Lab10 Electronic Organ report

106061146 陳兆廷

Exp1: 放出16個頻率的聲音，一個聲音持續一秒鐘。

這題需要用到Lab08的聲音module。我寫了一個新的module(note.v)用來讀取frequency divider的clk\_out，就是一秒鐘的頻率，輸出一個值note\_div，分別為16種不同的頻率，用50000000除以講義給的頻率表上的頻率，還有兩個[3:0]note1、[3:0]note2，用來輸出SSD的音名。將note\_div輸出至lab08之note\_gen.v，接著輸出至speaker\_control.v，再修改ssd.v之輸出七段顯示器，當note1、note2都為0時，分別在第三位及第四位顯示”D”及”o”，即可完成題目之要求。

Exp2:當按下0~9、A~F時，輸出16個頻率的音符，再顯示至SSD。

這個題目需要用到lab09的鍵盤讀取。因此我將lab09之讀取鍵盤的module(keyboard.v)用至這題，keyboard.v當讀取的8位數值剛好為鍵盤的0~9、A~F時，輸出0至15的值note，分別為16種頻率的音，再傳入上題之note.v，輸出note\_div後傳入note\_gen.v、speaker\_control.v，最後傳入ssd.v輸出在七段顯示器上，但我發現一個問題，我在存取USB輸入之值，也就是鍵盤傳入的值時，檢查even parity成功後存取8位數的值，再判斷這個值代表鍵盤上的哪個按鍵，而並沒有一個停止的機制。我再次研究，當鍵盤按下時會有一個訊號，當鍵盤放開時，有一個快速的”F0”會被讀取，接著又會有一個相同的訊號。因此我在讀取鍵盤數值的地方設計一個cnt，如圖(二)，當讀取一個新的值時會+1，因此當讀到F0時將cnt存入，令它要在讀完F0後的一個值停止，如圖(三)，直到下一個值的讀取才會輸出note。State Diagram如圖(四)。接著又有一個問題，顯示我的audio\_lrck等等的電壓為1.8V，與其他的3.3V不同，我查了許久，原來是XDC file裡所有東西都要設置成LVCMOS33，才排除了這個問題。其餘就如同exp1，輸出聲音及SSD即可。

按下A

讀取1C

輸出note = Ra

放開A

讀取F0

讀取1C

輸出note = X

存入cnt

讀取\_\_

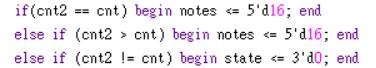
按下\_\_

Cnt + 1

Cnt條件吻合

輸出note

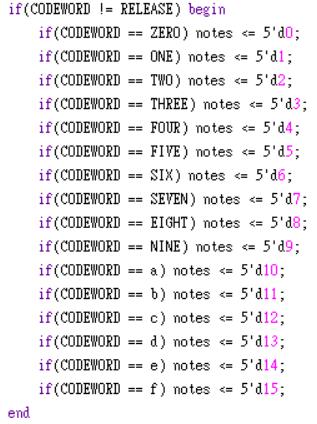
圖(四)



圖(三)



圖(二)



圖(一)

Bonus: 分成左、右聲道，以一個I/O switch控制單音或雙音模式，輸出兩種不同聲音。

這題需要修改Lab8的note\_gen.v，原本輸出audio\_right及audio\_left，現在只需要輸出一個audio\_out，呼叫note\_gen.v兩次，將兩個不同頻率的note\_div輸入後，即可輸出兩個不同的audio\_right及audio\_left。因此這題需要先在keyboard.v新增一個note2，我設計的是當按第一排的按鍵會輸出note，為低音部分的Do~Si，第二排會輸出note2，為高音部分的Do~Si，輸出後傳入note\_gen.v兩次，分別輸出audio\_left及audio\_right，接著在輸入speaker\_control.v，即可輸出雙聲道的電子琴。再將單聲道、雙聲道模式加入，只要在keyboard.v中，加入簡單的if、else即可，當switch是on時，note及note2會不同，當switch是off時，note及note2會相同。



圖(五)

心得：這次作業在製作按下馬上有聲音、放開馬上停止的鍵盤電子琴花非常久時間，因為很難用testbench看讀取的訊號為多少，我不清處它是怎麼傳入訊號的，也很難用顯示的方式顯示出按的東西是甚麼，用了許久及測試了許多方法才慢慢摸索出按鍵輸入的運作模式，我認為我跑了bitstream至少跑了50次。雖然真的用了很久，但是自己找出來真的非常開心，也從只能一直發出聲音的電子琴變成可以用按下及放開控制聲音的電子琴。