



## Lista de Exercícios #2

### Pado Labs - Microcontroladores

Registradores e IOs

*Tips and Tricks* : Utilizar os documentos do kit para resolver os exercícios.

*Requirements* : Resolva pelo menos 7 exercícios. Exercícios com a tag **Challenge** valem por dois.

*Requirements* : Exercícios que requerem desenvolvimento de um código deve ser separado e zipado, com título do projeto e nome do arquivo fácil de identificar como *Lista2-Ex2.rar*, *L2-E2.rar*.

**1:** Qual a vantagem de se trabalhar com os tipos da biblioteca

*stdint.h*

R: Uma biblioteca que pode ser utilizada para definir valores

específicos de largura para números inteiros é a *stdint.h*.

**2:** Qual a principal característica de uma variável do tipo *int fast X\_t*?

R: Variável inteira que irá alocar o número de bits que implicará em melhor

performance do chipset.

**3:** No nosso kit NUCLEO-G0B1RE, qual o tamanho da variável, em bytes, do *int\_fast8\_t*, *int\_fast16\_t*, *int\_fast32\_t* e *int\_fast64\_t*.

R: o tamanho da variável em bits

int\_fast8\_t = 4 bytes  
int\_fast16\_t = 4 bytes  
int\_fast32\_t = 4 bytes  
int\_fast64\_t = 8 bytes

**4:** Qual a função dos registradores:

- GPIOX\_MODER :configurar o modo de Entrada/Saída.
- GPIOx\_OTYPER :configurar o tipo de saída de E/S.
- GPIOx\_OSPEEDR :configurar a velocidade de saída de E/S.
- GPIOx\_PUPDR : configurar o pull-up ou pull-down de E/S.
- GPIOx\_IDR : contém o valor de entrada da porta de E/S, somente a leitura.
- GPIOx\_ODR : pode ser lido e escrito por software.
- GPIOx\_AFRL : configura a E/S da função alternativa.

**5:** Como posso fazer para ler diretamente o registrador sem utilizar a implementação da ST? (*tip*: lembre-se dos ponteiros!)

R: Usando o endereço de memória para ler usando um ponteiro.

**6:** Desenvolva um firmware que pisque o LD4 com uma frequência de 1Hz (500ms aceso, 500ms apagado)

1

**7:** Desenvolva um firmware que pisque o LD4 com uma frequência de 100Hz.

**8:** Faça um programa que pisque o LD4 em 20 Hz enquanto o botão USER é

pressionado e pisque com frequência de 5Hz ao ser solto.

**9:** Faça a leitura dos *switches* do tipo DIP de 4 posições utilizando os resistores de *pull-up* ou *pull-down* internos. Armazene em uma variável o valor correspondente, onde a chave 1 corresponde ao bit 0, e o bit 4 corresponde ao bit 3.

**10:** Aproveite o exercício anterior, monte 4 LEDs e associe cada um deles a uma chave do DIP switch de 4 posições, quando a chave estiver em ON, acenda o LED, e em OFF, apague o LED.

**11: Challenge:** Aproveite o exercício anterior novamente, mas sem os LEDs, e exiba em um display de 7 segmentos o valor correspondente em hexadecimal (0 à F).

**12: Challenge:** Existe uma técnica comumente chamada de *Varredura*, esta técnica consiste em ligar elemento em matriz para otimizar o uso de GPIOs, muito utilizada para acionar LEDs e realizar a leitura de botões e *keypads*. Nisto, como desafio, deve-se montar o circuito da figura 1 e desenvolver um firmware que faça a leitura dessas teclas e armazene em uma variável a linha e coluna da tecla pressionada (a linha e coluna deve ser numerada de 1 a 4, quando nenhuma tecla estiver pressionada, deve ser exibido o valor 0).

Figura 1: Esquemático de uma matriz de botões 4x4.

