

# Interrupções e Exceções

## Questões

Lucas Gonçalves Serrano - RA: 12.01328-5

Flávia Janine Béo Rosante - RA: 13.03188-0

Erica Yumi Kido - RA: 13.02422-0

4 de maio de 2016

## 1 Exceções

### Questão. 1.1: NMI vs IRQ

A exceção NMI tem uma prioridade de exceção maior que todas depois do *reset* fixada no -2; essa exceção não pode ser mascarada ou prevenida por qualquer outra exceção. Ela pode ser sinalizada por um periférico ou acionada pelo software.

A exceção IRQ é usada pelo periférico para se comunicar com o processador, sua prioridade é configurável. Ela é sinalizada por um periférico ou gerada pelo software. O CMSIS só usa números IRQ, os negativos para todas as exceções, que não sejam interrupções.

## 2 Interrupção

### Questão. 2.1: IRQ vs ISR

A exceção IRQ é uma chamada de interrupção pelo periférico, o ISR é um serviço, o "request" da interrupção é feito pelo ISR.

### 2.1 Prioridades

#### Questão. 2.2: SAM4S

Configuração de prioridades de interrupção em NVIC: Nested Vectored Interrupt Controller, prioridade vem do bit mais significativo. Configuração CMSIS: Cortex Microcontroller Software Interrupt Standart, não precisa fazer shift ao chamar o parâmetro na função. O nível de prioridades preempt priority define se a interrupção pode ser chamada quando já está rodando em outro handler do processador. O subpriority é usado quando duas exceções com o mesmo nível preempt estão pendentes.

#### Questão. 2.3: FIQ

Quando o FIQ é recebido uma sinalização é feita e o contador do programa e o registro da condição do código são armazenados numa pilha. Depois do final da interrupção, a interrupção pega o registro da condição do código, que armazena a condição inicial dos dados digitais processados, e checa se houve uma sinalização ou não, se houver a sinalização do FIQ o contador do programa não foi desempilhado.

**Questão. 2.4: IRQ vs FIQ**

FIQ tem uma prioridade maior.

**2.2 Interrupt Requests - IRQ****Questão. 2.5: SAM4S número da interrupção dos periféricos**

- **PIOA** → Id : 11
- **PIOB** → Id : 12
- **TC0** → Id : 23

**Questão. 2.6: Limpando a Interrupção**

Se não ocorrer a limpeza da interrupção ela fica setada, mesmo após sua ocorrência. Ou seja, mesmo que a interrupção já tenha ocorrido, continua havendo uma sinalização que há uma interrupção assinalada.

**2.3 Interrupt Service Routine - ISR****Questão. 2.7: Latência da interrupção.**

Latência na resolução é o tempo desde a chamada da interrupção até a resolução da mesma. Nesse tempo é feita a função definida na chamada da interrupção, depende do software implementado. O processador para e decide o que fazer.

**Questão. 2.8: Latência Cortex M4.**

Latência de interrupção de 12 ciclos.

**3 Software - CMSIS****4 Software - ASF****Questão. 4.1: ASF - PIO**

As funções desenvolvidas em sala podiam ser designadas para um PIO específico ou podiam se referir a um PIO genérico, como as funções disponíveis pela ASF.

**Questão. 4.2: ASF - Timer Counter (TC)**

- **tc\_init()**  
Instancia e inicializa o contador do timer que possui inúmeros canais para realizar funções diversas.
- **tc\_start()**  
Inicializa um contador passado nos parâmetros da função.
- **tc\_enable\_interrupt()**  
Permite a interrupção em um canal do timer passada no parâmetro da função.

## 5 PIO - Interrupção

### Questão. 5.1: PIO - Interrupção Botão

O PIO deve ser configurado como uma fonte NVIC também, o registrador de status do PIO correspondente deve ser colocado em em pull\_down, e prioridade zero para poder ativar a interrupção.

### Questão. 5.2: PIO - Interrupção

A interrupção é usada para interromper o controlador quando houver uma detecção do evento estar com a borda subindo ou descendo (PIO AIMMR e PIO FLRHSR), ou quando está em nível de subida ou descida (PIO ELSR). Quando o PIO ISR é setado todas as interrupções são lidas e feitas.

### Questão. 5.3: Registradores Interrupção

- **PIO\_IER / PIO\_IDR**
  - O PIO\_IER é o registrador que permite interrupções.
  - O PIO\_IDR é o registrador que desabilita interrupções.
- **PIO\_AIMER / PIO\_AIMDR**
  - O PIO\_AIMER é o registrador que habilita outros modos de interrupções.
  - O PIO\_AIMDR é o registrador que desabilita outros modos de interrupções.
- **PIO\_ELSR**

É o registrador que determina se interrupção será percebida na borda ou no nível.
- **PIO\_FRLHSR**

É o registrador que determina se interrupção será percebida na na borda de subida ou descida ou então no nível alto ou baixo, dependendo do registrador anterior.