COE 272
Digital Systems

Lecture 5: Combinational Circuit (Design Procedures, Adders, Subtractors)

- Combinational logic design procedure
- Classification of combinational circuits
- Adder circuit
- Subtractor circuit
- Multiplexer circuit
- Decoder circuit

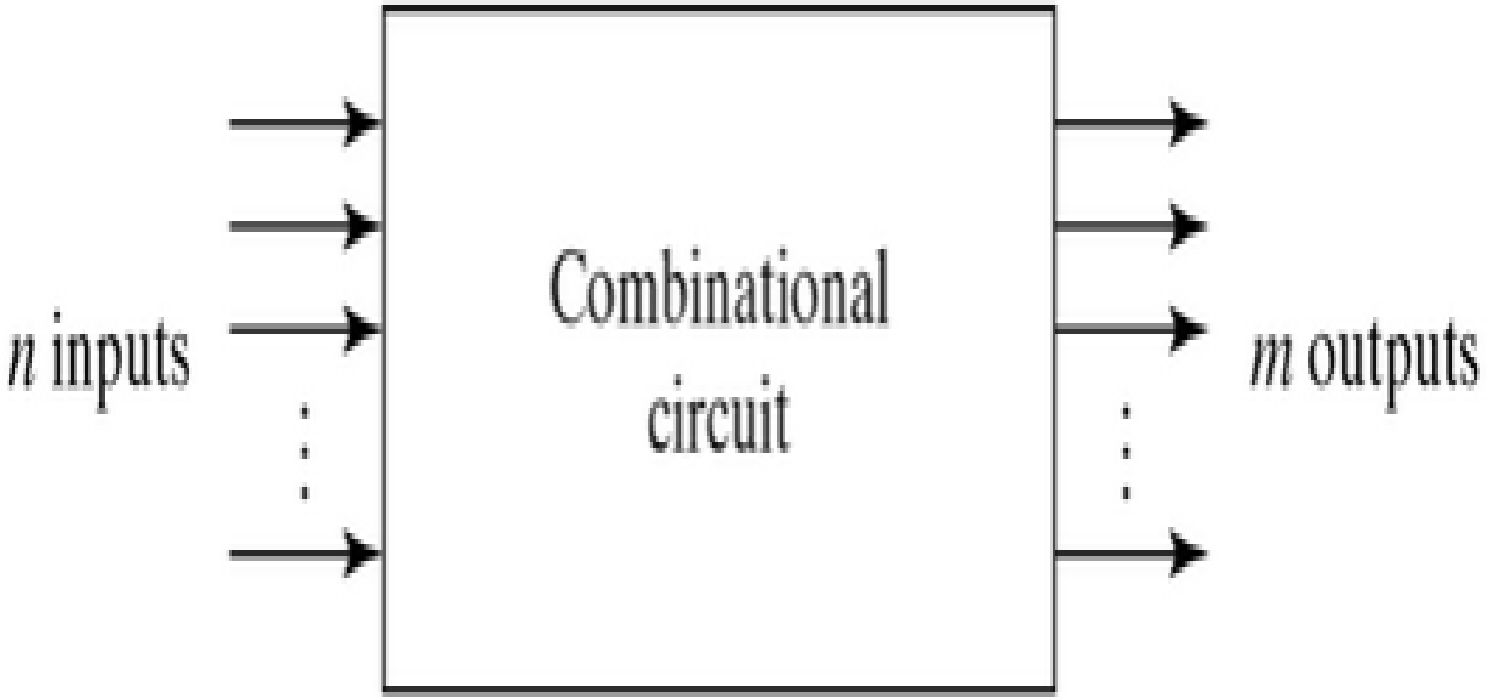
Combinational Circuit

- Combinational circuits are defined as logic circuits whose output of which depends only upon the combination of inputs
- The output does not depend on past values of inputs or outputs.
- Thus, combinational circuits do not need any memory
- Combinational circuits can have a number of inputs and a number of outputs

Combinational Circuit

- Combinational circuits operates in the following three steps
  - It accepts n-different inputs
  - The combination of gates operates on the inputs
  - m-different outputs are produced as per the requirement.

Combinational Circuit



Combinational Logic Design Procedure

- The various steps involved in designing a combinational logic are listed as:
  - A problem is given
  - The number of inputs and outputs are determined, and letter symbols are assigned to input and output variables
  - Generate a truth table relating the inputs and outputs
  - Create a K-map for each output and obtain the simplified Boolean expression for each output.
  - Draw the logic Diagram (combinational circuit)

Example

- A circuit has four inputs and two outputs. One of the output is high when majority of inputs are high. The second output is high only when all inputs are of same type. Design the combinational circuit

Solution

1. Assign symbols to inputs and output variables

   Let the four inputs be A, B, C, D and the two outputs be Y1 and Y2.

2. Write the truth table

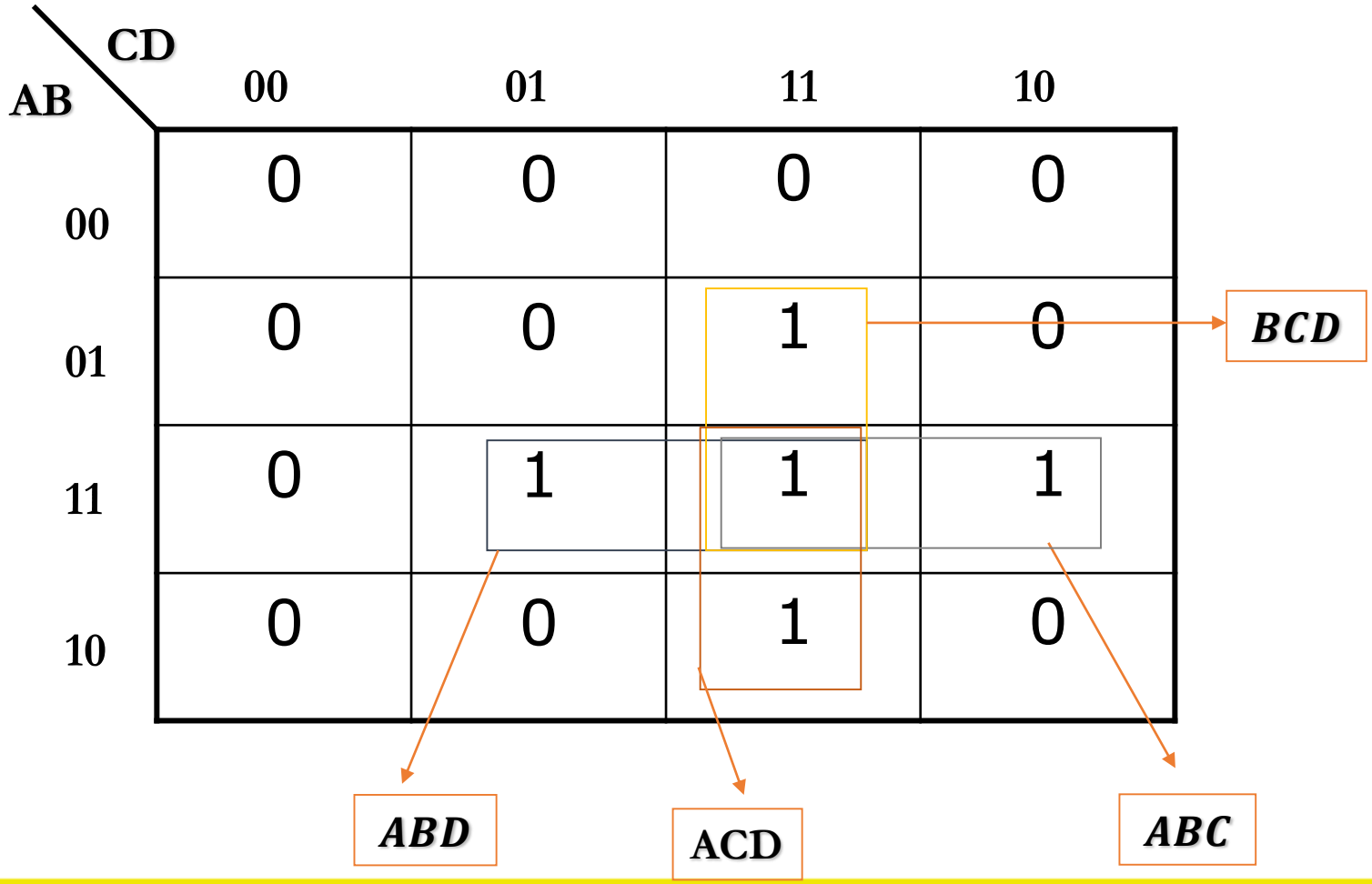
	Inputs				Outputs	
Decimal	A	B	C	D	Y_1	Y_2
0	0	0	0	0	0	1
1	0	0	0	1	0	0
2	0	0	1	0	0	0
3	0	0	1	1	0	0
4	0	1	0	0	0	0
5	0	1	0	1	0	0
6	0	1	1	0	0	0
7	0	1	1	1	1	0
8	1	0	0	0	0	0
9	1	0	0	1	0	0
10	1	0	1	0	0	0
11	1	0	1	1	1	0
12	1	1	0	0	0	0
13	1	1	0	1	1	0
14	1	1	1	0	1	0
15	1	1	1	1	1	1

Solution

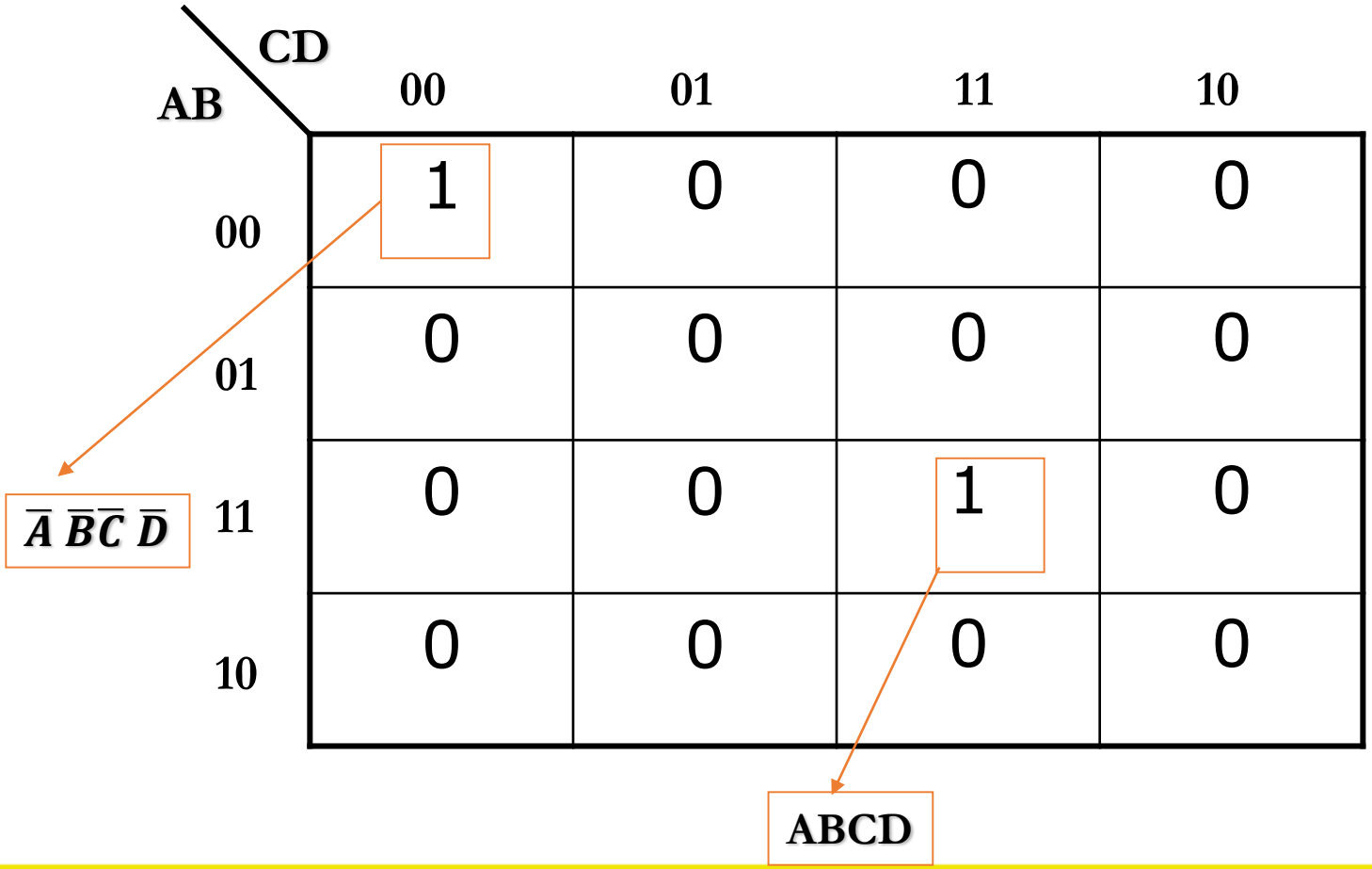
- From the truth table, the following are observed:
  - $Y_1 = 1$, *when the number of 1 inputs are greater than the number of 0 inputs*
  - $Y_2 = 1$, *when* $A=B=C=D$

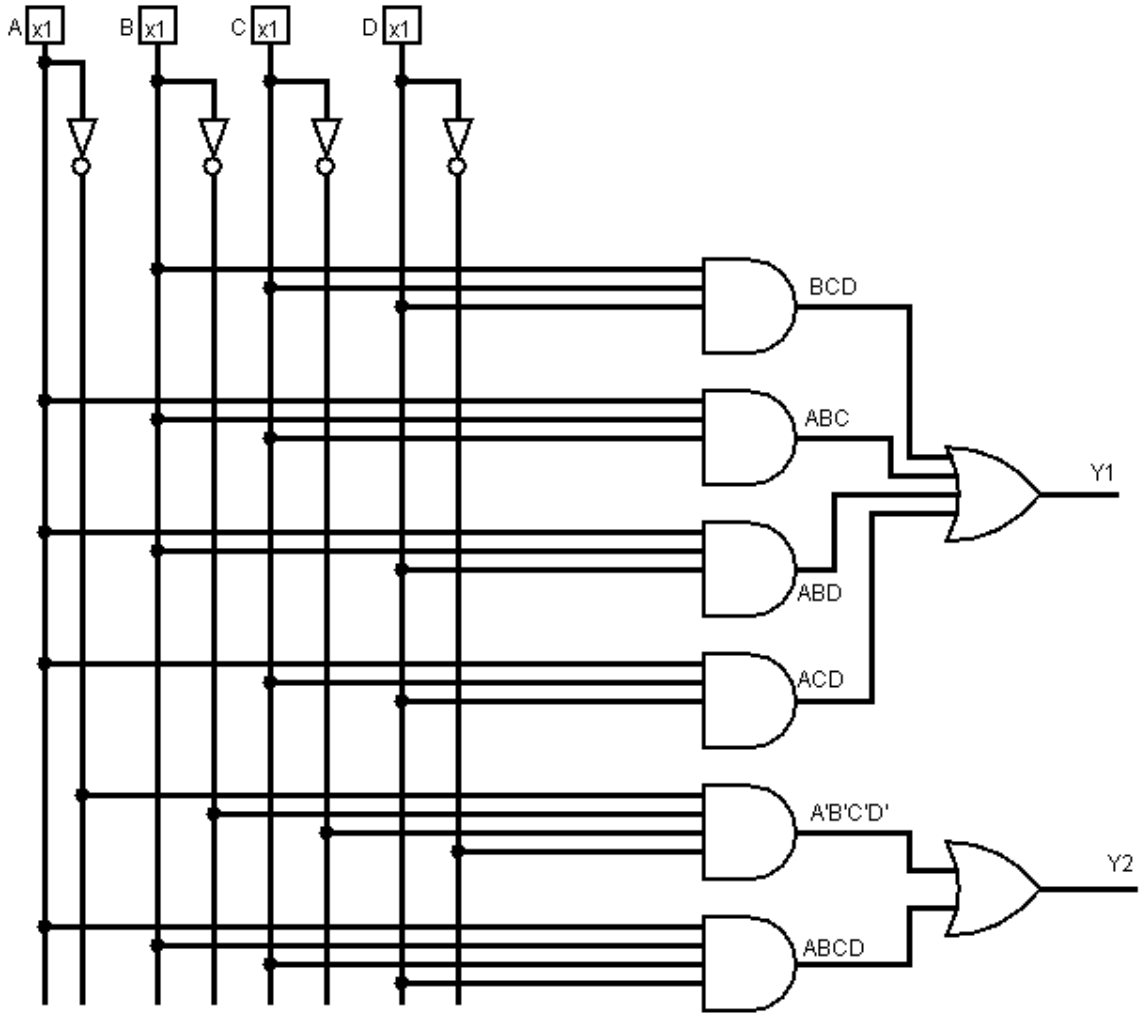
3. Draw K-maps for each of the two outputs and get simplified expressions

K-Map for Output Y_1



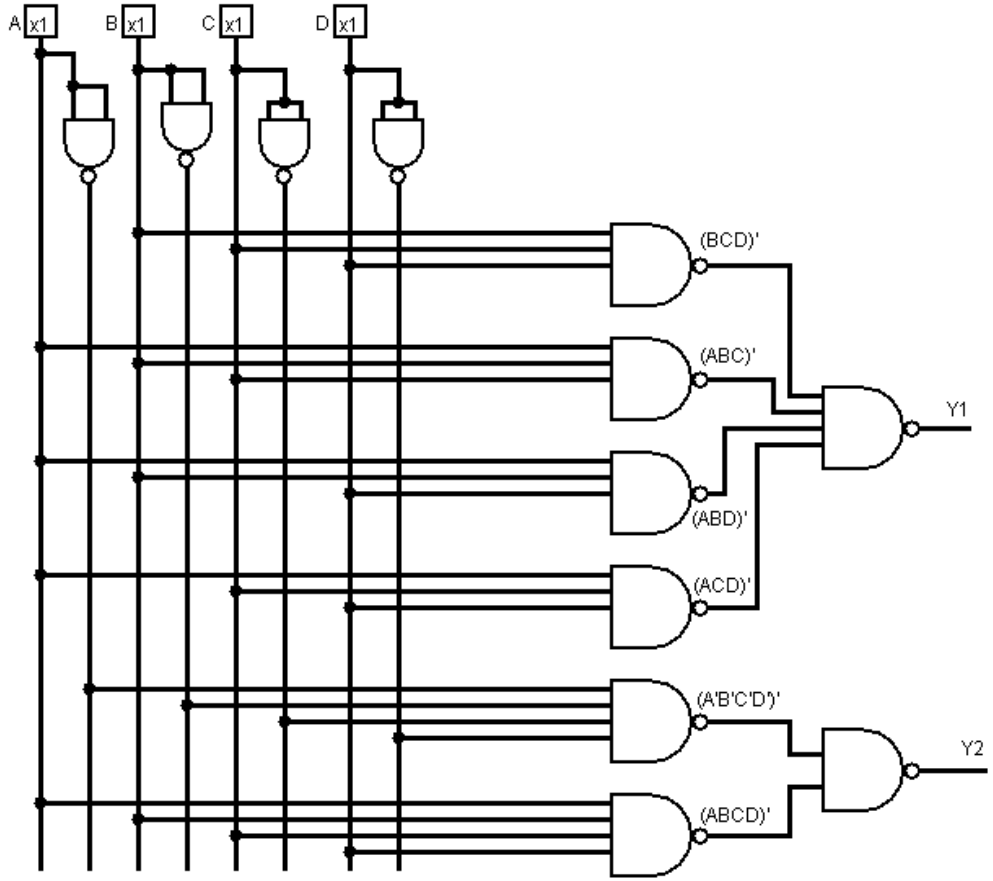
K-Map for Output Y_2




A
x1
B
x1
C
x1
D
x1
BCD
ABC
ABD
ACD
Y1
A'B'C'D'
ABCD
Y2

Example

- Implement the above example using only NAND gates



Classification of Combinational Circuits

- Combinational circuits can be classified under:
  - Code converters
  - Binary and BCD adders
  - Binary and BCD subtractors
  - Digital Comparators

Binary Adders

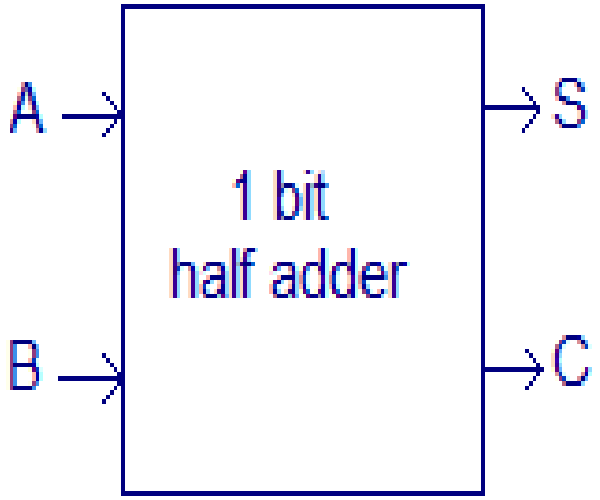
- Addition of two binary digits is a very basic operation performed by digital computers.
- Digital circuits that are used to add two binary numbers are called binary adders
- They are of two forms:
  - Half Adder
  - Full Adder

Half Adder

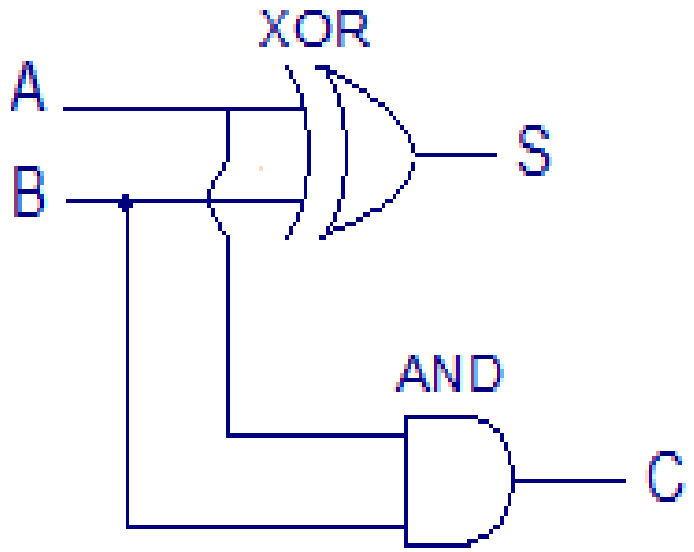
- A half adder is a combinational logic circuit with two inputs and two outputs.
- It is the basic building block for addition of two single bit numbers.
- The outputs of the circuit is namely carry and sum.

Inputs		Outputs	
A	B	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

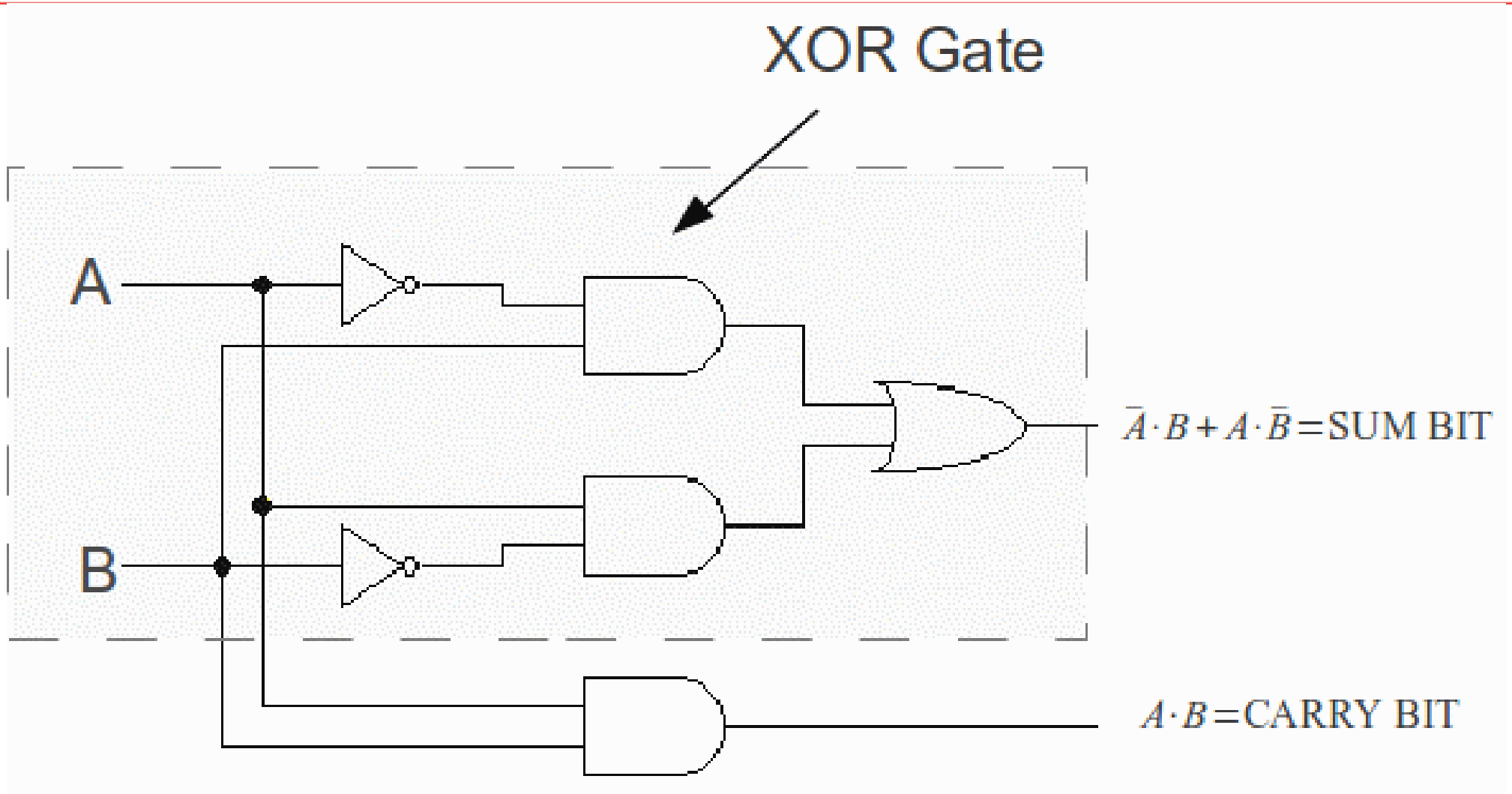
Truth table

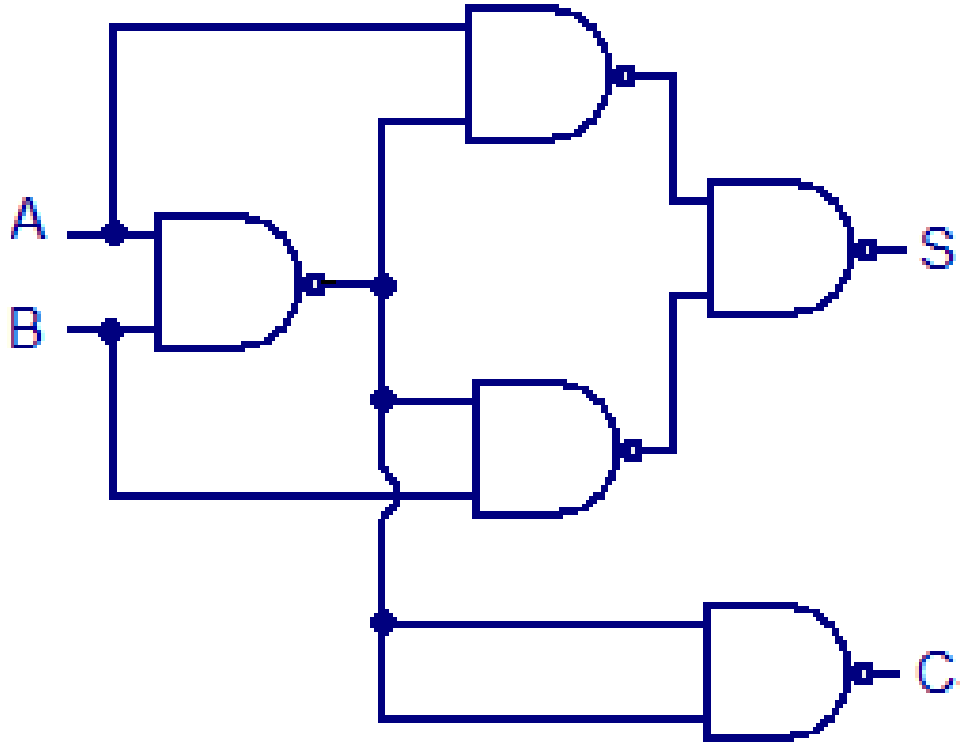


Schematic

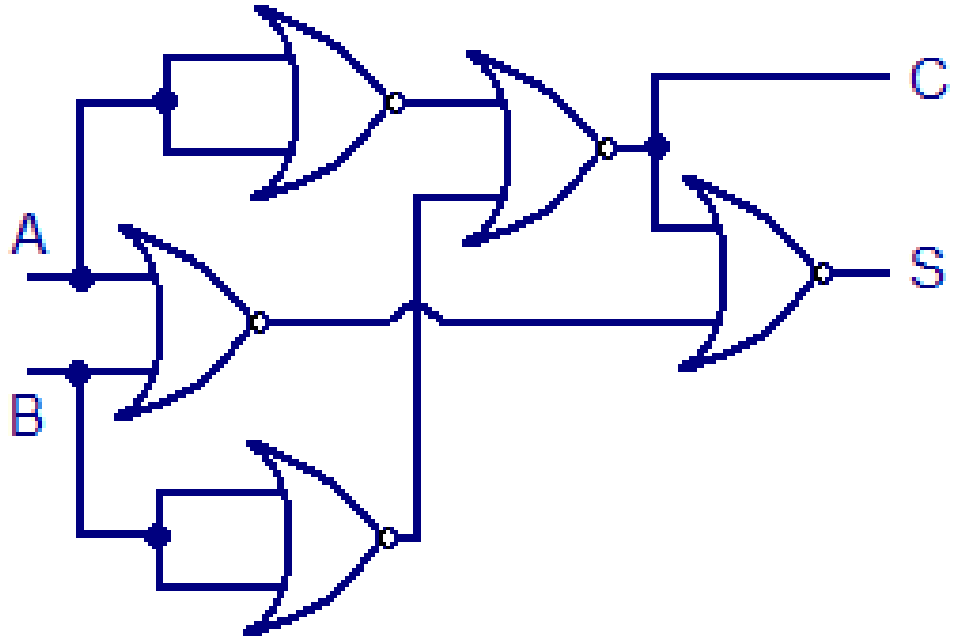


Realization


XOR Gate
A
B
$\bar{A} \cdot B + A \cdot \bar{B}$ = SUM BIT
$A \cdot B$ = CARRY BIT



Half adder using NAND logic



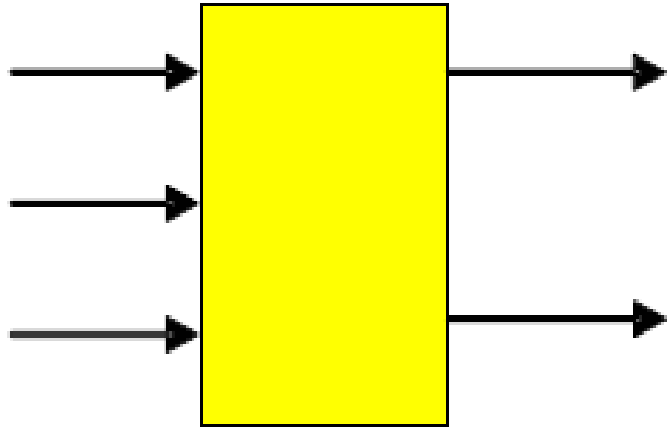
Half adder using NOR logic

Full Adder

- A full Adder (FA) is an arithmetic circuit that is used to **add three bits**

- The inputs of the FA are the 3 bits to be added;
  - *the augend, addend, and carry from previous lower significant position.*

- The output is the result of this addition,
  - *i.e. a sum bit (S) and a carry bit (C)*



Truth Table of a full Adder

Inputs			Outputs	
A	B	Cin	S	Co
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

Expression for sum output

$S = \bar{A}\bar{B}Cin + \bar{A}B\overline{Cin} + ABCin + A\bar{B}\overline{Cin}$

$S = Cin(\underbrace{\bar{A}\bar{B} + AB}_{\text{EX-NOR}}) + \overline{Cin}(\underbrace{\bar{A}B + A\bar{B}}_{\text{EX-OR}})$

Therefore, $S = Cin(\bar{A}\bar{B} + AB) + \overline{Cin}(\bar{A}B + A\bar{B})$

Let $X = \bar{A}B + A\bar{B}$

Therefore $S = Cin\bar{X} + \overline{Cin}\ X = Cin \oplus X$

or $S = Cin \oplus (\bar{A}B + A\bar{B})$

But , $\bar{A}B + A\bar{B}$

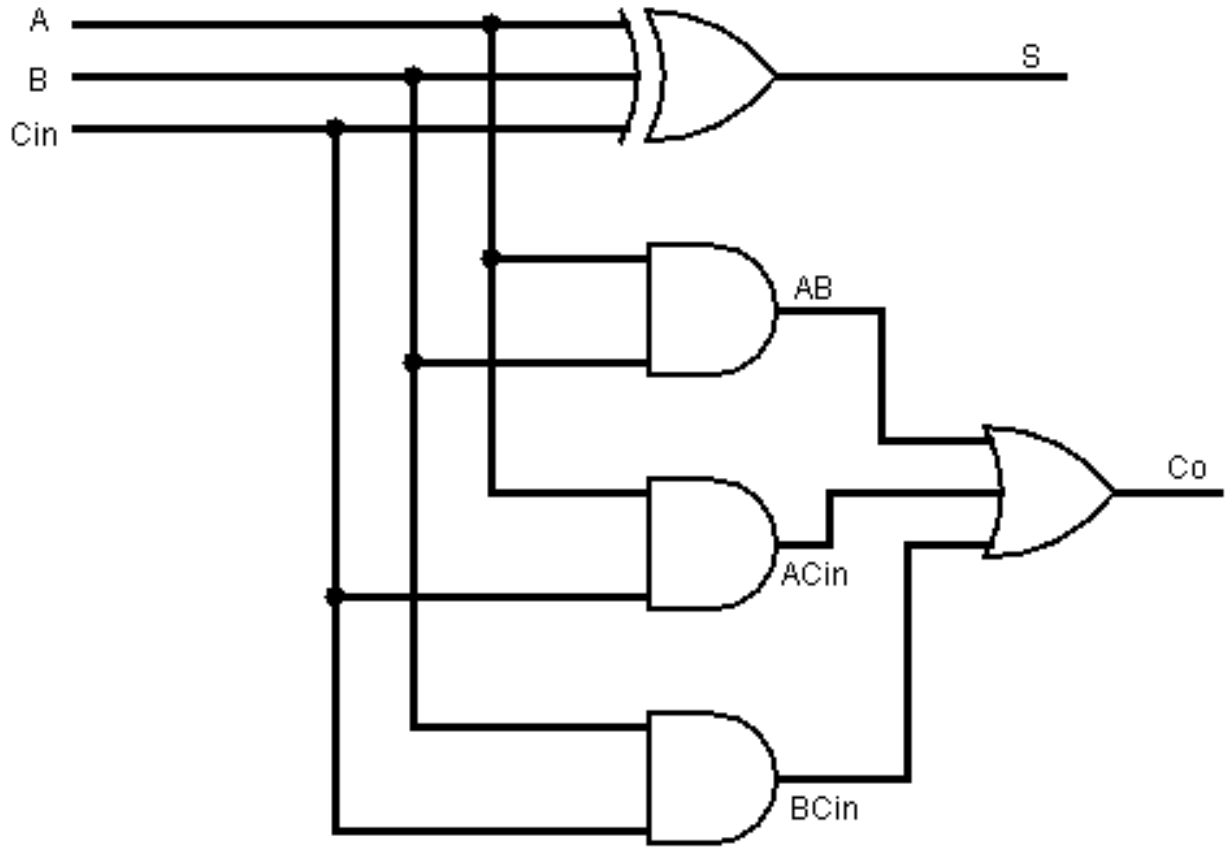
Therefore, **$S = Cin \oplus A \oplus B$**

Expression for Co Output

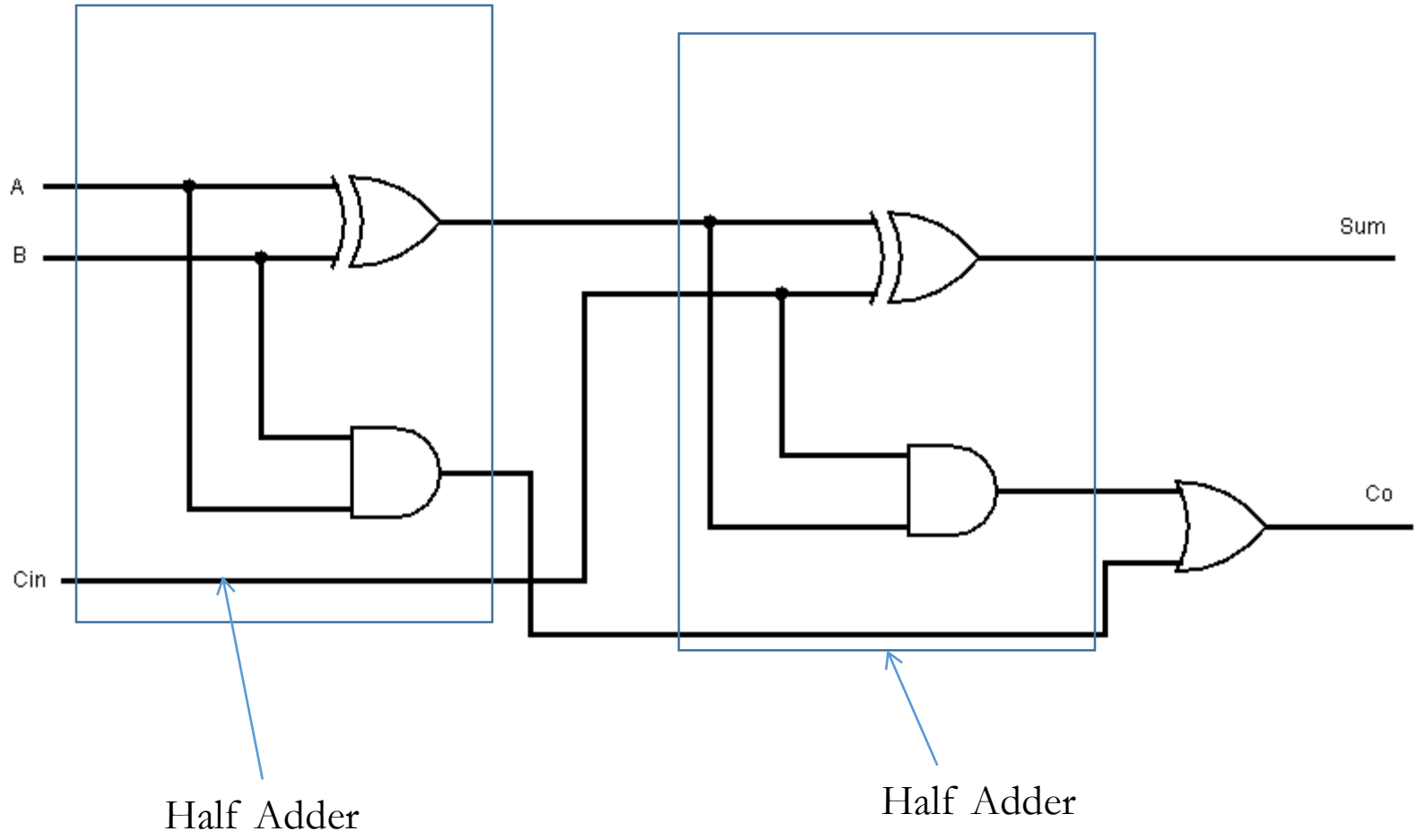
- Co = $\bar{A}$BCin + A$\bar{B}$Cin + ABCin + AB$\overline{Cin}$

- Co = AB + ACin + BCin

Logic Diagram for Full Adder



Full Adder using Half Adder



Applications of a Full Adder

- It acts as the basic building block of the following Adder ICs:
  - 4 bit
  - 8 bit
  - BCD

Binary Subtractor

- Rules of binary subtraction are:
- 0 – 0 = 0
- 0 – 1 = 1 with borrow of 1
- 1 – 0 = 1
- 1 – 1 = 0

Classification of Binary Subtractors

- Binary subtractors are classified as follows:
  - Half Subtractor
  - Full Subtractor

Half Subtractor

- Half subtractor is a combinational circuit with two inputs and two outputs (difference and borrower)

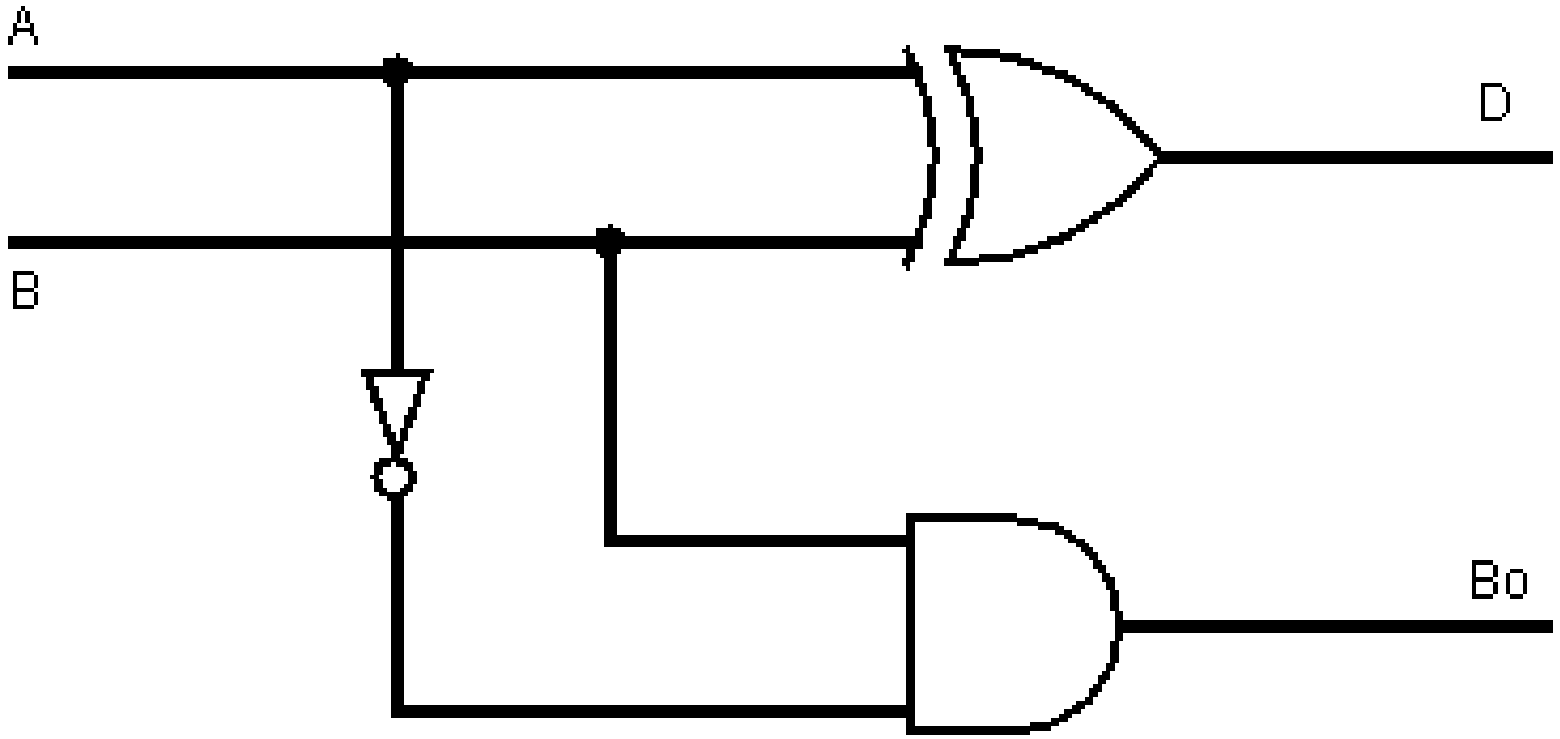
- In subtraction (A – B),
  - *A is known as the minuend and the B known as the subtrahend bit.*

Truth table for a half subtractor

Inputs		Outputs	
A	B	Difference D	Borrow Bo
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

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

Hence $D = A \oplus B$


A
B
D
Bo

Drawback of Half Subtractor

- A half subtractor can only perform the subtraction of two binary bits.
- While performing subtraction, it does not consider the borrow of the lower significant stage

Full Subtractor

- Drawbacks of half subtractor can be mitigated using the full subtractor

Inputs			Outputs	
A (Minuend)	B (subtrahend)	Bin (Previous borrow)	(A - B - Bin)	Bo
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

Output Expressions

- D = $\bar{A}\bar{B}$Bin + $\bar{A}$B$\overline{Bin}$ + A$\bar{B}\overline{Bin}$ + ABBin

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

EX-NOR EX-OR

Therefore, D = Bin($\bar{A}\bar{B}$ + AB) + $\overline{Bin}$($\bar{A}B$ + A$\bar{B}$)

Let X = $\bar{A}B$ + A$\bar{B}$

Therefore D = Bin$\bar{X}$ + $\overline{Bin}$ X = Bin $\oplus$ X

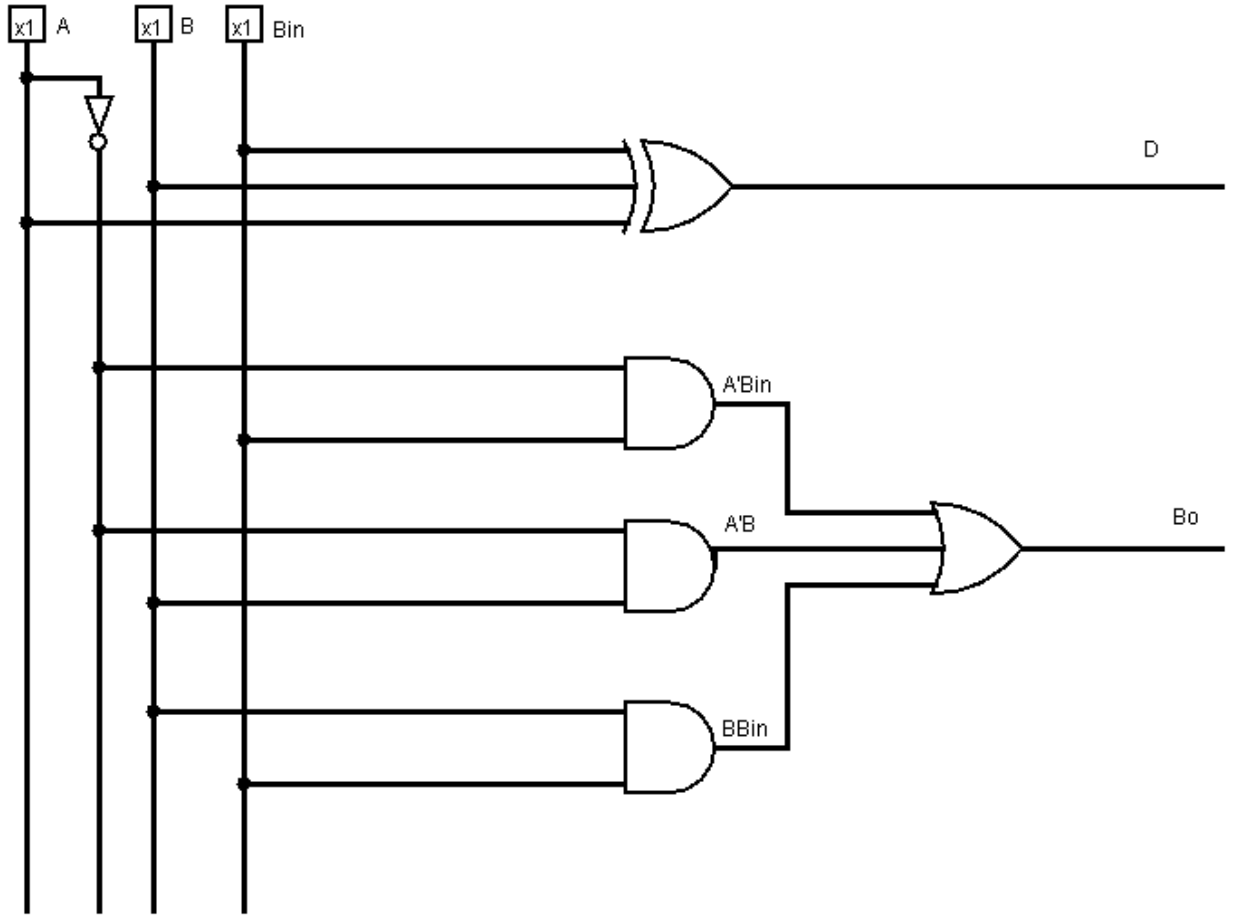
or D = Bin $\oplus$ ($\bar{A}B$ + A$\bar{B}$)

But , $\bar{A}B$ + A$\bar{B}$

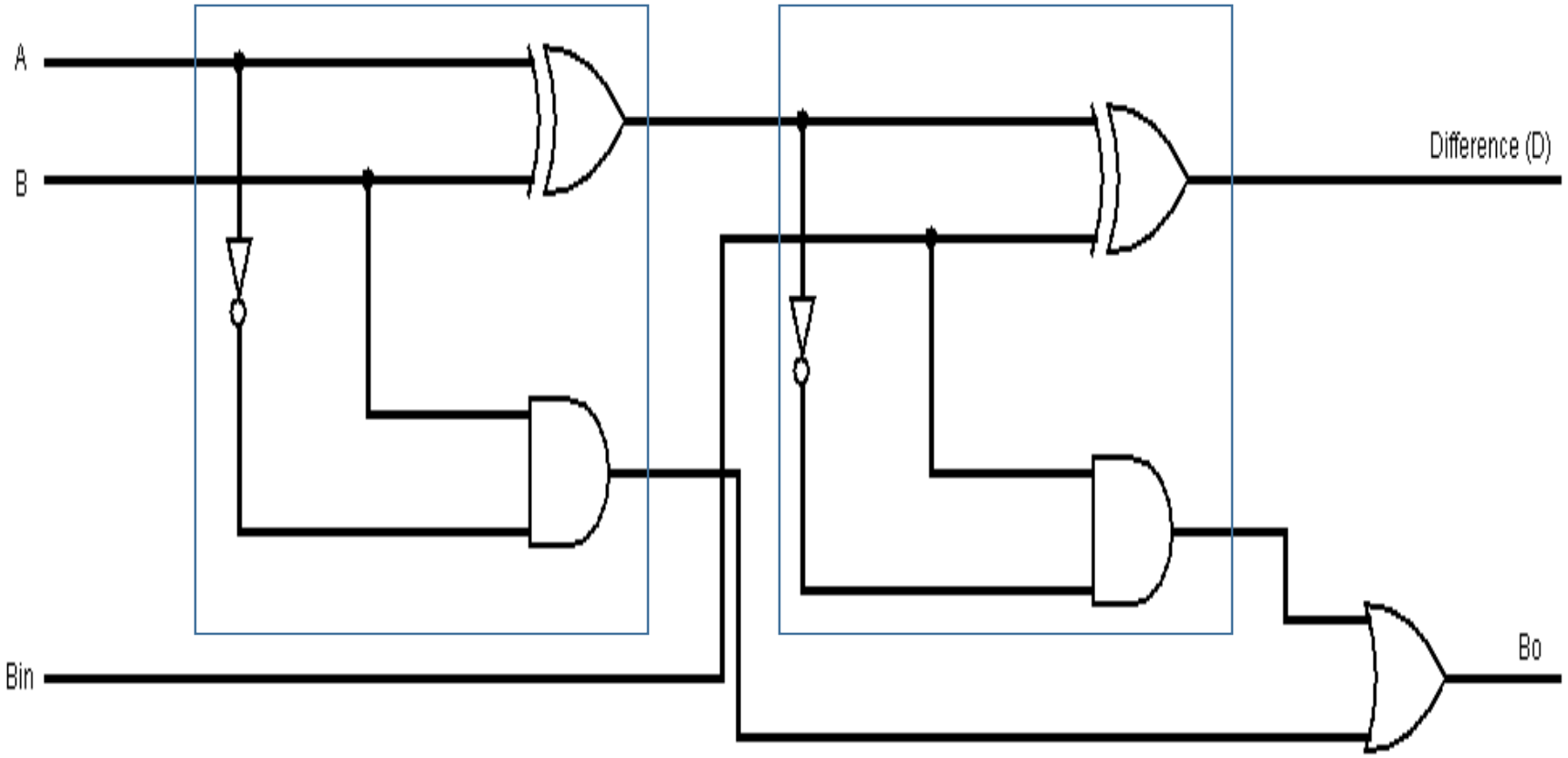
Therefore, **D= Bin $\oplus$ A $\oplus$ B**

- **Bo = $\bar{A}$Bin + $\bar{A}$B + BBin**

Logic Diagram of a full subtractor



Full subtractor using half subtractor



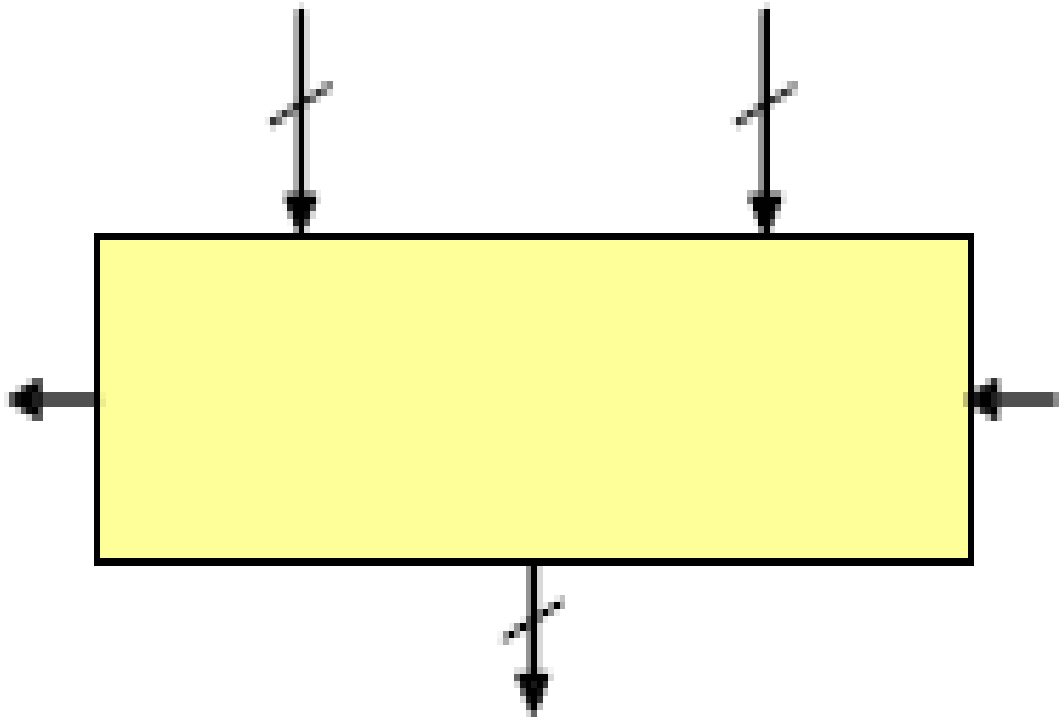
Practice Exercise

- Design a combinational circuit whose input is a four number and at the output we obtain the two's complement of the number

n-Bit Parallel Adder

- An n-bit adder is a circuit which adds two n-bits numbers, say, A and B.
- In addition, an n-bit adder will have another single-bit input which is added to the two numbers called the carry-in (Cin).
- The output of the n-bit adder is an n-bit sum (S) and a carry-out (Cout) bit. The block diagram of the n-bit adder is shown.

n-Bit Parallel Adder

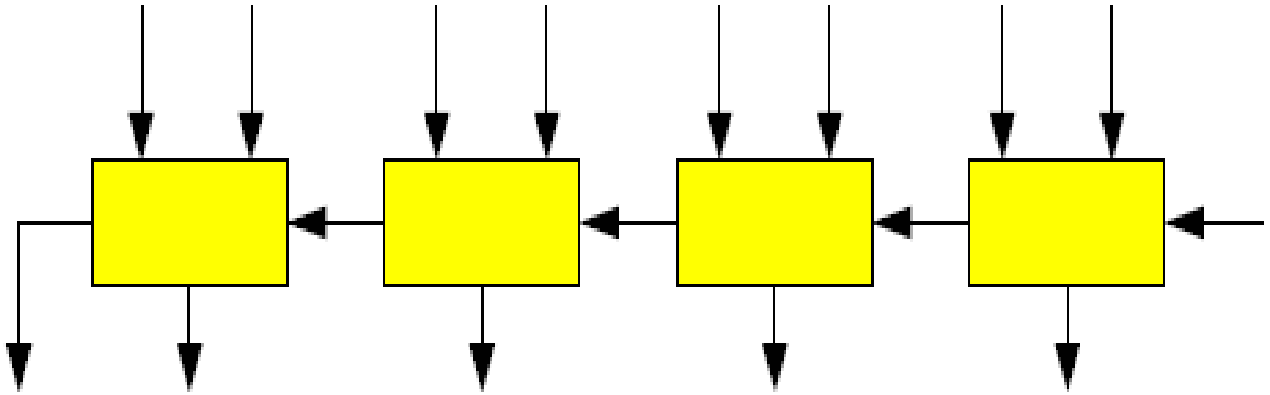


n-Bit Parallel Adder

- If all input bits of the two numbers (A & B) are applied simultaneously in parallel, the adder is termed a Parallel Adder.
- Consider the problem of designing a 4-bit binary parallel adder.
- The total number of inputs is 9, since the two numbers have 4-bits each in addition to the Cin bit. Using conventional techniques for design would require a truth table of 2^9=512 rows.
- This causes the conventional design procedure to be unacceptable in this case.

n-Bit Parallel Adder

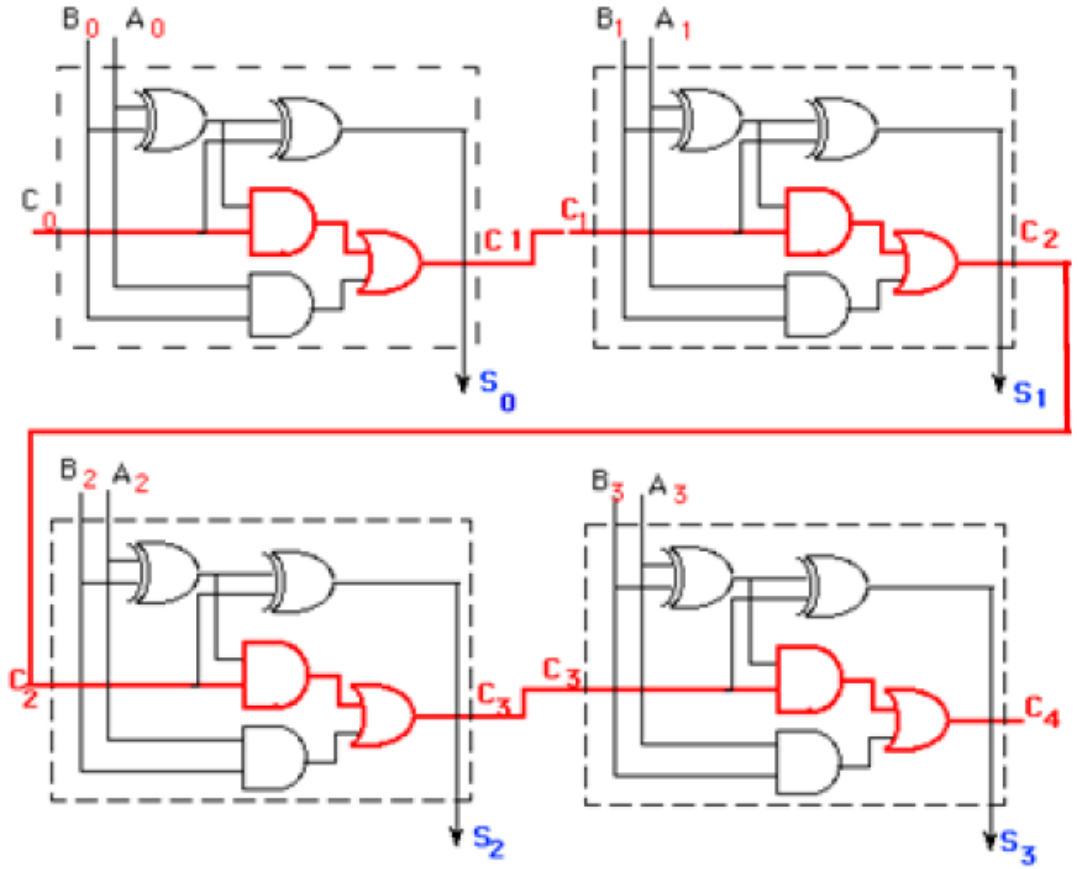
- Alternatively, the 4-bit binary parallel adder can be designed using 4 full adders connected in-cascade as shown in the figure.



n-Bit Parallel Adder

- The carry-out bit of one full adder stage is used as carry-in input to the next stage.
- In general, an n-bit binary parallel adder can be built out of n full adders connected in cascade.
- Since a carry of 1 may appear near the least significant bit of the adder and yet propagate through many full adders to the most significant bit, just as a wave ripples outward from a pebble dropped in a pond. That is why this parallel adder is also called as **ripple carry adder.**

- The disadvantage of the ripple-carry adder is that it can get very slow when one needs to add many bits.



n-Bit Parallel Adder

- The signal from the input carry to the output carry propagates through an AND gate and OR gate, which constitute two gate levels. If there are four full adders, the output carry would have 2 x 4 = 8 levels from C^0 to C^4.

- The total propagation time in this 4-bit adder would be the propagation time in one half adder (which is the first half adder) plus eight gate levels.

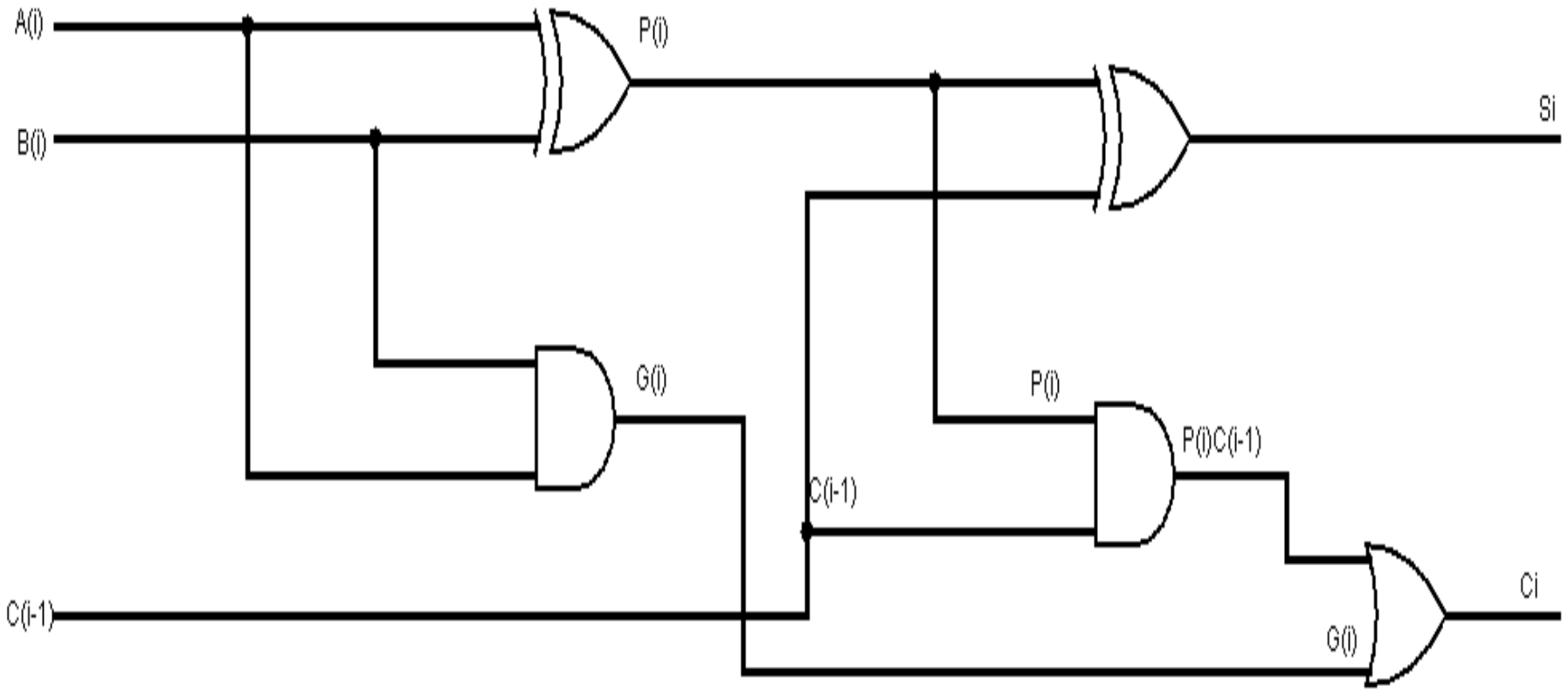
Carry Look Ahead (CLA)Addition

- This addition technique eliminates the interstage carry delay.

- CLA requires additional hardware but the speed is independent of the number of bits.

CLA Adder

- Considering the full adder below:



CLA Expressions

- $P(i) = A(i) \oplus B(i)$

And $G(i) = A(i)B(i)$

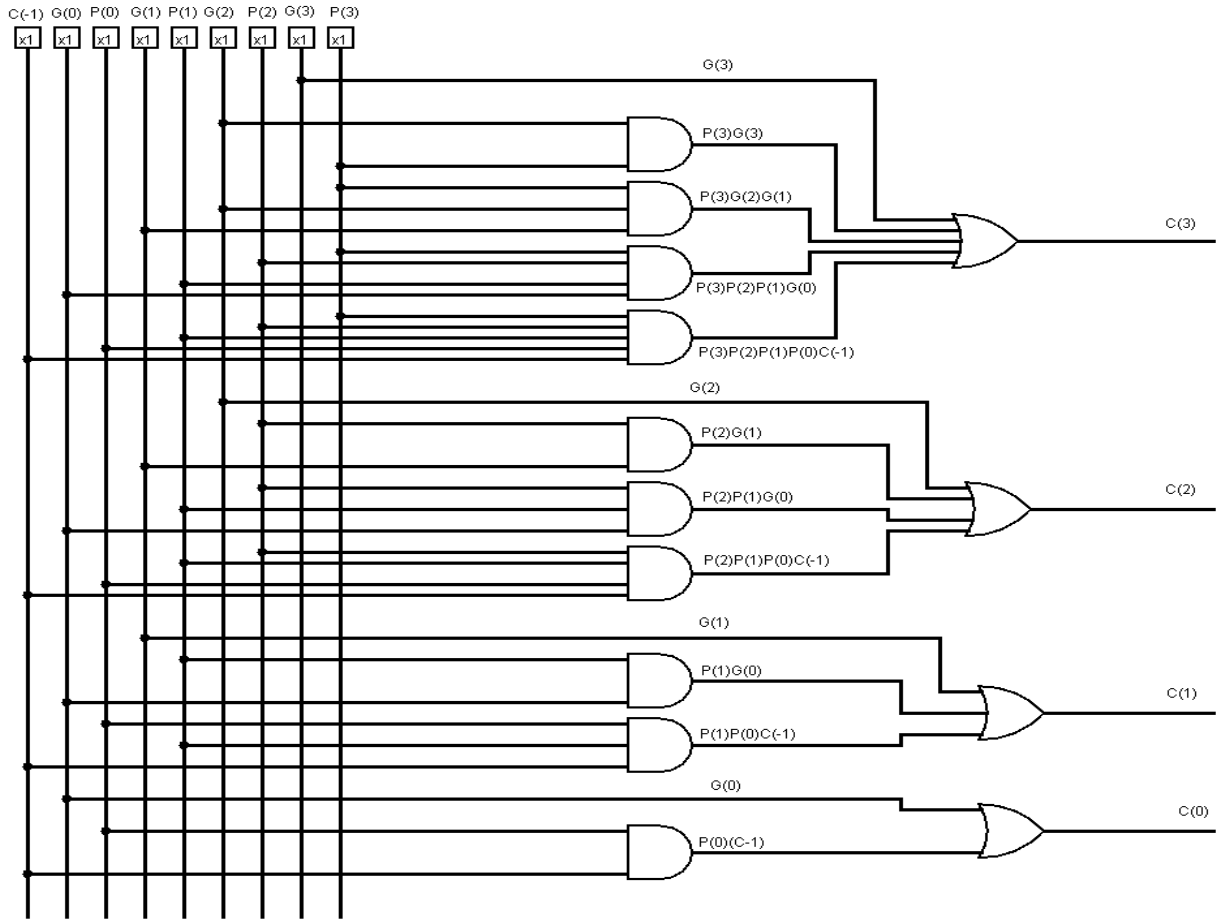
Also, $S(i) = P(i) \oplus C(i) = A(i) \oplus B(i) \oplus C(i-1)$

- And $C(i) = G(i) + P(i)C(i-1)$

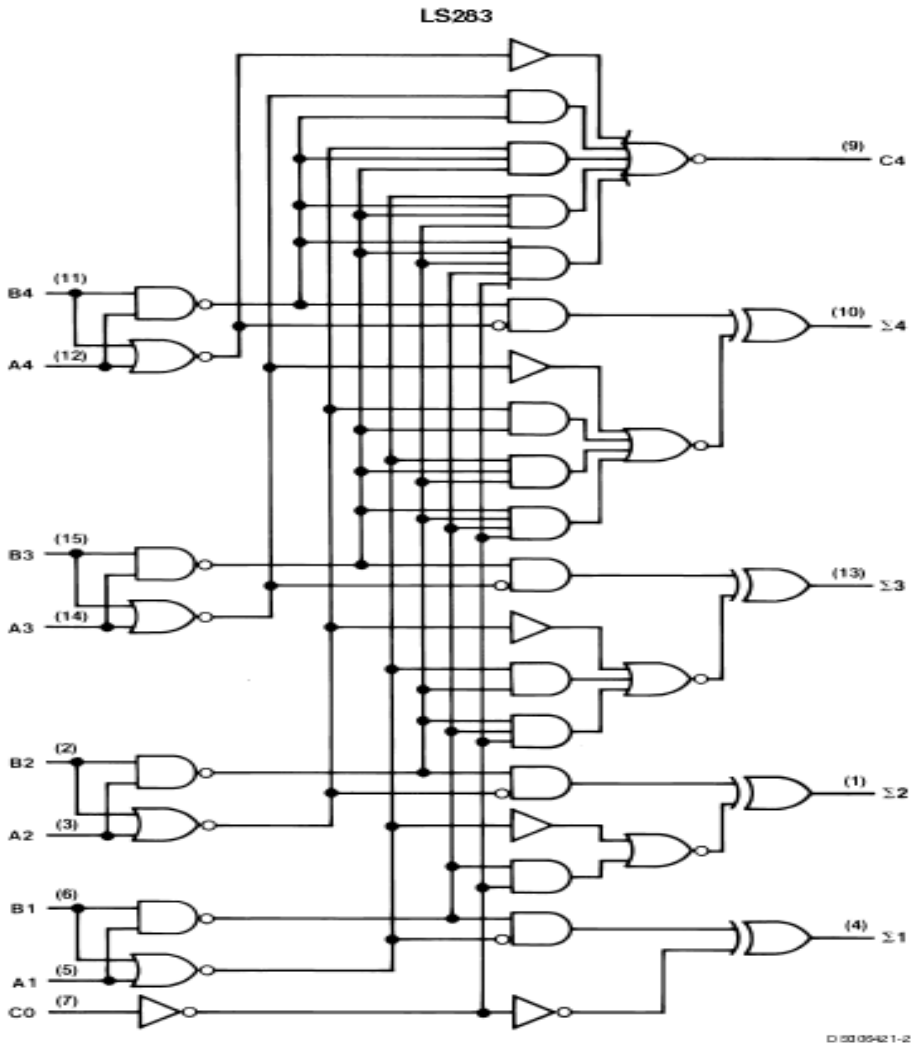
Expression for carry output

Stage	Expression for carry output
0	$C_0 = G_0 + P_0\ C_{-1}$
1	$C_1 = G_1 + P_1\ C_0 = G_1 + P_1\ (G_0 + P_0\ C_{-1})$ Therefore $G_1 + P_1G_0 + P_0P_1\ C_{-1}$
2	$C_2 = G_2 + P_2C_1 = G_2 + P_2(G_1 + P_1G_0 + P_0P_1\ C_{-1})$ Therefore, $G_2 + P_2G_1 + P_1P_2G_0 + P_0P_1P_2C_{-1}$
3	$C_3 = G_3 + P_3C_2 = G3 + P3(G_2 + P_2G_1 + P_1P_2G_0 + P_0P_1P_2C_{-1})$ Therefore, $G3 + P_3G_2 + P_2P_3G_1 + P_1P_2P_3G_0 + P_0P_1P_2P_3C_{-1}$

Logic Diagram for Carry Look Ahead

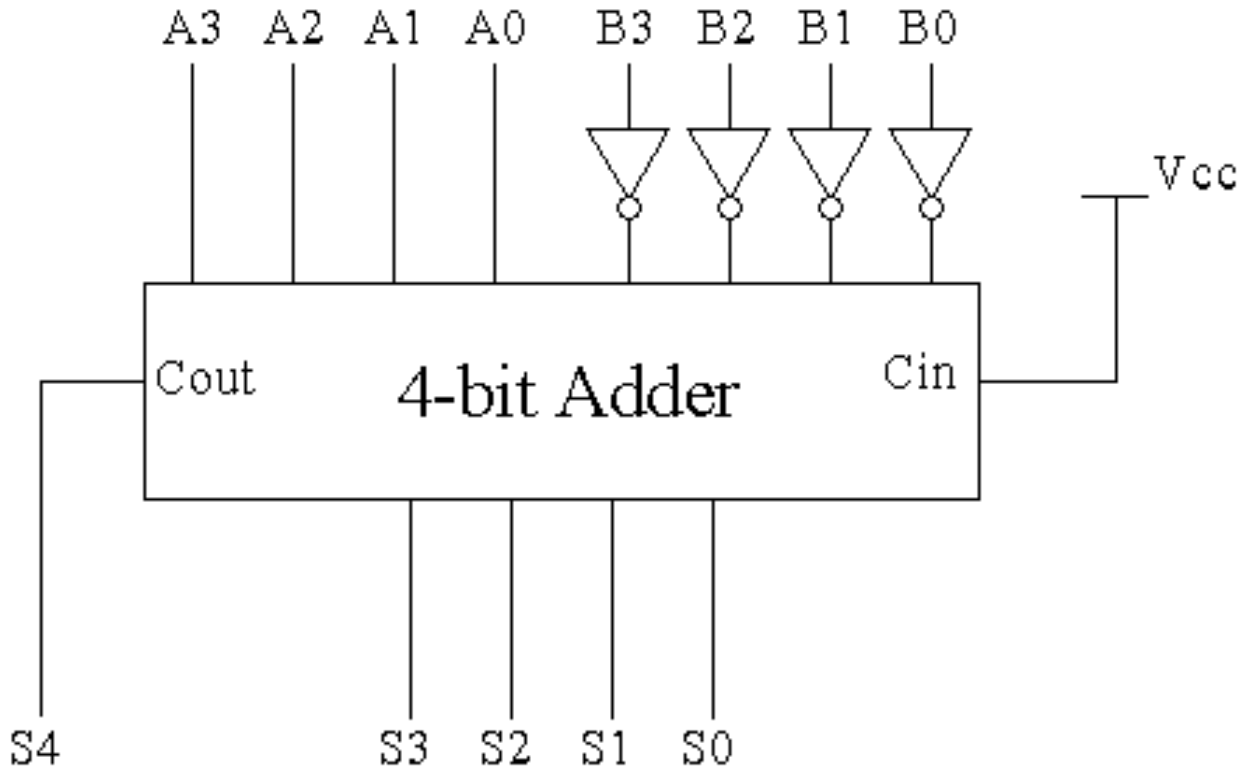


4 bit CLA Adder

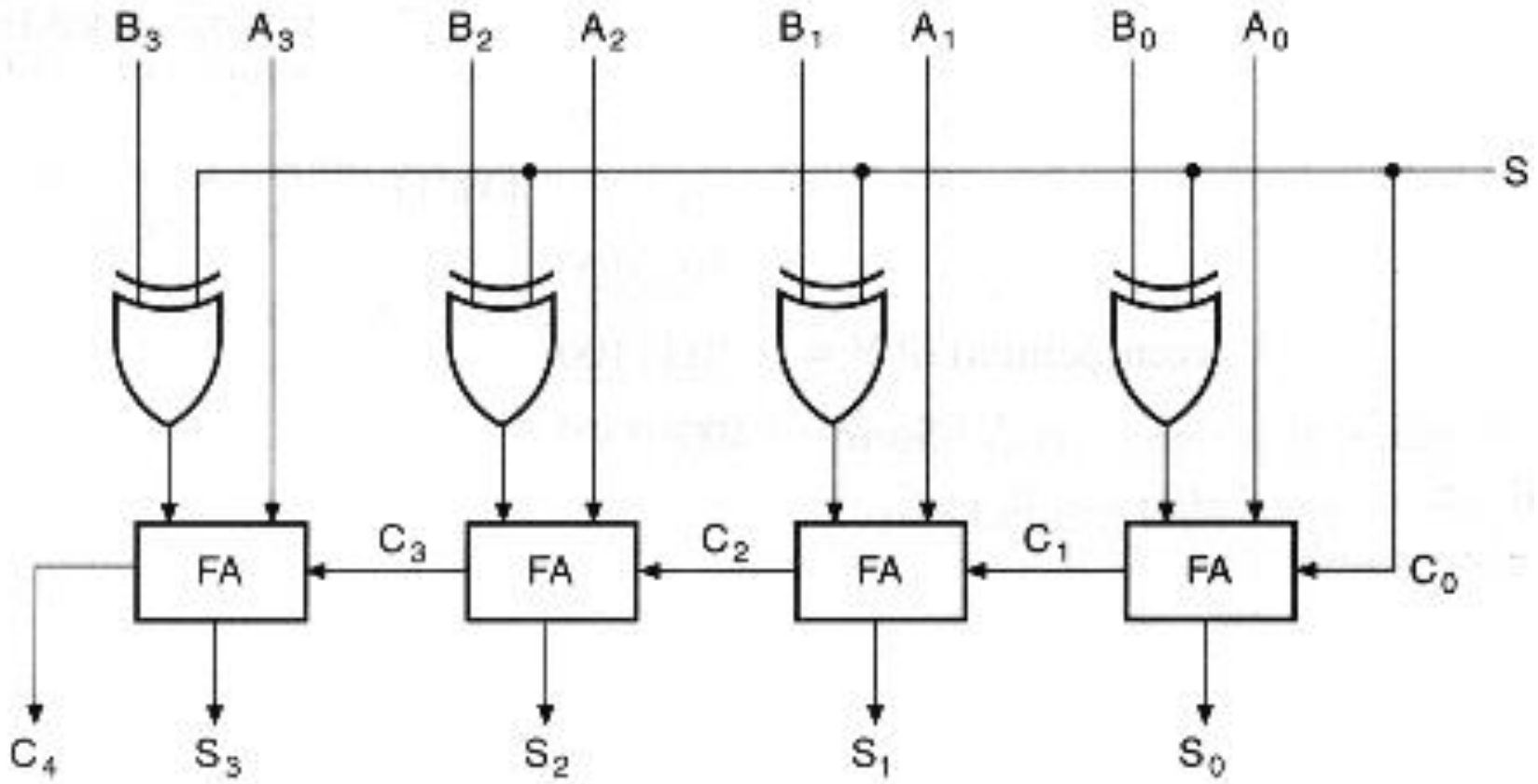


N-Bit Parallel Subtractor

- Implements two's complement



Subtractor/ Adder



Next Lecture

- Multiplexers
- Demultiplexers
- Decoders
- Encoders