Principles Of

Digital Design

Boolean Algebra

Everything about Boolean Algebra

- Axioms
- Theorems
- Canonical Forms
- Standard Forms
- Expression Optimization

Basic Algebraic Properties

- A **set** is a collection of objects with a common property
  - If S is a set and x is a member of the set S, then $x \in S$
    - *A = {1, 2, 3, 4} denotes the set A, whose elements are 1, 2, 3, 4*
- A **binary operator** on a set S is a rule that assigns to each pair of elements in S another element that is in S
- **Axioms** are assumption that are valid without proof

Examples of Axioms

- **Closure**
  - A set S is closed with respect to a binary operator $\bullet$, iff, for all $x, y \in S$,

$$(x \bullet y) \in S$$

    - $Z^+ = \{1, 2, 3, \ldots\}$ is closed to addition, all positive numbers are in Z^+

- **Associativity**
  - A binary operator $\bullet$ defined on a set S is associative, iff, for all $x, y, z \in S$

$$(x \bullet y) \bullet z = x \bullet (y \bullet z)$$

- **Identity Element**
  - A set S has an identity element e , *iff*, for every $x \in S$

$$e \bullet x = x \bullet e = x$$

$$\textit{(for example, } x + 0 = 0 + x = x)$$

Examples of Axioms

- **Commutativity**
  - A binary operator • is commutative, iff, for all $x, y \in S$

$$x \bullet y = y \bullet x$$

- **Inverse Element**
  - A set S has an inverse, iff, for every $x \in S$, there exists an element $y \in S$ such that

$$x \bullet y = e$$

- **Distributivity**
  - If • and □ are two binary operators on a set S, • is said to be distributive over □ if, for all $x, y, z \in S$

$$x \bullet (y \,\square\, z) = (x \bullet y) \,\square\, (x \bullet z)$$

Axiomatic Definition of Boolean Algebra

Boolean algebra is a set of elements B with two binary operators, + and ·, which satisfies the following six axioms:

- **Axiom 1 (Closure): (a) B is closed with respect to the operator +; (b) B is also closed with respect to the operator ·**
- **Axiom 2 (Identity): (a) B has an identity element with respect to +, designated by 0; (b) B also has an identity element with respect · , designated by 1**
- **Axiom 3 (Commutativity): (a) B is commutative with respect to +; (b) B is also commutative with respect to ·**
- **Axiom 4 (Distributivity): (a) The operator · is distributive over +; (b) similarly, the operator + is distributive over ·**
- **Axiom 5 (Complement Element): For every $x \in B$, there exists an element $x' \in B$ such that (a) $x + x' = 1$ and (b) $x \cdot x' = 0$**
  This second element x', is called the complement of x
- **Axiom 6 (Cardinality): There are at least two elements $x, y \in B$ such that $x \neq y$**

Axiomatic Definition of Boolean Algebra

Differences between Boolean algebra and ordinary algebra

- In ordinary algebra, + is not distributive over ·
- Boolean algebra does not have inverses with respect to + and · ; therefore, there are no subtraction or division operations in Boolean algebra
- Complements are available in Boolean algebra, but not in ordinary algebra
- Boolean algebra applies to a finite set of elements, whereas ordinary algebra would apply to the infinite sets of real numbers
- The definition above for Boolean algebra does not include associativity, since it can be derived from the other axioms

Two-valued Boolean Algebra (1)

- Set *B* has two elements: 0 and 1

- Algebra has two operators: AND and OR

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

AND Operator

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

OR Operator

Two-valued Boolean Algebra (2)

Two-valued Boolean algebra satisfies Huntington axioms

- **Axiom 1 (Closure Property): Closure is evident in the** AND/OR **tables, since the result of each operation is an element of** *B.*
- **Axiom 2 (Identity Element): The identity elements in this algebra are** 0 **for the operator + and** 1 **for the operator ·. From the** AND/OR **tables, we see that:**
  - **0 + 0 = 0, and 0 + 1 = 1 + 0 = 1**
  - **1 · 1 = 1, and 1 · 0 = 0 · 1 = 0**
- **Axiom 3 (Commutativity Property): The commutativity laws follow from the symmetry of the AND and OR operators in their operator tables.**

Two-valued Boolean Algebra (3)

- Axiom 4 (Distributivity): The distributivity of this algebra can be demonstrated by checking both sides of the equation.

$$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$$

x	y	z	$y + z$	$x \cdot (y + z)$	xy	xz	$(xy) + (xz)$
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

Proof of distributivity of ·

$$x + (y \cdot z) = (x + y)(x + z)$$

x	y	z	yz	$x + (yz)$	$x + y$	$x + z$	$(x + y)(x + z)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

Proof of distributivity of +

Two-valued Boolean Algebra (4)

- **Axiom 5 (Complement): 0 and 1 are complements of each other, since $0 + 0' = 0 + 1 = 1$ and $1 + 1' = 1 + 0 = 1$; furthermore, $0 \cdot 0' = 0 \cdot 1 = 0$ and $1 \cdot 1' = 1 \cdot 0 = 0$.**

x	x'
0	1
1	0

NOT Operator

- **Axiom 6 (Cardinality): The cardinality axiom is satisfied, since this two-valued Boolean algebra has two distinct elements, 1 and 0, and $1 \neq 0$.**

Boolean Operator Procedure

- **Boolean operators are applied in the following order.**

  - **Parentheses** ( )
  - **NOT** ′
  - **AND** ·
  - **OR** +

Example: Evaluate expression $(x + xy)'$ for $x = 1$ and $y = 0$:

$$(1 + 1 \cdot 0)' = (1 + 0)' = (1)' = 0$$

Duality Principle

- Any algebraic expression derived from axioms stays valid when
  - OR and AND
  - 0 and 1

  are interchanged.

Example:

If

$$X + 1 = 1$$

then

$$X \cdot 0 = 0$$

by the duality principle

Theorems of Boolean Algebra

Theorem 1 (Idempotency)	**(a)**	$x + x$	=	x
	(b)	xx	=	x
Theorem 2	**(a)**	$x + 1$	=	1
	(b)	$x \cdot 0$	=	0
Theorem 3 (Absorption)	**(a)**	$yx + x$	=	x
	(b)	$(y + x)x$	=	x
Theorem 4 (Involution)		$(x')'$	=	x
Theorem 5 (Associativity)	**(a)**	$(x + y) + z$	=	$x + (y + z)$
	(b)	$x(yz)$	=	$(xy)z$
Theorem 6 (De Morgan'sLaw)	**(a)**	$(x + y)'$	=	$x'y'$
	(b)	$(xy)'$	=	$x' + y'$

Basic Theorems of Boolean Algebra

Theorem Proofs in Boolean Algebra

- **Checking theorems for every combinations of variable value**

Example:

Theorem 6(a) DeMorgan's Law: $(x + y)' = x'y'$

x	y	$x + y$	$(x + y)'$	x'	y'	$x'y'$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Proof of Demorgan's First Theorem

Boolean Functions (1)

- **Algebraic expression with binary variables and Boolean operators AND, OR and NOT.**

Example:

$$F_1 = xy + xy'z + x'yz$$

This function would be equal to 1

if $x = 1$ and $y = 1$, or
if $x = 1$ and $y = 0$ and $z = 1$, or
if $x = 0$ and $y = 1$ and $z = 1$;

otherwise, $F_1 = 0$.

Note 1: When we evaluate Boolean expressions, we must follow a specific order of operations, namely, (1) parentheses, (2) NOT, (3) AND, (4) OR.

Note 2: A primed or unprimed variable is usually called a literal.

Boolean Functions (2)

- **Truth tables which list the functional value for all combinations of variable values.**

Example:

$$F_1 = xy + xy'z + x'yz$$

Row Numbers	Variable Values x	y	z	Function Values F_1
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Boolean Functions (3)

- **Complement of function F is function F', where F' can be obtained by interchanging 0 and 1 in the truth table.**

Example:

$$F_1 = xy + xy'z + x'yz$$

Row Numbers	Variable Values x	y	z	Function Values F_1	F_1'
0	0	0	0	0	1
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	0

Boolean Functions (4)

- **Complement of function F is function F', which can be obtained by:**
  - **Repeatedly applying DeMorgan's theorems.**

Example:

F_1'	$=$	$(xy + xy'z + x'yz)'$	by definition of F
	$=$	$(xy)'(xy'z)'(x'yz)'$	by DeMorgan's Th.
	$=$	$(x'+y')(x'+y+z')(x+y'+z')$	by DeMorgan's Th.

  - **Duality Principle**

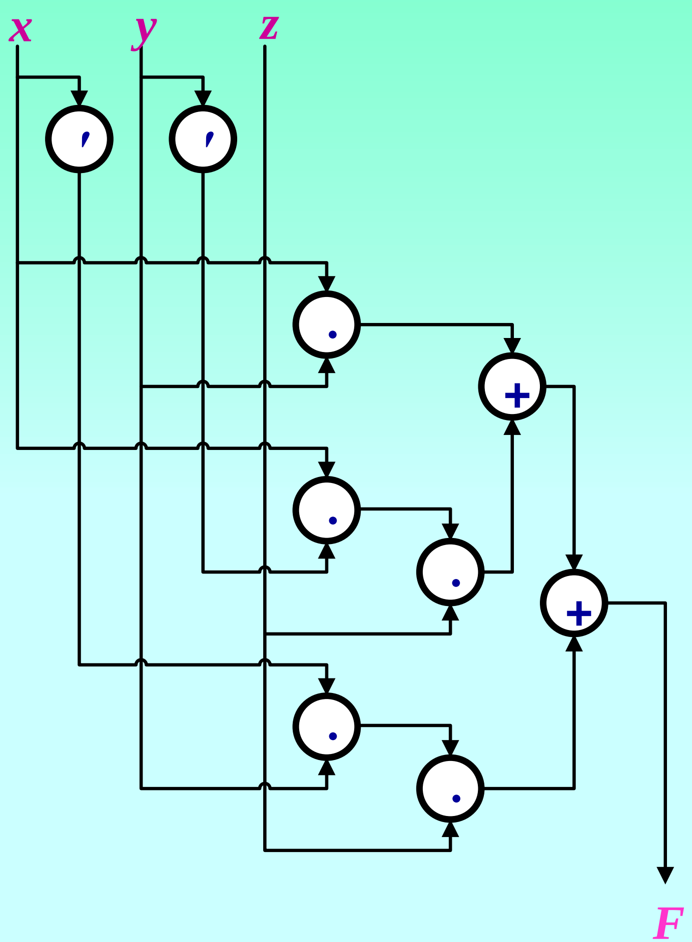
Example:

$$F_1 = (x \cdot y) + (x \cdot y' \cdot z) + (x' \cdot y \cdot z)$$

$$\downarrow$$

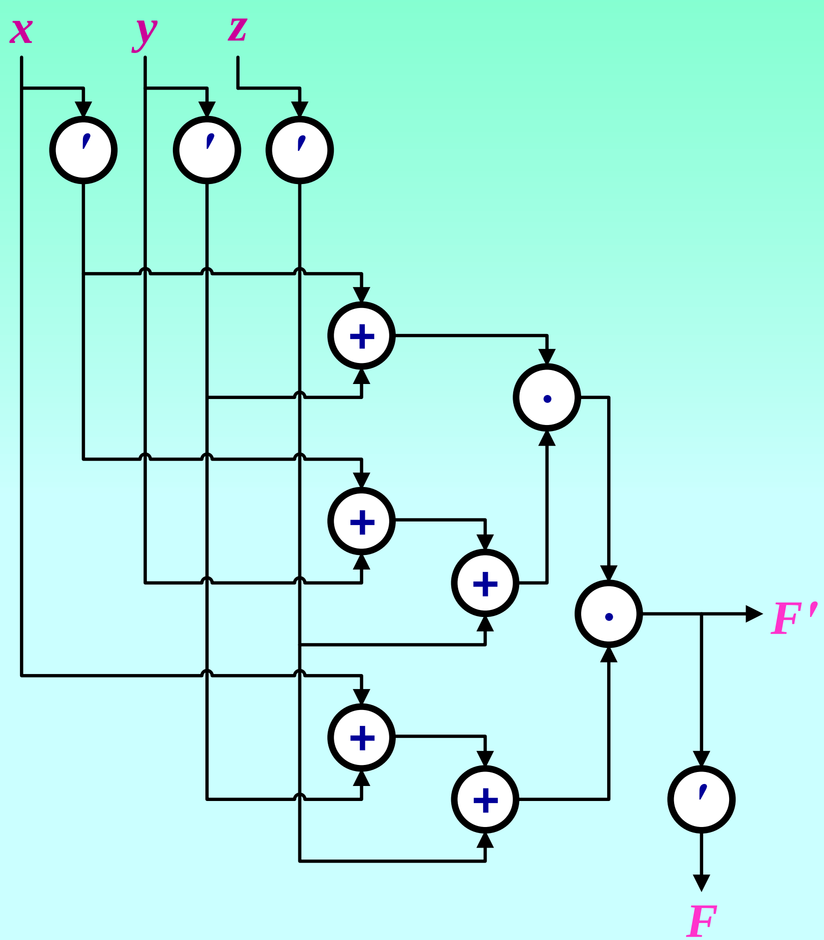
$$F_1' = (x'+y') \cdot (x'+y+z') \cdot (x+y'+z')$$

Graphic Representation of B Functions



$F_1 = (x \cdot y) + (x \cdot y' \cdot z) + (x' \cdot y \cdot z)$

F size = 5 ANDs + 2 ORs + 2 NOTs



$F_1' = (x' + y') \cdot (x' + y + z') \cdot (x + y' + z')$

F size = 2 ANDs + 5 ORs + 4 NOTs

Two different expressions have different sizes

Expression Equivalence

- We can prove expression equivalence by algebraic manipulation in which each transformation uses an axiom or a theorem of Boolean algebra.

Example:

$$F_1 = xy + xy'z + x'yz$$
$$= xy + xz + yz$$

Proof:

$xy + xy'z + x'yz$	$= xy + xyz + xy'z + x'yz$	by absorption
	$= xy + x(y + y')z + x'yz$	by distributivity
	$= xy + x1z + x'yz$	by complement
	$= xy + xz + x'yz$	by identity
	$= xy + xyz + xz + x'yz$	by absorption
	$= xy + xz + (x + x')yz$	by distributivity
	$= xy + xz + 1yz$	by complement
	$= xy + xz + yz$	by identity

$xy + xy'z + x'yz$	requires	5 ANDs	2 ORs	2 NOTs
$xy + xz + yz$	requires	3 ANDs	2 ORs	
Cost Difference:		2 ANDs	and	2 NOTs

Minterms (1)

- **Minterm definition**

A minterm of n variables $x_{n-1}, x_{n-2}, \ldots, x_0$, could be represented as:

$$m_i(x_{n-1}, x_{n-2}, \ldots, x_0) = y_{n-1} \ldots y_0$$

where $i = b_{n-1} \ldots b_0$ is a binary number and for all k, $0 \le k \le n-1$,

$$y_k = \begin{cases} x_k & \text{if } b_k = 1 \\ x_k' & \text{if } b_k = 0 \end{cases}$$

x	y	z	Minterms	Designation
0	0	0	$x'y'z'$	m_0
0	0	1	$x'y'z$	m_1
0	1	0	$x'yz'$	m_2
0	1	1	$x'yz$	m_3
1	0	0	$xy'z'$	m_4
1	0	1	$xy'z$	m_5
1	1	0	xyz'	m_6
1	1	1	xyz	m_7

Minterms for Three Binary Variables

Minterms (2)

- **Any Boolean function can be expressed as a sum (OR) of its 1-minterms:**

$$F(\text{list of variables}) = \Sigma(\text{list of 1-minterm indices})$$

Example:

Row Numbers	Variable Values			Function Values	
	x	y	z	F_1	F_1'
0	0	0	0	0	1
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	0

Truth Table

$$\begin{aligned} F_1(x, y, z) &= \Sigma(3, 5, 6, 7) \\ &= m_3 + m_5 + m_6 + m_7 \\ &= x'yz + xy'z + xyz' + xyz \end{aligned}$$

$$\begin{aligned} F_1'(x, y, z) &= \Sigma(0, 1, 2, 4) \\ &= m_0 + m_1 + m_2 + m_4 \\ &= x'y'z' + x'y'z + x'yz' + xy'z' \end{aligned}$$

Minterms (3)

- **Any Boolean function can be expanded into a sum-of-minterms form by expanding each term with $(x + x')$ for each missing variable x.**

Example:

$$
\begin{aligned}
F &= x + yz \\
&= x(y + y')(z + z') + (x + x')yz \\
&= xyz + xy'z + xyz' + xy'z' + xyz + x'yz \\
&= xyz + xy'z + xyz' + xy'z' + x'yz \\
&= m_7 + m_5 + m_6 + m_4 + m_3 \\
&= \Sigma(3, 4, 5, 6, 7)
\end{aligned}
$$

Minterms (4)

- **Each Boolean function can be converted into a sum-of-minterms form by generating the truth table and identifying 1-minterms.**

Example: $F = x + yz$

x	y	z	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$$F = m_3 + m_4 + m_5 + m_6 + m_7$$

Maxterms (1)

- **Maxterms can be defined as the complement of minterms:**

$$M_i = m_i' \text{ and } M_i' = m_i$$

x	y	z	Maxterms	Designation
0	0	0	$x + y + z$	M_0
0	0	1	$x + y + z'$	M_1
0	1	0	$x + y' + z$	M_2
0	1	1	$x + y' + z'$	M_3
1	0	0	$x' + y + z$	M_4
1	0	1	$x' + y + z'$	M_5
1	1	0	$x' + y' + z$	M_6
1	1	1	$x' + y' + z'$	M_7

Maxterms for Three Binary Variables

Maxterms (2)

- **Any Boolean function can be expressed as a product (AND) of its 0-maxterms:**

$$F(\textbf{list of variables}) = \Pi(\textbf{list of 0-maxterm indices})$$

Example:

Row Numbers	Variable Values			Function Values	
	x	y	z	F_1	F_1'
0	0	0	0	0	1
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	0

Equation Table

$$F_1(x, y, z) = \Pi(0, 1, 2, 4) = M_0 M_1 M_2 M_4 = (x + y + z)(x + y + z')(x + y' + z)(x' + y + z)$$

$$F_1'(x, y, z) = \Pi(3, 5, 6, 7) = M_3 M_5 M_6 M_7 = (x + y' + z')(x' + y + z')(x' + y' + z)(x' + y' + z')$$

Maxterms (3)

- **Any Boolean function can be expanded into a product-of-maxterms form by expanding each term with xx' for each missing variable x.**

Example:

$$
\begin{aligned}
F &= (x + y')(x' + z)(y' + z) \\
&= (x + y' + zz')(x' + z + yy')(y' + z + xx') \\
&= (x + y' + z)(x + y' + z')\ (x' + y + z)(x' + y' + z)\ (x + y' + z)(x' + y' + z) \\
&= (x + y' + z)(x + y' + z')\ (x' + y + z)(x' + y' + z) \\
&= \quad M_2 \qquad\qquad M_3 \qquad\qquad M_4 \qquad\qquad M_6 \\
&= \prod(2, 3, 4, 6)
\end{aligned}
$$

Maxterms (4)

- **Any Boolean expression can be converted into a sum-of-maxterms by generating the truth table and listing all the 0-maxterms.**
  - **Example:** $F = x'y' + xz$

x	y	z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$$F(x, y, z) = \Sigma(0, 1, 5, 7)$$

$$F(x, y, z) = \Pi(2, 3, 4, 6)$$

Canonical Forms

- **Two canonical forms:**
  - **Sum-of-minterms**
  - **Product-of-maxterms**
- **Canonical forms are unique.**
- **Conversion between canonical forms is achieved by:**
  - **Exchanging Σ and Π**
  - **Listing all the missing indices**

Standard Forms (1)

- **Two standard forms**
  - **Sum-of-products**
  - **Product-of-sums**
- **Standard forms are not unique.**

- **Sum-of-products is an** OR **expression with product terms that may have less literals than minterms**

Example:

$$F = xy + x'yz + xy'z$$

Standard Forms (2)

- **Product-of-sums is an AND expression with sum terms that may have less literals than maxterms**

  Example:

  $$F = (x' + y')(x + y' + z')(x' + y + z')$$

- **Standard forms have fewer operators (literals) than canonical forms**
- **Standard forms can be derived from canonical forms by eliminating 1-subcubes (= all subset minterms in the expression)**

  Example:

  $$\begin{aligned} F_1 &= xyz + xyz' + xy'z + xy'z' \\ &= xy(z + z') + xy'(z + z') \\ &= xy + xy' \\ &= x(y + y') \\ &= x \end{aligned}$$

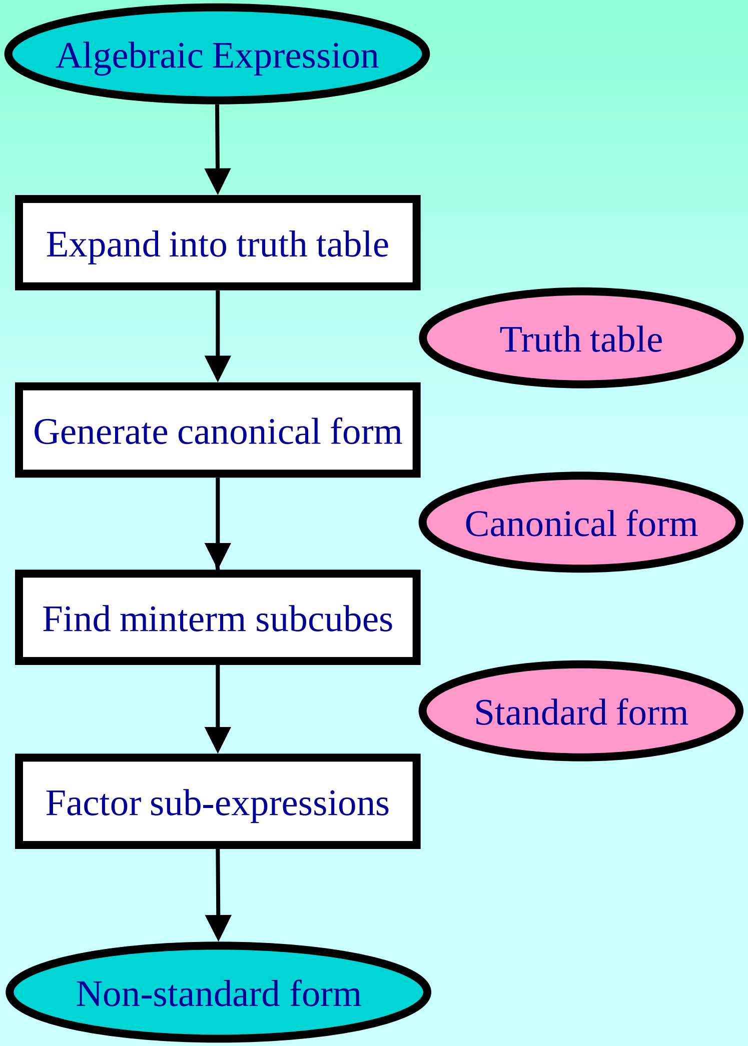
Non-standard Forms

- **Non-standard forms have fewer operators (literals) than standard forms.**
- **They are obtained by factoring variables.**

Example:

$$xy + xy'z + xy'w = x(y + y'z + y'w)$$
$$= x(y + y'(z + w))$$

Operator (Literal) Reduction in Boolean Expressions



Summary

- **Boolean Algebra**
  - **Axioms**
  - **Axiomatic definition**
  - **Basic theorems**

- **Boolean Functions**

- **Specification of Boolean Functions**
  - **Truth tables**
  - **Algebraic expressions**
    - ***Canonical forms***
    - ***Standard forms***
    - ***Non-standard forms***

- **Algebraic Optimization of Boolean Expressions**