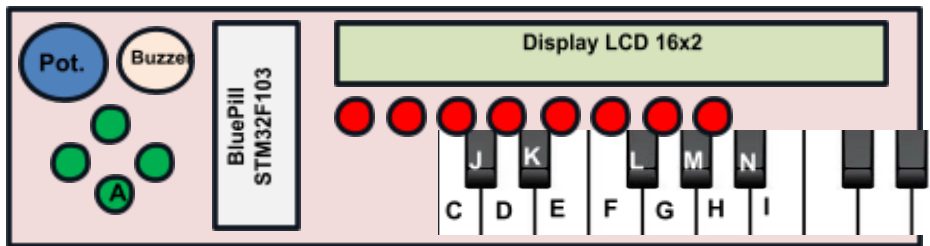


ECOS03 - Sistemas Operacionais Embarcados

Projeto Final 2025.1 - Prof. Otávio Gomes

Ambientação: Desenvolvimento de um sistema de controle de iluminação inteligente utilizando a placa STM32F103 e o FreeRTOS. O sistema deve gerenciar múltiplos padrões de iluminação em um painel. A luz de cada zona deve ser controlada com base em horários programados, pressionamentos manuais de botões e acionamentos de sensores.

Componentes da placa RedPill:



1) Desenvolva e execute as seguintes funções para a execução de cada um dos padrões para o acendimento contínuo e cíclico dos **LEDs de 0 a 7**:

- Função 01 - A lógica será para alternar o piscar entre LEDs pares e ímpares a cada segundo, criando uma animação sequencial.
- Função 02 - A lógica será para alternar o piscar entre LEDs seguindo o código Gray.
- Função 03 - Acende os LEDs conforme posição do potenciômetro. Assuma que o potenciômetro apresenta valores de 0 a 512 e defina intervalos para o acendimento dos 8 LEDs de modo a representar uma “barra de progresso” (acendimento sequencial crescente dos LEDs).
- Função 04 - Implementar feedback visual e sonoro. Por exemplo, adicione um sinal sonoro (*buzzer*) que aumenta sua frequência conforme mais LEDs são acesos (vide função 03). O som deve ser gerenciado em uma *thread* separada e sincronizado com a leitura do potenciômetro para evitar atrasos.

ECOS03 - Sistemas Operacionais Embarcados

Projeto Final 2025.1 - Prof. Otávio Gomes

2) Desenvolva e execute uma função para o controle de modo de piscar através dos **switches B e C**, do seguinte modo:

- Pressionar (uma vez) o botão C para selecionar o modo de piscar da função 01.
- Pressionar (uma vez) o botão D para selecionar o modo de piscar da função 02.
- Pressionar (uma vez) o botão E para selecionar o modo de piscar da função 03.
- Pressionar (uma vez) o botão F para selecionar o modo sonoro/mudo da função 04.

3) Utilizando os switches, desenvolva e execute uma função que controla a **velocidade** de contagem/piscar dos LEDs:

- Botão Y – Dobra o valor da velocidade atual
- Botão A – Diminui a velocidade atual para metade da velocidade atual
- Botão X – Retorna a velocidade ao valor-padrão definido no código
- Botão B – Define a velocidade de acordo com o valor estabelecido no potenciômetro.

4) Implementar um algoritmo de suavização, onde mudanças bruscas de velocidade (como apertar o botão Y ou A repetidamente) resultem em uma transição suave entre as velocidades, ao invés de mudanças instantâneas.

5) Utilizando o display, implemente um mostrador para as opções e teclas pressionadas e qual o estado atual das funções selecionadas.

6) Utilizando os switches, escolha botões que definirão 3 valores básicos de velocidade que possam ser escolhidos pelo usuário e que possam auxiliar na execução do item 3 desta proposta. Valores sugeridos: 0,5s (L), 2s(M), 10s(N).

Atenção:

- Todas as funções devem ser executadas através de Threads.
- Devem ser realizadas todas as integrações, adaptações e inserções necessárias no código de referência para a correta execução da função no sistema proposto.
- Não deve existir nenhum código dentro do loop infinito da main, isto é, todas as ações devem ser interrupções de tempo que executam as threads.

ECOS03 - Sistemas Operacionais Embarcados
Projeto Final 2025.1 - Prof. Otávio Gomes

- Com o aumento da complexidade e o uso de múltiplos recursos, como *display*, potenciômetro, buzzer, e LEDs, é interessante o uso de mutexes e semáforos para evitar condições de corrida e deadlocks.
- Algumas *threads* podem ter maior prioridade, como as de controle de interrupções e timers, enquanto outras, como o controle dos LEDs, podem ter prioridade mais baixa. O sistema deve ser configurado para garantir que threads de maior prioridade possam “*preemptar*” outras quando necessário.
- O desenvolvimento será individual.
- Esta atividade deverá ser **entregue** com o envio do código desenvolvido em arquivo compactado (formato zip), via **SIGAA**.