# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Luke Kenneth Casson Leighton

## Your email address (optional)

lkcl@lkcl.net

## Your organisation *

Independent Software and Hardware Libre Developer

## Your role *

Full transparent, auditable, independent and accountable processor development

## The group you are speaking for, if any

n/a

---

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ◉ | ○ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ◉ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ◉ | ○ | ○ |
| Floating point | ◉ | ○ | ○ | ○ |
| Use as an emulator | ○ | ◉ | ○ | ○ |
| Use as a test oracle in tandem verification | ○ | ◉ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ◉ | ○ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ◉ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ◉ | ○ | ○ | ○ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ○ | ◉ |

## Forvis - comments

the answer is the same for all of the formal models: it is too early to make a decision. each of the models is extremely good: it's just that they're (all of them) incomplete (still under development in some way). in addition, i think you'll find that even *making* a choice will result in that team becoming a critical dependency *for the entire RISC-V ecosystem*. if they're an academic team, that's unfortunate: once the project no longer receives funding or the research project ends, then so does RISC-V "conformance". and if they're a Corporation, the Corporation may manipulate the RISC-V ecosystem for profit-maximising purposes, and if it goes bust, the project ends, and so does RISC-V "conformance". so not only is it a bad idea to pick one *right now*, it's a bad idea to pick only *ONE* of these formal verification suites *at all*. instead it would be far, far better for the RISC-V Formal Verification Group to develop a *STANDARD* for Formal Verification, to which **ALL** of these may comply. that's what a Standards Organisation does: develop *STANDARDS*, *NOT* select some random codebase off the internet and say "here! this is now a standard!". so you need to define the *expected results*, in sufficient detail and with sufficient clarity such that *ALL* of the FIVE formal models may conform and comply with it, in a machine-executable fashion. if that's too challenging, then at least some human-verifiable expectations may be defined.

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

the answer is the same for all of the formal models: it is too early to make a decision. each of the models is extremely good: it's just that they're (all of them) incomplete (still under development in some way). in addition, i think you'll find that even *making* a choice will result in that team becoming a critical dependency *for the entire RISC-V ecosystem*. if they're an academic team, that's unfortunate: once the project no longer receives funding or the research project ends, then so does RISC-V "conformance". and if they're a Corporation, the Corporation may manipulate the RISC-V ecosystem for profit-maximising purposes, and if it goes bust, the project ends, and so does RISC-V "conformance". so not only is it a bad idea to pick one *right now*, it's a bad idea to pick only *ONE* of these formal verification suites *at all*. instead it would be far, far better for the RISC-V Formal Verification Group to develop a *STANDARD* for Formal Verification, to which **ALL** of these may comply. that's what a Standards Organisation does: develop *STANDARDS*, *NOT* select some random codebase off the internet and say "here! this is now a standard!". so you need to define the *expected results*, in sufficient detail and with sufficient clarity such that *ALL* of the FIVE formal models may conform and comply with it, in a machine-executable fashion. if that's too challenging, then at least some human-verifiable expectations may be defined.

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ○ | ○ | ◉ |

## Sail - comments

the answer is the same for all of the formal models: it is too early to make a decision. each of the models is extremely good: it's just that they're (all of them) incomplete (still under development in some way). in addition, i think you'll find that even *making* a choice will result in that team becoming a critical dependency *for the entire RISC-V ecosystem*. if they're an academic team, that's unfortunate: once the project no longer receives funding or the research project ends, then so does RISC-V "conformance". and if they're a Corporation, the Corporation may manipulate the RISC-V ecosystem for profit-maximising purposes, and if it goes bust, the project ends, and so does RISC-V "conformance". so not only is it a bad idea to pick one *right now*, it's a bad idea to pick only *ONE* of these formal verification suites *at all*. instead it would be far, far better for the RISC-V Formal Verification Group to develop a *STANDARD* for Formal Verification, to which **ALL** of these may comply. that's what a Standards Organisation does: develop *STANDARDS*, *NOT* select some random codebase off the internet and say "here! this is now a standard!". so you need to define the *expected results*, in sufficient detail and with sufficient clarity such that *ALL* of the FIVE formal models may conform and comply with it, in a machine-executable fashion. if that's too challenging, then at least some human-verifiable expectations may be defined.

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ○ | ◉ |

## RISC-V-PLV - comments

the answer is the same for all of the formal models: it is too early to make a decision. each of the models is extremely good: it's just that they're (all of them) incomplete (still under development in some way). in addition, i think you'll find that even *making* a choice will result in that team becoming a critical dependency *for the entire RISC-V ecosystem*. if they're an academic team, that's unfortunate: once the project no longer receives funding or the research project ends, then so does RISC-V "conformance". and if they're a Corporation, the Corporation may manipulate the RISC-V ecosystem for profit-maximising purposes, and if it goes bust, the project ends, and so does RISC-V "conformance". so not only is it a bad idea to pick one *right now*, it's a bad idea to pick only *ONE* of these formal verification suites *at all*. instead it would be far, far better for the RISC-V Formal Verification Group to develop a *STANDARD* for Formal Verification, to which **ALL** of these may comply. that's what a Standards Organisation does: develop *STANDARDS*, *NOT* select some random codebase off the internet and say "here! this is now a standard!". so you need to define the *expected results*, in sufficient detail and with sufficient clarity such that *ALL* of the FIVE formal models may conform and comply with it, in a machine-executable fashion. if that's too challenging, then at least some human-verifiable expectations may be defined.

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

the answer is the same for all of the formal models: it is too early to make a decision. each of the models is extremely good: it's just that they're (all of them) incomplete (still under development in some way). in addition, i think you'll find that even *making* a choice will result in that team becoming a critical dependency *for the entire RISC-V ecosystem*. if they're an academic team, that's unfortunate: once the project no longer receives funding or the research project ends, then so does RISC-V "conformance". and if they're a Corporation, the Corporation may manipulate the RISC-V ecosystem for profit-maximising purposes, and if it goes bust, the project ends, and so does RISC-V "conformance". so not only is it a bad idea to pick one *right now*, it's a bad idea to pick only *ONE* of these formal verification suites *at all*. instead it would be far, far better for the RISC-V Formal Verification Group to develop a *STANDARD* for Formal Verification, to which **ALL** of these may comply. that's what a Standards Organisation does: develop *STANDARDS*, *NOT* select some random codebase off the internet and say "here! this is now a standard!". so you need to define the *expected results*, in sufficient detail and with sufficient clarity such that *ALL* of the FIVE formal models may conform and comply with it, in a machine-executable fashion. if that's too challenging, then at least some human-verifiable expectations may be defined.

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Josh Scheid

## Your email address (optional)

jscheid@ventanamicro.com

## Your organisation *

Ventana Micro Systems

## Your role *

Engineer

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ○ | ◉ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ◉ | ○ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ◉ | ○ | ○ | ○ |
| Floating point | ○ | ◉ | ○ | ○ |
| Use as an emulator | ○ | ○ | ◉ | ○ |
| Use as a test oracle in tandem verification | ○ | ◉ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ○ | ○ | ◉ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ◉ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ○ | ◉ | ○ | ○ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ○ | ◉ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

The intent behind the content of the prose specification versus the formal specification should be explicit. This will help guide the content so that the two appear as complementary instead of redundant.

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ○ | ◉ |

## Forvis - comments

Concurrency.

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

Privilege levels and concurrency.

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ○ | ◉ | ○ |

## Sail - comments

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ○ | ◉ |

## RISC-V-PLV - comments

Concurrency.

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

Privilege levels and concurrency.

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Chuanhua Chang

## Your email address (optional)

chchang@andestech.com

## Your organisation *

Andes Technology

## Your role *

Senior director of RD/Architecture

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ○ | ◉ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ◉ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ○ | ○ | ◉ |
| Floating point | ○ | ◉ | ○ | ○ |
| Use as an emulator | ◉ | ○ | ○ | ○ |
| Use as a test oracle in tandem verification | ◉ | ○ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ○ | ◉ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ◉ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ○ | ◉ | ○ | ○ |
| Use in test generation | ○ | ◉ | ○ | ○ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ○ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ◉ | ○ |

## Forvis - comments

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ◉ | ○ |

## GRIFT - comments

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

More features, good readability, can generate C

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Frédéric Pétrot

## Your email address (optional)

frederic.petrot@univ-grenoble-alpes.fr

## Your organisation *

Grenoble-INP TIMA Laboratory

## Your role *

Professor

## The group you are speaking for, if any

None

---

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ◉ | ○ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ◉ | ○ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ◉ | ○ | ○ | ○ |
| Floating point | ○ | ○ | ○ | ◉ |
| Use as an emulator | ◉ | ○ | ○ | ○ |
| Use as a test oracle in tandem verification | ◉ | ○ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ○ | ◉ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ◉ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ○ | ◉ | ○ | ○ |
| Use in test generation | ○ | ○ | ○ | ◉ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ○ | ◉ | ○ | ○ |
| Plans for long-term development and maintenance | ○ | ○ | ○ | ◉ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ○ | ◉ |

## Forvis - comments

Supports neither instruction encodings and asm syntax nor conccurency

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

Seems to support the least features among the tools you propose

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

The tool covers encodings and concurrency, generation for theorem provers, and is the fastest one in simulation. Futhermore it has a BSD licence

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ○ | ◉ |

## RISC-V-PLV - comments

Does not cover most of the things I feel interesting (encoding) or very hard to have right (concurrency).

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ◉ | ○ |

## Kami - comments

Covers half my needs !

## Any additional comments

Note that I am not an expert in formal stuff, and that my analysis comes from your comparison chart, not from my own experience.

---

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Tariq Kurd

## Your email address (optional)

tariq.kurd@huawei.com

## Your organisation *

Huawei

## Your role *

CPU Architect

## The group you are speaking for, if any

Huawei RISC-V development

---

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ◉ | ○ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ○ | ○ | ◉ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ◉ | ○ | ○ | ○ |
| Floating point | ○ | ◉ | ○ | ○ |
| Use as an emulator | ◉ | ○ | ○ | ○ |
| Use as a test oracle in tandem verification | ○ | ◉ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ○ | ○ | ○ | ◉ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ◉ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ◉ | ○ | ○ | ○ |
| Use for hardware synthesis | ○ | ◉ | ○ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ○ | ◉ |

## Forvis - comments

too slow

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

too slow

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

Faster, specification language is better

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ○ | ◉ |

## RISC-V-PLV - comments

too slow

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

unknown speed, no privilege level support?

## Any additional comments

Still need SAIL to support F-extension, otherwise it's the best

---

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

andrew dellow

## Your email address (optional)

## Your organisation *

hisilicon

## Your role *

chief security architect

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ◉ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ○ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ◉ | ○ | ○ | ○ |
| Floating point | ○ | ○ | ◉ | ○ |
| Use as an emulator | ○ | ◉ | ○ | ○ |
| Use as a test oracle in tandem verification | ◉ | ○ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ○ | ◉ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ◉ | ○ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ○ | ◉ | ○ | ○ |
| Use for hardware synthesis | ◉ | ○ | ○ | ○ |
| Licencing, tool ecosystem, dependencies | ○ | ◉ | ○ | ○ |
| Ease of extensibility | ○ | ◉ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ◉ | ○ |

## Forvis - comments

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Håkan Thörngren

## Your email address (optional)

hth313@gmail.com

## Your organisation *

Coming startup

## Your role *

Principal developer

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ○ | ◉ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ◉ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ○ | ○ | ◉ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ○ | ◉ | ○ |
| Floating point | ○ | ◉ | ○ | ○ |
| Use as an emulator | ○ | ○ | ○ | ◉ |
| Use as a test oracle in tandem verification | ○ | ○ | ○ | ◉ |
| Generation of theorem-prover definitions for proof | ○ | ○ | ○ | ◉ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ○ | ○ | ◉ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ○ | ◉ | ○ | ○ |
| Use in test generation | ◉ | ○ | ○ | ○ |
| Use for hardware synthesis | ○ | ○ | ○ | ◉ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ◉ | ○ | ○ |

## Forvis - comments

Haskell is a strongly desired by me and Forvis has is a permissive license. It seems quite well executed. I can definitely see this one would be useful to me and it should serve well as a formal model.

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

GPL, totally unusable for partial inclusion in commercial products. While it is a formal specification, I can definitely see that there are chance for an executable specification to be at least partially included in actual products, test or development and there is a wide grey zone. People may say that a license will say what is allowed or not, but many commercial users will be very careful with this, and there are alternatives without this problem. I will for sure stay at a safe distance from this one.

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ○ | ◉ | ○ |

## Sail - comments

I put it on Adequate, I have somewhat limited use of it as I am focused on Haskell and will have most use for such specification, but I am not to dismiss it.

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

Haskell, permissive license, but it is an academic project and to me such projects are often orphaned as soon as the academic interest shifts, as it often does. Lots of good stuff come out of academia, but it is often not entirely complete or long-term.

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ◉ | ○ |

## Kami - comments

Another non-Haskell specification, see comments about Sail.

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Nathan Studer

## Your email address (optional)

Nathan.Studer@DornerWorks.com

## Your organisation *

DornerWorks

## Your role *

Senior Engineer

The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ○ | ◉ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ◉ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ◉ | ○ | ○ |
| Floating point | ○ | ○ | ◉ | ○ |
| Use as an emulator | ◉ | ○ | ○ | ○ |
| Use as a test oracle in tandem verification | ○ | ○ | ○ | ◉ |
| Generation of theorem-prover definitions for proof | ○ | ○ | ○ | ◉ |
| Use for lightweight formal verification (bounded model-checking etc.) | ○ | ○ | ○ | ◉ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ○ | ○ | ○ | ◉ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ◉ | ○ | ○ |

## Forvis - comments

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ◉ | ○ |

## Kami - comments

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Jesse Millwood

## Your email address (optional)

jesse.millwood@dornerworks.com

## Your organisation *

DornerWorks

## Your role *

Embedded Software Engineer

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ○ | ◉ | ○ | ○ |
| Assembly syntax and encoding specification | ◉ | ○ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ◉ | ○ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ◉ | ○ | ○ |
| Floating point | ○ | ○ | ○ | ◉ |
| Use as an emulator | ○ | ○ | ◉ | ○ |
| Use as a test oracle in tandem verification | ○ | ○ | ○ | ◉ |
| Generation of theorem-prover definitions for proof | ◉ | ○ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ◉ | ○ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ○ | ○ | ◉ | ○ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ◉ | ○ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ◉ | ○ |

## Forvis - comments

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ◉ | ○ |

## GRIFT - comments

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ◉ | ○ | ○ |

## Sail - comments

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ◉ | ○ |

## Kami - comments

## Any additional comments

# RISC-V Formal ISA Specification Public Review: Survey

For this public review, please first read the README and the descriptions of the current status of the various models at https://github.com/riscv/ISA_Formal_Spec_Public_Review, and the comparison table at https://github.com/riscv/ISA_Formal_Spec_Public_Review/blob/master/comparison_table.md, then fill in the form below with your opinions and rationale for them. After submission, you will be able to see the other responses so far, and to edit your response.

This survey is organised by the RISC-V Formal ISA Specification Task Group and the University of Cambridge (contacts: Nikhil, Task Group Chair <nikhil@bluespec.com> and Peter Sewell <Peter.Sewell@cl.cam.ac.uk>). The intention is to publish all responses to this survey, attributed to the names and contact details provided. In submitting a response you consent to this occurring; if you do not wish this to happen please do not complete the survey. For more details on how we will handle your personal information, please see https://www.information-compliance.admin.cam.ac.uk/data-protection/general-data.

## Your name *

Andrew Tolmach

## Your email address (optional)

tolmach@pdx.edu

## Your organisation *

Portland State University

## Your role *

Professor

## The group you are speaking for, if any

## How important is each aspect of a formal ISA specification for RISC-V?

| | Very important | Important | Not very important | No opinion |
|---|---|---|---|---|
| Functional coverage of ISA | ◉ | ○ | ○ | ○ |
| Assembly syntax and encoding specification | ○ | ◉ | ○ | ○ |
| Multicore concurrency (RVWMO+ZTSO) | ○ | ○ | ◉ | ○ |
| Single-core concurrency (instruction cache behaviour, interrupts, etc.) | ○ | ◉ | ○ | ○ |
| Floating point | ○ | ○ | ○ | ◉ |
| Use as an emulator | ◉ | ○ | ○ | ○ |
| Use as a test oracle in tandem verification | ○ | ◉ | ○ | ○ |
| Generation of theorem-prover definitions for proof | ◉ | ○ | ○ | ○ |
| Use for lightweight formal verification (bounded model-checking etc.) | ◉ | ○ | ○ | ○ |

| | | | | |
|---|---|---|---|---|
| Use in documentation, and readability | ◉ | ○ | ○ | ○ |
| Use in test generation | ○ | ◉ | ○ | ○ |
| Use for hardware synthesis | ○ | ○ | ◉ | ○ |
| Licencing, tool ecosystem, dependencies | ○ | ◉ | ○ | ○ |
| Ease of extensibility | ◉ | ○ | ○ | ○ |
| Plans for long-term development and maintenance | ◉ | ○ | ○ | ○ |

## Comments on any of the above

Obviously, each of these may be crucial to somebody: I've answered what is important to me.

---

## The Candidate Formal Models

For each of the candidate formal models, please give your overall view for whether it would be good, adequate, or inadequate for the needs of the RISC-V ecosystem, explaining why.

## Forvis (Bluespec)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Forvis spec: | ○ | ◉ | ○ |

## Forvis - comments

Uses Haskell to good effect without (hopefully) scaring off readers more used to conventional ISA descriptions. But I'm not sure how easy it will be to extract definitions suitable for use in a theorem prover.

## GRIFT (Galois)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the GRIFT spec: | ○ | ○ | ◉ |

## GRIFT - comments

Only accessible to Haskell experts.

## Sail (SRI/Cambridge)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Sail spec: | ○ | ◉ | ○ |

## Sail - comments

Probably the best choice at full scale; language and definition style seem a little heavy-weight for the minimal subsets of the ISA. Current extraction to Coq does not produce very idiomatic definitions, but this can no doubt be improved over time.

## RISC-V-PLV (MIT)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the RISC-V-PLV spec: | ○ | ◉ | ○ |

## RISC-V-PLV - comments

Similar to Forvis, but requires somewhat more Haskell expertise to read (OK for me, but probably not ideal for broader community).

## Kami (SiFive)

| | Good | Adequate | Inadequate |
|---|---|---|---|
| Overall, is the Kami spec: | ○ | ○ | ◉ |

## Kami - comments

Organization focused on synthesis is not so natural for other purposes. Coq is not very accessible to broader community.

## Any additional comments