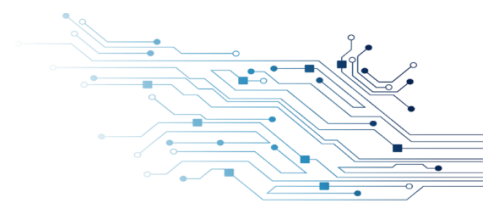




題目：Counter-like circuit(8-bit input & 16-bit output)





1. 實驗程式碼

```
module LAB4 (SW, KEY, HEX0);
    input [2:0] SW;
    input [0:0] KEY;
    output [6:0] HEX0;
    reg [3:0] count;

    seg_decoder display(count, HEX0);

    //KEY is manual clock input, SW[0] is active-low reset
    always @(negedge KEY or negedge SW[0]) begin
        if(~SW[0]) begin
            count = 0;
        end
        else begin
            if(SW[2] == 0 && SW[1] == 1) begin
                count = (count + 4'd1)%4'd10;
            end
            else if(SW[2] == 1 && SW[1] == 0) begin
                count = (count + 4'd2)%4'd10;
            end
            else if (SW[2] == 1 && SW[1] == 1) begin
                if(count == 0) begin
                    count = 4'd9;
                end
                else begin
                    count = count - 4'd1;
                end
            end
        end
    end
end
end
endmodule
```

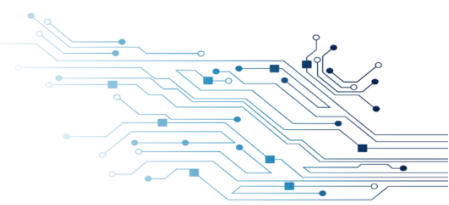
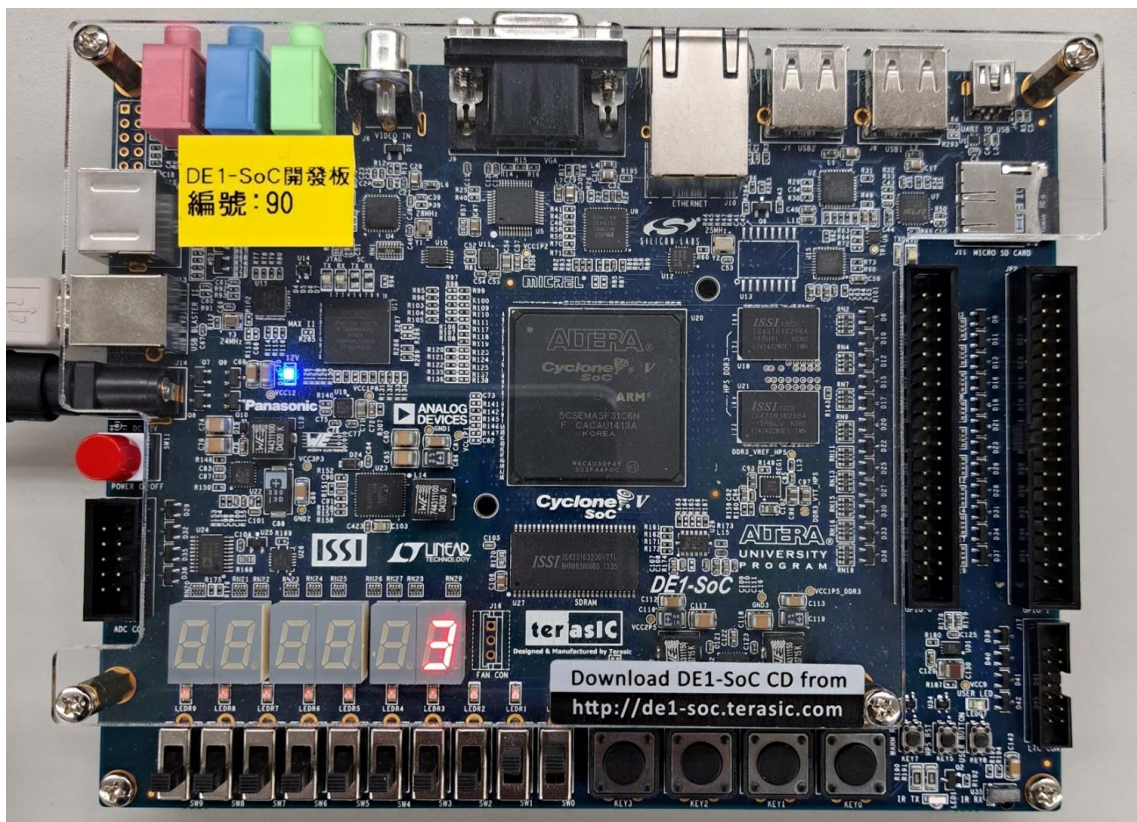


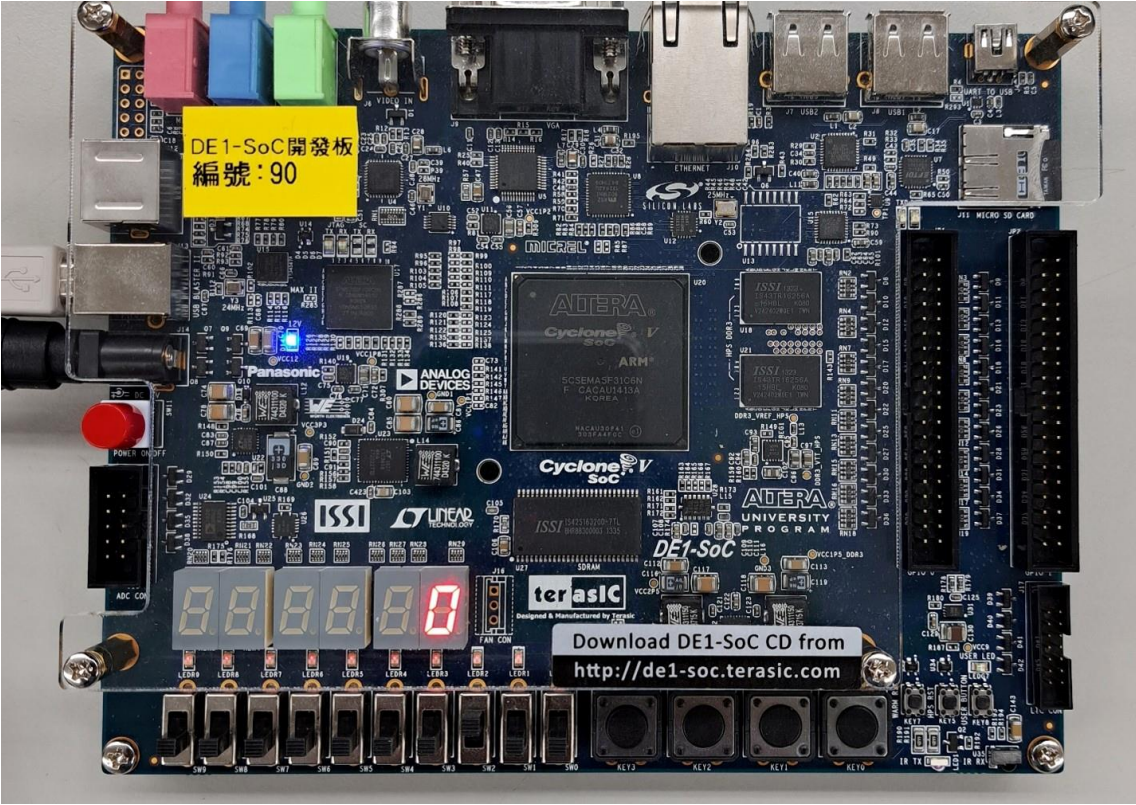
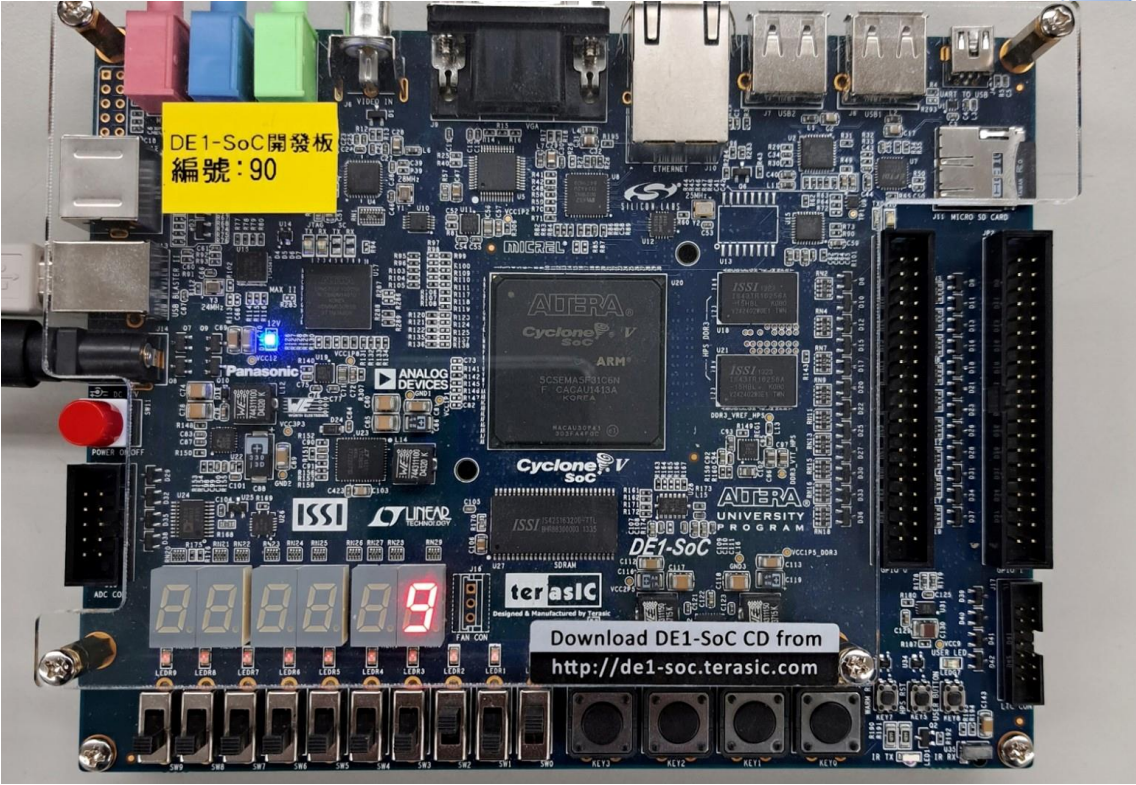


```
module seg_decoder (bcd, seg);
    input [3:0] bcd;
    output reg [6:0] seg;

    always @(bcd) begin
        case (bcd)
            0 : seg = 7'b1000000;
            1 : seg = 7'b1111001;
            2 : seg = 7'b0100100;
            3 : seg = 7'b0110000;
            4 : seg = 7'b0011001;
            5 : seg = 7'b0010010;
            6 : seg = 7'b0000010;
            7 : seg = 7'b1111000;
            8 : seg = 7'b0000000;
            9 : seg = 7'b0010000;
            //lights out
            default : seg = 7'b1111111;
        endcase
    end
endmodule
```

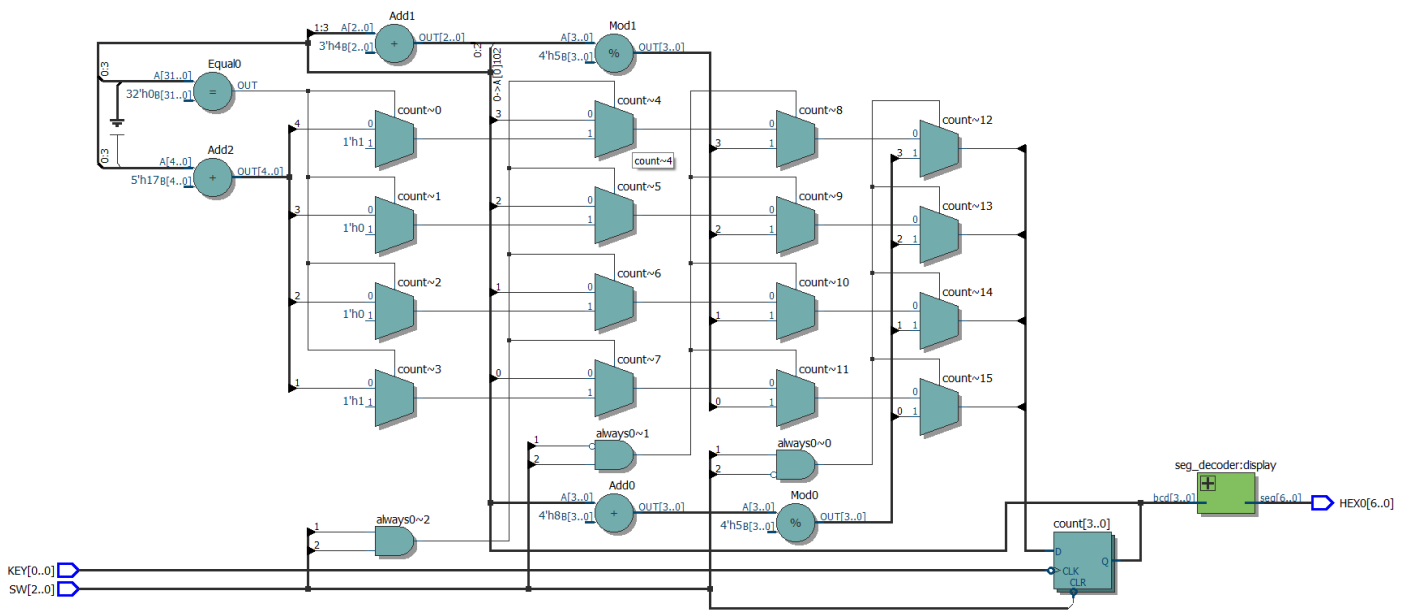
2. 實驗結果





3. RTL 佈局





4. 問題與討論

- ✚ 如果有些狀態沒有輸出到，要寫 default 做的事情。
- ✚ case 裡面通常不會直接放數字，用 parameter 為狀態命名。
- ✚ FPGA 由 ALMs 和高速公路組成，用既有的硬體做設定去符合 RTL 的邏輯功能，做乘除法會直接用 DSP 單元，加減法可直接用 LUT(look up table)就好。

