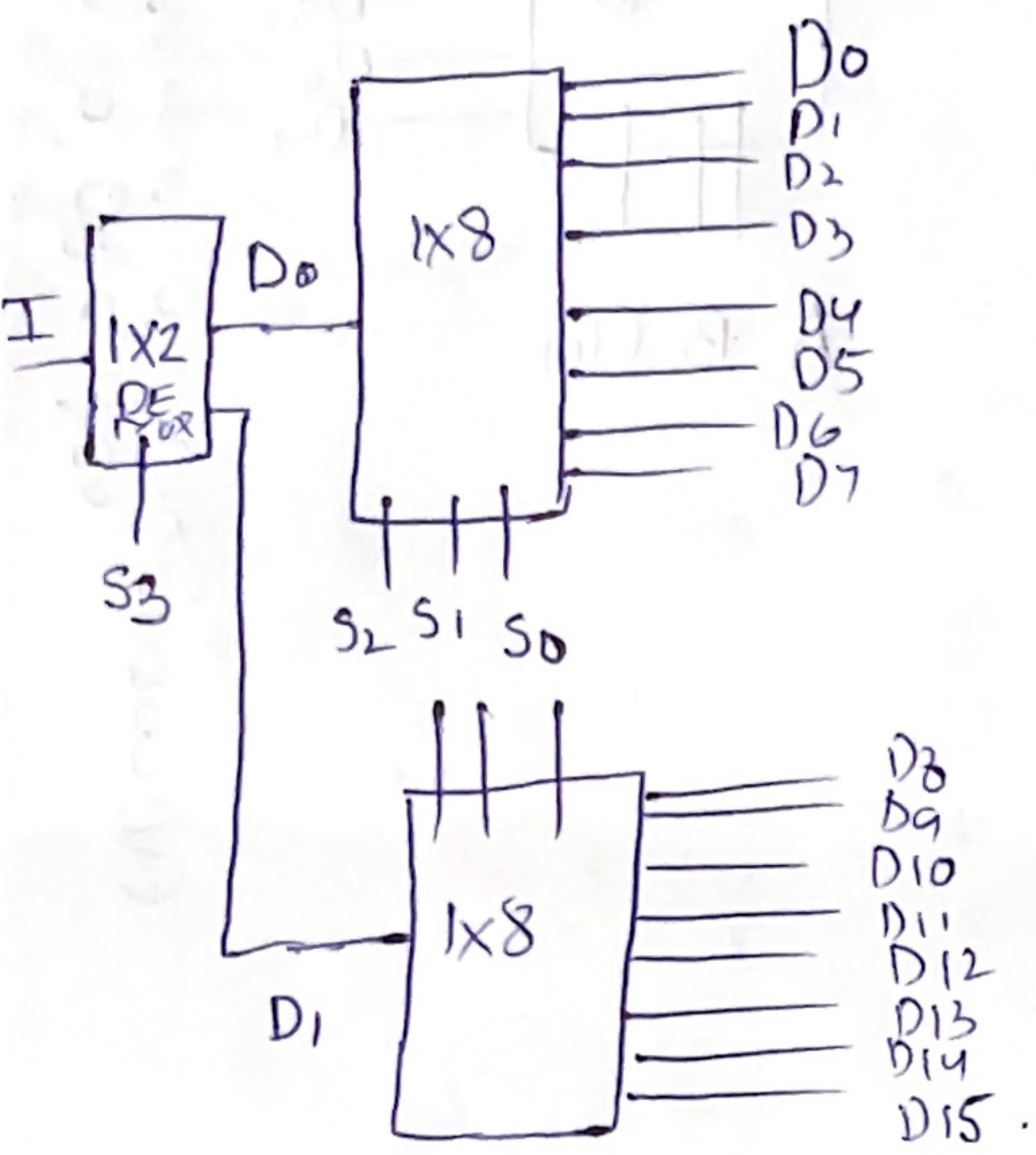


DEMUX CASCAADING.

As the size of operation in a MUX increases, it requires more inputs in and gate; we can cater this problem by cascading our multiplexers.

Example:-

1x16 DEMUX USING 1x8 DEMUX.

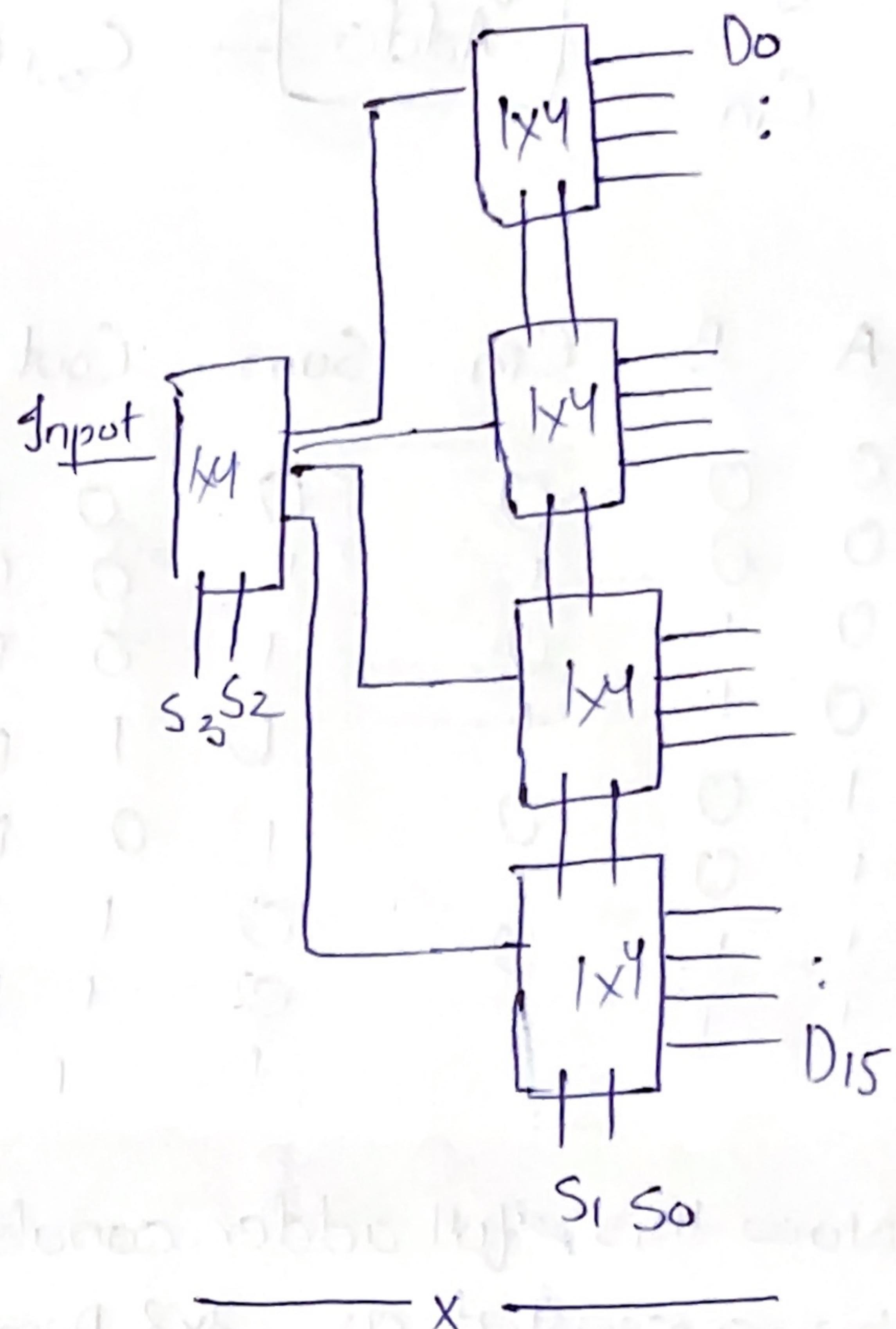


Now Remember

<u>S</u>	0	D0	} this is the pattern that 1x2 DEMUX will follow; same as 2x1 MUX.
	1	D1	

similary.

1x16 can also be made using 1x4 DEMUX.



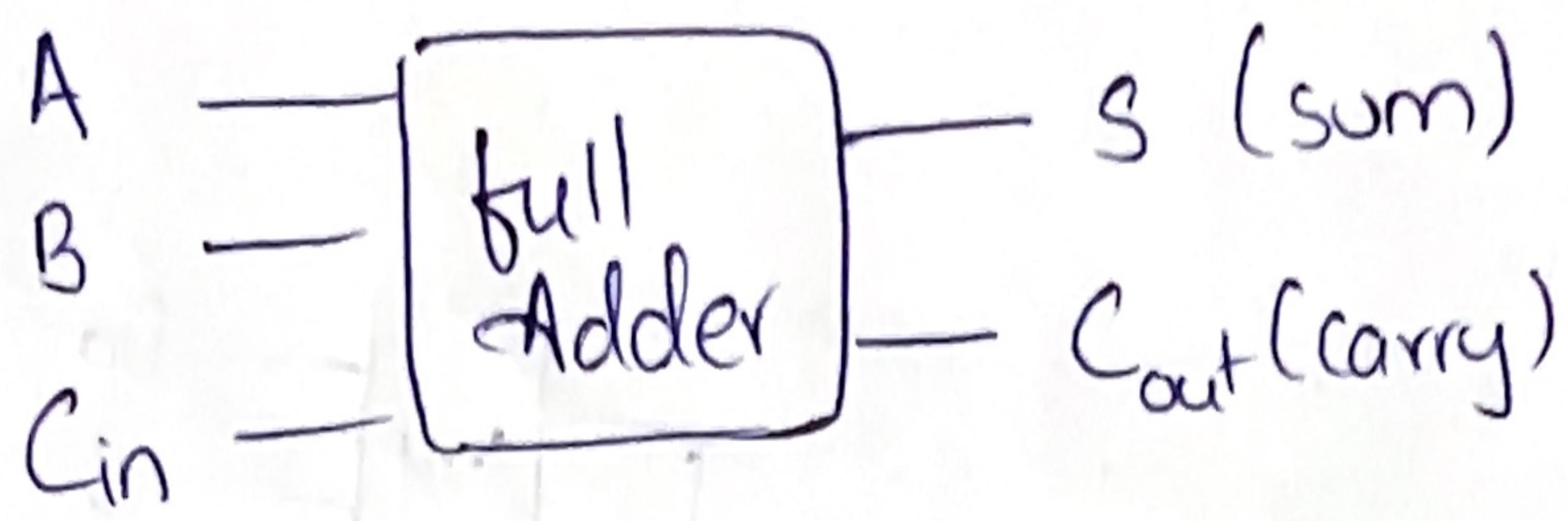
For Boolean function implementation;

no. of Boolean variables = no. of Selection lines.

① Implementing full Adder logic using DEMUX.

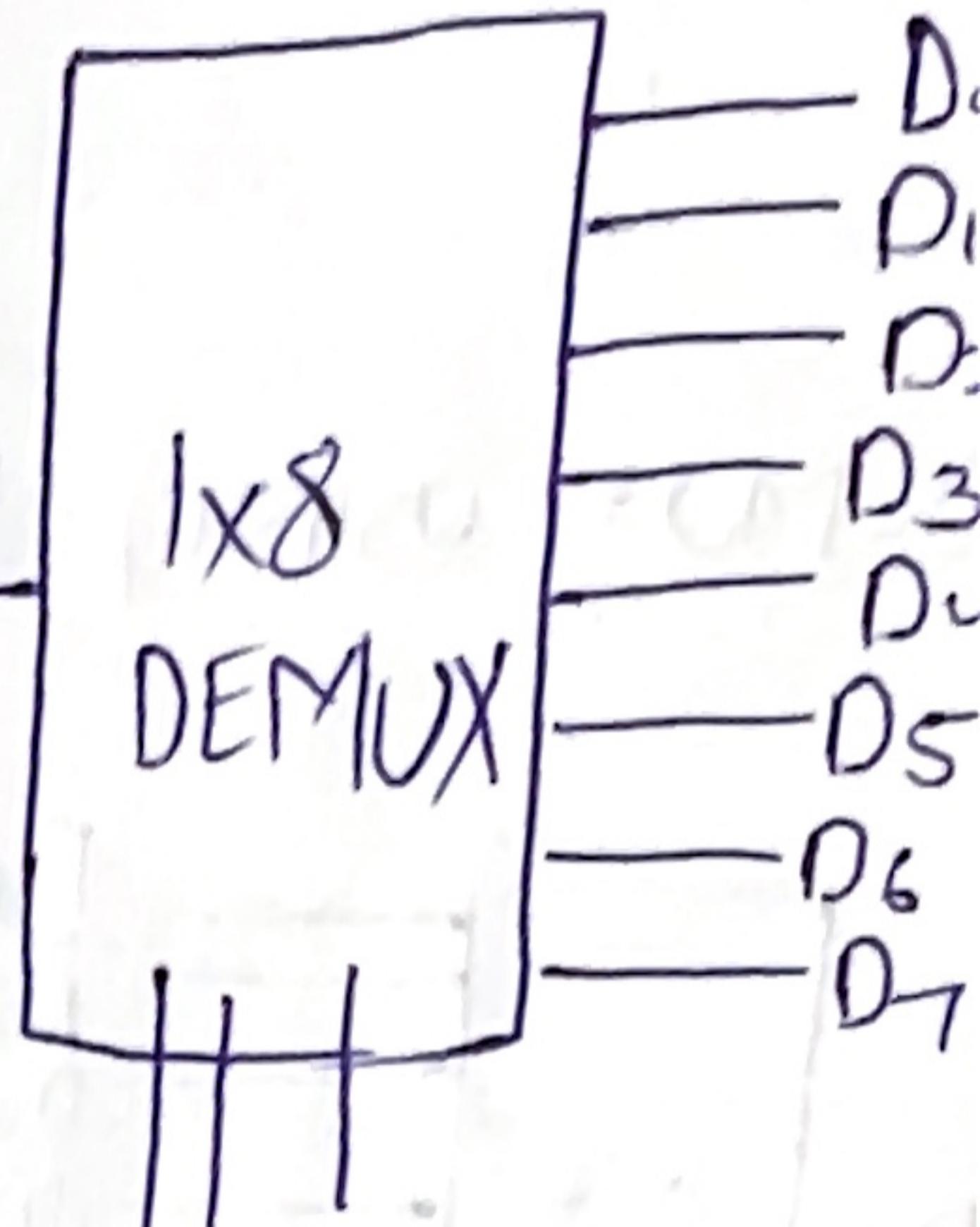
A full adder logic/Circuit

has three variables

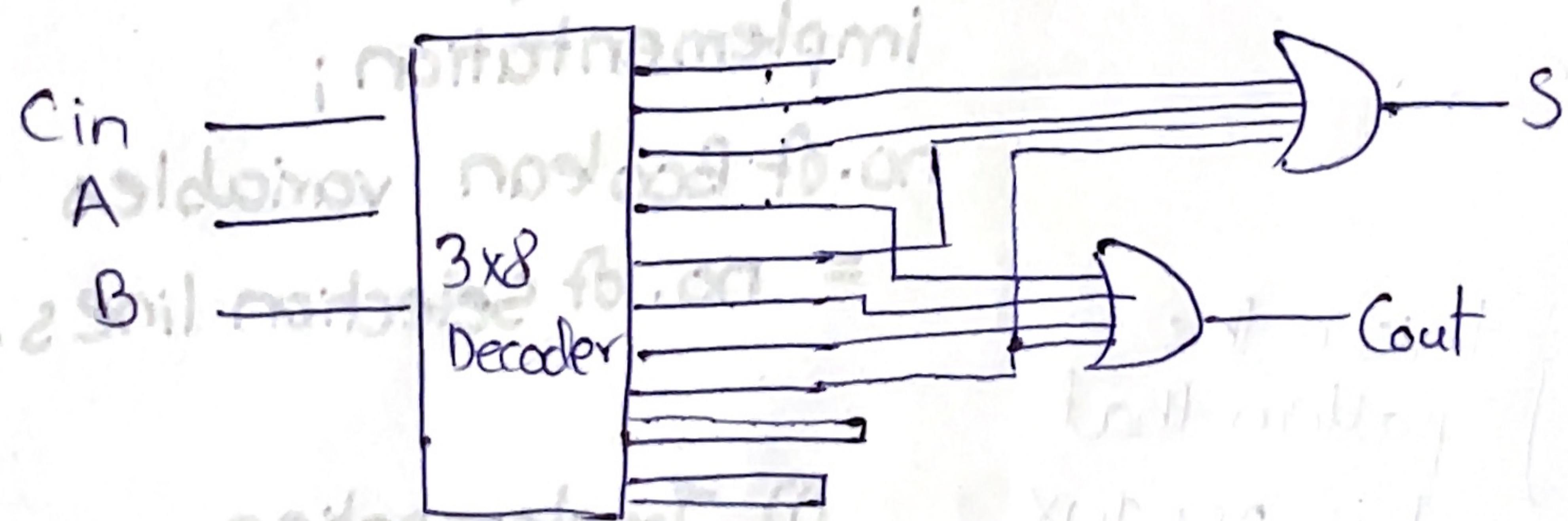


A B Cin Sum Cout Input:1

0	0	0	0	0	D0
0	0	1	1	0	D1
0	1	0	1	0	D2
0	1	1	0	1	D3
1	0	0	1	0	D4
1	0	1	0	1	D5
1	1	0	0	1	D6
1	1	1	1	1	D7



Now this full adder can also
be represented as 3x8 Decoder.



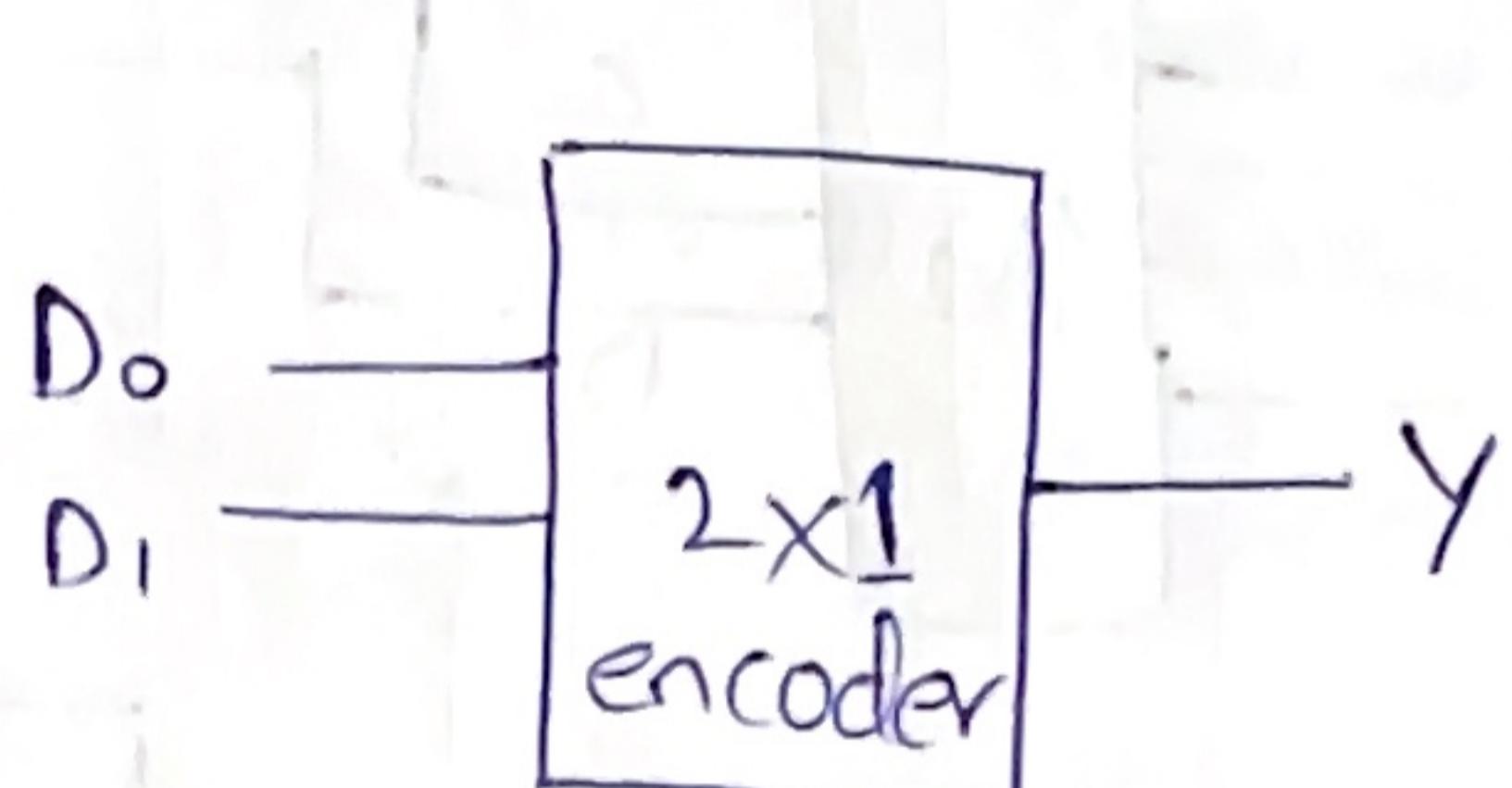
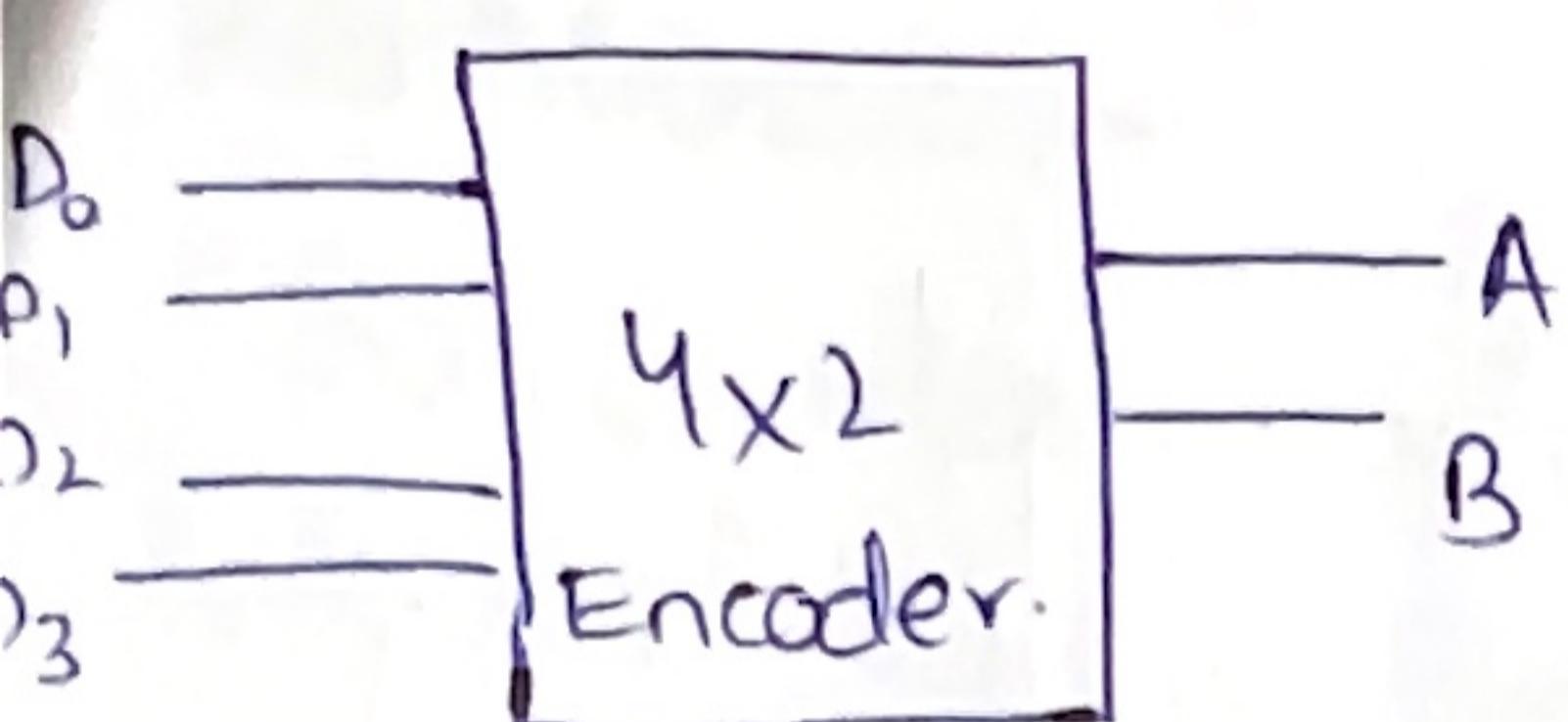
A LS : A^{LS} Magnitude of Bit Compared.

ENCODER:-

A combinational Circuit that has has 2^N inputs and N outputs.

If 2 inputs then 1 output.

4 inputs then 2 outputs.



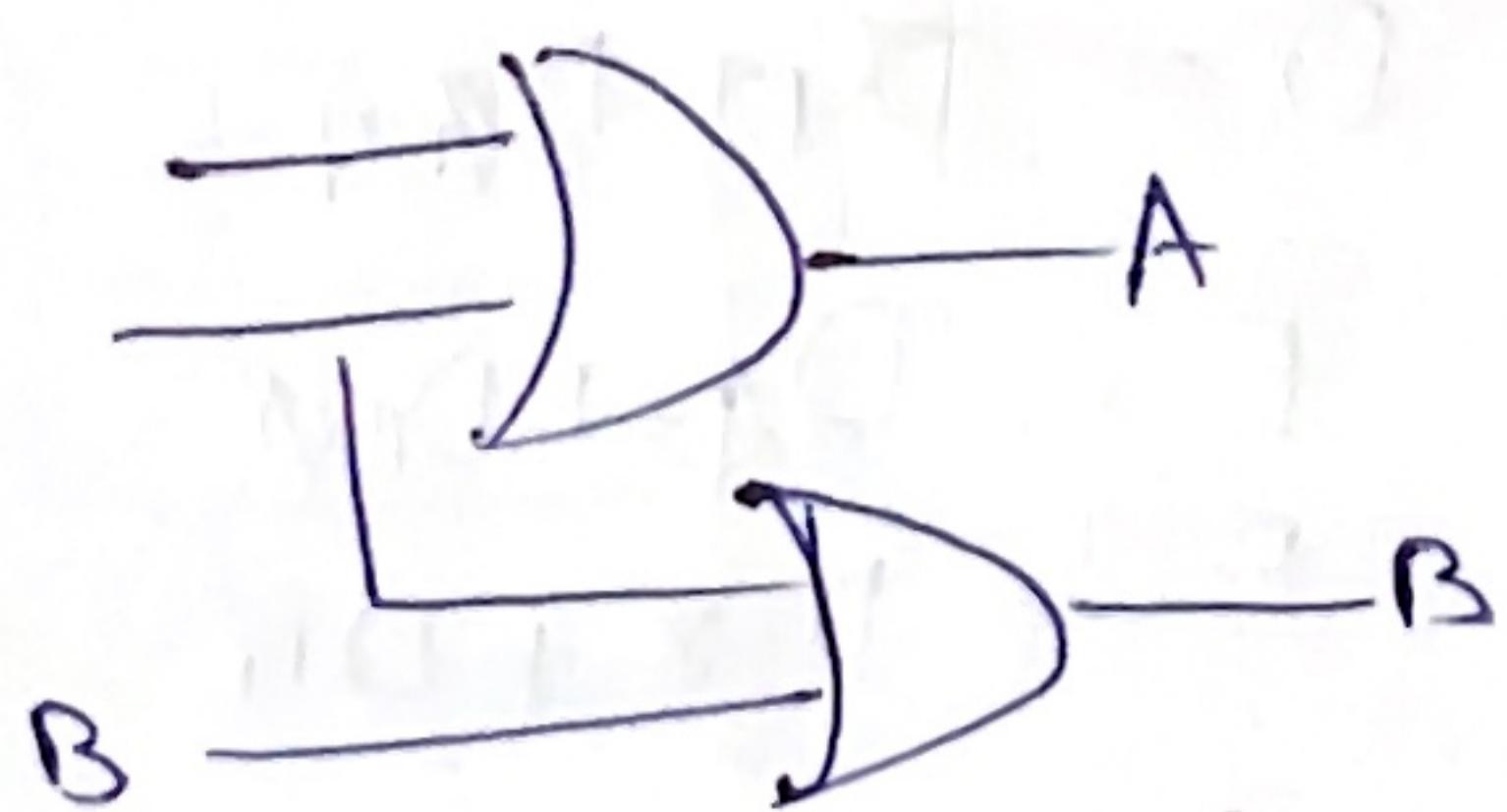
D ₀	D ₁	D ₂	D ₃	A	B
1				0	0
	1			0	1
		1		1	0
			1	1	1

D ₀	D ₁	Y
0	0	0
0	1	1
1	0	1
1	1	0

when I₁ = 0 & I₀ = 1

it is Y = 0.

I₁ = 1 & I₀ = 0
Y = 1.

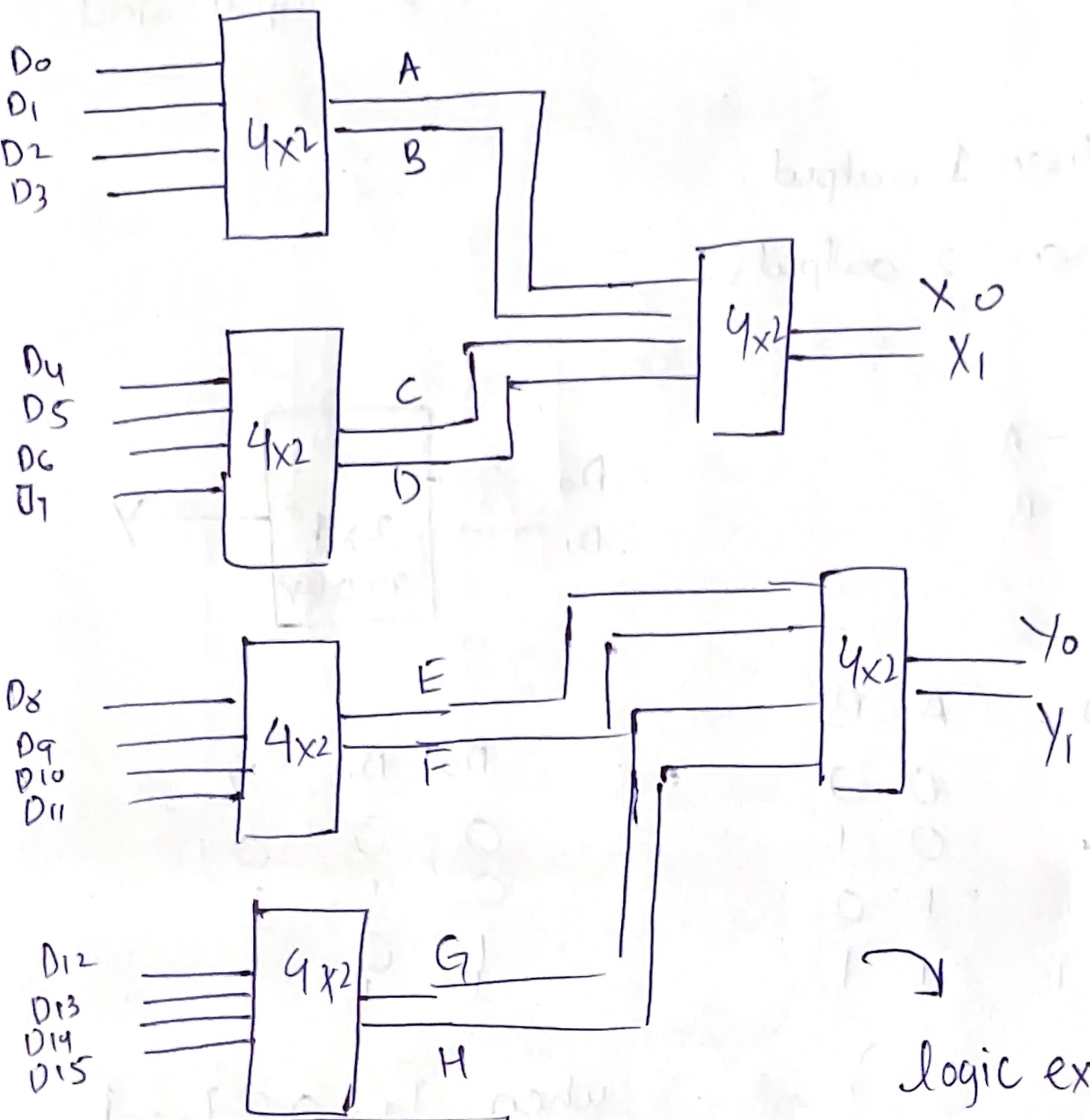


Similar procedure

when making

8x3 encoder.

16x4 encoder using 4x2.



Now Truth Table.
Same as for 16x4
encoder

Now

logic expression after
2nd stage. ←

$$X_0 = C + D \\ = D_6 + D_7 + D_4 + D_7$$

$$X_1 = A + D \\ = D_2 + D_3 + D_4 + D_7$$

$$\text{Similarly } Y_0 = G + H = D_{14} + D_{15} + D_{12} + D_{15}$$

$$A = D_2 + D_3$$

$$B = D_1 + D_3$$

$$C = D_5 + D_6 \oplus$$

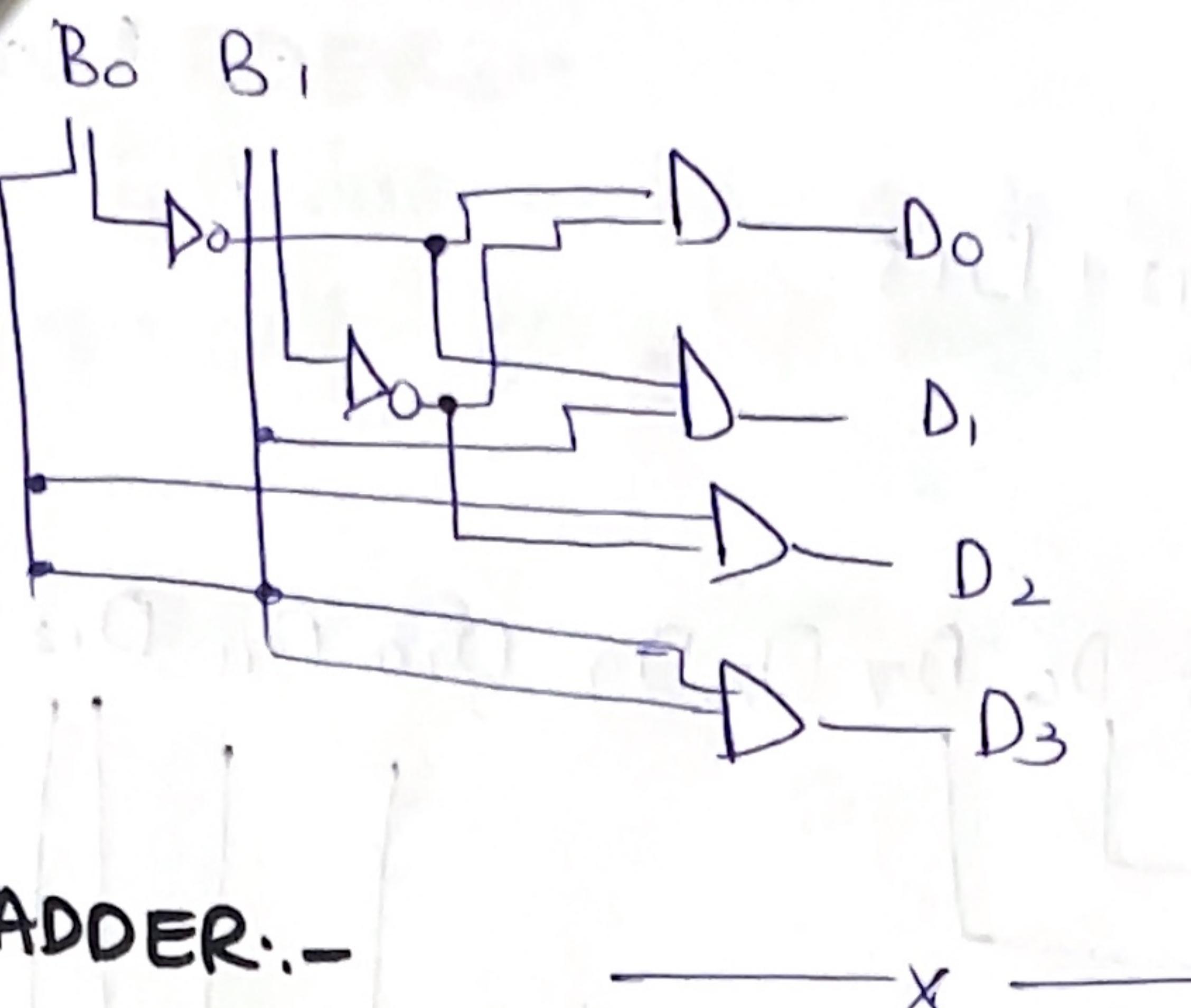
$$D = D_4 + D_6 \oplus$$

$$E = D_{10} + D_{11} \oplus$$

$$F = D_8 + D_{11}$$

$$G = D_{14} + D_{15}$$

$$H = D_{12} + D_{15}$$



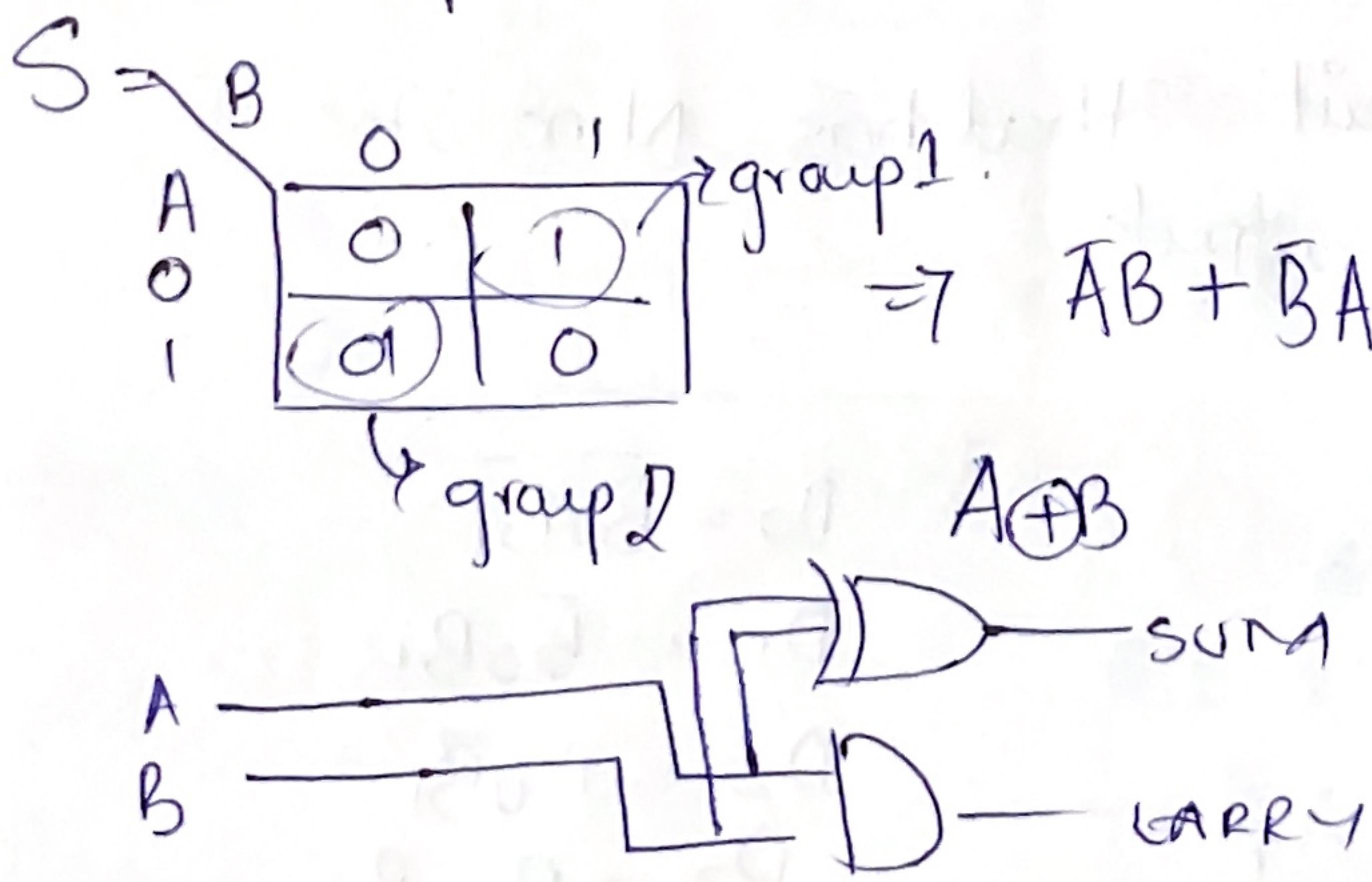
HALF ADDER:-

Basic Rules of addition are followed by two Binary input A & B in half adder.

A	B	S	Cout.
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Boolean expression.

By kmaps



Now the limited inputs are basically increasing the no. of outputs.

that is why a full adder with minterms at s and carry out part is expressed

using a (3×8) decoder

inputs

A
B
cin

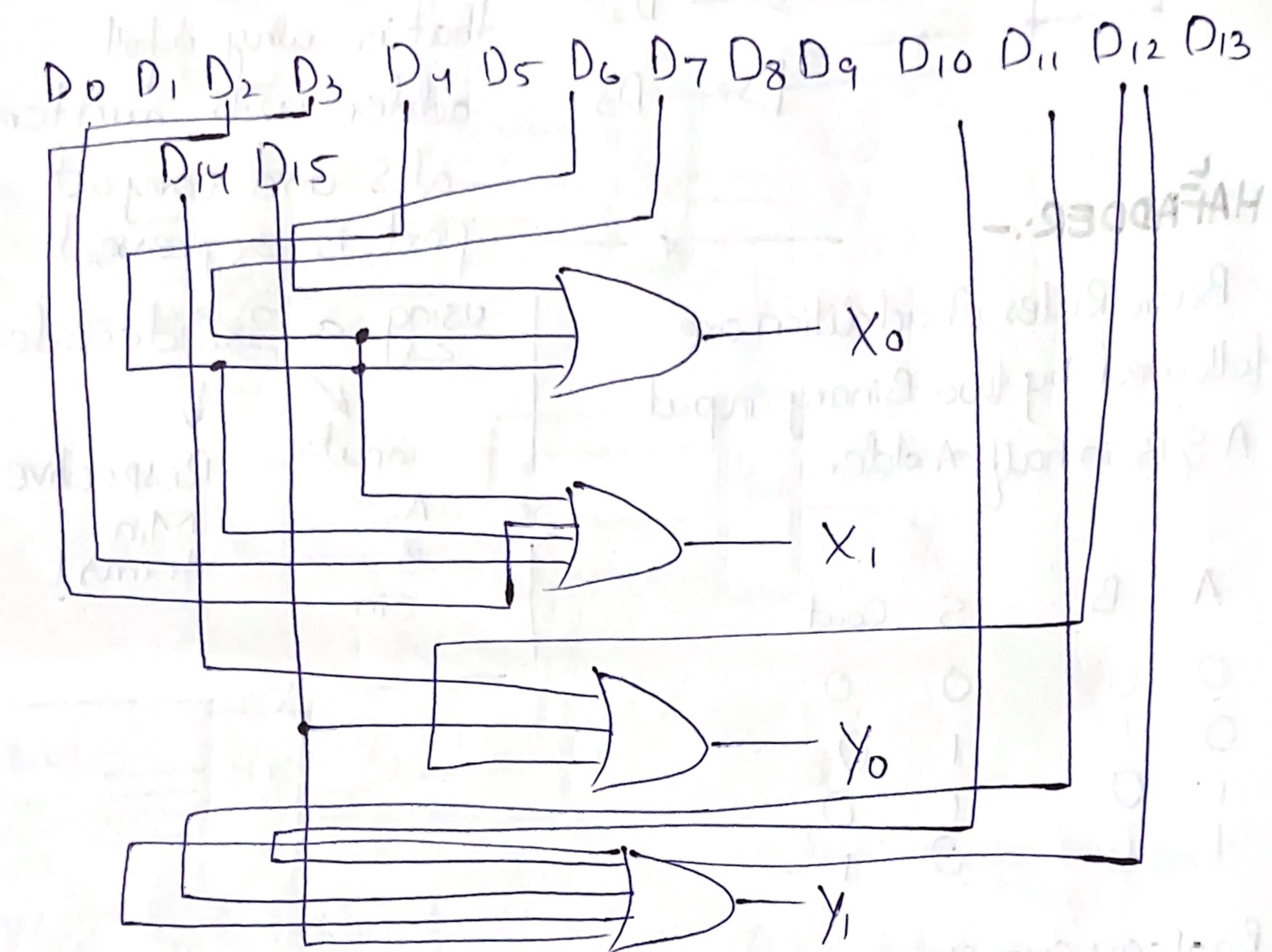
Respective
Min
terms!

C	0	1
A	0	0
B	0	1

$$C = AB$$

$$Y_1 = E + H \\ = D_{10} + D_{11} + D_{12} + D_{15}.$$

Now Logic Circuit:



Decoder:-

A combinational circuit that has N no. of inputs and 2^N no. of outputs.

2×4 Decoder.

B_0	B_1	D_0	D_1	D_2	D_3
0	0	1			
0	1		1		
1	0			1	
1	1				1

$$D_0 = \bar{B}_0 \bar{B}_1$$

$$D_1 = \bar{B}_0 B_1$$

$$D_2 = B_0 \bar{B}_1$$

$$D_3 = B_0 B_1$$

FULL ADDERS:-

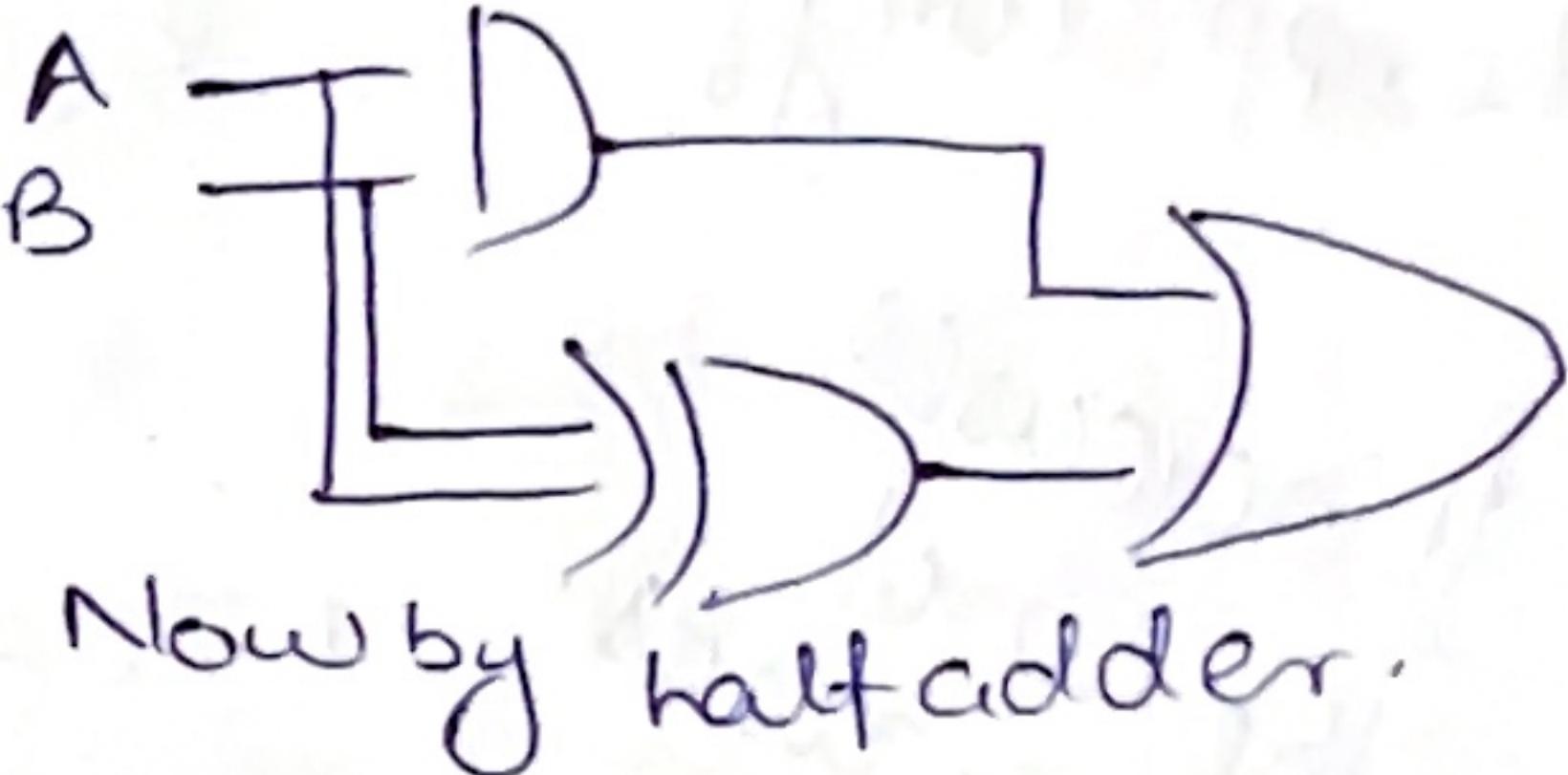
Now full adder works in combination with 2 halfadders.

Specifically for 2 Bits and 1 incoming carry Bit.

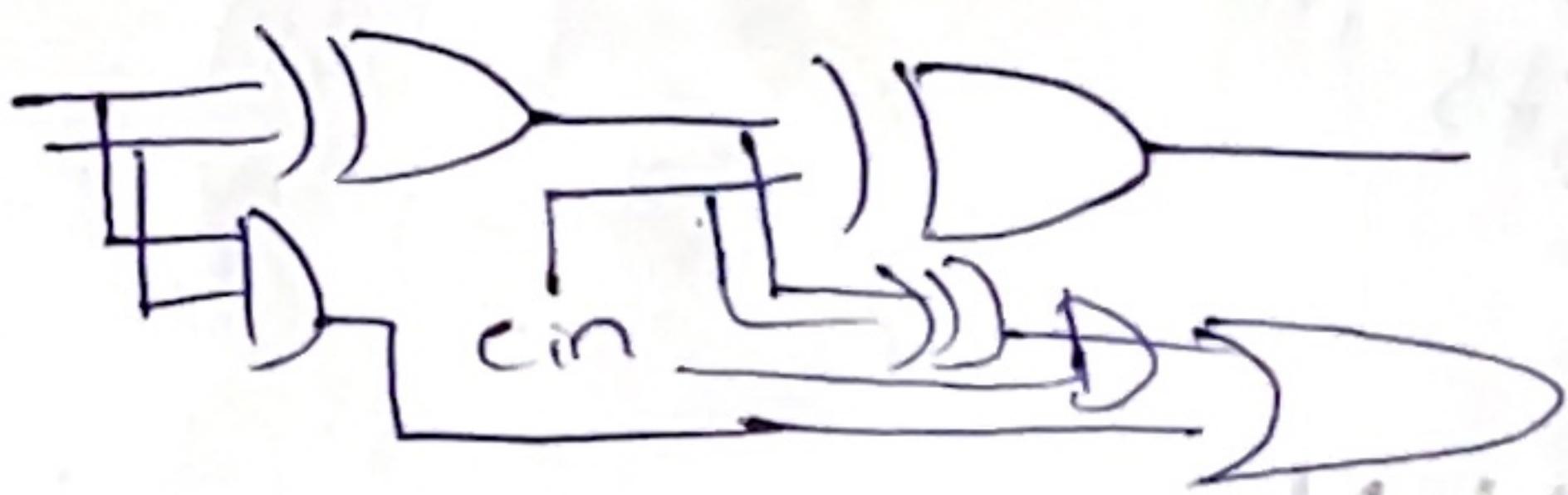
A	B	Cin	SUM	CARRY
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Similarly for carry
the simplified
expression is.

$$AB + \text{Cin} (A \oplus B)$$



Now by half adder.



$$\overline{A}\overline{B}\overline{\text{Cin}} + A\overline{B}\overline{\text{Cin}} + A\overline{B}C$$

Now simplifying.

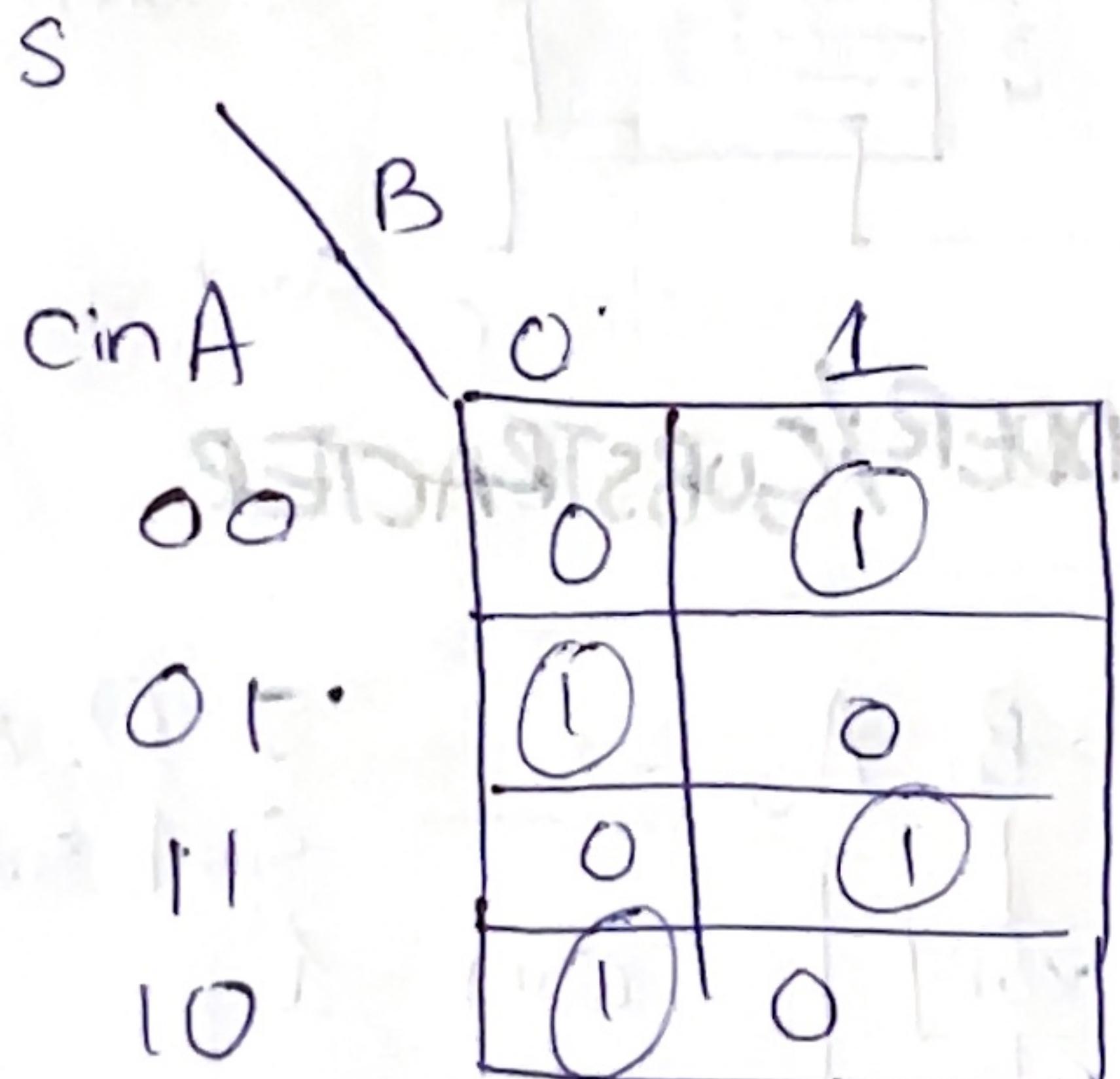
$$\overline{A}(B\overline{\text{Cin}} + \overline{B}\text{Cin}) + A(B\overline{\text{Cin}} + BC\text{Cin})$$

$$\overline{A}(B \oplus \text{Cin}) + A(\overline{B} \oplus \text{Cin})$$

$$\text{let } X = A \oplus \text{Cin}$$

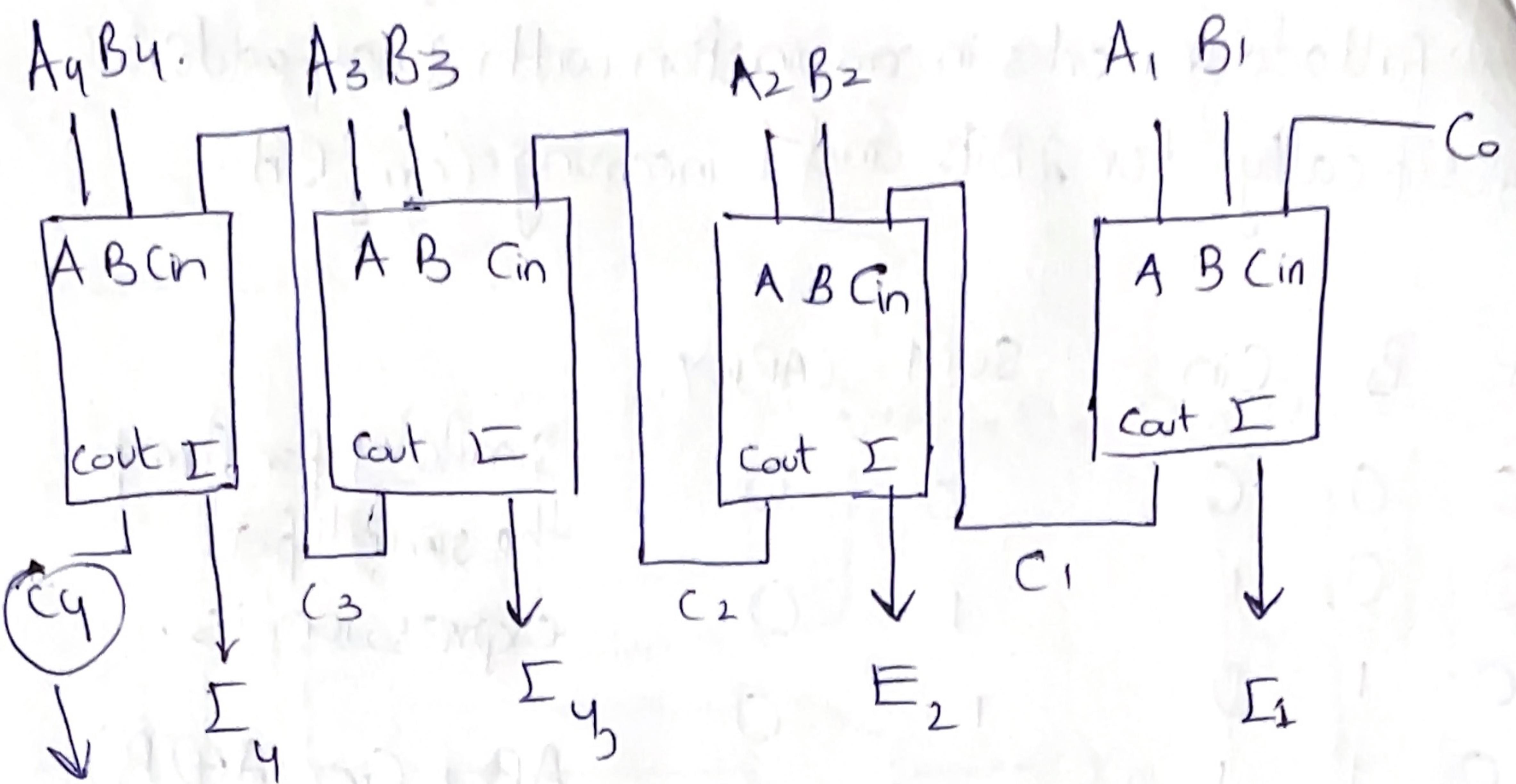
$$\overline{A}X + A\overline{X} \Rightarrow A \oplus X \\ \Rightarrow \overline{A} \oplus B \oplus \text{Cin}$$

Now By Kmaps.



4BIT PARALLEL ADDERS:-

(SUB: 4BIT
B with NOT
-12340011C_n
A = 1
C_n = 1)

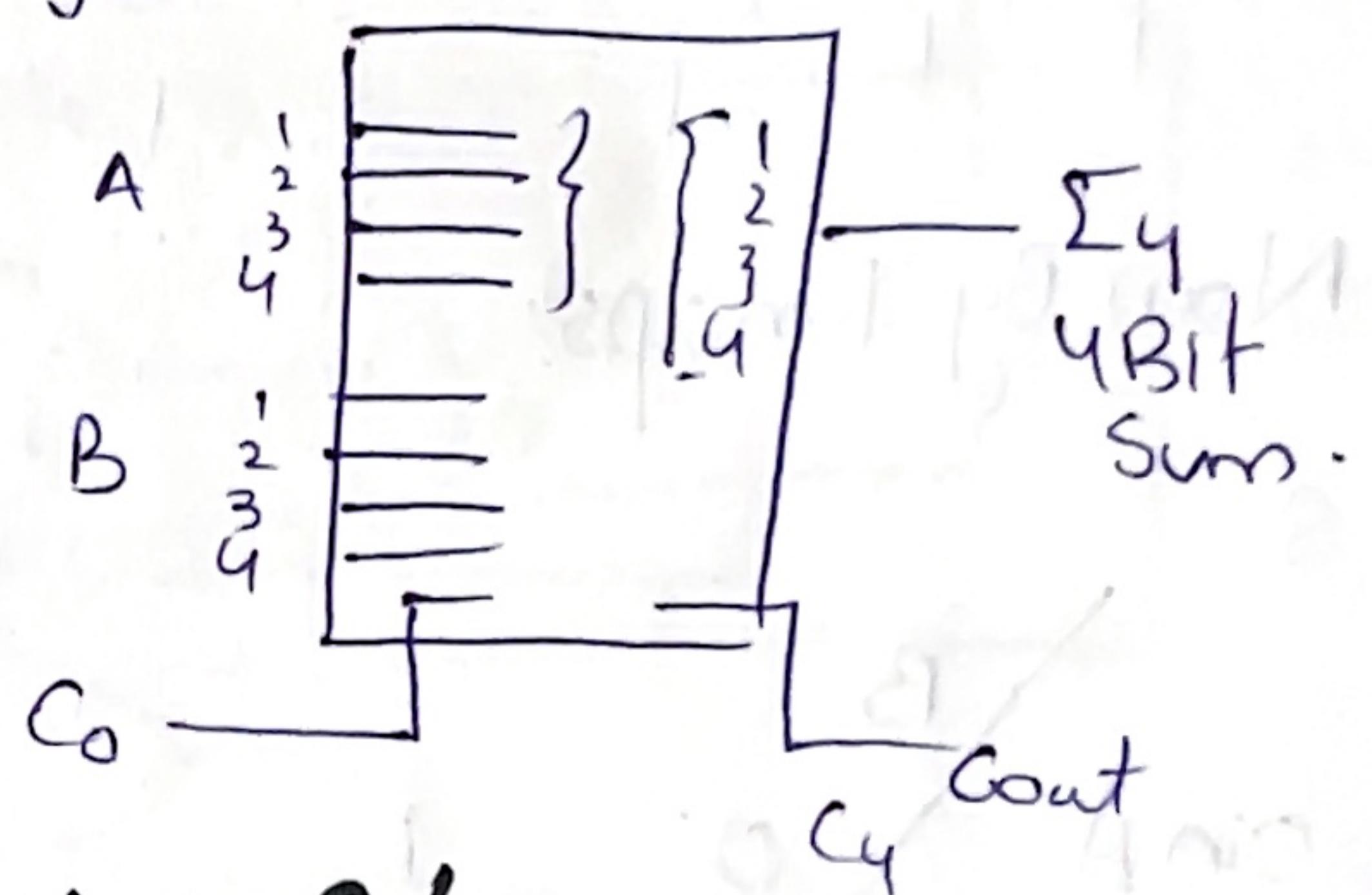


the ripple carry!

It increases as the no. of bits increases.

For a four Bit Parallel Adder

logic Symbol :-



NON 4 Bit / GENERAL PARALLEL ADDER / SUBTRACTER.

