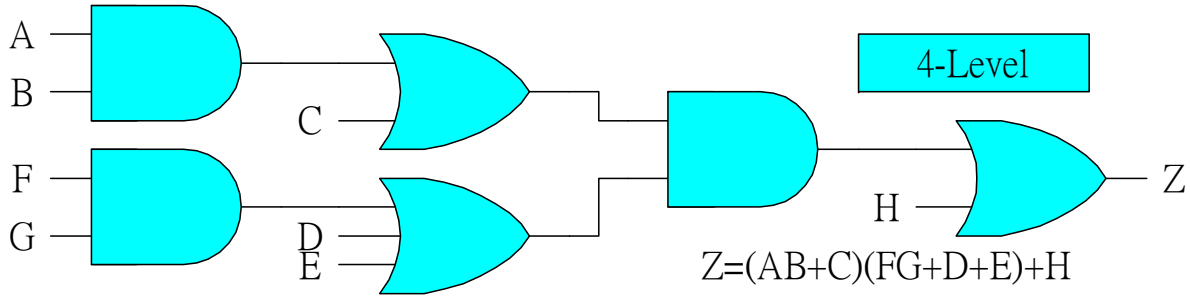
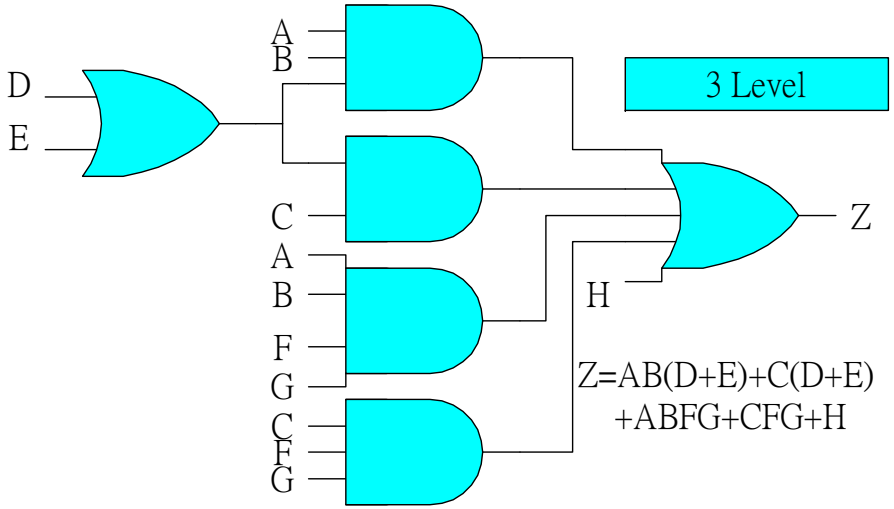
Unit 7.

Multi-Level Gate Networks
NAND and NOR Gates

Multi Level Network





AND – OR : 2 - level SOP

OR - AND : 2 - level POS

OR - AND - OR : 3 - level

Network of AND and OR

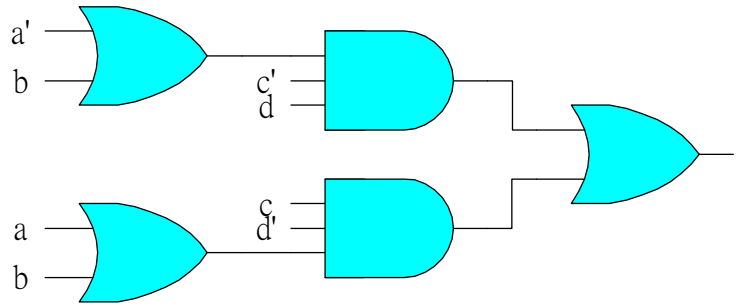
$\Rightarrow$ no particular ordering

Gate Number & Delays determine level

Factoring to accomplish different level..

Ex : Multi - Level Designed by Using AND and OR

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



Using K - map Method :

$$f = a'c'd + bc'd + bcd' + acd' \cdots\cdots (1)$$

2 - level AND - OR gate network

$\Rightarrow$ 2 levels, 5 gates, 16 gate inputs

Factoring :

$$f = c'd(a'+b) + cd'(a+b) \cdots\cdots (\boldsymbol{2})$$

$\Rightarrow$ 3 levels, 5 gates, 12 gate inputs

or Use POS :

$$f = (c+d)(a'+b+c)(c'+d')(a+b+c) \cdots\cdots (\boldsymbol{3})$$

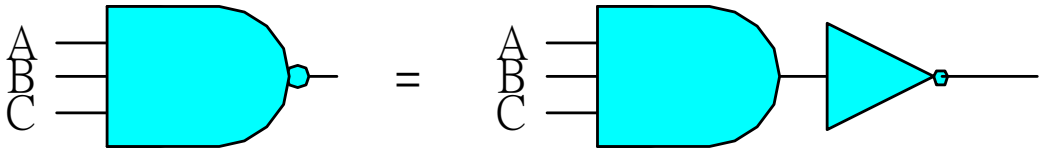
$\Rightarrow$ 2 levels, 5 gates, 14 gate inputs

$$\text{or } f = [c + d(a'+b)][c'+d'(a+b)] \cdots\cdots (\boldsymbol{4})$$

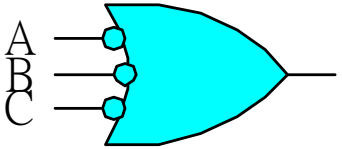
$\Rightarrow$ 3 levels, 7 gates, 12 gate inputs

Other Types of Gates

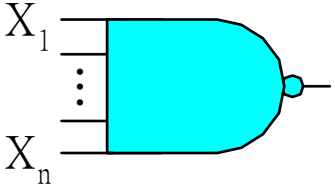
NAND :



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

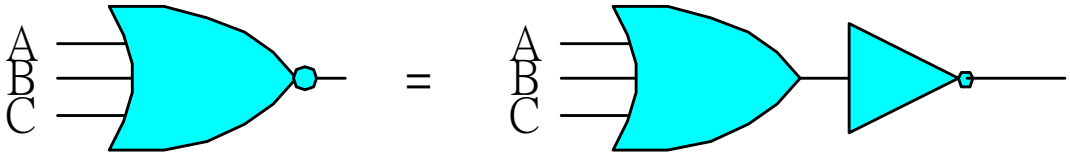


A	B	AB	$\overline{AB}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

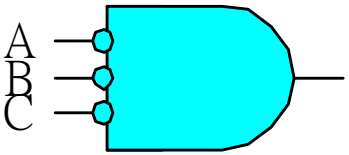


$$f=(x_1\cdot x_2\cdot \cdots \cdot x_n)'=x_1'+x_2'+\cdots+x_n'$$

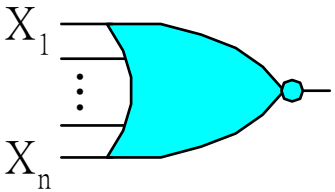
NOR :



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

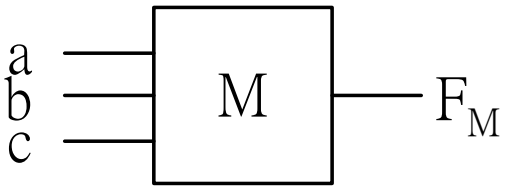


A	B	$A+B$	$\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

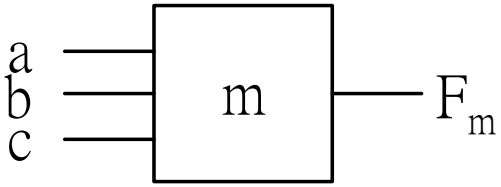


$$f=(x_1+x_2+\cdots+x_n)'=x_1'\cdot x_2'\cdot\cdots\cdot x_n'$$

Majority Gate



Minority Gate



$Majority\ of\ inputs = 1 \Rightarrow F_M = 1$

$Minority\ of\ inputs = 1 \Rightarrow F_m = 1$

Odd number of inputs

a	b	c	F_M	F_m
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

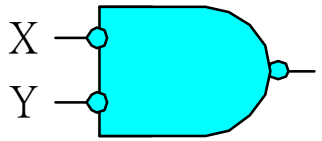
Functionally Complete Sets of Gates

A set of logic operations is Functionally Complete, if any Boolean function can be expressed in terms of this set of operations.

$\{AND, OR, NOT\}$

$\{AND, NOT\}$ $\because OR = X + Y = (X'Y')'$

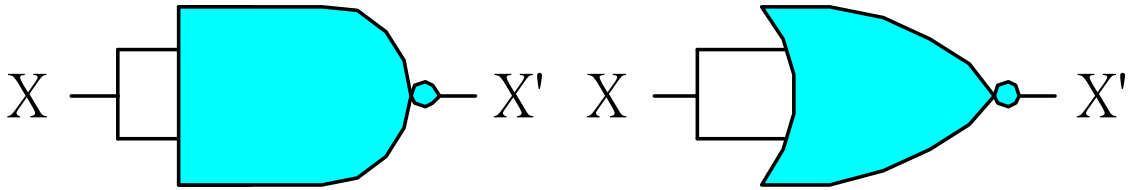
can be reduced by AND, NOT



$\{OR, NOT\}$

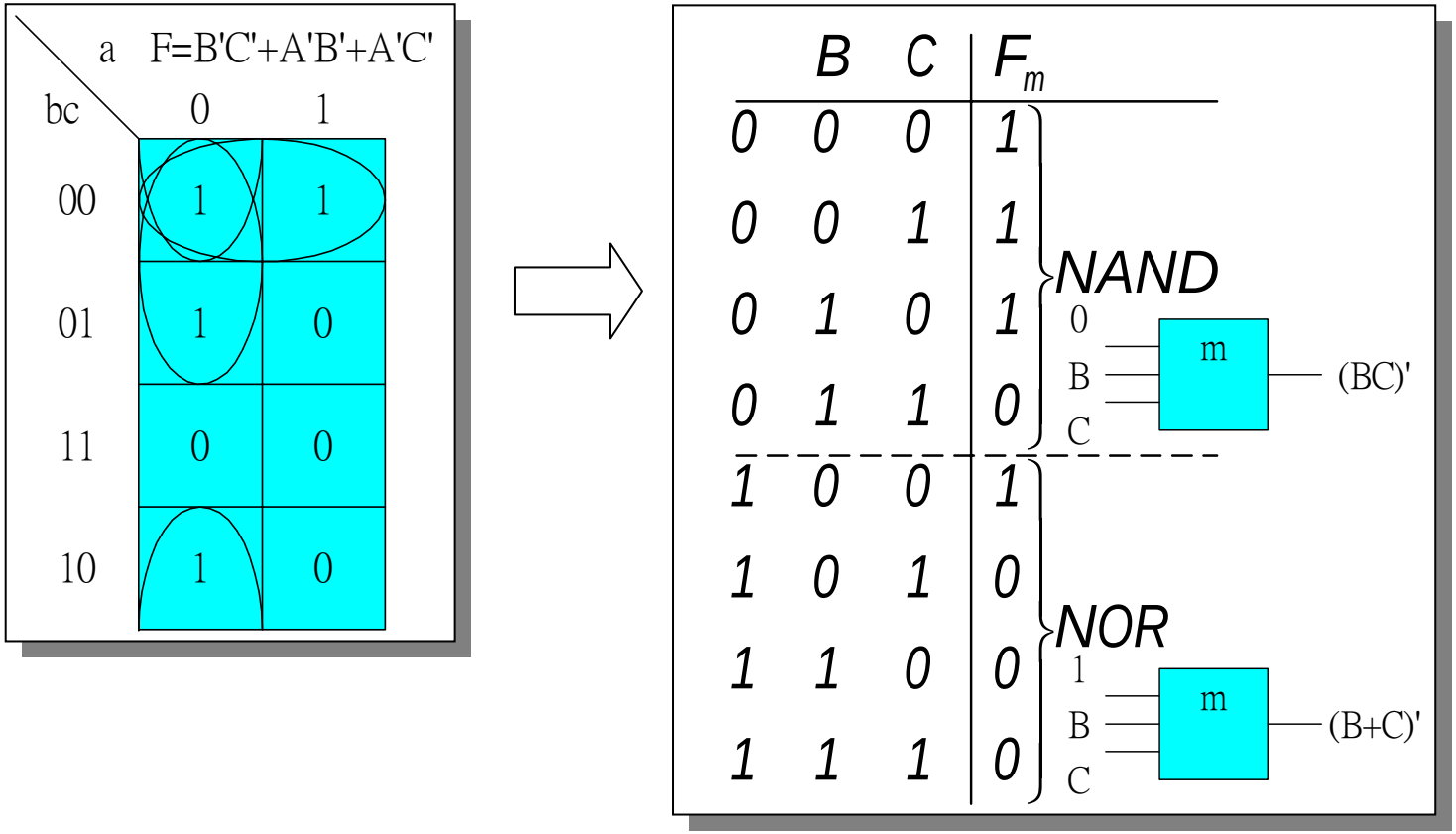
A functionally complete set of operations must contains NOT

NOT: Realized by NAND or NOR



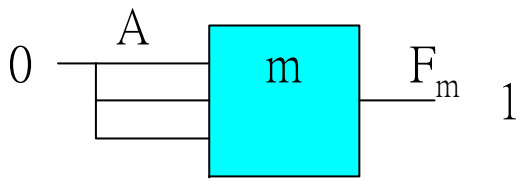
Example

3-input minority gates can realize any switching function

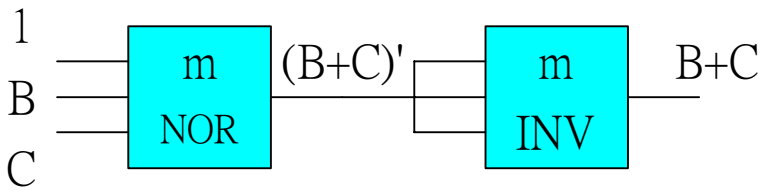


Example

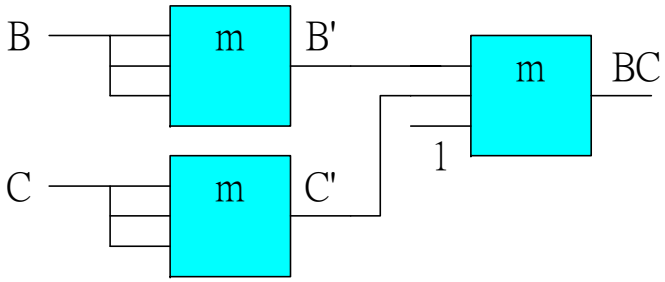
1. Inverter : Let C=B=A

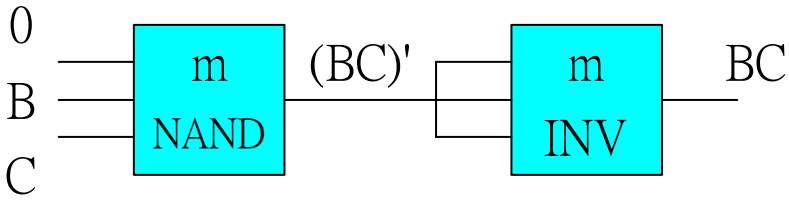


3. OR : B+C = 〔(B+C)'〕'



2. AND : Let A=1 then Fm =B'C'
Must invert B & C in advance

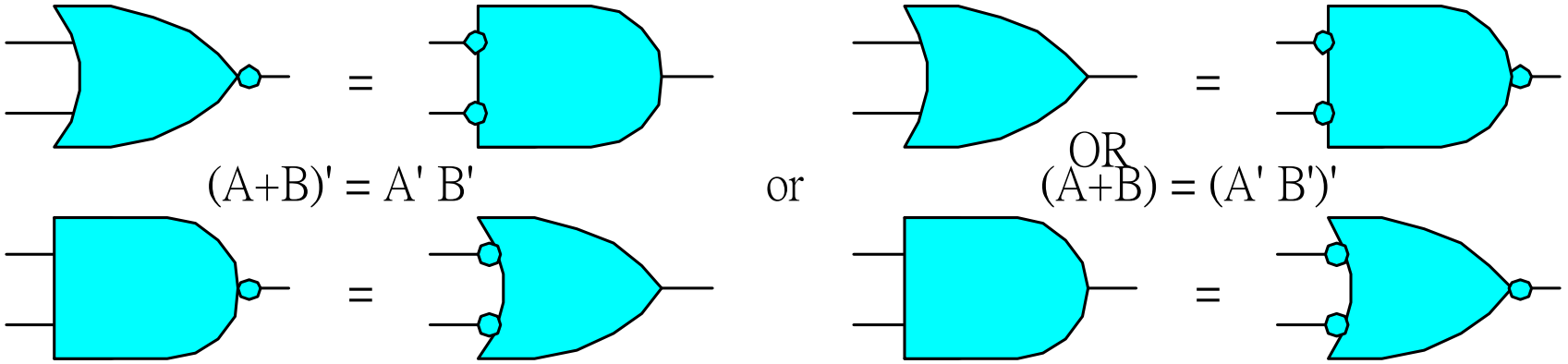




Design of 2-Level NAND-and-NOR Gate Networks

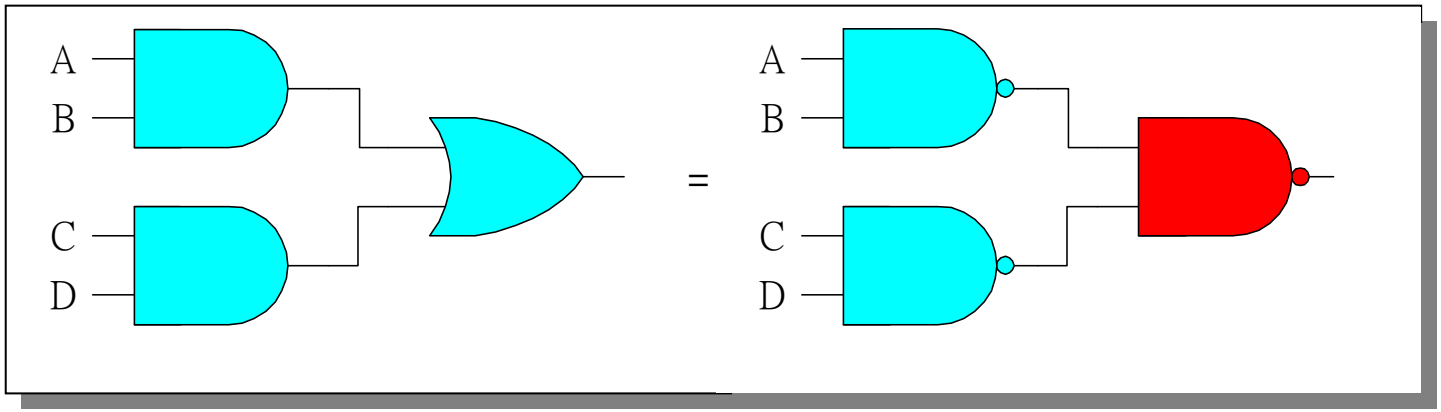
NAND & NOR : Readily available in IC form

DeMorgon's Law

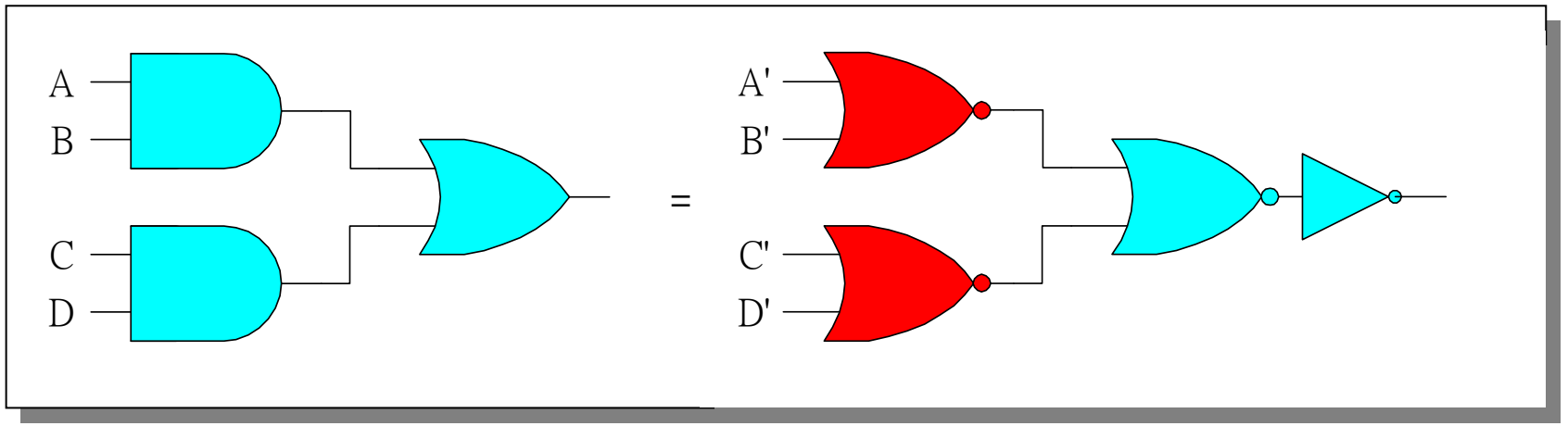


Sum of Product

Ex1 : AND / OR ⇒ NAND/NAND



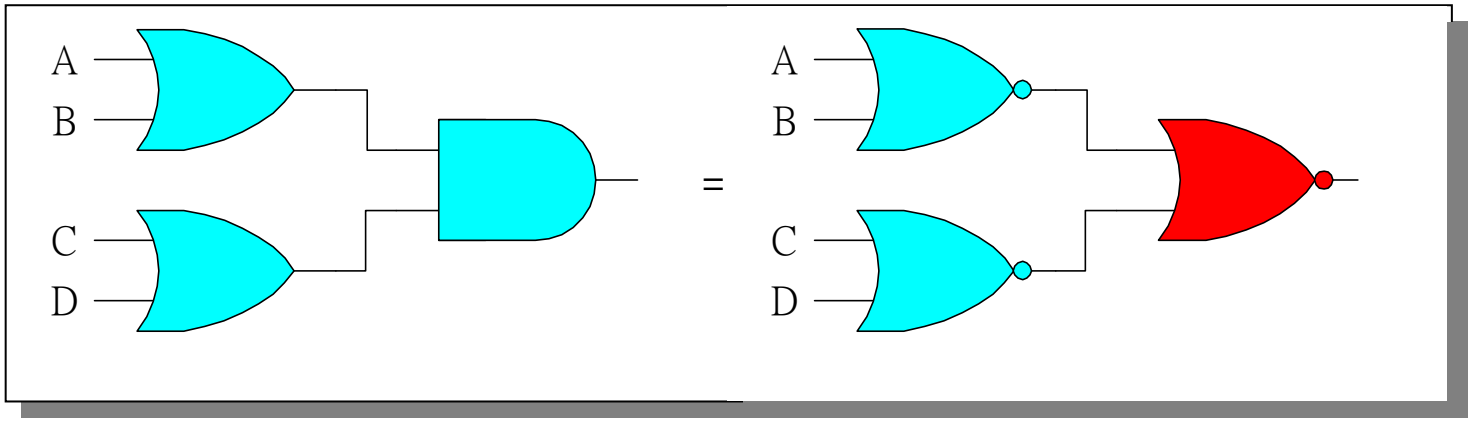
Ex2 : AND / OR ⇒ NOR/NOR



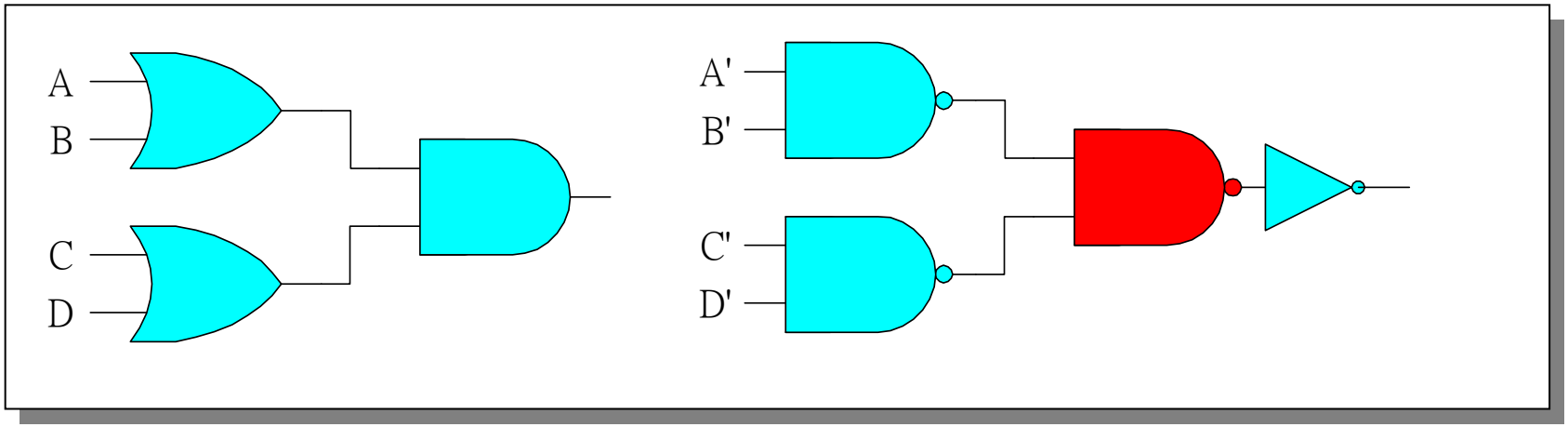
故，前者較自然..

Product of Sum

Ex1 : OR / AND ⇒ NOR/NOR

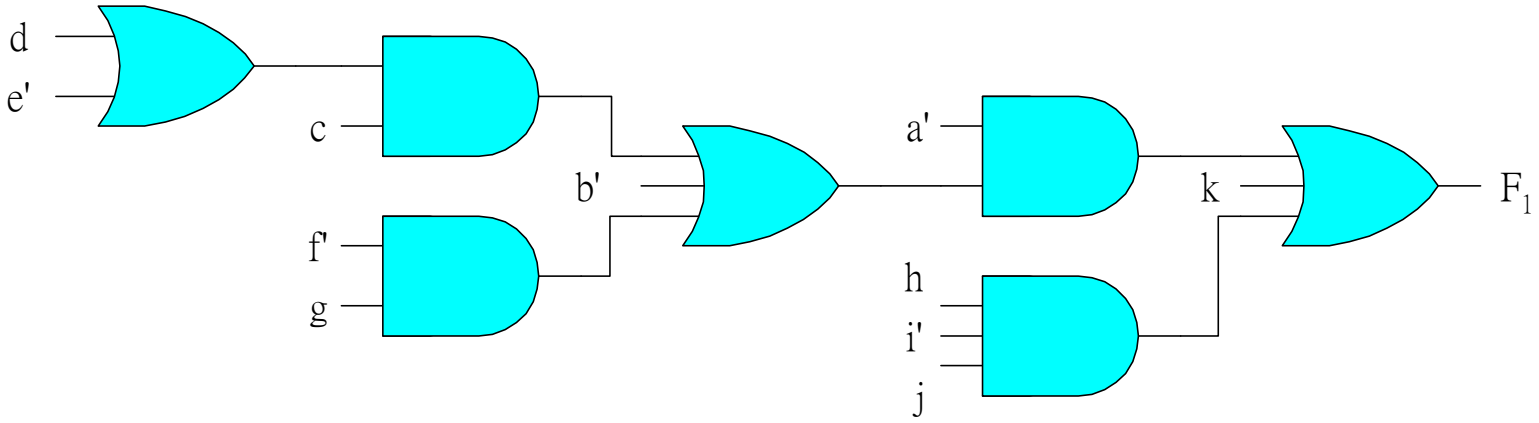


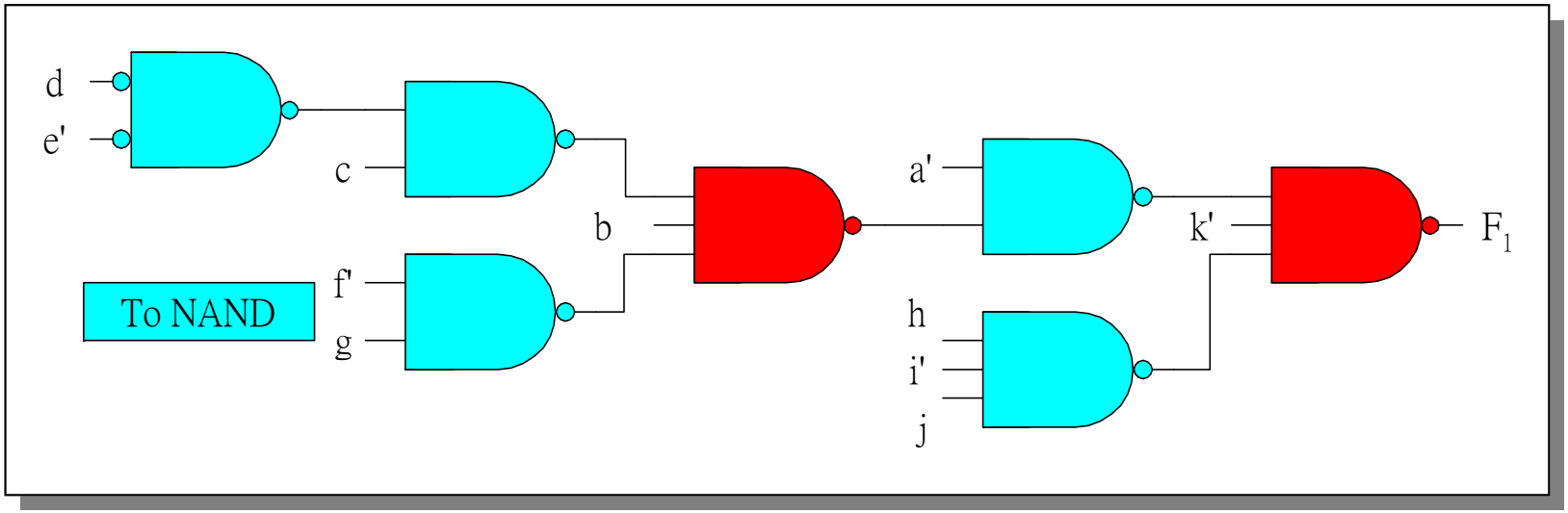
Ex2 : OR / AND ⇒ NAND/NAND



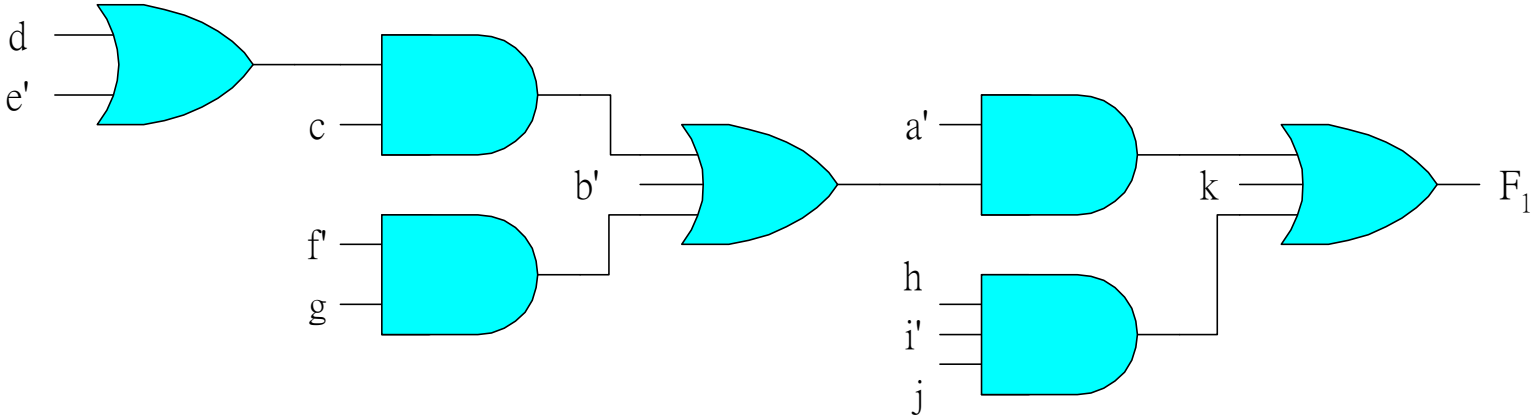
故，前者較自然..

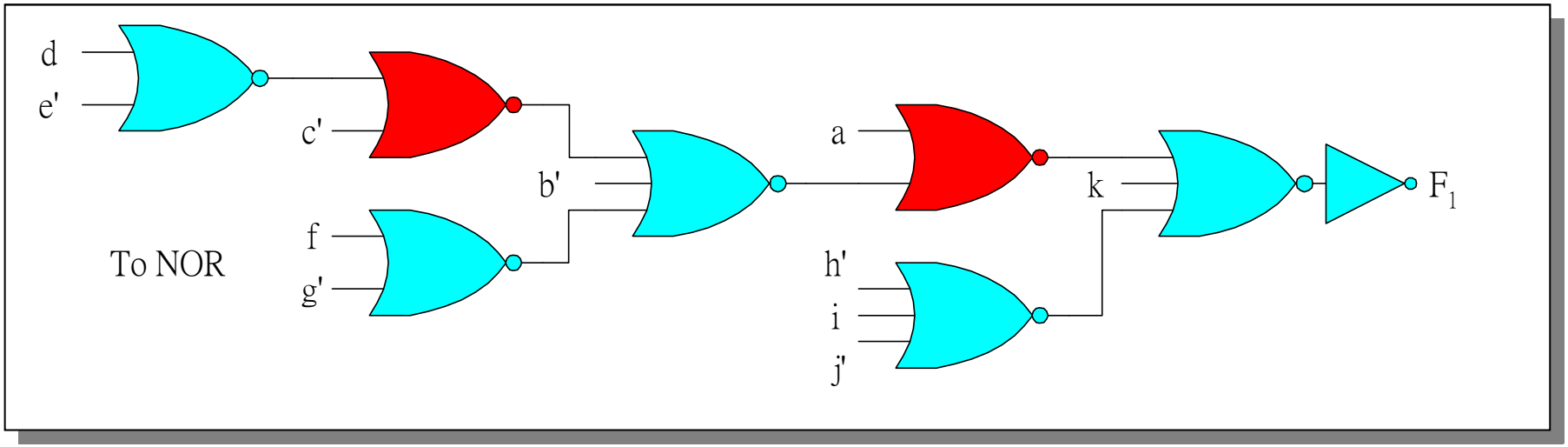
Multi-Level Network with NAND or NOR Gates





Multi-Level Network with NAND or NOR Gates (Cont.)





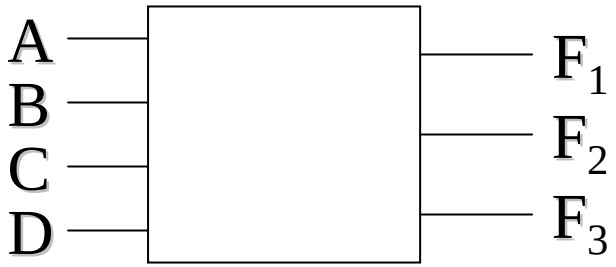
§ Design of Two Level Multiple - Output Networks

Ex :

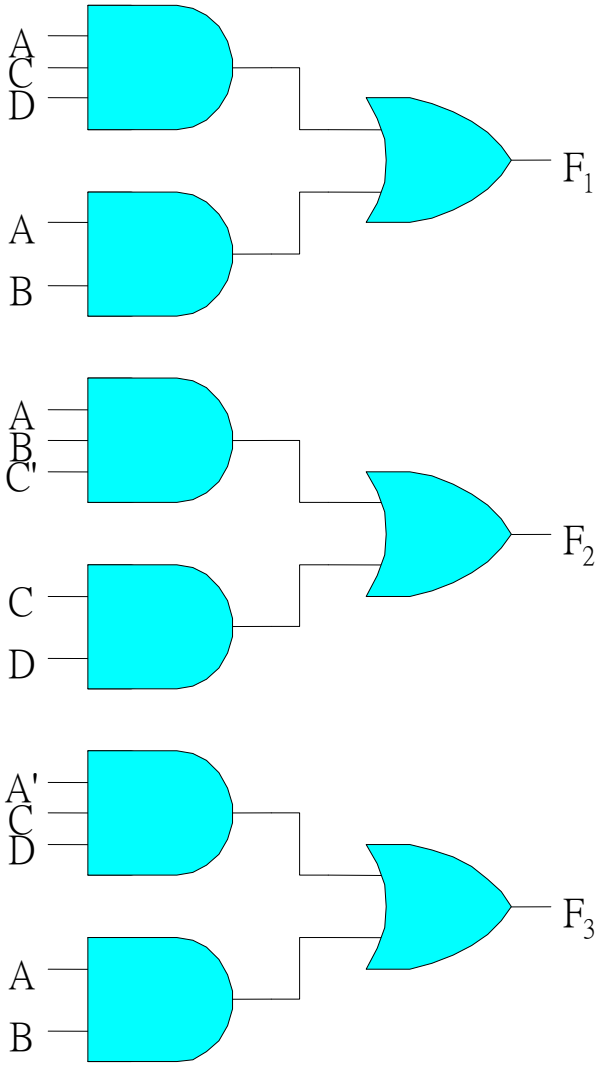
$$F_1(A,B,C,D) = \sum m(11,12,13,14,15)$$
$$F_2(A,B,C,D) = \sum m(3,7,11,12,13,15)$$
$$F_3(A,B,C,D) = \sum m(3,7,12,13,14,15)$$

From K - Map

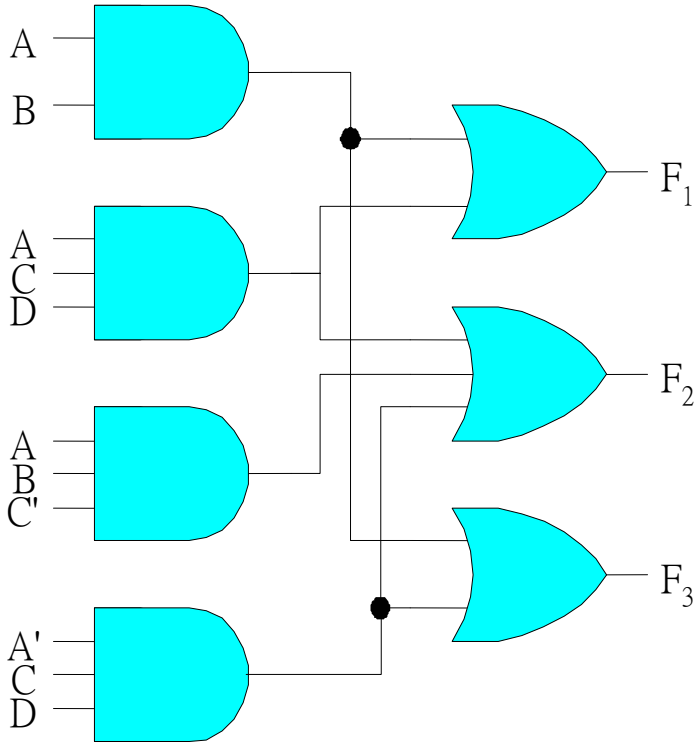
$$F_1 = AB + ACD$$
$$F_2 = ABC' + CD$$
$$F_3 = A'CD + AB$$



9 Gates, 21 Gate Inputs



7 Gates, 18 Gate Inputs



$$A'CD + ACD = CD$$

$$\rightarrow F_2 = (A'CD + ACD) + ABC'$$

$$= CD + ABC'$$

(1) 共用AB

(2) A'CD + ACD 可得CD，CD covers A'CD 與ACD

Ex 1:

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

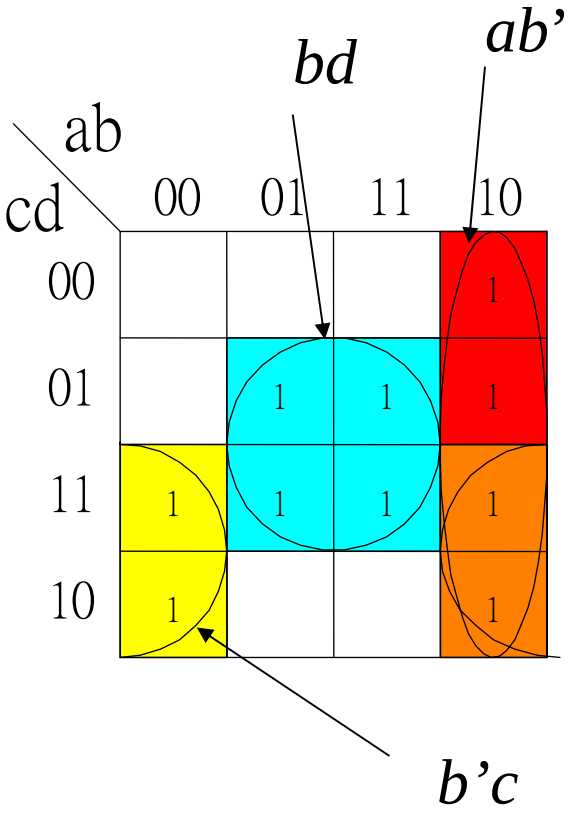
$$f_2 = \sum m(2,3,5,6,7,10,11,14,15)$$

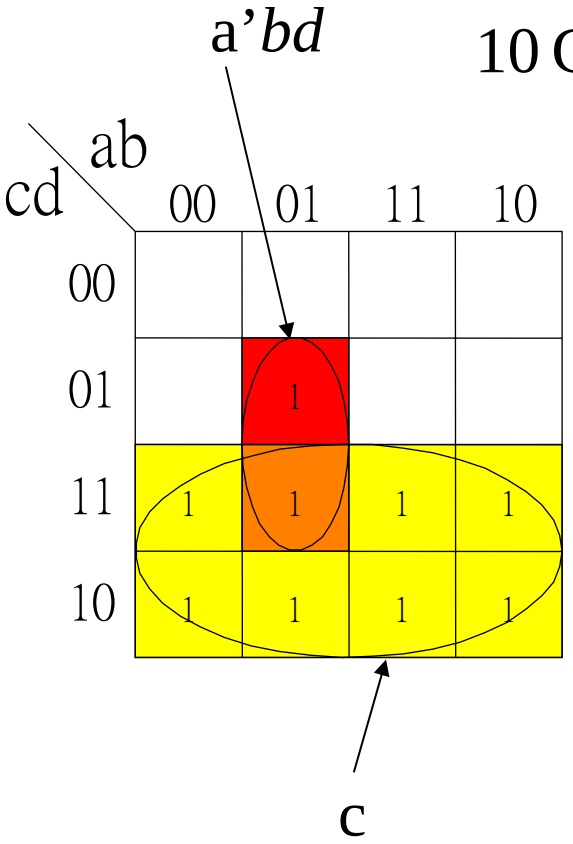
$$f_3 = \sum m(6,7,8,9,13,14,15)$$

$$f_1 = {}_1bd + {}_2b'c + {}_3ab'$$

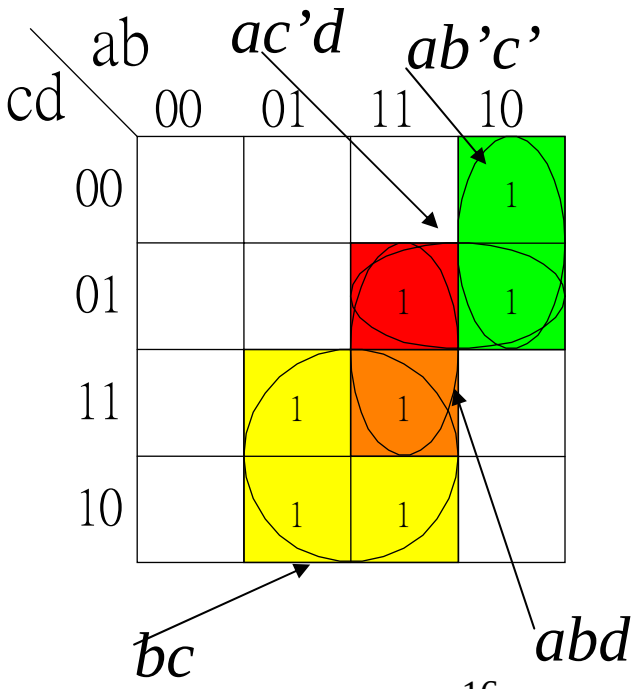
$$f_2 = {}_4a'bd + {}_5c$$

$$f_3 = {}_6bc + {}_7ab'c' + {}_8\begin{cases} ac'd \\ abd \end{cases}$$





10 Gates, 25 Gate Inputs

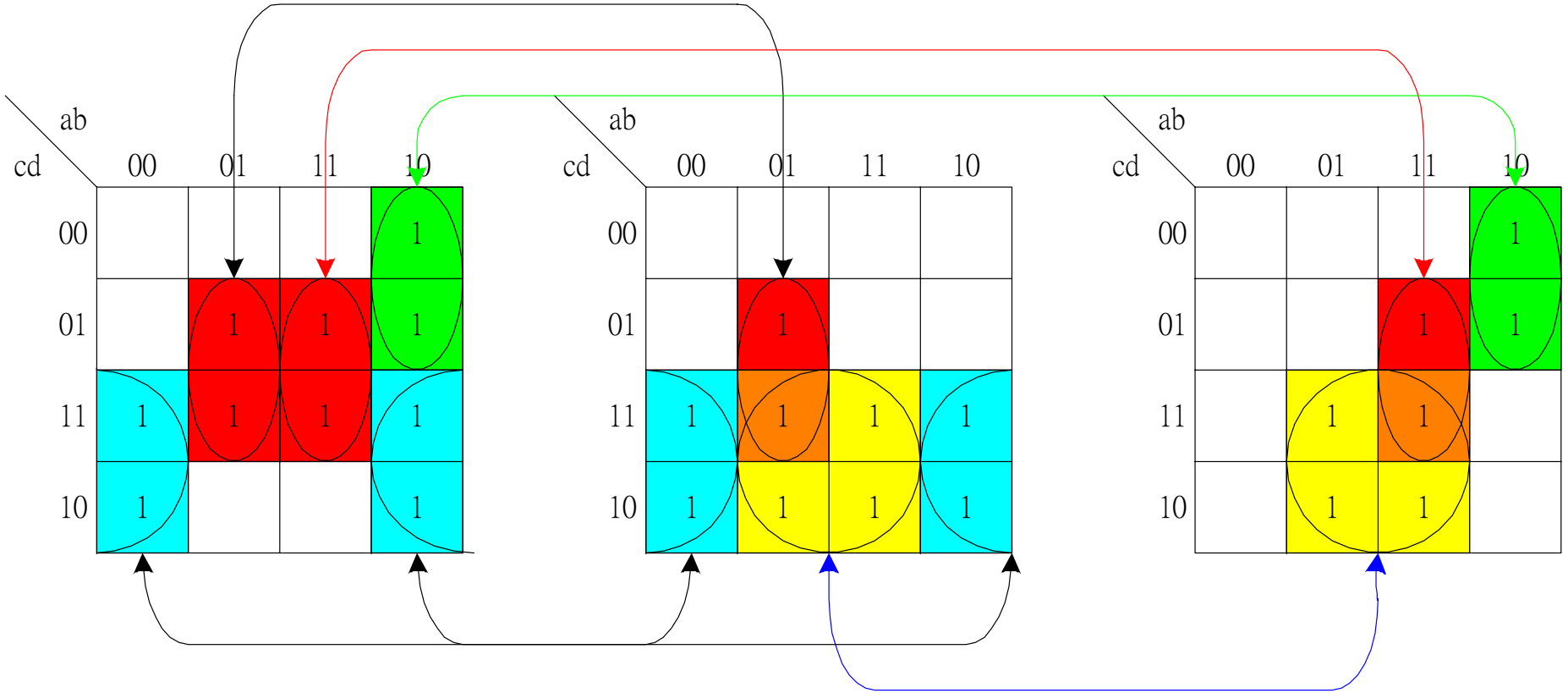


$$f_1 =_1 a'bd +_2 abd +_3 ab'c' +_4 b'c$$

$$f_2 =_1 a'bd + c$$

$$f_3 =_3 ab'c' +_2 abd + bc$$

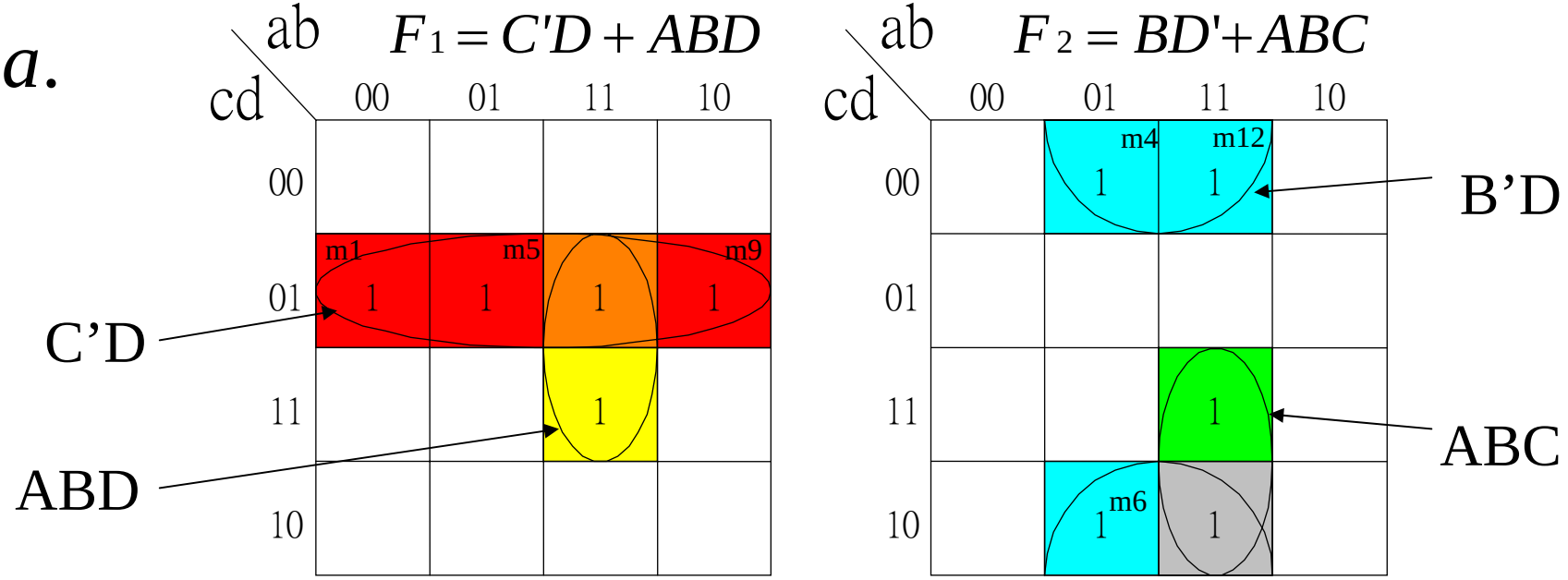
8 Gates, 22 Gate Inputs



Prime implicant 有時被拆開成 non prime implicants

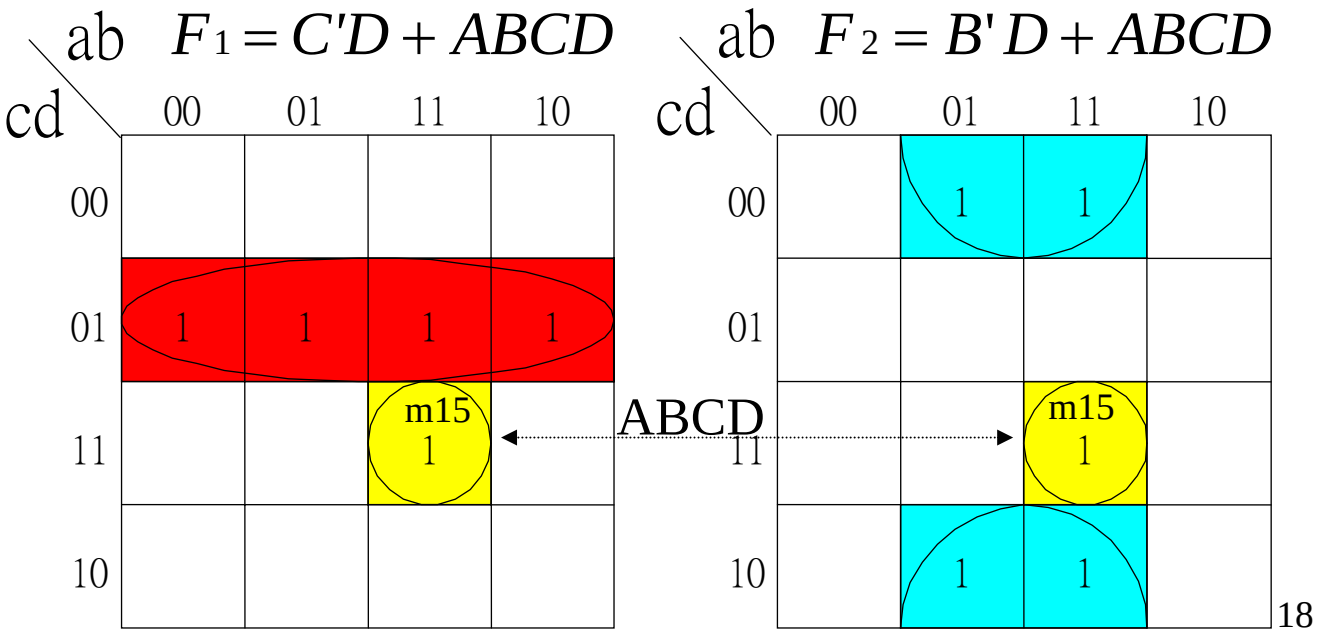
假如其他 F's有此 non prime implicants

Ex : a.



Better

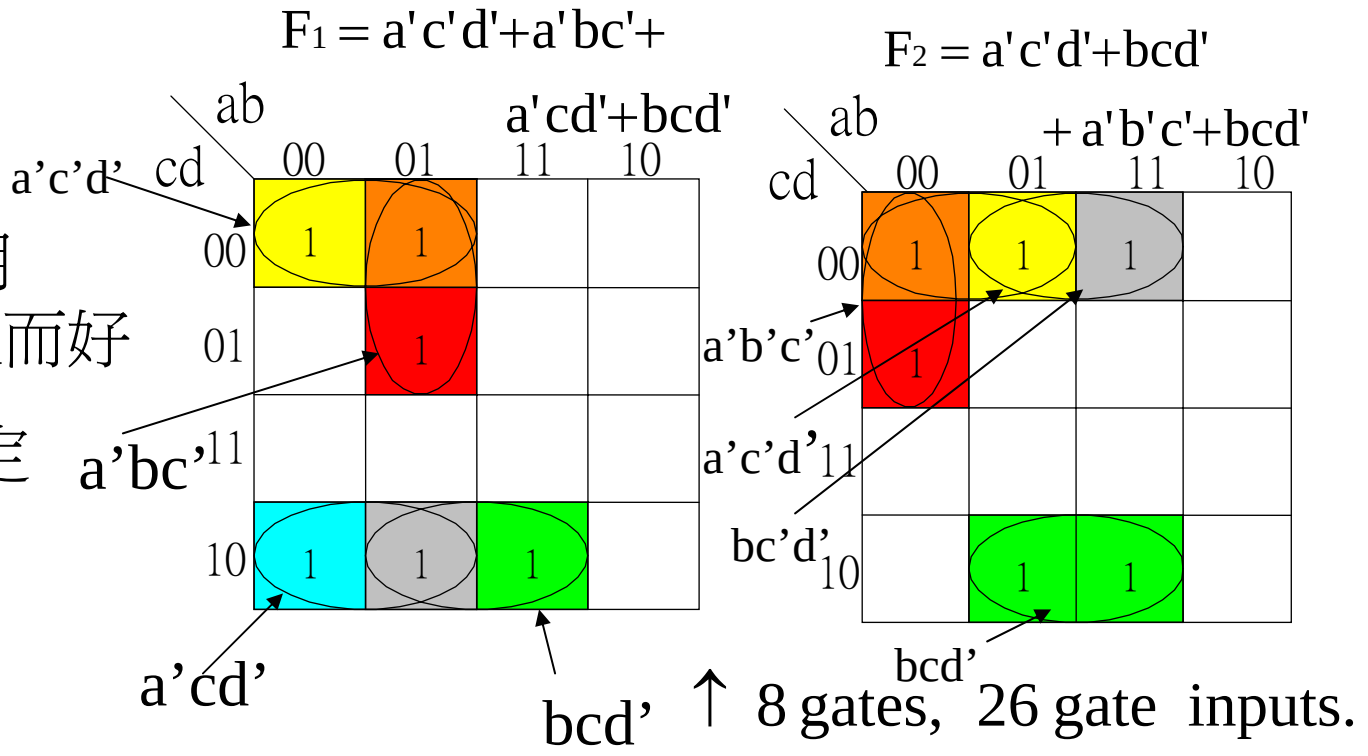
Solution



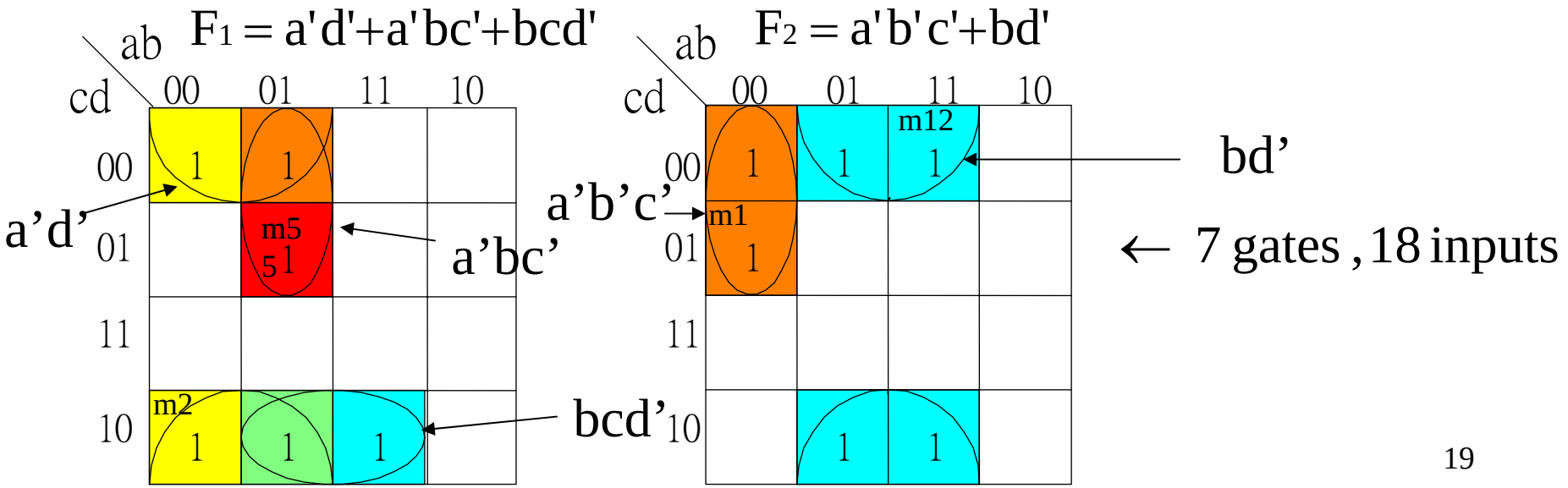
Ex : b.(反例)

如圖所示，不用
common term 反而好

→ 故視情況而定



↑ 8 gates, 26 gate inputs.



← 7 gates , 18 inputs

§ Determination of Essential Prime Implicants , for M-Output Realization

Check essential prime implicants for every function

Ex. A term in only one prime implicant, but not in other functions

In Ex a : $C'D(m_1, m_5, m_9)$, $BD'(m_4, m_6, m_{12})$ are essential,

ABD, ABC not essential.

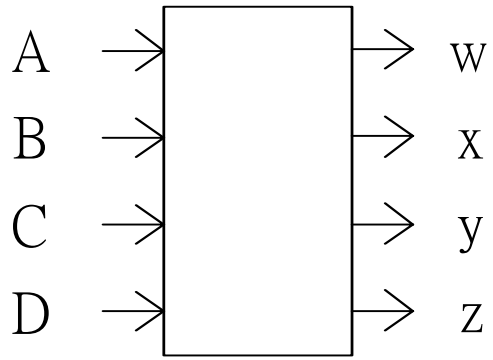
$\because m_{15}$ is in ABD of F_1, and in ABC of F_2

In Ex b : $a'd'(m_2)$, $a'bc(m_5)$, $a'b'c'(m_1)$, $bd'(m_{12})$ are essential.

Generally, essential terms 不適合部份拆開，
因其必定還是要包含此 term.

§ Design of Code Conversion Network:

Convert 8 - 4 - 2 - 1 BCD to Excess - 3 Code



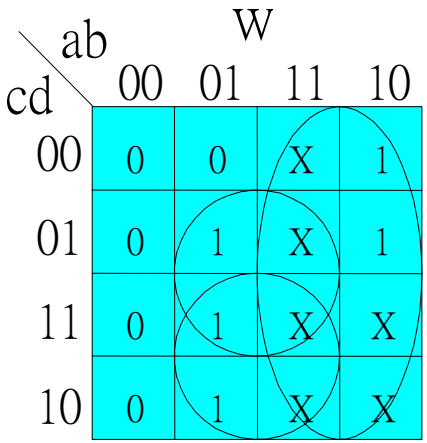
$$w = a + bc + bd$$

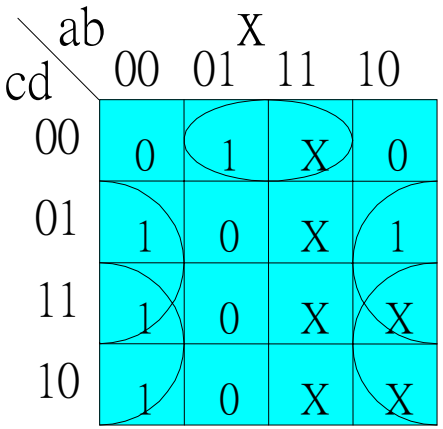
$$x = bc'd' + b'd + b'c$$

$$y = c'd' + cd$$

$$z = d'$$

a	b	c	d	w	x	y	z	
0	0	0	0	0	0	1	1	0
0	0	0	1	0	1	0	0	1
0	0	1	0	0	1	0	1	2
0	0	1	1	0	1	1	0	3
0	1	0	0	0	1	1	1	4
0	1	0	1	1	0	0	0	5
0	1	1	0	1	0	0	1	6
0	1	1	1	1	0	1	0	7
1	0	0	0	1	0	1	1	8
1	0	0	1	1	1	0	0	9
1	0	1	0	X	X	X	X	
1	0	1	1	X	X	X	X	
1	1	0	0	X	X	X	X	
1	1	0	1	X	X	X	X	
1	1	1	0	X	X	X	X	
1	1	1	1	X	X	X	X	





2-Level Network:

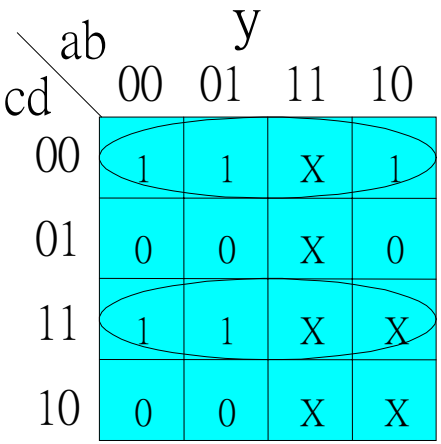
10 Gates

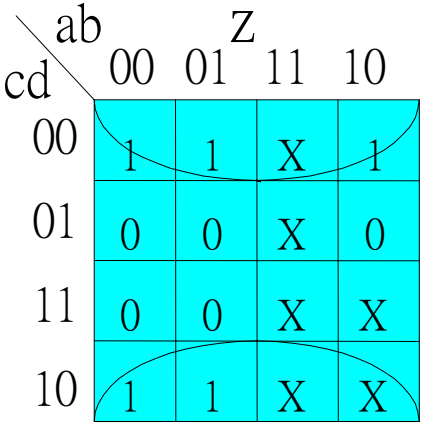
$$w = a + bc + bd$$

$$x = bc'd + b'd + b'c$$

$$y = c'd' + cd$$

$$z = d'$$





3-Level Network:

$$w = a + b(c + d)$$

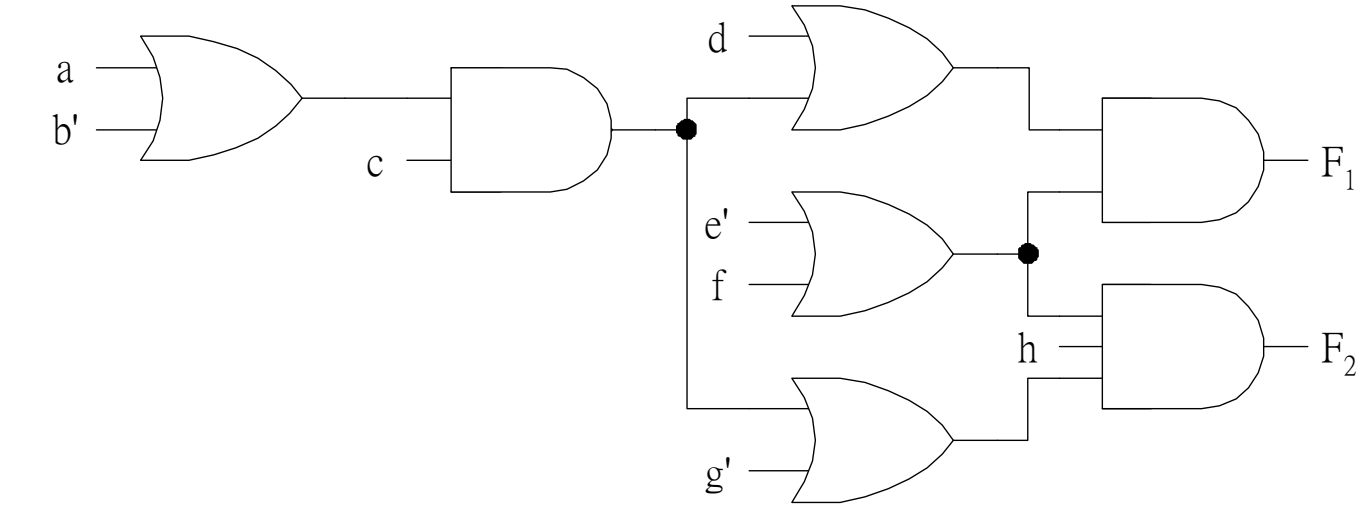
$$x = bc'd + b'(c + d)$$

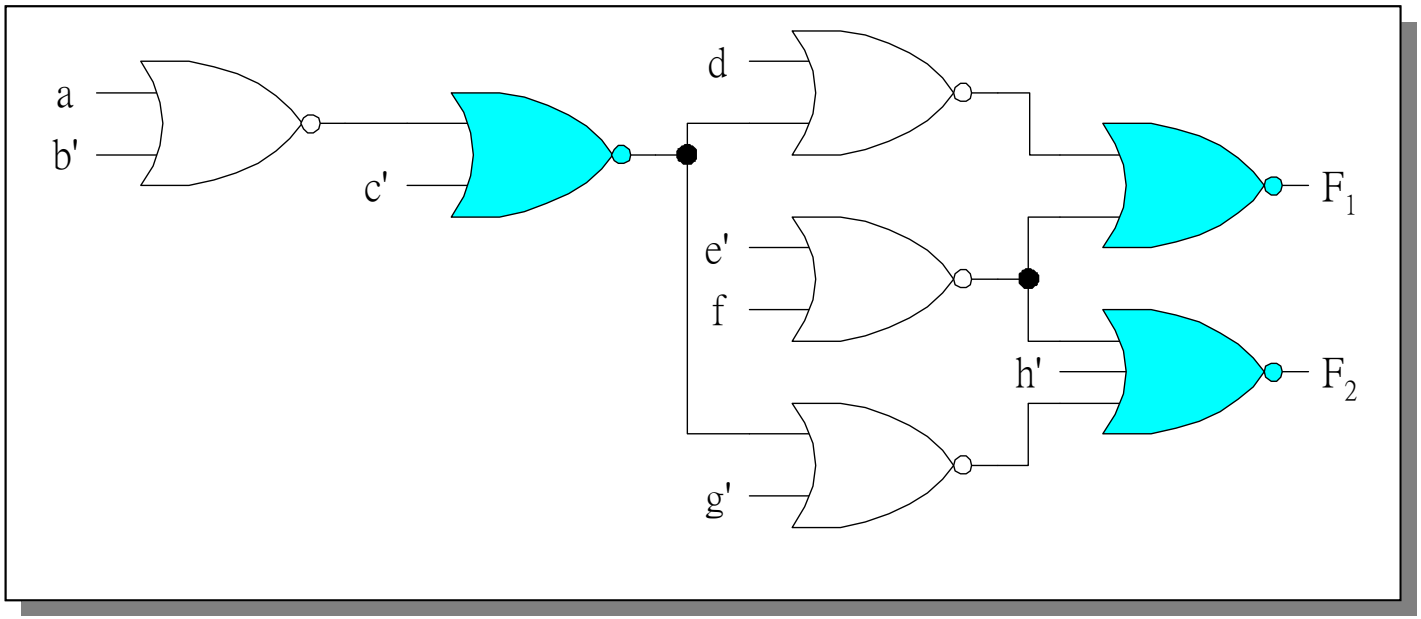
$$y = c'd' + cd$$

$$z = d'$$

9 Gates

Multi-Output NAND NOR Networks





Home Work: Unit 7

- 7-2(b)
- 7-6
- 7-8
- 7-13(a)
- 7-17
- 7-24
- 7-32
- 7-39