LOGIC GATES DC-III (NOTES)

Boolean functions are expressed in terms of AND,OR and NOT operations. These are also called logical functions. It is easier to implement Boolean functions using these gates.

Gates are electronic circuits that make logical decisions.

AND, OR and NOT are basic gates. NAND and NOR are said to be Universal gates since all gates can be derived from them.

EX-OR gate is another gate that can be implemented from basic gates.

AND: *AND* gate provides a 1 output when all the input variables are 1.

OR: OR gate provides a 1 output when any of the input variable is 1.

NOT: NOT gate compliments the variable.$\bar{A}$ is read as A NOT or complement of A.

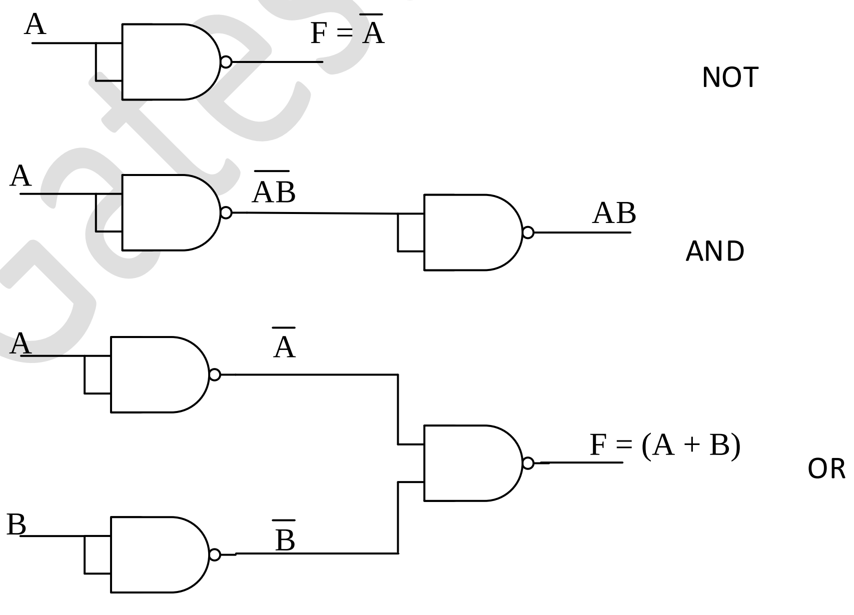
NAND: It is composed of AND and NOT functions .NAND is a contraction of NOT AND.

NOR: It is composed OR and NOT functions .NOR is contraction of NOT OR.

EX-OR: When either 1 or an odd number of inputs is 1,the out- put is 1.All other conditions give 0 output .As the name suggests that it excludes the case when even number of input variables are there.

EX-NOR: Complement of EX-OR .The out- put is 1 when number of input variables are even. It is also called equivalence or coincidence function.

Universal Gates: NAND and NOR are called Universal gates. Both can be used to realize basic gates (AND,OR and NOT)



Similarly, using NOR gates basic gates can be implemented.

NAND & NOR IMPLEMENTATIONS

Digital circuit are frequently constructed with NAND or NOR gates as compared to AND, OR,or NOT gates.

So any expression given in AND, OR, NOT can be converted into NAND and NOR logic diagram.

Use De Morgan's theorem

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

And

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

Above expressions can be written as

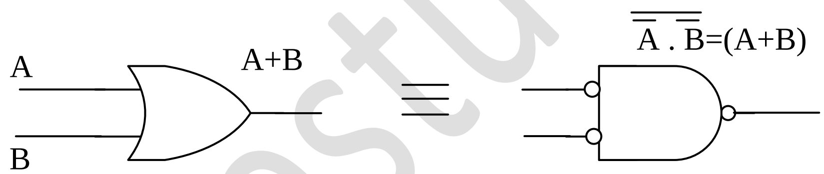
$\overline{\overline{A+B}} = (A+B) = \overline{\bar{A}.\bar{B}}$ (i)

And

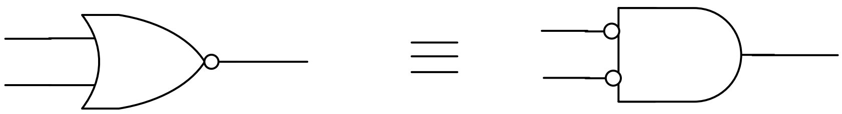
$\overline{\overline{AB}} = AB = \overline{\bar{A}+\bar{B}}$ (ii)

Equation (i) says that OR gate is equivalent to NAND gate with bubbles at its input

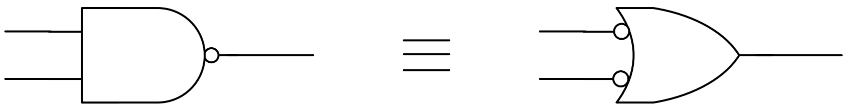
As per equation (ii) AND gate is equivalent to a NOR gate with bubbles at its input.



ALSO



Bubbled AND



Bubbled OR

Note that OR and AND have their equivalent with two inversions (i.e. circles at input and output) while NAND and NOR have only one bubble at the input side.

(i) **Using NAND gates:** Logic function is first written in SOP form. Then it can be realized by AND gates in first level and OR in the second level. This can be converted to NAND – NAND gates (Refer any text book)

So the procedure is

- Write Boolean expression is SOP form
- Simplify expression
- Double invert it

(ii) **Using NOR gates:** The logic function be written is POS form. Then its realization will involve OR gates at first level and AND gates in second level. This can be converted to NOR – NOR gate circuit.

Ex- OR and Ex -NOR Gates

Both are commutative and associative and can be extended to more than two inputs. These gates find wide applications such as adders / sub tractors, code converters, parity generators / checkers, comparators etc. An Ex OR function produces 1 Output when odd number of inputs are 1.

A two input Ex OR function is written as

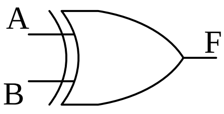
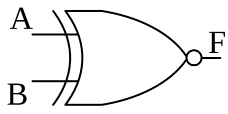
$F = (A \oplus B) = A\,\bar{B} + \bar{A}B$

An EX-NOR is also called equivalence function .The output is 1 when even number of inputs are 1.It is complement of EX-OR.

$F = \overline{A \oplus B} = (A \odot B) = \bar{A}\,\bar{B} + AB$

3 input or more input Ex OR and Ex NOR function can be derived from 2 input functions

Comparative study of EX-OR and EX-NOR Gates

EX-OR	EX-NOR
	
A B F 0 0 0 0 1 1 1 0 1 1 1 0	A B F 0 0 1 0 1 0 1 0 0 1 1 1
$F = A\bar{B} + \bar{A}B = A \oplus B\ (SOP)$ $F = (A + B)(\bar{A} + \bar{B})\ POS$	$F = AB + \bar{A}\bar{B} = \overline{A \oplus B} = A \odot B\ (SOP)$ $F = (\bar{A} + B)(A + \bar{B})\ POS$
(2) Output is 1 when input variables are odd (odd function)	Output is 1 when input variables are even (even function)
(3) If one input of the gate is high it behaves as controlled inverter. If low then buffer	Complement of EX-OR function
(4) Used as modulo 2 adder or half adder	Called coincidence or equivalence gate. When two inputs are same output is 1.
(5) 3 and more input EX-OR can be derived from 2 input.	3 and more input EX-NOR can be derived from 2 input.