



ELEX 7660: Digital System Design

Lab 1

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

Table of Contents

1	Screenshot of Waveforms	3
2	Source code of the module	4
3	Quartus compilation report	6
4	RTL Netlist.....	7

Table of Figures

Figure 1 – Waveform of bctid module	3
Figure 2 - Waveform of decode2 module	3
Figure 3 – Waveform of decode7 module	4
Figure 4 – Source code of the decode2 module	4
Figure 5 - Source code of the decode7 module	5
Figure 6 - Source code of the bctid module	6
Figure 7 - Quartus compilation report	6
Figure 8 – Final RTL Netlist	7

1 Screenshot of Waveforms



Figure 1 – Waveform of *bctid* module

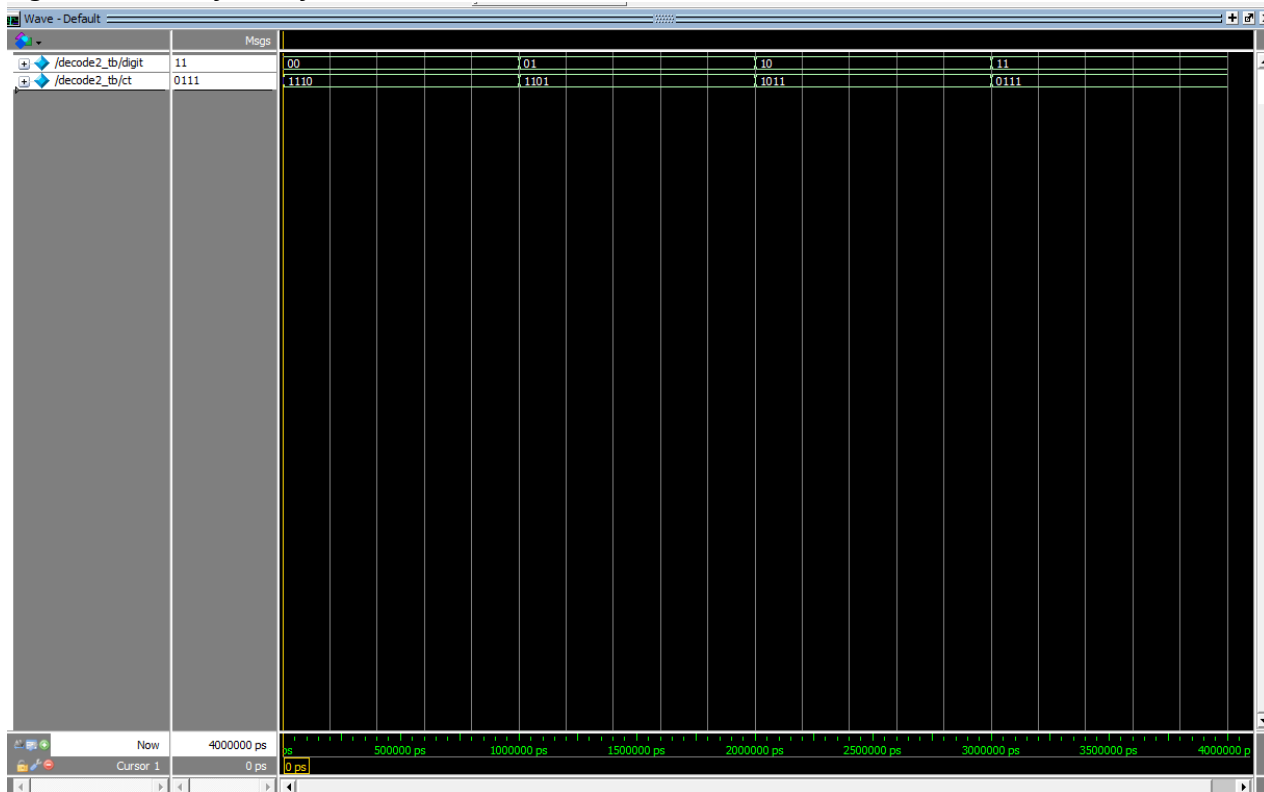


Figure 2 - Waveform of *decode2* module



Figure 3 – Waveform of decode7 module

2 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
// Description: 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

```
// bctid.sv
// Description: The bctid 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 bctid (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 bctid module

3 Quartus compilation report

The screenshot displays the Quartus Prime Lite Edition interface with the compilation report for 'lab1.sv' open. The report is titled 'Flow Summary' and shows a successful compilation status. Key details include:

- Flow Status:** Successful - Mon, Jan 20 22:01:49 2025
- Quartus Prime Version:** 18.1.0 Build 625 09/12/2018 S.J. Lite Edition
- Revision Name:** lab1
- Top-level Entity Name:** lab1
- Family:** Cyclone V
- Device:** 5CSEMA4U23C6
- Timing Models:** Final
- Logic utilization (in ALMs):** 11 / 15,880 (< 1 %)
- Total registers:** 17
- Total pins:** 13 / 314 (4 %)
- Total virtual pins:** 0
- Total block memory bits:** 0 / 2,764,800 (0 %)
- Total DSP blocks:** 0 / 84 (0 %)
- Total HSSI RX PCSs:** 0
- Total HSSI PHA RX Deserializers:** 0
- Total HSSI TX PCSs:** 0
- Total HSSI PHA TX Serializers:** 0
- Total PLLs:** 0 / 5 (0 %)
- Total DLLs:** 0 / 4 (0 %)

The bottom section of the report shows timing analysis results, including messages about clock uncertainty, setup/hold slack, and recovery paths. The final message states: 'Quartus Prime Timing Analyzer was successful. 0 errors, 6 warnings' and '293000 Quartus Prime Full compilation was successful. 0 errors, 148 warnings'.

Figure 7 - Quartus compilation report

4 RTL Netlist

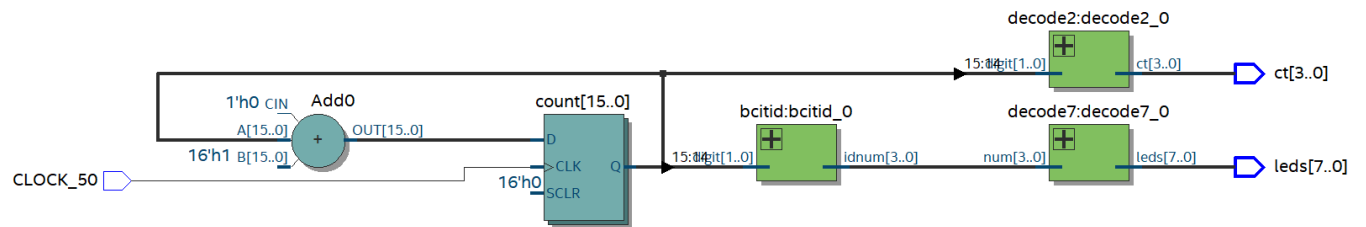


Figure 8 – Final RTL Netlist