//testbench

module testbench;

reg [1:0] jk;

reg clk;

wire q, qb;

flipflop uut(.jk(jk), .clk(clk), .q(q), .qb(qb));

initial begin

clk = 0;

forever #5 clk = ~clk;

end

initial begin

$dumpfile("flipflop.vcd");

$dumpvars(1);

end

initial begin

jk = 2'b00;

#10;

jk = 2'b01;

#10;

jk = 2'b10;

#10;

jk = 2'b11;

#10;

$finish;

end

endmodule

//actual design

module flipflop(jk, clk, q,qb);

input [1:0] jk;

input clk;

output q,qb;

reg q,qb;

always @(posedge clk) begin

case (jk)

2'b00: q <= q; //No change

2'b01: q <= 0; // Set to 0

2'b10: q <= 1; // Set to 1

2'b11: q <= ~q; // Toggle

endcase

end

endmodule