Discussion 7

Feb 20th

Outline

- Week 7 Quiz
- Overview on DINOCPU assignment 4

Week 7 Quiz

Question 1

1 pts

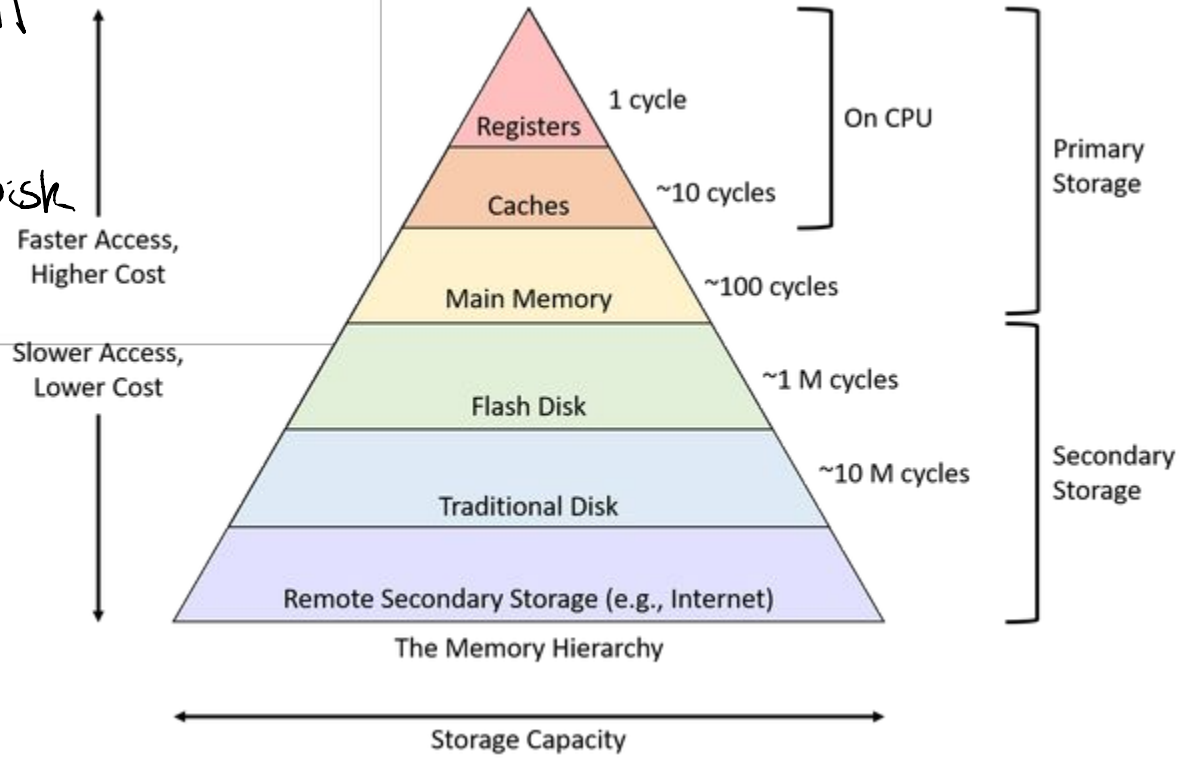
Rank the following memory technologies from fastest to slowest (lowest to highest *latency*).

Fastest [Select] SRAM

Second Fastest [Select] DRAM

Third Fastest [Select] Flash

Slowest [Select] Spining Disk



Question 2

1 pts

It doesn't matter what size the memory array is, the latency to access a bit only depends on the cell technology.

- ○ True
- ✓ False

Question 3

1 pts

Mark the statements that are true *for all volatile memories*.

- ☑ Data can be lost when the power is removed
- ☑ They can only be used for temporary data
- ☐ Data must be refreshed because charge leaks
- ☐ They are faster than non-volatile memories

Question 4

2 pts

A *monochrome* LCD or LED screen is a lot like a memory array. There are a set of "cells" each of which is addressable and can be set to either 1 (on) or 0 (off). The framebuffer is a region in memory which is *mapped* to the screen.

If you have a screen which is 1920x1080 (1080p-ish), how many *bytes* are required for the framebuffer? $\frac{1920 \times 1080}{8\ \text{bit/byte}} = 259{,}200\ \text{bytes}$

What if each cell (pixel) required 3 8-bit values to set the red, green, and blue values? How large would this framebuffer be if each pixel required 24-bits *in bytes*? $\frac{1920 \times 1080 \times 24}{8} = 6{,}220{,}800\ \text{bytes}$

You want to run your monitor at 90Hz. What is the *bandwidth* required in MiB/s to transfer the raw data from the framebuffer to the monitor? (round to the nearest MiB) $\frac{6{,}220{,}800}{1024 \times 1024} \times 90 = 533.9 \approx 534\ \text{MiB/s}$

$\text{Hz} = \frac{1}{s}$ $\frac{1000,000}{1024 \times 1024}\ \text{bytes}$

Finally, data movement requires energy. Let's assume that to transfer a bit requires 2 nJ. What is the *power* to drive the video to your monitor in Watts? (round to the nearest Watt)

$1\ \text{Watt} = 1\ \text{Joule/s}$ $2\text{nJ} \times 8 = 16\text{nJ}$

$534\text{MiB} \times 16\text{nJ} = 8.95 \approx 9\ \text{Watt}$.

NOTE: Give all answers with just numbers and no units.

Question 5 — 1 pts

Normal hardware caches are *transparent* to the programmer. This kind of design would be considered part of the [Select] microarchitecture.

Scratchpad "caches", which are often small arrays of SRAM, are exposed to the programmer. This kind of design would be considered part of the [Select] architecture.

Question 6 — 1 pts

Given the following address streams name the kind of locality they exploit

`0x31c4 | 0x31cc | 0x31d4 | 0x31dc | 0x31e4` ← 16 byte.

[Select] spacial locality.

`0x31c4 | 0x7808 | 0x31c4 | 0x7808 | 0x31c4` ←

[Select] temporal locality.

0x3210
0x321f

0x31e0
0x31ef

0x3220
0x322f

0x31d0
0x31df

Question 7

1 pts

Increasing the block size of the cache will help when there is what kind of locality?

- ○ Temporal
- ○ Associative
- ✓ Spatial

hex # = 0000

Question 8

2 pts

There are two cache systems. System A has a block size of 16 Bytes and System B has a block size of 256 Bytes. Assume the caches are initially empty and have a large capacity.

Given the following address stream, calculate the hit ratio for each cache

`0x3214 | 0x31e8 | 0x31e8 | 0x3220 | 0x31dc | 0x31e0 | 0x320c | 0x31d8 | 0x3224 | 0x31f8`

System A has a block size of 16 B. What is the hit ratio? [Select] $\frac{4}{10} = 0.4$

System B has a block size of 256 B. What is the hit ratio? [Select] $\frac{8}{10} = 0.8$

0x3200
0x32ff

0x3100
0x31ff

16 ⇒ $\log_2(16)$ = 4 bits.

$\log_2(256)$ = 8 bits.

Question 9

Use the following system characteristics.

L1 cache latency: 4.0 ns

L1 cache hit ratio: 82%

Memory latency: 60 ns

What is the average memory access time in ns?

AMAT = HT + MP.

4ns x 82% + 18%((4+60)ns)

= 14.8ns.

Question 10 1 pts

Use the following system characteristics.

L1 cache latency: 2 cycles

L1 cache hit ratio: 89%

Memory latency: 65 cycles

Instruction mix: 60% loads and stores

Assume 0 time for instruction fetch (for instance, there is a very good instruction prefetcher) and assume that all instructions other than loads and stores complete with a CPI of 1.

What is the CPI of this system with an L1 cache?

40% + 60%(89%x2 + 11%(2+65))

= 5.89.

Question 11

2 pts

Use the following system characteristics to answer the questions below.

L1 cache latency: 2 cycles

L1 cache hit ratio: 95.0%

Memory latency: 40 cycles

Instruction mix: 40.0% loads and stores

System A = 60% + 40%(2 × 95% + 5%((2+10)×80% + 20%(2+10+40))

= 1.76.

SystemB = 60% + 40%(2×95% + 5%((2+15)×90% + 10%(2+15+40))

= 1.78.

To improve performance you add another level of cache. You have two options:

	L2 Cache A	L2 Cache B
Hit ratio	80.0%	90.0%
Latency	10 cycles	15 cycles

$$\text{Speedup} = \frac{A}{B} = \frac{1.76}{1.78} = 0.98$$

What is the *speedup* of **System A over System B**? (If system A is better, then this should be above 1, if system B is better then this should be below 1). IPC.

Assume 0 time for instruction fetch (for instance, there is a very good instruction prefetcher) and assume that all instructions other than loads and stores complete with a CPI of 1. Also assume that the two systems are running the same program.

Hint: you need to calculate the CPI for each system.

Question 12

2 pts

Use the following system characteristics.

L1 cache latency: 3 cycles

L1 cache hit ratio: 88%

Memory latency: 50 cycles

Instruction mix: 0.4% loads and stores

Single level = 60% + 40%(3x88% + 12%((3+50))

53 = 60%(x+3) + 40%(x+3+50)

53 = (x+3) + 20.

x = 30 cycles.

Now, assume that you are going to design a second level of cache. The design will have a *hit ratio* of 60%.

What is the *latency* required of the L2 cache to give *the same performance* as the single-level system?

Question 13

2 pts

Use the following characteristics to answer the questions below.

Cache capacity: 1024 KiB

1024 × 1024 bytes.

Block size: 32 B

total cache block = $\frac{1024 \times 1024}{32} = 32768$ ←

Address size: 33 bits

Associativity: None, direct-mapped

What bits of the address are used to access the cache? Answer using chisel syntax (e.g., the lowest order 4 bits would be "(3,0)").

$\log_2(32) = 5$ bits. offset.

$\log_2(32768)$ bit ⇒ 15 bits index.

Tag: (32,20)

Offset into the block: (4,0)

33 − 5 − 15 = 13 bits ⇒ tag.

Index: (19,5)

Question 14

2 pts

Use the following characteristics to answer the questions below.

Cache capacity: 32 MiB

Block size: 8 B

Address size: 34 bits

Associativity: 16-way set associative

$\#\text{block} = \frac{32 \times 1024 \times 1024}{8} = 4194304$

$\#\text{entrence} = \frac{4194304}{16} = 262144$ entrences.

$\log_2(262144)$ bits $\Rightarrow$ index.

$= 18$ bits.

What bits of the address are used to access the cache? Answer using chisel syntax (e.g., the lowest order 4 bits would be "(3,0)").

$\log_2(8) = 3$ bits $\Rightarrow$ offset.

$34 - 3 - 18 = 13$ bits tag.

Tag: (33,21)

Offset into the block: (2,0)

Index: (20,3)

Use the follo.

Block size: 8 B ⇒ 3 bits offset.

Total blocks: 4096

$\frac{4096}{8} = 512$ entrences.

Associativity: 8-way Set asso.

$\log_2(512)$ bits $= 9$ bits. ⇒ index.

Address size: 33 bits

Meta data: 3 bits per block

$33 - 3 - 9 = 21$ bits. ⇒ tag.

Give the following answers *in bytes* unless otherw.

What is the capacity of the *data* array? $4096 \times 8B = 32768$ bytes.

How many tags are required? 4096.

What is the capacity of the SRAM (including the tags and extra meta data): ,

$\frac{(21+3)\times 4096}{8 \text{ bit/bytes}} + 32768 = 45056$ bytes.

How many *bits* are read on each access to the cache? Assume all ways are accessed in pa.

$(8 \times 8 \text{ bits}_{\text{bytes}} + 21 + 3) \times 8 = 704$ bits.

Question 16

1 pts

Which replacement policy will perform the best?

- ○ Least recently used
- ○ Random
- ✓ It depends on the workload
- ○ Most recently used
- ○ Pseudo least-recently used

Question 17

1 pts

Which policy has the *least* hardware overhead?

- ○ It depends on the workload
- ✓ Random
- ○ Most recently used
- ○ Least recently used
- ○ Pseudo least-recently used

Question 18

1 pts

Which of the following are benefits of a write-*through* policy?

- [ ] Lower latency for writes
- [x] Easier to handle faults in the cache
- [x] Less metadata in the cache
- [ ] Lower memory bandwidth

Question 19

1 pts

Assume that after analyzing your workload and cache design, you find that most of the misses are due to *capacity* misses. Which of the following *will not* improve the hit ratio.

- ( ) Increasing the capacity of the cache
- (x) Increasing the set associativity
- ( ) All of the above
- ( ) Increasing the block size of the cache
- ( ) None of the above

DINOCPU Assignment 4