Combinational Circuits

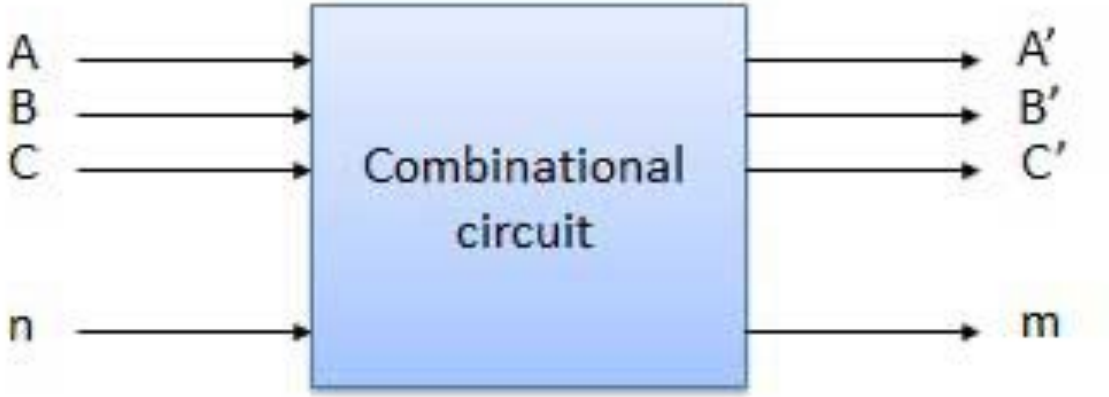
S.M. Shovan

Contents

- What is Combinational circuits?
- Binary logic gates
- Basic logic functions
- Universal gates and it’s applications
- Design of
    1. Adder
        a. Half adder
        b. Full Adder
        c. Parallel Adder
    2. Subtractor
        1. Half subtractor
        2. Full subtractor

What is combinational circuits?

- Combination of different gates (AND, OR, NOT)
- Output depends on present/current state
- No memory is used
- Can have n inputs and M outputs



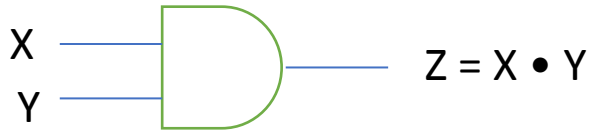
Binary variables

- Identifiers can be letters of Alphabet
  - A,B, x,y,z , X1 etc.

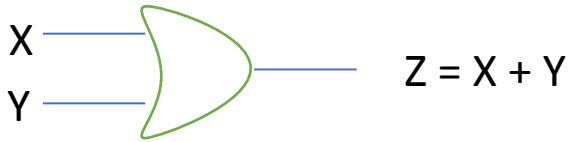
- Can have one of two binary values
  - True/False
  - On/Off
  - Yes/No
  - 1/0

Logical operations

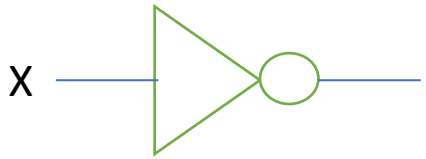
- Operations of binary variables
- Three basic logical operations
  - AND (denoted by a dot (•))



  - OR (denoted by a plus (+))



  - NOT (denoted by a overbar (¯), a single quote mark (‘) or tilde (~)



Truth tables

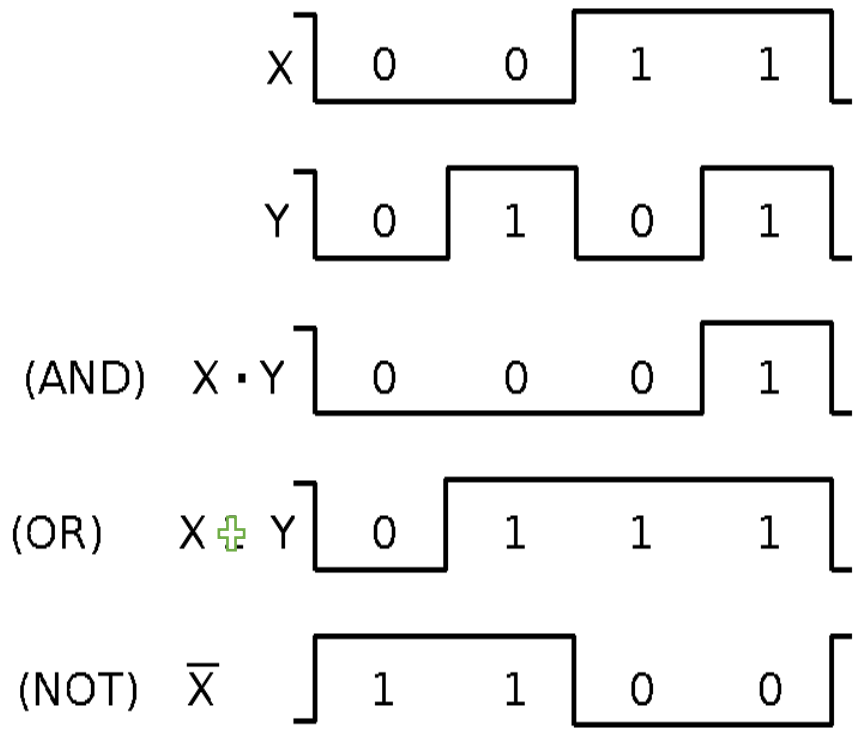
Truth table – a tabular listing of the values of a function for all possible combinations of values on its arguments

AND		
X	Y	Z = X·Y
0	0	0
0	1	0
1	0	0
1	1	1

OR		
X	**Y**	**Z = X+Y**
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

NOT	
X	$Z = \overline{X}$
0	1
1	0

Timing diagram



Logic diagram and Expressions

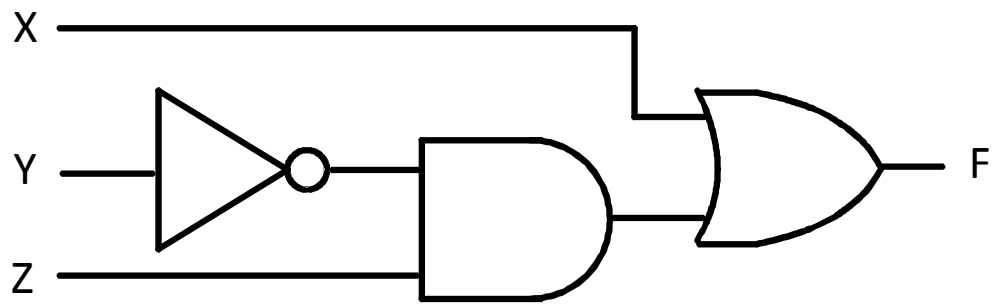
Truth Table

X Y Z	$F = X + \overline{Y} \cdot Z$
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

Equation

$$F = X + \overline{Y}\,Z$$

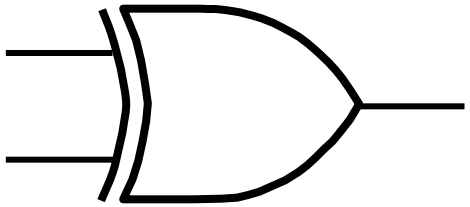
Logic Diagram



Additional gates

XOR Gates $X \oplus Y = X\overline{Y} + \overline{X}Y$

Symbol



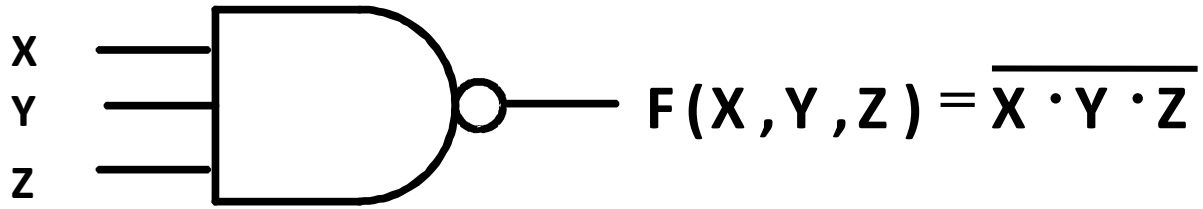
Truth table

X	Y	X ⊕ Y
0	0	0
0	1	1
1	0	1
1	1	0

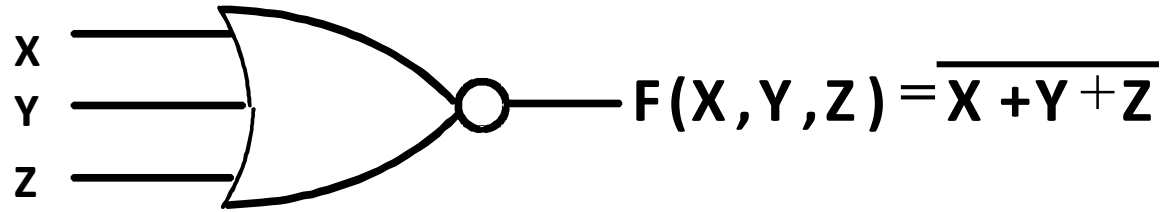
Additional gates (Universal gates)

Universal gate - a gate type that can implement any Boolean function.

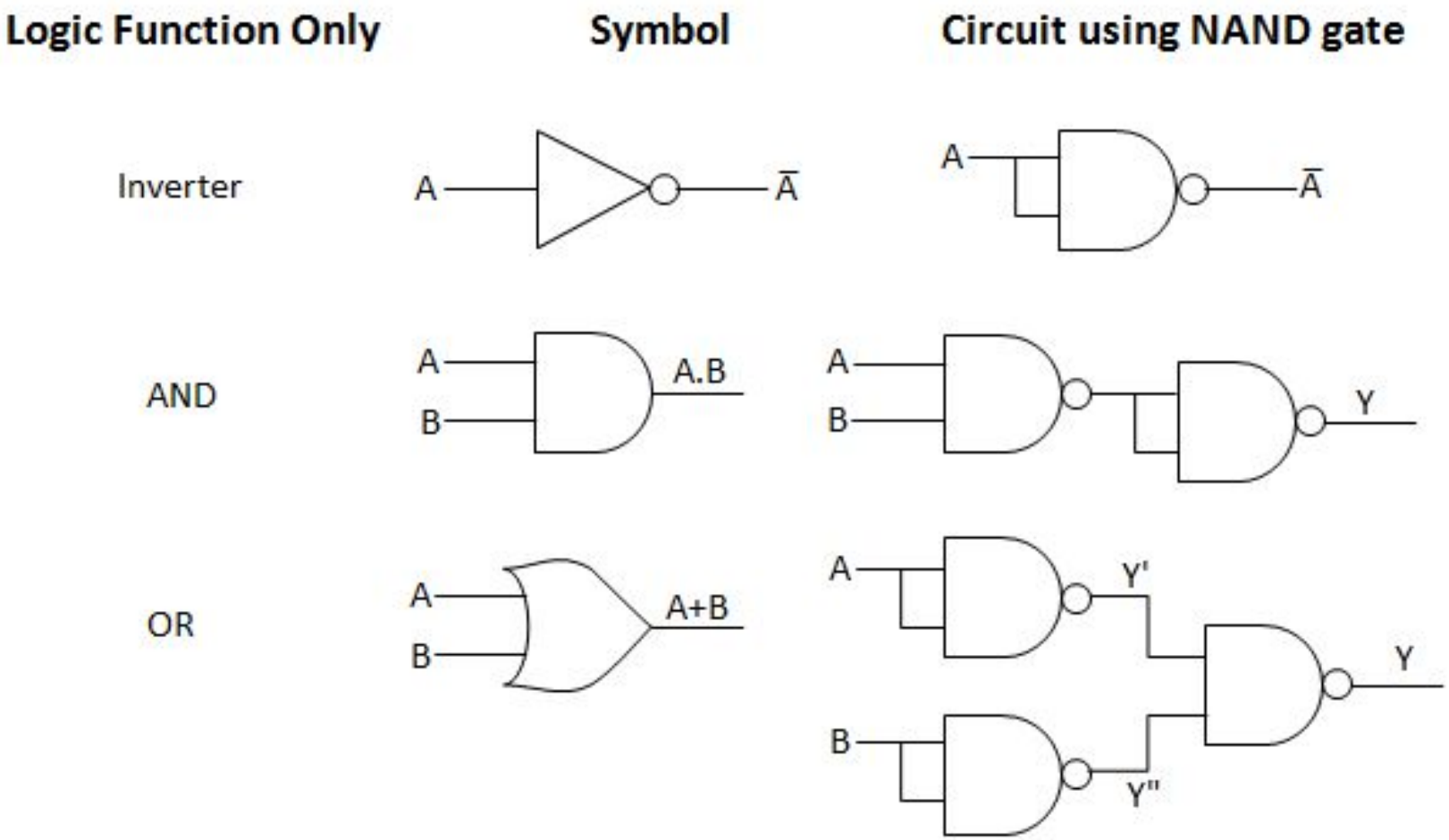
- NAND gates



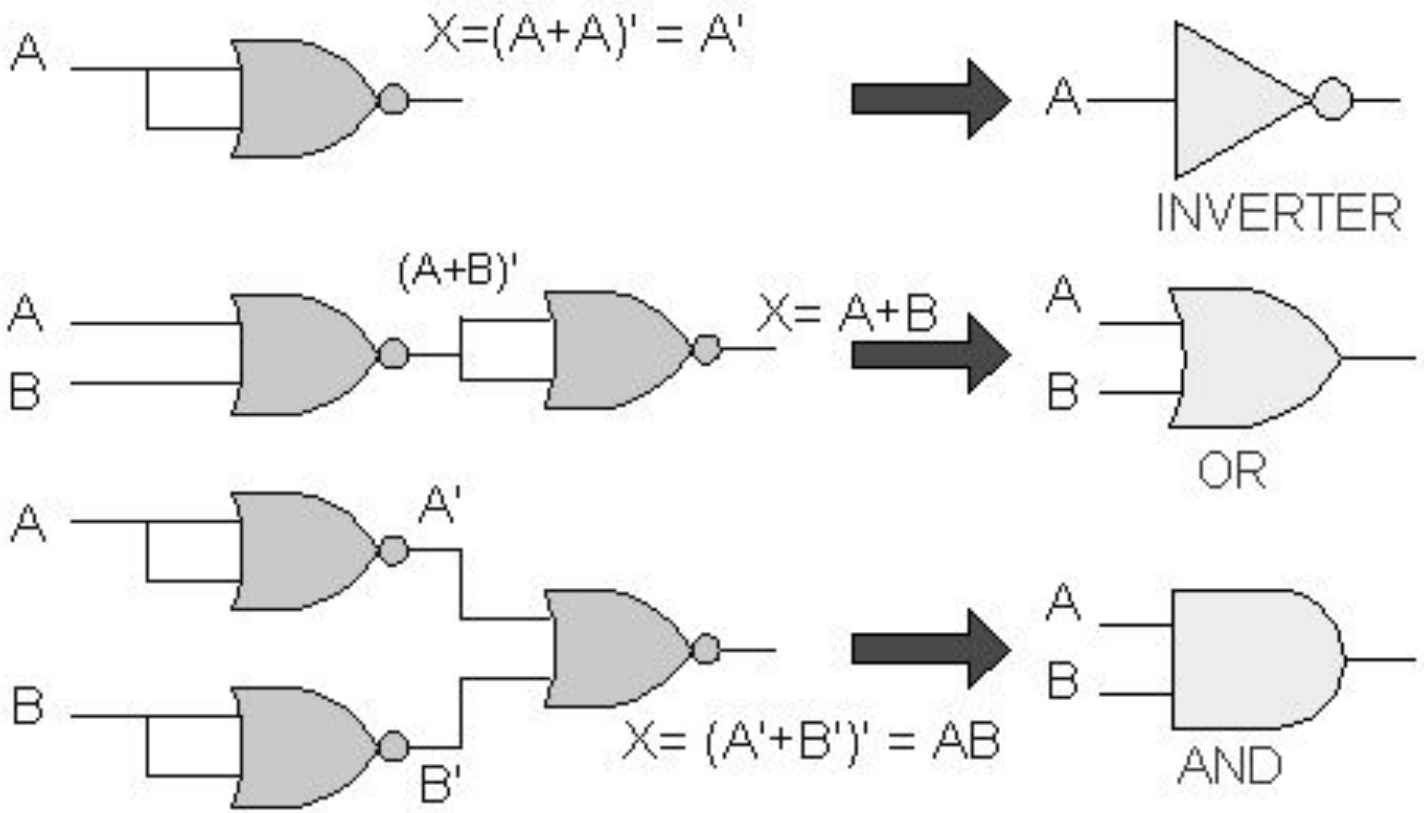
- NOR gates



Basic logic gates implementation using NAND



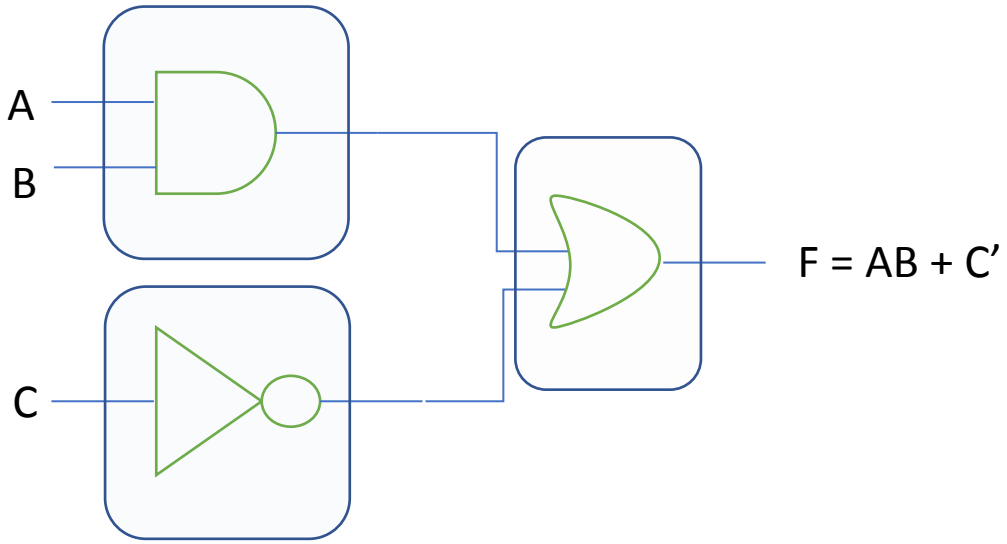
Basic logic gates implementation using NOR



NAND gate implementation of Boolean function

- F = AB + C’

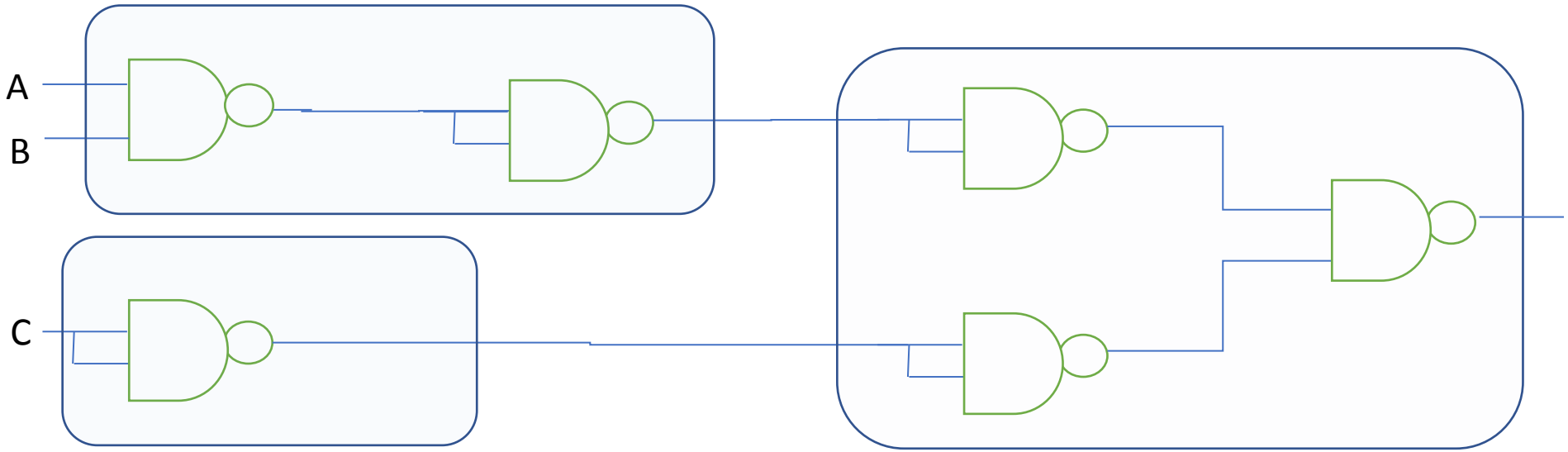
Using AND, OR, NOT gates



NAND gate implementation of Boolean function

- F = AB + C’

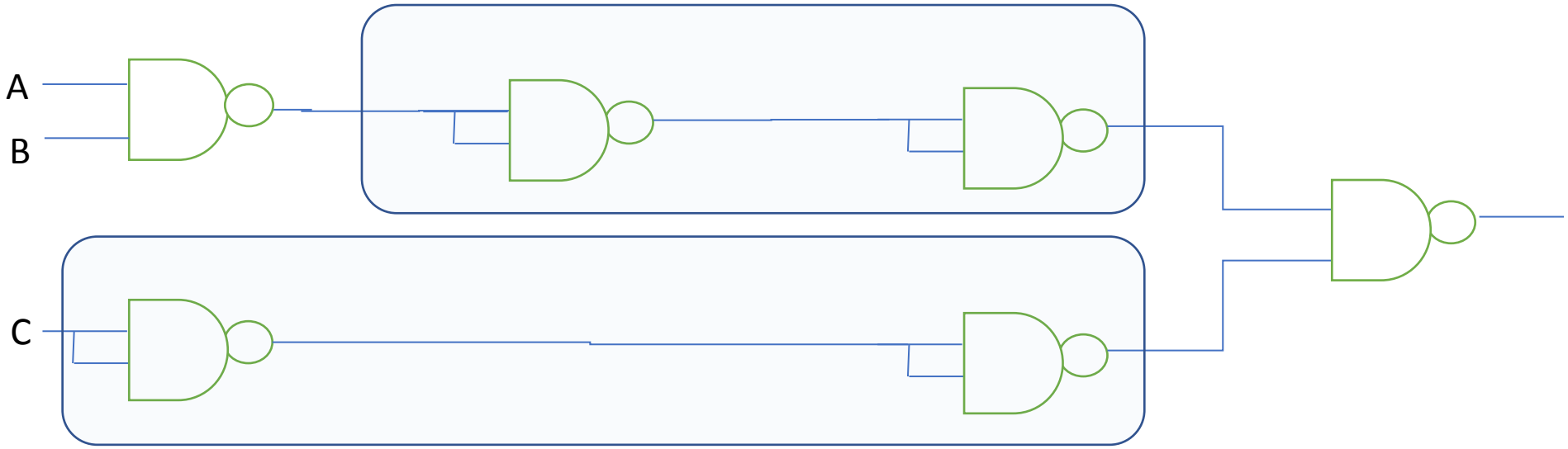
Using NAND gates



NAND gate implementation of Boolean function

- F = AB + C’

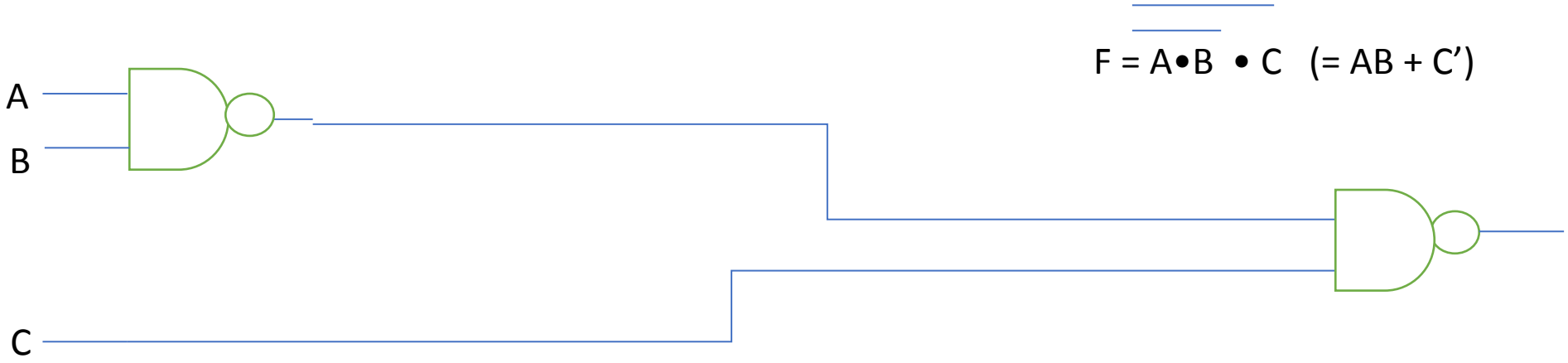
Using NAND gates



NAND gate implementation of Boolean function

- F = AB + C’

Using NAND gates



NOR gate implementation of Boolean function

- F = AB + C’

Try Yourself

Application of combinational circuit (Adder)

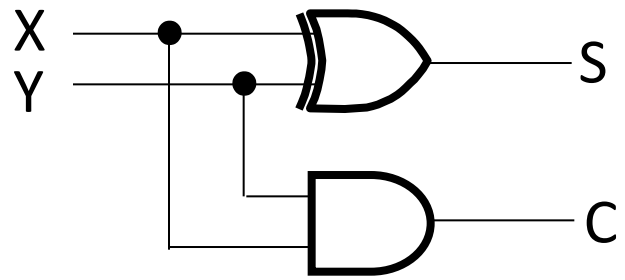
- Half Adder

X	0	0	1	1
+ Y	+ 0	+ 1	+ 0	+ 1
C S	0 0	0 1	0 1	1 0

X	Y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$S = X \cdot \overline{Y} + \overline{X} \cdot Y = X \oplus Y$$

$$C = X \cdot Y$$



Application of combinational circuit (Adder)

- Full Adder

X	Y	Z	C	S
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

$$S = X\,\overline{Y}\,\overline{Z} + \overline{X}\,Y\,\overline{Z} + \overline{X}\,\overline{Y}\,Z + X\,Y\,Z$$
$$C = X\,Y + X\,Z + Y\,Z$$

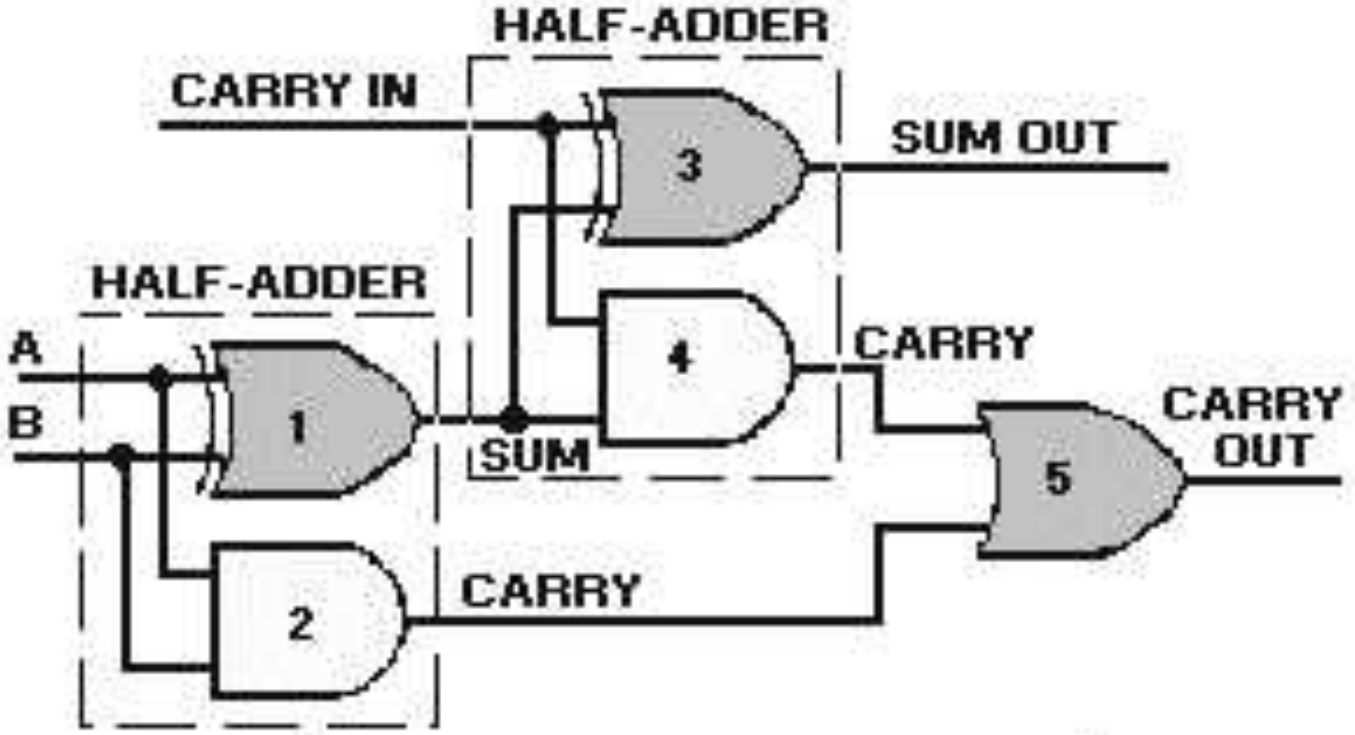
$$S = X \oplus Y \oplus Z$$

$$C = X\,Y + (X \oplus Y)\,Z$$

Full-Adder Implement

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

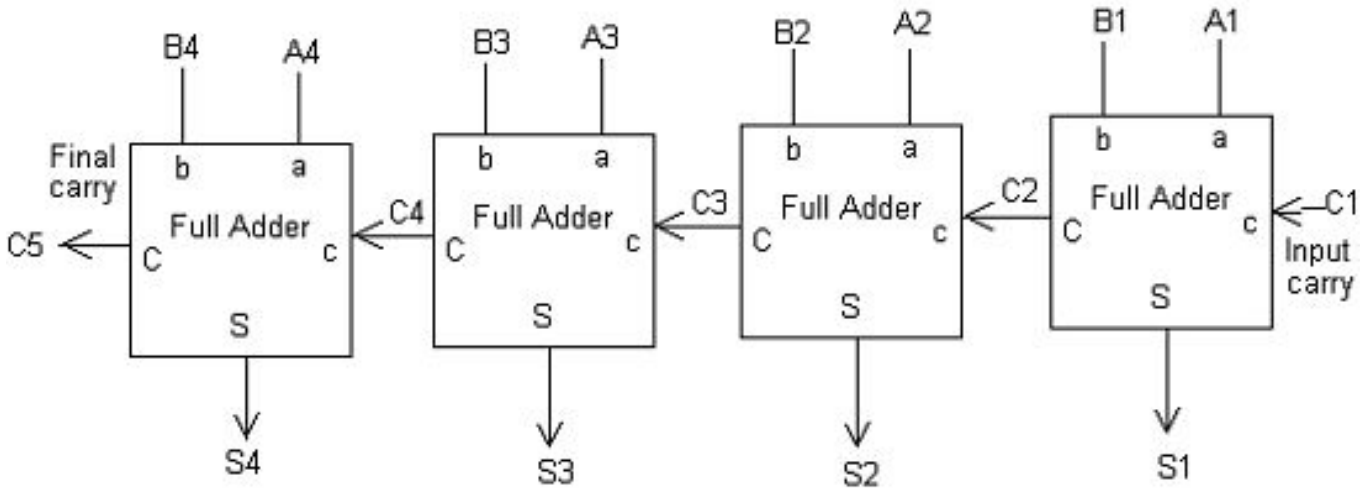
$$C_{out} = AB + (A \oplus B)C_{in}$$



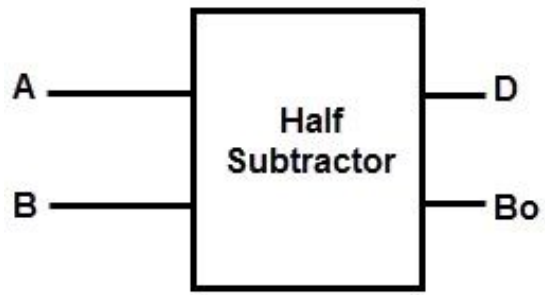
4 bit Parallel adder

A4 A3 A2 A1
B4 B3 B2 B1

C5 S4 S3 S2 S1



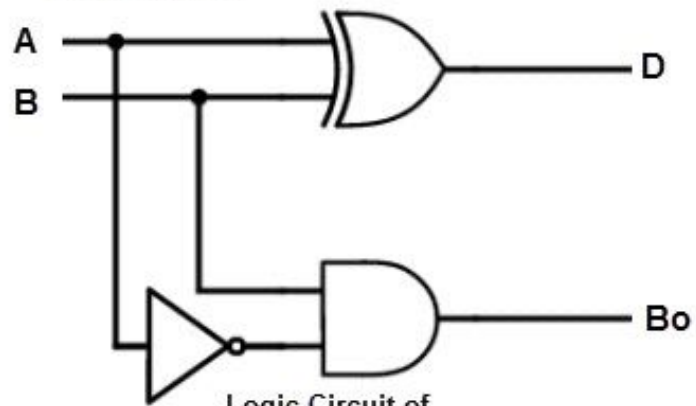
Half-Subtractor



Block Diagram

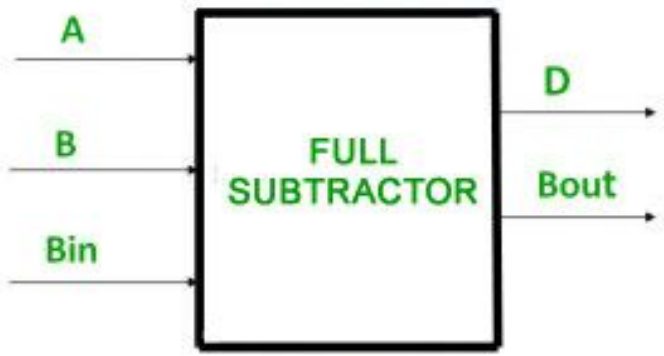
A	B	D	B_o
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Truth Table



Logic Circuit of
Half Subtractor

Full subtractor



INPUT			OUTPUT	
A	B	Bin	D	Bout
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

Full subtractor