Digital Logic Design

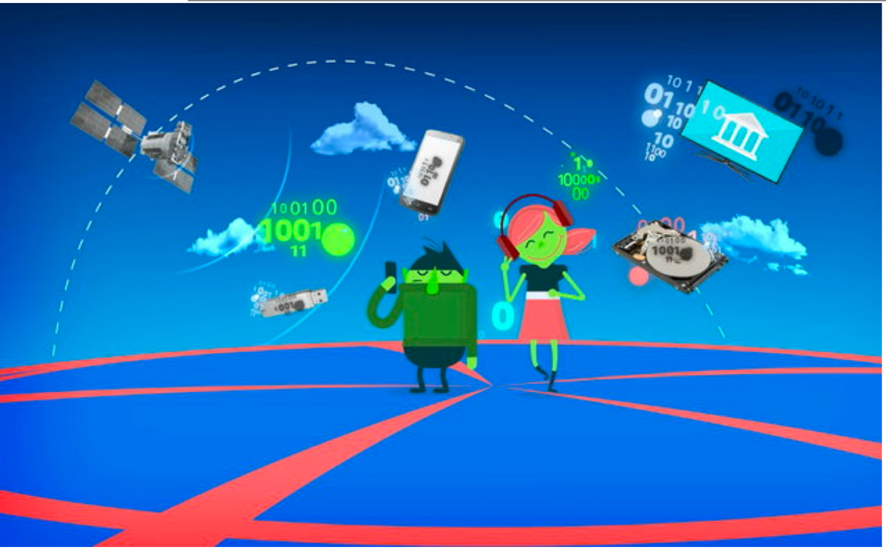
Hajar Falahati

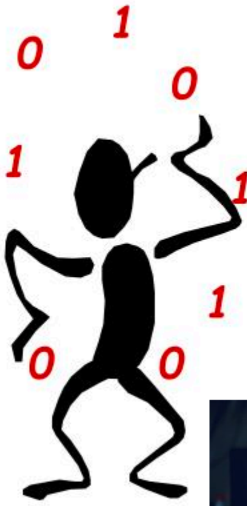
Department of Computer Engineering
IRAN University of Science and Technology

hfalahati@iust.ac.ir

Iran University of Science & Technology
IUST

Binary Streams!





Outline

- Binary Logic and Gates
- Boolean Algebra
- Switching Functions

Binary Logic

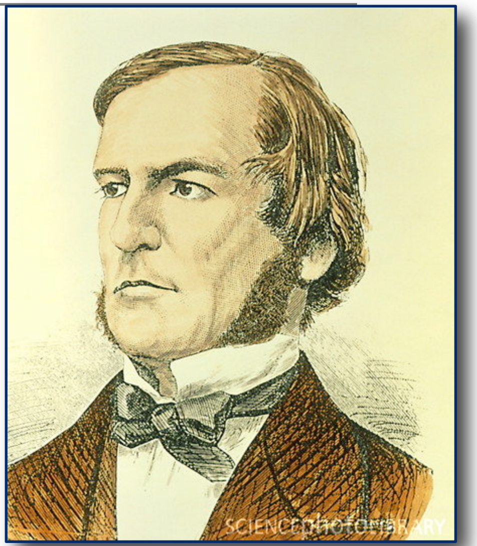
Boolean algebra: Big Picture

- An algebra on binary variables with logical operations
  - Mathematical system
  - Specify and transform logic functions
  - Design and analysis digital systems

- What you start with Axioms
  - Basic things about objects and operations you just assume to be true at the start

- What you derive first
  - Laws and theorems:
    - Allow you to manipulate Boolean expressions
    - …Also allow us to do some simplification on Boolean expressions

- What you derive later
  - More "sophisticated" properties useful for manipulating digital designs represented in the form of Boolean equations



George Boole, "The Mathematical Analysis of Logic," 1847.

Boolean algebra: Elements

- Binary variables
  - True/False, On/OFF, Yes/No, …
  - 1/0

- Logical operators
  - Operate on binary values
  - AND : (a.b)
  - OR : (a+b)
  - NOT : ($\bar{a}$), (a‘), (~a)

- Logical gates
  - Implement logic functions

Logical Operators

- Consider two binary values
  - And
  - OR
  - NOT

a	b	(a . B)	(a + b)	($\bar{a}$), (a‘), (~a)	($\bar{b}$), (b‘), (~b)
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	1
1	1	1	1	0	0

Truth Table

- Tabular listing of **all possible** combinations of argument values of a function

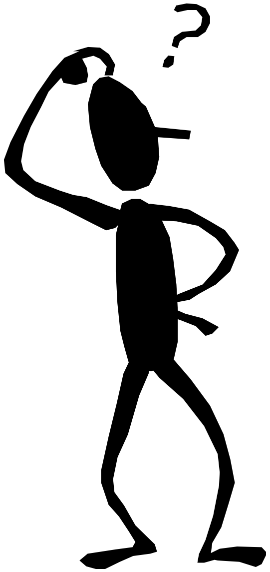
a	b	(a . B)
0	0	0
0	1	0
1	0	0
1	1	1

a	b	(a + b)
0	0	0
0	1	1
1	0	1
1	1	1

a	($\bar{a}$), (a‘), (~a)
0	1
1	0

Logic Function

- F (a, b) = a $\bar{b}$ +$\bar{b}$ a



Logic Function

- F (a, b) = a $\bar{b}$ + $\bar{a}$ b

a	b	$\bar{a}$	$\bar{b}$	(a $\bar{b}$)	($\bar{a}$ b)	a $\bar{b}$ + $\bar{a}$ b
0	0	1	1	0	0	0
0	1	1	0	0	1	1
1	0	0	1	1	0	1
1	1	0	0	0	0	0

Boolean Algebra

Boolean Algebra: Axioms

Formal version | *English version*

Formal version	English version
1. B contains at least two elements, 0 and 1, such that 0 ≠ 1	Math formality...
2. *Closure* $a,b \in B$, *(i)* $a + b \in B$ *(ii)* $a \bullet b \in B$	Result of AND, OR stays in set you start with
3. *Commutative Laws*: $a,b \in B$, (i) (ii)	For primitive AND, OR of 2 inputs, order doesn't matter
4. *Identities*: 0, 1 $\in B$ (i) (ii)	There are identity elements for AND, OR, that give you back what you started with
5. Associativity (i) (ii)	For the same operation, parenthesis order does not matter
5. *Distributive Laws*: (i) (ii)	• distributes over +, just like algebra ...but + distributes over •, also (!!)
6. *Complement*: (i) (ii)	There is a complement element; AND/ORing with it gives the identity elm.

Boolean Algebra: Duality

- Dual
  - Swapping all **operators** against their **counterparts**, + with . and vice versa
    - Every AND operation with... an OR operation
    - Every OR operation with... an AND
  - Swapping the **identity** elements with each other (i.e., 1 by 0 and vice versa)
    - Every constant 1 with... a constant 0
    - Every constant 0 with... a constant 1

- Example: Dual of (a+b).(a+c)
  - (a.b) + (a.c)

- If an expression is **valid** then the **dual** of that expression is also **valid**

Boolean Algebra: Useful Laws

Dual

Operations with 0 and 1:

1. X + 0 = X — 1D. X • 1 = X
2. X + 1 = 1 — 2D. X • 0 = 0

AND, OR with identities gives you back the original variable or the identity

Idempotent Law:

3. X + X = X — 3D. X • X = X

AND, OR with self = self

Involution Law:

4. $\overline{(\overline{X})} = X$

double complement = no complement

Laws of Complementarity:

5. $X + \overline{X} = 1$ — 5D. $X \bullet \overline{X} = 0$

AND, OR with complement gives you an identity

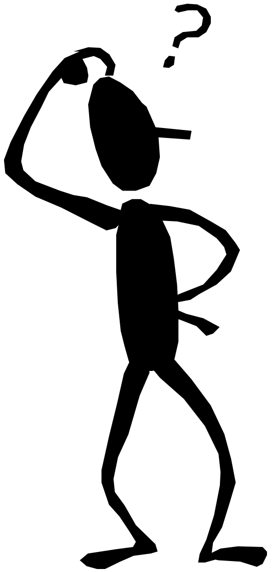
Commutative Law:

6. X + Y = Y + X — 6D. X • Y = Y • X

Just an axiom…

Boolean Algebra: Principle of Duality (cont'd)

- $a + (b.c)$
- $(c + \bar{a}) \cdot b + 0$
- $X \cdot Y + (W + Z)$
- $A \cdot B + A \cdot C + B \cdot C$
- $(A+B).(A+C).(B+C)$



Boolean Algebra: Principle of Duality (cont'd)

- $a + (b.c)$
  - Dual: $a \,.\, (b+c)$
- $(c + \bar{a}) \cdot b + 0$
  - Dual: $(c \,.\, \bar{a}) + b \,.\, 1$
- $X \cdot Y + (W + Z)$
  - Dual: $(X + Y) \,.\, (W \,.\, Z)$
- $A \cdot B + A \cdot C + B \cdot C$
  - Dual: $(A + B) \,.\, (A + C) \,.\, (B + C)$
- $(A+B).(A+C).(B+C)$
  - Dual: $(A.B)+(A.C)+(B.C)$

Boolean Algebra: Operator Precedence

- Order of evaluation
  - Parentheses ()
  - NOT
  - AND
  - OR
- $F = A(B + C)(C + D)$

Boolean Algebra: Fundamental Theorems

- **Theorem 1:** Idempotency
  - a + a = a
  - aa = a
- **Theorem 2:** Null element
  - a + 1 = 1
  - a0 = 0
- **Theorem 3:** Involution
  - $\bar{\bar{a}} = a$

OR	AND	Complement
a + 0 = a	a0 = 0	0' = 1
a + 1 = 1	a1 = a	1' = 0

Boolean Algebra: Fundamental Theorems (cont'd)

- **Theorem 4:** Absorption
  - a + ab = a
  - a(a + b) = a
- **Theorem 5:** Sudo Absorption
  - a + a'b = a + b
  - a(a' + b) = ab
- Examples:
  - (X + Y) + (X + Y)Z = X + Y
  - AB'(AB' + B'C) = AB'
  - B + AB'C'D = B + AC'D
  - (X + Y)((X + Y)' + Z) = (X + Y)Z

Boolean Algebra: Fundamental Theorems (cont'd)

- **Theorem 6:**
  - $ab + ab' = a$
  - $(a + b)(a + b') = a$
- **Theorem 7:**
  - $ab + ab'c = ab + ac$
  - $(a + b)(a + b' + c) = (a + b)(a + c)$
- Examples:
  - $ABC + AB'C = AC$
  - $(x'y' + z)(w + x'y' + z') = (x'y' + z)(w + x'y')$

Fundamentals of Boolean Algebra (6)

- **Theorem 8:** DeMorgan's Theorem
  - (a + b)' = a'b’
  - (ab)' = a' + b’

- Generalized DeMorgan's Theorem
  - (a + b + … z)' = a'b' … z'
  - (ab … z)' = a' + b' + … z’

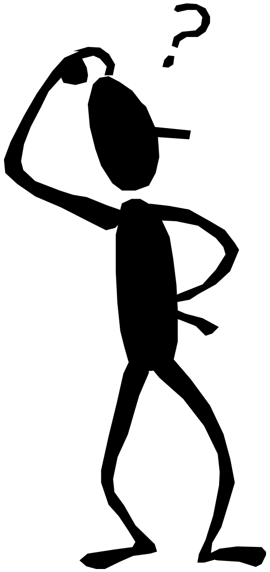
- Examples:
  - (a + bc)' = a'(bc)'
    = a'(b' + c')
    = a'b' + a'c'

Fundamentals of Boolean Algebra (8)

- **Theorem 9:** Consensus
  - ab + a'c + bc = ab + a’c
  - (a + b)(a' + c)(b + c) = (a + b)(a' + c)
- Proof
  - ab + a'c + bc = ab + a'c + bca + bca’ = ab (1 + c) + a’c (1+ b) = ab + a'c
- Examples:
  - AB + A'CD + BCD = AB + A'CD
  - (a + b')(a' + c)(b' + c) = (a + b')(a' + c)

Boolean Algebra: Fundamental Theorems (cont'd)

- $(W' + X' + Y' + Z')(W' + X' + Y' + Z)(W' + X' + Y + Z')(W' + X' + Y + Z) = ?$
- $wy' + wx'y + wxyz + wxz' = ?$
- $(a(b + z(x + a')))' = ?$
- $(a(b + c) + a'b)' = ?$
- $ABC + A'D + B'D + CD = ?$



Boolean Algebra: Fundamental Theorems (cont'd)

- $(W' + X' + Y' + Z')(W' + X' + Y' + Z)(W' + X' + Y + Z')(W' + X' + Y + Z)$

  $= (W' + X' + Y')(W' + X' + Y + Z')(W' + X' + Y + Z)$

  $= (W' + X' + Y')(W' + X' + Y)$

  $= (W' + X')$

- $wy' + wx'y + wxyz + wxz'$

  $= wy' + wx'y + wxy + wxz'$

  $= wy' + wy + wxz'$

  $= w + wxz'$

  $= \mathrm{w}$

Boolean Algebra: Fundamental Theorems (cont'd)

- $(a(b + z(x + a')))'$

$= a' + (b + z(x + a'))'$

$= a' + b' (z(x + a'))'$

$= a' + b' (z' + (x + a')')$

$= a' + b' (z' + x'(a')')$

$= a' + b' (z' + x'a)$

$= a' + b' (z' + x')$

- $(a(b + c) + a'b)'$

$= (ab + ac + a'b)'$

$= (b + ac)'$

$= b'(ac)'$

$= b'(a' + c')$

Boolean Algebra: Fundamental Theorems (cont’d)

- ABC + A'D + B'D + CD = ABC + (A' + B')D + CD [P5(b)]
  = ABC + (AB)'D + CD [T8(b)]
  = ABC + (AB)'D [T9(a)]
  = ABC + (A' + B')D [T8(b)]
  = ABC + A'D + B'D [P5(b)]

Thank You