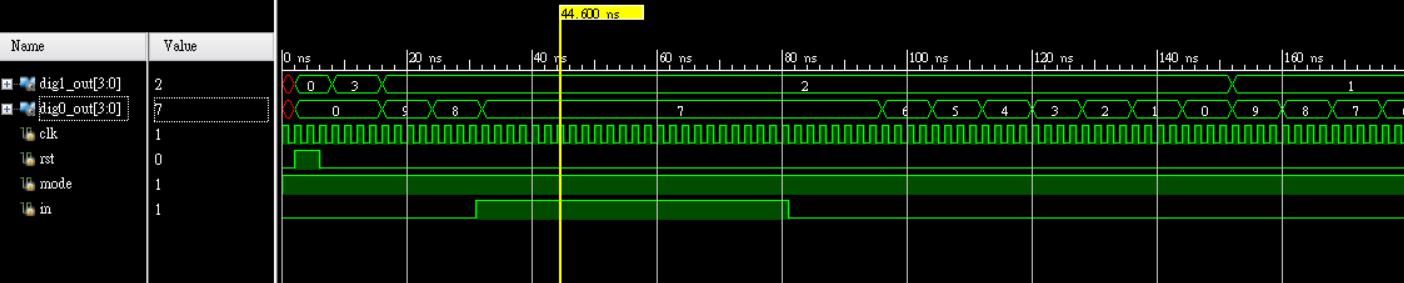
Lab05\_stop\_watches\_106061146\_陳兆廷

Pre-labs:

* 1. 計時器需要：一個input clk，一個input rst，一個input enable(控制暫停或開始計時)，還有一個mode來控制15或是30秒倒數計時。Output有輸出的digit1及digit0，還有題目需要的16-bit Led。
  2. 一張含有 文字, 白板 的圖片

     描述是以非常高的可信度產生 當en開始時，digit0會先作用，等到digit0 == 0時，digit1的en會開始作用，即digit1少一位，少一位的同時digit0會回到原本的極限值(ex. 9)而繼續開始digit0的運算。而當digit1 == 0 && digit0 == 0時，led會全部變為1；按rst時，會依照輸入的mode選擇回到15或是30秒。
  3. 一張含有 文字, 白板 的圖片

     描述是以非常高的可信度產生這題的FSM需要至少有兩個state，state 01代表運算中，state 10代表freeze(pause)，區分方法就是in\_latch的長短，而再設一個state 00 代表reset時的狀態，只要按超過三秒(in\_latch == 3’b111)，或是按rst就會跳到state 00。最後輸出count\_enable和freeze\_enable來顯示現在要進行的動作。



**In == 1時，暫停在27**

Experiments

* 1. 一張含有 螢幕擷取畫面 的圖片

     描述是以高可信度產生完成exp1除了pre-lab1之外還需要一些東西，需要一個大module將整個程式涵蓋住，並增加輸出SSD的小module，將FSM的count\_enable接到stopwatch的en，輸出的digit0以及digit1要經過ssd的轉換再輸出。

1. 一張含有 文字 的圖片

   描述是以非常高的可信度產生
   1. 第二題要將rst和pause(freeze)功能合併成一個按鍵，這時需要先從FSM來修改，我在FSM新增一個output叫做rst\_enable，並把input的rst刪除，當需要reset的時候就把rst\_enable打開，也就是在state 01(count)的時候，只要按超過三秒(in\_latch == 3’b111)時，rst\_enable = 1，並回到初始狀態state 00。這時候在大module(stopwatch\_disp)裡面，把所有的rst用經過FSM輸出的rst\_enable代替，再將input的rst刪除，就可以達成rst跟pause在同個按鈕，當短按時是pause，長按時是reset的功能。
   2. 要做到顯示1:00的倒數計時器，要先從stopwatch裡面修改，新增一位digit2，當其中一個mode時顯示1:00，另一個mode顯示0:30。其餘就跟exp2裡一樣。最後在輸出的時候，加入dig2\_out就可以了。