COMP 32211

Two hours

Question ONE is COMPULSORY

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Implementing System-on-Chip Designs

Date: Thursday 19th January 2017

Time: 09:45 - 11:45

Please answer Question 1 and also
TWO other questions from the remaining FOUR Questions provided

---

This is a CLOSED book examination

The use of electronic calculators is NOT permitted

[PTO]

Section A

This question is compulsory.
Answer any *ten* of the subsections.
Each subsection carries two marks.

1. a) Explain what is meant by “regression testing”. (2 marks)

   b) In digital electronics, give two distinct meanings of the term “threshold voltage”. (2 marks)

   c) Explain what is meant by “leakage” in a CMOS transistor (2 marks)

   d) What might cause *metastability* in a flip-flop or latch? (2 marks)

   e) A graphical display has a spatial resolution of 1920×1080 pixels. It is desired to increase this to 2048×1152 (i.e. the same, 16:9 aspect ratio) and retain the same frame update rate. *Estimate* the necessary increase in frame store bandwidth. (2 marks)

   f) How does a ‘FinFET’ vary from a more conventional CMOS FET? (2 marks)

   g) Give two distinct reasons why the power supply voltage of state-of-the-art VLSI devices has been falling over recent years. (2 marks)

   h) What is ‘clock jitter’? How is it accounted for during *timing closure*? (2 marks)

   i) A Verilog description may contain accurate, absolute delays (e.g. `#5`). The nominal delay of gates and networks can be *extracted* from a synthesized circuit, at least once layout is complete. Why, therefore, are such delays rarely or never used as part of the circuit function? (2 marks)

   j) In a particular Verilog simulation trace, outputs may appear in the states ‘X’ and ‘Z’. What might you reasonably deduce about these signals? (2 marks)

   k) Explain why a *scan chain* can reduce the number of test patterns needed when production-testing a complex SoC. (2 marks)

   l) What is meant if a data processing pipeline is said to be “well balanced”? (2 marks)

Section B

Answer any *two* questions from this section.

2. a) Outline how a *test coverage tool* can support the verification of an RTL design and where it fits in the tool flow when developing digital hardware.
What assistance can such a tool give in improving confidence in a functional design? (4 marks)

b) Give two distinct examples of the sort of information which can be obtained from a test coverage tool. (4 marks)

c) Briefly describe *three* different examples of problems in a Verilog specification of a hardware module which will *not* be uncovered by using a test coverage tool on a simulation run. (6 marks)

d) Imagine you have to develop *tests* for a hardware module which counts the number of '1' bits in a 128-bit word. Describe the strategy (or strategies) which you would employ in preparing a 'stimulus' file which might give reasonable confidence in the module's functionality.
(If it helps, you may use pseudo-code to illustrate your answer; syntactically correct Verilog is not necessary.) (6 marks)

3. Figure 1 shows a transistor-level schematic for a standard library component. Each transistor has been annotated with its *width* as a multiple of an arbitrary length λ; the *lengths* of the transistor gates are all the same. The drawn size of the transistors in the figure has no specific meaning; it is for emphasis to reflect their relative widths only. Subsections in this question refer to parts of this figure.

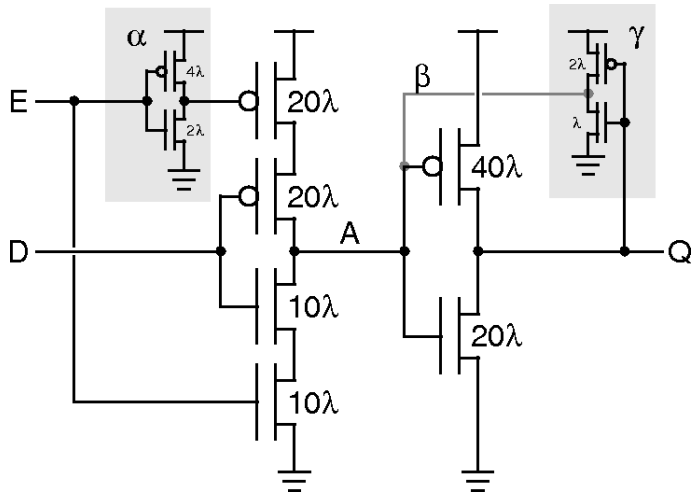

Figure 1: A typical library standard cell schematic.

a) What logical function does the subcircuit highlighted as α perform? (1 mark)

b) Write down the truth table for the intermediate node 'A' in terms of the inputs 'D' and 'E'.
For the purpose of this part of the question, the grey wire, labelled β*, should be imagined not to exist.* (2 marks)

c) From your previous answer, and still ignoring wire β, deduce the truth table for the output 'Q' as it would be evaluated using a Verilog digital simulation. (2 marks)

d) Taking the output of subcircuit γ into account via the wire β, there are two CMOS outputs connected to point A. This is normally a bad idea so why is this legitimate the circuit in figure 1? (3 marks)

e) Taking the output of γ into account via the wire β, what effect does that have on the behaviour of the circuit? Draw the simulation timing diagram for 'D', 'E' and 'Q' when the following 'stimulus' is applied.

```
initial
  begin
  D = 0; E = 0;
  #10;   E = 1;
  #10;   E = 0;
  #10;   D = 1;
  #10;   E = 1;
  #10;   E = 0;
  #10;   D = 0;
  #10;
  end
```

At any 'interesting' points, explain what will be happening at node 'A'.
(You may add this to the timing diagram if it makes things clearer.) (4 marks)

f) It may be observed that the width of the PMOS transistors in each 'stack' is always twice that of their NMOS complements: why is that? (2 marks)

g) The α subcircuit is not used in the same way as the γ subcircuit but still has significantly smaller transistor widths than the majority of the circuit. Suggest why this might be appropriate. (2 marks)

h) What is this component usually called? (2 marks)

i) Write a (simple) Verilog statement/block which implements the function performed by the circuit in figure 1. (2 marks)

4. a) Explain why the full speed *clock generation* for most SoC logic is done *on chip*. (2 marks)

   b) Write down a design of a circuit which would *divide* an input clock by four. The output should have a 50% duty cycle.
   You may express this in a form of your choosing which would be (in principle) suitable for entry into CAD tools. (4 marks)

   c) Describe how a phase-locked loop can *multiply* a clock frequency. (4 marks)

   d) Suggest, approximately, how the power consumption of a synchronous digital logic block on a CMOS SoC is related to its clock frequency. Justify your answer. (2 marks)

   e) Using enable signals, as is done implicitly in the Verilog code fragment below, is one way of controlling updates to a set of registers.

```
always @ (posedge clk)
  begin
  if (enabled)
    case (address[1:0])
      2'h0: register_0 <= data_in;
      2'h1: register_1 <= data_in;
      2'h2: register_2 <= data_in;
      2'h3: register_3 <= data_in;
    endcase
  end
```

   Another mechanism which prevents register updates is *clock gating*. Briefly discuss:

   (i) the relative merits of these two mechanisms
   (ii) where it is most appropriate to use each (4 marks)

   f) Explain what is meant by 'Dynamic Voltage and Frequency Scaling' (DVFS). What (if any) benefits can it confer? (4 marks)

5. a) When verifying a digital design it is normal to simulate at different levels of detail. Simulations may be conveniently classified into ‘digital switch-level’ simulations and ‘analogue’ simulations. State *two* ways in which these ‘classes’ differ. (4 marks)

   b) During the overall SoC development process, which type of simulator is usually applied first, and why? (2 marks)

   c) If a register is not explicitly initialised, how is its output represented in each type of simulation. (2 marks)

   d) Write some Verilog source code which will synthesize to a synchronous, positive edge-triggered, 8-bit register with an *active low, asynchronous* clear and a clock enable. (4 marks)

   e) In a synchronous logic block comprising several interacting FSMs it may be important that the *inactive* edge (at least) of reset is *synchronised* to the clock. Why is that? (2 marks)

   f) In a digital design it is important that some specific registers reset to *defined* values. What facilities are available in a Verilog-based design flow to determine if this has *not* has happened correctly? (4 marks)

   g) When developing a new ‘smartphone’ SoC give one example of some storage element which may reasonably *not* require an explicit reset. In one sentence, say why (or if) there may then be some benefit in not resetting it. (2 marks)

END OF EXAMINATION