Jan. 20, 2006

COMPUTER ENGINEERING DEPARTMENT

COE 561

Digital System Design and Synthesis

Final Exam

(Open Book Exam)

First Semester (061)

Time: 7:00-10:00 PM

Student Name : _____

Student ID. : _____

Question	Max Points	Score
Q1	25	
Q2	25	
Q3	10	
Q4	40	
Total	100	

[25 Points]

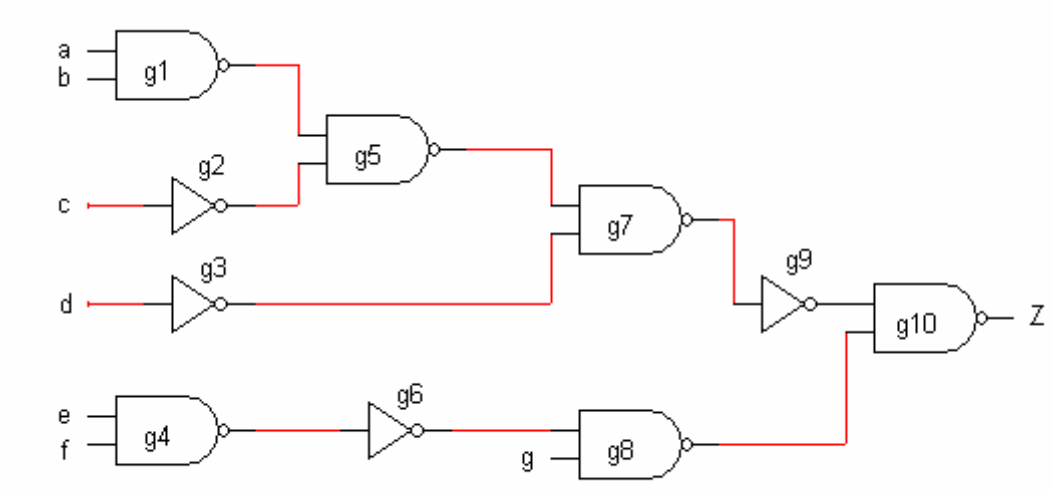
(Q1) Consider a technology library containing the following cells:

Cell	Area Cost
INV(x1)=x1’	1
NAND2(x1,x2)=(x1 x2)’	2
NAND3(x1m x2, x3) = (x1 x2 x3)’	3
NOR2(x1,x2)=(x1 + x2)’	2
AOI21(x1, x2, x3) = ((x1 x2) + x3)’	3
OAI21(x1, x2, x3) = ((x1+x2) x3)’	3

(i) Show the **pattern trees** of the library cells using **NAND2** and **INV** as base functions. Assume that symmetric representations do not need to be stored.

(ii) Decompose the function f = a + b’ + c using NAND2 and INV as base functions into **all** possible **non-symmetric decomposition**s. Then, **map** the decomposed circuits using the given library and determine the decomposition that leads to a **lower area cost**.

(iii) Using the dynamic programming approach, **map** the circuit given below using the given library into the **minimum area** cost solution. Inputs are *{a, b, c, d, e, f, g}* and output is *{Z}*.



(iv) Assuming **Boolean matching**, determine the number of ROBDD’s that need to be stored in the cell library for each of the following cells. Justify your answer.

a. f = a b + a c + b c
b. f = a b c + a’ b’ d

[25 Points]

(Q2) Consider the incompletely-specified FSM that has 4 states, two inputs and two outputs, represented by the following state table:

Product	Input	Present State	Next State	Output
P1	10	S1	S2	11
P2	00	S2	S2	11
P3	01	S2	S2	00
P4	00	S3	S2	00
P5	10	S2	S1	11
P6	10	S3	S1	11
P7	00	S1	S1	– –
P8	01	S3	S0	00
P9	11	S1	S1	10
P10	11	S3	S3	01
P11	11	S0	S0	11

(i) Assuming the following constraints, S0 = S1 OR S3, and that the code of S2 is covered by all other state codes, the state table can be reduced into the table shown below. Using implicant merging, covering and disjunctive relations show step by step how you can obtain the reduced state stable given below.

Input	Present State	Next State	Output
–0	S1, S2	S2	11
10	S2, S3	S1	11
00	S1	S1	– –
01	S3	S0	00
11	S0, S1	S1	10
11	S0, S3	S3	01

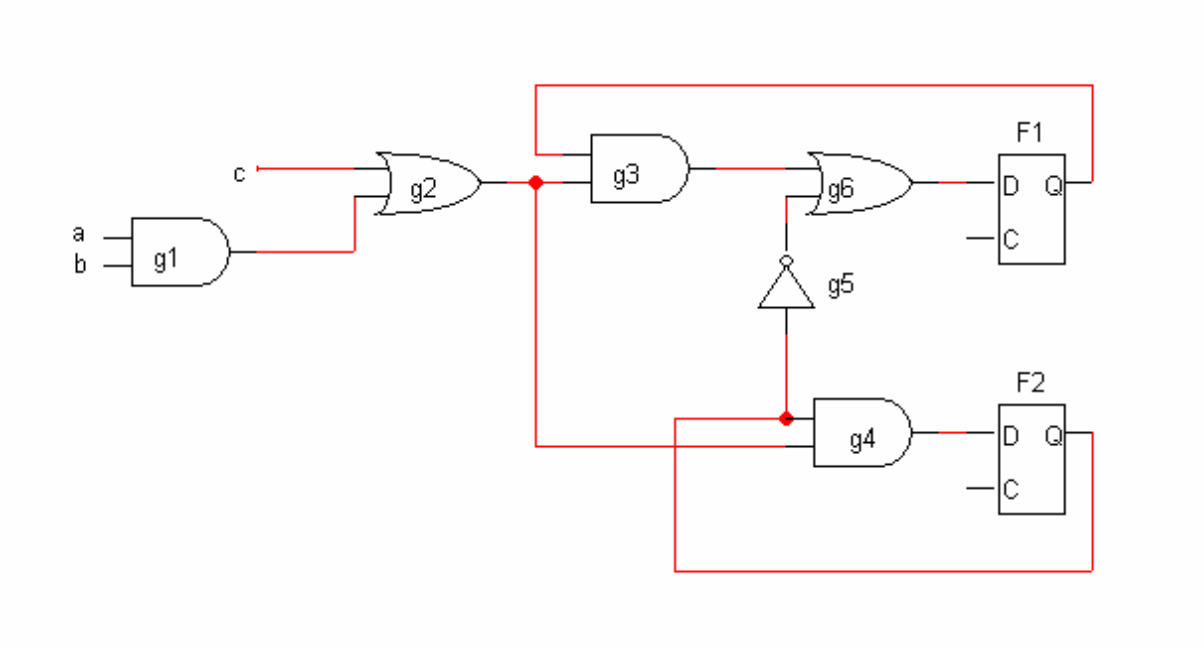
(ii) Show the encoding constraint matrix and compute all the seed dichotomies of the encoding constraint matrix. Then, eliminate seed dichotomies that violate the given covering and disjunctive constraints.

(iii) Compute all the prime dichotomies and eliminate those that violate the disjunctive constraints.

(iv) Find a state encoding satisfying the given constraints. Verify that your encoding satisfies all the constraints.

[10 Points]

(Q3) Assume that the delay of an Inverter is 1 unit delay, the delay of a 2-input AND gate is 2 unit delays, and the delay of a 2-input OR gate is 2 unit delays. Consider the circuit given below:



(i) Determine the critical path of this circuit and the maximum propagation delay.

(ii) Using only the **Retiming** transformation, reduce the critical path of this circuit with the minimum number of flip-flops possible.

[40 Points]

(Q4) Consider the equation given below for y:

[1]= a * b;	[2]=c+d;	[3]=e*f;	[4]=f+g
[5]= [1]+[2];	[6]=[5]-[3];	y=[6]+[4];	

Inputs are *{a, b, c, d, e, f, g}* and output is *{y}*. Assume that the delay of Multiplier is 50 ns and that of an Adder is 10ns. Assume that the cycle length is 55ns.

(i) Draw the **sequencing graph** for the above equation.

(ii) Show the **ASAP** and **ALAP** scheduling of the sequencing graph. Compute the **mobility** of each operation.

(iii) Using **List Scheduling** algorithm LIST_L, schedule the sequencing graph into the **minimum number of cycles** under the resource constraints of one multiplier and one adder. Show the details of the algorithm step by step and the resulting scheduled sequencing graph.

(iv) Show an **Integer Linear Programming (ILP)** model for scheduling the sequencing graph into the minimum number of cycles under the resource constraints of one multiplier and one adder. Assume an upper latency bound of **five** clock cycles.

(v) Consider the scheduled sequencing graph based on **ALAP scheduling**. Assume that the input values will be available to the circuit for only one clock cycle.

a. Show the life-time of all intermediate variables.

b. Determine the minimum number of registers that are required to store all the temporary variables. Show the mapping of variables to registers. Select a mapping that **minimizes the number of multiplexers and interconnect area** as much as possible. Assume that operations [1] and [3] are bound to the same multiplier, operations [2], [5], and [6] are bound to the same adder, and operations [4] and y are assigned to the same adder.

c. Draw the **data-path** implementing the ALAP scheduled sequencing graph based on the variable-register mapping that you obtained in (b).