

114 學年第一學期計算機系統設計期中考

- 請以圖 1 中的架構實現下面的運算，並設計一個 FSM 控制此架構。

$$\begin{bmatrix} M_0 & M_1 \\ M_2 & M_3 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \odot \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ag & bh \\ ce & df \end{bmatrix}$$

$$\begin{bmatrix} S_0 & S_1 \\ S_2 & S_3 \end{bmatrix} = \begin{bmatrix} M_0 & M_1 \\ M_2 & M_3 \end{bmatrix} + \begin{bmatrix} S_0 & S_1 \\ S_2 & S_3 \end{bmatrix}$$

$$W = S_0 - S_1 + S_2 - S_3$$

- 架構及演算法如下面說明，實現完畢請比對波形圖，評分標準在本卷最後面。

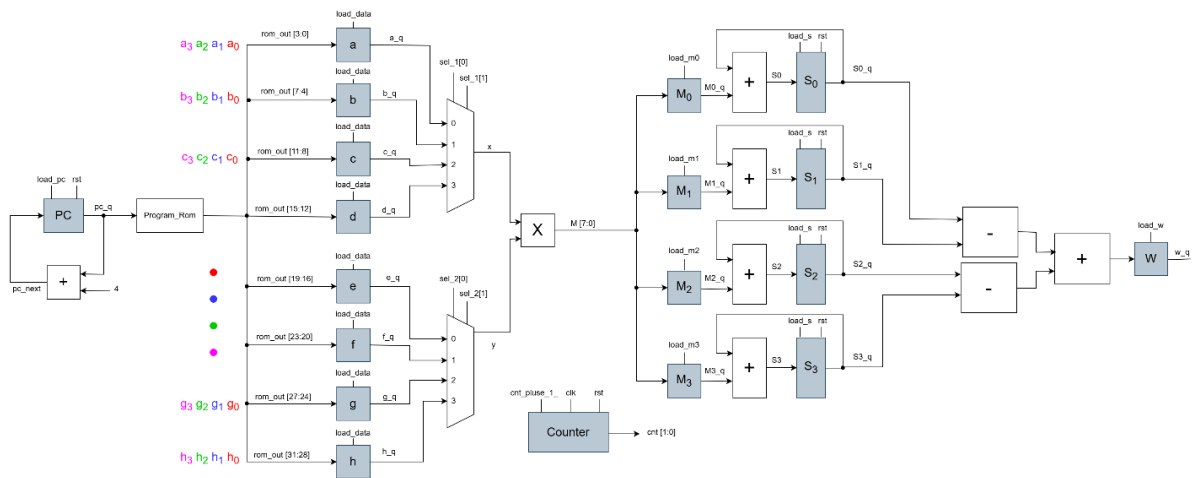
架構圖：



此顏色代表組合邏輯電路



此顏色代表 DFF，每個 DFF 都有 clk 跟 rst 腳



圖一

testbench:

```
module testbench;

    logic clk, rst;
    logic [7:0] w_q;

    TOP u_TOP(
        .rst(rst),
        .clk(clk),
        .w_q(w_q)
    );

    always #10 clk = ~clk;
    initial begin
        clk = 0; rst = 1;
        #15 rst = 0;
        #10000 $stop;
    end
endmodule
```

Program_ROM 的測試資料:

```
module Program_Rom(
    input  logic [3:0] Rom_addr_in,
    output logic [31:0] Rom_data_out
);

    always_comb begin
        case (Rom_addr_in)
            4'h0 : Rom_data_out = 32'h44216a93; // a0=3, b0=9, c0=a, d0=6, e0=1, f0=2, g0=4, h0=4
            4'h4 : Rom_data_out = 32'h65125748; // a1=8, b1=4, c1=7, d1=5, e1=2, f1=1, g1=5, h1=6
            4'h8 : Rom_data_out = 32'hffb00193; // a2=3, b2=9, c2=1, d2=0, e2=0, f2=b, g2=f, h2=f
            4'hc : Rom_data_out = 32'h156778dc; // a3=c, b3=d, c3=8, d3=7, e3=7, f3=6, g3=5, h3=1
            default: Rom_data_out = 32'h0;
        endcase
    end
endmodule
```

● 請實現圖 1 的硬體架構

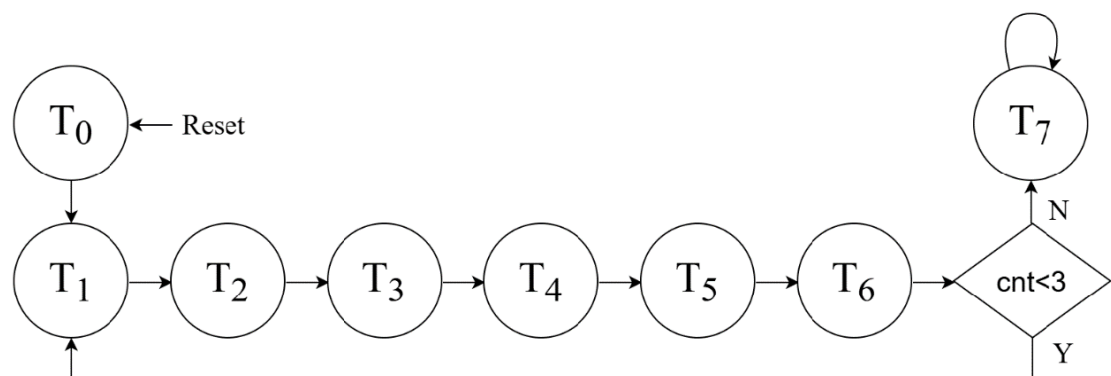
1. Program_Rom 的輸出為 32bit
2. a_q = rom_out [3:0]
3. b_q = rom_out [7:4]
4. c_q = rom_out [11:8]
5. d_q = rom_out [15:12]
6. e_q = rom_out [19:16]
7. f_q = rom_out [23:20]
8. g_q = rom_out [27:24]
9. h_q = rom_out [31:28]
10. 請使用助教提供的 Program_ROM 程式碼
11. $S_0_q = a_0g_0 + a_1g_1 + a_2g_2 + a_3g_3$
12. $S_1_q = b_0h_0 + b_1h_1 + b_2h_2 + b_3h_3$
13. $S_2_q = c_0e_0 + c_1e_1 + c_2e_2 + c_3e_3$
14. $S_3_q = d_0f_0 + d_1f_1 + d_2f_2 + d_3f_3$
15. $W_q = S_0 - S_1 + S_2 - S_3$

Datapath 描述:

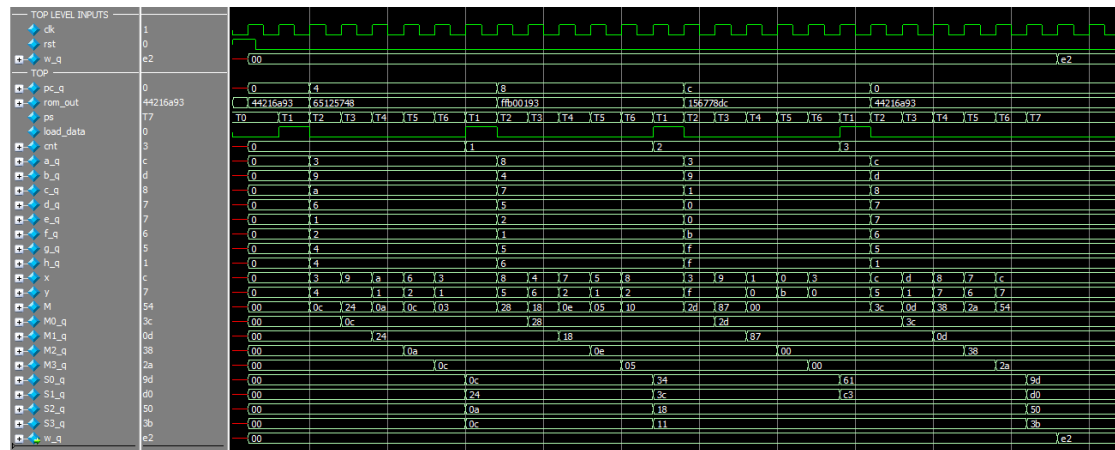
1. PC、a、b、c、d、e、f、g、h 為 4bit 暫存器
2. M_0 、 M_1 、 M_2 、 M_3 、 S_0 、 S_1 、 S_2 、 S_3 、W 為 8bit 暫存器
3. Counter 為一個兩位元的計數器，從零開始計數，每當完成一筆輸入資料後 (執行到載入 $S_{0\sim3}$ 暫存器)，計數器都會加 1，直到 Program_Rom 內的四筆測資都執行完畢。

Controller 描述(所有 FSM 的描述都會在晚 1 個 clk 後資料才真正發生改變)

- Reset 時，所有暫存器清除為 0，FSM 跳至 T_0 。
- T_0 ：不動作，下一個狀態跳到 T_1 。
- T_1 ：將 rom_out 讀入暫存器 a、b、c、d、e、f、g、h 中，同時將新的 pc_q 加 4 載入 PC 暫存器中，下一個狀態跳到 T_2 。
- T_2 ：將 $a \times g$ 的結果載入暫存器 M_0 ，下一個狀態跳到 T_3 。
- T_3 ：將 $b \times h$ 的結果載入暫存器 M_1 ，下一個狀態跳到 T_4 。
- T_4 ：將 $c \times e$ 的結果載入暫存器 M_2 ，下一個狀態跳到 T_5 。
- T_5 ：將 $d \times f$ 的結果載入暫存器 M_3 ，下一個狀態跳到 T_6 。
- T_6 ：將 $M_i_q + S_i_q$ 的結果載入暫存器 S_i ， $i = 0 \sim 3$ 。此時如果 Counter 內的數值(cnt) 小於 3，則發出 cnt_pluse_1_ 訊號使 Counter 加 1，下一個狀態跳到 T_1 ；否則下一個狀態跳到 T_7 。
- T_7 ：將 $S_0_q - S_1_q + S_2_q - S_3_q$ 的結果存入暫存器 w，下一個狀態跳到 T_7 。



輸出結果波形圖：



若結果無誤或是有任何提問請舉手

分數分配:

1. 越早交越高分
2. 正確描述完架構圖上的 Datapath，0 ~ 30 分
3. 可以順利將 Program_Rom 的值正確載入 a、b、c、d、e、f、g、h 暫存器中，30 ~ 50 分
4. 可以將乘法器的結果儲存到正確的 M₀ ~ M₃ 暫存器上，50 ~ 60 分
5. 可以至少運算出一筆正確的測資，60 ~ 80 分
6. 四筆測資全部算完並且跟波形圖一樣，80 ~ 100 分