RISC-V Reuse Made Easy by Interface Generation and Integration Automation

Ares Tahiraga, Said Llukaj, Wei Zhao, Endri Kaja, Sebastian Prebeck, Wolfgang Ecker

Infineon Technologies AG

accellera SYSTEMS INITIATIVE

2026 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES
SANTA CLARA, CA, USA
MARCH 2 - 5, 2026

INTRODUCTION

IP reuse brought hardware design productivity a big step forward by simplifying reuse. Unfortunately, it is not as simple as it seems: each IP must be understood, configured and then connected with the signals of the SoC. Despite de-facto standards for on-chip busses, scan signals, etc., integration is challenging in one major way: flexible interfaces such as APB, AHB, and AXI from the ARM family allow for variations in implementation, offer numerous configuration choices, and can be often tailored to enhance performance, power, area (PPA), or to address specific safety and security needs.

For RISC-V, functional configuration is relatively easy due to its modular ISA and profiles that offer pre-configured feature sets. However, the integration of RISC-V IPs remains challenging due to the fact that each RISC-V IP comes with its own communication and infrastructure demand.

OBJECTIVES

To tackle these challenges, we have developed generic modules to connect the RISC-V to its external world. Since modern Hardware Description Languages (HDLs) like SystemVerilog lack support for generic programming concepts, such as optional ports or automated object feature propagation, we have developed a Python-based code generation framework, called MetaRTL, to provide these features as a generation step for HDLs.

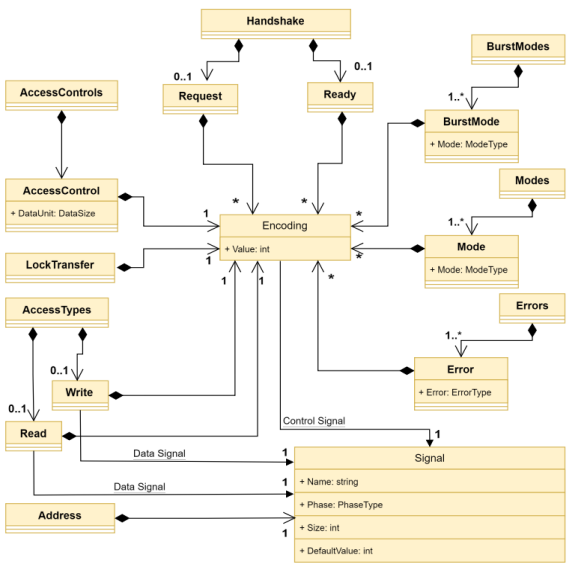
This generation framework is not only used to generate the RISC-V IP but also to generate the interface connections of the RISC-V to the SoC using introspection techniques also not supported by HDLs and HGLs. The goal is to create a framework able to generate different variants of CPU cores and to speed up the design cycles via automatic IP integration.

CONCEPT

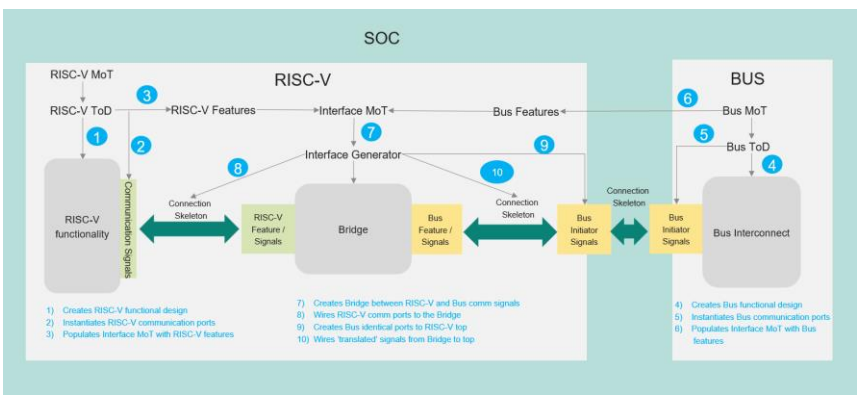
Two key observations:

1. Common bus protocols share a unified semantic model

2. CPU's communication signals implement a protocol of its own, conform to this model



RESULTS



Faster Cycles for:

1. Testing and regression
2. Benchmarking
3. Design Space Exploration
4. Product integration

CONCLUSIONS

We presented a model-driven, feature-oriented Bus Interface Generator that decouples RISC-V IP functionality from interconnect communication, enabling automated, single-pass generation of protocol-compliant interfaces across APB/AHB/AXI. This approach enables scalable exploration, benchmarking, and faster product integration. The method generalizes to other IP classes (e.g., DMA, accelerators, DSPs).

Future work will extend protocol coverage and add automated instrumentation for quantitative PPA and timing profiling. Overall, systematic interface generation makes plug-and-play RISC-V reuse practical and maintainable at scale.

REFERENCES

- Schreiner, J., Gerl, D., Kunzelmann, R., Sinha, P. K., & Ecker, W. (2024). The Argument for Meta-Modeling-Based Approaches to Hardware Generation Languages. arXiv preprint arXiv:2404.05599
- J. Schreiner, V. R. Gontia, S. Prebeck and W. Ecker, "Generator IP-reuse and Automated Infrastructure Generation for Model-based Full Chip Generation," MBMV 2023; 26th Workshop, Freiburg, 2023, pp. 1-12
- Truyen, F.: The Fast Guide to Model Driven Architecture [online]. Available: https://www.omg.org. [Accessed: 12-01-2026]
- RISC-V International, "The RISC-V Instruction Set Manual, Volume I: Unprivileged ISA," Version 20250508, 2025. Available at: riscv.org
- ARM, "AMBA APB Protocol Specification", ARM Limited, 2023. [Online]. Available at: https://developer.arm.com/documentation/ihi0024/latest/ [Accessed: 27-10-2025]
- ARM, "AMBA AHB Protocol Specification", ARM Limited, 2021. [Online]. Available at: https://developer.arm.com/documentation/ihi0033/latest/ [Accessed: 27-10-2025]
- ARM, "AMBA AXI Protocol Specification", ARM Limited, 2025. [Online]. Available at: https://developer.arm.com/documentation/ihi0022/latest/ [Accessed: 27-10-2025

infineon

This work was partly funded by the German Federal Ministry of Education and Research (BMBF) and EU Horizon-KDT within the project **ISOLDE** under contract number 16MEE03

ISOLDE RISC-V