1.7* Convert the hexadecimal number 68BE to binary, and then convert it from binary to octal.

1.8 Convert the decimal number 431 to binary in two ways: (a) Convert directly to binary; (b) convert first to hexadecimal and then from hexadecimal to binary. Which method is faster?

1.9 Express the following numbers in decimal:

(a)* $(10110.0101)_2$ (b)* $(16.5)_{16}$ (c)* $(26.24)_8$
(d) $(FAFA.B)_{16}$ (e) $(1010.1010)_2$ (f)* $(BABA)_{16}$
(g)* $(ABCD)_{16}$

1.10 Convert the following binary numbers to hexadecimal and to decimal: (a) 1.10010, (b) 110.010. Explain why the decimal answer in (b) is 4 times that in (a).

1.11 Perform the following division in binary: 111011 ÷ 101.

1.12* Add and multiply the following numbers without converting them to decimal.

(a) Binary numbers 1011 and 101.
(b) Hexadecimal numbers 2E and 34.

1.13 Do the following conversion problems:

(a) Convert decimal 27.315 to binary.
(b) Calculate the binary equivalent of 2/3 out to eight places. Then convert from binary to decimal. How close is the result to 2/3?
(c) Convert the binary result in (b) into hexadecimal. Then convert the result to decimal. Is the answer the same?

1.14 Obtain the 1's and 2's complements of the following binary numbers:

(a) 10000000 (b) 00000000 (c) 11011010
(d) 01110110 (e) 10000101 (f) 11111111
(g) 10101010 (h) 01010101

1.15 Find the 9's and the 10's complement of the following decimal numbers:

(a) 12,345,678 (b) 24,681,234 (c) 52,784,630
(d) 25,000,000 (e) 63,325,600 (f) 00,000,000.

1.16 (a) Find the 16's complement of BABA.
(b) Convert BABA to binary.
(c) Find the 2's complement of the result in (b).
(d) Convert the answer in (c) to hexadecimal and compare with the answer in (a).

1.17 Perform subtraction on the given unsigned numbers using the 10's complement of the subtrahend. Where the result should be negative, find its 10's complement and affix a minus sign. Verify your answers.

(a) 6,428 – 3,409 (b) 125 – 1,800
(c) 2,043 – 6,152 (d) 1,631 – 745

1.18 Perform subtraction on the given unsigned binary numbers using the 2's complement of the subtrahend. Where the result should be negative, find its 2's complement and affix a minus sign.

(a) 10011 – 10001 (b) 100010 – 100011
(c) 1001 – 101000 (d) 110000 – 10101

1.19* The following decimal numbers are shown in sign-magnitude form: +9,286 and +801. Convert them to signed-10's-complement form and perform the following operations (note that the sum is +10,627 and requires five digits and a sign).

(a) (+9,286) + (+ 801) (b) (+9,286) + (−801)
(c) (−9,286) + (+ 801) (d) (−9,286) + (−801)

1.20 Convert decimal +46 and +29 to binary, using the signed-2's-complement representation and enough digits to accommodate the numbers. Then perform the binary equivalent of (+29) + (−49), (−29) + (+49), and (−29) + (−49). Convert the answers back to decimal and verify that they are correct.

1.21 If the numbers $(+9{,}742)_{10}$ and $(+641)_{10}$ are in signed magnitude format, their sum is $(+10{,}383)_{10}$ and requires five digits and a sign. Convert the numbers to signed-10's-complement form and find the following sums:

(a) (+9,742) + (+ 641) (b) (+9,742) + (−641)
(c) (−9,742) + (+ 641) (d) (−9,742) + (−641)

1.22 (a) Convert decimal 8,723 to both BCD and ASCII codes. For ASCII, an even parity bit is to be appended at the left. (b) Repeat part (a) for $(1234)_{10}$.

1.23 Represent the unsigned decimal numbers 842 and 535 in BCD, and then show the steps necessary to form their sum.

1.24 Formulate a weighted binary code for the decimal digits, using weights

(a)* 6, 3, 1, 1
(b) 6, 4, 2, 1

1.25 Represent the decimal number 5,137 in (a) BCD, (b) excess-3 code, (c) 2421 code, and (d) a 6311 code.

1.26 Find the 9's complement of decimal 5,137 and express it in 2421 code. Show that the result is the 1's complement of the answer to (c) in Problem 1.25. This demonstrates that the 2421 code is self-complementing.

1.27 Assign a binary code in some orderly manner to the 52 playing cards. Use the minimum number of bits.

1.28 Write the expression "G. Boole" in ASCII, using an eight-bit code. Include the period and the space. Treat the leftmost bit of each character as a parity bit. Each eight-bit code should have even parity. (George Boole was a 19th century mathematician. Boolean algebra, introduced in the next chapter, bears his name.)

1.29* Decode the following ASCII code:
1000010 1101001 1101100 1101100 1000111 1100001 1110100 1100101 1110011.

1.30 The following is a string of ASCII characters whose bit patterns have been converted into hexadecimal for compactness: 73 F4 E5 76 E5 4A EF 62 73. Of the eight bits in each pair of digits, the leftmost is a parity bit. The remaining bits are the ASCII code.

(a) Convert the string to bit form and decode the ASCII.
(b) Determine the parity used: odd or even?

1.31* How many printing characters are there in ASCII? How many of them are special characters (not letters or numerals)?

1.32* What bit must be complemented to change an ASCII letter from capital to lowercase and vice versa?

1.33* The state of a 12-bit register is 100010010111. What is its content if it represents

(a) three decimal digits in BCD?
(b) three decimal digits in the excess-3 code?
(c) three decimal digits in the 84-2-1 code?
(d) a binary number?

1.34 List the ASCII code for the 10 decimal digits with an odd parity bit in the leftmost position.

1.35 By means of a timing diagram similar to Fig. 1.5, show the signals of the outputs f and g in Fig. P1.35 as functions of the three inputs a, b, and c. Use all eight possible combinations of a, b, and c.

1.36 By means of a timing diagram similar to Fig. 1.5, show the signals of the outputs f and g in Fig. P1.36 as functions of the two inputs a and b. Use all four possible combinations of a and b.

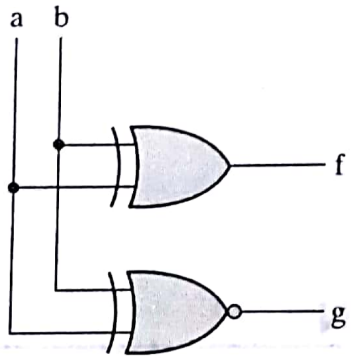


FIGURE P1.36

1.37 Draw timing waveforms of Fig. P1.37 considering all possible combinations of a and b. Compare the output with outputs f and g in Fig. 1.36.

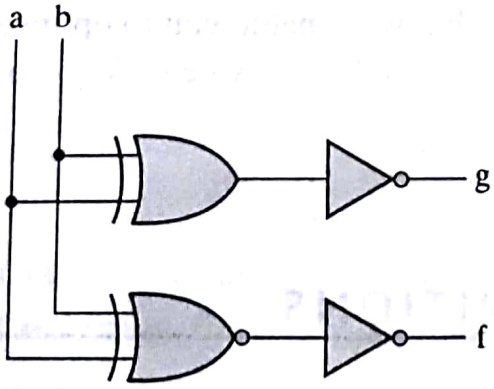


FIGURE P1.37

2.4 Reduce the following Boolean expressions to the indicated number of literals:

(a)* $A'C' + ABC + AC' + AB'$ to two literals

(b)* $(x'y' + z)' + z + xy + wz$ to three literals

(c)* $A'B(D' + CD) + B(A + A'CD)$ to one literal

(d)* $(A' + C)(A' + C')(A + B + CD)$ to four literals

(e) $ABCD + A'BD + ABC'D + A'D$ to four literals

2.5 Draw logic diagrams of the circuits that implement the original and simplified expressions in Problem 2.2.

2.6 Draw logic diagrams of the circuits that implement the original and simplified expressions in Problem 2.3.

2.7 Draw logic diagrams of the circuits that implement the original and simplified expressions in Problem 2.4.

2.8 Find the complement of $F = wx + yz$; then show that $FF' = 0$ and $F + F' = 1$.

2.9 Find the complement of the following expressions:

(a)* $xy' + x'y$

(b) $(A'B + CD)E' + E$

(c) $(x' + y + z')(x + y')(x + z)$

2.10 Given the Boolean functions F_1 and F_2, show that

(a) The Boolean function $E = F_1 + F_2$ contains the sum of the minterms of F_1 and F_2.

(b) The Boolean function $G = F_1F_2$ contains only the minterms that are common to F_1 and F_2.

2.16 The logical sum of all minterms of a Boolean function of n variables is 1.

(a) Prove the previous statement for $n = 3$.

(b) Suggest a procedure for a general proof.

2.17 Obtain the truth table of the following functions, and express each function in sum-of-minterms and product-of-maxterms form:

(a)* $(xy + z)(y + xz)$

(b) $(x + y')(y' + z)$

(c) $x'z + wx'y + wyz' + w'y'$

(d) $(xy + yz' + x'z)(x + z)$

2.18 For the Boolean function

$$F = xy'z + x'y'z + w'xy + wx'y + wxy$$

(a) Obtain the truth table of F.

(b) Draw the logic diagram, using the original Boolean expression.

(c)* Use Boolean algebra to simplify the function to a minimum number of literals.

2.20 Express the complement of the following functions in sum-of-minterms form:

(a) $F(A, B, C, D) = \Sigma(3, 5, 9, 11, 15)$ (b) $F(x, y, z) = \Pi(2, 4, 5, 7)$

2.21 Convert each of the following to the other canonical form:

(a) $F(x, y, z) = \Sigma(2, 4, 5, 6)$ (b) $F(A, B, C, D) = \Pi(0, 1, 2, 4, 7, 9, 12)$

2.22* Convert each of the following expressions into sum of products and product of sums:

(a) $(AB + C)(B + C'D)$ (b) $x' + x(x + y')(y + z')$

2.23 Draw the logic diagram corresponding to the following Boolean expressions without simplifying them:

(a) $BC' + AB + ACD$ (b) $(A + B)(C + D)(A' + B + D)$

(c) $(AB + A'B')(CD' + C'D)$ (d) $A + CD + (A + D')(C' + D)$

2.24 Show that the dual of the exclusive-OR is equal to its complement.

2.25 By substituting the Boolean expression equivalent of the binary operations as defined in Table 2.8, show the following:

(a) The inhibition operation is neither commutative nor associative.

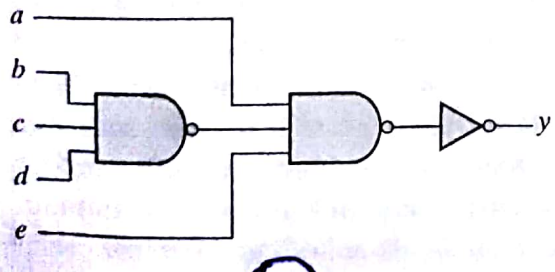
(b) The exclusive-OR operation is commutative and associative.

2.26 Show that a positive logic NAND gate is a negative logic NOR gate and vice versa.

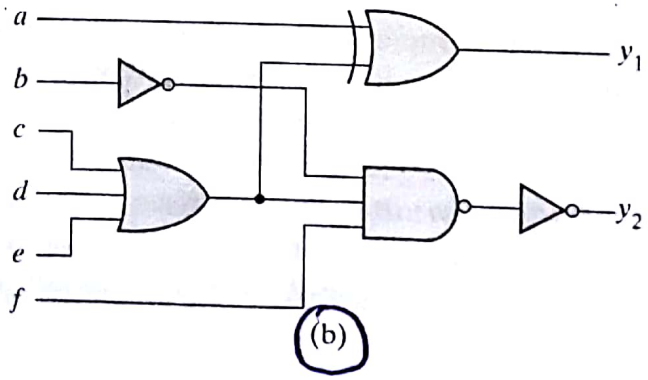
2.27 Write the Boolean equations and draw the logic diagram of the circuit whose outputs are defined by the following truth table:

f_1	f_2	a	b	c
1	0	0	0	0
0	0	0	0	1
0	1	0	1	0
1	1	0	1	1
0	1	1	0	0
0	1	1	0	1
1	1	1	1	0
1	0	1	1	1

2.28 Write Boolean expressions and construct the truth tables describing the outputs of the circuits described by the following logic diagrams:



(a)



(b)

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)* $F(A, B, C, D) = \Sigma(4, 5, 6, 7, 12, 13, 14)$
$d(A, B, C, D) = \Sigma(1, 9, 11, 15)$

(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) $F(A, B, C, D) = A'B'C + AC + ACD + ACD' + A'B'D' + B'CD$

(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:

$$F(A, B, C, D) = \Sigma(0, 1, 2, 3, 4, 8, 9, 10, 11, 12)$$

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) $F(w, x, y, z) = 2(1, 2, 13, 14)$

(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.

tion of the four inputs.

(b) List the truth table with 16 binary combinations of the four input variables. Then list the binary values for T_1 through T_4 and outputs F_1 and F_2 in the table.

(c) Plot the Boolean output functions obtained in part (b) on maps, and show that the simplified Boolean expressions are equivalent to the ones obtained in part (a).

4.2* Obtain the simplified Boolean expressions for outputs F and G in terms of the input variables in the circuit of Fig. P4.2.

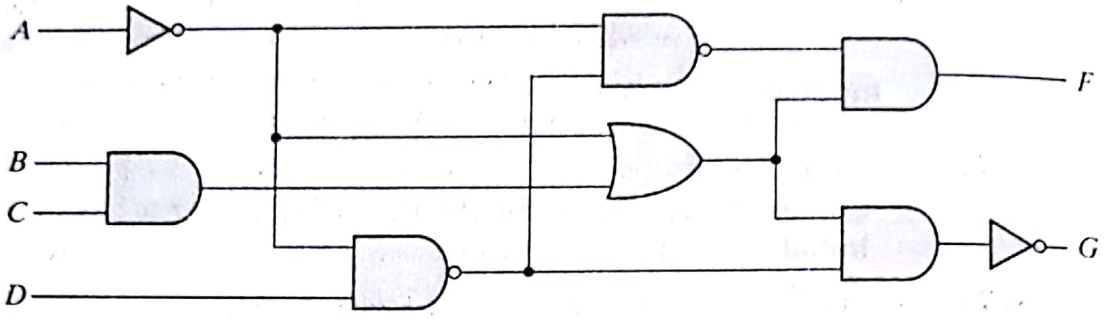


FIGURE P4.2

4.3 For the circuit shown in Fig. 4.33 (Section 4.11),

(a) Write the Boolean functions for the four outputs in terms of the input variables.

(b)* If the circuit is listed in a truth table, how many rows and columns would there be in the table? Justify your answer with proper reasoning.

4.4 Design a combinational circuit with three inputs and one output.

(a)* The output is 1 when the binary value of the inputs is less than or equal to 3. The output is 0 otherwise.

(b) The output is 1 when the binary value of the inputs is an odd number.

(c)* The output is 1 when the binary value of the inputs is an even number.

4.5 Design a combinational circuit with three inputs, x, y, and z, and three outputs, A, B, and C. When the binary input is 0, 1, 2, or 3, the binary output is two greater than the input. When the binary input is 4, 5, 6, or 7, the binary output is two less than the input.

4.6 A majority circuit is a combinational circuit whose output is equal to 1 if the input variables have more 1's than 0's. The output is 0 otherwise.

(a)* Design a three-input majority circuit by finding the circuits truth table, Boolean equation, and a logic diagram.

(b)* How many NAND gates (2 input) are needed to implement this function?

(c) Write and verify a Verilog dataflow model of the circuit.

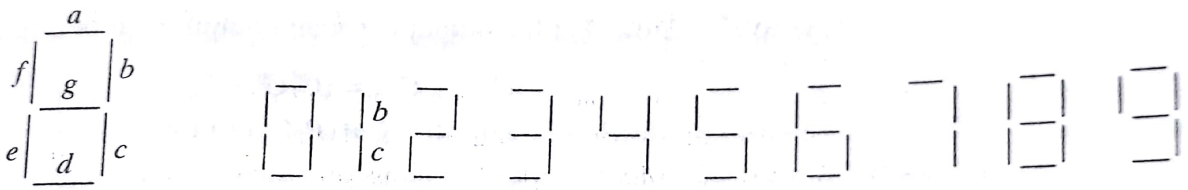
4.7 Design a combinational circuit that converts a four-bit Gray code (Table 1.6) to a four-bit binary number.

(a)* Implement the circuit with exclusive-OR gates.

(b) Using a case statement, write and verify a Verilog model of the circuit.

4.8* Design a code converter that converts a decimal digit from the 8, 4, –2, –1 code to BCD (see Table 1.5). (HDL— see Problem 4.52.)

4.9 An ABCD-to-seven-segment decoder is a combinational circuit that converts a decimal digit in BCD to an appropriate code for the selection of segments in an indicator used to display the



(a) Segment designation (b) Numerical designation for display

FIGURE P4.9

decimal digit in a familiar form. The seven outputs of the decoder (a, b, c, d, e, f, g) select the corresponding segments in the display, as shown in Fig. P4.9(a). The numeric display chosen to represent the decimal digit is shown in Fig. P4.9(b). Using a truth table and Karnaugh maps, design the BCD-to-seven-segment decoder, using a minimum number of gates. The six invalid combinations should result in a blank display. (HDL—see Problem 4.53.)

4.10* D

4.36 An 8 × 1 multiplexer has inputs A, B, and C connected to the selection inputs S_2, S_1, and S_0, respectively. The data inputs I_0 through I_7 are as follows:

(a)* $I_1 = I_2 = 0$; $I_3 = I_5 = I_7 = 1$; $I_0 = I_4 = D$; and $I_6 = D'$.

(b) $I_1 = I_2 = 0$; $I_3 = I_7 = 1$; $I_4 = I_5 = D$; and $I_0 = I_6 = D'$.

Determine the Boolean function that the multiplexer implements.

4.37 Implement the following Boolean function with a 4 × 1 multiplexer and external gates.

(a)* $F(A, B, C, D) = \Sigma(1, 3, 4, 11, 12, 13, 14, 15)$

(b) $F(A, B, C, D) = \Sigma(1, 2, 4, 7, 8, 9, 10, 11, 13, 15)$

Connect inputs A and B to the selection lines. The input requirements for the four data lines will be a function of variables C and D. These values are obtained by expressing F as a function of C and D for each of the four cases when $AB = 00$, 01, 10, and 11. The functions may have to be implemented with external gates and with connections to power and ground.

Common for both Batch I and Batch II

(1) Solve by tabulation method and verify using K Map.

$$\sum m\,(2, 4, 6, 8, 10, 12, 14) + \sum d\,(1, 3, 5, 7)$$

(2) Write short note on 7 segment display.