```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 05/12/2022 03:55:29 PM
// Design Name:
// Module Name: negative
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module negativeSign(

  input up,
    input down,
    input clk,
    output [7:0] q,
    output neg,
    input reset
    );
    //need to find out if possible to declare multiple busses at once
    wire dpos,dneg;
    wire [7:0]qneg;
    wire [7:0]qpos;
    wire [1:0] utcp;
    wire [1:0]dtcp;
    wire [1:0]dtcn;
    wire [1:0]utcn;

    wire [3:0] start;
    assign start =4'b0000;

    //postive
    countUD4L pos30(.clk(clk), .Up(up & (~dpos | dpos & dneg)), .Dw(down &(~dpos |
dpos & dneg & ~down)),
```

```
    .LW(reset), .d(start), .q(qpos[3:0]), .UTC(utcp[0]), .DTC(dtcp[0]));
    countUD4L pos47(.clk(clk), .Up(utcp[0] & up * (~dpos | dpos & dneg)),
.Dw(dtcp[0]& down & (~dpos | dpos & dneg & ~down) ),
     .LW(reset), .d(start), .q(qpos[7:4]), .UTC(utcp[1]), .DTC(dtcp[1]));

    //getting both dtc to put back into the logic as a whole
    assign dpos=(dtcp[0] & dtcp[1]);
    //negative
    countUD4L neg30(.clk(clk), .Up(down & (~dneg | dpos & dneg)), .Dw(up & (~dneg |
dpos & dneg  & ~up)),
    .LW(reset), .d(start), .q(qneg[3:0]), .UTC(utcn[0]), .DTC(dtcn[0]));
    countUD4L neg47(.clk(clk), .Up(utcn[0]&down & (~dneg | dpos & dneg)),
.Dw(dtcn[0]& up & (~dneg | dpos & dneg & ~up)),
    .LW(reset), .d(start), .q(qneg[7:4]), .UTC(utcn[1]), .DTC(dtcn[1]));

    assign dneg=(dtcn[0] & dtcn[1]);
    //when dneg is false that means that we are on a negative number
    assign neg = ~dneg;
    //chosing between which pos or neg number counter to run at the moment
    assign q= qpos & ~dpos | qneg & ~dneg;


endmodule
```