

The ARC Simulator, Version 1.x
vph 1/2/01

Please send bug reports to pocabugs@cs.rutgers.edu. Place "ARC Simulator Bug" in the message header.

I. What Is The ARC Simulator?

The ARC Simulator is a Java 1.1 Application that simulates the execution of an ARC machine language program. It is a "functional" simulator; it simulates the overall execution of an ARC program, but not the internal operation of any particular ARC architecture. It should be useful in demonstrating the execution behavior of ARC programs, but has no information related to timing. It executes each ARC instruction completely before beginning the execution of the next instruction.

II. What kind of computer and operating system will run this program?

This simulator, and all the tools in this package, will run on any computer that supports Java version 1.1 or higher. The package has been tested on the Macintosh, Windows NT, and Solaris platforms.

III. How do I start the simulator running?

First, you must have the Java 1.1 or higher runtime environment installed on your computer. Since this program is a Java Application and not an Applet, it cannot be run from within a Web Browser. How the simulator is launched depends upon which platform you are using. For the Mac, simply drag the ARCToolsv1.2.5.jar file that is in this directory to the JBindery icon, and click "RUN".

For Unix machines, and for PC's from within a DOS window, cd to the directory containing the .jar file and type the command:

```
java -classpath ARCToolsv1.2.5.jar ARCTools.ARCSim  
or  
jre -classpath ARCToolsv1.2.5.jar ARCTools.ARCSim
```

Note: in both cases, you should have the PATH and CLASSPATH environment variables set so that they include *this* directory, so the Java Runtime Environment knows where ARCToolsv1.2.5.jar file is located. Consult the Java documentation for details.

After a short delay you should be presented with a window titled, "ARC Simulator." In the window should appear, from top to bottom:

- Editable text fields displaying PC, program status register(which is not editable), and 32 registers numbered from r0 to r31.
- A row of control buttons.
- A display of 8 machine words with their addresses, complete with checkboxes for setting breakpoints, disassembled source code, and navigation buttons for displaying other memory locations. These fields are not editable.
- At the bottom of the window, an editable display of memory addresses and contents.

IV. What kind of input files does the ARC Simulator accept?

The ARC Simulator accepts the ASCII hexadecimal equivalent of

a binary executable file. That is, the file is composed of ASCII characters that represent the hexadecimal form of a binary file. This format was chosen so the user can view the executable code using any simple text editor.

Each entry in the first column represents the address at which the simulator will load the corresponding machine instruction in the second column.

In this document we will refer to these files as binary files, even though they are actually the ASCII representations.

These .bin files can be generated from the ARC Assembler accessible from the edit window which is brought up by pressing the edit button.

V. How do I run the simulator?

The buttons in the center of the window control the operation of the simulator. From left to- right, they are:

"Exit" exits the simulator by terminating the application.

"Load" allows the loading of a binary ARC file by bringing up the standard file opening dialog that your machine supports.

"Reload" will reload the current binary ARC file clearing the memory, program status register, and registers in the process.

"Edit" will bring up a text editor which will allow the creation of an ARC assembly program. Standard editing features are included. The "Assemble" button will activate the ARC Assembler on the program currently being modified in the text editor. The "Show Lst File" button on the text editor window will display the list file of the program being assembled in the text editor. The "Show bin File" button will show the binary file of the program being assembled in the text editor. The "Show Asm File" button will return the user to the .asm file after viewing the other file types.

"Step" executes the single machine instruction pointed to by the PC. Note that this instruction is displayed in the center of the screen.

"Run" runs the program beginning at the address pointed to by PC. Note that the machine will run until it encounters an arc stop instruction or a breakpoint. In the event that the program does not contain a stop instruction or a breakpoint, it can be stopped by pressing the "Stop" button.

"Clear BreakPts" and "Clear Memory" clear all breakpoints and all memory locations, respectively. Individual breakpoints can be set and cleared using the checkboxes in the center of the window. The contents of individual memory locations can be modified by editing the fields at the bottom of the window.

"Stop" stops execution of the simulator.

"Next" and "Prev" buttons are used to step through the machine code and memory display.

