CHAPTER 7

MULTI-LEVEL GATE CIRCUITS / NAND AND NOR GATES

Contents

7.1 Multi-Level Gate Circuits

7.2 NAND and NOR Gates

7.3 Design of Two-Level Circuits Using NAND and NOR Gates

7.4 Design of Multi-Level NAND- and NOR-Gate Circuits

7.5 Circuit Conversion Using Alternative Gate Symbols

7.6 Design of Two-Level, Multiple-Output Circuits

7.7 Multiple-Output NAND and NOR Circuits

Objectives

Topics introduced in this chapter:

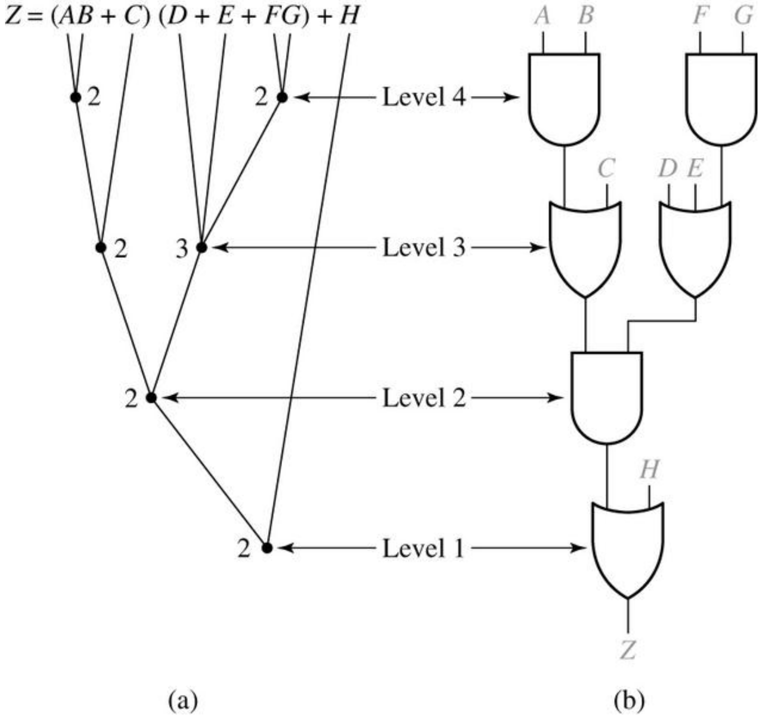
- Design a minimal two-level or multi-level circuit
- Design or analyze a two-level gate circuit
- Design or analyze a multi-level gate circuit
- Convert circuits by adding or deleting inversion bubbles
- Design a minimal two-level or multiple-output circuit using Karnaugh maps

7.1 Multi-Level Gate Circuits

- Terminology

AND-OR, OR-AND, OR-AND-OR, AND and OR

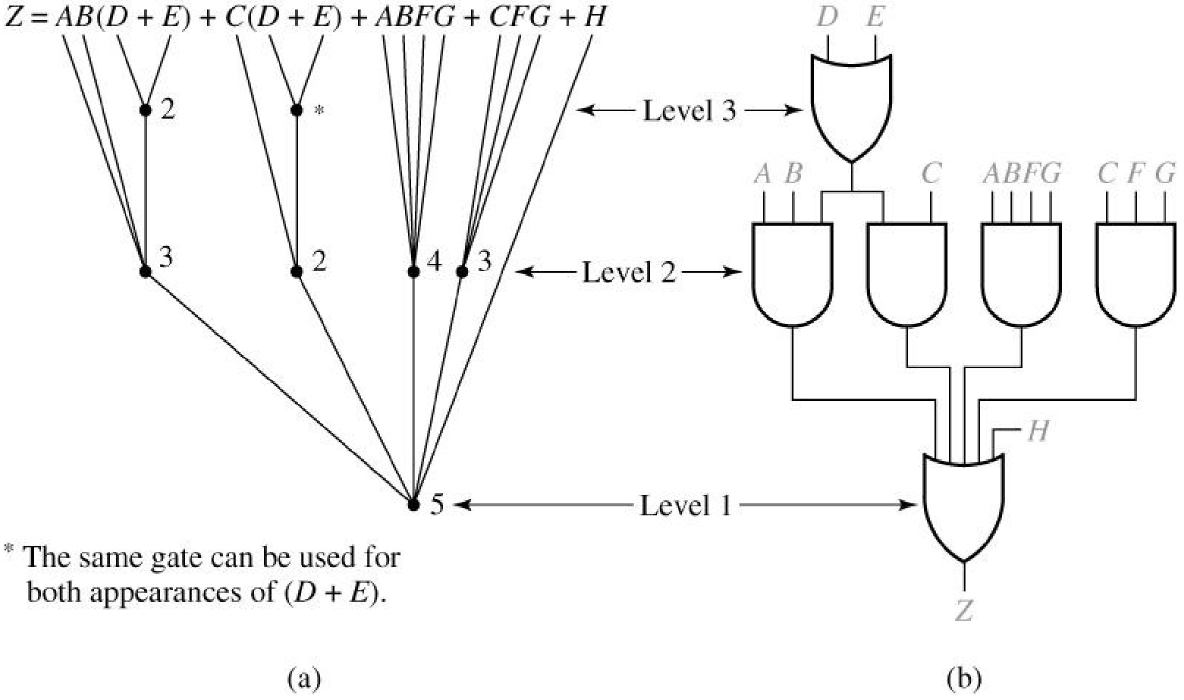
Four-Level Realization of Z



7.1 Multi-Level Gate Circuits

Three-Level Realization of Z

$$
\begin{aligned}
Z &= (AB+C)[(D+E)+FG]+H \\
&= AB(D+E)+C(D+E)+ABFG+CFG+H
\end{aligned}
$$

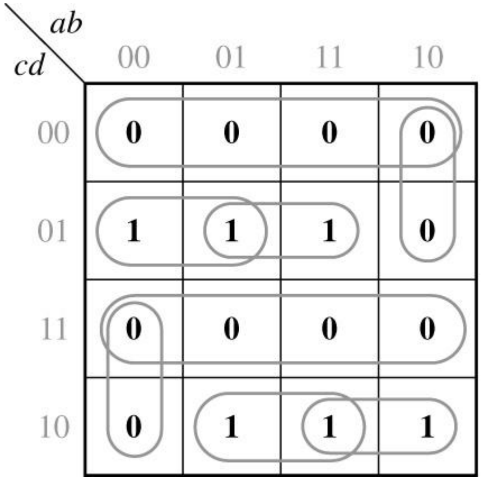


* The same gate can be used for both appearances of $(D + E)$.

7.1 Multi-Level Gate Circuits

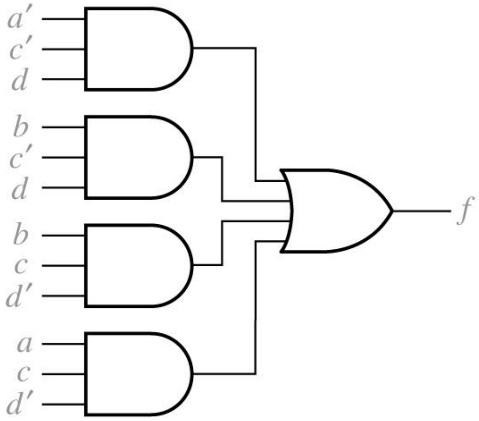
Example : Multi-Level Design Using AND and OR Gates

$$f(a,b,c,d) = \sum m(1,5,6,10,13,14)$$



cd \ ab	00	01	11	10
00	0	0	0	0
01	1	1	1	0
11	0	0	0	0
10	0	1	1	1

$f = a'c'd + bc'd + bcd' + acd'$ →

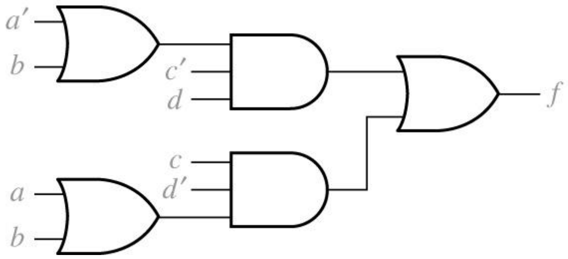


Two levels
Five gates
16 gate inputs

Two-level **AND-OR** gate

7.1 Multi-Level Gate Circuits

$$\begin{aligned} f &= a'c'd + bc'd + bcd' + acd' \\ &= c'd(a'+b) + cd'(a+b) \end{aligned} \longrightarrow$$



Three levels
Five gates
12 gate Inputs

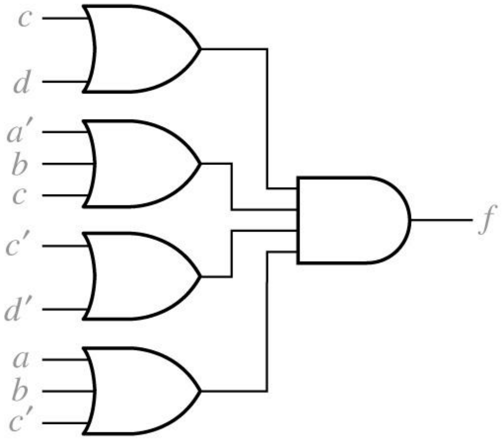
Three-level **OR-AND-OR** gate

From 0’s on the Karnaugh map

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

$$\Big\downarrow f''$$

$$f = (c+d)(a'+b+c)(c'+d')(a+b+c') \longrightarrow$$



Two levels
Five gates
14 gate inputs

Two-level **OR-AND** gate

7.1 Multi-Level Gate Circuits

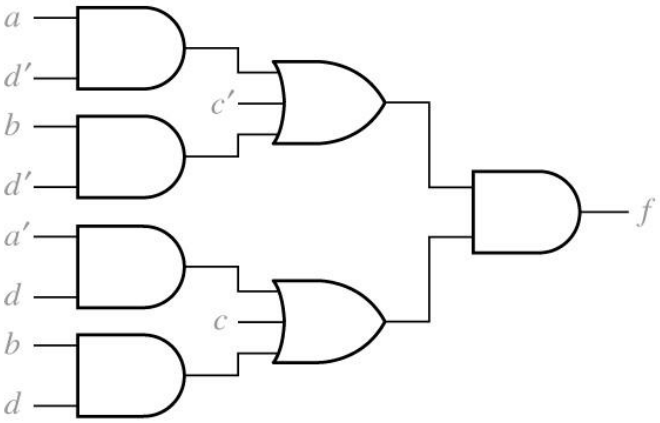
Using $(X+Y)(X+Z)=X+YZ$

$$f=[c+d(a'+b)][c'+d'(a+b)]$$

If we multiply out $d'(a+b)$ and $d(a'+b)$

$$f=(c+a'd+bd)(c'+ad'+bd')$$

$$f'=c'(d'+ab')+c(d+a'b')$$
$$=c'(d'+a)(d'+b')+c(d+a')(d+b')$$



Three levels
Seven gates
16 gate inputs

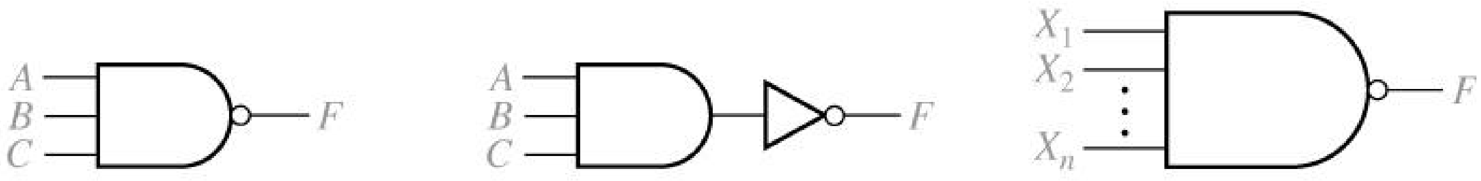
Three-level **AND-OR-AND** gate

7.2 NAND and NOR Gates

NAND gate

$$F = (ABC)' = A' + B' + C'$$

$$F = (X_1 X_2 \ldots X_n)' = X_1{}' + X_2{}' + \ldots + X_n{}'$$



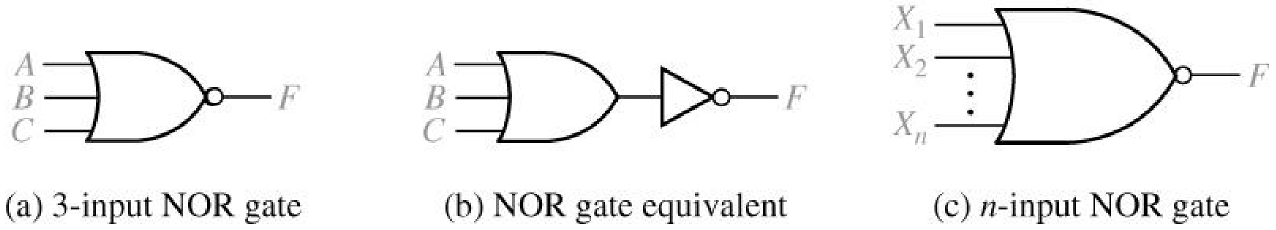
(a) 3-input NAND gate (b) NAND gate equivalent (c) *n*-input NAND gate

7.2 NAND and NOR Gates

NOR gate

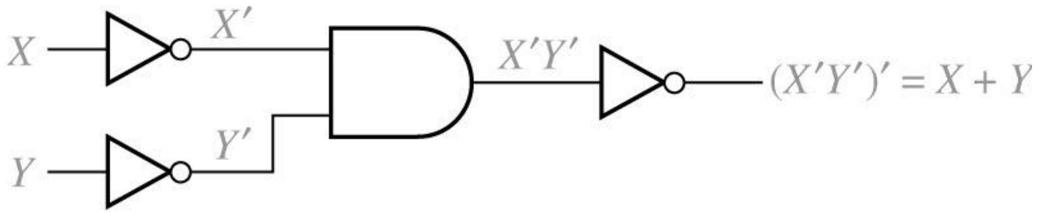
$$F = (A+B+C)' = A'B'C'$$

$$F = (X_1 + X_2 + \ldots + X_n)' = X_1{}' X_2{}' \ldots X_n{}'$$



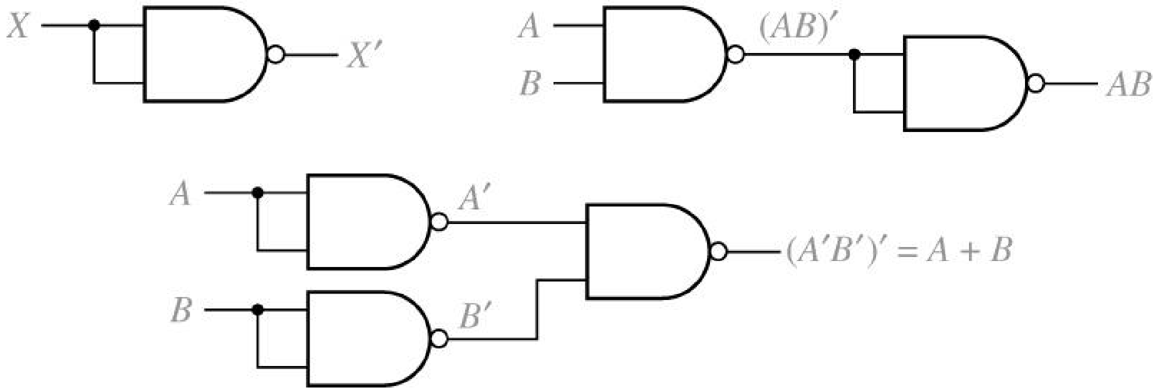
(a) 3-input NOR gate (b) NOR gate equivalent (c) *n*-input NOR gate

OR realized by using AND and NOT



7.2 NAND and NOR Gates

NAND gate realization of NOT, AND, and OR



AND realized by using OR and NOT

$$XY = \left(X' + Y'\right)'$$

7.3 Design of Two-Level Circuits Using NAND and NOR Gates

DeMorgan's laws

$$(X_1 + X_2 + \ldots + X_n)' = X_1{}' X_2{}' \ldots X_n{}'$$

$$(X_1 X_2 \ldots X_n)' = X_1{}' + X_2{}' + \ldots + X_n{}'$$

Conversion of a sum-of-products to several other two-level forms

$$F = A + BC' + B'CD = \left[(A + BC' + B'CD)' \right]' \qquad \text{AND-OR}$$

$$= \left[A' \cdot (BC')' \cdot (B'CD)' \right]' \qquad \text{NAND-NAND}$$

$$= [A' \cdot (B' + C) \cdot (B + C' + D')]' \qquad \text{OR-NAND}$$

$$= A + (B' + C)' + (B + C' + D')' \qquad \text{NOR-OR}$$

7.3 Design of Two-Level Circuits Using NAND and NOR Gates

$$F=\left\{\left[A+(B'+C)'+(B+C'+D')'\right]'\right\}'$$

NOR−NOR−INVERT

$$F=(A+B+C)(A+B'+C')(A+C'+D)$$

OR−AND

$$=\left\{\left[(A+B+C)(A+B'+C')(A+C'+D)\right]'\right\}'$$

$$=\left[(A+B+C)'+(A+B'+C')'+(A+C'+D)'\right]'$$

NOR−NOR

$$=(A'B'C'+A'BC+A'CD')'$$

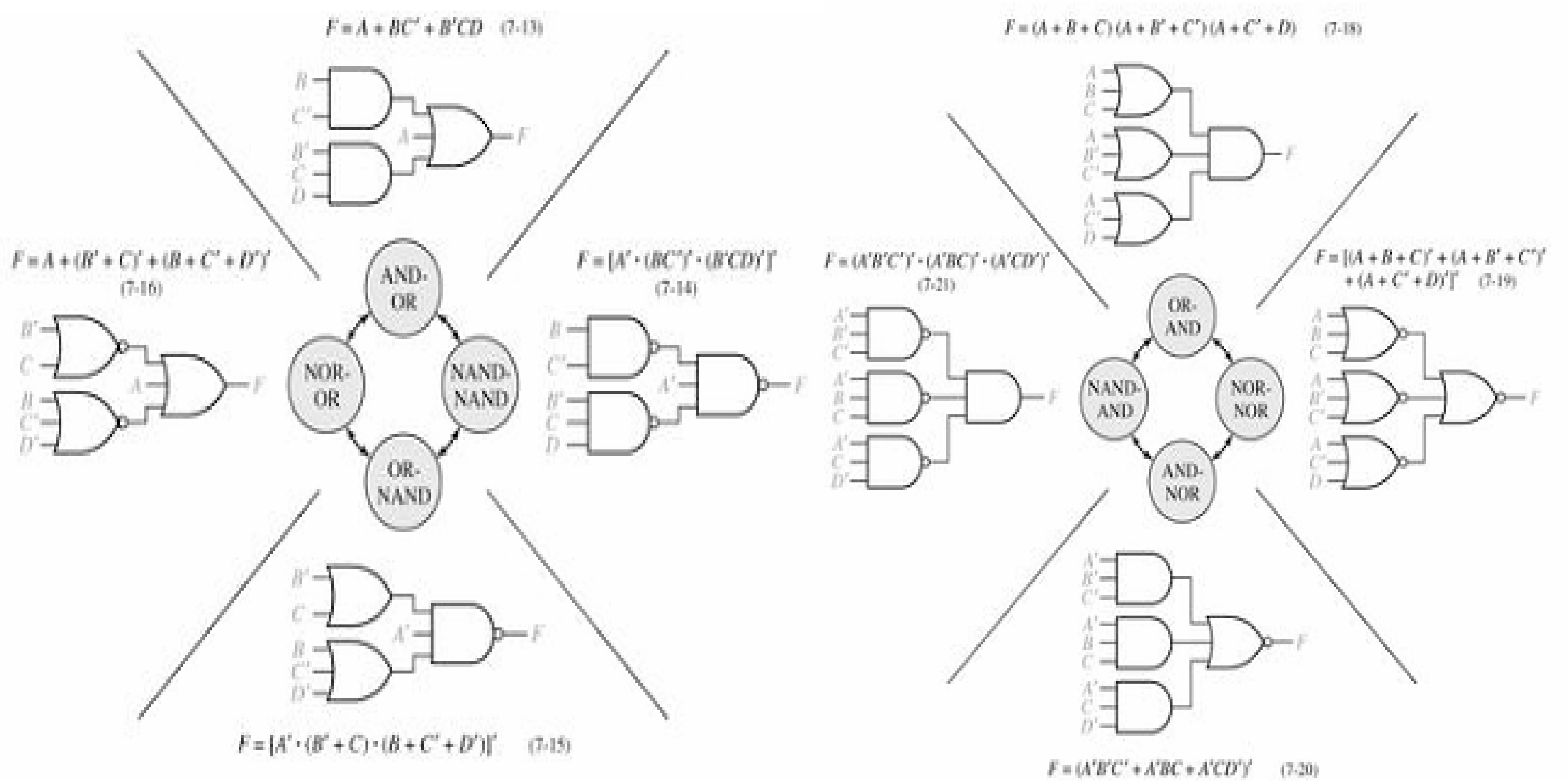
AND−NOR

$$=(A'B'C')'\cdot(A'BC)'\cdot(A'CD')'$$

NAND−AND

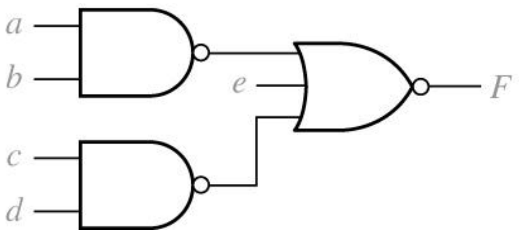
7.3 Design of Two-Level Circuits Using NAND and NOR Gates

Eight Basic Forms for Two-Level Circuits (p.186)


F = A + BC' + B'CD (7-13)
F = [A' · (BC')' · (B'CD)']' (7-14)
F = [A' · (B' + C) · (B + C' + D')]' (7-15)
F = A + (B' + C)' + (B + C' + D')' (7-16)
AND-OR
NAND-NAND
OR-NAND
NOR-OR
F = (A + B + C)(A + B' + C')(A + C' + D) (7-18)
F = [(A + B + C)' + (A + B' + C')' + (A + C' + D)']' (7-19)
F = (A'B'C' + A'BC + A'CD')' (7-20)
F = (A'B'C')' · (A'BC)' · (A'CD')' (7-21)
OR-AND
NOR-NOR
AND-NOR
NAND-AND
A
B
C
D
A'
B'
C'
D'
F

7.3 Design of Two-Level Circuits Using NAND and NOR Gates

NAND–NOR

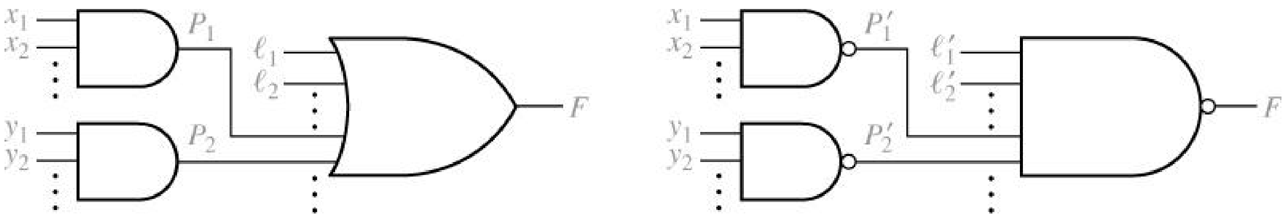


$F = [(ab)' + (cd)' + e]' = abcde'$

AND–OR to NAND–NAND Transformation

$(l_1, l_2 \ldots)$: literals $(P_1, P_2 \ldots)$: product terms

$$F = l_1 + l_2 + \ldots + P_1 + P_2 + \ldots = \left(l_1{}' l_2{}' \cdots P_1{}' P_2{}' \cdots \right)'$$



(a) Before transformation

(b) After transformation

7.4 Design of Multi-Level NAND- and NOR-Gates Circuits

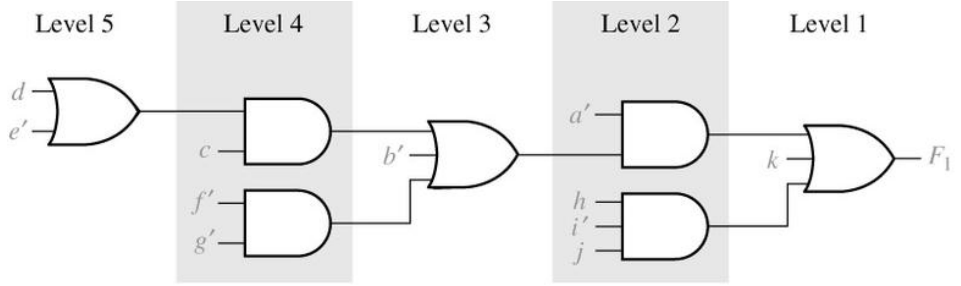
•Procedure : multi-level NAND-gate circuits

- Simplify the switching function
- Design a multi-level circuit of AND and OR gates
- Number the levels starting with the output gate as level 1
- Replace all gates with NAND gates, leaving all interconnections between gates unchanged
- Leave the inputs to levels 2,4,6,... unchanged
- Invert any literals which appear as inputs to levels 1, 3, 5, ...

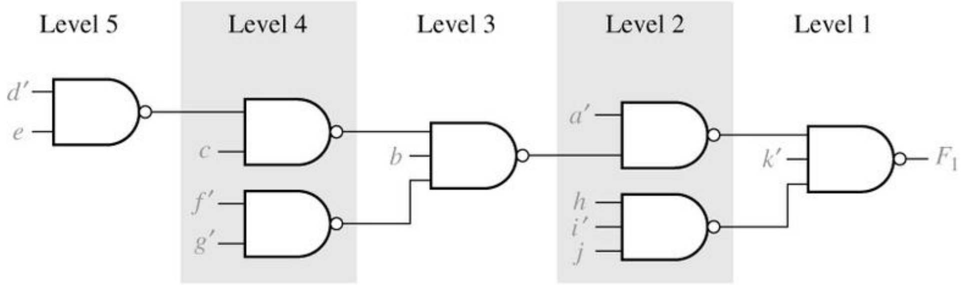
7.4 Design of Multi-Level NAND- and NOR-Gates Circuits

Example : Multi-Level Circuit Conversion to NAND Gates

$$F_1 = a'[b' + c(d + e') + f'g'] + hi'j + k$$



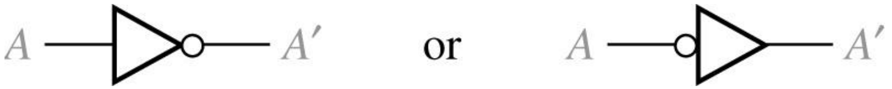
(a) AND-OR network



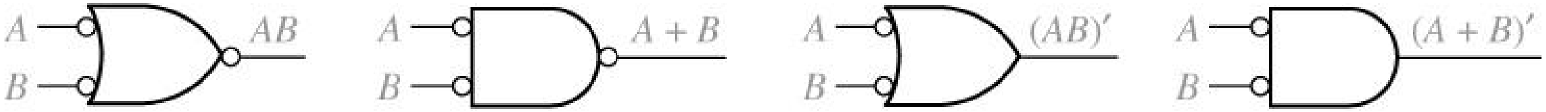
(b) NAND network

7.5 Circuit Conversion Using Alternative Gate Symbols

Inverter



Alternative Gate Symbols



$AB = (A' + B')'$

(a) AND

$A + B = (A'B')'$

(b) OR

$(AB)' = A' + B'$

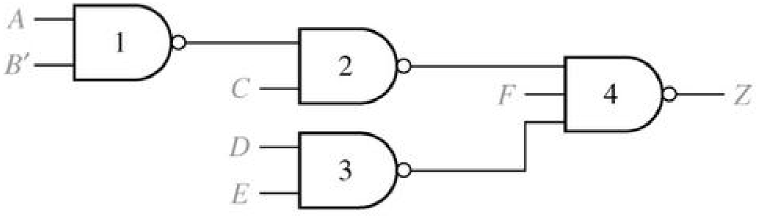
(c) NAND

$(A + B)' = A'B'$

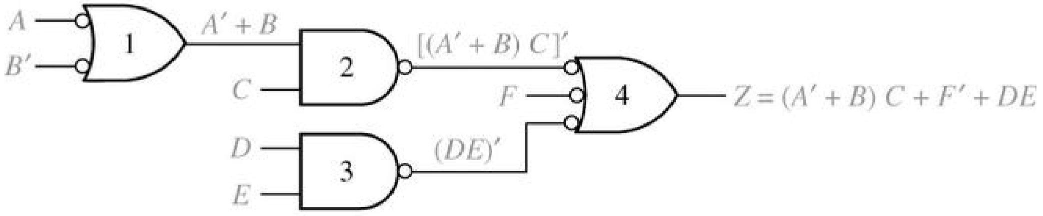
(d) NOR

7.5 Circuit Conversion Using Alternative Gate Symbols

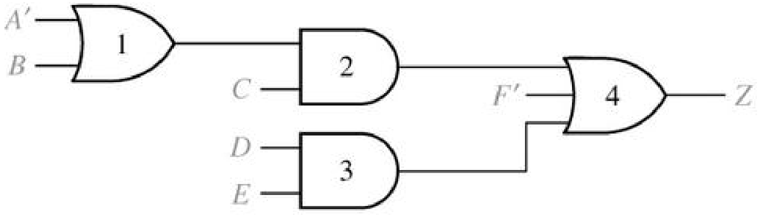
NAND Gate Circuit Conversion



(a) NAND gate network



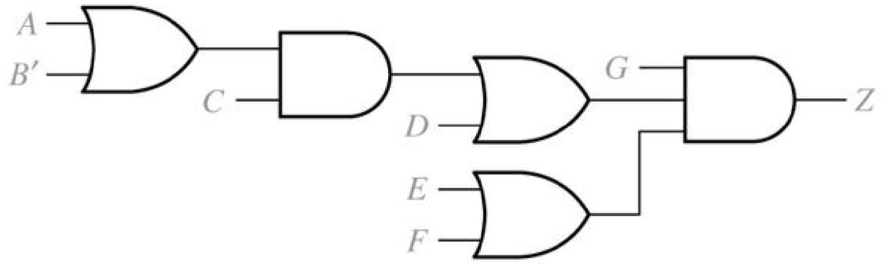
(b) Alternate form for NAND gate network



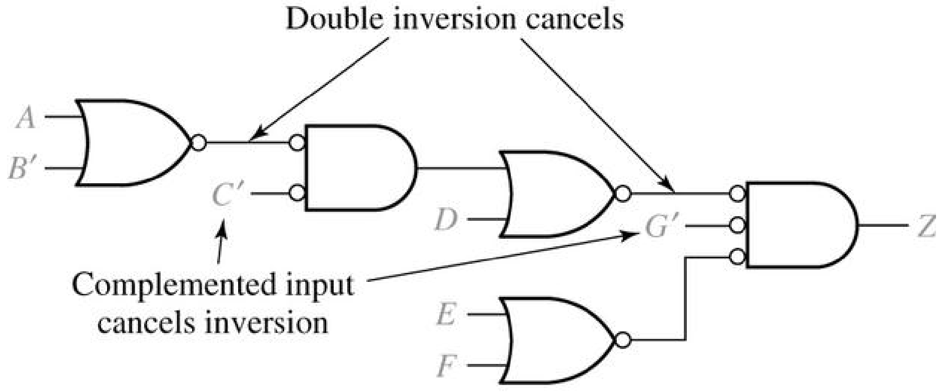
(c) Equivalent AND-OR network

7.5 Circuit Conversion Using Alternative Gate Symbols

Conversion to NOR Gates



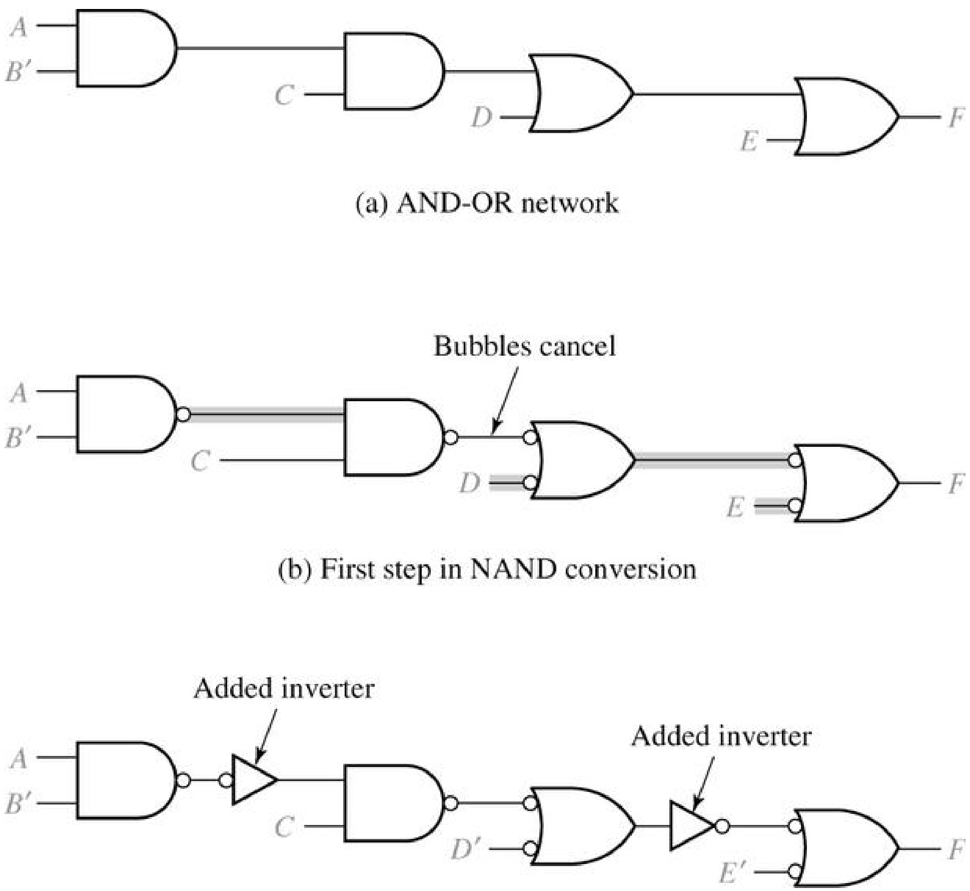
(a) Circuit with OR and AND gates



(b) Equivalent circuit with NOR gates

7.5 Circuit Conversion Using Alternative Gate Symbols

Conversion of AND-OR Circuits to NAND Gates



(a) AND-OR network

(b) First step in NAND conversion

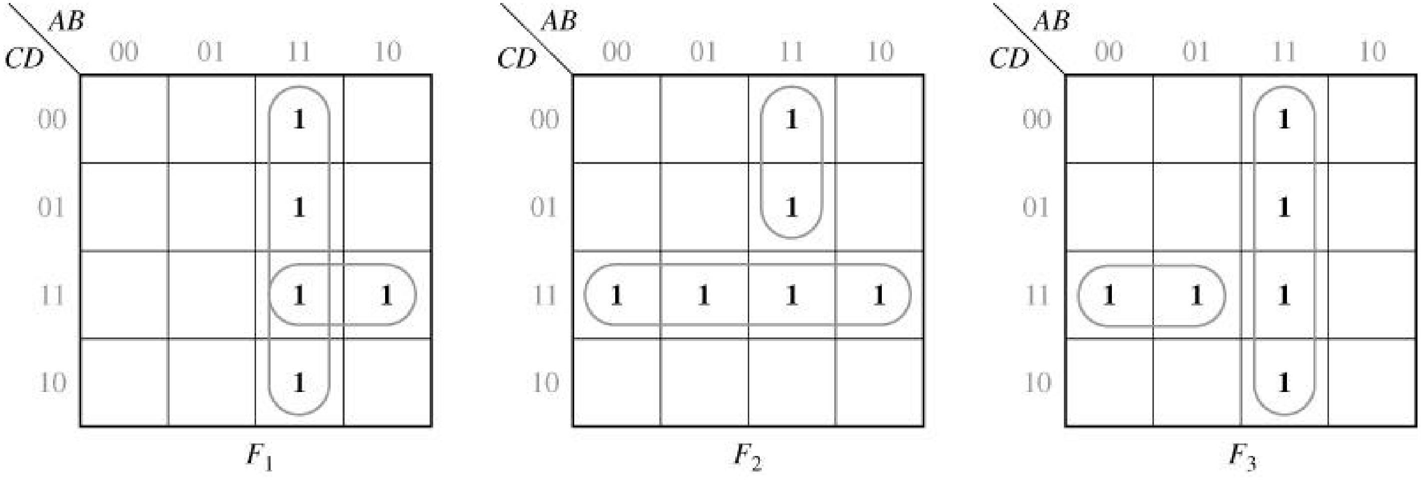
(c) Completed conversion

7.6 Design of Two-Level,Multiple-Output Circuits

Example : Design a circuit with four inputs and three outputs

$$F_1(A,B,C,D)=\sum m(11,12,13,14,15)$$
$$F_2(A,B,C,D)=\sum m(3,7,11,12,13,15)$$
$$F_3(A,B,C,D)=\sum m(3,7,12,13,14,15)$$

Karnaugh Maps for Equations



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

F_1

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

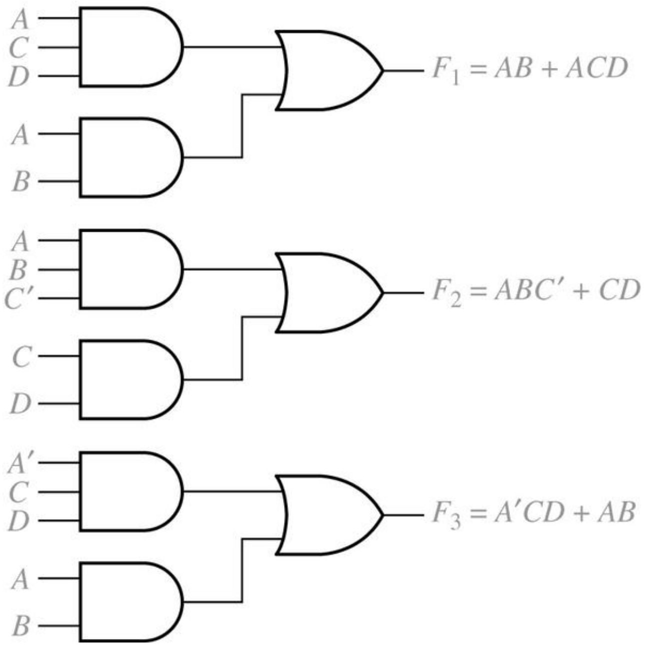
F_2

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

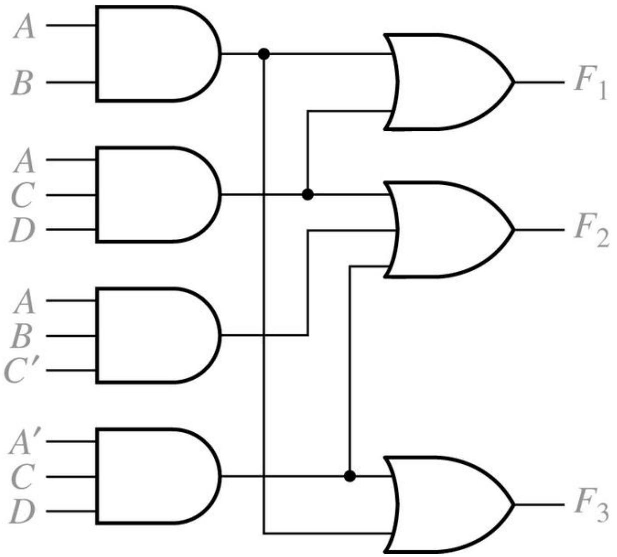
F_3

7.6 Design of Two-Level,Multiple-Output Circuits

Realization of Equations



Multiple-Output Realization of Equations

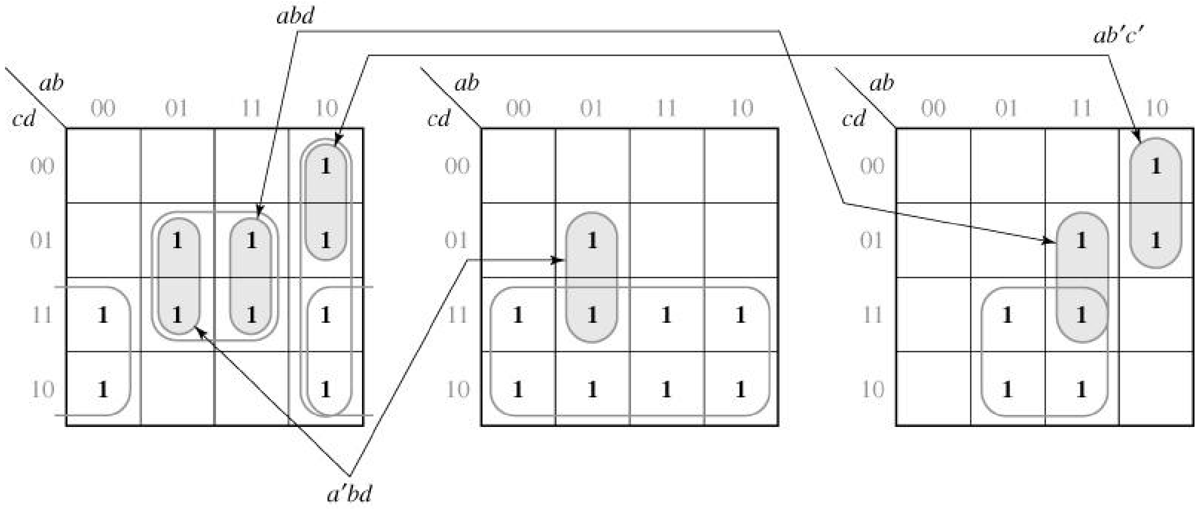


7.6 Design of Two-Level,Multiple-Output Circuits

Example : Design a multiple-output circuit with 4-inputs and 3-outputs

$$f_1 = \sum m(2,3,5,7,8,9,10,11,13,15)$$
$$f_2 = \sum m(2,3,5,6,7,10,11,14,15)$$
$$f_3 = \sum m(6,7,8,9,13,14,15)$$

Karnaugh Maps for Equations



cd \ ab	00	01	11	10
00				1
01		1	1	1
11	1	1	1	1
10	1			1

cd \ ab	00	01	11	10
00				
01		1		
11	1	1	1	1
10	1	1	1	1

cd \ ab	00	01	11	10
00				1
01			1	1
11		1	1	
10		1	1	

7.6 Design of Two-Level,Multiple-Output Circuits

Minimized equations

$$f_1 = bd + b'c + ab'$$

$$f_2 = c + a'bd$$

$$f_3 = bc + ab'c' + \left\{ \begin{matrix} abd \\ or \\ ac'd \end{matrix} \right\}$$

10gates,

25gate input

The minimal solution

$$f_1 = \underline{a'bd} + \underline{abd} + \underline{ab'c'} + b'c$$

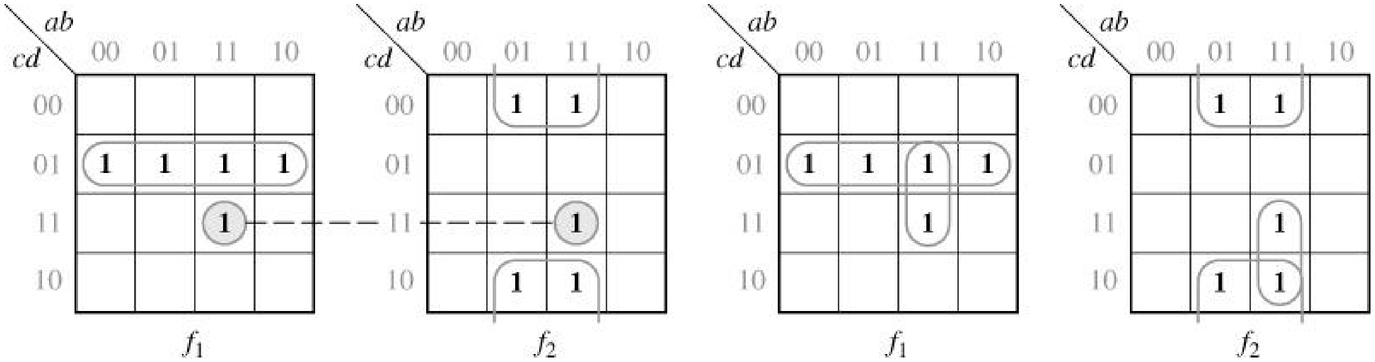
$$f_2 = c + \underline{a'bd}$$

$$f_3 = bc + \underline{ab'c'} + \underline{abd}$$

8 gates
22 gate inputs

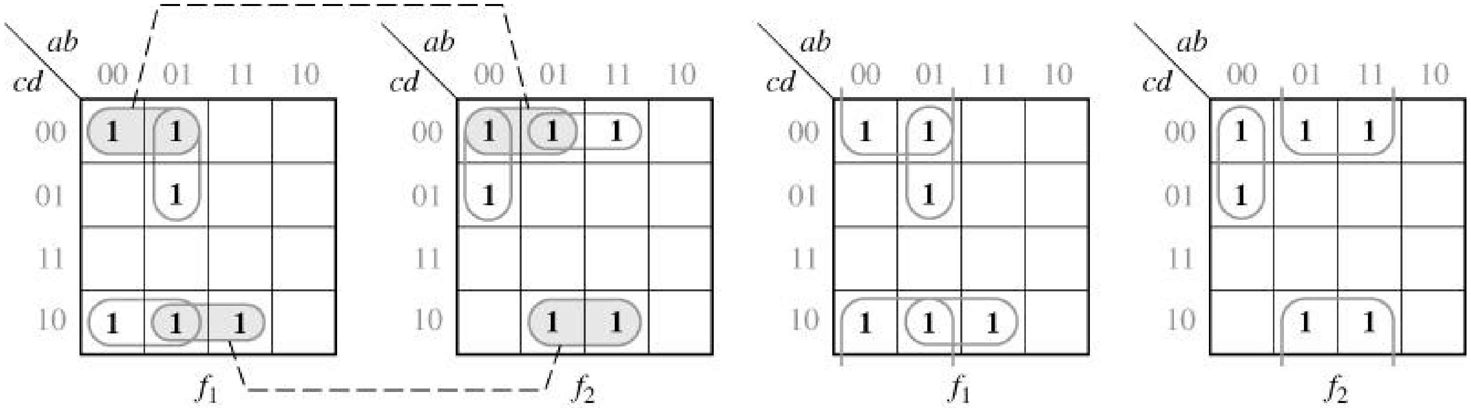
7.6 Design of Two-Level,Multiple-Output Circuits

Determination of Essential Prime Implicants for Multiple-Output Realization



(a) Best solution

(b) Solution requires and extra gate



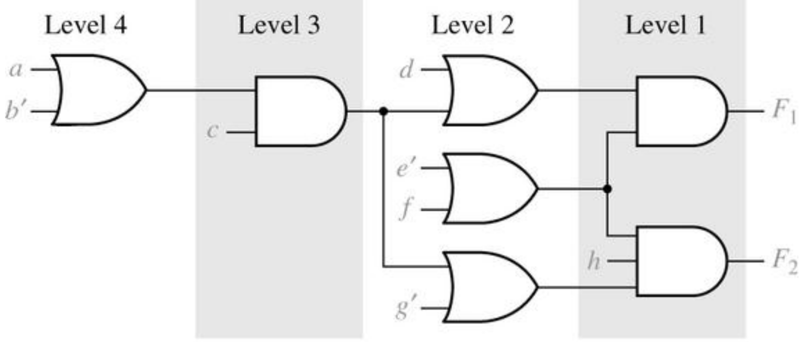
(a) Solution with maximum number of common terms requires 8 gates, 26 inputs

(b) Best solution requires 7 gates, 18 inputs and has no common terms

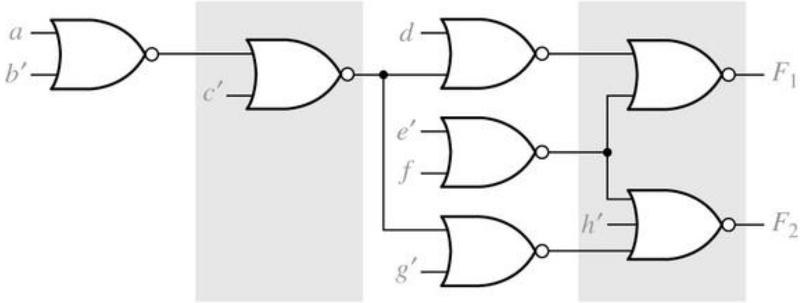
7.7 Multiple-Output NAND and NOR Circuits

Multi-level Circuits Conversion to NOR Gates

$$F_1 = [(a+b')c+d](e'+f) \qquad F_2 = [(a+b')c+g'](e'+f)h$$



(a) Network of AND and OR gates



(b) NOR network