

邏輯設計實驗 Lab02 結報

104060012 邱怡庭

1 Emulate exp1 in lab1 (a full adder $s+cout=x+y+cin$) with the following parameters.

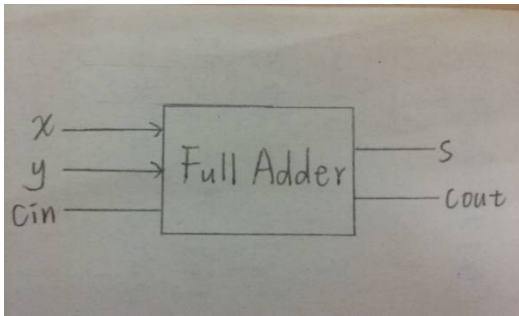
I/O	x	y	cin	s	cout
LOC	T1	P2	P1	H5	H6

Design Specification

Input: x,y,cin

Output: s,cout

block diagram:



2-1

Design Specification(2/2)

block diagram and logic diagram(3/3)

I/O pin assignment(5/5)

discussion and conclusion(3/5)

Design Implementation

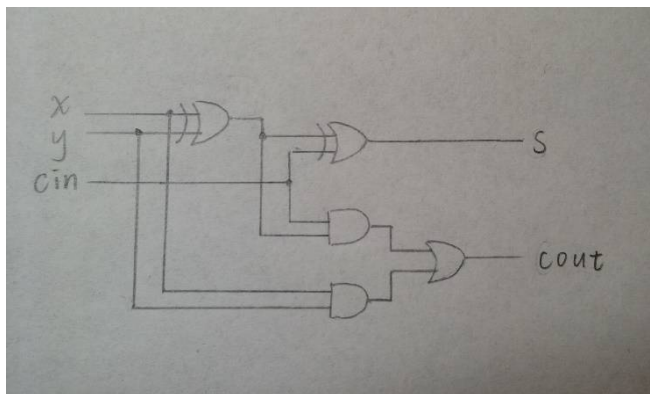
logic function:

full adder $s+cout=x+y+cin$

$$s=(x \oplus y) \oplus cin$$

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

logic diagram:



I/O pin assignment:

NET "x" LOC = T1;

NET "s" LOC = H5;

NET "y" LOC = P2;

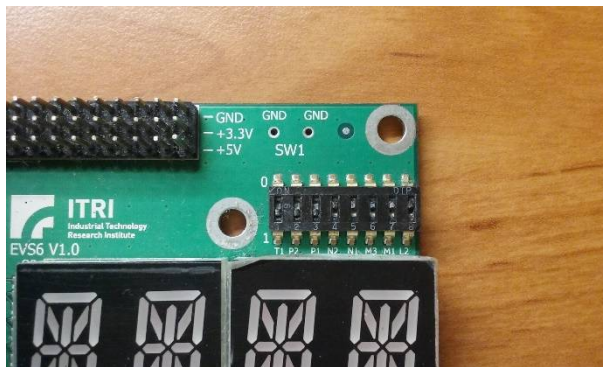
NET "cin" LOC = P1;

NET "cout" LOC = H6;

The Final Result:

X=1 T1down , y=1 P2down , cin=1 P1down

-> cout=1 H6 亮 , s=1 H5 亮



Discussion:

1.當輸出為 1 的時候，LED 會亮。

2 Derive a BCD ($i[3:0]$) to 14-segment display decoder ($D_ssd[14:0]$), and also use four LEDs ($d[3:0]$) to monitor the 4-bit BCD number. (Other values of i outside the range will show F).

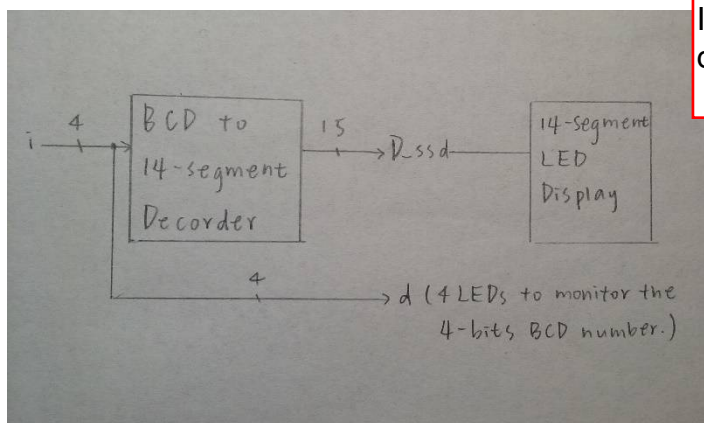
Design Specification

Input: $i[3:0]$

Output: $D_ssd[14:0]$

Output: $d[3:0]$

block diagram:



2-2

Design Specification(2/2)

block diagram and logic diagram(2/3)

block diagram圖中後面的14-segment LEDDisplay應該不用畫，因為 D_ssd 就已是你最後的output

I/O pin assignment(5/5)

discussion and conclusion(3/5)

Design Implementation

logic function:

BCD (i[3:0]) to 14-segment display decoder (D_ssd[14:0])

i=0000 -> D_ssd =0000_0011_1111_111 //0

i=0001 -> D_ssd =1111_1111_1011_011 //1

i=0010 -> D_ssd =0010_0100_1111_111 //2

i=0011 -> D_ssd =0000_1100_1111_111 //3

i=0100 -> D_ssd =1001_1000_1111_111 //4

i=0101 -> D_ssd =0100_1000_1111_111 //5

i=0110 -> D_ssd =0100_0000_1111_111 //6

i=0111 -> D_ssd =0001_1111_1111_111 //7

i=1000 -> D_ssd =0000_0000_1111_111 //8

i=1001 -> D_ssd =0000_1000_1111_111 //9

i=others -> D_ssd =0111_0000_1111_111 //F

use four LEDs (d[3:0]) to monitor the 4-bit BCD number

d=i

I/O pin assignment:

NET "d[0]" LOC = H4;

NET "d[1]" LOC = H6;

NET "d[2]" LOC = F1;

NET "d[3]" LOC = F2;

NET "D_ssd[14]" LOC = P6;

NET "D_ssd[13]" LOC = N4;

NET "D_ssd[12]" LOC = V5;

NET "D_ssd[11]" LOC = T5;

NET "D_ssd[10]" LOC = U7;

NET "D_ssd[9]" LOC = R3;

NET "D_ssd[8]" LOC = N5;

NET "D_ssd[7]" LOC = R5;

NET "D_ssd[6]" LOC = T3;

NET "D_ssd[5]" LOC = T4;

NET "D_ssd[4]" LOC = V4;

NET "D_ssd[3]" LOC = V7;

NET "D_ssd[2]" LOC = R7;

NET "D_ssd[1]" LOC = T7;

NET "D_ssd[0]" LOC = U5;

NET "i[3]" LOC = T1;

NET "i[2]" LOC = P2;

NET "i[1]" LOC = P1;

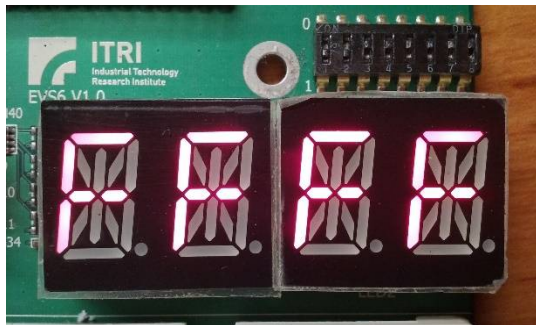
NET "i[0]" LOC = N2;

The Final Result:

i[3]=1 T1down , i[2]=1 P2down , i[1]=0 P1up , i[0]=0 N2up

-> D_ssd =0111_0000_1111_111 顯示 F

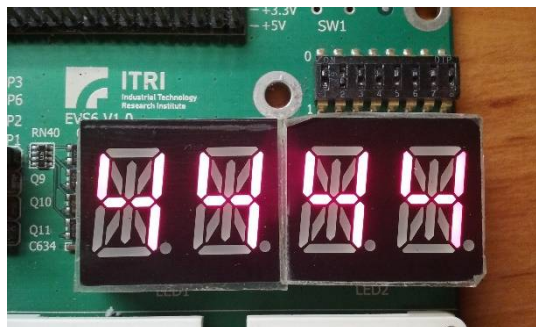
-> d[3]=1 F2 亮 , d[2]=1 F1 亮 , d[1]=0 H6 暗 , d[0]=0 H4 暗



i[3]=0 T1up , i[2]=1 P2down , i[1]=0 P1up , i[0]=0 N2up

-> D_ssd =1001_1000_1111_111 顯示 4

-> d[3]=0 F2 暗 , d[2]=1 F1 亮 , d[1]=0 H6 暗 , d[0]=0 H4 暗



Discussion:

1. i 與 d 之 pin 的設置原本想按照順序, 因輸入錯誤(i 的 3 到 0 位置由左至右, d 的 3 到 0 位置由右至左, 且 d[0]原本要設置在 H5 結果輸成 H4), 導致有些混亂, 雖然顯示結果仍正確, 但較不易觀察。
2. I/O pin assignment 較繁瑣, 需仔細檢查, 本次實驗一開始因 D_ssd 的 pin 輸入錯誤, 導致無法顯示正確數字。

Conclusion:

第一次使用 FPGA 板, 一開始有點搞不清楚如何將打出來的程式碼應用在 FPGA 板上, 甚至燒錄完後還是不知道怎麼測試是否正確, 後來慢慢摸索後才知道一開始打了那麼多奇怪的 pin 都是在設定開關和顯示的位置, 每次都在實驗後

才知道自己在做甚麼，雖然覺得很有趣也很有成就感，但希望以後自己可以更清楚實驗目的和程式碼代表的意義，而不是一昧地照著老師的講義，期待未來我能更獨立掌控自己的實驗!

2-3
Bonus(0/2)