Lab 12

Top module

```
module top(
    input clk,

    input [31:0] q1,
    output q1_ready,
    input q1_valid,

    input [31:0] q2,
    output q2_ready,
    input q2_valid,

    input [31:0] q3,
    output q3_ready,
    input q3_valid,

    input [31:0] q4,
    output q4_ready,
    input q4_valid,

    input out_ready,
    output out_valid,
    output reg [2:0] res=0
    );

    wire [7:0] temp1;
```

```
    wire [31:0] val1;
    wire [7:0] temp2;
    wire [31:0] val2;
    wire [7:0] temp3;
    wire [31:0] val3;

    floating_point_0 cmp1 (
      .aclk(clk),                                  // input wire aclk
      .s_axis_a_tvalid(q1_valid),            // input wire s_axis_a_tvalid
      .s_axis_a_tready(q1_ready),            // output wire
s_axis_a_tready
      .s_axis_a_tdata(q1),              // input wire [31 : 0]
s_axis_a_tdata
      .s_axis_b_tvalid(q2_valid),            // input wire s_axis_b_tvalid
      .s_axis_b_tready(q2_ready),            // output wire
s_axis_b_tready
      .s_axis_b_tdata(q2),              // input wire [31 : 0]
s_axis_b_tdata
      .m_axis_result_tvalid(valid1),  // output wire
m_axis_result_tvalid
      .m_axis_result_tready(ready1),  // input wire
m_axis_result_tready
      .m_axis_result_tdata(temp1)    // output wire [7 : 0]
m_axis_result_tdata
    );

    assign val1=(temp1)?(q1):(q2);

    floating_point_0 cmp2 (
```

```
        .aclk(clk),                                 // input wire aclk
        .s_axis_a_tvalid(valid1),            // input wire s_axis_a_tvalid
        .s_axis_a_tready(ready1),              // output wire
s_axis_a_tready
        .s_axis_a_tdata(val1),               // input wire [31 : 0]
s_axis_a_tdata
        .s_axis_b_tvalid(q3_valid),             // input wire
s_axis_b_tvalid
        .s_axis_b_tready(q3_ready),              // output wire
s_axis_b_tready
        .s_axis_b_tdata(q3),                // input wire [31 : 0]
s_axis_b_tdata
        .m_axis_result_tvalid(valid2),  // output wire
m_axis_result_tvalid
        .m_axis_result_tready(ready2),  // input wire
m_axis_result_tready
        .m_axis_result_tdata(temp2)    // output wire [7 : 0]
m_axis_result_tdata
    );

    assign val2=(temp2)?(val1):(q3);

    floating_point_0 cmp3 (
        .aclk(clk),                                 // input wire aclk
        .s_axis_a_tvalid(valid2),            // input wire s_axis_a_tvalid
        .s_axis_a_tready(ready2),              // output wire
s_axis_a_tready
        .s_axis_a_tdata(val2),               // input wire [31 : 0]
s_axis_a_tdata
```

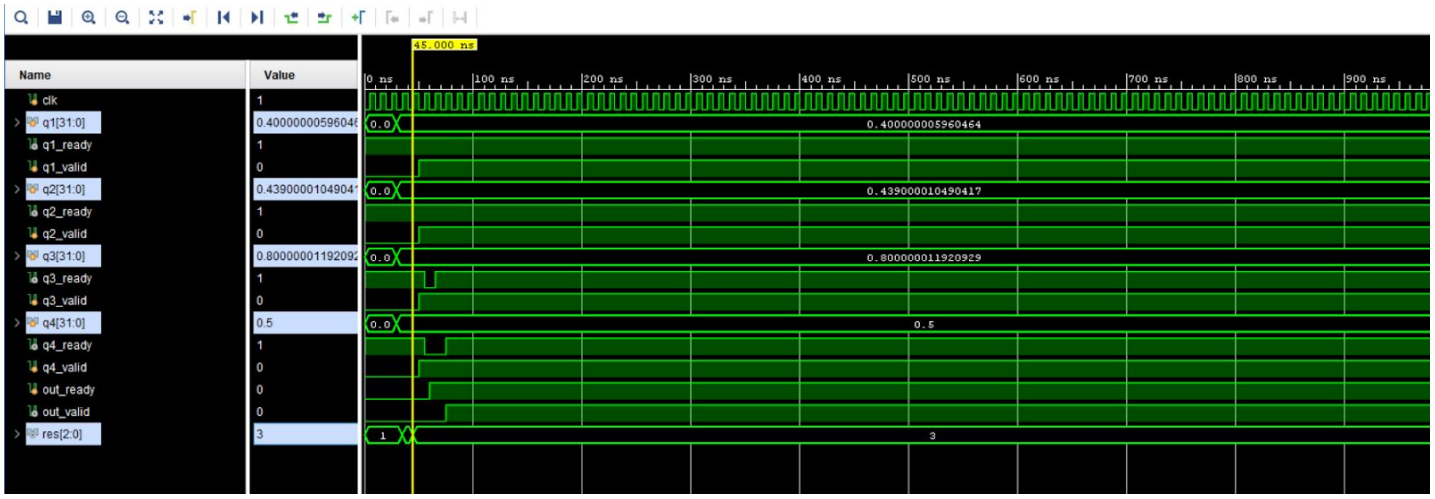
```
        .s_axis_b_tvalid(q3_valid),            // input wire
s_axis_b_tvalid
        .s_axis_b_tready(q4_ready),            // output wire
s_axis_b_tready
        .s_axis_b_tdata(q4),              // input wire [31 : 0]
s_axis_b_tdata
        .m_axis_result_tvalid(out_valid),  // output wire
m_axis_result_tvalid
        .m_axis_result_tready(out_ready),  // input wire
m_axis_result_tready
        .m_axis_result_tdata(temp3)    // output wire [7 : 0]
m_axis_result_tdata
    );

    assign val3=(temp3)?(val2):(q4);

    always @ (*)
    begin
        if(val3==q1)
            res<=3'b001;
        else if(val3==q2)
            res<=3'b010;
        else if(val3==q3)
            res<=3'b011;
        else if(val3==q4)
            res<=3'b100;
    end

endmodule
```

Testbench

```
module tb(
    );

    reg clk=0;

    always #5 clk=~clk;

    reg [31:0] q1=0;
    wire q1_ready;
    reg q1_valid=0;

    reg [31:0] q2=0;
    wire q2_ready;
    reg q2_valid=0;

    reg [31:0] q3=0;
    wire q3_ready;
    reg q3_valid=0;

    reg [31:0] q4=0;
    wire q4_ready;
    reg q4_valid=0;

    reg out_ready=0;
    wire out_valid;
    wire [2:0] res;
```

```
    top
fun1(clk,q1,q1_ready,q1_valid,q2,q2_ready,q2_valid,q3,q3_ready
,q3_valid,q4,q4_ready,q4_valid,out_ready,out_valid,res);

    initial
    begin

      #30;
          q1=32'b00111110_11001100_11001100_11001101;
          q1_valid=0;
          q2=32'b00111110_11100000_11000100_10011100;
          q2_valid=0;
          q3=32'b00111111_01001100_11001100_11001101;
          q3_valid=0;
          q4=32'b00111111_00000000_00000000_00000000;
          q4_valid=0;

      #20;
          q1_valid=1;
          q2_valid=1;
          q3_valid=1;
          q4_valid=1;

      #10;
          out_ready = 1'b1;

    end

endmodule
```

Output


45.000 ns
Name
Value
clk
1
q1[31:0]
0.4000000059604
0.0
0.400000005960464
q1_ready
1
q1_valid
0
q2[31:0]
0.4390000104904
0.0
0.439000010490417
q2_ready
1
q2_valid
0
q3[31:0]
0.8000000119209
0.0
0.800000011920929
q3_ready
1
q3_valid
0
q4[31:0]
0.5
0.0
0.5
q4_ready
1
q4_valid
0
out_ready
0
out_valid
0
res[2:0]
3
1
3
0 ns
100 ns
200 ns
300 ns
400 ns
500 ns
600 ns
700 ns
800 ns
900 ns