# *LAB* - 10

陳培殷
國立成功大學　資訊工程系

# Lab: *Dot matrix controller*

# Dot matrix controller (1/2)

- Please design a **Dot matrix controller** by using the following components:
  - 2 LED Dot Matrix Displays
  - 2 buttons
  - 1 reset button
  - 2 LED

# Dot matrix controller (2/2)

- Show the image 1.

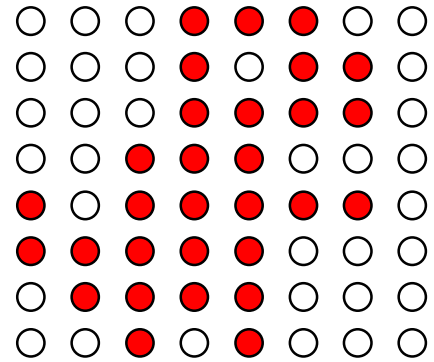
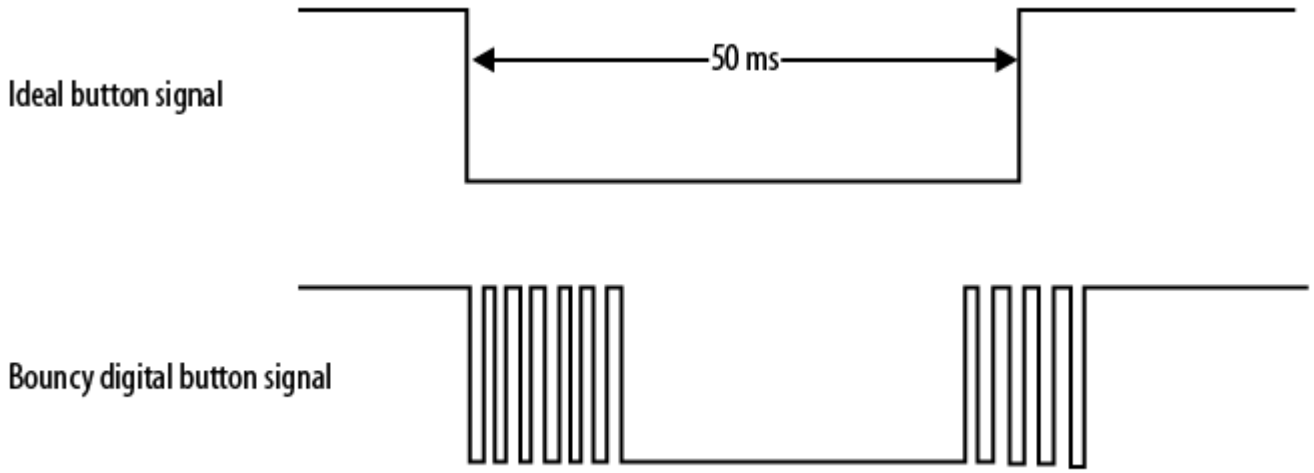


Image 1

- Clock Frequency : 4 Hz (button detection)
- Clock Frequency : 5000 Hz (dot matrix)
- Basic requirements :
  - The image moves right and set LED0 on when button0 is pressed.
  - The image moves left and set LED1 on when button1 is pressed .
  - When the reset button is pressed, go to the initial state (showing image 1 in the left of 2 LED Dot Matrix Displays).

# Button Bouncing

- **Use counter to deal button bouncing problem**




```
module btn_control(clk,rst,btn_signal,move);
   input clk,rst,btn_signal;
   output reg move;

   reg [31:0] cnt;

   always@(posedge clk or negedge rst)
   begin
      if (~rst)
      begin
         move <= 1'b0;
         cnt <= 32'd0;
      end
      else
      begin

         if (cnt == 12500000)
         begin
            move <= 1'b0;
            cnt <= 32'd0;
         end
         else if(cnt[7:0] == 8'd0)
         begin
            move <= (!btn_signal)? 1'b1 : move;
            cnt <= cnt + 32'b1;
         end
         else
         begin
            cnt <= cnt + 32'b1;
         end
      end
   end
```

# Notice

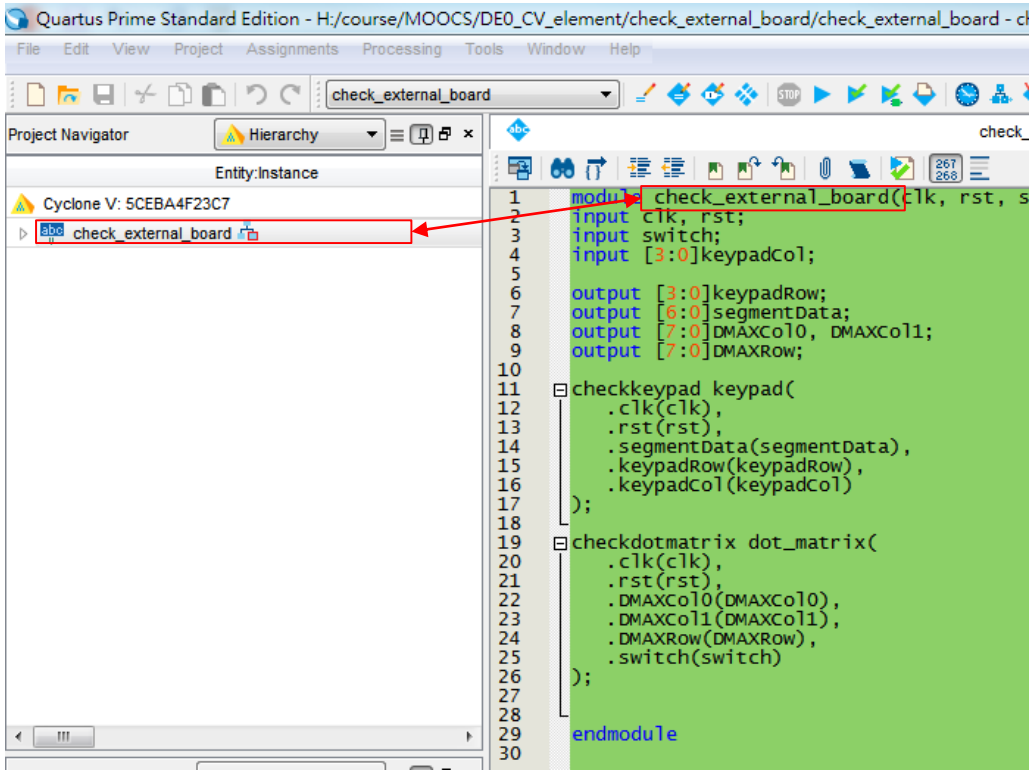
- wire and reg type define
  - always begin …裡面變數… end ，宣告 reg type
  - always begin …… end 外面變數，宣告 wire type
    - 需搭配 assign 使用

- 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**