**Đại học Bách Khoa TP.HCM  
Ngành Khoa Học Máy Tính**



**Đề tài:** Bài tập lớn   
Môn Kiến Trúc Máy Tính Nâng Cao

**GVHD:** Trần Ngọc Thịnh

**Học viên:**

Bùi Đức Hiếu 7140231

Trịnh Văn Giang 7140006

Lê Nguyên Dũng 7140224

Lê Nguyễn Khánh Duy 7140226

Đoàn Dũ 7140223

TP.HCM, 3-12-2014

# **Mục lục**

[Mục lục 2](#_Toc11070)

[Bài 1: 3](#_Toc2411)

[Bài 2: 4](#_Toc8227)

[Bài 3: 5](#_Toc30510)

# **Bài 1:**

Độ dài dịa chỉ N = 32 bit

Kích thước block = 32 bytes = 25 bytes (độ dài block offset = 5 bit)

Kích thước cache = 32 KB = 1024 x kích thước block = 210 x kích thước block

**A. Trường hợp dùng Dirrect Mapped**

Kích thước cache = 32 KB = 1024 x kích thước block = 210 x kích thước block ( độ dài index = 10 bit)

Cấu trúc địa chỉ:

|  |  |  |
| --- | --- | --- |
| Tag (17 bits) | Index (10 bits) | Block Offset (5 bits) |

**B. Trường hợp dùng Full Associative**

Cấu trúc địa chỉ:

|  |  |
| --- | --- |
| Tag (27 bits) | Block Offset (5 bits) |

**C. Trường hợp dùng 4-way Associative**

Kích thước set = 4 x kích thước block

Kich thước cache = 210 x kích thước block = 28 x 22 x kích thước block = 28 x kích thước set

Cấu trúc địa chỉ:

|  |  |  |
| --- | --- | --- |
| Tag (19 bits) | Index (8 bits) | Block Offset (5 bits) |

# **Bài 2:**

Với cache loại 4-way associative như bài 1. Ta có kết quả như sau:

Đối với trường hợp Write through no Write Allocation:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Address | Tag | Index | Block offset |  |
| RD | 0x00000000 | 000 0000 0000 0000 0000 | 0000 0000 | 0 0000 | Miss |
| WR | 0x01000000 | 000 0000 1000 0000 0000 | 0000 0000 | 0 0000 | Miss |
| RD | 0x01000010 | 000 0000 1000 0000 0000 | 0000 0000 | 1 0000 | Hit |
| WR | 0x02000050 | 000 0001 0000 0000 0000 | 0000 0010 | 1 0000 | Miss |
| RD | 0x02000058 | 000 0001 0000 0000 0000 | 0000 0010 | 1 1000 | Miss |

Đối với trường hợp Write back with Write Allocation:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Address | Tag | Index | Block offset |  |
| RD | 0x00000000 | 000 0000 0000 0000 0000 | 0000 0000 | 0 0000 | Miss |
| WR | 0x01000000 | 000 0000 1000 0000 0000 | 0000 0000 | 0 0000 | Miss |
| RD | 0x01000010 | 000 0000 1000 0000 0000 | 0000 0000 | 1 0000 | Hit |
| WR | 0x02000050 | 000 0001 0000 0000 0000 | 0000 0010 | 1 0000 | Miss |
| RD | 0x02000058 | 000 0001 0000 0000 0000 | 0000 0010 | 1 1000 | Hit |

# **Bài 3:**

**Unified Cache:**

Kích thước = 32 KB

Misses/1000 instructions = 43.3

Miss Rate = (43.3/1000)/(1 + 0.7) = 25.47x10-3

**Separate Cache:**

I-Cache:

Kích thước = 16 KB

Misses/1000 instructions = 3.82

Miss Rate = (3.82/1000)/(1) = 3.82x10-3

D-Cache:

Kích thước = 16 KB

Misses/1000 instructions = 40.9

Miss Rate = (40.9/1000)/(0.7) = 58.43x10-3

Thời gian truy xuất bộ nhớ trung bình (average memory access time) = %instructions x (Hit Time + InstructionMissRate x MissPenalty) + %data x (Hit Time + DataMissRate x MissPenalty)

**Separate Cache:**

Thời gian truy xuất bộ nhớ trung bình = 1 x (1 + 3.82x10-3 x 50) + 0.3 x (1 + 58.43x10-3 x 50) = 1.191 + 1.176 = 2.367

**Unified Cache (1 port):**

Thời gian truy xuất bộ nhớ trung bình = 1 x (1 + 25.47x10-3 x 50) + 0.3 x (1 + 1+ 25.47x10-3 x 50) = 2.274 + 0.982 = 3.256

**Kết luận:**

Trong trường hợp tính toán, Separate Cache nhanh hơn so với Unified Cache (3.256/2.367 = 1.38 lần)

------------------------------------------------------ Hết -----------------------------------------------------------------