

Digital Logic Design Lab Manual: Boolean Function Optimization

Lab Objectives:

1. Understand how to create a truth table for a Boolean function.
2. Learn how to implement a Boolean function using basic logic gates in hardware.
3. Apply Boolean algebra rules to optimize the given Boolean function.
4. Implement the optimized function using hardware and Proteus simulation software.
5. Compare the original and optimized implementations regarding complexity and hardware requirements.

Equipment Required:

- Breadboard
- Logic Gates ICs (AND, OR, NOT, NAND, NOR)
- Connecting wires
- Multimeter (optional for checking connections)
- Proteus Simulation Software
- Power Supply

Review the following Boolean identities and laws:

1. Commutative Law
2. Associative Law
3. Distributive Law
4. De Morgan's Theorems
5. Redundancy Laws

Lab Procedure:

Step 1: Create the Truth Table.

Step 2: Hardware & Software Implementation of the Unoptimized Boolean Function.

Step 3: Boolean Function Optimization.

Step 4: Hardware Implementation of the Optimized Boolean Function.

Step 5: Discuss the differences in terms of hardware complexity, gate count, and functionality between the original and optimized designs.

Lab Tasks: Boolean Functions to Implement:

Task 1: $F(A, B, C) = (A + B') (A' + C)$

Task 2: $F(A, B, C, D) = A'B + C'D + AB'$

Task 3: $F(A, B, C) = AB + A'BC + B'C$

Task 4: $F(A, B, C, D) = (A' + B) (C + D') + AB'$