## **LAB # 02**

Develop, implement & simulate Full Adder & Half Adder through Schematic & Gate Level Modeling

<u>Objective</u>: Develop, implement & simulate Half Adder & Full Adder through Schematic & Gate Level Modeling.

Task 1: Develop, implement & simulate Half Adder through Schematic & Gate Level Modeling

```
lns / lps
 module HalfAdderTf;
   // Inputs
   reg a;
reg b;
   // Outputs
   wire sum;
wire carry;
    // Instantiate the Unit Under Test (UUT)
   HalfAdderModule uut (
      .a(a),
                                                       `timescale lns / lps
       .b(b),
       .sum(sum),
                                                        module HalfAdderModule(
       .carry(carry)
                                                              input a,
   initial begin
      // Initialize
a = 0; b = 0;
                                                              input b,
      #100;
                                                              output sum,
      a = 0; b = 0;
      #1002
                                                              output carry
      a = 0; b = 1;
      #100;
                                                              );
       a = 1; b = 0;
      #100;
      a = 1; b = 1;
                                                        xor(sum,a,b);
      #100;
       // Wait 100 ns for global reset to finish
                                                        and(carry,a,b);
       // Add stimulus here
   end
                                                        endmodule
endmodule
           carry_imp_carry1
                                         LPM XOR2 1:1
                                   and2b1
                                                                 or2
                                  ю,
                      Result_and0001_imp_Result_and00011
                                                          Result_imp_Result1
                                   and2b1
                      Result_and0000_imp_Result_and00001
                                            Mxor_sum
                                        HalfAdderModule
                                 (100 ns
                                        200 ns
                                                300 ns
                                                        400 ns
                                                               500 ns
                                                                       600 ns
                                                                              700 ns
                                                                                      800 ns
                                                                                              900 ns
```

Task 2: Develop, implement & simulate Full Adder using schematic & Gate Level Modeling

```
lns / lps
                                     module FullAdderTf;
                                        // Inputs
                                        reg a;
                                        reg b;
                                        reg cin;
                                        // Outputs
                                        wire sum;
                                        wire cout;
                                        // Bidirs
                                        wire sl;
`timescale lns / lps
                                        wire cl;
                                        wire c2;
module FullAdderModule(
                                        // Instantiate the Unit Under Test (UUT)
    input a,
                                        FullAdderModule uut (
    input b,
                                           .a(a),
    input cin,
                                            .b(b),
    inout sl,
                                           .cin(cin),
    inout cl,
                                           .sl(sl),
                                           .cl(cl),
    inout c2,
                                            .c2(c2),
    output sum,
                                            .sum(sum),
    output cout
                                           .cout (cout)
    );
                                        );
xor(sl,a,b);
                                        initial begin
xor (sum, sl, cin);
                                           // Initialize Inputs
and(cl,cin,sl);
                                           a = 0;
                                           b = 0;
and(c2,a,b);
or (cout, c1, c2);_
                                           cin = 0;
                                           // Wait 100 ns for global reset to finish
                                           #100;
endmodule
                                           a=0; b=0; #100;
Name
                             100 ns
             Value
                                       FullAdderModule
```