UNIVERSITY OF DUBLIN

TRINITY COLLEGE

Faculty of Engineering and Systems Sciences

Department of Computer Science

B.A (Mod.) Computer Science
JS Examination

Trinity Term 2001

3BA4 - Computer Architecture II

Tuesday 29th May MANSION HOUSE 09.30 - 12.30

Dr. J.O. Jones and Dr. Andrew Butterfield

Attempt **FIVE** questions at least two from each section
Use separate answer books for each section.

SECTION A

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

What effect would a "simple" implementation of branch instructions have on the DLX pipeline. What is delayed branching? Explain how branch prediction can be used to speed up the execution of branch instructions.

Compare the effective cycles per instruction (CPI) of (i) a "simple" branch implementation (ii) delayed branching and (iii) a branch prediction scheme. Assume 20% branch instructions, 70% probability that the branch changes the PC, 90% probability of hitting the branch target buffer, 90% probability of a correct prediction, a 1 cycle penalty if the branch target buffer needs updating and an 80% probability of filling the branch delay slot with a useful instruction.

Q2. How are virtual addresses converted to physical addresses by a two level memory management unit? Illustrate the advantage of using a 2-level page table structure by comparing the amount of physical memory needed for the page tables of a small & a maximum sized process with that of using a single level page table structure. Assume 4GB virtual & physical address spaces with a 10-10-12 address configuration like the Intel 486.

What is demand paged memory management? Given the 2–level page table structure above, show the organisation of the initial page table for a user process which has 0x1800 bytes of code, 0x2800bytes of initialised data, 0x2800 bytes of uninitialised data and 0x0800 bytes of stack data copied from its parent. How many pages of memory need be allocated to the process initially? Explain what happens when code is executed and data accessed in the different memory regions? How are illegal memory accesses detected?

Q3. What is a cache? How does a cache reduce the effective memory access time? Explain how a cache organisation can be characterised by the three constants LKN. Explain in detail how a data item is searched for in an LKN cache. What special names are given to cache organisations where (i) N=1 (ii) K=1 and (iii) K=4.

Would you expect a 2-way cache to <u>always</u> outperform a 1-way cache of equal size? Identify an address sequence which produces, for equally sized caches, more misses for a 1-way cache than a 2-way cache and then an address sequence which produces more misses for the 2-way cache than the 1-way cache (assume a LRU replacement policy). Explain the reasoning behind your address sequences.

Q4. What is the cache coherency problem? Under what conditions are the caches in a system considered to be coherent?

Explain (i) the meaning of the 4 cache line states used in the write-once cache coherency protocol and (ii) the basic operation of the protocol. Given a 3 CPU multiprocessor system, where each CPU has its own local cache, illustrate the bus traffic and cache state transitions that would occur if the following CPU memory requests were issued:

CPU 0: read a0
CPU 1: read a0
CPU 0: write a0
CPU 0: write a0
CPU 0: write a0
CPU 1: read a0

What advantage does the write-once protocol have over the simpler write-through scheme?

SECTION B

Q5 Given the following logic function: Y = NOT(A*B*(C+D))

(i) Design a CMOS switch circuit that implements this function. Your design should include a determination of the widths of the transistors involved (assuming a minimum width of 0.2μm, and a μn/μp ratio of **2.5**).

(ii) Express the CMOS layout topology of your circuit using a Stick Diagram, subject to the following constraints: Power and Ground run horizontally across the top and bottom respectively, of the circuit, *inputs A and B enter on the right, while inputs C and D enter on the bottom*, on Polysilicon, and the *output emerges on the left*, also on Polysilicon.

Q6 (i) A microprocessor design requires an arithmetic logic unit (ALU) to drive a signal to a very long internal bus line, feeding other circuits around the chip. The speed performance of this signal is important. Explain how you might re-size transistors, and add extra if necessary, in order to maximise speed of this bus line.

(ii) A control wire in the microprocessor feeds a large number of inputs, all close together, forming a large capacitance load. The control circuit generating the control signal has an output driver that is too small to drive the control signals at the desired speed. Explain why making that driver with larger transistors does not solve the problem, and show how to add in extra inverters to give a solution.

Q7 (i) Explain the term "pseudo-nMOS", show how it is used to reduce the transistor count in CMOS logic gates, and explain its disadvantages.

(ii) Explain how the existence of two clock signals, one active low, the other active high, with non-overlapping active periods, can be used to produce "dynamic CMOS" logic. Illustrate you answer by showing how the logic function Y=NOT(A+B*C) would be implemented as a dynamic gate.

(iii) Explain where a dynamic gate stores information, and hence explain why processors implemented with dynamic logic have a minimum clocking speed.

Q8 (i) Show how dynamic CMOS can be used to produce regular programmable logic arrays (PLA).

(ii) Sketch out a PLA to implement the following logic:

X = /A+/B

Y= /A(B+/C)

Z = /AB+A/C+A

where A,B and C are inputs and X,Y and Z are outputs. Your answer should show how the PLA building blocks are put together and the effect of the programming.