

**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 101 (Fall 2010)**

**Major Exam 2**

**Sunday, December 26<sup>th</sup>, 2010**

**Time: 120 minutes**

Name: Key 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            | 10             |             |
| 2            | 10             |             |
| 3            | 20             |             |
| 4            | 15             |             |
| 5            | 15             |             |
| 6            | 20             |             |
| <b>Total</b> | <b>90</b>      |             |

**Question 1.**

**(10 points)**


**Fill in the Spaces:** (Show all work done to reach your answer)

a. For a 4-input function,  $F(A, B, C, D)$ , which of the following expressions represents the prime implicant having the largest area on the function's K-map?:

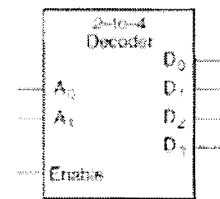
(i)  $AB+C+CD$

(ii)  $AB\bar{C}\bar{D}$

(iii)  $\bar{A}C$

b. The symbol  is an equivalent representation of the OR gate.

c. Given five 2-to-4 decoders with Enable (similar to the one shown opposite), the largest decoder we can build without any additional components is a 4-to-16 decoder.

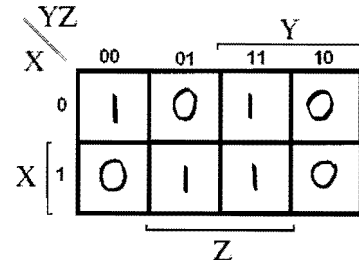


d. For  $XYZ = 100$ , the function  $F(X, Y, Z) = X \oplus Y \oplus Z$  is 1 (0/1). This function can be used to generate and detect even odd parity in digital communications.

e. Represent the function  $F(X, Y, Z)$  given below on the opposite K-map:

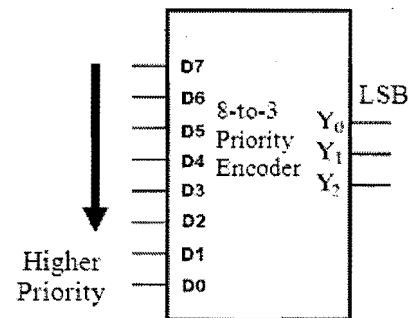
$$F(X, Y, Z) = (\bar{X} + Z) \cdot (X + Y + \bar{Z}) \cdot (X + \bar{Y} + Z)$$

$$\bar{F} = X\bar{Z} + \bar{X}\bar{Y}Z + \bar{X}Y\bar{Z}$$



f. In the priority encoder shown opposite, the output  $Y_2Y_1Y_0 = 101$ . Circle any statements that can correctly describe the status at inputs  $D_0$ - $D_7$ , with all other inputs being in a don't care condition:

- i.  $D_3 = 1$  and  $D_5 = 1$
- ii.  $D_6 = 1$  and  $D_5 = 1$
- iii.  $D_1 = 1$  and  $D_2 = 1$
- iv.  $D_0 = 1$  and  $D_2 = 1$



## Question 2.

(10 Points)

We would like to design a combinational circuit that takes as input a 3-bit unsigned binary number  $X$  and produces two binary numbers  $Y$  and  $Z$  as outputs which together represent the result of dividing the input number  $X$  by 3, where  $Y$  is the quotient and  $Z$  is the remainder.

- List in the first 3 columns in the table below all possible values for  $X$  and the corresponding values for  $Y$  and  $Z$  (all in decimal).
- The minimum number of bits needed to represent the output number  $Y$  is 2 (how many) bits and the minimum number of bits needed to represent the output number  $Z$  is 2 (how many) bits.
- In the next two columns of the table below, provide the binary truth table for the circuit, showing the bits of the inputs and outputs, with the MSB on the left in each case. Represent the outputs with the minimum number of bits obtained in (b) above. Show  $Y$  and  $Z$  bits appended together with  $Y$  at the left. Express the bit combination  $Y:Z$  in hex in the last column.

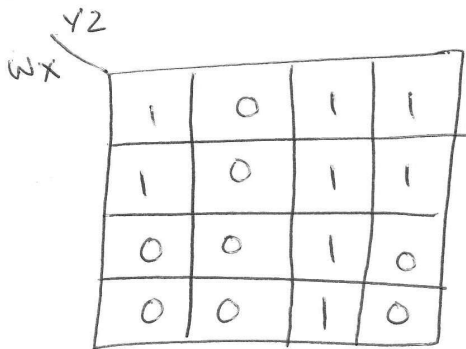
| Input X<br>(Decimal) | Output Y<br>(Decimal) | Output Z<br>(Decimal) | Input X<br>(Binary) | Output Y:Z<br>(Binary) | Output Y:Z<br>(Hex) |
|----------------------|-----------------------|-----------------------|---------------------|------------------------|---------------------|
| 0                    | 0                     | 0                     | 000                 | 0000                   | 0                   |
| 1                    | 0                     | 1                     | 001                 | 0001                   | 1                   |
| 2                    | 0                     | 2                     | 010                 | 0010                   | 2                   |
| 3                    | 1                     | 0                     | 011                 | 0100                   | 4                   |
| 4                    | 1                     | 1                     | 100                 | 0101                   | 5                   |
| 5                    | 1                     | 2                     | 101                 | 0110                   | 6                   |
| 6                    | 2                     | 0                     | 110                 | 1000                   | 8                   |
| 7                    | 2                     | 1                     | 111                 | 1001                   | 9                   |
|                      |                       |                       |                     |                        |                     |
|                      |                       |                       |                     |                        |                     |

(20 Points)

**Question 3.**

Consider the function  $F(W, X, Y, Z) = \bar{W}YZ + \bar{W}\bar{Z} + WYZ$

a. Represent the function on the K-map opposite, showing both the 1s and 0s of the function.



b. Express  $F(W, X, Y, Z) = \sum m(\underline{\hspace{2cm}})$

$$F(W, X, Y, Z) = \sum m(0, 2, 3, 4, 6, 7, 11, 15)$$

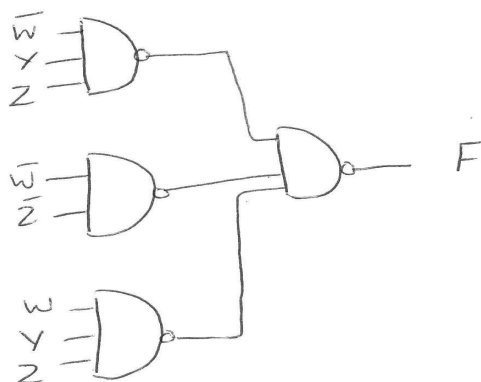
c. Express  $F(W, X, Y, Z) = \prod M(\underline{\hspace{2cm}})$

$$F(W, X, Y, Z) = \prod M(1, 5, 8, 9, 10, 12, 13, 14)$$

d. Use the K-map to minimize the expression above for F to 4 literals

$$F(W, X, Y, Z) = \bar{W}\bar{Z} + YZ$$

e. Draw the logic diagram for a 2-level implementation of  $F(W, X, Y, Z) = \bar{W}YZ + \bar{W}\bar{Z} + WYZ$  using only NAND gates. Assume that both the true and complemented forms of each variable are readily available.



**Question 4.****(15 Points)**Given the following K-map for a function  $F$ :

(a) List all Essential Prime Implicants

$$EPI: \{ AC, \bar{A}B\bar{C}, A\bar{B}D \}$$

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

(b) Provide a minimized **SOP** expression of  $F$ 

$$F(A, B, C, D) = AC + \bar{A}B\bar{C} + A\bar{B}D + \begin{cases} B\bar{C}\bar{D} \\ A\bar{B}\bar{D} \end{cases}$$

(c) Provide a minimized **POS** expression of  $F$ 

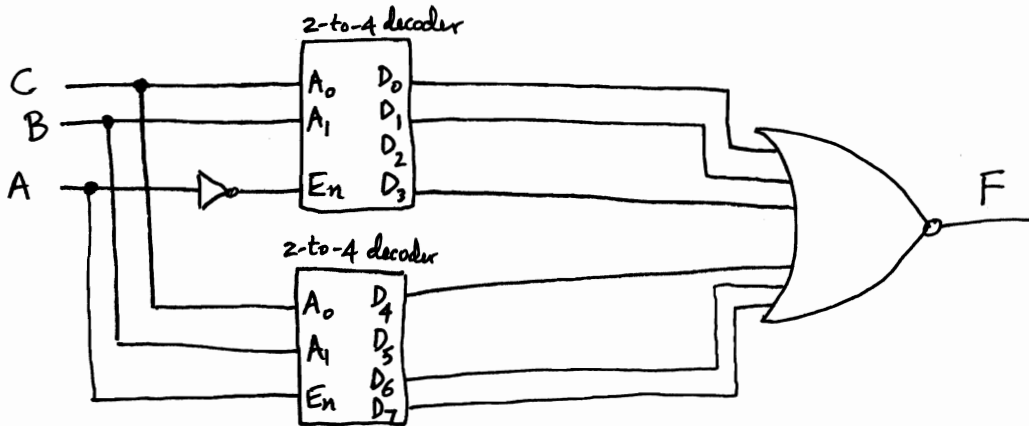
$$F(A, B, C, D) = (A+B)(A+\bar{C})(\bar{A}+\bar{B}+C+\bar{D})(B+C+D)$$

Question 5. Given the function

(15 Points)

$$F(A,B,C) = \Pi M(0,1,3,4,6,7)$$

a. Implement F using two 2-to-4 decoders (each has an enable  $E_n$  input), an inverter and a NOR.



b. Implement F using the 4-to-1 MUX shown below. Show how you obtained your solution.

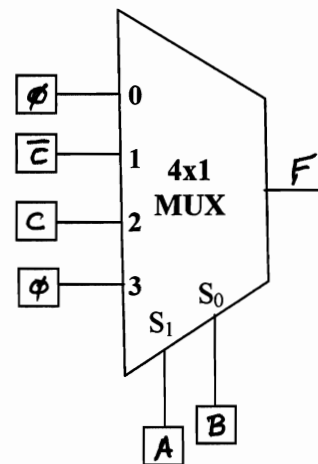
| A | B | C | F |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

$$I_0 = \phi$$

$$I_1 = \bar{C}$$

$$I_2 = C$$

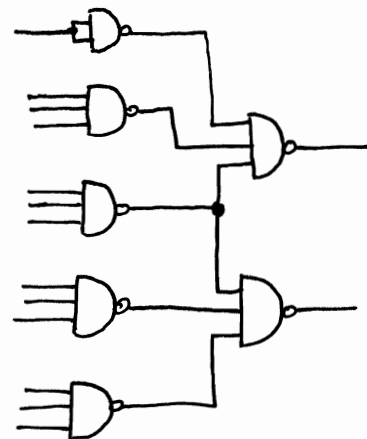
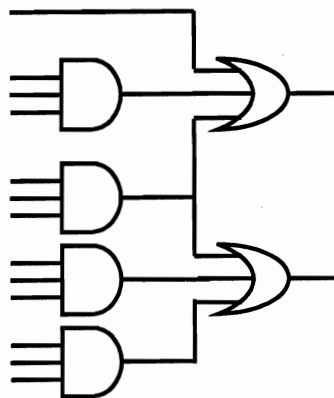
$$I_3 = \phi$$



OR

|           | $\bar{A}\bar{B}$ | $\bar{A}B$ | $A\bar{B}$ | $AB$   |
|-----------|------------------|------------|------------|--------|
| $\bar{C}$ | 0                | ②          | 4          | 6      |
| C         | 1                | 3          | ⑤          | 7      |
|           | $\phi$           | $\bar{C}$  | C          | $\phi$ |

c. Convert the following logic circuit to NAND-only:



**Question 6.**

**(20 Points)**

Use any number of the following MSI components: **inverters, decoders, encoders, multiplexers, adders, and/or magnitude comparators**, to design a circuit that takes two 4-bit binary numbers  $A = A_3A_2A_1A_0$  and  $B = B_3B_2B_1B_0$  and a 2-bit user selection input  $S = S_1S_0$ . The circuit should produce a 5-bit output  $O = O_4O_3O_2O_1O_0$  according to the following table:

| $S_1S_0$ | $O$ is equal to    |
|----------|--------------------|
| 00       | $\text{Max}(A, B)$ |
| 01       | $\text{Min}(A, B)$ |
| 10       | $2 \times A$       |
| 11       | $A - B$            |

Note that you must clearly label the MSI part used together with all inputs and outputs.

