**COE 561, Term 111**

**Digital System Design and Synthesis**

**HW# 3**

**Due date: Saturday, Dec. 3**

# Consider the following function:

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

## Compute all the kernels of *X* using the recursive kernel computation algorithm. Show all the steps.

## Compute all the kernels of X based on matrix representation. Compare your answer to the result obtained in (i).

## Use the sis command ***print\_kernel*** and compare the kernels obtained to your answers in (i) and (ii).

## Find a good factor of *X*. Assume that input variables are sorted in lexicographic order. Determine the number of literals obtained. Compare your solution with the result obtained by running the sis commands ***factor –g x; print\_factor; print\_stats –f.***

# Consider the following function:

## *x = a b d + a b' d' + a' c d + a' c' d' + e g h + e' f' g' + e' f' h' + e' f' i j + e' f' i' + e' f' j' + f g h*

## Compute all double-cube divisors of *x* along with their bases and their weights. Show only double-cube divisors that have non-empty bases.

## Apply the fast extraction algorithm based on extracting double-cube divisors along with complements or single-cube divisors with two-literals. Show all steps of the algorithm. Determine the number of literals saved. Compare your solution with the result obtained by running the sis commands ***fx***.

# Consider the logic network defined by the following expressions:

*X = A B;*

*Y = A B C X + A B' C' X';*

*Z = A' + Y;*

Inputs are {A, B, C} and output is {Z}.

## Compute the CDC set for the cut at the inputs of circuit Y.

## Compute the ODC set for node Y.

## Simplify the function of Y using both its ODC and CDC.

## Compute the ODC set for node X based on the resulting simplified function and simplify its function.

## Apply the sis command ***full\_simplify*** and compare the solution obtained with your obtained solution based in (iii) & (iv).

# Consider the logic network defined by the following expressions:

#

#  f = a + b

#  g = f c

#  h = f d

#  i = g + h

#  j = i e

#  k = b’ c’ d’

#  x =j + k

# Inputs are {a, b, c, d, e} and output is {x}. Assume that the delay of a gate is related to the number of its inputs. Also, assume that the input data-ready times are zero except for input a, which is equal to 2.

## Draw the logic network graph and compute the **data ready times** and **slacks** for all vertices in the network.

## Determine the **maximum propagation delay** and the **topological critical path**.

## Suggest an implementation of the function ***x*** to reduce the delay of the circuit. What is the **maximum propagation delay** after the modified implementation?