Indian Institute of Information Technology, Vadodara

International Campus Diu

Practical Work Book

Name: Kshitiz

Roll Number: 202311045

Branch: Computer Science Engineering

Batch: 2023

Subject: COA

Experiment no. 5

Aim:-

Design and implementation of a register bank in Verilog.

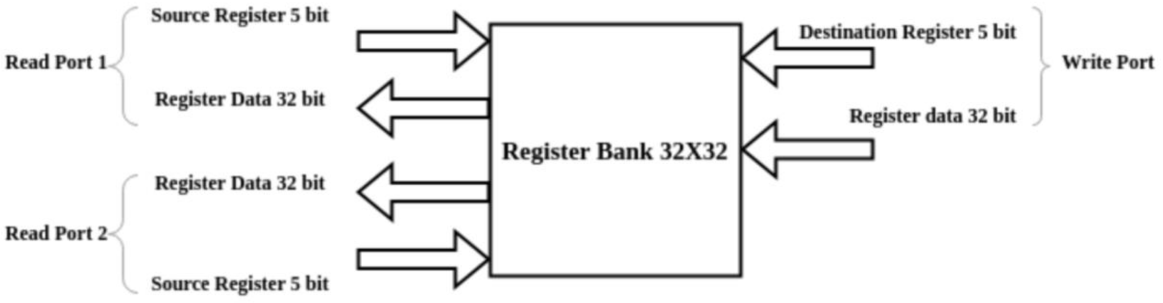
Prerequisites:-

A good understanding of register concepts and Verilog language.

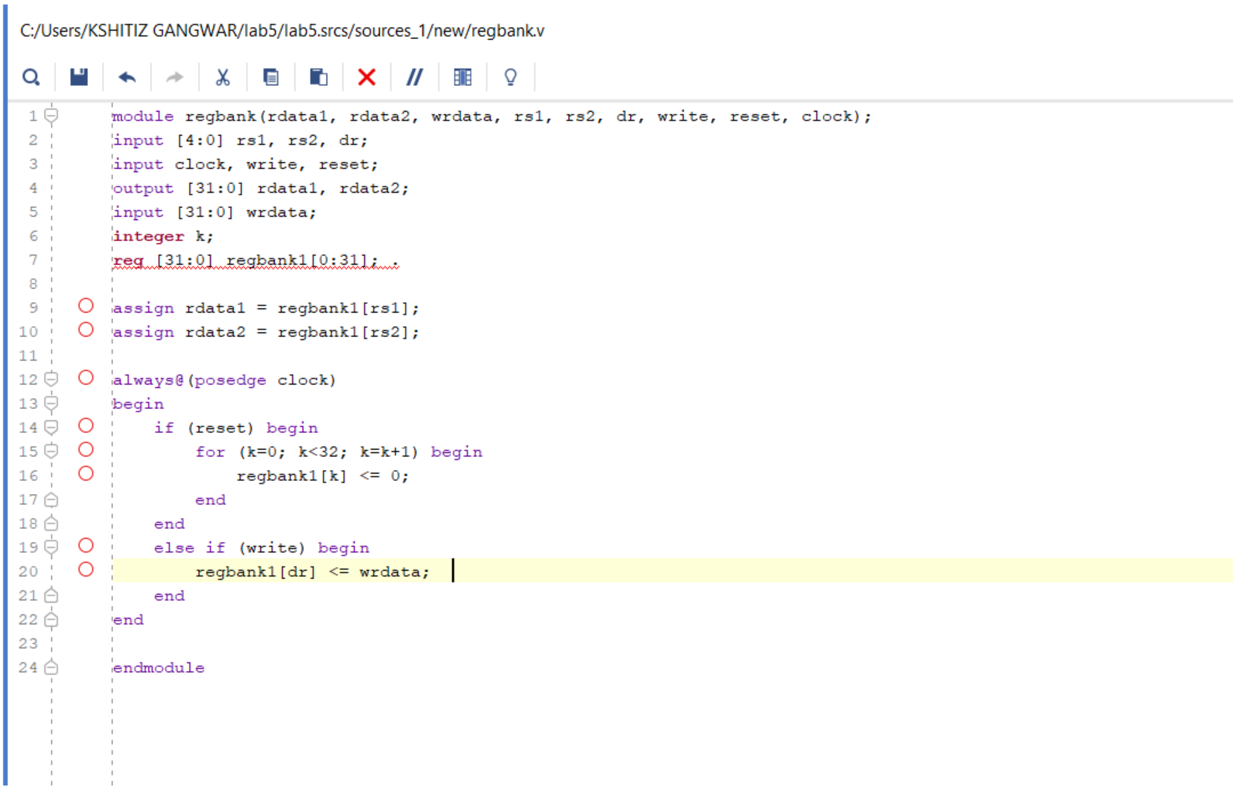
Theory:- Registers are the basic building block of any hardware design. The size of instructions executed by the processor depends on the size of registers.

Objective:-

● Design & simulate a register bank which has 2 register reads and 1 register write in every clock cycle. There is an assumption that the same reister can not be read and written at the same time.



CODE:



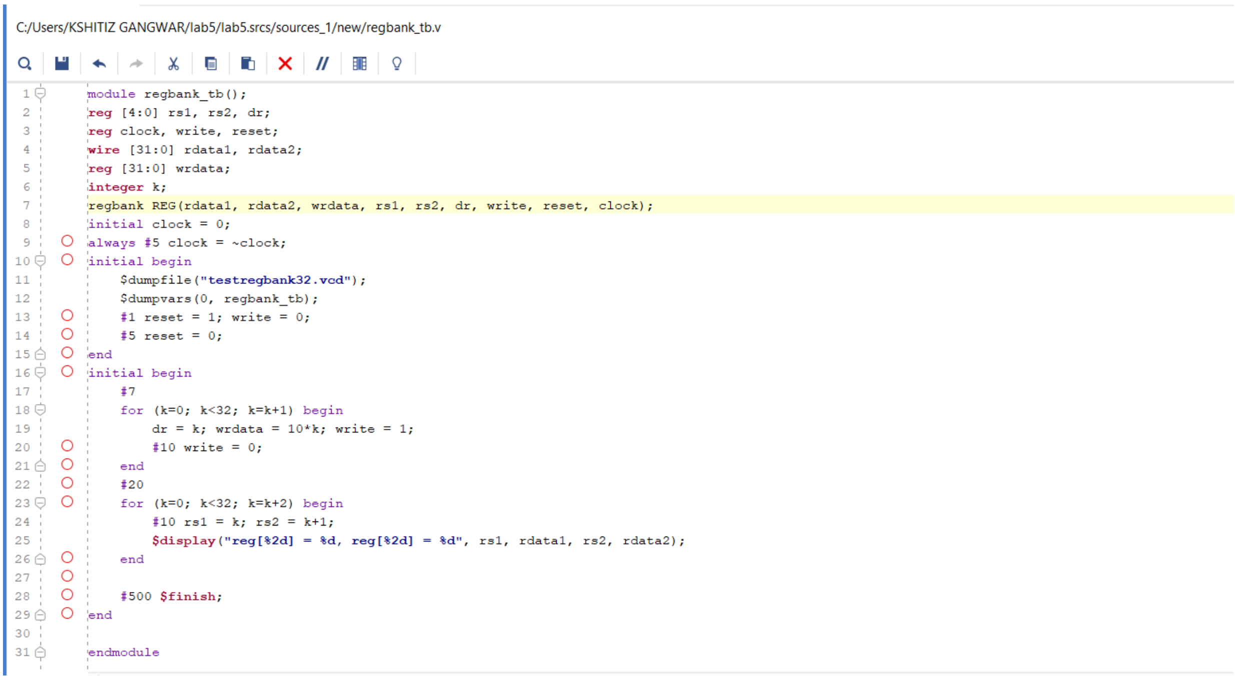
C:/Users/KSHITIZ GANGWAR/lab5/lab5.srcs/sources_1/new/regbank.v

```verilog
module regbank(rdata1, rdata2, wrdata, rs1, rs2, dr, write, reset, clock);
input [4:0] rs1, rs2, dr;
input clock, write, reset;
output [31:0] rdata1, rdata2;
input [31:0] wrdata;
integer k;
reg [31:0] regbank1[0:31]; .

assign rdata1 = regbank1[rs1];
assign rdata2 = regbank1[rs2];

always@(posedge clock)
begin
    if (reset) begin
        for (k=0; k<32; k=k+1) begin
            regbank1[k] <= 0;
        end
    end
    else if (write) begin
        regbank1[dr] <= wrdata;
    end
end

endmodule
```



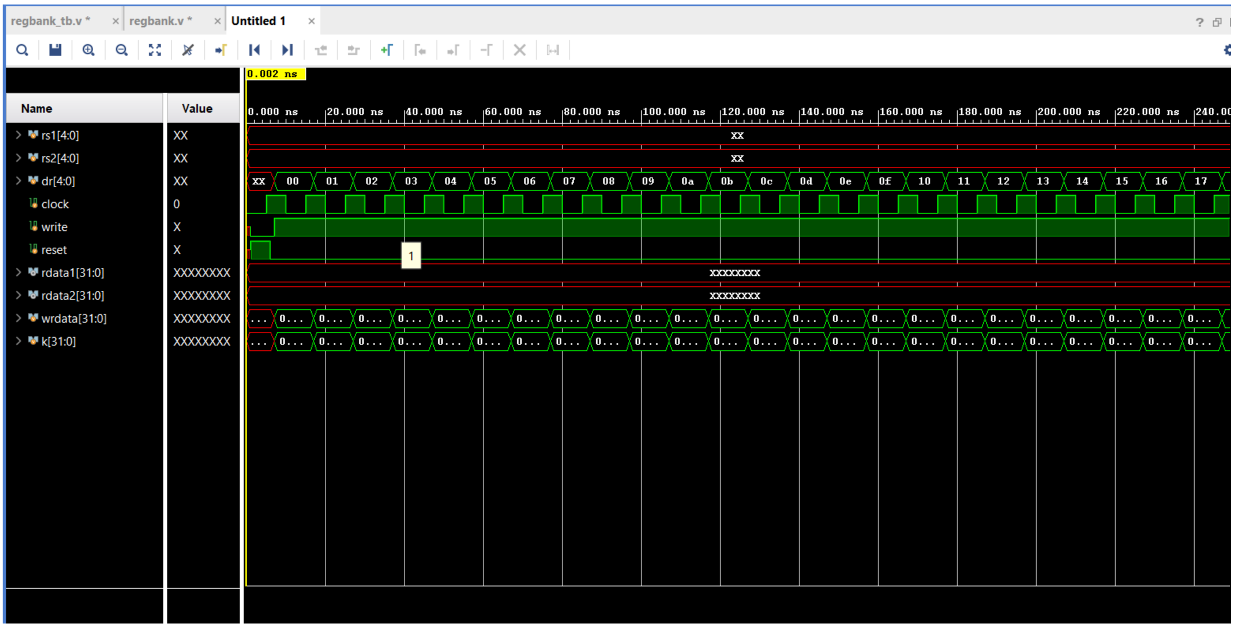
C:/Users/KSHITIZ GANGWAR/lab5/lab5.srcs/sources_1/new/regbank_tb.v

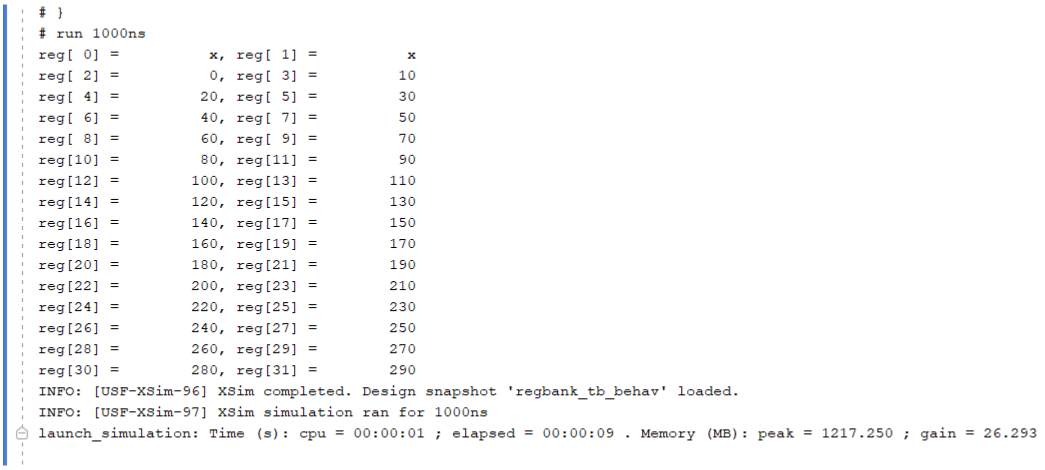
```verilog
module regbank_tb();
reg [4:0] rs1, rs2, dr;
reg clock, write, reset;
wire [31:0] rdata1, rdata2;
reg [31:0] wrdata;
integer k;
regbank REG(rdata1, rdata2, wrdata, rs1, rs2, dr, write, reset, clock);
initial clock = 0;
always #5 clock = ~clock;
initial begin
    $dumpfile("testregbank32.vcd");
    $dumpvars(0, regbank_tb);
    #1 reset = 1; write = 0;
    #5 reset = 0;
end
initial begin
    #7
    for (k=0; k<32; k=k+1) begin
        dr = k; wrdata = 10*k; write = 1;
        #10 write = 0;
    end
    #20
    for (k=0; k<32; k=k+2) begin
        #10 rs1 = k; rs2 = k+1;
        $display("reg[%2d] = %d, reg[%2d] = %d", rs1, rdata1, rs2, rdata2);
    end

    #500 $finish;
end

endmodule
```

OUTPUT:





```
# }
# run 1000ns
reg[ 0] =          x, reg[ 1] =          x
reg[ 2] =          0, reg[ 3] =         10
reg[ 4] =         20, reg[ 5] =         30
reg[ 6] =         40, reg[ 7] =         50
reg[ 8] =         60, reg[ 9] =         70
reg[10] =         80, reg[11] =         90
reg[12] =        100, reg[13] =        110
reg[14] =        120, reg[15] =        130
reg[16] =        140, reg[17] =        150
reg[18] =        160, reg[19] =        170
reg[20] =        180, reg[21] =        190
reg[22] =        200, reg[23] =        210
reg[24] =        220, reg[25] =        230
reg[26] =        240, reg[27] =        250
reg[28] =        260, reg[29] =        270
reg[30] =        280, reg[31] =        290
INFO: [USF-XSim-96] XSim completed. Design snapshot 'regbank_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:01 ; elapsed = 00:00:09 . Memory (MB): peak = 1217.250 ; gain = 26.293
```