Memória Cache

Sumário

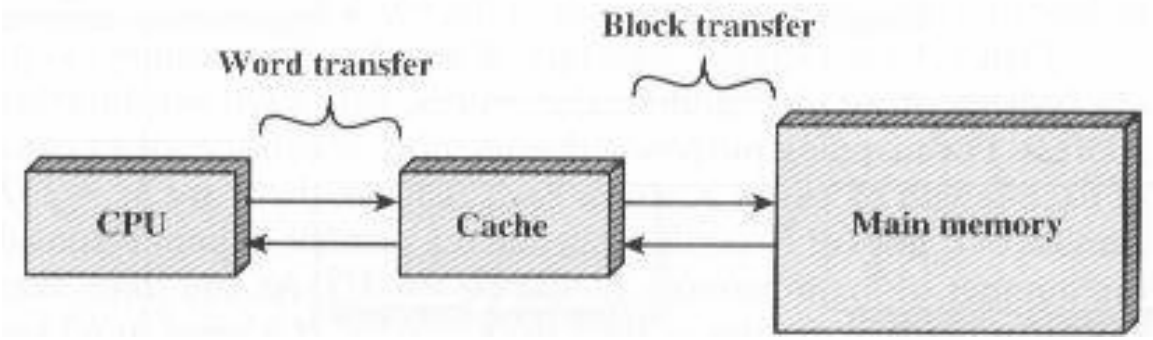
- Introdução;
- Projeto de Memórias Cache;
  - Tamanho;
  - Função de Mapeamento;
  - Política de Escrita;
  - Tamanho da Linha;
  - Número de Memórias Cache;
- Bibliografia.

Introdução

- A memória cache surgiu quando percebeu-se que as memórias não eram mais capazes de acompanhar o processador em velocidade;
- Dessa forma, a memória cache consiste numa pequena quantidade de memória SRAM, incluída no chip do processador.
- O uso de memória cache visa obter uma maior velocidade de acesso à memória, ao mesmo tempo, disponibilizar no sistema uma memória de grande capacidade;

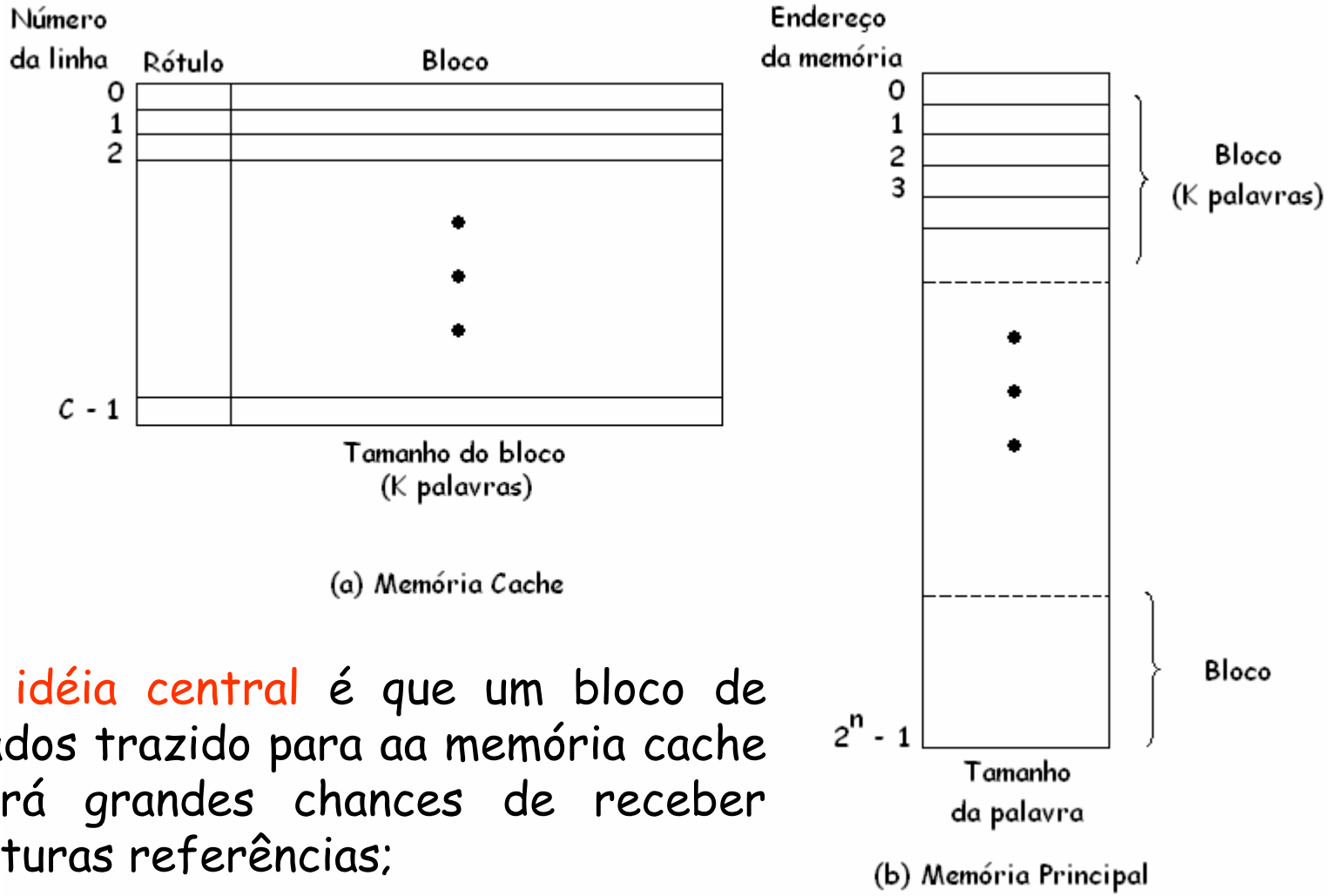
Introdução

- Eis uma figura que simplifica o conceito de memória cache:



  - A memória cache contém uma cópia de partes da memória principal;
  - Quando o processador deseja ler uma palavras da memória, é realizado um teste para determinar onde este dado está:
    - se a palavra estiver na memória cache, ela é fornecido ao processador diretamente;
    - caso contrário, um bloco de dados da memória principal, número fixo de palavras, é lido para a memória cache e em seguida a palavra requerida é entregue ao processador;

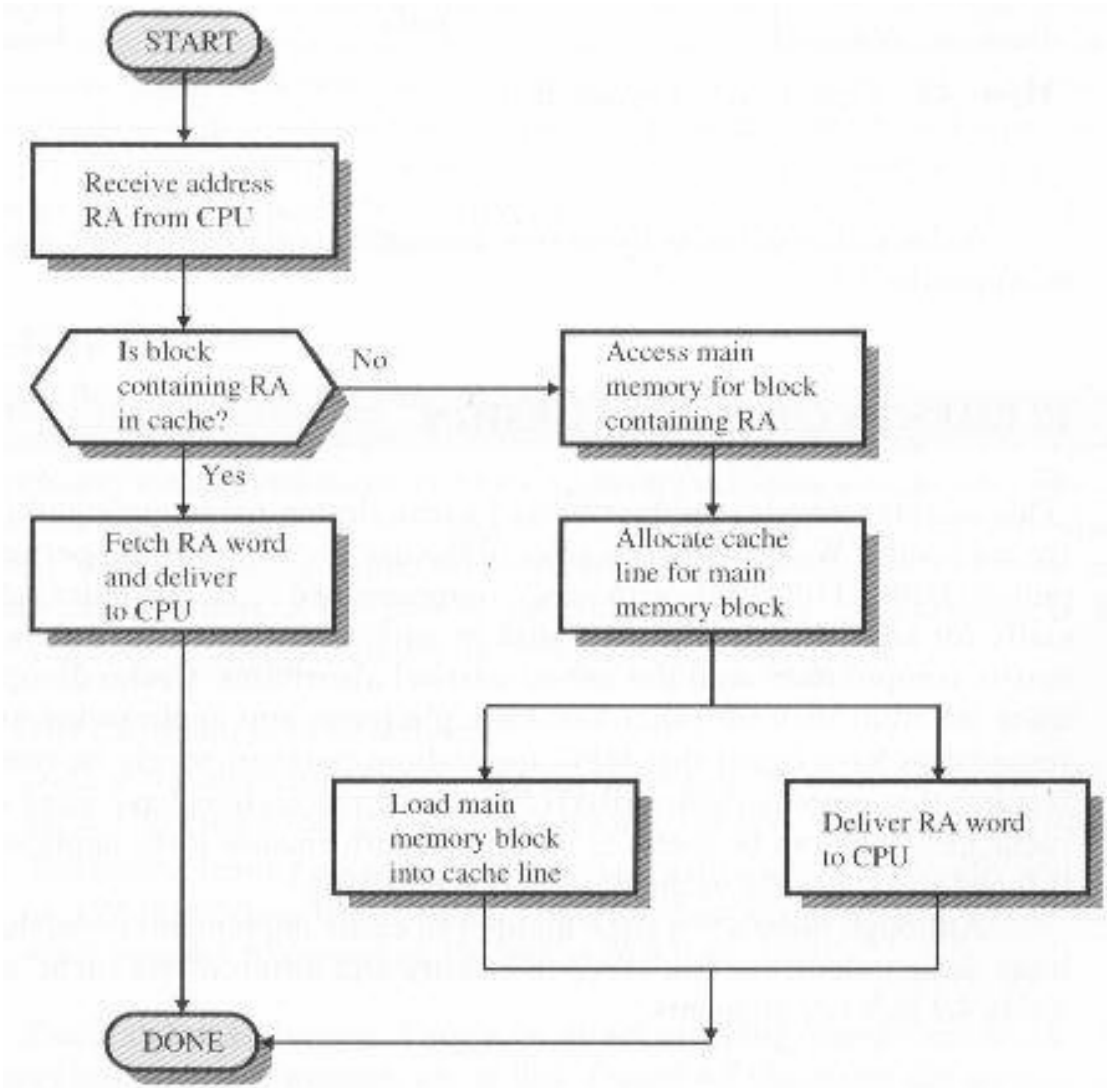
Introdução



- A idéia central é que um bloco de dados trazido para aa memória cache terá grandes chances de receber futuras referências;

Introdução

- Vejamos um fluxograma contendo a operação de leitura de um dado na memória cache:

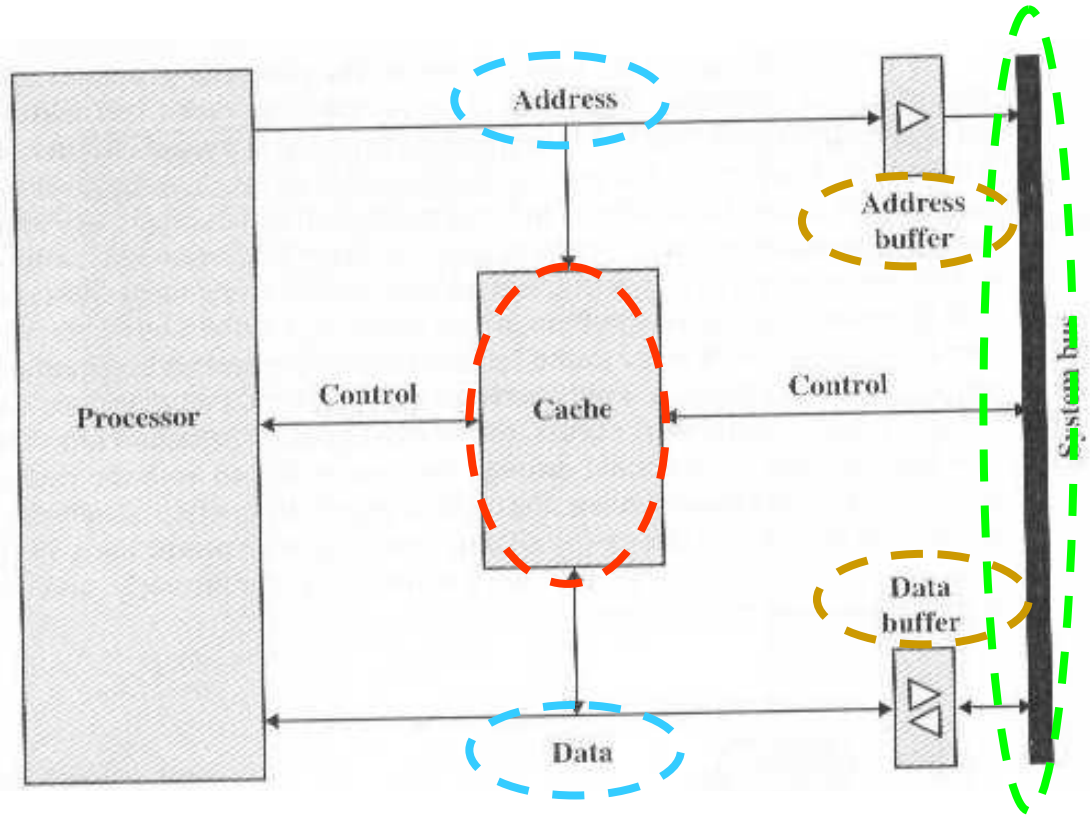


Introdução

- Considerações:
  - O processador gera um endereço, RA, da palavra a ser lida;
  - Se essa palavra estiver contida na memória cache, ela será entregue ao processador;
  - Caso contrário, o bloco que contém essa palavra será carregado na memória cache e a palavra entregue ao processador;
  - Essas duas últimas operações ocorrem paralelamente, refletindo a organização típica de memórias cache modernas;

Introdução

- Em memórias cache modernas as linhas de dados e de endereços são também conectadas a áreas de armazenamento temporário de dados e de endereços que se conectam ao barramento do sistema, por meio do qual é feito o acesso à memória principal.



Projeto de Memórias Cache

- Embora existam diversas implementações de memórias cache, poucos elementos básicos de projeto servem para classificar e diferenciar as diversas arquiteturas, são eles
  - Tamanho;
  - Função de Mapeamento;
  - Algoritmo de Substituição;
  - Política de Escrita;
  - Tamanho da Linha;
  - Número de Memórias Cache;

Tamanho

- O tamanho de uma memória cache deve ser:
  - Suficientemente pequeno para que o custo total médio por bit seja próximo do custo por bit da memória principal;
  - Suficientemente grande para que o tempo médio de acesso à memória seja próximo ao tempo de acesso da memória cache.

- "*Quanto maior é a memória cache, maior o número de portas envolvidas no seu endereçamento*" [Teoria Keciana e Manoelística]

- Entretanto o tamanho da memória cache tem crescido nos últimos anos, veja a tabela do próximo slide:

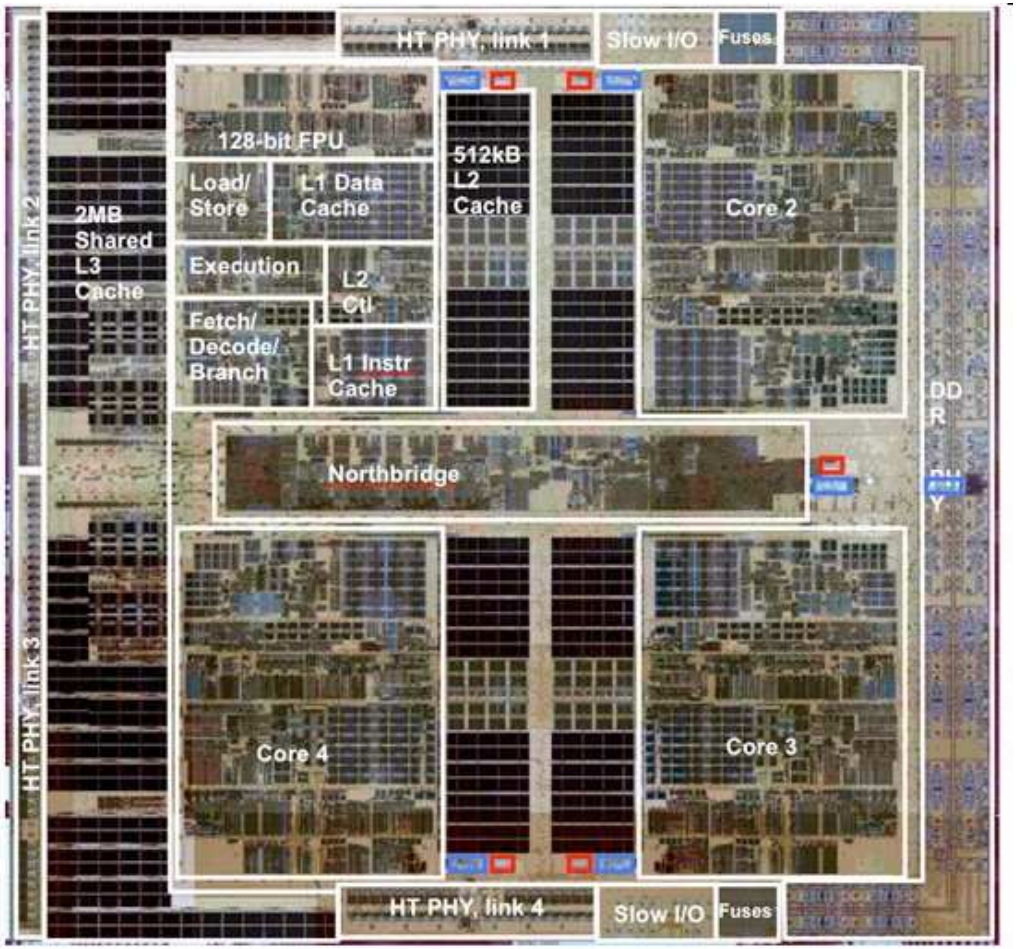
Tamanho

Processor	Type	Year of Introduction	L1 cache[a]	L2 cache	L3 cache
IBM 360/85	Mainframe	1968	16 to32 KB	—	—
PDP-11/70	Minicomputer	1975	1KB	—	—
VAX 11/780	Minicompterer	1978	16 KB	—	—
IBM 3033	Mainframe	1978	64 KB	—	—
IBM 3090	Mainframe	1985	128 to 256 KB	—	—
Intel 80486	PC	1989	8 KB	—	—
Pentium	PC	1993	8 KB/8 KB	256 to 512 KB	—
PowerPC 601	PC	1993	32 KB	—	—
PowerPC 620	PC	1996	32 KB/32 KB	—	—
PowerPC G4	PC/server	1999	32 KB/32 KB	256 KB to 1 MB	2 MB
IBM S/390 G4	Mainframe	1997	32 KB	256 KB	2 MB
IBM S/390 G6	Mainframe	1999	256 KB	8 MB	—
Pentium 4	PC/server	2000	8 KB/8 KB	256 KB	—
IBM SP	High-end server/ supercomputer	2000	64 KB/32 KB	8 MB	—
CRAY MTA[b]	PC/server	2001	16 KB/16 KB	96 KB	4 MB
Itanium	PC/server	2001	16 KB/16 KB	96 KB	4 MB
SGI Origin 2001	High-end server	2001	32 KB/32 KB	4 MB	—

Tamanho

- O tamanho de uma memória cache também é limitado pelo tamanho da pastilha e da placa de circuito;

- Em suma, como o desempenho de uma memória cache é muito sensível à natureza da carga de trabalho imposta, é impossível determinar um tamanho ótimo.



Função de Mapeamento

- Como vimos anteriormente, o número de linhas de memória cache é menor do que o de blocos da memória principal.

- Dessa forma, é necessário um algoritmo para mapear os blocos da memória principal em linhas da memória cache;

- Também é preciso um mecanismo para determinar o bloco da memória principal que ocupa uma dada linha da memória cache;

Função de Mapeamento

- A escolha da função que efetua esse mapeamento determina como a memória cache é organizada.

- Três técnicas diferentes podem ser usadas:

  - Mapeamento direto;

  - Mapeamento associativo;

  - Mapeamento associativo por conjuntos;

Função de Mapeamento

- Para cada função de mapeamento será mostrada a estrutura geral e, em seguida, um exemplo específico;

- O exemplo inclui os seguintes elementos:
  - A memória cache pode conter 64 Kbytes.
  - Os dados são transferidos entre a memória principal e a memória cache em blocos de 4 bytes. Dessa forma, a memória cache é organizada com 16K = 2^{14} linhas de 4 bytes cada uma.
  - A memória principal com 16 Mbytes, sendo cada byte endereçável diretamente por um endereço de 24 bits.

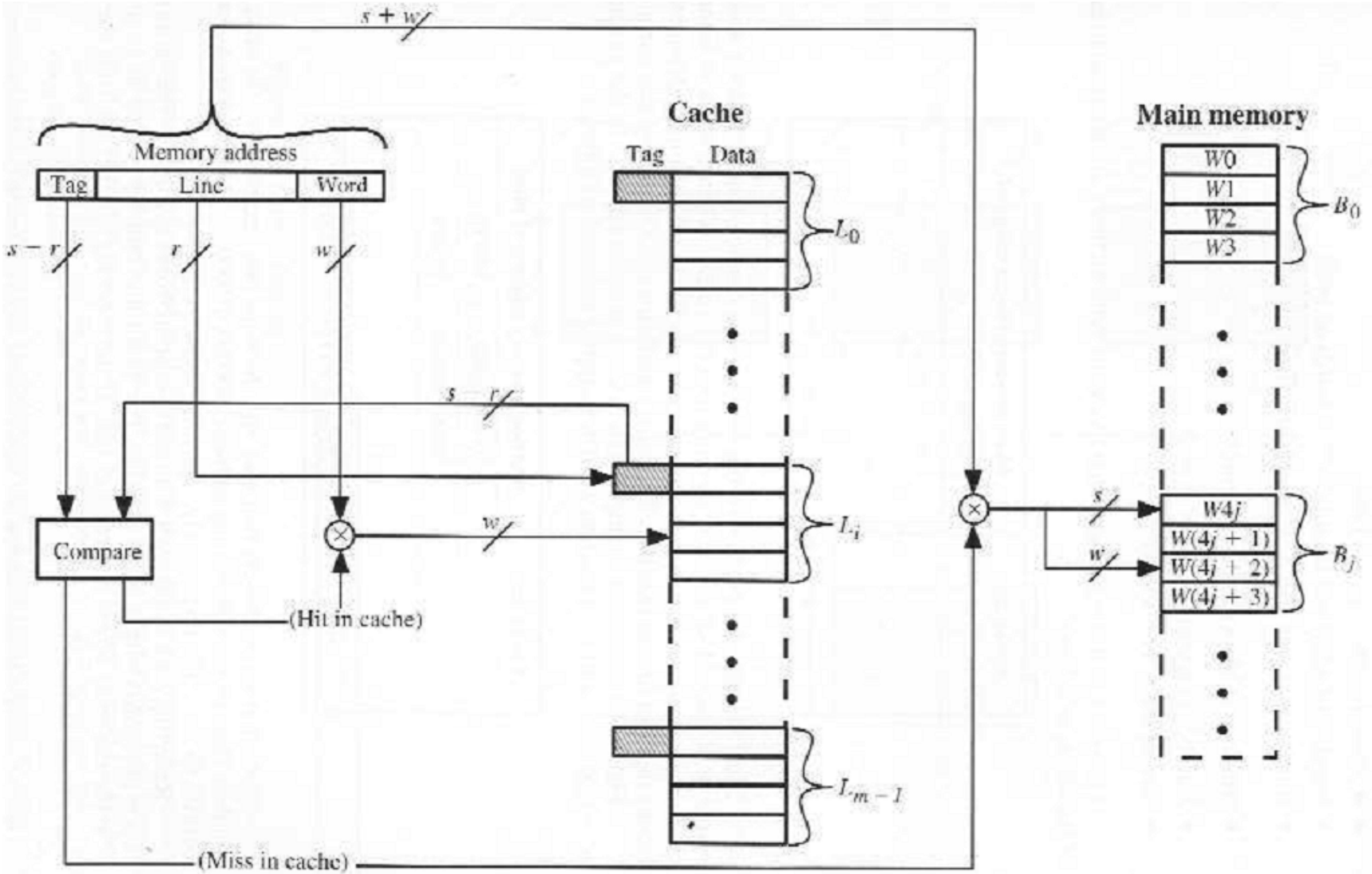
Mapeamento Direto

- No mapeamento direto, cada bloco da memória principal é mapeado em uma única linha de cache;

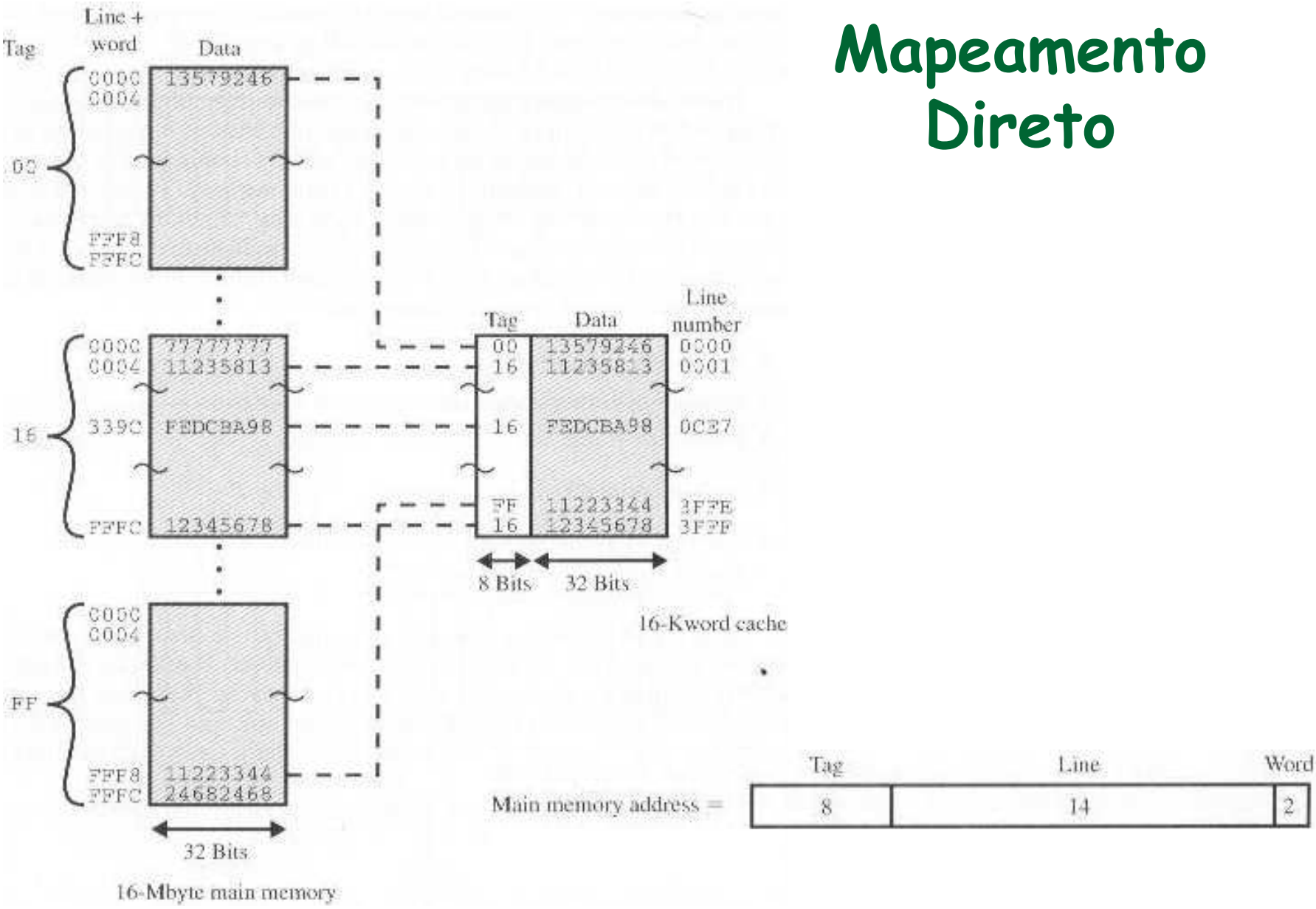
- Nos acessos à memória cache, um endereço da memória principal pode ser visto como constituído de três campos:
  - Os w bits menos significativos identificam uma única palavra ou byte dentro de um bloco da memória principal;

  - Os s bits restantes especificam um dos 2^s blocos da memória principal. Eles são interpretados na memória cache como compostos de um rótulo de s – r bits (mais significativos) e de um número de linha r bits que identifica uma das m = 2^r linhas da memória cache;

Mapeamento Direto



Mapeamento Direto



Mapeamento Direto

- A técnica de mapeamento direto é simples e tem custo de implementação baixo;
- Sua principal desvantagem é que cada bloco é mapeado em uma posição fixa na memória cache;
- Dessa forma, se um programa fizer repetidas referências a palavras de dois blocos distintos, mapeados em uma mesma linha, esses blocos será trocados continuamente na memória cache e a taxa de acertos à memória cache será baixa;

Mapeamento Associativo

- Evita as desvantagens do mapeamento direto, permitindo que cada bloco da memória principal seja carregado em qualquer linha de memória cache;

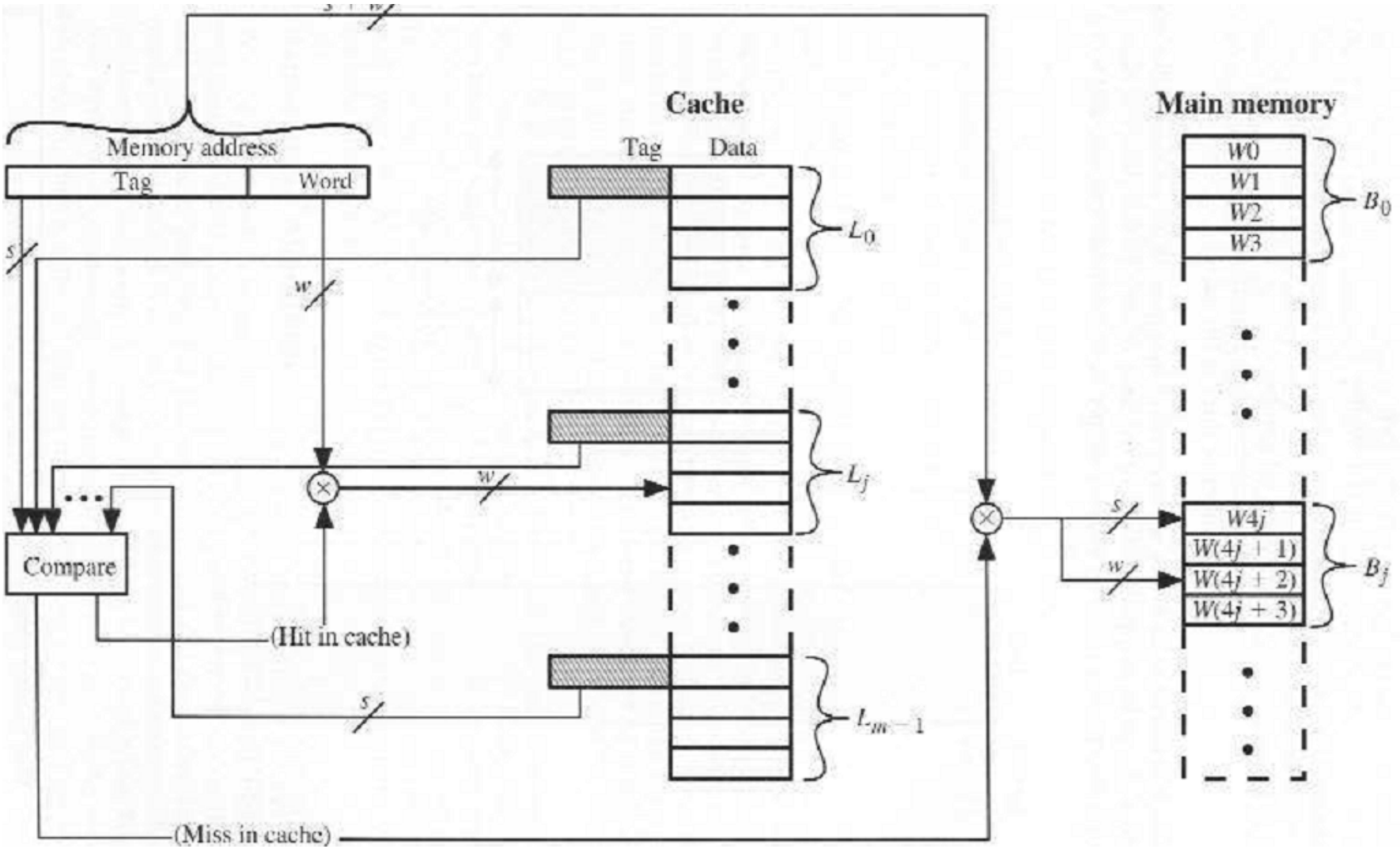
- Nesse caso, a lógica de controle da memória cache interpreta um endereço de memória como constituído simplesmente de um rótulo e um campo de palavra;

- O rótulo identifica um bloco da memória principal de modo unívoco.
  - Para determinar se um bloco está na memória cache, a lógica de controle da cache deve comparar simultaneamente o campo de rótulo do endereço do bloco acessado com os rótulos de todas as linhas;

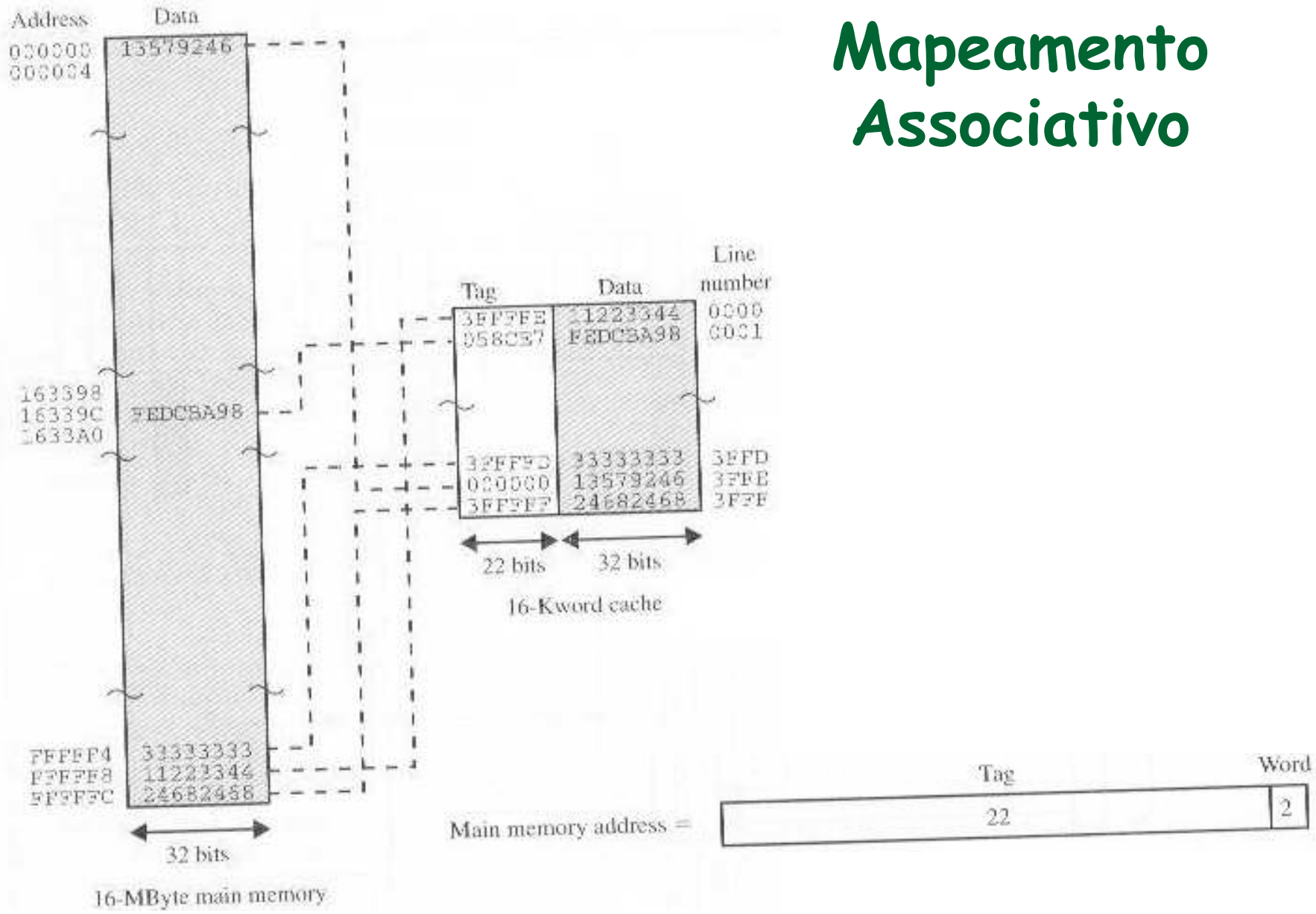
Mapeamento Associativo

- Um endereço de memória principal é composto de um rótulo de 22 bits e de um número de byte de 2 bits;
- Os 22 bits do rótulo são armazenados junto com os 32 bits de dados do bloco em cada linha de memória cache;
- Dessa forma, o endereço hexadecimal de 24 bits 16339C tem rótulo igual a 058CE7. Isso pode ser visto mais facilmente utilizando a notação binária:
  - Endereço de memória 0001 0110 0011 0011 1001 1100 (bin)
    1 6 3 3 9 C (hex)
  - Rótulo 00 0101 1000 1100 1110 0111 (bin)
    0 5 8 C E 7 (hex)

Mapeamento Associativo



Mapeamento Associativo



Mapeamento Associativo

- O mapeamento associativo oferece maior flexibilidade para a escolha do bloco a ser substituído quando um novo bloco é trazido para a memória cache;

- A principal desvantagem do mapeamento associativo é a complexidade do conjunto de circuitos para a comparação dos rótulos de todas as linhas da memória cache;

Mapeamento Associativo por Conjuntos

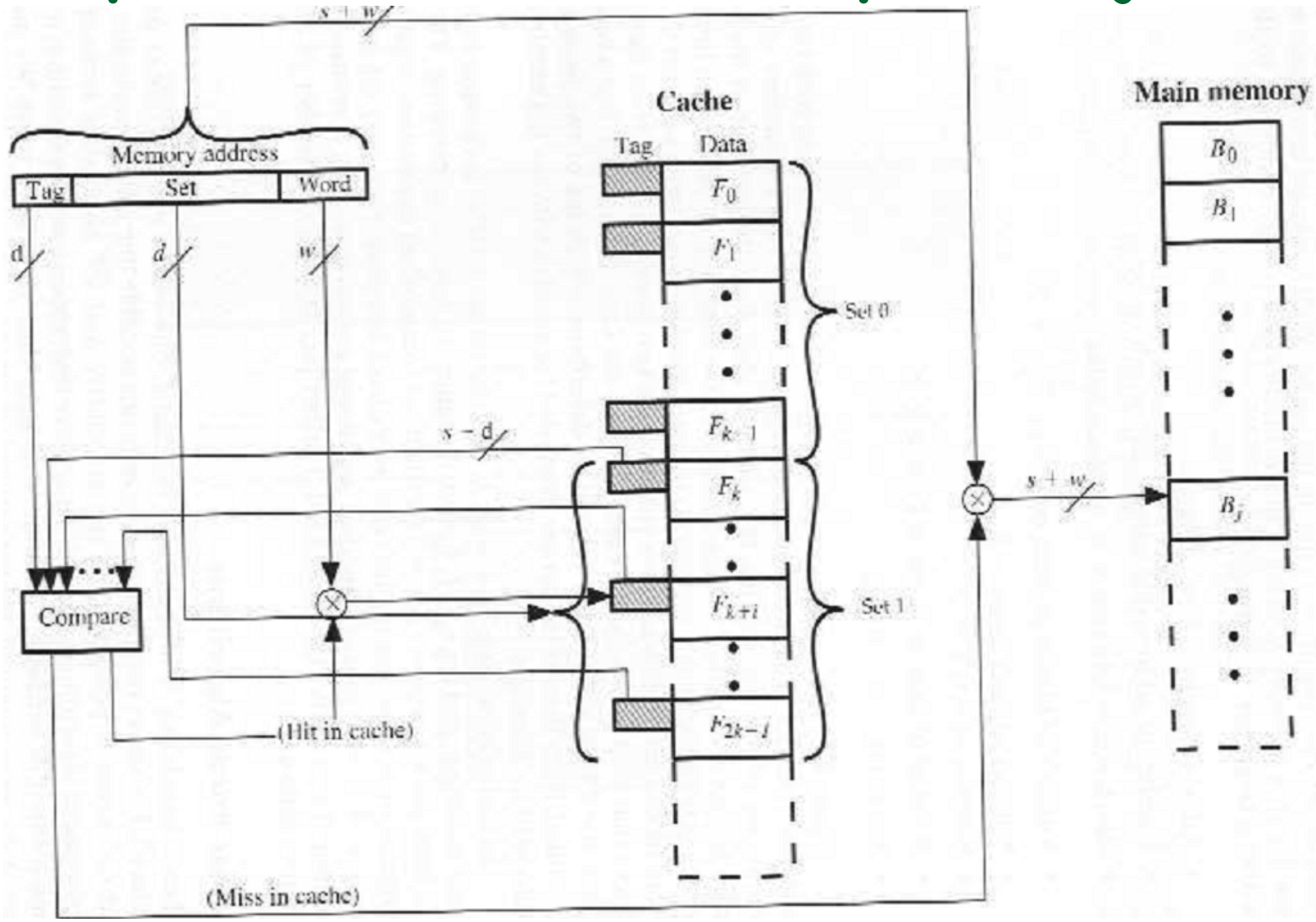
- Nesse mapeamento, é combinado vantagens do mapeamento direto e do mapeamento associativo e diminui suas desvantagens;
- A memória cache é dividida em v conjuntos, cada qual com k linhas.
- Em um mapeamento totalmente associativo, o rótulo de um endereço de memória é muito grande e é comparado com o rótulo de cada linha de memória cache;

Mapeamento Associativo por Conjuntos

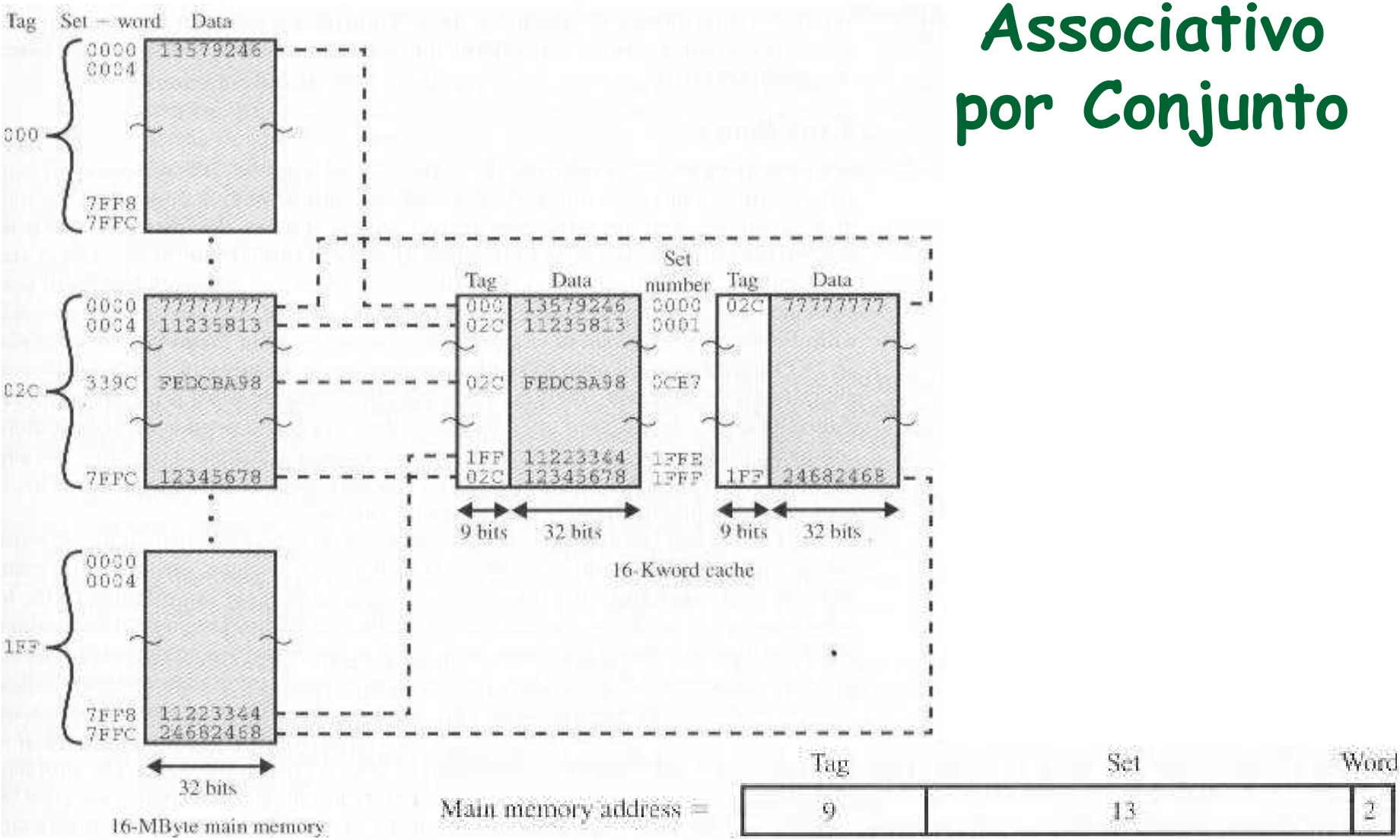
- Em um mapeamento associativo por conjuntos de k linhas, o rótulo de um endereço de memória é muito menor e é comparado apenas com k rótulos de um mesmo conjunto;

- A figura a seguir mostra nosso sistema utilizando um mapeamento associativo por conjuntos com duas linhas em cada conjunto, denominado mapeamento associativo por conjuntos de duas linhas;

Mapeamento Associativo por Conjuntos



Mapeamento Associativo por Conjunto



Mapeamento Associativo por Conjuntos

- A organização de mapeamento associativo por conjuntos mais comum é a que usa duas linhas por conjunto (v = m/2, k = 2);
- Sua taxa de acertos é significativamente maior do que no caso do mapeamento direto;
- Um maior número de linhas por conjunto não apresenta melhoras significativas de desempenho;

Algoritmo de Substituição

- Quando um novo bloco é trazido para a memória cache, um dos blocos existentes deve ser substituído;
- No mapeamento direto, cada bloco é mapeado em uma única linha, o que determina o bloco a ser substituído, não havendo alternativa possível;
- Nos demais mapeamentos, existe o uso de um algoritmo de substituição. Esse algoritmo é implementado em hardware para aumentar a velocidade de acesso à memória;

Algoritmo de Substituição

- Os quatro algoritmos mais comuns são:
  - Leaste recently used - LRU: substitui o bloco menos recentemente usado; É usado um bit de USO para indicar quando uma linha foi referenciada
  - First-in-First-out - FIFO: substitui o bloco do conjunto que foi armazenado primeiro na memória cache;
  - Least frequently used - LFU: substitui o bloco do conjunto menos freqüentemente utilizado; É usado um contado a cada linha da memória cache;
  - Substituição aleatória;

Política de Atualização

- Antes que um bloco residente na memória cache possa ser substituído, é necessário verificar se ele foi alterado na memória cache;
  - Se isso não ocorreu, então o novo bloco pode ser escrito sobre o bloco antigo;
  - Caso contrário, se pelo menos uma operação de escrita foi feita sobre uma palavra dessa linha de memória cache, então a memória principal deve ser atualizada;

- Dois problemas devem ser considerados:
  - Alteração da memória principal por um dispositivo de E/S;
  - Múltiplos processadores com múltiplas memórias cache;

Política de Atualização

- A técnica de atualização mais simples é denominada escrita direta (*write-through*);
  - Todas as operações são feitas tanto na memória cache quanto na memória principal;
  - A desvantagem é que ela gera um tráfego de memória considerável;

- Um técnica alternativa é a escrita de volta (write-back);
  - As escritas são feitas apenas na memória cache;
  - O bloco só é substituído se seu bit de ATUALIZAÇÃO estive com o valor 1;
  - Desvantagem: acesso de módulo de E/S direto à cache;

Tamanho da Linha

Número de Memórias Cache

Bibliografia

- Stallings, W. *Arquitetura e Organização de Computadores*, Pearson Hall, 5 ed. SP: 2002.