

```
// File: collatz.v
// John Hubbard, 08 Mar 2015
//
// hw6: Problem 4.7.3: Implement the Collatz conjecture, for n = 27.
//
// For n = 27, this will display:
//
// 27, 82, 41, 124, 62, 31, 94, 47, 142, 71, 214, 107, 322, 161, 484, 242,
// 121, 364, 182, 91, 274, 137, 412, 206, 103, 310, 155, 466, 233, 700,
// 350, 175, 526, 263, 790, 395, 1186, 593, 1780, 890, 445, 1336, 668,
// 334, 167, 502, 251, 754, 377, 1132, 566, 283, 850, 425, 1276, 638, 319,
// 479, 1438, 719, 2158, 1079, 3238, 1619, 4858, 2429, 7288, 3644, 1822,
// 911, 2734, 1367, 4102, 2051, 6154, 3077, 9232, 4616, 2308, 1154,
// 577, 1732, 866, 433, 1300, 650, 325, 976, 488, 244, 122, 61, 184, 92,
// 46, 23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5, 16, 8, 4, 2, 1
```

```
`timescale 1ns/1ns
```

```
module collatz(clk, reset, enableNext, startingN, f);
    input clk, enableNext, reset;
    output reg [31:0]f;
    input [31:0]startingN;

    always @(posedge clk)
    begin
        if (reset)
            f = startingN;
        else if (enableNext && f > 1)
            begin
                if (0 == (f % 2))
                    f = f / 2;
                else
                    f = 3*f + 1;
            end
    end

endmodule
```

```
module collatz_test(clk, btnU, seg, an);
    wire [31:0]startingN;
    wire [31:0]f;
    wire [15:0]text;
    input clk, btnU;
    output [0:7-1] seg ;
    output [4-1:0] an ;
    wire [15:0] text;

    assign startingN = 'd27;

    collatz DUT(clk, btnU, done, startingN, f);
    binary2bcd BCD(f, text) ;

    display_at_n_second #(1.0) D (clk, btnU, seg, an, text, done);
```

endmodule