



INSTITUTO FEDERAL

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

Campus: Campina Grande

Curso: Engenharia de Computação

Disciplina: Sistemas Embarcados

Discente: Alexandre Sales Vasconcelos

USO DA ESP32 PARA A MEDIÇÃO DE TEMPERATURA E UMIDADE DE UM TERRÁRIO PARA JABUTI PIRANGA

Maria Luíza Teixeira Nascimento

Renata da Silva Ferreira

Rayanne Kelly Marcelino Barros Elias

Dezembro de 2022

1 INTRODUÇÃO

A criação de um pequeno ecossistema natural, composto por componentes bióticos, sendo eles os seres vivos, e por componentes abióticos, como os fatores físicos, químicos e geológicos do ambiente, culminam na criação de um terrário.

Os terrários conseguem imitar em um espaço protegido, as características do habitat natural de répteis e anfíbios, como lagartos, lagartixas, tartarugas e até serpentes. Este projeto em questão, limita-se a ressaltar o controle dos terrários para o jabuti piranga.

Nota-se que dependendo do animal, alguns fatores são importantes para a manutenção da vida dos seres vivos dentro de um terrário, como por exemplo, a fiscalização da temperatura do ambiente e da umidade do terrário.

Este projeto de pesquisa concentra-se a um terrário simples, com a principal fonte de manutenção baseado em fatores humanos. E erros humanos são mais passíveis de ocorrer, e esses erros, poderá ter como resultado um desequilíbrio na temperatura e umidade. Consequentemente dentro de um sistema, culmina em muitos casos, na morte do jabuti e de outros seres vivos.

Sendo assim, para aqueles que desejam adquirir ou criar um terrário, é de fundamental importância dedicar tempo e atenção para assegurar a manutenção da temperatura e umidade. Entretanto, essa dedicação diária pode tornar-se um empecilho de tempo e esforço.

Uma solução possível, é a automatização do processo de inspeção das condições apresentadas utilizando um sistema embarcado, isto é, um sistema microprocessado em que um computador é anexado ao sistema que ele controla, visando realizar um conjunto de tarefas predefinidas, juntamente com o sensor de umidade e temperatura DHT22.

O microcontrolador ESP 32 foi escolhido por ser uma alternativa viável. Com módulos encontrados no mercado com um baixo custo e consumo de energia e um alto nível de integração com outros componentes, além de Wifi e Bluetooth. Assim, a ESP 32 e o sensor de umidade e temperatura DHT22, surgem como uma solução para a automatização do processo de verificação de temperatura e umidade de um terrário, auxiliando a rotina do criador do terrário e visando assegurar as condições de vida favoráveis para os jabutis.

2 OBJETIVOS

2.1 OBJETIVO GERAL

O objetivo geral deste trabalho é desenvolver um sistema embarcado capaz de realizar a medição de parâmetros como umidade e temperatura de um terrário para o **Jabuti Piranga**. Além disso, visa facilitar a comunicação via internet quando esses parâmetros estiverem fora dos limites determinados e permitir ao usuário modificar os mesmos de acordo com as necessidades observadas, através de uma LED UV.

2.2 OBJETIVOS ESPECÍFICOS

- Possibilitar a medição do nível de umidade do terrário, observando se os dados obtidos estão de acordo com os parâmetros adequados para o jabuti.
- Assegurar a verificação da temperatura do terrário para manter-se dentro dos limites previstos.
- Viabilizar o acesso a informações atualizadas sobre o ambiente do terrário, bem como a oportunidade de ajustar parâmetros de temperatura e umidade através do aplicativo do Telegram.

3 MATERIAIS E MÉTODOS

3.1 MÉTODOS E TÉCNICAS UTILIZADAS

O desenvolvimento do projeto deu-se baseado em fluxogramas, ou seja, uma série de passos que ajudou a refletir como seria melhor maneira de elaborar e desenvolver os códigos de modo mais assertivo, com foco em entregar um código de qualidade que atendesse ao que era esperado. Assim, começamos com a criação de um diagrama de blocos, utilizando o **Paint**.

Após a criação do diagrama de blocos, começamos a refletir sobre quais seriam os componentes mais adequados, começamos com o **ESP 32 WROOM WI-FI**, o sensor de umidade e temperatura **DHT22**, utilizamos também uma **LED UV**, como sendo os componentes essenciais para o projeto.

Ao chegarmos nos códigos, utilizamos o editor de código **VS Code**, com a extensão do **Espressif IDF**, que possibilitou uma comunicação facilitada entre os desenvolvedores com a placa ESP 32 WROOM WI-FI, no qual conseguimos aproveitar melhor as possibilidades que o microcontrolador e seus periféricos ofereciam.

Durante a escrita do código, uso de algumas **bibliotecas** e **recursos** como mostradas na tabela a seguir, contribuíram para o melhor entendimento e aproveitamento do microcontrolador. Sendo elas, bibliotecas prontas ou criadas para este fim específico:

dht.c	dht.h
http_server.c	http_server.h
rgb_led.c	rgb_led.h
wifi_app.c	wifi_app.h
esp_log.h	freertos/FreeRTOS.h
freertos/task.h	esp_system.h
esp_err.h	esp_wifi.h
nvs_flash.h	

Tabela 01: Algumas bibliotecas usadas.

3.2 COMPONENTES UTILIZADOS

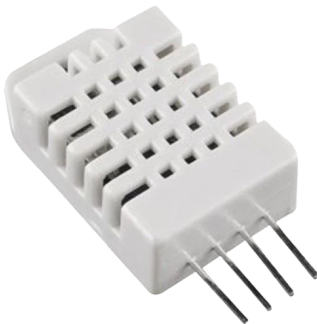
Módulo WiFi ESP32 Bluetooth



Especificações:

- ESP32 ([Datasheet - ESP 32 Wroom Wi Fi](#))
- CPU: Xtensa® Dual-Core 32-bit LX6
- ROM: 448 KBytes - RAM: 520 Kbytes - Flash: 4 MB
- Wireless padrão 802.11 b/g/n
- Conexão Wifi 2.4Ghz (máximo de 150 Mbps)
- Antena embutida
- Conector micro-usb
- Wi-Fi Direct (P2P), P2P Discovery
- Bluetooth BLE 4.2
- Portas GPIO: 11, com funções de PWM, I2C, SPI, etc
- Conversor analógico digital (ADC)

Sensor DHT22



Especificações:

- Modelo: DHT22 ([Datasheet - Sensor DHT22](#))
- Alimentação: 3,0 a 5,0 VDC (5,5 Vdc máximo)
- Corrente: 200uA a 500mA, em stand by de 100uA a 150 uA
- Faixa de medição de umidade: 20 a 90% UR
- Faixa de medição de temperatura: 0° a 50°C
- Precisão de umidade de medição: $\pm 5,0\%$ UR
- Precisão de medição de temperatura: ± 2.0 °C
- Tempo de resposta: $< 5s$

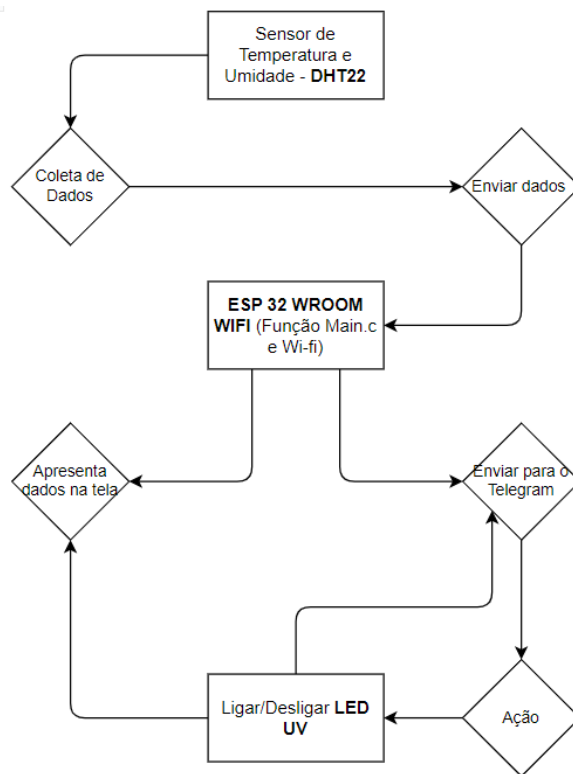
Led vermelho difuso de 5 mm



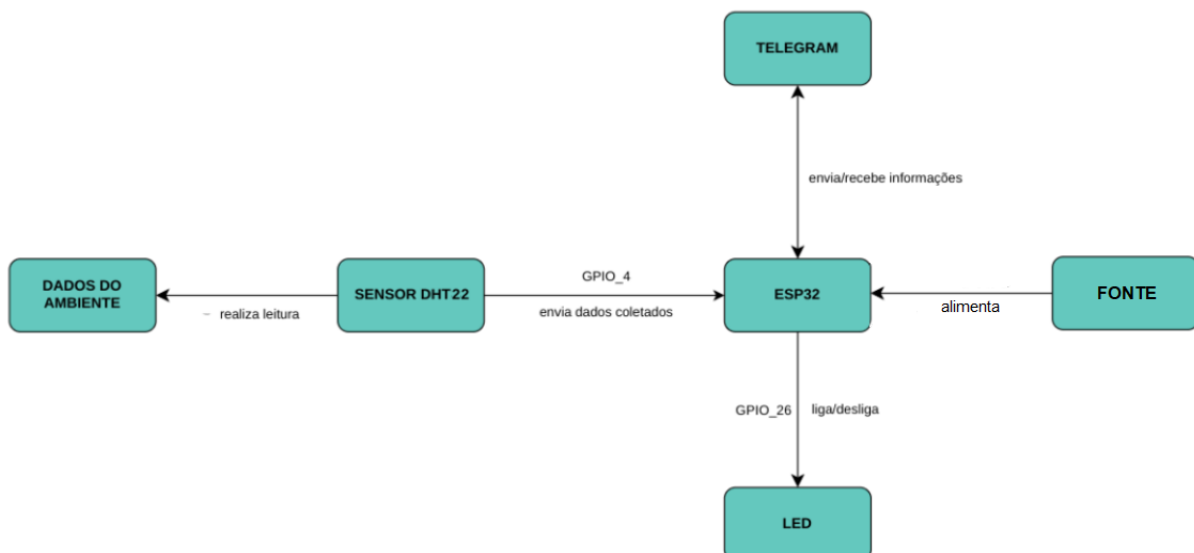
Especificações:

- Datasheet: [Datasheet LED Vermelha](#)
- Cor: Vermelho
- Diâmetro: 5mm
- Tensão: 2V
- Corrente: 20mA

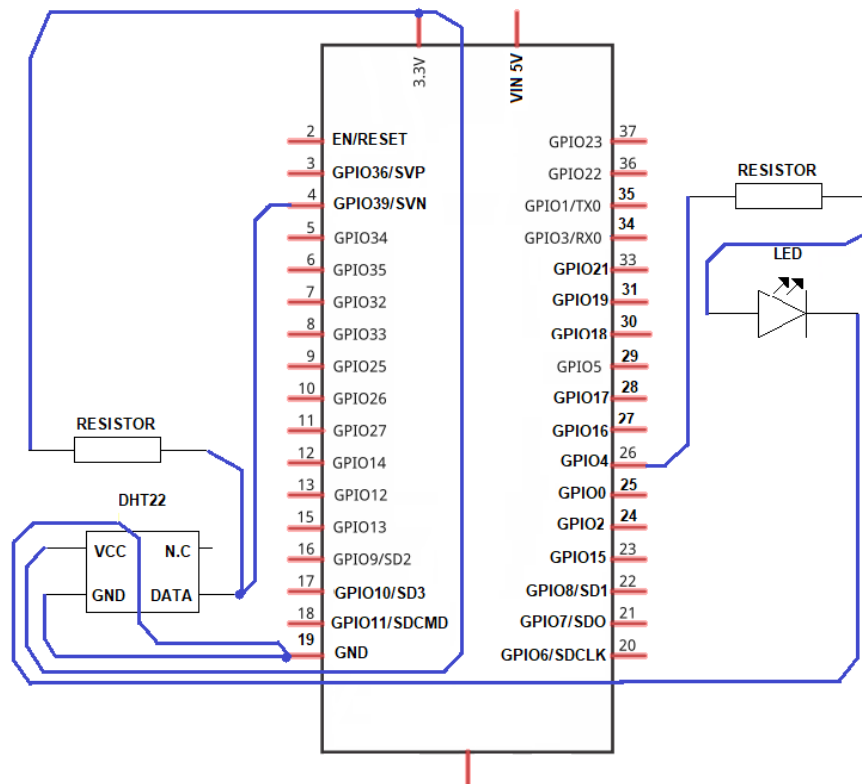
3.3 FLUXOGRAMA



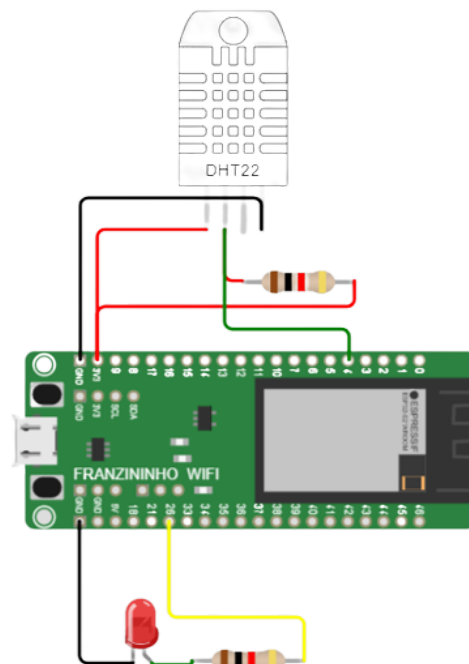
3.4 DIAGRAMA DE BLOCOS



3.5 CIRCUITO ELÉTRICO



3.6 CIRCUITO ELÉTRICO PROTOBOARD



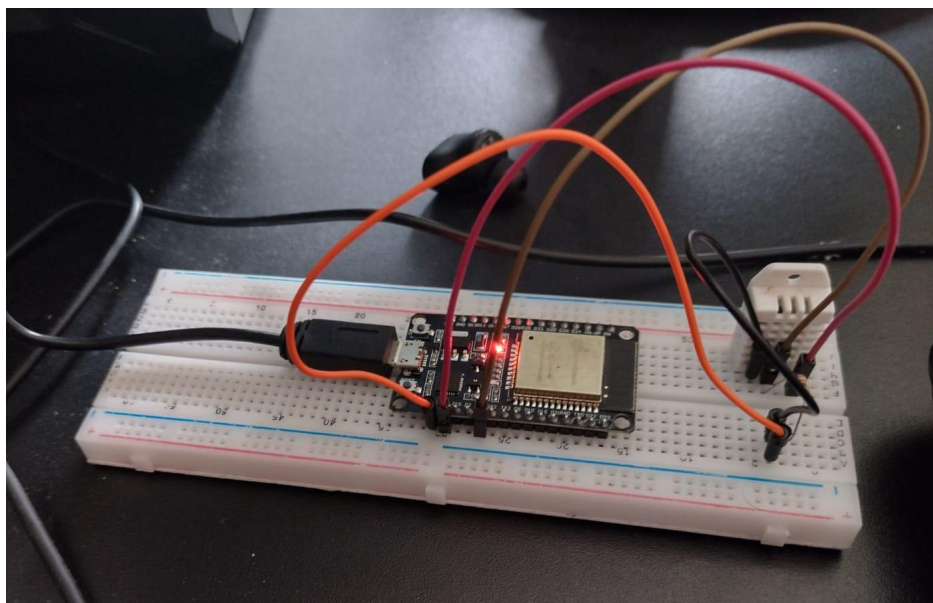
4 RESULTADOS

Neste projeto, apesar dos empecilhos que tivemos, em relação às configurações do ambiente de desenvolvimento e de encontrar materiais disponíveis de qualidade que melhor auxiliassem o desenvolvimento do projeto. Foi possível elaborar um protótipo simples, por causa do tempo estimado e de baixo custo, com o uso da ESP 32 WROOM e com o sensor DHT22, com a função de realizar o controle de temperatura e umidade, ambientado em um terrário para Jabuti Piranga. .

Além disso, o sistema consegue realizar a comunicação com o meio externo e repassá-lo para o usuário. Este usuário possui o acesso da temperatura e umidade em seu aplicativo de Telegram, permitindo também a verificação e controle do terrário, em pequenas e curtas distâncias necessitando apenas uma conexão com a internet, já que faz o uso do recurso Wi-Fi da placa do ESP 32 WROOM.

5 FOTOS DOS RESULTADOS





6 REFERÊNCIAS BIBLIOGRÁFICAS

CABANAC, Michel; BERNIERI, Cécile. Behavioral rise in body temperature and tachycardia by handling of a turtle (*Clemmys insculpta*). **Behavioural Processes**, v. 49, n. 2, p. 61-68, 2000.

FERREIRA, Maria Fernanda Ribeiro et al. Terrário como instrumento didático-pedagógico para o ensino sobre ecossistema. **Scientia Amazonia**, v. 9, n. 1, p. 10-15, 2020.

GONÇALVES, Vera et al. TERRÁRIO ECOSSISTEMA EM MINIATURA. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão**, v. 7, n. 1, 2015.

NEGRO, NA BACIA DO RIO. CONSERVANDO AS TARTARUGAS.

SANTOS, Rodrigo Jeremias et al. Estudo de Sistema de Controle automático para aquário utilizando uma plataforma de Prototipagem Eletrônica open source.. *Revista Ibero-Americana de Humanidades, Ciências e Educação*, v. 8, n. 4, p. 569-586, 2022.

VON GILSA, Vítor Alonzo Bordinhon et al. UM SUCINTO ECOSSISTEMA: ENTENDA COMO FUNCIONA UM TERRÁRIO. **Anais da Feira de Ciência, Tecnologia, Arte e Cultura do Instituto Federal Catarinense do Campus Concórdia**, v. 5, n. 1, p. 90-90, 2022.