

HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

---

Computer Arichtecture

**Lab 8   
Chapter 5 : Memory Caches**

Nguyễn Hữu Hiếu - 2013149

# Bài tập và Thực hành

## Xác định tag, index, offset.

### Bài 1: Cho bộ nhớ chính có không gian 32bit. 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:

1. Direct mapped
2. 4-way set associative
3. Fully associative

Số phần tử trong 1 block = (size of block) / (Đơn vị truy xuất) = 256 byte / 1 byte = 256 =

Số block trong cache = (size of cache) / (size of block) = 4 MB / 256 B

=16 blocks = blocks

Không gian địa chỉ là 32 bit.

* Direct mapped: byte-offset 8 bits, index = 4 bits, tag = 32 – 8 – 4 = 20 bits
* 4-way set associative: 4 blocks tạo thành 1 set mà có blocks nên có sets, byte-offset = 8 bits, index = 2 bits, tag = 32 – 2 – 8 = 22 bits
* Fully associative: byte-offset 8 bits, index 0 bits, tag = 32 – 8 – 0 = 24 bits

### Bài 2: Cho bộ nhớ chính tổng dung lượng là 256MB, 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:

1. Direct mapped
2. 4-way set associative
3. Fully associative

Số phần tử trong 1 block = (size of block) / (Đơn vị truy xuất) = 64words / 2byte =

Số block trong cache = (size of cache) / (size of block) = 256 KB / bytes

= 1024 blocks = blocks

Không gian địa chỉ là 256 MB. Vậy ta dùng thanh ghi **28** **bit** tính theo byte-offset

* Direct mapped: half-word offset 7 bits, index 10 bits, tag = 28 – 7 – 10 =11 bits
* 4-way set associative: 4 block tạo thành 1 set mà có block nên có sets, halft-word offset 7 bits, tag = 28 – 7 – 8 =13 bits
* Fully associative: half-word offset 7 bits, index 0 bits, tag = 28 – 7 – 0 = 21 bits

## 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 cache, 4 words block, đơn vị truy xuất là byte

### Bài 3: Xác đinh số lần HIT/MISS khi chạy chương trình trên với các cấu hình caches sau:

1. Direct mapped
2. 4-way set associative
3. Fully associative

Số phần tử trong 1 block = (size of block) / (Đơn vị truy xuất) = 4 words / 1byte

Số block trong cache = (size of cache) / (size of block) = 256 B / bytes = 16 blocks = blocks

1. Direct mapped

Byte offset : 4 bits

Index : 4 bits

Tag : 24 bits

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Add. | Tag | Offset | Index | Miss/Hit |  |
| 0 | 00000000 00000000 00000000 | 0000 | 0000 | Miss |  |
| 4 | 00000000 00000000 00000000 | 0100 | 0000 | Hit |  |
| 1 | 00000000 00000000 00000000 | 0001 | 0000 | Hit |  |
| 5 | 00000000 00000000 00000000 | 0101 | 0000 | Hit |  |
| 65 | 00000000 00000000 00000000 | 0001 | 0100 | Miss |  |
| 1 | 00000000 00000000 00000000 | 0001 | 0000 | Hit |  |
| 67 | 00000000 00000000 00000000 | 0011 | 0100 | Hit |  |
| 46 | 00000000 00000000 00000000 | 1110 | 0010 | Miss |  |
| 1 | 00000000 00000000 00000000 | 0001 | 0000 | Hit |  |
| 70 | 00000000 00000000 00000000 | 01100 | 0100 | Hit |  |
| 2 | 00000000 00000000 00000000 | 0010 | 0000 | Hit |  |
| 0 | 00000000 00000000 00000000 | 0000 | 0000 | Hit |  |

1. 4-way set associative

Byte offset : 4 bits

Index : 2 bits

Tag : 26 bits

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Add. | Tag | Offset | Index | Miss/Hit |  |
| 0 | 00000000 00000000 00000000 00 | 0000 | 00 | Miss |  |
| 4 | 00000000 00000000 00000000 00 | 0100 | 00 | Hit |  |
| 1 | 00000000 00000000 00000000 00 | 0001 | 00 | Hit |  |
| 5 | 00000000 00000000 00000000 00 | 0101 | 00 | Hit |  |
| 65 | 00000000 00000000 00000000 01 | 0001 | 00 | Miss |  |
| 1 | 00000000 00000000 00000000 00 | 0001 | 00 | Hit |  |
| 67 | 00000000 00000000 00000000 01 | 0011 | 00 | Hit |  |
| 46 | 00000000 00000000 00000000 00 | 1110 | 10 | Miss |  |
| 1 | 00000000 00000000 00000000 00 | 0001 | 00 | Hit |  |
| 70 | 00000000 00000000 00000000 01 | 0110 | 00 | Hit |  |
| 2 | 00000000 00000000 00000000 00 | 0010 | 00 | Hit |  |
| 0 | 00000000 00000000 00000000 00 | 0000 | 00 | Hit |  |

1. Fully associative

Byte offset : 4 bits

Index : 0 bits

Tag : 28 bits

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Add. | Tag | Offset | Miss/Hit |  |
| 0 | 00000000 00000000 00000000 0000 | 0000 | Miss |  |
| 4 | 00000000 00000000 00000000 0000 | 0100 | Hit |  |
| 1 | 00000000 00000000 00000000 0000 | 0001 | Hit |  |
| 5 | 00000000 00000000 00000000 0000 | 0101 | Hit |  |
| 65 | 00000000 00000000 00000000 0100 | 0001 | Miss |  |
| 1 | 00000000 00000000 00000000 0000 | 0001 | Hit |  |
| 67 | 00000000 00000000 00000000 0100 | 0011 | Hit |  |
| 46 | 00000000 00000000 00000000 0010 | 1110 | Miss |  |
| 1 | 00000000 00000000 00000000 0000 | 0001 | Hit |  |
| 70 | 00000000 00000000 00000000 0100 | 0110 | Hit |  |
| 2 | 00000000 00000000 00000000 0000 | 0010 | Hit |  |
| 0 | 00000000 00000000 00000000 0000 | 0000 | Hit |  |

## Tính thời gian truy xuất trung bình (AMAT)

### Bài 4: Xác định thời gian truy xuất trung bình (AMAT) ở Bài 3:, biết rằng Hit time = 5 cycles, thời gian truy xuất RAM là 10ns, tần số máy tính là 2Ghz.

### Bài 5: Cho biết hit time của L1 là 10 cycles, hit time của L2 là 15 cycles, 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.

## Tính CPI trung bình

### Bài 6: 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.

* I-cache miss rate = 5% I-cache:
* D-cache miss rate = 10% D-cache:
* Base CPI = 0
* Actual CPI = Base CPI + I-cache + D-cache = 0 + 5 + 1 = 6