邏輯運算

邏輯運算子

- 我們可以使用邏輯運算子，將兩個以上使用到比較運算子的運算式做結合。
- 常用的邏輯運算子如下：

名稱	運算子	範例	範例運算結果
AND	&&	(2>2) && (4<5)	False
OR	||	(2>2) || (4<5)	True
NOT	!	!(2 > 3)	True

邏輯運算子的真值表(Truth table)

- 以 A and B 為例
  - A有2種可能 True 或 False
  - B也有2種可能 True 或 False
  - 則 A and B 共有 4 種狀況要討論
- 你可以這樣想
  - A: 帳號正確
  - B: 密碼正確
  - A and B : 帳號正確 而且 密碼正確

A \ B	T	F
T	T	F
F	F	F

邏輯運算子的真值表(Truth table)

- 我們也常用 1 表示 True
  用0表示 False

B \ A	1	0
1	1	0
0	0	0

- 真值表也可以長這樣
  每一列是一種狀況

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

[是否需要去當兵]

if 年齡≥18 and 性別=男

	A	and	B	→	C
15，女	F		F		F
15，男	F		T		F
20，女	T		F		F
18，男	T		T		T

邏輯運算子的真值表(Truth table)

- or 的真值表

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

- not 的真值表

A	not A
0	1
1	0

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

[是否可拿到獎學金]

if 國文≥90 or 數學≥90

	A	or	B	→	C
65 , 43	F		F		F
85 , 92	F		T		T
95 , 75	T		F		T
92 , 95	T		T		T

A	not A
0	1
1	0

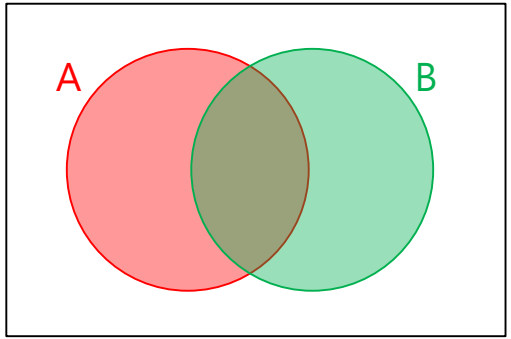
[是否通過機場安檢]

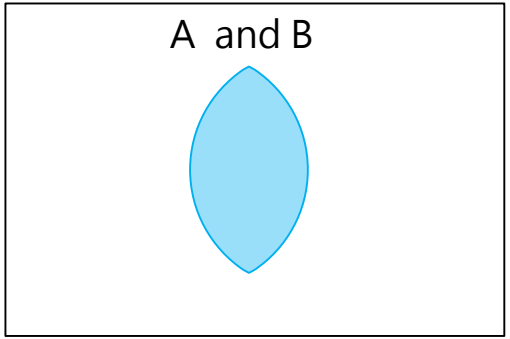
if not 金屬探測警報=響起

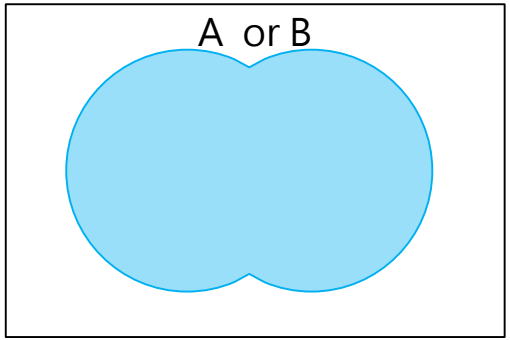
	not A	C
未響起	F	T
響起	T	F

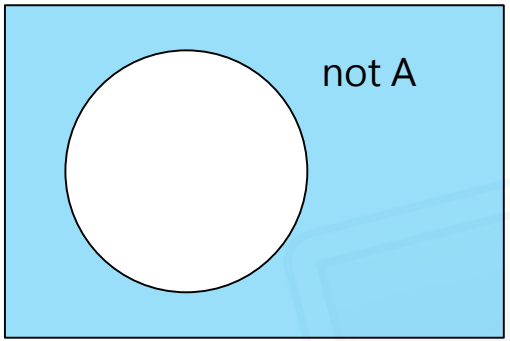
文氏圖

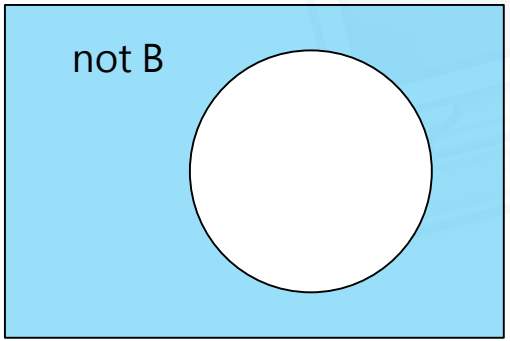
- 我們可以用文氏圖(Venn diagram)來視覺化理解 and, or, not





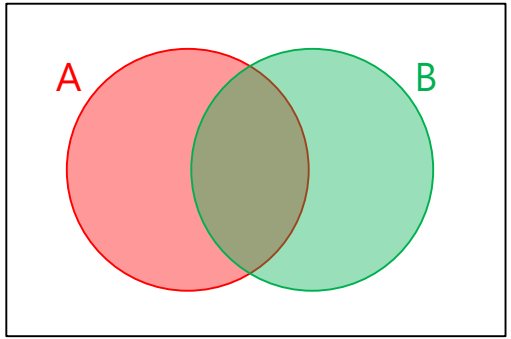


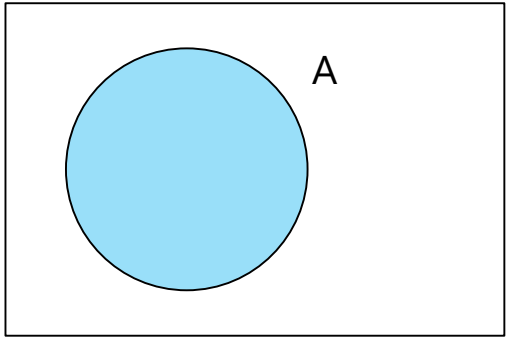


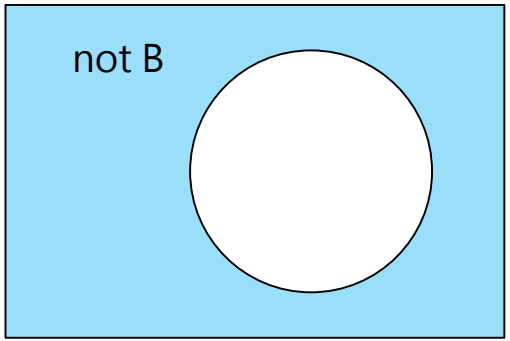


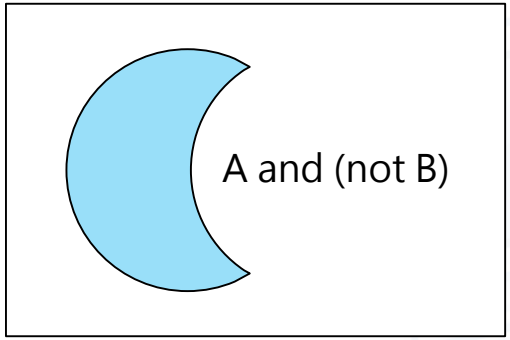
文氏圖

- A and (not B)



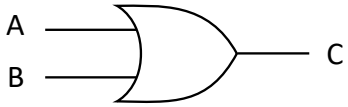






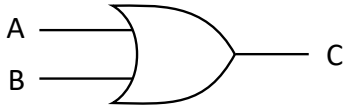
用邏輯閘來表示

- And 閘(gate)



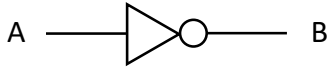
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

- Or 閘



A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

- Not 閘

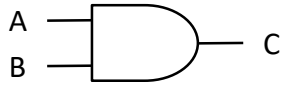


A	B
0	1
1	0

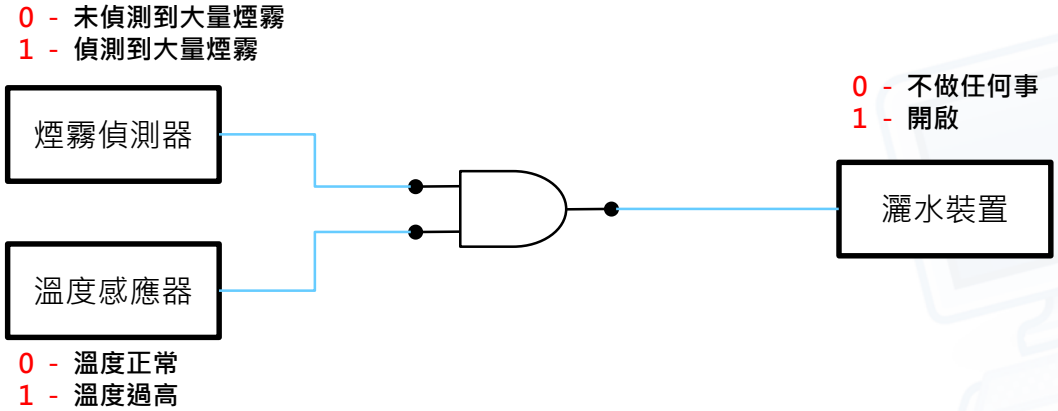
邏輯閘

- 數位電路裡常使用邏輯閘來達到控制的目的

AND 閘



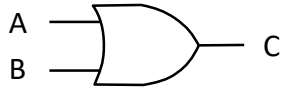
A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1



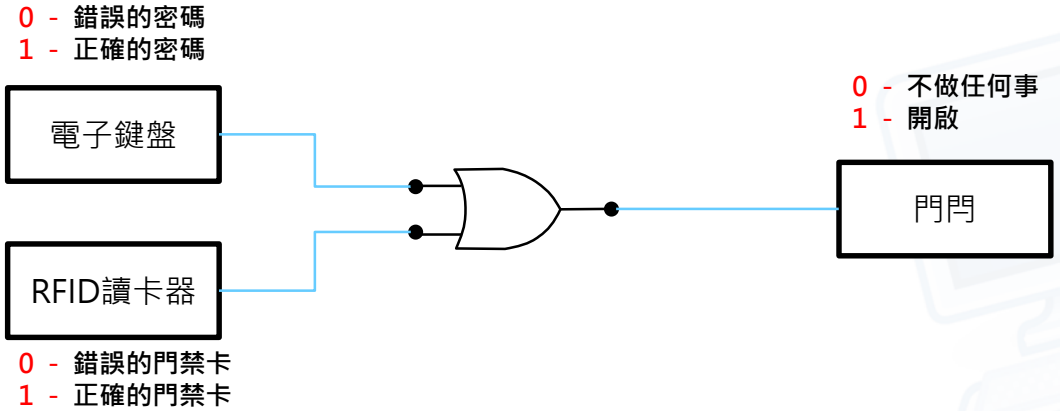
邏輯閘

- 數位電路裡常使用邏輯閘來達到控制的目的

OR 閘



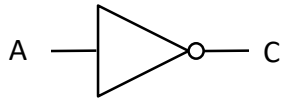
A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1



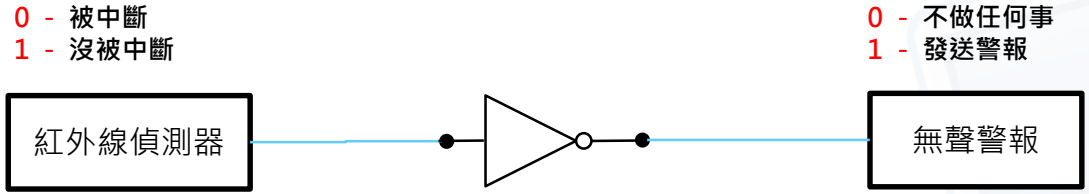
邏輯閘

- 數位電路裡常使用邏輯閘來達到控制的目的

NOT 閘

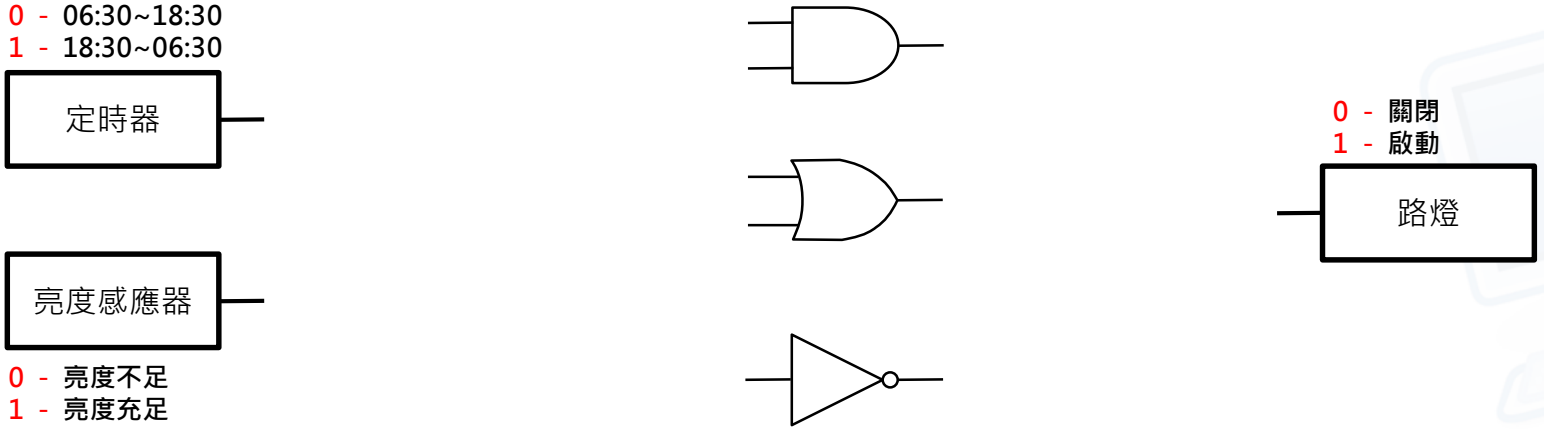


A	not A
0	1
1	0



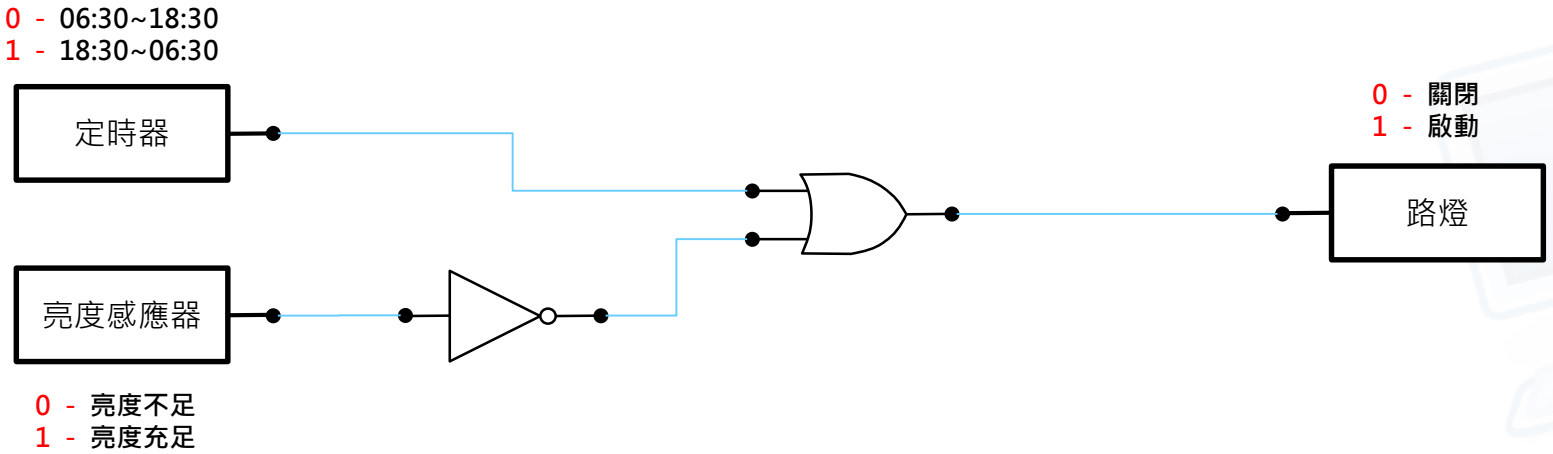
問題

- 我們有以下幾個元件
- 目的：當亮度不足或是入夜時，路燈要自動亮起。



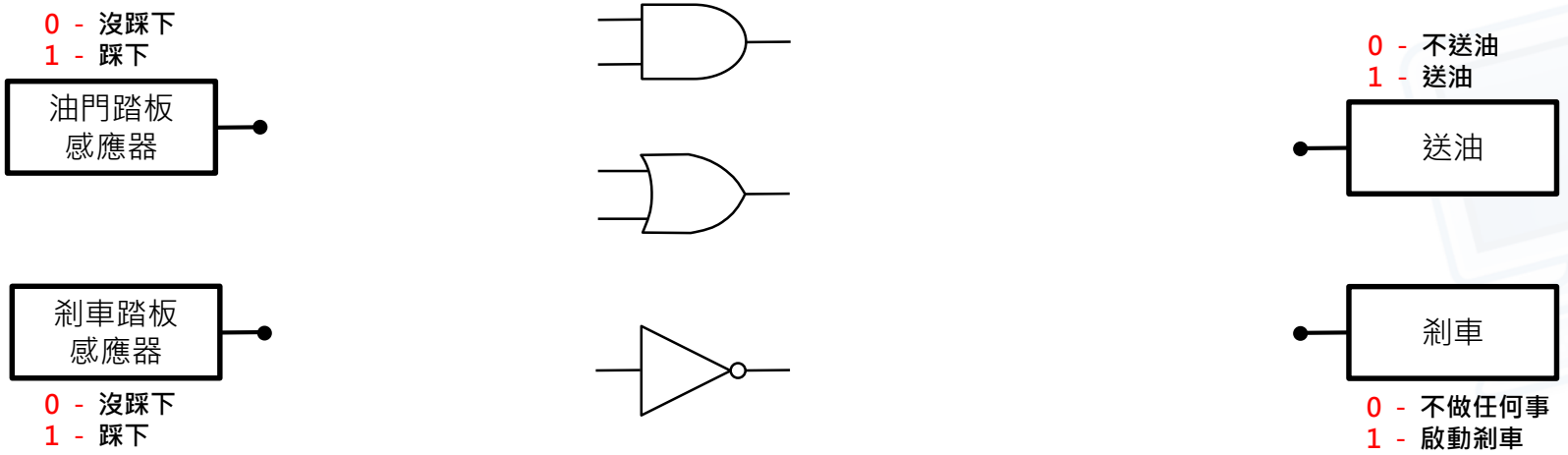
問題

- 我們有以下幾個元件
- 目的：當亮度不足或是入夜時，路燈要自動亮起。



問題

- 我們希望在駕駛人同時踩下油門和剎車踏板時，會因剎車優先設計而不送油，僅啟動剎車。



問題

- 我們希望在駕駛人同時踩下油門和剎車踏板時，會因剎車優先設計而不送油，僅啟動剎車。