**ĐẠI HỌC QUỐC GIA TP. HCM**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

🙞∙∙∙☼∙∙∙🙜

**BÀI TẬP LỚN MÔN HỌC**

**KIẾN TRÚC MÁY TÍNH (CO2008)**



***ĐỀ TÀI***

**BÁO CÁO BÀI TẬP LỚN CÁ NHÂN**

**LỚP: L14**

**HK231**

**GVHD: NGUYỄN XUÂN MINH**

**SINH VIÊN THỰC HIỆN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MSSV** | **HỌ** | **TÊN** | **% ĐIỂM** | **ĐIỂM** | **GHI CHÚ** |
| 2211185 | Lâm Tuấn | Huy |  |  |  |

**­­­­­­­­­­­­­­­­­­**

*Tp. Hồ Chí Minh, năm học 2023 -2024*

**YÊU CẦU**

**Đề 4:**

**Câu 1**: Viết chương trình MARS MIPS dùng chức năng set seed (syscall 40) theo time (syscall 30) và các chức năng phát số ngẫu nhiên để phát ra 3 số ngẫu nhiên fi (0<fi<1000).

Lưu các kết quả chạy chương trình theo 3 dạng số lẻ lên tập tin SOLE.TXT trên đĩa thành 3 dòng như sau:

2 so le: fff.ff

3 so le: fff.fff

4 so le: fff.ffff

**Câu 2:** Cho danh sách địa chỉ 32-bit truy xuất theo **địa chỉ word** như sau:

5*,* 164*,* 45*,* 4*,* 251*,* 90*,* 173*,* 164*,* 91*,* 44*,* 186*,* 252

* 1. Nếu dùng bộ nhớ cache Direct-mapped có 32 block, mỗi block chứa **1 word**. Hãy xác định địa chỉ theo bit, từ đó suy ra các vùng tag, index lưu trữ vào cache. Cho biết trạng thái Hit/Miss của chuỗi truy xuất trên.
  2. Làm lại câu a) với bộ nhớ cache Direct-mapped có 16 block, mỗi block chứa **2 word**.
  3. Hãy xác định tổng số bit bộ nhớ cần dùng để xây dựng bộ nhớ cache trong cả 2 trường hợp. Biết rằng 1 phần tử cache sẽ chứa 1 bit V, các bit tag và dữ liệu.

**BÀI LÀM**

**Câu 2**

# Trạng thái Hit/Miss:

* Bộ nhớ cache gồm 32 = 25 blocks, nên độ dài trường **index** là 5 bits.
* Mỗi block chứa 1 word = 20 word, nên word **offset** là 0 bit.
* Không gian địa chỉ là 8 word bits nên độ dài trường **tag** là 8 − 5 − 0 = 3 bits.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Word address | Binary | Tag | Index | Hit/Miss |
| 5 | 000 | 00101 | 000 | 00101 | Miss |
| 164 | 101 | 00100 | 101 | 00100 | Miss |
| 45 | 001 | 01101 | 001 | 01101 | Miss |
| 4 | 000 | 00100 | 000 | 00100 | Miss |
| 251 | 111 | 11011 | 111 | 11011 | Miss |
| 90 | 010 | 11010 | 010 | 11010 | Miss |
| 173 | 101 | 01101 | 101 | 01101 | Miss |
| 164 | 101 | 00100 | 101 | 00100 | Miss |
| 91 | 010 | 11011 | 010 | 11011 | Miss |
| 44 | 001 | 01100 | 001 | 01100 | Miss |
| 186 | 101 | 11010 | 101 | 11010 | Miss |
| 252 | 111 | 11100 | 111 | 11100 | Miss |

# Tổng bộ nhớ:

* Bộ nhớ cache Direct-mapped có 32 blocks.
* Mỗi block có 1 word = 1 x 4 bytes = 1 x 4 x 8 = 32 bits.
* Mỗi block có tag địa chỉ là 32 – index – (word offset) - 2 = 32 − 5 – 0 – 2 = 25 bits.
* Mỗi block có 1 valid bit.
* Do đó, tổng số bit bộ nhớ cache cần dùng:

32 x (1 + 25 + 32) = 1856 bit



# Trạng thái Hit/Miss:

* Bộ nhớ cache gồm 16 = 24 blocks, nên độ dài trường **index** là 4 bits.
* Mỗi block chứa 2 words = 21 words, nên word **offset** là 1 bit.
* Không gian địa chỉ là 8 word bits nên độ dài trường **tag** là 8 − 4 − 1 = 3 bits.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Word address | Binary | Tag | Index | Offset | Hit/Miss |
| 5 | 000 | 0010 | 1 | 000 | 0010 | 1 | Miss |
| 164 | 101 | 0010 | 0 | 101 | 0010 | 0 | Miss |
| 45 | 001 | 0110 | 1 | 001 | 0110 | 1 | Miss |
| 4 | 000 | 0010 | 0 | 000 | 0010 | 0 | Miss |
| 251 | 111 | 1101 | 1 | 111 | 1101 | 1 | Miss |
| 90 | 010 | 1101 | 0 | 010 | 1101 | 0 | Miss |
| 173 | 101 | 0110 | 1 | 101 | 0110 | 1 | Miss |
| 164 | 101 | 0010 | 0 | 101 | 0010 | 0 | Miss |
| 91 | 010 | 1101 | 1 | 010 | 1101 | 1 | Hit |
| 44 | 001 | 0110 | 0 | 001 | 0110 | 0 | Miss |
| 186 | 101 | 1101 | 0 | 101 | 1101 | 0 | Miss |
| 252 | 111 | 1110 | 0 | 111 | 1110 | 0 | Miss |

# Tổng bộ nhớ:

* Bộ nhớ cache Direct-mapped có 16 blocks.
* Mỗi block có 2 word = 2 x 4 bytes = 2 x 4 x 8 = 64 bits.
* Mỗi block có tag địa chỉ là 32 – index – (word offset) – 2 = 32 − 4 – 1 − 2 = 25 bits.
* Mỗi block có 1 valid bit.
* Do đó, tổng số bit bộ nhớ cache cần dùng:

16 x (1 + 25 + 64) = 1440 bit

1. **Tổng số bit cần dùng:** 1856 + 1440 = 3296.