

Quick Review: Caches

Your Uber Nintendo console has 64KiB of physical memory and a 512B direct mapped cache with 16B blocks. How big are the T, I, and O fields?

The other way now, suppose you know a physical address splits into the following fields:

Find the parameters listed above assuming a direct-mapped cache.

TTTTT	IIIIIII	OOOOO
5 bits	7 bits	5 bits

Average Memory Access Time

$$AMAT = Hit\ Time + Miss\ Rate * Miss\ Penalty$$

The Three Cs

Compulsory Miss –

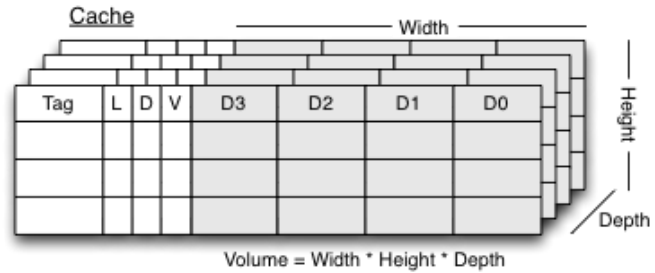
First time accessing data (e.g. startup)

Conflict Miss –

Two blocks map to the same cache line

Capacity Miss –

Just not enough room in the cache



Cache Knobs

Knob	Values	Effects
Block Size	Powers of 2	
Associativity	Direct Mapped N-Way Fully Associative	
Cache Size	Powers of 2	
Write Policy	Write Back Write Through	
Replacement Policy	LRU Random Other	
RAM Size	Powers of 2	

RAM

0x00c	0x008	0x004	0x000
0x01c	0x018	0x014	0x010
0x02c	0x028	0x024	0x020
0x03c	0x038	0x034	0x030
...
...
0x7cc	0x7c8	0x7c4	0x7c0
0x7dc	0x7d8	0x7d4	0x7d0
0x7ec	0x7e8	0x7e4	0x7e0
0x7fc	0x7f8	0x7f4	0x7f0

Cache Exercises

1) Assuming 32 bits of physical memory, how many total bits of storage are required for an 8-way set associative 4KiB cache, if blocks are 16B and it uses write back and LRU replacement?

2) You want your AMAT to be ≤ 2 cycles. You have two levels of cache.

L1 hit time is 1 cycle. L1 miss rate is 10%

L2 hit time is 3 cycles L2 miss penalty is 100 cycles

What does your L2 miss rate need to be?

3) Look at final questions F2 from Fa05 and F2 from Fa06. (on your own time)

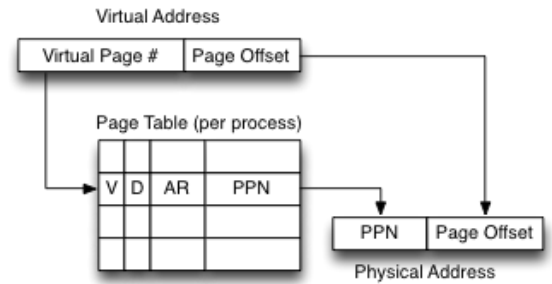
Virtual Memory

What are three specific benefits of using virtual memory? [there are *many*]

Virtual to Physical Address Translation

Page Table

Split memory into a bunch of equal sized chunks called pages. Better still, split a section of disk into these chunks as well. Map any virtual page into a physical page via table look up -- the page table. Each process gets its own page table.



Translation Lookaside Buffer (TLB)

A cache of page table entries. Each block is a single page table entry. If an entry (corresponding to a virtual page) is not in the TLB, it's a TLB miss. If that entry is also not in memory (it's been paged out), it's a page fault. Who gets kicked out on a page fault? What happens if there just isn't enough physical memory?

The TLB and the Page Table together make up a translation unit that maps from virtual addresses to physical addresses.

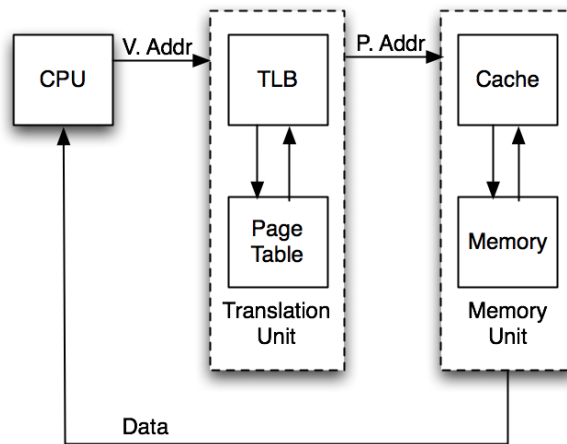


Figure 1: Logical Flow "Big Picture"

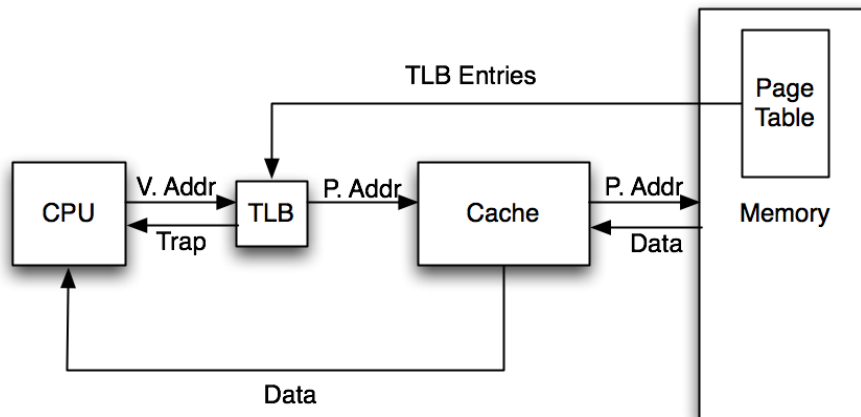


Figure 2: Physical Implementation "Big Picture"

