Appendix H reviews VLIW hardware and software, which, in contrast, are less popular than when EPIC appeared on the scene just before the last edition.

Appendix I describes large-scale multiprocessors for use in high-performance computing.

Appendix J is the only appendix that remains from the first edition, and it covers computer arithmetic.

Appendix K provides a survey of instruction architectures, including the 80x86, the IBM 360, the VAX, and many RISC architectures, including ARM, MIPS, Power, and SPARC.

We describe Appendix L below.

1.13 Historical Perspectives and References

Appendix L (available online) includes historical perspectives on the key ideas presented in each of the chapters in this text. These historical perspective sections allow us to trace the development of an idea through a series of machines or describe significant projects. If you're interested in examining the initial development of an idea or machine or interested in further reading, references are provided at the end of each history. For this chapter, see Section L.2, The Early Development of Computers, for a discussion on the early development of digital computers and performance measurement methodologies.

As you read the historical material, you'll soon come to realize that one of the important benefits of the youth of computing, compared to many other engineering fields, is that many of the pioneers are still alive—we can learn the history by simply asking them!

Case Studies and Exercises by Diana Franklin

Case Study 1: Chip Fabrication Cost

Concepts illustrated by this case study

- Fabrication Cost
- Fabrication Yield
- Defect Tolerance through Redundancy

There are many factors involved in the price of a computer chip. New, smaller technology gives a boost in performance and a drop in required chip area. In the smaller technology, one can either keep the small area or place more hardware on the chip in order to get more functionality. In this case study, we explore how different design decisions involving fabrication technology, area, and redundancy affect the cost of chips.

Chip	Die size (mm^2)	Estimated defect rate (per cm^2)	Manufacturing size (nm)	Transistors (millions)
IBM Power5	389	.30	130	276
Sun Niagara	380	.75	90	279
AMD Opteron	199	.75	90	233

Figure 1.22 Manufacturing cost factors for several modern processors.

1.1 [10/10] <1.6> Figure 1.22 gives the relevant chip statistics that influence the cost of several current chips. In the next few exercises, you will be exploring the effect of different possible design decisions for the IBM Power5.

a. [10] <1.6> What is the yield for the IBM Power5?

b. [10] <1.6> Why does the IBM Power5 have a lower defect rate than the Niagara and Opteron?

1.2 [20/20/20/20] <1.6> It costs $1 billion to build a new fabrication facility. You will be selling a range of chips from that factory, and you need to decide how much capacity to dedicate to each chip. Your Woods chip will be 150 mm^2 and will make a profit of $20 per defect-free chip. Your Markon chip will be 250 mm^2 and will make a profit of $25 per defect-free chip. Your fabrication facility will be identical to that for the Power5. Each wafer has a 300 mm diameter.

a. [20] <1.6> How much profit do you make on each wafer of Woods chip?

b. [20] <1.6> How much profit do you make on each wafer of Markon chip?

c. [20] <1.6> Which chip should you produce in this facility?

d. [20] <1.6> What is the profit on each new Power5 chip? If your demand is 50,000 Woods chips per month and 25,000 Markon chips per month, and your facility can fabricate 150 wafers a month, how many wafers should you make of each chip?

1.3 [20/20] <1.6> Your colleague at AMD suggests that, since the yield is so poor, you might make chips more cheaply if you placed an extra core on the die and only threw out chips on which both processors had failed. We will solve this exercise by viewing the yield as a probability of no defects occurring in a certain area given the defect rate. Calculate probabilities based on each Opteron core separately (this may not be entirely accurate, since the yield equation is based on empirical evidence rather than a mathematical calculation relating the probabilities of finding errors in different portions of the chip).

a. [20] <1.6> What is the probability that a defect will occur on no more than one of the two processor cores?

b. [20] <1.6> If the old chip cost $20 dollars per chip, what will the cost be of the new chip, taking into account the new area and yield?

Case Study 2: Power Consumption in Computer Systems

Concepts illustrated by this case study

- Amdahl's Law
- Redundancy
- MTTF
- Power Consumption

Power consumption in modern systems is dependent on a variety of factors, including the chip clock frequency, efficiency, disk drive speed, disk drive utilization, and DRAM. The following exercises explore the impact on power that different design decisions and use scenarios have.

1.4 [20/10/20] <1.5> Figure 1.23 presents the power consumption of several computer system components. In this exercise, we will explore how the hard drive affects power consumption for the system.

a. [20] <1.5> Assuming the maximum load for each component, and a power supply efficiency of 80%, what wattage must the server's power supply deliver to a system with an Intel Pentium 4 chip, 2 GB 240-pin Kingston DRAM, and one 7200 rpm hard drive?

b. [10] <1.5> How much power will the 7200 rpm disk drive consume if it is idle roughly 60% of the time?

c. [20] <1.5> Given that the time to read data off a 7200 rpm disk drive will be roughly 75% of a 5400 rpm disk, at what idle time of the 7200 rpm disk will the power consumption be equal, on average, for the two disks?

1.5 [10/10/20] <1.5> One critical factor in powering a server farm is cooling. If heat is not removed from the computer efficiently, the fans will blow hot air back onto the computer, not cold air. We will look at how different design decisions affect the necessary cooling, and thus the price, of a system. Use Figure 1.23 for your power calculations.

Component type	Product	Performance	Power
Processor	Sun Niagara 8-core	1.2 GHz	72–79 W peak
	Intel Pentium 4	2 GHz	48.9–66 W
DRAM	Kingston X64C3AD2 1 GB	184-pin	3.7 W
	Kingston D2N3 1 GB	240-pin	2.3 W
Hard drive	DiamondMax 16	5400 rpm	7.0 W read/seek, 2.9 W idle
	DiamondMax 9	7200 rpm	7.9 W read/seek, 4.0 W idle

Figure 1.23 Power consumption of several computer components.

a. [10] <1.5> A cooling door for a rack costs $4000 and dissipates 14 KW (into the room; additional cost is required to get it out of the room). How many servers with an Intel Pentium 4 processor, 1 GB 240-pin DRAM, and a single 7200 rpm hard drive can you cool with one cooling door?

b. [10] <1.5> You are considering providing fault tolerance for your hard drive. RAID 1 doubles the number of disks (see Chapter 6). Now how many systems can you place on a single rack with a single cooler?

c. [20] <1.5> Typical server farms can dissipate a maximum of 200 W per square foot. Given that a server rack requires 11 square feet (including front and back clearance), how many servers from part (a) can be placed on a single rack, and how many cooling doors are required?

1.6 [Discussion] <1.8> Figure 1.24 gives a comparison of power and performance for several benchmarks comparing two servers: Sun Fire T2000 (which uses Niagara) and IBM x346 (using Intel Xeon processors). This information was reported on a Sun Web site. There are two pieces of information reported: power and speed on two benchmarks. For the results shown, the Sun Fire T2000 is clearly superior. What other factors might be important and thus cause someone to choose the IBM x346 if it were superior in those areas?

1.7 [20/20/20/20] <1.6, 1.9> Your company's internal studies show that a single-core system is sufficient for the demand on your processing power; however, you are exploring whether you could save power by using two cores.

a. [20] <1.9> Assume your application is 80% parallelizable. By how much could you decrease the frequency and get the same performance?

b. [20] <1.6> Assume that the voltage may be decreased linearly with the frequency. Using the equation in Section 1.5, how much dynamic power would the dual-core system require as compared to the single-core system?

c. [20] <1.6, 1.9> Now assume the voltage may not decrease below 25% of the original voltage. This voltage is referred to as the *voltage floor*, and any voltage lower than that will lose the state. What percent of parallelization gives you a voltage at the voltage floor?

d. [20] <1.6, 1.9> Using the equation in Section 1.5, how much dynamic power would the dual-core system require as compared to the single-core system when taking into account the voltage floor?

	Sun Fire T2000	**IBM x346**
Power (watts)	298	438
SPECjbb (operations/sec)	63,378	39,985
Power (watts)	330	438
SPECWeb (composite)	14,001	4348

Figure 1.24 Sun power/performance comparison as selectively reported by Sun.

Exercises

1.8 [10/15/15/10/10] <1.4, 1.5> One challenge for architects is that the design created today will require several years of implementation, verification, and testing before appearing on the market. This means that the architect must project what the technology will be like several years in advance. Sometimes, this is difficult to do.

a. [10] <1.4> According to the trend in device scaling observed by Moore's law, the number of transistors on a chip in 2015 should be how many times the number in 2005?

b. [15] <1.5> The increase in clock rates once mirrored this trend. Had clock rates continued to climb at the same rate as in the 1990s, approximately how fast would clock rates be in 2015?

c. [15] <1.5> At the current rate of increase, what are the clock rates now projected to be in 2015?

d. [10] <1.4> What has limited the rate of growth of the clock rate, and what are architects doing with the extra transistors now to increase performance?

e. [10] <1.4> The rate of growth for DRAM capacity has also slowed down. For 20 years, DRAM capacity improved by 60% each year. That rate dropped to 40% each year and now improvement is 25 to 40% per year. If this trend continues, what will be the approximate rate of growth for DRAM capacity by 2020?

1.9 [10/10] <1.5> You are designing a system for a real-time application in which specific deadlines must be met. Finishing the computation faster gains nothing. You find that your system can execute the necessary code, in the worst case, twice as fast as necessary.

a. [10] <1.5> How much energy do you save if you execute at the current speed and turn off the system when the computation is complete?

b. [10] <1.5> How much energy do you save if you set the voltage and frequency to be half as much?

1.10 [10/10/20/20] <1.5> Server farms such as Google and Yahoo! provide enough compute capacity for the highest request rate of the day. Imagine that most of the time these servers operate at only 60% capacity. Assume further that the power does not scale linearly with the load; that is, when the servers are operating at 60% capacity, they consume 90% of maximum power. The servers could be turned off, but they would take too long to restart in response to more load. A new system has been proposed that allows for a quick restart but requires 20% of the maximum power while in this "barely alive" state.

a. [10] <1.5> How much power savings would be achieved by turning off 60% of the servers?

b. [10] <1.5> How much power savings would be achieved by placing 60% of the servers in the "barely alive" state?

c. [20] <1.5> How much power savings would be achieved by reducing the voltage by 20% and frequency by 40%?

d. [20] <1.5> How much power savings would be achieved by placing 30% of the servers in the "barely alive" state and 30% off?

1.11 [10/10/20] <1.7> Availability is the most important consideration for designing servers, followed closely by scalability and throughput.

a. [10] <1.7> We have a single processor with a failures in time (FIT) of 100. What is the mean time to failure (MTTF) for this system?

b. [10] <1.7> If it takes 1 day to get the system running again, what is the availability of the system?

c. [20] <1.7> Imagine that the government, to cut costs, is going to build a supercomputer out of inexpensive computers rather than expensive, reliable computers. What is the MTTF for a system with 1000 processors? Assume that if one fails, they all fail.

1.12 [20/20/20] <1.1, 1.2, 1.7> In a server farm such as that used by Amazon or eBay, a single failure does not cause the entire system to crash. Instead, it will reduce the number of requests that can be satisfied at any one time.

a. [20] <1.7> If a company has 10,000 computers, each with a MTTF of 35 days, and it experiences catastrophic failure only if 1/3 of the computers fail, what is the MTTF for the system?

b. [20] <1.1, 1.7> If it costs an extra $1000, per computer, to double the MTTF, would this be a good business decision? Show your work.

c. [20] <1.2> Figure 1.3 shows, on average, the cost of downtimes, assuming that the cost is equal at all times of the year. For retailers, however, the Christmas season is the most profitable (and therefore the most costly time to lose sales). If a catalog sales center has twice as much traffic in the fourth quarter as every other quarter, what is the average cost of downtime per hour during the fourth quarter and the rest of the year?

1.13 [10/20/20] <1.9> Your company is trying to choose between purchasing the Opteron or Itanium 2. You have analyzed your company's applications, and 60% of the time it will be running applications similar to wupwise, 20% of the time applications similar to ammp, and 20% of the time applications similar to apsi.

a. [10] If you were choosing just based on overall SPEC performance, which would you choose and why?

b. [20] What is the weighted average of execution time ratios for this mix of applications for the Opteron and Itanium 2?

c. [20] What is the speedup of the Opteron over the Itanium 2?

1.14 [20/10/10/10/15] <1.9> In this exercise, assume that we are considering enhancing a machine by adding vector hardware to it. When a computation is run in vector mode on the vector hardware, it is 10 times faster than the normal mode of execution. We call the percentage of time that could be spent using vector mode the *percentage of vectorization.* Vectors are discussed in Chapter 4, but you don't need to know anything about how they work to answer this question!

a. [20] <1.9> Draw a graph that plots the speedup as a percentage of the computation performed in vector mode. Label the *y*-axis "Net speedup" and label the *x*-axis "Percent vectorization."

b. [10] <1.9> What percentage of vectorization is needed to achieve a speedup of 2?

c. [10] <1.9> What percentage of the computation run time is spent in vector mode if a speedup of 2 is achieved?

d. [10] <1.9> What percentage of vectorization is needed to achieve one-half the maximum speedup attainable from using vector mode?

e. [15] <1.9> Suppose you have measured the percentage of vectorization of the program to be 70%. The hardware design group estimates it can speed up the vector hardware even more with significant additional investment. You wonder whether the compiler crew could increase the percentage of vectorization, instead. What percentage of vectorization would the compiler team need to achieve in order to equal an addition 2× speedup in the vector unit (beyond the initial 10×)?

1.15 [15/10] <1.9> Assume that we make an enhancement to a computer that improves some mode of execution by a factor of 10. Enhanced mode is used 50% of the time, measured as a percentage of the execution time *when the enhanced mode is in use*. Recall that Amdahl's law depends on the fraction of the original, *unenhanced* execution time that could make use of enhanced mode. Thus, we cannot directly use this 50% measurement to compute speedup with Amdahl's law.

a. [15] <1.9> What is the speedup we have obtained from fast mode?

b. [10] <1.9> What percentage of the original execution time has been converted to fast mode?

1.16 [20/20/15] <1.9> When making changes to optimize part of a processor, it is often the case that speeding up one type of instruction comes at the cost of slowing down something else. For example, if we put in a complicated fast floating-point unit, that takes space, and something might have to be moved farther away from the middle to accommodate it, adding an extra cycle in delay to reach that unit. The basic Amdahl's law equation does not take into account this trade-off.

a. [20] <1.9> If the new fast floating-point unit speeds up floating-point operations by, on average, 2×, and floating-point operations take 20% of the original program's execution time, what is the overall speedup (ignoring the penalty to any other instructions)?

b. [20] <1.9> Now assume that speeding up the floating-point unit slowed down data cache accesses, resulting in a 1.5× slowdown (or 2/3 speedup). Data cache accesses consume 10% of the execution time. What is the overall speedup now?

c. [15] <1.9> After implementing the new floating-point operations, what percentage of execution time is spent on floating-point operations? What percentage is spent on data cache accesses?

1.17 [10/10/20/20] <1.10> Your company has just bought a new Intel Core i5 dual-core processor, and you have been tasked with optimizing your software for this processor. You will run two applications on this dual core, but the resource requirements are not equal. The first application requires 80% of the resources, and the other only 20% of the resources. Assume that when you parallelize a portion of the program, the speedup for that portion is 2.

a. [10] <1.10> Given that 40% of the first application is parallelizable, how much speedup would you achieve with that application if run in isolation?

b. [10] <1.10> Given that 99% of the second application is parallelizable, how much speedup would this application observe if run in isolation?

c. [20] <1.10> Given that 40% of the first application is parallelizable, how much *overall system speedup* would you observe if you parallelized it?

d. [20] <1.10> Given that 99% of the second application is parallelizable, how much overall system speedup would you observe if you parallelized it?

1.18 [10/20/20/20/25] <1.10> When parallelizing an application, the ideal speedup is speeding up by the number of processors. This is limited by two things: percentage of the application that can be parallelized and the cost of communication. Amdahl's law takes into account the former but not the latter.

a. [10] <1.10> What is the speedup with N processors if 80% of the application is parallelizable, ignoring the cost of communication?

b. [20] <1.10> What is the speedup with 8 processors if, for every processor added, the communication overhead is 0.5% of the original execution time.

c. [20] <1.10> What is the speedup with 8 processors if, for every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

d. [20] <1.10> What is the speedup with N processors if, for every time the number of processors is doubled, the communication overhead is increased by 0.5% of the original execution time?

e. [25] <1.10> Write the general equation that solves this question: What is the number of processors with the highest speedup in an application in which $P\%$ of the original execution time is parallelizable, and, for every time the number of processors is doubled, the communication is increased by 0.5% of the original execution time?