

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. (2,0) Considere um microprocessador hipotético com um tamanho de palavra e barramento de dados de 40 bits e 256 instruções. 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. Ele contém somente uma placa de memória com 5 Kbytes.

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

Barramento de dados = 40 bits, como tamanho do barramento de dados igual ao tamanho do RDM, temos
RDM = 40bits

Tendo 256 instruções (2^8 instruções), temos o código de operação = 8bits
Tamanho da instrução: código de operação + operando (endereço) = tamanho da célula
Tamanho da palavra = tamanho da célula
Temos então: 8 + operando = 40, portanto, operando = 32.
O operando corresponde a um endereço de memória = 32 bits
REM = 32 bits.

- b) Seria possível aumentar a memória desta máquina ? Se possível, até quanto (em **bytes**) ?

Quantidade de células que podem ser acessadas = $2^{32} = 4 \text{ G células}$ (Giga células)

Cada célula = 40 bits = 5 bytes

Portanto temos: $4 \text{ G células} \times 5 \text{ bytes} = 20 \text{ Gbyte}$

- c) Se quisesse aumentar o número de instruções para 512, mantendo inalterado o tamanho do barramento de dados e da palavra, quais seriam os tamanhos do RDM e REM ?

Tendo 512 instruções (2^9 instruções), temos o código de operação = 9bits

Tamanho da instrução: código de operação + operando (endereço) = tamanho da célula

Tamanho da palavra = tamanho da célula

Temos então: $9 + \text{operando} = 40$, portanto, $\text{operando} = 31$.

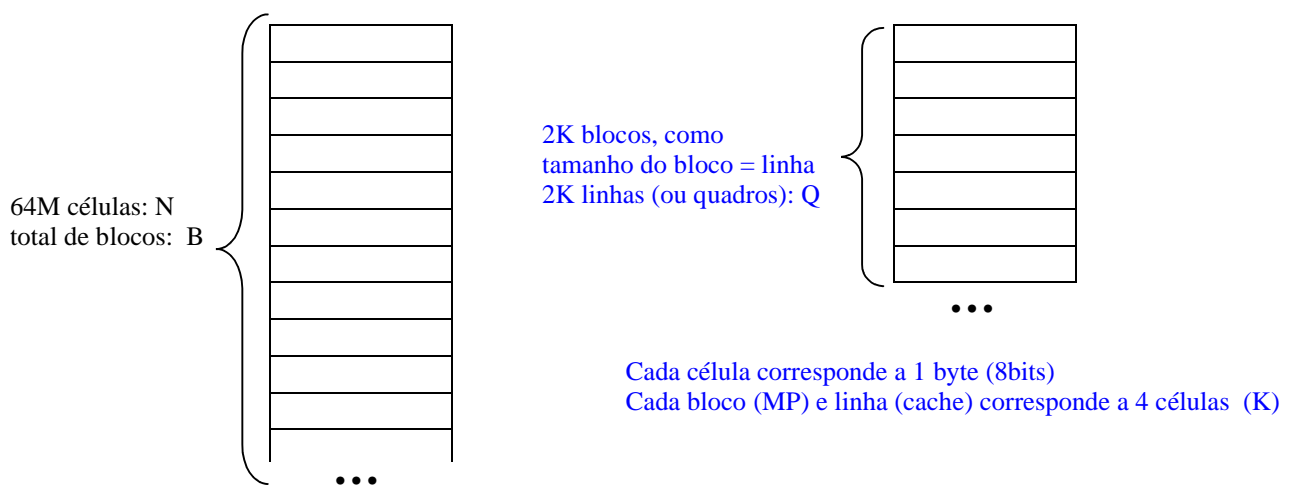
O operando corresponde a um endereço de memória = 31 bits

REM = 31 bits.

RDM = tamanho da palavra = 40bits

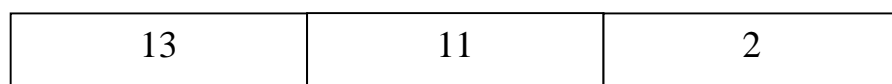
2. (2,0) Considere uma máquina que possa endereçar 64 Mbytes de memória física, sendo que cada endereço referencia uma célula de 1 byte. Ela possui uma memória cache que pode armazenar 2K blocos, sendo um bloco por linha e cada bloco possui 4 células. Mostre o formato da memória cache, indicando os campos 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:
- a) Mapeamento direto.

Tamanho da memória (em bytes) = 64Mbytes, como 1 célula = 1 byte, temos $N = 64M$ células



Total de blocos: $B = N / K \Rightarrow B = 64M \text{ células} / 4\text{células/bloco} \Rightarrow B = 16M \text{ blocos}$

Endereço da memória principal corresponde a E
sendo $N = 2^E \Rightarrow N = 64M \text{ células} \Rightarrow N = 2^{26} \Rightarrow E = 26 \text{ bits}$



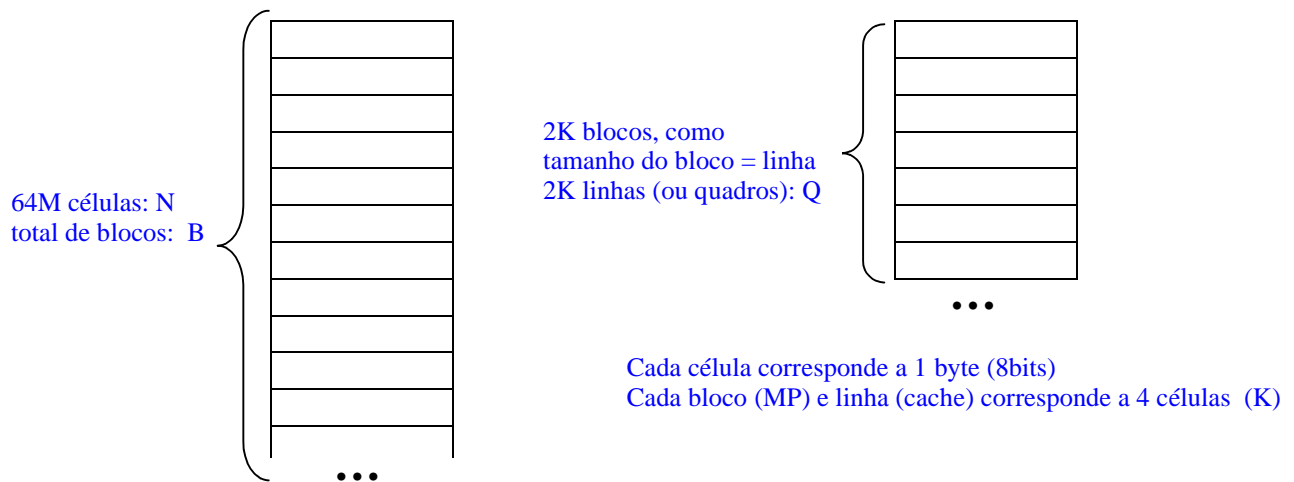
Número do bloco no quadro:
 $\text{tag} = B / Q = 16M / 2K = 8K = 2^{13}$

Corresponde ao nº do quadro:
 $Q = 2K \text{ linhas ou quadros (máximo)} \Rightarrow 2^{11}$

Corresponde ao endereço da palavra:
4 células por bloco = 2^2

b) Mapeamento totalmente associativo.

Tamanho da memória (em bytes) = 64Mbytes, como 1 célula = 1 byte, temos $N = 64M$ células



Total de blocos: $B = N / K \Rightarrow B = 64M \text{ células} / 4\text{células/bloco} \Rightarrow B = 16M \text{ blocos}$

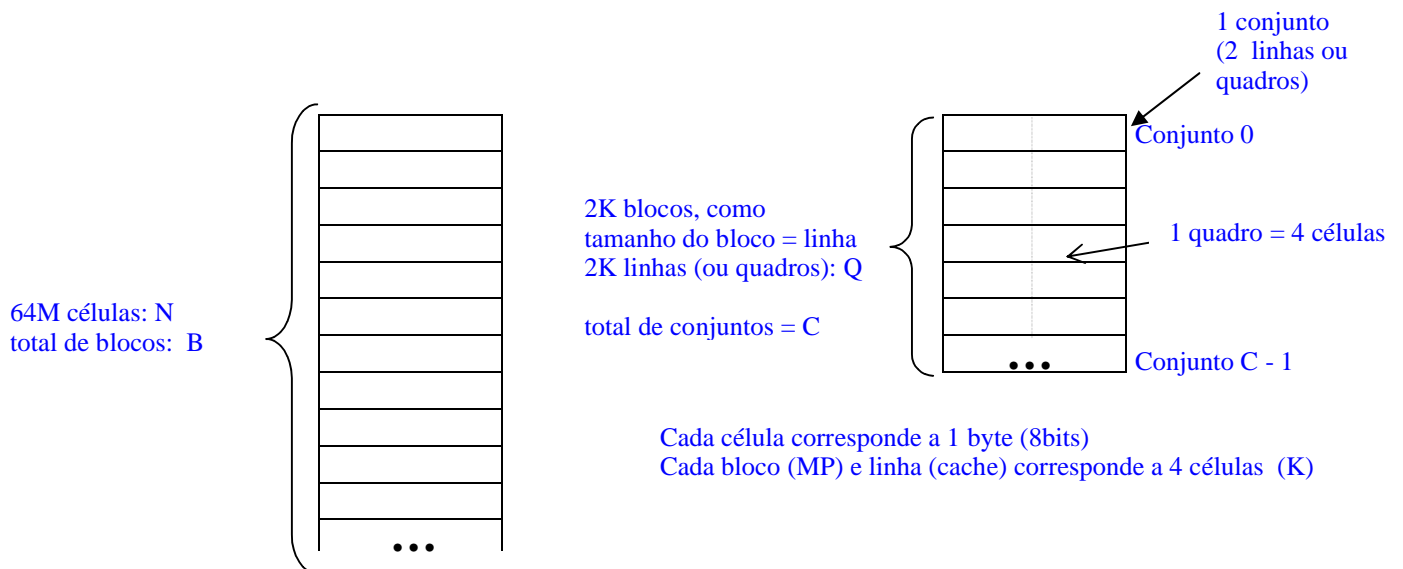
Endereço da memória principal corresponde a E
sendo $N = 2^E \Rightarrow N = 64M \text{ células} \Rightarrow N = 2^{26} \Rightarrow E = 26 \text{ bits}$



Corresponde ao bloco:
 $\text{tag} = B = 16M = 2^{24}$

Corresponde ao endereço da palavra:
4 células por bloco = 2^2

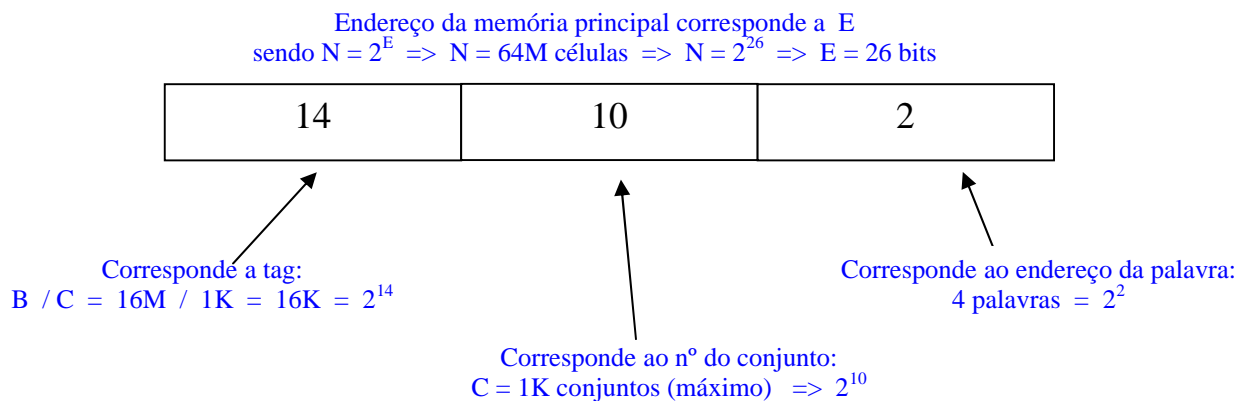
- c) Mapeamento associativo por conjunto, onde cada conjunto possui duas linhas, cada uma de um bloco.



Na memória cache, a quantidade de linhas (ou quadros): $Q = C \times D$ sendo,
 Q a quantidade de linhas (ou quadros)
 C o total de conjuntos
 D a quantidade de linhas (ou quadros) por cada conjunto.

Total de blocos na MP: $B = N / K \Rightarrow B = 64M \text{ células} / 4\text{células/bloco} \Rightarrow B = 16M \text{ blocos}$

Para sabermos a quantidade de conjuntos
 $Q = C \times D \Rightarrow 2K = C \times 2 \Rightarrow C = 1K \text{ conjuntos}$



3) (2,0) Suponha que você deve projetar uma máquina com as seguintes especificações:

- Capaz de endereçar 256 K células de memória principal.
- Deve possuir um registrador Acumulador, além do RDM (Registrador de Dados da Memória), REM (Registrador de Endereços da Memória), CI (Contador de Instrução) e RI (Registrador de Instrução).
- Cada instrução deve conter um código de operação com 4 bits e um operando como mostrado abaixo, onde o operando indica um endereço de memória:

Cód. Oper.	Operando
------------	----------

- A Unidade Central de Processamento obtém uma instrução da memória principal realizando somente um acesso à memória principal.

a) Calcule o tamanho mínimo em bits do REM e do barramento de endereços.

Barramento de endereços terá capacidade de endereçar 256K células (N)
Barramento de endereços = E, $N = 256K \Rightarrow N = 2^{18} \Rightarrow E = 18$
Barramento de endereços = 18 bits = REM

b) Calcule o tamanho mínimo em bits que a instrução deve ter.

Tamanho de bits da instrução = tamanho cód. oper. + tamanho operando
Tamanho da instrução = 4 bits + 18 bits (endereço de memória) = 22bits

c) Calcule o tamanho de RI e CI.

Considerando RI = tamanho de uma instrução, portanto RI = 22bits

CI = endereço de memória = tamanho do barramento de endereços, logo CI = 18bits

d) Calcule o número de instruções diferentes que o conjunto de instruções desta máquina pode ter.

Total de instruções = 2^4 (sendo 4 o tamanho do código de operação) = 16 instruções

e) A instrução “ADD operando” desta máquina, faz com que o conteúdo do acumulador seja somado ao conteúdo de memória cujo endereço é dado por operando. Indique como a instrução “ADD 05” será codificada em bits, sabendo que o código de ADD é 0001, e explique como será realizada a execução desta instrução nesta máquina.

Correspondência em binário:

ADD 05 \Rightarrow 000100 00000000 00000101

Como será realizada a execução desta instrução:

Passo 1: A CPU coloca no REM o valor do operando, e este é disponibilizado no barramento de endereços

Passo 2: A CPU disponibiliza o sinal de leitura de memória no barramento de controle

Passo 3: A memória coloca o conteúdo da memória cujo endereço é 05 no barramento de dados, e este é transferido para o RDM da CPU

Passo 4: A CPU executa a soma do valor armazenado em RDM com o valor armazenado no acumulador, e armazena o resultado no acumulador

4. (1,6) Considere as características de 4 barramentos síncronos relativas ao número de bits que são transmitidos em um ciclo de relógio e à frequência do relógio, conforme mostrado na tabela abaixo:

Tipo de barramento	Número de bits/ciclo	Frequência do relógio (MHz)
T1	64	800
T2	32	400
T3	40	500
T4	48	600

Indique, para cada um dos barramentos, a taxa de transferência em Gbytes/s.

OBS: a) $1 \text{ MHz} = 1 \times 10^6 \text{ ciclos/segundo}$

b) Para taxas de transferência considera-se $1 \text{ Kbyte/s} = 1.000 \text{ bytes/s}$ e $1 \text{ Mbyte} = 1.000.000 \text{ bytes/s}$.

Taxa de transf. (bytes/s) = (númerodebits/ciclo / 8bits/byte) x Frequência do relógio

T1 => taxa de transferência = $(64 \text{ bits/ciclo} / 8 \text{ bits/byte}) \times 800 \times 10^6 \text{ ciclos/segundo}$
⇒ taxa de transferência = $6400 \times 10^6 \text{ bytes/segundo}$
⇒ taxa de transferência = 6,4 Gbytes/s

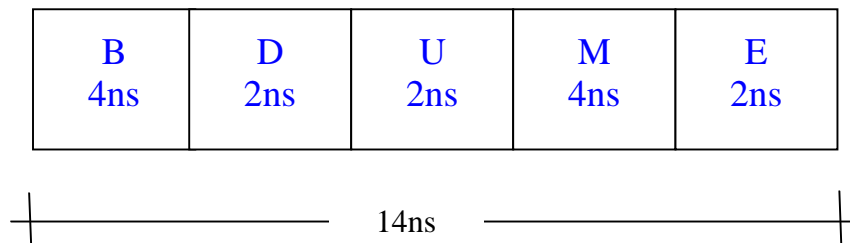
T2 => taxa de transferência = $(32 \text{ bits/ciclo} / 8 \text{ bits/byte}) \times 400 \times 10^6 \text{ ciclos/segundo}$
⇒ taxa de transferência = $1600 \times 10^6 \text{ bytes/segundo}$
⇒ taxa de transferência = 1,6 Gbytes/s

T3 => taxa de transferência = $(40 \text{ bits/ciclo} / 8 \text{ bits/byte}) \times 500 \times 10^6 \text{ ciclos/segundo}$
⇒ taxa de transferência = $2500 \times 10^6 \text{ bytes/segundo}$
⇒ taxa de transferência = 2,5 Gbytes/s

T4 => taxa de transferência = $(48 \text{ bits/ciclo} / 8 \text{ bits/byte}) \times 600 \times 10^6 \text{ ciclos/segundo}$
⇒ taxa de transferência = $3600 \times 10^6 \text{ bytes/segundo}$
⇒ taxa de transferência = 3,6 Gbytes/s

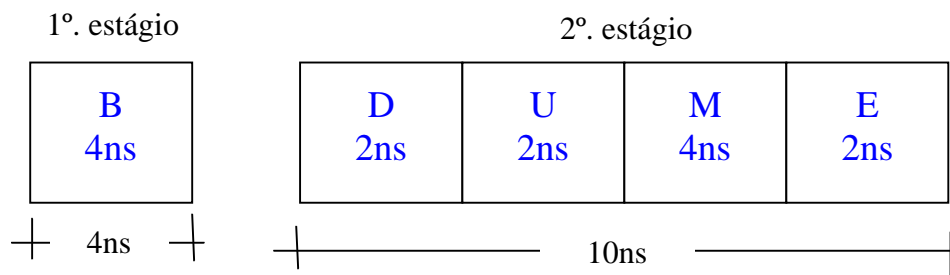
5. (2,4) Considere uma máquina, que pode ter o ciclo de busca e execução de uma instrução dividido em 5 blocos **independentes**: Busca de instrução (B), Decodificação/Leitura de registradores (D), Operação da ULA (U), Acesso à memória de dados (M) e Escrita nos registradores (E). Foram realizadas três implementações para esta máquina: (1) sem pipeline, ou seja, cada instrução deve ser completamente realizada em um ciclo de relógio e uma instrução só começa a ser realizada após o término da anterior, (2) pipeline de 2 estágios, sendo que o primeiro estágio é composto do bloco B e o segundo é composto dos blocos D, U, M e E e (3) pipeline de 5 estágios, onde cada estágio é composto de um bloco. Cada estágio B e M possui uma duração de 4 ns e cada estágio D, U e E tem duração de 2ns. Para cada implementação deve existir um relógio com frequência fixa, ou seja, a duração do ciclo de relógio é única, não pode variar ao longo da execução das instruções.

- a) (0,5) Calcule a duração do ciclo de relógio que a implementação monociclo sem pipeline deve possuir. Neste caso, qualquer instrução deve poder ser realizada em um ciclo de relógio.



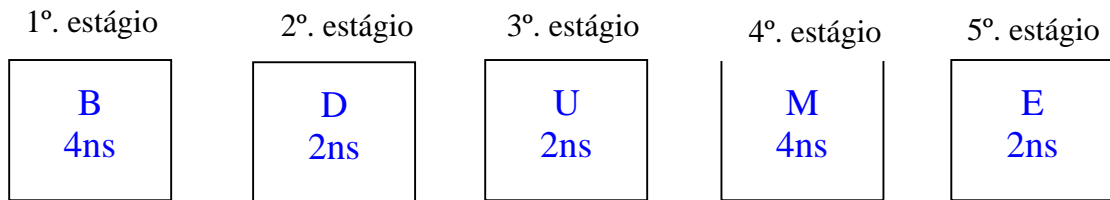
Ciclo de relógio para execução de uma instrução (sem pipeline) = $4ns + 2ns + 2ns + 4ns + 2ns = 14ns$

- b) (0,5) Calcule a duração do ciclo de relógio que a implementação pipeline com 2 estágios deve ter. Neste caso, qualquer estágio do pipeline deve poder ser realizado em um ciclo de relógio.



Ciclo de relógio será igual ao tempo para execução do estágio de maior tempo de execução = 10ns.

- c) (0,5) Calcule a duração do ciclo de relógio que a implementação pipeline com 5 estágios deve ter. Neste caso, qualquer estágio do pipeline deve poder ser realizado em um ciclo de relógio.



Ciclo de relógio será igual ao tempo para execução do estágio de maior tempo de execução = 4ns.

- d) (0,9) Calcule o tempo de execução de um programa no qual serão executadas 5 instruções para cada uma das três implementações. Considere que não existirá nenhum tipo de conflito para as implementações pipeline.

Seja T_{ex} = tempo de execução de uma instrução

= número de estágios x ciclo de relógio (determinado nos itens anteriores)

Caso 1 (sem pipeline) :

$$T_{ex} = 1 \text{ estágio} \times 14ns = 14ns$$

$$T_{total} = 5 \text{ instruções} \times T_{ex} = 70ns$$

Caso 2 (pipeline: 2 estágios) :

$$T_{ex} = 2 \text{ estágios} \times 10ns = 20ns$$

$$T_{total} = T_{ex} + 4 \times (\text{tempo de 1 estágio})$$

$$T_{total} = 20ns + 4 \times 10ns = 60ns$$

Caso 3 (pipeline: 5 estágios)

$$T_{ex} = 5 \text{ estágios} \times 4ns = 20ns$$

$$T_{total} = T_{ex} + 4 \times (\text{tempo de 1 estágio})$$

$$T_{total} = 20ns + 4 \times 4ns = 36ns$$