

## CS 2104 02 Hardware Design and Labs 2018

### Lab 8

學號：105072123 姓名：黃海茵

#### 1. 實作過程

首先說明除了 top module 以外，助教給的 sample code 中我改了哪些地方。

##### (1) Music.v

```
always @(*) begin
    if(!StartPause) begin
        tone_b = `NM0;
        tone_c = `NM0;
    end
    else if(en==0 && Music==0)begin //告白氣球
        case(ibeatNum)
            8'd0 :begin
                tone_b = `NM0;
                tone_c = `NM0;
            end
            8'd1 :begin
                tone_b = `NM0;
                tone_c = `NM0;
            end
            8'd2 :begin
                tone_b = `NMfs;
                tone_c = `NMf;
            end
        end
    end
end

else if(en==0 && Music==1) begin // jingle bell
    case(ibeatNum)
        8'd0 :begin
            tone_b = `NM0;
            tone_c = `NM0;
        end
        8'd1 :begin
            tone_b = `NM0;
            tone_c = `NM0;
        end
        8'd2 :begin
            tone_b = `NMfs;
            tone_c = `NMf;
        end
    end
end
```

我多加了一個判斷 StartPause 的部分，因為當音樂暫停時，不是播放當下的音頻，而是要無聲。以及把原本的 tone 改成了 tone\_b 和 tone\_c，分別是左聲道和右聲道的音頻，並加了 Music 等於 1 還是 0 的判斷，來選擇播放哪一首歌，其他都和原本的 sample code 相同。

##### (2) note\_gen.v

```
always @*
    if (clk_cnt == note_div_b)
        begin
            clk_cnt_next = 22'd0;
            b_clk_next = ~b_clk;
        end
    else
        begin
            clk_cnt_next = clk_cnt + 1'b1;
            b_clk_next = b_clk;
        end
end

always @*
    if (clk_cnt_2 == note_div_c)
        begin
            clk_cnt_next_2 = 22'd0;
            c_clk_next = ~c_clk;
        end
    else
        begin
            clk_cnt_next_2 = clk_cnt_2 + 1'b1;
            c_clk_next = c_clk;
        end
end

// Assign the amplitude of the note
//assign audio_left = (b_clk == 1'b0) ? 16'hE000 : 16'h2000;
//assign audio_right = (c_clk == 1'b0) ? 16'hE000 : 16'h2000;
assign audio_left = (note_div_b == 22'd1000 || Volume == 4'b0) ? 16'h0000 : (b_clk == 1'b0) ? 16'h8001 :
    (Volume == 4'b0001) ? 16'h8FFF : (Volume == 4'b0011) ? 16'hD556 : (Volume == 4'b0111) ? 16'h2AAB : 16'h7FFF;
assign audio_right = (note_div_c == 22'd1000 || Volume == 4'b0) ? 16'h0000 : (c_clk == 1'b0) ? 16'h8001 :
    (Volume == 4'b0001) ? 16'h8FFF : (Volume == 4'b0011) ? 16'hD556 : (Volume == 4'b0111) ? 16'h2AAB : 16'h7FFF;
```

我把原本的 note\_div 改成了 note\_div\_b 和 note\_div\_c，分別是傳入的 freq\_out\_b(左)和 freq\_out\_c(右)。還有加了 LED 燈(Volume)的判斷條件，來

改變音量大小，由小到大分別是 8FFF, D556, 2AAB, 7FFF，然後當 Mute 打開時(Volume 為 4'b0)為靜音。其他都和原本的 sample code 相同。

### (3) SevenSegment.v

```
4'b0111 : begin
    display_num <= tone_c;
    digit <= 4'b1110;
end
default : begin
    display_num <= 32'd0;
    digit <= 4'b1110;
end
endcase
end
end

always @ (*) begin
    case (display_num)
        32'd262 :display = 7'b1000110; // C
        32'd294 :display = 7'b0100001; // D
        32'd330 :display = 7'b0000110; // E
        32'd349 :display = 7'b0001110; // F
        32'd370 :display = 7'b0001110; // #F
        32'd392 :display = 7'b0010000; // G
        32'd440 :display = 7'b0001000; // A
        32'd494 :display = 7'b0000011; // B
        32'd524 :display = 7'b1000110; // C sharp
        32'd588 :display = 7'b0100001; // D sharp
        32'd660 :display = 7'b0000110; // E sharp
        32'd698 :display = 7'b0001110; // F sharp
        32'd740 :display = 7'b0001110; // #F sharp
        32'd784 :display = 7'b0010000; // G sharp
        32'd880 :display = 7'b0001000; // A sharp
        32'd988 :display = 7'b0000011; // B sharp
        default : display = 7'b0111111;
    endcase
end
```

讓 DIGIT[0]顯示右聲道的音名(顯示左聲道也一樣，反正是差八度)。

其餘有用到的 module: clock\_divider, debounce, onepulse, speaker\_contro，都和原本的 sample code 相同。

不過我把 PlayerCtrl.v 拉到 top module 裡跟 state 一起做了，以下說明。

#### (1) RESET

```
always @(*) begin
    case(state)
        RESET: begin
            next_ibeat = 0;
            next_state = (!StartPause)? RESET : (!Music)? BALLON : JINGEL;
        end
    endcase
end
```

用來把 ibeat 歸零的 state，然後判斷 StartPause 是否等於 1，來決定要不要開始撥放音樂。若要播放，再判斷 Music 等於 1 還是 0，決定要播告白氣球還是 JINGLE BELLS。

## (2) BALLON

```
BALLON: begin
  if (!Music) begin
    if(ibeat < BEATLEAGTH) begin
      next_ibeat = (!StartPause)? ibeat : ibeat + 1;
    end
    else if (ibeat == BEATLEAGTH)begin
      next_ibeat = (Repeat && StartPause)? 0 : ibeat;
    end
  end
  else if(Music) begin
    next_ibeat = 0;
  end
  next_state = (Music)? JINGEL : BALLON;
end
```

用來播放告白氣球的 state，若 StartPause 為 0，歌曲就會暫停。在 ibeat 等於 beat 長度時，只要 Repeat 為 1 且 StartPause 為 1，歌曲就會從頭再播放一遍(ibeat 歸零)，若沒有的話則停止，持續播放最後一個音(無聲)。因為切換歌曲要從頭開始撥放，所以當 Music 切為 1 時，ibeat 就要馬上歸零，並且進入播放 JINGEL BELLS 的 state。

## (3) JINGLE

```
JINGEL: begin
  if (Music) begin
    if(ibeat < BEATLEAGTH) begin
      next_ibeat = (!StartPause)? ibeat : ibeat + 1;
    end
    else if (ibeat == BEATLEAGTH)begin
      next_ibeat = (Repeat && StartPause)? 0 : ibeat;
    end
  end
  else if(!Music) begin
    next_ibeat = 0;
  end
  next_state = (!Music)? BALLON : JINGEL;
end
```

用來播放 JINGLE BELLS 的 state，和 BALLON 相同。若 StartPause 為 0，歌曲就會暫停。在 ibeat 等於 beat 長度時，只要 Repeat 為 1 且 StartPause 為 1，歌曲就會從頭再播放一遍(ibeat 歸零)，若沒有的話則停止，持續播放最後一個音(無聲)。唯一不同的是，在這個 state 是 Music 切為 0 時，ibeat 馬上歸零，並且進入播放告白氣球的 state。

以下說明 LED 燈(Volume)的部分。

```

always@(posedge clk16 or posedge Reset_one)begin
    if(Reset_one)begin
        Volume <= 4'b0001;
        temp <= 4'b0001;
    end
    else begin
        Volume <= (Mute)? 4'b0 : next_Volume;
        temp <= (Mute)? temp : next_Volume;
    end
end

always@(*)begin
    if(VolumeUp_one)begin
        next_Volume = (Mute)? Volume : (Volume == 4'b1111)? 4'b1111 : (Volume<<1)+1;
    end
    else if(VolumeDown_one)begin
        next_Volume = (Mute)? Volume : (Volume == 4'b0001)? 4'b0001 : Volume>>1;
    end
    else begin
        next_Volume = temp;
    end
end

```

只要 Mute 為 1，LED 燈即為全部熄滅的狀態。當 Mute 為 0 時，按下 VolumeUp 的按鈕，LED 燈往右邊多亮一個；按下 VolumeDown 的按鈕，LED 燈往左邊多熄一個。而 temp 是用來存 Mute 沒有打開時的 LED 燈狀態。(這部分遇到了一點小問題，下面遇到的困難會說明～)

## 2. 學到的東西與遇到的困難

當我使用原始的 clk trigger LED 燈時(debounce 跟 onepulse 也是用 clk)，就會有按一下按鈕，結果一次亮兩顆或熄兩顆，甚至是三顆的問題。但把 clk 改為  $\text{clk}/(2^{16})$  就沒有這個問題，其他的 code 完全沒有動過。求解～～Demo 時問了助教，助教也說不知道，然後我問了系上的同學，有人也有一樣的問題 QQ

## 3. 想對老師或助教說的話

終於打完這學期全部的 Lab 了，感動～～～・°・(PД`q。)・°・  
希望期末可以平安的度過這門課……