The University Of Sheffield.

DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING

Spring Semester 2010-2011   (2 hours)

EEE310 Introduction to VLSI Design 3

Answer **THREE** questions. **No marks will be awarded for solutions to a fourth question.** Solutions will be considered in the order that they are presented in the answer book. Trial answers will be ignored if they are clearly crossed out. **The numbers given after each section of a question indicate the relative weighting of that section.**
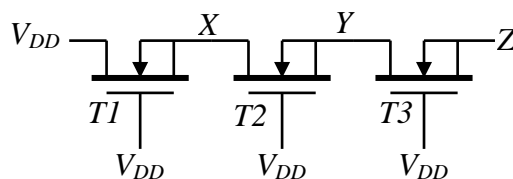
1. **a.** Show that the small-signal, on-state resistance of an *n*-type pass transistor as shown in **Figure 1(a)** is $r = \dfrac{1}{\beta_N \cdot (V_{DS} - V_T)}$ when $V_G = V_D$
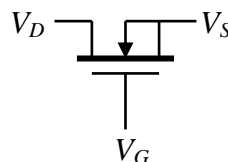
Figure 1(a): *n*-type Pass Transistor   (4)

   **b.** Three *n*-type pass transistors are cascaded in series as shown in **Figure 1(b)**.

Figure 1(b): Cascaded *n*-type Pass Transistors

What will the approximate voltages at points *X*, *Y*, and *Z* be for the voltage conditions shown. State any assumptions that you make.   (4)

   c. State whether each transistor in **Figure 1(b)** would be in the ohmic or saturated region if a current were to flow.   (4)

   d. In **Figure 1(b)**, the *substrate* connection is shown connected to the transistor's *source* connection. If this circuit were part of a digital MOS IC where would you normally expect the *substrate* to be connected for transistors such as these and why would this be the case?   (4)

   e. Show how two transmission gates and two inverters can be combined together to create an XOR gate.   (4)

2.   A logic circuit is shown in **Figure 2**.
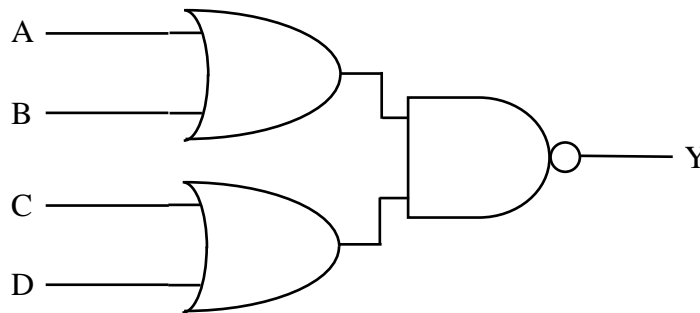
Figure 2: Logic Circuit

a.   Convert the logic circuit into a standard-CMOS transistor-level circuit. **(8)**

b.   Size the transistors (as a multiple of a minimum-sized *n*-type FET) for a minimum sized logic circuit, stating any assumptions that you make. **(6)**

c.   Estimate the capacitance associated with each of the inputs and wires within the circuit (you can neglect the interconnect capacitance), given that the gate capacitance of a minimum-sized *n*-type FET is 0.7fF. **(4)**

d.   Rather than implementing the circuit in **Figure 2** as a single, combined CMOS circuit, you decide to implement it as a combination of two separate CMOS OR gates and a CMOS NAND gate. How much less efficient will it be than the single CMOS circuit in terms of numbers of transistors used? **(2)**

3. **a.** **i)** What is *fault-coverage* when applied to testing? **(3)**

ii) Why is *fault-coverage* so important in ensuring that systems manufactured using ICs can be sold? **(1)**

iii) Show the effect of *fault coverage* using an IC yield of 0.7, with a fault-coverage of 98%.

You should assume that the ICs are tested, then 20 ICs are assembled into a system where the manufacturing yield on the system assembly is 99.5% and, again, the *fault-coverage* on testing the systems is, again, 98%. **(4)**

iv) How do IC designers achieve a high value of *fault-coverage*? **(2)**

v) What sort of things tends to limit *fault-coverage*? **(2)**

b. A section of combinatorial logic (between flip-flops) is shown in **Figure 3**.
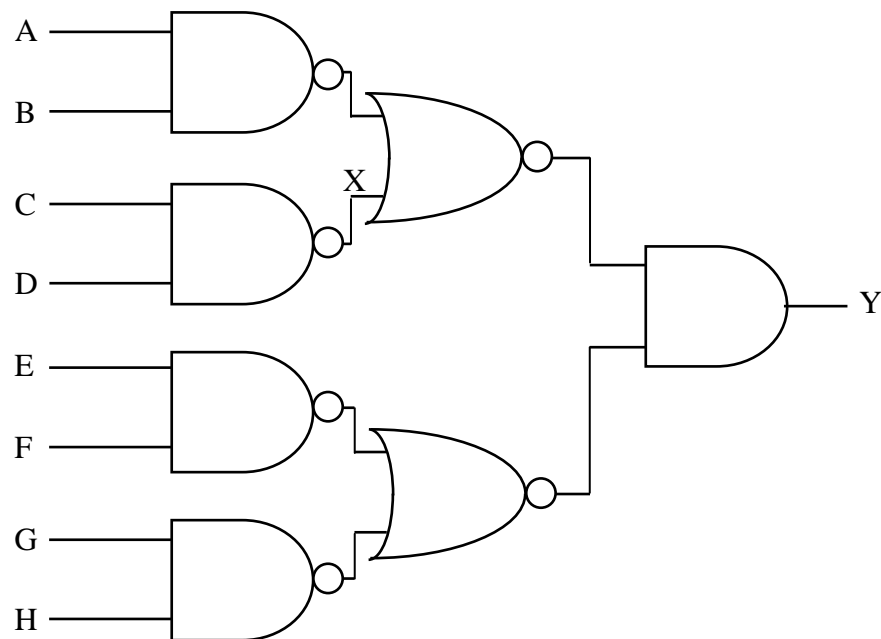
Figure 3: Combinatorial Logic to be Tested

i) If this logic were to be tested exhaustively then how many individual tests would be required? **(2)**

ii) Using a stuck-at fault-modelling approach, how many individual tests would be required? **(2)**

iii) In fault-modelling what is the difference between sensitisation and propagation? **(2)**

iv) Identify the input conditions that would be required to test node X for both stuck at 1 and stuck at 0. **(2)**

4. **a.** **i)** Describe the *Synchronous Design Methodology* and explain why it results in reliable designs. **(4)**

ii) How does synchronous design relate to a *Register Transfer Level* description of circuits? **(2)**

b. **i)** Languages used for hardware design, like VHDL, allow you to specify a delay between an input (on the RHS) in a logical expression changing state and the corresponding output (LHS) changing state. Why is this meaningless in logic synthesis? **(3)**

ii) When functionally simulating logic designs, where the delay between inputs to and outputs from logical expressions is not specified, how would the simulator ensure that causality is obeyed? **(3)**

iii) In this regard, what might happen if a simulator were to encounter the following statement (written in VHDL):

a <= not a; **(2)**

iv) What is the difference between functional simulation and timing simulation? **(2)**

v) Timing simulation can often be avoided by using static timing analysis. What is static timing analysis and why is it used? **(4)**

NLS