National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Electrical Engineering

Faculty Member: Sir Arshad Nazeer **Dated: 12th October 2023**

Semester: 3rd **Section: D**

Group No.: 10

EE-221: Digital Logic Design

Lab 5: Minimization of Boolean Functions

		PLO4/CLO4	PLO4/CLO4	PLO5/CLO5	PLO8/CLO6	PLO9/CLO7	
Name	**Reg. No**	**Viva / Lab Performance** 5 Marks	**Analysis of data in Lab Report** 5 Marks	**Modern Tool Usage** 5 Marks	**Ethics and Safety** 5 Marks	**Individual and Teamwork** 5 Marks	**Total marks Obtained** 25 Marks
Arooj Fatima	**423365**						
Irfa Farooq	**412564**						
Ahmad Nasir	**409959**						
Haseen Umer	**427442**						

Lab 5: Minimization of Boolean Functions

This Lab has been divided into two parts.

The first part is the hardware implementation of a Boolean function given to you. But you have to first minimize the Boolean functions to minimum number of literals.

In next part you will simulate the same circuit using Verilog.

Objectives:

- ✓ Understand Minimization of Boolean Functions
- ✓ Simulate Basic Circuits using Verilog.
- ✓ Hardware Implementation of Basic Logic Circuits

Lab Instructions

- ✓ This lab activity comprises three parts, namely Pre-lab, Lab tasks, and post-Lab Viva session.
- ✓ The lab report will be uploaded on LMS three days before the scheduled lab date. The students will get a hard copy of lab report, complete the Pre-lab task before coming to the lab and deposit it with teacher/lab engineer for necessary evaluation. Alternately each group to upload completed lab report on LMS for grading.
- ✓ The students will start lab task and demonstrate design steps separately for step-wise evaluation (course instructor/lab engineer will sign each step after ascertaining functional verification)
- ✓ Remember that a neat logic diagram with pins numbered coupled with nicely patched circuit will simplify trouble-shooting process.
- ✓ After the lab, students are expected to unwire the circuit and deposit back components before leaving.
- ✓ The students will complete lab task and submit complete report to Lab Engineer before leaving lab
- ✓ There are related questions at the end of this activity. Give complete answers.

.

Pre-Lab Tasks: (To be done before coming to the lab) (2 marks)

1. **Write the Boolean expression of the following two functions. Simplify the expression using algebraic manipulation and draw the logic diagram.**

   F (A, B, C) = ∑ (2, 3, 7)

Solution:

Binary representation of min-terms:

2 in Binary = 010 = A'BC'

3 in Binary = 011 = A'BC

7 in Binary = 111 = ABC

F (A, B, C) = ∑ (2, 3, 7) = A'BC' + A'BC + ABC (SOP Form)

Using Consensus Theorem and Algebraic manipulation:

A'BC' + A'BC + ABC

= A'B(C' + C) + ABC

= A'B + ABC

= A'B + ABC + BC

= A'B + (A + 1)BC

= A'B + BC

= B(A' + C) (POS Form

G (A, B, C) = ∑ (4, 5, 7)

Solution:

Binary representation of min-terms:

4 in Binary = 100 = AB'C'

5 in Binary = 101 = AB'C

7 in Binary = 111 = ABC

G (A, B, C) = ∑ (4, 5, 7) = AB'C' + AB'C + ABC (SOP Form)

Using Consensus Theorem and Algebraic manipulation:

AB'C' + AB'C + ABC

= AB'(C' + C) + ABC

= AB' + ABC

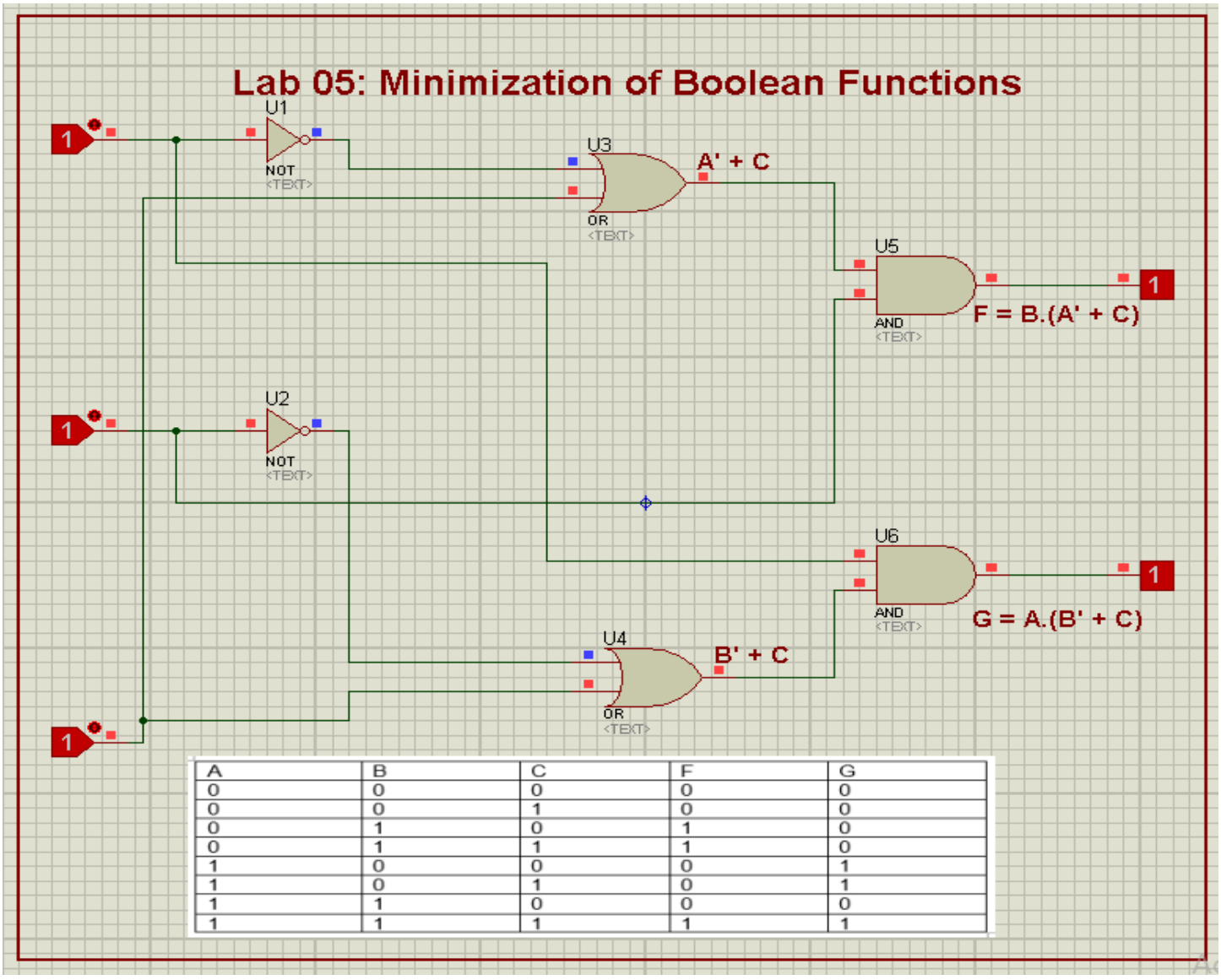
= AB' + ABC + AC

= AB' + (B + 1)AC

= AB' + AC

= A(B' + C) (POS Form)

Logic Diagram:



A	B	C	F	G
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

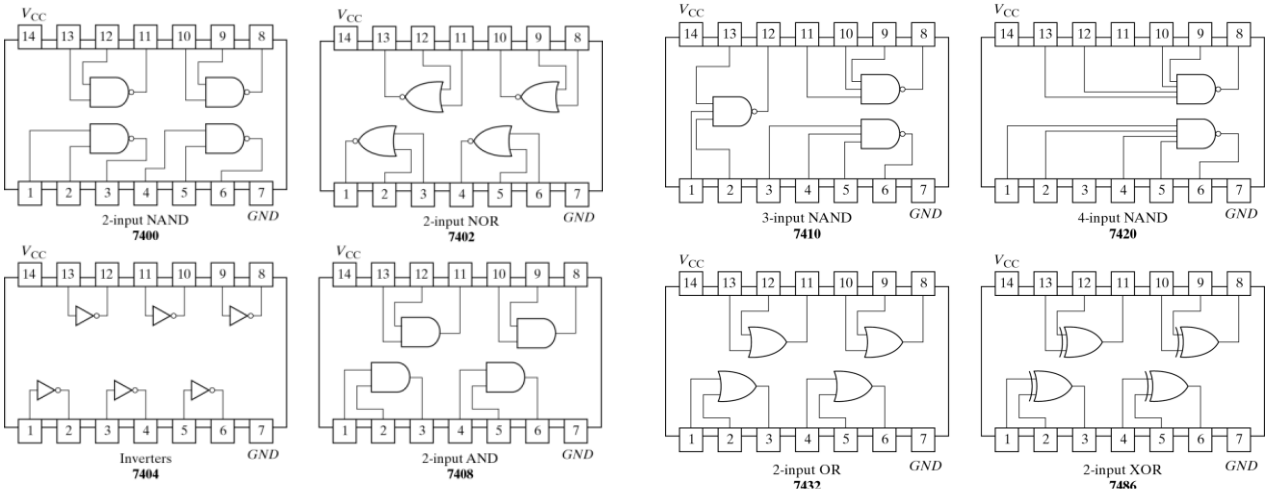
2. Mention the number of literals and gates needed for implementing the above function in hardware. (1 Mark)

   I. F (A, B, C) = ∑ (2, 3, 7)

   The number of literals in the expression B(A' + C) are three and the number of gates used to implement the above function are also three i.e.; one NOT Gate to perform A', one OR Gate to perform (A' + C) and one AND gate to get the final answer B(A' + C).

   II. G (A, B, C) = ∑ (4, 5, 7)

   The number of literals in the expression A(B' + C) are three and the number of gates used to implement the above function are also three i.e.; one NOT Gate to perform B', one OR Gate to perform (B' + C) and one AND gate to get the final answer A(B' + C).



Lab Tasks (3 marks)

Lab Task 1:

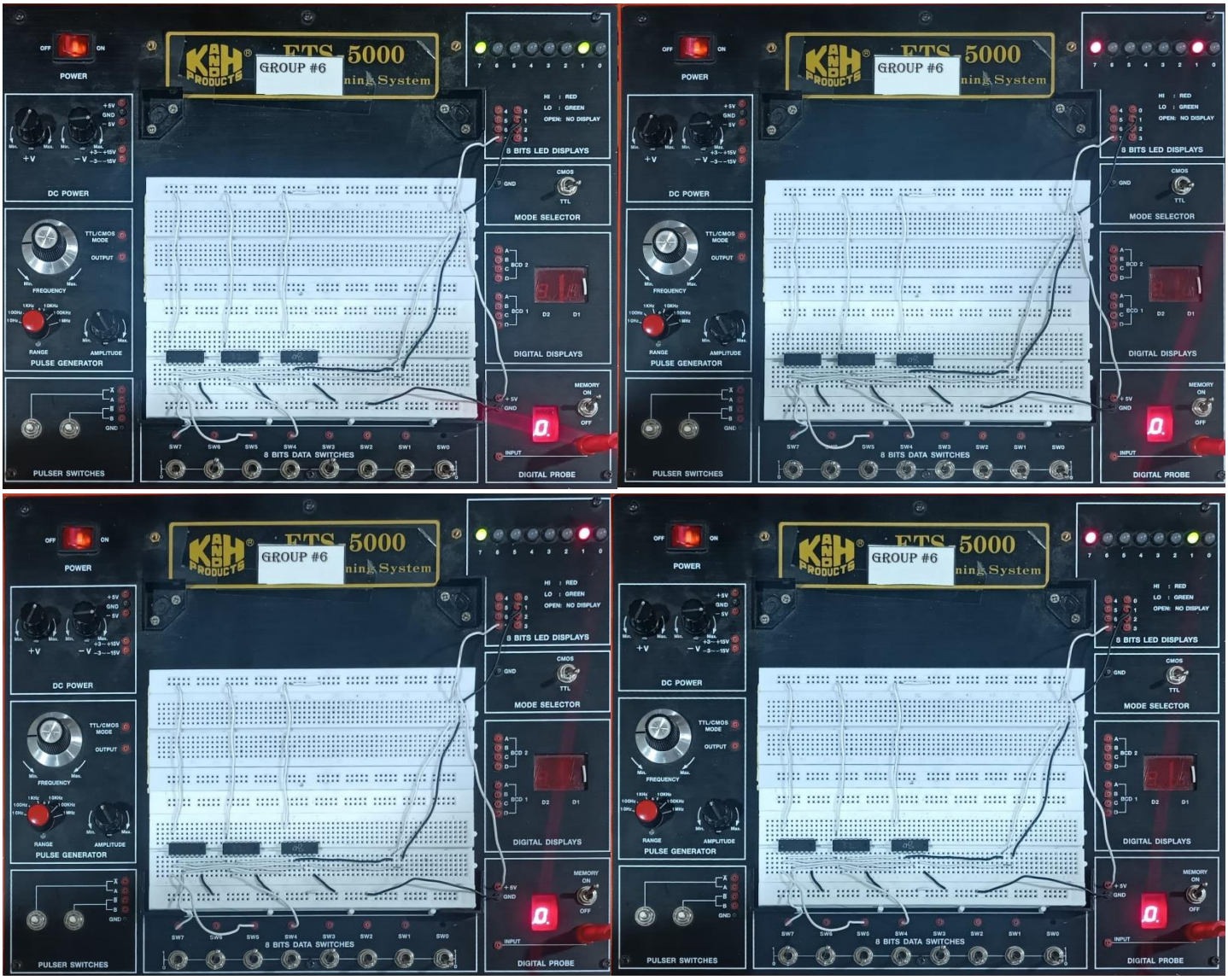
Implement the Boolean functions in hardware you simplified in your Pre-Lab Task. Make truth table and **Schematic**. Mention what and how many gates you would be using? The following gates are available to you.

Truth Table:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

A	B	C	G
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Hardware Screenshots:



Lab Task 2:

Write Verilog code for the minimized functions at gate-level and perform simulation. Attach the relevant snapshots below.

Verilog Code:

```
ln #
 1 module Lab_05(f, g, a, b, c);
 2 input a, b, c;
 3 output f, g;
 4 wire w1, w2, na, nb;
 5 not n1(na, a);
 6 or o1(w1, na, c);
 7 and a1(f, w1, b);
 8 not n2(nb, b);
 9 or o2(w2, nb, c);
10 and a2(g, w2, a);
11 endmodule
12
13 module T1;
14 reg A, B, C;
15 wire F, G;
16 Lab_05 T1(F, G, A, B ,C);
17 initial
18 begin
19         #100 A=1'b0; B=1'b0; C=1'b0;
20         #100 A=1'b0; B=1'b0; C=1'b1;
21         #100 A=1'b0; B=1'b1; C=1'b0;
22         #100 A=1'b0; B=1'b1; C=1'b1;
23         #100 A=1'b1; B=1'b0; C=1'b0;
24         #100 A=1'b1; B=1'b0; C=1'b1;
25         #100 A=1'b1; B=1'b1; C=1'b0;
26         #100 A=1'b1; B=1'b1; C=1'b1;
27 end
28 endmodule
```

Output Waveform:

Conclusion:

In this lab, we used the concept of min-terms to derive an expression for function F and G. Using algebraic manipulation, we were able to simplify the equations to a minimum of two terms and three literals. Hence, min-terms allowed us to easily plot the Proteus schematic and write the Verilog code.