Combinational Logic Design: Part 8: BCD to 7-segment display decoder

Author Adil R.

May 2025

1 Introduction

A BCD to 7-segment display decoder is a combinational circuit that converts a Binary Coded Decimal (BCD) input to the appropriate outputs for driving a 7-segment display. Each segment of the display (labeled a through g) is illuminated based on the 4-bit BCD input (0000 to 1001, representing digits 0-9). This report presents the design and implementation of such a decoder using Verilog HDL, along with its test bench and simulation results.

2 Design Description

The 7-segment display consists of seven LEDs arranged in a specific pattern that can display decimal digits when different segments are illuminated. The decoder takes a 4-bit BCD input and produces 7 outputs that control the segments.

- Input: 4-bit BCD (D, C, B, A) where D is MSB
- Output: 7-bit pattern (a, b, c, d, e, f, g) for segment control
- Active-low outputs (common cathode display)

2.1 Truth Table

D	C	B	A	a	b	c	d	e	f	g
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	0	1	1	0
0	1	0	0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0	1	0	0
0	1	1	0	0	1	0	0	0	0	0
0	1	1	1	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	1	0	0

Table 1: Truth table for BCD to 7-segment decoder (active-low outputs)

3 Verilog Implementation

```verilog
`timescale 1ns / 1ps


module bcd_to_7seg(
    input [3:0] bcd,
    output reg [6:0] seg  // [a, b, c, d, e, f, g]
);

always @(*) begin
    case(bcd)
        4'b0000: seg = 7'b0000001; // 0
        4'b0001: seg = 7'b1001111; // 1
        4'b0010: seg = 7'b0010010; // 2
        4'b0011: seg = 7'b0000110; // 3
        4'b0100: seg = 7'b1001100; // 4
        4'b0101: seg = 7'b0100100; // 5
        4'b0110: seg = 7'b0100000; // 6
        4'b0111: seg = 7'b0001111; // 7
        4'b1000: seg = 7'b0000000; // 8
        4'b1001: seg = 7'b0000100; // 9
        default: seg = 7'b1111111; // off
    endcase
end

endmodule
```

4 Test Bench

```verilog
`timescale 1ns / 1ps


module tb_bcd_to_7seg;

    reg [3:0] bcd;
    wire [6:0] seg;

    bcd_to_7seg uut (
        .bcd(bcd),
        .seg(seg)
    );

    initial begin
        $monitor("Time = %0t, BCD = %b, Segments = %b",
                $time, bcd, seg);

        bcd = 4'b0000; #10; // 0
        bcd = 4'b0001; #10; // 1
        bcd = 4'b0010; #10; // 2
        bcd = 4'b0011; #10; // 3
        bcd = 4'b0100; #10; // 4
        bcd = 4'b0101; #10; // 5
        bcd = 4'b0110; #10; // 6
        bcd = 4'b0111; #10; // 7
        bcd = 4'b1000; #10; // 8
        bcd = 4'b1001; #10; // 9
        bcd = 4'b1010; #10; // Invalid

        $finish;
    end

endmodule
```

5 Conclusion

- The BCD to 7-segment decoder was successfully designed and implemented in Verilog HDL.
- The test bench verified all possible BCD inputs (0-9) and confirmed the correct segment patterns were generated.

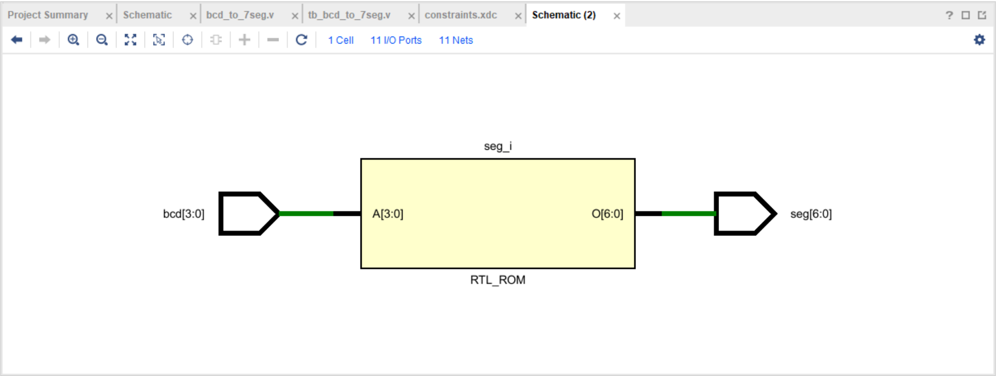

Figure 1: Schematic of the BCD to 7-segment decoder

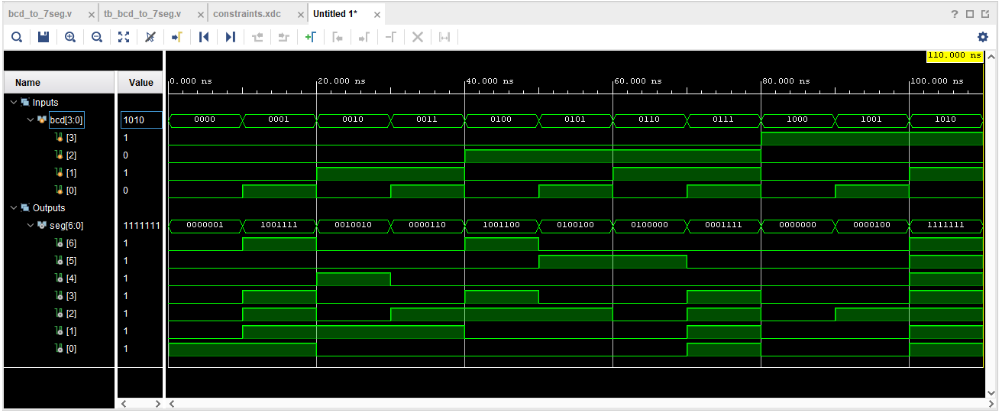

Figure 2: Simulation waveform showing BCD inputs and 7 segment outputs