



Sistemas Operacionais de Tempo Real: Escalonamento

Unidade 3 | Capítulo 2

Instrutores:

Prof. Antônio Gabriel Sousa Borralho

Prof. Diego Dutra Sampaio

Prof. Ernesto de Lucena Chagas

Prof. Francisco Borges Carreiro



Coordenação:

Cláudio Leão Torresjorge

Renato Santos da Silva

Waldenison Novaes Carneiro

Washington Luis Santos Silva

Executores:



Coordenação:

Plau



Rio Grande do Norte

Iniciativa:

Maranhão



UNIÃO E RECONSTRUÇÃO

Unidade 3

Aplicação

Multitarefa

Capítulo 2

Sistemas Operacionais de Tempo Real: Escalonamento

Objetivo: Desenvolver uma aplicação embarcada na placa BitDogLab, utilizando FreeRTOS e Pico-SDK, com o objetivo de testar e demonstrar, de forma prática e didática, o funcionamento de multitarefas no controle de periféricos como LEDs, botões, joystick, microfone e buzzer, além de aplicar conceitos de leitura analógica, temporização e resposta a eventos em tempo real.

Enunciado:

Aplicação Didática com FreeRTOS na BitDogLab: Multitarefa, Monitoramento e Autoteste de Periféricos

Desenvolva uma aplicação embarcada utilizando a placa BitDogLab, o FreeRTOS e a Pico-SDK, estruturada em três tarefas concorrentes, com foco em testes de hardware, monitoramento de sensores e sinalização visual/sonora. A aplicação deve ser iniciada automaticamente ao ligar a placa e funcionar conforme a descrição abaixo:

Tarefa 1: Self-Test

- Testar LEDs RGB: acender e apagar sequencialmente.
- Testar Buzzer: gerar som simples.
- Testar botões A, B e joystick SW.
- Testar Joystick analógico (ADC0 e ADC1).
- Testar Microfone (ADC2).
- Imprimir resultados na porta USB com pequenas pausas para visualização.
- Deletar a si própria após concluir (vTaskDelete(NULL)).

Tarefa 2: Alive Task

- Piscar o LED vermelho (GPIO 13) indicando o funcionamento.
- Ciclo de 1000ms (500ms ligado e 500ms desligado).

Tarefa 3: Monitor de Joystick e Alarme

- Ler e imprimir as tensões dos eixos X (ADC1) e Y (ADC0) do joystick.
- Imprimir leituras a cada 50ms na porta USB.
- Se qualquer eixo exceder 3.00V, ativar o buzzer (GPIO 21).
- Desligar o buzzer quando os valores voltarem ao normal.

Instruções:

Após a inicialização da aplicação, ficam em funcionamento somente a Tarefa 2 e Tarefa 3, pois a Tarefa 1 se auto deleta após a inicialização. Todavia se a placa for reinicializada todo o processo deve se repetir novamente.

Periférico	Função	GPIO / ADC
LED Vermelho (Alive)	Piscar Alive	13
LED RGB Verde	Self-Test	11
LED RGB Azul	Self-Test	12
Buzzer	Alarme e Self-Test	21
Botão A	Self-Test	5
Botão B	Self-Test	6
Joystick SW	Self-Test	22
Joystick VRy	Leitura Analógica	ADC0 (26)
Joystick VRx	Leitura Analógica	ADC1 (27)
Microfone	Leitura Analógica	ADC2 (28)

Materiais e Conceitos Envolvidos:

- **Placa BitDogLab**

- Microcontroladora baseada no chip RP2040, compatível com o Pico SDK e adequada para aplicações em tempo real utilizando o sistema operacional FreeRTOS.

- **Computador com ambiente de desenvolvimento configurado, contendo:**

- Visual Studio Code (VSCode)
- Pico SDK (Software Development Kit oficial para o RP2040)
- FreeRTOS integrado ao projeto

Instruções de Implementação:

1. Faça o projeto no VSCode e execute na placa BitdogLab.
2. Adicione pequenos delays entre os testes no Self-Test para garantir tempo de leitura via USB.
3. Utilize a função `sleep_ms()` antes de iniciar o scheduler para garantir estabilização da USB CDC.

Prazo: 5 dias

Feedback Automático: Será corrigido pelo professor/Monitor.

Forma de entrega: O aluno deverá enviar um arquivo em formato

pdf com nome completo, número de matrícula com enunciado da questão e código em C/VS-code da solução da questão.

