Jan. 7, 2010

COMPUTER ENGINEERING DEPARTMENT

COE 561

Digital System Design and Synthesis

MAJOR EXAM II

(Open Book Exam)

First Semester (091)

Time: 3:30-6:00 PM

Student Name : \_KEY\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID. : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| **Question** | **Max Points** | **Score** |
| **Q1** | **30** |  |
| **Q2** | **15** |  |
| **Q3** | **15** |  |
| **Q4** | **20** |  |
| **Q5** | **20** |  |
| **Total** | **100** |  |

#

#  **[30 Points]**

# **(Q1)** Consider the function F(A, B, C, D) with **ON-SET=Σm(5, 6, 7, 13)** and **OFF-SET=Σm(1, 8, 12)**. Note that you do not need to use the positional-cube notation in your solution.

## **Expand** the minterm **ABC'D** using ESPRESSO heuristics.

##  A cover of the function is given by F = C + BD. **Reduce** the cube **C** using Theorem 7.4.1.

## Use Corollary 7.4.1 to check if the implicant **C** is an **essential** prime implicant.

#

#

#  **[15 Points]**

# **(Q2)** Consider the logic network defined by the following expression:

#  *x = a c e + a c f + b c e + b c f + d e + d f*

## Using the recursive procedure **KERNELS**, compute all the kernels and co-kernels of *x*. Show all the steps of the algorithm. Assume the following lexicographic order: {a, b, c, d, e, f}.



##

#  **[15 Points]**

# **(Q3)** Consider the logic network defined by the following expression:

#   *x = a b e' f' + a b' e + a b' f + a' d e' f' + a' d' e + a' d' f + c e' f'*

## Compute the weight of the double cube divisors *d1 = a b + a'd* and *d2 = e + f*. Extract the double cube divisor with the highest weight and show the resulting network after extraction and the number of literals saved.

#

#

#  **[20 Points]**

# **(Q4)** Consider the logic network defined by the following expressions with inputs *{a, b, c, d, e}* and output {*y}*:

#

# *x = a b + a' d*

# *w = x' d' + b c'*

# *y = (w ⊕ e) a' b*

#

## Simplify the function *w* based on the utilization of don't care conditions.

## Based on perturbation analysis starting with the original network, determine if it is possible to change the implementation of *x* to *x= a.*



 ****

#  **[20 Points]**

# **(Q5)** Consider the logic network below with inputs *{a, b, c, d, e, f, g}* and output *{X}*:

#

# Assume that the delay of a gate is related to the number of its inputs i.e. the delay of a 2-input AND gate is 2. Also, assume that the input data-ready times are zero for all inputs.

## Compute the data ready times and slacks for all vertices in the network.

## Determine the topological critical path.

## Suggest an implementation of the function *X* using only 2-input gates to reduce the delay of the circuit to the minimum possible and determine the maximum propagation delay in the optimized circuit. Has the area been affected?

##

