CS 2110 LC-3 Assembler Example

Pulkit Gupta

October 16, 2019

Contents

1 What is this assembler

I recommend reading the lab guide for lab 14 to familiarize yourself with how the LC-3 assembler works.

This assembler (LC3asm.java) is a simple 2 pass assembler that performs most of the features of the actual LC-3 assembler as described in chapter 7 of the textbook. The whole purpose of releasing this assembler to students is to give you the ability to read the source of a 2 pass assembler and see how it would work.

Key features that are missing/changed include:

- aliases for traps are not implemented (IN, OUT, PUTS, etc.)

- immediate values and PC relative offsets are not checked for validity as to make the code a little easier to read

- the obj file output does not list the number of instructions at every .ORIG statement, instead it says "ORIG: x####" to indicate where a block of memory goes.

1.1 What it does

Given a valid input file, this assembler will generate 3 files.

- .debug - holds debugging and assembly process information (similar to .lst)

- .sym - the symbol table generated by the first pass

- .obj - stores the .orig addresses and hex of each instruction/memory address relevant to the program, intended for you to be able to use in a simulator

- .dat - stores the program as a format ready for use in the datapath assignment in case you want to write your own test cases

1.2 Using this Assembler

You can compile the assembler with: "javac LC3asm.java"

You can run it on a file with: "java LC3asm <my-file.asm>"

Once it finishes, you should be able to check out the files it generated, my-file.debug, my-file.sym, my-file.obj, my-file.dat