105061254 林士平 邏輯設計實驗報告 Lab2

1.

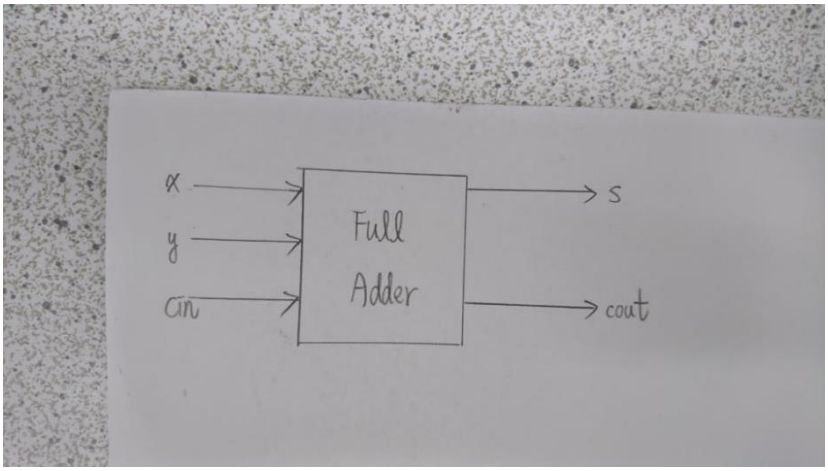
(1) Design specification :

A. Inputs and outputs(表一)：

Inputs	x, y, cin
Outputs	s, cout

↑表一：Inputs and outputs of 1.

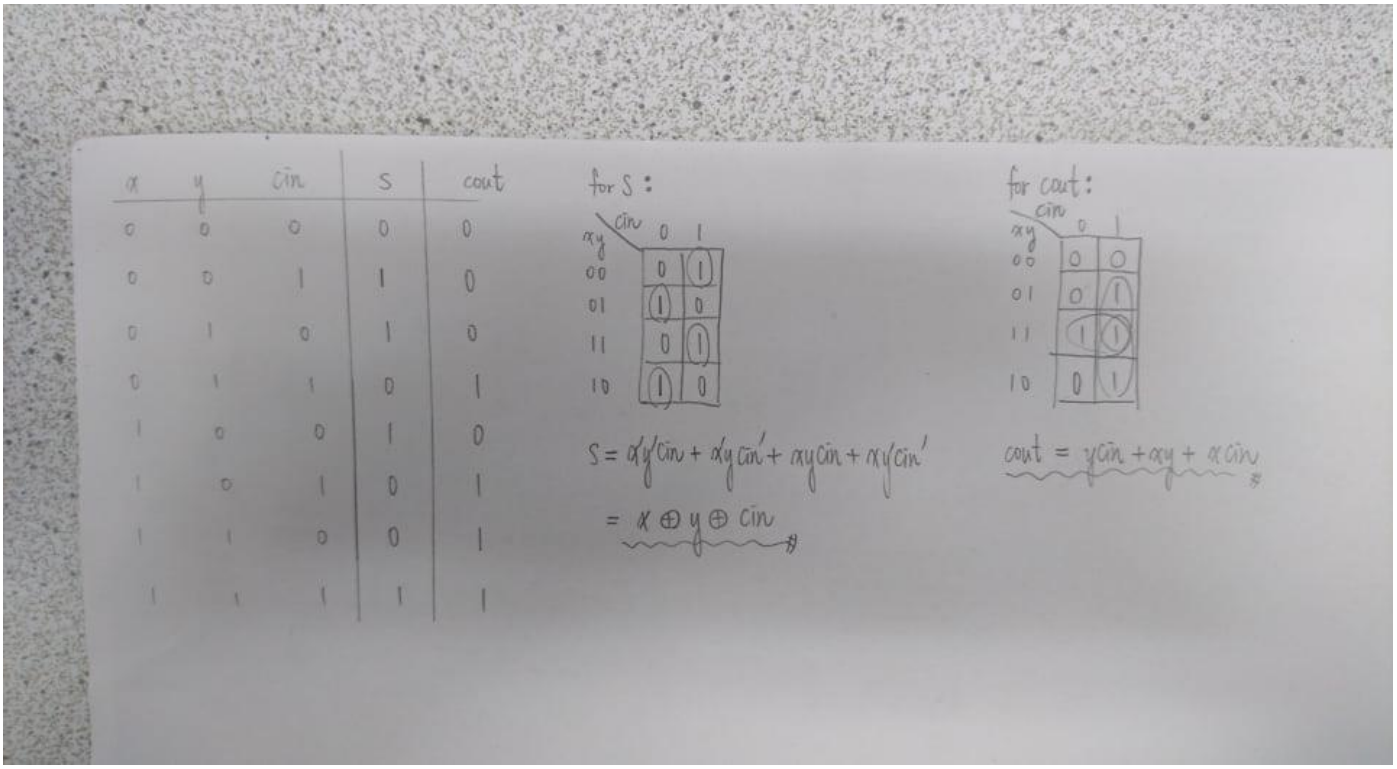
B. Block diagram(圖一)：



↑圖一：The block diagram of 1.

(2) Design implementation :

A. Logic function(圖二)：



↑圖二：logic funtion 計算過程

由(圖二)可知 logic function 為：

$$s = \mathrm{x} \oplus \mathrm{y} \oplus \mathrm{cin}$$

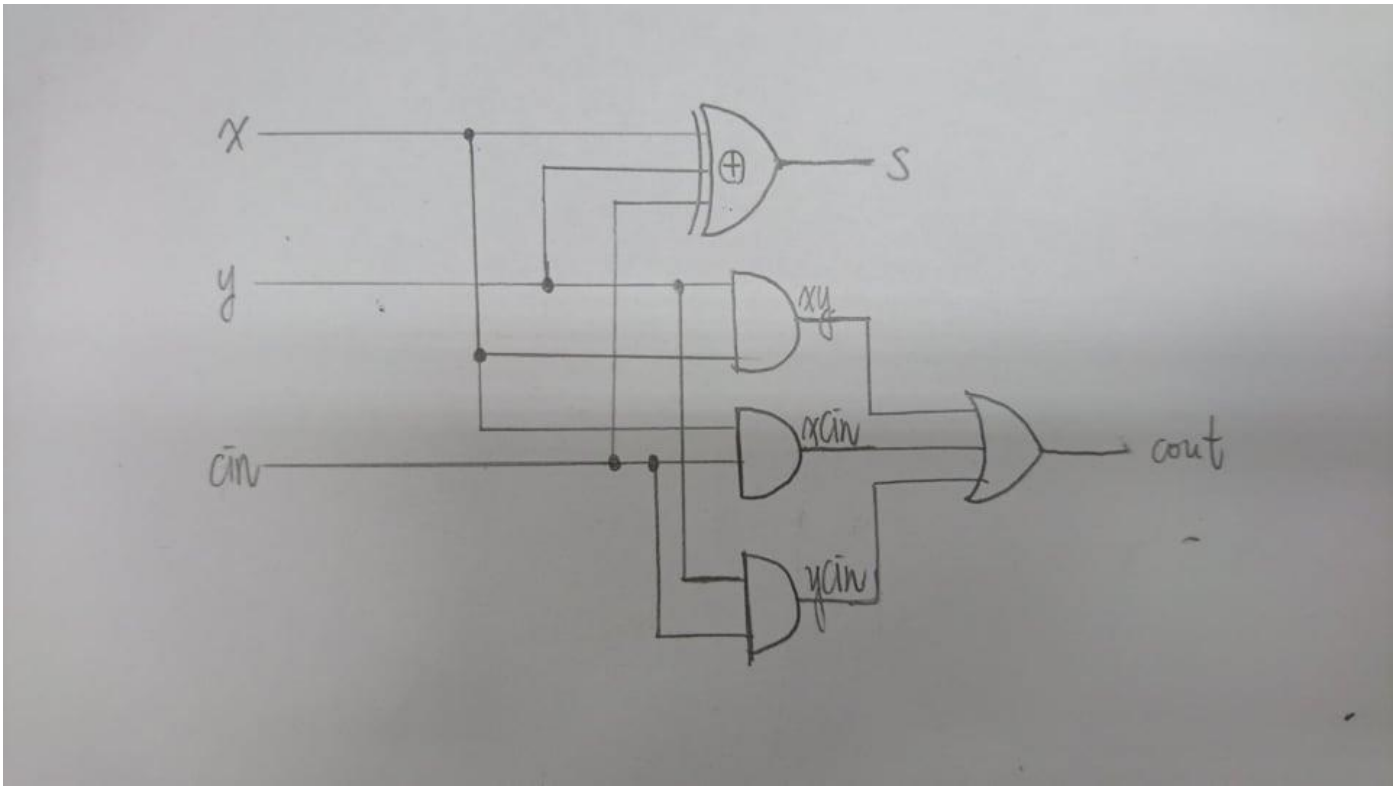
$$\mathrm{cout} = \mathrm{ycin} + \mathrm{xy} + \mathrm{xcin}$$

寫成 verilog 的形式如下：

$$s = x \ \hat{} \ y \ \hat{} \ cin$$

$$cout = (x \ \& \ y) \mid (x \ \& \ cin) \mid (y \ \& \ cin)$$

B. Logic diagram(圖三)：



↑圖三：logic diagram of 1.

C. I/O pin assignment(表二)：

I/O	x	y	cin	s	cout
LOC	V17	V16	W16	U16	E19
↑表二：I/O pin assignment of 1.					

D.功能與做法說明：

本題為一個 full-adder，input 是兩個 1 bit 的資料(x,y)和一個 1 bit 的 carry in(cin)，輸出為加總的結果 s 和 carry out(cout)。利用 truth table 以及 k-map 可以將 logic function 給推出來（圖二），並且利用得到的 logic function 可以寫成 verilog code 將 full-adder 的功能實現。最後再將 input 和 output assign 給 FPGA 板的接腳，便能藉操作 FPGA 板上的開關，從 LED 燈得到結果。

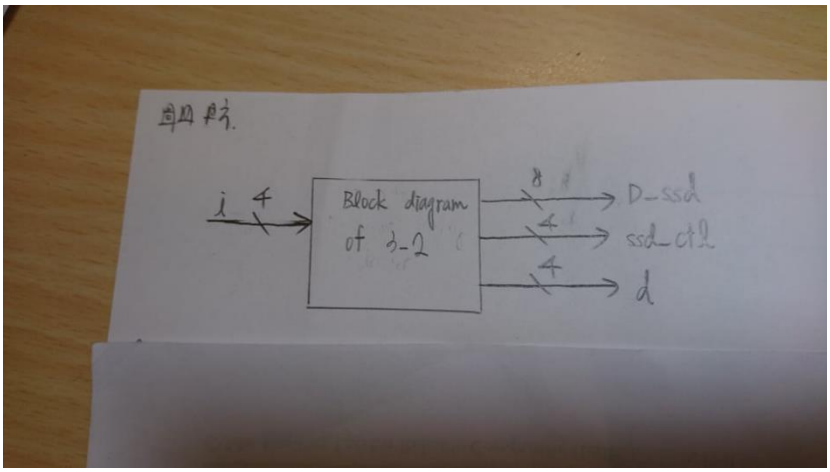
2.

(1) Design specification :

A. Inputs and outputs(表三)：

Inputs	I[3:0]
Outputs	D_ssd[7:0], d[3:0], ssd_ctl[3:0]
↑表三：Inputs and outputs of 2.	

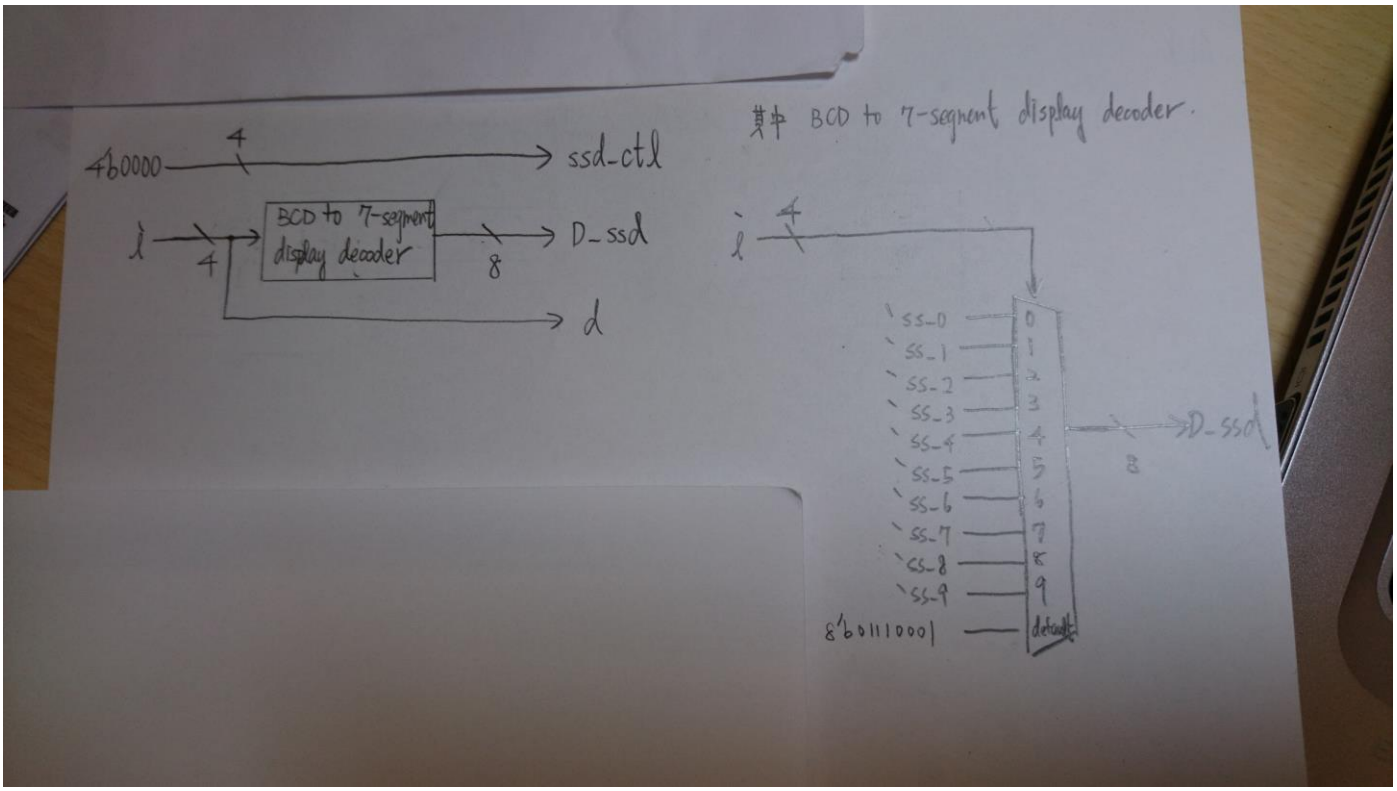
B. Block diagram(圖四)：



↑圖四：The block diagram of 2.

(2) Design implementation :

A. Logic diagram(圖五)：



↑圖五：logic diagram of 2.

B. I/O pin assignment(表四)：

I/O	i[3]	i[2]	i[1]	i[0]	ssd_ctl[3]	ssd_ctl[2]	ssd_ctl[1]
LOC	W17	W16	V16	V17	W4	V4	U4
I/O	ssd_ctl[0]	D_ssd[7]	D_ssd[6]	D_ssd[5]	D_ssd[4]	D_ssd[3]	D_ssd[2]
LOC	U2	W7	W6	U8	V8	U5	V5
I/O	D_ssd[1]	D_ssd[0]	d[3]	d[2]	d[1]	d[0]	
LOC	U7	V7	V19	U19	E19	U16	

↑表四：I/O pin assignment of 2.

C.功能與做法說明：

本題為輸入一個 BCD，讓它顯示在七段顯示器上(四個七段顯示器顯示相同數字)。我用了兩個.v 檔：top_module 和 display。top_module 負責接收輸入的 BCD(i)和輸出 output(ssd_ctl 用來控制哪個七段顯示器要顯示數值、D_ssd 用來控制七段顯示器要顯示的數值為何、d 用來控制哪個 LED 燈要亮)，其中 ssd_ctl = 4'b0000，讓四個七段顯示器均顯示；LED 要能顯示輸入(i)的值，只要將 i 和 LED 的輸出訊號(d)連在一起就可以了。display 則是 BCD to 7-segment display decoder，基本上就是一個 multiplexer，用來判斷輸入 BCD 相對應的 D_ssd，如果輸入不是 BCD 的話則顯示 F。

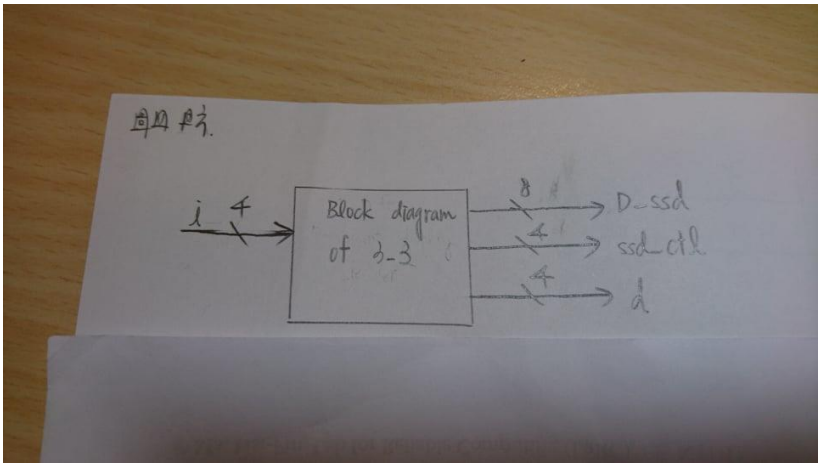
3.

(1) Design specification :

A. Inputs and outputs(表五)：

Inputs	I[3:0]
Outputs	D_ssd[7:0], d[3:0], ssd_ctl[3:0]
↑表五：Inputs and outputs of 3.	

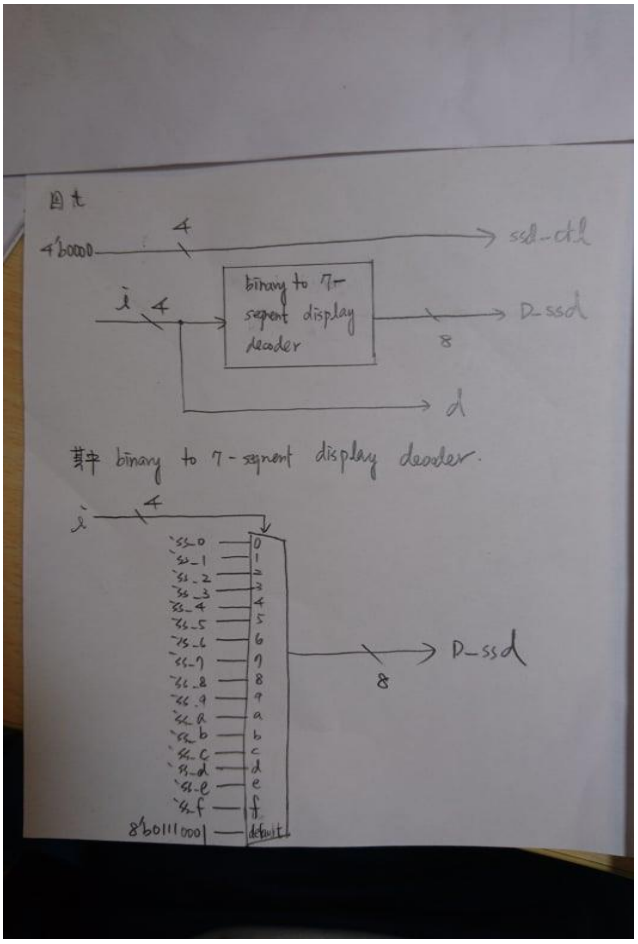
B. Block diagram(圖六)：



↑圖六：The block diagram of 3.

(2) Design implementation :

A. Logic diagram(圖七)：



←圖七：logic diagram of 3.

B. I/O pin assignment(表六)：

I/O	i[3]	i[2]	i[1]	i[0]	ssd_ctl[3]	ssd_ctl[2]	ssd_ctl[1]
LOC	W17	W16	V16	V17	W4	V4	U4
I/O	ssd_ctl[0]	D_ssd[7]	D_ssd[6]	D_ssd[5]	D_ssd[4]	D_ssd[3]	D_ssd[2]
LOC	U2	W7	W6	U8	V8	U5	V5
I/O	D_ssd[1]	D_ssd[0]	d[3]	d[2]	d[1]	d[0]	
LOC	U7	V7	V19	U19	E19	U16	

↑表六：I/O pin assignment of 3.

C.功能與做法說明：

本題和 2_2 題非常像，差別在於輸入(i)是 binary number，要讓七段顯示器能顯示 a, b, c, d, E, F。因此兩題只差在 display module 中的 multiplexer，只要多加幾個判斷式即可。

4.

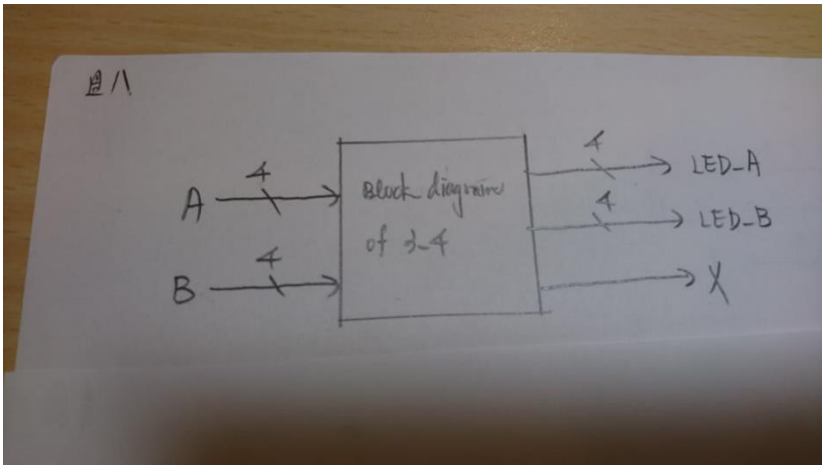
(1) Design specification :

A. Inputs and outputs(表七)：

Inputs	A[3:0], B[3:0]
Outputs	LED_A[3:0], LED_B[3:0], X

↑表七：Inputs and outputs of 4.

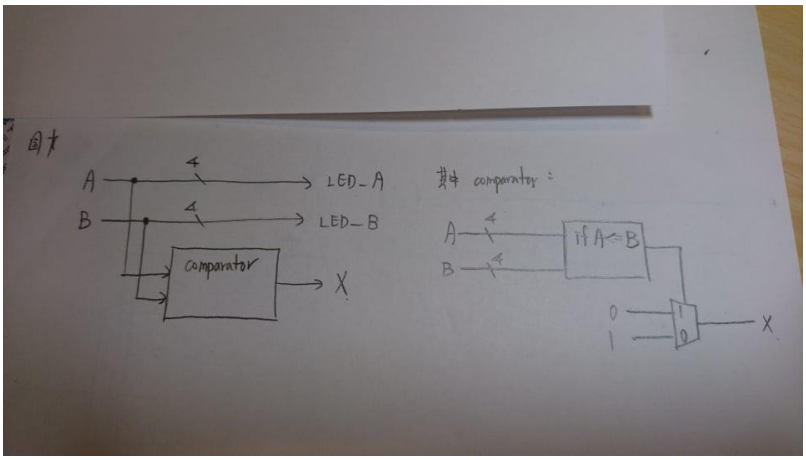
B. Block diagram(圖八)：



↑圖八：The block diagram of 4.

(2) Design implementation :

A. Logic diagram(圖九)



↑圖九：logic diagram of 4.

B. I/O pin assignment(表八)：

I/O	A[3]	A[2]	A[1]	A[0]	B[3]	B[2]	B[1]
LOC	W13	W14	V15	W15	W17	W16	V16
I/O	B[0]	LED_A[3]	LED_A[2]	LED_A[1]	LED_A[0]	LED_B[3]	LED_B[2]
LOC	V17	V14	U14	U15	W18	V19	U19
I/O	LED_B[1]	LED_B[0]	X				
LOC	E19	U16	V13				

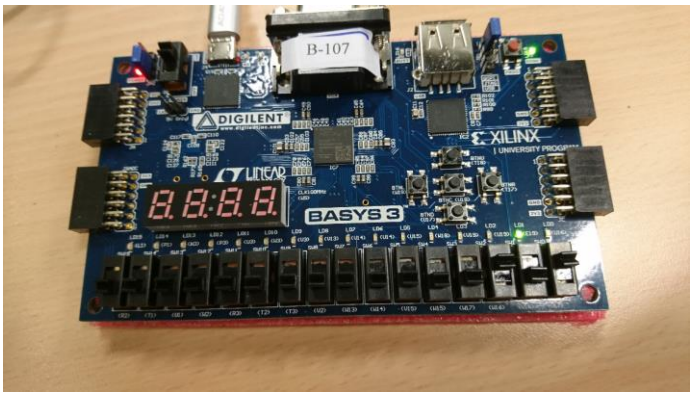
↑表八：I/O pin assignment of 4.

C.功能與做法說明：

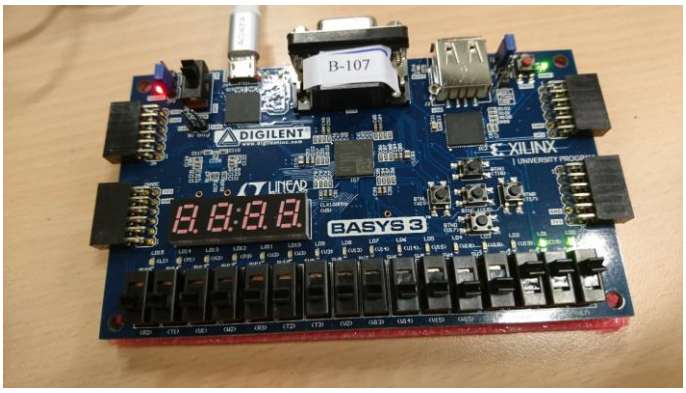
本題是一個 comparator，如果 A>B 就輸出 X = 1；如果 A<=B 就輸出 X = 0。基本上只要用一個 MUX 來做判斷即可，判斷式為 A 和 B 比大小，由判斷式的 true,false 來決定輸出的 X 值。LED 要能顯示輸入(A 和 B)的值，只要將 A, B 和他們各自的 LED 輸出訊號(LED_A, LED_B)連在一起就可以了。

5. Discussion

第 2_1 題要將 lab1 做的 full-adder 燒到 FPGA 版上實現，過程中可能會遇到的問題有：忘記更改 I/O Std 到 LVCMOS33、接腳選錯。不過基本上只要夠細心、按照講義上的步驟，就能做出來。

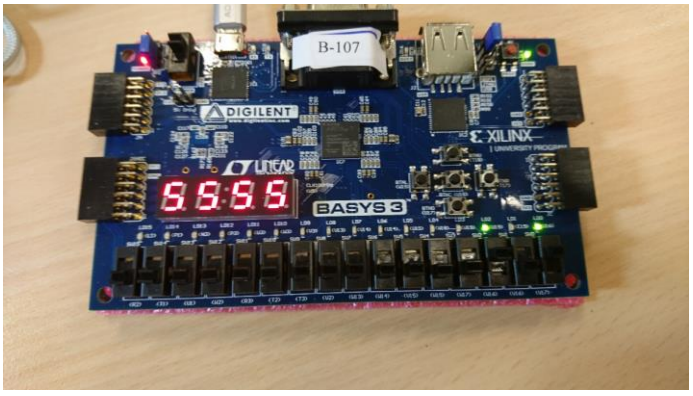


↑圖十(左)：1.的實作結果(x = 1, y = 0, cin = 1 所以 cout = 1, s = 0)

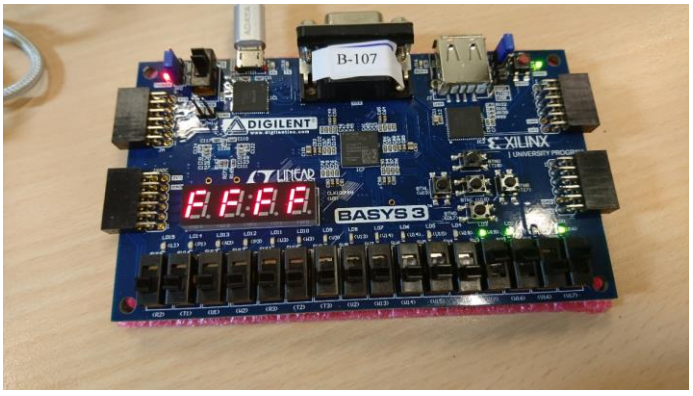


↑圖十(右)：1.的實作結果(x = 1, y = 1, cin = 1 所以 cout = 1, s = 1)

第 2_2 題是輸入一個 BCD，讓它顯示在七段顯示器上(四個七段顯示器顯示相同數字)。過程中最需要注意的是 7-seg Display 是 low active control，如果搞反結果會完全相反，還有要注意選擇七段顯示器接腳的時候，最高位元和最低位元對應的接腳為何，如果搞反顯示出來的結果也會不如預期。

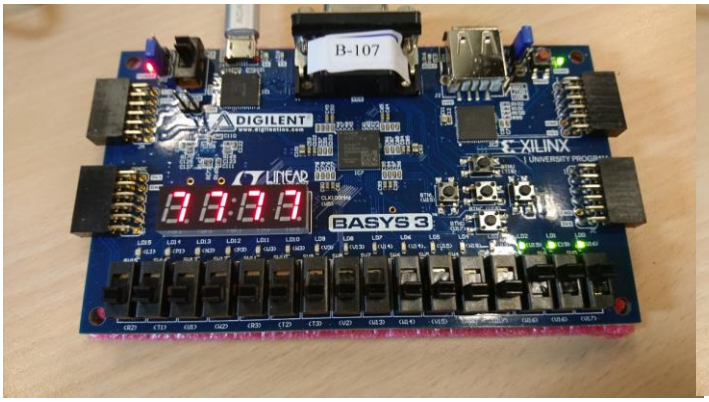


↑圖十一(左)：2.的實作結果(i[3:0] = 4'b0101 所以顯示 5)

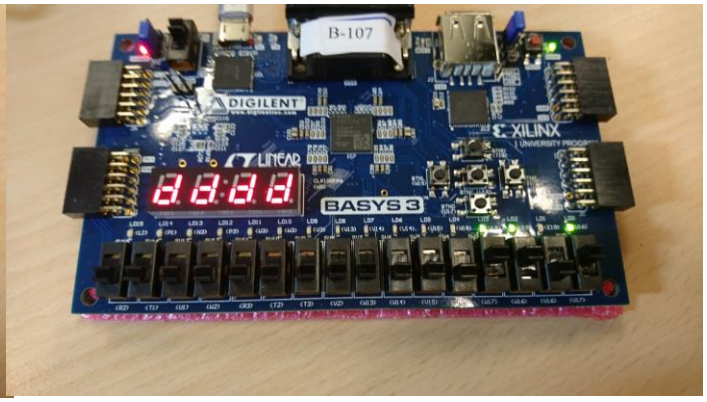


↑圖十一(右)：2.的實作結果(i[3:0] = 4'b1101 所以顯示 F(i is outside the range))

第 2_3 題和 2_2 題非常像，差別在於輸入(i)是 binary number，要讓七段顯示器能顯示 a, b, c, d, E, F。基本上這題可以將 2_2 題的.xdc 檔完全複製過來，.v 檔也幾乎一樣，所以不會很困難。

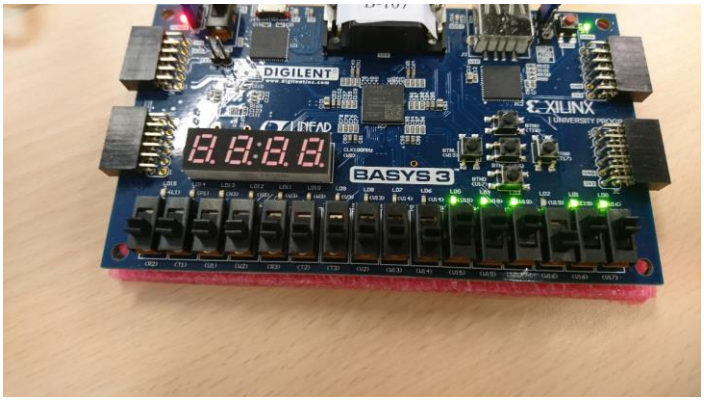


↑圖十二(左)：3.的實作結果(i[3:0] = 4’b0111 所以顯示 7)

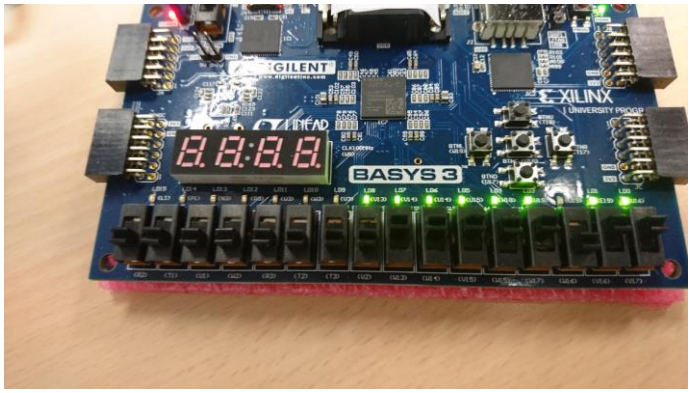


↑圖十二(右)：3.的實作結果(i[3:0] = 4’b1101 所以顯示 d

最後第 2_4 題主體其實就是一個 mux，在寫完 2_2 和 2_3 題之後，2_4 題寫起來沒甚麼困難。



↑圖十三(左)：4.的實作結果(A = 3, B = 11,所以 X = 0)



↑圖十三(右)：4.的實作結果(A = 16, B = 11,所以 X = 1)

6. Conclusion

這次的實驗讓我熟悉如何將 verilog code 燒到 FPGA 版上實現。從 I/O pin assignment 到產生 bitstream，我覺得自己更了解 verilog 在做甚麼。能實際看到成果感覺很有成就感。