Residency Project: Designing and Implementing a Microprocessor Using gem5 Simulation Software

(Deliverable 1)

Group 3

Shashwat Baral, Samrat Baral, Pawan Pandey, Sujan Lamgade, Bhawesh Shrestha, Bijayata Shrestha, Sakchham Sangroula

University of the Cumberlands

Computer Architecture and Design (MSCS-531-M51)

Dr. Charles Lively

Phase 1: Architecture Definition and Design

This phase focuses on establishing the microprocessor architecture and integrating low-power design elements. This phase is dedicated to defining the microprocessor architecture while incorporating key low-power design techniques. By the end of this phase, a detailed report will be produced, outlining the strategies implemented to achieve this objective. The objective is to develop an energy-efficient processor suitable for the selected target application. The outcome of this phase is a comprehensive report detailing the implemented strategies.

Step 1: Define Microprocessor Architecture

The target application sets the requirements and design constraints of the processor, influencing architectural decisions as different applications emphasize various characteristics such as power efficiency, performance, and scalability.

Target Use Case: For this project, the target application is an Internet of Things (IoT) device. IoT devices typically require microprocessors that operate under stringent power limitations to ensure extended battery life while effectively handling sensor data and communication tasks.

Rationale: The prevalence of IoT devices in modern technology is rapidly increasing. Designing a processor with energy efficiency in mind is essential, as many IoT applications—such as smart home systems and wearable devices—demand reliable operation over extended periods with minimal energy usage. According to Yassin et al. (2021), energy-efficient processing is critical in edge computing scenarios commonly found in IoT applications. Advancements in x86 processors have led to low-power CPUs suitable for IoT applications. These processors offer high performance and compatibility with a wide range of software, making them ideal for IoT gateways that need robust computational capabilities (Yassin et al., 2021).

Architectural Features are essential for the microprocessor to follow options:

- ***Instruction Set Architecture (ISA)*** defines the instructions the processor is to execute. Regarding the architectures used, x86 ISA was chosen due to its wide use in general-purpose computing and the availability of low-power x86 processors. x86 processors provide better performance and very strong compatibility with the existing software ecosystems, which are helpful for IoT gateway applications(Reddy et al., 2017).
- ***Pipeline depth*** is a 7-stage pipeline, which is fetch, decode, micro-op translation, register renaming, execute, memory access, and writeback stages. This setup allows higher instruction throughput while maintaining manageable complexity and energy consumption (Qureshi et al., 2019).
- ***Cache Hierarchy*** is a tiered cache architecture developed to reduce the energy expenditure associated with memory access. Sum up all hierarchies consist of:
    a. The *L1 Cache* is a small, high-speed cache directly linked to the processor cores, designed for optimal speed. The L1 cache features a capacity of 32 KB with a 4-way set-associative arrangement for both instruction and data caches.

b. The *L2 Cache* is a more extensive, slower cache intended for storing frequently accessed data exceeding the L1 cache capacity. The L2 cache is 256 KB and utilizes an 8-way set-associative configuration.

c. The *L3 Cache* enhances performance and reduces energy consumption from main memory access, we introduce a shared L3 cache among all processor cores. With a capacity of 2 MB and a 16-way set-associative configuration, the L3 cache serves as the final cache level before main memory, significantly decreasing memory latency and energy use by minimizing main memory accesses (Qureshi et al., 2019).

Power Management Features are essential for the microprocessor to follow options:

- ***Dynamic Voltage and Frequency Scaling (DVFS)*** capability enables the processor to adjust its voltage and frequency in response to the demands of the processes it needs. Reducing the frequency during less demanding operations, DVFS can lead to substantial energy savings (Yassin et al., 2021).
- ***Clock Gating*** is a technique that involves disabling certain processor sections when not actively in use, lowering dynamic power consumption (Reddy et al., 2017).

Justification for Architectural Choices

We chose the x86 ISA due to its compatibility with a wide range of software and the availability of low-power x86 processors suitable for IoT gateway devices (Yassin et al., 2021). x86 architectures provide the computational power needed for processing and aggregating data from multiple IoT devices.

Though power dissipation increases with performance and the computational demand of the IoT gateway also tends to increase, implementing a 7-stage pipeline proves to increase instruction throughput and balances increased complexity and power consumption constraints (Qureshi et al., 2019). Introducing a multi-level cache hierarchy has lower power and latency costs when trying to ensure that frequently used data is available quickly, thus enhancing the overall energetic efficiency of the design (Reddy et al., 2017).

DVFS and clock gating allow dynamic power consumption adjustment based on workload demands, which is essential for energy-constrained environments. Using gem5 simulation software enables detailed architectural exploration and power modeling. Reddy et al. (2017) showed that gem5 can be extended with empirical CPU power modeling to estimate power consumption accurately, which is crucial for evaluating the impact of low-power design techniques.

Moreover, Qureshi et al. (2019) introduced Gem5-X, a gem5-based system-level simulation framework optimized for many-core platforms. Gem5-X extends gem5's capabilities by providing support for advanced architectural features, beneficial for IoT applications requiring efficient data processing and low energy consumption.

Step 2: Design Low-Power Features

Energy efficiency is crucial in designing microprocessors for embedded systems and IoT devices. To enhance power management, the following low-power features are integrated into the design:

Dynamic Voltage and Frequency Scaling (DVFS) is a technique aimed at conserving power by dynamically adjusting the processor's voltage and clock frequency in response to workload (Yassin et al., 2021). During periods of light workload, both voltage and frequency can be reduced, leading to lower power consumption while maintaining acceptable performance levels.

- ❖ ***Implementation****:* In the gem5 simulation environment, DVFS can be modeled by modifying the processor's clock frequency and voltage levels in real-time, contingent upon the workload. Yassin et al. (2021) proposed an efficient implementation of DVFS in gem5 using a System called Emulation mode, which allows fast and accurate energy modeling in edge computing scenarios common in IoT devices.

- ❖ ***Benefits***: Implementing DVFS can result in substantial power savings, particularly in IoT devices that experience extended periods of inactivity or minimal processing demands. This technique effectively balances performance and energy efficiency, critical for prolonging battery life in portable devices.

Clock Gating is employed to decrease power consumption by deactivating the clock signal to certain processor sections that are not in active use (Reddy et al., 2017). For instance, if a functional unit is unnecessary for the current instruction, its clock signal can be halted, effectively reducing power usage.

- ❖ ***Implementation:*** In the Gem5 simulator, clock gating can be represented by managing the activity of specific components according to instruction usage patterns. The clock signal is directed only to essential functional units during each operational cycle. Reddy et al. (2017) demonstrated that incorporating clock gating into the Gem5 power model enhances the accuracy of power estimation, providing more reliable results for architectural exploration.

- ❖ ***Benefits:*** This technique significantly lowers dynamic power consumption, especially in architectures where not all units are needed simultaneously. By reducing unnecessary switching activities, clock gating effectively decreases overall power dissipation.

Power Gating involves completely shutting down certain processor sections during inactivity, effectively minimizing leakage power—a considerable contributor to energy loss in modern processors.

- ***Implementation:*** Power gating is utilized for cores or caches that remain unused for prolonged durations. The Gem5 simulator can be configured to replicate this functionality by deactivating inactive cores and memory segments. Qureshi et al. (2019) incorporated power gating mechanisms in Gem5-X to optimize energy consumption in many-core platforms.

- ***Benefits:*** Power gating offers substantial energy savings when processor components are dormant for extended periods, such as when the device transitions into a low-power or sleep mode. This technique is vital for reducing static power consumption in deep submicron technologies.

Rationale for Low-Power Design Strategies

These low-power strategies are grounded in contemporary microprocessor practices. DVFS facilitates dynamic power management, allowing adaptation to varying performance requirements (Yassin et al., 2021). Clock gating and power gating significantly enhance energy efficiency by minimizing dynamic and static power consumption, respectively (Reddy et al., 2017; Qureshi et al., 2019).

Using gem5 and its extensions, like Gem5-X, provides a robust simulation environment for evaluating these low-power techniques in detail. The support for x86 architectures allows for accurate modeling of performance and power consumption (Reddy et al., 2017). Gem5-X extends gem5's capabilities by enabling architectural exploration of many-core systems with advanced features, particularly relevant for optimizing power consumption and performance in IoT devices (Qureshi et al., 2019).

References

Qureshi, Y. M., Simon, W. A., Zapater, M., Atienza, D., & Olcoz, K. (2019). Gem5-X: A gem5-based system-level simulation framework to optimize many-core platforms. 2019 Spring Simulation Conference (SpringSim), 1–12. https://doi.org/10.23919/SpringSim.2019.8732862

Reddy, B. K., Walker, M. J., Balsamo, D., Diestelhorst, S., Al-Hashimi, B. M., & Merrett, G. V. (2017). Empirical CPU power modeling and estimation in the gem5 simulator. 2017 27th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS), 1–8. https://doi.org/10.1109/PATMOS.2017.8106988

Yassin, Y. H., Jahre, M., Kjeldsberg, P. G., & Aaberge, M. (2021). Fast and accurate edge computing energy modeling and DVFS implementation in gem5 using system call emulation mode. Journal of Signal Processing Systems, 93(1), 33–48. https://doi.org/10.1007/s11265-020-01544-z