model of a sequential UDP requires that its output be declared as a **reg** data type, and that a column be added to the truth table to describe the next state. So the columns are organizes as inputs : state : next state.

In this section, we introduced the Verilog HDL and presented simple examples to illustrate alternatives for modeling combinational logic. A more detailed presentation of Verilog HDL can be found in the next chapter. The reader familiar with combinational circuits can go directly to Section 4.12 to continue with this subject.

PROBLEMS

Answers to problems marked with * appear at the end of the book.

3.1* Simplify the following Boolean functions, using three-variable maps:
(a) $F(x, y, z) = \Sigma(0, 2, 6, 7)$ (b) $F(x, y, z) = \Sigma(0, 2, 3, 4, 6)$
(c) $F(x, y, z) = \Sigma(0, 1, 2, 3, 7)$ (d) $F(x, y, z) = \Sigma(3, 5, 6, 7)$

3.2 Simplify the following Boolean functions, using three-variable maps:
(a)* $F(x, y, z) = \Sigma(0, 1, 5, 7)$ (b)* $F(x, y, z) = \Sigma(1, 2, 3, 6, 7)$
(c) $F(x, y, z) = \Sigma(0, 1, 6, 7)$ (d) $F(x, y, z) = \Sigma(0, 1, 3, 4, 5)$
(e) $F(x, y, z) = \Sigma(1, 3, 5, 7)$ (f) $F(x, y, z) = \Sigma(1, 4, 5, 6, 7)$

3.3* Simplify the following Boolean expressions, using three-variable maps:
(a)* $F(x,y,z) = xy + x'y'z' + x'yz'$ (b)* $F(x,y,z) = x'y' + yz + x'yz'$
(c)* $F(x, y, z) = x'y + yz' + y'z'$ (d) $F(x, y, z) = xyz + x'y'z + xy'z'$

3.4 Simplify the following Boolean functions, using *Karnaugh* maps:
(a)* $F(x, y, z) = \Sigma(2, 3, 6, 7)$ (b)* $F(A, B, C, D) = \Sigma(4, 6, 7, 15)$
(c)* $F(A, B, C, D) = \Sigma(3, 7, 11, 13, 14, 15)$ (d)* $F(w, x, y, z) = \Sigma(2, 3, 12, 13, 14, 15)$
(e) $F(w, x, y, z) = \Sigma(1, 4, 5, 6, 7, 13)$ (f) $F(w, x, y, z) = \Sigma(0, 1, 5, 8, 9)$

3.5 Simplify the following Boolean functions, using four-variable maps:
(a)* $F(w, x, y, z) = \Sigma(1, 4, 5, 6, 12, 14, 15)$
(b) $F(A, B, C, D) = \Sigma(1, 5, 9, 10, 11, 14, 15)$
(c) $F(w, x, y, z) = \Sigma(0, 1, 4, 5, 6, 7, 8, 9)$
(d)* $F(A, B, C, D) = \Sigma(0, 2, 4, 5, 6, 7, 8, 10, 13, 15)$

3.6 Simplify the following Boolean expressions, using four-variable maps:
(a)* $A'B'C'D' + AC'D' + B'CD' + A'BCD + BC'D$
(b)* $x'z + w'xy' + w(x'y + xy')$
(c) $A'B'C'D' + A'CD' + AB'D' + ABCD + A'BD$
(d) $A'B'C'D' + AB'C + B'CD' + ABCD' + BC'D$

3.7 Simplify the following Boolean expressions, using four-variable maps:
(a)* $w'z + xz + x'y + wx'z$
(b) $C'D + A'B'C + ABC' + AB'C$
(c)* $AB'C + B'C'D' + BCD + ACD' + A'B'C + A'BC'D$
(d) $xyz + wy + wxy' + x'y$

3.8 Find the minterms of the following Boolean expressions by first plotting each function in a map:
(a)* $xy + yz + xy'z$ (b)* $C'D + ABC' + ABD' + A'B'D$
(c) $wyz + w'x' + wxz'$ (d) $A'B + A'CD + B'CD + BC'D'$

3.9 Find all the prime implicants for the following Boolean functions, and determine which are essential:

(a)* $F(w, x, y, z) = \Sigma(0, 2, 4, 5, 6, 7, 8, 10, 13, 15)$
(b)* $F(A, B, C, D) = \Sigma(0, 2, 3, 5, 7, 8, 10, 11, 14, 15)$
(c) $F(A, B, C, D) = \Sigma(1, 3, 4, 5, 10, 11, 12, 13, 14, 15)$
(d) $F(w, x, y, z) = \Sigma(1, 3, 6, 7, 8, 9, 12, 13, 14, 15)$
(e) $F(A, B, C, D) = \Sigma(0, 2, 3, 5, 7, 8, 10, 11, 13, 15)$
(f) $F(w, x, y, z) = \Sigma(0, 2, 7, 8, 9, 10, 12, 13, 14, 15)$

3.10 Simplify the following Boolean functions by first finding the essential prime implicants:

(a) $F(w, x, y, z) = \Sigma(0, 2, 4, 5, 6, 7, 8, 10, 13, 15)$
(b) $F(A, B, C, D) = \Sigma(0, 2, 3, 5, 7, 8, 10, 11, 14, 15)$
(c)* $F(A, B, C, D) = \Sigma(1, 3, 4, 5, 10, 11, 12, 13, 14, 15)$
(d) $F(w, x, y, z) = \Sigma(1, 3, 6, 7, 8, 9, 12, 13, 14, 15)$
(e) $F(A, B, C, D) = \Sigma(0, 2, 3, 5, 7, 8, 10, 11, 13, 15)$
(f) $F(w, x, y, z) = \Sigma(0, 2, 7, 8, 9, 10, 12, 13, 14, 15)$

3.11 Simplify the following Boolean functions, using five-variable maps:

(a)* $F(A, B, C, D, E) = \Sigma(0, 1, 4, 5, 16, 17, 21, 25, 29)$
(b) $F(A, B, C, D) = A'B'CE' + B'C'D'E' + A'B'D' + B'CD' + A'CD + A'BD$

3.12 Simplify the following Boolean functions to product-of-sums form:

(a) $F(w, x, y, z) = \Sigma(0, 1, 2, 5, 8, 10, 13)$
(b)* $F(A, B, C, D) = \Pi(1, 3, 5, 7, 13, 15)$
(c) $F(A, B, C, D) = \Pi(1, 3, 6, 9, 11, 12, 14)$

3.13 Simplify the following expressions to (1) sum-of-products and (2) products-of-sums:

(a)* $x'z' + y'z' + yz' + xy$
(b) $ACD' + C'D + AB' + ABCD$
(c) $(A + C' + D')(A' + B' + D')(A' + B + D')(A' + B + C')$
(d) $ABC' + AB'D + BCD$

3.14 Give three possible ways to express the following Boolean function with eight or fewer literals:

$$F = B'C'D' + AB'CD' + BC'D + A'BCD$$

3.15 Simplify the following Boolean function F, together with the don't-care conditions d, and then express the simplified function in sum-of-minterms form:

(a) $F(x, y, z) = \Sigma(2, 3, 4, 6, 7)$
$d(x, y, z) = \Sigma(0, 1, 5)$

(b)* $F(A, B, C, D) = \Sigma(0, 6, 8, 13, 14)$
$d(A, B, C, D) = \Sigma(2, 4, 10)$

(c) $F(A, B, C, D) = \Sigma(4, 5, 7, 12, 13, 14)$
$d(A, B, C, D) = \Sigma(1, 9, 11, 15)$

(d) $F(A, B, C, D) = \Sigma(1, 3, 8, 10, 15)$
$d(A, B, C, D) = \Sigma(0, 2, 9)$

3.16 Simplify the following functions, and implement them with two-level NAND gate circuits:

(a) $F(A, B, C, D) = A'B'C + AC' + ACD + ACD' + A'B'D'$
(b) $F(A, B, C, D) = AB + A'BC + A'B'C'D$
(c) $F(A, B, C) = (A' + B' + C')(A' + B')(A' + C')$
(d) $F(A, B, C, D) = A'B + A + C' + D'$

3.17* Draw a NAND logic diagram that implements the complement of the following function:

$$F(A, B, C, D) = \Sigma(0, 1, 2, 3, 4, 8, 9, 12)$$

3.18 Draw a logic diagram using only two-input NOR gates to implement the following function:

$$F(A, B, C, D) = (A \oplus B)'(C \oplus D)$$

3.19 Simplify the following functions, and implement them with two-level NOR gate circuits:
(a)* $F = wx' + y'z' + w'yz'$
(b) $F(w, x, y, z) = \Sigma(1, 2, 13, 14)$
(c) $F(x, y, z) = [(x + y)(x' + z)]'$

3.20 Draw the multi-level NOR and multi-level NAND circuits for the following expression:

$$(AB' + CD')E + BC(A + B)$$

3.21 Draw the multi-level NAND circuit for the following expression:

$$w(x + y + z) + xyz$$

3.22 Convert the logic diagram of the circuit shown in Fig. 4.4 into a multiple-level NAND circuit.

3.23 Implement the following Boolean function F, together with the don't-care conditions d, using no more than two NOR gates:

$$F(A, B, C, D) = \Sigma(2, 4, 6, 10, 12)$$
$$d(A, B, C, D) = \Sigma(0, 8, 9, 13)$$

Assume that both the normal and complement inputs are available.

3.24 Implement the following Boolean function F, using the two-level forms of logic (a) NAND-AND, (b) AND-NOR, (c) OR-NAND, and (d) NOR-OR:

$$F(A, B, C, D) = \Sigma(0, 4, 8, 9, 10, 11, 12, 14)$$

3.25 List the eight degenerate two-level forms and show that they reduce to a single operation. Explain how the degenerate two-level forms can be used to extend the number of inputs to a gate.

3.26 With the use of maps, find the simplest sum-of-products form of the function $F = fg$, where

$$f = abc' + c'd + a'cd' + b'cd'$$

and

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

3.27 Show that the dual of the exclusive-OR is also its complement.

3.28 Derive the circuits for a three-bit parity generator and four-bit parity checker using an odd parity bit.

3.29 Implement the following four Boolean expressions with three half adders

$$D = A \oplus B \oplus C$$
$$E = A'BC + AB'C$$
$$F = ABC' + (A' + B')C$$
$$G = ABC$$

3.30* Implement the following Boolean expression with exclusive-OR and AND gates:

$$F = AB'CD' + A'BCD' + AB'C'D + A'BC'D$$