



Development of Fedora Linux Distribution for RISC-V (RV64G) Architecture

Surendra Billa Arif Badar Rushikesh Jadhav Yogeshwar Sonawane Sanjay Wandhekar
{surendrab, jrushikesh, arifb, yogeshwars, sanjayw} @ cdac.in



Centre for Development of Advanced Computing, Pune, India, www.cdac.in

Abstract

The RISC-V architecture is gaining momentum due to its open-source nature and flexibility; however, Linux distributions for some non-mandated extensions remain scarce. This gap became evident during our efforts to develop a Fedora Linux distribution for the RV64G, distinct from the commonly supported RV64GC. Initially, we used Buildroot to create an embedded Linux, excluding compressed instructions and testing on QEMU. However, the lack of a native compiler limited package installation. To address this, we employed Linux From Scratch (LFS) to build a minimal Linux system for RV64G, which highlighted the absence of a full-scale distribution. Fedora’s current RISC-V support targets only RV64GC. Recognizing this limitation, we started developing a complete Fedora distribution for RV64G. Building on prior work, we configured the Koji build system to rebuild Fedora RPMs and ported the Linux kernel with UEFI support. After thorough testing, we produced a stable and scalable solution, offering a valuable reference for RISC-V developers targeting non-standard platforms.

Introduction

- RISC-V's RV64G variant is underserved by mainstream Linux distributions, unlike RV64GC, which has broader support. RV64G lacks a tailored comprehensive operating system.
- Motivated by this gap, we initiated the development of a Fedora-based Linux distribution explicitly for RV64G.
- Starting with lightweight builds, we progressed to native compilation, kernel tuning and scalable packaging.
- This poster presents our methods and insights to assist others working on unconventional RISC-V platforms.

Methodology

- To develop a Fedora Linux distribution for RV64G, we followed a structured, multi-phase process ensuring functionality and scalability.
- Creating a File System Hierarchy**
We established a robust filesystem hierarchy following the Filesystem Hierarchy Standard (FHS).
 - Cross-Compiler Development**
A cross-compilation toolchain for RV64G was built by configuring targets and compiling *Binutils*, *GCC* and *Glibc*.
 - Target Image Preparation**
BusyBox was cross-compiled for *init*, and the kernel configured for RISC-V with storage and network support.
 - Native GCC Integration**
We cross compiled GCC and *Binutils* to enable native package builds in the RV64G environment.
 - Koji Build System for Fedora RISC-V**
A Koji build system was set up using an x86 hub, QEMU instances and a *StarFive* board as a builders.
 - Linux Kernel Configuration**
Kernel changes enabled EFI support and PE32+ headers, ensuring boot via *bootefi* or GRUB with non-compressed instruction set compatibility.
 - Bootable Disk Image Creation**
Kickstart file and appliance creator generated a bootable disk image, including all rebuilt packages and configurations for deployment on RISC-V.

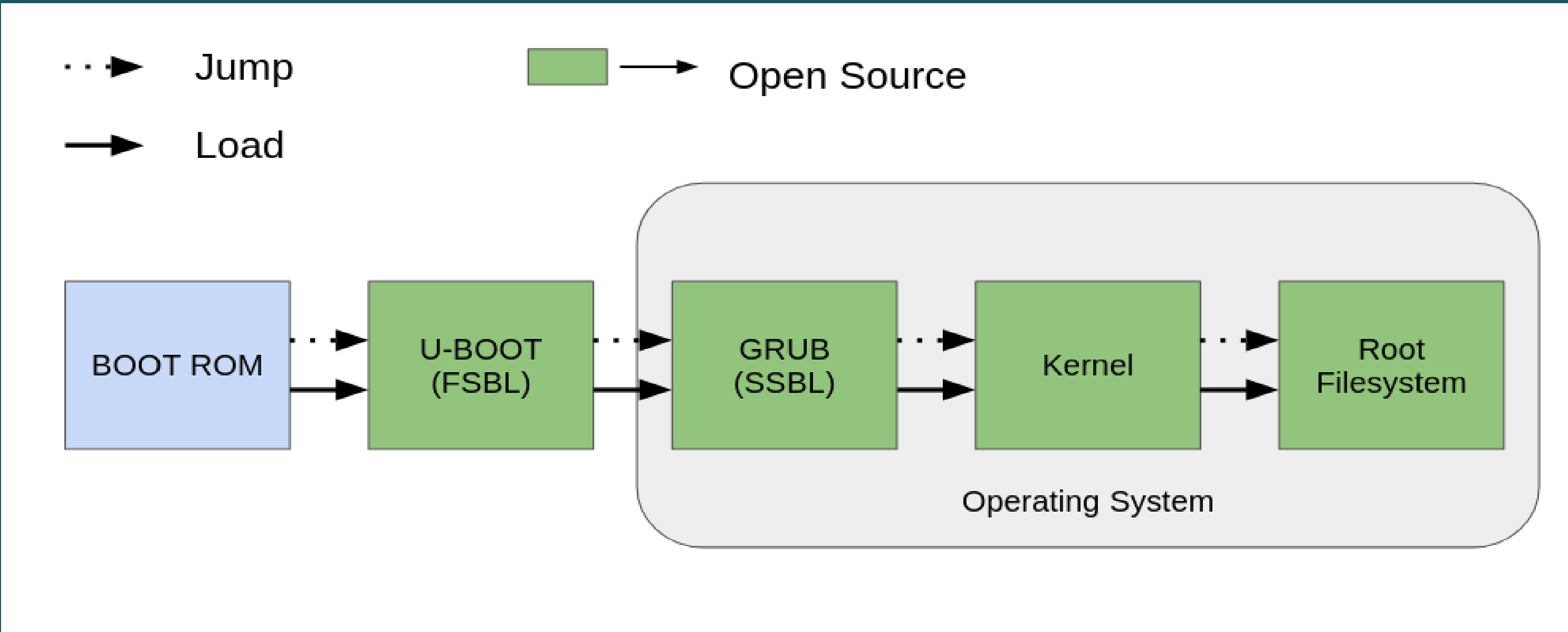


Fig [B] : Booting OS using u-boot and grub

Performance Evaluation

- A. QEMU Testing
- Booted successfully without compressed instructions
- B. Hardware Testing (C-DAC Vega & StarFive Boards)
- Custom kernels booted with stable performance.
- C. Application Testing
- Verified functionality using Dhrystone, CoreMark and HPL.

Conclusion

- Outlined key milestones in developing Fedora Linux for RV64G.
- By bridging key gaps in the current RISC-V ecosystem, this work delivers a scalable and maintainable solution that fosters broader adoption of RISC-V technology.
- The techniques are adaptable to other Linux distributions, making the work broadly useful for future RISC-V OS development.

Future Work

We plan to expand our work by developing a Fedora server version of our Linux distribution tailored for the RISC-V architecture, with a focus on rebuilding High-Performance Computing (HPC) RPM packages.

References

- 1) Surendra Billa et al. “Development of Fedora Linux Distribution for RISC-V (RV64G) Architecture”. In: SC24-W:Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2024, pp. 1685–1689. doi: 10.1109/SCW63240.2024.00210.



Fig [A] : RISC-V (RV64G) Koji Build System: Packages List