Boolean Logic

LOGIC GATES

George Boole, (1815-1864)

Did you know?

George Boole Inventor of the idea of logic gates. He was born in Lincoln, England and he was the son of a shoemaker in a low class family.

- □ In the **1850s** George **Boole** developed a new form of algebra, now called Boolean algebra in his honour.
- □ Boolean equations use the **binary** number system to provide a very precise way of illustrating the logic of **computer chips**.
- □ Interesting fact: Boolean equations were **used long before computers or even electricity was invented**!

Review

- 1 in the Binary System represents
  - ON or YES

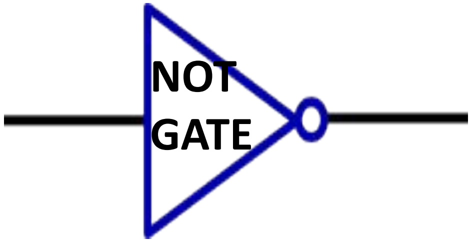
- 0 in the Binary System represents
  - OFF or NO

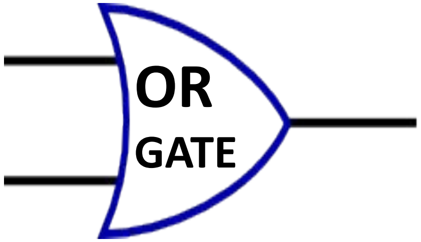
- Boolean logic is a part of almost every aspect of COMPUTER ELECTRONICS

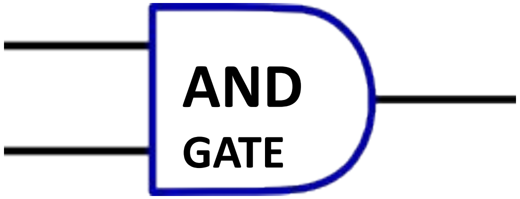
- Example: using the + while performing Google Searches

A logic gate is a digital circuit which either allows a signal to pass through it or to stop it.

There are seven basic logic gates: AND, OR, XOR, NOT, NAND, NOR, and XNOR.

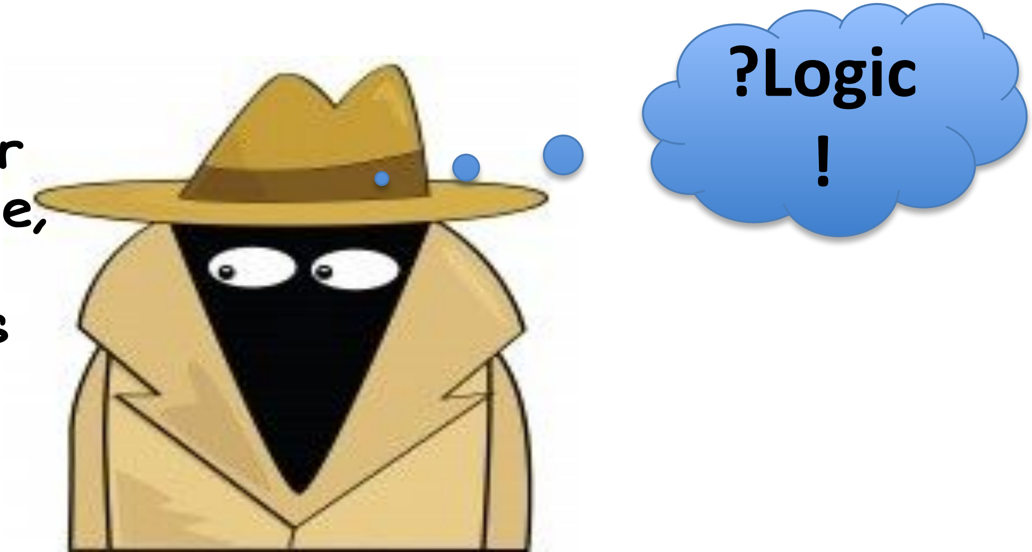


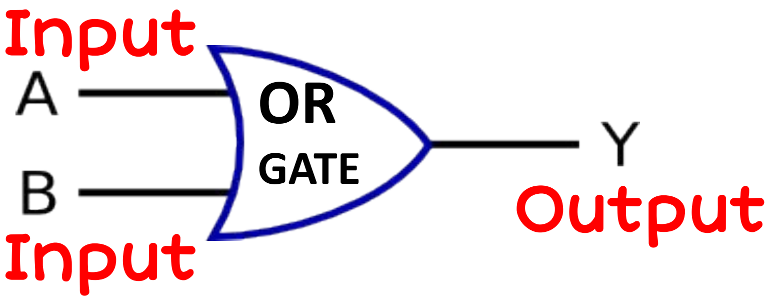




Did you know?

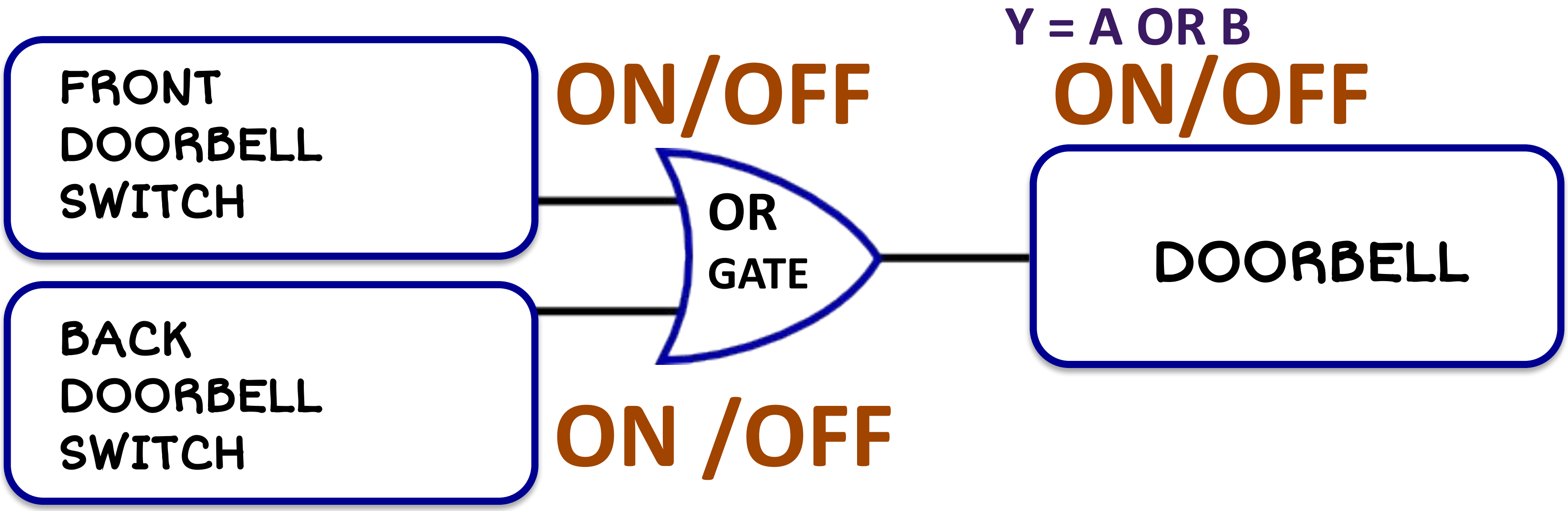
Logic gates allow the computer to do things such as add, divide, multiply, do simple yes and no reasoning in certain situations along with other things.





An OR gate can have two or more inputs.

The output will be positive (True) if at least one input is true.



Y = A OR B

FRONT DOORBELL SWITCH

ON/OFF

OR GATE

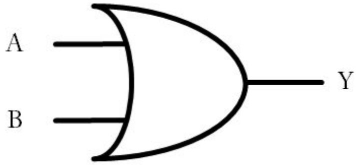
ON/OFF

DOORBELL

BACK DOORBELL SWITCH

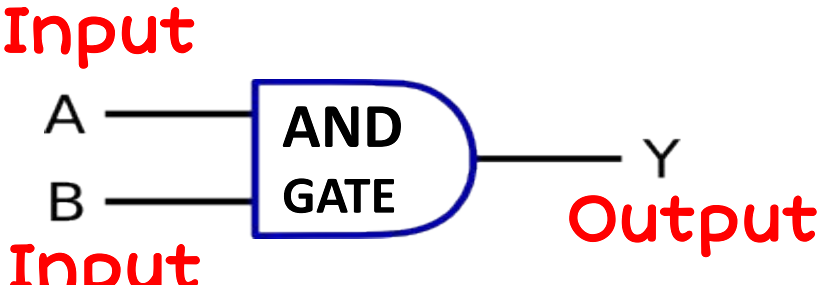
ON /OFF

OR



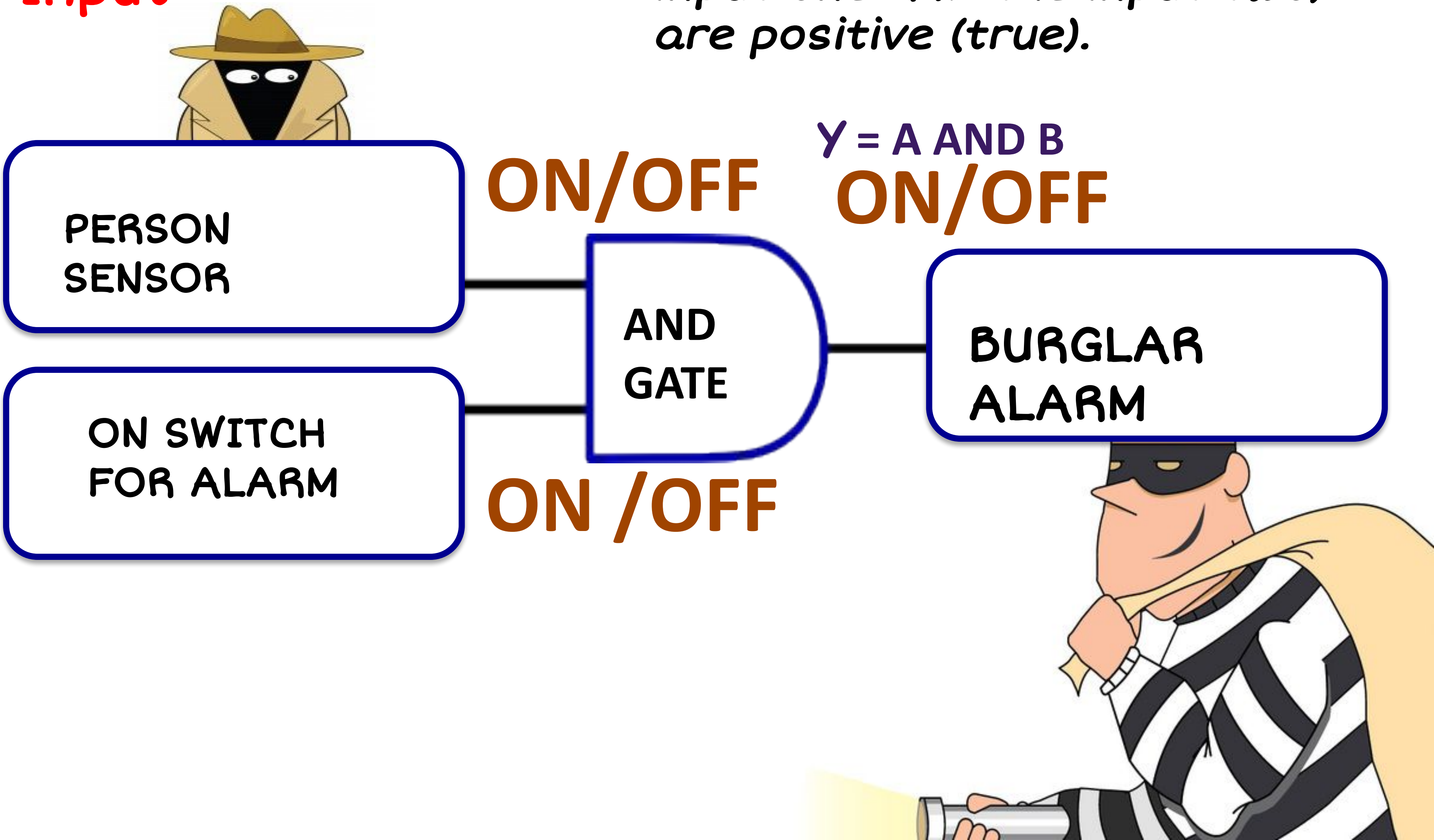
A	B	Y (Output)
0	0	0
0	1	1
1	0	1
1	1	1

If either input is 1 (YES) the output is 1 (YES).

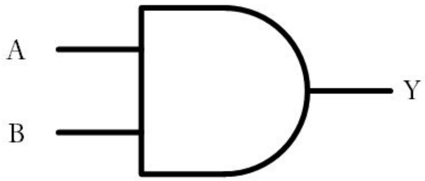


An AND gate can have two or more inputs.
The output will be positive (true) when both inputs (the input one AND the input two) are positive (true).

Y = A AND B



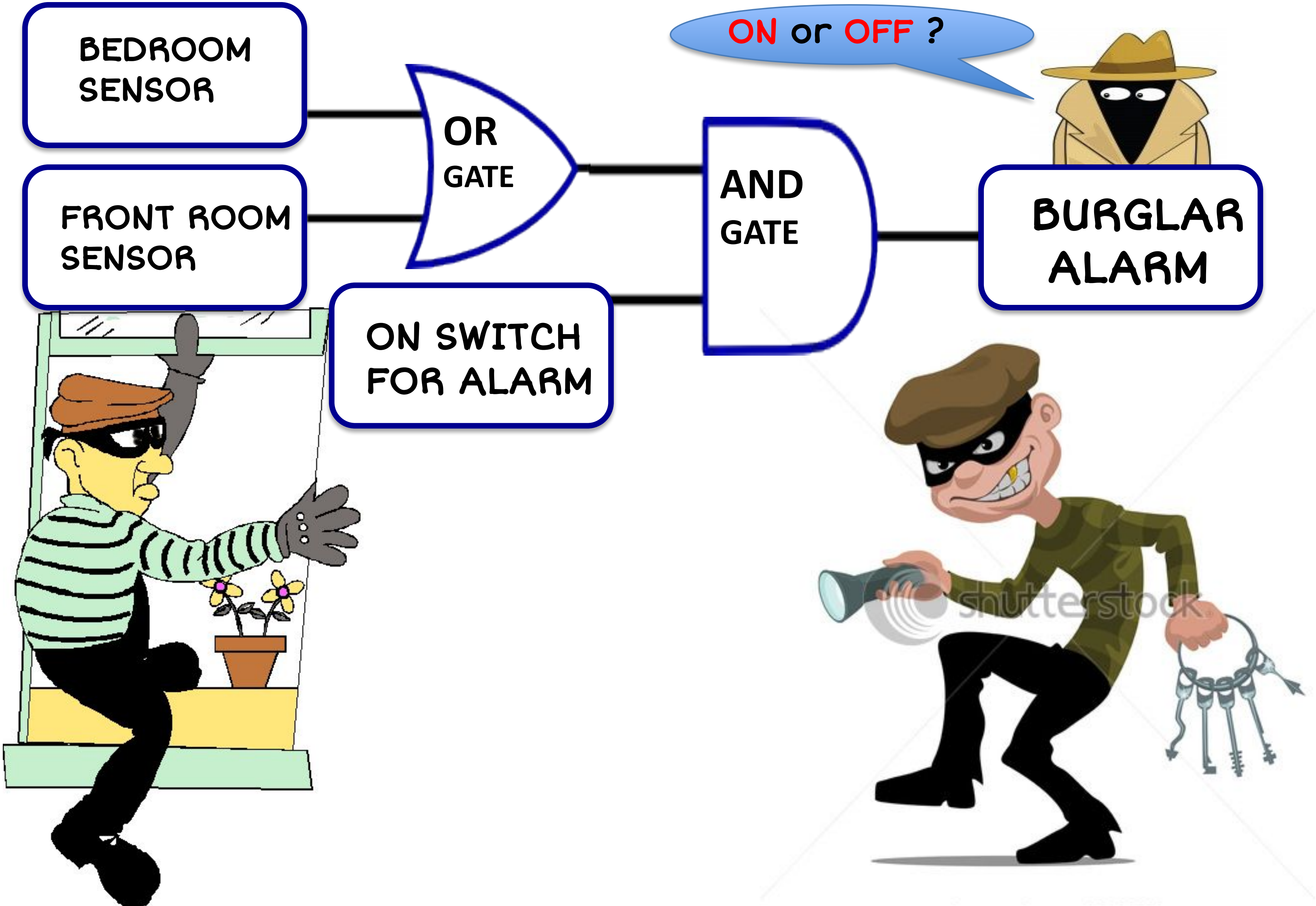
AND



A	B	Y (Output)
0	0	0
0	1	0
1	0	0
1	1	1

Both inputs must be 1(YES) to get a 1(YES) as output.

TO ADD MORE SENSORS TO ALARM



BEDROOM SENSOR

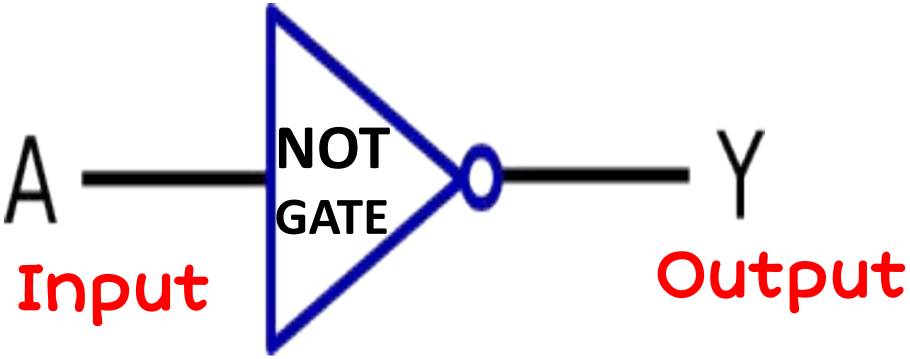
FRONT ROOM SENSOR

OR GATE

ON SWITCH FOR ALARM

AND GATE

BURGLAR ALARM


A

NOT GATE

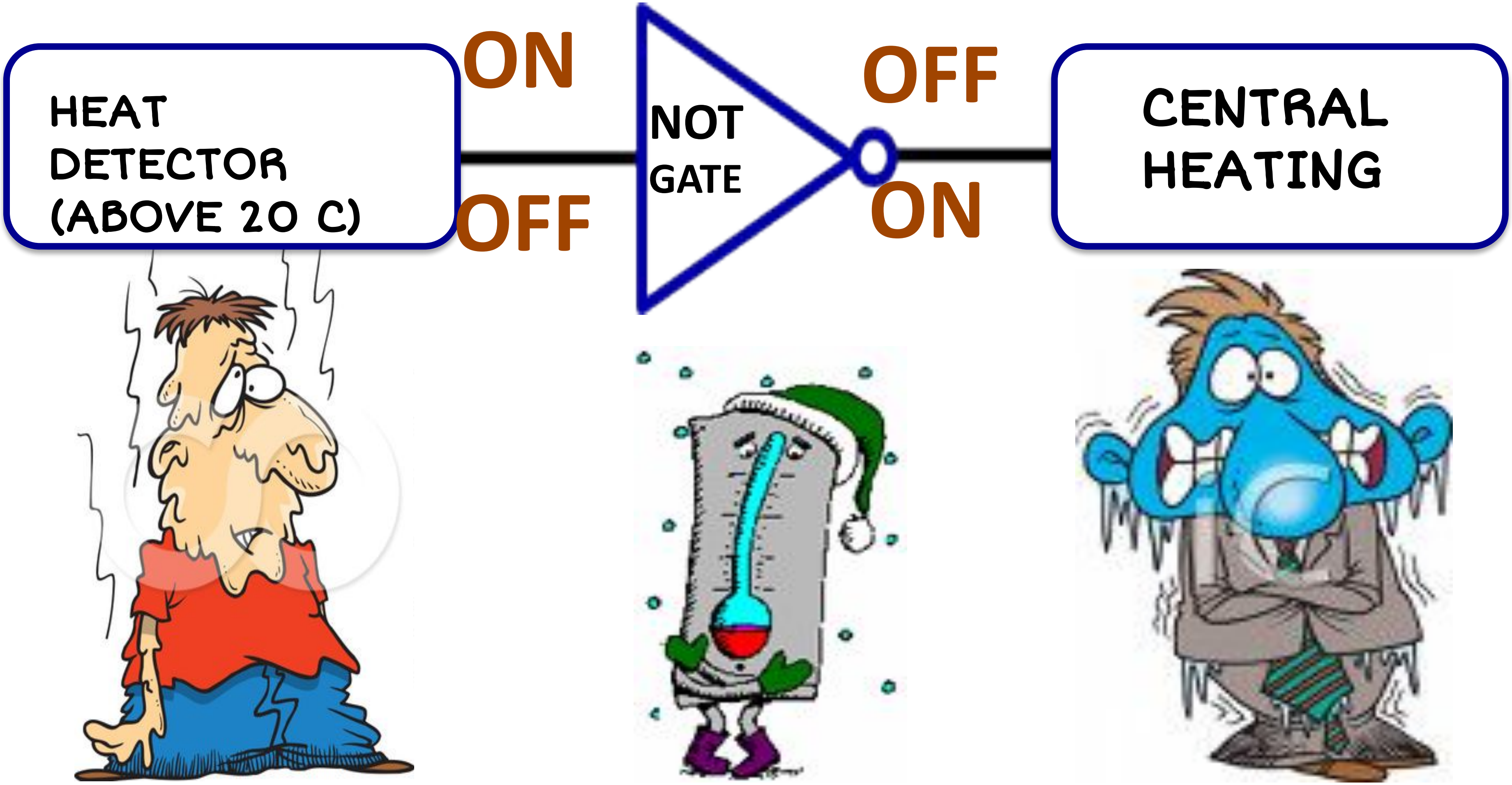
Y

Input

Output

A NOT gate (inverter) has only one input.
It reverses the logic state.

Y = NOT A


HEAT DETECTOR (ABOVE 20 C)

ON

OFF

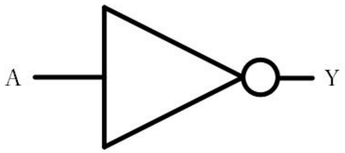
NOT GATE

OFF

ON

CENTRAL HEATING

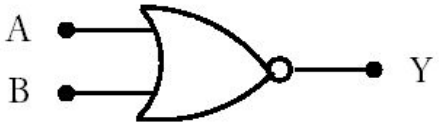
NOT



A	Y (Output)
0	1
1	0

Output is opposite of input.

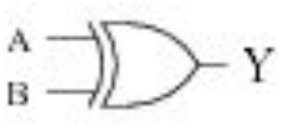
NOR



A	B	Y (Output)
0	0	1
0	1	0
1	0	0
1	1	0

For a value of 1(YES) – both inputs need to be 0(NO).
(OPPOSITE OF OR)

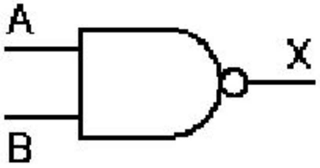
XOR



A	B	Y (Output)
0	0	0
0	1	1
1	0	1
1	1	0

Only one input can be 1(YES) to have a 1(YES) output.

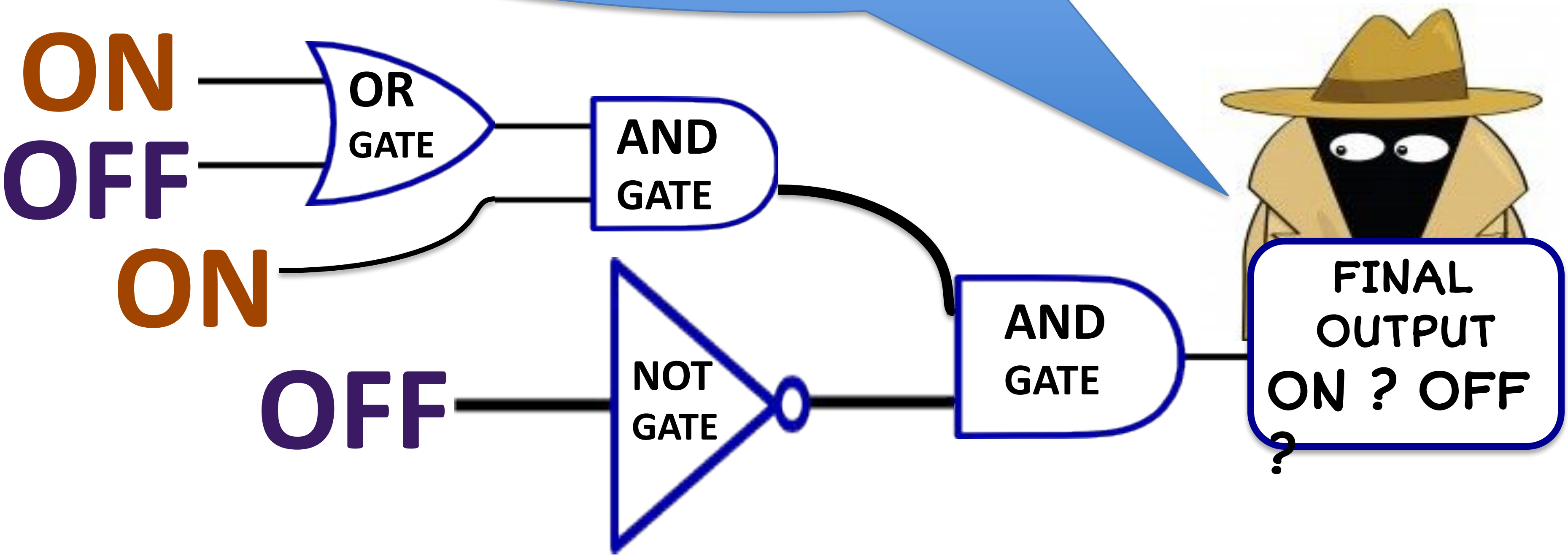
NAND



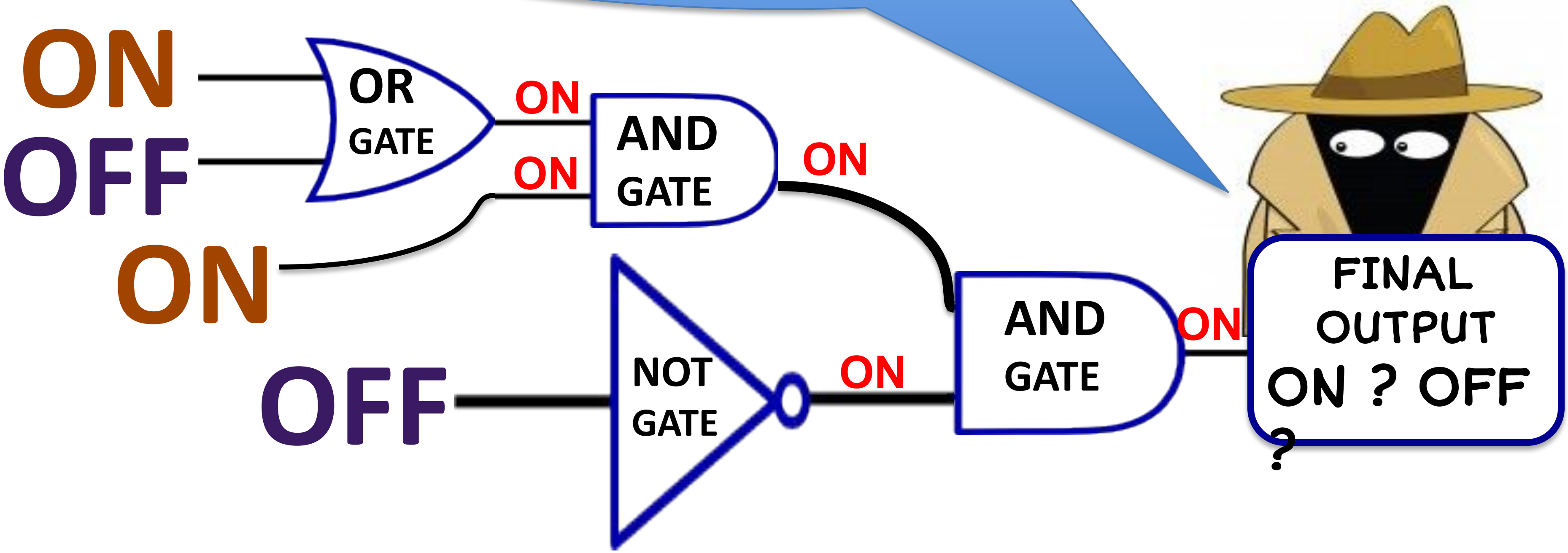
A	B	Y (Output)
0	0	1
0	1	1
1	0	1
1	1	0

If both inputs are 1(YES) output is 0(NO).
OPPOSITE OF AND!!

Is the Final Output ON (True)
or OFF (false) ?



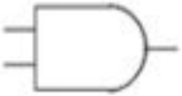
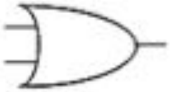
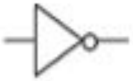
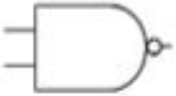
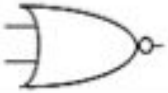
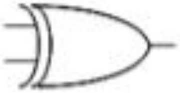
Is the Final Output ON (True)
or OFF (false) ?



ON
OFF
ON
OFF
OR
GATE
ON
ON
AND
GATE
ON
NOT
GATE
ON
AND
GATE
ON
FINAL
OUTPUT
ON ? OFF
?

ANSWER: ON

Boolean Expressions

Gate	Symbol	Operator
and		$\bullet$
or		$+$
not		$-$
nand		$\overline{\bullet}$
nor		$\overline{+}$
xor		$\oplus$

AND $Y = A \cdot B$

OR $Y = A + B$

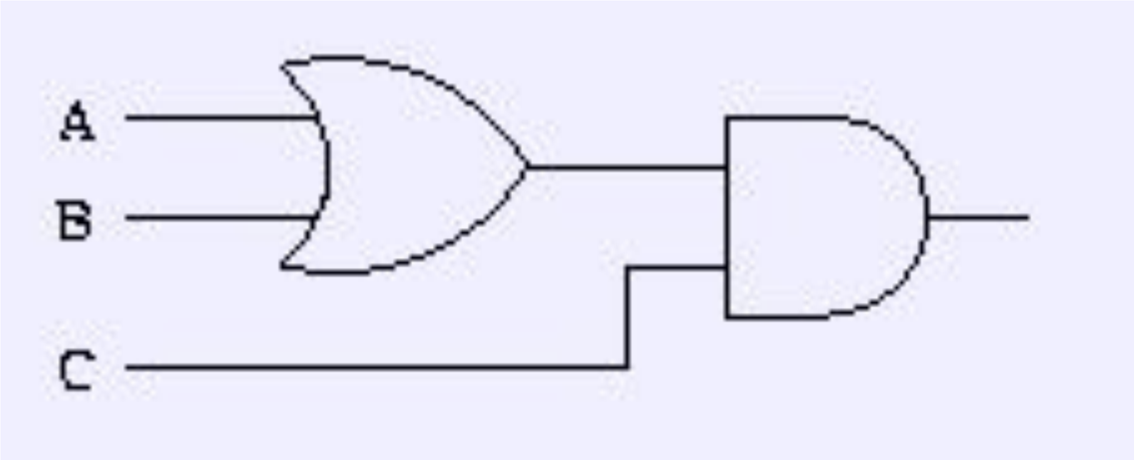
NOT $Y = \overline{A}$

NOR $Y = \overline{A + B}$

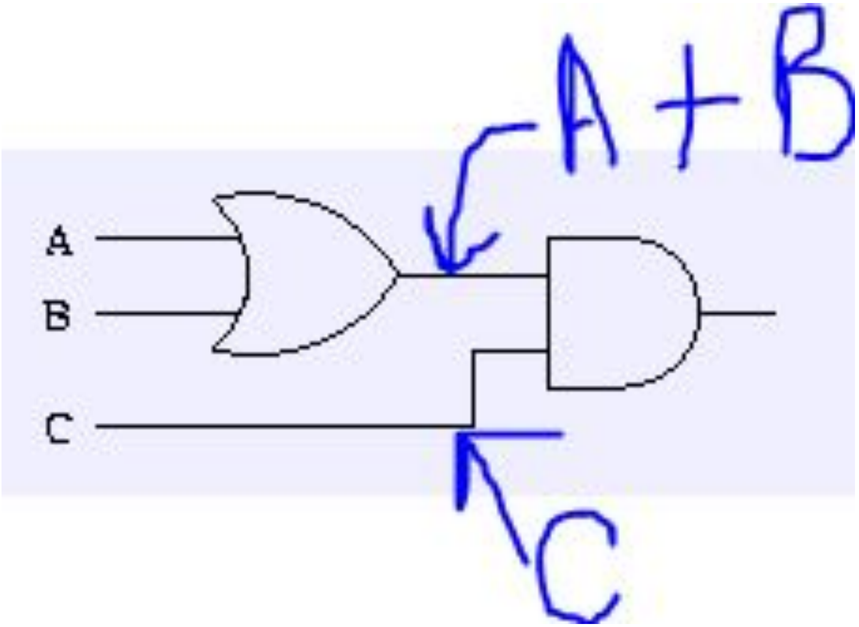
NAND $Y = \overline{A \cdot B}$

XOR $Y = A \oplus B$

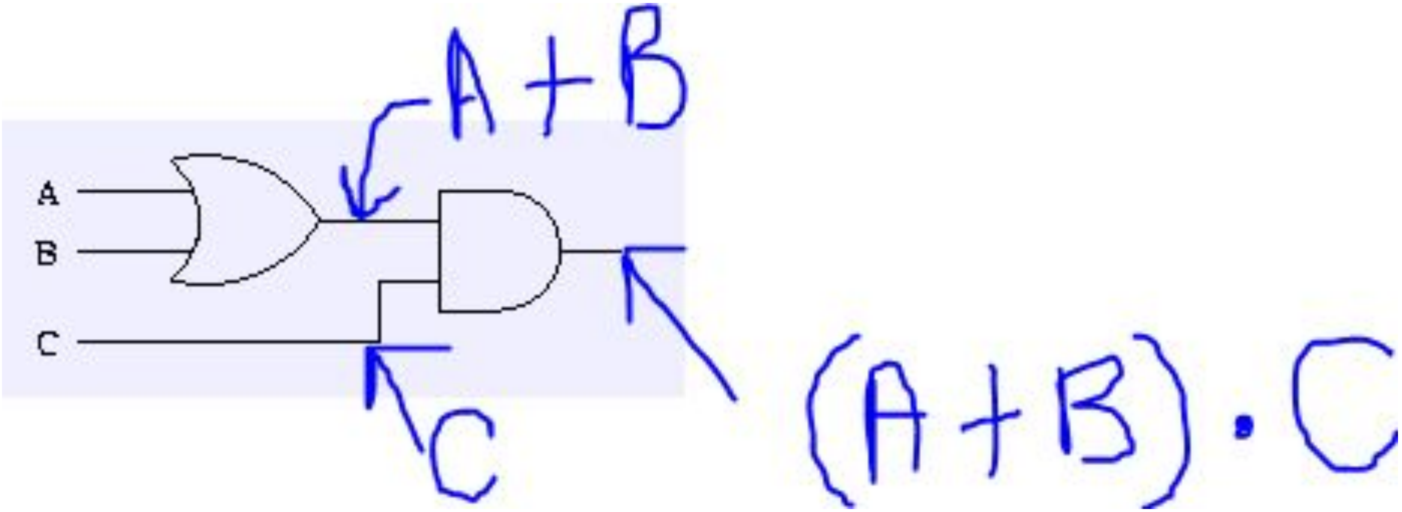
Boolean Algebra Example:



Boolean Algebra Example:



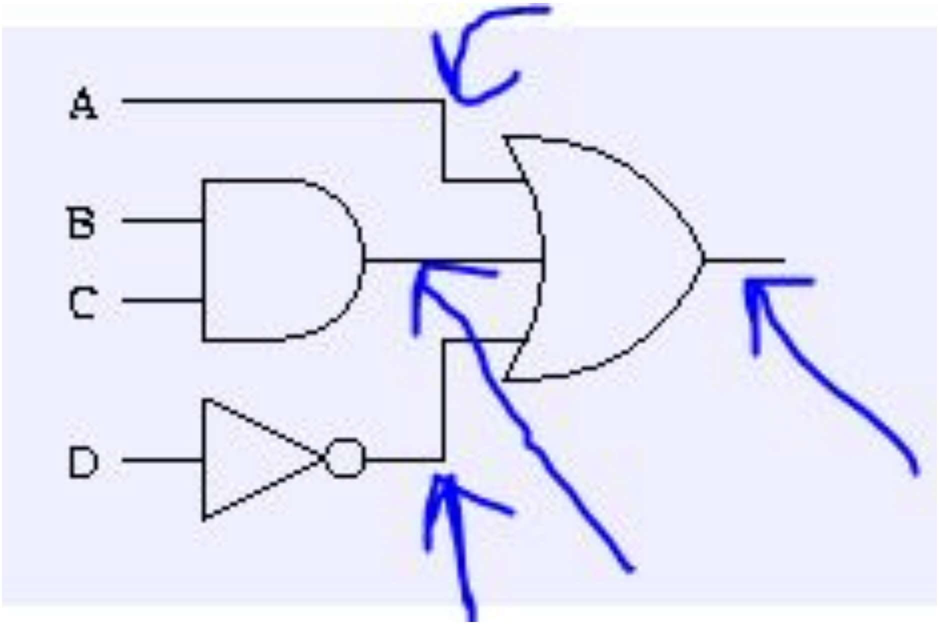
Boolean Algebra Example:



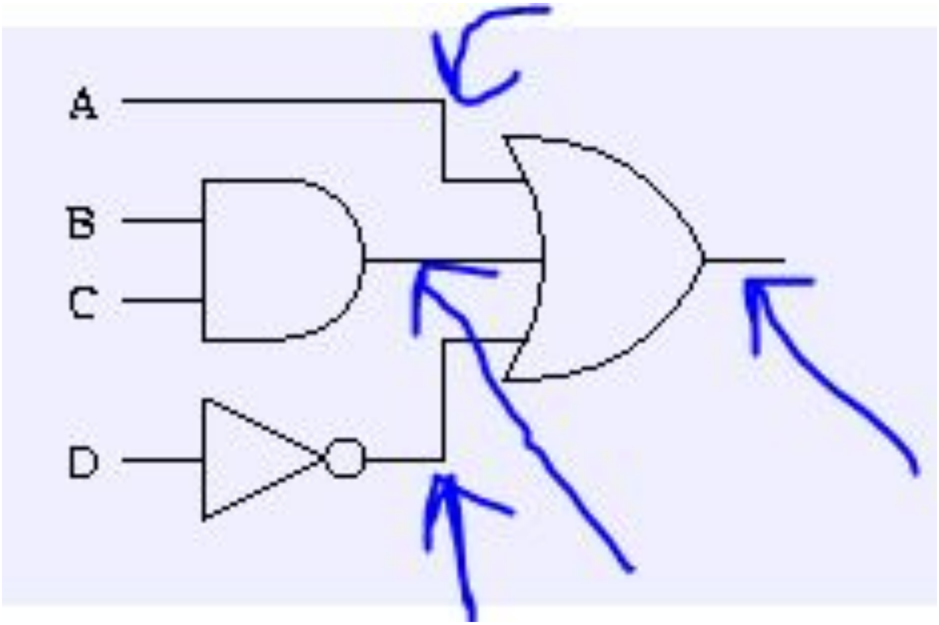
Your answer here would be:

(A + B)C

Try this one:



Try this one:



Answer: A + (BC) + D'

**The ' means NOT but when you write it on paper, you will use a horizontal bar

Question #	Answer: ON or OFF?
1	Off
2	On
3	Off
4	On
5	Off
6	On
7	On
8	Off
9	Off
10	On