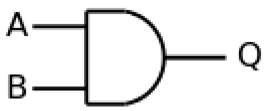
Boolean Algebra

Introduction

Boolean Algebra is a mathematical way of representing combinational logic circuits made from logic gates. The identities and theorems of Boolean Algebra allow complex logic circuits to be simplified.

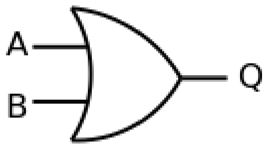
Logic Operations

The AND function is represented by a DOT like the product symbol sometimes used in mathematics



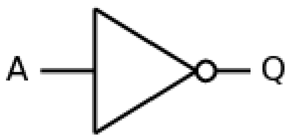
$$Q = A \, . \, B$$

The OR function is represented by a PLUS like the addition symbol used in mathematics



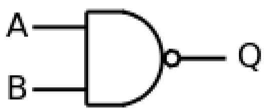
$$Q = A + B$$

The NOT function is represented by a BAR above the symbol



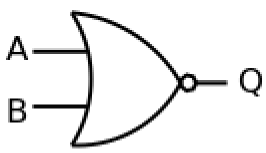
$$Q = \overline{A}$$

The NAND function is a combination of the DOT for the AND and a BAR for the NOT. The BAR is over the DOT and includes both inputs



$$Q = \overline{A \, . \, B}$$

The NOR function is a combination of the PLUS for the OR and a BAR for the NOT. The BAR is over the PLUS and includes both inputs



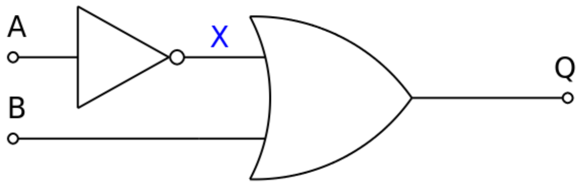
$$Q = \overline{A + B}$$

Using Boolean Algebra to describe logic circuits

Logic circuits can be described using Boolean algebra.

The aim is to express **Q** in terms of **A** and **B**

Example 1



$X = \overline{A}$

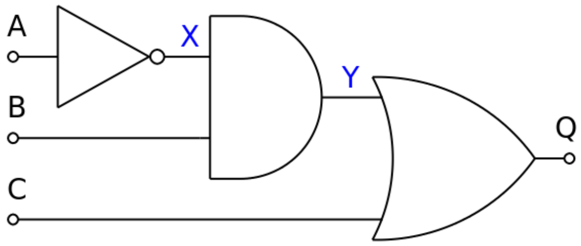
$Q = X + B$

Therefore:

$Q = \overline{A} + B$

This reads as Q equals (NOT A) OR B

Example 2



$X = \overline{A}$

$Y = X \,.\, B$

$Y = \overline{A} \,.\, B$

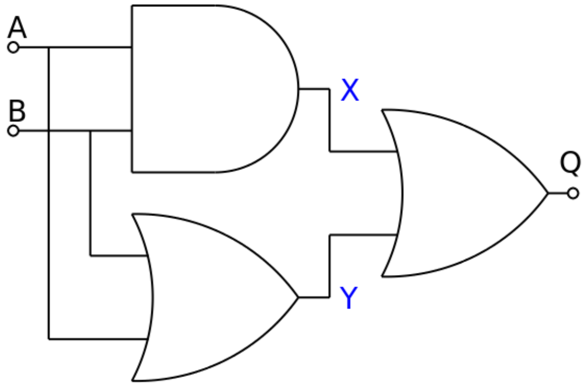
$Q = Y + C$

Therefore

$Q = (\overline{A} \,.\, B) + C$

This reads as Q equals ((NOT A) AND B) OR C

Example 3



$X = A . B$

$Y = A + B$

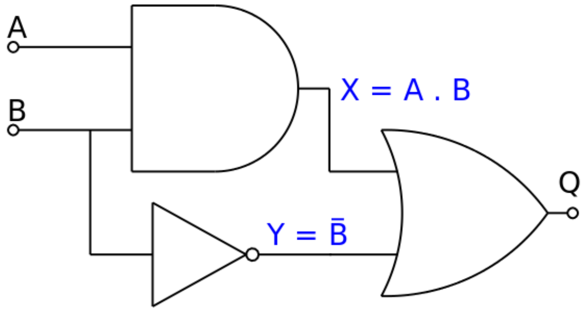
$Q = X + Y$

Therefore

$Q = (A . B) + (A + B)$

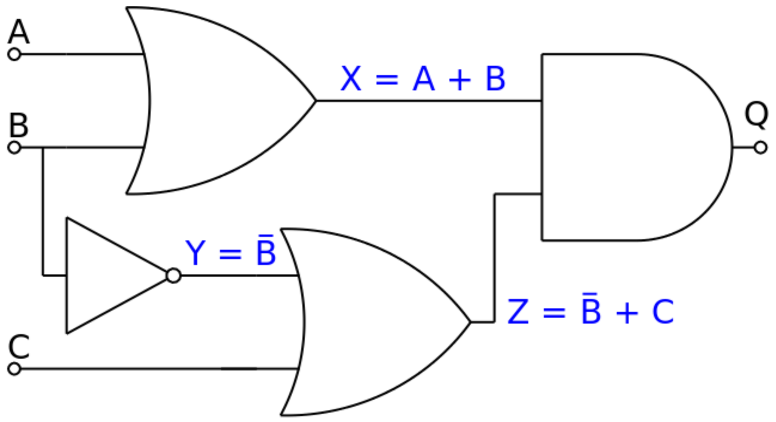
This reads as Q equals (A AND B) OR (A OR B)

Example 4



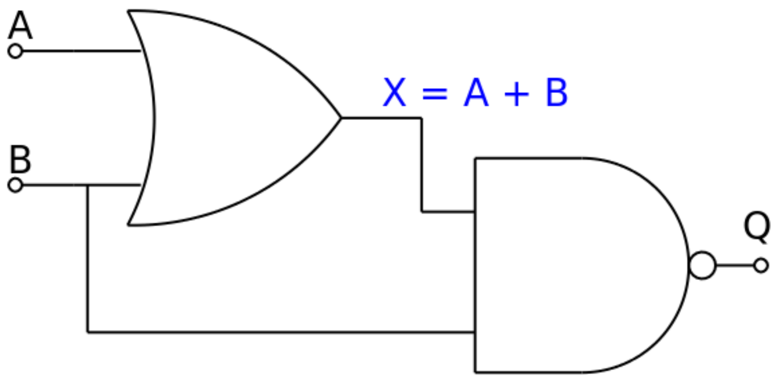
$Q = (A . B) + \overline{B}$

Example 5



$Q = (A + B) . (\overline{B} + C)$

Example 6



$Q = \overline{(A + B) . B}$

Using Boolean Algebra to describe truth tables

Truth tables can be represented using Boolean algebra.

For each row where Q = 1 there is a Boolean expression.

If more that one row has Q = 1 then the individual Boolean expressions are combined with the Logic OR function.

Example 1

A	B	Q
0	0	0
0	1	1
1	0	0
1	1	0

There is only one row where $Q = 1$ which is when $A = 0$ and $B = 1$

Alternatively, there is only one row where $Q = 1$ which is where $\overline{A} = 1$ and $B = 1$

This can be expressed as $Q = 1$ when $\overline{A} = 1$ AND $B = 1$ which is written as $Q = \overline{A}.\ B$

$$Q = \overline{A}.\ B$$

Example 2

A	B	Q
0	0	1
0	1	0
1	0	1
1	1	0

There are two rows where $Q = 1$ which are when $A = 0$ and $B = 0$ or when $A = 1$ and $B = 0$

Alternatively, there are two rows when $Q = 1$ which are when $\overline{A} = 1$ and $\overline{B} = 1$ or when $A = 1$ and $\overline{B} = 1$

This can be expressed as $Q = 1$ when ($\overline{A} = 1$ AND $\overline{B} = 1$) OR ($A = 1$ AND $\overline{B} = 1$) which is written as $Q = (\overline{A}\,.\,\overline{B}) + (A\,.\,\overline{B})$

$$Q = (\overline{A}\,.\,\overline{B}) + (A\,.\,\overline{B})$$

Example 3

A	B	Q
0	0	1
0	1	1
1	0	0
1	1	0

There are two rows where $Q = 1$ which are when $A = 0$ and $B = 0$ or when $A = 0$ and $B = 1$

Alternatively, there are two rows when $Q = 1$ which are when $\overline{A} = 1$ and $\overline{B} = 1$ or when $\overline{A} = 1$ and $B = 1$

This can be expressed as $Q = 1$ when ($\overline{A} = 1$ AND $\overline{B} = 1$) OR ($\overline{A} = 1$ AND $B = 1$) which is written as $Q = (\overline{A} . \overline{B}) + (\overline{A} . B)$

$$Q = (\overline{A} . \overline{B}) + (\overline{A} . B)$$

Example 4

A	B	Q
0	0	1
0	1	1
1	0	0
1	1	1

By considering each row where $Q = 1$ the truth table can be represented as:

$$Q = (\overline{A} . \overline{B}) + (\overline{A} . B) + (A . B)$$

Example 5

A	B	C	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

The same principle applies when there are three inputs (or more).

Consider each row where $Q = 1$ and combine the Boolean expressions with Logical OR functions (+)

By considering each row where $Q = 1$ the truth table can be represented as:

$$Q = (\overline{A} . B . C) + (A . \overline{B} . C) + (A . B . \overline{C})$$

Logic AND Identities

$A \cdot 0 = 0$

$A \cdot 1 = A$

$A \cdot A = A$

$A \cdot \overline{A} = 0$

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Each of the above identities can be shown to be correct by considering the truth table for the AND function.

In the identities **A is a variable** and can be either Logic 0 or Logic 1

Consider the case when A = 0

The four identities can now be written as:

0 . 0 = 0 which reads as "zero AND zero equals zero"

0 . 1 = 0

0 . 0 = 0

0 . 1 = 0

all of which agree with the truth table.

Consider the case when A = 1

The four identities can now be written as:

1 . 0 = 0 which reads as "one AND zero equals zero"

1 . 1 = 1

1 . 1 = 1

1 . 0 = 0

all of which agree with the truth table.

Logic OR Identities

$A + 0 = A$

$A + 1 = 1$

$A + A = A$

$A + \overline{A} = 1$

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Each of the following identities can be shown to be correct by considering the truth table for the OR function.

In the identities **A is a variable** and can be either Logic 0 or Logic 1

Consider the case when A = 0

The four identities can now be written as:

0 + 0 = 0 which reads as "zero OR zero equals zero"

0 + 1 = 1

0 + 0 = 0

0 + 1 = 1

all of which agree with the truth table.

Consider the case when A = 1

The four identities can now be written as:

1 + 0 = 1 which reads as "one OR zero equals one"

1 + 1 = 1

1 + 1 = 1

1 + 0 = 1

all of which agree with the truth table.

Logic NOT Identity

There is only one NOT identity.

A NOT followed by another NOT has no effect. Therefore we can state NOT (NOT A) = A

Or, alternatively $NOT\ (\overline{A}) = A$

The symbol is a double bar above the character. Therefore:

$$\overline{\overline{A}} = A$$

DeMorgan's Theorem

DeMorgan's Theorem(s) are particularly important because they relate the (N)OR function and the (N)AND function and allow one to be equated to the other.

$$\overline{A} . \overline{B} = \overline{A + B}$$

This reads as (NOT A) **AND** (NOT B) is the same as A **NOR** B.

It is easy to prove this relationship with a truth table.

$$\overline{A} + \overline{B} = \overline{A . B}$$

This reads as (NOT A) **OR** (NOT B) is the same as A **NAND** B.

Again, this can be verified with a truth table.

Website

https://www.electronicsteaching.com/Electronics_Resources/DocumentIndex.html