

Slides: <https://github.com/pdp7/talks/blob/master/rv-bof-elc.pdf>

RISC-V and Open Hardware BoF

Embedded Linux Conference 2020



Stephano Cetola <scetola@linuxfoundation.org>

Drew Fustini <drew@beagleboard.org>



RISC-V at ELC

- Linux on Open Source Hardware with RISC-V (Drew Fustini)
 - Tuesday, June 30 • 12:25pm - 1:15pm
 - <https://sched.co/c3Pn>
- RISC-V: Instruction Sets Want to be Free (Krste Asanovic)
 - Wednesday, July 1 • 10:35am – 10:55am
 - <https://sched.co/c3Yq>
- Go RISC-V Go: The State of Software Development Tools for RISC-V (Khem Raj)
 - Wednesday, July 1 • 12:15pm - 1:05pm
 - <https://sched.co/c3Yn>
- Ask the Expert Session with Calista Redmond, CEO of RISC-V International
 - Thursday, July 2 • 11:45am - 12:15pm
 - <https://sched.co/cosw>



Munich RISC-V Group

📍 München, Germany
👤 359 members · Public group ?
👤 Organized by Flo W. and 1 other

Share: [f](#) [t](#) [in](#)



Bay Area RISC-V Group

📍 San Jose, CA
👤 1,197 members · Public group ?
👤 Organized by Celeste Cooper and 4 others

Share: [f](#) [t](#) [in](#)

Many RISC-V meetups around the world!
<https://riscv.org/local/>



- **RISC-V: Free & Open RISC Instruction Set Arch**
 - “new instruction set architecture (ISA) that was originally designed to support computer architecture research and education and is now set to become a standard open architecture for industry”
 - The **5th** RISC instruction to come out out UC Berkeley



- **RISC-V: Free & Open RISC Instruction Set Arch**
 - Instruction Sets Want To Be Free: A Case for RISC-V
 - David Patterson, UC Berkeley – *co-creator of the original RISC!*
 - <https://www.youtube.com/watch?v=mD-njD2QKN0>
 - **RISC-V Summit 2019: State of the Union**
 - Krste Asanovic, UC Berkeley
 - https://www.youtube.com/watch?v=jdkFi9_Hw-c
 - Or watch the KEYNOTE THIS WEDNESDAY!

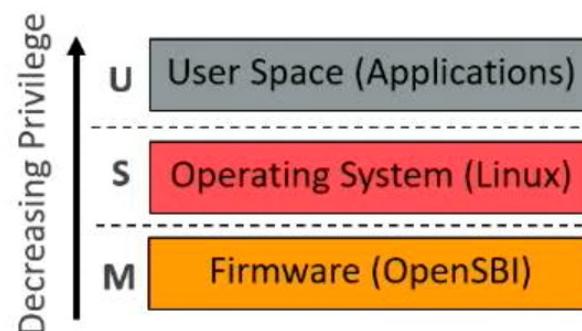
"Co-developing RISC-V Hypervisor Support" - Anup Patel

- <https://youtu.be/4GoikHB5Qcw>

What is RISC-V ?

Free and Open Instruction Set Architecture (ISA)

- Clean-slate and Extensible ISA
- XLEN (machine word length) can be 32 (RV32), 64 (RV64), and 128 (RV128)
- 32 general purpose registers
- Variable instruction length (instruction compression)
- Three privilege modes: Machine (M-mode), Supervisor (S-mode), and User (U-mode)
- Control and Status Registers (CSR) for each privilege mode



General Purpose Registers	
zero	Hardwired-zero register
ra	Return address register
sp	Stack pointer register
gp	Global pointer register
tp	Thread pointer register
a0-a7	Function argument registers
t0-t6	Caller saved registers
s0-s11	Callee saved registers

S-mode CSRs (Used By Linux)	
sstatus	Status
sie	Interrupt Enable
sip	Interrupt Pending
stvec	Trap Handler Base
sepc	Trap Program Counter
scause	Trap Cause
stval	Trap Value
satp	Address Translation
sscratch	Scratch

NOTE: **sdeleg**, **sideleg**, and **scounteren** not used currently

"Co-developing RISC-V Hypervisor Support" - Anup Patel

- <https://youtu.be/4GoikHB5Qcw>

RISC-V Extension

It is a set of ISAs and not one ISA

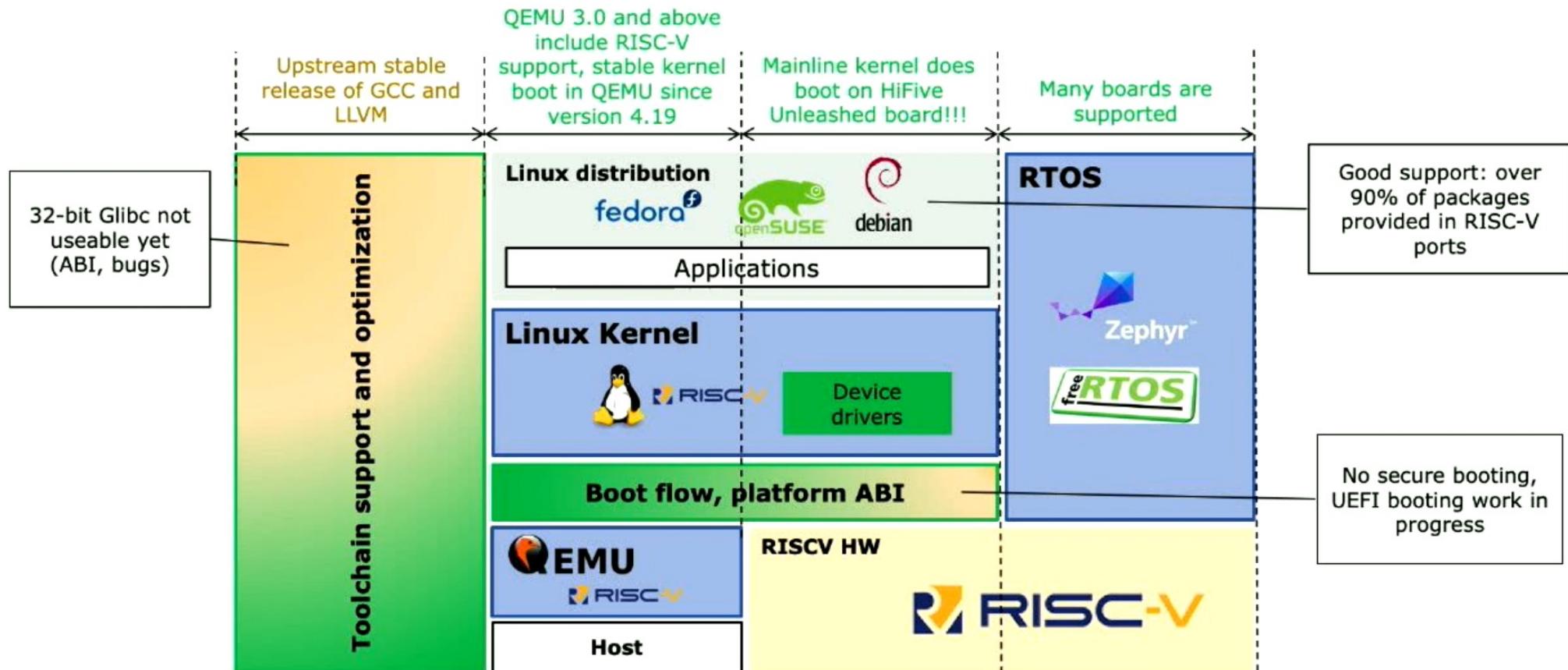
- RISC-V is a set of ISAs where each ISA is called an **RISC-V extension**
- **Standard naming notation** for ISAs supported by RISC-V implementation:
 - RV – Indicates RISC-V architecture
 - [###] – {32, 64, 128} Indicates register length
 - [abc..xyz] – Indicates set of extensions
- **Examples:** RV64IMA, RV64GC, RV32I, RV32IM
- **Linux needs:**
 - RV64IMAC
 - RV64IMAFDC (aka RV64GC)
 - RV32IMAC
 - RV32IMAFDC (aka RV32GC)

Extension	Description
I	Integer
M	Integer multiplication and division
A	Atomics
F	Single-precision floating point
D	Double-precision floating point
G	General purpose = IMAFD
C	16bit compressed instructions
H	Hypervisor
V	Vector
... And More ...	

"RISC-V software ecosystem in 2020" - Atish Patra (*LCA 2020*)
– <https://youtu.be/qwkab2Z44pk>

Linux Ecosystem: Status At-a-glance

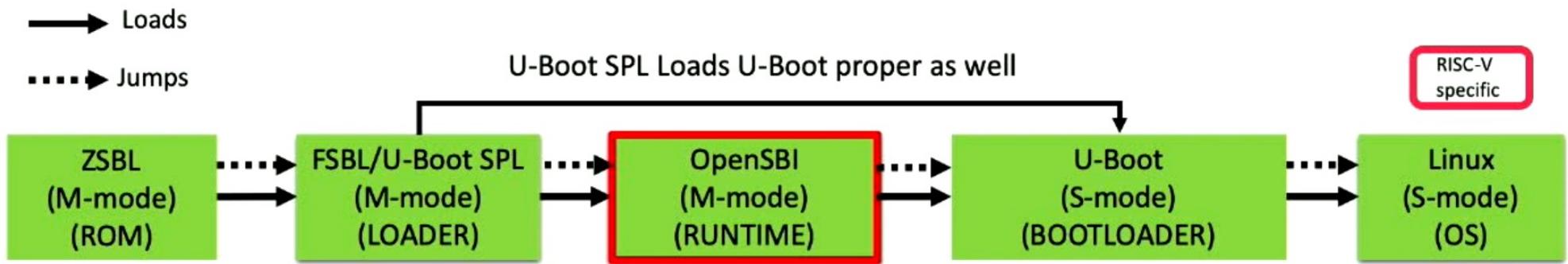
Fast progress but more work needed



"RISC-V software ecosystem in 2020" - Atish Patra (*LCA 2020*)
– <https://youtu.be/qwkab2Z44pk>

RISC-V Boot Flow

Follows commonly used multiple boot stages model



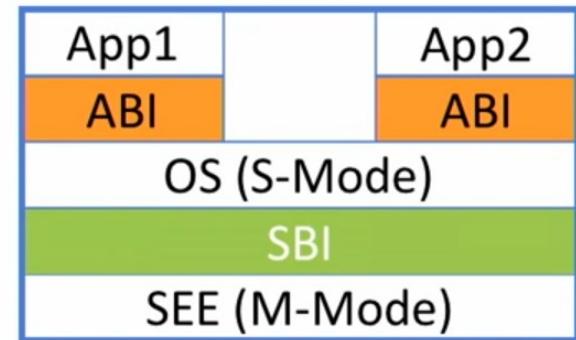
- Follows a standard boot flow
- Uses U-Boot proper as the last stage boot loader
- FSBL is SiFive specific and will be replaced by Coreboot/U-Boot SPL
- OpenSBI is a RISC-V specific runtime service provider
- All harts jump to Linux at the same time and a lottery-based approach chooses the booting hart

"OpenSBI Deep Dive" - Anup Patel (*RV Workshop 2019*)

- <https://youtu.be/jstwB-o9II0>

What is SBI ?

- SBI stands for RISC-V Supervisor Binary Interface
 - System call style calling convention between Supervisor (S-mode OS) and Supervisor Execution Environment (SEE)
- SEE can be:
 - A M-mode RUNTIME firmware for OS/Hypervisor running in HS-mode
 - A HS-mode Hypervisor for Guest OS running in VS-mode
- SBI calls help:
 - Reduce duplicate platform code across OSes (Linux, FreeBSD, etc)
 - Provide common drivers for an OS which can be shared by multiple platforms
 - Provide an interface for direct access to hardware resources (M-mode only resources)
- Specifications being drafted by the Unix Platform Specification Working group
 - Maintain and evolve the SBI specifications
 - Currently, SBI v0.1 in-use and SBI v0.2 in draft stage



"OpenSBI Deep Dive" - Anup Patel (*RV Workshop 2019*)
– <https://youtu.be/jstwB-o9II0>

What is OpenSBI ?

- OpenSBI is an open-source implementation of the RISC-V Supervisor Binary Interface (SBI) specifications
 - Licensed under the terms of the BSD-2 clause license
 - Helps to avoid SBI implementation fragmentation
- Aimed at providing RUNTIME services in M-mode
 - Typically used in boot stage following ROM/LOADER
- Provides support for reference platforms
 - Generic simple drivers included for M-mode to operate
 - PLIC, CLINT, UART 8250
 - Other platforms can reuse the common code and add needed drivers

UEFI support

- [RFC PATCH 00/11] Add UEFI support for RISC-V
 - Atish Patra (June 25, 2019)
 - This series adds UEFI support for RISC-V.
 - Linux kernel: 5.8-rc2
 - U-Boot: master
 - OpenSBI: master

"OpenSBI Deep Dive" - Anup Patel (RV Workshop 2019)

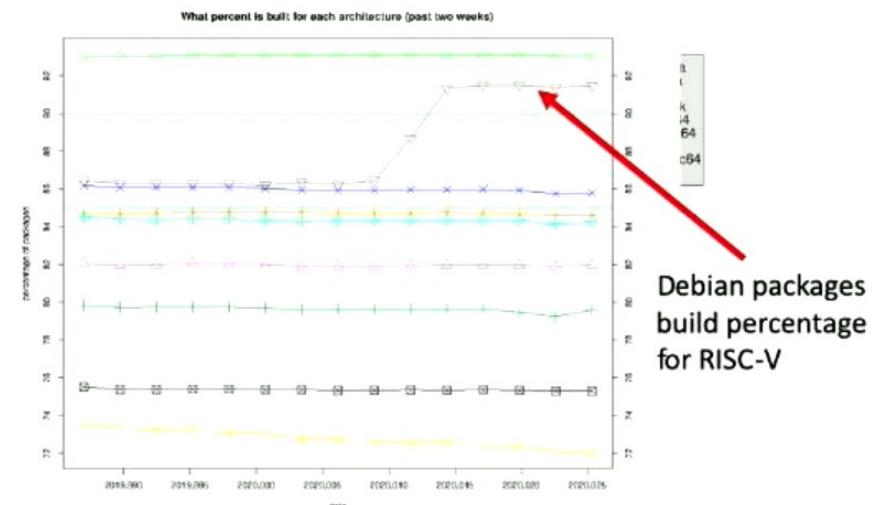
- <https://youtu.be/jstwB-o9lI0>

Linux Distributions

Fedora, Debian and OpenSuse are very active

- Fedora: stage 4 disk images released
 - <https://fedoraproject.org/wiki/Architectures/RISC-V>
 - Supports 3 boards + 142 QEMU VMs
 - Fedora 32/Rawhide including debug and source packages
- Debian
 - <https://wiki.debian.org/RISC-V>
 - ~92% of packages built
- OpenSuse Tumbleweed
 - <https://download.opensuse.org/ports/riscv/>
- OpenEmbedded has RISC-V support
 - Demos of running Plasma Mobile and Mycroft are available
 - <https://github.com/riscv/meta-riscv>

RISC-V graphical desktop on Fedora



"OpenSBI Deep Dive" - Anup Patel (*RV Workshop 2019*)

- <https://youtu.be/jstwB-o9II0>

QEMU

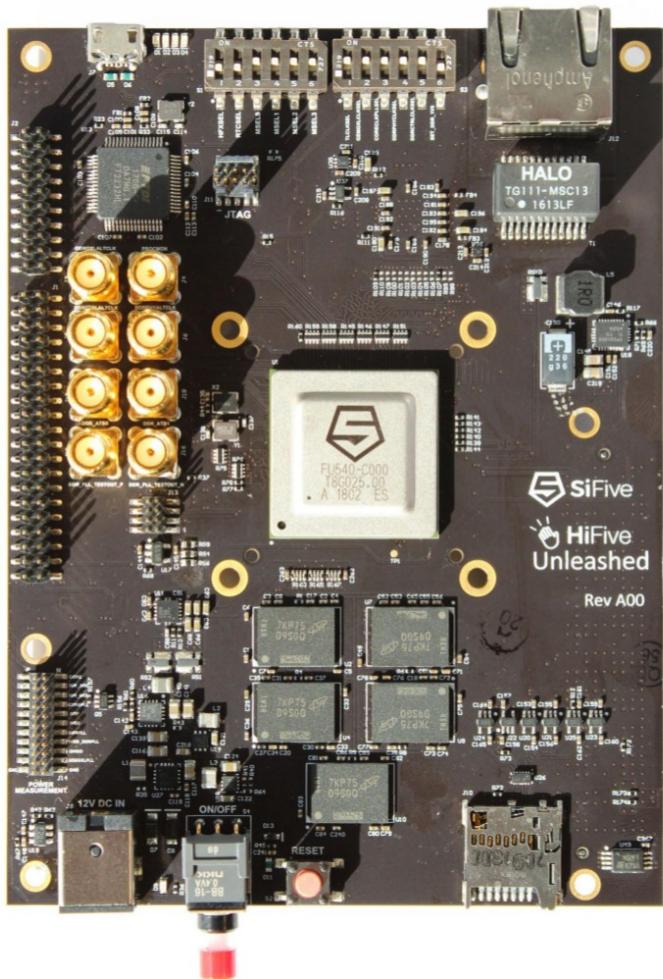
In very good shape

- Use mainline QEMU ! No forks !
- QEMU can boot 32-bit and 64-bit mainline Linux since kernel 4.19
- QEMU can run OpenSBI, U-Boot and Coreboot
- OpenSBI v0.5 included with QEMU
- Hypervisor extensions v0.5 supported
 - Patches waiting for review
 - Specifications not frozen yet
- Vector extensions
 - Patches are in mailing list
 - Specifications not frozen yet
- Newest QEMU 4.2 brings many improvements
 - QEMU sifive_u machine can now boot same binaries that run on HiFive Unleashed
 - SPI flash on virt machine and HiFive Unleashed modelled. This allows FSBL (oreboot and coreboot) testing on QEMU
 - PMP access improvements
 - Support loading initrd for sifive_u machine

SiFive: Linux on RISC-V



HiFive Unleashed

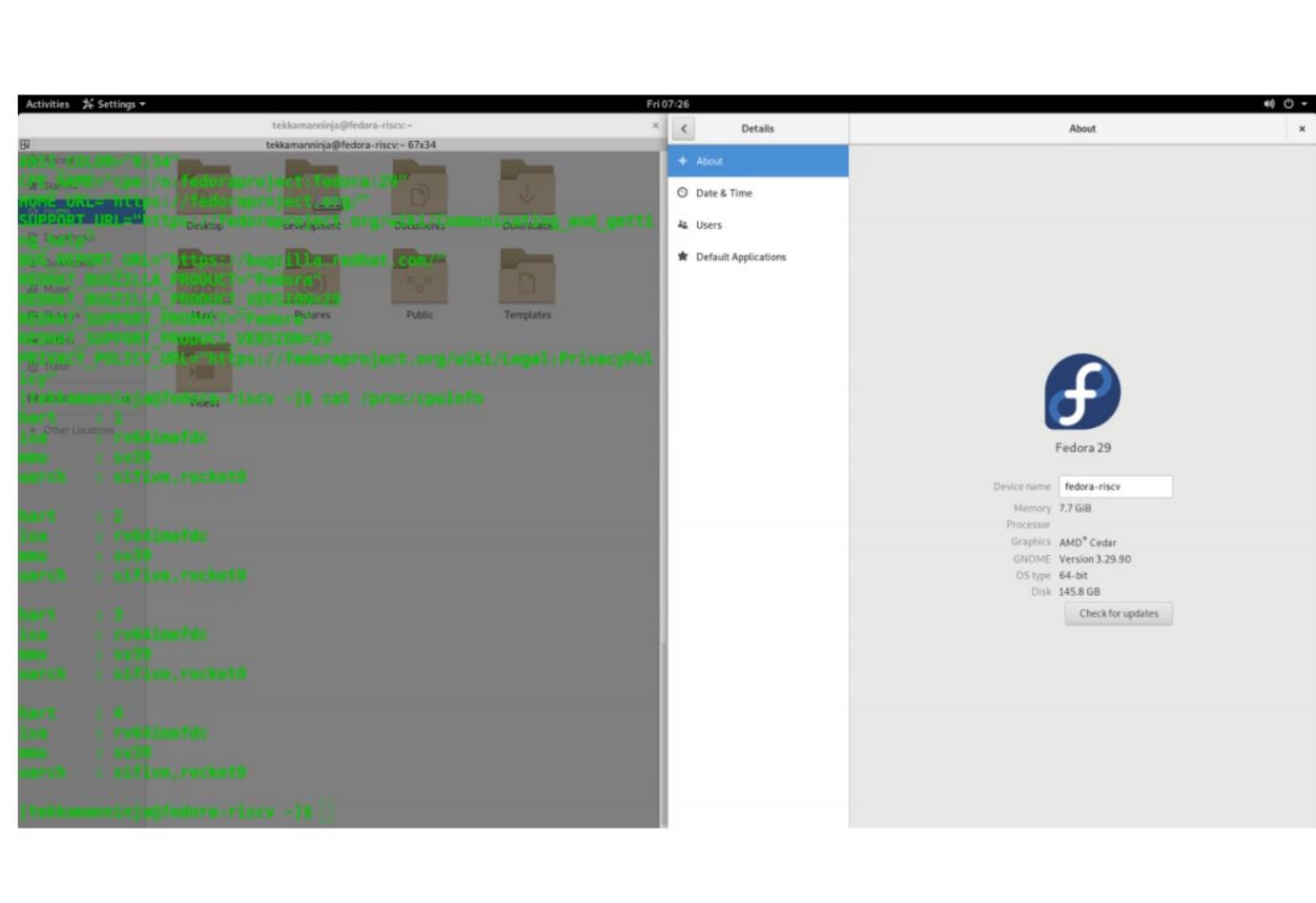
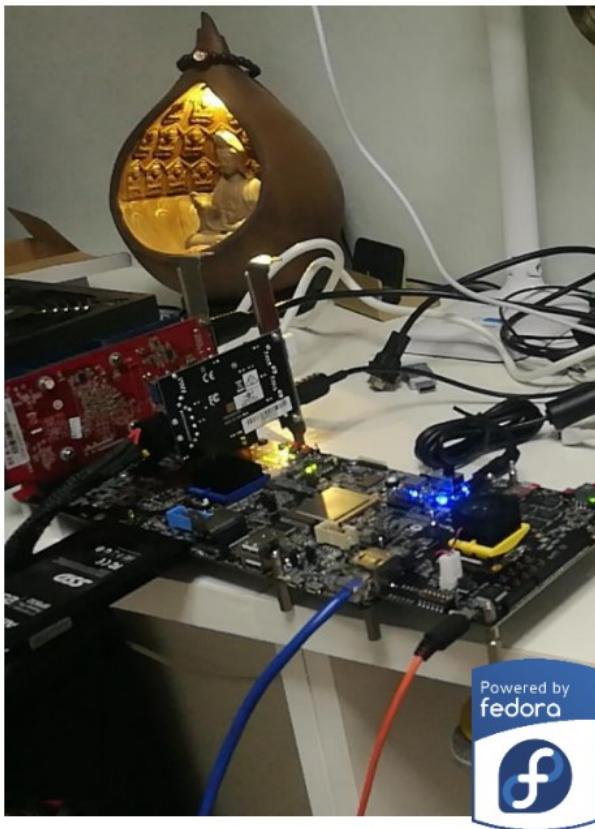


- World's First Multi-Core RISC-V Linux Development Board
 - SiFive FU540-C000 (built in 28nm)
 - 4+1 Multi-Core Coherent Configuration, up to 1.5 GHz
 - 4x U54 RV64GC Application Cores with Sv39 Virtual Memory Support
 - 1x E51 RV64IMAC Management Core
 - Coherent 2MB L2 Cache
 - 64-bit DDR4 with ECC
 - 1x Gigabit Ethernet
 - 8 GB 64-bit DDR4 with ECC
 - Gigabit Ethernet Port
 - 32 MB Quad SPI Flash
 - MicroSD card for removable storage
 - FMC connector for future expansion with add-in cards

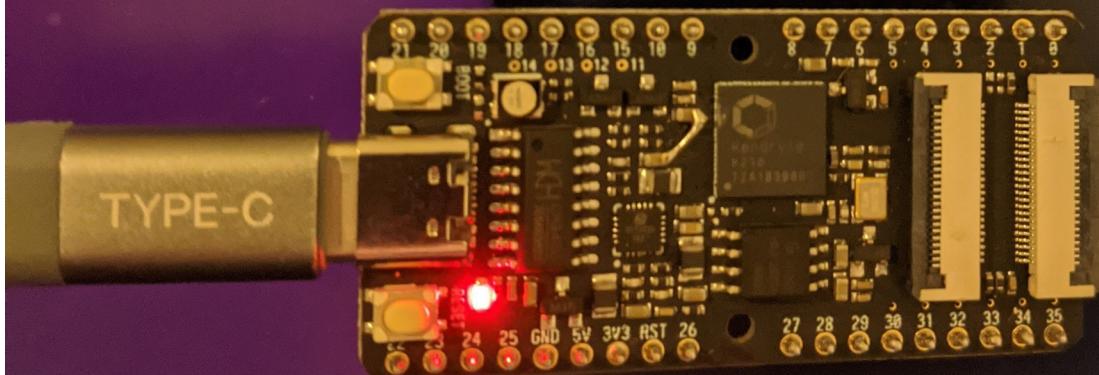
“Linux on RISC V Fedora” Wei Fu (RV Summit 2019)

- <https://www.youtube.com/watch?v=WC6e3g8uWdk>

Fedora GNOME Image on SiFive Unleashed

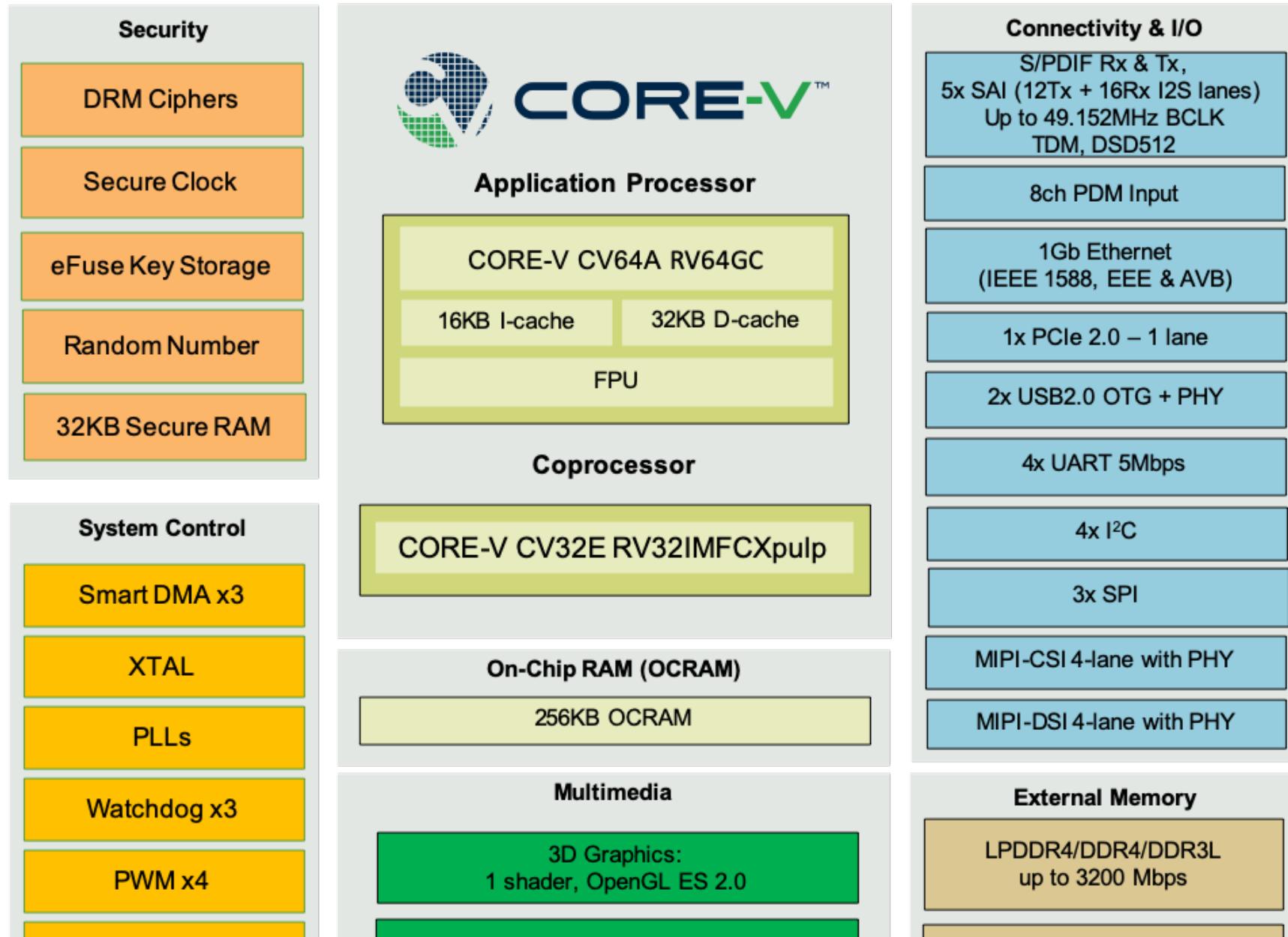


```
File Edit Log Configuration Control signals View Help  
/ # uname -a  
Linux k210 5.6.0-rc1vowstar #1 SMP Mon Feb 17 23:  
/ # cat /proc/meminfo |head  
MemTotal:           6656 kB  
MemFree:            2496 kB  
MemAvailable:       2080 kB  
Buffers:             0 kB  
Cached:              1916 kB  
SwapCached:          0 kB  
Active:              0 kB  
Inactive:            0 kB  
Active(anon):        0 kB  
Inactive(anon):      0 kB  
/ # cat /proc/cpuinfo  
processor      : 0  
hart          : 0  
isa           : rv64imafdc  
  
processor      : 1  
hart          : 1  
isa           : rv64imafdc  
  
/ # tcc -run -nostdlib hello.c  
hello.c:2: warning: implicit declaration of function  
hello show 'n tell  
/ #
```



- **Kendryte K210**
 - dual core 64-bit RISC-V at 400MHz with 8MB SRAM
 - Sipeed MAix BiT for RISC-V is only \$13!
 - Damien Le Moal at Linux Plumbers 2019
 - RISC-V NOMMU and M-mode Linux
 - Full support coming in Linux 5.8
 - Buildroot with busybox (<https://git.io/JJfIC>)
 - need NOMMU/FDPIE support for better userspace
 - <https://youtu.be/GydykyNjxs> (Maciej W. Rozycki)
 - 8MB runs out very quick!
 - there is a MMU but the wrong spec :(
 - u-boot patch series
 - [PATCH v14 00/20] riscv: Add Sipeed Maix support
 - Sean Anderson (June 24th)

- OpenHW Group Announces CORE-v Chassis SoC
 - similar to NXP iMX but with RISC-V cores
 - tape-out 2nd half of 2020





- Keynote at Hackday Supercon 2019 by Dr. Megan Wachs of SiFive
- **“RISC-V and FPGAs: Open Source Hardware Hacking”**
 - https://www.youtube.com/watch?v=vCG5_nxm2G4

Open Source toolchains for FPGAs

- Project Trellis for Lattice ECP5
 - “Project Trellis and nextpnr FOSS FPGA flow for the Lattice ECP5”
 - David Shah (@fpga_dave)
 - youtube.com/watch?v=0se7kNes3EU

Project Trellis and nextpnr FOSS FPGA flow for the Lattice ECP5

Project Trellis & nextpnr

FOSS Tools for ECP5 FPGAs

David Shah
@fpga_dave
Symbiotic EDA || Imperial College London

FOSDEM 19
org



Open Source ECP5 boards

- Radiona.org ULX3S

- <https://www.crowdsupply.com/radiona/ulx3s>

The screenshot shows the Crowd Supply campaign page for the ULX3S board. The top navigation bar includes links for CROWD SUPPLY, BROWSE, LAUNCH, ABOUT US, and a search bar. On the right, there are links for Open Hardware and Development Kits, along with a user profile icon.

The main content area features the title "ULX3S" by Radiona.org / Zagreb Makerspace. Below it, a sub-headline reads "A powerful, open hardware ECP5 FPGA dev board".

A detailed diagram of the ULX3S board is shown in the center, with various components labeled:

- USB 1 connected to FTDI FT231XS
- 3.5 mm audio jack with 4 contacts (analog stereo + digital audio or composite video)
- GPDI (digital video, audio, ethernet ready)
- USB 2 connected directly to FPGA
- 25 MHz onboard, external differential clock input
- 8 user LEDs
- 2 fire buttons
- Lattice ECP5 LFE5U-85F-6BG381C (85K LUT)
- 32MB SDRAM
- ADC: 8 channels, 12 bit, 1 MSa/s MAX11125
- 4 directional buttons
- Placeholder for 0.96" SPI color OLED display
- 2 USB LEDs
- 1 WiFi LED
- 00:15 SD slot

On the right side of the page, the campaign statistics are displayed:

- \$50,793 raised
- of \$15,000 goal
- 338% Funded!
- 5 updates
- 8 days left
- 300 backers

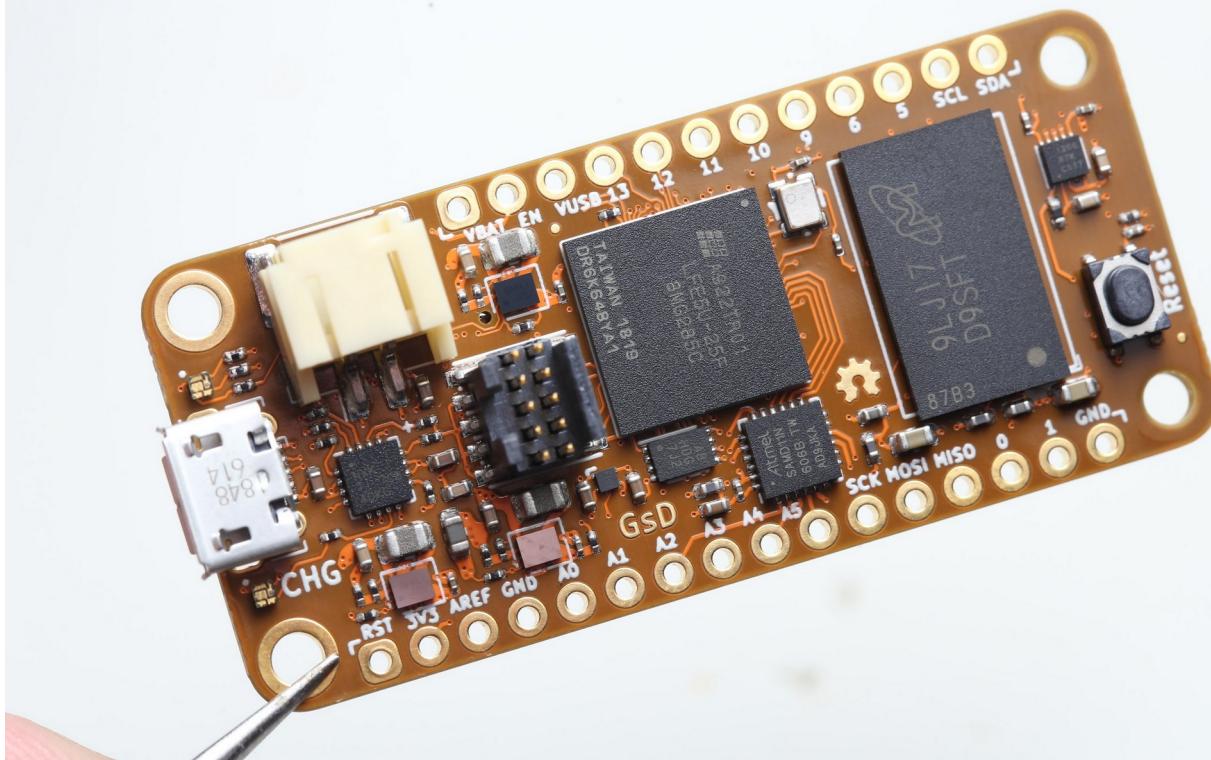
Below the stats, a message says "Last update posted Apr 03, 2020". There is also a field for entering an email address ("me@example.com") and a "Subscribe to Updates" button. Social media icons for Facebook and Twitter are present.

At the bottom, a section titled "PMOD Set" is shown with a price of \$36.

Recent Updates [View all 5 updates](#).

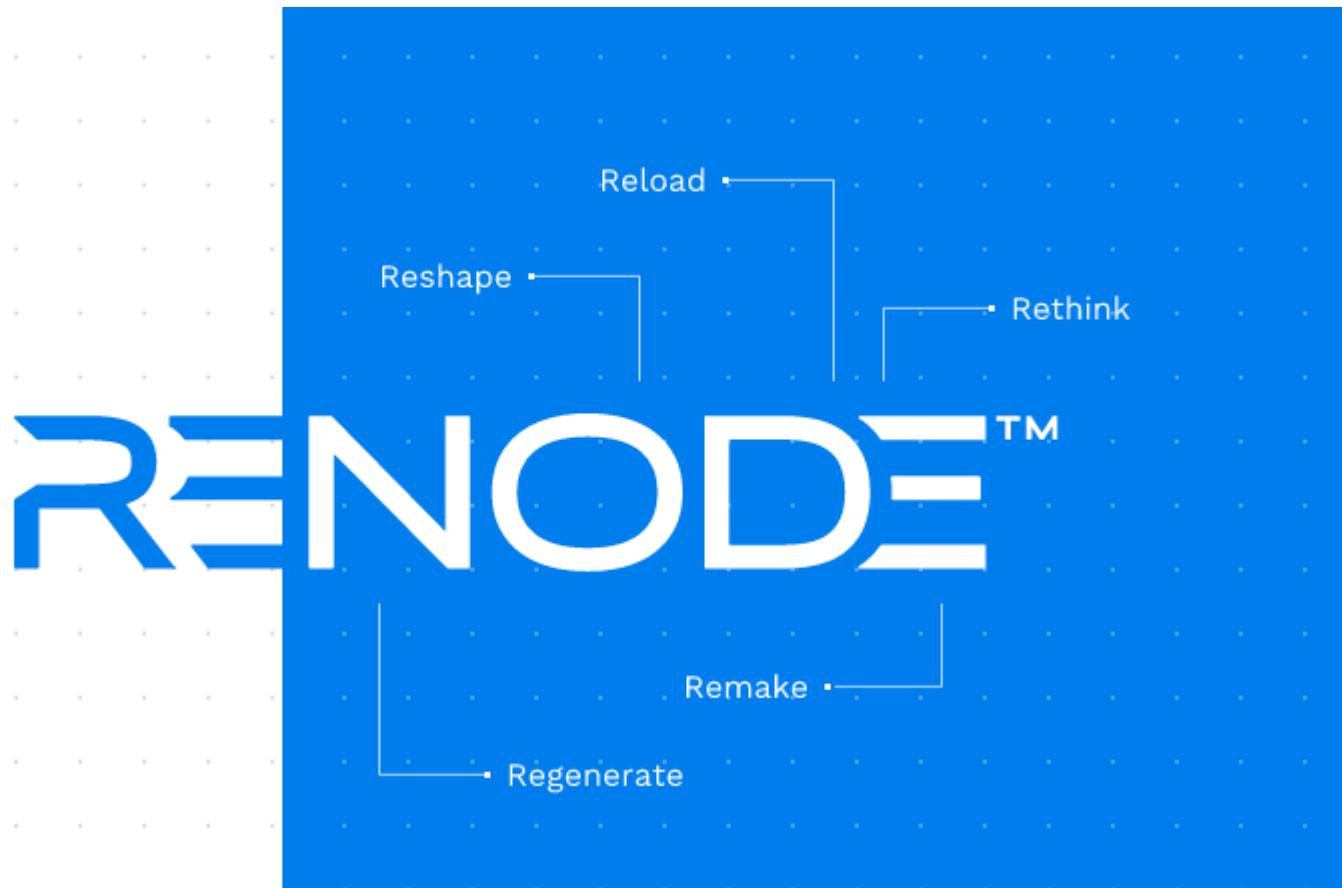
Open Source ECP5 boards

- Lattice ECP5 FPGA in Adafruit Feather form factor and 128MB DDR RAM:
 - Orange Crab by Greg Davill
 - <https://github.com/gregdavill/OrangeCrab>
 - <https://groupgets.com/campaigns/710-orangecrab>



No hardware? Try Renode!

- <https://renode.io/>



Activitie

RE Renode ▾

```
SiFive RISC-V Coreplex
[0.000000] OF: fdt: Ignoring memory range 0x80000000 - 0x80200000
[0.000000] Linux version 4.15.0-80044-g2b0aa1de45f6 (houen@bakura) (gcc version 7.2.0 (GCC)) #5 SMP Wed
[0.000000] bootconsole [early0] enabled
[0.000000] Initial ramdisk at: 0x (ptrval) (9593856 bytes)
[0.000000] Zone ranges:
[0.000000]   DMA32    [mem 0x0000000080200000-0x000000008ffffffff]
[0.000000]   Normal   [mem 0x0000000900000000-0x000008ffffffffffff]
[0.000000] Movable zone start for each node
[0.000000] Early memory node ranges
[0.000000]   node 0: [mem 0x0000000000000000-0x0000000000000000]
[0.000000]   node 1: [mem 0x0000000000000000-0x0000000000000000]
```

```
(hifive-unleashed) Sbin?=@http://antmicro.com/proj
.elf-s_17219640-c7e1b920bf81be4062f467d9ecf689dbf7f
(hifive-unleashed) $fdt?=@http://antmicro.com/proj
icetree.dtb-s_10532-70cd4fc9f3b4df929eba6e6f22d02e6
(hifive-unleashed) $vmlinux?=@http://antmicro.com/p
-vmlinux.elf-s_80421976-46788813c50dc7eb1a1a33c1736
(hifive-unleashed)
(hifive-unleashed) macro reset
> """
>     sysbus LoadELF $bin
>     sysbus LoadFdt $fdt 0x81000000 "earlyconsole"
>     # Load the Linux kernel symbols, as they are
>     sysbus LoadSymbolsFrom $vmlinux
>     # Device tree address is passed as an argument
>     e51 SetRegisterUnsafe 11 0x81000000
> """
(hifive-unleashed) runMacro $reset
(hifive-unleashed) start
Starting emulation...
(hifive-unleashed)
```

Linux kernel status

- <https://lore.kernel.org/linux-riscv/>
- Newly supported:
 - eBPF JIT
 - kgdb
 - kexec/kdump
 - generic vDSO support
 - Syszcaller
 - Build with clang

Linux kernel status

- Upcoming:
 - UEFI boot
 - CLINT driver
 - KVM
 - RFC: support raw event and DT for perf on RISC-V
 - CPUFreq support for FU540 was posted
 - (not merged yet)

“What's missing in RISC-V Linux, and how YOU can help!”

- Björn Töpel for last week for Munich meetup
- https://docs.google.com/presentation/d/1vfrVIWKYWHSw6Q5PDQBn6PC5jwXet4HPxN3f_nxtFm4/edit?usp=sharing

