//testbench

module lrrotatortb;

reg clk, rst;

reg [99:0] data;

reg [1:0] ena;

wire [99:0] q;

lrrotator uut( .clk(clk), .rst(rst), .data(data), .ena(ena), .q(q));

initial begin

$dumpfile ("lrrotator.vcd");

$dumpvars (1, lrrotatortb);

clk = 0;

forever #5 clk = ~clk;

end

initial begin

clk=0; rst=0; data = 100'b0; ena = 2'b00;

#10;

rst = 1;

#10;

rst = 0;

#10 data = 100'b1010101010; ena = 2'b00;

#10 ena = 2'b01;

#10 ena = 2'b00;

#10 ena = 2'b10;

#10 ena = 2'b00;

#10 ena = 2'b11;

#100 $finish;

end

endmodule

//design

module lrrotator( clk, rst data, ena, q);

input clk, rst;

input [99:0] data ;

input [1:0] ena;

output [99:0] q;

always @(posedge clk or negedge rst) begin

if (rst)

q <= 100'b0;

else begin

if (ena == 2'b01) begin //Rotate right

q[0] <= q[99];

q[99:1] <= q[98:0];

end else if (ena == 2'b10) begin // Rotate left

q[99] <= q[0];

q[98:0] <= q[99:1];

end else if (ena == 2'b00 || ena == 2'b11) begin // No rotation

q <= (ena == 2'b00) ? data : q;

end

end

end

endmodule