**EXP NO: 13 DATE:**

**MOD-10 COUNTER**

**AIM:**

To write a verilog HDL program for mod-10 counter and verify its output.

**ALGORITHM:**

Step1: Define the specifications and initialize the design.   
Step2: Write the source code in VERILOG.  
Step3: Check the syntax and perform synthesis .  
Step4: Write different combinations of input using the test bench.  
 Step5:Verify the output by simulating the source code

**VERILOG SOURCE CODE:**

module counter(clk,rst,count);

input clk,rst;

output reg [3:0]count;

always@(posedge clk) begin

if(rst == 1 || count == 9)

count <= 0;

else

count <= count + 1;

end

endmodule

**Test Bench code**

module tb();

reg clk,rst;

wire [3:0]count;

counter c1(clk,rst,count);

initial begin

clk = 0;

forever #5 clk = ~clk;

end

initial begin

rst = 0;

#5 rst = 1;

#10 rst = 0;

#200 $finish;

end

initial

$monitor("%d%d",clk,count);

initial begin

$dumpfile("dump.vcd");

$dumpvars;

end

endmodule

SIMULATION OUTPUT:

Diagram

Description automatically generated

**RESULT:**

Thus a verilog HDL program was written for mod-10 counter and its output was verified.