RISC-V SAIL and ACT Survey Summary
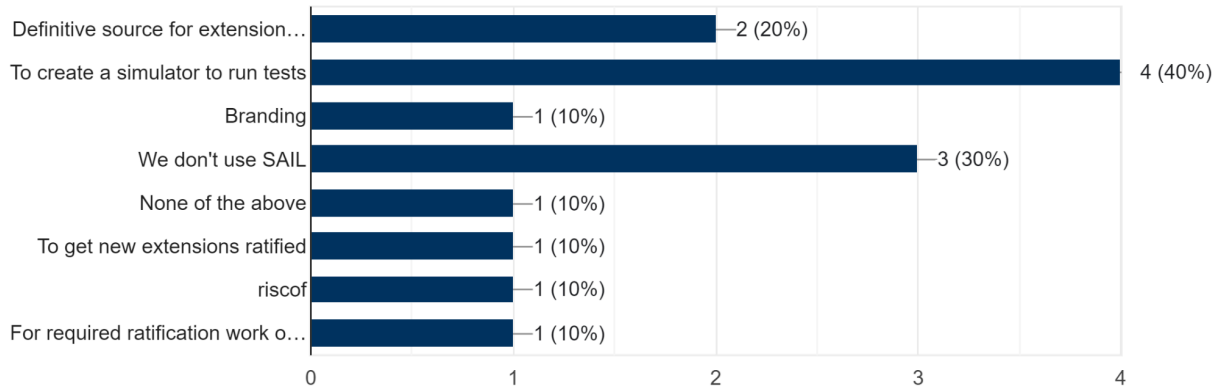
The following slides are from the January 2024 survey (link). No identification data has been included.
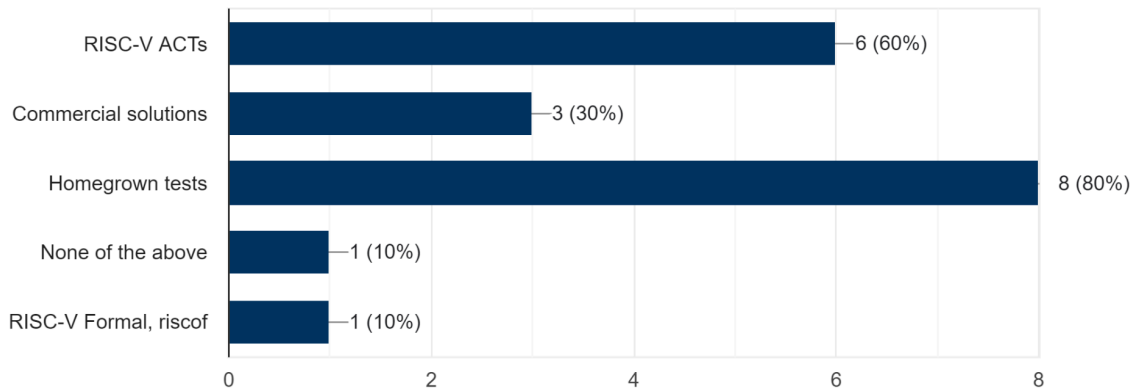
Why do you currently use or intend to use the RISC-V SAIL definition (check all that apply):
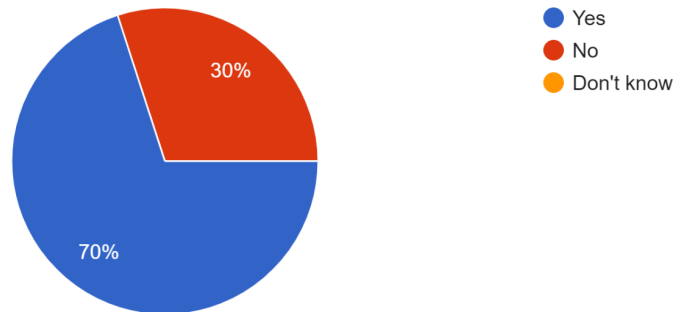
10 responses

What do you use to test your design for RISC-V compatibility (check all that apply):
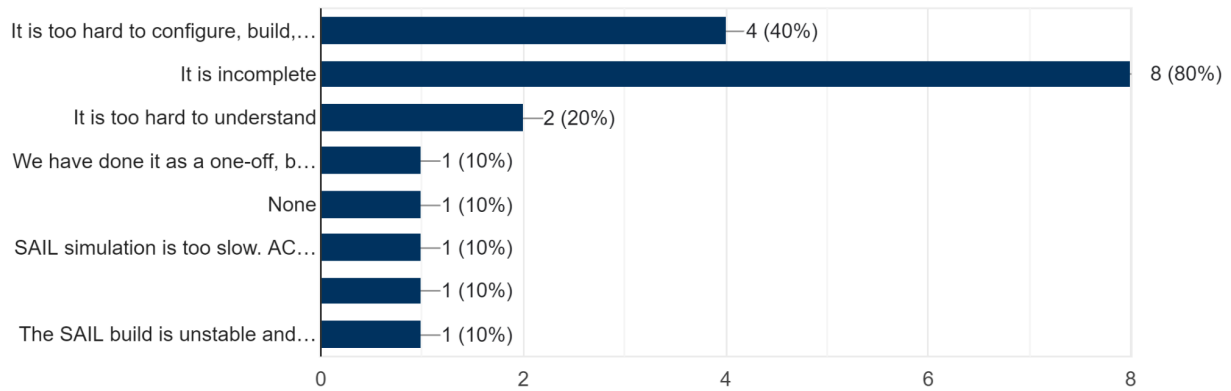
10 responses

Is "RISC-V Compatible" branding important to your company?

10 responses

- ● Yes
- ● No
- ● Don't know

70% Yes, 30% No

What, if anything, is stopping or slowing you from using RISC-V SAIL & ACTs to very compatibility (choose all that apply):

10 responses

Option	Count
It is too hard to configure, build,…	4 (40%)
It is incomplete	8 (80%)
It is too hard to understand	2 (20%)
We have done it as a one-off, b…	1 (10%)
None	1 (10%)
SAIL simulation is too slow. AC…	1 (10%)
	1 (10%)
The SAIL build is unstable and…	1 (10%)

SAIL does not have features that support all RISC-V extensions (e.g. WMO, caches, Atomicity, ...).
Are these features that we need to resolve? Are these important to your Company:

10 responses

- Yes
- No
- Don't know
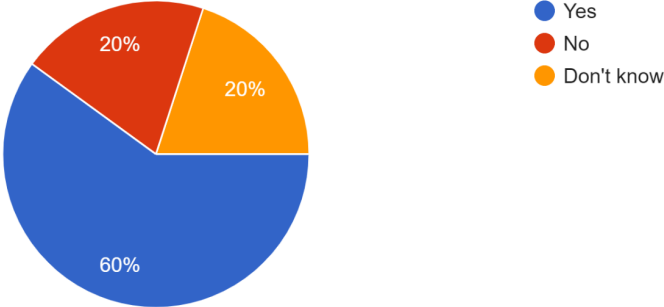
Do you have other comments regarding RISC-V SAIL or ACTs?

Unfortunately, I have not had a great experience with SAIL and ACT. I have found the software very complex to setup, often needing to add significant debug output to find issues related to setup to get to the bottom of confusing error messages. For example, one time I ran into a YAML parsing issue that said line ~12000 had poor spacing, but gave no path or filename. I searched and no files were of that length. Additional debugging showed it was concatenating files and then parsing.

The other thing is that ACT and the testcase generator are split into many repos with generated code checked in. This makes it a pain to try to apply external patches and get them working when trying to do follow-on work. If everything were in one repo with reliable CI that would be a big improvement imho.

SAIL and ACT support need to be part of "DoD" for extensions prior to Freeze. No more exceptions or grandfathering. The Privileged architecture also needs to be included. For architectural features with non-determinism, it may be acceptable to arrive at an alternative compliance framework instead of using the SAIL reference (e.g., RVWMO via implementing a suitable testbench using https://github.com/litmus-tests/litmus-tests-riscv).

We effectively use Spike as the human-readable implementation reference rather than SAIL, as the architecture specification tends to lack sufficient pseudocode. The project to incorporate SAIL pseudocode into the architecture specification was promising, and would provide a unified human-readable reference. That would require completion of the SAIL model for the entire architecture and for new extensions while they are being developed.

Major bugs go unfixed for years (https://github.com/riscv/sail-riscv/issues/49); major features completely missing (hypervisor). The Sail model feels like an academic plaything that nobody in industry actually uses (except my team, and lately only barely). Major feature development is done by incompetents (https://github.com/riscv/sail-riscv/pull/139; also 140 and 149). ACTs cover only the most trivial behavior. I have to wonder if the entire effort can ever be fruitful.

It is difficult to find people that know how to write SAIL.

Nothing

We also avoid SAIL because we haven't seen a credible plan from RVI to address its technical debt and maintain it as the standard evolves.

ACTs are a minor component to our compatibility validation.
Since SAIL is falling behind, and the infrastructure requires a golden model for signature checking, requiring it seems like it will continue to lessen the impact of the ACT suite. What role should SPIKE be playing in the infrastructure?