Full adder

→ 3 i/p to add → no. of o/p = ?

→ Similarly 4 i/p to add

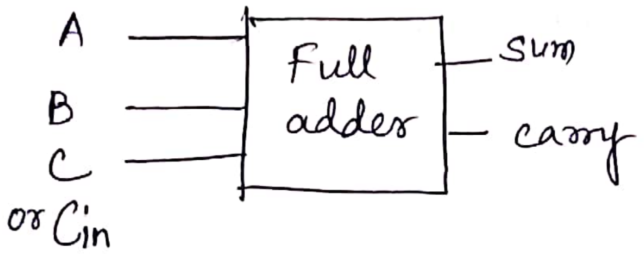
$$\begin{array}{r} 1 \\ 1 \\ 1 \\ 1 \\ \hline 100 \end{array}$$

100 3 o/p bits.

$$\begin{array}{r} 0 \\ 0 \\ 0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 1 \\ 1 \\ 1 \\ \hline 11 \end{array}$$

min^m max m

→



Truth table

A	B	C	~~Sum~~	Carry	Sum
0	0	0		0	0
0	0	1		0	1
0	1	0		0	1
0	1	1		1	0
1	0	0		0	1
1	0	1		1	0
1	1	0		1	0
1	1	1		1	1

$$Sum = \sum m\,(1, 2, 4, 7)$$

$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC \quad \text{(SOP form)}$$

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

$$= (A \oplus B) \oplus C = A \oplus B \oplus C$$

→ So, sum of any i/p bits follow the XOR pattern

Half adder sum, $sum = A \oplus B$

full adder ", $sum = A \oplus B \oplus C$

Similarly for a 8 bit adder, $sum = A \oplus B \oplus C \oplus \cdots \oplus h$

$Carry = \sum m(3, 5, 6, 7)$

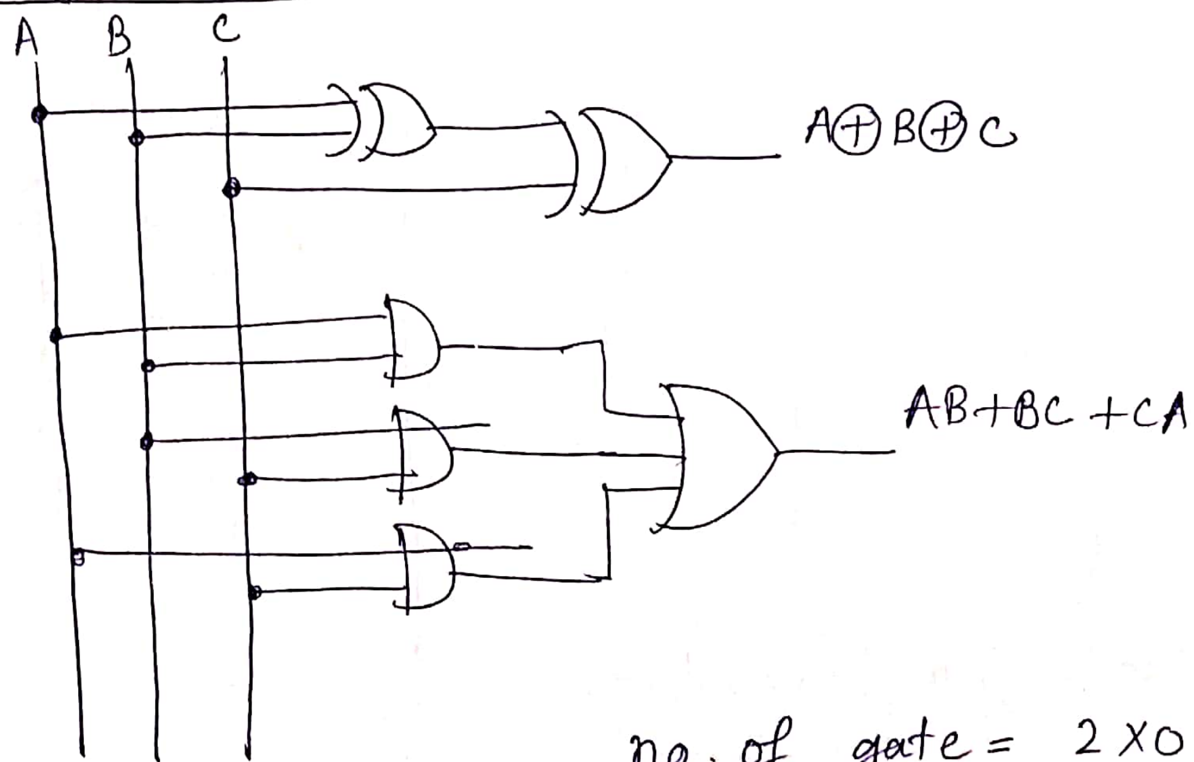
$= \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

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

$= BC + A(B \oplus C) = AB + C(A \oplus B) + CA + B(A \oplus C)$ → (1)

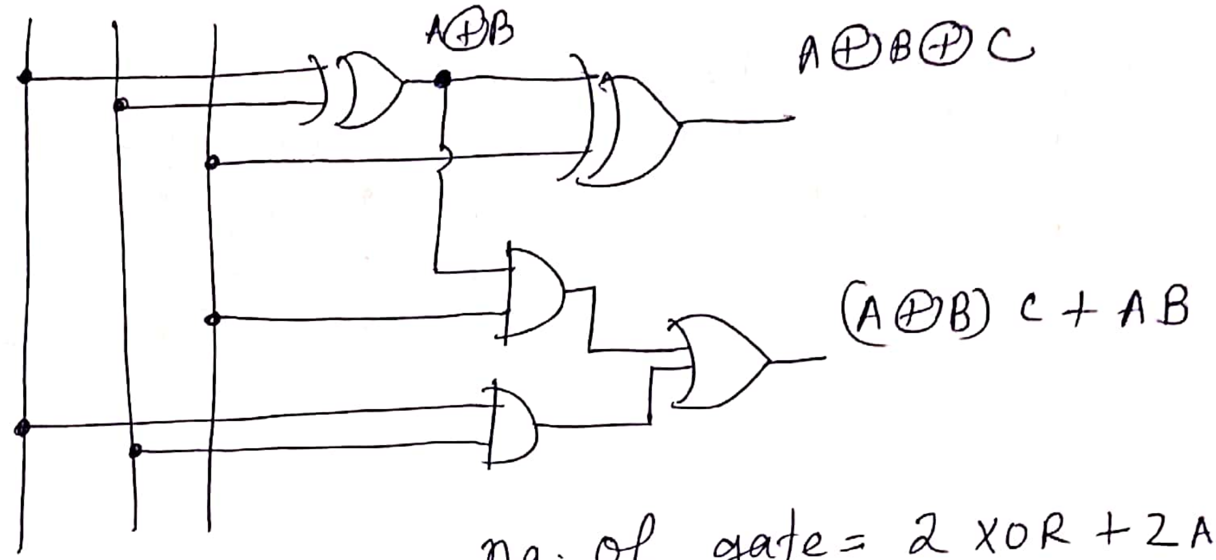
K-map minimization $= AB + BC + CA$

Logic circuit:



no. of gate = 2 XOR + 3 AND + 1 OR

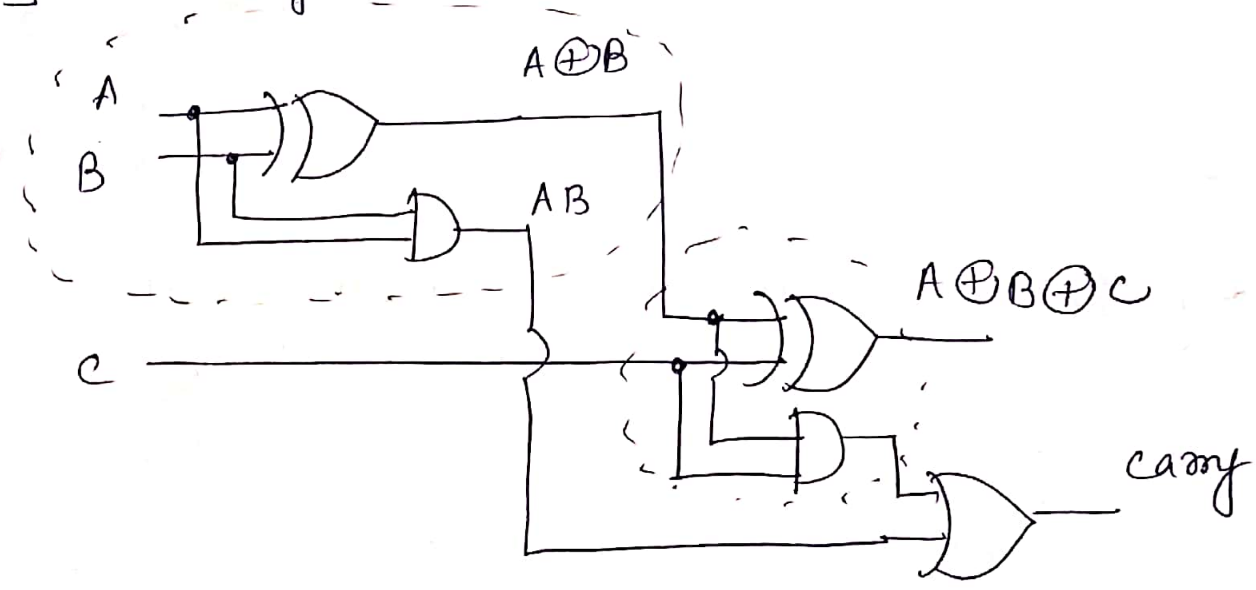
or, logic circuit:



no. of gate = 2 XOR + 2 AND + 1 OR

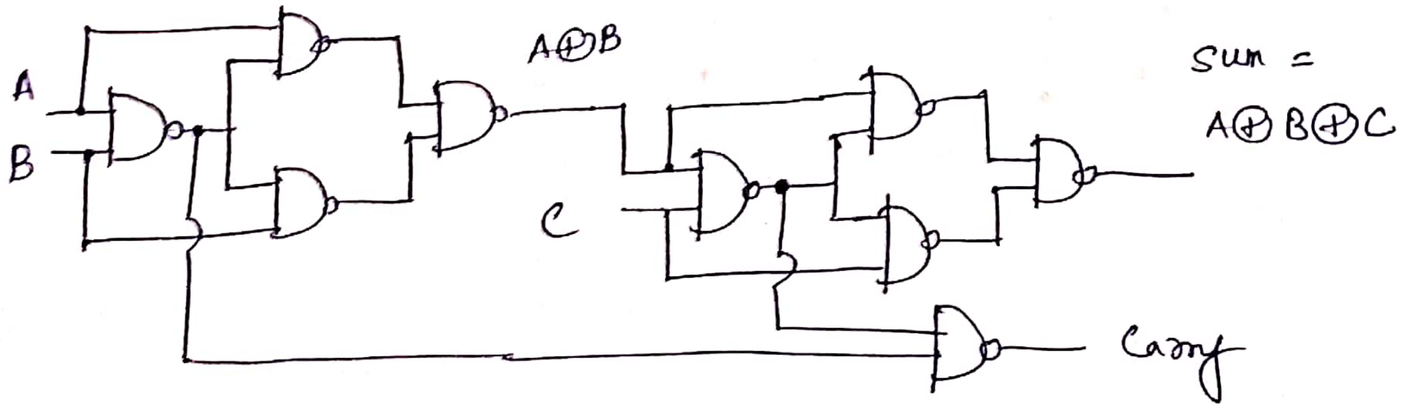
or, two other ways a/c eq^n 1

same diagram,



So, FA = 2HA + 1 OR gate

Full adder using NAND :



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

$$= \overline{\overline{AB} \cdot \overline{(A \oplus B)C}}$$

total NAND gate required to design Full adder = 9

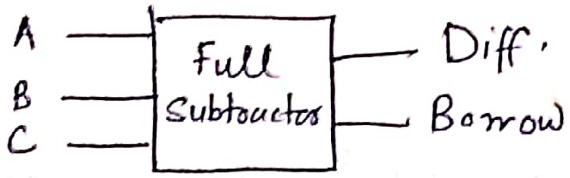
" " " " " " Sum = 8

" " " " " " Carry = 6

try full adder using NOR : . do it yourself.

Full subtractor:

(A-B-C)



Truth table:

A	B	C	~~Diff~~ Borrow	Diff.
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$\text{Diff.} = \sum m(1,2,4,7) = A \oplus B \oplus C$$ [same as sum of full adder]

$$\text{Borrow} = \sum m(1,2,3,7)$$

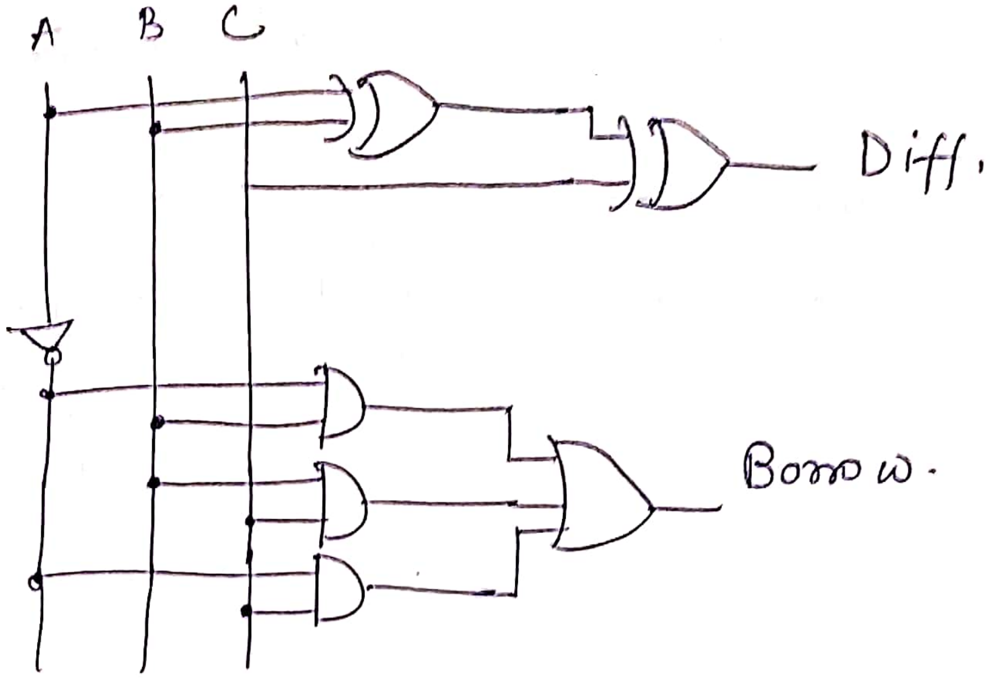
$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + ABC$$

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

K-map minimization

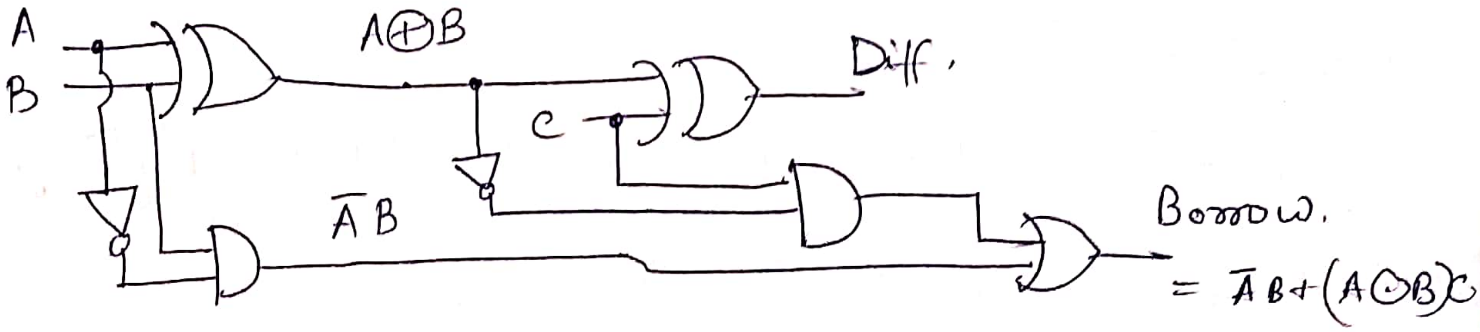
$$= \bar{A}C + \bar{A}B + BC$$

Logical circuit :



Borrow can also be written as $= \bar{A}B + (A \odot B) \cdot C$

Full subtractor using half subtractor :



try Full subtractor using NAND only & NOR only :

* Calculate no. of gates required.

Full adder as a counter:

Truth table:

A	B	C	Cy	Sum		
0	0	0	0	0	→	0
0	0	1	0	1	→	1
0	1	0	0	1	→	1
0	1	1	1	0	→	2
1	0	0	0	1	→	1
1	0	1	1	0	→	2
1	1	0	1	0	→	2
1	1	1	1	1	→	3

it counts no. of i/p which are '1'