The RISC-V OS Landscape

Palmer Dabbelt: RISC-V Software Team Lead at SiFive, Software Group Vice Chair at RISC-V Foundation
DAC Workshop: RISC-V Ecosystem - Reshaping the CPU Landscape
The State of RISC-V Software
GNU-Based Toolchains

- binutils, GCC: May, 2017
- glibc: February, 2018
  - only supports rv64i-based ISAs
- newlib: August, 2017
- “Probably not a compiler bug”
### GNU-Based Toolchains

- **binutils, GCC: May, 2017**
- **glibc: February, 2018**
  - only supports rv64i-based ISAs
- **newlib: August, 2017**
- **“Probably not a compiler bug”**

---

<table>
<thead>
<tr>
<th>Processor</th>
<th>Modes</th>
<th>Frequency</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>RV32IAC</td>
<td>M + U Modes</td>
<td>Up to 1.5 GHz in 28nm</td>
<td>1.61 DMIPS/MHz</td>
</tr>
<tr>
<td>ARMv7-M, Thumb, Thumb-2</td>
<td>Not Publicly Available</td>
<td>1.25 DMIPS/MHz</td>
<td></td>
</tr>
</tbody>
</table>
RISC-V Linux Kernel Port

- **Linux: January, 2018**
  - Only RV64I-based systems
  - Drivers are trickling in now
GNU/Linux Userspace

- More Debian packages than Itanium!
Open Source Hardware

- Fedora runs on open source hardware
- Reproduced open-source FPGA shell
  - [http://github.com/sifive/freedom](http://github.com/sifive/freedom)
Open Standards Work!

- Kito Cheng (Andes Technology): GCC and newlib
- Jim Wilson (SiFive): binutils and GCC
- Darius Rad (Bluespec): glibc
- Andrew Waterman (SiFive): binutils, GCC, and glibc
- DJ Delorie (RedHat): glibc
RISC-V Implementations

- You need somewhere to actually run this software
- Lots of simulators available
  - QEMU: full system simulator
  - spike: RISC-V golden model
  - Imperas ModelSIM: commercial full-system simulator
  - AntMicro Renode: IOT focused simulator
  - rv8: Very fast simulator
  - RISCVEMU: Runs X in a browser! [https://bellard.org/jslinux/](https://bellard.org/jslinux/)
- Dev boards available as well
  - SiFive HiFive1
  - MicroSemi Mi-V
  - GreenWaves GAP8
  - SiFive HiFive Unleashed
Bare-Metal Embedded Development

- All software is part of the application
  - No RTOS, no Linux
- Users cross-compile the whole world
  - Drivers, application code, interrupt handlers, etc
- Binaries are not portable
  - Source may or may not be portable
- Lots of custom devices (and therefore drivers)
- Toolchain isn’t tied to the environment
  - Minimal usage of libc
- Generally driven by an IDE
- External debugger (JTAG)
Bare Metal SDKs

- [link] https://github.com/sifive/freedom-e-sdk
- Linker scripts
- Device drivers
- Interrupt code
- Example applications
Pre-Build Cross Compilation Tools

- https://www.sifive.com/products/tools/
- https://gnu-mcu-eclipse.github.io/
JTAG Debug for RISC-V Cores

- **Out of tree OpenOCD port**
  - [http://github.com/riscv/riscv-openocd](http://github.com/riscv/riscv-openocd)
  - Multi-core, 64-bit, etc
  - Quite buggy

- **Segger Embedded Studio**
  - Single core, 32-bit only

- **Lauterbauch TRACE32**
IDEs for RISC-V

- SiFive’s FreedomStudio
  - https://www.sifive.com/products/tools/

- MicroSemi’s SoftConsole
RTOS-Based Embedded Development

- A logical split between the RTOS and the application
  - May or may not be protected
- Generally come with their own toolchain
Zeyhpr

- [http://docs.zephyrproject.org/boards/riscv32/index.html](http://docs.zephyrproject.org/boards/riscv32/index.html)
- Upstream support for HiFive1
- Zephyr SDK comes with RISC-V toolchains
- Used in a real product
  - [http://badge.antmicro.com/](http://badge.antmicro.com/)
FreeRTOS

- Out of tree patches
  - Not well supported
- [https://forums.sifive.com/t/freertos/552](https://forums.sifive.com/t/freertos/552)
RTEMS

- [https://devel.rtems.org/wiki/TBR/Website/Board_Support_Packages#RISC-VBSPs](https://devel.rtems.org/wiki/TBR/Website/Board_Support_Packages#RISC-VBSPs)
- Upstream support for simulators
- Upstream support in GCC and binutils
- Out-of-tree support for HiFive1
Standard UNIX Distributions

- Software is portable
- System is self-hosting
  - Native tools
- Strong separation of privilege levels
  - Bootloader/firmware
  - Possibly a hypervisor
  - Kernel
  - Userspace
RISC-V FreeBSD Port

- Upstream since early 2016
- Latest RISC-V Privileged Spec
- Demonstrated booting on HW
Fedora

- [link](https://fedoraproject.org/wiki/Architectures/RISC-V)

- Self hosting
  - HiFive Unleashed
  - QEMU

- Build farms running

![Image of Fedora hardware and logos]
Debian

- https://wiki.debian.org/RISC-V
Hypervisors, Bootloaders, and Firmware

- Berkeley Boot Loader (BBL) is simple but well supported
- CoreBoot port is upstream
  - Doesn’t support HiFive Unleashed
- U-Boot port is upstream
  - Doesn’t support HiFive Unleashed, but it’s in progress
- TianoCore port is in progress
  - Boots on HiFive Unleashed
- RISC-V Hypervisor Extension is in progress
  - Lots of community involvement
Embedded Linux

- Linux, but binaries are generally not portable
- Userspace aimed at one specific application
- Tends to use a less featured firmware/bootloader
OpenEmbedded/Yocto

- [https://github.com/riscv/microblaze](https://github.com/riscv/microblaze)
- Port is in good shape
  - Supported upstream, runs X
  - Works on HiFive Unleashed and QEMU
- Khem Raj gave a talk at ELC
  - [https://www.youtube.com/watch?v=tdsmjgwjmq](https://www.youtube.com/watch?v=tdsmjgwjmq)
OpenWRT

- Port in progress
  - [https://groups.google.com/a/groups.riscv.org/forum/#!topic/sw-dev/OYOY1U nSwFc](https://groups.google.com/a/groups.riscv.org/forum/#!topic/sw-dev/OYOY1UnSwFc)

- Not upstream yet

- Runs in QEMU
Custom Accelerators on Linux

- RISC-V is exciting because you can build your own hardware

<table>
<thead>
<tr>
<th>inst[4:2]</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111 (&gt; 32b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>inst[6:5]</td>
<td>LOAD</td>
<td>LOAD-FP</td>
<td><strong>custom-0</strong></td>
<td>MISC-MEM</td>
<td>OP-IMM</td>
<td>AUIPC</td>
<td>OP-IMM-32</td>
<td>48b</td>
</tr>
<tr>
<td></td>
<td>STORE</td>
<td>STORE-FP</td>
<td><strong>custom-1</strong></td>
<td>AMO</td>
<td>OP</td>
<td>LUI</td>
<td>OP-32</td>
<td>64b</td>
</tr>
<tr>
<td>01</td>
<td>MADD</td>
<td>MSUB</td>
<td>NMSUB</td>
<td>NMADD</td>
<td>OP-FP</td>
<td><strong>reserved</strong></td>
<td><strong>custom-2/rv128</strong></td>
<td>48b</td>
</tr>
<tr>
<td>10</td>
<td>BRANCH</td>
<td>JALR</td>
<td><strong>reserved</strong></td>
<td>JAL</td>
<td>SYSTEM</td>
<td><strong>reserved</strong></td>
<td><strong>custom-3/rv128</strong></td>
<td>≥ 80b</td>
</tr>
</tbody>
</table>

Table 19.1: RISC-V base opcode map, inst[1:0]=11
AI Acceleration

AI Accelerator
MicroSemi Development Board
Open-Silicon AI Presentation

- TSMC OIP Theater in Booth 1629
- Tuesday, June 26, 11:30 - 11:45 a.m.