Caliptra Contributing Process

Caleb Whitehead, Hardware Engineer, Microsoft
Tim Trippel, Software Engineer, Google

Caliptra

SECURITY

Objectives

1. Enable broader 3rd party contributions to Caliptra project that are:
    a. Compliant with existing standards
    b. Compatible with existing features and test environments
2. Provide guardrails and requirements to filter for high-quality contributions
3. Minimize overhead for Consortium representatives in reviewing and accepting incoming proposals
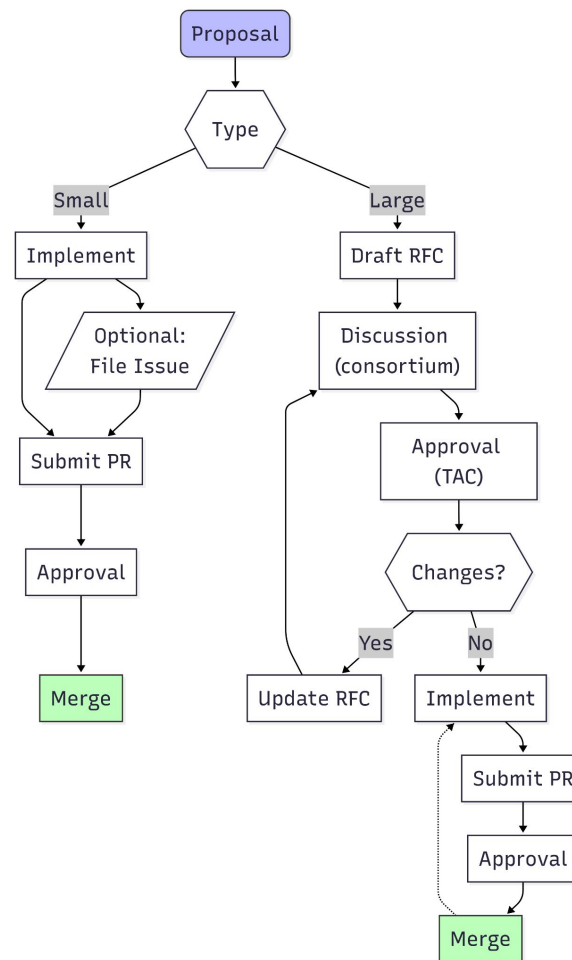
Step 1: Proposal (RFC)

Process

Two type of changes, both via GitHub:
1. **Small**
   a. [Optional]: File an issue track
   b. implement the change (via a PR)
2. **Large**
   a. draft an RFC as an issue
   b. seek discussion from consortium
   c. seek approval from TAC
   d. apply any changes requested
   e. implement the change (via several PRs)

4

What is a "Large" change?

- Architecture
  - New HW component
  - Large changes e.g. upgrade Veer from 32-bit to 64-bit
- Boot process
- API
  - Command set
  - Register layout
  - Input/Output
- Security posture
- Maintenance
  - Tools
  - Tool Versions

Structure of an RFC

- RFCs should be submitted in the form of a GitHub Issues—with the title prefix "`[RFC]:` …"— to the appropriate Caliptra repository.
- They should use the provided GitHub Issue template[1] for RFCs that contain fields for:
  - **Scope** - what parts of the project will be affected
  - **Rationale** - motivation for the change
  - **Implementation**
    - **Trade Offs** - various implementations being considered
    - **Timeline** - time estimate for completion of the work
  - **Test Plan** - plan for how HW or SW collateral will be tested
  - **Maintenance** - the individual or team that will serve as the point-of-contact for this feature now and in the future.

[1] https://github.com/chipsalliance/caliptra-rtl/blob/main/.github/ISSUE_TEMPLATE/rtl-rfc-issue-template.md

RFC Approval

- Caliptra TAC will review proposals biweekly
- Implementation permitted to begin (e.g. as RFC collateral)
    - Contributions not accepted or merged until RFC approval
- Caliptra TAC delegate will correspond with proposer and provide response on the RFC
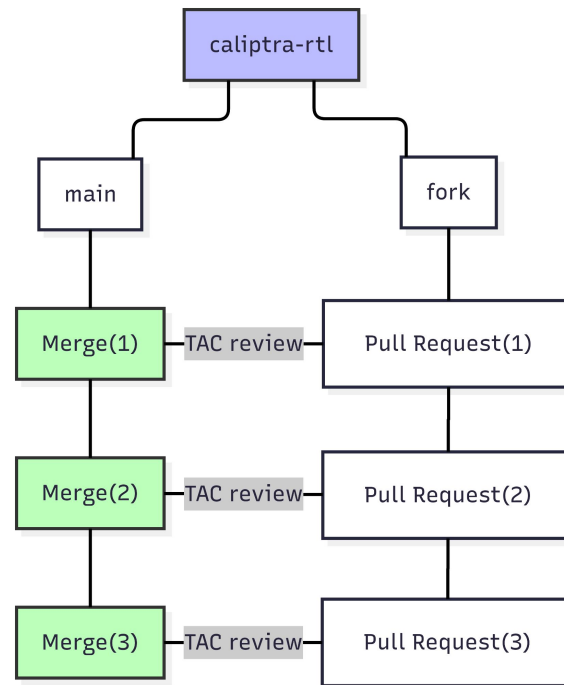
Step 2: Implementation

Implementation

- Contributions as fork and PR
- Small PR (<500 LoC) to ease review
- Development and review can be parallel

Step 3: Acceptance

Acceptance

- Consortium: review Pull Requests
- Consortium: run HW PR checklist:
    - Add new tests to regression
    - Do proprietary tool runs
    - Audit Coverage, Synthesis, CDC/RDC reports
    - RTL simulation test regression pass
    - FPGA test regressions pass
- Consortium: check for appropriate documentation
- Consortium: sign-off any security critical changes

Questions

OPEN
Compute
Project®