Arquitetura e Organização de Computadores – 5cop090

Memória cache

Objetivos

Objetivos

- Apresentar uma visão geral das características dos sistemas de memória do computador e do uso da hierarquia da memória.
- Descrever os conceitos básicos e o objetivo da memória cache.
- Discutir os elementos-chave do projeto da cache.
- Fazer distinção entre mapeamento direto, mapeamento associativo e mapeamento associativo por conjunto.
- Compreender as implicações do desempenho dos diversos níveis de memória.

Visão geral dos sistemas de memória

Principais pontos

- O desafio de projeto é organizar os dados e os programas na memória de modo que as palavras de memória acessadas normalmente estejam na memória mais rápida (cache).

- O sistema de computação possui uma hierarquia de subsistemas de memória, algumas internas ao sistema (acessível diretamente pelo processador) e algumas externas (acessíveis pelo processador por meio de um módulo de E/S).

Visão geral dos sistemas de memória

Características

- Localização.
- Capacidade.
- Unidade de transferência.
- Método de acesso.
- Desempenho.
- Tipo físico.
- Características físicas.
- Organização.

Visão geral dos sistemas de memória

Localização

- CPU.
- Interna.
- Externa.

Visão geral dos sistemas de memória

Capacidade

- Tamanho de palavra:
  - A unidade de organização natural.
- Número de palavras:
  - ou Bytes.

Visão geral dos sistemas de memória

Unidade de transferência

- Interna:
  - Normalmente controlada pela largura do barramento.
- Externa:
  - Normalmente um bloco que é muito maior que uma palavra.
- Unidade endereçável:
  - Menor local que pode ser endereçado exclusivamente.
  - Palavra internamente.
  - Cluster em discos.

Visão geral dos sistemas de memória

Métodos de acesso

- Sequencial:
  - Começa no início e lê em ordem.
  - Tempo de acesso depende da localização dos dados e local anterior.
  - Por exemplo, fita.
- Direto:
  - Blocos individuais possuem endereço exclusivo.
  - Acesso saltando para vizinhança, mais busca sequencial.
  - Tempo de acesso depende da localização e local anterior.
  - Por exemplo, disco.

Visão geral dos sistemas de memória

Métodos de acesso

- Aleatório:
  - Endereços individuais identificam localizações com exatidão.
  - Tempo de acesso é independente da localização ou acesso anterior.
  - Ex.: RAM.
- Associativo:
  - Dados são localizados por uma comparação com conteúdo de uma parte do armazenamento.
  - Tempo de acesso é independente do local ou acesso anterior.
  - Ex.: memória cache.

Visão geral dos sistemas de memória

Localização	Desempenho
Interna (por exemplo, registradores do processador, memória principal, cache)	Tempo de acesso
Externa (por exemplo, discos ópticos, discos magnéticos, fitas)	Tempo de ciclo
	Taxa de transferência
Método de acesso	**Tipo físico**
Sequencial	Semicondutor
Direto	Magnético
Aleatório	Óptico
Associativo	Magneto-óptico
Unidade de transferência	**Características físicas**
Palavra	Volátil/não volátil
Bloco	Apagável/não apagável
Capacidade	**Organização**
Número de palavras	Módulos de memória
Número de bytes	

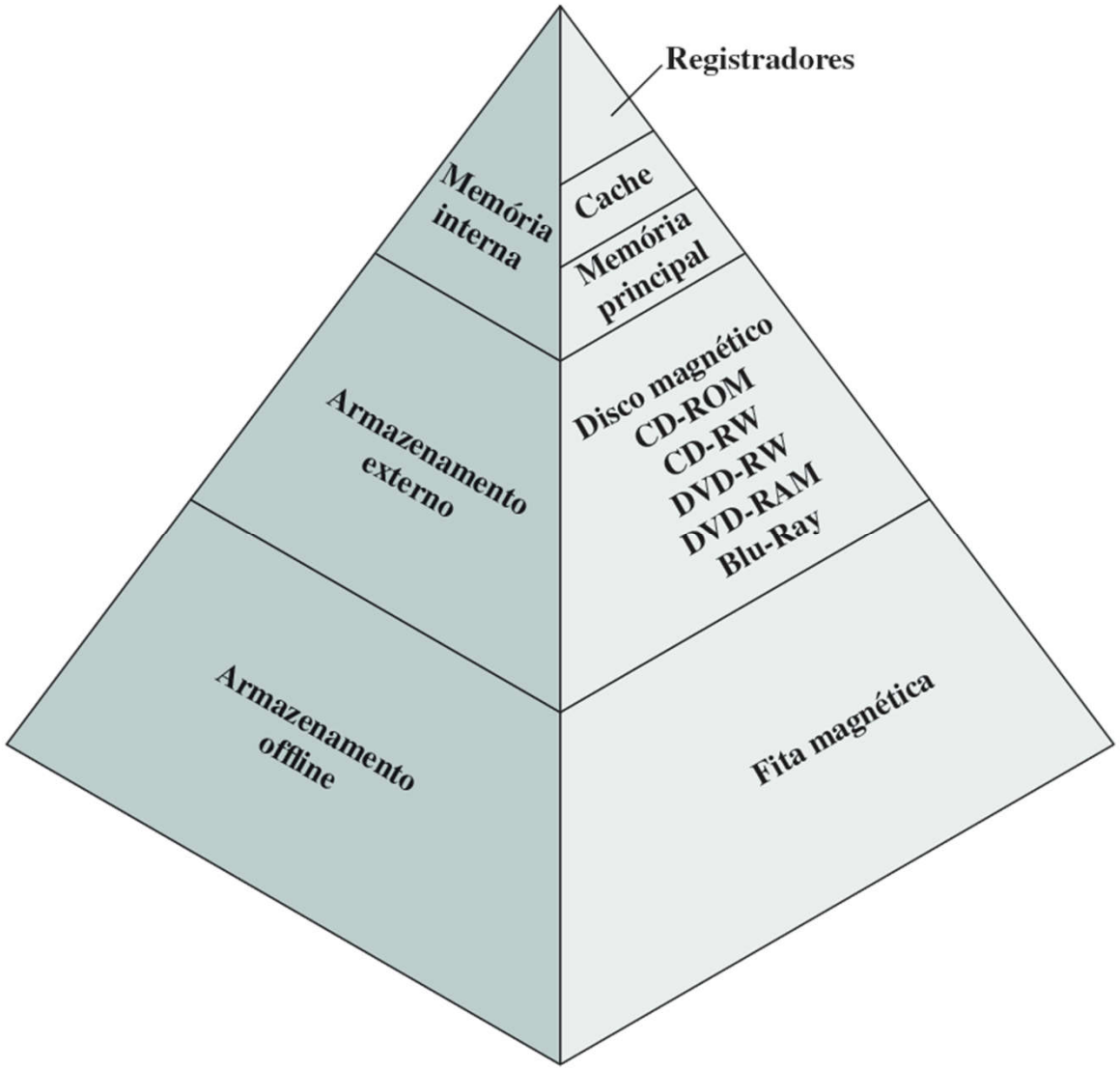
Memória Cache

Hierarquia de memória

- Registradores:
  - —Na CPU.
- Memória interna ou principal:
  - —Pode incluir um ou mais níveis de cache.
  - —“RAM”.
- Memória externa:
  - —Armazenamento de apoio.

Visão geral dos sistemas de memória

Hierarquia de memória – Diagrama



Visão geral dos sistemas de memória

Princípio da localidade de referência *[Denning, 1968]*

- Durante o curso de execução de um programa, as referências de memória pelo processador, para instruções e para dados, tendem a se agrupar.

- A propriedade de um processo ou sistema concentrar seus acessos em poucas áreas da memória a cada instante é chamada ***localidade de referências*** *[Denning, 2006].*

Visão geral dos sistemas de memória

Desempenho

- Tempo de acesso:
  - —Tempo entre apresentar o endereço e obter os dados válidos.
- Tempo de ciclo de memória
  - —Tempo que pode ser exigido para a memória se "recuperar" antes do próximo acesso.
  - —Tempo de ciclo é acesso + recuperação.
- Taxa de transferência:
  - —Taxa em que os dados podem ser movidos.

Visão geral dos sistemas de memória

Tipos físicos

- Semicondutor:
  - RAM.
- Magnético:
  - Disco e fita.
- Óptico:
  - CD e DVD.
- Outros:
  - Bolha.
  - Holograma.

Em um holograma óptico, diversas imagens podem ser gravadas em uma mesma superfície, variando a fase da luz coerente (laser) utilizada para gravá-las. Usando a mesma fase de luz coerente, as diferentes imagens podem ser recuperadas.

Visão geral dos sistemas de memória

Características físicas

- Deterioração.
- Volatilidade.
- Apagável.
- Consumo de energia.

Visão geral dos sistemas de memória

Organização

- Arranjo físico dos bits em palavras.
- Nem sempre de forma óbvia.
- Ex.: intercalada.

Visão geral dos sistemas de memória

A conclusão

- Capacidade
- Velocidade
- Custo

Visão geral dos sistemas de memória

Lista de hierarquia

- Registradores.
- Cache L1.
- Cache L2.
- Memória principal.
- Cache de disco.
- Disco.
- Óptica.
- Fita.

Localidade de referência

- Durante o curso da execução de um programa, as referências à memória tendem a se agrupar.
- Ex.: loops.

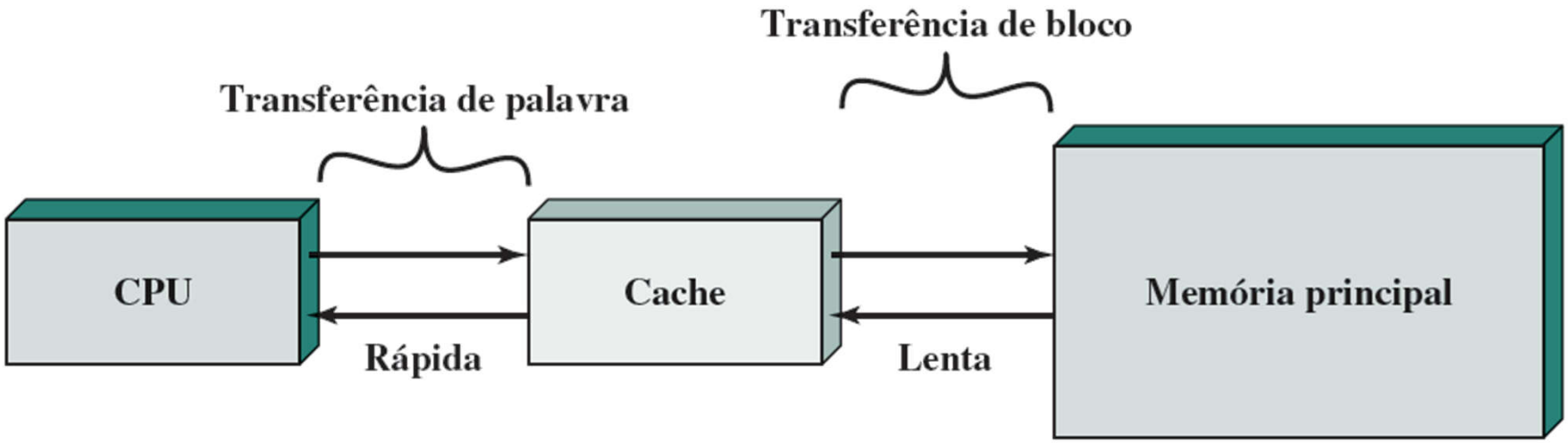
Princípios da memória cache

Cache

- Pequena quantidade de memória rápida.
- Fica entre a memória principal normal e a CPU.
- Pode estar localizada no chip da CPU ou módulo.

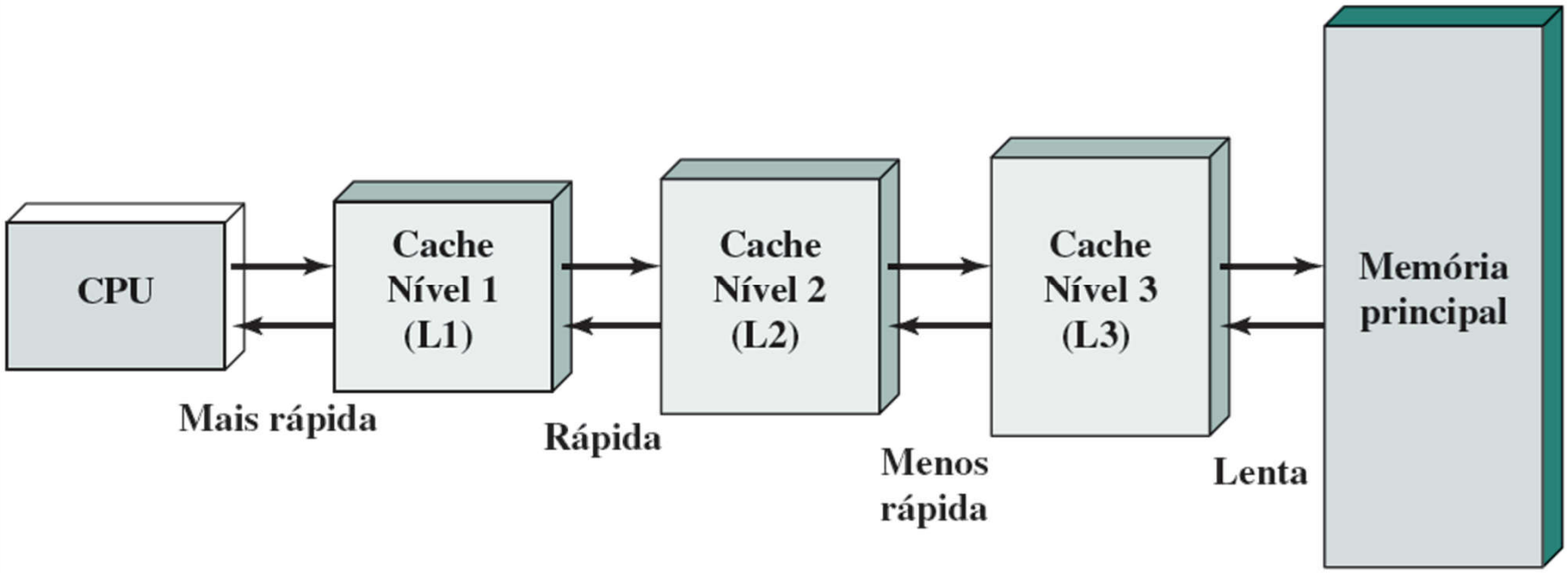
Princípios da memória cache

Cache e memória principal



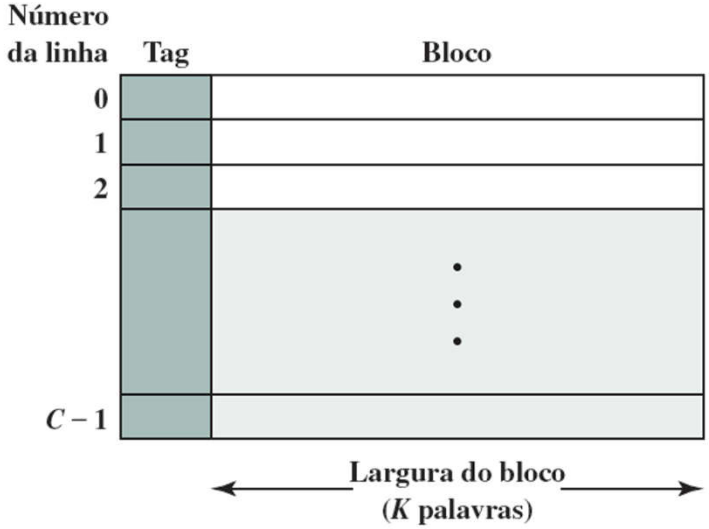
Princípios da memória cache

Cache e memória principal

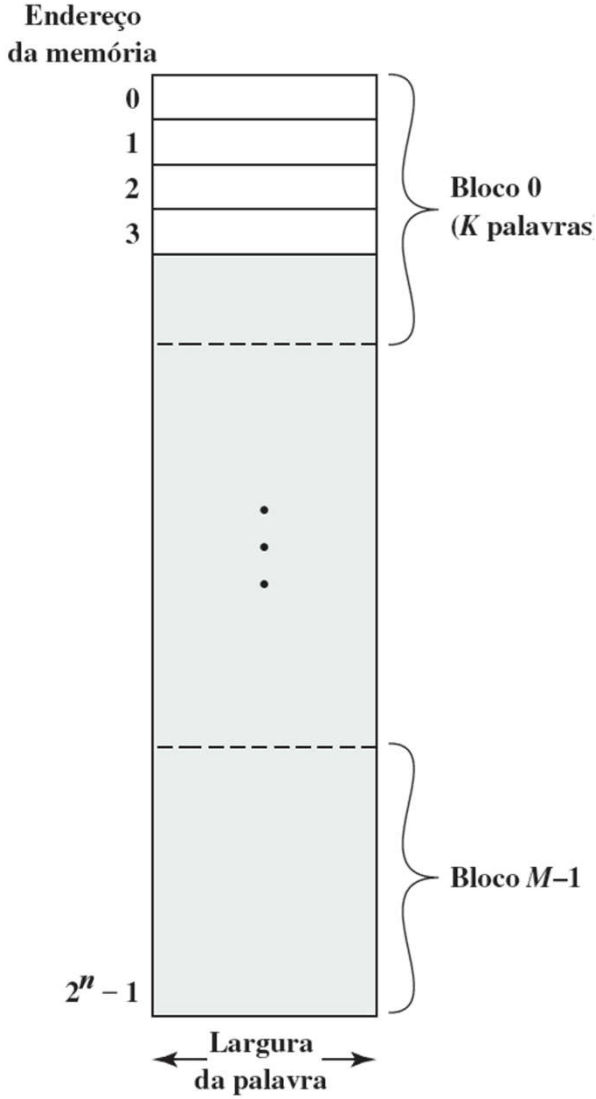


Princípios da memória cache

Estrutura de cache/memória principal



(a) Cache



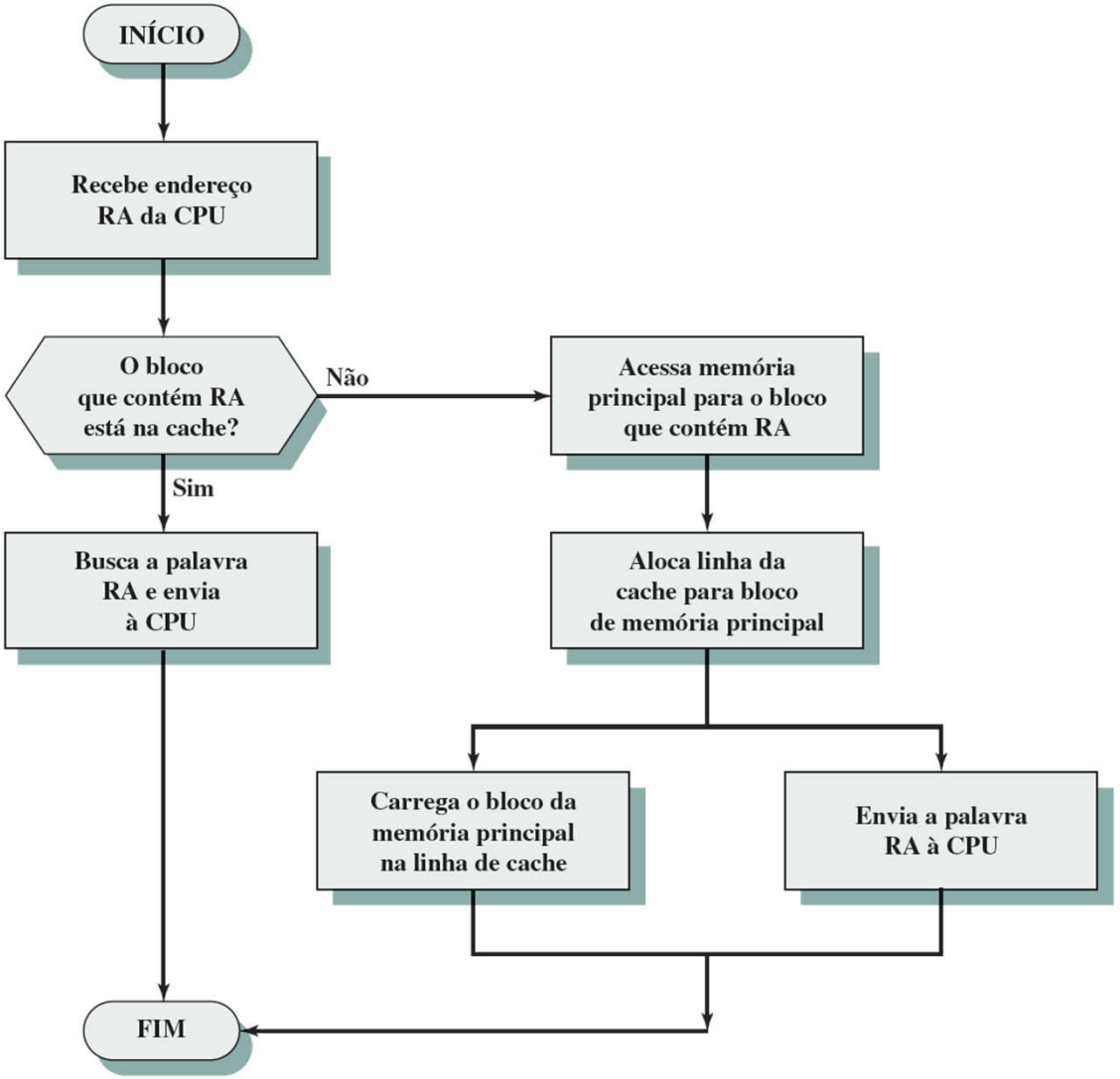
(b) Memória principal

Operação da cache – visão geral

- CPU requisita conteúdo do local de memória.
- Verifica se os dados estão em cache.
- Se estiverem, apanha da cache (rápido).
- Se não, lê bloco solicitado da memória principal para a cache.
- Depois, entrega da cache à CPU.
- Cache inclui tags para identificar qual bloco da memória principal está em cada slot da cache.

Princípios da memória cache

Operação de leitura de cache – fluxograma

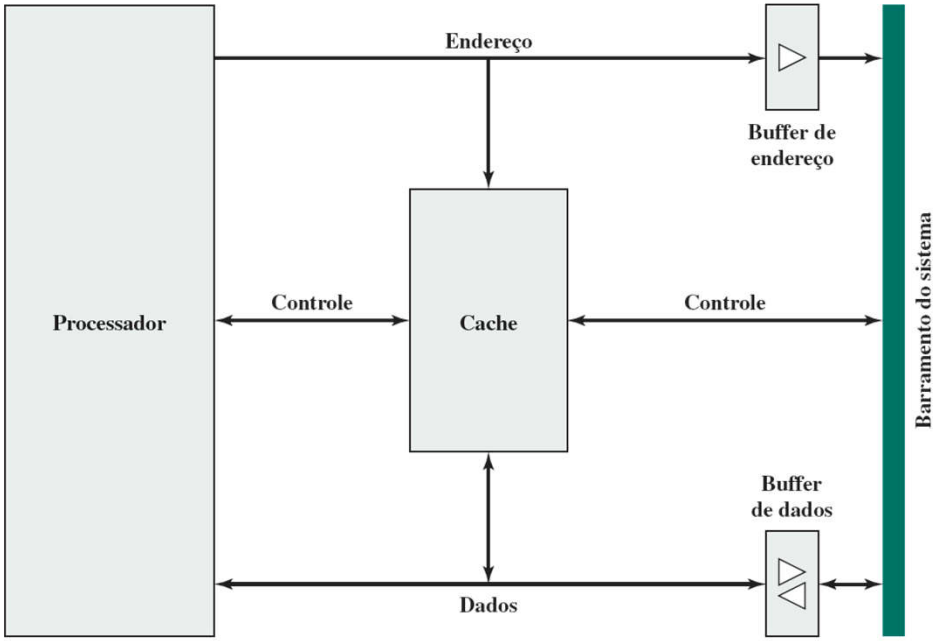


Princípios da memória cache

Organização típica da memória cache

Cache hit - os buffers de dados e endereço são desativados e a comunicação é apenas entre o processador e a memória cache, sem tráfego no barramento do sistema.

Cache miss - o endereço desejado é carregado no barramento do sistema e os dados são transferidos através do buffer de dados para a cache e para o processador.



Elementos do projeto da memória cache

Projeto de cache

- Endereçando.
- Tamanho.
- Função de mapeamento.
- Algoritmo de substituição.
- Política de escrita.
- Tamanho de bloco.
- Número de caches.

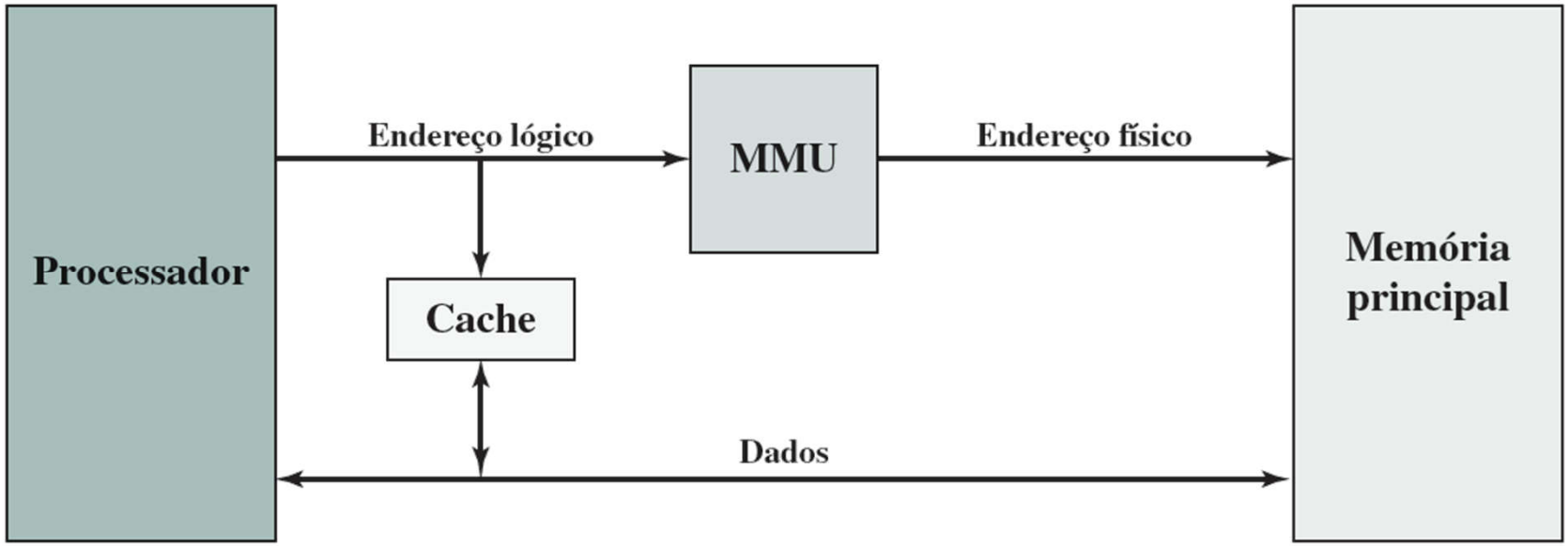
Memória Cache

Endereçamento de cache

- Onde fica a cache?
  - Entre processador e unidade de gerenciamento de memória virtual.
  - Entre MMU e memória principal.
- Cache lógica (cache virtual) armazena dados usando endereço virtual.
  - Processador acessa cache diretamente, não através da cache física.
  - Acesso à cache mais rápido, antes da tradução de endereço da MMU.
  - Endereços virtuais usam o mesmo espaço de endereços para diferentes aplicações.
    - Deve esvaziar cache a cada troca de contexto.
- Cache física armazena dados usando endereços físicos da memória principal.

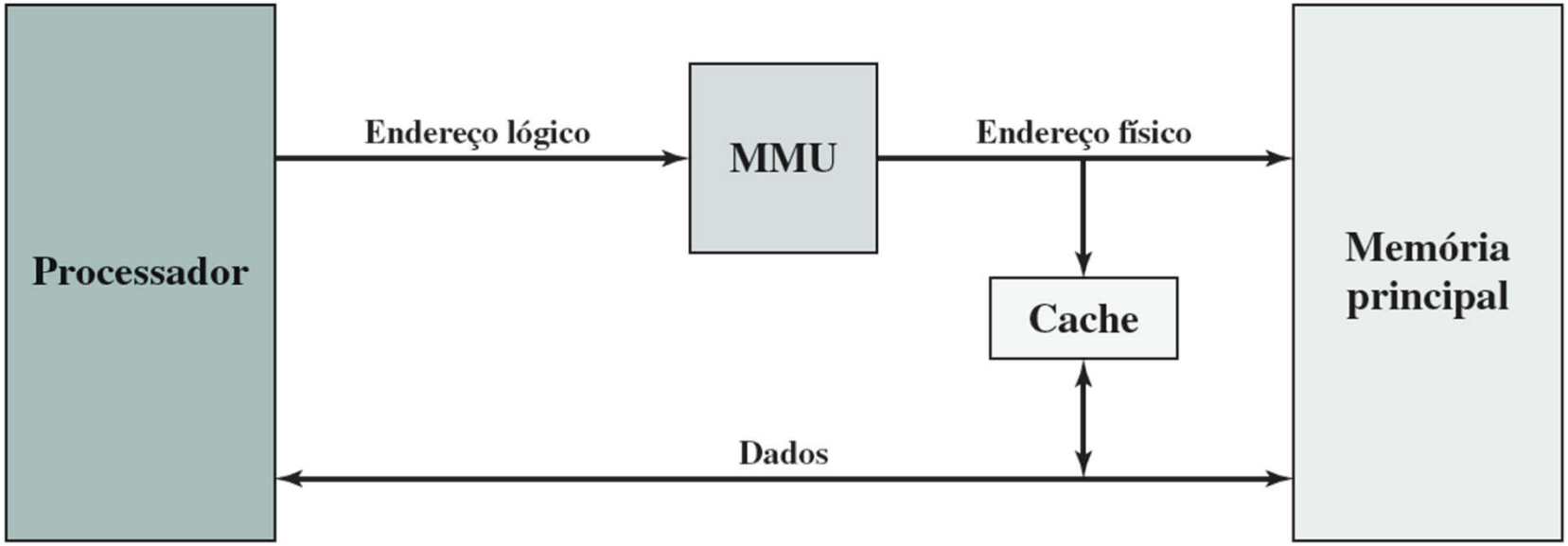
Elementos do projeto da memória cache

Organização típica da memória cache



Elementos do projeto da memória cache

Organização típica da memória cache



Elementos do projeto da memória cache

Tamanho

- Custo:
  - Quanto maior a cache – maior o custo.
- Velocidade:
  - Quanto mais cache => mais rápido (até certo ponto).
  - Verificar dados na cache leva tempo.

Elementos do projeto da memória cache

Função de mapeamento

- Função de mapeamento direta
- Função de mapeamento associativa
- Função de mapeamento associativa em conjunto (set associative)

Elementos do projeto da memória cache

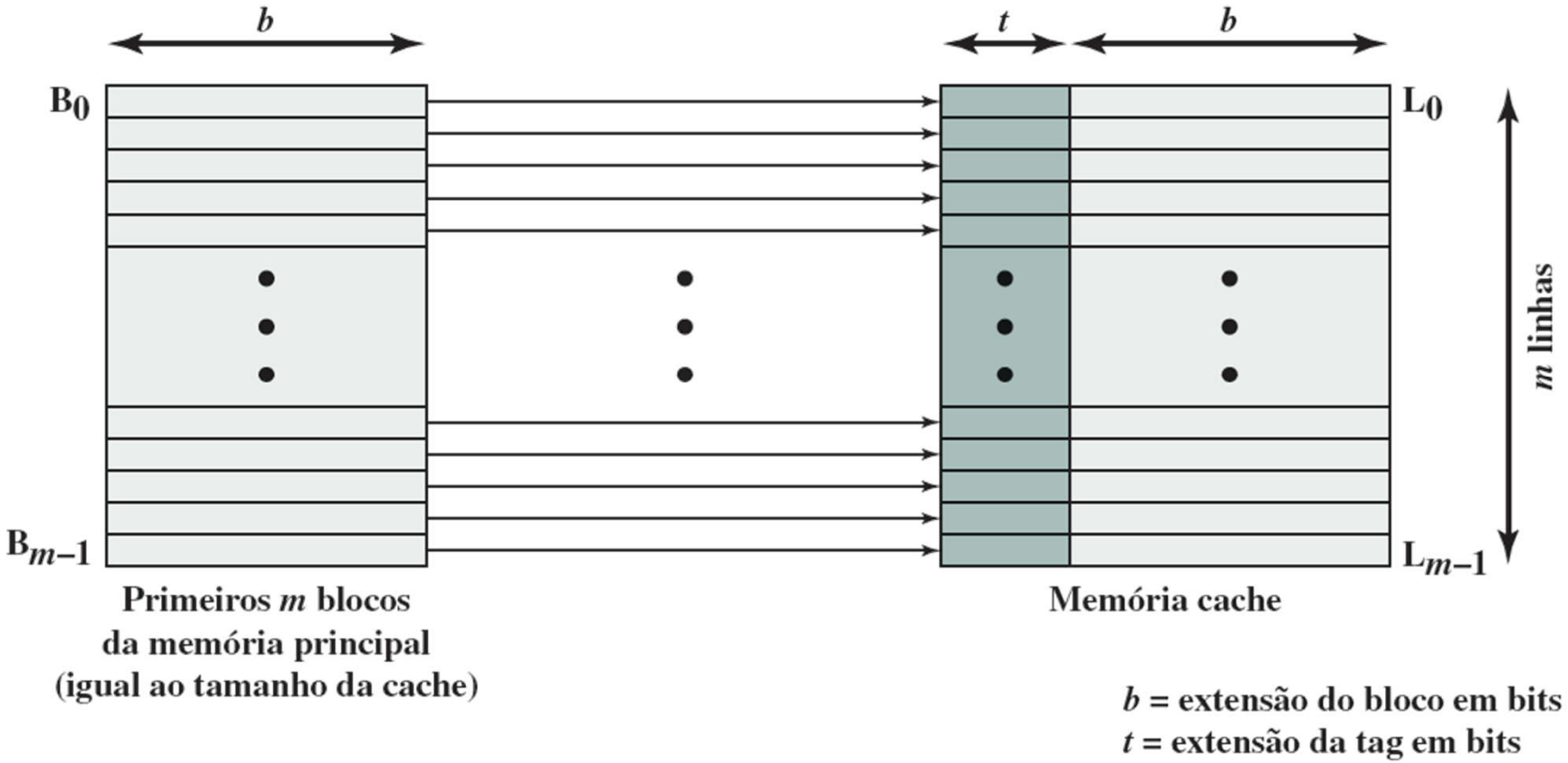
Função de mapeamento

Exemplo:

- Cache de 64 Kbytes.
- Bloco de cache de 4 bytes.
  - Ou seja, cache é de 16k (2^{14}) linhas de 4 bytes.
- 16 MB de memória principal.
- Endereço de 24 bits.
  - (2^{24}=16M)

Elementos do projeto da memória cache

Mapeamento direto da cache para memória principal



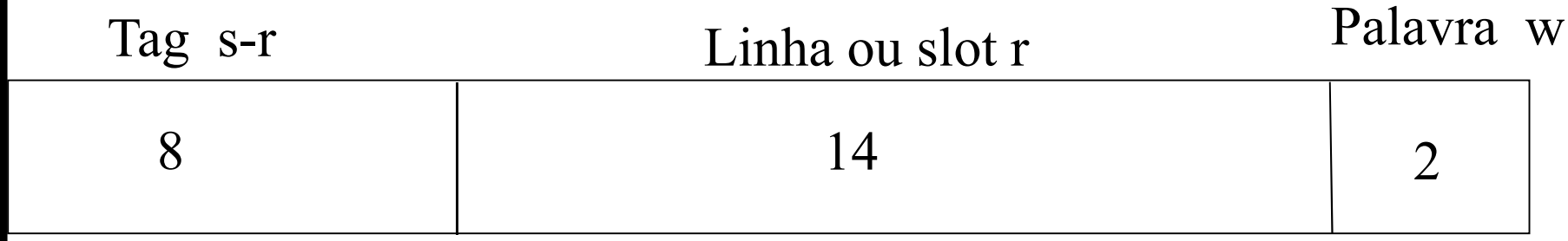
Elementos do projeto da memória cache

Mapeamento direto

- Cada bloco de memória principal é mapeado apenas para uma linha de cache.
  - Ou seja, se um bloco está na cache, ele deve estar em um local específico.
- Endereço está em duas partes.
- W bits menos significativos identificam word exclusiva.
- S bits mais significativos especificam um bloco de memória.
- Os MSBs são divididos em um campo de linha de cache r e uma tag de s-r (parte mais significativa).

Elementos do projeto da memória cache

Mapeamento direto
Estrutura de endereços



- Endereço de 24 bits.
- Identificador de palavra de 2 bits (bloco de 4 bytes).
- Identificador de bloco de 22 bits.
  - Tag de 8 bits (=22-14).
  - Slot ou linha de 14 bits.
- Dois blocos na mesma linha não têm o mesmo campo de tag.
- Verifica conteúdo da cache localizando linha e verificando tag.

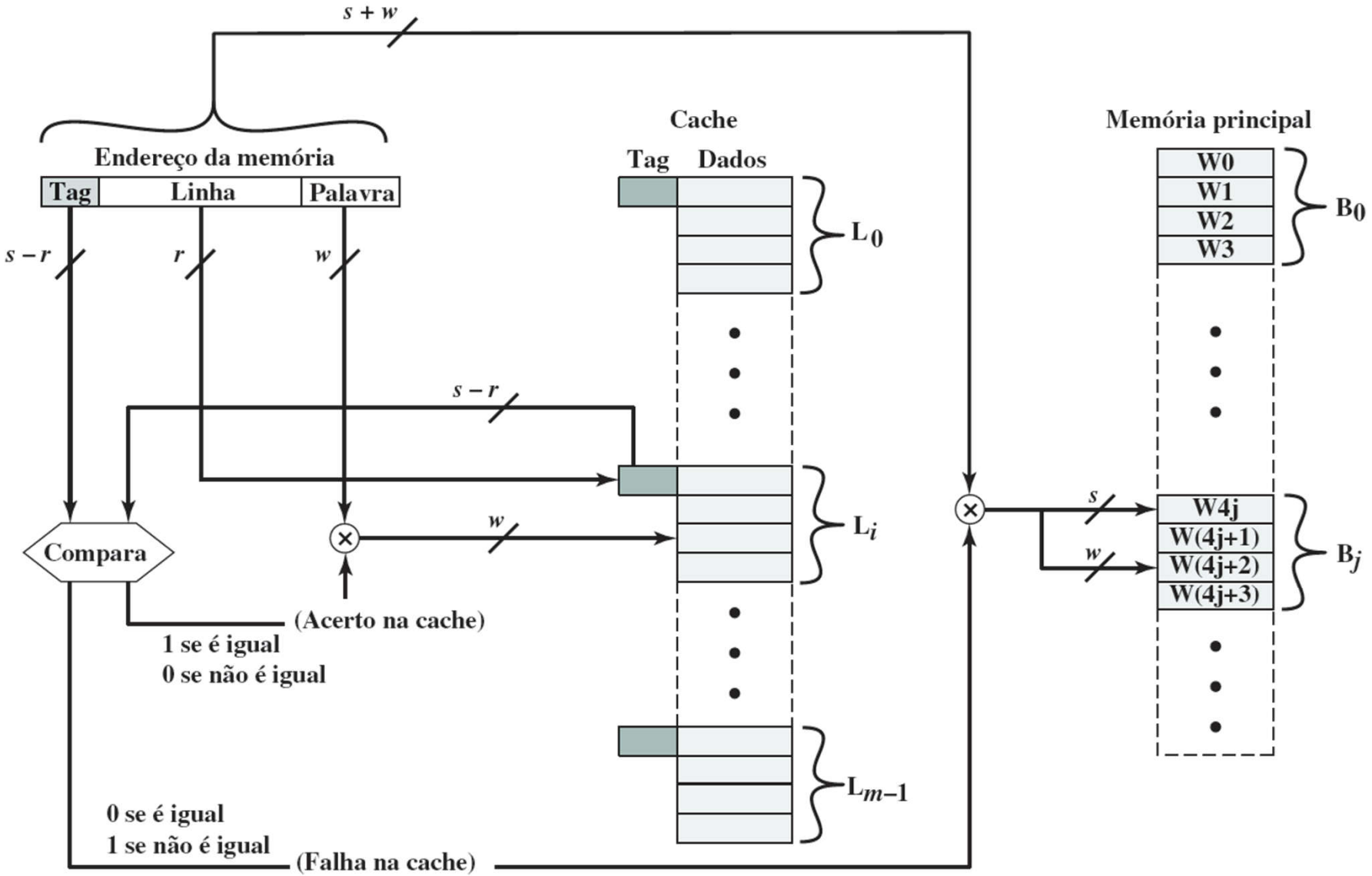
Elementos do projeto da memória cache

Mapeamento direto
Tabela de linhas de cache

Linha de cache	Blocos de memória principal mapeados
0	0, m, 2m, 3m…2s-m
1	1,m+1, 2m+1…2s-m+1
…	
m-1	m-1, 2m-1,3m-1…2s-1

Elementos do projeto da memória cache

Organização da cache com mapeamento direto



Elementos do projeto da memória cache

Mapeamento direto (resumo)

Tamanho do endereço $= (s + w)$ bits

Número de unidades endereçáveis $= 2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha $= 2^{w}$ palavras ou bytes

Número de blocos na memória principal $= \frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas na cache $= \mathrm{m} = 2^{\mathrm{r}}$

Tamanho da cache $= 2^{(r+w)}$ palavras ou bytes

Tamanho da tag $= (s - r)$ bits

Elementos do projeto da memória cache

Prós e contras do mapeamento direto

- Simples.
- Barato.
- Local fixo para determinado bloco.
  - Se um programa acessa 2 blocos que mapeiam para a mesma linha repetidamente, perdas de cache são muito altas.

Elementos do projeto da memória cache

Exercícios

1) Considere um computador com as seguintes características: total de 2 MBytes de memória principal; tamanho de palavra de 1byte; tamanho de bloco de 16 bytes; e tamanho de cache de 64 KBytes. Determine o formato dos endereços da memória principal para uma cache mapeada diretamente.

$$\text{Tamanho do endereço} = (s + w) \text{ bits}$$

$$\text{Número de unidades endereçáveis} = 2^{(s+w)} \text{ palavras ou bytes}$$

$$\text{Tamanho do bloco} = \text{tamanho da linha} = 2^{w} \text{ palavras ou bytes}$$

$$\text{Número de blocos na memória principal} = \frac{2^{(s+w)}}{2^{w}} = 2^{s}$$

$$\text{Número de linhas na cache} = \mathrm{m} = 2^{\mathrm{r}}$$

$$\text{Tamanho da cache} = 2^{(r+w)} \text{ palavras ou bytes}$$

$$\text{Tamanho da tag} = (s - r) \text{ bits}$$

Elementos do projeto da memória cache

Exercícios

2) Um computador possui uma memória principal (MP) com capacidade para 4 Gbytes. Cada célula desta memória tem capacidade para 8 bits. Foi colocada neste computador uma memória cache (MC) de mapeamento direto com capacidade para 256 Kbytes. Cada linha desta cache tem capacidade para 64 bytes. Supondo que a CPU faça um acesso ao endereço (53A249CE)16, determine:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) Total de bits para o número da linha	
(d) O total de bits para a TAG	
(e) O número da WORD (em hexadecimal)	
(f) O número da linha (em hexadecimal)	
(g) O valor da TAG (em hexadecimal)	

Tamanho do endereço $= (s + w)$ bits

Número de unidades endereçáveis $= 2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha $= 2^{w}$ palavras ou bytes

Número de blocos na memória principal $= \frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas na cache = m = 2^{r}

Tamanho da cache $= 2^{(r+w)}$ palavras ou bytes

Tamanho da tag $= (s - r)$ bits

Elementos do projeto da memória cache

Exercícios

3) Um computador possui uma memória principal (MP) com capacidade para 2 Gbits. Cada célula desta memória tem capacidade para 1byte. Foi colocada neste computador uma memória cache (MC) de mapeamento direto com capacidade para 512 Kbytes. Cada linha desta cache tem capacidade para 16 células. Supondo que a CPU faça um acesso ao endereço (035AFBE5)16, determine:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) Total de bits para o número da linha	
(d) O total de bits para a TAG	
(e) O número da WORD (em hexadecimal)	
(f) O número da linha (em hexadecimal)	
(g) O valor da TAG (em hexadecimal)	

Elementos do projeto da memória cache

Exercícios

4) Um computador possui uma memória principal (MP) com capacidade para 2 Gbits. Cada célula desta memória tem capacidade para 2 bytes. Foi colocada neste computador uma memória cache (MC) de mapeamento direto com capacidade para 1 Mbyte. Cada linha desta cache tem capacidade para 512 bits. Supondo que a CPU faça um acesso ao endereço (06EC78AE)16, determine:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) Total de bits para o número da linha	
(d) O total de bits para a TAG	
(e) O número da WORD (em hexadecimal)	
(f) O número da linha (em hexadecimal)	
(g) O valor da TAG (em hexadecimal)	

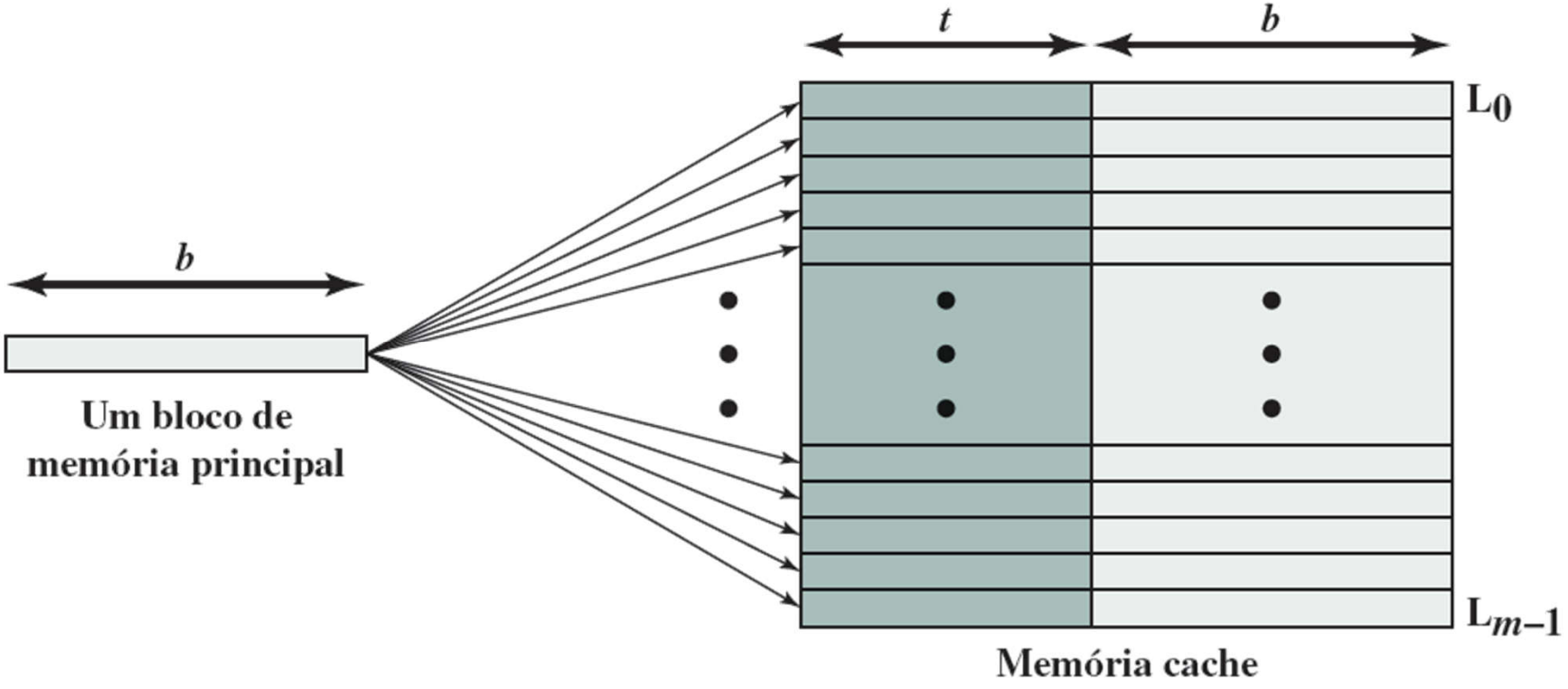
Elementos do projeto da memória cache

Mapeamento associativo

- Um bloco de memória principal pode ser carregado em qualquer linha de cache.
- Endereço de memória é interpretado como tag e palavra.
- Tag identifica exclusivamente o bloco de memória.
- Tag de cada linha é examinada em busca de combinação.
- Pesquisa da cache é dispendiosa.

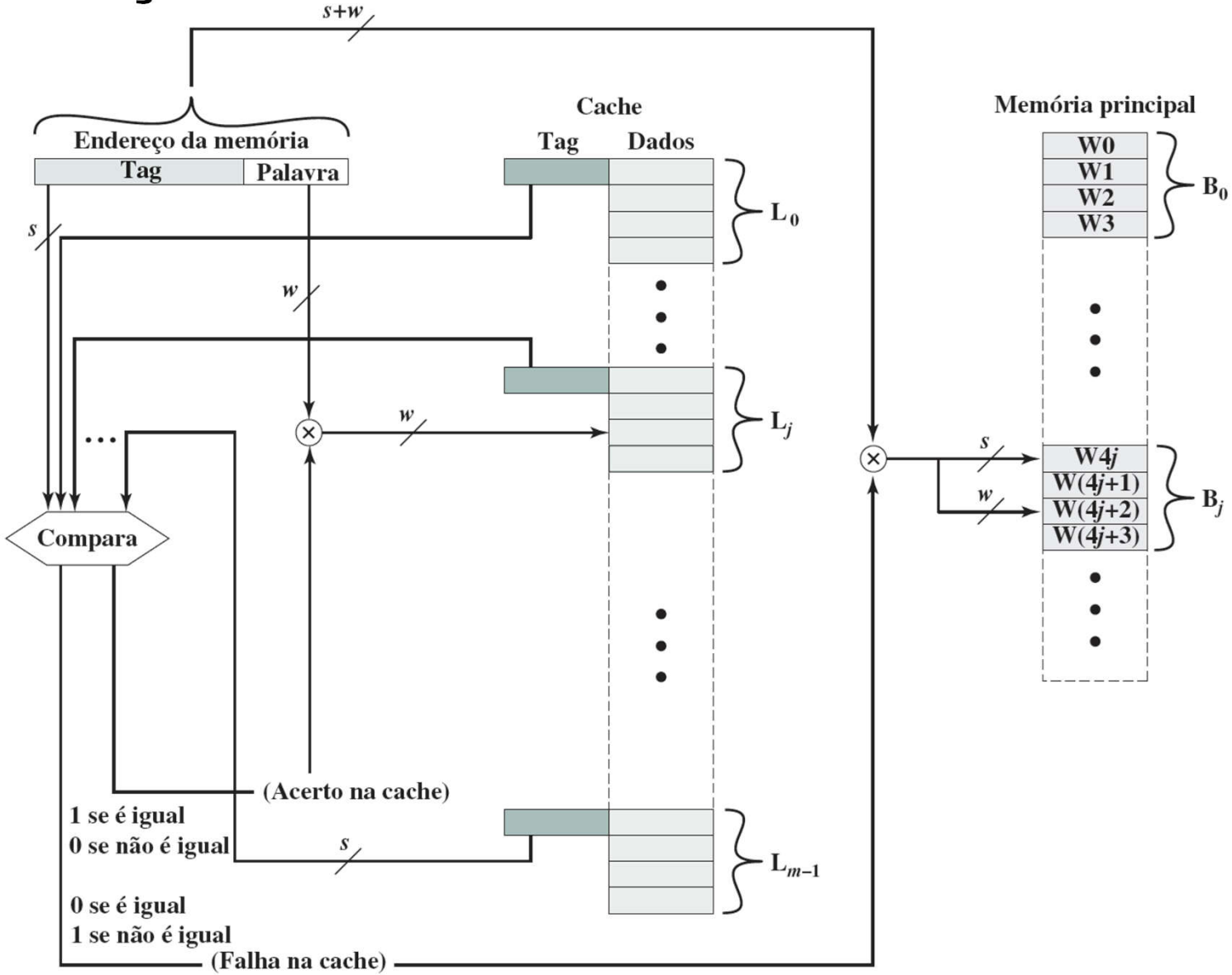
Elementos do projeto da memória cache

Mapeamento associativo da cache para a memória principal



Memória Cache

Organização de cache totalmente associativa



Memória Cache

Mapeamento associativo
Estrutura de endereço

Tag 22 bit	Palavra 2 bit

- Tag de 22 bits armazenado a cada bloco de 32 bits de dados.
- Compara campo de tag com entrada de tag na cache para procurar acerto.
- 2 bits menos significativos do endereço identificam qual word de 16 bits é exigida do bloco de dados de 32 bits.
- P.ex.:

Endereço	Tag	Dados	Linha de cache
FFFFFC	FFFFFC	24682468	3FFF

Memória Cache

Resumo do mapeamento associativo

Tamanho do endereço = $(s + w)$ bits

Número de unidades endereçáveis = $2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha = 2^{w} palavras ou bytes

Número de blocos na memória principal = $\frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas na cache = indeterminado

Tamanho da tag = s bits

Memória Cache

Exercícios

1) Considere um computador com as seguintes características: total de 2 MBytes de memória principal; tamanho de palavra de 1byte; tamanho de bloco de 16 bytes; e tamanho de cache de 64 KBytes. Determine o formato dos endereços da memória principal para uma cache totalmente associativa.

Tamanho do endereço $= (s + w)$ bits

Número de unidades endereçáveis $= 2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha $= 2^{w}$ palavras ou bytes

Número de blocos na memória principal $= \frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas na cache = indeterminado

Tamanho da tag $= s$ bits

Memória Cache

Exercícios

2) Um computador possui uma memória principal com capacidade para 4 Gbits. Cada célula desta memória tem capacidade para 1 byte. Foi colocado neste computador uma memória cache puramente associativa com capacidade para 512 kBytes. Cada linha desta cache tem capacidade pra 16 células. Supondo que a CPU faça um acesso ao endereço (036D7BC5)16, determine:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) O total de bits para a TAG	
(d) O número da WORD (em hexadecimal)	
(e) O valor da TAG (em hexadecimal)	

$$\text{Tamanho do endereço} = (s + w) \text{ bits}$$

$$\text{Número de unidades endereçáveis} = 2^{(s+w)} \text{ palavras ou bytes}$$

$$\text{Tamanho do bloco} = \text{tamanho da linha} = 2^{w} \text{ palavras ou bytes}$$

$$\text{Número de blocos na memória principal} = \frac{2^{(s+w)}}{2^{w}} = 2^{s}$$

$$\text{Número de linhas na cache} = \text{indeterminado}$$

$$\text{Tamanho da tag} = s \text{ bits}$$

Memória Cache

Mapeamento associativo em conjunto

- Cache é dividida em uma série de conjuntos.
- Cada conjunto contém uma série de linhas.
- Determinado bloco é mapeado a qualquer linha em determinado conjunto.
  - P.ex.:Bloco B pode estar em qualquer linha do conjunto i.
- P.ex.: 2 linhas por conjunto:
  - Mapeamento associativo com 2 linhas.
  - Determinado bloco pode estar em uma de 2 linhas em apenas um conjunto.

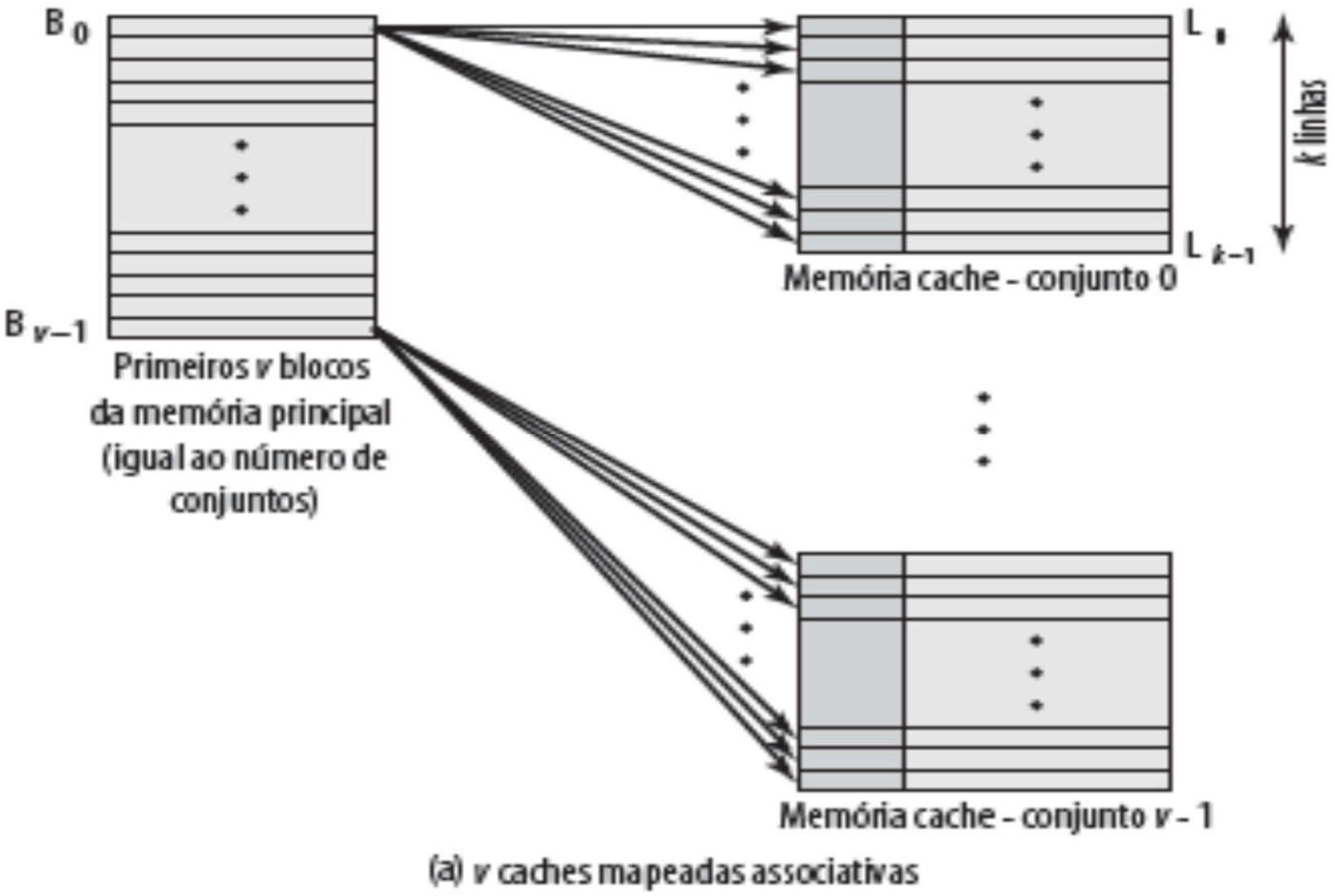
Memória Cache

Mapeamento associativo em conjunto
Exemplo

- Número de conjunto com 13 bits
- Número de bloco na memória principal é módulo 2^{13}
- 000000, 00A000, 00B000, 00C000 … mapeados no mesmo conjunto

Memória Cache

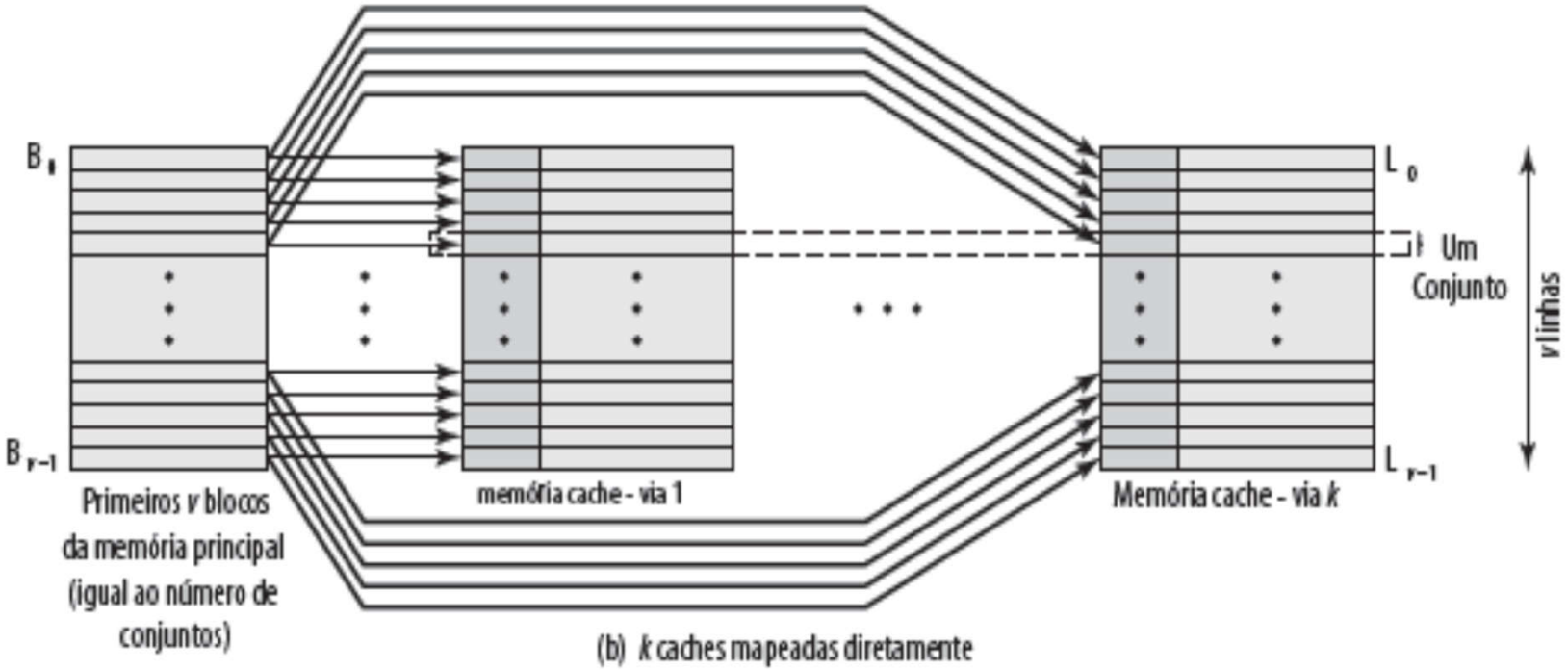
Mapeamento da memória principal para cache: associativo com *v* linhas



(a) *v* caches mapeadas associativas

Memória Cache

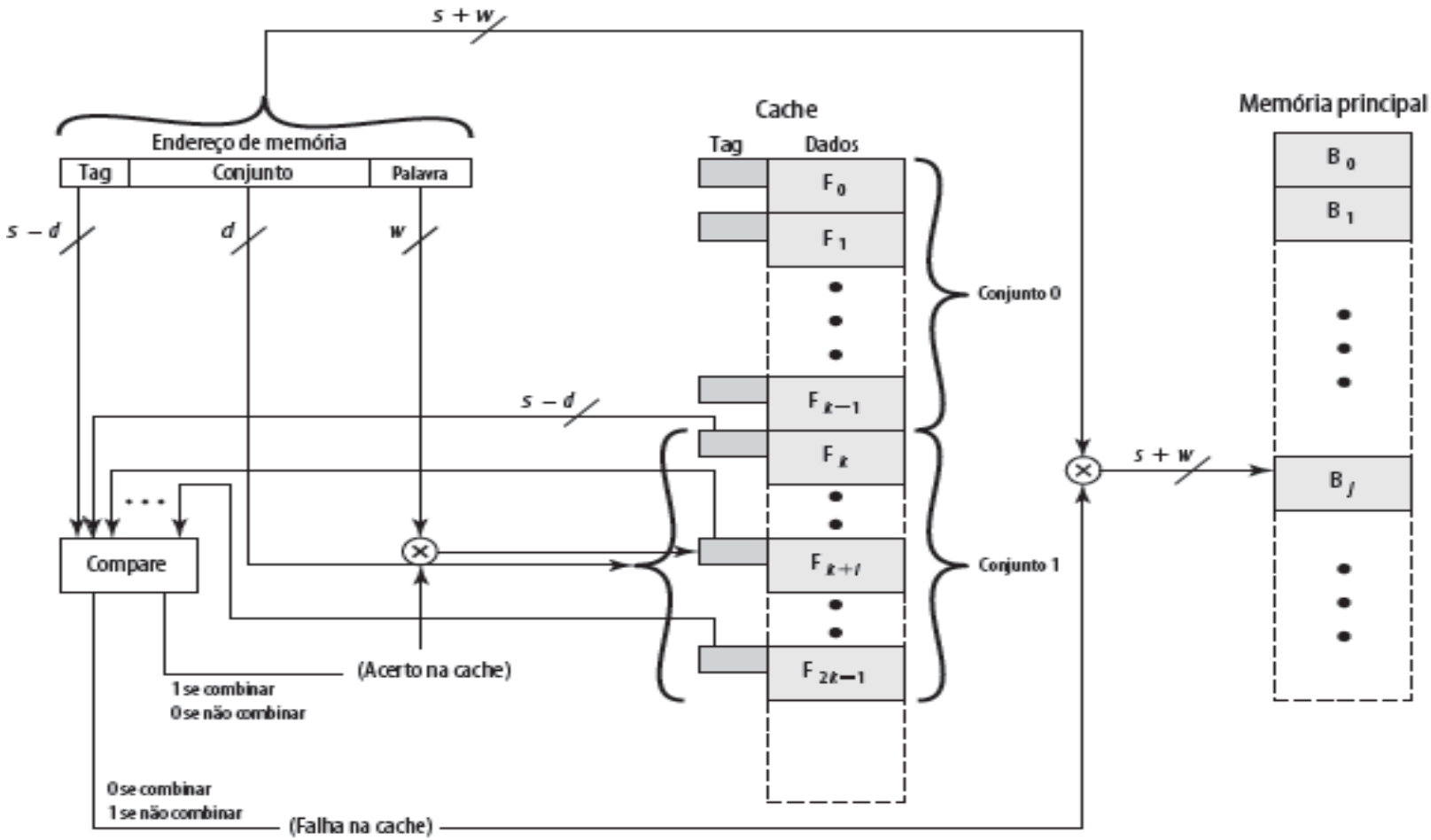
Mapeamento da memória principal para cache: associativo com *k* linhas



(b) *k* caches mapeadas diretamente

Memória Cache

Organização da cache associativa em conjunto com *k* linhas



Memória Cache

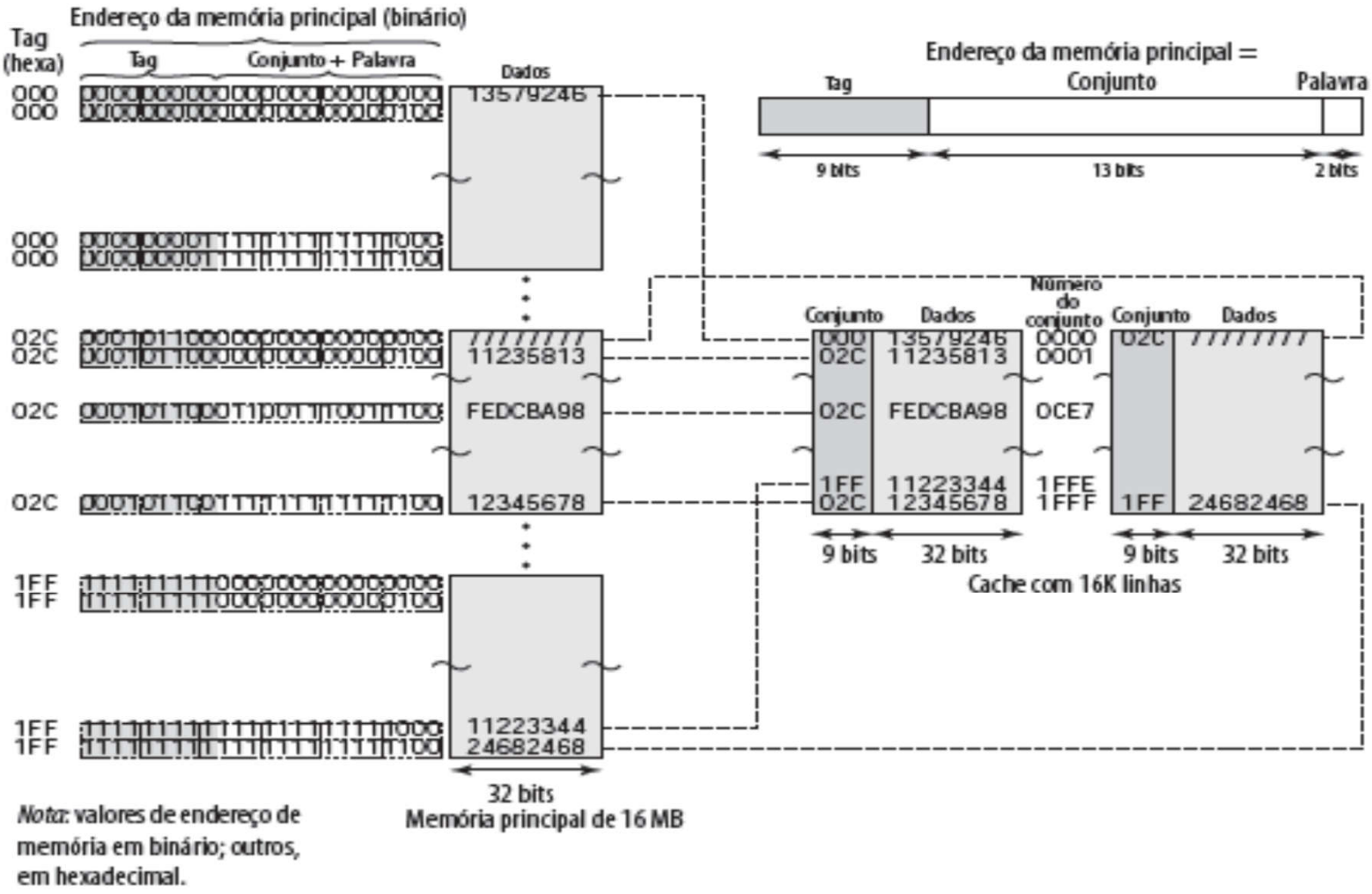
Mapeamento associativo em conjunto
Estrutura de endereços

Tag 9 bits	Conjunto 13 bit	Palavra 2 bit

- Use campo de conjunto para determinar conjunto de cache a examinar.
- Compare campo de tag para ver se temos um acerto.
- P.ex.,
  - Endereço Tag Dados Conjunto
  - 1FF 7FFC 1FF 12345678 1FFF
  - 001 7FFC 001 11223344 1FFF

Memória Cache

Exemplo de mapeamento associativo em conjunto com duas linhas



Memória Cache

Resumo de mapeamento associativo em conjunto

Tamanho do endereço = $(s + w)$ bits

Número de unidades endereçáveis = $2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha = 2^w palavras ou bytes

Número de blocos na memória principal = $\frac{2^{(s+w)}}{2^w} = 2^s$

Número de linhas no conjunto = k

Número de conjuntos = $v = 2^d$

Número de linhas na cache = $m = kv = k\,x\,2^d$

Tamanho da cache = $k\,x\,2^{(d+w)}$ palavras ou bytes

Tamanho da tag = $(s - d)$ bits

Memória Cache

Exercícios

1) Um computador possui uma memória principal com capacidade para 2 Gbits. Cada célula desta memória tem capacidade para 8 bits. Foi colocada neste computador uma memória cache associativa em conjunto com capacidade para 512 Kbytes. Cada linha desta cache tem capacidade para 16 células. Cada conjunto possui 4 linhas. Supondo que a CPU faça um acesso ao endereço $(02A854DB)_{16}$, calcule:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) Total de bits para o número do conjunto	
(d) O total de bits para a TAG	
(e) O número da WORD (em hexadecimal)	
(f) O número do conjunto (em hexadecimal)	
(g) O valor da TAG (em hexadecimal)	

Tamanho do endereço $= (s + w)$ bits

Número de unidades endereçáveis $= 2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha $= 2^{w}$ palavras ou bytes

Número de blocos na memória principal $= \frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas no conjunto $= k$

Número de conjuntos $= v = 2^{d}$

Número de linhas na cache $= m = kv = k\,x\,2^{d}$

Tamanho da cache $= k\,x\,2^{(d+w)}$ palavras ou bytes

Tamanho da tag $= (s - d)$ bits

Memória Cache

Exercícios

2) Um computador possui uma memória principal com capacidade para 16 Gbits. O Barramento de Endereços deste computador possui 30 bits. Foi colocado nele uma memória cache associativa por conjunto com capacidade para 1 Mbytes. Cada linha desta cache tem capacidade para 512 bits. Cada conjunto tem capacidade para 128 células. Supondo que a CPU faça um acesso ao endereço (0367 4AED)16, Calcule:

	Resposta
(a) Total de bits do endereço	
(b) Total de bits para a WORD	
(c) Total de bits para o número do conjunto	
(d) O total de bits para a TAG	
(e) O número da WORD (em hexadecimal)	
(f) O número do conjunto (em hexadecimal)	
(g) O valor da TAG (em hexadecimal)	

Tamanho do endereço $= (s + w)$ bits

Número de unidades endereçáveis $= 2^{(s+w)}$ palavras ou bytes

Tamanho do bloco = tamanho da linha $= 2^{w}$ palavras ou bytes

Número de blocos na memória principal $= \frac{2^{(s+w)}}{2^{w}} = 2^{s}$

Número de linhas no conjunto $= k$

Número de conjuntos $= v = 2^{d}$

Número de linhas na cache $= m = kv = k\,x\,2^{d}$

Tamanho da cache $= k\,x\,2^{(d+w)}$ palavras ou bytes

Tamanho da tag $= (s - d)$ bits