CPSC 2310

Homework – Learning about Cache

Due: Sunday, August 29, 2021

Submission: Canvas

Don’t panic!! This is a short article and 12 questions to answer. Approximately 1 hours worth of work

Read the following article and complete the assignment as described below:

<https://www.extremetech.com/extreme/188776-how-l1-and-l2-cpu-caches-work-and-why-theyre-an-essential-part-of-modern-chips>

I also uploaded a pdf of the article.

ALL ANSWERS MUST BE IN RED. SUBSTANTIAL POINTS WILL BE DEDUCTED IF YOUR ANSWERS ARE NOT IN RED. You may be asking yourself why I insist on the answers being in red. There are 120+ students between the 2 sections. Grading is hard and time consuming. It is much easier to distinguish your answers if the answers are in RED.

This is a 2 part assignment you must complete both parts.

The author of this article Joel Hruska makes the claim that “caches and caching is one of the most significant events in the history of computing.”

Part1:

**In addition to answering the questions below. You are to read the article, then in your own words explain why you think the author made this claim and do you agree, why or why not.** This part of the assignment cannot be answered in one sentence. It is actually going to required you to think and articulate your answer and opinion.

The author concluded his article by claiming that cache design, consumption, and performance is going to be critical to future processors. He also claims that the company that improves cache designs will have their status greatly boosted. The author made this claim because he claims that cache designs were essential in speeding up memory and improving performance.

In this article Hruska stated that memory was slow and expensive, and that CPUs were relatively slow as well. Caches were critical in solving this problem. He then goes on to show how the cache design can impact the performance and that with a poor cache design, the time needed to execute code may be very high. I agree with the authors claim that cache design is crucial to future processors and is one of the most significant events in computing. As more and more of our lives are dependent on computers and software, having computer systems that operate on a high level of performance and speed is critical. I also agree with his claim that the company that improves cache performance is going to gain immense status. Like I said, before computers and software are already a constant in our lives, so a company that improves the things so important to our lives will have a firm hold on a lot of the market.

Part2:

While you are reading the article, answer the following questions:

1. What are CPU Caches?
   1. They are small pools of memory that store the information that the CPU is most likely going to need next.
2. What determines what information is loaded into cache?
   1. The information loaded onto the cache is determined by sophisticated algorithms and assumptions about programming code.
3. What is a cache hit and a cache miss?
   1. A cache hit is when the next data that is needed by the CPU is already loaded into the cache when the CPU goes looking for it. A cache miss is when the CPU must go finding the data that it needs next outside of the cache.
4. Hruska touched on two types of cache design used by processors; inclusive and exclusive. Describe each.
   1. In an inclusive cache design, the data stored in the L1 cache is also duplicated and stored in the L2 cache. In an exclusive cache design, the L1 and L2 caches never share data.
5. The Hit Rates for Constant L1, Increasing L2 chart, depicts the relationship between an L1 cache with a constant hit rate, but a larger L2 cache. **In your words**, explain this chart.
   1. The chart shows that as the size of the L2 cache increases, the overall hit rate goes up significantly before flattening out. The L1 hit rate stays consistent as its size gets larger, so the increase in size of the L2 cache is what significantly increases total hit rate.
6. The author mentioned the term “die” or “on-die” several times. Personally, I had no idea what that meant, so I looked it up. If you do not know what this means you will need to look it up as well. Explain what “die” or “on-die” is referring to.
   1. Essentially “on-die” means that the L2 cache is on/fabricated to be on the same chip as the CPU/processor. The “die” is in reference to the rectangular pattern on the silicon wafer that contains circuit. The wafer with the die on it is then cut and made into chips.

**FYI:** another article I found interesting when writing and answering these questions can be found at: <https://www.geeksforgeeks.org/cache-memory-in-computer-organization/>

You are not required to read this one, but it is interesting and cleared up some questions I had when reading Hruska’s article.

1. What is “tag RAM”?
   1. “tag RAM” is a record of all of the memory locations that can map to any given block of cache.
2. What does it mean when cache is fully associative and what is the advantage?
   1. Fully associative means that any block of RAM data can be stored in any block of cache. The advantage of a fully associative cache is that the hit rate is high.
3. What is direct-mapped caches?
   1. Direct-mapped caches are caches that where each cache block can contain only one block of main memory. It can be searched very quickly, but it has a low hit rate because of the 1:1 maps to memory locations.
4. Why do CPU caches keep getting larger?
   1. Adding additional memory pools push back the need to access main memory and can improve performance in certain cases.
5. Summarize the section titled: How Cache Design Impacts Performance. Do not copy and paste this section. Read it and summarize what it means.
   1. Adding a CPU cache can significantly impact performance. If the data is not found in the current cache, trying to go to the next cache to find the data can significantly increase the time it takes to execute code. The section also goes into detail about cache contention. Cache contention is when two different threads are overwriting data in the same memory. This can hurt performance, as both threads are constantly overwriting the other’s data. Also, the section talks about the modern L0 cache. This is a very small cache used for storing micro-operations. They use the same principles as L1 and L2 but they just have a very small pool of memory.