

Lab 3: Counters

106061218 李丞恩

Design Specification

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

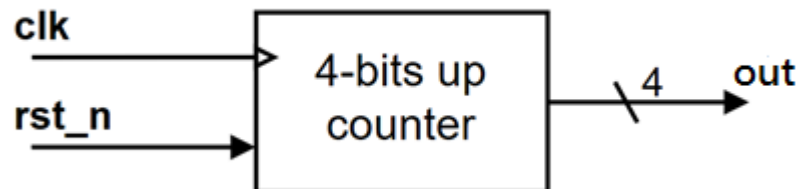
(1) input/output

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

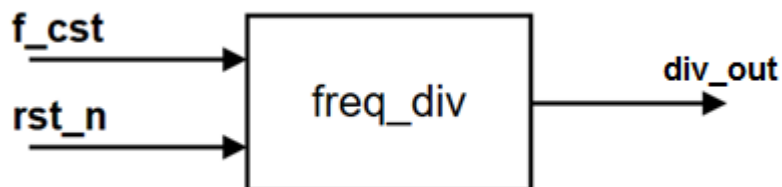
(1) input/output

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

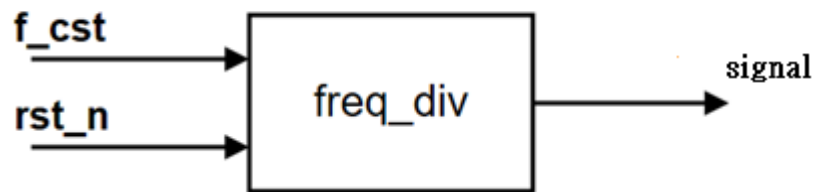
(1) input/output

input clk (global clock).

input rst_n (Switch of two FFs).

output signal (real output).

(2) Block Diagram

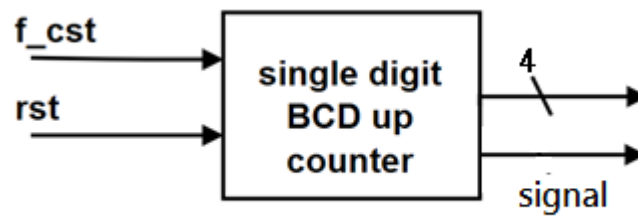


3. Single digit BCD up counter

(1) input/output

input clk
input rst_n
output [3:0] out
output signal (1Hz Clock)

(2) Block Diagram

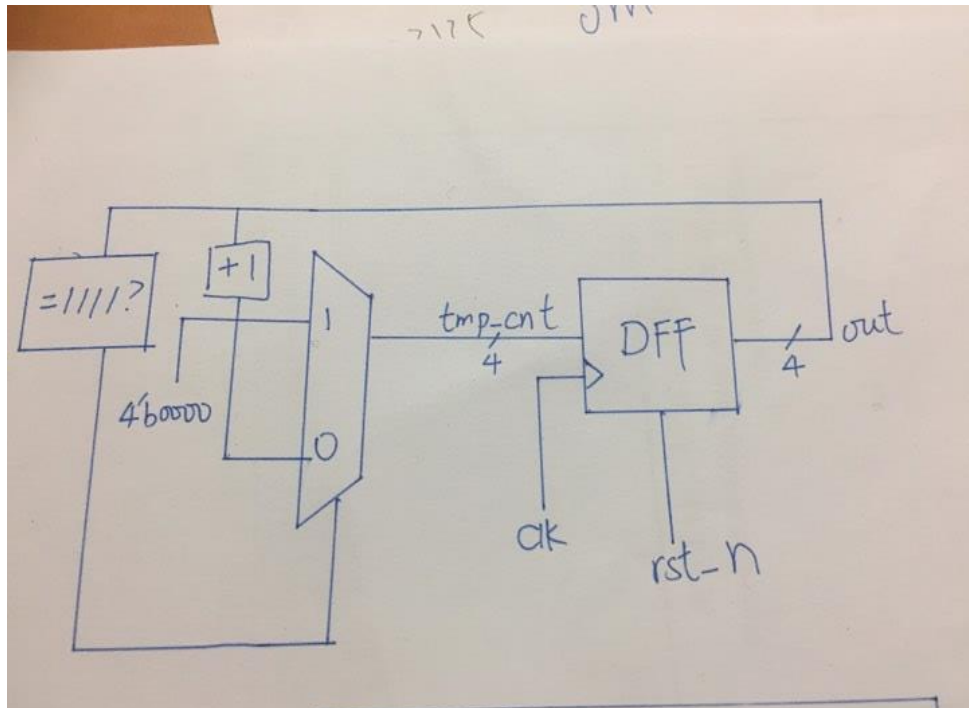


Design Implementation

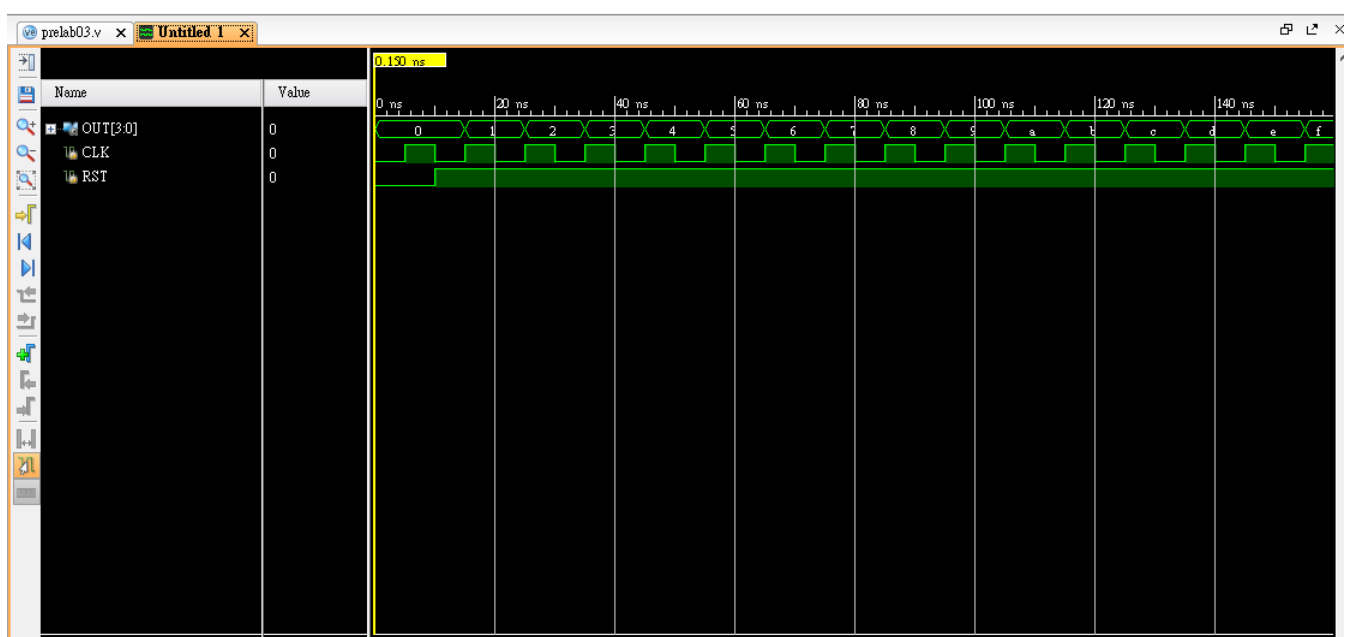
0. pre-lab

(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

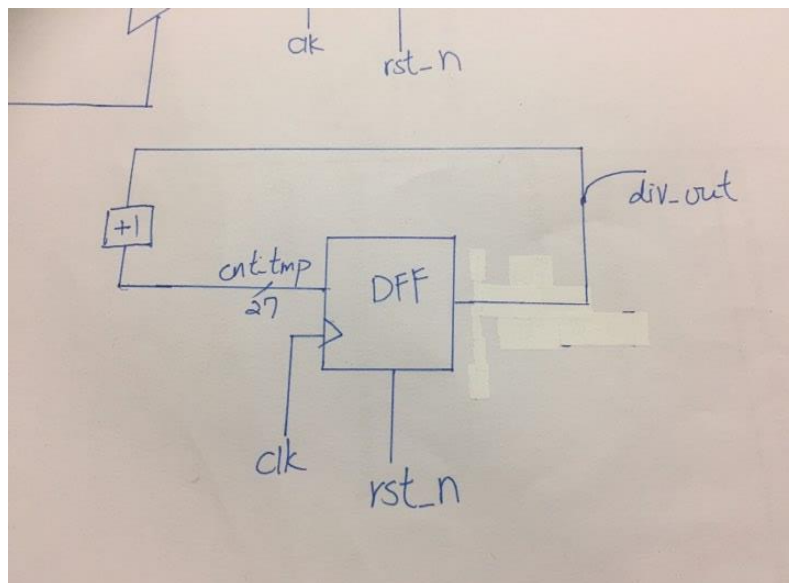


1. Frequency Divider

(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

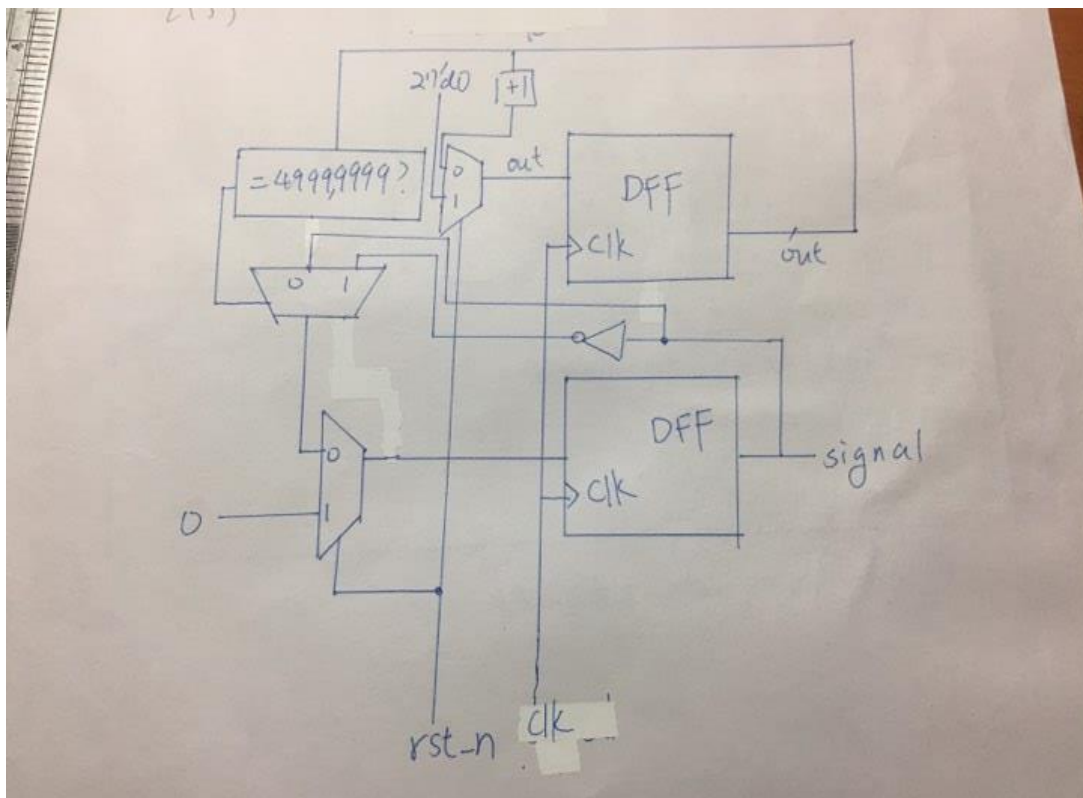


2. 1Hz Frequency Divider

(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

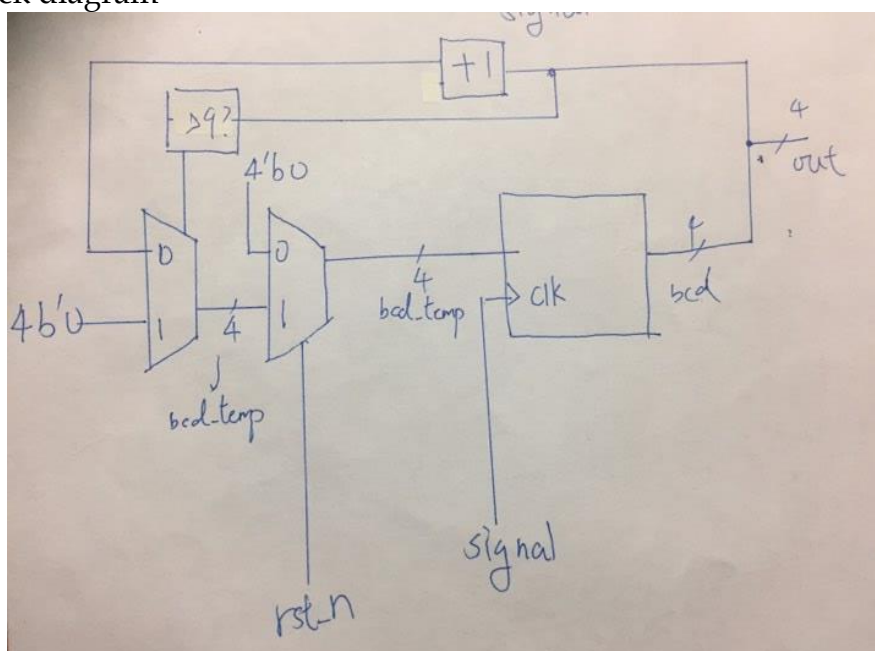


3. Single digit BCD up counter

(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)。