Oct. 20, 2011

COMPUTER ENGINEERING DEPARTMENT

COE 561

Digital System Design and Synthesis

Major Exam I

(Open Book Exam)

First Semester (111)

Time: 2:00-4:30 PM

Student Name : ____

Student ID. : ____

Question	Max Points	Score
Q1	10	
Q2	10	
Q3	10	
Q4	20	
Q5	30	
Q6	20	
Total	100	

[10 Points]

(Q1)

(i) Represent the cover $F(A,B,C) = \overline{A}\overline{B}\overline{C} + A\overline{B}C$, using positional cubical notation.

(ii) Compute the **complement** of the cover *F* using **sharp** operation.

(i)

	A	B	C
$\overline{A}\overline{B}\overline{C}$	10	10	10
$A\overline{B}C$	01	10	01

(ii) We first compute 11 11 11 # 10 10 10

This produces the following:

01 11 11
11 01 11
11 11 01

Next, we compute the result # 01 10 01

1. 01 11 11 # 01 10 01

$\Rightarrow$ 01 01 11
01 11 10

2. 11 01 11 # 01 10 01

$\Rightarrow$ 10 01 11 $\Rightarrow$ 11 01 11 as other cubes are covered by this cube
11 01 11
11 01 10

3. 11 11 01 # 01 10 01

$\Rightarrow$ 10 11 01
11 01 01

Next, we perform a union of the results and we eliminate cubes covered by other cubes based on single cube containment and we get the following:

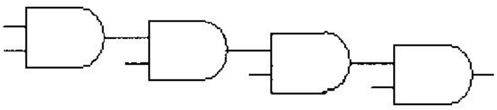
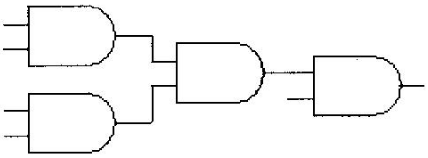
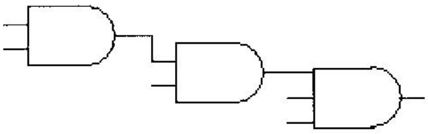
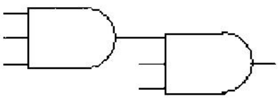
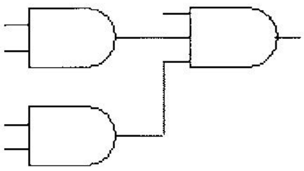
01	11	10
11	01	11
10	11	01

Thus, the complement of $F = a\bar{c} + b + \bar{a}c$

[10 Points]

(Q2) Consider the function $F = ABCDE$ and the set of implementations given below.

Assume that the area and delay of a gate are directly related to the number of its inputs. Compute the **area** and **delay** cost for each implementation and determine the **Pareto optimal points**.

Implementation	Area	Delay
	8	8
	8	6
	7	7
	6	6
	7	5

The pareto optimal points are based on the 4th and 5th implementations with cost (6,6) and (7,5).

[10 Points]

(Q3) Consider the following function:
$F(A,B,C,D) = AD + BC + A\overline{C} + \overline{B}\overline{D} + \overline{C}D + \overline{B}CD + B\overline{C}\overline{D}$.

Using recursive paradigm, determine if the function F is **Tautology** or not. You need to choose the right variable for expansion to minimize computations.

Since the cover is positive unate with respect to A, it is sufficient to show that $F_{\overline{A}}$ is tautology since $F_A \supseteq F_{\overline{A}}$,

Thus, $F_{\overline{A}} = BC + \overline{B}\overline{D} + \overline{C}D + \overline{B}CD + B\overline{C}\overline{D}$

$= \overline{B}\,[\,\overline{D} + \overline{C}D + CD\,]$
$+ B\,[\,C + \overline{C}D + \overline{C}\overline{D}\,]$

$= \overline{B}\,[\,\overline{D}\,[1] + D\,[\underbrace{\overline{C} + C}_{1}]\,]$

$+ B\,[\,\overline{C}\,[\underbrace{D + \overline{D}}_{1}] + C\,[1]\,]$

Thus, $F_{\overline{A}} = 1 \Rightarrow F_A$ is Tautology.

[20 Points]

(Q4) Consider the two Boolean functions F_1 and F_2 given below:

$$F_1(A,B) = A \oplus B$$

$$F_2(C,D) = C \oplus D$$

Draw the **ITE DAG** for the function $F_1.F_2$ using the variable order {A, B, C, D}. Show all the details of your solution using ITE procedure including the resulting unique table and computed table.

$F_1 \cdot F_2 = ITE(A \oplus B, C \oplus D, 0)$

- $x = A$

  $t = ITE(\overline{B}, C \oplus D, 0)$

  - $x = B$

    $t = ITE(0, C \oplus D, 0) = 0$ (trivial case)

    $\Rightarrow t = 1$

    $e = ITE(1, C \oplus D, 0)$

    - $x = C$

      $t = ITE(1, \overline{D}, 0) = \overline{D}$ (trivial case)

      we assign id $= 3 \Rightarrow t = 3$

      $e = ITE(1, D, 0) = D$ (trivial case)

      we assign id $= 4 \Rightarrow e = 4$

      since $t \neq e$, we add the entry $(C, 3, 4)$ in the unique table with id $= 5$.

      we add an entry in the computed table with $\{(1, C \oplus D, 0), 5\}$.

since $t \neq e$, we add the entry $(B, 1, 5)$ in the unique table with $id = 6$.
we add an entry in the computed table with
$\{(\overline{B}, C \oplus D, 0), 6\}$.

$e = ITE(B, C \oplus D, 0)$

- $x = B$

$t = ITE(1, C \oplus D, 0) = 5$ from computed table

$e = ITE(0, C \oplus D, 0) = 0$ (trivial case)

$\Rightarrow e = 1$

since $t \neq e$, we add the entry $(B, 5, 1)$ in the unique table with $id = 7$.
we add an entry in the computed table with
$\{(B, C \oplus D, 0), 7\}$.

since $t \neq e$, we add the entry $(A, 6, 7)$ in the unique table with $id = 8$.
we add an entry in the computed table with
$\{(A \oplus B, C \oplus D, 0), 8\}$.

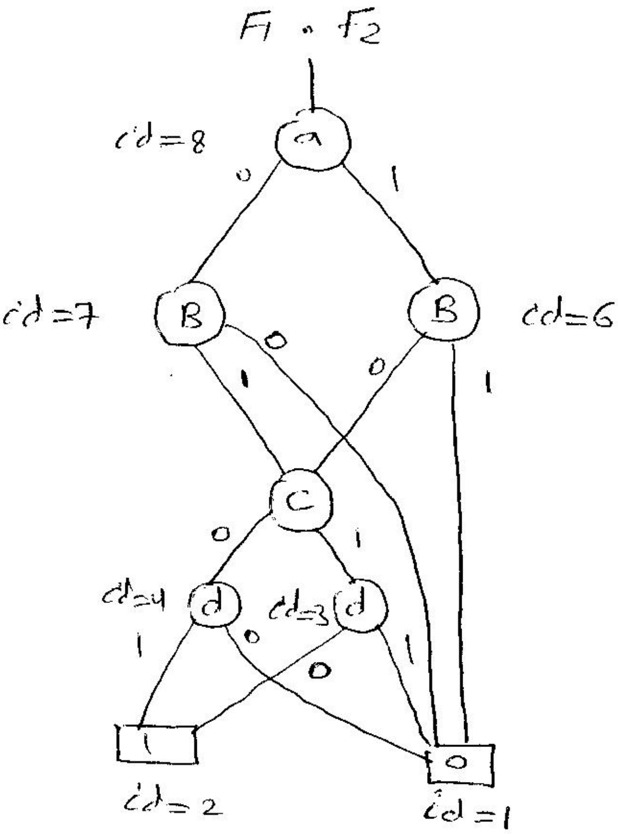
unique Table:

id	var	H	L
5	C	3	4
6	B	1	5
7	B	5	1
8	A	6	7

Computed Table:

f	g	h	id
1	$c \oplus d$	0	5
$\overline{B}$	$c \oplus d$	0	6
B	$c \oplus d$	0	7
$A \oplus B$	$c \oplus d$	0	8

ITE DAG



[30 Points]

(Q5) Consider the function $F(A,B,C,D) = BD + A\overline{C}\,\overline{D} + \overline{A}\,\overline{B}C + \overline{A}\,\overline{B}\,\overline{D} + A\overline{B}\,\overline{D}$

(i) Compute the **complement** of the function using the recursive complementation procedure outlined in section 7.3.4.

(ii) **C**ompute all the **prime implicants** of the function using the method outlined in section 7.3.4.

(i) $F = \overline{A}\,[\,BD + \overline{B}C + \overline{B}\,\overline{D}\,]$

$+ A\,[\,BD + \overline{C}\,\overline{D} + \overline{B}\,\overline{D}\,]$

$= \overline{A}\,[\,\overline{B}\,[\,C + \overline{D}\,] + B\,[\,D\,]\,]$

$+ A\,[\,\overline{D}\,[\,\overline{C} + \overline{B}\,] + D\,[\,B\,]\,]$

$= \overline{A}\,[\,\overline{B}\,[\,\overline{C}\,[\,\overline{D}\,] + C\,[\,1\,]\,] + B\,[\,D\,]\,]$

$+ A\,[\,\overline{D}\,[\,\overline{C}\,[\,1\,] + C\,[\,\overline{B}\,]\,] + D\,[\,B\,]\,]$

$\Rightarrow \overline{F} = \overline{A}\,[\,\overline{B}\,[\,\overline{C}\,[\,D\,] + C\,[\,0\,]\,] + B\,[\,\overline{D}\,]\,]$

$+ A\,[\,\overline{D}\,[\,\overline{C}\,[\,0\,] + C\,[\,B\,]\,] + D\,[\,\overline{B}\,]\,]$

$= \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}B\overline{D} + ABC\overline{D} + A\overline{B}D$

(ii) From part (i), we have

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

Prime implicants of $F_{\bar{A}} = scc\{\bar{B}C, \bar{B}\bar{D}, BD, CD\}$

$$= \{\bar{B}C, \bar{B}\bar{D}, BD, CD\}$$

Prime implicants of $F_A = scc\{\bar{B}\bar{D}, \bar{C}\bar{D}, BD, B\bar{C}\}$

$$= \{\bar{B}\bar{D}, \bar{C}\bar{D}, BD, B\bar{C}\}$$

Prime implicants of $F = scc\{\bar{A}\bar{B}C, \bar{A}\bar{B}\bar{D}, \bar{A}BD, \bar{A}CD, A\bar{B}\bar{D}, A\bar{C}\bar{D}, ABD, AB\bar{C}, \bar{B}C\bar{D}, \bar{B}\bar{D}, \bar{B}\bar{C}\bar{D}, BD, B\bar{C}D, BCD\}$

$$= \{\bar{A}\bar{B}C, \bar{A}CD, A\bar{C}\bar{D}, AB\bar{C}, \bar{B}\bar{D}, BD\}$$

[20 Points]

(Q6) Consider the following given matrix representing a covering problem:

$$
\begin{array}{c}
\\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} c_1 & c_2 & c_3 & c_4 & c_5 & c_6 & c_7 & c_8 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 & 1
\end{bmatrix}
\end{array}
$$

Find a **minimum cover** using **EXACT_COVER** procedure. Show all the details of the algorithm. Assume the following order in branching selection when needed: C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8.

There are no essential columns.

c1 dominates c8 ⇒ c8 is removed.

r6 dominates r3 ⇒ r6 is removed.

r7 dominates r5 ⇒ r7 is removed.

Thus, we select c1 and call exact cover with $x = (1,0,0,0,0,0,0,0)$ and $b = (1,1,1,1,1,1,1)$ and the matrix:

	c2	c3	c4	c5	c6	c7
r1	1	0	1	0	1	0
r2	0	1	0	1	0	1

c2 dominates c4 & c6 ⇒ c4 & c6 are removed

c3 dominates c5 & c7 ⇒ c5 & c7 are removed

Thus, c2 and c3 become essential and are selected.

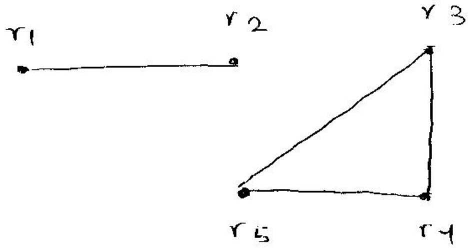
Since the matrix has no rows then the returned solution is $x = (1,1,1,0,0,0,0,0)$ and $b = (1,1,1,0,0,0,0,0)$.

Next, exact-cover is called with c1 not selected with $x = (0,0,0,0,0,0,0,0)$ and $b = (1,1,1,0,0,0,0,0)$, and the matrix:

	c2	c3	c4	c5	c6	c7
r1	1	0	1	0	1	0
r2	0	1	0	1	0	1
r3	1	1	0	0	0	0
r4	0	0	1	1	0	0
r5	0	0	0	0	1	1

There are no essential columns, no row dominance and no column dominance.

We compute the lower bound as follows:



Since the clique number is 3, the lower bound is 3.

Since the current estimate = |b|, return b.

Since the returned solution is not < |b|,

return $b = (1,1,1,0,0,0,0,0)$.