LE16.3

Bookmark this page

LE16.3.1: Page Map Arithmetic

0.0/1.0 point (ungraded)

Consider a virtual memory system that uses a single-level page map to translate virtual addresses into physical addresses. Each of the questions below asks you to consider what happens when **just ONE of the design parameters** (page size, virtual memory size, physical memory size) of the original system is changed. Select the correct answer.

1. If the physical memory size (in bytes) is doubled, the number of entries in the page table
   - ○ stays the same
   - ○ doubles
   - ○ is reduced by half
   - ○ increases by one
   - ○ decreases by one

2. If the page size (in bytes) is halved, the number of entries in the page table
   - ○ stays the same
   - ○ doubles
   - ○ is reduced by half
   - ○ increases by one
   - ○ decreases by one

3. If the virtual memory size (in bytes) is doubled, the number of bits in each entry of the page table
   - ○ stays the same
   - ○ doubles
   - ○ is reduced by half
   - ○ increases by one
   - ○ decreases by one

4. If the page size (in bytes) is doubled, the number of bits in each entry of the page table
   - ○ stays the same
   - ○ doubles
   - ○ is reduced by half
   - ○ increases by one
   - ○ decreases by one

Calculator

Consider a virtual memory system for the Gamma processor with 4096 ($\mathbf{2^{12}}$) virtual pages and 16384 ($\mathbf{2^{14}}$) physical pages where each page contains 1024 ($\mathbf{2^{10}}$) bytes. The first 8 entries of the current page map are shown below:

index	**D**	**R**	**PPN**
0	1	1	0×22
1	0	1	0×01
2	--	0	--
3	0	1	0×02
4	1	1	0×03
5	--	0	--
6	1	1	0×15
7	0	1	0×04
...			

5. What is the total number of bits in the page map?

   Total number of bits in the page map (you can express your answer as a mathematical expression like 2.5*2^7, for example):

6. Which address bits from the CPU are used to choose an entry from the page table?

   Address bits used to choose page table entry: A[:]

7. What is the physical address for the word at virtual location 0×1234?

   Physical address for byte at virtual address 0×1234: 0x

8. What action caused the D bit for page 6 to be 1?

   - A LD of address 0×1856
   - A LD of virtual page 6.
   - A ST instruction wrote to a location in physical page 6.
   - A ST instruction wrote to a location in virtual page 6.

Submit

LE16.3.2: Memory access time

0.0/1.0 point (ungraded)

The pagemap resides to main memory and accessed via a translation lookaside buffer (TLB). Each main memory access takes 50 ns and each TLB access takes 10 ns. Each virtual memory access involves:

- mapping VPN to PPN using TLB [10 ns]
- if TLB miss: mapping VPN to PPN using page map in main memory [50 ns]
- accessing main memory at appropriate physical address [50 ns]

Assuming no page faults (i.e., all virtual memory pages are resident) what TLB hit rate is required for an average virtual memory access time of 61 ns?

TLB hit rate for 61 ns access time:

Calculator

Submit

LE16.3.3: TLB

0.0/1.0 point (ungraded)

Consider a system with 40-bit virtual addresses, 36-bit physical addresses, and 64 KB (2^{16} bytes) pages. The system uses a page map to translate virtual addresses to physical addresses; each page map entry includes dirty (D) and resident (R) bits.

1. Assuming a flat page map, what is the size of each page map entry, and how many entries does the page map have? You can express your answers as mathematical expressions if you like, e.g., 10*(2^17).

   Size of page map entry in bits:

   Number of entries in the page map:

2. If we changed the system to use 16 KB (2^{14} bytes) pages instead of 64 KB pages, how would the number of entries in the page map change? Please give the ratio of the new size to the old size.

   (# entries with 16 KB pages) / (# entries with 64 KB pages):

   Assume 64 KB pages for the rest of this exercise.

3. The contents of the page map and TLB are shown below. The page map uses an LRU replacement policy, and the LRU page (shown below) will be chosen for replacement. For each of these four accesses, compute its corresponding physical address and indicate whether the access causes a TLB miss and/or a page fault. Assume each access starts with the TLB and Page Map state shown below.

   TLB:

VPN(tag)	V	D	PPN
0×0	1	0	0xBE7A
0×3	0	0	0×7
0×5	1	1	0xFF
0×2	1	0	0×900

   Page Table:

VPN	D	R	PPN
0×0	0	1	0xBE7A
0×1	--	0	--
0×2	0	1	0×900
LRU ⟶ 0×3	0	1	0×8
0×4	--	0	--
0×5	1	1	0xFF
0×6	0	1	0×70

Fill in the table below

Virtual Addr	PPN (in hex)	Physical Addr (in hex)	TLB Miss?	Page Fault?
0×06004	0x:	0x:	Select an option	Select an optic

Calculator

0×30286	0x:	0x:	Select an option	Select an option
0×68030	0x:	0x:	Select an option	Select an option
0×4BEEF	0x:	0x:	Select an option	Select an option

Submit

Discussion

Topic: 16. Virtual Memory / LE16.3

Hide Discussion

Add a Post

Show all posts by recent activity

- Physical page number outnumbers virtual page number — 3
  Physical page number outnumbers virtual page number in LE16.3.1. How is that possible?
- [STAFF] typo in LE16.3.3: TLB part C? — 5
- LE16.3.1 part D — 3
  Community TA
- [HTND] LE16.3.2 (Spoiler, don't read if you haven't done this exercise) — 1
  Community TA

Previous Next

Calculator