二、基本定理（Theorems）

定理1（等冪律）	(a)$x + x = x$ (b)$x \circ x = x$
定理2（自補性）	$(x')' = x$
定理3（結合律）	(a)$x + (y + z) = (x + y) + z$ (b)$x \circ (y \circ z) = (x \circ y) \circ z$
定理4（吸收律）	(1a) $x + x \circ y = x$ (1b) $x \circ (x + y) = x$ (2a) $x + x' \circ y = x + y$ (2b) $x \circ (x' + y) = xy$
定理5（一致性）	(a)$xy + yz + x'z = xy + x'z$ (b)$(x + y)(y + z)(x' + z) = (x + y)(x' + z)$

備註

如果要考定理的證明，可以用公理來證明推導，或者拿已證過的定理來證明推導也可以，但千萬不可拿未知的定理來證明目前將要證明的定理!

三、對偶性定理（Duality Principle）

對於上述的每一個公理或是定理而言，若是存在有(a)(b)兩部分，明顯地，我們把其中的一個式子的AND變OR，OR變AND，1變0，0變1，則我們可以得到相對應的另一部分，這就稱爲對偶性定理。

四、狄摩根定理（DeMorgan's Theorem）

狄摩根定理（DeMorgan's Theorem）有下列兩種情況：

1. 對於任意變數和的補數等於個別變數的補數之積。

$$(A + B + C + D + E + \cdots)' = (A' \circ B' \circ C' \circ D' \circ E' \circ \cdots)$$

2. 對於任意變數乘積的補數等於個別變數的補數之和。

$$(A \circ B \circ C \circ D \circ E \circ \cdots)' = (A' + B' + C' + D' + E' + \cdots)$$

3.8 卡諾圖法化簡

重要性：◈◈◈◈◇

一、卡諾圖（Karnaugh map, K-map）

卡諾圖就是將布林函數其所有的最小項，以一個二維陣列來表示，而其中的每一個格子中的最小項跟它的上、下、左、右四個鄰居只有一個位元的值不同。

1. 二變數卡諾圖

m0	m1
m2	m3

x \ y	0	1
0	x’y’	x’y
1	xy’	xy

2. 三變數卡諾圖

m0	m1	m3	m2
m4	m5	m7	m6

x \ yz	00	01	11	10
0	x’y’z’	x’y’z	x’yz	x’yz’
1	xy’z’	xy’z	xyz	xyz'

二、卡諾圖化簡步驟

利用卡諾圖來化簡布林函數步驟如下：

1. 在卡諾圖中將布林函數中是1的最小項填入1。

2. 如果有相鄰的1可以形成一個“矩形”，則將它們都框起來，記住要框最大的才行。注意！要框成2的指數次方倍才行（即2,4,8,16,...）。

3. 寫出簡化後的布林函數。

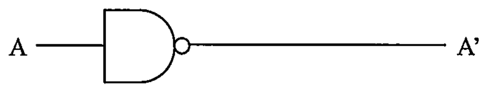
3.9 萬用閘（NAND & NOR gates） 重要性：◈◈◈◈◇

在我們前面所談的布林代數的化簡中，最後一定會成爲AND、OR及NOT三種閘的組合。現在我們介紹NAND及NOR這兩種萬用閘，萬用閘的好處，就是所有的布林函數，最後都可以用NAND或是NOR單一的閘（gate）就可以完成。

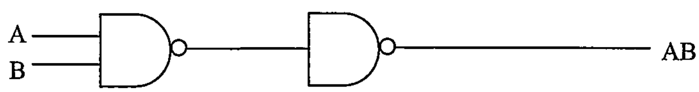
一、NAND閘

首先，我們先來看看如何利用NAND閘來取代基本的NOT閘、AND閘及OR等三類的基本邏輯閘。

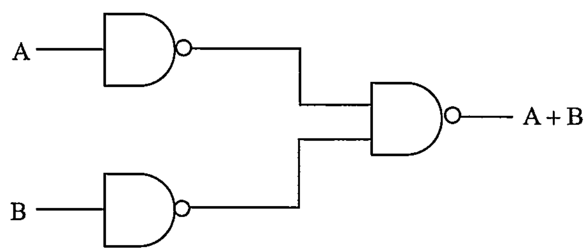
NOT閘（Inverter gate）：



AND閘：



OR閘：

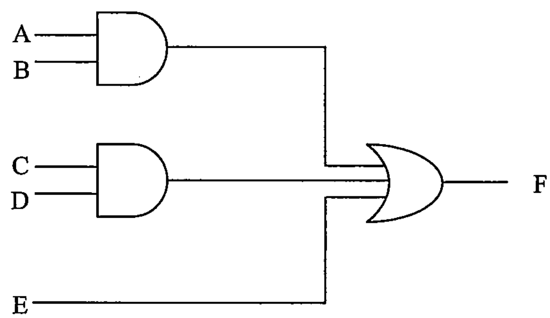


1. 積項之和轉成NAND閘

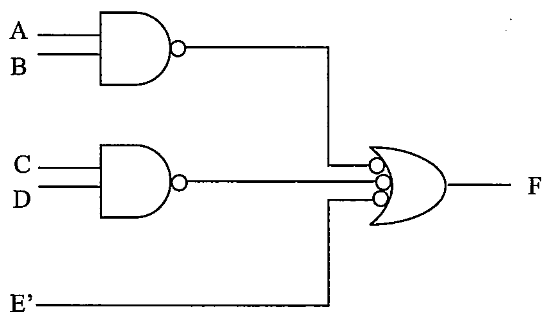
若如果題目說要把一個邏輯電路全都以NAND閘來做的話，我們只需把每一個AND閘、OR閘及NOT閘都換成上面所對應的NAND閘來表示即可。

下面我們來看一個積項之和（sum of products）是如何轉成全都是NAND閘的組合的例子。如：

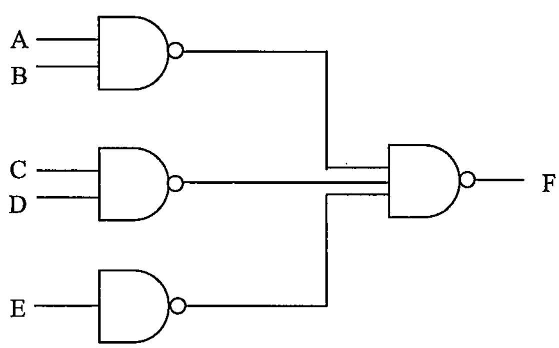
有一個布林函數：F = AB + CD + E，畫成如下圖：



明顯地，它又同等於下圖：



再將右邊OR閘的輸入爲反相的部分往右移，並利用狄摩根定理把它變成NAND閘，此外也把單一變數的輸入(E')以NAND閘來表示，就可以得到下圖。明顯地，全部都是以NAND閘來表示。



*2.*布林函數轉成NAND閘

如何將一個布林函數轉成全是NAND閘的步驟如下：

(1)簡化函數，並以積項之和來表示。

(2)對於每一個至少有兩個變數以上的項都畫成一個NAND閘，這些NAND閘的輸入就是積項的輸入。這是第一層。

(3)畫一個NAND閘在第二階層，它的輸入就是步驟2的NAND閘的輸出。

(4)對於第一層次中若是有單一變數也以NAND閘取代，並送入第二層次的NAND 閘當作輸入，即得。

備註

積項之和（sum of products）：

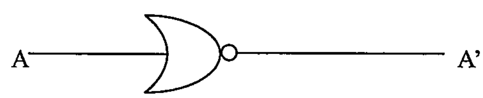
當布林函數都是利用OR把AND項串起來，而每一個AND項可以是一個以上變數所構成，此時稱為積項之和的表示法。例如：

$F_1 = y' + xy + x'yz'$

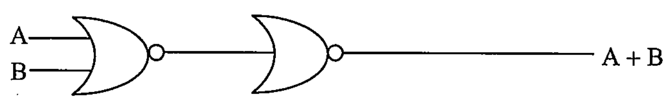
二、NOR閘

接下來，我們再看看如何利用NOR閘來實作出NOT閘、AND閘及OR閘等三類的邏輯閘，如下：

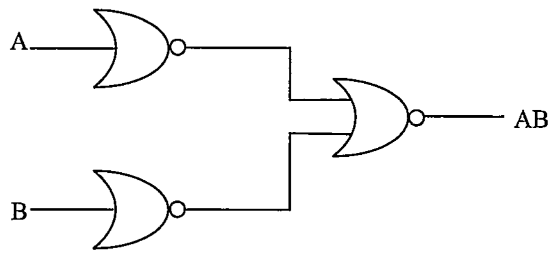
NOT閘（Inverter gate）



OR閘



AND閘

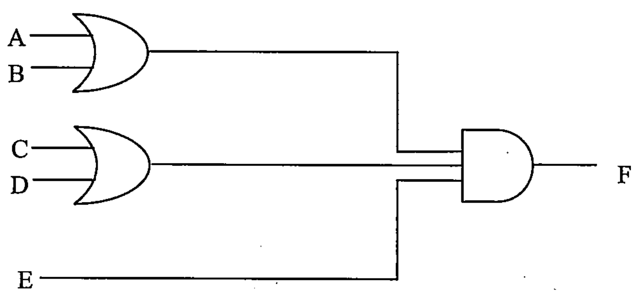


由以上的轉換，我們可以知道如果有題目說要把一個邏輯電路全以NOR閘來做，我們只需把每一個AND、OR閘與NOT閘都換成上面所對應的NOR閘來表示，最後再針對NOR閘化簡即可。

1. **將和項之積轉成全是NOR閘**

再來看和項之積（product of sums）如何轉換成全部都以NOR閘來表示。

假設$F = (A + B)(C + D)E$，則以AND與OR閘的邏輯圖如下：



明顯地，上面的邏輯圖等同於下面的邏輯圖：