Patrick Austin

CPE 301 – 1104

Assignment # 11

12/9/2016

Assignment description:

In this lab we programmed, uploaded, and tested two Verilog programs to the provided Altera FPGA board. One program mapped from two input switches to five LEDs on the board, denoting the result of AND, NAND, OR, NOR, and XNOR on the switch values. The other program mapped from two input switches and an enable switch onto a seven-segment LED in 2-4 decoder fashion- ie the LED showed values 0 (denoting enable off) and 1-3 (denoting enable on and decoder results 1-4.

Problems encountered:

This lab was quite straightforward. The tutorial provided was very useful and complete, and the first program simply involved following its instructions, with one exception. One of the output pin mappings specified in the lab document pointed to an erroneous location, which we fixed by looking at the papers included in the FPGA box for the appropriate pin. The second program took a little elbow grease to figure out, but no serious roadblocks were encountered.

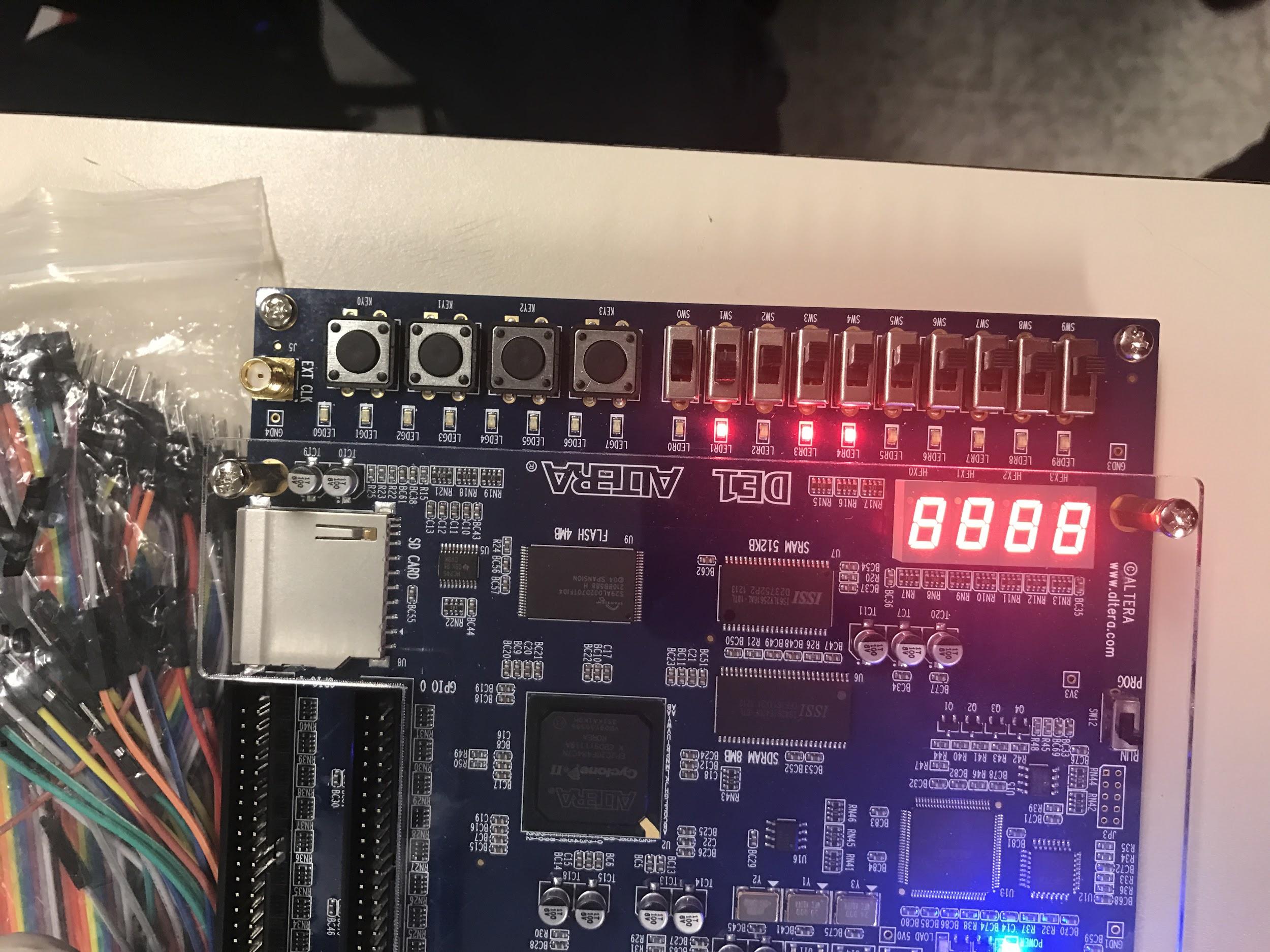
Lessons learned:

This was quite a neat lab, and I think it succinctly conveyed how one works with an FPGA board using Verilog. I was prepared to be overwhelmed by complexity, but the lab document kept things pretty simple for which I was grateful and the relative simplicity with which the board ‘just worked’ was neat to see. One major lesson was to carefully map pins, as even a correct Verilog program can look wrong if it’s going to inappropriate pins.

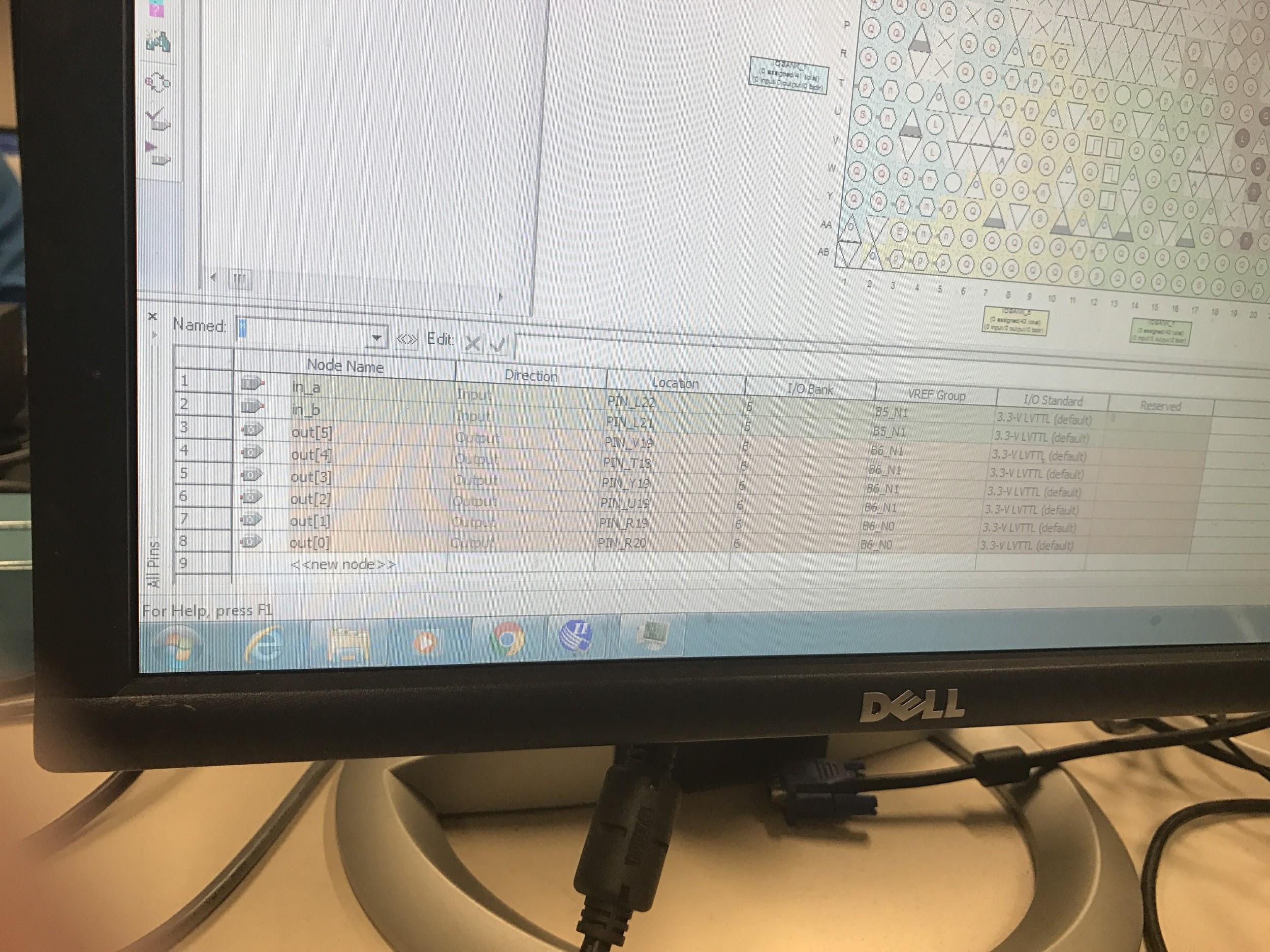
Description of Completed Lab:

Program 1

Shown running on the board here:

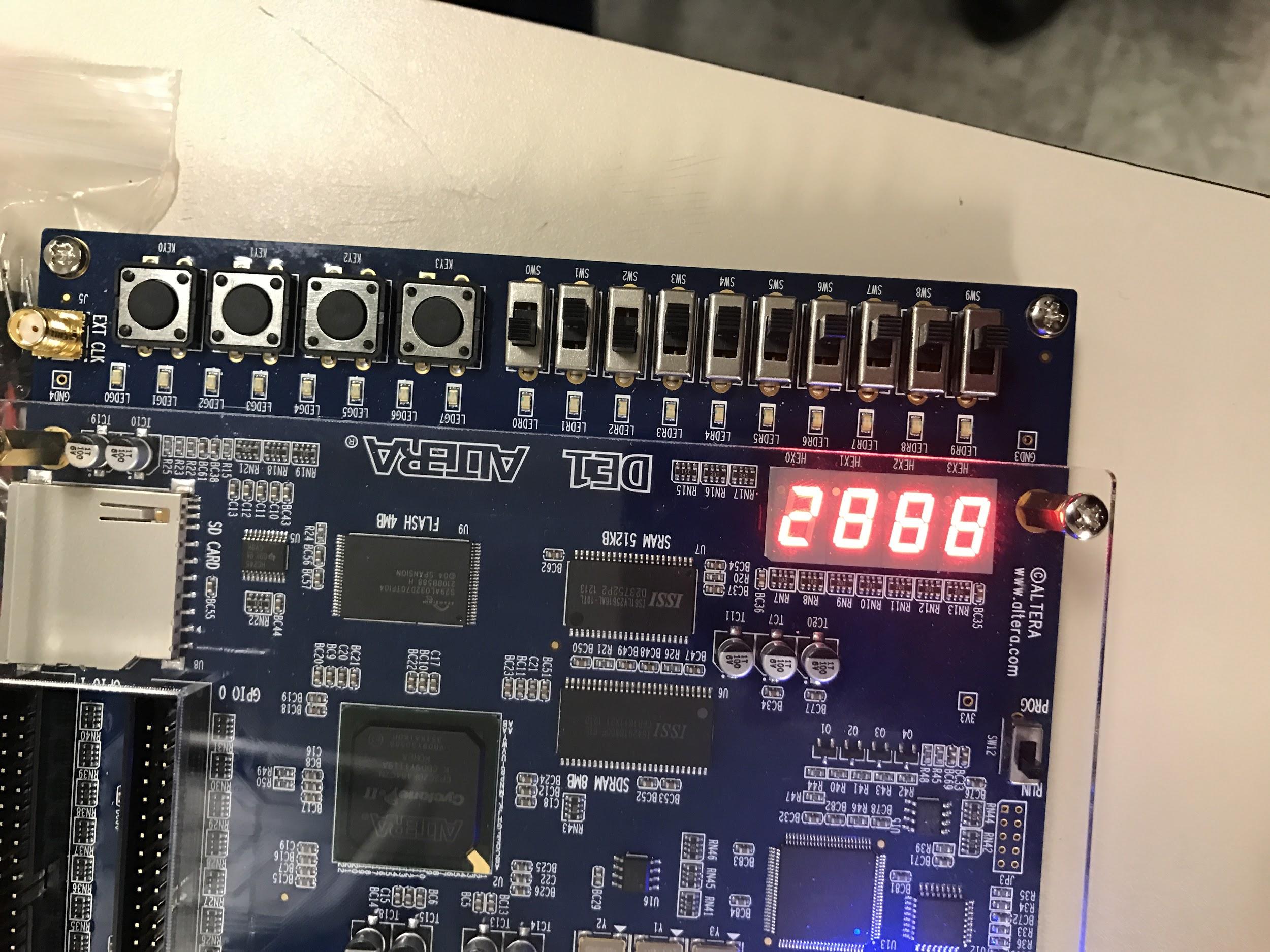


Fixed pin mapping shown here:



Program 2

Program shown running on the board here, with enable on, in\_a on, in\_b off, appropriately displaying ‘2’ on the seven segment:



Here is the specified source code:

module gate2 (in\_a, in\_b, in\_enable, out);  
input wire in\_a;  
input wire in\_b;  
input wire in\_enable;  
output wire [6:0] out; //only segments 0-7 used  
  
assign out[6] = ~( ( in\_enable & (in\_a & ~in\_b) ) | ( in\_enable & (~in\_a & in\_b) ) |

( in\_enable & (in\_a & in\_b) ) );

assign out[5] = ~((~in\_enable) | ( in\_enable & (in\_a & in\_b) ));  
assign out[4] = ~((~in\_enable) | ( in\_enable & (in\_a & ~in\_b) ));  
assign out[3] = ~((~in\_enable) | ( in\_enable & (in\_a & ~in\_b )) | ( in\_enable & (~in\_a & in\_b) ) );  
assign out[2] = ~((~in\_enable) | ( in\_enable & (~in\_a & ~in\_b)) | ( in\_enable & (~in\_a & in\_b) ) |

( in\_enable & (in\_a & in\_b) ));

assign out[1] = ~((~in\_enable) | ( in\_enable & (~in\_a & ~in\_b)) | ( in\_enable & (in\_a & ~in\_b) ) |

( in\_enable & (~in\_a & in\_b) ) | ( in\_enable & (in\_a & in\_b) ));

assign out[0] = ~((~in\_enable) | ( in\_enable & (in\_a & ~in\_b)) | ( in\_enable & (~in\_a & in\_b) ) );

endmodule

The specified truth table, boolean equations for each segment, and table reflecting pin assignments:

