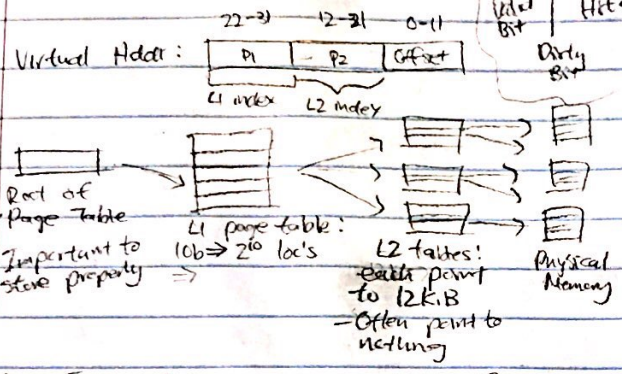
2^{14} pages 5 status bits
46b VA's
2^{32}B mem
2 way SA (TLB)

Lecture 30: Virtual Memory II

Hierarchical Page Tables

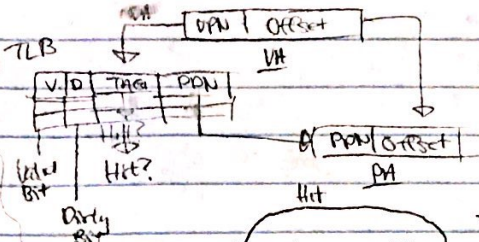
- 32 bit VA, 4 KiB Pages
  - For single page table: 2^{22} B
  - For 2^8 processes (each using own page table) $\Rightarrow 2^8 \cdot 2^{22} = 2^{30}$ B
- ↑ Page size ⇒ ↓ Page table size
  - Requires wasted memory
- Hierarchical Page Table more common solution — decreasing page size



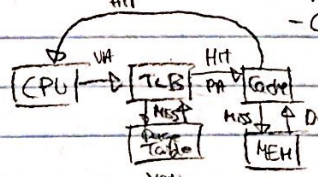
- Page Table Entry: contains 32b for status bits of protection & PPN[0], PPN[1] which are the physical page #'s corresponding to the virtual address

Translation Lookaside Buffers (TLB)

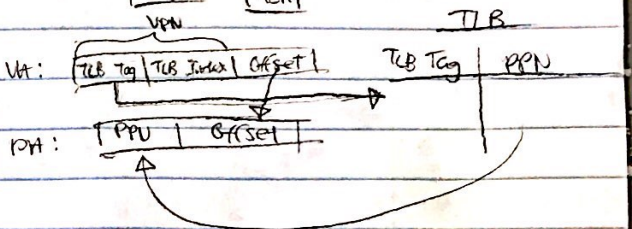
- Idea: Go from VA to PA through addr translation
  - Requires protection checks (R/W? raise exceptions)
- Addr translation: 1 M page table (2 mem accesses), 2 M (3 accesses)



Idea: Cache translations in TLB
- TLB design: 32-128 entries (fully associative)
- Random/FIFO replacement
- TLB reach: largest VM space size simultaneously mapped by TLB
- Check TLB before cache



- TLB miss? Need to use the page table & update TLB

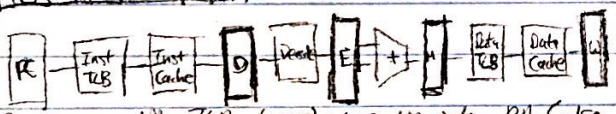


PA then split into T/I/O bits to reference cache

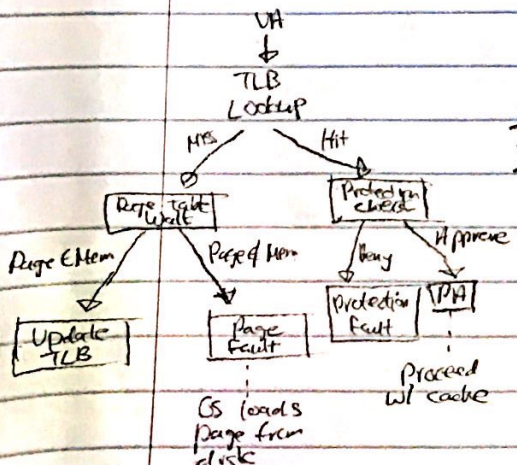
PA: Tag | Index | Offset

TLB in Datapath

Incorporate mem OVER, controller, Table walker, Page



- PC issues VA, TLB translates VA into PA (also check for miss, page fault, & violation) → stored in cache
- Mem req issues VA & same process occurs
- TLB miss? Hardware to fill the TLB
- Page fault? Use a precise trap
- Protection violation? Abort process



VM Performance

Cache	VM
Block/Line	Page
Miss	Page Fault
Block: 32-64B size	4k-8kiB
DMC, N-way SAC	FA
LRU/Random Replacement	LRU/FIFO/Random
WT or WB	WB

★ Parameters: L1 (HT=1 cycle, HR=0.95), L2 (HT=10 cycles, HR=0.6), DRAM=200 cycles, Disk = 20 million cycles

- No paging, AMAT = 1 + 0.05 · (10 + 0.4 (200)) = 5.5 cycles
- With paging AMAT = 5.5 + 0.05(0.4)(1 − HR_Mem) · 20x10^6
  - Very steep penalty for accessing the disk

How to run multiple programs @ once?

- Use context switch: toggle b/w multiple processes
  - Save reg values & change value in Supervisor Page Table Base Reg (SPTBR)
  - Invalidate current values in TLB b/c for diff. process