

2018 IC Design Contest Preliminary

大學部標準元件數位電路設計

Image Display Control

1. 問題描述

請完成影像顯示控制(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水平及垂直方向的平移(Shift)、影像資料取最大值(Max)、影像資料取最小值(Min)、影像資料平均(Average)、逆時針旋轉(Counterclockwise Rotation)及順時針旋轉(Clockwise Rotation)與X軸鏡像(Mirror X)及Y軸鏡像(Mirror Y)功能。

本題請完成一Image Display Control電路(後文以LCD_CTRL電路表示)，其輸入灰階影像存放於Host端的**輸入圖像ROM模組(IROM)**中，LCD_CTRL電路須從Host端的IROM記憶體模組讀取灰階影像資料，再依題目要求完成影像資料取最大值(Max)、影像資料取最小值(Min)、影像資料平均(Average)、逆時針旋轉(Counterclockwise Rotation)及順時針旋轉(Clockwise Rotation)與X軸鏡像(Mirror X)及Y軸鏡像(Mirror Y)與水平及垂直方向的平移(Shift)運算，運算後的結果需寫入Host端的**輸出結果圖像RAM模組(IRAM)**內，並在整張圖像處理完成後，將done訊號拉為High，接著系統會比對整張圖像資料的正確性。有關本題電路訊號的定義與LCD_CTRL運算方式，詳述於後。

本電路各輸入輸出信號的功能說明，請參考表一。本電路系統方塊圖請參考圖1。

每個參賽隊伍必須根據下一節所給的設計規格及附錄B中的測試樣本完成設計驗證。

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。

軟體環境及設計檔案說明請參考附錄A與附錄B。

2. 設計規格

請注意：

1. 各模組及檔案名稱、大小寫須完全符合**附錄B規範**，若有引入其它模組或檔案請自行寫在設計檔內，測試檔不允許任何修改否則不予計分。
2. 最後評分方式為使用最後上傳檔案版本評分，並以最後上傳檔案版本時間為依據，請參考3.評分標準。

2.1 系統方塊圖

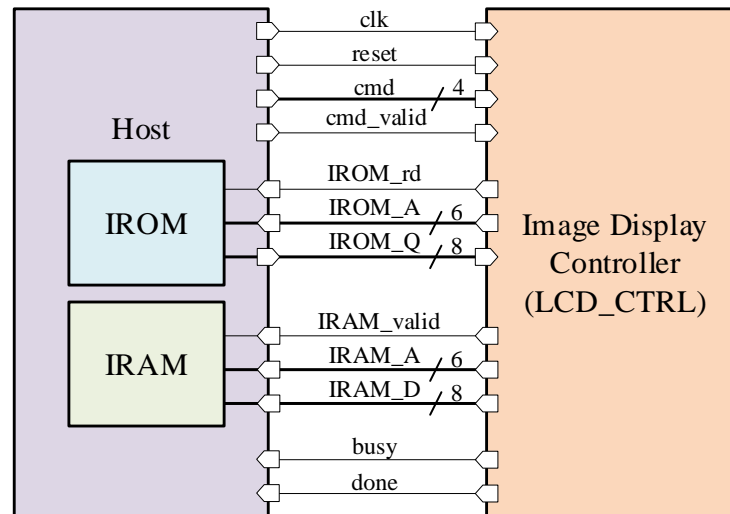


圖 1、系統方塊圖

2.2 輸出/輸入介面

表 一、輸入/輸出訊號

信號名稱	I/O	位元寬度	說明
<i>clk</i>	input	1	此系統為同步於時脈正緣(posedge)之同步設計。
<i>reset</i>	input	1	高位準非同步(active high asynchronous)之系統重置信號。
<i>cmd</i>	input	4	指令輸入信號。 說明:本控制器共有十一種指令輸入，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	當本信號為 high 時表示 <i>cmd</i> 指令為有效指令輸入。
<i>IROM_rd</i>	output	1	Image ROM 記憶體讀取致能訊號。當為 High 時，表示 LCD_CTRL 端要向 Host 端索取資料。
<i>IROM_A</i>	output	6	Image ROM 位址匯流排。LCD_CTRL 端需透過此匯流排向 Host 的灰階圖像記憶體索取該位址的灰階影像資料。
<i>IROM_Q</i>	input	8	Image ROM 資料匯流排。Host 端利用此匯流排將灰階圖像記憶體內的灰階圖像資料送到 LCD_CTRL 端。
<i>IRAM_valid</i>	output	1	Image RAM 記憶體資料致能訊號。當為 High 時，表示 LCD_CTRL 端所傳輸之灰階圖像資料及位址匯流排為有效的。
<i>IRAM_A</i>	output	6	Image RAM 位址匯流排。LCD_CTRL 端需透過此匯流排向 Host 端的 Image RAM 記憶體指示寫入該位址的灰階圖像資料。
<i>IRAM_D</i>	output	8	Image RAM 資料匯流排。LCD_CTRL 端利用此匯流排將圖像資料寫入到 Host 端的 Image RAM 記憶體模組。
<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。 <i>reset</i> 時，default 設定為 high。
<i>done</i>	output	1	當控制器完成寫入 IRAM 時，將 <i>done</i> 設為 high 表示完成。

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，使得顯示端達到平移、平均、旋轉以及鏡像功能，並將經過指令處理完的影像資料寫入IRAM，如圖2所示。

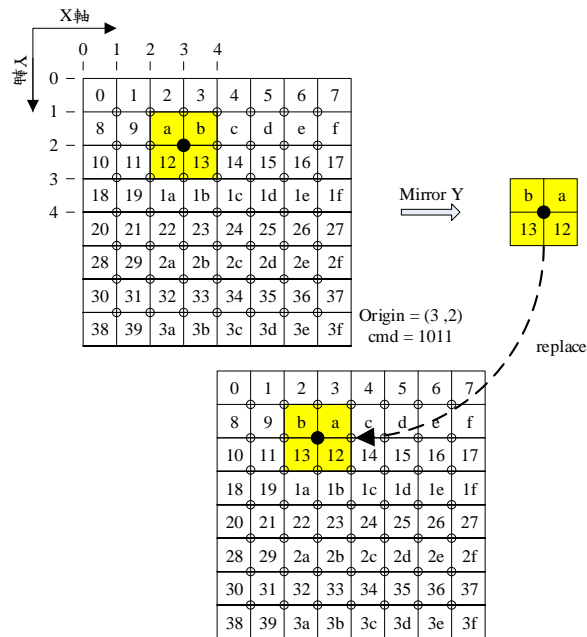


圖 2、系統功能圖

[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3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)，如下圖所示。

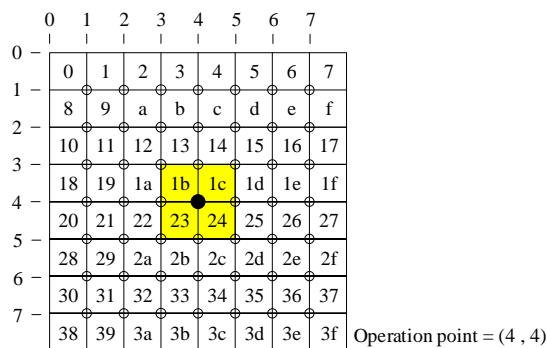


圖 3、輸入端影像操作點

2.3.1 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 (0000)	Write
1 (0001)	Shift Up
2 (0010)	Shift Down
3 (0011)	Shift Left
4 (0100)	Shift Right
5 (0101)	Max
6 (0110)	Min
7 (0111)	Average
8 (1000)	Counterclockwise Rotation
9 (1001)	Clockwise Rotation
a (1010)	Mirror X
b (1011)	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IRAM。

◆ 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。

◆ 影像資料取最大值(Max)

- 本試題定義，當執行影像資料取最大值(Max)指令時，將執行目前操作點座標之影像資料取最大值之計算，即將目前座標所對應的4筆影像資料比較出最大值(即輸出13)，輸出影像資料4筆皆輸出計算後的最大值，並改變原始影像資料。如圖4所示

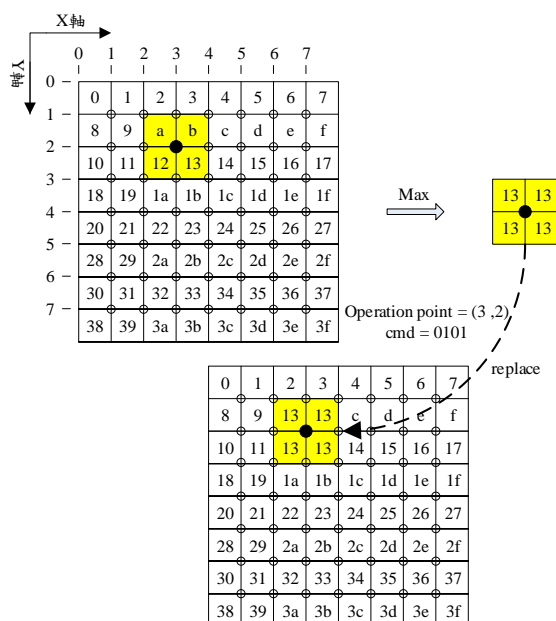


圖 4、影像資料取最大值影像輸出

◆ 影像資料取最小值(Min)

- 本試題定義，當執行影像資料取最小值(Min)指令時，將執行目前操作點座標之影像資料取最小值之計算，即將目前座標所對應的4筆影像資料比較出最小值(即輸出a)，輸出影像資料4筆皆輸出計算後的最大值，並改變原始影像資料。如圖5所示

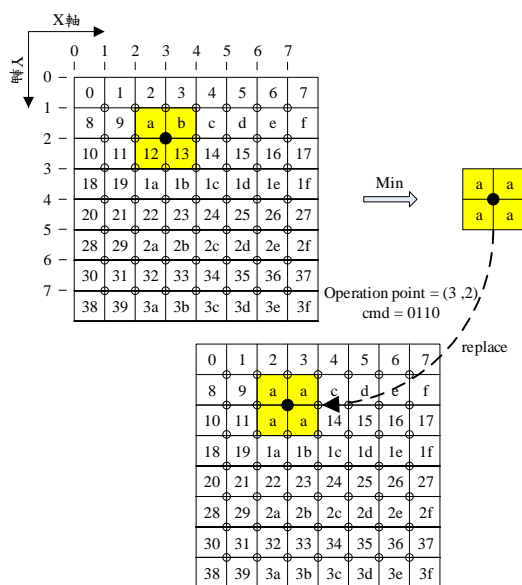


圖 5、影像資料取最小值影像輸出

◆ 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 $(a+b+12+13)/4=e.8$ ，即輸出e)，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6所示。

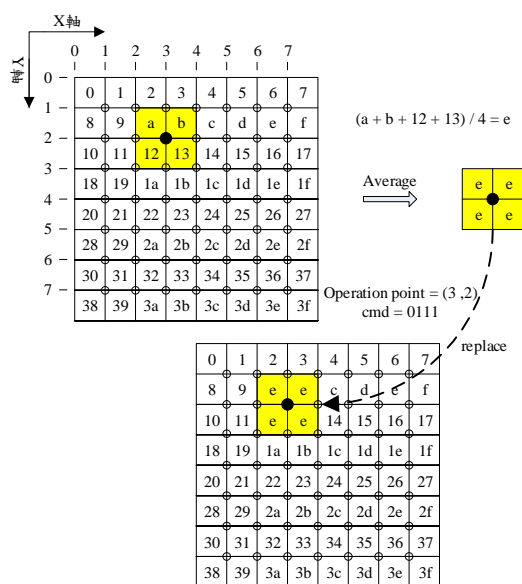


圖 6、取近似平均數影像輸出

◆ 影像資料逆時針旋轉(Counterclockwise Rotation)

- 本試題定義，當執行向左旋轉(Counterclockwise Rotation)指令時，將輸出目前操作點座標之影像資料皆逆時針旋轉，並改變原始影像資料。如圖7所示。

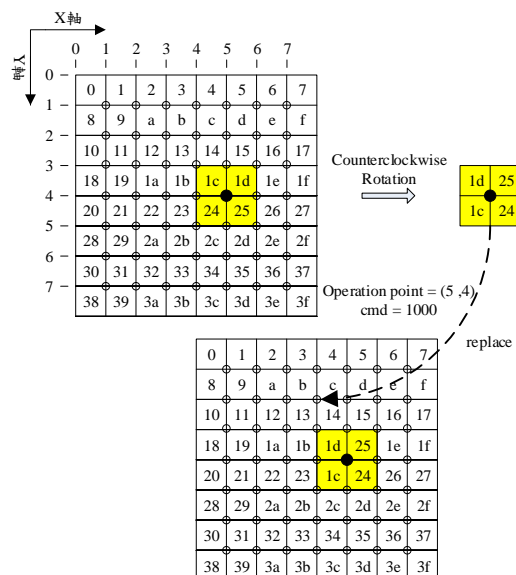


圖 7、逆時針旋轉資料影像輸出

◆ 影像資料順時針旋轉(Clockwise Rotation)

- 本試題定義，當執行順時針旋轉(Clockwise Rotation)指令時，將輸出目前操作點座標之影像資料皆順時針旋轉，並改變原始影像資料。如圖8所示。

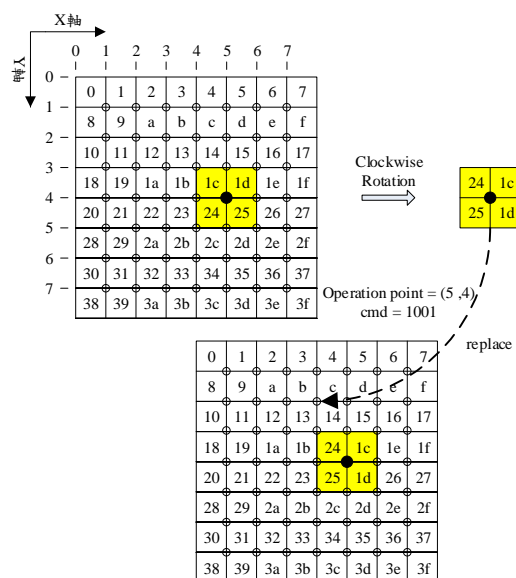


圖 8、向右旋轉資料影像輸出

◆ 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9所示。

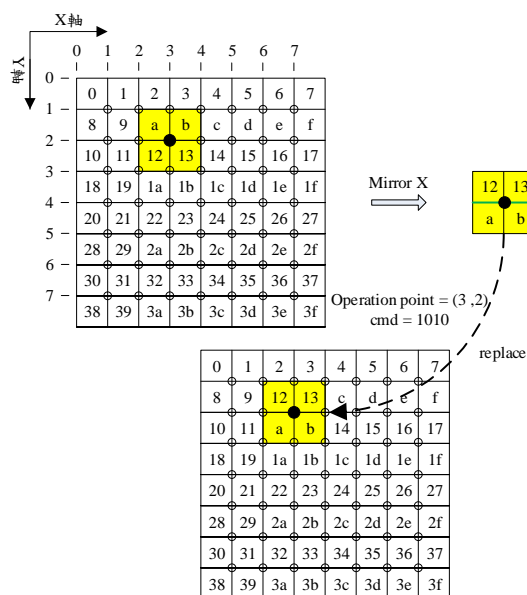


圖 9、X 軸鏡像資料影像輸出

◆ 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10所示。

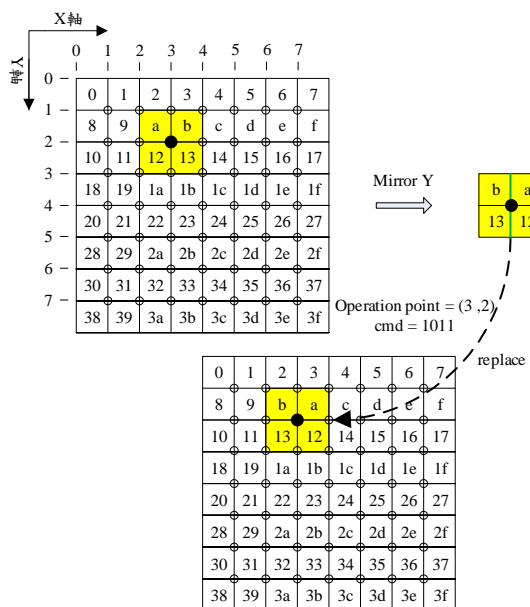


圖 10、Y 軸映像資料影像輸出

2.3.2 輸入原始灰階圖像記憶體(IROM)對應方式及時序規格

輸入原始灰階圖像大小固定為8x8 pixels，每個pixel為8 bits資料，因此Host端的灰階圖像共64個pixels；而IROM記憶體大小為8 bits資料寬度和64個位址。每個位址的8 bits資料剛好存放1個pixels的灰階圖像資料，對應方式如下圖11：

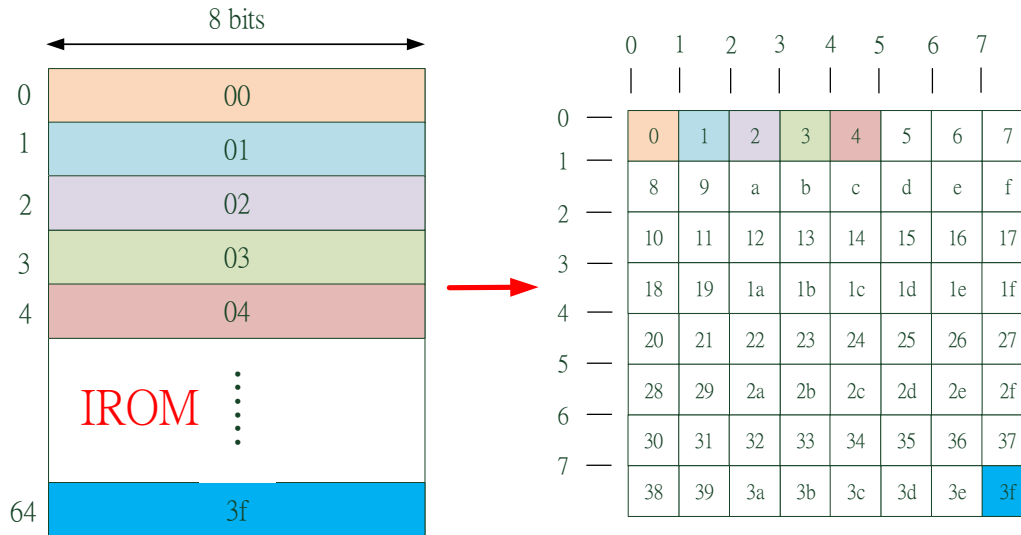


圖 11、輸入灰階圖像記憶體(IROM)對應方式

IROM 記憶體的時序如下圖所示。IROM 會在每個負緣時脈訊號時，若 IROM_rd 訊號為 High(如下圖 12 的 T1 及 T2 時間)，則會將 IROM_A 訊號所指定之位址的資料立即由 IROM_Q 匯流排送到 LCD_CTRL 端；如果 IROM_rd 為 Low(如下圖 12 的 T3 時間)，則 IROM_ROM 將不會進行任何動作。本記憶體不需考慮讀取延遲的問題。

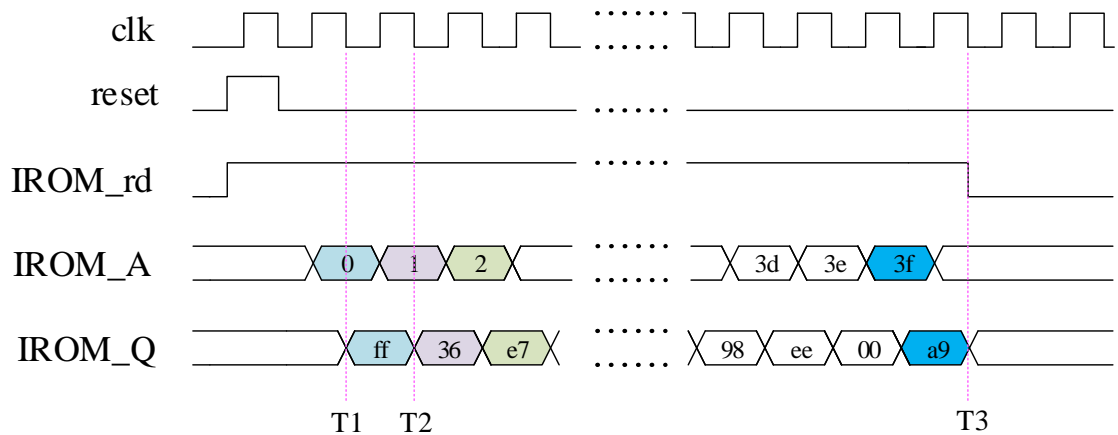


圖 12、輸入灰階圖像記憶體(IROM)時序

2.3.3 輸出結果灰階圖像記憶體(IRAM)對應方式及時序規格

經LCD_CTRL運算完成的輸出灰階圖像為8x8 pixels，每個pixel為8 bits資料，因此Host端的輸出結果灰階圖像記憶體(IRAM)共有64個位址；用以存放各pixel的處理結果。對應方式如下圖13

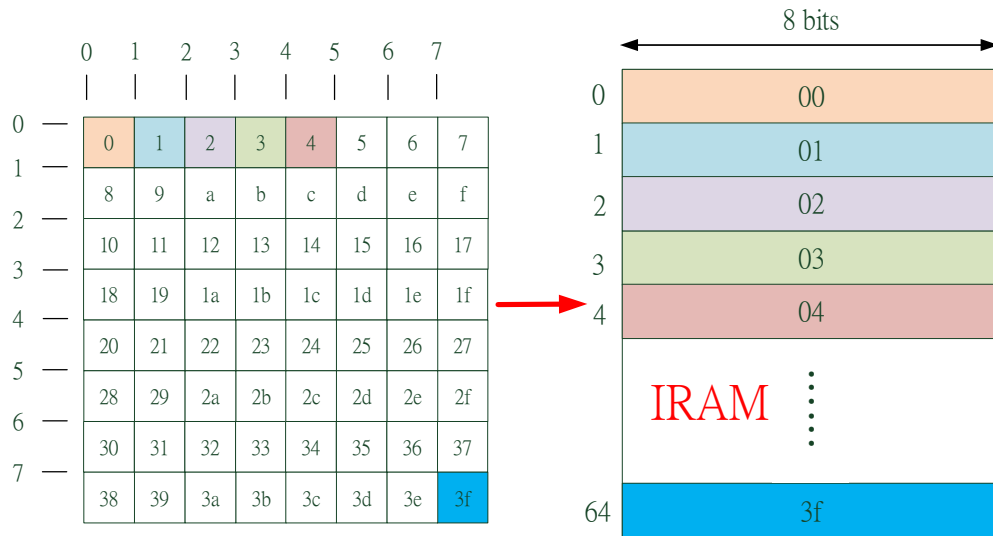


圖 13、輸出結果灰階圖像記憶體(IRAM)對應方式

輸出結果灰階圖像記憶體的時序如下圖 14 所示。IRAM 會在每個負緣時脈訊號時，若 IRAM_valid 訊號拉為 High(如下圖 14 的 T4 時間)，同時把欲寫入的位址及資料分別放在 IRAM_A 及 IRAM_D 匯流排；HOST 在 T5 時間點的時脈訊號負緣觸發時，就會進行寫入的動作。連續寫入資料時，則持續將 IRAM_valid 維持在 High 後改變 IRAM_A 及 IRAM_D 即可。要結束寫入資料的話，請在 T6 時間點將 IRAM_valid 拉為 low。本記憶體不需考慮寫入及讀取延遲的問題。

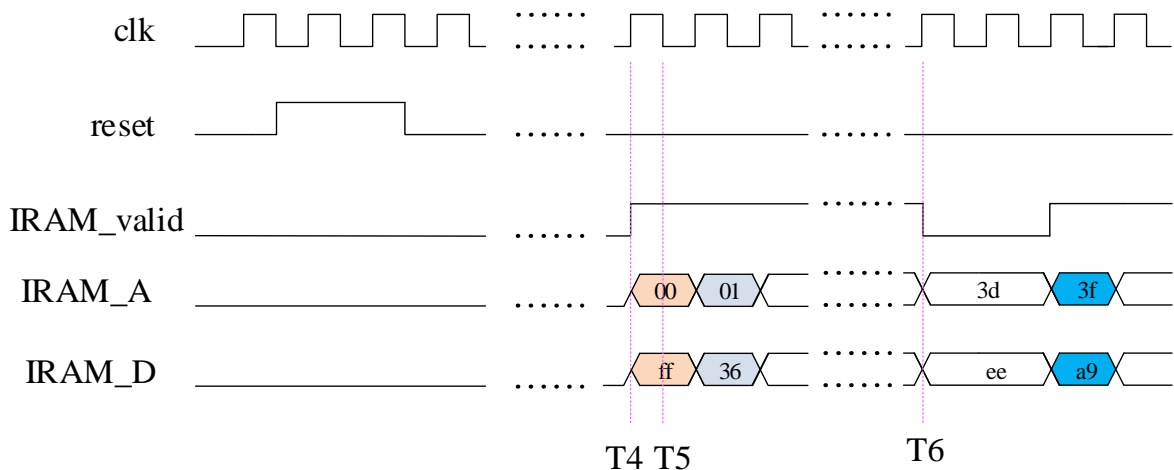


圖 14、輸出結果灰階圖像記憶體(IRAM)時序

2.4 時序規格圖

◆ 重置(Reset)後之時序規格圖，如下圖15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_rd為high時，表示讀取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。

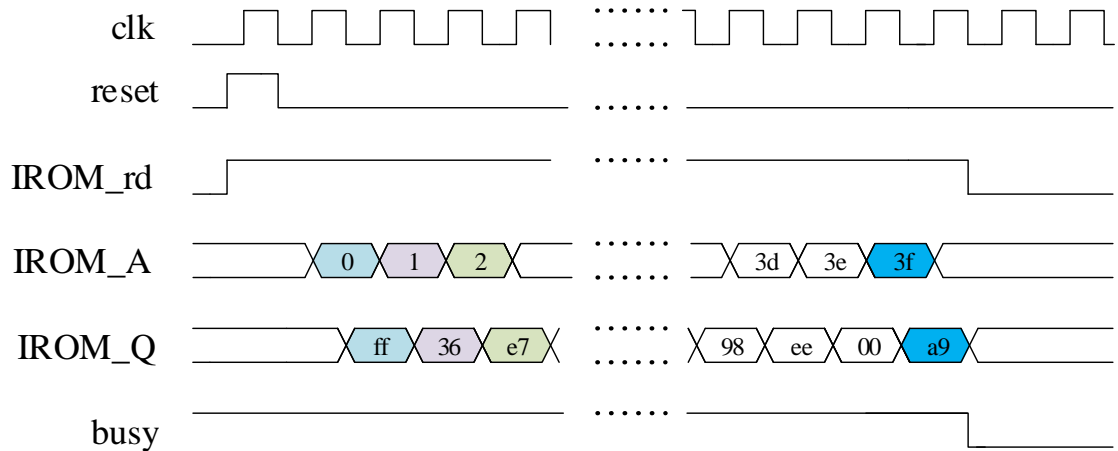


圖 15、資料讀取之時序規格圖

◆ 其他控制指令(Shift up、Shift down、Shift left、Shift right、Max、Min、Average、Counterclockwise rotation、Clockwise rotation、Mirror X、Mirror Y)之時序規格圖，如下圖16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。

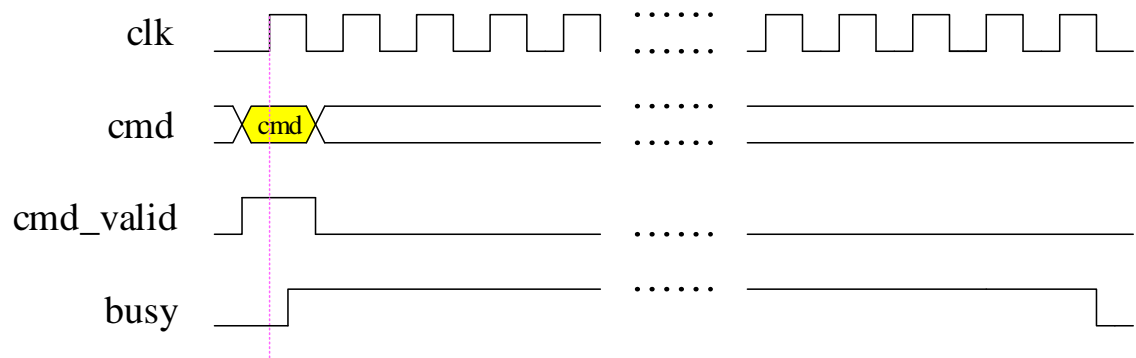


圖 16、其他控制指令之時序規格圖

◆ 寫入指令(write)之時序規格圖，如下圖17所示。

- 執行寫入指令時控制器會將處理完的影像資料寫入IRAM。
- 當IRAM_valid為high時，表示對IRAM寫入，即可輸入位址信號將影像資料寫入IRAM。
- 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IRAM的資料與golden pattern比對。

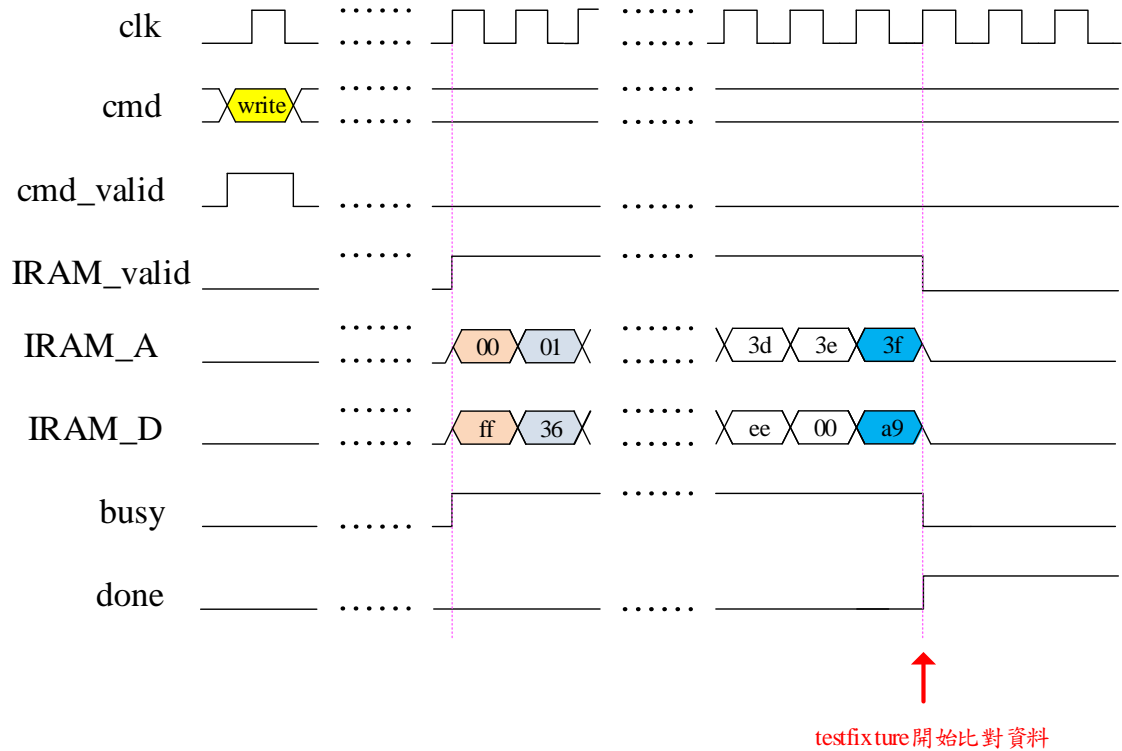


圖 17、寫入指令之時序規格圖

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，並完全符合主辦單位所提供的標準設計結果。

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，請在每完成新等級時即將結果上傳，CIC會以最後上傳版本作為評分結果。

1. A等級：完成測試樣本一、二與三 (tb1 & tb2 & tb3)之RTL與gate-level simulation
2. B等級：完成測試樣本一、二與三 (tb1 & tb2 & tb3)之RTL simulation
3. C等級：完成測試樣本一與二 (tb1 & tb2)之RTL simulation
4. D等級：完成測試樣本一 (tb1)之RTL simulation

B等級、C 等級與D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，完成同一等級者，最後上傳時間越早者為優先錄取。

附錄

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為上傳檔案說明，亦即參賽者必須回傳至CIC的檔案資料；

附錄D為設計檔上傳步驟說明；

附錄E為測試樣本說明。

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三。驗證評分時，係以所列軟體及版本作為驗證依據。

表三、設計軟體版本

Functionality	Corresponding EDA tools
Logic Simulator	Cadence NC-Verilog (INCISIVE_15.20.039) Mentor Modelsim (v10.5c) Synopsys VCS-MX (2017.03)
Logic Synthesizer	Synopsys Design Compiler (2017.09-sp2)
HDL Debugging	Synopsys Verdi (2016.06-sp1-1)、nLint (2016.06-sp1-1)

附錄B 設計檔案說明

1. 以下表四為主辦單位所提供各參賽者的設計檔案

表四、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及預期輸出信號。
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
cmd1.dat	第一組測試樣本之指令
cmd2.dat	第二組測試樣本之指令
cmd3.dat	第三組測試樣本之指令
image1.dat	第一組測試樣本檔案
tb1_goal.dat	第一組比對樣本檔案
image2.dat	第二組測試樣本檔案
tb2_goal.dat	第二組比對樣本檔案
image3.dat	第三組測試樣本檔案
tb3_goal.dat	第三組比對樣本檔案
LCD_CTRL.sdc	Design Compiler 電路合成規範檔。 本規範檔只有 cycle 可以修改，其餘 constraints 皆不可修改。
DC_syn.tcl	Design Compiler 合成之參考 script。
.synopsys_dc.setup	Design Compiler 合成之初始化設定檔。

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/入埠宣告如下列所示：

```

module LCD_CTRL(clk, reset, cmd, cmd_valid, IROM_Q, IROM_rd, IROM_A, IRAM_valid, IRAM_D, IRAM_A, busy, done);
input clk;
input reset;
input [3:0] cmd;
input cmd_valid;
input [7:0] IROM_Q;
output IROM_rd;
output [5:0] IROM_A;
output IRAM_valid;
output [7:0] IRAM_D;
output [5:0] IRAM_A;
output busy;
output done;
endmodule

```


3. 比賽共提供三組測試樣本，參賽者可依下面範例來進行模擬：

RTL Simulation 時使用指令如下：

- 使用 ncverilog 模擬指令範例如下：
`ncverilog testfixture.v LCD_CTRL.v +define+tb1 +access+r`
- 使用 modelsim 模擬，則是在 compiler verilog 時，使用下面指令：
`Vlog testfixture.v +define+tb1 +access+r`

Gate-Level Simulation 時使用指令如下：

- 使用 ncverilog 模擬指令範例如下：
`ncverilog testfixture.v LCD_CTRL_syn.v -v tsmc13_neg.v +define+SDF +define+tb1 +access+r`
define 中加上 SDF 可讓測試程式引入 gate level netlist 的 SDF 檔案資訊。
(需在 testfixture.v 定義 SDF 檔案名稱)

- 上述指令中 `+define+tb1` 指的是使用第一組測試樣本模擬，當使用第二組測試樣本 請自行修改此參數為 `+define+tb2`；當使用第三組測試樣本 請自行修改此參數為 `+define+tb3`。
- `tsmc13_neg.v` 為 Gate-level simulation 時，使用到的 Standard cell simulation model，該檔案位於 CBDBK_IC_Contest_v2.1 裡(參考路徑為 CBDBK_IC_Contest_v2.1/Verilog/tsmc13_neg.v)。

4. Dump 波形檔請參考下列指令：

- NC-Verilog 使用者若想要輸出 FSDb 格式波形檔案:(請先 source verdi 的環境設定檔)，參考上面 ncverilog 模擬指令範例，加入 `+access+r`
- Modelsim 使用者，請直接使用內建波形來除錯

5. Design Compiler 電路合成規範檔 LCD_CTRL.sdc，僅可以修改 cycle 部分，其餘 constraint 皆不可修改：

```
# operating conditions and boundary conditions #
set cycle 10.0 ;#clock period defined by designer
create_clock -period $cycle [get_ports clk]
set_dont_touch_network [get_clocks clk]
set_clock_uncertainty 0.1 [get_clocks clk]
set_clock_latency 0.5 [get_clocks clk]

set_input_delay 5 -clock clk [remove_from_collection [all_inputs] [get_ports clk]]
set_output_delay 0.5 -clock clk [all_outputs]
set_load 1 [all_outputs]
set_drive 1 [all_inputs]

set_operating_conditions -max slow -min fast
set_wire_load_model -name tsmc13_wl10 -library slow
set_max_fanout 20 [all_inputs]
```

← 僅可以修改cycle部分其餘constraint皆不可修改

6. Design Compiler 合成之初始化設定檔.synopsys_dc.setup，需將裡面合成 Library 之搜尋路徑 **search_path** 部分，修改為同學使用之工作站環境之路徑。

```
set company "CIC"
set designer "Student"
set search_path "/usr/cad/designkit/CBDK_IC_Contest_v2.1/SynopsysDC/db" $search_path
set target_library "slow.db fast.db"
set link_library "* $target_library dw_foundation.sldb"
set symbol_library "generic.sdb"
set synthetic_library "dw_foundation.sldb"

set hdlin_translate_off_skip_text "TRUE"
set edifout_netlist_only "TRUE"
set verilogout_no_tri true

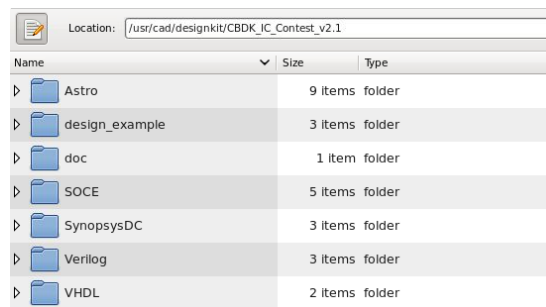
set hdlin_enable_presto_for_vhdl "TRUE"
set sh_enable_line_editing true
set sh_line_editing_mode emacs
history keep 100
alias h history

set bus_inference_style {%s[%d]}
set bus_naming_style {%s[%d]}
set hdlout_internal_busses true
define_name_rules name_rule -allowed {a-z A-Z 0-9 _} -max_length 255 -type cell
define_name_rules name_rule -allowed {a-z A-Z 0-9 _[]} -max_length 255 -type net
define_name_rules name_rule -map {"\\*cell\\" "cell"}
```

↑
修改為同學使用之
工作站環境之路徑

7. CBDK_IC_Contest_v2.1 內容：

- Astro
- Design_example
- doc
- SOCE
- SynopsysDC
- Verilog
- VHDL



Name	Size	Type
Astro	9 items	folder
design_example	3 items	folder
doc	1 item	folder
SOCE	5 items	folder
SynopsysDC	3 items	folder
Verilog	3 items	folder
VHDL	2 items	folder

SynopsysDC 資料夾放置 Design Compiler 合成時，使用到的合成 Library (參考路徑為 CBDK_IC_Contest_v2.1/SynopsysDC/db)

Verilog 資料夾放置 Gate-level simulation 時，使用到的 Standard cell simulation model(參考路徑為 CBDK_IC_Contest_v2.1/Verilog/tsmc13_neg.v)

附錄C 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18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)

表五、評分用檔案

RTL category		
Design Stage	File	Description
N/A	report.xxx	design report
RTL Simulation	*.v or *.vhd	Verilog (VHDL) synthesizable RTL code
Gate-Level category		
Design Stage	File	Description
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
	*_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
通過gate-level simulation之clock cycle time (ns) : 10
--- 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 18、report.000 的範本

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 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 C 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。

本題限制上傳之設計檔僅可使用 tar 或 zip 壓縮格式，使用 rar 或其他格式者一律不予計分。

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

FTP site2 (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。

附錄E 測試樣本

比賽共提供三組測試樣本，為方便設計者除錯之用，將測試樣本之圖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

圖 19、測試樣本一之 image1.dat 圖像資料

- 指令輸入順序：

表六、測試樣本一之 cmd1.dat 輸入指令

順序	指令
0	Write

● 測試樣本二(tb2)

■ 相關資料：image2.dat, cmd2.dat, tb2_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

圖 20、測試樣本二之 image2.dat 圖像資料

■ 指令輸入順序：

表七、測試樣本二之 cmd2.dat 輸入指令

順序	指令	順序	指令
0	Shift Left	8	Shift Right
1	Max	9	Min
2	Shift Up	10	Shift Down
3	Counterclockwise Rotation	11	Average
4	Shift Right	12	Shift Left
5	Mirror X	13	Mirror Y
6	Shift Down	14	Write
7	Clockwise Rotation		

- 測試樣本三(tb3)

- 相關資料：image3.dat, cmd3.dat, tb3_goal.dat

- 影像資料：

cd	12	f3	33	87	24	7a	59
80	41	36	71	bf	ae	ff	66
49	27	94	79	3a	b8	69	d2
a7	3b	30	98	18	25	da	20
22	38	c6	00	26	83	68	25
d7	0a	e3	41	25	e1	37	80
87	94	af	77	28	ec	56	47
5a	37	c1	04	13	66	ff	c0

圖 21、測試樣本三之 image3.dat 圖像資料

■ 指令輸入順序：

表八、測試樣本三之 cmd3.dat 輸入指令

順序	指令	順序	指令
0	Shift Right	23	Shift Left
1	Shift Right	24	Shift Down
2	Min	25	Clockwise Rotation
3	Shift Right	26	Shift Left
4	Clockwise Rotation	27	Max
5	Shift Right	28	Shift Down
6	Mirror Y	29	Mirror X
7	Shift Up	30	Shift Left
8	Average	31	Min
9	Shift Left	32	Shift Left
10	Counterclockwise Rotation	33	Counterclockwise Rotation
11	Shift Up	34	Shift Down
12	Max	35	Mirror Y
13	Shift Left	36	Shift Right
14	Mirror X	37	Average
15	Shift Up	38	Shift Down
16	Clockwise Rotation	39	Shift Down
17	Shift Up	40	Clockwise Rotation
18	Mirror X	41	Shift Down
19	Shift Up	42	Shift Right
20	Mirror Y	43	Mirror X
21	Mirror X	44	Clockwise Rotation
22	Shift Down	45	Write