University of Asia Pacific

Department of Computer Science & Engineering

Lab Report

Course Title: *Digital Logic & System Design Lab.*

Course Code:CSE 210

Experiment No: 02

Experiment Name: Simplify the given logic expression truth table.

i) (A + C) (A D + A D’) + A C + C

ii) A’ B C + A B’ C + A B C’ + A B C (Using K-map)

Submitted By:

Sohag Hossen

Reg. No. 18201075

Section- B2

Dept. of CSE

Submission date :12.07.2020

Problem Statement : **Simplify the given logic expression and verify the truth table**

i) (A + C) (A D + A D’) + A C + C
ii) A’ B C + A B’ C + A B C’ + A B C (Using K-map)

Instruments :

1. **AND(7408)**
2. **OR (IC 7432)**
3. **NOT(IC7404)**

1. Simplification of the equation :

Y = (A + C) (A D + A D’) + A C + C
=(A+C)A(D+D’)+AC+C
= (A+C)A.1+AC+C
=(A+C)A.C+1
=A.A+A.C+C
=(A+C)(A+1)
=A.A+C
= A + C (simplified)

Y = A’ B C + A B’ C + A B C’ + A B C
= AB + BC + AC (simplified)

ii) Y=A’BC+AB’C+ABC’+ABC(using k-map)

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

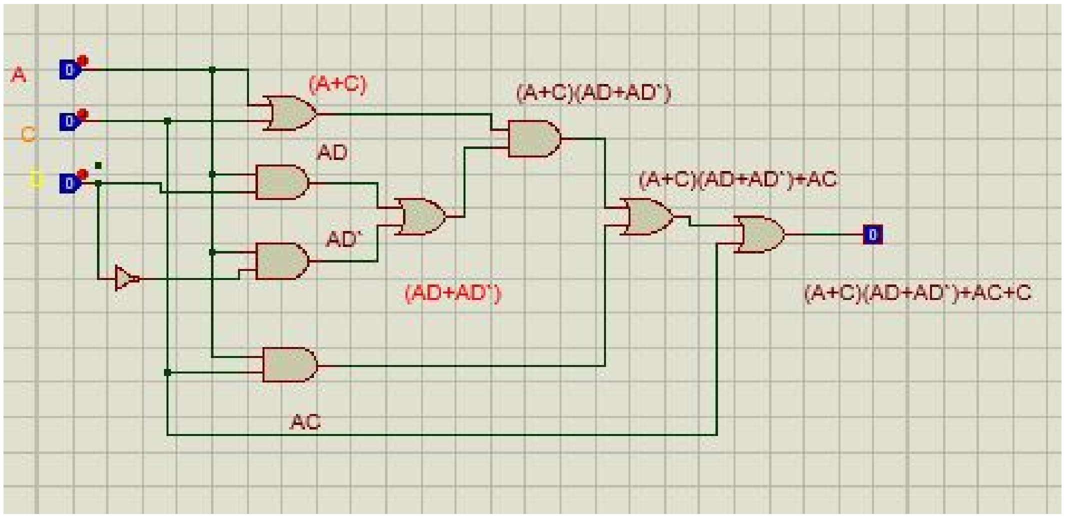
Karnaugh map(k-map):

	00	01	11	10
0	0	0	1	0
1	0	1	1	1

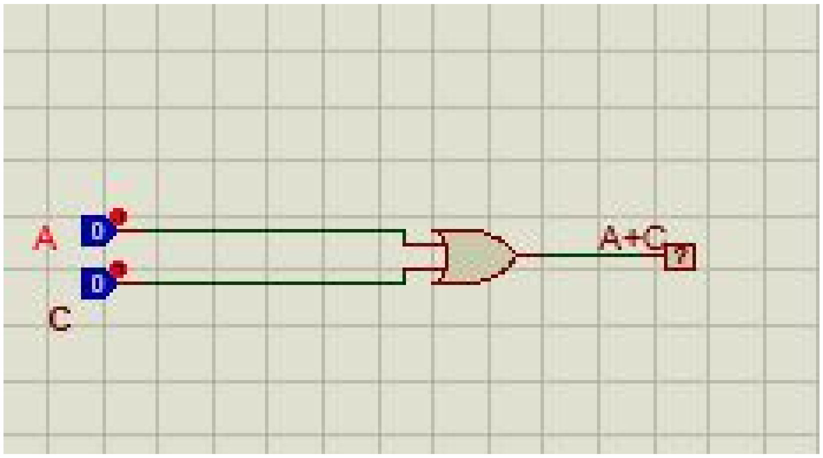
For k-map simplification output is Y=AB+BC+AC(Simplified)

1. Circuit Diagram of the equations (both simplified and not simplified) :

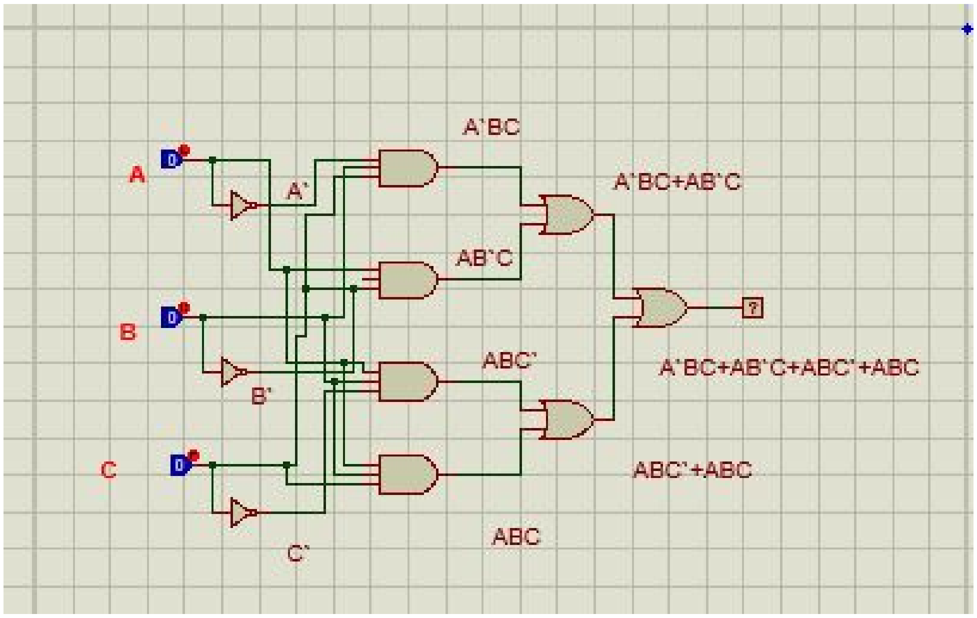
i)**Not Simplified : (A + C) (A D + A D’) + A C + C**



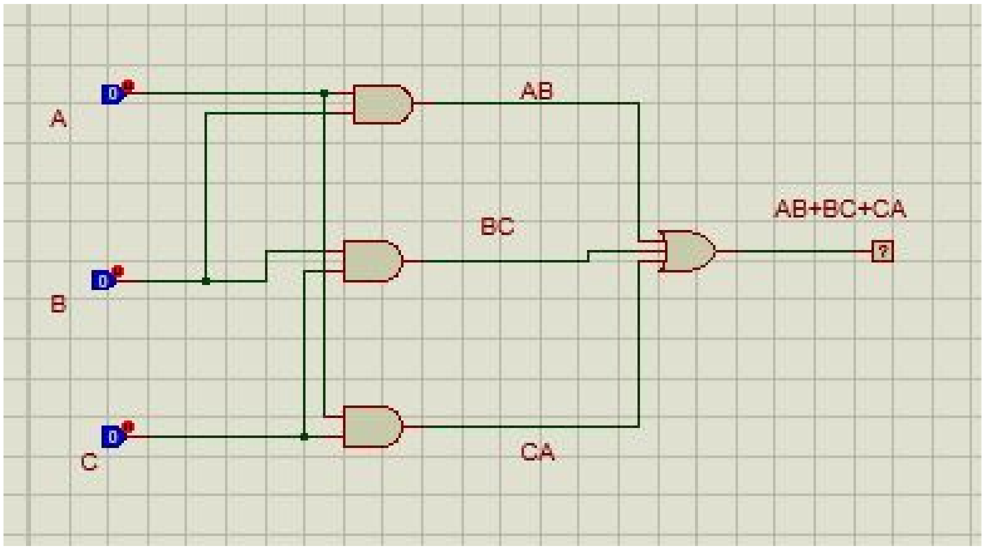
Simplified : **A+C**



ii) **Not Simplified :** A’ B C + A B’ C + A B C’ + A B C (Using K-map)



Simplified : AB + BC + AC



5. Truth table of the equations (both simplified and not simplified) :

i)Y=(A + C) (A D + A D’) + A C + C [Not Simplified]

A	C	D	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Y = A+C   [Simplified]

A	C	D	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

ii) Y= A’ B C + A B’ C + A B C’ + A B C [Not Simplified]

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Y= AB + BC + AC [Simplified]

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Discussion:

I have observed the works of the basic logic operator. Often, when you deduce a Boolean expression from the system's specifications, you will get a complex expression that can be simplified. When you simplify a Boolean expression, you aim to get a simpler or less complex circuit in terms of the hardware. Designers always strive to build less complex circuits as this leads to many desirable features, such as less power consumption and less expensive circuits.There are many methods to simplify a logic expression. Some of these methods are using Boolean Algebra laws, Karnaugh maps and Quine-McCluskey algorithms. Usually, we use Boolean Algebra laws for simple expressions, Karnaugh maps for expressions .