CS224 - Spring 2022 - Lab #6 (Version 2: April 19, 22:49)

Examining the Effect of Cache Parameters and Program Factors on Cache Hit Rate

Utku Kurtulmuş-Section-02-21903025

Answers to problems 1-4

Part 1. Preliminary Work (50 points)

You have to provide a neat presentation prepared by Word or a word processor with similar output quality. Handwritten answers will not be accepted.

1. (5 points: With 3 or more errors you get 0 points. Otherwise full point.) Fill in the empty cells of the following table. Assume that main memory size is 2 GB. **Index Size**: No. of bits needed to express the set number in an address, **Block Offset**: No. of bits needed to indicate the word offset in a block, **Byte Offset**: No. of bits needed to indicate the byte offset in a word. **Block Replacement Policy Needed**: Indicate if a block replacement policy such as FIFO, LRU, LFU (Least Frequently Used) etc. is needed (yes) or not (no). If some combinations are not possible mark them.

No.	Cache Size KB	N way cache	Word Size (no. of bits)	Block size (no. of words)	No. of Sets	Tag Size in bits	Index Size (Set No.) in bits	Block Offset Size in bits[1]	Byte Offset Size in bits[2]	Block Replacement Policy Needed (Yes/No)
1	64	1	32	4	2^12	16	12	2	2 bits	no
2	64	2	32	4	2^11	17	11	2	2 bits	yes
3	64	4	32	8	2^9	18	9	3	2 bits	yes
4	64	Full	32	8	1	27	0	3	2 bits	yes
9	128	1	16	4	2^14	X	14	2	1 bit	no
10	128	2	16	4	2^13	0	13	2	1 bit	yes
11	128	4	16	16	2^10	1	10	4	1 bit	yes
12	128	Full	16	16	1	10	0	4	1 bit	yes

X: since word size 16 bits not 32 bits as usual, address size is 16 bits too. So, wo can't provide enough bits for that set up. But if we assume 32 bits again while calculating the tag size, we can say 32- (14 + 2 + 1) = 15. If we assume that, answers for the last 4 rows also change. Also the valid big is ignored, there is no information about that.

[1] **Block Offset Size in bits:** Log_2(No. of words in a block)

[2] **Byte Offset Size in bits:** Log_2(No. of bytes in a word)

2. **(5 points: With 3 or more errors you get 0 points. Otherwise full point.)** Consider the following MIPS code segment. (Remember MIPS memory size is 4 GB.) Cache capacity is 16 words, Block size: 4 words, N= 2.

```
        addi    $t0, $0, 5
loop:   beq     $t0, $0, done
        lw      $t1, 0x24($0)
        lw      $t2, 0xAC($0)
        lw      $t3, 0xC8($0)
        addi    $t0, $t0, -1
        j       loop
done:
```

a. In the following table indicate the type of miss, if any: Compulsory, Conflict, Capacity.

Instruction	**Iteration No.**				
	1	**2**	**3**	**4**	**5**
lw $t1, 0x24($0)	compulsory	capacity	capacity	capacity	capacity
lw $t2, 0xAC($0)	compulsory	capacity	capacity	capacity	capacity
lw $t3, 0xC8($0)	capacity	capacity	capacity	capacity	capacity

I assumed that there is no block replacement since it is not mentioned.

b. What is the total cache memory size in number of bits? Include the V bit your calculations. Show the details of your calculation.

Number of Sets * N(way) * (V + Tag + #of words in a block * wordsize)

2*2*(1 + 27 + 4*32) = 624 bits

c. State the number of AND and OR gates, EQUALITY COMPARATORs and MULTIPLEXERs needed to implement the cache memory. No drawing is needed.

AND = 2

OR = 1

EC = 2, one for each way. To compare the tags.

MUX =2 (4:1) mux to select data within the block, one for way0 one for way1, 1 2:1 mux to select from way1 or way0. In total 3 mux.

3. **(5 points: With 3 or more errors you get 0 points. Otherwise full point.)** Consider the above MIPS code segment. Block size is 1 word. There is only 1 set. Cache memory size is 2 blocks. The block replacement policy is LRU.

a. In the following table indicate the type of miss, if any: Compulsory, Conflict, Capacity.

Instruction	**Iteration No.**				
	1	**2**	**3**	**4**	**5**
lw $t1, 0x24($0)	compulsory	capacity	capacity	capacity	capacity
lw $t2, 0xAC($0)	compulsory	capacity	capacity	capacity	capacity
lw $t3, 0xC8($0)	capacity	capacity	capacity	capacity	capacity

Since capacity becomes full, it can be considered as conflict miss as well.

b. How many bits are needed for the implementation of LRU policy? What is the total cache memory size in number of bits? Include the V bit and the bit(s) used for LRU in your calculations. Show the details of your calculation.
(#u bit/set) + (#sets)*N*(Vbit+tag+data) = 1 + 1 * 2 * (1 + 30 + 32) = 1 + 126 = 127bits

c. State the number of AND and OR gates, EQUALITY COMPARATORs and MULTIPLEXERs needed to implement the cache memory. No drawing is needed.
MUX = 1 2:1 mux for selecting data from ways.
EC = 2
AND = 2
OR = 1
No hardware is needed for u bits, we can use hit1 for the way0 and hit 0 for the way1. In this case ,if we use the data from way0 this means that we got a hit0 then the data in the way1 should be the least recently used (u = 1) and ready to be evicted. We can also connect hit0 to way0 and hit1 to way1 but we would need inverters.

4. **(5 points, With 1 or more errors you get 0 points. Otherwise full point.)** Consider a three level memory: L1 and L2 are for cache memory and the third level is for the main memory. Access time for L1 is 2 clock cycle, the access time for L2 is 4 clock cycles and main memory access time is 20 clock cycles. The miss rate for L1 is 10% and the miss rate for L2 is 5%. What is the effective clock cycle for memory access (AMAT in number of clock cycles)?

AMAT = TL1 + MissRateL1 * (TL2 + MissRateL2*Tmain)
= 2 + 0.1 * (4 + 0.05 * 20) = 2 + 0.1 * (5) = 2.5 CLK
With 2 GHz clock rate how much time is needed for a program with 10^{10} instructions to execute?
2GHz clock rate ---> TC = 0.5 ns
Total time = 10^10 * 2.5 * 0.5 =1.25 * 10^10 ns = 1.25 * 10 s = 12.5 s