

## Lab10 report

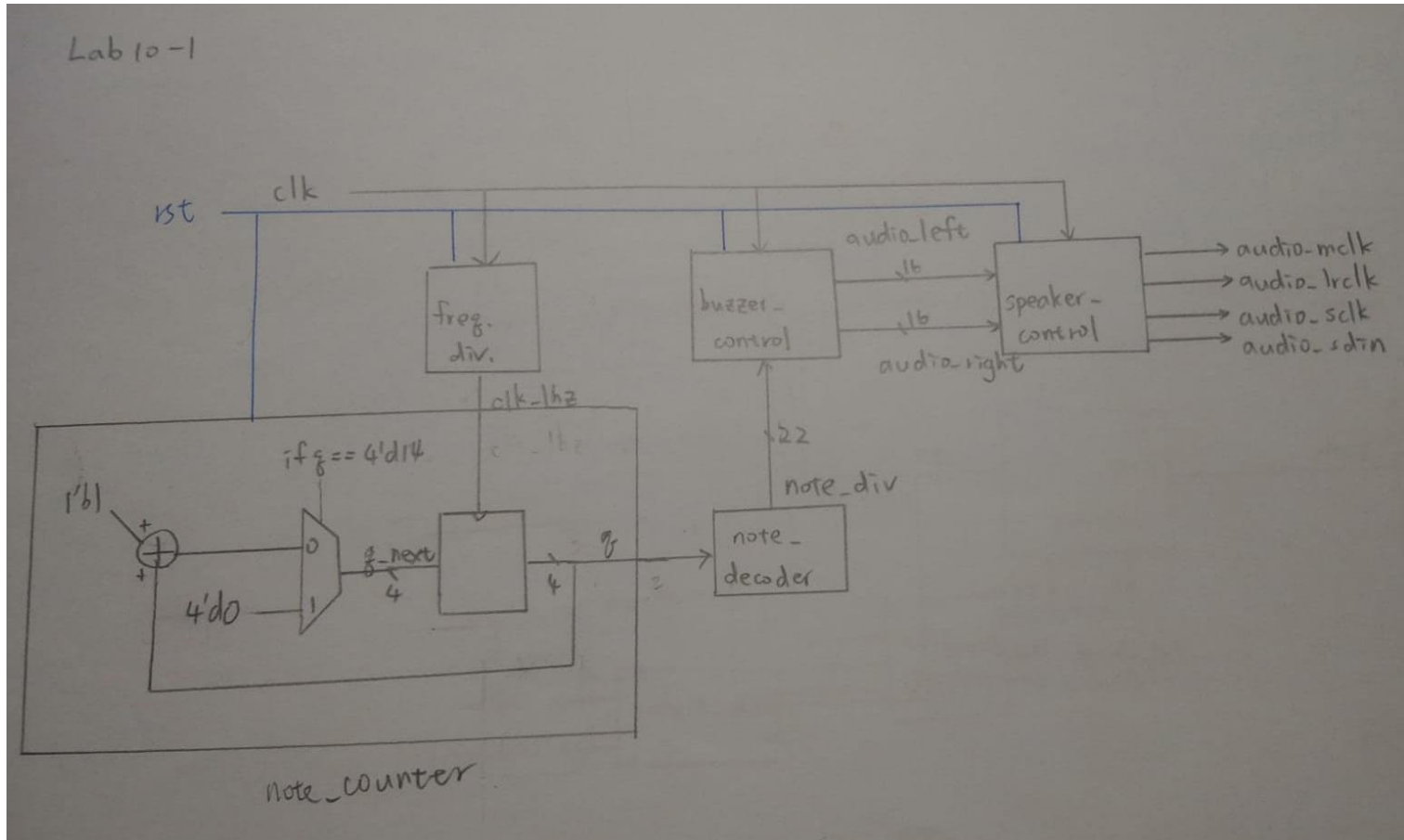
107061218 謝霖泳

1.

### ➤ Design Specification

Input: clk, rst

Output: audio\_mclk, audio\_lclk, audio\_sclk, audio\_sdin



### ➤ Design Implementation

題目要求以 1 秒為週期播放中音 Do 到高音 Ti 共 14 個音高。因此，先做出一個可以從 0 屬到 13，頻率為 1Hz 的 up counter，每當現在屬到不同的值時，就會在 note\_decoder module 中對應到不同的音高，意即不同的 note\_div，再將所得之 note\_div 傳到 speaker 的 buzzer control module 中，即可連續自動播放不同的音高。

IO pin assignment:

audio_mclk	audio_lclk	audio_sclk	audio_sdin	clk	rst
A14	A16	B15	B16	W5	V17

➤ Discussion

當我將 rst 的開關往上撥之後，便能聽到相對應之音高依序播出。

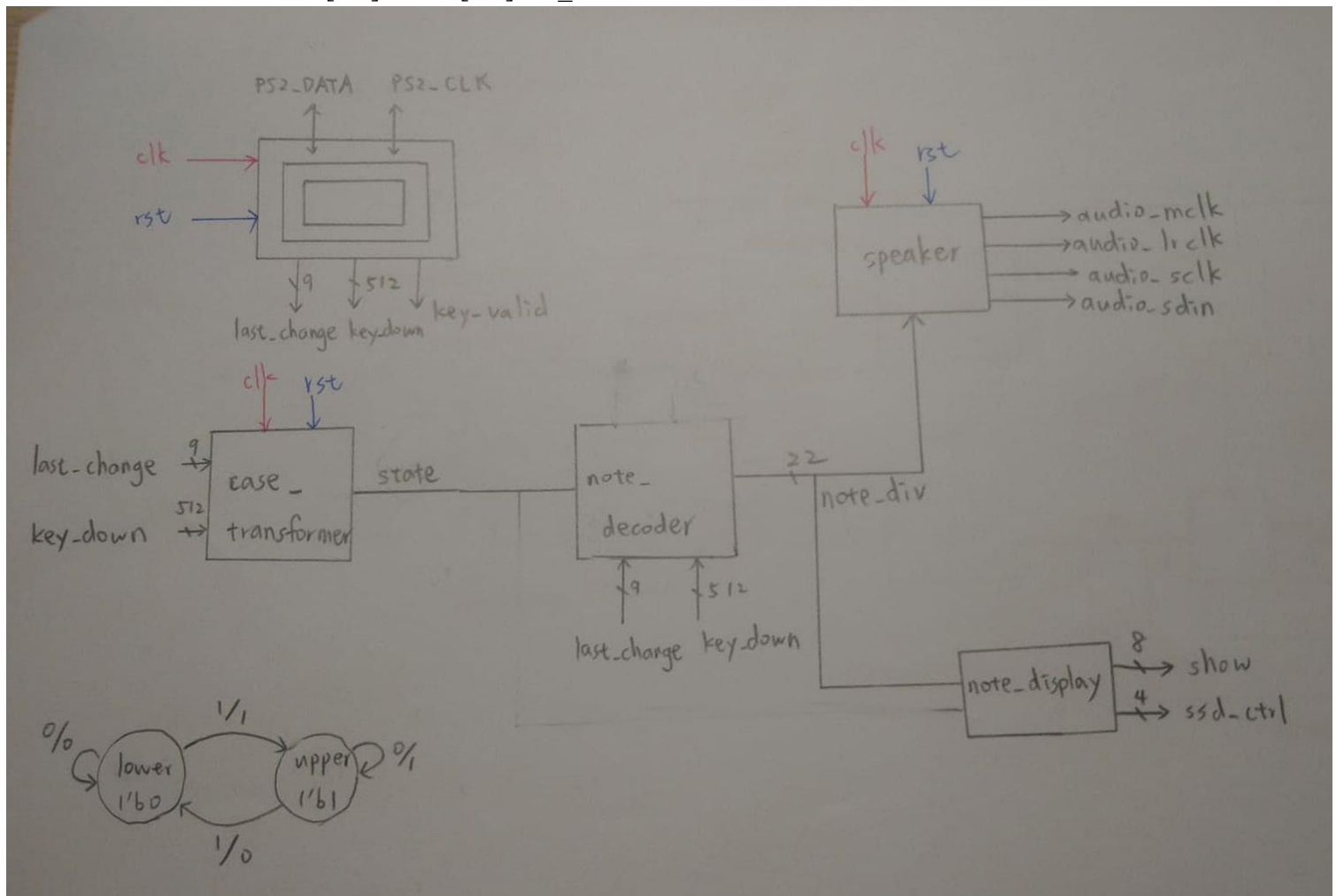
2.

➤ Design Specification

Inout: PS2\_CLK, PS2\_DATA

Input: clk, rst

Output: audio\_mclk, audio\_lrcclk, audio\_sclk, audio\_sdin, letter\_case, [7:0] show, [3:0] ssd\_ctrl



➤ Design Implementation

與上題類似，都會有一個 `note_decoder` 去對應我現在要發出的 `note_div`，只是，這一題因為有區分高音跟中音，因此，我多設了一個 `case_transformer` 去判斷。

在 `case_transformer` 中，我是用字母的大小寫來控制高音或中音。Reset 後，預設為中音的音域。當我按下鍵盤上的 caps lock 一下以後，再

來按的按鍵就都變成高音的音域。case 的意思就是大小寫，即 uppercase(大寫)與 lowercase(小寫)。因此 case\_transformer 主要就是一個 FSM，有兩個 state，分別表示大寫和小寫。

將大小寫的結果與現在按下的按鍵傳入 note\_decoder 中，便會跑出相對應的 note\_div，再將這個 note\_div 傳入 speaker 中，便可以發出按下按鍵相對應之音高，也就像系卡 K 時的表演那樣，可以當成一個簡易版的音樂 keyboard。此外，也將 note\_div 傳到 note\_display 中，便可以在七段顯示器上顯示相對應的數字。為了區分高音域和中音域，當現在為高音域時，左邊的一顆 LED 燈會亮，也就是我的其中一個 output letter\_case。

IO pin assignment :

audio_mclk	audio_lclk	audio_sclk	audio_sdin	clk	rst	letter_case
A14	A16	B15	B16	W5	V17	L1

show[7]	show[6]	show[5]	show[4]	show[3]	show[2]	show[1]	show[0]
W7	W6	U8	V8	U5	V5	U7	V7

PS2_CLK	PS2_DATA	ssd_ctrl[3]	ssd_ctrl[2]	ssd_ctrl[1]	ssd_ctrl[0]
C17	B17	W4	V4	U4	U2

### ➤ Discussion

這一題的概念與上提相似，因此沒有花太多時間便完成了。

3.

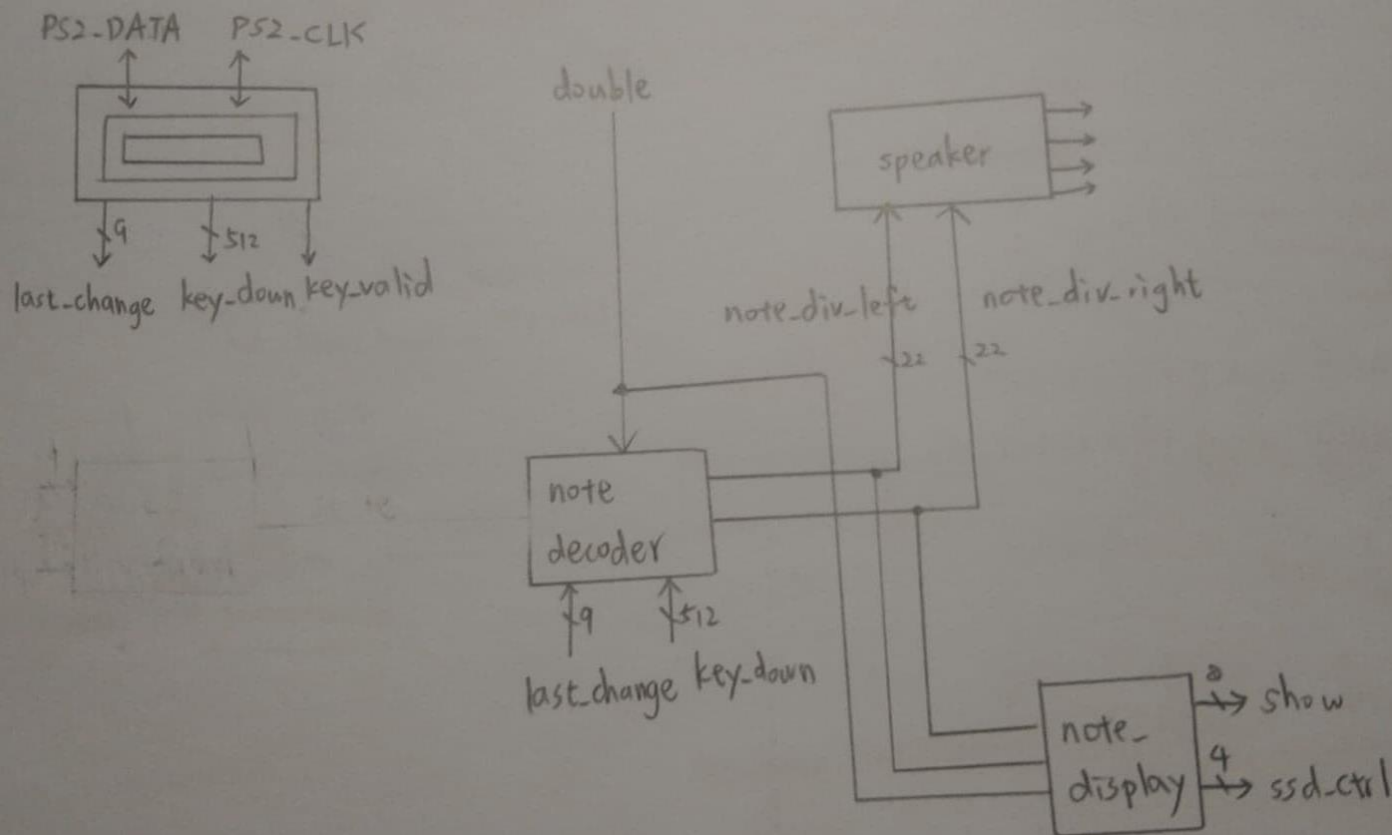
### ➤ Design Specification

Inout: PS2\_CLK, PS2\_DATA

Input: clk, rst, double

Output: audio\_mclk, audio\_lclk, audio\_sclk, audio\_sdin, [7:0] show,  
[3:0] ssd\_ctrl

Lab10-3



### ➤ Design Implementation

為了製作出類似和弦的聲音，必須讓左右兩聲道擁有不同的 `note_div` 值，因此，在 `speaker` module 中，需要呼叫兩次的 `buzzer control`，輸入 `note_div_left [21:0]` 的負責產生 `audio_left [15:0]`，反之，輸入 `note_div_right [21:0]` 的負責產生 `audio_right [15:0]`，再將這個 32-bit 訊號傳入 `speaker control` 中，便可以讓兩聲道發出不同頻率的聲音。

其中，我的 `input double` 為一個 DIP switch，用來控制我現在要產生單音還是和弦。若 `double == 1'b1` 時產生和弦，`double == 1'b0` 時產生單音，也就是上一小題的結果。

IO pin assignment:

audio_mclk	audio_lrcclk	audio_sclk	audio_sdin	clk	rst	double
A14	A16	B15	B16	W5	V17	R2

show[7]	show[6]	show[5]	show[4]	show[3]	show[2]	show[1]	show[0]
W7	W6	U8	V8	U5	V5	U7	V7

PS2_CLK	PS2_DATA	ssd_ctrl[3]	ssd_ctrl[2]	ssd_ctrl[1]	ssd_ctrl[0]
C17	B17	W4	V4	U4	U2

#### ➤ Discussion

要讓左右兩聲道產生不同頻率聲音的關鍵就在於：給定左右兩聲道不同的 note\_div，即可達成和弦的效果。

#### 4. conclusion

這次的 lab 算是比較輕鬆的，因為就是把前兩次 lab 的東西，也就是 speaker 和 keyboard 合在一起。只要把 block diagram 畫清楚了，要執行成功應非難事。