BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSE306 (Computer Architecture Sessional), January 2022 Term
All Lab Sections, May 23, 2022

1 Introduction

As part of this assignment, you have to submit both software simulation and hardware implementation of a simplified Arithmetic Logic Unit (ALU). The functional design specification for each group of each section can be found in Appendix A. This is a group assignment, and all group members have to participate equally. The software simulation submission deadline for all sections is the same, and it is June 8, 2022 (Wednesday) at 11:59 PM. On the other hand, the hardware implementation deadline is the next sessional day and is different for each section (see the Deadline section for details). You also have to write a group report and submit it during the evaluation day.

2 Specification for 4-bit ALU Simulation

- The functional design specification for each group of each section can be found in Appendix A. First, read the specification of your group carefully. Then, go through the following requirements/instructions of this section.
- Efficiently design (with minimum possible ICs) the ALU according to the specification.
- In addition, you need to implement the following flags.
  - Carry (C)
  - Sign (S)
  - Overflow (V)
  - Zero (Z)
- Flags will be affected as per the rules of Assembly Language. However, we have added some exceptions (only for this assignment) to incorporate flexibility to flag status bits after logical operations. Remember that this flexibility is to make your assignment easier, although it breaks some of the rules of the assembly programming language.
  1. **For NOT Operation:**
     - After the NOT operation, which makes the result to 0000, if Z becomes 0 from 1, it will not be accepted. However, if Z becomes 1 or if Z value remains unchanged, both will be accepted.
     - After the NOT operation, if S remains unchanged or it reflects the highest order bit of the result, both will be accepted. But if S bit changed and it changed to a wrong value, it will not be accepted.

- To make your life easier, we shall not check the C and V bits after NOT operation, i.e., you can consider these as Don't care.

2. **For AND/OR/XOR Operation:**
   - C and V should be cleared (0) after the operation.
   - S and Z should be changed according to the output.

- Any **2-input** SSI (AND, OR, NOT, XOR etc.) and MSI (MUX, Decoder, Adder etc.) chip can be used.
- Emphasis should be given on efficiency of design and minimization of ICs used.
- For simulation, you can use any simulation software.
- Your software design must be in IC-level.
- **Software Simulation Submission:** A submission link will be opened on Moodle for submitting your ALU simulation. Make a folder containing all your simulation project files, zip it and submit it following the naming format. If you submit only a single project file, then just submit the file following the naming format. The naming format should be your section name followed by your group id (e.g., B1_Group7). Please ensure a single submission from each group (only a single member of the group should submit).
- **Report Preparation Guideline:** Your have to write a report containing the followings:
  - Introduction
  - Problem Specification with assigned instructions
  - Detailed design steps with k-maps (if applicable)
  - Truth Table
  - Block Diagram
  - Complete Circuit diagram
  - ICs used with count as a chart
  - Simulator used along with the version number
  - Discussion

  The report must be handwritten.
- **Software Simulation, Hardware Implementation Evaluation and Report Submission:** Both software simulation and hardware implementation will be evaluated in the regular laboratory for respective sections (See the Deadline section for the specific time and date). Each group has to bring at least one laptop to show the software simulation for the group. As an alternative, a group can install the required simulator on a laboratory PC before their evaluation day. You have to show the full working hardware implementation during the sessional time. Also, you need to submit your report at the beginning of the sessional.
- For hardware implementation, you can lend the required instruments from the laboratory from now on (on the condition of returning these immediately after the evaluations and without any alternation), or you can use your own instruments.
- Any type of plagiarism will be punished.
- All the specified date and time is according to the Bangladesh Time. Late submission is not allowed.

3 Deadline

Software Simulation Submission Deadline: For all sections: June 8, 2022 (Wednesday) Bangladesh Time.

Software Simulation and Hardware Evaluation Timeline:

- A1: June 14, 2022 (11 AM)
- A2: June 21, 2022 (11 AM)
- B1: June 12, 2022 (2:30 PM)
- B2: June 19, 2022 (2:30 PM)

4 Where to Ask?

For any query, you can ask your instructor during the theory or sessional class. You can also email at "madhusudan.buet@gmail.com". Please allow at least two days to respond to your answer (In most cases, your question will be answered in the shortest possible time). Hope you will enjoy the assignment. Best wishes to you!

5 Version

This section contains the version of the assignment. It starts with Version 0. If we find some major problems in this assignment description file, then we shall change this pdf. If that case, we shall increase the version number and list the changes in this section. So, keep an eye on this version number of the pdf in the moodle to see whether the version has been changed or not. If it is changed, first read this section to see where the changes have been made and whether it is applicable to your group. On the other hand, if the changes are minor (for example, correcting the grammatical mistakes), then the version number will not be changed.

5.1 Version 0

This is the initial version of the problem description pdf.

5.2 Version 1

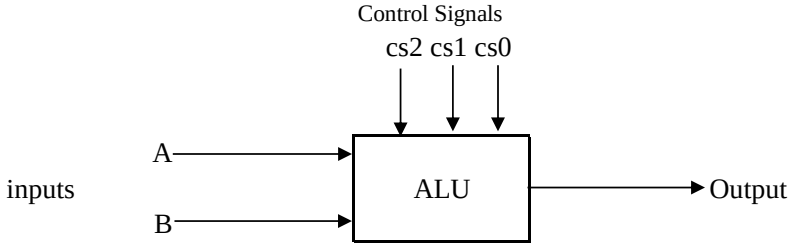
- In the "Specification for 4-bit ALU Simulation" sections following updates have been made.
  - SSI chip has been restricted to 2-input only.
  - Your software design must be in IC-level.
  - A step in report writing, "Detailed design steps with k-maps (if applicable)", has been added.
  - It has been specified that report writing must be handwritten.

Appendix A

Functional Design Specifications for All Groups

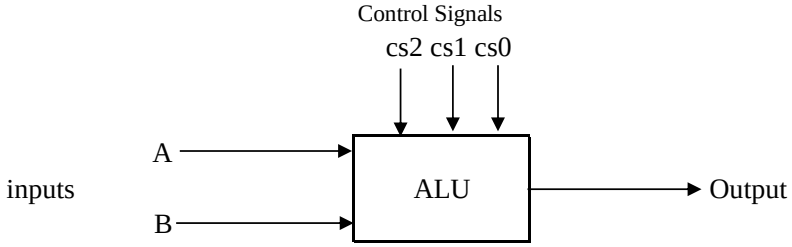
Section A1

	cin		Functions for					
cs2	cs1	cs0	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
0	0	0	Transfer A	Add	Subtract with borrow	Add	Subtract with borrow	Decrement A
0	0	1	Subtract with borrow	Decrement A	Transfer A	Transfer A	Decrement A	Add
0	1	0	Incerement A	Add with carry	Subtract	Add with carry	Subtract	Transfer A
0	1	1	Subtract	Transfer A	Increment A	Increment A	Transfer A	Add with carry
1	x	0	AND	AND	AND	Complement A	XOR	OR
1	x	1	XOR	Complement A	OR	AND	AND	AND



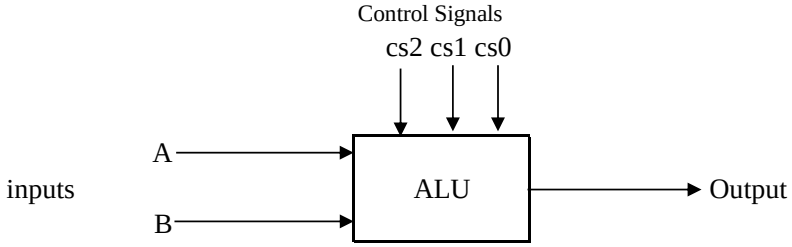
Section A2

cin			Functions for					
cs2	cs1	cs0	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
0	0	0	Decrement A	Transfer A	Add	Decrement A	Add	Subtract with borrow
0	0	1	Add	Subtract with borrow	Decrement A	Subtract with borrow	Transfer A	Transfer A
x	1	0	AND	AND	AND	OR	XOR	Complement A
1	0	0	Transfer A	Increment A	Add with carry	Transfer A	Add with carry	Subtract
1	0	1	Add with carry	Subtract	Transfer A	Subtract	Increment A	Increment A
x	1	1	XOR	Complement A	OR	AND	AND	AND



Section B1

		cin	Functions for					
cs2	cs1	cs0	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
0	0	0	Add	Subtract with borrow	Transfer A	Transfer A	Decrement A	Decrement A
0	0	1	Add with carry	Subtract	Increment A	Increment A	Transfer A	Transfer A
0	1	x	AND	AND	AND	OR	XOR	Complement A
1	0	0	Transter A	Decrement A	Add	Subtract with borrow	Add	Subtract with borrow
1	0	1	Increment A	Transfer A	Add with carry	Subtract	Add with carry	Subtract
1	1	x	XOR	Complement A	OR	AND	AND	AND



Section B2

cin			Functions for					
cs2	cs1	cs0	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
0	0	0	Subtract with borrow	Add	Decrement A	Decrement A	Subtract with borrow	Transfer A
0	0	1	Decrement A	Transfer A	Subtract with borrow	Add	Transfer A	Add
1	0	0	Subtract	Add with carry	Transfer A	Transfer A	Subtract	Increment A
1	0	1	Transfer A	Increment A	Subtract	Add with carry	Increment A	Add with carry
x	1	0	AND	AND	AND	Complement A	XOR	OR
x	1	1	XOR	Complement A	OR	AND	AND	AND