Resumo do Projeto: Medição de Tempo em Tarefas com FreeRTOS no Raspberry Pi Pico

# 1) Objetivos

Este projeto demonstra a criação de múltiplas tarefas com FreeRTOS em um sistema embarcado com o Raspberry Pi Pico, com foco na medição do tempo de execução real de cada tarefa.  
  
O objetivo principal é mostrar como tarefas concorrentes podem ser temporizadas com precisão em microssegundos, utilizando a função `time\_us\_64()`.

# 2) Implementação e funcionalidades

Três tarefas são criadas, cada uma com comportamento e prioridade diferentes:  
  
 tarefa\_led (Prioridade 5): Alterna o estado de um LED no GPIO 13 a cada 250ms, e mede o tempo gasto para isso.  
 tarefa\_usb (Prioridade 4): Envia uma mensagem no terminal USB a cada 500ms e mede o tempo de execução de cada envio.  
 tarefa\_monitoramento (Prioridade 3): Lê os valores dos canais ADC0 (GPIO26) e ADC1 (GPIO27), converte para tensão e imprime no terminal. Mede também o tempo total da operação.  
  
A estrutura do código permite observar como o FreeRTOS trata prioridades e como medir o uso de CPU por tarefa.

# 3) Onde visualizar os resultados

Conectando o Raspberry Pi Pico ao computador via USB, e abrindo um terminal serial (como PuTTY ou VSCode Serial Monitor), o usuário verá mensagens como:  
  
Tarefa 1 (LED): LED ON | Tempo: 3 us  
Tarefa 2: Sistema operando normalmente. | Tempo: 5 us  
Tarefa 3: Tensões -> X: 1.734 V | Y: 2.012 V | Tempo: 94 us  
  
Cada linha traz uma medição real de tempo de CPU em microssegundos para aquela execução da tarefa.

# 4) Priorização e escalonamento

As tarefas são executadas com prioridades distintas (5, 4 e 3):  
  
- O FreeRTOS garante que tarefas com maior prioridade são executadas primeiro.  
- Como todas as tarefas usam `vTaskDelay()`, elas liberam o processador após executar, permitindo um agendamento cooperativo.

# 5) Aplicações educacionais e práticas

Este projeto serve como base para:  
  
- Treinamento de alunos em sistemas operacionais de tempo real.  
- Análise de desempenho de tarefas.  
- Projetos embarcados que necessitam conhecer o uso da CPU por tarefa.  
- Diagnóstico de consumo de tempo em sistemas multitarefa.