

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

Name: Architectural Compatibility Test Policy
Version: 1.2
One line description: These are the rules that explain the rationale for, and the requirements for running and reporting results of RISC-V architectural tests, and conditions for waiving failures.
Author(s): Allen Baum, Ken Dockser
Status: Approved
Source: [link](#)

Logo:



Version History

Ver	Date	Details	Email
1.2	2024-01-16	Change escalation path to occur through CTO	jeff@riscv.org
1.1	2021-09-30	added discussion about non-standard extensions.	allen.baum@esperantotech.com
1.0	2021-03-20	Initial version	allen.baum@esperantotech.com , Ken Dockser

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

Rationale:

RISC-V Architectural Tests (ACT) are an evolving set of tests that are created to help ensure that SW written for a given RISC-V Profile will run on all implementations that comply with that profile.

These tests also help ensure that the implementer has both understood and implemented the specification.

The RISC-V Architectural Tests test suite is a minimal filter. Passing the tests and having the results approved by RISC-V International is one of the prerequisites to being listed as RISC-V ISA Compatible. Details of RISC-V trademark usage may be found on the RISC-V International Web page at <https://riscv.org/>.

Passing the RISC-V Architectural Tests does **not** mean that the design complies with the RISC-V Architecture. These are only a basic set of tests.

The RISC-V Architectural Tests are **not** a substitute for rigorous design verification; it is the responsibility of the implementer to deploy extensive testing.

Who Must Test:

Anyone who develops RTL or a functional model for a RISC-V core and wants to be listed as RISC-V ISA Compatible (e.g., in order to use the RISC-V Compatible trademarks) needs to run the tests and file the resulting test report as a step in the RISC-V Compatible listing process.

An entity (e.g., a value-added reseller) who has licensed core IP from a developer and does not alter that IP can rely on the developer's RISC-V ISA Compatible listing. This includes cases where the entity has selected configuration parameters from a range of values provided by the developer, as long as the core with those chosen parameters is listed as RISC-V Compatible. An entity that has otherwise modified the IP must get the new design listed as RISC-V ISA Compatible by passing the appropriate RISC-V ACT and by self-certifying that the changed design is RISC-V compatible.

What Must be Tested:

The RTL or functional model of the core design must run and pass all of the tests within the ACT that RISC-V International says are required for the targeted profile.

In any of these cases, implementers must provide a simulation environment (RTL or functional) with sufficient resources to run the required ACT tests in the framework that selects tests, configures the reference model to generate appropriate reference signatures, and generates a test report based on comparing the reference model's signature with the implementation's signature file. These simulation environment resources include, but are not necessarily limited to: sufficient memory capacity to load and run the tests, the reference model, the architectural test repository, the tools needed to compile, link, load and run the tests, the ability to respond to

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

the signal that the test is complete, and the ability to extract the resulting test signature from the memory and save it in a file.

Unless specified otherwise by RVI, the reference model to be used is the Sail RISC-V Sequential Emulator produced from the most recently released Sail Formal Specification of the RISC-V ISA. When the Sail Formal Specification doesn't yet fully support all of the necessary features of the ISA, RVI may specify that the SPIKE emulator is to be used. In some rare cases, RVI may specify that the QEMU emulator is to be used. In all cases, RVI will specify which versions of the emulators to use.

Each release version of ACT will document the version of the toolchain utilities required to support the instructions used for that version of the tests in a repository README file. This will also include which reference model to use for the test as well as any other environmental requirements (e.g., amount of RAM).

Where to Report Test Results

The ACT framework will generate a test report file with the name

<instanceID>-<test-date>.html with the <test-date> in YYYY-MM-DD format using GMT as the time zone.

The <instanceID> is a string that identifies which version of an implementer's core is being tested..

In addition to pass/fail indications for each individual test that is run, the report will include:

- For any failures, the test case that failed, the expected value, and the actual value found,
 - The ISA string that describes the ISA, extensions, and sub-extensions implemented,
 - Any optional feature and configuration parameters allowed by the architecture that are used, defined by listing a YAML formatted file using the schema defined by the riscv-config format (see <https://riscv-config.readthedocs.io/en/latest/overview.html>),
 - The vendor and implementation IDs that the DUT will report in those respective CSRs (should be in the YAML feature/config file),
 - Note that these may be zero if unimplemented
 - Name, commit hash, and either version tags or git commit date (in ISO 8601 UTC format w/ offset 00:00)
- of tools used :
- Toolchain
 - reference model
 - Architecture Compatibility Test (ACT) suite ,
 - riscv-config (for v3 of the framework)

Vendors can submit their test results by filing a Pull Request that adds the test report into a directory named for the vendor in the <https://github.com/riscv/riscv-arch-test-reports> repository

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

When to Report Test Results

To retain their RISC-V ISA Compatibility listing, cores must pass the ACT yearly if ACT tests, toolchain, reference simulators, or framework change from those used to generate the test report, and issue errata if new issues are found.

Test-Case Waivers:

There are some special cases where it is not possible for an implementation to pass all of the tests due to issues that may be outside of the implementer's control. A Test-Case Waiver is special permission granted to exclude a test-case (or in some situations, test cases) from the required tests. This allows implementations to conditionally pass the ACT as long as all non-waived test cases pass.

A waiver request must be filed as an issue in the test-report github repository specifically requesting the waiver and mentioning the Pull Request number of the filed test report. In all cases, the appropriate ISA Committee and the SW Committee (or other appropriate non-ISA committee) must approve using a majority vote to approve each waiver for the specific design and test. The approval will be documented by filing it into the same test-report github repository with a name containing the same PR number as the waiver.

Waivers are categorized into Flawed-Test, Constraint-Based, and Errata-Based types.

Flawed-test test-case waiver:

If there is a demonstrable bug in the test, reference model, or tool chain that causes a test to fail, a Flawed-Test Test-Case Waiver may be requested. Flawed-Tests include cases where the reference model produces a result that differs from what the implementation produces, and that result is one of several results that is considered correct by the rules of the ISA (e.g. WARL behavior that tests don't support).

To be clear, a waiver is not to be requested --- and will not be granted --- due to

- a bug in the design.
- Cases that result from toolchain version that has not been tested to work with the version listed as being used in the reference model test-report for the most recent test-suite.
- cases where there is an ambiguity or misinterpretation of the architecture. Cases that involve architectural ambiguity should be resolved by clarification in the specification.

Constraint-based test-case waiver:

Constraint waivers may be granted in special cases including when:

- 1) An implementation has architecturally valid configurations that interfere with the ability of tests to run at reset
- 2) An Implementation establishes locked architecturally valid configurations after reset that interfere with the ability of tests to run.

In order to qualify for and be granted a constraint-based waiver:

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

- The implementer must document the constraint and its rationale to TSC, and demonstrate that it is an architecturally valid configuration.

Errata-based test-case waiver:

Errata-based waivers may be granted in special cases including when:

- 1) An implementation fails an existing architectural test
- 2) An implementation has previously passed architectural tests, but fails a subsequently released test after production or within 6 months before product manufacturing release, such that fixing the design would cause an undue hardship to the implementer (e.g., the design is in final manufacturing stages).
- 3) An implementation has a known architectural flaw that is not covered by an architectural test (in which case a test for that case should be added to the test suite if possible).

1) a RISC-V Compatible listed design either

- a) fails subsequently released ACT tests, or
- b) is reported to have an architectural incompatibility, or

2) a design was given a test-case waiver, fails the new test due to a bug, but the new test-case was not made available in a reasonable amount of time such that fixing the design would cause an undue hardship to the implementer (e.g., the design is in final manufacturing stages).

In order to qualify for and be granted an errata-based waiver,

- The implementer must provide the appropriate ISA committee (i.e., privileged or unprivileged) and SW or other HC with a detailed *erratum* (as to be defined in the forthcoming Errata Policy Document) covering the bugs resulting in any and all failures of test cases.
- the implementer must properly classify each individual erratum in the errata as “low impact” (to be defined in the Errata Policy, until then defined as: easily worked around with minimal impact to general performance and to SW complexity), and provide a remediation process, or justification of why remediation is unnecessary.

If approved, the implementer is required to publish the errata on the RISC-V website (and/or other appropriate location as determined by the TSC) and make it otherwise readily available to users and coders. The RVI Board and Marketing organization and TSC will be informed of the decision by the ISA committee and the approval or disapproval filed in the same github folder as the waiver request. If the implementer wants to continue the non-conformant behavior, then they must change the designation of that instanceID from compatible to custom. If they will be conformant in the next revision of their product, they can mark themselves as compatible with errata.

o *(this must be mentioned in branding or ACT policy)*

Note: Errata-based waivers do not transfer to subsequent physical implementations (e.g., the physical design is substantially changed) as these are considered new designs; implementers are expected to use this opportunity to fix known bugs.

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

Test-case waiver conditions

When a test-case waiver is granted, it

- only applies to a specific test case, not to an entire test
- only applies to the version of the design to which it was granted
- only lasts until the test case has been corrected or replaced (flawed-test waiver) at which point the design must pass the corrected test or request an errata-based waiver.

In no circumstances shall a test-case waiver be viewed as a waiver of an architectural requirement that is not subsequently relaxed in the spec. Furthermore, no software or hardware may rely on the behavior of the design in the waived test case other than to determine if the design contains the issue.

If any test-cases are granted waivers, and all other required test cases and tests have passed, the design will be considered to have *conditionally* passed. This will allow the design to move forward in the RISC-V Compatible listing process. For flawed-test waivers, once a replacement test is available, the design will need to be retested and must pass all tests to change the *conditional pass* to a *pass*. If the test continues to fail, then it is an errata and subject to the errata-based waiver process.

All waivers must be approved by a majority vote of the appropriate ISA Committee (i.e., privileged or unprivileged) and the SW or other related non-ISA HC..

Test Case Failures due to ISA restrictions or ambiguity

If a design fails on a test case, and the implementer believes that the failure is due to either an ambiguity in the ISA or that the ISA is inappropriately restrictive in the allowed behavior, the implementer should make a request to the appropriate ISA Committee to have the architecture amended. No waivers will be granted for such a case. However, if the architecture is subsequently changed, the appropriate tests will also be changed.

It is important to keep in mind that changes to the semantics of the ISA must go through the entire ratification process. Clarifications to the ISA that don't change the semantics are subject to a lighter-weight process that is beyond the scope of this document.

Architectural Ambiguity cases are very sensitive; if unintended behavior is allowed it might result in fragmentation of the architecture. These cases must undergo the utmost scrutiny by appropriate experts to avoid any unintended consequences.

It is incumbent upon the implementer to run the ACT early enough in the design process so that any failures can be investigated and design changes incorporated. Likewise, any issues with the tests, including unexpected results from the reference design (especially in the case where more than one result can be considered correct) need to be brought to the attention of the appropriate ISA Committee as soon as possible. That said, the implementer needs to have performed enough verification on the design before attempting to run the ACT, such that the ACT is not used to find bugs. The rationale here

RISC-V Architectural Compatibility Test:

Rationale, Requirements, Reporting and Waivers

is the ACT is a spot check intended to find major flaws in a verified design; any failures in the ACT point to a major gap in the understanding of the architecture or the design verification process.

Exceptions:

Exceptions to the test requirements are handled through the waiver process and changes to the ISA as mentioned above.

Implementers releasing non-standard extensions must label them as “X” extensions as per the unprivileged specification and even though they may fully pass ACT, any support for software ecosystem components will be only supported through vendor efforts and not through RISC-V.

Any escalation should occur to the CTO who may choose to resolve the issue, or escalate to the TSC or the CEO or the BOD.