UNIVERSITY OF EDINBURGH

FACULTY OF SCIENCE AND ENGINEERING

COMPUTER SCIENCE THREE

COMPUTER DESIGN

Tuesday 17 September 2002

14:00 to 15:30

Examiners:

Prof. K. Turner (External)

Prof. C. Stirling (Chairman)

INSTRUCTIONS TO CANDIDATES

Answer any TWO questions.

All questions carry equal weight.

1. (a) A circuit is required which will convert a four bit binary number, presented on four wires, into the same number plus five (modulo 16), output on four wires.

   Design this circuit, as a *minimised* sum-of-products design. Give a logic diagram for the circuit, using only NAND gates and inverters. [*10 marks*]

   (b) A combinational logic circuit with four inputs, a, b, c and d, and a single output z, has this design: $z = b\overline{c}\overline{d} + ac$.

   Explain how this circuit can momentarily produce an incorrect output when one of its inputs changes. Show how you would modify the circuit to remove this behaviour. [*8 marks*]

   (c) Briefly outline the different kinds of programmable logic device available for implementing combinational logic circuits. What are their relative advantages and disadvatages? [*7 marks*]

2. (a) Give a truth table for a *full adder* circuit, and from it derive a logic diagram for an efficient implementation for the full adder. In terms of gate delays, what would be the maximum propagation delay of a 32-bit ripple-carry adder using your circuit? [*8 marks*]

   (b) Describe the design of each of the following, in each case giving Boolean expressions or logic diagrams to describe the various parts of the circuit. Derive the maximum propagation delay, and an estimate for the number of gates used, for each design:

      i. a 4-bit look-ahead carry adder; [*8 marks*]

      ii. a 32-bit look-ahead carry adder. [*9 marks*]

3. Provide short answers to *three* of the following (all parts carry equal credit):

   (a) What is the *Huffman* model for synchronous sequential logic circuitry? Illustrating your answer with Huffman model diagrams and sample state diagrams for each, explain the difference between the *Moore* and *Mealy* finite state machine models. Are there any practical reasons for designing sequential logic using one of these two models rather than the other?

   (b) What is the difference between *level-triggered* and *edge-triggered* flip-flops? Explain the meaning of the terms *setup time* and *hold time* of each of these flip-flops. What are the implications of the set-up and hold time limits for the connection of asynchronous input signals to a synchronous sequential circuit?

   (c) What are the functions of the *data path* and *control section* of an instruction-set processor, and what sub-units might each be constructed from?

   (d) Outline the design of a hardware multiplier for floating-point numbers.

[*25 marks*]