

數位電路實驗Lab3 report

錄音機

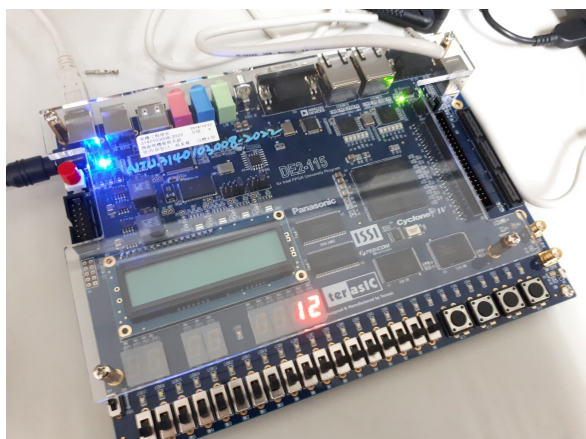
組別：team09

組員：鐘民憲(B06901017)

吳睿哲(B06901018)

謝兆和(B06901026)

一、使用器材與架設方式



FPGA板



筆電



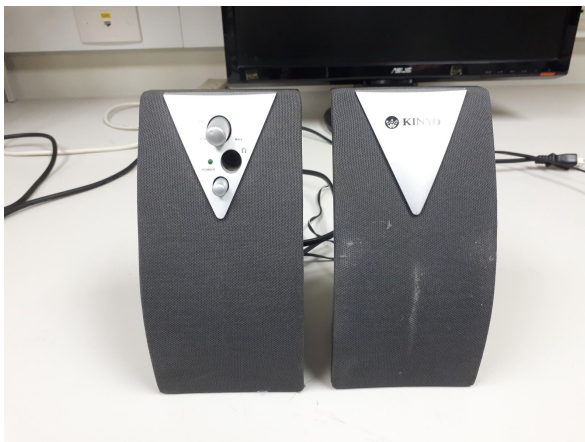
傳輸線



電源線



麥克風



喇叭

二、使用方式與詳細步驟

我們的錄音機具備的功能：錄音及播放

在錄音時可進行的操作：開始錄音、暫停錄音、結束錄音

在播放時可進行的操作：開始播放、暫停播放、結束播放、快速播放(最高15倍速)、慢速播放(最低1/15倍速)(慢速播放有零次內插和一次內插兩種模式)

(1)錄音：

- 1.按下key3進行reset，放開後會自動進行initialize
- 2.按下key0開始錄音，再按一次可以暫停，再按一次繼續，停止錄音為key1。

(2)播放：

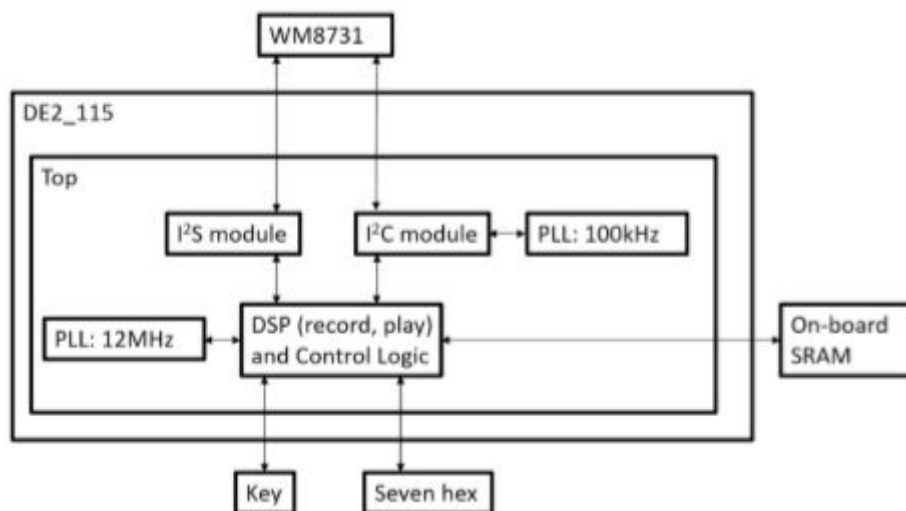
- 1.按下key0開始播放，再按一次可以暫停，再按一次繼續，停止播放為key1。
- 2.在播放暫停時，可以改變播放模式與播放倍數，按key1可改變播放模式，播放模式有：0(原速),1(加速),2(慢速0階interpolation),3(慢速1階interpolation)，而SW[3..0]可以改變播放倍數，以二進位制表示，例如：SW[3..0]={0,1,0,1}即代表播放速度為5倍速，而SW[3..0]={0,0,0,1}則是1倍速。

三、實作設計技術細節及巧思

(1)Hierarchy：

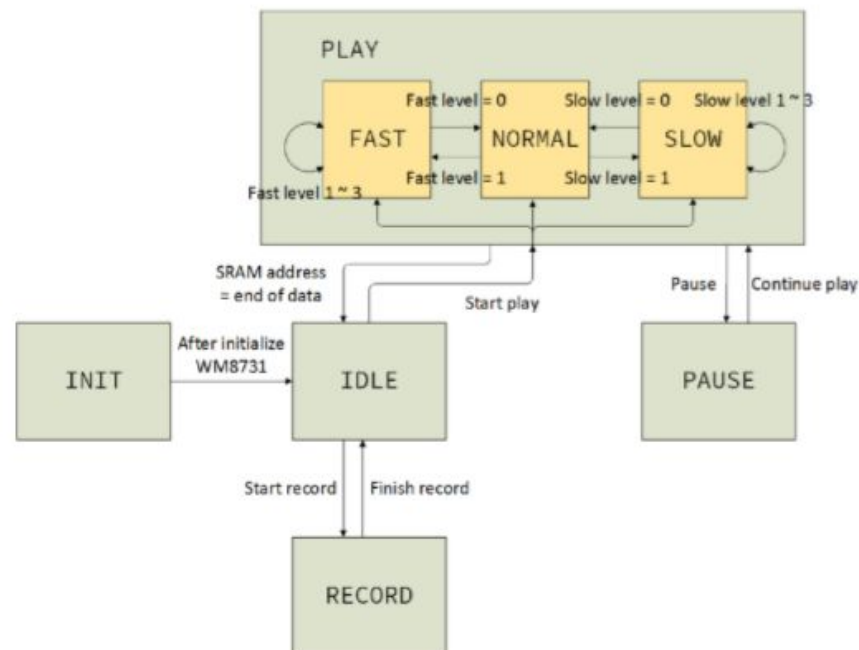
- Qsys //產生12MHz,800kHz及100kHz三種clock
- DE2_115 //對應FPGA的接腳
- Debounce //按鍵緩衝
- Top //實際錄音機的控制，處理各submodule之間的訊號溝通
 - I2cInitializer //用I2C通訊協定去初始化WM8731音效晶片
 - AudDSP //將資料從SRAM取出然後根據要求進行加速或減速，然後傳給AudPlayer
 - AudPlayer //將從AudDSP傳來的資料送入WM8731音效晶片
 - AudRecorder //將晶片處理完產生的資料傳給AudDSP去存入SRAM

(2)Structure Diagram：

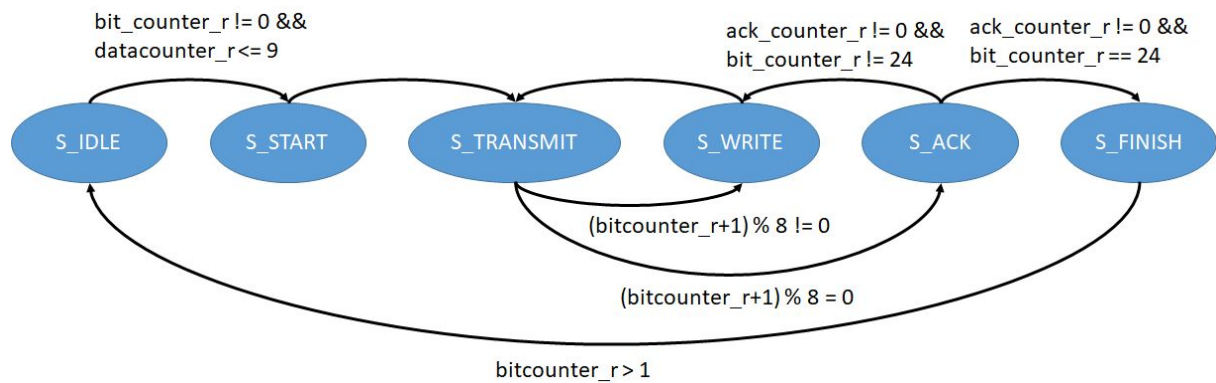


(3) Finite State Machine

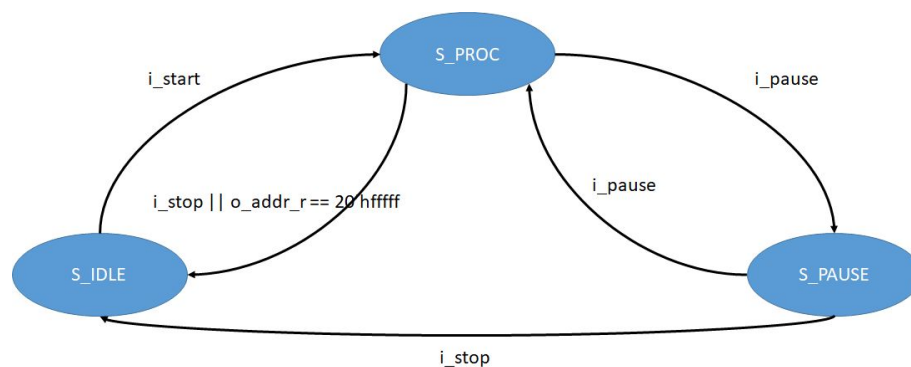
1. FSM of Top :



2.FSM of I2cInitializer : 共輸出10筆初始化資料，每輸出一筆進入S_FINISH, 此時 o_finish = 0, 直到全部資料傳輸完畢，o_finish 才會 = 1, 進入 recording state.

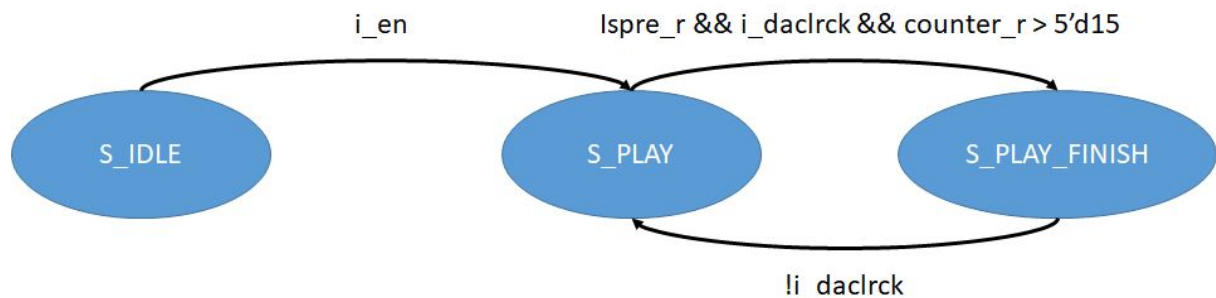


3.FSM of AudDSP :

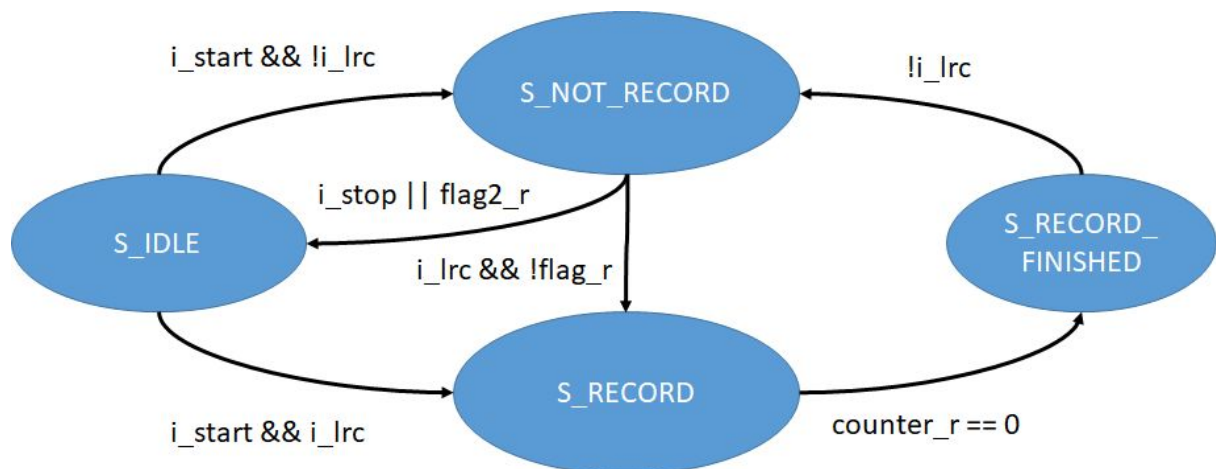


S_PROC 內分成4個modes: mode 0 正常速度播放, mode 1 快速播放, mode 2 零階內插慢速, mode 3 一階內插慢速。0階內插是直接輸出 sram_data i_speed times, 1階內插計算上要吃2筆資料, 皆可即時輸出。另外, 播放到一半可以調整播放速度及 mode, 無須pause. 快/慢支持2到15倍速, 音高無法維持不變。

4.FSM of AudPlayer :



5.FSM of AudRecorder :



四、碰到的問題或挑戰與解決方式

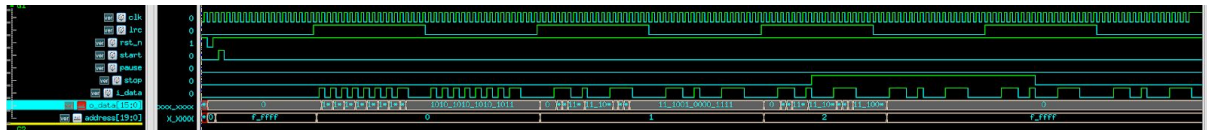
(1)有很長的一段時間(好幾天), 我們Quatus在compile時會一直跳出相同的warning:No exact pin location assignment for 480 pins of 480 total pins。也就是說我們的所有pin都沒有接到, 可是看code都完全沒有任何問題, 後來我們研究lab2的資料夾儲存方式, 發現要把qsf檔跟DE2_115專案檔放在同一層才行, 該死的Quatus。

(2)助教雷, 助教寫的DE2115.sv有錯, 裡面的key3down不知道用來幹嘛, 使得initializer沒有被reset, 還有在Top.sv的always_ff裡使用的clock不是最快的clock。另外, Lab3 tutorial 中 I2C initialization data 有 7 筆, 但我們請教了大四學長, 發現要傳10筆才能初始化成功。

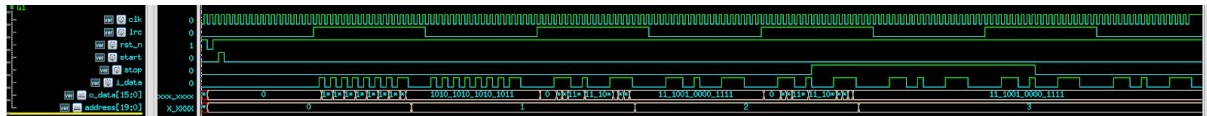
(3)自己寫testbench去測試各個submodules往往沒辦法測到所有可能的情况, 也有可能自己假想的情况與實際上的情况不同, 因此就算順利通過testbench, 看nWave都沒什麼問題, 有可能在實際燒錄後還是無法正常運行, 又或者是運行結果非常奇怪。另外,

(4)在將code燒入板子後, 就無法像模擬一樣去對每個signal去看wave, 因此解決方式就是使用板子上的顯示裝置去顯示signal, 如led、七段顯示器等。

(5)我們對於SRAM的寫入抱有疑問，理論上在WE_N=0的時候可以寫入，而我們在record state的時候WE_N一直都為0，所以應該是在持續寫入的狀態。以下為我們自己寫testbench去測試AudRecorder.sv跑出來的waveform：



S_IDLE我們將address設為ffff(最後一個地址)，然後每當lrc變為1要吃data時，將address加1，最後再次將address設回ffff。然而這樣寫會導致SRAM吃不進任何data。因此我們改寫AudRecorder.sv使其waveform變為如下：



S_IDLE我們將address設為0，每次data吃完隔一個clk再將address加1，然而最後不將address設回0。這樣寫就ok了。

總結來說，造成error的可能原因有二：1.初始值設定問題。2. data吃完後address要多維持一個clk。不過由於已經做得心很累了，所以就懶得去驗證了。

五、Demo

<https://youtu.be/4FGhS2W4xYU>