EECS 105 Prof. Wawrzynek
Spring 2003 MT2

1. (10 Points) Short Answer

a) The largest (and most advanced) chips produced today contain approximately how many transistors? (100 thousand, 100 million, 100 billion)

b) Moores law says that the number of transistors on a chip doubles every how many months?

c) FPGAs are often used instead of ASICs in digital systems because they (reduce time to market, improve performance, or both)?

d) Internally an n-LUT may be implemented using how many latches?

e) An n-LUT can implement how many different functions?

f) Write the Boolean equation for the sum output (s) of a full-adder cell:

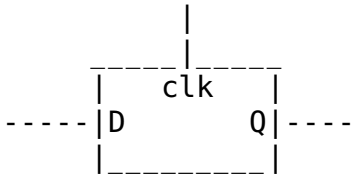
g) Write the Boolean equation for the carry output (c) of a full-adder cell:

h) Sketch the circuit for a 4-bit ripple-adder based on instances of the full-adder cell:

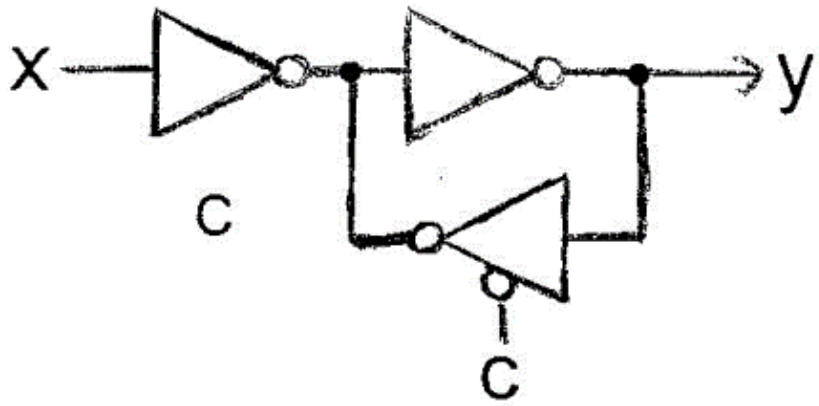
i) Simplify (x + y)(x + y):

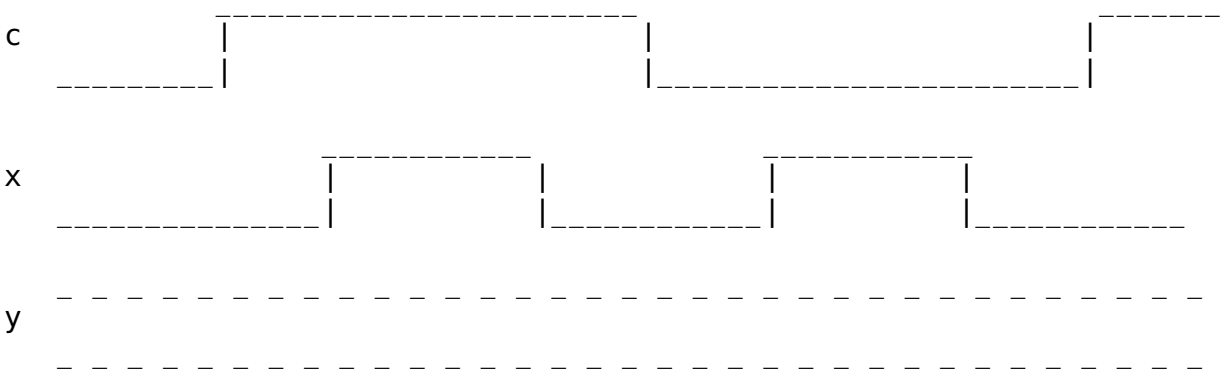
j) Write the canonical product-of-sums form for exor(a, b):

2. a) (3 Points) Using instances of a positive-level sensitive latch, shown below, along with any simple logic gates required, draw the circuit for a *negative-edge triggered D-type flip-flop*.

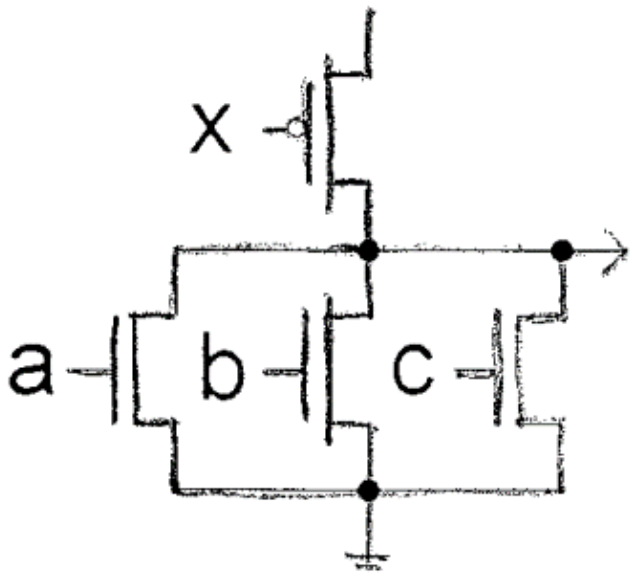


b) (5 Points) The waveforms below for the signal c and x are applied to the circuit to the right. Fill in the resulting waveform that would appear at y.

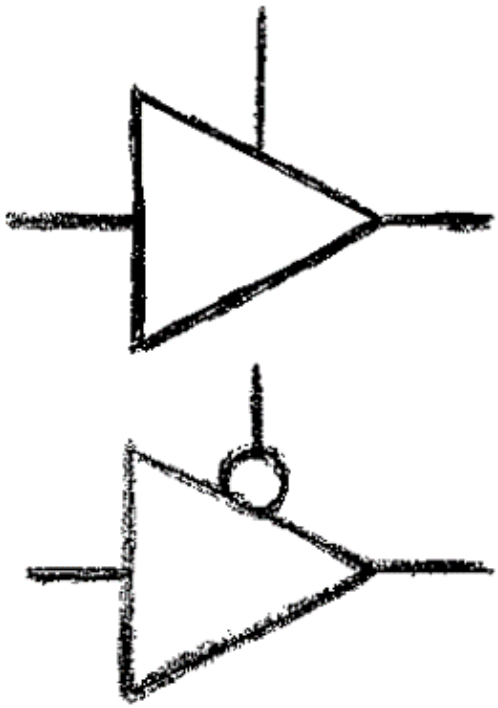




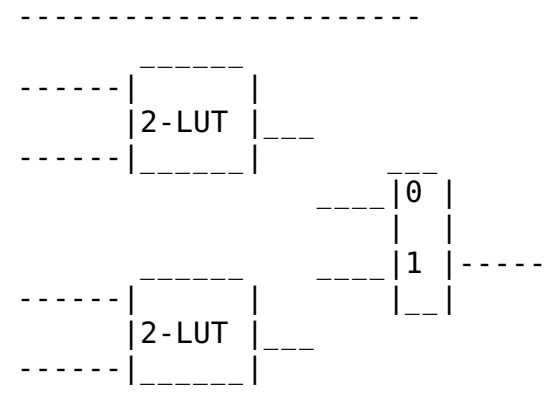
3. a) (3 Points) The circuit shown below is a kind of 3-input NOR-gate. Write a Boolean expression in terms of a, b, and c, that would need to be applied to x for correct operation:



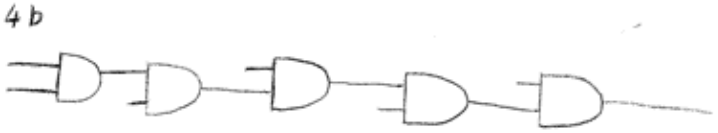
b) (7 Points) Using nothing but non-inverting tristate buffers, shown below, draw a circuit that implements a 3-input multiplexor. Label the inputs as a, b, and c, the control as s0 and s1, and the output as x. The multiplexor has the following action: is [s1 s0] = 00, output a; if [s1 s0] = 10, output b: if [s1 s0] = 01 or 11, output c.



4. a) (5 Points) Is it possible to map the following Boolean expression to the CLB structure shown below? If so, prove it by labeling the inputs with the appropriate variables and the internal wires with the corresponding Boolean expressions. Assume that complemented inputs are not available.

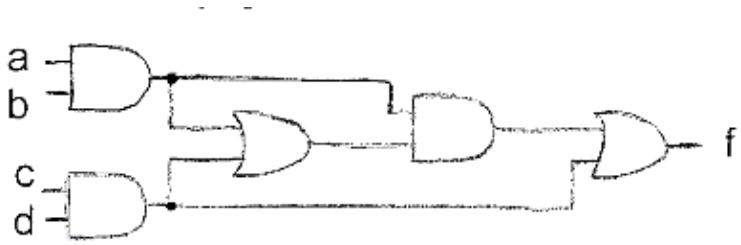


b) (3 Points) Based on the CLB structure from part a, show how to partition the following circuit into the minimum number of CLBs by carefully drawing circles around sets of gates, one circle per CLB:



c) (4 Points) Reorganize the circuit from part b) and partition it, based on the CLB structure from part a), to achieve less delay. Draw your result below:

5. Consider the following logic circuit.



a) (3 Points) Write an unreduced expression for f that corresponds directly to the circuit:

b) (4 Points) Simplify the expression (minimize the number of literals). Show all steps and your final result below:

6. (7 Points) Convert the following sum-of-products expression to reduced product-of-sums form. *Hint: use a K-map.*

ABC + ABC + ABC + ABD

7. (7 Points) Derive a circuit that implements the following Boolean expression using as few 2-input NAND-gates as possible. Show your work and the resulting circuit below. *Complemented inputs are not available.*

$$(A + B)(C + D) + (E + F)(G + H)$$

8. Consider the design of a Verilog module that computes the maximum value of two unsigned 2-bit integers, A an B.

a) (5 Points) Write the *behavioral* Verilog description below:

b) (12 Points) Using instances ony of simple logic gates (with any number of inputs), write a structural Verilog description. Your description must use the minimum total number of logic gates. Show your work.

Logic Derivation and simplification:

Write the *structural* Verilog description below (gates only):

9. Consider the design of a Moore style finite state machine (FSM) with two inputs, clk and CE, one 2-bit wide output X. clk is the clock signal

and CE is the "count enable" signal. While CE=1, the FSM behaves as a "binary counter", i.e. its output cycles through the pattern 00, 01, 10 11, 00, moving from one output value to the next on each positive edge of clk. If CE = 0 the output value remains unchanged.

Note that FSM has no reset input signal. You can assume that it starts up in an
a) (3 Points) Sketch the state transition diagram that represents the behavior of this FSM:

b) (2 Points) Sketch the circuit diagram for the FSM. You do not need to show the details of the combinational logic part of the circuit (just show it as a box labeled "CL"). Label all inputs and outputs:

c) (7 Points) Write a Verilog description of this FSM. You are allowed to describe the combinational logic part of the circuit using a behavioral description:

d) (10 Points) Draw a circuit diagram for a *one-hot encoded* version of the FSM. For this part, show all details of any combinational logic. Dont forget to show the output circuitry.