Computer Architecture

Fall, 2023

Week 16

2023.12.25

組別：_____ 簽名：_____

[group4]

1. Consider the following table, calculate the total page table size for a system running 5 programs. (Each program has their own page table)

Virtual address size	Page size	Page table entry size
32 bits	4KB	4 bytes

Ans:

4KB page → 12 offset bits, 20 page number bits

2^{20} = 1M page table entries.

1M entries * 4 bytes (per entry) = 4MB

4MB * 5 program = 20MB

[group5]

2. How would cache size, block size and associativity change effect miss rate of those miss types. What are the possible effects for overall performance? Please complete the following table.

Design change	Effect on miss rate	Possible effects

Ans:

Design change	Effect on miss rate	Possible effects
size ↑	capacity miss ↓	access time ↑
associativity ↑	conflict miss ↓	access time ↑
block size ↑	spatial locality ↑	miss penalty ↑

[group10]

3. Please fill the blank below：

Cache	TLB	Page table	Possible? Conditions?
Miss	Hit	Hit	
Hit	Miss	Hit	
Miss	Miss	Hit	
Miss	Miss	Miss	
Miss	Hit	Miss	
Hit	Hit	Miss	
Hit	Miss	Miss	

Ans:

Cache	TLB	Page table	Possible? Conditions?
Miss	Hit	Hit	Yes; but page table never checked if TLB hits
Hit	Miss	Hit	TLB miss, but entry found in page table;after retry, data in cache
Miss	Miss	Hit	TLB miss, but entry found in page table; after retry, data miss in cache
Miss	Miss	Miss	TLB miss and is followed by a page fault; after retry, data miss in cache
Miss	Hit	Miss	impossible; not in TLB if page not in memory
Hit	Hit	Miss	impossible; not in TLB if page not in memory
Hit	Miss	Miss	impossible; not in cache if page not in memory

[group6]

4. Correct the following statements:
   a) the entire virtual address, including page offset needs to be translated into physical address before use.
   b) despite its name, page table register is stored in main memory as it is the first entry of the page table.
   c) page fault is detected by operating system and handled by sophisticated hardware design.
   d) both software and hardware could be used to handle when a PTE isn't found in TLB, depending on the complexity of page table's structure.
   e) in case of cache hit, TLB miss, and page table Hit: TLB is updated but not used to access cache
   f) in case of cache hit, page table miss: we need to update page table
   g) n-way and full associativity negates the possibility of conflict misses

Ans:

   a) only virtual page number needs to be translated
   b) page table register is stored in CPU, and it stores the address of the first entry != the first entry
   c) detected by hardware, handled by software
   d) TRUE
   e) TLB is updated and used again during retry
   f) case is impossible
   g) only full associativity

[group1]

5. True or false for the following statements
   a. A page fault typically takes less than 10 cycles to process.
   b. According to the slides, to reduce page fault rate, least-recently used replacement (LRU) are preferred.
   c. Using bounds register to limit table size, and add more if exceed is one of the possible solutions to handle huge page table.
   d. It is possible that cache hits, TLB hits, and page table misses.
   e. If page is not in memory, first, OS handles fetching the page and updating the page table, second, it then restarts the faulting instruction.

Ans:

   a. false, may take million cycles to process
   b. true

c. true

d. false, it's impossible, not in TLB if page not in memory

e. true

[group13]

6. Please choose the wrong statements and explain why.

(A) The page table is stored in the main memory.

(B) Suppose the page offset has 12 bits. Given the virtual address = 0x11A7, we can know its page offset of the corresponding physical address is 0x00A7.

(C) In the virtual machine, we first lookup TLB. If TLB misses, we will access to the page table to get the PTE.

(D) Conflict misses will occur in the fully associative caches.

Ans: (A) (C)

(B) The page offset of the physical address is 0x01A7.

(D) Conflict misses are more likely to occur in the non-fully associative caches instead of the fully associative caches.

[group3]

7. True or False

(A) It is possible to have a situation where there is a cache miss, a TLB hit, and a page table hit, and yet the page table never needs to be checked because the TLB hit provides the necessary information.

(B) The size of data blocks transferred from disk to main memory is irrelevant to the performance of virtual memory.

(C) Typical RISC processors do not include a memory management unit (MMU) that contains a TLB for page table lookups.

(D) The miss rate for TLBs can range from 0.01% to 1%, with the number of cycles for handling a miss varying between 10 to 100 cycles.

(E) It is possible for a cache hit to occur at the same time as a TLB miss and a page table miss.

Ans:

(A) True

(B) False: The size of data blocks transferred from disk to main memory is quite relevant to the performance of virtual memory. Larger blocks can reduce the number of transfers needed but may lead to wasted space if the large block is not fully utilized.

(C) False: Typical RISC processors include a memory management unit (MMU) that contains a TLB for page table lookups.

(D) True: The miss rate for TLBs can range from 0.01% to 1%, with the number of cycles for handling a miss varying between 10 to 100 cycles, according to the information provided.

(E) False: It is not possible for a cache hit to occur at the same time as a TLB miss and a page table miss, as the data would not be in the cache if it is not in the TLB or page table.

[group2]

8. Which of the following statements are true.

(a) Page fault is means that page is not resident in disk.

(b) The possible solution for huge page table is to let pages to grow in both directions, one for hash, and one for stack.

(c) A page with reference bit is 1 in LRU, means it hasn't been used recently.

(d) We use both write-through and write-back method in disk write.

(e) A translation lookaside buffer (TLB) is a memory cache that stores the recent translations of virtual memory to physical memory. It is used to reduce the time taken to access a user memory location.

Ans:

b, e