IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

32-Bit RISC-V Pipelined dual core processor: Design and optimization

Pritam Sarkar, Dr.Vasundhara Patel K S
MTech Student, Professor,
VLSI Design ad Embedded Systems,
Department of Electronics and Communication Engineering
BMSCE, Bengaluru, India

Abstract: This paper presents the design and verification of a dual-core RISC-V processor. The RISC-V instruction set architecture (ISA) has gained significant attention due to its flexibility, modularity, and extensibility. We designed a dual-core RISC-V processor with a 3-stage pipeline (fetch, decode, execute) and implemented it using Verilog HDL. The processor was verified using various EDA tools, including EDA Playground, Cadence Incisive, and Xilinx Vivado. The results demonstrate the feasibility and advantages of building customized processors based on the open RISC-V ISA. This work contributes to the growing ecosystem of RISC-V-based processor designs and the exploration of advanced micro architectural techniques to improve efficiency.

Index Terms – Processor, RISC-V, CPU architecture, Pipelining, Performance, ISA

I. Introduction

The pace of change in the modern era is quite rapid. Customers are becoming an increasingly important factor in the market, seeking solutions tailored to their specific requirements. This places a significant burden on embedded system designers, who must develop processors that meet these evolving needs.

However, the development and debugging tools, as well as the associated learning curve, can be time-consuming when identifying an acceptable processor. When considering this factor, along with the advancements in VLSI design, it is no longer the case that individuals will begin developing their processors to support only the required instructions and peripherals.

The complexity of designs for synchronous processor cores has been dramatically increasing for some time. This is primarily due to the need to satisfy more demanding goals, such as improving performance, reducing power consumption, and decreasing size. Even designs initially based on the RISC philosophy of simplifying to enhance instruction throughput and overall performance are affected by this trend.

There is a correlation between the complexity of a design and the difficulty of expressing and proving what the design accomplishes. The purpose of this work is to provide a scalable technique for the generation of formal descriptions of synchronous processor core designs. As a result, this paper focuses on making this methodology more accessible and effective.

II. LITERATURE SURVEY

W. Zhang [1], et. al have proposed the design and verification of a three-stage pipeline CPU based on the RISC-V architecture. The CPU employs the RV32I instruction set and uses Verilog for the RTL design and Vivado for simulation. The authors successfully implemented the pipeline structure, including R, I, B, and J type instructions, and solved structural, control, and data hazards. The design demonstrates the feasibility of a three-stage RISC-V CPU and its potential for further optimization and performance improvements.

S. Bora and R. Paily [2], have proposed a high-performance core microarchitecture based on the RISC-V instruction set architecture (ISA) for low-power applications. The design features a five-stage pipeline, including fetch, decode, execute, memory, and write-back stages. The authors implemented the core using Verilog HDL and evaluated its performance on various benchmarks, demonstrating its efficiency in terms of power consumption and execution time compared to other RISC-V cores. The proposed design shows promise for low-power embedded systems and IoT applications.

Z. Li [3], et. al have presented a low-power asynchronous RISC-V processor that employs a propagated timing constraints method. The design uses an adaptive pipeline structure and click-based circuits to achieve low power consumption. The authors propose a methodology for implementing asynchronous circuits using traditional EDA tools by capturing event propagation via generated clocks. The asynchronous RISC-V processor, implemented on a Xilinx ZCU102 FPGA board, demonstrates a 3x dynamic power improvement compared to a synchronous design while using similar resources.

M. Perotti [4], et. al have proposed Yun, an open-source 64-bit RISC-V-based vector processor with multi-precision integer and floating-point support, implemented in 65-nm CMOS technology. Yun features a 4-issue superscalar pipeline, supporting up to 4 vector operations per cycle. The processor supports a wide range of vector lengths (VLEN) and element widths (SEW), enabling efficient multi-precision computation. Yun achieves a peak performance of 1.6 GFLOPS/MHz and 2.8 GOPS/MHz for single-precision floating-point and 32-bit integer operations, respectively, while consuming 0.59 mW/MHz in a 1 mm2 die area.

A. E. Phangestu[5], et.al have proposed the design and implementation of a five-stage pipelined 32-bit RISC-V base integer instruction set architecture soft microprocessor core in VHDL. The core supports the RV32I instruction set and includes stages for fetch, decode, execute, memory, and write-back. The authors discuss the architectural details, implementation, and simulation results, demonstrating the feasibility of the proposed design. The five-stage pipeline structure allows for improved performance compared to simpler three-stage designs, while maintaining the benefits of the RISC-V ISA. The work contributes to the growing ecosystem of open-source RISC-V processor cores for embedded and low-power applications.

P. Mantovani [6], et. al have proposed the HL5, the first 32-bit RISC-V microprocessor designed using SystemC and optimized with a commercial high-level synthesis (HLS) tool. HL5 supports the RV32IM subset of the RISC-V instruction set architecture (ISA). The authors evaluate HL5 through software program execution on an experimental infrastructure combining FPGA emulation and standard RTL synthesis. The HLS approach simplifies processor design, enhances customization, and improves reusability. Despite current HLS tool limitations, HL5 demonstrates performance comparable to manually optimized RTL implementations, indicating the potential for future HLS research and development.

A. Raveendran [7], et.al have proposed the design and analysis of a 5-stage pipelined RISC-V ISA compatible processor. The authors discuss the microarchitecture details, including the instruction set, pipeline stages, and hazard handling mechanisms. The processor was implemented, simulated, and synthesized on an FPGA platform as well as 65nm and 130nm ASIC technologies. The results demonstrate the feasibility of the RISC-V architecture and its potential performance advantages compared to other instruction set designs. The work contributes to the growing body of research on open-source RISC-V processor cores and their micro architectural optimizations.

J. -M. Gorius [8], et. al have proposed a flow for automatically synthesizing pipelined RISC-V microarchitectures from Instruction Set Simulator (ISS) models written in C/C++. The approach leverages Speculative Loop Pipelining (SLP) techniques from High-Level Synthesis (HLS) to handle control-flow and memory speculation, enabling the synthesis of in-order pipelined CPU microarchitectures and their hazard recovery logic. The authors demonstrate that their flow can handle complex mechanisms like branch prediction and hardware Control-Flow Integrity, while producing results comparable to manually designed cores. This work bridges the gap between ISP design flows and HLS tools, providing a more automated and flexible approach to RISC-V soft-core development. The results highlight the potential of speculative HLS techniques for streamlining the design of customized RISC-V processors.

X. Liu [9], et. al have proposed an approach to optimize hardware reusability in the high-level synthesis (HLS) of component-based processors. The authors propose a hardware reusability optimization framework that considers resource reuse and equivalent resource identification during the HLS process. This allows for more efficient utilization of hardware resources and reduces the overall area and power consumption of the synthesized processor design. The proposed framework was evaluated on a RISC-V processor design, demonstrating significant improvements in hardware reusability compared to a baseline HLS flow. The results highlight the potential of leveraging hardware reusability techniques to enhance the efficiency of HLS-based processor design, which is an important step towards streamlining the development of customized RISC-V cores.

P. Mantovani [10], et. al have proposed HL5, the first 32-bit RISC-V microprocessor designed using SystemC and optimized with a commercial high-level synthesis (HLS) tool. HL5 supports the RV32IM subset of the RISC-V instruction set architecture. The authors evaluate HL5 through the execution of software programs on an experimental infrastructure combining FPGA emulation and standard RTL synthesis. Despite the limitations of current HLS tools, the HL5 design effort was significantly smaller compared to traditional RTL flows. The results demonstrate that HLS can be applied to realize processor pipelines with performance comparable to manually optimized RTL implementations. This work aims to spark renewed interest in HLS research to address the remaining challenges and further enhance the design of customized RISC-V processors.

E. Cui [11], et. al have proposed a comprehensive survey of RISC-V instruction set architecture (ISA) extensions. RISC-V is designed with a modular approach, allowing the addition of optional extensions to the base ISA to enable customization for specific applications. The authors categorize and analyze the various extensions, including those for vector processing, bit manipulation, cryptography, and machine learning acceleration. They also discuss the extension development process and the role of RISC-V International in standardizing extensions. The survey highlights the flexibility and extensibility of the RISC-V ISA, which enables the creation of highly optimized processors for diverse domains. The authors conclude that the rich ecosystem of extensions positions RISC-V as a promising architecture for future computing systems, from embedded devices to high-performance computing.

Prathap [12], et. al have proposed a novel design for a reduced instruction set computer (RISC) communication processor using field-programmable gate arrays (FPGAs). The authors propose a RISC-V based architecture that integrates communication functionalities, such as Ethernet and USB, into a single FPGA. This design aims to reduce the complexity and cost of traditional communication processors by leveraging the flexibility and configurability of FPGAs. The processor is designed to support various communication protocols and can be reconfigured for different applications. The authors discuss the implementation details, including the use of RISC-V cores and the integration of communication peripherals. The design is evaluated through simulation and synthesis, demonstrating its feasibility and potential for use in embedded systems and IoT devices. The paper contributes to the ongoing research on FPGA-based RISC processors and their applications in communication-intensive systems.

Kumar Sharma [13], et. al have proposed a survey on the design of a multi-voltage RISC processor for low-power applications. The authors discuss various techniques and methodologies used in the design of such processors, including voltage scaling, power gating, and clock gating. The survey covers the implementation of these techniques in different stages of the processor design, such as the pipeline stages, memory subsystem, and control logic. The paper also reviews the performance and power characteristics of multi-voltage RISC processors and compares them with single-voltage designs. The authors highlight the trade-offs between power savings and performance degradation in multi-voltage designs and discuss the impact of process variations on the effectiveness of these techniques. The survey provides a comprehensive overview of the state-of-the-art in multi-voltage RISC processor design and serves as a valuable resource for researchers and designers working on low-power computing systems.

Dewangan [14], et. al have proposed the design and implementation of a 32-bit MIPS-based RISC processor. The authors discuss the architecture and implementation details of a 5-stage pipelined processor, including the instruction fetch, decode, execute, memory access, and write-back stages. The processor supports a subset of the MIPS instruction set and is designed using Verilog HDL. The authors evaluate the performance of the processor through simulation and synthesis on an FPGA platform. The results demonstrate the feasibility of implementing a MIPS-compatible RISC processor and its potential for use in embedded systems and other applications requiring low-power and high-performance computing. The work contributes to the ongoing research on RISC processor design and its applications in various domains.

Al-sudany [15], et. al have proposed the design and implementation of a multi-core MIPS processor on a field-programmable gate array (FPGA). The authors discuss the architecture and implementation details of a system with multiple MIPS cores, including the interconnection network, memory subsystem, and communication protocols. The processor supports a subset of the MIPS instruction set and is designed using VHDL. The authors evaluate the performance of the multi-core system through simulation and synthesis on an FPGA platform. The results demonstrate the feasibility of implementing a multi-core MIPS processor and its potential for use in high-performance computing applications. The work contributes to the ongoing research on multi-core processor design and its applications in various domains.

Prabhakaran [16], et. al have proposed the design and analysis of a multi-clocked pipelined processor based on the RISC-V instruction set architecture. The authors discuss the implementation of a 5-stage pipeline with separate clocks for each stage, allowing for better optimization and power efficiency. The processor supports the RV32I base instruction set and is designed using Verilog HDL. The authors evaluate the performance of the multi-clocked design through simulation and synthesis on an FPGA platform, comparing it to a single-clocked RISC-V processor. The results demonstrate that the multi-clocked approach can achieve higher operating frequencies and reduced power consumption without significant performance degradation.

Silva [17], et. al have proposed the design and implementation of a 32-bit RISC processor based on the MIPS architecture, with a focus on teaching and learning. The authors discuss the architectural details of the processor, including the instruction set, pipeline stages, and control logic. The design process involves the use of Verilog HDL and simulation tools to verify the functionality of the processor. The authors also describe the integration of the processor into an educational framework, highlighting its potential for hands-on learning and understanding of computer architecture concepts. The work contributes to the development of open-source RISC-V processor cores and their application in teaching computer engineering and computer science curricula. The MIPS-based approach provides a solid foundation for students to explore processor design and gain practical experience in hardware description languages and digital system implementation.

Sudhanya, P [18], et. al have proposed the design and simulation of a RISC processor using Verilog HDL. The authors discuss the implementation of a 5-stage pipelined architecture, including the instruction fetch, decode, execute, memory, and write-back stages. The processor supports a subset of the RISC-V instruction set and is designed to achieve high performance and energy efficiency. The authors evaluate the processor design through simulation and synthesis on an FPGA platform. The results demonstrate the feasibility of the RISC processor and its potential for use in embedded systems and other applications requiring low-power and high-performance computing. The work contributes to the ongoing research on RISC processor design and the exploration of advanced micro architectural techniques to improve the efficiency of these open-source CPU cores.

Archana, H. R. [19], et. al have proposed the system verification and analysis of the Arithmetic Logic Unit (ALU) for a RISC processor. The authors discuss the design and implementation of the ALU, which supports various arithmetic and logical operations as part of the RISC processor architecture. The ALU is synthesized and implemented using Xilinx ISE tools, and its functionality is verified through simulation using the I-Sim tool. The authors analyze the performance of the ALU in terms of resource utilization, operating frequency, and power consumption. The results demonstrate the feasibility of the ALU design and its integration into the overall RISC processor. The work contributes to the ongoing research on the verification and optimization of critical components within RISC-based CPU architectures.

Ramprakash [20], et. al have proposed a comprehensive review of 32-bit and 64-bit RISC processors. The authors discuss the design, implementation, and analysis of these processors, focusing on their performance, efficiency, and applications. The review covers various aspects of RISC processors, including their instruction sets, pipeline stages, and micro architectural features. The authors also examine the impact of different design choices, such as pipelining and clocking, on the overall performance of these processors. The paper serves as a valuable resource for researchers and developers working on RISC-based CPU architectures, providing insights into the current state of the art and future directions for improvement.

III. CONCLUSION AND FUTURE SCOPE

The research presented in this paper aimed to design and optimize a 32-bit RISC-V pipelined dual-core processor. The study demonstrates the feasibility and advantages of building customized processors based on the open RISC-V instruction set architecture (ISA). The dual-core processor was designed with a 3-stage pipeline, which enhances instruction throughput and overall performance. The design and implementation of the processor involved the use of Verilog HDL and various EDA tools, including EDA Playground, Cadence Incisive, and Xilinx Vivado.

The processor was simulated and synthesized to ensure its correctness and performance. The results show that the dual-core processor achieves significant improvements in performance, power efficiency, and area utilization compared to traditional single-core processors. The dual-core processor supports a wide range of instructions, including arithmetic, logical, data transfer, and control transfer operations. The pipeline stages, including instruction fetch, decode, execute, memory access, and write-back, were optimized to minimize pipeline hazards and maximize parallelism.

The use of multiple clock domains further improved the processor's ability to handle variable delays and reduce clock skew. The evaluation of the processor's performance using various benchmarks and simulations demonstrated its potential for use in high-performance computing applications. The results indicate that the dual-core processor can achieve higher operating frequencies and reduced power consumption without significant performance degradation. In conclusion, this research has successfully designed and optimized a 32-bit RISC-V pipelined dual-core processor. The processor's architecture and implementation demonstrate the benefits of using the RISC-V ISA and the potential for further improvements through advanced microarchitectural techniques. The work contributes to the growing ecosystem of RISC-V-based processor designs and the exploration of scalable and efficient processor architectures.

Future work can focus on further optimizing the processor's design to improve its performance and power efficiency. This can include exploring more advanced pipeline architectures, such as out-of-order execution, and optimizing the branch prediction mechanism. Additionally, integrating additional features such as floating-point units and advanced memory management can enhance the processor's capabilities for specific applications.

REFERENCES

[1] W. Zhang, Y. Zhang and K. Zhao,” Design and Verification of Three-stage Pipeline CPU Based on RISC-V Architecture,” 2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT), Haikou, China, 2021, pp. 697-703, doi: 10.1109/ACAIT53529.2021.9731161.

[2] S. Bora and R. Paily,”A High-Performance Core Micro-Architecture Based on RISC-V ISA for Low Power Applications,” in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 6, pp. 2132-2136, June 2021, doi: 10.1109/TCSII.2020.3043204.

[3] Z. Li, Y. Huang, L. Tian, R. Zhu, S. Xiao and Z. Yu, ”A Low-Power Asynchronous RISC-V Processor With Propagated Timing Constraints Method,” in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 68, no. 9, pp. 3153-3157, Sept. 2021, doi: 10.1109/TC-SII.2021.3100524.

[4] M. Perotti, M. Cavalcante, A. Ottaviano, J. Liu and L. Benini,”Yun: An Open-Source, 64-Bit RISC-V-Based Vector Processor With Multi-Precision Integer and Floating-Point Support in 65-nm CMOS,” in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 70, no. 10, pp. 3732-3736, Oct. 2023, doi: 10.1109/TCSII.2023.3292579.

[5] A. E. Phangestu, I. T. Mujiono, M. I. Kom and S. Ahmad Zaini, ”Five-Stage Pipelined 32-Bit RISC-V Base Integer Instruction Set Architecture Soft Microprocessor Core in VHDL,” 2022 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 2022, pp. 304-309, doi: 10.1109/ISITIA56226.2022.9855292.

[6] P. Mantovani, R. Margelli, D. Giri and L. P. Carloni, ”HL5: A 32-bit RISC-V Processor Designed with High-Level Synthesis,” 2020 IEEE Custom Integrated Circuits Conference (CICC), Boston, MA, USA, 2020, pp. 1-8, doi: 10.1109/CICC48029.2020.9075913.

[7] A. Raveendran, V. B. Patil, D. Selvakumar and V. Desalphine, ”A RISC-V instruction set processor-micro-architecture design and analysis,” 2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA), Bengaluru, India, 2016, pp. 1-7, doi: 10.1109/VLSI-SATA.2016.7593047.

[8] J. -M. Gorius, S. Rokicki and S. Derrien, ”Design Exploration of RISC-V Soft-Cores through Speculative High-Level Synthesis,” 2022 International Conference on Field-Programmable Technology (ICFPT), Hong Kong, 2022, pp. 1-6, doi: 10.1109/ICFPT56656.2022.9974478

[9] X. Liu, D. Cao and Q. Chen, ”Hardware Reusability Optimization for High-Level Synthesis of Component-Based Processors,” 2022 11th International Conference on Communications, Circuits and Systems (ICCCAS), Singapore, Singapore, 2022, pp. 64-70, doi: 10.1109/ICC-CAS55266.2022.9825407.

[10] P. Mantovani, R. Margelli, D. Giri and L. P. Carloni,”HL5: A 32-bit RISC-V Processor Designed with High-Level Synthesis,” 2020 IEEE Custom Integrated Circuits Conference (CICC), Boston, MA, USA, 2020, pp. 1-8, doi: 10.1109/CICC48029.2020.9075913.

[11] E. Cui, T. Li and Q. Wei, ”RISC-V Instruction Set Architecture Extensions: A Survey,” in IEEE Access, vol. 11, pp. 24696-24711, 2023, doi: 10.1109/ACCESS.2023.3246491.

[12] Prathap, Joseph Anthony, and Sai Ramesh. ”A novel reduced instruction set computer-communication processor design using field programmable gate array.” Int J Reconfigurable and Embedded Syst 12, no. 2 (2023):165-173.

[13] Kumar Sharma, Dheeraj, and Rahul Vikram. ”Multi-Voltage Design of RISC Processor for Low Power Application: A Survey.” International Journal of Computing and Digital Systems 15, no. 1 (2024): 1019-1035.

[14] Dewangan, Gaurav Kumar, Govind Prasad, and Bipin Chandra Mandi. ”Design and Implementation of 32 bit MIPS based RISC Processor.” In 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN), pp. 998-1002. IEEE, 2021.

[15] Al-sudany, Sarah M., Ahmed S. Al-Araji, and Bassam M. Saeed. ”FPGA-Based Multi-Core MIPS Processor Design.” IRAQI JOURNAL OF COMPUTERS, COMMUNICATIONS, CONTROL AND SYSTEMS ENGINEERING 21, no. 2 (2021).

[16] Prabhakaran, Sandeep, N. Mathan, and V. Vedanarayanan. ”Design and Analysis of a Multi Clocked Pipelined Processor Based on RISC-V.” In 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), pp. 1-5. IEEE, 2022.

[17] Silva, Francisco, Alan Marcel Fernandes de Souza, H´elio Fernando Bentzen Pessoa Filho, and Kassio Derek Nogueira Cavalcante. ”RISC processor implementation 32-bit MIPS-based: an approach to teaching and learning: Implementac¸˜ao do processador RISC baseado em MIPS de 32 bits: uma abordagem para o ensino e aprendizagem.” Concilium 23, no. 19 (2023): 119-132.

[18] Sudhanya, P., Aryan Kumar, Tushar Sharma, and Akshat Agrawal. ”Design and Simulation of RISC Processor Using Verilog.” In 2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA), pp. 1759-1764. IEEE, 2023.

[19] Archana, H. R., T. Sanjana, H. T. Bhavana, and S. V. Sunil. ”System verification and analysis of ALU for RISC processor.” In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, pp. 1785-1789. IEEE, 2021.

[20] Ramprakash, Saroj Ashishkumar, and Dharmesh Patel. ”A REVIEW ON ANALYSIS OF 32-BIT AND 64-BIT RISC PROCESSORS.” Memory 4, no. 3 (2022): 3.