

Lab 4

學號: 109062202

姓名: 陳禹辰

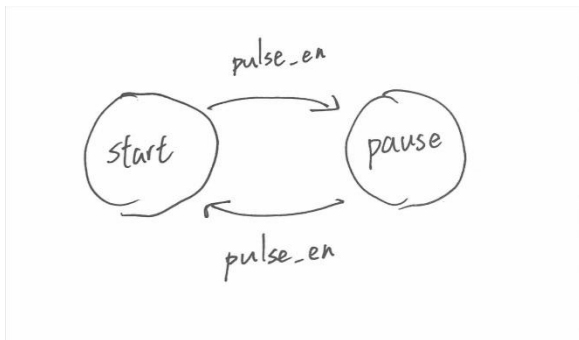
1. 實作過程

Lab4_1 :

我是把 7-segment 從左到右編號 BCD3, BCD2, BCD1, BCD0 四個來看。因為右邊兩個是負責顯示 counter 而左邊兩個則是 BCD3 負責顯示 speed state 而 BCD2 負責顯示向上或向下數，而且兩個部分所需要的 clk 也不同，因為一個是要再 0.5, 1, 2 秒的時候變換，而另一個則是要較快去更新他的 state 或是去判斷有沒有按下按鈕，所以我就把這兩個部分分開成兩個 sequential block，其中 counter 的 clk 是會根據 speed state 而有所變化的。

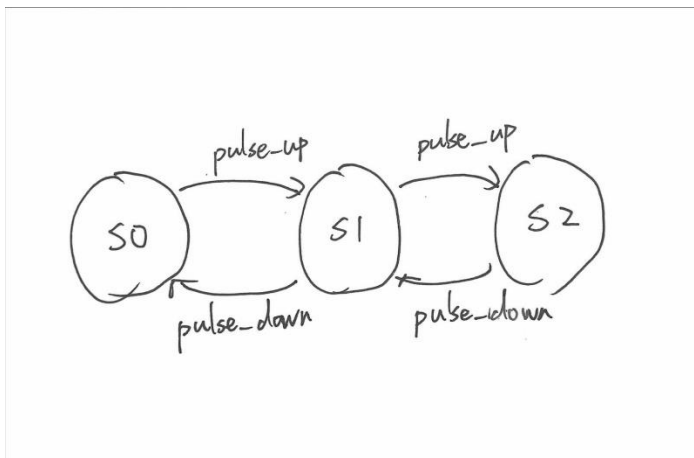
首先是 BCD3, BCD2 的部分我用兩個 combinational block 來判斷，並 assign 值到 next_BCD3 跟 next_BCD2 上，再用 sequential block assign 到 BCD3 跟 BCD2 上。

一個用來判斷是暫停或開始，如果按了 pulse_en 他就會在暫停跟開始兩個 state 之間轉換。

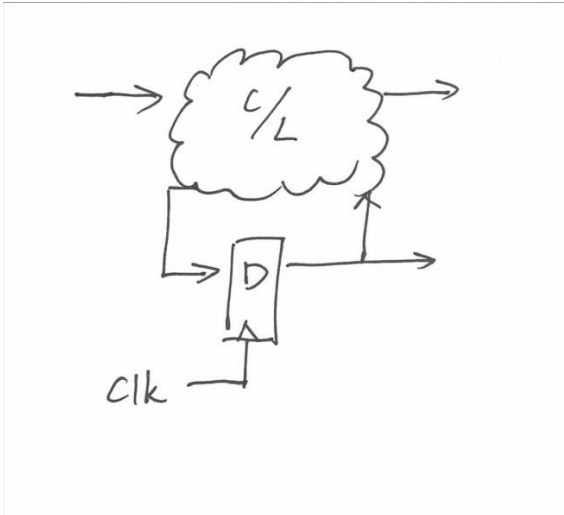


還有判斷他現在是要往上或往下數，如果按著 debounce_down 那就會變成往下數而 next_BCD2 的值就會變顯示往下符號的值，然後因為題目要求的是壓著不放的時候才會往下，所以這裡的 input_down 不需要經過 one_pulse 的處理。

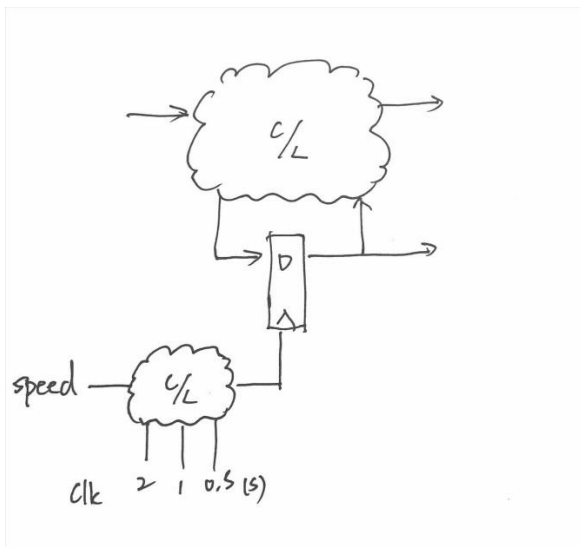
另一個則是用來判斷 counter speed 如果按了 pulse_up 就變成速度較快的 speed，反之若按了 speed_down 則 speed 會變慢，而 next_BCD3 也會再 speed state 改變的時候 assign 值。



然後用 sequential block 去更新在 speed 跟開始的兩個狀態，以及 BCD3, BCD2 所要顯示的燈。



而在另一個 sequential block 的部分，因為 speed 有可能會有變化所以 clk 要根據前面的輸出來決定，這邊我是有三個 module 然後一直輸出三種 clk 的值，然後再判斷要用哪個值。



至於 counter 的部分我則是把兩個燈看成是十位數跟個位數然後個位數一值變換到邊界值的時候再更新十位數，一直到數到臨界值為止。這邊舉例往上數得部分。

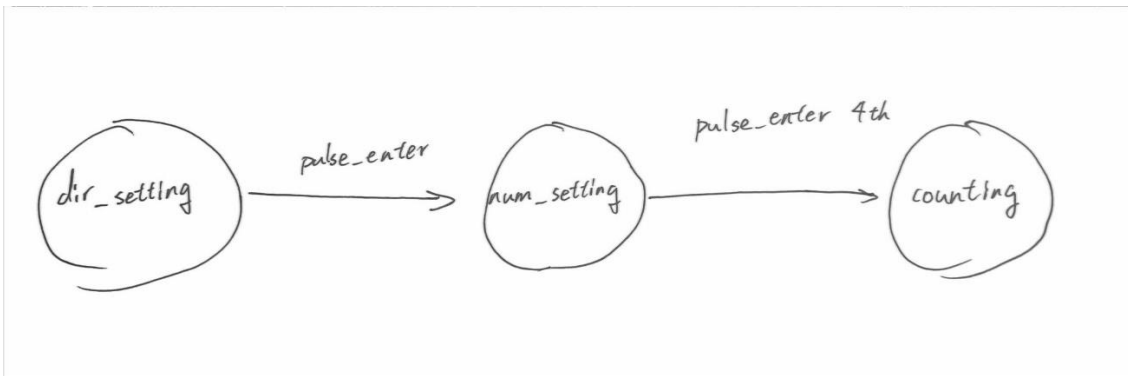
```
if(BCD0 == 4'd9)begin
    if(BCD1 == 4'd9)begin
        next_BCD0 = BCD0;
        next_BCD1 = BCD1;
    end else begin
        next_BCD0 = 4'd0;
        next_BCD1 = BCD1 + 4'd1;
    end
end else begin
    next_BCD0 = BCD0 + 4'd1;
    next_BCD1 = BCD1;
end
```

就是 BCD0 一直加一直到 9 的時候，下一個會變成 BCD1 加一而 BCD0 歸 0，然後數到 BCD1 = 9 且 BCD0 = 9 的時候就會停下來了。

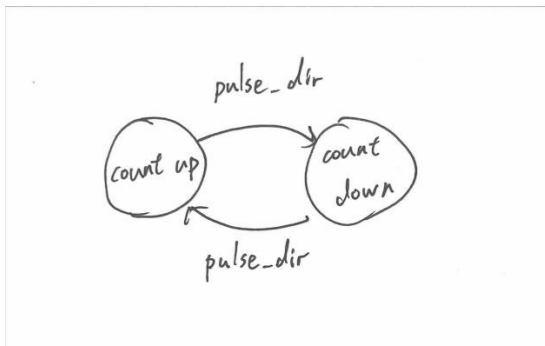
Lab4_2 :

Lab4_2 我也是寫兩個 sequential block 因為在不同的 state 燈的更新速度會不一樣，例如在 num setting state 跟 counting state 時就不能用相同的 clk，所以我就一個負責更新各個 state 的變化一個負責更新 BCD 燈的變化，而處理 BCD 的我則會用 clk_change 當作 clk，而 clk_change 則會因為處在不同的 state 有所變化。

我把整個分成三個主要的 state 分別是 dir setting, num setting, counter 三種，在 dir setting state 按下 pulse_enter 時會進入 num setting state，而在 num setting state 按下第四次 pulse_enter 時則會進入 counting state。



首先是 dir setting state，如果按下 pulse_dir(input 的 count_down 經過 debounce, one pulse)就會改變他數的方向，然後 led0 會在往下數的時候亮，往上數的時候暗。



接著是 num setting state，因為要紀錄這邊的值所以我用了 mem0-3 來紀錄每個 BCD 的值。接著如果按下 pulse_inputnum 則會改變當下那個 BCD 所顯示的數字，然後每個所能顯示的最大值不太一樣，BCD3 只能顯示 0 或 1，BCD2 則是 0 到 5，BCD1 跟 BCD0 則是 0 到 9 都能顯示，然後按下 pulse_enter 則會轉換可以改的 BCD，順序則是從左便往右邊。然後我用了一個 entertimes 來記錄 pulse_enter 按的次數。

在按第四次的時候，我會先判斷在前一個 state 所設的是往上數還是往下數，如果是往下數那我就讓它目標是 0 然後在 counting state 時就從紀錄的 mem 值開始往下數。而如果是往上數則讓他的目標是 mem 值，然後從 0 開始數到紀錄的 mem 值為止。

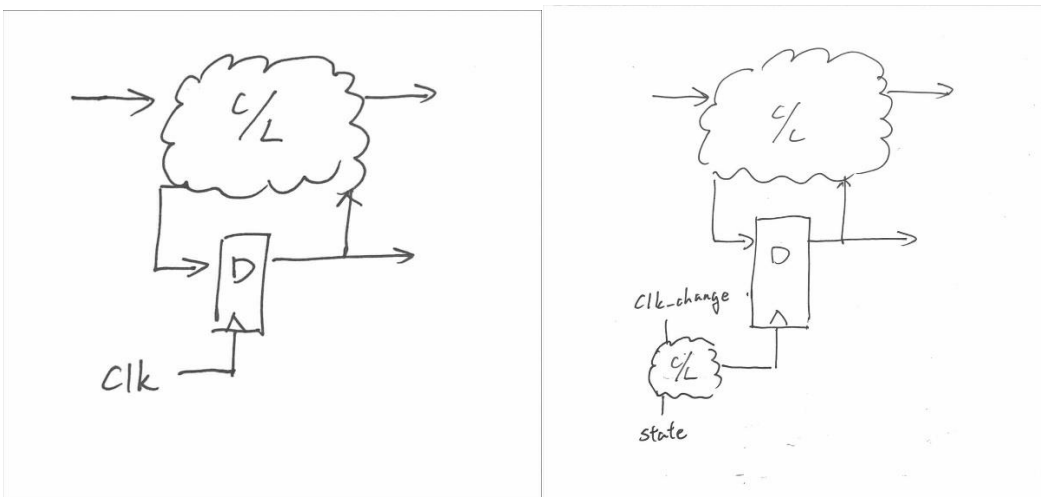
```

if (counting_state == counting_up) begin
    next_BCD0 = 4'd0;
    next_BCD1 = 4'd0;
    next_BCD2 = 4'd0;
    next_BCD3 = 4'd0;
end else begin
    next_BCD0 = mem0;
    next_BCD1 = mem1;
    next_BCD2 = mem2;
    next_BCD3 = mem3;
end

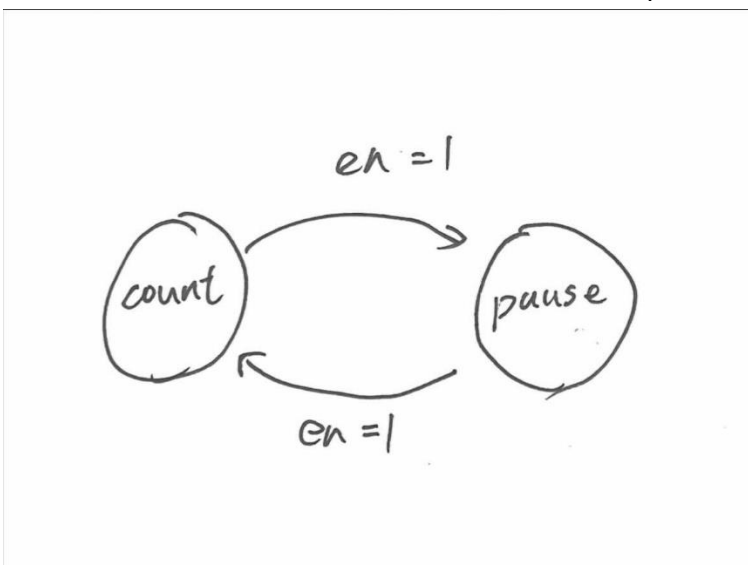
```

而上面這三個 state 我都是寫成 combinational block 總共有兩個，一個負責 BCD 的值，一個負責 counting direction 在 state 是 dir setting state 時。

然後 sequential 則是兩個，一個是處理 BCD 的燈變化(clk 值會根據 state 改變)，一個則是處理 state 的變化或其他值的紀錄(entertimes, mem0-3, etc)。



在 counting 時，若果是暫停的狀況那他就會維持原本的值，反之則會根據 dir 來往上數或往下數，而這邊的 en 不用經過 debounce 或是 one pulse 因為輸入用的是 slide 所以可以直接用。



然後我 counter 寫的方法跟 lab4_1 是差不多的，就是分成四個位數來看，然後在每個位數的邊界值時，讓下一個位數做改變，只是這邊因為 BCD3 跟 BCD2 一個最多到 1 一個最多到 5 所以就稍微注

意一下就可以了。

2. 學到的東西與遇到的困難

Lab4_1 :

我遇到的第一個大問題是 speed 會有問題，例如可能在 2 秒的 speed 但是會第一個 2 秒數一次然後下一個 2 秒數兩次，我後來發現是我 clk 的判斷會有問題，因為在 2 秒的 clk 是 1 的時候 0.5 秒的 clk 會觸動兩次，然後因為我的寫法在判斷式時會讓 0.5 秒的 clk 影響到所以才會造成這個問題。

下一個問題是 light control 的 clk，因為要在一個剛好的速度，不能太快或太慢，才能讓四個燈因為視覺暫留而看起來像是四個燈同時在亮。

最後一個問題則是 BCD3 也就是顯示 speed 要等於多少的，我在 speed change 時，因為判斷沒有寫好，所以導致 speed 沒辦法加到 2 或是只能往上加或是只能往下減之類的各種奇怪情況。

Lab4_2 :

這部分我遇到最大的問題是，一開始沒有把 BCD 跟其他的 state 分開寫，然後讓所有東西的 clk 一起改變，但是這樣就會遇到有問題的狀況，就是如果 state 改變了然後要 assign 在下一個 state next_BCD 的初始值=0 的話就會有問題，因為 clk 也改變了所以就會造成他抓不到上一個 state 要丟過來的值。

然後學到的東西就是 clk 的值在不同的 state 要怎麼去做更改會比較方便，然後 7-segment 顯示的操作跟按鈕的使用。遇到的最大困難大概就是 debug 了，有時候會一直盯著一個部分找問題，結果過了幾個小時才發現問題根本不是出在檢查的那一塊...。

3. 想對老師或助教說的話

因為這次 lab 時間拉得比較常所以就比較能提早開始做，時間上比較充裕，也比較不會跟其他科的期中考時間卡在一起。然後辛苦助教幫忙 demo 跟解決我們的問題了，也謝謝老師辛苦上課跟處理我們的疑問。

附上兩個小小的笑話以及可愛貓貓圖：

地瓜拍照會變什麼？

地瓜葉(Ya)

地瓜被抓去關會變什麼？

地瓜球(囚)

