| EX. NO: 06 | **RANDOM ACCESS MEMORY** |
|---|---|
| DATE : 11/10/2024 | |

**AIM:** To design and Implement the RAM with 1 Kbyte of memory using Verilog HDL.

**SOFTWARE USED:** Xilinx Vivado

**HARDWARE USED:** Basys3 FPGA Board

**RAM:**

**Verilog HDL Code:**

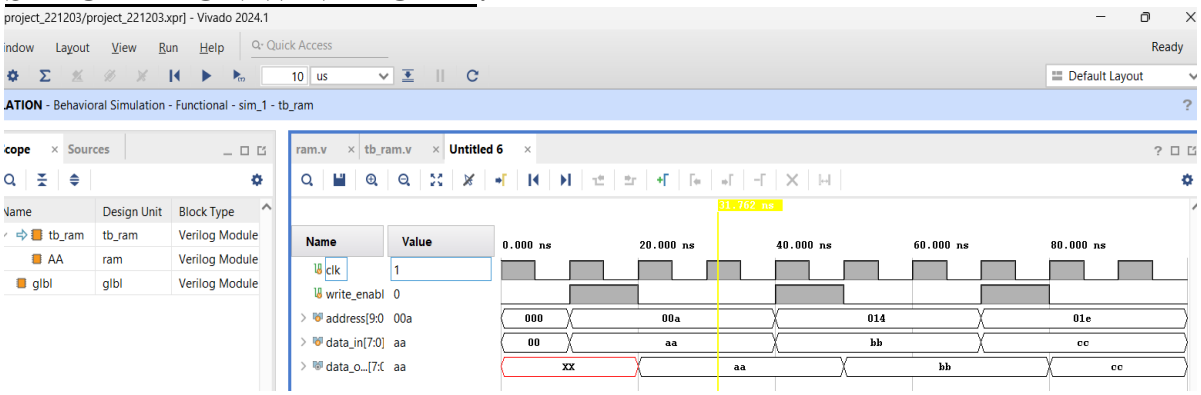
```
module ram(
    input clk,
    input write_enable,
    input [9:0] address,
    input [7:0] data_in,
    output reg [7:0] data_out
);
reg [7:0] ram_block[0:1023];
always @(posedge clk) begin
    if (write_enable) begin
        ram_block[address] <= data_in; // Write data to RAM
    end else begin
        data_out <= ram_block[address]; // Read data from RAM
    end
end
endmodule
```

**Test bench for RAM:**

```
module tb_ram;
    reg clk;
    reg write_enable;
    reg [9:0] address;
    reg [7:0] data_in;
    wire [7:0] data_out;
    ram AA(
        .clk(clk),
        .write_enable(write_enable),
        .address(address),
        .data_in(data_in),
        .data_out(data_out)
    );
        always #5 clk = ~clk;
    initial begin
        clk = 1;
```

```
        write_enable = 0;
        address = 0;
        data_in = 0;
        #10 write_enable = 1; address = 10; data_in = 8'hAA;
        #10 write_enable = 0; address = 10;
        #10 if (data_out !== 8'hAA) $display("Error: data_out = %h", data_out);
        #10 write_enable = 1; address = 20; data_in = 8'hBB;
        #10 write_enable = 0; address = 20;
        #10 if (data_out !== 8'hBB) $display("Error: data_out = %h", data_out);
        #10 write_enable = 1; address = 30; data_in = 8'hCC;
        #10 write_enable = 0; address = 30;
        #10 if (data_out !== 8'hCC) $display("Error: data_out = %h", data_out);
        #10 $finish;
    end
    initial begin
    $monitor("clk=%b,write_enable=%b,address=%b,data_in=%b,data_out=%b",
clk,write_enable,address,data_in,data_out);
    end
endmodule
```

## SIMULATION WAVEFORM:




## HARDWARE OUTPUT:

## I/O ports:

**Inputs:**

data_in[7]: T1
data_in[6]: U1
data_in[5]: W2
data_in[4]: R3
data_in[3]: T2
data_in[2]: T3
data_in[1]: V2
data_in[0]: W13
clk: W5
write_enable: R2

address[9]: W14
address[8]: V15
address[7]: W15
address[6]: W17
address[5]: W16
address[4]: V16
address[3]: V17
address[2]: T18
address[1]: T17
address[0]: U18

**Outputs:**

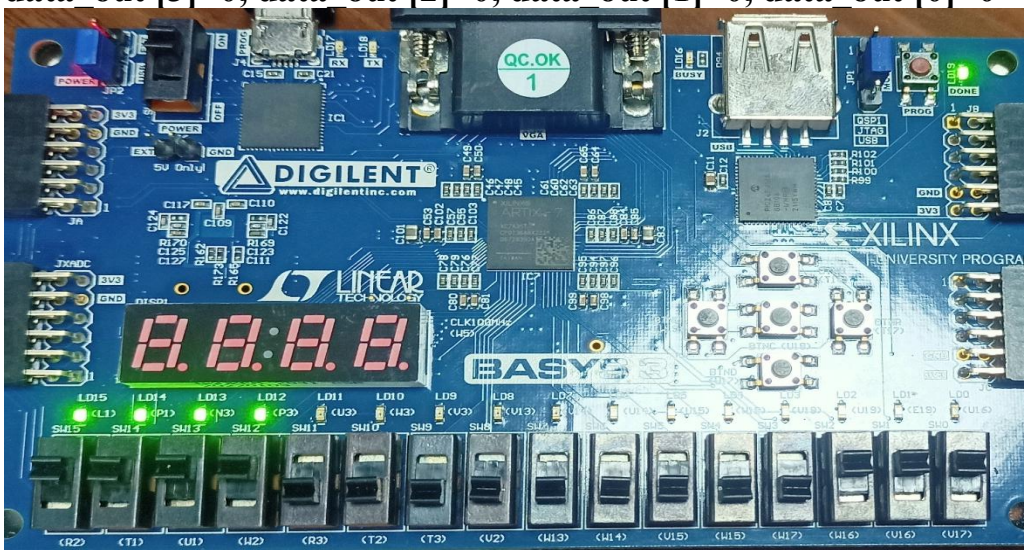
data_out[7]: L1
data_out [6]: P1
data_out [5]: N3
data_out [4]: P3
data_out [3]: U3
data_out [2]: W3
data_out [1]: V3
data_out [0]: V13

**For** data_in[7] = 1
data_in[6] = 1
data_in[5] = 1
data_in[4] = 0
data_in[3] = 0
data_in[2] = 0
data_in[1] = 0
data_in[0] = 0
clk = 1
write_enable = 1

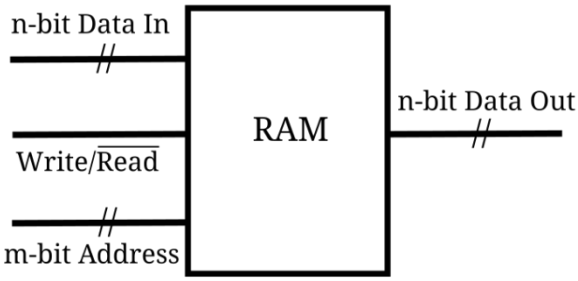
address[9] = 0
address[8] = 0
address[7] = 0
address[6] = 0
address[5] = 1
address[4] = 1
address[3] = 1
address[2] = 0
address[1] = 0
address[0] = 0

**Output:**

**Previously stored data as output(**since wirte_enable is in active state**):**
**data_out [7]=1, data_out [6]=1, data_out [5]=1, data_out [4]=1, data_out [3]=0, data_out [2]=0, data_out [1]=0, data_out [0]=0**




## CIRCUIT DIAGRAM:




## RESULT:

The RAM has been successfully designed and implemented using Verilog HDL, and its functionalities have been thoroughly verified through simulation with Xilinx Vivado software on the Basys3 FPGA board.