University of California, San Diego

M.S. Exam: Computer Engineering (ECE260A)

Spring 2014

NAME : _____

- All work to be done on the attached sheets.
- Write your name at the top of every sheet.
- 25 points/problem .

Question 1.

An AOI22 standard logic gate executes: Y = !((A & B) | (C & D)), i.e., the logical equivalent of two AND gates feeding a NOR gate. (In verilog, '&' is AND, '|' is OR, and '!' is NOT .)

(a) Design one in static CMOS logic which minimizes the number of transistors.

(b) Show the sizes of the transistors needed to obtain the same drive as a unit inverter. (Assume P mobility is 1/2 that of N.)

(c) Compute the logical effort of each of the four inputs.

(d) Provide a rough estimate of propagation delay, based on total output diffusion capacitance.

Question 2.

Use the following timing parameters and consider a combinational logic circuit which sits between two flip-flops or two latches.

	t_{setup}	clk-to-Q delay	D-to-Q delay	contamination delay	t_{hold}	clock cycle	clock pulse width
Flip-Flop	65ps	50ps		35ps	30ps	500ps	80ps
Latch	25ps	50ps	40ps	35ps	30ps	500ps	80ps

(a) What is the maximum allowable logic propagation delay between two flip-flops, with zero clock skew?

(b) Repeat (a) for two pulsed latches instead of two flip-flops.

(c) Repeat (a) with a clock skew between the two flip-flops or latches of +/− 50ps.

(d) Repeat (b) with a clock skew between the two flip-flops or latches of +/− 50ps.

Question 3.

Starting with a group of conventional 1-bit full adders, design a comparator accepts two *N*-bit binary numbers and returns: $a > b$, $a = b$, and $a < b$ when:
(You may sketch a schematic of your solution and/or write it in structural gate-level [System]Verilog or VHDL. You do not need to show how the 1-bit full adders themselves are made.)

(a) *a* and *b* are unsigned.

(b) *a* and *b* are two's complement.

(c) *a* is unsigned, but *b* is two's complement.

Question 4.

A 4:7 Hamming forward error correction (FEC) encoder generates three parity bits, C[2:0], from four data bits, D[3:0], using the formula:

C[0] = D[3]^D[1]^D[0], C[1] = D[3]^D[2]^D[0], C[2] = D[3]^D[2]^D[1].

(a) What is the Hamming distance between any two 7-bit codes so generated?

(b) Design the corresponding decoder. You may write Verilog or VHDL code or draw a schematic.

(c) How many errors can your design detect, and how many can it correct?

(d) Verify your design by showing what happens if D[3] arrives inverted, whereas D[2:0] and C[2:0] all arrive correctly.