

第 3 章 6：數位邏輯基礎

數位電路是現代計算的基石。本章從二進位開始，建立完整的數位邏輯框架。

一、二進位 (Binary)

為什麼用二進位？

- 電子元件最容易區分兩個狀態：「有電 vs 無電」
- 抗雜訊：只要區分 0 與 1，雜訊容忍度大
- 邏輯運算簡單

進位制

進位	字元	範例
二進位 (Binary)	0, 1	1101 ₂
八進位 (Octal)	0-7	15 ₈
十進位 (Decimal)	0-9	13
十六進位 (Hex)	0-9, A-F	0x0D

轉換

二進位 → 十進位：

$$1101_2 = 1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 = 13$$

二進位 → 十六進位：每 4 位一組：1101 = D = 0x0D

表示範圍

n 位：可表示 2^n 個值

位數	範圍
8 bit	0-255 (無號) / -128 到 127 (有號)
16 bit	0-65535 / -32768 到 32767
32 bit	0 - 4.29×10^9
64 bit	0 - 1.84×10^{19}

帶號表示

- 符號-絕對值：MSB 是符號（少用）
- 2 的補數 (Two's Complement)：主流
- 正數：直接二進位
- 負數：反相 + 1
- 範例：-5 (8-bit) = $\sim 00000101 + 1 = 11111011$

二、邏輯閘 (Logic Gates)

基本閘

閘	符號	真值表	布林式
NOT (反相)		$0 \rightarrow 1, 1 \rightarrow 0$	$Y = \neg A$
AND		$A \cdot B$	$Y = AB$
OR		$A + B$	$Y = A + B$
NAND		$\neg(A \cdot B)$	$Y = \neg(AB)$
NOR		$\neg(A + B)$	$Y = \neg(A + B)$
XOR		不同為 1	$Y = A \oplus B$
XNOR		相同為 1	$Y = \neg(A \oplus B)$

真值表 (Truth Table)

A	B	AND	OR	NAND	XOR
0	0	0	0	1	0
0	1	0	1	1	1
1	0	0	1	1	1
1	1	1	1	0	0

通用閘

NAND 或 NOR 都可以實現所有邏輯，稱為「通用閘」。所以早期 IC 大量是 NAND (74LS00)。

三、布林代數 (Boolean Algebra)

基本定律

定律	表達式
交換律	$A + B = B + A$; $A \cdot B = B \cdot A$
結合律	$(A+B)+C = A+(B+C)$
分配律	$A \cdot (B+C) = AB + AC$
恆等律	$A + 0 = A$; $A \cdot 1 = A$
互補律	$A + !A = 1$; $A \cdot !A = 0$
吸收律	$A + AB = A$; $A(A+B) = A$

笛摩根定律 (DeMorgan's Theorems)

$$!(A \cdot B) = !A + !B$$

$$!(A + B) = !A \cdot !B$$

反相分配進去，AND/OR 交換。

化簡實例

化簡： $F = A \cdot !B + A \cdot B + !A \cdot B$

$$F = A(!B + B) + !AB = A + !AB = A + B$$

四、卡諾圖 (Karnaugh Map)

化簡布林函數的視覺工具。

2 變數 K-Map

	B=0	B=1
A=0	[]	[]
A=1	[]	[]

3 變數

	BC=00	BC=01	BC=11	BC=10
A=0	[]	[]	[]	[]
A=1	[]	[]	[]	[]

注意：BC 的順序是 00, 01, 11, 10 (Gray Code，相鄰只差 1 位)。

4 變數

	CD=00	CD=01	CD=11	CD=10
AB=00	[]	[]	[]	[]
AB=01	[]	[]	[]	[]
AB=11	[]	[]	[]	[]
AB=10	[]	[]	[]	[]

化簡規則

把相鄰的 1 圈起來 (2、4、8 個一組)。組數越大，化簡越多。邊緣繞回 (4 個角是相鄰)。

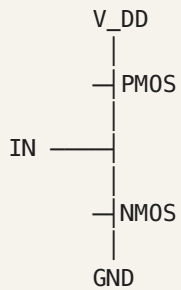
範例

$$F(A,B,C,D) = \Sigma(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$$

填表後可化簡為： $F = !B \cdot !C + !A \cdot D' + \dots$ (略)

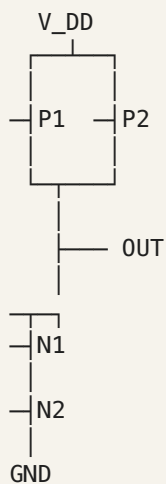
五、CMOS 數位邏輯

CMOS 反相器



$A=0 \rightarrow \text{PMOS 開、NMOS 關} \rightarrow \text{OUT} = V_{DD} = 1$ $A=1 \rightarrow \text{PMOS 關、NMOS 開} \rightarrow \text{OUT} = 0$

CMOS NAND



$A \rightarrow P1, N1$
 $B \rightarrow P2, N2$

當 $A=B=1$ 時兩 NMOS 都導通 $\rightarrow \text{OUT} = 0$ 其他情況： $\text{OUT} = 1$

$\rightarrow \text{NAND 邏輯。}$

數位電源規範

邏輯族	V _{DD}	邏輯 H 範圍	邏輯 L 範圍
5V CMOS	5V	> 3.5V	< 1.5V
3.3V CMOS	3.3V	> 2V	< 0.8V
1.8V CMOS	1.8V	> 1.2V	< 0.6V
LVDS	差動 350 mV	—	—

雜訊邊際 (Noise Margin)

訊號被視為 H/L 的「緩衝區」。

$$NMH = V_{OH(min)} - V_{IH(min)} \quad NML = V_{IL(max)} - V_{OL(max)}$$

雜訊邊際大 → 抗雜訊強。

六、邏輯族 (Logic Families)

TTL (Transistor-Transistor Logic)

老技術，BJT 為主：- 7400 系列（標準）：1970s - 74LS (Low Power Schottky)：低耗 - 74AS、74F（更快）

CMOS

- 4000 系列（早期 CMOS，慢、高雜訊邊際）
- 74HC (High-Speed CMOS)：替代 LS
- 74AHC、74LVC、74AVC（更快、低電壓）

其他

- ECL (Emitter Coupled Logic)：超高速但耗能
- LVDS (Low Voltage Differential Signaling)：差動，現代高速
- LVCMOS、LVTTTL：低電壓 CMOS

電平轉換 (Level Shifter)

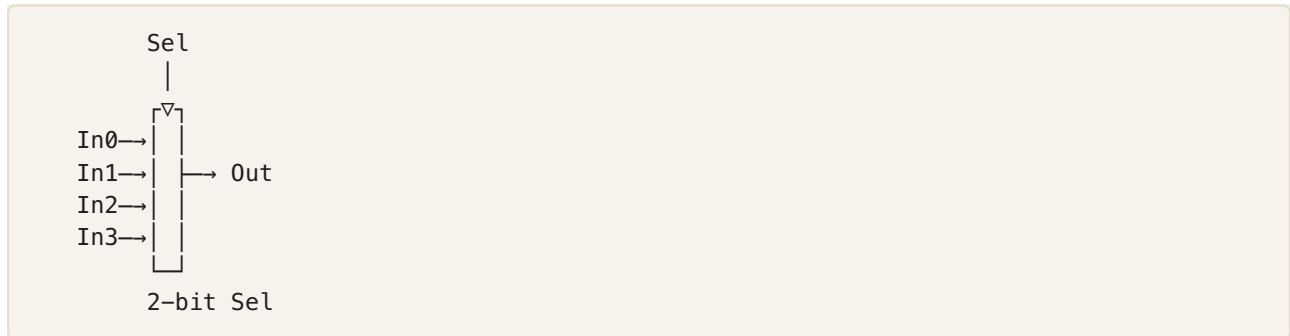
不同電壓系統間需要轉換：- 雙向：BSS138、TXS0108E - 單向：74LVC1T45 - I²C 專用：PCA9306

七、組合邏輯 (Combinational Logic)

輸出只依當前輸入（無記憶）。

1. 多工器 (Multiplexer, MUX)

選擇多個輸入中的一個：



n 個 MUX : $\log_2(n)$ 位選擇線。

2. 解多工器 (Demultiplexer)

反向：1 個輸入分配到多個輸出。

3. 解碼器 (Decoder)

n 個輸入 $\rightarrow 2^n$ 個輸出（每次只激活一個）。範例：3-to-8 decoder (74HC138)

4. 編碼器 (Encoder)

反向。優先編碼器 (priority encoder) 處理多輸入同時激活。

5. 加法器 (Adder)

半加法器 (Half Adder)

$$A + B \rightarrow S, C_{out} \quad S = A \oplus B \quad C_{out} = A \cdot B$$

全加法器 (Full Adder)

$$A + B + C_{in} \rightarrow S, C_{out} \quad S = A \oplus B \oplus C_{in} \quad C_{out} = A \cdot B + C_{in} \cdot (A \oplus B)$$

多位元加法器

8 個 FA 串聯 $\rightarrow$ 8-bit ripple carry adder (RCA)。但 carry 傳遞慢，現代用 carry-look-ahead (CLA) 加速。

6. 比較器

$A > B$, $A = B$, $A < B$ 三個輸出。

八、組合邏輯設計流程

1. 寫真值表

依需求列出所有輸入組合與期望輸出。

2. 化簡 (K-Map 或布林代數)

3. 畫邏輯圖

4. 用 IC 實現

74xx 系列、CPLD、FPGA。

範例：4-to-1 MUX

Sel1 Sel0	Out
0 0	I0
0 1	I1
1 0	I2
1 1	I3

布林：

$$\text{Out} = !S_1!S_0I_0 + !S_1S_0I_1 + S_1!S_0I_2 + S_1S_0I_3$$

可用一系列 AND-OR 實現，或用 74HC153 IC。

九、本章總結

✅ 您應該掌握：

- [x] 二進位、十六進位、轉換
- [x] 7 個基本邏輯閘

- [x] 布林代數定律與化簡
- [x] 卡諾圖
- [x] CMOS 邏輯實現
- [x] 邏輯族與電平轉換
- [x] 組合邏輯：MUX、Decoder、Adder

下一章

07-?????.md 一組合邏輯深入。