

**Instituição:** Instituto Federal de Educação, Ciência e Tecnologia - Paraíba (IFPB).

**Disciplina:** Microprocessadores e Microcontroladores.

**Professor:** Fagner de Araujo Pereira.

Alunos: \_\_\_\_\_

\_\_\_\_\_

## Exercícios – Programação de GPIO no STM32F407

**Realize a programação do GPIO no STM32F407, escolhendo corretamente os pinos, para implementar as seguintes tarefas:**

*(\*Utilizar resistores limitadores de  $470\Omega$  para os LEDs externos)*

*(\*Solicite ao professor o acesso à biblioteca com funções de temporização)*

*(\*Solicite ao professor o acesso à biblioteca do Display LCD)*

- ☐ 1. Piscar um LED numa frequência de 2Hz, com largura de pulso de 250ms.
- ☐ 2. Piscar um LED numa frequência de 0,5Hz, com largura de pulso de 100ms, produzindo um comportamento semelhante aos sinalizadores para aeronaves de baixa altitude que são instalados no topo de edifícios.
- ☐ 3. Piscar um LED com largura de pulso de 50ms, obedecendo o seguinte ciclo: pisca uma vez, aguarda um tempo e, em seguida, pisca duas vezes seguidas, aguardando mais um tempo.
- ☐ 4. Piscar um LED com largura de pulso fica de 50ms, mas com frequência crescente, iniciando em 1Hz.
- ☐ 5. Acionar um *buzzer* para emitir um som semelhante ao de um alarme sonoro: 4 *beeps* curtos de 50ms de largura, seguidos de uma pausa.
- ☐ 6. Mostrar um LED aceso com diferentes intensidades de brilho, selecionadas em tempo de compilação, utilizando uma técnica de PWM por meio de software (utilize uma frequência PWM de 100Hz).
- ☐ 7. Acender um LED com diferentes intensidades de brilho, alterando a intensidade em tempo de execução, utilizando uma técnica de PWM por meio de software (utilize uma frequência PWM de 100Hz). O LED deve iniciar apagado, aumentar gradativamente o brilho até o máximo e em seguida reduzir gradativamente o brilho até se apagar, repetindo o processo indefinidamente.
- ☐ 8. Piscar dois LEDs, alternadamente, com largura de pulso de 50% do período, com frequência de 2Hz, produzindo um comportamento semelhante aos sinalizadores de saída de veículos em garagens. (*obs.: verificar a simultaneidade de acionamento dos LEDs*).
- ☐ 9. Apresentar uma contagem binária de 2 bits a partir de 2 LEDs.
- ☐ 10. Acender dois LEDs com diferentes intensidades de brilho, alterando alternadamente a intensidade em tempo de execução, semelhantemente ao que foi feito na questão 7.
- ☐ 11. Apresentar uma contagem binária de 8 bits a partir de 8 LEDs externos. (*obs.: para facilitar a implementação do código, conecte os LEDs a pinos contíguos de um mesmo GPIO*).
- ☐ 12. Implementar o efeito do LED andante nos 8 LEDs da questão anterior: ative o primeiro LED e faça com que o LED aceso se desloque do primeiro ao último, ou seja, se o primeiro está aceso, após um tempo apaga-se esse LED enquanto acende-se o segundo, e assim sucessivamente até o oitavo LED. O LED andante deve se deslocar de um extremo a outro e em seguida fazer a operação no sentido inverso.
- ☐ 13. Simular o funcionamento de um cruzamento com dois semáforos de trânsito usando 6 LEDs.
- ☐ 14. Apresentar uma contagem decimal/hexadecimal (crescente e decrescente) de um dígito em um display de 7 segmentos.

- ☐ 15. Apresentar uma contagem decimal/hexadecimal (crescente e decrescente) de dois dígitos em dois displays de 7 segmentos. (obs: utilize a técnica de multiplexação para usar a menor quantidade possível de terminais). Utilize transistores como drive de potência para fazer a comutação do sistema de multiplexação.
- ☐ 16. Exibir o seu nome na primeira linha de um display de cristal líquido enquanto uma contagem regressiva iniciando em 10 aparece na segunda linha.
- ☐ 17. Controlar um micro servomotor para que ele alterne suavemente a posição do eixo entre direita e esquerda, continuamente.
- ☐ 18. Implementar o controle de velocidade de um motor DC para que ele acelere suavemente até sua velocidade máxima e em seguida desacelere e repita o mesmo procedimento no sentido contrário. Utilize uma ponte H como drive de potência para o motor.
- ☐ 19. Realizar o acionamento de uma carga em corrente alternada (AC) de 220V a partir de um relé.
- ☐ 20. Realizar o acionamento de um motor de passo com velocidade e sentido variáveis considerando que o motor seja monopolar e, também, bipolar. Para cada caso, realize o acionamento através de passo completo, meio passo e micro passo. Quando o motor for monopolar, implemente um drive de potência com transistores. Quando o motor for bipolar, utilize duas pontes H como drive de potência ou utilize um drive de potência integrado.
- ☐ 21. Reproduzir em uma saída com um LED o estado de uma entrada com um *push-button*.
- ☐ 22. Reproduzir em duas saídas com dois LEDs os estados de duas entradas com dois *push-buttons*.
- ☐ 23. Utilizar uma entrada com um *push-button* para alternar o estado lógico de uma saída conectada a um LED.
- ☐ 24. Utilizar dois *push-button* (K0 e K1) para acionar um LED com a seguinte regra: o LED só deve acender se o usuário pressionar os dois botões juntos, mas primeiro o botão K0 deve ser pressionado e só depois o botão K1.
- ☐ 25. Utilizar dois *push-button* (K0 e K1) para acionar um LED com a seguinte regra: o LED só deve acender se o usuário pressionar os dois botões juntos, mas primeiro o botão K0 e em seguida o botão K1. Se o botão K1 não for pressionado dentro de no máximo 1 segundo após K0 ser pressionado, o LED não deve acender.
- ☐ 26. Controlar um micro servomotor para que ele alterne suavemente a posição do eixo entre direita e esquerda por meio de dois *push-buttons*, sendo um para posicionar o eixo do motor mais à esquerda e outro mais à direita.
- ☐ 27. Utilizar um teclado de membrana, 4x4, e fazer sua decodificação, apresentando a tecla pressionada em um display de 7 segmentos.
- ☐ 28. Realizar o acionamento de uma carga em corrente alternada (AC) de 220V a partir de um relé, usando um sensor digital de proximidade (indutivo ou capacitivo) como sinal de gatilho.
- ☐ 29. Usar um sensor óptico e fazer a detecção de um objeto quando este cortar o feixe luminoso do sensor.
- ☐ 30. Implementar o antigo jogo da memória “Genius”. (Utilize as funções `rand()` ou `srand()` para criar uma sequência aleatória a ser memorizada)