**List of contents:**

1. Lab Description

* Problem statement
* 7-Segment Display of Hexadecimal Numbers

1. Verilog Code

* Source code
* Testbench

1. Simulation results

* Waveforms
* Summary table

1. Synthesis results

* NEXYS4 DDR board results
* Summary Table

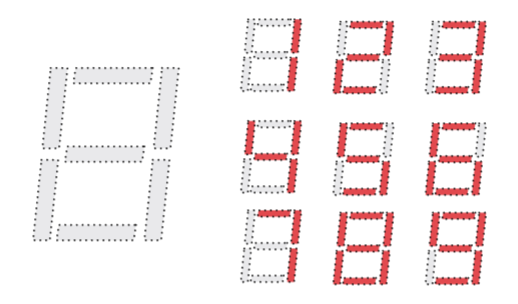
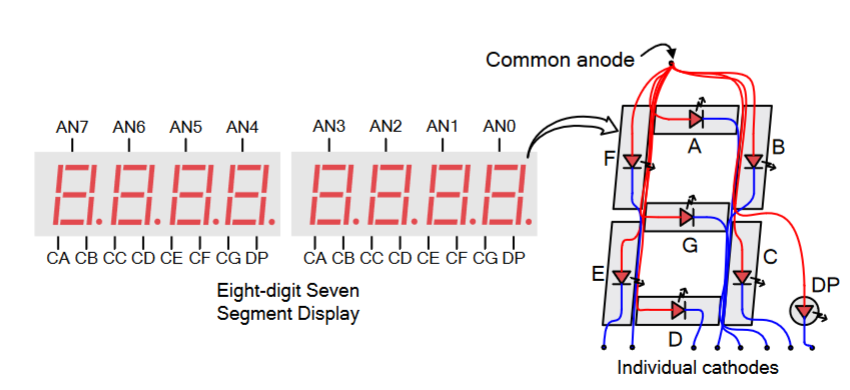
1. **Lab description:**

* **Problem statement:**

To write Verilog code for the 7-Segment Display of Hexadecimal Number using any modelling style, run the code and verify simulation results, implement it on the NEXYS4 DDR board and verify the synthesized results

* **Ripple carry adder operation:**

7-Segment Display of Hexadecimal Number is used to display any given binary or decimal number in hexadecimal form. The Nexys4 DDR board contains two four-digit common anode seven-segment LED displays, configured to behave like a single eight-digit display. Each of the eight digits is composed of seven segments arranged in a figure pattern, with an LED embedded in each segment. Segment LEDs can be individually illuminated, so any one of 128 patterns can be displayed on a digit by illuminating certain LED segments and leaving the others dark

****

The anodes of the seven LEDs forming each digit are tied together into one “common anode” circuit node, but the LED cathodes remain separate, as shown in above figure. The common anode signals are available as eight “digit enable” input signals to the 8-digit display. The cathodes of similar segments on all four displays are connected into seven   
circuit nodes labeled CA through CG. For example, the eight “D” cathodes from the eight digits are grouped together into a single circuit node called “CD.” These seven cathode signals are available as inputs to the 8-digit display. This signal connection scheme creates a multiplexed display, where the cathode signals are common to all digits but they can only illuminate the segments of the digit whose corresponding anode signal is asserted.   
To illuminate a segment, the anode should be driven high while the cathode is driven low. However, since the Nexys4 DDR uses transistors to drive enough current into the common anode point, the anode enables are inverted. Therefore, both the AN0..7 and the CA..G/DP signals are driven low when active.

1. **Verilog code:**

* **Source code:**

module segment(input [3:0] i, output reg [7:0] a, output reg [6:0] z);

always @(\*)

case (i)

4'b0000 :  begin a=8'b11111110; z = 7'b1000000; end

4'b0001 : begin a=8'b11111110; z = 7'b1111001; end

4'b0010 : begin a=8'b11111110; z = 7'b0100100; end

4'b0011 : begin a=8'b11111110; z = 7'b0110000; end

4'b0100 :  begin a=8'b11111110; z = 7'b0011001; end

4'b0101 :  begin a=8'b11111110; z = 7'b0010010; end

4'b0110 :  begin a=8'b11111110; z = 7'b0000010; end

4'b0111 :  begin a=8'b11111110; z = 7'b1111000; end

4'b1000 : begin a=8'b11111110; z = 7'b0000000; end

4'b1001 : begin a=8'b11111110; z = 7'b0010000; end

4'b1010 : begin a=8'b11111110; z = 7'b0001000; end

4'b1011 : begin a=8'b11111110; z = 7'b0000011; end

4'b1100 :  begin a=8'b11111110; z = 7'b1000110; end

4'b1101 :  begin a=8'b11111110; z = 7'b0100001; end

4'b1110 :  begin a=8'b11111110; z = 7'b0000110; end

4'b1111 :  begin a=8'b11111110; z = 7'b0001110; end

endcase

endmodule

* **Testbench**

module segment\_tb();

  reg [3:0] i;

  wire [7:0] a;

    wire [6:0] z;

    segment segment7(.i(i), .z(z));

    initial begin

     i = 0;       #20 i = 1;

      #20 i = 2;      #20 i = 3;

      #20 i = 4;      #20 i = 5;

      #20 i = 6;      #20 i = 7;

      #20 i = 8;      #20 i = 9;

      #20 i = 10;     #20 i = 11;

      #20 i = 12;     #20 i = 13;

      #20 i = 14;     #20 i = 15;

      #40;

end

    initial begin

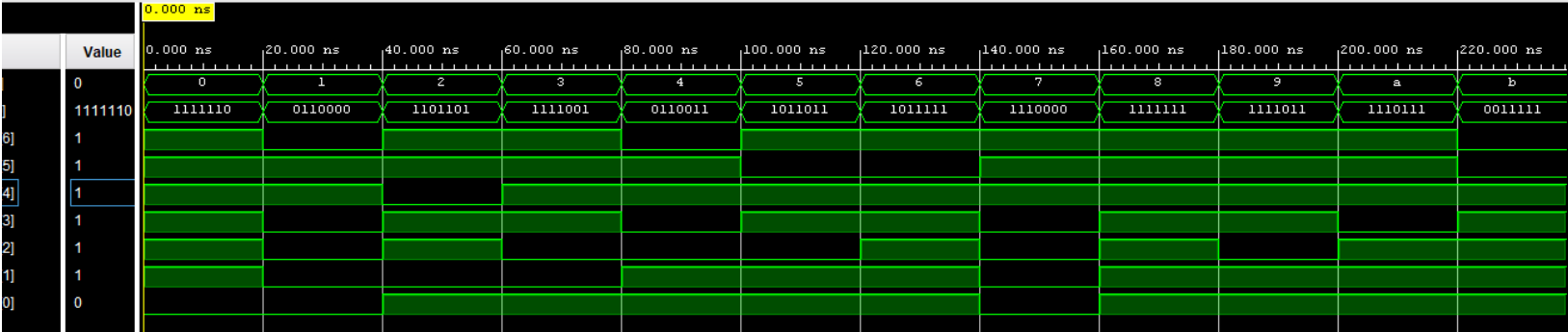
$monitor("i=%h,z=%7b",i,z);

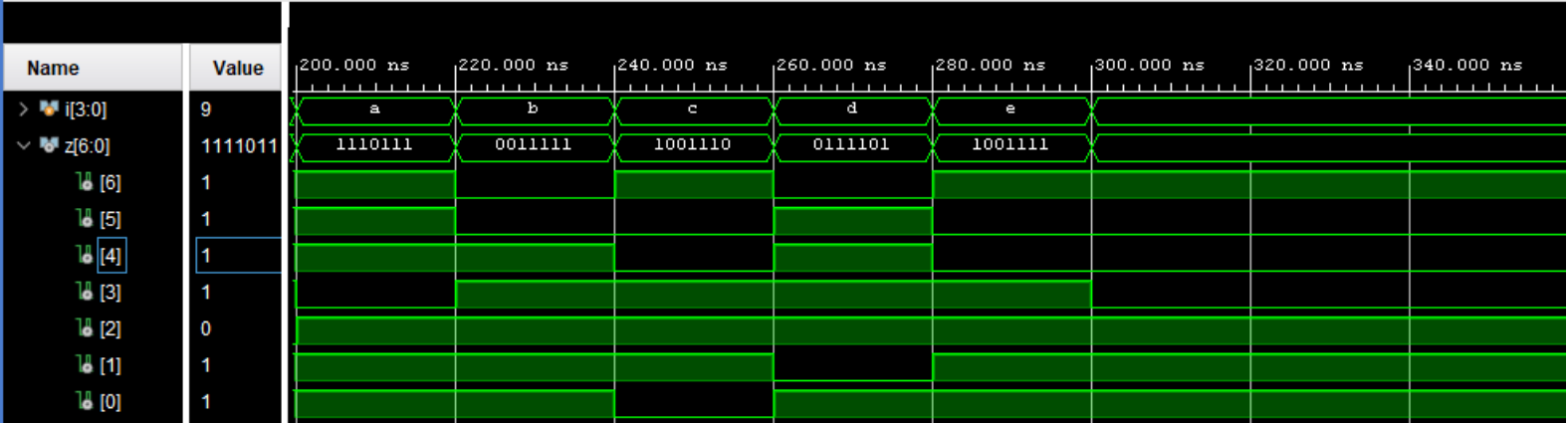
end

endmodule

1. **Simulation results:**

* **Waveforms:**

****

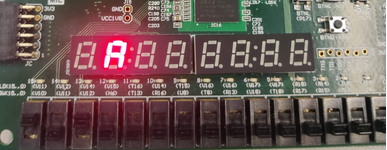
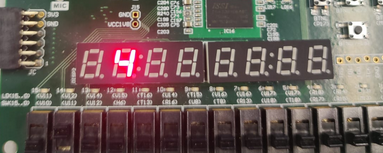
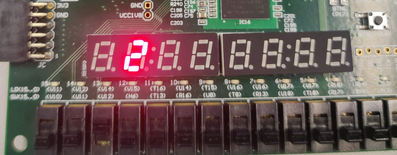
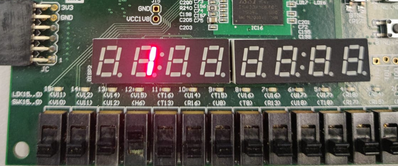
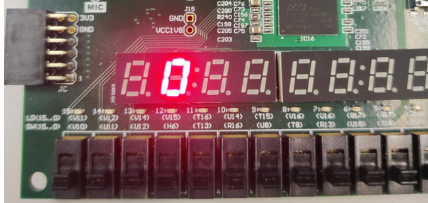
****

* **Summary Table:**

|  |  |  |
| --- | --- | --- |
| **i** | **z** | **Hex\_conversion** |
| 0 | 1000000 | 0 |
| 1 | 1111001 | 1 |
| 2 | 0100100 | 2 |
| 3 | 0110000 | 3 |
| 4 | 0011001 | 4 |
| 5 | 0010010 | 5 |
| 6 | 0000010 | 6 |
| 7 | 1111000 | 7 |
| 8 | 0000000 | 8 |
| 9 | 0010000 | 9 |
| 10 | 0001000 | A |
| 11 | 0000011 | B |
| 12 | 1000110 | C |
| 13 | 0100001 | D |
| 14 | 0000110 | E |
| 15 | 0001110 | F |

1. **Synthesis results**

* **FPGA4 DDR board results:**



* **Summary Table:**

|  |  |  |
| --- | --- | --- |
| **i** | **z** | **Hex\_conversion\_display** |
| 0 | 1000000 | 0 |
| 1 | 1111001 | 1 |
| 2 | 0100100 | 2 |
| 3 | 0110000 | 3 |
| 4 | 0011001 | 4 |
| 7 | 1111000 | 7 |
| 10 | 0001000 | A |
| 12 | 1000110 | C |
| 15 | 0001110 | F |