EEE130 Digital Electronics I
Lecture #5_1
- Combinational Logic Analysis -

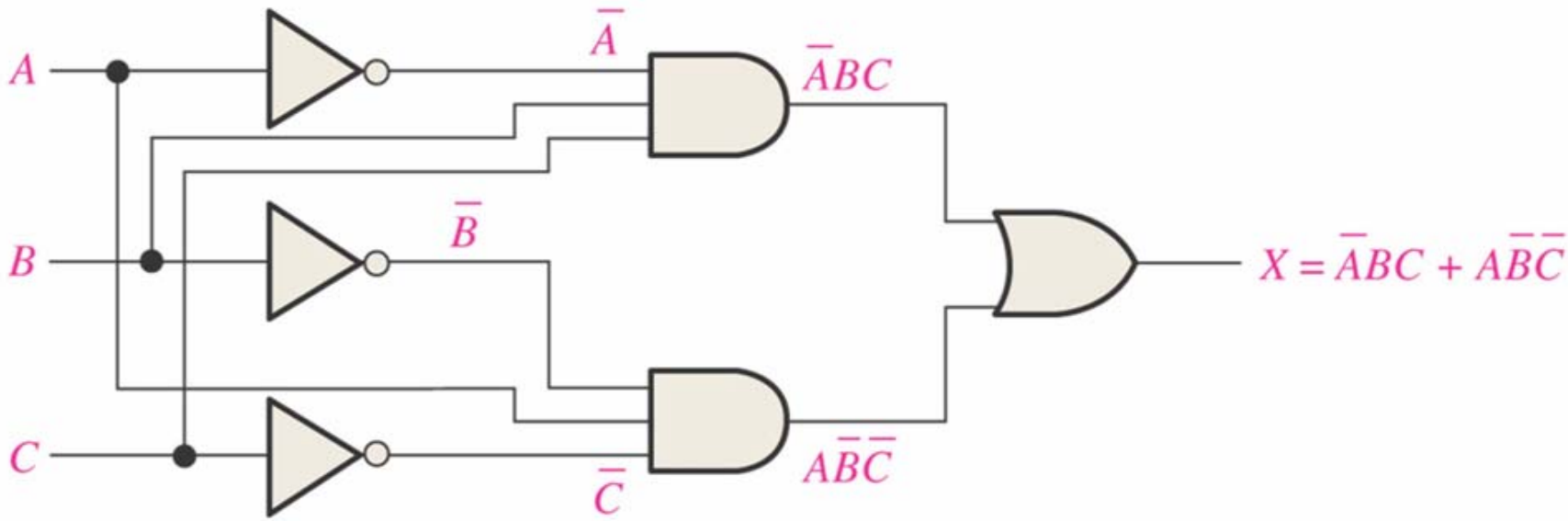
By Dr. Shahrel A. Suandi

5-2 Implementing Combinational Logic

- From a truth table to a logic circuit
  - The Boolean SOP expression can be obtained from the truth table by ORing the product terms for which X=1(output = 1)
  - Example: $X = \bar{A}BC + A\bar{B}\bar{C}$
- More intuitive explanation is as follows

Inputs			Output
A	B	C	X
0	0	0	0
0	1	0	0
0	0	1	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

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



Example 5-3

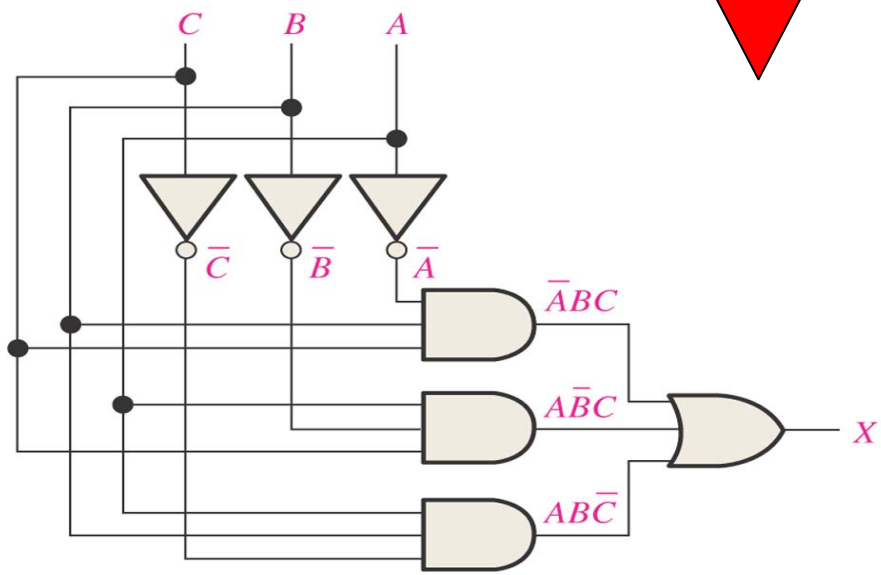
Inputs			Output
A	B	C	X
0	0	0	0
0	1	0	0
0	0	1	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

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

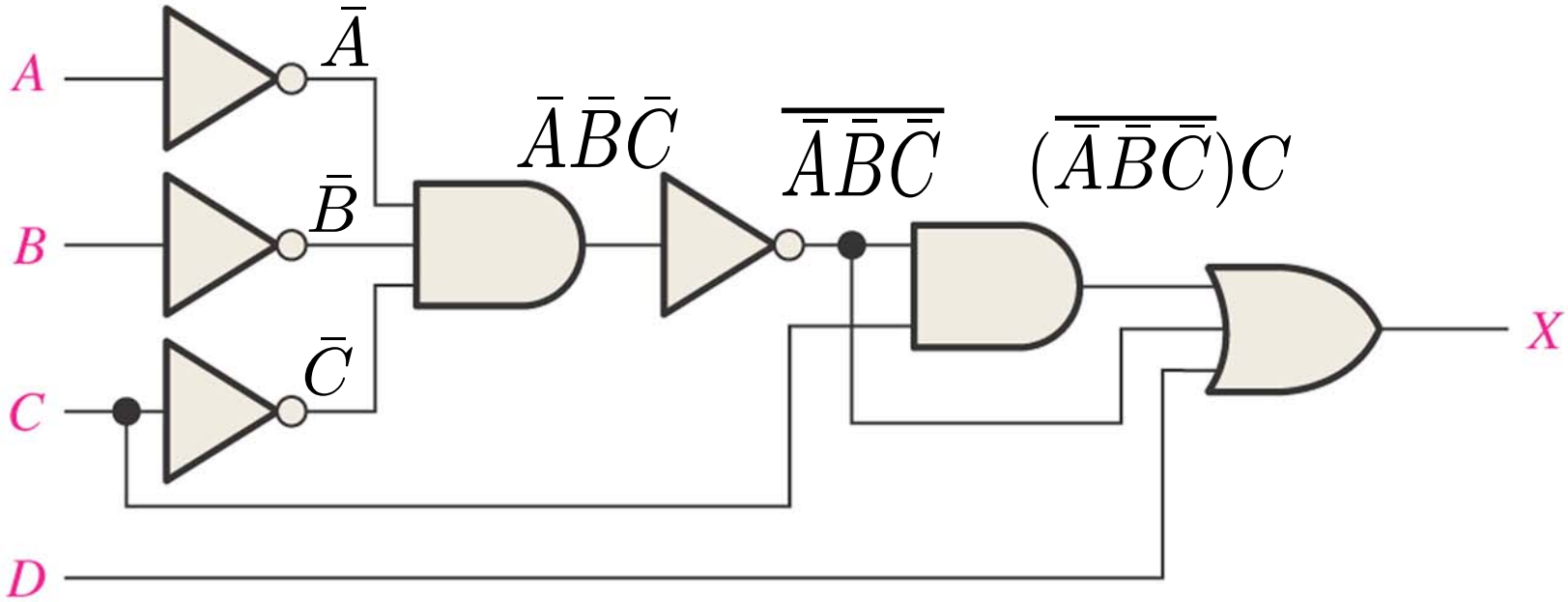
$$X = \bar{A}BC$$

$$X = A\bar{B}C$$

$$X = AB\bar{C}$$

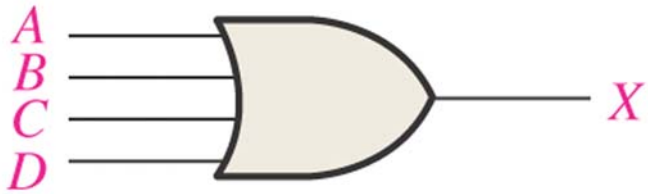


Example 5-5



$$X = (\overline{\bar{A}\bar{B}\bar{C}})C + \overline{\bar{A}\bar{B}\bar{C}} + D$$

$$X = A + B + C + D$$

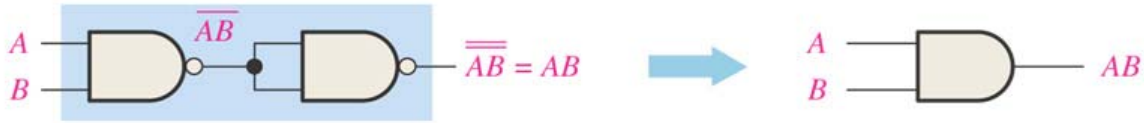


5-3 The Universal Property of NAND and NOR Gates

- In this section, we will learn about using NAND and NOR as an inverter (NOT), AND, OR, NOR (NAND)
- Meaning of “universal gate” → can be used to produce other gates
  - NAND – can be used to produce NOT, AND, OR and NOR

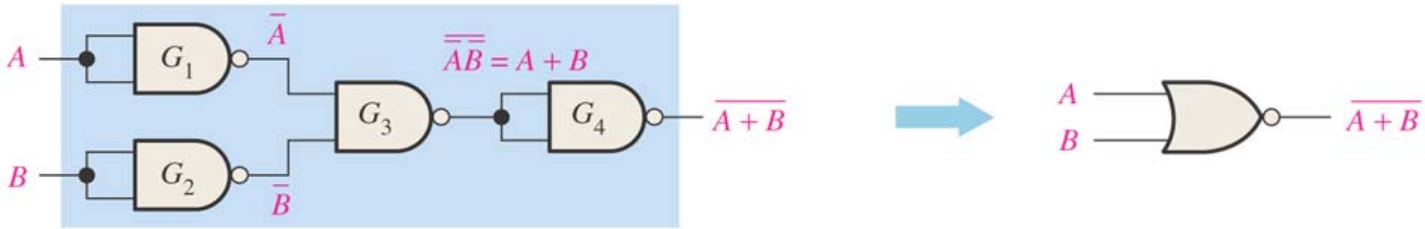
NAND as universal gate

(a) One NAND gate used as an inverter



(b) Two NAND gates used as an AND gate

(c) Three NAND gates used as an OR gate

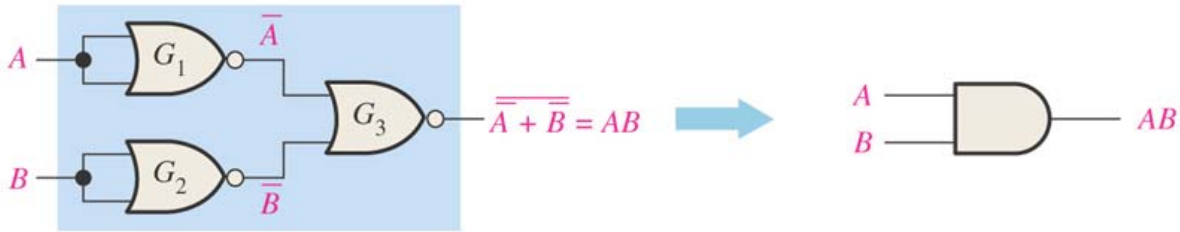


(d) Four NAND gates used as a NOR gate

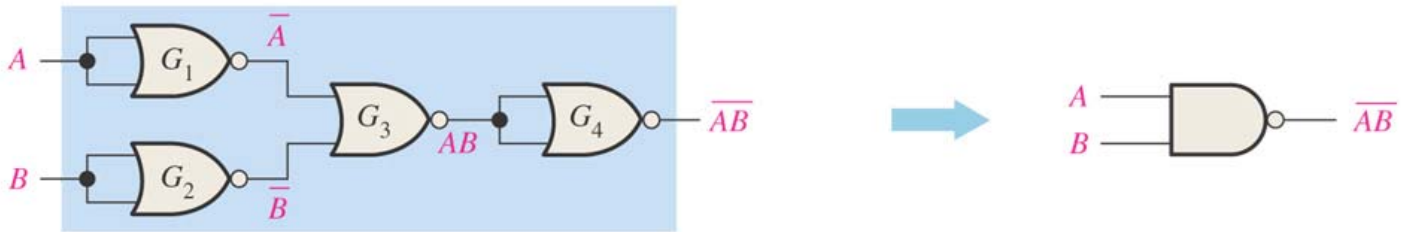
NOR as universal gate

(a) One NOR gate used as an inverter

(b) Two NOR gates used as an OR gate



(c) Three NOR gates used as an AND gate



(d) Four NOR gates used as a NAND gate

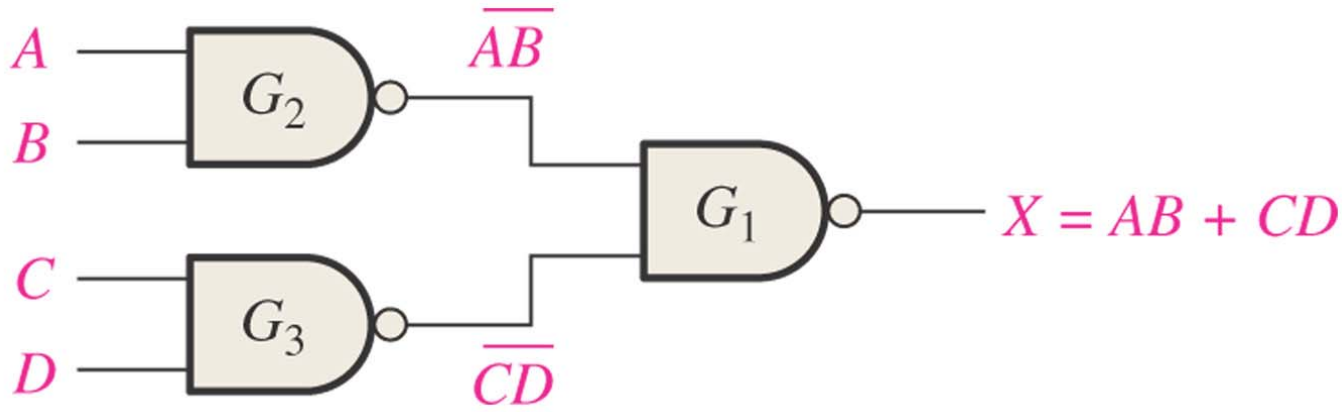
5-4 Combinational Logic Using NAND and NOR Gates

- Using whatever we have learnt in Chapter 3 (those related with NOR=Neg-AND and NAND=Neg-OR), we will try to make things easier… ie., to read logic circuits

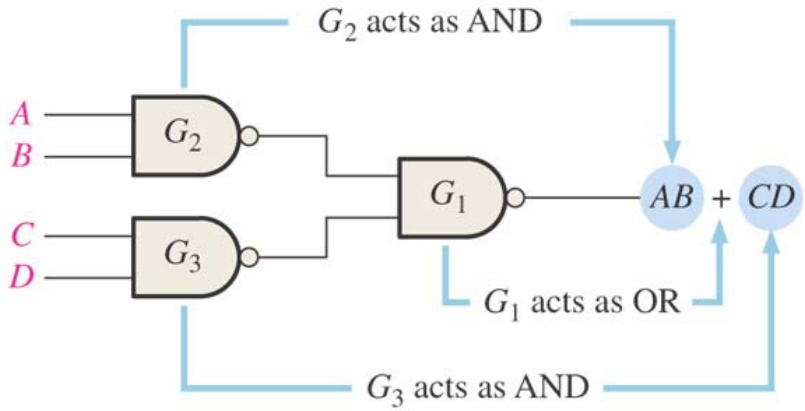
Dual Symbols

NAND Logic

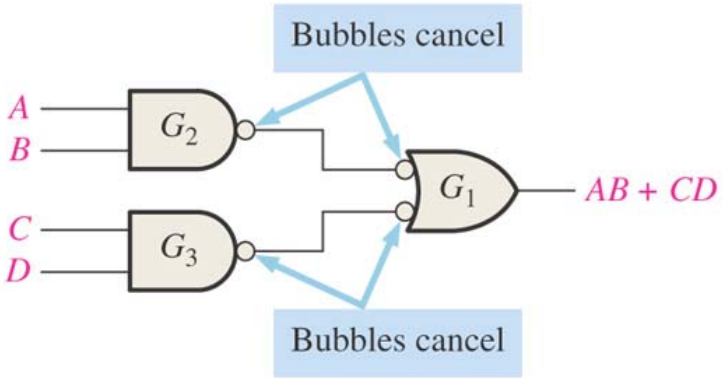
- Let's consider this logic circuit and its' Boolean expression



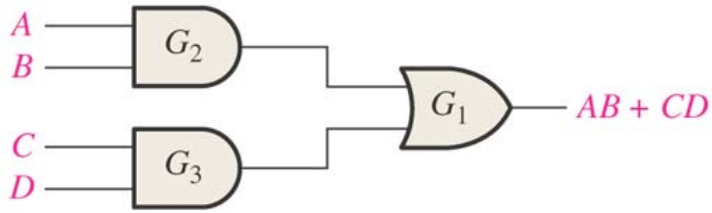
Development of AND-OR equivalent circuit



(a) Original NAND logic diagram showing effective gate operation relative to the output expression



(b) Equivalent NAND/Negative-OR logic diagram

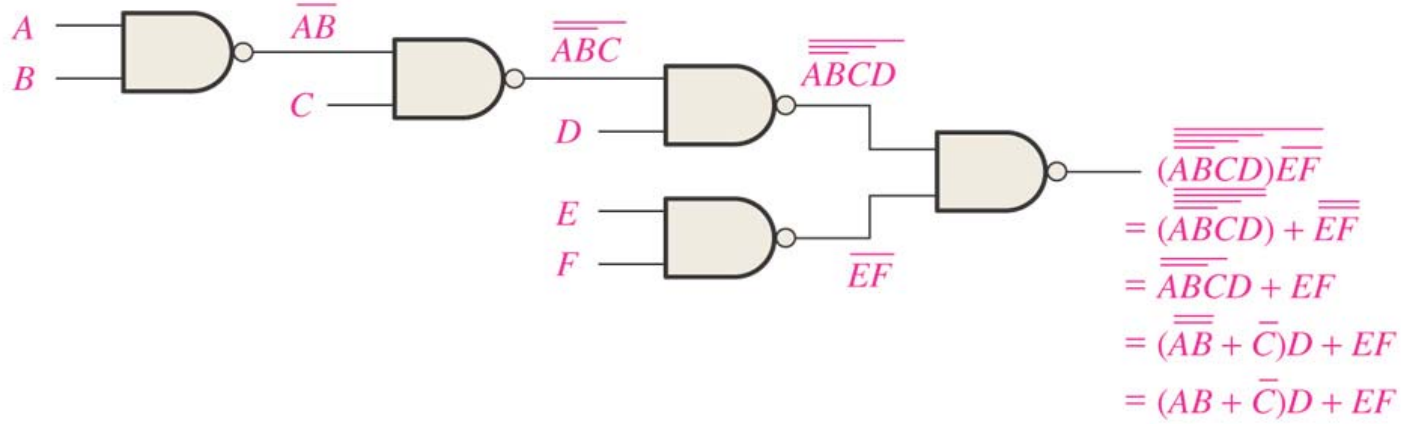


(c) AND-OR equivalent

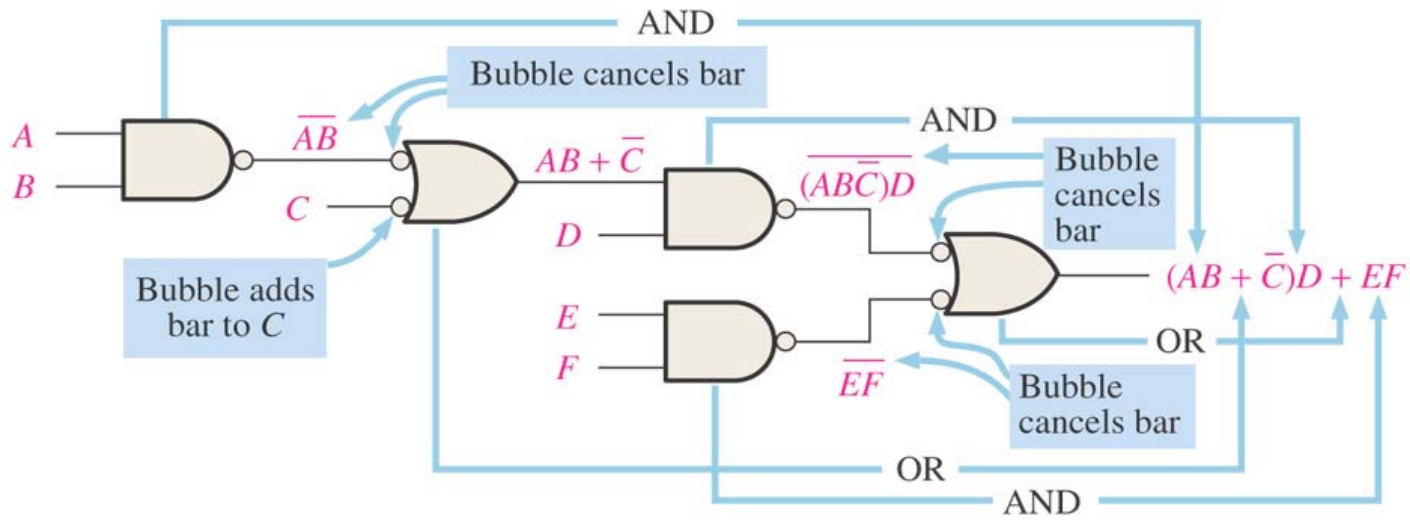
NAND logic diagrams using dual symbols

- Dual symbols – NAND symbols and Neg-OR symbols
- Correct method of drawing a NAND logic diagram:
  - Always use the symbols in such a way that every connection between a gate output and a gate input is either *bubble-to-bubble* or *nonbubble-to-nonbubble*
  - A bubble output *should not be connected* to a nonbubble input or vice versa

Illustration of using appropriate dual symbols in a NAND logic diagram



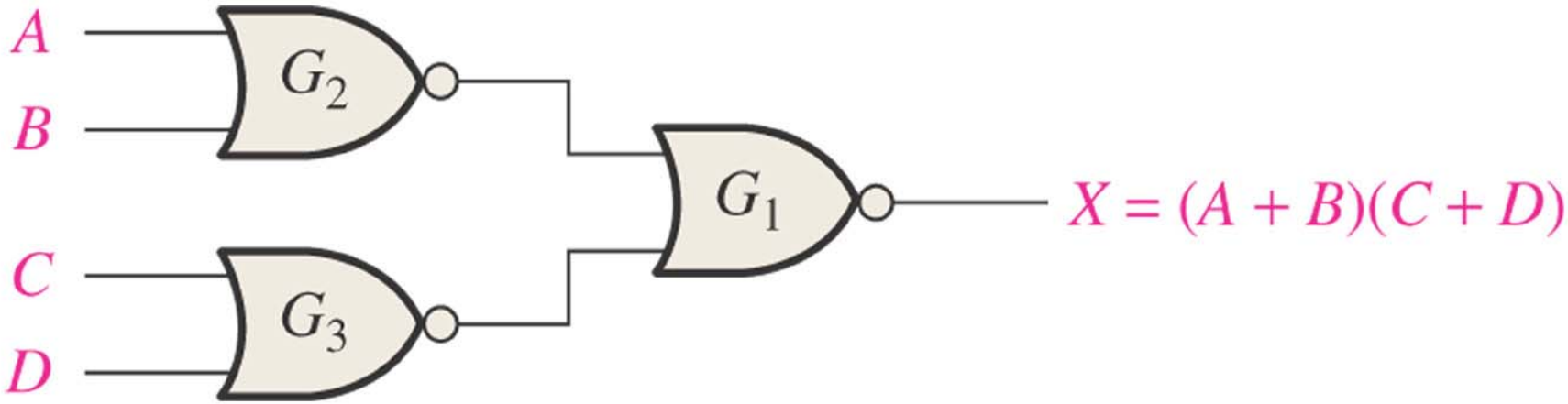
(a) Several Boolean steps are required to arrive at final output expression.



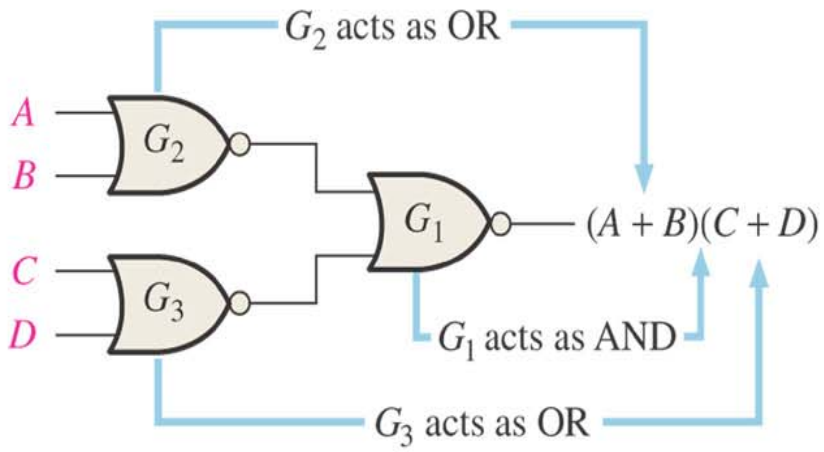
(b) Output expression can be obtained directly from the function of each gate symbol in the diagram.

NOR Logic

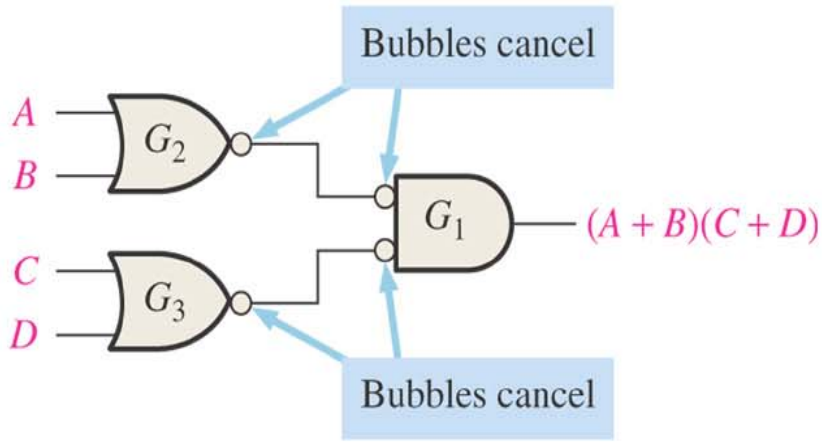
- Let's consider this logic circuit and its' Boolean expression



Development of OR-AND equivalent circuit



(a)

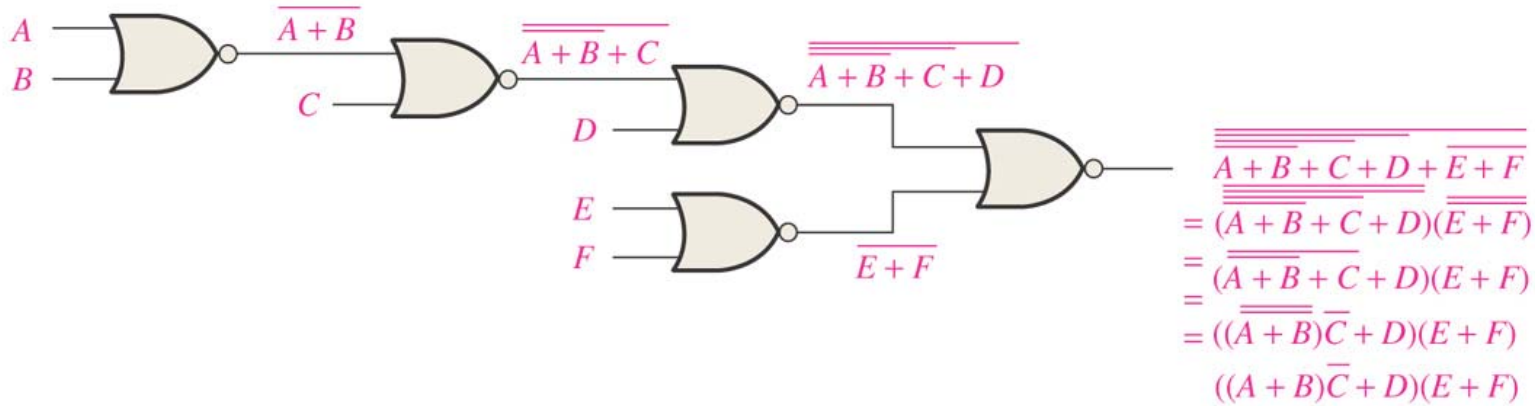


(b)

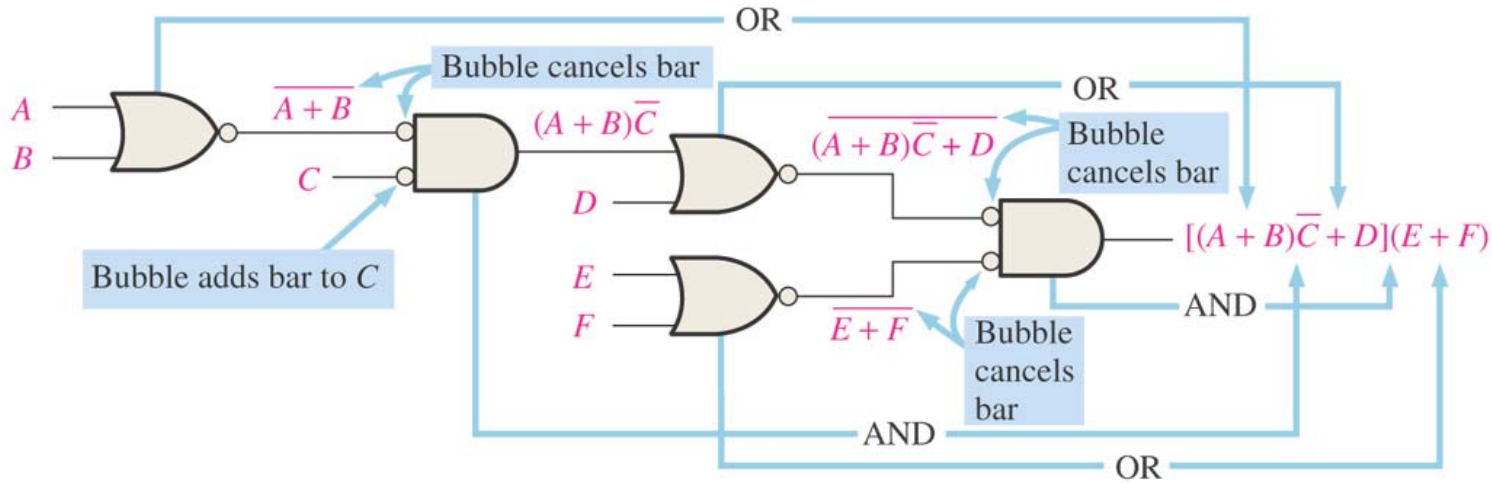
NOR logic diagrams using dual symbols

- Dual symbols – NOR symbols and Neg-NAND symbols
- Correct method of drawing a NOR logic diagram:
  - Similar to drawing a NAND logic diagram, i.e., bubble-to-bubble and nonbubble-to-nonbubble connections

Illustration of using appropriate dual symbols in a NOR logic diagram



(a) Final output expression is obtained after several Boolean steps.

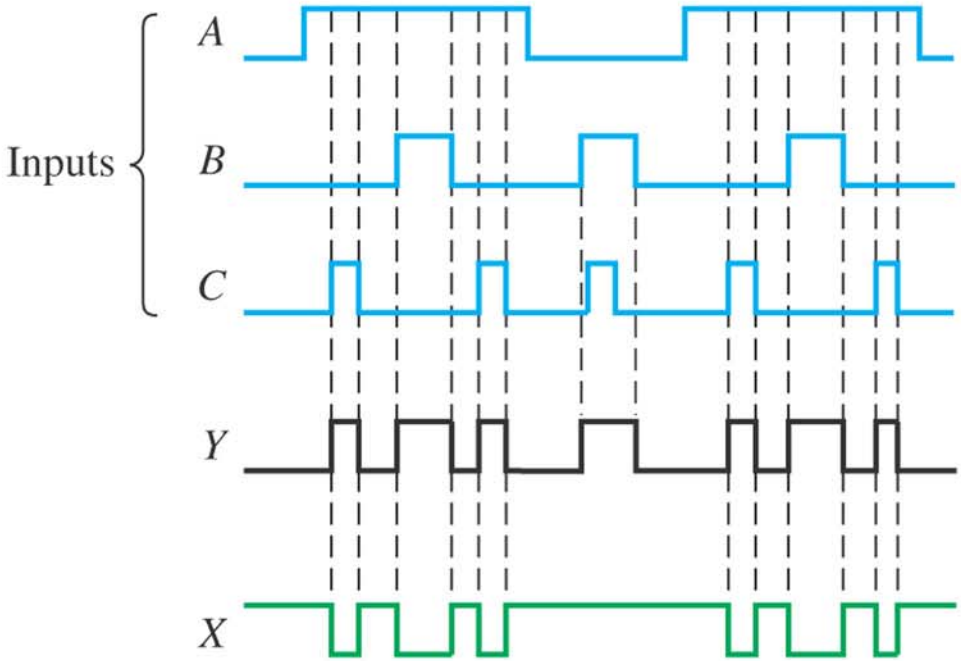


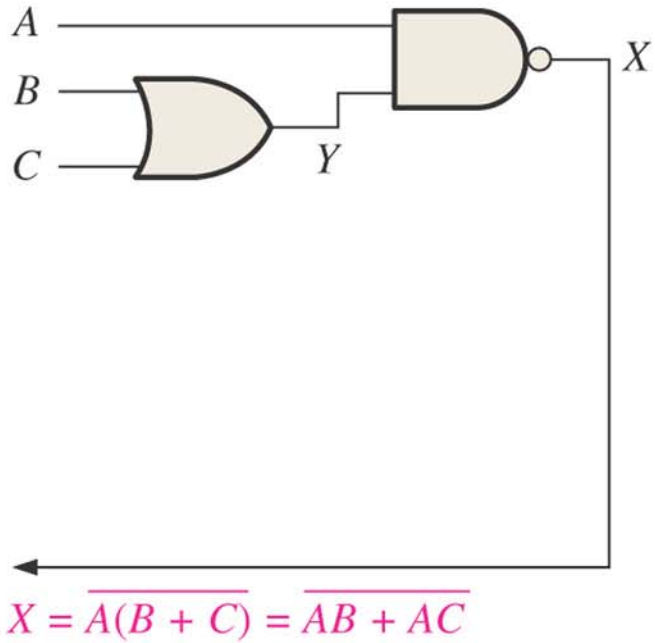
(b) Output expression can be obtained directly from the function of each gate symbol in the diagram.

5-5 Logic Circuit Operation With Pulse Waveform Inputs

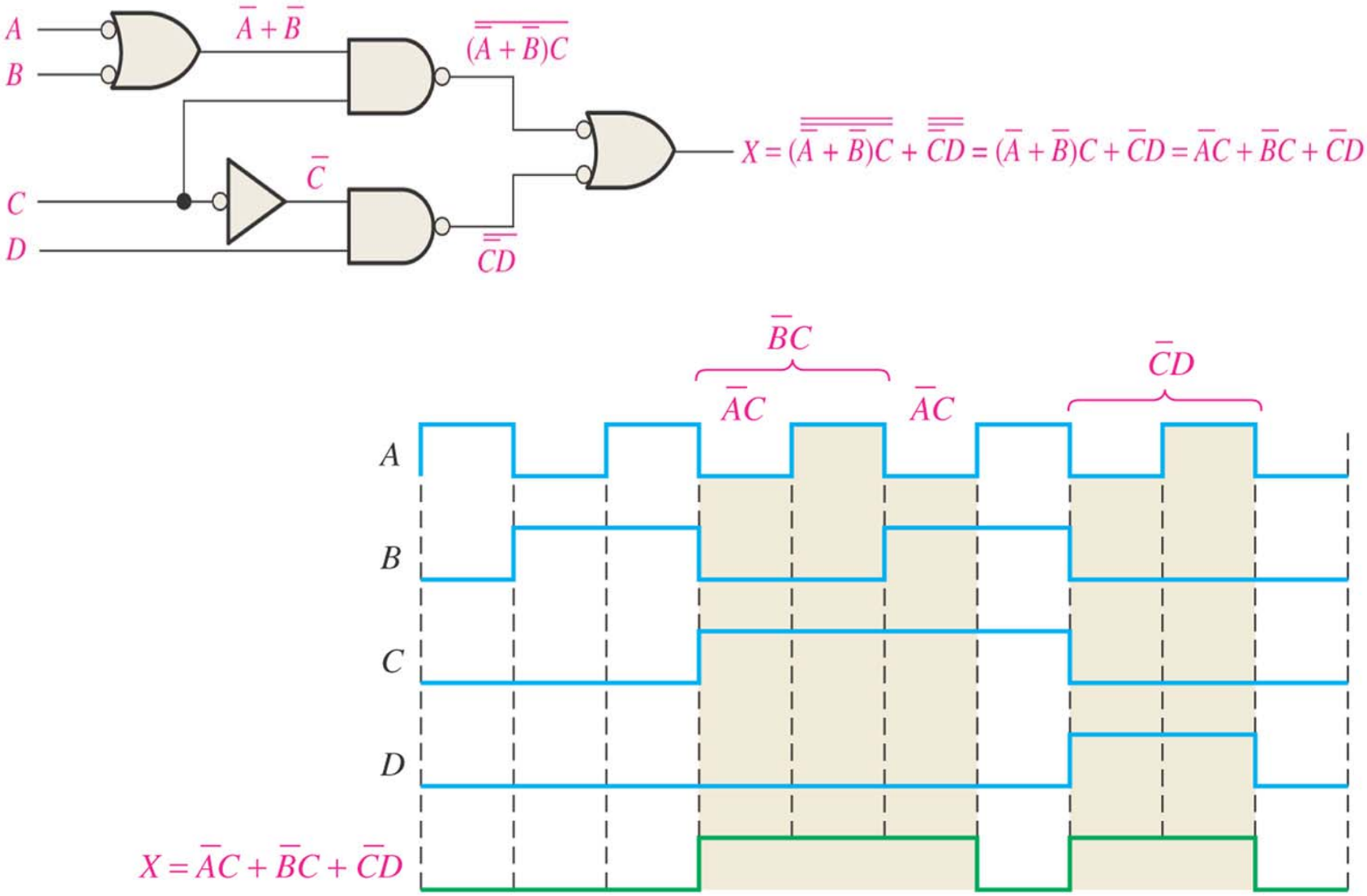
- We must note that the operation of any gate is the same regardless of whether its inputs are pulsed or constant levels
  - Pulsed means there are trains of logical ‘0’ and ‘1’ to represent LOW(OFF) and HIGH(ON)
  - Constant levels means a same value, either being always LOW or HIGH
- In both cases, the truth table remains the same
- One should know these four ‘information’ to ease analysis of combinational circuits
  1. AND: HIGH only when all inputs are HIGH at the same time
  2. OR: HIGH only when at least one of its inputs is HIGH
  3. NAND: LOW only when all inputs are HIGH at the same time
  4. NOR: LOW only when at least one of its inputs is HIGH

Example 5-10





Example 5-13


A
B
C
D
$\overline{A}+\overline{B}$
$\overline{C}$
$\overline{(\overline{A}+\overline{B})C}$
$\overline{\overline{CD}}$
$X=\overline{\overline{(\overline{A}+\overline{B})C}+\overline{\overline{CD}}}=(\overline{A}+\overline{B})C+\overline{C}D=\overline{A}C+\overline{B}C+\overline{C}D$
$\overline{B}C$
$\overline{A}C$
$\overline{A}C$
$\overline{C}D$
A
B
C
D
$X=\overline{A}C+\overline{B}C+\overline{C}D$

Summary

- We have studied the advanced method of using logic gates to produce a complex logic circuits
  - This is called “combinational logic”
- We have also discussed about how to write the logic diagram appropriately, so that we can easily read the diagram for analysis purpose
- Indirectly, we know the importance of using DeMorgan Theorem to relate NOR with Neg-AND, and NAND with Neg-OR
- At this point, we must be able to determine the output from either pulsed or constant inputs