**Assignment 4**

Design 2:1 MUX with D flipflop

//Verilog Code

module mux2to1\_dff (

input wire a,

input wire b,

input wire sel,

input wire clk,

output reg);

wire mux\_out;

assign mux\_out = sel ? b : a;

always @(posedge clk) begin

y <= mux\_out;

end

endmodule

//Test Bench

// Code your testbench here

timescale 1ns / 1ps

module tb\_full\_adder;

reg a, b, cin;

wire sum, cout;

full\_adder uut (

.a(a),

.b(b),

.cin(cin),

.sum(sum),

.cout(cout)

);

initial begin

a = 0; b = 0; cin = 0; #10;

a = 0; b = 0; cin = 1; #10;

a = 0; b = 1; cin = 0; #10;

a = 0; b = 1; cin = 1; #10;

a = 1; b = 0; cin = 0; #10;

a = 1; b = 0; cin = 1; #10;

a = 1; b = 1; cin = 0; #10;

a = 1; b = 1; cin = 1; #10;

end

endmodule