3-1 BOOLEAN CONSTANTS AND VARIABLES

Boolean algebra differs in a major way from ordinary algebra because Boolean constants and variables are allowed to have only two possible values, 0 or 1. A Boolean variable is a quantity that may, at different times, be equal to either 0 or 1. Boolean variables are often used to represent the voltage level present on a wire or at the input/output terminals of a circuit. For example, in a certain digital system, the Boolean value of 0 might be assigned to any voltage in the range from 0 to 0.8 V, while the Boolean value of 1 might be assigned to any voltage in the range 2 to 5 V.*

Thus, Boolean 0 and 1 do not represent actual numbers but instead represent the state of a voltage variable, or what is called its **logic level**. A voltage in a digital circuit is said to be at the logic 0 level or the logic 1 level, depending on its actual numerical value. In digital logic, several other terms are used synonymously with 0 and 1. Some of the more common ones are shown in Table 3-1. We will use the 0/1 and LOW/HIGH designations most of the time.

TABLE 3-1

Logic 0	Logic 1
False	True
Off	On
Low	High
No	Yes
Open switch	Closed switch

As we said in the introduction, **Boolean algebra** is a means for expressing the relationship between a logic circuit's inputs and outputs. The inputs are considered logic variables whose logic levels at any time determine the output levels. In all our work to follow, we shall use letter symbols to represent logic variables. For example, the letter A might represent a certain digital circuit input or output, and at any time we must have either $A = 0$ or $A = 1$: if not one, then the other.

Because only two values are possible, Boolean algebra is relatively easy to work with compared with ordinary algebra. In Boolean algebra, there are no fractions, decimals, negative numbers, square roots, cube roots, logarithms, imaginary numbers, and so on. In fact, in Boolean algebra there are only *three* basic operations: *OR*, *AND*, and *NOT*.

These basic operations are called *logic operations*. Digital circuits called *logic gates* can be constructed from diodes, transistors, and resistors connected so that the circuit output is the result of a basic logic operation *(OR, AND, NOT)* performed on the inputs. We will be using Boolean algebra first to describe and analyze these basic logic gates, then later to analyze and design combinations of logic gates connected as logic circuits.

3-2 TRUTH TABLES

A **truth table** is a means for describing how a logic circuit's output depends on the logic levels present at the circuit's inputs. Figure 3-1(a) illustrates a truth table for one type of two-input logic circuit. The table lists all possible

*Voltages between 0.8 and 2 V are undefined (neither 0 nor 1) and should not occur under normal circumstances.

EXAMPLE 3-11

Implement the logic circuit that has the expression $x = \overline{AB \cdot (\overline{C + D})}$ using only NOR and NAND gates.

Solution

The $(\overline{C + D})$ term is the expression for the output of a NOR gate. This term is ANDed with A and B, and the result is inverted; this, of course, is the NAND operation. Thus, the circuit is implemented as shown in Figure 3-24. Note that the NAND gate first ANDs the A, B, and $(\overline{C + D})$ terms, and then it inverts the *complete* result.

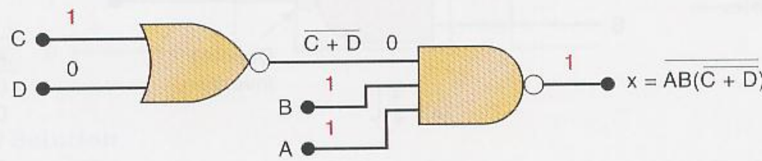


FIGURE 3-24 Examples 3-11 and 3-12.

EXAMPLE 3-12

Determine the output level in Figure 3-24 for $A = B = C = 1$ and $D = 0$.

Solution

In the first method we use the expression for x.

$$\begin{aligned} x &= \overline{AB(\overline{C + D})} \\ &= \overline{1 \cdot 1 \cdot (\overline{1 + 0})} \\ &= \overline{1 \cdot 1 \cdot (\overline{1})} \\ &= \overline{1 \cdot 1 \cdot 0} \\ &= \overline{0} = 1 \end{aligned}$$

In the second method, we write down the input logic levels on the circuit diagram (shown in color in Figure 3-24) and follow these levels through each gate to the final output. The NOR gate has inputs of 1 and 0 to produce an output of 0 (an OR would have produced an output of 1). The NAND gate thus has input levels of 0, 1, and 1 to produce an output of 1 (an AND would have produced an output of 0).

REVIEW QUESTIONS

1. What is the only set of input conditions that will produce a HIGH output from a three-input NOR gate?
2. Determine the output level in Figure 3-24 for $A = B = 1$, $C = D = 0$.
3. Change the NOR gate of Figure 3-24 to a NAND gate, and change the NAND to a NOR. What is the new expression for x?

3-10 BOOLEAN THEOREMS

We have seen how Boolean algebra can be used to help analyze a logic circuit and express its operation mathematically. We will continue our study of Boolean algebra by investigating the various **Boolean theorems** (rules) that can help us to simplify logic expressions and logic circuits. The first group of theorems is given in Figure 3-25. In each theorem, x is a logic variable that

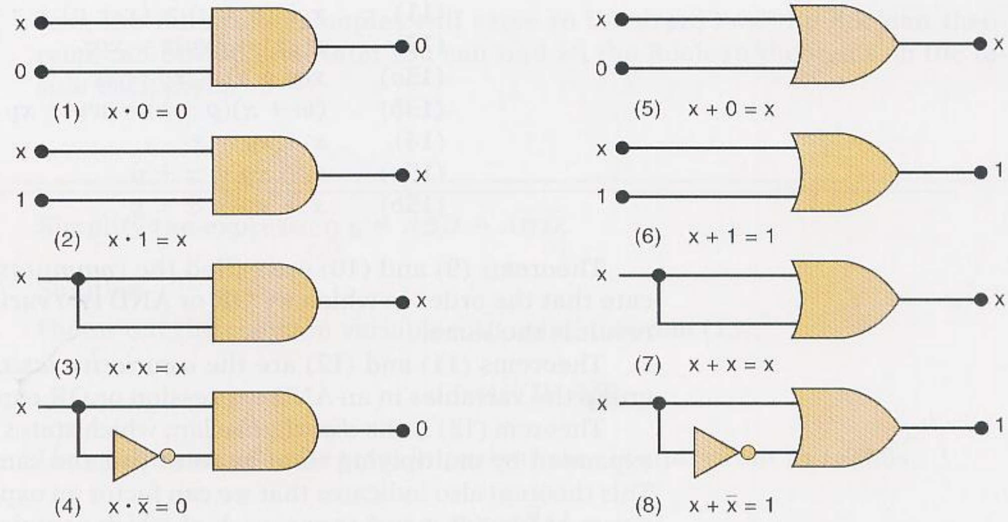


FIGURE 3-25 Single-variable theorems.

can be either a 0 or a 1. Each theorem is accompanied by a logic-circuit diagram that demonstrates its validity.

Theorem (1) states that if any variable is ANDed with 0, the result must be 0. This is easy to remember because the AND operation is just like ordinary multiplication, where we know that anything multiplied by 0 is 0. We also know that the output of an AND gate will be 0 whenever any input is 0, regardless of the level on the other input.

Theorem (2) is also obvious by comparison with ordinary multiplication.

Theorem (3) can be proved by trying each case. If $x = 0$, then $0 \cdot 0 = 0$; if $x = 1$, then $1 \cdot 1 = 1$. Thus, $x \cdot x = x$.

Theorem (4) can be proved in the same manner. However, it can also be reasoned that at any time either x or its inverse $\overline{x}$ must be at the 0 level, and so their AND product always must be 0.

Theorem (5) is straightforward, since 0 *added* to anything does not affect its value, either in regular addition or in OR addition.

Theorem (6) states that if any variable is ORed with 1, the result will always be 1. We check this for both values of x: $0 + 1 = 1$ and $1 + 1 = 1$. Equivalently, we can remember that an OR gate output will be 1 when *any* input is 1, regardless of the value of the other input.

Theorem (7) can be proved by checking for both values of x: $0 + 0 = 0$ and $1 + 1 = 1$.

Theorem (8) can be proved similarly, or we can just reason that at any time either x or $\overline{x}$ must be at the 1 level so that we are always ORing a 0 and a 1, which always results in 1.

Before introducing any more theorems, we should point out that when theorems (1) through (8) are applied, the variable x may actually represent an expression containing more than one variable. For example, if we have $A\overline{B}(\overline{A\overline{B}})$, we can invoke theorem (4) by letting $x = A\overline{B}$. Thus, we can say that $A\overline{B}(\overline{A\overline{B}}) = 0$. The same idea can be applied to the use of any of these theorems.

Multivariable Theorems

The theorems presented below involve more than one variable:

$$(9) \quad x + y = y + x$$
$$(10) \quad x \cdot y = y \cdot x$$

(11) $x + (y + z) = (x + y) + z = x + y + z$
(12) $x(yz) = (xy)z = xyz$
(13a) $x(y + z) = xy + xz$
(13b) $(w + x)(y + z) = wy + xy + wz + xz$
(14) $x + xy = x$
(15a) $x + \overline{x}y = x + y$
(15b) $\overline{x} + xy = \overline{x} + y$

Theorems (9) and (10) are called the *commutative laws.* These laws indicate that the order in which we OR or AND two variables is unimportant; the result is the same.

Theorems (11) and (12) are the *associative laws,* which state that we can group the variables in an AND expression or OR expression any way we want.

Theorem (13) is the *distributive law,* which states that an expression can be expanded by multiplying term by term just the same as in ordinary algebra. This theorem also indicates that we can factor an expression. That is, if we have a sum of two (or more) terms, each of which contains a common variable, the common variable can be factored out just as in ordinary algebra. For example, if we have the expression $A\overline{B}C + \overline{A}\,\overline{B}\,\overline{C}$, we can factor out the $\overline{B}$ variable:

$$A\overline{B}C + \overline{A}\,\overline{B}\,\overline{C} = \overline{B}(AC + \overline{A}\,\overline{C})$$

As another example, consider the expression $ABC + ABD$. Here the two terms have the variables A and B in common, and so $A \cdot B$ can be factored out of both terms. That is,

$$ABC + ABD = AB(C + D)$$

Theorems (9) to (13) are easy to remember and use because they are identical to those of ordinary algebra. Theorems (14) and (15), on the other hand, do not have any counterparts in ordinary algebra. Each can be proved by trying all possible cases for x and y. This is illustrated (for theorem 14) by creating an analysis table for the equation $x + xy$ as follows:

x	y	xy	x + xy
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Notice that the value of the entire expression $(x + xy)$ is always the same as x.

Theorem (14) can also be proved by factoring and using theorems (6) and (2) as follows:

$$\begin{aligned} x + xy &= x(1 + y) \\ &= x \cdot 1 \qquad \text{[using theorem (6)]} \\ &= x \qquad \text{[using theorem (2)]} \end{aligned}$$

All of these Boolean theorems can be useful in simplifying a logic expression—that is, in reducing the number of terms in the expression. When this is done, the reduced expression will produce a circuit that is less complex than the one that the original expression would have produced. A good portion of the next chapter will be devoted to the process of circuit simplification. For now, the following examples will serve to illustrate how the Boolean theorems can be applied. **Note:** You can find all the Boolean theorems on the inside back cover.

EXAMPLE 3-13

Simplify the expression $y = A\overline{B}D + A\overline{B}\,\overline{D}$.

Solution

Factor out the common variables $A\overline{B}$ using theorem (13):

$$y = A\overline{B}(D + \overline{D})$$

Using theorem (8), the term in parentheses is equivalent to 1. Thus,

$$\begin{aligned} y &= A\overline{B} \cdot 1 \\ &= A\overline{B} \qquad \text{[using theorem (2)]} \end{aligned}$$

EXAMPLE 3-14

Simplify $z = (\overline{A} + B)(A + B)$.

Solution

The expression can be expanded by multiplying out the terms [theorem (13)]:

$$z = \overline{A} \cdot A + \overline{A} \cdot B + B \cdot A + B \cdot B$$

Invoking theorem (4), the term $\overline{A} \cdot A = 0$. Also, $B \cdot B = B$ [theorem (3)]:

$$z = 0 + \overline{A} \cdot B + B \cdot A + B = \overline{A}B + AB + B$$

Factoring out the variable B [theorem (13)], we have

$$z = B(\overline{A} + A + 1)$$

Finally, using theorems (2) and (6),

$$z = B$$

EXAMPLE 3-15

Simplify $x = ACD + \overline{A}BCD$.

Solution

Factoring out the common variables CD, we have

$$x = CD(A + \overline{A}B)$$

Utilizing theorem (15a), we can replace $A + \overline{A}B$ by $A + B$, so

$$\begin{aligned} x &= CD(A + B) \\ &= ACD + BCD \end{aligned}$$

REVIEW QUESTIONS

1. Use theorems (13) and (14) to simplify $y = A\overline{C} + AB\overline{C}$.
2. Use theorems (13) and (8) to simplify $y = \overline{A}BC\overline{D} + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$.
3. Use theorems (13) and (15b) to simplify $y = \overline{A}D + ABD$.

3-11 DEMORGAN'S THEOREMS

Two of the most important theorems of Boolean algebra were contributed by a great mathematician named DeMorgan. **DeMorgan's theorems** are extremely useful in simplifying expressions in which a product or sum of variables is inverted. The two theorems are:

$$(16)\quad \overline{(x + y)} = \overline{x} \cdot \overline{y}$$
$$(17)\quad \overline{(x \cdot y)} = \overline{x} + \overline{y}$$

Theorem (16) says that when the OR sum of two variables is inverted, this is the same as inverting each variable individually and then ANDing these inverted variables. Theorem (17) says that when the AND product of two variables is inverted, this is the same as inverting each variable individually and then ORing them. Each of DeMorgan's theorems can readily be proven by checking for all possible combinations of x and y. This will be left as an end-of-chapter exercise.

Although these theorems have been stated in terms of single variables x and y, they are equally valid for situations where x and/or y are expressions that contain more than one variable. For example, let's apply them to the expression $\overline{(A\overline{B} + C)}$ as shown below:

$$\overline{(A\overline{B} + C)} = \overline{(A\overline{B})} \cdot \overline{C}$$

Note that we used theorem (16) and treated $A\overline{B}$ as x and C as y. The result can be further simplified because we have a product $A\overline{B}$ that is inverted. Using theorem (17), the expression becomes

$$\overline{A\overline{B}} \cdot \overline{C} = (\overline{A} + \overline{\overline{B}}) \cdot \overline{C}$$

Notice that we can replace $\overline{\overline{B}}$ by B, so that we finally have

$$(\overline{A} + B) \cdot \overline{C} = \overline{A}\,\overline{C} + B\overline{C}$$

This final result contains only inverter signs that invert a single variable.

EXAMPLE 3-16

Simplify the expression $z = \overline{(\overline{A} + C) \cdot (B + \overline{D})}$ to one having only single variables inverted.

Solution

Using theorem (17), and treating $(\overline{A} + C)$ as x and $(B + \overline{D})$ as y, we have

$$z = \overline{(\overline{A} + C)} + \overline{(B + \overline{D})}$$

We can think of this as breaking the large inverter sign down the middle and changing the AND sign (·) to an OR sign (+). Now the term $(\overline{\overline{A} + C})$ can be simplified by applying theorem (16). Likewise, $(\overline{B + \overline{D}})$ can be simplified:

$$\begin{aligned} z &= (\overline{\overline{A} + C}) + (\overline{B + \overline{D}}) \\ &= (\overline{\overline{A}} \cdot \overline{C}) + \overline{B} \cdot \overline{\overline{D}} \end{aligned}$$

Here we have broken the larger inverter signs down the middle and replaced the (+) with a (·). Canceling out the double inversions, we have finally

$$z = A\overline{C} + \overline{B}D$$

Example 3-16 points out that when using DeMorgan's theorems to reduce an expression, we may break an inverter sign at any point in the expression and change the operator sign at that point in the expression to its opposite (+ is changed to ·, and vice versa). This procedure is continued until the expression is reduced to one in which only single variables are inverted. Two more examples are given below.

Example 1

$$\begin{aligned} z &= \overline{A + \overline{B} \cdot C} \\ &= \overline{A} \cdot (\overline{\overline{B} \cdot C}) \\ &= \overline{A} \cdot (\overline{\overline{B}} + \overline{C}) \\ &= \overline{A} \cdot (B + \overline{C}) \end{aligned}$$

Example 2

$$\begin{aligned} \omega &= \overline{(A + BC) \cdot (D + EF)} \\ &= (\overline{A + BC}) + (\overline{D + EF}) \\ &= (\overline{A} \cdot \overline{BC}) + (\overline{D} \cdot \overline{EF}) \\ &= [\overline{A} \cdot (\overline{B} + \overline{C})] + [\overline{D} \cdot (\overline{E} + \overline{F})] \\ &= \overline{A}\,\overline{B} + \overline{A}\,\overline{C} + \overline{D}\,\overline{E} + \overline{D}\,\overline{F} \end{aligned}$$

DeMorgan's theorems are easily extended to more than two variables. For example, it can be proved that

$$\begin{aligned} \overline{x + y + z} &= \overline{x} \cdot \overline{y} \cdot \overline{z} \\ \overline{x \cdot y \cdot z} &= \overline{x} + \overline{y} + \overline{z} \end{aligned}$$

Here, we see that the large inverter sign is broken at *two* points in the expression and the operator sign is changed to its opposite. This can be extended to any number of variables. Again, realize that the variables can themselves be expressions rather than single variables. Here is another example.

$$\begin{aligned} x &= \overline{\overline{AB} \cdot \overline{CD} \cdot \overline{EF}} \\ &= \overline{\overline{AB}} + \overline{\overline{CD}} + \overline{\overline{EF}} \\ &= AB + CD + EF \end{aligned}$$

Implications of DeMorgan's Theorems

Let us examine theorems (16) and (17) from the standpoint of logic circuits. First, consider theorem (16):

$$\overline{x + y} = \overline{x} \cdot \overline{y}$$

The left-hand side of the equation can be viewed as the output of a NOR gate whose inputs are x and y. The right-hand side of the equation, on the other

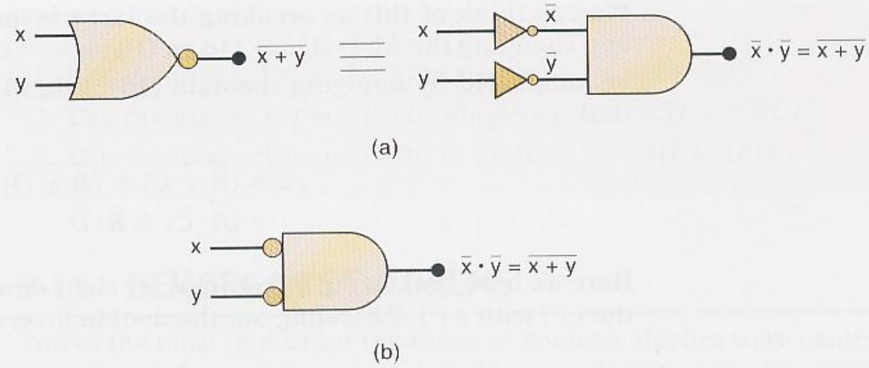


FIGURE 3-26 (a) Equivalent circuits implied by theorem (16); (b) alternative symbol for the NOR function.

hand, is the result of first inverting both x and y and then putting them through an AND gate. These two representations are equivalent and are illustrated in Figure 3-26(a). What this means is that an AND gate with INVERTERs on each of its inputs is equivalent to a NOR gate. In fact, both representations are used to represent the NOR function. When the AND gate with inverted inputs is used to represent the NOR function, it is usually drawn as shown in Figure 3-26(b), where the small circles on the inputs represent the inversion operation.

Now consider theorem (17):

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

The left side of the equation can be implemented by a NAND gate with inputs x and y. The right side can be implemented by first inverting inputs x and y and then putting them through an OR gate. These two equivalent representations are shown in Figure 3-27(a). The OR gate with INVERTERs on each of its inputs is equivalent to the NAND gate. In fact, both representations are used to represent the NAND function. When the OR gate with inverted inputs is used to represent the NAND function, it is usually drawn as shown in Figure 3-27(b), where the circles again represent inversion.

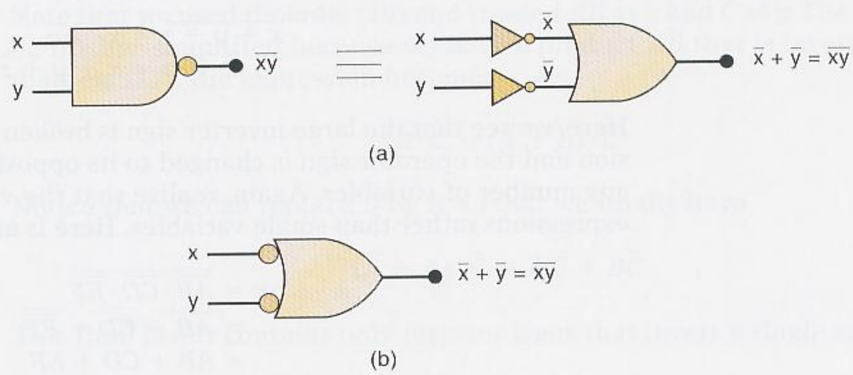


FIGURE 3-27 (a) Equivalent circuits implied by theorem (17); (b) alternative symbol for the NAND function.

EXAMPLE 3-17

Determine the output expression for the circuit of Figure 3-28 and simplify it using DeMorgan's theorems.

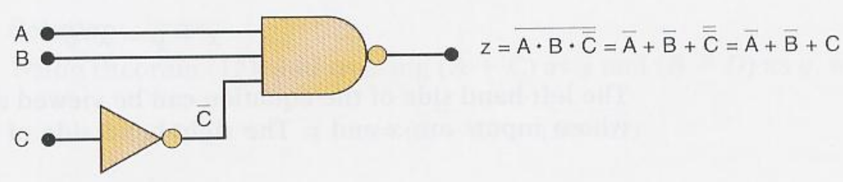


FIGURE 3-28 Example 3-17.

Solution

The expression for z is $z = \overline{AB\overline{C}}$. Use DeMorgan's theorem to break the large inversion sign:

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

Cancel the double inversions over C to obtain

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

REVIEW QUESTIONS

1. Use DeMorgan's theorems to convert the expression $z = \overline{(A + B)\cdot\overline{C}}$ to one that has only single-variable inversions.
2. Repeat question 1 for the expression $y = \overline{R\overline{S}T + \overline{Q}}$.
3. Implement a circuit having output expression $z = \overline{\overline{A}\,\overline{B}C}$ using only a NOR gate and an INVERTER.
4. Use DeMorgan's theorems to convert $y = \overline{\overline{A} + \overline{B} + \overline{C}D}$ to an expression containing only single-variable inversions.

3-12 UNIVERSALITY OF NAND GATES AND NOR GATES

All Boolean expressions consist of various combinations of the basic operations of OR, AND, and INVERT. Therefore, any expression can be implemented using combinations of OR gates, AND gates, and INVERTERs. It is possible, however, to implement any logic expression using *only* NAND gates and no other type of gate. This is because NAND gates, in the proper combination, can be used to perform each of the Boolean operations OR, AND, and INVERT. This is demonstrated in Figure 3-29.

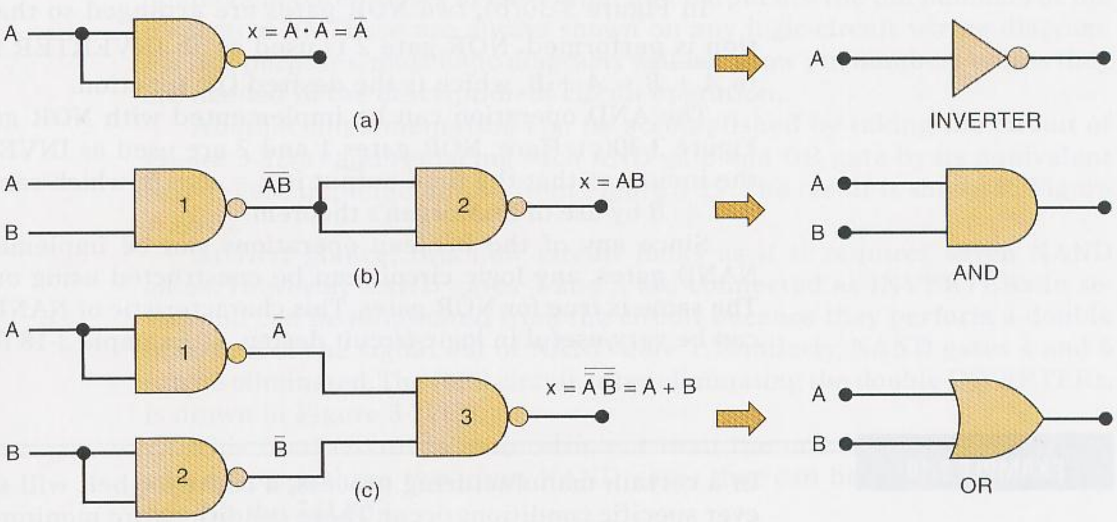


FIGURE 3-29 NAND gates can be used to implement any Boolean function.

With the expression now in SOP form, we should look for common variables among the various terms with the intention of factoring. The first and third terms above have AC in common, which can be factored out:

$$z = AC(B + \overline{B}) + A\overline{B}$$

Since $B + \overline{B} = 1$, then

$$\begin{aligned} z &= AC(1) + A\overline{B} \\ &= AC + A\overline{B} \end{aligned}$$

We can now factor out A, which results in

$$z = A(C + \overline{B})$$

This result can be simplified no further. Its circuit implementation is shown in Figure 4-2(b). It is obvious that the circuit in Figure 4-2(b) is a great deal simpler than the original circuit in Figure 4-2(a).

EXAMPLE 4-2

Simplify the expression $z = A\overline{B}\,\overline{C} + A\overline{B}C + ABC$.

Solution

The expression is already in SOP form.

Method 1: The first two terms in the expression have the product $A\overline{B}$ in common. Thus,

$$\begin{aligned} z &= A\overline{B}(\overline{C} + C) + ABC \\ &= A\overline{B}(1) + ABC \\ &= A\overline{B} + ABC \end{aligned}$$

We can factor the variable A from both terms:

$$z = A(\overline{B} + BC)$$

Invoking theorem (15b):

$$z = A(\overline{B} + C)$$

Method 2: The original expression is $z = A\overline{B}\,\overline{C} + A\overline{B}C + ABC$. The first two terms have $A\overline{B}$ in common. The last two terms have AC in common. How do we know whether to factor $A\overline{B}$ from the first two terms or AC from the last two terms? Actually, we can do both by using the $A\overline{B}C$ term *twice.* In other words, we can rewrite the expression as:

$$z = A\overline{B}\,\overline{C} + A\overline{B}C + A\overline{B}C + ABC$$

where we have added an extra term $A\overline{B}C$. This is valid and will not change the value of the expression because $A\overline{B}C + A\overline{B}C = A\overline{B}C$ [theorem (7)]. Now we can factor $A\overline{B}$ from the first two terms and AC from the last two terms:

$$\begin{aligned} z &= A\overline{B}(C + \overline{C}) + AC(\overline{B} + B) \\ &= A\overline{B}\cdot 1 + AC\cdot 1 \\ &= A\overline{B} + AC = A(\overline{B} + C) \end{aligned}$$

Of course, this is the same result obtained with method 1. This trick of using the same term twice can always be used. In fact, the same term can be used more than twice if necessary.

EXAMPLE 4-3

Simplify $z = \overline{A}C(\overline{\overline{A}BD}) + \overline{A}B\overline{C}\,\overline{D} + A\overline{B}C$.

Solution

First, use DeMorgan's theorem on the first term:

$$z = \overline{A}C(A + \overline{B} + \overline{D}) + \overline{A}B\overline{C}\,\overline{D} + A\overline{B}C \qquad \textbf{(step 1)}$$

Multiplying out yields

$$z = \overline{A}CA + \overline{A}C\overline{B} + \overline{A}C\overline{D} + \overline{A}B\overline{C}\,\overline{D} + A\overline{B}C \qquad (2)$$

Because $\overline{A} \cdot A = 0$, the first term is eliminated:

$$z = \overline{A}\,\overline{B}C + \overline{A}C\overline{D} + \overline{A}B\overline{C}\,\overline{D} + A\overline{B}C \qquad (3)$$

This is the desired SOP form. Now we must look for common factors among the various product terms. The idea is to check for the largest common factor between any two or more product terms. For example, the first and last terms have the common factor $\overline{B}C$, and the second and third terms share the common factor $\overline{A}\,\overline{D}$. We can factor these out as follows:

$$z = \overline{B}C(\overline{A} + A) + \overline{A}\,\overline{D}(C + B\overline{C}) \qquad (4)$$

Now, because $\overline{A} + A = 1$, and $C + B\overline{C} = C + B$ [theorem (15a)], we have

$$z = \overline{B}C + \overline{A}\,\overline{D}(B + C) \qquad (5)$$

This same result could have been reached with other choices for the factoring. For example, we could have factored C from the first, second, and fourth product terms in step 3 to obtain

$$z = C(\overline{A}\,\overline{B} + \overline{A}\,\overline{D} + A\overline{B}) + \overline{A}B\overline{C}\,\overline{D}$$

The expression inside the parentheses can be factored further:

$$z = C(\overline{B}[\overline{A} + A] + \overline{A}\,\overline{D}) + \overline{A}B\overline{C}\,\overline{D}$$

Because $\overline{A} + A = 1$, this becomes

$$z = C(\overline{B} + \overline{A}\,\overline{D}) + \overline{A}B\overline{C}\,\overline{D}$$

Multiplying out yields

$$z = \overline{B}C + \overline{A}C\overline{D} + \overline{A}B\overline{C}\,\overline{D}$$

Now we can factor $\overline{A}\,\overline{D}$ from the second and third terms to get

$$z = \overline{B}C + \overline{A}\,\overline{D}(C + B\overline{C})$$

If we use theorem (15a), the expression in parentheses becomes $B + C$. Thus, we finally have

$$z = \overline{B}C + \overline{A}\,\overline{D}(B + C)$$

This is the same result that we obtained earlier, but it took us many more steps. This illustrates why you should look for the largest common factors: it will generally lead to the final expression in the fewest steps.

Example 4-3 illustrates the frustration often encountered in Boolean simplification. Because we have arrived at the same equation (which appears irreducible) by two different methods, it might seem reasonable to conclude that this final equation is the simplest form. In fact, the simplest form of this equation is

$$z = \overline{A}B\overline{D} + \overline{B}C$$

But there is no apparent way to reduce step (5) to reach this simpler version. In this case, we missed an operation earlier in the process that could have led to the simpler form. The question is, "How could we have known that we missed a step?" Later in this chapter, we will examine a mapping technique that will always lead to the simplest SOP form.

EXAMPLE 4-4

Simplify the expression $x = (\overline{A} + B)(A + B + D)\overline{D}$.

Solution

The expression can be put into sum-of-products form by multiplying out all the terms. The result is

$$x = \overline{A}A\overline{D} + \overline{A}B\overline{D} + \overline{A}D\overline{D} + BA\overline{D} + BB\overline{D} + BD\overline{D}$$

The first term can be eliminated because $\overline{A}A = 0$. Likewise, the third and sixth terms can be eliminated because $D\overline{D} = 0$. The fifth term can be simplified to $B\overline{D}$ because $BB = B$. This gives us

$$x = \overline{A}B\overline{D} + AB\overline{D} + B\overline{D}$$

We can factor $B\overline{D}$ from each term to obtain

$$x = B\overline{D}(\overline{A} + A + 1)$$

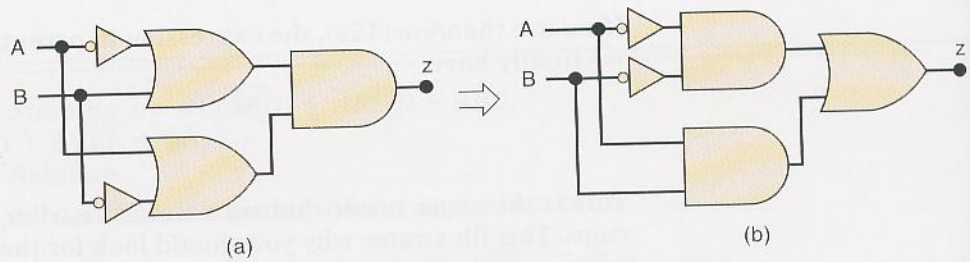
Clearly, the term inside the parentheses is always 1, so we finally have

$$x = B\overline{D}$$

EXAMPLE 4-5

Simplify the circuit of Figure 4-3(a).

FIGURE 4-3 Example 4-5.



Solution

The expression for output z is

$$z = (\overline{A} + B)(A + \overline{B})$$

Multiplying out to get the sum-of-products form, we obtain

$$z = \overline{A}A + \overline{A}\,\overline{B} + BA + B\overline{B}$$

We can eliminate $\overline{A}A = 0$ and $B\overline{B} = 0$ to end up with

$$z = \overline{A}\,\overline{B} + AB$$

This expression is implemented in Figure 4-3(b), and if we compare it with the original circuit, we see that both circuits contain the same number of gates and connections. In this case, the simplification process produced an equivalent, but not simpler, circuit.

EXAMPLE 4-6

Simplify $x = A\overline{B}C + \overline{A}BD + \overline{C}\,\overline{D}$.

Solution

You can try, but you will not be able to simplify this expression any further.

REVIEW QUESTIONS

1. State which of the following expressions are *not* in the sum-of-products form:
   (a) $RS\overline{T} + \overline{R}S\overline{T} + \overline{T}$
   (b) $A\overline{DC} + \overline{A}DC$
   (c) $MN\overline{P} + (M + \overline{N})P$
   (d) $AB + \overline{A}B\overline{C} + A\,\overline{B}\,\overline{C}D$
2. Simplify the circuit in Figure 4-1(a) to arrive at the circuit of Figure 4-1(b).
3. Change each AND gate in Figure 4-1(a) to a NAND gate. Determine the new expression for x and simplify it.