

CS 252 Spring 2008

LC3 Demo

Spyros Blanas, Priyananda
Shenoy, Shengnan Wang

LC3 Tools

Now that you know how the LC3 works, you would want to write programs for it.

Problem!

The LC3 doesn't exist!

Solution

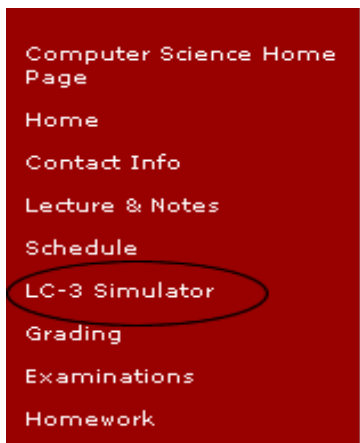
“simulate” it on a real computer

LC3 Tools

- The LC3 Editor
 - Lets you write programs
 - Converts the program into something which the Simulator can understand
- The LC3 Simulator
 - You can load the program and execute it
 - You can see what effect the program has
 - You can debug the program

How do you get it?

- Already installed on Windows machines in CS1368.
- Windows only ☹️ *
- Link to the installation is present on the class webpage.



TAs Spyridon Blanas, Priyananda Shenoy & Shengnan Wang
URL: <http://www.cs.wisc.edu/~david/courses/cs252/Spring2008/>

LC-3 Simulator

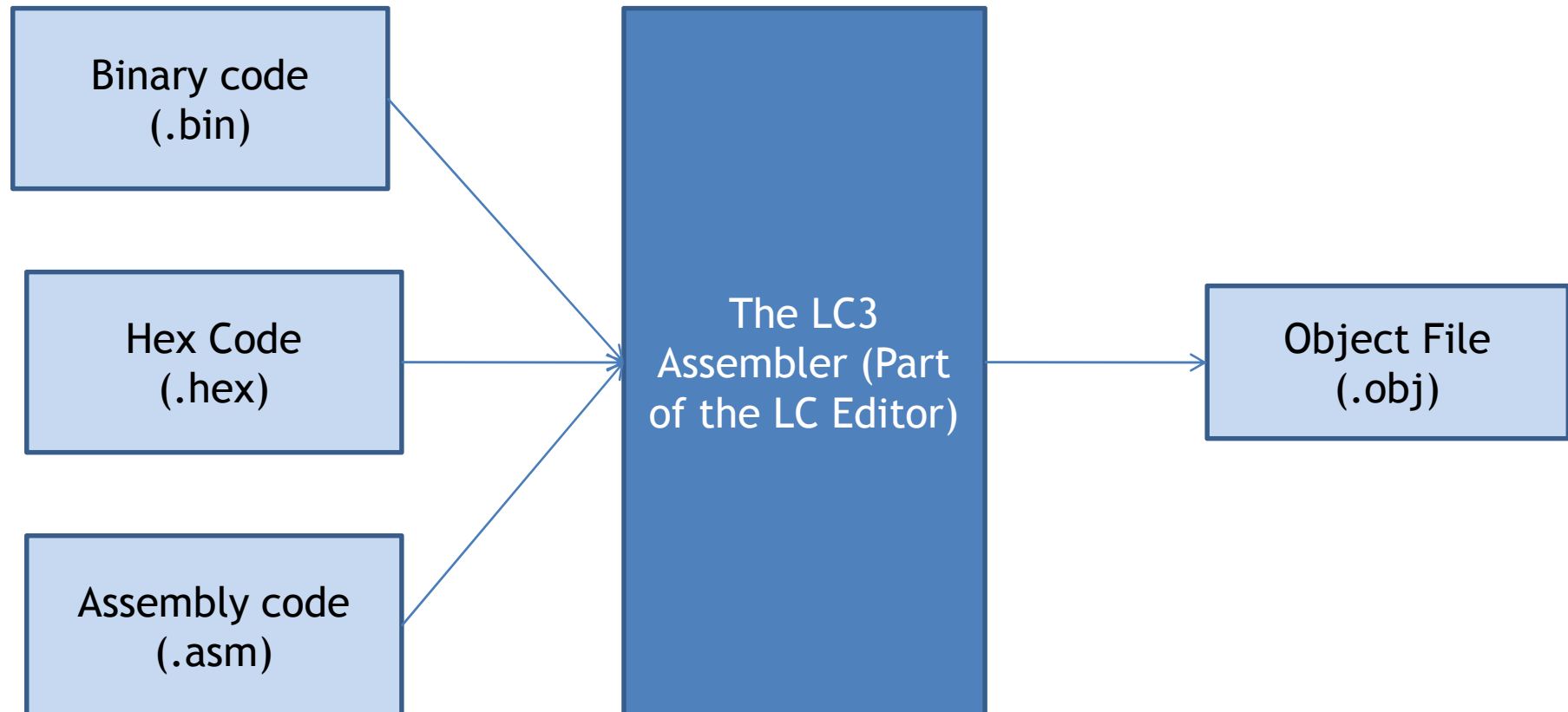
You may either use your own Windows computer to run the LC-3 pub-xp workstations in rooms 1351 and 1368, which have the LC

- [LC-3 Simulator Windows Executable](#)
- [LC-3 Simulator Guide \(PDF\)](#)
- LC-3 Instruction Set on One Page (with ASCII character tabl
- From LC-3 in-class tutorial:


* If you're adventurous there is a Unix version which you can compile.

LC3 Editor


- Lets you type programs in three ways




Writing programs in Binary

- Each line has 16 characters (spaces are ok).
- First line of the program must be the starting address of the program (Usually $x3000 = 0011000000000000$).
- Each line is one instruction.
- Press the  button to compile it into object file.

Writing programs in hex

- Each line has 4 hex characters.
- Don't need to put 'x' in front of the numbers.
- First line is starting address of program. (Usually 3000)
- Each line is one instruction
- Press the  button to convert it into object code.

Writing programs in assembly

- The first line should be a “.ORIG” pseudo-opcode. This specifies the starting address of the program.
- Each line is one instruction
- Instead of coding the instruction bits by ourselves, we write it in a “high level symbolic” manner.
- More on this in future lectures.
- Press the  button to convert this into object file.

Writing programs

- For most assignments in this class, the starting address of your program will be x3000.
- The last instruction of your program **must** be HALT. The instruction is:
 - In Binary: **1111000000100101**
 - In Hex: **F025**
 - In ASM: **HALT**

LC3 Editor Demo

The LC3 Simulator

- Creates a “virtual machine” for programs to run.
- You can load your program’s object code(.obj) and data using ‘File’-> ‘Load Program’.
- Shows the contents of:
 - All the registers
 - Memory locations

Executing Programs

- Steps
 - Load the program
 - Make the PC point to the first instruction
 - Execute the program
 - Run the whole program. Stops when it reached the Halt Instruction.
 - Step over the program one instruction at a time.

Debugging

- The top of the screen shows contents of all registers
- Breakpoints - pauses execution when a breakpoint is hit, until the user resumes the execution.
- Editing memory locations - you can manually change the contents of any memory location.

LC3 Simulator Demo

Resources

- There is a LC3 simulator guide available (Link on class web page).