

2012 University/College IC Design Contest

Cell-Based IC Design Category for Graduate Level

Multi-Bank Filter

1.問題描述

請完成一 Multi-Bank Filter (後文以 **MBF 表示**) 的電路設計，如圖一所示。此電路可以將任意 8bits(含正負號)的數位訊號 x ，(1)進行低通濾波器(Low Pass Filter 後文以 **LPF 表示**)運算，並將其運算結果輸出至 8bits 之匯流排 y ，(2)進行高通濾波器(High Pass Filter 後文以 **HPF 表示**)運算，並將其運算結果輸出至 8bits 之匯流排 z 。有關 MBF 詳細規格將描述於後。

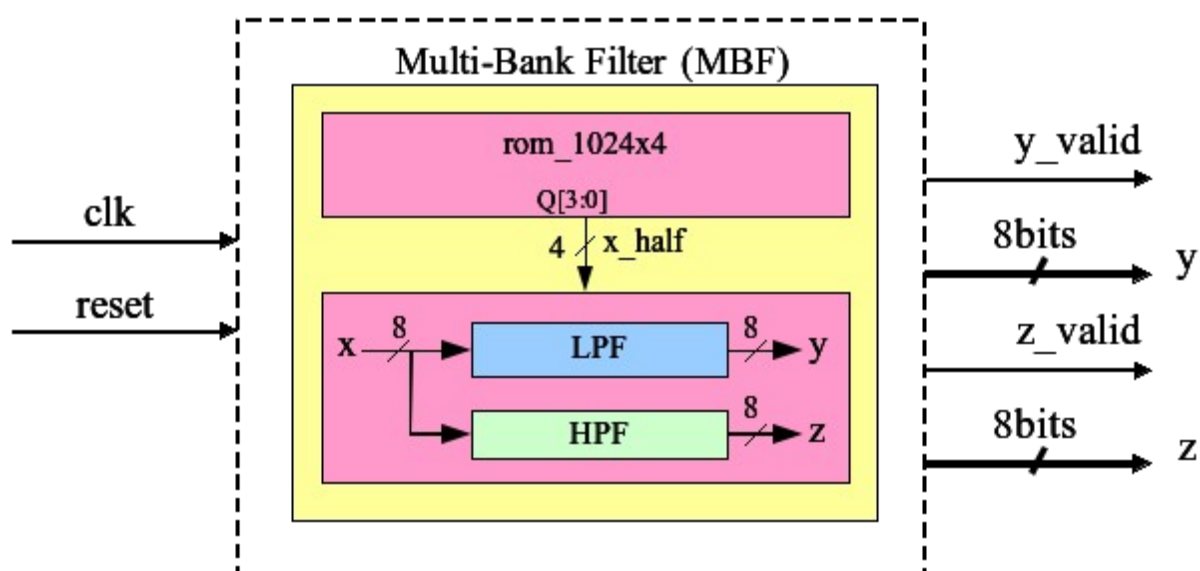
本電路各輸入輸出信號的功能說明，請參考表一。每個參賽隊伍必須根據下一節所給的設計規格及附錄 A 中的測試樣本完成設計驗證。

本次 IC 設計競賽比賽時間為上午 08:30 到下午 20:30。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 E 中所列的要求，附上評分所需要的檔案。

本題目之測試樣本置於 [/usr/cad/icc2012/gcb/icc2012cb.tar](#)，請執行以下指令取得測試樣本：

```
tar xvf /usr/cad/icc2012/gcb/icc2012cb.tar
```

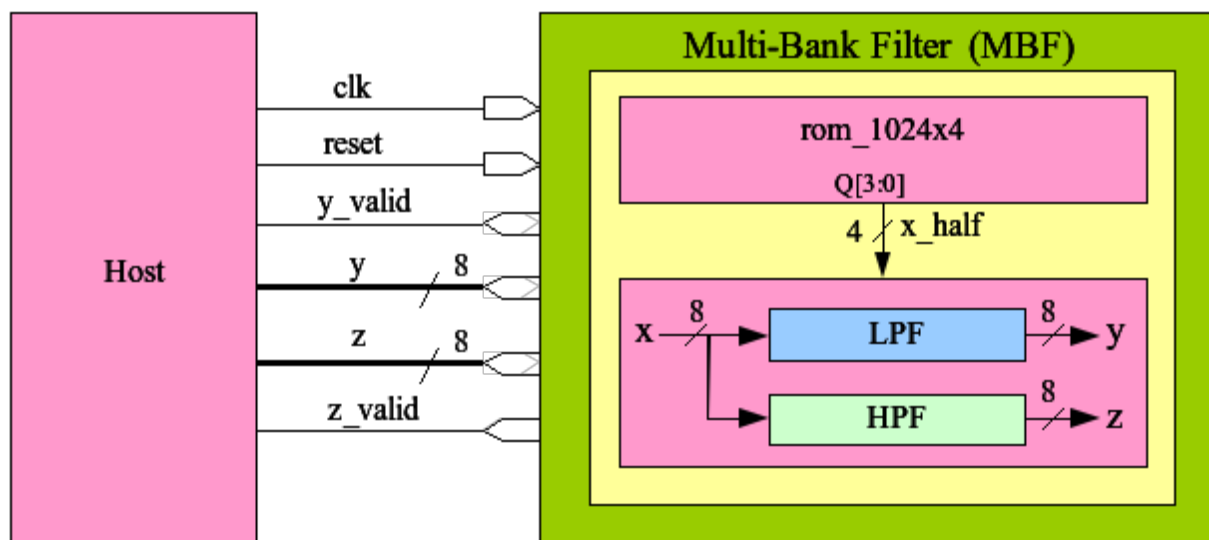
軟體環境及設計資料庫說明請參考附錄 F 與附錄 G。



圖一、Multi-Bank Filter 之方塊圖

2.設計規格

2.1 系統方塊圖

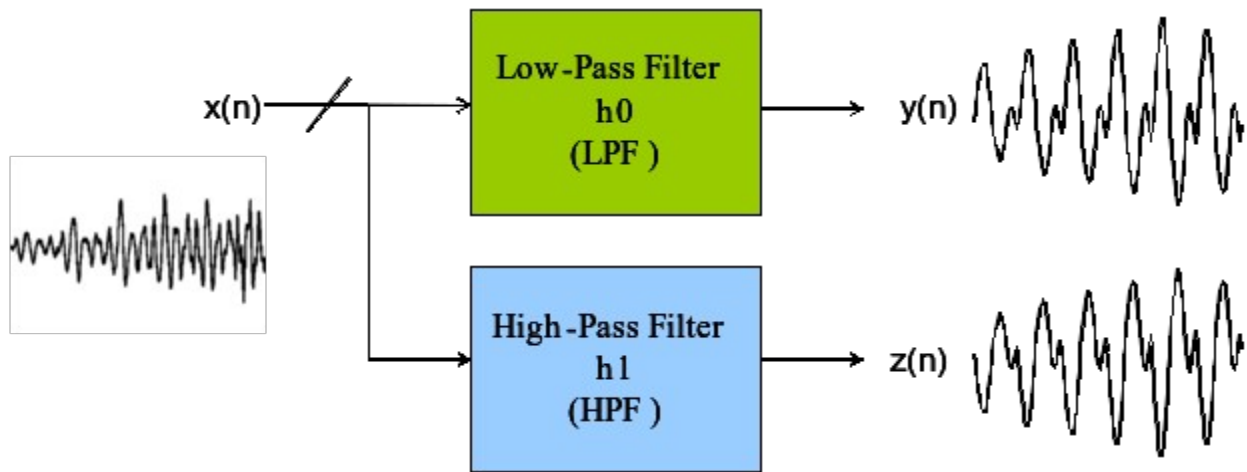


圖二、系統方塊圖

2.2 輸入/輸出介面

表 1 -輸入/輸出訊號

Signal Name	I/O	Width	Simple Description
clk	I	1	本系統為同步於時脈正緣之同步設計。
reset	I	1	高位準”非”同步(active high asynchronous)之系統重置信號。
y_valid	O	1	LPF 輸出資料之有效控制訊號。當為 High 時，表示目前輸出的資料為有效的；反之，當為 Low 時，表示目前輸出資料為無效的，即不被採用。
y	O	8	LPF 資料輸出的匯流排。當 LPF 計算完畢後，可透過此匯流排將運算完畢的數值，輸出至 Host 端。 注意：每一個週期僅能輸出一筆值。
z_valid	O	1	HPF 輸出資料之有效控制訊號。當為 High 時，表示目前輸出的資料為有效的；反之，當為 Low 時，表示目前輸出資料為無效的，即不被採用。
z	O	8	HPF 資料輸出的匯流排。當 HPF 計算完畢後，可透過此匯流排將運算完畢的數值，輸出至 Host 端。 注意：每一個週期僅能輸出一筆值。



圖三、Multi-bank Filter 架構

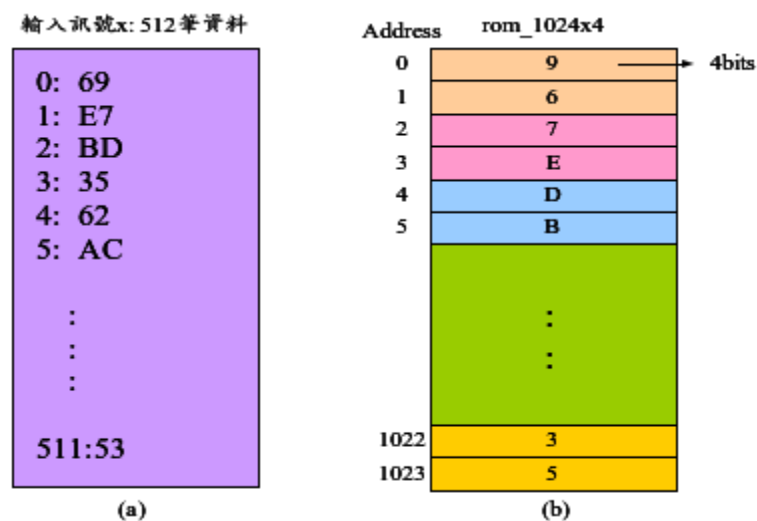
2.3 系統描述

2.3.1 MBF 系統架構

MBF 電路包含了兩種濾波器的運算，LPF 與 HPF 運算，如圖三所示。其中輸入訊號 x 為 8bits 表示，LPF 輸出訊號 y 與 HPF 輸出訊號 z 也皆為 8bits 表示，主辦單位所提供之輸入訊號 x 範圍限定為 $-110 \sim 110$ ，因此濾波器運算後的結果 y 與 z ，其輸出必落在 $-127 \sim 127$ 之範圍。

2.3.2 MBF 運算之輸入方式

本題之輸入訊號 x 並非由輸入腳位(Input Port)作輸入，而是從晶片內部的 ROM(規格 $1024 \times 4\text{bits}$)來讀取資料，該 Rom 的資料匯流排僅有 4bits，因此僅可以擺放 512 筆 8bits 的資料，即位址 0 擺放第一筆資料的 LSB 4bits，位址 1 擺放第一筆資料的 MSB 4bits，位址 2 擺放第二筆資料的 LSB 4bits，位址 3 擺放第二筆資料的 MSB 4bits，依此類推，如圖四所示。



圖四、輸入資料 x 內建在 ROM 的範例(a)512 筆輸入訊號 (b)被儲存在 ROM 的方式

2.3.3 低通濾波器：LPF 運算方式

LPF 其實就是作旋積(Convolution)運算，如式(1)所示，式中 x 為輸入訊號， n 為第 n 筆

資料之意， h_0 為 LPF 係數，其 16 組係數定義如表 2 所示。經過 Convolution 運算後的輸出結果為 y 。

$$y(n) = \sum_{k=0}^{15} x(n) * h_0(n-k) \quad (1)$$

其中，

表 2、LPF 係數

Low-pass Filter Coefficient (h_0)			
$h_0(0)$	- 1.9531250e-003	$h_0(8)$	5.0000000e-001
$h_0(1)$	- 3.9062500e-003	$h_0(9)$	1.5625000e-001
$h_0(2)$	7.8125000e-003	$h_0(10)$	- 7.8125000e-002
$h_0(3)$	2.3437500e-002	$h_0(11)$	- 4.6875000e-002
$h_0(4)$	- 4.6875000e-002	$h_0(12)$	2.3437500e-002
$h_0(5)$	- 7.8125000e-002	$h_0(13)$	7.8125000e-003
$h_0(6)$	1.5625000e-001	$h_0(14)$	- 3.9062500e-003
$h_0(7)$	5.0000000e-001	$h_0(15)$	- 1.9531250e-003

2.3.4 高通濾波器：HPF 運算方式

HPF 其實也是作旋積(Convolution)運算，如式(2)所示，式中 x 為輸入訊號， n 為第 n 筆資料之意， h_1 為 HPF 係數，其 16 組係數定義如表 3 所示。經過 Convolution 運算後的輸出結果為 z 。

$$z(n) = \sum_{k=0}^{15} x(n) * h_1(n-k) \quad (2)$$

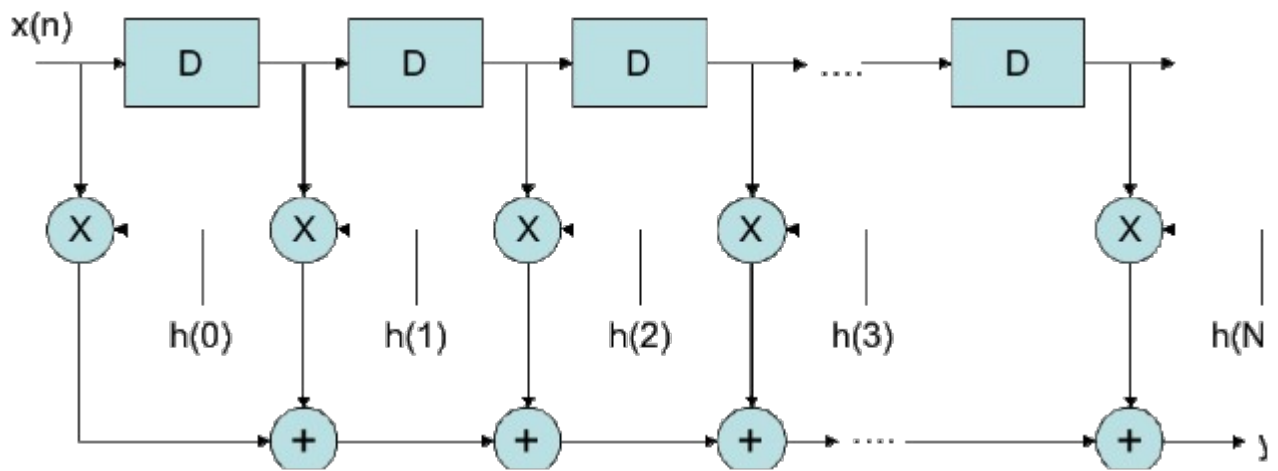
其中，

表 3、HPF 係數

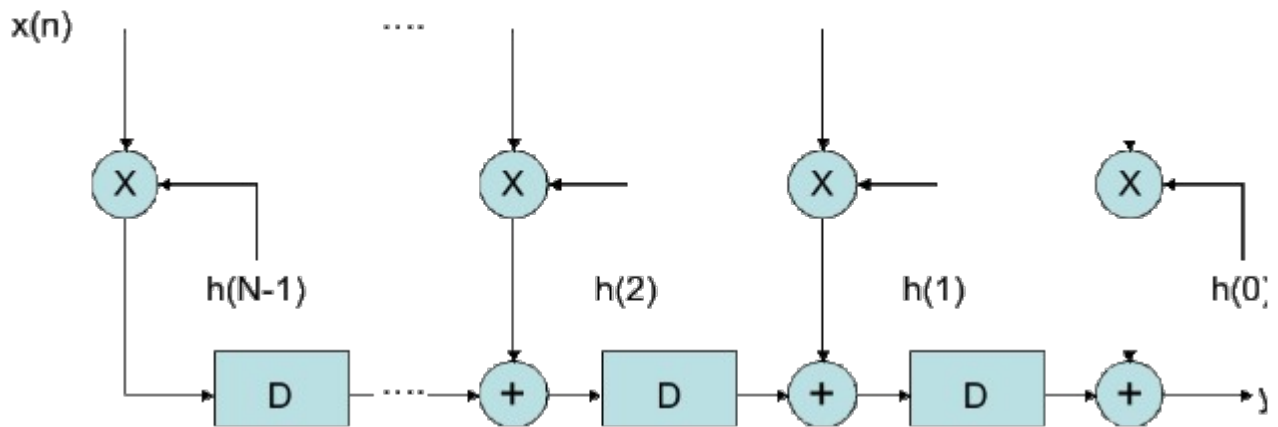
High-pass Filter Coefficient (h_1)			
$h_1(0)$	- 1.9531250e-003	$h_1(8)$	5.0000000e-001
$h_1(1)$	3.9062500e-003	$h_1(9)$	- 1.5625000e-001
$h_1(2)$	7.8125000e-003	$h_1(10)$	- 7.8125000e-002
$h_1(3)$	- 2.3437500e-002	$h_1(11)$	4.6875000e-002
$h_1(4)$	- 4.6875000e-002	$h_1(12)$	2.3437500e-002
$h_1(5)$	7.8125000e-002	$h_1(13)$	- 7.8125000e-003
$h_1(6)$	1.5625000e-001	$h_1(14)$	- 3.9062500e-003
$h_1(7)$	- 5.0000000e-001	$h_1(15)$	1.9531250e-003

2.3.5 MBF 運算之 VLSI 參考架構

關於式(1)、式(2)之旋積運算，VLSI 主要有兩種架構，如圖五(a)、(b)，參賽者可自行決定要使用何種架構。



圖五(a)、旋積運算之 VLSI 架構一

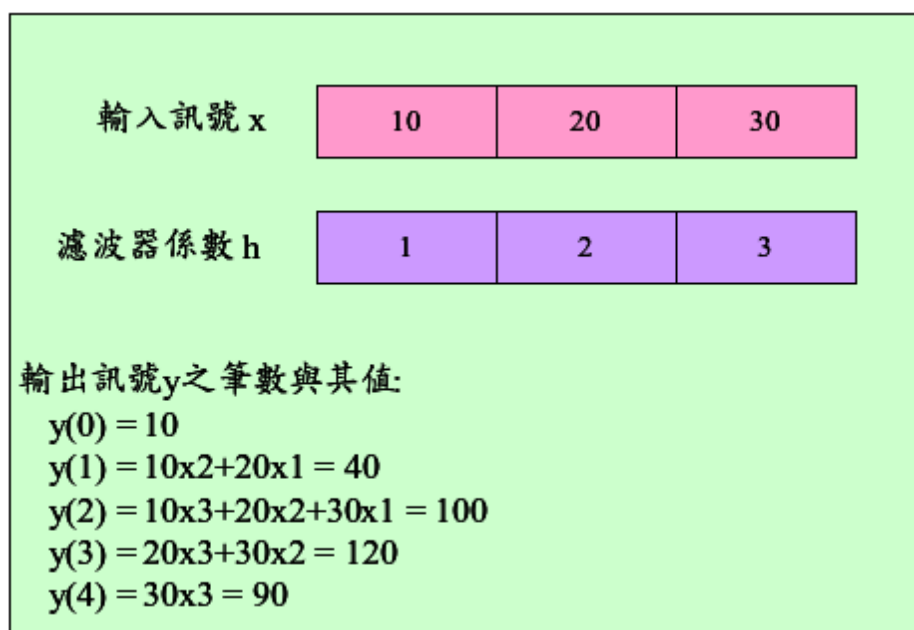


圖五(b)、旋積運算之 VLSI 架構二

2.3.6 MBF 運算之輸出方式

MBF 包含 LPF、HPF 運算，參賽者可自由選擇，先輸出 LPF 或 HPF 的運算結果，也可

以 LPF 與 HPF 一起輸出或交叉輸出皆可。至於輸出的筆數，主辦單位在此規定任何一筆的 Convolution 運算都要作輸出，假設輸入訊號 x 有 N 筆，濾波器係數 h 有 M 筆，其輸出筆數為 $(N+M-1)$ 筆，可參考圖六之範例。

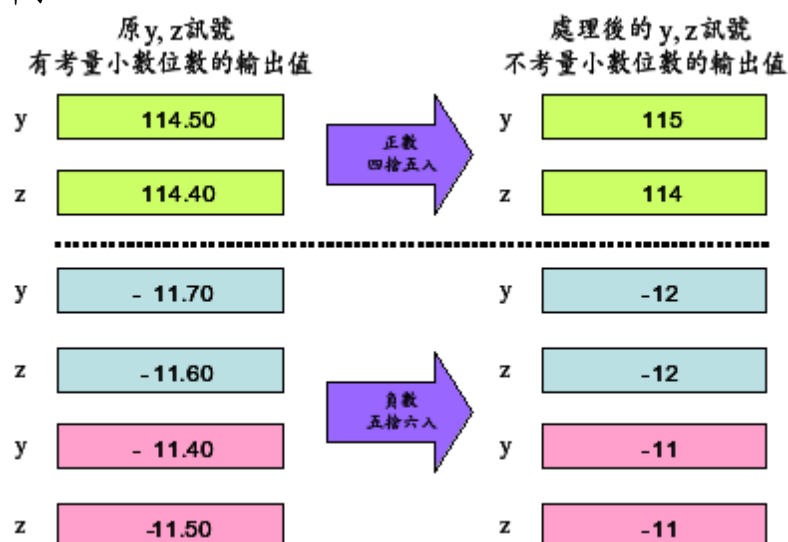


圖六、MBF 輸出筆數與其值之範例

另外，當 y 與 z 訊號輸出有小數位數的部分，處理法則規定如下:

1. 若為正數，採用四捨五入法取到整數。
2. 若為負數，採用五捨六入法取到整數。

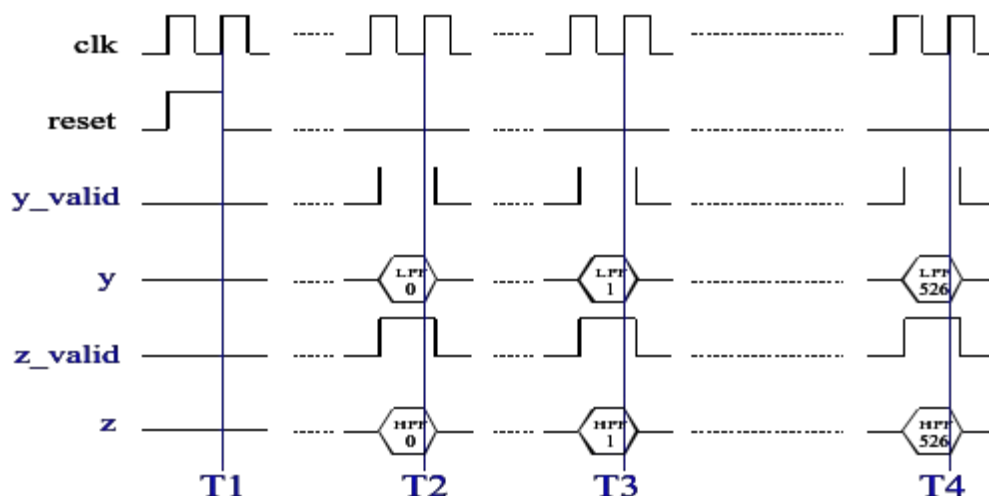
範例如圖七所示。



圖七、MBF 輸出之 y 、 z 訊號的小數位數處理方式之範例

2.4 電路時序規格

2.4.1 MBF 電路時序規格



圖八、MBF 電路時序圖

1. T1 時間點，reset 一個 Cycle 的時間，MBF 電路初始化結束，參賽者便要開始自行從 ROM 讀取輸入訊號 **x** 的值，**注意，ROM 每次讀取只有 4bits 資料，一筆完整 8bits 資料要分兩次(即兩個 Clock 週期)讀取。**
2. T2 時間點，經過 LPF 或 HPF 電路計算後，欲將其運算結果輸出，請將 **y_valid** 拉為 High 以通知 Host 端有一筆 LPF 計算後的值要輸出，或將 **z_valid** 拉為 High 已通知 Host 端有一筆 HPF 計算後的值要輸出。**注意：LPF 與 HPF 輸出可以不必同時輸出，兩者關係為獨立的。**
3. T3 時間點，動作與前述相同，進行第二筆的資料輸出。
4. T4 時間點，動作與前述相同，進行最後一筆的資料輸出。當 LPF 與 HPF 都輸出至最後一筆後，模擬立即結束。

2.4.2 ROM 記憶體規格與時序規格

製作 MBF 電路，由於所有輸入訊號 **x** 已被放置在 ROM 裡，**參賽者必須將 ROM 擺置於晶片內部，並於撰寫 RTL Code 時，呼叫此塊記憶體進來，將其一起做合成與 APR，以完成整個設計。**

有關 ROM 1024x4bits 記憶體細節規格與記憶體之時序圖，詳如記憶體附件中。

3. 評分標準

評分方式會依設計完成程度，分成 A、B、C 三種等級，排名順序為 A>B>C，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，**請參賽者提供一組正確的週期時間**

(CYCLE TIME)給評分人員驗證本電路之正確性。

◇ 評分項目一：依”模擬時間”(Time)長短評分

各參賽隊伍將 APR 完成後，執行 Gate-level Post-layout Simulation 模擬完後，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 = 10556NS** 做評分。

```
-----
Congratulations! All data have been generated successfully!
-----PASS-----
Simulation complete via $finish(1) at time 10556 NS + 0
```

◇ 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. IC Compiler Report Area 範例:

```
icc_shell> get_attribute [get_die_area] bbox
{0.000 0.000} {300.00 300.11}
=> Area = 300.00 x 300.11 = 90033 um2
```

2. SOC Encounter Report Area 範例:

```
encounter > analyzeFloorplan
***** Analyze Floorplan *****
Die Area(um^2)      : 93417.33
Core Area(um^2)     : 87289.16
Number of instance(s) : 4786
Number of Macro(s)   : 1
Number of IO Pin(s)  : 20
Number of Power Domain(s) : 0
***** Estimation Results *****
*****

=> Area = 93417.33 um2
```

設計完成程度三種等級，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 Golden Pattern 比對完全正確。
- b、完成 Synthesis，且 Gate-Level Pre-layout Simulation 結果正確。
- c、完成 APR，並達成 APR 必要項目，Gate-Level Post-layout Simulation 結果正確。

註：完成 APR 之必要項目

- 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**，不必做 interleaving。
- iii. 不要加 Dummy Metal。
- iv. 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，水平方向無任何限制)

- v. Power Rail 務必要加。
- vi. Core Filler 務必要加。
- vii. 內建的記憶體 ROM，其 VDD、VSS Pin 務必要連接至 Core Power Ring。
- viii. APR 後之 GDSII 檔案務必產生。
- ix. 完成 APR，DRC/LVS 完全無誤。

等級 A 之評分方法：

$$\text{Score} = \text{Time} \times \text{Area}$$

例如：

在前一頁範例中， $\text{Score} = \text{Time} \times \text{Area} = 10556 \times 90033 = 950388348$

註: Score 越小者，同級名次越好!

- ◇ **等級 B**：達成等級 A 之要求，但”APR 必要項目”有部分不符合，DRC/LVS 錯誤總數量容許 5 個(含)以下

此等級之成績計算方式如下：

$$\text{Score} = \text{Time} \times \text{Area} \times (\text{DRC} + \text{LVS 的錯誤總數量})$$

註: Score 越小者，同級名次越好!

- ◇ **等級 C**：達成等級 A 之要求，DRC/LVS 錯誤總數量超過 5 個以上，或是未完成 APR
- 此等級之成績計算方式如下：

$$\text{Score} = \text{Time} \times \text{Area}$$

註:

1. Score 越小者，同級名次越好!
2. 等級 C，視 APR 為 Fail，Area 以 Design Compiler 所 Report 的 Cell Area 為主。
3. 等級 C，視 APR 為 Fail，Time 以 Gate-level Pre-layout Simulation 為主。

附錄

附錄 A 為主辦單位所提供各參賽者的設計檔案說明；附錄 B 為主辦單位提供的測試樣本說明；附錄 C 為設計驗證說明；附錄 D 為評分用檔案，亦即參賽者必須繳交的檔案資料；附錄 E 則為設計檔案壓縮整理步驟說明；附錄 F 中說明本次競賽之軟體環境；附錄 G 中說明

本次競賽使用之設計資料庫。

附錄 A 設計檔(For Verilog)

1. 下表為主辦單位所提供各參賽者的設計檔

檔名	說明
MBF.v	參賽者所使用的設計檔，已包含系統輸/出入埠之宣告
testfixture1.v testfixture2.v	兩個 Test Bench 檔案。每個 Test Bench 已自動加入對應的 golden 比對檔案。
rom_1024x4_t13_verilog1.rcf rom_1024x4_t13_verilog2.rcf	兩組測試樣本 ROM Data ，每組提供 512 筆輸入訊號 x 於 ROM 裡。註：兩個 rcf 檔案，是存放 rom 的 Data，參賽者可以參考 LSB+MSB 沒有分開擺的版本 pattern1.dat 、 pattern2.dat 以方便模擬與除錯用。
LPF_golden1.dat HPF_golden1.dat LPF_golden2.dat HPF_golden2.dat	兩組測試樣本的 Golden Pattern 。每一個樣本提供 LPF 及 HPF 運算後的 golden pattern，各 527 筆資料需要作比對。註：兩組 Golden Pattern 皆以十六進制表示。
.synopsys_dc.setup	使用 Design Compiler (DC) 作合成或 IC Compiler Layout 之初始化設定檔。參賽者請依 Library 實際擺放位置，自行填上 Search Path 的設定。注意：無論合成或 APR，只需使用 worst case library ；
MBF_DC.sdc	Design Compiler(DC)作合成之 Constraint 檔案。參賽者可依需求自行修改部分設定，（見附錄 A 說明）。
MBF_APR.sdc	SOC Encounter(SOCE)或 IC Compiler(ICC)作 Layout 之 Constraint 檔案。參賽者可依需求自行修改部分設定（見附錄 A 說明）。
rom_1024x4_t13_sim1.v rom_1024x4_t13_sim2.v	ROM Verilog simulation model。rom_1024x4_t13_sim1.v 會自動 Include 第一組 Pattern（rom_1024x4_t13_verilog1.rcf）。rom_1024x4_t13_sim2.v 會自動 Include 第二組 Pattern（rom_1024x4_t13_verilog2.rcf）
rom_1024x4_t13_slow_syn.db	ROM timing library (For DC and ICC) 注意：無論 DC 合成或用 ICC 做 APR，只需使用 worst case library ，例如： slow.db 、 rom_1024x4_t13_slow_syn.db 。
rom_1024x4_t13_slow_syn.lib	ROM timing library(For SOCE)。 注意：用 SOCE 做 APR，只需使用 worst case library 例如： slow.db 、 rom_1024x4_t13_slow_syn.db 。
rom_1024x4_t13/	ROM frame view，ICC APR 使用
rom_1024x4_t13.vclef	ROM lef，SOCE APR 使用
rom_1024x4_t13.gds	ROM gds，作為 Stream Out GDSII 之用。

表
4、
設
計
檔
案
說
明

請使用 **MBF.v**，進行 MBF 電路之設計。其模組名稱、輸出/入埠宣告如下所示：

```
module MBF(clk, reset, y_valid, z_valid, y, z);
input  clk;
input  reset;
output y_valid;
output z_valid;
output [7:0] y;
output [7:0] z;
endmodule
```

2. 主辦單位提供兩個 Test Bench 檔案 **testfixture1.v**, **testfixture2.v** 分別對應到 **LPF_golden1.dat**、**HPF_golden1.dat** 及 **LPF_golden2.dat**、**HPF_golden2.dat**，這些都已加入了，參賽者只要注意這些檔案的路徑即可。

例如：

第一個 Test Bench 模擬，使用 **testfixture1.v**：

```
`define EXP1    "./LPF_golden1.dat"
`define EXP2    "./HPF_golden1.dat"
```

註：參賽者無須作修改，只需注意 **LPF_golden1.dat** 與 **HPF_golden1.dat** 的檔案位置即可，預設路徑為”目前目錄”。

3. 主辦單位提供兩個 ROM 的模擬檔案 **rom_1024x4_t13_sim1.v**、**rom_1024x4_t13_sim2.v**，會自動呼叫 ROM 的 Data(即輸入訊號 **x** 的來源)，參賽者只要注意這些檔案的路徑即可。

例如：

第一個 ROM 的模擬檔案，使用 **rom_1024x4_t13_sim1.v**，在該檔案的第 126 行可看到：

```
initial
$readmemb("rom_1024x4_t13_verilog1.rcf", mem );
```

註：參賽者無須作修改，只需注意 **rom_1024x4_t13_verilog1.rcf** 的檔案位置即可，預設路徑為”目前目錄”。

4. 主辦單位所提供的兩個 Test Bench 檔案，多加敘述如下：

```
`define End_CYCLE 10000000
`define SDFFILE    "./MBF_syn.sdf"
`ifdef SDF
    initial $sdf_annotate(`SDFFILE, MBF);
`endif
```

註：

1. **End_CYCLE** 預設 1000 萬個 Cycles，其目的可以防止參賽者因電路有錯，模擬陷入無窮

回圈之境，倘若參賽者確定模擬需超過 1000 萬個 Cycles 以上，可自行再加大此 Cycle 數。

2. SDF 檔案，請自行修改 SDF 實際檔名後模擬。
3. 在 Test Bench 中，主辦單位提供`ifdef SDF 的描述，其目的是讓本 Test Bench 可以作為 RTL 模擬、合成後模擬與 Layout 後模擬使用。注意：當參賽者在合成或 Layout 後模擬，請務必多加一個參數” +define+SDF”，方可順利模擬。

例如：當合成後，使用 NC-Verilog 模擬第一組樣本，在 UNIX 下執行下面指令

```
> ncverilog testfixture1.v MBF_syn.v rom_1024x4_t13_sim1.v -v tsmc13_neg.v  
+define+SDF +access+rw
```

註：模擬 rom_1024x4_t13_sim1.v 時，需要注意 rom_1024x4_t13_verilog1.rcf 檔案的路徑。

```
ncsim> run  
Novas FSDb Dumper for IUS, Release 2010.10 (Linux x86_64/64bit) 10/01/2010  
Copyright (C) 1996 - 2010 by SpringSoft, Inc.  
*Novas* : Create FSDb file 'MBF1.fsdb'  
*Novas* : Begin traversing the scopes, layer (0).  
*Novas* : End of traversing.  
-----  
Congratulations! All data have been generated successfully!  
-----PASS-----  
  
Simulation complete via $finish(1) at time 10556 NS + 0  
../testfixture.v:130      #(`CYCLE/2); $finish;  
ncsim> exit
```

5. 主辦單位已提供合成及 APR 的 SDC 檔(MBF_DC.sdc, MBF_APR.sdc)，檔案內關於環境設定部分不可更改，其它部分可根據參賽者設計需求進行增減或修改，再行使用 SDC 檔案。

例如：MBF_DC.sdc 檔案內容如下：

- A. 需要自行修改 clock constraints 及依照自己電路設計需求，增加額外的 Constraints 設定：

```
#You may modified the clock constraints  
#or add more constraints for your design  
#####  
##  
set cycle 10  
create_clock -period $cycle [get_ports clk]  
...  
...  
#####  
##
```

B. 下述幾行的 Constrains 為主辦單位規定的基本的環境設定，參賽者不可以作任何更改：

```
#The following are design spec. for synthesis
#You can NOT modify this section
#####
#####
set_clock_uncertainty 0.1 [all_clocks]
set_clock_latency 0.5 [all_clocks]
set_input_delay 1 -clock clk [remove_from_collection [all_inputs]
[get_ports clk]]
set_output_delay 1 -clock clk [all_outputs]
set_load 1 [all_outputs]
set_drive 1 [all_inputs]

set_operating_conditions -max_library slow -max slow
set_wire_load_model -name tsmc13_wl10 -library slow
#####
#####
```

附錄 B 測試樣本

主辦單位提供兩組測試樣本，爲了讓參賽者看完題目後，更能明確題意，主辦單位在此以 Pattern1 之測試樣本爲例，如圖四(a)所示，爲第一組 Pattern，其實際擺放在 ROM 的方式爲 Address 0 擺放第一筆資料的 LSB 4bits，Address 1 擺放第一筆資料的 MSB 4bits，其餘資料依此類推，如圖四(b)所示。經過 LPF 與 HPF 運算後，前 8 組的輸出結果如表五所示，提供參賽者模擬與除錯使用。參賽者也可以直接參考 LPF_golden1.dat、HPF_golden1.dat 及 LPF_golden2.dat、HPF_golden2.dat 的值。

表五、第一組 Pattern LPF&HPF 前八筆計算結果

LPF y(n)		HPF z(n)	
n	y(n)	n	z(n)
0	00	0	00
1	00	1	00
2	01	2	01
3	02	3	FD
4	FA	4	FB
5	F8	5	0C
6	18	6	11
7	35	7	BD

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。

注意：每組限定只能使用 1 license, 勿使用 Multi-CPU。

- RTL 與 Gate-Level 階段：參賽者必須進行 RTL simulation 及 Gate-Level simulation，模擬結果必須於參賽者提供之 CYCLE 數值下，功能完全正確。
- Physical 階段，包含三項驗證重點：
 1. 依主辦單位各項要求，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 檔與 sdf 檔完成 post-layout gate-level simulation**，以下分爲 IC Compiler、SOC Encounter 兩種軟體說明 netlist 與 sdf 寫出步驟。

- i. 使用 Synopsys IC Compiler 者，執行步驟如下：
在 IC Compiler 主視窗底下點選

“File > Export > Write SDF...”

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

按 。

對應指令：write_sdf -version 2.1 MBF_pr.sdf

“File > Export > Write Verilog...”

先按

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

按 。

- ii. 使用 Cadence SOC Encounter 者，執行步驟如下：
在 SOC Encounter 視窗下點選：

“Design ▾ Save ▾ Netlist...”

Netlist File	MBF_pr.v
--------------	----------

All other options	Default value
-------------------	---------------

按 。

“Timing ▢ Extract RC...” ，按 。

“ Timing ▢ Calculate Delay...”

存成 MBF_pr.sdf，按 。

3. 完成 DRC 與 LVS 驗證：參賽者必須以其所使用之 **P&R 軟體內含之 DRC 與 LVS 驗證功能完成 DRC 與 LVS 驗證**，以下分為 IC Compiler、SOC Encounter 兩種軟體說明執行步驟。

- i. 使用 Synopsys IC Compiler 者，驗證 DRC 與 LVS 步驟如下：
在 IC Compiler Layout 視窗底下點選

“Route > Verification > DRC ...”

Read child cell from	Cell view
All other options	Default value

按 。

將跳出 Error Browser 視窗，請參賽者自行查看是否有錯，若有請自行修改 Layout 到 0 個 Violation 為止。

“Route > Verification > LVS ...”

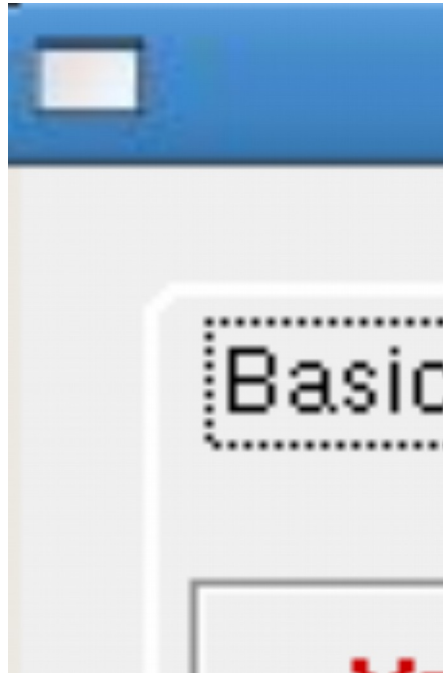
Pins not connected to a wire segment(Floating port)	disable
All other options	Default value

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。

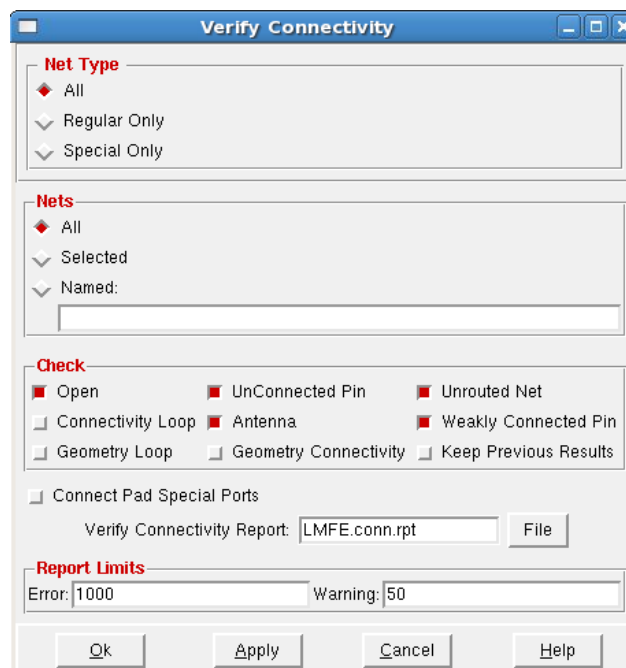
- ii. 使用 Cadence SOC Encounter 者，驗證 DRC 與 LVS 步驟如下：
在 SOC Encounter 視窗下點選

1. DRC 驗證：請選 “Verify ▢ Verify Geometry...” Default 值，按 。



註: 若 DRC 有發生錯誤，請選 “Verify ▢ Violation Browser...” 查明原因。

2. LVS 驗證：請選 “Verify ▢ Verify Connectivity...” Default 值，按 **OK**。



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

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 IC Compiler** 者，請記得將整

個 Milkyway Library 等相關的 design database，壓縮成一個檔案。使用 Cadence SOC Encounter 者，請將 SOC Encounter 相關的 design database（包含 .enc 檔案與 .enc.dat 目錄），壓縮成一個檔案。壓縮的檔案格式如下：假設參賽者的 design database 目錄名稱爲” your_lib”，請執行底下的 UNIX 指令，最後可以得到” your_name.tar”的檔案。

> ***tar cvf your_name.tar your_lib***

在執行以上的指令之前，請確定將你使用的 P&R Tool 儲存後關閉，再執行上述的指令，否則在壓縮的過程會出現錯誤。

表 6

<i>RTL category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
N/A	N/A	Design Report Form
RTL Simulation	*.v or *.vhd	Verilog (or VHDL) synthesizable RTL code
<i>Gate-Level category</i>		
<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	Pre-layout gate-level sdf
<i>Physical category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
P&R	*.tar	archive of the design database directory
	*.gds	GDSII layout
	DRC/LVS report	不用儲存 DRC/LVS Report 檔案!只需在 Design Report Form 上填寫 DRC/LVS 錯誤總數量即可。(目標要做到 0 個錯誤!)
Post-layout Gate-level Simulation	*_pr.v	Verilog gate-level netlist generated by Cadence SOC Encounter or Synopsys IC Compiler
	*_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 6 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 MBF.v ~/result/
```

```
> cp MBF_syn.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. 主辦單位已將所有軟體環境設定於：/usr/cad/cshrc/env.cshrc，參賽同學不需再做任何設定。
2. 在 env.cshrc 所設定好的軟體環境包括：

Vendor	Tool	Executable
Cadence	Virtuoso	icfb
	Composer	icfb
	NC-Verilog	ncverilog
	SOC Encounter	encounter
Synopsys	design vision	dv, dc_shell
	VCS	vcs
	IC compiler	icc_shell -gui
	Hspice	hspice
	Cosmos Scope	scope
	Spice explorer	sx -w , wv
Mentor	Calibre	calibre
	ModelSim	vsim
Spring Soft	Laker	laker
	Verdi	verdi, nWave, nLint
Utility	vi	vi, vim, gvim
	gedit	gedit
	nedit	nedit
	pdf reader	acroread
	calculate	gnome-calculator, bc -l
	gcc	gcc

※ gnome-calculator (工程計算機執行檔，可開啓 View -> Scientific mode)

EDA 軟體所須使用的 license 皆已設定完成，不須額外設定，且每隊限定**每個 EDA 軟體只能使用一套 license**。

附錄 G 設計資料庫

設計資料庫位置： /usr/cad/icc2012/CBDK_IC_Contest_v2.1

目錄架構

ICC/	
tsmc13gfsf_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
SOCE/	
lef/	
tsmc13fsg_8lm_cic.lef	LEF for core cell
antenna_8.lef	LEF for antenna
lib/	
slow.lib	worst case for core cell
streamOut.map	Layout map for GDSII out
SynopsysDC/	
db/	
slow.db	Synthesis model (slow)
lib/	
slow.lib	timing and power model
Verilog/	
tsmc13_neg.v	Verilog simulation model
VHDL/	
tsmc13.vhd	VHDL simulation model
Memory/	
rom_1024x4_t13/	
rom_1024x4_t13_slow_syn.db	DC ICC APR 可使用(worst-case)
rom_1024x4_t13_slow_syn.lib	SOCE APR 可使用(worst-case)
rom_1024x4_t13_sim1.v	Verilog simulation model for sim1
rom_1024x4_t13_verilog1.rcf	ROM data of sim1
rom_1024x4_t13_sim2.v	Verilog simulation model for sim2
rom_1024x4_t13_verilog2.rcf	ROM data of sim2
rom_1024x4_t13.vclef	SOCE APR 可使用
rom_1024x4_t13_ant.lef	SOCE 可使用(LEF for antenna)
rom_1024x4_t13/	ICC 可使用(ROM Fram View)
rom_1024x4_t13.pdf	ROM Spec Document
rom_1024x4_t13.gds	ROM GDSII file

Design Report Form

登入帳號(login-id)			
RTL category			
<i>Design Stage</i>	<i>Description</i>	<i>File Name</i>	
RTL Simulation	使用之 HDL 名稱 (請填入 Verilog 或 VHDL)		
RTL Simulation	RTL 檔案名稱 (RTL Netlist file name)		
Gate-Level category			
<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 CYCLE Time (請確定模擬功能正確)	() ns	
Physical category			
<i>Design Stage</i>	<i>Description</i>	<i>File Name or Value</i>	
P&R	使用之 P&R Tool (請填入 IC compiler 或 SOC Encounter)		
	設計資料庫檔案名稱 (Library name) (ICC: Milkyway Library Name, SOCE: xxx.enc.dat)		
	DRC 錯誤總數量 (ex: 0 個)		
	LVS 錯誤總數量 (ex: 0 個)		
	佈局檔檔案名稱 (GDSII file name)		
	佈局面積 (Layout Area)	() um X () um	
Post-layout Gate-level Simulation	Gate-Level 檔案名稱 (Gate-Level Netlist file name)		
	Post-layout sdf 檔案名稱		
	Post-layout Simulation 所使用的 CYCLE Time (請確定模擬功能正確) Ex: 10ns		
	Post-layout Simulation Time (Simulation Time, ex: 10556 ns) Time = ?		
Over All	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			

