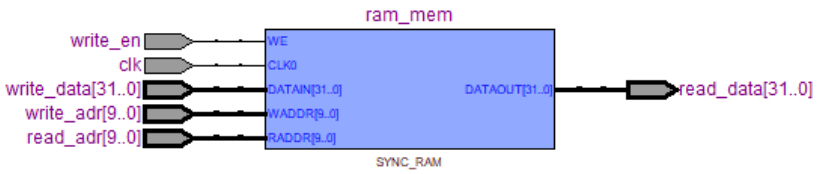
Computer-Aided Design for VLSI Design

Homework4（Student ID: 61247046S｜Name: 鄭琮達）

1. Provide a simple explanation of your code.
   在程式撰寫中，我自定義了一個 type ram_type is array (0 to 1023) of std_logic_vector(31 downto 0); 來設計 1024 entries(that has 32 bits)。
   以下是我的 VHDL 以及 Verilog 程式碼以及 RTL Viewer:

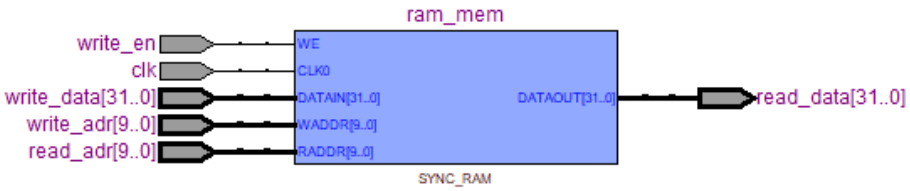
   VHDL

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity my_ram is
    port (
        clk      : in std_logic;
        write_en  : in std_logic;
        read_adr  : in std_logic_vector(9 downto 0);
        write_adr : in std_logic_vector(9 downto 0);
        write_data: in std_logic_vector(31 downto 0);
        read_data : out std_logic_vector(31 downto 0)
    );
end my_ram;

architecture behavioral of my_ram is
    type ram_type is array (0 to 1023) of std_logic_vector(31 downto 0);

begin
    process (clk, write_en, read_adr, write_adr)
    variable ram_mem : ram_type;
    begin
         read_data <= ram_mem(to_integer(unsigned(read_adr)));
        if rising_edge(clk) and write_en = '1' then
          ram_mem(to_integer(unsigned(write_adr))) := write_data;
        end if;
    end process;
end behavioral;
```

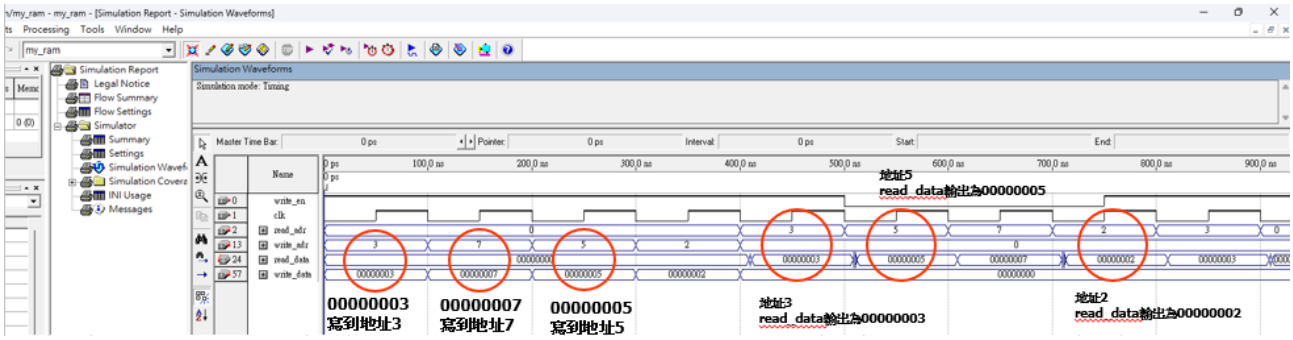


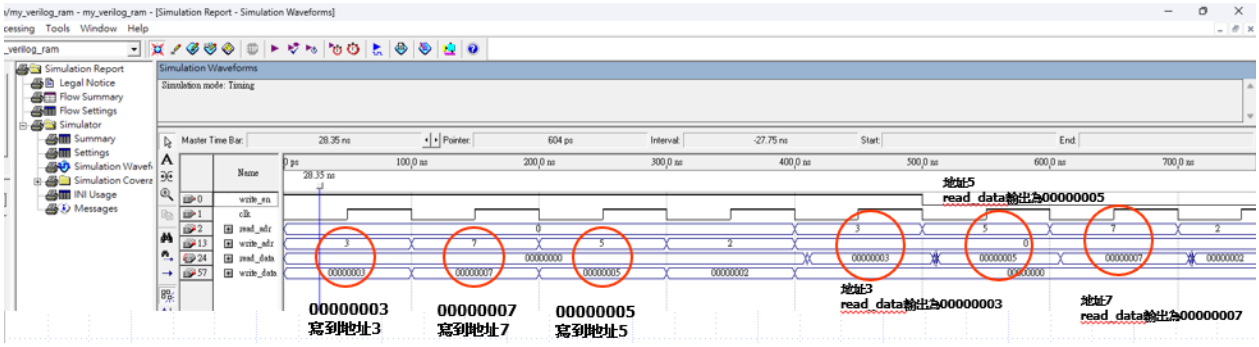
Verilog

```
module my_verilog_ram (
    input clk,
    input write_en,
    input [9:0] read_adr,
    input [9:0] write_adr,
    input [31:0] write_data,
    output reg [31:0] read_data
);

    reg [31:0] ram_mem [0:1023];

    always @(posedge clk) begin
        if (write_en)
            ram_mem[write_adr] <= write_data;
    end

    always @(*) begin
        read_data = ram_mem[read_adr];
    end

endmodule
```



2. Waveform diagram here (Simulation Results)





3. Reflections and discussions

謝謝教授以及助教提供作業設計 RAM 讓我學習 VHDL 以及 Verilog 的撰寫，兩者間的語法大同小異，但 VHDL 會較容易讓他人看懂以及追蹤程式碼，而 Verilog 的功能強大，可以幫我們省略很多撰寫過程，更盡人性化一點。總而言之，面對 Design 需求的不同，隨時切換應用是必要的能力。