A blue square with white text

Description automatically generated

ELEX 7660: Digital System Design

Lab 1

|  |  |  |
| --- | --- | --- |
| Student Name: Taewoo Kim | Student Number: A01284763 | Set: T |

Table of Contents

[1 Screenshot of Waveforms 3](#_Toc188303008)

[2 Source code of the module 4](#_Toc188303009)

[3 Quartus compilation report 6](#_Toc188303010)

[4 RTL Netlist 7](#_Toc188303011)

Table of Figures

[Figure 1 – Waveform of bcitid module 3](#_Toc188302982)

[Figure 2 - Waveform of decode2 module 3](#_Toc188302983)

[Figure 3 – Waveform of decode7 module 4](#_Toc188302984)

[Figure 4 – Source code of the decode2 module 4](#_Toc188302985)

[Figure 5 - Source code of the decode7 module 5](#_Toc188302986)

[Figure 6 - Source code of the bcitid module 6](#_Toc188302987)

[Figure 7 - Quartus compilation report 6](#_Toc188302988)

[Figure 8 – Final RTL Netlist 7](#_Toc188302989)

# Screenshot of Waveforms



Figure 1 – Waveform of bcitid moduleA screenshot of a computer

Description automatically generated

Figure 2 - Waveform of decode2 module



Figure 3 – Waveform of decode7 module

# Source code of the module

|  |
| --- |
| // decode2.sv  // Description: The decode2 module implements a 2-to-4 decoder.  // author: Taewoo Kim  // date: Jan 14, 2025  module decode2 (input logic [1:0] digit, // 2-bit input digits                  output logic [3:0] ct) ; // 4-bit active low output            // use case statement to map the input number to the corresponding  // ct (output)          always\_comb begin              case(digit)                  2'b00 : ct = 4'b1110; // Activate output 0 (active low)                  2'b01 : ct = 4'b1101; // Activate output 1 (active low)                  2'b10 : ct = 4'b1011; // Activate output 2 (active low)                  2'b11 : ct = 4'b0111; // Activate output 3 (active low)              endcase          end    endmodule |

Figure 4 – Source code of the decode2 module

|  |
| --- |
| // decode7.sv  // Decription: The decode7 module converts any 4 bit number num (0, 1, 2, ...E, F)  //             the signals necessary to control the 7-segment display.  // author: Taewoo Kim  // date: Jan 14, 2025  module decode7 (input logic [3:0] num,      // 4-bit input number                  output logic [7:0] leds) ;  // 7-segment LED cathods          // use case statement to map the input number to the corresponding          // 7-seg display          always\_comb begin              case(num)                  4'h00 : leds = 8'h3F;       // display 0                  4'h01 : leds = 8'h06;       // display 1                  4'h02 : leds = 8'h5B;       // display 2                  4'h03 : leds = 8'h4F;       // display 3                  4'h04 : leds = 8'h66;       // display 4                  4'h05 : leds = 8'h6D;       // display 5                  4'h06 : leds = 8'h7D;       // display 6                  4'h07 : leds = 8'h07;       // display 7                  4'h08 : leds = 8'h7F;       // display 8                  4'h09 : leds = 8'h67;       // display 9                  4'h0A : leds = 8'h77;       // display A                  4'h0B : leds = 8'h7C;       // display b                  4'h0C : leds = 8'h39;       // display C                  4'h0D : leds = 8'h5E;       // display d                  4'h0E : leds = 8'h79;       // display E                  4'h0F : leds = 8'h71;       // display F              endcase          end  endmodule |

Figure 5 - Source code of the decode7 module

|  |
| --- |
| // bcitid.sv  // Description: The bcitid module implements a 4x4 bit memory  //              that will store the last four digits of our BCIT student ID.  // author: Taewoo Kim  // date: Jan 14, 2025  module bcitid (input logic [1:0] digit, // 2-bit input digit                  output logic [3:0] idnum); // 4-bit output for id #        // Use case statement to map the input digit to the corresponding      // student ID number      always\_comb begin          case (digit)               2'b11: idnum = 4'h4; // Leftmost digit               2'b10: idnum = 4'h7; // Third digit               2'b01: idnum = 4'h6; // Second digit               2'b00: idnum = 4'h3; // Rightmost digit          endcase      end    endmodule |

Figure 6 - Source code of the bcitid module

# Quartus compilation report

A screenshot of a computer

Description automatically generated

Figure 7 - Quartus compilation report

# RTL Netlist

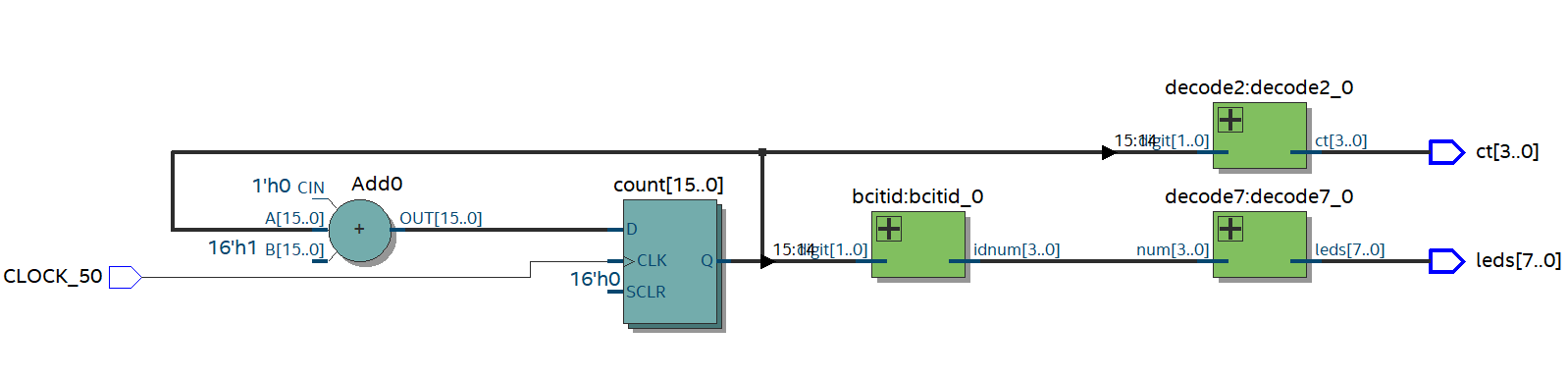


Figure 8 – Final RTL Netlist