Ryerson University

Department of Electrical and Computer Engineering

COE/BME 328 – Digital Systems

Lab 3 - Adder and Subtractor Unit

15 Marks (2 weeks)
Due Date: Week 7

1   Objectives

- To design and build a 4-bit Adder/Subtractor unit (ASU) using an Altera CPLD chip.
- To design the Adder/Subtractor unit that multiplexes add and subtract operations with a common Cin input.
- To implement the 4-bit ASU using VHDL coding.
- To design a Combinational Circuit that takes the decoded output of ASU as input. This Combinational Circuit will have the design logic to output individual digits of the Student Identification number of the student performing laboratory work (use tutorials in course website to set up pin assignments).

2   Pre-Lab Preparation

1. Modify the VHDL code from Figure 5.28 (see course text) to implement the ASU represented in Figure 5.13 (see course textbook). Create a file *ASU.vhd* to accomplish this task.
       Note: change **ieee.std_logic_signed.all** to **ieee.std_logic_unsigned.all**.
2. Copy the code that implements a 4 bit-to-7-segment converter from Figure 6.47 (course textbook) into a file *sseg.vhd*.
3. Modify the code in file *sseg.vhd* to represent the sequence **0-F** on a seven segment display. It must also be able to show the sign (-) in case of negative numbers.
4. Minimize the logic expressions for your customized L3, L2, L1 and L0 -signals. These minimized logic expressions are to be implemented as outputs of the combinational circuit C in Figure 2 (shown below).
5. Create a new VHDL file *C.vhd* to implement the minimized logic expressions for the L3, L2, L1 and L0 signals that correspond to a 4-bit representation of your student identification number. The minimized logic expressions for these signals can be generated using K-Maps and converting each of the resulting expressions to VHDL code.
6. Using the Functional Simulator (waveforms), verify the circuit described by the VHDL file obtained in step 3. By setting values for X3, X2, X1, X0, Y3, Y2, Y1, Y0 and Cin, observe the following signals: S3, S2, S1, S0and Cout. The magnitude (decimal representation) S3, S2, S1, S0 of the sum must be displayed using the Right Seven-Segment display. The sign of the sum must be displayed using the Left Seven-Segment Display (See Figure 1). This ensures that two displays will show the **signed sum**.

3   Laboratory Work

The procedure is divided into two parts:

Part A.

1. This part must be completed during the week 1 of this lab experiment.
2. Compile your modified 4-bit Adder/Subtractor unit (**ASU**) file *ASU.vhd* and create a symbol file *ASU.bsf*
3. Compile your seven segment code *sseg.vhd* as a separate project and create a symbol file *sseg.bsf*
4. Start a new Project **CombinedASU1** and create a block schematic file *CombineASU1.bdf*.
5. Import the symbols **ASU** and **sseg** into the block schematic file *CombinedASU1.bdf*.
6. Complete the wiring and pin assignments by following the connections described in the Figure 1 (Typical Schematic generated using Quartus II depicted in).

S (ASU Output)				STUDENT_ID (C Output)				
S_3	S_2	S_1	S_0	L_3	L_2	L_1	L_0	
0	0	0	0	0	0	0	1	
0	0	0	1	0	0	0	1	
0	0	1	0					
0	0	1	1					
0	1	0	0					Fill your Student ID in Binary form
0	1	0	1					
0	1	1	0					
0	1	1	1					
1	0	0	0					
1	0	0	1					
X	X	X	X	1	1	1	0	
X	X	X	X	1	1	1	0	Don't Cares
X	X	X	X	1	1	1	0	
X	X	X	X	1	1	1	0	

Table 1. .   4-bit Student ID generated by combinatorial logic

7. Assign the following signals to the dedicated I/O-pins of the Altera Cyclone-II EP2C35F672C6 FPGA on the prototype board (see Table 1 and Fig.1), using either Pin Assignment menu (Use Pin Assignment Manual on the course website)
    a. X3, X2, X1, X0, Y3, Y2, Y1, Y0 to SW7, SW6, … , SW1, SW0
    b. Cin to pushbutton [0]
    c. S3, S2, S1, S0, Cout to the Right/Left Seven-Segment Displays, LEDR0 respectively.
    Note: 1. All the LEDs are of common anode type (active LOW)
          2. All the 7-segment displays are of common cathode type (active HIGH)
8. Recompile the design and implement/program it into the Cyclone- II EP2C35F672C6 FPGA. Note: before programming double-check (a) and (b). Incorrect pin assignment can result in failure of the Cyclone-II EP2C35F672C6 FPGA.
9. Install jumpers to the prototype board in order to connect I/O pins of the Cyclone- II EP2C35F672C6 FPGA to switches and LEDs.

10. Test your design and demonstrate the results on the FPGA board to your lab instructor. Note ; the maximum number of ASU results should be 9.
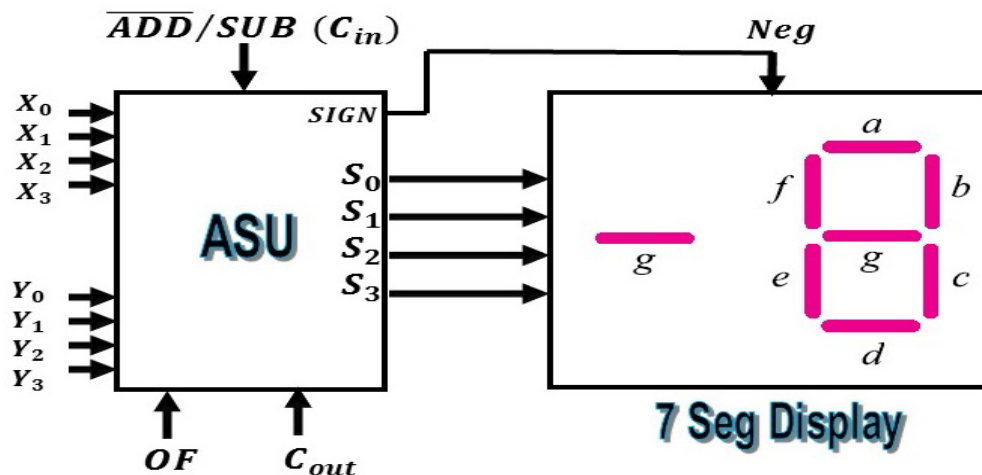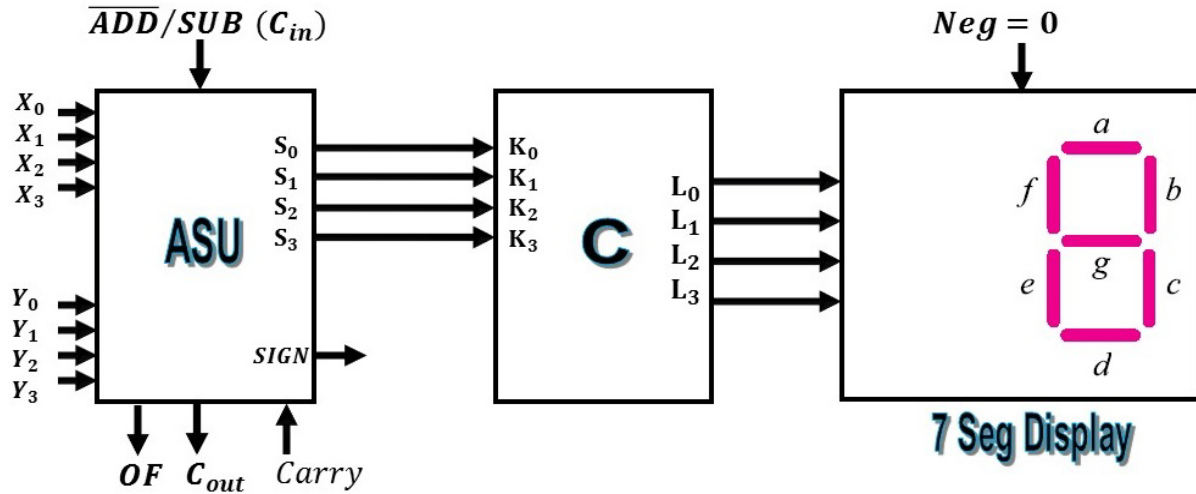

**Figure 1. Block Diagram for PART A implementation**

Part B.

1. This part must be completed during the week 2 of this lab experiment.
2. Consider the 9 digits of the student identifier D = {d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9} in its general representation. Write a table that represents each digit of your student ID in four bits. For example, if a student's ID is: **111726015**, it will likely follow a 4-bit sequence {**0001, 0001, 0001, 0111,……, 0101**} as depicted in Table 1.
3. Compile your customized Combinatorial Unit *C.vhd* and create a symbol file *c.bsf*
4. Start a new Project **CombinedASU2** and create a block schematic file *CombineASU2.bdf*.

Figure 2. Block Diagram for PART B implementation

5. Import the symbols **ASU, combinatorial** and **sseg** (refer Part A) into the block schematic file *CombinedASU2.bdf*.
6. Complete the wiring and pin assignments by following the connections described in the Figure 2 (Typical Schematic generated using Quartus II depicted in).
7. Repeat the steps **7-10** from **Part A**.

- ### **A**ppendix A
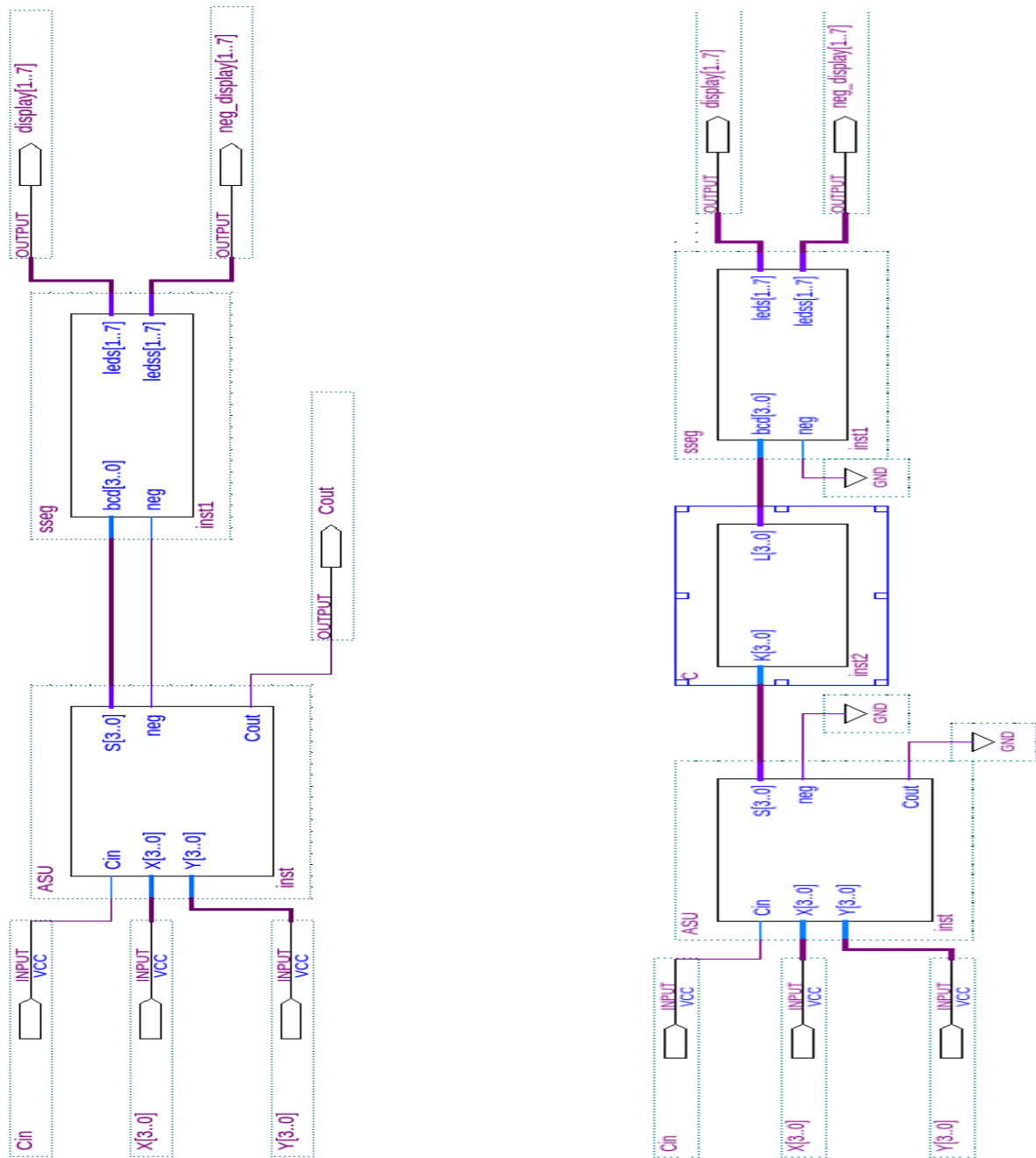
Figure 3. Typical Block schematic for PART A & B implementation using Quartus II