Lab 1: Machine Code Programming

Assigned: Mar. 15, 2024
Due: to be announced

Instructor: Jingwen Leng
Teaching Assistant: Hu Weiming, Huang Ziyu

1 Introduction

There will be 3 labs in this semester to let you play around the LC-3 Simulator (`https://highered.mheducation.com/sites/0072467509/student_view0/lc-3_simulator_lab_manual.html`). In this very first lab, we start with two problems including debugging and developing programs in LC-3 machine code. You are supposed to submit files according to the instructions of the two following problems.

2 Run A Program

As discussed in Chap.4 in the lecture, LC-3 Simulator is build with the LC-3 instruction set architecture (ISA). The following machine code is a simple program taking one input at memory position x3014 and implement some function. The output of the program is stored at memory position `x3015`. Please load the following program into the simulator and try the simulator with it. You can change the input by modifying the last second line of the code or replacing the memory contents of `x3014`. For example, the current program is using $5_{(10)}$ as input. And then answer the following questions.

Listing 1: Machine code for the buggy program.

```
3000
E213
6240
14A0
16E1
1920
1B60
127E
0C09
9C7F
1DA1
1D85
0405
1883
1B61
14E0
1720
0FF7
B801
F025
3015
0005
0000
```

Please **1) run the program** with the last digit of your student ID as input and report the data at memory `x3015`, **2) explain the function** of the code, **3) locate the bug** in the code and **provide a possible solution**. You are asked to summarize the answers of the above three questions into a brief report named **q1.doc/pdf/md** for submission.

Hints: The program may not work properly for all inputs.

3 Write A Program

In this part, you are supposed to write a machine code program that shifts a bit pattern to the right by a certain amount (a number between 0 and 16, not including 0 and 16).

Problem: In specific, you are asked to write a program in LC-3 machine language to shift a bit pattern some number of bits to the right and store the result in memory. The number of bits the bit pattern should be shifted is called *shift amount*. The shift amount is a positive number between 0 and 16, exclusively (that is 0 and 16 are invalid shift amounts). Your program should assume that the initial bit pattern to be shifted is in memory location `x3100` and the shift amount is stored in memory location `x3101`. Using those values, your program should perform the right shift and store the result in another memory location `x3102`. Your program should start at memory location `x3000`.

Example: If the memory location `x3100` contains the bit pattern `1001011001011101` and memory location `x3101` contains the value `0000000000000101` (decimal 5), then your program needs to shift `1001011001011101` by 5 bits to the left and store the bit pattern `0000010010110010` in memory location `x3102`. Note that when you shift a bit pattern *n* bits to the right, you fill the highest *n* bits of the bit pattern with `0`s.

Hint: You may shift the pattern right by shifting something left.

Please write **1) a machine code program** in a file named **q2.bin** *(or you can also write in assembly language but submit the binary program in the end)*, and **2) a brief explanation** of your code within 200 words named **q2.doc/pdf/md** (text exceeding the requirement may not be considered).
We will check the correctness of your program with a set of test cases. Besides, we also check for plagiarism, so do the programming BY YOURSELF.

The way to place the array to the main memory can be found in “Loading the data (ten numbers) into memory” of the Chap.3 in the LC-3 user guide.