

# Project1

## Image Display Controller

### 1. 問題描述

請完成一影像顯示控制(Image Display Control)電路設計。此控制電路，可依指定之操控指令，使顯示端的影像進行影像放大(Zoom in)、縮小(Zoom out)與水平及垂直方向的平移(Shift)功能。

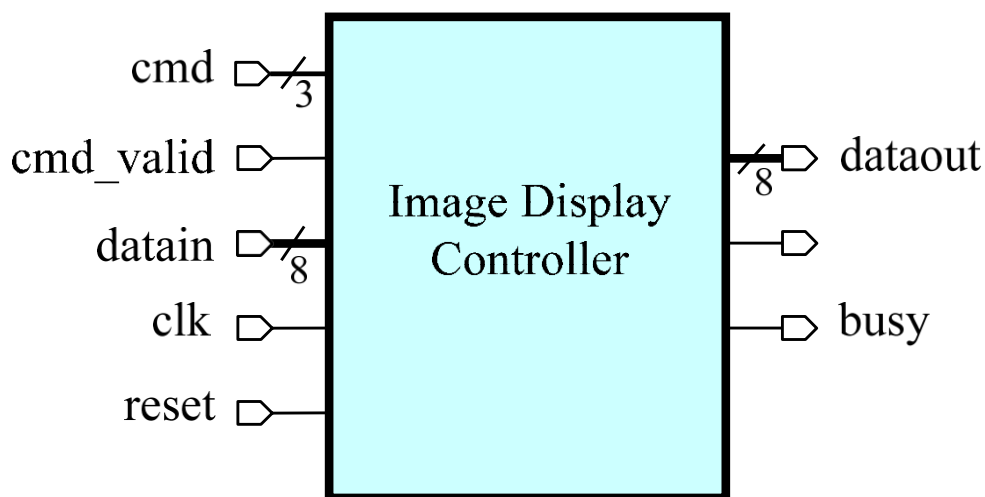
本控制電路有 5 只信號輸入(cmd, cmd\_valid, datain, clk, reset)及 3 只信號輸出(dataout, output\_valid, busy)，關於各輸入輸出信號的功能說明，請參考表一。

每組隊伍必須根據第 2 節所給的設計規格完成設計。同學可藉由課程助教所提供的輸入指令及正確結果檔來檢查設計是否有達到要求，詳情請參考附錄 B。

本次專案的設計時間為 10 月 24 日 星期一 上午 9:00 到 11 月 4 日 星期五 下午 05:00。各組同學請務必於 11 月 4 日 星期五 下午 06:00 前，將檔案上傳至各組資料夾內，詳情請參考附錄 C、D。課程助教會根據第 3 節中的評分標準進行評分。

### 2. 設計規格

#### 2.1 系統方塊圖



圖一、系統方塊圖

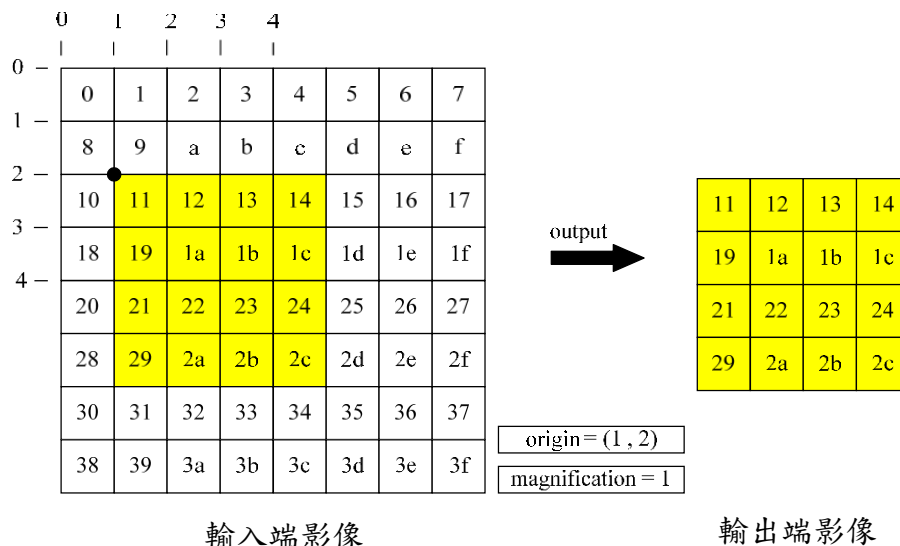
## 2.2 輸入/輸出介面

表一、輸入/輸出訊號

信號名稱	輸/出入	位元寬度	說明
<i>reset</i>	input	1	高位準非同步(active high asynchronous)之系統重置信號。 說明:本信號應於系統啟動時送出。
<i>clk</i>	input	1	時脈信號。 說明:此系統為同步於時脈正緣(posedge)之同步設計。
<i>cmd</i>	input	3	指令輸入信號。 說明:本控制器共有八種指令輸入，相關指令說明請參考表二。指令輸入只有在 <i>cmd_valid</i> 為 high 及 <i>busy</i> 為 low 時，為有效指令。
<i>cmd_valid</i>	input	1	有效指令輸入信號。 說明:當本信號為 high 時表示 <i>cmd</i> 指令為有效指令輸入。
<i>datain</i>	input	8	八位元影像資料輸入埠。
<i>dataout</i>	output	8	八位元影像資料輸出埠。
<i>output_valid</i>	output	1	有效資料輸出信號。 說明:當本信號為 high 時表示 <i>dataout</i> 為有效資料輸出。
<i>busy</i>	output	1	系統忙碌信號。 說明:當本信號為 high 時，表示此控制器正在執行現行(current)指令，而無法接收其他新的指令輸入。

## 2.3 系統功能描述

影像顯示控制器之輸入端，為一張 8x8 大小的影像資料。而輸出端則為 4x4 大小的顯示影像，如圖二所示。影像顯示控制器必須處理使用者輸入之指令，取得顯示相關之座標(origin)及倍率(magnification)參數，使顯示端達到放大、縮小及平移功能。



圖二、系統功能圖

### 2.3.1 輸入與輸出端之影像及參數規範

[影像輸入]

課程助教將提供輸入端影像資料，此影像資料為 8x8 共 64 筆測試樣本，每筆樣本為 8 位元資料。並且依由左而右；由上而下，且以序列(Serial)的方式循序輸入至同學的影像控制電路中。(如圖三所示資料，輸入順序為 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f, 10, ……., 3d, 3e, 3f)  
註:以下僅為圖例示範，詳細的輸入影像內容值未必如下圖三所示。

	0	1	2	3	4			
0	0	1	2	3	4	5	6	7
1	8	9	a	b	c	d	e	f
2	10	11	12	13	14	15	16	17
3	18	19	1a	1b	1c	1d	1e	1f
4	20	21	22	23	24	25	26	27
	28	29	2a	2b	2c	2d	2e	2f
	30	31	32	33	34	35	36	37
	38	39	3a	3b	3c	3d	3e	3f

圖三、輸入端影像資料

### [影像輸出]

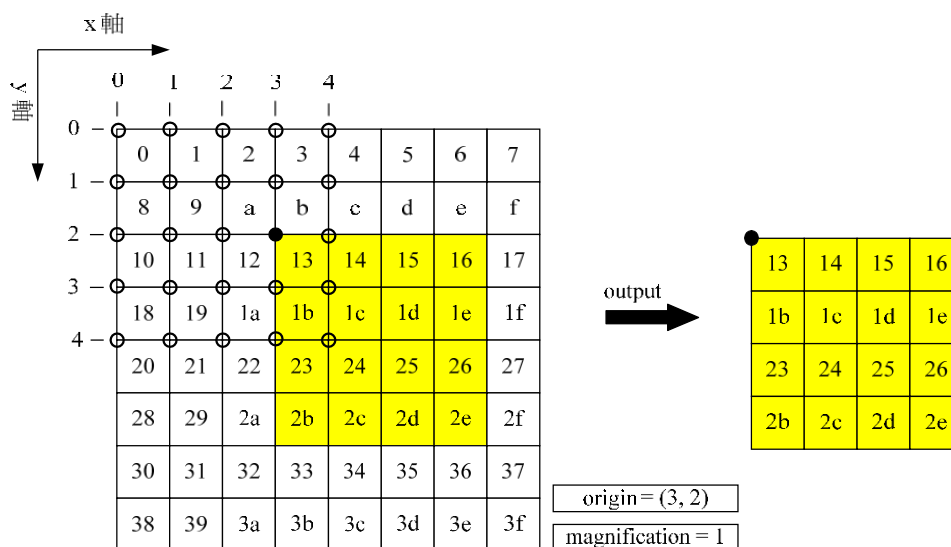
輸出端影像為 4x4 共 16 筆樣本輸出，每筆樣本為 8 位元資料。並且依由左而右；由上而下，且以序列 (Serial) 的方式循序輸出結果。(如圖四所示資料，輸出順序為 12,13,14,15,1a,1b,1c,1d,22, 23,24,25,2a,2b,2c,2d)。註:以下僅為圖例示範，詳細的輸出影像內容值未必如下圖四所示。

12	13	14	15
1a	1b	1c	1d
22	23	24	25
2a	2b	2c	2d

圖四、輸出端影像資料

### [原點座標]

原點座標(origin)指的是輸出影像相對於輸入影像的參考座標，本專案已定義輸入端影像之座標軸。輸入端影像之水平方向為 X 軸，垂直方向為 Y 軸，而原點座標表示輸出影像之左上端對應輸入影像的位置。此外，**X 軸與 Y 軸座標範圍為 0~4**。(如圖五所示，為確保輸出影像不超出輸入影像邊界，因此限制原點之 X 軸與 Y 軸範圍最大至 4)。同學需根據此參考座標，進行顯示端的畫面平移(Shift)功能設計。



圖五、輸入端影像參考座標

### [倍率參數]

本專案定義，當倍率參數(magnification)為 0，將執行縮小(Zoom Out)功能(如圖七)。反之，當倍率參數為 1，將執行放大(Zoom Out)功能(如圖六)。同學需根據指令輸入來修改倍率參數，並完成影像控制器之放大及縮小功能設計。

## 2.3.2 影像顯示控制器功能規範

### [指令定義]

影像控制器電路有 8 項控制指令功能。輸入指令(cmd)所對應之功能如表二所示。

表二、控制指令定義

cmd 編號	控制指令說明
0	Reflash
1	Load Data
2	Zoom In
3	Zoom Out
4	Shift Right
5	Shift Left
6	Shift Up
7	Shift Down

對於所有有效之控制指令，不論其是否會更變原點或倍率等顯示參數，皆必須在指令處理結束後輸出 16 筆顯示端影像資料。以下分別就各項指令進行說明。

#### ◆ 顯示端影像更新(Reflash)

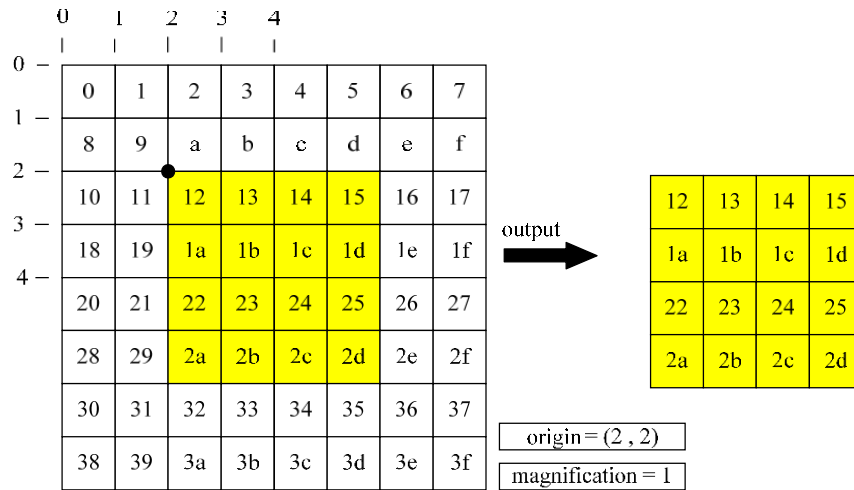
- ⇒ 更新顯示端的輸出畫面
- ⇒ Reflash 指令不作任何處理，僅將現行顯示之影像資料重新輸出。

#### ◆ 資料載入(Load Data)

- ⇒ 將 64 筆影像資料依序載入於 8x8 影像緩衝器中。
- ⇒ Load Data 時，系統將輸入一張全新的 8x8 影像，且輸出影像必須自動設為 Zoom Out 模式，也就是說，Load Data 內定會：
  - 設定原點座標(origin)為(0,0)
  - 設定倍率(magnification)為 0
- ⇒ Load Data 必定為系統的第一個指令輸入。

#### ◆ 畫面放大(Zoom In)

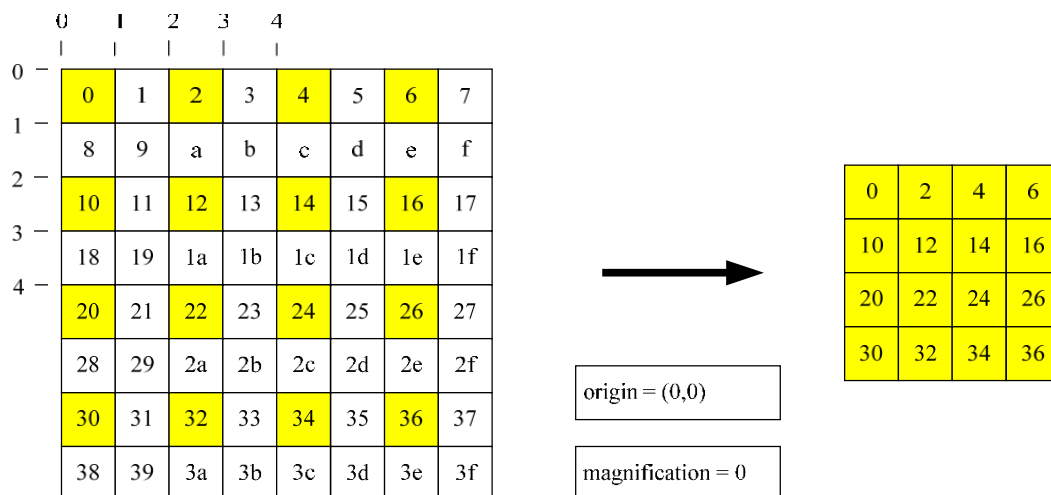
- ⇒ Zoom In 模式，可以 1:1 的顯示比例來顯示原輸入影像的部分區塊，且原點座標自動指向(2,2)，亦即自動顯示原輸入影像的中央區塊。
- ⇒ 進入 Zoom In 模式後，即可使用移動指令(Shift)移動顯示區域。  
Zoom In 指令執行之動作：
  - 設定倍率(magnification)參數為 1
  - 設定原點座標為(2,2)



圖六、畫面放大(Zoom In)之取樣相對位置

#### ◆ 畫面縮小(Zoom Out)

⇒ Zoom out 模式，是將 8x8 影像，並依圖七之取樣方式，顯示於4x4 顯示端。



圖七、畫面縮小(Zoom Out)之取樣相對位置

⇒ Zoom Out 指令執行之動作:

- 設定倍率(magnification)參數為 0
- 設定原點座標(origin)為(0,0)

⇒ 當進入 Zoom Out 模式時，原點座標應限制為(0,0)，且無法做上下左右移動等功能。

若於 Zoom Out 模式時收到上下左右移動(Shift Up/Down/Left/Right)的指令時，其原點座標仍為(0,0)，顯示結果仍維持不變，僅重覆輸出相同之取樣結果。

◆ 畫面右移(Shift Right)

- ⇒ 右移顯示區塊。執行此Shift Right 指令，使原點的 X 座標增加 1，但 X 軸座標最大不可超過 4。
- ⇒ 當 X 座標等於 4 時，倘若再收到右移指令，則 X 座標仍將維持 4，僅重覆輸出和現行指令前相同之顯示結果。

◆ 畫面左移(Shift Left)

- ⇒ 左移顯示區塊。執行此Shift Left 指令，將使原點的 X 座標刪減 1，但 X 軸座標最小不可低於 0。
- ⇒ 當 X 座標等於 0 時，倘若再收到左移指令，則 X 座標將仍維持 0，僅重覆輸出和現行指令前相同之顯示結果。

◆ 畫面上移(Shift UP)

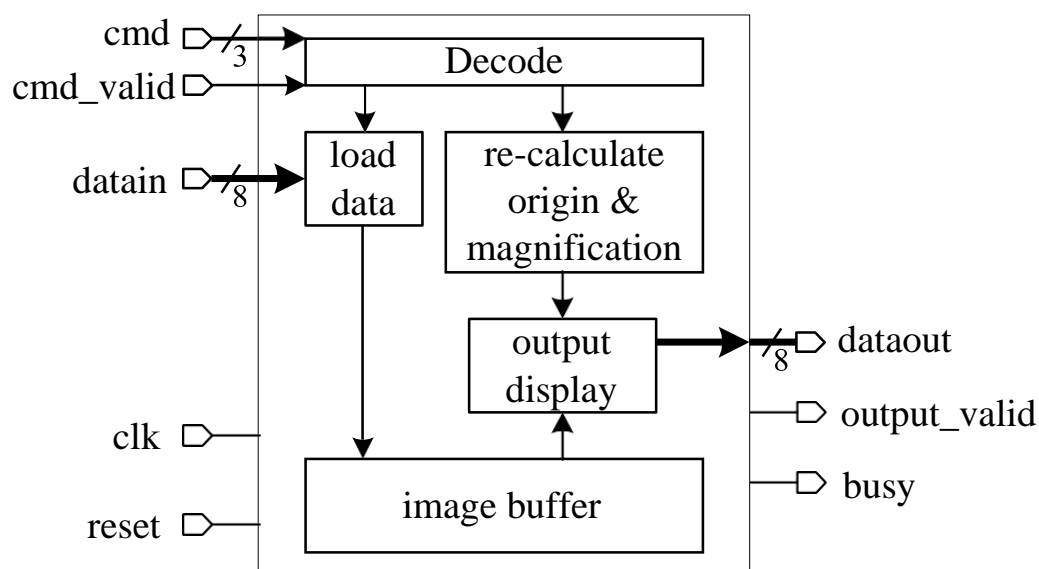
- ⇒ 上移顯示區塊。執行此Shift UP 模式，將使原點的 Y 軸座標減少 1，但 Y 軸座標最小不可低於 0。
- ⇒ 當 Y 軸座標等於 0 時，倘若再收到上移指令，則 Y 軸座標將仍為持 0，僅重覆輸出和現行指令前相同之顯示結果。

◆ 畫面下移(Shift Down)

- ⇒ 下移顯示區塊。執行此Shift Down 指令，將使原點的 Y 軸座標增加 1，但 Y 軸座標最大不可大於 4。
- ⇒ 當 Y 軸座標等於 4 時，倘若再次收到下移指令，則 Y 軸座標仍維持 4，僅重覆輸出和現行指令前相同之顯示結果。

### 2.3.3 影像顯示控制器範例

以下為影像控制器電路方塊圖範例，僅供同學作為設計之參考，同學們亦可自行發揮創意!

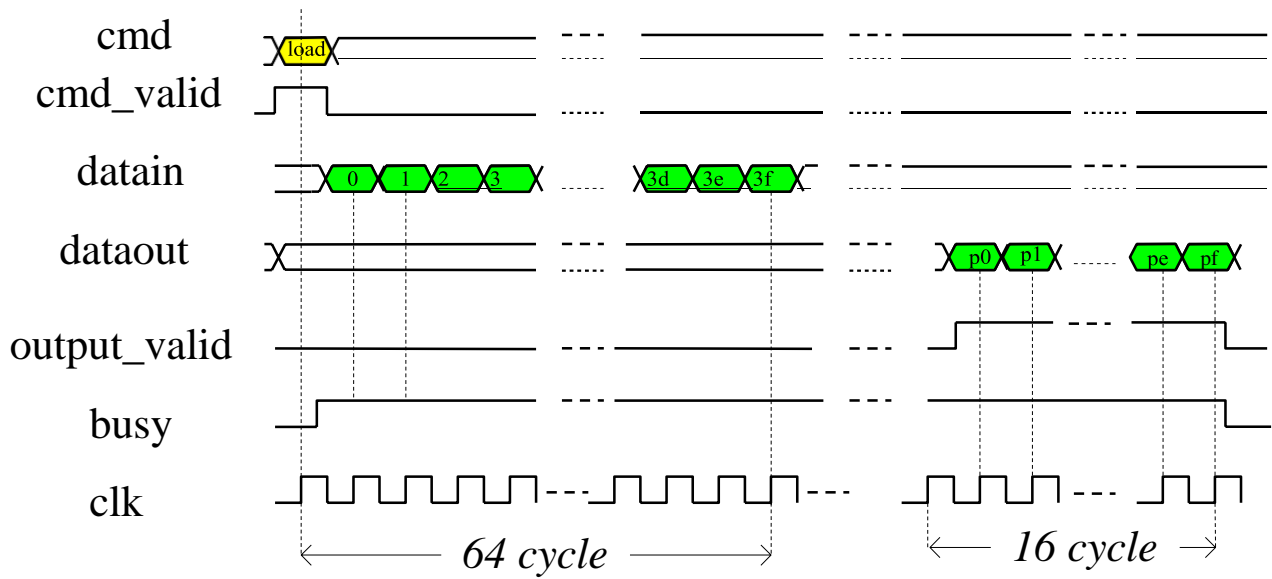


圖八、影像控制器電路方塊圖

## 2.4 時序規格圖

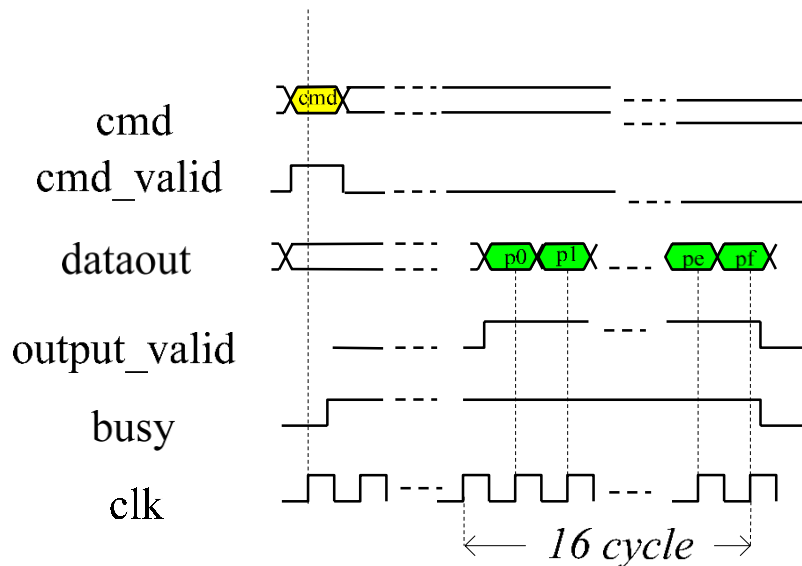
- ◆ 載入資料(Load Data)之時序規格圖，如下圖九所示。
  - ⇒ 在有效的 **Load Data** 指令之後，會緊接著連續輸入 64 筆影像資料
  - ⇒ 當影像顯示控制電路內部處理完成後，**連續輸出 16 筆顯示資料**，輸出同時須將 **output\_valid** 設為 **high**。
  - ⇒ 在整個處理過程中，**busy** 皆維持為 **high**。並在輸出完成後，將 **busy** 設回 **low** 以接受新指令輸入。





圖九、資料載入之時序規格圖

- ◆ 其它控制指令(reflash、zoom out、zoom in、shift right、shift left、shift up、shift down)之時序規格圖，如下圖十所示。
- ⇒ 當影像顯示控制電路內部處理完成，連續輸出 16 筆顯示資料，輸出同時須將 **output\_valid** 設為 high。
- ⇒ 在整個處理過程中，busy 皆維持為 high。並在輸出完成後，將 busy 設回 low 以接受新指令輸入。



圖十、其它控制指令之時序規格圖

### 3. 評分標準

課程助教將依照各組同學提供之系統時脈進行 RTL simulation 與 gate-level simulation，以驗證設計正確性。各組同學應於本專案定義的系統時脈下，確保輸出結果無設置與保持時間 (setup/hold time) 的問題，並完全符合課程助教所提供的標準設計結果為準。

表三為本次專案的評分標準，一旦設計經課程助教驗證後，且同學的設計結果正確 (post-synthesis) 將可得到基本分 80 分，而其次課程助教將依據 Performance Metric 的表現，來給予而外的分數，而 Performance Metric 的計算方式如下所示，其數值越低表示設計的電路效能越好。

$$\bullet \quad \text{Performance Metric} = \text{Time Complexity} \times \text{Area Complexity}$$

**Time Complexity:** 當你完成 gate level netlist 的 ncverilog 的驗證時，你可以從 “ncverilog.log” 的檔案中得到你的 Time Complexity 結果，如圖十一所示。這個時間代表整個電路完成工作所需的時間，與電路運作頻率 (operating frequency) 及電路效能 (operations per cycle) 有關。

**Area Complexity:** Area Complexity 可以從 Design Compiler 的 Area report 獲得，如圖十二所示。

表三、評分項目與得分

評分項目	得分
Functional Correction	80
Top 1 of Performance Metric	+20
The 2 <sup>nd</sup> place of Performance Metric	+15
The 3 <sup>rd</sup> place of Performance Metric	+10
The 4 <sup>th</sup> place of Performance Metric	+5

圖十一、Time complexity 的範例檔案

```
LCD_CTRLArea (~\Project1\DC\20M\Report) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
LCD_CTRLArea
3 Report : area
4 Design : LCD_CTRL
5 Version: H-2013.03-SP1
6 Date : Thu Oct 20 19:06:15 2016
7 *****
8
9 Library(s) Used:
10
11 fsa0m_a_generic_core_sslp62v125c (File: /Vol1/cad/CBDK018_UMC_Faraday_v1.0/CIC/SynopsysDC/db/fsa0m_a_generic_core_sslp62v125c.db)
12
13 Number of ports: 24
14 Number of nets: 9412
15 Number of cells: 9372
16 Number of combinational cells: 8702
17 Number of sequential cells: 669
18 Number of macros/black boxes: 0
19 Number of buf/inv: 2014
20 Number of references: 44
21
22 Combinational area: 210089.679615
23 Buf/Inv area: 23973.465345
24 Noncombinational area: 35697.715855
25 Macro/Black Box area: 0.000000
26 Net Interconnect area: undefined (Wire load has zero net area)
27
28 Total cell area: 245787.395469
29 Total area: undefined
30 1
```

圖十二、Area complexity 的範例檔案

## 附錄

### 附錄 A 設計檔 (For Verilog or VHDL)

1. 下表四為課程助教所提供各參賽者的設計檔，同學可於(File System/Project/Project 1)的位置找到這些檔案。

表四、設計檔

檔名	說明
testfixture.v	測試樣本檔。此測試樣本檔定義了時脈週期與測試樣本之輸入信號
lcd_ctrl.v	同學所使用的設計檔，已包含系統輸/出入埠之宣告
.synopsys_dc.setup	design compiler 初始設定範例檔案
Script.tcl	design compiler 合成指令的範例檔
cmd1.dat	第一組測試樣本之指令
image1.dat	第一組測試樣本使用之影像檔
out_golden1.dat	第一組測試樣本之正確結果
cmd2.dat	第二組測試樣本之指令
image2.dat	第二組測試樣本使用之影像檔
out_golden2.dat	第二組測試樣本之正確結果

2. 請使用 **lcd\_ctrl.v(vhd)**，進行影像顯示控制器之設計。其模組名稱、輸出/入埠宣告如下所示：

```
module LCD_CTRL(clk, reset, datain, cmd, cmd_valid, dataout, output_valid,
busy); input      clk;
input            reset;
input  [7:0]    datain;
input  [2:0]    cmd;
input
            cmd_valid; output
            [7:0]    dataout;
output
            output_valid;
output            busy;
endmodule
```

3. 本專案共提供二組測試樣本，請自行修改 **testfixture.v** 內容，以使用第二組測試樣本，修改方法如下：

```
`define IMAGE “./image1.dat”  
`define CMD “./cmd1.dat”  
`define EXPECT “./out_golden1.dat”
```

將以上三行修改為：

```
`define IMAGE “./image2.dat”  
`define CMD “./cmd2.dat”  
`define EXPECT “./out_golden2.dat”
```

4. 請每組同學使用課程助教所提供的測試樣本(testfixture.v)，來進行設計之模擬驗證。測試樣本(testfixture.v) 包含 *clk*, *reset*, *datain*, *cmd*, 與 *cmd\_valid* 測試信號。而於測試樣本內所定義的時脈週期參數(CYCLE)，可由各組同學依需求自行修改，若設計結果正確，testbench 將會產生 PASS 字樣，如圖十一所示。

## 附錄 B 測試樣本

本專案共提供兩組測試樣本，樣本位置放置於 Moodle 和 “140.118.121.247/Computer/File System/Project/Project 1” 中，為方便設計者除錯之用，將測試樣本之影像資料及指令輸入詳列如下：

### ◆ 測試樣本一

⇒ 相關檔案：image1.dat , cmd1.dat , out\_golden1.dat

⇒ 影像資料：

0	1	2	3	4	5	6	7
8	9	a	b	c	d	e	f
10	11	12	13	14	15	16	17
18	19	1a	1b	1c	1d	1e	1f
20	21	22	23	24	25	26	27
28	29	2a	2b	2c	2d	2e	2f
30	31	32	33	34	35	36	37
38	39	3a	3b	3c	3d	3e	3f

⇒ 指令輸入順序：

load, refresh, zoom in, shift right, shift down, shift right, shift down, shift down, shift right, refresh, shift left, shift left, shift left, shift left, shift left, shift up, shift up, shift up, shift up, shift up, refresh, shift right, shift right, shift right, shift right, shift right, load, zoom in, zoom out, zoom out, zoom in, zoom in, shift down, shift left, shift down, shift left, shift down, shift down, shift left, zoom out, shift up, shift down, shift left, shift right, load

◆ 測試樣本二

⇒ 相關檔案: image2.dat , cmd2.dat , out\_golden2.dat

⇒ 影像資料:

ff	36	e7	f0	55	32	75	42
18	20	57	30	eb	af	ec	11
61	49	93	22	67	a0	05	c5
28	44	62	66	cc	76	97	79
56	28	09	ff	40	18	80	33
e6	f0	e9	ea	87	dd	ed	95
78	d4	d3	bb	f4	77	52	c3
c4	aa	b5	92	98	ee	00	a9

⇒ 指令輸入順序

load, zoom out, load, shift up, shift down, shift left, shift right, zoom in, zoom in, shift left, shift down, shift down, shift left, shift left, refresh, zoom out, load, zoom in, shift right, shift right, shift up, shift up, shift right, shift up, shift down, shift down, shift down, shift down, shift down, shift right, shift right, zoom out, zoom out, zoom in, shift up, shift up, shift up, shift up, shift left, shift right, shift left, shift left, shift left, zoom out, refresh

## 附錄 C 評分用檔案

評分所須檔案可分為兩部份：(1) RTL design，即各團隊在該次專案所設計的 RTL code，若設計採模組化而有多個設計檔，請務必將合成所要用到的 module 檔放進來，以避免課程助教進行評分時，無法進行編譯；(2) gate-level design，即由合成軟體所產生的 gate-level netlist，以及對應的 SDF 檔和指定相關檔案。

表五、評分用檔案

<b><i>RTL category</i></b>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
RTL Simulation	*.v or *.vhd	Verilog (or VHDL) synthesizable RTL code
<b><i>Gate-Level category</i></b>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
Pre-layout Gate-level Simulation	*_syn.v	Verilog gate-level netlist generated by Synopsys Design Compiler
	*_syn.sdf	SDF timing information generated by Synopsys Design Compiler
	*.vcd	Wave information generated by Ncverilog
	ncverilog.log	The log file generated by Ncverilog
	*.area	Area information generated by Synopsys Design Compiler



## 附錄 D 檔案上傳

所有包含於如附錄 C 中表格所示的檔案，均需要提交至課程助教指定的資料夾內。步驟如下：

1. 參考附錄 C 評分用檔案，將所有繳交檔案複製到(140.118.121.247/Computer/File System/Project/Group\_X) 資料夾中，其中 X 表示組別，請務必確保檔案放入正確的資料夾內。

若有任何問題，請聯絡課程助教：

陳偉哲 s2017091@gmail.com；辦公室: IB714-1

曾昱豪 M10402146@mail.ntust.edu.tw 辦公室: IB714-1