

HOMEWORK 4

1. 問題描述

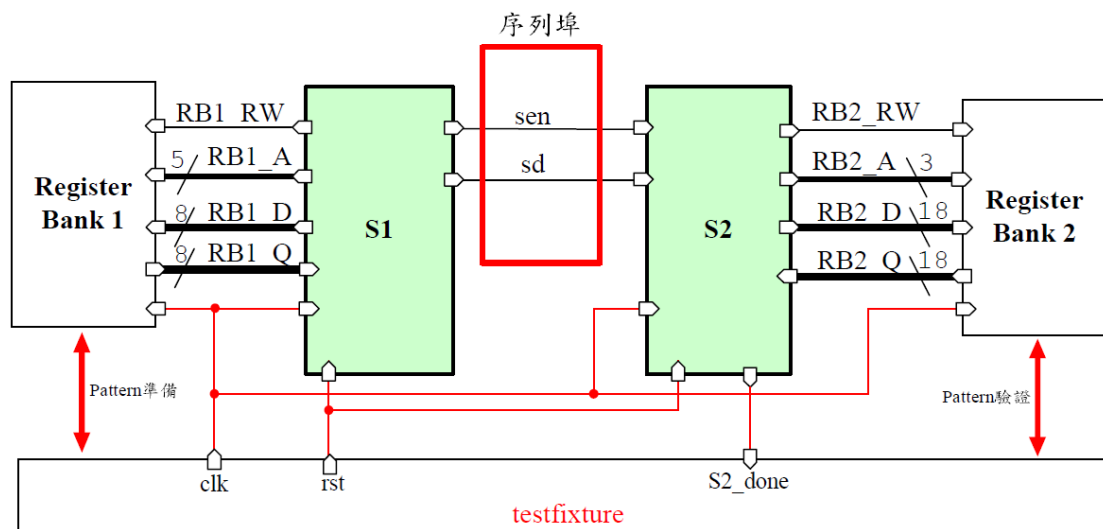
請完成一序列傳輸/接收電路(Serial Interface transmitter/receiver)(如圖一)。需要完成序列埠(sd)兩端 S1 及 S2 的電路設計。本序列傳輸/接收電路負責將記憶體 RB1(Register Bank1)的資料經由序列埠上傳到記憶體 RB2(Register Bank2)。

本傳輸/接收電路中，S1 有三只信號輸入(clk、rst、RB1_Q)、五只信號輸出(RB1_RW、RB1_A、RB1_D、sen、sd)。S2 有五只信號輸入(clk、rst、RB2_Q、sen、sd)、四只訊號輸出(S2_done、RB2_RW、RB2_A、RB2_D)。相關信號的說明，請參考表一和表二。

作業中有使用到記憶體模型(memory simulation model)，詳細的記憶體規格描述在 2.2.1。此記憶體模型(RB1、RB2)不需合成，僅需將 S1 與 S2 模組成。

2. 設計規格

2.1. 系統方塊圖



圖一、系統方塊圖

2.2. 輸出入訊號和記憶體描述

表一、S1 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	系統提供的時脈信號。
<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>RB1_RW</i>	output	1	Register Bank1 讀寫控制信號。 說明：當此信號為 high 時是對 Register Bank1 進行讀取(read)；low 則是進行對 Register Bank1 寫入(write)。
<i>RB1_A</i>	output	5	Register Bank1 五位元位址信號。
<i>RB1_D</i>	output	8	Register Bank1 八位元資料輸入埠。
<i>RB1_Q</i>	input	8	Register Bank1 八位元資料輸出埠。
<i>sen</i>	output	1	致能信號。 說明：當此信號為 low 時，sd 傳輸的資料被認為是有效的。
<i>sd</i>	output	1	序列傳輸中的一位元資料輸出埠。資料在 clk 負緣的時候觸發。

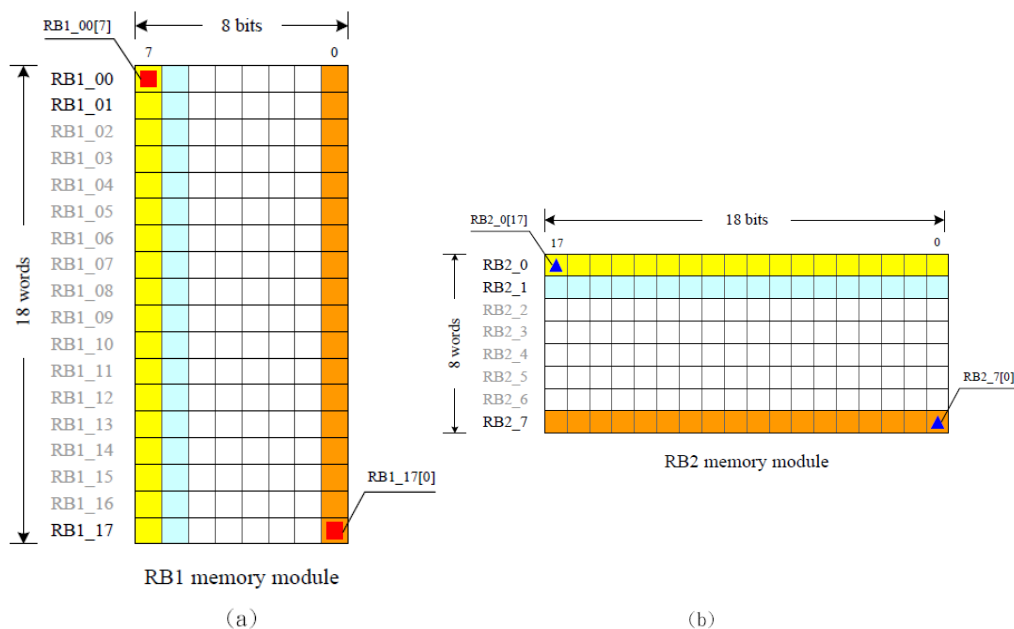
表二、S2 輸入/輸出信號

信號名稱	輸入/輸出	位元寬度	說明
<i>clk</i>	input	1	系統提供的時脈信號。
<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>S2_done</i>	output	1	當 S2 寫入 RB2 完成的時後，將 S2_done 設為 1 表示完成。
<i>RB2_RW</i>	output	1	Register Bank2 讀寫控制信號。 說明：當此信號為 high 時是對 Register Bank2 進行讀取(read)；low 則是進行對 Register Bank2 寫入(write)。
<i>RB2_A</i>	output	3	Register Bank2 三位元位址信號。
<i>RB2_D</i>	output	18	Register Bank2 十八位元資料輸入埠。
<i>RB2_Q</i>	input	18	Register Bank2 十八位元資料輸出埠。
<i>sen</i>	input	1	致能信號。 說明：當此信號為 low 時，sd 傳輸的資料被認為是有效的。
<i>sd</i>	input	1	序列傳輸中的一位元資料輸入埠。S2 於 clk 正緣時對該訊號取樣(sampling)。

2.2.1. Register Bank 1 (RB1)和 Register Bank2 (RB2)規格描述

本次作業使用到兩個記憶體模組 Register Bank 1 (以下稱為 RB1)及 Register Bank 2 (以下稱為 RB2),兩者都是單埠(single port)的 Register File 格式。RB1 的記憶體寬度是 8 位元,而記憶體深度是 18 個 word (請參考圖二(a))。如果要描述 RB1 中第一個 word 中的話,會用 RB1_00 表示;若要描述該 word 的 MSB 的話,會用 RB1_00[7]表示;同理,如果要描述 RB1 中最後一個 word 中的話,會用 RB1_17 表示;而最後一個 word 的 LSB 會用 RB1_17[0]表示。

RB2 的記憶體寬度是 18 位元,而記憶體深度是 8 個 word (請參考圖二(b))。同上面說明, RB2 中第一個 word 中的 MSB 的話會用 RB2_0[17]表示,最後一個 word 的 LSB 會用 RB2_7[0]表示。



圖二、RB1 示意圖(a)和 RB2 示意圖(b)

2.3. 系統功能描述

本序列/傳輸接收電路具有資料上傳(將 RB1 資料搬移至 RB2 中)的功能: 當發生系統重置(rst)時, S1 將 RB1 中的資料,轉換成為帶有記憶體位址和資料的封包(詳細的封包格式請參考 2.3.1),透過序列埠傳送到 S2。S2 收到訊號之後,根據封包內的記憶體位址將資料存放到 RB2 中。即本系統共實現 S1 及 S2 兩塊電路; S1 為一個模組, S2 為另一個模組。

整個系統工作方式如下:

- I. 首先測試樣本(testbench)會將 RB1 中填滿資料,並且把重置訊號 rst 設為 0
- II. Testfixture 將 rst 設為 1,對系統進行 RESET。此時 S1 的輸出 RB1_RW 及 sen 為 1,其餘(RB1_A、RB1_D、sd)輸出為 0;此時 S2 的輸出中 RB2_RW

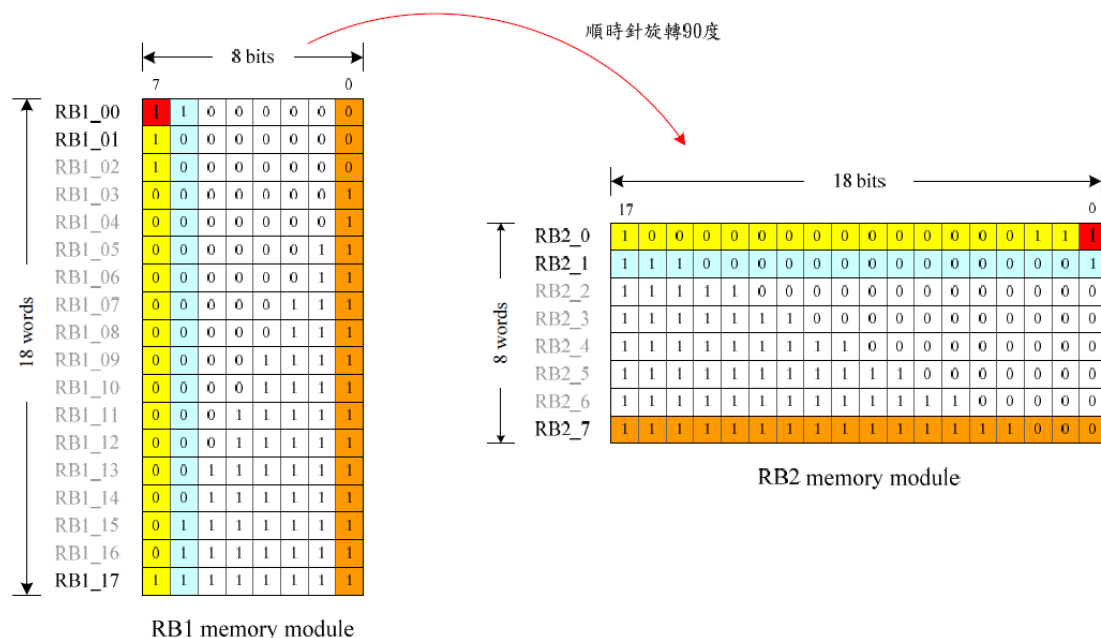
為 1，其餘(RB1_A、RB1_D、S2_done)輸出為 0

III. 當 S1 及 S2 電路完成 RESET 後，測試樣本將 rst 設為 0，此時 S1 開始將 RB1 的資料轉換成為序列封包的格式傳送，每次傳送是將 RB1_00 到 RB1_17 的第 n 個 bit (n 是 MSB 到 LSB)包裝成一個封包送出 (詳見 2.3.1)；S2 自 RESET 後即等著接收封包，一旦接收到封包並根據封包內所存的位址，將資料儲存在 RB2 對應的位址中。

IV. 當 RB2 所有資料都儲存完成之後(共八個封包)，S2 將 S2_done 設為 1 表示已完成上傳的動作。

RB1 和 RB2 資料對應關係：

RB1 及 RB2 記憶體模組是用來儲存傳送及接收的資料。當 RB1 中的資料依照系統功能描述的要求，完成上傳到 RB2 的動作後，RB2 中的資料分布方式將有如把 RB1 整個「順時針旋轉 90 度」(如圖三所示)。也就是說原本資料是用 18x8 的方式儲存在 RB1 中，上傳後在 RB2 中將以 8x18 的方式儲存。其中 RB1_17[7]這一 bit 資料將被上傳至 RB2_0[17]，RB1_16[7]這一 bit 資料將被上傳至 RB2_0[16]，...同理，RB1_17[6]這一 bit 資料將被上傳到 RB2_1[17]，RB1_16[6]這一 bit 資料將被上傳至 RB2_1[16]的位置，...其餘類推。圖三中，資料僅為舉例說明，並不代表實際測資。



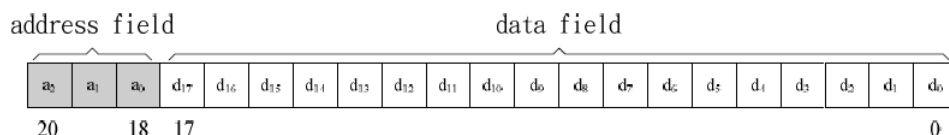
圖三、RB1 資料上傳到 RB2 範例

2.3.1. 封包格式

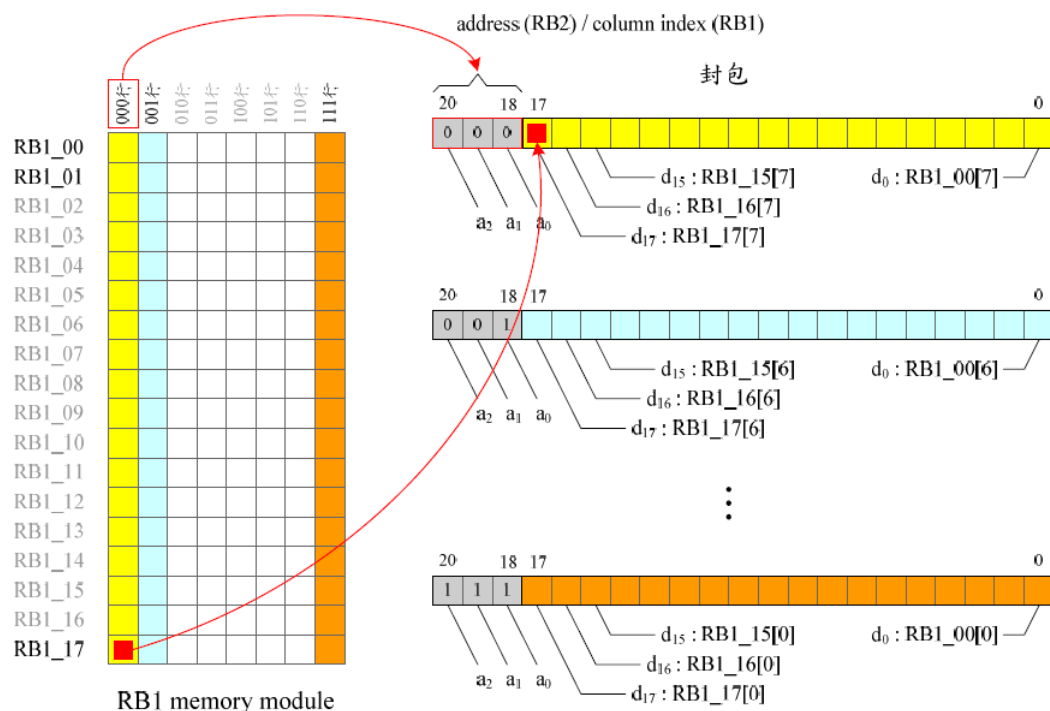
如圖四(a)所示，資料封包由 3 bit 的位址欄(address field)及 18 bit 的資料欄(data field)兩部份組成。S1 電路將整塊 RB1 記憶體中的資料以「行」(column)的方式切分成八個集合，從 MSB 到 LSB 依序編號(index)為 000、001、010、...、111。因為編號 000 的資料集合經過傳送後將儲存在 RB2 的 000 位址處，編號 001 的資料集合經過傳送後將儲存在 RB2 的 001 位址處，其餘類推；由此可知該編號恰好代表此封包內資料預定儲放在 RB2 的位址(address)。封包的第一部份(位址欄)即由此 3 bit 編號組成。此 3 bit 於傳送時將由 MSB 開始傳送。

而 RB1 的資料切分出的每一個集合，即構成各封包的資料欄(即封包的第二部份)。以編號 000 的資料部份為例，其 18 個 bit 的資料欄的第一個 bit 是編號 000 的資料集合與 RB1_17 的交集，接著第 2 個 bit 是編號 000 的資料集合與 RB1_16 的交集，...，第 18 個 bit (最後一個 bit)則是編號 000 的資料集合與 RB1_00 的交集(即由 RB1 的高位址往低位址方向依序排列)，如圖四(b)所示。

資料進行傳送時，S1 將由編號 000 的封包開始依序(sequential)傳送到 S2 側。



(a)封包格式

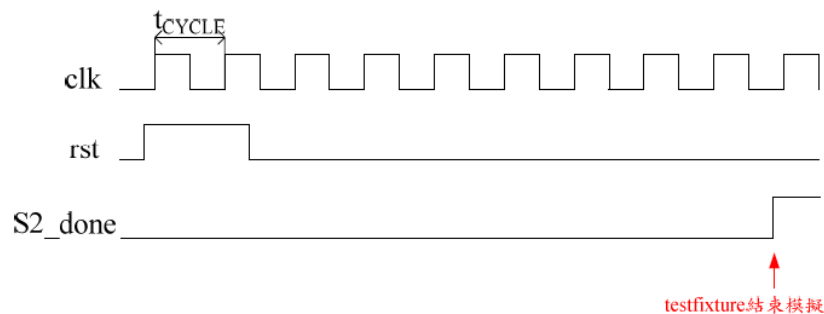


(b)封包的組成

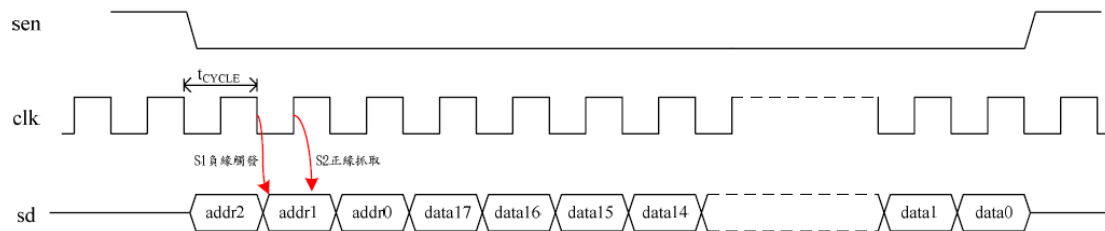
圖四、RB1 資料與封包關係圖

2.3.2. 時序規格圖

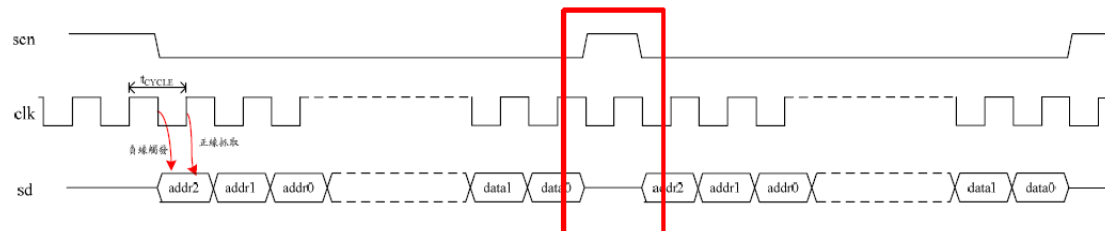
圖五為本系統中 clk、rst 及 S2_done 訊號之時序圖，本電路工作時脈 clk 之週期 t_{CYCLE} 固定為 100ns。rst 會舉為 high 至少維持一個週期以上。測試樣本會在時脈正緣的時候檢查 S2_done 是否為 1；當 S2_done 為 1 的時候，測試樣本會檢查 RB2 中的資料是否正確，然後結束模擬。圖六是 S1 傳送序列資料到 S2 的時候，序列傳輸埠 sd 及 sen 的波形圖，可以看到 S1 在負緣觸發的時候送出資料，而整個封包有效的情況是在 sen 維持 low 的時候，之後在由 S2 在 clk 正緣的時候抓取資料。而在傳輸過程中，封包與封包間至少要間隔一個週期（請參考圖七說明）。



圖五、系統時序圖



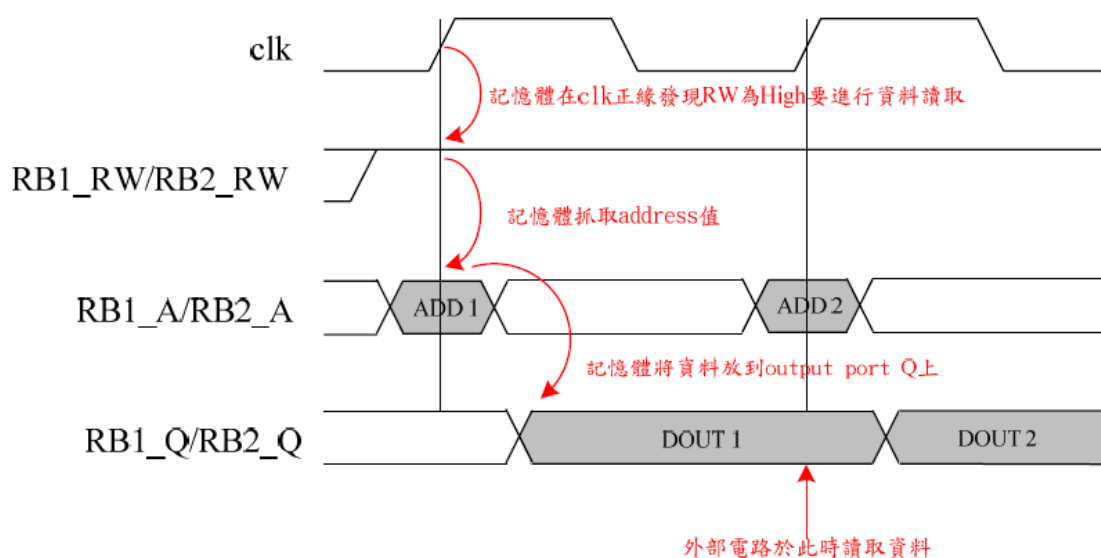
圖六、S1 上傳 S2 操作關係圖



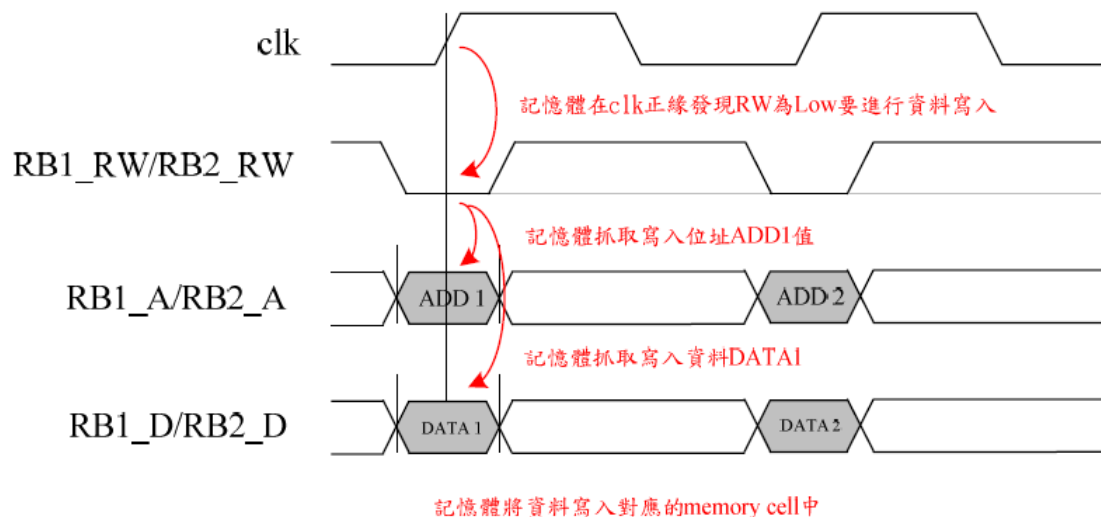
圖七、連續封包上傳關係圖

2.4. RB1 和 RB2 之時序規格

RB1 和 RB2 的讀取與寫入動作主要是以 RB1_RW/RB2_RW 這個信號來做控制。圖八為對記憶體進行資料讀取的波形時序圖，記憶體在 clk 的正緣確認到 RB1_RW/RB2_RW 為 High 時，會將此時的 RB1_A/RB2_A 位址匯流排上的 address 所對應到的資料，放到資料讀取匯流排(Q) RB1_Q/RB2_Q 上，以便在下一個 clk 正緣供電路讀取。圖九為對記憶體進行資料寫入的波形時序圖，記憶體在 clk 的正緣確認到 RB1_RW/RB2_RW 為 Low 時，會將此時的資料寫入匯流排(D) RB1_D/RB2_D 上的資料，寫到此時 RB1_A/RB2_A 位址匯流排所對應的記憶體單元上。**特別注意 若沒有要進行寫入時，請將 RB1_RW/RB2_RW 保持為 high，維持在讀取狀態。**



圖八、RB1/RB2 資料讀取波形



圖九、RB1/RB2 資料寫入波形

3. 評分標準

- 3.1. 電路可以完成 S1 與 S2 間封包傳輸正確, 35%
- 3.2. RB2 的內容正確(必須在 3.1 封包傳輸正確完成條件下), 35%
- 3.3. 電路可以完成 Gate-level Simulation(Timing Simulation), 20%
- 3.4. 電路 $Area \times Clock_rate$ 越小者, 分數越高, 10%

注意：除了所附的兩筆測試資料，助教會另行用第三筆測試資料來驗證功能的正確性。**模擬時必須 S1、S2 之間封包傳輸與 RB2 內容皆正確才算功能完整。**

4. 繳交方式

請將作業上傳到 moodle，並且壓縮成以下格式

#HW4_學號_版本.zip

附註：

第一版為 HW4_學號_001.zip

第一版為 HW4_學號_002.zip

繳交檔案：分三個資料夾

目錄名稱	檔案名稱
Pre_sim	S1.v、S2.v
Post_sim	S1.v、S1.sdo、S2.v、S2.sdo
說明文件	HW4_學號_版本.doc

說明文件：報告內容包含以下資訊

- (a) 電路設計概念
- (b) 電路 block diagram
- (c) 電路合成後的面積
- (d) 電路合成後的速度
- (e) 其他有助於助教了解同學設計的資訊

5. 評分方式

- (1) 作業 deadline、demo 的時間將公布在網頁
- (2) 遲交或上傳檔案有病毒者一律 0 分
- (3) 抄襲他人作業者一律 0 分

6. Q&A

有問題請 mail 給助教(p76014101@mail.ncku.edu.tw)