CSS422 Homework 6

Vu Dinh

# Q1. Cache

Suppose a byte-addressable memory has 2M byte capacity and cache consists of 64 blocks, where each block contains 32 bytes.

1. Direct Mapping

1) Divide the bits in main memory into tag, block and offset bits.

|  |  |  |
| --- | --- | --- |
| 10 | 6 | 5 |

2) What is the tag, line and offset for the address $173A62, in hexadecimal?

$173A62 = %1 0111 0011 1010 0110 0010

=>

|  |  |  |
| --- | --- | --- |
| 10 1110 0111 | 01 0011 | 0 0010 |

tag: 0x\_2E7\_\_

line: 0x\_\_13\_\_\_

offset: 0x\_\_\_02\_\_\_\_

2. Fully associative mapping

1) Divide the bits in main memory into tag and offset bits.

|  |  |
| --- | --- |
| 16 | 5 |

2) What is the tag and offset for the address $173A62, in hexadecimal?

|  |  |
| --- | --- |
| 1011 1001 1101 0011 | 0 0010 |

tag: 0xB9D3

offset: 0x2

3. 4-way set associative mapping

1) Divide the bits in main memory into tag, set and offset bits

|  |  |  |
| --- | --- | --- |
| 12 | 4 | 5 |

2) What is the tag, set and offset for the address $173A62, in hexadecimal?

|  |  |  |
| --- | --- | --- |
| 1011 1001 1101 | 0011 | 0 0010 |

tag: 0xB9D

set: 0x3

offset: 0x02

# Q2. Cache search

|  |  |  |
| --- | --- | --- |
| 4 | 2 | 2 |

* 0x91 = 0b10010001 = 9/0/1  
  Block 0, tag 9: miss.  
  Replace block 0 with: 0x90, 0x91, 0x92, 0x93
* 0xA8 = 0b10101000 = A/2/0  
  Block 2, tag A: miss.  
  Replace block 2 with: 0xA8, 0xA9, 0xAA, 0xAB
* 0xA9 = 0b10101001 = A/2/1  
  Block 2, tag A: hit
* 0xAB = 0b10101011 = A/2/3  
  Block 2, tag A: hit
* 0xAD = 0b10101101 = A/3/1  
  Block 3, tag A: miss  
  Replace block 3 with 0xAC, 0xAD, 0xAE, 0xAF
* 0x93 = 0b10010011 = 9/0/3  
  Block 0, tag 9: hit
* 0x4E = 0b01001110 = 4/3/2  
  Block 3, tag 4: miss  
  Replace block 3 with 0x4C, 0x4D, 0x4E, 0x4F
* 0x4F = 0b 01001111 = 4/3/3  
  Block 3, tag 4: hit
* 0x50 = 0b01010000 = 5/0/0  
  Block 0, tag 5: miss  
  Replace block 0 with 0x50, 0x51, 0x52, 0x53
* 0xA4 = 0b10100100 = A/1/0  
  Block 1, tag A: miss  
  Replace block 1 with 0xA4, 0xA5, 0xA6, 0xA7

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tag | Block | Offset 0 | Offset 1 | Offset 2 | Offset 3 |
| 0101 | 00 | 50 | 51 | 52 | 53 |
| 1010 | 01 | A4 | A5 | A6 | A7 |
| 1010 | 10 | A8 | A9 | AA | AB |
| 0100 | 11 | AC | AD | AE | AF |

4 hits / 10 requests = 40% hit ratio.

# Q3. Virtual memory

Virtual Address partition:

|  |  |
| --- | --- |
| 3 | 15 |

Physical address partition:

|  |  |
| --- | --- |
| 2 | 15 |

Cache partition:

|  |  |  |
| --- | --- | --- |
| 6 | 3 | 8 |

**0x32764 =** 0b11 0010 0111 0110 0100

=> Virtual page %110 = $6 ----- page table ----> not found ---> **PAGE FAULT**

Virtual page 6 then maps to physical frame 0.

The address is translated as: **0b**0 0010 0111 0110 0100 = **0x**02764

Page fault → update TLB

|  |  |  |
| --- | --- | --- |
| Virtual page # | Physical frame # | Valid bit |
| 6 | 0 | 1 |
| 5 | 3 | 1 |

TLB LRU:

|  |
| --- |
| 6 |
| 0 |

Memory request → update LRU MEM

|  |
| --- |
| 6 |
| 0 |
| 5 |
| 4 |

Page fault → update page table

|  |  |  |
| --- | --- | --- |
| Virtual page # | Physical frame # | valid bit |
| 0 | 2 | 1 |
| 1 | 1 | 1 |
| 2 | --- | 0 |
| 3 | --- | 0 |
| 4 | --- | 0 |
| 5 | 3 | 1 |
| 6 | 0 | 1 |
| 7 | --- | 0 |

Physical address requested: 0x02764 = 0b0 0010 0111 0110 0100 = 04/7/64

Block 7, tag 04: miss

Replace block 7 with 02700 to 027FF

|  |  |  |
| --- | --- | --- |
| Line # | Tag | Data block |
| 0 | 10 |  |
| 1 | 0A |  |
| 2 | 3C |  |
| 3 | 14 |  |
| 4 | 28 |  |
| 5 | 04 |  |
| 6 | 37 |  |
| 7 | 04 | 0x02700 - 0x027FF |