



Bringing Tier-1 support for 64-bit RISC-V Linux to Rust

RISC-V Summit Europe 2024

Contents

1

Speaker's
introduction

2

About the
project

3

Rust's Tier
System

4

Progress so
Far

5

Remaining
Progress

6

Going
Forward

7

Join In

Speaking Today

Joshua
Zivkovic

ROLE

Software Engineer at
Codethink

Lukas
Wirth

ROLE

Software Engineer at
Ferrous Systems



Also on the Project

Ana
Hobden

ROLE

Software Engineer at
Ferrous Systems



About the Project

Spearheaded by the RISE Project

Why Rust?

- Rust is repeatedly the most desired language among developers
 - Rust is becoming more and more prevalent amongst embedded and safety-conscious projects
 - Rust is becoming more and more prevalent in dependencies which you may be using and not even realising already.
-

Why RISC-V/Linux with Rust?

- More and more, Linux is entering embedded spaces previously the domain of QNX, etc.
- As more application processors are made, RISC-V is entering the space traditionally occupied by Arm A-class and similar processors



The RISE Project

Rust's Tier System

Rust's tiers show the level of support for a platform

Tier 3

Builds are not guaranteed to succeed

Tier 2

Builds succeed

riscv64gc

Tier 1

Builds and tests succeed

Tier 1 Linux support is currently limited to:

i686

X86_64

Aarch64

riscv64gc

Progress so far

- ✓ **Determined failing tests**
 - debuginfo-emit-llvm-ir-and-split-debuginfo.rs
 - riscv-abi/call-llvm-intrinsics.rs
 - riscv64-lp64d-abi.rs
 - riscv64-lp64f-lp64d-abi.rs

- ✓ **Opened and merged PR to fix codegen tests**

- ✓ **Opened and merged PR to fix and update riscv64gc-linux test job**

- ✓ **Opened and merged PR to fix bug discovered in the previous PR**

Codegen PR



Job PR



Bug Fix PR



Remaining Progress

Enable RISC-V testing in upstream CI

Establish a maintenance team for RISC-V
in Rust going forwards

Investigate `catch-unwind.rs`

May involve LLVM upstream changes

Raise the RFC to request the move to Tier 1

Potential future work might include
considering what might be needed to support
other RISC-V triplets on Linux

Going Forward

Complete the upstream CI setup

Open RFC for RISC-V to be considered as tier 1 and begin discussion of any missing parts.

Beyond this project, maybe further operating system support, or further combinations of extensions.

Join In



Use Rust nightly to build for riscv64gc
-unknown-linux-gnu

```
$ rustup default nightly  
$ rustup target add riscv64gc-unknown-linux-gnu  
$ cargo build ...
```



Report any problems you find

<https://github.com/rust-lang/rust/issues>



Test your own project on RISC-V



Come and find us at the poster
sessions to discuss how this affects
your use of Rust



Thank You.

Codethink Ltd.

3rd Floor Dale House,
35 Dale Street,
MANCHESTER,
M1 2HF,
United Kingdom

