Unit-3 (Digital Electronics)

BCD Codes :- (Binary Coded Decimal) - Each decimal digit is represented by a 4-bit binary number.

Decimal Number	Binary	BCD
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001
10	1010	0001 0000
11	1011	0001 0001
12	1100	0001 0010
13	1101	0001 0011
14	1110	0001 0100
15	1111	0001 0101

Convert the following decimal numbers to BCD

(a) - 35 (b) 174 (c) 2479

Excess-3 Code:-

It is a modified form of a BCD number. It can be derived from the natural BCD by adding 3 to each coded number.

0	~~Excess-3~~ 0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1001

Gray Code:-

It is a special unit distance code. In this code bit patterns for two ~~succ~~ consecutive numbers differ in only one bit position. These codes are also called cyclic codes.

Binary to Gray Code Conversion

Binary $\rightarrow B_1 \oplus B_2 \oplus B_3 \oplus B_4$

$$B_1 \downarrow G_1 \quad B_2 \downarrow G_2 \quad G_3 \quad G_4$$

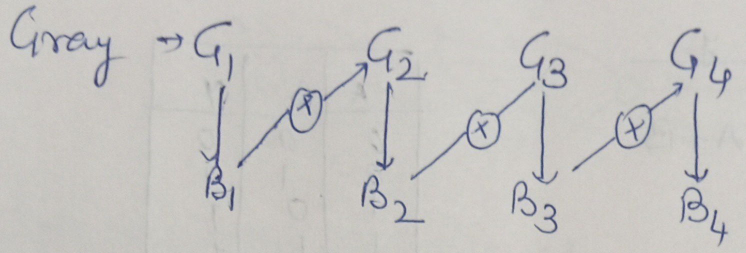
Ex

$$1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1$$

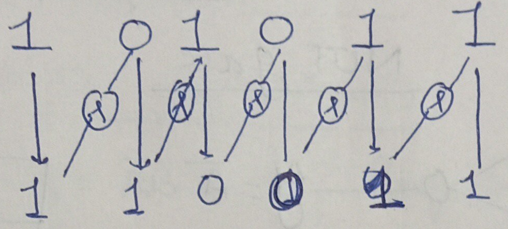
$$\downarrow$$

$$1 \; 1 \; 1 \; 0 \; 0 \; 1 \; 1 \; 0.$$

Gray to Binary



Ex



Logic gates

Basic gates

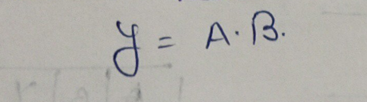
1. AND
2. OR
3. ~~NAND~~
4. ~~NOR~~
5. Ex-OR
6. Ex-NOR
7. NOT.

Universal gates.

1. NAND
2. NOR

By using Universal gates any basic gates can be formed.

AND gate

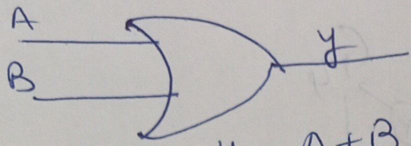


$y = A \cdot B.$

Truth Table

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

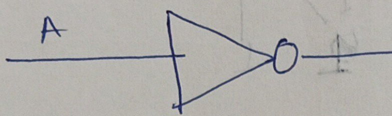
OR gate



$y = A + B$

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

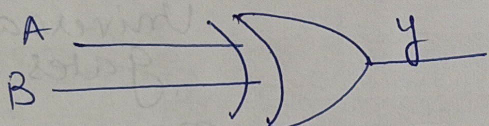
NOT gate



$y = \overline{A}$

A	Y
0	1
1	0

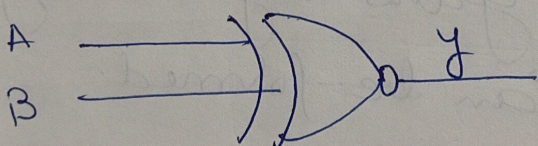
Ex-OR



$y = A \oplus B$
$= A\overline{B} + \overline{A}B$

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

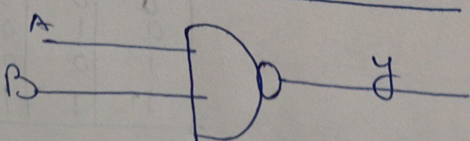
Ex-NOR



$y = A \odot B$
$= \overline{A}\,\overline{B} + AB$

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

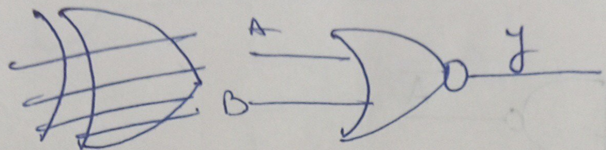
NAND



$y = \overline{AB}$

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

NOR

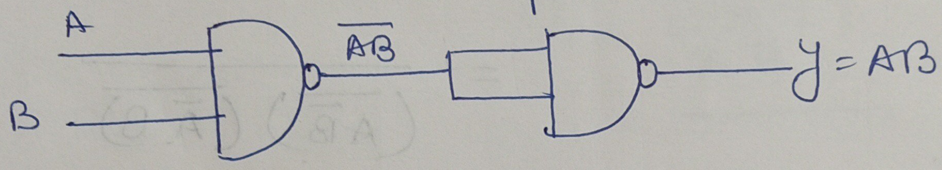


A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Conversion of Gates

1)- NAND to AND

Given - $y = \overline{A.B}$ | Required $y = A.B$



2)- NAND to OR

$y = \overline{AB}$

$\bar{\bar{y}} = \cancel{\overline{\overline{AB}}}$

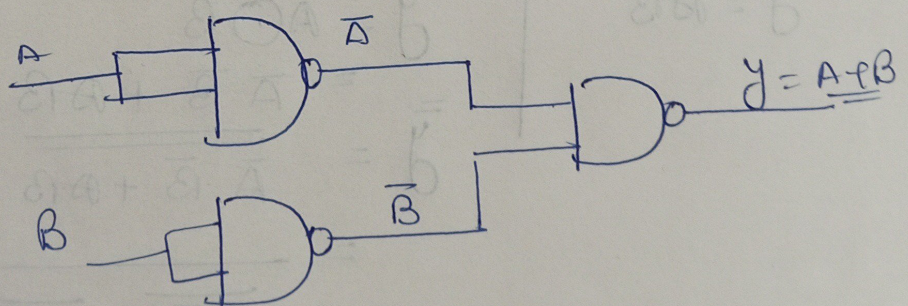
$y = \cancel{\overline{\overline{\bar{A} + \bar{B}}}} = \cancel{\bar{\bar{A}} \cdot \bar{\bar{B}}}$

$y = A + B$

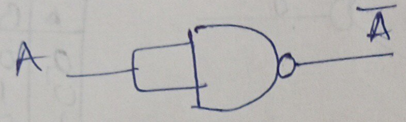
$\bar{\bar{y}} = \overline{\overline{A + B}}$

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

$= A + B$



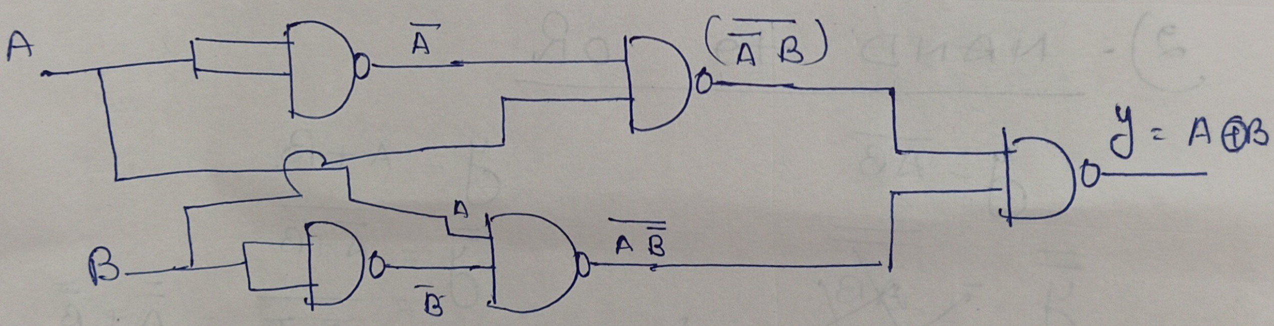
NAND to NOT



NAND to Ex-OR

$y = \overline{AB}$

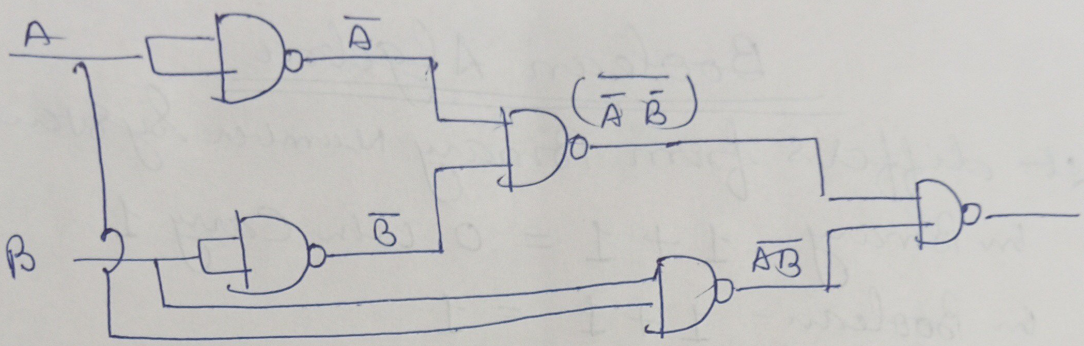
$$y = A \oplus B$$
$$= A\bar{B} + \bar{A}B$$
$$\bar{\bar{y}} = \overline{\overline{A\bar{B} + \bar{A}B}}$$
$$= \overline{(\overline{A\bar{B}})(\overline{\bar{A}B})}$$



NAND to Ex-NOR

$y = \overline{AB}$

$$y = A \odot B$$
$$= \bar{A}\bar{B} + AB$$
$$\bar{\bar{y}} = \overline{\overline{\bar{A}\bar{B} + AB}}$$
$$= \overline{(\overline{\bar{A}\bar{B}}) \cdot \overline{AB}}$$



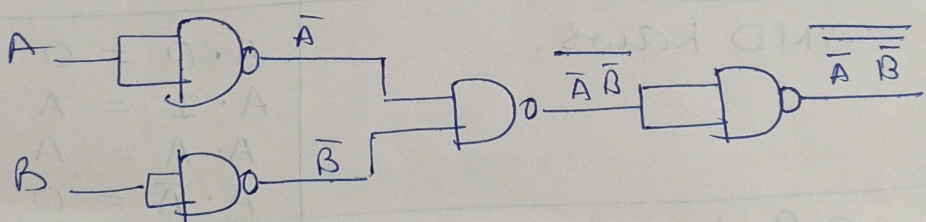
NAND to NOR

$y = \overline{AB}$ | $y = \overline{A+B}$

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

$$= \overline{\overline{\bar{A}\,\bar{B}}}$$

$$= \underline{\qquad\qquad}$$



<u>Similarly</u> NOR to AND, OR, NOT, Ex-OR, Ex-NOR, NAND

Boolean Algebra

It differs from Binary Number System

In Binary $1 + 1 = 0$ with Carry 1

In Boolean $= 1 + 1 = 1$

Boolean Algebra Laws

S.No.	Name of Law	Statement
1 -	Commutative Law	$A \cdot B = B \cdot A$ $A + B = B + A$
2.	Associative Law	$(A \cdot B) \cdot C = A \cdot (B \cdot C)$ $(A + B) + C = A + (B + C)$
3.	Distributive Law	$A(B + C) = AB + AC$
4	AND Laws.	$A \cdot 0 = 0$ $A \cdot 1 = A$ $A \cdot A = A$ $A \cdot \overline{A} = 0$
5	OR Laws	$A + 0 = A$ $A + 1 = 1$ $A + A = A$ $A + \overline{A} = 1$
6	Inversion Laws	$\overline{\overline{A}} = A$
7	Other Important Laws.	$A + BC = (A + B)(A + C)$ $\overline{A} + AB = \overline{A} + B$ $\overline{A} + A\overline{B} = \overline{A} + \overline{B}$ $A + AB = A$ $A + \overline{A}B = A + B$

Demorgan's theorems

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

(2) $\overline{A + B} = \overline{A} \cdot \overline{B}$

Ex $A + BC = (A+B)(A+C)$

L.H.S. $A + BC$

$\Rightarrow A \cdot 1 + BC$

$\Rightarrow A \cdot (1 + C) + BC$

$\Rightarrow A \cdot 1 + AC + BC$

$\Rightarrow \cancel{A \cdot 1 + C(A+B)}$

$= A \cdot (1 + B) + AC + BC$

$= A \cdot 1 + AB + AC + BC$

$= A \cdot A + AB + AC + BC$

$= AA + AC + AB + BC$

$= A(A+C) + B(A+C)$

$= (A+C)(A+B)$

Q. → (1) $A + AB = A$

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

(3) $(A+B)(A+C) = A + BC$

(4) $\cancel{(A+\overline{B})}((A+\overline{B}) + AB)(A+\overline{B})(\overline{A}B) = 0$

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

(6) $ABCD + A\overline{B}CD = ACD$

(7) $\overline{(\overline{AB} + \overline{A} + AB)} = 0$

(8) $AB + ABC + A\overline{B} = A$

SOP & POS Representation for logic Expression

SOP Form (Sum of Product)

$$y = \underbrace{A \cdot B + AC + BC}_{\text{product terms}} \quad \text{(+ : Sum)}$$

$$y = \bar{A} B \bar{C} + A \bar{B} \bar{C} + \bar{A} \bar{B} \bar{C}$$

POS Form (Product of Sum)

$$y = (A+B) \cdot (B+\bar{C}) \cdot (\bar{A}+B)$$

Standard or Canonical SOP & POS forms

Standard SOP → $y = f(A, B, C)$

$$= \underbrace{ABC}_{\text{minterm 1}} + \underbrace{A\bar{B}C}_{\text{minterm 2}} + \underbrace{\bar{A}BC}_{\text{minterm 3}}$$

All terms have all variables either in its origional form or in its complemented form.

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

Convert into standard form.

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

$$= ABC + AB\bar{C} + A\bar{B}C + \bar{A}\bar{B}C + \bar{A}[BC + B\bar{C} + \bar{B}C + \bar{B}\bar{C}]$$

$$= ABC + AB\bar{C} + A\bar{B}C + \bar{A}\bar{B}C + \bar{A}BC + \bar{A}B\bar{C} + \bar{A}\bar{B} \ldots$$

Standard POS form

$$y=\underbrace{(A+B+\overline{C})}_{\text{Maxterm 1}}\ \underbrace{(A+\overline{B}+C)}_{\text{maxterm 2}}\ \underbrace{(\overline{A}+\overline{B}+\overline{C})}_{\text{maxterm 2}}$$

$$y=(A+B)(A+C)(\overline{A})$$

$$=(A+B+0)(A+C+0)(\overline{A}+0)$$

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

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

$$(A+B+C)(A+\overline{B}+C)$$

Minterms & Maxterms for 3 variables

Variables	minterms	maxterm.
0 0 0	$\overline{A}\,\overline{B}\,\overline{C}=m_0$	$A+B+C=M_0$
0 0 1	$\overline{A}\,\overline{B}\,C=m_1$	$A+B+\overline{C}=M_1$
0 1 0	$\overline{A}\,B\,\overline{C}=m_2$	$A+\overline{B}+C=M_2$
0 1 1	$\overline{A}\,B\,C=m_3$	$A+\overline{B}+\overline{C}=M_3$
1 0 0	$A\,\overline{B}\,\overline{C}=m_4$	$\overline{A}+B+C=M_4$
1 0 1	$A\,\overline{B}\,C=m_5$	$\overline{A}+B+\overline{C}=M_5$
1 1 0	$A\,B\,\overline{C}=m_6$	$\overline{A}+\overline{B}+C=M_6$
1 1 1	$A\,B\,C=m_7$	$\overline{A}+\overline{B}+\overline{C}=M_7$

Ex

$y = A\overline{B}C + \overline{A}\,\overline{B}C + A\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C}$

$= m_5 + m_1 + m_4 + m_0$

$= m_0 + m_1 + m_4 + m_5$

$$\boxed{y = \sum m(0, 1, 4, 5)}$$

$y = (\underset{0}{A} + \underset{1}{\overline{B}} + \underset{0}{C})(\overline{A} + \overline{B} + C)(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + \overline{C})$

$= M_2 \cdot M_6 \cdot M_3 \cdot M_7$

$$\boxed{y = \prod M(2, 3, 6, 7)}$$

K- MAP

K-map is used to minimize the boolean function.

K-map for 3 variables.

A \ BC	00	01	11	10
0	0	1	3	2
1	4	5	7	6

K-map for 4 variables

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

Ex $y = \Sigma m(0, 1, 2, 5, 13, 15)$

Ans $\bar{A}\bar{B}\bar{D} + \bar{A}\bar{C}D + ABD$

$y = \Sigma m(1, 5, 7, 9, 11, 13, 15)$

Ans. $\bar{A}D + BD + AD$

$y = \Sigma m(1, 3, 5, 9, 11, 13)$

$y = \bar{B}D + \bar{C}D$

$y = \Sigma m(1, 2, 3, 4, 5, 7, 9, 11, 13, 15)$

$y = \bar{A}B\bar{C} + D$

$y = \Sigma m(1, 2, 9, 10, 11, 14, 15)$

Ans. $\bar{B}(C \oplus D) + AC$

$y = \Sigma m(4, 5, 8, 9, 11, 12, 13, 15)$

Ans. $\bar{C}(A+B) + AD$

$y = \Sigma m(1, 3, 7, 11, 15) + d(0, 2, 5)$

$y = CD + \bar{A}\bar{B}$

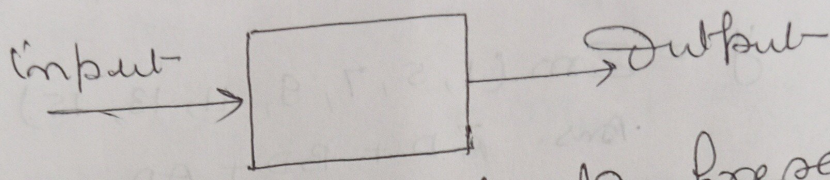
$y = \Sigma m(0, 1, 5, 9, 13, 14, 15) + d(3, 4, 7, 10, 11)$

Ans. $(D + (A \odot C)$

$y = \Pi M(0, 2, 3, 5, 7)$

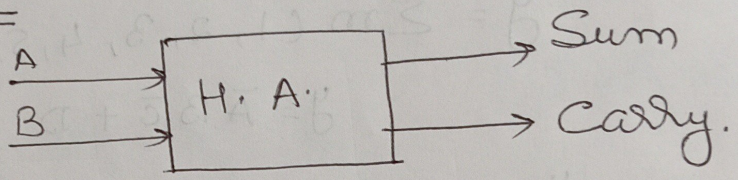
Ans $(A+C)(\bar{A}+\bar{C})(\bar{B}+\bar{C})$

Combinational circuits



Output is only depend upon present input there is no memory element.

Half Adder

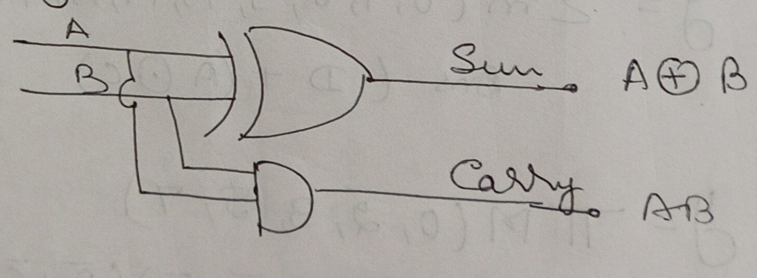


A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

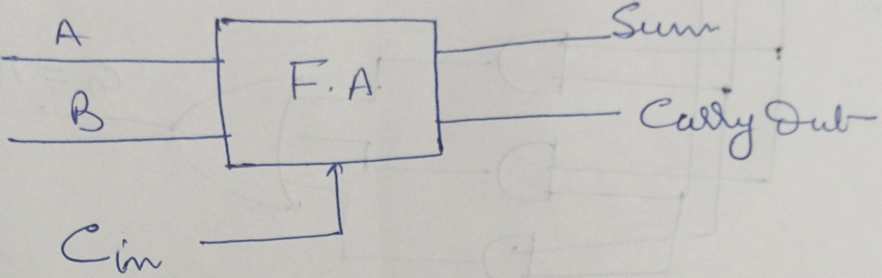
$$Sum = \overline{A}B + A\overline{B}$$

$$= A \oplus B.$$

$$Carry = AB$$



Full Adder



A	B	Cin	Sum	Carry
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

$$S = \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$$

$$= (\bar{A}\bar{B} + AB)C_{in} + (\bar{A}B + A\bar{B})\bar{C}_{in}$$

$$= (\overline{\bar{A}B + A\bar{B}})C_{in} + (\bar{A}B + A\bar{B})\bar{C}_{in}$$

$$= \bar{A} \oplus B \oplus C_{in}$$

$$C = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\bar{C}_{in} + ABC_{in}$$

$$= \bar{A}BC_{in} + \underline{A\bar{B}C_{in}} + \underline{ABC_{in}} + AB\bar{C}_{in}$$

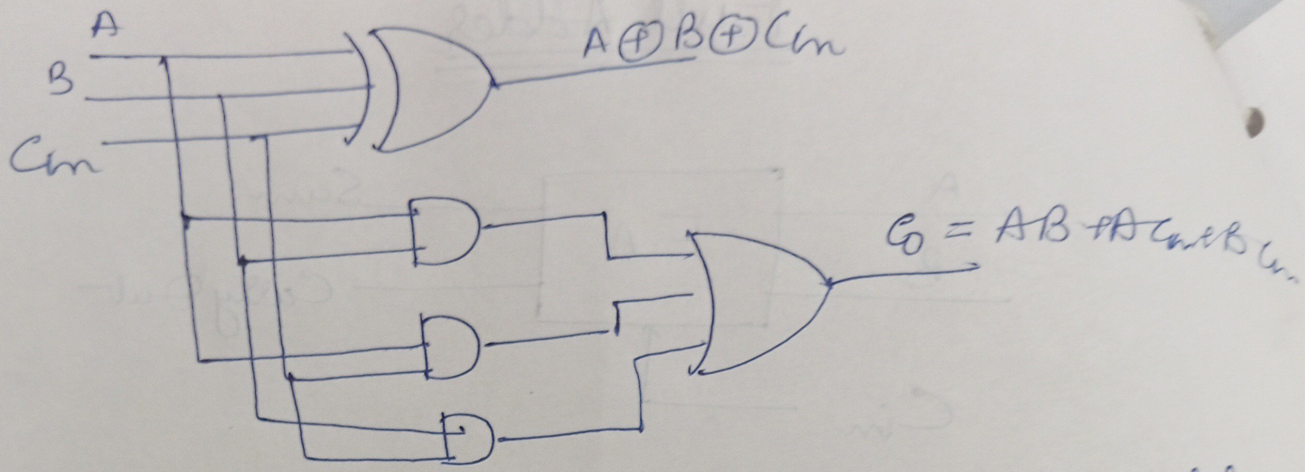
$$= \bar{A}BC_{in} + AC_{in} + AB\bar{C}_{in}$$

$$= (A+B)C_{in} + AB\bar{C}_{in}$$

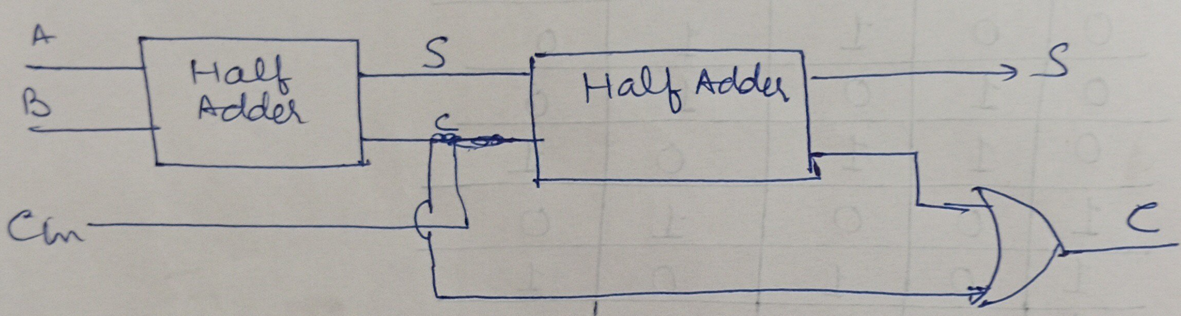
$$= AC_{in} + BC_{in} + AB\bar{C}_{in}$$

$$= \bar{A}(B + C_{in}) + BC_{in}$$

$$AB + AC_{in} + BC_{in}$$



Construction of full adder using Half Adder



Binary Subtractor

$0 - 0 = 0$, $0 - 1 = 1$ with borrow 1

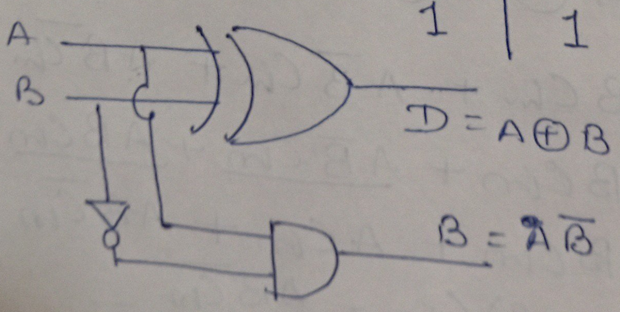
$1 - 0 = 1$, $1 - 1 = 0$

Half Subtractor :-

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

$B = A\bar{B}$

A	B	D	B
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0



Full Subtractor

A	B	B_{in}	D	B_0
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

A \ $B B_{in}$	00	01	11	10
0		1		1
1	1		1	

A \ $B B_{in}$	00	01	11	10
0		1	1	1
1			1	

$$D = A\bar{B}\bar{B}_{in} + \bar{A}\bar{B}B_{in} + ABB_{in} + \bar{A}B\bar{B}_{in}$$

$$= A \oplus B \oplus B_{in}.$$

$$B_{out} = \bar{A}B_{in} + \bar{A}B + BB_{in}.$$