LAB 10

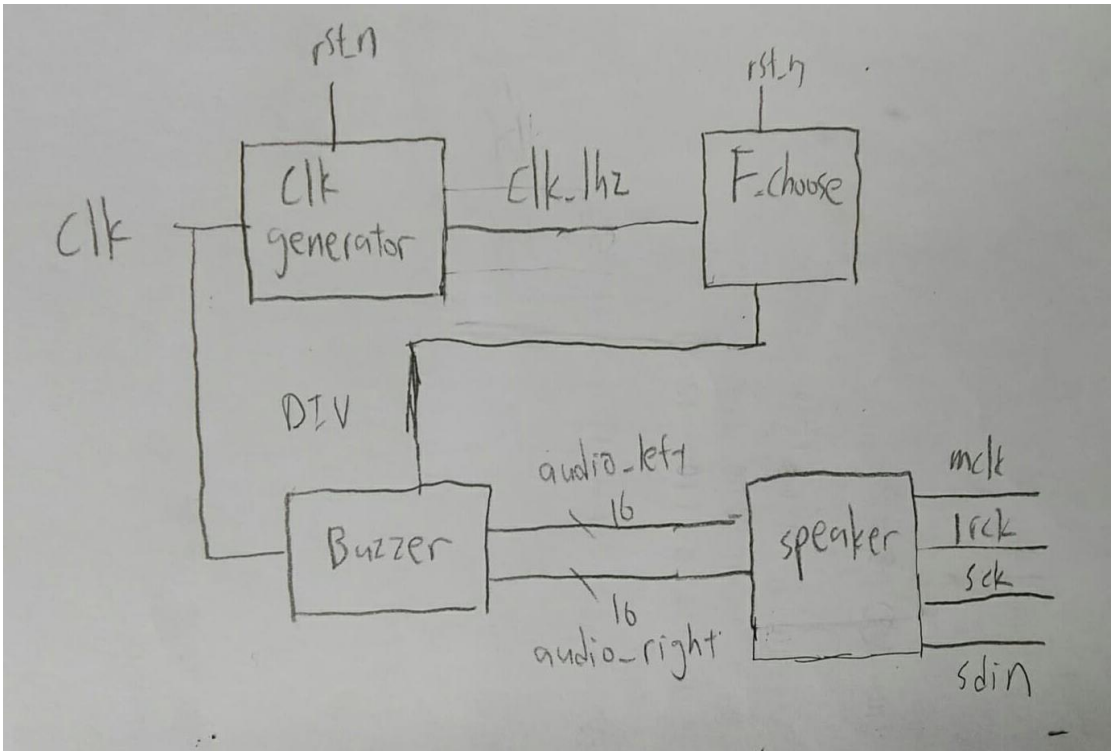
第一題

DESIGN SPCIFICATION

(1) INPUT / OUTPUT

```
module top(clk, rst_n, , mclk, lrck, sck, sdin);
        input clk;
        input rst_n;
        output mclk;
        output lrck;
        output sck;
       output sdin;

module F_choose(clk, rst_n, DIV);
        input clk;
        input rst_n;
      output reg [22:0]DIV;

module buzzer(clk, rst_n, note_div, audio_left, audio_right);
        input clk;
        input rst_n;
        input [21:0]note_div;
        output reg [15:0]audio_left;
        output reg [15:0]audio_right;

module speaker(clk, rst_n, audio_mclk, audio_lrck, audio_sck,
audio_sdin, audio_in_left, audio_in_right);
        input clk;
        input rst_n;
        input [15:0]audio_in_left;
        input [15:0]audio_in_right;
        output reg audio_mclk;
        output reg audio_lrck;
        output reg audio_sck;
        output reg audio_sdin;
```

(2) BLOCK DIAGRAM



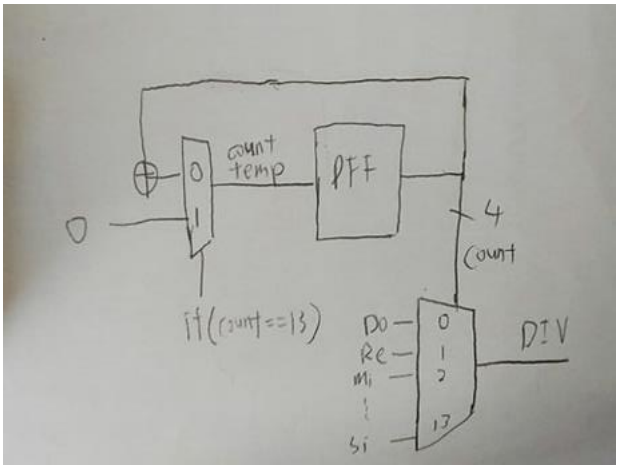
DESIGN IMPLEMENTATION

(1) I / O PINS

clk	IN		W5
lrck	OUT		A16
mclk	OUT		A14
rst_n	IN		V17
sck	OUT		B15
sdin	OUT		B16

(2) Verilogcode

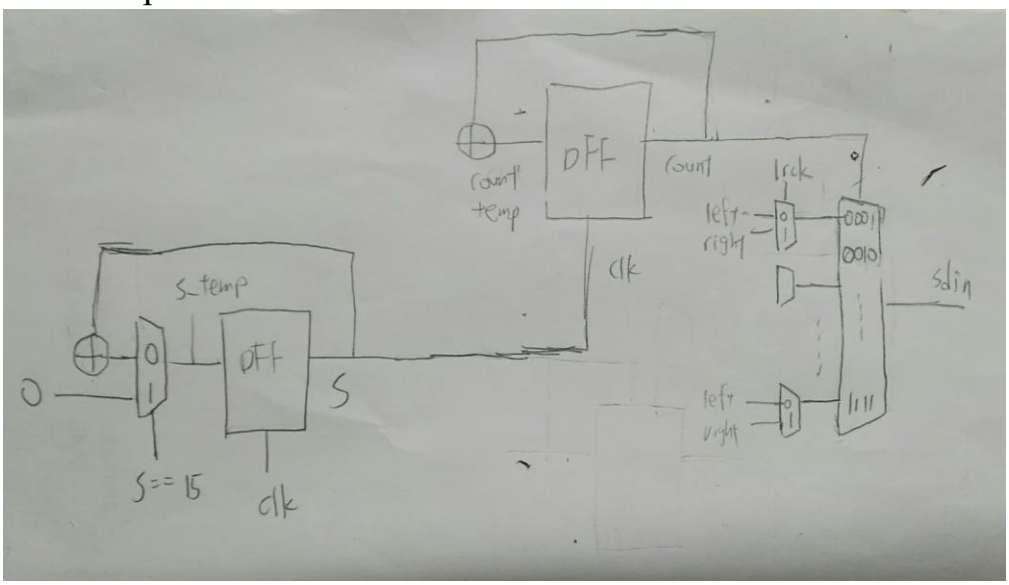
Module F_choose



Module buzzer

同老師 lab8 的範例

Module speaker



DISSCUSSION

這題看到就有想法了，很快就完成了，唯一的問題就是開啟

RESET 第一個 DO 持續時間短於一秒，於是我把

CLK__GENERATOR 加上 RESET 就解決了。

執行結果就是 14 個音階輪流撥放，各持續一秒。

CONCLUSION

把 14 個音階打上去其實也要很久

第二題

DESIGN SPECIFICATION

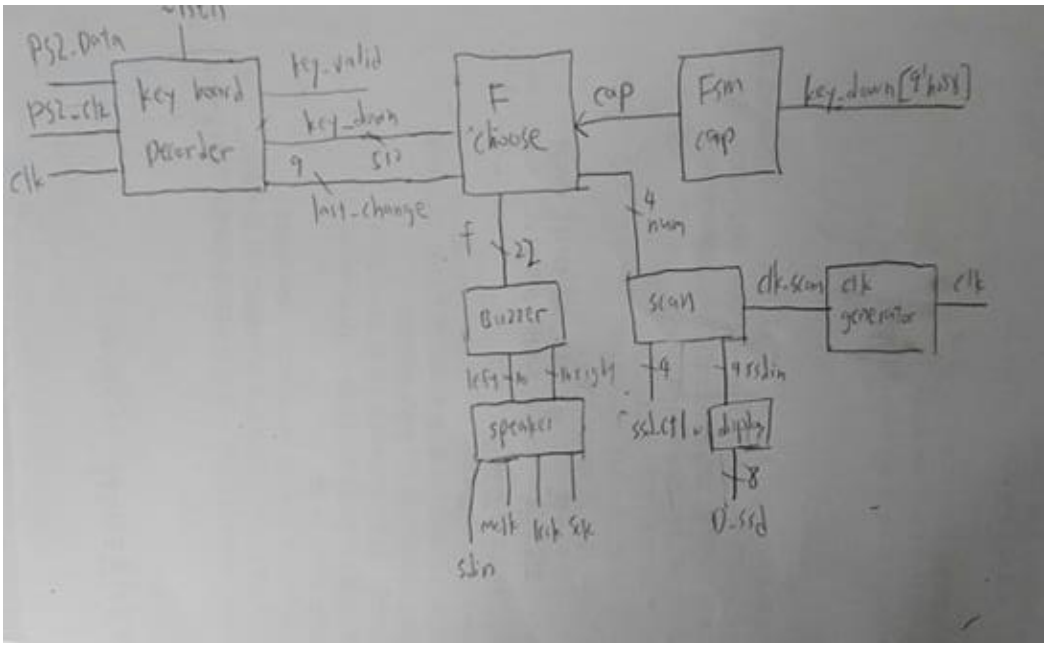
(1) INPUT / OUTPUT

module top(clk, rst_n, LED_CAP, PS2_DATA, PS2_CLK, ssd_ctl, D_ssd, mclk, lrck, sck, sdin);

```
        input clk;
        input rst_n;
        inout PS2_DATA;
        inout PS2_CLK;
        output LED_CAP;
        output [3:0]ssd_ctl;
        output [7:0]D_ssd;
        output mclk;
        output lrck;
        output sck;
        output sdin;

module KeyboardDecoder(
      output reg [511:0] key_down,
      output wire [8:0] last_change,
      output reg key_valid,
      inout wire PS2_DATA,
      inout wire PS2_CLK,
      input wire rst,
      input wire clk
       );

module FSM_CAP(in, rst_n, state);
       input in;
       input rst_n;
       output reg state;

module F_choose(last_change, key_down, CAP, clk, rst_n, f, num);
       input clk;
       input rst_n;
       input [8:0]last_change;
       input [511:0]key_down;
       input CAP;
       output reg [22:0]f;
       output reg [3:0]num;
       reg [22:0]f_temp;
       reg [3:0]num_temp;
```

Module buzzer, speaker 同第一題

(2) BLOCK DIAGRAM



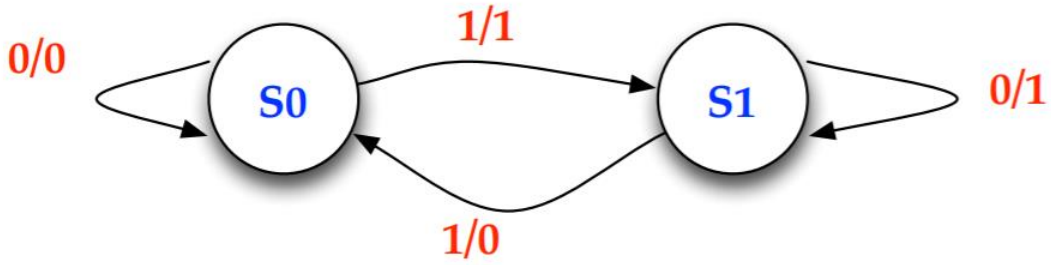
DESIGN IMPLEMENTATION

(1) I / O PINS

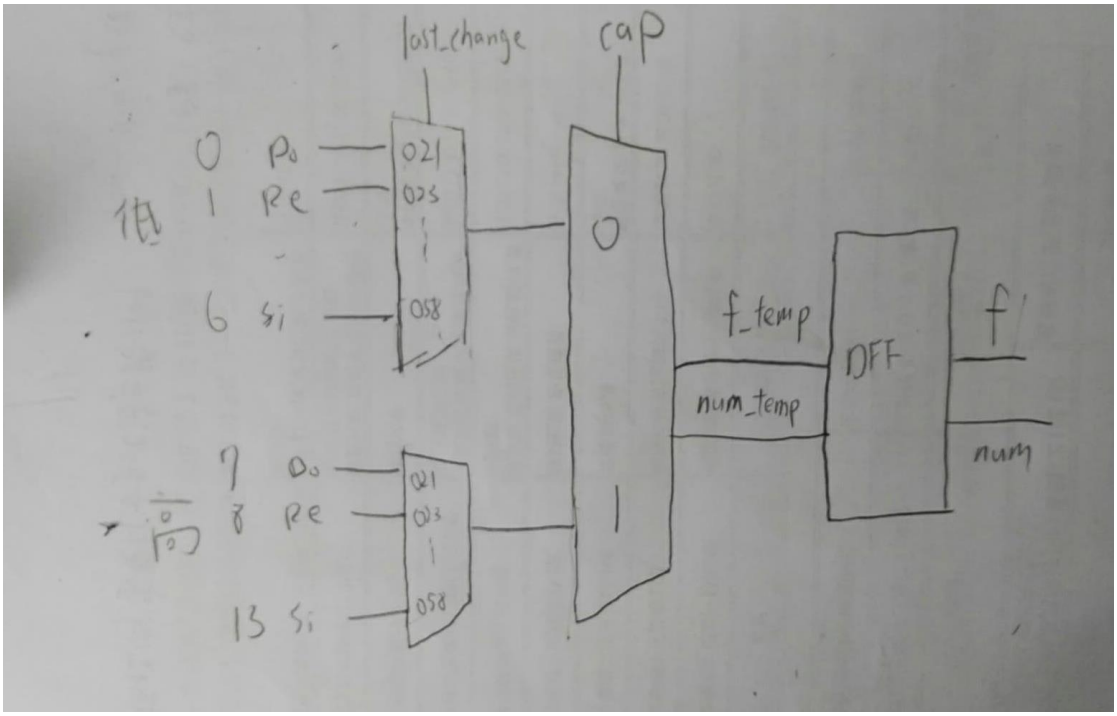
D_ssd[7]	OUT		W7
D_ssd[6]	OUT		W6
D_ssd[5]	OUT		U8
D_ssd[4]	OUT		V8
D_ssd[3]	OUT		U5
D_ssd[2]	OUT		V5
D_ssd[1]	OUT		U7
D_ssd[0]	OUT		V7
ssd_ctl (4)	OUT		
ssd_ctl[3]	OUT		W4
ssd_ctl[2]	OUT		V4
ssd_ctl[1]	OUT		U4
ssd_ctl[0]	OUT		U2
Scalar ports (9)			
clk	IN		W5
LED_CAP	OUT		L1
lrck	OUT		A16
mclk	OUT		A14
PS2_CLK	INOUT		C17
PS2_DATA	INOUT		B17
rst_n	IN		V17
sck	OUT		B15
sdin	OUT		B16

(2) VERILOG CODE

Module FSM_CAP



Module F_choose



DISCUSSION

這題是 KEYBOARD 和 SPEAKER 的整合，題目大致上不難，我也是很快就解決了。

按 CDEFGAB 分別發出 7 個音階的音，按 CAP 會變成高八度的 7 個音階，14 跟音階對應的數字是 1~14

CONCLUSION

KEYBOARD 跟 SPEAKER 有學好基本上沒問題!

第三題

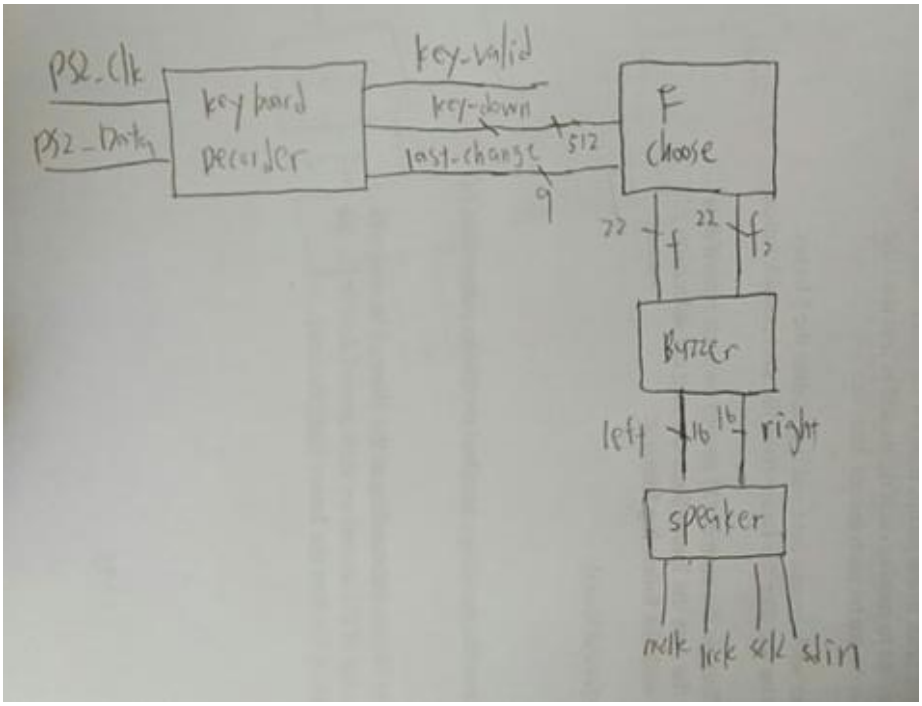
DESIGN SPECIFICATION

(1) INPUT/ OUTPUT

```
module top(clk, rst_n, PS2_DATA, PS2_CLK, mclk, lrck, sck, sdin,
double);
        input clk;
        input rst_n;
        inout PS2_DATA;
        inout PS2_CLK;
        input double;
        output mclk;
        output lrck;
        output sck;
       output sdin;

module F_choose(last_change, key_down, clk, rst_n, f, f2, double);
        input clk;
        input rst_n;
        input [8:0]last_change;
        input [511:0]key_down;
        input double;
        output reg [21:0]f;
        output reg [21:0]f2;
        reg [21:0]f_temp;
           reg [21:0]f2_temp;
module buzzer(clk, rst_n, f, f2, audio_left, audio_right);
 input clk;
 input rst_n;
 input [21:0]f;
 input [21:0]f2;
 output wire [15:0]audio_left;
 output wire [15:0]audio_right;
```

module keyboard_decorder 跟 speaker 同上

(2) BLOCK DIAGRAM



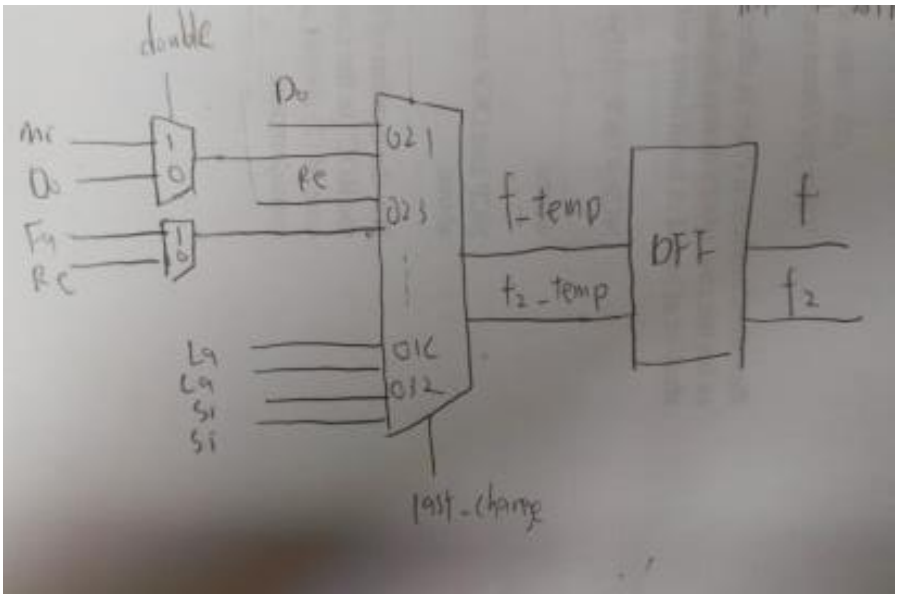
DESIGN IMPLEMENTAION

(1) I/O PINS

clk	IN	W5
double	IN	R2
lrck	OUT	A16
mclk	OUT	A14
PS2_CLK	INOUT	C17
PS2_DATA	INOUT	B17
rst_n	IN	V17
sck	OUT	B15
sdin	OUT	B16

(2) VERILOG CODE

Module F choose



Module BUZZER

同上，只有一個地方不同

```
assign audio_left = (b_clk == 1'b0) ? 16'hFE00 : 16'h01FF;
assign audio_right = (b_clk2 == 1'b0) ? 16'hFE00 : 16'h01FF;
```

F choose 的 f 是 b_clk，f2 是 b_clk2

DISSCUSSION

這題也不難，只是要思考要怎麼控制左右聲道，知道了就很快了。

按下 DIP SWITCH 切到雙聲道模式，關就同 10-2

CONCLUSION

這次 LAB 難度不高，讓我們練習整合的能力而已