Organização de Computadores I
DCC006

Aula 12 – Memória (Continuação)

Prof. Omar Paranaiba Vilela Neto

UFMG - ICEx
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Caches Associativas

- Totalmente associativa
  - Permite que um bloco vá para qualquer local na cache
  - Tem que buscar tudo de uma só vez
    - Um comparador por entrada (caro)

Caches Associativas

- Associativa por conjunto (*n-ways*)
  - Cada conjunto contém n entradas
  - Número do bloco determina o conjunto
    - (número bloco) módulo (#conjuntos na cache)
  - Busca todos de um mesmo conjunto
    - *n* comparadores (menos caro)

Espectro de associatividade

- Considere uma cache de 8 entradas

One-way set associative (direct mapped)

Block	Tag	Data
0		
1		
2		
3		
4		
5		
6		
7		

Two-way set associative

Set	Tag	Data	Tag	Data
0				
1				
2				
3				

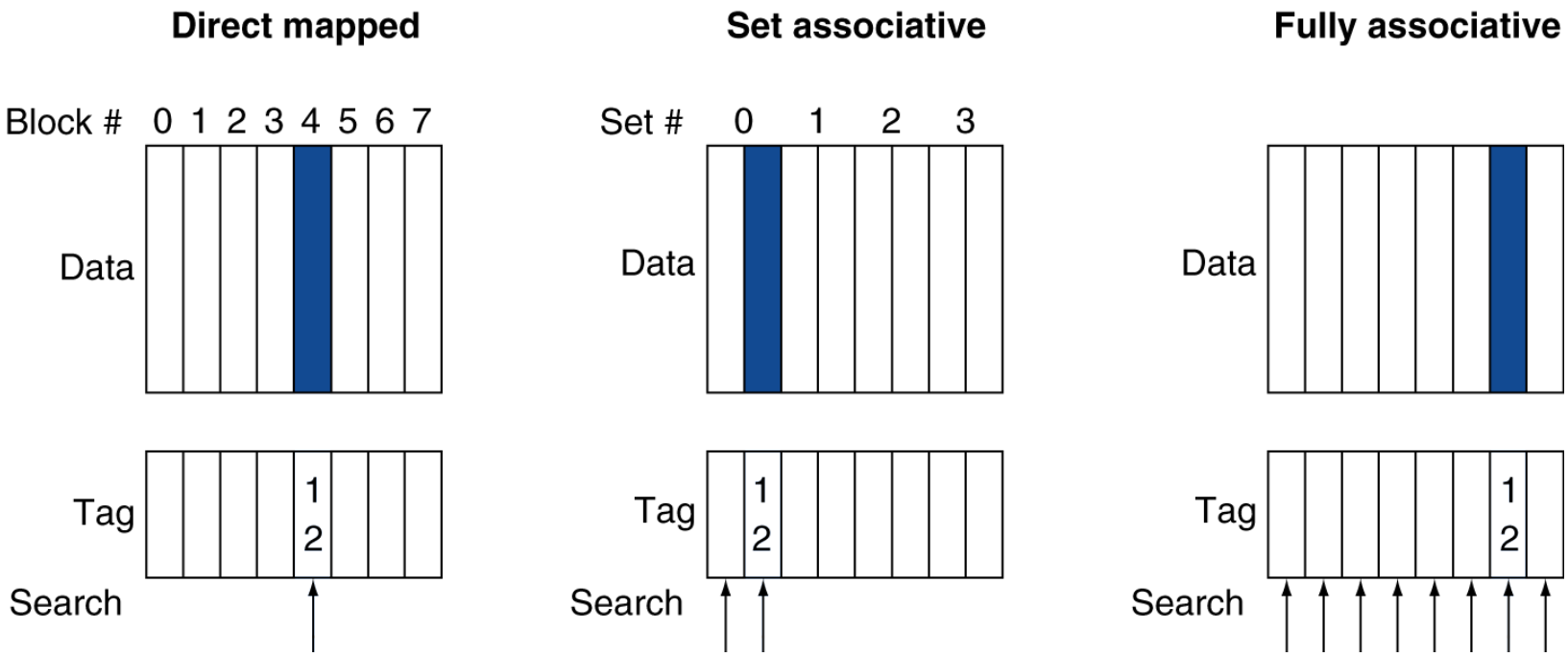
Four-way set associative

Set	Tag	Data	Tag	Data	Tag	Data	Tag	Data
0								
1								

Eight-way set associative (fully associative)

Tag	Data	Tag	Data	Tag	Data	Tag	Data	Tag	Data	Tag	Data	Tag	Data	Tag	Data

Exemplo de Associatividade



Exemplo de Associatividade

- Compare caches de 4 blocos
  - Mapeamento direto, 2-ways associativa por conjunto e totalmente associativa
  - Sequência de acesso aos blocos: 0, 8, 0, 6, 8

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3
0	0	miss	Mem[0]			

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3
0	0	miss	Mem[0]			
8	0	miss	Mem[8]			

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3
0	0	miss	Mem[0]			
8	0	miss	**Mem[8]**			
0	0	miss	**Mem[0]**			

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3
0	0	miss	Mem[0]			
8	0	miss	**Mem[8]**			
0	0	miss	**Mem[0]**			
6	2	miss	Mem[0]		Mem[6]	

Exemplo de Associatividade

- Mapeamento Direto

Block address	Cache index	Hit/miss	Cache content after access			
			0	1	2	3
0	0	miss	**Mem[0]**			
8	0	miss	**Mem[8]**			
0	0	miss	**Mem[0]**			
6	2	miss	Mem[0]		**Mem[6]**	
8	0	miss	**Mem[8]**		Mem[6]	

Exemplo de Associatividade

- 2-way associativa por conjunto

Block address	Cache index	Hit/miss	Cache content after access			
			Set 0		Set 1	
0	0	miss	Mem[0]			

Exemplo de Associatividade

- 2-way associativa por conjunto

Block address	Cache index	Hit/miss	Cache content after access			
			Set 0		Set 1	
0	0	miss	Mem[0]			
8	0	miss	Mem[0]	Mem[8]		

Exemplo de Associatividade

- 2-way associativa por conjunto

Block address	Cache index	Hit/miss	Cache content after access			
			Set 0		Set 1	
0	0	miss	**Mem[0]**			
8	0	miss	Mem[0]	**Mem[8]**		
0	0	hit	**Mem[0]**	Mem[8]		

Exemplo de Associatividade

- 2-way associativa por conjunto

Block address	Cache index	Hit/miss	Cache content after access			
			Set 0		Set 1	
0	0	miss	**Mem[0]**			
8	0	miss	Mem[0]	**Mem[8]**		
0	0	hit	**Mem[0]**	Mem[8]		
6	0	miss	Mem[0]	**Mem[6]**		

Exemplo de Associatividade

- 2-way associativa por conjunto

Block address	Cache index	Hit/miss	Cache content after access			
			Set 0		Set 1	
0	0	miss	**Mem[0]**			
8	0	miss	Mem[0]	**Mem[8]**		
0	0	hit	**Mem[0]**	Mem[8]		
6	0	miss	Mem[0]	**Mem[6]**		
8	0	miss	**Mem[8]**	Mem[6]		

Exemplo de Associatividade

- Totalmente associativa

Block address		Hit/miss	Cache content after access			
0		miss	Mem[0]			

Exemplo de Associatividade

- Totalmente associativa

Block address		Hit/miss	Cache content after access			
0		miss	Mem[0]			
8		miss	Mem[0]	Mem[8]		

Exemplo de Associatividade

- Totalmente associativa

Block address		Hit/miss	Cache content after access			
0		miss	**Mem[0]**			
8		miss	Mem[0]	**Mem[8]**		
0		hit	**Mem[0]**	Mem[8]		

Exemplo de Associatividade

- Totalmente associativa

Block address		Hit/miss	Cache content after access			
0		miss	**Mem[0]**			
8		miss	Mem[0]	**Mem[8]**		
0		hit	**Mem[0]**	Mem[8]		
6		miss	Mem[0]	Mem[8]	**Mem[6]**	

Exemplo de Associatividade

- Totalmente associativa

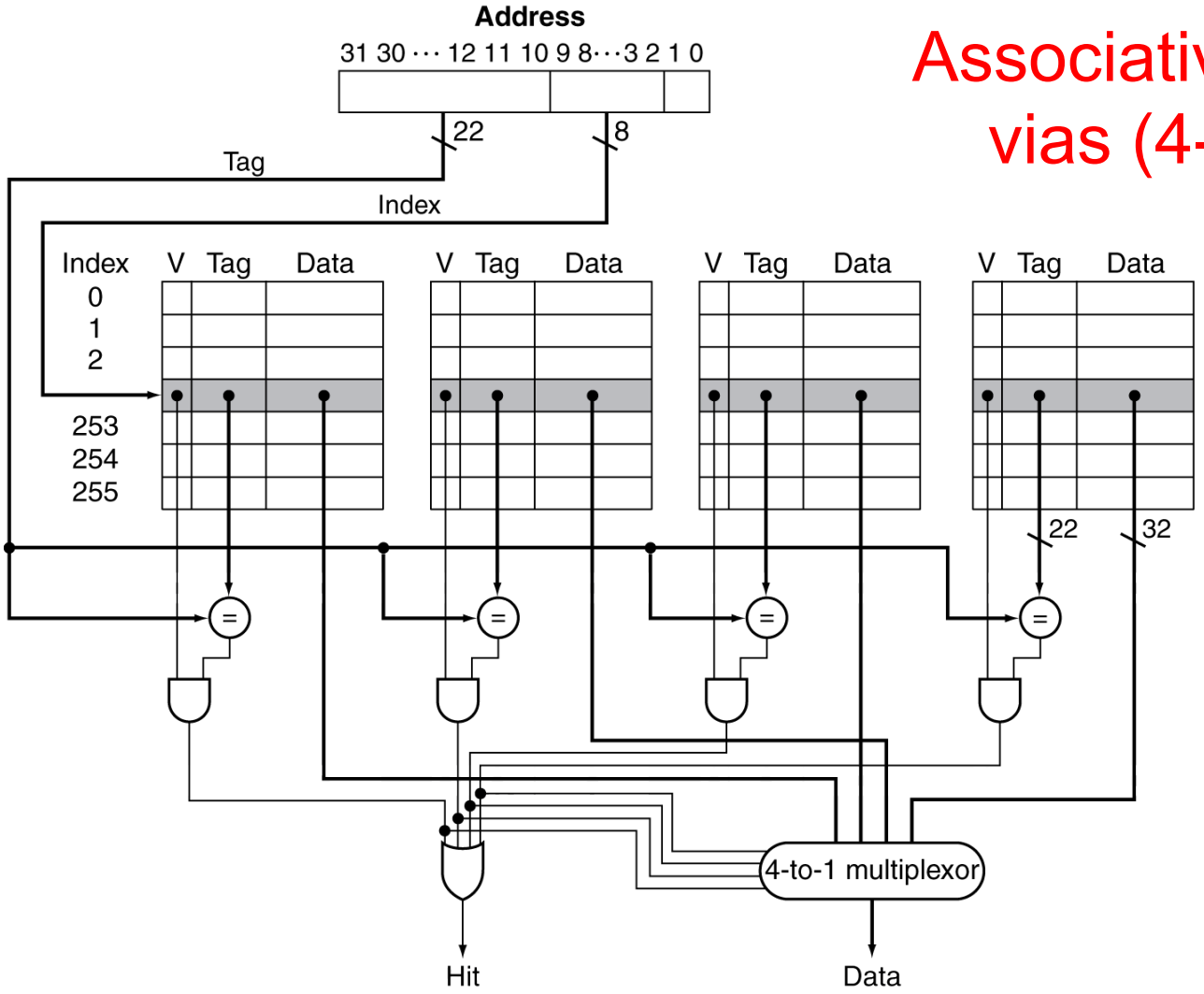
Block address		Hit/miss	Cache content after access			
0		miss	**Mem[0]**			
8		miss	Mem[0]	**Mem[8]**		
0		hit	**Mem[0]**	Mem[8]		
6		miss	Mem[0]	Mem[8]	**Mem[6]**	
8		hit	Mem[0]	**Mem[8]**	Mem[6]	

Quanta associatividade?

- Aumentar a associatividade diminui a taxa de falha
  - Mas com melhorias cada vez menores
- Simulação de um sistema com uma D-cache, 16-words bloco, SPEC2000
  - 1-way: 10.3%
  - 2-way: 8.6%
  - 4-way: 8.3%
  - 8-way: 8.1%

Organização da cache associativa por conjunto

Associativa de 4 vias (4-way)



Política de troca

- Mapeamento direto: sem escolha
- Associativas
  - Prefere entradas vazias, se houver
  - Caso contrário, escolha um para troca
- Least-recently used (LRU)
  - Escolha a usada a menos recente
    - Simples para 2-way, possível for 4-way, muito difícil para outra associatividades
- Aleatório
  - Aproximademtne o mesmo desempenho que LRU para altas associatividades

Escrita na Cache

- Até então focamos na leitura....
- E quando escrevemos na cache???

Write-Through

- No hit de escrita, apenas atualize o bloco na cache
  - Mas a cache e a memória estarão inconsistente
- Write through: também atualiza a memória
- Mas faz a escrita demorar mais
  - Ex. Se CPI = 1, 10% de instruções são store, escrita na memória leva 100 ciclos
    - CPI efetiva = 1 + 0,1 x100 = 11
- Solução: write buffer
  - Guarda dado esperando a ser escrito na memórias
  - CPU continua imediatamente
    - Só para se write buffer estiver cheio

Write-Back

- Alternativa: na escrita, apenas atualiza o bloco na cache
  - Verifique se os blocos estão “sujos”
- Quando um bloco sujo é substituído
  - Escreva na memória
  - Pode usar write buffer para permitir trocar o bloco que vai ser lido antes

Write Allocation

- O que ocorre em um miss de escrita?
- 2 alternativas para write-through
  - Allocate on miss: busca o bloco
  - Write around: não busca o bloco
    - Já que normalmente programas escrevem em um bloco inteiro antes de lê-lo
- Para write-back
  - Normalmente busca o bloco

Medindo desempenho de cache

- Componentes de tempo de CPU
  - Ciclos de execução do programa
    - Incluí tempo de Hit
  - Ciclos de stall de memória
    - Principalmente de cache miss
- Com algumas simplificações:

$$\text{Memory stall cycles} = \frac{\text{Memory accesses}}{\text{Program}} \times \text{Miss rate} \times \text{Miss penalty}$$

$$= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Misses}}{\text{Instruction}} \times \text{Miss penalty}$$

Exemplo de desempenho

- Dado
  - I-cache miss rate = 2%
  - D-cache miss rate = 4%
  - Miss penalty = 100 cycles
  - Base CPI (ideal cache) = 2
  - Load & stores are 36% of instructions
- Ciclos de miss por instrução
  - I-cache: 0.02 × 100 = 2
  - D-cache: 0.36 × 0.04 × 100 = 1.44
- Atual CPI = 2 + 2 + 1.44 = 5.44

Tempo médio da acesso

- Hit time também importante no desempenho
- Average memory access time (AMAT)
  - AMAT = Hit time + Miss rate × Miss penalty
- Example
  - CPU com clock de 1ns, hit time = 1 ciclo, miss penalty = 20 ciclos, I-cache miss rate = 5%
  - AMAT = 1 + 0.05 × 20 = 2ns
    - 2 ciclos por instrução

Resumo de desempenho

- Quando desempenho da CPU aumenta
  - Miss penalty se torna mais importante
- Diminuindo o CPI base
  - Grande proporção do tempo gasto no stall de memória
- Aumentando taxa de clock
  - Stalls de memória gasta mais clocks de CPU
- Não se pode negligenciar o comportamento da cache ao avaliar desempenho

Caches multi-níveis

- Cache primária próxima à CPU
  - Pequena, mas rápida
- Cache Nível-2 fornece dados de misses da cache primária
  - Maior, mais lenta, mas ainda assim mais rápida que memória principal.
- Memória principal serve misse de L2
- Alguns sistemas modernos incluem L3

Caches multi-níveis - Exemplo

- Dado
  - CPU base CPI = 1, clock rate = 4GHz
  - Miss rate/instruction = 2%
  - Main memory access time = 100ns
- Somente com cache primária
  - Miss penalty = 100ns/0.25ns = 400 cycles
  - Effective CPI = 1 + 0.02 × 400 = 9

Exemplo (cont.)

- Adiciona L-2 cache
  - Access time = 5ns
  - Global miss rate to main memory = 0.5%
- Primary miss with L-2 hit
  - Penalty = 5ns/0.25ns = 20 cycles
- CPI = 1 + 0.02 × 20 + 0.005 × 400 = 3.4
- Taxa de melhora = 9/3.4 = 2.6