Enhancing Cryptographic Application Performance on RISC-V Processors with B and K Extensions

1st Harsh Mayank Dabhi
dept. Electronics and computer engineering
Dublin city university
Dublin, Ireland
harshmayank.dabhi2@mail.dcu.ie

***Abstract*—Cryptographic algorithms such as AES (Advanced Encryption Standard) and SHA (Secure Hash Algorithm) are critical for ensuring data security in modern computing systems. However, their computational complexity poses challenges for throughput, energy efficiency, and resource utilization. The advent of RISC-V architecture's B (Bit-Manipulation) and K (Cryptography) extensions offers promising avenues to accelerate cryptographic workloads by leveraging hardware-level enhancements. This study evaluates the performance benefits of these extensions using RISC-V simulators, focusing on key metrics such as throughput, energy efficiency, and resource utilization. A comprehensive literature survey highlights existing optimization techniques for AES and SHA and identifies gaps in hardware-assisted implementations. The findings provide a foundation for assessing the potential of RISC-V cryptographic extensions to achieve significant performance improvements, supporting their adoption in security-critical applications.**

I. Introduction

Cryptographic algorithms play a pivotal role in securing modern communication systems, ensuring data confidentiality, integrity, and authenticity. Among these, the Advanced Encryption Standard (AES) and Secure Hash Algorithm (SHA) are widely utilized across industries due to their robustness and versatility. However, the computational intensity of these algorithms can lead to bottlenecks, especially in resource-constrained environments, where throughput, energy efficiency, and hardware utilization are critical considerations.

The RISC-V architecture, an open and extensible Instruction Set Architecture (ISA), has gained significant attention for its modular design and flexibility. To address the performance demands of cryptographic workloads, RISC-V introduces specialized instruction set extensions, namely the B (Bit-Manipulation) and K (Cryptography) extensions. These extensions aim to accelerate operations fundamental to cryptographic algorithms, such as bitwise manipulation, modular arithmetic, and other cryptographic primitives, thereby enhancing performance metrics like speed, energy consumption, and resource utilization.

This literature survey examines the state-of-the-art research and advancements related to the implementation and performance evaluation of AES and SHA algorithms on RISC-V platforms. It explores the impact of the B and K extensions on cryptographic operations, reviews prior methodologies for optimization, and analyzes the role of simulation tools in assessing these enhancements. By identifying the strengths and limitations of existing approaches, this study sets the stage for a well-informed evaluation of the RISC-V cryptographic extensions and their potential to transform cryptographic workloads in modern applications.

II. Review and Analysis of Prior Work

A. RISC-V Architecture and Its Extensions

The RISC-V architecture (as shown in Fig. 1)[10] has emerged as a transformative platform in the field of computing due to its open-source nature and modular design, which allows for the development of domain-specific extensions. Among these, the B (Bit-Manipulation) and K (Cryptography) extensions have been specifically designed to improve the performance of cryptographic and other computationally intensive workloads.

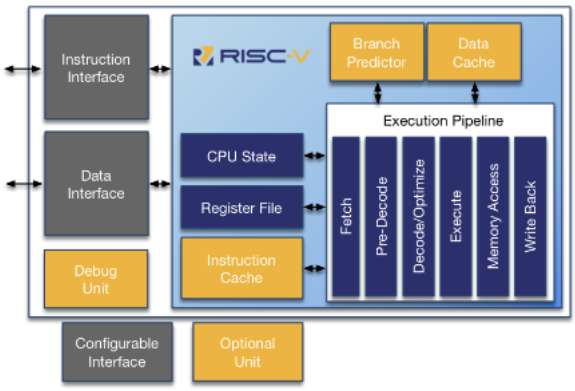

Fig. 1. Risc-V block diagram.

The B extension introduces a range of bit-level instructions, including bitwise shifts, rotations, population count, and bit extraction, which are essential for optimizing cryptographic operations. Cryptographic algorithms such as AES (Advanced Encryption Standard) and SHA (Secure Hash Algorithm) depend heavily on bit-level manipulation during key expansion, substitution steps, and hashing processes. Research has shown

that implementations leveraging the cryptography set extension, which incorporates these bit manipulation instructions, achieve a 1.5× to 8.6× faster execution speed compared to those relying solely on the base RV32I instruction set and 1.2× to 5.8× less program memory for five of the eleven algorithms. This performance improvement is particularly significant for algorithms like SHA-256, which require extensive bitwise operations[1].

The RISC-V K extension introduces specialized instructions to accelerate cryptographic primitives, such as those used in AES (e.g., SubBytes, MixColumns, and AddRoundKey) and SHA (e.g., compression functions and message schedules). These instructions are designed to execute specific steps of cryptographic algorithms in hardware, thereby bypassing the inefficiencies of software-only implementations[2][3].Evaluations of the K extension have demonstrated significant performance improvements. For instance, research indicates that implementing AES-128 encryption with dedicated instruction set extensions can achieve performance gains on 32-bit architectures and on 64-bit architectures, compared to software-only implementations[2].

Although the RISC-V K extension accelerate cryptographic operations. However, this enhancement comes with trade-offs in hardware complexity and energy consumption. Implementing the K extension necessitates additional silicon area for the specialized instructions, leading to a moderate increase in power consumption due to the higher density of hardware resources. While this trade-off is generally acceptable for high-performance systems, it can pose challenges for power-constrained environments such as IoT devices.For instance, a study on integrating a hybrid encryption accelerator into a low-power RISC-V processor for IoT applications highlights the importance of balancing security and energy efficiency. The research emphasizes that while hardware accelerators enhance performance, they must be carefully designed to meet the stringent energy constraints of IoT devices.[4]

B. Optimization Techniques for AES and SHA

AES and SHA have been subjects of extensive research and optimization, with efforts focusing on both software and hardware implementations.

Software-based cryptographic implementations are typically designed for general-purpose processors without hardware acceleration. They rely on algorithmic optimizations, such as loop unrolling, table-driven approaches, and SIMD (Single Instruction Multiple Data) instructions, to enhance performance. For example, AES implementations often utilize precomputed lookup tables for the SubBytes operation to reduce computational overhead.

Hardware-based solutions, such as Instruction Set Extensions (ISEs) like the K extension in RISC-V, integrate cryptographic primitives directly into the processor, leveraging parallelism and hardware-level optimizations to achieve significant performance improvements. For instance, implementing AES round functions in hardware drastically reduces cycle counts compared to software-based implementations. Similarly, hardware acceleration of SHA-256 within a RISC-V General-Purpose Graphics Processing Unit (GPGPU) has shown substantial speedups over traditional software methods, as demonstrated in the research paper ”Cryptography Acceleration in a RISC” [5]. GPUs, with their parallel processing capabilities, are particularly well-suited for accelerating cryptographic operations. By offloading SHA-256 computations to a RISC-V GPGPU, the research shows that the GPU's ability to process multiple data streams in parallel significantly reduces the time required for cryptographic tasks. This is especially effective for algorithms like SHA-256, which involve repetitive and parallelizable computations, allowing the GPU's architecture to deliver notable performance gains over CPU-based software implementations.

C. Performance Metrics and Benchmarking

Performance evaluation is a critical component in assessing the impact of the RISC-V extensions on cryptographic workloads. Key metrics include:

Throughput: This metric measures the number of cryptographic operations performed per unit of time. Studies have demonstrated that RISC-V implementations utilizing the K extension achieve significant improvements in throughput for AES encryption.

Energy Efficiency: This evaluates the power consumption per cryptographic operation. By offloading cryptographic computations to hardware, the K extension reduces the energy cost per operation by minimizing execution time and the number of active processor cycles. Research indicates that integrating a crypto-extension into an open source RISC-V core improves energy efficiency while maintaining flexibility [6].

Resource Utilization: Resource utilization measures the hardware cost associated with implementing the extensions. While the K extension requires additional hardware resources, its area overhead is minimized through careful design.Hardware cost of implementing the K extension is comparable to existing cryptographic accelerators while delivering superior performance.

D. Role of Simulation in Performance Evaluation

Simulation tools play a vital role in evaluating the performance of RISC-V extensions, particularly in the absence of physical hardware. Tools like Spike, QEMU, and Gem5 are widely used to model the behavior of RISC-V processors with and without extensions.

*1) **Spike and QEMU**:* These simulators provide cycle-accurate modeling of RISC-V processors and enable researchers to evaluate the impact of the B and K extensions on cryptographic workloads. They are particularly useful for comparing execution times and energy consumption for different configurations.

*2) **Gem5 and Power Modeling**:* Gem5 extends the capabilities of standard simulators by integrating power models, allowing researchers to estimate energy efficiency alongside throughput and resource utilization. This holistic approach provides a comprehensive view of the trade-offs associated with implementing RISC-V extensions.

*3) **Challenges and Limitation**:* While simulators are invaluable for early-stage evaluation, they often fail to capture real-world constraints, such as thermal effects and hardware-level interactions. These limitations highlight the need for experimental validation on physical RISC-V hardware platforms.

E. Comparative Analysis of Approaches

The reviewed literature provides a clear distinction between software-only and hardware-accelerated approaches for cryptographic workloads:

- **Software-Only Solutions:** Ideal for general-purpose use cases where flexibility is prioritized over performance.
- **Instruction Set Extensions:** Suitable for high-performance applications, offering superior throughput and energy efficiency.

The B and K extensions in RISC-V demonstrate substantial potential to enhance cryptographic operations, particularly AES and SHA. However, their adoption must be guided by application-specific requirements, considering factors like energy constraints, hardware cost, and desired performance levels.

III. Relation of Prior Work to the Project Problem

The reviewed literature provides a comprehensive understanding of the capabilities and limitations of RISC-V's B (Bit-Manipulation) and K (Cryptography) extensions in enhancing cryptographic workloads such as AES and SHA. This section evaluates how prior studies relate to the objectives of this project and identifies areas where existing approaches align or diverge from the project's focus. Additionally, it highlights the need for specific adaptations to fully address the research goals of this work.

*a) **Comparison of Prior Work with the Project Focus**:* The RISC-V architecture's B and K extensions are designed to enhance the performance of cryptographic operations, particularly algorithms like AES and SHA. Studies have shown that these extensions can significantly reduce execution cycles and improve throughput.

For instance, the paper "Symmetric Cryptography on RISC-V: Performance Evaluation of Standardized Algorithms" discusses the impact of these extensions on AES and SHA algorithms. The study found that implementations utilizing the cryptography extension achieved a 1.5× to 8.6× increase in execution speed and a 1.2× to 5.8× reduction in program memory usage compared to implementations using only the base rv32i instruction set[1].

Another relevant work is "The Design of Scalar AES Instruction Set Extensions for RISC-V," which evaluates various instruction set extensions for AES. The authors report performance improvements of 4× for 32-bit and 10× for 64-bit architectures when compared to software-only implementations. The hardware cost for these enhancements is relatively modest, at 1.1K and 8.2K gates, respectively[2].

However, these studies primarily focus on metrics like execution speed and cycle counts. There is a noted gap in the literature regarding comprehensive evaluations that also consider energy efficiency and resource utilization. Addressing this gap is crucial, especially for applications in resource-constrained environments such as IoT devices, where energy efficiency and minimal hardware overhead are critical.

To bridge this gap, future research should incorporate realistic workload scenarios and expand performance evaluations to include multiple dimensions: throughput, energy efficiency, and hardware cost. Such holistic assessments will provide a more complete understanding of the trade-offs involved in implementing these extensions in various applications.

*b) **Applicability and Adaptation of Prior Methods**:* To effectively evaluate the RISC-V B and K extensions, it's essential to adapt existing simulation tools to assess not only performance metrics but also power consumption and resource utilization. While simulators like Spike and QEMU are proficient in modeling RISC-V processors and providing cycle-accurate simulations, they lack inherent capabilities for power analysis.

To bridge this gap, integrating power modeling tools such as McPAT with simulators like Gem5 is a viable approach. McPAT offers detailed power, area, and timing models, enabling comprehensive energy consumption analysis. The gem5 simulator, known for its detailed microarchitectural modeling, can be extended to support power estimation by exporting simulation statistics to McPAT. This integration facilitates the evaluation of energy efficiency alongside performance metrics. For instance, the McPAT-Calib framework enhances McPAT to support RISC-V BOOM microarchitectures, providing a calibrated power modeling solution[7].

Existing studies often utilize isolated test cases, which may not accurately reflect real-world applications. To address this, it's crucial to design test scenarios that encompass dynamic key generation, varying input sizes, and mixed cryptographic operations. This approach ensures that evaluations are representative of practical use cases, providing insights into the trade-offs between performance, energy consumption, and hardware resources. The GVSoC simulator, for example, offers a highly configurable and accurate full-platform simulation environment for RISC-V-based IoT processors, enabling the modeling of complex workloads. [8].

*c) **Unsuitability of Certain Approaches**:* Software-only optimizations, such as loop unrolling and table-driven techniques, can enhance performance to a certain extent but often fall short in accelerating computationally intensive cryptographic workloads. These methods are limited by the underlying hardware architecture and the absence of specialized instructions for cryptographic operations. In the paper "Efficient Cryptography on the RISC-V Architecture," Ko Stoffelen [9] explores the performance of cryptographic primitives on the RISC-V architecture and highlights the limitations of software-only optimizations. The study emphasizes that while software optimizations can improve performance, they are often insufficient for achieving significant speedups in cryptographic computations. Stoffelen suggests that hardware-level enhancements, such as the RISC-V B extension (bit manipulation) and other potential extensions, could provide substantial performance improvements by introducing specialized instructions for operations like rotations and carry handling, which are critical for cryptographic algorithms[9].

Additionally, proprietary hardware accelerators, though effective in high-performance scenarios, are often closed-source, limiting transparency and reproducibility. The open-source nature of the RISC-V architecture offers a more accessible platform for evaluating general-purpose cryptographic acceleration, aligning with the project's objectives. This openness facilitates collaborative development and comprehensive assessment of hardware-accelerated cryptographic solutions.

Therefore, relying solely on software optimizations or proprietary hardware accelerators is unsuitable for this project's goals, which focus on evaluating open-source, hardware-accelerated solutions for cryptographic workloads.

*d) **Proposed Solution Strategy**:* Drawing on the insights from prior work, this project proposes a multi-faceted solution strategy that builds on the strengths of existing approaches while addressing their limitations:

1) **Comprehensive Benchmarking Framework**:
   The project will utilize RISC-V simulators, such as Spike and QEMU, to evaluate AES and SHA performance with and without the B and K extensions. Key metrics will include throughput, execution cycles, and energy consumption, providing a well-rounded performance assessment.
2) **Integration of Enhanced Power Models**:
   To address the limitations of prior studies, the project will incorporate power modeling tools, such as McPAT or custom extensions to Spike, to estimate energy efficiency. This adaptation is critical for understanding the trade-offs in resource-constrained environments.
3) **Realistic Workload Design**:
   Test cases will emulate real-world cryptographic operations, including dynamic key expansion for AES, multi-message hashing for SHA, and mixed cryptographic workloads. These scenarios will provide insights into the practical applicability of the RISC-V extensions in diverse applications.
4) **Holistic Analysis of Trade-Offs**: The project will evaluate not only performance improvements but also the hardware costs associated with implementing the B and K extensions. This analysis will ensure that the findings are relevant to both high-performance and resource-constrained applications. Instead of focusing on speed of execution, the evaluation will utilize other benchmarking tools such as resource utilization, throughput, and efficiency metrics.

By leveraging state-of-the-art methods and addressing the gaps identified in the literature, this project aims to provide a nuanced and comprehensive evaluation of the RISC-V B and K extensions. The proposed solution strategy balances performance gains with hardware and energy trade-offs, offering valuable insights for future adoption of these extensions in cryptographic applications.

IV. Conclusion

This study aims to evaluate the performance acceleration achieved by leveraging the RISC-V architecture's B (Bit-Manipulation) and K (Cryptography) extensions for cryptographic workloads, specifically AES and SHA. The literature review highlights the significant benefits of these extensions in improving throughput, reducing execution cycles, and enhancing energy efficiency. However, it also reveals gaps in existing research, particularly in the holistic evaluation of multiple performance metrics and the applicability of findings to real-world cryptographic scenarios.

By building on state-of-the-art methods, this project proposes a comprehensive evaluation framework that integrates advanced power modeling and resource utilization analysis into RISC-V simulators. Additionally, the adoption of realistic workload scenarios will ensure the relevance of the results to practical applications. Through this approach, the project not only seeks to quantify the benefits of the B and K extensions but also to understand the trade-offs involved in their implementation, particularly in resource-constrained environments like IoT and embedded systems.

Ultimately, the findings of this project will contribute to a deeper understanding of how hardware-level enhancements in open-source architectures like RISC-V can transform cryptographic operations. By addressing existing gaps and providing actionable insights, this work aims to guide future research and development in the design and optimization of cryptographic solutions using extensible architectures.

V. References

[1] G. Nişancı, P. G. Flikkema and T. Yalçın "Symmetric Cryptography on RISC-V: Performance Evaluation of Standardized Algorithms". https://www.mdpi.com/2410-387X/6/3/41

[2] B. Marshall, G. R. Newell, D. Page, M. O. Saarinen and C. Wolf "The design of scalar AES Instruction Set Extensions for RISC-V". https://eprint.iacr.org/2020/930.pdf

[3] H. Li, N. Mentens and S. Picek “Maximizing the Potential of Custom RISC-V Vector Extensions for Speeding up SHA-3 Hash”. https://eprint.iacr.org/2022/868.pdf
[4] S. Yang, L. Shao, J. Huang and W. Zou “Design and Implementation of Low-Power IoT RISC-V Processor with Hybrid Encryption Accelerator”. https://www.mdpi.com/2079-9292/12/20/4222
[5] A. Adams, P. Gupta, B. Tine, H. Kim “Cryptography Acceleration in a RISC”.
[6] W.wang, J. Han, X. Cheng, X. Zeng “An energy-efficient crypto-extension design for RISC-V”.
[7] J. Zhai, C. Bai, B. Zhu, Y. Cai, Q. Zhou, B. Yu “McPAT-Calib: A RISC-V BOOM Microarchitecture Power Modeling Framework”.
[8] N. Bruschi, G. Haugou, G. Tagliavini, F. Conti, L. Benini, D. Rossi “A Highly Configurable, Fast and Accurate Full-Platform Simulator for RISC-V based IoT Processors”. https://arxiv.org/abs/2201.08166
[9] Ko Stoffelen “Efficient Cryptography on the RISC-V Architecture”. https://eprint.iacr.org/2019/794
[10] https://roalogic.github.io/RV12/DATASHEET.html