NAME : KHUSHI VISHWAKARMA

CLASS : SYBSC(IT)

ROLL NO. : 18071

SUBJECT : (VSC)

PROJECT TOPIC : Image Pixel Converter using Verilog

AIM : To demonstrate a Image Pixel Converter program using iverilog.

PROGRAM:

DESIGN FILE

module grayscale\_converter(

input clk,

input reset,

input [7:0] R,

input [7:0] G,

input [7:0] B,

output reg [7:0] Gray

);

always @(posedge clk or posedge reset)

begin

if (reset)

Gray <= 8'd0;

else

Gray <= (R + G + B) / 3;

end

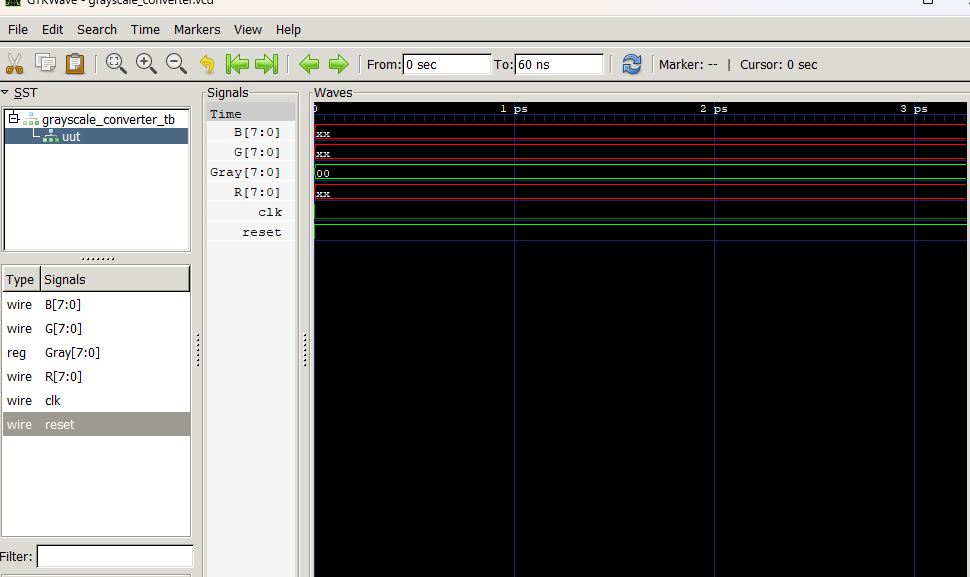
Endmodule

Testbench for Grayscale Converter

module grayscale\_converter\_tb;

reg clk, reset;  
reg [7:0] R, G, B;  
wire [7:0] Gray;  
  
// Instantiate the module  
grayscale\_converter uut(  
 .clk(clk),  
 .reset(reset),  
 .R(R),  
 .G(G),  
 .B(B),  
 .Gray(Gray)  
);  
  
// Clock generation  
always #5 clk = ~clk; // 10ns clock period  
  
initial begin  
 $dumpfile("grayscale\_converter.vcd");  
 $dumpvars(0, grayscale\_converter\_tb);  
  
 // Initialize  
 clk = 0;  
 reset = 1;  
  
 // Display header  
 $display("Time\tR\tG\tB\tGray");  
 $monitor("%0t\t%d\t%d\t%d\t%d", $time, R, G, B, Gray);  
  
 #10 reset = 0;  
  
 // Test pixel 1: (255, 0, 0) -> Red  
 R = 8'd255; G = 8'd0; B = 8'd0;  
 #10;  
  
 // Test pixel 2: (0, 255, 0) -> Green  
 R = 8'd0; G = 8'd255; B = 8'd0 #10;  
  
 // Test pixel 3: (0, 0, 255) -> Blue  
 R = 8'd0; G = 8'd0; B = 8'd255;  
 #10;  
  
 // Test pixel 4: (128, 128, 128) -> Gray  
 R = 8'd128; G = 8'd128; B = 8'd128;  
 #10;  
  
 // Test pixel 5: (255, 255, 255) -> White  
 R = 8'd255; G = 8'd255; B = 8'd255;  
 #10;  
  
 $finish;  
end

Endmodule

OUTPUT:

CONCLUSION : The above program has been executed successfully.