Chapter 2
Boolean Algebra and Logic Gates

Definitions

Theorems

Functions

Canonical and Standard Forms

Operations

Gates

Integrated Circuits

Algebraic Manipulation

- **By reducing the number of terms, the number of literals (single variable) or both in a Boolean function, it is possible to obtain a simpler circuit, as each term requires a gate and each variable within the term designates an input to the gate .**
  - **$F_1 = x'y'z + x'yz + xy'$ contains 3 terms and 8 literals**
  - **$F_2 = x'z + xy'$ contains 2 terms and 4 literals.**

Example Manipulations

- **The following are some example manipulations:**
  1. $x(x' + y) = xx' + xy = 0 + xy = xy$
  2. $x + x'y = (x + x')(x + y) = 1(x + y) = x + y$
  3. $(x + y)(x + y') = x + xy + xy' + yy' = x(1 + y + y') = x$
  4. $xy + x'z + yz = xy + x'z + yz(x + x')$
     $= xy + x'z + xyz + x'yz$
     $= xy(1 + z) + x'z(1 + y)$
     $= xy + x'z$
  5. $(x + y)(x' + z)(y + z) = (x + y)(x' + z)$ – HOME ASSIGNMENT!

Complement of a Function

- **The complement of a function F is F'.**
  - **It is obtained by interchanging 0's for 1's and 1's for 0's in the value of F.**
- **The** complement **of a function may be derived algebraically through** DeMorgan's theorem**.**
  - **Theorem 5(a) (DeMorgan): $(x + y)' = (x' \cdot y')$**
  - **Theorem 5(b) (DeMorgan): $(x \cdot y)' = (x' + y')$**
- **Example:**
  - $F_1 = x'yz' + x'y'z$

    $F_1' = (x'yz' + x'y'z)'$

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

Complement of a Function (Example)

- If $F_1 = A+B+C$
- Then F_1'

$=(A+B+C)'$

$= (A+X)'$ let $B+C = X$

$= A'X'$ by DeMorgan's

$= A'(B+C)'$

$= A'(B'C')$ by DeMorgan's

$= A'B'C'$ associative

Complement of a Function (More Examples)

- **$(x'yz' + x'y'z)'$**

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

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

- **$[x(y'z'+yz)]'$**

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

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

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

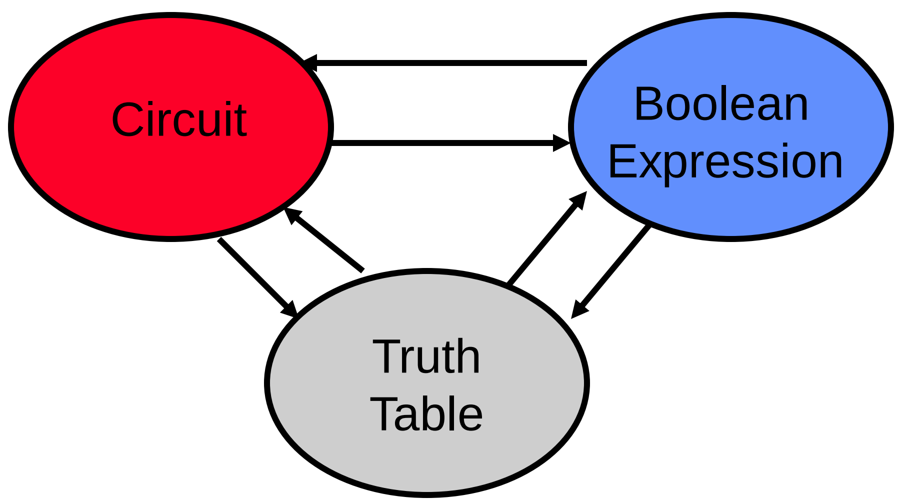
- A simpler procedure
  - **take the** dual **of the function (interchanging AND and OR operators and 1's and 0's) and** complement **each literal. {DeMorgan's Theorem}**
  - **$x'yz' + x'y'z$**

    The dual **of function is $(x'+y+z') (x'+y'+z)$**

    Complement **of each literal: $(x+y'+z)(x+y+z')$**

Representation Conversion

- **Need to transition between boolean expression, truth table, and circuit (symbols).**
- **Converting between truth table and expression is easy.**
- **Converting between expression and circuit is easy.**
- **More difficult to convert to truth table.**



Canonical Forms

- **A canonical form is a standard method for representing Boolean functions.**
- **The two canonical forms that are used are:**
  - **Sum of Minterms**
  - **Product of Maxterms**
- **These forms are sometimes considered the “brute force” method of representing functions as they seldom represent a function in a minimized form.**

Minterms

- Any given binary variable can be represented in two forms:
  - x, its normal form, and
  - x', its complement
- If we consider two binary variables and the AND operation, there are four combinations of the variables:
  - xy
  - xy'
  - x'y
  - x'y'
- Each of the above four AND terms is called a minterm or a standard product.
- n variables can be combined to form 2^n minterms.

Minterms Expressed

				Minterms	
x	**y**	**z**		**Term**	**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		x'yz'	m_5
1	1	0		xyz'	m_6
1	1	1		xyz	m_7

Maxterms

- **Any given binary variable can be represented in two forms:**
  - **x, its normal form, and**
  - **x', its complement**
- **If we consider two binary variables and the OR operation, there are four combinations of the variables:**
  - **x + y**
  - **x + y'**
  - **x' + y**
  - **x' + y'**
- **Each of the above four OR terms is called a maxterm or a standard sum.**
- **n variables can be combined to form 2^n maxterms.**
- Each maxterm is the complement of its corresponding minterm and vice-versa.

Maxterms Expressed

				Maxterms	
x	y	z		Term	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

Truth Table to Expression (Sum of Minterms)

- Any Boolean function can be expressed as a sum of minterms or sum of products (i.e. the ORing of terms).
  - You can form the function algebraically by forming a minterm for each combination of the variables that produces a 1 in the function. (Each row with output of 1 becomes a product term)
  - Sum (OR) product terms together.

x	y	z	G
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

xyz + xyz' + x'yz

Sum of Minterms Example

x	y	z		Function F_1	Required Minterms
0	0	0		1	x'y'z'
0	0	1		0	
0	1	0		0	
0	1	1		1	x'yz
1	0	0		1	xy'z'
1	0	1		0	
1	1	0		0	
1	1	1		0	

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

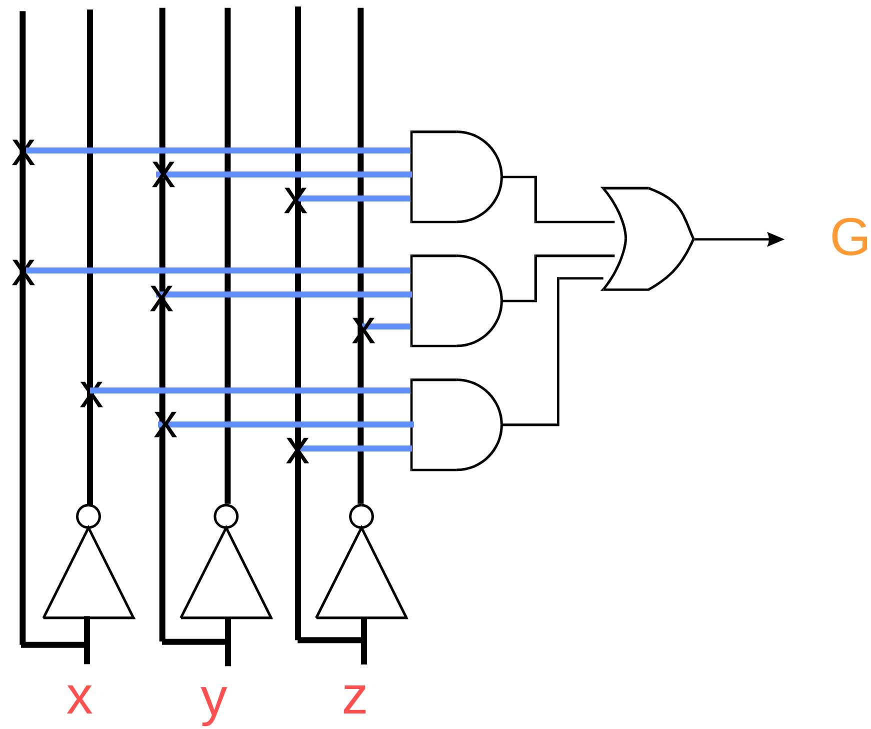
$$= m_0+m_3+m_4$$

$$= \sum(0,3,4)$$

Equivalent Representations of Circuits

- All three formats are equivalent
- Number of 1's in truth table output column equals AND terms for Sum-of-Products (SOP)

x	y	z	G
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



G = xyz + xyz' + x'yz