Project Concept Proposal

Date of proposal - 2021-06-28

Author(s) - Trefor Southwell

High Level Summary of project, project components, and deliverables

This project is the creation of the specification of the CV-X-IF interface between RISC-V cores and co-processors by the **OpenHW Cores TG**.

[CV-X-IF] (https://github.com/pulp-platform/riscv-extension-interface) is the specification of a processor-coprocessor interface applicable to RISC-V cores.

The input specification as above is written with 32-bit RISC-V cores in mind and this project will validate the specification for 64-bit cores and make modifications if required.
This project will extend the specification to specify how to support the compressed instruction format.

CV-X-IF is intended to provide a generalized framework in which custom co-processors (accelerators) provide ISA extensions for RISC-V CPU cores. It features independent channels for accelerator-agnostic offloading of instructions and writeback of the result. It supports pseudo dual-issue behaviour, and allows sharing of accelerators across multiple cores.

Test vehicles:

- The primary OpenHW target for OpenHW's implementation is the CV32E40X core, this work will gate the specification signoff.
- There are also plans to review the extension for the CVA6 core, but this implementation will not gate the work.
- There are also activities to use this interface for the RV32F extensions are on-going for the CV32E40P core, but this implementation will not gate the work.

Deliverables: Specification document submitted into GIT Hub and released to v1.0
This specification has been tested with an implementation of the interface (as part of the CV32E40X core project).

Summary of market or input requirements

Known market/project requirements at PC gate

a) Provide a unified accelerator interface.
b) Decouple development of accelerators and CPU cores
c) Re-use extension accelerators with different cores
d) Share expensive accelerator units across multiple cores in a cluster.
e) Where possible minimise the overhead (performance, area and clock frequency) of using the extension interface instead of modifying the core internally.

Nice to have: Additional specification information such as timing diagrams, architectural examples, connections, examples, etc.

Potential future enhancements

Support for multiple outstanding load/store operations

Ability to kill instructions in the accelerator

Who would make use of OpenHW output

- OpenHW Group members and other CORE-V developers who want to design CPU cores that support a range of custom accelerators (including the CV32E40X and the CVA6 projects inside OpenHW).
- Designers of RISC-V accelerators who want to be able to use them on multiple different cores.
- Any RISC-V user who wants to find optimised solutions for their application can experiment with a selection of RISC-V cores and accelerators without having to undertake major design work.

Initial Estimate of Timeline

June 2021 - V0.1 moved to GIT Hub (https://github.com/openhwgroup/core-v-xif)
April -> October 2021 - Feedback from interested parties implementation extensions based on the specification:

- Feedback from the implementation of the RV32F extension in CV32E40P using the X Interface.
- Feedback from the CVA6 security extension implementation work.
- Other interested parties review
- **Output**: v0.9 Interface document ready to start implementation of the CV32E40X

October 2021 -> June 2022

- Implementation of CV32E40X core.

- Development of formal specification of the X-Interface.
- Development of compiler for custom accelerators.

June 2022

- Release v1.0 - text specification and formal specification.

Explanation of why OpenHW should do this project

Open Hardware is member-driven, multiple members including SiLabs, Yosys, Embecosm, Thales and Imagination Technologies have expressed an interest in this project and are willing to provide resources to make it successful.

The X-Interface promotes the sharing of RISC-V accelerators across multiple cores, thus improving the overall ecosystem.

The X-Interface specification is not a RISC-V ISA specification but a hardware specification and hence is more suitable for OpenHW than RISC-V International. Any RISC-V processor hardware implementation could utilise this interface.

Industry landscape: description of competing, alternative, or related efforts in the industry

Many processor extension interfaces exist for specific cores, however this interface aims to be core independent allowing the sharing of accelerators between different RISC-V CPU cores.

OpenHW Members/Participants committed to participate

ETH - RV32F implementation in CV32E40P and subsequent feedback on the specification.
Sillabs - Implementation of the interface into CV32E40X and feedback into the specification based on this work.
Thales - Feedback on the interface when used with CVA6 (this will not gate the project).
Yosys - Formal interface specification.
Embecosm - Compiler support for custom extensions.

Project Leader(s)

Project Manager

Trefor Southwell, Imagination Technologies

Next steps/Investigation towards Project Launch (PC only)

Prepare project launch documents including a project plan, list of deliverables/releases and review within the X-Interface working group.

Define how the formal specification will be used.

Agree who will be responsible for maintaining the specification document during the 0.1-0.9 development phase.

Target Date for PL

End of July 2021