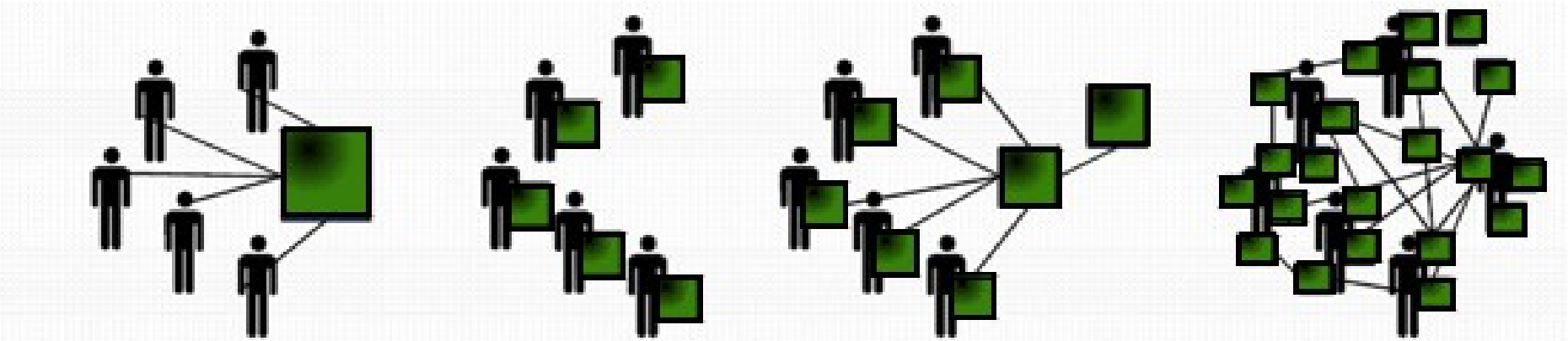
Disciplina
Sistemas de Computação

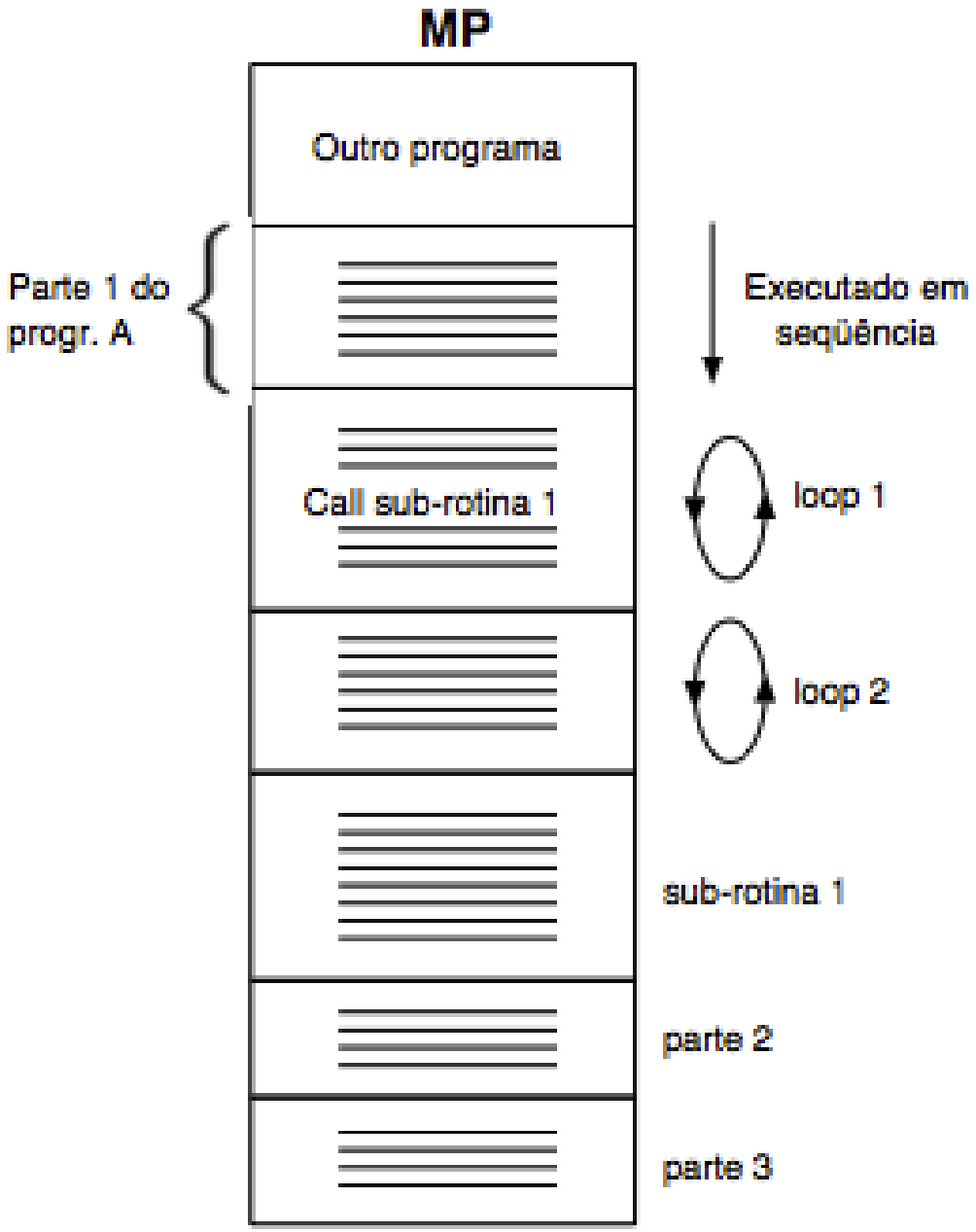
Aula 06



Memória Cache
Conceitos de Localidade

- A execução de programas se realiza, na média, em pequenos grupos de instruções.
  - Possibilita uso do conceito de localidade
- Duas abordagens:
  - Localidade Espacial: dado que o programa acessou um palavra de memória, tem grande probabilidade de acessar a palavra (ou endereço) subjacente
  - Localidade Temporal: dado que o programa acessou uma palavra de memória, tem grande probabilidade de acessar em breve a mesma palavra novamente

Memória Cache
Conceitos de Localidade

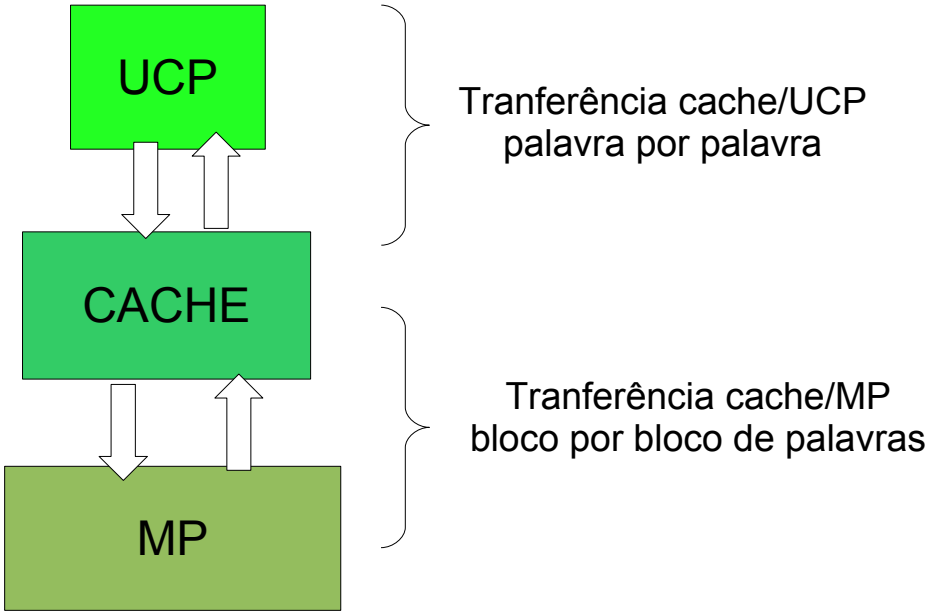


Como é utilizada a Memória Cache?

- Sempre que a UCP vai buscar uma nova instrução (ou dado), ela acessa a memória cache
- Se a instrução estiver na cache, chama-se de acerto (ou cache hit)
  - Ela é transferida em alta velocidade
- Se a instrução não estiver na cache, chama-se de falta (ou cache miss).

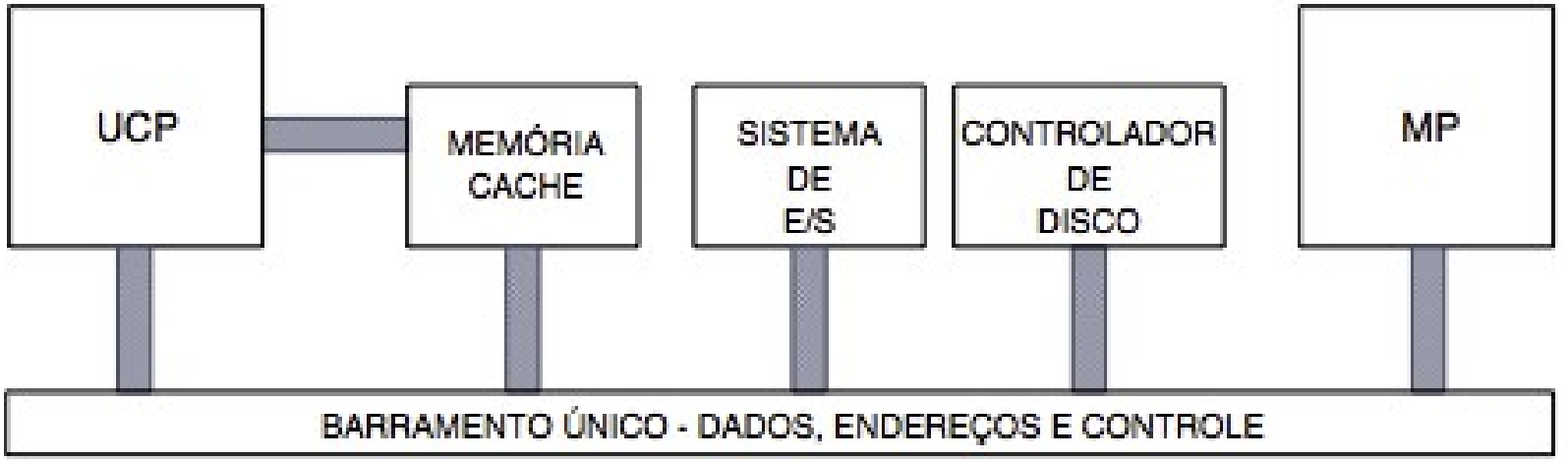
Utilização da Memória Cache

- O grupo de instruções a que a instrução pertence é transferida da MP para a cache, considerando o princípio da localidade



Utilização da Memória Cache

- Para melhorar o desempenho é necessário muito mais acertos do que faltas



Tipos de Memória Cache

- Dois tipos básicos de emprego de cache:
  - Na relação UCP/MP (cache de RAM)
  - Na relação MP/disco (cache de disco)
  - Funciona segundo o mesmo princípio da cache de memória RAM porém em vez de utilizar a memória de alta velocidade SRAM para servir de cache, o sistema usa uma parte da memória principal, DRAM como se fosse um espaço em disco

Níveis de Cache da Memória RAM

- Para não aumentar muito o custo da cache, conforme o aumento da sua capacidade: sistema hierárquico de caches
  - Nível 1 ou L1 sempre localizada no interior do processador
  - Nível 2 ou L2 localizada em geral na placa mãe, externa ao processador
  - Nível 3 ou L3 existente em poucos processadores, localizada externamente ao processador

- Cache também pode ser dividida: dados e instrução

Elementos de Projeto da Memória Cache

- Definição do tamanho das memórias cache (L1, L2 e L3)
- Função de mapeamento de dados MP/cache
- Algoritmos de substituição de dados na cache
- Política de escrita pela cache

Elementos de Projeto da Memória Cache

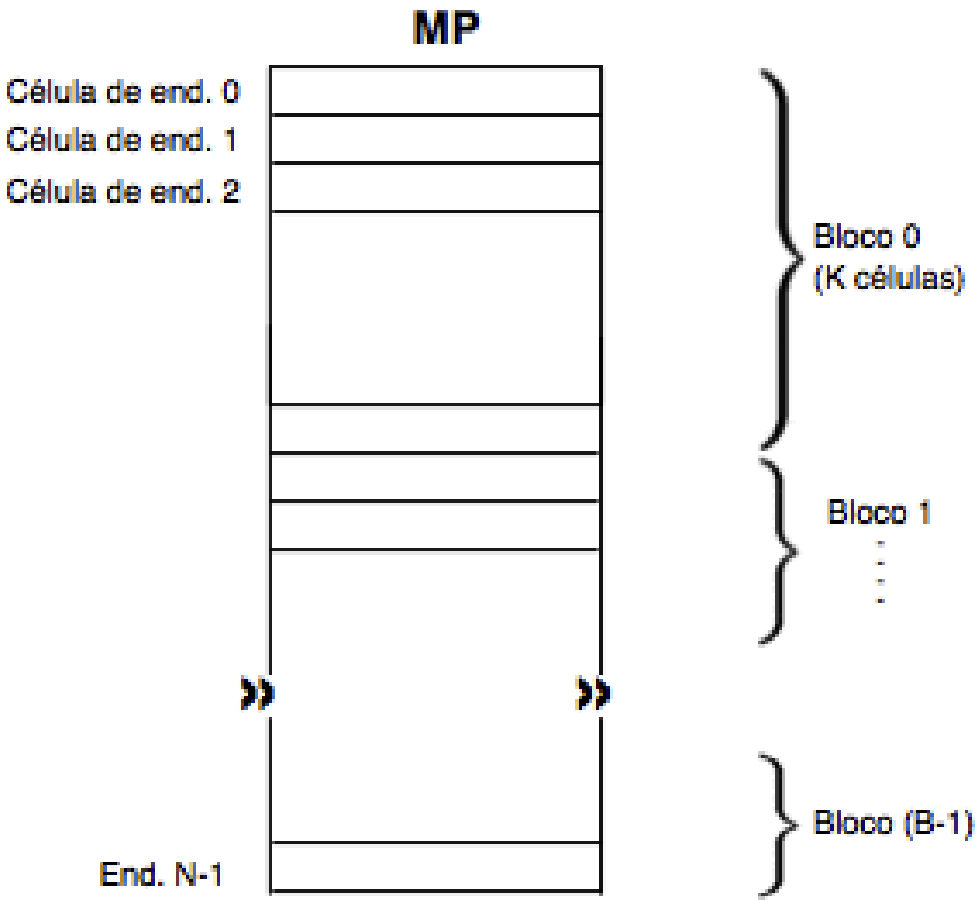
- Tamanho da Memória Cache (fatores):
  - Tamanho da memória principal
  - Relação acertos/faltas
  - Tempo de acesso da MP
  - Custo médio por bit, da MP, e da memória cache L1, L2 e L3
  - Tempo de acesso da cache L1, L2 e L3
  - Natureza do programa em execução (princípio da localidade)

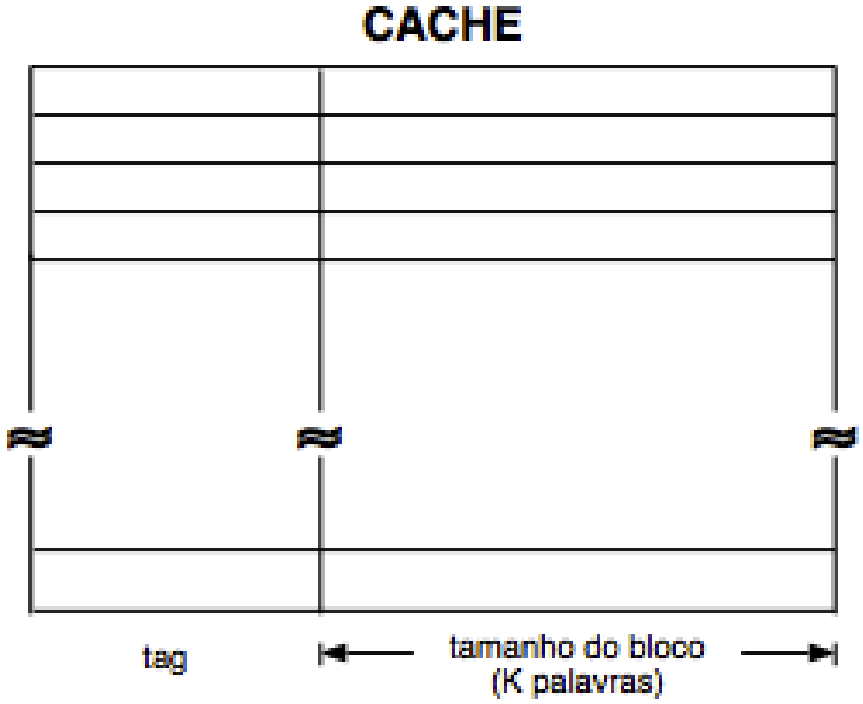
Elementos de Projeto da Memória Cache

- Mapeamento de dados MP/Cache:
  - A memória RAM está dividida em conjuntos de B blocos, cada um com K células e a cache com Q linhas, cada uma com K células.
  - Q é muito menor do que B
  - Para garantir acerto de 90% a 95% - conceito da localidade

Elementos de Projeto da Memória Cache

- Mapeamento de dados MP/Cache:



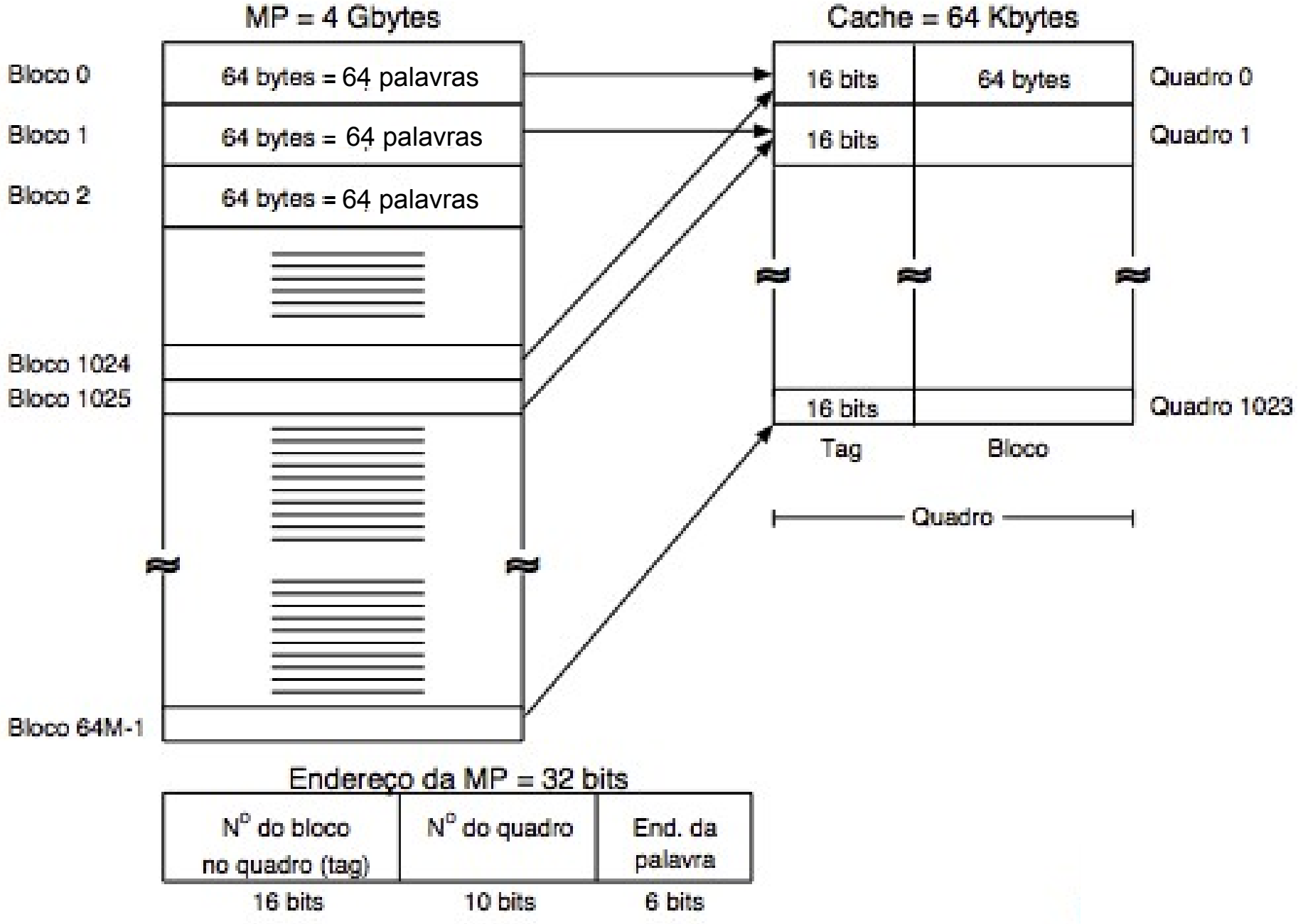


- MP possui N palavras (células).
- MP dividida em B blocos com K palavras cada.
- Cache possui Q quadros (linhas de dados), cada uma contendo K palavras = tamanho do bloco da MP.
- Cada quadro possui um campo(campo TAG) que contém o número de identificação do bloco que está nele armazenado.

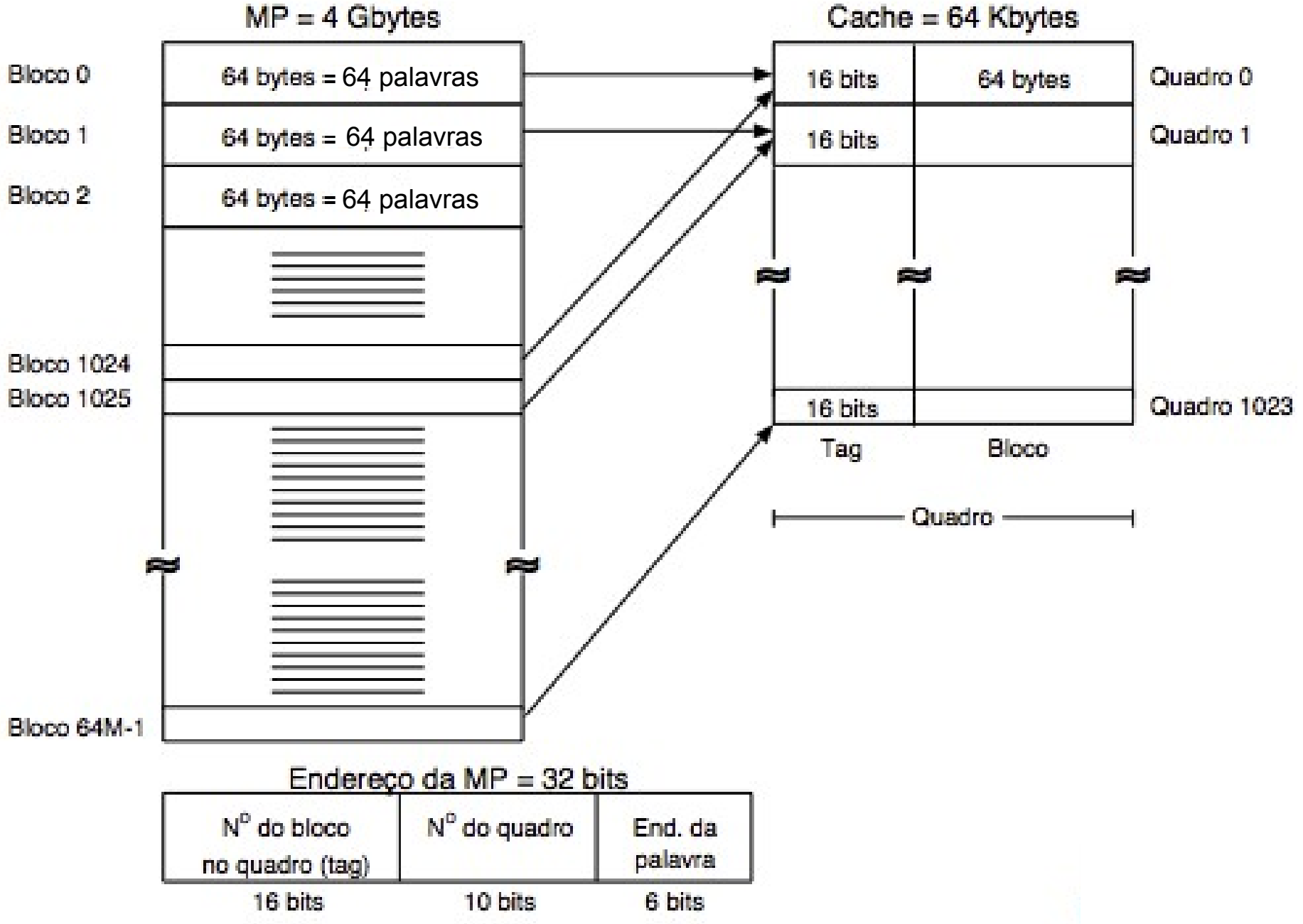
Elementos de Projeto da Memória Cache

- Para efetuar a transferência de um bloco da MP para uma específica linha da memória cache, escolhe-se um das 3 alternativas:
  - Mapeamento Direto
  - Mapeamento Associativo
  - Mapeamento Associativo por conjuntos

Mapeamento Direto



Mapeamento Direto



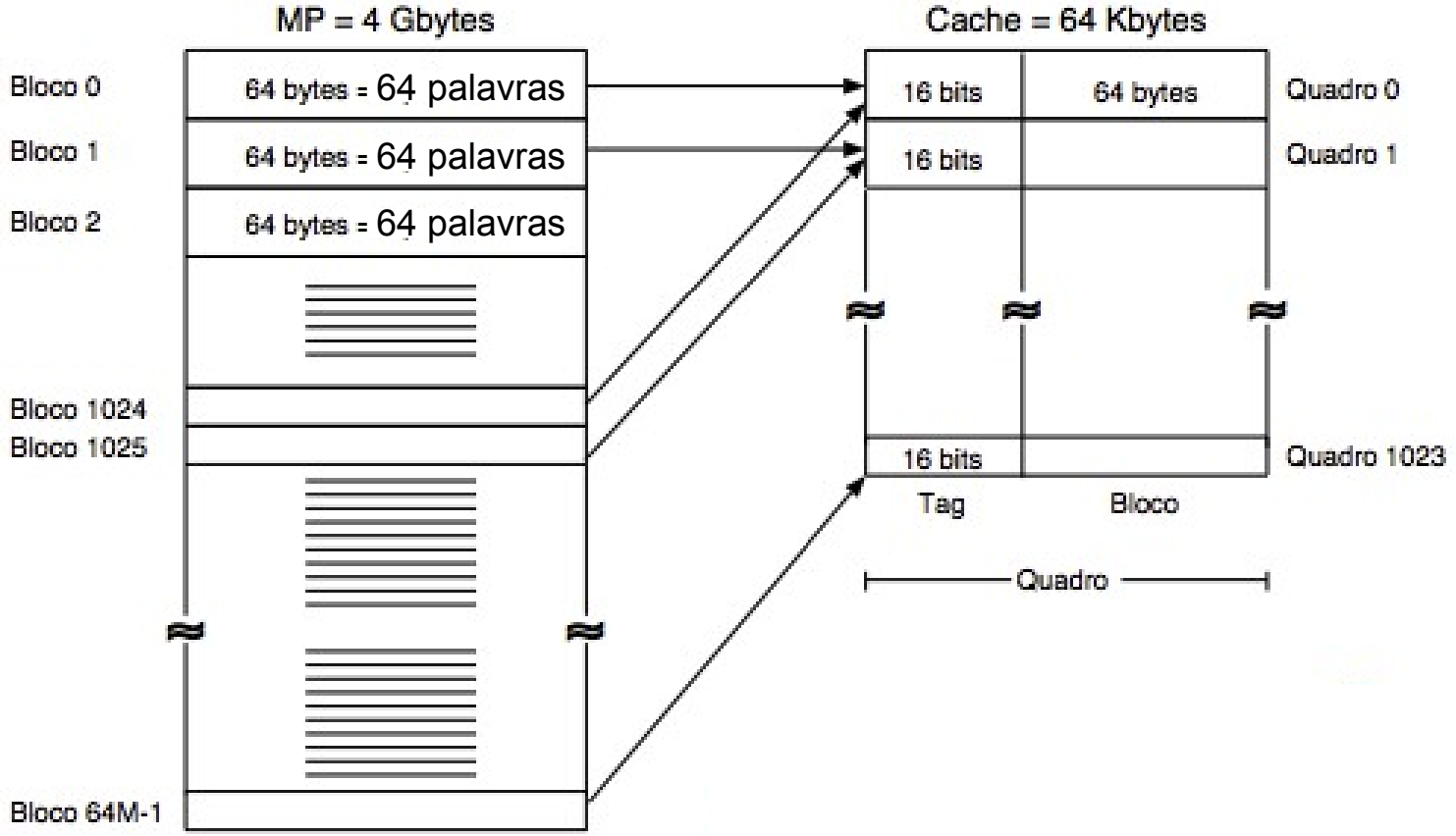
Mapeamento Direto

- Cada bloco da MP tem uma linha de cache
- Como há mais blocos do que linhas de cache, muitos blocos vão ser destinados a uma mesma linha

Mapeamento Direto

- Exemplo:

  MP com 4G palavras (N) e endereços de 32 bits
  Cache com 64Kbytes, 1024 linhas (Q) com 64 bytes de dados cada uma (K)
  Número de blocos da MP = B = N/K = 4G/64= 64 M blocos
  Para localizar um endereço de MP (end) em Cache (EQ): EQ= end módulo 1024



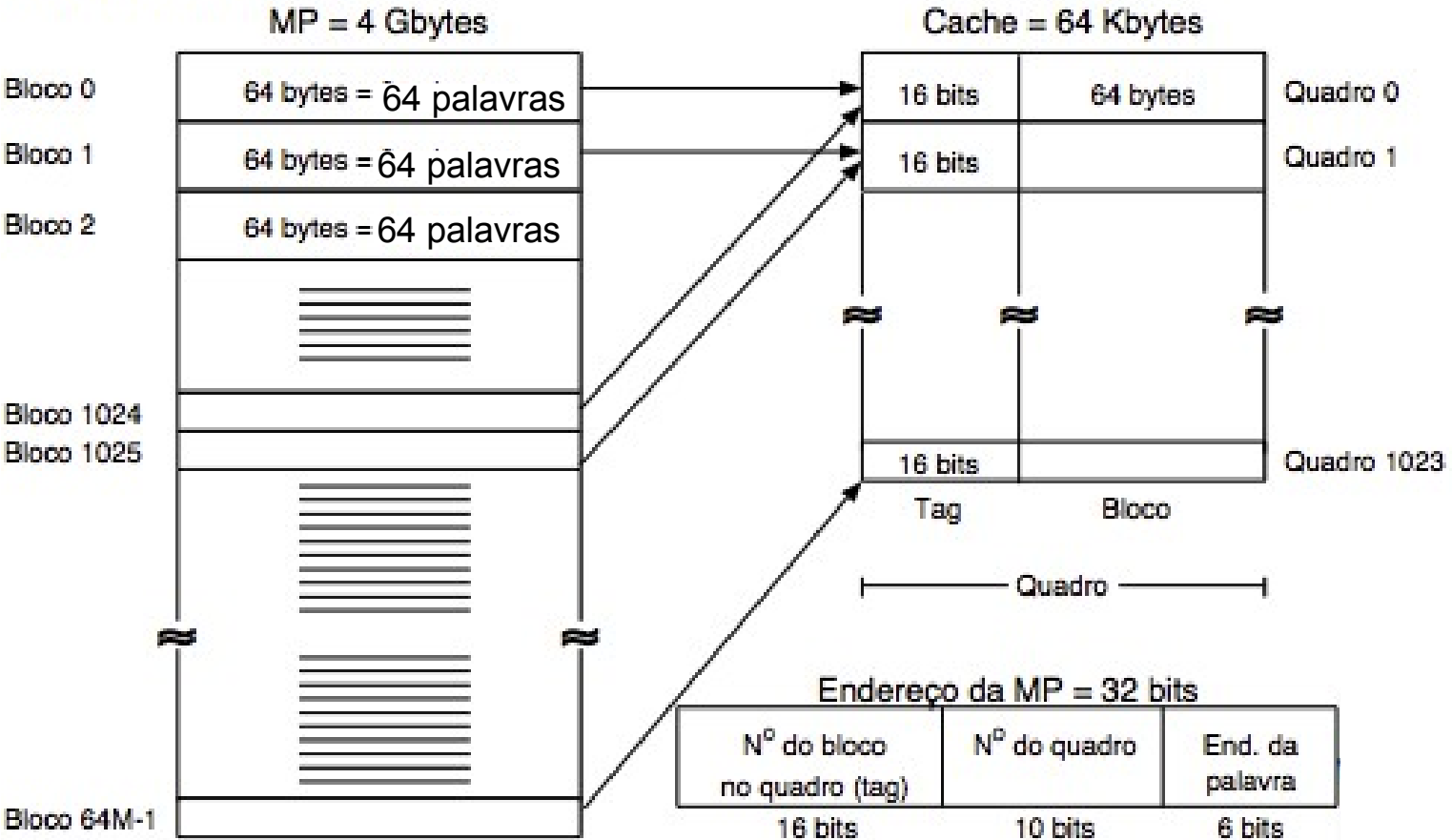
Mapeamento Direto

- Cada endereço da memória pode ser dividido da seguinte forma:

6 bits menos significativos: indicam a palavra (2^6= 64 palavras no bloco B e na linha Q)

10 bits do meio: indicam o endereço da linha da cache (2^{10} = 1024 linhas)

16 bits mais significativos: qual o bloco dentre os 64 K blocos que podem ser alocados na linha



Mapeamento Direto

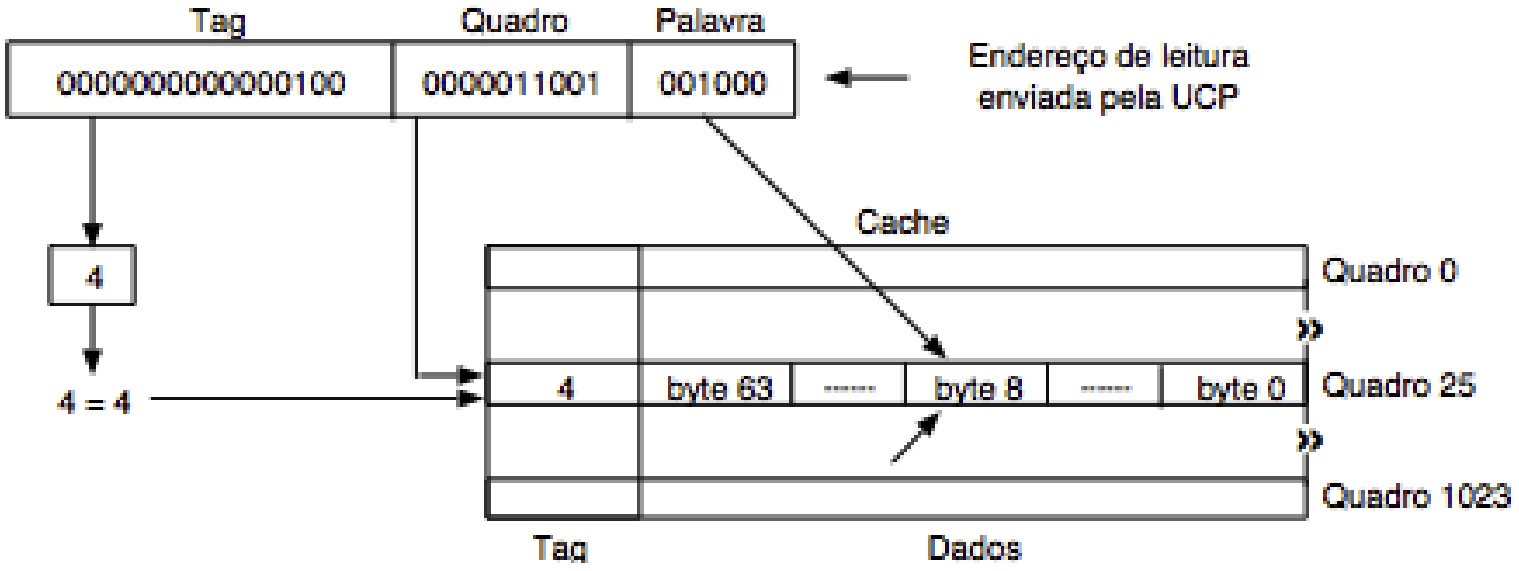
- Exemplo:

UCP apresenta endereço de 32 bits ao circuito da cache: 00000000000010000000011001001000

Parte 1: 0000000000000100 (comparado com o tag do quadro 25 da cache)

Parte 2: 0000011001 (quadro 25)

Parte 3: 001000 (palavra 8 é acessada)



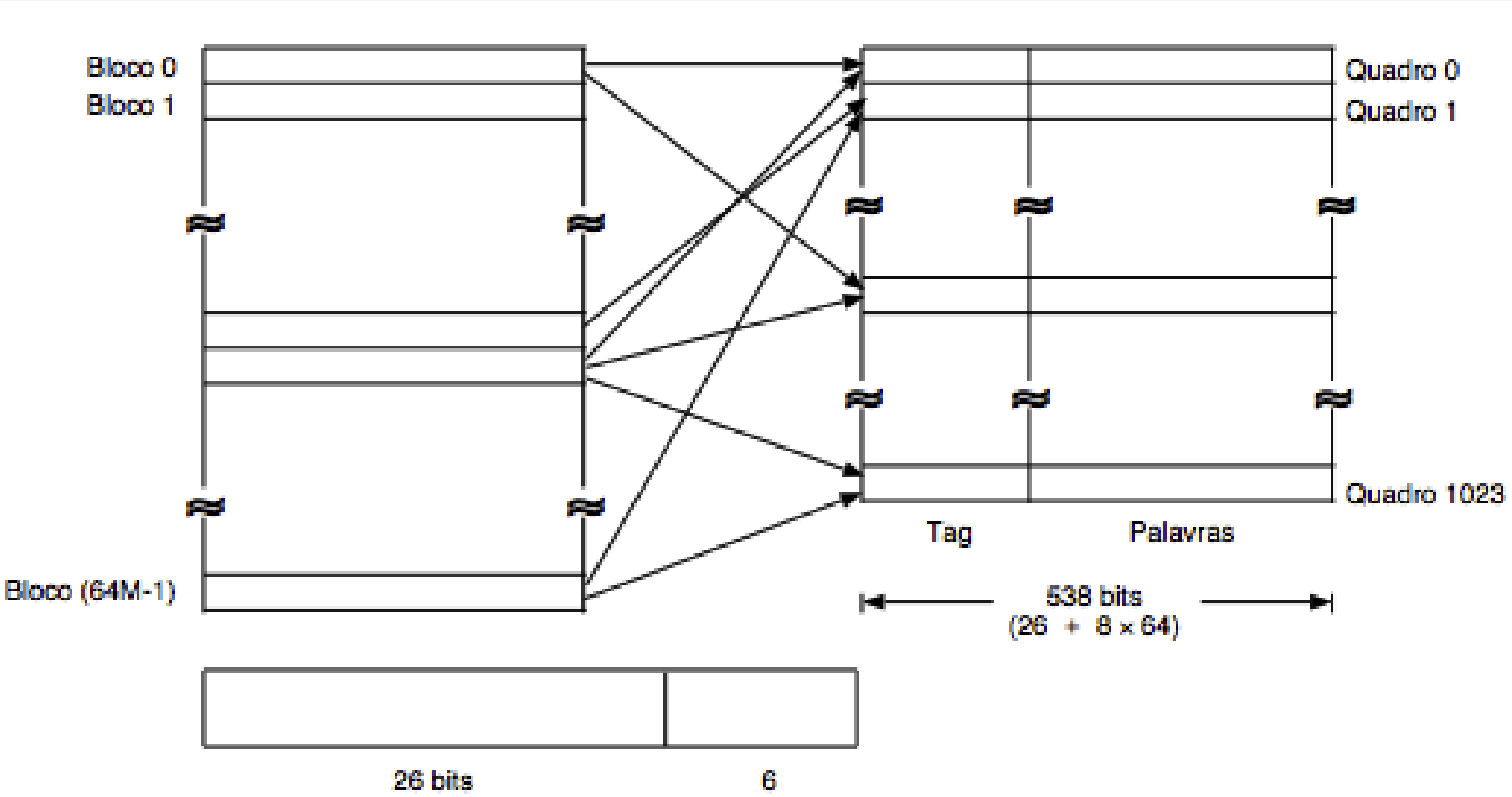
Mapeamento Direto

- Considerações:
  - simples, de baixo custo, não acarreta sensíveis atrasos de processamento de endereços
  - Problema: fixação da localização para os blocos (65.536 blocos destinados a uma linha)
- Se durante a execução houver repetidas referências a palavras situadas em blocos alocados na mesma linha: muitas substituições de blocos

Mapeamento Associativo

- Os blocos não têm uma linha fixada previamente para seu armazenamento
- Se for verificado que o bloco não está armazenado na cache, este será transferido, substituindo um bloco já armazenado
- Endereço da MP é dividido em duas partes:
- 6 bits menos significativos: palavra desejada
- 26 bits restantes: endereço do bloco desejado

Mapeamento Associativo



- Sempre que a UCP realizar um acesso, o controlador da cache deve examinar e comparar os 26 bits de endereço do bloco com o valor dos 26 bits do campo de tag de todas as 1024 linhas.

Mapeamento Associativo

- Considerações:
  - Evita a fixação de blocos às linhas
  - Necessidade de uma lógica complexa para examinar cada campo de tag de todas as linhas de cache

Mapeamento Associativo por Conjuntos

- Compromisso entre as duas técnicas anteriores: tentar resolver o problema de conflito de blocos e da busca exaustiva e comparação do campo tag
- Organiza as linhas da cache em grupos, denominados conjuntos
- Nos conjuntos, as linhas são associativas

Mapeamento Associativo por Conjuntos

- A cache é dividida em C conjuntos de D quadros:
  - Quantidade de quadros Q = C x D
  - Endereço da linha no conjunto K = E módulo C

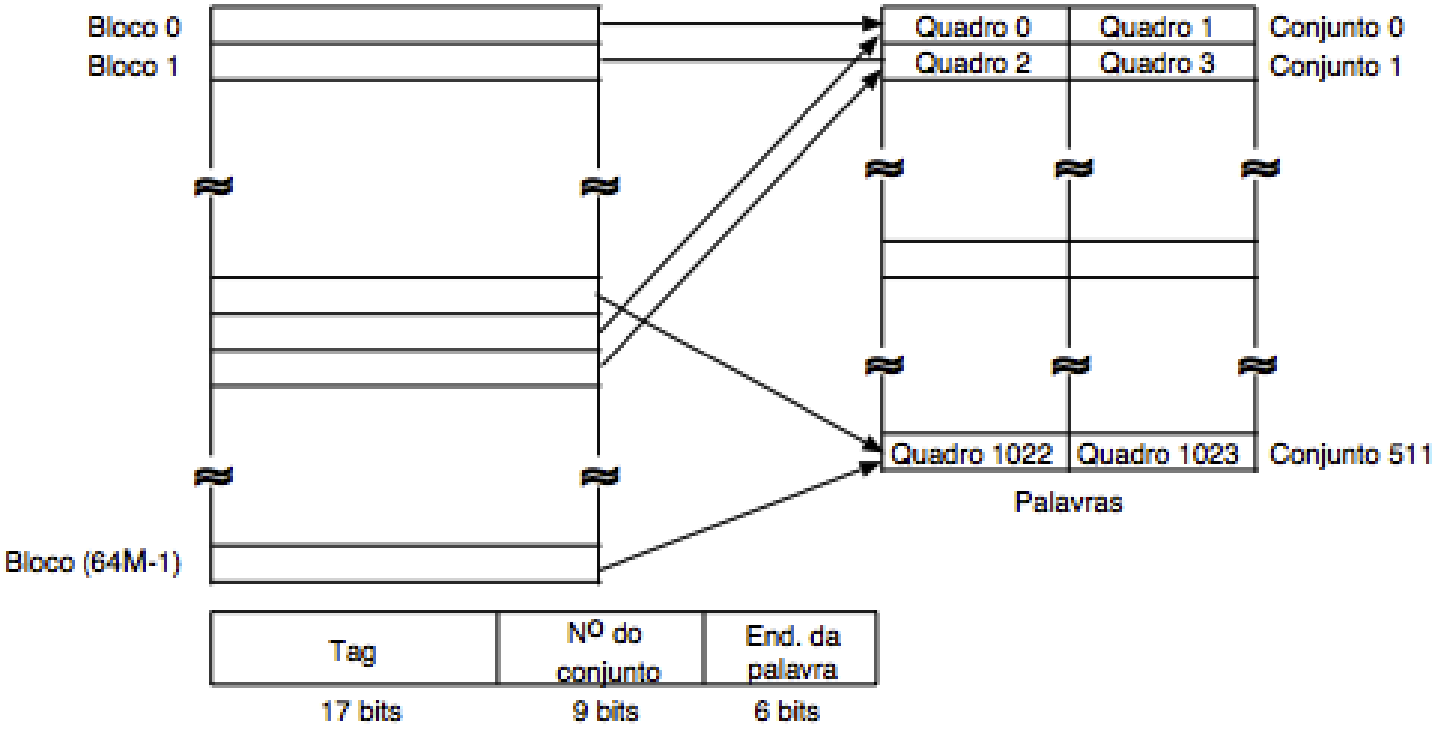
Mapeamento Associativo por Conjuntos

- O algoritmo estabelece que o endereço da MP é dividido da seguinte forma:

Tag	Número do conjunto	Endereço da palavra
17 bits	9 bits	6 bits

Mapeamento Associativo por Conjuntos

- O algoritmo estabelece que o endereço da MP é dividido da seguinte forma:
  - Ao se iniciar uma operação de leitura, o controlador da cache interpreta os bits do campo de conjuntos para identificar qual o conjunto desejado.
  - Em seguida, o sistema compara, no conjunto encontrado, o valor do campo tag do endereço com o valor do campo tag de cada quadro do conjunto encontrado.



Algoritmos de Substituição de Dados na Cache

- Definir qual dos blocos atualmente armazenados na cache deve ser retirado para dar lugar a um novo bloco que está sendo transferido (já que Q<<B).

- Quando isso se aplica?

Algoritmos de Substituição de Dados na Cache

- Dependendo de qual técnica de mapeamento se esteja usando, pode-se ter algumas opções de algoritmos:
  - Se o método de mapeamento for o direto, somente há uma única linha possível para um dado bloco
  - Para os outros dois métodos - associativo e associativo por conjunto - existem várias opções

Algoritmos de Substituição de Dados na Cache

- LRU: o sistema escolhe para ser substituído o bloco que está há mais tempo sem ser utilizado
- FILA: o primeiro a chegar é o primeiro a ser atendido. O sistema escolhe o bloco que está armazenado há mais tempo na cache.
- LFU: o sistema escolhe o bloco que tem tido menos acessos por parte da CPU
- Escolha aleatória: trata-se de escolher aleatoriamente um bloco para ser substituído

Política de Escrita pela Memória Cache

- Toda vez que a UCP realiza uma operação de escrita, esta ocorre imediatamente na cache.
- Quando atualizar a MP?
- Por que isso é importante?

Política de Escrita pela Memória Cache

- Considerações:
  - MP pode ser acessada tanto pela cache quanto por elementos de E/S. É possível que uma palavra da MP tenha sido alterada só na cache, ou um elemento de E/S pode ter alterado a palavra da MP e a cache esteja desatualizada
  - MP pode ser acessada por várias UCP´s. Uma palavra da MP é atualizada para atender à alteração de uma cache específica e as demais caches estarão desatualizadas.

Política de Escrita pela Memória Cache

- Técnicas:
  - Escrita em ambas (write through): cada escrita em uma palavra de cache acarreta escrita igual na palavra correspondente da MP
  - Escrita somente no retorno (write back): atualiza a MP apenas quando o bloco for substituído e se tiver ocorrido alguma alteração na cache.
    - Uso do bit ATUALIZA: 0, sem alterações; 1, houve alterações.
  - Escrita uma vez (write once): é uma técnica apropriada para sistemas multi UCP/cache, que compartilhem o mesmo barramento.
    - Primeira atualização: write through + alerta os demais componentes que compartilham o barramento único.

Política de Escrita pela Memória Cache

- Comparações:
  - Com write through pode haver uma grande quantidade de escritas desnecessárias na MP
  - Com write back, a MP fica desatualizada para dispositivos de E/S, por exemplo, o que os obriga a acessar o dado através da cache (problema!)
  - write once é conveniente para sistemas com múltiplas UCP´s
- Estudos mostram que a percentagem de escritas na MP é baixa (15%), o que aponta para uma simples política write through