

```
1  `timescale 1ns / 1ps
2  /*****
3   * File Name: displayController.v
4   * Project: Counter using TramelBlaze
5   * Designer: Marc Cabote
6   * Email: marcdominic011@gmail.com
7   * Rev. Date: 20 September, 2017
8   *
9   * Purpose: The Display Controller module is a top-level sequential logic circuit
10  *           designed to display a 4-bit hexadecimal input on the Nexys4 DDR seven-
11  *           segment display. This module consists of an LED Clock module used to
12  *           reduce the incoming clock frequency of the Nexys4 DDR board clock, an LED
13  *           Controller module used to cycle between turning on one of eight anodes
14  *           of the seven-segment display, an 8-to-1 Multiplexer module used to select
15  *           which data to display on each anode of the seven-segment display, and a
16  *           Hex-to-7-Segment module used to convert the 4-bit hexadecimal input to a
17  *           7-bit segment output.
18  *
19  * Notes: - This module has a synchronous reset input coming from the AIS0.
20  *****/
21  module displayController(input    clk, rst,
22                          input    [3:0] seg7, seg6, seg5, seg4, seg3, seg2, seg1, seg0,
23                          output   a7, a6, a5, a4, a3, a2, a1, a0,
24                          output   a, b, c, d, e, f, g
25                          );
26
27      wire    clk_out; // Connects led_clk to led_controller
28      wire [2:0] seg_sel; // Connects led_controller to mux_8to1
29      wire [3:0] Y; // Connects mux_8to1 to hex_to_7seg
30
31      wire tick;
32      reg[16:0] count;
33
34
35      // Module instantiations for creating the Display Controller
36      //led_clk          m0 (rst, clk, clk_out);
37
38      //=====
39      // counter to generate 1KHz clock
40      //=====
41      assign tick = (count == 99999); //tick will then be fed to led_controller clock
42      //assign tick = clk;
43      always @ (posedge clk, posedge rst)
44          if (rst) count <= 17'b0; else
45              if (tick) count <= 17'b0; else //check if tick is 1
46                  count <= count + 17'b1; //increment count
47
48
49      // Module instantiations for creating the Display Controller
50      //led_clk          m0 (reset, clk, clk_out);
51
52      led_controller     m1 (tick, rst, a7, a6, a5, a4, a3, a2, a1, a0, seg_sel);
53      mux_8to1           m2 (seg_sel, seg7, seg6, seg5, seg4, seg3, seg2, seg1, seg0, Y);
54      hex_to_7seg        m3 (Y, a, b, c, d, e, f, g);
55
56  endmodule
57
```