CS221: Logic Design

Instructors:
Dr. Ahmed Shalaby http://bu.edu.eg/staff/ahmedshalaby14#
Dr. Fatma Saker

QUIZ

As part of an aircraft's functional monitoring system, a circuit is required to indicate the status of the landing gears prior to landing.

A green LED display turns on if all three gears are properly extended when the "gear down" switch has been activated in preparation for landing. A red LED display turns on if any of the gears fail to extend properly prior to landing.

When a landing gear is extended, its sensor produces a LOW voltage. When a landing gear is retracted, its sensor produces a HIGH voltage.

Implement a circuit to meet this requirement.

SOP & POS

EXAMPLE 4–20

Develop a truth table for the standard SOP expression $\overline{A}\,\overline{B}C + A\overline{B}\,\overline{C} + ABC$.

TABLE 4–6

Inputs			Output	
A	B	C	X	Product Term
0	0	0	0	
0	0	1	1	$\overline{A}\,\overline{B}C$
0	1	0	0	
0	1	1	0	
1	0	0	1	$A\overline{B}\,\overline{C}$
1	0	1	0	
1	1	0	0	
1	1	1	1	ABC

TABLE 4–7

Inputs			Output	
A	B	C	X	Sum Term
0	0	0	0	$(A + B + C)$
0	0	1	1	
0	1	0	0	$(A + \overline{B} + C)$
0	1	1	0	$(A + \overline{B} + \overline{C})$
1	0	0	1	
1	0	1	0	$(\overline{A} + B + \overline{C})$
1	1	0	0	$(\overline{A} + \overline{B} + C)$
1	1	1	1	

EXAMPLE 4–21

Determine the truth table for the following standard POS expression:

$$(A + B + C)(A + \overline{B} + C)(A + \overline{B} + \overline{C})(\overline{A} + B + \overline{C})(\overline{A} + \overline{B} + C)$$

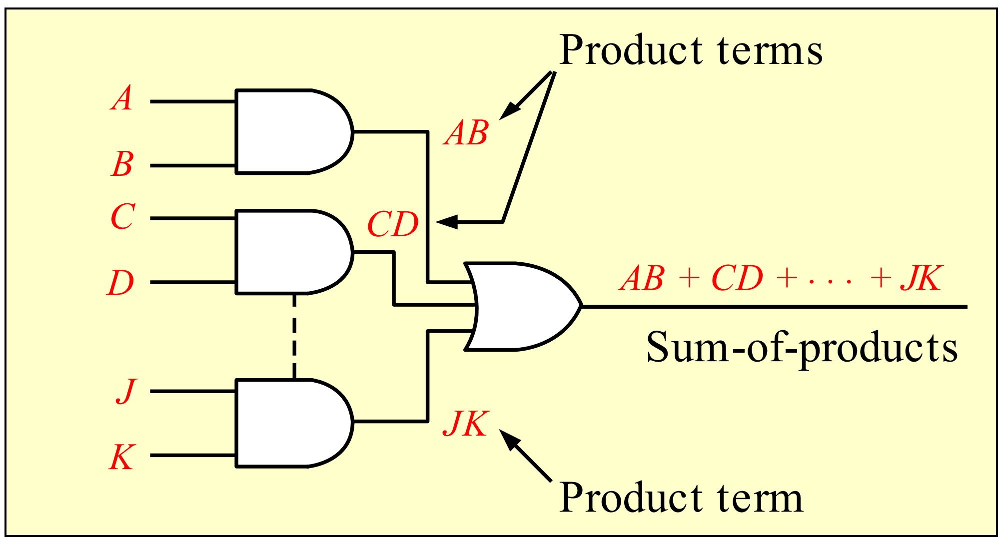
Digital Fundamentals

CHAPTER 5
Combinational Logic Analysis

Combinational Logic Analysis

Combinational Logic Circuits

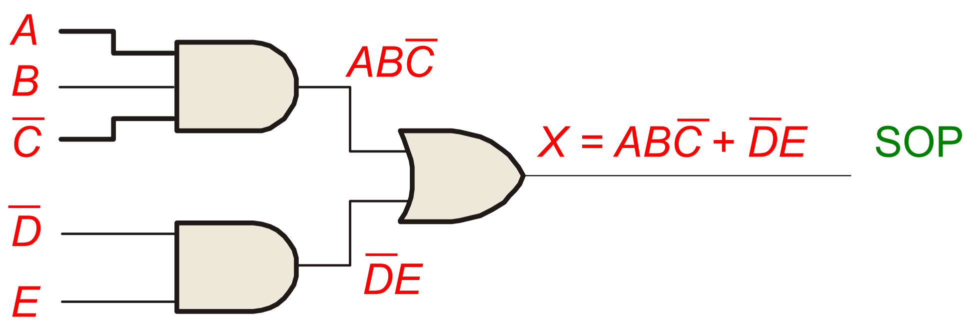
In Sum-of-Products (SOP) form, basic combinational circuits can be directly implemented with AND-OR combinations if the necessary complement terms are available.



Combinational Logic Analysis

Combinational Logic Circuits

An example of an SOP implementation is shown. The SOP expression is an AND-OR combination of the input variables and the appropriate complements.

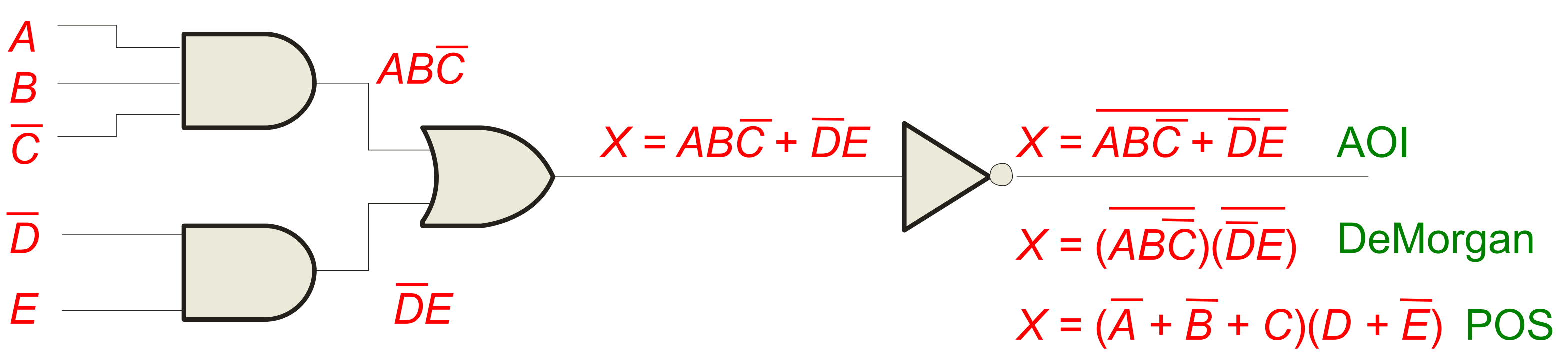


Combinational Logic Analysis

Combinational Logic Circuits

When the output of a SOP form is inverted, the circuit is called an AND-OR-Invert (AOI) circuit. The AOI configuration lends itself to product-of-sums (POS) implementation.

An example of an AOI implementation is shown. The output expression can be changed to a POS expression by applying DeMorgan's theorem twice.



Combinational Logic Analysis

Exclusive-OR Logic

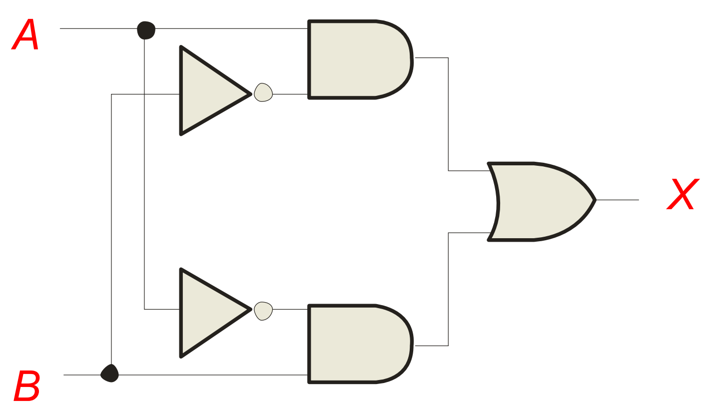
The truth table for an exclusive-OR gate is

Inputs		Output
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	0

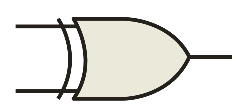
Notice that the output is HIGH whenever *A* and *B* <u>disagree</u>.

The Boolean expression is $X = \bar{A}B + A\bar{B}$

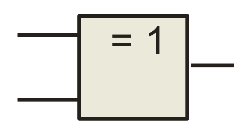
The circuit can be drawn as



Symbols:



Distinctive shape



Rectangular outline

Combinational Logic Analysis

Exclusive-NOR Logic

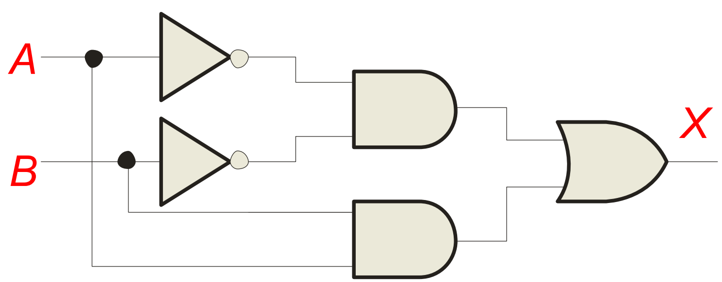
The truth table for an exclusive-NOR gate is

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

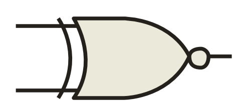
Notice that the output is HIGH whenever *A* and *B* <u>agree</u>.

The Boolean expression is $X = \overline{A}\overline{B} + AB$

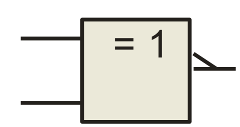
The circuit can be drawn as



Symbols:



Distinctive shape

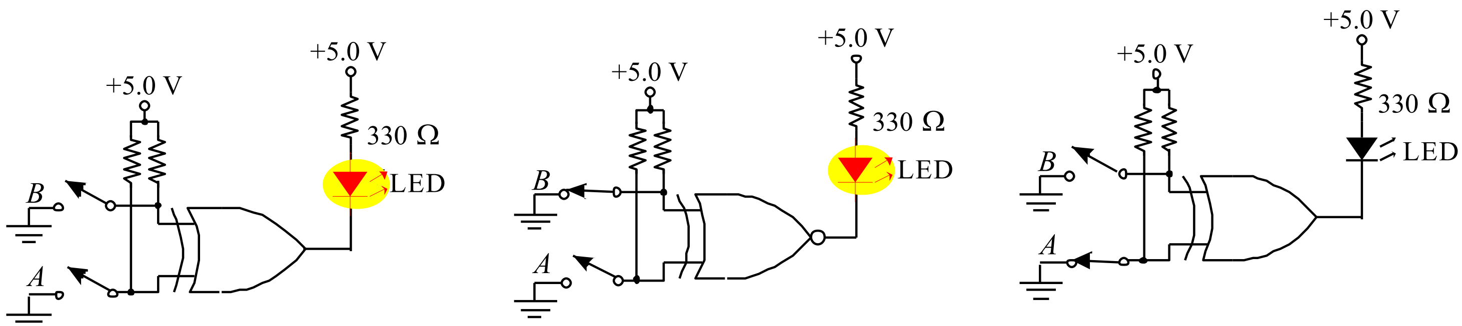


Rectangular outline

Combinational Logic Analysis

Example

For each circuit, determine if the LED should be on or off.



(a) (b) (c)

Solution

Circuit (a): XOR, inputs agree, output is LOW, LED is ON.

Circuit (b): XNOR, inputs disagree, output is LOW, LED is ON.

Circuit (c): XOR, inputs disagree, output is HIGH, LED is OFF.

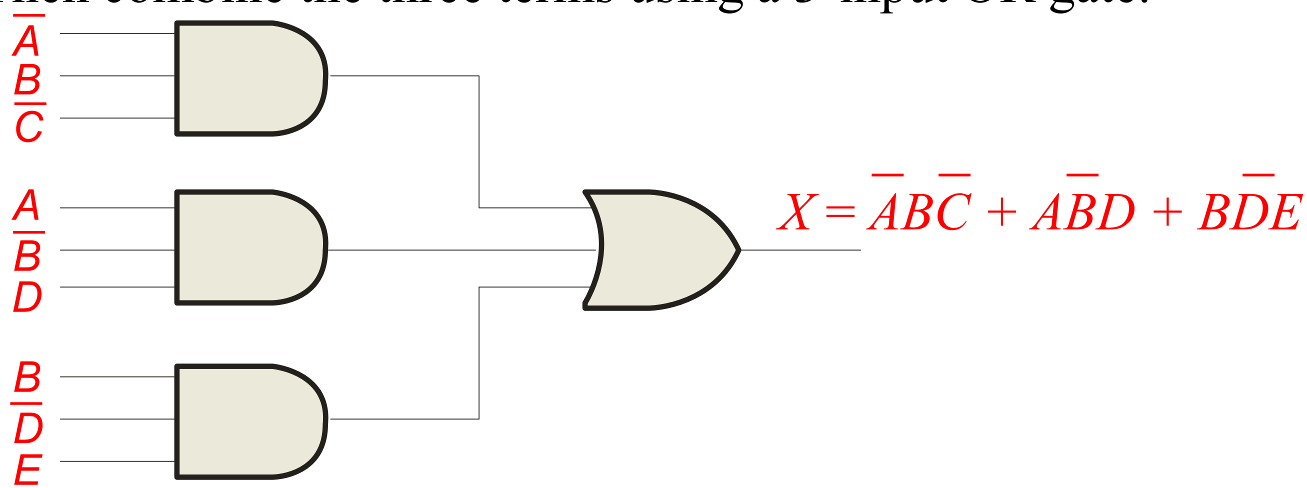
Implementing Combinational Logic

Implementing Combinational Logic

Implementing a SOP expression is done by first forming the AND terms; then the terms are ORed together.

Example Show the circuit that will implement the Boolean expression $X = \overline{A}B\overline{C} + A\overline{B}D + B\overline{D}E$. (Assume that the variables and their complements are available.)

Solution Start by forming the terms using three 3-input AND gates. Then combine the three terms using a 3-input OR gate.

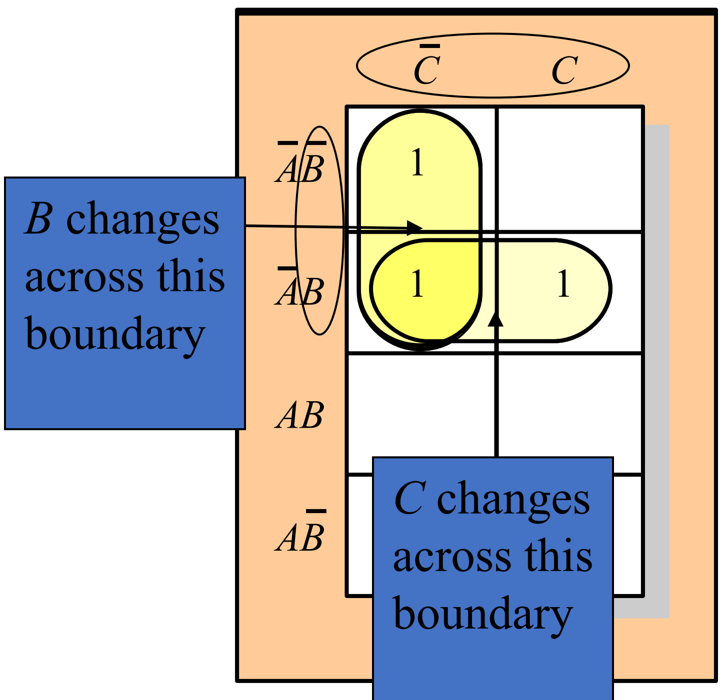


Implementing Combinational Logic

Karnaugh Map Implementation

For basic combinational logic circuits, the Karnaugh map can be read and the circuit drawn as a minimum SOP.

Example A Karnaugh map is drawn from a truth table. Read the minimum SOP expression and draw the circuit.



Solution

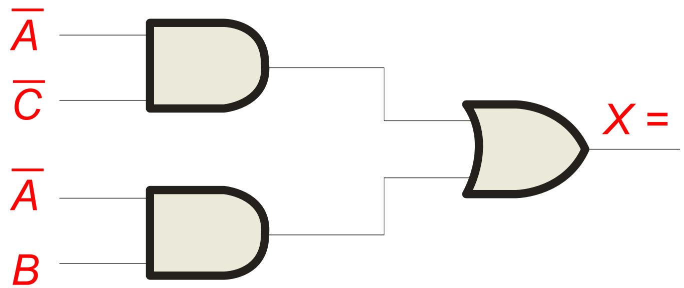
1. Group the 1's into two overlapping groups as indicated.
2. Read each group by eliminating any variable that changes across a boundary.
3. The vertical group is read $\overline{A}\,\overline{C}$.
4. The horizontal group is read $\overline{A}B$.

The circuit is on the next slide:

Implementing Combinational Logic

Solution *continued…*

Circuit:



$X = \overline{A}\,\overline{C} + \overline{A}B$

The result is shown as a sum of products.

It is a simple matter to implement this form using only NAND gates as shown in the text and following example.

The Universal Property of NAND and NOR Gates

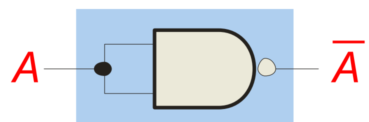
NAND and NOR gates are “universal” because they can used to produce any of the other logic functions.

Combinational Logic Analysis

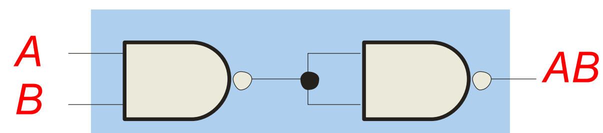
Universal Gates

NAND gates are sometimes called **universal** gates because they can be used to produce the other basic Boolean functions.

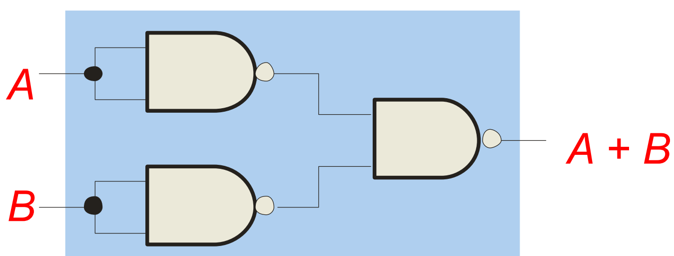
Inputs		Output
A	*B*	*X*
0	0	1
0	1	1
1	0	1
1	1	0



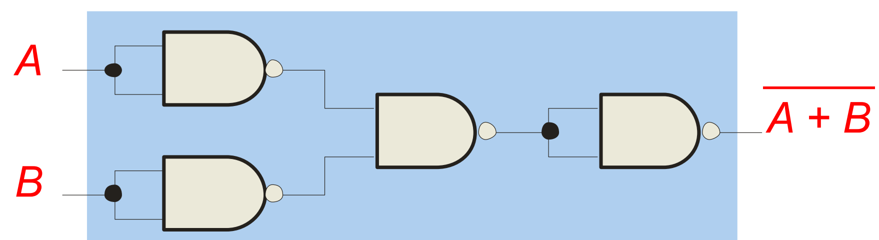
Inverter



AND gate



OR gate



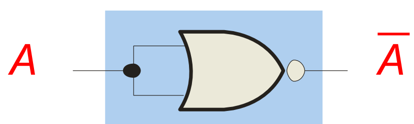
NOR gate

Combinational Logic Analysis

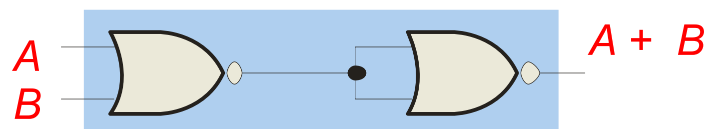
Universal Gates

NOR gates are also **universal** gates and can form all of the basic gates.

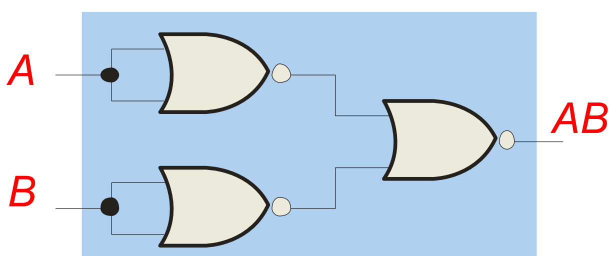
Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	0



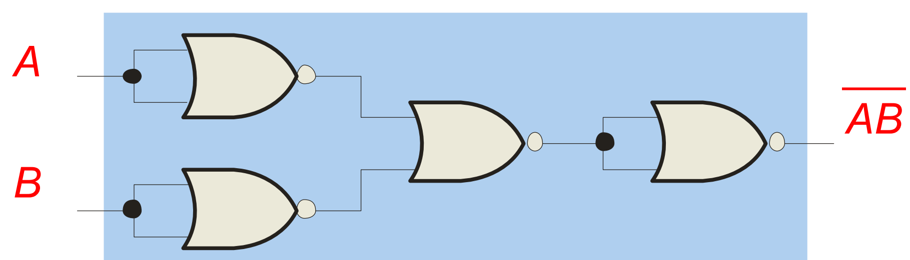
Inverter



OR gate



AND gate

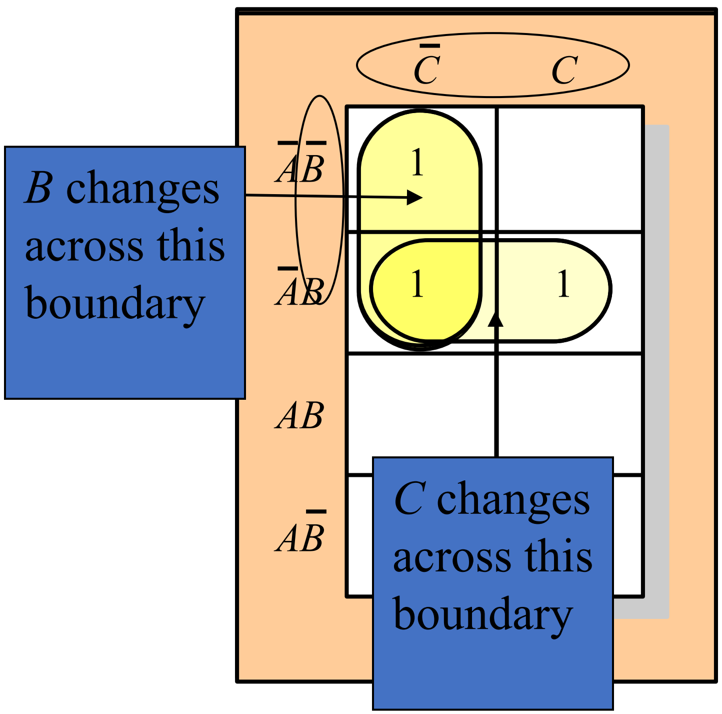


NAND gate

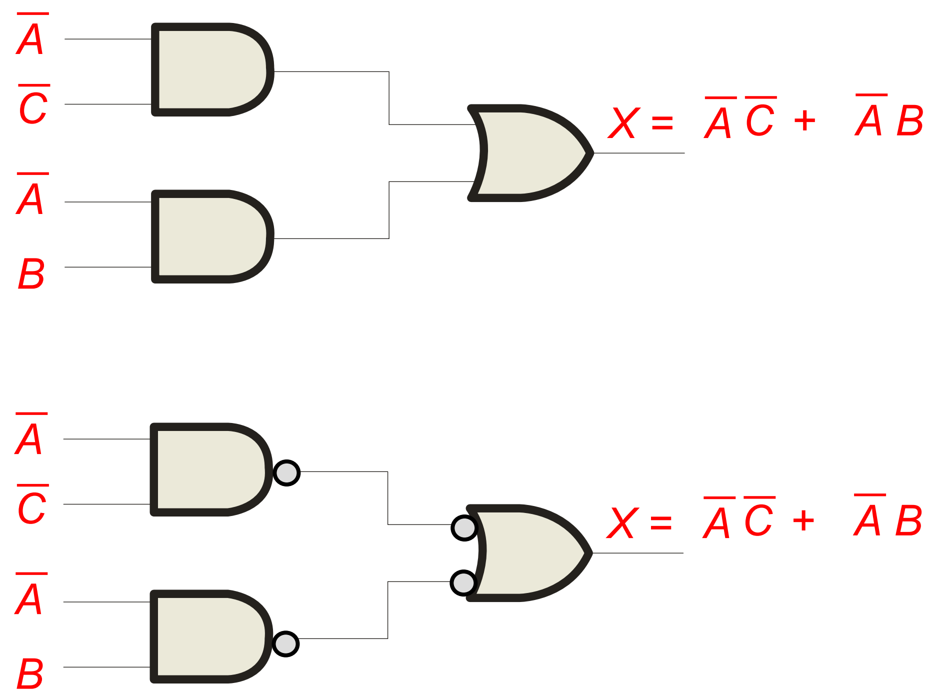
Combinational Logic Analysis

Solution

continued…Slide 12



Circuit:



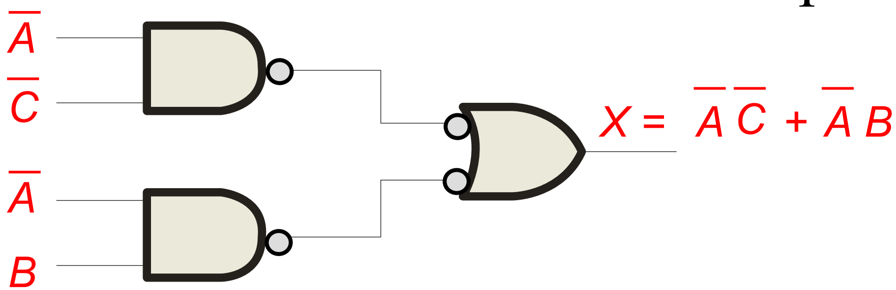
Recall from Boolean algebra that double inversion cancels. By adding inverting bubbles to above circuit, it is easily converted to NAND gates.

Combinational Logic Analysis

NAND Logic

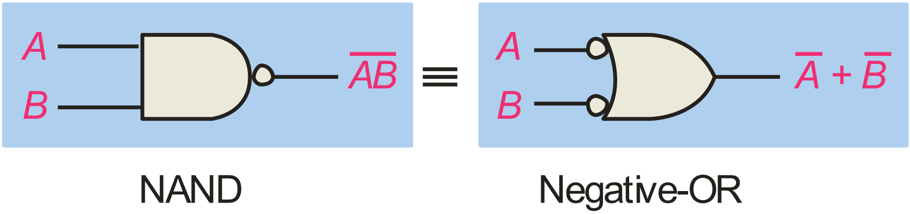
Recall from DeMorgan's theorem that $\overline{AB} = \overline{A} + \overline{B}$.

By using equivalent symbols, it is simpler to read the logic of SOP forms. The earlier example shows the idea:



The logic is easy to read if you (mentally) cancel the two connected bubbles on a line.

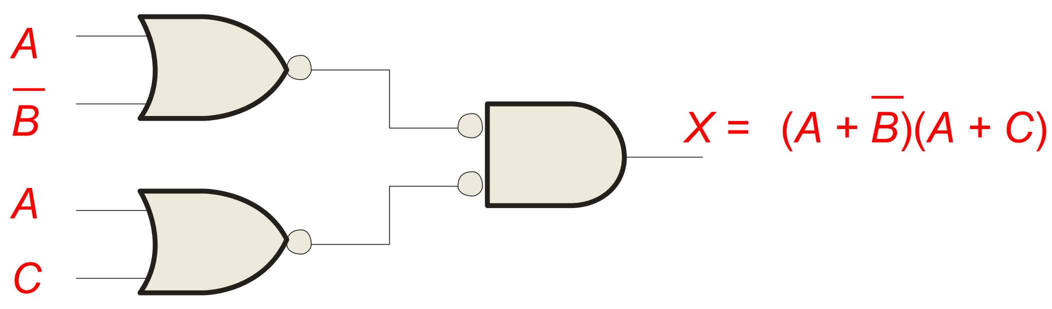
Inputs A	Inputs B	Output $\overline{AB}$	Output $\overline{A} + \overline{B}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0



Combinational Logic Analysis

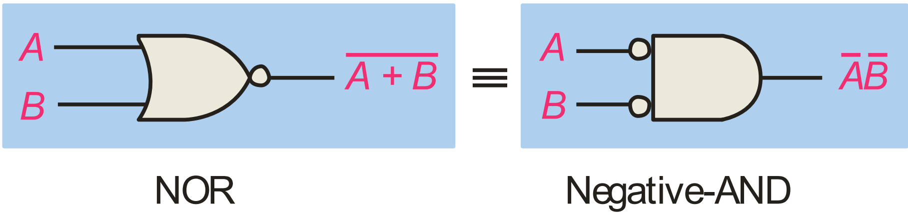
NOR Logic

Alternatively, DeMorgan's theorem can be written as $\overline{A + B} = \overline{A}\,\overline{B}$. By using equivalent symbols, it is simpler to read the logic of POS forms. For example,



Inputs		Output	
A	B	$\overline{A + B}$	$\overline{A}\,\overline{B}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

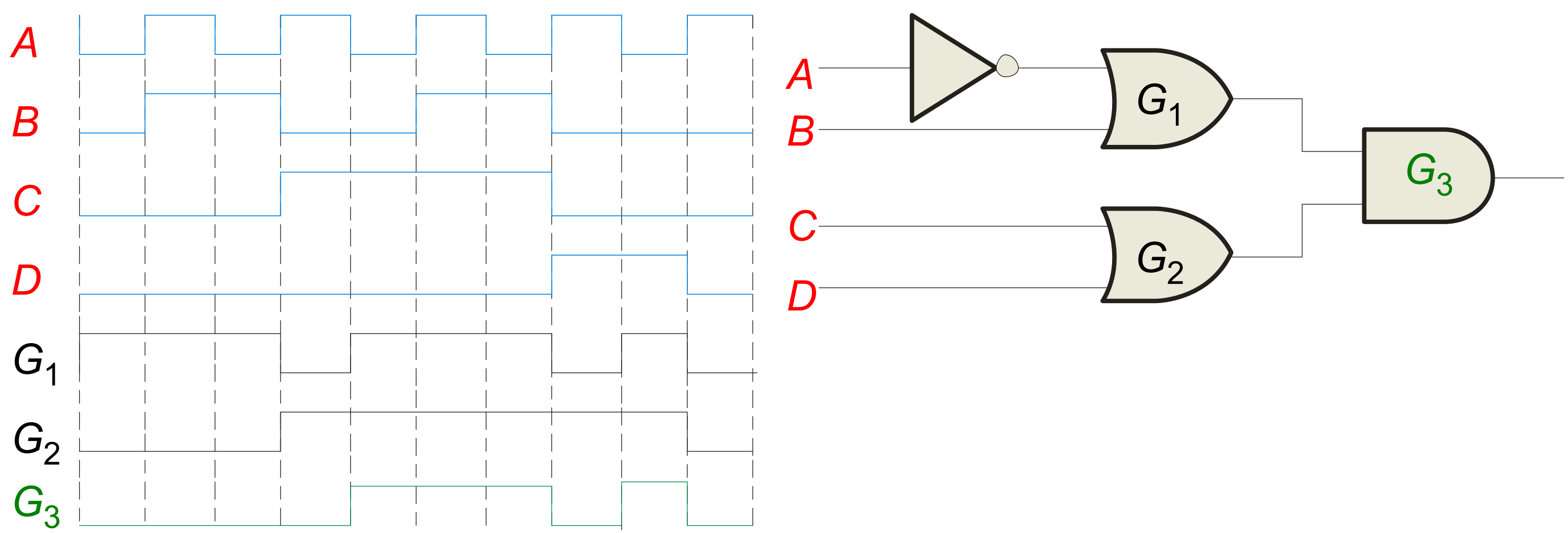
Again, the logic is easy to read if you cancel the two connected bubbles on a line.



Combinational Logic Analysis

Pulsed Waveforms

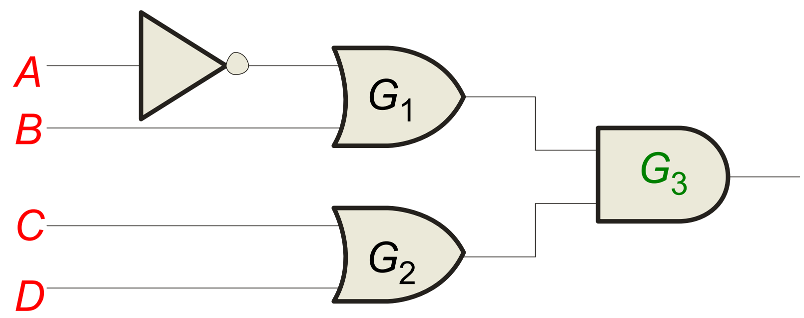
For combinational circuits with pulsed inputs, the output can be predicted by developing intermediate outputs and combining the result. For example, the circuit shown can be analyzed at the outputs of the OR gates:

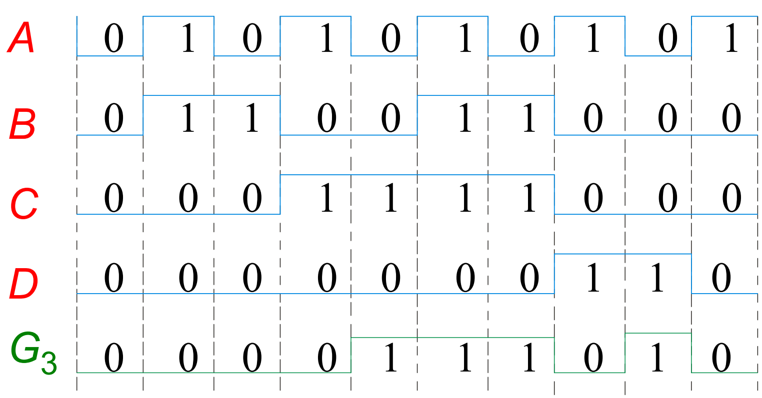


Combinational Logic Analysis

Pulsed Waveforms

Alternatively, you can develop the truth table for the circuit and enter 0's and 1's on the waveforms. Then read the output from the table.





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

Combinational Logic Analysis

Universal gate	Either a NAND or a NOR gate. The term universal refers to a property of a gate that permits any logic function to be implemented by that gate or by a combination of gates of that kind.
Negative-OR	The dual operation of a NAND gate when the inputs are active-LOW.
Negative-AND	The dual operation of a NOR gate when the inputs are active-LOW.

QUIZ

Analyze the Output of the shown Circuit Using timing diagram and truth tables for the input waveforms.