Assignment 4

SID: 12110813
Name:刘圣鼎

Problem 1.

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

1. What is the cache block size (in words)?

Block size=$2^5 = 32\ bytes = 8\ words$

2.How many entries does the cache have?

Number of entries=$2^5 = 32$

3.What is the ratio between total bits required for such a cache implementation over the data storage bits? (Assume the cache has valid bit, dirty bit and reference bit)

The bits of Tag are $32 - 10 = 22$

And the bits of the cache are composed of the 3 parts.

Assume the cache has valid bit, dirty bit and reference bit.

The first part is the valid bit, dirty bit and reference bit.

The second part is the tag, which is 22 bits.

The third part is the data storage bits which are $32 * 8 = 256$ bits.

Therefore, there are 256 bits for the data.

The ratio is $(256 + 22 + 3)/256 \approx 1.09$

Starting from power on, the following byte-addressed cache references are recorded.

Address											
0	4	16	132	232	160	1024	30	140	3100	180	2180

4.How many blocks are replaced?

ADDRESS	INDEX	TAG	REPLACE
0	00000=0	0	
4	00000=0	0	
16	00000=0	0	
132	00100=4	0	
232	00111=7	0	
160	00101=5	0	
1024	00000=0	1	Y
30	00000=0	0	Y
140	00100=4	0	
3100	00000=0	11=3	Y
180	00101=5	0	
2180	00100=4	10=2	Y

As the above table shows, there are 4 blocks replaced.

5.What is the hit ratio?

$$hit\ ratio = \frac{hit\ times}{access\ times} = \frac{4}{12} = \frac{1}{3}$$

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

INDEX	TAG	DATA
0	0	Mem[1 1110]
1		
2		
3		
4	2	Mem[1000 1000 0100]
5	0	Mem[1011 0100]
6		
7	0	Mem[1000 0100]
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		
21		
22		
23		

INDEX	TAG	DATA
24		
25		
26		
27		
28		
29		
30		
31		

Problem 2.

Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 32-bit memory address references, given as word addresses

3, 180, 43, 2, 191, 88, 190, 14, 181

The exercise examines the impact of different cache designs, specifically comparing associative caches to the direct-mapped caches. The address stream is shown above.

1. For a three-way set associative cache with two-word blocks and a total size of 24 words, which bits represent index and which bits represent tag in 32-bit memory address (e.g. Index: 9-5 Tag: 31-10) ?

Offset:29-31 Index: 27-28 Tag:0-26

2. Using the given sequence, show the final cache contents for a three-way set associative cache with two-word blocks and a total size of 24 words. Use LRU replacement. For each reference identify the index bits, the tag bits, the block of set bits, and if it is a hit or a miss

Because the block size is 8 bytes, we access address 3, 180, 43, 2, 191, 88, 190, 14, 181 sequentially, then, the block sequence we accessed should be: 0, 10, 5,0,11,10,1,10

Base CPI, No Memory Stalls	Processor Speed	Main Memory Access Time	First Level Cache MissRate per Instruction	Second Level Cache, Direct-Mapped Speed	Global Miss Rate with Second Level Cache, Direct-Mapped	Second Level Cache, Eight-Way Set Associative Speed	Global Miss Rate with Second Level Cache, Eight-Way Set Associative
1.5	2 GHz	100 ns	7%	12 cycles	3.5%	28 cycles	1.5%

5.Calculate the CPI for the processor in the table using: 1) only a first level cache, 2) a second level direct-mapped cache, and 3) a second level eight-way set associative cache.

Given that

CPU base CPI = 1.5 , clock rate = 2GHz

Miss rate/instruction = 7%

Main memory access time = 100ns

1)For a first level cache:

Miss Penalty=$7\% * 100 * 2 = 14\ cycles$

Effective CPI=Base CPI+Miss Penalty=$1.5 + 14$=15.5

2)For a second level direct-mapped cache

Given that

Access time = 12 cycles

Global miss ratio of L2 cache = 3.5% (Local miss ratio of L2 = 3.5%/7% = 50%)

There are 3 cases.

- L-1 hit Base CPI=1.5
- L-1 miss with L-2 hit Penalty = 12+1.5=13.5 cycles
- L-1 miss with L-2 miss Penalty = 200+12+1.5=213.5 cycles

Effective CPI = 0.93 × 1 + 0.035 × 13.5 + 0.035 × 213.5 = 8.875

3)For a second level eight-way set associative cache

Given that

Access time = 28 cycles

Global miss ratio of L2 cache = 1.5% (Local miss ratio of L2 = 1.5%/7%)

There are 3 cases.

- L-1 hit Base CPI=1.5
- L-1 miss with L-2 hit Penalty = 28+1.5=29.5 cycles
- L-1 miss with L-2 miss Penalty = 200+28+1.5=229.5 cycles

Effective CPI = 0.93 × 1 + 0.055 × 29.5 + 0.015 × 229.5 = 0.93+1.6225+3.4425=5.995

Problem 3.

This Exercise examines the single error correcting, double error detecting (SEC/DED) Hamming code.

1. What is the minimum number of parity bits required to protect a 128-bit word using the SEC/DED code?

To use Hamming code for error correction, the first step is to determine the number of digits (also known as "Hamming code length") required for the sent data.

Assuming that N represents the binary digits of the entire information after adding a checksum bit, K represents the effective information bits, and r represents the added checksum bit, the relationship between them should satisfies: $N = K + r \leq 2^r - 1$.

Suppose the minimum number of parity bits required to protect a 128-bit word using the SEC/DED code is r.

$2^r - r \geq K - 1 = 128 - 1$

Therefore, $r \geq 8$.

the minimum number of parity bits required to protect a 128-bit word using the SEC/DED code is 8

2.Consider a SEC code that protects 8 bit words with 4 parity bits. If we read the value 0x375, is there an error? If so, correct the error.

$0x375 = (0011\ 0111\ 0101)_2$

	P1	P2	B1	P3	B2	B3	B4	P4	B5	B6	B7	B8
	0	0	1	1	0	1	1	1	0	1	0	1
P1	×		×		×		×		×		×	
P2		×	×			×	×			×	×	
P3				×	×	×	×					×
P4								×	×	×	×	×

After calculation, there is only the P3 parity bit not different from the bit in the SEC code.

P3=$1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0 \neq 1$

As a result, there is an error. And the error is the P3 parity bit.

The correct SEC code is $0x275 = (0010\ 0111\ 0101)_2$

Problem 4.

Virtual memory uses a page table to track the mapping of virtual addresses to physical addresses. This exercise shows how this table must be updated as addresses are accessed. The following data constitutes a stream of virtual addresses as seen on a system. Assume page size is 4 KiB, a 4-entry fully associative TLB, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number (i.e. If the current maximum physical page number is 12, then the next physical page brought in from disk has a page number of 13).

4669, 2227, 13916, 34587, 12608

TLB:

Valid	Tag	Physical Page Number
1	11	12
1	7	4
1	3	6
0	4	9

Page table:

Valid	Physical Page or in Disk
1	5
0	Disk
0	Disk
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

1.Which bits represent virtual page number and which bits represent page offset in 32- bit virtual memory address?

Because page size is 4 KiB, the offset is 12 bits, i.e. 0-11 bits represent page offset.

Therefore,

virtual page number:12-31

page offset:0-11

2.Given the address stream shown, and the initial TLB and page table states provided above, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.

4669=0001 0010 0011 1101

2227=0000 1000 1011 0011

13916=0011 0110 0101 1100

34587=1000 0111 0001 1011

12608=0011 0001 0100 0000