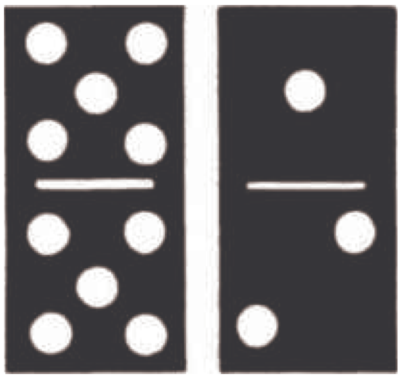
Chapter 13



BOOLEAN ALGEBRA

GOALS

In this section we will develop an algebra that is particularly important to computer scientists, as it is the mathematical foundation of computer design, or switching theory. This algebra is called Boolean algebra after the mathematician George Boole (1815-64). The similarities of Boolean algebra and the algebra of sets and logic will be discussed, and we will discover special properties of finite Boolean algebras.

George Boole, 1815 - 1864

In order to achieve these goals, we will recall the basic ideas of posets introduced in Chapter 6 and develop the concept of a lattice, which has applications in finite-state machines.

The reader should view the development of the topics of this chapter as another example of an algebraic system. Hence, we expect to define first the elements in the system, next the operations on the elements, and then the common properties of the operations in the system.

13.1 Posets Revisited

From Chapter 6, Section 3, we recall the following definition:

***Definition**: Poset. A set L on which a partial ordering relation (reflexive, antisymmetric, and transitive) r is defined is called a partially ordered set, or poset, for short.*

We recall a few examples of posets:

(1) $L = \mathbb{R}$ and r is the relation $\leq$.

(2) $L = \mathcal{P}(A)$ where $A = \{a,\ b\}$ and r is the relation $\subseteq$.

(3) $L = \{1,\ 2,\ 3,\ 6\}$ and r is the relation $\mid$ (divides). We remind the reader that the pair $(a,\ b)$ as an element of the relation r can be expressed as $(a,\ b) \in r$, or $a\, r\, b$, depending on convenience and readability.

The posets we will concentrate on in this chapter will be those which have maxima and minima. These partial orderings resemble that of $\leq$ on $\mathbb{R}$, so the symbol $\leq$ is used to replace the symbol r in the definition of a partially ordered set. Hence, the definition of a poset becomes:

***Definition**: Poset. A set on which a partial ordering, $\leq$, is defined is called a partially ordered set, or, in brief, a poset. Here, $\leq$ is a partial ordering on L if and only if for all $a,\ b,\ c \in L$:*

(1) $a \leq a$(*reflexivity*),

(2) $a \leq b$ *and* $b \leq a \Rightarrow a = b$ (*antisymmetry*), *and*

We now proceed to introduce maximum and minimum concepts. To do this, we will first define these concepts for two elements of the poset L, and then define the concepts over the whole poset L.

***Definition**: Lower Bound, Upper Bound. Let $a,\ b \in L$, a poset. Then $c \in L$ is a lower bound of a and b if $c \leq a$ and $c \leq b$. $d \in L$ is an upper bound of a and b if $a \leq d$ and $b \leq d$.*

***Definition**: **Greatest Lower Bound.** Let L be a poset and $\leq$ be the partial ordering on L. Let $a,\ b \in L$, then $g \in L$ is a greatest lower bound of a and b, denoted glb(a, b), if and only if*

- $g \leq a$,
- $g \leq b$, *and*
- *if* $g' \in L$ *such that if* $g' \leq a$ *and* $g' \leq b$, *then* $g' \leq g$.

The last condition says, in other words, that if g' is also a lower bound, then g is "greater" than g', so g is a greatest lower bound.

The definition of a least upper bound is a mirror image of a greatest lower bound:

***Definition: Least Upper Bound.** Let L be a poset and $\leq$ be the partial ordering on L. Let $a,\ b \in L$, then $\ell \in L$ is a least upper bound of a and b, denoted lub(a, b), if and only if*

- $a \leq \ell$,
- $b \leq \ell$, *and*
- *if* $\ell' \in L$ *such that if* $a \leq \ell'$ *and* $b \leq \ell'$, *then* $\ell \leq \ell'$.

Notice that the two definitions above refer to "...a greatest lower bound" and "a least upper bound." Any time you define an object like these you need to have an open mind as to whether more than one such object can exist. In fact, we now can prove that there can't be two greatest lower bounds or two least upper bounds.

***Theorem 13.1.1.** Let L be a poset and $\leq$ be the partial ordering on L, and $a,\ b \in L$. If a greatest lower bound of a and b exists, then it is unique. The same is true of a least upper bound, if it exists.*

Proof: Let g and g' be greatest lower bounds of a and b. We will prove that $g = g'$.

(1) g a greatest lower bound of a and $b \Rightarrow g$ is a lower bound of a and b.

(2) g' a greatest lower bound of a and b and g a lower bound of a and $b \Rightarrow g \leq g'$ by the definition of greatest lower bound.

(3) g' a greatest lower bound of a and $b \Rightarrow g'$ is a lower bound of a and b.

(4) g a greatest lower bound of a and b and g' a lower bound of a and $b \Rightarrow g' \leq g$ by the definition of greatest lower bound.

(5) $g \leq g'$ and $g' \leq g \Rightarrow g = g'$ by the antisymmetry property of a partial ordering.

The proof of the second statement in the theorem is almost identical to the first and is left to the reader. ■

***Definition: Greatest Element, Least Element.** Let L be a poset. $M \in L$ is called the greatest (maximum) element of L if, for all $a \in L$, $a \leq M$. In addition, $m \in L$ is called the least (minimum) element of L if for all $a \in L$, $m \leq a$.*

Note: The greatest and least elements, when they exist, are frequently denoted by 1 and 0 respectively.

Example 13.1.1. Let $L = \{1,\ 3,\ 5,\ 7,\ 15,\ 21,\ 35,\ 105\}$ and let $\leq$ be the relation $\mid$ (divides) on L. Then L is a poset. To determine the *lub* of 3 and 7, we look for all $\ell \in L$, such that $3 \mid \ell$ and $7 \mid \ell$. Certainly, both $\ell = 21$ and $\ell = 105$ satisfy these conditions and no other element of L does. Next, since $21 \mid 105$, then $21 = \text{lub}(3, 7)$. Similarly, the *lub*$(3, 5) = 15$. The greatest element of L is 105 since $a \mid 105$ for all $a \in L$. To find the *glb* of 15 and 35, we first consider all elements g of L such that $g \mid 15$ and $g \mid 35$. Certainly, both $g = 5$ and $g = 1$ satisfy these conditions. But since $1 \mid 5$, then *glb*$(15, 35) = 5$. The least element of L is 1 since $1 \mid a$ for all $a \in L$.

Henceforth, for any positive integer n, D_n will denote the set of all positive integers which are divisors of n. For example, the set L of Example 13.1.1 is D_{105}.

Example 13.1.2. Consider the poset $\mathcal{P}(A)$, where $A = \{a, b, c\}$, with the relation $\subseteq$ on $\mathcal{P}(A)$. The *glb* of the $\{a, b\}$ and $\{a, c\}$ is $g = \{a\}$. For any other element g' of M which is a subset of $\{a, b\}$ and $\{a, c\}$ (there is only one; what is it?), $g' \subseteq g$. The least element of $\mathcal{P}(A)$ is $\emptyset$ and the greatest element of $\mathcal{P}(A)$ is $A = \{a, b, c\}$. The Hasse diagram of $\mathcal{P}(A)$ is shown in Figure 13.1.1.

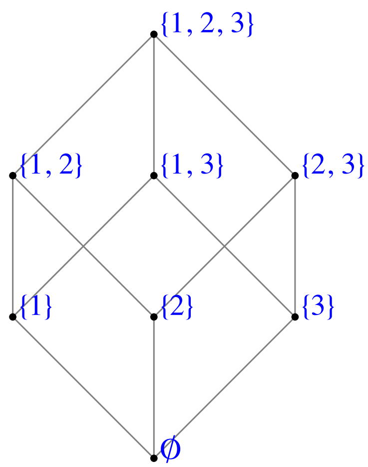


Figure 13.1.1
Example 13.1.2

With a little practice, it is quite easy to find the least upper bounds and greatest lower bounds of all possible pairs in $\mathcal{P}(A)$ directly from the graph of the poset.

The previous examples and definitions indicate that the *lub* and *glb* are defined in terms of the partial ordering of the given poset. It is not yet clear whether all posets have the property such every pair of elements has both a *lub* and a *glb*. Indeed, this is not the case (see Exercise 3).

EXERCISES FOR SECTION 13.1

A Exercises

1. Let $D_{30} = \{1, 2, 3, 5, 6, 10, 15, 30\}$ and let the relation | be a partial ordering on D_{30}.

   (a) Find all lower bounds of 10 and 15.

   (b) Find the *glb* of 10 and 15.

   (c) Find all upper bounds of 10 and 15.

   (d) Determine the *lub* of 10 and 15.

   (e) Draw the Hasse diagram for D_{30} with |. Compare this Hasse diagram with that of Example 13.1.2. Note that the two diagrams are structurally the same.

2. List the elements of the sets D_8, D_{50}, and D_{1001}. For each set, draw the Hasse diagram for "divides."

3. Figure 13.1.2 contains Hasse diagrams of posets.

   (a) Determine the *lub* and *glb* of all pairs of elements when they exist. Indicate those pairs that do not have a *lub* (or a *glb*).

   (b) Find the least and greatest elements when they exist.

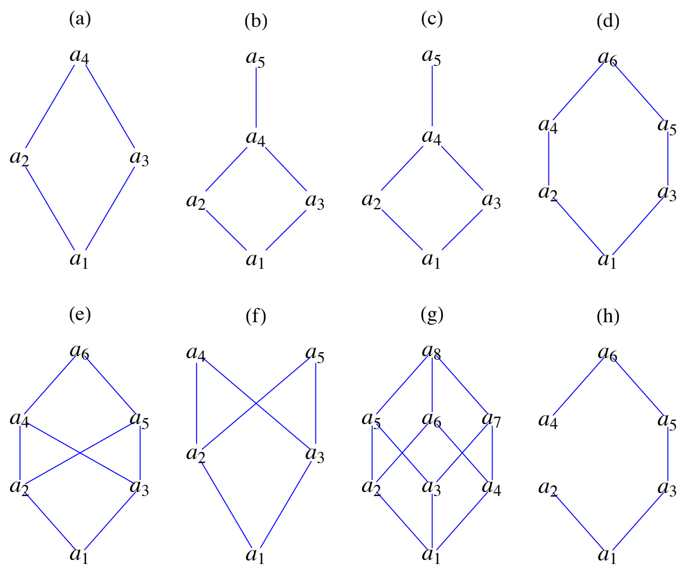


Figure 13.1.2
Exercise 3

4. For the poset $(\mathbb{N}, \leq)$, what are $glb(a, b)$ and $lub(a, b)$? Are there least and/or greatest elements?

5. (a) Prove the second part of Theorem 13.1.1, the least upper bound of two elements in a poset is unique, it one exists.

(b) Prove that if a poset L has a least element, then that element is unique.

6. We naturally order the numbers in $A_m = \{1, 2, \ldots, m\}$ with "less than or equal to," which is a partial ordering. We may order the elements of $A_m \times A_n$ by $(a, b) \leq (a', b') \Leftrightarrow a \leq a'$ and $b \leq b'$.

(a) Prove that this defines a partial ordering of $A_m \times A_n$.

(b) Draw the ordering diagrams for $\leq$ on $A_2 \times A_2$, $A_2 \times A_3$, and $A_3 \times A_3$.

(c) What are $glb((a, b), (a', b'))$ and $lub((a, b), (a', b'))$?

(d)Are there least and/or greatest elements in $A_m \times A_n$?

13.2 Lattices

In this section, we restrict our discussion to *lattices*, those posets where every pair of elements has a *lub* and a *glb*. We first introduce some notation.

Definitions: Join, Meet. *Let L be a poset under an ordering* $\leq$. *Let* $a, b \in L$. *We define:*

$a \vee b$ (read "a join b") as the least upper bound of a and b, and

$a \wedge b$ (read "a meet b") as greatest lower bound of a and b.

Since the join and meet operations produce a unique result in all cases where they exist, by Theorem 13.1.1, we can consider them as binary operations on a set if they aways exist. Thus the following definition:

Definition: Lattice. *A lattice is a poset L (under $\leq$) in which every pair of elements has a lub and a glb. Since a lattice L is an algebraic system with binary operations $\vee$ and $\wedge$, it is denoted by* $[L; \vee, \wedge]$.

In Example 13.1.2. the operation table for the *lub* operation is easy, although admittedly tedious, to do. We can observe that every pair of elements in this poset has a least upper bound. In fact, $A \vee B = A \cup B$.

The reader is encouraged to write out the operation table for the *glb* operation and to note that every pair of elements in this poset also has a *glb*, so that $\mathcal{P}(A)$ together with these two operations is a lattice. We further observe that:

(1) $[\mathcal{P}(A); \vee, \wedge]$ is a lattice (under $\subseteq$) for any set A, and

(2) the join operation is the set operation of union and the meet operation is the operation intersection; that is, $\vee = \cup$ and $\wedge = \cap$.

It can be shown (see the exercises) that the commutative laws, associative laws, idempotent laws, and absorption laws are all true for any lattice. An example of this is clearly $[\mathcal{P}(A); \cup, \cap]$, since these laws hold in the algebra of sets. This lattice is also distributive in that join is distributive over meet and meet is distributive over join. This is not always the case for lattices in general however.

Definition: Distributive Lattice. *Let* $[L; \bigvee, \bigwedge]$ *be a lattice (under* $\leq$*).* $[L; \bigvee, \bigwedge\}$ *is called a distributive lattice if and only if the distributive laws hold; that is, for all* $a, b, c \in L$, *we have:*

$$a \bigvee (b \bigwedge c) = (a \bigvee b) \bigwedge (a \vee c) \text{ and}$$

$$a \bigwedge (b \bigvee c) = (a \bigwedge b) \bigvee (a \wedge c).$$

Example 13.2.1. If A is any set, the lattice $[\mathcal{P}(A); \bigcup, \bigcap]$ is distributive.

Example 13.2.2. We now give an example of a lattice where the distributive laws do not hold. Let $L = \{1, 2, 3, 5, 30\}$. Then L is a poset under the relation divides. The operation tables for $\vee$ and $\wedge$ on L are:

$\bigvee$	1	2	3	5	30
1	1	2	3	5	30
2	2	2	30	30	30
3	3	30	3	30	30
5	5	30	30	5	30
30	30	30	30	30	30

$\bigwedge$	1	2	3	5	30
1	1	1	1	1	1
2	1	2	1	1	2
3	1	1	3	1	3
5	1	1	1	5	5
30	1	2	3	5	30

Since every pair of elements in L has both a join and a meet, $[L; \bigvee, \bigwedge]$ is a lattice (under divides). Is this lattice distributive? We note that:

$$2 \bigvee (5 \bigwedge 3) = 2 \bigvee 1 = 2 \text{ and}$$

$$(2 \bigvee 5) \bigwedge (2 \bigvee 3) = 30 \bigwedge 30 = 30,$$

so that $a \bigvee (b \bigwedge c) \neq (a \bigvee b) \bigwedge (a \vee c)$ for some values of $a, b, c \in L$. Hence L is not a distributive lattice.

It can be shown that a lattice is nondistributive if and only if it contains a sublattice isomorphic to one of the lattices in Figure 13.2.1.

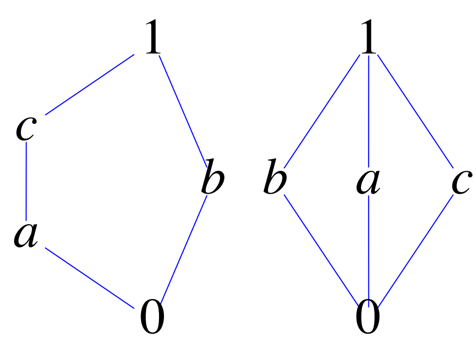


Figure 13.2.1
Nondistributive lattices

It is interesting to note that for the relation "divides" on $\mathbb{P}$, if $a, b \in \mathbb{P}$ we have:

$a \bigvee b = lcm(a, b)$, the least common multiple of a and b; that is, the smallest integer (in $\mathbb{P}$) that is divisible by both a and b;

$a \bigwedge b = gcd(a, b)$, the greatest common divisor of a and b; that is, the largest integer that divides both a and b.

EXERCISES FOR SECTION 13.2

A Exercises

1. Let L be the set of all propositions generated by p and q. What are the meet and join operations in this lattice. What are the maximum and minimum elements?

2. Which of the posets in Exercise 3 of Section 13.1 are lattices? Which of the lattices are distributive?

B Exercises

3. (a) State the commutative laws, associative laws, idempotent laws, and absorption laws for lattices.

(b) Prove these laws.

4. Let $[L; \bigvee, \bigwedge]$ be a lattice based on a partial ordering $\leq$. Prove that if $a, b, c \in L$,

(a) $a \bigvee b \geq a$.

(b) $a \bigwedge b \leq a$.

(c) $a \geq b$ and $a \geq c \Rightarrow a \geq b \bigvee c$.

13.3 Boolean Algebras

In order to define a Boolean algebra, we need the additional concept of complementation.

Definition: Complemented Lattice. *Let $[L; \bigvee, \bigwedge]$ be a lattice that contains a least element, 0, and a greatest element, 1. $[L; \bigvee, \bigwedge]$ is called a complemented lattice if and only if for every element $a \in L$, there exists an element $\bar{a}$ in L such that $a \wedge \bar{a} = 0$ and $a \vee \bar{a} = 1$. Such an element $\bar{a}$ is called a complement of the element a.*

Example 13.3.1. Let $L = \mathcal{P}(A)$, where $A = \{a, b, c\}$. Then $[L; \bigcup, \bigcap]$ is a bounded lattice with $0 = \emptyset$ and $1 = A$. Then, to find if it exists, the complement, $\overline{B}$, of, say $B = \{a, b\} \in L$, we want $\overline{B}$ such that

$$\{a, b\} \bigcap \overline{B} = \emptyset \text{ and } \{a, b\} \bigcup \overline{B} = A .$$

Here, $\overline{B} = \{c\}$, and since it can be shown that each element of L has a complement (see Exercise 1), $[L; \bigcup, \bigcap]$ is a complemented lattice. Note that if A is any set and $L = \mathcal{P}(A)$, then $[L; \bigcup, \bigcap]$ is a complemented lattice where the complement of $B \in L$ is $\overline{B} = B^c = A - B$.

In Example 13.3.1, we observe that the complement of each element of L is unique. Is this always the case? The answer is no. Consider the following.

Example 13.3.2. Let $L = \{1, 2, 3, 5, 30\}$ and consider the lattice $[L; \bigvee, \bigwedge]$ (under "divides"). The least element of L is 1 and the greatest element is 30. Let us compute the complement of the element $a = 2$. We want to determine $\bar{a}$ such that $2 \bigwedge \bar{a} = 1$ and $2 \bigvee \bar{a} = 30$. Certainly, $\bar{a} = 3$ works, but so does $\bar{a} = 5$, so the complement of $a = 2$ in this lattice is not unique. However, $[L; \bigvee, \bigwedge]$ is still a complemented lattice since each element does have at least one complement.

The following theorem gives us an insight into when uniqueness of complements occurs.

Theorem 13.3.1. *If $[L; \bigvee, \bigwedge]$ is a complemented and distributive lattice, then the complement $\bar{a}$ of any element $a \in L$ is unique.*

Proof: Let $a \in L$ and assume to the contrary that a has two complements, namely a_1 and a_2. Then by definition of complement,

$$a \wedge a_1 = 0 \text{ and } a \vee a_1 = 1,$$

Also,

$$a \wedge a_2 = 0 \text{ and } a \vee a_2 = 1.$$

So that

$$\begin{aligned} a_1 &= a_1 \bigwedge 1 = a_1 \wedge (a \vee a_2) \\ &= (a_1 \wedge a) \vee (a_1 \wedge a_2) \\ &= 0 \bigvee (a_1 \wedge a_2) \\ &= a_1 \wedge a_2. \end{aligned}$$

On the other hand,

$$\begin{aligned} a_2 &= a_2 \bigwedge 1 = a_2 \wedge (a \vee a_1) \\ &= (a_2 \wedge a) \vee (a_2 \wedge a_1) \\ &= 0 \bigvee (a_2 \wedge a_1) \\ &= a_2 \wedge a_1. \end{aligned}$$

Hence $a_1 = a_2$, which contradicts the assumption that a has two different complements, a_1 and a_2. ■

Definition: Boolean Algebra. *A Boolean algebra is a lattice that contains a least element and a greatest element and that is both complemented and distributive.*

Since the complement of each element in a Boolean algebra is unique (by Theorem 13.3.1), complementation is a valid unary operation over the set under discussion, and we will list it together with the other two operations to emphasize that we are discussing a set together with three operations. Also, to help emphasize the distinction between lattices and lattices that are Boolean algebras, we will use the letter B as the generic symbol for the set of a Boolean algebra; that is, $[B; -, \bigvee, \bigwedge]$ will stand for a general Boolean algebra.

Example 13.3.3. Let A be any set, and let $B = \mathcal{P}(A)$. Then $[B; c, \bigcup, \bigcap]$ is a Boolean algebra. Here, c stands for the complement of an element of B with respect to A, $A - B$.

This is a key example for us since all finite Boolean algebras and many infinite Boolean algebras look like this example for some A. In fact, a glance at the basic Boolean algebra laws in Table 13.3.1, in comparison with the set laws of Chapter 4 and the basic laws of logic of Chapter 3, indicates that all three systems behave the same; that is, they are isomorphic.

The "pairing" of the above laws reminds us of the principle of duality, which we state for a Boolean algebra.

Definition: Principle of Duality for Boolean Algebras. *Let* $[B; -, \vee, \wedge]$ *be a Boolean algebra (under* $\leq$*), and let S be a true statement for* $[B; -, \vee, \wedge]$. *If* S^* *is obtained from S by replacing* $\leq$ *by* $\geq$ *(this is equivalent to turning the graph upside down),* $\vee$ *by* $\wedge$, $\wedge$ *by* $\vee$, 0 *by* 1, *and* 1 *by* 0, *then* S^* *is also a true statement*.

TABLE 13.3.1

Basic Boolean Algebra Laws

Commutative Laws	
1. $a \vee b = b \vee a$	1.' $a \wedge b = b \wedge a$
Associative Laws	
2. $a \vee (b \vee c) = (a \vee b) \vee c$	2.' $a \wedge (b \wedge c) = (a \wedge b) \wedge c$
Distributive Laws	
3. $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$	3.' $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
Identity Laws	
4. $a \vee 0 = 0 \vee a = a$	4.' $a \wedge 1 = 1 \wedge a = a$
Complement Laws	
5. $a \vee \overline{a} = 1$	5.' $a \wedge \overline{a} = 0$
Idempotent Laws	
6. $a \vee a = a$	6.' $a \wedge a = a$
Null Laws	
7. $a \vee 1 = 1$	7.' $a \wedge 0 = 0$
Absorption Laws	
8. $a \vee (a \wedge b) = a$	8.' $a \wedge (a \vee b) = a$
DeMorgan's Laws	
9. $\overline{a \vee b} = \overline{a} \wedge \overline{b}$	9.' $\overline{a \wedge b} = \overline{a} \vee \overline{b}$
Involution Law	
10. $\overline{\overline{a}} = a$	

Example 13.3.4. The laws 1' through 9' are the duals of the Laws 1 through 9 respectively. Law 10 is its own dual.

We close this section with some comments on notation. The notation for operations in a Boolean algebra is derived from the algebra of logic. However, other notations are used. These are summarized in the following chart;

Notation used in this text (Mathematics notation)	Set Notation	Logic Design (CS/EE notation)	Read as
$\vee$	$\cup$	$\oplus$	join
$\wedge$	$\cap$	$\otimes$	meet
$-$	c	$-$	complement
$\leq$	$\subseteq$	$\leq$	underlying partial ordering

Mathematicians most frequently use the notation of the text, and, on occasion, use set notation for Boolean algebras. Thinking in terms of sets may be easier for some people. Computer designers traditionally use the arithmetic and notation. In this latter notation, DeMorgan's Laws become:

$$(9)\quad \overline{a \oplus b} = \overline{a} \otimes \overline{b}$$

and

$$(9')\quad \overline{a \otimes b} = \overline{a} \oplus \overline{b}.$$

EXERCISES FOR SECTION 13.3

A Exercises

1. Determine the complement of each element $B \in L$ in Example 13.3.1. Is this lattice a Boolean algebra? Why?

2. (a) Determine the complement of each element of D_6 in $[D_6; \vee, \wedge]$.

   (b) Repeat part a using the lattice in Example 13.2.2.

   (c) Repeat part a using the lattice in Exercise 1 of Section 13.1.

   (d) Are the lattices in parts a, b, and c Boolean algebras? Why?

3. Determine which of the lattices of Exercise 3 of Section 13.1 are Boolean algebras.

4. Let $A = \{a, b\}$ and $B = \mathcal{P}(A)$.

   (a) Prove that $[B; c, \cup, \cap]$ is a Boolean algebra.

   (b) Write out the operation tables for the Boolean algebra.

5. It can be shown that the following statement, S, holds for any Boolean algebra $[B; -, \vee, \wedge]$: $(a \wedge b) = a$ if $a \leq b$.

   (a) Write the dual, S^*, of the statement S.

   (b) Write the statement S and its dual, S^*, in the language of sets.

   (c) Are the statements in part b true for all sets?

   (d) Write the statement S and its dual, S^*, in the language of logic.

   (e) Are the statements in part d true for all propositions?

6. State the dual of:

   (a) $a \vee (b \wedge a) = a$.

   (b) $a \vee (\overline{(\overline{b} \vee a)} \wedge b) = 1$.

   (c) $(\overline{a \wedge \overline{b}}) \wedge b = a \vee b$.

   B Exercises

7. Formulate a definition for isomorphic Boolean algebras.

13.4 Atoms of a Boolean Algebra

In this section we will look more closely at previous claims that every finite Boolean algebra is isomorphic to an algebra of sets. We will show that every finite Boolean algebra has 2^n elements for some n with precisely n generators, called *atoms*.

Consider the Boolean algebra $[B; -, \vee, \wedge]$, whose graph is:

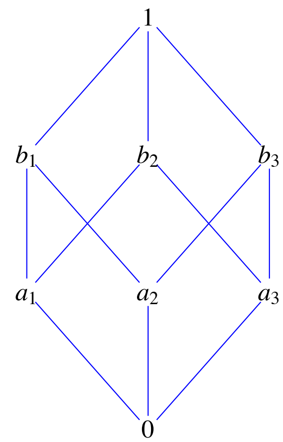


Figure 13.4.1
Illustration of the atom concept

We note that $1 = a_1 \vee a_2 \vee a_3$, $b_1 = a_1 \vee a_2$, $b_2 = a_1 \vee a_3$, and $b_3 = a_2 \vee a_3$; that is, each of the elements above level one can be described completely and uniquely in terms of the elements on level one. The a_i s have uniquely generated the nonzero elements of B much like a basis in linear algebra generates the elements in a vector space. We also note that the a_i s are the immediate successors of the minimum element, 0. In any Boolean algebra, the immediate successors of the minimum element are called *atoms*. Let A be any nonempty set. In the Boolean algebra $[\mathcal{P}(A);\ c,\ \bigcup,\ \bigcap]$ (over $\subseteq$), the singleton sets are the generators, or atoms, of the algebraic structure since each element $\mathcal{P}(A)$ can be described completely and uniquely as the join or union of singleton sets.

Definition: Atom. *A nonzero element a in a Boolean algebra $[B;\ -,\ \vee,\ \wedge]$ is called an atom if for every $x \in B$, $x \wedge a = a$ or $x \wedge a = 0$.*

The condition that $x \wedge a = a$ tells us that x is a successor of a; that is, $a \leq x$, as depicted in Figure 13.4.2a.

The condition $x \wedge a = 0$ is true only when x and a are "not connected." This occurs when x is another atom or if x is a successor of atoms different from a, as depicted in Figure 13.4.2b.

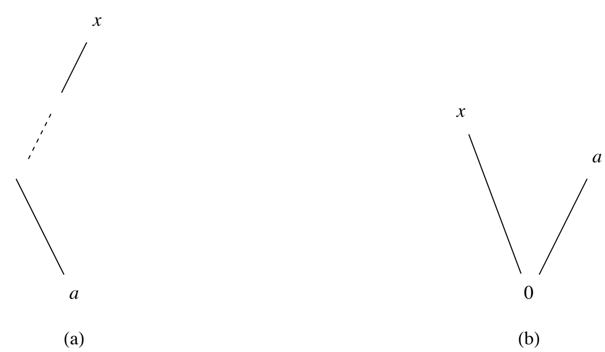


Figure 13.4.2

Example 13.4.1. The set of atoms of the Boolean algebra $[D_{30};\ -,\ \bigvee,\ \bigwedge]$ is $M = \{2,\ 3,\ 5\}$. To see that $a = 2$ is an atom, let x be any nonzero element of D_{30} and note that one of the two conditions $x \bigwedge 2 = 2$ or $x \bigwedge 2 = 1$ holds. Of course, to apply the definition to this Boolean algebra, we must remind ourselves that in this case the 0-element is 1, the operation $\wedge$ is *gcd*, and the poset relation $\leq$ is "divides." So if $x = 10$, we have $10 \bigwedge 2 = 2$ (or $2 \mid 10$), so Condition 1 holds. If $x = 15$, the first condition is not true. (Why?) However, Condition 2, $15 \bigwedge 2 = 1$, is true. The reader is encouraged to show that each of the elements 2, 3, and 5 satisfy the definition (see Exercise 13.4.1). Next, if we compute the join (*lcm* in this case) of all possible combinations of the atoms 2, 3, and 5, we will generate all nonzero elements of D_{30}. For example, $2 \bigvee 3 \bigvee 5 = 30$ and $2 \bigvee 5 = 10$. We state this concept formally in the following theorem, which we give without proof.

Theorem 13.4.1. *Let $[B;\ -,\ \vee,\ \wedge]$ be any finite Boolean algebra. Let $A = \{a_1,\ a_2, \ldots, a_n\}$ be the set of all n atoms of $[B;\ -,\ \vee,\ \wedge]$. Then every nonzero element in B can be expressed uniquely as the join of a subset of A.*

We now ask ourselves if we can be more definitive about the structure of different Boolean algebras of a given order. Certainly, the Boolean algebras $[D_{30};\ -,\ \bigvee,\ \bigwedge]$ and $[\mathcal{P}(A);\ c,\ \bigcup,\ \bigcap]$ have the same graph (that of Figure 13.4.1), the same number of atoms, and, in all respects, look the same except for the names of the elements and the operations. In fact, when we apply corresponding operations to corresponding elements, we obtain corresponding results. We know from Chapter 11 that this means that the two structures are isomorphic as Boolean algebras. Furthermore, the graphs of these examples are exactly the same as that of Figure 13.4.1, which is an arbitrary Boolean algebra of order $8 = 2^3$.

In these examples of a Boolean algebra of order 8, we note that each had 3 atoms and $2^3 = 8$ number of elements, and all were isomorphic to $[\mathcal{P}(A); c, \bigcup, \bigcap]$, where $A = \{a, b, c\}$. This leads us to the following questions:

(1) Are there any other different (nonisomorphic) Boolean algebras of order 8?

(2) What is the relationship, if any, between finite Boolean algebras and their atoms?

(3) How many different (nonisomorphic) Boolean algebras are there of order 2? Order 3? Order 4? And so on.

The answers to these questions are given in the following theorem and corollaries. We include the proofs of the corollaries since they are instructive.

Theorem 13.4.2. *Let* $[B; -, \lor, \land]$ *be any finite Boolean algebra, and let A be the set of all atoms in this Boolean algebra. Then* $[B; -, \lor, \land]$ *is isomorphic to* $[\mathcal{P}(A); c, \bigcup, \bigcap]$.

Corollary 13.4.1. *Every finite Boolean algebra* $[B; -, \lor, \land]$ *has* 2^n *elements for some positive integer* n.

Proof: Let A be the set of all atoms of B and let $|A| = n$. Then there are exactly 2^n elements (subsets) in $\mathcal{P}(A)$, and by Theorem 13.4.2, $[B; -, \bigvee, \bigwedge]$ is isomorphic to $[\mathcal{P}(A); c, \bigcup, \bigcap]$. ■

Corollary 13.4.2. All Boolean algebras of order 2^n are isomorphic to each other. (The graph of the Boolean algebra of order 2^n is the n-cube).

Proof: By Theorem 13.4.2, every Boolean algebra of order 2^n is isomorphic to $[\mathcal{P}(A); c, \bigcup, \bigcap]$ when $|A| = n$. Hence, they are all isomorphic to one another. ■

The above theorem and corollaries tell us that we can only have finite Boolean algebras of orders $2^1, 2^2, 2^3, \ldots, 2^n$, and that all finite Boolean algebras of any given order are isomorphic. These are powerful tools in determining the structure of finite Boolean algebras. In the next section, we will try to find the easiest way of describing a Boolean algebra of any given order.

EXERCISES FOR SECTION 13.4

A Exercises

1. (a) Show that $a = 2$ is an atom of the Boolean algebra $[D_{30}; -, \bigvee, \bigwedge]$.

   (b) Repeat part a for the elements 3 and 5 of D_{30}.

   (c) Verify Theorem 13.4.1 for the Boolean algebra $[D_{30}; -, \bigvee, \bigwedge]$.

2. Let $A = \{a, b, c\}$.

   (a) Rewrite the definition of atom for $[\mathcal{P}(A); c, \bigcup, \bigcap]$. What does $a \leq x$ mean in this example?

   (b) Find all atoms of $[\mathcal{P}(A); c, \bigcup, \bigcap]$.

   (c) Verify Theorem 13.4.1 for $[\mathcal{P}(A); c, \bigcup, \bigcap]$.

3. Verify Theorem 13.4.2 and its corollaries for the Boolean algebras in Exercises 1 and 2 of this section.

4. Give a description of all Boolean algebras of order 16. (*Hint*: Use Theorem 13.4.2.) Note that the graph of this Boolean algebra is given in Figure 9.4.5.

5. Corollary 13.4.1 states that there do not exist Boolean algebras of orders 3, 5, 6, 7, 9, etc. (orders different from 2^n). Prove that we cannot have a Boolean algebra of order 3. (*Hint*: Assume that $[B; -, \bigvee, \bigwedge]$ is a Boolean algebra of order 3 where $B = \{0, x, 1\}$ and show that this cannot happen by investigating the possibilities for its operation tables.)

6. (a) There are many different, yet isomorphic, Boolean algebras with two elements. Describe one such Boolean algebra that is derived from a power set, $\mathcal{P}(A)$, under $\subseteq$. Describe a second that is described from D_n, for some $n \in P$, under "divides."

   (b) Since the elements of a two-element Boolean algebra must be the greatest and least elements, 1 and 0, the tables for the operations on $\{0, 1\}$ are determined by the Boolean algebra laws. Write out the operation tables for $[\{0, 1\}; -, \bigvee, \bigwedge]$.

B Exercises

7. Find a Boolean algebra with a countably infinite number of elements.

8. Prove that the direct product of two Boolean algebras is a Boolean algebra. (*Hint*: "Copy" the corresponding proof for groups in Section 11.6.)

13.5 Finite Boolean Algebras as n-tuples of 0's and 1's

From the previous section we know that all finite Boolean algebras are of order 2^n, where n is the number of atoms in the algebra. We can therefore completely describe every finite Boolean algebra by the algebra of power sets. Is there a more convenient, or at least an alternate way, of defining finite Boolean algebras? In Chapter 11 we found that we could produce new groups by taking Cartesian products of previously known groups. We imitate this process for Boolean algebras.

The simplest nontrivial Boolean algebra is the Boolean algebra on the set $B_2 = \{0, 1\}$. The ordering on B_2 is the natural one, $0 \leqslant 0$, $0 \leqslant 1$, $1 \leqslant 1$. If we treat 0 and 1 as the truth values "false" and "true," respectively, we see that the Boolean operations $\vee$ (join) and $\wedge$ (meet) are nothing more than the logical connectives $\vee$ (or) and $\wedge$ (and). The Boolean operation, $-$, (complementation) is the logical $\neg$ (negation). In fact, this is why the symbols $-$, $\vee$, and $\wedge$ were chosen as the names of the Boolean operations. The operation tables for $[B_2; -, \vee, \wedge]$ are simply those of "or," "and," and "not," which we repeat here:

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

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

u	$\bar{u}$
0	1
1	0

By Theorem 13.4.2 and its corollaries, all Boolean algebras of order 2 are isomorphic to this one.

We know that if we form $B_2 \times B_2 = B_2^2$ we obtain the set $\{(0, 0), (0, 1), (1, 0), (1, 1)\}$, a set of order 4. We define operations on B_2^2 the natural way, namely, componentwise, so that $(0, 1) \vee (1, 1) = (0 \vee 1, 1 \vee 1) = (1, 1)$, $(0, 1) \wedge (1, 1) = (0 \wedge 1, 1 \wedge 1) = (0, 1)$ and $\overline{(0, 1)} = (\overline{0}, \overline{1}) = (1, 0)$. We claim that B_2^2 is a Boolean algebra under the componentwise operations. Hence, $[B_2^2; -, \vee, \wedge]$ is a Boolean algebra of order 4. Since all Boolean algebras of order 4 are isomorphic to each other, we have found a simple way of describing all Boolean algebras of order 4.

It is quite clear that we can describe any Boolean algebra of order 8 by considering $B_2 \times B_2 \times B_2 = B_2^3$ and, in general, any Boolean algebra of order 2^n— that is, all finite Boolean algebras—by $B_2^n = B_2 \times B_2 \times \cdots \mathrm{B}_2$ (n factors).

EXERCISES FOR SECTION 13.5

A Exercises

1. (a) Write out the operation tables for $[B_2^2; -, \vee, \wedge]$.

(b) Draw the Hasse diagram for $[B_2^2; -, \vee, \wedge]$ and compare your results with Figure 9.4.6.

(c) Find the atoms of this Boolean algebra.

2. (a) Write out the operation table for $[B_2^3; -, \vee, \wedge]$.

(b) Draw the Hasse diagram for $[B_2^3; -, \vee, \wedge]$ and compare the results with Figure 9.4.6.

3. (a) List all atoms of B_2^4.

(b) Describe the atoms of B_2^n $n \geqslant 1$.

B Exercise

4. Theorem 13.4.2 tells us we can think of any finite Boolean algebra in terms of sets. In Chapter 4, Section 3, we defined the terms *minset* and *minset normal form*. Rephrase these definitions in the language of Boolean algebra. The generalization of minsets are called *minterms*.

13.6 Boolean Expressions

In this section, we will use our background from the previous sections and set theory to develop a procedure for simplifying Boolean expressions. This procedure has considerable application to the simplification of circuits in switching theory or logical design.

Definition: *Boolean Expression. Let $[B; -, \vee, \wedge]$ be any Boolean algebra. Let $x_1, x_2, \ldots, x_k$ be variables in B; that is, variables that can assume values from B. A Boolean expression generated by $x_1, x_2, \ldots, x_k$ is any valid combination of the x_i and the elements of B with the operations of meet, join, and complementation.*

This definition, as expected, is the analog of the definition of a proposition generated by a set of propositions, presented in Section 3.2.

Each Boolean expression generated by k variables, $e(x_1, \ldots, x_k)$, defines a function $f : B^k \to B$ where $f(a_1, \ldots, a_k) = e(a_1, \ldots, a_k)$. If B is a finite Boolean algebra, then there are a finite number of functions from B^k into B. Those functions that are defined in terms of Boolean expressions are called *Boolean functions*. As we will see, there is an infinite number of Boolean expressions that define each Boolean function. Naturally, the "shortest" of these expressions will be preferred. Since electronic circuits can be described as Boolean functions with $B = B_2$, this economization is quite useful.

Example 13.6.1. Consider any Boolean algebra $[B;\ -,\ \lor,\ \land]$ of order 2. How many functions $f : B^2 \to B$ are there? First, all Boolean algebras of order 2 are isomorphic to $[B_2;\ -,\ \lor,\ \land]$ so we want to determine the number of functions $f : B_2^2 \to B_2$. If we consider a Boolean function of two variables, x_1 and x_2, we note that each variable has two possible values 0 and 1, so there are 2^2 ways of assigning these two values to the $k = 2$ variables. Hence, the table below has $2^2 = 4$ rows. So far we have a table such as that labeled 13.6.1.

x_1	x_2	$f(x_1, x_2)$
0	0	?
0	1	?
1	0	?
1	1	?

Table 13.6.1
General Form Of Boolean Function $f(x_1,\ x_2)$ of Example 13.6.1

How many possible different function values $f(x_1,\ x_2)$ can there be? To list a few: $f_1(x_1,\ x_2) = x_1$, $f_2(x_1,\ x_2) = x_2$, $f_3(x_1,\ x_2) = x_1 \lor x_2$, $f_4(x_1,\ x_2) = (x_1 \land \overline{x_2}) \lor x_2$, $f_5(x_1,\ x_2) = x_1 \land x_2 \lor \overline{x_2}$, etc. Each of these will give a table like that of Table 13.6.1. The tables for f_1, and f_3 appear in Table 13.6.2.

x_1	x_2	$f_1(x_1, x_2)$
0	0	0
0	1	0
1	0	1
1	1	1

x_1	x_2	$f_3(x_1, x_2)$
0	0	0
0	1	1
1	0	1
1	1	1

Table 13.6.2
Boolean Functions f_1 and f_3 of Example 13.6.1

Two functions are different if and only if their tables (values) are different for at least one row. Of course by using the basic laws of Boolean algebra we can see that $f_3 = f_4$. Why? So if we simply list by brute force all "combinations" of x_1 and x_2 we will obtain unnecessary duplication. However, we note that for any combination of the variables x_1, and x_2 there are only two possible values for $f(x_1,\ x_2)$, namely 0 or 1. Thus, we could write $2^4 = 16$ different functions on 2 variables.

Now let's count the number of different Boolean functions in a more general setting. We will consider two cases: first, when $B = B_2$, and second, when B is any finite Boolean algebra with 2^n elements.

Let $B = B_2$. Each function $f : B^k \to B$ is defined in terms of a table having 2^k rows. Therefore, since there are two possible images for each element of B^k, there are 2 raised to the 2^k, or 2^{2^k} different functions. *We claim that every one of these functions is a Boolean function.*

Now suppose that $|B| = 2^n > 2$. A function from B^k into B can still be defined in terms of a table. There are $|B|^k$ rows to each table and $|B|$ possible images for each row. Therefore, there are 2^n raised to the power 2^{nk} different functions. If $n > 1$, then not every one of these functions is a Boolean function. Notice that in counting the numbers of functions we are applying the result of Exercise 5 of Section 7.1.

Since all Boolean algebras are isomorphic to a Boolean algebra of sets, the analogues of statements in sets are useful in Boolean algebras.

Definition: *Minterm. A Boolean expression generated by $x_1,\ x_2,\ \ldots,\ x_k$ that has the form*

$$\bigwedge_{i=1}^{k} y_i,$$

where each y_i may be either x_i or $\overline{x_i}$ is called a minterm generated by $x_1,\ x_2,\ \ldots,\ x_k$.

By a direct application of the Product Rule we see that there are 2^k different minterms generated by $x_1,\ \ldots,\ x_k$.

Definition: *Minterm Normal Form. A Boolean expression generated by $x_1,\ \ldots,\ x_k$ is in minterm normal form if it is the join of expressions of the form $a \land m$, where $a \in B$ and m is a minterm generated by $x_1,\ \ldots,\ x_k$. That is, it is of the form*

$$\bigvee_{j=1}^{p} \left(a_j \land m_j\right),$$

where $p = 2^k$ and $m_1, m_2,\ \ldots,\ m_p$ are the minterms generated by $x_1,\ \ldots,\ x_k$

If $B = B_2$, then each a_j in a minterm normal form is either 0 or 1. Therefore, $a_j \land m_j$ is either 0 or m_j.

Theorem 13.6.1. *Let $e(x_1, \ldots, x_k)$ be a Boolean expression over B. There exists a unique minterm normal form $M(x_1,\ \ldots,\ x_k)$ that is equivalent to $e(x_1,\ \ldots,\ x_k)$ in the sense that e and M define the same function from B^k into B.*

The uniqueness in this theorem does not include the possible ordering of the minterms in M (commonly referred to as "uniqueness up to the order of minterms"). The proof of this theorem would be quite lengthy, and not very instructive, so we will leave it to the interested reader to attempt. The implications of the theorem are very interesting, however.

If $|B| = 2^n$, then there are 2^n raised to the 2^k different minterm normal forms. Since each different minterm normal form defines a different function, there are a like number of Boolean functions from B^k into B. If $B = B_2$, there are as many Boolean functions (2 raised to the 2^k) as there are functions from B^k into B, since there are 2 raised to the 2^n functions from B^k into B. The significance of this result is that any desired function can be obtained using electronic circuits having 0 or 1 (off or on, positive or negative) values, but more complex, multivalued circuits would not have this flexibility.

We will close this section by examining minterm normal forms for expressions over B_2 , since they are a starting point for circuit economization.

Example 13.6.2. Consider the Boolean expression $f(x_1, x_2) = x_1 \lor \overline{x_2}$. One method of determining the minterm normal form of f is to think in terms of sets. Consider the diagram with the usual translation of notation in Figure 13.6.1. Then $f(x_1,\ x_2) = (\overline{x_1} \land \overline{x_2}) \lor (x_1 \land \overline{x_2}) \lor (x_1 \land x_2)$.

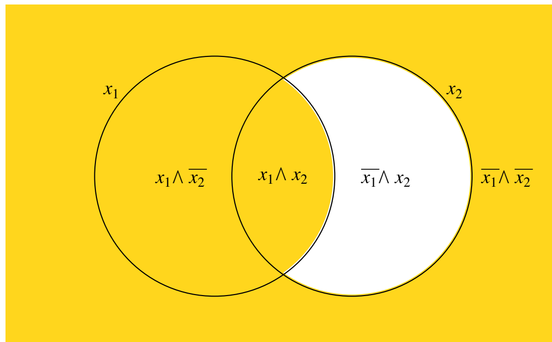


Figure 13.6.1

Example 13.6.3. Consider the function $f : B_2^3 \to B_2$ defined by Table 13.6.3. The minterm normal form for f can be obtained by taking the join of minterms that correspond to rows that have an image value of 1. If $f(a_1,\ a_2,\ a_3) = 1$, then include the minterm $y_1 \land y_2 \land y_3$ where

$$y_j = \begin{cases} x_j & \text{if } a_j = 1 \\ \overline{x_j} & \text{if } a_j = 0 \end{cases}$$

TABLE 13.6.3

Boolean Function of $f(a_1,\ a_2,\ a_3)$ Of Example 13.6.3

a_1	a_2	a_3	$f(a_1, a_2, a_3)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	0	1	0

Therefore,

$$f(x_1,\ x_2,\ x_3) = (\overline{x_1} \land \overline{x_2} \land \overline{x_3}) \lor (\overline{x_1} \land x_2 \land x_3) \lor (x_1 \land x_2 \land \overline{x_3}).$$

The minterm normal form is a first step in obtaining an economical way of expressing a given Boolean function. For functions of more than three variables, the above set theory approach tends to be awkward. Other procedures are used to write the normal form. The most convenient is the Karnaugh map, a discussion of which can be found in any logical design/switching theory text (see, for example, Hill and Peterson).

EXERCISES FOR SECTION 13.6

A Exercises

1. (a) Write the 16 possible functions of Example 13.6.1. (*Hint:* Find all possible joins of minterms generated by x_1 and x_2 .)

(b) Write out the tables of several of the above Boolean functions to show that they are indeed different.

(c) Determine the minterm normal form of

$$f_1(x_1,\ x_2) = x_1 \lor x_2,$$

$$f_2(x_1,\ x_2) = \overline{x_1} \lor \overline{x_2}$$

$$f_3(x_1,\ x_2) = 0,\ f_4(x_1,\ x_2) = 1.$$

2. Consider the Boolean expression $f(x_1,\ x_2,\ x_3) = (\overline{x_3} \land x_2) \lor (\overline{x_1} \land x_3) \lor (x_2 \land x_3)$ on $[B_2;\ -,\ \lor,\ \land]$.

(a) Simplify this expression using basic Boolean algebra laws.

(b) Write this expression in minterm normal form.

(c) Write out the table for the given function defined by f and compare it to the tables of the functions in parts a and b.

(d) How many possible different functions in three variables on $[B_2; -, \lor, \land]$ are there?

B Exercise

3. Let $[B; -, \lor, \land]$ be a Boolean algebra of order 4, and let f be a Boolean function of two variables on B.

(a) How many elements are there in the domain of f?

(b) How many different Boolean functions are there of two, variables? Three variables?

(c) Determine the minterm normal form of $f(x_1, x_2) = x_1 \lor x_2$.

(d) If $B = \{0, a, b, 1\}$, define a function from B^2 into B that is not a Boolean function.

13.7 A Brief Introduction to the Application of Boolean Algebra to Switching Theory

The algebra of switching theory is Boolean algebra. The standard notation used for Boolean algebra operations in most logic design/switching theory texts is + for $\lor$ and • for $\land$. Complementation is as in this text. Therefore, $(x_1 \land \overline{x_2}) \lor (x_1 \land x_2) \lor (\overline{x_1} \land x_2)$ becomes $x_1 \bullet \overline{x_2} + x_1 \bullet x_2 + \overline{x_1} \bullet x_2$, or simply $x_1 \overline{x_2} + x_1 x_2 + \overline{x_1} x_2$. All concepts developed previously for Boolean algebras hold. The only change is purely notational. We make the change in this section solely to introduce the reader to another frequently used notation. Obviously, we could have continued the discussion with our previous notation.

The simplest switching device is the on-off switch. If the switch is closed, on, current will pass through it; if it is open, off, current will not pass through it. If we designate on by true or the logical, or Boolean, 1, and off by false, the logical, or Boolean, 0, we can describe electrical circuits containing switches by logical, or Boolean, expressions. The expression $x_1 \bullet x_2$ represents the situation in which a series of two switches appears in a circuit (see Figure 13.7. 1a). In order for current to flow through the circuit, both switches must be on, that is, have the value 1.

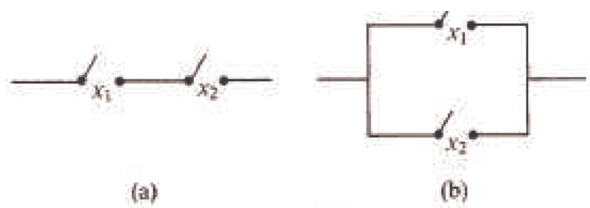


FIGURE 13.7.1

Similarly, a pair of parallel switches, as in Figure 13.7.1b, is described algebraically by $x_1 + x_2$. Many of the concepts in Boolean algebra can be thought of in terms of switching theory. For example, the distributive law in Boolean algebra (in +, • notation) is:$x_1 \bullet (x_2 + x_3) = x_1 \bullet x_2 + x_1 \bullet x_3$. Of course, this says the expression on the left is always equivalent to that on the right. The switching circuit analogue of the above statement is that Figure 13.7.2a is equivalent (as an electrical circuit) to Figure 13.7.2b.

The circuits in a digital computer are composed of large quantities of switches that can be represented as in Figure 13.7.2 or can be thought of as boxes or gates with two or more inputs (except for the NOT gate) and one output. These are often drawn as in Figure 13.7.3. For example, the OR gate, as the name implies, is the logical/Boolean OR function. The on-off switch function in Figure 13.7.3a in gate notation is Figure 13.7.3b.

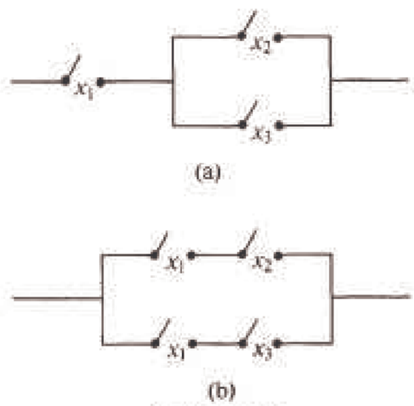


FIGURE 13.7.2

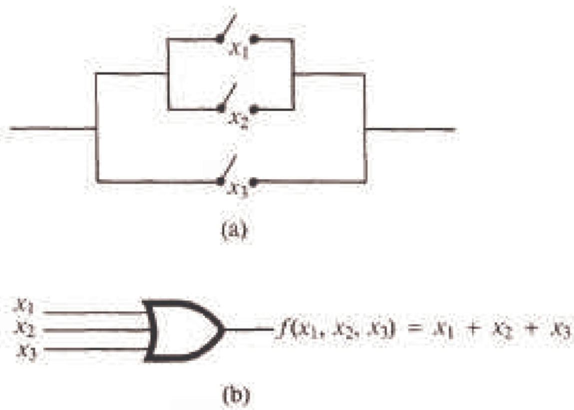
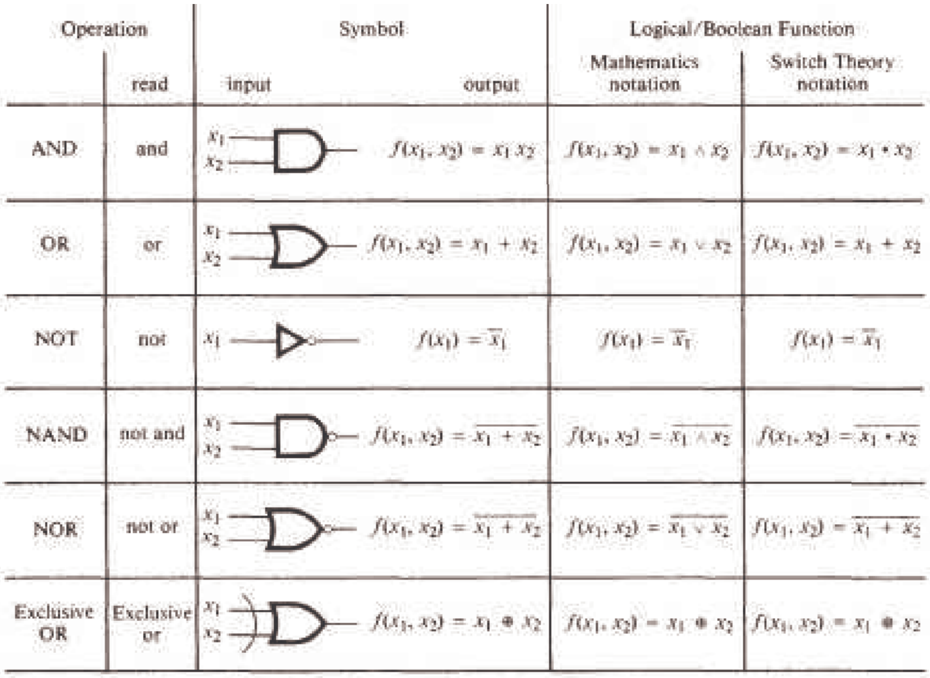


FIGURE 13.7.3

Either diagram indicates that the circuit will conduct current if and only if $f(x_1, x_2, x_3)$ is true, or 1. We list the gate symbols that are widely used in switching theory in Figure 13.7.4 with their names. The names mean, and are read, exactly as they appear. For example, NAND means "not x_1 and x_2" or algebraically, $\overline{x_1 \wedge x_2}$, or $\overline{x_1 \bullet x_2}$.

The circuit in Figure 13.7.5a can be described by gates. To do so, simply keep in mind that the Boolean function $f(x_1, x_2) = x_1 \bullet \overline{x_2}$ of this circuit contains two operations. The operation of complementation takes precedence over that of "and," so we have Figure 13.7.5b.

Example 13.7.1. The switching circuit in Figure 13.7.6a can be expressed through the logic, or gate, circuit in Figure 13.7.6b.



Operation	read	Symbol: input → output	Logical/Boolean Function: Mathematics notation	Logical/Boolean Function: Switch Theory notation
AND	and	$f(x_1, x_2) = x_1\, x_2$	$f(x_1, x_2) = x_1 \wedge x_2$	$f(x_1, x_2) = x_1 \bullet x_2$
OR	or	$f(x_1, x_2) = x_1 + x_2$	$f(x_1, x_2) = x_1 \vee x_2$	$f(x_1, x_2) = x_1 + x_2$
NOT	not	$f(x_1) = \overline{x_1}$	$f(x_1) = \overline{x_1}$	$f(x_1) = \overline{x_1}$
NAND	not and	$f(x_1, x_2) = \overline{x_1 + x_2}$	$f(x_1, x_2) = \overline{x_1 \wedge x_2}$	$f(x_1, x_2) = \overline{x_1 \bullet x_2}$
NOR	not or	$f(x_1, x_2) = \overline{x_1 + x_2}$	$f(x_1, x_2) = \overline{x_1 \vee x_2}$	$f(x_1, x_2) = \overline{x_1 + x_2}$
Exclusive OR	Exclusive or	$f(x_1, x_2) = x_1 \oplus x_2$	$f(x_1, x_2) = x_1 \oplus x_2$	$f(x_1, x_2) = x_1 \oplus x_2$

FIGURE 13.7.4

We leave it to the reader to analyze both figures and to convince him- or herself that they do describe the same circuit. The circuit can be described algebraically as

$$f(x_1,\ x_2,\ x_3) = ((x_1 + x_2) + (x_1 + x_3)) \bullet x_1 \bullet \overline{x_2}.$$

We can use basic Boolean algebra laws to simplify or minimize this Boolean function (circuit):

$$\begin{aligned}
f(x_1,\ x_2,\ x_3) &= ((x_1 + x_2) + (x_1 + x_3)) \bullet x_1 \bullet \overline{x_2} \\
&= (x_1 + x_2 + x_3) \bullet x_1 \bullet x_2 \\
&= (x_1 \bullet x_1 \bullet \overline{x_2} + x_2 \bullet x_1 \bullet \overline{x_2} + x_3 \bullet x_1 \bullet \overline{x_2} \\
&= x_1 \bullet \overline{x_2} + 0 \bullet x_1 + x_3 \bullet x_1 \bullet \overline{x_2} \\
&= x_1 \bullet \overline{x_2} + x_3 \bullet x_1 \bullet \overline{x_2} \\
&= x_1 \bullet (\overline{x_2} + \overline{x_2} \bullet x_3) \\
&= x_1 \bullet \overline{x_2} \bullet (1 + x_3) \\
&= x_1 \bullet \overline{x_2}\,.
\end{aligned}$$

The circuit for *f* may be described as in Figure 13.7.5. This is a less expensive circuit since it involves considerably less hardware.

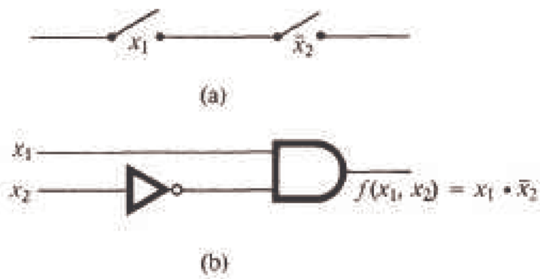


FIGURE 13.7.5

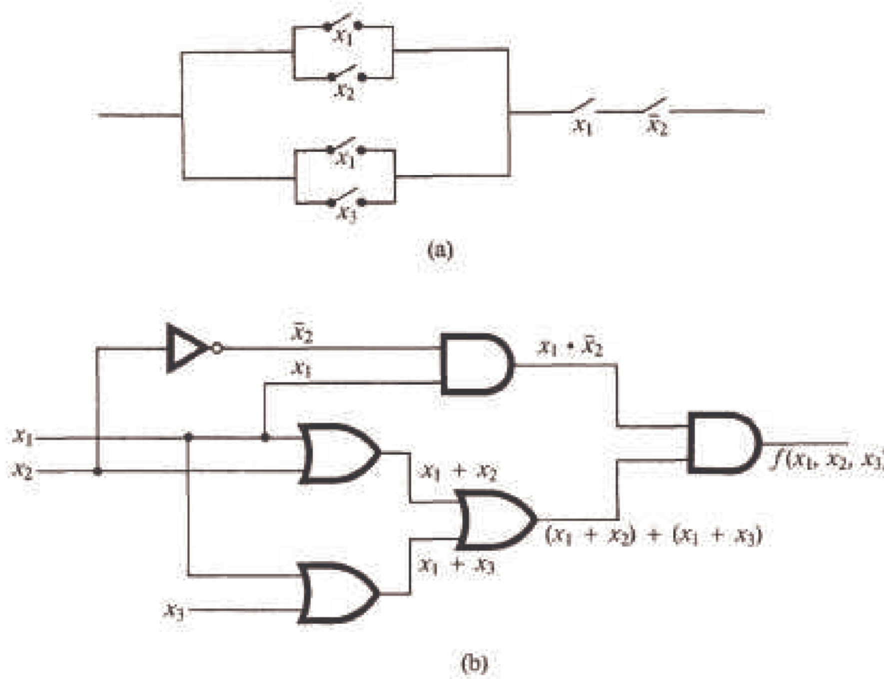


FIGURE 13.7.6

The table for *f* is:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

The Venn diagram that represents *f* is the shaded portion in Figure 13.7.7. From this diagram, we can read off the minterm normal form of *f*:

$$f(x_1,\ x_2,\ x_3) = x_1 \bullet \overline{x_2} \bullet \overline{x_3}\ +\ x_1 \bullet \overline{x_2} \bullet x_3.$$

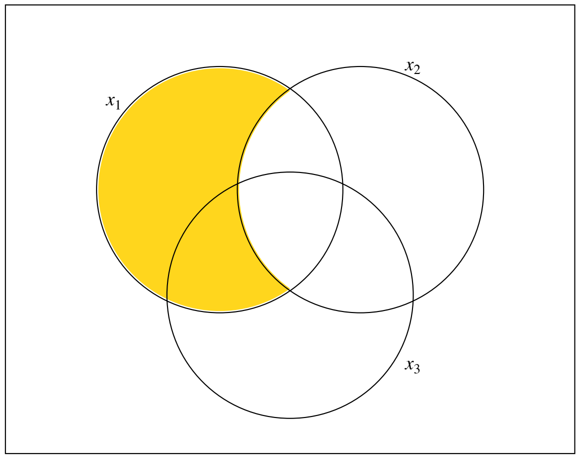


Figure 13.7.7

The circuit (gate) diagram appears in Figure 13.7.8.

How do we interpret this? We see that $f(x_1, x_2, x_3) = 1$ when $x_1 = 1$, $x_2 = 0$, and $x_3 = 0$ or $x_3 = 1$. Current will be conducted through the circuit when switch x_1 is on, switch x_2 is off, and when switch x_3 is either off or on.

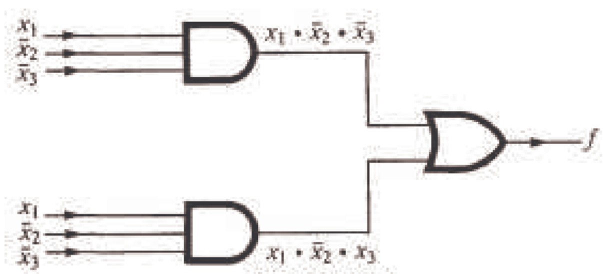
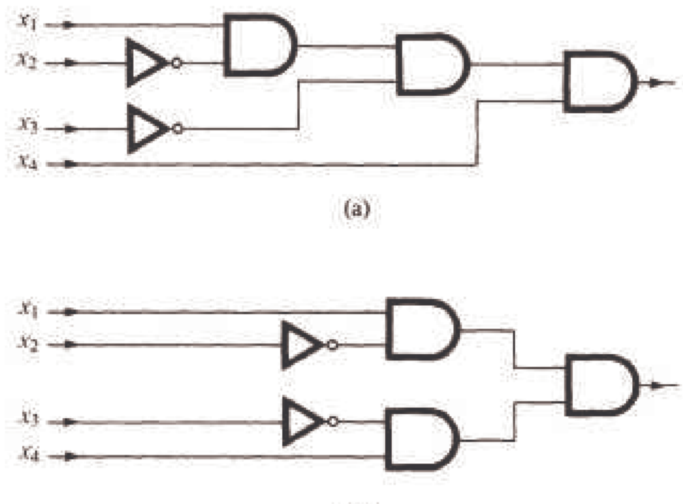


FIGURE 13.7.8

We close this section with a brief discussion of minimization, or reduction, techniques. We have discussed two in this text: algebraic (using basic Boolean rules) reduction and the minterm normal form technique. Other techniques are discussed in switching theory texts. When one reduces a given Boolean function, or circuit, it is possible to obtain a circuit that does not look simpler, but may be more cost effective, and is, therefore, simpler with respect to time. We illustrate with an example.

Example 13.7.2. Consider the Boolean function of Figure 13.7.9a is $f(x_1, x_2, x_3, x_4) = \left(\left(x_1 \bullet \overline{x_2}\right) \bullet \overline{x_3}\right) \bullet x_4$, which can also be diagrammed as in Figure 13.7.9b.



(a)
(b)
FIGURE 13.7.9

Is Circuit b simpler than Circuit a? Both circuits contain the same number of gates, so the hardware costs (costs per gate) would be the same. Hence, intuitively, we would guess that they are equivalent with respect to simplicity. However, the signals x_3 and x_4 in Circuit a pass through three levels of gating before reaching the output. All signals in Circuit b go through only two levels of gating (disregard the NOT gate when counting levels). Each level of logic (gates) adds to the time delay of the development of a signal at the output. In computers, we want the time delay to be as small as possible. Frequently, speed can be increased by decreasing the number of levels in a circuit. However, this frequently forces a larger number of gates to be used, thus increasing costs. One of the more difficult jobs of a design engineer is to balance off speed with hardware costs (number of gates).

One final remark on notation: The circuit in Figure 13.7.10a can be written as in Figure 13.7.10b, or simply as in Figure 13.7.10c.

EXERCISES FOR SECTION 13.7

A Exercises

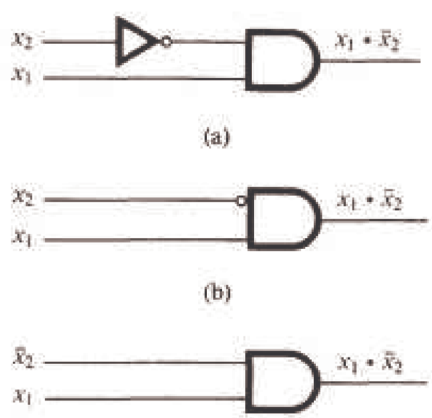
1. (a) Write all inputs and outputs from Figure 13.7.11 and show that its Boolean function is $f(x_1,\ x_2,\ x_3) = \overline{((x_1 + x_2) \bullet x_3)} \bullet (x_1 + x_2)$.

(b) Simplify f algebraically.

(c) Find the minterm normal form of f.

(d) Draw and compare the circuit (gate) diagram of parts b and c above.

(e) Draw the on-off switching diagram of f in part a.



(a)
(b)
(c)
FIGURE 13.7.10

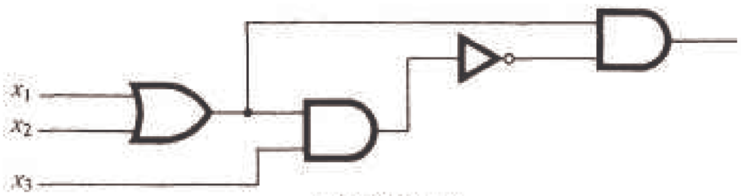


FIGURE 13.7.11

(f) Write the table of the Boolean function f in part a and interpret the results.

2. Given Figure 13.7.12:

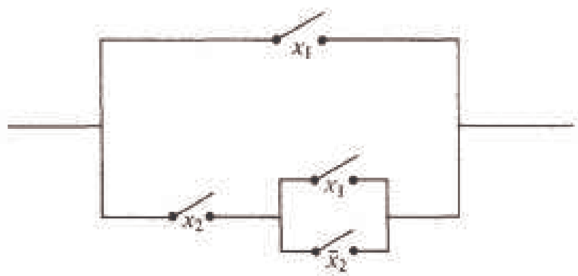


FIGURE 13.7.12

(a) Write the Boolean function that represents the given on-off circuit.

(b) Show that the Boolean function obtained in answer to part a can be reduced to $f(x_1, x_2) = x_1$. Draw the on-off circuit diagram of this simplified representation.

(c) Draw the circuit (gate) diagram of the given on-off circuit diagram.

(d) Determine the minterm normal of the Boolean function found in the answer to part a or given in part b; they are equivalent.

(e) Discuss the relative simplicity and advantages of the circuit gate diagrams found in answer to parts c and d.

3. (a) Write the circuit (gate) diagram of

$f(x_1, x_2, x_3) = (x_1 \bullet x_2 + x_3) \bullet (x_2 + x_3) + x_3$.

(b) Simplify the function in part a by using basic Boolean algebra laws.

(c) Write the circuit (gate) diagram of the result obtained in part b.

(d) Draw the on-off switch diagrams of parts a and b.

4. Consider the Boolean function

$$f(x_1, x_2, x_3, x_4) = x_1 + (x_2 \bullet (\overline{x_1} + x_4) + x_3 \bullet (\overline{x_2} + \overline{x_4})).$$

(a) Simplify f algebraically.

(b) Draw the switching (on-off) circuit of f and the reduction of f obtained in part a.

(c) Draw the circuit (gate) diagram of f and the reduction of f obtained in answer to part a.

SUPPLEMENTARY EXERCISES FOR CHAPTER 13

Section 13.1

1. (a) Draw the Hasse diagram of the relation divides on the set $A = \{1, 2, 3, \ldots, 12\}$.

(b) For the same set A draw the Hasse diagram for the relation $\leq$ on A.

2. (a) For the poset $A = \{1, 2, 3, \ldots, 12\}$ under the relation divides find the *lub* and *glb* of the following pairs of numbers if possible: 4 and 6, 2 and 3, 10 and 4, 6 and 9.
(b) Repeat part a for the set A, but use the relation $\leq$.

Section 13.2

3. Consider the poset $\mathbb{P}$ under the relation "divides."

(a) Compute: $4 \bigvee 8, 3 \bigvee 15, 3 \bigvee 5, 4 \bigwedge 8, 3 \bigwedge 15, 3 \bigwedge 5$ for $[\mathbb{P}, \bigvee, \bigwedge]$.

(b) Is $[\mathbb{P}, \bigvee, \bigwedge]$ a distributive lattice? Explain.

(c) Does $[\mathbb{P}, \bigvee, \bigwedge]$ have a least element? Does it have a greatest element? If so, what are they?

4. Let $[L, \bigvee . \bigwedge]$ be a lattice and $a, b \in L$. Prove:

(a) $a \bigvee b = b$ if and only if $a \leq b$.

(b) $a \bigwedge b = a$ if and only if $a \leq b$.

5. Let $L = \{0, 1\}$ and define $\leq$ on L by $0 \leq 0 \leq 1 \leq 1$.

(a) Draw the Hasse diagram of this poset.

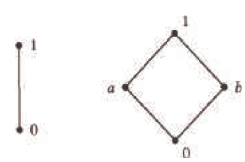
(b) Write out the operation table for $\vee$ and $\wedge$ on L observing that they are essentially the standard logical connectives.

(c) Define the operations on L^2 componentwise and draw the Hasse diagram for L^2 .

(d) Repeat part (c) for L^3.

6. (a) Let $[L_1, \bigvee, \bigwedge]$ and $[L_2, \bigvee, \bigwedge]$ be lattices. Prove that $[L_1 \times L_2, \bigvee, \bigwedge]$ is a lattice when the operations are defined componentwise as we did for algebraic systems in Section 11.6.

(b) Let L_1 and L_2 be lattices whose posets have the following Hasse diagrams respectively. List the elements in the lattice $L_1 \times L_2$.



(c) Compute:

$(0, a) \bigvee (0, b)$

$(0, a) \bigwedge (0, b)$

$(1, a) \bigvee (1, b)$

$(1, a) \bigwedge (1, b)$

$(0, 1) \bigvee (1, 0)$

and $(0, 1) \bigwedge (1, 0)$.

Use this information as an aid to draw the Hasse diagram for $L_1 \times L_2$.

7. (a) Is $A = \{1, 2, 3, \ldots, 12\}$ a lattice under the relation “divides”? Explain.

(b) Is the set A above a lattice under the relation “less than or equal to”? Explain.

Section 13.3

8. Using the rules of Boolean algebra, reduce the expression $\overline{(x_1 \vee x_2)} \bigvee (\overline{x_1} \wedge x_2) \bigvee (x_1 \wedge x_2)$ to the equivalent expression $\overline{x_1} \vee x_2$. Justify each step.

9. Using the rules of Boolean algebra, reduce the expression $(x + y) \cdot (x + \overline{y})$ to a simpler expression.

10. Even a cursory examination of the basic laws for Boolean algebra (Table 13.3.1), for logic (Table 3.4.1), and for sets (Section 4.2) will indicate that they are the same in three different languages: they are isomorphic to one another as Boolean algebras.

(a) Fill out the following table to illustrate the above concept:

	comparable connectives			
Sets	$\cup$			
Logic			$\wedge$	$\neg$
Boolean Algebraa		$\leq$		

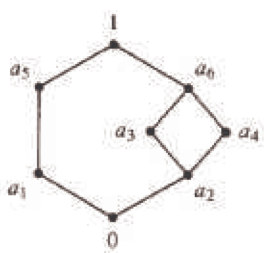
(b) Since the above algebras are isomorphic as Boolean algebras, any theorem true in one is true in the other two. Translate each of the following statements into the language of the other two.

(i) $p \to q$ if and only if $\neg q \to \neg p$.

(ii) If $A \subseteq B$ and $A \subseteq C$ then $A \subseteq B \cap C$

(iii) If $a \geq b$ and $a \geq c$ then $a \geq b \vee c$.

11. (a) Determine the complements of each element described by the following Hasse diagram:



(b) Is the above lattice a Boolean algebra?

12. (a) Determine the complement of each element in the lattice D_{50}.

(b) Is D_{50} a Boolean algebra? Explain.

Section 13.4

13. (a) Use the Theorem 13.4.2 and its Corollaries to determine which ofthe following are Boolean algebras:

(a) D_{20} (b) D_{27} (c) D_{35} (d) D_{210}

(b) Notice that D_n is a Boolean algebra if and only if n is a product of distinct primes. Such an integer is called *square free*. What are the atoms of D_n if n is square free?

14. Let $[B, -, \vee . \wedge]$ be any Boolean algebra of order 8. Find a Boolean algebra of sets that is isomorphic to B. How many atoms must B have?

Section 13.5

15. (a) List all sub-Boolean algebras of order 4 in ${B_2}^3$

(b) How many sub-Boolean algebras of order 4 are there in ${B_2}^n$, $n \geq 4$?

(c) Discuss how the selection of atoms in a sub-Boolean algebra can be used to answer questions such as the one in part (b).

16. Prove that Boolean algebras ${B_2}^m \times {B_2}^n$ and ${B_2}^{m+n}$ are isomorphic.

Section 13.6

17 Find the minterm normal form of the Boolean expression $(\overline{x_1} \vee x_2) \wedge x_3$

18. Find the rninterm normal form of the Boolean expression

$$x_4 \wedge (x_3 \vee x_2 \vee x_1) \vee x_3 \wedge (x_2 \vee x_1) \vee x_2 \wedge x_1$$

19. Let B be a Boolean algebra of order 2.

(a) How many rows are there in the table of a Boolean function of 3 variables? Of n variables?

(b) How many different Boolean functions of 3 variables and of *n* variables are there?

20. Let *B* be a Boolean algebra of order 2.

(a) How many different minterm normal forms are there for Boolean expressions of 2 variables over *B*? List them.

(b) How many different minterm normal forms are there for Boolean expressions of 3 variables over *B*?

Section 13.7

21. Consider the following Boolean expression:

$$f(x_1, x_2, x_3) = ((x_1 + x_2 + x_3)\cdot\overline{x_1} + x_1 + \overline{x_2})\cdot x_1 \cdot \overline{x_3}$$

(a) Draw the switching circuit of *f*.

(b) Draw the gate diagram of *f*.

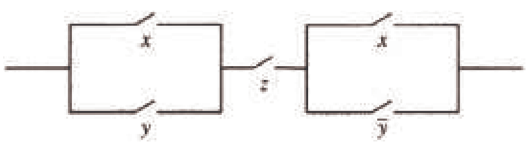
(c) Simplify *f* algebraically and draw the switching circuit and gate diagrams of this simplified version of *f*.

22. Assume that each of the three members of a committee votes *yes* or *no* on a proposal by pressing a button that closes a switch for *yes* and doesn nothing for *no*. Devise as simple a switching-circuit as you can that will allow current to pass when and only when at least two of the members vote in the affirmative.

23. (a) Find the Boolean function of this network:

(b) Draw an equivalent

24. Given the switching circuit



(a) Express the the switching circuit algebraically.

(b) Draw the gate diagram of the expression obtained in part a.

(c) Simplify the expression in part a and draw the switching-circuit and gate diagram for the simplified expression.