National University

of Computer and Emerging Sciences Chiniot -Faisalabad Campus

EE1005 – Digital Logic Design
Quiz# 3

Instructor: Muhammad Adeel Tahir **Section:** SE – 2A **Time:** 20 Minutes

Name: ___________________________

Roll No: ___________________________ **Total:** 20 Marks

Note: *Use the back side of the page if needed. Make sure the handwriting is neat while drawing the circuit, quiz will be marked as 0 if attempted in a writing that is not readable at all.* ***Cutting will result in negative marking.***

Q1: Draw a NAND logic diagram that implements the complement of the following function, use 2 input NAND gate for your implementation, label each output of gate carefully to avoid deduction of marks.

(10 marks)

$$F(A,B,C,D) = \Sigma(0,1,2,3,6,10,11,14)$$

Solution:

Marking Criteria is as followed:

- **For correctly identifying the equation 5 marks are awarded.**
- **For correctly identifying the circuit 5 marks are awarded. Binary checking is followed in the circuit, no partial marking whatsoever.**

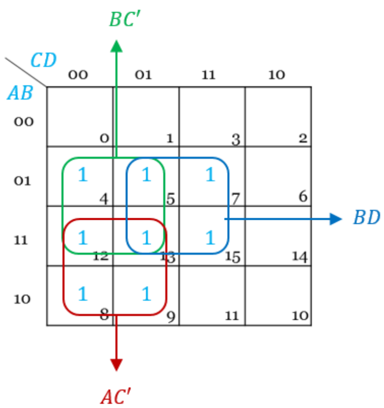
The complement of the given function:

$$F(A,B,C,D) = \sum(0,1,2,3,6,10,11,14)$$

$$F' = \sum(4,5,7,8,9,12,13,15)$$

We plot and simplify the function *G* (the complement of *F*) using a 4-variable k-map as shown below. The simplified function is

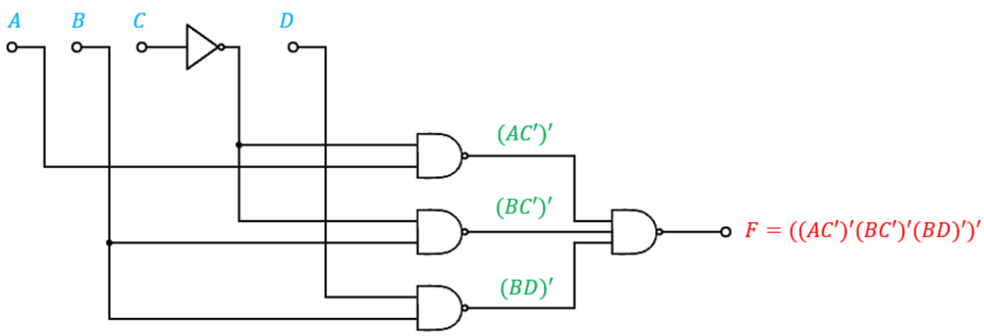
$$G(A,B,C,D) = AC' + BC' + BD$$



The simplified function *G* can be expressed as

$$G(A,B,C,D) = [G'(A,B,C,D)]' = [(AC' + BC' + BD)']' = [(AC')'(BC')'(BD)']' \Rightarrow Demorgan's\ Theorem$$

Therefore, we implement this function using a two-level NAND gate circuit as shown below.



Q2: Implement the following Boolean function F, together with the don't-care conditions d , using no more than two NOR gates: (10 marks)

$$\mathbf{F(A,B,C,D) = \Sigma(2,4,10,12,14)}$$

$$\mathbf{d(A,B,C,D) = \Sigma(0,1,5,8)}$$

Solution:

Marking Criteria is as followed:

- **For correctly identifying the equation 5 marks are awarded. Incase k-map is wrong, 0 is awarded.**
- **For correctly identifying (with steps) the circuit 5 marks are awarded. Binary checking is followed in the circuit, no partial marking whatsoever.**

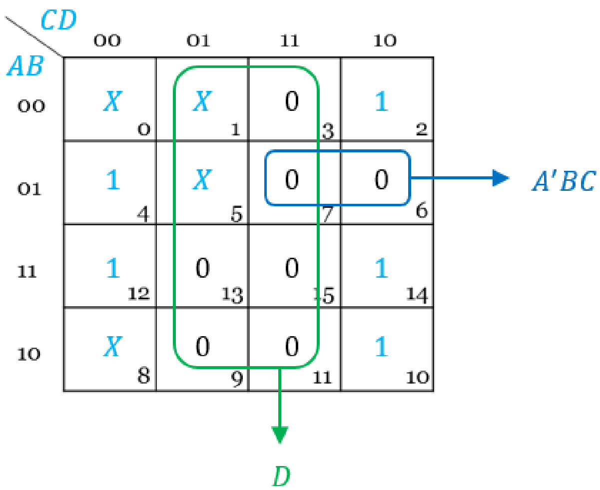
First, the 1's and X's of the given function:

From the 0's and X's, we obtain the simplified-complemented function:

$$F' = D + A'BC$$

Thus, the minimum product of sums for *F* is

$$F = D'(A + B' + C')$$



We reformulate the function *F* so that it can be implemented using NOR gates as follows:

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