BOOLEAN ALGEBRA

9/30/2019

INTRODUCTION

Modern algebra is centered around the concept of an algebraic system: A, consisting of a *set* of *elements*: a_i, i=1, 2, …, which are combined by a set of *operations* such as addition, subtraction, or multiplication.

Some important mathematical sets are:

Z: the set of all integers, positive, negative, or zero.
Q: the set of all rational numbers.
R: the set of all real numbers.
C: the set of all complex numbers.

These sets have an infinite number of elements. Some infinite sets are *countable* and can be described by indicating how their elements can be enumerated in a certain way. For example, the set of all natural numbers can be expressed as:

$$P = \{1, 2, 3, \dots \},$$

The set of all non-negative integers can be written as:

$$N = \{0, 1, 2, 3, \dots \},$$

And the set of all integers can be written as:

$$Z = \{0, \pm 1, \pm 2, \pm 3, \dots\}$$

When an element of a set belongs to a set S, we express this fact by stating that: "a is a member of S" and write it as:

$$a \in S .$$

If it does not belong to S we write:

$$a \notin S .$$

SUBSETS

A set S whose elements all belong to a set T is called: a subset of T. That is, the fact that $a \in S$ implies that $a \in T$. This relation between S and T is expressed as:

$$S \subset T ,$$

or: "S is contained or included in T." This inclusion relation has the following properties:

Reflexive law: $S \subset S$ for any set S (1)

Transitive law: $S \subset T$ and $T \subset U$ imply $S \subset U$ (2)

Since a set is completely determined by its elements, two sets are equal if they have exactly the same elements:

S=T means that $x \in S$ if and only if $x \in T$,
or:
S=T iff $S \subset T$ and $T \subset S$.

The "iff" symbol implies "a necessary and sufficient" condition.

Sets can be specified in the following way:

$$S=\{x \in U \mid P(x)\} \text{ or: } S=\{x \mid P(x)\} \qquad (3)$$

which states that: "S is the set of all members x of U such that the condition P(x) holds."
For example, to express the set of all positive integers, we can write:

$$P = \{x \mid x \in Z \text{ and } x>0\} \qquad (4)$$

P(x) is a basic axiom of logic which determines the properties of a given set, and is called: "The axiom of specification."
The "power set" of U is the set of all subsets of U. It contains the null or void set $\varnothing$ and U itself. If U has n > 1 elements, the power set contains a number of 2^n "proper subsets" S satisfying:

$$\varnothing \subset S \subset U, S \neq \varnothing \text{ and } S \neq U \qquad (5)$$

For instance, the power set of the set U = {a, b, c} is:

$$\Pi(U) = [\{a\}, \{b\}, \{c\}, \{a , b\}, \{a , c\}, \{b, c\}, \{a , b, c\}, \varnothing\} \qquad (6)$$

BINARY OPERATIONS, VENN DIAGRAMS

A "binary operation" on a set S is a rule which assigns each ordered pair (a, b) of elements of S some "value" in S. The operation + or arithmetic on the set Z is a binary operation, as well as subtraction and multiplication. Logically, a binary operation: (4 – 2) is distinct from a unary operation (- 2). A unary operation on S involves just one operand, whereas a binary operation involves two operands.

On the subsets of a given set U there are two basic binary operations and one unary operation. There are the Boolean or set-theoretic operations on intersection, union, and complement.

The intersection of two subsets R and S is defined as:

$$R \cap S = \{x \mid x \in R \text{ and } x \in S\} \tag{7}$$

The union of the two subsets R and S is the set of all those elements of U which belong either to R or to S or to both:

$$R \cup S = \{x \mid x \in R \text{ or } x \in S\} \tag{8}$$

The Venn diagrams of Fig. 1 illustrate graphically the meaning of the union and intersection operations.

As an example if we consider:

$E = \{0, \pm 2, \pm 4, \ldots\}$, the set of all even integers,

$P = \{1, 2, 3, 4, 5, 6, \ldots\}$, the set of all positive integers,

then:

$E \cap P = \{2, 4, 6, \ldots\}$, the set of all positive even integers,

$E \cup P = \{0, 1, \pm 2, 3, \pm 4, \ldots\}$, the set of all integers which are not both negative and odd, (not -1, -3, -5, ...)

If n(S) is the number of elements in S, n(R) is the number of elements in R, where R and S are finite sets, then the number of elements in the union of R and S is:

$$n(R \cup S) = n(R) + n(S) - n(R \cap S), \tag{9}$$

where we have subtracted those elements that are counted twice. For instance, if we consider:

$R = \{1, 2, 3, 4, 5\}$

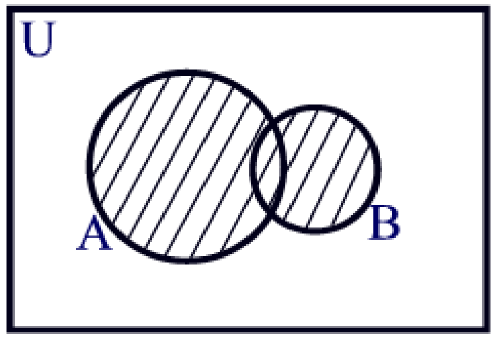
$S = \{4, 5, 6, 7\}$

$R \cup S = \{1, 2, 3, 4, 5, 6, 7\}$

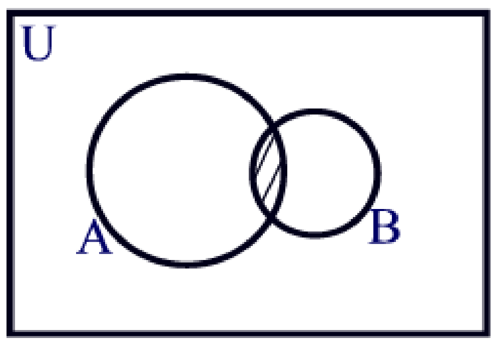
$R \cap S = \{4, 5\}$

then:

$n(R \cup S) = n(R) + n(S) - n(R \cap S) = 5 + 4 - 2 = 7.$



Shaded area is $A \bigcup B$.



Shaded area is $A \bigcap B$.

Figure 1. The Venn diagrams for the union and intersection operations.

When:

$$R \bigcap S = \varnothing ,$$

R and S are said to be "disjoint." A division of a set L into a family of disjoint subsets is called a "partition" of L as shown in Fig. 2.

The complement $\overline{S}$ of S in U is defined as:

$$\overline{S} = \{x \in U | x \notin S\}, \tag{10}$$

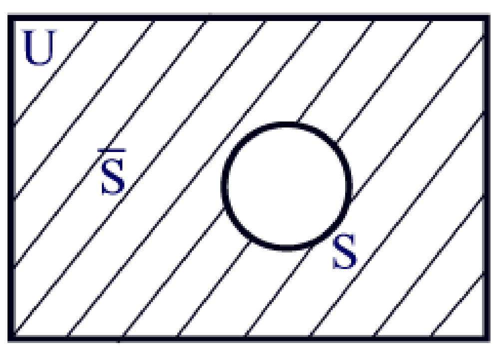
as shown in Fig. 2. Two sets S, T are complementary when:

$$T = \bar{S} \quad \text{or} \quad \bar{T} = S. \tag{11}$$

We also have the Boolean equations:

$$S \cap \bar{S} = \varnothing \quad \text{and} \quad S \cup \bar{S} = U. \tag{12}$$

$R \cap S = \varnothing$



Shaded area is $\bar{S}$.

Figure 2. Venn diagrams for disjoint sets and the complementary set.

BOOLEAN ALGEBRAS

A "Boolean Algebra":

$B = \{A, \wedge, \vee, \bar{\ }, 0, I\}$

is a set A with:

1. Two binary operations: $\{\wedge, \vee\}$: {Wedge, Vee}: {Meet, Join}: {Product, Sum}
2. One unary operation: $\bar{\ }$
3. Two universal bounds: 0, I

such that for all $x, y, z \in A$, the following axioms or laws apply:

L1: Idempotent: $x \wedge x = x, \quad x \vee x = x$

L2: Commutative: $x \wedge y = y \wedge x, \quad x \vee y = y \vee x$

L3: Associative: $x \wedge (y \wedge z) = (x \wedge y) \wedge z$

$x \vee (y \vee z) = (x \vee y) \vee z$

L4: Absorption: $x \wedge (x \vee y) = x$

$x \vee (x \wedge y) = x$

L5a: Modularity: $x \wedge [y \vee (x \wedge z)] = (x \wedge y) \vee (x \wedge z)$

L5b: $x \vee [y \wedge (x \vee z)] = (x \vee y) \wedge (x \vee z)$

L6: Distributive: $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$

$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$

L7: Universal Bounds: $x \wedge 0 = 0, \quad x \vee 0 = x$

$x \wedge I = x, \quad x \vee I = I$

L8: Complements: $x \wedge \bar{x} = 0, \quad x \vee \bar{x} = I$

L9: Involution: $\overline{(\bar{x})} = x$

L10: de Morgan: $\overline{(x \wedge y)} = \bar{x} \vee \bar{y}$

$\overline{(x \vee y)} = \bar{x} \wedge \bar{y}$

Here we have defined five operations: two binary, one unary, and two zero-ary. Then we listed 21 identities grouped into 10 "axioms," "laws," or "postulates." An algebraic system qualifies as a Boolean algebra if and only if it has two binary, one unary, and two zero-ary operations which satisfy the postulated identities. Boolean algebras could represent sets, logical propositions, or mathematical descriptions of electronic hardware of gating networks.

EXAMPLE OF A BOOLEAN ALGEBRA

The "two-element" Boolean algebra is:

$$B\{[0,1], \wedge, \vee, \bar{\ }, 0, 1\}$$

where: $\wedge$ means the lesser of,

$\vee$ means the greater of,

$\bar{\ }$ means the opposite of.

For example:

$$0 \wedge 1 = 0,$$
$$0 \vee 1 = 1,$$
$$\bar{0} = 1$$

For this Boolean algebra, the following operation or truth tables apply:

$\bar{\ }$	
0	1
1	0

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

BINARY STATES

Digital computers are composed of logical circuitry that is "bistable" in its operation. The reason is purely a technical one: circuits can be operated faster if they are at the two extremes of their operating points. This also makes circuits more reliable since operation at the extreme points is not affected by minor characteristics of the circuit. Thus computer logical circuits and signals are two-valued, with 1 representing a positive electrical signal, and 0 is used to represent the zero or "ground" signal level.

Normally a threshold is set. For instance, 0.75 volt is used for resistor-transistor logic, and 1.5 volts for diode-transistor logic. Signals below that threshold are assigned the value 0, and those above it are assigned the value 1 (Fig. 3).

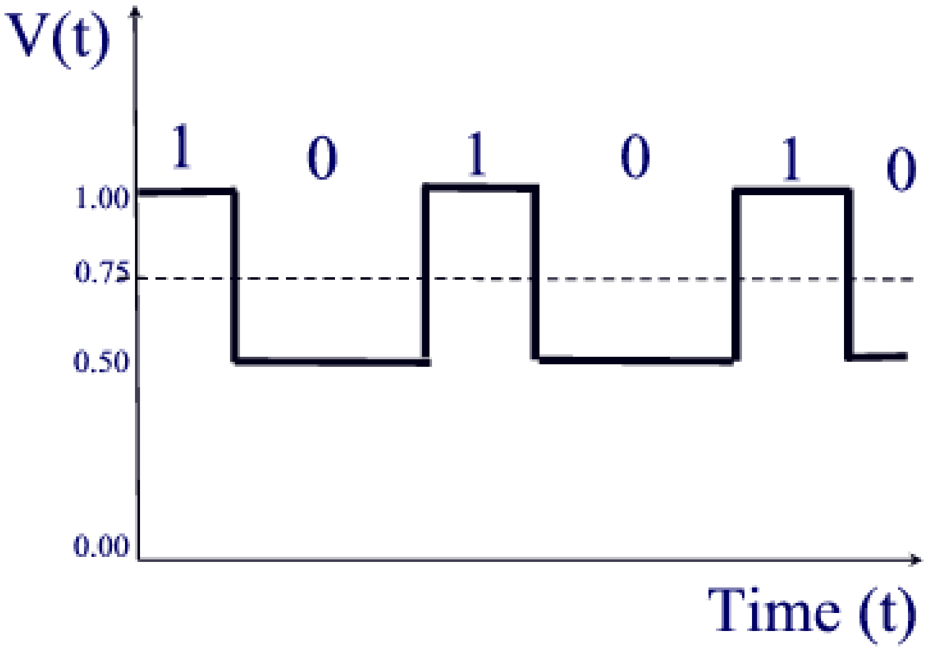


Figure 3. Current voltage levels and associated binary states.

The essential point is that signals in computers and logic circuits are binary, and consequently the variables in the corresponding mathematics will be two-valued too.

Mathematically, the state of any machine or automaton can be specified in the following way:

1. Identify r bistable devices in the machine.
2. Enumerate in any order the devices of which the machine is composed-of as:

$$\ell_1, \ell_2, \ell_3, \ldots, \ell_r$$

3. Assign to each stable state of the machine, or to each one of its components a "state vector;"

$$\bar{x} = \{x_1, x_2, x_3, \ldots, x_r\}$$

by giving each x_i the value 1 or 0 according to whether the device ℓ_i is in the state labeled 1 or the state labeled 0.

The stable states of any elements can also be represented by "on" or "off" and by "true" or "false."

GATING NETWORKS OR LOGICAL GATES

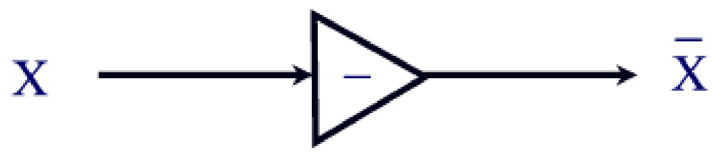
A "gating network" or "logical gate" is a special kind of logical network designed to produce from n input signals from memory elements in states:

$$X_1, X_2, X_3, \dots, X_n \quad ,$$

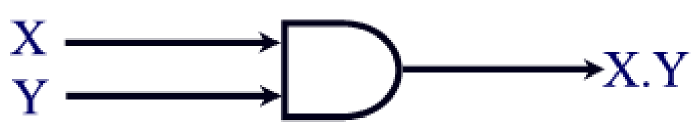
one or more specified binary outputs:

$$F(X_1, X_2, X_3, \dots, X_n) \quad ,$$

so as to realize in each output some specified function of the inputs. Boolean algebra provides a convenient symbolism for their description.



Inverter



AND Gate

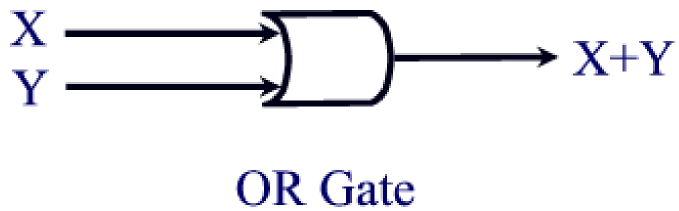


Figure 4. Representation of logic gates.

The three basic Boolean operations of:

$$(\bar{\ },\wedge,\vee),$$

are affected by the three types of gates shown in Fig. 4:

1. The inverter corresponds to $\bar{\ }$,
2. The AND gate corresponds to $\wedge$,
3. The OR gate corresponds to $\vee$.

Their representation is specified by the United States of America Standards Institute and the Military Standards Specifications. Further + is used instead of $\vee$ and $\wedge$ is not shown explicitly or replaced by a period.

The inverter has the following function: if the input signal X is a level 0, the output signal is at level 1, or vice versa. Thus if the input is X, the output is $\overline{X}$.

In the AND gate, the output of the gate will be 1 if both inputs X and Y to the gate are 1's.

In the OR gate, the output signal from this gate will be 1 unless both the input signals are 0's.

For this Boolean algebra, the following operation or truth tables thus apply:

$\bar{\ }$	
0	1
1	0

AND	0	1
0	0	0
1	0	1

OR	0	1
0	0	1
1	1	1

In a related exclusive OR gate designated as XOR, the truth table is:

XOR	0	1
0	0	1
1	1	0

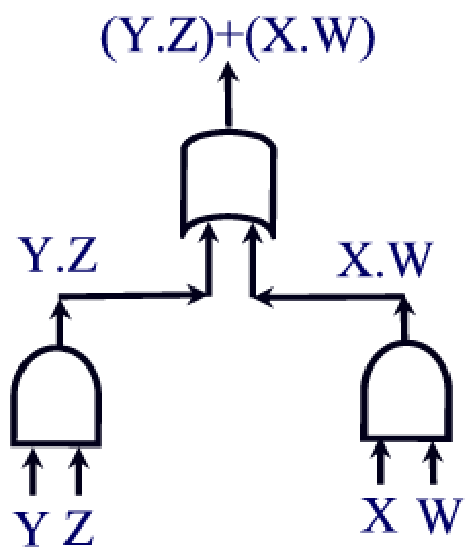


Figure 5. The AND-to–OR gating network.

Combination of gates forms gating networks. The AND-to-OR gating network is shown in Fig. 5. The output function in this case is:

$$(Y \wedge Z) \vee (X \wedge W)$$
$$or:$$
$$(Y.Z) + (X.W)$$

Consequently, if the inputs to:

$$(Y\ AND\ Z)\ OR\ (X\ AND\ W)$$

are 1's, the output from the gating network will represent 1. Otherwise, it will be 0.

TABLES OF COMBINATIONS

For a given combinatorial network if we have n input variables, we can have 2n different input values. It is possible to write down each output value of the network corresponding to the value of the Boolean algebra expression for a given value of the input variables. A table listing the output value next to each possible set of input values thus completely describes the operation of the network.

The analysis of gating networks thus involves the derivation of the Boolean algebra expression for each combination of the input variables.

In Fig. 6, we consider the n = 3 system having $2^3 = 8$ possible input values, with its associated table of combinations shown in Table 1.

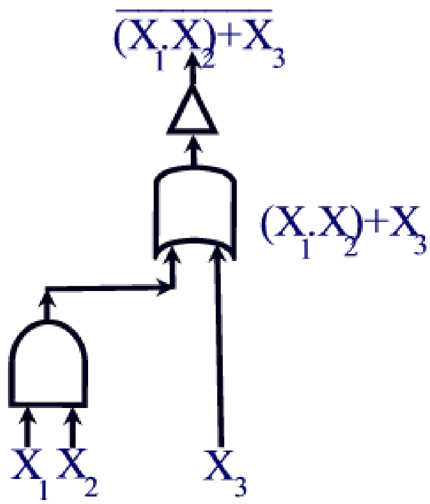


Figure 6. Gating network.

Table 1. Table of combinations of gating network.

Inputs				*Output*
X_1 X_2 X_3	$X_1.X_2$	$(X_1.X_2)+X_3$		$\overline{(X_1.X_2)+X_3}$
0 0 0	0	0		1
0 0 1	0	1		0
0 1 0	0	0		1
0 1 1	0	1		0
1 0 0	0	0		1
1 0 1	0	1		0
1 1 0	1	1		0
1 1 1	1	1		0

CONCLUSION

Boolean algebra applies to logic as well as to sets, and this forms the basis to its application to the Fault Tree Analysis methodology. When systems are too complex to model using the methods of differential and integral calculus, the Algebra of Logic provides us with an exact mathematical method for modeling and analyzing the behavior of such complex systems.

EXERCISES

1. Use Venn diagrams to prove the L10 de Morgan law or axiom of a Boolean Algebra.
2. Graph then construct a table of combinations for the gating network given by the Boolean expression: $(X_1+X_2).X_3$.