

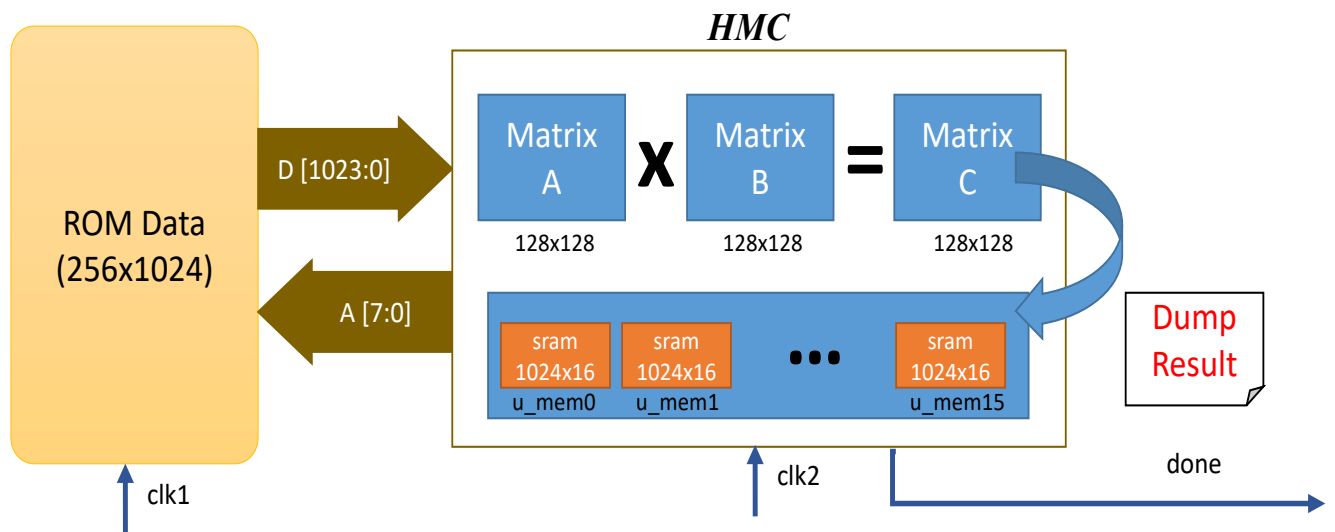
# 2024 IC Design Contest

## Cell-Based IC Design Category for Graduate Level

### *Huge Matrix Calculations*

#### 1.問題描述

請完成 Huge Matrix Calculations (後文以 **HMC** 表示) 的電路設計。HMC 電路功能如圖一所示，從 ROM Data 載入 128x128 的 Matrix A 及 Matrix B 後，將矩陣作相乘運算得到 128x128 的 Matrix C 依序存入至電路內部的 SRAM 後，將本電路的輸出腳位 done 拉為 High，即完成整個電路動作。有關 HMC 電路運作的細項說明將描述於 2.3 節。



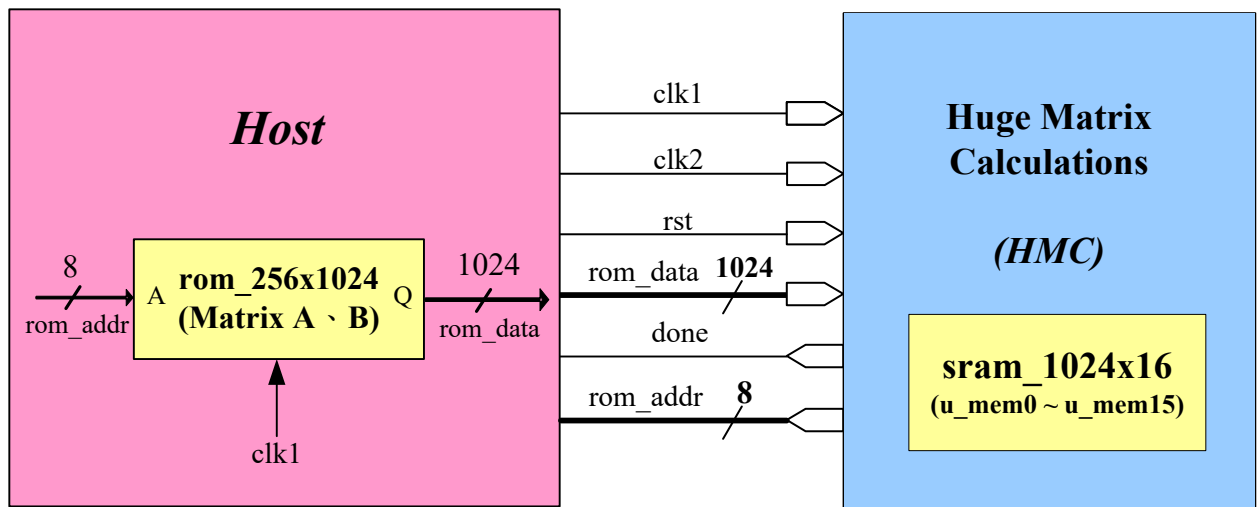
圖一 HMC 電路功能

本次 IC 設計競賽比賽時間為上午 08:30 到下午 20:30。當 IC 設計競賽結束後，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 E 中所列的要求，附上評分所需要的檔案。本題 icc2024cb.tar 之解壓縮方式如下：

***tar xvf /usr/cad/icc2024/bgc/icc2024cb.tar***

## 2.設計規格

### 2.1 系統方塊圖



圖二、系統方塊圖

### 2.2 輸入/輸出介面

表 1 -輸入/輸出訊號

Signal Name	I/O	Width	Simple Description
clk1	I	1	Clock Signal (positive edge trigger) 主要提供給 ROM 使用，使用 Multi-Clock 架構，讓 ROM 的 Load 速率與 Matrix 運算速率得以錯開，以增加本電路效能。 註：可自行與 clk2 搭配作 N 倍數的同步或非同步電路設計。
clk2	I	1	Clock Signal (positive edge trigger) 主要提供給 HMC 內部電路使用，使用 Multi-Clock 架構，讓 ROM 的 Load 速率與 Matrix 運算速率得以錯開，以增加本電路效能。 註：可自行與 clk1 搭配作 N 倍數的同步或非同步電路設計。
rst	I	1	Reset Signal (active high)。 要作同步或非同步 Reset 皆可，可自行決定。
rom_data	I	1024	ROM 的 Data Bus。ROM 的位址區間 0 ~ 127 擺放 Matrix A 的資料，ROM 的位址區間 128 ~ 255 擺放 Matrix B 的資料，每存取一次 1024bits 的資料量相當於一次存取 128 個 8bits Matrix A 或 B 內的數值。

rom_addr	O	8	ROM 的 Address Bus。使用 rom_addr 讀取 ROM 的資料值，Matrix A、B 的同一筆數值可以重覆讀取，並無次數上的限制。
done	O	1	HMC 電路運算完畢的指示訊號。當為 High，表示 Matrix C 運算完畢，整個 Matrix C 的運算結果已依序存放於 SRAM 中，反之，為 Low 表示 Matrix C 的運算尚未完成。

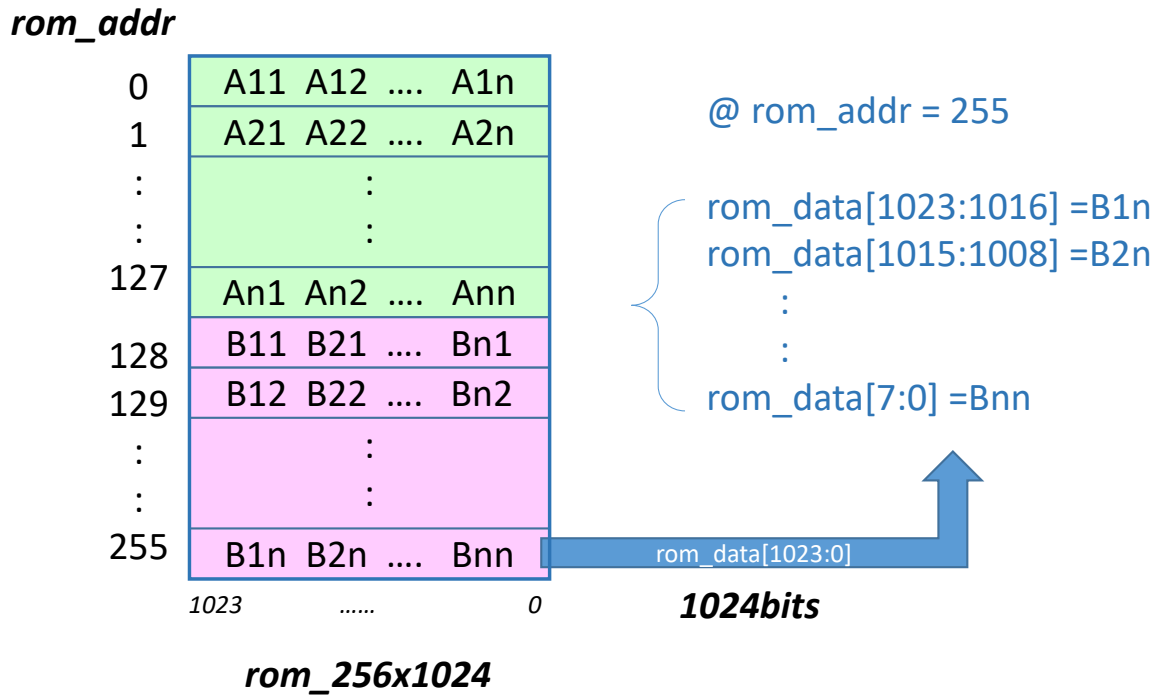
A11	A12		A1n		B11	B12		B1n		C11	C12		C1n
A21	A22		A2n		B21	B22		B2n		C21	C22		C2n
An1	An2		Ann		Bn1	Bn2		Bnn		Cn1	Cn2		Cnn

$$\begin{cases}
 C11 = A11 \times B11 + A12 \times B21 + A13 \times B31 + \dots + A1n \times Bn1 \\
 C12 = A11 \times B12 + A12 \times B22 + A13 \times B32 + \dots + A1n \times Bn2 \\
 \vdots \\
 C21 = A21 \times B11 + A22 \times B21 + A23 \times B31 + \dots + A2n \times Bn1 \\
 C22 = A21 \times B12 + A22 \times B22 + A23 \times B32 + \dots + A2n \times Bn2 \\
 \vdots \\
 \vdots
 \end{cases} \quad (1)$$

圖三、 HMC 電路 – Matrix 運算方式 (n=128)

## 2.3 系統描述

HMC 電路功能如圖三所示，將 128x128 的 Matrix A 及 Matrix B 的數值分別從 ROM 載入後，進行式(1)的矩陣運算後得到的 Matrix C 存放至 HMC 電路中內建的 SRAM 即完成整個電路動作。關於 HMC 電路細項的運作方式說明如下：



圖四、 Matrix A、B 儲存於 ROM 的方式 (n=128)

### 2.3.1 HMC 電路的輸入

如圖四所示，128x128 的 Matrix A 已依序儲存 ROM 的位址區間 0 ~ 127，128x128 的 Matrix B 已依序儲存 ROM 的位址區間 128 ~ 255。本題 Matrix A、B 中的每個數值皆為 **8bits 之有號整數值**，因此數值範圍為 **-128 ~ 127**，而 ROM 的一次資料讀取為 1024bits，因此一次資料讀取剛好為 128 個矩陣數值，如圖四範例，當 rom\_addr 為 255，可讀取到 rom\_data 共 1024bits，其中 rom\_data[1023:1016] 相當於式(1)中 B1n 數值，rom\_data[1015:1008] 相當於式(1)中 B2n 數值，依此類推。

註：Matrix A 儲存於 ROM，是將式(1)中 Matrix A 的 **整個 row 儲存成 1024 bits**。

註：Matrix B 儲存於 ROM，是將式(1)中 Matrix B 的 **整個 column 儲存成 1024 bits**，**存法不同**。

註：已從 ROM 讀取過的 Matrix 數值，若有需要仍可重覆讀取，並無次數上的限制。

註：本題式中的 **n = 128**。

### 2.3.2 HMC 電路的運算方法

HMC 電路的運算即為 Matrix 的運算，如圖三所式，請將 128x128 之 Matrix C 計算出來，運算過程中的演算法或 VLSI 架構設計有很多種設計方法，像是 Adder Tree、分塊矩陣(Partitioned Matrices)作法、Systolic Array 架構設計等均可以作考量，在此並未有任何的演算法或 VLSI 架構的限制，但本題 clk2 的頻率必須作到 100MHZ 或以上(即  $CYCLE2 \leq 10ns$ )。

另外，Matrix C 運算後會以 16bits 儲存在 SRAM 中，因此 Matrix C 的「最後結果」，若有 overflow 要以最大正值(32767)或最小負值(-32768)呈現。

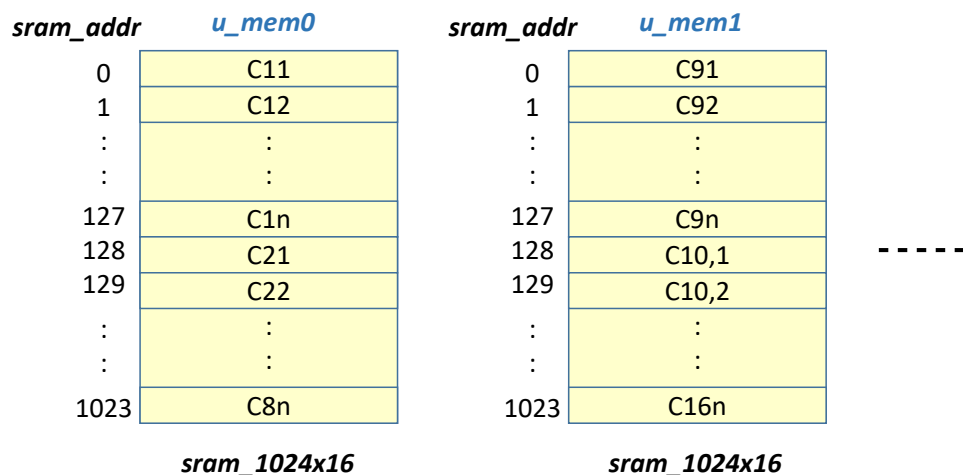
ex:

C11 = 32500      => 以 32500 當作 C11 最後結果  
 C11 = 56878      => 以 32767 當作 C11 最後結果  
 C11 = -68642      => 以 -32768 當作 C11 最後結果

註：請勿使用 Golden Out 的作法，評分時一旦有發現，將視為 Fail。

註：本題規定， $CYCLE2 > 10ns$ ，將視為最差等級-等級 D，詳見評分標準。

註：本題規定， $CYCLE2 \leq 10ns$ ，方可挑戰等級 A ~ 等級 C，詳見評分標準。



圖五、 Matrix C 儲存於 SRAM 的方式 (n=128)

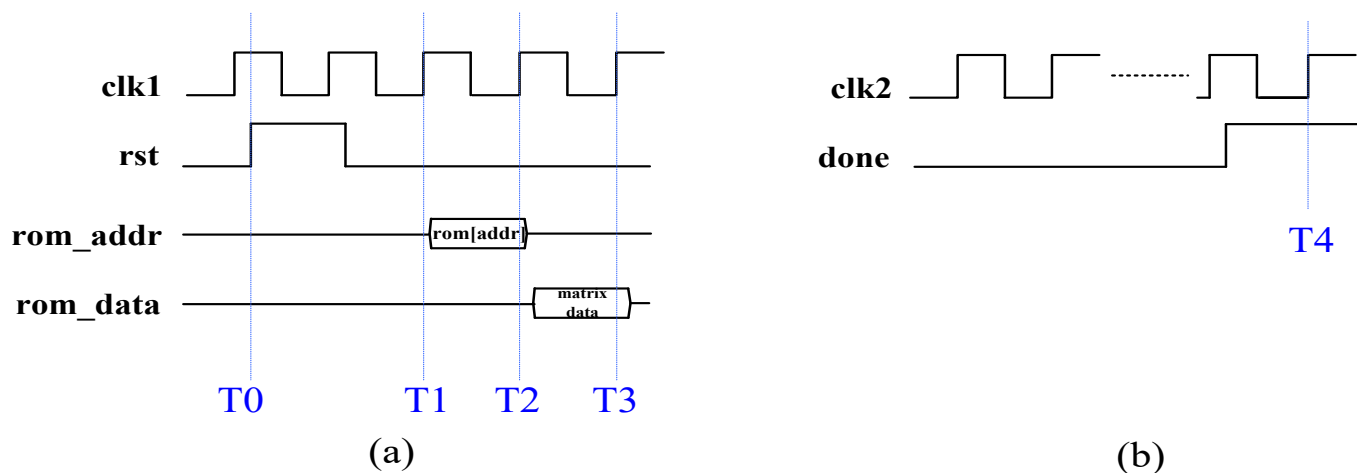
### 2.3.3 HMC 電路的輸出

HMC 電路運算完畢後，把 Matrix C 的 16384 筆(128x128 = 16384)數值，依序存入至 HMC 電路內建的 SRAM 記憶體，儲存順序如圖五所式，完成後可將腳位 done 拉為 High，以告知 Host 端開始進行比對，即為所求。

註: Host 端進行 SRAM 內容值比對時，並不會使用到任何模擬時間。

## 2.4 時序規格

### 2.4.1 HMC 電路輸入與輸出之時序圖



圖六、(a) HMC 電路輸入之時序圖、(b) HMC 電路輸出之時序圖

1. `T0` ~ `T1` 時間點，HMC 電路初始化。(在此假設採同步的 Reset 機制)
2. `T1` ~ `T3` 時間點，為參賽者想到 ROM 去讀取某條 Matrix 數值的範例，`T2` 時間點，Host 端收到 `rom_addr` 位址值，經過 ROM 的  $t_a$  存取時間後，即可從 HMC 電路的 `rom_data` 腳位獲得一整條 Matrix Data，因此從發送位址出去到收到資料，需花一個 Cycle 時間。剩餘的 Matrix Data 也依此類推作讀取，在此不再贅述。
3. `T4` 時間點，當參賽者已將 Matrix C 整個都運算完成後，也將 Matrix C 共 16384 筆結果依序存入 SRAM 記憶體後，可將腳位 `done` 拉為 high，模擬隨即終止。此時 Host 端將會進行 Matrix C 共 16384 筆資料的比對，由於比對期間是不花費任何模擬時間的，因此在此無法繪出比對之時序圖。

### 2.4.2 SARM 記憶體規時序規格

本題在 HMC 電路內建 16 顆 SRAM 記憶體，規格為 `sram_1024x16`，單一顆面積為  $77644.567\mu\text{m}^2$ ，合計 16 顆面積共  $1242313\mu\text{m}^2$ ，評分時 SRAM 面積不納入 score 之計算。有關該記憶體細項規格與記憶體時序圖，詳如記憶體目錄的 PDF 檔中。

註：本題 ROM 並未在電路內部，因此 gate-sim 時 TestBench 不會將 SDF 貼入 ROM 裡，因此 ROM 的各腳位 Timing 規格(包括 setup/hold time)，會參考 `rom_256x1024.v` 的 specify 語法，例如：`rom_addr` 會用到 `$setuphold(posedge CLK &&& re_flag, posedge A[0], 1.000, 0.500, NOT_A0)`；此行解讀為 `rom_addr[0]` 其 setup time 要求 1.0ns、hold time 要求 0.5ns(即 `rom_addr` 的舊值要維持 0.5ns 以上時間)，這要如何實現？提示：可從 `HMC_DC.sdc` 及 `HMC_APR.sdc` 設定 input delay、output delay 的 max、min 自行填入適當的值(也可改用 `set_max_delay` 或 `set_min_delay`)，一旦上述設定有誤，gate-sim 時一定會有 setup、hold time Violations，讀取 ROM 資料也將會有錯。這些為本次競賽實作能力檢定項目之一，請注意。

### 3. 評分標準

評分方式會依設計完成程度，分成 A、B、C、D 四種等級，排名順序為 A>B>C>D，評分項目有兩個，分別為**模擬時間**、**面積**，主辦單位會依此兩項目做為同等級之評分。另外，請參賽者提供一組可正確模擬的 CYCLE1、CYCLE2 TIME 給評分人員驗證本電路正確性之用。

#### ☆ 評分項目一：依”模擬時間”(Time)長短評分

各參賽隊伍將合成或 APR 完成後，執行 Gate-level Simulation 後，會出現模擬時間，評分人員會以此模擬時間如下面範例，紀錄成 **Time = 2162691.800 ns** 做評分。

```
*** Correct at u_mem15.mem[1021]=0cc5 == expect 0cc5 ***
*** Correct at u_mem15.mem[1022]=3dfb == expect 3dfb ***
*** Correct at u_mem15.mem[1023]=0055 == expect 0055 ***

-----
Congratulations! All data have been generated successfully!
-----PASS-----

$finish called from file "testfixture.v", line 398.
$finish at simulation time          2162691800
      V C S   S i m u l a t i o n   R e p o r t
Time: 2162691800 ps
CPU Time:      22.830 seconds;      Data structure size:   0.2Mb
Fri Apr 12 13:27:02 2024
```

**註：** 假若不同測試樣本模擬時間不同，則**以最大的模擬時間作評分**。

#### ☆ 評分項目二：依”面積”(Area)大小評分

各參賽隊伍將 APR 完成後，面積分析方法如下範例，請任選其一 APR 軟體做分析。

1. ICC、ICC2 Report Area 範例:

```
icc_shell> get_attribute [get_die_area] bbox
{0.000 0.000} {2051.640 908.670} => Area =2051.640 x 908.670 = 1864263 um2
```

```
icc2_shell> get_attribute -name boundary -objects HMC
{0.0000 0.0000} {0.0000 897.9100} {2040.0000 897.9100} {2040.0000 0.0000}
=> Area =2040.00 x 897.91 = 1831736 um2
```

2. Innovus Report Area 範例:

```
innovus > analyzeFloorplan
```

```

***** Analyze Floorplan *****
Die Area(um^2)      : 1883777.35
Core Area(um^2)     : 1835988.00
Chip Density (Counting Std Cells and MACROs and IOs): 85.065%
Core Density (Counting Std Cells and MACROs): 87.279%
Average utilization  : 100.000%
Number of instance(s) : 18645
Number of Macro(s)    : 16
Number of IO Pin(s)   : 1036
Number of Power Domain(s) : 0
***** Estimation Results *****
*****

```

=> Area = 1883777 um<sup>2</sup>

**註：指令 analyzeFloorplan 會破壞已完成 routing 的結果，執行該指令後絕對不可再存檔。**

設計完成程度四種等級，說明如下：

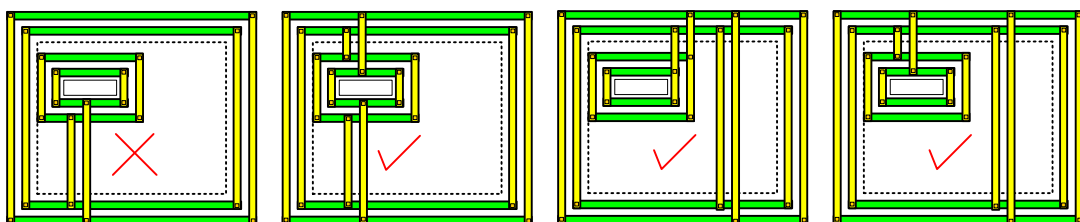
◇ **等級 A：達成“完成設計”之三項要求**

- a、功能正確，RTL 模擬與標準答案比對完全正確。
- b、完成 Synthesis，且 Gate-Level Pre-layout Simulation 結果正確，且無 Setup time、Hold time Violations。
- c、完成 APR，並達成 APR 必要項目，Gate-Level Post-layout Simulation，結果正確@ CYCLE2 ≤10，且無 Setup time、Hold time Violations。

註：完成 APR 必要項目

- i. 只需做 Marco layout (即不用包含 IO Pad、Bonding Pad)。
- ii. VDD 與 VSS Power Ring 寬度請各設定為 2um，只須做一組。
- iii. 不需加 Dummy Metal。
- iv. 內建的所有記憶體 SRAM，其 VDD、VSS Pin 務必要連接至 Core Power Ring，寬度請各設定為 2um。
- v. Power Stripe 務必至少加一組，其 VDD、VSS 寬度各設定為 2um。  
(Power Stripe 垂直方向至少一組，水平方向可不加)
- vi. 務必要加 Power Rail (follow pin)。
- vii. Core Filler 務必要加。
- viii. APR 後之 GDSII 檔案務必產生。
- ix. 完成 APR，DRC/LVS 完全無誤(見附錄 C 說明)。

註: Power Stripe 指的是直接穿過 core area 的 power line，見下圖





等級 A 評分方法：

$$\text{Score} = (\text{Area} - 1242313) \times \text{Time}^2$$

註：Score 越小者，同級名次越好！

註：此 Area 為 APR 後之結果，SRAM 面積 1242313，計算 Score 時將予以扣除。

註：檔案 testfixture.v 的 CYCLE2 ≤ 10，將視為等級 A。

- ◇ 等級 B：已做到 APR，但等級 A 之”APR 必要項目”有部分不符合，DRC/LVS 錯誤總數量容許 10 個(含)以下者

此等級之成績計算方式如下：

$$\text{Score} = (\text{Area} - 1242313) \times \text{Time}^2 \times (\text{DRC} + \text{LVS 的總錯誤量})$$

註：Score 越小者，同級名次越好！

註：此 Area 為 APR 後之結果，SRAM 面積 1242313，計算 Score 時將予以扣除。

註：testfixture.v 的 CYCLE2 > 10，將視為等級 D，請注意。

- ◇ 等級 C：僅完成合成，或做到 APR 但 DRC/LVS 錯誤總數量超過 10 個以上

此等級之成績計算方式如下：

$$\text{Score} = (\text{Area} - 1242313) \times \text{Time}^2$$

註：Score 越小者，同級名次越好！

註：等級 C，視 APR 為 Fail，因此此 Area 為合成的 Cell Area，SRAM 面積同樣予以去除。

註：testfixture.v 的 CYCLE2 > 10，將視為等級 D，請注意。

- ◇ 等級 D：未達成前三等級者，成績計算方式為 RTL Simulation 時的 errors 總數量，errors 越少者分數越高。

$$\text{Score} = \text{Total errors of All RTL Simulations}$$

註：等級 D，Score 評分方式為模擬的 error 總數作相加。

註：等級 D，視合成與 APR 皆為 Fail，Area、Time 將不予考慮。

註：等級 D，只以 RTL Simulation 正確率為主，Score 越小者(即 error 越少)，名次越好。

## 附錄 A 設計檔

1. 下表為主辦單位所提供各參賽者的設計檔

表 2、設計檔案說明

檔名 / 目錄	說明
HMC.v	本題之設計檔，請以此檔案作為 HMC 電路之設計。
testfixture.v	本題的 TestBench。
ROM_PAT0.rcf、ROM_PAT1.rcf (ROM_PAT0.dat、ROM_PAT1.dat)	rcf 檔案為 ROM 的格式，因此 ROM_PAT0.rcf 為第 0 組 Matrix A、B 儲存於 ROM 的測試樣本，ROM_PAT1.rcf 為第 1 組測試樣本。 註：dat 格式僅提供參賽者 Debug 之用，其 Matrix 的數值與存放於 ROM 裡的 rcf 格式皆為相同。
GOLD0.dat ~ GOLD1.dat	HMC 電路輸出 – 本題 Matrix C 的標準解答。 註：檔案已加入至 TestBench，無需額外設定。
HMC_DC.sdc	Design Compiler 作合成之 Constraint 檔案，請自行設定 period 的期望值，input delay 及 output delay 數值請自行修改，要設多少 gate-sim 才能模擬正確，此為本題實作的考試項目之一。
HMC_APR.sdc	Innovus、ICC、ICC2 作 APR 之 Constraint 檔案，請自行設定 period 的期望值，input delay 及 output delay 數值請自行修改，要設多少 gate-sim 才能模擬正確，此為本題實作的考試項目之一。
.synopsys_dc.setup	使用 Design Compiler 作合成或 ICC、ICC2 Layout 之初始化設定檔。參賽者請依 Library 實際擺放位置，自行修改 Search Path 的設定。 註：無論合成或 APR，只需使用 slow library。

2. 本題會「有」隱藏的測試樣本，參賽者仍可依本題 Matrix 數值特性作最佳化處理以達到最佳的面積、時間，因為隱藏版樣本只是將 Matrix 數值的順序作對調。若參賽者以 Golden Output 的方式製作本題，評分等級將被列為 Fail，請注意!

3. 使用 Innovus、ICC2 作 APR 請注意，模擬時請務必自行加上+maxdelays 參數。另外，2 種測試樣本的模擬切換，請自行用+define+sim0、+define+sim1 作切換。

模擬時參考語法如下：

```
> ncverilog +ncmaxdelays testfixture.v HMC_pr.v -v tsmc13_neg.v -v sram_1024x16.v
-v rom_256x1024.v +define+SDF+sim0 +access+r
> vcs -R +maxdelays testfixture.v HMC_pr.v -v tsmc13_neg.v -v sram_1024x16.v
-v rom_256x1024.v -debug_access+all -debug_region+cell +memcbk +define+SDF+sim0
```

## 附錄 B 測試樣本

圖七 ROM\_PAT0.dat 為第 0 組測試樣本的 Matrix A、B 數值，左半邊以 16 進制表示 128x128 Matrix 的每一細項資料，右半邊註解區則以 10 進制表示，以方便參賽者 Debug 之用，其對應到真實擺放於 ROM 的特殊格式檔案為 ROM\_PAT0.rcf，圖七 GOLD0.dat 為第 0 組測試樣本經由 HMC 電路計算後之結果，左半邊以 16 進制表示計算後 Matrix C 的值，右半邊註解區則以 10 進制表示以方便參賽者 Debug 之用。後續的 GOLD1 其格式也是如此，不再贅述。

E9	//matrixA(0,0) = -23	F739	//matrixC(0,0) = -2247
F3	//matrixA(0,1) = -13	F80F	//matrixC(0,1) = -2033
00	//matrixA(0,2) = 0	2FF6	//matrixC(0,2) = 12278
CC	//matrixA(0,3) = -52	CFEB	//matrixC(0,3) = -12309
F0	//matrixA(0,4) = -16	EEA9	//matrixC(0,4) = -4439
3E	//matrixA(0,5) = 62	7FFF	//matrixC(0,5) = 32767
08	//matrixA(0,6) = 8	7A8B	//matrixC(0,6) = 31371
15	//matrixA(0,7) = 21	DA15	//matrixC(0,7) = -9707
23	//matrixA(0,8) = 35	08AD	//matrixC(0,8) = 2221
C6	//matrixA(0,9) = -58	F035	//matrixC(0,9) = -4043
F8	//matrixA(0,10) = -8	DC7E	//matrixC(0,10) = -9090
1E	//matrixA(0,11) = 30	0319	//matrixC(0,11) = 793
19	//matrixA(0,12) = 25	D63F	//matrixC(0,12) = -10689
22	//matrixA(0,13) = 34	3819	//matrixC(0,13) = 14361
F8	//matrixA(0,14) = -8	0AAE	//matrixC(0,14) = 2734
E4	//matrixA(0,15) = -28	1E60	//matrixC(0,15) = 7776
20	//matrixA(0,16) = 32	058D	//matrixC(0,16) = 1421
0E	//matrixA(0,17) = 14	1B7E	//matrixC(0,17) = 7038
E6	//matrixA(0,18) = -26	A679	//matrixC(0,18) = -22919
	⋮		⋮

ROM\_PAT0.dat

GOLD0.dat

圖七、第 0 組測試樣本之 Matrix A、B 數值及 Matrix C 計算之結果

## 附錄 C 設計驗證說明

參賽者繳交資料前應完成 RTL，Gate-Level 與 Physical 三種階段驗證，以確保設計正確性。RTL 與 Gate-Level 階段：進行 RTL simulation、Gate-Level simulation，功能要完全正確。

➤ Physical 階段，包含三項驗證重點：

1. 依本題各項要求，實現完整且正確的 layout (詳細之各項要求，請見評分標準)。
2. 完成 post-layout simulation：參賽者必須使用 P&R 軟體寫出之 Netlist 檔、SDF 檔完成 post-layout gate-level simulation，以下分為 ICC、ICC2、Innovus 三種軟體說明 Netlist、SDF 寫出方式。

i. 使用 Synopsys ICC 者，執行步驟如下：

在 ICC 主視窗底下點選

“File > Export > Write SDF...”

Specify Version	Version 2.1
Instance	空白即可
File name	HMC_pr.sdf
Significant digits	2

按 。

對應指令：**write\_sdf -version 2.1 HMC\_pr.sdf**

“File > Export > Write Verilog...”

先按

Output verilog file name	HMC_pr.v
Output physical only cells	disable
Wire declaration	enable
Backslash before Hierarchy Separator	Enable
All other options	Default value

按 。

ii. 使用 Synopsys ICC2 者，請打指令如下：

```
write_verilog -top_module_first -hierarchy all HMC_pr.v
              -exclude {filler_cells pg_netlist leaf_module_declarations}
write_sdf HMC_pr.sdf
```

**註：“leaf\_module\_declarations”等參數務必要打，以免影響 Gate-level Simulation 正確結果。**

iii. 使用 Cadence Innovus 者，執行步驟如下：

在 Innovus 視窗下點選：

“File → Save → Netlist...”

Netlist File	HMC_pr.v
All other options	Default value

按 。

“Timing → Write SDF...”

Ideal Clock	Disable
SDF Output File:	HMC_pr.sdf

按 。

3. 完成 DRC 與 LVS 驗證：(驗證方法與以往不同，請注意)

**A. DRC: 務必使用 Synopsys ICV 或 Siemens Calibre DRC 作驗證**

a. 準備 P&R 後的 GDSII 檔案，檔名請命名為 HMC\_pr.gds

b. 準備 DRC 驗證檔案

選擇一：Synopsys ICV DRC 驗證檔案，檔名為 **ICV13S\_8M.drc.rs**

選擇二：Siemens Calibre DRC 驗證檔案，檔名為 **Calibre-drc-cur**

c. DRC 驗證方法：

選擇一：使用 Synopsys ICV 作 DRC 驗證

**./run\_icv\_drc**

註：此批次檔可從/usr/cad/icc2024/CBDK\_IC\_Contest\_v2.5/icv/drc 複製過來使用。

正確結果：開啟 HMC.LAYOUT\_ERRORS 檔案，看到 CLEAN 表示正確，如圖。

```
LAYOUT ERRORS RESULTS: CLEAN

#####
# # # # #
# # # # #
# # # # #
# # # # #
#####

=====

Library name:      ./HMC_pr.gds
Structure name:    HMC
Generated by:      IC Validator RHEL64 S-2021.06.6545598 2021/05/29
Runset name:       ICV13S_8M.drc.rs
User name:        andy
Time started:      2024/04/08 11:35:24AM
Time ended:        2024/04/08 11:35:59AM

Called as: icv -vue -i ./HMC_pr.gds -c HMC -host_init 4 -host_login ssh ICV13S_8M.drc.rs

ERROR SUMMARY
```

選擇二：使用 Siemens Calibre 作 DRC 驗證

**calibre -drc -hier Calibre-drc-cur**

正確結果：作到 TOTAL RESULT GENERATED = 0 表示正確，如圖。

```
--- CALIBRE::DRC-H EXECUTIVE MODULE COMPLETED. CPU TIME = 31 REAL TIME = 16
--- TOTAL RULECHECKS EXECUTED = 701
--- TOTAL RESULTS GENERATED = 0 (0)
--- DRC RESULTS DATABASE FILE = DRC_RES.db (ASCII)

--- CALIBRE::DRC-H COMPLETED - Fri Apr 12 14:40:42 2024
--- TOTAL CPU TIME = 31 REAL TIME = 19
--- PROCESSOR COUNT = 16
--- SUMMARY REPORT FILE = DRC.rep
```

**B. LVS: 務必使用 P&R 軟體內建之 LVS 作驗證**

以下分為 ICC、ICC2、Innovus 軟體執行步驟說明。

i. 使用 Synopsys ICC 者，驗證 LVS 步驟如下：

在 ICC Layout 視窗底下點選

“Verification > LVS ...”

Pins not connected to a wire segment(Floating port)	disable
All other options	Default value

按 。

將跳出 Error Browser 視窗，檢查看看是否有錯，若有請自行修正到 0 個 Violation 為止。

- ii. 使用 ICC2 者，驗證 LVS 指令如下：

**check\_lvs**

註：作 check\_lvs 之前，建議再作一次 connect\_pg\_net，以避免有假錯產生。

註：若 LVS 有發生錯誤，請選“View → Error Browser...”查明原因。

- iii. 使用 Cadence Innovus 者，驗證 LVS 步驟如下：

在 Innovus 視窗下點選

請選“Verify → Verify Connectivity...” Default 值，按 。

註：若 LVS 有發生錯誤，請選“Tools → Violation Browser...”查明原因。

### C. Macro Layout 的注意事項

- a. 使用 ICC2 作 Macro Layout 的 Pin 腳位擺放方法

**place\_pins -self**

即可！

- b. 使用 Innovus 作 Macro Layout 的 Pin 腳位擺放方法

請在 Innovus 視窗下點選“File → Save → I/O File...”

Save IO	sequence
To File	HMC.io
Generate template IO File	enable

按 。

請選“File → Load → I/O File...”

請點選 HMC.io，按 。

即可！

## 附錄 D 評分用檔案

評分所須檔案可以下幾個部份：(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的各 module 檔放進來，以免評審進行評分時，無法進行模擬；(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)Physical design，使用 **Synopsys ICC、ICC2** 者，請分別將整個 **Milkyway、NDM Library** 等相關的 design database，壓縮成一個檔案(指令打法：**`tar cvf apr.tar your_apr_database`**)。使用 **Cadence Innovus** 者，**假設我是 B70 這組**，我就**在 Innovus 軟體裡輸入(指令打法：`saveTestcase -name B70 -gzip`)**，即可產生 **B70.tar.gz** 檔案，請將此檔案繳交至 **~/result** 去即可，此檔案就是 APR 整個完整結果。

表 3

<i>RTL category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
N/A	N/A	Design Report Form
<i>Gate-Level category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
Pre-layout	HMC_syn.v	Verilog gate-level netlist generated by Synopsys Design Compiler
Gate-level Simulation	HMC_syn.sdf	Pre-layout gate-level sdf
<i>Physical category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
P&R	*.tar	<b>ICC/ICC2 design database or Innovus testcase</b>
	*.gds	GDSII layout
Post-layout	HMC_pr.v	Verilog gate-level netlist generated by Cadence Innovus or Synopsys ICC、ICC2
Gate-level Simulation	HMC_pr.sdf	Post-layout gate-level sdf

## 附錄 E 檔案整理步驟

當所有的文件準備齊全如表 3 所列，請按照以下的步驟指令，提交相關設計檔案，將所有檔案複製至同一個資料夾下，步驟如下：

1. 在自己的 home directory 建立一個新目錄，名稱叫做“**result**” 例如：

```
> mkdir ~/result
```

2. 將附錄 D 要求的檔案複製到 result 這個目錄。例如：

```
> cp HMC.v ~/result/
```

```
> cp HMC_pr.v ~/result/
```

```
.....
```

3. 在 Design Report Form 中，填入所需的相關資訊。

## 附錄 F 軟體環境

1. 使用者登入後自動會設定好以下軟體環境：

Vendor	Tool	Executable
Cadence	Virtuoso *1	icfb
	Composer	icfb
	NC-Verilog	ncverilog
	Innovus	innovus
Synopsys	Design Compiler	dv, dc_shell
	<b>VCS</b>	<b>vcs</b>
	IC Compiler	icc_shell -gui
	IC Compiler 2	icc2_shell -gui
	Hspice	hspice
	Cosmos Scope *1	cscope
	Custom Compiler *1	custom_compiler
	Custom Explorer *1	wv
	Laker *1	laker
	Laker ADP*1	adp
	Verdi *1	verdi, nWave
	IC Validator *4	icv -vue
Siemens	Calibre *3	calibre
	QuestaSim	vsim
Utility	vi	vi, vim
	gedit	gedit
	nedit	nedit
	pdf reader	acroread
	calculate	gnome-calculator, bc -l
	gcc	gcc
	<b>Matlab</b>	<b>matlab</b>

EDA 軟體所須使用的 license 皆已設定完成，不須額外設定

\*1 該軟體限定使用 1 套 license

\*3 該軟體限定使用 3 套 license



## 附錄 G 設計資料庫

設計資料庫位置： /usr/cad/icc2024/CBDK\_IC\_Contest\_v2.5

### 目錄架構

#### ICC/

tsmc13gfsg_fram/	ICC core library
tsmc13_CIC.tf	ICC technology
macro.map	layer mapping file
tluplus/	
t013s8mg_fsg_typical.tluplus	t13 tluplus file
t013s8mg_fsg.map	t13 tluplus mapping file

#### ICC2/

tsmc13gfsg.ndm/	ICC2 core NDM
tsmc13gfsg_physical_only.ndm/	ICC2 core filler NDM

#### SOCE/

lef/	
tsmc13fsg_8lm_cic.lef	LEF for core cell
lib/	
slow.lib	worst case for core cell
streamOut.map	layout map for GDSII out

#### SynopsysDC/ db/

slow.db	synthesis model (slow)
---------	------------------------

#### Verilog/

tsmc13_neg.v	verilog simulation model
--------------	--------------------------

#### Phantom/

tsmc13gfsg_fram.gds	standard Cell GDSII file
---------------------	--------------------------

#### Calibre/

drc/	
Calibre-drc-cur	verify DRC for Calibre

#### icv/

drc/	
ICV13S_8M.drc.rs	verify DRC for ICV
run_icv_drc	run ICV batch file

## Design Report Form

登入帳號(login-id)	(例如: <b>B70</b> )	
<b>RTL category</b>		
<i>Design Stage</i>	<i>Description</i>	<i>File Name</i>
RTL Simulation	RTL 檔案名稱 (RTL file name)	
<b>Gate-Level category</b>		
<i>Design Stage</i>	<i>Description</i>	<i>File Name</i>
Pre-layout Gate-level Simulation	Gate-Level 檔案名稱 (Gate-Level Netlist file name)	
	Pre-layout sdf 檔案名稱	
	Gate-Level simulation, 所使用的 <b>CYCLE1、2 (超過 10ns 為等級 D)</b>	(                      ) ns
<b>Physical category</b>		
<i>Design Stage</i>	<i>Description</i>	<i>File Name or Value</i>
P&R	使用之 P&R Tool (請填入 ICC 或 ICC2 或 Innovus)	
	晶片布局資料庫(下面請擇一) 1. ICC: 請填 Milkyway 名稱 2. ICC2: 請填 NDM 名稱 3. Innovus: <b>假設我是 B70 這組就打 saveTestcase -name B70 -gzip 然後 cp B70.tar.gz ~/result 即可</b>	
	<b>Synopsys ICV / Siemens Calibre</b> DRC 錯誤總數量 (ex: 0 個)	
	<b>APR Tool LVS</b> 錯誤總數量 (ex: 0 個)	
Score (Pre-layout or Post-layout Result) Over All	Gate-level Simulation Total <b>Simulation Time = ?</b> (ex: Time=2162691.8ns 填最長的)	
	Area (扣除 <b>1242313</b> 後的值) (ex: Area = 589423 um <sup>2</sup> ) <b>Area = ?</b>	
	填寫之面積為哪一階段的結果? (請填 Pre-Sim 或 Post-Sim)	
	最後完成之等級?(ex: 等級 A)	
其他說明事項(Any other information you want to specify:(如設計特點 ...)如寫不下 可寫於背面		