Homework 4:
A New Golden Age for Computer Architecture

Pupipat Singkhorn

November 18, 2024

Introduction

We live in a world that is constantly changing. As you graduate from this course, you should keep up with current trends and emerging techniques. This assignment serves as a practice for your lifelong learning journey. You will read a review article that talks about the past, the present, and the future of computer architecture. Even though the past might seem boring and irrelevant at first, you will notice that many of the techniques in the past keep reappearing in new forms. This is true in many fields of study. You will find many concepts that we learned in the course, and it should serve as a good review. This exercise will have you read and answer questions based on a recent article by computer architecture experts, John L. Hennessy and David A. Patterson. You can access the article by going to the materials section in MCV. Lastly, important concepts from this article will also appear in the final exam.

Questions

1. According to the CPU time formula, why is RISC better than CISC?

   Answer:

   $$\text{CPU Time} = \text{Instruction Count} \times \text{Cycles Per Instruction (CPI)} \times \text{Cycle Time}$$

   (a) Lower CPI: simpler instructions, but at the cost of the number of instructions.

   (b) Faster clock: shorter cycle time.

2. Why did the 'Itanic' ISA become unused?

   Answer: It struggled to achieve high performance for integer programs that had less predictable cache misses or less-predictable branches. It also faced challenges with x86 compatibility and market adoption.

3. What type of parallel architecture according to Flynn's taxonomy is Itanic?

   Answer: Multiple Instruction Multiple Data (MIMD)

4. Explain 'Dennard scaling' in your own words.

   Answer:
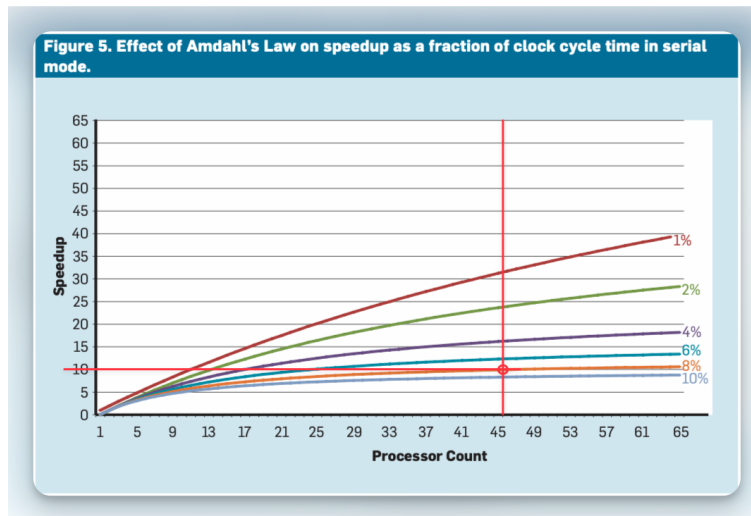 $$\text{Power density} = \frac{\text{Power } (\downarrow)}{\text{Area } (\downarrow)} = \text{Constant}$$

   Transistors get smaller(area decrease), their power density(power/area) stays constant.

5. Why is it not practical to keep increasing the number of pipeline stages and thus increasing the ILP?

   Answer:

   (a) **Branch Misprediction Inefficiency**: Increasing ILP amplifies the impact of branch mispredictions, leading to wasted computational work and energy.

   (b) **Accuracy Challenge**: Achieving high prediction accuracy is particularly difficult for general-purpose programs with unpredictable branching patterns.

   (c) **Limited Practical Gains**: Frequent mispredictions reduce the performance and energy efficiency benefits of increased ILP, making it less practical for most workloads.

6. According to Figure 5, when 8% of the time is serial, what is the percentage of energy wasted by a 45-processor configuration?

   Answer:

   Figure 5. Effect of Amdahl's Law on speedup as a fraction of clock cycle time in serial mode.

   $$\therefore \text{Energy Wasted} = 1 - \frac{10}{45} = 0.7778 = 77.78\%$$

7. Why can VLIW be a good fit for DSA?

   Answer: VLIW (Very Long Instruction Word) suits DSAs (Domain-Specific Architectures) as it exploits predictable parallelism, simplifies hardware by relying on static scheduling, and achieves energy efficiency, making it ideal for specialized, repetitive workloads.

8. What is your impression regarding Figure 7? Does it change your view of how to program? What are the things that you should be aware of in order to write more efficient software?

   Answer:

   Impression of figure 7.
   It shows a dramatic and significant performance improvement, with approximately a 10x speedup from Python to C and up to a 60,000x boost using SIMD optimization.

   Does it change my view of how to program?
   Yes, it reinforces the importance of understanding the underlying hardware and choosing the right tools and techniques for computationally intensive tasks.

   Things to be aware of to write more efficient software.

   (a) Choose the right language

   (b) Leverage parallelism

   (c) Optimize memory usage

   (d) Hardware-specific features

9. Explain what systolic arrays are. How are systolic arrays different from typical SIMD architectures? When were they first created? You should read other sources to get a better understanding of them. Write down the sources you used.

   Answer:

   Explain what systolic arrays are.

   How are systolic arrays different from typical SIMD architectures?

   When were they first created?

   Sources: In the references section.

10. Explain the similarities and differences between Agile software development and Agile hardware development.

References

1. https://cs.stanford.edu/people/eroberts/courses/soco/projects/risc/risccisc/

2. https://en.wikipedia.org/wiki/Systolic_array

3. https://en.wikipedia.org/wiki/Flynn%27s_taxonomy

4. https://en.wikipedia.org/wiki/Dennard_scaling

5. https://en.wikipedia.org/wiki/Itanium

6. https://chatgpt.com/