Lab 2: Hardware and Software for Polled I/O

In this lab, you will build the low-level hardware and software support for input and output for the HERA processor. You will need to check out projects from the course shared repository.

Divide into groups of 2-3 people each, and work on one of the following projects. Coordinate with a group of people doing the other project, and produce a working hardware/software combination:

**HARDWARE.** Add a terminal to the HERA system you built in CS240. You may create new instructions for INPUT and OUTPUT, or use the “memory-mapped” approach (add the data transfer and status/control signals can be added as memory-mapped addresses, so that LOAD can be used to receive status information or transferred data, and STORE can be used to send control signals and data). Coordinate with the software group to set standards for the operations or addresses to be used and for the bits to be used for various status/control information.

**SOFTWARE.** Create functions to input or output a “tiger string”, i.e. a string represented by the lowest address of a sequence of memory cells, the first of which is a count of the number of characters, and the rest of which are the characters themselves, one per memory cell. The output function should be named `print` and print an entire string; for input you should provide two functions: `getchar` to read a single character and `getline` to read an entire line (up to a newline character, but not including that newline character in the input). (Optional: If anyone in your group has a working tiger compiler for HERA, compile some programs and try to run them.) If you are including the Tiger-stdlib-stack.hera library and it complains about lacking a `flush` function, just create one that does nothing for now.

You should use CVS to communicate between the groups and make periodic backup copies of your work. This will be a bit different from work you’ve done with CVS previously. Please

a) connect to the new repository `/home/courses/shared-class-repositories/cs356s13` to access the shared projects;

b) remember to do a Team→Update each time you log in to start work — if you forget to do so you may see the dreaded “conflicts during merge” message, in which case you should contact your professor :-(

c) do a Team→Commit each time you’ve completed some work — if you have (hopefully temporarily) broken something that had been working, send email to your classmates to warn them and let them know when you’ll fix it;

d) use clear but brief comments when you do a Team→Commit.