CA Lab: Homework 6

student: Dimitri Tabatadze

April 14, 2023

Task Description

Write the code for ROM which has 3-bit inputs and 8-bit outputs.

1. Use Case block for the address with the following conditions: for each input, the output will be from 1 to 64. *50 points*
2. Write the testbench and make the analyze of timing diagram. *40 points*

Please, solve the problems (write the comments) *5 points*, take the clear screenshots and combine all your solutions in one pdf file, then upload in teams. *5 points*

Solution

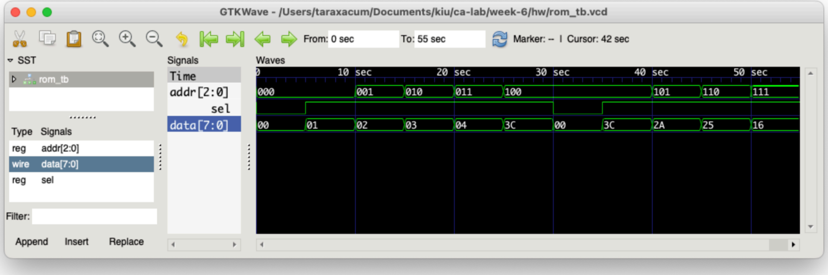
1. The Verilog code:

```verilog
module rom (
    input [2:0] address,
    input sel,
    output reg [7:0] data
);

always @ (sel or address) begin
    if (~sel)
        data = 8'd00;
    else begin
        case (address)
            0: data = 8'd01;
            1: data = 8'd02;
            2: data = 8'd03;
            3: data = 8'd04;
            4: data = 8'd60;
            5: data = 8'd42;
            6: data = 8'd37;
            7: data = 8'd22;
        endcase
    end
end

endmodule
```

2. The code for the testbench:

```
`include "rom.v"

module rom_tb();

reg [2:0] addr;
reg sel;
wire [7:0] data;

rom UUT(.address(addr), .sel(sel), .data(data));

initial begin
    $dumpfile("rom_tb.vcd");
    $dumpvars(0, rom_tb);
    addr = 0;
    sel = 0;
    #5; sel = 1;
    #5; addr = 1;
    #5; addr = 2;
    #5; addr = 3;
    #5; addr = 4;
    #5; sel = 0;
    #5; sel = 1;
    #5; addr = 5;
    #5; addr = 6;
    #5; addr = 7;
    #5;
    $finish;
end

endmodule
```

and the resulting timing diagram:



GTKWave - /Users/taraxacum/Documents/kiu/ca-lab/week-6/hw/rom_tb.vcd

From: 0 sec To: 55 sec Marker: -- | Cursor: 42 sec

Conclusion

The task was *almost* very clear. I think I did what I was supposed to do with the memory stored in the rom. I just put random integers in there.

Reference

- Me
- Myself
- I
- The slides (specifically `CE_6.pdf`)