

Computer-Aided Design for VLSI Design

Homework1 (Student ID: 41147046S | Name: 楊子萱)

1. Provide a simple explanation of your code.

```
1 module dual_port_ram (  
2     input wire clk,  
3     input wire write_en,  
4     input wire [8:0] write_adr, // 512 = 2^9 → 9-bit address  
5     input wire [31:0] write_data,  
6     input wire [8:0] read_adr,  
7     output reg [31:0] read_data  
8 );
```

- clk：時脈訊號（所有動作都在時脈上升沿執行）
- write_en：寫入使能，當為 1 時才會寫資料
- write_adr：寫入資料的目標位址（因為記憶體深度是 512 → 需要 9-bit）
- write_data：要寫入的 32-bit 資料
- read_adr：要讀取的位址
- read_data：輸出讀取的 32-bit 資料

```
9     reg [31:0] mem [0:511];
```

宣告一個記憶體陣列，總共有 512 筆，每筆是 32-bit。

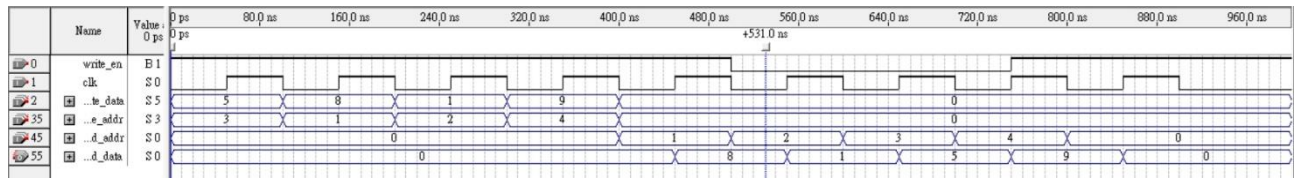
```
11     always @(posedge clk) begin  
12         if (write_en)  
13             mem[write_adr] <= write_data;  
14         read_data <= mem[read_adr];  
15     end
```

在每一個上升沿（posedge clk）：

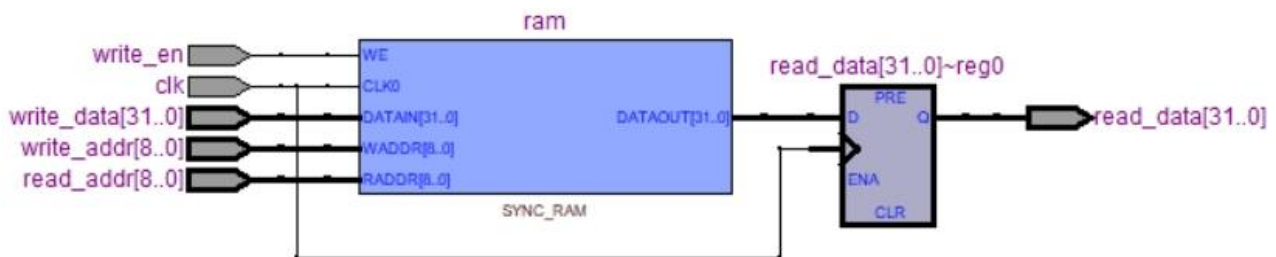
- 如果 write_en 為 1，就將 write_data 寫到 write_adr 指定的位址中。

- 同時，從 read_adr 指定的位址讀資料，放到 read_data。

2. Waveform diagram here (Simulation Results)



RTL



3. Reflections and discussions

實作了基於 LE 的 dual-port RAM，使用 always 區塊搭配正緣觸發時脈，完成同步寫入與讀取，學會了如何使用 reg 與記憶體 array 表達內部儲存結構。為了進一步比較資源使用，我也挑戰了 optional 部分，透過 MegaWizard Plug-In Manager 嘗試建立 Block RAM。雖然成功產生記憶體模組並寫好 top module 做對接，但在功能模擬階段出現 Netlist 缺失錯誤而卡住，尚未確認是否成功完成。這讓我意識到硬體設計不只寫 Verilog，工具設定與模擬流程同樣關鍵，未來會更注意設計與工具整合的完整性。