Chapter-2

Data Representation and Boolean logic

Number System:-

Number system are the technique to represent numbers in the computer system architecture. Every number system includes a set of unique characters or literals. It is a way to represent a number in different forms.

→ In digital representation, the number systems are classified into four fundamental types:

- Number Systems
  - → Decimal Number System
  - → Binary Number System
  - → Octal Number System
  - → Hexadecimal Number System

Decimal number system:-

Decimal number system has ten (10) digits ranging from 0 to 9 with base value 10. Every number can be represented with 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9 in this number system. Each value in this number system has the positional value expressed as a power of 10.

Page No.
Date : / /20

Binary number system:-

One Binary chip can have millions of transistors activated by the electronic signals and continually switching between ON and OFF which is represented using two digits, 1 and 0, respectively. These digits are called binary number or machine code.

Octal number system:-

Octal number system has only eight (8) digits from 0 to 7. Every number can be represented with 0, 1, 2, 3, 4, 5, 6, and 7 in this number system. The base of octal number system is 8 because it has only 8 digits.

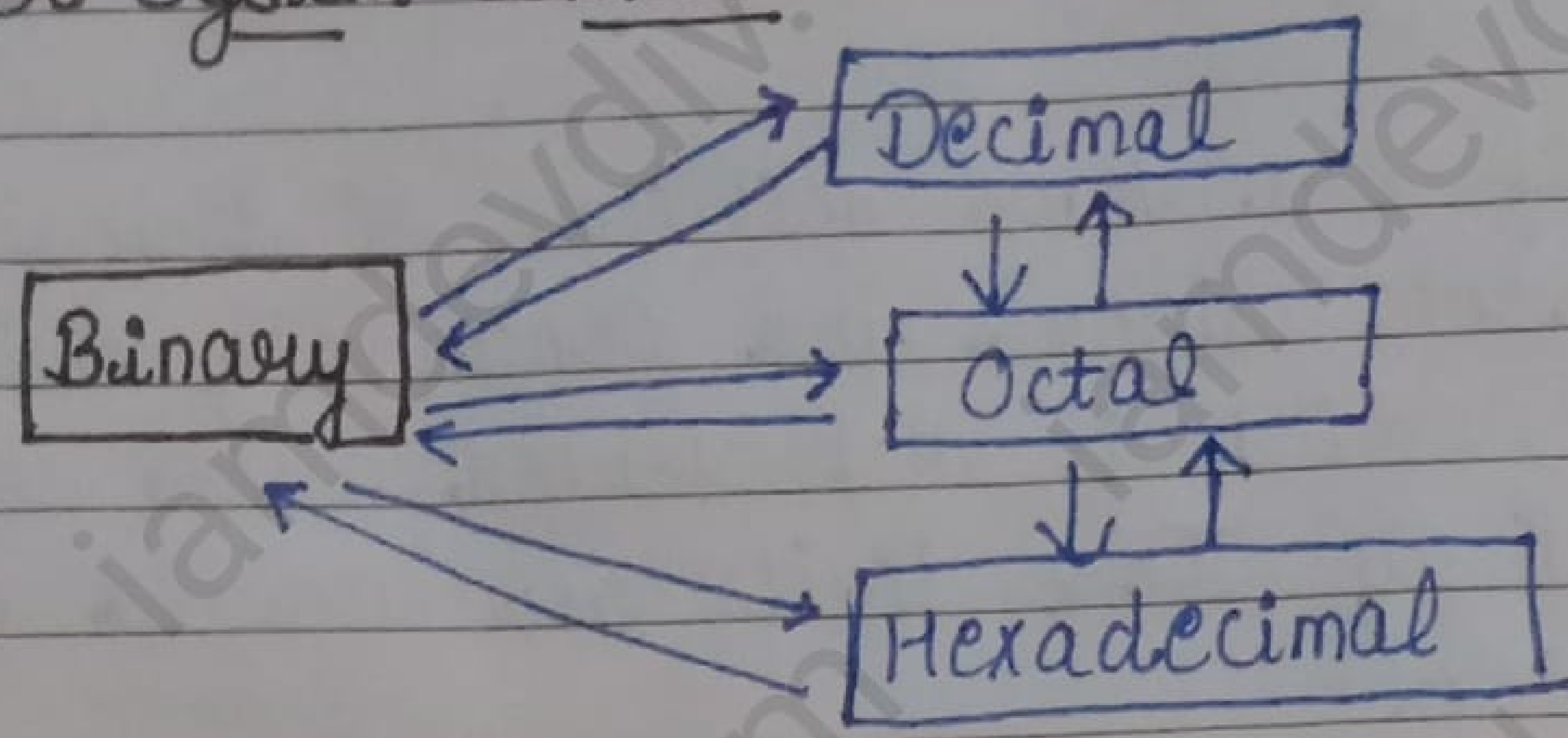
Hexadecimal number system:-

A hexadecimal number system has 16 alphanumeric values from 0 to 9 and A to F. Every number can be represented with 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F in this number system.

* • The left-most bit of a number having the greatest value is known as Most Significant Bit (MSB).

• The right-most bit of a number having the lowest value is known as Least Significant Bit (LSB). It is same for all number systems.

* Number System Conversions



→ Decimal number system to Other Base

- Decimal number system to Binary number System.
- Decimal number system to Octal number system.
- Decimal number system to Hexadecimal number system.

→ other Base to Decimal Number System

- Binary Number system to Decimal number system
- Octal Number system to Decimal number system.
- Hexadecimal Number system to Decimal number system.

→ One Base to Other Base system

- Binary Number system to Hexadecimal Number System.
- Hexadecimal number system to Binary Number system.
- Binary number system to octal number system.
- Octal Number system to Binary number system.
- Hexadecimal Number system to Octal Number system.
- Octal Number system to Hexadecimal number system.

→ Learning Tip

To convert decimal number into other number system, we write all remainder values moving from bottom to top.

Examples:-

→ Convert the Decimal number $(125)_{10}$ into its Binary equivalent

2	125	Remainder
2	62	1
2	31	0
2	15	1
2	7	1
2	3	1
2	1	1
	0	1

LSB

↑

MSB

$(125)_{10} = (1111101)_2$

- Convert $(95)_{10} = (\ )_2$

2	95	Remainder
2	47	1
2	23	1
2	11	1
2	5	1
2	2	1
2	1	0
	0	1

LSB

↑

MSB

$(95)_{10} = (1011111)_2$

→ Convert $(105.15)_{10}$ into binary

2	105	Remainder
2	52	1
2	26	0
2	13	0
2	6	1
2	3	0
2	1	1
	0	1

LSB ↑ MSB

The binary equivalent of integer part $(105)_{10} = (1101001)_2$

Now, let us convert $(0.15)_{10}$

	Integer	fraction
Multiply 0.15 by 2	0	.30
Multiply 0.30 by 2	0	.60
Multiply 0.60 by 2	1	.20
Multiply 0.20 by 2	0	.40
Multiply 0.40 by 2	0	.80
Multiply 0.80 by 2	1	.60

MSB ↓ LSB

$(0.15) = (0.001001)_2$

final result = (105.15)

* Internal storage Encoding of Characters

The data in computers is stored in a binary form. The data can be numeric data or non-numeric data, which can be alphabets or any other special characters. These non-numeric data are said to be alphanumeric data. Therefore, when a character, letter or symbol is pressed on the keyboard, it is internally mapped to a unique code which is further converted to binary.

- The various encoding schemes for data representation are:

→ ASCII (American Standard Code for Information Interchange)
→ ISCII (Indian Standard code for Information Interchange)
→ UNICODE (Universal Coding standard)

→ ASCII (American Standard Code for Information Interchange)

ASCII is the most widely-used alphanumeric code which is used in computers to translate text (letters, numbers, and symbols) into a form that can be sent to and understood by other computers and devices such as modems and printers.
The standard ASCII characters set uses just 7 bits for each character and so it has $2^7 = 128$ possible code groups.

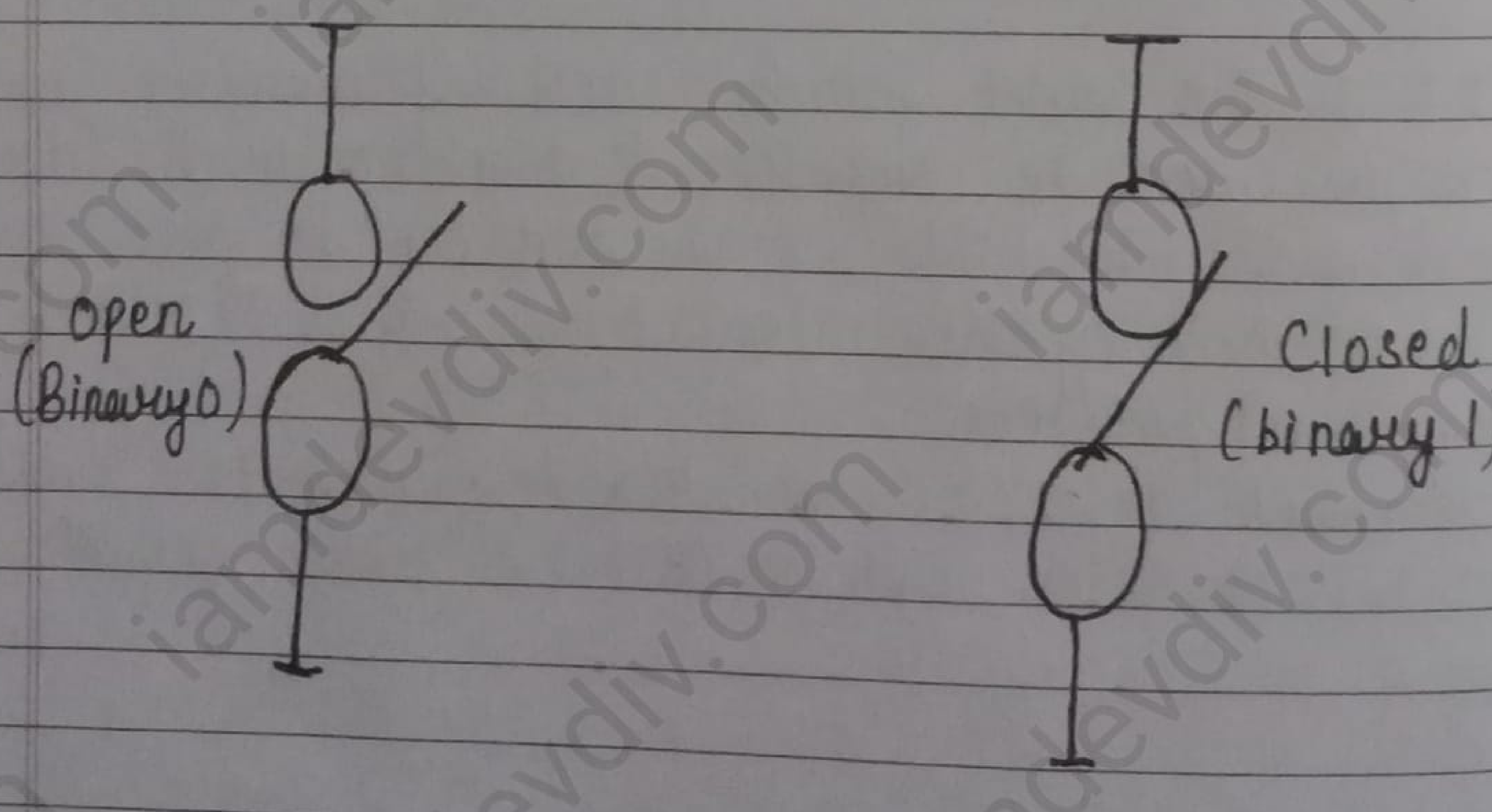
Page No
Date / /20

→ ISCII (Indian Standard Code for Information Interchange)

In 1991, the Bureau of Indian Standards adopted the ISCII. It is an 8-bit code which allows English and Indian scripts' alphabets to be used simultaneously. It is an 8-bit code capable of representing 256 characters. There are 22 officially recognized languages in India: Hindi, Marathi, Sanskrit, Punjab, Gujarati, Oriya, Bengali, etc.

→ Unicode (for Multilingual Computing)

Unicode is a new universal coding standard adopted by all new platforms. It is promoted by Unicode consortium which is a non-profit organization. Unicode provides a unique number for every character irrespective of the platform, program and the language.



→ **Boolean Variable:-**

A Boolean variable is also known as binary variable or logical variable that takes its values from Boolean algebra. A Boolean variable can take only one binary-valued quantity out of the two possible value i.e. Yes/ No, 1/0, true/false.

→ **Boolean Constant:-**

The values which are stored in binary variable are known as Boolean constants. Therefore, the values true/ false, Yes/No or 1/0 are boolean constant.

- **Boolean variable:-**

A variable which holds false/true value is known as Boolean variable (X, Y, Z, etc.)

- **Boolean Expression:-**

A meaningful combination of Boolean operators (AND/OR/NOT), Boolean operand / variable (X, Y, Z, etc.) and Boolean Constant (0 or 1) is known as Boolean Expression (Logical expression)

For ex:- (i) $X + Y . Z$

(ii) $A . (B + C) + B . C'$

- **Boolean operators**

Operators used in Boolean algebra are known as Boolean operators

Examples:-

AND	OR	NOT
$.$	$+$	$!$
$\wedge$	$\vee$	$\neg$
$X . Y$	$X + Y$	$!X$
$X \wedge Y$	$X \vee Y$	$\bar{X}$

- **AND Operator**
AND operator is a binary operator that operates on two variables and the operation performed by AND operator is known as logical multiplication. The symbol used for logical multiplication is dot (.) operator.

- **OR operator**
The OR operator is a binary operator that operates on two variables and the operation performed by OR operator is known as logical addition. The symbol used for logical addition is plus (+) operator

- **NOT operator**
The NOT operator is a unary operator that operates on one variable and the operation performed by NOT operator is known as negation or complementation.

* Truth Table:-

A truth table is a representation of a Boolean function or expression containing all possible combinations of input values and their result in a tabular format.

→ If the result of a logical expression or logical statement is always true or 1, it is known as Tautology

→ If the result is always false or 0, it is known as Falla

★ The number of rows in a truth table is calculated as 2^n where n is total of Boolean variables, therefore if their are two variables A & B.

$2^2 = 4$ combinations

→ **Logic Circuit**

A Logic circuit is the one which takes one or more inputs and generates an output. Logic Gates use the binary operators AND, OR and NOT.

- There are three fundamental Logic Gates which are as follows:-

1. **NOT Gate:-**
A NOT gate, sometimes called a logical inverter gate to differentiate it from other types of electronic inverter devices, has only one unit. It reverses the logical state. The output state is a negation or complement of an input signal.

2. **AND Gate:-**
The AND Gate is so named because if 0 is called "false" and 1 is called "true", the gate acts in the same way as the logical "AND" operator. The output is "true" when the both the inputs are "true". Otherwise the output is "false". AND gate works on two or more inputs which result in a single output.

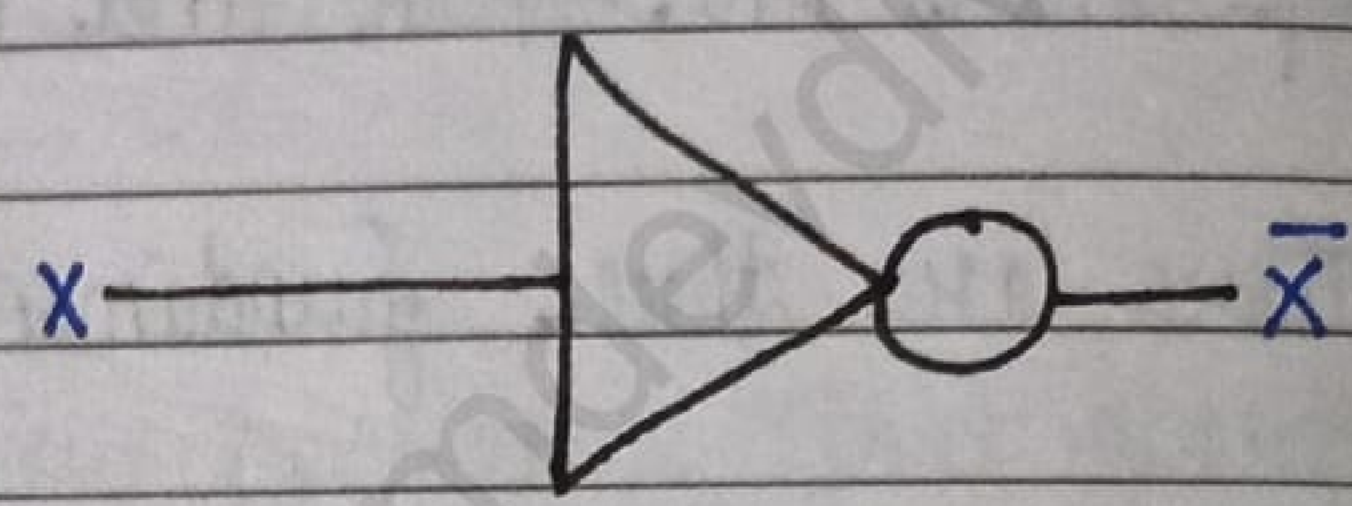
3. **OR Gate:-**
The OR gate gets its name from the fact that it behaves like logical inclusive "OR". The output is "true" if either or both of the inputs are "true". If both inputs are "false", then the output is "false". OR gate works on two or more inputs which results in a single output.

→ The ~~Not~~ NAND and NOR gates are called universal gates because any digital circuits can be implemented using these gates.

Operator | Truth Table | Logic gate

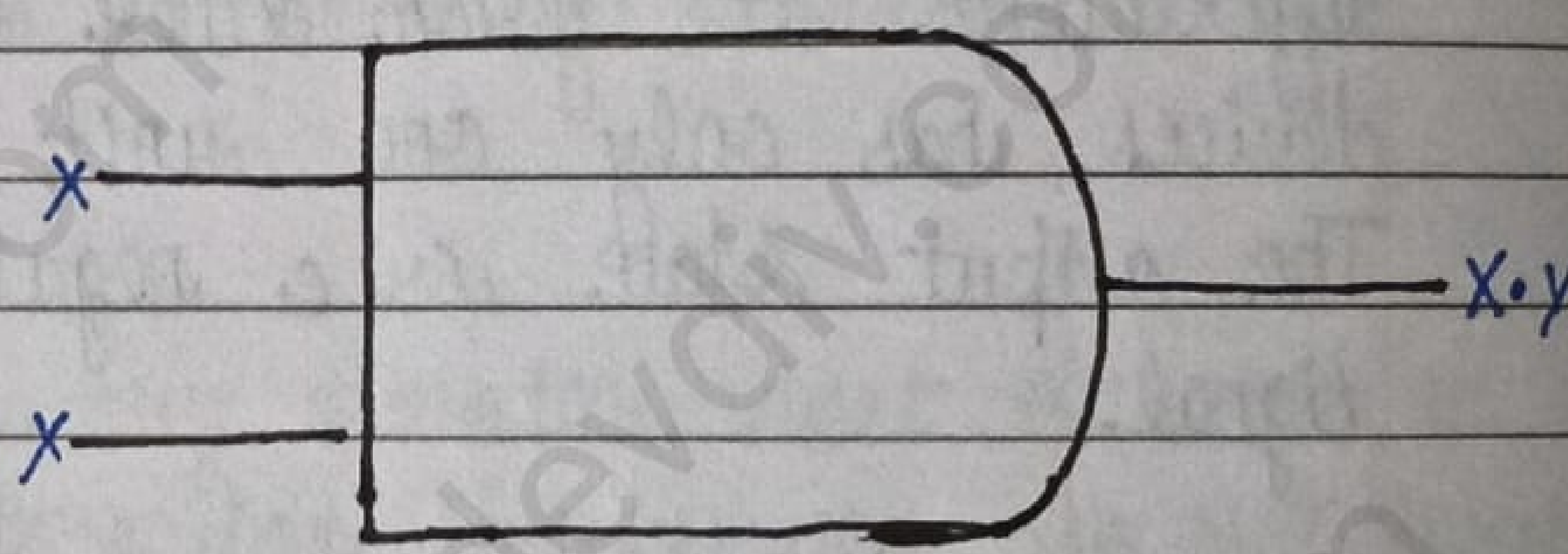
NOT

X	X'
0	1
1	0



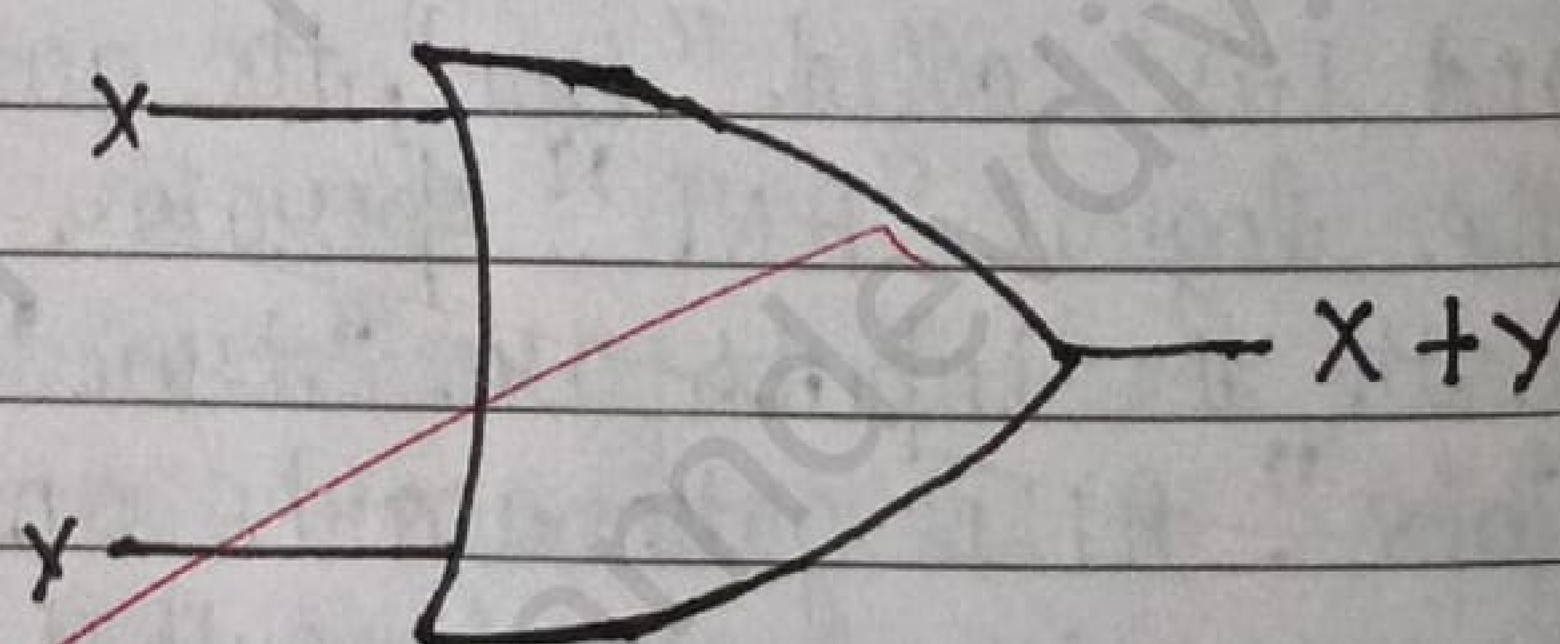
AND

X	Y	X·Y
0	0	0
0	1	0
1	0	0
1	1	1



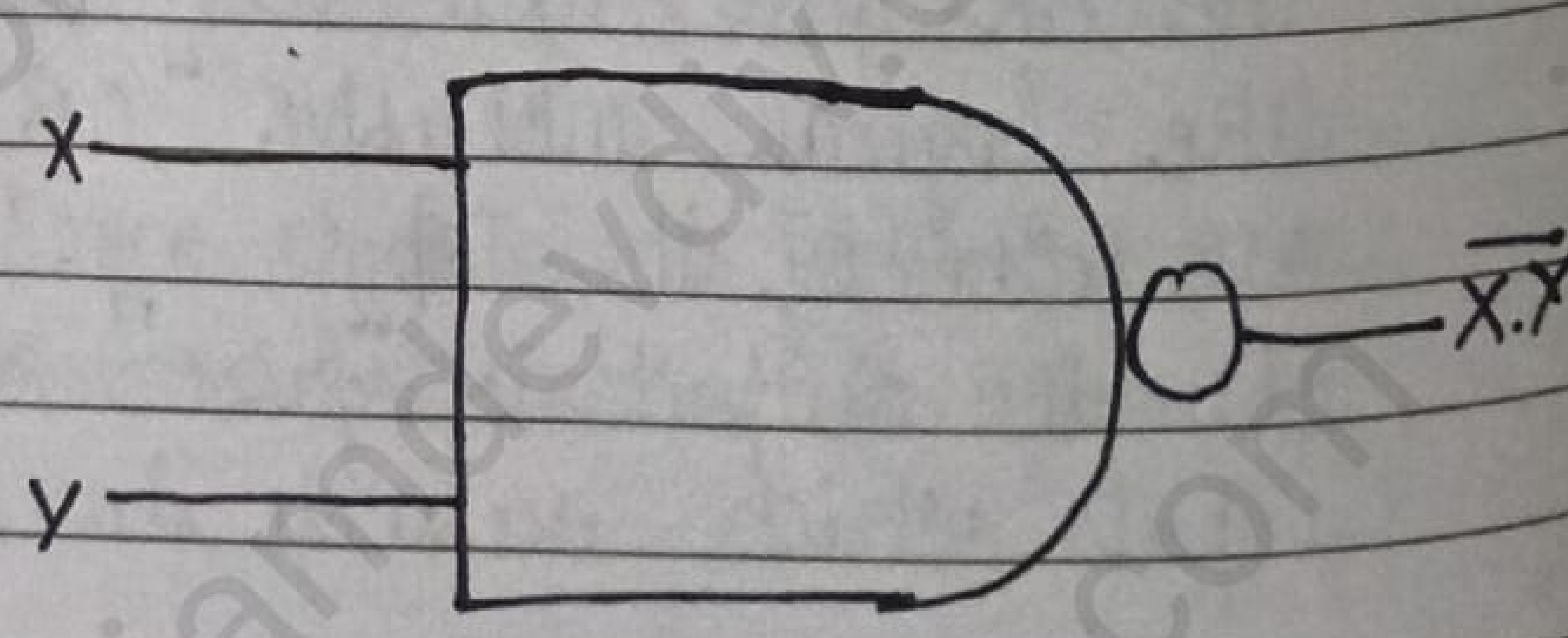
OR

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1



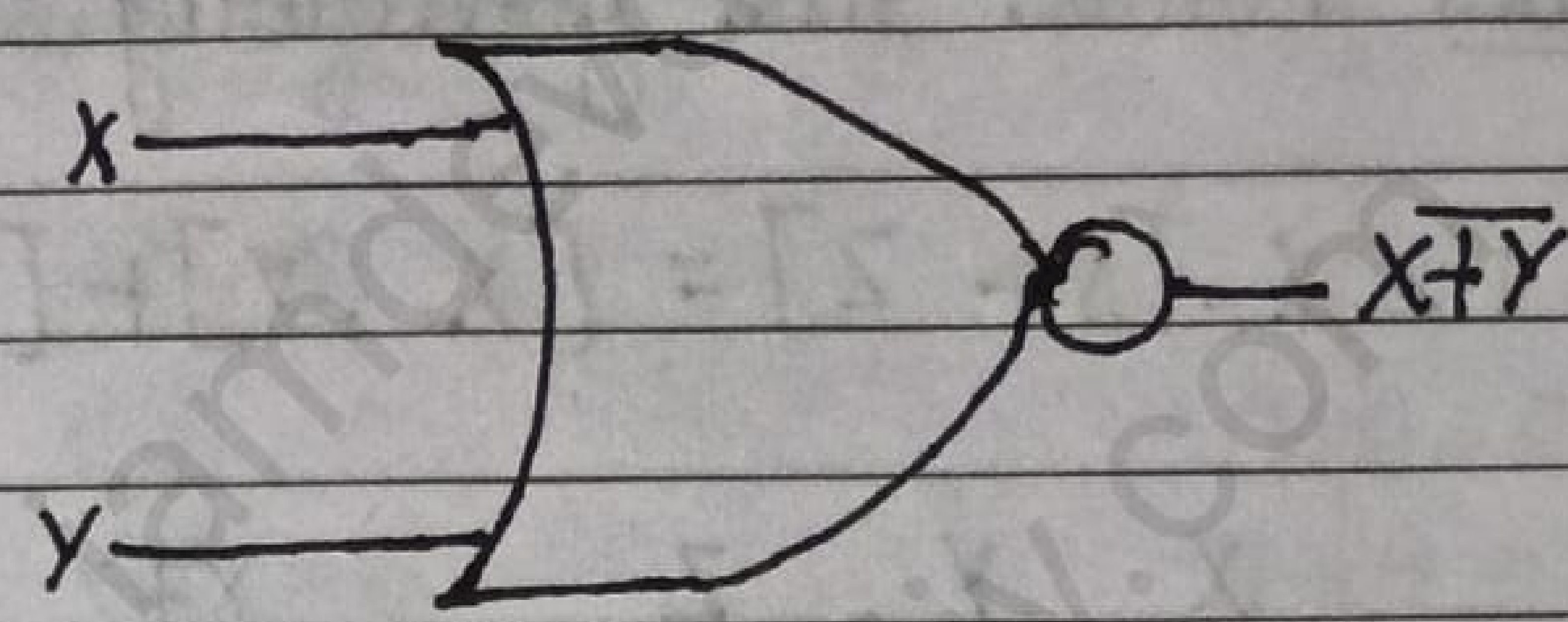
NAND

X	Y	X·Y	(X.Y)'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0



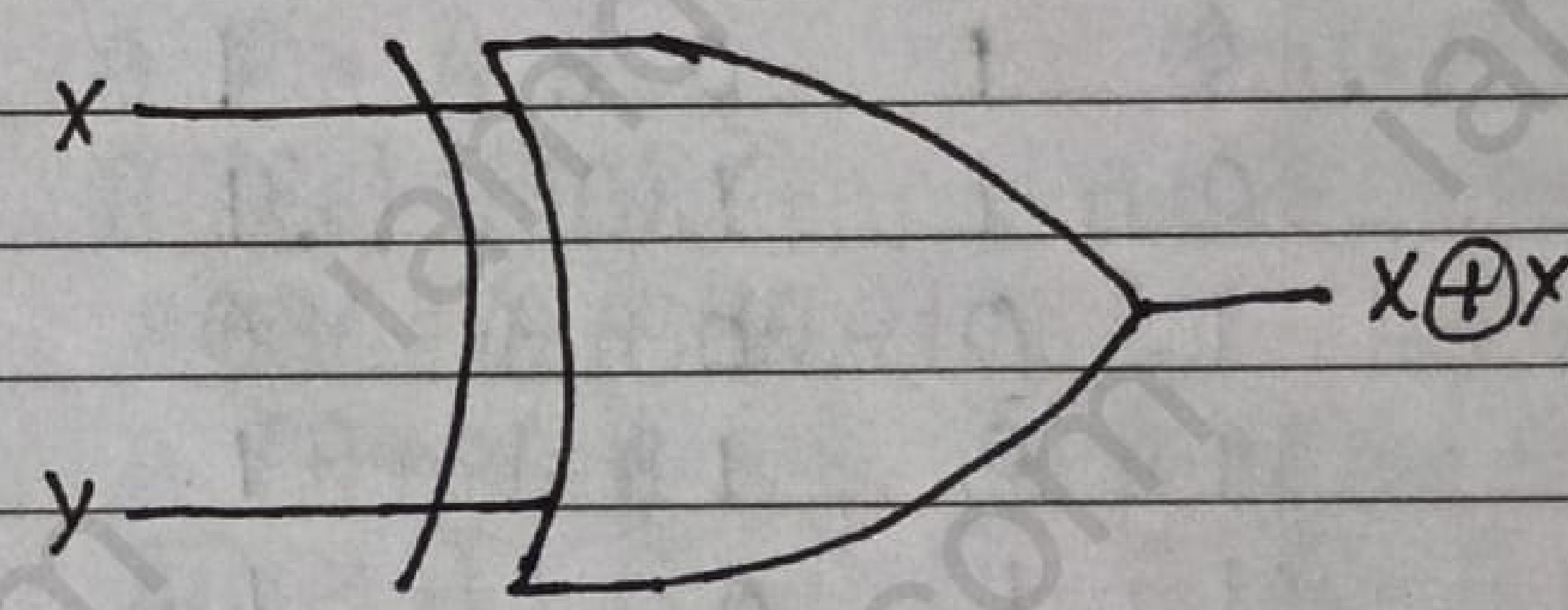
NOR

X	Y	X+Y	$\overline{(X+Y)}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0



XOR

X	Y	$X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0



→ Duality principle in Boolean Algebra

This law explains that replacing the variables doesn't change the Boolean function. The dual of the expression can be achieved by

- Replacing each AND [.] sign with an OR [+] sign.
- Replacing each OR [+] sign with and AND [.] sign.
- Replacing each variables, such as replacing 1 with 0 and replacing 0 with 1

If the operators and variables of an equation or function produce no change in the output of the equation, though they are interchanged, it is called "Duals".

Ques- Prove the following with the help of a truth table.

1. $X.[Y+Z] = [X.Y]+[X.Z]$

X	Y	Z	X+Z	X.(Y+Z)	X.Y	X.Z	[X.Y]+[X.Z
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

Ques- Write a truth table for the Boolean Expression $f = A.B' + C$ by writing each step.

A	B	C	B'	A.B'	A.B'+C
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	0	0	0
1	1	1	0	0	1

Evergreen
Page No.
Date: / /20

→ Some rules/ Laws of Boolean Algebra

	Law Name	AND	OR
1	Property of 0	$0.A = 0$	$0+A = A$
2	Property of 1	$1.A = A$	$1+A = 1$
3	Idempotent Law	$A.A = A$	$A+A = A$
4	Inverse/ Complement Law	$A.A' = 0$	$A+A' = 1$
5	~~Associ~~ Commutative Law	$A.B = B.A$	$A+B = B+A$
6	Associative Law	$A.(BC) = (AB).C$	$A+(B+C) = (A+B)+C$
7	Distributive Law	$A+BC = (A+B)(A+C)$	$A(B+C) = AB+AC$
8	Absorption Law	$A.(A+B) = A$	$A+AB = A$
9	De Morgan's Law	$\overline{AB} = \bar{A}+\bar{B}$	$\overline{A+B} = \bar{A}.\bar{B}$

* DE MORGAN'S LAW

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

Let $P = (A+B)$

So, $P + \bar{P} = 1$

$(A+B) + (\overline{A+B}) = 1$

L.H.S. $(A+B) + (\overline{A+B})$

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

$= (A+B+\bar{A})\ (A+B+\bar{B})$

$= (A+\bar{A}+B)\ (A+B+\bar{B})$

$= (1+B)(A+1)$

$= (1).(1)$

$= 1$

L.H.S. = R.H.S.

* Proof Using truth table :-

A	B	A'	B'	$A+B$	$(A+B)'$ ~~$(A.B)'$~~	$A'.B'$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

Fill in the Blanks

a. In Binary number system, the left - most bit is called the ___Most___ significant bit

b. Base of a number is also known as ___Radix___ of a number.

c. The decimal system is composed of ___10___ symbols

d. ___ASCII___ code is the most widely used alphanumeric code which is used in computers to translate text (letters, numbers and symbols).

e. Unicode is a new universal coding standard adopted by all new ___platforms___.

f. The ___AND___ operation is Boolean multiplication and the ___OR___ operation is Boolean addition.

g. A statement is said to be a ___Boolean / logical statement___ if it has a definite value, which is either True or False.

(h) NOT gate has only one input and it complements an ~~that~~ input signal.

(i) The values which are stored in binary variables are known as Boolean constants.

(j) A Truth table is a representation of a Boolean function or expression containing all possible combinations of input values and their result in a tabular format.

→ Multiple Choice Questions:-

(a) What is information?
Ans Processed data

(b) What is the base of binary number system?
Ans 2

(c) What is the base of decimal number system?
Ans 10

(d) What is the base of octal number system?
Ans 8

(e) Which of the following is binary equivalent of $(43)_{10}$?

$(101011)_2$

(f) Which of the following is decimal equivalent of $(10010)_2$?
18

(g) What is the full form of ISCII?
Indian Standard Code for Information Interchange

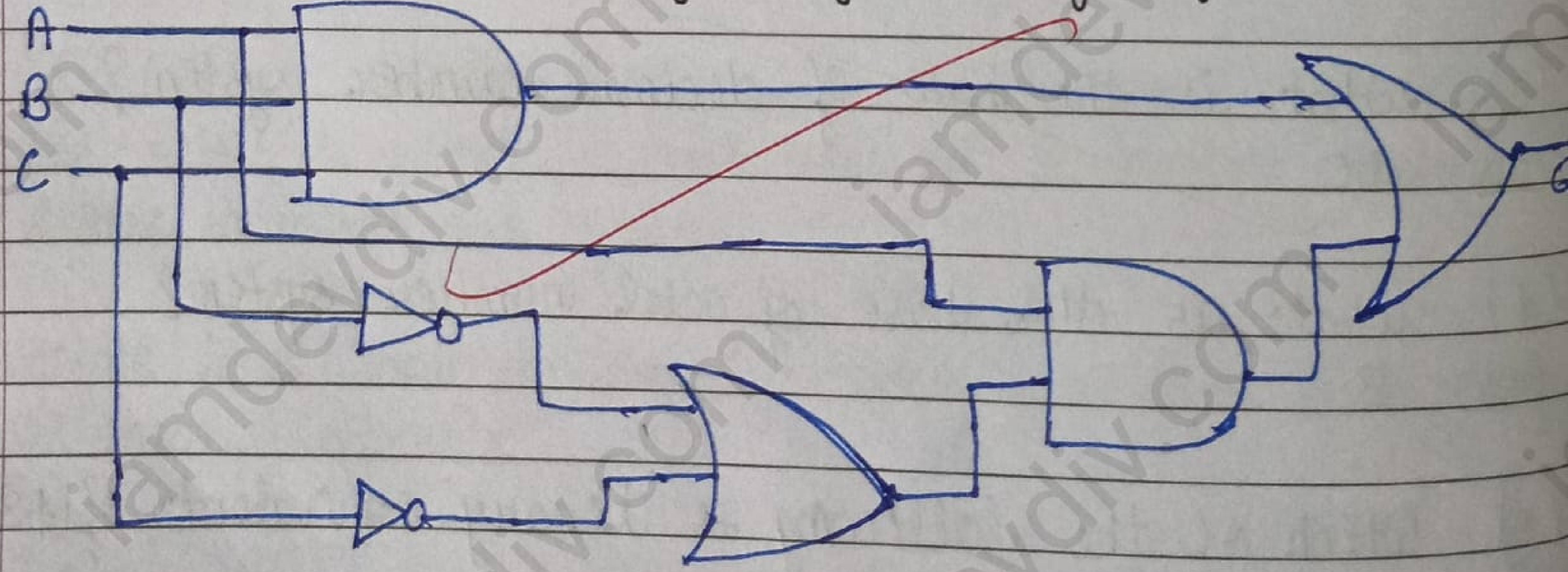
(h) Hexadecimal number system is composed of?
16 symbols

(i) Which of the following is not a binary number?
11E

(j) Which numbering system uses numbers and letters as symbols
Hexadecimal

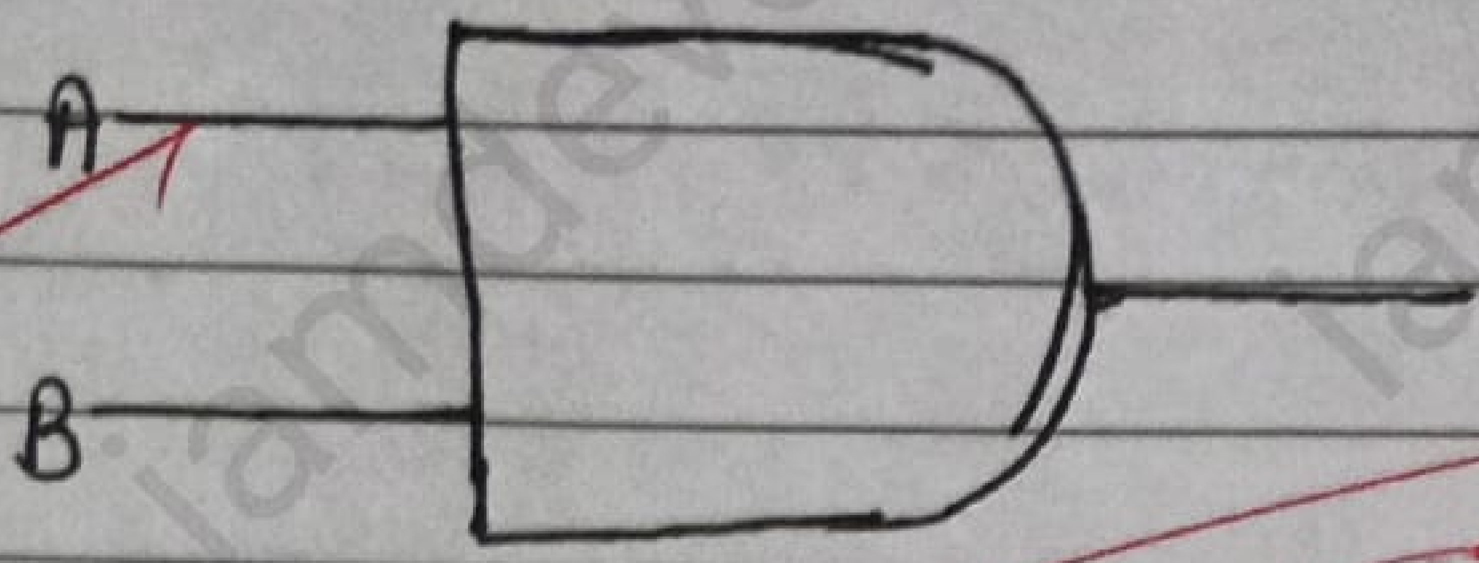
(k) Identify the logical statement:?
Should I wear the mask or not?

(l) The boolean expression of the following logic circuit is:



$A \cdot B \cdot C + (B' + C') \cdot A$

(m) The following is a AND Logical Gate.



Ans AND

6/7/23

25/8/22

- **BOOLEAN LOGIC PRACTICE QUESTIONS**

AND

A	B	A . B
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

NOT

A	A
0	1
1	0

SOLVE $x + yz$

x	y	z	y.z	x + yz
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

SOLVE

(i) $X \cdot Y' + Z$

X	Y	Z	Y'	X.Y'	X.Y' + Z
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	0	0	0
1	1	1	0	0	1

(ii) $A + B \cdot C' + A' \cdot C'$

A	B	C	A'	C'	B.C'	A'.C'
0	0	0	1	1	0	1
0	0	1	1	0	0	0
0	1	0	1	1	1	1
0	1	1	1	0	0	0
1	0	0	0	1	0	0
1	0	1	0	0	0	0
1	1	0	0	1	1	0
1	1	1	0	0	0	0

$A + B \cdot C' + A' \cdot C'$
1
0
1
0
1
1
1
1

(iii) $(A+B+C)'$

A	B	C	A + B + C	$(A+B+C)'$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(iv) $(X+Y)' . (X+Z)'$

X	Y	Z	$(X+Y)$	$(X+Y)'$	$(X+Z)$	$(X+Z)'$
0	0	0	0	1	0	1
0	0	1	0	10	1	0
0	1	0	1	0	0	1
0	1	1	1	0	1	0
1	0	0	1	0	1	0
1	0	1	1	0	1	0
1	1	0	1	0	1	0
1	1	1	1	0	1	0

$(X+Y)' . (X+Z)'$
1
0
0
0
0
0
0
0

(v) $(x'+y').(x'.z')$

(v)

x	y	z	x'	y'	$(x'+y')$	z'	$(x'.z')$
0	0	0	1	1	1	1	1
0	0	1	1	1	1	0	0
0	1	0	1	0	1	1	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	1	0
1	0	1	0	1	1	0	0
1	1	0	0	0	0	1	0
1	1	1	0	0	0	0	0

$(x'+y').(x'.z')$
1
0
1
0
0
0
0
0

(vi) $(x+x+z')'$

(vi)

x	y	z	z'	$(x+y+z')$	$(x+y+z')'$
0	0	0	1	1	0
0	0	1	0	0	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	0	1	0
1	1	0	1	1	0
1	1	1	0	1	0

(vii) $(\overline{X+Y}) \cdot Z$

(vii)

X	Y	Z	X + Y	$(\overline{X+Y})$	$(\overline{X+Y}) \cdot Z$
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	0	0

(viii) $(\overline{X} + Y + Z) \cdot \overline{Y} \cdot \overline{Z}$

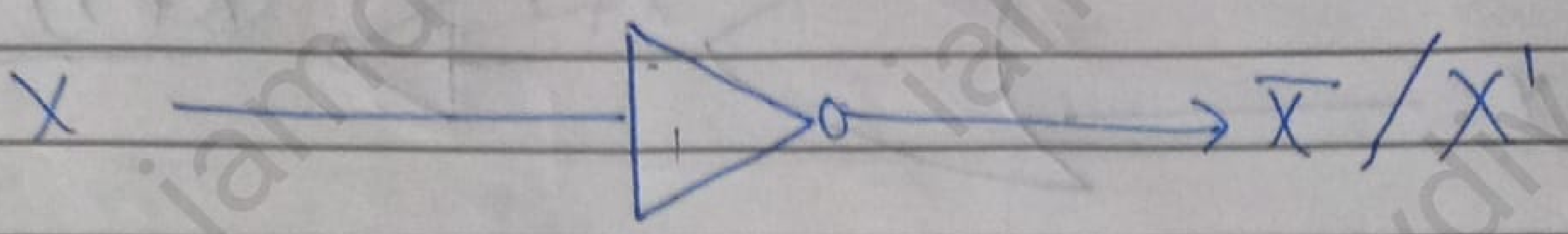
(viii)

X	Y	Z	$\overline{X}$	$(\overline{X}+Y+Z)$	$\overline{Y}$	$\overline{Z}$	$\overline{Y} \cdot \overline{Z}$
0	0	0	1	1	1	1	1
0	0	1	1	1	1	0	0
0	1	0	1	1	0	1	0
0	1	1	1	1	0	0	0
1	0	0	0	0	1	1	1
1	0	1	0	1	1	0	0
1	1	0	0	1	0	1	0
1	1	1	0	1	0	0	0

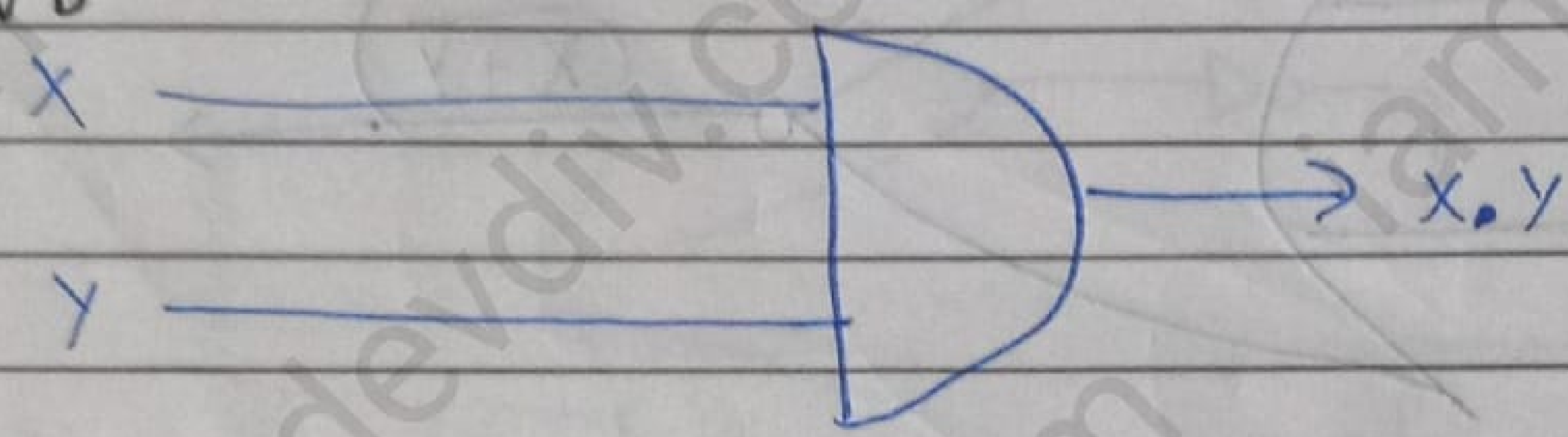
$(\overline{X}+Y+Z) \cdot \overline{Y} \cdot \overline{Z}$
1
0
0
0
0
0
0
0

✱ LOGIC CIRCUIT

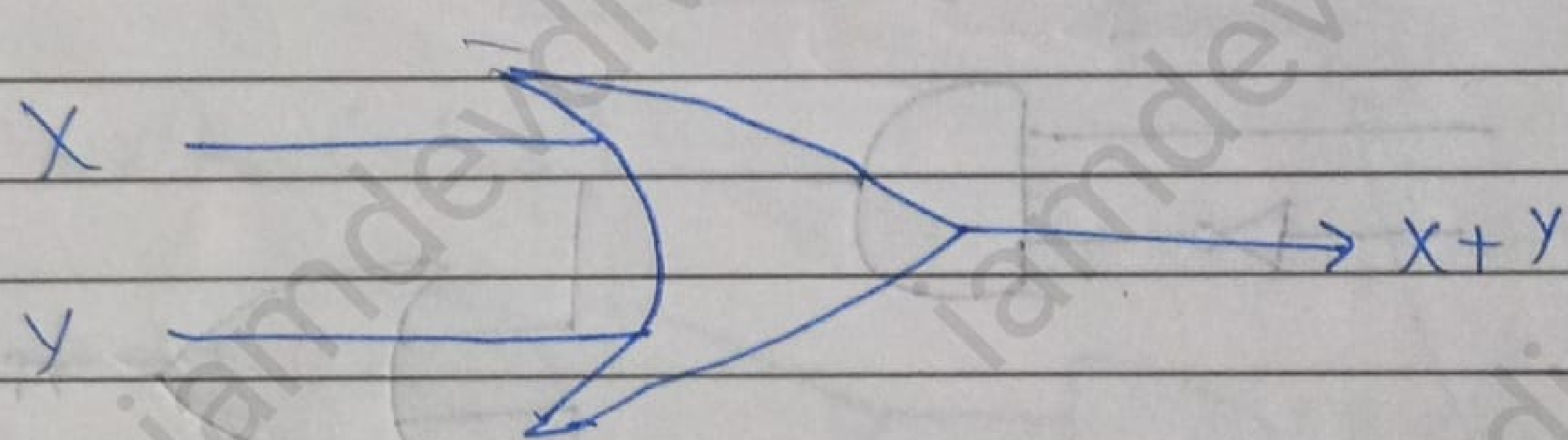
• NOT



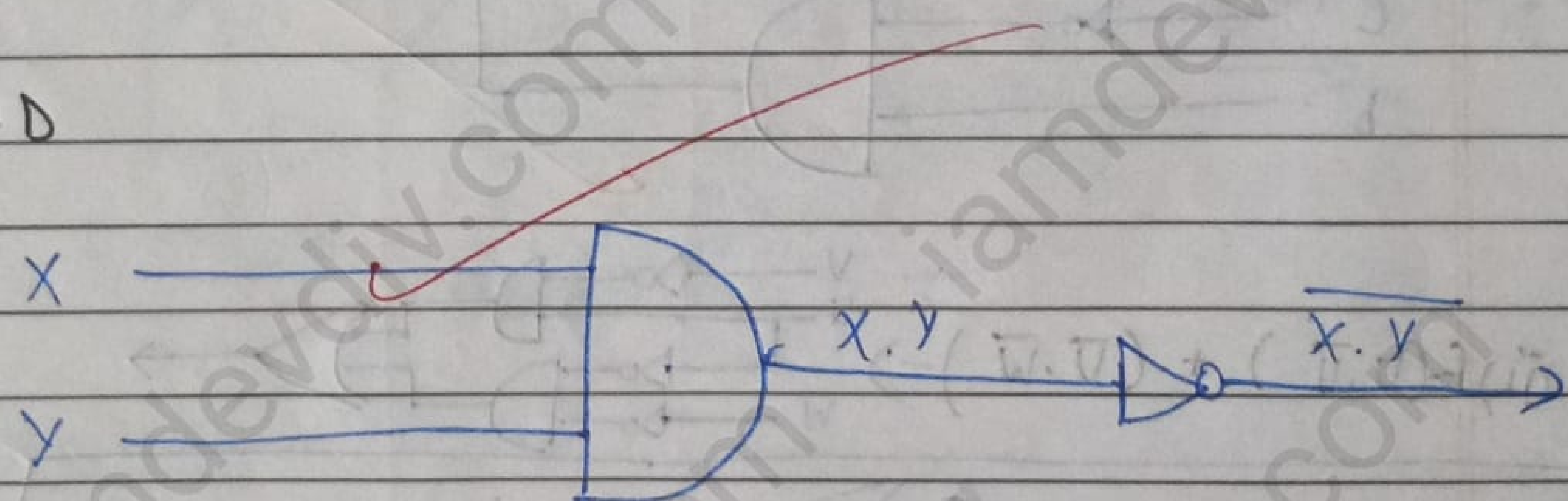
• AND

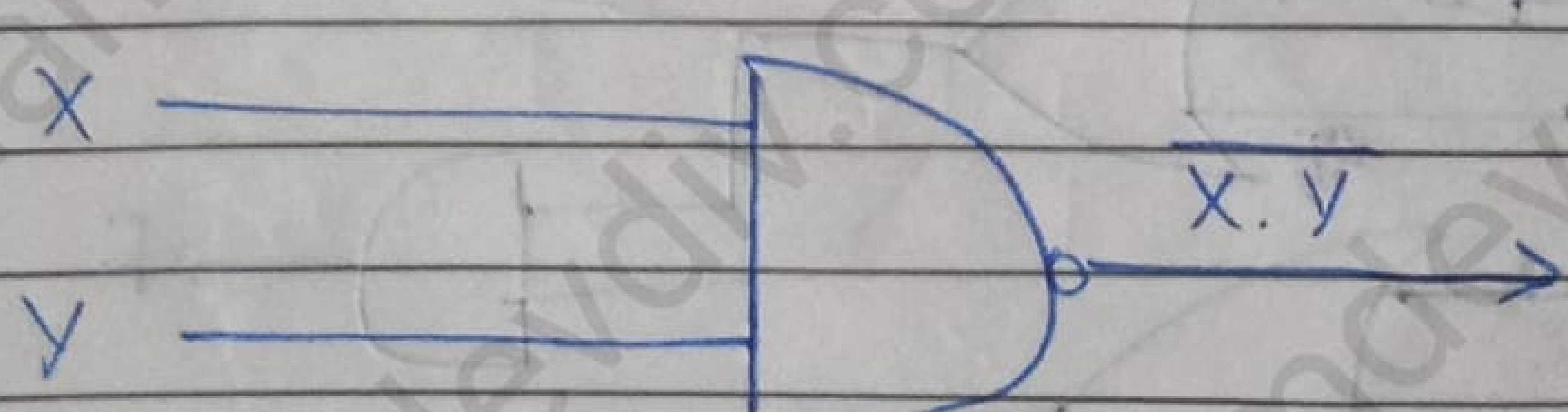


• OR

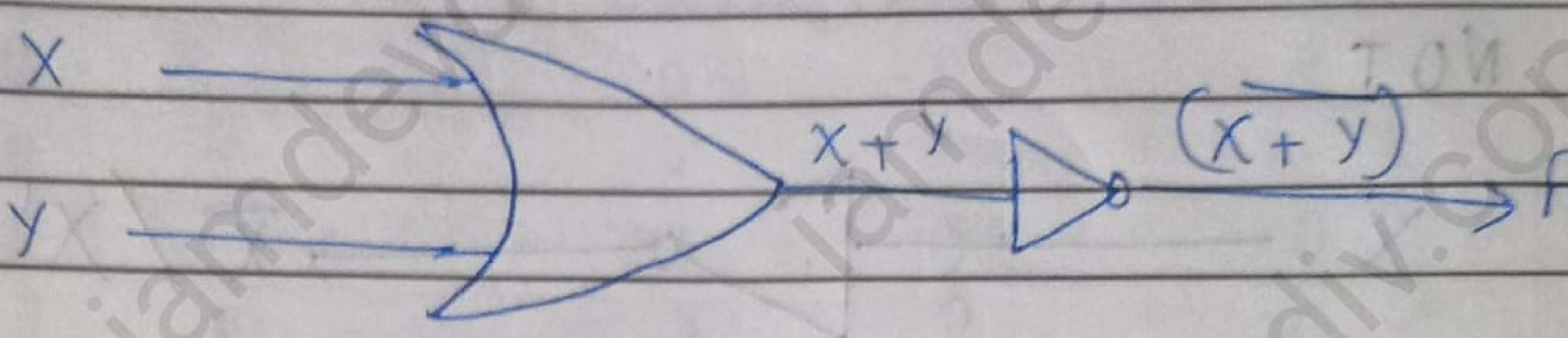


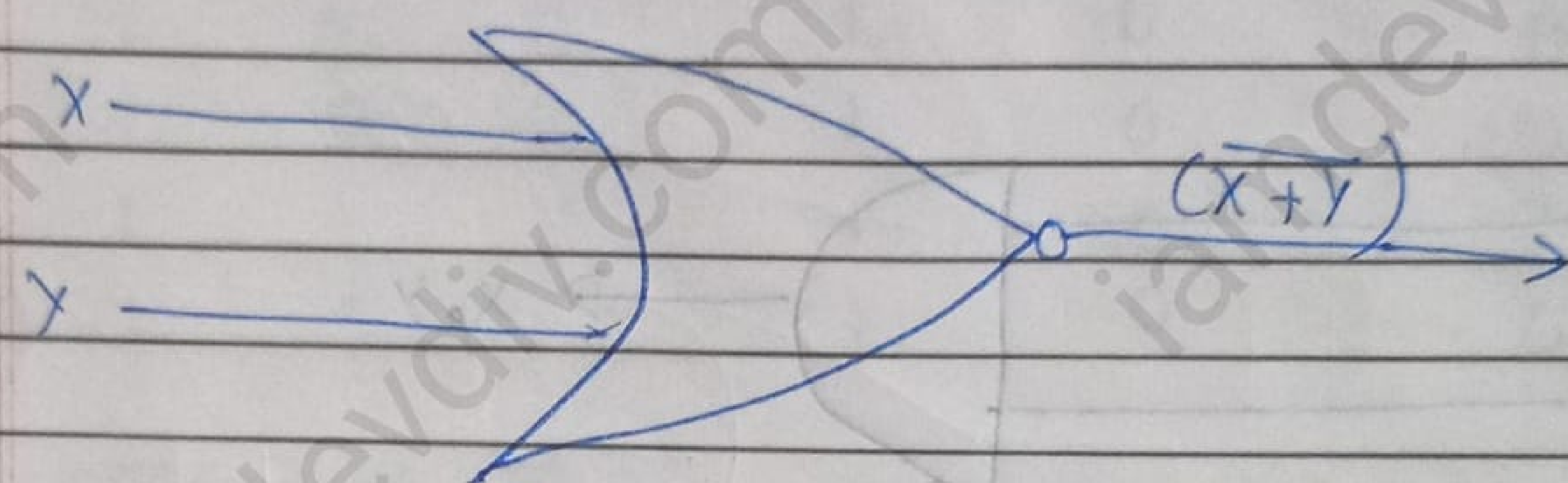
• NAND



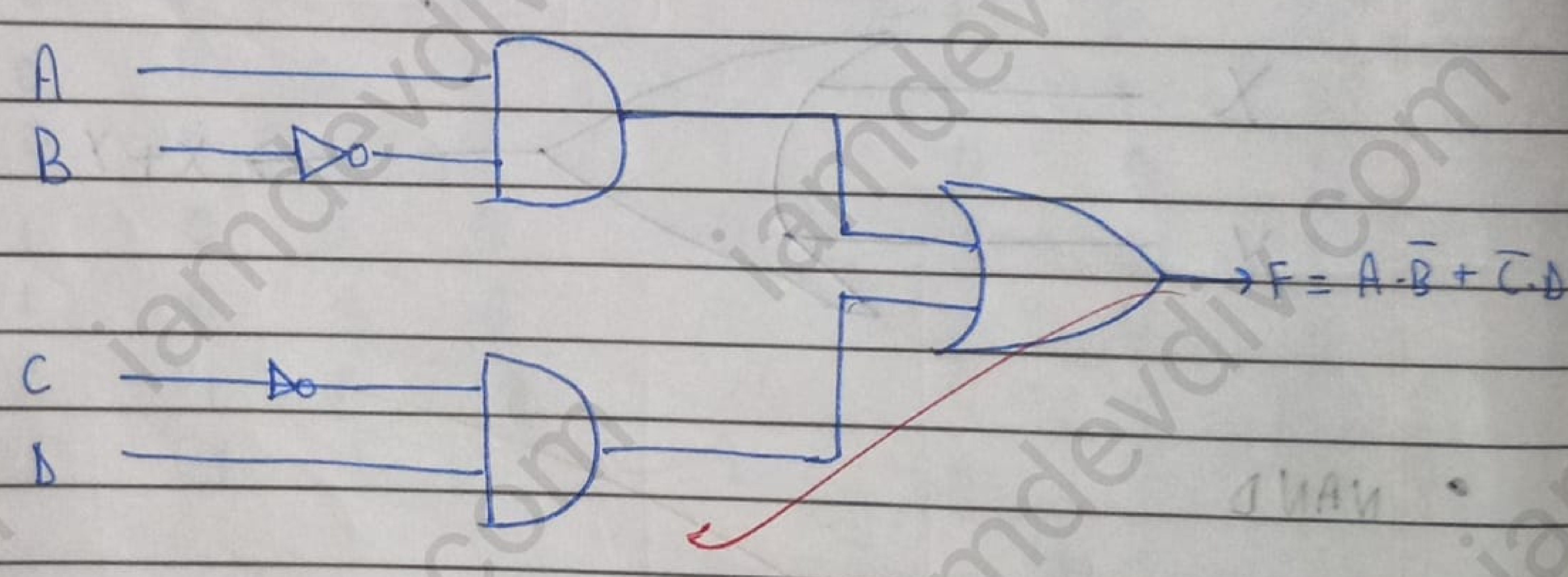


- NOR

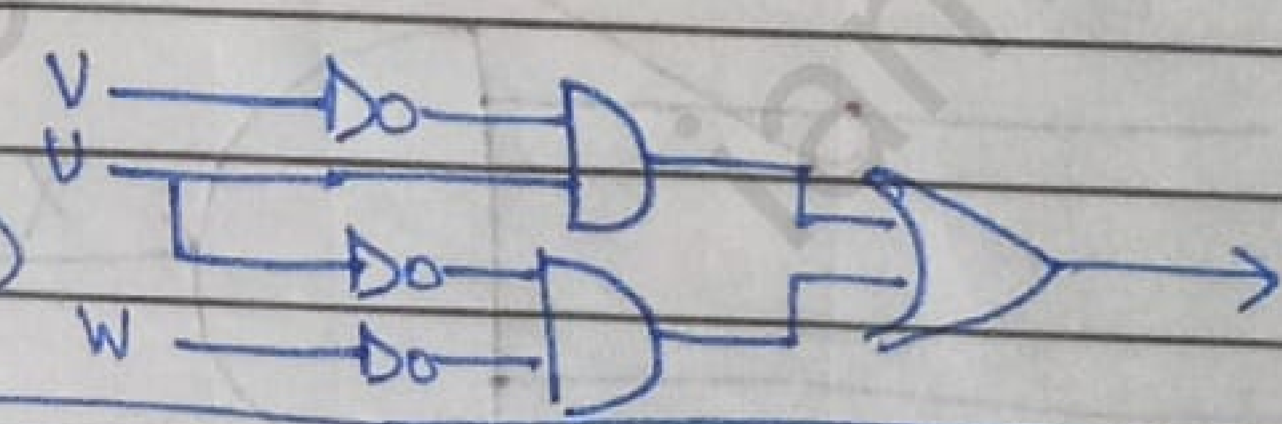




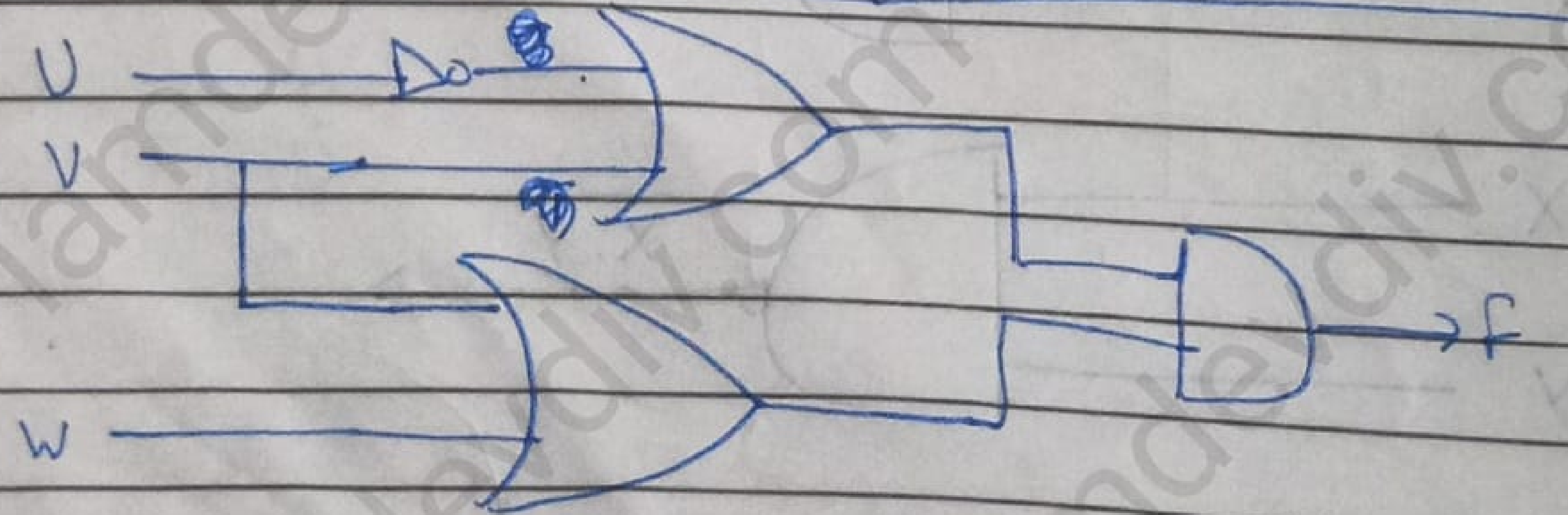
(i) $F = A \cdot \overline{B} + \overline{C} \cdot D$



(ii) $F = (U \cdot \overline{V}) + (\overline{U} \cdot \overline{W}) \Rightarrow$



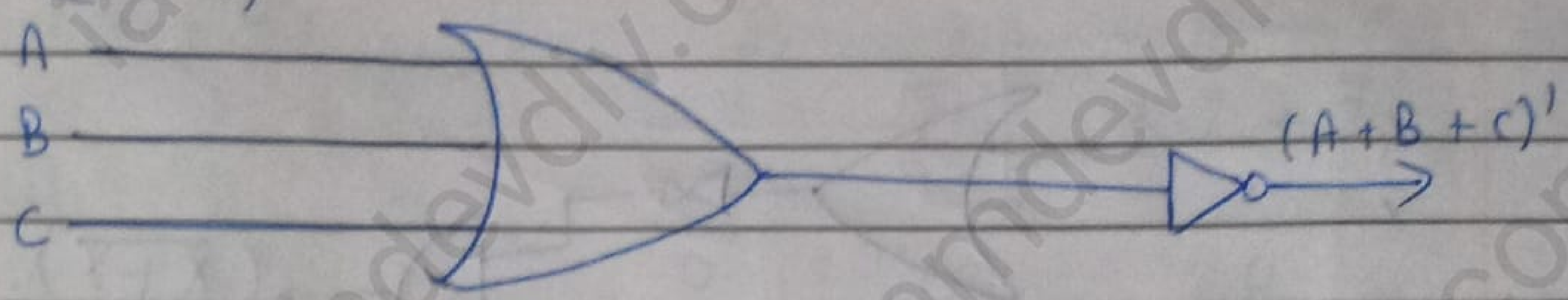
#



$F = (\overline{U} + V) \cdot (V + W)$

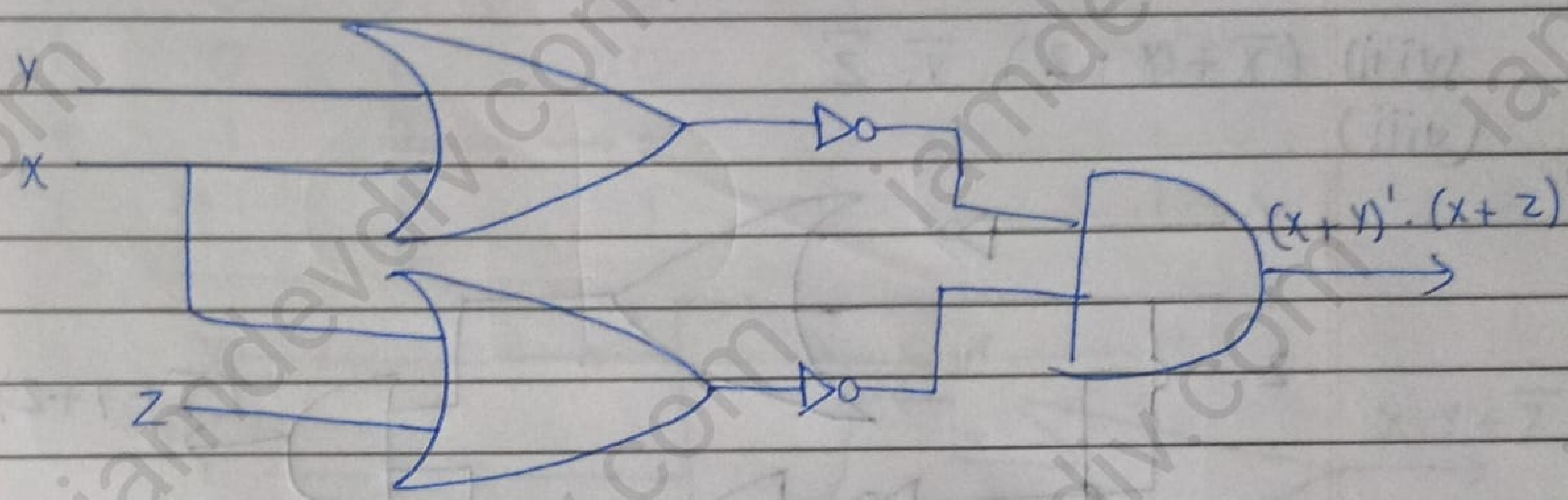
$(A+B+C)'$

(iii)



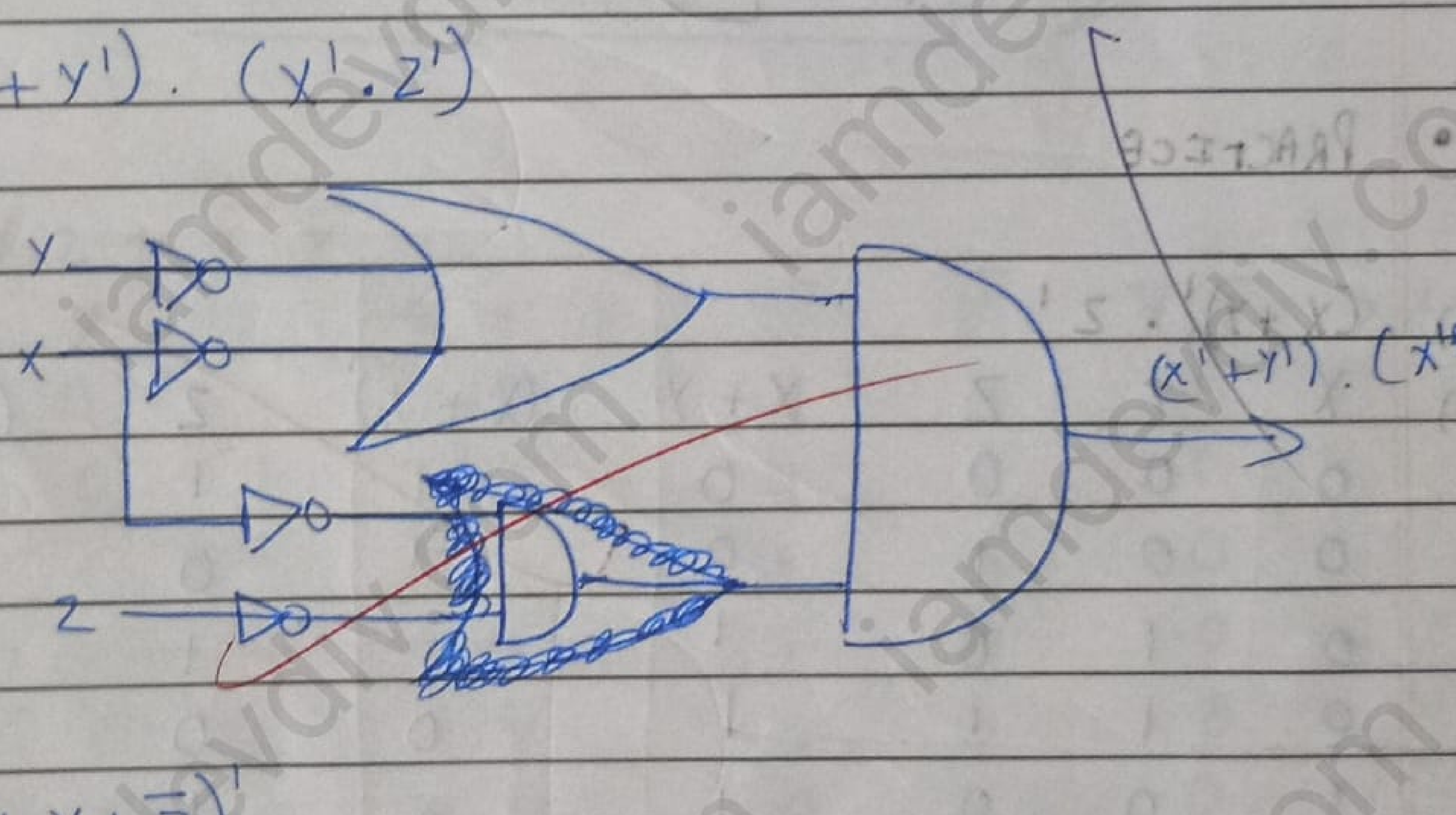
(iv) $(X+Y)' \cdot (X+Z)'$

(iv)



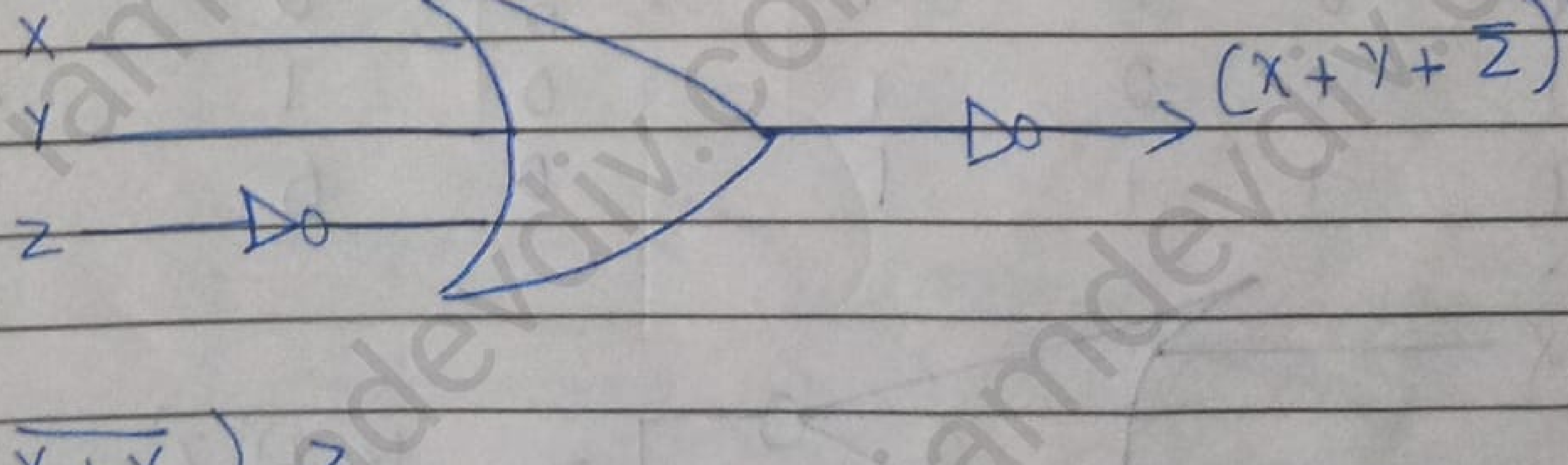
(v) $(X'+Y') \cdot (X' \cdot Z')$

(v)



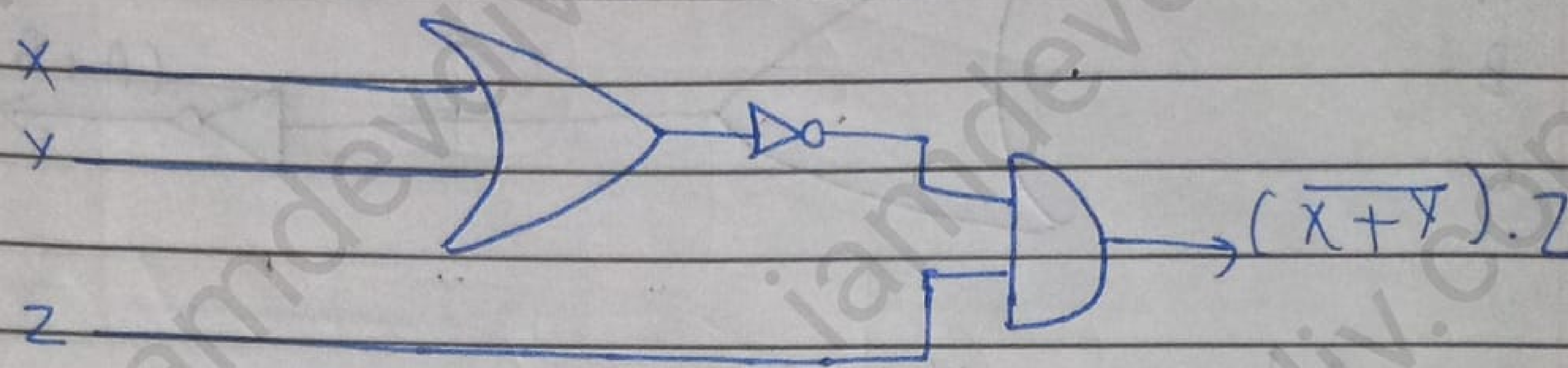
(vi) $(X+Y+\bar{Z})'$

(vi)



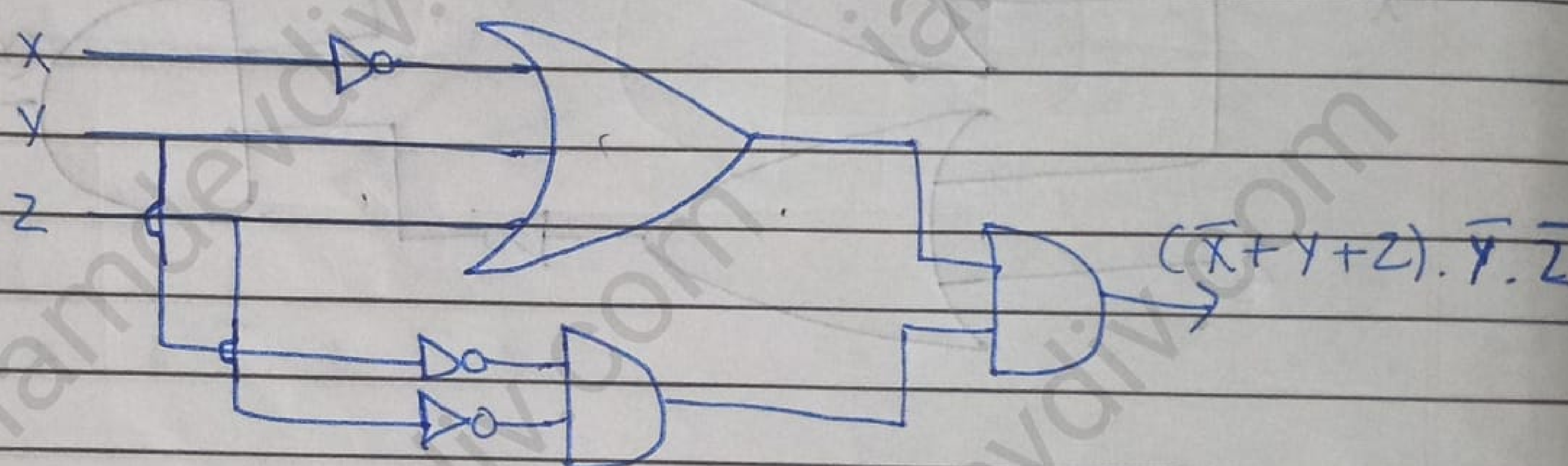
(vii) $(\overline{X+Y}) \cdot Z$

(vii) P.T.O.



(viii) $(\overline{X}+Y+Z).\overline{Y}.\overline{Z}$

(viii)

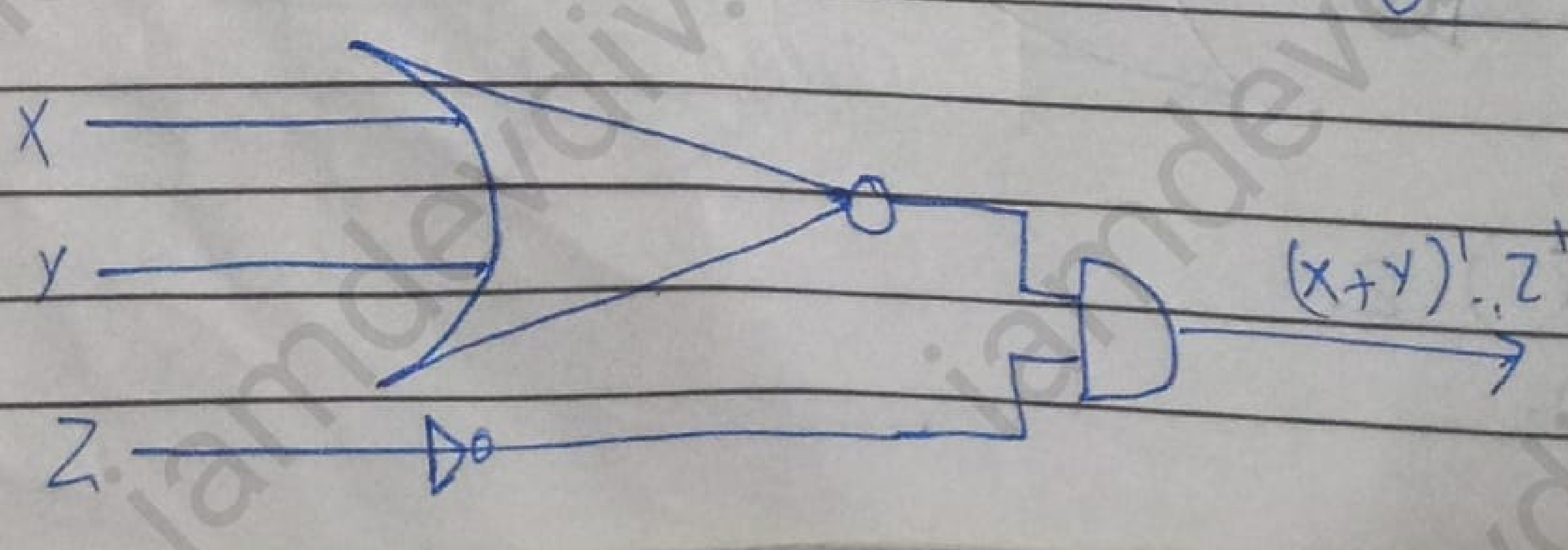


- PRACTICE

i) $(x+y)'.z'$

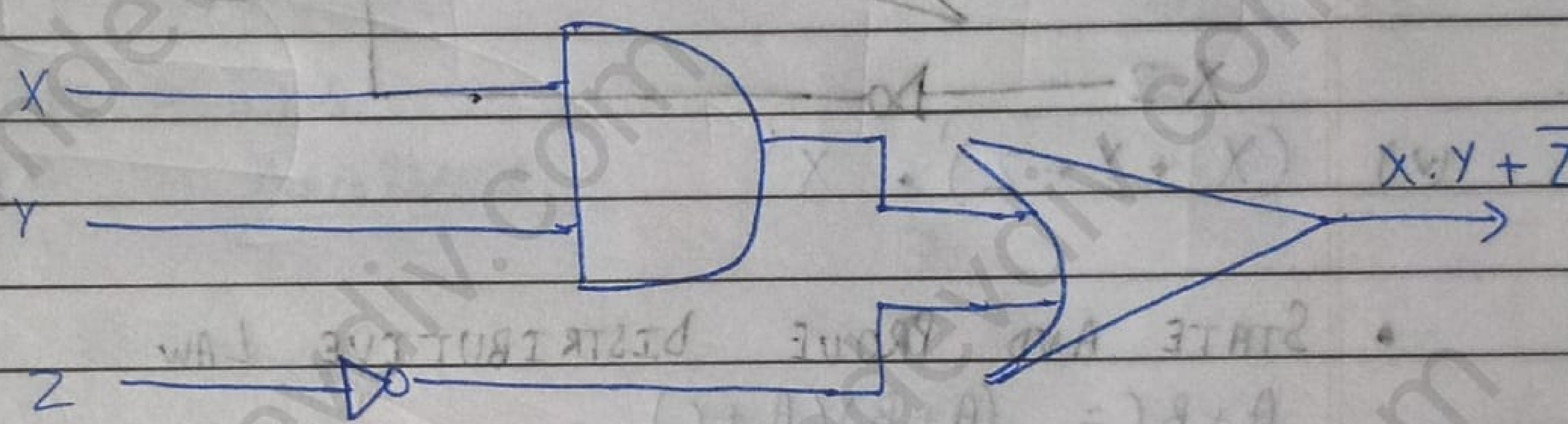
ii)

X	Y	Z	X + Y	$(X+Y)'$	Z'	$(X+Y)'.Z'$
0	0	0	0	1	1	1
0	0	1	0	1	0	0
0	1	0	1	0	1	0
0	1	1	1	0	0	0
1	0	0	1	0	1	0
1	0	1	1	0	0	0
1	1	0	1	0	1	0
1	1	1	1	0	0	0



(ii) $X \cdot Y + \overline{Z}$

X	Y	Z	$X \cdot Y$	$\overline{Z}$	$X \cdot Y + \overline{Z}$
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	0	1	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	1	0	1



(iii) $X \cdot Y \cdot Z + \overline{X} \cdot \overline{Y}$

X	Y	Z	$X \cdot Y \cdot Z$	$\overline{X}$	$\overline{Y}$	$\overline{X} \cdot \overline{Y}$	$X \cdot Y \cdot Z + \overline{X} \cdot \overline{Y}$
0	0	0	0	1	1	1	1
0	0	1	0	1	1	1	1
0	1	0	0	1	0	0	0
0	1	1	0	1	0	0	0
1	0	0	0	0	1	0	0
1	0	1	0	0	1	0	0
1	1	0	0	0	0	0	0
1	1	1	1	0	0	0	1