PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS

LEONARDO ZANETTI MACHADO

LORA APLICADO AO AGRONEGÓCIO

CAMPINAS

2021

pontifícia universidade católica de campinas

centro de ciências exatas, ambientais e de tecnologias

faculdade de engenharia elétrica

leonardo zanetti machado

lora aplicado ao agronegócio

Trabalho de Conclusão de Curso apresentado como exigência para graduação no curso de engenharia elétrica, do Centro de Ciências Exatas, Ambientais e de Tecnologias, da Pontifícia Universidade Católica de Campinas.

Orientador: Prof. Dr. Frank Herman Behrens

PUC - CAMPINAS

2021

PONTIFícia universidade católica de campinas

CENTRO DE CIÊNCIAS EXATAS, AMBIENTAIS E DE TECNOLOGIAS

FACULDADE DE ENGENHARIA ELÉTRICA

LEONARDO ZANETTI MACHADO

lora aplicado ao agronegócio

Dissertação defendida e aprovada em <colocar data> pela comissão examinadora:

Prof. Dr. Frank Herman Behrens

Orientador e presidente da comissão examinadora.

Pontifícia Universidade Católica de Campinas

<Adicionar Examinador 1 aqui>

<Adicionar Examinador 2 aqui>

Campinas

2021

Dedico este trabalho de conclusão de curso aos meus pais por terem moldado o ser humano que me tornei a exemplo de pessoas tão corretas como eles. Dedico em especial ao meu pai, por ser uma das pessoas mais incríveis que já conheci e por ser a minha grande inspiração.

AGRADECIMENTOS

**A Deus,**

Por ter sempre me guiado ao longo de toda a minha vida, iluminando meus caminhos, me dando força e coragem para seguir em frente mesmo durante os momentos mais difíceis.

**Ao meu pai (*em memória)*,**

Por sempre me inspirar a bondade e a benevolência, a simplicidade e a atenção ao que realmente importa; por incentivar sempre o meu interesse pela ciência e me apoiar em todos os projetos realizados. Por todos os momentos e conversas que tivemos juntos, por me fazer gostar de café, futebol, música boa e carros antigos.

**À minha mãe,**

Por todo o suporte, carinho, afeto e por ter me ensinado a ser uma pessoa forte, batalhadora, corajosa e que sempre enfrenta os desafios, por maiores que sejam, com a cabeça erguida. Por sempre me incentivar a correr atrás dos meus sonhos e me prover os pilares da educação para a minha formação.

**Aos primos, tios, avós e padrinhos,**

Por todo o carinho e suporte, principalmente nos momentos mais difíceis, por todos os ensinamentos que ajudaram a moldar a pessoa que me tornei.

**À minha namorada,**

Por todo o suporte afetivo e incentivo durante a faculdade, por sempre estar comigo e me apoiar nas minhas decisões e por todos os momentos bons compartilhados em todos esses anos juntos.

**Ao Professor Frank Herman Behrens,**

Por todo o conhecimento compartilhado durante os anos da faculdade em suas disciplinas e toda a orientação durante a realização do trabalho, além de todas as ideias agregadas ao projeto.

**Ao Professor Antonio Demanboro,**

Por ter sido meu orientador no projeto de Iniciação Científica, responsável por despertar meu interesse na área de comunicação sem fio e radiofrequência.

**Aos meus melhores amigos Matheus Bueno, Júlia Grannier e Gabriel Rosa,**

Por sempre estarem comigo, por todo o apoio, conversas e momentos juntos.

**Aos meus colegas da faculdade,**

Por todos os momentos e conhecimentos compartilhados.

**Ao Centro de Pesquisa e Desenvolvimento (CPQD),**

Pela oportunidade de estágio, proporcionando o maior desenvolvimento técnico da minha carreira e por todas as pessoas que foram colocadas em meu caminho.

“A vida é aquilo que acontece enquanto você está ocupado fazendo outros planos.”

John Lennon (1940 – 1980)

resumo

O trabalho realizado tem como finalidade a elaboração de um projeto para a conclusão do curso de bacharelado em Engenharia Elétrica, baseado nas tecnologias de sensores sem fio acopladas ao âmbito do agronegócio. A partir do uso de comunicação via radiofrequência, a solução proposta tem como objetivo promover um sistema de monitoramento para pequenos e médios produtores, proprietários de granjas e armentos, através de uma plataforma de gerenciamento de dados, exibindo ao usuário as diversas variáveis críticas que precisam ser analisadas para uma maior eficiência no cultivo dos produtos a partir da união entre agricultura de precisão, sistemas embarcados e a ferramenta Django.

Palavras-chave: LoRa, LPWAN, Agronegócio, IoT, Django.

abstract

The work carried out has as purpose the elaboration of a project for the conclusion of the bachelor's degree in Electrical Engineering, based on wireless sensor technologies coupled to the agribusiness scope. From the use of radio frequency communication, the proposed solution aims to promote a monitoring system for small and medium producers, owners of farms through a data management platform, showing to the user the various critical variables that need to be analysed for greater efficiency in the cultivation of products through de combination of precision agriculture, embed systems and the Django framework.

Keywords: LoRa, LPWAN, Agrobusiness, IoT, Django.

lista de figuras

Figura 1 - Valor bruto da Produção Agropecuária. 16

Figura 2 - Imagem do “DSKY”. 18

Figura 3 - LGM-30 Minuteman. 19

Figura 4 - Arduino. 21

Figura 5 - ESP-32. 21

Figura 6 - ESP-32 com conectividade de rádio. 22

Figura 7 – Data-bus de 8 bits. 24

Figura 8 – Interface serial. 25

Figura 9 – Processo de modulação em AM. 26

Figura 10 – Processo de modulação em FM. 27

Figura 11 – Processo de modulação em ASK. 28

Figura 12 – Processo de modulação em FSK. 29

Figura 13 – Modulação a partir da CSS. 30

Figura 14 – Gráfico temporal de fator de espalhamento. 30

Figura 15 – Ilustração da técnica de Polling. 32

Figura 16 – Ilustração da técnica de Round Robin. 33

Figura 17 – Topologia broadcast. 34

Figura 18 – Topologia multicast. 35

Figura 19 – Topologia unicast. 35

Figura 20 – Arquitetura do projeto. 37

Figura 21 – Sensor DHT22. 39

Figura 22 – Sensor de umidade do solo. 39

Figura 23 – Arduino IDE. 40

Figura 24 – ESP-32 LoRa e DHT22 em operação. 41

Figura 25 – Código para leitura dos valores. 41

Figura 26 – Empacotamento dos valores de temperatura e umidade. 43

Figura 27 – Parte do código executada em looping. 44

Figura 28 – Módulo de decodificação dos dados. 44

Figura 29 – Leitura da entrada de valores com Python. 45

Figura 30 – Representação do código de uma página web. 47

Figura 31 – Função de inserção dos primeiros dados. 48

Figura 32 – Função de atualização incremental dos dados. 48

Figura 33 – Gráfico para valores de temperatura. 49

Figura 34 – Gráfico para valores de umidade. 49

lista de gráficos

lista de tabelas

Tabela 1 – Faixa de frequências LoRa de acordo com as principais regiões. 31

Tabela 2 – Especificações do da placa ESP-32 LoRa. 38

Tabela 3 – Especificações do transceptor LoRa. 38

Tabela 4 – Especificações do sensor DHT22. 39

Tabela 5 – Especificações do sensor de umidade do solo. 39

Tabela 6 – Número de componentes utilizados. 40

lista de abreviações

|  |  |
| --- | --- |
| IoT | *Internet of Things* |
| WAN | *Wide Area Network* |
| CSS | *Chirp Spread Spectrum* |
| SF | *Spreading Factor* |
| UR | *Umidade Relativa* |
| IDE | *Integrated Development Envionment* |
| COM | *Component Object Model* |
| UTF | *Unicode Transformation Format* |
| HTML | *HyperText Markup Language* |
| JSON | *JavaScript Object Notation* |

sumário

# INTRODUÇÃO

* 1. **Agricultura no Brasil**

A agricultura é definida pelo conjunto de técnicas de cultivo da terra para a obtenção de produtos e garantia da subsistência alimentar do ser humano e, também, a exploração de matérias-primas para a produção de combustíveis, medicamentos, ferramentas e roupas. Historicamente, a atividade agrícola teve início há mais de 12 mil anos, durante o período neolítico com a formação das primeiras civilizações. Além do plantio de sementes para o consumo, o homem passa a explorar a domesticação e criação de animais, surgindo então, a pecuária. Com o desenvolvimento das práticas e costumes, surge, também, a consolidação do escambo envolvendo produtos cultivados e até mesmo a troca de animais por outras mercadorias (ainda sem o uso da moeda), o que indica o começo de um valor econômico relacionado às atividades rurais.

No Brasil, as principais atividades agrárias se iniciam no século XVI, com as Capitanias Hereditárias, que foram um sistema administrativo implementado pela Coroa portuguesa no Brasil nos anos 1500. Esse sistema foi caracterizado pela divisão do território brasileiro pertencente a Portugal, definido no Tratado de Tordesilhas, que tinha como finalidade a exploração territorial e econômica de cada região, utilizando, principalmente, práticas agrícolas de monocultura com mão-de-obra escrava e grandes latifúndios.

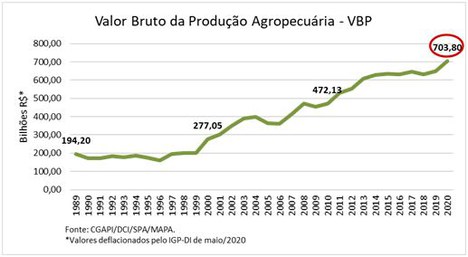
A partir do desdobramento do sistema implementado, a produção passa a ter grandes avanços e novas opções. Juntamente com a exploração de minérios, durante o século XVIII, novos tipos de sementes e outros tipos de vegetais são adicionados no leque do cultivo. Nesse momento, já existe um importante interesse comercial relacionado às atividades agrícolas, indicando sua potencialidade econômica para o país que, durante o século XIX, tem a cafeicultura como principal fonte de insumos para exportação.

Essa linha do tempo conclui-se atualmente mostrando o grande e importante papel da agricultura na economia do Brasil, que possui uma vasta extensão territorial com solos muito férteis e que possibilitam diversos tipos de cultivo, aumentando ainda mais as oportunidades econômicas.

Nos dias atuais, o Brasil é caracterizado por uma grande diversidade de produtos cultivados nas diferentes regiões. Por exemplo, na região Sul, que possui descendência de imigrantes europeus, cultiva-se principalmente sementes como soja e milho assim como cana-de-açúcar e algodão. O setor pecuário, por sua vez, é predominantemente preenchido pela cultura de aves e suínos.

Segundo a Embrapa, até o mês de setembro do ano de 2020, o Brasil produziu 124,845 milhões de toneladas de soja, em uma área plantada de 36,950 milhões de hectares. Assim como indicado pelos números expressivos, o seguinte gráfico ilustra o processo de crescimento no valor de produção agropecuária no Brasil desde 1989 até o ano de 2020.

Figura 1 - Valor bruto da Produção Agropecuária.



Fonte: EMBRAPA, 2020.

A partir da análise do gráfico presente na imagem, pode-se observar que é notável a participação do setor agrônomo como grande fonte de renda para o país e também o coloca entre as principais nações exportadoras de produtos agrícolas.

Devido à grande importância do setor agropecuário no Brasil, juntamente com a transformação das técnicas e modernização das atividades, o campo também sofreu mudanças positivas em relação aos métodos de cultivo e às ferramentas utilizadas são cada vez mais tecnológicas para que seja atingida maior eficiência na produção.

* 1. **Revoluções tecnológicas**

A história é marcada por diversas revoluções e, entre as mais importantes, estão as Revolução Industriais. A primeira, iniciada entre os séculos XVIII e XIX, foi marcada pela invenção das máquinas a vapor e, com isso, a utilização das vias férreas cujas máquinas eram movidas a esse tipo de energia recém-descoberto. Em seguida, já no século XX, tem-se a segunda revolução industrial, caracterizada pela implementação do modelo de linhas de produção em massa desenvolvido por Henry Ford.

Na segunda metade do século XIX, é iniciada a terceira revolução industrial, representada pelo surgimento da internet, possibilitando a conexão entre dispositivos, através do tráfego de dados. Em seguida, é estabelecida a 4ª e atual revolução industrial, que tem a premissa de representar uma nova tendência no cenário industrial com a introdução de dispositivos mais inteligentes e que podem ser interconectados, incluindo aparelhos que, essencialmente, não foram produzidos para isso. A esse cenário, dá-se o nome de “Internet das Coisas” (*Internet of Things*).

* 1. **Sistemas Embarcados**

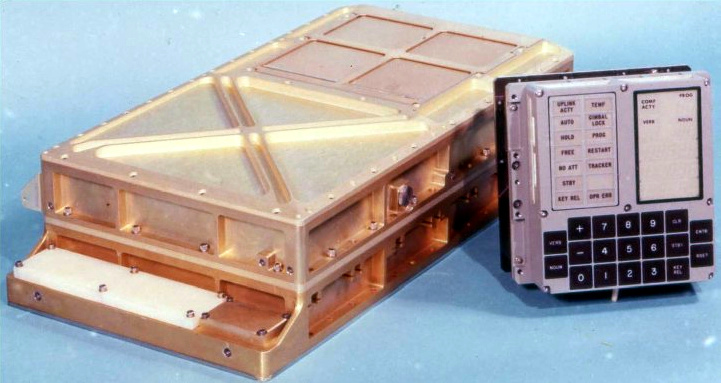
Paralelamente às revoluções industriais presenciadas pela humanidade, desde o surgimento dos primeiros computadores, acompanha-se o desdobramento de uma evolução tecnológica extremamente veloz, contando com dispositivos cada vez mais potentes e versáteis. Entretanto, no início da descoberta desses instrumentos, em 1940, os computadores digitais ainda tinham um grande tamanho, portanto não poderiam ser transportados com facilidade e nem utilizados de maneira remota. Nesse contexto, destaca-se a importância dos sistemas de processamento encapsulado, que promovem mobilidade e praticidade aos processos computacionais.

* + 1. ***Apollo Guidance Computer***

É reconhecido que, em 1965, ao desenvolver um sistema de guia para o projeto Apollo, Charles Stark Draper, no MIT, acaba criando o primeiro sistema embarcado, que se tratava de um guia de movimentação, orientação e controle (computador de bordo), operando em tempo real, o qual recebeu o nome de *Apollo Guidance Computer*. Para que tivesse seu tamanho e peso reduzidos, além de ser móvel, o dispositivo contava com o uso de circuitos integrados, que são circuitos eletrônicos de tamanho reduzido, compostos por materiais elétricos (principalmente transistores) encapsulados, sendo um dos primeiros a fazer uso de tal tecnologia ainda recente na época.

O sistema contava com portas lógicas e suas combinações, como os *flip-flops*, que eram responsáveis por realizar as operações lógicas e, assim, possibilitar o controle de partes do módulo lunar via interface de usuário, a qual acredita-se que seja uma das primeiras da história, a partir de um teclado e um *display*, denominado *“DSKY”* (Fig. 2).

Figura 2 - Imagem do “DSKY”.



Fonte: Wikipedia, 2018.

* + 1. ***LGM-30 Minuteman***

O míssil *LGM-30 Minuteman* (Fig. 3) é um projeto balístico nuclear norte-americano e foi lançado em 1961, marcando a data em que os sistemas embarcados começam a ser maciçamente produzidos. Além do processamento modular, o sistema contava com um disco rígido para o armazenamento das informações. Com a evolução e descobertas acerca dos operadores lógicos, também foram implementadas novas alternativas de portas lógicas como o NAND, o que corroborou para o seu barateamento, diminuindo os custos de produção dos sistemas embarcados até chegar no estado em que estão atualmente.

Figura 3 - LGM-30 Minuteman.



Fonte: MISSILETHREAT, 2018.

Com o aumento da produção e utilização dos sistemas embarcados, seu preço vem diminuindo desde a década de 60 e seu poder de processamento aumentando expressivamente. Para regulamentar o processo de produção e uso dos dispositivos, em 1978, é lançada pela associação americana *National Electrical Manufacturers Association* (criada em 1926) uma norma referente aos controladores programáveis, que trata de uma padronização para o uso de produtos elétricos e eletrônicos.

Com a difusão dos sistemas embarcados, os componentes eletrônicos externos, antes utilizados de forma modular e separada, foram incorporados pelos processadores dos sistemas, trazendo assim, maior redução em seu tamanho e uma maior capacidade de processamento para a realização de um número maior de tarefas simultaneamente.

Além do tamanho reduzido e grande poder, com a evolução da eletrônica e o surgimento dos sistemas digitais, alguns componentes mais caros puderam ser substituídos por circuitos integrados que têm a capacidade de desempenhar a mesma função, o que contribuiu para uma redução ainda maior em seu preço, tornando-os cada vez mais acessíveis.

Atualmente, os sistemas embarcados são parte fundamental do panorama tecnológico, que contribuem para a dição do rumo em que os avanços tecnológicos caminham. Através deles, é possível entender como acontece a 4ª Revolução Industrial, também intitulada de surgimento da Indústria 4.0.

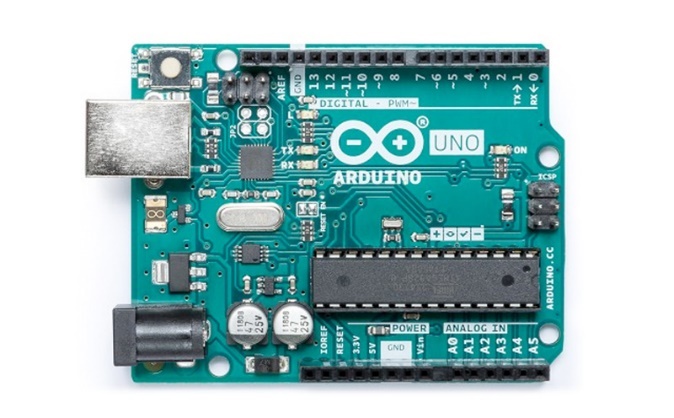
* 1. **Internet das Coisas**

A partir do avanço da microeletrônica, do surgimento dos diversos tipos de sensores, do desenvolvimento de protocolos de comunicação de dados *online* e *offline*, bem como da evolução dos sistemas, é possível entender o significado do termo “Internet das Coisas” (*Internet of Things*), utilizado para caracterizar o atual momento tecnológico.

“É a rede de objetos físicos, sistemas, plataformas e aplicativos com tecnologia embarcada para comunicar, sentir ou interagir com ambientes internos e externos. Permite que as “coisas” interajam umas com outras e que tomada de decisões sejam feitas. A internet das coisas é a base da Indústria 4.0 (HERMANN, PENTEK, OTTO, 2015).”

Os sistemas embarcados têm protagonismo nessa área, porque possibilitam, a partir de suas interfaces, conectar dispositivos como sensores e atuadores a objetos e aparelhos que, inicialmente, não possuem conexão, para inúmeras finalidades, como o monitoramento de temperatura, pressão, umidade do ar e até mesmo o controle de motores através da internet. Além da versatilidade, os dispositivos são caracterizados pelo tamanho reduzido e por seu alto poder de processamento de dados. Um exemplo comum e bastante popular é o Arduino, uma placa de desenvolvimento para aplicações em IoT.

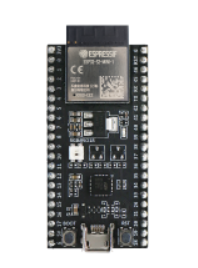
Figura 4 - Arduino.



Fonte: GOOGLE, 2021.

Além dos mais simples, existem aparelhos voltados à Internet das Coisas que possuem especificações ainda mais robustas, como é o caso dos modelos produzidos pela empresa *Espressif Systems*, que não só conta com o processamento, mas também a possibilidade de tráfego de dados via internet com o modelo ESP-32 (Fig. 5) e, até mesmo, radiofrequência, com o modelo ESP-32 LoRa (Fig. 6), desenvolvido por empresas como a *Heltec* a partir do modelo inicial da *Espressif*.

Figura 5 - ESP-32.



Fonte: ESPRESSIF, 2021.

Figura 6 - ESP-32 com conectividade de rádio.



Fonte: MERCADO LIVRE, 2021.

A partir da utilização de dispositivos como esses, é possível a criação de redes de comunicação entre unidades de processamento que monitoram diversos tipos de ambientes.

# FUNDAMENTAÇÃO TEÓRICA

Através da introdução, pode-se entender o escopo geral do trabalho, apresentando os principais pontos de forma abrangente. Assim, para que o objetivo do projeto possa ser compreendido, faz-se necessário o entendimento específico de cada uma de suas partes.

* 1. **O que é a agricultura de precisão**

Por definição, o termo “Agricultura de precisão” remete à “utilização de aparelhagem tecnológica avançada para avaliar e acompanhar de maneira mais precisa as condições das áreas de atividades agronômicas baseada no princípio da variabilidade do solo e clima” (WIKIPEDIA, 2020). Entre outras palavras, está ligada ao monitoramento refinado das diversas variáveis críticas e exploração de dados relacionadas ao cultivo dos produtos rurais e, para isso, utiliza-se de diversas opções presentes no mercado.

De acordo com o “Manual de Segurança e Qualidade para a Cultura da Soja”, disponibilizado pela Embrapa, a soja, principal semente cultivada no Brasil, requer valores específicos relacionados ao clima para que se tenha o rendimento esperado em sua colheita. Por exemplo, “a faixa de temperatura ideal para o desenvolvimento da soja está entre 20ºC e 30ºC. Sob temperaturas menores ou iguais a 10ºC ou temperaturas acima de 40ºC, a soja sofre redução de crescimento ou distúrbios na floração e diminuição na capacidade de retenção de vagens” (EMBRAPA, 2005).

Outro grande integrante do grupo de maiores produtos presentes na economia brasileira é o café que, por exemplo, passa por um processo de secagem antes de ser torrado e comercializado. Nesse processo, a umidade do café deve atingir valores entre 11% e 12% de acordo com a região em que é cultivado, a fim de que esteja preparado para a torra e, posteriormente, para sua comercialização. Com isso, entende-se que é necessário um controle preciso e refinado sobre a umidade do local em que o café passa por esse processo, para que a qualidade seja atingida de acordo com o esperado pelo produtor.

Além das condições climáticas, há a preocupação em relação ao tipo de solo, sua fertilidade e, também, o manejo correto do cultivo, juntamente com outros fatores de extrema importância, os quais necessitam ser avaliados pelo produtor para que, por sua vez, tome as decisões adequadas a fim de que obtenha o maior rendimento possível.

Para isso, são utilizadas tecnologias recentes, contando com sensores analógicos e digitais, microcontroladores capazes de processar os dados de monitoramento e, até mesmo, transceptores que são responsáveis pelo envio dos dados processados de maneira sem fio, gerando uma versatilidade muito grande para a aplicação.

* 1. **Comunicação de dados**

A comunicação de dados representa o transporte de valores entre dispositivos eletrônicos, permitindo que a informação seja veiculada de um ponto de origem a um ponto de destino. Esse processo pode acontecer tanto em meios confinados, como ambientes cabeados (computadores, por exemplo) nos quais os dados são trafegados a partir da corrente elétrica, quanto em meios não-confinados, como o ar, no qual os dados são trafegados através de ondas eletromagnéticas, portanto sem a necessidade de conexão física (cabeamento).

* + 1. **Comunicação em meios confinados**

A tecnologia dos sistemas embarcados demanda que os circuitos interligados possam trocar informações entre si através de canais confinados ou barramentos e para que isso seja alcançado de forma íntegra e eficiente, é necessária a definição de um conjunto de regras e procedimentos que controlem todo esse processo, que são os chamados protocolos de comunicação.

Em meios confinados, os protocolos de comunicação podem ser organizados em 2 categorias: comunicação paralela e comunicação serial.

* + - 1. **Comunicação Paralela**

Esse tipo de comunicação é realizado através da transmissão simultânea de maneira rápida, a partir de “ondas” de *bits*. Entretanto, são necessárias múltiplas conexões no barramento com 8, 16, ou mais fios conectados, formando os chamados *“data-buses*”, que carregam todo o conteúdo da informação de uma vez.

Figura 7 – Data-bus de 8 bits.

Uma imagem contendo Gráfico

Descrição gerada automaticamente

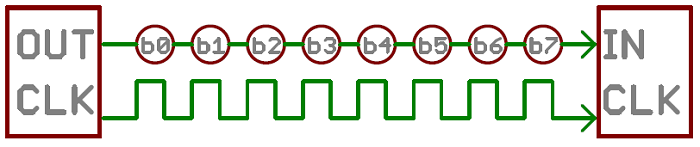
Fonte: SPARKFUN.

A Figura 13 mostra um *data-bus* de 8 *bits*, que envia 8 informações a cada pulso de *clock*, utilizando 8 fios para a transmissão dos dados e 1 fio para a contagem de tempo.

* + - 1. **Comunicação Serial**

Esse tipo de comunicação, diferentemente da anterior, utiliza a transmissão de informações através do envio sequencial de dados, um por vez. Ao mesmo tempo em que o resultado obtido é uma comunicação mais lenta em relação à paralela, usa-se muito menos fios nas conexões. A Figura 14 mostra a interface de uma comunicação serial, transmitindo um *bit* a cada pulso de *clock*, utilizando apenas 2 fios.

Figura 8 – Interface serial.



Fonte: SPARKFUN.

* + 1. **Comunicação em meios não confinados**

Para que os dados possam ser transportados de maneira remota, sem fio, é necessário que eles passem por um processo chamado de “modulação”, que codifica os valores binários para a transmissão de forma que possam ser decodificados em sua recepção.

A modulação ocorre na modificação de parâmetros de uma forma de onda eletromagnética e pode ser feita de forma analógica, para a transmissão de sinais analógicos, como a voz (exemplo: telefone antigo) e pode ser feita de forma digital, para a transmissão de sinais digitais, os *bits* (exemplo: celulares). Assim que o espectro de dados é definido, utiliza-se uma onda chamada de Portadora que tem a função de adequar o sinal ao espectro de frequência na qual será transmitido.

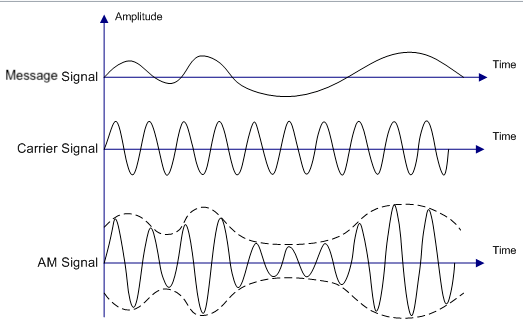
* + - 1. **Modulações Analógicas**

As modulações analógicas foram as primeiras a ser utilizadas na transmissão de dados, de acordo com os recursos que eram disponíveis. Dois exemplos muito utilizados são a modulação AM e FM.

* + - * 1. **Modulação AM (*Amplitude Modulation*)**

A modulação AM é feita a partir da alteração da amplitude de um sinal eletromagnético de acordo com o sinal que carrega os valores que serão transmitidos (*Message Signal*), gerando padrões que podem ser identificados na recepção e “demodulados”. Então, o sinal da onda portadora (*Carrier Signal*) é encapsulado a partir do sinal da mensagem e transmitida. A Figura 7 ilustra o processo de modulação.

Figura 9 – Processo de modulação em AM.



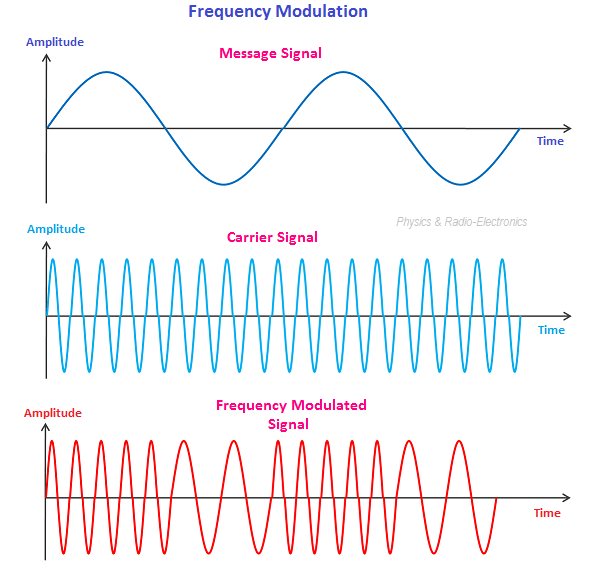
Fonte: WIKIPÉDIA, 2021 (adaptado).

Por se tratar de uma modificação na amplitude do sinal, a modulação em AM tem uma limitação muito grande, pois é bastante suscetível a interferências (ruídos).

* + - * 1. **Modulação FM (*Frequency Modulation*)**

A modulação FM, por sua vez, é realizada na modificação da frequência do sinal de acordo com a mensagem a ser transmitida. Com a variação dos valores de amplitude do sinal, a frequência é aumentada ou diminuída, como mostra a Figura 8.

Figura 10 – Processo de modulação em FM.



Fonte: PHYSICS AND RADIO ELECTRONICS, 2021.

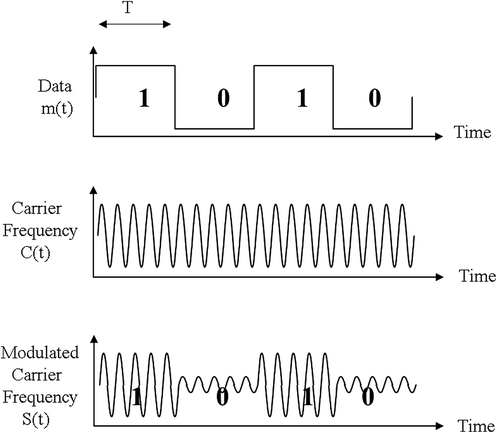
* + - 1. **Modulações Digitais**

Com o desenvolvimento tecnológico, novas possibilidades surgem e passam a ser utilizadas as modulações digitais que tratam *bits*, no lugar de valores analógicos, também modificando parâmetros de ondas eletromagnéticas. Dois exemplos são ASK e FSK, porém existem outros métodos e, até mesmo, combinações de métodos, como o QAM, que utilizam a união entre modulações em amplitude e fase.

* + - * 1. **Modulação ASK (*Amplitude Shift-Keying*)**

A modulação digital de amplitude trata de um chaveamento (por isso a origem do nome “*shift*”) entre valores binários (0 e 1), variando a amplitude do sinal da portadora (*Carrier*) de acordo com os valores binários presentes no sinal da mensagem (*Data*) (Fig. 9).

Figura 11 – Processo de modulação em ASK.

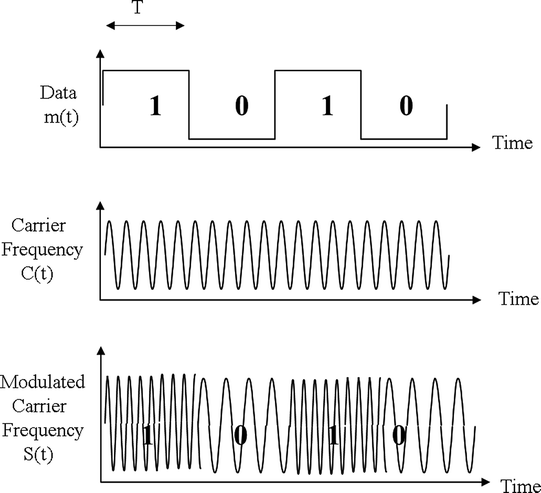


Fonte: SPRINGERLINK, 2021.

* + - * 1. **Modulação FSK (*Frequency Shift-Keying*)**

No caso da modulação em frequência, em vez do chaveamento de amplitude, ocorre o chaveamento de frequência de acordo com os *bits* presentes no sinal da mensagem que é transmitida (Fig. 10).

Figura 12 – Processo de modulação em FSK.



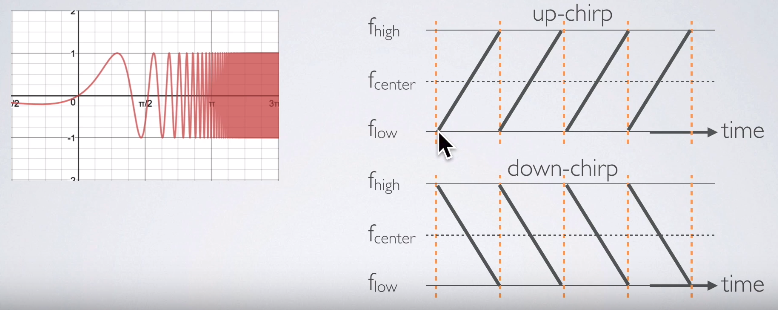
Fonte: SPRINGERLINK, 2021.

* + 1. ***Chirp Spread Spectrum* e LoRa**

A partir das técnicas de modulação digital, surgem métodos similares como o CSS (*Chirp Spread Spectrum*), que é uma forma semelhante ao chaveamento de frequências, porém a partir de saltos de frequência.

Tecnicamente, o CSS funciona a partir de saltos (“*chirp”*, em inglês) de frequência (Fig. 11) entre uma faixa de valores, podendo ser *up-chirp*, caso o salto seja para uma frequência superior ou *down-chirp*, caso o salto seja para uma frequência inferior. Cada um dos saltos representa um símbolo que será decodificado e cada símbolo, por sua vez, pode representar um ou mais *bits* de dados.

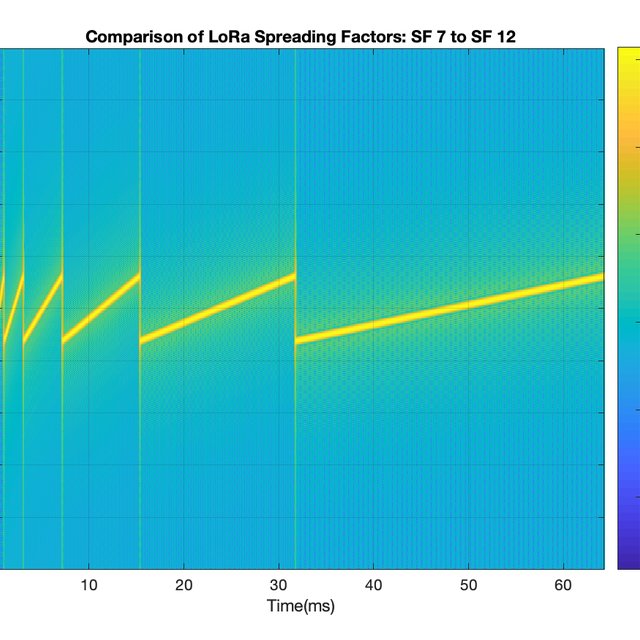
Figura 13 – Modulação a partir da CSS.



Fonte: GTA, UFRJ.

O número de símbolos gerados pode variar de acordo com o fator de espalhamento “SF”, que representa o número de *bits* os quais serão utilizados na modulação. Esse está diretamente ligado a distância que um dispositivo pode se comunicar com outro, visto que, quanto maior o fator de espalhamento, maior a distância capaz de percorrer. Entretanto, quanto maior o fator de espalhamento, maior o tempo de transmissão, diminuindo assim, sua taxa de dados. A figura 12 ilustra o comportamento dos diferentes valores de fator de espalhamentos em um gráfico temporal.

Figura 14 – Gráfico temporal de fator de espalhamento.



Fonte: RESEARCHGATE, 2021.

Percebe-se que a imagem mostra que o menor valor possível de fator de espalhamento (7) gera uma comunicação bem mais rápida, enquanto o maior valor representa, também, uma maior latência de transmissão.

Uma das ferramentas derivadas da tecnologia CSS e que vem se tornando muito popular no cenário da IoT é a recém-apresentada LoRa (*Long Range*, “grande distância”, em inglês), que trata de uma técnica de modulação para redes de áreas extensas (WAN) utilizando frequências de espectro não-licenciados, variando de acordo com o continente em que é aplicada. A tabela 1 mostra a faixa de frequência utilizada pela tecnologia LoRa nas principais regiões em que é utilizada.

Tabela 1 – Faixa de frequências LoRa de acordo com as principais regiões.

|  |  |
| --- | --- |
| Região | Frequência (MHz) |
| Europa | 433 ou 863-870 |
| Estados Unidos | 902-928 |
| China | 470-510 ou 779-787 |
| Austrália | 915-928 |
| Índia | 865-867 |
| Ásia | 433 |
| América do Norte | 915 |

Fonte: MOKOSMART.

N Brasil, a faixa de frequência mais comum para a utilização do LoRa é entre 868 MHz e 915 MHz.

A partir da utilização da tecnologia LoRa, é possível conectar dispositivos que estão localizados em pontos distantes, o que traz proximidade entre a tecnologia e nichos específicos, como a agricultura de precisão.

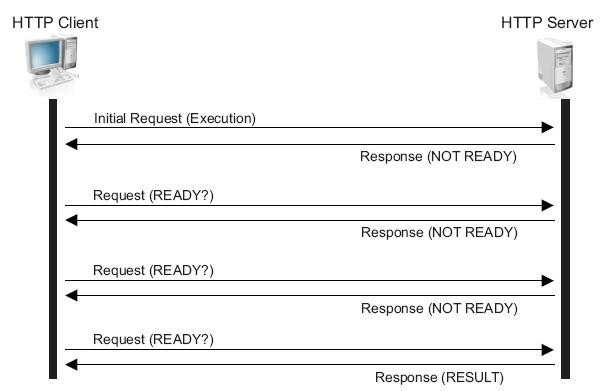
* + 1. **Tecnologia *Master/Slave***

Essa tecnologia é definida por um modelo de comunicação com uma hierarquia básica na qual um dispositivo (*Master*) tem o controle unidirecional sobre todos os outros (*Slaves*), que são os subordinados. Portanto, pode ser aplicada em sistemas nos quais é necessária a requisição de pelo dispositivo controlador aos dispositivos subordinados através, por exemplo, de iterações ou escalonamento.

* + - 1. **Técnica de *Polling***

Essa técnica é baseada em uma estrutura na qual um registrador é dedicado a armazenar o status de um processo e o sistema verifica periodicamente qual é esse status. Para a hierarquia de *Master/Slave*, seria como se o dispositivo controlador enviasse requisições e checasse de tempos em tempos se a resposta dos dispositivos subordinados foi recebida, assim como ilustrado pela figura 13.

Figura 15 – Ilustração da técnica de Polling.



Fonte: RESEARCHGATE.

Em um sistema com mais de um dispositivo subordinado, isso causaria um atraso muito grande na execução de processos, visto que o *Master* só enviaria a requisição ao próximo *Slave*, caso verificasse que a resposta do *Slave* anterior foi recebida.

* + - 1. **Técnica *Round Robin***

A técnica de *Round Robin* é desenvolvida a partir de um algoritmo escalonador de processos, os quais são organizados em filas e tem um tempo definido para serem executados, chamado de *“Time-Slice”*. Assim que cada fatia de tempo é encerrada, o próximo processo da fila é iniciado e assim sucessivamente. A figura a seguir mostra um diagrama representativo dessa técnica.

Figura 16 – Ilustração da técnica de Round Robin.



Fonte: EMBARCADOS.

Em uma topologia de rede na qual é empregado esquema de *Master/Slave*, o *Round Robin* pode ser aplicado em uma situação em que o dispositivo controlador envia requisições de forma escalonada entre os subordinados a cada fatia de tempo e durante o *Time-Slice*, o *Master* fica disponível para receber a resposta A partir do momento que ela se encerra, mesmo caso não receba resposta alguma, a requisição é enviada ao próximo subordinado. Assim que todos são acionados, o processo retorna ao primeiro *Slave*.

Embora a ideia por trás da técnica seja majoritariamente aplicada para a comunicação entre dispositivos eletrônicos em meio confinado, como circuitos interligados, ela pode ser transposta a outros modelos e arquiteturas, como a interação entre rádios em uma rede, de acordo com a sua topologia. Dessa forma, um dos dispositivos atua como *Master* enviando requisições e monitorando respostas dos rádios que fazem o papel de *Slaves*.

* + 1. **Topologias de Rede**

No âmbito das telecomunicações, a forma com a qual os dispositivos são interligados em uma rede e como trocam dados e informações entre si recebe o nome de topologia. Existem inúmeros tipos de topologia de rede, mas levando em consideração o foco do projeto, as mais importantes são *Broadcast*¸*Multicast* e *Unicast*.

* + - 1. **Topologia *Broadcast***

A topologia *broadcast* é dada pela existência de um dispositivo central que consegue enviar a todos os outros receptores informações simultaneamente através de rádio difusores, por exemplo. Sinais midiáticos como televisão, rádio etc. são enviados utilizando essa técnica, pois a mesma informação precisa ser enviada da emissora a todos os receptores simultaneamente. A figura a seguir ilustra a topologia de *broadcast*, na qual o dispositivo central (em vermelho) envia o mesmo sinal aos receptores (em verde) ao mesmo tempo.

Figura 17 – Topologia broadcast.

Uma imagem contendo relógio

Descrição gerada automaticamente

Fonte: WIKIPEDIA.

* + - 1. **Topologia *Multicast***

No caso da topologia *multicast*, o funcionamento do tráfego de informações é muito semelhante ao *Broadcast*, simultaneamente, com a exceção de que os dados são enviados apenas a um grupo seletivo de receptores, ponto a ponto. Isso pode ser observado, por exemplo, em transmissões de dados com base na rede em que os dispositivos estão conectados, portanto, só receberão as informações, aqueles que estiverem na mesma rede. A figura 16 ilustra o funcionamento da topologia *multicast*.

Figura 18 – Topologia multicast.

Padrão do plano de fundo

Descrição gerada automaticamente

Fonte: WIKIPEDIA.

* + - 1. **Topologia *Unicast***

A última topologia de rede citada se refere à transmissão de informações endereçadas um único destino, também de forma ponto a ponto. A figura a seguir ilustra a comunicação *unicast*, em que o ponto em vermelho envia os dados que são endereçados somente ao ponto em verde.

Figura 19 – Topologia unicast.

Desenho de um cachorro

Descrição gerada automaticamente com confiança baixa

Fonte: WIKIPEDIA.

# PROPOSTA

Atualmente, ainda que a evolução tecnológica tenha passado por diversos desdobramentos e tenham surgido novas propostas de comunicação de dados, a Internet das Coisas no ambiente rural ainda se encontra dando seus primeiros passos.

Visando incorporar conectividade à agricultura de precisão, o presente projeto tem como proposta apresentar uma aplicação, que promove ao pequeno e médio produtor agropecuário, o protótipo (utilizando um leque de sensores que pode ser expandido) de um sistema responsivo de monitoramento e gerenciamento de dados importantes, com a finalidade de aumentar a eficiência do cultivo de seus produtos.

Para isso, a arquitetura proposta consiste em rádios interligados em topologia *unicast* com escalonamento de requisições enviadas pelo *master* aos *slaves* conectados a ele, utilizando a técnica de *Round Robin*.

A figura a seguir ilustra o fluxograma de proposta do projeto.

Figura 20 – Arquitetura do projeto.

Uma imagem contendo Diagrama

Descrição gerada automaticamente

Fonte: AUTORIA PRÓPRIA.

# RECURSOS

* 1. **Recursos de *Hardware***
     1. **ESP-32 LoRa**

Para o processamento remoto dos dados obtidos pelos sensores e envio deles até a central para sua exibição, foi utilizada a placa ESP-32 (Fig. 14), da *Heltec Automation*, que conta com comunicação LoRa através de um transceptor (modelo SX1276) ligado à placa, Wi-Fi e Bluetooth, além de um display OLED que possibilita a leitura de informações úteis do processo de medição, transmissão e recepção dos dados.

Nas tabelas 3 e 4 é possível visualizar as especificações técnicas do módulo e do transceptor LoRa.

Tabela 2 – Especificações do da placa ESP-32 LoRa.

|  |  |
| --- | --- |
| Tensão de Operação | 2,2 V a 3,6 V |
| *Clock* do processador | 240 MHz |
| Memória interna | 502 KB |
| Corrente de consumo típica | 80 mA |
| Corrente de consumo máxima | 500 mA |

Fonte: FILIPEFLOP, MERCADO LIVRE.

Tabela 3 – Especificações do transceptor LoRa.

|  |  |
| --- | --- |
| Frequência de operação | 915 MHz |
| Sensibilidade | -139 dBm (máximo) |
| Tensão de alimentação | 1,8 V a 3,7 V |
| Corrente de consumo (recepção) | 10,8 mA |
| Corrente de consumo (transmissão) | 120 mA |
| Potência máxima | 20 dBm |
| Modos de modulação | LoRa, FSK, GFSK e OOK |
| Temperatura de trabalho | -40ºC a 85ºC |

Fonte: MERCADOLIVRE, GITHUB.

* + 1. **DHT 22**

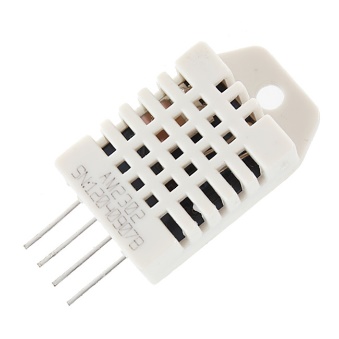
Para realizar as medições de temperatura do ambiente, o sensor escolhido foi o DHT22, capaz de medir tanto a temperatura como a umidade relativa do ar. É a versão mais precisa da linha DHT, sucessor do DHT11. A tabela 5 apresenta as características do sensor seguida pela figura 14, que mostra seu encapsulamento.

Tabela 4 – Especificações do sensor DHT22.

|  |  |
| --- | --- |
| Tensão de operação | 3-5VDC (5,5VDC máximo) |
| Faixa de medição de umidade | 0 a 100% UR |
| Faixa de medição de Temperatura | -40º a +80ºC |
| Corrente | 2,5mA máx. durante uso |
| Precisão de medição da umidade | ± 2,0% UR |
| Precisão de medição da temperatura | ± 0,5 ºC |

Fonte: FILIPEFLOP.

Figura 21 – Sensor DHT22.

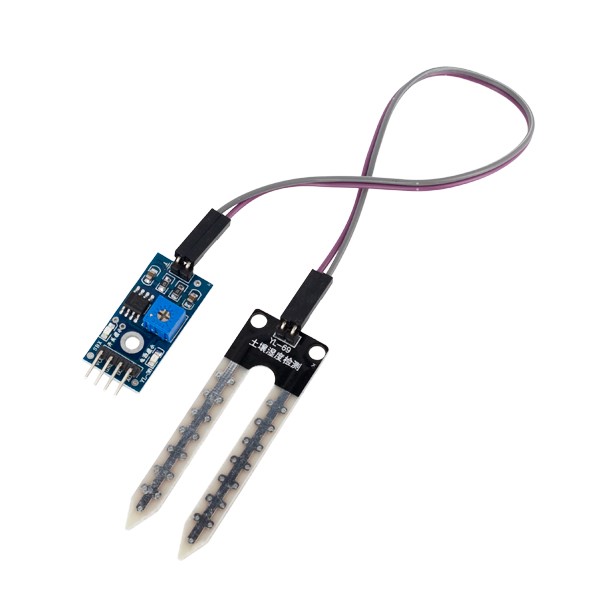


Fonte: FILIPEFLOP, 2021.

* + 1. **Sensor de Umidade do Solo**

Para realizar as medições da umidade do solo, foi escolhido o sensor analógico, mostrado na figura 16.

Figura 22 – Sensor de umidade do solo.



A tabela 6 apresenta as características eletrônicas e de medição do sensor.

Tabela 5 – Especificações do sensor de umidade do solo.

|  |  |
| --- | --- |
| Tensão de operação | 3,3V até 5V |
| Precisão de medida | Regulável com o potenciômetro |

Fonte: FILIPEFLOP, 2021.

A tabela 7 lista os componentes utilizados no projeto e suas quantidades.

Tabela 6 – Número de componentes utilizados.

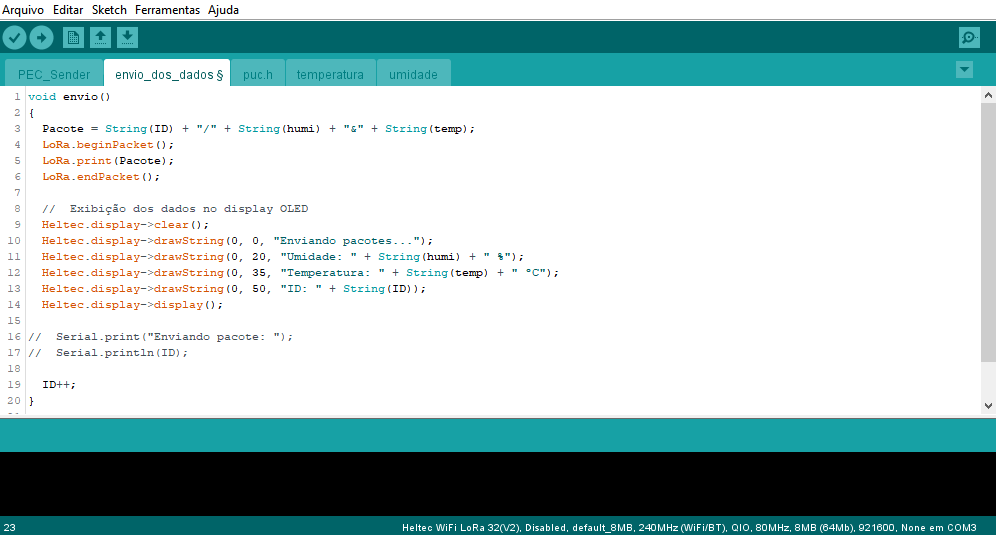
|  |  |
| --- | --- |
| Componentes | Quantidades |
| Placa ESP32 LoRa | 3 |
| Antenas | 3 |
| Sensor DHT22 | 1 |
| Sensor de Umidade do Solo | 1 |
| Resistor 10kΩ | 1 |
| Protoboard | 2 |
| Computador | 1 |

Fonte: AUTORIA PRÓPRIA.

* 1. **Recursos de *Software***

Para o desenvolvimento do *software* do projeto, é necessária uma interface para que seja possível programar as funções das placas em que estão localizadas os microcontroladores e sensores. Isso pode ser feito através de um ambiente integrado de desenvolvimento chamado “Arduino IDE” (Fig. 15), que reconhece quando a placa está conectada ao computador e permite que seja programada a partir de linhas de código baseadas em C++.

Figura 23 – Arduino IDE.



Fonte: AUTORIA PRÓPRIA, 2021.

Além da interface para a programação das placas, é necessário um ambiente de desenvolvimento para a exibição dos dados, que será feita através de um *dashboard*. Para isso, foi escolhida a ferramenta Django na versão 3.2.3, que trata de uma forma comum e atual para a combinação entre funções de linguagem Python e JavaScript, que possibilita a elaboração de páginas *web*, contendo os dados obtidos.

# METODOLOGIA

A metodologia utilizada na elaboração do projeto consiste em três grandes partes: coleta dos dados pelos sensores, transmissão dos dados através do protocolo LoRa e desenvolvimento da plataforma de gerência e monitoramento.

* 1. **Coleta dos dados**

A partir da programação do sensor DHT22, é definido que a cada 1 segundo, será realizada uma medida da temperatura e umidade relativa do ar e exibido no *display* OLED presente na placa (Fig. 16). Também é adicionado ao esquemático do circuito um resistor de 10kΩ, que tem a função de assegurar que os valores obtidos não sejam corrompidos por quaisquer interferências eletrônicas.

Figura 24 – ESP-32 LoRa e DHT22 em operação.

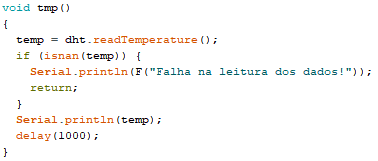
Imagem de jogo de vídeo game

Descrição gerada automaticamente com confiança baixa

Fonte: AUTORIA PRÓPRIA.

Para isso, a placa utiliza uma biblioteca pronta que identifica o sensor DHT22 e faz a leitura dos valores a partir da função “dht.readTemperature()”, presente no módulo “tmp()”, responsável pela leitura e inserção dos valores na parte principal do código (Fig. 17).

Figura 25 – Código para leitura dos valores.



Fonte: AUTORIA PRÓPRIA.

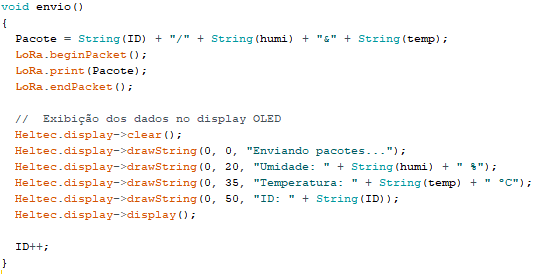
Após a identificação dos dados, é feita uma avaliação sobre sua qualidade, se realmente significam valores possíveis, através da função “isnan”, que basicamente verifica se o número obtido é um número. Caso não seja, é avisado ao usuário na porta serial do seu computador que o valor é inválido.

Para a medição dos valores de umidade relativa, é realizado o mesmo procedimento, com as mesmas funções utilizadas para a temperatura, apenas com a mudança dos nomes para as variáveis.

* 1. **Transmissão e recepção dos dados através do protocolo LoRa**

A partir do momento em que os dados de temperatura e umidade são obtidos, eles são empacotados em um formato que o protocolo LoRa consiga enviá-los. Para isso, é feita uma combinação dos valores lidos em uma palavra composta pelos *bits*, separados por símbolos como “/” e “&”, a fim de que a placa responsável pela recepção possa identificá-los e separá-los para serem exibidos de forma correta. A figura 18 ilustra o trecho de código utilizado para o empacotamento dos dados.

Figura 26 – Empacotamento dos valores de temperatura e umidade.

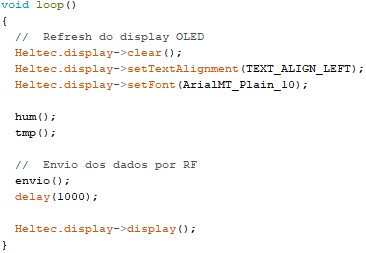


Fonte: AUTORIA PRÓPRIA.

Através das funções “LoRa.beginPacket()” “LoRa.print()” e “LoRa.endPacket()”, o pacote com os dados de temperatura e umidade são montados. Há, também, a variável “ID”, que contém um número inteiro (incrementado em 1 toda vez que o módulo é chamado na função principal) que representa a identificação de cada pacote, para que na recepção, seja possível organizá-lo de forma correta.

Após o encapsulamento, os dados que serão enviados e seus respectivos números identificadores são exibidos no *display* da placa, para que o usuário tenha a certificação de todo o processo. Enfim, os valores são inseridos no código principal que é executado em “*looping*” (enquanto o dispositivo estiver ligado) e são transmitidos através do LoRa (Fig. 19).

Figura 27 – Parte do código executada em looping.

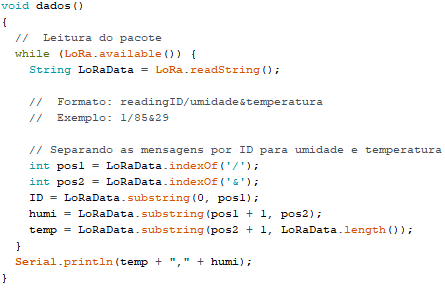


Fonte: AUTORIA PRÓPRIA.

Nessa seção, o *display* da placa sofre atualizações para que os novos valores sejam exibidos a cada *loop*. Em seguida, são “chamadas” as funções responsáveis por obter os valores de umidade (função “hum()”, derivada da palavra “*humidity*”, que significa “umidade” em inglês) e temperatura (função “tmp()”, derivada da palavra “*temperature*”, que significa “temperatura” em inglês). Por fim, é executada a função “envio()”, que empacota os dados e faz a transmissão ao nó receptor.

Para a recepção dos dados, é necessário o processo de decodificação da palavra recebido, que contém os dados de identificação do pacote, umidade e temperatura, realizado através do módulo “dados()”, mostrado na figura 20.

Figura 28 – Módulo de decodificação dos dados.



Fonte: AUTORIA PRÓPRIA.

Primeiramente, o módulo verifica se a comunicação LoRa está disponível entre os pontos. Caso o resultado seja positivo, através da função “LoRa.readString()”, a palavra é lida e, a partir da função “indexOf()”, cada símbolo utilizado para a representação dos valores é identificado. Nesse momento, a aplicação separa por posições os valores de “ID” umidade e temperatura. Em seguida, os valores são exibidos (com o espaçamento de uma linha entre cada nova entrada) na porta serial do computador em que o dispositivo está conectado, através da função “Serial.println()”.

A partir desse momento, encerra-se a programação em linguagem C++ e inicia-se a análise dos dados com a utilização de código em linguagem Python, através da ferramenta Django. Para isso, é criada uma função que faz a leitura da entrada de valores na porta serial selecionada (fig. 21).

Figura 29 – Leitura da entrada de valores com Python.



Fonte: AUTORIA PRÓPRIA.

No cabeçalho do código, são feitas importações de bibliotecas da própria ferramenta Django, responsáveis por toda a comunicação entre as partes do núcleo da aplicação. Também são realizadas importações de bibliotecas como a “serial”, responsável por fazer o acesso à porta serial do computador (no exemplo, a porta “COM3”), a “time” que é de extrema importância para a realização de intervalos de tempo entre funcionalidades e, por fim, a “json”, responsável por formatar os dados lidos em um tipo de objeto capaz de ser identificado por linguagem JavaScript, para que seja inserido nas páginas *web*.

A função “index()” é responsável apenas por renderizar uma página em formato html, que serve de base para que os dados sejam exibidos de maneira simples e compreensível.

A função “getData()” representa todo o processo de leitura dos dados que chegam à porta serial COM3, através do método “readline()”, seguido pelo “decode()”. O primeiro trata da leitura de uma linha da porta serial e o segundo decodifica os *bits* lidos para o padrão de codificação de caracteres UTF-8, transformando-os em números legíveis e armazenando-os na variável “sensorData”. Os dados então são colocados em um vetor (“*array*”) e separados por vírgulas que determinam suas posições.

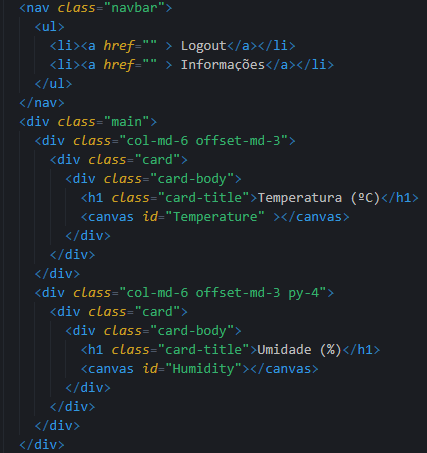
Em seguida, são colocados no vetor “data”, com suas respectivas posições e encapsulados no formato *json* que é, então, passado como parâmetro para a renderização da página *web*, para que sejam exibidos na tela.

A última parte do processo é construída pelas linguagens HTML e JavaScript, responsáveis por tratar os dados de forma que sejam exibidos em uma página *web*, utilizando uma tecnologia que interpreta os dados formatados em *json* e realiza atualizações incrementais neles, possibilitando a leitura em tempo real dos valores de temperatura e umidade.

A página *web* (no formato “.html”) é constituída por *tags* que indicam palavras-chave, para que o interpretador presente no navegador de internet possa identificá-las e transformá-las em elementos separados em cabeçalho, corpo da página e rodapé. Em cada seção podem ser inseridos elementos visuais, como títulos (representados pela *tag* “<h1>”).

A figura 22 representa o “esqueleto” de uma página HTML.

Figura 30 – Representação do código de uma página web.

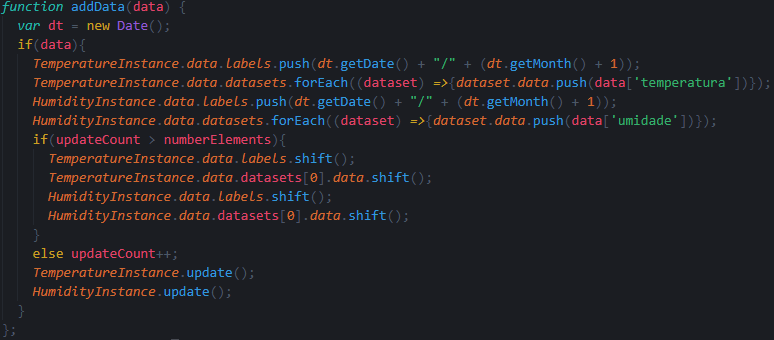


Fonte: AUTORIA PRÓPRIA.

Além dos elementos visuais, o corpo de uma página suporta a inserção de *tags* do tipo “<script>”, responsáveis por comportar códigos em linguagem JavaScript, que interpreta os dados da aplicação em Python e, também, pode realizar operações com elementos das páginas, entre outras diversas funções. Isso acaba agregando à aplicação uma versatilidade muito grande.

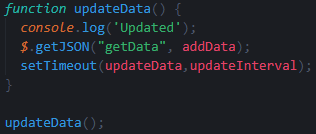
Na parte do *script*, é introduzida a função “addData()” (fig. 23) que é responsável por popular os primeiros dados obtidos nos gráficos de temperatura e umidade através do método AJAX – presente na função “updateData()” (fig. 24), que consiste na interpretação e requisição dos dados formatados no tipo *json*, para que sejam exibidos na tela.

Figura 31 – Função de inserção dos primeiros dados.



Fonte: AUTORIA PRÓPRIA.

Figura 32 – Função de atualização incremental dos dados.



Fonte: AUTORIA PRÓPRIA.

Assim que os primeiros dados são populados pela primeira função (“addData()”), a segunda função (“updateData()”) realiza a atualização incremental dos valores em tempo real, a partir de um intervalo de 4 segundos (“*updateInterval*”).

Assim que a leitura é completa e os dados são requisitados, os valores são exibidos em gráficos de Temperatura por tempo e Umidade por tempo separadamente mostrados pelas figuras 25 e 26.

Figura 33 – Gráfico para valores de temperatura.

Gráfico

Descrição gerada automaticamente

Fonte: AUTORIA PRÓPRIA.

Figura 34 – Gráfico para valores de umidade.

Gráfico, Gráfico de linhas

Descrição gerada automaticamente

Fonte: AUTORIA PRÓPRIA.

# RESULTADOS

# CONCLUSÃO

# REFERÊNCIAS