Examination Feedback for EEE411/6031 – Advanced Computer Architecture
Autumn Semester 2006-07

Feedback for EEE411/6031 Session:2006-2007

Feedback: Please write simple statements about how well students addressed the exam paper in general and each individual question in particular including common problems/mistakes and areas of concern in the boxes provided below. Increase row height if necessary.

General Comments:

Overall, performance was a bit mixed. It is clear that people did not have recall of facts and this, in turn, hampered responses to simple questions of fact and, moreover, their ability to apply this knowledge!

Question 1:

This question was generally not well attempted. Even basic facts such as the way in which to construct a banyan network from s, f, and l were missing. For example, the fact that the Banyan is constructed from s x f crosspoint switches seemed to elude many people. Furthermore, having constructed a network, the connectivity was wrong in many cases. There is a simple acid test. If you use digit selectable routing, expressing an output port as an l+1 digit, base s number, and using each digit of the output port to select the port through which to exit from each crosspoint along a route from input to output, you should end up at the same output port *independent of where you started*!

Similarly, part b was done badly. The question asked you to make a choice based on basic facts and empirical data. The basic thing that people forgot to do was to determine the rate at which a processor would be able to access memory *in the absence of contention from any other processor*. This is calculated as access time to cache * hit rate + (1-hit rate)* transfer time. This shows that the 32 line length is best. This rate can then be used to estimate (to first order) the effect of contention, using the equation given and this also shows that the 32 line length has the highest probability of a memory access succeeding. Based on these two facts, the decision should be clear.

Question 2:

This question was generally quite well done. There were hidden depths, however. A simple analysis of the Reservation table gives a throughput of ¼ datum per clock. Careful examination, however, shows that there is contention in the 2nd multiplexer when this solution is adopted. In the event, however, nobody noticed this and I was prepared to accept both answers: the analysis from the Reservation table, or a more in depth treatment. Some people drew out the reservation table and then failed to notice that the next datum could be put in 4 clock cycles later.

Some people did not recognize (or remember) that simple retiming of the pipeline does not improve things when one or more of the resources is saturated!

When it came to the re-design of the pipeline I was pretty much prepared to accept any reasoned argument. Quite a lot of people noted that replicating B gave the best improvement. However, some people posited this and then passed most of the data through one copy rather than sharing the load between both of them. The simplest solution is to recognize that in A-B-C-B-B-C-B-D there is a repeated sequence B-C-B and merely interposing the new B between C and the following multiplexer (which becomes redundant) is the best solution.

Question 3:

Few people did this which is unfortunate because it was generally well done.

Question 4:

This question was very descriptive and I hoped that people would recall many of the facts from the course. However, people's performance was somewhat mixed. Many people just did not do part 5 concerning the vector processors and could not distinguish in parts 9 and 10 between a test-and-set instruction and read-lock/write-unlock sequence of instructions. Indeed, very few people spotted what part 9 was asking. Similarly, not many people remembered that shared data is uncacheable in static coherence (part 6).