

```

timescale 1ns / 1ps
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 10/17/2023 02:42:13 PM
// Design Name:
// Module Name: countUD4L
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

module countUD4L(

    input clk,          // the system clock,
    input btnU,          // (increment)
    input btnD,          // (decrement)
    input LD,            // (load control)
    input [3:0] Din,     // the 4-bit bus Din that will be loaded on the clock
                        // edge if LD is high.

    output [3:0] Q,      // a 4-bit bus Q which is the current value held by
                        // the counter,
    output UTC,          // the signal UTC (btnU Terminal Count) which is 1 only
                        // when the counter is at 4'b1111 (15 in decimal), and
    output DTC           // the signal DTC (Down Terminal Count) which is 1 only
                        // when the counter is at 4'b0000.
);
    wire Dwire [3:0];
    wire Inc;
    assign Inc = (btnU ^ btnD | LD);

    assign Dwire[0] = ((Q[0]^Inc) & btnU & ~btnD) | (((Q[0]^Inc) & ~btnU & btnD) &
~LD) | (LD & Din[0]);
    assign Dwire[1] = ((Q[1]^ (Inc & Q[0])) & btnU & ~btnD) | (((Q[1]^ (Inc &
~Q[0])) & ~btnU & btnD) & ~LD) | (LD & Din[1]);

```

```

    assign Dwire[2] = ((Q[2]^ (Inc & Q[0] & Q[1])) & btnU & ~btnD) | (((Q[2]^ (Inc &
~Q[0] & ~Q[1])) & ~btnU & btnD) & ~LD) | (LD & Din[2]);

    assign Dwire[3] = ((Q[3]^ (Inc & Q[0] & Q[1] & Q[2])) & btnU & ~btnD) | (((Q[3]^
(Inc & ~Q[0] & ~Q[1] & ~Q[2])) & ~btnU & btnD) & ~LD) | (LD & Din[3]));

    assign UTC = &(Q[3:0]);
    assign DTC = &(~Q[3:0]);

    FDRE #(.INIT(1'b0)) b0_ff (.C(clk), .R(1'b0), .CE(Inc), .D(Dwire[0]), .Q(Q[0]));
    // From "Using Flip-Flops in Vivado"
https://classes.soe.ucsc.edu/cse100/Fall23/lab/FDRE/FDRE.html
    FDRE #(.INIT(1'b0)) b1_ff (.C(clk), .R(1'b0), .CE(Inc), .D(Dwire[1]), .Q(Q[1]));
    FDRE #(.INIT(1'b0)) b2_ff (.C(clk), .R(1'b0), .CE(Inc), .D(Dwire[2]), .Q(Q[2]));
    FDRE #(.INIT(1'b0)) b3_ff (.C(clk), .R(1'b0), .CE(Inc), .D(Dwire[3]), .Q(Q[3]));

endmodule

```