embarca tech | Residência Tecnológica em Sistemas Embarcados

Aplicação Multitarefa

Unidade 1 | Capítulo 2

Elias Teodoro S Jr

Executores:
INSTITUTO FEDERAL Piauí | INSTITUTO FEDERAL Rio Grande do Norte | INSTITUTO FEDERAL Maranhão | INSTITUTO FEDERAL Ceará | cepedi | INSTITUTO HARDWARE BR

Coordenação: Softex

Iniciativa: MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO | GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

Unidade 1

Aplicação Multitarefa

Capítulo 2

Objetivo: Realizar práticas utilizando multitarefa em arquitetura microcontrolada.

Enunciado:

Sistema de Monitoramento Simples com 3 Tarefas

Criar uma aplicação embarcada no FreeRTOS com 3 tarefas que simulam o monitoramento de um sistema com sensores (como um botão e um LED). As tarefas irão cooperar para realizar diferentes funções, como ler o estado do botão e controlar o LED.

Instruções:

Descrição do Sistema:

1. Tarefa 1 (Leitura do Botão):

- Essa tarefa será responsável por simular a leitura de um botão. Ela será executada periodicamente, a cada 100ms, e enviará o estado do botão para a próxima tarefa.

2. Tarefa 2 (Processamento do Botão):

- Receberá o estado do botão da Tarefa 1. Caso o botão seja pressionado (simulado com a variável), ela acionará a próxima tarefa (a de controlar o LED). Caso contrário, apenas aguardará o próximo ciclo de leitura.

3. Tarefa 3 (Controle do LED):

- Controlará um LED (simulado como uma variável ou saída digital). Se o botão for pressionado, o LED será aceso, caso contrário, será apagado.

A tarefa será executada sempre que for acionada pela Tarefa 2.

Detalhamento da Implementação:

1. Definições de variáveis:

- Defina variáveis para armazenar o estado do botão e do LED.
- Utilize filas ou variáveis globais para compartilhar o estado entre as tarefas.

2. Criação das tarefas no FreeRTOS:

- Tarefa 1: Leitura do botão (criada com uma prioridade baixa, executada a cada 100ms).
- Tarefa 2: Processamento e decisão (executada dependendo do estado do botão).
- Tarefa 3: Controle do LED (executada apenas quando acionada pela Tarefa 2).

Sincronização entre as tarefas:

Pode-se usar semáforos ou filas para que as tarefas se comuniquem e cooperem entre si.

A Tarefa 2 pode enviar um sinal para a Tarefa 3 quando detectar o estado do botão.

Exemplo de Pseudocódigo (simplificado):

```
// Variáveis globais para armazenar o estado
int button_state = 0;   // 0 = não pressionado, 1 = pressionado
int led_state = 0;      // 0 = apagado, 1 = aceso

// Função da Tarefa 1 - Leitura do Botão
void button_task(void *pvParameters) {
    for(;;) {
        button_state = read_button();  // Função fictícia para ler o botão
        vTaskDelay(pdMS_TO_TICKS(100)); // Delay de 100ms
    }
}

// Função da Tarefa 2 - Processamento do Botão
void process_button_task(void *pvParameters) {
    for(;;) {
        if (button_state == 1) {
            xTaskNotifyGive(led_task_handle); // Notifica a tarefa de LED
        }
        vTaskDelay(pdMS_TO_TICKS(10)); // Delay de 10ms
    }
}

// Função da Tarefa 3 - Controle do LED
void led_task(void *pvParameters) {
    for(;;) {
        ulTaskNotifyTake(pdTRUE, portMAX_DELAY); // Aguarda notificação
        if (button_state == 1) {
            led_state = 1;  // Acende o LED
        } else {
            led_state = 0;  // Apaga o LED
        }
        control_led(led_state);  // Função fictícia para acender o LED
    }
}

int main() {
    // Criação das tarefas
    xTaskCreate(button_task, "Button Task", 128, NULL, 1, NULL);
    xTaskCreate(process_button_task, "Process Button Task", 128, NULL, 2, NULL);
    xTaskCreate(led_task, "LED Task", 128, NULL, 3, &led_task_handle);
    // Inicia o agendador
    vTaskStartScheduler();
    return 0;
}
```

Instruções de Implementação:

1. Configuração de Hardware:

- Substitua as funções de leitura e controle do LED para trabalhar com pinos GPIO.

2. Teste da Aplicação:

- Você pode simular o sistema sem hardware físico, utilizando variáveis ou funções que representem o botão e o LED.

Prazo:	4 dias
Feedback Automático:	Será corrigido pelo professor/Tutor.
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.


embarca
tech
Residência
Tecnológica
em Sistemas
Embarcados