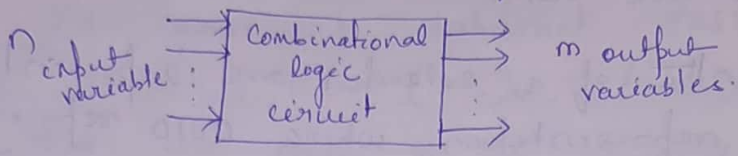
Combinational circuits

Digital circuit is two types:
- Combinational logic circuits
- Sequential logic circuits.

→ A combinational circuit consists of logic gates whose output depends on its present inputs only.



(A block diagram of combinational circuit).

→ A combinational circuit consists of input variables, logic gates and output variables.

→ For n input variables, there are 2^n possible combinations of binary input values. Each output function is expressed in terms of the n input variables.

Binary Adder

The most basic arithmatic operation is the addition of two binary bits.

A combinational circuit that performs the addition of two bits is called a half adder.

A combinational circuit that performs the addition of three bits (two significant bits and a previous carry) is a full adder.

Half-adder

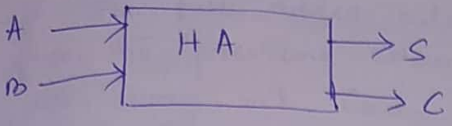
A half adder is a combinational circuit with two binary inputs (augend and addend bits) and two binary outputs (sum and carry bits). It adds the two inputs (A and B) and produces the sum (S) and the carry (C) bits.

T.T

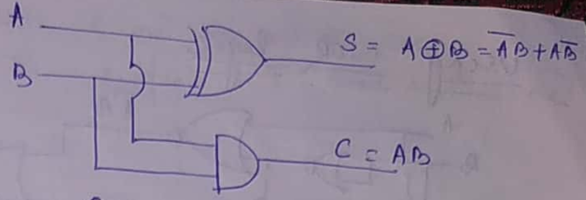
I/P		O/P	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

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

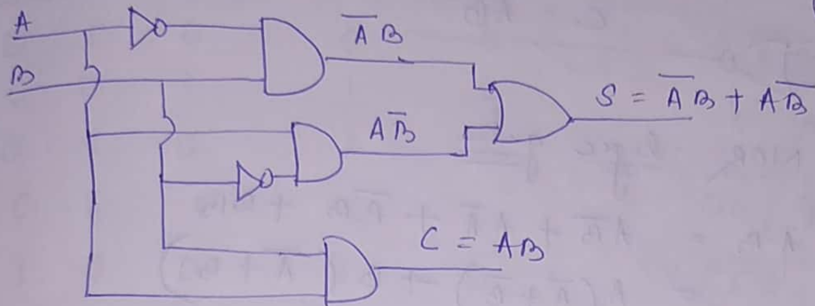
$C = AB$



(Block diagram)

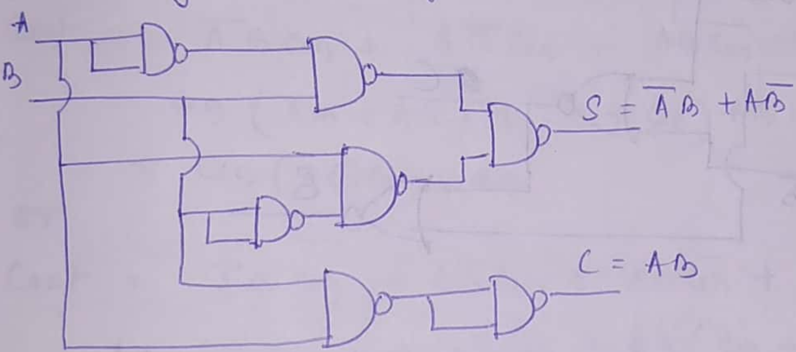


(logic diagram)



(logic diagram using AOI logic)

Using NAND logic



Using minimum number of NAND gates

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

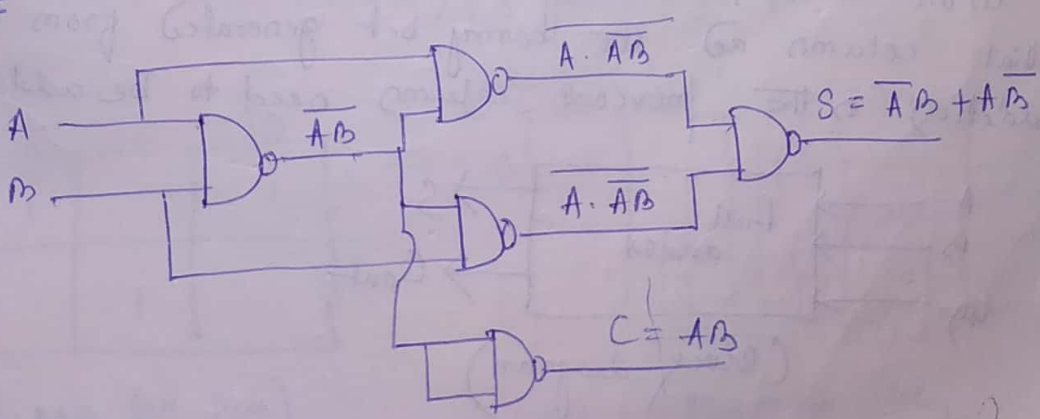
$$= A(\bar{A} + \bar{B}) + B(\bar{A} + \bar{B})$$

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

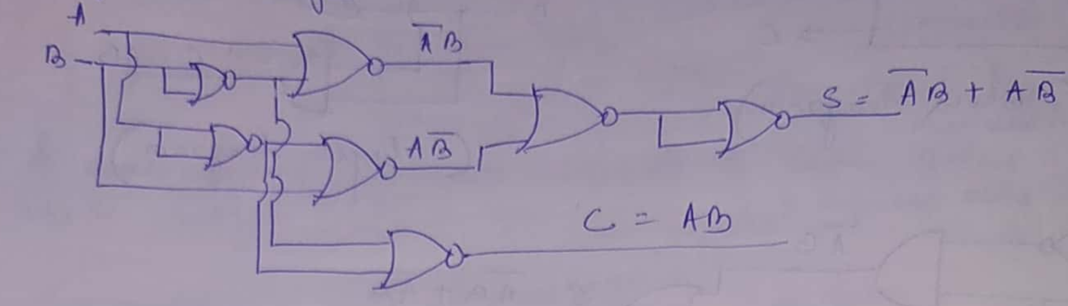
$$= A \cdot \overline{AB} + B \cdot \overline{AB}$$

$$= \overline{\overline{A \cdot \overline{AB}} \cdot \overline{B \cdot \overline{AB}}}$$

$$C = \overline{\overline{AB}}$$



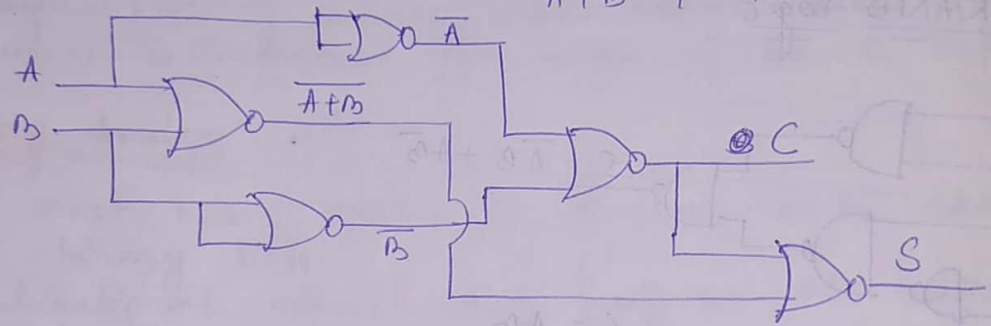
Using NOR logic



Using minimum NOR logic gates

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

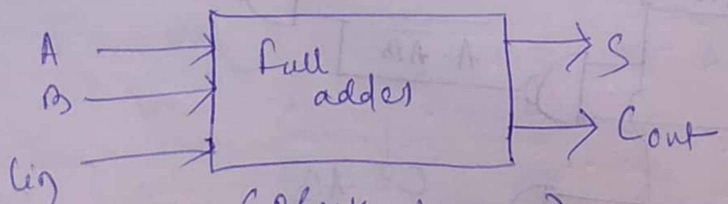
$$C = AB = \overline{\bar{A}+\bar{B}}$$



Full - Adder

→ A full adder is a combinational circuit that adds two bits and a carry, and outputs a sum bit and a carry bit.

→ The LSBs of a two binary number can be added using a half-adder. The carry resulted from the addition of the LSBs is carried over to the next significant column and added to the two bits in that column. So, in the second and higher column, the two data bits of that column and the carry bit generated from the addition in the previous column need to be added.



(Block diagram)

→ The full adder adds the bits A and B, and the carry from the previous column C_{in}. The outputs are sum bit S and the carry bit C_{out}.

i/p			o/p	
A	B	Cin	S	Cout
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

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

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

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

$$= C_{in}(\overline{A \oplus B}) + \overline{C_{in}}(A \oplus B)$$

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

$$C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\overline{C_{in}} + ABC_{in}$$

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

$$= C_{in}(A \oplus B) + AB$$

or

$$C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\overline{C_{in}} + ABC_{in}$$

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

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

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

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

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

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

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

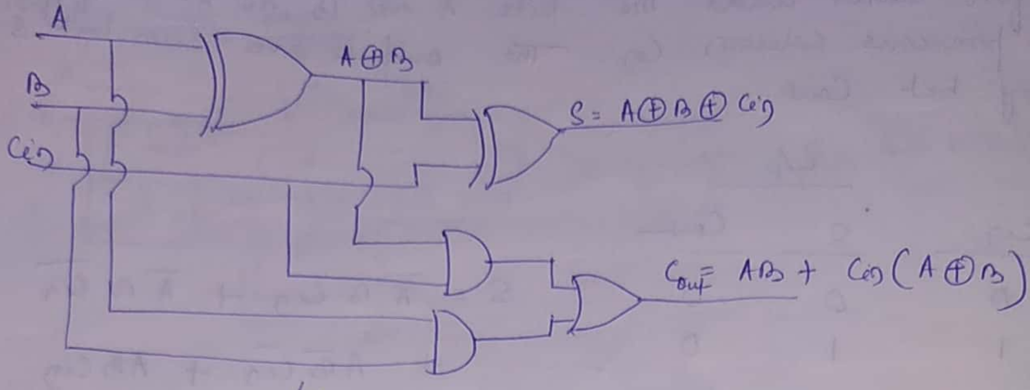
In K-map

A \ B Cin	00	01	11	10
0		1		1
1	1		1	

(K-map for sum)

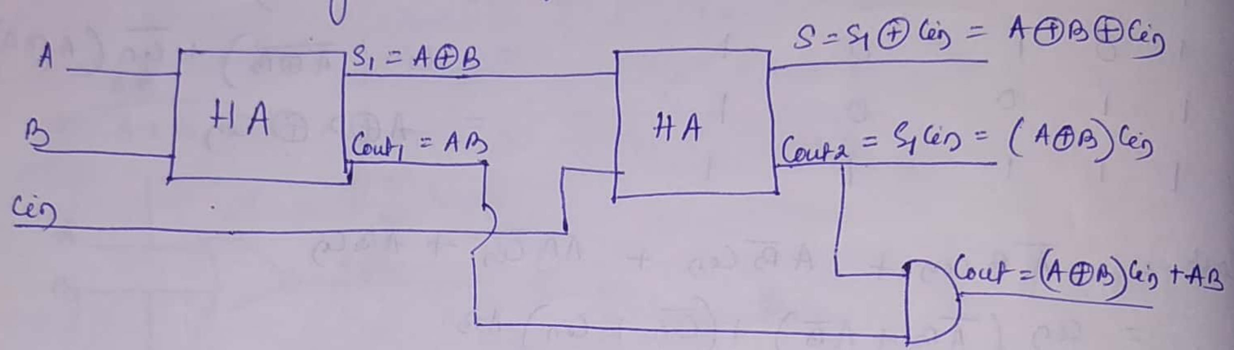
A \ B Cin	00	01	11	10
0			1	
1		1	1	1

$$C_{out} = AC_{in} + AB + BC_{in}$$



(logic diagram)

Full adder using two half adders :-



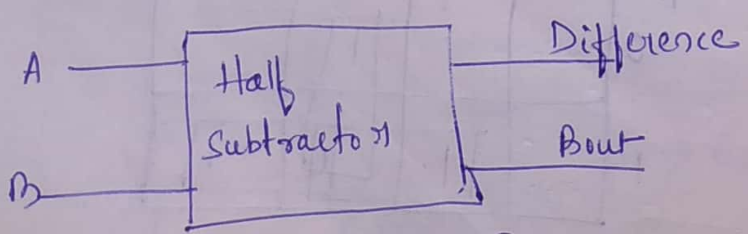
Half - Subtractor

→ In subtraction, each subtrahend bit of the number is subtracted from its corresponding significant minuend bit to form a difference bit.

→ If the minuend bit is smaller than the subtrahend bit, a 1 is borrowed from the next significant position.

→ A half - subtractor is a combinational circuit that subtracts one bit from the other and produces the difference.

→ The LSB of the subtrahend is subtracted from the LSB of the minuend. The two outputs are difference D and borrow output (Bout).



(Block diagram)