

1

INTRO TO BOOLEAN ALGEBRA

2

BOOLEAN LOGIC

- Boolean logic allows us to represent logic circuits with mathematical statements.
- Since logic statements can only involve 1's and 0's we have to be creative with how we represent different gates.

3

BOOLEAN AND GATE

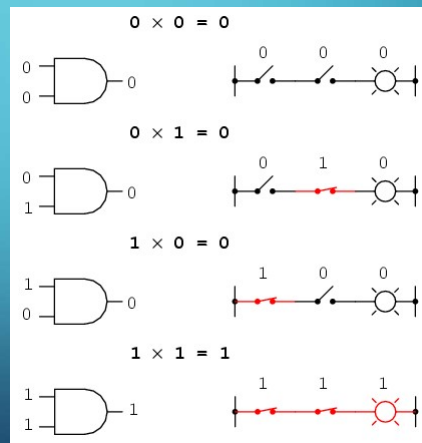
- An AND gate is represented by multiplication

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$



4

BOOLEAN OR GATE

- An OR gate is represented by addition

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 1$$

The diagram illustrates the Boolean OR gate through four cases. Each case shows a truth table entry, a logic symbol (a D-shaped gate with a curved input side), and a circuit diagram. The circuit diagrams use switches and a light bulb to represent the logic: a switch is closed for '1' and open for '0'. The light bulb glows (indicated by a red circle) when the output is '1'.

- Case 1:** $0 + 0 = 0$. Both switches are open, the light bulb is off.
- Case 2:** $0 + 1 = 1$. The top switch is open, the bottom switch is closed, the light bulb is on.
- Case 3:** $1 + 0 = 1$. The top switch is closed, the bottom switch is open, the light bulb is on.
- Case 4:** $1 + 1 = 1$. Both switches are closed, the light bulb is on.

5

BOOLEAN NOT GATE

- The NOT gate is represented as it's inversion

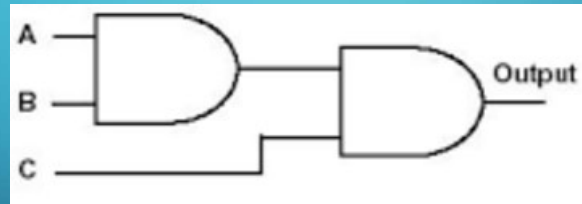
The diagram illustrates the Boolean NOT gate through two cases. Each case shows a truth table entry, a logic symbol (a triangle with a small circle at its tip), and a circuit diagram. The circuit diagrams use a switch and a light bulb: the light bulb glows when the output is '1' and is off when the output is '0'.

- Case 1:** If $A=0$, Then $\bar{A}=1$. The input switch is open, the output light bulb is on.
- Case 2:** If $A=1$, Then $\bar{A}=0$. The input switch is closed, the output light bulb is off.

6

REPRESENTING LOGIC

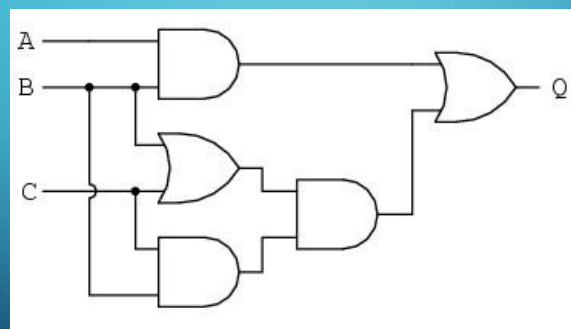
- Express the following with Boolean algebra



7

REPRESENTING LOGIC

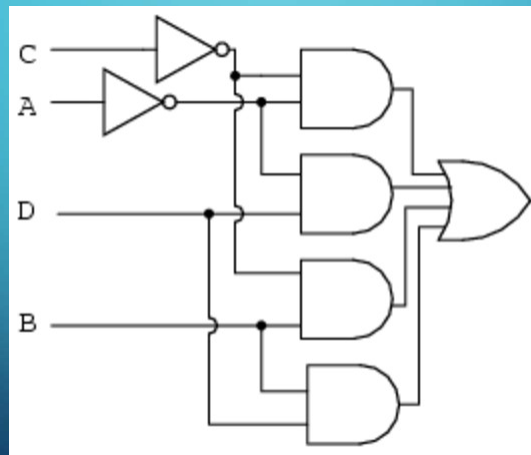
- Express the following with Boolean algebra



8

REPRESENTING LOGIC

- Express the following with Boolean algebra



9

10