An Update on Building the RISC-V Software Ecosystem

Arun Thomas
BAE Systems
4th RISC-V Workshop, July 2016
Way Back in January

Let’s build out the RISC-V Software Stack in 2016.
Arun's 2016 RISC-V Wishlist

- Upstream support for GNU toolchain, clang/LLVM, and QEMU
- Upstream support for Linux kernel, Yocto, Gentoo, and BSD
- Debian/RISC-V port
Making Progress

• Upstream support for GNU toolchain, clang/LLVM, and QEMU
  • Getting ready to send patches for review
  • Let’s focus on upstreaming in the next 6 months
• Upstream support for Linux kernel, Yocto, Gentoo, and BSD
  • FreeBSD 11 will officially support RISC-V!
  • Let’s focus on Linux kernel, Yocto, and Gentoo
• Debian/RISC-V port
  • See Manuel’s talk tomorrow
Can we do better?
My Take in January

- Recruit developers
- Reduce startup costs
- Specs. Docs
My Take in July

Foundation should **fund** developers to build core SW infrastructure

We should decide on a **process** for proposing ISA enhancements. **Start now.**
Funding Developers
What Might These Folks Do?

- Upstreaming and Maintainership
- Porting software to RISC-V
- Performance optimization/analysis
- Enhancing test suites and methodologies
- Continuous integration
- Release management
Funding Models
Proposing ISA Enhancements
Standards, Drafts, RFCs, Proposals, ...
RISC-V RFCs

• A straw man proposal to spark discussion: github.com/arunthomas/riscv-rfcs.git

• RFCs are written in Markdown text format

• Submit RFCs via GitHub pull requests

• Discussion on isa-dev and GitHub
Requesting Comments

RFC 1: RFC Process

Introduction

As the RISC-V community grows, it will need to decide on a process for defining RISC-V standards and enhancements. One option is to use a GitHub repository such as this to capture proposals to enhance the RISC-V ISA or other aspects of the RISC-V project.

Proposal

- Use a GitHub repository to host RISC-V RFCs
Requesting Comments

The RISC-V Instruction Set Manual
Volume III: Privileged Architecture
Privileged Architecture Version 1.9
Document Version 1.9

Warning! This draft specification will change before being accepted as standard, so implementations made to this draft specification will likely not conform to the future standard.

Andrew Waterman, Yunsup Lee, Rinac Avanisie, David Patterson, Krite Anzovil
CS Division, EECS Department, University of California, Berkeley
{waterman,yunsup,riina,pattren,krite}@eecs.berkeley.edu
July 8, 2016

This document is also available as Technical Report ECR/EECS-2016-129.
Summary

• We are making progress on the software stack, but we could move faster

• Foundation should **fund** software developers

• Community should decide on a **process** for proposing ISA enhancements
Make ISAs great again.
Questions/Comments?

• What do you think we should be doing to build the RISC-V software stack?

• Let’s keep this discussion going at the workshop/committee meetings and on the mailing lists.

• Contact: arun.thomas@acm.org, @arunthomas