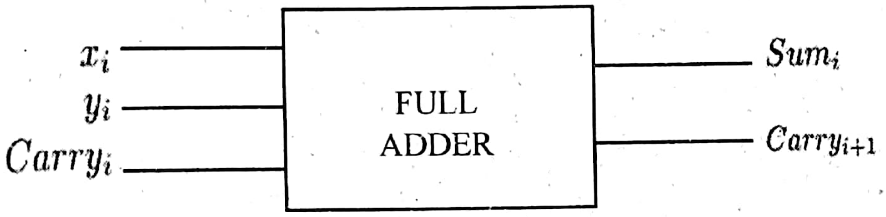
Chapter 2 (Digital Logic)

1. Realize full adder circuit using decoders and gates. Subtract $(43)_{10}$ from $(57)_{10}$ using 2's complement method. [3+3] 2078 Kartik

Part 1:

Realization of full adder using decoder and gates:

i. Block diagram:



ii. Truth Table:

Inputs			Outputs	
X_i	Y_i	$Carry_i$	Sum_i	$Carry_{i+1}$
0	0	0	0	0
0	0	1	**1**	0
0	1	0	**1**	0
0	1	1	0	**1**
1	0	0	**1**	0
1	0	1	0	**1**
1	1	0	0	**1**
1	1	1	**1**	**1**

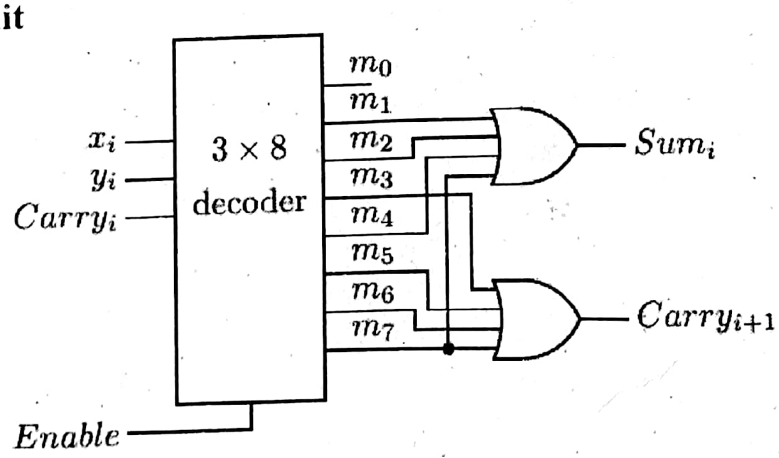
iii. Expression for Sum

$Sum_i = \sum(m1, m2, m4, m7)$

iv. Expression for Carry

$Carry_{i+1} = \sum(m3, m5, m6, m7)$

v. Logical Circuit



Part 2:

Subtracting $(43)_{10}$ from $(57)_{10}$ using 2's complement method.

+57	→	0 0 1 1	1 0 0 1
+43	→	0 0 1 0	1 0 1 1
&-43	→	1 1 0 1	0 1 0 1 (2's Complement)

So,

+57	→		0 0 1 1	1 0 0 1
+(-43)	→		1 1 0 1	0 1 0 1
14	→	1	0 0 0 0	1 1 1 0

We discard the final carry because it is meaning less in 8-bit arithmetic.
Therefore, $(-43+57)_{10} = (0\ 0\ 0\ 01\ 1\ 1\ 0)_2$

2. State and prove De-Morgan's theorems with necessary diagrams. Construct XOR gate using minimum number of NAND gates. [2+4] 2078 Bhadra

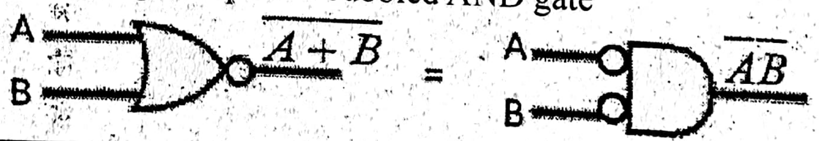
Part 1: <u>Theorem 1:</u>

The complement of a sum is equal to the product of complement.

$$\overline{A+B} = \bar{A}.\bar{B}$$

$$\overline{A+B+C} = \bar{A}.\bar{B}.\bar{C}$$

In term of circuit a NOR gate equals a bubbled AND gate



A	B	$\overline{A+B}$	$\bar{A}$	$\bar{B}$	$\bar{A}.\bar{B}$
0	0	1	1	1	1
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	0

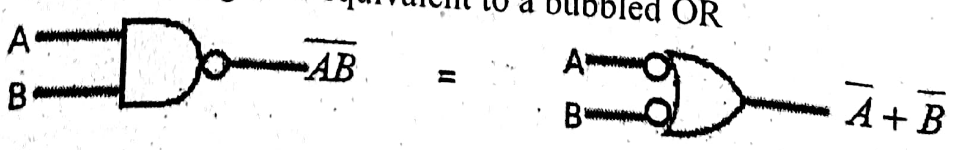
<u>Theorem 2:</u>

The complement of product is equal to the sum of individual complement.

$$\overline{AB} = \bar{A}+\bar{B}$$

$$\overline{ABC} = \bar{A}+\bar{B}+\bar{C}$$

In term of circuit a NAND gate is equivalent to a bubbled OR



A	B	$\overline{AB}$	$\bar{A}$	$\bar{B}$	$\bar{A}+\bar{B}$
0	0	1	1	1	1
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	0	0	0

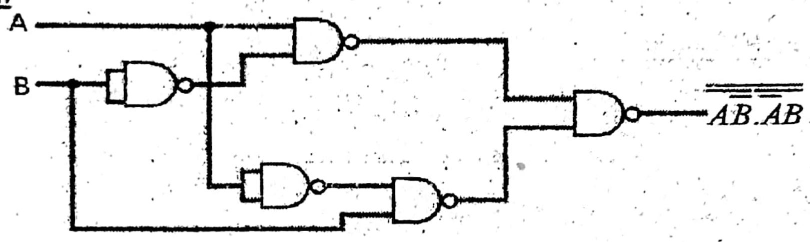
Part 2:

XOR from NAND:

$Y = A \oplus B$

$= \overline{\overline{A\bar{B} + \bar{A}B}} = \overline{\overline{A\bar{B}}.\overline{\bar{A}B}}$

In symbol:



3. **State and prove De-Morgan's theorem and perform the addition (-47+27) by using 2's Complement method.**

Part 1: Refer to 2078 Bhadra

Part 2:

Subtracting $(47)_{10}$ from $(27)_{10}$ using 2's complement method.

+27	→	0 0 0 1	1 0 1 1
+47	→	0 0 1 0	1 1 1 1

&-47	→	1 1 0 1	0 0 0 0 (1's Complement)
			+1
		1 1 0 1	0 0 0 1
&-47	→	1 1 0 1	0 0 0 1 (2's Complement)

So,

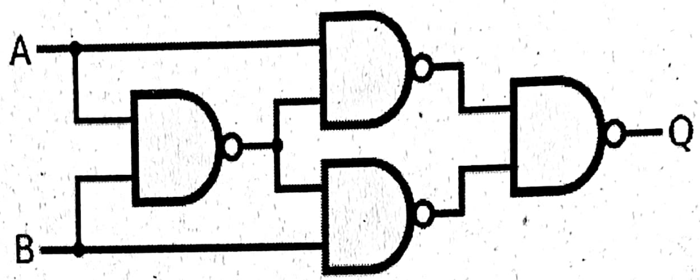
+27	→	0 0 0 1	1 0 1 1
+(-47)	→	1 1 0 1	0 0 0 1
-20	→	1 1 1 0	1 1 0 0

Therefore, $(-47+27)_{10} = (1\ 1\ 1\ 0\ 1\ 1\ 0\ 0)_2$

4. Describe De-Morgan's law with examples. Construct XOR gate using only 3-inputs NAND gates. [2+3] 2076 Ashwin

Refer to 2078 Bhadra

Part 2: XOR gate using NAND Gates



5. State and prove De-Morgan's theorems with necessary diagrams. Prove that negative logic OR Gate is equivalent to positive logic AND gate. [4+2] 2074 Chaitra

1st part: Refer to 2078 Bhadra Q No.1

2nd part:

In positive logic we use binary 1 for high voltage and binary 0 for low voltage while in negative logic just opposite happens.

Positive logic AND truth table is shown below:

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

Negative logic OR truth table can be obtained by complementing the result of positive logic OR as shown below:

A	B	Y	$\bar{Y}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

6. What is gray code? Explain with example. [2] 2074 Chaitra
Refer to 2076 Chaitra

7. What is the importance of De-Morgan's laws? Show how a two-input OR gate can be constructed from a two-input NAND gate. [4] 2074 Ashwin

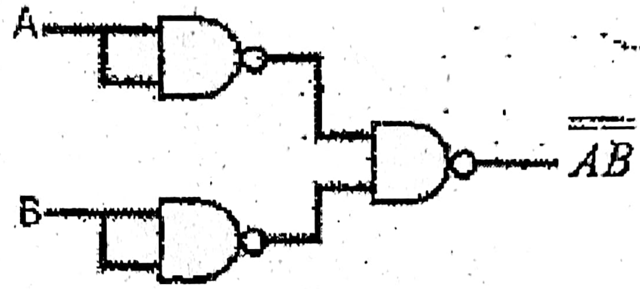
De Morgan's theorems prove very useful for simplifying Boolean logic expressions because of the way they can 'break' an inversion, which could be the complement of a complex Boolean expression.

One important consequence of De-Morgan's laws is that, since all logical operations can be expressed in terms of AND, OR, and NOT, all logical operations can also be expressed in terms of just AND and NOT, or just OR and NOT.

OR gate using NAND Gate:

$$Y = A + B = \overline{\overline{A + B}} = \overline{\bar{A}.\bar{B}}$$

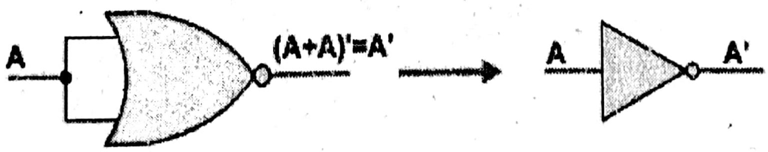
<u>In symbol:</u>



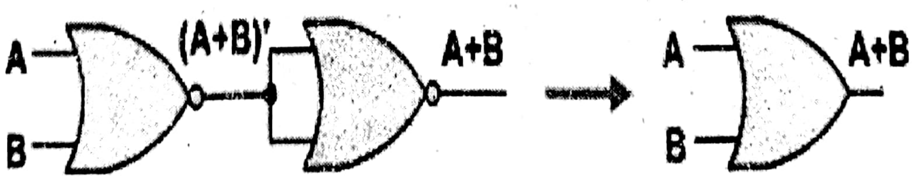
8. Prove that NOR gate is a universal gate. Realize XOR gate using only NAND gate. [6] 2073 Chaitra

A universal gate is a gate which can implement any Boolean function without need to use any other gate type.
To prove that any Boolean function can be implemented using only NOR gates, we will show that the AND, OR, and NOT operations can be performed using only these gates.

Implementing an Inverter Using only NOR Gate
All NOR input pins connect to the input signal A gives an output A'.

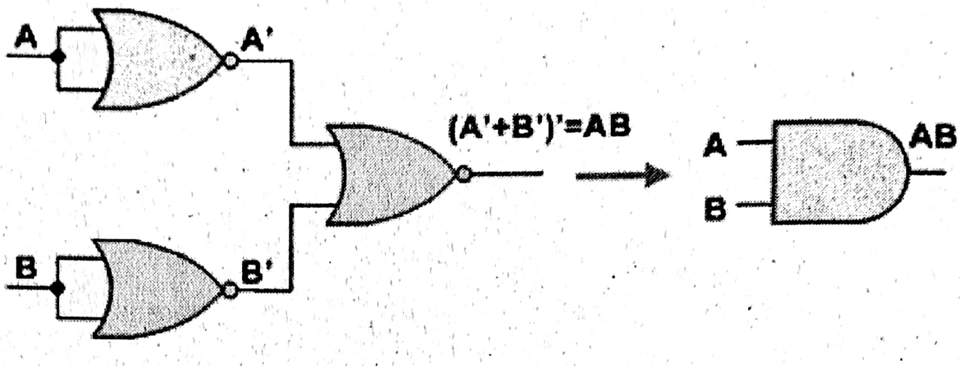


Implementing OR Using only NOR Gates
An OR gate can be replaced by NOR gates as shown in the figure (The OR is replaced by a NOR gate with its output complemented by a NOR gate inverter)



Implementing AND Using only NOR Gates

An AND gate can be replaced by NOR gates as shown in the figure (The AND gate is replaced by a NOR gate with all its inputs complemented by NOR gate inverters)



Thus, the NOR gate is a universal gate since it can implement the AND, OR and NOT functions.

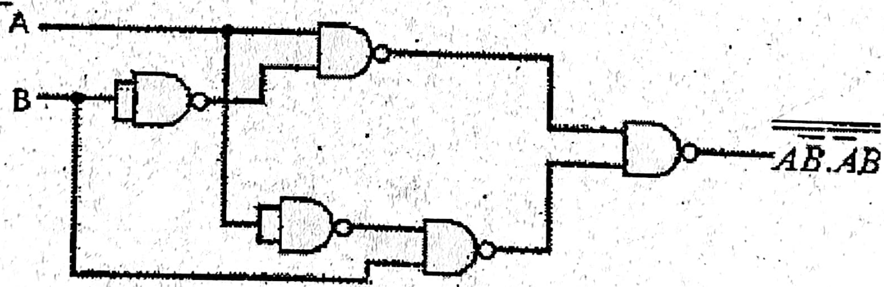
9. Construct two input XOR gate using minimum number of 2-input NAND gates only.

XOR gate using only NAND gate only:

$Y = A \oplus B$

$= \overline{\overline{A\bar{B} + \bar{A}B}} = \overline{\overline{A\bar{B}}.\overline{\bar{A}B}}$

In symbol:



10. Implement Exclusive OR gate by using NAND gates only. [4] 2072 Chaitra

XOR gate using only NAND gate only:

$Y = A \oplus B$

$= \overline{\overline{A\bar{B} + \bar{A}B}} = \overline{\overline{A\bar{B}}.\overline{\bar{A}B}}$

In symbol: