Lab 2
Tiffany-Ellen Vo
January 19, 2020
Section C: Tues/Thurs 1:30PM-3:45PM

Description:

The purpose of this lab is to learn how to use hierarchies in designs by using modules, how to use a Verilog testbench, and also how to simulate a design with the Vivado Simulator. This lab also introduces students to the 7-segment display and how to implement that.

Methods:

The first part was figuring out the syntax and understanding how to implement hierarchies. I followed the article on how to do this part, which helped a lot. Afterward, three full adders were combined in the top-level module.

Going into the 7-segment display part, the first step was to create a truth table for the different displays. This part was a bit hard in understanding the different pin numbers used, and how to do the truth table. Once that was created, the sum of products operation was used for each display, in hex, in order to create the numbers on the board. These equations were then coded in the 7 segment converter module. Afterward, the 7 segment converter was added to the top-level module.

The next part was to download the Verilog testbench and add more tests for eleven through fifteen. Once that was done and I read through the basic commands on how to control the simulator, I ran the simulation for the testbench, and also displayed the four wires connecting the adder created to the 7-segment converter, without making them pins of the top-level module. After looking through the simulation, I implemented my design, configured the FPGA and demonstrated the design to a TA.

Results:

Lab Questions:

The longest path for any input to any output in the 3-bit adder is going through either a or b into any adder; if you go through the not gate, the number of gates the input goes through before reaching an output is 6. For the longest path for any input to any output for an n-bit adder, the formula found is $2^n - 2$.

For documenting which pins of the FPGA used and how they were connected to the switches and 7-segment display was found in the constraint file. For the 7 segment display: ports ca (pin W7), cb (pin W6) cc (pin U8), cd (pin V8), ce (pin U5), cf (pin V5), cg (pin U7), dp (pin V7), an0 (pin U2), an1 (pin U4), an2 (pin V4), an3 (pin W4) were used. For the switches, sw0 (pin V17), sw1 (pin V16), sw2 (pin W16), sw3 (pin W17), sw4 (pin W15), sw5 (pin V15), sw6 (pin W14) were used.

For calculating the number of possible input values, that would be 2^7 because of the input values being 7-bit vectors for the adder. This equals 128 possible input values. In the testbench for the simulation, 16 hex values were tested. This means the percentage of the possible input values that were tested is 12.5%.

Conclusion:

During this lab, I learned about how to use the simulator. This is especially helpful for debugging and testing to see if the outputs will be what is expected before taking the time to generate a bitstream. It was also interesting to learn about how to create hierarchies in vivado, since it makes modules seem more organized to me. I also learned about how a 7 segment display works, which is a more detailed learning experience than something similar in CSE 12, which I also found interesting.

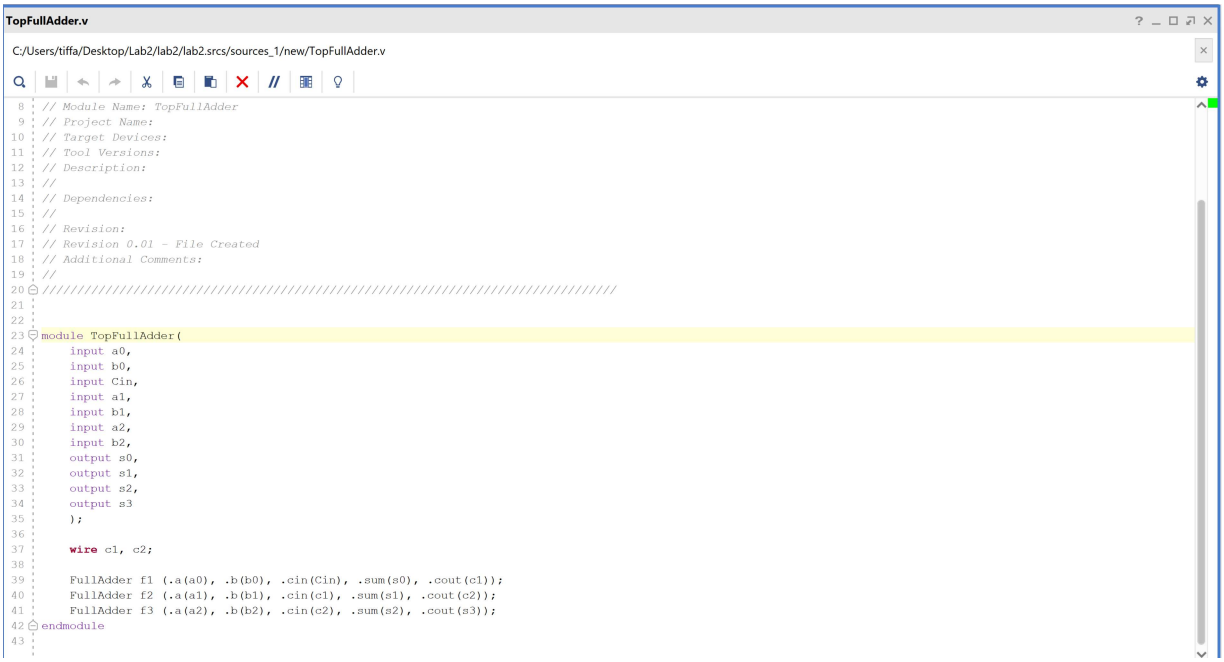
Some mistakes I have made during the lab was assuming that the equations that we were supposed to find for the 7 segment display were one giant equation instead of separate equations for a through f. I also had trouble understanding how to add onto the test bench.

Another issue I had encountered was that after archiving my lab, I tried to open the same zipped file on my laptop to work on the lab report. However, my lab testbench file was lost. However, I did make a testbench, and also had help from the TA, Thomas, of section C about how to add more tests for numbers eleven through fifteen. I included a screenshot of the lab 2 test module in the appendices, specifically the part for testing.

Appendices:

Print out of all modules:

The module of the three full adders:



The Module of one of the full adders:

FullAdder.v

C:/Users/tiffa/Desktop/Lab2/lab2/lab2.srcs/sources_1/new/FullAdder.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 01/14/2020 01:54:30 PM
// Design Name:
// Module Name: FullAdder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module FullAdder(
    input a,
    input b,
    input cin,
    output sum,
    output cout
    );

    assign sum = (~a & ~b & cin) | (~a & b & ~cin) | (a & ~b & ~cin) | (a & b & cin); // sop of full adder
    assign cout = (cin & (a ^ b) | a & b);
endmodule

```

The Module for the seven-segment display:

SevenSegment.v

C:/Users/tiffa/Desktop/Lab2/lab2/lab2.srcs/sources_1/new/SevenSegment.v

```
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module SevenSegment(
    input n0,
    input n1,
    input n2,
    input n3,
    output a,
    output b,
    output c,
    output d,
    output e,
    output f,
    output g
    );

    assign a = ((~n3&~n2&~n1&n0) | (~n3&n2&~n1&~n0) | (n3&~n2&n1&n0) | (n3&n2&~n1&n0));
    assign b = ((~n3&n2&~n1&n0) | (~n3&n2&n1&~n0) | (n3&~n2&n1&n0) | (n3&n2&~n1&~n0) | (n3&n2&n1&~n0) | (n3&n2&n1&n0));
    assign c = ((~n3&~n2&n1&~n0) | (n3&n2&~n1&~n0) | (n3&n2&n1&~n0) | (n3&n2&n1&n0));
    assign d = ((~n3&~n2&~n1&n0) | (~n3&n2&~n1&~n0) | (~n3&n2&n1&n0) | (n3&~n2&n1&~n0) | (n3&n2&n1&n0));
    assign e = ((~n3&~n2&~n1&n0) | (~n3&~n2&n1&n0) | (~n3&n2&~n1&~n0) | (~n3&n2&~n1&n0) | (~n3&n2&n1&n0) | (n3&~n2&~n1&n0));
    assign f = ((~n3&~n2&~n1&n0) | (~n3&~n2&n1&~n0) | (~n3&~n2&n1&n0) | (~n3&n2&n1&n0) | (n3&n2&~n1&n0));
    assign g = ((~n3&~n2&~n1&~n0) | (~n3&~n2&~n1&n0) | (~n3&n2&n1&n0) | (n3&n2&~n1&~n0));
endmodule

```

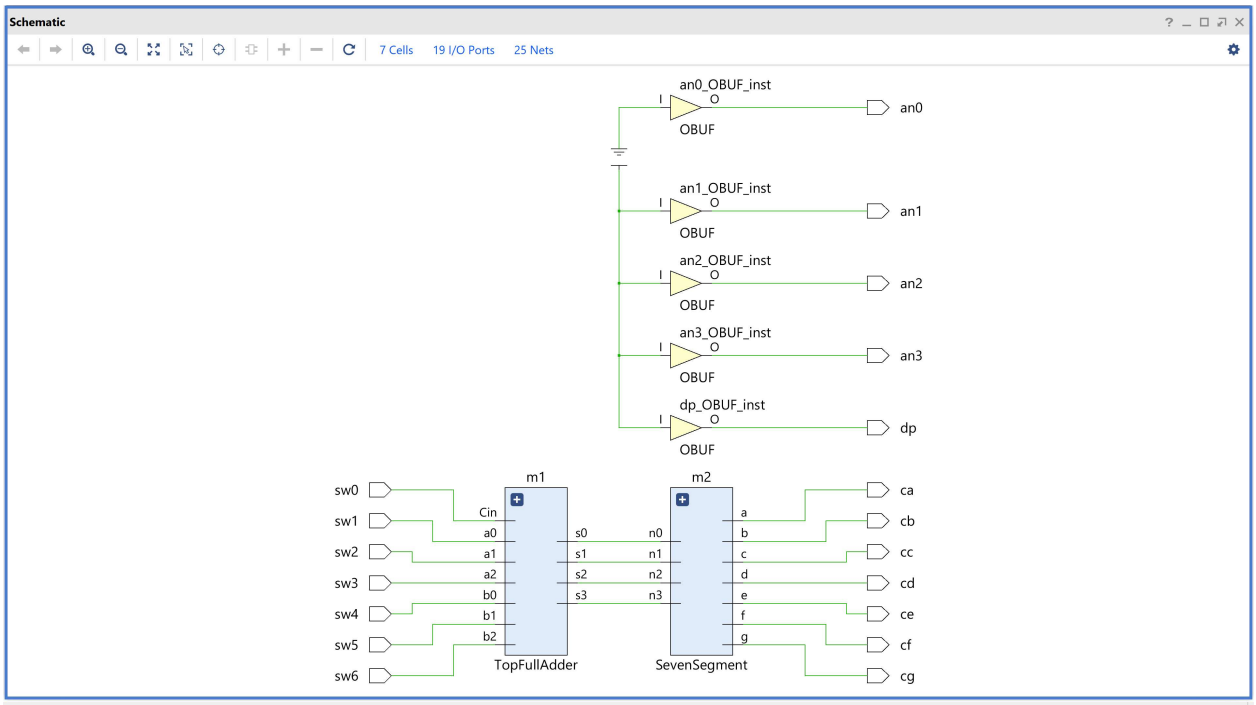
The top-level module with the adders and 7 segment display:

TopAdderSegment.v

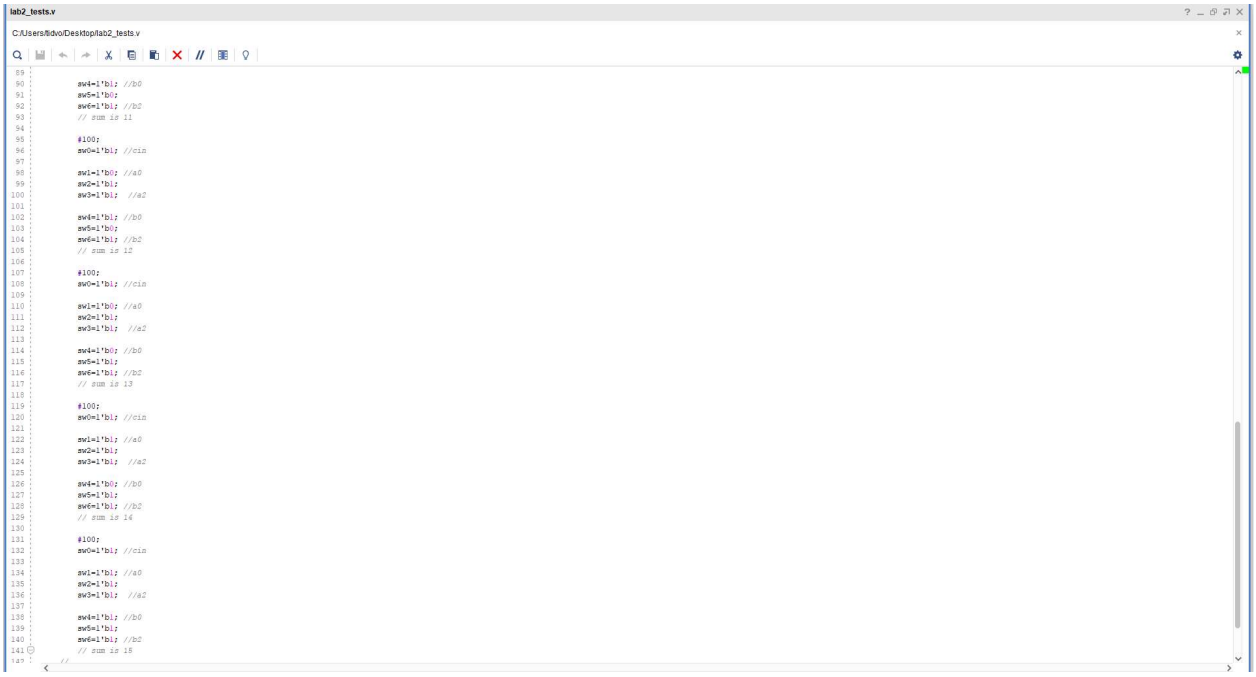
C:/Users/tiffa/Desktop/Lab2/lab2/lab2.srcs/sources_1/new/TopAdderSegment.v

```
//////////////////////////////////////////////////////////////////////////////////


module TopAdderSegment(
    input sw0,
    input sw1,
    input sw2,
    input sw3,
    input sw4,
    input sw5,
    input sw6,
    output ca,
    output cb,
    output cc,
    output cd,
    output ce,
    output cf,
    output cg,
    output dp,
    output an0,
    output an1,
    output an2,
    output an3
    );

    wire three, two, one, zero;

    TopFullAdder m1 (.Cin(sw0), .a0(sw1), .b0(sw4), .a1(sw2), .b1(sw5), .a2(sw3), .b2(sw6), .s3(three), .s2(two), .s1(one), .s0(zero));
    SevenSegment m2 (.n3(three), .n2(two), .n1(one), .n0(zero), .a(ca), .b(cb), .c(cc), .d(cd), .e(ce), .f(cf), .g(cg));

    assign dp = 1;
    assign an0 = 0;
    assign an1 = 1;
    assign an2 = 1;
    assign an3 = 1;
endmodule

```

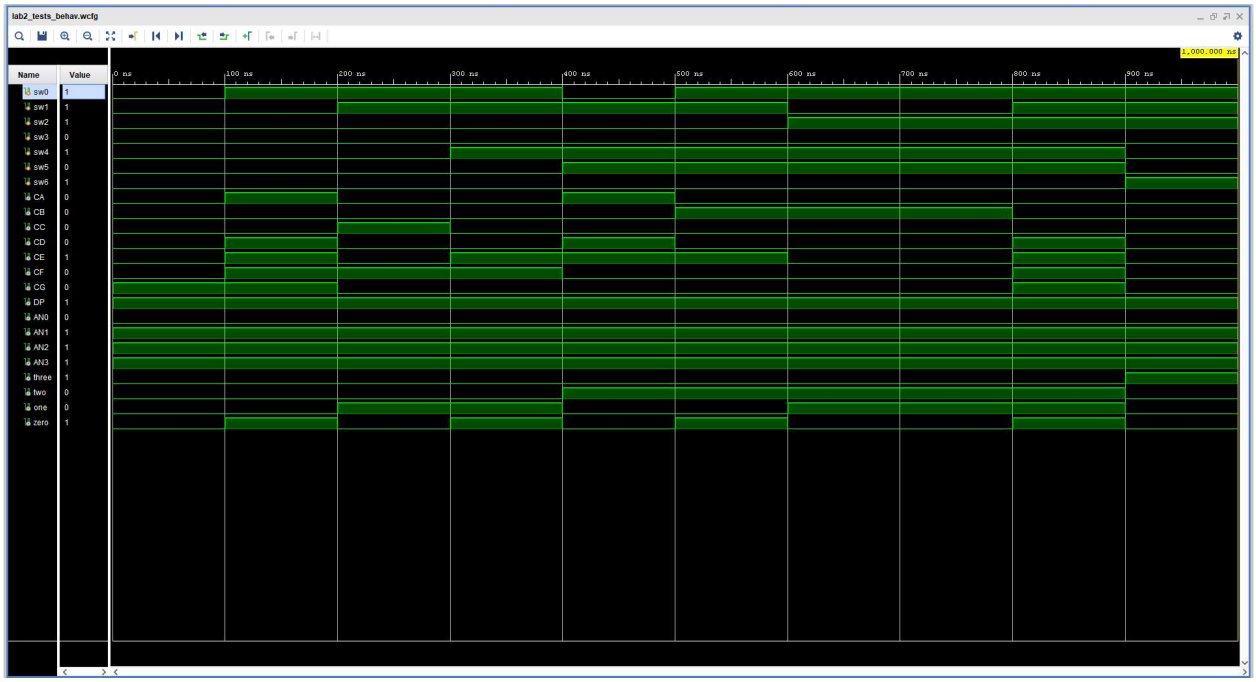
Top-level schematic:



Lab 2 testbench:



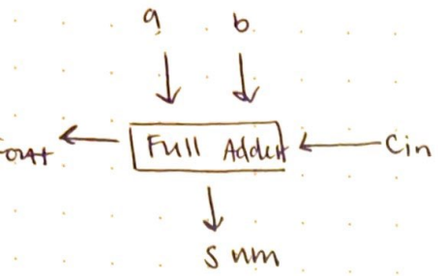
Waveform viewer simulation results:



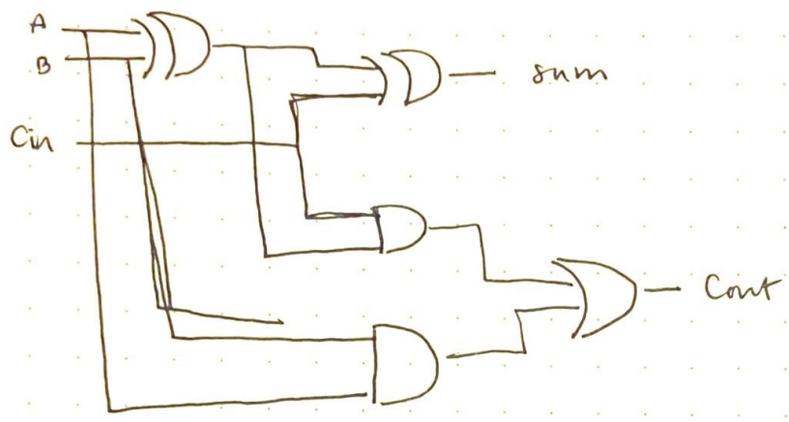
Notebook pages:

LAB 2

January 14, 2020



a	b	cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

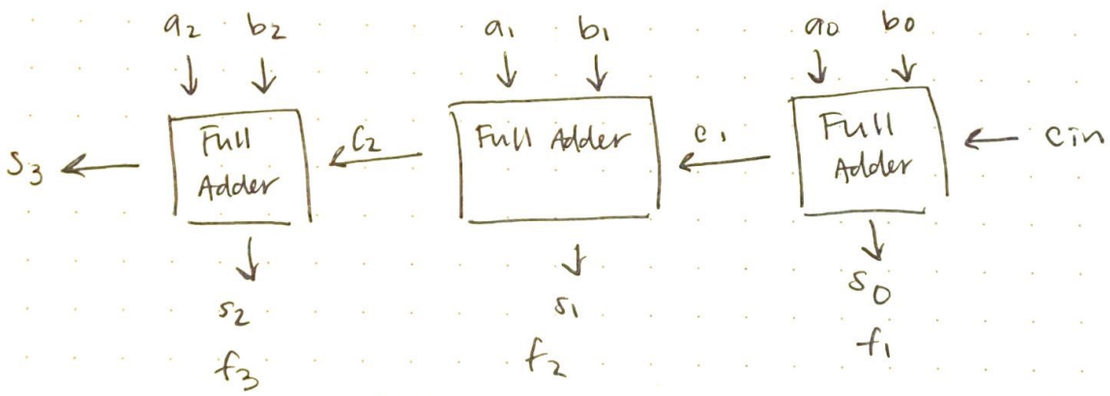


~~sum = (A ⊕ B) ⊕ C~~

~~Cout~~ =

snm = $\bar{A}\,\bar{B}\,Cin + \bar{A}\,B\,\overline{Cin} + A\,\overline{BCin} + ABCin$

Cout = $\bar{A}\,BCin + A\,\bar{B}Cin + ABCin + ABCin$



fi (.a(a0), .b(b0), .cin(Cin), .sum(s0), .cout(c1));

	n_3	n_2	n_1	n_0	A a_{n0}	B a_{n1}	C a_{n2}	D a_{n3}	E	F	G
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	0	1	1	1	1
2	0	0	1	0	0	0	1	0	0	1	0
3	0	0	1	1	0	0	0	0	1	1	0
4	0	1	0	0	1	0	0	1	1	0	0
5	0	1	0	1	0	1	0	0	1	0	0
6	0	1	1	0	0	1	0	0	0	0	0
7	0	1	1	1	0	0	0	1	1	1	1
8	1	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	1	0	0
A	1	0	1	0	0	0	0	1	0	0	0
B	1	0	1	1	1	1	0	0	0	0	0
C	1	1	0	0	0	1	1	0	0	0	1
D	1	1	0	1	1	0	0	0	0	1	0
E	1	1	1	0	0	1	1	0	0	0	0
F	1	1	1	1	0	1	1	1	0	0	0

a_2 not sig

OFF = 1
ON = 0

NOTE $a_n = 0$ for all

~~$a_{n0} = \overline{ABCDEFG} + A\bar{B}CDEFG + \bar{A}\bar{B}CDEFG +$~~

~~$a_{n0} = \bar{A}\bar{B}CDEFG + AB\bar{C}DEFG + \bar{A}\bar{B}\bar{C}DEFG + \bar{A}BC\bar{D}EFG$~~

~~$+ A\bar{B}C\bar{D}E\bar{F}\bar{G} + AB\bar{C}\bar{D}E\bar{F}\bar{G} + A\bar{B}\bar{C}\bar{D}\bar{E}\bar{F}\bar{G} + AB\bar{C}\bar{D}\bar{E}\bar{F}\bar{G}$~~

~~$+ \overline{ABCDEFG} + \bar{A}BC\bar{D}EFG + A\bar{B}CD\bar{E}FG + \bar{A}BCDE\bar{F}G + AB\bar{C}D\bar{E}\bar{F}G$~~

~~$+ \bar{A}B\bar{C}\bar{D}\bar{E}FG + A\bar{B}C\bar{D}\bar{E}FG + \bar{A}\bar{B}CDE\bar{F}G + A\bar{B}\bar{C}DE\bar{F}G + \bar{A}\bar{B}\bar{C}\bar{D}EFG$~~

$A = \bar{n}_3\bar{n}_2\bar{n}_1 n_0 + \bar{n}_3 n_2 \bar{n}_1 \bar{n}_0 + n_3\bar{n}_2 n_1 n_0 + n_3 n_2 \bar{n}_1 n_0$

$B = \bar{n}_3 n_2 \bar{n}_1 n_0 + \bar{n}_3 n_2 n_1 \bar{n}_0 + n_3\bar{n}_2 n_1 n_0 + n_3 n_2 \bar{n}_1 \bar{n}_0 + n_3 n_2 n_1 \bar{n}_0 + n_3 n_2 n_1 n_0$

$C = \bar{n}_3\bar{n}_2 n_1 \bar{n}_0 + n_3 n_2 \bar{n}_1 \bar{n}_0 + n_3 n_2 n_1 \bar{n}_0 + n_3 n_2 n_1 n_0$

$D = \bar{n}_3\bar{n}_2\bar{n}_1 n_0 + \bar{n}_3 n_2 \bar{n}_1 \bar{n}_0 + \bar{n}_3 n_2 n_1 n_0 + n_3\bar{n}_2\bar{n}_1 \bar{n}_0 + n_3 n_2 n_1 n_0$

$E = \bar{n}_3\bar{n}_2\bar{n}_1 n_0 + \bar{n}_3\bar{n}_2 n_1 n_0 + \bar{n}_3 n_2 \bar{n}_1 \bar{n}_0 + \bar{n}_3 n_2 \bar{n}_1 n_0 + \bar{n}_3 n_2 n_1 n_0 + n_3\bar{n}_2\bar{n}_1 n_0$

$F = \bar{n}_3\bar{n}_2\bar{n}_1 n_0 + \bar{n}_3\bar{n}_2 n_1 \bar{n}_0 + \bar{n}_3\bar{n}_2 n_1 n_0 + \bar{n}_3 n_2 n_1 n_0 + n_3 n_2 \bar{n}_0 n_0$

$G = \overline{n_3 n_2 n_1 \bar{n}_0} + \overline{n_3 n_2 \bar{n}_1 \bar{n}_0} + \bar{n}_3 n_2 n_1 n_0 + n_3 n_2 \overline{n_1 n_0}$

Lab 2
1/16 3:07 PM
6758

Cin & (a ^ b) | a & b ;