भारतीय प्रौद्योगिकी संस्थान गांधीनगर
पालज, गांधीनगर, गुजरात 382 055
INDIAN INSTITUTE OF TECHNOLOGY GANDHINAGAR
PALAJ, GANDHINAGAR, GUJARAT 382 055

Name : Bhavik Patel
Roll no. : 22110047

IITGN

Digital Systems
LAB -8

Question:
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. The register file allows moving contents of one register to another. The source and destination registers are given in the testbench.

For example, MOV R0, R1 is like saying move the contents of Register-1 to Register-0.

Verilog Code for 8 32-bit registers

```
`timescale 1ns / 1ps
module bit32Register(rst,D,clk,src,dest);
input rst,clk;
input [255:0]D;
input [2:0]src,dest;
reg [31:0]out[7:0];
integer i;
always @(*)
begin
    if(rst) begin
            out[0]<=D[31:0];
            out[1]<=D[63:32];
            out[2]<=D[95:64];
            out[3]<=D[127:96];
            out[4]<=D[159:128];
            out[5]<=D[191:160];
            out[6]<=D[223:192];
            out[7]<=D[255:224];
        end

end

always @(posedge clk) begin
    if(~rst) begin
        out[dest]<=out[src];
    end
end
endmodule
```

We have an rst (reset) input; if it is high, we initialize all 8 registers with some input given in the test bench.
We will store the registers as a 2-D array.
At the posedge of clk we will parallely load the value in src(source) register to dest(destination) register.

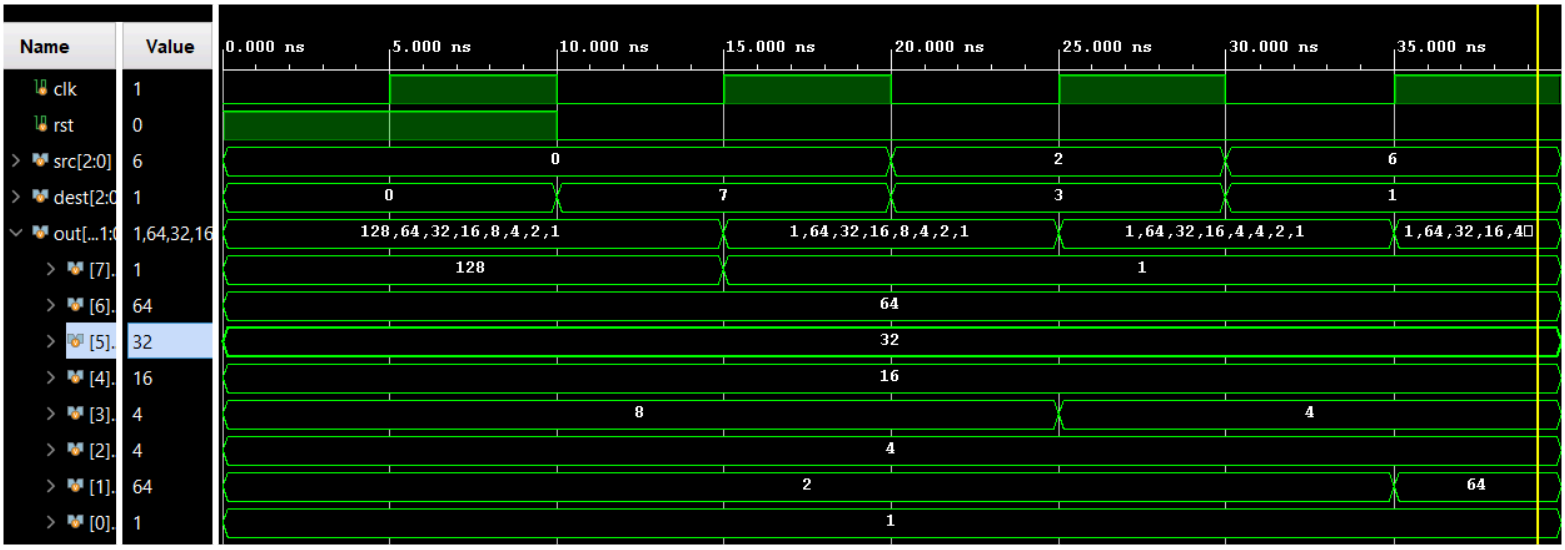
Registers are encoded as
R0=3'b000
R1=3'b001
R2=3'b010
R3=3'b011
R4=3'b100
R5=3'b101
R6=3'b110
R7=3'b111

Test bench

```
`timescale 1ns / 1ps

module test_bench();
reg rst,clk;
reg [2:0]src,dest;
reg [255:0]D;
integer i;
bit32Register uut(rst,D,clk,src,dest);

initial begin
    clk=0;
    forever #5 clk=~clk;
end

initial
begin
    rst=1;
    //Initialise the registers
    D=0;
    for(i=0; i<256;i=i+33) begin
        D[i]=1;
    end
    src=3'b000;
    dest=3'b000;
    D<=D;
    #10;
    rst=0;

    src=3'b000;
    dest=3'b111;
    #10;

    src=3'b010;
    dest=3'b011;
    #10;

    src=3'b110;
    dest=3'b001;
    #10;
```

NOTE: Parallel loading of one register to another register will happen at the posedge of the clock signal.

Simulation



1. First we have reset high so that we can initialize the registers.
2. Then we make rst=0 and at the first posedge of the clk we have src=0 and dest=7 which means we need to load or copy the content of register 0 to register7
3. We could see that the value stored in register 0 which is 1 is copied to register 7 at the psoedge of the clock.
4. Similar operations are carried out in the next inputs of the test bench also which can be verified from the simulation results.