**Experiment-8 : Full adder with SC Test bench**

**Objective:**

To design a Full Adder and write a self-checking test bench that takes stimuli from a stored file.stimuli.tv. The test bench should check the generated output with the expected output and prints pass/fail messages.

**Tool Used:**

Xilinx ISE.

**Theory:**

fopen, foef, fscanf opens and reads the file for the written inputs. They are predefined functions to read the text file.

**DUT Code:**

module FA(input a,b,c, output s,cy);

    assign s = a^b^c, cy = a&&b||a&&c||b&&c;

endmodule

module mux(input a,b,s, output x);

    assign x = s ? b:a;

endmodule

module adder(input [15:0] a,b, input cin, output cout,output [15:0] sum);

    wire [16:0] car\_zero,car\_one;

    wire [15:0] sum\_zero,sum\_one;

    genvar i;

    assign car\_zero[0] = 1'b0, car\_one[0] = 1'b1;

    generate for(i=0;i<16;i=i+1)begin:adder\_circ

        FA zero(a[i],b[i],car\_zero[i],sum\_zero[i],car\_zero[i+1]);

        FA one(a[i],b[i],car\_one[i],sum\_one[i],car\_one[i+1]);

        mux opt(sum\_zero[i],sum\_one[i],cin,sum[i]);

    end endgenerate

    mux (car\_zero[16],car\_one[16],cin,cout);

endmodule

**TB Code:**

module tb;

    reg [15:0] a,b;

    reg cin;

    wire cout;

    wire [15:0] sum;

    integer fx,fy1,A;

    adder uut (a,b,cin,cout,sum);

    initial begin

        fx = $fopen("C:\\Users\\User\\Documents\\Code-sync\\M.Tech\_Verilog\\Labs\\Read\_Write\\test\_values.txt", "r");

        fy1= $fopen("C:\\Users\\User\\Documents\\Code-sync\\M.Tech\_Verilog\\Labs\\Read\_Write\\output\_values.txt", "w");

        while (! $feof(fx)) begin

         $fscanf(fx,"%d\n",A);

            a = A;

            $fscanf(fx,"%d\n",A);

            b = A;

            $fscanf(fx,"%d\n",A);

            cin = A;

            #1;

            #1;

            if ({cout,sum} == a+b+cin)begin

                $display("success");

                $fdisplay(fy1,"%d",{cout,sum});

            end

            else $display("failure %d, %d",{cout,sum} ,a+b+cin);

        end

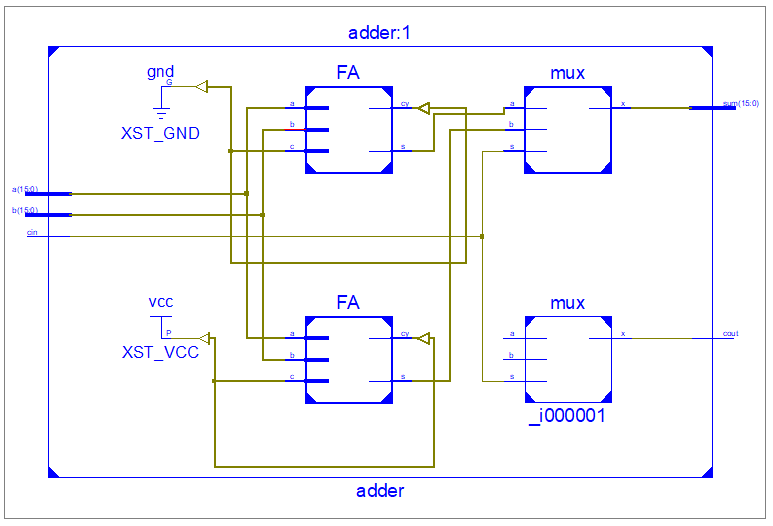
        $fclose(fx);

        $fclose(fy1);

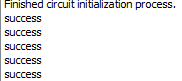
    end

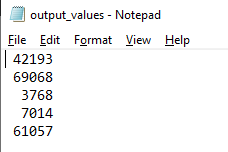
endmodule

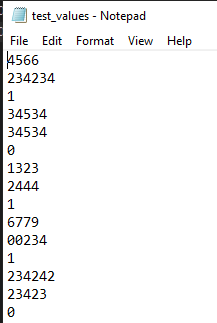
**RTL Diagram:**

****

**Simulation Output:**

****

****

****

**Result:**

The simulation output and the RTL diagram is observed and found to be valid.