

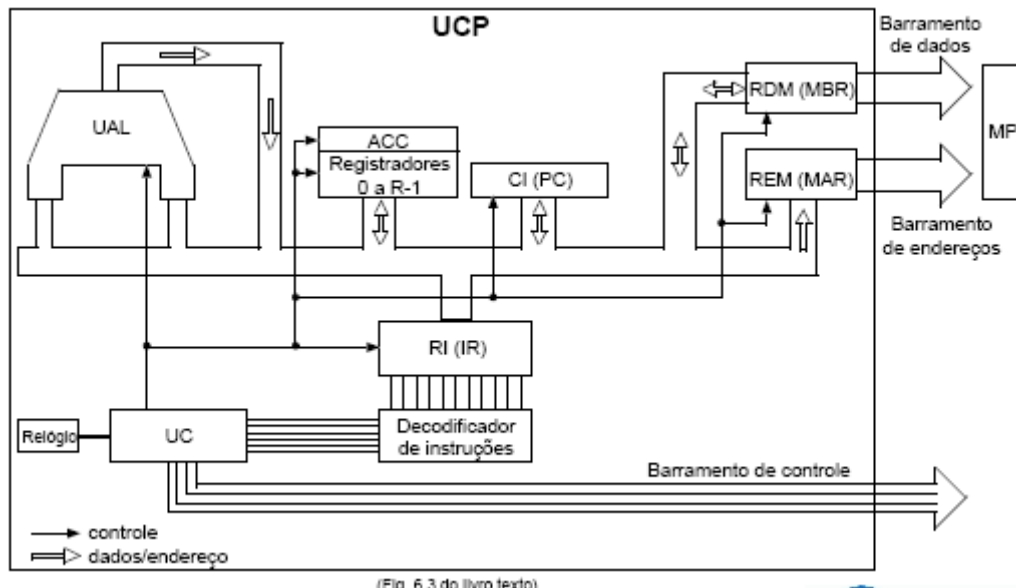
Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
3. Você pode usar lápis para responder as questões.
4. Ao final da prova devolva as folhas de questões e as de respostas.
5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.

1. (2,5) Considere o sistema apresentado em aula mostrado na figura abaixo.



Suponha que uma nova instrução denominada “**DIV Op.**” foi inserida no conjunto de instruções desta máquina. Na execução desta instrução o conteúdo da célula de memória cujo endereço é **Op.** é dividido pelo conteúdo do acumulador e o resultado é armazenado na memória no endereço **Op.**

Descreva **detalhadamente** a execução desta instrução, indicando como o Acumulador (ACC), RDM, REM, Unidade Aritmética Lógica (UAL) e Barramento de controle, de dados e de endereços são utilizados na execução desta nova instrução.

Resposta:

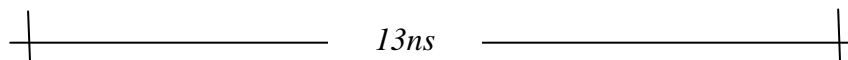
- a) $RI <- (CI)$
b) $CI <- CI + 1$
c) Decodificação do código de operação
- recebe os bits do código de operação
- produz sinais para a execução da operação de divisão
d) Busca do operando na memória
- A UC emite sinais para que o valor do campo operando (Op) seja transferido para o REM
- A UC ativa a linha READ do barramento de controle
- Conteúdo de memória do endereço (Op) é transferido para o RDM
e) Execução da operação
- Dados para a operação de divisão são transferidos para a UAL:
 $UAL(A) <- RDM$ e $UAL(B) <- ACC$ (A e B são as entradas da UAL)
- UAL executa a operação de divisão: $UAL(A) / UAL(B)$, ou seja, RDM / ACC
- RDM recebe o resultado: $RDM <- RDM / ACC$
e) Armazenando o resultado
- A UC emite sinais para que o valor do campo operando (Op) seja transferido para o REM
- O RDM contém o resultado a ser guardado no endereço da memória
- A UC ativa a linha WRITE do barramento de controle
- Conteúdo do RDM é transferido, através do barramento de dados, para o endereço Op da memória, endereço este transferido do REM a partir do barramento de endereços.

2. (2,0) Considere uma máquina que pode ter seu ciclo de busca e execução de uma instrução dividido em 5 estágios totalmente independentes: Busca de Instrução (BI), Decodificação (DI), Cálculo de Endereços de Operandos (CO), Busca dos Operandos (BO) e Execução (EX). Cada um dos estágios BI, BO e EX possui a duração de 3 ns e cada estágio DI e CO tem duração de 2 ns. Cada instrução desta máquina precisa executar os 5 estágios que serão sempre executados na sequência BI, DI, CO, BO e EX.

- a) (0,2) Uma implementação desta máquina foi realizada de modo que cada instrução deve ser completamente realizada **em um único ciclo de relógio**. Calcule **a duração do ciclo de relógio** que esta implementação deve possuir. Lembre-se que todas as instruções necessitam dos 5 estágios.

Resposta:

<i>BI</i> <i>3ns</i>	<i>DI</i> <i>2ns</i>	<i>CO</i> <i>2ns</i>	<i>BO</i> <i>3ns</i>	<i>EX</i> <i>3ns</i>
-------------------------	-------------------------	-------------------------	-------------------------	-------------------------



Ciclo de relógio para execução de uma instrução (sem pipeline) =
 $3ns + 2ns + 2ns + 3ns + 3ns = 13ns$

- b) (0,8) Como cada estágio é independente um do outro, deseja-se implementar uma **nova** arquitetura utilizando-se um pipeline de 5 estágios. Nesta nova implementação **cada estágio do pipeline** deve ser executado em um ciclo de relógio. Calcule **a duração do ciclo de relógio** que esta implementação pipeline deve possuir.

Resposta:

1°. estágio	2°. estágio	3°. estágio	4°. estágio	5°. estágio
<i>BI</i> <i>3ns</i>	<i>DI</i> <i>3ns</i>	<i>CO</i> <i>3ns</i>	<i>BO</i> <i>3ns</i>	<i>EX</i> <i>3ns</i>

Ciclo de relógio será igual ao tempo para execução do estágio com maior tempo de execução = 3ns.

- c) (1,0) Considere um programa que necessita executar 100 instruções. Calcule o tempo de execução deste programa na máquina do item **a** e na máquina do item **b**. Caso as duas máquinas custem o mesmo preço, qual das duas voce compraria para executar este programa? **Explique sua resposta.**

Resposta:

Seja T_{ex} = tempo de execução de uma instrução = número de estágios \times ciclo de relógio (determinado nos itens anteriores)

Para o item a (sem pipeline) :

$$T_{ex} = 1 \text{ estágio} \times 13ns = 13ns$$

$$T_{total} = 100 \text{ instruções} \times T_{ex} = \underline{1300ns}$$

Para o item b (pipeline: 5 estágios) :

$$T_{ex} = 5 \text{ estágios} \times 3ns = 15ns$$

$$T_{total} = T_{ex} + 99 \times \text{tempo de 1 estágio}$$

$$T_{total} = 15ns + 99 \times 3ns = \underline{312ns}$$

Este programa executa mais rapidamente na máquina do item b. Como o preço das duas máquinas é o mesmo, eu compraria a máquina do item b porque fornece melhor desempenho na execução deste programa.

3. (1,5) Explique como funciona uma Unidade Central de Processamento (UCP) cujo controle é realizado por hardware e uma UCP com controle por microprograma.

Resposta:

Em uma UCP cujo controle é realizado por hardware, o desenvolvimento da unidade de controle consiste essencialmente em projetar circuitos combinatórios. Os sinais lógicos de entrada na unidade devem ser transformados em um conjunto lógico de sinais que controlam a execução da instrução. Para implementar a unidade, necessita-se derivar, para cada sinal de controle a ser gerado para que cada instrução seja executada de forma correta, uma expressão booleana que define esse sinal em função dos sinais de entrada referentes à instrução.

Em uma UCP cujo controle é realizado por microprograma, a unidade de controle é projetada de modo a executar uma sequência de microinstruções (ou conjunto de microoperações) e gerar sinais de controle para os componentes da UCP para a execução de cada microinstrução. A Unidade de Controle microprogramada é composta pelos seguintes componentes: Memória de controle, Contador de microprograma e Sequenciador.

4. (2,0) Considere um microprocessador hipotético com um tamanho de palavra e barramento de dados de 40 bits e 256 códigos de operação diferentes. As instruções são todas do tamanho de uma célula de memória, de mesmo tamanho da palavra e são compostas por dois campos: o primeiro contém o código de operação e o restante contém o endereço do operando.

- a) Qual o tamanho, em bits, do RDM e do REM?

Resposta:

Barramento de dados = 40 bits

*RDM = barramento de dados = **40 bits***

REM = barramento de endereços = bits necessários para acessar toda memória

Bits necessários para endereçar toda memória = operando

Uma instrução = tamanho da palavra = 40 bits

uma instrução = código de operação + operando

40 bits = 8 bits (necessários para 256 instruções diferentes) + operando

operando = 32bits

*portanto, REM = barramento de endereços = **32 bits***

- b) Descreva o processo de leitura e gravação na memória, explicitando o uso do RDM e REM.

Resposta:

Os processos de leitura e gravação serão apresentados na forma de passos:

Passos para uma operação de escrita

1) (REM) <- (outro registrador)

1.1) O endereço é colocado no barramento de endereços

2) (RDM) <- (outro registrador)

2.1) O dado é colocado no barramento de dados

3) Sinal de escrita é colocado no barramento de controle

4) (MP(REM)) <- (RDM)

Passos para uma operação de leitura

1) (REM) <- (outro registrador da UCP)

1.1) O endereço é colocado no barramento de endereços

2) Sinal de leitura é colocado no barramento de controle

2.1) Decodificação do endereço e localização da célula na memória

3) (RDM) <- (MP(REM)) pelo barramento de dados

4) (outro registrador da UCP) <- (RDM)

5. (2,0) Considere uma máquina que possa endereçar 4 Gbytes de memória física, sendo que cada endereço referencia uma célula de 1 byte. Ela possui uma memória cache que pode armazenar 1K blocos, sendo um bloco por linha e cada bloco possui 2 K células. Mostre o formato da memória cache, indicando os campos necessários (tag, bloco) e o número de bits para cada campo, o formato de um endereço da memória principal, indicando os bits que referenciam os campos da cache, e a capacidade em bits que a memória cache deve possuir (pode deixar a conta indicada) para os seguintes mapeamentos:

Resposta:

a) Mapeamento direto.

Memória Principal

- ⇒ Tamanho da memória (em bytes) = 4Gbytes, como 1 célula referencia a 1 byte, temos $N = 4G$ células
- ⇒ Será organizada em blocos de 2K bytes, como 1 célula = 1 byte, temos cada bloco = 2K células, $K = 2K$
- ⇒ Sendo N o tamanho endereçável da memória e K que é a quantidade de células por blocos temos:
- $N = 4G$ células e $K = 2K$ células / blocos o total de blocos da MP (B) será:
- Total de blocos: $B = N / K \Rightarrow B = 4G \text{ células} / 2K \text{ células/bloco} \Rightarrow$
- $B = 2M$ blocos

Memória Cache

OBS: O K (quantidade de células/bloco) tem de ser igual a MP.

- ⇒ Tamanho da memória cache (em blocos ou linhas) $\Rightarrow Q = 1K$ blocos
- ⇒ Tamanho da memória cachê em células = $Q \times K =$
- $1K \text{ blocos} \times 2K \text{ células/bloco} = 2M \text{ células}$

Cada célula possui 1 byte = 8bits, então,
como a cache possui 2M células x 8bits, que totaliza 16M bits

Endereço da MP: Para endereçarmos toda a MP precisamos da seguinte quantidade de bits (E) sendo $N = 2^E \Rightarrow N = 4G \text{ células} \Rightarrow N = 2^{32} \Rightarrow E = 32 \text{ bits}$

Composição do endereço em função da memória cache

- $\Rightarrow \text{tag} = B / Q = 2M / 1K = 2K = 2^{11} = 11\text{bits}$
- $\Rightarrow n^\circ \text{ da linha: } Q = 1K \text{ linhas ou quadros (máximo)} \Rightarrow 2^{10} \Rightarrow 10\text{bits}$
- $\Rightarrow \text{células por bloco: } 2K \text{ células por bloco} = 2^{11} \Rightarrow 11\text{bits}$

32bits		
Tag = 11bits	No. Linha =10bits	Célula no bloco=11bits

b) Mapeamento totalmente associativo.

Memória Principal

- $\Rightarrow N = 4G \text{ células}$
- $\Rightarrow K = 2K \text{ células/bloco}$
- $\Rightarrow B = 2M \text{ blocos}$

Memória Cache

OBS: O K (quantidade de células/bloco) tem de ser igual a MP.

- $\Rightarrow Q = 1K \text{ blocos}$
- $\Rightarrow \text{Tamanho da memória cache} = 2M \text{ células ou } 2M\text{bytes ou } 16M\text{bits}$

Endereço da MP = 32 bits

Composição do endereço em função da memória cache

=> $tag = B = 2M = 2^{21} \Rightarrow tag = 21bits$

=> células por bloco: $2K \text{ células por bloco} = 2^{11} \Rightarrow 11bits$

32bits

$Tag = 21bits$	$Célula \text{ no } \text{bloco} = 11bits$
----------------	--