

<b>Disciplina</b>	Sistemas Embarcados 1
<b>Professor</b>	Marcelo Barros de Almeida
<b>Entrega</b>	Via Tarefa no Teams
<b>Link</b>	

## Arquitetura de Computadores

- 1) Explique, com suas palavras e em detalhes, a finalidade dos registros **PC** (Program Counter), **SR** (Status Register) e **SP** (Stack Pointer).
- 2) Faça um comparativo entre os processadores RISC-V (use um core RV32IMAC), ARM Cortex M3 e MSP430 com relação aos registros internos (quantidade de registros, tamanho, uso principal, finalidade geral ou específicos, etc), arquitetura básica (CISC x RISC), alinhamento de memória (big ou little endian), tamanho do conjunto de instruções e mapeamento de memória para flash, RAM e periféricos.
- 3) Ainda sobre os processadores RISC-V, ARM Cortex M3 e MSP430, compare os flags existentes dentro do registro de status de cada um deles. Comente as diferenças e igualdades.
- 4) Descreve o processo de inicialização dos controladores RISC-V, ARM Cortex M3 e MSP430, evidenciando de onde começam a buscar instruções, como funcionam os vetores de interrupção, definição do stack, etc. Comente sobre os pontos positivos e negativos de cada uma e qual seria a sua escolha.
- 5) Como é o contexto de interrupção dos controladores RISC-V, ARM Cortex M3 e MSP430 ? Descreva-os detalhadamente e adicione desenhos mostrando a estrutura criada na pilha para o tratamento da interrupção. Em quais plataformas pode existir preempção ?
- 6) Monte em hexadecimal o código das instruções abaixo. Explique também o seu significado. Use como referência o conjunto de instruções da arquitetura ARMv6-M (usado no Cortex M0, M0+ e M1):  
  
ASR R7, R4, #9  
BICS R4, R1  
LDMIA R3!, {R1,R4,R7}
- 7) Em modo de execução normal, o Cortex M pode fazer uma chamada de função usando a instrução BL, que muda o PC para o endereço de destino e salva o ponto de execução atual no registro LR. Ao final da função, é possível recuperar esse contexto usando uma instrução BX LR, por exemplo, que atualiza o PC para o ponto anterior. No entanto, quando acontece uma interrupção, o LR é preenchido com um valor completamente diferente, chamado de EXC\_RETURN. Explique o funcionamento desse mecanismo e especifique como o Cortex consegue fazer o retorno da interrupção.
- 8) Apesar de o Cortex M3 e M4 serem baseado na arquitetura ARMv7-M, eles possuem algumas diferenças. Cite pelo menos 3 diferenças entre eles.
- 9) Descreva como funciona o mecanismo de priorização de interrupções no Cortex M, descrevendo a estratégia de “group priority” e “sub-priority” presente nesse processo.
- 10) Qual a diferença entre as instruções WFI e WFE ? Qual a diferença entre evento e interrupção ?
- 11) Cite pelo menos 3 recursos presentes no Cortex M3/M4 que favorecem a implementação de sistemas operacionais de tempo real (RTOS).
- 12) Qual a diferença no salvamento de contexto, durante a chegada de uma interrupção, entre os processadores Cortex M3 e Cortex M4F (com ponto flutuante) ? Descreva em termos de tempo e também de uso da pilha. Explique também o que é “lazy stack” e como ele é configurado.