Tutorial: Caches

Bookmark this page

Caches: 1

2/2 points (ungraded)

Two otherwise-identical Beta systems have 1024-line two-way set-associative caches using LRU and FIFO replacement, respectively. For what programs does the **FIFO** cache show a higher hit ratio than the LRU cache?

FIFO outperforms on:

- ○ **ALL**
- ○ **MOST**
- ◉ **A FEW**
- ○ **NONE**

✔

Two otherwise-identical Beta systems have fully-associative LRU caches. One cache has four lines, each caching a single 32-bit word; the other cache has eight lines, each also holding a single 32-bit word. For what programs does the **smaller** cache show a higher hit ratio?

***Small* cache outperforms on (circle one):**

- ○ **ALL**
- ○ **MOST**
- ○ **A FEW**
- ◉ **NONE**

✔

Submit

Caches: 2

4/4 points (ungraded)

1. You are evaluating two tiny 4-line caches, each with a total storage of four data words. Model DM is direct-mapped, and model FA is fully associative (LRU). Each uses **word addressing** (hence consecutive addresses differ by 1, not 4 as in the Beta). The benchmark involves just six memory reads, starting with an empty (invalidated) cache. Finish the reference string given below to yield an access pattern that will give a **better hit rate on DM than on FA**. Fill in the final two additional memory addresses. **Use single-digit decimal addresses.**

   0, 1, 2, 3, 5 ✔ , 0 ✔

2. Same setup as above; this time give references that make FA look better. Again, **use single-digit decimal addresses.**

   0, 1, 2, 3, 5 ✔ , 1 ✔

Submit

Caches: 3

9/9 points (ungraded)

You are considering three possible caches for use in a Beta-like computer system that uses 24-bit byte add
to access 32-bit (4-byte) words in main memory – hence each word address is divisible by four (as in the Be

Calculator

Each of the caches you are considering has a total of 64 cache lines, each holding one 32-bit data word. The three cache models are as follows:

Cache	Description
DM	Direct-mapped, 64 lines
4Way	4-way set associative: 16 sets of 4 lines each.
FA	Fully-associative: one set of 64 lines.

To compare caches, you have collected some additional parameters about each model and have run a tiny benchmark program to measure the performance of each. Your benchmark program is a tight loop that repeatedly accesses the **4** locations given by the **hex byte addresses**:

0×100, 0×200, 0×104, 0×108, 0×100, 0×200, 0×104, 0×108, 0×100, ...

You have been asked to complete the following table. Fill in blank entries; use “NA” (Not Applicable) for inapplicable parameters.

Assume that the incoming address from the CPU is A[23:0], that each cache stores the minimum number of address bits in the tag field for each line. The “# Lines for Mem[0]” column lists the number of lines in each cache that might hold the contents of main memory location zero, and the Hit Rate column reflects each cache’s steady-state performance on the little benchmark described above.

Cache Model	Bits of A[23:0] used to address cache	# Tag bits Per line	Replacement Strategy	# Lines for Mem[0]	Hit Rate
DM	A[7:2]	16 ✔	NA ✔	1	50% ✔
4Way	A[5 ✔ : 2 ✔]	18 ✔	LRU	4 ✔	100% ✔
FA	NA	22	LRU	64 ✔	100%

Submit

Discussion

Topic: 14. Caches and the Memory Hierarchy / Tutorial: Caches

Hide Discussion

Add a Post

Show all posts by recent activity

? Caches:2 — 4
I need more details on this question

I want to discuss a question here ,need more explanation ,Kindly Help! — 4
Here is the Question : > Assume that the CPI of a computer is 1 when all memory accesses hit in > the cache. If 30% of the instructio...

Hit Ratio format — 1
Note that the hit ratio should either be a number between 0 and 1 (f.e. 0.34), or a percentage including the % sign (f.e. 75%).

Calculator

< Previous Next >

edX

edX

About
Affiliates
edX for Business
Open edX
Careers
News

Legal

Terms of Service & Honor Code
Privacy Policy
Accessibility Policy
Trademark Policy
Sitemap

Connect

Blog
Contact Us
Help Center
Media Kit
Donate

GET IT ON Google play
Download on the App Store

Calculator