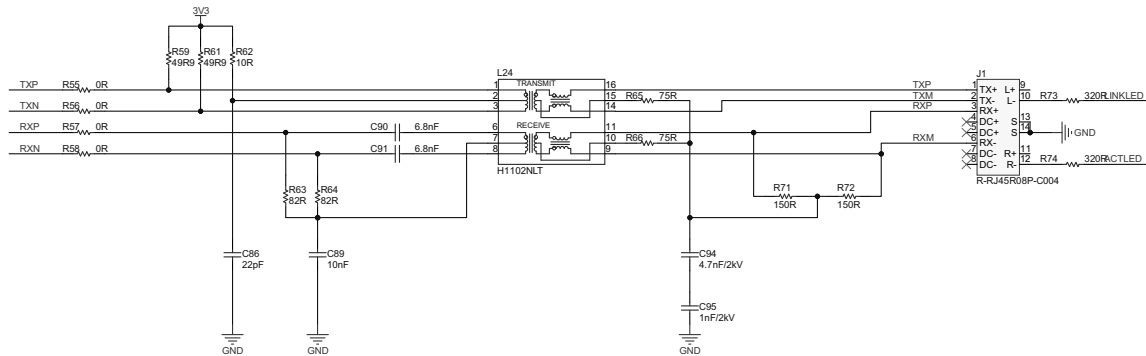


Fonte: Própria (2025).

O sinal diferencial de transmissão de dados é enviado pelos pinos TXP e TXN, enquanto a recepção ocorre por meio dos pinos RXP e RXN. De acordo com a norma EN60950¹⁰ que trata da segurança de equipamentos de tecnologia da informação, a Ethernet é classificada como um circuito de tensão de rede de telecomunicações (TNV) e, portanto, deve ser isolada dos circuitos de extra baixa tensão (SELV).

Assim é utilizado um transformador próprio para redes Ethernet, como visto na Figura 30, com a finalidade de realizar o isolamento do sinal de saída do W5500 com o sinal que trafega pelo cabo de par metálico. Para atender a norma EN60950 o isolamento deve suportar uma descarga de 1,5 KV. Deste modo, no circuito foi utilizado dois capacitores, C94 e C95, para drenar a descarga que possa ocorrer na carcaça ou no cabo para o terra do circuito, suportando os 1,5 KV.

¹⁰ EN60950 - *Information technology equipment Safety*. Mais informações em: https://webstore.iec.ch/-preview/info_iec60950-1%7Bed2.0%7Den_d.pdf

Figura 30 – Circuito para isolamento da linha de transmissão da camada física *Ethernet*.

Fonte: Própria (2025).

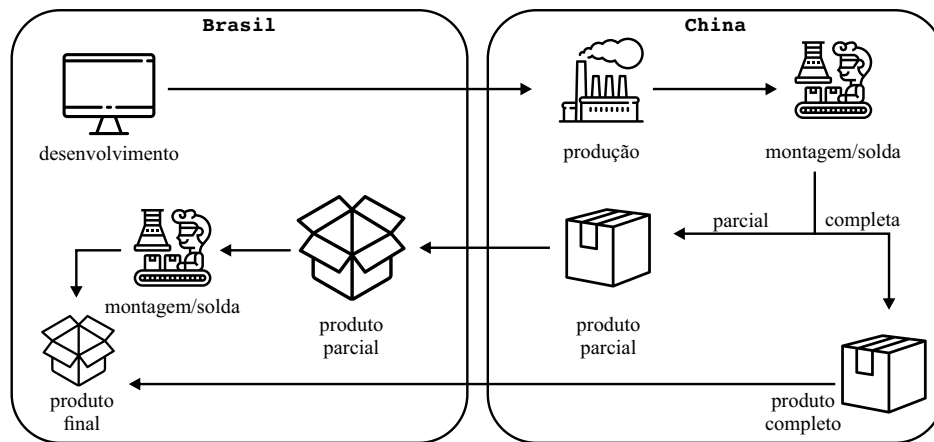
5.2 Manufatura das placas de circuito impresso

O processo de confecção de PCBs pode ser realizado por meio de técnicas manuais, como transferência térmica ou litografia. No entanto, tais métodos, por serem de caráter artesanal, não proporcionam resultados adequados para projetos de maior complexidade. No caso específico deste trabalho, a placa possui quatro camadas, sendo uma destinada ao plano de terra (GND), outra ao roteamento da alimentação (plano VCC) e duas camadas dedicadas ao roteamento de sinais, com largura mínima de trilha de 8 mil (0,2 mm). Além disso, a utilização de componentes SMD (*Surface Mount Device*) foi adotada com o intuito de reduzir as dimensões físicas da placa. Diante dessas exigências técnicas, optou-se pela fabricação do circuito por meio de processo industrial.

No Brasil existem empresas que realizam a confecção de placas, mas os custos não são atraentes tornando o produto final menos competitivo. Países como o Japão e a China, possuem empresas de grande porte para fabricação de placas eletrônicas, dando a possibilidade de produção, compra dos componentes e soldagem dos mesmos, com custos acessíveis.

A produção das placas foi realizada em uma unidade fabril no Japão. A Figura 31 apresenta as alternativas de produção disponíveis. Inicialmente, o projeto finalizado é enviado para a confecção da placa. Em seguida, é possível definir se a aquisição dos componentes será realizada em conjunto com a etapa de montagem/soldagem. Nessa fase, pode-se optar por receber o produto totalmente finalizado, parcialmente montado (com apenas alguns componentes específicos soldados) ou apenas a PCB. Neste último caso, o processo de soldagem pode ser terceirizado a outra empresa ou realizado manualmente.

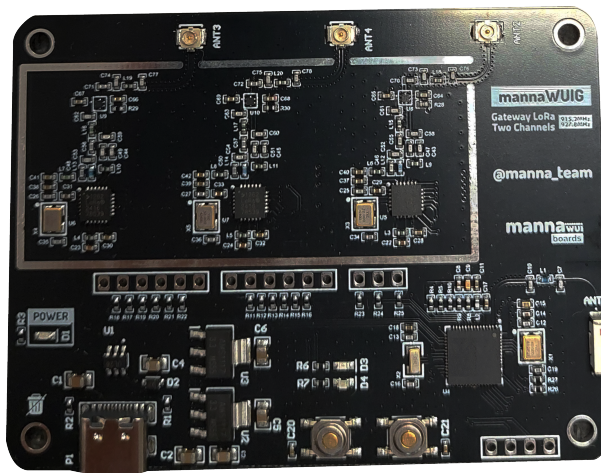
Figura 31 – Etapas de produção fornecidas por empresas de manufatura de placas de circuito impresso.



Fonte: Própria (2025).

Com a fabricação da PCB e a aquisição dos componentes necessários, foi realizada a soldagem manual dos dispositivos eletrônicos na placa. A Figura 32 apresenta imagem do dispositivo finalizado, com todos os componentes devidamente montados.

Figura 32 – Placa finalizada do gateway de canal duplo.



Fonte: Própria (2025).

5.3 Aquisição dos componentes e valores envolvidos

Adquirir componentes específicos em território nacional é um desafio. Principalmente quando se deseja obter os componentes específicos e avulsos. Alguns componentes são encontrados no formato de módulos e com grande dificuldade no modo avulso. Assim todos os componentes e conectores foram adquiridos por meio de distribuidores chineses

ou americanos como Mouser¹¹, DigiKey¹² e LCSC¹³.

Com objetivo de obter um projeto do *gateway* de baixo custo, a Tabela 10 apresenta uma projeção do custo para construção do dispositivo. Com objetivo de simplificar a tabela, todos os componentes como resistores, capacitores, indutores, LEDs e diodos foram contabilizados como sendo suplementos. A cotação para fabricação da PCB foi considerada 10 unidades, tendo seu valor dividido para obter o valor unitário. Para esta projeção não foram considerados valores de mão de obra para soldagem dos componentes. Os valores são apresentados em dólares, considerando que todos os componentes necessários são importados.

Tabela 10 – Custo para construção de um *gateway*.

Componentes	Quantidade	Valores (US\$)
ESP32-S3FH4R2	1	4.25
W25Q16JVSNIQ/ MX66U1G45G	1	9.46
APS6404L-3SQR	1	1.82
SX1262IMLTRT	3	9.48
BGS12SN6	3	0.90
W5500	1	2.68
FT232RL-Reel	1	4.71
MCP1826ST-3302E	2	3.00
PRTR5V0U2F,115	2	1.30
MC-311D	1	1.00
TCXO 32.768KHz	1	2.57
Crystal 25MHz	1	0.87
Crystal 40MHz	1	1.33
Crystal 32MHz	3	0.81
Suplementos	-	8.00
PCB	-	5.92
Total	-	58.10

Fonte: Própria (2025).

Também foi elaborada uma tabela comparativa entre o *gateway* proposto e modelos comerciais disponíveis no mercado. A Tabela 11 apresenta os preços de diferentes *gateways* LoRa comerciais, que variam de US\$401,25 até US\$1.429,90. A última linha da tabela corresponde ao *gateway* proposto, cujo custo estimado é de US\$58,10. É importante

¹¹ A Mouser Electronics é uma distribuidora global de semicondutores e componentes eletrônicos. Mais informações em: www.mouser.com

¹² A Digi-Key oferece milhões de produtos de milhares de fabricantes de componentes eletrônicos. Mais informações em: www.digikey.com.br

¹³ LCSC Electronic components oferece uma grande seleção de produtos de alta qualidade de muitas marcas asiáticas. Mais informações em: www.lsc.com

destacar que esse valor refere-se ao protótipo, não representando o preço comercial final, pois não inclui os impostos incidentes sobre a fabricação e a comercialização do produto. Para fins de estimativa, foi calculado também o valor aproximado em reais, considerando o imposto de importação do Brasil de 60% e o ICMS de 19,5% vigente no estado do Paraná, sem contemplar tributos adicionais relacionados à produção nacional, como o IPI. Ainda assim, a comparação demonstra que a proposta se manteria como uma solução de baixo custo para implementação em redes de pequeno e médio porte, configurando-se como alternativa intermediária entre os *gateways* de canal único de menor custo e os modelos multicanais robustos de alto valor.

Tabela 11 – Comparação dos valores de *gateways* comerciais.

Marca / Fabricante	Modelo	Preço Aprox. (US\$)	Preço Aprox. (R\$)
Kerlink	iBTS LoRaWAN Gateway	1,429.90	14.116,69
RAK Wireless	WisGate Edge Ultra Full-Duplex	859.00	8.480,48
Milesight	UG65 Robust	460.00	4.541,35
Milesight	UG67 Outdoor Gateway	1,275.00	12.587,44
Dragino	DLOS8N	401.25	3.961,34
PLANET	LCG-300W-US	1,027.60	10.144,98
MultiTech	Conduit AP (LTE Cat 1)	509.00	5.025,10
MultiTech	Conduit LTE Cat 4 (GNSS, Wi-Fi)	825.00	8.144,81
MultiTech	Conduit 300 Series	1,403.00	13.851,12
mannaWUIG	mannaWUIG (Wi-Fi/Ethernet)	58.10	573,59

Fonte: Própria (2025).

5.4 Conclusão e análise de execução do projeto

Neste capítulo foi apresentada a composição do *hardware* proposto para o *gateway* de canal duplo, com ênfase na descrição de cada subsistema necessário ao funcionamento do dispositivo. Destacou-se a utilização de um MCU com capacidade de processamento de algoritmos leves de IA, alinhado à necessidade de aproximar os serviços tradicionalmente executados em nuvem para a borda da rede (KANSAL; KUMAR; VERMA, 2022). Também foram discutidas as etapas de projeto do circuito de RF para o transceptor baseado na tecnologia LoRa, considerando a busca pelo melhor desempenho associado ao menor consumo de energia.

A proposta contempla o uso de três camadas físicas para o atendimento simultâneo de dispositivos, sendo duas delas dedicadas ao *uplink*, em virtude da predominância do envio de dados por dispositivos IoT (LAVRIC; PETRARIU, 2018). A terceira camada física é destinada à execução do mecanismo CAD e às operações de *downlink*.

Por fim, foram descritos o processo de manufatura das placas de circuito impresso, a montagem do dispositivo e a análise dos custos envolvidos. No capítulo seguinte, é apresentado um estudo de caso empregando o *gateway* de canal duplo, conduzido a partir da avaliação de três cenários distintos, de modo a validar o desempenho e a aplicabilidade da proposta.

Capítulo 6

mannaWUIG: *Gateway* LoRa de canal duplo

A IoT impulsiona o crescimento de dispositivos conectados e, conseqüentemente, o volume de dados trafegados, impondo desafios à escalabilidade das redes. Nesse cenário, as redes LPWAN, em especial o LoRaWAN, destacam-se por oferecer longo alcance e baixo consumo de energia, embora apresentem limitações decorrentes do uso do protocolo ALOHA e da dependência de *gateways* de baixo custo, muitas vezes semelhantes aos nós sensores.

Como discutido no Capítulo 3, a literatura aponta que, apesar dos avanços em algoritmos e soluções de *software* voltados à eficiência da comunicação, as restrições de *hardware* e custo ainda conduzem à adoção de redes de canal único, mais suscetíveis a colisões em cenários densos. Estratégias como o CAD e o uso adaptativo de múltiplos canais têm se mostrado promissoras para mitigar interferências e ampliar a capacidade da rede, evidenciando a necessidade de soluções intermediárias entre os *gateways* simples e os modelos multicanais de alto custo.

Esses desafios necessitam de soluções mais acessíveis e eficientes, como a proposta apresentada neste trabalho, que visa combinar múltiplos canais de recepção e mecanismos CAD em um *gateway* de baixo custo, contribuindo para o avanço das redes LoRaWAN em diferentes contextos de aplicação.

Este capítulo está organizado da seguinte forma: inicialmente, apresenta-se a adaptação realizada no protocolo LoRaWAN para o *gateway* de canal duplo (Seção 6.1). Em seguida, descreve-se o estudo de simulação do *gateway* proposto (Seção 6.2). Por fim, são discutidos os resultados de um estudo de caso em cenário real (Seção 6.3.4).

6.1 Adaptação do protocolo LoRaWAN para o *gateway* de canal duplo

Propõe-se um protocolo que mantém a compatibilidade com a pilha LoRaWAN Classe A e substitui o mecanismo de salto de frequência pseudoaleatório por uma seleção determinística de canais, fundamentada em um *timestamp* (h, m) compartilhado entre o dispositivo e o *gateway*. Esse *timestamp* é fornecido pelo *gateway* durante o processo de junção à rede e passa a constituir a referência temporal para determinar, a cada minuto, o canal ativo de *uplink* e o canal destinado ao *downlink*.

O protocolo tem como objetivo garantir a sincronização entre o dispositivo e *gateway* de canal duplo. Os canais são alocados de forma coordenada e previsível, garantindo que, a cada minuto, eles cubram dois canais distintos do conjunto AU915-928. Tanto os dispositivos quanto o *gateway* utilizam o *timestamp*, composto pelas variáveis hora (h) e minuto (m), como base para calcular o canal de comunicação ativo, conforme descrito na Equação 6.1 para a seleção do canal A (ch_A):

$$ch_A = ((h \times 60) + m) \times MOD(CH) \quad (6.1)$$

Onde h representa a hora atual, m corresponde ao minuto atual e CH indica a quantidade de canais disponíveis para *uplink* ou *downlink*.

O canal B (ch_B) é derivado do canal A por meio de um deslocamento fixo, como apresentado na Equação 6.2.

$$ch_B = (ch_A + \Delta) \times MOD(CH) \quad (6.2)$$

Onde Δ é um valor constante definido pelo *gateway* que assegura que os dois rádios *front-end* nunca coincidam no mesmo canal. Dessa forma, foi implementada a função de seleção de canais apresentada no Algoritmo 1.

Algorithm 1 Seleção determinística de canais no *gateway*.

Require: h, m

Require: Δ

Require: $CH_{UL} \leftarrow 64$

Require: $CH_{DL} \leftarrow 8$

```

1: function SELECTGATEWAYCHANNELS( $h, m, \Delta$ )
2:    $ch\_A \leftarrow ((h \times 60) + m) \bmod CH_{UL}$ 
3:    $ch\_B \leftarrow (ch\_A + \Delta) \bmod CH_{UL}$ 
4:    $f\_UL\_A \leftarrow 915.2 \text{ MHz} + 200 \text{ kHz} \times ch\_A$ 
5:    $f\_UL\_B \leftarrow 915.2 \text{ MHz} + 200 \text{ kHz} \times ch\_B$ 

6:    $ch'\_A \leftarrow ((h \times 60) + m) \bmod CH_{DL}$ 
7:    $ch'\_B \leftarrow (ch'\_A + \Delta) \bmod CH_{DL}$ 
8:    $f\_RX1\_A \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch'\_A$ 
9:    $f\_RX1\_B \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch'\_B$ 
10:   $ch^{(2)}\_A \leftarrow ((h \times 60) + ((m + 1) \bmod 60)) \bmod CH_{DL}$ 
11:   $ch^{(2)}\_B \leftarrow (ch^{(2)}\_A + \Delta) \bmod CH_{DL}$ 
12:   $f\_RX2\_A \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch^{(2)}\_A$ 
13:   $f\_RX2\_B \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch^{(2)}\_B$ 
14:  return ( $f\_UL\_A, f\_UL\_B, f\_RX1\_A, f\_RX1\_B, f\_RX2\_A, f\_RX2\_B$ )

```

Para evitar que todos os nós transmitam no mesmo canal ativo, aplica-se um mecanismo de dispersão determinística. O identificador único do dispositivo (*DevEUI*) é utilizado para selecionar se o nó transmitirá em ch_A ou em ch_B , conforme apresentado na Equação 6.3.

$$ch(DevEUI) = \begin{cases} ch_A, & \text{se } ID(DevEUI) \bmod 2 = 0, \\ ch_B, & \text{se } ID(DevEUI) \bmod 2 = 1. \end{cases} \quad (6.3)$$

Onde $ID(DevEUI)$ corresponde ao valor numérico associado ao identificador do dispositivo.

Para *downlink*, aplica-se a mesma lógica ao conjunto de canais. O *gateway* seleciona dois canais de *downlink* por minuto, e cada dispositivo segue o mesmo critério baseado em $ID(DevEUI)$ para decidir qual canal monitorar em RX1/RX2. O Algoritmo 2 apresenta o pseudocódigo correspondente, implementado na pilha LoRaWAN.

Algorithm 2 Seleção de canal no dispositivo.

Require: h, m
Require: Δ
Require: $ID(DevEUI)$
Require: $CH_UL \leftarrow 64, CH_DL \leftarrow 8$
Require: $forceCH \in \{auto, force_A, force_B\}$

1: **function** SELECTULCHANNEL($h, m, \Delta, ID, forceCH$)
2: $ch_A \leftarrow ((h \times 60) + m) \bmod CH_UL$
3: $ch_B \leftarrow (ch_A + \Delta) \bmod CH_UL$
4: **if** $forceCH = force_A$ **then**
5: $ch_UL \leftarrow ch_A$
6: **else if** $forceCH = force_B$ **then**
7: $ch_UL \leftarrow ch_B$
8: **else**
9: **if** $ID \bmod 2 = 0$ **then**
10: $ch_UL \leftarrow ch_A$
11: **else**
12: $ch_UL \leftarrow ch_B$
13: $f_UL \leftarrow 915.2 \text{ MHz} + 200 \text{ kHz} \times ch_UL$
14: **return** f_UL

15: **function** SELECTDLCHANNELS($h, m, \Delta, ID, forceCH$)
16: $ch'_A \leftarrow ((h \times 60) + m) \bmod CH_DL$
17: $ch'_B \leftarrow (ch'_A + \Delta) \bmod CH_DL$
18: **if** $forceCH = force_A$ **then**
19: $ch'_{RX1} \leftarrow ch'_A$
20: **else if** $forceCH = force_B$ **then**
21: $ch'_{RX1} \leftarrow ch'_B$
22: **else**
23: **if** $ID \bmod 2 = 0$ **then**
24: $ch'_{RX1} \leftarrow ch'_A$
25: **else**
26: $ch'_{RX1} \leftarrow ch'_B$
27: $f_RX1 \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch'_{RX1}$
28: $ch'_A^{(2)} \leftarrow ((h \times 60) + ((m + 1) \bmod 60)) \bmod CH_DL$
29: $ch'_B^{(2)} \leftarrow (ch'_A^{(2)} + \Delta) \bmod CH_DL$
30: **if** $forceCH = force_A$ **then**
31: $ch'_{RX2} \leftarrow ch'_A^{(2)}$
32: **else if** $forceCH = force_B$ **then**
33: $ch'_{RX2} \leftarrow ch'_B^{(2)}$
34: **else**
35: **if** $ID \bmod 2 = 0$ **then**
36: $ch'_{RX2} \leftarrow ch'_A^{(2)}$
37: **else**
38: $ch'_{RX2} \leftarrow ch'_B^{(2)}$
39: $f_RX2 \leftarrow 923.3 \text{ MHz} + 600 \text{ kHz} \times ch'_{RX2}$
40: **return** (f_RX1, f_RX2)

Para balancear a carga entre os dois canais ativos por minuto, o *gateway* pode realinhar cada dispositivo após a junção a rede por meio de um comando MAC proprietário, mantendo o comportamento determinístico e a sincronização baseada em (h,m). Nesse mecanismo, o *gateway* envia ao dispositivo um comando MAC que pode forçar a utilização de ch_A ou ch_B , alternativamente, mantê-lo no modo automático definido a partir do $ID(DevEUI)$. O *gateway* envia ao dispositivo o comando no próximo *downlink* disponível, definindo a variável *forceCH* do Algoritmo 2.

O protocolo preserva todas as demais funcionalidades do LoRaWAN padrão, incluindo autenticação, criptografia e mecanismos de *Adaptive Data Rate* (ADR). A sincronização temporal é mantida por meio de mensagens MAC como *DeviceTimeReq* ou *DeviceTimeAns*, permitindo corrigir desvios do relógio interno do dispositivo.

6.2 Simulação do *gateway* de canal duplo

Para avaliar o desempenho da adaptação do protocolo LoRaWAN proposto, foram realizadas simulações utilizando o ambiente LoRaWAN-SIM¹, uma ferramenta utilizadas em outros trabalhos científicos como em Zorbas (2022), Javed e Zorbas (2023), Zorbas (2024) e Ndie et al. (2025).

O objetivo foi comparar o comportamento do *gateway* de canal único (GW1), operando apenas em um canal de recepção e *gateway* de canal duplo (GW2), configurado com o protocolo proposto na Seção 6.1. A Tabela 12 apresenta os parâmetros utilizados no simulador LoRaWAN-SIM.

Tabela 12 – Parâmetros de simulação.

Parâmetros	Valores
Tempo de simulação	10000 s
Dispositivos finais/Distribuição	10 – 500 (Uniforme)
Gateways	1
Raio do terreno	1500 m
<i>Spreading Factors</i>	7 - 12
Largura de banda do canal <i>uplink</i>	125 KHz
Largura de banda do canal <i>downlink</i>	500 KHz
Símbolos de preâmbulo	8
CR	4/5
Mapa de canais para <i>Uplink/Downlink</i>	AU915-928
Tamanho do <i>payload</i>	17 bytes
Sensibilidades do receptor	Semtech SX1276
Potência de transmissão	10 dBm

Fonte: Própria (2025).

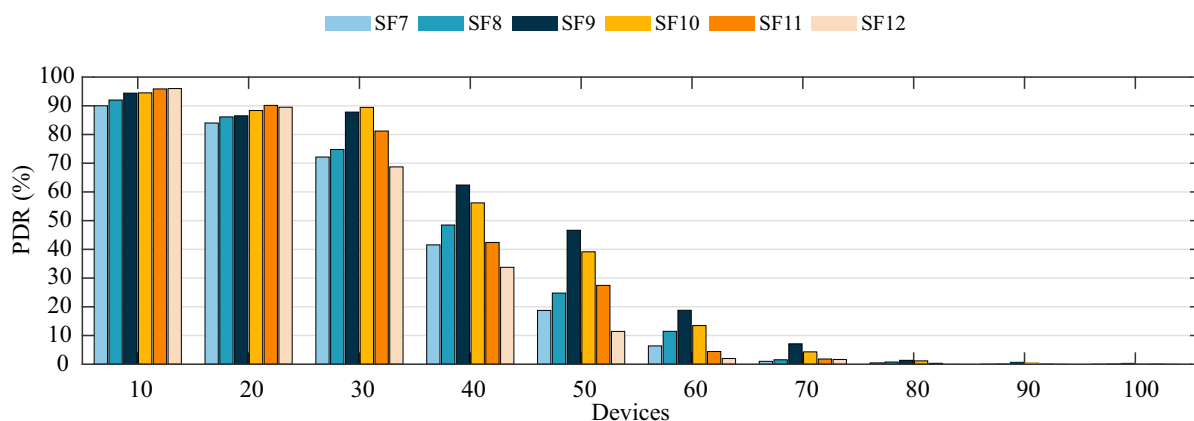
As métricas foram obtidas variando-se o número de dispositivos finais entre 10 e 100 para o *gateway* de canal único e entre 10 e 500 para o *gateway* de canal duplo, analisando-se

¹ O LoRaWAN-SIM é um simulador do protocolo LoRaWAN utilizado em pesquisas científicas e também por empresas. Mais informações podem ser encontradas em: <<https://github.com/deltazita/LoRaWAN-SIM>>

o PDR (*Packet Delivery Ratio*) em função dos diferentes fatores de espalhamento (SF7 a SF12).

A Figura 33 apresenta o desempenho da rede considerando a utilização de um *gateway* de canal único. Observa-se que o PDR diminui progressivamente à medida que o número de dispositivos aumenta. Embora os fatores de espalhamento apresentem desempenho inicial, com PDR próximos de 90% para até 20 dispositivos, essa taxa declina rapidamente com o aumento de nós na rede. Para os fatores mais elevados (SF10, SF11 e SF12), a queda de desempenho é ainda mais acentuada, alcançando valores inferiores a 40% de PDR a partir de 50 dispositivos.

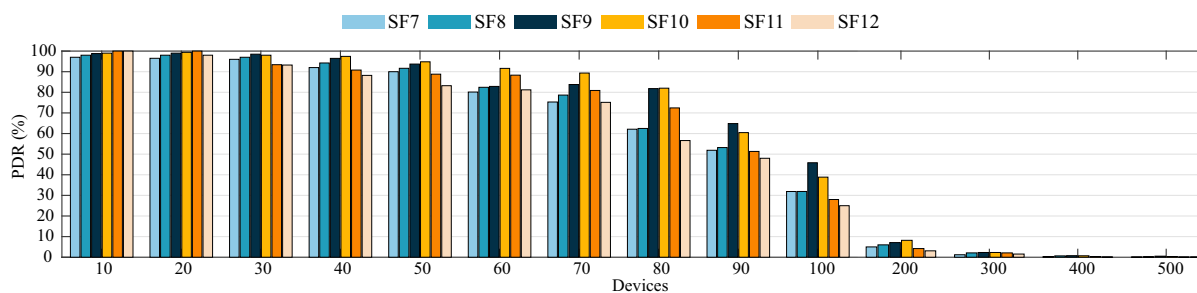
Figura 33 – Simulação *gateway* de canal único.



Fonte: Própria (2025).

A Figura 34 apresenta os resultados do cenário com *gateway* de canal duplo, configurado com o protocolo proposto. Em comparação com a primeira simulação (Figura 33), observa-se uma melhora significativa no PDR, mantendo-se próximo de 90% para aproximadamente 40 dispositivos, ponto em que, no cenário de canal único, inicia-se um declínio mais acentuado na taxa de recepção.

Figura 34 – Simulação *gateway* de canal duplo.



Fonte: Própria (2025).

Para fatores de espalhamento mais baixos (SF7, SF8 e SF9), observa-se que até aproximadamente 60 dispositivos na rede o PDR permanece próximo de 80%. Com o aumento da densidade de nós, os fatores SF9 e SF10 passam a apresentar o melhor

desempenho em comparação aos demais, evidenciando maior eficiência nessas condições. Por outro lado, a utilização de SFs mais elevados, como SF11 e SF12, prolonga o tempo de ocupação do canal, de modo que os dispositivos operando nessas configurações passam a apresentar pior desempenho na rede, comparado aos dispositivos que utilizam SFs mais baixos.

Conforme ilustrado na Figura 34, o *gateway* de canal duplo apresenta maior dificuldade de desempenho quando o número de dispositivos conectados ultrapassa 60. Entretanto, observa-se que, ao operar com fatores de espalhamento na faixa de SF9 a SF10, o *gateway* consegue manter a taxa de PDR em torno de 80% mesmo com aproximadamente 80 dispositivos conectados. Essa característica de melhor desempenho nos SF9, SF10 e SF11 também é observada em trabalhos que buscam aprimorar redes LoRaWAN, mas que partem da premissa da utilização de *gateways* de alto desempenho, como discutido em Javed e Zorbas (2023) e Zorbas (2024).

A partir dessas simulações, define-se que redes de pequeno porte correspondem a configurações com até 50 dispositivos, enquanto redes de médio porte alcançam até 100 dispositivos. Embora ainda se observe uma redução gradual da PDR para o *gateway* de canal duplo com o aumento da densidade de nós, os valores obtidos permanecem significativamente superiores aos verificados no *gateway* de canal único, chegando em muitos casos a apresentar mais do que o dobro de desempenho. Na próxima seção, apresenta-se um estudo de caso em ambiente real, no qual o protótipo do *gateway* de canal duplo é avaliado em três cenários distintos.

6.3 Estudo de caso

Com o aumento da popularidade da IoT e dos dispositivos que utilizam redes LPWAN, como a tecnologia LoRa, a implementação de novas aplicações pode ocorrer em locais onde já existam redes privadas ou públicas, seja nas proximidades ou no próprio local.

Para avaliar o desempenho do *gateway* de canal duplo, foi adotado o design apresentado no Capítulo 4, Figura 16, que serviu como base para o desenvolvimento e a produção do protótipo. A arquitetura do sistema foi concebida utilizando transceptores LoRa SX1262, responsáveis pela comunicação nos dois canais distintos, enquanto a unidade central de processamento baseou-se no microcontrolador ESP32-S3FH4R2, que integra 4 MB de memória flash e 2 MB de RAM internos. Além disso, o projeto prevê a possibilidade de expansão da capacidade de armazenamento e processamento com a incorporação de memória flash externa (MX66U1G45G, 128 MB) e RAM adicional (APS6404L-3SQR, 8 MB), o que oferece flexibilidade para aplicações mais exigentes, mantendo o baixo custo de prototipação.

A escolha da família ESP32-S3 é pela presença de unidade de aceleração vetorial, projetada para atender aplicações com requisitos computacionais associados à inteligência artificial embarcada (AIoT). Esse recurso permite a execução eficiente de algoritmos de inferência, como redes neurais convolucionais (CNNs), diretamente no dispositivo, sem a necessidade de *offloading* para servidores externos. O ESP32-S3 oferece suporte nativo a *frameworks* de aprendizado de máquina otimizados para sistemas embarcados, como o TensorFlow Lite for Microcontrollers² e a biblioteca Espressif ESP-DL³, possibilitando a integração de modelos de IA leves para análise em tempo real, com baixa latência e reduzido consumo de energia, características essenciais para computação na borda.

A versão final do dispositivo, utilizada nos testes experimentais, é apresentada na Figura 32. Os nós sensores empregados nesta aplicação correspondem a uma versão aprimorada do dispositivo descrito em (BINE; RUIZ, 2020).

Para fins comparativos, a aplicação foi inicialmente desenvolvida utilizando apenas nós sensores, sendo que um dispositivo idêntico a um nó sensor foi configurado para operar como *gateway* de canal único. Posteriormente, para cada cenário a mesma aplicação foi executada com o *gateway* de canal duplo proposto. O protocolo LoRaWAN baseado em ALOHA foi adaptado e empregado em todos os cenários, e os experimentos foram conduzidos em uma universidade com a presença de redes LoRa coexistentes, permitindo uma análise do *gateway* proposto em condições reais de operação.

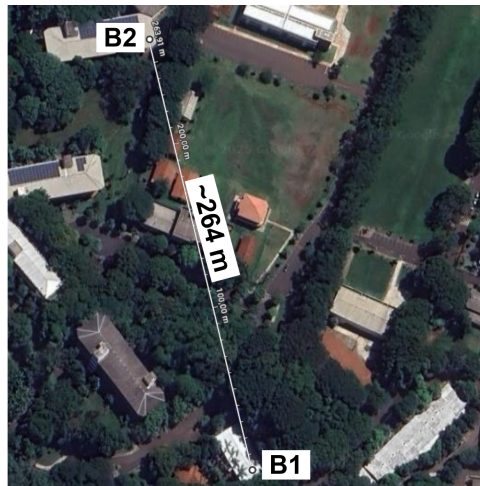
Ambos os *gateways*, o de canal único e o de canal duplo, foram instalados no bloco do laboratório de pesquisa. Os nós sensores foram distribuídos entre dois blocos da universidade, como mostra a Figura 35, separados por uma distância em linha reta de aproximadamente 264 metros.

A Figura 36 apresenta o *gateway* de canal duplo e os 20 nós sensores utilizados, disponíveis no laboratório Manna Team, para o monitoramento de temperatura e umidade do ambiente. A rede LoRa foi configurada com os seguintes parâmetros: BW de 125kHz, CR de 4/5 e potência de transmissão ajustada para 10dBm.

² TensorFlow Lite for Microcontrollers é uma versão otimizada do TensorFlow projetada para execução em dispositivos com recursos extremamente limitados, como microcontroladores. Mais informações em: <<https://www.tensorflow.org/lite/microcontrollers>>.

³ ESP-DL é uma biblioteca de deep learning desenvolvida pela Espressif Systems para execução em dispositivos embarcados, oferecendo suporte a operações de inferência otimizadas e aceleradores de *hardware* dedicados. Mais informações em: <<https://docs.espressif.com/projects/esp-dl>>.

Figura 35 – Mapa onde os dispositivos foram instalados, distribuídos entre os blocos B1 e B2.

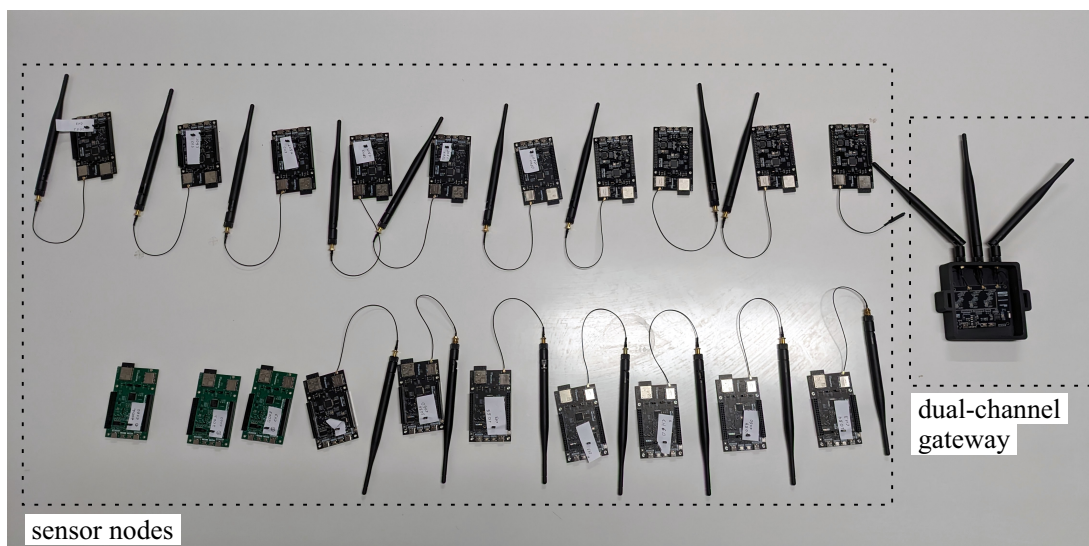


Fonte: Própria (2025).

Em todos os cenários avaliados, o *payload* útil transmitido foi composto por 4 bytes, representando os valores de temperatura e umidade. Considerando os cabeçalhos e comandos do protocolo LoRaWAN, o tamanho total do pacote transmitido foi de 17 bytes.

Durante o período de realização dos experimentos, as condições climáticas permaneceram estáveis, sem ocorrência de chuvas. A temperatura média máxima registrada ao longo dos sete dias de testes foi de aproximadamente 29°C .

Figura 36 – Nós sensores usados para testes e o *gateway* de canal duplo.



Fonte: Própria (2025).

6.3.1 Cenário 1

Para a avaliação do primeiro cenário, os dispositivos realizam a medição da temperatura e umidade do ambiente, transmitindo os dados em intervalos de 1 a 2 minutos.

Em cada transmissão, o tempo foi selecionado de forma aleatória, utilizando-se uma semente fixa para todos os cenários, a fim de garantir a consistência dos resultados. Os dispositivos foram configurados para operar na classe A, abrindo até duas janelas de recepção após cada transmissão com o objetivo de receber a confirmação do *gateway*.

O *gateway* de canal único seleciona aleatoriamente seu canal de transmissão durante a etapa de configuração inicial, mantendo-o fixo ao longo de toda a operação. Para o *gateway* de canal duplo, a medida que os dispositivos ingressam na rede, é determinado em qual dos dois canais disponíveis cada dispositivo será alocado, transmitindo as informações de configuração por meio de um canal de *downlink* previamente definido pelo algoritmo.

Para garantir a sincronização, o protocolo LoRaWAN implementado nos nós sensores foi adaptado de modo que cada dispositivo realize suas transmissões exclusivamente no canal previamente atribuído pelo *gateway*.

Neste cenário, os dois canais nos quais o *gateway* irá operar são selecionados aleatoriamente durante sua inicialização e configuração. A partir desse ponto, todos os dispositivos que ingressam na rede passam a operar entre esses dois canais. Para cada novo dispositivo integrado, o *gateway* avalia a distribuição de dispositivos em cada canal e aloca o novo dispositivo no canal com menor número de dispositivos, visando manter o balanceamento entre os dois canais disponíveis.

6.3.2 Cenário 2

Neste cenário, foi utilizada a CAD antes de cada transmissão. Mantiveram-se as mesmas configurações do primeiro cenário, mas, adicionalmente, registrou-se a quantidade de verificações CAD realizadas antes da efetiva transmissão dos dados.

6.3.3 Cenário 3

Neste cenário, é avaliado o desempenho do *gateway* de canal duplo operando com uma política de mudança dinâmica de canal, baseada na sincronização temporal entre os dispositivos e o *gateway*.

Em redes LoRaWAN, os dispositivos normalmente realizam transmissões de *uplink* em canais selecionados aleatoriamente, dentre os 64 canais disponíveis. No entanto, quando se adota um *gateway* com suporte a apenas dois canais, torna-se necessário garantir que o dispositivo e o *gateway* estejam operando no mesmo canal no momento da comunicação.

Para garantir essa sincronização, foi implementado um mecanismo determinístico de seleção de canal, utilizando um *timestamp* obtido durante o processo de junção à rede, como discutido na Seção 6.1. Tanto os dispositivos quanto o *gateway* utilizam este *timestamp*, composto pelas variáveis hora (h) e minuto (m), como base para calcular o

canal de comunicação ativo.

Este mecanismo garante que, a cada novo minuto, o canal de transmissão seja alternado de forma sincronizada entre o dispositivo e o *gateway*, sem a necessidade de sinalização adicional. Como o cálculo é baseado em uma função determinística comum a ambas as extremidades, o sincronismo se mantém desde que a base de tempo esteja alinhada.

Para garantir essa sincronização temporal, tanto o *gateway* quanto os dispositivos sensores são equipados com módulos RTC+TCXO. O *timestamp* inicial é fornecido pelo *gateway* no momento da associação do dispositivo à rede e, a partir desse ponto, cada nó mantém sua contagem de tempo local utilizando o RTC+TCXO, assegurando a precisão necessária para que a troca de canais ocorra de maneira coordenada. Caso sejam identificados desvios no RTC, o *gateway* pode transmitir mensagens MAC do tipo DeviceTimeAns, em resposta a requisições DeviceTimeReq enviadas pelo nó, possibilitando a correção desses desvios.

A utilização dos 64 canais disponíveis para transmissões LoRaWAN contribui significativamente para a mitigação de congestionamentos em ambientes com múltiplas redes coexistentes. Em cenários onde aplicações simples, muitas vezes baseadas em *gateways* de canal único, operam de forma fixa em determinadas frequências, há uma tendência à sobrecarga de canais específicos, o que aumenta a probabilidade de colisões.

Nesse contexto, a alternância periódica e sincronizada dos canais de transmissão adotada pelo *gateway* de canal duplo proposto representa uma estratégia para distribuir de forma mais equilibrada o tráfego no espectro. Embora o *gateway* opere simultaneamente em apenas dois canais, o mecanismo de comutação dinâmica permite explorar melhor a diversidade de frequências disponíveis, reduzindo a interferência e aumentando a robustez da comunicação, especialmente em redes com alta densidade de dispositivos ou coexistência de múltiplas redes LoRaWAN.

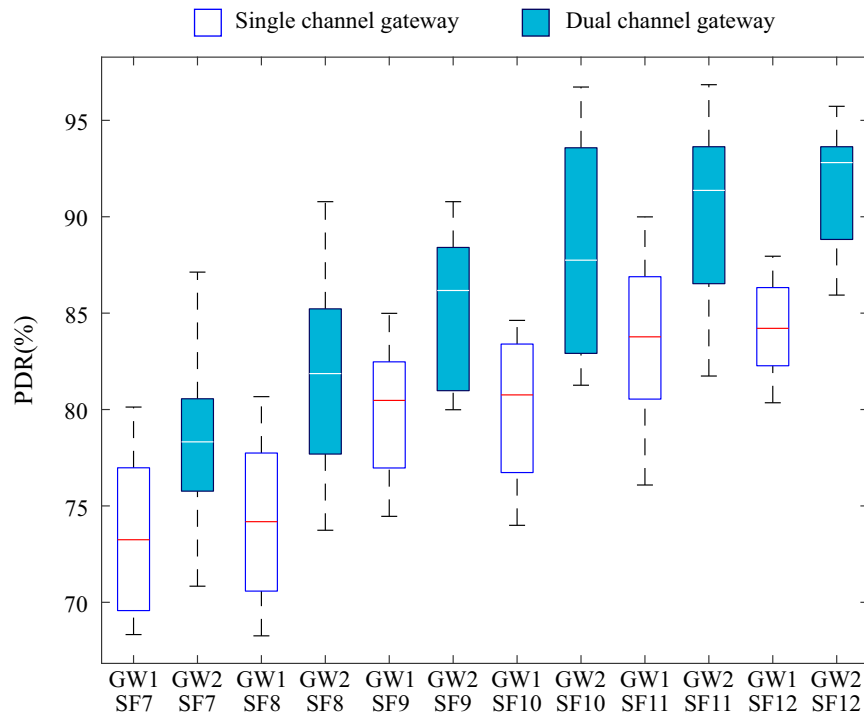
6.3.4 Resultados

Esta seção apresenta a análise do desempenho do *gateway* de canal duplo desenvolvido, com base nos cenários experimentais descritos anteriormente. O objetivo é demonstrar que a solução proposta, de baixo custo, oferece desempenho satisfatório em redes IoT de pequeno e médio porte.

Para avaliar o desempenho da rede em cada cenário, foi utilizada a métrica PDR, que representa a razão entre o número de pacotes recebidos com sucesso e o número total de pacotes transmitidos. Os experimentos foram conduzidos durante sete dias consecutivos em cada um dos três cenários avaliados, totalizando 21 dias de coleta de dados experimentais. Em cada cenário, cada dispositivo transmitiu, em média 6.270 pacotes.

A Figura 37 apresenta a distribuição do PDR para o primeiro cenário, segmentado por SF (7 a 12) para os dois tipos de *gateway*. O GW1, correspondente ao *gateway* de canal único, e GW2, correspondente ao *gateway* de canal duplo.

Figura 37 – PDR segmentado por SF para o primeiro cenário.



Fonte: Própria (2025).

Para todos os valores de SF, o *gateway* de canal duplo apresentou valores de PDR superiores aos obtidos com o *gateway* de canal único, o que evidencia a maior eficiência da comunicação quando múltiplos canais de recepção estão disponíveis.

O desempenho do GW1 nos SFs 7 e 8 (Figura 37) apresenta medianas mais baixas, o que está diretamente relacionado às limitações do *gateway* de canal único e à utilização do protocolo LoRaWAN baseado no ALOHA puro. Nesse contexto, a ausência de coordenação entre os dispositivos pode levar a transmissões simultâneas, resultando em colisões.

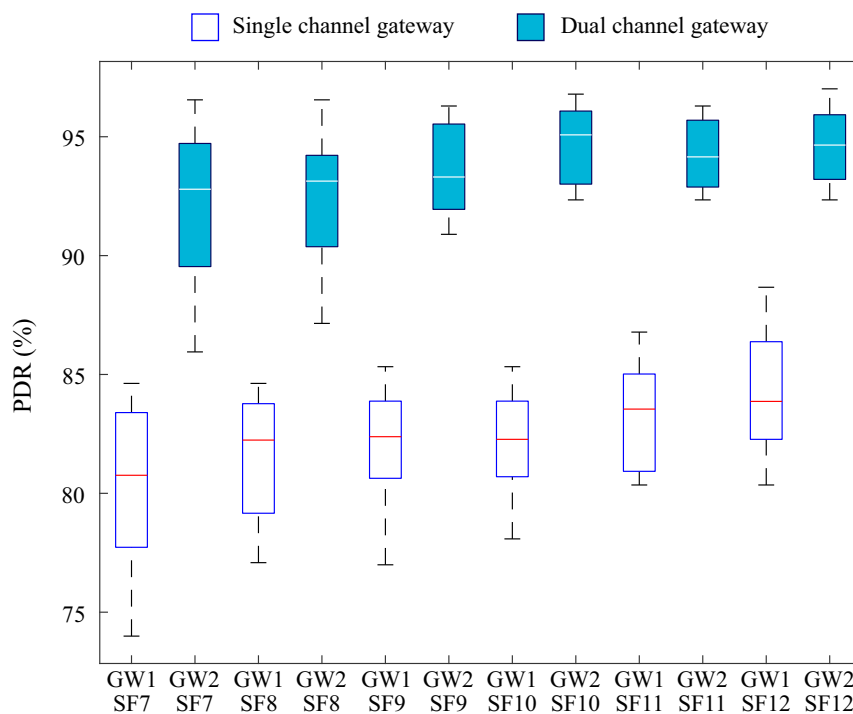
Observa-se que, à medida que o SF aumenta, o PDR melhora em ambos os *gateways*. No entanto, o *gateway* de canal único mantém uma maior variabilidade. Por outro lado, o *gateway* de canal duplo passa a apresentar maior consistência a partir do SF9, mantendo o PDR constantemente acima de 85%.

O uso de fatores de espalhamento elevados, como SF11 ou SF12, está associado a um aumento no PDR devido à maior robustez do sinal em longas distâncias. No entanto, esse ganho ocorre ao custo de um tempo de transmissão mais longo, o que eleva o consumo de energia. Ademais, em cenários com alta densidade de dispositivos, o maior tempo de ocupação do canal associado a SFs altos pode intensificar a ocorrência de colisões, comprometendo o desempenho do PDR (ZORBAS, 2024).

A utilização de um *gateway* de canal duplo oferece o benefício de distribuir os dispositivos entre diferentes canais, reduzindo a densidade de transmissões concorrentes em um mesmo canal. Isso contribui para a diminuição do tempo médio de ocupação do meio por canal e, conseqüentemente, reduz a probabilidade de colisões. Em contrapartida, essa abordagem implica um aumento no custo do *hardware*, devido à necessidade de suportar múltiplos canais simultaneamente, quando comparada a *gateways* de canal único.

O segundo cenário apresenta um PDR mais concentrado em comparação ao primeiro, indicando uma melhoria na taxa de entrega de pacotes. A Figura 38 apresenta os valores do PDR para os SFs de 7 a 12, considerando ambos os *gateways*. Observa-se um aumento geral no PDR em comparação ao primeiro cenário, resultado da aplicação do mecanismo CAD, que contribui para evitar transmissões simultâneas e minimizar colisões.

Figura 38 – PDR segmentado por SF para o segundo cenário.



Fonte: Própria (2025).

A adoção do mecanismo CAD resultou em uma melhora significativa no desempenho do *gateway* de canal único, com as medianas do PDR superando 80% em todos os SFs avaliados. No caso do *gateway* de canal duplo, a capacidade intrínseca de operar com múltiplos canais já proporciona maior eficiência na recepção dos pacotes. Quando combinada com a aplicação do CAD, esse *design* apresentou um desempenho superior, evidenciado por medianas do PDR superiores a 92%, destacando-se como um *design* de *hardware* mais eficiente em comparação ao de canal único. Esse comportamento está de acordo com a literatura, que demonstra que a utilização do CAD permite reduzir significativamente a ocorrência de colisões, além de contribuir para a economia de energia dos dispositivos finais, uma vez que evita retransmissões desnecessárias e reduz o tempo de

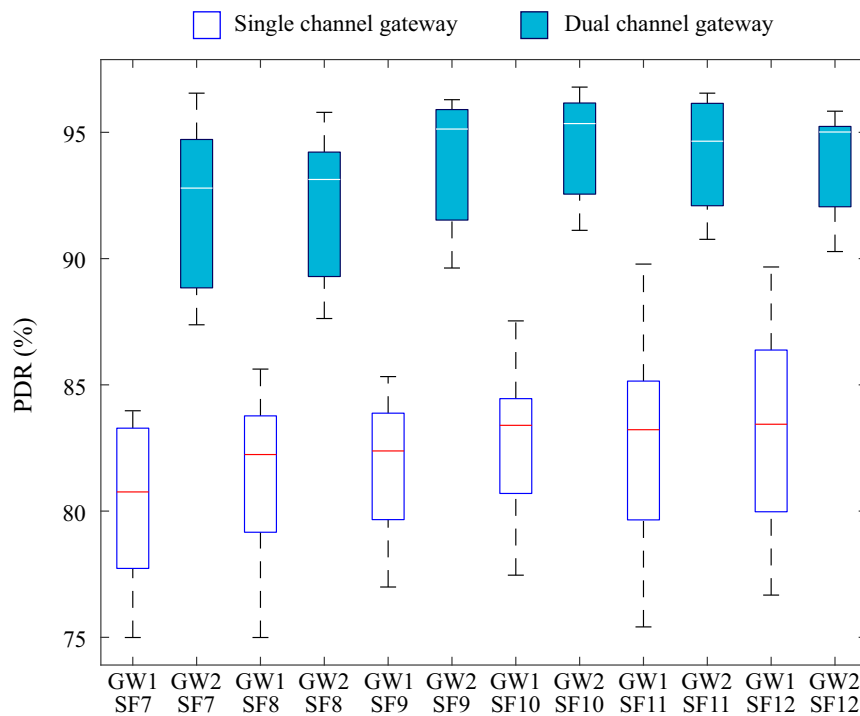
ativação do transceptor durante as tentativas de envio de pacotes (ZHANG et al., 2024).

No terceiro cenário, foi introduzido um mecanismo de alternância de canal no *uplink*, utilizando os 64 canais disponíveis, com o objetivo de otimizar o uso do espectro e reduzir a concentração de tráfego em poucos canais. Esta abordagem foi combinada com o uso de CAD, já presente no segundo cenário.

A Figura 39 apresenta uma maior dispersão nos valores de PDR entre os dispositivos, em comparação com o segundo cenário, onde os dois canais do *gateway* eram fixados no início de forma aleatória.

A utilização dos 64 canais disponíveis para *uplink*, em ambientes com redes LoRa coexistentes, aumenta a probabilidade do *gateway* de canal duplo estar operando em um canal com maior volume de tráfego em determinados momentos. Isso ocorre quando um ou mais dispositivos pertencentes a outras redes transmitem simultaneamente no mesmo canal em que o *gateway* está sintonizado, elevando a chance de colisões e impactando negativamente a recepção dos pacotes.

Figura 39 – PDR segmentado por SF para o terceiro cenário.



Fonte: Própria (2025).

Mesmo com um leve aumento na dispersão dos dados, as medianas do PDR para o *gateway* de canal duplo apresentaram melhora em comparação aos cenários anteriores, atingindo 96% para SF9, 95% para SF10, 94% para SF11 e 95% para SF12. A estratégia de troca de canais também foi aplicada ao *gateway* de canal único, que em comparação com o segundo cenário, a mediana do PDR permaneceu entre 80% e 85%.

Observa-se, ainda, que em ambos os cenários, o aumento do SF proporcionou uma melhoria no PDR. Isso ocorre porque o número de símbolos transmitidos por *bit* aumenta exponencialmente com o crescimento do SF, resultando em um tempo de transmissão mais longo para cada pacote (SEMTECH, 2020). Essa expansão torna o sinal mais robusto contra interferências e ruídos, pois proporciona uma maior distância entre símbolos.

Embora o aumento do SF torne o sinal mais robusto e menos suscetível a interferências, isso não garante uma melhora significativa no desempenho do *gateway* de canal único. Em SFs mais altos, o tempo de ocupação do canal aumenta consideravelmente, fazendo com que o *gateway* permaneça mais tempo dedicado à recepção de um único pacote. Como resultado, o canal pode estar ocupado no momento em que outros dispositivos tentam transmitir, aumentando a probabilidade de colisões e perdas.

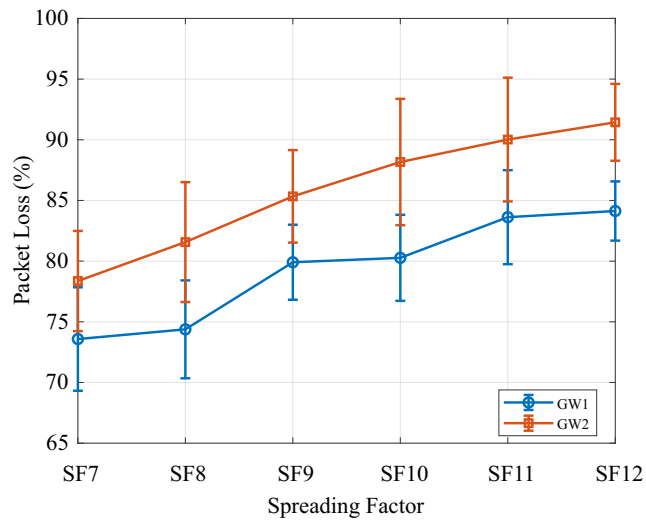
Esse fenômeno também ocorre no *gateway* de canal duplo, no entanto, o fato de operar simultaneamente em dois canais oferece uma vantagem significativa em relação ao *gateway* de canal único. A capacidade de recepção paralela permite uma melhor distribuição das transmissões entre os canais disponíveis, reduzindo o risco de colisões mesmo em situações de elevado tempo de ocupação do canal. Quando associado ao mecanismo CAD, esse recurso resulta em uma melhora no desempenho da rede, como evidenciado nos resultados apresentados.

Com os resultados obtidos nos diferentes cenários também foi realizado a análise de variância de duas vias (*Two-Way* ANOVA). Os resultados estão ilustrados na Figura 40, que apresenta os gráficos de interação entre o *Gateway* e SF nos três cenários analisados: Cenário 1 (a), Cenário 2 (b) e Cenário 3 (c).

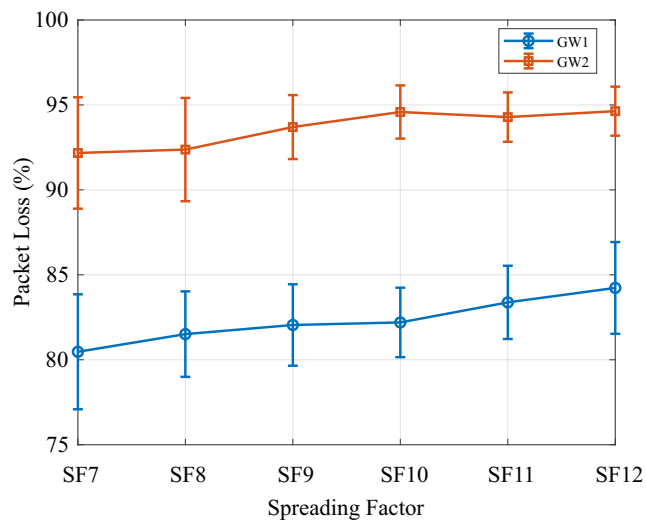
Em todos os cenários, o fator *Gateway* demonstrou efeito estatisticamente significativo sobre a perda de pacotes para o primeiro cenário $F(1,228) = 154.77$, $p < 0.0001$, segundo cenário $F(1,228) = 1321.11$, $p < 0.0001$ e terceiro respectivamente $F(1,228) = 916.17$, $p < 0.0001$). Esses resultados indicam que o *gateway* de canal duplo apresentou desempenho consistentemente superior ao *gateway* de canal único em termos de taxa de entrega de pacotes, independentemente do SF utilizado.

O fator SF também exerceu influência significativa em todos os casos para o primeiro cenário $F(5,228) = 55.02$, $p < 0.0001$, segundo cenário $F(5,228) = 9.41$, $p < 0.0001$ e terceira $F(5,228) = 4.61$, $p = 0.0005$. A perda de pacotes tende a aumentar com o incremento do SF, o que é coerente com a modulação LoRa, na qual fatores de espalhamento mais altos resultam em maior tempo de transmissão, devido ao aumento da quantidade de símbolos, podendo deixar de receber outros pacotes.

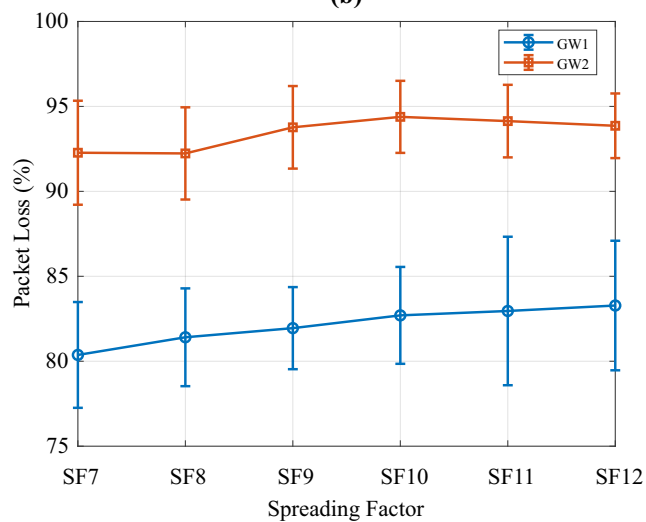
Figura 40 – Gráficos de interação entre o tipo de *gateway*, canal único (GW1) e canal duplo (GW2), e o fator de espalhamento (SF7 a SF12) sobre a taxa de perda de pacotes (%) nos três cenários experimentais: (a) Cenário 1, (b) Cenário 2 e (c) Cenário 3.



(a)



(b)



(c)

Por outro lado, a interação entre os fatores *Gateway* \times SF não foi estatisticamente significativa em nenhum dos cenários. Para a primeira $p = 0.4962$, segunda $p = 0.4835$ e terceira $p = 0.8716$, indicando que o efeito do tipo de *gateway* é consistente ao longo dos diferentes valores de SF. O desempenho superior do *gateway* de canal duplo em relação ao *gateway* de único canal é mantido independentemente do fator de espalhamento utilizado. Essa ausência de interação também é visualmente confirmada nos gráficos da Figura 40, nos quais as linhas representando *gateway* de canal único (GW1) e o *gateway* de canal duplo (GW2) mantêm-se aproximadamente paralelas ao longo dos diferentes valores de SF.

Os resultados apresenta melhor desempenho para o *gateway* de canal duplo em todos os cenários, além de demonstrarem que, embora o aumento do SF impacte negativamente a taxa de entrega de pacotes, essa influência não altera a vantagem relativa entre os dois *gateways* avaliados.

A Tabela 13 apresenta a média de tentativas de detecção de atividade no canal necessárias antes do envio de cada pacote para o *gateway* de canal duplo, mostrando uma média de 1,34 vezes para o segundo cenário e 1,51 para o terceiro. Observa-se que, no terceiro cenário, alguns dispositivos exigiram um número maior de tentativas de CAD.

Tabela 13 – Média do CAD.

Dispositivos	Cenário 2	Cenário 3
Node 01	1,4	1,3
Node 02	1,3	1,3
Node 03	1,3	2,0
Node 04	1,3	1,3
Node 05	1,4	1,4
Node 06	1,3	1,3
Node 07	1,4	1,4
Node 08	1,4	2,0
Node 09	1,4	2,0
Node 10	1,4	1,4
Node 11	1,3	1,3
Node 12	1,3	1,4
Node 13	1,3	1,3
Node 14	1,4	2,0
Node 15	1,3	1,3
Node 16	1,3	1,3
Node 17	1,3	2,0
Node 18	1,3	1,3
Node 19	1,3	1,4
Node 20	1,4	1,5
Média	1,34	1,51

Esse comportamento está relacionado à estratégia de alternância de canal adotada

pelo *gateway* de canal duplo, que muda para um novo canal a cada minuto com base na seleção de canais proposta na Seção 6.1. Como consequência, o *gateway* pode permanecer, por determinados intervalos, operando em canais com maior ocupação por redes coexistentes, o que eleva a probabilidade do canal estar ocupado realizando mais verificações antes de transmitir.

O experimento foi conduzido nas dependências de uma universidade, onde coexistem projetos de pesquisa que fazem uso da tecnologia LoRa. É plausível que algumas dessas redes empreguem *gateways* de canal único fixados em canais específicos. Assim, quando o *gateway* de canal duplo alterna para um canal coincidente com o de uma dessas redes, observa-se um aumento na quantidade de tentativas de CAD, como visto no terceiro cenário, em virtude da elevação da ocupação momentânea do canal.

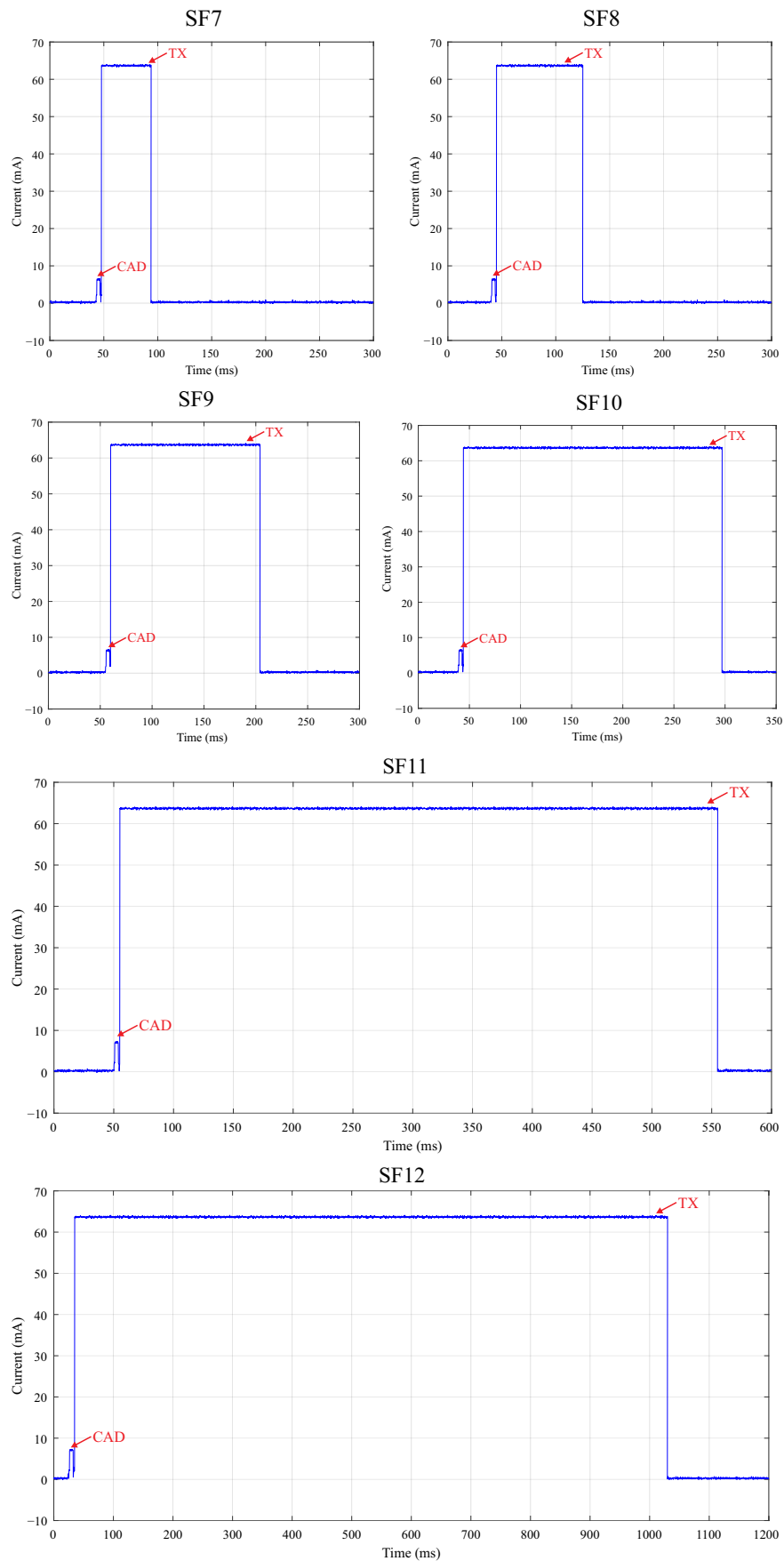
No estudo de caso, os dispositivos foram configurados para transmitir com uma potência de 10 dBm, resultando em um consumo de corrente aproximado de 64 mA durante a transmissão e 6,19 mA durante a execução do CAD que precede a transmissão, conforme ilustrado na Figura 41.

A medida que o SF aumenta, o transceptor permanece ativo por períodos mais longos durante a transmissão, resultando em maior consumo de energia, sendo um fator crítico para dispositivos alimentados por bateria. No contexto do uso do *gateway* de canal duplo em ambientes com redes LoRa coexistentes, a alternância de canais, conforme implementado no terceiro cenário experimental, que visa otimizar a utilização do espectro disponível. Apesar da necessidade de realizar um número maior de verificações CAD antes da transmissão, o consumo energético e o tempo de duração do procedimento de detecção de canal são significativamente inferiores, como visto na Figura 41, quando comparados à transmissão propriamente dita.

Nos estudos de casos vistos, uma aplicação com 20 dispositivos operando em ambientes com redes LoRa coexistentes, a utilização de um *gateway* de canal único pode limitar a taxa de entrega de pacotes a cerca de 80% nos SFs mais baixos.

Embora o aumento do SF possa contribuir para a melhoria da taxa de entrega de pacotes, mesmo em dispositivos localizados próximos ao *gateway*, essa estratégia implica em um tempo de ocupação do canal significativamente maior. Como evidenciado na Figura 41, uma transmissão utilizando SF12 apresenta duração próxima de um segundo, o que impacta diretamente no consumo energético do dispositivo. Utilizando o design do *gateway* proposto mesmo para SFs baixo é possível obter uma taxa de entrega média acima de 90%.

Figura 41 – Consumo de corrente durante a transmissão para os fatores de espalhamento SF7 a SF12.



Fonte: Própria (2025).

O uso do CAD contribui para reduzir colisões, mas não há garantia de que o canal permanecerá desocupado no exato momento em que a transmissão se inicia. Essa limitação também se torna mais evidente em *gateways* de canal único, nos quais a probabilidade de colisão é maior, especialmente a medida que cresce o número de dispositivos na rede ou em ambientes com redes LoRa coexistentes.

O *design* do *gateway* de canal duplo proposto neste trabalho apresenta-se como uma alternativa eficiente para redes de pequeno e médio porte, ao permitir a recepção simultânea de pacotes em múltiplos canais mantendo o baixo custo de implementação e melhorando a taxa de entrega de pacotes e auxiliando no baixo consumo energético dos dispositivos da rede.

Capítulo 7

Conclusão

Este trabalho apresentou contribuições relevantes para a área de IoT, contemplando a descrição de plataformas de prototipagem, a caracterização dos principais componentes dos elementos de IoT, bem como uma síntese da tecnologia LoRa e do protocolo LoRaWAN. Além disso, foram discutidos trabalhos encontrados na literatura que podem vir a se beneficiar da proposta aqui desenvolvida e a lacuna encontrada. Foi proposto e avaliado o *design* de um *gateway* de canal duplo de baixo custo voltado a redes LoRaWAN de pequeno e médio porte, com foco na mitigação de limitações associadas ao uso do protocolo ALOHA em *gateways* de canal único.

No decorrer deste trabalho, foram enfrentados desafios no desenvolvimento do *gateway*, especialmente relacionados à dificuldade de encontrar componentes em estoque para a construção do primeiro protótipo, em um período em que o mercado ainda se recuperava da crise de semicondutores provocada pela pandemia. Soma-se a isso a complexidade do processo de importação de componentes junto a distribuidores confiáveis, o que representou um obstáculo adicional ao andamento do projeto.

Desse modo, o projeto precisou ser adaptado em diferentes etapas, utilizando componentes de maior disponibilidade no mercado. Entre os itens que apresentaram maior dificuldade de aquisição destacam-se os *switches* de RF, os reguladores de tensão com alta eficiência energética e baixo I_Q , além de memórias RAM compatíveis com o ESP32-S3FH4R2.

Apesar dessas limitações, foi possível consolidar a principal contribuição deste trabalho que é a entrega de um modelo e projeto de *hardware* detalhado, constituindo uma referência para pesquisadores e desenvolvedores da área. A arquitetura proposta integra múltiplos transceptores LoRa para operação em múltiplos canais, incluindo uma unidade dedicada ao CAD e uma unidade de *timestamp* integrando um RTC+TCXO para garantir a sincronização, se necessário, entre os dispositivos, mesmo sem conexão com a Internet.

A proposta também apresenta o uso de um conjunto de antenas associado a

um transceptor exclusivo para o processo de CAD. Essa configuração assegura que os dois canais principais permaneçam integralmente disponíveis para a recepção de dados, enquanto o canal dedicado ao CAD monitora continuamente a ocupação do espectro. Dessa forma, garante-se que as transmissões de retorno para os dispositivos da rede ocorram em canais livres, minimizando interferências e reduzindo colisões.

Para o funcionamento do *gateway* de canal duplo, foi necessária uma adaptação do protocolo LoRaWAN em razão das limitações de canais quando comparado a *gateways* de alto desempenho. Nessa adaptação, propôs-se substituir o mecanismo de salto de frequências pseudoaleatório por uma seleção determinística de canais, fundamentada em um *timestamp*. Para validar a proposta, foram realizadas simulações no ambiente LoRaWAN-SIM, comparando a eficiência entre um *gateway* de canal único e a versão proposta do *gateway* de canal duplo.

A simulação do *gateway* de canal duplo apresentou alto desempenho ao operar com 40 dispositivos conectados, mantendo uma taxa de PDR próxima de 90%. Além disso, o *gateway* conseguiu sustentar valores em torno de 80% de PDR mesmo com aproximadamente 80 dispositivos conectados, quando operando nos fatores de espalhamento SF9 e SF10.

O estudo de caso, conduzido em ambiente real com a presença de redes LoRa coexistentes, contemplou três cenários distintos. No primeiro, foi realizada a comparação entre um *gateway* de canal único e o *gateway* de canal duplo operando de forma balanceada entre dois canais, com alocação dinâmica dos dispositivos. No segundo cenário, avaliou-se o impacto do uso do mecanismo CAD antes de cada transmissão. Por fim, no terceiro cenário, investigou-se uma política de comutação dinâmica de canais, baseada em sincronização temporal entre dispositivos e *gateway*, explorando de forma mais eficiente a diversidade espectral disponível em redes LoRaWAN.

Os resultados evidenciou ganhos concretos por meio da avaliação em três cenários distintos. No primeiro cenário, com alocação estática de canais e sem mecanismos adicionais, o *gateway* de canal duplo apresentou uma PDR estável acima de 85% a partir de SF9. No segundo cenário, com a introdução do CAD, observou-se uma maior concentração dos valores de PDR, com resultados superiores a 92%, demonstrando redução de colisões e aumento da confiabilidade. Já no terceiro cenário, que adotou a troca dinâmica de canal com base em *timestamp*, o *gateway* de canal duplo atingiu taxas de entrega superiores a 94% para fatores de espalhamento entre SF7 e SF12, superando de forma consistente o *gateway* de canal único, cuja PDR permaneceu na faixa de 80% a 85%.

Os experimentos também demonstraram que o aumento do SF contribui para uma maior robustez do sinal e melhora na taxa de entrega de pacotes. Contudo, esse benefício está associado a um maior tempo de ocupação do canal e a um aumento no consumo energético, aspectos críticos para dispositivos alimentados por bateria. Em ambientes com a presença de redes LoRa coexistentes, o *gateway* de canal duplo proposto mostrou-se

eficaz, especialmente por sua capacidade de executar do CAD antes das transmissões de *downlink*. Esse mecanismo reduz o risco de colisões com pacotes já em trânsito em diferentes SFs, promovendo maior confiabilidade na comunicação. Além disso, tais características contribuem diretamente para a economia de energia nos dispositivos finais, especialmente aqueles que operam em SFs mais elevados.

Os resultados obtidos validam a proposta como uma alternativa técnica e economicamente viável (ver Tabela 11) para ampliar a escalabilidade, confiabilidade e eficiência energética em redes IoT operando com restrições orçamentárias, especialmente em contextos urbanos ou rurais densamente povoados por dispositivos.

Como trabalhos futuros, propõe-se investigar a integração do *gateway* com algoritmos adaptativos de alocação de canais e mecanismos de coordenação de transmissão, visando à otimização do protocolo de acesso ao meio. Nesse contexto, pretende-se explorar abordagens baseadas em CSMA e TDMA, de modo a aprimorar o desempenho e a eficiência da rede em ambientes com múltiplas redes coexistentes, considerando as limitações impostas pelo *hardware* de apenas dois canais. Pretende-se, ainda, aprimorar a visualização dos dados recebidos pelos dispositivos conectados, por meio de uma interface gráfica baseada em ambiente web, com objetivo de auxiliar outros pesquisadores a utilizar o *gateway* proposto. Adicionalmente, está sendo realizado o processo de uma patente referente ao *gateway* de canal duplo desenvolvido, reforçando a originalidade e relevância da contribuição apresentada.

REFERÊNCIAS

- ABDELFADEEL, K. Q. et al. *free* fine-grained scheduling for reliable and energy-efficient data collection in lorawan. *IEEE Internet of Things Journal*, v. 7, n. 1, p. 669–683, 2020.
- AFHAMISIS, M.; PALATTELLA, M. R. Salsa: A scheduling algorithm for lora to leo satellites. *IEEE Access*, v. 10, p. 11608–11615, 2022.
- AKHTER, F. et al. Iot enabled intelligent sensor node for smart city: pedestrian counting and ambient monitoring. *Sensors*, MDPI, v. 19, n. 15, p. 3374, 2019.
- ALI, Z. et al. Performance evaluation of lorawan for green internet of things. *IEEE Access*, v. 7, p. 164102–164112, 2019.
- ASTERIOU, V. et al. Lorawan-based adaptive macs for event response applications. *IEEE Access*, v. 10, p. 97465–97480, 2022.
- AUGUSTIN, A. et al. A study of lora: Long range & low power networks for the internet of things. *Sensors*, MDPI, v. 16, n. 9, p. 1466, 2016.
- BARAI, S.; BISWAS, D.; SAU, B. Estimate distance measurement using nodemcu esp8266 based on rssi technique. In: *2017 IEEE Conference on Antenna Measurements & Applications (CAMA)*. [S.l.: s.n.], 2017. p. 170–173.
- BINE, W. I. S. *Um elemento para internet das coisas*. Dissertação (Mestrado) — Programa de Pós-Graduação em Ciência da Computação, Centro de Tecnologia Departamento de Informática, Universidade Estadual de Maringá - UEM, Maringá, 2021.
- BINE, W. I. S.; RUIZ, L. Manna-wui: Um nó sensor versátil para a internet das coisas. In: SBC. *Anais do XIX Workshop em Desempenho de Sistemas Computacionais e de Comunicação*. [S.l.], 2020. p. 109–120.
- CASS, S. *The Arduino popularity contest*. 2018. [Último acesso em 15 de setembro de 2022]. Disponível em: <<https://spectrum.ieee.org/tech-history/silicon-revolution/chip-hall-of-fame-intel-4004-microprocessor>>.
- CATTANI, M.; BOANO, C. A.; RÖMER, K. An experimental evaluation of the reliability of lora long-range low-power wireless communication. *Journal of Sensor and Actuator Networks*, MDPI, v. 6, n. 2, p. 7, 2017.
- CHASSERAT, L.; ACCETTURA, N.; BERTHOU, P. Short: Achieving energy efficiency in dense lorawans through tdma. In: *2020 IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. [S.l.: s.n.], 2020. p. 26–29.

- CHAUDHARI, S. S.; BHOLE, V. Y. Solid Waste Collection as a Service using IoT-Solution for Smart Cities. In: *2018 International Conference on Smart City and Emerging Technology (ICSCET)*. [S.l.: s.n.], 2018. p. 1–5.
- CHEN, C.-H.; LIN, M.-Y.; LIU, C.-C. Edge computing gateway of the industrial internet of things using multiple collaborative microcontrollers. *IEEE Network*, v. 32, n. 1, p. 24–32, 2018.
- CHEN, C.-H.; LIU, C.-T. A 3.5-tier container-based edge computing architecture. *Computers & Electrical Engineering*, v. 93, p. 107227, 2021. ISSN 0045-7906.
- CHEN, C.-W.; FAYED, A. A low-power dual-frequency simo buck converter topology with fully-integrated outputs and fast dynamic operation in 45 nm cmos. *IEEE Journal of Solid-State Circuits*, IEEE, v. 50, n. 9, p. 2161–2173, 2015.
- CHENG, Y. et al. Design of Air Quality Monitoring System Based on NB-IoT. In: *2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*. [S.l.: s.n.], 2019. p. 385–388.
- CHINCHILLA-ROMERO, N. et al. Collision avoidance resource allocation for lorawan. *Sensors*, MDPI, v. 21, n. 4, p. 1218, 2021.
- CHOUDHARY, V. et al. AirQ: A Smart IoT Platform for Air Quality Monitoring. In: *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*. [S.l.: s.n.], 2020. p. 1–2.
- CITONI, B. et al. Internet of Things and LoRaWAN-Enabled Future Smart Farming. *IEEE Internet of Things Magazine*, v. 2, n. 4, p. 14–19, 2019.
- CIZESKI, E. M. A. Uma aplicação IoT para o monitoramento da saúde no dia a dia. In: (Graduação) - Curso de Ciência da Computação, Centro de Tecnologia Departamento de Informática. *Universidade Estadual de Maringá - UEM*. Maringá, 2022. p. 43.
- DONNO, M. D.; TANGE, K.; DRAGONI, N. Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *Ieee Access*, Ieee, v. 7, p. 150936–150948, 2019.
- EDWARD, P. et al. On the Coexistence of LoRa- and Interleaved Chirp Spreading LoRa-Based Modulations. In: *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. [S.l.: s.n.], 2019. p. 1–6.
- ELDEEB, E.; SHEHAB, M.; ALVES, H. A learning-based fast uplink grant for massive iot via support vector machines and long short-term memory. *IEEE Internet of Things Journal*, v. 9, n. 5, p. 3889–3898, 2022.
- ELHEBEARY, M.; YANG, C.-K. K. A 92%-efficiency battery powered hybrid dc-dc converter for iot applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, IEEE, v. 67, n. 10, p. 3342–3351, 2020.
- EMBARCADOS. *Pesquisa sobre o mercado brasileiro de sistemas embarcados e IoT 2021*. 2021. [Último acesso em 02 de outubro de 2022]. Disponível em: <<https://embarcados.com.br/relatorio-da-pesquisa-sobre-o-mercado-brasileiro-de-sistemas-embarcados-e-iot-2021/>>.

ERGUN, K. et al. Dynamic Reliability Management of Multi-Gateway IoT Edge Computing Systems. *IEEE Internet of Things Journal*, p. 1–1, 2022.

Espressif Systems. *ESP32-S3 Series - 2.4 GHz Wi-Fi + Bluetooth LE SoC*. 2022. Online, accessed 07/06/22. https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf.

Espressif Systems. *ESP32-S3 Series - Hardware Design Guidelines*. 2022. Online, accessed 07/06/22. https://www.espressif.com/sites/default/files/documentation/esp32-s3_hardware_design_guidelines_en.pdf.

FROIZ-MÍGUEZ, I. et al. Design, implementation, and empirical validation of an iot smart irrigation system for fog computing applications based on lora and lorawan sensor nodes. *Sensors*, MDPI, v. 20, n. 23, p. 6865, 2020.

GAILLARD, G.; PHAM, C. Canl lora: Collision avoidance by neighbor listening for dense lora networks. In: IEEE. *2023 IEEE Symposium on Computers and Communications (ISCC)*. [S.l.], 2023. p. 1293–1298.

GAMAGE, A. et al. Lmac: Efficient carrier-sense multiple access for lora. *ACM Trans. Sen. Netw.*, Association for Computing Machinery, New York, NY, USA, v. 19, n. 2, fev. 2023. ISSN 1550-4859. Disponível em: <<https://doi-org.ez27.periodicos.capes.gov.br/10.1145/3564530>>.

Gliwa, P. *Microprocessor Technology Basics*. Springer international publishing. [S.l.: s.n.], 2021. v. 1.

GOUDARZI, M. et al. An Application Placement Technique for Concurrent IoT Applications in Edge and Fog Computing Environments. *IEEE Transactions on Mobile Computing*, v. 20, n. 4, p. 1298–1311, 2021.

HASSIJA, V. et al. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access*, v. 7, p. 82721–82743, 2019.

HERRERÍA-ALONSO, S. et al. Enhancing lorawan scalability with longest first slotted csma. *Computer Networks*, Elsevier, v. 216, p. 109252, 2022.

HEUSSE, M.; CAILLOUET, C.; DUDA, A. Performance of unslotted ALOHA with capture and multiple collisions in LoRaWAN. *IEEE Internet of Things Journal*, IEEE, v. 10, n. 20, p. 17824–17838, 2023.

HLAING, W. et al. Implementation of wifi-based single phase smart meter for internet of things (iot). In: *2017 International Electrical Engineering Congress (iEECON)*. [S.l.: s.n.], 2017. p. 1–4.

IKPEHAI, A. et al. Low-power wide area network technologies for internet-of-things: A comparative review. *IEEE Internet of Things Journal*, v. 6, n. 2, p. 2225–2240, 2019.

ILLIAS, H. A. et al. Iot-based hybrid renewable energy harvesting system from water flow. In: IEEE. *2020 IEEE International Conference on Power and Energy (PECon)*. [S.l.], 2020. p. 204–208.

- ISLAM, N.; RAY, B.; PASANDIDEH, F. Iot based smart farming: Are the lpwan technologies suitable for remote communication? In: *2020 IEEE International Conference on Smart Internet of Things (SmartIoT)*. [S.l.: s.n.], 2020. p. 270–276.
- JAVED, S.; ZORBAS, D. Downlink traffic demand-based gateway activation in lorawan. In: *2023 IEEE Symposium on Computers and Communications (ISCC)*. [S.l.: s.n.], 2023. p. 669–674.
- JIANG, X. et al. Hybrid low-power wide-area mesh network for iot applications. *IEEE Internet of Things Journal*, v. 8, n. 2, p. 901–915, 2021.
- JIMÉNEZ, M.; PALOMERA, R.; COUVERTIER, I. *Embedded Peripherals*. Springer new york. [S.l.: s.n.], 2014.
- KANSAL, P.; KUMAR, M.; VERMA, O. P. Classification of resource management approaches in fog/edge paradigm and future research prospects: a systematic review. *The Journal of Supercomputing*, Springer, p. 1–60, 2022.
- KENNEDY, H. *Texas Instruments: Quiescent Current vs Shutdown Current for Load Switch Power Consumption*. 2016. Último acesso em 13 de setembro de 2022. Disponível em: <<http://www.ti.com/lit/an/slva757/slva757.pdf>>.
- KHAMPHROO, M. et al. Integrating micropython-based educational mobile robot with wireless network. In: IEEE. *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*. [S.l.], 2017. p. 1–6.
- KHAMPHROO, M. et al. Micropython-based educational mobile robot for computer coding learning. In: IEEE. *2017 8th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*. [S.l.], 2017. p. 1–6.
- KHAN, M. A. A. et al. Performance of slotted aloha for lora-esl based on adaptive backoff and intra slicing. In: *2022 6th International Conference on Communication and Information Systems (ICCIS)*. [S.l.: s.n.], 2022. p. 169–173.
- KHUTSOANE, O. et al. Watergrid-sense: A lora-based sensor node for industrial iot applications. *IEEE Sensors Journal*, v. 20, n. 5, p. 2722–2729, 2020.
- KONG, X. et al. RMGen: A Tri-Layer Vehicular Trajectory Data Generation Model Exploring Urban Region Division and Mobility Pattern. *IEEE Transactions on Vehicular Technology*, v. 71, n. 9, p. 9225–9238, 2022.
- KUMAR, A.; KUMARI, M.; GUPTA, H. Design and Analysis of IoT based Air Quality Monitoring System. In: *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*. [S.l.: s.n.], 2020. p. 242–245.
- LAVRIC, A.; PETRARIU, A. I. Lorawan communication protocol: The new era of iot. In: *2018 International Conference on Development and Application Systems (DAS)*. [S.l.: s.n.], 2018. p. 74–77.
- LE, T. B. et al. An Approach to Design a Multi-Protocol Gateway Device for Internet of Things System. In: *2021 8th NAFOSTED Conference on Information and Computer Science (NICS)*. [S.l.: s.n.], 2021. p. 452–457.

- LI, Y.; ZHENG, Y. Citywide Bike Usage Prediction in a Bike-Sharing System. *IEEE Transactions on Knowledge and Data Engineering*, v. 32, n. 6, p. 1079–1091, 2020.
- LIU, P. et al. A low-quiescent current off-chip capacitor-less ldo regulator with ugcc compensation. In: IEEE. *2019 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*. [S.l.], 2019. p. 1–3.
- LIU, Y. et al. Toward edge intelligence: Multiaccess edge computing for 5g and internet of things. *IEEE Internet of Things Journal*, v. 7, n. 8, p. 6722–6747, 2020.
- LOPEZ, V. et al. Evaluation of communication delays in lorawan networks for indoor emergency scenarios. *Journal of Communications Technology and Electronics*, Springer, v. 66, n. 2, p. S149–S158, 2021.
- LoRa Alliance. LoRaWAN 1.0. 2 Regional Parameters. *LoRa Alliance Technical Committee: Fremont, CA, USA*, 2020.
- LOSADA, M. et al. A Flexible Fog Computing Design for Low-Power Consumption and Low Latency Applications. *Electronics*, v. 10, n. 1, 2021.
- LOZANO, Á. et al. Smart waste collection system with low consumption lorawan nodes and route optimization. *Sensors*, MDPI, v. 18, n. 5, p. 1465, 2018.
- MAHMOOD, Z. *The Internet of Things in the Industrial Sector*. [S.l.]: Springer, 2019.
- MARAIS, J. M.; MALEKIAN, R.; ABU-MAHFOUZ, A. M. Evaluating the LoRaWAN Protocol Using a Permanent Outdoor Testbed. *IEEE Sensors Journal*, v. 19, n. 12, p. 4726–4733, 2019.
- MEYER-BAESE, U. *Embedded Microprocessor Systems Basics*. Springer international publishing. [S.l.: s.n.], 2021. v. 1.
- MISHRA, A. et al. Garbage management with Smart trash using IoT. In: *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*. [S.l.: s.n.], 2020. p. 1–6.
- MORABITO, R. et al. Reprint of : Legiot: A lightweight edge gateway for the internet of things. *Future Generation Computer Systems*, v. 92, p. 1157–1171, 2019. ISSN 0167-739X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0167739X18325123>>.
- MUKHIA, R.; JAYARATHNA, K. G. S.; LERTSINSRUBTAVEE, A. Performance evaluation of lorawan forest fire monitoring network in the wild. In: *Proceedings of the 18th Asian Internet Engineering Conference*. New York, NY, USA: Association for Computing Machinery, 2023. (AINTEC '23), p. 96104. ISBN 9798400709395. Disponível em: <<https://doi-org.ez27.periodicos.capes.gov.br/10.1145/3630590.3630602>>.
- NDIE, T. D. et al. A k-means based approach for optimal gateway deployment in lorawan-sim. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, v. 587 LNICST, p. 163–178, 2025.
- NETO, A. T. et al. Enhancing soil measurements with a multi-depth sensor for iot-based smart irrigation. In: *2020 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*. [S.l.: s.n.], 2020. p. 78–82.

- NGUYEN, D. C. et al. 6G Internet of Things: A comprehensive survey. *IEEE Internet of Things Journal*, IEEE, 2021.
- NOLAN, K. E.; GUIBENE, W.; KELLY, M. Y. An evaluation of low power wide area network technologies for the internet of things. In: IEEE. *2016 international wireless communications and mobile computing conference (IWCMC)*. [S.l.], 2016. p. 439–444.
- NUGRAHA, I. G. D.; ASHADI, E. Z.; EFENDI, A. M. Performance evaluation of collision avoidance for multi-node lora networks based on tdma and csma algorithm. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, Innovative Information Science & Technology Research Group (ISYOU), v. 15, n. 1, p. 53–74, 2024.
- PASOLINI, G. et al. Smart city pilot projects using lora and ieee802. 15.4 technologies. *Sensors*, MDPI, v. 18, n. 4, p. 1118, 2018.
- PHAM, C.; RAHIM, A.; COUSIN, P. Low-cost, long-range open iot for smarter rural african villages. In: IEEE. *2016 IEEE International Smart Cities Conference (ISC2)*. [S.l.], 2016. p. 1–6.
- POLAK, L.; MILOS, J. Performance analysis of LoRa in the 2.4 GHz ISM band: coexistence issues with Wi-Fi. *Telecommunication Systems*, Springer, v. 74, n. 3, p. 299–309, 2020.
- POLONELLI, T. et al. Slotted aloha on lorawan-design, analysis, and deployment. *Sensors*, MDPI, v. 19, n. 4, p. 838, 2019.
- QIU, T. et al. Edge computing in industrial internet of things: Architecture, advances and challenges. *IEEE Communications Surveys & Tutorials*, v. 22, n. 4, p. 2462–2488, 2020.
- RAJ, T. et al. Ambient parametric monitoring of farms using embedded iot & lora. In: *2019 IEEE Bombay Section Signature Conference (IBSSC)*. [S.l.: s.n.], 2019. p. 1–6.
- RAYES, A.; SALAM, S. *Internet of things from hype to reality*. [S.l.]: Springer, 2019.
- RAZA, U.; KULKARNI, P.; SOORIYABANDARA, M. Low power wide area networks: An overview. *IEEE Communications Surveys & Tutorials*, v. 19, n. 2, p. 855–873, 2017.
- RUOTSALAINEN, H.; ZHANG, J.; GREBENIUK, S. Experimental investigation on wireless key generation for low-power wide-area networks. *IEEE Internet of Things Journal*, v. 7, n. 3, p. 1745–1755, 2020.
- SEHRAWAT, D.; GILL, N. S. Smart sensors: Analysis of different types of iot sensors. In: *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*. [S.l.: s.n.], 2019. p. 523–528.
- Semtech. *AN1200.19 - SX127x Reference Design Overview*. 2014. Online, accessed 02/05/22. <https://docs.rs-online.com/d865/0900766b814e212c.pdf>.
- Semtech. *Datasheet - SX1261/2 Long Range, Low Power, sub-GHz RF Transceiver*. 2017. Online, accessed 02/05/22. https://www.mouser.com/datasheet/2/761/DS_SX1261-2_V1.1-1307803.pdf.
- Semtech. *Application Note: Reference Design Explanation SX1261/2*. 2018. Online, accessed 02/05/22. <https://www.semtech.com/products/wireless-rf/lora-connect/sx1262>.

- SEMTECH. *SX1276*. 2020. Accessed: Out. 19, 2024. Disponível em: <<https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>>.
- SHAO, C.; MUTA, O. Heterogeneous carrier-sense multiple access for improved energy fairness in lorawan. In: *2022 Tenth International Symposium on Computing and Networking (CANDAR)*. [S.l.: s.n.], 2022. p. 172–178.
- SHAYO, E.; MAFOLE, P.; MWAMBELA, A. A survey on time division multiple access scheduling algorithms for industrial networks. *SN Applied Sciences*, Springer, v. 2, n. 12, p. 1–10, 2020.
- SHI, W. et al. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, v. 3, n. 5, p. 637–646, 2016.
- SILVEIRA, J. D. F. et al. A new low-cost lorawan power switch for smart farm applications. In: *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.: s.n.], 2021. p. 3330–3335.
- SINHA, S. *State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally*. 2024. Online, accessed 09/01/25. <https://iot-analytics.com/number-connected-iot-devices/>.
- SINHA, S.; REDDY, G. P.; PARK, S.-H. Channel Selection Using Machine Learning. In: *2024 International Conference on Artificial Intelligence in Information and Communication (ICAIC)*. [S.l.: s.n.], 2024. p. 164–168.
- SOHAIL, H. et al. Intelligent Trash Bin (ITB) with Trash Collection Efficiency Optimization using IoT Sensing. In: *2019 8th International Conference on Information and Communication Technologies (ICICT)*. [S.l.: s.n.], 2019. p. 48–53.
- SPARKFUN; CODEBENDER. *Chip Hall of Fame: Intel 4004 Microprocessor*. 2015. [Último acesso em 11 de abril de 2021]. Disponível em: <<https://www.sparkfun.com/news/1982>>.
- SWAMY, S. N.; KOTA, S. R. An empirical study on system level aspects of internet of things (iot). *IEEE Access*, IEEE, v. 8, p. 188082–188134, 2020.
- TAIVALSAARI, A.; MIKKONEN, T. A taxonomy of iot client architectures. *IEEE software*, IEEE, v. 35, n. 3, p. 83–88, 2018.
- TANGE, K. et al. A Systematic Survey of Industrial Internet of Things Security: Requirements and Fog Computing Opportunities . *IEEE Communications Surveys and Tutorials*, v. 22, n. 4, p. 2489–2520, 2020.
- TRIANAFYLLOU, A.; ZORBAS, D.; SARIGIANNIDIS, P. Time-slotted lora mac with variable payload support. *Computer Communications*, v. 193, p. 146–154, 2022. ISSN 0140-3664. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0140366422002444>>.
- TRIGO, J. D. et al. Patient tracking in a multi-building, tunnel-connected hospital complex. *IEEE Sensors Journal*, v. 20, n. 23, p. 14453–14464, 2020.

- TUVAKOV, J.; PARK, K. On the fog node model for multi-purpose fog computing systems. In: IEEE. *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. [S.l.], 2018. p. 1211–1214.
- VANGELISTA, L.; CALVAGNO, G. On the channel activity detection in lorawan networks. *IEEE Open Journal of the Communications Society*, v. 5, p. 5598–5607, 2024.
- VEDAEI, S. S. et al. Covid-safe: An iot-based system for automated health monitoring and surveillance in post-pandemic life. *IEEE Access*, v. 8, p. 188538–188551, 2020.
- WANG, Q. et al. Learning Shared Mobility-Aware Knowledge for Multiple Urban Travel Demands. *IEEE Internet of Things Journal*, v. 9, n. 9, p. 7025–7035, 2022.
- WANG, Z.-H. et al. Deployment and evaluation of lora network configuration based on random forest. In: *2020 International Computer Symposium (ICS)*. [S.l.: s.n.], 2020. p. 262–265.
- WEGNER, P. *Global IoT market size grew 22% in 2021 these 16 factors affect the growth trajectory to 2027*. 2022. Online, accessed 23/02/22. <https://iot-analytics.com/iot-market-size/>.
- XU, J. et al. Discrete particle swarm optimization based antenna selection for mimo lora iot systems. In: *2019 Computing, Communications and IoT Applications (ComComAp)*. [S.l.: s.n.], 2019. p. 204–209.
- XUEWEN, H.; KAIHUA, C. Lora network communication protocol based on location and time planning. *Peer-to-Peer Networking and Applications*, Springer, v. 16, n. 4, p. 1596–1608, 2023.
- YANG, C. et al. Predictive modeling with vehicle sensor data and iot for injury prevention. In: IEEE. *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*. [S.l.], 2018. p. 293–298.
- YAPAR, G.; TUGCU, T.; ERMIS, O. Time-slotted aloha-based lorawan scheduling with aggregated acknowledgement approach. In: *2019 25th Conference of Open Innovations Association (FRUCT)*. [S.l.: s.n.], 2019. p. 383–390.
- YASCARIBAY, G. et al. Performance evaluation of communication systems used for internet of things in agriculture. *Agriculture*, v. 12, n. 6, 2022. ISSN 2077-0472. Disponível em: <<https://www.mdpi.com/2077-0472/12/6/786>>.
- YU, H.-C. et al. Development of miniaturized water quality monitoring system using wireless communication. *Sensors*, MDPI, v. 19, n. 17, p. 3758, 2019.
- YU, Y. et al. Adaptive multi-channels allocation in LoRa networks. *IEEE Access*, IEEE, v. 8, p. 214177–214189, 2020.
- YUAN, J.; LI, X. A reliable and lightweight trust computing mechanism for iot edge devices based on multi-source feedback information fusion. *IEEE Access*, v. 6, p. 23626–23638, 2018.
- ZHANG, H. et al. Lora network protocol design based on csma/ca with asynchronous downlink mode. In: SPIE. *Fourth International Conference on Digital Signal and Computer Communications (DSCC 2024)*. [S.l.], 2024. v. 13214, p. 201–206.

- ZORBAS, D. Improving lorawan downlink performance in the eu868 spectrum. *Computer Communications*, v. 195, p. 303–314, 2022. ISSN 0140-3664. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0140366422003401>>.
- ZORBAS, D. Downlink spreading factor selection in lorawan. *Computer Communications*, Elsevier, v. 215, p. 112–119, 2024.

Anexo A

Plataformas de IoT para prototipagem eletrônica

Tabela 14 – Principais placas de prototipagem eletrônica

Nome	MPU/MCU	Conectividade				Áudio/Vídeo		Sensores Integrados	Comercial	Importado/Nacional
		WiFi	Bluetooth	LoRa	Outro	Câmera	Microfones			
Arduino Uno	ATmega328P	Não	Não	Não	Não	Não	Não	Não	Sim	Importado/Nacional
Arduino Nano	ATmega328	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Arduino Nano 33 IoT	ATSAMD21G18	Sim	Sim	Não	Não	Não	Não	Sim	Sim	Importado
Arduino Mega	ATmega2560	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Arduino Micro	ATmega32U4	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Arduino Yun	ATmega32u4	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado
Arduino Lillypad	ATmega168V ou ATmega328V	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
NodeMCU	ESP8266	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado/Nacional
ESP32	ESP32-D0WDQ6-V3	Sim	Sim	Não	Não	Não	Não	Sim	Sim	Importado
ESP32 + LoRa	ESP32-D0WDQ6-V3	Sim	Sim	Sim	Não	Não	Não	Sim	Sim	Importado
BeagleBone Black	AM335x	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Raspberry PI 3	BCM2837	Sim	Sim	Não	Não	Sim	Não	Não	Sim	Importado
Raspberry Pi 4	BCM2711	Sim	Sim	Não	Não	Sim	Não	Não	Sim	Importado
Raspberry Pi Zero	BCM2835	Sim	Sim	Não	Não	Não	Não	Não	Sim	Importado
UDOO X86 ULTRA	Intel Pentium N3710	Não	Não	Não	Não	Sim	Sim	Não	Sim	Importado
FriendlyARM Mini2440	S3C2440	Não	Não	Não	Não	Sim	Sim	Não	Sim	Importado
Intel Galileo	Intel Quark X1000	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Cubieboard 4	Allwinner A80	Sim	Sim	Não	Não	Não	Não	Não	Sim	Importado
Z1	XC7Z020-1CLG400C	Não	Não	Não	Não	Sim	Sim	Não	Sim	Importado
T-Mote Sky	TI MSP430F1611	Não	Não	Não	Não	Não	Não	Umidade, Temperatura, Luz	Sim	Importado
Particle Photon	STM32F205	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado
Tessel 2	MT7620N	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado
Adafruit Feather M4	ATSAMD51J19	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
LightBlue Bean/Bean+	ATmega328	Sim	Sim	Não	Não	Não	Não	Acelerômetro, Temperatura	Sim	Importado
Intel Edison	Intel Atom e Intel Quark	Sim	Sim	Não	Não	Não	Não	Não	Sim	Importado
Intel Joule 570x	Intel Atom T5700	Sim	Sim	Não	Não	Não	Não	Não	Sim	Importado
Libelium Waspnote	ATmega1281	Não	Não	Não	Não	Não	Não	Temperatura, Acelerômetro	Sim	Importado
DragonBoard 410c	SnapDragon 410	Sim	Sim	Não	Não	Não	Não	GPS	Sim	Importado
Micro:bit	nRF52833	Não	Sim	Não	Não	Não	Sim	Acelerômetro, Compasso, Temperatura	Sim	Importado/Nacional
Banana Pi M5	S905X3	Sim	Não	Não	Não	Sim	Sim	Não	Sim	Importado
Lemaker Guitar	Actions S500	Sim	Não	Não	Não	Sim (baseboard)	Sim (baseboard)	Não	Sim	Importado
Arduino Pro-Mini	ATmega328	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
B-L475E-IOT01A Discovery kit for IoT	STM32L475VGT6	Sim	Sim	Não	NFC, FSK Sub-GHz (868 ou 915 MHz)	Não	Sim	Acelerômetro, Giroscópio, Temperatura, Umidade, Barômetro, Magnetômetro	Sim	Importado
B-L462E-CELL1 Cellular IoT Discovery kit	STM32L462REY6TR	Não	Não	Não	LTE Cat.M1/NB-IoT	Não	Não	Display OLED, Acelerômetro 3D, magnetômetro, Temperatura, umidade, pressão	Sim	Importado
STM32L562E-DK Discovery kit	STM32L562QEI6QU	Não	Sim	Não	Não	Não	Sim	Display LCD 1,54", Acelerômetro 3D, giroscópio 3D	Sim	Importado

Nome	MPU/MCU	Conectividade				Áudio/Vídeo		Sensores Integrados	Comercial	Importado/Nacional
		WiFi	Bluetooth	LoRa	Outro	Câmera	Microfones			
B-L4S5L-IOT01A Discovery kit for IoT node	STM32L4S5VIT6	Sim	Sim	Não	NFC	Não	Sim	Acelerômetro, Giroscópio, temperatura, umidade, pressão, magnetômetro	Sim	Importado
B-L072Z-LRWAN1 Discovery kit	STM32L072CZ	Sim	Não	Sim	Sigfox	Não	Não	Não	Sim	Importado
Giant Board	ATSAMA5D27C-D1G	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
STM32MP157C-DK2	STM32MP157	Sim	Sim	Não	Ethernet (IEEE-802.3ab)	Não	Sim	Display (480E800)	Sim	Importado
OSD32MP1-BRK	STM32MP157C	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
SimpleLink Wi-Fi CC3200 LaunchPad	CC3200	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado
Particle Boron LTE CAT-M1	nRF52840	Não	Sim	Não	LTE Cat.M1 / 3GPP	Não	Não	Não	Sim	Importado
NPi i.MX6ULL Dev Board	MCIMX6Y2CVM08AB	Não	Não	Não	Ethernet	Não	Não	Não	Sim	Importado
Onion Omega2	MT7688	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado
Arduino MKR NB 1500	SAMD21	Não	Não	Não	LTE Cat.M1/NB-IoT	Não	Não	Não	Sim	Importado/ Nacional
Teensy 4.1	ARM Cortex-M7	Não	Não	Não	Ethernet PHY	Não	Não	Temperatura	Sim	Importado
NVIDIA Jetson Nano	NVIDIA Maxwell e ARM Cortex-A57	Não	Não	Não	Gigabit Etherne	Sim	Sim	Não	Sim	Importado
Meadow F7	STM32F7	Sim	Sim	Não	Não	Não	Não	Não	Sim	Importado
ODROID boards XU4	Exynos5422	Sim	Não	Não	Gigabit Ethernet	Não	Não	Não	Sim	Importado
Rock Pi 4C	RK3399	Sim	Sim	Não	Gigabit Ethernet	Sim	Sim	Não	Sim	Importado
Adafruit FONA	SIM800L	Não	Não	Não	GSM 2G	Sim	Sim	Não	Sim	Importado
Particle Electron	SARA-U201/U260/U270	Não	Não	Não	LTE Cat.M1, 3G, 2G	Não	Não	Não	Sim	Importado
NanoPi NEO Plus2 (SBC)	Allwinner H5	Sim	Sim	Não	Gigabit Ethernet	Sim	Sim	Não	Sim	Importado
Raspberry Pi Pico	RP2040	Não	Não	Não	Não	Não	Não	Não	Sim	Importado/ Nacional
Udoo Neo	NXP i.MX 6SoloX	Sim	Sim	Não	Não	Sim	Sim	Acelerômetro, Giroscópio, Magnetômetro	Sim	Importado
Google Dev Board	NXP i.MX 8M	Sim	Sim	Não	Não	Sim	Sim	Não	Sim	Importado
Pycom Fipy	ESP32	Sim	Sim	Sim	Sigfox, LTE Cat.M1/NB-IoT	Não	Não	Não	Sim	Importado
Libelium Waspnote	ATmega1281	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Airboard	ATmega238P	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Azure Sphere	MT3620AN	Sim	Não	Não	Não	Não	Não	Acelerômetro, Giroscópio, temperatura, umidade, pressão	Sim	Importado
Mediatek Linkit One	MT2502A	Sim	Sim	Não	GSM, GPRS	Sim	Sim	Não	Sim	Importado/ Nacional
Adafruit Flora	ATmega32U4	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
ESP8266 - ESP-12E IoT Board	ESP8266	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado/ Nacional
Seeeduino XIAO	SAMD21G18	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Adafruit Gemma M0	ATSAMD21E18	Não	Não	Não	Não	Não	Não	Não	Sim	Importado
Adafruit Industries ESP8285	ESP8285	Sim	Não	Não	Não	Não	Não	Não	Sim	Importado

Fonte: Própria (2025).

Anexo B

Principais sistemas operacionais de tempo real (RTOS)

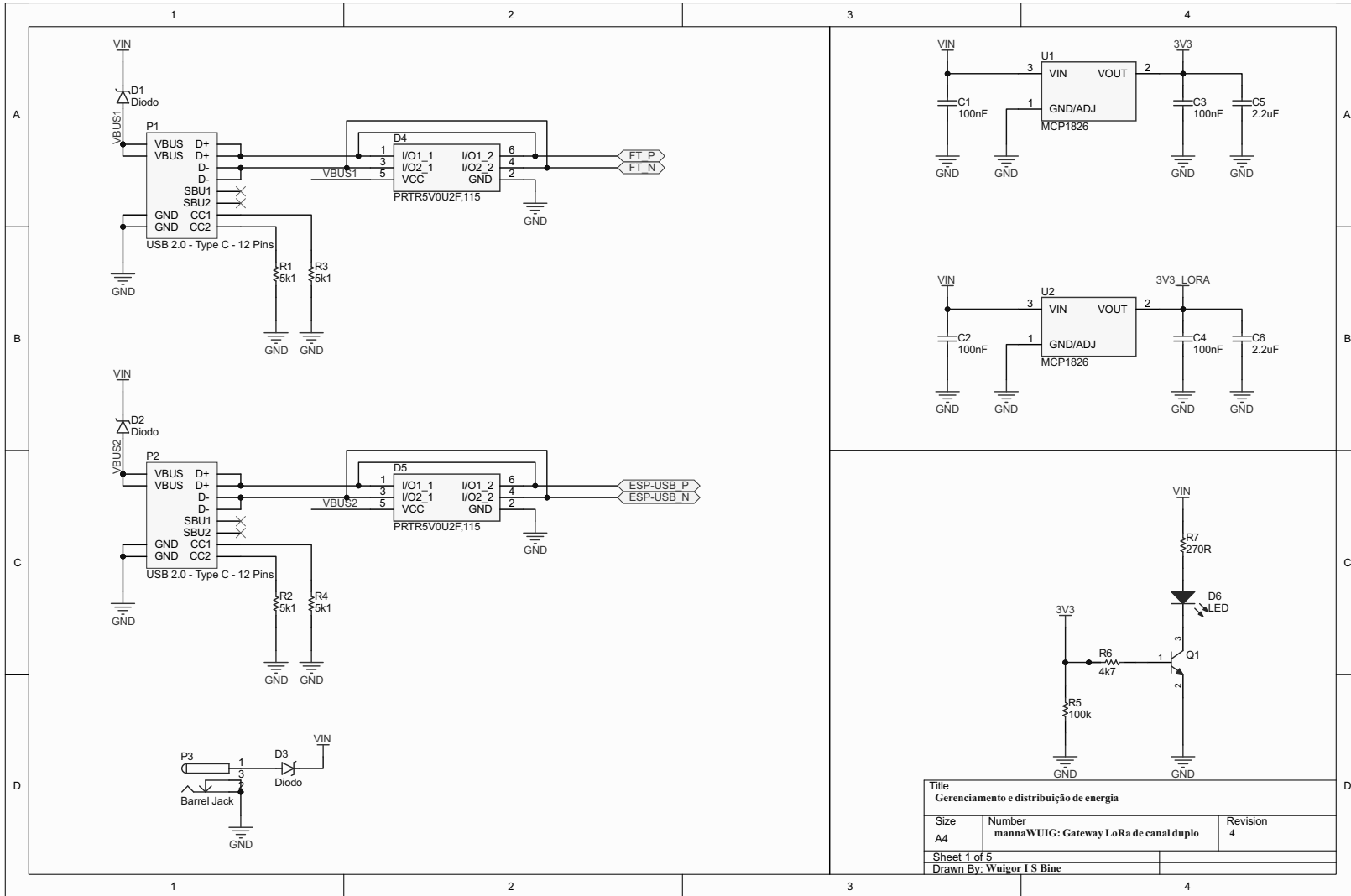
Tabela 15 – Principais sistemas operacionais de tempo real (RTOS)

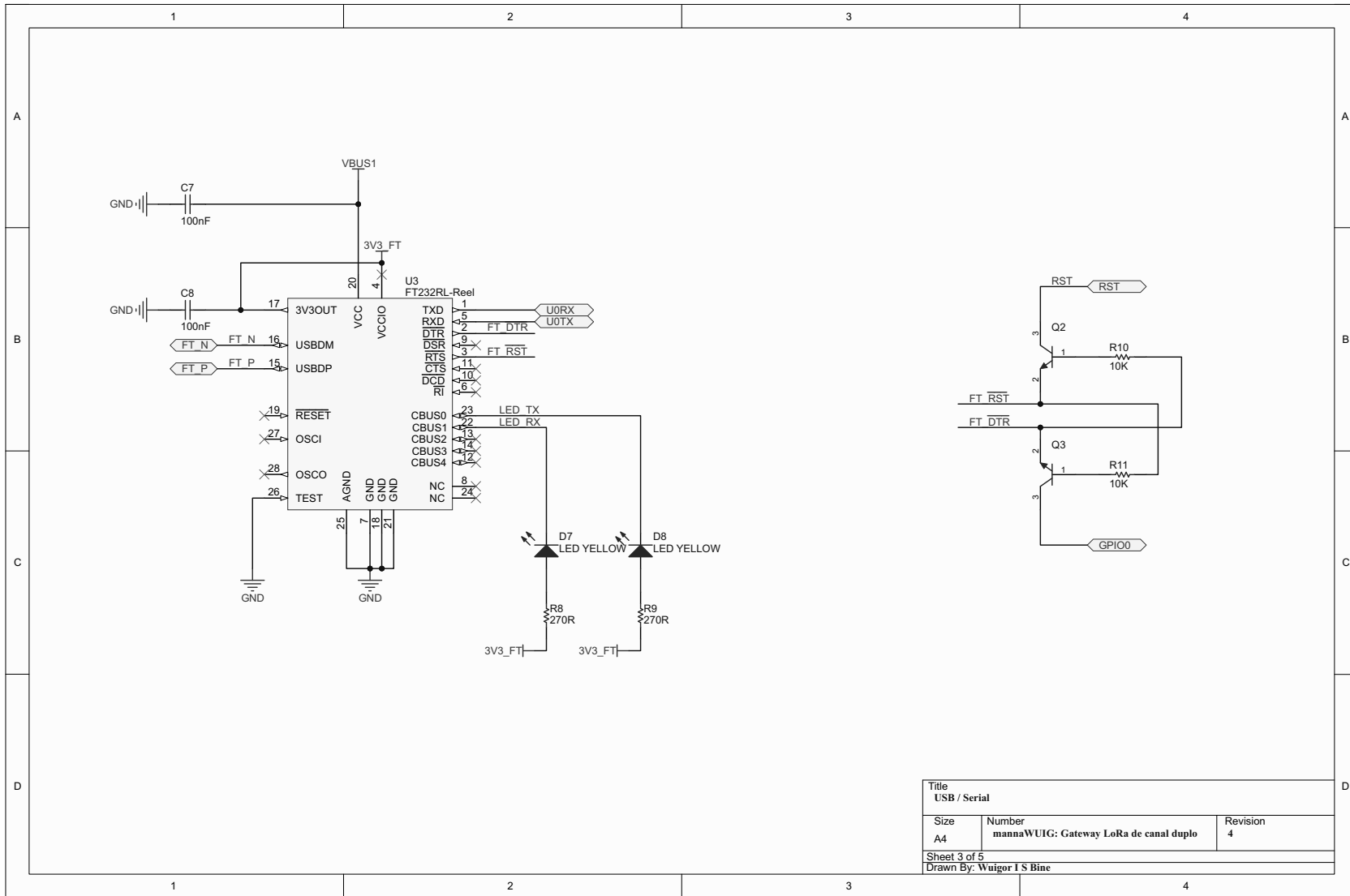
Característica	ContikiOS	FreeRTOS	RIOT	TinyOS	LiteOS	VxWorks	Erika	NutOS
Versão	V3.0, 2015	V10.2.1, 2019	V2.1	V2.1.2,2012	V2.1, 2018	V7.0	V3.0	V5.2.4
Arquitetura	Modular	Microkernel RTOS	Microkernel RTOS	Monolítico	Modular	Modular	Modular	Modular
Hardware Suportado	AVR, TI CC2538, MSP430, MSP430x	ARM Cortex – A5, M3, M4, M7, MSP432, MSP430, Xilinx	AVR, ARM, Cortex-M, X86, MSP430	Atmel AVR, ATmega1281, TI – MSP430, CC2420	Mica Z e IRIS	Intel X86, Fujitsu FR-V, MIPS	Atmel AVR, ARM Cortex M4, MSP 430, Mica Z	8-bit AVR, AVR32, ARM7, ARM9, ARM Cortex M3
Gerenciamento de Memória	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Gerenciamento de Energia	Sim	Sim	Sim	Sim	Sim	-	-	-
Escalonamento	Cooperativo	Preemptivo	Preemptivo	Não preemptivo	Na conclusão da tarefa	Preemptivo	Preemptivo & Não preemptivo	Não Preemptivo
Algoritmos de Escalonamento	Prioridade (Round Robin)	Prioridade (Round Robin)	Prioridade (Round Robin)	FIFO	Prioridade (Round Robin)	Prioridade/ Adaptativo	Dinâmico (Prioridade Fixa, Termina Mais Rápido Primeiro)	-
Modelos de Programação	Multithreading	Multithreading	Multithreading	Dirigido a Eventos	Multithreading	Multiprocessamento	Multithreading	Multithreading
Concorrência	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Requisitos Mínimos de Memória (KiB)	RAM: 2 ROM: 30	RAM: 10 ROM: 12	RAM: 1,5 ROM: 5	RAM: 1 ROM: 4	RAM: 4 ROM: 128	RAM: 2MiB ROM: 512	RAM: 2 ROM: 12	RAM: 4 ROM: 128
Classes de Dispositivos Alvo	C0 e C1	C1 e C2	C1 e C2	C0 e C1	C0 e C1	-	C0 e C1	C0 e C1
Simulador Cooja, Netsim	-	IoT-Lab	TOSSIM, PowerTossim	AVRORA	-	-	-	-
Base de Dados	Coffee, Raima	ITTIA DB, Raima	Raima	TinyDB, Raima	-	Raima	-	-
Pilha de Protocolo IoT	Sim	Sim	Sim	Sim	Sim	Sim	Parcial	Parcial
Segurança	TLS	TLS (WolfSSL)	-	Cipher Block Chaining	-	SSL/TLS, Criptografia, SecureBOOT	-	TLS (WolfSSL)
Pilha de Rede	uIP, RIME	-	OpenWSN	BLIP	-	Sim	-	Nut/Net(TCP/IP)
Aplicações	Casas Inteligentes, Agricultura Inteligente, Cidades Inteligentes, Industria 4.0	Casas Inteligentes, Cidades Inteligentes, Industria 4.0	Wearables, Casas Inteligentes, Agricultura Inteligente, Cidades Inteligentes, Industria 4.0	Casas Inteligentes, Agricultura Inteligente, Cidades Inteligentes, Industria 4.0	Casas Inteligentes, Redes Veiculares, Malha Hídrica Inteligente, Pesca Inteligente	Aeroespacial, Defesa, Industria 4.0, Saúde Inteligente, Eletrônicos de Consumo	Casas Inteligentes, Wearables	-

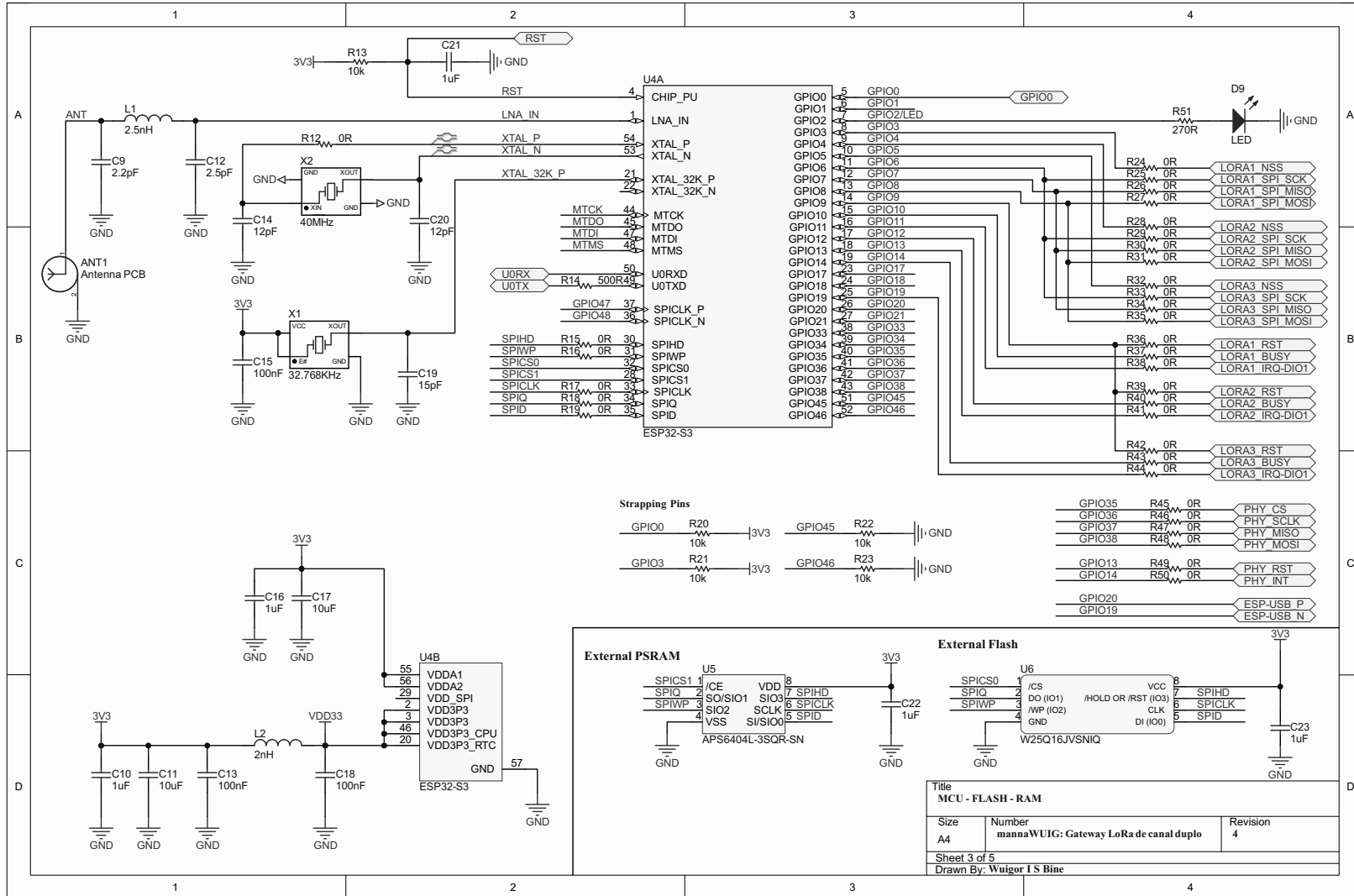
Fonte: Adaptado de Swamy e Kota (2020).

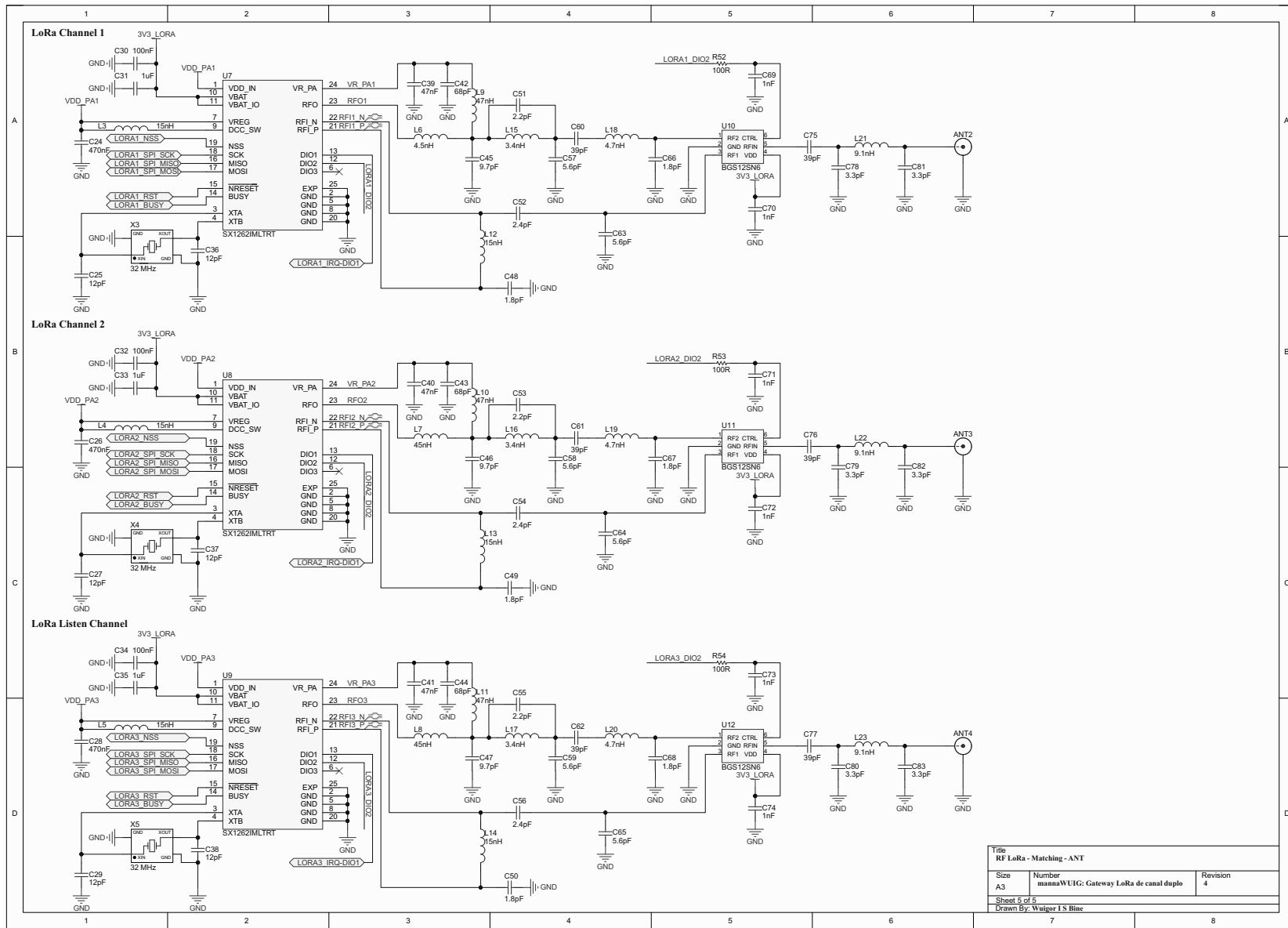
Anexo C

Projeto do *hardware* proposto para o
Gateway LoRa de canal duplo









Title RF LoRa - Matching - ANT		
Size A3	Number mannaWUIG: Gateway LoRa de canal duplo	Revision 4
Sheet 5 of 5		
Drawn By: Wuijor I S Bine		

