Habib University

EE-371/CS-330/CE-321 Computer Architecture – Spring 2023
Instructors: Dr. Tariq Kamal, Dr. Farhan Khan, Dr. Waseem Hassan

Time = 50 minutes **Quiz 04 SOL** **Max Points: 20**

Instructions:

i. ***Smart watches, laptops, and similar electronics are strictly NOT allowed.***
ii. ***Answer sheets should contain all steps, working, explanations, and assumptions.***
iii. Attempt the quiz on clean papers with black/blue ink.
iv. Print your name and HU ID on all sheets.
v. Turn in your question paper along with your answer sheets.
vi. You are not allowed to ask/share your method or answer with your peers. The work submitted by you is solely your own work. Any violation of this will be the violation of HU Honor code and proper action will be taken as per university policy if found to be involved in such an activity.

CLO Assessment:

This assignment assesses students for the following course learning outcomes.

Course Learning Outcomes		**CLO Assessed**
CLO 1	***Explain*** the role of ISA in modern processors and instruction encodings and assembly language programming	
CLO 2	***Explain*** the architecture and working of a single cycle processor	
CLO 3	***Design*** the architecture to mitigate issues of a pipelined processor	
CLO 4	***Analyze the*** performance of cache operations	✓

Question 1 [8 points]:

A direct-mapped cache with 1024 blocks has been proposed for your newly designed Fast-MATH embedded processor that uses RISC-V architecture.

Each block can hold 4 words. The memory address space is 32 bits (4GB), and the cache is byte-addressable.

(Note: Each word contains 4 bytes)

a) [4 points] How many bits from the main memory address will be allocated for tag, index, and block offset?

Tag	Index	Block Offset

(Note: Block offset also includes byte offset in its least significant bits)

b) [4 points] Draw a diagram for the cache organization specifying the relevant multiplexer(s) and other logic.

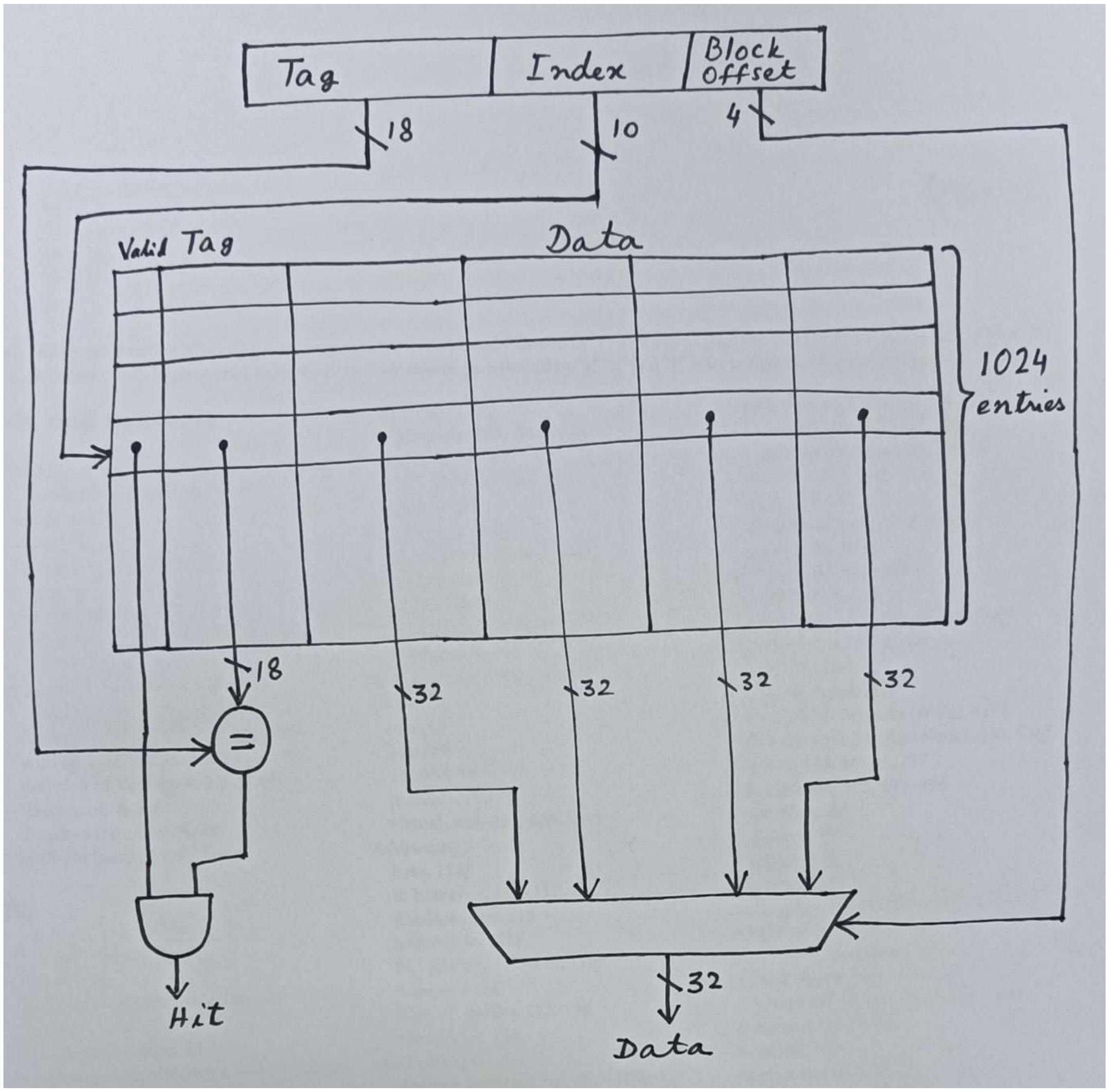
Answer:

a)

- Each block holds 4 words = 4 x 4 = 16 bytes = 2^4 bytes, therefore, 4 bits are allocated for Block Offset.
- 1024 blocks in Cache = 2^{10} blocks in cache, therefore, 10 bits are allocated for index.
- Total bits in memory address are 32, therefore, remaining 32 – 10 – 4 = 18 bits are allocated for tag.

Tag = 18	Index = 10	Block Offset = 4

b)



Question 2 [4 points]:

Suppose we have a processor with a base CPI of 1, assuming all references hit in the primary cache, and a clock rate of 4 GHz. Assume a main memory access time of 100 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 3%.

a) [2 points] How much faster will the processor be if we add a secondary cache (L2 cache) that has a 10-ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 0.75%?

b) [2 points] Repeat part a if the secondary cache (L2 cache) has a 20-ns access time for either a hit or a miss.

Answer:

a)

The miss penalty to main memory is:

$$\frac{100\,ns}{0.25\,ns} = 400\;clock\;cycles$$

The effective CPI with one level of caching is given by:

Total CPI = Base CPI + Memory-stall cycles per instruction

For the processor with one level of caching,

Total CPI = 1.0 + Memory-stall cycles per instruction = 1 + 3% × 400 = 13

With two levels of caching, a miss in the primary (or first-level) cache can be satisfied either by the secondary cache or by main memory.

The miss penalty for an access to the second-level cache is:

$$\frac{10\,ns}{0.25\,ns} = 40\;clock\;cycles$$

If the miss is satisfied in the secondary cache, then this is the entire miss penalty. If the miss needs to go to main memory, then the total miss penalty is the sum of the secondary cache access time and the main memory access time.

Thus, for a two-level cache, total CPI is the sum of the stall cycles from both levels of cache and the base CPI:

Total CPI = 1.0 + Primary stalls per instruction + Secondary stalls per instruction= 1.0 + 3% × 40 + 0.75% × 400 = 1.0 + 1.2 + 3 = 5.2

Thus, the processor with the secondary cache is faster by:

$$\frac{13}{5.2} = 2.5$$

ALTERNATIVE SOLUTION:

Alternatively, we could have computed the stall cycles by summing the stall cycles of those references that hit in the secondary cache ((3% − 0.75%) × 40 = 0.9). Those references that go to main memory, which must include the cost to access the secondary cache as well as the main memory access time, are (0.75% × (40 + 400) = 3.3). The sum, 1.0 + 0.9 + 3.3, is again 5.2.

b)

The miss penalty for an access to the second-level cache is:

$$\frac{20\ ns}{0.25\ ns} = 80\ clock\ cycles$$

If the miss is satisfied in the secondary cache, then this is the entire miss penalty. If the miss needs to go to main memory, then the total miss penalty is the sum of the secondary cache access time and the main memory access time.

Thus, for a two-level cache, total CPI is the sum of the stall cycles from both levels of cache and the base CPI:

Total CPI = 1.0 + Primary stalls per instruction + Secondary stalls per instruction= 1.0 + 3% × 80 + 0.75% × 400 = 1.0 + 2.4 + 3 = 6.4

Thus, the processor with the secondary cache is faster by:

$$\frac{13}{6.4} = 2.03$$

ALTERNATIVE SOLUTION:

Alternatively, we could have computed the stall cycles by summing the stall cycles of those references that hit in the secondary cache ((3% − 0.75%) × 80 = 1.8). Those references that go to main memory, which must include the cost to access the secondary cache as well as the main memory access time, are (0.75% × (80 + 400) = 3.6). The sum, 1.0 + 1.8 + 3.6, is again 6.4.

Question 3 [8 points]:

a) [4 points] Assume a two-way set-associative cache with four sets and a total of eight one-word blocks. Consider the following word address sequence:
0, 1, 3, 7, 8, 9, 10, 0, 8, 6, 11, 12

Assuming an LRU replacement policy, which accesses are hits?

b) [4 points] Assume a fully-associative cache with eight one-word blocks. Consider the following word address sequence:
0, 1, 3, 7, 8, 9, 10, 0, 8, 6, 11, 12

Assuming an LRU replacement policy, which accesses are hits?

(Hint: Draw a table and specify each memory access as a Hit or a Miss.

Answer:

a)

Block Address	Cache Set Index	Hit/ Miss	Cache content after access							
			Set 0		Set 1		Set 2		Set 3	
0	0 mod 4 = 0	Miss	M[0]							
1	1 mod 4 = 1	Miss	M[0]		M[1]					
3	3 mod 4 = 3	Miss	M[0]		M[1]				M[3]	
7	7 mod 4 = 3	Miss	M[0]		M[1]				M[3]	M[7]
8	8 mod 4 = 0	Miss	M[0]	M[8]	M[1]				M[3]	M[7]
9	9 mod 4 = 1	Miss	M[0]	M[8]	M[1]	M[9]			M[3]	M[7]
10	10 mod 4 = 2	Miss	M[0]	M[8]	M[1]	M[9]	M[10]		M[3]	M[7]
0	0 mod 4 = 0	Hit	M[0]	M[8]	M[1]	M[9]	M[10]		M[3]	M[7]
8	8 mod 4 = 0	Hit	M[0]	M[8]	M[1]	M[9]	M[10]		M[3]	M[7]
6	6 mod 4 = 2	Miss	M[0]	M[8]	M[1]	M[9]	M[10]	M[6]	M[3]	M[7]
11	11 mod 4 = 3	Miss	M[0]	M[8]	M[1]	M[9]	M[10]	M[6]	M[11]	M[7]
12	12 mod 4 = 0	Miss	M[12]	M[8]	M[1]	M[9]	M[10]	M[6]	M[11]	M[7]

There are 2 hits.

b)

Block Address	Hit/Miss	Cache content after access							
		Block 0	Block 1	Block 2	Block 3	Block 4	Block 5	Block 6	Block 7
0	Miss	M[0]							
1	Miss	M[0]	M[1]						
3	Miss	M[0]	M[1]	M[3]					
7	Miss	M[0]	M[1]	M[3]	M[7]				
8	Miss	M[0]	M[1]	M[3]	M[7]	M[8]			
9	Miss	M[0]	M[1]	M[3]	M[7]	M[8]	M[9]		
10	Miss	M[0]	M[1]	M[3]	M[7]	M[8]	M[9]	M[10]	
0	Hit	M[0]	M[1]	M[3]	M[7]	M[8]	M[9]	M[10]	
8	Hit	M[0]	M[1]	M[3]	M[7]	M[8]	M[9]	M[10]	
6	Miss	M[0]	M[1]	M[3]	M[7]	M[8]	M[9]	M[10]	M[6]
11	Miss	M[0]	M[11]	M[3]	M[7]	M[8]	M[9]	M[10]	M[6]
12	Miss	M[0]	M[11]	M[12]	M[7]	M[8]	M[9]	M[10]	M[6]

There are 2 hits.