Feedback for EEE411/6031 Session: 2011-2012

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:

This was a pretty straightforward paper and I expected that there would be few problems with it.

Question 1:

This question was reasonably well answered. Everybody managed to get the reservation table although working out the throughput and utilization foxed some people. Note, I wanted throughput overall i.e. how much data emerges from the pipeline per clock cycle – not data flowing through each element. Similarly, most people could prove the value of the collision vector (I only gave marks to those people who actually showed why – simply writing 01000 or 00010 = 01010 was not enough!). However, fewer people could draw the state table. Finally, part e) knocked-on from part d) (but was only worth two marks) and only a few people could give a reasoned answer.

Question 2:

This question was also reasonably well answered in the main although whilst I got a lot of diagrams of buffer based CPS, I tended not to get too many multiplexer based CPS. Some people drew Delta and Banyan networks instead! Most people could work out the values of f,s, and l and draw the network although a few people produced networks with 27 inputs and 8 outputs and a few people made a complete hash of the network.

Question 3:

This question was answered well – in parts. The major problem was the calculation of access time. Many people got mixed up with the probabilities and read or write and some people did not bother with writing the blocks back. To recap if 0.7 are reads and 0.3 are writes then:
tacc = 0.7 x tread + 0.3 x twrite
If read hit is 0.96 and cache read is 1 cycle then tread = 0.96 x tclk + 0.04 * tcachemiss
if read hit is 0.96 and cache write is 2 cycles then twrite = 0.98 x 2 x tclk + 0.02 * tcachemiss
tcachemiss = 32ns (reading a block) + 0.4 x 32ns (writing back only the dirty blocks)

Question 4:

This question was attempted by fewer candidates than the other questions. However, apart from a few cases, most people made a relatively good attempt. The difficult part was, clearly, the program to move data – which was, in truth, relatively straightforward. Most people understood the need to synchronise between the CTB and the array. However, the ordering of data 7…0 rather than 0…7 was wrong in most cases as was, more subtly, the ordering of INW and BCAST instructions. People invariable put them in this order – which is wrong.