ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

Computer Architecture (CO2008)

Bài báo cáo chương 5: Bộ nhớ chính, bộ nhớ đệm và hiệu năng bộ nhớ (Lab 8)

Họ và tên	Nguyễn Phúc Tiến
MSSV	2014725
Lớp	L03

Ngày 19 tháng 5 năm 2022

Mục lục

1 Bài 1: Xác định tag, index, offset.

Cho bộ nhớ chính có không gian 32-bit, bộ nhớ cache có kích thước là 4MB, 1 block 256B, đơn vị truy xuất của hệ thống là 1 byte. Xác định tag, index, byte-offset với cấu hình cache sau:

- Số phần tử trong 1 block = (size of block)/(size of phần tử truy xuất) = 256byte = 2^8.
- Số block trong cache = (size of cache) / (size of block) = 4MB / 256byte = $\frac{4 * 2^{20}}{2^8}$ =2^{14} blocks.

(a) Direct mapped: index = 14 bits, offset = 8 bits, tag = 32 - 14 - 8 = 10 bits.

(b) 4-way set associative: 4 blocks tạo thành 1 set mà có 2^{14} blocks nên có 2^{12} set → index = 12 bits, offset = 8 bits, tag = 32 - 12 - 8 = 12 bits.

(c) Fully associative: index = 0 bit, offset = 8 bits, tag = 32 - 8 = 24 bits.

2 Bài 2: Xác định tag, index, haft-word offset

Cho bộ nhớ chính tổng dung lượng là 256M, bộ nhớ cache có kích thước là 256KB, 1 block 64 words, Đơn vị truy xuất của hệ thống là 2 byte. Xác định tag, index, half-word offset với cấu hình cache sau:

- Số phần tử trong 1 block = (size of block)/(size of phần tử truy xuất) = 64words / 2byte = 2^7.
- Số block trong cache = (size of cache) / (size of block) = 256KB / 256byte = $\frac{2^8 * 2^{10}}{2^8}$ =2^{10} blocks.
- Không gian địa chỉ là 256M = 2^{28}, do đó ta dùng thanh ghi 28 bits tính theo byte-offset.

(a) Direct mapped: index = 10 bits, offset = 7 bits, tag = 28 - 10 - 7 = 11 bits.

(b) 4-way set associative: 4 blocks tạo thành 1 set mà có 2^{10} blocks nên có 2^8 set → index = 8 bits, offset = 7 bits, tag = 28 - 8 - 7 = 13 bits.

(c) Fully associative: index = 0 bit, offset = 7 bits, tag = 28 - 7 = 21 bits.

3 Bài 3: Xác định HIT/MISS

Cho dãy địa chỉ (words) sau: 0, 4, 1, 5, 65, 1, 67, 46, 1, 70, 2, 0 Biết hệ thống có 256B caches, 4 words block, đơn vị truy xuất là byte.
Xác định số lần HIT/MISS khi chạy chương trình trên với các cấu hình caches sau:

- Lấy địa chỉ chia cho kích thước của block được kết quả (A) dùng để xác định block trong RAM.
- Lấy kết quả (A) modulo số set được kết quả là index.
- Lấy kết quả (A) chia số set được kết quả là tag.

Số block trong cache = $\frac{256}{4 * 4}$ = 16 blocks

(a) Direct mapped:
Direct map 16 blocks, mỗi block 4 words.
Index = 4 bits, offset = 2 bits, các bit còn lại sẽ là tag.

Word address	Binary address	Tag	Index	Miss/Hit	Giải thích
0	0000 00	0	0	Miss	First access
4	0001 00	0	1	Miss	First access
1	0000 01	0	0	Hit	
5	0001 01	0	1	Hit	
65	1 0000 01	1	0	Miss	Khác tag
1	0000 01	0	0	Miss	Khác tag
67	1 0000 11	1	0	Miss	Khác tag
46	1011 10	1	3	Miss	First access
1	0000 01	0	0	Miss	Khác tag
70	1 0001 10	1	1	Miss	Khác tag
2	0000 10	0	0	Hit	
0	0000 00	0	0	Hit	

⇒ có 4 Hit và 8 Miss

(b) 2-way set associative:
2-way set associative, số set giảm 2 tức index giảm 1 bit

Word address	Binary address	Tag	Index	Miss/Hit	Giải thích
0	000 00	0	0	Miss	First access
4	001 00	0	1	Miss	First access
1	000 01	0	0	Hit	
5	001 01	0	1	Hit	
65	10 000 01	2	0	Miss	Khác tag
1	000 01	0	0	Hit	
67	10 000 11	2	0	Hit	
46	1 011 10	1	3	Miss	First access
1	000 01	0	0	Hit	
70	10 001 10	2	1	Miss	First access
2	000 10	0	0	Hit	
0	000 00	0	0	Hit	

⇒ có 7 Hit và 5 Miss

(c) Fully associative:
Offset = 2 bits, còn lại là tag.

Word address	Binary address	Tag	Miss/Hit	Giải thích
0	000 00	0	Miss	First access
4	001 00	1	Miss	First access
1	000 01	0	Hit	
5	001 01	1	Hit	
65	10000 01	16	Miss	First access
1	000 01	0	Hit	
67	10000 11	16	Hit	
46	1011 10	11	Miss	First access
1	000 01	0	Hit	
70	10001 10	17	Miss	First access
2	000 10	0	Hit	
0	000 00	0	Hit	

⇒ Có 6 Hit và 6 Miss.

4 Bài 4: Tính thời gian truy xuất trung bình (AMAT)

Xác định thời gian truy xuất trung bình(AMAT) ở Bài 3: biết rằng Hit time = 5 cycles, thời gian truy xuất RAM là 10 ns, tần số máy tính là 2Ghz.

f = 2Ghz → T = 1 / f = 0.5ns.
Miss penalty = $\frac{10ns}{0.5ns}$ = 20 cycles.

- Direct mapping: AMAT (cycles) = Hit time + Miss rate * Miss penalty = $5 + \frac{8}{12} * 20 = \frac{55}{3}$ cycles.
  ⇒ AMAT (time) = $\frac{55}{3}$ * 0.5ns = 9.1667ns.
- 2-way set associative: AMAT (cycles) = Hit time + Miss rate * Miss penalty = $5 + \frac{5}{12} * 20 = \frac{40}{3}$ cycles.
  ⇒ AMAT (time) = $\frac{40}{3}$ * 0.5ns = 6.6667ns.
- Full associative: AMAT (cycles) = Hit time + Miss rate * Miss penalty = $5 + \frac{6}{12} * 20 = 15$ cycles.
  ⇒ AMAT (time) = 15 * 0.5ns = 7.5ns.

5 Bài 5

Cho biết hit time của L1 là 10 cycles, hit time của L2 là 15 cycle, thời gian truy xuất của RAM (main memory) là 100 cycles. L1 tỉ lệ miss là 20%, L2 tỉ lệ miss là 10%. Xác định thời gian truy xuất vùng nhớ trung bình của hệ thống trên.

AMAT (cycles) = 10 + 15 * 20% + 100 * 10% = 23 cycles.
f = 2Ghz → T = 1 / f = 0.5ns.
→ AMAT = 23 x 0.5ns = 11.5ns

6 Bài 6: Tính CPI trung bình

Tính CPI trung bình của hệ thống pipeline khi biết tỉ lệ miss của bộ nhớ lệnh là 5%, tỉ lệ miss của bộ nhớ dữ liệu là 10%. Biết đoạn chương trình có 1000 lệnh, trong đó có 100 lệnh là lệnh load và store. Thời gian miss penalty là 100 cycles.

Memory Stall Cycles Per Instruction = Combined Misses Per Instruction × Miss Penalty
$= (5\% + 10\% * \frac{100}{1000}) * 100 = 6$ cycles.
⇒ CPI trung bình = CPI Cache lý tưởng + Mem Stalls cho mỗi lệnh
= CPI Cache lý tưởng + 6 cycles.