

# 電機一乙 數位邏輯實習

## 第五週實習作業報告

### I.實驗目的與原理

#### A.作業 1






































利用 VHDL 實作 1 位元的加減法器，將 1 位元加減法器串接為 4 位元加減法器。完成下列真值表，並推導 Cout 與 Sum 的布林代數式，Sub 為 0 時進行加法運算，為 1 時進行減法運算。

表一：加減法器真值表

A	B	C <sub>in</sub>	Sub	C <sub>out</sub>	Sum
0	1	0	0	0	1
1	0	1	0	1	0
1	1	1	0	1	1
1	0	1	1	0	0
0	1	0	1	1	1
1	1	1	1	1	1
0	1	1	1	1	0

#### B.作業 2

利用 VHDL 實作 Braille 點字系統(圖一)。以 A、B、C、D 作為輸入，輸出參考表二，導出最簡化的布林代數式後利用 VHDL 實作電路。

												
A	B	C	D	E	F	G	H	I	J	K	L	M
												
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
												
#	1	2	3	4	5	6	7	8	9	0		

圖一：Braille 點字系統英文字母對照表

表二：點字系統輸入輸出對照表

A	B	C	D	輸出	A	B	C	D	輸出
0	0	0	0	A	1	0	0	0	I
0	0	0	1	B	1	0	0	1	J
0	0	1	0	C	1	0	1	0	K
0	0	1	1	D	1	0	1	1	L
0	1	0	0	E	1	1	0	0	M
0	1	0	1	F	1	1	0	1	N
0	1	1	0	G	1	1	1	0	O
0	1	1	1	H	1	1	1	1	P

## II. 實驗過程

### A. 作業 1

完成真值表(表三)後利用卡諾圖(圖二、三)推導出輸出 Cout 與 Sum 之布林代數式-(1)及式-(2)。

表三：加減法器真值表

A	B	Ci	Sub	Co	Sum
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	0	1
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	1	1	1
0	1	1	0	1	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	0	1	0	0
1	1	1	0	1	1
1	1	1	1	1	1

		Ci Sub			
		00	01	11	10
AB	00	0	0	1	0
	01	0	1	1	1
	11	1	0	1	1
	10	0	0	0	1

圖二：Cout 卡諾圖

		Ci Sub			
		00	01	11	10
AB	00	0	0	1	1
	01	1	1	0	0
	11	0	0	1	1
	10	1	1	0	0

圖三：Sum 卡諾圖

$$C_{out} = BC_i + ABS' + AC_iS' + A'BS + A'C_iS \text{ ---- (1)}$$

$$Sum = A'B'C_i + A'BC_i' + ABC_i + ABC_i + AB'C_i' \text{ ---- (2)}$$

其 VHDL 如下，輸入進 external 後，得到 1 位元加減法器(圖四)。

```

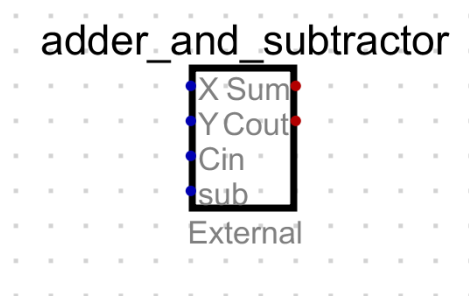
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.numeric_std.all;

entity adder_and_subtractor is
  port (
    X, Y, Cin, sub: in std_logic;
    Sum, Cout: out std_logic);
end adder_and_subtractor;

architecture Equations of adder_and_subtractor is
begin
  -- concurrent assignment statements
  cout <= (y and cin) or (x and y and not sub) or (x and cin and not sub) or
    (not x and y and sub) or (not x and cin and sub);
  sum <= (not x and not y and cin) or (not x and y and not cin) or (x and y and cin) or
    (x and not y and not cin);
end Equations;

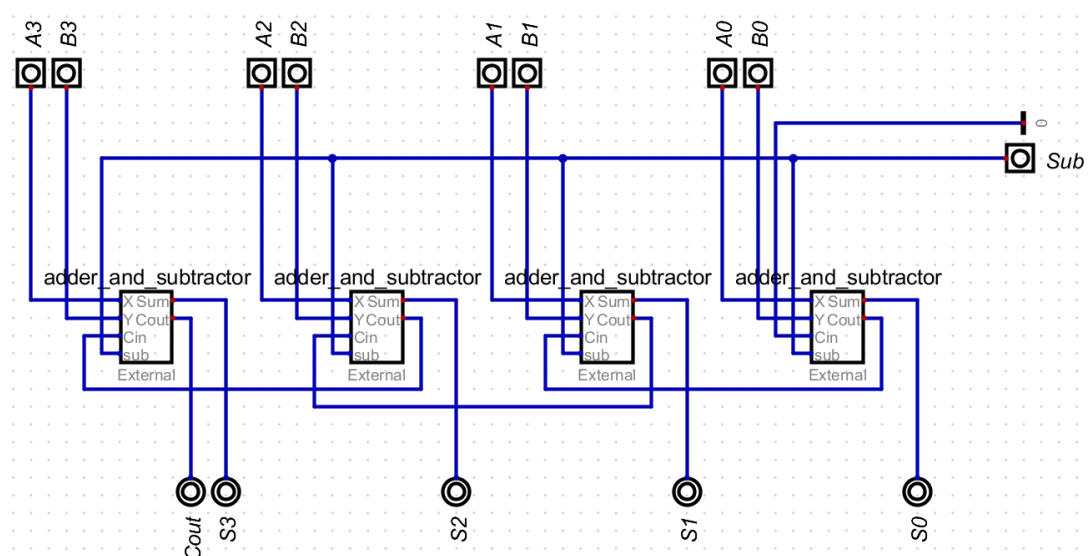
```

布林代數式



圖四：1 位元加減法器

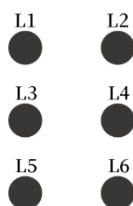
將四個 1 位元加減法器的 Ci、Co 互相連接、Sub 接出，得到 4 位元加減法器(圖五)。



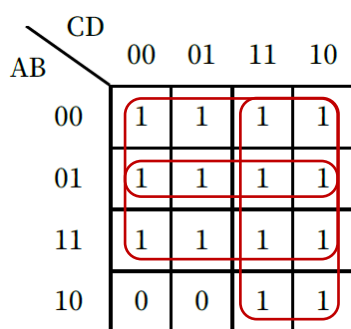
圖五：4 位元加減法器

## B. 作業 2

在這裡我們使用 LED 來代替點字卡表示轉換後的輸出，為此我們需要求出每顆 LED (L1~L6) 的布林代數式，L1~L6 與原點字系統對照如下圖六，L1~L6 經卡諾圖化簡(如下圖七~十二)後得布林代數式-3~式-8。

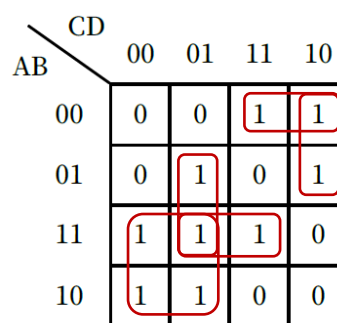


圖六：LED 腳位對照



圖七：L1 卡諾圖

$$L1 = C + A' + B' - (3)$$



圖八：L2 卡諾圖

$$L2 = AC' + BC'D + ABD + A'B'C + A'CD' - (4)$$

AB \ CD				
	00	01	11	10
00	0	1	0	0
01	0	1	1	1
11	0	0	1	0
10	1	1	1	0

圖九：L3 卡諾圖

$$L3 = A'C'D + A'BC + AB'C' + ACD - (5)$$

AB \ CD				
	00	01	11	10
00	0	0	1	0
01	1	0	1	1
11	0	1	0	1
10	0	1	0	0

圖十：L4 卡諾圖

$$L4 = A'BD' + A'CD + BCD' + AC'D - (6)$$

AB \ CD				
	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

圖十一：L5 卡諾圖

$$L5 = AB + AC - (7)$$

AB \ CD				
	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	0	0
10	0	0	0	0

圖十二：L6 卡諾圖

$$L6 = 0 - (8)$$

其 VHDL 如下，輸入進 external 後，得 Braille 點字系統轉換器(圖十三)。

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.numeric_std.all;
```

```
entity hw2 is
```

```
port (
```

```
  A, B, C, D: in std_logic;
```

```
  L1, L2, L3, L4, L5, L6: out std_logic);
```

```
end hw2;
```

```
architecture Equations of hw2 is
```

```
begin
```

```
  -- concurrent assignment statements
```

輸入輸出

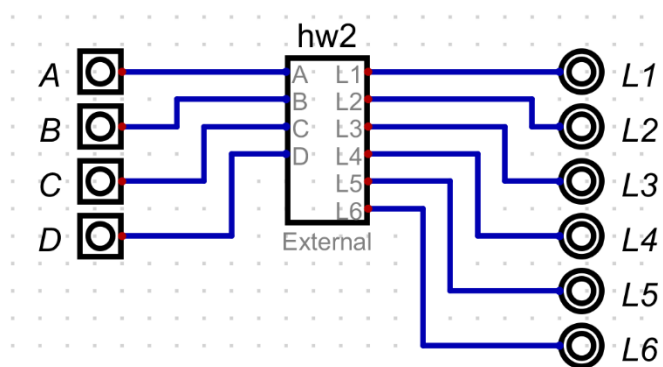
```

L1 <= not a or b or c;
L2 <= (a and not c) or (b and not c and d) or (a and b and d) or
      (not a and not b and c) or (not a and c and not d);
L3 <= (not a and not c and d) or (not a and b and c) or
      (a and not b and not c) or (a and c and d);
L4 <= (not a and b and not d) or (not a and c and c) or
      (b and c and not d) or (a and not c and d);
L5 <= (a and b) or (a and c);
L6 <= '0';

```

end Equations;

布林代數式



圖十三：Braille 點字系統轉換器

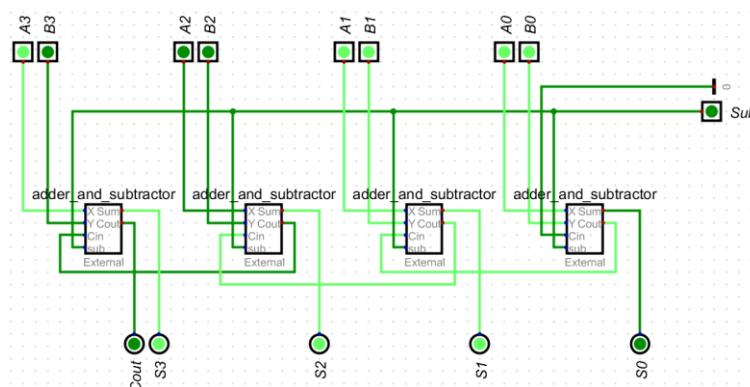
### III. 模擬驗證

#### A. 作業 1

利用 Digital 進行模擬，確認輸出與真值表無誤。

#### 1. 相加(sub=0)

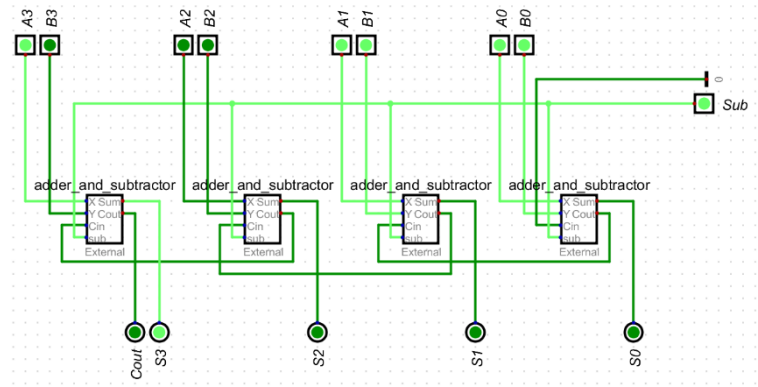
當  $A=11(1011)$ 、 $B=3(0011)$   
時，加減法器輸出為  $11+3 =$   
 $14(1110)$ 。



圖十四：加減法器模擬電路 a

## 2. 相減(sub=1)

當  $A=11(1011)$ 、 $B=3(0011)$   
時，加減法器輸出為  $11-3 =$   
 $8(1000)$ 。

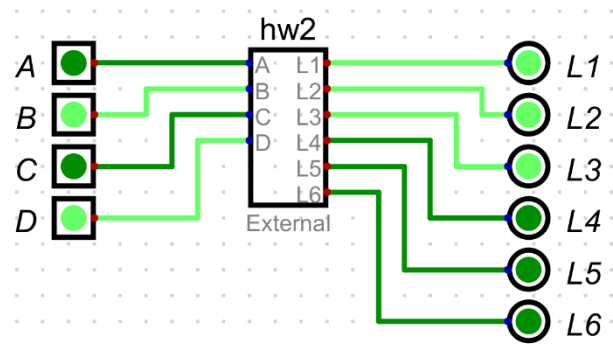


圖十五：加減法器模擬電路 b

## B. 作業 2

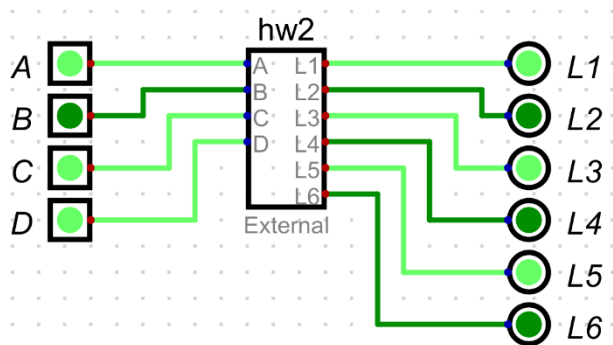
同樣利用 Digital 進行模擬，檢驗結果與預期無誤。

1. 當輸入為 0101 時，輸出為 F，L1、L2、L3 亮。



圖十六：Braille 點字系統模擬電路 a

2. 當輸入為 1011 時，輸出為 L，L1、L3、L5 亮。



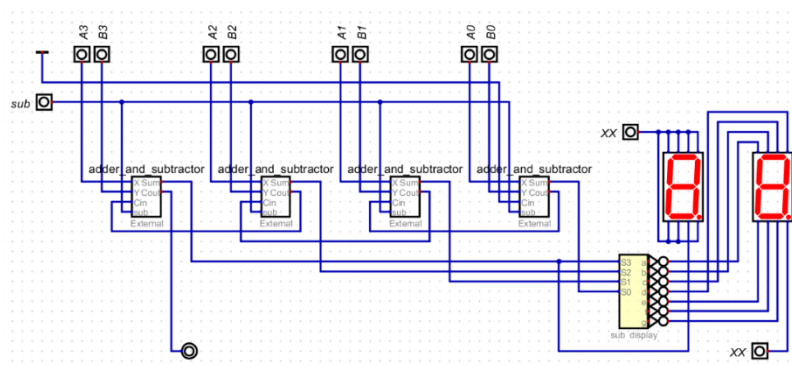
圖十七：Braille 點字系統模擬電路 b

## IV. 實驗結果與成果討論

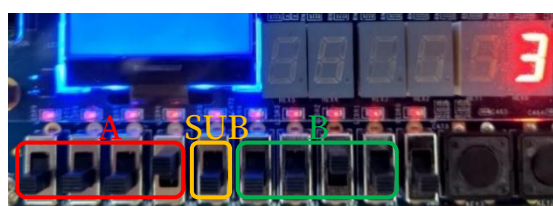
### A. 作業 1

在測試實際電路的部分，我們以開關為輸入，七段顯示器為輸出，結合之前所做的七段顯示解碼器來測試此 4 位元加減法器，方便觀察，電路如下圖十八，燒入電路板後實際

操作如下圖十九及圖二十。



圖十八：作業 1 測試電路



$$1(0001) + (0) \quad 2(0010) = 3$$

圖十九：作業 1 實際操作 a



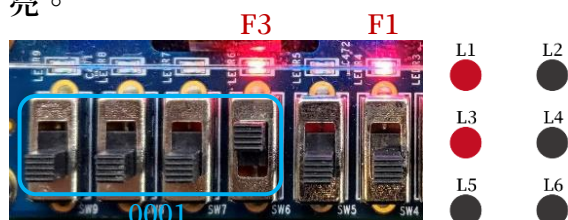
$$2(0010) - (1) \quad 6(0110) = -4$$

圖二十：作業 1 實際操作 a

## B. 作業 2

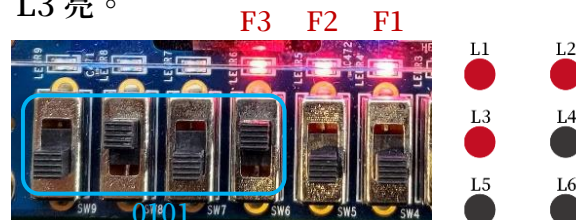
我們以開關作為輸入、LED 作為輸出，開關由 SW9~SW6 依序為 A、B、C、D，LED 由右而左依序為 L1~L6，燒入後實際操作如下圖。

當輸入為 0001 時，輸出為 B，L1、L3 亮。



圖二十一：作業 2 實際操作 a

當輸入為 0101 時，輸出為 F，L1、L2、L3 亮。



圖二十二：作業 2 實際操作 b

## V. 實驗心得

雖然這次組員不在，但意外地很快就做完了，在打 VHDL 的時候也沒碰到甚麼問題，整體上來說還蠻順利的。希望下次難度增加後也能順利的完成。