



Coláiste na Tríonóide, Baile Átha Cliath
Trinity College Dublin

Ollscoil Átha Cliath | The University of Dublin

Faculty of Engineering, Mathematics and Science

School of Computer Science and Statistics

Integrated Computer Science
Year 3 Annual Examinations

Hilary Term 2017

CS3021 Computer Architecture II

9 January 2017

Goldsmith Hall

09.30 – 11.30

Dr Jeremy Jones

Instructions to Candidates:

Answer **THREE** questions.

All questions carry equal marks

All questions are marked out of 20

- Q1. What is a pipelined processor? What are the benefits of pipelining? Explain the organization and operation of the DLX five stage execution pipeline.

[4 marks]

What are data hazards? Describe two techniques to prevent stalls which can be used to overcome data hazards in the DLX pipeline. Show, using diagrams, how the data hazards in the following code sequence are overcome by the DLX CPU.

1. $r1 \leftarrow r2 + r3$
2. $r4 \leftarrow r1 + r2$
3. $r10 \leftarrow r1 + r4$
4. $r11 \leftarrow r4 + r1$

[10 marks]

What is a load hazard? For the two code segments below, explain why the load hazard arises in the 5 stage DLX pipeline and its effect in terms of stall cycles. Use diagrams to illustrate your answer. What steps, if any, can be taken to avoid such load hazards?

- (i) `ld r1, 0(r2) // r1 = [r2]`
 `add r1, r1, r1 // r1 = r1 + r1`
- (ii) `ld r1, 0(r2) // r1 = [r2]`
 `bnez r1, L // branch if r1 != 0 to L`

[6 marks]

- Q2. How are virtual addresses converted to physical addresses by a two level page table? Assume a page table structure as per an IA32 based CPU. Draw appropriate diagrams to illustrate your answer.

[4 marks]

What is the advantage of using a two level page table compared with a single level page table? Comment on the amount of physical memory needed for the page tables of (i) a “small” process and (ii) a “maximum sized” process using a single level and a two level page table. Draw appropriate diagrams to illustrate your answer.

[4 marks]

A user process is created which has 0x7000 bytes of code, 0x1678 bytes of initialised data, 0x2888 bytes of uninitialised data and 648 bytes of stack data copied from its parent. Draw a diagram that shows the structure and size of the 2-level page table which is initially created for the process. What size (in pages) is the initial memory foot print of the processes?

[5 marks]

If the following valid memory accesses are made by the process, draw a diagram which shows the changes that would be made to the initial page table structure and memory foot print as a result of handling any page faults (U = User address). Provide a brief explanation for each change.

U 0x00001000
 U 0x00001004
 U 0x00001008
 U 0x00002000
 U 0x00007000
 U 0x00009000
 U 0xFFFFE800
 U 0xFFFFD800

(Question 2 continues on next page)

U 0x00002000

U 0x00802000

U 0x00002000

U 0x00009020

[7 marks]

- Q3. With the aid of a diagram, explain how a cache organisation can be characterised by the three constants LKN. Explain, in detail, how a data item is accessed in an LKN cache. Why does a cache normally reduce the effective memory access time?

[6 marks]

Compute the number of hits and misses if the following list of hexadecimal addresses is applied to (1) a 64 byte direct mapped cache with 16 bytes per line and (2) a 64 byte fully associative cache with 16 bytes per line.

0x0000 → 0x0010 → 0x0020 → 0x0030 →
 0x003c → 0x0050 → 0x0010 → 0x000c →
 0x0010 → 0x0050 → 0x002c → 0x0010 →
 0x0050 → 0x0020 → 0x0010 → 0x0000

Assume that (i) the low 4 bits of the address is used as the offset into the cache line and the next $\log_2(N)$ bits select the set (ii) the cache is initially empty and (iii) a LRU replacement policy is used.

[10 marks]

Would you expect an associative cache of size N and line size L to always have a higher hit rate than direct mapped cache of size N and line size L. Explain the reasoning behind your answer with an example?

[4 marks]

- Q4. What is the cache coherency problem? Briefly explain the states and operation of the Firefly cache coherency protocol.

(8 marks)

Consider a three CPU multiprocessor system with a Firefly cache coherent bus. Each CPU has a first level cache which is direct mapped with 2 sets. Even addresses (a0, a2) map to set 0 and odd addresses (a1, a3) map to set 1. The caches initially contain a0 and a1 with data identical to memory. Explain in detail the bus traffic and cache line state transitions that occur when the following sequence of "memory" accesses are made.

- 1 CPU 0: read a0
- 2 CPU 0: read a2
- 3 CPU 0: write a2
- 4 CPU 0: write a2
- 5 CPU 1: read a2
- 6 CPU 2: read a2
- 7 CPU 0: write a2
- 8 CPU 0: write a2
- 9 CPU 0: read a0
- 10 CPU 1: read a0
- 11 CPU 2: write a2
- 12 CPU 2: write a2

(10 marks)

What difference would it make if the protocol didn't have a shared and dirty state (SD)?

(2 marks)