UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1997

BEng Honours Degree in Computing Part I
MEng Honours Degrees in Computing Part I
BSc Honours Degree in Mathematics and Computer Science Part I
MSci Honours Degree in Mathematics and Computer Science Part I
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Associateship of the Royal College of Science
Associateship of the City and Guilds of London Institute

PAPER 1.4 / MC1.4

ARCHITECTURE I

Friday, April 25th 1997, 2.00 - 3.30

Answer THREE questions

For admin. only: paper contains 4 questions

1a Define a 48-bit instruction format **of your own** that can encode a 68000 MOVE instruction with a displacement mode source operand and an index mode destination operand. Ensure that your definition indicates the purpose and length of each instruction field as well as the binary value of each instruction field for the following MOVE instruction:

```
MOVE.L   6(A2), 8(A5, D3.L)
```

State any additional assumptions that your make.

b For each of the following 68000 instructions give a functionally equivalent 68000 instruction sequence. Use of the post-increment, pre-decrement, displacement or index addressing modes **is not permitted** in your instruction sequences.

```
i)   MOVE.W   -(A3),   (A4)+

ii)  MOVE.L   6(A2),   8(A5, D3.L)
```

Your instruction sequences must leave the 68000 address and data registers in an equivalent state to that which would have occurred if the corresponding single instruction was executed. Note: use of the system stack **is permitted** within your instruction sequences.

c Comment on the pros and cons of a hypothetical architecture that lacks the pre-decrement, post-increment, displacement and index modes.

The three parts carry, respectively, 40%, 40% and 20% of the marks.

2a Layout the elements of the following 2-dimensional integer array

```
X : array 0..2, 0..3 of int
```

in main memory and work out an expression for calculating the address of array element X(R,C) given the address *addr* of the first element of the array. Assume integers are 16-bit.

b Write a commented 68000 version of the following procedure:

```
procedure Transpose (var A : array 0..3, 0..2 of int,
                     var T : array 0..2, 0..3 of int)
var row, col : integer;
begin
     for row : 0 .. 3
          for col : 0 .. 2
               T(row, col) = A(col, row)
          end for
     end for
end Transpose
```

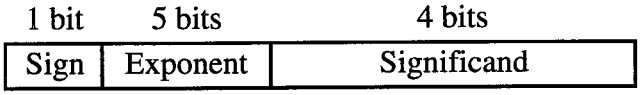
c Explain how memory-interleaving can speedup array operations.

The three parts carry, respectively, 20%, 60% and 20% of the marks.

Turn over ...

3a Suppose that the IEEE defines a new 10-bit floating point format called Tiny Precision that follows the same general rules as the IEEE Single Precision format except that the Exponent field is 5 bits and the Significand field is 4 bits.

Tiny Precision Format



For this format:

i) determine the number immediately after 1.5? Express your answer as a decimal number in the form $1.5 + 2^X$

ii) determine the number immediately after 3? Express your answer as a decimal number in the form $3 + 2^X$

iii) determine the number immediately after 6? Express your answer as a decimal number in the form $6 + 2^X$

iv) devise a formula for the determining the next number after any normalised number N. Express your answer in the form $N + 2^X$. Assume that N is less than the highest normalised number, i.e. that the next number will not be IEEE infinity.

b Using only normalised floating-point values and normalised results explain how

i) $x + y = x$ for some non-zero value of y

ii) the associative law $x + (y + z) = (x + y) + z$ does not hold for floating point addition.

Give an example in each case.

c Explain why the exponent field in the IEEE floating-point representation is placed to the left of the significand and why exponents are represented in Excess notation.

The three parts carry, respectively, 30%, 50% and 20% of the marks.

4a Consider a DMA I/O controller for a disk drive that accepts requests to write data starting at a particular track number and sector number. Describe the likely sequence of events from the time a program is about to request a block of data to write, until the time the program is notified that the block has been transferred to disk. Assume that the I/O controller is connected to a 68000 CPU and has an internal 1Kb buffer onboard.

b If the disk drive is capable of transferring 2,500,000 words per second to the memory system, calculate the *percentage* of the available memory cycles that its DMA controller might use if the memory system has an access time of 80 nanoseconds.

c In many processors an interrupt is only handled after the execution of the current instruction. Consider the possibility of a processor recognising and processing interrupts while an instruction is executing. Discuss the difficulties that would arise and also give a situation where such interrupt processing is desirable.

The three parts carry, respectively, 50%, 25% and 25% of the marks.

End of paper