Advanced Computer Architecture (EEE411/6031) January 2009 - Exam Feedback

General

A number of people did not write question numbers on the front of their exam scripts. Whilst we try our best to avoid missing sections of questions (and we have procedures in place to help us in this), if you do not put the question numbers on the front of the script it can only increase the chance that we will miss a question.

Question 1

Most people did this question and it was generally well answered. However, some people did not seem to have looked at the number of marks allocated for each section and gave either extremely verbose or terse answers, which wasted time or lost marks respectively. In the section of the reorder buffer, some people confused committing and completing and a few people seemed to think that instructions were only executed when they reached the head of the queue. This is not the case: instructions on the queue are executed as soon as their operands become ready and results are written back to the reorder buffer (completing). An instruction at the head of the queue will be committed (i.e. removed from the queue and the result of the instruction written back to the register) only if it has completed (it can be completed before or after it has arrived at the head of the queue but will only be committed when both conditions are met).

Question 2

Most people did this question and it was surprisingly well answered – despite the fact that nobody got the answer that I was *really* looking for. When drawing out the reservation table, people missed opportunities to launch data into the pipeline, preferring a more relaxed schedule that was easier to deal with. Consequently, they got marks but not as many as they might have done if they had developed the table far enough (circa 25 timeslots) to see what the established pattern of data entry should be (at which point all the processors would be used 100%). I gave credit where I could for throughput and utilization. The second part, asking about correcting the data order was answered relatively well. The simplest solution is to use feed-forward registers at the output to correct the ordering. Some people opted for delay registers feeding back into the set of processors but the risk in this is that it affects other operands in flight. Again, credit was given where it was due.

Question 3

Again, most people did this question. It was well answered on average. However, with reference to part a): if the question says *draw a schematic* then if you do not you will inevitably lose marks. In part b), not many people drew the state transition diagram for the dynamic coherence scheme – this was helpful to the answer. The worst answered part was d) (although most people made a reasonable attempt). The proportion of writes only really makes a difference at the top level: the question was constructed to say the memory used at the top level had an access time of one clock

cycle. Hence, reading takes one clock cycle whilst writing, inevitably, takes two clock cycles. Between the L1 and L2 cache, you can only account the time for reading (writing effectively takes no time because any write to the L1 cache is transferred through to level 2). Furthermore, you cannot use the proportion of writes to deal with the interface between L2 cache and main memory because a write back can be caused on a memory read causing a miss at the L2 cache – it is independent of writes. All you can say is that an additional 10% of the time is incurred over and above transferring blocks from main memory to the L2 cache. Some people also ignored the probabilities – they are crucial to the operation of a memory hierarchy. Finally, I used a simple model for the time taken to transfer a block between levels. Some people augmented the times by assuming that the TLB lookups occurred serially in different levels – in practice, this is not the case: lookups run concurrently to save time when there are misses on upper levels. I was quite generous, as long as it was clear what people's assumptions were and if they were reasonable.

Question 4

Very few people did this question. It was slightly unfamiliar and this might have put people off. However, those who did the past paper may have noticed that the solution to the first part was well-described in the solutions to last year's exam paper (as additional information). The second part, whereby the expression for the communication time was developed, was woefully done. The expression for p_A had been carefully approximated – in the question – to make the analysis tractable: there was no need to prove that this approximation is valid: it is (it is a Taylor expansion taken to 2^{nd} order). Following this, equating the expression for p_A with the expression from the first part of the question (given in the text) allows the final expression to be developed.