Nov. 9, 2009

COMPUTER ENGINEERING DEPARTMENT

COE 561

Digital System Design and Synthesis

Major Exam I

(Open Book Exam)

First Semester (091)

Time: 8:00-10:30 PM

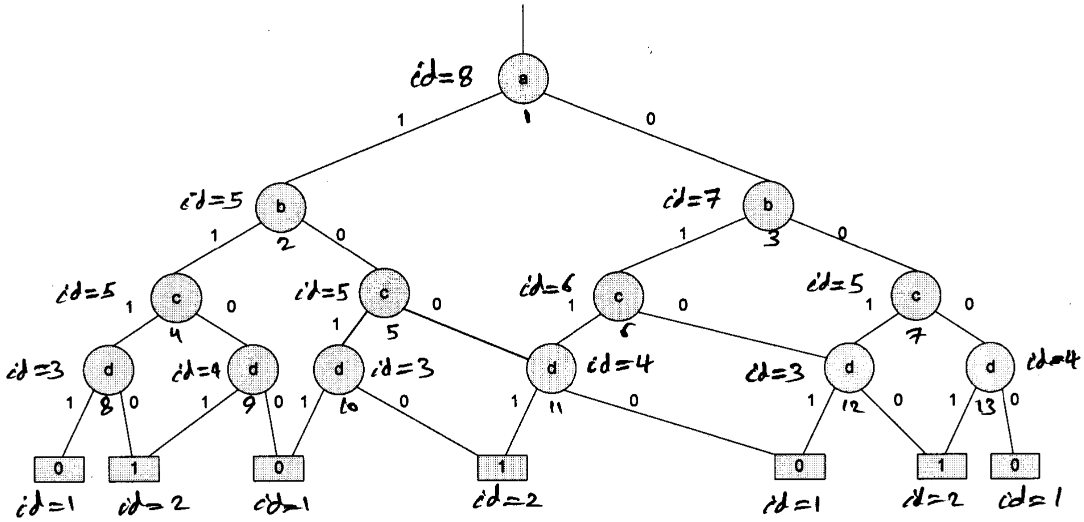
Student Name : _KEY____

Student ID. : ____

Question	Max Points	Score
Q1	15	
Q2	10	
Q3	10	
Q4	25	
Q5	20	
Q6	20	
Total	100	

[15 Points]

(Q1) Consider the OBDD given below with the variable ordering {a, b, c, d}. Apply the **Reduce** function to obtain the **ROBDD**. Show the details of applying the algorithm step by step.



First, we set id(v) = 1 for all leaf vertices with value 0 and id(v) = 2 for leaf vertices with value 1.
We initialize ROBDD with two leaf vertices for 0 and 1.
Then, we process vertices at level 4, i.e. nodes with index = d. V = {8, 9, 10, 11, 12, 13}. oldKey = (0,0).
None of the vertices is removed since id(low(v)) ≠ id(high(v)). We assign keys to all vertices ∈ V.
Key(8) = (2,1), Key(9) = (1,2), Key(10) = (2,1), Key(11) = (1,2), Key(12) = (2,1), Key(13) = (1,2).
We next sort the vertices in V according to their key.
Thus, V = {8, 10, 12, 9, 11, 13}.

v = {8}: since Key(8) ≠ oldKey, nextid = 3, id(8) = 3, oldKey = (2,1). We add v = {8} to the ROBDD.

v = {10}: since Key(10) = oldKey, id(10) = 3.

v = {12}: since Key(12) = oldKey, id(12) = 3.

v = {9}: since Key(9) ≠ oldKey, nextid = 4, id(9) = 4, oldKey = (1,2). We add v = {9} to the ROBDD.

v = {11}: since Key(11) = oldKey, id(11) = 4.

v = {12}: since Key(12) = oldKey, id(12) = 4.

Next, we process vertices at level 3 with index = c.

$V = \{4, 5, 6, 7\}$.

None of the vertices is removed since $id(low(v)) \neq id(high(v))$.

We assign keys to all vertices $\in V$.

$key(4) = (4,3)$, $key(5) = (4,3)$, $key(6) = (3,4)$, $key(7) = (4,3)$.

We next sort the vertices in V according to their key.

Thus, $V = \{4,5,7,6\}$. oldkey = (0,0).

$v = \{4\}$: since $key(4) \neq oldkey$, nextid = **5**, $id(4) = 5$, oldkey = (4,3). We add $v = \{4\}$ to the ROBDD.

$v = \{5\}$: since $key(5) = oldkey$, $id(5) = 5$.

$v = \{7\}$: since $key(7) = oldkey$, $id(7) = 5$.

$v = \{6\}$: since $key(6) \neq oldkey$, nextid = 6, $id(6) = 6$, oldkey = (3,4). We add $V = \{6\}$ to the ROBDD.

Next, we process vertices at level 2 with index = b.

$V = \{2,3\}$. Since $id(low(2)) = id(high(2))$, $id(2) = 5$ and vertex 2 is removed from V.

$key(3) = (5,6)$, oldkey = (0,0).

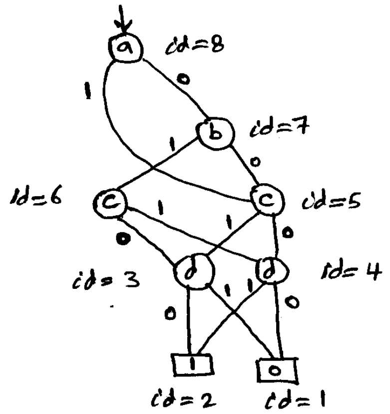
Since $key(3) \neq oldkey$, nextid = 7, $id(3) = 7$ and we add $v = \{3\}$ to the ROBDD.

Finally, we process vertices at level 1 with index = a.

$V = \{1\}$. Since $id(low(1)) \neq id(high(1))$, the vertex is not removed. $key(1) = (7,6)$, oldkey = (0,0).

Since $key(1) \neq oldkey$, nextid = 8, $id(1) = 8$ and we add $v = \{1\}$ to the ROBDD.

Thus, ROBDD :



[10 Points]

(Q2) Perform the following operations using positional cube notation:

(i) Cofactor of $A\bar{B}C\bar{D} + \bar{A}\bar{B}\bar{D} + CD$ with respect to $A\bar{B}$.

(ii) $\bar{B} \# A\bar{B}C$.

(iii) Consensus of BD and $A\bar{B}C$.

(i)

	A	B	C	D
$A\bar{B}C\bar{D}$	01	10	01	10
$\bar{A}\bar{B}\bar{D}$	10	10	11	10
CD	11	11	01	01
$A\bar{B}$	01	10	11	11
$C\bar{D}$	11	11	01	10

void since distance is 1

CD	11	11	01	01

Thus, the result of cofactor with respect to $A\bar{B}$ is $C\bar{D} + CD = C$

(ii) $\bar{B} \# A\bar{B}C$

	A	B	C	
$\bar{B}$	11	10	11	
$A\bar{B}C$	01	10	01	
$\bar{A}\bar{B}$	10	10	11	
	11	00	11	void
$\bar{B}\bar{C}$	11	10	10	

Thus, the result of sharp operation is $\bar{A}\bar{B} + \bar{B}\bar{C}$

(iii)

	A	B	C	D	
BD	11	01	11	01	
$A\bar{B}C$	01	10	01	11	
	11	00	01	01	void
ACD	01	11	01	01	
	01	00	11	01	void
	01	00	01	11	void

Thus, the result of consensus is ACD.

[10 Points]

(Q3) Consider the function $F(A,B,C,D) = \overline{A}\overline{B} + D + \overline{C} + \overline{B}C + \overline{A}C + B$. 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 F is negative unate with respect to A, $F_{\overline{A}} \supseteq F_A$, it is sufficient to check that F_A is tautology.

$F_A = D + \overline{C} + \overline{B}C + B$

Since F_A is positive unate with respect to D, $F_D \supseteq F_{\overline{D}}$, it is sufficient to check that $(F_A)_{\overline{D}}$ is tautology.

$(F_A)_{\overline{D}} = \overline{C} + \overline{B}C + B$

Next, we can choose any variable for expansion. Let us expand on C.

$F_{A\overline{D}\overline{C}} = 1 \Rightarrow$ Tautology

$F_{A\overline{D}C} = \overline{B} + B \Rightarrow$ Tautology

Thus, F is Tautology.

[25 Points]

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

$$F_1(A,B,C,D) = \overline{A}\overline{B}D + A\overline{C}D + A\overline{B}C$$

$$F_2(A,B,C,D) = \overline{B}\overline{C}\overline{D} + \overline{A}\overline{B}\overline{D} + BD$$

(i) Compute the expansion of F_1 and F_2 using the **Orthonormal Basis** {$\varnothing_1 = \overline{A}\overline{B}$, $\varnothing_2 = \overline{A}B$, $\varnothing_3 = A\overline{B}$, $\varnothing_4 = AB$ }.

(ii) Compute the function $F_1 + F_2$.

(iii) Draw the **ITE DAG** for the function $F_1.F_2$ using the variable order {A, B, C, D}. Use the given functions as is and do not start with the minimized result of $F_1.F_2$. Show all the details of your solution using ITE procedure including the resulting unique table.

(i) $F1 = \bar{A}\bar{B}\,[D] + \bar{A}B\,[0] + A\bar{B}\,[\bar{C}D + C] + AB\,[\bar{C}D]$

$F2 = \bar{A}\bar{B}\,[\bar{C}\bar{D} + \bar{D}] + \bar{A}B\,[D] + A\bar{B}\,[\bar{C}\bar{D}] + AB\,[D]$

(ii) $F1+F2 = \bar{A}\bar{B}\,[1] + \bar{A}B\,[D] + A\bar{B}\,[\bar{C}D + C + \bar{C}\bar{D}]$
$+ AB\,[\bar{C}D + D]$
$= \bar{A}\bar{B}\,[1] + \bar{A}B\,[D] + A\bar{B}\,[1] + AB\,[D]$

(iii) $F1.F2 = ITE(F1, F2, 0)$
$= ITE(\bar{A}\bar{B}D + A\bar{C}D + A\bar{B}C,\ \bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{D} + BD, 0)$

- $x = a$
  $t = ITE(\bar{C}D + \bar{B}C,\ \bar{B}\bar{C}\bar{D} + BD, 0)$
  - $x = b$
    $t = ITE(\bar{C}D, D, 0)$
    - $x = c$
      $t = (0, D, 0) = 0 \Rightarrow t = 1$ (trivial case)
      $e = (D, D, 0) = D \Rightarrow e = 3$ (trivial case)
      since $t \neq e$, an entry will be added in the table for $(c, 1, 3)$ with id = 4
    $\Rightarrow t = 4$

$e = ITE(\bar{c}D + c,\ \bar{c}\bar{D},\ 0)$

- $x = c$

  $t = (1, 0, 0) = 0 \Rightarrow t = 1$ (trivial case)

  $e = (D, \bar{D}, 0) = 0 \Rightarrow e = 1$ (trivial case)

  Since $t \neq e \Rightarrow e = 1$

Since $t \neq e$, an entry will be added in the table for $(b, 4, 1)$ with $id = 5$.

$\Rightarrow t = 5$

$e = ITE(\bar{B}D,\ \bar{B}\bar{C}\bar{D} + \bar{B}\bar{D} + BD,\ 0)$

- $x = b$

  $t = ITE(0, D, 0) = 0 \Rightarrow t = 1$ (trivial case)

  $e = ITE(D,\ \bar{C}\bar{D} + \bar{D},\ 0)$

  - $x = c$

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

    $e = ITE(D, \bar{D}, 0) = 0 \Rightarrow e = 1$ (trivial case)

    since $t = e \Rightarrow e = 1$

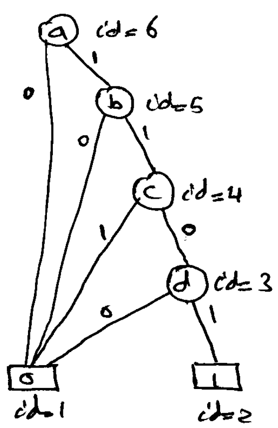
  since $t = e \Rightarrow e = 1$

since $t \neq e$, an entry will be added in the table for $(a, 5, 1)$ with $id = 6$.

Produced unique table:

id	var	right (1)	left (0)
3	d	2	1
4	c	1	3
5	b	4	1
6	a	5	1

ITE DAG



[20 Points]

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

- **(i)** Compute the **complement** of the function using the recursive complementation procedure outlined in section 7.3.4. You need to choose the right variable for expansion to minimize computations.
- **(ii)** Compute all the **prime implicants** of the function using the method outlined in section 7.3.4. You need to choose the right variable for expansion to minimize computations.

(i) We will expand on unate variables

$$\Rightarrow F_B + \bar{B} F_{\bar{B}}$$

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

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

$$= [\,[\bar{A}\,[1] + A\,[\underbrace{\bar{C} + C}_{1}]] + \bar{D}\,[\underbrace{[\bar{A}+A]}_{1} + \bar{C}[0]]\,] + \bar{B}\,[A\bar{C}D]$$

$$\Rightarrow \bar{F} = [\,[\bar{A}\,[0] + A\,[0]] + \bar{D}\,[\bar{C}\,[1]]\,] + \bar{B}\,[\bar{A} + C + \bar{D}]$$

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

(ii)
$$F = \bar{A}\,[BC + BD] + A\,[\bar{C}D + BC]$$

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

Prime implicants of f_A $= scc\{\bar{C}D, CB, DB\}$
$= \{\bar{C}D, CB, DB\}$

Prime implicants of $f_{\bar{A}}$ $= \{BC, BD\}$

Prime implicants of f $= scc\{A\bar{C}D, ACB, ADB,$
$\bar{A}BC, \bar{A}BD, BC, BCD, B\bar{C}D, BCD, BD\}$
$= \{A\bar{C}D, BC, BD\}$

[20 Points]

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

$$\begin{array}{c} \\ r1 \\ r2 \\ r3 \\ r4 \\ r5 \\ r6 \\ r7 \end{array} \begin{array}{c} \begin{array}{cccccccc} c1 & c2 & c3 & c4 & c5 & c6 & c7 & c8 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

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.

c7 is essential. Thus, it is selected and r5 & r6 are removed.

c5 dominates c8, thus c8 is removed.

r7 dominates r1, thus r7 is removed.

Thus, the reduced matrix is:

	c1	c2	c3	c4	c5	c6
r1	1	0	1	0	1	0
r2	1	0	0	1	0	1
r3	0	1	1	0	0	1
r4	0	1	0	1	1	0

Next, we select c1 and call exact cover with the matrix:

	c2	c3	c4	c5	c6
r3	1	1	0	0	1
r4	1	0	1	1	0

$x = (1,0,0,0,0,0,1,0)$

$b = (1,1,1,1,1,1,1,1)$

c2 dominates all columns and becomes essential and is selected.

Thus, $x = (1,1,0,0,0,0,1,0)$

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

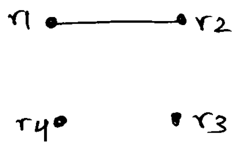
Next, Exact-Cover is called with c1 not selected with the following matrix:

$$\begin{array}{c|ccccc} & c2 & c3 & c4 & c5 & c6 \\ \hline r_1 & 0 & 1 & 0 & 1 & 0 \\ r2 & 0 & 0 & 1 & 0 & 1 \\ r3 & 1 & 1 & 0 & 0 & 1 \\ r4 & 1 & 0 & 1 & 1 & 0 \end{array}$$

$x = (0,0,0,0,0,0,1,0)$

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

The matrix can't be reduced. Thus, we compute the lower bound.



The lower bound $= 1+2 = 3$

Since the lower bound is equal to the best solution, the best solution is returned.