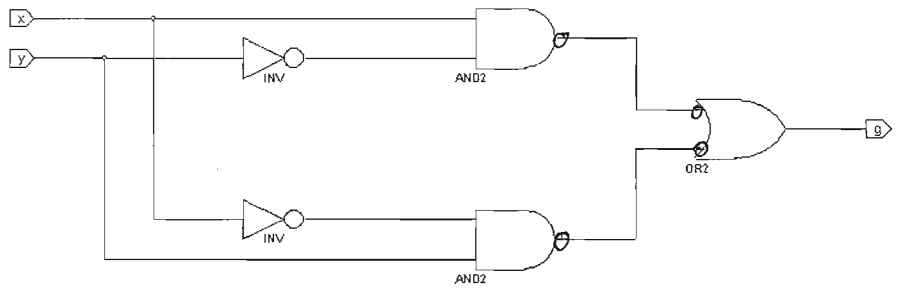
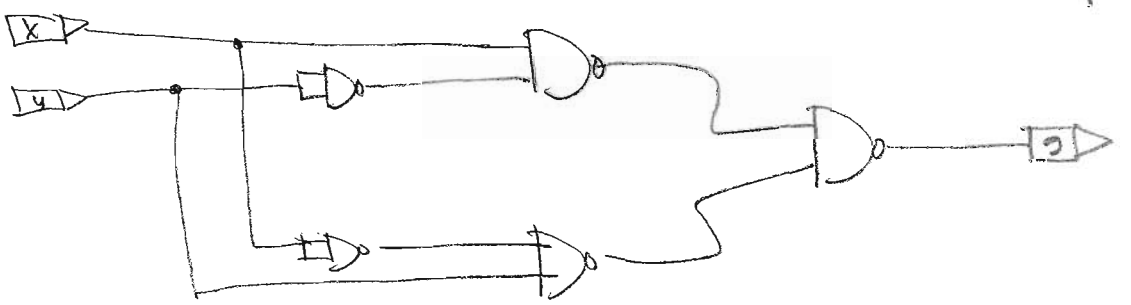
Boise State University
Department of Electrical and Computer Engineering
EE 230 Digital Systems
Test 1 – February 17, 2005

Name: _____

Instructions: Show all steps and logic circuits for full or partial credits. It is very important that you write clearly, so that your test can be graded appropriately and fairly. This is a closed book and closed notes test. A Boolean algebraic basic rules table has been provided.

1. Convert the following logic circuits using just NAND gate.





De Morgan's law

$\overline{xy} = x' + y'$

$\overline{x+y} = x'y'$

2. Convert the following decimal numbers to binary representation.

a. 44

$(101100)_2$

2 | 44 → 0
2 | 22 → 0
2 | 11 → 1
2 | 5 → 1
2 | 2 → 0
2 | 1 → 1
0

b. 11.375

$(1011.011)_2$

2 | 11 → 1
2 | 5 → 1
2 | 2 → 0
2 | 1 → 1
0

0.375 × 2 = 0.75
0.75 × 2 = 1.50
0.50 × 2 = 1.00

3. Find the octal and hexadecimal representations of the following binary numbers.

Binary Number	Octal Representation	Hexadecimal Representation
a. 1011010011110	13236	169E
b. 000001010101	0125	055

(10)

4. Perform the following additions and subtractions using 2's complement (8-bit representation).

(10) (a)

$$\begin{array}{r} 50 \\ +\ 75 \\ \hline -125 \end{array} \Rightarrow \begin{array}{r} 00110010 \\ +\ 01001011 \\ \hline 01111101 \end{array}$$

Overflow? YES / NO

$C_n = 0 \quad C_{n-1} = 0 \quad C_n \oplus C_{n-1} = 0$

2|50
2|25 1
2|12 0
2|6 0
2|3 1
2|1 1
0

(b)

$$\begin{array}{r} (-50) \\ -\ (+75) \\ \hline -125 \end{array} \Rightarrow \begin{array}{r} 11001110 \\ -\ 01001011 \\ \hline \end{array}$$

Overflow? Yes / No

$$\begin{array}{r} 11001110 \\ +\ 10110101 \\ \hline 1\ 10000011 \end{array}$$

this bit can be ignored!

$C_{n-1} = 1 \quad C_n = 1 \quad C_{n-1} \oplus C_n = 0$

2|75
2|37
2|18
2|9
2|4
2|2
2|1
0

5. Discuss the pros and cons of using discrete components versus FPGA to build logic circuit.

(10)

Discrete : pros : ① cost effective
② can be easy to build for simple circuit

cons : ① old / slow technology
② potentially involved many wires ⇒ to many weak links.

FPGA : pros : ① no wire to deal with
② complicated design can be built easily

cons : ① can be expensive
② could be complicated.

6. In lab 4, your constructed a signed 4-bit adder/subtractor using a 1-bit full adder (with 2's complement). Describe your design.

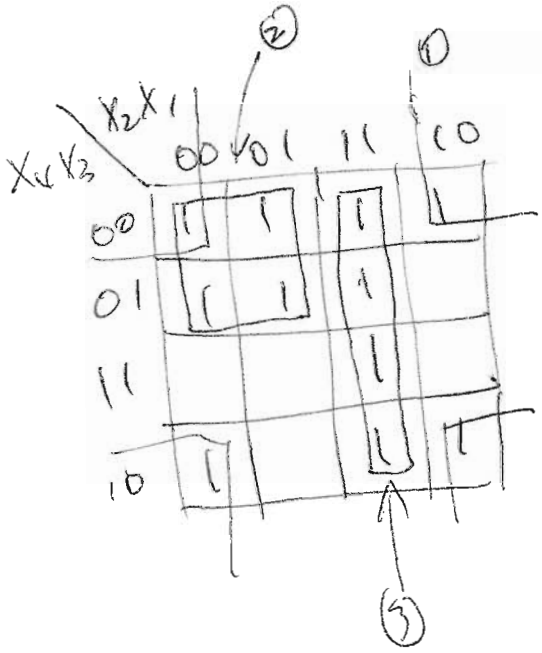
[Your description should (at least) answer these questions: Show your 1-bit design and how the 1-bit module was used to construct the 4-bit module. Describe how the overflow condition can be detected. How did you test your design using ISE and Modelsim? How was the design tested using FPGA?]

7. Logic function optimization

[Definition of cost calculation: The input variables are only available in their true form. The cost of a logic circuit is determined by number of gates plus the total number of inputs to all gates in the circuit.]

a. Simply $f(x_4, x_3, x_2, x_1) = x_1'x_3' + x_2'x_4' + x_2x_3'x_4' + x_1x_2x_3 + x_2x_3'x_4$, where x_4 is MSB and x_1 is LSB, using k-map.

7.5



$f = x_1'x_3' + x_2'x_4' + x_1x_2$ (terms ①, ②, ③)

b. Simply $f(x_4, x_3, x_2, x_1) = x_1'x_3' + x_2'x_4' + x_2x_3'x_4' + x_1x_2x_3 + x_2x_3'x_4$, where x_4 is MSB and x_1 is LSB, using algebraic manipulation.

7.5

$f = x_1'x_3' + x_2'x_4' + x_1x_2x_3 + x_2x_3'(x_4' + x_4)$

$= x_1'x_3' + x_2'x_4' + x_1x_2x_3 + x_2x_3'$ ← (term #1 and #2 a… in the form we need, so add $(x_1' + x_1)$ to te… #4 is our best bet to get what we need)

$= x_1'x_3' + x_2'x_4' + x_1x_2x_3 + (x_1' + x_1)x_2x_3'$

$= x_1'x_3' + x_2'x_4' + x_1x_2x_3 + x_1'x_2x_3' + x_1x_2x_3'$

$= x_1'x_3' + x_2'x_4' + x_1x_2(x_3 + x_3') + x_1'x_2x_3'$

$= x_1'x_3' + x_2'x_4' + x_1x_2 + x_1'x_2x_3'$

$= x_1'x_3'(1 + x_2) + x_2'x_4' + x_1x_2$

$= x_1'x_3' + x_2'x_4' + x_1x_2$

4 inverters
3 AND gates
1 OR gate
13 inputs to gates

21

c. Calculate the cost for *a* and *b*, which one better based on the cost criteria given?

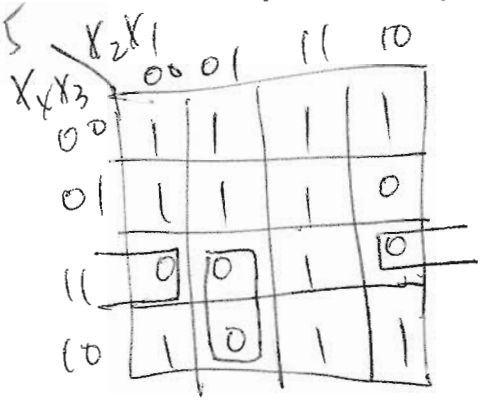
7.5

a = b

21 = 21

d. Find the POS representation of *f* and the cost.

7.5



$$f = (x_1 + x_3' + x_4')(x_1' + x_2 + x_4')$$

or

$$f' = x_1'x_3x_4 + x_1x_2'x_4$$

↓ Demorgan's Law

$$\overline{f'} = \overline{x_1'x_3x_4 + x_1x_2'x_4}$$

$$f = (x_1 + x_3' + x_4')(x_1' + x_2 + x_4')$$

3 inverters
2 OR gates
1 AND gate
11 input to gates

17