Paper 7, TDC Part-3
Chapter– 4, Combinational Logic Design
Lecture - 1

By:

Mayank Mausam

Assistant Professor (Guest Faculty)

Department of Electronics

L.S. College, BRA Bihar University,

Muzaffarpur, Bihar

Combinational Logic Design

Introduction : -

A logic expression can be realized using logic gates.

Before realizing the logical expression it can be reduced using boolean algebraic theorems. This reduces the number of logic gates and input lines in the realized logic expression. Therefore, the simplification of logic expression is very important as it saves the hardware required to design a specific system.

That is, our aim is to minimize the number of IC packages and thereby power consumption. An IC consists of large number of logic gates (logic function).

Combinational Logic Design

Basically, digital circuits are divided into two broad categories:

1. Combinational Circuits
2. Sequential Circuits

Combinational Circuits :-

In combinational circuits, the outputs at any instant of time depend upon the inputs present at that instant of time. In other words we can say the digital circuits where the logic gates are connected together (combine) to provide the resultant output for certain specified combinations of input variables, with no storage.

So there is no memory in these circuits.

Combinational Logic Design

Design Requirements of combinational circuits may be specified in one of the following ways:

1. A set of statements 2. Boolean expression

3. Truth Table

There can be two different approaches to the design of combinational circuits. One of these is the traditional method, wherein the given Boolean expression or the truth table is simplified by using standard methods and the simplified expression is realized using the gates.

In the other method the logic expression or truth table does not require any simplification. The complex logic functions available in medium scale integrated circuits (MSI) can be used. Computer-aided design (CAD) tools are used for the design using PLDs and FPGAs.

Combinational Logic Design

Methods used to simplify boolean expressions or functions :-

1. Algebraic method
2. Karnaugh map method
3. Quine-McCluskey method
4. Variable entered mapping (VEM) technique.

Standard representations for logic functions :-

Logic functions are expressed in terms of logical variables. As Values assumed by the logic variables is in binary form so the logic functions also assume value in binary form.

Combinational Logic Design

Any arbitrary logic function can be expressed in the following forms:

1. Sum-of-products (SOP) form
2. Product-of-sum (POS) form

SOP form is called AND-OR logic where different ANDed functions are ORed together. It is the basic form for realizing standard Boolean Function.

POS form is called OR-AND logic in which different ORed functions are ANDed together.

Logic functions can be written in other form too, but SOP & POS form are conveniently suited in arriving at the standard methods for designing the circuits.

Combinational Logic Design

1. **SOP Form: -**

1.) SOP Form :-

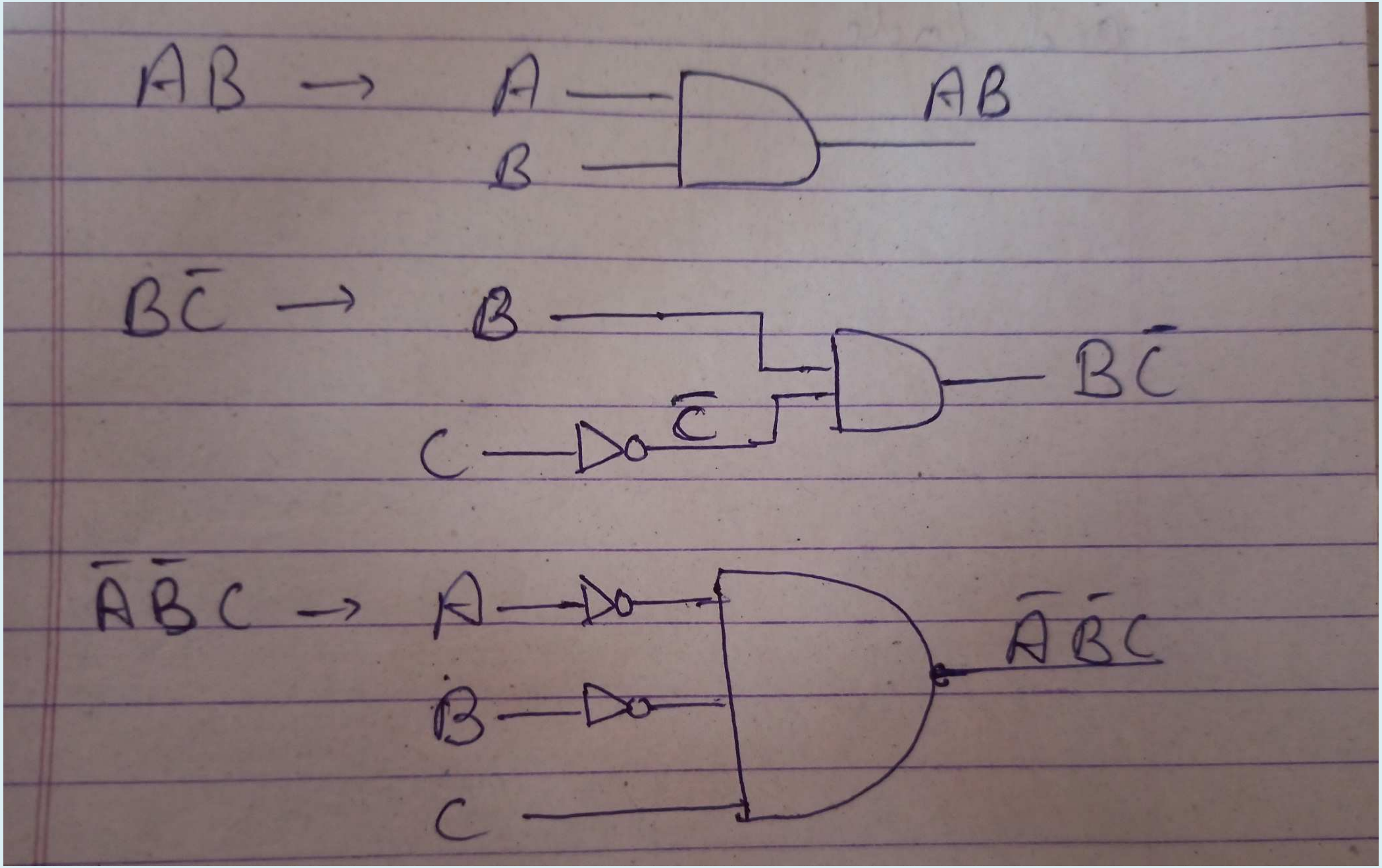
Consider a function F is given by.

$$F = AB + B\bar{C} + \bar{A}\bar{B}C$$

In this function F is in SOP form.

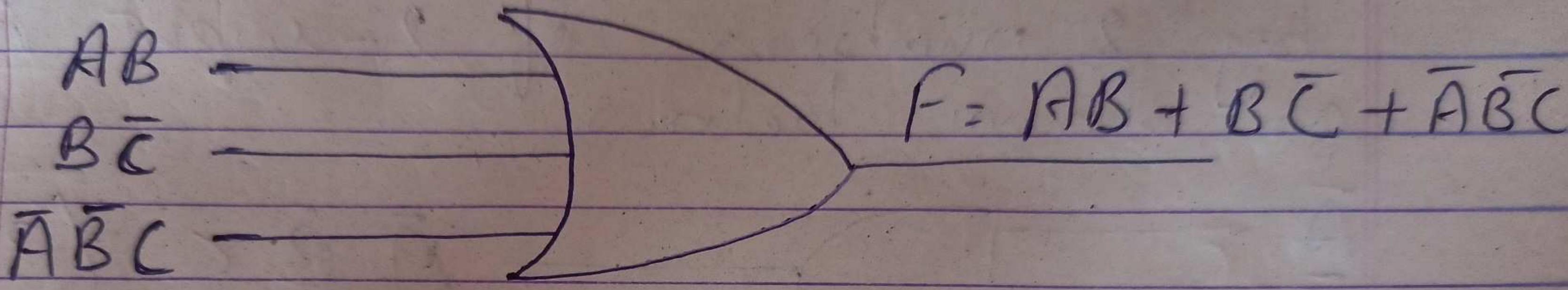
where AB, $B\bar{C}$ & $\bar{A}\bar{B}C$ are ANDed logic and these '3' ANDed logic are ORed together, and the function F can be implemented as below.

Combinational Logic Design



Combinational Logic Design

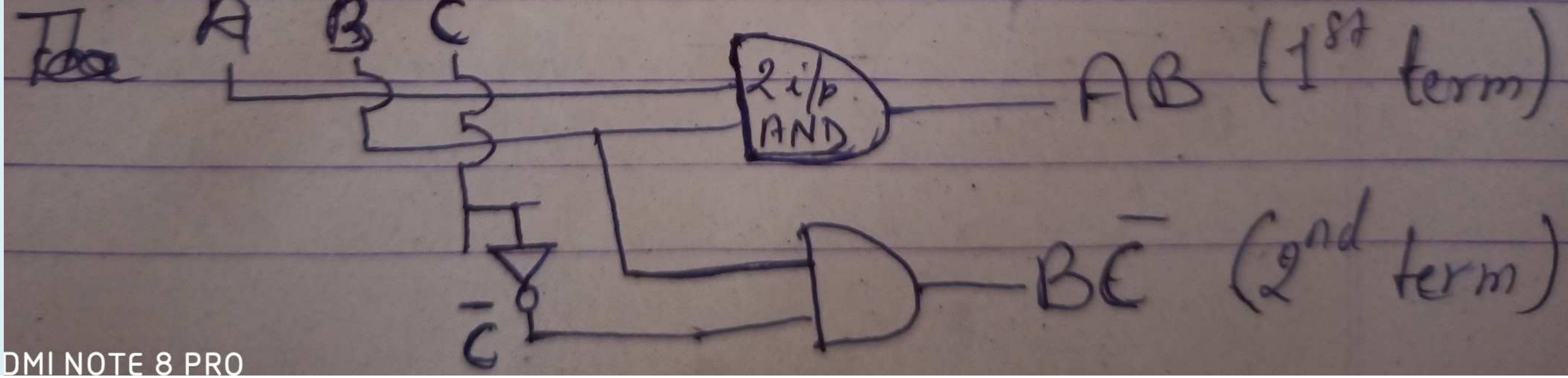
$F = AB + B\bar{C} + \bar{A}\bar{B}C$



From above implementation it can be also observed that there is no memory i.e. present O/P of the Function does not depend on the past output.

In function $F = AB + B\bar{C} + \bar{A}\bar{B}C$, A, B, $\bar{C}$, $\bar{A}$, $\bar{B}$ & C are literals. The 1st term & 2nd term have 2 literals each, A & B and B & $\bar{C}$ respectively, while 3rd term has 3 literals namely $\bar{A}$, $\bar{B}$ & C.

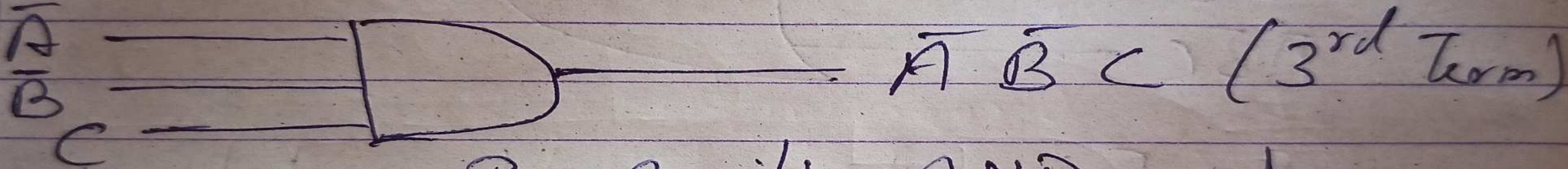
1st term (AB) & 2nd term (B$\bar{C}$) both can be realised by using 2-input AND gate, as shown below.



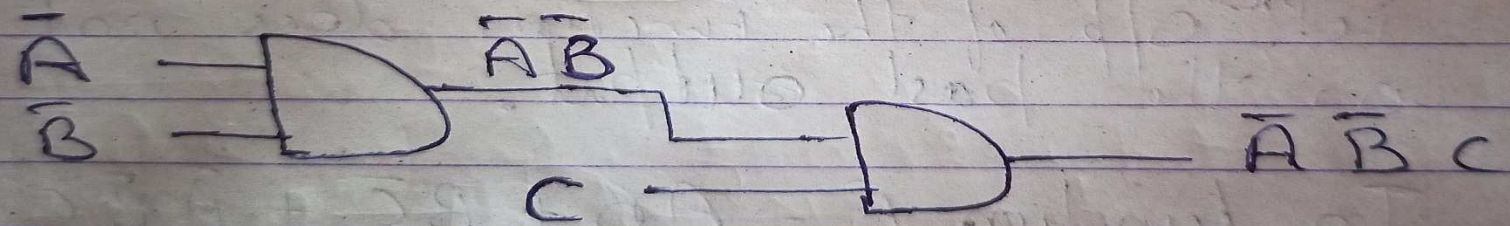
Combinational Logic Design

Note:→ A input variable in uncomplemented (A) or complemented form ($\bar{A}$), is known as literal.

3rd term ($\bar{A}\bar{B}C$) can be realized using either 3-input AND gate or by 2 numbers of 2-input AND gate as shown below.



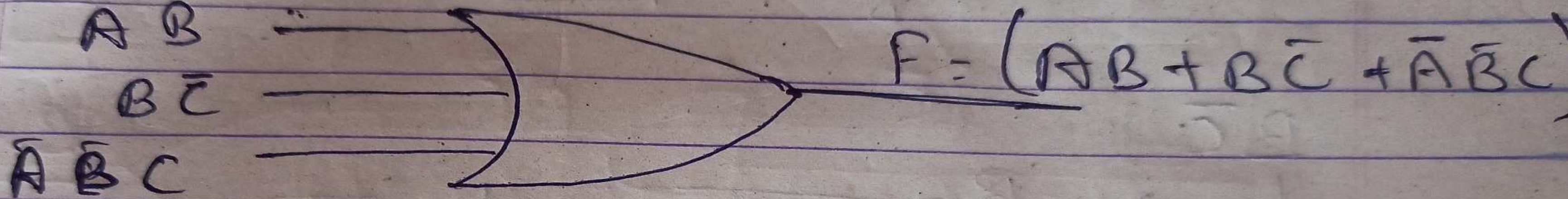
By 3-i/p AND gate



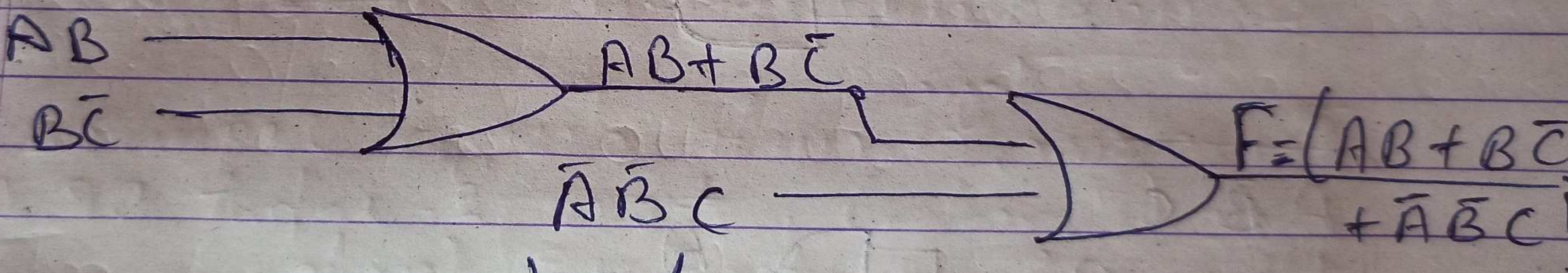
By 2 numbers of 2-i/p AND gate

Combinational Logic Design

The complete function F can be realised by 3-input OR gate or using 2 numbers of 2-i/p OR gate as shown below.



Using 3-i/p OR gate



Using 2 numbers of 2-i/p OR gate.

Combinational Logic Design

* Implementation (Realization of Function using NAND Gate).

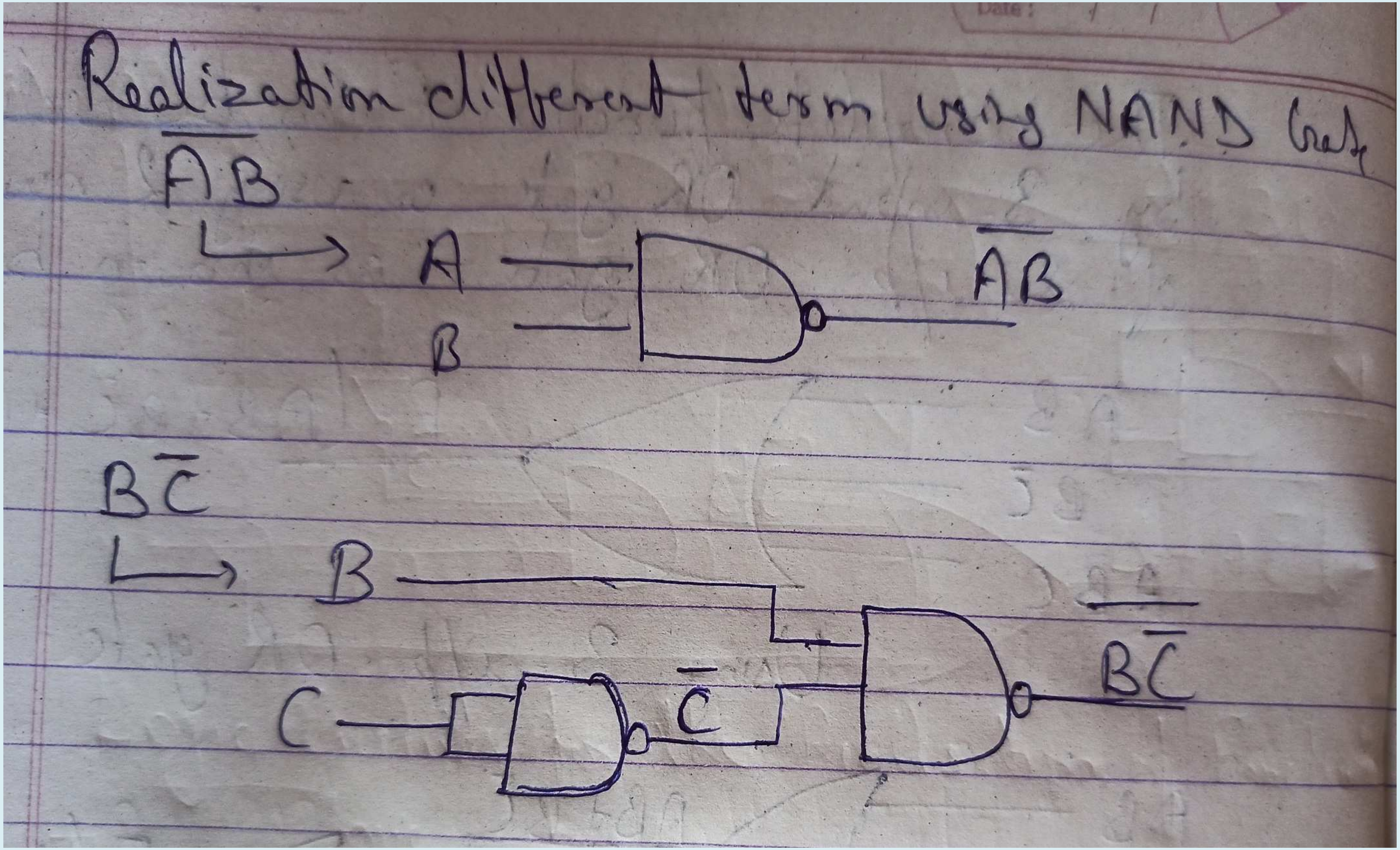
$$F = (AB + B\bar{C} + \bar{A}\bar{B}C)$$

$$F = \overline{\overline{F}} = \overline{\overline{(AB + B\bar{C} + \bar{A}\bar{B}C)}}$$

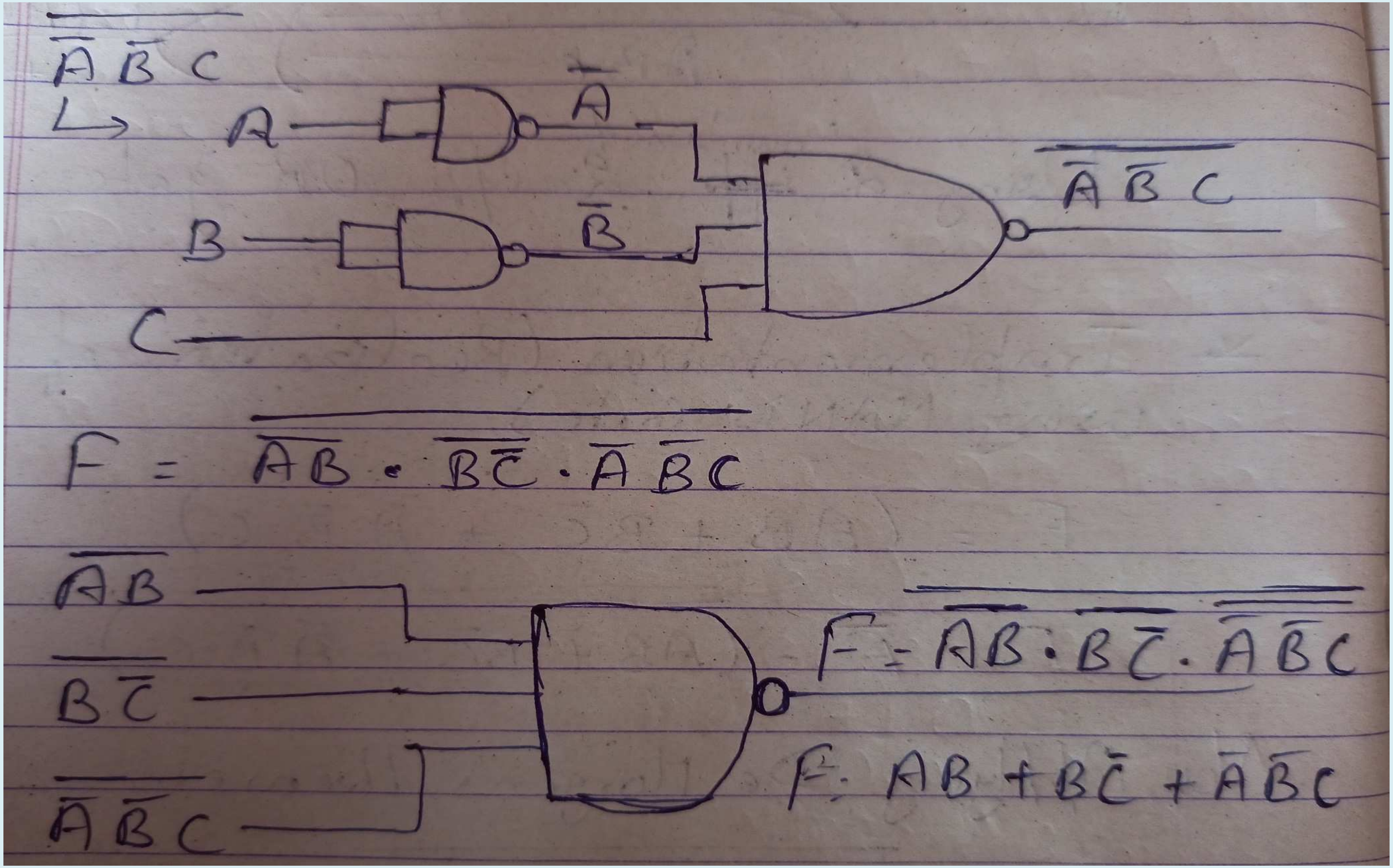
Now, Applying De Morgan's Theorem

$$F = \overline{\overline{AB} \cdot \overline{B\bar{C}} \cdot \bar{A}\bar{B}C}$$

Combinational Logic Design



Combinational Logic Design



Number Systems and Codes

Refer book- Modern Digital Electronics by RP Jain.

Thank You