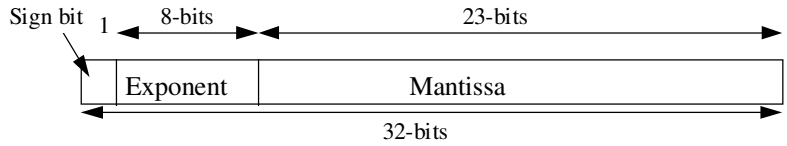
2.5 HOURS

ANSWER 3 QUESTIONS

Question 1

(i) The hardware support provided for floating point numbers varies between computer systems. Outline the range of support found explaining the major differences and similarities. [10 Marks]

(ii) The figure shows the fields for a 32-bit representation of a floating point number.



Explain how the IEEE single precision floating point number standard makes use of these fields to represent a number and give a formula for the relationship between the number represented and the fields in the representation. [8 Marks]

(iii) Give the smallest and largest positive normalised numbers that can be represented in single precision. [4 Marks]

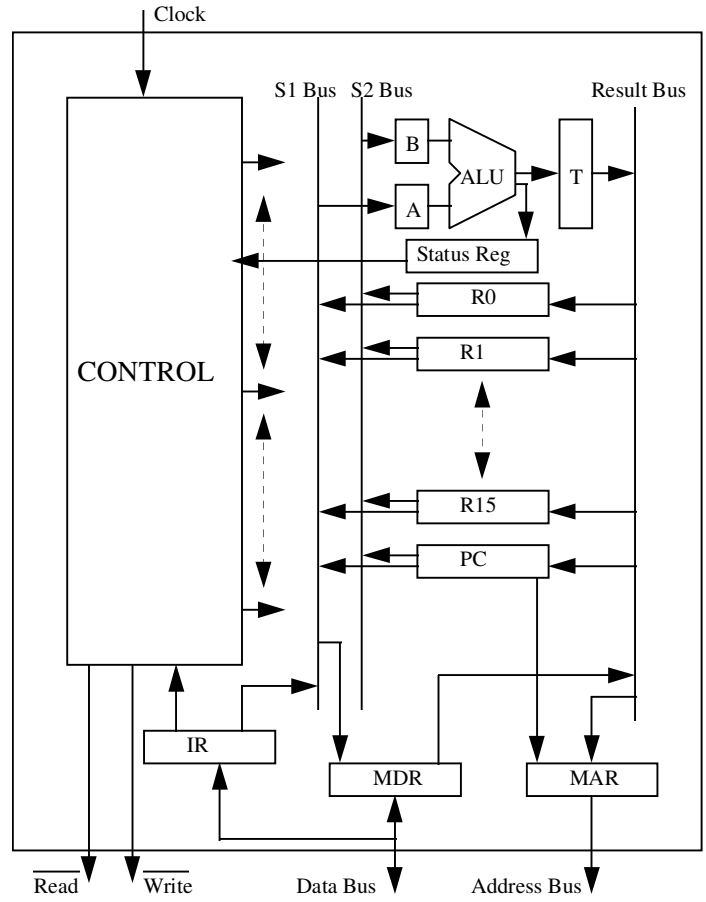
(iv) Express the hexadecimal floating point number 78C•19C in single precision format. [7 Marks]

(v) Explain the purpose and encoding of the following IEEE single precision forms
(a) Non-normalised numbers (b) NaN (Not-A-Number)
[4 Marks]

END OF QUESTION 1

TURN OVER

Describe the function of the major elements of the CPU in the diagram below and give an outline of the organisation and operation of these elements. [16 Marks]



Give a detailed description of the sequence of operations involved in the fetch and execution of an instruction to perform the operation with the Register-Transfer language description $R1 + R2 \longrightarrow R2$. [11 Marks]

Draw a state diagram for the high-level control of the CPU and briefly explain its operation. [6 Marks]

END OF QUESTION 2

QUESTION 3 C1 1996

What fundamental groups of instructions would you expect to find in the instruction set for a CPU? Identify the basic set of instructions that you might expect to be in each group and briefly explain why these are fundamental.

Explain why many CPUs have more than this basic set of instructions, illustrating your answer with at least 2 examples. [12 Marks]

The addressing modes - *literal, absolute, register,* and *displacement* - can be considered to form a basic set. The Motorola 68000 CPU has all these modes, but also has the additional modes *register indirect, index, pre-decrement,* and *post-increment*.

Explain each of the addressing modes of the basic set with examples of use in appropriate instructions and with Register-Transfer-Language descriptions. Show how each of the additional 68000 addressing modes can be implemented with the basic set of addressing modes through the use of appropriate instruction sequences. [13 Marks]

What are the key issues in the encoding of instructions (i.e. the representation of an instruction as a set of bits)? How and why do *load/store* architectures encode all instructions in the same number of bits? [8 Marks]

END OF QUESTION 3

QUESTION 4

Several forms of storage can be identified in typical computer systems:-

registers, cache memory, main memory, read-only-memory, magnetic disk and *tape storage*

Discuss the differences between these with particular reference to their usage, styles of access and access times. [18 Marks]

A CPU has 32-bit registers, a 16-bit data bus, and byte-addressable memory (i.e. each memory location holds a byte). Describe the typical memory organisation for such a system and discuss how 32-bit transfers are performed between registers and memory, with particular reference to the addresses (odd & even) output by the CPU. [10 Marks]

What is the difference between *big-endian* and *little-endian* systems and when is the difference important? [5 Marks]

END OF QUESTION 4

Discuss the role, operation and hardware requirements of **3** of the following:-

i) Paging

ii) DMA

iii) Pipelining

iv) Interrupts

v) Superscalar & VLIW

END OF QUESTION 5

END OF PAPER