UNIVERSITY OF DUBLIN

TRINITY COLLEGE

Faculty of Engineering and Systems Sciences
Department of Computer Science

B.A.Computer Science
Junior Sophister Examination

Trinity Term 2003

3BA5 - Computer Engineering

Saturday 24th May GMB 09.30-12.30

Mr. J. Dukes, Dr. Andy Nisbet

Attempt FIVE questions

1. a) Draw a block diagram of an AND array, Wallace tree of Carry-Save-Adders and a final Carry-Lookahead-Adder which will produce the product, Z, of two numbers X and Y, where both X and Y are 8 bits wide.

(7 marks)

b) Describe Booth's technique for halving the number of partial products that need to be added to produce the product of a multiplier X and a multiplicand Y.

(7 marks)

c) Show an example of shift-and-add multiplication of two six-bit numbers using Booth's technique.

(6 marks)

2. a) Draw the topology and state the degree and diameter of both of the following interconnection networks:

   i) a binary tree with 15 nodes
   ii) a hypercube with 16 nodes

   (8 marks)

   b) Define the functions **shuffle(*i*)** and **exchange(*i*)** used to produce the links in a shuffle-exchange network. Draw a shuffle exchange network with 8 nodes.

   (4 marks)

   c) Describe an algorithm to efficiently determine the sum of n values, $a_0 \ldots a_{n-1}$, using a shuffle-exchange interconnect with n nodes. Assume that each node i initially contains the value a_i in a register and that $n = 2^k$. Demonstrate the algorithm for a network with 8 nodes, showing clearly the contents of the registers after each step.

   (8 marks)

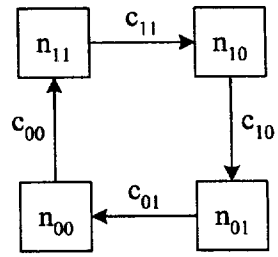
3. a) Describe <u>separate</u> algorithms for routing a message M from a source node S to a destination node D for the two network topologies listed below. In each case, describe the addressing of the nodes, the labelling of the ports on each node, the routing information added to M and the algorithm used to determine which output port is used to forward the message to the next intermediate node.

   i) a 2-dimensional mesh with $n = k_0 \times k_1$ nodes
   ii) a 3-dimensional cube with 8 nodes

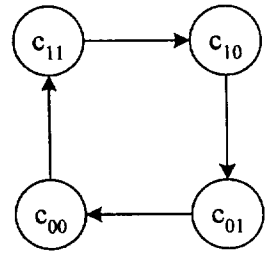
   (12 marks)

   b) When using wormhole routing, the output channel for a leading flit is dependant on the input channel and destination node. The figure below illustrates this dependency for a simple unidirectional ring with four nodes. Explain why deadlock may occur and describe a technique for avoiding the deadlock.

   (8 marks)



Network



Dependency Graph

4. a) List three types of data hazard that can exist between instructions in an instruction pipeline and give an example of each one.

(6 marks)

b) A static pipeline with three stages has the following reservation table:

	Time					
	t_0	t_1	t_2	t_3	t_4	t_5
Stage 1	X					X
Stage 2		X		X		
Stage 3			X		X	

Derive the forbidden list and collision vector for the pipeline and draw the associated state diagram. What is the minimum average latency (MAL) for the pipeline? Redraw the reservation table showing the initiations from the cycle that yields the MAL. Comment on the implications of implementing the cycle that yields the MAL. Is there a more suitable cycle?

(14 marks)

5. a) Explain the motivation for "normalising" floating point numbers. What is the "hidden-bit"? Give an example of normalisation.

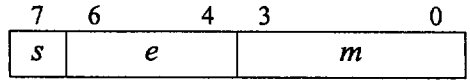
(3 marks)

b) The addition of two floating point numbers may be performed using a pipeline of three or more stages. Suggest a design for such a pipeline, describing in high-level terms the operation performed by each pipeline stage.

(7 marks)

c) A pair of floating point numbers, X and Y, are stored in the following 8-bit format:

$(-1)^s \times 2^e \times 1.m$ =



If $X = 00110110$ and $Y = 00101010$, calculate the intermediate result of each stage of the pipeline you described in part (b).

(4 marks)

d) Draft a partial circuit schematic showing how you would modify the mantissa in the normalisation stage of your pipeline in part (b).

(6 marks)

6. a) Describe what is meant by loop permutation, loop tiling and padding transformations using words.

(4 marks)

b) Illustrate the effect of the listed transformations on the 2nd loop nest in the matrix multiplication benchmark listed below: (i) loop permutation only, (ii) loop tiling only, (iii) padding only. Your code should include any additional variable declarations, and, or modifications to current variable declarations.

(7 marks)

```
#include <stdio.h>
#define N 4096
int A[N][N], B[N][N], C[N][N];

    int main() {
    int i,j,k;
    for(i = 0; i < N;i++) { // initialise the data
        for(j = 0 ; j < N;j++) {
            A[i][j] =  j;
            B[i][j] = j;
            C[i][j] = 0;
        }
    }

    // illustrate transformations here!
    for(i=0; i < N;i++) {
        for(j = 0; j < N;j++) {
            for(k = 0; k< N;k++){
                C[i][j] = C[i][j] + A[i][k]*B[k][j];
            }
        }
    }
    return 0;
}
```

c) Discuss how the appropriate application of source code transformations can improve the performance of an application with reference to architectural features typically found on a general purpose processor. Describe the principle type of performance overhead that each transformation can reduce. Describe the main feature of an application that does not require such transformations to achieve good performance.

(7 marks)

d) Describe when it is legal to apply loop permutation, loop tiling and padding transformations.

(2 marks)

7. a) Define what is meant by "latency" and "throughput" in the context of a clocked synchronous digital system.

(2 marks)

b) Describe pipelining and how it can influence clock period, latency and throughput.

(5 marks)

c) Illustrate your description by pipelining the following HandelC code and defining its latency and throughput.

(4 marks)

```
int 4 a,b,c,d,e,f,g,h,i;
// chanin declarations are slightly simplified for brevity
chanin int 4 chan1, chan2, chan3, chan4;
chanin int 4 chan5, chan6, chan7, chan8;
while(1)
    par{
        chan1 ? b;
        chan2 ? c;
        chan3 ? d;
        chan4 ? e;
        chan5 ? f;
        chan6 ? g;
        chan7 ? h;
        chan8 ? i;
        a = b+c+d+e+f+g+h+i;
    }
```

d) Describe the semantics of read and write accesses to a single registered variable in HandelC with reference to a clock cycle.

(2 marks)

e) List general rules for optimising the performance of a HandelC program.

(4 marks)

f) Discuss the architectural features of FPGAs enabling applications to achieve significant performance improvements over state of the art microprocessors.

(2 marks)

g) Describe the principal advantage FPGAs have in implementing cryptography applications over general purpose CPUs.

(1 mark)