**Lab 3: Counters**

**106061218 李丞恩**

**Design Specification**

**0. pre-lab: 4-bit synchronous binary up counter.**

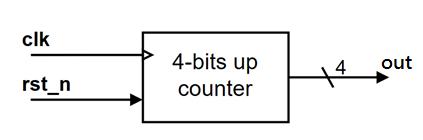
(1) input/output

Input: clk (global clock).

Input: rst\_n (Reset signal, can also be viewed as a switch).

Output: out[3:0].

(2) Block Diagram



**1. Frequency Divider**

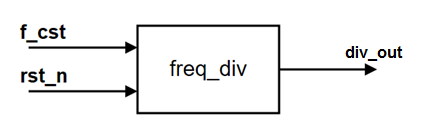
(1) input/output

Input clk (global clock).

Input rst\_n (Reset signal).

Output div\_out (The approximately 1 Hz signal output).

(2) Block Diagram



**2. 1Hz Frequency Divider**

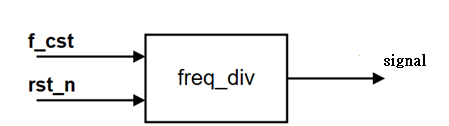
(1) input/output

input clk (global clock).

　 input rst\_n (Switch of two FFs).

output signal (real output).

(2) Block Diagram



**3. Single digit BCD up counter**

(1) input/output

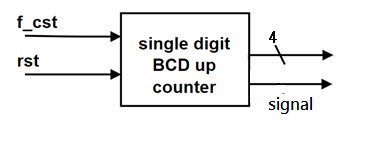
input clk

input rst\_n

output [3:0] out

output signal (1Hz Clock)

(2) Block Diagram

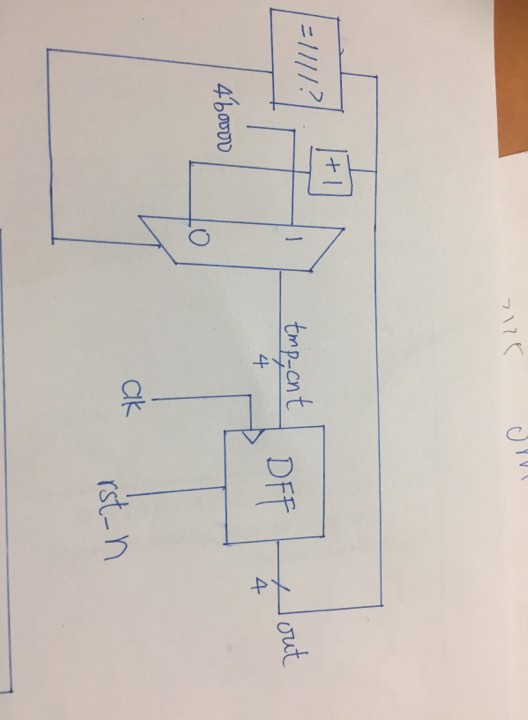
****

**Design Implementation**

**0. pre-lab**

(1) Logic diagram

我的想法是讓經過正反器的訊號不斷加一，直到加到15就歸0。因此需要一個與15的比較器與一個多工器。



(2) Verilog verification

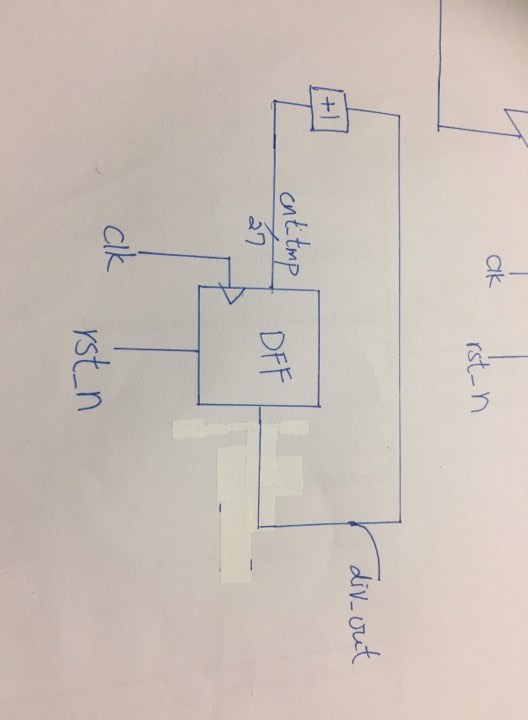


**1. Frequency Divider**

(1) Specification

　　設定一個27位元長的訊號，因為2的27次方大約就是1億(正確來說是1億3千萬左右)，然後讓溢位自然發生。因此第一個訊號就會以略小於1Hz的頻率在0與1之間反覆。

(2) Logic diagram

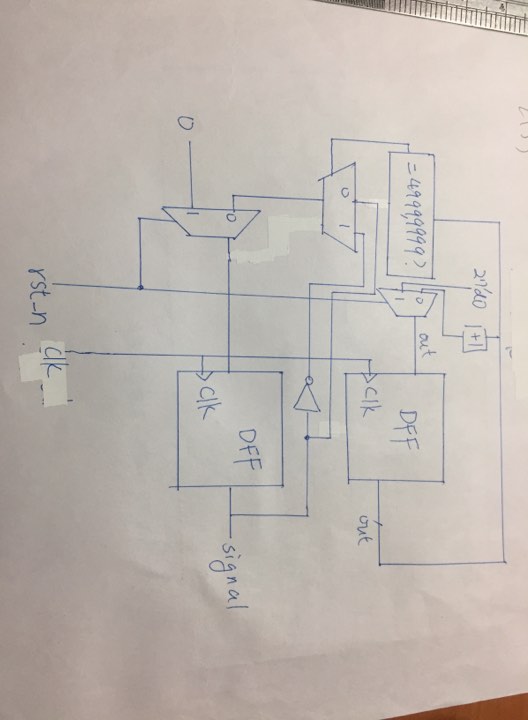


**2. 1Hz Frequency Divider**

(1) Specification

把上一題的概念作一些修改，只要27bits的信號與49999999相等時就將它歸零，並產生一個信號，相等時為1，不相等時為0。並引進另外一個信號signal，用前述的訊號觸發它的改變，則signal這個訊號就會恰好有1Hz的頻率。

(2) Block diagram

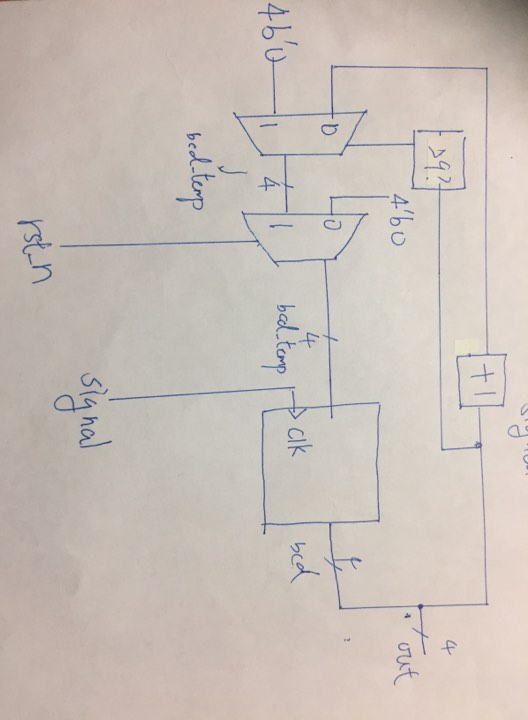


**3. Single digit BCD up counter**

(1) Specification

這一題就把prelab跟第二題合併起來就可以了。把第二題產生的signal作為prelab的clock，並把prelab的4個output接到LED上就完成了。雖然題目沒有規定，但我還是把signal這個訊號接到另一個LED燈上，方便觀察與檢查上數計數器的頻率。

(2) Block diagram



**Discussion**

寫prelab時可說是吃足了苦頭。因為我還不熟悉always敘述對於組合邏輯和循序邏輯的不同。經過助教指導以後才知道只要()裡面有clock，就是循序邏輯，能動的只有FF的輸出；反之如果()裡面是其他訊號就只是組合邏輯，這時才可以去給FF的輸入賦值。同時也要注意會不會讓訊號卡住。

第一題的除頻器還算好做，但第二題我原本畫的電路太複雜了，導致有bug也不知從何檢查起。後來助教建議我先不要畫圖重想一次，發現原來有更簡單的邏輯。從這個經驗我學到寫電路模擬的code比畫圖起來還是比較自由一點，用程設時的流程圖轉化為電路圖也是一個方法。

順帶一題我這一周真的太忙了，下一周還有期中考，因此第四題會晚一點才補交。可能會到禮拜一的班去補做。

**Conclusion**

1. always裡帶clock就是循序邏輯，只可以處理正反器的輸出；如果()是其他訊號就可以當成組合邏輯，可給正反器的輸入賦值。

2. 先把流程想好再畫logic diagram。

**References**

上課的講義

《Verilog硬體描述語言數位電路》，鄭信源，儒林圖書公司(2016)。