Data Provided: None

The University Of Sheffield.

DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING

Autumn Semester 2008-2009 (2 hours)

Advanced Computer Architectures 4

Answer **THREE** questions. **No marks will be awarded for solutions to a fourth question.** Solutions will be considered in the order that they are presented in the answer book. Trial answers will be ignored if they are clearly crossed out. **The numbers given after each section of a question indicate the relative weighting of that section.**

1. a. A pipelined processor will suffer from problems arising from data dependencies. Describe:

i) how these problems arise; **(3)**

ii) the methods (excluding reorder buffers) used to overcome these problems (only a brief description is required). **(4)**

b. In some cases, a *reorder* buffer is used to solve problems caused by data/control dependencies in such processors.

i) Describe how such buffers are used and, in particular, identify the difference between instructions *completing* and *committing*. **(5)**

ii) For the following code snippet, show how a reorder buffer might be used:

```
     LOAD   @addr,R1      ; R1 ← addr
     CMP    R1,3          ; is R1 equal to 3
     JEQ    L1
     ADD    R2,R3         ; R3 ← R2 + R3
L1   SUB    R4,R5         ; R5 ← R5 + R4
     …
```

(6)

c. What is meant by *precise interrupts* and a *precise architectural state* and what is their relevance to reorder buffers? **(2)**

2. A processing pipeline consists of four processing blocks, A…D, connected as shown in **Figure 2**.

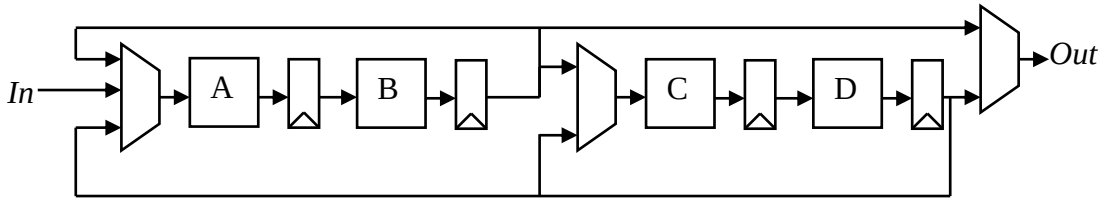

Figure 2: Processing Pipeline

A sequence of 4 datum v_i, w_i, x_i, y_i are to be processed to produce outputs V_i, W_i, X_i, Y_i. The sequence is then repeated: i.e. the input sequence is:

$v_0, w_0, x_0, y_0, v_1, w_1, x_1, y_1, v_2, w_2, x_2, y_2, v_3, w_3, x_3, y_3, \ldots$

Each datum is processed in a different way:

$v_i \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow V_i$

$w_i \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow C \rightarrow D \rightarrow W_i$

$x_i \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow X_i$

$y_i \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow A \rightarrow B \rightarrow Y_i$

a. **i)** Draw out the reservation table for the sequence of activity. **(8)**

ii) Calculate the throughput of the pipeline. **(2)**

iii) Calculate the utilisation of the processing blocks. **(2)**

b. You notice that the data comes out in a different order to the order in which it entered the pipeline. Show how the pipeline might be altered, simply, to restore the data to its original order. **(8)**

3. **a.** Draw a schematic diagram of a 4-way set associative cache system, identifying how it operates. **(4)**

b. **i)** Caches will be used in multiprocessor systems, but they present a particular problem that relates to data coherency. Describe the approach taken if Dynamic Coherence is used. **(4)**

ii) In particular, a typical multiprocessing desktop system might employ a shared *mezzanine* bus to connect the processors to the rest of the system. What particular functionality should such a bus possess to enable data coherency between multiple caches? **(2)**

d. A 2-level cache memory system is as shown in **Figure 3**. The two, set-associative caches are controlled synchronously using a 3GHz clock.

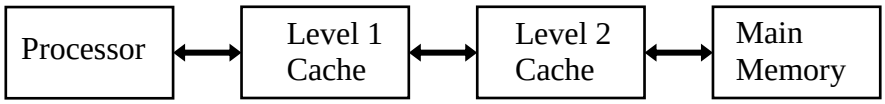

Figure 3: Memory Organisation

The *basic* access time for the memory used to construct the Level 1 Cache is one clock cycle, the time taken to transfer a line between the Level 2 Cache and the Level 1 cache is 10 clock cycles, whilst the time taken to transfer a line between Main Memory and the Level 2 Cache is 64 clock cycles.

The probability of a memory cycle being a memory read is 0.75, the hit rate of the Level 1 Cache is 0.7 and it is a *write through* cache. The hit rate of the Level 2 Cache is 0.85 and it is a *write back* cache (it has been estimated that 10% of the lines in the Level 2 Cache are *dirty*). All the memory accessed by the processor fits into the Main Memory.

Estimate the *effective*, average time for a memory access, stating any assumptions that you make. **(10)**

4. A network of *m* processors is arranged as shown in **Figure 4**. The common buses linking all of the processors together will support four communications in parallel.

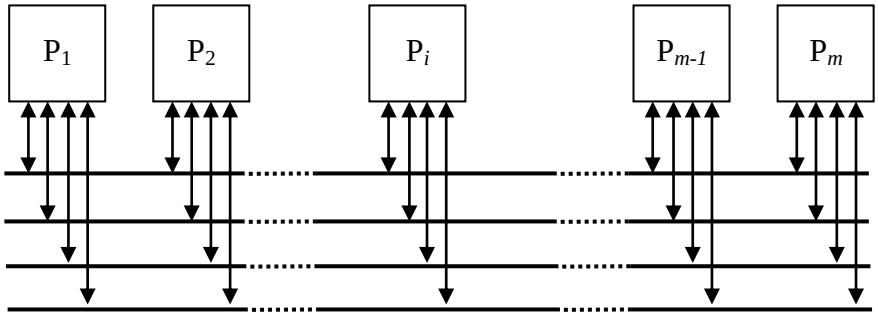

Figure 4: Network of Processors

n tasks are to be partitioned across *m* processors. During execution, each task sends exactly one message to each other task. The time taken to send a message between two tasks is t_{ext} if the two tasks reside on different processors (assuming that there no contention on the external buses).

a. You recognise that the 4 buses behave like an *m* input, 4 output *cross point switch* (assuming that each processor distributes its messages equally between the 4 buses) and that the probability that the message transfers will be accepted is, therefore:

$$p_A = \frac{4}{mRt_{ext}}\left(1-\left(1-\frac{Rt_{ext}}{4}\right)^m\right) \approx 1-\frac{Rt_{ext}(m-1)}{4} \quad \text{(for modest values of } Rt_{ext}\text{)}$$

where *R* is the rate, on average, at which messages are transmitted by each processor.

Show that the effective time taken to communicate a message between a pair of tasks on separate processors, across the buses – taking contention into account – is t_{ext}/p_A. **(10)**

Show that the *communication time* as a consequence of the organisation is:

$$t_{com} \approx \frac{n}{4}\cdot\frac{m-1}{m}\cdot(4n+m-1)\cdot t_{ext}$$

Stating any assumptions that you made to arrive at this answer.

(Hint: you will need to work out what the rate at which messages are transmitted by each processor is) **(10)**

NLS / NA