14:332:231
DIGITAL LOGIC DESIGN

Ivan Marsic, Rutgers University

Electrical & Computer Engineering

Fall 2013

Lecture #4: Boolean Algebra, Theorems,
Standard Representation of Logic Functions

Boolean Algebra

- a.k.a. "switching algebra"
  - Deals with Boolean values → 0, 1
- Positive-logic convention
  - Analog voltages LOW, HIGH → 0, 1
- Negative logic -- seldom used
- Signal values denoted by variables (X, Y, FRED, etc)

Boolean Algebra is Just Like Boolean Logic …

- NOT is a *prime* (′):
  - $0' = 1$
  - $1' = 0$
- OR is a *plus* (+):
  - $0 + 0 = 0$
  - $0 + 1 = 1$
  - $1 + 0 = 1$
  - **$1 + 1 = 1$**
- AND is *multiplication dot* (·):
  - $0 \cdot 0 = 0$
  - $0 \cdot 1 = 0$
  - $1 \cdot 0 = 0$
  - $1 \cdot 1 = 1$

Axioms (will lead to Theorems)

- Variable X can take only one of two values:

(A1) $X = 0$ if $X \neq 1$ (A1′) $X = 1$ if $X \neq 0$

- Complement:

(A2) if $X = 0$, then $X' = 1$ (A2′) if $X = 1$ if $X' = 0$

- Three axioms to define the AND and the OR operations:

(A3) $0 \cdot 0 = 0$ (A3′) $1 + 1 = 1$

(A4) $1 \cdot 1 = 1$ (A4′) $0 + 0 = 0$

(A5) $0 \cdot 1 = 1 \cdot 0 = 0$ (A5′) $1 + 0 = 0 + 1 = 1$

Boolean Operators

- Complement: X′ (opposite of X)
- AND: X · Y
- OR: X + Y

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

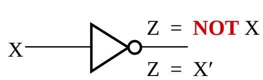
X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

X	NOT X
0	1
1	0

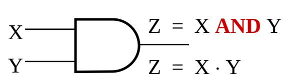
- Axiomatic definition: A1 – A5, A1′ – A5′

binary operators, described functionally by truth table

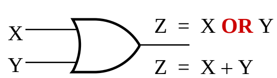
Logic Symbols



NOT (complement)



AND



OR

Duality

- Swap 0 & 1, AND & OR
  - Result: Theorems still true
- Why?
  - Each axiom (A1 – A5) has a dual (A1′ – A5′)

Some Definitions

- *Literal*: a variable or its complement
  - X, X′, FRED′, CS_L
- *Expression*: literals combined by AND, OR, parentheses, complementation
  - X + Y
  - P · Q · R
  - A + B · C
  - ((FRED · Z′) + CS_L · A · B′ · C + Q5) · RESET′
- *Equation*: Variable = Expression
  - P = ((FRED · Z′) + CS_L · A · B′ · C + Q5) · RESET′

Theorems – One Variable

(T1)	$X + 0 = X$	(T1′)	$X \cdot 1 = X$	(Identities)
(T2)	$X + 1 = 1$	(T2′)	$X \cdot 0 = 0$	(Null elements)
(T3)	$X + X = X$	(T3′)	$X \cdot X = X$	(Idempotency)
(T4)	$(X')' = X$			(Involution)
(T5)	$X + X' = 1$	(T5′)	$X \cdot X' = 0$	(Complements)

- Proofs by *perfect induction*
- Axiom (A1) is the key (a variable can take only one of two values: 0 or 1)

Proofs of One-Variable Theorems

(*perfect induction*)

(T3) idempotency:

$X + X = X$	[X=0]	$0+0 = 0$	true, according to (A4′)
	[X=1]	$1+1 = 1$	true, according to (A3′)

(T4) involution:

$(X')' = X$	[X=0]	$(0')' = 1' = 0$	true, according to (A2)
	[X=1]	$(1')' = 0' = 1$	& (A2′)

Etc.

Boolean Operator Precedence

- The order of evaluation is:
  - Parentheses
  - NOT
  - AND
  - OR
- Consequence: Parentheses appear around OR expressions
- Example:

$$F = A \cdot (B + C) \cdot (C + D)$$

Theorems – Two or Three Variables

(T6)	$X + Y = Y + X$	(T6′)	$X \cdot Y = Y \cdot X$	(Commutativity)
(T7)	$(X + Y) + Z = X + (Y + Z)$	(T7′)	$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$	(Associativity)
(T8)	$X \cdot Y + X \cdot Z = X \cdot (Y + Z)$	(T8′)	$(X + Y) \cdot (X + Z) = X + Y \cdot Z$	(Distributivity)
(T9)	$X + X \cdot Y = X$	(T9′)	$X \cdot (X + Y) = X$	(Covering)
(T10)	$X \cdot Y + X \cdot Y' = X$	(T10′)	$(X + Y) \cdot (X + Y') = X$	(Combining)
(T11)	$X \cdot Y + X' \cdot Z + Y \cdot Z = X \cdot Y + X' \cdot Z$			(Consensus)
(T11′)	$(X + Y) \cdot (X' + Z) \cdot (Y + Z) = (X + Y) \cdot (X' + Z)$			

Boolean Algebraic Proof – Example

$X + X \cdot Y = X$ ← Covering Theorem (T9)

Proof Steps:	Justification:
$X + X \cdot Y$	
$= X \cdot 1 + X \cdot Y$	Identity element: $X \cdot 1 = X$ (T1′)
$= X \cdot (1 + Y)$	Distributivity (T8)
$= X \cdot 1$	Null elements (T2): $1 + Y = 1$
$= X$	Identity element (T1′)

Why Theorems and Proofs?

- These theorems are useful ***rules of substitution*** for logic expressions
- Why substitution? —Because we may want to:
  - Design a simpler circuit (faster, easier to implement, cheaper, more reliable)
  - Use different gates for implementation (same reasons)
- Our primary reason for doing proofs is to learn:
  - Careful and efficient use of the identities and theorems of Boolean algebra, and
  - How to choose the appropriate substitution ("theorem") to apply to make forward progress, irrespective of the application

Distributivity (dual)

(T8′)

$$(X + Y) \cdot (X + Z) = X \cdot X + X \cdot Z + Y \cdot X + Y \cdot Z$$
$$= X + X \cdot Z + X \cdot Y + Y \cdot Z = X + X \cdot Y + Y \cdot Z$$
$$= X + Y \cdot Z$$

$(X + Y) \cdot (X + Z) = X + Y \cdot Z$ (Distributivity)

$(3 + 5) \cdot (3 + 7) \neq 3 + 5 \cdot 7$!!!

parentheses, operator precedence!

Consensus Theorem

$X \cdot Y + X' \cdot Z + Y \cdot Z = X \cdot Y + X' \cdot Z$ Consensus (T11)

Proof Steps:	Justification:
$X{\cdot}Y + X'{\cdot}Z + Y{\cdot}Z$	
$= X{\cdot}Y + X'{\cdot}Z + 1 \cdot Y{\cdot}Z$	Identity (T1′)
$= X{\cdot}Y + X'{\cdot}Z + (X + X') \cdot Y{\cdot}Z$	Complement (T5)
$= X{\cdot}Y + X'{\cdot}Z + X{\cdot}Y{\cdot}Z + X'{\cdot}Y{\cdot}Z$	Distributive (T8)
$= X{\cdot}Y + X{\cdot}Y{\cdot}Z + X'{\cdot}Z + X'{\cdot}Z{\cdot}Y$	Commutative (T6)
$= X{\cdot}Y \cdot 1 + X{\cdot}Y{\cdot}Z + X'{\cdot}Z \cdot 1 + X'{\cdot}Z{\cdot}Y$	Identity (T1′)
$= X{\cdot}Y \cdot (1+Z) + X'{\cdot}Z \cdot (1 + Y)$	Distributive (T8)
$= X{\cdot}Y{\cdot}1 + X'{\cdot}Z{\cdot}1$	1+X = 1 (T2)
$= X{\cdot}Y + X'{\cdot}Z$	Identity (T1′)

Theorems for Expressions

The theorems remain valid if a variable is replaced by an expression.

$X \rightarrow U \cdot W$

$U \cdot W + Y \cdot Z = (U \cdot W + Y) \cdot (U \cdot W + Z) =$

$= (U + Y) \cdot (W + Y) \cdot (U + Z) \cdot (W + Z)$ ← distributivity (dual)

$Z \rightarrow X'$

$(X + Y) \cdot (X + X') = X + Y \cdot X' = X + Y$

distributivity (dual)

17 of 23

N-variable Theorems

(T12)	$X + X + \ldots + X = X$	(Generalized idempotency)
(T12′)	$X \cdot X \cdot \ldots \cdot X = X$	
(T13)	$(X_1 \cdot X_2 \cdot \ldots \cdot X_n)' = X_1' + X_2' + \ldots + X_n'$	(DeMorgan's theorems)
(T13′)	$(X_1 + X_2 + \ldots + X_n)' = X_1' \cdot X_2' \cdot \ldots \cdot X_n'$	
(T14)	$[F(X_1, X_2, \ldots, X_n, +, \cdot)]' = F(X_1', X_2', \ldots, X_n', \cdot, +)$	↑ (Generalized DeMorgan's theorem) ↓ (Shannon's expansion theorems)
(T15)	$F(X_1, X_2, \ldots, X_n) = X_1 \cdot F(1, X_2, \ldots, X_n) + X_1' \cdot F(0, X_2, \ldots, X_n)$	
(T15′)	$F(X_1, X_2, \ldots, X_n) = [X_1 + F(0, X_2, \ldots, X_n)] \cdot [X_1' + F(0, X_2, \ldots, X_n)]$	

- Prove using *finite induction*
- Most important: DeMorgan's theorems

18 of 23

DeMorgan's Theorems

Proof by *finite induction*: (basis step, n=2; induction step, n=i $\rightarrow$ n=i+1)

$\mathbf{A} = X_1 + X_2$ $\qquad$ $\mathbf{B} = X_1' \cdot X_2'$ $\qquad$ $(X_1 + X_2)' = X_1' \cdot X_2'$

If $A \cdot B = 0$ and $A + B = 1$ then $A' = B$

$$A \cdot B = (X_1 + X_2) \cdot (X_1' \cdot X_2') = \mathbf{0}$$

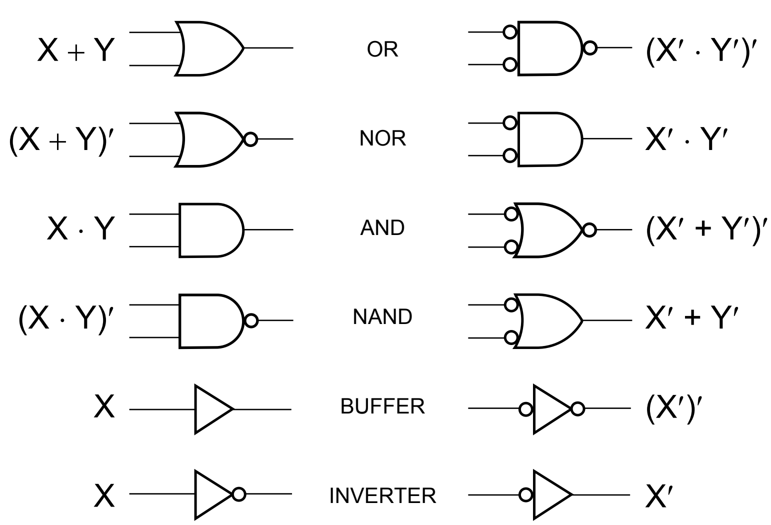
$$\begin{aligned} A + B &= X_1 + X_2 + X_1' \cdot X_2' \\ &= X_1 + X_2 \cdot X_1 + X_2 \cdot X_1' + X_1' \cdot X_2' \\ &= X_1 + X_1' + X_1 \cdot X_2 = \mathbf{1} \end{aligned}$$

} basis step

induction step { assume n = i true, then for n = i + 1

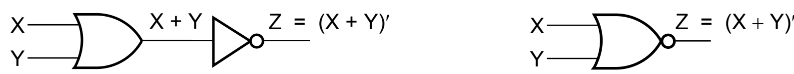
$$(\mathbf{A}_i + X_{i+1})' = \mathbf{B}_i \cdot X_{i+1}'$$

DeMorgan Symbols

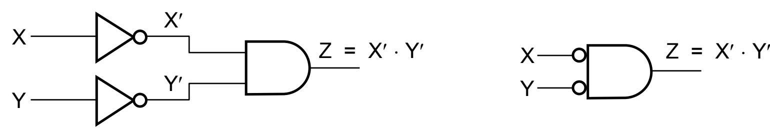


DeMorgan Symbol Equivalence for NOR

NOR



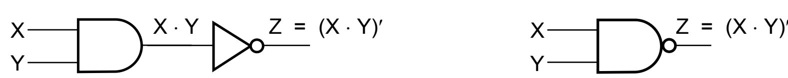
is the equivalent to



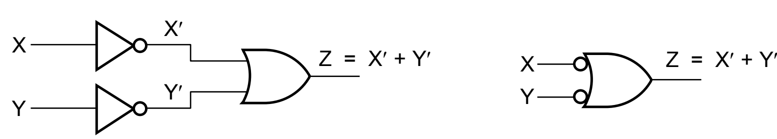
21 of 23

DeMorgan Symbol Equivalence for NAND

NAND



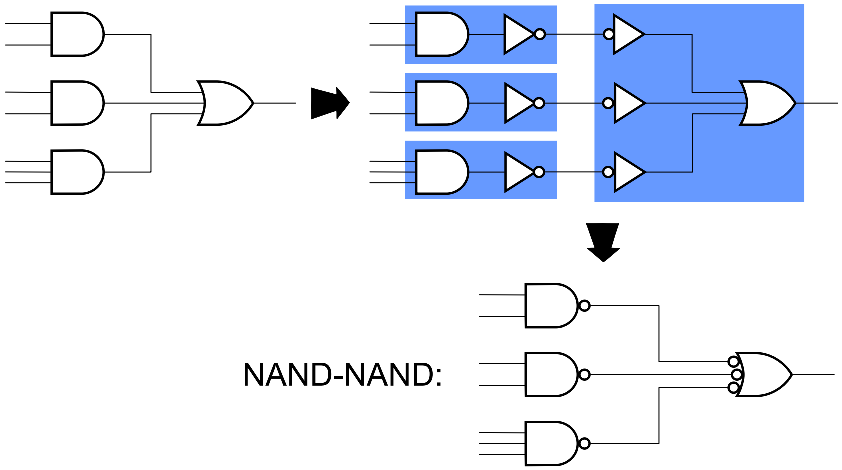
is the equivalent to



22 of 23

Sum-of-Products Form

AND-OR:

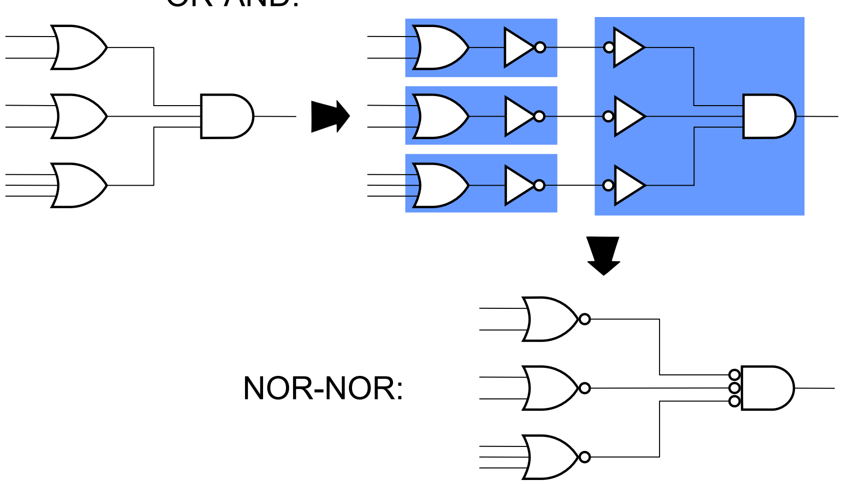


NAND-NAND:

NAND-NAND preferred in TTL technology.

Product-of-Sums Form

OR-AND:



NOR-NOR:

Product-of-sums preferred in CMOS technology.