**Experiment – 7**

**Verilog code for designing a Full Adder using 4:1 Multiplexer.**

**design.sv**module mux\_4\_to\_1(EN, S, I, Y);  
 input EN;  
 input [1:0]S;  
 input [0:3]I;  
 output Y;  
  
 assign Y = ~EN ? (~S[1] ? (~S[0] ? I[0] : I[1]) : (~S[0] ? I[2] : I[3])) : 0;  
endmodule  
  
module full\_adder\_using\_4\_to\_1\_mux(A, B, C, S, C\_out);  
 input A, B, C;  
 output S, C\_out;  
   
 mux\_4\_to\_1 sum(1'b0, {A, B}, {C, ~C, ~C, C}, S);  
 mux\_4\_to\_1 carry(1'b0, {A, B}, {1'b0, C, C, 1'b1}, C\_out);  
endmodule

**testbench.sv**module full\_adder\_using\_4\_to\_1\_mux\_test();  
 reg A, B, C;  
 wire S, C\_out;  
  
 full\_adder\_using\_4\_to\_1\_mux full\_adder\_using\_4\_to\_1\_mux\_dut(A, B, C, S, C\_out);  
   
 initial begin  
 A = 0; B = 0; C = 0; #10;  
 A = 0; B = 0; C = 1; #10;  
 A = 0; B = 1; C = 0; #10;  
 A = 0; B = 1; C = 1; #10;  
 A = 1; B = 0; C = 0; #10;  
 A = 1; B = 0; C = 1; #10;  
 A = 1; B = 1; C = 0; #10;  
 A = 1; B = 1; C = 1; #10;  
 $finish;  
 end  
  
 initial begin  
 $dumpfile("test.vcd");  
 $dumpvars(1, full\_adder\_using\_4\_to\_1\_mux\_test);  
 end  
endmodule

**Output Waveform**

