**PWM GENERATOR USING VERILOG HDL**

**DESIGN MODULE**

module pwm\_generator (

input wire clk,

input wire reset,

input wire [7:0] duty, // Duty cycle (0-255)

output reg pwm\_out

);

reg [7:0] counter;

always @(posedge clk or posedge reset) begin

if (reset) begin

counter <= 8'b0;

pwm\_out <= 0;

end else begin

counter <= counter + 1;

pwm\_out <= (counter < duty) ? 1 : 0;

end

end

endmodule

**TESTBENCH**

`timescale 1ns / 1ps

module pwm\_generator\_tb;

reg clk;

reg reset;

reg [7:0] duty;

wire pwm\_out;

pwm\_generator uut (

.clk(clk),

.reset(reset),

.duty(duty),

.pwm\_out(pwm\_out)

);

initial begin

clk = 0;

forever #5 clk = ~clk;

end

initial

begin

reset = 1;

duty = 8'd0;

$dumpfile("pwm\_waveform.vcd"); // VCD file for waveform

$dumpvars(0, pwm\_generator\_tb); // Dump all variables in this scope

#20 reset = 0;

#50 duty = 8'd64; // 25% duty cycle

#100 duty = 8'd128; // 50% duty cycle

#100 duty = 8'd192; // 75% duty cycle

#100 duty = 8'd255; // 100% duty cycle

#100 duty = 8'd0; // 0% duty cycle

#200 $finish;

end

endmodule

**WAVEFORM**

