

B 組- 大學/研究所 標準元件數位電路設計

初賽時間為3月21日(三) 8:30~20:30，初賽當日請密切注意競賽網頁公告及 Facebook “IC競賽專頁”(<http://www.facebook.com/IcContest20122014ByNcu>)，所有最新訊息將做即時動態更新。

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。

參賽作品在今天20:30前務必根據初賽前寄發之ftp 帳號密碼上傳至國家晶片中心之相關ftp site。

初賽登錄：請將您的隊伍參賽資料e-mail至 b.icdesign.ncu@gmail.com

信件主旨為：

IC設計競賽登錄(報名ID: B?????)

(請填上自己的報名ID)

信件內容：

參賽組別：B組

報名ID: B????? (例: B00001)

參賽姓名：李大華、王小明

2012 IC Design Contest Preliminary

標準元件數位電路設計 – 大學/研究所組

NAND Flash Memory Controller

1. 問題描述

請完成一快閃記憶體控制(NAND Flash Memory Control)電路設計。此控制電路的功能是將快閃記憶體 A 的資料完全複製至快閃記憶體 B。

本控制電路有 4 只信號輸入(clk, rst, F_RB_A, F_RB_B)、9 只信號輸出(done, F_CLE_A, F_ALE_A, F_REN_A, F_WEN_A, F_RB_A, F_CLE_B, F_ALE_B, F_REN_B, F_WEN_B)及 2 只記憶體雙向輸出入信號(F_IO_A, F_IO_B)，如圖一所示，關於各輸入/輸出信號的功能說明，請參考表一。

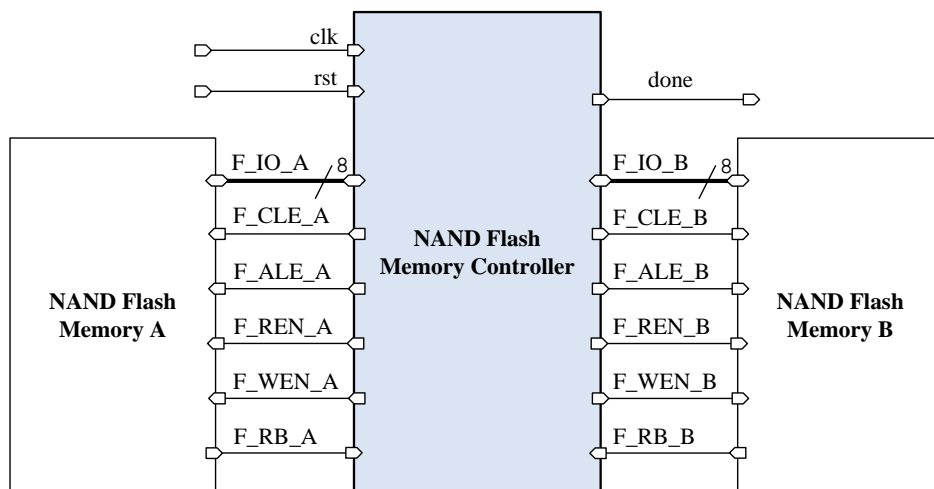
本試題有使用到快閃記憶體模型(flash simulation model)，其中內含時序檢查，若要避免 RTL 模擬時所產生的時序檢查錯誤(setup or hold violation)，可參考附錄 B 的第 4 點來進行模擬。

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 輸入/輸出介面

表 一、輸入/輸出信號

信號名稱	輸入/輸出	位元寬度	說明
<i>clk</i>	input	1	時脈信號。 說明：控制器與內部記憶體於時脈正緣時動作。
<i>rst</i>	input	1	高位準非同步(active high asynchronous)之系統重置信號。 說明：此信號於系統啟動時送出。
<i>done</i>	output	1	完成指令輸出信號。 說明：當控制器完成所有工作時，將 done 設為 high 表示完成。
<i>F_IO_A</i>	inout	8	快閃記憶體 A 八位元雙向輸出入埠。 說明：此輸出入埠可做為對快閃記憶體 A 指令輸入、位址輸入及資料讀寫。在本題中，F_IO_A 在資料讀寫部分，僅做為資料讀取使用。
<i>F_CLE_A</i>	output	1	快閃記憶體 A 指令致能信號。 說明：當此信號為 high 時，F_IO_A 會被視為「指令」，且快閃記憶體 A 於 F_WEN_A 為正緣時讀取指令。
<i>F_ALE_A</i>	output	1	快閃記憶體 A 位址致能信號。 說明：當此信號為 high 時，F_IO_A 會被視為「位址」，且快閃記憶體 A 於 F_WEN_A 為正緣時讀取位址。
<i>F_REN_A</i>	output	1	快閃記憶體 A 連續讀取致能信號。 說明：在進行資料的連續讀取時，資料會在此信號負緣後出現於 F_IO_A，同時自動將快閃記憶體 A 內的位址加 1。(如圖九)
<i>F_WEN_A</i>	output	1	快閃記憶體 A 寫入致能信號。 說明：當 F_IO_A 為指令、位址輸入及資料寫入時，由此信號的正緣來對快閃記憶體 A 進行寫入(如圖六、七、八)。
<i>F_RB_A</i>	input	1	快閃記憶體 A 完成/忙碌信號。 說明：當此信號為 low 表示快閃記憶體 A 正在忙碌，要等待此信號為 high 時才能進行動作。
<i>F_IO_B</i>	inout	8	快閃記憶體 B 八位元雙向輸出入埠。 說明：此輸出入埠可做為對快閃記憶體 B 指令輸入、位址輸入及資料讀寫。在本題中，F_IO_B 在資料讀寫部分，僅做為資料寫入使用。
<i>F_CLE_B</i>	output	1	快閃記憶體 B 指令致能信號。 說明：當此信號為 high 時，F_IO_B 會被視為「指令」，且快閃記憶體 B 於 F_WEN_B 為正緣時讀取指令。(如圖九)
<i>F_ALE_B</i>	output	1	快閃記憶體 B 位址致能信號。 說明：當此信號為 high 時，F_IO_B 會被視為「位址」，且快閃記

			憶體 B 於 F_WEN_B 為正緣時讀取位址。
<i>F_REN_B</i>	output	1	快閃記憶體 B 連續讀取致能信號。 說明：在進行資料的連續讀取時，資料會在此信號 負 緣後出現於 F_IO_B，同時自動將快閃記憶體 B 內的位址加 1。（如圖九）
<i>F_WEN_B</i>	output	1	快閃記憶體 B 寫入致能信號。 說明：當 F_IO_B 為指令輸入、位址輸入及資料寫入時，由此信號的正緣來對快閃記憶體 B 進行寫入。
<i>F_RB_B</i>	input	1	快閃記憶體 B 完成/忙碌信號。 說明：當此信號為 low 表示快閃記憶體 B 正在忙碌，要等待此信號為 high 時才能進行動作。

2.3 系統功能描述

快閃記憶體控制電路來對快閃記憶體進行讀寫功能。

系統啟動時將對快閃記憶體 A 讀取資料，並將快閃記憶體 A 內的所有資料全部寫入至快閃記憶體 B 相同位址之中，如圖 二所示。快閃記憶體 A 與快閃記憶體 B 的規格及大小完全相同。

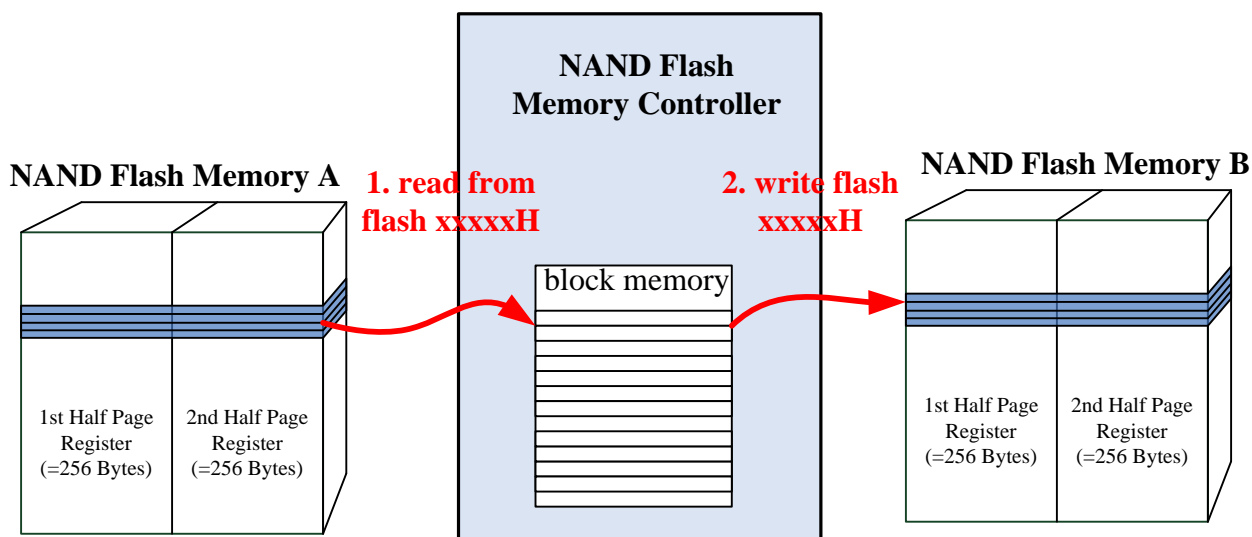


圖 二、快閃記憶體讀取寫入之範例

2.4 時序規格圖

快閃記憶體控制電路與主控端之間的時序波形如圖 三所示，其中的時脈週期 t_{CYCLE} 預設為 20ns。當系統啟動時，即開始將快閃記憶體A內的所有資料搬動至快閃記憶體B，搬動結束時並將 done 信號設定為high，並立即結束動作。

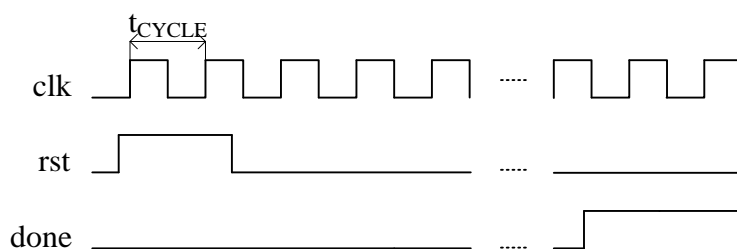


圖 三、系統時序圖

2.5 快閃記憶體(NAND Flash Memory)功能描述

本題目中所使用的快閃記憶體容量為 256K 位元組(byte)，其具有 512 個列(row)或頁(page)，且每頁有 512 個位元組。快閃記憶體陣列如圖 四所示，此快閃記憶體共分為 512 頁(page)，且每頁又可分為前半頁(1st Half Page Register)及後半頁(2nd Half Page Register)。快閃記憶體之特性有下面幾點：

1. 進行讀取或寫入時是以頁為單位，只要給予起始位址，便可開始進行連續的讀取或寫入，最多可讀寫一整個頁(512 byte)。換言之，只要在同一頁內就可被連續地讀寫，但無法連續至下一頁。如圖 十、十一所示，為連續讀寫十筆之範例。
2. 若連續寫入長度跨越兩個頁，則跨頁時須重新給予讀寫指令。
3. 如果從前半頁開始連續讀取或寫入，後半頁可不必重送指令。

此快閃記憶體的位址是由 8 位元的 F_IO 來多次傳輸，這種方式不僅可減少快閃記憶體的腳位數且無論容量多寡也不須增減腳位數。此記憶體共有 256K 位元組需要 18 條位址，因此使用 F_IO 需要 3 個週期來分別輸入欄位址(Column address)與列位址(Row address)。在一般的讀寫時，需要使用到 3 個週期的位址且置於指令之後。指令、位址及資料要經由 F_IO 來輸入時，F_WEN 必須為”Low”，而資料將會在 F_WEN 為正緣時被讀取。當 F_CLE 為 High 時，F_IO 會被視為指令輸入；當 F_ALE 為 High 時，F_IO 會被視為位址輸入。

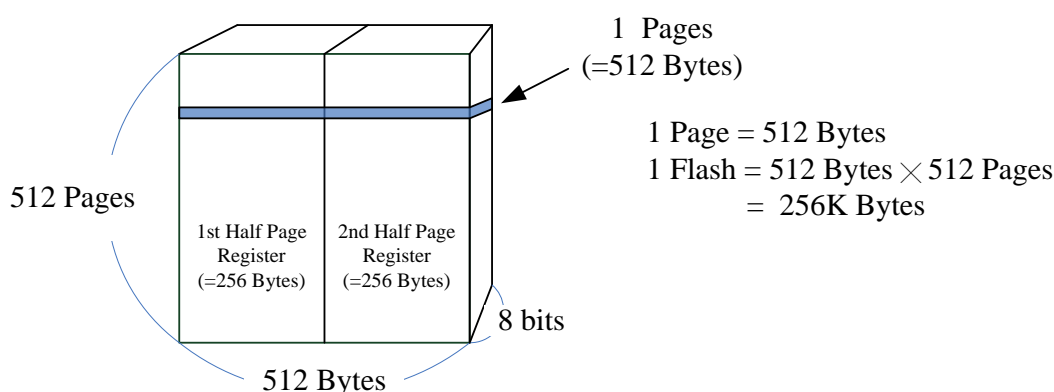


圖 四、快閃記憶體之陣列結構

	F_IO[7]	F_IO[6]	F_IO[5]	F_IO[4]	F_IO[3]	F_IO[2]	F_IO[1]	F_IO[0]	
1 st cycle	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	欄位址
2 nd cycle	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	列位址(頁位址)
3 rd cycle	L	L	L	L	L	L	L	A ₁₇	

註：1. A₈ 是經由指令 00H(A₈ = 0)或 01H(A₈ = 1)來指定為”Low”或”High”。

指令 00h (Read)：定義起始位址在前半頁(1st Half Page Register)

指令 01h (Read)：定義起始位址在後半頁(2nd Half Page Register)

2. “L”是指要設定為”Low”。

3. 當輸入位址的週期數比所需位址的還多時，快閃記憶體會忽略多餘的位址。

快閃記憶體的動作是經由將指令寫入其內部的暫存器來進行之，我們將此快閃記憶體之指令整理如下表所示：

表 二、快閃記憶體之指令集

功能	指令 1	指令 2
讀取(Read)	00h/01h	
寫入(Write)	80h	10h

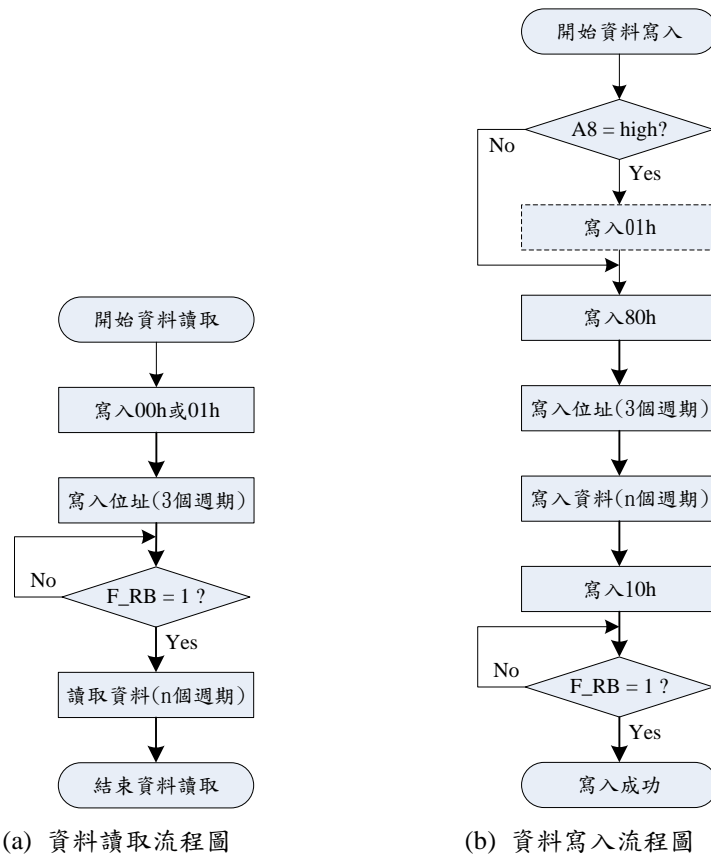


圖 五、快閃記憶體動作流程圖

快閃記憶體讀寫動作如上方的流程圖所示，須配合表二的指令再加上指定位址來進行之。資料讀取的動作較為簡單，只要判斷 A8 來決定指令是要使用 00h 或是 01h，再加上欲讀取之起始位址後，便可進行資料的連續讀取。資料寫入的指令一般是以 80h 為起始，但若是想直接對 A8 為 high 的位址做寫入，則可在 80h 前加入 01h 的指令，經由一連串資料寫入，在寫入 10h 的指令後，資料才會由暫存器寫入快閃記憶體之中，隨即便會進入忙碌狀態，等待 F_RB 為 high 時表示完成寫入動作。

2.5.1 快閃記憶體之時序規格

以下將快閃記憶體先細分成四個週期來表示時序，分別是圖 六至圖 九的指令週期、位址週期、資料輸入週期及資料連續讀取週期。接著，圖 十至圖 十一則是以波形來說明資料讀取、資料寫入等動作。最後，時序的數值整理於表 三。

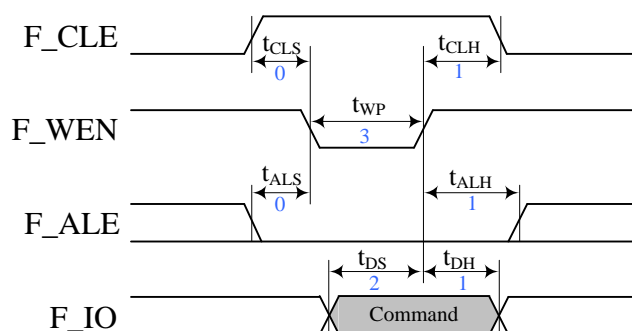


圖 六、指令週期

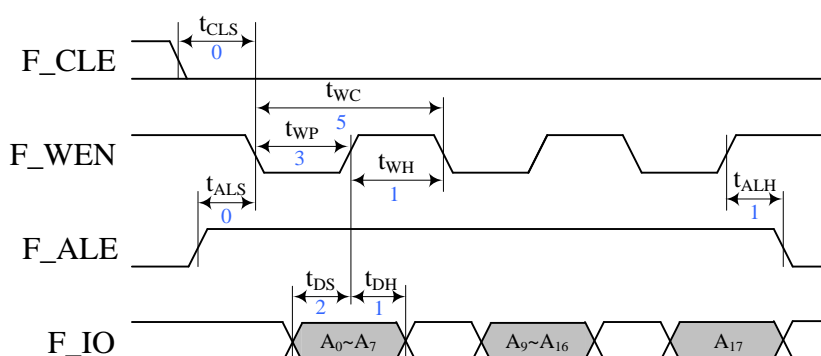


圖 七、位址週期

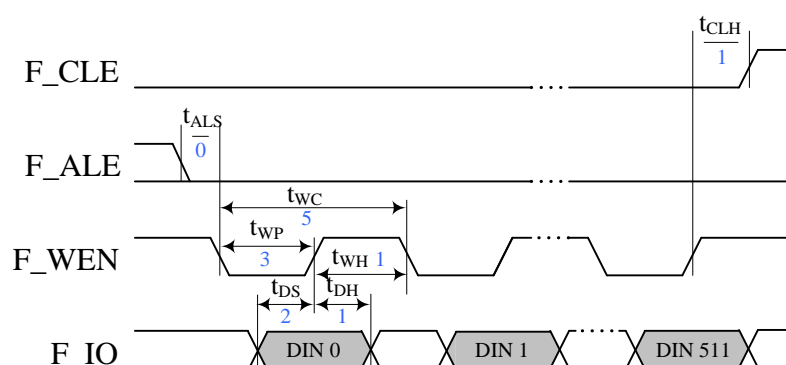


圖 八、資料輸入週期

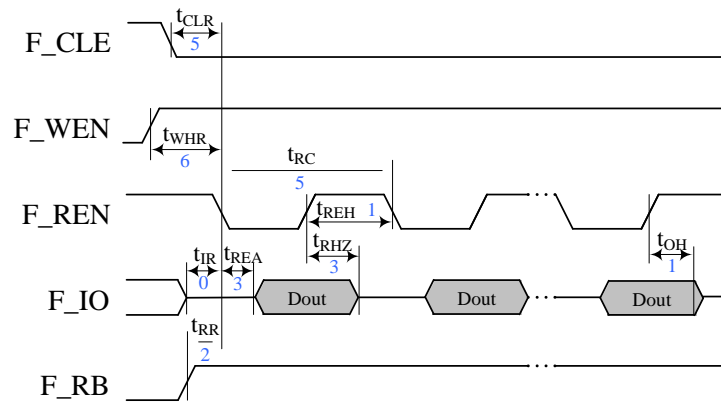


圖 九、資料連續讀取週期

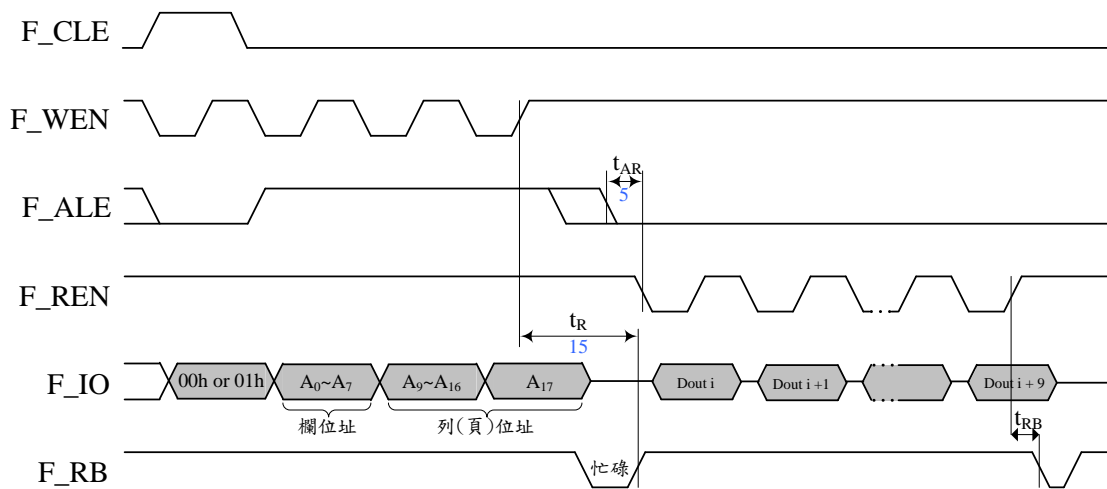


圖 十、資料連續讀取十筆之波形

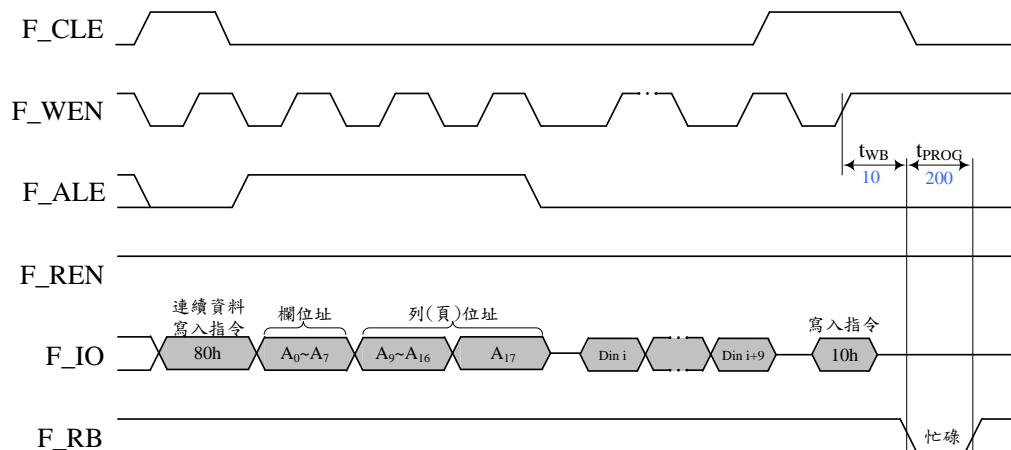


圖 十一、資料連續寫入十筆之波形

由於F_IO資料信號為雙向埠，因此在verilog coding時可能會需要使用到下列的語法：

```
assign output = (enable) ? input : 'bz;
```

表 三、快閃記憶體時序參數表

Symbol	Description	Min.	Max.	Unit
t _{PROG}	Program Time	-	200	ns
t _{BERS}	Block Erase Time	-	1	μs
t _{CLS}	F_CLE Setup Time	0	-	ns
t _{CLH}	F_CLE Hold Time	1	-	ns
t _{WP}	F_WEN Pulse Width	3	-	ns
t _{ALS}	F_ALE Setup Time	0	-	ns
t _{ALH}	F_ALE Hold Time	1	-	ns
t _{DS}	Data Setup Time	2	-	ns
t _{DH}	Data Hold Time	1	-	ns
t _{WC}	Write Cycle Time	5	-	ns
t _{WH}	F_WEN High Hold Time	1	-	ns
t _R	Data Transfer from Cell to Register	-	15	ns
t _{AR}	F_ALE to F_REN Delay	5	-	ns
t _{CLR}	F_CLE to F_REN Delay	5	-	ns
t _{RR}	Ready(F_RB=1) to F_REN Low	2	-	ns
t _{WB}	F_WEN High to Busy(F_RB=0)	-	10	ns
t _{RC}	Read Cycle Time	5	-	ns
t _{REA}	F_REN Access Time	-	3	ns
t _{RHZ}	F_REN High to Output Hi-Z	-	3	ns
t _{OH}	F_REN High to Output Hold Time	1	-	ns
t _{REH}	F_REN High Hold Time	1	-	ns
t _{IR}	Output Hi-Z to F_REN Low	0	-	ns
t _{RB}	Last F_REN High to Busy	-	10	ns
t _{WHR}	F_WEN High to F_REN Low	6	-	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 E)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，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 (p1 & p2)之 RTL 與 gate-level simulation
2. B 等級：完成測試樣本一至二 (p1 & p2) 之 RTL
3. C 等級：完成測試樣本一 (p1) 之 RTL

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。但倘若參賽者於競賽時間內無法完成設計，也請務必將設計檔案內容上傳，主辦單位亦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 為測試樣本詳細資料；附錄 D 為評分用檔案，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 四。驗證評分時，係以所列軟體及版本作為驗證依據。

表 四、設計軟體版本

Functionality	Corresponding EDA tools
Logic Simulator	nc-verilog v8.2 modelsim v10.1a vcs v2010.06-sp1
Logic Synthesizer	design-compiler v2010.03-sp5
Cell Library	Cell-based design kit for IC contest v2.1

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. 下表 五為主辦單位所提供各參賽者的設計檔案

表 五、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
NFC.v (NFC.vhd)	參賽者所使用的設計檔，已包含系統輸/出入埠之宣告
flash_a.v	快閃記憶體 A 模擬檔
flash_b.v	快閃記憶體 B 模擬檔
synopsys_dc.setup	Design Compiler 初始設定範例檔案
NFC.sdc	Design Compiler 電路合成規範檔
p1_mem_goal.dat	測試樣本一之記憶體比對檔
p2_mem_goal.dat	測試樣本二之記憶體比對檔
p1_mem.dat	測試樣本一輸入檔案
P2_mem.dat	測試樣本二輸入檔案
report.000	結果報告範本

2. 請使用 *NFC.v(vhd)*，進行快閃記憶體控制電路之設計。其模組名稱、輸出/入埠宣告如下所示：

```

module NFC(clk, rst, done, F_IO_A, F_CLE_A, F_ALE_A, F_REN_A, F_WEN_A, F_RB_A
    , F_IO_B, F_CLE_B, F_ALE_B, F_REN_B, F_WEN_B, F_RB_B);

    input clk;
    input rst;
    output done;
    inout [7:0] F_IO_A;
    output F_CLE_A;
    output F_ALE_A;
    output F_REN_A;
    output F_WEN_A;
    input F_RB_A;
    inout [7:0] F_IO_B;
    output F_CLE_B;
    output F_ALE_B;
    output F_REN_B;
    output F_WEN_B;
    input F_RB_B;
endmodule

```

3. 快閃記憶體為非同步信號，參賽者可自行修改電路合成規範檔(*NFC.sdc*)以符合自己的電路設計。以下為快閃記憶體時序設定之範例，參賽者可參考使用：

```
set_input_delay 4 -clock clk [get_ports F_IO*]  
  
set_output_delay 1 -clock clk [get_ports done]  
  
set_output_delay 2 -max -clock clk [get_ports F_IO*] -add_delay  
set_output_delay -1 -min -clock clk [get_ports F_IO*] -add_delay  
set_output_delay -1 -min -clock clk [get_ports F_CLE*]  
set_output_delay -1 -min -clock clk [get_ports F_ALE*]  
set_output_delay 2 -max -clock clk [get_ports F_REN*]  
set_output_delay 2 -max -clock clk [get_ports F_WEN*]  
  
set_min_delay 1 -from [get_ports F_RB*]
```

4. 比賽共提供兩組測試樣本，**參賽者可依下面範來進行模擬：**

- ncverilog 指令範例如下：
ncverilog testfixture.v NFC.v +define+p1
- 若使用 modelsim，則是在 compiler verilog 時，使用下面指令：
vlog testfixture.v +define+p1
- 上述指令中***+define+p1*** 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+p2***。
- 關於模擬時使用的快閃記憶體 (flash_a.v, flash_b.v)，因已經以 include 方式加在 testfixture.v 裏，所以不需加在模擬指令裏。
- 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***，但在 gate level 模擬時不可加入***+notimingchecks***

5. 因波形檔很大，請以 fsdb dump 取代 vcd dump，dump fsdb 須使用指令如下：

- ncverilog 指令範例如下：
ncverilog testfixture.v NFC.v +define+p1+FSDB +access+r
- modelsim 使用者，請直接使用內建波形來進行除錯。

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，將測試樣本之快閃記憶體 A 資料詳列如下：

◆ 測試樣本一

位址	資料	位址	資料	位址	資料
00	00	0D	0D	1A	1A
01	01	0E	0E	1B	1B
02	02	0F	0F	1C	1C
03	03	10	10	1D	1D
04	04	11	11	1E	1E
05	05	12	12	1F	1F
06	06	13	13	20	FF
07	07	14	14	21	FF
08	08	15	15	.	FF
09	09	16	16	.	FF
0A	0A	17	17	.	FF
0B	0B	18	18	.	FF
0C	0C	19	19	3FFFF	FF

◆ 測試樣本二

位址	資料	位址	資料	位址	資料
00	F8	0D	63	1A	98
01	DC	0E	38	1B	33
02	DE	0F	8B	1C	32
03	7C	10	78	1D	B9
04	FA	11	C2	1E	DC
05	EA	12	62	1F	75
06	E6	13	8A	20	85
07	D8	14	C7	21	47
08	94	15	B2	.	.
09	01	16	27	.	.
0A	8A	17	EC	.	.
0B	C9	18	B3	.	.
0C	80	19	CF	3FFFF	AB

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 十二所示。(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 六、評分用檔案

RTL category		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
N/A	report.xxx	design report
RTL Simulation	*.v or *.vhd	Verilog (or VHDL) synthesizable RTL code
Gate-Level category		
<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
	*_syn.ddc	design database generated by Synopsys Design Compiler

ftp 帳號(FTP number): 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): NFC.v, sub1.v, sub2.v

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

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

gate-level sdf filename: NFC_syn.sdf

design compiler 合成資料庫(dc library): NFC_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 十二、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**。

ftp 的帳號和密碼在 4 天前已用 email 寄給各參賽者。若有任何問題，請聯絡 CIC

台灣大學：icftp.ee.ntu.edu.tw (140.112.20.92)

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

南區晶片中心：icftp2.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。