Digital Logic Design

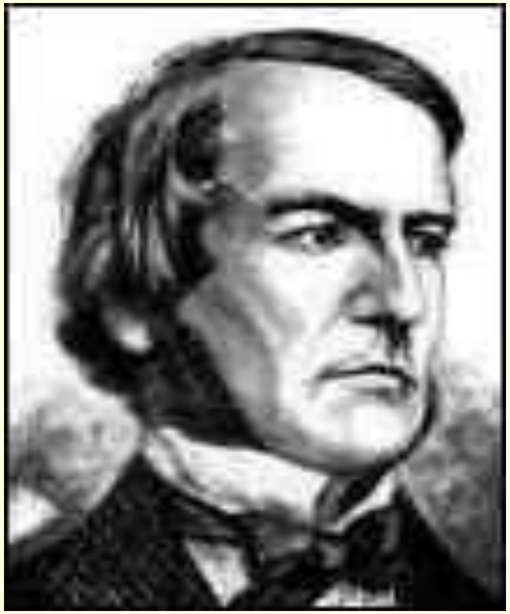
Boolean Algebra and Logic Gate

Algebras

- What is an algebra?
  - Mathematical system consisting of
    - Set of elements
    - Set of operators
    - Axioms or postulates
- Why is it important?
  - Defines rules of “calculations”
- Example: arithmetic on natural numbers
  - Set of elements: $N = \{1,2,3,4,\ldots\}$
  - Operator: +, –, *
  - Axioms: associativity, distributivity, closure, identity elements, etc.
- Note: operators with two inputs are called ***binary***
  - Does not mean they are restricted to binary numbers!
  - Operator(s) with one input are called *unary*

George Boole

- **Father of Boolean algebra**
- He came up with a type of linguistic algebra, the three most basic operations of which were (and still are) AND, OR and NOT. It was these three functions that formed the basis of his premise, and were the only operations necessary to perform comparisons or basic mathematical functions.
- Boole's system (detailed in his 'An Investigation of the Laws of Thought, on Which Are Founded the Mathematical Theories of Logic and Probabilities', 1854) was based on a binary approach, processing only two objects - the yes-no, true-false, on-off, zero-one approach.
- Surprisingly, given his standing in the academic community, Boole's idea was either criticized or completely ignored by the majority of his peers.
- Eventually, one bright student, Claude Shannon (1916-2001), picked up the idea and ran with it



George Boole (1815 - 1864)

Boolean Algebra

- Terminology:
  - *Literal:* A variable or its complement
  - *Product term:* literals connected by •
  - *Sum term:* literals connected by +

Postulates of Two-Valued Boolean Algebra

- $B = \{0, 1\}$ and two binary operations, + and.
- The rules of operations: AND、OR and NOT.

AND

x	y	$x.\ y$
0	0	0
0	1	0
1	0	0
1	1	1

OR

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

NOT

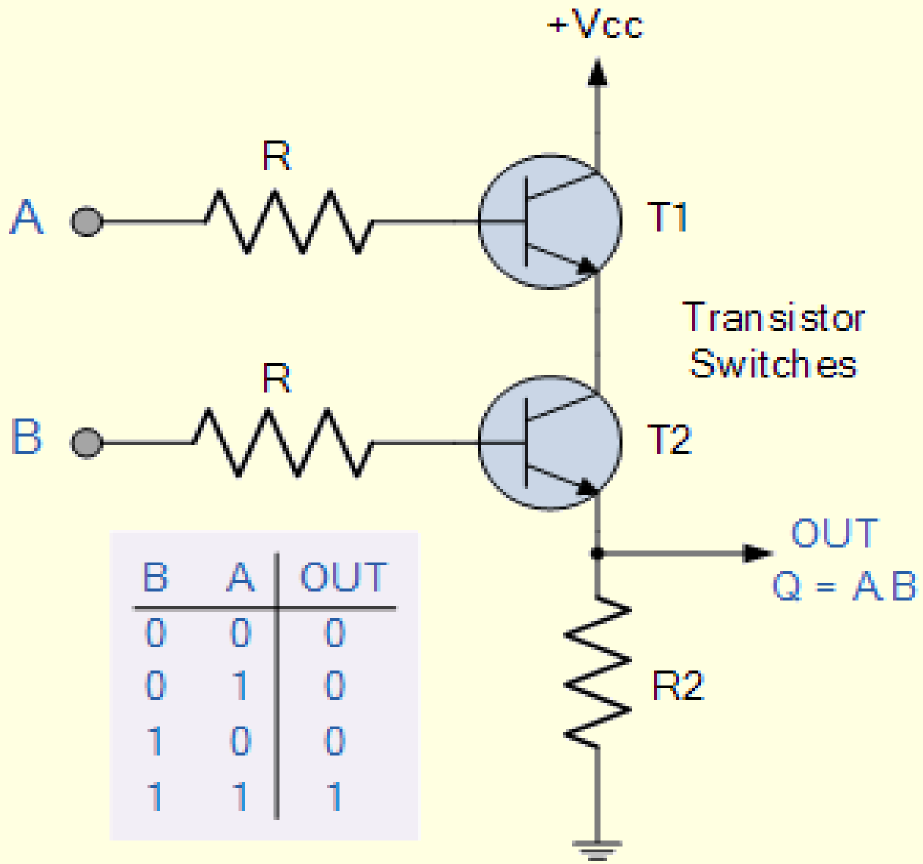
x	x'
0	1
1	0

1. Closure (+ and·)
2. The identity elements

   (1) +: 0

   (2). : 1

How is the behavior implemented



B	A	OUT
0	0	0
0	1	0
1	0	0
1	1	1

Postulates of Two-Valued Boolean Algebra

3. The commutative laws
4. The distributive laws

x	y	z	$y+z$	$x.(y+z)$	$x.y$	$x.z$	$(x.y)+(x.z)$
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

Postulates of Two-Valued Boolean Algebra

5. Complement
   - $x+x'=1 \rightarrow 0+0'=0+1=1;\ 1+1'=1+0=1$
   - $x \cdot x'=0 \rightarrow 0 \cdot 0'=0 \cdot 1=0;\ 1 \cdot 1'=1 \cdot 0=0$
6. Has two distinct elements 1 and 0, with $0 \neq 1$

- Note
  - A set of two elements
  - + : OR operation; . : AND operation
  - A complement operator: NOT operation
  - Binary logic is a two-valued Boolean algebra

Duality

- The principle of *duality* is an important concept. This says that if an expression is valid in Boolean algebra, the dual of that expression is also valid.
- To form the dual of an expression, replace all + operators with . operators, all . operators with + operators, all ones with zeros, and all zeros with ones.
- Form the dual of the expression

  $a + (bc) = (a + b)(a + c)$
- Following the replacement rules…

  $a(b + c) = ab + ac$
- Take care not to alter the location of the parentheses if they are present.

Basic Theorems

Table 2.1
Postulates and Theorems of Boolean Algebra

Postulate 2	(a)	$x + 0 = x$	(b)	$x \cdot 1 = x$
Postulate 5	(a)	$x + x' = 1$	(b)	$x \cdot x' = 0$
Theorem 1	(a)	$x + x = x$	(b)	$x \cdot x = x$
Theorem 2	(a)	$x + 1 = 1$	(b)	$x \cdot 0 = 0$
Theorem 3, involution		$(x')' = x$		
Postulate 3, commutative	(a)	$x + y = y + x$	(b)	$xy = yx$
Theorem 4, associative	(a)	$x + (y + z) = (x + y) + z$	(b)	$x(yz) = (xy)z$
Postulate 4, distributive	(a)	$x(y + z) = xy + xz$	(b)	$x + yz = (x + y)(x + z)$
Theorem 5, DeMorgan	(a)	$(x + y)' = x'y'$	(b)	$(xy)' = x' + y'$
Theorem 6, absorption	(a)	$x + xy = x$	(b)	$x(x + y) = x$

Proof of $x+x=x$

- We can only use Huntington postulates:

<u>Huntington postulates</u>:

Post. 2: (a) $x+0=x$, (b) $x \cdot 1=x$
Post. 3: (a) $x+y=y+x$, (b) $x \cdot y=y \cdot x$
Post. 4: (a) $x(y+z) = xy+xz$,
(b) $x+yz = (x+y)(x+z)$
Post. 5: (a) $x+x'=1$, (b) $x \cdot x'=0$

- Show that $x+x=x$.

$x+x$	$= (x+x) \cdot 1$	by 2(b)
	$= (x+x)(x+x')$	by 5(a)
	$= x+xx'$	by 4(b)
	$= x+0$	by 5(b)
	$= x$	by 2(a)
	Q.E.D.	

- We can now use Theorem 1(a) in future proofs

Proof of $x \cdot x = x$

- Similar to previous proof

<u>Huntington postulates</u>:

Post. 2: (a) $x+0=x$, (b) $x \cdot 1=x$
Post. 3: (a) $x+y=y+x$, (b) $x \cdot y=y \cdot x$
Post. 4: (a) $x(y+z) = xy+xz$,
(b) $x+yz = (x+y)(x+z)$
Post. 5: (a) $x+x'=1$, (b) $x \cdot x'=0$
Th. 1: (a) $x+x=x$

- Show that $x \cdot x = x$.

$x \cdot x$	$= xx+0$	by 2(a)
	$= xx+xx'$	by 5(b)
	$= x(x+x')$	by 4(a)
	$= x \cdot 1$	by 5(a)
	$= x$	by 2(b)

Q.E.D.

Proof of $x+1=1$

- Theorem 2(a): $x + 1 = 1$

$x + 1 = 1. \quad (x + 1)$ by 2(b)

$=(x + x')(x + 1)$	5(a)
$= x + x'\,1$	4(b)
$= x + x'$	2(b)
$= 1$	5(a)

Huntington postulates:

Post. 2: (a) $x+0=x$, (b) $x\cdot 1=x$
Post. 3: (a) $x+y=y+x$, (b) $x\cdot y=y\cdot x$
Post. 4: (a) $x(y+z) = xy+xz$,
(b) $x+yz = (x+y)(x+z)$
Post. 5: (a) $x+x'=1$, (b) $x\cdot x'=0$
Th. 1: (a) $x+x=x$

- Theorem 2(b): $x.\ \ 0 = 0$ by duality
- Theorem 3: $(x')' = x$
  - Postulate 5 defines the complement of x, $x + x' = 1$ and $x\,x' = 0$
  - The complement of x' is x is also $(x')'$

Absorption Property (Covering)

- Theorem 6(a): $x + xy = x$
  - $x + xy = x.\ 1 + xy$ *by* 2(b)

    $= x\,(1 + y)$ 4(a)

    $= x\,(y + 1)$ 3(a)

    $= x.\ 1$ Th 2(a)

    $= x$ 2(b)

Huntington postulates:

Post. 2: (a) $x+0=x$, (b) $x\cdot 1=x$
Post. 3: (a) $x+y=y+x$, (b) $x\cdot y=y\cdot x$
Post. 4: (a) $x(y+z) = xy+xz$,
(b) $x+yz = (x+y)(x+z)$
Post. 5: (a) $x+x'=1$, (b) $x\cdot x'=0$
Th. 1: (a) $x+x=x$

- Theorem 6(b): $x\,(x + y) = x$ by duality
- By means of truth table (another way to proof)

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

DeMorgan's Theorem

- Theorem 5(a): $(x + y)' = x'y'$
- Theorem 5(b): $(xy)' = x' + y'$
- By means of truth table

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

Consensus Theorem

1. $xy + x'z + yz = xy + x'z$
2. $(x+y)\cdot(x'+z)\cdot(y+z) = (x+y)\cdot(x'+z)$ -- (dual)

- **Proof:**

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

QED (2 true by duality).

Operator Precedence

- The operator precedence for evaluating Boolean Expression is
  - Parentheses
  - NOT
  - AND
  - OR
- Examples
  - $x\,y' + z$
  - $(x\,y + z)'$

Boolean Functions

- A Boolean function
  - Binary variables
  - Binary operators OR and AND
  - Unary operator NOT
  - Parentheses
- Examples
  - $F_1 = x\, y\, z'$
  - $F_2 = x + y'z$
  - $F_3 = x'\, y'\, z + x'\, y\, z + x\, y'$
  - $F_4 = x\, y' + x'\, z$

Boolean Functions

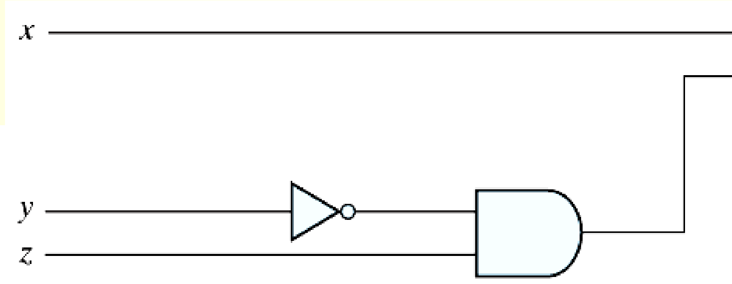
- The truth table of 2^n entries

x	y	z	F_1	F_2	F_3	F_4
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	0	0	0	0
0	1	1	0	0	1	1
1	0	0	0	1	1	1
1	0	1	0	1	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	0

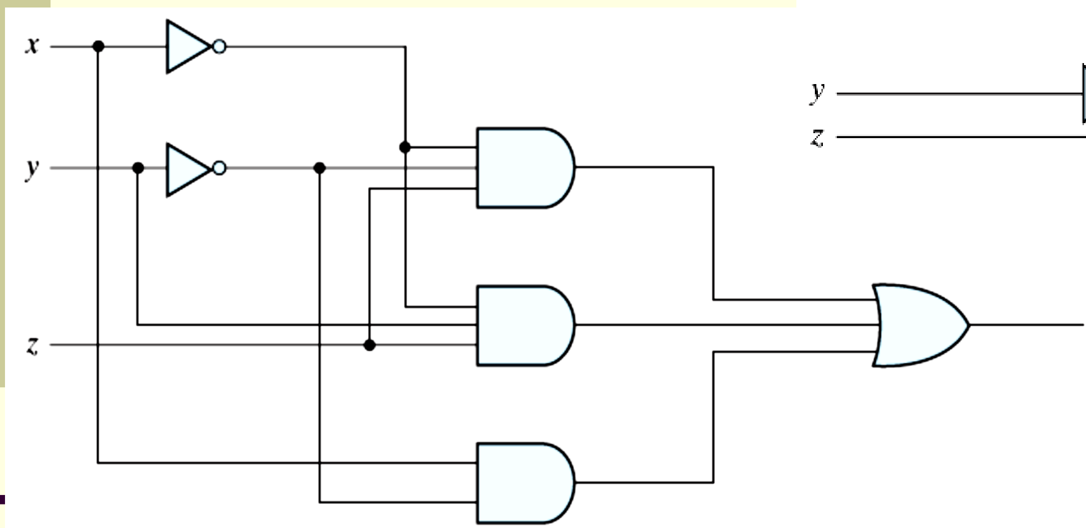
- Two Boolean expressions may specify the same function
  - $F_3 = F_4$

Boolean Functions

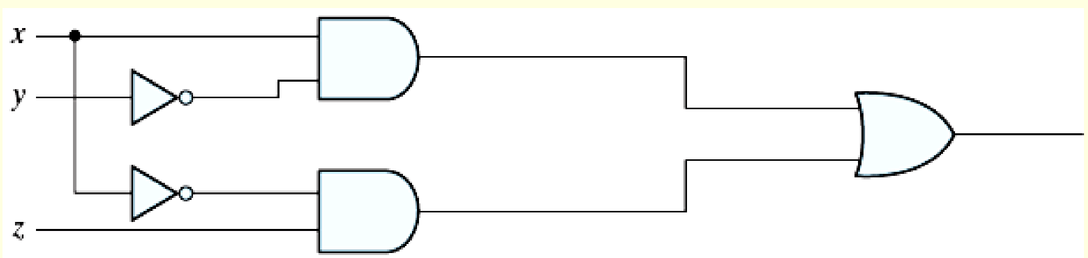
- Implementation with logic gates
  - F_4 is more economical



$F_2 = x + y'z$



$F_3 = x'y'z + x'yz + xy'$



$F_4 = xy' + x'z$

Algebraic Manipulation

- To minimize Boolean expressions
  - Literal: a primed or unprimed variable (an input to a gate)
  - Term: an implementation with a gate
  - The minimization of the number of literals and the number of terms → a circuit with less equipment
  - It is a hard problem (no specific rules to follow)
- Example 2.1
  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. $xy + x'z + yz = xy + x'z + yz(x+x') = xy + x'z + yzx + yzx' = xy(1+z) + x'z(1+y) = xy +x'z$
  5. $(x+y)(x'+z)(y+z) = (x+y)(x'+z)$, by duality from function 4. (*consensus theorem* with duality)

Complement of a Function

- An interchange of 0's for 1's and 1's for 0's in the value of F
  - By DeMorgan's theorem
  - $(A+B+C)' = (A+X)'$ let $B+C = X$

    $= A'X'$ by theorem 5(a) (DeMorgan's)

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

    $= A'(B'C')$ by theorem 5(a) (DeMorgan's)

    $= A'B'C'$ by theorem 4(b) (associative)
- *Generalizations*: a function is obtained by interchanging AND and OR operators and complementing each literal.
  - $(A+B+C+D+ \ldots +F)' = A'B'C'D' \ldots F'$
  - $(ABCD \ldots F)' = A'+ B'+C'+D' \ldots +F'$

Examples

- Example 2.2
  - $F_1' = (x'yz' + x'y'z)' = (x'yz')' (x'y'z)' = (x+y'+z) (x+y+z')$
  - $F_2' = [x(y'z'+yz)]' = x' + (y'z'+yz)' = x' + (y'z')' (yz)'$

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

    $= x' + yz'+y'z$
- Example 2.3: a simpler procedure
  - Take the dual of the function and complement each literal

1. $F_1 = x'yz' + x'y'z$.

   The dual of F_1 is $(x'+y+z') (x'+y'+z)$.

   Complement each literal: $(x+y'+z)(x+y+z') = F_1'$
2. $F_2 = x(y' z' + yz)$.

   The dual of F_2 is $x+(y'+z') (y+z)$.

   Complement each literal: $x'+(y+z)(y' +z') = F_2'$

2.6 Canonical and Standard Forms

Minterms and Maxterms

- A minterm (standard product): an AND term consists of all literals in their normal form or in their complement form.
  - For example, two binary variables *x* and *y,*
    - *xy, xy', x'y, x'y'*
  - It is also called a standard product.
  - *n* variables con be combined to form 2^n minterms.
- A maxterm (standard sums): an OR term
  - It is also call a standard sum.
  - 2^n maxterms.

Minterms and Maxterms

- Each *maxterm* is the complement of its corresponding *minterm*, and vice versa.

Table 2.3
Minterms and Maxterms for Three Binary Variables

			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

Minterms and Maxterms

- An Boolean function can be expressed by
  - A truth table
  - Sum of minterms
  - $f_1 = x'y'z + xy'z' + xyz = m_1 + m_4 + m_7$ (Minterms)
  - $f_2 = x'yz + xy'z + xyz' + xyz = m_3 + m_5 + m_6 + m_7$ (Minterms)

Table 2.4
Functions of Three Variables

x	y	z	Function f_1	Function f_2
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Minterms and Maxterms

- The complement of a Boolean function
  - The minterms that produce a 0
  - $f_1' = m_0 + m_2 + m_3 + m_5 + m_6 \qquad = x'y'z'+x'yz'+x'yz+xy'z+xyz'$
  - $f_1 = (f_1')'$
    $= (x+y+z)(x+y'+z)\ (x+y'+z')\ (x'+y+z')(x'+y'+z) = M_0\ M_2\ M_3\ M_5\ M_6$
  - $f_2 = (x+y+z)(x+y+z')(x+y'+z)(x'+y+z) = M_0M_1M_2M_4$
- Any Boolean function can be expressed as
  - A sum of minterms ("sum" meaning the ORing of terms).
  - A product of maxterms ("product" meaning the ANDing of terms).
  - Both boolean functions are said to be in Canonical form.

Sum of Minterms

- Sum of minterms: there are 2^n minterms and 2^{2n} combinations of function with *n* Boolean variables.
- Example 2.4: express $F = A+BC'$ as a sum of minterms.
  - $F = A+B'C = A\,(B+B') + B'C = AB + AB' + B'C = AB(C+C') + AB'(C+C') + (A+A')B'C = ABC+ABC'+AB'C+AB'C'+A'B'C$
  - $F = A'B'C + AB'C' + AB'C + ABC' + ABC = m_1 + m_4 + m_5 + m_6 + m_7$
  - $F(A, B, C) = \Sigma(1, 4, 5, 6, 7)$
  - or, built the truth table first

Table 2.5
Truth Table for $F = A + B'C$

A	*B*	*C*	*F*
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

Product of Maxterms

- Product of maxterms: using distributive law to expand.
  - $x + yz = (x + y)(x + z) = (x+y+zz')(x+z+yy') = (x+y+z)(x+y+z')(x+y'+z)$
- Example 2.5: express $F = xy + x'z$ as a product of maxterms.
  - $F = xy + x'z = (xy + x')(xy +z) = (x+x')(y+x')(x+z)(y+z) = (x'+y)(x+z)(y+z)$
  - $x'+y = x' + y + zz' = (x'+y+z)(x'+y+z')$
  - $F = (x+y+z)(x+y'+z)(x'+y+z)(x'+y+z') = M_0M_2M_4M_5$
  - $F(x, y, z) = \Pi(0, 2, 4, 5)$

Conversion between Canonical Forms

- The complement of a function expressed as the sum of minterms equals the sum of minterms missing from the original function.
  - $F(A, B, C) = \Sigma(1, 4, 5, 6, 7)$
  - Thus, $F'(A, B, C) = \Sigma(0, 2, 3)$
  - By DeMorgan's theorem

    $F(A, B, C) = \Pi(0, 2, 3)$

    $F'(A, B, C) = \Pi(1, 4, 5, 6, 7)$
  - $m_j' = M_j$
  - Sum of minterms = product of maxterms
  - Interchange the symbols Σ and Π and list those numbers missing from the original form
    - Σ of 1's
    - Π of 0's

- Example
  - $F = xy + x'z$
  - $F(x, y, z) = \Sigma(1, 3, 6, 7)$
  - $F(x, y, z) = \Pi\ (0, 2, 4, \cancel{6})$
    5

Table 2.6
Truth Table for $F = xy + x'z$

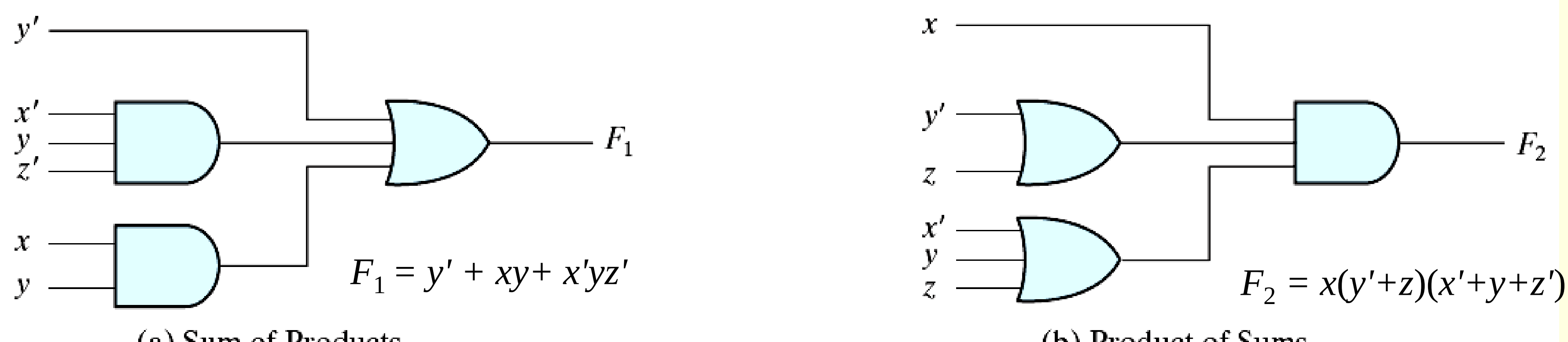
x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Standard Forms

- Canonical forms are very seldom the ones with the least number of literals.
- Standard forms: the terms that form the function may obtain one, two, or any number of literals.
  - Sum of products: $F_1 = y' + xy + x'yz'$
  - Product of sums: $F_2 = x(y'+z)(x'+y+z')$
  - $F_3 = A'B'CD+ABC'D'$

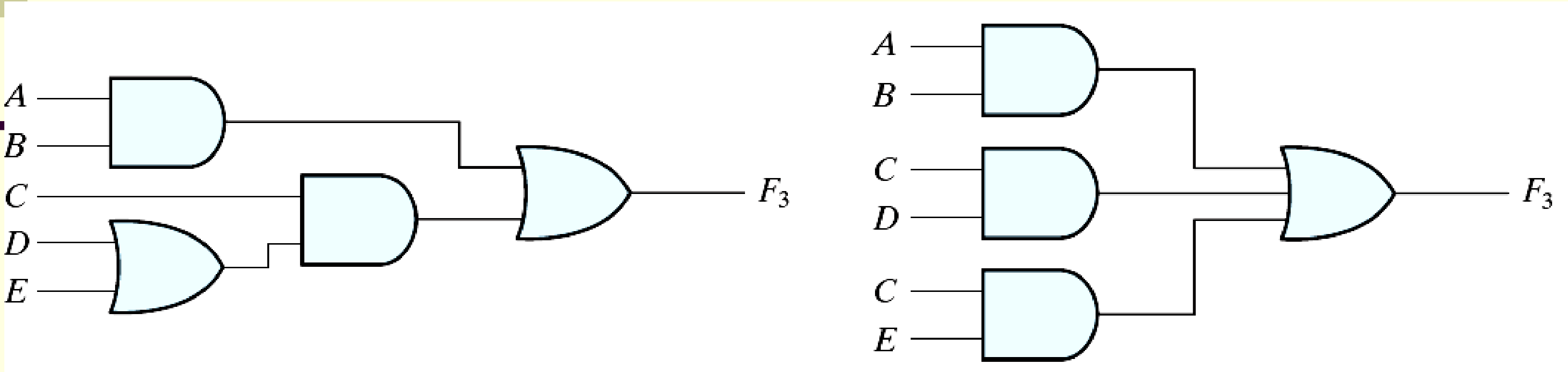
Implementation

- Two-level implementation



(a) Sum of Products (b) Product of Sums

- Multi-level implementation



(a) $AB + C(D + E)$ (b) $AB + CD + CE$

Thanks To

Mustafa Kemal Uyguroğlu