

B 組-標準元件數位電路設計

初賽時間為 3 月 23 日(三) **8:30~20:30**，初賽當日請密切注意 IC 競賽網頁與 CIC 網頁，所有最新訊息將於網頁上公告。請參賽隊伍於**早上 10 點半前**完成下列步驟進行初賽登錄，主辦單位將依完成此步驟之隊伍數決定各組最後得獎名額，請務必完成登錄動作，以免影響您的權益。

※請參賽者將作品在**今天 20:30** 前上傳至 FTP。

請將您的隊伍參賽資料E-mail至 icdesign_b@yuntech.edu.tw

- 信件內容格式如下:
 - 參賽組別：B 組-標準元件數位電路設計
 - 參賽編號：(例：20001)
 - 參賽姓名：張三、李四
- 信件標題請標示為「IC 設計競賽初賽資料登錄」

※初賽之注意事項請參閱競賽手冊

2011 IC Design Contest Preliminary

標準元件數位電路設計

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平均(Average)、X軸及Y軸鏡像(Mirror)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, IROM_Q, clk, reset)及 7 只信號輸出(IROM_EN, IROM_A, IRB_RW, IRB_D, IRB_A, busy, done)，關於各輸入輸出信號的功能說明，請參考表 一。

每個參賽隊伍必須根據下一節所給的設計規格完成設計。參賽隊伍可藉由CIC 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IC 設計競賽比賽時間為上午08:30 到下午08:30。當IC 設計競賽結束後，CIC 會根據第三節中的評分標準進行評分。為了評分作業的方便，各參賽隊伍應參考附錄D 中所列的要求，附上評分所需要的檔案。

2. 設計規格

2.1 系統方塊圖

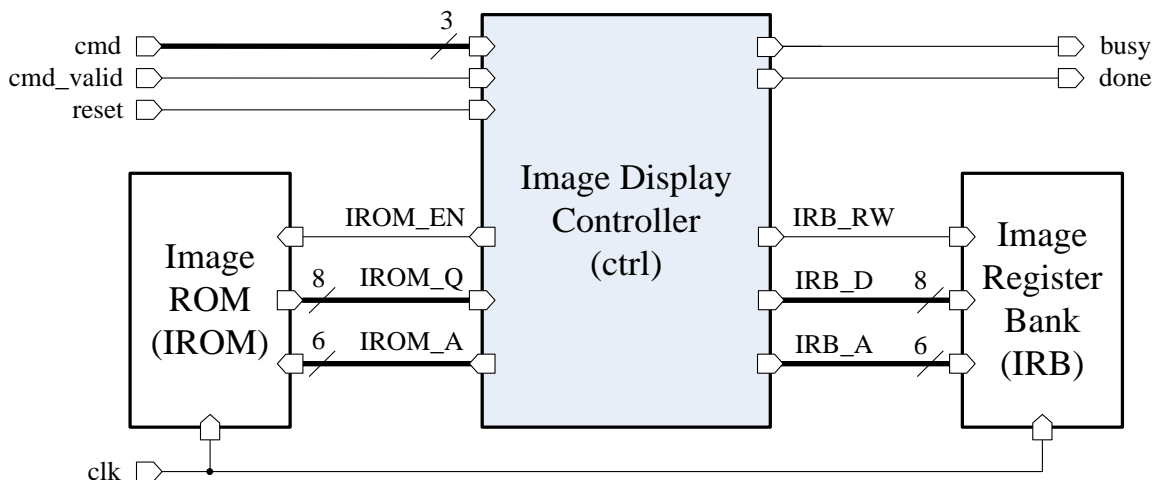


圖 一、系統方塊圖

2.2 輸出/輸入介面

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 cmd_valid 為 high 及 busy 為 low 時為有效指令
<i>cmd_valid</i>	input	1	讀寫控制訊號。 說明:當本信號為 high 時表示 cmd 指令為有效指令輸入。
<i>IROM_Q</i>	input	8	Image ROM 八位元資料輸出埠。
<i>IROM_A</i>	output	6	IROM 六位元位址信號。
<i>IROM_EN</i>	output	1	IROM 致能控制訊號。 說明:當本信號為 low 時，表示是將啟動 IROM 進行讀取；信號為 high 時，表示將關閉 IROM。
<i>busy</i>	output	1	系統忙碌訊號。 說明:當本信號為 high 時，表示此控制器正在執行現行指令(current)，而無法接受其他新的指令輸入；當本信號為 low 時，系統會開始輸入指令。reset 時，default 設定為 high。
<i>done</i>	output	1	當控制器完成寫入 IRB 時，將 done 設為 high 表示完成。
<i>IRB_A</i>	output	6	IRB 六位元位址信號。
<i>IRB_D</i>	output	8	IRB 八位元資料輸入埠。
<i>IRB_RW</i>	output	1	IRB 讀寫控制訊號。(本試題只有使用寫入功能) 說明:當本信號為 low 時，表示是將啟動 IRB 進行寫入

2.2.1 Image ROM(IROM)與Image Register Bank(IRB)規格描述

本試題使用到兩個記憶體模組，一為唯讀記憶體(ROM)格式另一為單埠Register file格式。Image ROM(IROM)模組的記憶體寬度為8位元，而記憶體深度為64個word。

Image Register Bank(IRB)模組的記憶體寬度為8位元，而記憶體深度也為64個word。如下圖二所示。

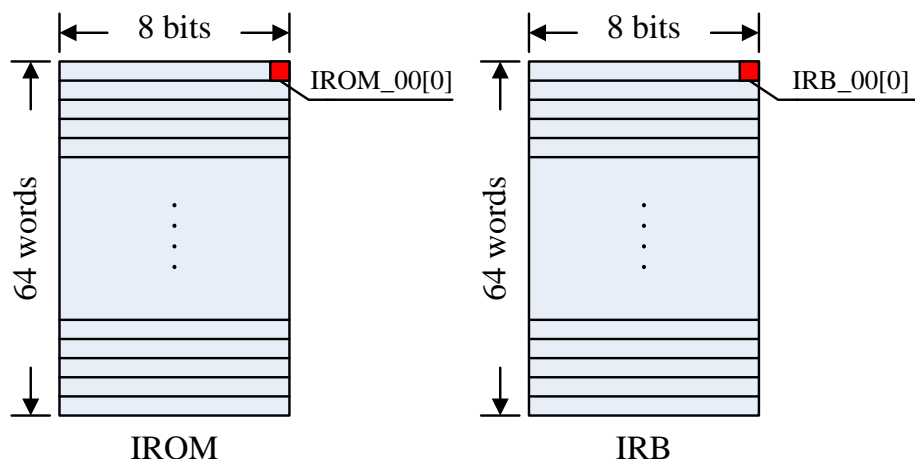


圖 二、系統功能圖

2.3 系統功能描述

當reset結束後，影像顯示控制器之輸入端從IROM讀取一張8x8大小的影像。資料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)及資料參數，使得顯示端達到平均、平移以及鏡像功能，並將經過指令處理完的影像資料寫入IRB，如圖三所示。

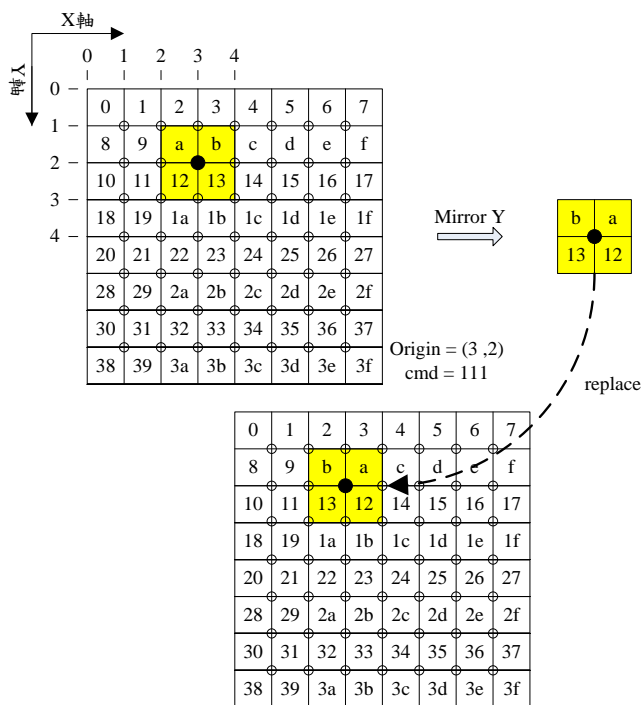


圖 三、系統功能圖

2.3.1 輸入與輸出端之影像及參數規範

[影像輸入]

主辦單位將提供輸入端影像資料存至IROM。此影像為8x8共64筆測試樣本，每筆樣本為8位元資料，並且依左而右；由上而下，參賽者必須由IROM讀取影像資料，並且依左而右；由上而下存至影像控制電路中。(如圖三所示資料，輸入的順序為 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，詳細輸入影像值未必如下圖 四所示。

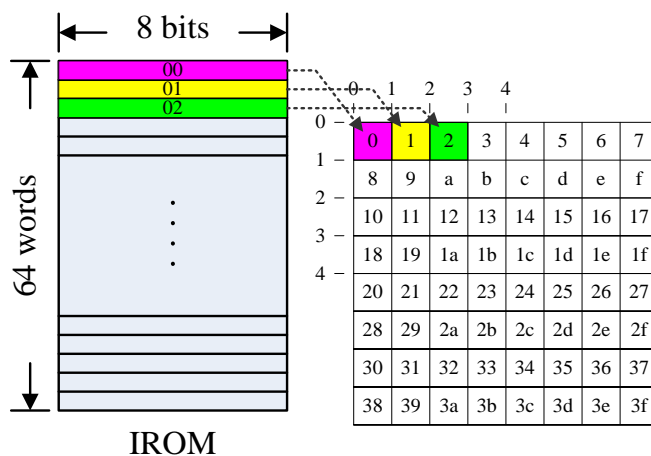


圖 四、輸入端影像

[影像輸出]

輸出端影像為8x8共64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，且已序列(Serial)的方式循序寫入IRB內。(如圖 五所示資料)。

註: 以下僅為圖例示範，詳細輸入影像值未必如下圖 五所示。

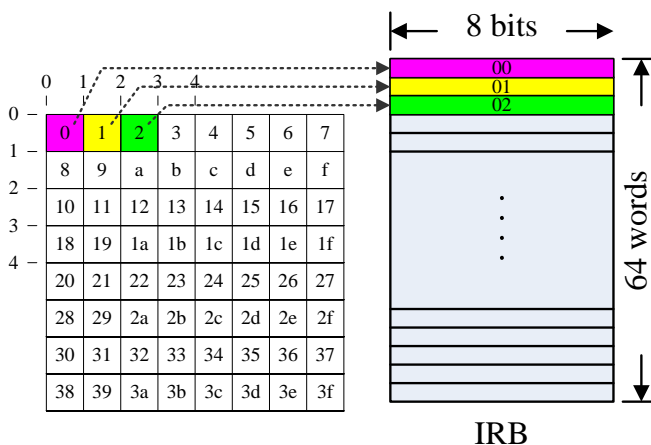


圖 五、輸出端影像資料

[操作點]

操作點(operation point)指的是在影像資料的座標點，在操作點的上下左右四格為操作影像資料，控制器將使用操作影像資料來運算。本試題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。此外，X軸與Y軸座標範圍為0~+8。(如圖六所示，為確保操作影像資料不超過對應輸入影像邊界，因此限制原點之X軸與Y軸範圍最大為+1~+7)。參賽者須根據此座標軸，進行顯示端的畫面，進行顯示端的畫面平移(Shift)功能設計。

註：本試題規定讀入控制器後影像資料初始操作點座標為(4, 4)，如下圖 六所示。

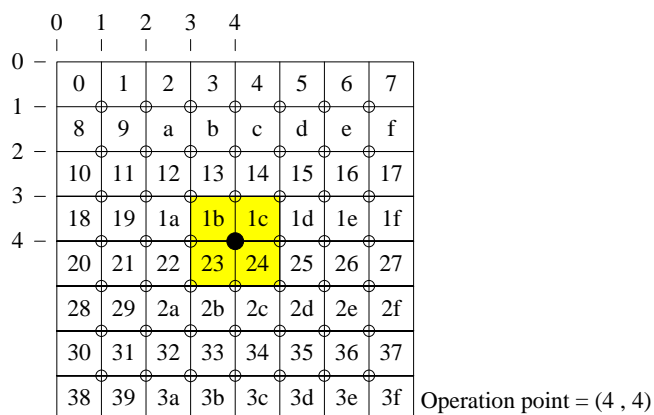


圖 六、輸入端影像操作點

2.3.2 影像顯示控制器功能規範

[指令定義]

影像控制器電路控制指令。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	Write
1	Shift Up
2	Shift Down
3	Shift Left
4	Shift Right
5	Average
6	Mirror X
7	Mirror Y

■ 寫入(Write)

- 當執行寫入(Write)指令時，控制器會依由左而右；由上而下將影像資料寫入IRB。

■ 畫面上移(Shift Up)

- 上移顯示區塊。執行此Shift Up模式，將使操作點的Y減少1，但Y軸座標最小不可低於1。
- 當Y座標等於1時，倘若再收到上移指令，則Y軸座標將仍維持為1，操作點維持不變。

■ 畫面下移(Shift Down)

- 下移顯示區塊。執行此Shift Down模式，將使操作點的Y軸增加1，但Y軸座標最大不可大於7。
- 當Y座標等於7時，倘若再收到下移指令，則Y軸座標將仍維持為7，操作點維持不變。

■ 畫面左移(Shift Left)

- 左移顯示區塊。執行此Shift Left模式，將使操作點的X軸刪減1，但X軸座標最小不可低於1。
- 當X座標等於1時，倘若再收到左移指令，則X軸座標將仍維持為1，操作點維持不變。

■ 畫面右移(Shift Right)

- 右移顯示區塊。執行此Shift Right模式，將使操作點的X軸增加1，但X軸座標最小不可大於7。
- 當X座標等於7時，倘若再收到右移指令，則X軸座標將仍維持為7，操作點維持不變。

■ 影像資料平均(Average)

- 本試題定義，當執行平均(Average)指令時，將執行目前操作點座標之影像資料取近似平均數之計算，即將目前座標所對應的4筆影像資料相加之後再除以4，當有小數點時則以無條件捨去法處理(例如 $(a+b+12+13)/4=14.5$ ，即輸出14)，輸出影像資料4筆皆輸出計算後之近似平均數，並改變原始影像資料。如圖 七所示。

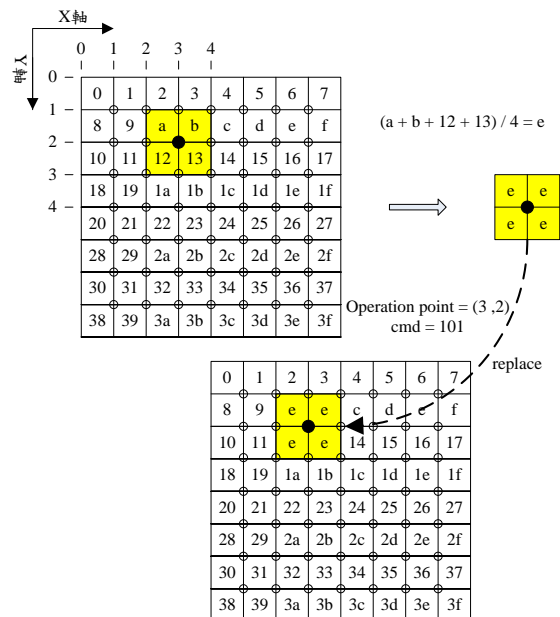
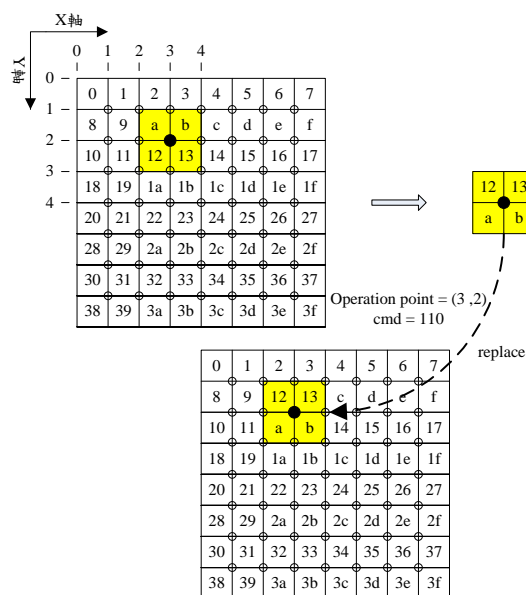


圖 七、取近似平均數影像輸出

■ 影像資料X軸鏡像(Mirror X)

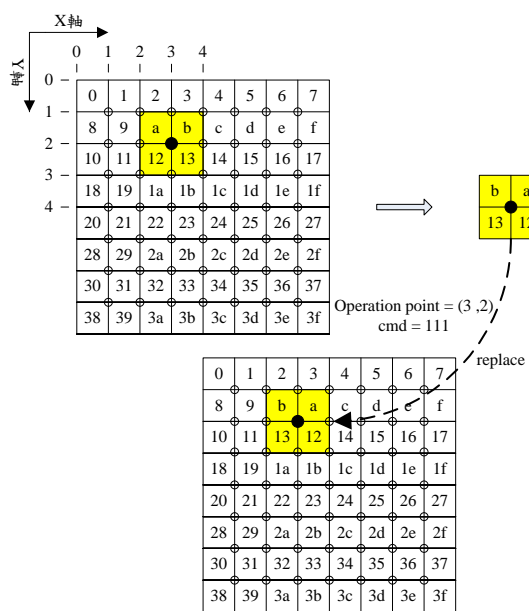
- 本試題定義，當執行X軸鏡像(Mirror X)指令時，將輸出目前操作點座標之影像資料皆以目前座標對X軸翻轉，並改變原始影像資料。如圖八所示。



圖八、X 軸鏡像資料影像輸出

■ 影像資料Y軸鏡像(Mirror Y)

- 本試題定義，當執行Y軸鏡像(Mirror Y)指令時，將輸出目前操作點座標之影像資料皆以目前座標對Y軸翻轉，並改變原始影像資料。如圖九所示。

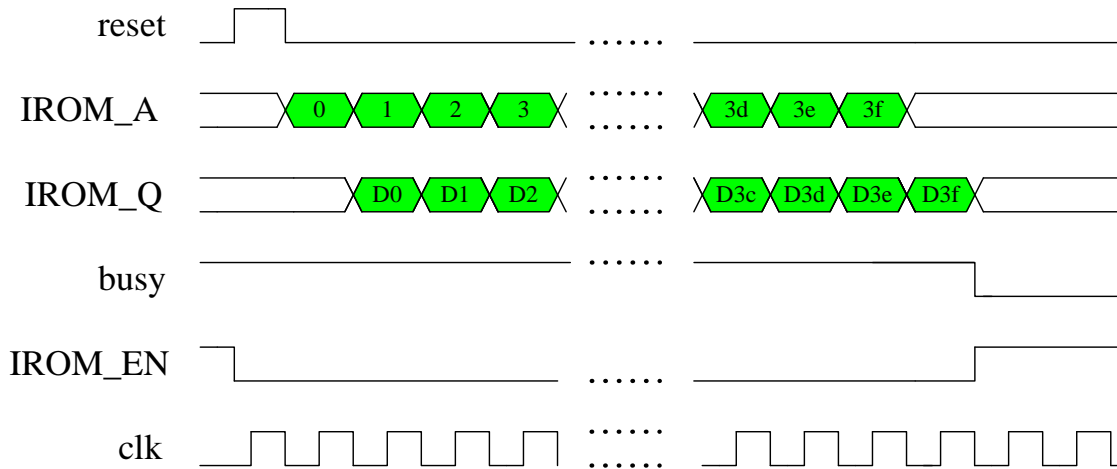


圖九、Y 軸映像資料影像輸出

2.4 時序規格圖

■ 重置(Reset)後之時序規格圖，如下圖所示。

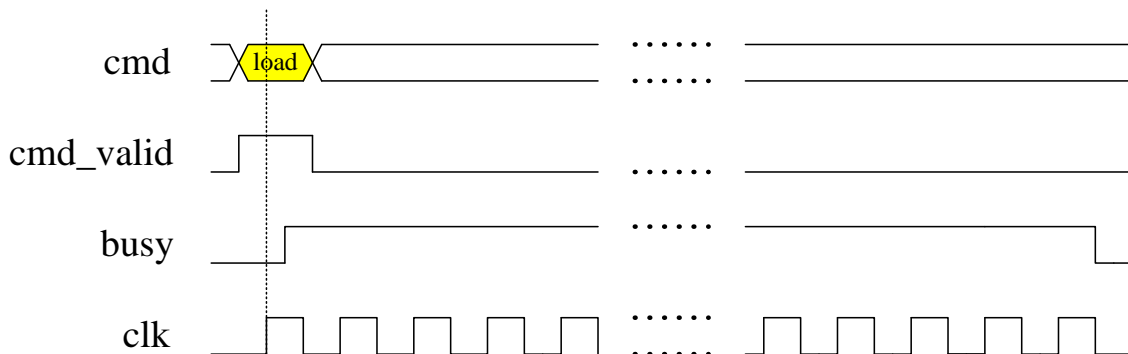
- 在電路重置(Reset)之後，控制器將會由IROM讀取64筆影像資料。
- 當IROM_EN為low時，表示啟動IROM，即可輸入位址信號讀取IROM內的影像資料。
- 在整個處理過程中，busy皆維持為high。並在讀取完成後，將busy設回low以接受新指令輸入。



圖十、資料讀取之時序規格圖

■ 其他控制指令(average、shift up、shift down、shift left、shift right)之時序規格圖，如下圖十一所示。

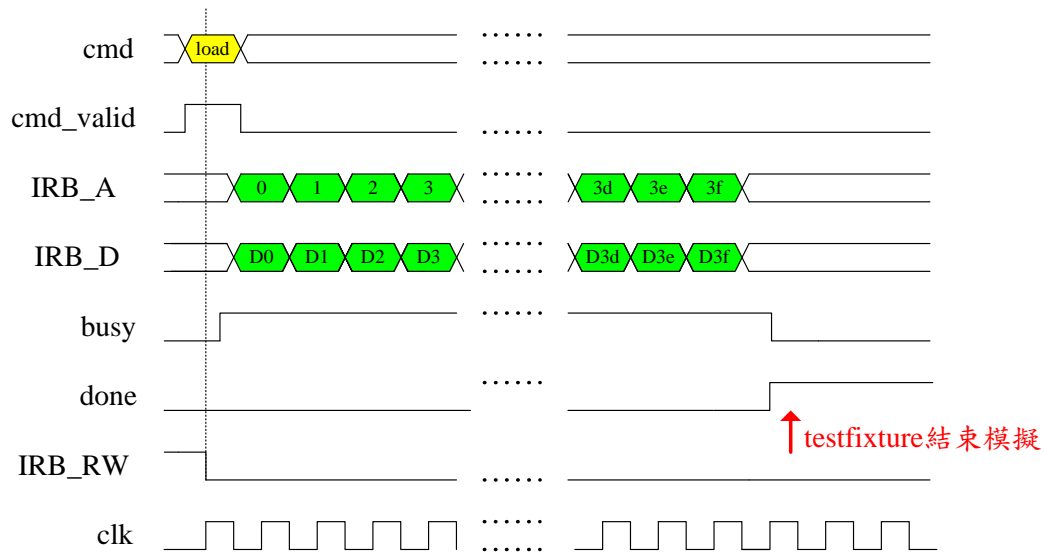
- 在整個處理過程中，busy皆維持為high。並在輸出完成後，將busy設回low以接受新指令輸入。



圖十一、其他控制指令之時序規格圖

■ 寫入指令(write)之時序規格圖，如下圖十二所示。

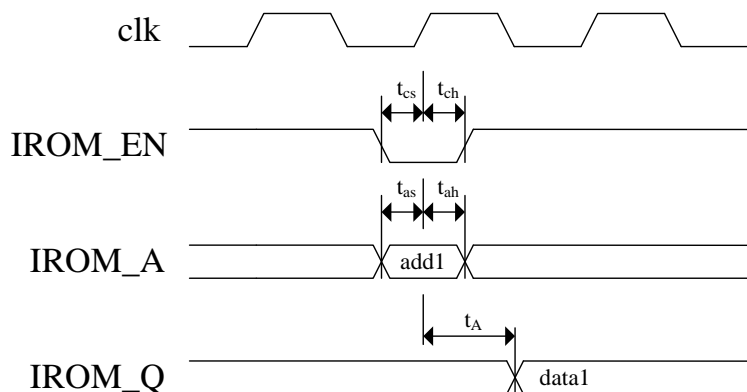
- 執行寫入指令時控制器會將處理完的影像資料寫入IRB。
- 當IRB_RW為low時，表示對IRB寫入，即可輸入位址信號將影像資料寫入IRB。
- 在整個處理過程中，busy皆維持為high。並在輸出完成後，將busy設回low以接受新指令輸入。
- 寫入完成後，並將done信號設為high，表示寫入完成，此時testfixture會拿寫入IRB的資料與golden pattern比對。



圖十二、寫入指令之時序規格圖

2.5 IROM與IRB之時序規格

IROM 讀取動作主要是以 IROM_EN 啟動 IROM 後，輸入 address 讀取資料，其資料讀取之波形時序圖，如下圖十三所示。**特別注意若沒有要進行讀取時，請將 IROM_EN 保持為 high。**最後，時序數值整理於表三。

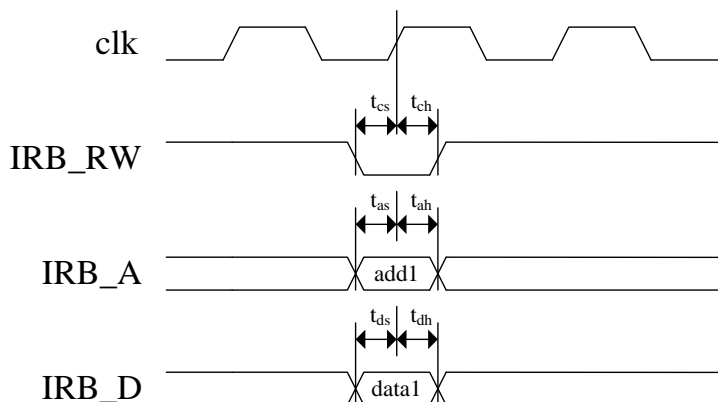


圖十三、IROM 讀取之時序規格圖

表三、IROM 時序參數表

Symbol	Description	Min	Unit
t_A	Access time	1.16	ns
t_{cs}	Chip Enable setup time	0.34	ns
t_{ch}	Chip Enable hold time	0	ns
t_{as}	Address setup time	0.29	ns
t_{ah}	Address hold time	0	ns

IRB 讀取動作主要是以 IRB_RW 這個信號來控制，其資料讀取之波形時序圖，如下圖十四所示。**特別注意若沒有要進行寫入時，請將 IRB_RW 保持為 low。**最後，時序數值整理於表四。



圖十四、寫入指令之時序規格圖

表四、IRB 時序參數表

Symbol	Description	Min	Unit
t_{ws}	Write Enable setup time	0.24	ns
t_{wh}	Write Enable hold time	0	ns
t_{as}	Address setup time	0.19	ns
t_{ah}	Address hold time	0.05	ns
t_{ds}	Data setup time	0.11	ns
t_{dh}	Data hold time	0	ns

3. 評分標準

主辦單位的評分人員將依照參賽者提供之系統時脈進行RTL simulation 與gate-level simulation，以驗證設計正確性，並且依據設計檔上傳至CIC FTP 檔案伺服器(請參閱附錄D)的時間來進行排名。各參賽隊伍應於參賽者定義的系統時脈下，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CIC 所提供的標準設計結果為準。

CIC 將本試題區分為下面三個等級來作為功能完成度之評分，完成度越高者優先錄取。若為同一等級則以檔案上傳時間來評分，且對每個等級分別進行RTL與gate-level simulation：

1. A等級：完成測試樣本一至二 (tb1 & tb2)之RTL與gate-level simulation
2. B等級：完成測試樣本一至二 (tb1 & tb2)之RTL simulation
3. C等級：完成測試樣本一 (tb1)的上傳功能 之RTL simulation

B等級至C 等級雖不須進行synthesis，但RTL code須為synthesizable RTL code
--

請注意，我們將以各參賽隊伍的設計結果正確為前提，並以**最後上傳檔案的時間**為依據。一旦設計經評審驗證後，**完成同一等級者，上傳時間越早，其所得到的分數就越高。建議每完成一個等級就先將設計檔案內容上傳**，主辦單位將根據設計內容的完成度給予記分。審查成績將另擇期通知。

附錄

在附錄A中說明本次競賽之軟體環境；附錄B為主辦單位所提供各參賽者的設計檔說明；附錄C 為測試樣本，亦即參賽者必須回傳至CIC 的檔案資料；附錄D 為上傳檔案說明，亦即參賽者必須回傳至CIC 的檔案資料；附錄E則為設計檔上傳步驟說明。

附錄A 軟體環境

競賽中所提供的設計軟體環境與版本如下表五。驗證評分時，係以所列軟體及版本作為驗證依據。

表五、設計軟體版本

Functionality	Corresponding EDA tools
Logic Simulator	nc-verilog v8.2 modelsim v6.3a vcs v2010.06-sp1
Logic Synthesizer	design-compiler v2010.03-sp5

注意！ 評分時將以nc-verilog v8.2版的結果為主。

附錄B 設計檔案說明

1. 以下表六為主辦單位所提供各參賽者的設計檔案

表六、設計檔

檔名	說明
testfixture .v	測試樣本檔(testbench)。此測試樣本檔定義了時脈週期與測試樣本之輸入信號，module 名稱為 test
lcd_ctrl.v (lcd_ctrl.vhd)	參賽者影像顯示控制器 lcd_ctrl 的設計檔範本，已包含系統輸/出入埠之宣告 注意!!!設計者繳交之檔案，包含檔名、top module name、port name 皆不能更改，在評分時因為檔名或是 top module name 及 port name 不同之問題而無法模擬，將視為設計錯誤
IROM.v	ROM 模擬檔 (simulation model)
IRB.v	Register Bank 模擬檔 (simulation model)
cmd1.dat	第一組測試樣本之指令
cmd2.dat	第二組測試樣本之指令
image1.dat	第一組測試樣本輸入檔案
tb1_goal.dat	第二組測試樣本之 IRB 比對檔
image2.dat	第一組測試樣本輸入檔案
tb2_goal.dat	第二組測試樣本之 IRB 比對檔
report_000.txt	結果報告範本

2. 請使用 *lcd_ctrl.v(vhd)*，進行影像顯示控制電路之設計。其模組名稱、輸出/入埠宣告如下列範本所示：

```
// Verilog user: lcd_ctrl.v
module lcd_ctrl(clk, reset, cmd, cmd_valid, IROM_Q, IROM_A, IROM_EN, busy,
done, IRB_A, IRB_D, IRB_RW);
    input      clk, reset;
    input  [2:0] cmd;
    input      cmd_valid;
    input  [7:0] IROM_Q;
    output [5:0] IROM_A;
    output      IROM_EN, busy, done;
    output [5:0] IRB_A;
    output [7:0] IRB_D;
    output      IRB_RW;
endmodule
```



```
-- VHDL user: lcd_ctrl.vhd
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity lcd_ctrl is Port (
    clk      : in    STD_LOGIC;
    rst      : in    STD_LOGIC;
    cmd      : in    STD_LOGIC_VECTOR (2 downto 0);
    cmd_valid : in    STD_LOGIC;
    IROM_Q   : in    STD_LOGIC_VECTOR (7 downto 0);
    IROM_A   : out   STD_LOGIC_VECTOR (5 downto 0);
    IROM_EN  : out   STD_LOGIC;
    busy     : out   STD_LOGIC;
    done     : out   STD_LOGIC;
    IRB_A    : out   STD_LOGIC_VECTOR (5 downto 0);
    IRB_D    : out   STD_LOGIC_VECTOR (7 downto 0);
    IRB_RW   : out   STD_LOGIC);
end lcd_ctrl;

architecture lcd_ctrl_arc of ctrl is

begin

end lcd_ctrl_arc;
```

3. 比賽共提供兩組測試樣本，**參賽者可依下面範例來進行模擬：**

- ncverilog 指令範例如下：
ncverilog testfixture.v lcd_ctrl.v +define+tb1
- 若使用modelsim，則是在compiler verilog時，使用下面指令：
vlog testfixture.v lcd_ctrl.v +define+tb1
- 若使用vcs，則是在compiler verilog時，使用下面指令：
vcs -R +v2k testfixture.v lcd_ctrl.v +define+tb1
- 上述指令中***+define+tb1*** 指的是使用第一組測試樣本模擬，若須使用其它測試樣本請自行修改此參數。以第二組測試樣本為例：***+define+tb2***。
- 關於模擬時使用的一些記憶體，因已經以include方式加在testfixture.v 裏，所以不需加在模擬指令裏。

- IROM及IRB simulation model 包含timing constrain，在RTL模擬時，為符合timing constrain的要求，建議address及data要加上delay，以符合setup time的要求。
- 若RTL模擬時，要避免時序檢查以減少錯誤訊息，可於模擬指令中加入 *+notimingchecks*
範例如：*ncverilog testfixture.v lcd_ctrl.v +define+tb1 +notimingchecks*

vlog testfixture.v lcd_ctrl.v +define+tb1 +notimingchecks
vcs -R +v2k testfixture.v lcd_ctrl.v +define+tb1 +notimingchecks

4. dump 波形檔請參考下列指令：

- ncverilog 指令範例如下：(請先source Verdi 的環境設定檔)
ncverilog testfixture. lcd_ctrl.v +define+tb1 +FSDB +access+r
- modelsim使用者，請直接使用內建波形來除錯
vcs -R +v2k testfixture.v lcd_ctrl.v +define+tb1 +FSDB
-P <Your_Verdi_Path>/Verdi.tab <Your_Verdi_Path>/pli.a

5. 請參賽隊伍使用主辦單位所提供的測試樣本(testfixture.v)，來進行設計之模擬驗證。主辦單位除了將利用本試題所提供的測試樣本進行各參賽隊伍之設計測試外，我們亦準備另一份測試樣本，進行設計之二次驗證。唯通過二次驗證且均能符合輸出結果，始能稱為無誤之設計。

附錄C 測試樣本

比賽共提供兩組測試樣本，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(tb1)

- 相關資料：image1.dat, cmd1.dat, tb1_goal.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：

shift up, average, shift left, mirror x, shift down, shift right, mirror y ,write

● 測試樣本二(tb2)

■ 相關資料：image2.dat, cmd2.dat, tb2_goal.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：

shift up ,shift up ,mirror y ,shift up ,mirror x ,shift up ,shift left ,shift down ,shift left ,average ,shift left
,mirror y ,shift down ,shift down ,mirror x ,shift right ,average ,shift down ,mirror y ,shift down ,mirror x
,shift right ,average ,shift right ,mirror y ,shift right ,mirror x ,shift down ,average ,shift right ,mirror
x ,shift right ,mirror y ,shift up ,average ,shift up ,shift up ,shift left ,mirror x ,shift up ,average ,mirror y
,shift right ,mirror y ,write

附錄D 評分用檔案

評分所需檔案可分為三部份：(1)RTL design，即各參賽隊伍對該次競賽設計的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各module檔放進來，以免評審進行評分時，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；(3)report file，參賽隊伍必須依照自己的設計內容，撰寫report.000 檔，以方便主辦單位進行評分，report.000 的格式如圖十所示。(report 檔以後三碼序號表示版本，若繳交檔案更新版本，則新版的report 檔檔名為report.001，依此類推)

注意!!!設計者繳交之檔案請依照表七所列名稱。

表七、設計檔

<i>RTL category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
N/A	report.xxx	design report
RTL Simulation	lcd_ctrl.v or lcd_ctrl.vhd	Verilog (VHDL) synthesizable RTL code
<i>Gate-Level category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
Pre-layout Gate-level Simrlation	lcd_ctrl_syn.v	Verilog gate-level netlist generated by Synopsys Design Compiler
	lcd_ctrl_syn.sdf	SDF timing information generated by Synopsys Design Compiler
	lcd_ctrl_syn.ddc	Design database generated by Synopsys Design Compiler

ftp帳號: 999999

完成模擬之測試樣本等級(level of test pattern) : A

--- RTL category---

使用之HDL模擬器名稱(HDL simulator): nc-verilog

RTL檔案名稱(RTL filename): lcd_ctrl.v, 以及使用到的子模組檔案...

--- Pre-layout gate-level ---

gate-level 檔案名稱(gate_level filename): lcd_ctrl_syn.v

gate-level sdf filename: lcd_ctrl_syn.sdf

design compiler合成資料庫(dc library): lcd_ctrl_syn.ddc

-----注意事項(annotation)-----

(其餘注意事項依各參賽隊伍的需求填寫)

圖十五、report.000 的範本

附錄E 檔案上傳

所有包含於如附錄D中表格所示的檔案，均需要提交至CIC。並且，提交的設計檔案，需要經過壓縮於同一個資料夾下，步驟如下：

1. 建立一個result_xxx 資料夾。其中“xxx”表示繳交版本。例如“000”表示為第一次上傳；“001”表示為第二度上傳；002 表示為第三度上傳，以此類推...。
2. 參考附錄D 評分用檔案，將所有繳交檔案複製到result_xxx 資料夾
3. 執行tar 指令將result_xxx 資料夾包裝起來，tar 的指令範例如下：
tar cvf result_xxx.tar result_xxx
其中xxx 表示繳交版本
執行完後應該會得到result_xxx.tar 的檔案
4. 使用ftp 將result_xxx.tar 及report.xxx 一併上傳至CIC 提供的ftp server，result_xxx.tar與report.xxx 之“xxx”編號需一致，評審將以**最後上傳的設計檔及報告檔編號進行評分作業，由於比賽以最後上傳檔案時間作為名次排序標準，若原上傳版本已經正確，請勿再次上傳新版本。**

請注意!!上傳之FTP 需切換為二進制模式(binary mode)，且傳輸埠均設為21(port:21)。

ftp 的帳號和密碼在賽前已用email 寄給各參賽者。若有任何問題，請聯絡CIC

FTP site1 (台灣大學)：iccftp.ee.ntu.edu.tw (140.112.20.92)

FTP site2 (新竹晶片中心)：iccftp.cic.org.tw (140.126.24.18)

FTP site3 (南區晶片中心)：iccftp2.cic.org.tw(140.110.117.9)

5. 若你需要繳交更新版本，請重覆以上步驟，並記得修改report 檔及tar 檔的版本編號，因為你無法修改或刪除或覆蓋之前上傳的資料。
6. 為確保ftp網路傳送之正確性，ftp site提供MD5檔案檢查機制，請自行檢查上傳檔之MD5校驗碼，MD5使用方法請參閱IC競賽參賽手冊附件一。