Ryan Young

CSE 3666

12/5/2019

Homework 8

***5.2.1)*** *For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with two-word blocks and a total size of 16 one-word blocks. Also list if each reference is a hit or miss, assuming the cache is initially empty.*

No offset is needed because each block is 1 word in size. Therefore, no offset bit is needed.

These numbers should all be padded so that there are 32 bits in each cell.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Decimal Address | Binary Address | Tag (shifted left 4) | Index ( last 4 bits) | Hit or Miss |
| 3 | 0000 0011 | 0000 | 0011 | Miss |
| 180 | 1011 0100 | 1011 | 0100 | Miss |
| 43 | 0010 1011 | 0010 | 1011 | Miss |
| 2 | 0000 0010 | 0000 | 0010 | Miss |
| 191 | 1011 1111 | 1011 | 1111 | Miss |
| 88 | 0101 1000 | 0101 | 1000 | Miss |
| 190 | 1011 1110 | 1011 | 1110 | Miss |
| 14 | 0000 1110 | 0000 | 1110 | Miss |
| 181 | 1011 0101 | 1011 | 0101 | Miss |
| 44 | 0010 1100 | 0010 | 1100 | Miss |
| 186 | 1011 1010 | 1011 | 1010 | Miss |
| 253 | 1111 1101 | 1111 | 1101 | Miss |

***5.2.2)*** *For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with two-word blocks and a total size of 8 blocks. Also list if each reference is a hit or miss, assuming the cache is initially empty.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Address | Binary | Tag (Address >4) | Index(last four of address) | Offset | Hit or Miss |
| 3 | 0000 0011 | 0000 | 001 | 1 | Miss |
| 180 | 1011 0100 | 1011 | 010 | 0 | Miss |
| 43 | 0010 1011 | 0010 | 101 | 1 | Miss |
| 2 | 0000 0010 | 0000 | 001 | 0 | Hit |
| 191 | 1011 1111 | 1011 | 111 | 1 | Miss |
| 88 | 0101 1000 | 0101 | 100 | 0 | Miss |
| 190 | 1011 1110 | 1011 | 111 | 0 | Hit |
| 14 | 0000 1110 | 0000 | 111 | 0 | Miss |
| 181 | 1011 0101 | 1011 | 010 | 1 | Hit |
| 44 | 0010 1100 | 0010 | 110 | 0 | Miss |
| 186 | 1011 1010 | 1011 | 101 | 0 | Miss |
| 253 | 1111 1101 | 1111 | 110 | 1 | Miss |

***5.3****) For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache.*

Tag: Index: Offset:

31-10 9-5 4-0

***5.3.1)*** *What is the cache block size (in words)?*

*Cache block size = 2offset bits = 25 bytes*

25 bytes = 23 words so 23 = 8 so 8 words long.

***5.3.2)*** *How many entries does the cache have?  
 Number of Entries = 2index bits*

25 = 32 entries

***5.3.3)*** *What is the ratio between total bits required for such a cache implementation over the data storage bits?*

Block size = from 5.3.2, tag size given, valid field size = 1 because there is data

Total bits = 2index x (block size + tag size + valid field size) = 25 x ((8\*32) + 22 + 1) = 8928 bits

Data storage bits = 2index  x (block size) = 25 x 8 = 256

Data storage bits in the actual cache = data storage bits \* number of entries = 8192 bits

Ratio = Total Bits / Data Storage bits = 8928/8192 = 1.089

***5.3.4)*** *How many blocks are replaced?*

A hit is when the index and the tag are the same.

1024: Shares an index with another value but has a different tag so it must be replaced

30: Shares an index with another value and has the same tag it has to be replaced.

3100: index 0 is used again so it needs to be replaced.

2180: index 4 has a tag 2 so it replaced 0 at index 4.

So there are four blocks being replaced.

***5.3.5)*** *What is the hit ratio?*

Hit Ratio: Number of hits / Number of cache access

A hit is when the index and the tag are the same.

Hits found at 4, 16, 140, and 180 because the tag and index are the same

4/12 = 0.3334 \* 100 = 33.34%

*5.3.6) List the final state of the cache, with each valid entry represented as a record of <index, tag, data>.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | Index in Binary | Tag | Tag in Binary | Data(reference) |
| 0 | 00000 | 3 | 0011 | 3100 |
| 4 | 01000 | 0 | 0000 | 2180 |
| 5 | 00101 | 0 | 0000 | 160, 180 |
| 7 | 00111 | 0 | 0000 | 232 |

***5.6)***

***5.6.1)*** *Assuming that the L1 hit time determines the cycle times for P1 and P2, what are their respective clock rates?*

Clock rate = 1 / (cycle time)

P1 clock rate = 1 / L1 hit time = 1/(0.66) = 1.51 GHz

P2 clock rate = 1/(0.90) = 1.11 GHz

***5.6.2)****What is the Average Memory Time for P1 and P2?*

P1 miss time in cycle = Main memory access time / L1 hit time

70/ 0.66 = 106 cycles

P1 miss time = 106 cycles \* 0.66 ns = 70 ns

(Hit rate x hit time) + ( miss rate x miss time )

92% x 0.66ns + 8% x 70ns

P1 = 6.21ns

94% x 0.90 ns + 6% x 70 ns

P2 = 5.05ns

***5.6.3)***

Percentage of memory accesses of 0.36

L1 miss penalty = 107 cycles

0.36 x 0.08 x 107 = 4.08 cycles

70/0.9 = 78 cycles

0.36x0.06x78 = 1.68

1+1.68 = 2.68

Time = CPI x Hit time

4.08 x 0.66

= 2.69

2.68x0.9

= 2.41

P2 is much faster than P1.