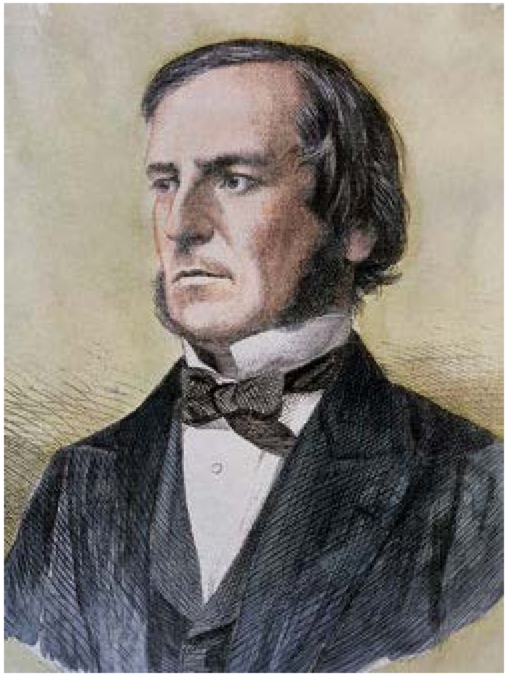
Topic 3

Boolean Algebra & Optimization

Boolean Algebra



- **"Traditional" algebra**
  - Variables represent real numbers
  - Operators operate on variables, and return real numbers
- ***Boolean Algebra***
  - Developed mid-1800's by George Boole to formalize human thought
  - Variables represent 0 or 1 only
  - Operators return 0 or 1 only
  - Basic operators
    - AND, OR, NOT

Boolean Algebra Terminology

- **Example equation:** F(a,b,c) = a'bc + abc' + ab + c
- *Variable*
  - Represents a value (0 or 1)
  - Three variables: a, b, and c
- *Literal*
  - Appearance of a variable, in true or complemented form
  - Nine literals: a', b, c, a, b, c', a, b, and c
- *Product term*
  - AND of literals
  - Four product terms: a'bc, abc', ab, c
- *Sum term*
  - OR of literals
  - No sum terms
- *Sum-of-products*
  - Equation written as OR of product terms only
  - Above equation is in sum-of-products form. "F = (a+b)c + d" is not.

Basic Theorems of Boolean Algebra

- **(a) $x + 0 = x$; (b) $x \bullet 0 = 0$; (theorem 1)**
- **(a) $x + x' = 1$; (b) $x \bullet x' = 0$; (theorem 2)**
- **(a) $x + x = x$; (b) $x \bullet x = x$; (theorem 3)**
- **(a) $x + 1 = 1$; (b) $x \bullet 1 = x$; (theorem 4)**
- **$(x')' = x$; (involution)**

Basic Theorems of Boolean Algebra

• (a) $x + y = y + x$;	(b) $xy = yx$;	(commutative)
• (a) $x + (y + z) = (x + y) + z$;	(b) $x(yz) = (xy)z$;	(associative)
• (a) $x(y + z) = xy + xz$;	(b) $x + yz = (x+y)(x+z)$;	(distributive)
• (a) $x + xy = x$;	(b) $x(x + y) = x$;	(absorption)
• (a) $xy + xy' = x$;	(b) $(x + y)(x + y') = x$	(theorem 5)
• (a) $x + x'y = x + y$	(b) $x(x' + y) = xy$	(theorem 6)

Operator Precedence

- **The operator precedence for evaluating basic Boolean expressions is:**
  - Parenthesis
  - NOT
  - AND
  - OR
- **Example: (x + y)'**
  - Evaluate the parenthesized expression (x + y) first and then the inversion
- **Example: x + xy**
  - Evaluate xy first and then OR it with the value of x

Application of Basic Theorems

- **Prove theorem 5(a): $xy + xy' = x$**

$$
\begin{aligned}
& xy + xy' \\
= \; & x(y + y') && \text{(distributive (a))} \\
= \; & x \bullet 1 && \text{(theorem 2(a))} \\
= \; & x && \text{(theorem 4(b))}
\end{aligned}
$$

Application of Basic Theorems

- **Prove theorem 5(b): $(x + y)(x + y') = x$**

$$
\begin{aligned}
& (x + y)(x + y') \\
&= x + yy' && \text{(distributive (b))} \\
&= x + 0 && \text{(theorem 2(b))} \\
&= x && \text{(theorem 1(a))}
\end{aligned}
$$

Application of Basic Theorems

- **Prove theorem 5(b): $(x + y)(x + y') = x$, alternatively**

$$
\begin{aligned}
& (x + y)(x + y') \\
= \; & (x + y)x + (x + y)y' && \text{(distributive (a))} \\
= \; & xx + xy + xy' + yy' && \text{(distributive (a))} \\
= \; & x + xy + xy' + 0 && \text{(theorem 2(b), 3(b))} \\
= \; & x + x(y + y') && \text{(theorem 1(a), distributive (a))} \\
= \; & x + x && \text{(theorem 2(a), 4(b))} \\
= \; & x && \text{(theorem 3(a))}
\end{aligned}
$$

Application of Basic Theorems

- **Prove theorem 6(a): $x + x'y = x + y$**

$$
\begin{aligned}
& x + x'y \\
= \; & (x + x')(x + y) && \text{(distributive (a))} \\
= \; & 1 \bullet (x + y) && \text{(theorem 2(a))} \\
= \; & x + y && \text{(theorem 4(b))}
\end{aligned}
$$

Application of Basic Theorems

- **Exercises**
  1. $x'y + x'$
  2. $a'bc + a'$
  3. $a'b'c + (a'b'c)'$
  4. $(a + b)(c + b)(d' + b)(acd' + e)$
  5. $wx'y' + wxz' + wx'yz'$

DeMorgan's Law

(a) $(x + y)' = x'y'$

(b) $(xy)' = x' + y'$

- **Very Useful**

Applications of DeMorgan’s Law

- **Find the complement of $F = x(y'z' + yz)$**
- **$F' = (x(y'z' + yz))'$ (All steps by DeMorgan’s law)**

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

  $= x' + (y'z')' \bullet (yz)'$

  $= x' + (y + z)(y' + z')$
- **Exercise**

  $((AB' + C)D' + E)'$

XOR Properties

$x \oplus 0 = x$ (a)	$x \oplus 1 = x'$ (b)	(theorem 1)
$x \oplus x = 0$ (a)	$x \oplus x' = 1$ (b)	(theorem 2)
$x \oplus y' = x' \oplus y = (x \oplus y)'$		(theorem 3)
$x \oplus y = y \oplus x$		(commutative)
$(x \oplus y) \oplus z = x \oplus (y \oplus z) = x \oplus y \oplus z$		(associative)

Boolean Representation: Minterm and Maxterm

- A binary literal may be in the unprimed (true) form and primed (false) forms, representing true and false conditions respectively
  - E.g. a vs. a'
- **Minterm** is a product of n literals in which each literal appears exactly once in either true or complemented form, but not both
  - Minterm is represented by $\boldsymbol{m_i}$
- **Maxterm** is a sum of n literals in which each literal appears exactly once in either true or complemented form, but not both
  - Maxterm is represented by $\boldsymbol{M_i}$

Minterm and Maxterm

			Minterms		Maxterms	
x	**y**	**z**	**Term**	**Designation**	**Term**	**Designation**
0	0	0	x'y'z'	m_0	x+y+z	M_0
0	0	1	x'y'z	m_1	x+y+z'	M_1
0	1	0	x'yz'	m_2	x+y'+z	M_2
0	1	1	x'yz	m_3	x+y'+z'	M_3
1	0	0	xy'z'	m_4	x'+y+z	M_4
1	0	1	xy'z	m_5	x'+y+z'	M_5
1	1	0	xyz'	m_6	x'+y'+z	M_6
1	1	1	xyz	m_7	x'+y'+z'	M_7

Subscription *i* of minterm is the decimal equivalent of the corresponding binary combination

Minterm in Truth Table

x con1	y con2	z con3	F result
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Result would happen if con1 is **false** AND con2 is **false** AND con3 is **true**, **x'y'z**

Result would happen if con1 is **false** AND con2 is **true** AND con3 is **true**, **x'yz**

Result would happen if con1 is **true** AND con2 is **false** AND con3 is **false**, **xy'z'**

Result would happen if con1 is **true** AND con2 is **false** AND con3 is **true**, **xy'z**

Result would be true if any of these four conditions is true, implies OR logic, This relationship is expressed by:
F = x'y'z + x'yz + xy'z' + xy'z

Minterm Expression From Truth Table

- A Boolean Equation can be derived from a truth table and expressed as a sum-of-minterms (**standard-sum-of-products**)
- **The minterms chosen in the sum-of-minterms expression are those which produce a logic 1 for the corresponding output**
- Example:

  $\mathbf{F = x'y'z + x'yz + xy'z' + xy'z}$

  $\mathbf{= m_1 + m_3 + m_4 + m_5}$

  $\mathbf{= \Sigma\ m(1, 3, 4, 5)}$

x con1	y con2	z con3	F result
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Exercise

- **Find minterm logic equation from these truth table**

x	y	z	F
0	0	0	1
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

W	X	Y	Z	F		
0	0	0	0	1	**m0**	**W’X’Y’Z’**
0	0	0	1	0	**m1**	**W’X’Y’Z**
0	0	1	0	0	**m2**	**W’X’YZ’**
0	0	1	1	1	**m3**	**W’X’YZ**
0	1	0	0	0	**m4**	**W’XY’Z’**
0	1	0	1	0	**m5**	**W’XY’Z**
0	1	1	0	0	**m6**	**W’XYZ’**
0	1	1	1	1	**m7**	**W’XYZ**
1	0	0	0	1	**m8**	**WX’Y’Z’**
1	0	0	1	0	**m9**	**WX’Y’Z**
1	0	1	0	0	**m10**	**WX’YZ’**
1	0	1	1	0	**m11**	**WX’YZ**
1	1	0	0	0	**m12**	**WXY’Z’**
1	1	0	1	0	**m13**	**WXY’Z**
1	1	1	0	0	**m14**	**WXYZ’**
1	1	1	1	1	**m15**	**WXYZ**

Minterms and Maxterms

- **The complement of Minterm is the corresponding Maxterm, vice versa**
  - $m_i' = M_i$
  - e.g.: $m_0 = x'y'z'$

    $m_0' = (x'y'z')' = x + y + z = M_0$ (DeMorgan's)
- **Conversion between Standard Forms**
  - the term numbers missing from one form will be the term numbers used in the other form
  - e.g.: if all the terms are indexed by 0 ~ 7, then

    $F = \Sigma\, m(1, 2, 4, 7) = \Pi\, M(0, 3, 5, 6)$

Minterms and Maxterms

- **Example: In the given truth table, F1 is output of a 3-input device**

Truth Table

x	y	z	F1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Sum-of-minterms

$F1 = x'y'z + xy'z' + xy'z$
$xyz' + xyz$

$F1 = m_1+m_4+m_5+m_6+m_7$

$F1 = \Sigma\ (1, 4, 5, 6, 7)$

Product-of-maxterms

$F1 = (x+y+z) \bullet (x+y'+z) \bullet$
$(x+y'+z')$

$F1 = M_0 \bullet M_2 \bullet M_3$

$F1 = \Pi\ (0, 2, 3)$

Incompletely Specified Functions

- **In a circuit, some input conditions may never happen, then the output is not completely specified**
- **The corresponding output is designated as "x", called *don't care***
- **A *don't care* output could be either 0 or 1**
- **$F = \Sigma m(1, 3, 4)$ with $d(2, 5)$**

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

Simplified Forms

- **The minterm and maxterm forms can be further simplified**
  - Boolean function may contain less number of terms
  - Each term may have less literals
  - e.g.:

Simplified SOP
F1 = x + y'z

Simplified POS
F1 = (x + y')(x + z)

Why to simplify? &
How to?

  - Why?
  - How to? Boolean theorems. And more….