ES204 Digital Systems
LAB Assignment - 8

Indian Institute of Technology, Gandhinagar
March 13, 2024

Marks : 30

Submission instructions:

Only one student from the team will submit with the word doc name Rollno1_Rollno2.pdf. The PDF will contain the code, testbench and simulation results.

1. **[30 Marks**]
   You have designed a parallel register where all the bits are simultaneously loaded into the register. In this lab, we will make a register file (where there are several registers).

   Design 32-bit register file with 8 registers. Name these registers as R0, R1, ..R7. You are to implement an instruction that allows **swapping** the contents of the two registers. The inputs to the Register file will be two addresses – ADDR0, ADDR1. These are given through the testbench.

   For example, SWAP R0, R1 is like saying copies the contents of Register-1 to Register-0 and the contents of Register-0 to Register-1.

   You should try and make a parameterized code.

Lab 8 ES 204

Birudugadda Srivibhav (22110050)
Reddybathuni Venkat (22110220)

Register file swap

1. Code

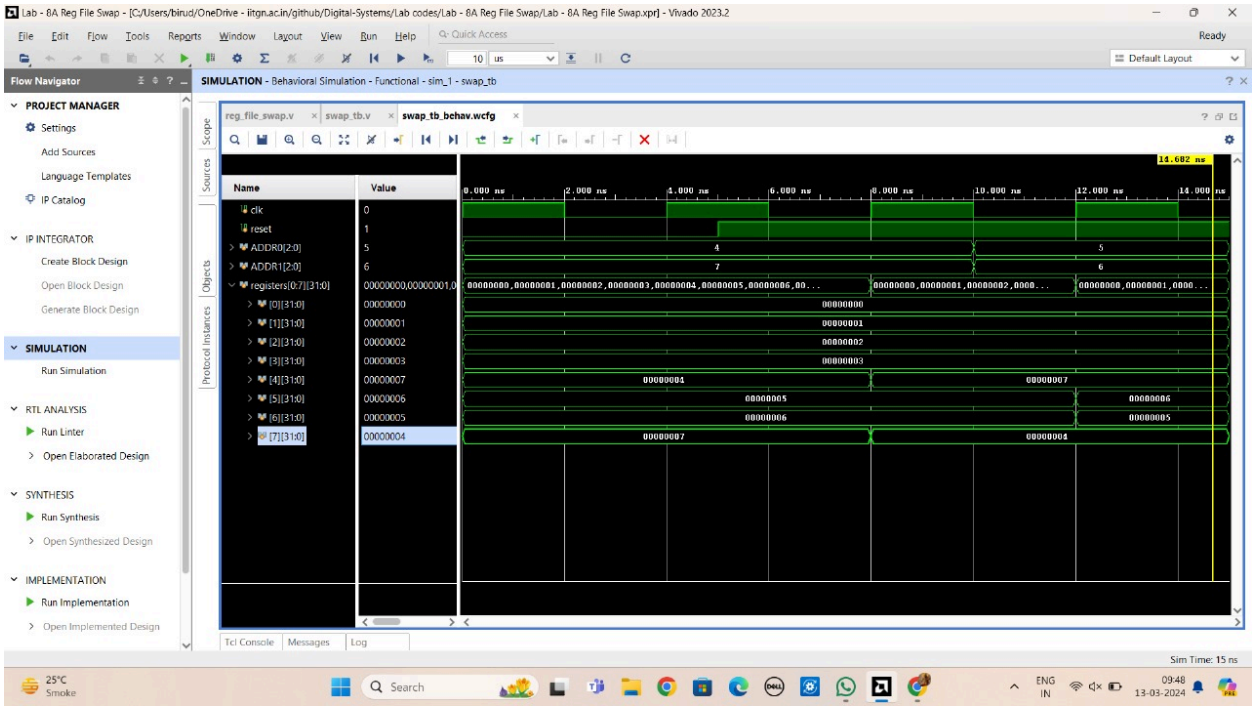
```
`timescale 1ns / 1ps
module reg_file #(parameter n = 32, k = 8)(
    input clk, reset,
    input [2:0] ADDR0, ADDR1
);

reg [n - 1:0] registers [0:k - 1];

always @(posedge clk) begin
    if (reset == 0) begin
        registers[0] <= 32'b000;
        registers[1] <= 32'b001;
        registers[2] <= 32'b010;
        registers[3] <= 32'b011;
        registers[4] <= 32'b100;
        registers[5] <= 32'b101;
        registers[6] <= 32'b110;
        registers[7] <= 32'b111;

    end else begin
            registers[ADDR0] <= registers[ADDR1];
            registers[ADDR1] <= registers[ADDR0];
    end
end

endmodule
```

2. Test bench

```
`timescale 1ns / 1ps
module swap_tb();
reg clk, reset;
reg [2:0] ADDR0,ADDR1;

reg_file uut(.clk(clk), .reset(reset), .ADDR0(ADDR0), .ADDR1(ADDR1));

initial
clk = 1;
always #2 clk = ~clk;

initial
begin
ADDR0 = 3'b100;
ADDR1 = 3'b111;
reset = 0;
#5;
reset = 1;
#5;
ADDR0 = 3'b101;
ADDR1 = 3'b110;
#5;
$finish();
end
endmodule
```

3. Simulation

First we swapped 4th and 7th address registers. Later 5th and 6th were swapped as shown in the figure below.


Lab - 8A Reg File Swap - [C:/Users/birud/OneDrive - iitgn.ac.in/github/Digital-Systems/Lab codes/Lab - 8A Reg File Swap/Lab - 8A Reg File Swap.xpr] - Vivado 2023.2
File Edit Flow Tools Reports Window Layout View Run Help
Ready
Default Layout
Flow Navigator
PROJECT MANAGER
Settings
Add Sources
Language Templates
IP Catalog
IP INTEGRATOR
Create Block Design
Open Block Design
Generate Block Design
SIMULATION
Run Simulation
RTL ANALYSIS
Run Linter
Open Elaborated Design
SYNTHESIS
Run Synthesis
Open Synthesized Design
IMPLEMENTATION
Run Implementation
Open Implemented Design
SIMULATION - Behavioral Simulation - Functional - sim_1 - swap_tb
reg_file_swap.v
swap_tb.v
swap_tb_behav.wcfg
Name
Value
clk
reset
ADDR0[2:0]
ADDR1[2:0]
registers[0:7][31:0]
[0][31:0]
[1][31:0]
[2][31:0]
[3][31:0]
[4][31:0]
[5][31:0]
[6][31:0]
[7][31:0]
14.682 ns
Tcl Console
Messages
Log
Sim Time: 15 ns
25°C
Smoke
Search
ENG
IN
09:48
13-03-2024