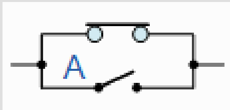
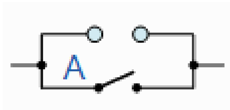
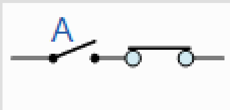
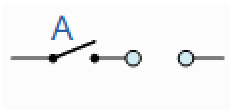
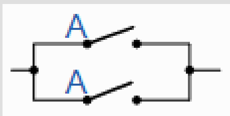
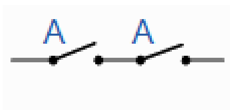
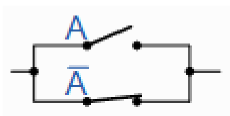
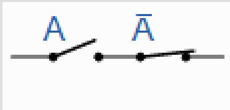
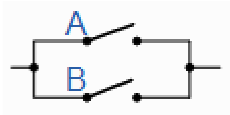
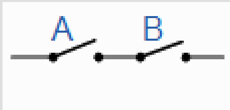
Laws and Theorems of Boolean Algebra

Identity laws (operations with 0 and 1):	
1. X + 0 = X	1D. X • 1 = X
Annulment laws (operations with 0 and 1):	
2. X + 1 = 1	2D. X • 0 = 0
Idempotent laws:	
3. X + X = X	3D. X • X = X
Involution law:	
4. (X') ' = X	
Laws of complementarity:	
5. X + X' = 1	5D. X • X' = 0
Commutative laws:	
6. X + Y = Y + X	6D. X • Y = Y • X
Associative laws:	
7. (X + Y) + Z = X + (Y + Z) = X + Y + Z	7D. (XY)Z = X(YZ) = XYZ
Distributive laws:	
8. X(Y + Z) = XY + XZ	8D. X + YZ = (X + Y) (X + Z)
Simplification theorems:	
9. X Y + X Y' = X	9D. (X + Y) (X + Y') = X
10. X + XY = X	10D. X (X + Y) = X
11. (X + Y') Y = XY	11D. XY' + Y = X + Y
DeMorgan's laws:	
12. (X + Y + Z + ...)' = X'Y'Z'...	12D. (X Y Z ...)' = X' + Y' + Z' + ...
13. [f (X1, X2, ... XN, 0, 1, +, •)]' = f (X1', X2', ... XN', 1, 0, •, +)	
Duality:	
14. $(X + Y + Z + \dots)^D$ = X Y Z ...	14D. $(X Y Z\dots)^D$ = X + Y + Z + ...
15. $[f (X1, X2, \dots XN, 0, 1, +, \bullet)]^D$ = f (X1, X2, ... XN, 1, 0, •, +)	
Theorem for multiplying out and factoring:	
16. (X + Y) (X' + Z) = X Z + X' Y	16D. XY + X' Z = (X + Z) (X' + Y)
Consensus theorem:	
17. XY + YZ + X'Z = XY + X'Z	17D. (X + Y)(Y + Z)(X' + Z) = (X + Y)(X' + Z)

Truth Tables for the Laws of Boolean

Boolean Expression	Description	Equivalent Switching Circuit	Boolean Algebra Law or Rule
A + 1 = 1	A in parallel with closed = "CLOSED"		Annulment
A + 0 = A	A in parallel with open = "A"		Identity
A . 1 = A	A in series with closed = "A"		Identity
A . 0 = 0	A in series with open = "OPEN"		Annulment
A + A = A	A in parallel with A = "A"		Idempotent
A . A = A	A in series with A = "A"		Idempotent
NOT $\bar{A}$ = A	NOT NOT A (double negative) = "A"		Double Negation
A + $\bar{A}$ = 1	A in parallel with NOT A = "CLOSED"		Complement

$A \cdot \bar{A} = 0$	A in series with NOT A = "OPEN"		Complement
$A+B = B+A$	A in parallel with B = B in parallel with A		Commutative
$A.B = B.A$	A in series with B = B in series with A		Commutative
$\overline{A+B} = \bar{A}.\ \bar{B}$	invert and replace OR with AND		de Morgan's Theorem
$\overline{A.B} = \bar{A}+\bar{B}$	invert and replace AND with OR		de Morgan's Theorem

The basic **Laws of Boolean Algebra** that relate to

- The **Commutative Law** allowing a change in position for addition and multiplication.
- The **Associative Law** allowing the removal of brackets for addition and multiplication.
- The **Distributive Law** allowing the factoring of an expression are the same as in ordinary algebra.
- Each of the **Boolean Laws** above are given with just a single or two variables, but the number of variables defined by a single law is not limited to this as there can be an infinite number of variables as inputs too the expression.
- These Boolean laws detailed above can be used to prove any given Boolean expression as well as for simplifying complicated digital circuits.

A brief description of the various **Laws of Boolean** are given below with A representing a variable input.

Description of the Laws of Boolean Algebra

- Annulment Law – A term AND´ed with a “0” equals 0 or OR´ed with a “1” will equal 1.
  - $A . 0 = 0$ A variable AND’ed with 0 is always equal to 0.
  - $A + 1 = 1$ A variable OR’ed with 1 is always equal to 1.
- Identity Law – A term OR´ed with a “0” or AND´ed with a “1” will always equal that term.
  - $A + 0 = A$ A variable OR’ed with 0 is always equal to the variable.
  - $A . 1 = A$ A variable AND’ed with 1 is always equal to the variable.
- Idempotent Law – An input that is AND´ed or OR´ed with itself is equal to that input.
  - $A + A = A$ A variable OR’ed with itself is always equal to the variable.
  - $A . A = A$ A variable AND’ed with itself is always equal to the variable.
- Complement Law – A term AND´ed with its complement equals “0” and a term OR´ed with its complement equals “1”.
  - $A . \overline{A} = 0$ A variable AND’ed with its complement is always equal to 0.
  - $A + \overline{A} = 1$ A variable OR’ed with its complement is always equal to 1.
- Commutative Law – The order of application of two separate terms is not important.
  - $A . B = B . A$ The order in which two variables are AND’ed makes no difference.
  - $A + B = B + A$ The order in which two variables are OR’ed makes no difference.
- Double Negation Law – A term that is inverted twice is equal to the original term.
  - $\overline{\overline{A}} = A$ A double complement of a variable is always equal to the variable.
- de Morgan´s Theorem – There are two “de Morgan´s” rules or theorems,
- (**1**) Two separate terms NOR´ed together is the same as the two terms inverted (Complement) and AND´ed for example, A+B = A. B.
- (**2**) Two separate terms NAND´ed together is the same as the two terms inverted (Complement) and OR´ed for example, A.B = A +B.

Other algebraic Laws of Boolean not detailed above include:

- Distributive Law – This law permits the multiplying or factoring out of an expression.
    - A(B + C) = A.B + A.C   (OR Distributive Law)
    - A + (B.C) = (A + B).(A + C)   (AND Distributive Law)
- Absorptive Law – This law enables a reduction in a complicated expression to a simpler one by absorbing like terms.
    - A + (A.B) = A   (OR Absorption Law)
    - A(A + B) = A   (AND Absorption Law)
- Associative Law – This law allows the removal of brackets from an expression and regrouping of the variables.
    - A + (B + C) = (A + B) + C = A + B + C   (OR Associate Law)
    - A(B.C) = (A.B)C = A . B . C   (AND Associate Law)

Boolean Algebra Functions

Using the information above, simple 2-input AND, OR and NOT Gates can be represented by 16 possible functions as shown in the following table.

Function	Description	Expression
1.	NULL	0
2.	IDENTITY	1
3.	Input A	A
4.	Input B	B
5.	NOT A	$\overline{A}$
6.	NOT B	$\overline{B}$
7.	A AND B (AND)	A . B
8.	A AND NOT B	$A . \overline{B}$
9.	NOT A AND B	$\overline{A} . B$
10.	NOT AND (NAND)	$\overline{A.B}$
11.	A OR B (OR)	A + B
12.	A OR NOT B	$A + \overline{B}$
13.	NOT A OR B	$\overline{A} + B$
14.	NOT OR (NOR)	$\overline{A+B}$

15.	Exclusive-OR	A . B + A . B
16.	Exclusive-NOR	$\overline{A.B+A.B}$

Laws of Boolean Algebra Example No1

Using the above laws, simplify the following expression: (A + B) . (A + C)

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

A.A + A.C + A.B + B.C – Distributive law

A + A.C + A.B + B.C – Idempotent AND law (A.A = A)

A(1 + C) + A.B + B.C – Distributive law

A.1 + A.B + B.C – Identity OR law (1 + C = 1)

A(1 + B) + B.C – Distributive law

A.1 + B.C – Identity OR law (1 + B = 1)

Q = A + (B.C) – Identity AND law (A.1 = A)

Then the expression: (A + B) (A + C) can be simplified to A + (B.C) as in the Distributive law.