

## **GABARITO DA AD1 - Organização de Computadores 2010.2**

**Data de entrega 28/08/2010**

**Atenção:**

1. ADS enviadas pelo correio devem ser postadas cinco dias antes da data final de entrega estabelecida no calendário de entrega de ADs.
2. Como a avaliação a distância é individual, caso seja constatado que provas de alunos distintos sejam cópias umas das outras, independentemente de qualquer motivo, a todas será atribuída a nota ZERO. As soluções para as questões podem ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual.

1. (1,0) Um computador possui um RDM com 16 bits de tamanho e um REM com capacidade para armazenar números com 20 bits. Sabe-se que a célula deste computador armazena dados com 8 bits de tamanho e que ele possui uma quantidade N de células, igual à sua capacidade máxima de armazenamento. Pergunta-se:

- a) Qual é o tamanho do barramento de endereços?

*Barramento de endereços = REM = 20bits*

- b) Quantas células de memória são lidas em uma única operação de leitura?

*Célula = 8 bits*

*RDM = barramento de dados = 16 bits*

*Portanto, são lidas 2 células de memória durante uma única operação de leitura*

- c) Quantos bits têm a memória principal?

*$N = \text{quantidade de células} = \text{capacidade máxima} = 2^{20} = 1.048.576 \text{ células}$*

*Total de bits da memória =  $T = N \times M$*

*Como  $M = \text{quantidade de bits em uma célula, então } M = 8\text{bits}$*

*$T = N \times M = 2^{20} \times 2^3 = 2^{23} = 8.388.608 \text{ bits}$*

2. (1,0) Uma memória associativa por conjunto consiste em 64 quadros divididos em conjuntos de 4 quadros cada. A memória principal contém 4 M blocos de 128 palavras cada um. Mostre o formato de um endereço de MP.

*Memória Principal*

*$\Rightarrow B = 4 \text{ M blocos}$*

*$\Rightarrow K = 128 \text{ (considerando tamanho da palavra = tamanho da célula)}$*

*$\Rightarrow N = 4 \text{ M blocos} \times 128 \text{ células/bloco} = 512 \text{ M células}$*

*Memória Cache*

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

*$\Rightarrow Q = 64 \text{ quadro (ou linhas)}$*

*$\Rightarrow \text{Tamanho da memória cache} = 64 \text{ quadros} \times 128 \text{ células/bloco} = 8 \text{ K células}$*

*$\Rightarrow 1 \text{ conjunto} = 4 \text{ linhas (ou quadros)} \Rightarrow$*

*Total de conjuntos  $\Rightarrow C = 64 \text{ quadros} / 4 \Rightarrow C = 16 \text{ conjuntos}$*

Endereço da MP = 29 bits

Composição do endereço em função da memória cache

$$\Rightarrow \text{tag} = B / C = 4 \text{ M} / 16 = 256 \text{ K} = 2^{18} \Rightarrow \text{tag} = 18 \text{ bits}$$

$$\Rightarrow \text{n}^\circ \text{ do Conjunto: } C = 16 \text{ conjuntos} \Rightarrow 2^4 \Rightarrow 4 \text{ bits}$$

$$\Rightarrow \text{células por bloco: } 128 \text{ células por bloco} = 2^7 \Rightarrow 7 \text{ bits}$$

29 bits

Tag = 18 bits	No. Conjunto = 4 bits	Célula no bloco = 7 bits
---------------	--------------------------	-----------------------------

3. (1,0) Considere um sistema de computação que possui uma memória principal (RAM) com capacidade máxima de endereçamento de 64 K células, sendo que cada célula armazena um byte de informação. Para criar um sistema de controle e funcionamento da sua memória cache, a memória RAM é constituída de blocos de 8 bytes cada. A memória cache do sistema é do tipo mapeamento direto, contendo 32 linhas. Pergunta-se:

- a) Como seria organizado o endereço da MP (RAM) em termos de etiqueta (tag), número de linha e do byte dentro da linha?

**Memória Principal**

$\Rightarrow$  Tamanho da memória (em bytes) = 64K células, cada 1 célula armazena 1 byte, temos  $N = 64\text{K}$  células (ou 64 Kbytes)

$\Rightarrow$  Será organizada em blocos de 8bytes, como 1 célula = 1byte, temos cada bloco = 8 células,  $K = 8$

$\Rightarrow N = 64\text{K}$  células e  $K = 8$  células por bloco, o total de blocos da MP ( B ) será:

$$\text{Total de blocos: } B = N / K \Rightarrow B = 64\text{K células} / 8 \text{ células por bloco} \Rightarrow B = 8 \text{ K células}$$

**Memória Cache**

$\Rightarrow$  O K (quantidade de células/bloco) tem de ser igual a MP.

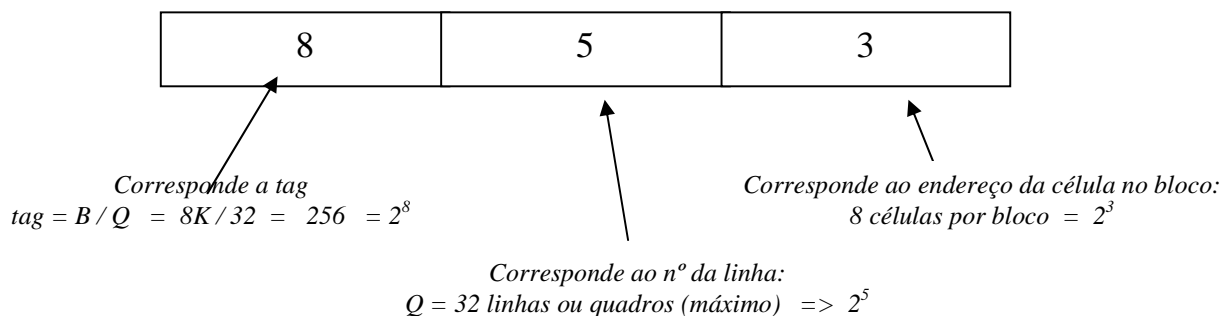
$\Rightarrow$  Tamanho da memória cache (em blocos ou linhas)  $\Rightarrow Q = 32$  linhas

$\Rightarrow$  Tamanho da memória cachê em células =  $Q \times K = 32 \text{ linhas} \times 8 \text{ células/linha} = 256 \text{ células}$  (ou 256 bytes)

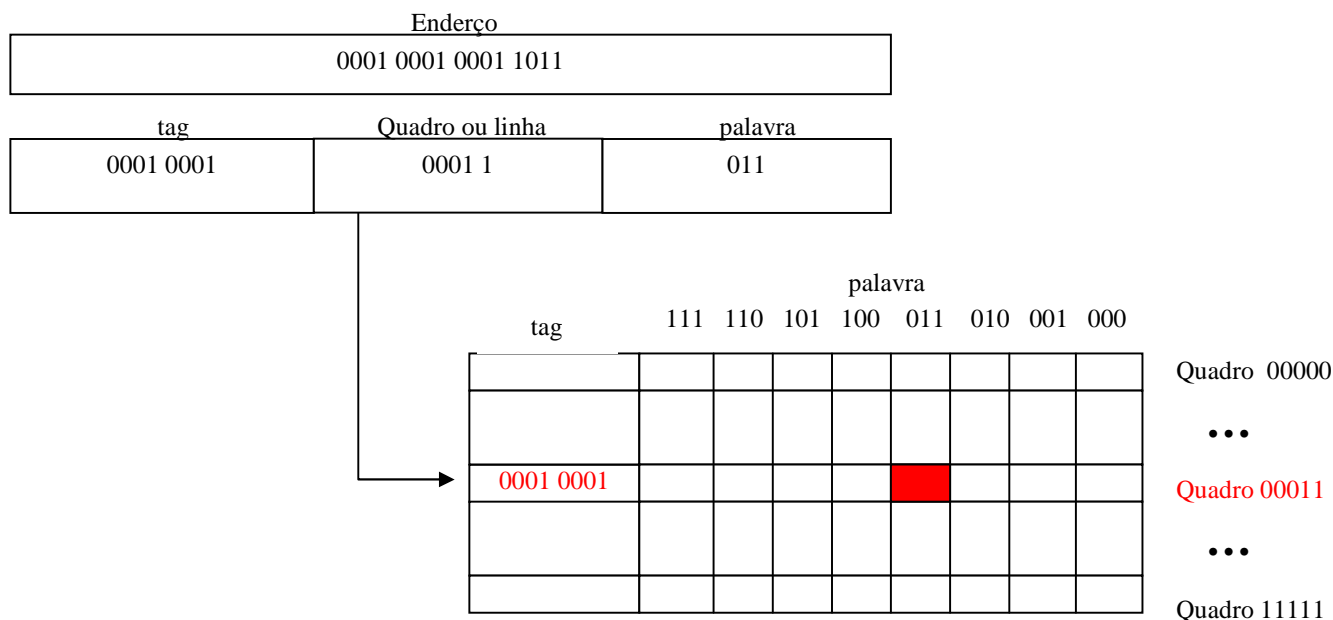
**Memória principal**

$\Rightarrow$  Para endereçarmos toda a MP precisamos da seguinte quantidade de bits ( E )

$$\text{sendo } N = 2^E \Rightarrow N = 64\text{K células} \Rightarrow N = 2^{16} \Rightarrow E = 16 \text{ bits}$$



**b) Em que linha estaria contido o byte armazenado no seguinte endereço da MP: 0001 0001 0001 1011?**



c) Qual é capacidade da memória cache em bytes?

*Capacidade da cache =  $Q \times K = 32 \text{ linhas} \times 8 \text{ palavras/linha}$ , considerando neste problema, tamanho da palavra = tamanho da célula e tamanho da célula = 1 byte, então:*

$$\text{Capacidade da cache} = 32 \text{ linhas} \times 8 \text{ bytes} = 256 \text{ bytes ou } 2^8 \text{ bytes}$$

4. (1,0) Explique as diversas memórias que constituem a hierarquia de memória comparando-as em termos de tempo de acesso, capacidade, e temporariedade de armazenamento de dados.

*O subsistema de memória é interligado de forma bem estruturada e organizado hierarquicamente em uma pirâmide com os níveis descritos a seguir.*

No topo da pirâmide teríamos os registradores, que são pequenas unidades de memória que armazenam dados na UCP. São dispositivos de maior velocidade com tempo de acesso em torno de 1 ciclo de memória, menor capacidade de armazenamento além de armazenar as informações por muito pouco tempo.

Em um nível abaixo teríamos a memória cache, cuja função é acelerar a velocidade de transferência das informações entre UCP e MP e, com isso, aumentar o desempenho do sistema. A UCP procura informações primeiro na Cache. Caso não as encontre, as mesmas são transferidas da MP para a Cache. A cache possui tempo de acesso menor que a da Memória principal, porém com capacidade inferior a esta, mas superior ao dos registradores e o suficiente para armazenar uma apreciável quantidade de informações, sendo o tempo de permanência do dado menor do que o tempo de duração do programa a que pertence.

Abaixo da memória cache teríamos a memória básica de um sistema de computação, que é a memória principal. Dispositivo onde o programa (e seus dados) que vai ser executado é armazenado para que a UCP busque instrução por instrução para executá-las. A MP são mais lentas que a cache e mais rápidas que a memória secundária, possui capacidade bem superior ao da cache e os dados ou instruções permanecem na MP enquanto durar a execução do programa.

Finalmente, na base da pirâmide teríamos a memória secundária, memória auxiliar ou memória de massa, que fornece garantia de armazenamento mais permanente aos dados e programas do usuário. Alguns dispositivos são diretamente ligados: disco rígido, outros são conectados quando necessário: disquetes, fitas de armazenamento, CD-ROM. São os mais lentos em comparação com os outros níveis de memória, mas possuem a maior capacidade de armazenamento e armazenam os dados de forma permanente.

5. (1,0) Considere uma máquina que possa endereçar 2 Gbytes de memória física, utilizando endereço referenciando byte, e que tenha a sua memória organizada em blocos de 4K bytes. Ela possui uma memória cache que pode armazenar 8K blocos, sendo um bloco por linha. Mostre o formato da memória cache, indicando os campos necessários (válido, tag, bloco) e o número de bits para cada campo, e o formato de um endereço da memória principal, indicando os bits que referenciam os campos da cache, para os seguintes mapeamentos:

a) Mapeamento direto.

a) Mapeamento direto.

**Memória Principal**

⇒ Tamanho da memória (em bytes) = 2Gbytes, como 1 célula referencia a 1 byte, temos  $N = 2\text{ G células}$

⇒ Será organizada em blocos de 4K bytes, como 1 célula = 1 byte, temos cada bloco = 4K células,  $K = 4K$

⇒ Sendo  $N$  o tamanho endereçável da memória e  $K$  que é a quantidade de células por blocos temos:

$N = 2\text{ G células}$  e  $K = 4K\text{ células / blocos}$  o total de blocos da MP (  $B$  ) será:

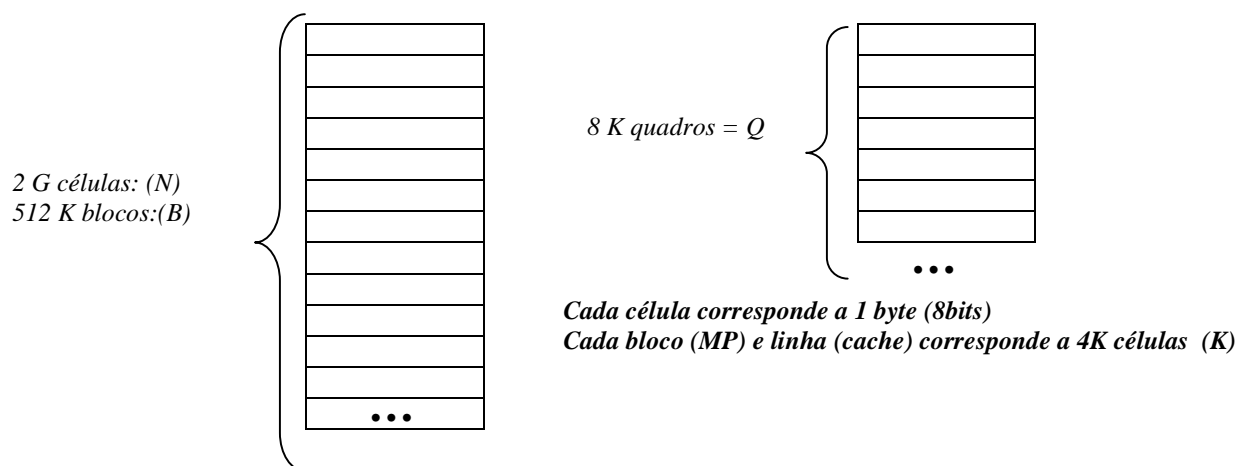
Total de blocos:  $B = N / K \Rightarrow B = 2\text{ G células} / 4\text{ K células/bloco} \Rightarrow B = 512\text{ K blocos}$

**Memória Cache**

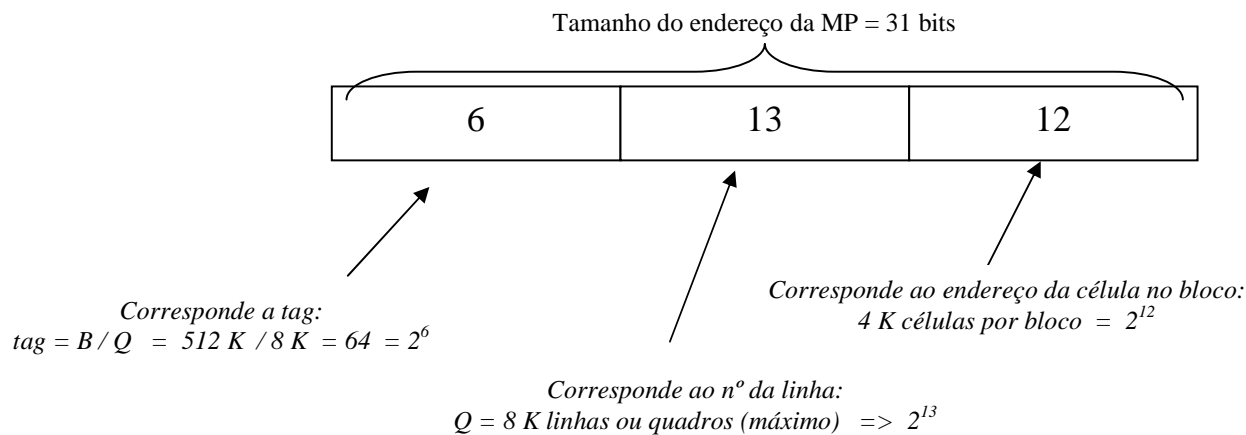
OBS: O  $K$  (quantidade de células/bloco) tem de ser igual a MP.

⇒ Tamanho da memória cachê em blocos = 8K blocos

⇒ Tamanho da memória cache em células = 8K blocos  $\times$  4 K células/bloco = 32 M células



Para endereçarmos toda a MP precisamos da seguinte quantidade de bits (  $E$  )  
sendo  $N = 2^E \Rightarrow N = 2\text{G células} \Rightarrow N = 2^{31} \Rightarrow E = 31\text{ bits}$



**b) Mapeamento totalmente associativo.**

**Memória Principal**

=>  $N = 2G$  células

=>  $K = 4K$  células por bloco

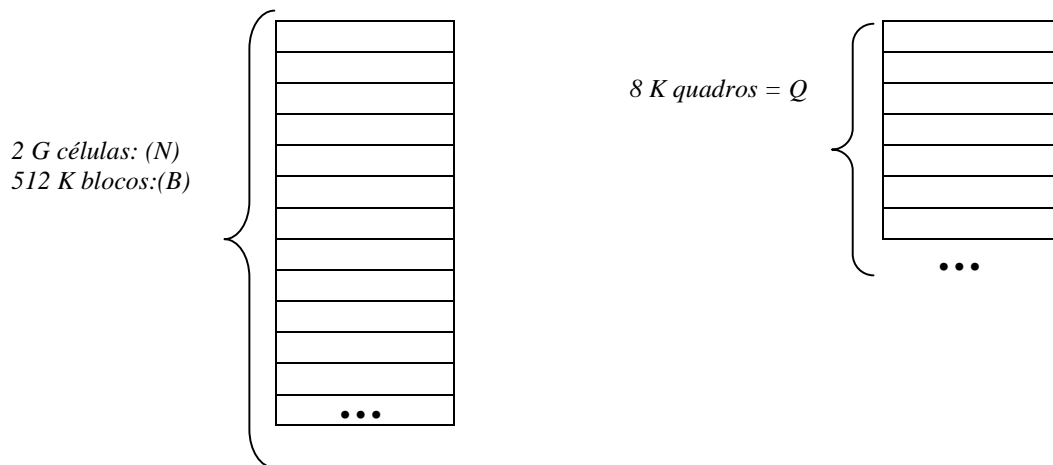
=>  $B = 512 K$  blocos

**Memória Cache**

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

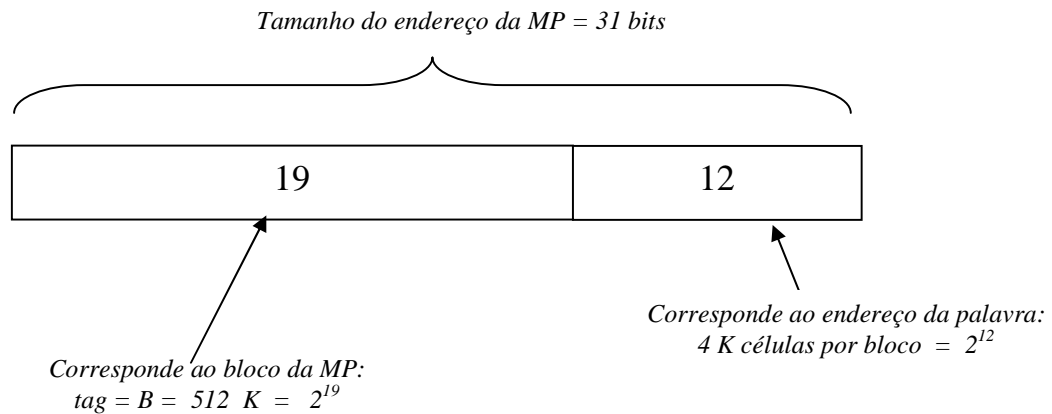
=>  $Q = 8 K$  blocos

=> Tamanho da memória cache =  $32 M$  células



Para endereçarmos toda a MP precisamos da seguinte quantidade de bits:  $E = 31$  bits

Como o bloco pode ser alocado em qualquer posição da memória cache a tag indicará qual dos blocos da MP está alocado naquela posição da memória cache



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

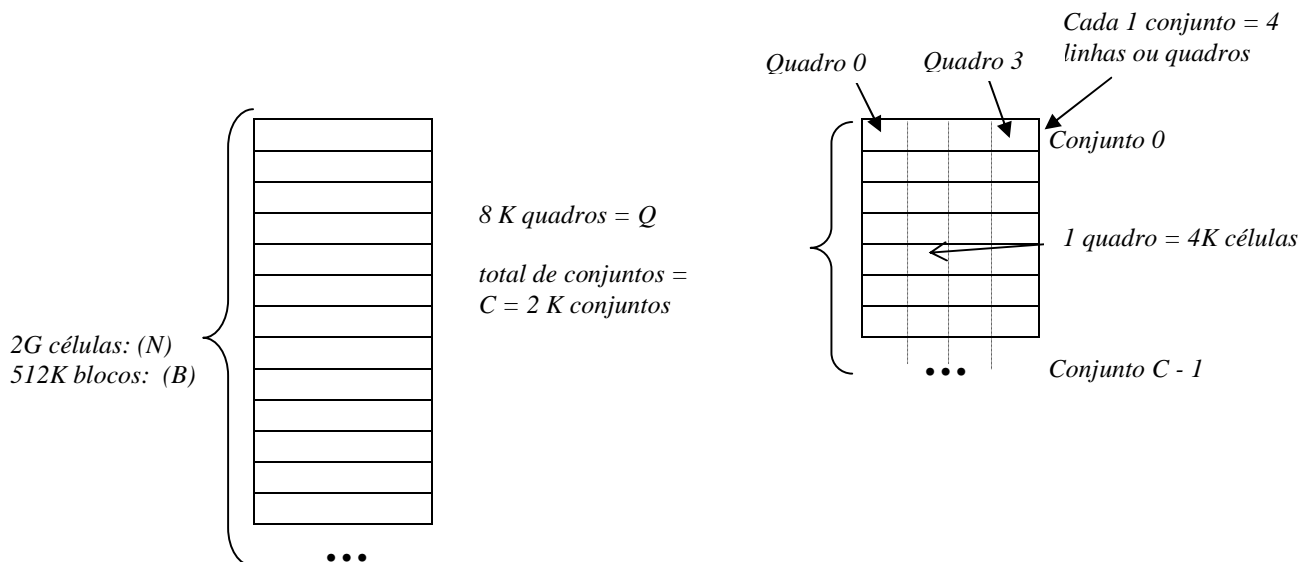
#### Memória Principal

=>  $N = 2\text{ G células}$   
=>  $K = 4\text{ K células}$   
=>  $B = 512\text{ K blocos}$

#### Memória Cache

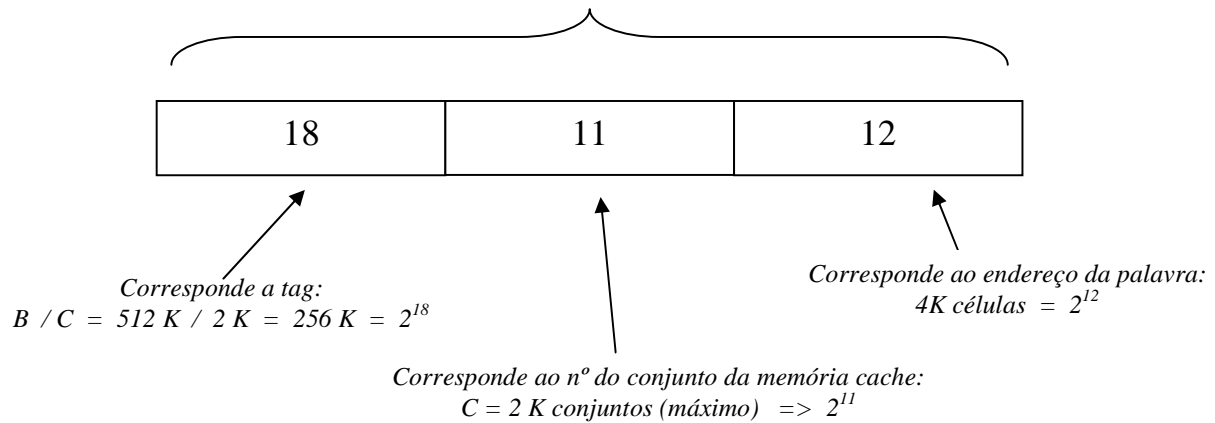
OBS:  $O$   $K$  (quantidade de células/bloco) tem de ser igual a MP.

=>  $Q = 8\text{ K blocos}$   
=> Tamanho da memória cache =  $32\text{ M células}$   
=> 1 conjunto = 4 linhas (ou quadros) =>  
Total de conjuntos =>  $C = 8\text{ K células} / 4 \Rightarrow C = 2\text{ K conjuntos}$



Para endereçarmos toda a MP precisamos da seguinte quantidade de bits:  $E = 31$  bits

Tamanho do endereço da MP = 31 bits



6. (1,0) Considere a máquina apresentada na aula 4. Descreva detalhadamente (do mesmo modo que é apresentado na aula 4) como é realizada a execução das seguintes instruções:

a) STR 20

1.  $RI \leftarrow$  Instrução lida
2.  $CI \leftarrow CI + 1$
3. Decodificação do código de operação
4. A Unidade de Controle (UC) emite sinais para que o valor do campo operando = 20 seja transferido para o REM
5. A UC emite sinais para que o valor do registrador acumulador (ACC) seja transferido para o RDM
6. A UC ativa a linha WRITE do barramento de controle
7. O conteúdo do RDM é transferido, através do barramento de dados, para o endereço 20 da memória, endereço este transferido para a memória do REM a partir do barramento de endereços.

b) JN 50

1.  $RI \leftarrow$  Instrução lida
2.  $CI \leftarrow CI + 1$
3. Decodificação do código de operação
4. A UC emite sinal para transferir conteúdo do acumulador para UAL  
-  $UAL \leftarrow ACC$
5. UAL executa operação de comparação  
5.1 Se conteúdo de  $ACC < 0$   
 $CI \leftarrow 50$

7. (1,0) Escreva um programa que utilize as instruções de linguagem de montagem apresentadas na aula 4 para executar o seguinte procedimento. O conteúdo da memória cujo endereço é 10 é lido e verifica-se se o seu valor é igual a 0. Caso seu valor seja igual a 0, o conteúdo de memória cujo endereço é 20 é somado com o conteúdo de memória cujo endereço é 16 e o resultado é armazenado no endereço 30. Caso contrário, o conteúdo de memória cujo endereço é 20 é subtraído do conteúdo de memória cujo endereço é 16 e o resultado é armazenado no endereço 30.

Além de apresentar seu programa escrito em linguagem de montagem, apresente também o programa traduzido para linguagem de máquina. Os endereços estão expressos em decimal e portanto devem ser convertidos para a base 2 utilizando 8 bits, quando o programa for convertido para linguagem de máquina.

<b>Endereço</b>	<b>Instrução</b>	<b>Descrição</b>	<b>Linguagem Máquina (bin / hexa)</b>
00	LDA 10	ACC <- (10)	( 000100001010 / 10A )
01	JZ 06	CI <- 06 se ACC=0	( 010100000110 / 506 )
02	LDA 16	ACC <- (16)	( 000100010000 / 110 )
03	SUB 20	ACC <- ACC - (20)	( 010000010100 / 414 )
04	STR 30	(30) <- ACC	( 001000011110 / 21E )
05	HLT	Encerra	( 000000000000 / 000 )
06	LDA 20	ACC <- (20)	( 000100010100 / 114 )
07	ADD 16	ACC <- ACC + (16)	( 001100010000 / 310 )
08	STR 30	(30) <- ACC	( 001000011110 / 21E )
09	HLT	Encerra	( 000000000000 / 000 )

Outra solução,

<b>Endereço</b>	<b>Instrução</b>	<b>Descrição</b>	<b>Linguagem Máquina (bin / hexa)</b>
00	LDA 10	ACC <- (10)	( 000100001010 / 10A )
01	JZ 05	CI <- 05 se ACC=0	( 010100000101 / 505 )
02	LDA 16	ACC <- (16)	( 000100010000 / 110 )
03	SUB 20	ACC <- ACC - (20)	( 010000010100 / 414 )
04	JMP 07	CI <- 07	( 100000000111 / 807 )
05	LDA 20	ACC <- (20)	( 000100010100 / 114 )
06	ADD 16	ACC <- ACC + (16)	( 001100010000 / 310 )
07	STR 30	(30) <- ACC	( 001000011110 / 21E )
08	HLT	Encerra	( 000000000000 / 000 )

8. (1,0) Considere uma máquina com arquitetura semelhante àquela apresentada em aula. Pode-se endereçar no máximo 256 M células de memória, sendo que cada célula tem tamanho igual a 8 bits. Em cada acesso à memória, obtém-se o conteúdo de uma célula. Todas as instruções desta máquina possuem dois campos: o primeiro indica o código de operação e o segundo indica endereço de célula de memória onde se encontra o operando. Esta máquina possui 15 códigos de operação diferentes.

- a) Calcule a capacidade mínima de endereçamento em bits do REM, considerando que os bits armazenados no REM são utilizados para endereçar uma célula de memória.

$$REM = E = \text{tamanho em bits necessários para acessar toda a memória (N)}$$

$$N = 2^E = 256M \text{ células} = 2^{28} \text{ células} \Rightarrow E = 28 \Rightarrow REM = 28 \text{ bits}$$

- b) Calcule o número de bits que devem poder ser transmitidos no barramento de endereços em cada acesso à memória.

$$\text{Barramento de endereços} = REM = 28 \text{ bits}$$

- c) Calcule o tamanho do RI (Registrador de Instruções).

$$\text{Tamanho mínimo de RI} = \text{tamanho da instrução}$$

$$\text{Tamanho da instrução} = \text{código de operação} + 1 \text{ operando}$$

$$\text{Tamanho da instrução} = (\text{tamanho p/ endereçar 15 instruções}) + 1 \times (\text{endereço de memória})$$

$$\text{Tamanho da instrução} = 4 \text{ bits} + 1 \times 28 \text{ bits} = 32 \text{ bits}$$

$$\text{Tamanho mínimo de RI} = 32 \text{ bits}$$

- d) Calcule o número de células que uma instrução ocupa.

$$\text{Tamanho mínimo de 1 instrução} = 32 \text{ bits}$$

$$\text{Tamanho de 1 célula} = 8 \text{ bits}$$

$$\text{Serão necessárias pelo menos 4 células para uma instrução}$$



- e) Calcule a capacidade máxima de armazenamento da memória deste sistema em bits.

$$T = N \times M \Rightarrow T = 256 \text{ Mcélulas} \times 8 \text{ bits/célula} \Rightarrow T = 2048 \text{ Mbits}$$

- f) Calcule a capacidade mínima de endereçamento em bits do CI (Contador de Instrução), considerando que os bits armazenados no CI são utilizados para endereçar a primeira célula de uma instrução armazenada na memória.

$$CI = \text{tamanho necessário para endereçar uma célula da memória} = 28 \text{ bits}$$

9. (1,0) Considere uma máquina cujo relógio possui uma frequência de 1 GHz e um programa P1 no qual são executadas 1000 instruções.

- a) Calcule o tempo para executar o programa P1, considerando que cada instrução é executada em 10 ciclos de relógio e a execução de uma instrução só se inicia quando a execução da instrução anterior é finalizada.

$$\text{Tempo de um ciclo de relógio} = 1/1.000.000.000 = 0,000\ 000\ 001 \text{ seg ou } 1 \text{ ns (nanosegundos)}$$

$$\text{Tempo de execução de 1 instrução} = 10 \text{ ciclo de relógio} = 10 \times 1 \text{ ns} = 10 \text{ ns}$$

$$1000 \text{ instruções executadas seqüencialmente} = 1000 \times 10 \text{ ns} = 10.000 \text{ ns}$$

- b) Uma nova implementação dessa máquina utiliza um pipeline de 2 estágios, todos de duração igual a 6 ciclos de relógio. Calcule o tempo para executar uma instrução nesta nova máquina e o tempo para executar o programa P1, considerando que não existem conflitos de qualquer tipo.

$$\text{Tempo para um estágio} = 6 \text{ ciclos de relógio} = 6 \times 1 \text{ ns} = 6 \text{ ns}$$

$$\text{Para execução da 1ª instrução} = 2 \text{ estágios} \times 6 \text{ ns} = 12 \text{ ns}$$

$$\text{Para execução das instruções posteriores} = \text{tempo de 1 estágio devido ao pipeline} = 6 \text{ ns}$$

$$\text{Tempo total para execução das 1000 instruções} = 12 \text{ ns} + 999 \times 6 \text{ ns} = 6006 \text{ ns}$$

- c) Uma nova implementação dessa máquina utiliza um pipeline de 5 estágios, todos de duração igual a 3 ciclos de relógio. Calcule o tempo para executar uma instrução nesta nova máquina e tempo para executar o programa P1, considerando que não existem conflitos de qualquer tipo.

$$\text{Tempo para um estágio} = 3 \text{ ciclos de relógio} = 3 \times 1 \text{ ns} = 3 \text{ ns}$$

$$\text{Para execução da 1ª instrução} = 5 \text{ estágios} \times 3 \text{ ns} = 15 \text{ ns}$$

$$\text{Para execução das instruções posteriores} = \text{tempo de 1 estágio devido ao pipeline} = 3 \text{ ns}$$

$$\text{Tempo total para execução das 1000 instruções} = 15 \text{ ns} + 999 \times 3 \text{ ns} = 3012 \text{ ns}$$

- d) Indique qual das 3 máquinas executa uma instrução em menos tempo e qual das três executa o programa P1 em menos tempo.

A arquitetura do item a executa uma instrução em menos tempo. Já a arquitetura do item c executa o programa P1 em um tempo total menor.

10. (1,0) Explique como funciona uma unidade de controle microprogramada e indique uma vantagem e uma desvantagem em relação a uma unidade de controle implementada em hardware.

A unidade de controle microprogramada é utilizada para se desenvolver a implementação de um conjunto de instruções que apresenta muita complexidade para ser implementado somente em hardware. A execução de uma instrução é composta da execução de microinstruções referentes a ela e a unidade de controle microprogramada é projetada de modo a executar estas microinstruções. Ela é composta por: Memória de controle, Contador de microprograma e Seqüenciador.

*A Memória de controle armazena as microinstruções que compõem uma instrução e o Contador de microprograma armazena a localização da próxima microinstrução a ser executada. O Seqüenciador é o componente que controla a sequência de execução das microinstruções, informando o local da próxima microinstrução que deve ser executada e armazenada no Contador de microprograma.*

*Em uma unidade controlada por hardware, o seu desenvolvimento 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.*

*A vantagem de se utilizar uma unidade de controle microprogramada é a maior facilidade do seu projeto, da sua implementação e da verificação de erros, pois o número de expressões booleanas pode ser muito grande, o que pode tornar difícil a implementação por hardware de um conjunto de instruções que possua um grande número de instruções.*

*A desvantagem de uma unidade de controle microprogramada é ser mais lenta que uma unidade de controle implementada por hardware.*