**DESIGN CODE**

module arbiter(clk,rst,req\_a,req\_b,done,gnt\_a,gnt\_b,busy);

input clk,rst,req\_a,req\_b;

output reg gnt\_a,gnt\_b;

output reg busy;

output reg done;

always@(posedge clk)

begin

if(rst)

begin

busy=1'b0;

gnt\_a=1'b0;

gnt\_b=1'b0;

done=1'b0;

end

else

begin

if((req\_a==1'b1) && (!busy))

begin

gnt\_a<=1'b1;

busy<=1'b1;

end

if((req\_b==1'b1) && (!busy))

begin

gnt\_b<=1'b1;

busy<=1'b1;

end

if(gnt\_a==1'b1 || gnt\_b==1'b1)

begin

done<=1'b1;

end

if(done==1'b1)

begin

busy=1'b0;

done=1'b0;

end

end

end

endmodule

**TESTBENCH**

module arbiter\_test;

reg clk,rst,req\_a,req\_b;

wire gnt\_a,gnt\_b;

wire busy;

wire done;

arbiter m(clk,rst,req\_a,req\_b,done,gnt\_a,gnt\_b,busy);

always

begin

#5 clk=~clk;

end

initial

begin

$dumpfile("dump.vcd");

$dumpvars(1);

rst=1;clk=1; req\_a=1'b0;req\_b=1'b0;

#5rst=0;

#10req\_a=1'b1;

#50req\_b=1'b1;

#100$finish;

end

endmodule