DIGITAL DESIGN AND COMPUTER ORGANIZATION

Boolean Algebra and Boolean Function

Team DDCO
Department of Computer Science and Engineering

Introduction

Mathematical function: defines a relationship between an independent variable and a dependent variable. E.g. $A = \pi r^2$

- Example: Parabola
- Domain and range are set of real numbers
- Specified on Cartesian Plane

Boolean function: A mathematical function whose arguments, as well as the function itself, assume values from a two-element set (usually $\{0,1\}$).

Introduction

Boolean algebra, is a mathematical system, defined with a set of elements, a set of operators, and a number of unproved axioms or postulates.

A **Boolean function** is a function whose domain and range are the set {0, 1}.

A Boolean function has:

- At least one Boolean variable
- At least one Boolean operator
- At least one input from the set {0, 1}

It produces an output that is also a member of the set {0, 1}

“Digital Design”, M Morris Mano, Section 2.2

Introduction

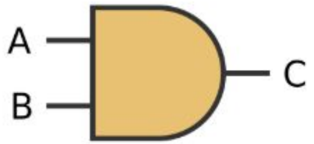
Boolean Constants and Variables

- Inputs and outputs of Boolean function are from the set {0, 1}
    - **0 and 1 are called Boolean constants**
- In general, inputs and outputs of mathematical functions are represented by variables (like $y = x^2$)
    - **Inputs and outputs of Boolean functions are called as Boolean variables (like a, b and y)**

Specified as a truth table:

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

Specified as a logic gate:



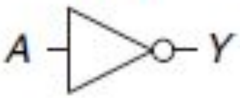
Basic Logic Gates

Logic gates are **physical implementations of Boolean functions.**
A logic gate is a hardware component that performs a specific Boolean function.

<u>Single Input Logic gates</u>

The NOT gate's output is the inverse of its input

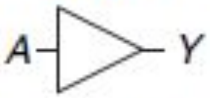
NOT



$Y = \overline{A}$

A	Y
0	1
1	0

BUF



$Y = A$

A	Y
0	0
1	1

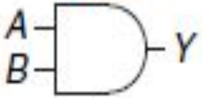
Buffer gate simply copies the input to the output.

Digital Design & Computer Architecture, David Money Harris, Section 1.5

Basic Logic Gates

<u>Two-Input Logic Gates</u>

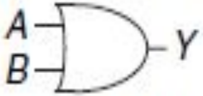
AND



$Y = AB$

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR



$Y = A + B$

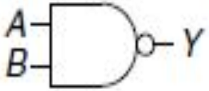
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

XOR

$Y = A \oplus B$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

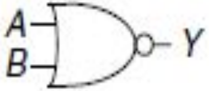
NAND



$Y = \overline{AB}$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

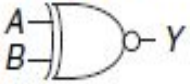
NOR



$Y = \overline{A + B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



$Y = \overline{A \oplus B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Basic Logic Gates

Multiple Input Logic Gates

NOR3

$Y = \overline{A + B + C}$

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Figure 1.20 **Three-input NOR truth table**

AND4

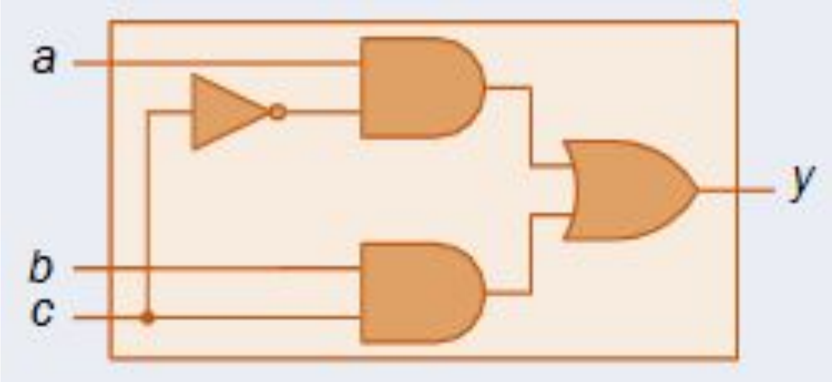
$Y = ABCD$

A	C	B	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Figure 1.22 **Four-input AND truth table**

What is a logic circuit?

Multiple logic gates combined together, with the output of one gate being connected to the input of another, form a ***logic circuit.***



A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

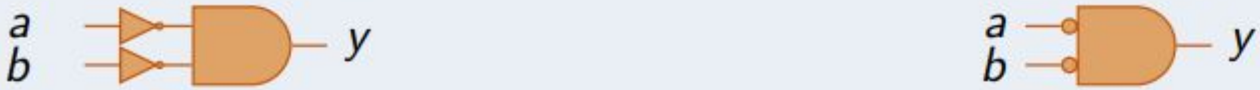
Logic Gates with Inverted Inputs

Bubble

If a logic gates has a NOT gate on an input, the NOT gate can be replaced by a **bubble** on that input

Bubble Example

Bubbled AND Example



Boolean Algebra(T1-Section 2.4)

PES UNIVERSITY

Algebra

In mathematics, an Algebra is composed of four things: a set of elements, operations on those elements, identity elements and laws/identities

Standard Algebra

- **Set** Real numbers
- **Operations** Add, subtract, multiply, divide
- **Identity elements** 0 (for add), 1 (for multiply)
- **Laws/Identities** Commutative, associative, distributive, . . .

Boolean Algebra

- **Set** {0, 1}
- **Operations** AND, OR, NOT
- **Identity elements: 1** (for AND), **0** (for OR)
- **Laws/Identities** Commutative, associative, distributive, . . .

Boolean Laws

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)	$x(x + y) = x$

Useful Identity $a + \bar{a} \cdot b = a + b$ $a \cdot (\bar{a} + b) = a \cdot b$

Boolean Laws

THEOREM 1(a): $x + x = x$.

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

THEOREM 1(b): $x \cdot x = x$.

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

Boolean Laws

THEOREM 2(a): $x + 1 = 1.$

Statement	Justification
$x + 1 = 1 \cdot (x + 1)$	postulate 2(b)
$= (x + x')(x + 1)$	5(a)
$= x + x' \cdot 1$	4(b)
$= x + x'$	2(b)
$= 1$	5(a)

Boolean Laws

THEOREM 6(a): $x + xy = x$.

Statement	Justification
$x + xy = x \cdot 1 + xy$	postulate 2(b)
$= x(1 + y)$	4(a)
$= x(y + 1)$	3(a)
$= x \cdot 1$	2(a)
$= x$	2(b)

Operator Precedence

The operator precedence for evaluating Boolean expressions is

(1) parentheses,

(2) NOT,

(3) AND, and

(4) OR.

In other words, expressions inside parentheses must be evaluated before all other operations. The next operation that holds precedence is the complement, and then follows the AND and, finally, the OR.

Boolean Functions (T1- section 2.5)

Boolean function described by an **algebraic expression** consists of binary variables, the **constants 0 and 1**, and the logic operation symbols

A Boolean function expresses the **logical relationship** between binary variables and is evaluated by determining the binary value of the expression for all possible values of the variables.

EX: $F_1 = x + y'z$

Value of F_1 if $x = 1$ or $y = 0$ and $z = 1$?

Boolean Functions

A Boolean function can be transformed from an algebraic expression into a circuit diagram composed of logic gates connected in a particular structure.

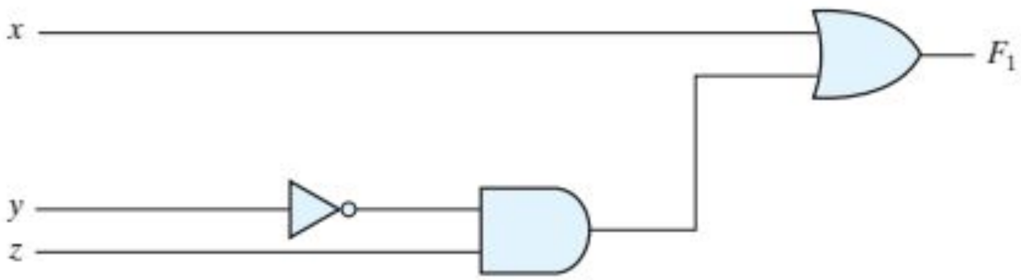


FIGURE 2.1
Gate implementation of $F_1 = x + y'z$

Boolean Functions

For N = Number of variables in the function:

- There exists 2^N possible input combinations
- This results in 2^N rows in the truth table
- Binary numbers counting from 0 through $2^N - 1$
- So, there are 2^{2^N} different truth tables for Boolean functions of N variables.
- 2^{2^N} different Boolean functions

x	y	z	F_1
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

Boolean Functions

There is only one way that a Boolean function can be represented in a truth table. However, when the function is in algebraic form, it can be expressed in a variety of ways, all of which have equivalent logic.

Boolean algebra, it is sometimes possible to obtain a simpler expression for the same function and thus reduce the number of gates in the circuit and the number of inputs to the gate.

Designers are motivated to reduce the complexity and number of gates because their effort can significantly reduce the cost of a circuit.

Boolean Functions

What is simplest form of below equation?

$$F_2 = x'y'z + x'yz + xy'$$

Boolean Functions

What is simplest form of below equation?

$$F_2 = x'y'z + x'yz + xy'$$

$$F_2 = x'y'z + x'yz + xy'$$

$$F_2 = x'y'z + x'yz + xy' = x'z(y' + y) + xy' = x'z + xy'$$

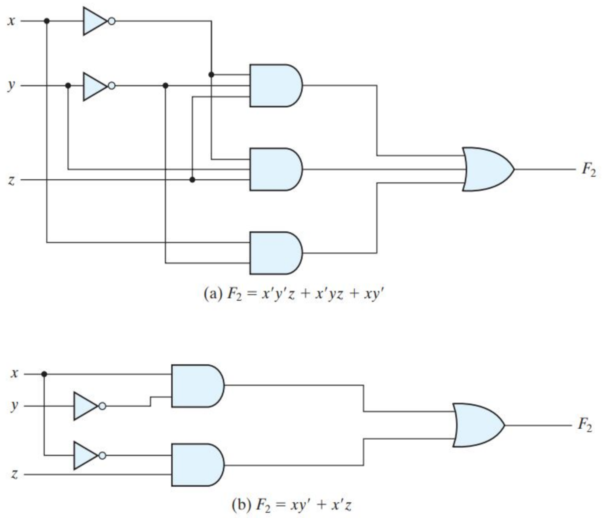


FIGURE 2.2
Implementation of Boolean function F_2 with gates

"Digital Design", M Morris Mano, Section 2.5

Questions

Simplify the following Boolean expressions to a minimum number of literals:

1. $x(x' + y)$
2. $x + x'y$
3. $(x + y)(x + y')$
4. $xy + x'z + yz$
5. $(x + y)(x' + z)(y + z)$

Questions

Simplify the following Boolean functions to a minimum number of literals.

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. $$\begin{aligned} xy + x'z + yz &= xy + x'z + yz(x + x') \\ &= xy + x'z + xyz + x'yz \\ &= xy(1 + z) + x'z(1 + y) \\ &= xy + x'z. \end{aligned}$$

5. $(x + y)(x' + z)(y + z) = (x + y)(x' + z)$, by duality from function 4.

(4) and (5) are together known as consensus theorem.

Complement of a Function

The complement of a function F is F` and is obtained from an interchange of 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 theorems.

$$
\begin{aligned}
(A + B + C)' &= (A + x)' && \text{let } B + C = x \\
&= A'x' && \text{by theorem 5(a) (DeMorgan)} \\
&= A'(B + C)' && \text{substitute } B + C = x \\
&= A'(B'C') && \text{by theorem 5(a) (DeMorgan)} \\
&= A'B'C' && \text{by theorem 4(b) (associative)}
\end{aligned}
$$

Questions

Find the complement of the functions $F_1 = x'yz' + x'y'z$ ***and*** $F_2 = x(y'z' + yz)$.

Questions

Find the complement of the functions F1 = x'yz' + x'y'z and F2 = x(y'z' + yz).

Find the complement of the functions $F_1 = x'yz' + x'y'z$ and $F_2 = x(y'z' + yz)$. By applying DeMorgan's theorems as many times as necessary, the complements are obtained as follows:

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

$$\begin{aligned} F_2' = [x(y'z' + yz)]' &= x' + (y'z' + yz)' = x' + (y'z')'(yz)' \\ &= x' + (y + z)(y' + z') \\ &= x' + yz' + y'z \end{aligned}$$

A simpler procedure for deriving the complement of a function is to take the dual of the function and complement each literal. This method follows from the generalized forms of DeMorgan's theorems. Remember that the dual of a function is obtained from the interchange of AND and OR operators and 1's and 0's.

Questions

Find the complement of the functions $F1 = x'yz' + x'y'z$ and $F2 = x(y'z' + yz)$ taking their duals and complementing each literal.

Questions

Find the complement of the functions $F1 = x'yz' + x'y'z$ and $F2 = x(y'z' + yz)$ taking their duals and complementing 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: $x' + (y + z)(y' + z') = F_2'$.

Boolean Formula

Each Boolean formula means a Boolean function as well as a logic circuit.

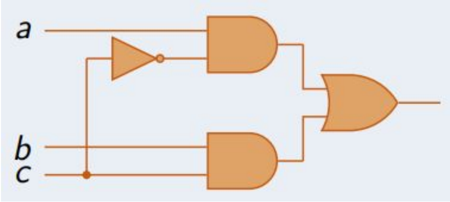
Syntax Rules for Boolean Formulas

1. A Boolean constant (0 or 1) is a Boolean Formula
2. A Boolean variable (say x) is a Boolean formula
3. If P and Q are Boolean formulas then so are:
   a. $(P \cdot Q)$
   b. $(P + Q)$
   c. P'

Example of Boolean Formula: $((a \cdot c' + (b \cdot c))$

- From rule 2, Boolean variable a is a Boolean formula
- From rule 2, Boolean variable b is a Boolean formula
- From rule 2, Boolean variable c' is a Boolean formula
- From rule 3c and step (iii) above c', is a Boolean formula
- From rule 3a, and steps (i) and (iv) above, $(a \cdot c')$ is a Boolean formula
- From rule 3a, and steps (ii) and (iii) above, $(b \cdot c)$ is a Boolean formula
- From rule 3b, and steps (v) and (vi) above,
- $((a \cdot c') + (b \cdot c))$ is a Boolean formula

The Boolean formula can be converted into a combinational logic circuit:



Logic Minimization

From Truth Table to Boolean Formula and its Minimization

- Given a combinational logic circuit or Boolean formula, we have learnt to construct its truth table
- **But, given a truth table, how to construct a Boolean formula (or combinational logic circuit) for it?**
- **Also, as there are multiple Boolean formulas / logic circuits for each truth table, how to pick the minimal one?**
- Above problem is called **logic minimization**
  - many metrics: smallest, fastest, least power cosumption
  - our metric: smallest two level Sum of Products formula
  - may be more than one solution

Questions

- ***Consider Boolean functions of four inputs. What is the number of rows in the truth table of a four input Boolean function?***

- ***What is the total number of Boolean functions of four inputs? Specify your answer as an integer.***

Questions

- ***Consider Boolean functions of four inputs. What is the number of rows in the truth table of a four input Boolean function?***

  A four-input Boolean function's truth table will have 16 rows. (2^4)

- ***What is the total number of Boolean functions of four inputs? Specify your answer as an integer.***

  There are $2^{2^\wedge N}$ different Boolean functions on n Boolean variables

  Here = 65536

Questions

- ***Is a logic gate. . .***
  - ***A Boolean function? or***
  - ***A digital electronic circuit?***

Questions

- ***Is a logic gate. . .***
  - ***A Boolean function? or***
  - ***A digital electronic circuit?***

It is both
This wonderful fact enables us to create the machines that perform mathematics, which we call computers.

Questions

➢ *There are 16 different truth tables for Boolean functions of two variables. List each truth table. Give each one a short descriptive name (such as OR, NAND, and so on).*

A	B	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	YA	YB	YC	YD	YE	YF
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Questions

➢ ***There are 16 different truth tables for Boolean functions of two variables. List each truth table. Give each one a short descriptive name (such as OR, NAND, and so on).***

A	B	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	YA	YB	YC	YD	YE	YF
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
		Zeros	NOR	A' B	NOT A	AB'	NOT B	XOR	NAND	AND	XNOR	B	A' +B	A	A+B'	OR	Ones

Questions

- simplify the Boolean expression:

  $x(x' + y)$

A. 1

B. $x + y$

C. $x \cdot y$

D. y

Questions

- simplify the Boolean expression:

  $x(x' + y)$

A. 1

B. $x + y$

C. $x \cdot y$

D. y

Ans: C

THANK YOU

Team DDCO
Department of Computer Science and Engineering