Homework 8 solution

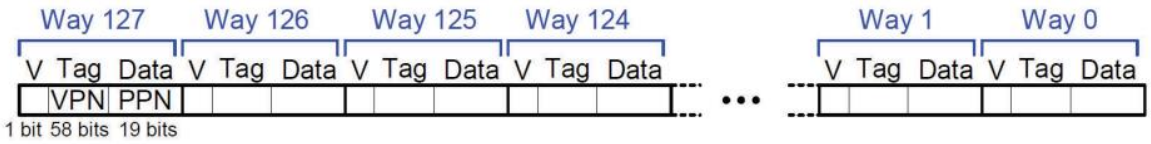
Exercise 8.21

(a) **31 bits**
(b) $2^{50}/2^{12}$ = **2^{38} virtual pages**
(c) 2 GB / 4 KB = $2^{31}/2^{12}$ = **2^{19} physical pages**
(d) virtual page number: **38 bits**; physical page number = **19 bits**
(e) **2^{38}** page table entries (one for each virtual page).
(f) Each entry uses 19 bits of physical page number and 2 bits of status information. Thus, **3 bytes** are needed for each entry (rounding 21 bits up to thevnearest number of bytes).
(h)The total table size is **3 x 2^{38} bytes**.

Exercise 8.23

(a) 1 valid bit + 19 data bits (PPN) + 38 tag bits (VPN) x 128 entries = 58 × 128 bits = **7424 bits**

(b)



(c) 128 × 58-bit SRAM

Exercise 8.25

(a) Each entry in the page table has 2 status bits (*V* and *D*), and a physical page number (22-16 = 6 bits). The page table has $2^{25-16} = 2^9$ entries.
Thus, the total page table size is 2^9 × 8 bits = **4096 bits**

(b) This would increase the virtual page number to 25 - 14 = 11 bits, and the physical page number to 22 - 14 = 8 bits. This would increase the page table size to:

2^{11} × 10 bits = **20480 bits**

This increases the page table by 5 times, wasted valuable hardware to store the extra page table bits.

(c) Yes, this is possible. In order for concurrent access to take place, the number of set + block offset + byte offset bits must be less than the page offset bits.

(d) It is impossible to perform the tag comparison in the on-chip cache concurrently with the page table access because the upper (most significant) bits of the physical address are unknown until after the page table lookup (address translation) completes.