Architectural Test Task Group Call – Minutes

Thur, 11Feb2021 8am Pacific →Standard← Time

See slide 6 for agenda

Antitrust Policy Notice

RISC-V International meetings involve participation by industry competitors, and it is the intention of RISC-V International to conduct all its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

Examples of types of actions that are prohibited at RISC-V International meetings and in connection with RISC-V International activities are described in the RISC-V International Regulations Article 7 available here: https://riscv.org/regulations/

If you have questions about these matters, please contact your company counsel.

RISC-V International Code of Conduct

RISC-V is a free and open ISA enabling a new era of processor innovation through open standard collaboration. Born in academia and research, RISC-V ISA delivers a new level of free, extensible software and hardware freedom on architecture, paving the way for the next 50 years of computing design and innovation.

We are a transparent, collaborative community where all are welcomed, and all members are encouraged to participate.

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone.

https://riscv.org/risc-v-international-community-code-of-conduct/

(old) Charter

The Compliance Task Group will

- Develop compliance tests for RISC-V implementations, taking into account approved specifications for:
  - Architectural versions (e.g. RV32I, RV32E, RV64I, RV128I)
  - Standard Extensions (H,S,U,A,B,C,D,F,J,K,M,N,P,Q,T,V,N), Priv Mode <red is ratified >
  - All spec'ed implementation options
    - (incl. MHSU modes, optional CSRs, optional CSR bits)
- Develop a method for selecting and configuring appropriate tests for a RISC-V implementation, taking into account:
  - Platform profile and Execution Environment (EE)
  - Implemented architecture, extensions, and options
- Develop a framework to apply the appropriate tests to an implementation and verify that it meets the standard
  - test result signature stored in memory will be compared to a golden model result signature

Adminstrative Pointers

- Chair – Allen Baum allen.baum@esperantotech.com Co-chair – Bill McSpadden bill.mcspadden@seagate.com
- TG Email tech-compliance@lists.riscv.org ← now changed to sig-arch-test@lists.riscv.org
    - Notetakers: please send emails to allen.baum@esperantotech.com
- Meetings -Bi-monthly at 8am Pacific time on 2nd/4th Wednesdays.
    - See https://docs.google.com/spreadsheets/d/1L15_gHl5b2ApkcHVtpZyl4s_A7sgSrNN zoom link
- Documents, calendar, roster, etc. in
    - https://lists.riscv.org/tech-compliance/ see /documents & /calendars subdirectories ← changed to sig-arch-test
    - https://drive.google.com/drive/folders/1DemKMAD3D0Ka1MeESRoVCJipSrwiUlEs (lifecycle in ”policies/supporting docs” folder, gaps in “planning” folder, compliance specific in “compliance folder”)
- Git repositories ←docs riscv → tools
    - https://github.com/riscv/riscv-compliance/tree/master/doc/ tests https://github.com/riscv/riscv-compliance/ ← to be changed
    - https://riscof.readthedocs.io/en/latest/index.html riscof https://gitlab.com/incoresemi/riscof/
    - https://riscv-isac.readthedocs.io/ ISA coverage https://gitlab.com/incoresemi/riscv-compliance/riscv_isac ←TBC
    - https://riscv-ctg.readthedocs.io/ Test Gen. https://gitlab.com/incoresemi/riscv-compliance/riscv_ctg ←TBC
    - https://github.com/riscv/riscv-config/tree/master/docs YAML, WARL config https://github.com/riscv/riscv-config/
    - https://github.com/rems-project/sail-riscv/tree/master/doc Sail formal model https://github.com/rems-project/sail-riscv ← to be chgd
- JIRA: https://jira.riscv.org/projects/CSC/issues/CSC-1?filter=allopenissues
- Sail annotated ISA spec: in https://github.com/rems-project/riscv-isa-manual/blob/sail/
    - README.SAIL ←how to annotate annotated unpriv spec→ release/riscv-spec-sail-draft.pdf
    - release/riscv-spec-sail-draft.pdf ← annotated source annotated priv spec→ release/riscv-privileged-sail-draft.pdf
    - https://us02web.zoom.us/rec/share/-XIYazzhIBbQoiZdarCfebdjxjDWiVhf-LxnuVrliN4Bc30yf17ztKkKDU4Og54b.fArPPqnuR-NiXpQU Tutorial
      Access Passcode: tHAR#5$V

Meeting Agenda

0. **Looking for more admins, maintainers for riscv-compliance git repo !!**

I. Updates, Status, Progress:

1. Election Cancelled (without an armed mob !) and transition to SIG
2. Next meeting: cancelled or moved? (conflict with all-hands meeting)
3. Rename of TG and Repo to “arch-test” – group name will change after this meeting, repo to follow later

II. Next steps and Ongoing maintenance

1. SIG Charter discussion – slide 8
2. The 3 W’s of testing: Who, What, and Waivers
3. Close github issues as a result of repo v2.1
4. Maintenance updates to V2 to enable future tests
   a) update RVTEST_SIGUPD to keep automatically adjust base/hidden offset when offset>2K,
   b) Enable use Sail model results as the assertion value
   c) add assertion macros for FP, DP, Vreg to arch_test.h and test_format spec
   d) add trap handlers for S, VS modes
5. Migration to Framework v.3.0 (riscof). video: https://youtu.be/VIW1or1Oubo, slides: https://lists.riscv.org/g/tech-compliance/files/Presentations/TestFormatSpec.pdf
   - What steps do we need to complete to cut over to V.2 (see slide 13)
     - (e.g. Sail model updates, pipecleaning, N people have run it, testing all the “fixed in riscof” issues
     - Review Pipecleaner tests:What do we need to do to exercise capabilities for Priv Mode tests
6. Tests for non-deterministic result (see attached discussion in email)
   1. Provide a reference RTL test fixture (as opposed to SW functional model). See. JIRA CSC-6
   2. Define hooks for concurrency tests
7. Specific Compliance Policy/Process Gaps:
   a. Identify Tool providers, e.g. coverage model, test generation for new features/extensions

Repository, Group Name Change

- Group will be renamed next Thursday (after group meeting)
  - From: tech-compliance
  - To: sig-arch-test
  - All current membership, mailing list, files, messages, and chairs unchanged
- Mailing list will change
  - From: tech-compliance@list.riscv.org
  - To: sig-arch-test@list.riscv.org
  - Minutes from next weeks meeting will be sent from that address
- Next step is changing the repository name from
  - From: https://github.com/riscv/riscv-compliance
  - To: https://github.com/riscv/riscv-arch-test
  - A dummy repo with old name will be set up with README pointing to new name
  - Will also provide instructions on how to switch/add a new origin to local cloned github repo
- For both steps, expect a transition period as we track down documentation, scripts, makefiles, etc. that point to riscv/riscv-compliance and update them (e.g. in spike arch-test/README), https://riscv.org/technical/specifications/

Draft Proposed Charter Revision

The Architectural Compatibility Test SIG is an umbrella group that will provide guidance, strategy and oversight for the development of tests used in the process of self-certifying that an implementation meets RISC-V ISA architectural compatibility requirements.

The group will:

- Guide Development of:
  - Architectural tests for RISC-V implementations covering ratified and in-flight specifications for
    - Architectural versions, standard extensions, and implementation options.
  - Tools and infrastructure to verify that an implementation is architecturally compatible
- Work with LSM and Chairs for resources to get the above work done.
- Mentor or arrange for mentoring for the resources to get the above work done

The 3 W's of testing:
Who, What, and Waivers

- Who: the last developer that makes *any* change to the RTL
  - Derived from the Branding policy
  - Configurable IP must be re-certified by whoever chooses the config (unless tested by provider)
- What: Ideally, run on physical chips, but could be SW or FPGA RTL sim
  - Implementor must provide enough resources to load and run tests and extract signature
- Waivers: HC→TSC approval required, Board/Mkting informed (mention in branding or ACT policy)
  - Non-deterministic result is legal but not modeled by formal model
  - Test fails because of demonstrable bug (in test or formal model) (test/model must be fixed)
  - Written spec has ambiguity (spec must be fixed) ; not granted if it affects SW compatiblity
- Errata:
  - cores failing tests added after certification must list errata if not fixed in silicon
  - Self-reported , not currently tested
  - Cores should re-test yearly if changes in tests, tools, or simulators

TGs under the SIG

- IF you’re creating work product, you should be a TG
- If changing requirements, ABIs, etc
    - Test plan==SOW

- The Architectural Testing Task Group(s) will define and maintain specifications for
    - test formats
    - test-benches and test frameworks needed for
        - privilege testing,
        - Concurrency/ Memory model testing
        - Asynchrnous event testing (interrupts)
        - Nondeterministic tests
    - ISA test coverage goals
    - test tools (e.g. coverage measurement, test generators)

- The Architectural Testing Task Group will maintain the appropriate GitHub:
    - tests for the individual ISA extensions
    - issues related to the tests
    - the operation and issues related to the framework

- The Architectural Testing Task Group will
    - work with the different privilege and un-privilege ISA extension Task Groups
        - to help them write test plans/specs for the ISA tests
        - to help them work with the sub-contractors (IITMadras, RIOS, CAS, etc) to deliver the tests
    - assess quality of delivered tests and be maintainer for the test GitHub

Discussion

Announcements

Next meeting March 11
Names are changing tonight (repo, etc.) tech-compliance → sig-arch-test
moving from a tech group to a SIG.

Charter revision

QC: passing arch-tests doesn't mean self-certified. don't want to imply that tests define self-certification
Chair: the charter is inward facing, not outward. The doc files in the repo say exactly what you say above.
CTO: metaphor: arch-tests are a lock on the front-door. much more needs to be done to ensure that an implementation is compatible; arch-tests: necessary but not sufficient
QC: worried about confusion
CTO: here's what i sent to J-extension re: I/D Synchronization:

"The Arch tests are not intended to verify a design or interoperability between implementations.
That is the job of the implementer. The arch tests are a prerequisite to the above.
The analogy is that the arch tests are locking the front door and not an elaborate alarm system.
That being said, we know some things are hard to test in SAIL or even SPIKE.
It is up to you all in conjunction with the arch test group to decide what you can specify that is useful and is representative and not exhaustive."

Who, What, Waivers <see slide 9 of agenda>

Chair: Who approves a waiver?
CTO: HC gives recommendation. Then sent to TSC for a vote. Need to talk to Krste about whether it is a majority vote, super-majority, unanimity. Need to talk to Calista as well. This is a marketing issue.
Chair: Where will this be documented?
CTO: Working with QC on this. Could end up in one of several places.
Chair: Waiver conditions: (see slide9, bullet 3). <bug in test, errata, discussion>
CTO: Errata: passed tests, but fail new tests. Waiver: don't pass test(s). can get waiver?
QC:Can "self waive"?? (Depends on when failure is found.)
CTO: - There are 2 cases <see slide 9, bullet 4> *AI: <CTO: and Chair to discuss offline>*
<examples of inability to run tests, e.g. being hardwired mtvec >
Imperas: how did they run code to begin with?
Inspire: ROM code could be 16k. too small.
Chair: we're testing the RTL. that is settled; RTL must provide enough resources
Chair: -f you run code before test, you might set config that screws up the test results.
Inspire: it's up to the implementer to figure out how to run the test (for example, a secure boot environment). it's up to them to figure out how to run the test. we supply the tests that must be run. implementer must figure out how to run the test.

AI: Errata needs its own bullet. <done>

Imperas email : from 2021/02/09, "moving from working group to SIG")

My understanding is that this group - the architectural test work group.. is responsible for specifications of :
- the test formats
- ISA test coverage goals
- test framework
- test tools.

and maintains the GitHub repo for:
- the tests for the individual ISA extensions
- the issues related to the tests
- the operation and issues related to the framework

where it manages and accepts pulls requests/merges/changes etc to the above.
(above added to slide 10)

Currently the working group does this work, owns the specifications and moves them forward as needed to get their approval or 'ratification' - they are very important specifications - maybe as important or more so than some ISA specs.
[and yes I am aware that we are contracting out the work of developing the actual tests to [group contributors]- that is not part of the question.]

There is also significant work needed focused on architecting privilege tests and ... framework - there has been no solid progress on that in the last 3 years – e.g. interrupts & debug mode testing? - this is all work still needing to be done.
Industry [..]. core and chip developers all state how important the tests and framework are for ensuring RISC-V doesn't fragment and have many non-compliant chips.
So : if the working group is disbanded and [now] a SIG - who is actually going to do the above work.? How are we as an industry going to make any progress with architectural validation and compliance?

Chair: The SIG can fire up TGs specific to a particular task if necessary
CTO: test plan == SOW; we don't and won't have manpower to write tests.
QC: someone here can write the testplan
CTO: I don't see that as work-product – that's the job of extension TG
Imperas :a good example is Crypto testplan. they wrote it. but need a template.
QC: look at **Bristol** testplan for Crypto as a good example
CTO: should we be a TG or a SIG?
Imperas: we're at the beginning not at the end
Chair: <to **QC**> the base ISA: you claim that test and testplan are not complete. Please explain, provide evidence. until there's evidence, we'll be a SIG not a TG

Previous Decisions & Action Items

Decisions

Modify wording of SIG charter to clarify self-certification rationale and relationship to arch tests <done>

Remove RVTEST_IO_CHECK macro from test format spec

Outstanding Action Items

NEW

Chair : document target process for removing target environment files from riscv-compliance repo into a target repo and contact all model maintainers to inform them of the process and timeline. <ongoing>

Old

Inspire: add support for QEMU target <?>

Chair: get SAIL repo moved into a riscv repo <ongoing>

QC: extract bits of FAQ as guidelines for test writing <?>
Incore: Try YAML version of SAIL to see if it works <not done>

Imperas: make pull request for updated assertion macro <removed, replaced with code from TGChair>

SH: write up coverage taxonomy

BACKUP

Non-determinism in Architectural Tests

The RV architecture defines optional and model/µarch defined behavior.
This implication: there are tests that have multiple correct answers. E.g.:

- Misaligned accesses: can be handled in HW, by "invisible" traps w/ either misaligned or illegal access causes, and do it differently for the same op accessing the same address at different times (e.g. if the 2nd half was in the TLB or not)
- Unordered Vector Reduce ops: (different results depending on ordering & cancellation)
- Tests involving concurrency will have different results depending on speculation or timing between concurrent threads (e.g. modifying page table entry without fencing)

From the point of view of ACTs, there are 2 (and sometimes more) legal answers. Usually, the golden model only generates one. Possible mechanisms to test include:

- Modify (if necessary) & configure reference model to generate either result, run it with each and accept either result from the DUT.
- Provide specific handlers for optional traps
- Use self-testing tests(compare with list or range of allowed outcomes from litmus tests)
- Avoid tests that can generate non-deterministic results

Test Report Requirements

- Architecture test version policy (mechanism TBD by ACT group)
  - Architectural test reports must include:
    - The ISA string that describes the ISA and extensions that claim to be implemented
    - The vendor and implementation IDs that the part will report
    - Test pass/fail reports
    - The YAML description of the features and options implemented (for v3 of the framework)
  - Architecture test reports must include version numbers of *
    - Toolchain,
    - reference model,
    - Architecture Compatibility Test (ACT) suite , riscv-config (for v3 of the framework)
- Each release version of ACT will document the minimum version of the toolchain utilities required to support the instructions used for that version of the tests in a repository README file
- *Remaining questions:*
  - *Where does this report get sent?   Do they publicly/internally announce who has passed?*
  - *Who reviews the report? (who is the signing authority?)(who can approve waivers?)*

* *Need to ensure these get updated automatically*

Architectural Test Rationale – Intent and Limits

RISC-V Architectural Tests 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 a prerequisite to licensing the RISC-V trademarks in connection with the design.

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.

To be added to the riscv/riscv-compliance/doc/ directory as “RISC-V Architectural Test Rationale” satisfying Jira CSC-2

Test Acceptance Criteria – second cut

Tests must:

- conform to current standard of test spec (macros, labels, directory structure)
- use only files that are part of the defined support files in the repository
- run in framework
- run in SAIL and not fail any tests
- generate a valid signature using SAIL (that can be saved & compared with another DUT/sim)
- Report that test results propagate to signature
- has a clear configuration - i.e. which ISA extension it can be used with
- improves coverage
- use only standard instructions (fixed size per architecture LI, LA allowed)
- must be commented in test_case header

Framework Requirements – first cut

The framework must:

- Use the TestFormat spec and macros described therein
  - (which must work - including assertions)
- Choose test cases according to equations that reference the YAML configuration
- Define macro variables that can be used inside tests based on the YAML configuration
- Include the compliance trap handler, & handle its (separate) signature area
- Load, initialize, and run selected tests between two selected models, extract the signatures, compare results, and write out a report file
- Exist in a riscv github repo, with a more than one maintainer.
- Be easy to get running, e.g.:
  - run under a variety of OSes with the minimum number of distro specific tools.
  - Not require sudo privileges
- Have the ability to measure and report coverage
  - Coverage specification is a separate file
  - Could be a separate app

Pull/Issue Status

Issue#	Date	submitter	title	status	comments
#22	24-Nov-18	brouhaha	I-MISALIGN_LDST-01 assumes misaligned data access will trap	^	HW misalign support not configurable
#40	4-Feb-19	debs-sifive	Usage of tohost/fromhost should be removed	|	now
#90	11-Feb-20	towoe	Report target execution error	|	
#106	22-Apr-20	jeremybennett	Use of pseudo instructions in compliance tests	fixed in RFQ tests	Will be closed in 2.1 or 2.2
#142	17-Nov-20	subhajit26	Not able to run compliance test for rv32E device and RV32E ISA	RV32E only	Not RV32EC or RV32EM
#145-9	01-Dec-20	Imperas	Test I EBREAK,ECALL, MISALIGN_JMP/LDST, OpenHW	v	HW misalign support not configurable
#107	22-Apr-20	jeremybennett	Clang/LLVM doesn't support all CSRs used in compliance test suite	under discussion	-can we add an alias?
#109	06-May-20	Olofk	Swerv fails because parallel make	under discussion	May be fixed?
#115	06-jun-20	adchd	How to support on-board execution?	under discussion	
#157	15-Dec-20	stnolting	Memory requirement for new test framework	Unfixable?	
pull#128	29-jul-20	nmeum	grift: update for new directory structure	Correction made	Review by Marc, needs corectiono
pull#129	31-jul-20	nmeum	sail-riscv-ocaml: Disable RVC extension on all devices not using it	In process	Who can review this?
pull#163	11-jan-21	snolting	Makefile improvements	In process	Needs review
#45	12-Feb-19	debs-sifive	Reorganization of test suites for code maintainability	deferred	fixed in v2
#63	13-Aug-19	jeremybennett	Global linker script is not appropriate	fixed	Needs target provided linker scripts
#72	26-Oct-19	vogelpi	Allow for non-word aligned \`mtvec\`	deferred	needs v.2
#78	26-Jan-20	bobbl	RV_COMPLIANCE_HALT must contain SWSIG	Fixed	
#105	22-Apr-20	jeremybennett	Non-standard assembler usage	under discussion	Simple fix
#108	22-Apr-20	bluewww	RI5CY's \`compliance_io.h\` fails to compile with clang	Pull #152 fixes it	close after merge
#116	06-jun-20	simon5656	loss of 64bit test infrastucture	under discussion	Will be fixed by RFQ tests
#119	17-jun-20	allenjbaum	Missing RV32i/RV64i test: Fence	Test has been written	Close when RFQ test is merged
#132	15-aug-20	davidmlw	Why not just use mepc for mret?	Answered - close	Should be resolved
#135	04-sep-20	MikeOpenHWGroup	Request for a Tag on this Repo	assigned	Req. for non-hash tag; needs process
#155	03-Dec-20	bluewww	RI5CY: add minimum clang version#	Fixes issue #108	Merge after review
#156	08-Dec-20	panda1628	PMP/PMA Tests	Question answered	Can be closed
#157	15-dec-20	Stnolting	Memory requirement for new test framework	Question answered	Can be closed
#158/164	23-dec-20	Stnolting	Add white space in verify report [absolutely uncritical]	Non-critical	Should be accepted (Pull #164)
#165	12-jan-21	Towoe	Version numbering	Non-critical	Close?, will use semantic vers from now on
#169	22-jan-21	Towoe	RISCOF redefine of TEST_CASE_1	Question answered	Can be closed

JIRA Status

Issue#	Date	submitter	title	status	comments
IT-1	27Aug/20	Allen Baum	Need to modify the description of compliance in https://riscv.org/technical/specifications/	done	
IT-4	01/Sep/20	Allen Baum	Add Jira link to TG home pages	In prog	
CSC-1	20/Aug/20	Ken Dockser	Come up with names for the tests suites that we are creating		1st step done
CSC-2	20/Aug/20	Ken Dockser	Produce concise text to explain the Architecture Tests intent and Limits		Written, needs pull req
CSC-3	20/Aug/20	Ken Dockser	Come up with an internal goal for what we wish to accomplish with the Architectural Tests		Not written
CSC-4	20/Aug/20	Ken Dockser	Develop a roadmap for all the different categories of test suites that will need to be created		Not written
CSC-5	20/Aug/20	Ken Dockser	Develop a roadmap for releases of single-instruction Architecture Tests		Not written
CSC-6	20/Aug/20	Ken Dockser	Develop a reference RTL test fixture that can stimulate and check the CPU under test		Needs more discussion