Megan Leonard

6/13/2021

COMP 201 Homework 4

Answer the following questions based on your reading of the textbook, the module study notes, the videos, and the instructor’s presentation this week.

1. (Englander, exercise 8.3)
2. Consider a CPU with two parallel integer execution units. An addition instruction requires 2 clock pulses to complete execution, and a multiplication requires 15 clock pulses. Now assume the following situation: the program is to multiply two numbers, located in registers R2 and R4, and store the results in R5. The following instruction adds the number in R5 to the number in R2 and stores the result in R5. The CPU does not stall for data dependencies, and both instructions have access to an execution unit simultaneously. The initial values of R2, R4, and R5 are 3, 8, and 0, respectively. What is the result? Now assume that the CPU does handle data dependencies correctly. What is the result? If we define wasted time as time in which an execution unit is not busy, how much time is wasted in this example?

The outcome without the stalls for data dependencies would be 11 placed into R5 as the add would run before the multiply. If the CPU does handle data dependencies correctly then it would have the outcome of 24 from the multiply step. Based off the clock pulses we can say that the time wasted would be 14 clock pulses as it takes 15 for it to be multiplied.

1. Now assume that a later instruction in the fetch pipeline has no data dependencies. It adds the value in R1, initially 4, to the value in R4 and stores the result in R5. Data dependencies are handled correctly. There are no rename registers, and the CPU retires instructions in order. What happens? If the CPU provides rename registers, what happens? What effect does out-of-order execution have upon the time required to execute this program?

Following the speed that the instructions run, it would store 12 into R5 prior to the multiplication. The instructions are retired so that would lead to 12 being the final value in R5 since it would retire the store step for this location. If the CPU does provide rename registers, then it would be able to save the multiplied instruction into the R5 giving the answer 24 as we received above. The out-of-order execution leads to trouble with the time as it can lead to instructions being run before needed which would make the time seem less than it needed, it would lead to not being able to finish the needed parts.

1. (Englander, exercise 8.5) Some systems use a branch prediction method known as static branch prediction, so called because the prediction is made on the basis of the instruction, without regard to history. One possible scenario would have the system predict that all conditional backward branches are taken and all forward conditional branches are not taken. Recall your experience with programming in the Little Man Computer language. Would this algorithm be effective? Why or why not? What aspects of normal programming, in any programming language, support your conclusion?

This could be useful when a person is creating conditional statements. The programmer sets up a list of conditions like an if/else statement and the system predicts that the conditions would be met. If the person knows that the input data is supposed to pass the conditions this can help save time and even problem, check to make sure it runs as it is supposed to. The if statement is used in most languages and would support this conclusion as it uses the condition set to branch the program.

1. (Englander, exercise 8.12) Describe the trade-offs between the memory cache write-through and write-back techniques.

The memory cache write-through creates two copies and will update when the cache lines are altered. This makes it easier to avoid the loss of data. The memory cache write-back is faster then the write-through but could lead to the loss of data.

1. (Englander, exercise 9.1) Why would DMA be useless if the computer did not have interrupt capability?

The DMA runs in four steps which uses the interrupt technique to pause to show that it finishes with a part and can move to the next. For computers that do not have this capability, they would run until they run out of lines to run.

1. (Englander, exercise 9.4) Consider the interrupt that occurs at the completion of a disk transfer.
2. "Who" is interrupting "whom"?

The disk is transferred to the memory who will then interrupt the CPU.

1. Why is the interrupt used in this case? What would be necessary if there were no interrupt capability on this computer?

The interrupt is used to say that the block is finished and if there needs to be any other things done by the CPU involving this, then it can do them before the next block runs. If there are no interrupt capabilities, then they would need to have any necessary changes happen after and possible loops depending on the CPU.

1. Describe the steps that take place after the interrupt occurs.

After the interrupt, the CPU will be able to alter anything that it needs to then it will send a message saying that the interrupt is complete so the affected data can be run.

1. (Englander, exercise 9.6) What is an interrupt vector?

An interrupt vector is the location in the memory that handles the interrupts and when they should be run.

1. (Englander, exercise 9.10) The UNIX operating system differentiates between block-oriented and character-oriented devices. Give an example of each, explain the differences between them, and explain how the I/O process differs for each.

The block-oriented would be where the data is transferred to the memory in blocks of bytes like how a picture can be sent from a camera onto a storage drive. The character device will send a single character at a time like a keyboard which takes the input of one symbol at a time. The block system is set up for faster transfer of large files like how a picture is made up of a bunch of tiny pictures. The character device helps with smaller tasks like typing a word out using the keyboard and sending that to the storage.

1. In two to three paragraphs of prose (i.e. sentences, not bullet lists, and 350+ words) using APA style citations if needed, summarize, and interact with the content that was covered this week in class. In your summary, you should highlight the major topics, theories, practices, and knowledge that were covered. Your summary should also interact with the material through personal observations, reflections, and applications to the field of study. In particular, highlight what surprised, enlightened, or otherwise engaged you. Make sure to include at least one thing that you’re still confused about. In other words, you should think and write critically not just about what was presented but also what you have learned through the session. Feel free to ask questions in this as well since it will be returned to you with answers.

This week we went over chapters 8 and 9. Chapter 8 helps show the features of the CPU and the memory. The CPU is a very important part of the computer as it processes the data a person gives it. The CPU is something that I did not know many details about so seeing how it works like with the instructions it can execute with the information. For example, when we are typing up paragraphs on the keyboard, the CPU is being given that data so it can be transferred from the input of the keyboard to the indicated location that the words are needed. It is something that I do but never have thought deeply about.

The input and output of chapter 9 helps build on the thoughts that start from chapter 8. Input is something that I am used to as we always must have some form of input to do anything on the computer from the mouse to the keyboard. Interrupts are a new idea to me and do make sense in their use with the data transfer. It makes me think of how we can be typing on a site only for a block up ad to jump up and makes it so we essential “click off” of where we are typing. It stops our typing process until we interact with the ad then it will let up resume. I find it easier to relate ideas to familiar actions such as this to help me learn what topics entail.