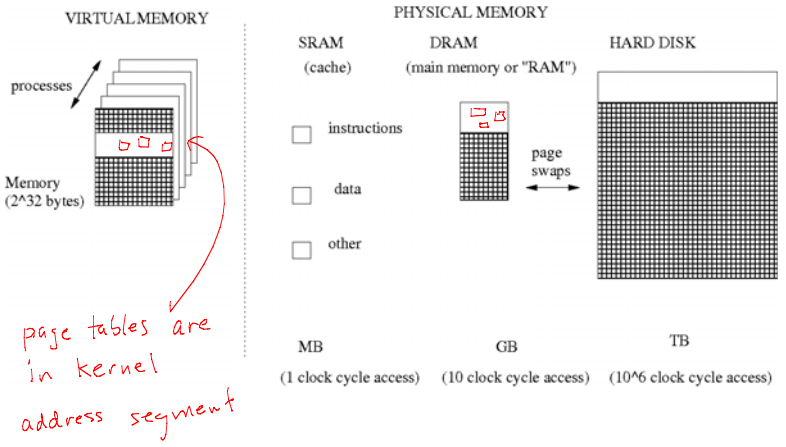
lecture 18

Cache 1

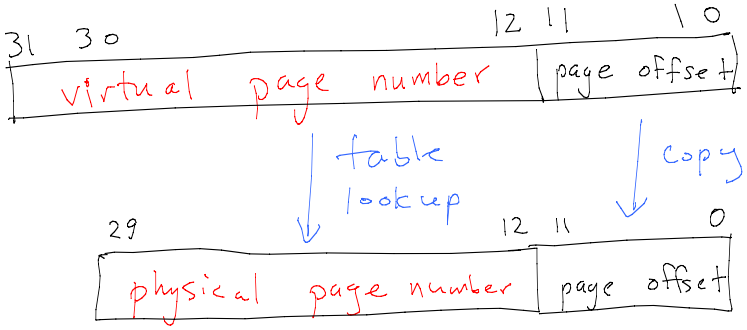
review: last lecture

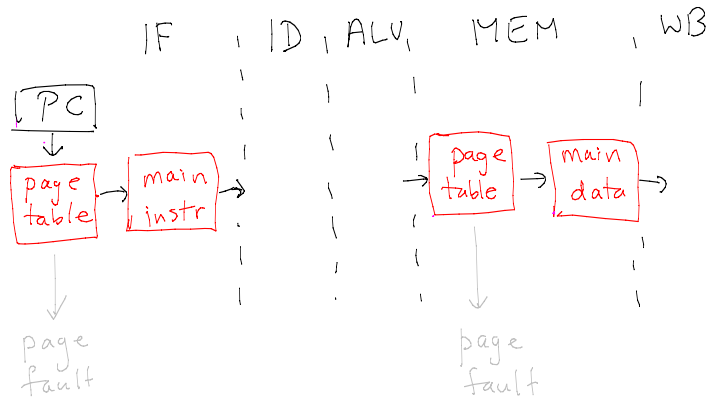


page tables are in kernel address segment

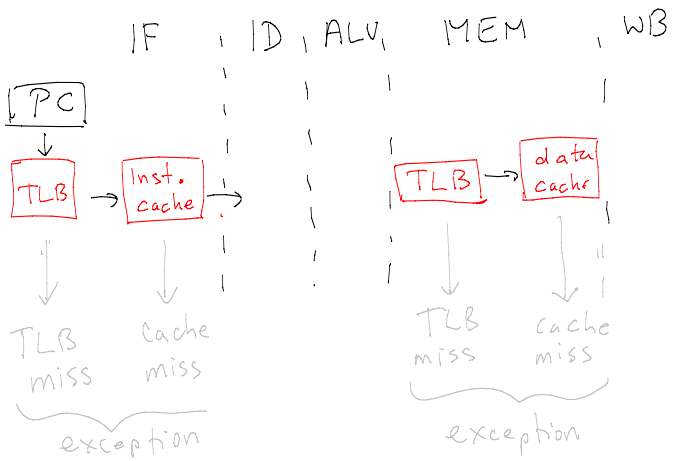
Translation:

virtual address. ⟶ physical address in main memory - RAM





Main memory (DRAM) is too slow!

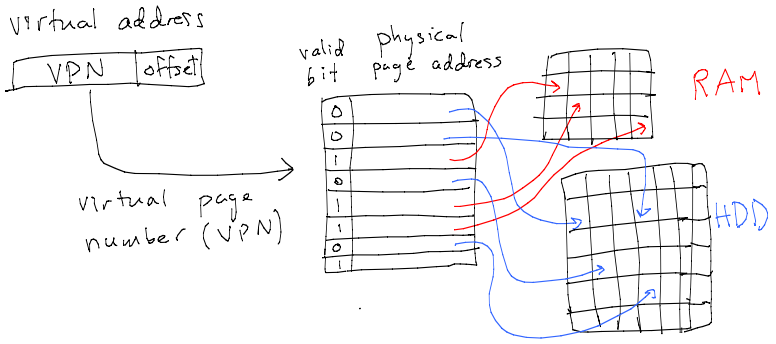


TLB & cache (SRAM) can be accessed in 1 clock cycle.

Today

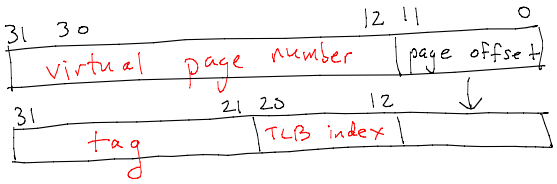
- TLB
- cache (read, next lecture we do write)
- direct vs. associative map (full vs. set)

Recall how page table is organized and indexed.

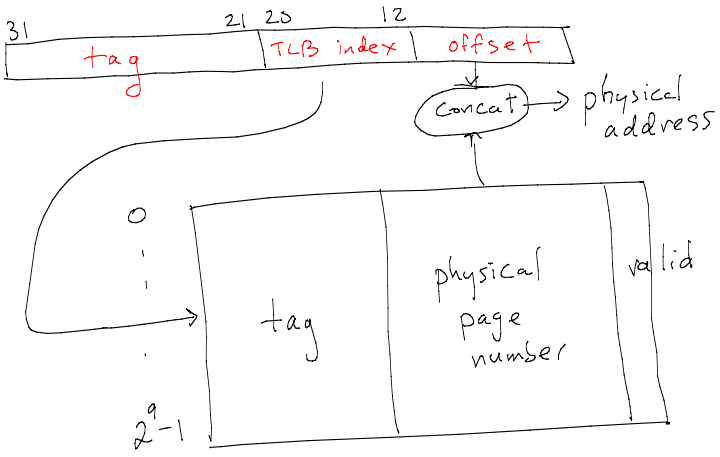


How is the TLB organized?

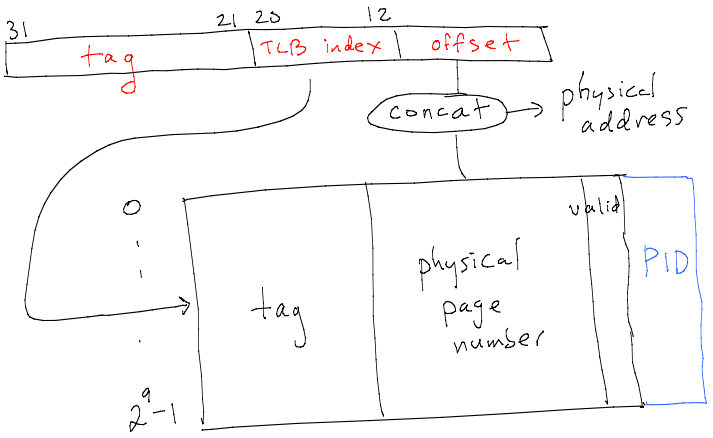
Suppose TLB has $2^9 = 512$ entries.

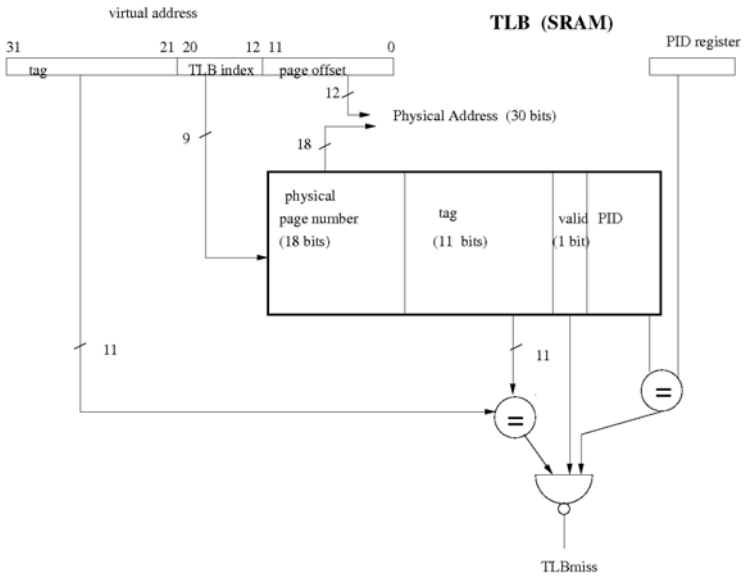


TLB index = VPN mod number of TLB entries



Recall there are many processes, each with own page table. Thus, add a process ID field (PID).

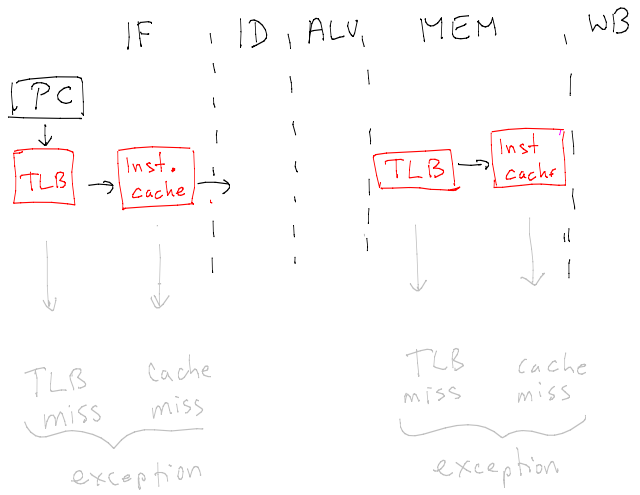




Today

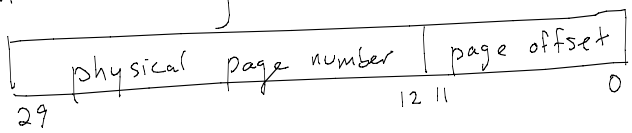
- TLB
- cache (read, next lecture we do write)
- direct vs. associative map (full vs. set)

Next, consider data and instruction caches.



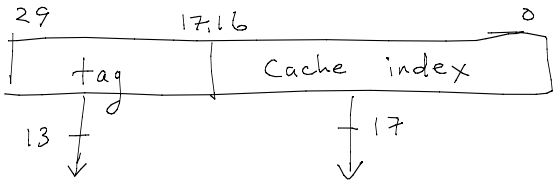
Suppose:

- main memory has 1 GB.

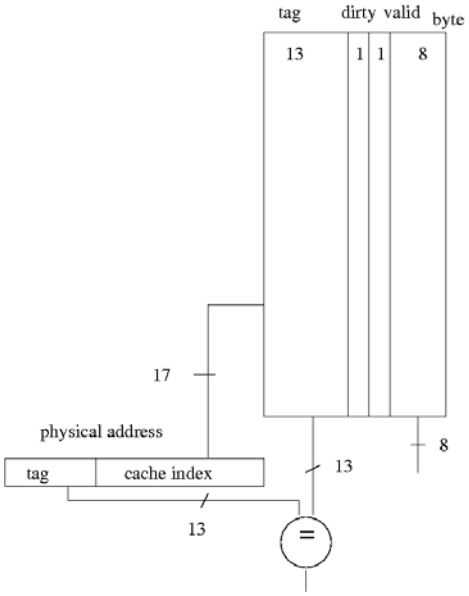


- instruction and data cache each have 2^{17} bytes (128 KB).

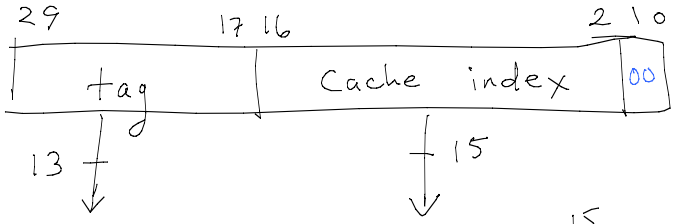
Example 1: one byte a time



Can address 2^{17} individual bytes in cache

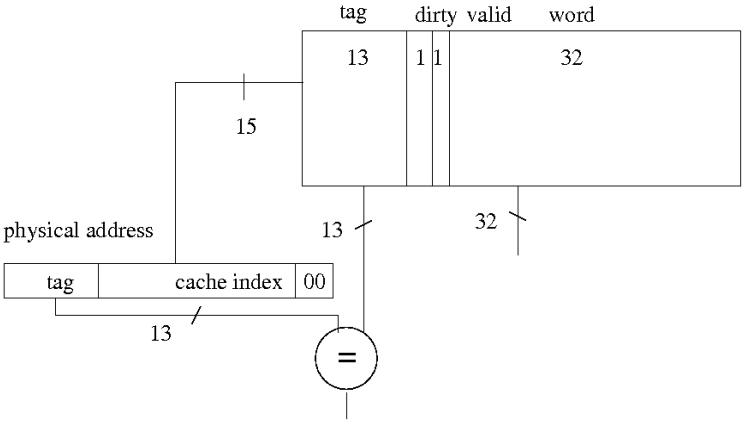


Example 2: one word a time

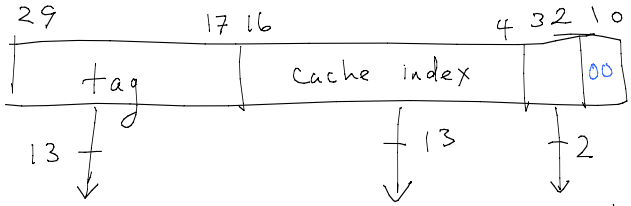


Can address 2^{15} words in cache

(assume word aligned)



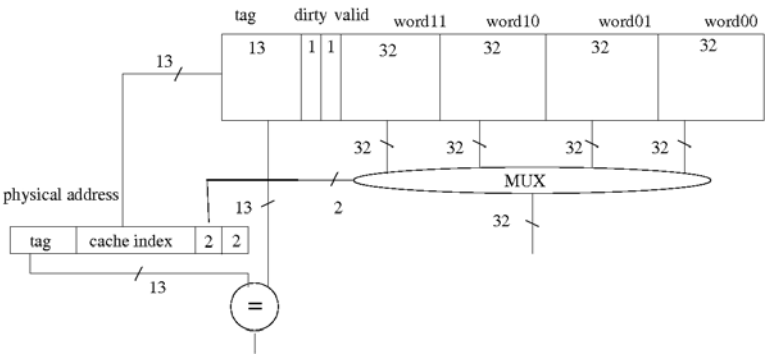
Example 3: one block at a time

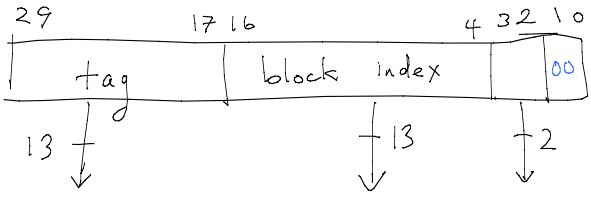


Can address 2^{13} blocks (4 words per block)

word selector in block

A memory block is a set of adjacent words (block aligned)





word selector in block

main memory block number = main memory address / bytes per block

block index = main memory block number mod number of blocks in cache

Today

- TLB
- cache (read, next lecture we do write)
- direct vs. associative map (full vs. set)

↑ what we've been doing

Fully associative cache

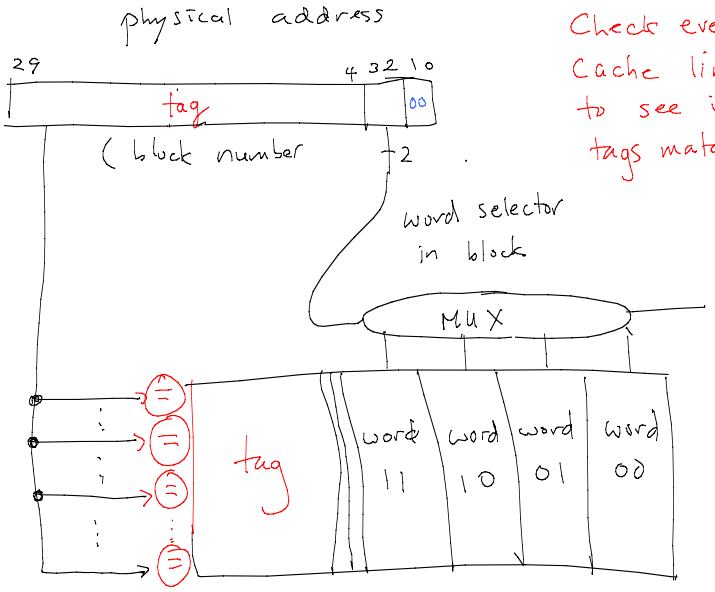
Allow a block to be in any cache line.

The tag is the block number!

Check every cache line to see if tags match

✓ Maximizes chances of hit

✗ Lots of extra circuitry needed

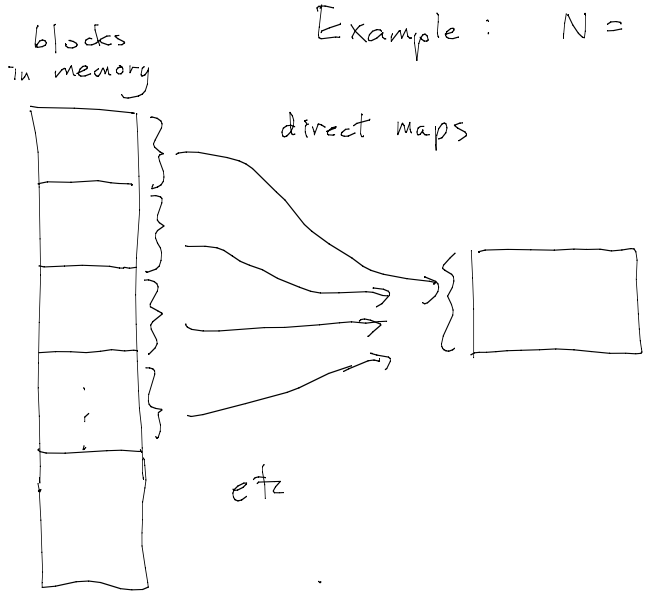


Check every cache line to see if tags match

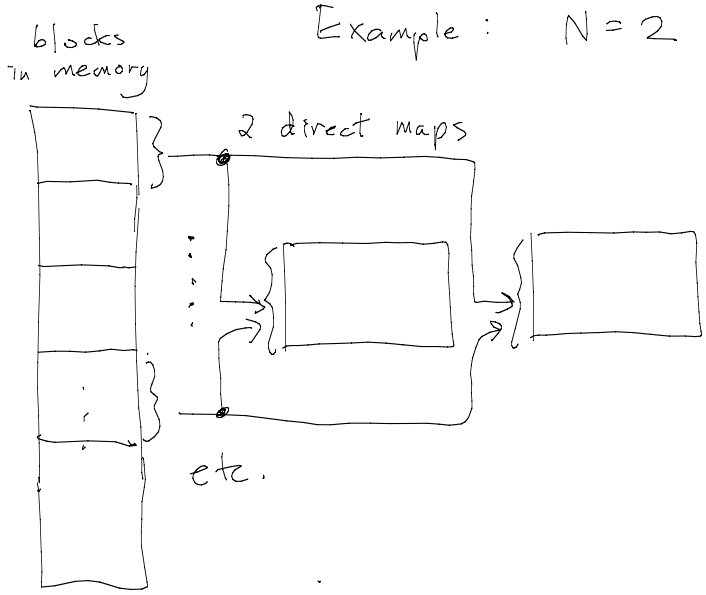
Less Radical Solution

- use N direct maps ("N-way set associative")
- a block can go into any of N possible cache lines

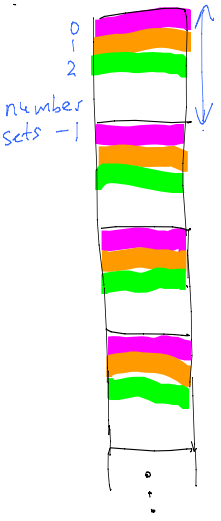
Example: N = 1



Example: N = 2



blocks in main memory

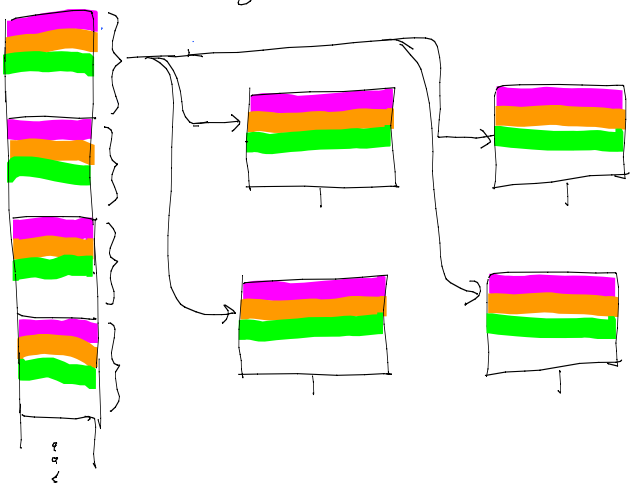


number of sets = number blocks in cache / N

Each set can have up to N blocks in the cache.

N = 4 way associative mapping

blocks in main memory



set index (0, 1, 2, ...) ≡ block number mod number of sets

block number in main memory