Exp. 7: Iterative Circuits – Adders and Subtractors

Engineering 357 - Digital Design Lab

Fall 2015

FarnamAdelkhani

Date Experiment Performed: 10-1-2015

Date Report Submitted: 10-2-2015

FarnamAdelkhani
Engineering 357 - Digital Design Lab – 10/1/15
Experiment #7

Exp. 4 Combinational Circuit Design

Abstract and Objective:

This lab is an introduction to adders and subractors in circuit logic design and more specifically how to implement and test them with Xilinx software. The lab also introduces iterative circuits and its related design techniques. Typical word lengths are in the powers of 2 and so one typically needs to design the circuit for one bit and then imply interconnect multiple copies of that same adder circuit to form a complete circuit for the word, this approach is known as iterative design. Of course, making copies of a circuit is easy when CAD software is being utilized, the experimenter will utilize it to chain several full adders together in order to create a ripple-carry adder.

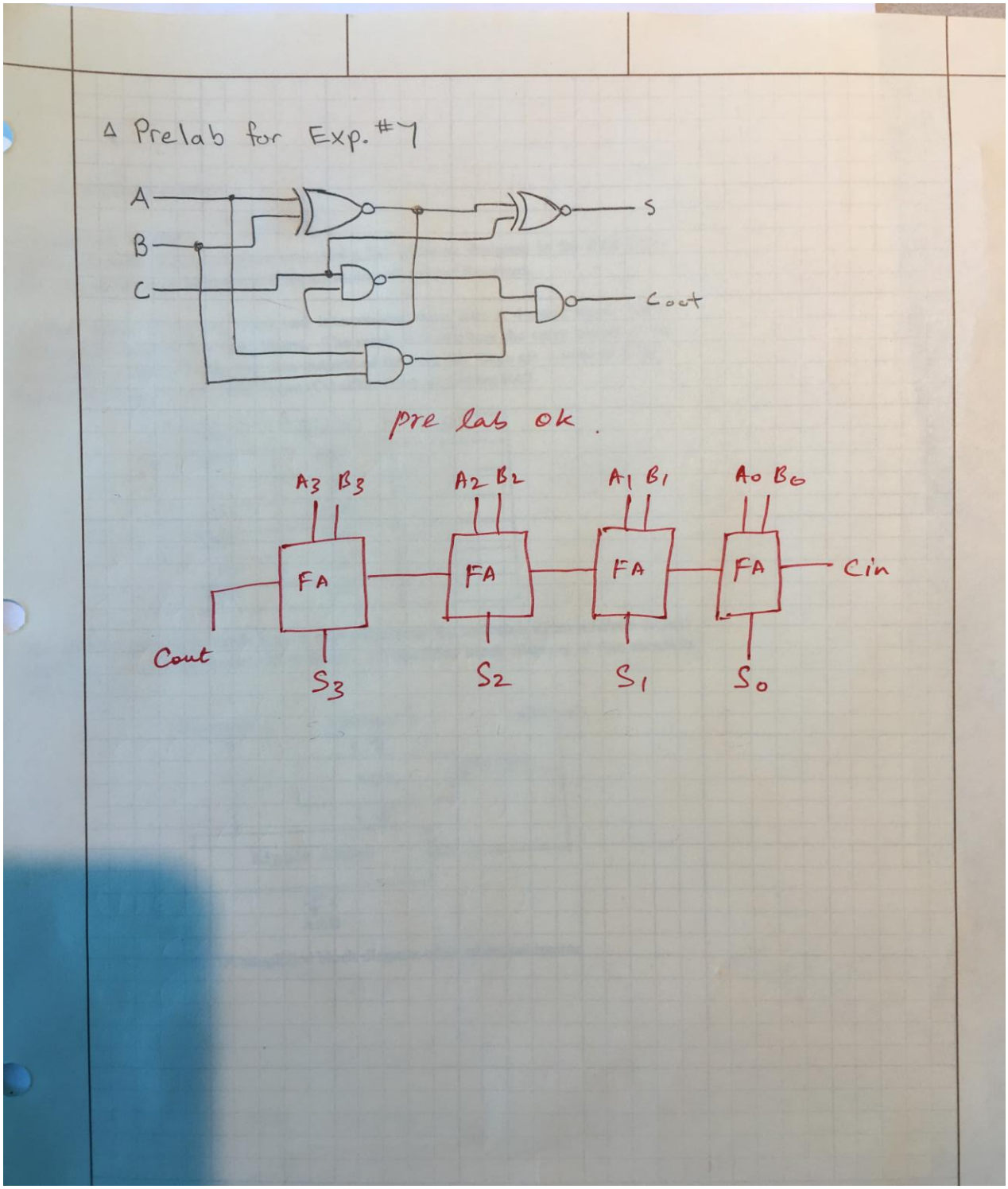
This lab is also about the design of basic binary add and/or subtract circuits. An adder/subtractor circuit can be implemented from an adder circuit by the inclusion of an add/sub control signal and a XOR gate (1s complementer). Inputs will be used that can be easily verified mathematically to confirm that the circuit is functioning as intended.

Components Used:

- Xilinx software on PC
- The following gates/integrated circuits:
  - 3 - nand2 quad 2-input NAND
  - 3 – XOR2 quad XOR
  - Other components as needed

Procedures and result:

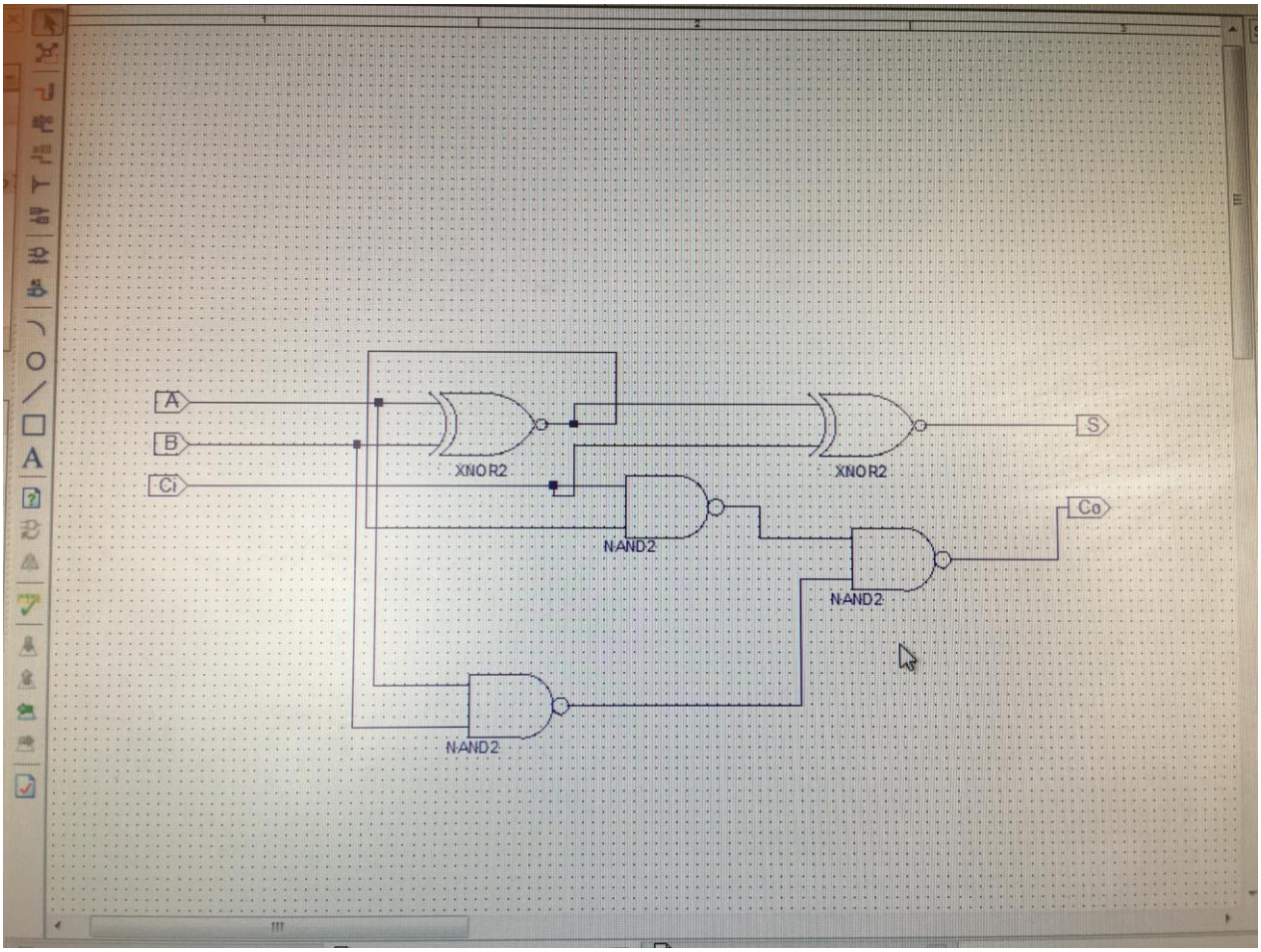
1. The first step is to complete the pre-lab assignment before the lab class. The prelab assignment in this case required the experimenters to design a full adder circuit using as few logic gates as possible. The circuit that was designed is shown on the next page…



Our full adder was approved (see arrow above).

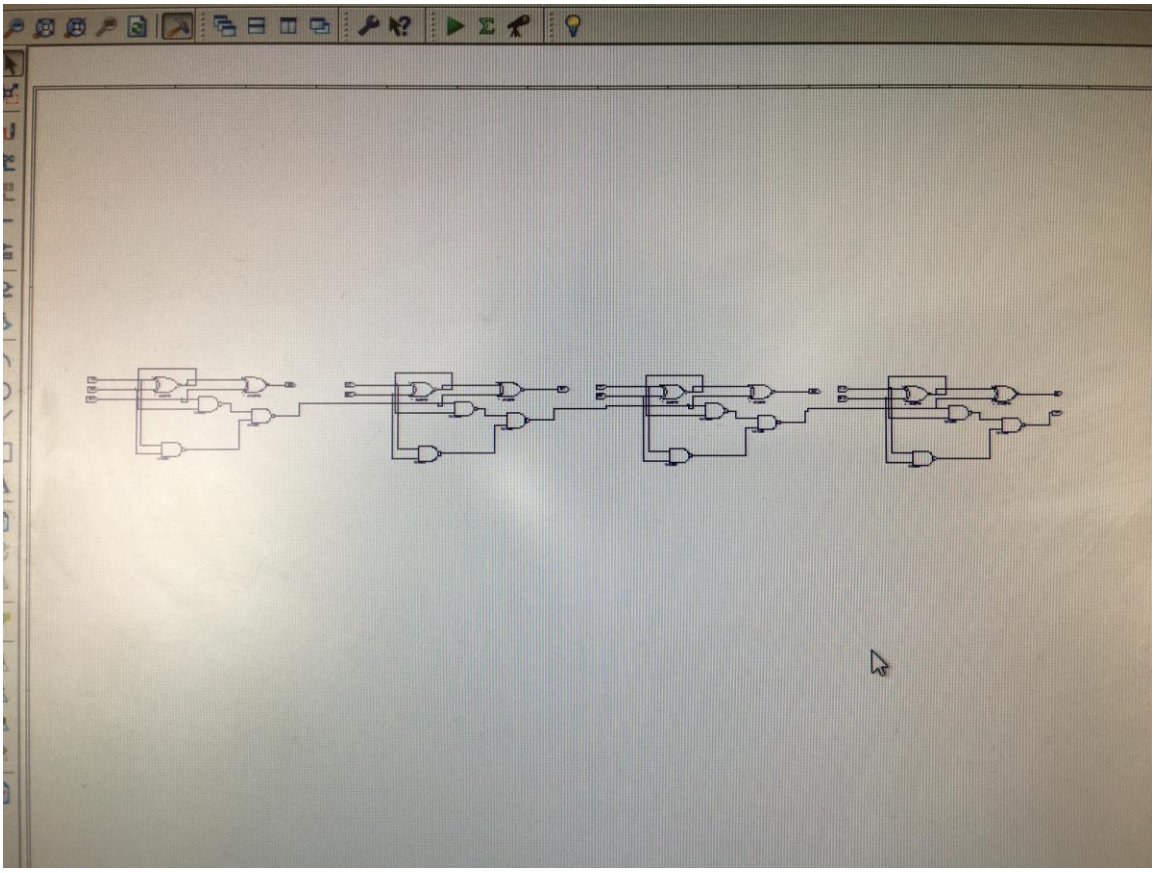
2. Next step is to construct and implement the full adder circuit we designed during the prelab, in Xilinx. Which just requires creating a project and placing

the components and wires on the given area. Our adder circuit in Xilinx is pictured below:

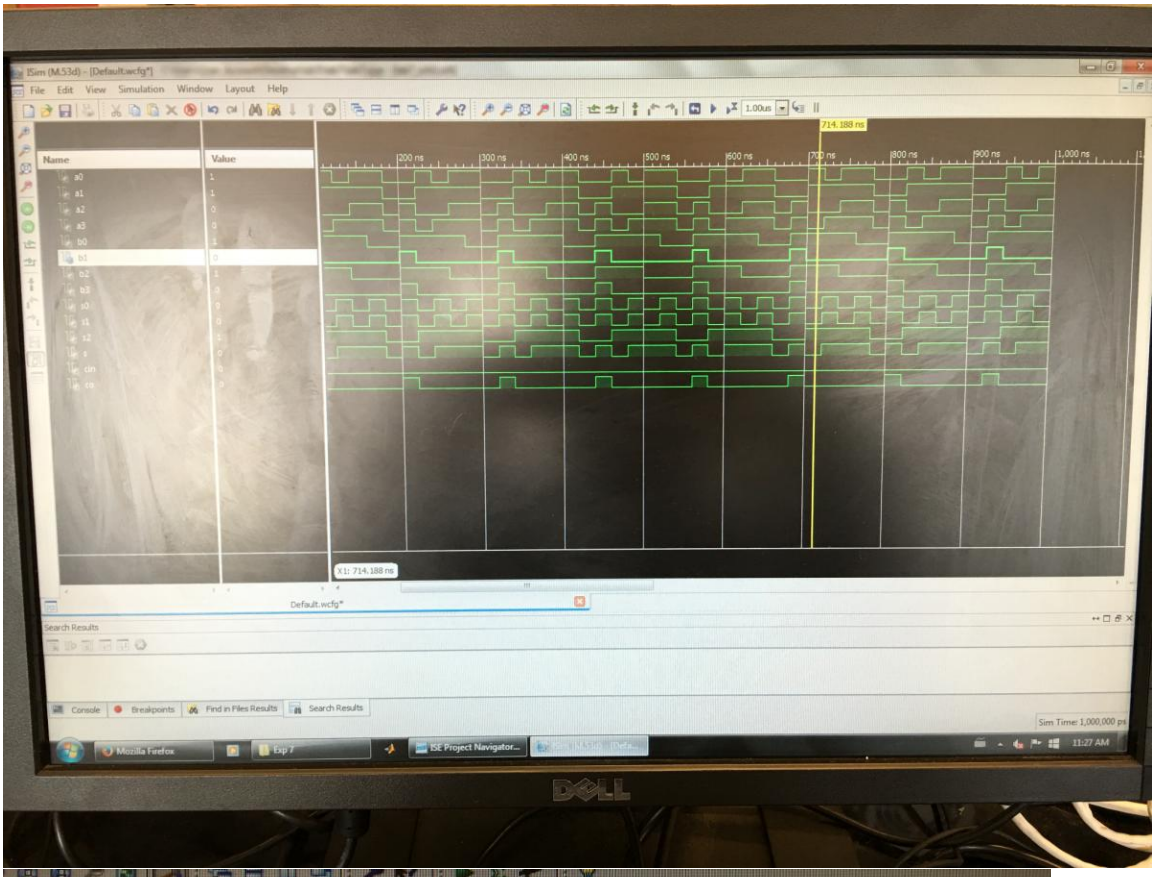


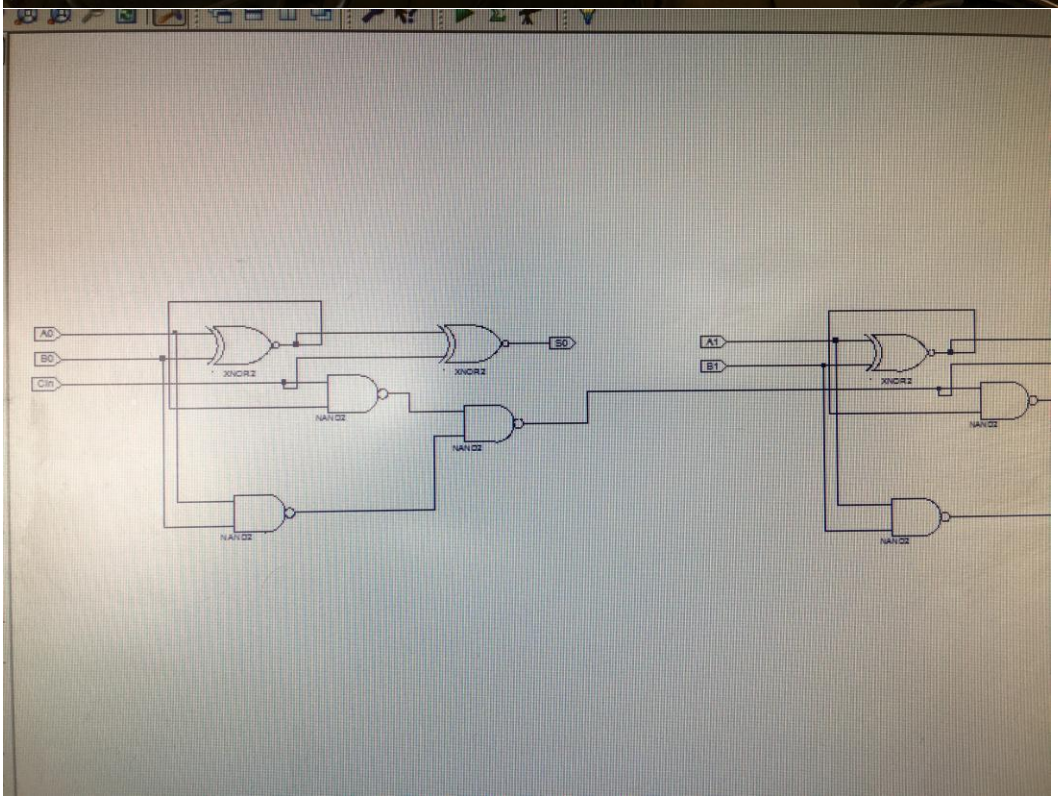
The above circuit was simulated and found to be functional.

3. The next step in this lab is to make four copies of the full adder and interconnect them into a 4-stage ripple adder. We will test the ripple adder by experimentally completing a table where the inputs can easily be used to confirm the given output in Xilinx is correct. The multi stage ripple adder is shown on the next page in Xilinx:



```
    SIGNAL Co   : STD_LOGIC;

BEGIN

    UUT: EXP7_SCH PORT MAP(
        A  -> A,
        B  -> B,
        Ci -> Ci,
        S  -> S,
        Co -> Co
    );

-- *** Test Bench - User Defined Section ***
    tb : PROCESS
    BEGIN
        A  <='0';
        B  <='0';
        Ci <='0';
        wait for 20ns;
        A  <='0';
        B  <='0';
        Ci <='1';
        wait for 20ns;
        A  <='0';
        B  <='1';
        Ci <='0';
        wait for 20ns;
        A  <='0';
        B  <='1';
        Ci <='1';
        wait for 20ns;
        A  <='1';
        B  <='0';
        Ci <='0';
        wait for 20ns;
        A  <='1';
        B  <='0';
        Ci <='1';
        wait for 20ns;
        A  <='1';
        B  <='1';
        Ci <='0';
        wait for 20ns;
        A  <='1';
        B  <='1';
        Ci <='1';
        wait for 20ns;
    END PROCESS;
-- *** End Test Bench - User Defined Section ***

END;
```


ISim (M.53d) - [Default.wcfg*]
File Edit View Simulation Window Layout Help
Name
Value
200 ns
300 ns
400 ns
500 ns
600 ns
700 ns
800 ns
900 ns
1,000 ns
714.188 ns
X1: 714.188 ns
Default.wcfg*
Search Results
Console
Breakpoints
Find in Files Results
Search Results
Sim Time: 1,000,000 ps
Mozilla Firefox
Exp 7
ISE Project Navigator...
11:27 AM
DELL


A0
B0
Cin
XNOR2
XNOR2
S0
NAND2
NAND2
NAND2
A1
B1
XNOR2
NAND2
NAND2

V. LABORATORY WORK

1. Using the schematic capture software construct a full adder as designed in the PRE-LAB. Simulate the circuit to make sure that it is performing all desired functions.

2. Make four copies of the full adder and interconnect them into a 4-stage ripple adder. Experimentally complete the table below. The result is formed by the carry output of the most significant stage and the four sum outputs of the FAs (so there are a total of 5 bits). What should be applied to the carry input (C_0) of the least significant FA?

operand 1	operand 2	result
1010	0101	1111
1111	0001	(1) 0000
1111	0000	1111
0000	0000	0000
1100	0111	(1) 0011
0011	0101	0100

3. The adder circuit can be made into an adder/subtractor by inclusion of an add/sub control signal and a 1's complementer (XOR gates). A simplified block diagram of this circuit is shown below:

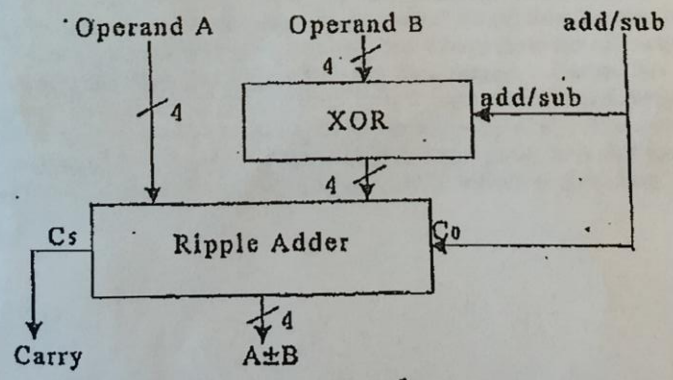
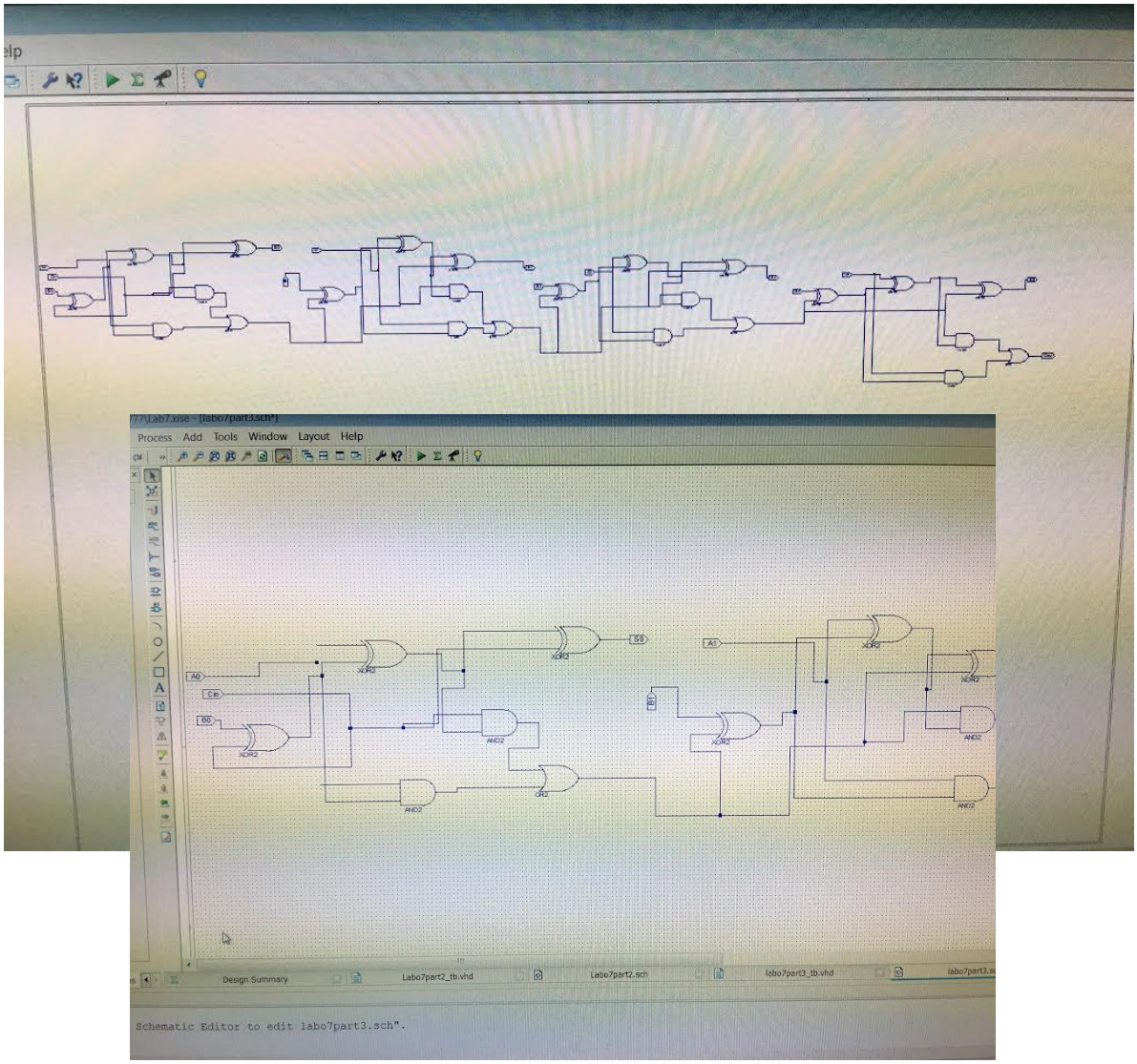
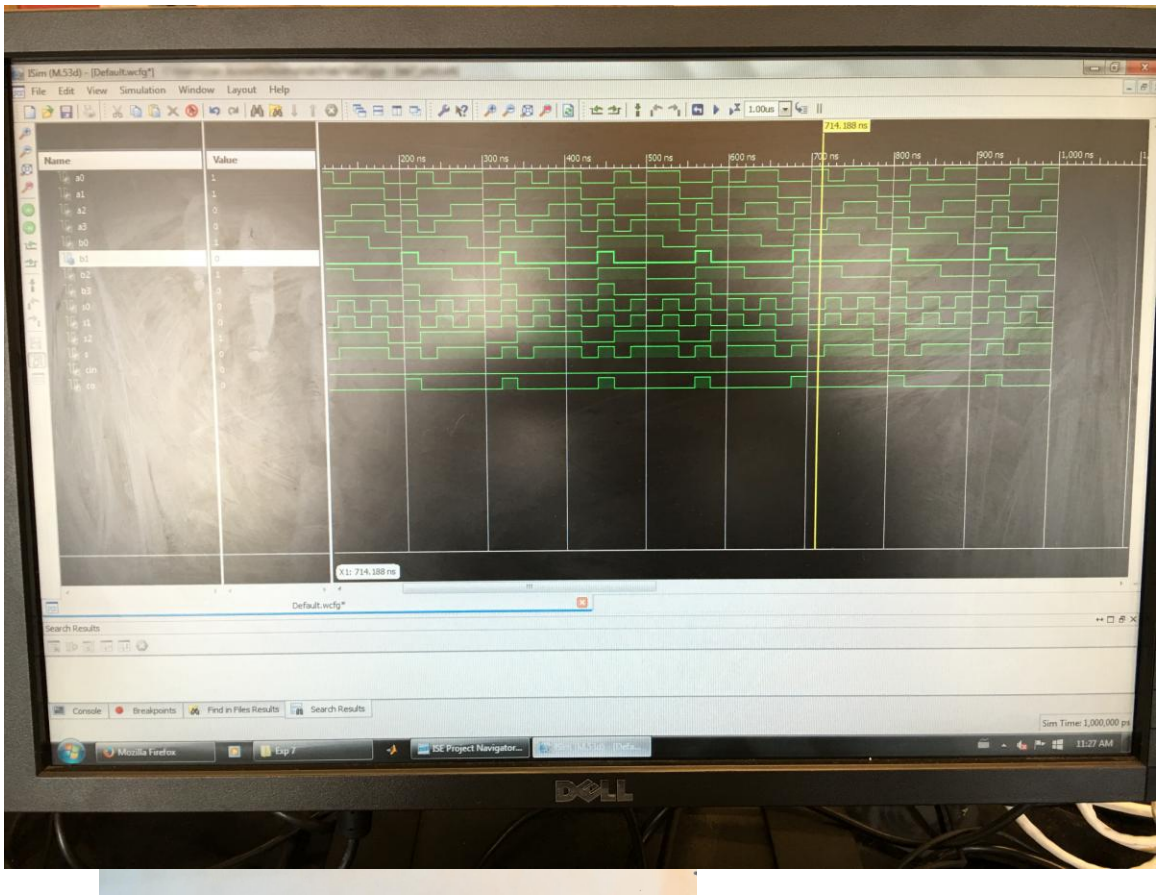


Fig. 7-3. A simplified block diagram of an adder/subtractor

4. The adder circuit can also be made into an adder/subtractor by inclusion of an add/sub control signal and a XOR gate. This circuit was designed and then implementing using the full adder ripple schematics we had already drawn up. We implements an XOR gate at each B input with only the first carry bit also feeding into the XOR gate. Every gate there after will have the same original carry in value while using the next series of B values. The diagram we designed is shown below:





7-4 Experiment 7

4. Experimentally complete the table shown below. Make sure to include the final carry output (C_3) in the result.

operation	operand 1	operand 2	result
sub	1010	0101	0101
sub	0000	0001	(1)1101
sub	0000	1111	(0)0001
sub	1111	0000	(0)1111
add	1101	0111	(1)0100
sub	1011	0111	(0)0100
add	1111	0101	(1)0100

Demonstrate the operations of your adder/subtractor circuit to the lab instructor.

Instructor's signature: _____ Date: 10/01/15

VI. POST-LAB

Write a report summarizing your experience with this experiment, lessons learned, and any other comments you may have. Make sure to include the originals of all schematic and simulation printouts.

58

Conclusion and Summary:

So we have gained familiarity with adder circuit and ripple circuit design while also learning to implement our circuit to perform as an adder/subtractor. It is useful to be given data points that we an confirm by hand in order to confirm our system is working as expected. There are often issues that arise in the Xilinx software itself. You might assume a 10 gb program is free of bugs but not so by any means. Often the best fix is to save the circuit and reload it. I do have some concern that these software issues with Xilinx may become overbearing as a problem once the circuit is much more complex. It is important and vital that we are gaining familiarity with the Xilinx design suite.