

Projeto SE 2024.1

Equipe

- Lucas Bivar Fonseca Tavares
- João Victor Negreiros da Silva
- Luís Henrique Lima Santos
- Lucas Alves Fidelis Araújo
- Lohan Yrvine Oliveira Pinheiro

Sobre o Projeto

Itens a Serem Entregues

- Firmware contendo um exemplo de utilização da biblioteca.
- Máquina de estado do firmware.
- Documentação da biblioteca.
- Diagrama de bloco para o protótipo do hardware.
- Esquemático do hardware.

Descrição da Biblioteca

Estrutura de Pastas

```
assets/  
    block_diagram.jpeg  
    electrical_diagram.jpeg  
    state_machine_v1.png  
    state_machine_v2.png  
mpu-code/  
    .devcontainer/  
    .vscode/  
    build/  
    components/  
        imu_tools/  
            include/  
            CMakeLists.txt  
            imu_tools.c  
        sensor_imu/  
            include/  
                sensor_imu.h
```

```
    CMakeLists.txt
    sensor_imu.c
main/
    CMakeLists.txt
    main.c
    diagram.json
    sdkconfig
    sdkconfig.ci
    sdkconfig.old
    wokwi.toml
README.md
```

Componente **imu_tools.h**

Esta biblioteca fornece funções para calcular as três variáveis dos ângulos de Euler e as quatro do quaternion a partir dos dados de um sensor IMU.

Estruturas

- **Quaternion**: Representa um quaternion.
- **EulerAngle**: Representa os ângulos de Euler.
- **IMUData**: Representa os dados do IMU 6DOF.

Funções

- `esp_err_t imu_read_data(IMUData *data)`: Obtém os dados do sensor IMU e armazena na estrutura **IMUData**. Retorna **ESP_OK** em caso de sucesso ou **ESP_FAIL** se ocorreu falha na inicialização.
- `esp_err_t imu_calculate_quaternion(const IMUData *data, Quaternion *quaternion)`: Calcula o quaternion com base nos dados do sensor IMU fornecidos e armazena o resultado na estrutura **Quaternion**. Retorna **ESP_OK** em caso de sucesso ou **ESP_FAIL** se ocorreu falha na inicialização.
- `esp_err_t imu_calculate_euler_angles(const Quaternion *quaternion, EulerAngle *euler)`: Calcula os ângulos de Euler a partir do quaternion fornecido e armazena o resultado na estrutura **EulerAngle**. Retorna **ESP_OK** em caso de sucesso ou **ESP_FAIL** se ocorreu falha na inicialização.

Componente **sensor_imu.h**

Esta camada de abstração é utilizada entre a biblioteca **imu_tools.h** e o sensor inercial, de onde serão obtidos os dados de aceleração e giroscópio.

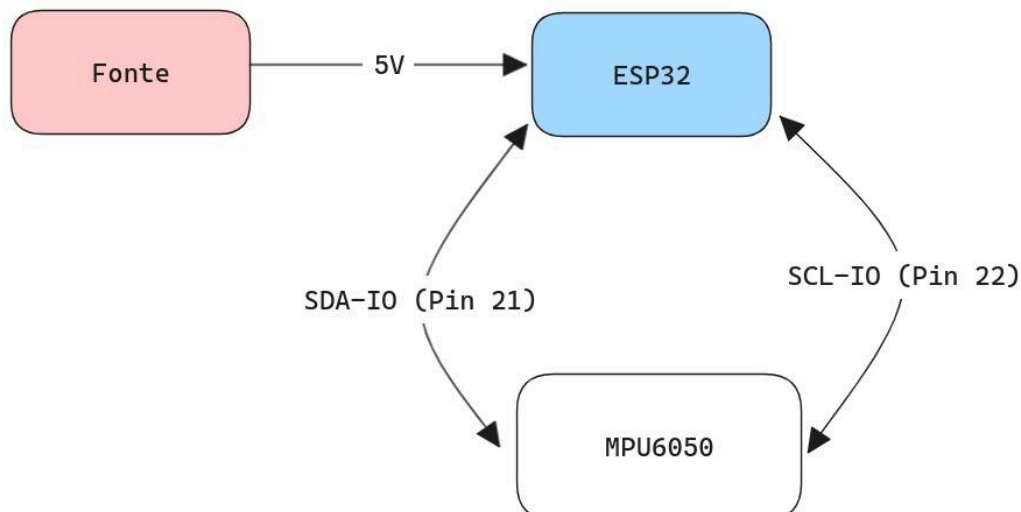
Estruturas

- `AccelerationData`: Representa os dados de aceleração.
- `GyroscopeData`: Representa os dados do giroscópio.

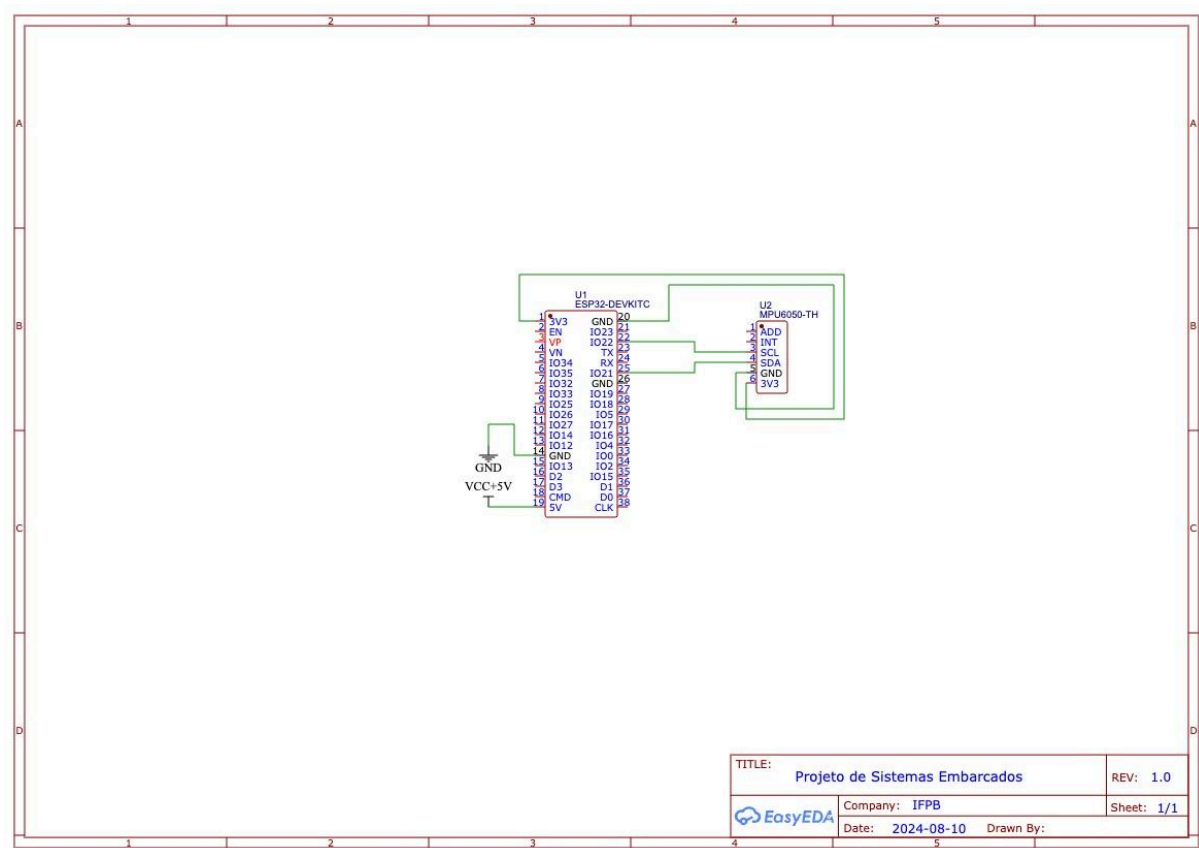
Funções

- `esp_err_t imu_init(uint8_t devAddr, gpio_num_t sda_pin, gpio_num_t scl_pin)`: Verifica se o sensor IMU está conectado e o inicializa passando o endereço I2C e as GPIOs utilizadas. Retorna `ESP_OK` em caso de sucesso ou `ESP_ERR_NOT_FOUND` se ocorreu falha na inicialização.
- `esp_err_t imu_get_acceleration_data(AccelerationData *data)`: Obtém os dados de aceleração do sensor inercial e armazena-os na estrutura `AccelerationData`. Retorna `ESP_OK` em caso de sucesso ou `ESP_FAIL` se ocorreu falha na inicialização.
- `esp_err_t imu_get_gyroscope_data(GyroscopeData *data)`: Obtém os dados do giroscópio do sensor inercial e armazena-os na estrutura `GyroscopeData`. Retorna `ESP_OK` em caso de sucesso ou `ESP_FAIL` se ocorreu falha na inicialização.
- `esp_err_t imu_deinit()`: Desabilita a comunicação I2C e libera os recursos. Retorna `ESP_OK` em caso de sucesso ou `ESP_FAIL` se ocorreu falha na inicialização.

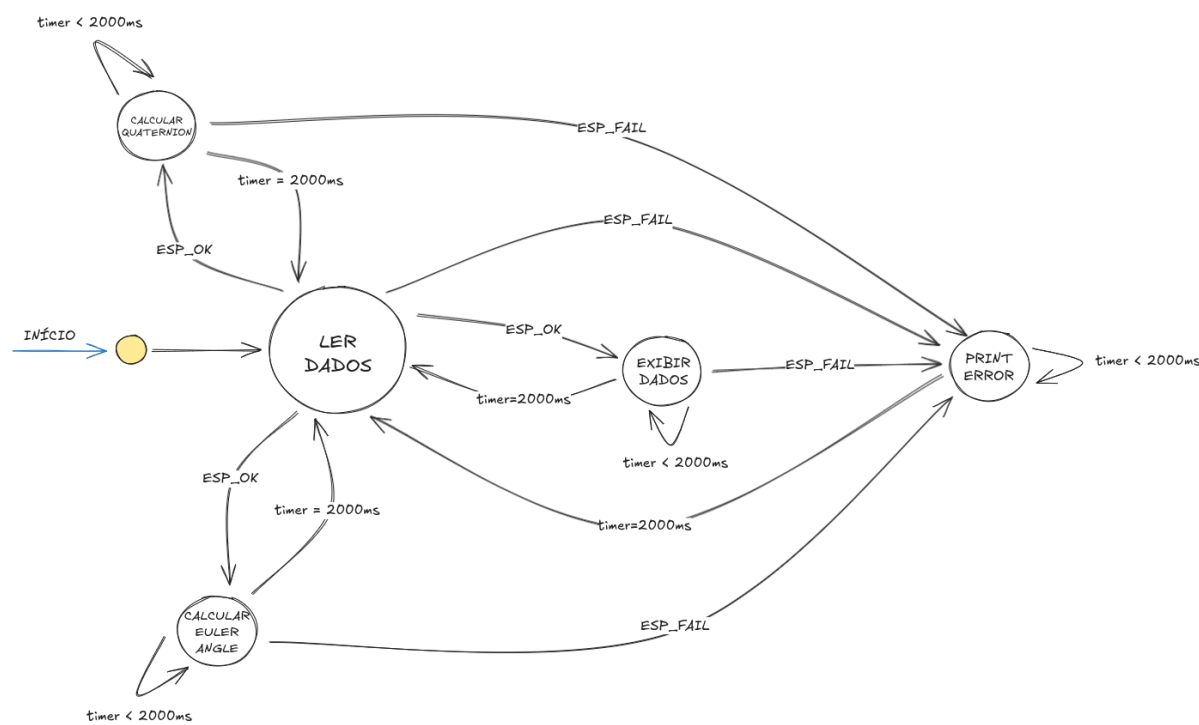
Diagrama de Bloco do Protótipo do Hardware



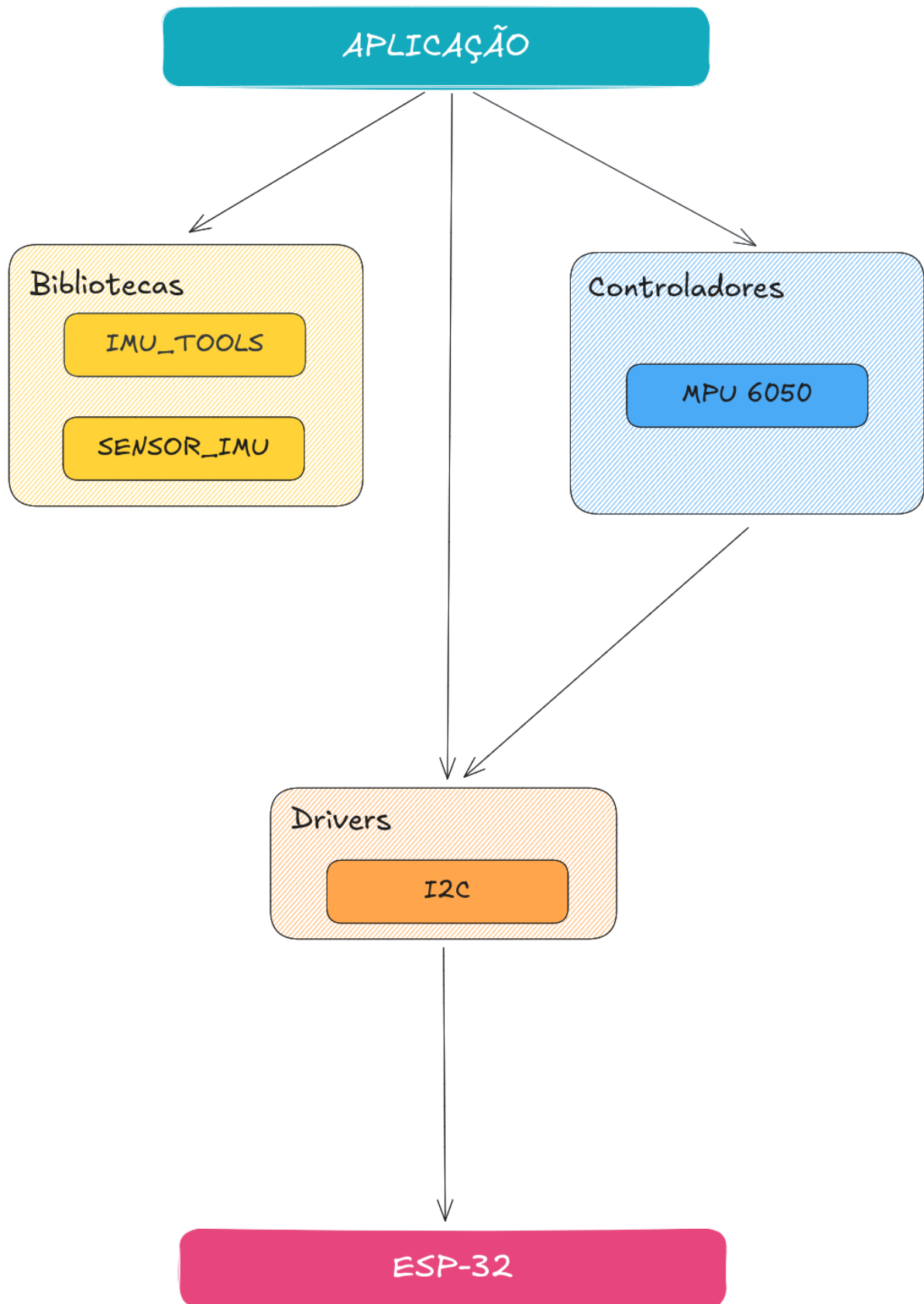
Esquemático do Hardware



Máquina de Estados



Arquitetura



Como Configurar o Wokwi no VSCode e Rodar a Simulação

Instalação da Extensão

1. **Instalar a Extensão Wokwi:** Primeiro, você precisa instalar a extensão "Wokwi for VS Code" no seu ambiente de desenvolvimento. Para fazer isso, abra o Visual Studio Code, vá até a aba de extensões (**Ctrl+Shift+X** ou **Command+Shift+P**), procure por "Wokwi" e instale a extensão oficial do Wokwi.

Solicitação de Licença

2. **Solicitar uma Nova Licença:** Depois de instalar a extensão, pressione **F1** para abrir a paleta de comandos e selecione "Wokwi: Request a new License". O VS Code irá pedir para confirmar a abertura do site do Wokwi no seu navegador. Confirme clicando em "Open". Em seguida, clique no botão que diz "GET YOUR LICENSE". Você pode ser solicitado a fazer login na sua conta do Wokwi. Se você ainda não tem uma, pode criar gratuitamente. O navegador solicitará confirmação para enviar a licença para o VS Code. Confirme novamente (você pode ter que confirmar duas vezes, uma vez no navegador e outra vez no VS Code). Você verá uma mensagem no VS Code que diz "License activated for [your name]".

Iniciar a Simulação

3. **Iniciar a Simulação com `diagram.json`:**
 - Navegue até a pasta do seu projeto no VSCode.
 - Procure pelo arquivo `diagram.json`.
 - Abra o arquivo `diagram.json`.
 - Com a simulação do seu projeto carregada, será possível iniciar a simulação.

Alternativa Online

Se, por algum motivo, a configuração local não funcionar, você ainda pode acessar o Wokwi online através do seguinte link:

[Wokwi - Projeto SE 2024.1](#)