

Sistema Embarcado para Monitoramento da Frequência de Tremores nas Mãos Humanas Utilizando um Joystick

Fellipe Patrick Lima de Brito
Área temática: Saúde.

RESUMO

A experiência do projeto foi altamente enriquecedora, permitindo a exploração dos conhecimentos adquiridos ao longo do curso. O objetivo central foi desenvolver um sistema de monitoramento de tremores das mãos, utilizando um joystick conectado a um microcontrolador, com visualização dos dados na plataforma ThingSpeak. A metodologia envolveu a coleta e análise da frequência dos tremores, possibilitando o acompanhamento da evolução da condição. Os resultados demonstraram a viabilidade da proposta, embora estudos adicionais sejam necessários para avaliar sua aplicabilidade clínica. O projeto oferece uma alternativa acessível e de baixo custo para o monitoramento, com potencial para expandir sua aplicação.

Palavras-chave: tremores, monitoramento, joystick, ThingSpeak, saúde.

INTRODUÇÃO

Os tremores acompanham os seres humanos há séculos, sendo mencionados desde o Antigo Egito e até mesmo em textos bíblicos, como nos Salmos, onde há referências a tremores corporais. Essas manifestações involuntárias podem ser classificadas em diferentes categorias, como tremor de repouso, fisiológico, essencial, parkinsoniano e de ação. O tremor de ação, por sua vez, pode ser subdividido em baixa, média ou alta frequência (Borges & Ferraz, 2006).

A análise da frequência do tremor de ação é essencial para o diagnóstico e monitoramento de condições neuromotoras, permitindo avaliar a progressão da doença, a eficácia de medicamentos ou os efeitos de tratamentos fisioterapêuticos (Ferraz et al., 1994). Nesse contexto, Paixão et al. desenvolveram uma luva instrumentada para registrar e quantificar os tremores do punho, proporcionando uma abordagem mais precisa na avaliação dessa condição.

No entanto, o estudo realizado por Paixão concentra-se nos tremores do punho, enquanto o presente trabalho propõe como objetivo o desenvolvimento de um sistema de monitoramento para avaliar os tremores da mão utilizando um joystick conectado a

um microcontrolador. Essa abordagem busca oferecer uma alternativa mais acessível e prática para o acompanhamento da condição, permitindo que profissionais da saúde, avaliem a evolução dos tremores ao longo do tempo. Os dados coletados serão analisados e visualizados por meio de gráficos na plataforma ThingSpeak, facilitando a interpretação dos resultados e auxiliando na tomada de decisões clínicas.

Dessa forma, este projeto visa facilitar o acesso ao monitoramento de tremores, reduzindo custos e proporcionando uma ferramenta eficaz para avaliar o impacto de intervenções terapêuticas, como o uso de medicamentos ou fisioterapia, na redução dos sintomas.

METODOLOGIA

Este projeto utiliza a placa BitDogLab como sua principal unidade, incorporando vários componentes eletrônicos para detectar e avaliar tremores nas mãos por meio de um joystick.

Componentes Utilizados

- **Raspberry pi pico w** – Microcontrolador principal para processamento e comunicação de dados.
- **Display OLED** – Utilizado para exibir informações sobre os tremores capturados, informações sobre a conexão wi-fi e manda mensagens durante o monitoramento para o paciente ficar mais calmo.
- **Matriz de LEDs** – Implementada para ser um estímulo visual para o usuário durante o exame, através de uma seta indicando para que lado o joystick deve apontar.
- **Joystick** – Sensor principal para detecção dos tremores da mão, registrando os deslocamentos, durante 60 segundos que o paciente fica com a sua mão no joystick em uma posição.
- **LED RGB** – Led para estímulo, para o paciente e profissional que estiverem utilizando reconhecerem por meio da cor do led, qual é o grau de tremor.
- **Botão** – Utilizado para iniciar ou interromper o fluxo do monitoramento, seja para iniciar o monitoramento ou finalizar a qualquer momento.
- **Wi-Fi (HTTP)** – Comunicação sem fio para envio dos dados coletados à plataforma **ThingSpeak** para análise e visualização. A escolha do protocolo HTTP ao invés do MQTT é pela simplicidade de implementação e compatibilidade com

a plataforma **ThingSpeak**, que já oferece suporte nativo para requisições HTTP via API REST. Além disso, o protocolo HTTP permite uma comunicação direta e eficiente sem a necessidade de um broker MQTT, reduzindo a complexidade do sistema e tornando a integração mais acessível.

Desenvolvimento e Abordagem

O projeto segue uma abordagem estruturada em três etapas principais:

1. **Integração dos componentes** – Conexão do joystick, display OLED, matriz de LEDs, LED RGB e botões à placa BitDogLab, garantindo a comunicação entre os módulos.
2. **Validação e testes** – Realização de testes para aferir a precisão das leituras do joystick, a estabilidade da transmissão de dados e a eficácia da visualização na plataforma.

Coleta e Análise de Dados

Os dados capturados pelo joystick representam as oscilações da mão do usuário. Esses valores são registrados e enviados para a **ThingSpeak**, onde são processados e exibidos em gráficos para facilitar a interpretação. Essa abordagem permite avaliar a evolução dos tremores ao longo do tempo e identificar possíveis melhorias a partir de tratamentos como fisioterapia ou uso de medicamentos. No fluxograma (figura 1) é possível ver como o sistema se comporta.

Diagrama de Fluxo

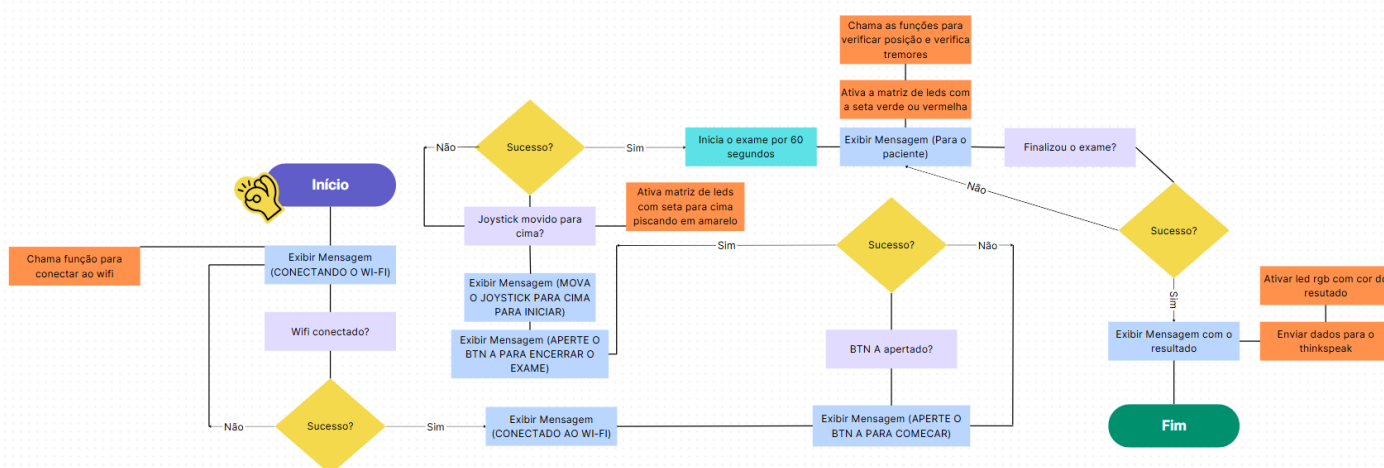


Figura 1 Fluxograma do sistema

DESENVOLVIMENTO

A etapa de desenvolvimento vai explicar como o sistema funciona e está arquitetado.

Integração dos componentes

1. Controle do Fluxo pelo Botão (BTN A)

- O botão **BTN A** atua como um seletor de estado. Ele pode alternar entre dois modos principais:
 - **Modo Não Foi Pressionado (Modo início):** Estado inicial do sistema, aguardando interação.
 - **Modo Foi Pressionado (Modo exame):** Inicia a análise dos movimentos do **joystick**.
- O botão funciona via **interrupções**, ou seja, ao ser pressionado, ele gera um evento que altera o estado do programa.

2. Exibição de Mensagens no Display OLED

- O **display OLED** será responsável por indicar qual o modo atual do sistema.
- Quando o **BTN A** é pressionado:
 - Se o sistema estava no **Modo Início**, o display pode exibir: "Mova o joystick para cima para iniciar o exame".
 - Se já estava no **Modo Exame**, ele exibe mensagens para ajudar o paciente.
 - Ao final do exame ele mostra o nível de tremor

3. Controle da Matriz de LEDs pelo Joystick

- O **joystick** possui sensores para capturar deslocamento nos eixos **X e Y**, além do botão de clique.
- A **matriz de LEDs** responde ao estado do joystick:
 - **Movimentos para cima, baixo, esquerda, direita** → Definem a cor da seta para cima, se o movimento para cima led verde, se não vermelho.

4. Monitoramento de Tremores no Joystick

- O **joystick** também pode ser utilizado para detectar tremores ou instabilidades na mão do usuário, durante o período de 60 segundos.
- Essa captação pode ser feita através de variações no eixo **X e Y**.
- Se for detectado um nível significativo de tremor, isso pode influenciar o resultado.

5. LED RGB e Display OLED Finalizam o Exame

- Após coletar os dados do **joystick**, o sistema exibe o resultado de acordo com a detecção de tremores.
- O **LED RGB** indica visualmente o resultado:
 - **Apagado** → Nenhum tremor detectado (normal).
 - **Verde** → Pequena instabilidade detectada.
 - **Amarelo** → Média instabilidade detectada.
 - **Vermelho** → Tremores significativos detectados.
- O **display OLED** também mostra a análise, por exemplo:
 - "Nível de tremor - Baixo".
 - "Nível de tremor - Médio".
 - "Nível de tremor - Alto".

Funções impactantes

```
void verificar_tremor(uint adc_x_raw, uint adc_y_raw, uint *last_x, uint *last_y) {  
    if (has_changed(adc_x_raw, *last_x) || has_changed(adc_y_raw, *last_y)) {  
        if (is_shaking(adc_x_raw, *last_x) || is_shaking(adc_y_raw, *last_y)) {  
            tremor_count++;  
        }  
        *last_x = adc_x_raw;  
        *last_y = adc_y_raw;  
    }  
}
```

```
}
```

```
int determinaNivelTremor(float tremoresPorSegundo) {  
  
    // o calculo é bem simples, ja que se trata de um prototipo inicial, se ele tiver 1  
    tremor por segundo é baixo, 2 é medio, maior que 5 é alto  
  
    if (tremoresPorSegundo > 5.0) {  
  
        return 3; // Alto  
  
    } else if (tremoresPorSegundo > 2.0) {  
  
        return 2; // Médio  
  
    } else if (tremoresPorSegundo > 1.0) {  
  
        return 1;  
  
    } else  
  
        return 0;  
  
}
```

As funções mencionadas desempenham um papel crucial na detecção de tremores. A primeira função monitora variações para identificar um possível tremor e, caso ocorra, incrementa a variável `tremor_count`. Essas variações são determinadas com base em variáveis limiares, que armazenam limites definidos manualmente pelo desenvolvedor. Se o movimento do joystick ultrapassar esses limites, as variáveis retornam um valor booleano, permitindo que o sistema prossiga com a análise. Se o movimento do

joystick ultrapassar esses limites, as variáveis retornam um valor booleano, permitindo que o sistema prossiga com a análise.

O segundo método, `determinaNivelTremor`, é mais direto e se baseia, em parte, no estudo de Borges & Ferraz (2006). Ele verifica frequências específicas associadas ao tremor essencial para classificá-lo com maior precisão.

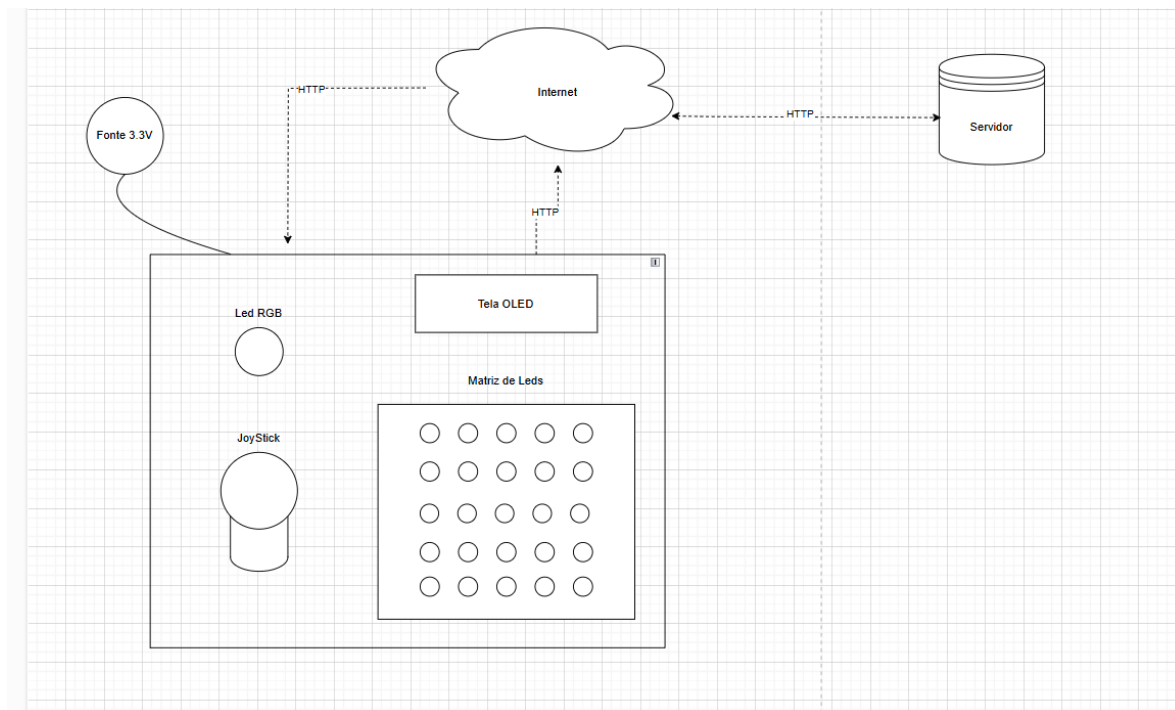


Figura 2 Diagrama do sistema

O diagrama acima tenta mostrar como os componentes devem estar empregados, vale salientar que o raspberry pi pico w está aí nesse sistema internamente controlando tudo. O protocolo HTTP foi escolhido pela sua implementação mais simples e compatibilidade com o serviço de nuvem adotado o `thinkspeak`. Então por meio da internet, com a placa de rede do sistema, é capaz de enviar os dados até a nuvem (servidor do `thinkspeak`). O Sistema é alimentado com 3.3V para seu funcionamento.

Arquitetura do Projeto e Como Utilizar

A arquitetura do projeto foi organizada de forma modular para facilitar o desenvolvimento e a manutenção. O código-fonte está estruturado em duas pastas principais:

- **Controller:** Contém os serviços responsáveis pelo funcionamento do sistema, incluindo a lógica de controle e interação com os periféricos.

- **View:** Contém apenas o arquivo `main.c`, que serve como ponto de entrada do programa e gerencia a execução principal.

O projeto é um sistema embarcado que integra-se à API do **ThingSpeak** para envio de dados. Para garantir um funcionamento eficiente, utilizamos diversas bibliotecas específicas do Raspberry Pi Pico, adicionadas ao projeto por meio da diretiva `target_link_libraries(main ...)`, incluindo:

- **hardware_spi:** Comunicação SPI com periféricos.
- **hardware_i2c:** Comunicação I2C para sensores e dispositivos externos.
- **pico_stdlib:** Biblioteca padrão do Raspberry Pi Pico.
- **hardware_adc:** Leitura de sinais analógicos via conversor AD.
- **hardware_pio:** Uso de periféricos programáveis PIO.
- **hardware_pwm:** Controle de sinais PWM para acionamento de dispositivos.
- **hardware_clocks:** Gerenciamento de clocks internos.
- **pico_cyw43_arch_lwip_threadsafe_background:** Suporte à conectividade Wi-Fi com o módulo CYW43.
- **pico_lwip_mbedtls** e **pico_mbedtls:** Implementação de protocolos de rede seguros.

Essa organização garante um sistema eficiente e modular, facilitando futuras expansões e manutenção.

A utilização começa na conexão do wifi que é feita nas linhas 15 e 17, nelas é importante fornecer uma rede, com acesso à internet, pelo hostname e senha wifi, para que o sistema funcione. Caso não seja modificado, o sistema não irá iniciar. O exemplo abaixo mostra como fazer isso.

Linha 15 arquivo (`main.c`) - `char hostName[] = "internet";`

Linha 17 arquivo (`main.c`) - `char senhaWifi[] = "12345678";`

Altere para as suas credenciais, feito isso é só passar o projeto para sua placa bitdoglab e aproveitar, você pode ver suas estatísticas da frequência de tremores por meio do link [SIMOTE - ThingSpeak IoT](#).

Validação e testes

Os testes foram realizados para garantir a **precisão das leituras do joystick**, a **estabilidade da transmissão de dados** e a **eficácia da visualização na plataforma ThingSpeak**.

1. **Precisão do Joystick:** O sensor foi calibrado e testado com movimentos controlados para verificar a consistência das leituras e a sensibilidade às variações de tremor.
2. **Transmissão de Dados via Wi-Fi:** Foram avaliadas a conectividade da placa BitDogLab com o acesso de envio para o thinkspeak.

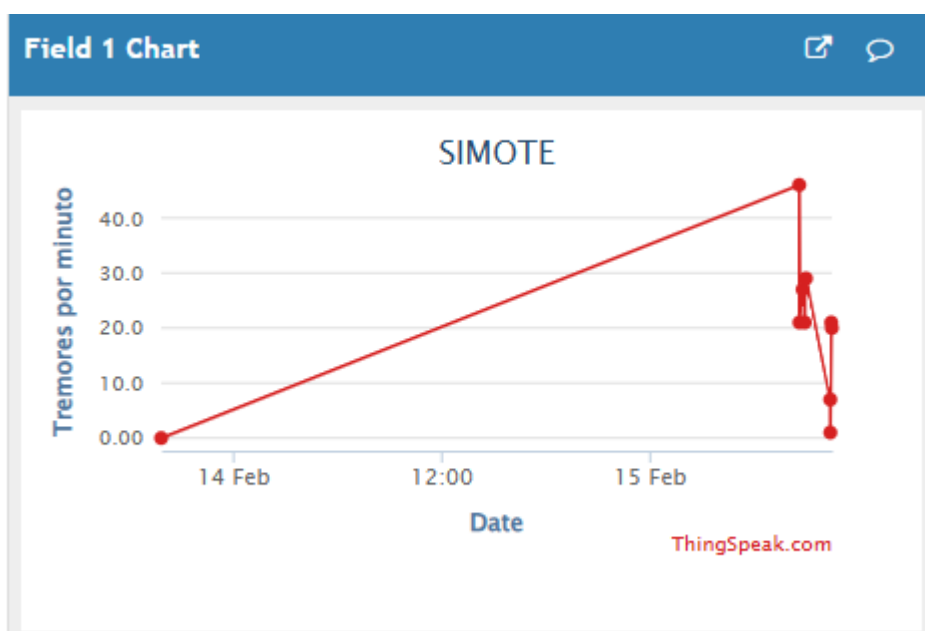


Figura 3 Gráfico do ThingSpeak

O gráfico (figura 3) valida os testes de envio da taxa de tremores por minuto dos pacientes possibilitando a visualização de forma fácil e eficiente da frequência e a análise ao decorrer do tempo. É possível monitorar esse gráfico por esse link pulico [SIMOTE - ThingSpeak IoT](#).

5 CONCLUSÃO

O objetivo principal do projeto foi alcançado, demonstrando a viabilidade de medir a frequência dos tremores das mãos por meio do joystick e monitorá-la via plataforma ThingSpeak. No entanto, estudos mais aprofundados são necessários para

avaliar sua real utilidade clínica. A participação de especialistas na área da saúde é essencial para validar essa abordagem, visto que a análise profissional leva em consideração diversos fatores além da frequência do tremor, como o estado emocional do paciente e fatores hereditários (Ferraz et al., 1994).

Apesar das limitações, este protótipo representa um avanço na tentativa de tornar o monitoramento dos tremores mais acessível e econômico. Futuras versões do projeto podem explorar formas de capturar tremores em outras regiões do corpo, ampliando sua aplicação para além das mãos.

O sistema desenvolvido já pode complementar exames na área da saúde, oferecendo um meio simples e de baixo custo para monitoramento. Com mais tempo e recursos, melhorias na precisão da leitura dos tremores e na análise dos dados podem tornar essa solução ainda mais útil no auxílio ao diagnóstico e acompanhamento de pacientes.

REFERÊNCIAS

BORGES, V.; FERRAZ, H. B. Tremors. *Revista Neurociências*, v. 14, n. 1, p. 43-47, 2006. Disponível em: <https://www.academia.edu/download/99792438/6320.pdf>. Acesso em: 15 fev. 2025.

FERRAZ, H. B.; ANDRADE, L. A. F. D.; AZEVEDO SILVA, S.; BORGES, V.; ROCHA, M. S. G. Tremor postural e distonia: aspectos clínicos e considerações fisiopatológicas. *Arquivos de Neuro-Psiquiatria*, v. 52, p. 466-470, 1994. Disponível em: <https://www.scielo.br/j/anp/a/Q4gzrNy9yzMg7rzwxnK8pZx/>. Acesso em: 15 fev. 2025.

PAIXÃO, A. P. S. et al. Luva instrumentada para registro e quantificação do tremor de punho. 2014. Disponível em: https://www.canal6.com.br/cbeb/2014/artigos/cbeb2014_submission_136.pdf. Acesso em: 15 fev. 2025.