Organização e Arquitetura de Computadores

Aula 05

Memória Cache

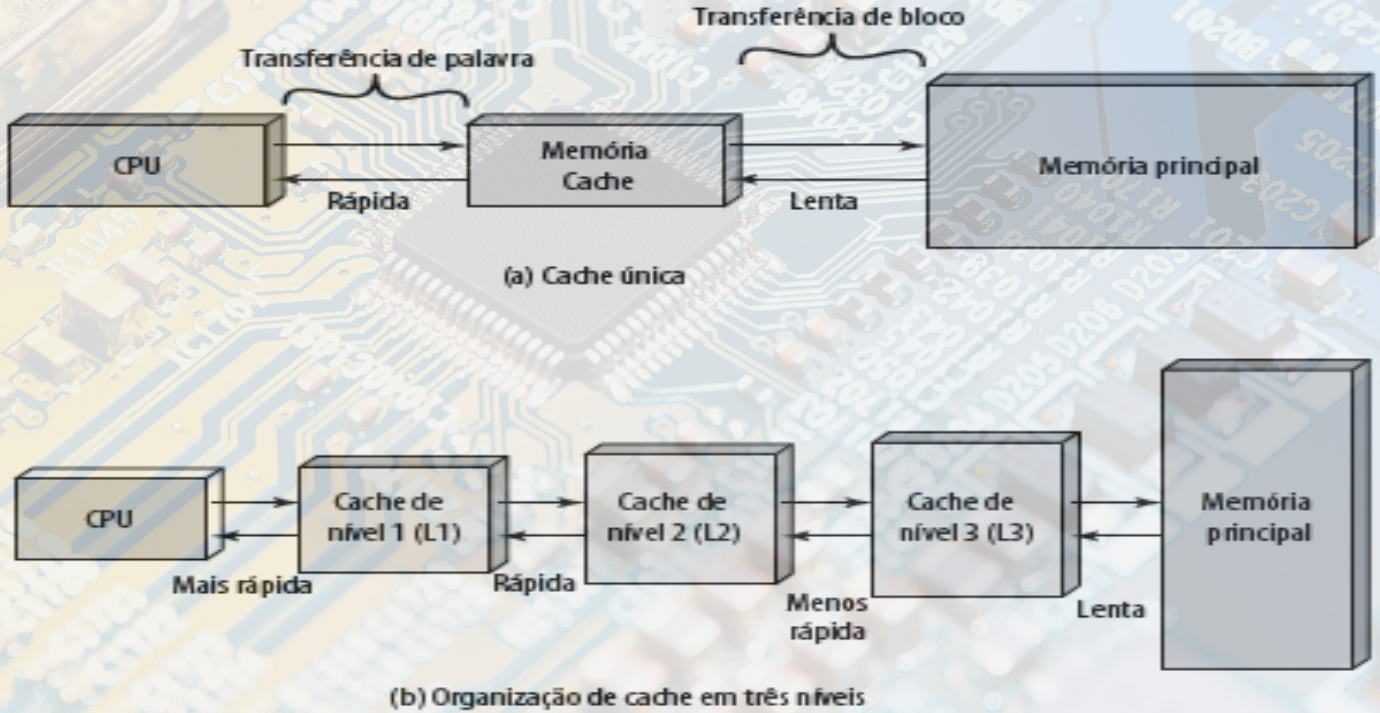
Prof. Msc. Francisco de Assis de Souza Rodrigues

UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO

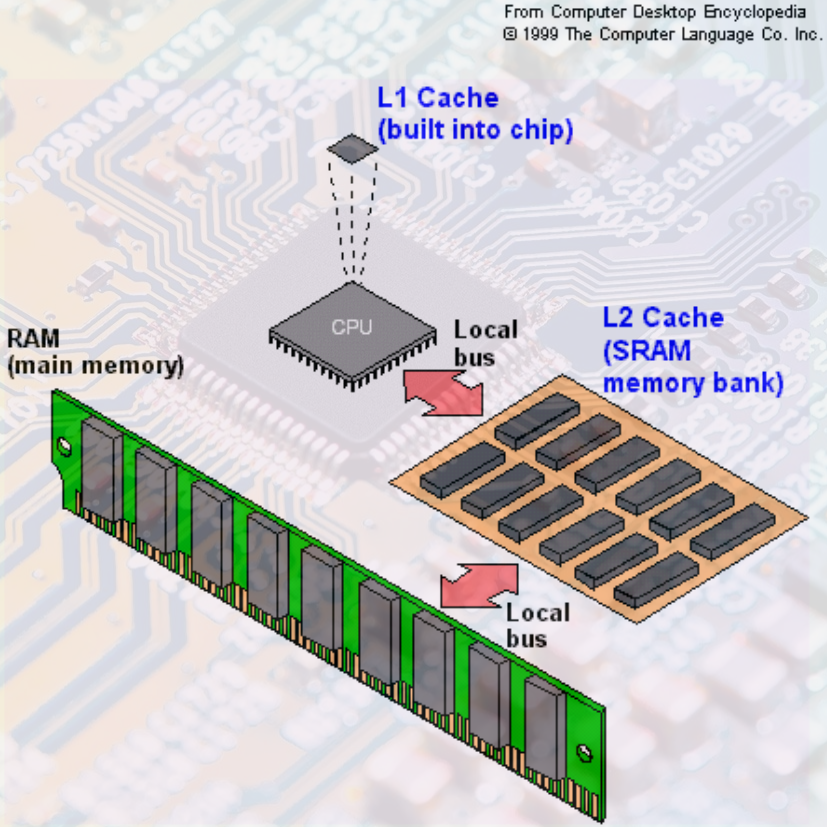
Cache

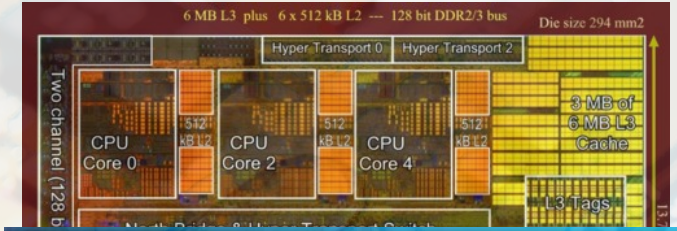
- Pequena quantidade de memória rápida.
- Fica entre a memória principal normal e a CPU.
- Pode estar localizada no chip da CPU ou módulo.

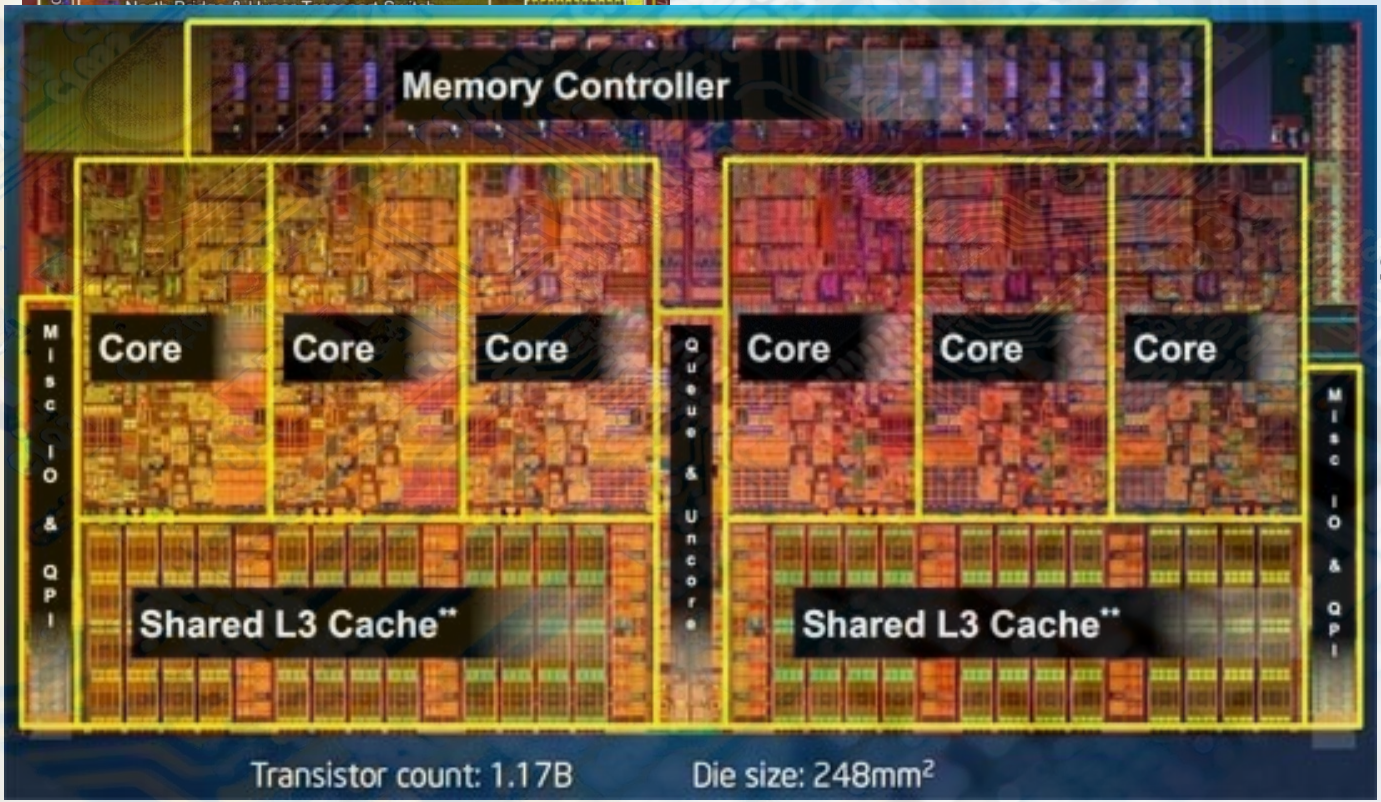
Cache e memória principal



Cache e memória principal

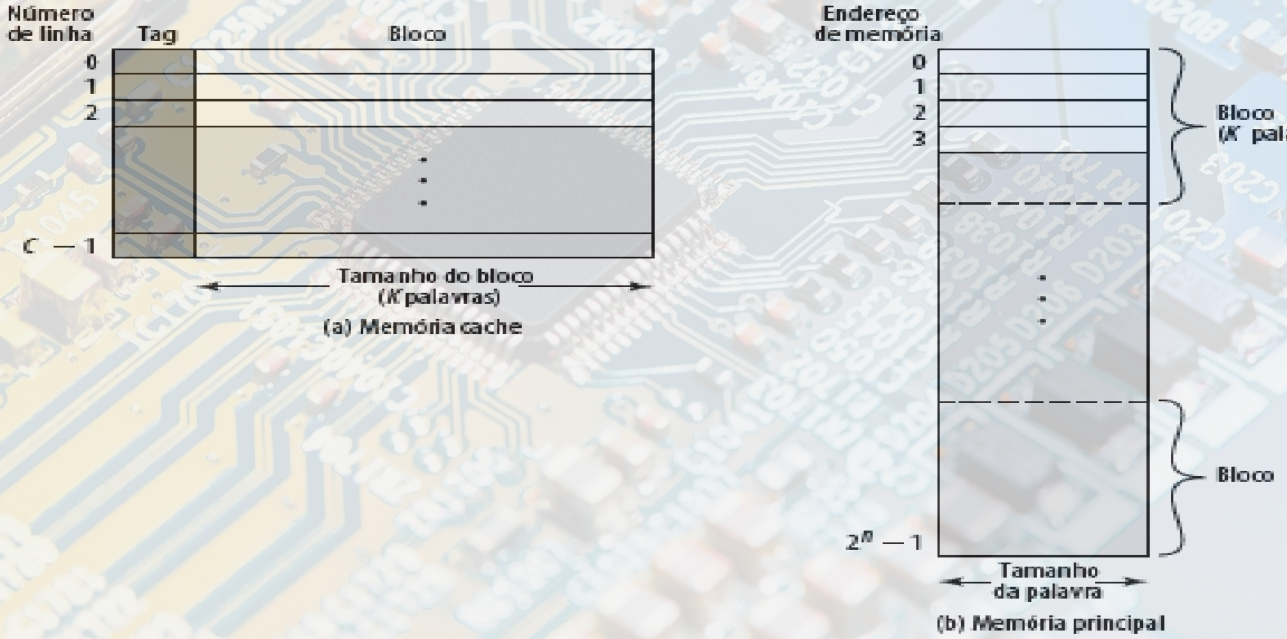






sador Core i7

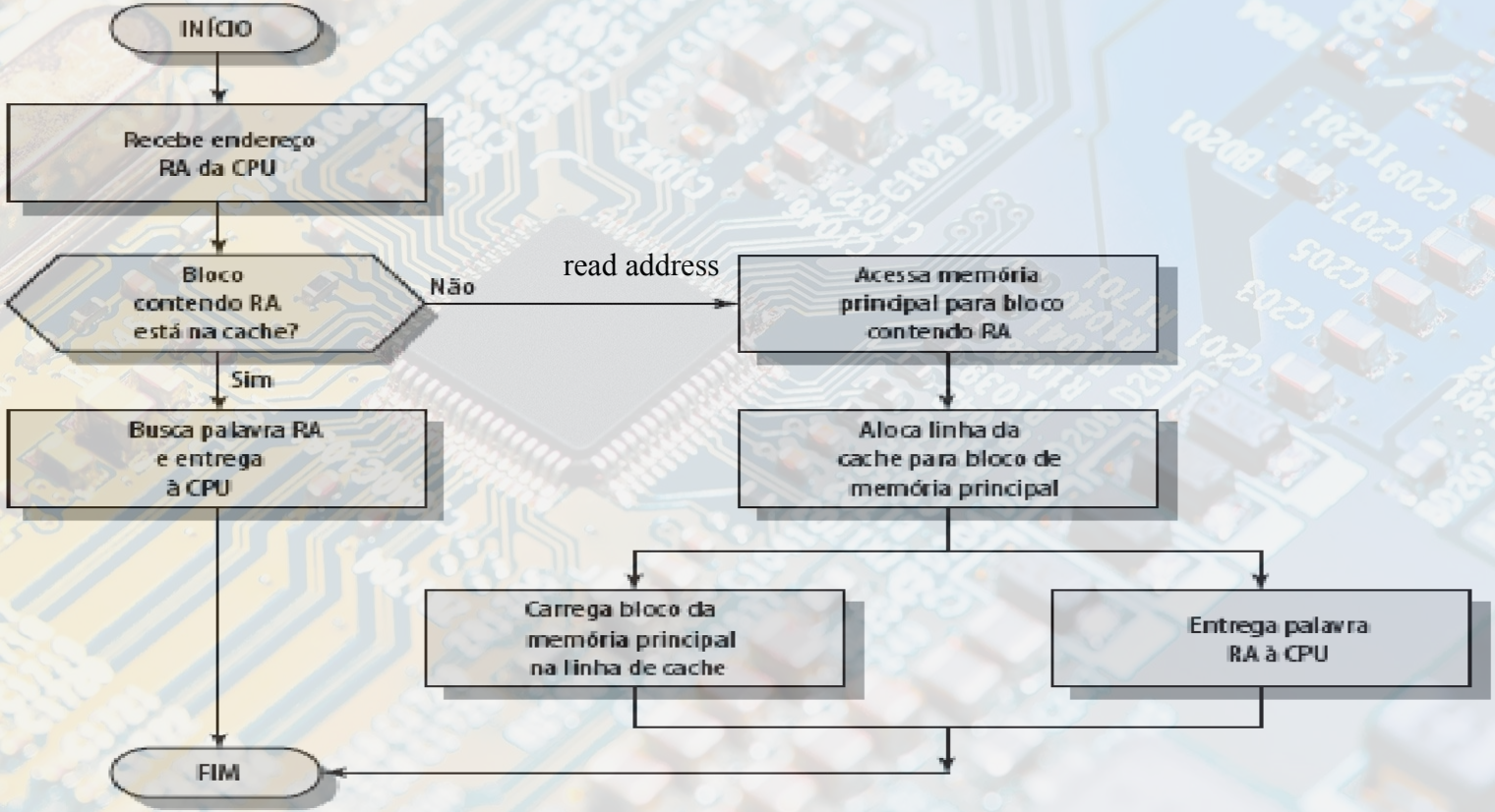
Estrutura de cache/memória principal



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.

Operação de leitura de cache – fluxograma



Projeto de cache

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

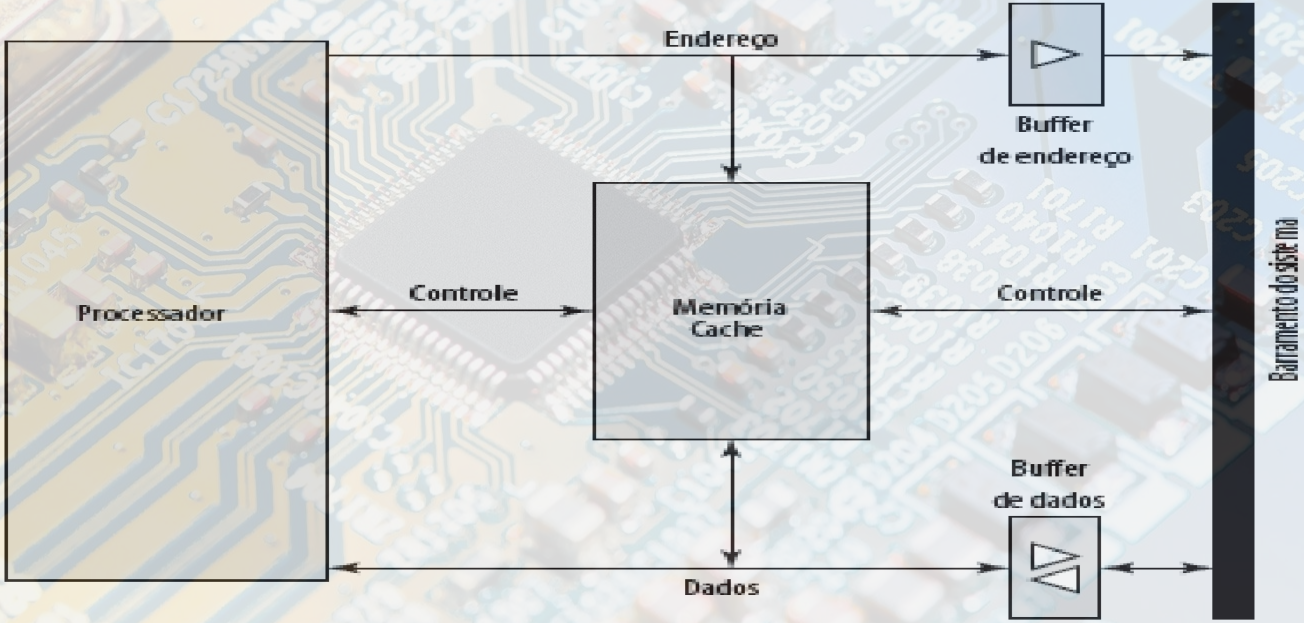
Endereçamento de cache

- 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

Tamanho não importa

- Custo:
  - Mais cache é caro.
- Velocidade:
  - Mais cache é mais rápido (até certo ponto).
  - Verificar dados na cache leva tempo.

Organização típica da memória cache



Comparação de tamanhos de memória cache

Processador	Tipo	Ano de introdução	Cache L1[a]	Cache L2	Cache L3
IBM 360/85	Mainframe	1968	16 a 32 KB	—	—
PDP-11/70	Minicomputador	1975	1 KB	—	—
VAX 11/780	Minicomputador	1978	16 KB	—	—
IBM 3033	Mainframe	1978	64 KB	—	—
IBM 3090	Mainframe	1985	128 a 256 KB	—	—
Intel 80486	PC	1989	8 KB	—	—
Pentium	PC	1993	8 KB/8 KB	256 a 512 KB	—
PowerPC 601	PC	1993	32 KB	—	—
PowerPC 620	PC	1996	32 KB/32 KB	—	—
PowerPC G4	PC/servidor	1999	32 KB/32 KB	256 KB a 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/servidor	2000	8 KB/8 KB	256 KB	—
IBM SP	Servidor avançado/ Supercomputador	2000	64 KB/32 KB	8 MB	—
CRAY MTA[b]	Supercomputador	2000	8 KB	2 MB	—
Itanium	PC/servidor	2001	16 KB/16 KB	96 KB	4 MB
SGI Origin 2001	Servidor avançado	2001	32 KB/32 KB	4 MB	—
Itanium 2	PC/servidor	2002	32 KB	256 KB	6 MB
IBM POWER5	Servidor avançado	2003	64 KB	1,9 MB	36 MB
CRAY XD-1	Supercomputador	2004	64 KB/64 KB	1 MB	—
IBM POWER6	PC/servidor	2007	64 KB/64 KB	4 MB	32 MB
IBM z10	Mainframe	2008	64 KB/128 KB	3 MB	24 a 48 MB

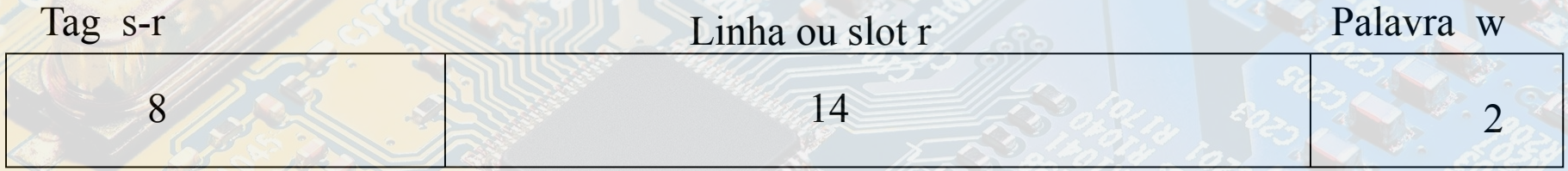
Função de mapeamento

- 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)

Mapeamento direto

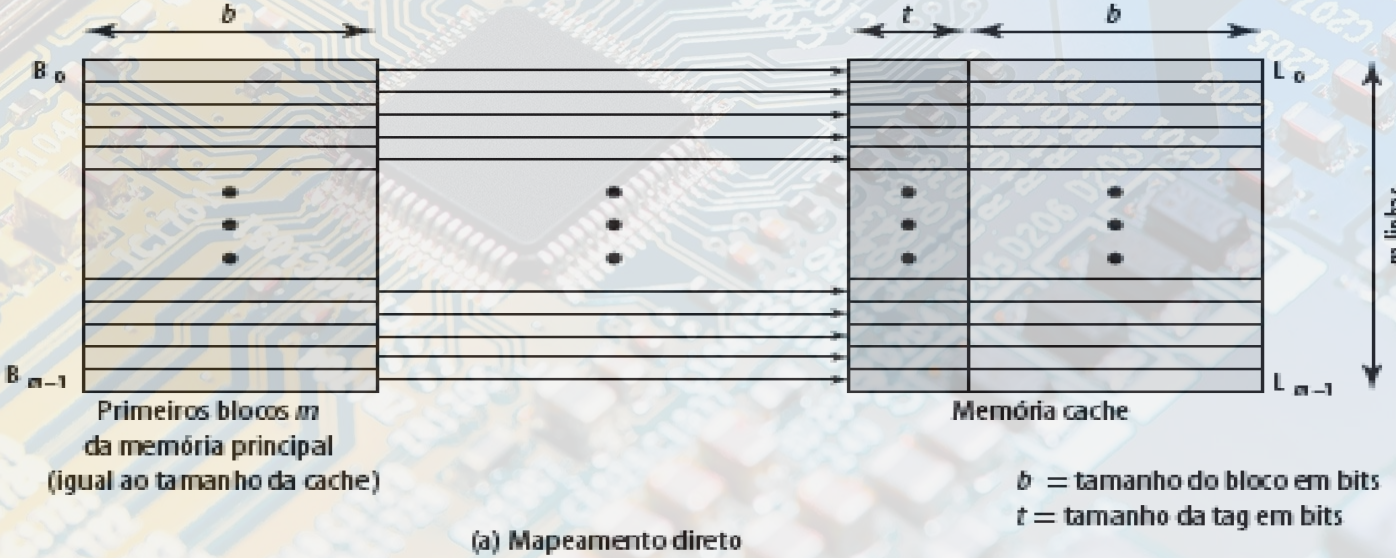
- Cada bloco de memória principal mapeado apenas para uma linha de cache.
  - Ou seja, se um bloco está na cache, ele deve estar em um local específico.
- Endereço está em duas partes.
- W bits menos significativos identificam word exclusiva.
- S bits mais significativos especificam um bloco de memória.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

Mapeamento direto
Estrutura de endereços



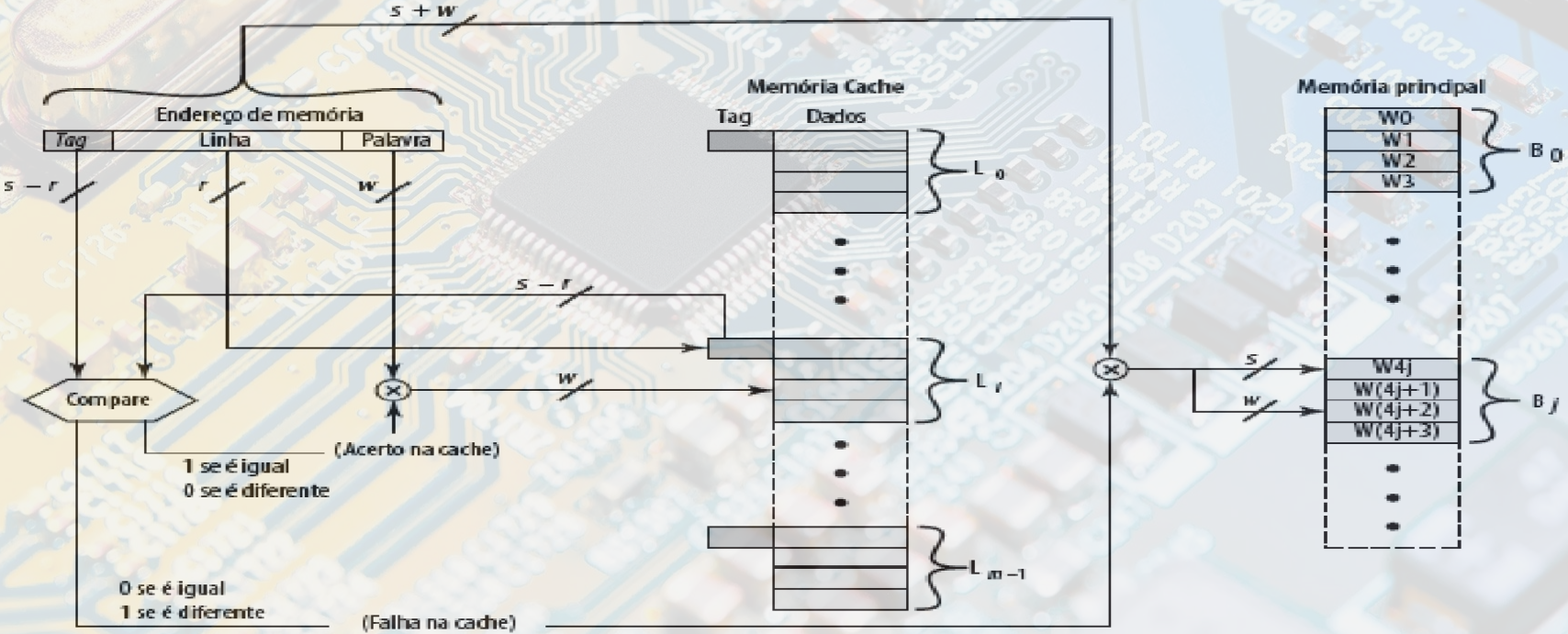
- Endereço de 24 bits.
- Identificador de palavra de 2 bits (bloco de 4 bytes).
- Identificador de bloco de 22 bits.
  - Tag de 8 bits (=22-14).
  - Slot ou linha de 14 bits.
- Dois blocos na mesma linha não têm o mesmo campo de tag.

Mapeamento direto da cache para memória principal



(a) Mapeamento direto

Organização da cache com mapeamento direto



Resumo de mapeamento direto

- Tamanho de endereço= (s + w) bits.
- Número de unidades endereçáveis= 2s+w palavras ou bytes.
- Tamanho de bloco= tamanho de linha= 2w words ou bytes.
- Número de blocos na memória principal= 2s+ w/2w = 2s.
- Número de linhas na cache= m= 2r.
- Tamanho da tag= (s – r) bits.

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.

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.

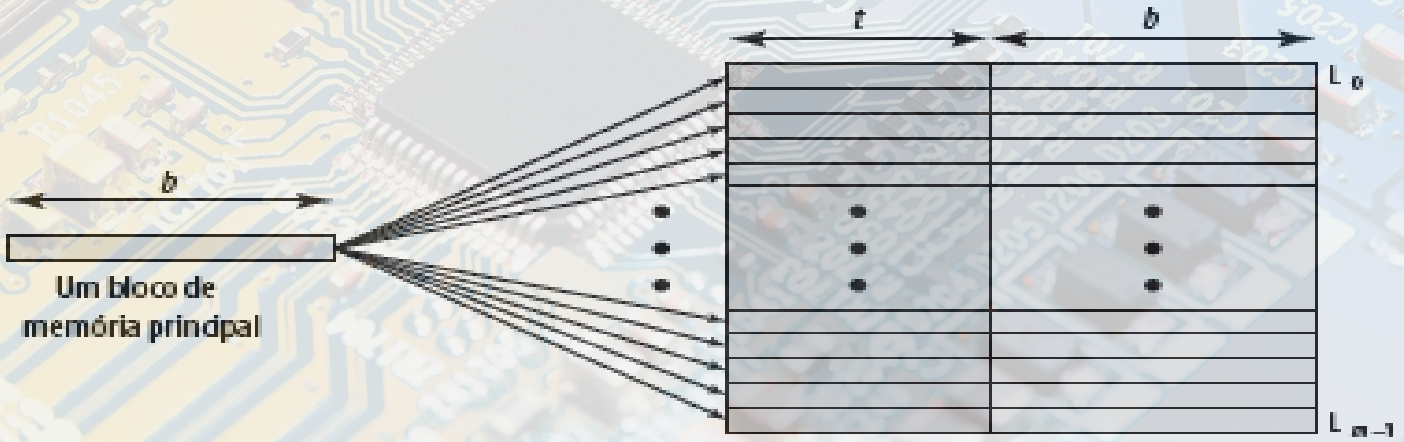
Mapeamento associativo

Estrutura de endereço



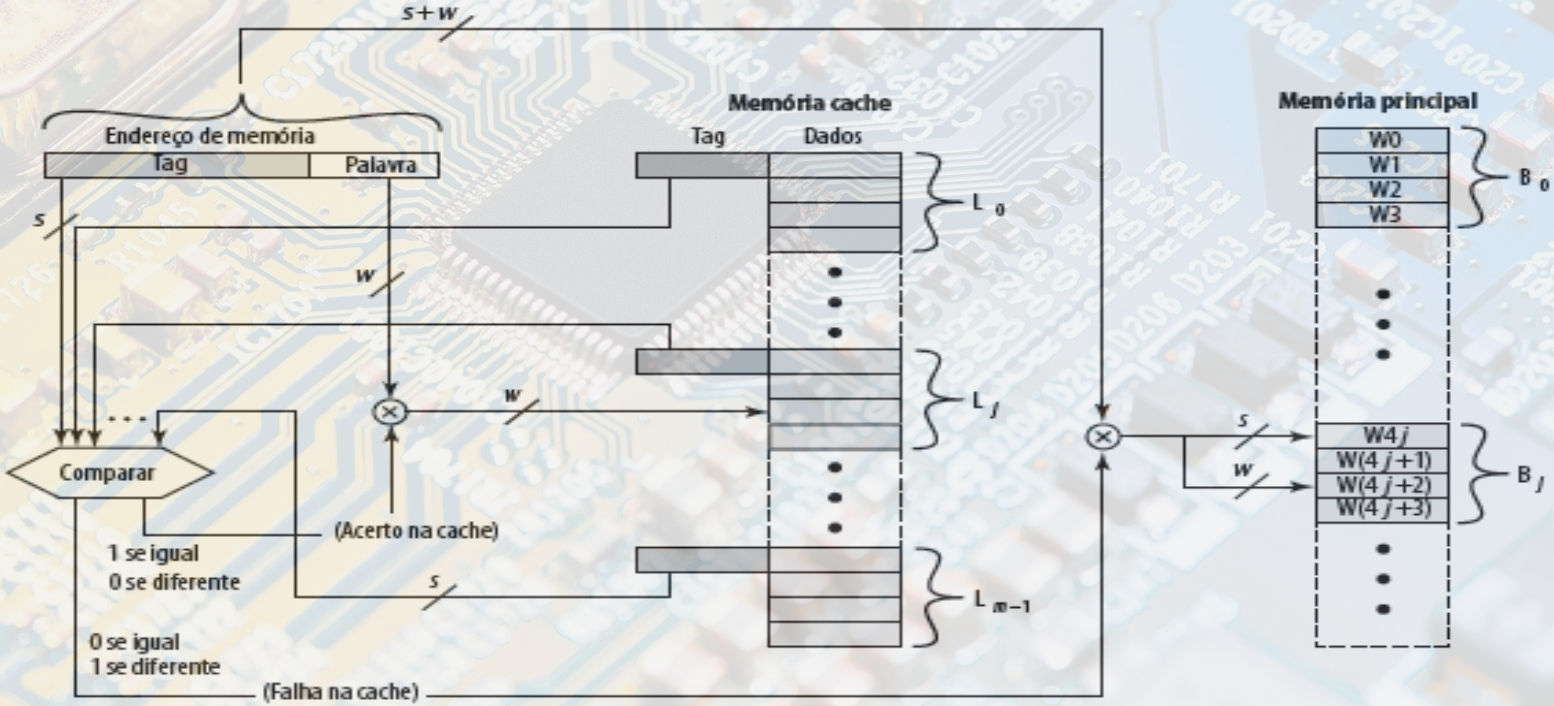
- Tag de 22 bits armazenado a cada bloco de 32 bits de dados.
- Compara campo de tag com entrada de tag na cache para procurar acerto.
- 2 bits menos significativos do endereço identificam qual word de 16 bits é exigida do bloco de dados de 32 bits.
- P.e.:
  - Endereço Tag Dados Linha de cache
  - FFFFFC FFFFFC 24682468 3FFF

Mapeamento associativo da cache para a memória principal



(b) Mapeamento associativo

Organização de cache totalmente associativa



Resumo do mapeamento associativo

- Tamanho do endereço= (s + w) bits.
- Número de unidades endereçáveis= 2s+w words ou bytes.
- Tamanho do bloco= tamanho de linha= 2w palavras ou bytes.
- Número de blocos na memória principal= 2s+ w/2w = 2s.
- Número de linhas na cache= indeterminado.
- Tamanho da tag= s bits.

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.e.,Bloco B pode estar em qualquer linha do conjunto i.
- P.e., 2 linhas por conjunto:
  - Mapeamento associativo com 2 linhas.
  - Determinado bloco pode estar em uma de 2 linhas

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

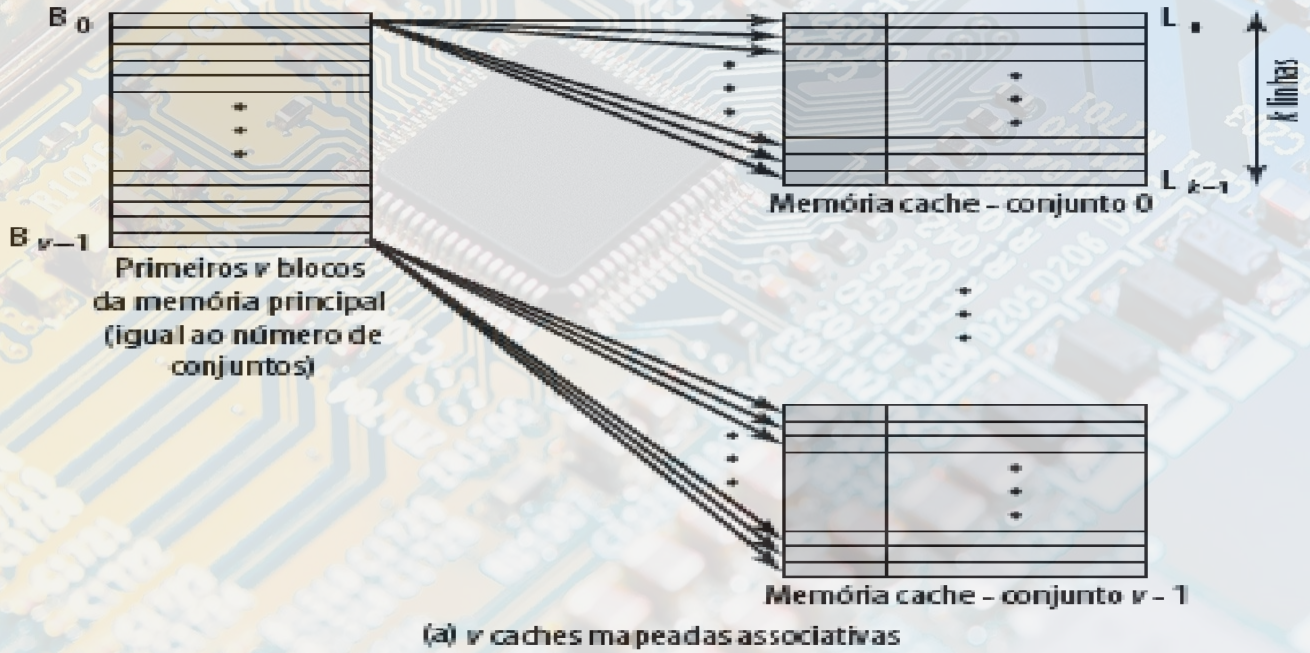
Mapeamento associativo em conjunto

Estrutura de endereços



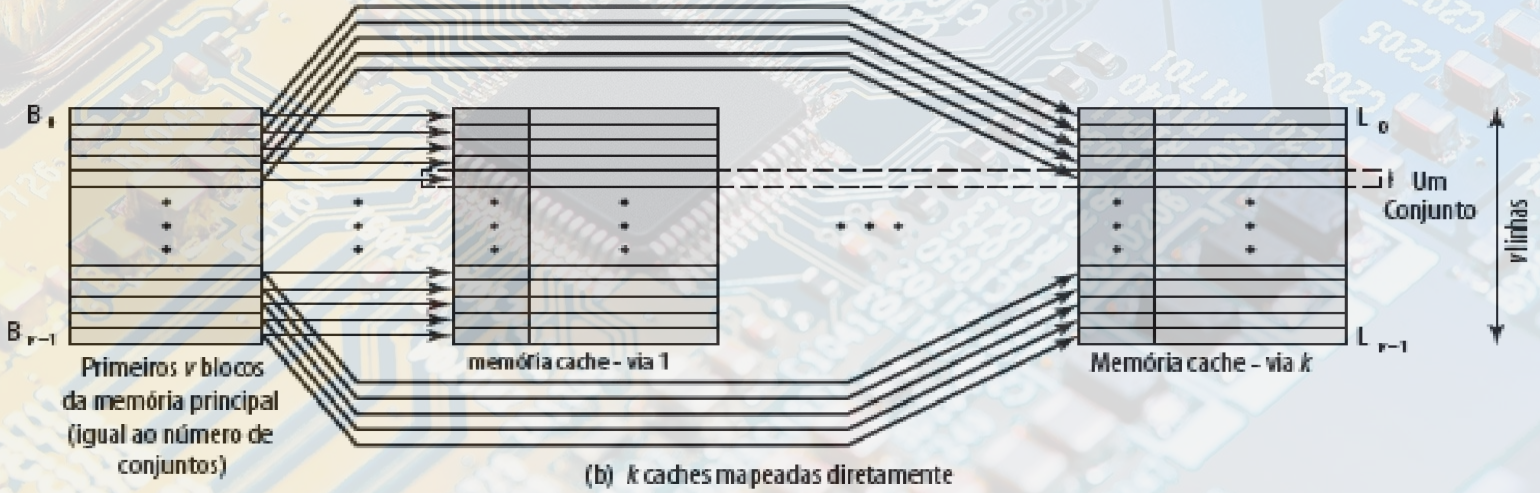
- Use campo de conjunto para determinar conjunto de cache a examinar.
- Compare campo de tag para ver se temos um acerto.
- P.e.,
  - Endereço Tag Dados Conjunto
  - 1FF 7FFC 1FF 12345678 1FFF

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



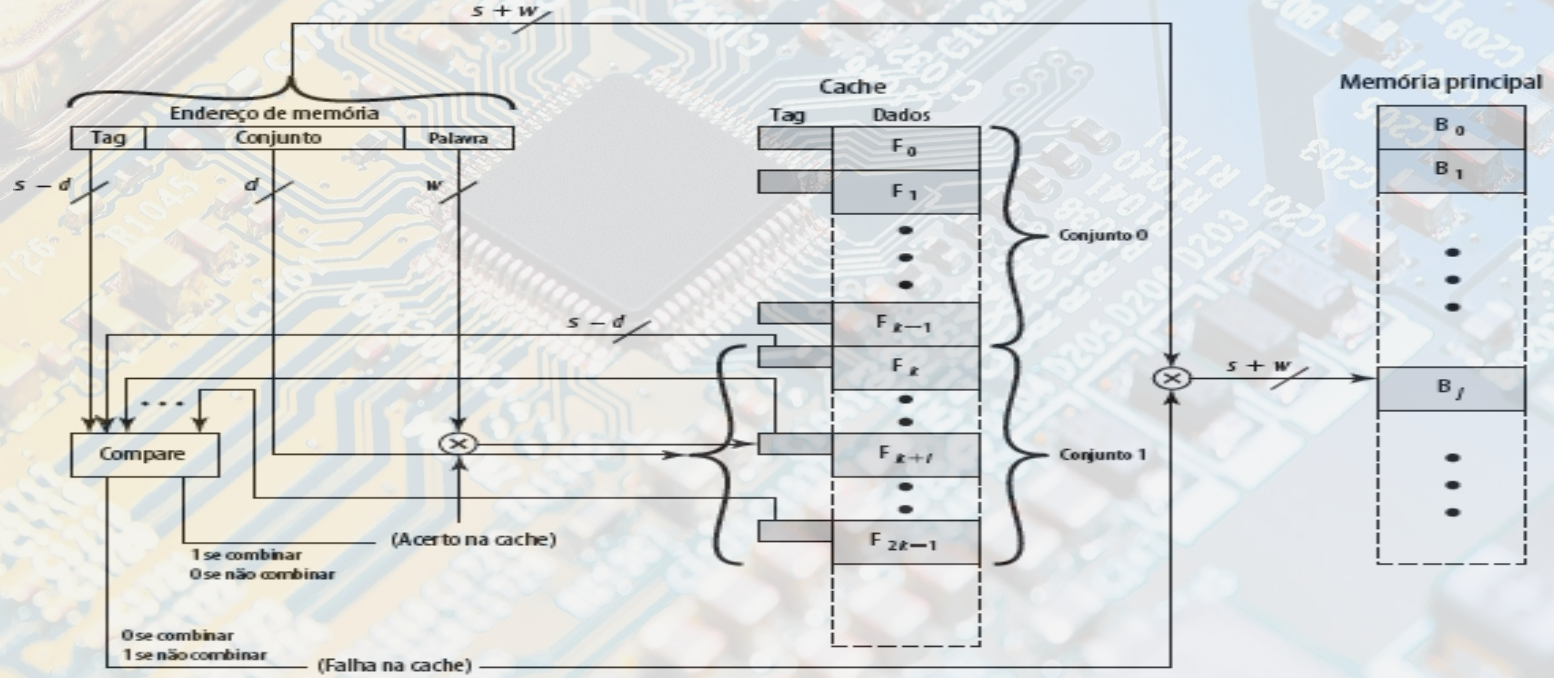
(a) *v* caches mapeadas associativas

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



(b) *k* caches mapeadas diretamente

Organização da cache associativa em conjunto com *k* linhas



Resumo de mapeamento associativo em conjunto

- Tamanho do endereço= (s + w) bits.
- Número de unidades endereçáveis= 2s+w palavras ou bytes.
- Tamanho do bloco= tamanho da linha= 2w palavras ou bytes.
- Número de blocos na memória principal= 2d.
- Número de linhas no conjunto= k.
- Número de conjuntos= v= 2d.
- Número de linhas na cache= kv= k * 2d.

Cache direta e associativa em conjunto

Diferenças de desempenho

- Significativo até pelo menos 64kB para 2 linhas.
- Diferença entre 2 e 4 linhas em 4 KB muito menor que 4 KB para 8 KB.
- Complexidade da cache aumenta com a associatividade.
- Não justificado contra aumento da cache para 8kB ou 16kB.
- Acima de 32 KB não gera melhoria.

Algoritmos de substituição / Politica de Escrita /

Algoritmos de substituição

Mapeamento direto

- Sem escolha.
- Cada bloco mapeado apenas a uma linha.
- Substitui essa linha.

Algoritmos de substituição

Associativa e associativa em conjunto

- Algoritmo implementado no hardware (velocidade).
- Least Recently Used (LRU).
- P.e., na associativa em conjunto com 2 linhas.
  - Qual dos 2 blocos é LRU?
- First In First Out (FIFO).
  - Substitui bloco que está na cache há mais tempo.
- Least Frequently Used (LFU).
  - Substitui bloco que teve menos acertos.

Política de escrita

- Não deve sobrescrever bloco de cache a menos que a memória principal esteja atualizada.
- Múltiplas CPUs podem ter caches individuais.
- E/S pode endereçar memória principal diretamente.

Write-through

- Todas as escritas vão para a memória principal e também para a cache.
- Múltiplas CPUs podem monitorar o tráfego da memória principal para manter a cache local (à CPU) atualizada.
- Muito tráfego.
- Atrasa as escritas.
- Lembre-se de caches *write-through* falsos!

Write-back

- Atualizações feitas inicialmente apenas na cache.
- Bit de atualização para slot de cache é definido quando ocorre a atualização.
- Se o bloco deve ser substituído, escreve na memória principal apenas se o bit atualizado estiver marcado.
- Outras caches saem de sincronismo.
- E/S deve acessar a memória principal através da cache.
- 15% das referências de memória são escritas.

Tamanho de linha

- Recupere não apenas a palavra desejada, mas também uma série de palavras adjacentes.
- Tamanho de bloco aumentado aumentará razão de acerto a princípio.
  - O princípio da localidade.
- Razão de acerto diminuirá à medida que o bloco se tornar ainda maior.
  - Probabilidade de uso de informações recém-buscadas torna-se menor que probabilidade de reutilizar informações substituídas.

- Blocos maiores:
  - Reduzem número de blocos que cabem na cache.
  - Dados sobrescritos pouco depois de serem buscados.
  - Cada palavra adicional é menos local, de modo que é menos provável de ser necessária.
- Nenhum valor ideal definitivo foi descoberto.
- 8 a 64 bytes parece ser razoável.
- Para sistemas HP, 64 e 128 bytes mais comum.

Caches multinível

- Alta densidade lógica permite caches no chip.
  - Mais rápido que acesso ao barramento.
  - Libera barramento para outras transferências.
- Comum usar cache dentro e fora do chip.
  - L1 no chip, L2 fora do chip na RAM estática.
  - Acesso L2 muito mais rápido que DRAM ou ROM.
  - L2 normalmente usa caminho de dados separado.
  - L2 pode agora estar no chip.
  - Resultando em cache L3.

Caches unificadas *versus* separadas

- Uma cache para dados e instruções ou duas, uma para dados e uma para instruções.
- Vantagens da cache unificada:
  - Maior taxa de acerto.
    - Equilibra carga entre buscas de instrução e dados.
    - Apenas uma cache para projetar e implementar.
- Vantagens da cache separada:
  - Elimina disputa pela cache entre a unidade de busca/decodificação de instrução e a unidade de execução.

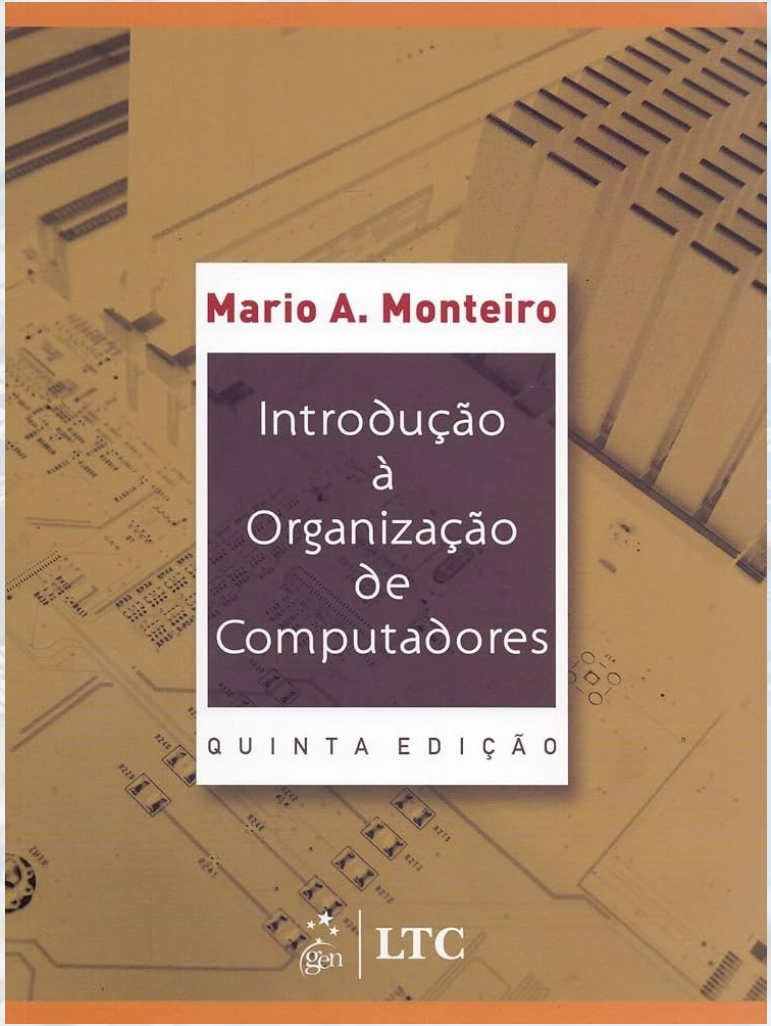
Pentium 4 – cache

- 80386 – nenhuma cache no chip.
- 80486 – 8 KB usando linhas de 16 bytes organização associativa em conjunto com 4 linhas.
- Pentium (todas as versões) – duas caches L1 no chip.
  - Dados e instruções:
- Pentium III – cache L3 adicionada fora do chip.
- Pentium 4:
  - Caches L1.
    - 8 KB.
    - Linhas 64 bytes.

- Cache L2:
  - Alimentando ambas as caches L1.
  - 256k.
  - Linhas de 128 bytes.
  - Associativa em conjunto com 8 linhas.
- Cache L3 cache no chip.

Bibliografia

MONTEIRO, M. A. Introdução à organização de computadores. 5 ed. Rio de Janeiro: LTC, 2007.



UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO

Bibliografia

HENESSY, J., PATTERSON, D.A. Arquitetura de computadores: uma abordagem quantitativa. 4 ed. Rio de janeiro: Campus, 2008.



UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO

Bibliografia

STALLINGS, W. Arquitetura e organização de computadores. 5 ed. São Paulo: Prentice Hall, 2002.

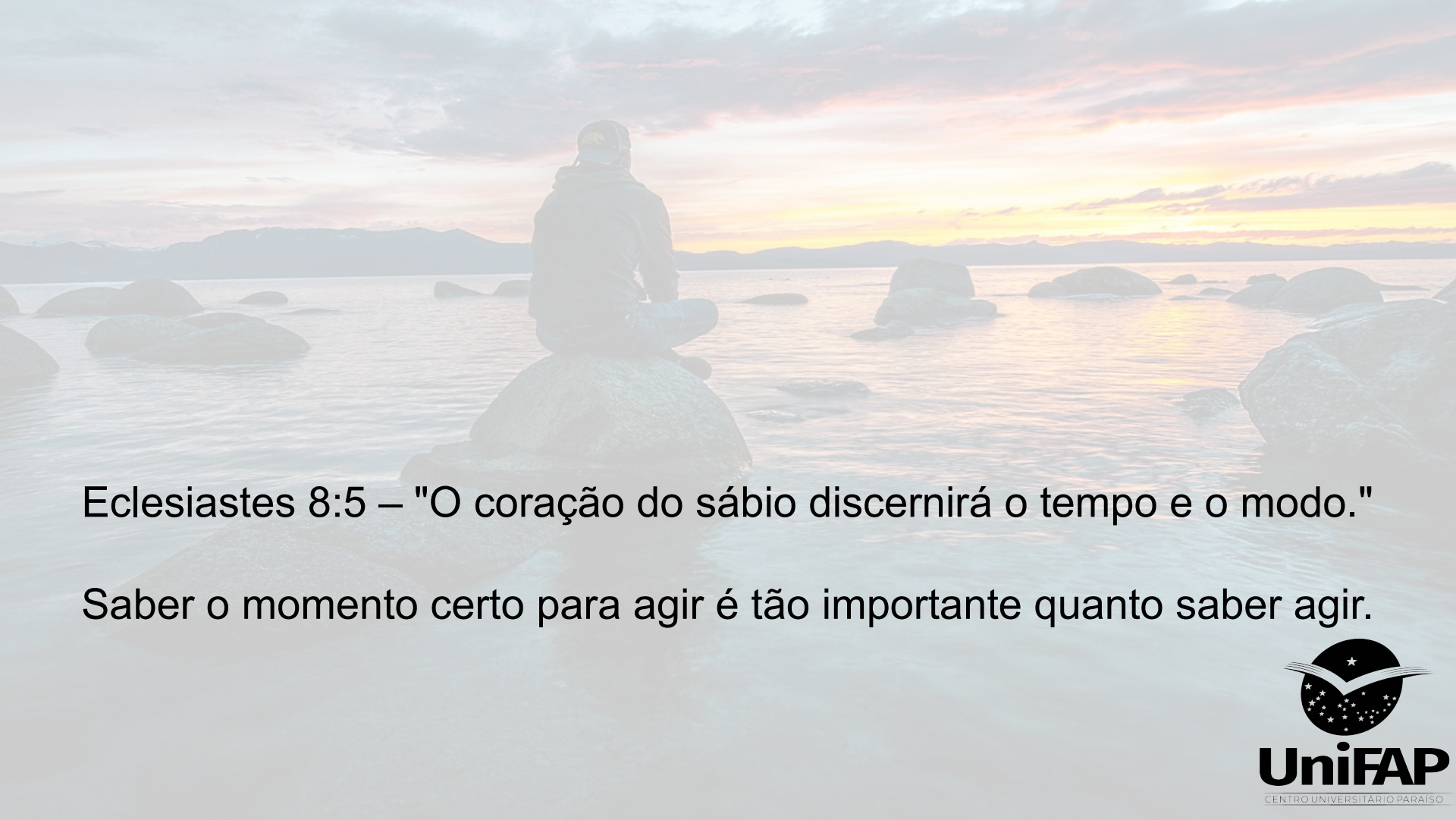


UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO

Perguntas?!



UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO



Eclesiastes 8:5 – "O coração do sábio discernirá o tempo e o modo."

Saber o momento certo para agir é tão importante quanto saber agir.

UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO



Obrigado!

UniFAP
CENTRO UNIVERSITÁRIO PARAÍSO