



# OpenHW Group

Proven Processor IP

**CORE-V-VERIF:**

**An Industrial-Grade Verification Platform  
for RISC-V Cores**

Mike Thompson

[mike@openhwgroup.org](mailto:mike@openhwgroup.org)



**OPENHW** GROUP  
PROVEN PROCESSOR IP

© OpenHW Group

DV Club Europe 2022-11-29



**OPENHW**<sup>GROUP</sup><sup>TM</sup> and  
— PROVEN PROCESSOR IP —



**CORE-V**<sup>TM</sup>



- OpenHW Group is a not-for-profit, global organization driven by its members and individual contributors where HW and SW designers collaborate in the development of open-source cores, related IP, tools and SW such as the **CORE-V** Family of open-source RISC-V cores.
  - International footprint with developers in North America, Europe and Asia.
  - Providing an infrastructure for hosting high quality open-source HW developments in line with industry best practices.
  - Strong support from industry, academia and individual contributors worldwide.



**OPENHW**<sup>GROUP</sup><sup>TM</sup>  
— PROVEN PROCESSOR IP —

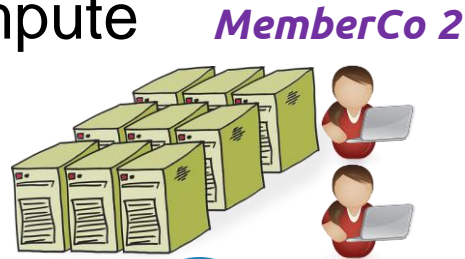


**OPENHW**™  
GROUP  
— PROVEN PROCESSOR IP —

# Industry Ecosystem 90+ Members & Partners



Member companies provide human, compute and tool resources.



**GitHub**

*MemberCo 1*

Open source code base administered on GitHub.

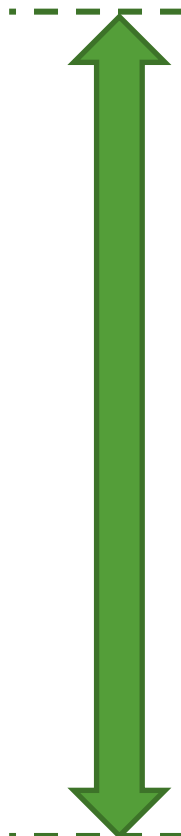


**OPENHW**™  
GROUP  
— PROVEN PROCESSOR IP —

# Technology Readiness Levels



IDEA	TRL-0	Idea Unproven product/technology idea
	TRL-1	Basic research Basic principles observed
	TRL-2	Concept formulation Potential application found and validated. Basic principles have been studied and practical applications identified
R&D	TRL-3	Proof-of-Concept A crude POC prototype is crafted, demonstrating the core technology (principle) feasibility
	TRL-4	Component and/or Breadboard Lab prototype Start of engineering R&D: multiple component and subsystems are tested in lab environment.
DEVELOPMENT	TRL-5	Subsystems designed and tested in a real life Validation of the subsystems and engineering units with rigorous testing in relevant (real life) environment
	TRL-6	Functional prototype system (alpha prototype) Integration of the previously designed sub-systems into a functional ALPHA prototype with first tests
	TRL-7	"Field" demonstration prototype system Working model or prototype (still ALPHA) demonstrated in relevant environment
PRODUCTION	TRL-8	BETA prototype (commercial ready system) A "flight qualified design" embracing DFM approach. Batch production launched and product is being implemented for the intended purpose
	TRL-9	Commercial application. Successful mission Mass-production. Product/technology is available to all customers



*OpenHW  
Technology  
Outputs*



*OpenHW IP  
Adopters*



# Verification Task Group Mandate



- From the [CORE-V Verification Strategy](#):
  - *The OpenHW Group will, together with its Member Companies, execute a complete, industrial grade pre-silicon verification of the first generation of CORE-V IP, the CV32E4 and CVA6 cores. Experience has shown that “complete” verification requires the application of both dynamic (simulation, FPGA prototyping, emulation) and static (formal) verification techniques. All of these techniques will be applied to both CV32E4 and CVA6.*

## Some big statements made here:

- Complete, industrial grade pre-silicon verification.
- Simulation, Formal and FPGA prototyping.

A screenshot of a web browser displaying the "OpenHW Group CORE-V Verification Strategy" document. The browser address bar shows the URL: https://core-v-docs-verif-strat.readthedocs.io/en/latest/. The page header includes the OpenHW Group logo and the text "CORE-V Verification Strategy". Below the header is a search bar labeled "Search docs". The main content area is titled "OpenHW Group CORE-V Verification Strategy" and lists the editor as Michael Thompson (mike@openhwgroup.org). A "Contents:" section is visible, listing various topics such as "Introduction", "License", "CORE-V Projects", "Definition of Terms", "Conventions Used in this Document", "CORE-V Genealogy", "A Note About EDA Tools", "Verification Planning and Requirements", "Base Instruction Set", "Privileged Spec", "XPULP Instruction Extensions", "Custom Circuitry", "Interrupts", "Debug", and "RVI-Compliant Interface". A sidebar on the left contains a "CONTENTS:" section with a list of document sections: Introduction, Verification Planning and Requirements, PULP-Platform Simulation Verification, CORE-V Verification Environment, CV32E40P Simulation Testbench and Environment, CV6A Simulation Testbench and Environment, Simulation Tests in the UVM Environments, Test Program Environment (BSP), CORE-V Formal Verification, and CORE-V FPGA Prototyping.



# Open Source Implementations

...but not necessarily open source tools



- In order to achieve “complete, industrial grade verification”, industrial quality tools are used.
- CORE-V-VERIF UVM Verification verification environment runs on all of the major commercial SystemVerilog simulators



# Verification Planning



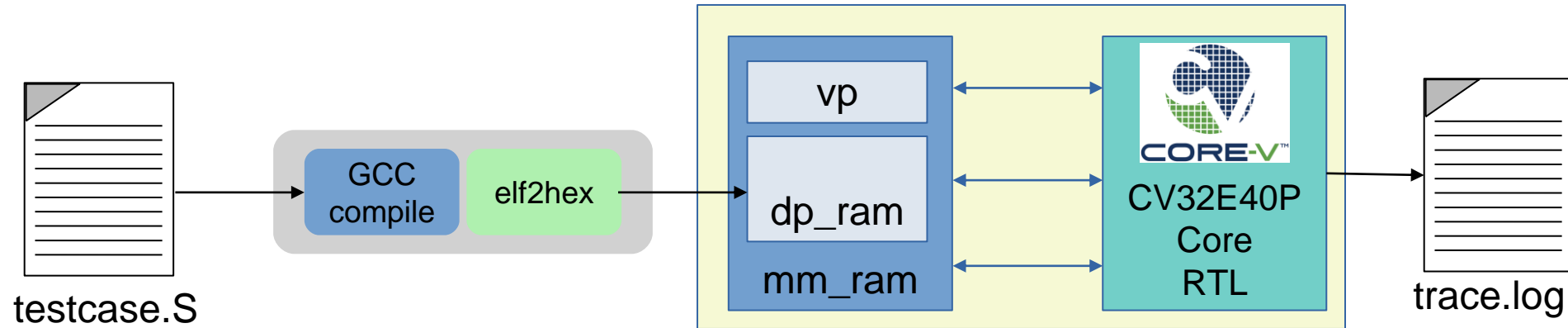
Requirement Location	Feature	Sub Feature	Description	Verification Goal	Pass/Fail Criteria	Test Type	Coverage Method
ISA Chapter 2	RV32I, Instruction non-specific			Coverage that need not be crossed with any specific RV32 instruction.	Add coverage to ensure toggles of all bits on all GPRs and all bits of immediates.	Compliance tests: correct test signature. Random tests: RTL matches ISS.	Non-test-specific Functional coverage of GPR and immediate values. Note that this could also be done with code-coverage, but this will be micro-arch specific.
ISA Chapter 2.4	RV32I Register-Immediate Instructions	ADDI	addi rd, rs1, imm[11:0] rd = rs1 + Sext(imm[11:0]) Arithmetic overflow is lost and ignored	Exercise instruction using all combinations of source and destination operands.  Exercise overflow and underflow.	Compliance tests: correct test signature. Random tests: RTL matches ISS.	Compliance / Random	Test case (Compliance). Functional coverage of verification goals, with special attention to <b>NOP</b> .
		SLTI	slti rd, rs1, imm[11:0] rd = (rs1 < Sext(imm[11:0])) ? 1 : 0 Both imm and rs1 treated as <b>signed</b> numbers	Exercise instruction using all combinations of source and destination operands.	Compliance tests: correct test signature. Random tests: RTL matches ISS.	Compliance / Random	Test case (Compliance). Functional coverage of verification goals.
		SLTUI	sltui rd, rs1, imm[11:0] rd = (rs1 < Sext(imm[11:0])) ? 1 : 0	Exercise instruction using all combinations of source and	Compliance tests: correct test signature.	Compliance / Random	Test case (Compliance). Functional coverage of

Hierarchically organized:  
- one per High-level Feature

- All CORE-V-VERIF DV Plans are open-source artifacts:
  - ...both formal and simulation



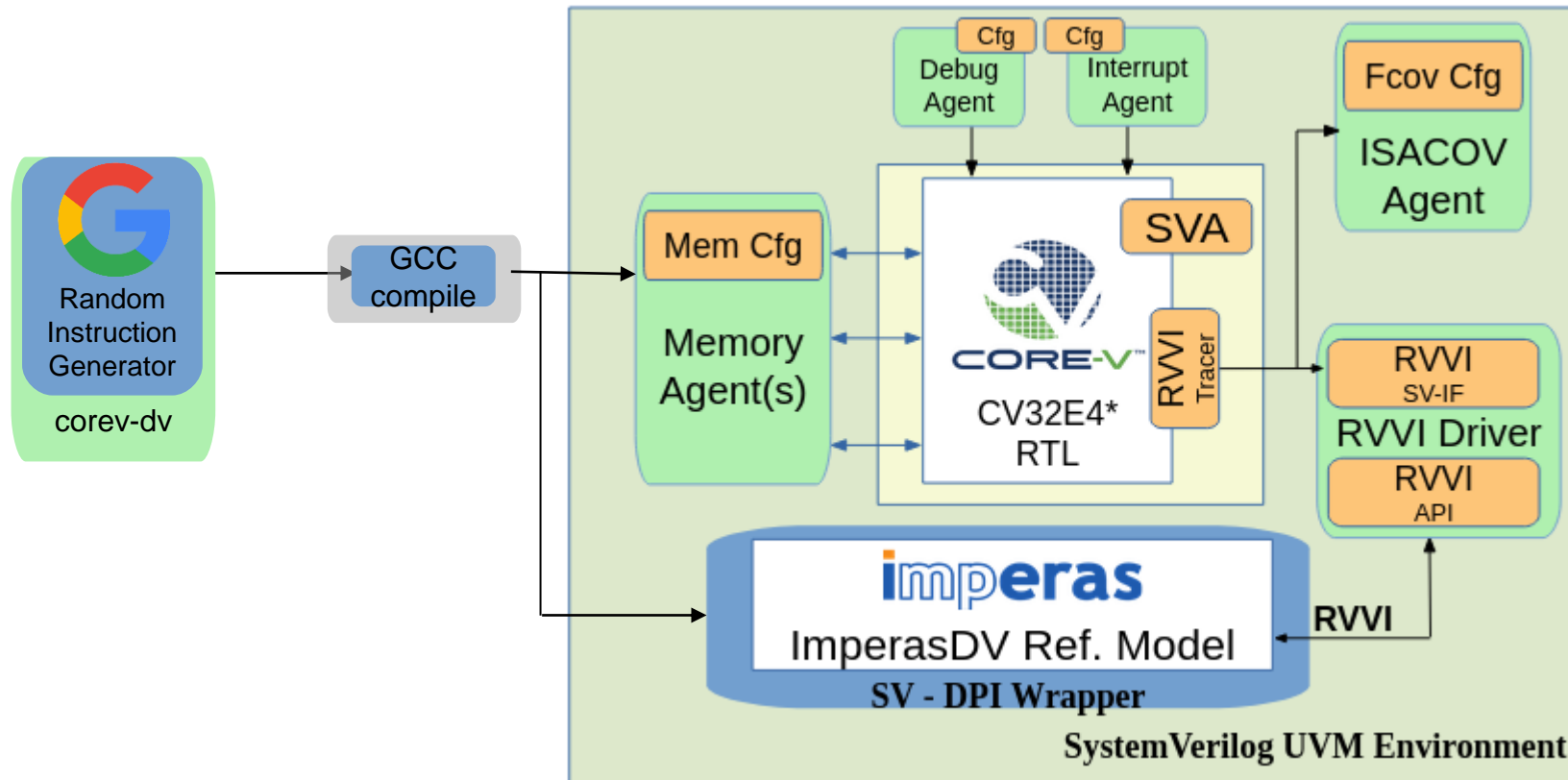
# “CORE” Testbench



- CORE Testbench capabilities:
  - Demonstration platform
  - RISC-V Compliance



# CORE-V-VERIF UVM Environment




- UVM Environment capabilities:
  - All core testbench capabilities, plus
  - 100% code and functional coverage.

# Is this approach to verification necessary?

- It depends on your goals:
  - For TRL-3 maybe the CORE TB is sufficient.
  - For TRL-5 the bar is higher.
- Released versions of OpenHW IP:
  - Open source, easy to access User Manual on ReadTheDocs.
  - Well structured RTL implementations.
  - Complete verification environment capable of achieving 100% coverage.



🏠 CORE-V CV32E40P User Manual



latest

**CONTENTS:**

- Introduction
- Getting Started with CV32E40P
- Core Integration
- Pipeline Details
- Instruction Fetch
- Load-Store-Unit (LSU)
- Register File
- Auxiliary Processing Unit (APU)
- Floating Point Unit (FPU)
- Sleep Unit
- PULP Hardware Loop Extensions
- Control and Status Registers
- Performance Counters
- Exceptions and Interrupts
- Debug & Trigger
- Tracer
- PULP Instruction Set Extensions
- Glossary

```
function countOccurrences (
  str, char) {
  let count = 0;
  for (let i = 0; i < str.length; i++) {
    if (str[i] === char) count++;
  }
  return count;
}
```

**TAKE THE QUIZ!**

Beat Triplebyte's online coding quiz. Get offers from top companies. Skip resumes & recruiters.

Sponsored - Ads served ethically

Read the Docs v: latest

## OpenHW Group CV32E40P User Manual

Editor: Davide Schiavone [davide@openhwgroup.org](mailto:davide@openhwgroup.org)

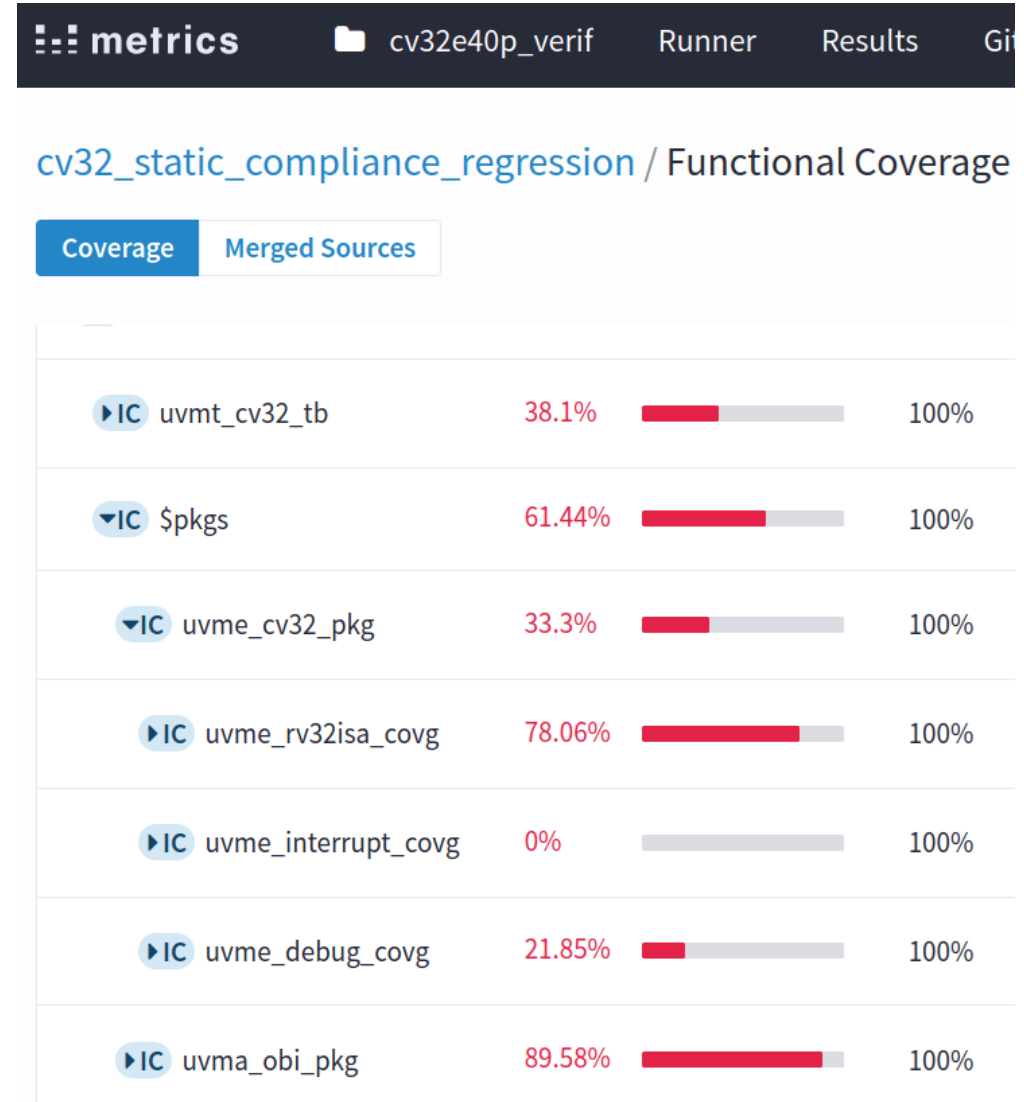
### Contents:

- Introduction
  - License
  - Standards Compliance
  - ASIC Synthesis
  - FPGA Synthesis
  - Verification
  - Contents
  - History
  - References
  - Contributors
- Getting Started with CV32E40P
  - Register File
  - Clock Gating Cell
- Core Integration
  - Instantiation Template
  - Parameters
  - Interfaces
- Pipeline Details
  - Multi- and Single-Cycle Instructions
  - Hazards
- Instruction Fetch
  - Misaligned Accesses
  - Protocol
- Load-Store-Unit (LSU)
  - Misaligned Accesses
  - Protocol
  - Post-Incrementing Load and Store Instructions
- Register File
  - Flip-Flop-Based Register File
  - Latch-based Register File
  - FPU Register File
- Auxiliary Processing Unit (APU)
  - Auxiliary Processing Unit Interface
  - Protocol
  - Connection with the FPU
  - APU Tracer
  - Output file

# Why not just run Compliance?



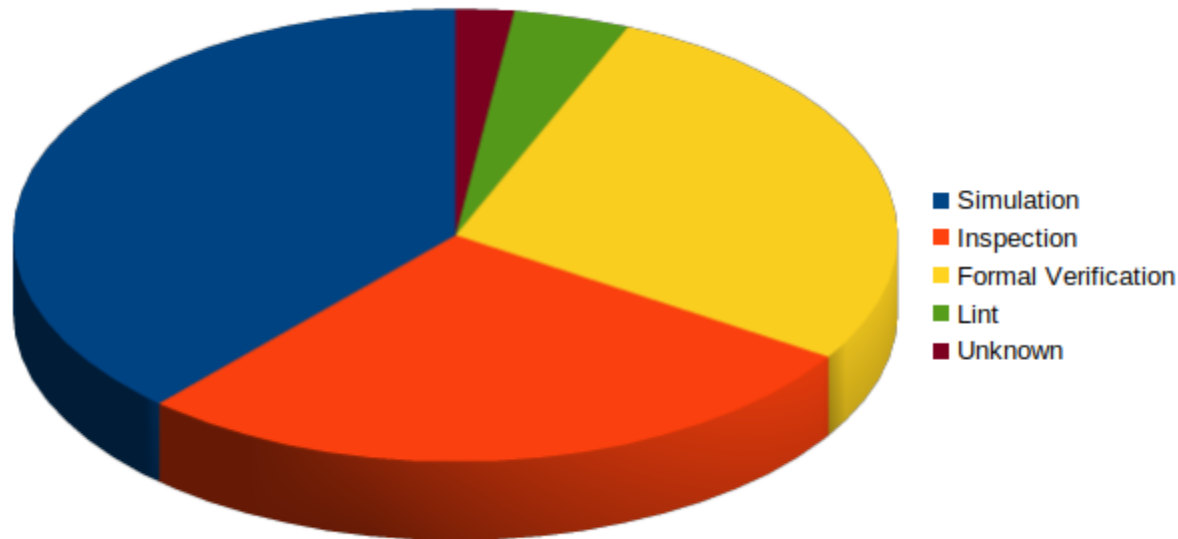
- The RISC-V ISA Compliance Test-Suite is 'necessary but insufficient' for complete verification.
- In order to cover all Features in the DV plans, a much more comprehensive verification effort is required:
  - Goals defined by DVplans.
  - Progress measured by coverage.
- None of the 47 bugs we found were triggered by compliance testing.



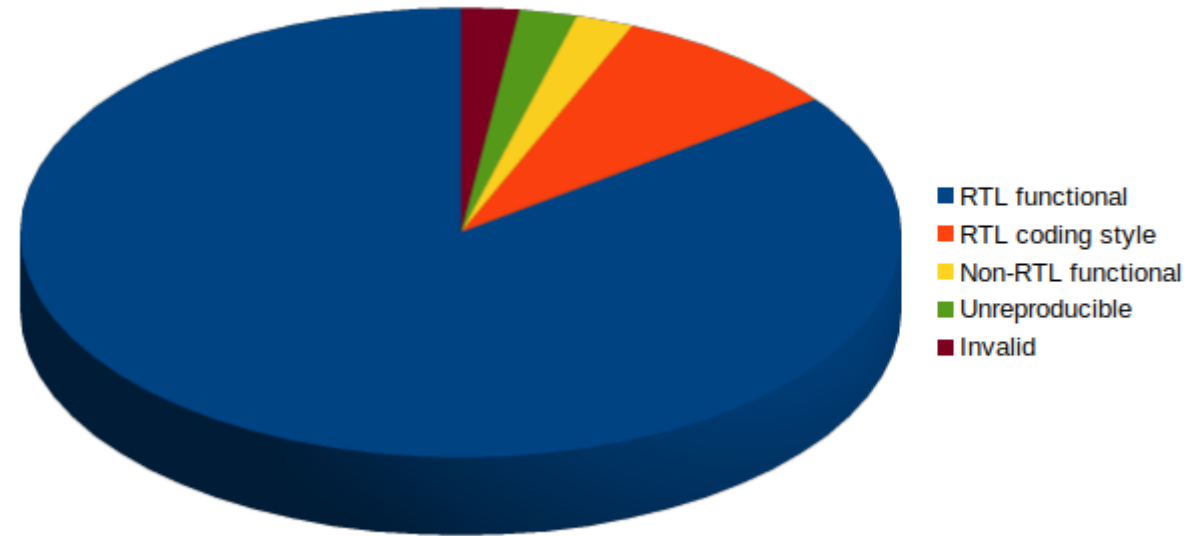
# An Example: CV32E40P Verification Results



- The CV32E40P core started its life as the RI5CY core:
  - Known to be high-quality open source implementation.
  - Implemented in silicon multiple times.
- 47 RTL bugs were found.



How Bugs Were Found



Types of Bugs Found



**OPENHW**<sup>GROUP</sup><sup>TM</sup> and  
— PROVEN PROCESSOR IP —



**CORE-V**<sup>TM</sup>



- OpenHW Group is a not-for-profit, member driven organization collaboratively developing of the **CORE-V** Family of open-source RISC-V cores.
- We apply Industrial practises to achieve industrial quality results.

- Find us on GitHub!

**OpenHW Group**  
 163 followers · Ottawa, Ontario, Canada · <http://www.openhwgroup.org> · @openhwgroup · info@openhwgroup.org

Overview · Repositories 46 · Projects 3 · Packages · Teams 11 · People 110 · Settings

Popular repositories

- cva6** (Public) · SystemVerilog · 1.6k stars · 457 forks
- cv32e40p** (Public) · SystemVerilog · 652 stars · 288 forks
- core-v-verif** (Public) · Assembly · 235 stars · 136 forks
- cvfpu** (Public) · SystemVerilog · 228 stars · 77 forks
- force-riscv** (Public) · C++ · 141 stars · 36 forks
- programs** (Public) · Makefile · 134 stars · 69 forks

View as: **Public** ▾  
 You are viewing the README and repositories as a public user.  
 You can [create a README file](#) or [pin repositories](#) visible to anyone.  
 Get started with tasks that most successful organizations complete.

People  
 View all  
 Invite someone



**OPENHW**<sup>GROUP</sup><sup>TM</sup>  
— PROVEN PROCESSOR IP —

# Thank You!