

King Fahd University of Petroleum and Minerals
College of Computer Science and Engineering
Computer Engineering Department

COE 202: Digital Logic Design (3-0-3)
Term 132 (Spring 2013)
Major Exam II
Saturday April 19, 2014

Time: 120 minutes, Total Pages: 12

Name: _____ **ID:** _____ **Section:** _____

Notes:

- Do not open the exam book until instructed
- **Calculators are not allowed** (*basic, advanced, cell phones, etc.*)
- Answer all questions
- All steps must be shown
- Any assumptions made must be clearly stated

| Question | Maximum Points | Your Points |
|-----------------|-----------------------|--------------------|
| 1 | 8 | |
| 2 | 12 | |
| 3 | 8 | |
| 4 | 12 | |
| 5 | 13 | |
| 6 | 10 | |
| 7 | 7 | |
| Total | 70 | |

[8 Points]

Question 1

Shown to the right is the K-Map of the Boolean function **F** subject to the don't care conditions **d**

$$F(A, B, C, D) = \sum(0, 1, 2, 4, 6, 10, 12)$$

$$d(A, B, C, D) = \sum(7, 13, 14, 15)$$

a) Derive the minimum **SOP** expression of **F**.

| CD \ AB | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | 1 | 1 | 0 | 1 |
| 01 | 1 | 0 | X | 1 |
| 11 | 1 | X | X | X |
| 10 | 0 | 0 | 0 | 1 |

$$F = C\bar{D} + B\bar{D} + \bar{A}\bar{B}\bar{C}$$

| CD \ AB | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | 1 | 1 | 0 | 1 |
| 01 | 1 | 0 | X | 1 |
| 11 | 1 | X | X | X |
| 10 | 0 | 0 | 0 | 1 |

Shown to the right is the K-Map of the Boolean function **G** subject to the don't care conditions **D**

$$G(A, B, C, D) = \sum(1, 4, 5, 6, 9, 12)$$

$$D(A, B, C, D) = \sum(0, 7, 10, 13, 15)$$

b) Derive the minimum **POS** expression of **G**.

$$G = (B + D)(\bar{A} + \bar{C})(\bar{C} + \bar{D})$$

| CD \ AB | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | X | 1 | 0 | 0 |
| 01 | 1 | 1 | X | 1 |
| 11 | 1 | X | X | 0 |
| 10 | 0 | 1 | 0 | X |

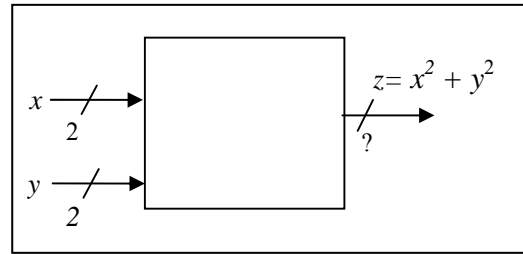
Alternatively

$$G = (B + D)(\bar{A} + \bar{C})(B + \bar{C})$$

| CD \ AB | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | X | 1 | 0 | 0 |
| 01 | 1 | 1 | X | 1 |
| 11 | 1 | X | X | 0 |
| 10 | 0 | 1 | 0 | X |

Question 2

A logic circuit has *two* inputs x & y each is a 2-bit *unsigned* number. It has an output number z such that $z = x^2 + y^2$.



- What is the minimum number of bits required for the output number z ?
- Construct the truth table of the circuit.
- Derive the Boolean expressions of the two least significant output bits (z_0, z_1) using basic gates (NO MSI parts)

Solution:

a. $\text{Max}(z) = (3)^2 + (3)^2 = 18 \rightarrow$ Requires 5-Bits \rightarrow Outputs : $Z_4 Z_3 Z_2 Z_1 Z_0$

| | | | | | |
|-----------|-----------|----|----|----|----|
| | $y_1 y_0$ | 00 | 01 | 11 | 10 |
| $x_1 x_0$ | 00 | 0 | 1 | 1 | 0 |
| | 01 | 1 | 0 | 0 | 1 |
| | 11 | 1 | 0 | 0 | 1 |
| | 10 | 0 | 1 | 1 | 0 |

| x_1 | x_0 | y_1 | y_0 | Z_4 | Z_3 | Z_2 | Z_1 | Z_0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |

$$Z_0 = \overline{x_0} y_0 + \overline{y_0} x_0$$

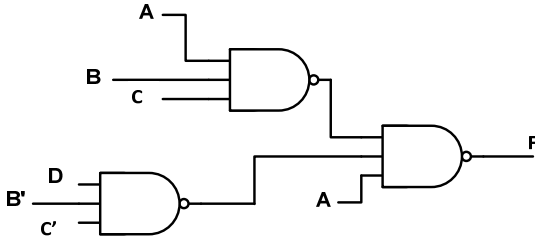
$$= y_0 \oplus x_0$$

| | | | | | |
|-----------|-----------|----|----|----|----|
| | $y_1 y_0$ | 00 | 01 | 11 | 10 |
| $x_1 x_0$ | 00 | 0 | 0 | 0 | 0 |
| | 01 | 0 | 1 | 1 | 0 |
| | 11 | 0 | 1 | 1 | 0 |
| | 10 | 0 | 0 | 0 | 0 |

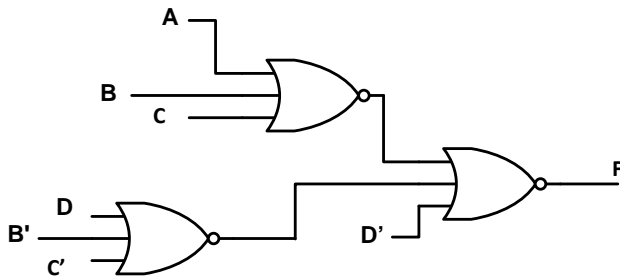
$$Z_1 = y_0 x_0$$

Question 3

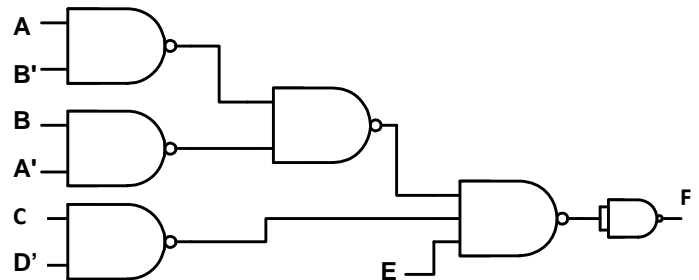
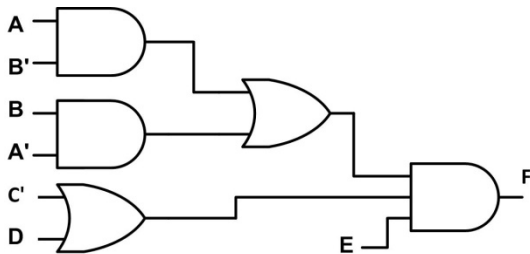
- a. Assuming the availability of the true and complement of signals A, B, C, and D, implement the function $F = ABC + DB'C' + A'$ using a minimum number of one gate type only.



- b. Assuming the availability of the true and complement of signals A, B, C, and D, implement the function $F = (A+B+C)(D+B'+C').D$ using a minimum number of one gate type only.



- c. Assuming the availability of the true and complement of signals A, B, C, D and E, implement the shown circuit using minimum number of NAND gates only.



Question 4.**(12 Points)**

Assuming that all numbers are held in 6-bit storage registers, answer the following:

- a. If 2's complement binary representation is used, what is the range of values that each number may assume? **(2 points)**

$$-2^5 \text{ to } 2^5 - 1$$

$$[-32, 31]$$

- b. The largest number that can be subtracted from (-15) without causing overflow is

$$\underline{\quad 17 \quad}$$

(2 points)

- c. Perform the following arithmetic operations *in the indicated number representation*. Then, convert the result to decimal and indicate if an *overflow* has occurred: **(8 points)**

- (i) $(10)_{10} - (24)_{10}$ (using sign-magnitude binary representation).

$$10 < 24 \rightarrow \text{-ive sign.}$$

$$10 = 001010, \quad 24 = 011000$$

$$11000$$

$$01010-$$

$$01110$$

$$= 14$$

$$\Rightarrow \text{the result} = 101110 = -14$$

No overflow

(ii) $010010 - 111111$ (using 1's complement binary representation).

$$010010 + 000000 = 010010 = 18$$

No overflow

(iii) $100000 - 100011$ (using 2's complement binary representation).

$$= 100000 + 011101$$

$$100000$$

$$011101+$$

$$111101$$

$$=-000011 = -3$$

No overflow

(iv) $010111 - 11\ 0111$ (using 2's complement binary representation).

$$= 010111 + 001001$$

$$010111$$

$$001001+$$

$$100000$$

$$=-32$$

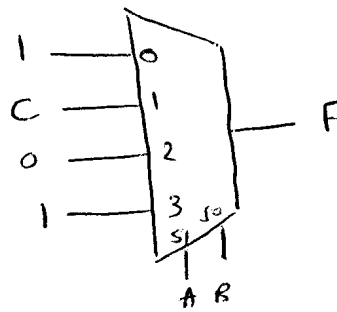
➔ No overflow

(13 Points)

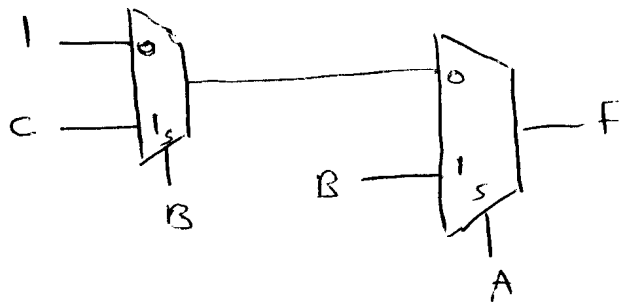
Question 5.the Boolean function: $F(A, B, C) = AB + \bar{A}C + \bar{A}\bar{B}$

- Using a single 4x1 multiplexer. (4 Points)
- Using a minimum number of 2x1 multiplexers. (2 Points)
- Using a single 3x8 decoder and an OR gate. (3 Points)
- Using a single NOR gate and the minimum number of 2x4 decoders with enable. (4 Points)

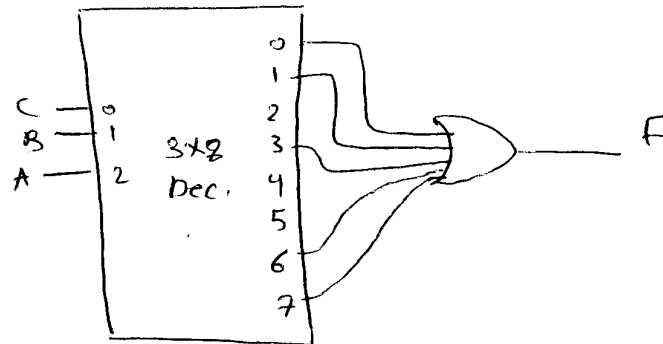
$$a. \quad F = \bar{A}\bar{B} [1] + \bar{A}B [C] \\ + A\bar{B} [0] + AB [1]$$



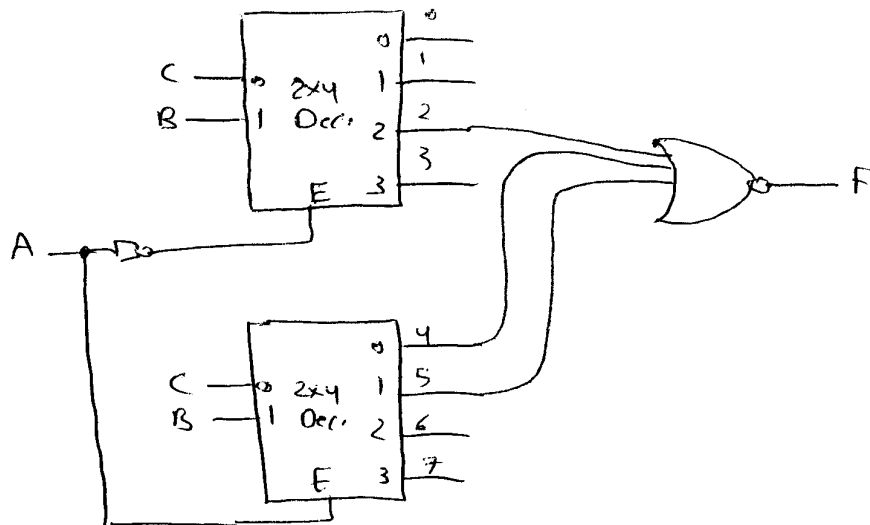
b.



c. $F = \sum m(0, 1, 3, 6, 7)$



d.

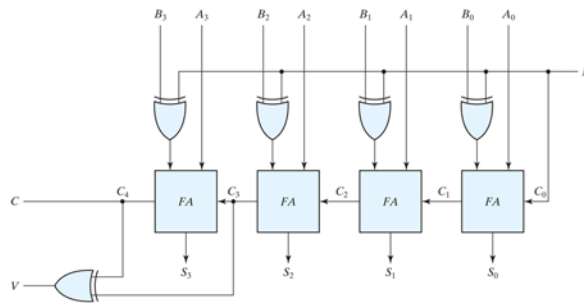


Question 6.

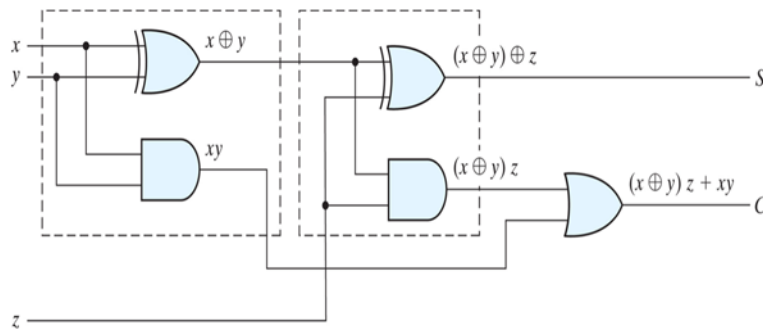
(10 Points)

- a. Design a 4-bit adder/subtractor circuit which uses the least number of Full-Adders (FAs). The circuit receives two 4-bit signed numbers **A** and **B** (2's complement representation) and one control input (**M**). If the control input $M = 0$, the 4-bit circuit output equals $(\mathbf{A}+\mathbf{B})$. If the control input $M = 1$, it equals $(\mathbf{A}-\mathbf{B})$. The circuit has another output **V** which equals **1** only in case of overflow.

| Gate | Delay (ns) |
|------|------------|
| AND | 2 |
| OR | 2 |
| XOR | 3 |



- b. Given the FA circuit shown below, calculate the worst-case delay of this adder/subtractor circuit assuming gate delays as given in the table to the right.



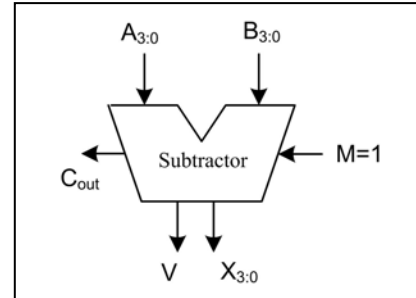
Delay = 3 for B xor + 3 for 1st HA + 4 × (4 for 2nd HA with C OR) + 3 for v xor
Delay = 25

(7 Points)

Question 7.

A 4-bit adder/subtractor circuit like the one designed in problem 6, is used here as a **subtractor** with the input control $M=1$ (see Figure).

It subtracts two 4-bit numbers (A , and B) producing a 4-bit result (X). It also produces the overflow flag V , and C_{out} .



This **subtractor** can be used to compare both *unsigned* and *signed* 4-bit input numbers (A and B) by computing $(A-B)$. It can be shown that the comparator output ($A \geq B$) is given by:

| Type of Input Operands (A & B) | Comparator Output ($A \geq B$) |
|--------------------------------------|--|
| Unsigned | = 1 iff $C_{out} = 1$ = 0 otherwise |
| Signed (2's Complement) | = 1 iff $V = \mathbf{Sign}$ of the result X = 0 Otherwise |

Using this subtractor, design a circuit that compares two 4-bit input numbers A_{3-0} and B_{3-0} to output the larger of the two. The input numbers (A & B) may be signed or unsigned. An additional input signal S indicates whether the input numbers are **signed** ($S=1$) or **unsigned** ($S=0$).

In addition to the subtractor, you may use multiplexers of any size, and other needed gates. **You MAY NOT USE** any magnitude comparator. (7 Points)

