

Universidade Tecnológica Federal do Paraná
Campus Apucarana
Engenharia de Computação

Projeto Final: Cofre Interativo com FreeRTOS

Sistemas Embarcados

Alunos: Angélica B. G. Luciano RA: 2047063
Professor: Marcelo de Oliveira

Julho de 2025

Sumário

1	Introdução	2
2	Desenvolvimento do Projeto	2
2.1	Componentes de Hardware	2
2.2	Tarefas (Tasks)	2
2.3	Mecanismos de Comunicação	3
2.4	Os enigmas de Durin	4
2.5	Montagem do Protótipo	5
3	Resultados	6
4	Conclusão	6
	Referências	6

1 Introdução

O desenvolvimento de sistemas embarcados modernos enfrenta o desafio crescente de gerenciar múltiplas operações concorrentes de forma eficiente e previsível. A utilização de um *loop* sequencial monolítico torna-se inadequada à medida que a complexidade do sistema aumenta, levando a problemas de responsividade e manutenção. Neste contexto, os Sistemas Operacionais de Tempo Real (RTOS) apresentam-se como uma solução robusta, fornecendo mecanismos para escalonamento de tarefas, comunicação interprocessos e sincronização. Este relatório detalha a implementação de um sistema embarcado interativo, concebido como um protótipo de cofre com múltiplos desafios, utilizando a plataforma Arduino Mega 2560 e o escalonador **FreeRTOS**.

O sistema desafia o “viajante” a decifrar uma sequência de quatro enigmas, cada um uma homenagem a um elemento da rica literatura de J.R.R. Tolkien. O objetivo é demonstrar como a arquitetura multitarefa pode gerenciar processos complexos de forma elegante e robusta, criando uma experiência funcional e encantadora.

2 Desenvolvimento do Projeto

O projeto consiste em um cofre eletrônico que permanece selado até que quatro desafios sejam superados em sequência. A lógica é governada por uma máquina de estados, e a interação com o usuário é guiada pelos periféricos: *display* OLED, teclado matricial, sensor de vibração, *buzzer*, LEDs, LDR e um potenciômetro. A arquitetura do FreeRTOS foi fundamental para organizar o sistema como uma “sociedade”, onde cada tarefa tem um papel vital e se comunica com as outras para alcançar o objetivo comum.

2.1 Componentes de Hardware

- Microcontrolador: Arduino Mega 2560
- *Display*: Display OLED I2C 128x64 (Driver SSD1306)
- Sensores: Sensor de Vibração SW-420, Resistor Dependente de Luz (LDR), Potenciômetro e Teclado Matricial 4x3.
- Atuadores: Servo Motor e *Buzzer*.
- *Feedback* Visual: 5 LEDs (um para status geral e quatro para indicar o progresso dos enigmas).

2.2 Tarefas (Tasks)

A arquitetura do software foi dividida em quatro tarefas principais, cada uma nomeada em homenagem a um elemento da Terra-média, refletindo sua função no sistema.

- `Task_ConselhoDeElrond` (Tarefa de Controle): Assim como o Conselho de Elrond em Valfenda tomava as decisões cruciais que guiavam o destino da Terra-média, esta tarefa é o cérebro do sistema. Com a mais alta prioridade, ela executa a máquina de estados, interpreta os eventos enviados pelos sensores e decide o rumo da jornada, despachando ordens para as outras tarefas.

- Task_Palantir (Tarefa de Monitoramento de Sensores): As Pedras Videntes, ou Palantíri, permitiam ver eventos em locais distantes. De forma análoga, esta tarefa de prioridade intermediária “vê” o mundo físico. Ela monitora continuamente os sensores (vibração, luz, teclado, potenciômetro) e, ao detectar um evento, relata-o fielmente ao “Conselho”, sem interpretar seu significado.
- Task_EspelhoDeGaladriel (Tarefa de Interface Gráfica): O Espelho de Galadriel mostrava visões do passado, presente e futuro. Esta tarefa de baixa prioridade controla o *display* OLED, nosso espelho. Ela recebe comandos do “Conselho” e traduz essas ordens em imagens e textos, guiando o usuário através dos enigmas.
- Task_VozDeSaruman (Tarefa de Atuação): A voz de Saruman era uma ferramenta poderosa de comando e influência. Esta tarefa, também de baixa prioridade, comanda os atuadores. Ela materializa as decisões do “Conselho” em ações físicas, seja através de sons de vitória ou falha (*Buzzer*) ou movendo a própria trava do cofre (*Servo Motor*).

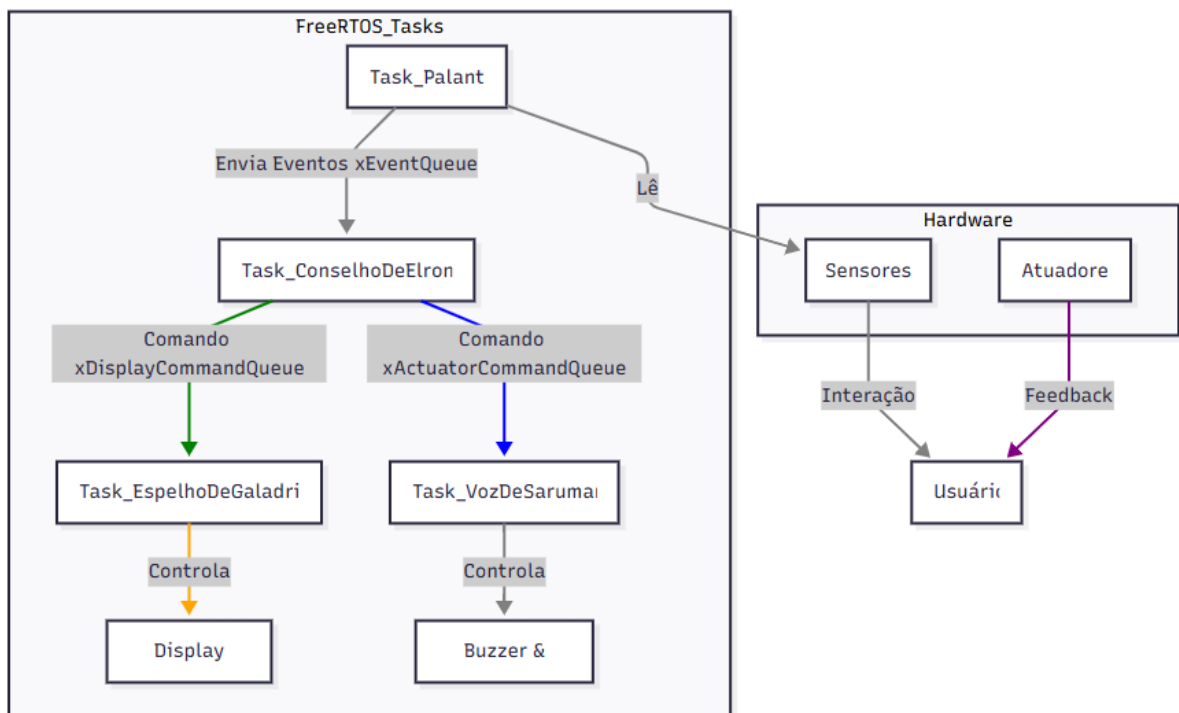


Figura 1: Diagrama de Interação das Tarefas.

2.3 Mecanismos de Comunicação

Em um sistema multitarefa, é crucial que as tarefas possam se comunicar e sincronizar de forma segura. O FreeRTOS oferece vários mecanismos para isso, sendo os mais comuns as filas e os semáforos.

- Filas (Queues): São estruturas de dados do tipo FIFO (First-In, First-Out) projetadas para passar mensagens (cópias de dados) entre tarefas de forma segura (thread-safe). Uma tarefa pode escrever em uma fila e outra pode ler dela, mesmo

que executem em momentos diferentes. Neste projeto, as filas foram o mecanismo de comunicação principal, pois era necessário não apenas sinalizar um evento, mas também enviar os dados associados a ele. Foram utilizadas três filas:

- xEventQueue: Transporta dados dos sensores da Task_Palantir para a Task_ConselhoDeElrond.
- xDisplayCommandQueue: Envia comandos de renderização da Task_ConselhoDeElrond para a Task_EspelhoDeGaladriel.
- xActuatorCommandQueue: Despacha comandos de ação da Task_ConselhoDeElrond para a Task_VozDeSaruman.

Neste projeto, as filas foram a escolha mais adequada porque a comunicação envolvia a transferência de dados estruturados (eventos dos sensores e comandos para os atuadores). Um semáforo poderia apenas notificar que “algo aconteceu”, mas não “o que aconteceu”. As filas, ao contrário, permitem enviar a informação completa, desacoplando completamente a lógica de controle das tarefas de I/O.

2.4 Os enigmas de Durin

Cada etapa para abrir o cofre é um enigma temático. Se o usuário falhar ou o tempo se esgotar, o cofre emite um som de falha e retorna ao seu estado adormecido.

1. O Despertar da Pedra (Vibração): O primeiro desafio é despertar o cofre. Assim como as Portas de Durin, que precisavam de uma batida, o sistema é ativado por uma vibração detectada pelo sensor, simulando uma batida na porta de pedra.
2. A Luz de Eärendil (LDR): No covil de Laracna, a luz do Frasco de Galadriel foi a única esperança de Frodo. Neste enigma, o usuário deve “brandir” uma luz contra a escuridão. O sensor LDR precisa ser iluminado intensamente, simbolizando o afastamento das sombras.
3. O Portão de Moria (Teclado): A palavra “Amigo” em élfico abriu os portões de Moria. Aqui, o usuário deve digitar a senha secreta (“**635566**”, que corresponde a “Mellon” em um teclado T9) no teclado matricial. O *display* mostra o progresso, e apenas a senha correta permite avançar.
4. A Balança de Durin (Potenciômetro): A perícia e a paciência dos Anões eram lendárias. O último enigma exige equilíbrio e precisão. O usuário deve girar um potenciômetro para alinhar um marcador dentro de uma pequena zona-alvo no *display* e mantê-lo lá por três segundos. É um teste de firmeza, digno de um mestre artesão anão.

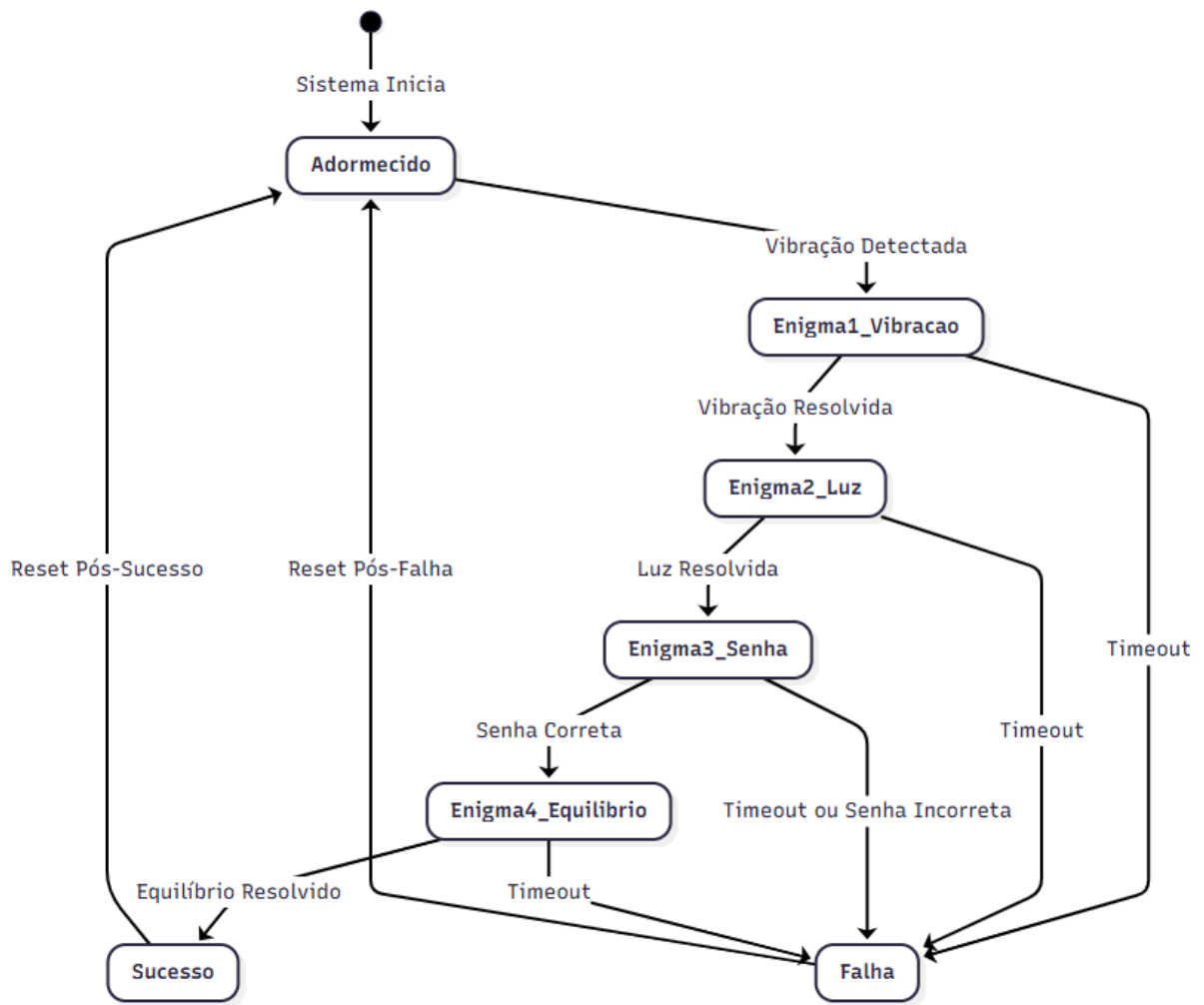


Figura 2: Fluxo da Máquina de Estados.

2.5 Montagem do Protótipo

A implementação física do projeto envolveu a conexão dos sensores e atuadores à placa Arduino Mega, conforme o esquemático de pinos definido no código-fonte. A Figura 3 ilustra a montagem final do circuito em protoboard.

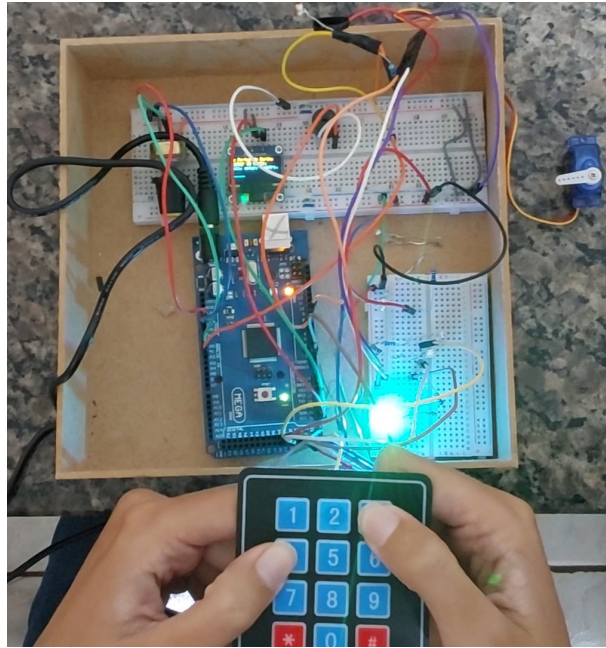


Figura 3: Montagem do circuito em protoboard.

3 Resultados

O sistema demonstrou-se funcional e robusto, com as tarefas executando concorrentemente sem conflitos de acesso ou perda de dados. A máquina de estados respondeu corretamente a todas as entradas do usuário, transições de estado, condições de sucesso e falha (incluindo timeouts). O funcionamento completo do protótipo, demonstrando a sequência de enigmas e a resposta dos atuadores, pode ser visualizado no [vídeo de demonstração](#).

4 Conclusão

A construção do protótipo permitiu validar de forma eficaz a aplicação dos conceitos de sistemas operacionais de tempo real em um projeto prático. A arquitetura multitarefa com FreeRTOS provou ser indispensável para gerenciar a complexidade do sistema, resultando em um código modular, resiliente e de fácil manutenção. A utilização de filas como mecanismo de comunicação foi crucial para desacoplar as responsabilidades e garantir a responsividade do sistema. Conclui-se que a adoção de um RTOS é uma prática fundamental para o desenvolvimento de sistemas embarcados de complexidade moderada a alta. O código-fonte completo do projeto, para fins de auditoria e replicação, está disponível no repositório listado na seção de Referências.

Referências

ADAFRUIT. *Adafruit GFX Library*. GitHub. Disponível em: <<https://github.com/adafruit/Adafruit-GFX-Library>>. Acesso em: 2 de julho de 2025.

ADAFRUIT. *Adafruit SSD1306 Library*. GitHub. Disponível em: <https://github.com/adafruit/Adafruit_SSD1306>. Acesso em: 2 de julh de 2025.

FEILIPU. *Arduino_FreeRTOS_Library*. GitHub. Disponível em: <https://github.com/feilipu/Arduino_FreeRTOS_Library>. Acesso em: 3 de julho de 2025.

FEILIPU. *Arduino FreeRTOS (Blog Post)*. Disponível em: <https://feilipu.me/2015/11/24/arduino_freertos/>. Acesso em: 3 de julho de 2025.

Luciano, Angélica B. G. Código-fonte do projeto Cofre com Enigmas. Repositório de Software, 2025. Disponível em: <https://github.com/AngelicaBGLuciano/Cofre_com_enigmas>.

Tolkien, J.R.R. O Senhor dos Anéis: A Sociedade do Anel. 1954.