

NTUEE DCLAB

LAB 3: 數位錄音機

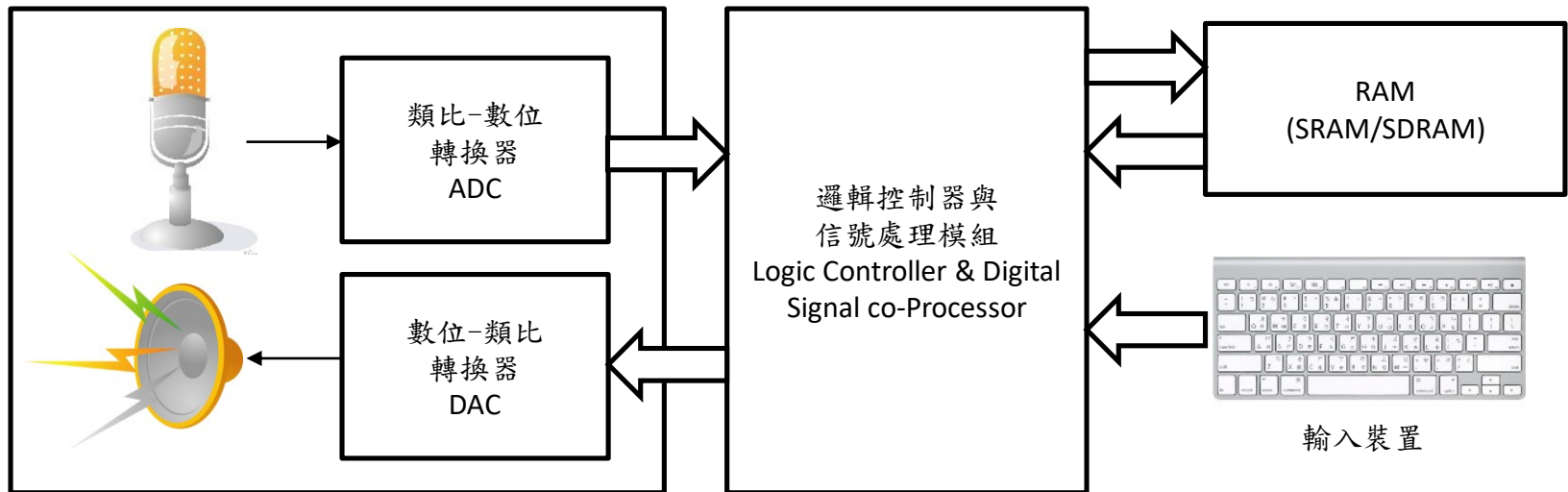
Graduate Institute of Electronics Engineering
National Taiwan University

Outline

- Introduction
 - Lab requirements
- Audio CODEC
- Memory devices
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Report regulations

Introduction

- 數位錄音機
 - 運用麥克風與電腦喇叭連接 FPGA 板的 Audio CODEC 模組(內含 ADC 與 DAC)
 - 對音訊資料進行訊號處理
 - 對記憶體模組進行存取



Lab Requirements

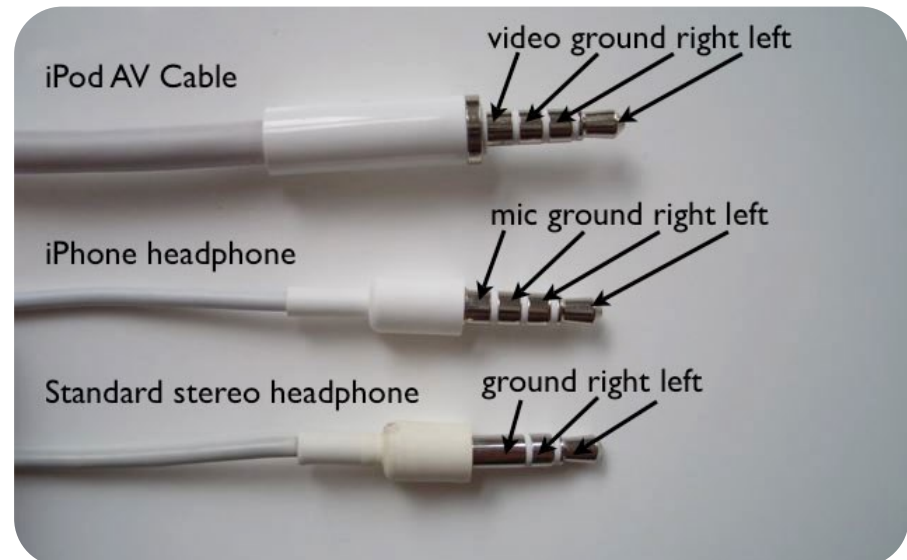
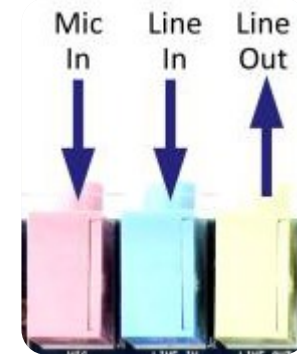
- 需具備下列功能
 - 可錄音、播放
 - 播放時可以暫停、停止
 - 取樣值為16-bit signed，可錄製時間達32秒
 - 需支援快速播放(2, 3, 4, 5, 6, 7, 8 倍速)以及慢速播放($1/2$, $1/3$, $1/4$, $1/5$, $1/6$, $1/7$, $1/8$ 倍速)
 - 慢速播放時要包含零次內插與一次內插兩種模式
- Bonus (demo時與report中皆應清楚詳細說明)
 - 使用其它模組顯示錄音機運作狀態
 - 使用SDRAM增加可錄製時間
 - 以訊號產生器和示波器展示不同內插模式下的波形
 - 其他訊號處理功能等

Outline

- Introduction
 - Lab requirements
- **Audio CODEC**
- Memory devices
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Report regulations

Audio Signal & Connectors

- An audio signal is representation of sound
 - Usually as an electrical voltage
- Connection on devices
 - Line in, line out and mic in
- Phone connectors
 - Cylindrical in shape
 - With 2~4 contacts



Audio CODEC

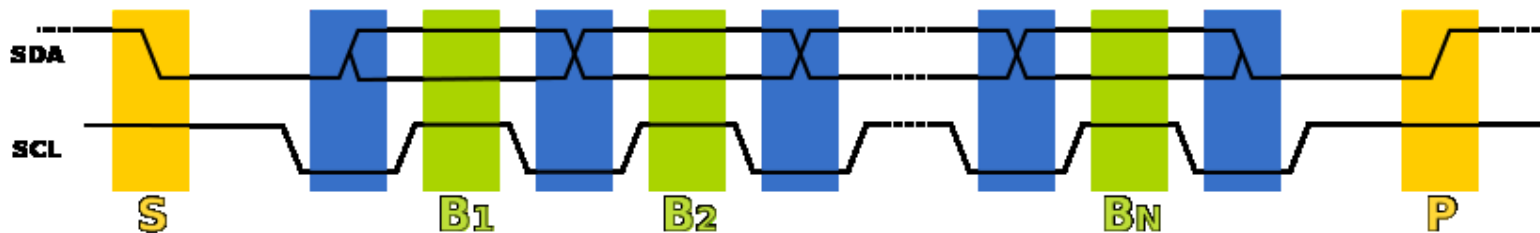
- WM8731
 - IP for audio transmission
 - 32kHz sampling rate
 - 16-bit audio data input
- Usage
 - Initialize by setting registers via I²C interface
 - After successful initialization, receive or transmit audio data via I²S interface

Initialization with I²C

- I²C (Inter-Integrated Circuit) protocol
 - Referred to as “I-squared-C”
- Serial Data Line (SDA)
 - Data being send
 - 1 bit at a time
- Serial Clock (SCL)
 - Control whether data is valid
 - 0 for data changing, 1 for data ready

I²C Protocol

- **S:** initiate data transfer
 - SDA pulls to 0 while SCL stays at 1
- **Blue:** SDA sets transfer bit when SCL is 0
- **Green:** data is sent when SCL is 1
- **P:** end of transfer
 - SDA pulls to 1 while SCL stays at 1



	起始	傳送(更改)	傳送(被讀取)	終止
SDA	1 → 0	X → X'	X'	0 → 1
SCL	1	0	1	1

Acknowledge

- For every 8 bits data sent
 - Set SDA to high impedance (1 cycle should be enough)
 - Allow receiver to return acknowledgement bit (0)

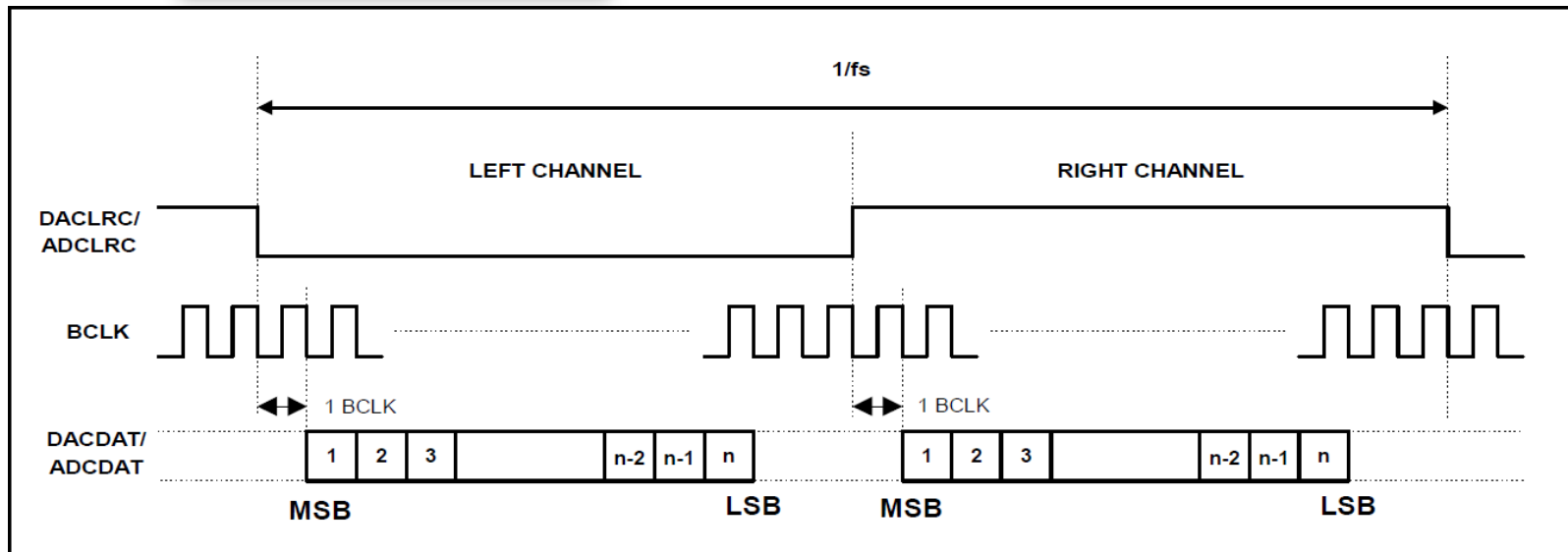
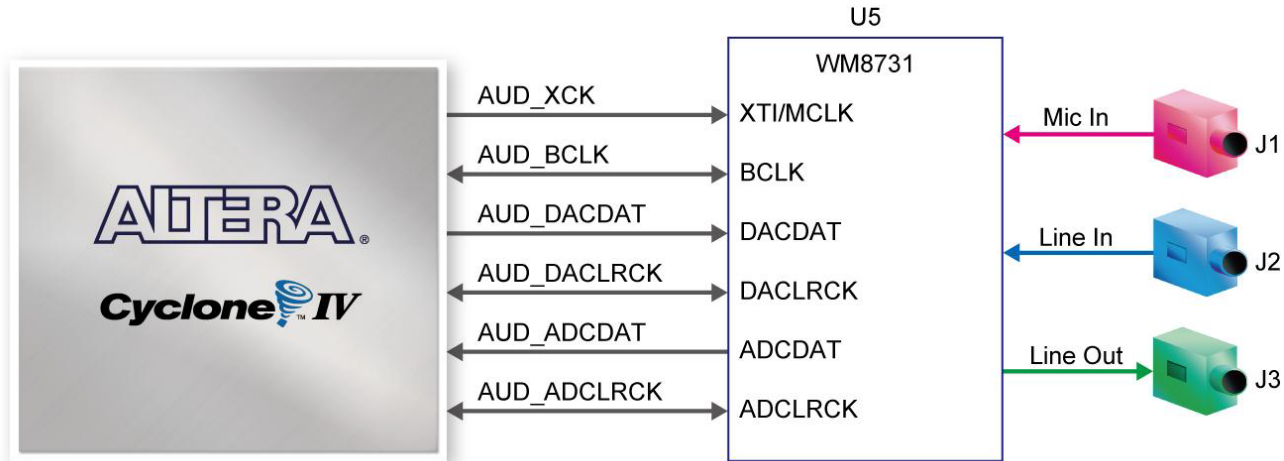
```
module inout_port(oe, clk, SDA);  
  
    input  oe; // output enable  
    input  clk;  
    inout  SDA;  
  
    logic a; // output data  
    logic b; // input data  
  
    assign SDA = oe? a: 1'bz;  
  
    always @(posedge clk) begin  
        b <= SDA;  
    end  
  
endmodule
```

Initialization Setting

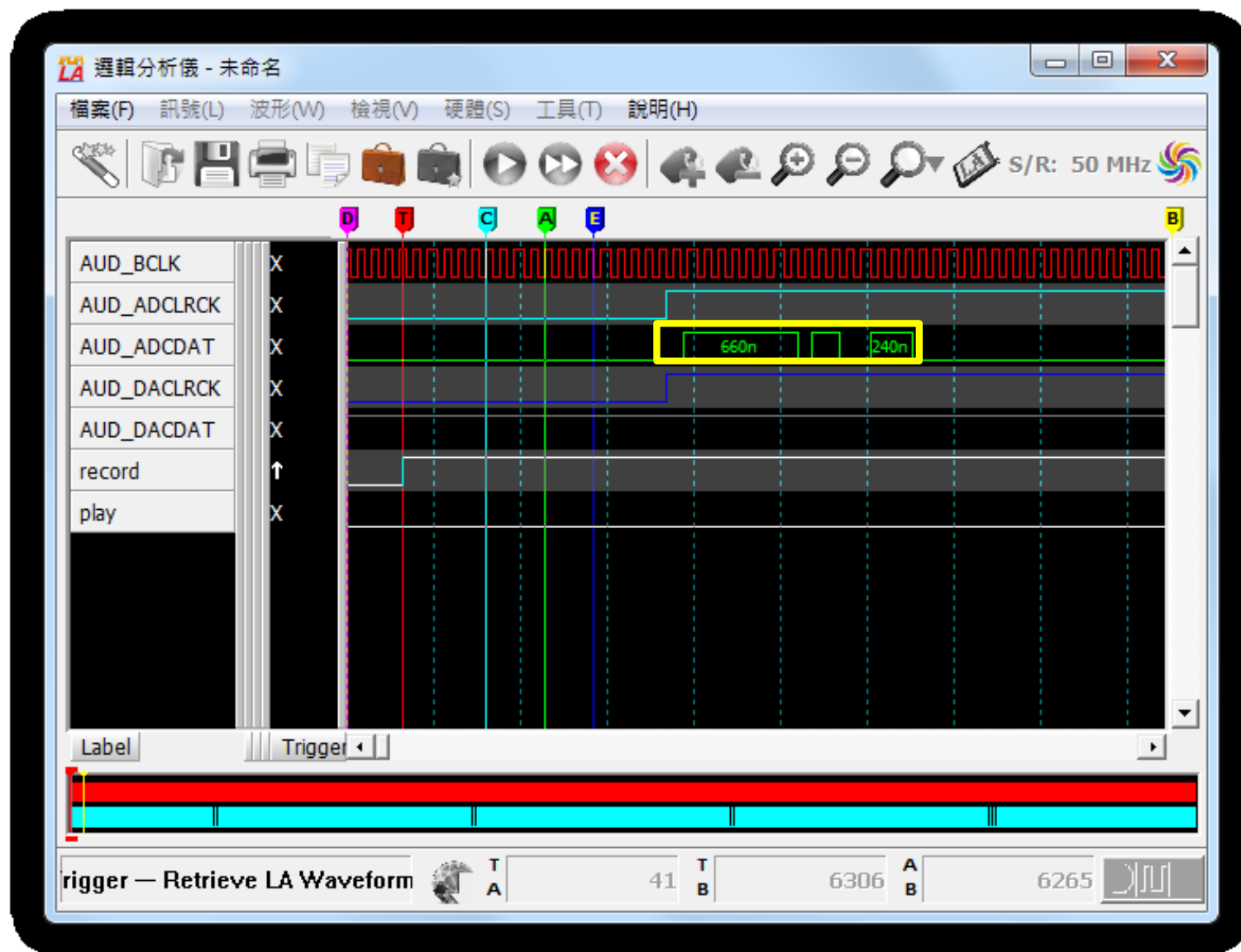
Reset	0011_0100_000_1111_0_0000_0000
Analogue Audio Path Control	0011_0100_000_0100_0_0001_0101
Digital Audio Path Control	0011_0100_000_0101_0_0000_0000
Power Down Control	0011_0100_000_0110_0_0000_0000
Digital Audio Interface Format	0011_0100_000_0111_0_0100_0010
Sampling Control	0011_0100_000_1000_0_0001_1001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

WM8731 Audio Operations

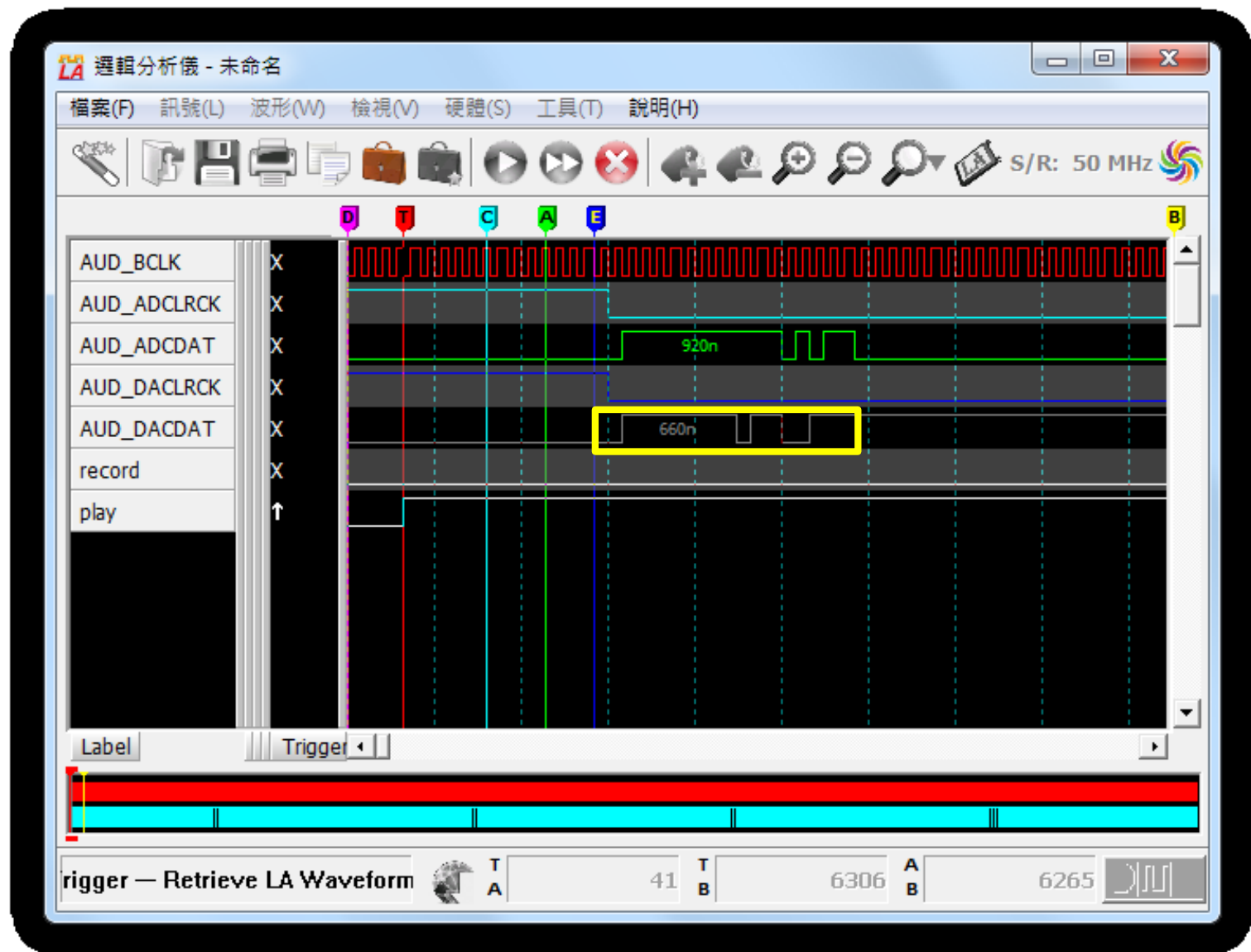
- Programmed to have 16-bit data ($n = 16$)



Digital Audio Interface – Record



Digital Audio Interface – Play



Outline

- Introduction
 - Lab requirements
- Audio CODEC
- **Memory devices**
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Report regulations

Memory Devices

- 本實驗主要會使用到的為SRAM
 - 2MB organized as 1024K words by 16 bits
 - 以WM8731取樣頻率32kHz計算，單聲道可存32秒音訊
- 需要操作的訊號
 - `SRAM_ADDR[19:0]`決定要讀或要寫的位址
 - `SRAM_DQ[15:0]`為輸入輸出雙向皆可操作，寫值時直接用，讀值時要設成1'bz

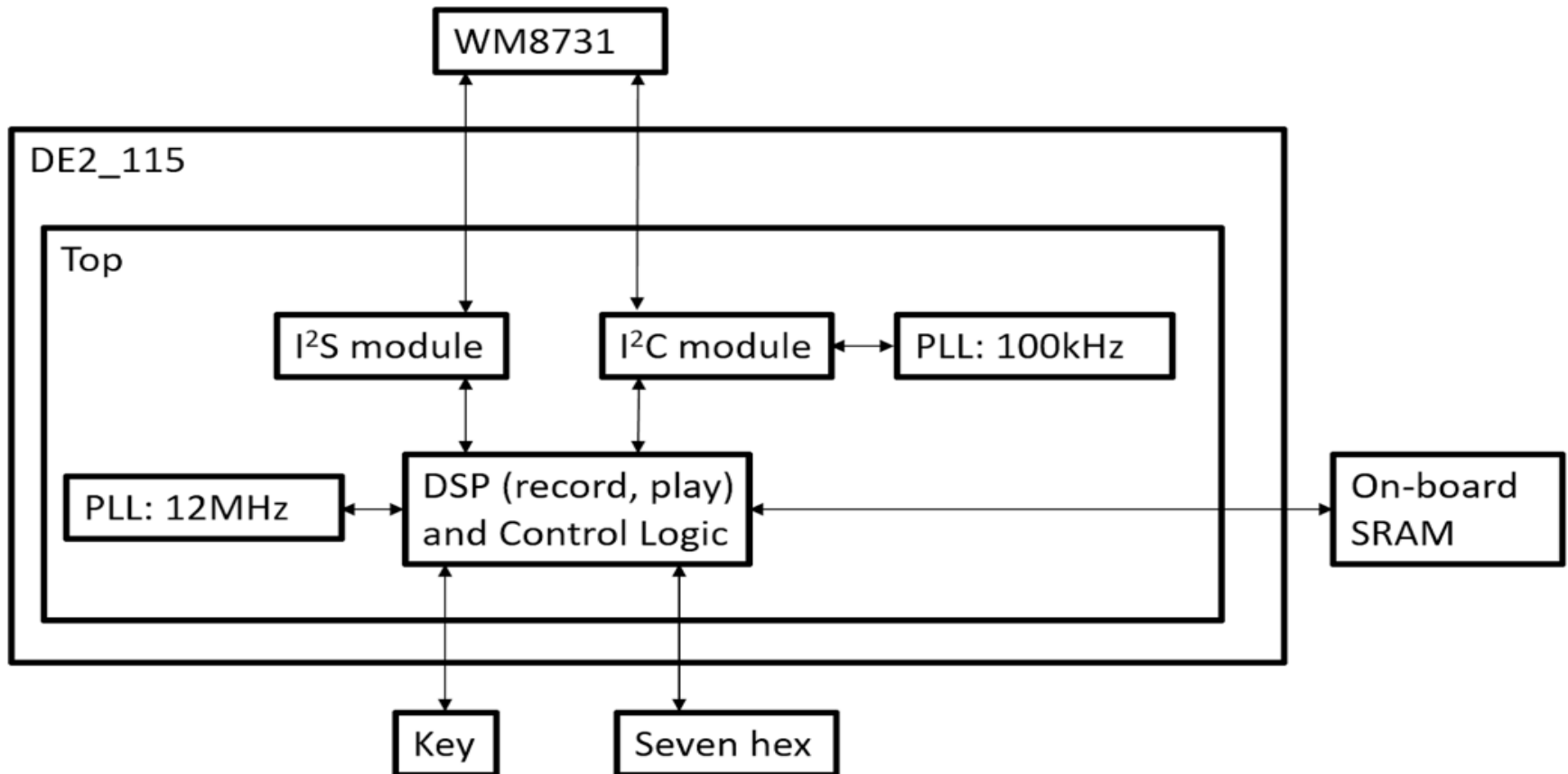
```
assign io_SRAM_DQ  = (state_r == S_REC'D) ? data_record : 16'dz; // sram_dq as output
assign data_play    = (state_r != S_REC'D) ? io_SRAM_DQ : 16'd0; // sram_dq as input
```

- `SRAM_WE_N`設定目前操作模式，0為寫，1為讀

Outline

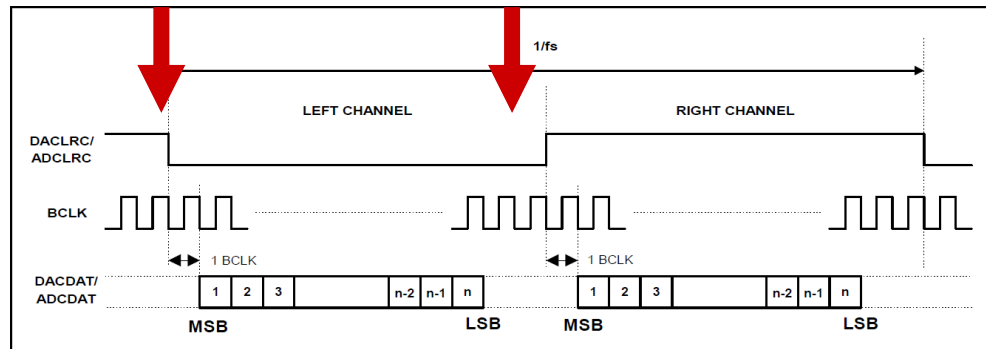
- Introduction
 - Lab requirements
- Audio CODEC
- Memory devices
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Report regulations

System Architecture



Work with WM8731

- I²C Module
 - 初始化WM8731
 - 使用100kHz clock運作
- I²S Modules
 - 接收與傳輸音訊，皆只需要處理其中一個聲道即可
 - 注意
 - 資料傳送前要先等一個cycle
 - 資料傳完後(16 cycles)後面還會有若干cycle才會切換LRC



Digital Signal Processing (DSP)

- 以signed訊號進行運算
 - logic signed [7:0] a, b, c;
 - $c = \text{\$signed}(a) + \text{\$signed}(b);$
- 快速播放
 - Down sampling
 - 以不同取樣間格達到不同倍數加速
- 慢速播放
 - Up sampling
 - 零次內插(piecewise-constant interpolation)
 - 內插資料與前一資料點相同
 - 一次內插(linear interpolation)
 - 內插點為前後點線性組合

Clock

- 用 Qsys 合成 PLL (請參考lab2做法)
 - 輸入是原本的 50MHz
 - 輸出一個是給I²C用的100kHz，另一個是給WM8731用的12MHz
 - 請不要自己用counter寫除頻電路
- 當I²C初始化完成後，將 12MHz 的clock訊號送給AUD_XCLK，WM8731就會生成BCLK以及兩個LRCLK
- I²S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

Outline

- Introduction
 - Lab requirements
- Audio CODEC
- Memory devices
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Report regulations

Code Template

- Top.sv
 - 包含一些模組分割範例
 - I2cInitializer: 以I²C初始化WM8731
 - AudDSP: 負責快速與慢速播放資料點處理
 - AudPlayer: 以I²S將DSP處理後的音訊資料傳出
 - AudRecorder: 以I²S接收音訊資料儲存到SRAM
 - 可以自行改變設計
- 建議事項
 - 設計testbench來單獨測試各個module運作情況
 - 確認無誤後才合併起來

Outline

- Introduction
 - Lab requirements
- Audio CODEC
- Memory devices
- Implementation
 - System architecture
 - Work with WM8731
 - Digital signal processing (DSP)
 - Clock
- Code template
- Submission
 - Report regulations

Submission

- 繳交項目
 - Report(.pdf)
 - source code(.v)
- 繳交檔案架構

```
team01_lab1
|-team01_lab1_report.pdf
|-src
|  |-<all of your verilog code>.v
```

- 先創一個teamXX_labX的資料夾再壓縮，不要直接壓縮繳交項目

Submission Rules

- 將**teamXX_labX**包成一個**zip**上傳到實驗室 NAS 各組的 submission 資料夾
 - 命名方式：teamXX_labX.zip、teamXX_final.zip，全小寫，組別數字 2 digit
 - Src 內的 verilog 可自行命名，只要在 report 中說明架構即可
 - Lab繳交期限：demo當天午夜
 - 若未遵守繳交格式會酌情扣分

Report Regulations

- 內容應包含
 - 層級架構
 - Block Diagram (必須包含Data Path，control signal可有可無)
 - FSM or Scheduling
 - Fitter Summary截圖
 - Timing Analyzer截圖
 - 遇到的問題與解決辦法
- 一組交一份，以pdf檔繳交
- 命名方式：teamXX_lab3_report.pdf
 - Ex: team01_lab3_report.pdf
- 繳交期限：demo當天午夜
 - 遲交每三天*0.7

Questions?