UM TAGUM COLLEGE

DEPARTMENT OF ENGINEERING EDUCATION

COMPUTER ENGINEERING PROGRAM

Dataflow Modelling

DRILL 6

NAME:

STUDENT NUMBER:

TERMINAL NUMBER:

DATE OF PERFORMANCE:

DATE OF SUBMISSION:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PROFESSOR

I. DISCUSSION

Dataflow modelling of combinational logic uses a number of operators that act on operands to produce desired results. It uses continuous assignments and the keyword **assign**. If the identifier of a net is a left-hand side of a continuous assignment statement or a procedural assignment statement, the value assigned to the net is specified by an expression that uses operands and operators.

A continuous assignment assigns a value to a net; it cannot be used to assign a value to a register. The syntax of a continuous assignment is:

***assign*** *[ delay ] LHS\_net = RHS\_expression ;*

A continuous assignment can appear as part of a net declaration itself. Such an assignment is called a net declaration assignment.

***wire*** *[ 3 : 0 ] Sum = 4 ‘b0*

***wire*** *Clear**= ‘b1 ;*

II. Drill Exercises

1. Create a dataflow description of a 4-bit comparator (use relational operators *greater than* or *less than* and the equality operator *is equal to*). Save the file as drill 6\_1.vl

*module magcom (input [3:0] A, B, output lt, gt, eq);*

*assign lt=(A<B);*

*assign gt=(A>B);*

*assign eq=(A==B);*

*endmodule*

*module testMagCom();*

*reg [3:0] A, B;*

*wire lt, gt, eq;*

*magcom tb(A, B, lt, gt, eq);*

*reg [255:0]string1, string2, string3;*

*initial fork*

*A=1'b0; B=1'b0;*

*$display(" A B \t\t\t\t A<B \t\t \t\t A>B\t\t\t\t A==B");*

*$monitor("%d %d %s %s %s",A, B, string1, string2, string3);*

*join*

*initial begin*

*#1 A=4'd7;*

*B=4'd7;*

*if (lt==1) string1="true";*

*else string1="false";*

*if (gt==1) string2="true";*

*else string2="false";*

*if (eq==1) string3="true";*

*else string3="false";*

*#3 A=4'd4;*

*B=4'd6;*

*if (lt==1) string1="true";*

*else string1="false";*

*if (gt==1) string2="true";*

*else string2="false";*

*if (eq==1) string3="true";*

*else string3="false";*

*#5 A=4'd9;*

*B=4'd8;*

*if (lt==1) string1="true";*

*else string1="false";*

*if (gt==1) string2="true";*

*else string2="false";*

*if (eq==1) string3="true";*

*else string3="false";*

*#7 A=4'd10;*

*B=4'd1;*

*if (lt==1) string1="true";*

*else string1="false";*

*if (gt==1) string2="true";*

*else string2="false";*

*if (eq==1) string3="true";*

*else string3="false";*

*#9 $finish;*

*end*

*endmodule*

1. Design an all-bit zero/one detector using dataflow modelling. Save the file as drill6\_2.vl

*module AllBit(input [31:0]x, output zero, one);*

*assign zero=~(|x);*

*assign one=&x;*

*endmodule*

*module AllBitTest();*

*reg [31:0] inputX;*

*wire outputZ, outputO;*

*AllBit Abit(inputX, outputZ, outputO);*

*initial fork*

*$monitor($time,,," %h",inputX," %h",outputO," %h",outputZ);*

*inputX=32'h0;*

*#1 inputX=32'h12345678;*

*#2 inputX=0;*

*#3 inputX=32'hFFFFFFFF;*

*#4 inputX=32*

*#4 $finish;*

*join*

*endmodule*

1. Construct a JK flip-flop from a D flip-flop and gates, and use the circuit to design the Verilog program. Use behavioural model for the D flip-flop and dataflow model for the JK flip-flop. Save the file as drill6\_3.vl

*module JK\_flipflop(output A, input J, K, clk, reset);*

*wire AofJK;*

*assign AofJK=(J&~A)|(~K&A);*

*D\_flipflop JKf(A,AofJK,clk, reset);*

*endmodule*

*module D\_flipflop(output reg B, input D, clk, reset);*

*always@(posedge clk, negedge reset)*

*begin*

*if(~reset)*

*B<=1'b0;*

*else*

*B<=D;*

*end*

*endmodule*

*module TestFlipFlop;*

*reg J, K, clk, reset;*

*wire Q;*

*always #1 clk=~clk;*

*JK\_flipflop JKF(Q,J,K,clk,reset);*

*initial clk=0;*

*initial reset=0;*

*initial J=0;*

*initial K=0;*

*initial $monitor("clk=%b reset=%b, J=%b, K=%b, Q=%b",clk,reset,J,K,Q);*

*initial fork*

*#28 $finish;*

*#2 reset=1;*

*#4 J=1;*

*#8 K=1;*

*#12 J=0;*

*#16 K=0;*

*#20 J=1;*

*#24 J=0;*

*join*

*endmodule*

III. Programming Exercise

1. Create a dataflow model of a circuit that compares two 8-bit numbers to check which among the input has the greater number of bits that are set (bit is equal to 1). Save the file exercise 6\_1.vl
2. A certain counter follows the following sequence: 5-2-7-0-3-1-6 back to 5. Create a behavioural model of the circuit. Use T flip-flop designed under dataflow model of Verilog. Save the file as exercise 6\_2.vl.

IV. Review Questions

1. Why is *dataflow* modelling named as such?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Why is *continuous* assignment in dataflow modelling named as such?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. What is the effect of imposing a dataflow assignment to a nonblocking procedure?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_