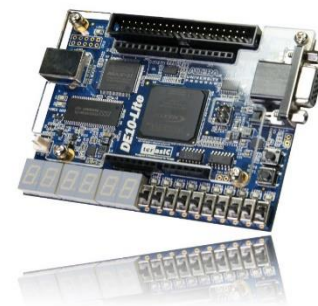


# 算術邏輯電路設計

# Outline

- 加法器
  - 半加器
  - 全加器
  - 並列加法器
- 減法器
  - 半減器
  - 全減器
- 1's補數減法電路
- 2's補數加減法電路
- BCD加法器

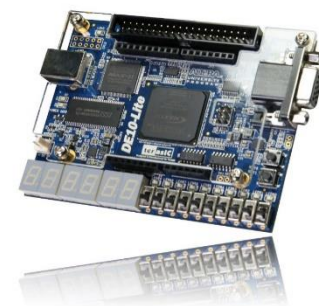


# 加法器



# 加法器

- 加法器 (adder) 是數位計算機最基本的算術運算電路，包括：
  - 半加器  
只能處理2 個位元相加的加法電路，我們稱為半加器 (Half-Adder, HA)。
  - 全加器  
能處理3 個位元相加的加法電路，我們稱為全加器 (Full-Adder, FA)。



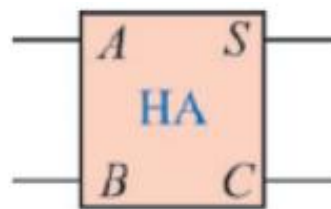
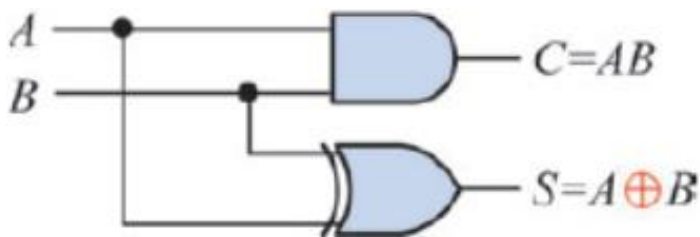
# 半加器

- 設有兩個1 位元的數字分別為A 與B，而A + B 結果為和（S）與進位（C），則其真值表如下圖所示；由真值表我們可得其和（S）與進位（C）的布林代數式為

輸入		輸出	
A	B	和	進位
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S = \overline{A}B + A\overline{B} = A \oplus B$$

$$C = AB$$



# 全加器(1/3)

- 加法運算無可避免的**一定要考慮前一級的進位輸出**，因此想要進行二進位加法運算，**一定要有能同時處理3 個位元相加的全加器才行**
- 設全加器的3 個輸入位元分別為A、B與進位輸入  $C_i$ ，相加結果為和 (S) 與進位輸出  $C_o$ ，則真值表如下圖所示；其輸出之布林代數為：

$C_i$	A	B	S	$C_{i+1}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = \overline{C_i}\overline{A}B + \overline{C_i}A\overline{B} + C_i\overline{A}\overline{B} + C_iAB$$

$$C_{i+1} = \overline{C_i}AB + C_i\overline{A}B + C_i\overline{A}\overline{B} + C_iAB$$

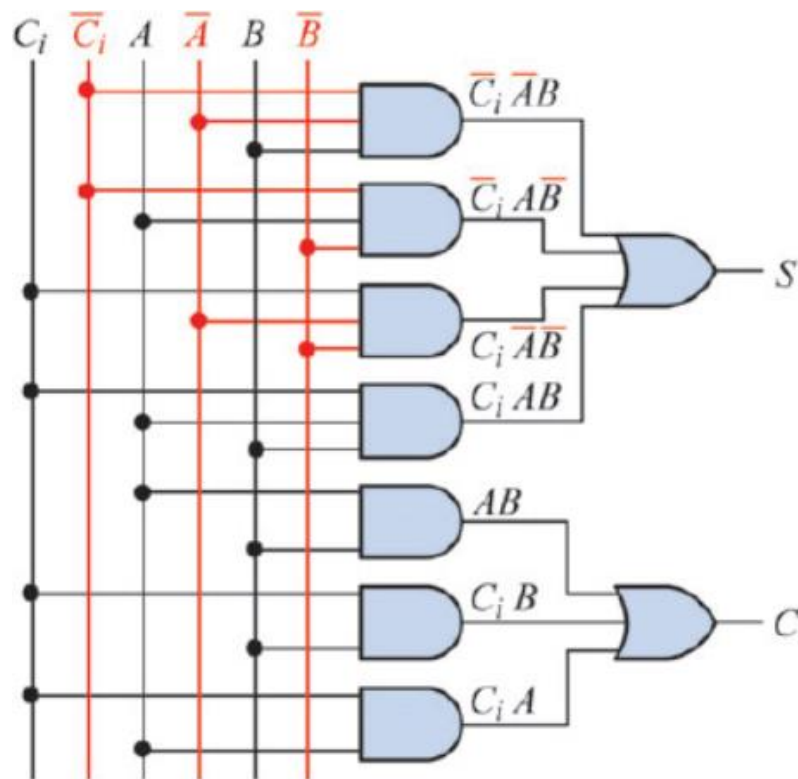
$$= \overline{C_i}AB + C_i\overline{A}B + C_i\overline{A}\overline{B} + C_iAB + C_i\overline{A}\overline{B} + C_iAB$$

$$= AB(\overline{C_i} + C_i) + C_iB(\overline{A} + A) + C_iA(\overline{B} + B)$$

$$= AB + C_iB + C_iA$$



# 全加器(2/3)



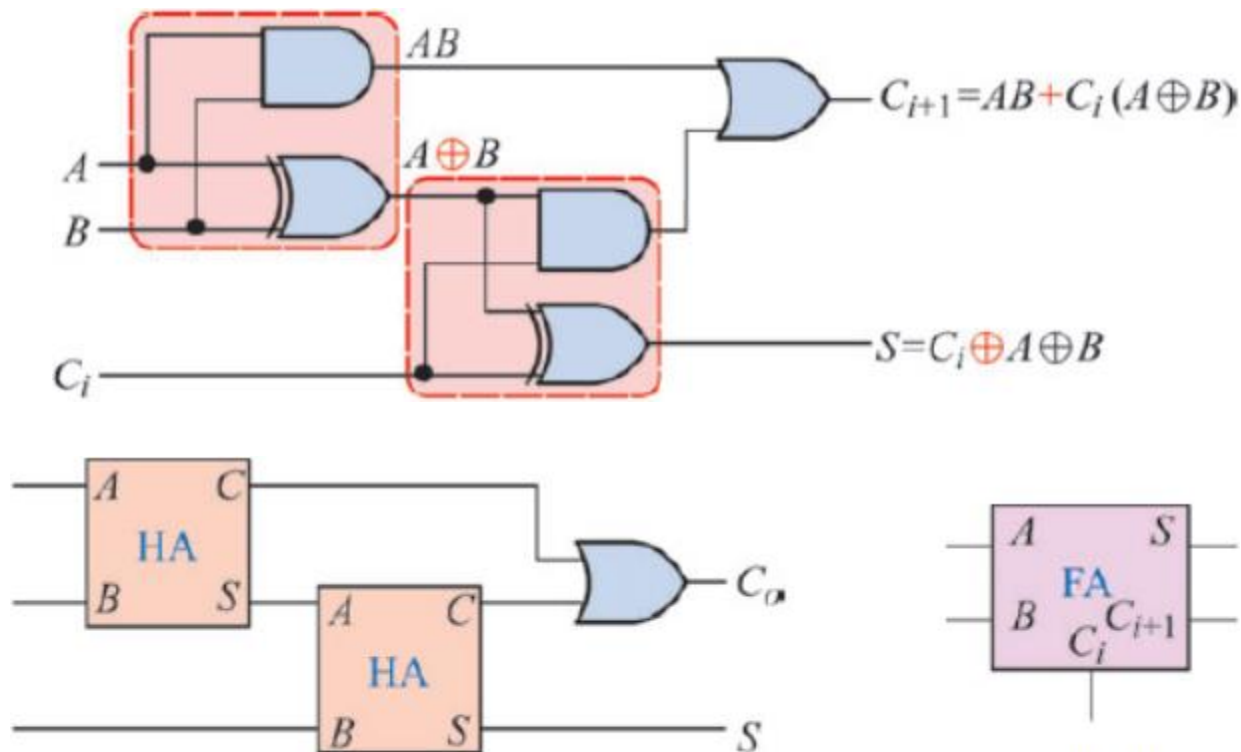
$$\begin{aligned}
 S &= \overline{C_i} \overline{A} B + \overline{C_i} A \overline{B} + C_i \overline{A} \overline{B} + C_i A B \\
 &= \overline{C_i} (\overline{A} B + A \overline{B}) + C_i (\overline{A} \overline{B} + A B) \\
 &= \overline{C_i} (A \oplus B) + C_i (\overline{A \oplus B}) \\
 &= C_i \oplus A \oplus B
 \end{aligned}$$

$$\begin{aligned}
 C_{i+1} &= \overline{C_i} A B + C_i \overline{A} B + C_i A \overline{B} + C_i A B \\
 &= A B (\overline{C_i} + C_i) + C_i (\overline{A} B + A \overline{B}) \\
 &= A B + C_i (A \oplus B)
 \end{aligned}$$

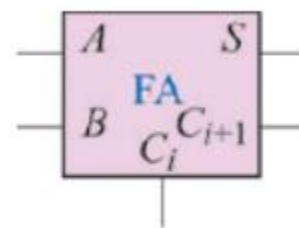


# 全加器 (3/3)

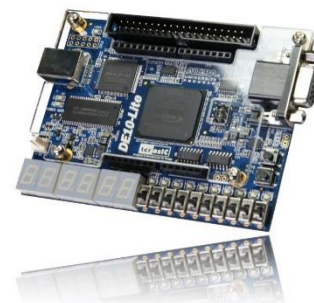
- 經整合得全加器的電路下圖所示；若將其與半加器電路來比較，**可知全加器可以由兩組半加器加上一只或閘來組成。**



半加器組成全加器



方塊圖

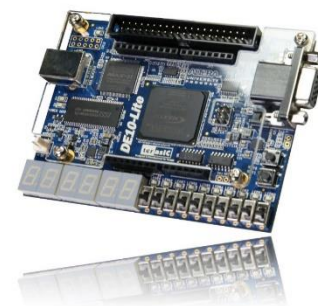




# 並列加法器(1/3)

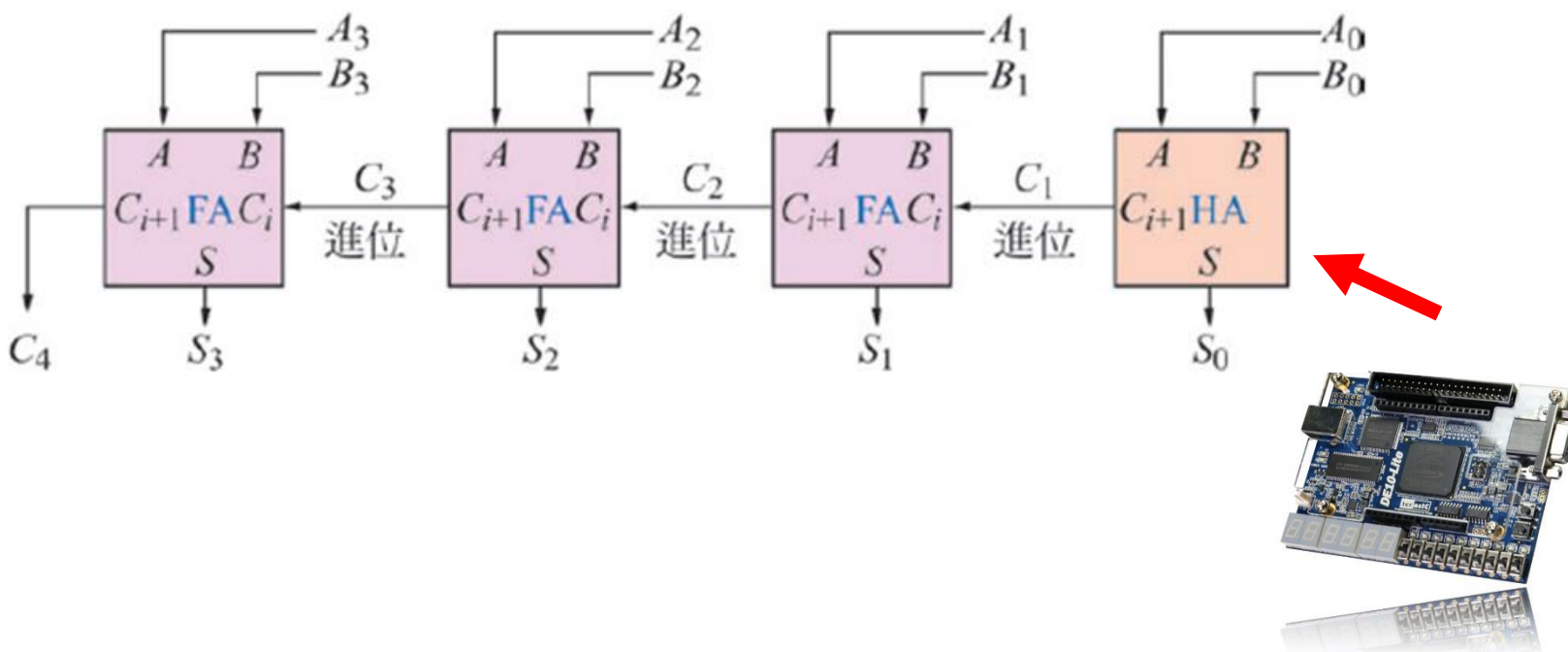
- 加法運算的操作，都是由最低位元逐級往上的，例如有兩筆二進位數 $A_3 A_2 A_1 A_0$  與  $B_3 B_2 B_1 B_0$  相加。
- 則其運算順序由最低位元 $A_0 + B_0$  開始，在 $A_0 + B_0$  後可得和 $S_0$  與進位 $C_1$ ，接下來再由 $A_1 + B_1 + C_1$  得到和 $S_1$  與進位 $C_2$  等依序進行

		$A_3$	$A_2$	$A_1$	$A_0$	← 被加數
		$B_3$	$B_2$	$B_1$	$B_0$	← 加 數
+	$C_4$	$C_3$	$C_2$	$C_1$		← 進 位
<hr/>						
	$C_4$	$S_3$	$S_2$	$S_1$	$S_0$	← 和



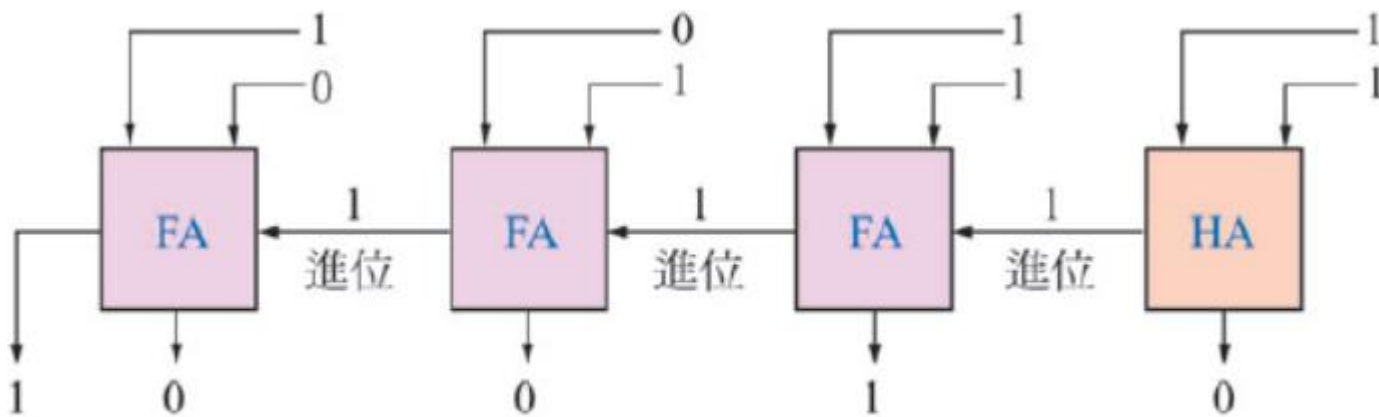
# 並列加法器(2/3)

- 只要將全加器下圖般並列串接起來，就可進行兩個多位元二進位數的加法運算了。這種可以同時將2個多位元二進位數直接相加的加法器，稱為並列加法器（parallel binary adder）。



# 並列加法器(3/3)

- 假設有2 個十進位數11 和7 要相加，由於11 之二進位數為1011B，而7 之二進位數為0111B，故最後結果為10010B 即十進制的18，如下圖所示



4位二進位數加法範例

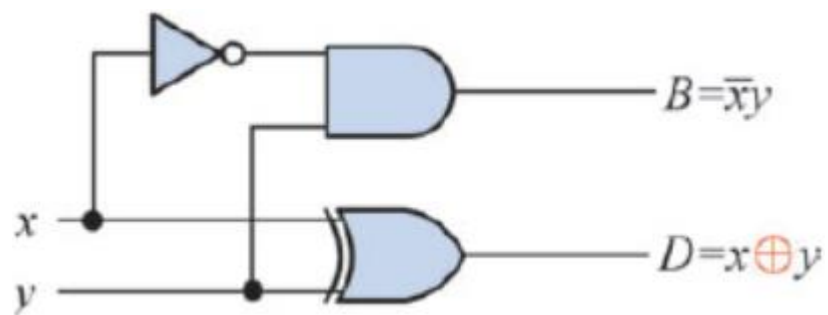


# 減法器



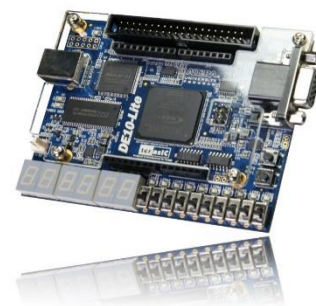
# 半減器

$x$	$y$	差 ( $D$ )	借位 ( $B$ )
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



差  $D = \bar{x}y + x\bar{y} = x \oplus y$

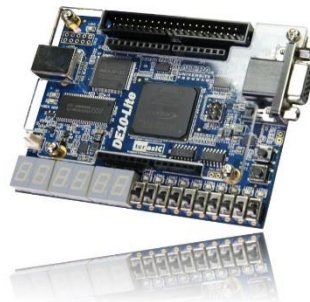
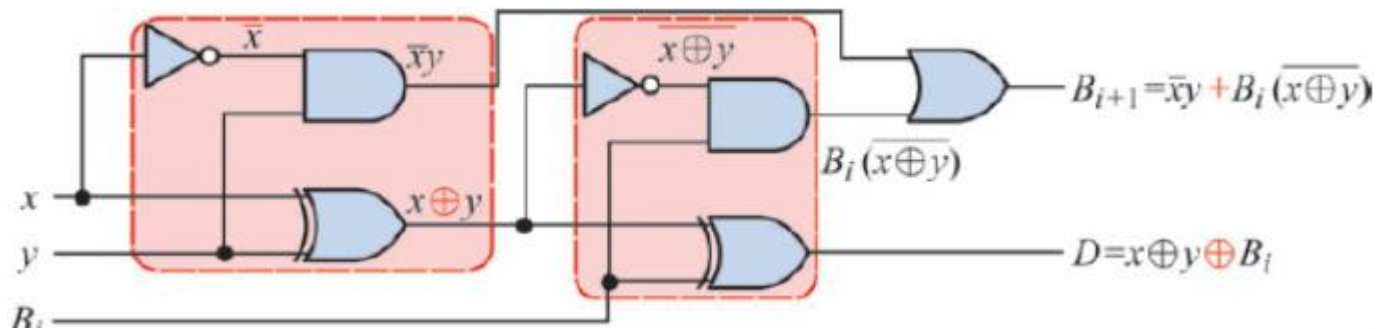
借位  $B = \bar{x}y$



# 全減器

- 它跟全加器與半加器關係相同，仍是可由兩個半減器和一個或閘來組成。

$$\begin{aligned}\text{差}(D) &= \bar{x}\bar{y}B_i + \bar{x}y\bar{B}_i + x\bar{y}\bar{B}_i + xyB_i & \text{借位}(B_{i+1}) &= \bar{x}\bar{y}B_i + \bar{x}y\bar{B}_i + \bar{x}yB_i + xyB_i \\ &= \bar{B}_i(\bar{x}y + yx) + B_i(\bar{x}\bar{y} + xy) & &= B_i(\bar{x}\bar{y} + xy) + \bar{x}y(\bar{B}_i + B_i) \\ &= \bar{B}_i(x \oplus y) + B_i(\overline{x \oplus y}) & &= B_i(\overline{x \oplus y}) + \bar{x}y \\ &= B_i \oplus x \oplus y\end{aligned}$$



# 1's補數減法電路



# 1's補數減法電路(1/2)

1補數

最高位有進位 (表示結果為正) → 端迴處理 (把進位的1加到LSB)

最高位無進位 (表示結果為負)

└→ 此時的負數值為1's，要轉回10進制時，再做一次1's，並且加上「-」號

1's範例：35 - 21

35(8位元表示)：0010 0011

21(8位元表示)：0001 0101

$$\begin{array}{r} 0010\ 0011 \\ +) 1110\ 1010 \\ \hline 1\ 0000\ 1101 \\ \hline 0000\ 1110 \end{array}$$

Diagram showing the 1's complement subtraction of 21 from 35. The carry-out of 1 from the MSB is added back to the LSB of the intermediate result (10000 1101) to produce the final result 0000 1110.

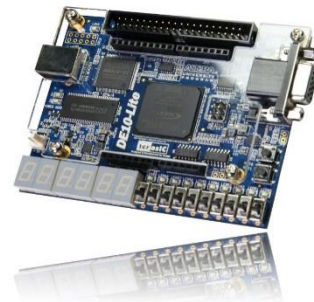
即為正解

1's範例：21 - 35

$$\begin{array}{r} 0001\ 0101 \\ +) 1101\ 1100 \\ \hline 1111\ 0001 \\ - 0000\ 1110 \\ \hline -\ 10 \end{array}$$

因為是負數，要變為原本的10進制  
要再做一遍1's轉換

後面為一般二進位轉換，前面加負號即可

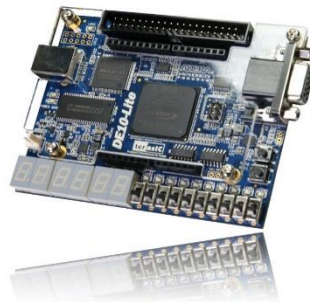
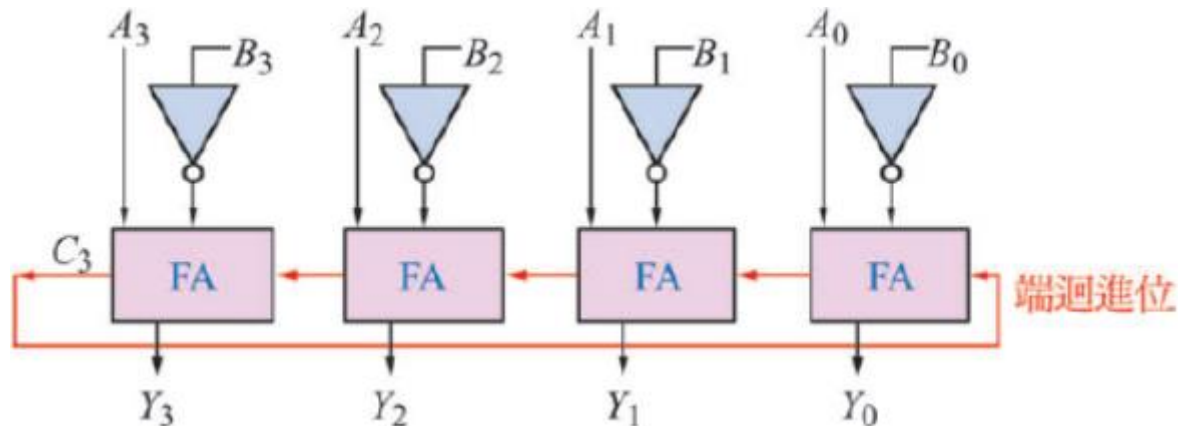




# 1's補數減法電路(1/2)

ex. 將0111減去0011，則 $A_3A_2A_1A_0=0111$ ， $B_3B_2B_1B_0=0011$ ，步驟如下：

1. 將減數 $B_3B_2B_1B_0$ 取補數， $\overline{B_3}\overline{B_2}\overline{B_1}\overline{B_0}=1100$ 。
2. 經加法器相加，得C3  $Y_3Y_2Y_1Y_0=10011$ 。
3. 將端迴進位加入，即 $Y_3Y_2Y_1Y_0 + C_3 = 0100$ 。



# 2's補數減法電路



# 2's補數減法電路(1/2)

**2's補數**

- 最高位(MSB)有進位 (表示結果為正) → 直接捨棄，即為正確結果
- 最高位(MSB)無進位 (表示結果為負)  
→ 此時的負數值為2's，要轉回10進制時，再做一次2's，並且加上「-」號

2's範例：35 - 21

35(8位元表示)：0010 0011

21(8位元表示)：0001 0101

$$\begin{array}{r} 0010\ 0011 \\ +) 1110\ 1011 \\ \hline \text{1} 0000\ 1110 \end{array}$$

14

2's範例：21 - 35

$$\begin{array}{r} 0001\ 0101 \\ +) 1101\ 1101 \\ \hline 1111\ 0010 \end{array}$$
$$- 0000\ 1101$$

+ 1

$$- 0000\ 1110$$

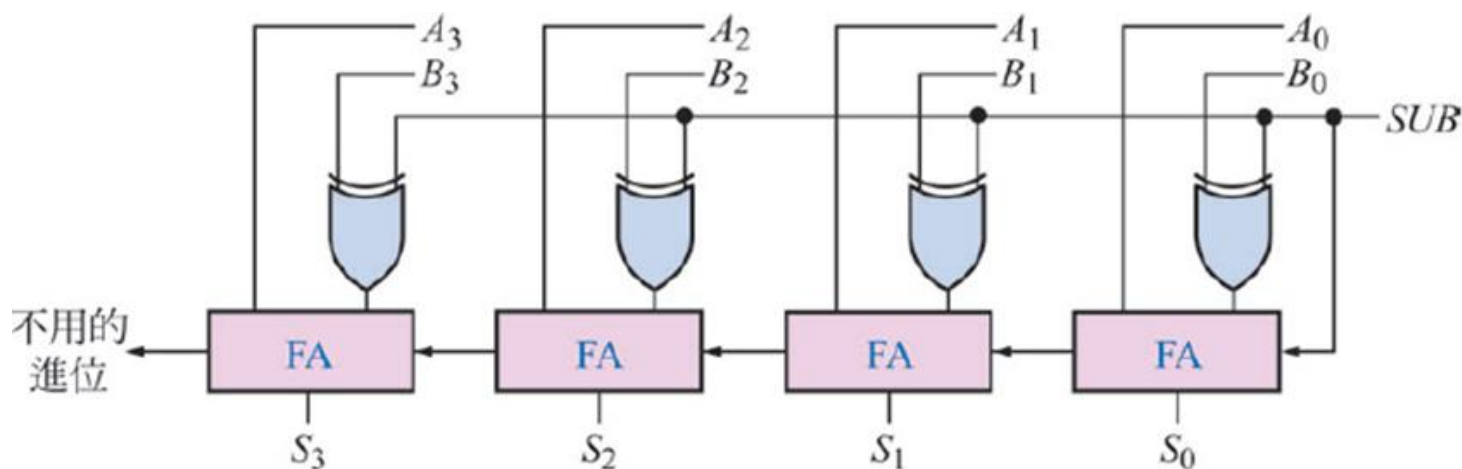
因為是2's所以要加1

-14



# 2's補數加減法電路(2/2)

- 在微電腦系統中，整數的有號數都是用2's 補數來表示的；2's 補數表示法是先把二進位數字變成1 的補數，然後再加1

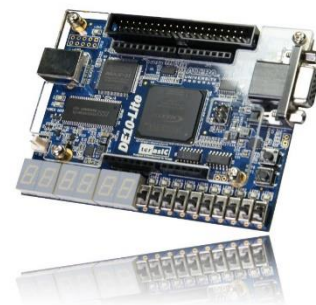


2's補數加/減法器



# BCD加法器

- 由於人類習慣於十進制，所以大多數的設備，不論是輸入或最終結果的輸出，都是以十進制來表示的，如數位式溫度計或數位電表等。
- BCD碼是以4 位元的二進位數（由0000 到1001）來代表十進制的十個數目，其加法運算與二進制大致是一樣的，但要注意的是，BCD 碼最大只到9 而已，不像4 位元的二進制最大到15；因此兩數相加結果是有必要再加以調整的，茲舉例說明如下



# BCD加法器

- 每位數的合 $\leq 9$ ，如 $45 + 33$ 為：

45	0100	0101	←45 的 BCD 碼
+ 33	0011	0011	←33 的 BCD 碼
78	0111	1000	←78 的 BCD 碼

此情況不會出現錯誤

- 有任何數字合 $> 9$ 或這有進位時，例如 $4 + 7$ ：

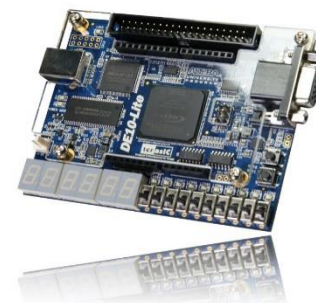
6	0110	←6 的 BCD 碼
+ 7	0111	←7 的 BCD 碼
13	1101	←BCD 碼中無此數

$$6 + 7 = 13$$

BCD Code : 00010011<sub>(BCD)</sub>

Binary Code : 00001101B

相差6 (00010011B - 00001101B = 00000110B)



# BCD加法器

- 再如  $47 + 39$  :

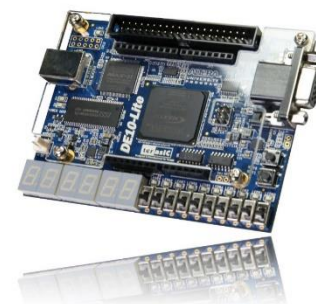
		1	←進位值
47	0100	0111	←47 的 BCD 碼
+ 39	0011	1001	←39 的 BCD 碼
<hr/>	<hr/>	<hr/>	
86	1000	0000	←結果與正確值差 6

- 從上二個例子中我們發現，在兩BCD 碼相加時，若其和有大於9 或有進位時，**其結果皆與正確值差6，即差0110B**
- (a) **BCD 的加法可先按原來的二進位加法相加**
- (b) **若和不大於9，不須修正即是正確的BCD 形式**
- (c) **若結果有進位或和大於9，則須加6 來修正；修正後，如果因進位使下一位數的和又大於9，則將該位數再加6，直到每一位數都小於9 為止**



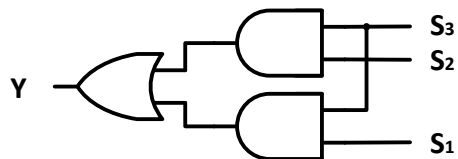
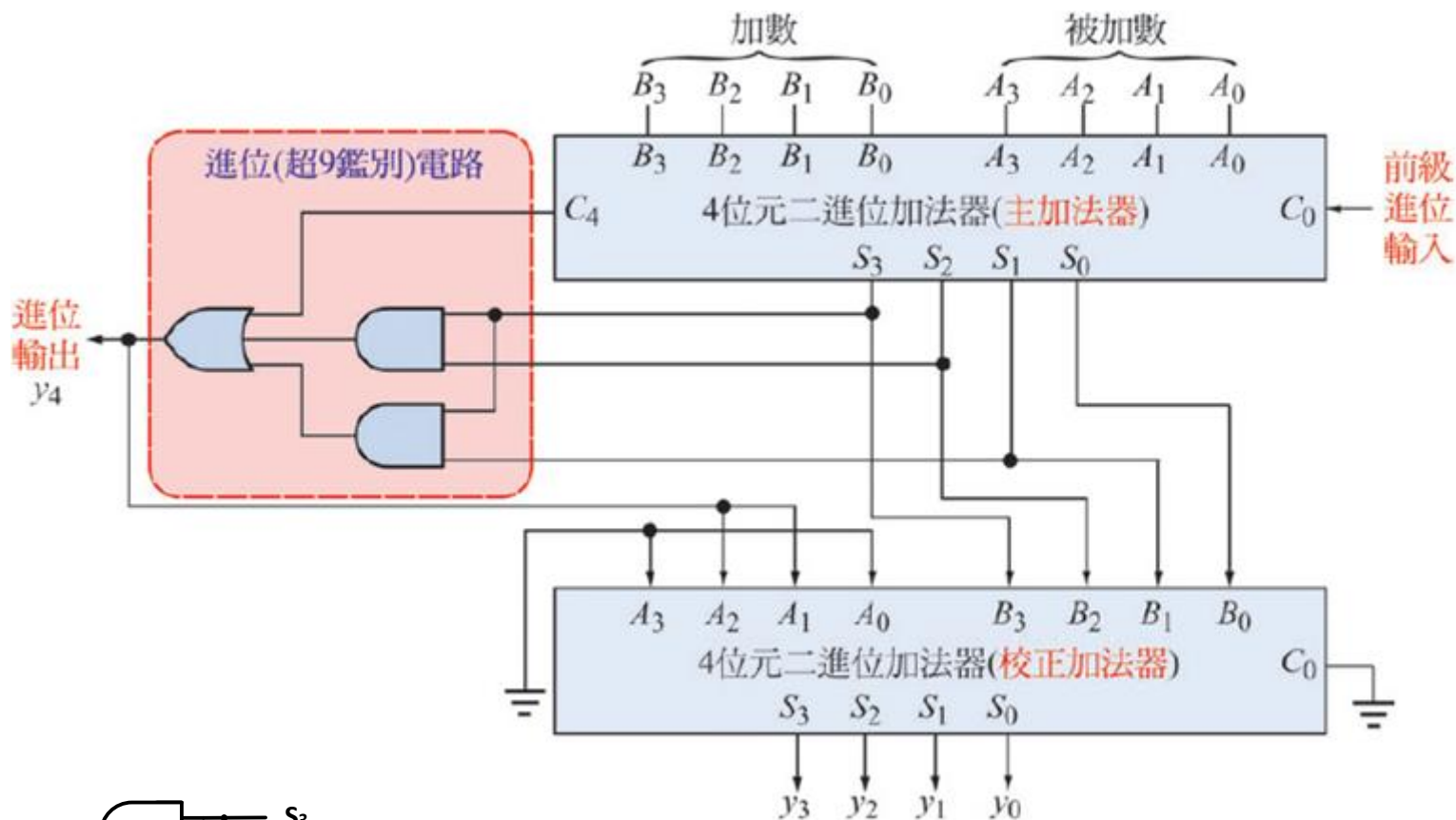
# BCD加法器

- BCD 加法器可先用二進位加法器來處理，當兩數相加結果大於9 或有進位時，再加“0110”來修正
- 因此在電路的設計上我們需要兩組加法器，一組做加數與被加數相加，另一組則用來做加6或加0，即當兩數相加結果有進位或大於9 時加6，否則則加0(鑑別電路)





# BCD加法器

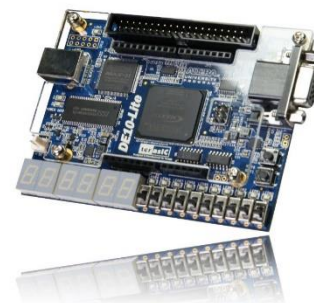
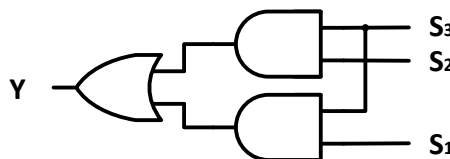


# BCD加法器-超9鑑別電路化簡

S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

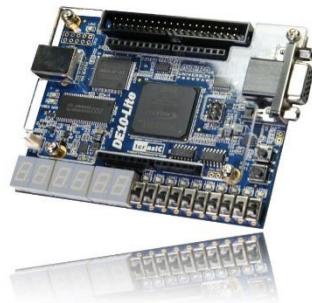
$\begin{matrix} S_1 S_0 \\ S_3 S_2 \end{matrix}$	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

$$Y = S_1 S_3 + S_2 S_3$$



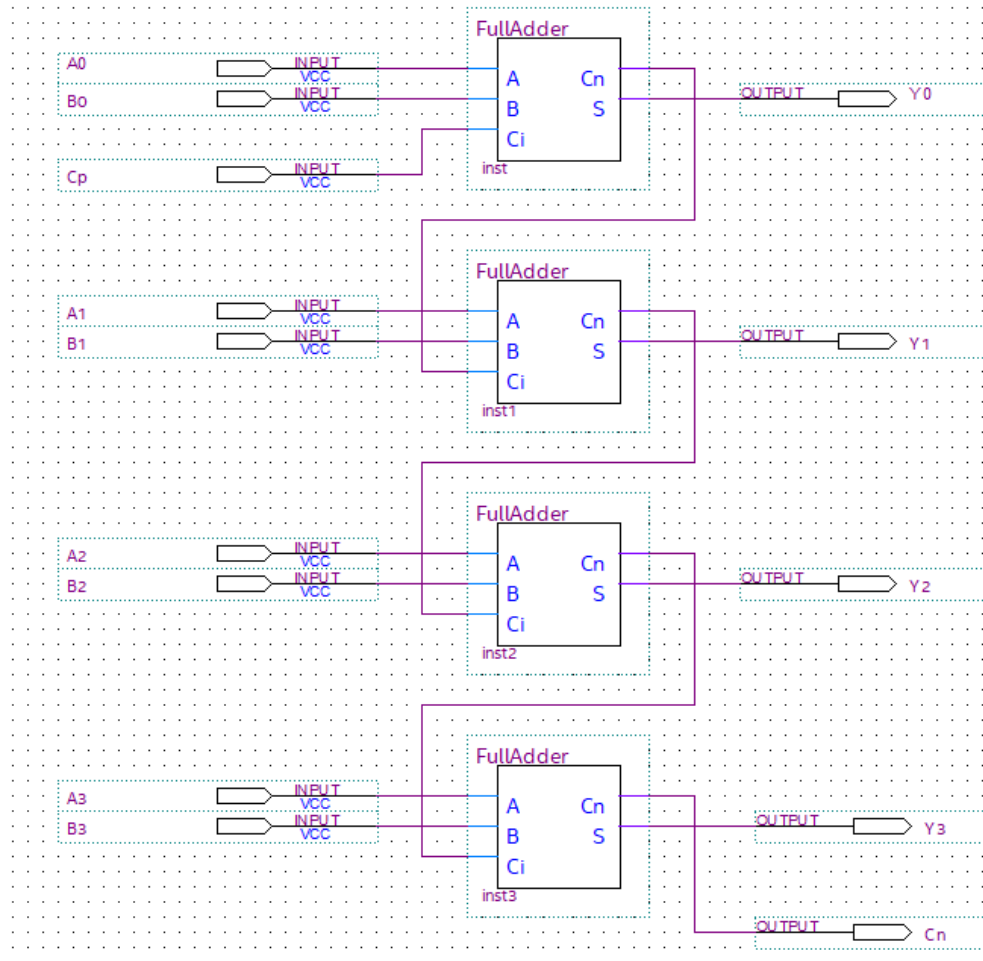
# 隨堂練習

- 請完成一位數BCD加法器電路，並完成紀錄，包括**電路設、模擬波形圖、真值表**(與模擬波形相符)。
- 本次每一個實驗完成後需助教確認正確，在做下一題，全部完成後，將專案壓縮上傳EE-Class。
- Lecture6\_組別XX.ZIP



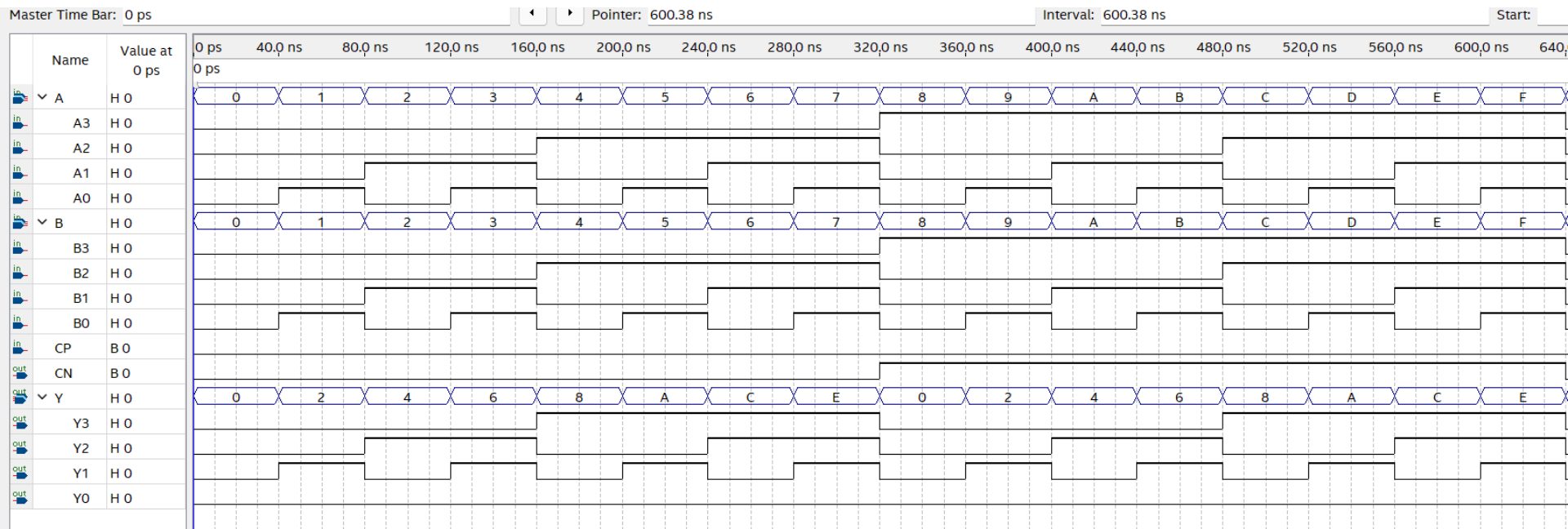
# 隨堂練習

- 4-Bit Adder



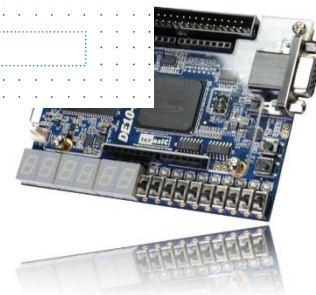
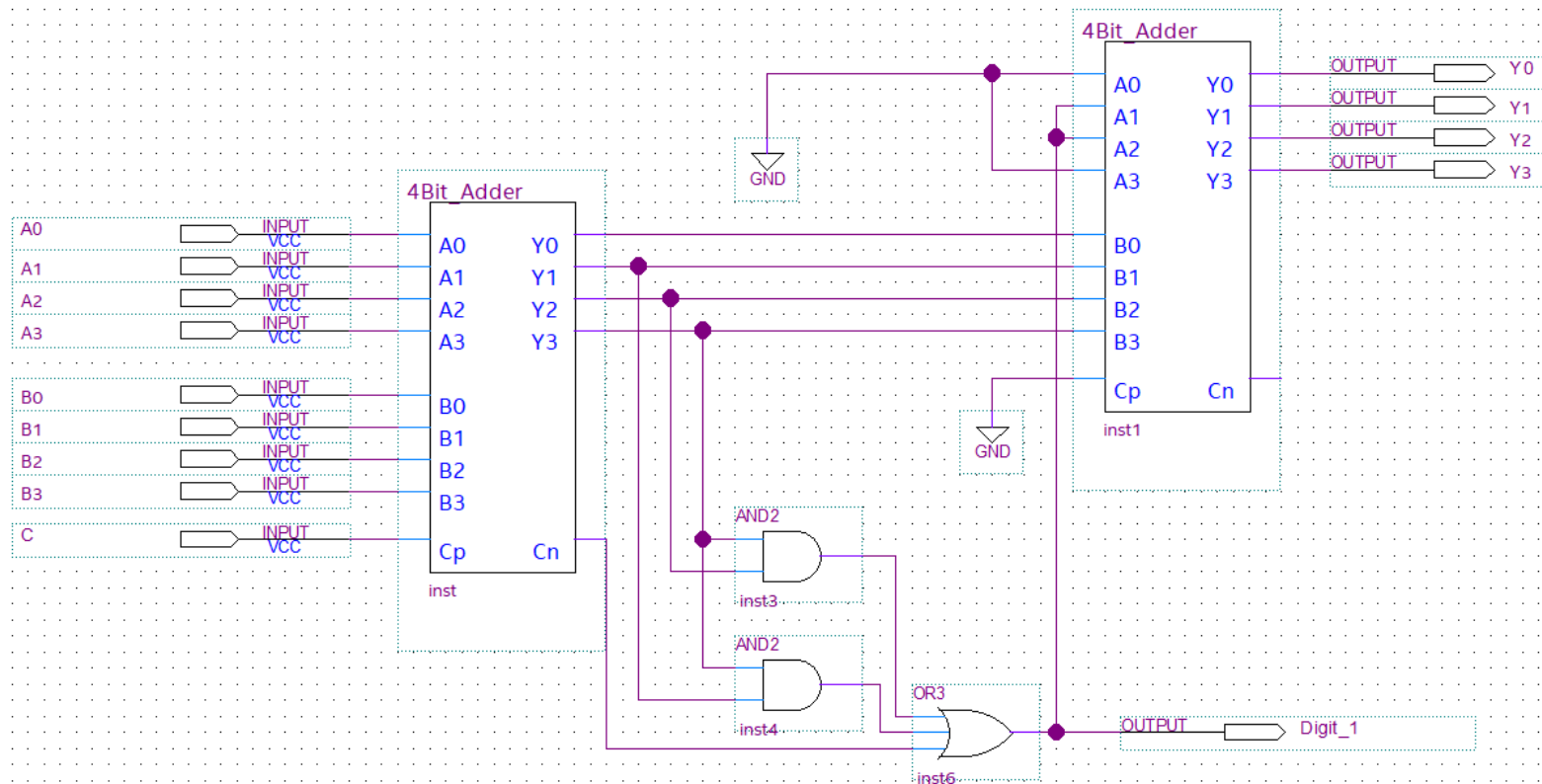
# 隨堂練習

- 4-Bit Adder



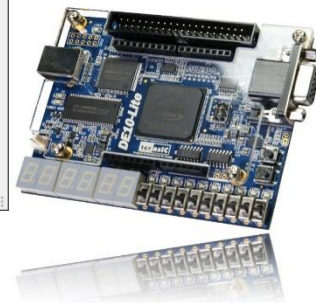
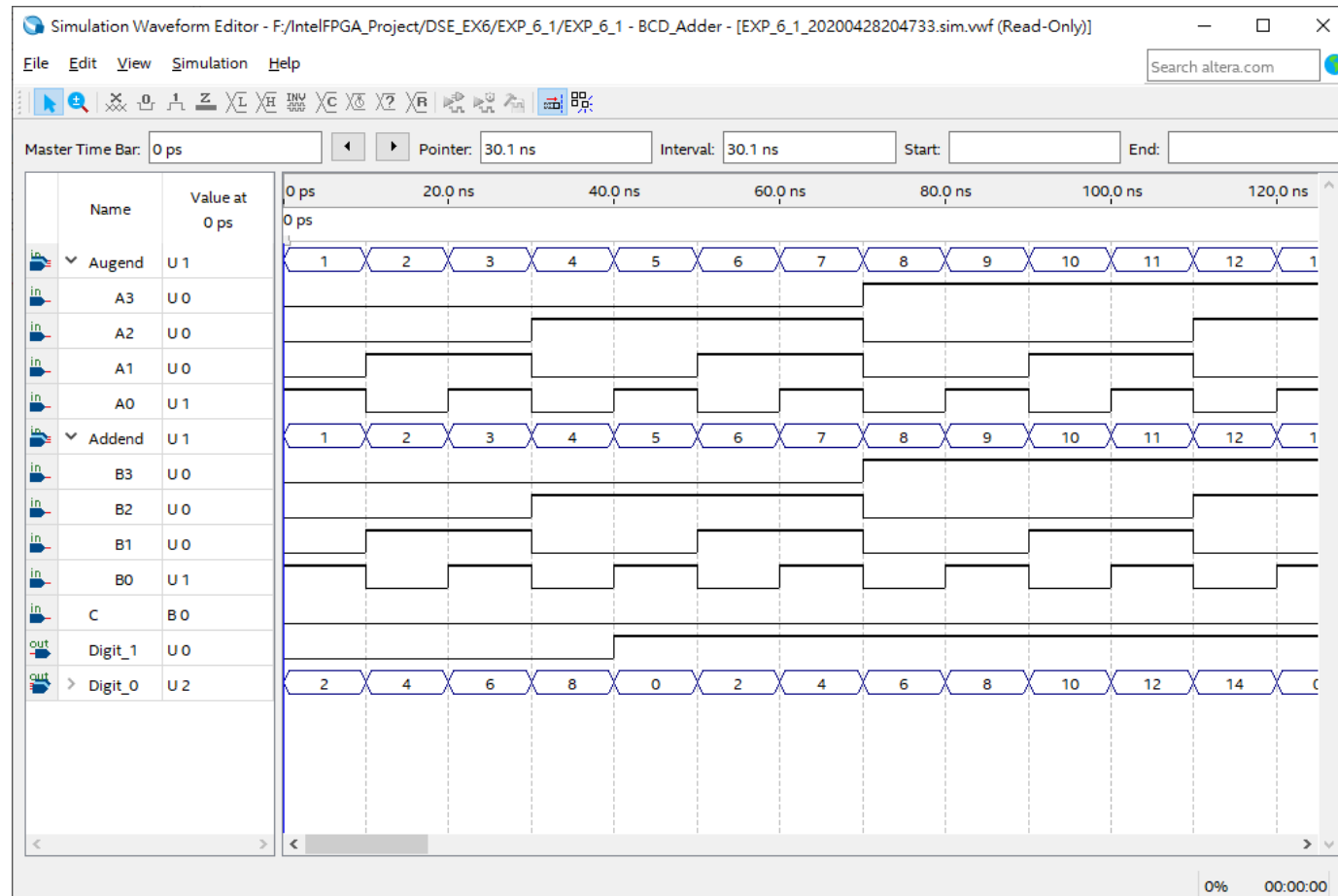
# 隨堂練習

- Top-Level Entity



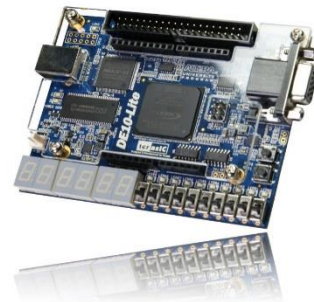
# 隨堂練習

- Simulation



# 隨堂練習

- 注意：Top-Level Entity





# 隨堂練習

- 注意：Create Symbol Files for Current File

