King Fahd University of Petroleum and Minerals
Department of Computer Engineering

DIGITAL LOGIC DESIGN COE 202

Exam 2, December 27, 2008

Problems	Grading
1	
2	
3	
TOTAL	

Student Name:……

Student ID:……

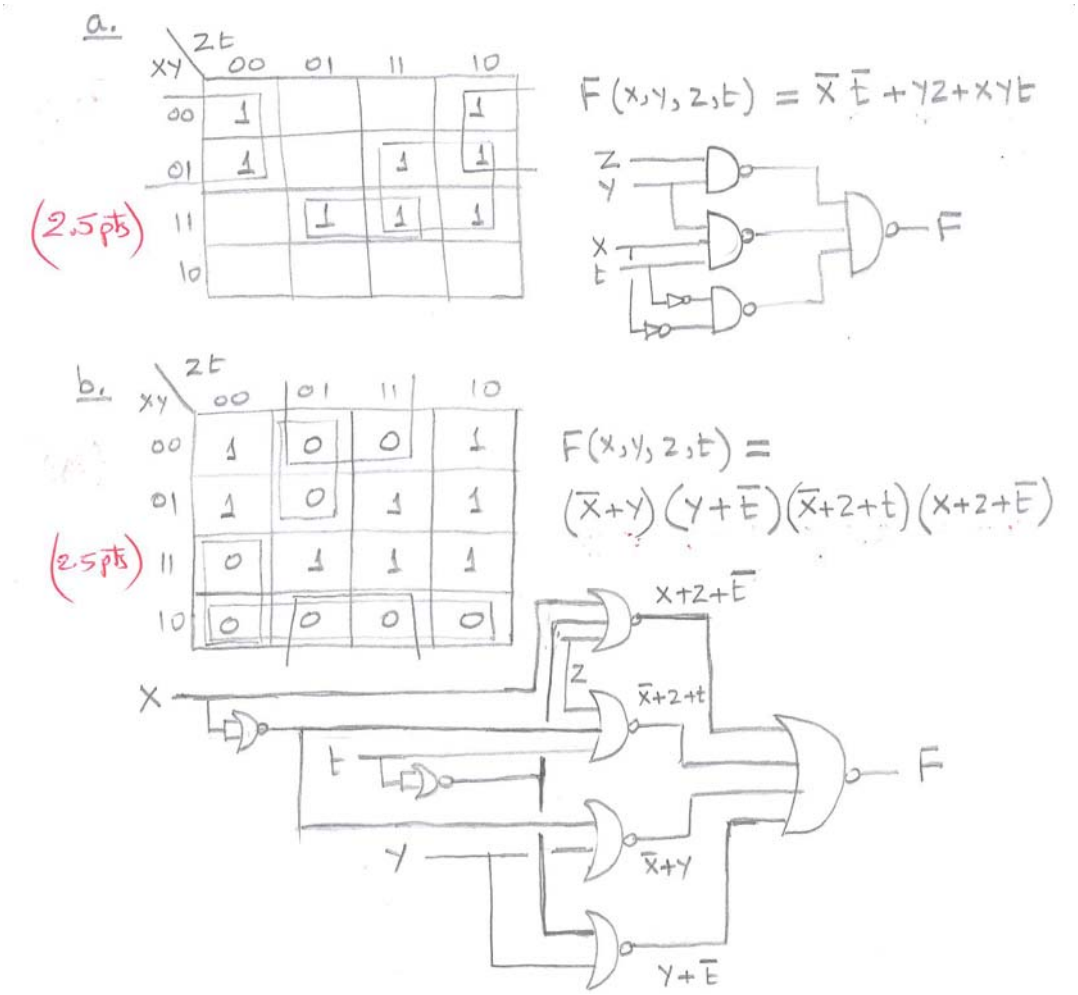
Problem-1: Optimizing combinational logic functions and their implementation

Consider the combinational logic function F(x, y, z, t) = Σ (0, 2, 4, 6, 7, 13, 14, 15) which is expressed as a sum of minterms. Answer each of the following questions:

a. Express F(x, y, z, t) as a minimal sum of products (SOP) and implement it using only NAND gates.

b. Express F(x, y, z, t) as a minimal product of sums (POS) and implement it using only NOR gates.

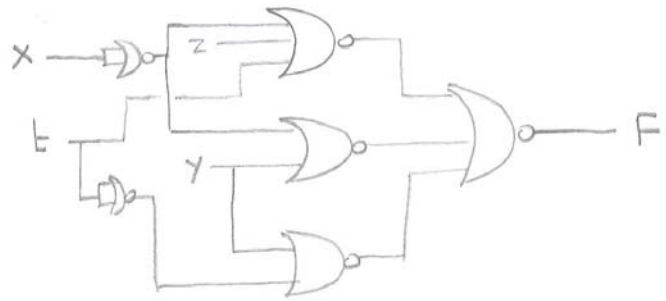
c. Suppose that combinations (x, y, z, t) = 1, 5, 10, and 11 cannot occur and can be considered as don't care conditions. Express the above function F(x, y, z, t) as a minimal product of sums (POS) and implement it using only NOR gates.



C. (3pts)

XY \ ZT	00	01	11	10
00	1	X	0	1
01	1	X	1	1
11	0	1	1	1
10	0	0	X	X

$F(x,y,z,t) = (\overline{x}+z+t)(\overline{x}+y)(y+\overline{t})$

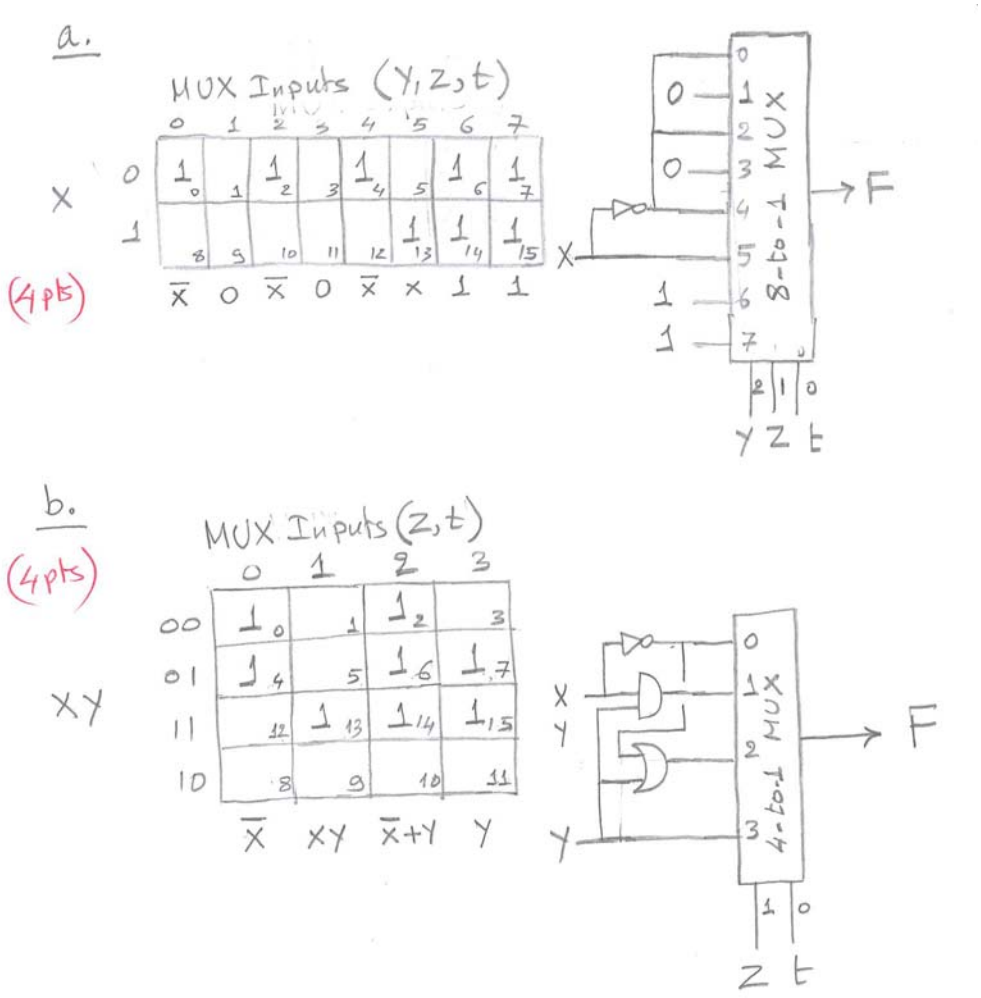


Problem-2: Implementing combinational logic functions using multiplexers.

Consider the combinational logic function F(x, y, z, t) = Σ (0, 2, 4, 6, 7, 13, 14, 15) which is expressed as a sum of minterms. We assume 8-to-1 Multiplexers each has (1) three control lines X, Y, and Z, (2) eight inputs labelled a_0, a_1, a_2, a_3, a_4, a_5, a_6, and a_7, and (3) one output B. Answer each of the following questions:

a. Implement function F(x, y, z, t) using **ONLY** an 8-to-1 Multiplexer and some additional logic.

b. Implement function F(x, y, z, t) using **ONLY** a 4-to-1 Multiplexer and some additional logic.



Problem-3: Design of a 12-bit parallel adder using Carry-Look-Ahead adders

It is desired to design a fast 12-bit parallel binary adder based on the use of the half-adder (HA) and Full-Adder (FA). The basic designs of HA and FA are as follows:

- The HA has inputs A and B and output S = A xor B and C = A.B
- The FA has inputs A, B, and C_{in} and output S = A xor B xor C_{in} and C_{out} = AB + (A xor B) C_{in} .

It is assumed that each logic gate incurs a fixed delay of T seconds.

For each of the following questions you are requested to show your steps and provide your justification for each result:

a. Determine the total delay through the above HA and FA.

b. Determine the total delay through a 4-bit Carry-Ripple Adder designed using the above HA and FA.

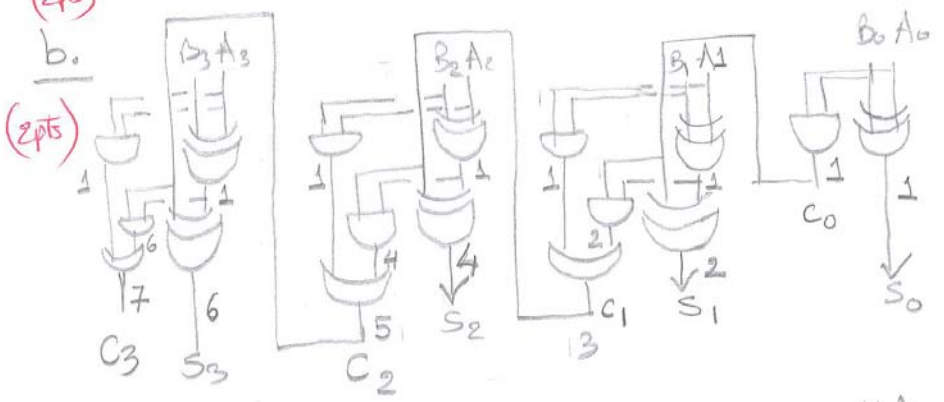
c. How to modify the 4-bit Carry-Ripple Adder made using the above HA and FA to a 4-bit Carry-look-ahead adder!

d. Design a 4-bit Carry-look-ahead Adder and determine its total delay.

e. Design a 12-bit parallel adder using the previously designed 4-bit bit Carry-look-ahead Adder and determine its total delay.

a. Delay through HA is 1T and through FA is 3T. (2pts)

b. (2pts)



Total delay = { 7T if we first use an HA.
9T " " FA.

C. Denote by $P_i = A_i \cdot B_i$ and $E_i = A_i \oplus B_i$

(2pts) $S_0 = E_0 \oplus C_{in}$

$C_0 = P_0 + E_0 \cdot C_{in}$

$S_1 = E_1 \oplus C_0 = E_1 \oplus P_0 + E_1 \oplus E_0 \cdot C_{in}$

$C_1 = P_1 + E_1 \cdot C_0 = P_1 + E_1 P_0 + E_1 E_0 \cdot C_{in}$

$C_2 = P_2 + E_2 \cdot C_1 = P_2 + E_2 P_1 + E_2 E_1 P_0 + E_2 E_1 E_0 \cdot C_{in}$

$C_3 = P_3 + E_3 C_2 = P_3 + E_3 E_2 P_1 + E_3 E_2 E_1 P_0 + E_3 E_2 E_1 E_0 C_{in}$

Since C_3 is the longest path, then we need

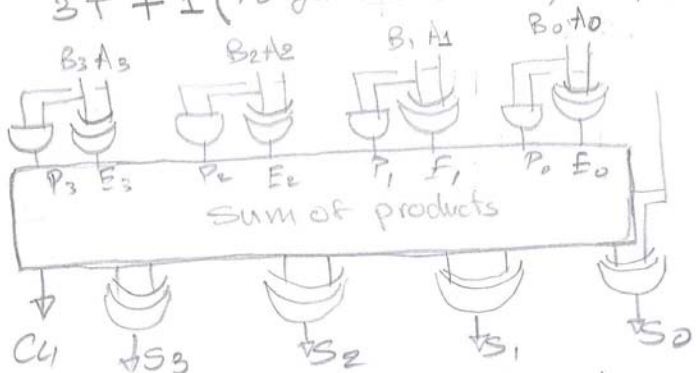
1T For all P_i and E_i terms

1T For all product terms in C_i $(0 \le i \le 3)$

1T For all sum of products " "

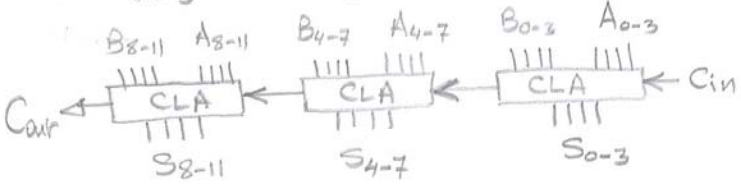
3T + 1 (To get the S bits) = 4T

d. (2pts)



Total delay is 4T

e. (1 pt.)



Total delay is $4T \times 3 = 12T$.