

Lab 7 report

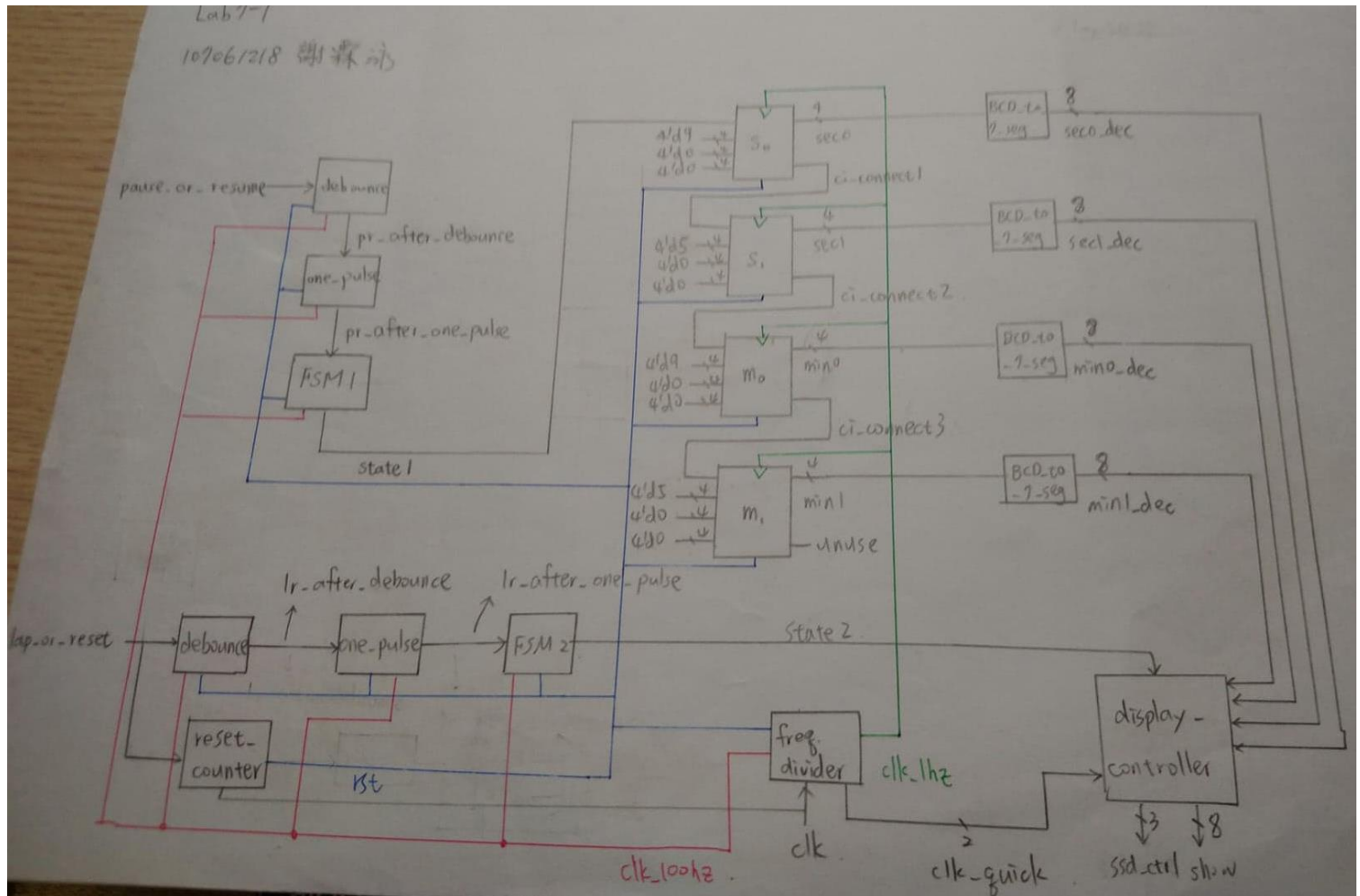
107061218 謝霖泳

1.

➤ Design Specification

Input: clk, pause_or_resume, lap_or_reset

Output: [3:0] ssd_ctrl, [7:0] show, p



➤ Design Implementation

本設計的輸入除了 clk 以外，還包含兩個按鈕的輸入。其中一個為控制碼表的暫停或繼續，名為 pause_or_resume，另一個名為 lap_or_reset，其功能為短按時，控制螢幕顯示的數字要不要凍結，長按 3 秒以上為 reset。在這邊要注意的是，螢幕上顯示的數字是否凍結與現在碼表真正時間有沒有在跑並沒有關係，凍結與否與暫停繼續係為兩個不同的 FSM 所控制也就是我的 block diagram 中的 FSM1 與 FSM2。

其中，pause_or_resume 按鈕的訊號與 lap_or_reset 按鈕訊號一樣照慣例進入一組 debounce 和 one pulse，並分別進入 FSM1 與 FSM2。值得注意

的是，lap_or_reset 信號除了經過上所提及一般按鈕的處理外，也會經過一個 reset counter，用來計算我已經連續按下這個按鈕多久的時間，在這個 reset counter 的 module 中，包含了一個 counter 與一個 MUX 的構造，當我按住的秒數達到三秒以上時，rst 訊號便會輸出 1'b0，讓整個 design 得以 reset，若連續按住時間未達三秒，則 rst 訊號輸出為 1'b1，也就是不會 reset。

FSM1 出來的結果控制現在要不要數，並傳入最低級的 single bit up counter，作為最低級位數的 count enable。在途中，可以看到 4 個 up counter 層層相接，上數的機制與之前多次 lab 均相同，在此不再贅述。這 4 個位數出來的 4-bit binary value 分別經過 4 個 BCD to 7-segment decoder，得到他們分別的 decode 結果。傳入 display controller 中。

在 display controller 中，設有一組 FF，用來記錄應該顯示的值。若 FSM2 出來的結果告訴我現在要 freeze 我的顯示，那麼，這組 FF 儲存的值會停在我現在的值並保持不變，換句話說，不會讓新的值傳進來。反之，若 FSM2 的結果告訴我不要 freeze，那麼，這個 FF 的值就會一直更新成我現在的時間。有了我現在應該顯示的值後，便利用之前視覺暫留的技巧將他們正常的顯示到七段顯示器上面。

➤ Discussion

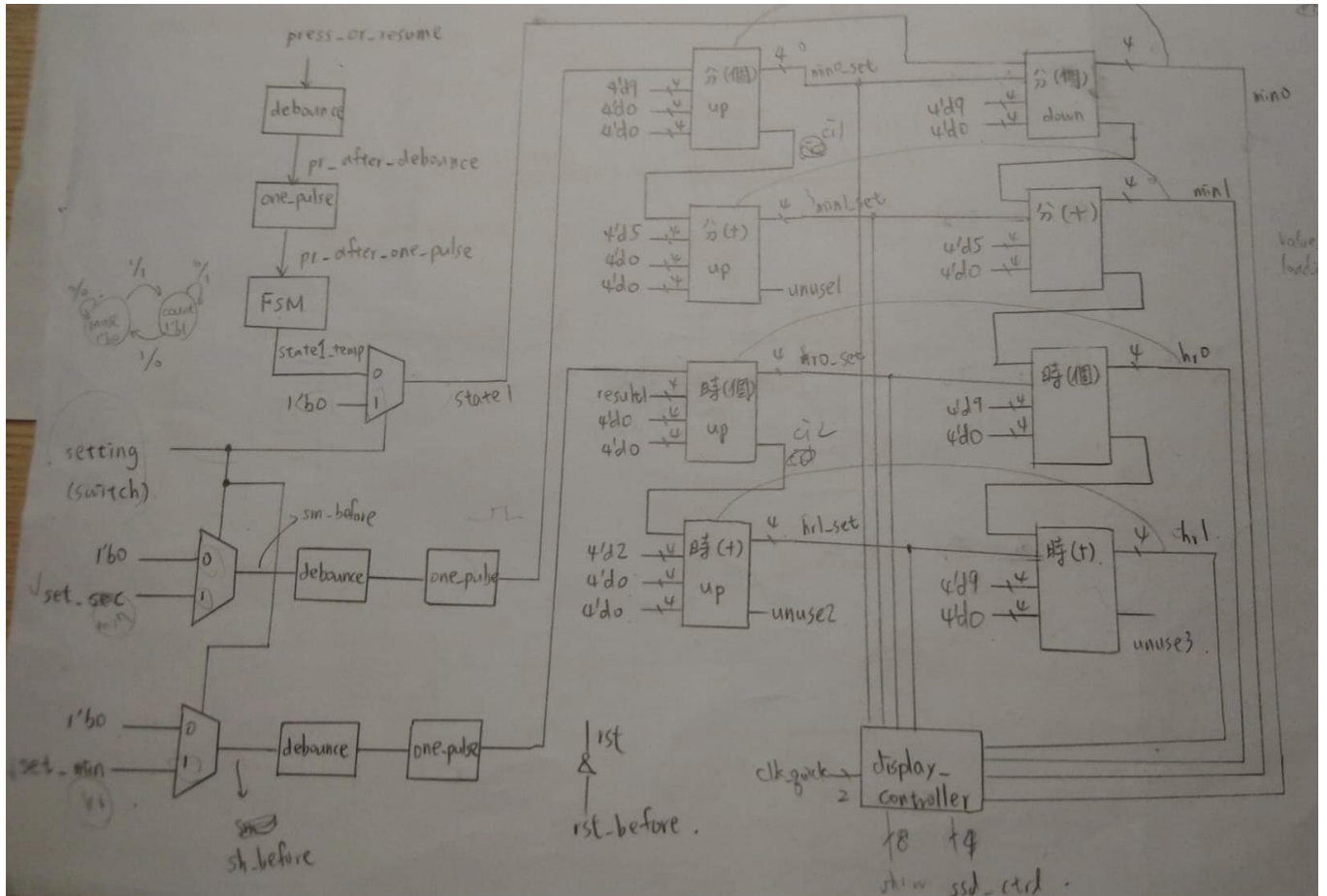
當我將它燒到板子上後，便執行了預期中的功能，而且，當我將 lap 那顆按鈕長按超過 3 秒後，顯示的時間便會回到 0 分 0 秒，也就是成功達成「長按 reset」的功能。

2.

➤ Design Specification

Input: setting, clk, rst_before, set_min, set_hr, pause_or_resume

Output: [7:0] show, [3:0] ssd_ctrl



➤ Design Implementation

依照題目要求，在上述幾個 input 中，setting 這個 input 為 switch，用來控制我現在能不能設定倒數計時器的時間。當我們將該按鈕剝起來，也就是這個訊號為 1'b1 的時候，始可設定時間。另外兩個設定相關的按鈕為 set_min 以及 set_hr，可以看到他們進入 debounce 以及 one pulse 之前有先經過一個 MUX，而 MUX 的 selection 就是剛剛所提及的 setting 輸入。這個用意就是，當我的 setting=1'b1 時，這兩個設定時間的訊號才會傳入 debounce 及 one pulse，若現在按鈕沒有被撥起來，選擇 0 進入後方 debounce 及 one pulse，也就是當我沒有允許設定時間的時候，不管這兩顆按鈕怎麼按都不會有用，因為進入 debounce 及 one pulse 的訊號都是 1'b0。

在 block diagram 的中右側，可以看到兩排的 single digit counter，現在分別介紹它們的功能。左邊那 4 個 single digit counter 為 up counter，其功能為設定時間用，也就是當我按一下 set_min，分鐘值就+1，同理，當我按一下 set_hr，小時的值就+1。

右側那 4 個 single digit counter 為 down counter，用來倒數計時，而其初始值就是我剛剛在前面所設定好的時間，而下屬的方法與上數雷同，而且前面 lab 的 bonus 已經做過 30 秒的倒數計時，因為概念重複，所以在此亦不再贅述。

最後，將目前的數值送入 BCD to 7-segment decoder，得到 decode 的結果，並傳入 display controller 中並傳到七段顯示器上顯示出來。

➤ Discussion

這次 lab 讓我對於多功能電子鐘有了更深入的認識，在學玩 speaker 的設計後，應該能多出一個鬧鐘的功能，就是到達設定的時間時，會從 speaker 的小板子中傳出應有的音調。

3.Conclusion

這次的 lab 花了我比較多時間，主要是在思考如何將數字凍結，也就是第一題如何讓時間繼續流但是數字卻停在剛才按下 lap 按鈕時的時間。後來的解決方法是在我的 display controller 中加入一組 FF，頻率為 100MHz，也就是以快速的頻率去抓我現在七段顯示器上的值，當我按下 lap 後，便不再讓新的值流進來，就達到了凍結數字卻讓時間得以繼續數的功能。