DAY-81
#100DAYSOFRTL

AIM:--- IMPLEMENTATION OF BARREL SHIFT REG.

VERILOG CODE:--

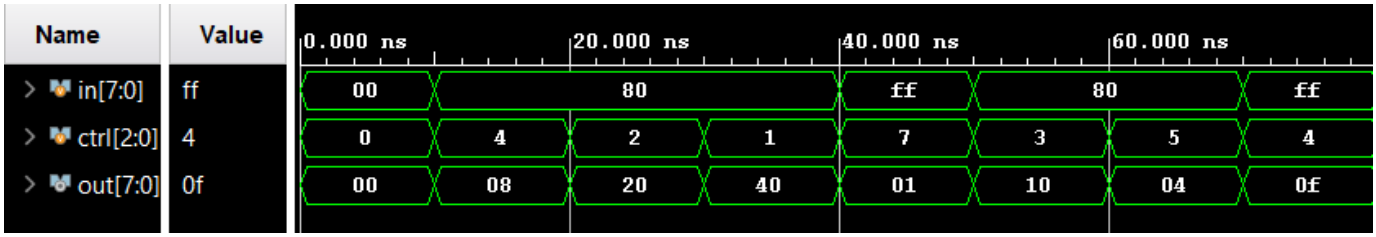
```
module barrelshifter(in, ctrl, out);
  input  [7:0] in;
  input [2:0] ctrl;
  output [7:0] out;
  wire [7:0] x,y;
mux2X1  ins_17 (.in0(in[7]),.in1(1'b0),.sel(ctrl[2]),.out(x[7]));
mux2X1  ins_16 (.in0(in[6]),.in1(1'b0),.sel(ctrl[2]),.out(x[6]));
mux2X1  ins_15 (.in0(in[5]),.in1(1'b0),.sel(ctrl[2]),.out(x[5]));
mux2X1  ins_14 (.in0(in[4]),.in1(1'b0),.sel(ctrl[2]),.out(x[4]));
mux2X1  ins_13 (.in0(in[3]),.in1(in[7]),.sel(ctrl[2]),.out(x[3]));
mux2X1  ins_12 (.in0(in[2]),.in1(in[6]),.sel(ctrl[2]),.out(x[2]));
mux2X1  ins_11 (.in0(in[1]),.in1(in[5]),.sel(ctrl[2]),.out(x[1]));
mux2X1  ins_10 (.in0(in[0]),.in1(in[4]),.sel(ctrl[2]),.out(x[0]));
mux2X1  ins_27 (.in0(x[7]),.in1(1'b0),.sel(ctrl[1]),.out(y[7]));
mux2X1  ins_26 (.in0(x[6]),.in1(1'b0),.sel(ctrl[1]),.out(y[6]));
mux2X1  ins_25 (.in0(x[5]),.in1(x[7]),.sel(ctrl[1]),.out(y[5]));
mux2X1  ins_24 (.in0(x[4]),.in1(x[6]),.sel(ctrl[1]),.out(y[4]));
mux2X1  ins_23 (.in0(x[3]),.in1(x[5]),.sel(ctrl[1]),.out(y[3]));
mux2X1  ins_22 (.in0(x[2]),.in1(x[4]),.sel(ctrl[1]),.out(y[2]));
mux2X1  ins_21 (.in0(x[1]),.in1(x[3]),.sel(ctrl[1]),.out(y[1]));
mux2X1  ins_20 (.in0(x[0]),.in1(x[2]),.sel(ctrl[1]),.out(y[0]));
mux2X1  ins_07 (.in0(y[7]),.in1(1'b0),.sel(ctrl[0]),.out(out[7]));
mux2X1  ins_06 (.in0(y[6]),.in1(y[7]),.sel(ctrl[0]),.out(out[6]));
mux2X1  ins_05 (.in0(y[5]),.in1(y[6]),.sel(ctrl[0]),.out(out[5]));
mux2X1  ins_04 (.in0(y[4]),.in1(y[5]),.sel(ctrl[0]),.out(out[4]));
mux2X1  ins_03 (.in0(y[3]),.in1(y[4]),.sel(ctrl[0]),.out(out[3]));
mux2X1  ins_02 (.in0(y[2]),.in1(y[3]),.sel(ctrl[0]),.out(out[2]));
mux2X1  ins_01 (.in0(y[1]),.in1(y[2]),.sel(ctrl[0]),.out(out[1]));
mux2X1  ins_00 (.in0(y[0]),.in1(y[1]),.sel(ctrl[0]),.out(out[0]));

endmodule
```

TESTBENCH CODE:---

```
1   module barrelshifter_tb;
2     reg [7:0] in;
3     reg [2:0] ctrl;
4     wire [7:0] out;
5
6   barrelshifter uut(.in(in), .ctrl(ctrl), .out(out));
7
8   initial
9    begin
10          in= 8'd0;  ctrl=3'd0; //no shift
11     #10 in=8'd128; ctrl= 3'd4; //shift 4 bit
12     #10 in=8'd128; ctrl= 3'd2; //shift 2 bit
13     #10 in=8'd128; ctrl= 3'd1; //shift by 1 bit
14     #10 in=8'd255; ctrl= 3'd7; //shift by 7bit
15     #10 in=8'd128; ctrl= 3'd3; //shift 3 bit
16     #10 in=8'd128; ctrl= 3'd5; //shift by 5 bit
17     #10 in=8'd255; ctrl= 3'd4; //shift by 5 bit
18    end
19      initial begin
20          #80 $finish;
21      end
22  endmodule
```

WAVEFORM:-----



SCHEMATIC BLOCK :-------