Chapter 13 Instructor Notes

Chapter 13 is a stand-alone chapter that does not require much more than a general introduction to the idea of analog and digital signals as a prerequisite. Thus, the chapter could be covered as early as desired. Some instructors may find it desirable to first introduce the basics of electronic switching circuits by covering the appropriate sections of Chapters 9-11.

The first section introduces the ideas of analog and digital signals, and the concepts of sampling and quantization, in an intuitive fashion. Section 13.2 introduces the binary number system, and binary codes; the box *Focus on Measurements: Digital Position Encoders* (pp. 622-623) discusses optical position encoders of the type commonly encountered in many industrial applications (e.g., robotics). The third section presents the foundations of Boolean algebra, and defines the properties of logic gates; the box *Focus on Measurements: Fail-safe Autopilot Logic* (p. 629) illustrates a simple application of digital logic to motivate the content of the section from a more practical perspective. Combinational logic design through the use of Karnaugh maps is presented in section 13.4; the boxes *Focus on Methodology: Sum-of-Products Realizations* (p. 639) and *Focus on Methodology: Products-of-Sums Realizations* (p. 644) summarize design procedures for sum-of-products and product-of-sums circuits. The box *Focus on Measurements: Safety Circuit for the Operation of a Stamping Press* (pp. 646-648) demonstrates of the usefulness of even the simplest logic circuits in an industrial setting. A brief survey of digital logic could stop here, if desired.

Section 13.5 describes more advanced combinational logic modules; The box *Focus on Measurements: EPROM-based Look-up Table for Automotive Fuel Injection System Control* (pp. 654-655) is centered around the air-to-fuel ratio control problem in an internal combustion engine, and illustrates a truly wide-spread application of digital logic, since this type of circuit is present in virtually every modern automobile.

The end-of-chapter problems are divided into four sections. The first contains a few simple exercises related to number systems; the second section on combinational logic offers a selection of simple problems that are extensions of the examples given in the text, and also includes 5 applied problems (13. 19-13.23) that demonstrate the use of Boolean logic in five every-day situations. The third section covers logic design, and also includes a couple of problems with an applied flavor (13.38, 13.53), in addition to a variety of more traditional design problems. Problems 13.48 - 13.52 are related to number codes. The fourth section contains a few problems related to combinational logic modules.

Learning Objectives

1. Understand the concepts of analog and digital signals and of quantization. Section 1.
2. Convert between decimal and binary number system and use the hexadecimal system and BCD and Gray codes. Section 2.
3. Write truth tables, realize logic functions from truth tables using logic gates. Section 3.
4. Systematically design logic functions using Karnaugh maps. Section 4.
5. Study various combinational logic modules, including multiplexers, memory and decoder elements, and programmable logic arrays. Section 5.

Section 13.2: The Binary Number System

Problem 13.1

Solution:

Known quantities:
The base 10 representation of five numbers: 401_{10}, 273_{10}, 15_{10}, 38_{10}, 56_{10}.

Find:
The hex and the binary representation for these numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 191_{16}, 110010001_2 b) 111_{16}, 100010001_2 c) F_{16}, 1111_2 d) 26_{16}, 100110_2 e) 38_{16}, 111000_2

Problem 13.2

Solution:

Known quantities:
The hex representation of five numbers: A_{16}, 66_{16}, 47_{16}, 21_{16}, 13_{16}.

Find:
The base 10 and the binary representation for these numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 10_{10}, 1010_2 b) 102_{10}, 1100110_2 c) 71_{10}, 1000111_2 d) 33_{10}, 100001_2 e) 19_{10}, 10011_2

Problem 13.3

Solution:

Known quantities:
The base 10 representation of four numbers: 271.25_{10}, 53.375_{10}, 37.32_{10}, 54.27_{10}.

Find:
The binary representation for these numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 100001111.01_2 b) 110101.011_2 c) 100101.01010_2 d) 110110.010001_2

Problem 13.4

Solution:

Known quantities:
The binary representation of six numbers: 1111_2, 1001101_2, 1100101_2, 1011100_2, 11101_2, 101000_2.

Find:
The hex and the base 10 representation for these numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) F_{16}, 15_{10} b) $4D_{16}$, 77_{10} c) 65_{16}, 101_{10} d) $5C_{16}$, 92_{10} e) $1D_{16}$, 29_{10} f) 28_{16}, 40_{10}

Problem 13.5

Solution:

Known quantities:
Three couples of binary numbers.

Find:
The addition for each couple.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 11111010 b) 100010100 c) 110000100

Problem 13.6

Solution:

Known quantities:
Three couples of binary numbers.

Find:
The subtraction for each couple.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 11100 b) 1101110 c) 1000

Problem 13.7

Solution:

Known quantities:
Three eight-bit binary numbers in sign-magnitude form.

Find:
The decimal value of these numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) -120 b) -31 c) 121

Problem 13.8

Solution:

Known quantities:
Three decimal numbers.

Find:
The sign-magnitude form binary representation.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 01111110 b) 11111110 c) 01101100 d) 11100010

Problem 13.9

Solution:

Known quantities:
Four binary numbers.

Find:
The two's complement of these four numbers.

Analysis:
Using the methodologies introduced in paragraph 13.2:

a) 2^4 - 1111 = 10000 -1111 = 0001
b) 2^7 - 1001101 = 0110011
c) 2^7 - 1011100 = 0100100
d) 2^5 - 11101= 100000 - 11101 = 00011

Section 13.3: Boolean Algebra

Problem 13.10

Solution:

Known quantities:

The expression $B = AB + \overline{A}B$.

Find:

The truth table that proves that the expression is true.

Analysis:

Using a truth table as explained in paragraph 13.3:

A	B	$\overline{A}$	AB	$\overline{A}B$	$AB + \overline{A}B$
0	0	1	0	0	0
0	1	1	0	1	1
1	0	0	0	0	0
1	1	0	1	0	1

we prove that the expression is true.

Problem 13.11

Solution:

Known quantities:

The expression $BC + B\overline{C} + \overline{B}A = A + B$.

Find:

The truth table that proves that the expression is true.

Analysis:

Using a truth table as explained in paragraph 13.3:

A	B	C	BC	$B\overline{C}$	$\overline{B}A$	$BC + B\overline{C} + \overline{B}A$	$A+B$
1	1	1	1	0	0	1	1
1	1	0	0	1	0	1	1
1	0	1	0	0	1	1	1
1	0	0	0	0	1	1	1
0	1	1	1	0	0	1	1
0	1	0	0	1	0	1	1
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0

we prove that the expression is true.

Problem 13.12

Solution:

Known quantities:

The expression $(X+Y)\cdot(\overline{X}+X\cdot Y)=Y$.

Find:

The proof that the expression is true using the perfect induction method.

Analysis:

Using a truth table as explained in paragraph 13.3:

X	Y	$X+Y$	$\overline{X}$	$X\cdot Y$	$\overline{X}+X\cdot Y$	$(X+Y)\cdot(\overline{X}+X\cdot Y)$
0	0	0	1	0	1	0
0	1	1	1	0	1	1
1	0	1	0	0	0	0
1	1	1	0	1	1	1

we prove that the expression is true.

Problem 13.13

Solution:

Known quantities:

The logic function $F(X,Y,Z)=\overline{X}\cdot\overline{Y}\cdot\overline{Z}+\overline{X}\cdot Y\cdot Z+X\cdot(\overline{Y+Z})$.

Find:

The simplification of the expression using the rules of Boolean algebra and De Morgan's theorems.

Analysis:

Applying De Morgan's theorems, $F(X,Y,Z)=\overline{X}\cdot\overline{Y}\cdot\overline{Z}+\overline{X}\cdot Y\cdot Z+X\cdot\overline{Y}\cdot\overline{Z}$

Applying the rules of Boolean algebra,

$F(X,Y,Z)=\overline{X}\cdot\overline{Y}\cdot\overline{Z}+X\cdot\overline{Y}\cdot\overline{Z}+\overline{X}\cdot Y\cdot Z=\overline{Y}\cdot\overline{Z}+\overline{X}\cdot Y\cdot Z$

Problem 13.14

Solution:

Known quantities:

The logic function $f(A,B,C,D)=A\cdot B\cdot C+\overline{A}\cdot C\cdot D+\overline{B}\cdot C\cdot D$.

Find:

The simplification of the expression using the rules of Boolean algebra and De Morgan's theorems.

Analysis:

$$f(A,B,C,D)=ABC+\overline{A}CD+\overline{B}CD$$
$$=ABC+CD(\overline{A}+\overline{B})\leftarrow Distributive$$

Problem 13.15

Solution:

Known quantities:

The logic function $F(A,B,C) = \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C + A \cdot B \cdot \overline{C} + A \cdot B \cdot C$.

Find:

The simplification of the expression using the rules of Boolean algebra.

Analysis:

Applying the rules of Boolean algebra,

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

Problem 13.16

Solution:

Known quantities:

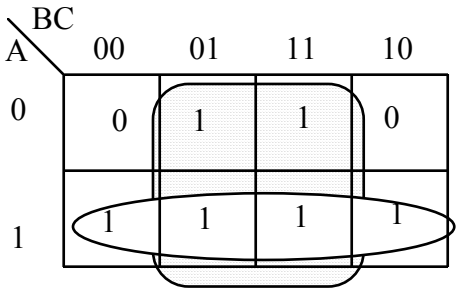
The truth table in Figure P13.16 for a logic function.

Find:

The expression for the logic function.

Analysis:

We first find the Karnaugh map for the function.



From the map, we can see that: $F = A + C$

Problem 13.17

Solution:

Known quantities:

The circuit of Figure P13.17.

Find:

The Boolean function describing the operation of the circuit.

Analysis:

$$F = \overline{AB} \cdot \overline{CD} \cdot \overline{E} = \overline{AB + CD + E}$$

where the second expression is a result of applying DeMorgan's theorem to the first.

Problem 13.18

Solution:

Known quantities:

The circuit of Figure P13.18.

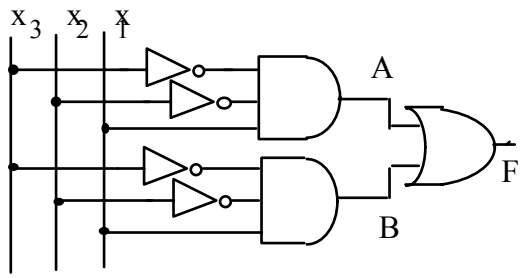
Find:

The truth table of the circuit.

Analysis:

The truth table can be obtained from the circuit considering the steps A, B and F, as reported in the figure of the circuit.

x_3	x_2	x_1	A	B	F
0	0	0	0	0	0
0	0	1	1	1	1
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	0	0



Problem 13.19

Solution:

Known quantities:
The rules that have to be followed in order to implement a strategy able to decide when the steal sign has to be given, for a baseball team.
The steal sign has to be given if:

a) there are no other runners, the pitcher is right-handed and the runner is fast, or
b) there is no one other runner on third-base, and one of the runner is fast, or
c) there is one other runner on second-base, the pitcher is left-handed, and both runners are fast.
d) Under no circumstances should the steal sign be given if all three bases have runners.

Find:
The circuit that implements these rules.

Analysis:
The table below lists the 8 possible conditions under which the steal sign should be given, using the following logic notations:
A runner on a base is 1, no runner is 0
A fast runner on a base is 1, a non-fast runner is 0
A right-handed pitcher is 1, a left-handed pitcher is 0

Base			**Runner**			**Pitcher**	**Output**
b_1	b_2	b_3	f_1	f_2	f_3	P	**Light On, y**
1	0	0	1	0	0	1	1
1	0	1	1	0	0	0	1
1	0	1	0	1	0	0	1
1	0	1	1	1	1	0	1
1	0	1	1	0	1	1	1
1	0	1	0	1	1	1	1
1	0	1	1	1	0	1	1
1	1	0	1	0	0	0	1

For all other conditions, the output is off.

Next, convert the truth table to a logical expression for the output:

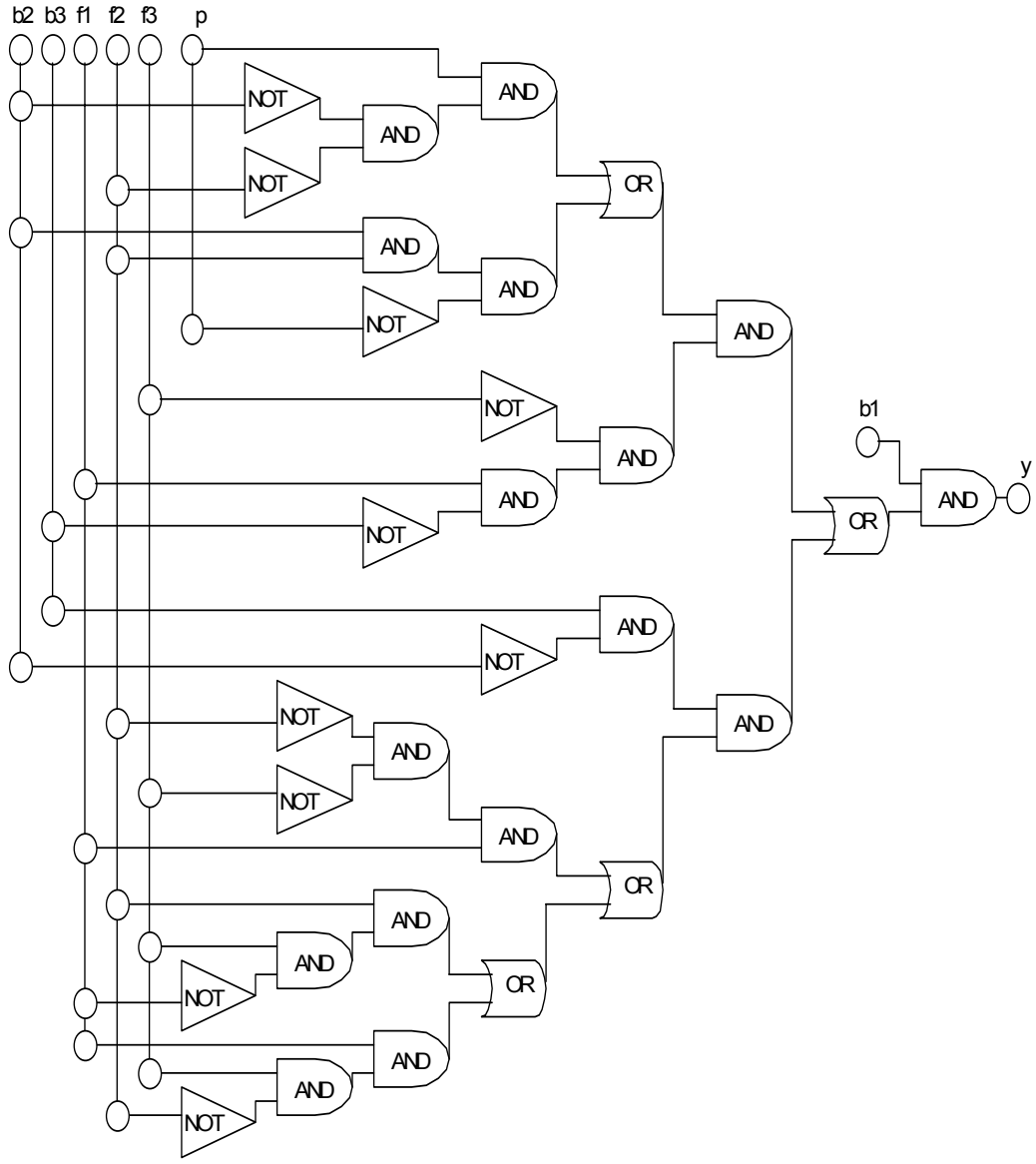
$$y = b_1 \cdot \bar{b}_2 \cdot \bar{b}_3 \cdot f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 \cdot P + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 \cdot \bar{P} + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot \bar{f}_1 \cdot \bar{f}_2 \cdot f_3 \cdot \bar{P}$$
$$b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot f_3 \cdot \bar{P} + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 \cdot P + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot \bar{f}_1 \cdot \bar{f}_2 \cdot f_3 \cdot P$$
$$+ b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot f_3 \cdot P + b_1 \cdot b_2 \cdot \bar{b}_3 \cdot f_1 \cdot f_2 \cdot \bar{f}_3 \cdot \bar{P}$$

$$y = b_1 \cdot \bar{b}_2 \cdot \bar{b}_3 \cdot f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 \cdot P + b_1 \cdot b_2 \cdot \bar{b}_3 \cdot f_1 \cdot f_2 \cdot \bar{f}_3 \cdot \bar{P} +$$
$$\left(b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot \bar{f}_1 \cdot \bar{f}_2 \cdot f_3 + b_1 \cdot \bar{b}_2 \cdot b_3 \cdot f_1 \cdot \bar{f}_2 \cdot f_3\right)\left(\bar{P} + P\right)$$

$$y = \left(b_1 \cdot \bar{b}_3 \cdot f_1 \cdot \bar{f}_3\right)\left(\bar{b}_2 \cdot \bar{f}_2 \cdot P + b_2 \cdot f_2 \cdot \bar{P}\right) + \left(b_1 \cdot \bar{b}_2 \cdot b_3\right)\left(f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 + \bar{f}_1 \cdot \bar{f}_2 \cdot f_3 + f_1 \cdot \bar{f}_2 \cdot f_3\right)$$

$$y = b_1\left[\left(\bar{b}_3 \cdot f_1 \cdot \bar{f}_3\right)\left(\bar{b}_2 \cdot \bar{f}_2 \cdot P + b_2 \cdot f_2 \cdot \bar{P}\right) + \left(\bar{b}_2 \cdot b_3\right)\left(f_1 \cdot \bar{f}_2 \cdot \bar{f}_3 + \bar{f}_1 \cdot \bar{f}_2 \cdot f_3 + f_1 \cdot \bar{f}_2 \cdot f_3\right)\right]$$

From this final expression, a logic circuit diagram can be made to achieve the correct response:



Note: The circles in the figure are nodes, not inversions.

Problem 13.20

Solution:

Known quantities:
The rules to pass a law in a small county.

Find:
The logic circuit that takes three votes as inputs and lights either a green or red light to indicate whether or not a measure passed.

Assumptions:

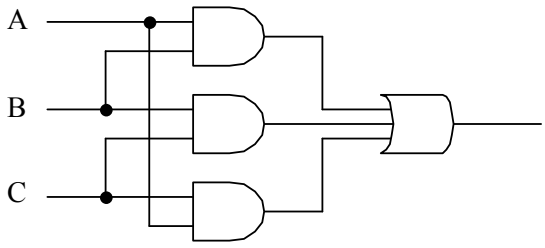
Green is enlightened if the law passes.

Analysis:

The function that implements the rules is:
$f(A,B,C) = AB + BC + AC$

If $f(A,B,C)$ is equal to 1 then the green light is lighted, otherwise the red one.

The circuit that implement this function is reported in the figure besides.



Problem 13.21

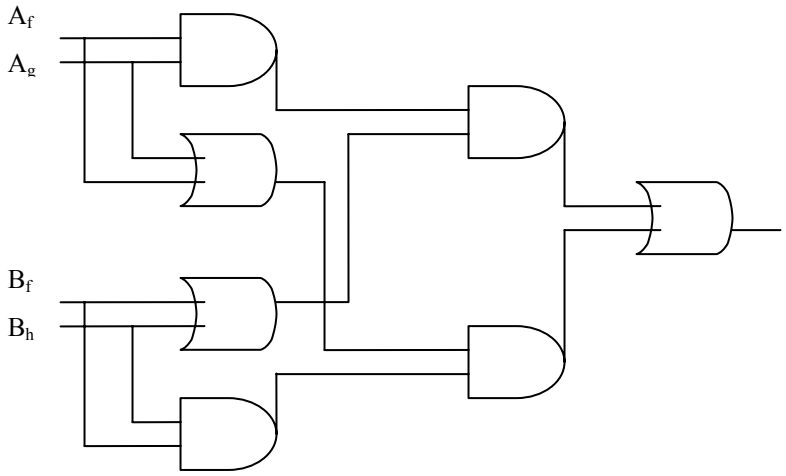
Solution:

Known quantities:
The set-up for a water purification plant.

Find:
The logic circuit that sounds an alarm if a dangerous situation occurs.

Analysis:
The four variables are the flow and the height of water in the tank A and in the tank B:
A_f, A_h, B_f, B_h.
From the Karnaugh map reported below we can find the minimum expression:
$A_f A_h (B_f + B_h) + B_f B_h (A_f + A_h)$
and the following realization of the circuit.

A_fA_h \ B_fB_h	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	0	1	0	1
10	0	0	1	0



Problem 13.22

Solution:

Known quantities:

The rules for an alert system designed for cars.

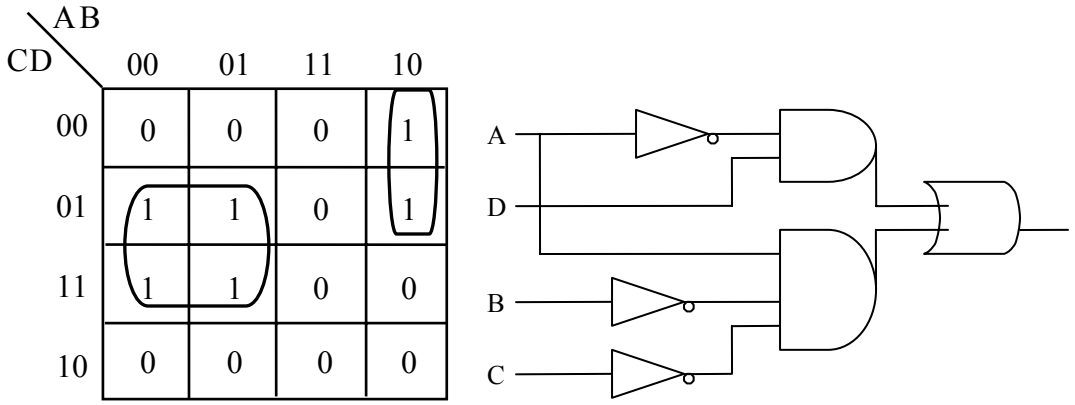
Find:

The logic circuit that implements these functions.

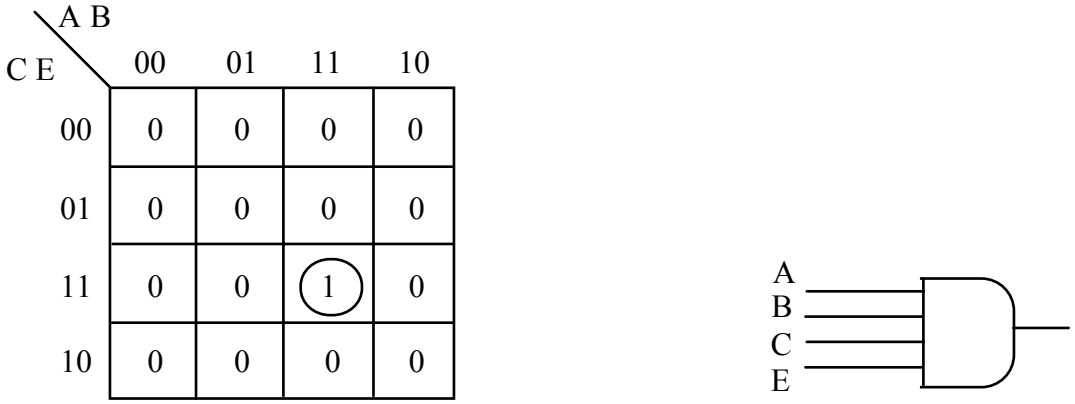
Analysis:

Let's design two different circuits for the buzzer and the starting conditions.
For both circuits we find first the Karnaugh map and then the minimum expression and the circuit to implement it.
A = Ignition key (1 if turned); B = Door (1 if closed); C = Seat belt (1 if fasten); D = Lights (1 if on);
E = Park (1 if on);
For the first circuit:

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



Start = ABCE



Problem 13.23

Solution:

Known quantities:

The on/off strategy for the compressor motor of a large commercial air conditioning unit.

Find:

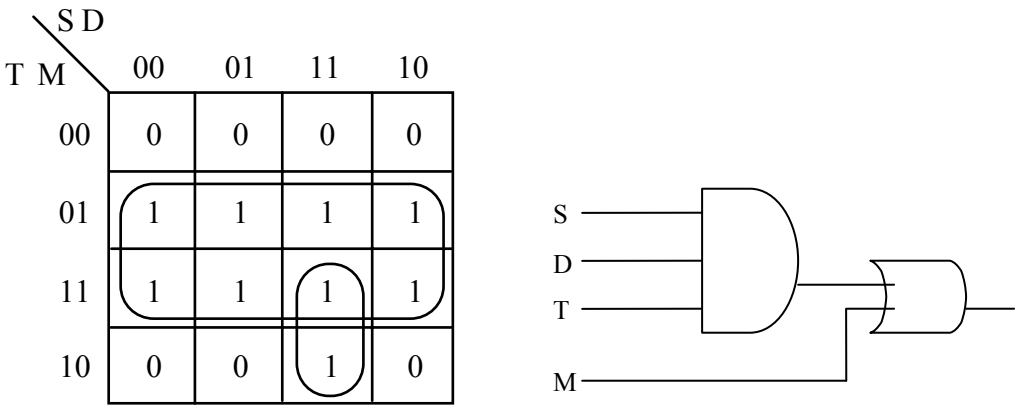
The logic diagram that incorporates the state of four devices (S, D, T and M) and produces the correct on/off condition for the motor startup.

Analysis:

From the Karnaugh map we determine the minimum expression:

$$SDT + M$$

that is implemented by the following logic diagram.


S D
T M
00
01
11
10
00 0 0 0 0
01 1 1 1 1
11 1 1 1 1
10 0 0 1 0
S
D
T
M

Section 13.4: Karnaugh Maps and Logic Design

Focus on Methodology: Sum-of-Products Realizations

1. Begin with isolated cells. These must be used as they are, since no simplification is possible.
2. Find all cells that are adjacent to only one cell, forming two-cell subcubes.
3. Find cells that form four-cell subcubes, eight-cell subcubes, and so forth.
4. The minimal expression is formed by the collection of the *smallest number of maximal subcubes*.

Focus on Methodology: Products-of Sums Realizations

1. Solve for the 0s exactly as for the 1s in the sum-of-products expressions.
2. Complement the resulting expression.

Problem 13.24

Solution:

Known quantities:

The truth table of Figure P13.24.

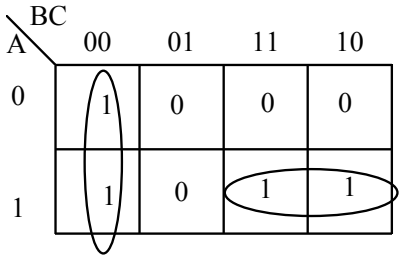
Find:

The logic function corresponding to the truth table.

Analysis:

From the map, we can see that:

$$F = A \cdot B + \overline{B} \cdot \overline{C}$$



Problem 13.25

Solution:

Known quantities:

The circuit shown if Figure P13.25.

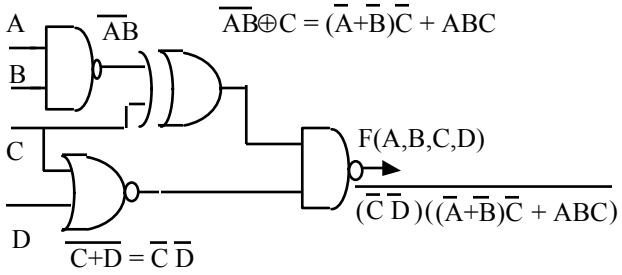
Find:

The minimum expression for the output.

Analysis:

The logic circuit gives the following expression:

$$F = \left(\overline{C+D}\right)\left(\overline{A \cdot B} \oplus C\right) = \left(\overline{C} \cdot \overline{D}\right)\left(\left(\overline{A \cdot B} + C\right) \cdot \left(\overline{\overline{A \cdot B} \cdot C}\right)\right) = \left(\overline{C} \cdot \overline{D}\right)\left(\left(\overline{A \cdot B} + C\right) \cdot \left(A \cdot B \cdot \overline{C}\right)\right) =$$
$$= \left(\overline{C} \cdot \overline{D}\right)\left(A \cdot B \cdot C + \overline{A \cdot B} \cdot \overline{C}\right) = \left(\overline{C} \cdot \overline{D}\right)\left(A \cdot B \cdot C + \left(\overline{A} + \overline{B}\right) \cdot \overline{C}\right)$$



Problem 13.26

Solution:

Known quantities:

The logic function $f(A,B,C) = A \cdot B \cdot C + A \cdot B \cdot \overline{C} + A \cdot \overline{B \cdot C}$.

Find:

The minimum expression for the function.

Analysis:

Using a Karnaugh map (reported besides) we determine:

$$f = AB + A\overline{C}$$

BC \ A	0	1
00	0	1
01	0	0
11	0	1
10	0	1

Problem 13.27

Solution:

Known quantities:

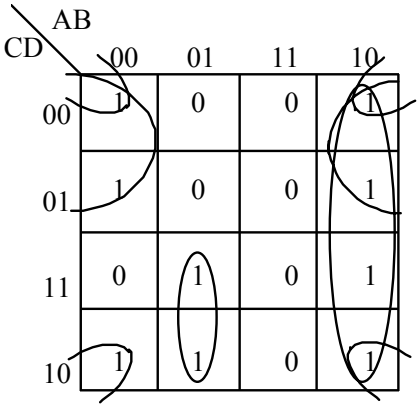
The truth table of Figure P13.27.

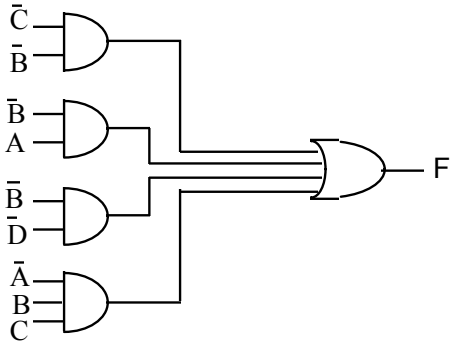
Find:

a) The Karnaugh map for the logic function defined by the truth table.
b) The minimum expression for the function.
c) The realization of the function using AND, OR and NOT gates.

Analysis:

a) The Karnaugh map is shown below.





b) The map leads to the expression:

$$f = \overline{B} \cdot \overline{C} + A \cdot \overline{B} + \overline{A} \cdot B \cdot C + \overline{B} \cdot \overline{D}$$

c) and to the gate realization shown above.

Problem 13.28

Solution:

Known quantities:

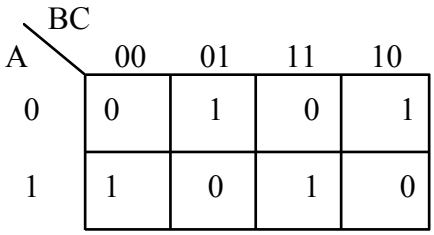
The truth table of Figure P13.28.

Find:

The Karnaugh map and the minimum expression for the function defined by the truth table.

Analysis:

The Karnaugh map is shown besides.



The minimum expression is reported below.

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

Problem 13.29

Solution:

Known quantities:

The rules that define a logic function.

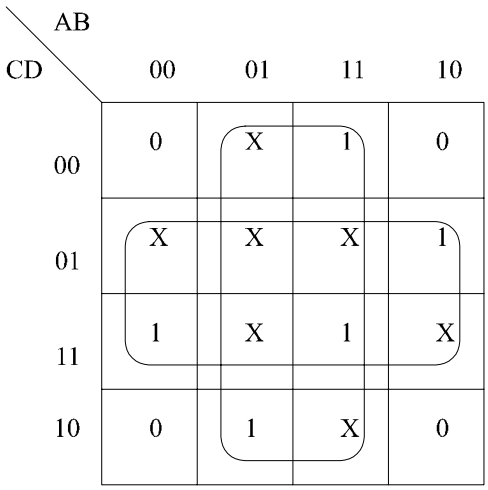
Find:

The minimum expression for this function and a sketch of a circuit to implement this function using only AND and NOT gates.

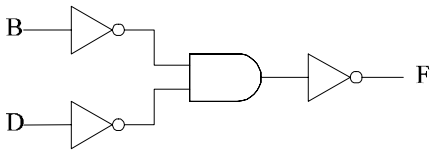
Analysis:

The truth table for the function is reported below as well as the Karnaugh map the minimum expression for the function and its realization using AND and NOT gates.

A	B	C	D	F
0	0	0	0	0
0	0	0	1	X
0	0	1	0	0
0	0	1	1	1
0	1	0	0	X
0	1	0	1	X
0	1	1	0	1
0	1	1	1	X
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	X
1	1	0	0	1
1	1	0	1	X
1	1	1	0	X
1	1	1	1	1



$$F = B + D$$



Problem 13.30

Solution:

Known quantities:

The truth table that defines a logic function shown in Figure P13.30.

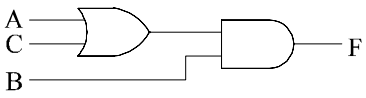
Find:

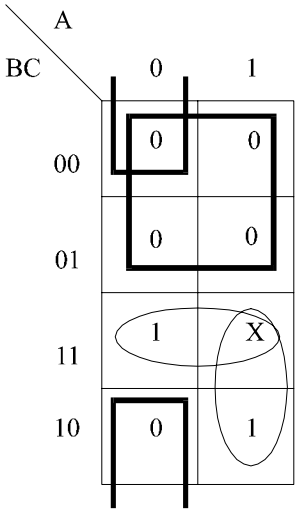
The construction of a logic circuit describing the function using only two gates.

Analysis:

SOP: $F = AB + BC$ ⇨ 3 gates

POS: $\overline{F} = \overline{B} + \overline{A} \cdot \overline{C} \Rightarrow F = B(A + C)$ ⇨ 2 gates





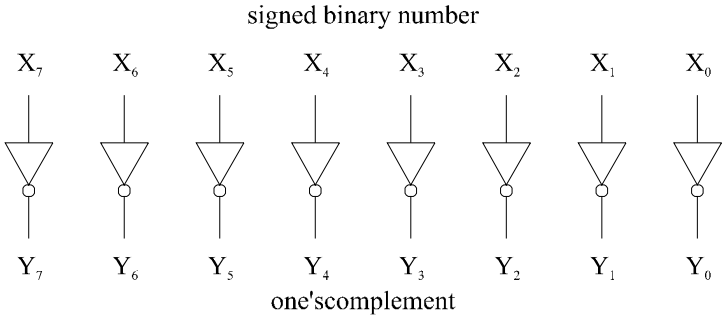
Problem 13.31

Solution:

Known quantities:
The requirement for the circuit to be designed: to produce the one's complement of an 8-bit signed binary number.

Find:
The design of the logic circuit.

Analysis:



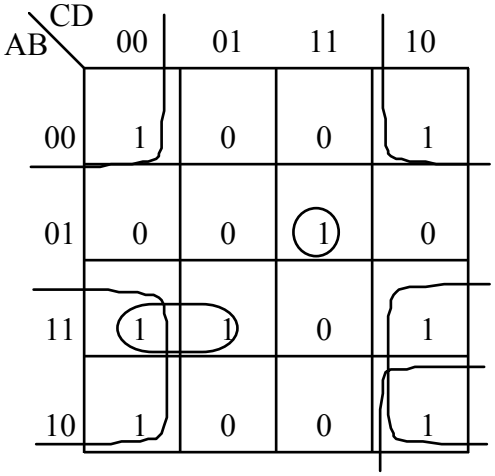
Problem 13.32

Solution:

Known quantities:
The truth table of Figure P13.32.

Find:
The Karnaugh map and the minimum expression for the logic function.

Analysis:
The Karnaugh map and the expression for the function are reported below.



$$F = \overline{B} \cdot \overline{D} + A \cdot \overline{D} + A \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C \cdot D$$

Problem 13.33

Solution:

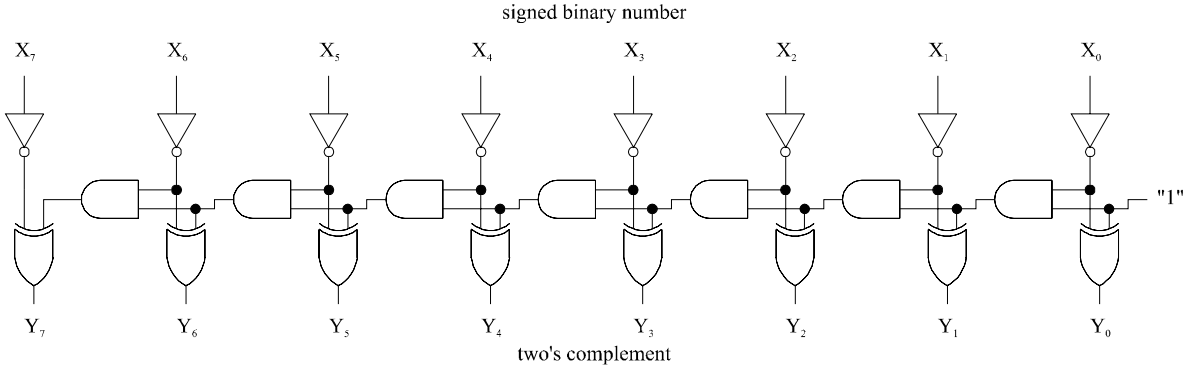
Known quantities:
The requirement for the circuit to be designed: to produce the two's complement of an 8-bit signed binary number.

Find:
The design of the logic circuit.

Analysis:
The two’s complement is the one’s complement plus one.

X	C_{IN}	SUM	C_{OUT}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$SUM = X \oplus C_{IN}$$
$$C_{OUT} = X \bullet C_{IN}$$



Problem 13.34

Solution:

Known quantities:
The circuit of Figure P13.34.

Find:
The minimum output expression for the circuit.

Analysis:

$$f = \overline{A \cdot \overline{B} + B + \overline{C}} = \overline{A \cdot \overline{B}} \cdot \overline{B + \overline{C}} = A \cdot B \cdot (B + C) = A \cdot B(1 + C) = A \cdot B$$

Problem 13.35

Solution:

Known quantities:
The combinational logic to be designed: the addition between two 4-bit binary numbers.

Find:
The circuit to implement this operation.

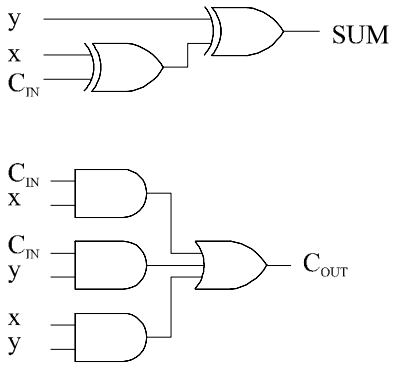
Analysis:

A one-bit adder truth table is as follows, and from this table we find the following expression, and the two circuits for those functions:

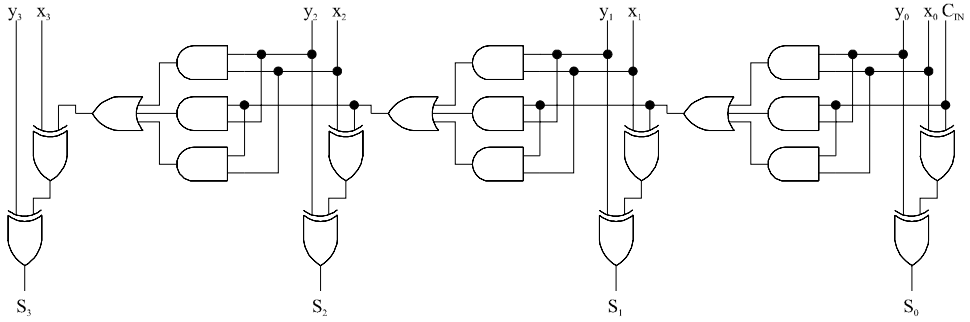
C_{IN}	x	y	SUM	C_{OUT}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$SUM = C_{IN} \oplus x \oplus y$$

$$C_{OUT} = C_{IN} + xy$$



The complete 4-bit adder can be constructed as shown below:



Note that this circuit assumes a carry-in for the lsb. If this is not necessary, then the circuit can be reduced correspondingly.

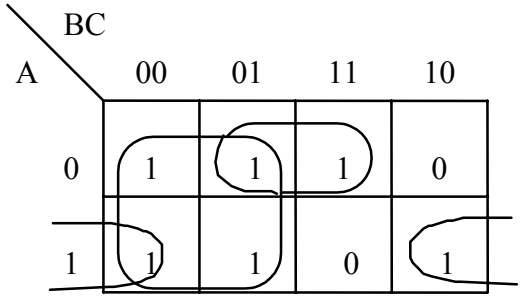
Problem 13.36

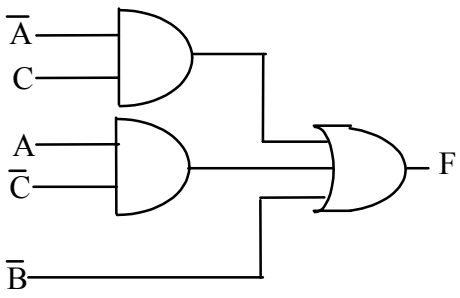
Solution:

Known quantities:
The truth table of Figure P13.36.

Find:
The minimum output expression and the circuit for the function.

Analysis:
Reported below are the Karnaugh map, the expression for the function and circuit realization.





$$F = \overline{B} + A \cdot \overline{C} + \overline{A} \cdot C$$

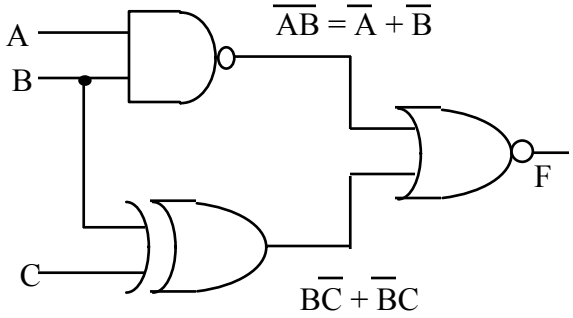
Problem 13.37

Solution:

Known quantities:
The logic circuit of Figure P13.37.

Find:
The minimum output expression for the circuit.

Analysis:



f(A,B,C) = ABC

Problem 13.38

Solution:

Known quantities:
The rules for blood donation.

Find:
The circuit which will approve or disapprove any particular transfusion between donator and receiver.

Analysis:

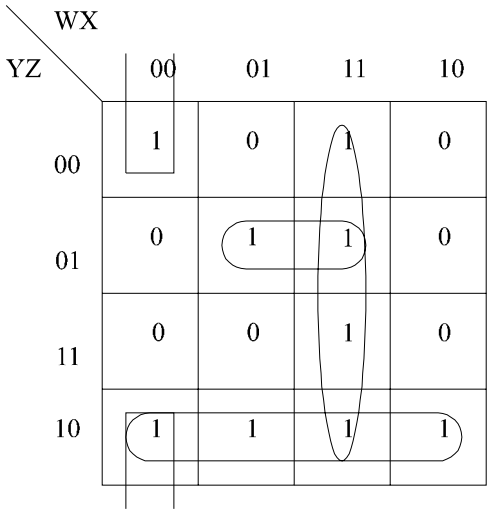
Let WX represent a 2-bit code for the donor blood type, and let yz represent a 2-bit code for the recipient blood type. Then WXYZ will represent a donor-recipient pair. Let F be true if a transfusion can be made. Blood type codes may be assigned as follows:

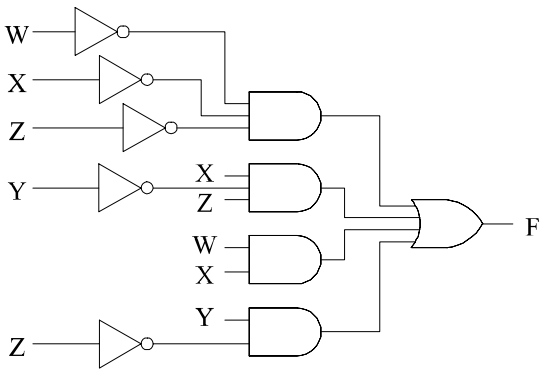
	WX	YZ
A	00	00
B	01	01
AB	10	10
O	11	11

The truth table is:

Donor–Recipient	WX	YZ	F
A-A	00	00	1
A-B	00	01	0
A-AB	00	10	1
A-O	00	11	0
B-A	01	00	0
B-B	01	01	1
B-AB	01	10	1
B-O	01	11	0
AB-A	10	00	0
AB-B	10	01	0
AB-AB	10	10	1
AB-O	10	11	0
O-A	11	00	1
O-B	11	01	1
O-AB	11	10	1
O-O	11	11	1

And the Karnaugh map, then, is as follows:





From the Karnaugh map,

$$F = \overline{W} \cdot \overline{X} \cdot \overline{Z} + X \cdot \overline{Y} \cdot Z + WX + Y\overline{Z}$$

and the resulting circuit is shown above.

Problem 13.39

Solution:

Known quantities:

The logic circuit of Figure P13.39.

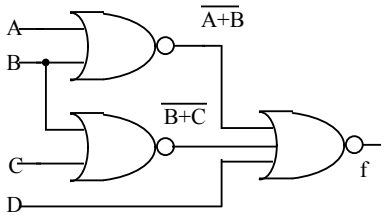
Find:

The minimum expression for the logic function.

Analysis:

The minimum expression is:

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



Problem 13.40

Solution:

Known quantities:

The combinational logic that have to be followed.

Find:

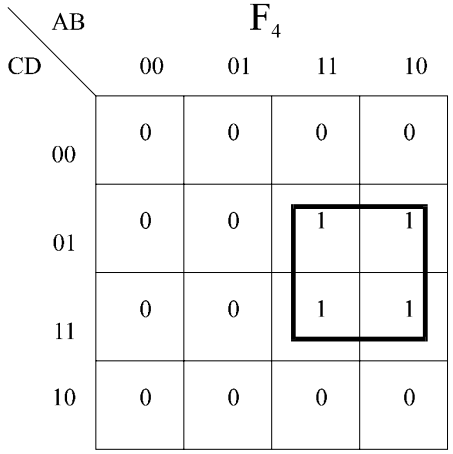
The circuit that implements this logic.

Analysis:

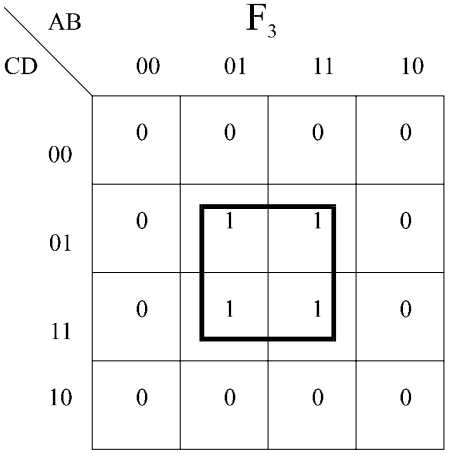
The appropriate truth table can be constructed as follows:

A B C D	F4	F3	F2	F1	F0
0 0 0 0	0	0	0	0	0
0 0 0 1	0	0	0	1	0
0 0 1 0	0	0	0	0	1
0 0 1 1	0	0	1	1	0
0 1 0 0	0	0	0	1	0
0 1 0 1	0	1	0	1	0
0 1 1 0	0	0	0	1	1
0 1 1 1	0	1	1	1	0
1 0 0 0	0	0	1	0	0
1 0 0 1	1	0	0	1	0
1 0 1 0	0	0	1	0	1
1 0 1 1	1	0	1	1	0
1 1 0 0	0	0	1	1	0
1 1 0 1	1	1	0	1	0
1 1 1 0	0	0	1	1	1
1 1 1 1	1	1	1	1	0

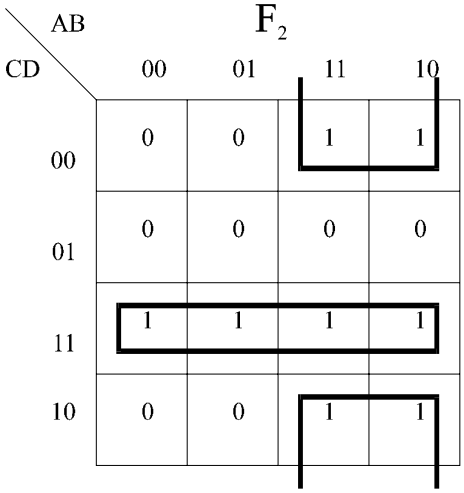
Next, we construct a Karnaugh map for each bit of the output, and determine its corresponding function.



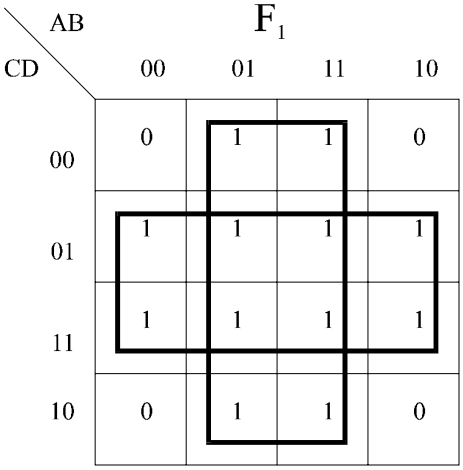
$F_4 = AD$



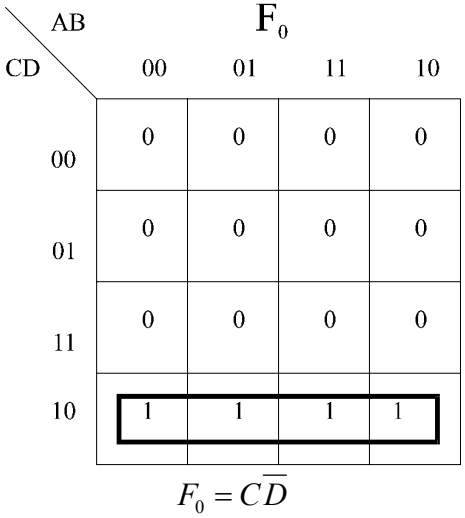
$F_3 = BC$



$F_2 = AB + A\overline{D}$

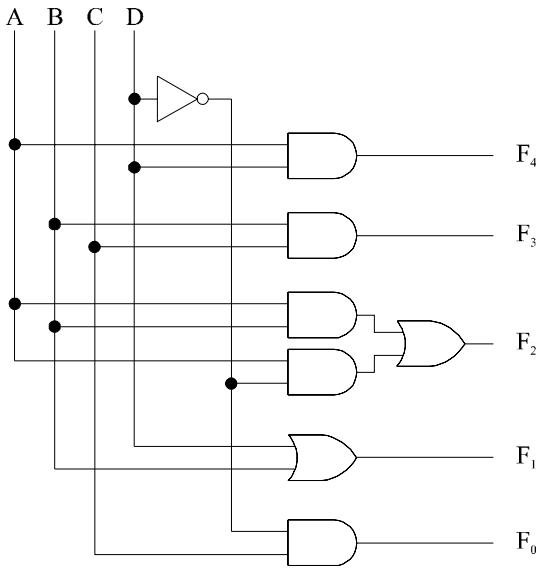


$F_1 = B + D$



$F_0 = C\overline{D}$

This completes the design.



Problem 13.41

Solution:

Known quantities:

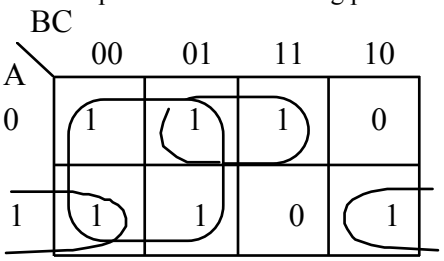
The truth table of Figure P13.41.

Find:

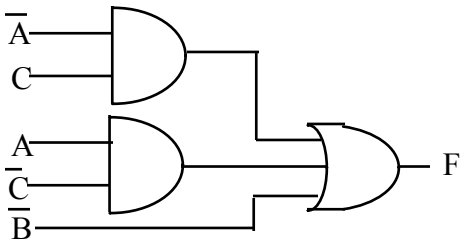
a) The Karnaugh map for the function defined in the truth table.
b) The minimum expression for the function.
c) The circuit using AND, OR, NOT gates.

Analysis:

The answers are reported in the following plots.



$$f = \bar{B} + \bar{A}C + A\bar{C}$$



Problem 13.42

Solution:

Known quantities:

The truth table of Figure P13.42.

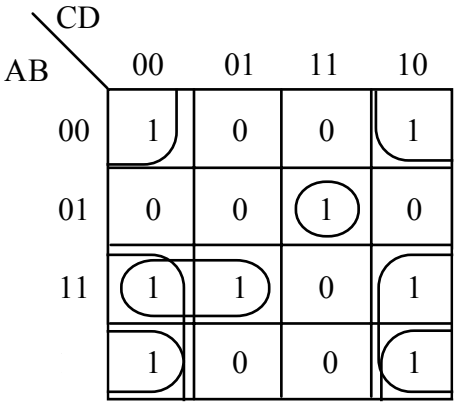
Find:

a) The Karnaugh map for the function defined in the truth table.
b) The minimum expression for the function.

Analysis:

The answers are reported in the following plots.

a)



b)

$$F = \overline{B} \cdot \overline{D} + A \cdot \overline{D} + A \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C \cdot D$$

Problem 13.43

Solution:

Known quantities:

The truth table of Figure P13.43.

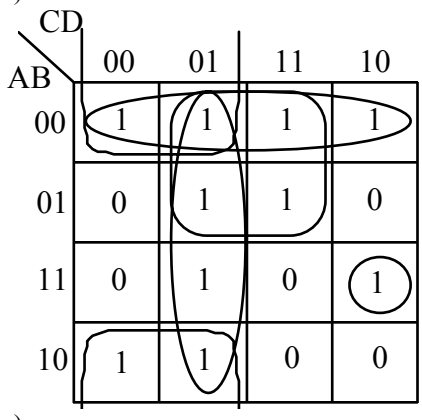
Find:

a) The Karnaugh map for the function defined in the truth table.
b) The minimum expression for the function.
c) The circuit implementation using only NAND gates.

Analysis:

The answers are reported in the following plots.

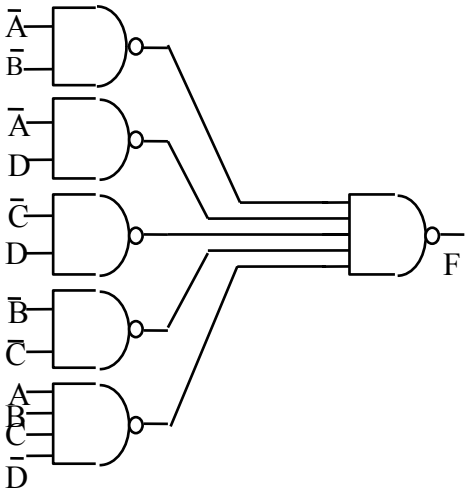
a)



b)

$$F = \overline{A} \cdot \overline{B} + \overline{A} \cdot D + \overline{C} \cdot D + \overline{B} \cdot \overline{C} + A \cdot B \cdot C \cdot \overline{D}$$

c)



Problem 13.44

Solution:

Known quantities:

The logic that defines a function.

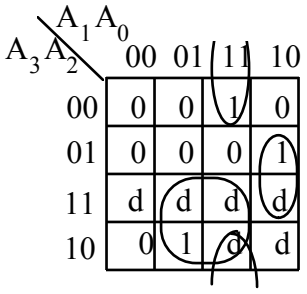
Find:

a) The Karnaugh map for the function and the truth table.
b) The minimum expression for the function.
c) The circuit using only AND, OR, and NOT gates.

Analysis:

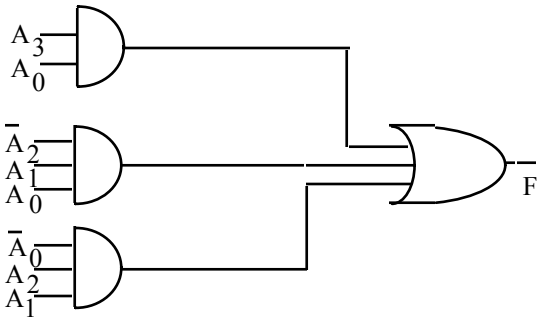
a)

A3	A2	A1	A0	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	d
1	0	1	1	d
1	1	0	0	d
1	1	0	1	d
1	1	1	0	d
1	1	1	1	d



b) From K-Map groupings, $F = A_3 \cdot A_0 + A_2 \cdot A_1 \cdot \overline{A_0} + A_1 \cdot A_0 \cdot \overline{A_2}$

c) The circuit for this function is:



Problem 13.45

Solution:

Known quantities:

The Karnaugh map reported in Figure P13.45.

Find:

The simplified sum-of-products representation of the function.

Analysis:

AB / CD	00	01	11	10
00	0	1	0	0
01	1	1	0	0
11	0	x	1	0
10	0	0	1	0

$$F = \overline{A} \cdot \overline{C} \cdot D + \overline{A} \cdot B \cdot \overline{C} + A \cdot B \cdot C$$

Problem 13.46

Solution:

Known quantities:

The semplification for the circuit of Problem 13.40 if it is known that the input represents a BCD (binary-coded decimal) number.

Find:

The simplified circuit or the reason for which it is not possible to simplify the circuit.

Analysis:

For this problem, the truth table is as follows:

A B C D	F_4	F_3	F_2	F_1	F_0
0 0 0 0	0	0	0	0	0
0 0 0 1	0	0	0	1	0
0 0 1 0	0	0	0	0	1
0 0 1 1	0	0	1	1	0
0 1 0 0	0	0	0	1	0
0 1 0 1	0	1	0	1	0
0 1 1 0	0	0	0	1	1
0 1 1 1	0	1	1	1	0
1 0 0 0	0	0	1	0	0
1 0 0 1	1	0	0	1	0
1 0 1 0	X	X	X	X	X
1 0 1 1	X	X	X	X	X
1 1 0 0	X	X	X	X	X
1 1 0 1	X	X	X	X	X
1 1 1 0	X	X	X	X	X
1 1 1 1	X	X	X	X	X

Now, we construct the Karnaugh maps and determine the corresponding functions.

F_4

CD \ AB	00	01	11	10
00	0	0	X	0
01	0	0	X	1
11	0	0	X	X
10	0	0	X	X

$F_4 = AD$

F_3

CD \ AB	00	01	11	10
00	0	0	X	0
01	0	1	X	0
11	0	1	X	X
10	0	0	X	X

$F_3 = BC$

F_2

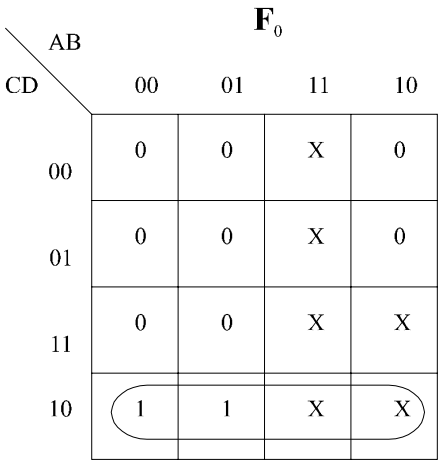
CD \ AB	00	01	11	10
00	0	0	X	1
01	0	0	X	0
11	1	1	X	X
10	0	0	X	X

$F_2 = AB + A\overline{D}$

F_1

CD \ AB	00	01	11	10
00	0	1	X	0
01	1	1	X	1
11	1	1	X	X
10	0	1	X	X

$F_1 = B + D$



$$F_0 = C\overline{D}$$

These expressions are identical to those obtained in Problem 12.36. Surprisingly, the presence of the don't cares did not change (or simplify) the solution.

Problem 13.47

Solution:

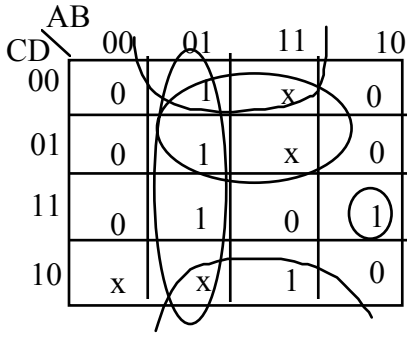
Known quantities:

The Karnaugh map shown in Figure P13.47.

Find:

The simplified sum-of-product representation of the function.

Analysis:



$$F = A \cdot \overline{B} \cdot C \cdot D + B \cdot \overline{D} + \overline{A} \cdot B$$

Problem 13.48

Solution:

Known quantities:

The parity bit method to ensure reliability in data transmission systems.

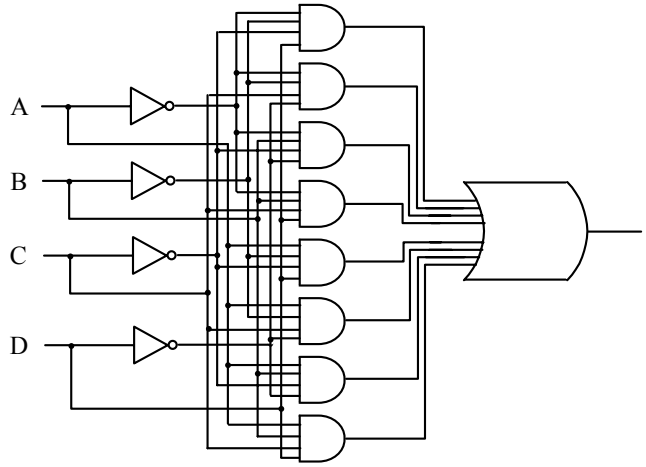
Find:

The logic circuit that checks the nibble of data and transmits the proper parity bit for even-parity systems.

Analysis:

If we write the nibble in the form ABCD, the Karnaugh map for this problem is:

AB \ CD	00	01	11	10
00	0	1	0	1
01	1	0	1	0
11	0	1	0	1
10	1	0	1	0



Hence the function that gives the parity bit can be written as:

$$F = A\overline{B}\,\overline{C}\,\overline{D} + \overline{A}B\overline{C}\,\overline{D} + \overline{A}\,\overline{B}C\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}BCD + A\overline{B}CD + AB\overline{C}D + ABC\overline{D}$$

The logic circuit is shown above.

Problem 13.49

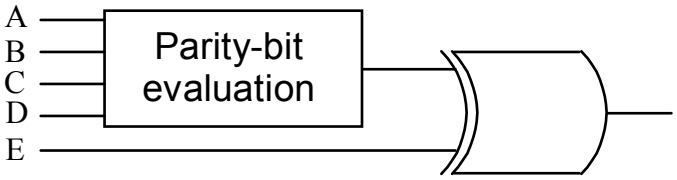
Solution:

Find:

The logic circuit that check a nibble of data and its parity-bit.

Analysis:

If we write the nibble in the form ABCD and E is the parity-bit, the logic circuit is:



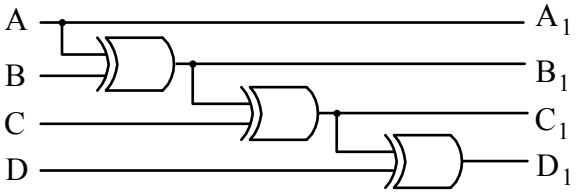
Problem 13.50

Solution:

Find:

A logic circuit that takes a 4-bit gray code input into a 4-bit nibble of BCD code.

Analysis:



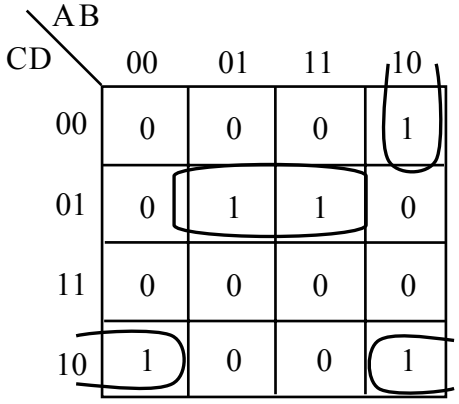
Problem 13.51

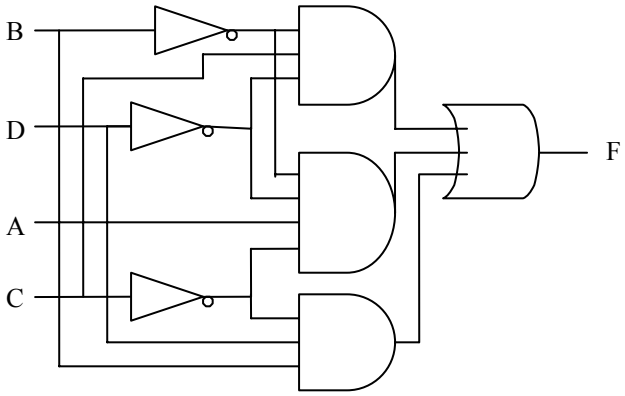
Solution:

Find:

A logic circuit that takes a 4-bit gray-code

Analysis:





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

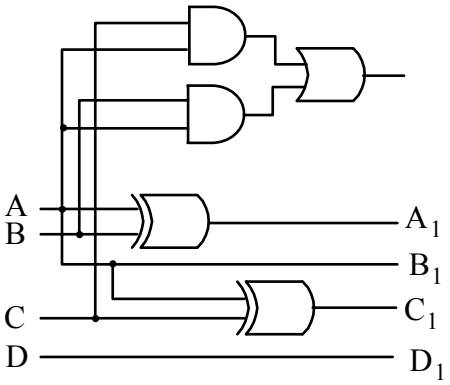
Problem 13.52

Solution:

Find:

A logic circuit that takes a BCD nibble as input and converts it into its 4221 equivalent, and reports an error if the BCD value exceeds 1001.

Analysis:



Problem 13.53

Solution:

Find:

A logic circuit that reports an error if the 4-bit outputs of two sensors differ by more than one part per 30 seconds period.

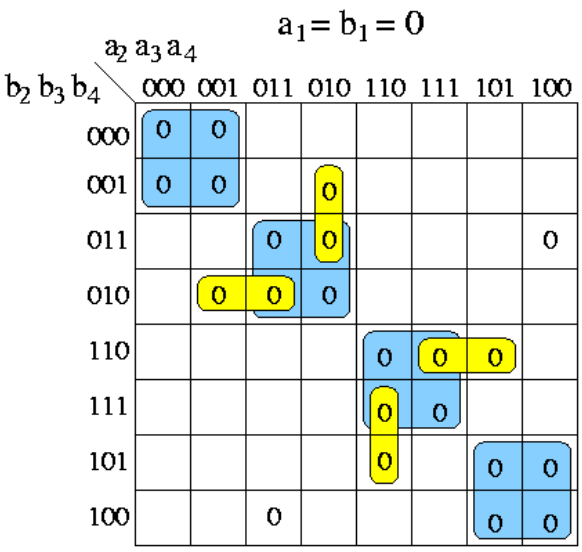
Analysis:

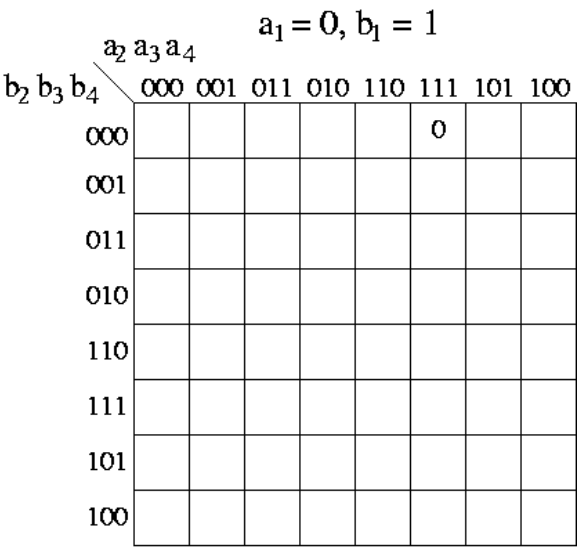
The output of the logic circuit will be one when a difference greater or equal to 2 is detected between the two sensor outputs.

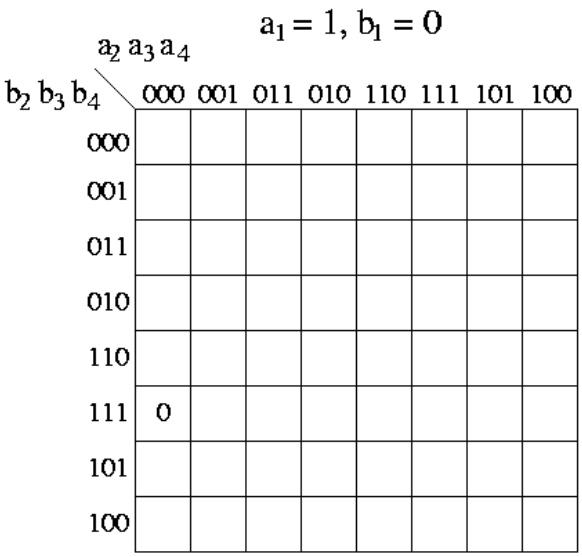
By defining with a_1, a_2, a_3, a_4 the four bits of the first sensor, and with b_1, b_2, b_3, b_4 the bits of the second sensor, the following Karnaugh maps can be derived.

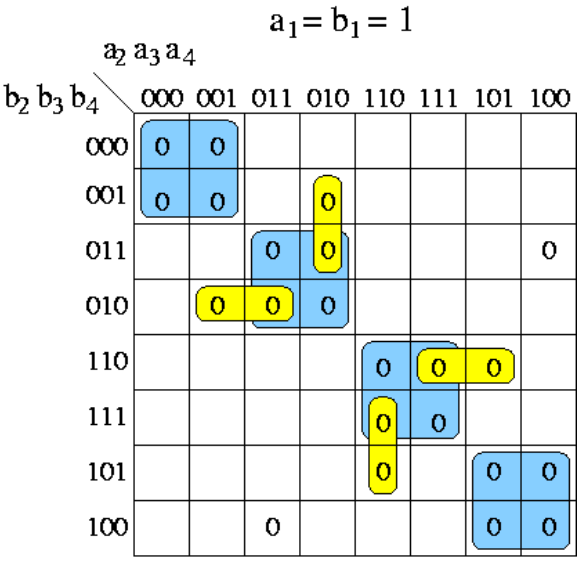
From the maps, considering the zeros, the following expression can be obtained

$$y = (a_1 + a_2 + a_3 + b_1 + b_2 + b_3)(a_1 + a_2 + \overline{a}_3 + b_1 + b_2 + \overline{b}_3)(a_1 + \overline{a}_2 + \overline{a}_3 + b_1 + \overline{b}_2 + \overline{b}_3)$$
$$(a_1 + \overline{a}_2 + a_3 + b_1 + \overline{b}_2 + b_3)(\overline{a}_1 + \overline{a}_2 + a_3 + \overline{b}_1 + \overline{b}_2 + b_3)(\overline{a}_1 + \overline{a}_2 + \overline{a}_3 + \overline{b}_1 + \overline{b}_2 + \overline{b}_3)$$
$$(\overline{a}_1 + a_2 + \overline{a}_3 + \overline{b}_1 + b_2 + \overline{b}_3)(\overline{a}_1 + a_2 + a_3 + \overline{b}_1 + b_2 + b_3)(a_1 + a_2 + \overline{a}_3 + a_4 + b_1 + b_2 + \overline{b}_4)$$
$$(a_1 + a_2 + \overline{a}_4 + b_1 + b_2 + \overline{b}_3 + b_4)(a_1 + \overline{a}_2 + \overline{a}_4 + b_1 + \overline{b}_2 + \overline{b}_3 + b_4)(a_1 + \overline{a}_2 + \overline{a}_3 + a_4 + b_1 + \overline{b}_2 + \overline{b}_4)$$
$$(\overline{a}_1 + a_2 + \overline{a}_3 + a_4 + \overline{b}_1 + b_2 + \overline{b}_4)(\overline{a}_1 + a_2 + \overline{a}_4 + \overline{b}_1 + b_2 + \overline{b}_3 + b_4)(\overline{a}_1 + \overline{a}_2 + \overline{a}_4 + \overline{b}_1 + \overline{b}_2 + \overline{b}_3 + b_4)$$
$$(\overline{a}_1 + \overline{a}_2 + \overline{a}_3 + a_4 + \overline{b}_1 + \overline{b}_2 + \overline{b}_4)(a_1 + \overline{a}_2 + a_3 + a_4 + b_1 + b_2 + \overline{b}_3 + \overline{b}_4)(a_1 + a_2 + \overline{a}_3 + \overline{a}_4 + b_1 + \overline{b}_2 + b_3 + b_4)$$
$$(\overline{a}_1 + \overline{a}_2 + a_3 + a_4 + \overline{b}_1 + b_2 + \overline{b}_3 + \overline{b}_4)(\overline{a}_1 + a_2 + \overline{a}_3 + \overline{a}_4 + \overline{b}_1 + \overline{b}_2 + b_3 + b_4)$$
$$(a_1 + \overline{a}_2 + \overline{a}_3 + \overline{a}_4 + \overline{b}_1 + b_2 + b_3 + b_4)(\overline{a}_1 + a_2 + a_3 + a_4 + b_1 + \overline{b}_2 + \overline{b}_3 + \overline{b}_4)$$









The expression can be simplified as follows

$$y = (a_1 \oplus b_1 + a_2 + a_3 + b_2 + b_3)(a_1 \oplus b_1 + a_2 + \overline{a}_3 + b_2 + \overline{b}_3)(a_1 \oplus b_1 + \overline{a}_2 + \overline{a}_3 + \overline{b}_2 + \overline{b}_3)$$
$$(a_1 \oplus b_1 + \overline{a}_2 + a_3 + \overline{b}_2 + b_3)(a_1 \oplus b_1 + a_2 + \overline{a}_3 + a_4 + b_2 + \overline{b}_4)(a_1 \oplus b_1 + a_2 + \overline{a}_4 + b_2 + \overline{b}_3 + b_4)$$
$$(a_1 \oplus b_1 + \overline{a}_2 + \overline{a}_4 + \overline{b}_2 + \overline{b}_3 + b_4)(a_1 \oplus b_1 + \overline{a}_2 + \overline{a}_3 + a_4 + \overline{b}_2 + \overline{b}_4)(a_1 \oplus b_1 + \overline{a}_2 + a_3 + a_4 + b_2 + \overline{b}_3 + \overline{b}_4)$$
$$(a_1 \oplus b_1 + a_2 + \overline{a}_3 + \overline{a}_4 + \overline{b}_2 + b_3 + b_4)(a_1 + \overline{a}_2 + \overline{a}_3 + \overline{a}_4 + \overline{b}_1 + b_2 + b_3 + b_4)(\overline{a}_1 + a_2 + a_3 + a_4 + b_1 + \overline{b}_2 + \overline{b}_3 + \overline{b}_4)$$

Using De Morgan's Theorem

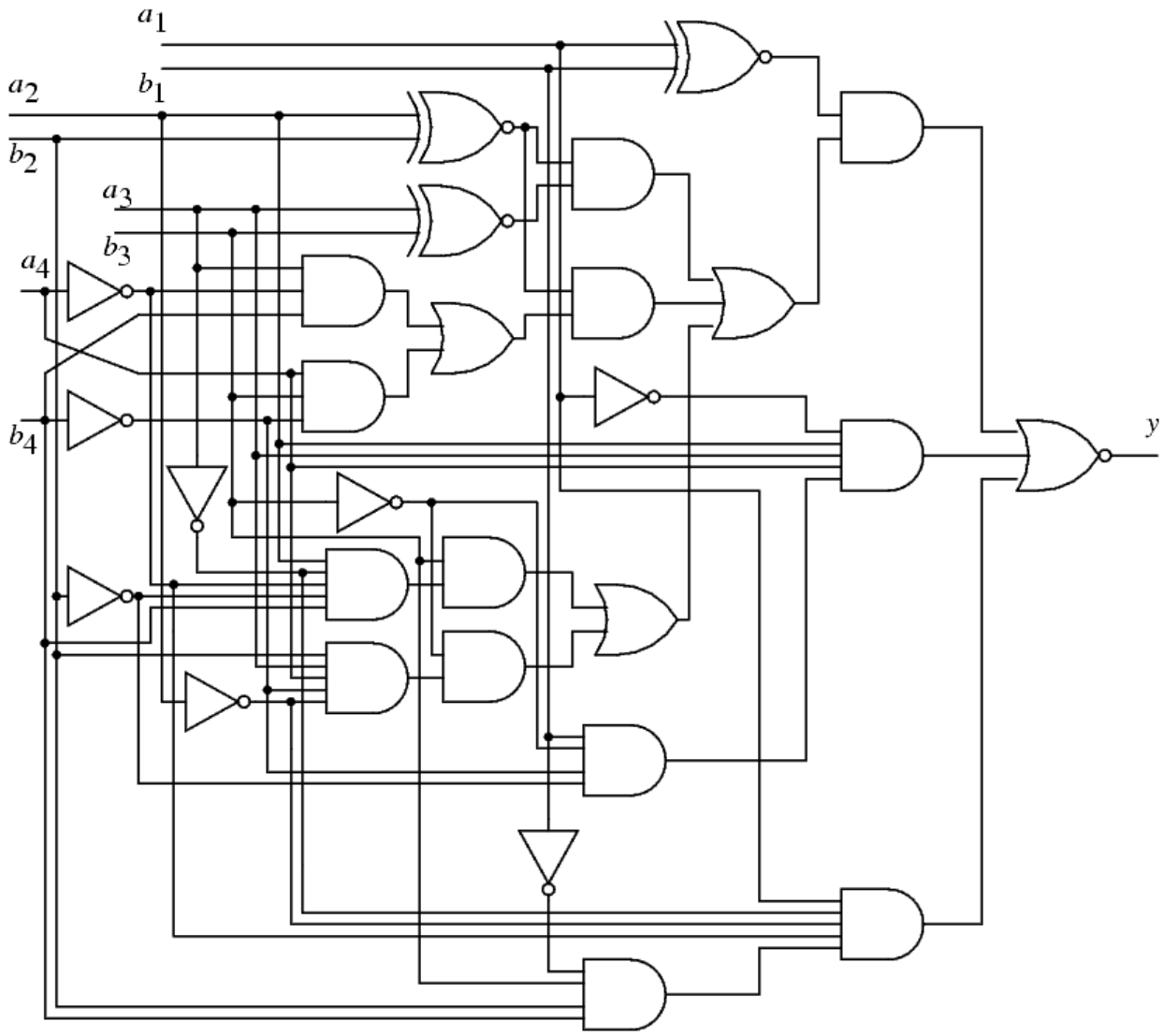
$$\overline{y = (a_1 \otimes b_1)(\overline{a}_2\overline{a}_3\overline{b}_2\overline{b}_3 + \overline{a}_2 a_3\overline{b}_2 b_3 + a_2 a_3 b_2 b_3 + a_2\overline{a}_3 b_2\overline{b}_3 + \overline{a}_2 a_3\overline{a}_4\overline{b}_2 b_4 + \overline{a}_2 a_4\overline{b}_2 b_3\overline{b}_4 + a_2 a_4 b_2 b_3\overline{b}_4 +}$$

$$\overline{+ a_2 a_3 \overline{a}_4 b_2 b_4 + a_2 \overline{a}_3 \overline{a}_4 \overline{b}_2 b_3 b_4 + \overline{a}_2 a_3 a_4 b_2 \overline{b}_3 \overline{b}_4 \big) + \overline{a}_1 a_2 a_3 a_4 b_1 \overline{b}_2 \overline{b}_3 \overline{b}_4 + a_1 \overline{a}_2 \overline{a}_3 \overline{a}_4 \overline{b}_1 b_2 b_3 b_4}$$

$$= \overline{(a_1 \otimes b_1)\big((a_2 \otimes b_2)(a_3 \otimes b_3) + (a_2 \otimes b_2)(a_3 \overline{a}_4 b_4 + a_4 b_3 \overline{b}_4\big) + a_2 \overline{a}_3 \overline{a}_4 \overline{b}_2 b_3 b_4 + \overline{a}_2 a_3 a_4 b_2 \overline{b}_3 \overline{b}_4 \big) +}$$

$$\overline{+ \overline{a}_1 a_2 a_3 a_4 b_1 \overline{b}_2 \overline{b}_3 \overline{b}_4 + a_1 \overline{a}_2 \overline{a}_3 \overline{a}_4 \overline{b}_1 b_2 b_3 b_4}$$

From the expression above, it is possible to derive the following logic circuit



Section 13.5: Combinational Logic Modules

Problem 13.54

Solution:

Known quantities:

The truth table for a function.

Find:

a) The Karnaugh map for the function;
b) Its minimum expression
c) Realize the function using a 1-of-8 multiplexer

Analysis:

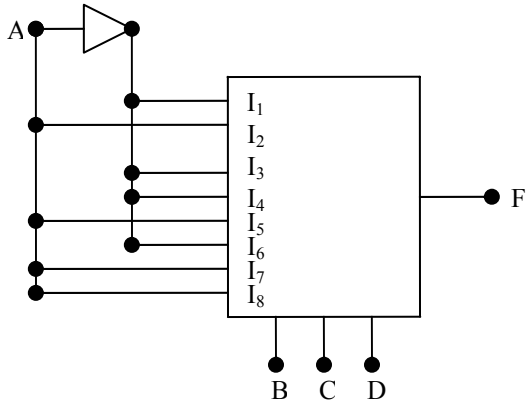
a)

CD \ AB	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	1	0	1	0
10	1	0	1	0

b)

$$F = \overline{A}\,\overline{B}\,\overline{D} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C}D + AB\overline{D} + ABC + A\overline{B}\,\overline{C}D = \left(\overline{A}\,\overline{B} + AB\right)\left(C + \overline{D}\right) + \overline{C}D\left(\overline{A}B + A\overline{B}\right)$$

c)



Problem 13.55

Solution:

Known quantities:

The multiplexer circuit shown in Figure P13.55.

Find:

a) The truth table for the multiplexer
b) The binary function performed by the multiplexer.

Analysis:

a)

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

b) Binary Addition - S is the sum, and C is the carry.

Problem 13.56

Solution:

Known quantities:

A circuit that can operate as a 4-to-16 decoder is shown in Figure P13.56.

Find:

The operation of the 4-to-16 decoder, and the role of the logic variable A.

Analysis:

Assuming that the enable input (EN) is active high, when EN is logic 0 (A is logic 1), all decoder outputs of the first decoder are
forced to logic 1 independent of the inputs. However, when EN is logic 1 (A is logic 0), all decoder outputs of the second decoder are forced to logic 1 independent of the select inputs. Therefore, A functions as the fourth bit of the select inputs. Thus, the circuit operates as a 4 of 16 decoder.

Problem 13.57

Solution:

Known quantities:

The circuit of Figure P13.57.

Find:

The ability of the circuit of performing a conversion from 4-bit binary numbers to 4-bit Gray code.

Analysis:

We construct the truth table for this circuit as shown below:

Binary Input $B_3B_2B_1B_0$	G_3	G_2 $B_2 \oplus B_3$	G_1 $B_1 \oplus B_2$	G_0 $B_0 \oplus B_1$
0 0 0 0	0	0	0	0
0 0 0 1	0	0	0	1
0 0 1 0	0	0	1	1
0 0 1 1	0	0	1	0
0 1 0 0	0	1	1	0
0 1 0 1	0	1	1	1
0 1 1 0	0	1	0	1
0 1 1 1	0	1	0	0
1 0 0 0	1	1	0	0
1 0 0 1	1	1	0	1
1 0 1 0	1	1	1	1
1 0 1 1	1	1	1	0
1 1 0 0	1	0	1	0
1 1 0 1	1	0	1	1
1 1 1 0	1	0	0	1
1 1 1 1	1	0	0	0

The output is clearly a Gray code since each number only changes by one bit relative to the previous number.

Problem 13.58

Solution:

Known quantities:

Four Boolean expressions.

Find:

a) Show that these four expressions represent the conversion from 4-bit Gray code to 4-bit binary numbers.
b) Draw the circuit which implements the conversion.

Analysis:

a) Note that:

$$B_3 = G_3$$

$$B_2 = G_3 \oplus G_2 = B_3 \oplus G_2$$

$$B_1 = G_3 \oplus G_2 \oplus G_1 = B_2 \oplus G_1$$

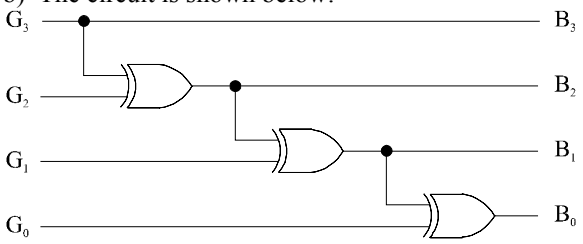
$$B_0 = G_3 \oplus G_2 \oplus G_1 \oplus G_0 = B_1 \oplus G_0$$

Then, the truth table is:

$G_3G_2G_1G_0$	B_3 G_3	B_2 $G_3 \oplus G_2$	B_1 $B_2 \oplus G_1$	B_0 $B_1 \oplus G_0$
0 0 0 0	0	0	0	0
0 0 0 1	0	0	0	1
0 0 1 0	0	0	1	0
0 0 1 1	0	0	1	1
0 1 0 0	0	1	0	0
0 1 0 1	0	1	0	1
0 1 1 0	0	1	1	0
0 1 1 1	0	1	1	1
1 0 0 0	1	0	0	0
1 0 0 1	1	0	0	1
1 0 1 0	1	0	1	0
1 0 1 1	1	0	1	1
1 1 0 0	1	1	0	0
1 1 0 1	1	1	0	1
1 1 1 0	1	1	1	0
1 1 1 1	1	1	1	1

The table verifies that the claim is correct.

b) The circuit is shown below:



Problem 13.59

Solution:

Known quantities:

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

Find:

The inputs for a 4-input multiplexer to implement the function.

Assumptions:

The inputs I_0, I_1, I_2, I_3 correspond to $\overline{A}\overline{B}$, $\overline{A}B$, $A\overline{B}$ and AB respectively, and each input may be 0, 1, $\overline{C}$ or C.

Analysis:

$f = \overline{A}B\overline{C} + A\overline{B}\overline{C} + AC$

BC \ A	0	1
00	0	1
01	0	1
11	0	1
10	1	0

From the truth table it is clear that:

$I_0 = 0$

$I_1 = \overline{C}$

$I_2 = 1$

and

$I_3 = C$

Problem 13.60

Solution:

Known quantities:

The function $f(A,B,C,D)=\sum(2,5,6,8,9,10,11,13,14)_{10}$.

Find:

The inputs for an 8-bit multiplexer to implement the function.

Assumptions:

The inputs I_0 through I_7 correspond to $\overline{A}\overline{B}\overline{C}$, $A\overline{B}\overline{C}$, $\overline{A}B\overline{C}$, $AB\overline{C}$, $\overline{A}\overline{B}C$, $A\overline{B}C$, $\overline{A}BC$ and ABC respectively, and each input may be 0, 1, $\overline{D}$ or D.

Analysis:

CD \ AB	00	01	11	10
00	0	0	0	1
01	0	1	1	1
11	0	0	0	1
10	1	1	1	1

From the truth table it is clear that:

$I_0 = 0$

$I_1 = \overline{D}$

$I_2 = D$

$I_3 = \overline{D}$

$I_4 = 1$

$I_5 = 1$

$I_6 = D$

and

$I_7 = \overline{D}$
