```
1   // Eugene Ngo
2   // 1/20/2023
3   // EE 371
4   // Lab 2 Task 3
5
6   // DE1_SoC is the top-level module that defines the I/Os for the DE1 SoC board.
7   // DE1_SoC is the top-level module for the entire FIFO system. It instantiates the FIFO
    module. Additionally, it
8   // connects KEY3 to the read signal, KEY2 to the write signal, and KEY0 to the reset
    signal. It also displays the
9   // content of the input bus in hexadecimal on HEX5 and HEX4, and displays the content of
    the output bus in hexadecimal
10  // on HEX1 and HEX0.
11
12  module DE1_SoC_task3 (HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, KEY, SW, LEDR, CLOCK_50);
13     output logic [6:0]  HEX0, HEX1, HEX2, HEX3, HEX4, HEX5;
14     output logic [9:0]  LEDR;
15     input  logic [3:0]  KEY;
16     input  logic [9:0]  SW;
17     input logic CLOCK_50;
18
19     // turning unused hex displays off
20     assign HEX3 = 7'b1111111;
21     assign HEX2 = 7'b1111111;
22
23     // Setting up buses
24     logic [7:0] inputBus, outputBus;
25     assign inputBus = SW[7:0];
26
27     // Cleaning up button presses to be one clock cycle long.
28     logic reset, read, write;
29     // ip1-ip3 takes the corresponding buttons as inputs, and outputs to intermediate logics
    that will be used by
30     // the FIFO module.
31     input_process ip1 (.A(~KEY[0]), .out(reset), .clk(CLOCK_50));
32     input_process ip2 (.A(~KEY[3]), .out(read), .clk(CLOCK_50));
33     input_process ip3 (.A(~KEY[2]), .out(write), .clk(CLOCK_50));
34
35     // Setting up the drivers for the hex displays.
36     // hex5, hex4, hex1, and hex0 all takes parts of the I/O bus as inputs, and outputs to
    the corresponding hex
37     // displays on the DE1-SoC board.
38     seg7 hex5 (.in(inputBus[7:4]), .out(HEX5));
39     seg7 hex4 (.in(inputBus[3:0]), .out(HEX4));
40     seg7 hex1 (.in(outputBus[7:4]), .out(HEX1));
41     seg7 hex0 (.in(outputBus[3:0]), .out(HEX0));
42
43     // FIFO takes the processed input from the input_processing modules, takes the data from
    the input bus, uses the
44     // 50MHz clock, outputs "full" and "empty" signals to the corresponding LEDs, and
    outputs data onto the output Bus.
45     FIFO queue (.clk(CLOCK_50), .reset, .read, .write, .inputBus, .empty(LEDR[8]), .full(LEDR
    [9]), .outputBus);
46
47  endmodule
48
49  // DE1_SoC_task3_testbench tests all expected, unexpected, and edgecase behaviors
50  `timescale 1 ps / 1 ps
51  module DE1_SoC_task3_testbench();
52     logic [6:0]  HEX0, HEX1, HEX2, HEX3, HEX4, HEX5;
53     logic [9:0]  LEDR;
54     logic [3:0]  KEY;
55     logic [9:0]  SW;
56     logic clk;
57
58     DE1_SoC_task3 dut (.HEX0, .HEX1, .HEX2, .HEX3, .HEX4, .HEX5, .KEY, .LEDR, .SW, .CLOCK_50(
    clk));
59
60     // Setting up logics to make the code more readable.
61     logic reset, read, write;
62     logic [7:0] inputBus;
63     assign KEY[0] = ~reset;
64     assign KEY[3] = ~read;
```

```
65       assign KEY[2] = ~write;
66       assign SW[7:0] = inputBus;
67
68       // Setting up a simulated clock.
69       parameter CLOCK_PERIOD = 100;
70       initial begin
71          clk <= 0;
72          forever #(CLOCK_PERIOD/2) clk <= ~clk; // Forever toggle the clock
73       end
74
75       initial begin
76          reset <= 1; repeat(5) @(posedge clk);
77          // put in excessive data;
78          reset <= 0; read <= 0; write <= 1; inputBus <= 8'h00;          @(posedge clk);
79                                 write <= 0;                             @(posedge clk);
80                                 write <= 1; inputBus <= 8'h01;          @(posedge clk);
81                                 write <= 0;                             @(posedge clk);
82                                 write <= 1; inputBus <= 8'h02;          @(posedge clk);
83                                 write <= 0;                             @(posedge clk);
84                                 write <= 1; inputBus <= 8'h03;          @(posedge clk);
85                                 write <= 0;                             @(posedge clk);
86                                 write <= 1; inputBus <= 8'h04;          @(posedge clk);
87                                 write <= 0;                             @(posedge clk);
88                                 write <= 1; inputBus <= 8'h05;          @(posedge clk);
89                                 write <= 0;                             @(posedge clk);
90                                 write <= 1; inputBus <= 8'h06;          @(posedge clk);
91                                 write <= 0;                             @(posedge clk);
92                                 write <= 1; inputBus <= 8'h07;          @(posedge clk);
93                                 write <= 0;                             @(posedge clk);
94                                 write <= 1; inputBus <= 8'h08;          @(posedge clk);
95                                 write <= 0;                             @(posedge clk);
96                                 write <= 1; inputBus <= 8'h09;          @(posedge clk);
97                                 write <= 0;                             @(posedge clk);
98                                 write <= 1; inputBus <= 8'h0a;          @(posedge clk);
99                                 write <= 0;                             @(posedge clk);
100                                write <= 1; inputBus <= 8'h0b;          @(posedge clk);
101                                write <= 0;                             @(posedge clk);
102                                write <= 1; inputBus <= 8'h0c;          @(posedge clk);
103                                write <= 0;                             @(posedge clk);
104                                write <= 1; inputBus <= 8'h0d;          @(posedge clk);
105                                write <= 0;                             @(posedge clk);
106                                write <= 1; inputBus <= 8'h0e;          @(posedge clk);
107                                write <= 0;                             @(posedge clk);
108                                write <= 1; inputBus <= 8'h0f;          @(posedge clk);
109                                write <= 0;                             @(posedge clk);
110                                // attempt to write more data
111                                write <= 1; inputBus <= 8'h10;          @(posedge clk);
112                                write <= 0;                             @(posedge clk);
113                                write <= 1; inputBus <= 8'h11;          @(posedge clk);
114                                write <= 0;                             @(posedge clk);
115                                write <= 1; inputBus <= 8'h12;          @(posedge clk);
116                                write <= 0;                             @(posedge clk);
117                                write <= 1; inputBus <= 8'h13;          @(posedge clk);
118                                write <= 0;                             @(posedge clk);
119                                write <= 1; inputBus <= 8'h14;          @(posedge clk);
120                                write <= 0;                             @(posedge clk);
121          // attempt to read off 20 sets of data
122                       read <= 1; write <= 0;                           @(posedge clk);
123                       read <= 0;                                       @(posedge clk);
124                       read <= 1;                                       @(posedge clk);
125                       read <= 0;                                       @(posedge clk);
126                       read <= 1;                                       @(posedge clk);
127                       read <= 0;                                       @(posedge clk);
128                       read <= 1;                                       @(posedge clk);
129                       read <= 0;                                       @(posedge clk);
130                       read <= 1;                                       @(posedge clk);
131                       read <= 0;                                       @(posedge clk);
132                       read <= 1;                                       @(posedge clk);
133                       read <= 0;                                       @(posedge clk);
134                       read <= 1;                                       @(posedge clk);
135                       read <= 0;                                       @(posedge clk);
136                       read <= 1;                                       @(posedge clk);
137                       read <= 0;                                       @(posedge clk);
```

```
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);
                  read <= 1;                                          @(posedge clk);
                  read <= 0;                                          @(posedge clk);

      // simultaneous read/write
      reset <= 0; read <= 0; write <= 1; inputBus <= 8'h00;           @(posedge clk);
                             write <= 0;                              @(posedge clk);
                             write <= 1; inputBus <= 8'h01;           @(posedge clk);
                             write <= 0;                              @(posedge clk);
                             write <= 1; inputBus <= 8'h02;           @(posedge clk);
                             write <= 0;                              @(posedge clk);
                             write <= 1; inputBus <= 8'h03;           @(posedge clk);
                             write <= 0;                              @(posedge clk);
                             write <= 1; inputBus <= 8'h04;           @(posedge clk);
                             write <= 0;                              @(posedge clk);
      reset <= 0; read <= 1; write <= 1; inputBus <= 8'h05;           @(posedge clk);
                  read <= 0; write <= 0;                              @(posedge clk);
                  read <= 1; write <= 1; inputBus <= 8'h06;           @(posedge clk);
                  read <= 0; write <= 0;                              @(posedge clk);
                  read <= 1; write <= 1; inputBus <= 8'h07;           @(posedge clk);
                  read <= 0; write <= 0;                              @(posedge clk);
                  read <= 1; write <= 1; inputBus <= 8'h08;           @(posedge clk);
                  read <= 0; write <= 0;                              @(posedge clk);
                  read <= 1; write <= 1; inputBus <= 8'h09;           @(posedge clk);
                  read <= 0; write <= 0;                              @(posedge clk);
      $stop;
   end
endmodule
```