# RISC-V Lifecycle Guide

Version 1.10

# Table of Contents

# 1. Introduction

This document has been created to facilitate RISC-V member participation in the key activities involved in creating and running groups, writing of specifications, and contributing open-source software in support of RISC-V architectures. It is a guide, not the rules. The “rules” exist in the supporting RISC-V Policy documents (see the Reference Information section). **Thus, any discrepancy between this guide and the policies, means the guide is out-of-date.**

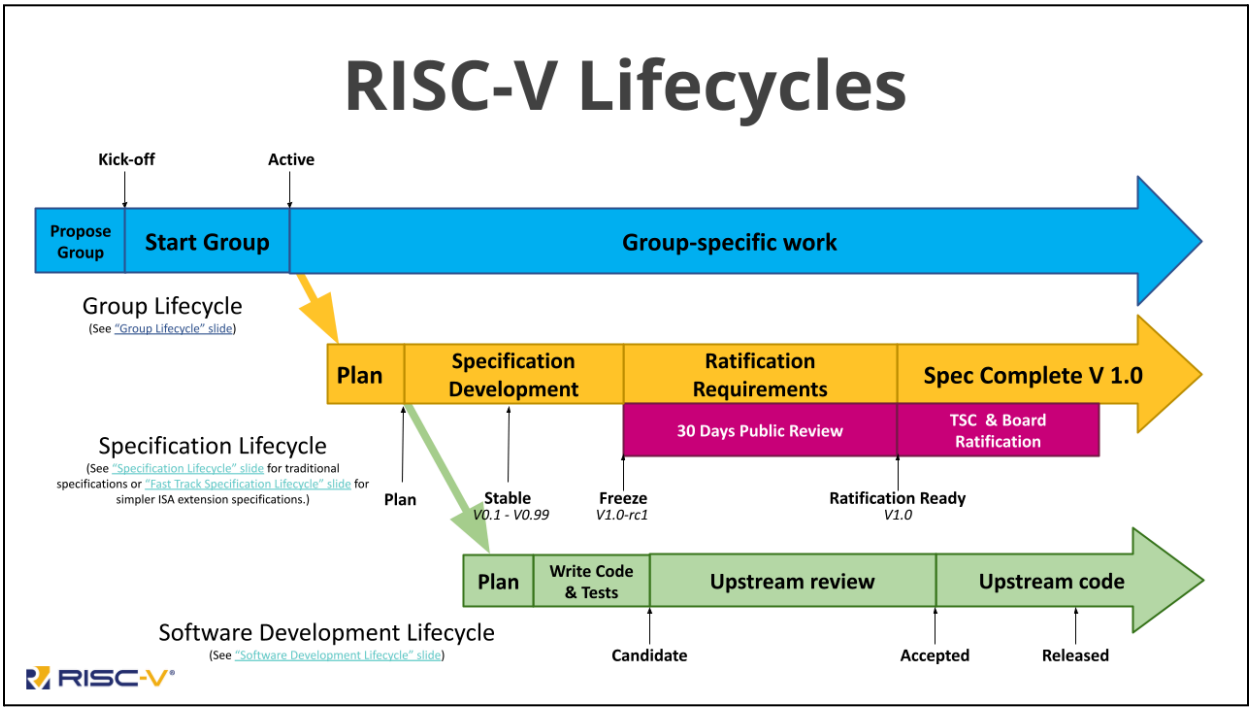
If something is unclear or if you have suggestions on how to improve the information in this document, submit your comments against the source document.

## Overview

RISC-V has 3 main lifecycles:

- Group Lifecycle
- Specification Lifecycle
- Software Development Lifecycle

The interactions of these 3 lifecycles can be generally depicted as:



RISC-V lifecycles include unique milestone names such as “Kick-off”, “Plan”, or “Reviewed”. Each milestone contains a set of steps to achieve it and often require one or more approvals (formal or informal) to consider the milestone complete. These steps and approvals are detailed in the lifecycle subsections of the document. For consistency of discussion, “status” is generally indicated by referencing the last completed (approved) milestone and the current set of activities, e.g. the “Pickle Slicer” specification has achieved the “Freeze Milestone” and is currently awaiting “Committee Chair Sign-off for Ratification”.

Groups provide the foundation for all activities in RISC-V. Groups get proposed, started, and then generally perform one or more tasks. The Group Lifecycle section of this document provides greater details about RISC-V groups, including the steps necessary to create a new group.

A common work activity for a RISC-V group is the development of a specification, either tied to the RISC-V Instruction Set Architecture (ISA) or in some piece of the software ecosystem. RISC-V specification development is generally performed by Task Groups which perform these activities in these general steps: creating a plan, writing the specification, and then completing requirements to achieve ratification. More details can be found in the Specification Lifecycle section.

A specialized process, known as the Fast Track Extension (or simply Fast Track) lifecycle also exists for smaller, simpler documents. In this process, the work is driven from an ISA Committee (IC) but largely follows the same development steps. More details can be found in the Fast Track Extension Specification Lifecycle section.

RISC-V has created a Software Lifecycle to facilitate development of open-source software for RISC-V members and ensure that software impacts are consistently addressed. This lifecycle includes some up-front work during the specification planning phase, but really begins active work after a specification becomes stable enough to trust for software development. Code and tests are generally written, reviewed within RISC-V communities, and then sent up-stream

for inclusion in the proper mainline communities. The details of these steps are further described in the Software Lifecycle section.

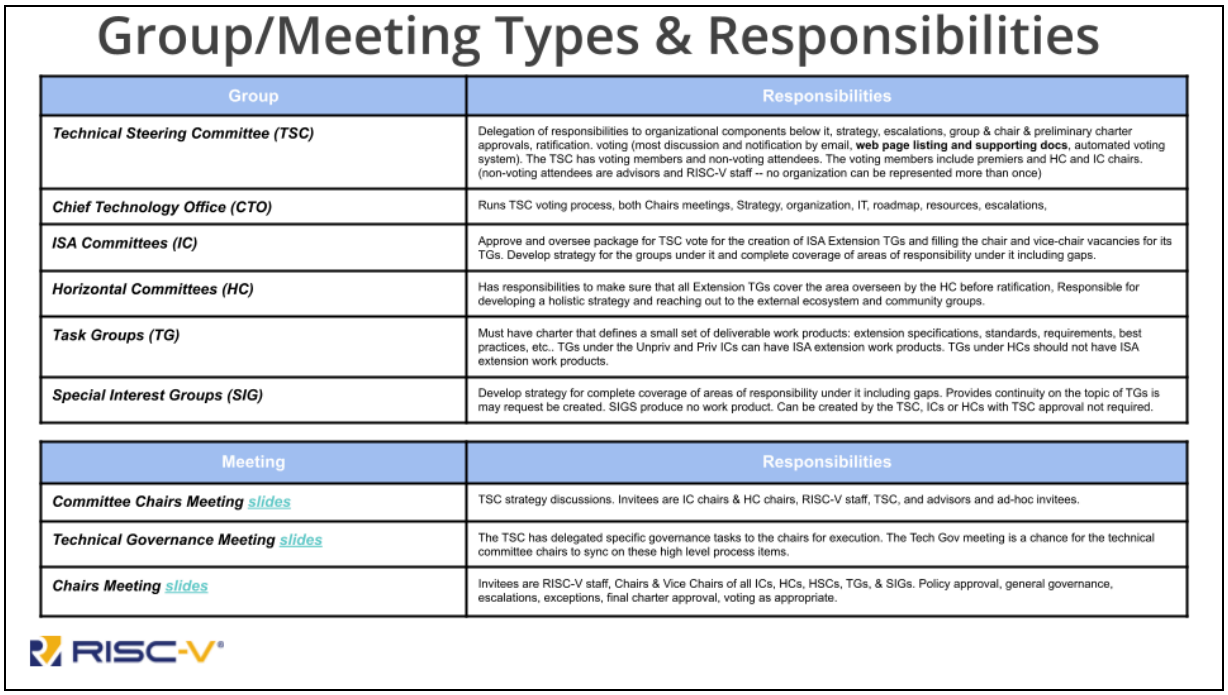
## Reference Information

The following documents provide deeper context or explanation for the information in this document:

| Document | Type | Description |
|---|---|---|
| Groups & Chairs | Policy | Process requirements document for group formation |
| Ratification Policy | Policy | Process requirements document for specification development |
| Acceptance Criteria Policy | Policy | Process requirements document for each milestone in the specification development lifecycle. |
| Chairs Best Practices | Policy | General guidance document for Chairs and Vice-chairs |
| GitHub Repo Structure & Administration | Policy | Requirements document about the handling of GitHub repositories |
| RISC-V Fast Track Extension Process | Policy | Process requirements for ratifying simple specifications. |
| RISC-V Lifecycles and Milestones Overview | Presentation | Overview presentation of main RISC-V lifecycles |
| RISC-V Technical Organization | Presentation | Overview of the RISC-V organizational structure |
| Specification Status | Wiki page | Dashboard of all active specifications |

# 2. Group Lifecycle

RISC-V has 5 types of work groups as shown below in the RISC-V Technical Organization presentation.



## Group/Meeting Types & Responsibilities

| Group | Responsibilities |
|---|---|
| ***Technical Steering Committee (TSC)*** | Delegation of responsibilities to organizational components below it, strategy, escalations, group & chair & preliminary charter approvals, ratification. voting (most discussion and notification by email, **web page listing and supporting docs**, automated voting system). The TSC has voting members and non-voting attendees. The voting members include premiers and HC and IC chairs. (non-voting attendees are advisors and RISC-V staff -- no organization can be represented more than once) |
| ***Chief Technology Office (CTO)*** | Runs TSC voting process, both Chairs meetings, Strategy, organization, IT, roadmap, resources, escalations, |
| ***ISA Committees (IC)*** | Approve and oversee package for TSC vote for the creation of ISA Extension TGs and filling the chair and vice-chair vacancies for its TGs. Develop strategy for the groups under it and complete coverage of areas of responsibility under it including gaps. |
| ***Horizontal Committees (HC)*** | Has responsibilities to make sure that all Extension TGs cover the area overseen by the HC before ratification, Responsible for developing a holistic strategy and reaching out to the external ecosystem and community groups. |
| ***Task Groups (TG)*** | Must have charter that defines a small set of deliverable work products: extension specifications, standards, requirements, best practices, etc., TGs under the Unpriv and Priv ICs can have ISA extension work products. TGs under HCs should not have ISA extension work products. |
| ***Special Interest Groups (SIG)*** | Develop strategy for complete coverage of areas of responsibility under it including gaps. Provides continuity on the topic of TGs is may request be created. SIGS produce no work product. Can be created by the TSC, ICs or HCs with TSC approval not required. |

| Meeting | Responsibilities |
|---|---|
| ***Committee Chairs Meeting*** *slides* | TSC strategy discussions. Invitees are IC chairs & HC chairs, RISC-V staff, TSC, and advisors and ad-hoc invitees. |
| ***Technical Governance Meeting*** *slides* | The TSC has delegated specific governance tasks to the chairs for execution. The Tech Gov meeting is a chance for the technical committee chairs to sync on these high level process items. |
| ***Chairs Meeting*** *slides* | Invitees are RISC-V staff, Chairs & Vice Chairs of all ICs, HCs, HSCs, TGs, & SIGs. Policy approval, general governance, escalations, exceptions, final charter approval, voting as appropriate. |

RISC-V

They are:

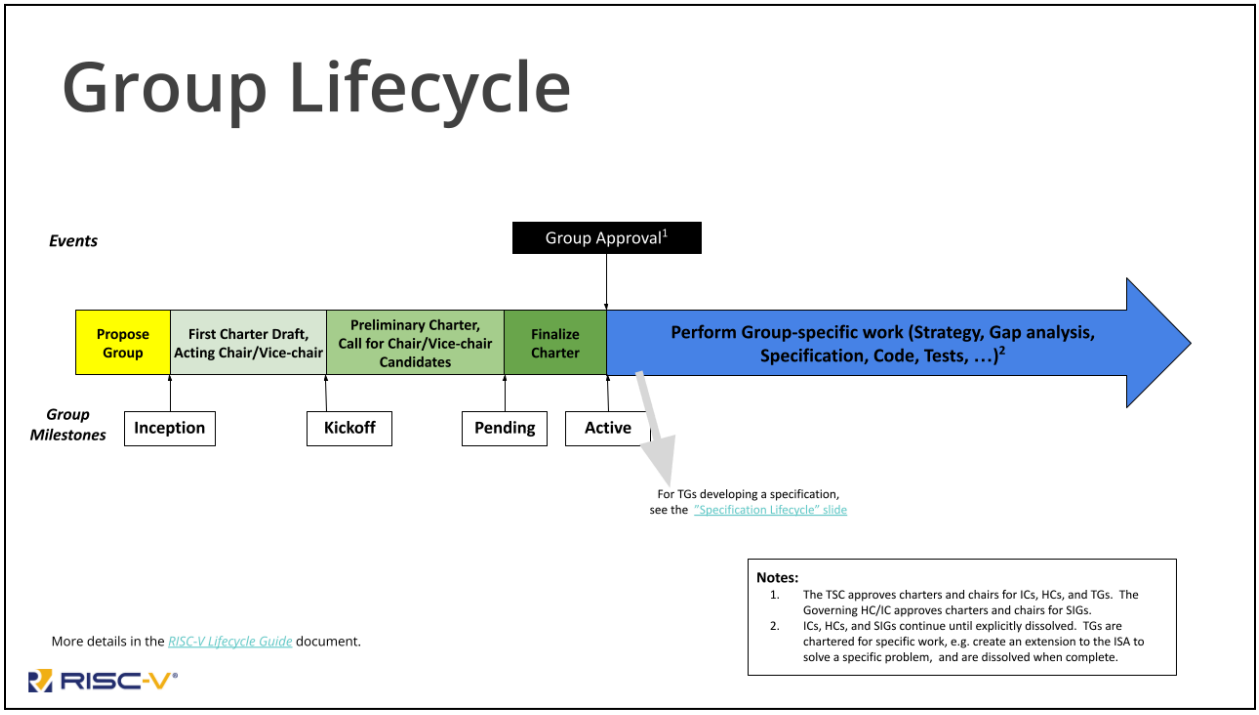
- Technical Steering Committee (TSC)
- Instruction Set Architecture (ISA) Committees (ICs)
- Horizontal Committees (HCs)
- Task Groups (TGs)
- Special Interest Groups (SIGs)

Within the context of specification development, these groups interact as follows:

1. ICs and HCs and serve as the Governing Committees which sponsor TGs to create specifications to define ISA extensions or software interface.
2. SIGs own the strategy for HCs and identify the gaps or needs which map into specification creation or updates.
3. TGs are created under an IC or HC and may receive additional technical guidance from other ICs or HCs (Dotted-line Committees). For example, a crypto ISA extension would be sponsored by the Unprivileged ICs while being dotted line to the Security HC

4. The TSC provides oversight to RISC-V technical execution, including but not limited to approval of the TG, approval of the specification readiness for ratification, and general oversight and prioritization of the technical activities.

The RISC-V Group Lifecycle consists of 4 milestones – Inception, Kick-off, Pending, Active – which fit together as shown.




The following sections detail each phase of the lifecycle and serve as the step-by-step guide for how to start a new group in RISC-V. They are intended to be worked in the order they are presented.

Additional details about the groups can be found in the RISC-V Groups & Elections Policy.

## Group Inception Milestone

This milestone is the first and most simple one to achieve: it simply requires starting discussion about a new group and then informing the RISC-V Staff of your intentions so that they can help coach you through the process.

Once complete, continue with the Group Kick-off Milestone.

### Background

All groups start with a discussion about the needs for a new group. A discussion can begin in multiple places such as on an public mailing list like isa-dev or sw-dev, on an internal group mailing list like tech-privileged, or in an email to one or more community leaders. While there's no predetermined place to get started, proposing a new group on an ISA Committee (IC) or Horizontal Committee (HC) mailing list ensures that the proposal reaches a broad group of RISC-V leadership who can provide guidance on the new group.

If you are unsure of where to propose the group, reach out to the RISC-V Staff for guidance.

### Steps to Complete the Inception Milestone

1. Begin discussions.
2. Notify the RISC-V Staff by creating a new "New Group Request" issue at help.riscv.org with a proposed group name, brief description, and any additional information you have.

### Work Product of the Inception Milestone

- Start of discussions

### Approvals for the Inception Milestone

- No approvals are needed. This milestone is achieved simply by starting discussions about the new group.

## Group Kick-off Milestone

This second milestone represents the base work needed – deciding how the new group should be situated in the RISC-V organization and what needs it will generally address. The work is generally driven within one or two HCs/ICs.

Once complete, continue with the Group Pending Milestone.

### Background

This first step in a new group focuses on discussions hosted by an HC or IC to identify the goals and rationale for the effort. Areas of discussions can be, but not limited to:

- Is the goal to create an SIG, ISA, or joint working group?
- In addition to the hosted HC, what other HCs' need to be involved or informed? (ie. Security, SoC Infra)

Best practices for these discussions is to include them as part of the existing HC/IC meetings or to include them on a mailing list. This technique documents all inputs and enables broad participation.

## Steps to Complete the Kick-off Milestone

1. Once a need for a TG has been generally identified, deeper discussions must occur. The first decision to be made is that an HC or IC needs to "sponsor" the exploration by hosting the discussion. This group is typically the Governing Committee. If the TG involves work on the ISA, identify which IC should be the Governing Committee. Explore whether any other HC should be included as a Dotted-line Committee.
2. Discussion about the TG should include, but not limited to:
    - Is the goal to create an SIG, ISA, or joint working group?
    - In addition to the hosted HC, what other HCs' need to be involved or informed? (ie. Security, SoC Infra)
3. Draft a proposed charter that defines the new group scope and deliverables. A template has been provided in the template-group-admin/Charter.md file which will be primed into your new community repository in the next milestone. A Google Doc-friendly version can also be found in the Charter template section in the Appendix. To see other group charters, view the Charter.md files in other group repos of the riscv-admin GitHub organization.
4. Review the draft charter within the context of the Governing and Dotted-line Committees. This may be done via mailing list, slide presentations, or within meeting discussions.
5. Obtain Governing Committee approval (informal) for the draft charter.
6. Governing Committee Chair and/or Vice-chair identify committee Acting Chair and (if needed) Vice Chair.

## Work Product of the Kick-off Milestone

- Preliminary draft charter (Reviewed, not yet Approved)
- Acting Chair, an Acting Vice-chair (if needed) for the new group identified

### Approvals for the Kick-off Milestone

- The Governing Committee and Dotted-line Committee (if present) need to informally approve (no vote required) the draft charter

## Group Pending Milestone

This phase begins with the formation of the group and continues through the approval of its preliminary charter and final (non-acting) leadership.

Once complete, continue with the Group Active Milestone.

### Background

While the previous phase was focused on defining the group, this phase is more about getting the group ready to perform its duties.

### Steps to Complete the Pending Milestone

1. Request RISC-V community tools/resources for the new group by creating an issue in the help.riscv.org GitHub repository. The Acting Group Chair or Vice-chair can use the “Group Tools Request” form to provide information about the new group. The RISC-V Staff will then correspond with the requester through the new issue.
2. Announce the early formation of the group to tech-announce and the Governing and Dotted-line Committee mailing lists. This step is generally handled by the Governing Committee Chair or Vice-chair with input from the Acting Chair and Vice-chair. An announcement template (link) can be used to craft the announcement email.
3. Begin meetings of the new group under the leadership of the Acting Chair and Vice-chair. See the Chairs Best Practices Policy bullet noted “Getting started”.
4. Continue to review the charter in the new group meeting and on the new group mailing list until both the new group and the Governing Committee are satisfied.
5. When the Governing and Dotted-line Committees, as well as the new Group are satisfied with the Charter, the Governing Committee Chair sends an email to help@riscv.org and requests to review the draft charter with the Technical Chairs.
6. Discuss within the new group the skills and requirements for group leadership. Use this information to run a Call for Candidates.
7. Once the Governing Committee has selected the Chair and Vice-chair nominees, they should seek approval for the group. (See approvals.)

8. After approval, the Governing Committee announces formation of the group via email to tech-announce, the Governing and Dotted-line Committee, and the new group mailing lists. Announcement templates can be used for TGs and SIGs.

### Work Product of the Pending Milestone

- Community tooling
- Approved Preliminary Draft Charter in the new community GitHub repository in the riscv-admin organization, Charter.md file.
- Nominees for Chairs and Vice-chair
- An announcement of the new community
- A new community ready to do work

### Approvals for the Pending Milestone

- The RISC-V Staff will review the Group Tools Request issue and work with the Acting Chair and Vice-chair to clarify any items, and create the tooling for the new community.
- The preliminary draft charter should be informally approved by the new group, any Dotted-line HC, and the Governing HC/IC.
- The group approval varies by group type. For HCs and TGs, the TSC needs to approve the charter and nominees for Chair and Vice-chair. Contact RISC-V Staff (help@riscv.org) to initiate the vote with the TSC. Note: the TSC will often require context for a new TG or HC such as the Governing Committee strategy, SIG gap analysis and prioritization for the topic area. This information enables the TSC to evaluate if all the pieces needed for group success.
  For SIGs, the Governing IC/HC approves charter and leadership.

## Group Active Milestone

The final group milestone before starting group technical work is very straightforward and quick: approve the final charter. After this is completed, the group can get to their real mission, whatever that may be!

Once complete and all group work has completed, the group will be finished.

### Background

Once the group is approved, the last activity of the group is to refine the Charter one final time. This final refinement typically occurs in parallel with, but no later than other

work plans, depending on the group types. TGs traditionally begin working on their Plan Milestone (in the Specification Lifecycle) by building a Ratification Plan for their specification. SIGs traditionally perform gap analysis and prioritization.

As reviews begin of a TG's Ratification Plan or a SIG's Gap Analysis, a final review of the charter should occur.

## Steps to Complete the Active Milestone

1. Review the charter, make any updates due for completeness or to correct for new information/direction.

## Work Product of the Active Milestone

- A Final Charter

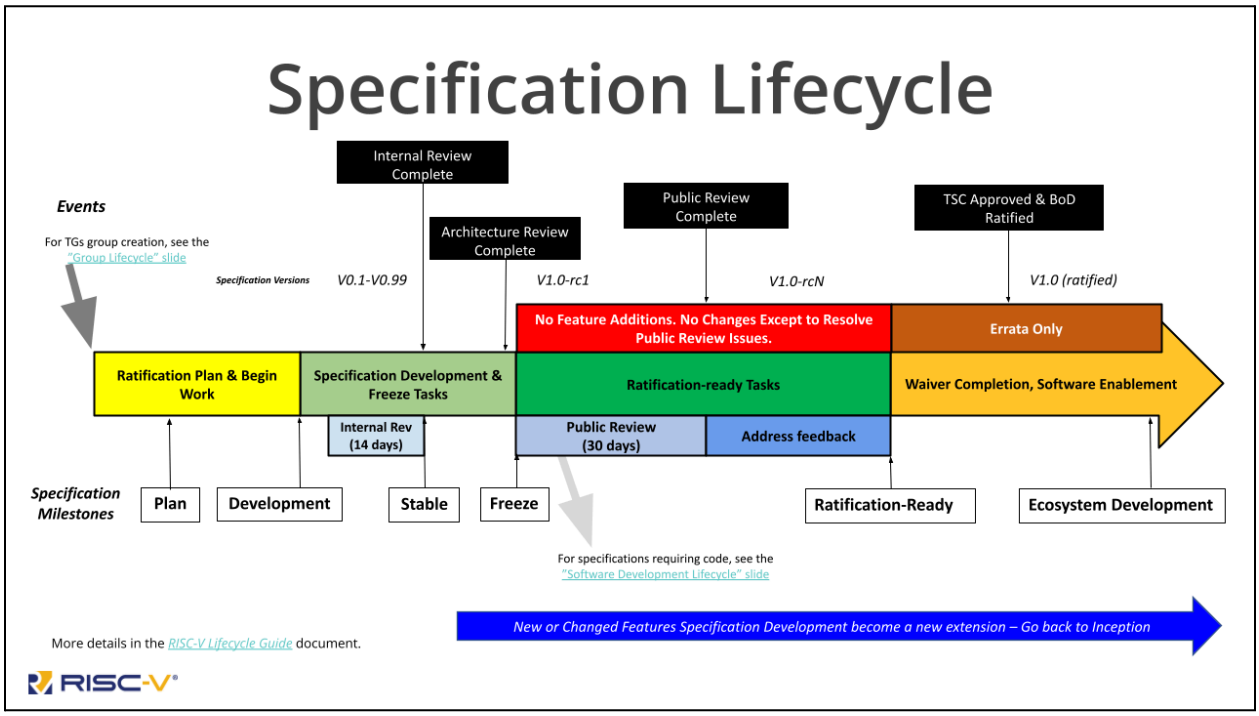
## Approvals for the Active Milestone

- The workgroup should review and informally approve the charter.
- The Governing HC should review and informally approve the charter.
- If the group is a TG or HC, the Tech Chairs should review and approve the final charter. Send an email to help@riscv.org to schedule a final charter review with the Tech Chairs.
  Note: this review is typically included in the Ratification Plan Milestone presentation.

# 3. Specification Lifecycle

The primary deliverable of the RISC-V Foundation is the Specification which defines the RISC-V ISA. This fact makes the Specification Lifecycle our most important process. It begins with a stimulus from Groups Lifecycle such as a Task Group being formed to develop a specification and will likely drive Software Lifecycle activity somewhere around the Freeze milestone.

Generally speaking, the traditional Specification Lifecycle can be represented this way:




The lifecycle is comprised of 5 sequential milestones:

1. The Plan Milestone creates two key documents for use throughout the whole lifecycle, the Ratification Plan and the Status Checklist.
2. The Development Milestone creates the early versions of the specification and declares them reasonably stable.
3. The Stable Milestone continues specification development until it is self-consistent and stable.
4. The Freeze Milestone brings the document to feature complete, declares that no substantive changes are planned, and prepares the document for Public Review.
5. The Ratification-Ready Milestone holds the Public Review, addresses any issues, and performs final activities before asking for TSC Approval and Board of Directors Ratification.

6. The Ecosystem Development Milestone (not listed in the picture) manages on-going open-source software community work across the multitudes of projects which can be impacted by a specification in a continued effort to achieve upstream community support.

RISC-V has two types of Specifications, ISA and non-ISA. The main ISA specification, *RISC-V Instruction Set Manual*, has two volumes, *Volume 1: Unprivileged ISA* and *Volume 2: Privileged Architecture*. New ISA extensions are generally grouped by function, reviewed, and ratified in a single Specification, also known as a Ratification Package.

There are two subtypes of specifications – traditional and Fast Track. Traditional specifications are driven by a Task Group who work under the guidance of a Governing ISA Committee (IC) or Horizontal Committee (HC) while Fast Track are owned and driven directly by an IC or HC. While Fast Track specifications generally have the same set of steps, these specifications are simpler architecturally, thus supporting a lighter-weight process.

**Note:** While Fast Track specifications have historically been ISA specifications, non-ISA Fast Tracks are allowed by the existing policies.

RISC-V also has ratified numerous non-ISA specifications, for software and hardware interfaces which support the platform but do not impact the ISA. Examples include the *RISC-V ABIs Specification*, *RISC-V Supervisor Binary Interface Specification*, and many others listed on the "RISC-V Technical Specifications" wiki page.

Each specification has 4 keys roles which contribute to the development:

1. The Initiator or requester of the work
2. The Governing Committee who sponsors the work
3. The Owner who drives the work
4. The Author who organizes the specification document

While all roles will find the information in the guide as helpful, the RISC-V Staff will primarily be interacting with the Owner. More information about these roles can be found in their linked definitions.

The development of all RISC-V specifications is facilitated by the two documents created in the Plan milestone: the Ratification Plan document and a Status Checklist spreadsheet. These two documents will be used and updated throughout the whole

lifecycle. The Ratification Plan document contains the schedule of when various milestones will be completed. The Status Checklist contains the items for each milestone (tab in the spreadsheet) which either need to be completed or have an approved Waiver.

The following subsections detail step-by-step activities for how to achieve each of these milestones. Throughout this work, the RISC-V Staff will serve as a resource and guide as-needed.

Additional details about specification development can be found in the RISC-V Ratification Policy.

In cases where simple updates are made to the specification and the creation of a Task Group (TG) to drive the specification work seems too heavyweight, an HC or IC may utilize the Fast Track Extension specification process with the approval of the Architecture Review Committee. See the Fast Track Extension Specification section for more details.

# Specification Plan Milestone

The first milestone in the Specification Lifecycle focuses on planning the activities and sharing that plan with the RISC-V organization.

Once complete, the lifecycle continues in the Specification Development Milestone section.

## Background

As previously described, the Ratification Plan document and a Status Checklist spreadsheet are crucial supporting documents for the Specification Lifecycle. This milestone establishes the schedule (Ratification Plan) and the contents (Status Checklist) of the target specification. Time spent understanding these documents and planning for the specification in this phase will pay-off during the execution of future milestones.

## Steps to Complete the Plan Milestone

1. Request a Ratification Plan document and a Status Checklist spreadsheet by creating an issue in the help.riscv.org GitHub repository. Use the “Specification

Status Documents" form, complete the information, and submit. The RISC-V Staff will use templates to start documents based on your specification type – Plan Document (ISA, non-ISA) and Status spreadsheet (ISA, non-ISA). They will provide links back to you with your new documents. All interactions will occur within the GitHub issue.
2. Finish filling out your Ratification Plan document and Status spreadsheet per the directions in the front material of each.
3. Create a working version of the Ratification Plan Milestone presentation (see the link for a template),preferably in your Group Folder in Google Drive, and edit the new copy to include necessary information.
4. Review ratification plan documents and the Ratification Plan Milestone presentation with and obtain (informal) approval of the Governing Committee.
5. Notify RISC-V Staff when you are ready to present your Ratification Plan document and Status Checklist spreadsheet to the Chairs and request approval of Plan milestone.

### Work Product of the Plan Milestone

- Ratification Plan document
- Status Checklist

### Approvals for the Plan Milestone

The Ratification Plan document and Status Checklist spreadsheet need the following approvals in order:

1. TG (informal)
2. Governing Committee (informal)
3. Chairs (informal, at the conclusion of the presentation in Step #5).
   Note: The TSC is informed of the plan milestone details via email following Chairs approval.

## Specification Development Milestone

This milestone starts work on the document.

Once complete, the lifecycle continues in the Specification Stable Milestone section.

### Background

All specification development occurs in GitHub repositories. Request one if needed.

New specifications should be developed in Asciidoc using the Asciidoctor tools. If you need more information on working in Asciidoc or writing specifications for RISC-V, see the *Authoring and Editing RISC-V Specifications* document.

### Steps to Complete the Development Milestone

1. If you need a new repo for the specification, create an issue in the help.riscv.org GitHub repository. Use the "Technical Repository Request" form, complete the information and submit.

   Within a day or two after the form is submitted, the RISC-V Staff will create a GitHub repository in one of the RISC-V GitHub Organizations and provide a working repository to build a basic document.

2. Begin specification development.

   The specification should have its State set to "Development".

   The specification Version should be `1.0.0-draft1` for a completely new specification or an ISA extension specification. Non-ISA specifications can be set to `MAJOR.MINOR.0-draft1` where the `MAJOR` and `MINOR` values are incremented as appropriate given the current version of the document and consistent with the proposed changes.

3. When a specification has stabilized to the point where it is ready for feedback, the specification should be sent out for Internal Review.

### Work Product of the Development Milestone

- A specification being built, marked "Development"

### Approvals for the Development Milestone

- Task Group Chair

## Specification Stable Milestone

This milestone continues development until the Task Group declares the document as "Stable" enough for consumers (like compiler writers or kernel developers) to start working on proof-of-concept (alpha) level code.

Once complete, the lifecycle continues in the Specification Freeze Milestone section.

### Background

The RISC-V Specification States define *Stable* as “Assume anything could still change, but limited change should be expected.” This criteria should help guide the decision to declare stability of the document.

While the criteria of stability is one that may be tempting to declare as soon as possible, Task Groups should carefully declare this achieved only when they understand who the consumers of the specification are and what their expectations may be.

### Steps to Complete the Stable Milestone

1. When the versions are self-consistent and stable, request approval from the Task Group to change the document state to “Stable”.

### Work Product of the Stable Milestone

- A specification, marked “Stable”

### Approvals for the Stable Milestone

- Task Group Chair

## Specification Freeze Milestone

A frozen document is both feature-complete and stable to the point that future updates are minimal, non-substantive in nature. This milestone includes all steps needed to have a frozen document in preparation for Public Review.

Once complete, the lifecycle continues in the Specification Ratification-Ready Milestone section.

### Background

RISC-V treats the Freeze Milestone as a major event by striving for no substantive changes after Freeze. This makes the work in getting to Freeze as the largest milestone in the process because work is being done in the specification and in the ecosystem. While the Steps to Complete section for this Milestone appears to have only a few steps, the real work exists in the number of items on the “Freeze” tab of the Status Checklist and effort required to accomplish them.

Additionally, Freeze is the milestone which must be achieved prior to the start of the Public Review period.

### Steps to Complete the Freeze Milestone

1. Complete Freeze Acceptance Criteria in the Status Checklist spreadsheet. Note: the items on this tab require significant work to complete in several areas. Review them immediately when starting to work on this step and ask the RISC-V Staff (help@riscv.org) any questions you have.
2. Notify RISC-V Staff all Acceptance Criteria items for the Freeze Milestone are complete or to request Waivers. The RISC-V Staff will schedule voting to obtain approvals.
3. Once the document has been approved as "Frozen" by all parties, produce a review-ready draft of the specification. Set the Version to `MAJOR.MINOR.0-rc1` and State to "Frozen". Increment the `-rcN` value with each update up through Ratification.

### Work Product of the Freeze Milestone

- All items on the "Freeze" tab of the Status spreadsheet are either "Done" or "Waivers granted" state
- A specification, marked "Frozen"

### Approvals for the Freeze Milestone

The following approvals are included in the "Freeze" tab of the Status spreadsheet:

- Architecture Review Committee (review of the technical content of the document)
- Committee Chairs (all must sign-off on the document to complete the Milestone)
- RISC-V Staff (review of specification against RISC-V policies)
- RISC-V CTO

## Specification Ratification-Ready Milestone

This milestone contains the Public Review phase, any updates to the document from the review, and any remaining Acceptance Criteria work such as previous waiver cleanup. Once complete, the approvals for Ratification are obtained.

Once complete, the lifecycle continues in the Specification Ecosystem Development Milestone section.

## Background

Given the previously mentioned RISC-V philosophy of no changes post Freeze, this step generally contains minor documentation cleanup (clarifications) and then work to prepare the final document version for ratification.

On the rare chance that a substantive change occurs following post review, the Architecture Review Committee will guide the specification back through the appropriate steps of the process as-needed.

## Steps to Complete the Ratification-Ready Milestone

1. Draft and review a review announcement email. (Public review email templates are available.) Contact the RISC-V Staff (help@riscv.org) before sending this email out to your review lists (generally isa-dev).
2. Collect and disposition all Public Review comments during the 30-day review period. Archive comments and responses in the Status spreadsheet, "Review" tab.
3. Confirm completion of Public Review with HC or IC.
4. Announce completion of Public Review (on the same lists as #1) and include a list of all issues raised and actions taken for each issue.
5. If needed based on the review, produce an updated, ratification-ready specification (version 1.0.0_rcX, "Frozen" state).
6. Complete work on any waivers from the Freeze Milestone.
7. Complete Ratification-Ready Acceptance Criteria and update the Status spreadsheet.
8. Notify RISC-V Staff (help@riscv.org) when you complete this milestone or to request any waivers for incomplete milestone Acceptance Criteria
9. When approved by the BoD, produce a ratified specification (version 1.0.0, "Ratified" state).
10. If the specification is an ISA specification, create a Branch of the riscv-isa-manual repository to add the specification to either the Unprivileged or Privileged specification as a chapter. Please email help@riscv.org for assistance with this task.
11. Issue a Pull Request to add the new chapter to the upstream riscv-isa-manual repository.
12. Notify RISC-V Staff requesting archival of the ISA specification original repository.

## Work Product of the Ratification-Ready Milestone

- All review comments logged on the "Review" tab of the Status spreadsheet
- All items on the "Ratification-Ready" tab of the Status spreadsheet are either "Done" or "Waivers granted" state.

Note: Ratification waivers require a vote of the TSC and thus are generally discouraged unless they are very pervasive or difficult to overcome.
- A specification, marked “Ratified”

### Approvals for the Ratification-Ready Milestone

The following approvals are included in the “Ratification-Ready” tab of the Status spreadsheet:

- Task Group approval of all issues and actions.
- Governing Committee approval of all issues and actions.
- Architecture Review Committee (re-review of changes to the document)
- Committee Chairs (all must sign-off on the document to complete the Milestone)
- RISC-V Staff (review of specification against RISC-V policies)
- RISC-V CTO

In addition, the RISC-V Staff will work with the specification owners to obtain these approvals (in order):

1. Technical Steering Committee approval for the specification to proceed to the Board of Directors for ratification
2. Board of Directors ratification

## Specification Ecosystem Development Milestone

Work on a specification does not complete with ratification of a specification, nor may it ever fully complete, especially for software. This milestone represents the on-going work which must occur post-ratification.

Once complete, the specification lifecycle formal work is complete. Maintenance of the document will be addressed by the Governing Committee.

### Background

This milestone represents on-going specification management and general open-source software development. The most frequent task is achieving upstream acceptance of RISC-V patches.

### Steps to Complete the Ecosystem Development Milestone

1. Complete work on any waivers from the Ratification-Ready milestone
2. Continue working Ecosystem Development Acceptance Criteria and updating the

Status spreadsheet.
3. Manage specification errata as needed (version 1.0.X, “Ratified” state)
4. Notify RISC-V Staff when you complete this milestone or need help managing errata

## Work Product of the Ecosystem Development Milestone

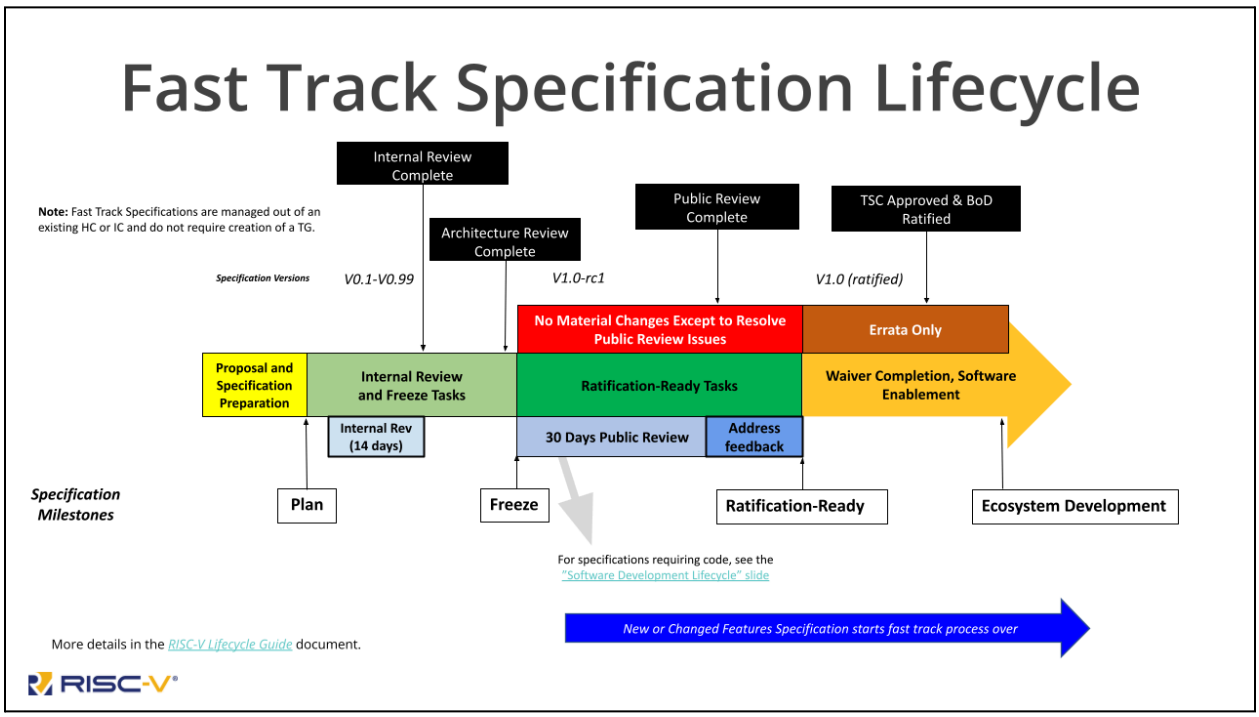
- All items complete on the “Ecosystem Development” tab in the Status spreadsheet

## Approvals for the Ecosystem Development Milestone

- No formal approvals needed, just TG and Governing HC informal approvals

# 4. Fast Track Extension Specification Lifecycle

Specifications which are so simple that they can be managed by a Horizontal Committee (HC) or ISA Committee (IC) instead of a Task Group (TG). The Architecture Review Committee determines whether proposed extensions qualify for consideration as Fast Track items. This determination is made in the Plan milestone, resulting in the need to detail this process in a separate section. However, after the Plan Milestone, Fast Track specifications complete the same milestones of Freeze, Ratification-Ready, and Ecosystem Development as traditional specifications. Thus, the process overview of Fast Track specification appears quite familiar as shown below.




Like the traditional Specification Lifecycle, development of Fast Track specifications is facilitated by the two additional documents: the Fast Track Proposal document (instead of the Ratification Plan) and a Status Checklist spreadsheet. The proposal document is used to determine if using the Fast Track process is appropriate. The Status Checklist contains the items for each milestone (tab in the spreadsheet) which either need to be completed or have an approved Waiver.

The two remaining sections detail the unique process for Fast Track Plan Milestone and then route the users to common activities in Fast Track Freeze, Ratification-Ready, Ecosystem Development Milestones.

For more in depth details about this process, the Fast Track Policy should be reviewed.

# Fast Track Plan Milestone

Even Fast Tracked Specifications have a Plan Milestone – where the specification development activities are determined. Specifically, the team needs to determine whether the specification is a candidate for a potential Fast Tracked Specification and discuss it with the Governing Committee (HC or IC).

Once complete, the lifecycle continues in the Fast Track Freeze, Ratification-Ready, Ecosystem Development Milestones section.

## Background

Like the Specification Lifecycle, the Fast Track Plan Milestone builds the Ratification Plan document and a Status Checklist spreadsheet, establishing the schedule (Ratification Plan) and the contents (Status Checklist) of the target specification.

## Steps to Complete the Fast Track Plan Milestone

1. Create a Fast Track proposal document for the new extension with rationale describing why the new extension does not need a full TG to develop it. Store the document in the Google Drive Group Folder for the Governing Committee. Review the proposal with the owning HC or IC and get their permission to proceed to approval from the Architecture Review Committee.
2. Request approval via email (template) from the Architecture Review Committee. When received, notify the RISC-V Staff via email.
3. Request a Status Checklist spreadsheetby creating an issue in the help.riscv.org GitHub repository. Use the “Specification Status Documents” form, complete the information, and submit. The RISC-V Staff will use a template to create a Status spreadsheet (ISA template). When they will provide a link back to you, begin a discussion with them on which checklist criteria are needed. All communication will occur through the GitHub issue.
4. Finish filling out your Status spreadsheet per the directions in the front material of the document. Because this is a Fast Track item, many items in the list may not apply. Mark these items as “N/A” in the “Status” column and provide justification in the “Notes” column justifying the status.
5. Create a working version of the Ratification Plan Milestone presentation (see the link for a template), preferably in your Group Folder in Google Drive, and edit the new copy to include necessary information.
6. Review Fast Track proposal and the Ratification Plan Milestone presentation with and obtain (informal) approval of the Governing Committee.

7. Notify RISC-V staff (help@riscv.org) when you are ready to present your Fast Track proposal and Status Checklist spreadsheet to the Chairs and request approval of Plan milestone.

### Work Products of the Fast Track Plan Milestone

- Fast Track proposal document
- Status Checklist

### Approvals for the Fast Track Plan Milestone

The Fast Track Proposal document needs formal approval from the Architecture Review Committee.

Additionally, the Fast Track Proposal document and Status Checklist spreadsheet need the following approvals in order:

1. Governing Committee (informal)
2. Chairs (informal, at the conclusion of the presentation in Step #6).
   Note: The TSC is informed of the plan milestone details via email following Chairs approval.

## Fast Track Freeze, Ratification-Ready, Ecosystem Development Milestones

These Fast Track Milestones follow the standard specification milestones defined in the Specification Lifecycle sections:

- Specification Freeze Milestone
- Specification Ratification-Ready Milestone
- Specification Ecosystem Development Milestone

These sections should be followed, in order, for Fast Track extensions.

Note: Fast Track specifications may use the Specification Development Milestone and Specification Stable Milestone milestones, but typically skip directly from Plan to Freeze due to the simplicity of their specifications.

# 5. Software Development Lifecycle

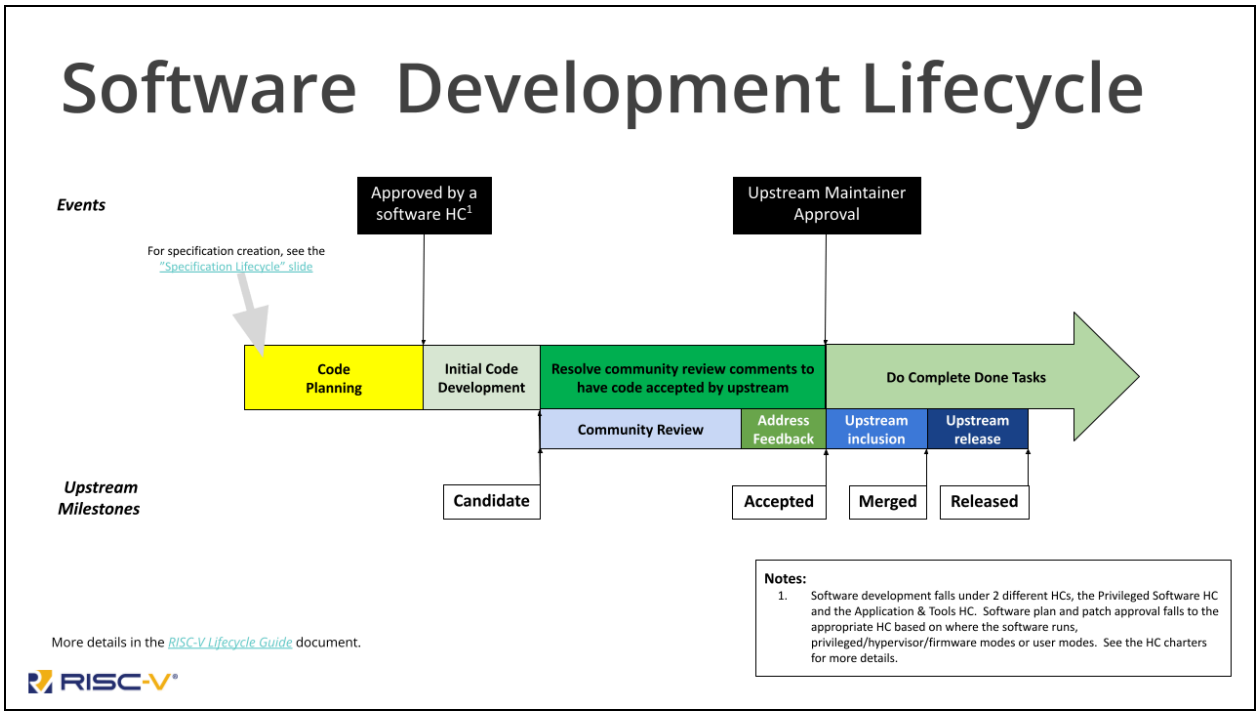
While enablement of platform-independent, open-source software remains a priority for RISC-V, the process to do so may be overshadowed by more hardware-centric activities such as designing new extensions to the RISC-V ISA. However, to ensure that all RISC-V members can benefit from flexibility of the platform, we should make sure to build complete plans to include software enablement with extension development.

The Software Development Lifecycle actually begins during the Plan Milestone of Specification Lifecycle. Many of the activities in the Acceptance Criteria Status Checklist involve software, especially on the “Freeze” and “Ecosystem Development” tabs. Thus, completion and approval of a complete specification plan, must include the plans for software as well.

Beyond the planning, the RISC-V Software Development Lifecycle has 5 additional milestones:

1. Candidate - The code and tests are written and available in preliminary form.
2. Accepted - The code and tests have been submitted to the appropriate up-stream mailing list for review and has been accepted by the community maintainer.
3. Merged - The code and tests have been merged into the upstream community tree.
4. Released - The code and tests have been bundled into a community release and are available.

The relationships, activities, and milestone are visually represented below:




The Software Development Lifecycle should begin with the Specification Plan Milestone or Fast Track Plan Milestone which should include which software items in the Status Checklist spreadsheet will be completed and who will complete them. Core software (e.g. basic toolchain enablement) then needs to complete the Candidate milestone to satisfy the Specification Freeze milestone. All remaining software lifecycle work, such as community acceptance, can be completed independent of the specification lifecycle and frequently occurs post specification ratification.

The subsequent sections describe in detail the steps necessary to complete each milestone. They should generally be worked in the order they are presented with one exception: exceptionally tricky code, exceptionally innovative code, or exceptionally pervasive code may need to begin work upstream sooner than depicted by the process. If you have such a feature, please work with the appropriate software HC to modify the order of the process to increase efficiency and chance of success.

Unlike the Group and Specification Lifecycles, the Software Development Lifecycle has not been committed to a policy. Therefore, for more information, one should work directly with the software HCs for guidance.

# Software Candidate Milestone

This milestone provides a first-pass, functionally complete implementation of the code and test cases to meet the requirements.

When complete with this milestone, continue work in the Software Accepted Milestone section.

## Background

While this is arguably the simplest task to describe, it takes the majority of the developer effort as it really is the core of the work.

## Steps to Complete the Candidate Milestone

1. Implement test cases
2. Complete functional implementation of code
3. Pass developer testing with the defined scope (test cases, applications, benchmarks)
4. Collect performance data available (if applicable)

## Work Product of the Candidate Milestone

- Functional tests
- Functionally complete code, ready for review

## Approvals for the Candidate Milestone

- The TG Chair for the feature or component under development (i.e. the TG who owns the spec being used for reference during development) will determine when the candidate code has been accepted for the Candidate Milestone. This is an informal approval.

# Software Accepted Milestone

After completion of the candidate code, the next step is to receive upstream approval so that inclusion of the patches in the main community will occur.

When complete with this milestone, continue work in the Software Merged Milestone section.

### Background

External submission of patches to the various communities can often be the hardest step in the process. Coding styles must be followed. Patch submission protocols must be followed. Commit messages need to be in the proper form. And, just when you think you understand how things will go, someone or something surprises you.

Remember, we are often the newcomers in these communities and therefore must follow the guidance we are given, even when it makes no sense to us: we need them, and they are helping us. The key to successfully completing this phase is patience, humility, flexibility, and persistence. Expect this set of activities to take more time than you expect and do not try to rush it.

### Steps to Complete the Accepted Milestone

1. Rebase code and test cases to upstream head version, if needed.
2. Submit patches to the upstream community, following community norms for requesting acceptance. For key considerations, see the RISC-V wiki page How to contribute to the RISC-V SW ecosystem.
3. Resolve comments and re-post patches to mailing lists as needed.
4. Receive upstream ("root" project — e.g. mainline gcc, mainline qemu, mainline linux) maintainer's approval for merge (ACK, OK, …) according to the upstream project.

### Work Product of the Accepted Milestone

- RISC-V code and test patches which are ready for inclusion in the upstream community.

### Approvals for the Accepted Milestone

- Community maintainer must agree to accept the code and patches.

## Software Merged Milestone

Just because a community agrees to a patch, does not mean that the work is done yet. The code and tests must be merged into the upstream tree.

When complete with this milestone, continue work in the Software Released Milestone section.

### Background

Once a patch is accepted, the next step is inclusion of the code into the main community tree. Most often, this occurs using a PR (Pull Request) to the head or main branch of the tree. Sometimes, however, communities will stage the next release into a branch that the maintainer will merge into main. All communities are different. Thus, it is important to follow the guidance you receive when your patch is accepted and continue to track your code until it is in the upstream tree.

### Steps to Complete the Merged Milestone

1. Await patch merge into an upstream repository (next-branch, main/master-branch).
2. If delayed, politely request status. Goto #1.
3. If included, feel happy!!!

### Work Product of the Merged Milestone

- RISC-V code and test patches included in the upstream community.

### Approvals for the Merged Milestone

- The community maintainer who approved the patch will typically drive and handle inclusion of all patches.

## Software Released Milestone

The final step for a patch within the RISC-V Software Lifecycle is that of releasing the software so that developers and users can benefit from the work.

Once complete, the Software Lifecycle work for this activity is finished.

### Background

While most operating systems have a distribution that bundles all packages into whole releases, most software communities have their own cycle of bundling up the software for their project and releasing it. Some even have releases that they provide bug fixes against (named “Long Term Support” or “LTS” release) for a fixed period of time.

Once code is included in the tree, developers usually have almost nothing to do but wait. However, if the maintainer does find a problem in the area of code which has been changed, one can and should help triage the problems just in case we introduced

it. Even if we did not, earning community Karma points for engaging helps all members of the RISC-V community.

## Steps to Complete the Released Milestone

1. Await patch release as part of a generally-available, mainline release.
   *Note: release via a third-party distribution like Fedora or SLE is not required.*

## Work Product of the Released Milestone

- A community release that contains the RISC-V patch.

## Approvals for the Released Milestone

- None usually required as this step is automatic. Larger communities may have a Build Team that facilitates the build and they should be supported as needed.

# Appendix

## Definitions

| Term | Definition |
|---|---|
| Acceptance Criteria | The Ratification Policy requires that each milestone has items which must be completed which are generally referred to as “Acceptance Criteria”. The governance of the criteria for each milestone are governed by the Acceptance Criteria Policy. The instantiation of the Acceptance Criteria for a given specification is contained in the Status Checklist spreadsheet. |
| Acceptance Criteria Policy | The RISC-V policy governing the criteria which must be met is defined in the Acceptance Criteria Policy (formerly the Definition of Done Policy).<br><br>The approved document:<br>Acceptance Criteria Policy |
| Architecture Review Committee | A RISC-V special committee that generally has the responsibility for reviewing all ISA architecture on its technical merits to make sure new extensions conform to the broader architecture and vision of the RISC-V Platform. This committee also owns the review and approval of Fast Track specification proposals. For non-ISA documents, the Architecture Review Committee further plays the role of general specification reviewer.<br><br>Information about the Architecture Review Committee can be found on the RISC-V Architecture Review wiki page.<br><br>The proposed policy document:<br>architecture review |
| Authoring and Editing RISC-V Specifications | Also known as the “Doc Dev Guide”. This document provides guidance on writing RISC-V specifications in Asciidoctor. |

| Term | Definition |
|---|---|
| | The GitHub project:<br>https://github.com/riscv/docs-dev-guide/<br><br>The latest PDF can be found in the Release Assets (dev-guide.pdf):<br>https://github.com/riscv/docs-dev-guide/releases |
| Call for Candidates | The process by which RISC-V leadership is selected. There are 3 basic steps:<br>1. The group defines the skills and qualifications for the new leader(s).<br>2. The Acting Chair and/or the Governing Committee Chair generates an announcement email by opening a "New Call for Candidates" help issue at help.riscv.org. The template email and directions will be posted as a comment after submission.<br>3. The Governing Committee Chair or Vice-chair sends an email to tech-announce, the governing IC/HC, any dotted-line lists, and the group mailing list that includes the skills/qualifications or the new leader(s), references the group charter, and asks for candidates by a date at least 2 weeks later.<br><br>Note: The call announcement should be reviewed and approved (informally) by the Governing Committee and Dotted-line Committee Chairs before sending.<br><br>Note: Incumbent or Acting Chairs for the group holding the call may open the issue, but the call email should be sent to the mailing lists by the Governing Committee Chair or Vice-chair so as to maintain an open process.<br>4. The nominees are selected from the candidates by the Governing Committee in consultation with any Dotted-line Committees.<br>More details can be found in the Groups & Elections Policy. |

| Term | Definition |
|---|---|
| Chairs Best Practices Policy | The document providing guidance how Chairs and Vice-chairs should run their meetings.<br><br>The approved document:<br>Chairs Best Practices |
| Committee Chairs | The Committee Chairs refers to the Chairs and Vice-chairs of all ISA Committees (ICs) and Horizontal Committees. In the context of specification sign-off for Freeze or Ratification-ready, this applies to the Chairs only. In the context of meetings (CCM), this applies to both Chairs and Vice-chairs. |
| Definition of Done Policy | The policy governing the criteria for specification milestone completion.<br><br>The approved document:<br>Definition of Done<br><br>It has been deprecated in favor of the Acceptance Criteria Policy. |
| docs-spec-template | A GitHub project template used for priming new specifications. It includes a basic document that contains enough basics to ensure a proper foundation.<br><br>Project URL:<br>https://github.com/riscv/docs-spec-template/ |
| Dotted-line Committee | The Dotted-line Committee is a co-sponsor committee that provides additional, technical oversight for TGs. Typically, this occurs for ISA extensions which need additional domain expertise such as Security, Firmware, etc. |
| Fast Track Policy | The RISC-V policy that describes a simplified process for specification ratification. This process delegates specification oversight from a Task Group back to the governing Horizontal or ISA Committee. Approval to use this process for an ISA specification rests with the Architecture Review Committee.<br><br>The approved document:<br>Fast Track Architecture Extension Process |

| Term | Definition |
|---|---|
| Fast Track Proposal | A proposal document requesting than a proposed specification be allowed to follow the Fast Track Policy process for development.<br><br>A template for this proposal is available:<br>Fast Track ISA Extension Proposal Template |
| GitHub Repo Structure & Administration Policy | The RISC-V policy describing how GitHub should be used for hosted projects.<br><br>The approved document:<br>GitHub Repo Structure & Administration |
| Governing Committee | The Governing Committee is the parent committee for one or more TGs or SIGs within a common area of expertise, e.g. Privileged ISA, Security, etc. and provides sponsorship and governance for the group. The Governing Committee Chair and Vice-chair work to create the sub-group, organize all groups, navigate the RISC-V processes, and deliver the work products created by their sub-groups (strategy, gap analysis, specifications, code, etc.) to the TSC.<br><br>For specifications, the Governing Committee approves progression to the next milestone. If there are issues during the life cycle, they should be addressed by the Governing Committee. A Governing Committee may delegate day to day execution of the development of a specification but must be the one approving and sponsoring the steps in the life cycle. |
| Groups & Chairs Policy | The document which defines how groups get formed and how they select their leadership.<br><br>The approved document:<br>Groups & Chairs |
| Group Folder | A folder for the group (HC, IC, TG, SIG, etc.). It is generally a sub-folder of the RISC-V Google Drive “Workgroups” folder:<br>for risc-v members -> Workgroups |
| HC(Horizontal) | Horizontal Committee. Has responsibilities to make sure that all Extension TGs cover the area overseen by the HC before ratification. Responsible for developing a holistic strategy and reaching out to the external ecosystem and community |

| Term | Definition |
|---|---|
| | groups.<br><br>RISC-V has 6 HCs – Applications & Tools Software HC, ISA Infrastructure HC, Privileged Software HC, Security HC, SOC Infrastructure HC, and Technology HC. |
| IC (Vertical) | Instruction Set Architecture (ISA) Committee. Approves and oversees packages for TSC vote for the creation of ISA Extension TGs and filling the chair and vice-chair vacancies for its TGs. Develops strategy for the groups under it and complete coverage of areas of responsibility under it including gaps.<br><br>RISC-V has 2 ICs, the Unprivileged IC and the Privileged IC, each covering their respective portions of the ISA. |
| Internal Review | The phase in the specification development process where comments are generally solicited from RISC-V members across various communities before a document goes to the Architectural Review Committee.<br><br>Internal Reviews generally are announced on the tech-announce, tech-chairs (Tech Chairs), Governing Committee, and any Dotted-line Committee mailing lists and remain open for 2 weeks.<br><br>NOTE: While this review has no formal process requirements for documenting and addressing issues raised during review, tracking issues in GitHub and summarizing them to a group mailing list post review are strongly encouraged. |
| isa-dev | The public mailing list (non-members may post) for RISC-V ISA discussions hosted on Google Groups: https://groups.google.com/a/groups.riscv.org/g/isa-dev/about<br><br>Note: To post to this list, one must be a member. So, if your post fails with a message like “We're writing to let you know that the group you tried to contact (isa-dev) may not exist, or you may not have permission to post messages to the group…”, please check your group membership.<br><br>Email reflector: |

| Term | Definition |
|---|---|
| | isa-dev@groups.riscv.org |
| Lifecycles and Milestones Overview | An overview presentation of the RISC-V Lifecycles and their Milestones.<br><br>The document:<br>RISC-V Lifecycle and Milestone Definitions -- WIP |
| New Technical Repo Form | A Google document, created from the New Technical Repo Form template document which requests one or more GitHub technical repositories. |
| Public Review | The phase in the specification development process where comments are generally solicited from non-members. Public Reviews generally are announced on the tech-announce and isa-dev mailing lists and remain open for 45 days.<br><br>Note: RISC-V Public Review documents have spent considerable time being developed prior to the Public Review phase such that significant changes are generally unanticipated and considered unlikely. Thus, those members wishing to engage in deep technical discussion and to influence a desire should be working in the Task Group (TG) which is developing the specification, not waiting for Public Review to provide feedback. |
| riscv-admin organization | One of several RISC-V GitHub organizations. The riscv-admin organization is used to host all repositories which are used for group administration. It typically contains the group's Charter in the Charter.md file as well as all meeting minutes and content. |
| Ratification Package | A group of ISA extensions which are grouped together, generally by domain or function, into a single Specification which is ratified as a single unit. The ratified contents of this specification (and ratification package) is later included into the appropriate volume of the RISC-V ISA manual.<br><br>Note: Because non-ISA specifications generally are grouped by functionality and because non-ISA specifications continue to be maintained as-reviewed (not included in another document), the concept of Ratification Packages does not apply to non-ISA specifications. |

| Term | Definition |
|---|---|
| Ratification Plan | The document for a given specification that describes when various specification development events will occur. Ratification Plan documents typically link to a Status Checklist Spreadsheets which reflect the Acceptance Criteria necessary for each specification milestone.<br><br>A specification's ratification plan document is generally derived from a template, based on whether the specification is for an ISA extension or is non-ISA. The templates are:<br>• ISA<br>• non-ISA<br><br>Once created, a specification's ratification plan is linked into the RISC-V Specification Dashboard. More generally, all specification Ratification Plan documents can be found in a subdirectory of Google Drive:<br>for risc-v members -> Status -> 1 - Ratification Plans<br><br>See the Specification Milestone 1: Plan for information about how to create a Ratification Plan. |
| Ratification Plan Milestone Presentation | To facilitate the Ratification Plan Milestone review with the Tech Chairs, this template presentation has been created:<br>Ratification Plan Milestone Review Template<br><br>To create your own version, open the document, make a copy to your Group Folder, and edit the items as described in the directions on the README slide. |
| Ratification Policy | The document which defines how specifications get developed and ratified by RISC-V.<br><br>The approved document:<br>Ratification Policy |
| RISC-V GitHub Organizations | For organizational purposes, RISC-V has established several GitHub organizations into which all repositories are collected. The list of organizations and their purposes are detailed on the RISC-V GitHub Repo Map wiki page. |

| Term | Definition |
|---|---|
| RISC-V Patch Mailing Lists | RISC-V member mailing lists used to review patch submissions prior to being sent upstream. Details about the lists are found on the Development partner lists wiki page. |
| RISC-V Staff | Sometimes called, “The TPMs”. One or more RISC-V Technical Program Managers who facilitate RISC-V operations.<br><br>Create issues in the help.riscv.org GitHub repository to request assistance. Forms are provided for common requests.<br><br>Unique requests can be submitted via a “General Request” issue or emailed to help@riscv.org. |
| SIG | Special Interest Group. Develops strategy for complete coverage of areas of responsibility under it including gaps. Provides continuity on the topic of TGs and may request TGs be created. SIGS produce no work product. Can be created by the TSC, ICs or HCs with TSC approval not required. |
| Software HC | Prior to early 2022, all RISC-V software was handled by a single Horizontal Committee (HC). Today, software falls into either the Privileged Software HC (https://lists.riscv.org/g/privileged-software) or the Application & Tools HC (https://lists.riscv.org/g/software) depending on its function. For simplicity, the term “software HC" has been left to describe either HC and which one should be used is left for the reader to determine. Consult the HC Charters (in their riscv-admin GitHub repos) for a more detailed explanation of which software each group supports. |
| Specification | The formal document which gets ratified by the RISC-V Board of Directors via the Ratification Policy. |
| Specification Dashboard | See Specification Status. |
| Specification Author | The primary writer, editor, and organizer of a specification.<br><br>The Specification Owner works with the Author to help drive a specification through the lifecycle but the Author is not directly responsible for interfacing with the Governing Committee or |

| Term | Definition |
|---|---|
| | RISC-V Staff, but likely will be involved in the specification review with the Architecture Review Committee. |
| Specification Initiator | The requester of an extension or specification.<br><br>Anyone can request that a committee consider creating a new specification (and Task Group) or Fast Track specification. |
| Specification Owner | The first point of contact for the specification. The Owner is responsible for driving the specification through the lifecycle and ensuring all the work gets completed to ratify the specification.<br><br>For standard (non-Fast Track) specifications, the Owner is the Chair of the Task Group. For Fast Track specifications, the Owner is typically the Specification Initiator but the Governing Committee may identify an alternative Owner.<br><br>If the Task Group members and Development Partners cannot help, the Owner needs to escalate to the Governing Committee to resolve the resource gaps. The Governing Committee o in turn may escalate to the RISC-V CTO, who may escalate to TSC, the RISC-V CEO, or the BOD.<br><br>Resources should generally be identified in the Plan Milestone and any gaps identified during the presentation to the Tech Chairs.<br><br>Governing Committees should not sponsor a Fast Track without an identified Owner. |
| Specification States | The RISC-V specification states are defined on the Specification States wiki page at: https://wiki.riscv.org/display/HOME/Specification+States<br><br>The text associated with each of these states should appear in each specification, in the front matter, such that readers have appropriate expectations. Documents created by the RISC-V Staff based on docs-spec-template will contain the state information in the `:revremark:` text (usually found in the `[[header]]` section). Additionally, this text will be in be contained in a `[WARNING]` in the Preamble, just ahead of the |

| Term | Definition |
|---|---|
| | copyright and licensing information. It should look something like:<br><br>`[WARNING]`<br>`.This document is in the`<br>`link:http://riscv.org/spec-state[Development state]`<br>`====`<br>`Assume everything can change. This draft specification will change before being accepted as standard, so implementations made to this draft specification will likely not conform to the future standard.`<br>`====` |
| Specification Status | Sometimes called the “Specification Dashboard”. A wiki page where the status of all active specifications is maintained.<br><br>URL to page:<br>https://wiki.riscv.org/display/HOME/Specification+Status |
| Specification Versioning | Versioning of RISC-V specification is managed by the Versioning policy.<br><br>Key highlights:<br>● Versioning will follow the `MAJOR.MINOR.PATCH` paradigm<br>● Early development versions should be marked with an appended `-draftN` where `N` is a number that increments with new versions up until the Freeze Milestone. This means the first document created should be versioned `1.0.0-draft1`.<br>● Beginning with Freeze, all documents should be marked using `MAJOR.0.0-rcN` where `N` is a number that increment with new versions up to Freeze, e.g. `1.0.0-rc3`<br>● The first ratified version will be `1.0.0`.<br><br>Note: New ISA extension specifications by definition will always be 1.0.0, but new non-ISA specifications may have MAJOR numbers other than 1.<br><br>In the RISC-V Asciidoc template, the version is set using the `:revnumber:` variable which is usually found in the `[[header]]` section. |

| Term | Definition |
|---|---|
| | The proposed document:<br>Versioning |
| Status Checklist Spreadsheet | The spreadsheet for a given specification that describes what Acceptance Criteria are necessary for each specification milestone.<br><br>A specification's status checklist spreadsheet is generally derived from a template, based on whether the specification is for an ISA extension or is non-ISA. The templates are:<br>• ISA<br>• non-ISA<br><br>Once created, a specification's status checklist is linked into the RISC-V Specification Dashboard. More generally, all specification Status Checklist documents can be found in a subdirectory of Google Drive:<br>for risc-v members -> Status -> 2 - Status Checklists<br><br>See the Specification Milestone 1: Plan for information about how to create a Status Checklist Spreadsheet. |
| sw-dev | The public mailing list (non-members may post) for RISC-V software discussions hosted on Google Groups:<br>https://groups.google.com/a/groups.riscv.org/g/sw-dev/about<br><br>Note: To post to this list, one must be a member. So, if your post fails with a message like "We're writing to let you know that the group you tried to contact (sw-dev) may not exist, or you may not have permission to post messages to the group…", please check your group membership.<br><br>Email reflector:<br>sw-dev@groups.riscv.org |
| tech-announce | The main mailing list in the RISC-V portal where all technical announcements are posted:<br>https://lists.riscv.org/g/tech-announce<br><br>The email reflector exists at tech-announce@riscv.org. |
| tech-privileged | The Groups.IO Privileged (Priv) IC community portal: |

| Term | Definition |
|---|---|
| | https://lists.riscv.org/g/tech-privileged<br><br>Email reflector:<br>tech-privileged@lists.riscv.org. |
| Tech Chairs | Short for “Technical Chairs”. A regularly occurring meeting that occurs usually on Wednesday mornings, U.S. time. All Active group (HC, IC, SIG, and TG) Chairs and Vice-chairs are invited. An invite-only group mailing list is hosted in Groups.IO at https://lists.riscv.org/g/tech-chairs and the email reflector is tech-chairs@lists.riscv.org. Membership is managed by the RISC-V Staff. |
| Technical Organization | A presentation describing the RISC-V technical organization.<br><br>Document at:<br>RISC-V Technical Organization |
| Task Group (TG) | Task Group. Must have a charter that defines a small set of deliverable work products: extension specifications, standards, requirements, best practices, etc. TGs under the Unpriv and Priv ICs can have ISA extension work products. TGs under HCs should not have ISA extension work products. |
| TSC | Technical Steering Committee. Owns technical responsibilities to organizational components below it – strategy, escalations, group & chair & preliminary charter approvals, ratification voting. Many of the TSC’s responsibilities are delegated to the organizational components it leads. |
| Waiver | A formally approved deviation of one or more Acceptance Criteria items in the Status Checklist and used during the Specification Lifecycle or Fast Track Extension Specification Lifecycle. Waivers are requested by the specification TG, sponsored by their Governing Committee, and approved by either the Tech Chairs (Freeze Waiver) or TSC (Ratification Waiver).<br><br>Typical reasons to request a waiver are:<br>● An ISA extension has a feature not easily supported within the framework of an existing test area such as a |

| Term | Definition |
|---|---|
| | simulator or test harness.<br>● The resources needed to accomplish work are backlogged such that the work may not occur in a timely fashion and the desire is to proceed without waiting. Note, the waiver still requires a commitment and plan to complete.<br><br>Because approval of waivers requires a vote, it is strongly recommended that as soon as you think you might need such a waiver that you begin a discussion with the RISC-V Staff. |

# Frequently Asked Questions (FAQs)

## I have a specification from my company, how do we get it ratified?

RISC-V does not have a process for simply ratifying externally developed specifications. All specifications are developed by a Task Group (TG) whose need has been identified by a Special Interest Group (SIG) or a Committee (an ISA Committee or IC or a Horizontal Committee HC).

So, what really needs to occur is the following set of steps:

1. Instead of identifying the solution, up-level the discussion to what problem is being solved.
2. Locate the IC, HC, or SIG which has a vested interest in solving this problem. This is your potential “sponsor”. If you cannot find one, reach out to the RISC-V Staff for guidance.
3. Meet with the potential sponsors and see if they agree that this is a problem, and is high enough priority to begin solving. If so, work with the sponsor group, and likely the RISC-V CTO to build the correct path for defining the problem to be solved. If there are multiple solutions, a SIG may be required to explore all solutions and pick.
4. When it has been determined that a new group needs to be formed, follow the directions in the Groups Lifecycle.
5. Be ready to volunteer to drive any work that needs to be done. RISC-V is a “contributor culture”.
6. Accept that the current solution implementation may not be the final implementation. Open development is a grass-roots, organic, unique, and sometimes frustrating process.

Thank you for your interest!

## How do I propose an ISA Extension?

Evaluation of future RISC-V ISA extensions should begin with a discussion on the isa-dev mailing list. Generally speaking, this list contains many experts who have been contributing to and discussing the ISA for many years.

The recommended way to start such a discussion is with a question such as, “Has the RISC-V community ever considered an extension to accomplish <your idea here>?” or “What extensions does RISC-V have to accomplish <your idea here>?” This approach avoids jumping to conclusions about an implementation and acknowledges that this idea may have already been discussed in the past or might even already exist.

If your idea is one that is recommended to be pursued, the next question would be whether this is a full specification, requiring a task group to write the specification or whether this can be done as a Fast Track extension under the guidance of an ISA Committee (IC). If the former, you will need to begin reading the Group Lifecycle section. If the latter, you can start in the Fast Track Extension Specification Lifecycle section.

Questions at any point along this journey can be routed to the RISC-V Staff.

### How do I know if my document is an ISA or Non-ISA document?

ISA documents are any documents that impact the processor instruction encoding or state. Any document that requires even the slightest change should be treated as ISA. Everything else is non-ISA.

There is no such thing as a “hybrid” document, it’s an ISA document if you need any changes.

## Does RISC-V support non-ISA specifications using the Fast Track process?

While RISC-V has historically used the Fast Track process as defined by the Fast Track Policy for ISA specifications, the process may also be applied to non-ISA documents. If you need help or have questions, please contact the RISC-V Staff with questions or to request assistance.

## How do I become a Chair or Vice-chair?

As with many groups in life, leaders are most frequently selected organically from within the organization. Thus, the best way to become a group leader is to join and to contribute technically to the group before seeking leadership.

The RISC-V process by which leaders are solicited is the Call for Candidates during which potential leaders are identified (usually self-identified, not nominated). Interested candidates provide a brief bio and a short Statement of Intent (description of how they plan to lead and/or what they want to accomplish). The call is generally held open for 2 weeks.

Upon closure, the candidates are reviewed and a selection of a nominee is generally made. Final approval of a nominee varies by the group and may simply involve approval of the Governing Committee (SIGs only) or may require vote by the TSC.

More details about the Call for Candidates and the approval process can found in the Groups & Elections policy.

## I have a concern with a specification which is under development, how do I raise an issue?

First there are a couple of policies that you should check out: Ratification Policy, Acceptance Criteria Policy, and Fast Track Extension Policy. We don't profess to be perfect so if you find something that we could communicate better about or improve please contact the RISC-V Staff.

During the specification lifecycle, the natural ways to raise an issue for a specification are:

- For standard (non-fast track) specifications, join the owning task group's email list and go to meetings. This is the earliest way to influence a specification. Issues can be posted to the email list or github issues.
- Participate in the Internal Review. All specifications need a Ratification Plan that includes an Internal Review. The best way to find out about the ratification plans is to find the specification in the Specification Dashboard and review the details.
- Participate in the Public Review. Review details are sent to the tech-announce and isa-dev mailing lists and occur for at least a 30 day period.

We require that all open issues are resolved before going for ratification by the TSC and BOD. However, we cannot satisfy everyone's requests but we aim to make sure everyone feels heard.

If you have gone through this process and don't feel heard or you think we missed something, you can either contact help@riscv.org or your elected official at TSC (the elected officials are announced on tech-announce).

### I didn’t see any plans for SIMD extensions. Do they exist?

We plan to define more optional instruction set extensions for RISC-V beyond the ones we already have, including Packed-SIMD Instructions, Decimal Floating-Poin and Transactional Memory. One goal for RISC-V International is to manage development of these future standard instruction set extensions.

The currently defined extensions to the base Integer (I) ISA are Multiply-Divide (M), Atomic (A), Floating-point in multiple precisions (F, D, and Q), and Compressed Instructions (C).

### How fast are RISC-V processors compared to x86 or Arm processors?

This depends entirely on the quality of the implementation, including microarchitectural design, circuit design and process technology used. We believe there are no fundamental reasons that a RISC-V implementation should be less efficient than x86 or Arm, and indeed that the ISA design should enable implementations to be somewhat more efficient than either.

### Are RISC-V processors lower power than Arm processors?

This depends entirely on the quality of the implementation, but we feel RISC-V implementations should be at least comparable in energy efficiency to Arm cores built in the same microarchitectural style and with the same engineering effort in the same process technology.

### What about Vendor ID assignment for RISC-V implementations?

RISC-V International Vendor ID assignment uses JEDEC manufacturer IDs as defined in the RISC-V ISA Privileged Architecture Specification v1.10 Section 3.1.2 Machine Vendor ID Register mvendorid and repeated here for convenience.  Refer to the full Privileged Architecture Specification here.

*3.1.2 Machine Vendor ID Register mvendorid*
*The mvendorid CSR is an 32-bit read-only register providing the JEDEC manufacturer ID of the provider of the core. This register must be readable in any implementation, but a value of 0 can be returned to indicate the field is not implemented or that this is a non-commercial implementation.*

# Charter template

The information in this next section of the document should be considered as a template for creating a template charter in Google Doc. Information in [ALL CAPS] should be replaced appropriately. The final location for your group charter will be in the Charter.md file provided in the community riscv-admin organization repo requested in Milestone #3: Pending, Step #1.

A good Task Group (TG) charter describes how it achieves filling in a gap defined by the Special Interest Group (SIG) or Committee that spawned it (directly or dotted line). It lists the specific small set of deliverables it will deliver.

A SIG is an extension of a Committee, in that its only deliverables are strategy, gaps, and prioritizations, and helping spawn other SIGs or TGs to fill the gaps. A good SIG charter spells out the small set of topic areas their strategy will address along with its responsibilities as laid out in this bullet.

If you wish to start your charter work in a document, copy and paste the rest of this information in this section into a new document for your charter.

---

## [GROUP NAME] [GROUP TYPE, e.g. “SIG” or “TG”] Charter

```
The [GROUP NAME] [GROUP TYPE] will [OVERALL MISSION STATEMENT in
2-3
SENTENCES]

[THE NEXT PARAGRAPH IS OPTIONAL]
The [TERM 1] IS [DEFINITION 1].  [EXPLANATION  OF IMPORTANCE OF
TERM 1]. [MORE TERMS AND DEFINITIONS AS NEEDED]

[BACKGROUND INFORMATION ABOUT RELEVANCE OF GROUP/TECHNOLOGY]
```

The [GROUP NAME] [GROUP TYPE] will [DELIVER SOMETHING] [WITH THESE ATTRIBUTES]:

- [ATTRIBUTE 1]
- [ATTRIBUTE 2]
- [... AS NEEDED]

[THE FOLLOWING PARAGRAPH AND LIST ARE OPTIONAL, ESPECIALLY IF THE LIST IS EMPTY]
The following items are presently not planned to be delivered as part of this work, but may be considered in future versions:

1. [FEATURE 1]
2. [FEATURE 2]
3. [... AS NEEDED]

To achieve its goals, the [GROUP NAME] [GROUP TYPE], will interact with the following groups: [GROUP NAME 1] [GROUP TYPE 1], [GROUP NAME 2] [GROUP TYPE 2], […] and [GROUP NAME N] [GROUP TYPE N].