

Project Proposal

NanoVM On the ZNEO

February 27th, 2011
Nathan Scott

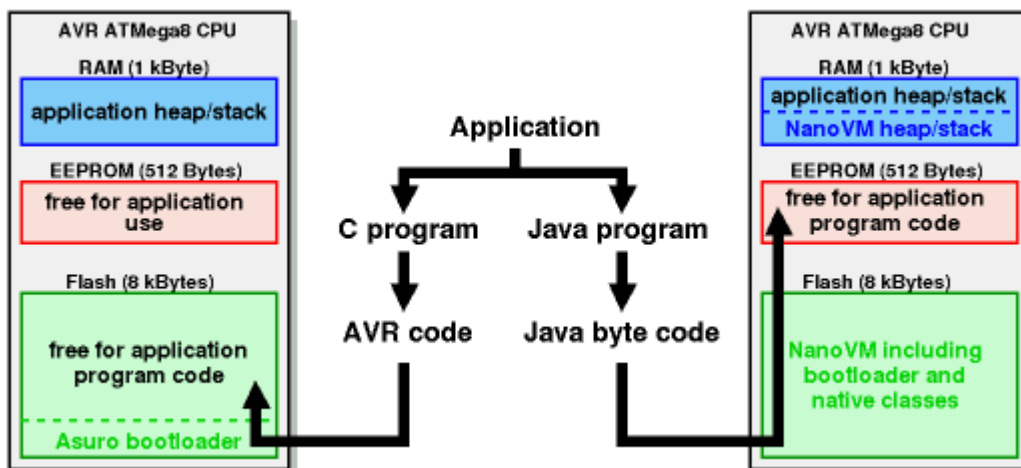
Project Abstract

NanoVM allows a small subset of Java to be used on microcontrollers by adding support for the Java virtual machine at a firmware level. The whole point of porting the software over is to allow quicker, easier prototyping in Java and the ability to nearly completely avoid the proprietary IDE that is shipped with the ZNEO. Naturally, the project needs to include hardware as well, not just a software port. The project will scope beyond just migrating NanoVM and also support a more user-friendly API to take advantage of the new VM features (e.g. print will go to the LEDs, simplified way to configure the ports, and so on. Less bit toggling).

Strategy

The Zilog ZNEO will be the primary board on this project.

Honestly, the amount of genuine software engineering here will be the only hurdle – and a large portion of that is already done, since the software has already been written. The biggest challenge will be reading through the Atmel AVR ATmega8 specification, making sure I understand what was already written, and porting only the appropriate parts and re-writing code where needed. I have already taken a peek into the source code, and luckily it does not look like the Linux kernel – although there *is* a lot of code, and the commenting seems sparse. For the actual application of the VM, I shamelessly took this picture from the NanoVM website that states how the typical application flow will be:



Unknowns

As stated above, the ATmega8 is something I have yet to personally tussle with. Also, I'm running on the assumption that NanoVM will fit into memory (if it does for the ATmega8 it should for the ZNEO as well, but sometimes these things do not work as expected). Also, considering it will be running the JVM, I need to worry about how programmed interrupts would mess with the program, and what interrupts I need to make sure become reserved. These are things I can't really anticipate until I see how the code is currently set up.

Implementation Plan

1. Get all the specification sheets for the ATmega8 and read through them.
2. Read through all the NanoVM docs
3. Make sure that serial is working on the ZNEO before continuing, as serial is how the Java programs will be loaded once NanoVM is onboard.
4. **Milestone 1:** Start digging into the code, with the first step successfully flashing it onto the board and getting serial to print "Hello"
5. **Milestone 2:** Verify that all ported libraries work by creating unit tests
6. Build abstracted API on top of that to automate simple tasks on the ZNEO and simplify port usage.
7. **Milestone 3:** Test aforementioned API

Resources

1. Zilog ZNEO – Provided by GWU
2. NanoVM software - <http://www.harbaum.org/till/nanovm/index.shtml>
3. ATmega8 Specifications - http://www.atmel.com/dyn/products/product_docs.asp?category_id=163&family_id=607&subfamily_id=760&part_id=2004