

Abajur Inteligente com previsão do tempo

Felipe Ferreira Martins da Silva
Área temática: Automação Residencial

RESUMO

A experiência adquirida nesse projeto evidencia a importância da integração entre hardware e software em sistemas embarcados aplicados à Internet das coisas (IoT). O objetivo central do projeto foi desenvolver um Abajur inteligente com previsão do tempo, utilizando a BitDogLab, permitindo o controle de iluminação (ligar e desligar) e evidenciar dados climáticos obtidos pela API OpenWeather. A metodologia envolveu implementação de uma matriz de LEDs, um display OLED e conexão Wi-Fi para exibição de informações meteorológicas. Os resultados demonstram um sistema funcional, com atualização periódica dos dados e uma interface intuitiva, destacando seu potencial para futuras expansões e aprimoramentos.

Palavras-chave: Sistemas embarcados. Iluminação. Wi-Fi. Previsão do tempo.

1. INTRODUÇÃO

Com o avanço da tecnologia e a crescente digitalização dos dispositivos eletrônicos, a automação residencial tem se tornado uma tendência cada vez mais presente no cotidiano. A Internet das Coisas (IoT) possibilita a criação de soluções que tornam o dia a dia mais prático e eficiente, permitindo que aparelhos eletrônicos interajam entre si e com o usuário. No entanto, muitos sistemas de iluminação ainda operam de forma manual, sem integração com dados ambientais, o que pode gerar desperdício de energia e falta de adaptação às necessidades do usuário.

Pensando nisso, este projeto propõe o desenvolvimento de um Abajur Inteligente com Previsão do Tempo, utilizando a BitDogLab e a. O objetivo principal é oferecer uma experiência mais interativa e funcional ao usuário, permitindo que a iluminação se ajuste automaticamente com base nas condições meteorológicas. O sistema faz uso de um display OLED para exibição de informações climáticas e botões físicos para interação, garantindo um controle intuitivo e acessível.

Além disso, a matriz de LEDs RGB (que simula o abajur) é capaz de alterar suas cores conforme a previsão do tempo, indicando visualmente se há possibilidade de chuva, garantindo um uso mais dinâmico e útil da iluminação. O projeto também visa demonstrar a aplicação prática da integração entre hardware e software

dentro do conceito de IoT, abrindo possibilidades para futuras melhorias, como integração com assistentes virtuais e controle via aplicativos móveis.

Com essa abordagem, o sistema não apenas promove o aprendizado em sistemas embarcados, mas também reforça a importância da IoT no cotidiano, mostrando como soluções inteligentes podem tornar o ambiente mais eficiente e interativo



Fig 1 – Menu interativo

2. METODOLOGIA

Esta seção detalha os recursos utilizados no desenvolvimento do projeto, abrangendo hardware, software, tecnologias de comunicação e os métodos de teste e validação empregados.

2.1 Hardware Utilizado

O hardware utilizado no projeto foi escolhido com o objetivo de garantir eficiência e compatibilidade entre os componentes. A lista de dispositivos inclui:

- BitDogLab com Raspberry Pi Pico W: responsável pelo processamento e controle do sistema.
- Matriz de LEDs WS2812B: utilizada para representar visualmente as condições climáticas.
- Display OLED SSD1306 (I2C): exibição de informações sobre a previsão do tempo.
- Botões GPIO (5 e 6): utilizados para o controle do abajur e interação do usuário

2.2 Software Utilizado

A programação do sistema foi realizada em Linguagem C com Pico SDK, utilizando bibliotecas específicas para comunicação e manipulação de dados. As principais tecnologias utilizadas incluem:

- Biblioteca lwIP: implementação da comunicação HTTP para obtenção dos dados climáticos.
- OpenWeather: fonte dos dados climáticos utilizados no sistema
- cJSON: manipulação e extração de informações do JSON recebido pela API OpenWeather
- ThingSpeak: armazenamento e monitoramento dos dados meteorológicos.

2.3 Abordagem de desenvolvimento

O desenvolvimento do sistema foi estruturado em três etapas principais: Conexão e teste inicial dos componentes, garantindo o funcionamento adequado de cada um. Implementação da lógica de controle do abajur, integração com API OpenWeather e configurações dos periféricos. Avaliação do funcionamento do sistema, ajustes na precisão das informações exibidas e otimização da comunicação Wi-Fi.

2.4 Estratégias de Teste e Validação

Os testes foram conduzidos para garantir a confiabilidade e precisão do sistema, avaliando o funcionamento da comunicação com a API, a exibição correta das informações no display e a resposta do abajur às condições climáticas simuladas.

Os principais aspectos avaliados incluíram:

Comparação de dados da API: Os valores de temperatura e umidade recebidos da API OpenWeather foram comparados com outras fontes externas, como previsões do Google Weather e aplicativos meteorológicos nativos do sistema operacional do celular. Essa verificação ajudou a validar a consistência dos dados obtidos.

Precisão da exibição no abajur: Foi testado se as cores da matriz de LEDs mudavam corretamente conforme a previsão meteorológica simulada. Como a API fornece apenas temperatura e umidade, foi utilizada uma fórmula baseada na relação entre esses fatores para estimar a possibilidade de chuva.

Cálculo do ponto de orvalho: A equação utilizada para a estimativa de formação de orvalho, neblina ou chuva foi:

$$Td = T - \frac{(100 - UR)}{5}$$

Onde:

Td = Ponto de orvalho (em °C)

T = Temperatura do ar (em °C)

UR = Umidade relativa do ar (em %)

O ponto de orvalho é um indicativo da umidade presente na atmosfera. Se a temperatura do ar estiver próxima ao ponto de orvalho, a umidade relativa está alta e há maior probabilidade de condensação, o que pode resultar em chuva.

Essa estimativa ajuda a prever a formação de orvalho, neblina ou até mesmo a chance de chuva.

Se a temperatura do ar estiver próxima ao ponto de orvalho, a umidade relativa está alta e há grande chance de condensação.

Se UR for alta (>85%) e a diferença entre T e Td for pequena (<2°C), há grande probabilidade de chuva.

Se UR for moderada (70-85%) e a diferença for pequena (<4°C), pode haver chuva leve ou neblina.

Como o sistema não possui sensores específicos para pressão atmosférica, velocidade do vento e nebulosidade, essa previsão deve ser interpretada como uma simulação aproximada. Para maior precisão, seria necessária a aquisição de dados adicionais, como pressão barométrica e velocidade do vento, permitindo cálculos meteorológicos mais avançados. Dessa forma esse cálculo não reflete exatamente no comportamento do mundo real.

3 DESENVOLVIMENTO

3.1 Estrutura do sistema

O projeto foi desenvolvido de forma modular, garantindo separação de funcionalidades principais em arquivos distintos:

Projeto_embarcatech.c – Onde se encontra o código principal.

wifi_clima.c – Onde se faz a conexão Wi-Fi e a aquisição de dados da OpenWeather.

matriz_led.c – Onde é feita a comunicação com a matriz de LED utilizada para simular o Abajur.

ssd1306.c – Interface com display OLED.

3.2 Comunicação HTTP com a API OpenWeather e ThingSpeak

O sistema embarcado utiliza o protocolo HTTP (Hypertext Transfer Protocol) para realizar uma requisição GET à API OpenWeather, obtendo informações climáticas em tempo real. Para iniciar essa comunicação a Raspberry Pi Pico W precisa estar conectada à internet, que é feito utilizando a biblioteca lwIP, que permite a configuração da interface de rede e a conexão a um roteador Wi-Fi. A requisição é montada seguindo o formato exigido pela API. O sistema gera uma URL contendo os parâmetros necessários, sendo assim, o código estabelece uma conexão TCP com o servidor da API e envia uma requisição HTTP GET, solicitando os dados climáticos, a API retorna um JSON contendo as informações meteorológicas da cidade requisitada. Após receber os dados em JSON é utilizada a biblioteca cJSON, para fazer a extração desses dados, Cidade, Temperatura e Umidade, esses valores são armazenados convertidos para uma String, onde podem ser utilizados para exibição no display OLED e para o controle da matriz de LEDs, ajustando a cor de acordo com a previsão.

O ThingSpeak é um serviço baseado em IoT que permite o armazenamento, visualização e análise de dados em tempo real também usa o protocolo de comunicação HTTP, antes de enviar as informações para o ThingSpeak o sistema processa os dados relevantes, no caso do projeto, Temperatura do ambiente, Umidade do ar e estado do abajur (Ligado ou Desligado), esses dados são armazenados em variáveis que respectivamente são enviados para a plataforma e a cada 30 segundos utilizando um temporizador.

3.3 Fluxograma do código

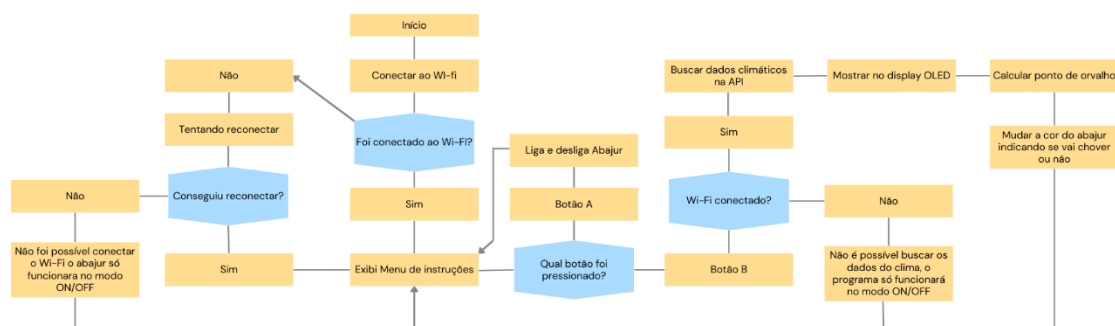


Fig 2 - Fluxograma

4 CONCLUSÃO

O desenvolvimento do Abajur inteligente com Previsão do Tempo atingiu com sucesso os objetivos propostos, demonstrando viabilidade de integração entre hardware e software para automação residencial baseada em IoT. A utilização Da BitDogLab e da Raspeberry Pi Pico W permitiu criar um sistema funcional que ajusta sua iluminação com base nas condições climáticas.

Os resultados obtidos confirmaram a capacidade do sistema de coletar dados meteorológicos em tempo real via API OpenWeather, exibi-los no display OLED e ajustar a matriz de LEDs de acordo com a previsão do tempo. A implementação de um temporizador para envio de dados de forma automática para o ThingSpeak.

Durante o desenvolvimento, alguns desafios foram enfrentados, como a necessidade de otimizar a conexão Wi-Fi, para evitar desconexões e a implementação de uma fórmula simplificada para previsão do tempo. Essas limitações podem ser superadas em versões futuras com a aquisição de dados mais complexos para melhorar o cálculo da previsão do tempo, e adicionar um temporizador para atualizar o clima periodicamente, na forma atual toda vez que o botão b é pressionado ele atualiza o dado do clima, mas uma futura melhoria é adicionar um temporizador para que esse dado seja atualizada separadamente do fluxo do código principal.

Outra melhoria futura é expandir o sistema para ter controle via aplicativo móvel, integração com assistentes virtuais e uso de machine learning para prever condições climáticas com maior precisão, essas adições tornaria o abajur ainda mais inteligente e adaptável às necessidades do usuário.

De tal maneira que mais uma vez esse projeto reforça a importância da IoT na automação residencial, demonstrando seu potencial para criar soluções inovadoras, eficientes e acessíveis. O aprendizado adquirido com a implementação desse sistema contribui não apenas para a prática de sistemas embarcados, mas também para o desenvolvimento de novas aplicações que podem transformar a forma como interagimos com o ambiente doméstico.

REFERÊNCIAS

1. COSTA, Judson dos Santos. **Arquitetura de um sistema para automação residencial baseado em IoT**. Natal: Universidade Federal do Rio Grande do Norte, 2022. Acesso em: 11 fev. 2025.
2. SOUZA, Thales Ruano Barros de et al. **Residential smart plug with bluetooth communication**. arXiv preprint arXiv:2103.15757, 2021. Acesso em: 12 fev. 2025.
3. BOHARA, Bharat; MAHARJAN, Sunil; SHRESTHA, Bibek Raj. **IoT Based Smart Home using Blynk Framework**. arXiv preprint arXiv:2007.13714, 2020. Acesso em: 13 fev. 2025.
4. SILVA, Vinícius Vieira da et al. **Produto educacional: automação residencial com uso de Arduino e IoT**. ResearchGate, 2022. Acesso em: 14 fev. 2025.
5. ALMEIDA, Alexia Nicole Pereira de et al. **Aplicações de IoT em automação residencial para idosos e pessoas com mobilidade reduzida**. Manaus: Universidade do Estado do Amazonas, 2023. Acesso em: 15 fev. 2025.
6. SOUZA, Thales Ruano Barros de et al. **Automação residencial com uso de Arduino e IoT**. ResearchGate, 2022.
7. COSTA, Judson dos Santos. **Arquitetura de um sistema para automação residencial baseado em IoT**. Natal: Universidade Federal do Rio Grande do Norte, 2022.
8. SOUZA, Thales Ruano Barros de et al. **Residential smart plug with bluetooth communication**. arXiv preprint arXiv:2103.15757, 2021.
9. SILVA, Vinícius Vieira da et al. **Produto educacional: automação residencial com uso de Arduino e IoT**. ResearchGate, 2022.