



*Department of Computer Science and Information Engineering*

*National Cheng Kung University*

# **LAB - 8**

陳培殷

國立成功大學 資訊工程系



---

# Outline

- Video preview for 晶片實現 + HDL介紹
- Lab I--Moore machine (sequential circuit)

# Lab I -- Moore machine (1/3)

- 完成一個Moore machine，其 I/O 與 state 變化如下：

- 頻率1HZ

- Pause 為1則狀態不變

- Pause 為0則依右邊狀態表為主

- 一開始進行Reset，將目前狀態給S0

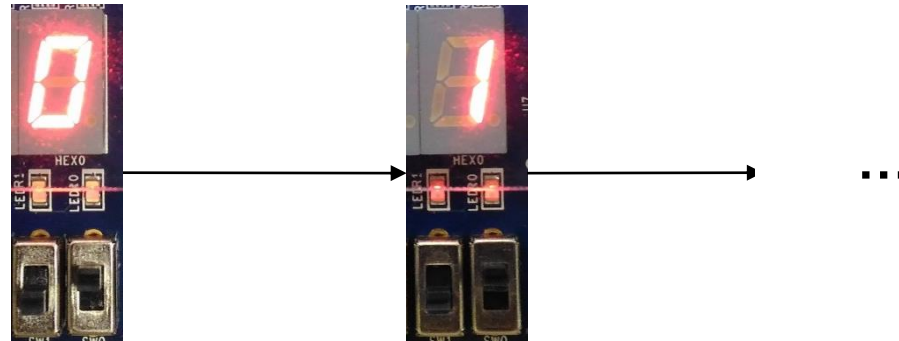
目前狀態 (current-state)	下一個狀態 (next-state)		七段顯示器 輸出 (output)
	In=0	In=1	
S0	S0	S1	0
S1	S0	S2	1
S2	S3	S4	2
S3	S4	S3	3
S4	S1	S5	4
S5	S2	S5	5

## Lab II -- Moore machine (2/3)

- 畫出 FSM、說明運作過程 (輸出結果會根據你目前的狀態來決定，例如：S0 輸出 0、S1 輸出 1 ...)
- 以七段顯示器顯示輸出

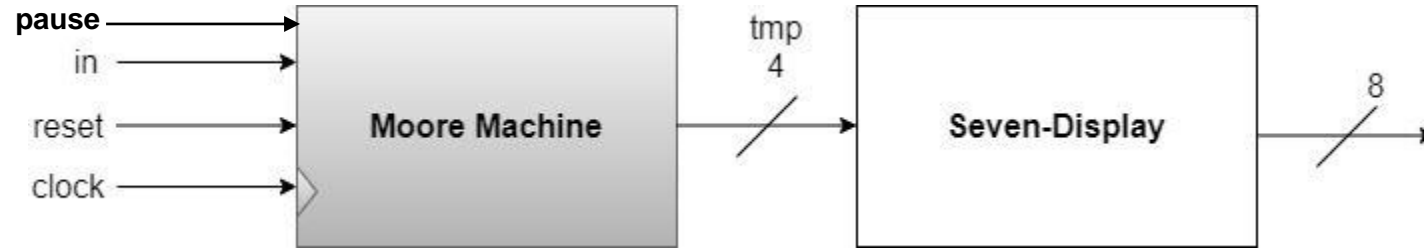
**Input : clock(Clock\_50), pause(SW0), reset(RESET\_N), in(SW1);**

**Output : out(HEX00~HEX06)**



# Lab II -- Moore machine (3/3)

## ■ 系統架構圖



## ■ module Moore machine (sequential circuit)

- 計算計數器正確的值

## ■ module Seven\_Display (combinational circuit)

- 將Moore machine的輸出值轉為output的七段顯示器

## ■ 兩個module之間溝通方式

- `SevenDisplay DIS (.data(tmp), .out(out));`

# Lab – Hint(1/2)

## ■ 除頻器

- FPGA的時脈頻率為 50 MHz

  - 表示每秒可以執行  $50 \times 10^6$  次

- 現假設要求變換頻率為 1 Hz

  - 表示每秒變換1次

- 但因為時脈頻率無法改變，所以需要透過除頻器將50MHz降為1Hz

- 除頻器實現方式是透過一個計數器，計算經過幾個時脈正(負)緣，當計數到  $50 \times 10^6$  即代表經過一秒

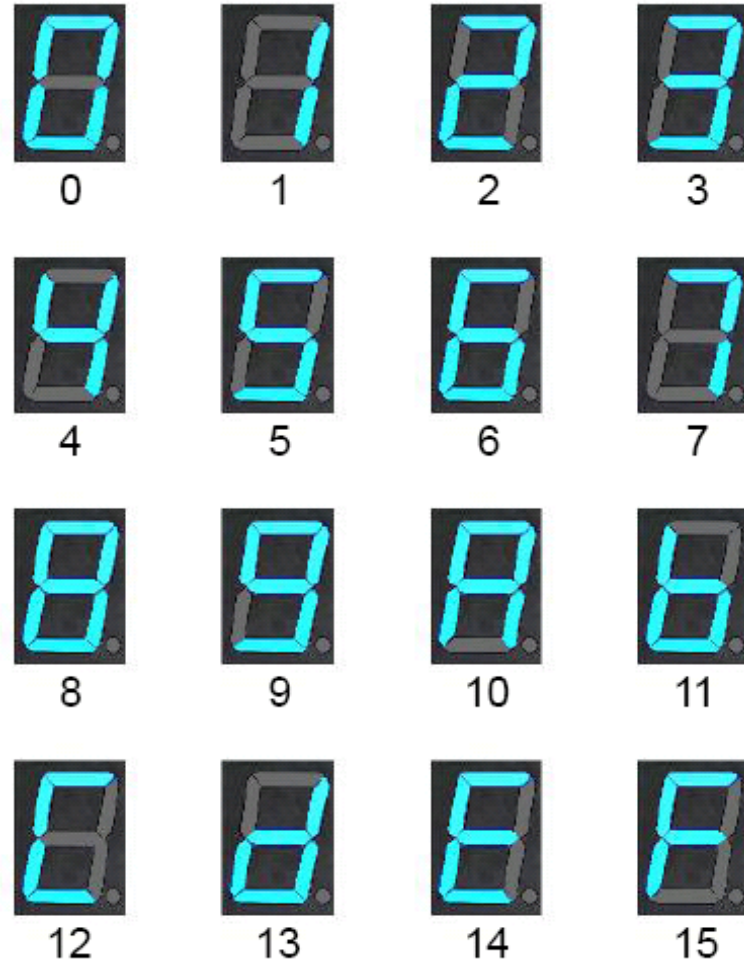
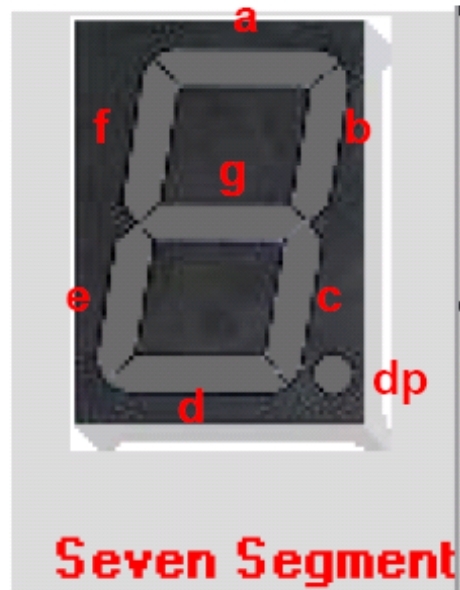
# Lab – Hint(2/2)

## ■ 除頻器範例：

- 每0.5秒改變一次訊號(div\_clk)

```
1  `define TimeExpire 32'd25000000
2
3  module clk_div(clk,rst,div_clk);
4  input clk,rst;
5  output div_clk;
6
7  reg div_clk;
8  reg [31:0]count;
9
10 always@(posedge clk)
11 begin
12     if(!rst)
13     begin
14         count <= 32'd0;
15         div_clk <= 1'b0;
16     end
17     else
18     begin
19         if(count == `TimeExpire)
20         begin
21             count <= 32'd0;
22             div_clk <= ~div_clk;
23         end
24         else
25         begin
26             count <= count + 32'd1;
27         end
28     end
29 end
30
31 endmodule
```


# Seven-segment display






# Seven-segment display (2/2)

- 0 is on, 1 is off
- dp is useless in DE0-CV board

■ Ex:  0 out=7'b1000000;  
g=1

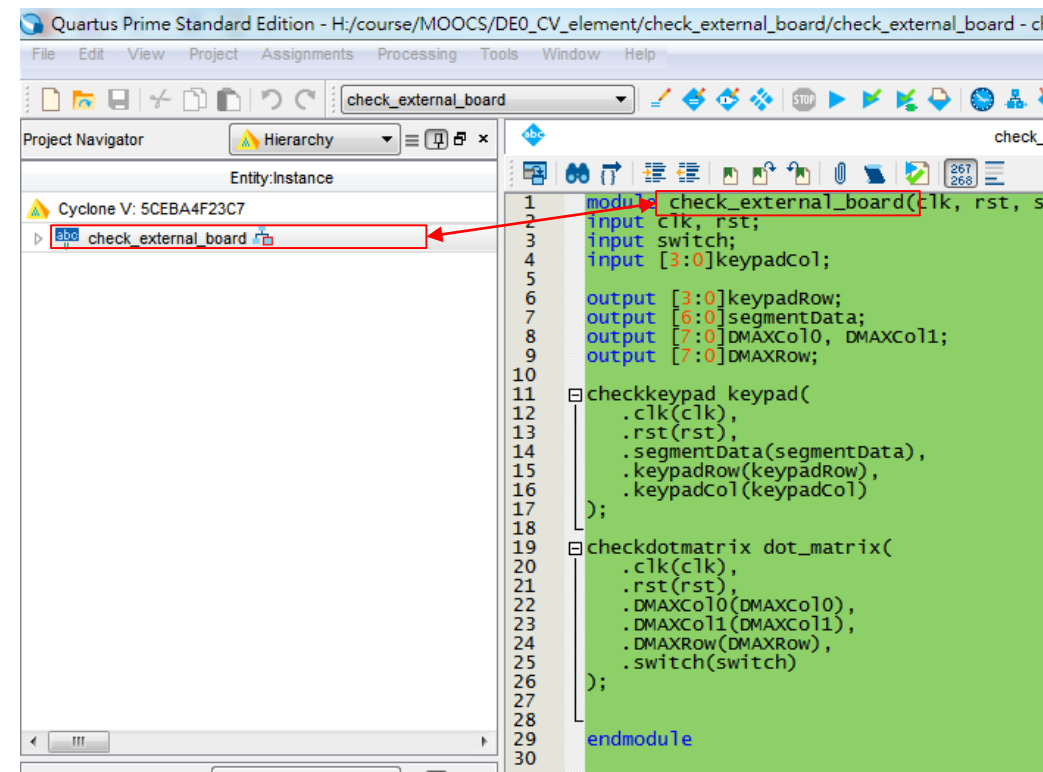
■ Ex:  5 out=7'b0010010;  
b=1, e=1

# Notice

- wire and reg type define
  - `always begin ...裡面變數... end`，宣告 **reg** type
  - `always begin ..... end` 外面變數，宣告 **wire** type
    - 需搭配 **assign** 使用
- `reg == register`
  - 在**組合**電路中使用 `reg type`，合成 → **線 (net)**
  - 在**循序**電路中使用 `reg type`，合成 → **Flip-flop (register)**
- Inferred latch
  - 在組合電路中，`case`、`if...else...`若**沒有寫滿**，合成後會產生latch

# Notice

- 請勿命名中文資料夾或數字開頭資料夾
- 請確認 Device family 是否與 FPGA 晶片符合
  - Family: **Cyclone** / Device: **5CEBA4F23C7**
- top module name & project name 需要一致
- 燒錄檔案至 FPGA 前，Double-check **Pin Assignment**
  - 設定錯誤的 Pin，會導致 FPGA 無法正確執行
- 連接 FPGA 板後，請先確認是否可以正常燒錄與動作
  - **USB Blaster**，指定到 USB Blaster Driver 目標資料夾 C:\altera\16.0\quartus\drivers\usb-blaster



# Number Representation

- May be represented using
  - Binary, decimal, hexadecimal,
- Format
  - `<size>'<base_format> <number>`
  - `base_format`:
  - `b, d, h,`
- Example
  - `4'b1111; 16'd255`
  - `23456` (32-bit decimal # by default); `'hc3` (32 bit)
  - `12'b1111_0000_1010`