**Lab 5**

**Objective:**

To design and test a circuit to display hex values on a seven-segment display, and integrate it with the previous projects 3-bit full adder.

**Design:**

Formulas:

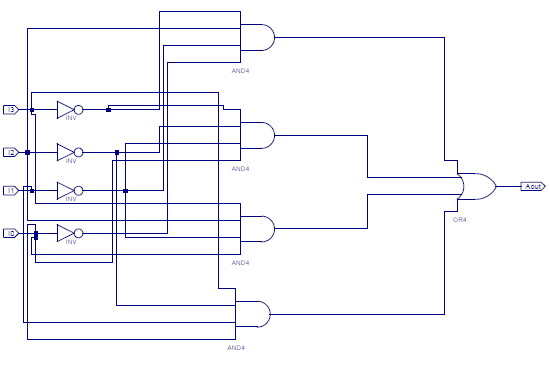
Circuits: 

Figure - Segment A Circuit

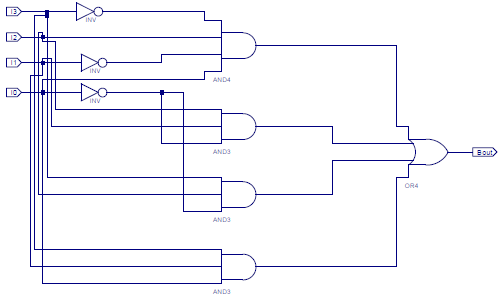


Figure - Segment B Circuit

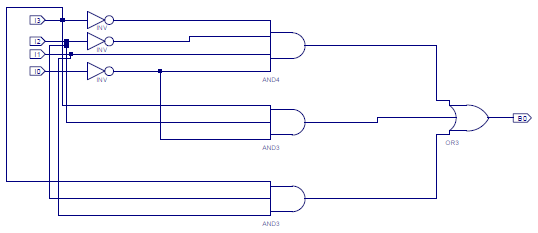


Figure - Segment C Circuit

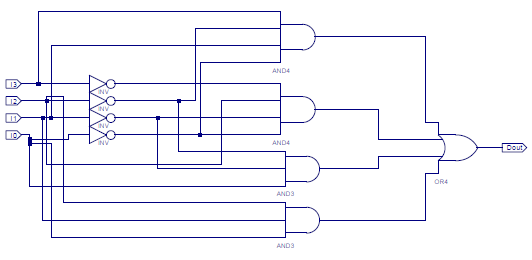


Figure - Segment D Circuit

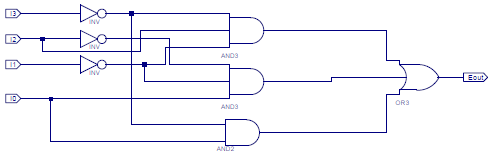


Figure - Segment E Circuit

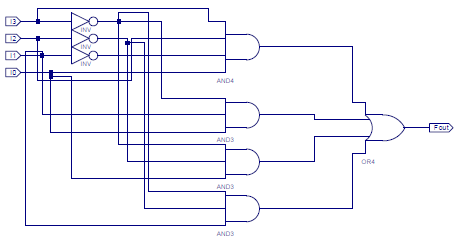


Figure - Segment F Circuit

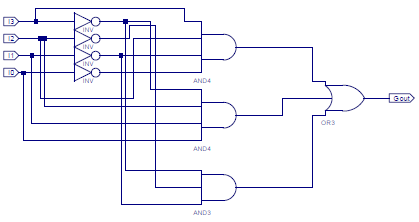


Figure - Segment G Circuit

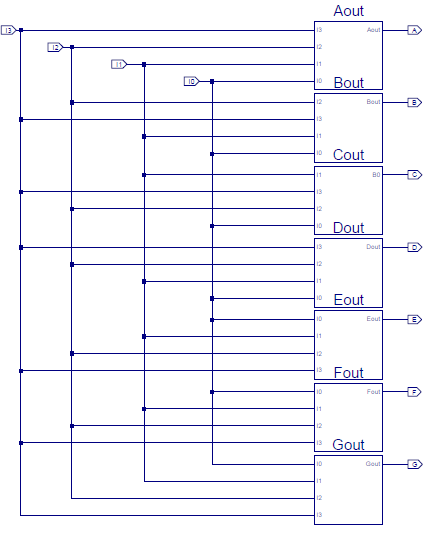


Figure - Seven Segment Decoder Circuit

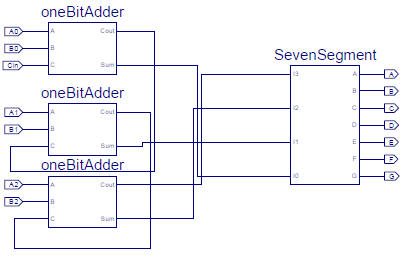


Figure - Seven Segment Decoder Connected to Three-Bit Adder

Tables:



Table - Seven Segment Truth Table



Table - Test Vectors for Post-Route Simulation of the Seven Segment Display



Table - Decoder with Adder Test Vectors

**Procedure:**

Part 1 – Seven Segment Display:

* Create schematic files for segments A-G according for prelab formulas, as in previous labs
  + Results in Figures 1-7
* Make schematic symbols for each segment as in the previous lab
  + Highlight the schematic file from the design window
  + Run “Create Schematic Symbol” in the process window
* Make a new schematic symbol compiling all 7 segment circuits with the same inputs
  + Results in Figure 8
* Run Behavioral Simulation for all 16 inputs in the 7 segment truth table (Table 1)
  + Print results (Figure 10)
* Run Post-Route Simulation to see the time delays for provided test vectors (Table 2)
  + Print results (Figure 11)
* Download to the FPGA as in previous labs
  + Connect the 4 inputs to switches
  + Connect the 7 outputs to the appropriate segments on the FPGA’s seven segment display
    - A to CA (L18)
    - B to CB (F18)
    - Etc. for remaining outputs C through G
* Test for all 16 inputs in the 7 segment truth table (Table 1)
* Create a schematic symbol for the 7-segment display circuit
  + Highlight the schematic file from the design window
  + Run “Create Schematic Symbol” in the process window

Part 2 – Seven Segment Display with Three-Bit Adder:

* Copy the schematic files from Lab 4
  + Open Lab 4 project file
  + Copy the .sch files for the onebitadder and threebitadder
  + Open Lab5 project file
  + Past the .sch files that were copied
* Use “Add Source” to add the two schematic files to the Lab5 project
* Run “Create Schematic Symbol” for the one-bit adder
* Set the three-bit adder to the top level
* Open the three-bit adder schematic file
  + Remove the output markers
  + Add a 7-segment symbol
  + Connect the final Cout from the three-bit adder to the most significant bit of the 7-segment symbol’s inputs (I3)
  + Connect the remaining three-bit adder outputs to the appropriate 7-segment inputs
    - Sum2 -> I2
    - Sum1 -> I1
    - Sum0 -> I0
  + Place output markers on the 7-Segment symbol
  + Results in Figure 9
* Save and Synthesize the schematic file
* Download to the FPGA as in previous labs
  + Connect the 7 inputs to switches
  + Connect the outputs as in the previous part
    - A to CA (L18)
    - B to CB (F18)
    - Etc. for remaining outputs C through G
* Test for using the test vectors provided (Table 3)

**Data:**

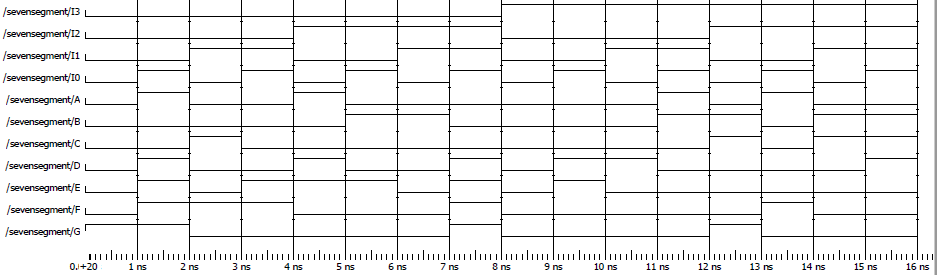


Figure - Behavioral Simulation for Seven Segment Display

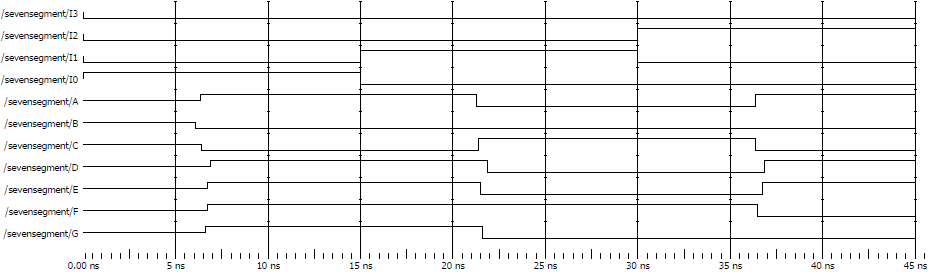


Figure - Simulation with Time Delay for Seven Segment Display

**Data Analysis:**

There were some mistakes made in my original function for my D segment, which resulted in incorrect outputs in my original Behavioral Simulations. Once corrected, the simulations all resulted in the expected truth tables and outputs. There were later mistakes when downloading the circuits to the FPGA that mainly reemphasized how simple it is to make a mistake as the circuit complexity increases. One mistake was connecting the adder’s Cout to I2 instead of I3, another was connecting the circuit inputs to the FPGA LED’s instead of switches. Once these mistakes were corrected, the circuit gave the appropriate readings corresponding to the respective truth tables (Tables 1 & 3).

**Question:** “In your report state your observations of time delays in the seven segment decoder.”

As noted in previous labs, there is a measurable delay between the input changing and the output reacting. With there being a larger number of outputs, it is easier to see that the delay is not uniform across all circuits. While all of the outputs for these test vectors still took approximately 6ns as in previous labs, it can be seen that some segments had shorter delays than others. For example, with the first test input: B switched sooner than A and C, which switched faster than E, F, and G, which switched faster than D.

**Conclusion:**

As with the previous lab, this lab demonstrates the importance of modular design when it comes to increasingly complex circuit design. With each output being an independent module, it was much simpler to troubleshoot mistakes that were made, such as my bad original formula for segment D or my misconnected inputs in part 2. It also shows the benefit of using modules in future projects, taking connecting the three-bit adder to the seven segment display brings it closer to a more user-friendly product rather than a very specific, purpose built design.