



A Comparative Design and Analysis of RISC-V Core

¹ Dr. Arokia Priya Charles, ² Abhiram Deshmukh,

¹Head of Department, Department of Semiconductor Engineering, D Y Patil International University, Pune, India

²Student, Department of Electronics and Telecommunication, Dr. D Y Patil Institute of Engineering, Management and Research, Pune, India

Email - ¹priya.charles@dypiu.ac.in ²abhiramm004@gmail.com

Abstract: *This work presents a comparative design and analysis of two RISC-V processor architectures: a single-cycle RV32I core and a five-stage pipelined core. The single-cycle design establishes a baseline implementation, while the pipelined version integrates instruction fetch, decode, execute, memory, and write-back stages along with hazard detection, data forwarding, and branch resolution mechanisms. Both processors are described in RTL and synthesized on an FPGA platform using a unified workflow based on open-source EDA tools, ensuring accessibility and reproducibility of results. Performance is assessed using the Embench-IoT benchmark suite, which provides realistic workloads representative of embedded and IoT domains. Key metrics such as Cycles per Instruction (CPI), maximum achievable clock frequency, FPGA resource consumption (LUTs, flip-flops, and memory blocks), and relative design complexity are measured. The analysis highlights the quantitative trade-offs between architectural simplicity and throughput, while also demonstrating the viability of open-source tool chains for advanced processor design and verification. This work thus provides practical insight into processor architecture evaluation and encourages the adoption of open-source methodologies in resource-constrained embedded system design.*

Key Words: *RISC-V, Open-source, Embench-IoT, Verilog HDL, Embench-IoT.*

1. INTRODUCTION:

The RISC-V ISA has quickly become one of the leading open standards in processor design due to its modular structure, extensibility, and the fact that it is license-free. Unlike proprietary ISAs, RISC-V allows both academic and industrial personals to innovate in a wide range of computing domains-from minimalist embedded processors to complex application-class cores. Early frameworks, like the Rocket Chip generator, showed how to realize parameterized architectures within open hardware ecosystems, while lightweight implementations such as Ibex and SERV have demonstrated the adaptability of the ISA in low-power and educational settings, respectively [5, 6].

A major driver behind RISC-V's broad adoption is the maturation of its open-source electronic design automation toolchain. Projects such as Yosys for logic synthesis, Verilator for simulation, and OpenROAD or OpenLane for physical design have formed a complete end-to-end open workflow. These tools bring reproducibility, transparency, and accessibility that allow for comparative architectural research without using proprietary environments. Simultaneously, cloud-based development interfaces like Makerchip and EDA Playground have further democratized processor design and verification by eliminating much of the computational and licensing complexity.

Various recent works have shown that indeed it is possible to implement scalable RISC-V pipelines in completely open or at least freely available ecosystems. Ahmed and Rashid [2] implemented and verified a five-stage pipelined RV32I core, which was described in SystemVerilog, simulated via EDA Playground, and synthesized via the Intel Quartus Prime FPGA flow. Independently, Zhang et al. [1] implemented a three-stage pipelined RISC-V processor, whose verification was performed using Icarus Verilog, while its FPGA hardware implementation operated stably at 50 MHz. Ahmed [3] also presented a five-stage pipelined RISC-V implementation, with emphasis on reproducibility, accessibility, and welldefined hazard-handling mechanisms. All these works collectively indicate that open-source and academic toolchains are capable of producing verifiable high frequency FPGA designs that can match those developed using commercial toolchains.

Benchmarking represents another equally important element for validation of such designs. Traditional suites such as Dhrystone or CoreMark often fail to capture representative workloads for embedded applications. The Embench-IoT benchmark suite addresses this limitation by offering a set of representative compute kernels-FFT, CRC, matrix



multiplication-tailored for constrained IoT platforms [6]. Automated testing infrastructures such as Antmicro's Embench tester allow for consistent cross-core testing in open workflows [6]. While there is an increasing number of RISC-V implementations, few studies perform a head-to-head comparison among single-cycle and pipelined RV32I cores within the same open-source toolchain. Most analyses target heterogeneous designs across ISA subsets or optimization objectives. This gap begs for open, reproducible, end-to-end comparative frameworks that integrate design, verification, and benchmarking into transparent EDA environments. The coincidence of open-source ISAs and open-source EDA ecosystems enables complete research pipelines—from RTL creation through synthesis and benchmarking—to be entirely performed with community-driven tools, ensuring accessibility, verifiability, and long-term sustainability in processor design research.

2. LITERATURE REVIEW:

The RISC-V ISA evolution has redefined processor design by finally offering an open, modular, license-free alternative to traditional proprietary ISAs. It further enables the implementation of processors in the wide spectrum from educational FPGA prototypes up to sophisticated high-performance system-on-chips. This openness has fostered a substantial body of scholarly and industrial work on architecture design, verification, and benchmarking using open-source electronic design automation tools.

Eljhani et al. [9] designed a five-stage pipelined RISC processor on an FPGA to explain the advantages of instruction pipelining for latency reduction and increasing throughput. The processor was realized using Verilog HDL on the Intel Cyclone IV FPGA, executing a 16-bit instruction set in the stages IF, ID, EX, MEM, and WB sequentially. This design presents improvement in instruction execution rate with low resource utilization. During the study, data hazards are documented, which were mitigated using load-use handling and hazard detection mechanisms. This proved the educational value and performance potential of the pipeline.

The implementation of the 32-bit RV32I RISC-V processor in Verilog HDL and its verification in ModelSim are presented by Jin-Yang et al. [10]. It describes a five-stage pipeline architecture that implements 13 basic instructions of the RISC-V processor, including ADD, SUB, XOR, LW, SW, and JAL. The simulation results indicate proper instruction sequencing and pipeline working; the instruction addresses are incremented by four every clock cycle. This paper again highlights the potential for RISC-V in low-power IoT and embedded applications by referring to its simplicity and modularity.

Saiprathyusha and Chandrasekhar [11] presented the optimized pipelined RISC-V design in pursuit of high throughput with low power consumption for applications related to both artificial intelligence and signal processing. According to the authors, a modular datapath consists of an ALU, an ALU decoder, a main decoder, instruction and data memories, and pipeline registers, which have been verified by using Verilog simulation and synthesized on an FPGA. The methodology also encompasses design optimizations for cache access, dynamic scheduling, and data forwarding, reflecting modern trends in research toward high-efficiency soft-core architectures.

Earlier comparative studies have explored various RISC-V implementations to evaluate trade-offs among throughput, resource utilization, and power efficiency. Zhang et al. [1] and Ahmed et al. [2] proposed educational, FPGA-based, three- and five-stage pipelined RV32I cores and showed that pipelining greatly improves instruction throughput while keeping hardware complexity at a reasonable level. In contrast, Heinz et al. [4] carried out an in-depth review of open-source RISC-V cores using FPGA-based benchmarks and highlighted the role of reproducible, open workflows. Their results show that architectural decisions, like pipeline depth and hazard mitigation, have a direct impact on area efficiency and energy consumption.

Benchmarking and validation frameworks have similarly developed alongside this architectural advancement. The Embench-IoT benchmark suite [6] presents workloads representative of embedded and IoT processors, overcoming some of the shortcomings of previous synthetic tests like Dhrystone and CoreMark. Work such as Fu et al. [8] used Embench-IoT to investigate the energy efficiency of in-order and superscalar out-of-order RISC-V cores and demonstrated that their performance–area trade-off rises significantly with a deeper pipeline and increasingly sophisticated control logic. Open-source EDA environments like GreenRio [7], along with toolchains including Yosys, Verilator, and OpenROAD, further facilitate reproducible research in processors. These frameworks eliminate proprietary licensing barriers that allow researchers to design, verify, and synthesize full processors in open environments. Their availability has considerably contributed to various academic research and educational applications, informing studies such as the present work.



Table 1: Summary of RISC-V implementations relevant to this review.

Reference	Work Description	Toolchain Used	Target Hardware	Key Findings
[1] Zhang <i>et al.</i>	Designed a 3-stage pipelined RISC-V processor verified using Icarus Verilog	Icarus Verilog + Yosys	FPGA (Custom board)	Demonstrated stable 50 MHz operation and simple hazard-free pipeline.
[2] Ahmed & Harun-Ur-Rashid	Implemented and verified a 5-stage pipelined RV32I core with hazard detection and forwarding.	EDA Playground + Quartus Prime	Intel Cyclone IV FPGA	Improved throughput; validated standardized pipeline stages.
[3] Ahmed	FPGA-based implementation of a 5-stage RV32I RISC-V processor emphasizing reproducibility.	Verilog HDL + Quartus	Intel Cyclone IV	Highlighted performance-complexity trade-offs in open workflows.
[4] Heinz <i>et al.</i>	Comparative catalog of open-source RISC-V cores.	Open-source benchmarking framework	FPGA (Various)	Identified reproducibility issues; analyzed performance vs. area.
[5] Dörflinger & Albers	Surveyed open-source application-class RISC-V projects.	-	-	Discussed design trade-offs, emphasizing reproducible workflows.
[6] Patterson <i>et al.</i>	Introduced Embench-IoT benchmark suite for embedded systems.	GCC + Python framework	-	Provided realistic IoT workloads beyond CoreMark and Dhrystone.
[7] Bertacco <i>et al.</i>	Proposed the GreenRio open-source educational EDA framework.	Yosys + Verilator + OpenROAD	-	Demonstrated feasibility of academic open-source design flows.
[8] Fu <i>et al.</i>	Evaluated superscalar and out-of-order RISC-V cores.	Open-source ASIC flow (GF22FDX)	ASIC	Quantified power-performance trade-offs using Embench-IoT.
[9] Eljhani <i>et al.</i>	Implemented a 5-stage pipelined RISC-V processor on FPGA.	Verilog HDL + ModelSim	Cyclone IV FPGA	Improved execution rate and resource utilization; educational pipeline design.
[10] Jin-Yang <i>et al.</i>	Implemented a 32-bit RV32I RISC-V processor in Verilog HDL.	ModelSim	FPGA	Validated correct instruction sequencing and pipeline operation.
[11] Saiprathyusha & Chandrasekhar	Optimized pipelined RISC-V design for AI and signal processing workloads.	Verilog HDL + Quartus	FPGA	Proposed modular design with efficient cache access and forwarding logic.



In summary, prior works collectively demonstrate that pipelining and open-source RISC-V implementations significantly improve performance for educational and embedded domains. However, a critical research gap persists: few studies directly compare single-cycle and pipelined RISC-V RV32I cores within a unified, open-source workflow and benchmarked using Embench-IoT. This gap motivates the proposed research, which aims to systematically evaluate both architectures using identical design conditions and benchmarking tools to provide a transparent and reproducible comparative analysis.

3. PROPOSED FUTURE WORK :

The following research agenda proposes to expand this comparative analysis by devising a more comprehensive testing framework for both single-cycle and pipelined RISC-V RV32I cores in the same open-source toolchain. To this end, the execution of Embench-IoT benchmarks on both will be carried out during the upcoming phases to obtain measurable data regarding performance (cycles per instruction, latency), resource utilization, and maximum attainable clock frequency. The experimental design will be expanded to see how pipeline depth affects real embedded workloads.

In addition, power and energy efficiency for both architectures will be emphasized in this study through post-synthesis analysis and FPGA power-monitoring tools. This testbed will establish the relationship between the throughput and energy efficiency gains. The analysis will further consider whether additional logic overhead due to pipeline registers and hazard-detection circuitry is justified by the observed performance improvement in IoT-class applications.

In order to ensure that this process is transparent and reproducible, it will be automated from design to benchmarking within a unified open-source workflow, combining tools such as Yosys, Verilator, and visualization of results in Python. This framework will act as a unified reference for future comparative queries targeting open-source RISC-V architectures and will give concrete insights into choosing suitable core architectures for embedded and low-power computing platforms.

4. METHODOLOGY :

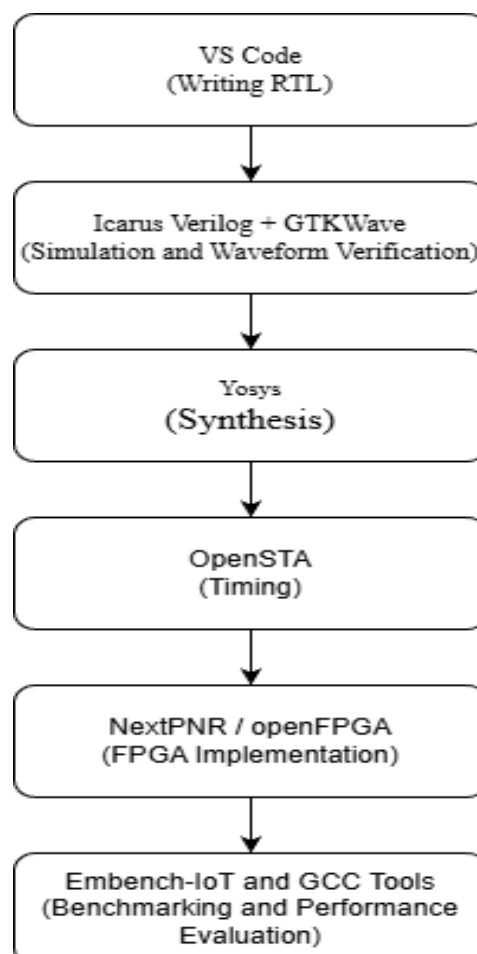


Figure 1: Open-source toolchain workflow



The proposed methodology outlines a step-by-step process for the implementation, verification, and comparative analysis of two RISC-V processor architectures, namely a single-cycle and a five-stage pipelined design, using open-source design and verification tools.

The first step is the RTL design for both architectures in Verilog HDL. First, the single-cycle processor, which contains all core parts like PC, Instruction Memory, Control Unit, ALU, Register File, and Data Memory, is implemented as a basic design. Then, after verification, a five-stage pipelined processor is developed by adding pipeline registers between the stages: IF, ID, EX, MEM, and WB. Further, the design has been enhanced with more modules that include hazard detection and forwarding units for data and control hazards. Simulation will be performed by using Icarus Verilog; waveforms were analyzed in GTKWave, assuring that all instructions are executed correctly and within the correct timing.

After successful simulation, both designs have been synthesised with Yosys to determine logic cell utilization, area, and timing characteristics. Reports from post-synthesis allow quantitative comparisons between the single-cycle and pipelined cores. Optionally, synthesized designs can be implemented on FPGA hardware-a platform such as Basys-3 or DE2-115-to validate real-time functionality and confirm timing performance and maximum frequency of operation achievable by both cores.

After verification, both cores run the Embench-IoT benchmark suite compiled using the RISC-V GCC toolchain. Benchmarks are executed on each processor to measure a set of essential performance metrics, such as CPI, execution latency, throughput, and resource utilization. Results will be visualized and analyzed to explore the trade-offs in performance, area efficiency, and design complexity.

5. CONCLUSION :

This review offers a broad survey of existing RISC-V implementations, underlining the new importance of open-source design methodologies in processor research and education. The works reviewed here show that RISC-V's modular and open instruction set architecture allows efficient processor architectures to be designed-from basic educational prototypes to sophisticated pipelined and superscalar cores-using fully open and reproducible flows. Integration of tools such as Yosys, Verilator, and OpenROAD is an indication of the maturity achieved by the open-source EDA ecosystem, which is now supporting end-to-end design, simulation, and synthesis at a level comparable to commercial environments. The trade-off between architectural simplicity and throughput efficiency is a recurrent theme throughout the reviewed literature. Single-cycle cores minimize hardware overhead by reducing control logic. As such, they are good targets for implementation in an embedded system or for educational purposes. Pipelining can enhance performance by exploiting instruction-level parallelism. However, this introduces several challenges: hazard detection, data forwarding, and control dependency management. Benchmarking undertaken using Embench-IoT and other lightweight performance suites has demonstrated that pipelining indeed improves throughput, but at the cost of additional resources and higher power consumption.

It is clear from the synthesis of prior work that few, if any, works have performed a direct comparison of single-cycle and pipelined RISC-V cores within one open-source toolchain. This research will investigate this gap by providing an implementation for both cores on an integrated flow-from Verilog-based RTL design to benchmarking-entirely done using open-source tools. The results from this in-progress investigation are foreseen to provide reproducible insights into the ways in which pipeline depth impacts performance, area, and energy metrics in embedded-class processors.

By covering this methodological gap, the work hereby presented enriches the scientific knowledge on RISC-V microarchitectural design and fosters the greater goal of open, transparent, and sustainable hardware development. The results obtained in this comparative study are expected to provide a valuable guide for educators, researchers, and practitioners interested in open-source processor development in both academic and industrial contexts.

REFERENCES:

1. W. Zhang, Y. Zhang, and K. Zhao, "Design and Verification of Three-Stage Pipeline CPU Based on RISC-V Architecture," *Proc. IEEE 5th Asian Conf. on Artificial Intelligence Technology (ACAIT)*, 2021.
2. S. Ahmed and A. B. M. Harun-Ur-Rashid, "Design, Implementation and Verification of Five-Stage Pipeline RISC-V Core (RV32I ISA)," *Proc. IEEE Int. Conf. on Electrical, Computer and Communication Engineering (ECCE)*, 2025.
3. S. Ahmed, "Design and FPGA Implementation of Five-Stage Pipelined RISC-V Processor," *IEEE Xplore Digital Library*, 2025.
4. C. Heinz, A. Herzog, M. Wenzl, and A. Herkersdorf, "A Catalog and In-Hardware Evaluation of Open-Source Drop-in RISC-V Cores," *Proc. IEEE Int. Conf. on Reconfigurable Computing and FPGAs (ReConFig)*, pp. 1–8, 2019.



5. A. Dörflinger and R. Albers, “A Comparative Survey of Open-Source Application-Class RISC-V Projects,” *Technical Report*, 2021.
6. D. Patterson, J. Bennett, and K. Eder, “Embench: A Modern Benchmark Suite for Embedded Systems,” *IEEE Computer*, vol. 54, no. 5, pp. 45–53, 2021.
7. V. Bertacco, A. Madhav, and M. C. Molina, “GreenRio: An Open-Source Educational Framework for Digital Design,” *Proc. IEEE Int. Conf. on Microelectronics Systems Education (MSE)*, 2022.
8. Z. Fu, Y. Li, J. He, and D. Rossi, “Ramping Up Open-Source RISC-V Cores: Assessing the Energy Efficiency of Superscalar, Out-of-Order Execution,” *Proc. ACM Computing Frontiers Conference (CF)*, pp. 110–118, 2025.
9. M. M. Eljhani, M. Benmohamed, and A. Boulahlib, “Design and Implementation of Five Stages Pipelined RISC Processor on FPGA,” *Proc. IEEE Mediterranean and African Conference on Information Systems and Technologies (MI-STA)*, 2023.
10. J. Yang, W. Zhao, and L. Liu, “Implement 32-bit RISC-V Architecture Processor Using Verilog HDL,” *Proc. IEEE Int. Symp. on Intelligent Signal Processing and Communication Systems (ISPACS)*, 2021.
11. P. Saiprathyusha and C. Chandrasekhar, “Implementation of RISC-V Processor,” *ITM Web of Conferences*, vol. 74, no. 02006, 2025.