```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 04/25/2021 07:25:15 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,
    input Up,
    input Dw,
    input LD,
    input [3:0] Din,
    output [3:0] Q,
    output UTC,
    output DTC
    );
    //wire [3:0] G;
    wire [3:0] D;
    wire [3:0] up;
    wire [3:0] down;

    //assign D[0] = (~LD & ((~Up & ~Dw) & Q[0]) | (~(Up & Dw)
    //& (~Q[0] & (Up | Dw))))
    //| (LD & Din[0]);
    //assign D[1] = (~LD & ((~Up & ~Dw) & Q[1]) | (~(Up & Dw)
    //& ((~Q[1] & Q[0] & Up & ~Dw) | (Q[1] & ~Q[0] & Up & ~Dw) | (Q[1] & Q[0] & Dw
& ~Up)
    //| (~Q[3] & ~Q[2] & ~Q[1] & ~Q[0] & Dw & ~Up) | (~Q[1] & ~Q[0] & Dw & ~Up))))
    //| (LD & Din[1]);
    //assign D[2] = (~LD & ((~Up & ~Dw) & Q[2]) | (~(Up & Dw)
```

```
    //& ((~Q[2] & Q[1] & Q[0] & Up & ~Dw) | (Q[2] & ~(Q[1] & Q[0])  & Up & ~Dw)
    //| (~Q[2] & ~Q[1] & ~Q[0] & Dw & ~Up) | (~Q[3] & ~Q[2] & ~Q[1] & ~Q[0] & Dw & ~U
    //| (Q[2] & ~(~Q[1] & ~Q[0]) & Dw & ~Up))))
    //| (LD & Din[2]);
    //assign D[3] = (~LD & (((~Up & ~Dw) & Q[3]) | (~(Up & Dw)
    //& ((~Q[3] & Q[2] & Q[1] & Q[0] & Up & ~Dw) | (Q[3] & ~(Q[2] & Q[1] & Q[0]) &
Up & ~Dw)
    //| (~Q[3] & ~Q[2] & ~Q[1] & ~Q[0] & Dw & ~Up) | (Q[3] & ~(~Q[2] & ~Q[1] &
~Q[0]) & Dw & ~Up)))))
    //| (LD & Din[3]);

    assign D[0] = (LD & Din[0]) | ((~LD & Up & ~Dw) & (Up ^ Q[0])) | ((~LD & Dw &
~Up) & (Dw ^ Q[0]));
    assign D[1] = (LD & Din[1]) | ((~LD & Up & ~Dw) & ((Up & Q[0]) ^ Q[1])) | ((~LD
& Dw & ~Up) & ((Dw & ~Q[0]) ^ Q[1]));
    assign D[2] = (LD & Din[2]) | ((~LD & Up & ~Dw) & ((Up & Q[0] & Q[1]) ^ Q[2])) |
((~LD & Dw & ~Up) & ((Dw & ~Q[0] & ~Q[1]) ^ Q[2]));
    assign D[3] = (LD & Din[3]) | ((~LD & Up & ~Dw) & ((Up & Q[0] & Q[1] & Q[2]) ^
Q[3])) | ((~LD & Dw & ~Up) & ((Dw & ~Q[0] & ~Q[1] & ~Q[2]) ^ Q[3]));

    FDRE #(.INIT(1'b0) ) ff_instance_1095 (.C(clk), .CE(LD | (Up ^ Dw)), .D(D[0]),
.Q(Q[0]));
    FDRE #(.INIT(1'b0) ) ff_instance_1096 (.C(clk), .CE(LD | (Up ^ Dw)), .D(D[1]),
.Q(Q[1]));
    FDRE #(.INIT(1'b0) ) ff_instance_1097 (.C(clk), .CE(LD | (Up ^ Dw)), .D(D[2]),
.Q(Q[2]));
    FDRE #(.INIT(1'b0) ) ff_instance_1098 (.C(clk), .CE(LD | (Up ^ Dw)), .D(D[3]),
.Q(Q[3]));



    assign UTC = Q[3] & Q[2] & Q[1] & Q[0];
    assign DTC = ~Q[3] & ~Q[2] & ~Q[1] & ~Q[0];
endmodule
```