

# Lab3 Tutorial

## 數位錄音機

### *Table of Contents*

1. 實驗目的
2. 實驗器材
3. 裝置接線
4. 整體架構設計
  - 4.1. I<sup>2</sup>C Module
  - 4.2. I<sup>2</sup>S Module
  - 4.3. DSP and Control Logic
  - 4.4. Clock
5. 使用 I<sup>2</sup>C 初始化 WM8731
  - 5.1. 信號傳輸方式
  - 5.2. ACT Bit
  - 5.3. nWave 模擬結果
6. I<sup>2</sup>S
7. Digital Signal Processing (DSP)
  - 7.1. Signed 運算
  - 7.2. 快速播放
  - 7.3. 慢速播放
8. SRAM
9. LCD
  - 9.1. Initialization
  - 9.2. Module Pins
  - 9.3. Instructions
10. 整體架構設計
11. 其他建議

## 1. 實驗目的

此實驗為設計數位錄放音機，具備錄音、播放、暫停、停止播放、加速播放(2, 3, 4, 5, 6, 7, 8 倍速)、慢速播放( $1/2$ ,  $1/3$ ,  $1/4$ ,  $1/5$ ,  $1/6$ ,  $1/7$ ,  $1/8$  倍速)等功能，可錄製的時間總長度為 32 秒。此外，慢速播放要能夠支援零次內插與一次內差兩種模式。

## 2. 實驗器材



DE2-115 FPGA 板



電源線



USB 傳輸線



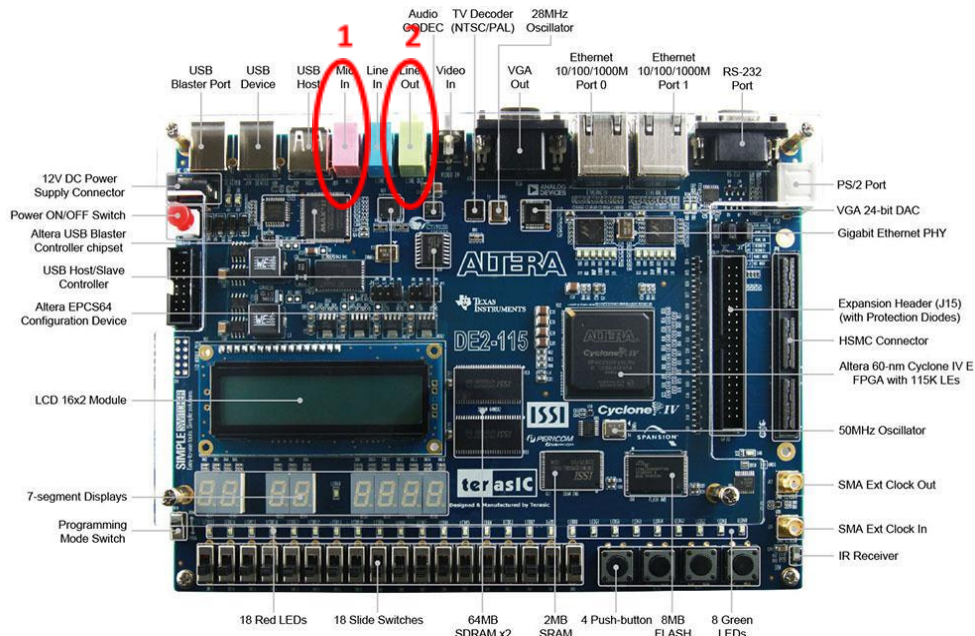
麥克風



喇叭 (或耳機)

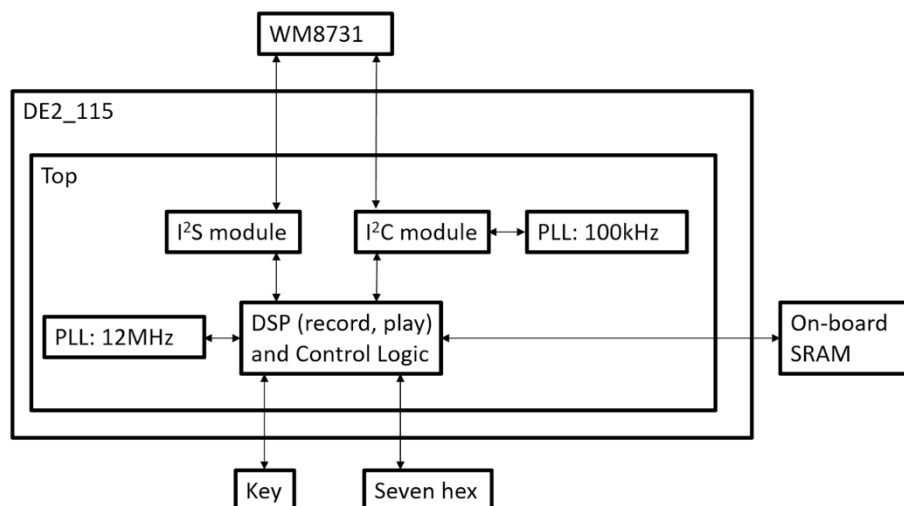
### 3. 裝置接線

將麥克風音源線插入 1 的位置，耳機或喇叭則插入 2 的位置。



### 4. 整體架構設計

下圖為一架構設計的範例供同學參考。



#### 4.1. I<sup>2</sup>C Module

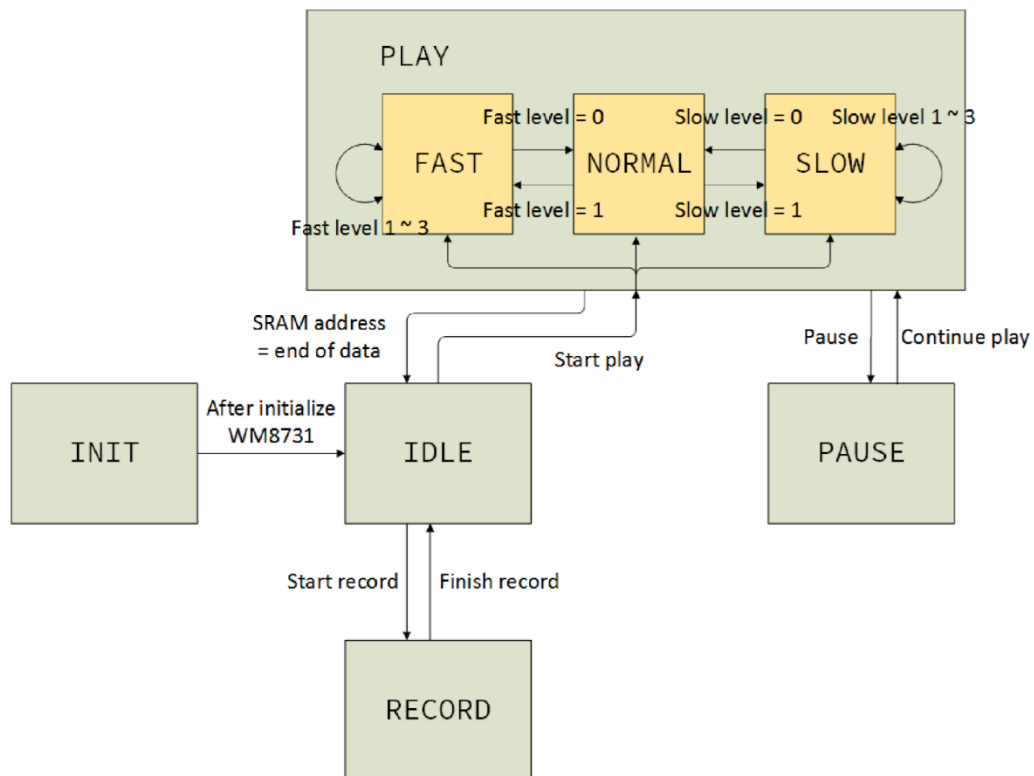
使用 I<sup>2</sup>C 傳輸協定初始化 WM8731 音效晶片。

## 4.2. I<sup>2</sup>S Module

I<sup>2</sup>S 傳輸協定是用來規範與 WM8731 音效晶片傳輸音訊資料的方法。

## 4.3. DSP and Control Logic

根據目前要錄音還是播放，來控制是從 WM8731 讀取資料寫到 SRAM 或是反過來將 SRAM 的資料讀出送到 WM8731。其中慢速播放須支援零次和一次內插。下圖為一簡單的 FSM 架構範例，僅供參考：



## 4.4 Clock

- (1) 這個實驗需要用 Qsys 合成 PLL，輸入是原本的 50MHz，輸出一個是給 I<sup>2</sup>C 用的 100kHz，另一個是給 WM8731 用的 12MHz。這些訊號產生的模組請拿 Qsys 生成的模組，請不要自己寫除頻電路。
- (2) 當 I<sup>2</sup>C 初始化完成後，要將 12MHz 的 clock 訊號送給 AUD\_XCLK，WM8731 就會生成 BCLK 以及兩個 LRCLK。
- (3) I<sup>2</sup>S 在收或傳資料時會需要用到 BCLK 控制，其他 DSP 跟控制用 FSM 從 BCLK、12MHz 或原本的 50MHz 則一使用即可。

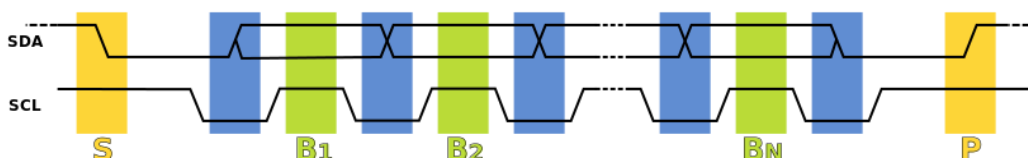
## 5. 使用 I<sup>2</sup>C 初始化 WM8731

依序將下表這 7 筆資料寫入來進行初始化：

<b>Reset</b>	0011_0100_000_1111_0_0000_0000
<b>Analogue Audio Path Control</b>	0011_0100_000_0100_0_0001_0101
<b>Digital Audio Path Control</b>	0011_0100_000_0101_0_0000_0000
<b>Power Down Control</b>	0011_0100_000_0110_0_0000_0000
<b>Digital Audio Interface Format</b>	0011_0100_000_0111_0_0100_0010
<b>Sampling Control</b>	0011_0100_000_1000_0_0001_1001
<b>Active Control</b>	0011_0100_000_1001_0_0000_0001

### 5.1. 信號傳輸方式

I<sup>2</sup>C 的訊號傳輸仰賴兩條信號線：SDA 與 SCL (SCLK)。SCL 掌控資料傳輸的「狀態」，而 SDA 則代表要傳輸的「資料」本身。發送資料端控制 SCL 訊號線告訴接收端 SDA 的資料是否已穩定可被讀取，1 代表資料穩定，0 則是正在改動資料。SDA 除了代表被傳送的資料外，也負責標示傳送過程的開始與結束。下圖為傳輸信號線控制方式示意圖：

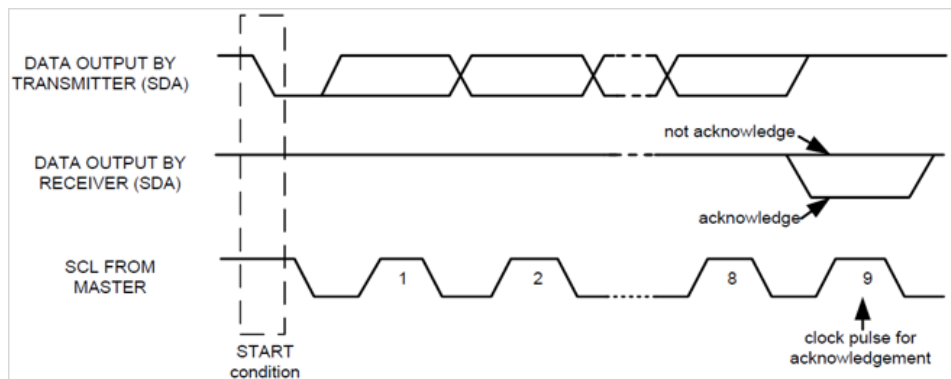


其中整個傳輸過程可以分成三個階段：起始(S)、傳送(B)、終止(P)。

	起始	傳送(更改)	傳送(被讀取)	終止
<b>SDA</b>	1 → 0	X → X'	X'	0 → 1
<b>SCL</b>	1	0	1	1

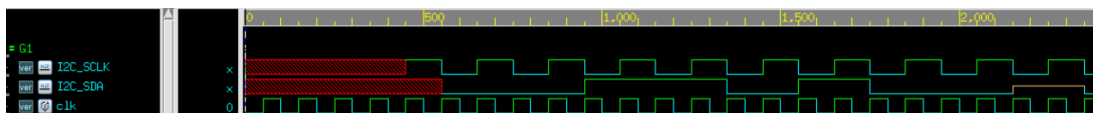
## 5.2. ACT Bit

除了上述過程外，每送出 8 個 bit，還需要把 SDA 設成 high impedance (1'bz)，使接收端可以回傳 ACT (acknowledgement bit)來告訴傳輸者是否已正確接收到訊息，如下圖所示：

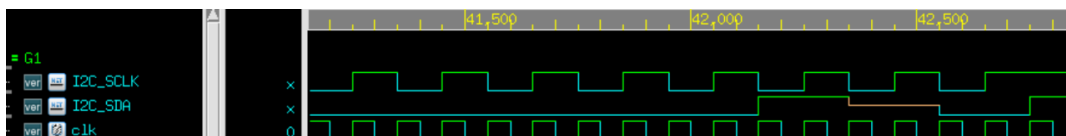


## 5.3. nWave 模擬結果

(1) 開始用 I<sup>2</sup>C 傳資料 (clk 為 100 kHz，且 SCL 的 duty cycle 可以不用跟下面一樣是 50%)，注意最後面傳完 8 bits 會有 acknowledge。

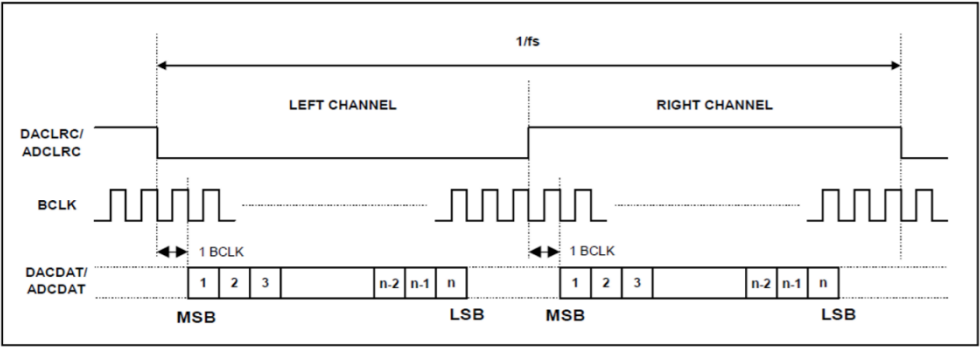


(2) 資料傳輸完畢



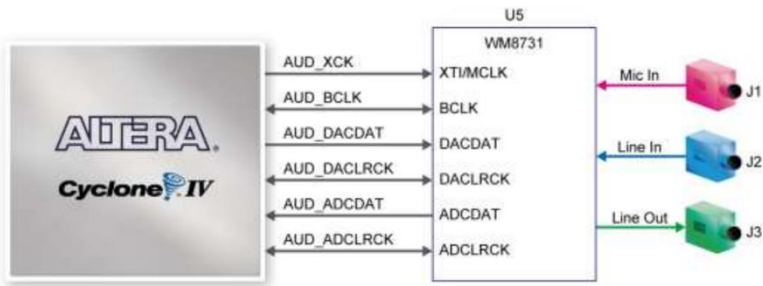
6. I<sup>2</sup>S

I<sup>2</sup>S 傳輸協定如下圖所示，DACLRCK 表示使用者將處理後的音訊送進數位類比轉換器，而 ADCLRCK 則代表使用者由類比數位轉換器接收音訊。在一個 clock 中，低電位表示左聲道，高電位表示右聲道。當 LRC 改變後等經過一個 BCLK 後即可開始記錄資料；位元數 n 可自由設定，通常為 16；而傳輸順序是從 MSB 到 LSB。



在錄音時，由於左右聲道是相同的，所以我們可以只取一個聲道就可以了。但需特別注意的是，傳輸完成後後面會有數個 BCLK 的時間會繼續給 0，因此要藉由 LRC 的改變來判斷聲道切換，而不要直接計數 n 次之後就跳轉至另一聲道做讀取。

下圖為 WM8731 各腳位 ( I/O 訊號線 ) 的功能說明：



Signal Name	FPGA Pin No.	Description	I/O Standard
AUD_ADCLRCK	PIN_C2	Audio CODEC ADC LR Clock	3.3V
AUD_ADCDAT	PIN_D2	Audio CODEC ADC Data	3.3V
AUD_DACLCK	PIN_E3	Audio CODEC DAC LR Clock	3.3V
AUD_DACDAT	PIN_D1	Audio CODEC DAC Data	3.3V
AUD_XCK	PIN_E1	Audio CODEC Chip Clock	3.3V
AUD_BCLK	PIN_F2	Audio CODEC Bit-Stream Clock	3.3V
I2C_SCLK	PIN_B7	I2C Clock	3.3V
I2C_SDAT	PIN_A8	I2C Data	3.3V

## 7. Digital Signal Processing (DSP)

### 7.1. Signed 運算

一階內插時要注意正負號正負的運算，如以下範例：

```
logic signed [7:0] a, b, c;
c = $signed(a) + $signed(b);
```

### 7.2. 快速播放

藉由間隔取樣點讀取，我們就可以增加播放速度。根據訊號處理的簡單定律，時域縮放(time domain scaling)將訊號時間長度減小，會使得頻域(frequency domain)變寬，也就是聽覺上的音調變高。如果要試試維持音調不變，可以在一段時間內，捨棄後半段的資料，不過本實驗並不要求需要維持音調不變。

### 7.3. 慢速播放

在取樣點兩兩間插入資料即可降低播放速度。本實驗需要實作以零次內插(內差多出的資料與上個取樣點一樣)及一次內插(資料為前後取樣點線性內插)完成慢速播放。

## 8. SRAM

在本實驗中，CE\_N、LB\_N、UB\_N 的值皆設為 0。DE2-115 上記憶體字元長度(word length)為 16bit，共有可存 1024k 個字元，總容量約為 2MB。若以單聲道 16bit、32kHz 設定音訊資料規格，共可記錄長度為  $1024000/32000 = 32$  秒的聲音訊號。另外，SRAM 讀取與寫入均為 16bit 並列(parallel) 格式，恰巧與音訊晶片的取樣位元深度相同，因此每一筆取樣資料即為一個字元的長度。

另外可以注意的是，音效晶片數位端以序列(serial)方式傳輸 2's complement 格式的二進位資料，而 SRAM 則是以並列(parallel)格式傳輸。換句話說，從音效晶片依續讀入或讀出共 16 個 bit 的數位訊號，可以用一個 cycle 就寫入 SRAM 或從 SRAM 讀取。



## 9. LCD

### 9.1. Initialization

LCD module 開機之後需要一段時間來初始化，並且把需要的設定值 (Instructions) 傳入。詳細時程可參考助教投影片或使用說明書。

### 9.2. Module Pins

LCD\_DATA[0]~[7] 分別是放 Instruction Codes 或 DDRAM/CGRAM Address。

LCD_EN	有點類似 I2C 裡面 SCL 的作用，不過是在 negative edge 時，才讓 LCD 讀資料
LCD_RW	控制讀寫資料，如果只是要顯示可以直接設為 0 (WRITE)
LCD_RS	傳 Instructions 的時候設為 0，傳 Data (CGRAM) 的時候設為 1
LCD_ON	設為 1 是開機
LCD_BLON	設為 0 關掉 Back Light

### 9.3. Instructions

詳細資訊可參考投影片中的 Instruction Table，比較需要的基本設定如下：

FUNCTION_SET	8'h38
DISPLAY_ON	8'h0C
CLEAR_DISPLAY	8'h01
ENTRY_MODE_SET	8'h06
DDRAM_ADDR_SET	8'h80
RETURN	8'hC0

## 10. 訊號電平(Level)

因為錄音機以數位方式處理訊號，當超過位元深度(如 16bit)上限的話會被轉換為相同的值，會產生俗稱的「破音」。因此可以適當方式提醒使用者輸入的音量過大，例如可以用 DE2-115 的 LED 燈數量顯示目前音量大小。

## 11. 其他建議

這個實驗要盡量把每個部分都分成一個 submodule 來處理，會比較容易 debug。每個 submodule 完成後最好也要自己寫個簡單的 testbench 測試一下 I/O 的運作情況是否如預期，如此一來之後全部合起來一起跑的時候才比較不會出錯。