# Sequential Logic(循序邏輯)

## -計數器電路設計

(Behavior Modeling-II)

電子實驗

組別:第三組

系級:資訊二乙

組員 1:10927202 陽彩柔

組員 2:10927207 蒲品憶

### 組員 3:10927248 連翊安

## (1)程式碼

#### Counter

```
module Counter( clk, rst ,dout) ;
     input clk ;
     output reg[3:0] dout;
 3
 4
      input[3:0] rst;
 5
     always@(posedge clk) begin
        if (rst==0)
 6
 7
           dout <= 4'b0000 ;
8
         else
9
           dout <= dout + 4'b0001;
10
     end
11
    endmodule
```

#### TM

```
module TM;
1
 2
 3
                    clk;
     reg
 4
      reg[3:0] rst;
 5
     wire [3:0]
                     dout;
 6
     parameter t = 200;
parameter th= 100;
 7
 8
      parameter
                    th= 100;
 9
      Counter U_cnt ( .clk(clk), .rst(rst), .dout(dout) );
10
11
12
     always #th clk=~clk;
      initial begin
13
14
       rst = 1;
15
        clk =0 ;
16
        #t rst = 0 ;
17
        #t rst = 1 ;
18
19
        #(t*14)
20
21
22
       #t $stop;
23
      end
24
     endmodule
```

## (2)模擬 waveform

說明:當 rst 為 0 時,dout 會初始化為 0;當 rst 為 1 時 dout = dout + 1。

## (3)心得

組員 1(陽彩柔): 這次的實驗難度和以往的不太一樣,有比較多地方是需要自己思考的,就算將 ppt 的 code 打上還要加上別的東西,一開始我們出現很多error,就得要去思考原因並解決,就花了我們一段時間,原以為經過的時間 t 要一行一行打,但在助教的幫助下知道了時間可以用 t \* 14 的方式,減少重複的程式,並且看起來更簡潔易懂。

組員 2(蒲品憶): 這次實驗了解 alway@裡面的參數,<= 左邊被賦值的參數不能為 output,改善方式是將參數改成 output reg 才能被使用,這部分老師上課沒有特別說明,實做過後才發現這次 Lab 沒這麼簡單。還有一個小瑕疵,是當時我們在 TM 每過 t 時間讓 rst 加 1,但是其實沒必要,因為 rst 只要不是0,dout 會在每過 t 時間後加 1,所以我們只要為 rst 設定一個大於 0 的定值即可。

組員 2(連翊安): 這次的 lab 很多的 code 是需要自己去思考的,不像前幾次看 ppt 就可以很快打出來,在編譯的時候會出現 error,要想為什麼型別錯誤,並且去修改,還有給助教檢查的時候知道了時間可以用 t\*14 的方式,減少重複的程式,感覺對 verilog 更加熟悉了。