COMPUTER SCIENCE TRIPOS Part Ib

Monday 2 June 2003 1.30 to 4.30

Paper 3

Answer **five** *questions.*
Submit the answers in five **separate** *bundles, each with its own cover sheet. On each cover sheet, write the numbers of* **all** *attempted questions, and circle the number of the question attached.*

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

1 Computer Design

The way in which an instruction's operands are specified is dependent on the type of internal storage in the processor. Processors which have no internal storage use memory.

(*a*) Describe the advantages of including some form of internal storage, such as an accumulator, in the processor. [5 marks]

(*b*) Describe *three* types of internal storage. In each case describe the format of an addition instruction. Indicate the presence of any implicit operands where necessary. [7 marks]

(*c*) Describe how data-forwarding improves performance in a pipelined load/store architecture (RISC). What common characteristic of programs often causes this performance improvement to be very significant? [8 marks]

2 ECAD

A Verilog programmer has written the code below in an attempt to produce a periodic square wave with a programmable high:low ratio. This module forms part of an interface between a microprocessor software algorithm and a variable speed electric motor.

The correct behaviour should be (1) when `go` is low the output is zero, (2) when `go` is high the output oscillates, (3) `tcycle` and `ton` are read at the start of each oscillation cycle.

```
module pwmcontroller(out, tcycle, ton, go, clk);
  output out;           // controlled output, 1=on, 0=off
  input [15:0] tcycle;  // cycle (on+off) time in clock cycles
  input [15:0] ton;     // on time in clock cycles
  input        go;      // enable the output
  input        clk;     // system clock
  parameter    sReload=0, sOn=1, sOff=2;
  reg [1:0]    state;
  reg          oncount, offcount;
  always @(posedge clk) begin
    if (!go) state <= sReload;
    case (state)
      sReload: begin
          oncount  <= ton;
          offcount <= tcycle-ton-1;
          state <= sOn;
        end
      sOn:  if (oncount==0)  state <= sOff; else oncount  <= oncount - 1;
      sOff: if (offcount==0) state <= sOn;  else offcount <= offcount - 1;
    endcase
  end
  assign out <= state==sOn;
endmodule
```

(*a*) The above code contains errors. Write a corrected version marking any changes. [6 marks]

(*b*) Draw a labelled state diagram of the corrected circuit's operation. [3 marks]

(*c*) Explain what clock gating is and why it is used. How might it be applied to the above circuit? [6 marks]

(*d*) What functionality could easily be moved to the software algorithm to reduce the size of the circuit? How would this change the module interface? [2 marks]

(*e*) Sketch an example standard cell ASIC design flow and state the importance of production testing. [3 marks]

3 Data Structures and Algorithms

(a) In the first phase of heapsort, an initially random vector is rearranged to satisfy the heap structure constraints. Describe what these are, how the rearrangement is done, and prove that it can be done in $O(n)$ time, where n is the number of elements in the vector. [7 marks]

(b) Complete the description of heapsort and show that its worst case performance is $O(n \log n)$. [7 marks]

(c) How many element comparisons would your implementation use to sort the integers 1 to 8 if they were (i) initially in sorted order, and (ii) initially in reverse sorted order? Explain how you obtained your answers. [6 marks]

4 Comparative Programming Languages

It has been said that "you can only become a really effective user of a programming language if you have a good understanding of how all its features are implemented". Discuss to what extent this is true. Your answer should include consideration of the representation of data, function calling mechanisms, space allocation and the implementation of object oriented features for a variety of different languages.

[20 marks]

5 Operating Systems II

Modern server-class machines often use a Redundant Array of Inexpensive Disks (RAID) to provide non-volatile storage.

(*a*) What is the basic motivation behind this? [2 marks]

(*b*) Describe RAID level 0. What are the benefits and drawbacks of this scheme? [3 marks]

(*c*) Describe RAID level 1. What are the benefits and drawbacks of this scheme? [3 marks]

(*d*) Compare and contrast RAID levels 3, 4 & 5. What problem(s) with the former pair does the last hope to avoid? [6 marks]

A server machine has k identical high-performance IDE disks attached to independent IDE controllers. You are asked to write operating system software to treat these disks as a RAID level 5 array containing a single file system. Your software will include routines to read file-system data, write file-system data, and to schedule these read and write requests. What difficulties arise here and how may they be addressed? [6 marks]

6 Numerical Analysis I

(*a*) For IEEE Double Precision $\beta = 2$, $p = 53$, $e_{\min} = -1022$, $e_{\max} = 1023$. Explain the meaning of these parameters and deduce the number of bits required to store the *sign*, *exponent* and *significand*. How many bytes are required in total? [5 marks]

(*b*) What is the *hidden bit* and what is its value for *normalised numbers*, and for *denormal numbers*? [2 marks]

(*c*) Define *machine epsilon* ϵ_m. What is its value for IEEE Double Precision? [3 marks]

(*d*) Suppose $f(x) = O(1)$, $f'(x) = O(1)$ and

$$\frac{f(x+h) - f(x)}{h}$$

is to be used with IEEE Double Precision to estimate $f'(x)$ and $f''(x)$. State what value of h you would use in each case, and what absolute accuracy (as a power of 2) you would expect to achieve. [4 marks]

(*e*) Special purpose floating-point hardware is to be designed with the following specification. Each number is to occupy 6 bytes but otherwise obey the principles of IEEE arithmetic as far as possible. The arithmetic must be sufficiently accurate that second derivatives can be computed to an absolute accuracy of 10^{-3} if $f(x) = O(1)$, $f'(x) = O(1)$. Deduce the parameters of this arithmetic. [Hint: $10^{-3} \simeq 2^{-10}$ is sufficiently accurate.] [6 marks]

7 Computation Theory

What is meant by a *register machine*? Explain the action of a register machine program. [6 marks]

What does it mean for a partial function $f(x_1, \ldots, x_n)$ of n arguments to be *register machine computable*? [3 marks]

Design register machines to compute the following functions.

$f(x_1, x_2) = x_1 + x_2$ [2 marks]

$g(x_1) = \begin{cases} 42 & \text{if } x_1 > 0 \\ \text{undefined} & \text{otherwise} \end{cases}$ [2 marks]

$h(x_1) = 2^{x_1}$ [4 marks]

Give an example of a function that is not register machine computable, stating clearly any well-known results you use. [3 marks]

8 Computer Graphics and Image Processing

(*a*) Pick *two* of these three colour spaces: Lab, CMYK, HLS. For each of your chosen two colour spaces, explain what each of the dimensions represents and for what uses the colour space is best suited. [6 marks]

(*b*) Describe a run-length encoding method for greyscale images. [6 marks]

(*c*) The following is a Bezier curve drawing algorithm which includes bounding box clipping. Provide pseudocode for the functions `InBoundingRect` and `DrawUnclippedBezier`.

```
function DrawClippedBezier(float x1, y1, x2, y2, x3, y3, x4, y4)
begin
  if NearlyStraight(x1, y1, x2, y2, x3, y3, x4, y4)
  then DrawClippedLine(x1, y1, x4, y4)
  else begin
      r = InBoundingRect(x1, y1, x2, y2, x3, y3, x4, y4) ;
      if(r==0) then DrawUnclippedBezier(x1, y1, x2, y2, x3, y3, x4, y4);
      if(r==1) then begin
          DrawClippedBezier(x1,y1, (x1+x2)/2,(y1+y2)/2,
            (x1+2*x2+x3)/4,(y1+2*y2+y3)/4,
            (x1+3*x2+3*x3+x4)/8,(y1+3*y2+3*y3+y4)/8);
          DrawClippedBezier((x1+3*x2+3*x3+x4)/8,(y1+3*y2+3*y3+y4)/8,
            (x2+2*x3+x4)/4,(y2+2*y3+y4)/4,(x3+x4)/2,(y3+y4)/2,x4,y4);
      end ;
      if(r==2) then return ;
  end ;
end;
```

Notes: The bounding rectangle is defined by the four (global) floating point variables `left`, `right`, `top`, and `bottom`. You may assume that we have two line drawing functions available `DrawClippedLine` and `DrawUnclippedLine`. The former draws a line having first clipped it to the bounding rectangle, the latter just draws a line without regard for the bounding rectangle (which should therefore only be used if the programmer has assured him- or herself that the line will not extend beyond the bounding rectangle). The two functions `DrawClippedBezier` and `DrawUnclippedBezier` do the same for Bezier curves. The function `NearlyStraight` returns `true` if the Bezier curve lies within half a pixel of a straight line from its first to its last point along its entire length, otherwise it returns `false`.

[8 marks]

9 Introduction to Security

(*a*) Explain the difference between mandatory and discretionary access control. [4 marks]

(*b*) (*i*) Explain the purpose and operation of cipher-block chaining (CBC). [4 marks]

(*ii*) Explain how to decrypt a message in CBC. [4 marks]

(*c*) To protect her interview partners, a journalist needs to ensure that what she records with her digital camera cannot be viewed by anyone before she returns to her home country. You were asked to design for her a camera that encrypts recordings immediately before they are stored on tape. The question arises, how to handle the encryption key. If it is stored in the camera, it could be extracted if the hardware were confiscated and analysed. A key memorised by the user might be obtained using coercion, so this is not a suitable solution either.

Suggest *two* alternative convenient ways of arranging the encryption inside the camera such that decryption of the tape is possible only on the journalist's home computer. [8 marks]

END OF PAPER