

LAB3 數位錄音機 教學手冊

第九組：劉德元 彭俊又 陶昇永

一. 簡介

本次實驗是要完成一個具有以下功能的數位錄音機：

1. 具備錄音、播放、暫停、停止播放等功能。
2. 取樣頻率至少為 32kHz，每個取樣至少為 16 bits。
3. 可以錄製的時間至少為 32 秒，並且可以正常播放。
4. 具備快速播放(2, 3, 4, 5, 6, 7, 8 倍速)以及慢速播放(1/2, 1/3, 1/4, 1/5, 1/6, 1/7, 1/8 倍速，且須含內插 信號功能(0 次內插與 1 次內插皆要支援)。
5. 須以容易了解的方式顯示目前秒數與系統狀態。

二. 實作

數位錄音機的系統可以分為三個部分：

- a. Initialize
- b. Top
- c. Display

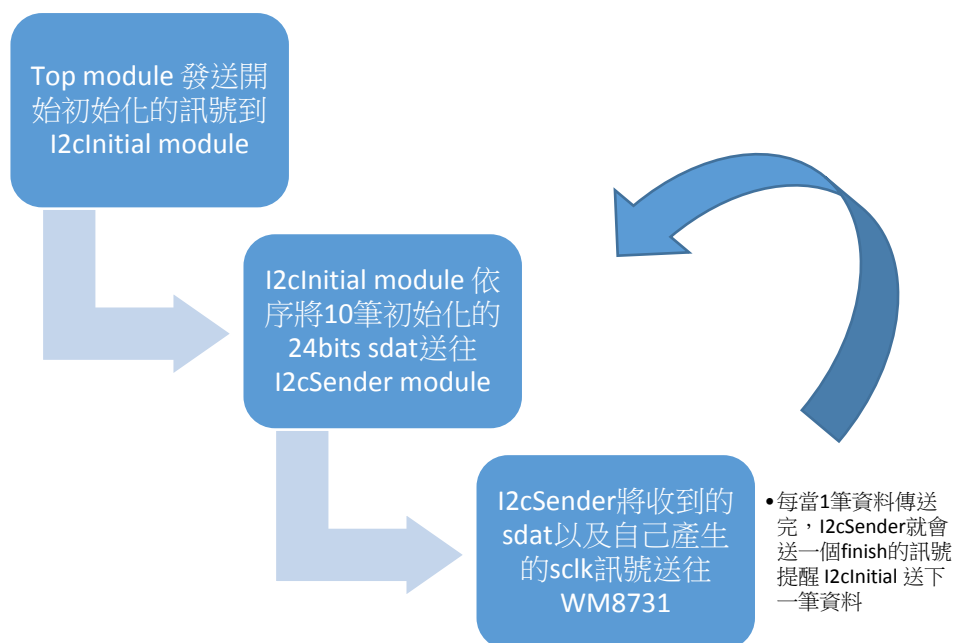
其中 Initialize 包含了初始化 WM8731 晶片所需的 I2cInitial 和 I2cSender 模組。Top 中則是包含了 RECORD 和 PLAY 這兩個模組，並且接收來自 FPGA 上按鈕或開關的訊號並做出相對應的處理。Display 中主要是將狀態圖和秒數顯示在七段顯示器上。接下來就是針對每個部份的說明。

1. Initialize

由於這次的實驗會用到 FPGA 上的 WM8731 晶片來傳送聲音的資料，所以為了順利使用該晶片，必須先將其初始化才能進行之後錄音跟播放的動作。

學生可以藉由設計出一個 I2cInitial 和一個 I2cSender 的模組來進行初始化的動作，兩者都是接收 100kHz 的工作頻率。I2cInitial 負責接收來自總控制模組 Top 的訊號來決定是否開始初始化，當接收到開始的訊號後，I2cInitial 就將第一筆 24bits 的訊號一次送到 I2cSender 模組，再由 I2cSender 將資料從 most significant bit 開始 1bit、1bit 將 I2C_SDAT 的資料送往 WM8731。I2cSender 還有一個任務就是要在傳送 I2C_SDAT 的同時，產生 I2C_SCLK 的訊號並傳給 WM8731 讓其知道何時要接收資料。

初始化的流程參考如下：



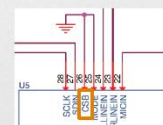
十筆資料由右方二張圖構成，順序分別是：
ADDRESS + CSB STATE + SETTINGS，所以第一筆資料就是 0011_0100_0000_0000_1001_0111。
 對於初始化資料中各個 bit 的含義多加了解對於之後的調整很重要。

例如，若要快速檢查初始化是否成功，則可以將 Analogue Audio Path Control 的最小八個 bit 改成 0010_0101，使得 micro in 的訊號直接連至 line out，傳入麥克風的聲音即可透過喇叭即時播出。

2-Wire Interface (1/2)

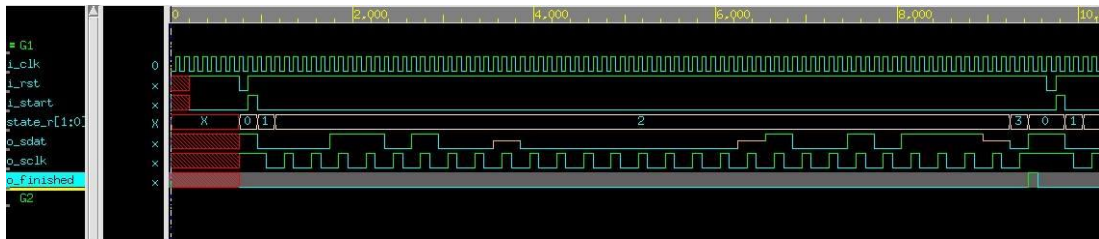
- The device operates as a **slave device** only.
- The WM8731/L has one of two slave address that are selected by setting the state of the **CSB** pin.

CSB STATE	ADDRESS
0	0011010
1	0011011



Recommended settings

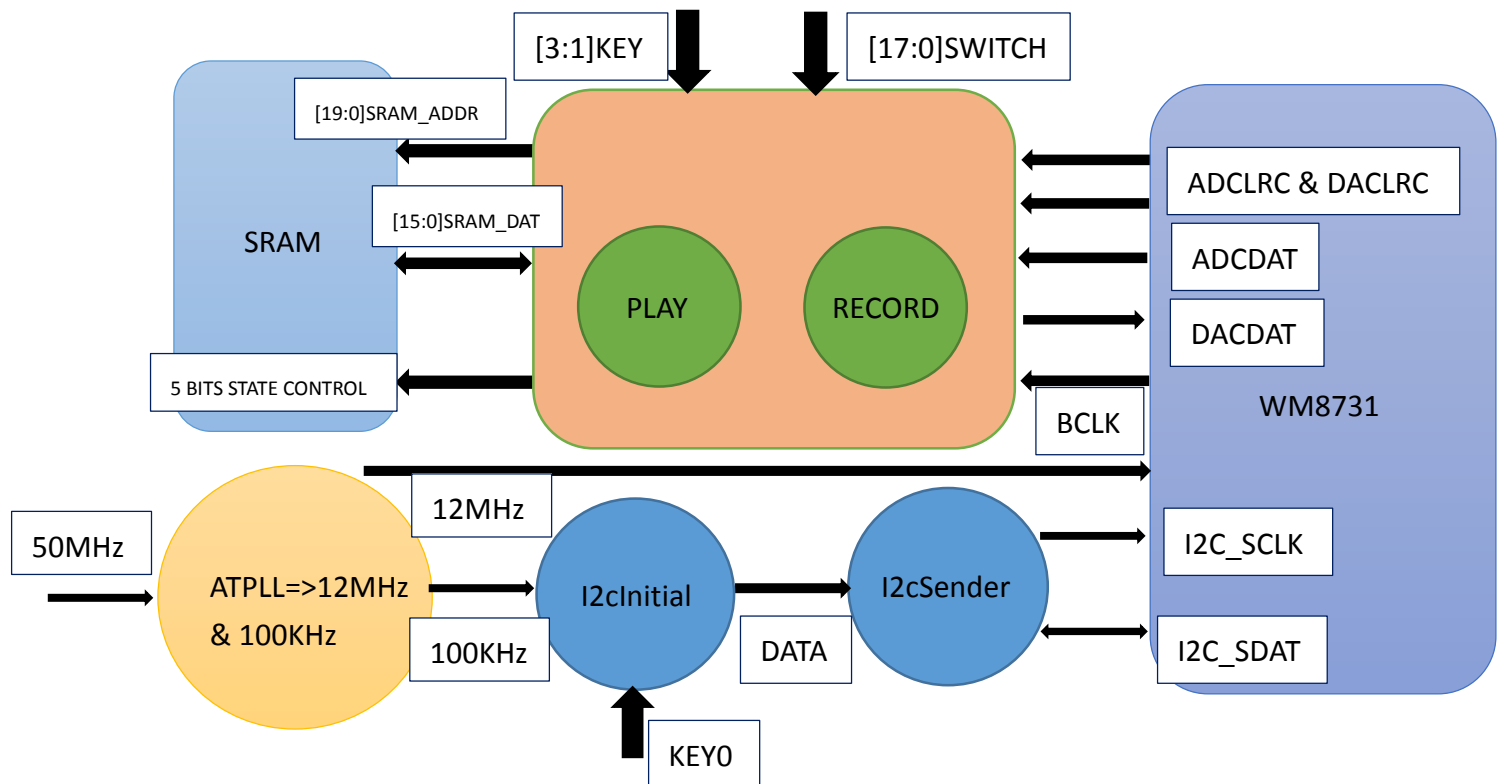
Left Line In	000_0000_0_1001_0111
Right Line In	000_0001_0_1001_0111
Left Headphone Out	000_0010_0_0111_1001
Right Headphone Out	000_0011_0_0111_1001
Analogue Audio Path Control	000_0100_0_0001_0101
Digital Audio Path Control	000_0101_0_0000_0000
Power Down Control	000_0110_0_0000_0000
Digital Audio Interface Format	000_0111_0_0100_0010
Sampling Control	000_1000_0_0001_1001
Active Control	000_1001_0_0000_0001



每次傳送初始化資料時，都是 SDAT 先由高電位降成低電位、SCLK 才跟著由高電位降成低電位。結束傳送時 SCLK 先回到高電位，SDAT 才隨之回到高電位。另外，由於 WM8731 晶片每收到 8 個 bit 就會經由 I2C_SDAT 傳送一個 ACK 回來到 I2cSender，所以注意要在每傳送 8 個 bits 後將 I2cSender 中連往 I2C_SDAT 的 inout logic 設為 1'bz，以將傳送權交給 WM8731，在上圖中就是 o_sdat 電位介於 0 跟 1 之間的地方。

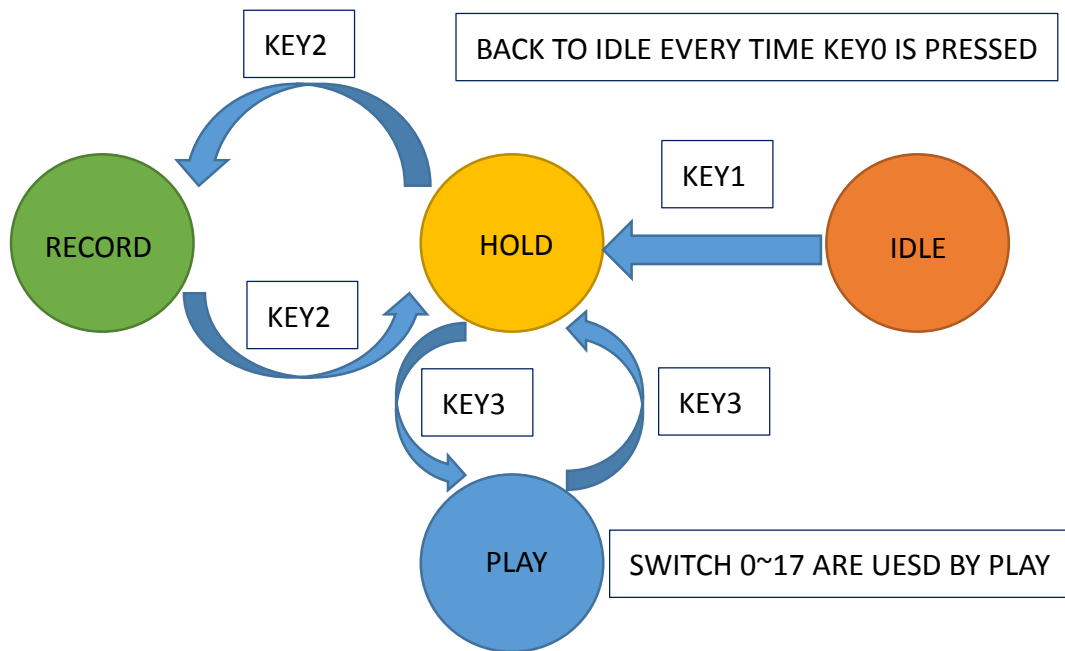
關於 SCLK 的產生則建議對於初始化資料中每一個 bit 讓 SDAT 持續三個 clock 並且讓 SCLK 在中間那個 clock 時為高電位，其餘兩個 clock 為低電位以讓 WM8731 讀取資料。

2. TOP:



在 TOP 模組中包含了 RECORD 跟 PLAY 兩個子模組，其最主要的功能就是接收 FPGA 上的按鈕和開關訊號，再對這兩個子模組進行操作。

下圖是一個範例的 TOP 狀態圖：



值得一提的是，因為 SRAM_DAT 是 inout，所以應該用以下的方式切換 SRAM_DAT 和 RECORD 或 PLAY 的连接。

```
assign SRAM_DAT = SRAM_OE_N ? RECORD_data : 1'bz;
```

```
assign PLAY_data = (!SRAM_OE_N)? SRAM_DAT : 1'bz;
```

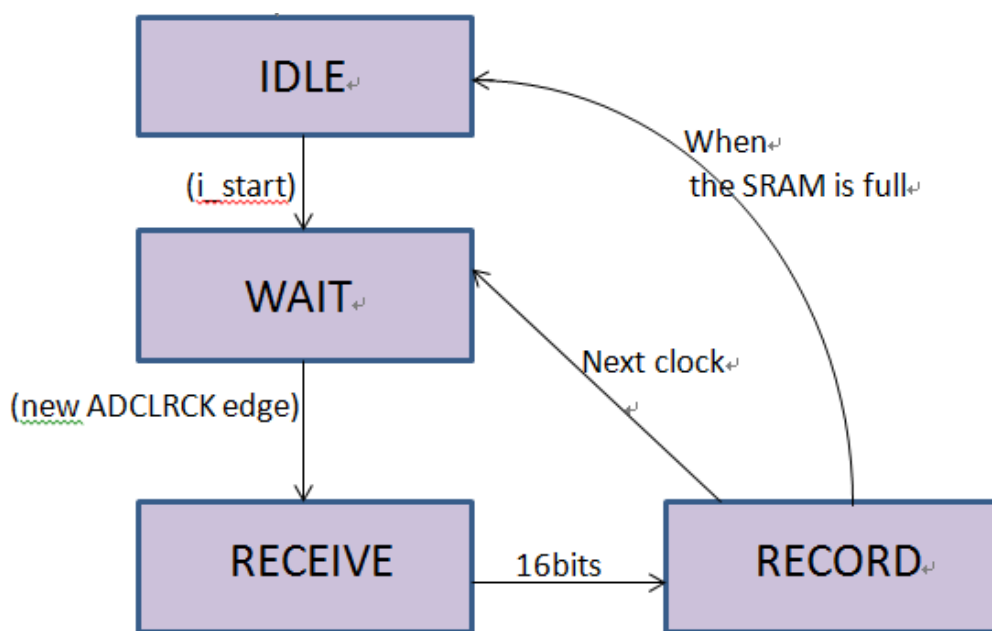
在 TOP 模組中的 RECORD 是接下來先介紹的對象。

RECORD：

RECORD 主要的部份，就是參照 I2S 的規範，做出相對應的 Finite State Machine，讓輸入的 sequential 信號，累積 16bits 後，parallel 的送往 SRAM。這邊要注意的是，ADCLRCK 和 BCLK 是由 WM8731 傳入，由於 RECORD 和 PLAY 是包在 TOP 模組中的，所以統一由 TOP 接收來自晶片的時脈後再接給這兩個子模組

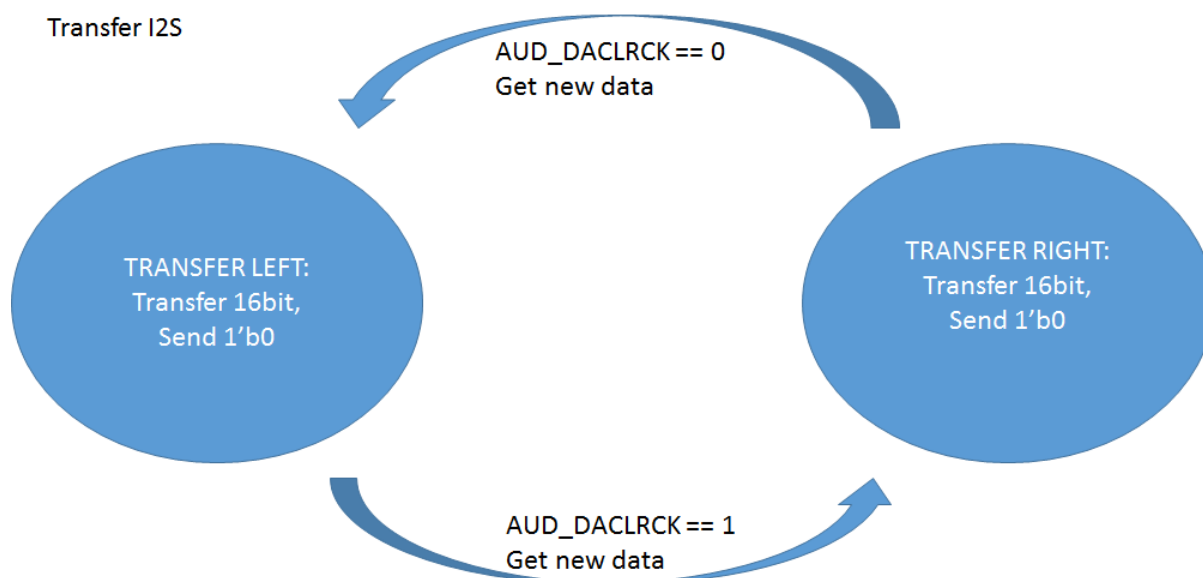
至於輸出的部份，我們只需要控制 SRAM 的 5 個 enable 信號，在適當的情況下啟動 we(由於 SRAM 接收的輸入是~we，故若是要啟動則是傳送低電位給 SRAM 相對應的 PIN 角)，再將 16bits 輸出給 SRAM 即可。

FINITE STATE MACHINE 參考：

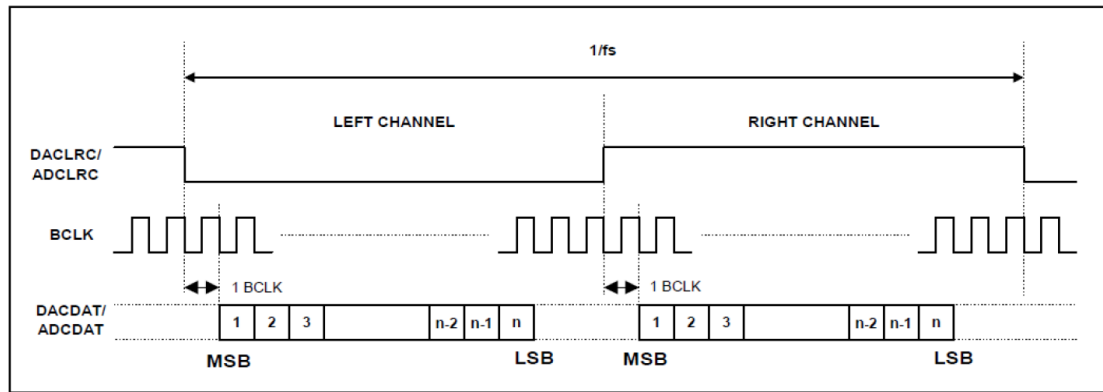


PLAY：

PLAY 模組中的子模組 DSP 負責接收外界的指令並且進行播放、暫停、快轉、慢轉等功能。



DACLRC 視為 input 訊號，DACDAT 為 output 訊號，每個 BCLK 傳送 1-bit，從 MSB 開始傳，傳送完 16-bit 後，將 output 訊號設為 0。
BCLK 為 negative edge trigger，(always_ff(negedge BCLK))



DSP:

所有與 data 運算相關的數字都要被設為 signed (logic signed [15:0] data)

接收到 pop 訊號後，才進行下一筆資料的處理

慢放

零次內插：

同一筆 data 播 n 次

一次內插：

算出內差的值(increase)， $(\text{nextdata} - \text{data})/(\text{speed})$

每次將 data 加上 increase 值

快放

升頻快放：

SRAM Address 每 N 筆資料，播放一筆

同頻率快放：

連著播 512 筆資料，再跳過 $N \times 512$ 筆資料

這種方法較容易出現雜音

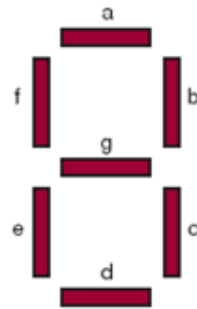
```

case (FASTorSLOW)
  SLOW: begin
    case (ZEROorONE)
      ZEROINTER: begin
        if (i_pop) begin
          if (count_r == maxcount) begin
            addr_w = addr_r + 1;
            data_w = nextdata_r;
            nextdata_w = i_SRAM_DAT;
          end else begin
            count_w = count_r + 1;
          end
        end else begin
          end
        end
      end
      ONEINTER: begin
        if (i_pop) begin
          if (count_r == maxcount) begin
            addr_w = addr_r + 1;
            data_w = nextdata_r;
            nextdata_w = i_SRAM_DAT;
            add_data_w = (i_SRAM_DAT - nextdata_r)/speed_r;
          end else begin
            data_w = data_r + add_data_r;
            count_w = count_r + 1;
          end
        end else begin
          end
        end
      end
    endcase
  end
  FAST: begin
    if (i_pop) begin
      addr_w = addr_r + speed_r;
      data_w = nextdata_r;
      nextdata_w = i_SRAM_DAT;
    end else begin
      end
    end
  end
endcase

```

3. DISPLAY:

Display 可以使用七段顯示器，或是 LCD 顯示器。七段顯示器的基本程式即為實驗一的內容，只需增加新的 mapping 進去就可以了(本次實驗中使用七段顯示器顯示 TOP 模組的狀態，如 IDLE、HOLD、RECORD 和 PLAY，也可以用來顯示 RECORD 和 PLAY 狀態中的秒數顯示)



要注意的是，右圖為 7 hex decoder 的編碼
其中 a 為 LSB，而非 MSB。

三. 除錯

1. 對於初始化的部分，因為 testbench 相對好掌握，建議先用 nWave 跑出波形並檢查後再燒錄測試。若是使用助教提供的 tb_i2c.sv，則應該檢查其中的 sdat 是否為 wire 而非 logic，以免不被 ncverilog 接受。
2. 注意條件式(if, else if, else)中應該對於所有的 sequential logic 都有所定義，以免在燒錄時出現錯誤。