Boolean Algebra

Abhijeet

August 29, 2025

Table of Contents

What is Boolean Algebra?

- A branch of algebra in which the values of the variables are the truth values *true* and *false*, usually denoted 1 and 0 respectively.
- It is used to analyze and simplify digital circuits.
- It is also called Binary Algebra or logical Algebra.
- George Boole, an English mathematician, developed this algebra in 1854.

Basic Operations

The basic operations in Boolean algebra are:

- **Conjunction** ($\wedge$): Corresponds to the AND operator.
- **Disjunction** ($\vee$): Corresponds to the OR operator.
- **Negation** ($\neg$): Corresponds to the NOT operator.

AND Operator

The AND operator is denoted by a dot ($\cdot$) or by the absence of an operator.

- $A \cdot B = Y$ or $AB = Y$
- The expression is true (1) only if both A and B are true (1).

A	**B**	**A AND B**
0	0	0
0	1	0
1	0	0
1	1	1

Table: Truth Table for AND

OR Operator

The OR operator is denoted by a plus sign $(+)$.

- $A + B = Y$
- The expression is true (1) if either A or B or both are true (1).

A	**B**	**A OR B**
0	0	0
0	1	1
1	0	1
1	1	1

Table: Truth Table for OR

NOT Operator

The NOT operator is denoted by a prime (A').

- $A' = Y$
- It inverts the input. If the input is true (1), the output is false (0).

A	**NOT A**
0	1
1	0

Table: Truth Table for NOT

Basic Laws

Commutative Law

- $A + B = B + A$
- $A \cdot B = B \cdot A$

Associative Law

- $(A + B) + C = A + (B + C)$
- $(A \cdot B) \cdot C = A \cdot (B \cdot C)$

Distributive Law

- $A \cdot (B + C) = A \cdot B + A \cdot C$
- $A + (B \cdot C) = (A + B) \cdot (A + C)$

Other Important Laws in Boolean Algebra

- **Identity Law**: $A + 0 = A$, $A \cdot 1 = A$
- **Annulment Law**: $A + 1 = 1$, $A \cdot 0 = 0$
- **Idempotent Law**: $A + A = A$, $A \cdot A = A$
- **Complement Law**: $A + A' = 1$, $A \cdot A' = 0$
- **Involution Law**: $(A')' = A$
- **Absorption Law**: $A + (A \cdot B) = A$, $A \cdot (A + B) = A$
- **De Morgan's Laws**:
  - $(A + B)' = A' \cdot B'$
  - $(A \cdot B)' = A' + B'$

Minterms

- A minterm is a product term (AND operation) that contains all variables of the function, either in their normal or complemented form.
- For a given row of a truth table, the minterm is 1 if and only if the input combination is the one for that row.
- A variable appears in its normal form (X) if its value is 1 in the input combination.
- A variable appears in its complemented form (X') if its value is 0 in the input combination.
- Example for two variables A and B:
  - $A'B'$ (for A=0, B=0)
  - $A'B$ (for A=0, B=1)
  - AB' (for A=1, B=0)
  - AB (for A=1, B=1)

Maxterms

- A maxterm is a sum term (OR operation) that contains all variables of the function, either in their normal or complemented form.
- For a given row of a truth table, the maxterm is 0 if and only if the input combination is the one for that row.
- A variable appears in its normal form (X) if its value is 0 in the input combination.
- A variable appears in its complemented form (X') if its value is 1 in the input combination.
- Example for two variables A and B:
  - $A + B$ (for A=0, B=0)
  - $A + B'$ (for A=0, B=1)
  - $A' + B$ (for A=1, B=0)
  - $A' + B'$ (for A=1, B=1)

Sum of Products (SOP)

- A Boolean expression can be represented as a sum of minterms.
- This form is called the Sum of Products (SOP) or Disjunctive Normal Form (DNF).
- To get the SOP form from a truth table, we sum (OR) all the minterms for which the function's output is 1.

Product of Sums (POS)

- A Boolean expression can also be represented as a product of maxterms.
- This form is called the Product of Sums (POS) or Conjunctive Normal Form (CNF).
- To get the POS form from a truth table, we multiply (AND) all the maxterms for which the function's output is 0.

Example: Generating Compound Statements

Let's consider the compound statement "If A, then B", which is denoted as $A \to B$. The truth table is as follows:

A	**B**	**A $\to$ B**
0	0	1
0	1	1
1	0	0
1	1	1

Table: Truth Table for $A \to B$

Example: SOP and POS from Truth Table

From the truth table for $A \to B$:

- **Sum of Products (SOP):** We look for rows where the output is 1.
  - Row 1 (A=0, B=0): Minterm is $A'B'$
  - Row 2 (A=0, B=1): Minterm is $A'B$
  - Row 4 (A=1, B=1): Minterm is AB

  The SOP expression is $Y = A'B' + A'B + AB$. This can be simplified to $A' + B$.
- **Product of Sums (POS):** We look for rows where the output is 0.
  - Row 3 (A=1, B=0): Maxterm is $A' + B$

  The POS expression is $Y = A' + B$.

In this case, the simplified SOP and the POS are the same, which is often the case. The expression for "If A, then B" is logically equivalent to "Not A or B".

Derivation of the Simplification

Let's derive the simplification of the expression $Y = A'B' + A'B + AB$.

$$
\begin{aligned}
Y &= A'B' + A'B + AB \\
&= A'(B' + B) + AB && \text{Distributive Law} \\
&= A'(1) + AB && \text{Complement Law: } B + B' = 1 \\
&= A' + AB && \text{Identity Law: } A \cdot 1 = A \\
&= (A' + A)(A' + B) && \text{Distributive Law} \\
&= (1)(A' + B) && \text{Complement Law: } A + A' = 1 \\
&= A' + B && \text{Identity Law: } 1 \cdot A = A
\end{aligned}
$$

This shows that the expression $A'B' + A'B + AB$ simplifies to $A' + B$. This is a common simplification that is useful to remember, and it corresponds to the logical statement for implication, $A \to B \equiv \neg A \lor B$.

For statements P and Q, show that P implies $(P \vee Q)$ is a tautology (1)

We can write the above expression as:

$$P \Rightarrow (P + Q)$$

We know that

$$A \Rightarrow B \equiv A' + B$$

(that is, not A or B).
Hence,

$$P \Rightarrow (P + Q) \equiv P' + (P + Q)$$

Using the associative law:

$$P' + (P + Q) = (P' + P) + Q$$

For statements P and Q, show that P implies $(P \vee Q)$ is a tautology (2)

By the annulment law:

$$(P' + P) + Q = 1 + Q$$

Again, by the annulment law:

$$1 + Q = 1$$

Thus, $P \Rightarrow (P + Q)$ is always true, so it is a tautology.

For statements P and Q, show that $(P \wedge \neg Q) \wedge (P \wedge Q)$ is a contradiction

Start with the Boolean expression:

$$(P \cdot Q') \cdot (P \cdot Q)$$

Rearrange terms:

$$(P \cdot Q') \cdot (Q \cdot P) \quad \text{(Commutative Law)}$$

Group terms:

$$P \cdot (Q' \cdot Q) \cdot P \quad \text{(Associative Law)}$$

Simplify complements:

$$P \cdot 0 \cdot P = 0 \quad (Q' \cdot Q = 0, \text{ Complement Law})$$

Hence, $(P \cdot Q') \cdot (P \cdot Q)$ is always false, so it is a contradiction.

Show that $(P \to \neg Q) \wedge (P \wedge Q)$ is a contradiction

Convert the implication using the formula $A \Rightarrow B \equiv A' + B$:

$$P \Rightarrow \neg Q \equiv P' + Q'$$

So the statement becomes:

$$(P' + Q') \cdot (P \cdot Q)$$

Apply distributive law:

$$(P' + Q') \cdot (P \cdot Q) = P \cdot Q \cdot P' + P \cdot Q \cdot Q' \quad \text{(Distributive Law)}$$

Simplify using complement law:

$$P \cdot Q \cdot P' + P \cdot Q \cdot Q' = 0 + 0 = 0 \quad (\text{Complement Law, } P \cdot P' = 0,\ Q \cdot Q' = 0)$$

Hence, $(P \Rightarrow \neg Q) \cdot (P \cdot Q)$ is always false, so it is a contradiction.

For statements P, Q, and R, show that $((P \Rightarrow Q) \wedge (Q \Rightarrow R)) \Rightarrow (P \Rightarrow R)$ is a tautology (1)

Turn into Boolean expression using $A \Rightarrow B = A' + B$:

$$(P' + Q)(Q' + R) \Rightarrow (P' + R)$$

Apply implication law again:

$$((P' + Q)(Q' + R))' + (P' + R)$$

Use De Morgan's Law:

$$(P' + Q)' + (Q' + R)' + P' + R = PQ' + QR' + P' + R$$

For statements P, Q, and R, show that $((P \Rightarrow Q) \wedge (Q \Rightarrow R)) \Rightarrow (P \Rightarrow R)$ is a tautology (2)

Regroup terms:

$$(PQ' + P') + (QR' + R)$$

Use distributive law:

$$(P' + P)(P' + Q') + (R + Q)(R + R')$$

Use Complement Law($P' + P = 1$) and Annulment Law($1 + A = 1$) and simplify:

$$(P' + Q') + (Q' + R) = P' + Q' + Q + R$$

Use Complement Law on $Q + Q' = 1$:

$$P' + R + 1 = 1 + (P' + R)$$

Finally, use Annulment Law:

$$1$$

Hence, $((P \Rightarrow Q) \wedge (Q \Rightarrow R)) \Rightarrow (P \Rightarrow R)$ is always true, so it is a tautology.

Let P and Q be statements. Show that $[(P \vee Q) \wedge \neg(P \wedge Q)] \equiv \neg(P \Leftrightarrow Q)(1)$

Let us Construct truth table for $\neg(P \Leftrightarrow Q)$ to derive the logical equivalence of the same

P	Q	$P \Leftrightarrow Q$	$\neg(P \Leftrightarrow Q)$
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	0

$\neg(P \Leftrightarrow Q)$ is TRUE when $(P, Q) = (0, 1), (1, 0)$.

Let P and Q be statements. Show that $[(P \vee Q) \wedge \neg(P \wedge Q)] \equiv \neg(P \Leftrightarrow Q)(2)$

From the truth table, the minterms corresponding to TRUE outputs of $\neg(P \Leftrightarrow Q)$ are:

$$P'Q \quad \text{and} \quad PQ'$$

Hence, the sum-of-products (SOP) form:

$$\neg(P \Leftrightarrow Q) = P'Q + PQ'$$

This is the RHS logical equivalent of the negation of the biconditional.

Let P and Q be statements. Show that $[(P \vee Q) \wedge \neg(P \wedge Q)] \equiv \neg(P \Leftrightarrow Q)(3)$

LHS statement:

$$(P \vee Q) \wedge \neg(P \wedge Q) = (P + Q)(P \wedge Q)'$$

Apply De Morgan's law:

$$(P + Q)(P' + Q')$$

Use distributive law:

$$P(P' + Q') + Q(P' + Q') = PP' + PQ' + QP' + QQ'$$

By complement law: $PP' = 0$, $QQ' = 0$, so we have:

$$PQ' + P'Q$$

Hence, LHS = RHS, proving the logical equivalence.