Low power RISC-V Processor Design with Hardware optimization for Edge Computing Application

Durgadevi M
Department of Electronics And Communication Engineering
Sir issac newton college of engineering and technology
Nagapattinam
Email id:Durga0203@gmail.com

Gokulalakshmi R
Department of Electronics And Communication Engineering
Sir issac newton college of engineering and technology
Nagapattinam
Email id: gokul121@gmail.com

Guhan T Department of Electronics And Communication Engineering
Sir issac newton college of engineering and technology
Nagapattinam
Email id: guhanece@gmail.com

Guna K
Department of Electronics And Communication Engineering
Sir issac newton college of engineering and technology
Nagapattinam
Email id:gunaguna@gmail.com

Durgalakshmi,
Assistant Professor,
Department of Electronics And Communication Engineering
Sir issac newton college of engineering and technology
Nagapattinam
Email id: durgalakshmi@sincet.ac.in

Abstract

Edge computing requires processors that can perform real-time computation with low power consumption and limited hardware resources. RISC-V provides an open and flexible instruction set architecture suitable for designing energy-efficient processors for edge devices. This paper presents a low-power RISC-V processor design with hardware-level optimization techniques for edge computing applications. The proposed approach focuses on reducing unnecessary switching activity, optimizing the processor datapath, simplifying control logic, and improving memory access efficiency. These techniques can reduce dynamic power consumption while maintaining suitable computational performance. The design is targeted toward resource-constrained applications such as Internet of Things (IoT) devices, wearable electronics, environmental monitoring systems, and intelligent sensors. The proposed architecture demonstrates how hardware optimization can provide a balance between power efficiency, processing capability,

and implementation complexity. The study highlights the suitability of optimized RISC-V processors for next-generation low-power edge computing systems.

Keywords— RISC-V, Low Power, Hardware Optimization, Edge Computing, IoT, Processor Design, Energy Efficiency, Embedded Systems.

I. INTRODUCTION

The rapid growth of Internet of Things (IoT), artificial intelligence, and smart sensing applications has increased the demand for processing data closer to where it is generated. Edge computing reduces the dependency on remote cloud servers by performing computation near the data source. However, edge devices generally operate with limited power, memory, and processing resources. Therefore, the processor used in an edge device must provide sufficient computational capability while consuming minimum energy.

RISC-V is an open standard instruction set architecture that provides flexibility for processor design. Its modular structure allows designers to select suitable instruction extensions according to application requirements. This makes RISC-V an attractive choice for embedded and edge computing systems.

A low-power processor can be achieved through several hardware optimization techniques. Reducing unnecessary switching activity, optimizing datapath components, improving clock management, and reducing memory accesses are important approaches for minimizing power consumption. The objective of this paper is to study a low-power RISC-V processor architecture and discuss hardware optimization techniques suitable for edge computing applications.

II. RISC-VPROCESSOR ARCHITECTURE

RISC-V follows the Reduced Instruction Set Computer (RISC) principle. It uses a simple and modular instruction set that can be extended according to application requirements. A basic RISC-V processor consists of an instruction fetch unit, instruction decoder, register file, arithmetic logic unit (ALU), control unit, and memory interface.

The modular architecture of RISC-V provides flexibility for developing application-specific processors. For low-power edge devices, unnecessary processor features can be avoided and only the required instruction extensions can be implemented. This can reduce hardware complexity and power consumption.

A. Instruction Fetch Unit

The instruction fetch unit obtains instructions from instruction memory and provides them to the processor pipeline. Efficient instruction fetching can reduce unnecessary memory activity and improve energy efficiency.

B. Register File

The register file stores temporary data required during instruction execution. Optimizing register access and avoiding unnecessary read/write operations can reduce switching activity.

C. Arithmetic Logic Unit

The ALU performs arithmetic and logical operations. Hardware optimization of the ALU can reduce area and switching power while maintaining the required processing performance.

D. Control Unit

The control unit generates control signals required for instruction execution. Simplifying control logic and reducing unnecessary transitions can contribute to overall power reduction.

III. HARDWARE OPTIMIZATION TECHNIQUES

Hardware optimization is important for developing energy-efficient processors for edge applications. The following techniques can be applied to a RISC-V processor.

A. Clock Gating

Clock gating reduces dynamic power by disabling the clock signal to processor units that are not currently required. For example, if a functional unit is inactive, its clock can be temporarily disabled. This reduces unnecessary switching activity.

B. Datapath Optimization

The datapath contains several computational components such as ALUs, multiplexers, registers, and shifters. Optimizing these components can reduce hardware complexity and switching activity. Application-specific datapath configurations can further improve energy efficiency.

C. Memory Access Optimization

Memory operations can consume significant energy in an embedded processor. Reducing unnecessary memory accesses and improving data reuse can lower energy consumption. Small local memories or optimized cache configurations can be considered depending on application requirements.

D. Pipeline Optimization

Pipelining can improve processor throughput, but excessive pipeline stages may increase hardware complexity and switching activity. Therefore, an appropriate number of pipeline stages should be selected according to the target edge application.

E. Operand Isolation

Operand isolation prevents unnecessary data propagation through inactive functional units. When a particular unit is not required, its input activity can be restricted. This technique helps reduce dynamic power consumption.

IV. PROPOSED LOW-POWER RISC-V DESIGN

The proposed design combines a basic RISC-V processor architecture with hardware-level power optimization techniques. The processor consists of an instruction fetch unit, decoder, register file, optimized ALU, control unit, memory interface, and clock management logic.

The clock management unit controls the clock signals supplied to different functional blocks. Functional units that are not required during a particular operation can be placed in an inactive state. The datapath is optimized to minimize unnecessary switching, while the memory interface is designed to reduce redundant data transfers.

The proposed architecture is suitable for low-power edge devices where computational requirements are moderate and energy efficiency is a major design constraint.

A. Proposed Processing Flow

The processor first fetches an instruction from memory. The instruction is then decoded and the required functional unit is activated. Operands are obtained from the register file and processed using the optimized ALU or another required unit. The result is stored in the destination register or transferred to memory. During this process, unused hardware blocks remain inactive whenever possible.

This approach aims to maintain processing capability while reducing unnecessary power consumption.

V. APPLICATIONS IN EDGE COMPUTING

The optimized RISC-V processor can be used in several edge computing applications.

A. Internet of Things

IoT sensor nodes require low-power processors because many devices operate from batteries or energy-limited sources. The proposed processor can perform local sensor processing while reducing energy consumption.

B. Wearable Devices

Wearable devices require compact and energy-efficient processing systems. A low-power RISC-V processor can perform sensor data processing locally without continuously transmitting data to a cloud server.

C. Smart Agriculture

Edge processors can be used in agricultural monitoring systems to process temperature, humidity, soil moisture, and other sensor data locally. This reduces communication requirements and enables faster responses.

D. Environmental Monitoring

Low-power processors can process data from air-quality, temperature, and other environmental sensors. Local processing reduces the amount of data that needs to be transmitted to remote servers.

E. Smart Home Systems

Smart home devices can use optimized processors for local control and sensor processing. Low power consumption is particularly useful for continuously operating smart sensors.

VI. PERFORMANCE AND POWER CONSIDERATIONS

Processor performance and power consumption must be considered together when designing an edge computing processor. Increasing computational performance may increase power consumption, while excessive power reduction can affect processing speed.

The proposed design focuses on achieving a suitable balance between power, performance, and hardware complexity. Clock gating and operand isolation can reduce dynamic switching, while datapath and memory optimization can reduce unnecessary operations. The modular nature of RISC-V also allows the processor to be customized according to application requirements.

The effectiveness of the design can be evaluated using parameters such as power consumption, operating frequency, execution time, hardware area, and energy per operation. These parameters can be compared with a conventional RISC-V implementation to determine the benefits of the proposed optimization techniques.

VII. ADVANTAGES

The proposed low-power RISC-V processor provides several advantages:

Reduced dynamic power consumption.

Flexible and modular processor architecture.

Reduced unnecessary hardware activity.

Suitable for resource-constrained edge devices.

Potential reduction in hardware complexity.

Local data processing with reduced communication requirements.

Suitable for IoT and embedded applications.

VIII. CHALLENGES AND FUTURE SCOPE

Although hardware optimization can reduce power consumption, achieving a suitable balance between power, performance, and area remains a major challenge. Clock gating and other optimization techniques may also introduce additional control complexity. The processor must therefore be carefully designed and verified.

Future work can focus on integrating hardware accelerators for machine learning and signal processing applications. Further optimization of cache memory, branch prediction, instruction scheduling, and power management can improve processor efficiency. FPGA and ASIC implementations can also be used to evaluate the practical performance, power consumption, and area of the optimized design.

IX. CONCLUSION

This paper presented a low-power RISC-V processor design approach for edge computing applications. Hardware optimization techniques such as clock gating, datapath optimization, memory access optimization, pipeline optimization, and operand isolation can reduce unnecessary switching activity and improve energy efficiency.

The flexibility of the RISC-V architecture makes it suitable for developing application-specific processors for IoT, wearable devices, smart agriculture, environmental monitoring, and other edge computing applications. By balancing power consumption, performance, and hardware complexity, an optimized RISC-V processor can provide an effective solution for resource-constrained edge devices.

ACKNOWLEDGMENT

The author would like to express sincere gratitude to the Department of Electronics and Communication Engineering for providing the necessary support and facilities to carry out this study.

REFERENCES

[1] A. Waterman and K. Asanović, The RISC-V Instruction Set Manual, RISC-V Foundation.

[2] D. Patterson and J. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Morgan Kaufmann.

[3] J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann.

[4] RISC-V International, "RISC-V Specifications," RISC-V International. [5] M. Horowitz, "1.1 Computing's Energy Problem (and What We Can Do About It)," IEEE International Solid-State Circuits Conference, 2014.

[6] N. E. Weste and D. Harris, CMOS VLSI Design: A Circuits and Systems Perspective, Pearson.

[7] ARM, Low Power Design Techniques for Embedded Systems, technical documentation