Architectural Analysis and Optimization for Sustainable Community Wi-Fi Edge Nodes

Introduction

Community Wi-Fi edge nodes are a critical infrastructure for routing, educational caching and intrusion detection in remote regions. This scenario needs a rigorous computer architecture analysis to balance high-performance memory and concurrency requirements as power source is limited. This research evaluates two hardware systems to propose a sustainable system optimized for the community's needs.

Background and Related Work

Memory Hierarchy and Throughput

Efficient memory hierarchy is crucial for performance in Wi-Fi edge nodes especially in caching and IDS works. System A is designed to be optimized for reusing local data as it contains a 1 MB L2 cache for its two processor cores. While System B has 512 KB L2 cache which is distributed among its four cores. In Computer Architecture, larger caches is very important as it can significantly reduce main memory access by absorbing most L1 cache misses (Patterson & Hennessy, 2020). Again, Anderson (2025) further points out that larger L2 caches for example 128KB to 1MB can also notably improve throughput of the system workloads with keeping the data close to CPU. However, System A's 512 MB DDR3 memory capacity is half the capacity of System B's 1GB LPDDR4 main memory. Despite System A having a larger cache, modern IDS software like Snot3 will crash immediately with an "Out Of Memory" error message on this system (faux123, 2020). Furthermore, LPDDR4 is known to operate in lower voltages like DDR3 memory's equivalent 70% active power and 10% of the standby power (Uppala, 2025). So, despite having a larger cache the limited memory of System A poses a high risk of system failure as the workload can exceed 512 MB in heavy workloads.

Concurrency and Multi-Threading

When it comes to computing concurrency the proposed systems are fundamentally different. The powerhouse of system A is a dual-core RISC-V CPU, whereas System B has four ARM Cortex-A53 cores to compute. Parallel processing greatly increases performance for multi-threaded systems which combine video caching, IDS analysis and network routing. System A's two cores are limited to CPU-bound tasks if forced to time-share between routing and packet inspection which heavily depends on CPU. Modern IDS softwares like Snort3 and Suricata are suitable for multi-threaded workloads to scale with CPU cores of the systems (Boukebous et al., 2023). Hence, System B has significantly higher packet processing throughput for network routing. Due to its small memory running multi threads on System A will lead to its eventual memory exhaustion. Since System B has 1 GB RAM it has sufficient local memory for its four cores to work comfortably.

Storage Capacity and Reliability

Caching educational videos heavily depended on local storage. The System A contains only 128 MB of NOR flash storage which is very low compared to System B's 64 GB eMMC storage module. This difference is huge because typically for firmware and configuration files to store system needs at least 128MB, leaving no room for video caching for System A (*[OpenWrt Wiki] Recommended Routers*, n.d.). Conversely, System B's larger storage capacity allows it to cache hundreds of hours of educational content. Reliability of these storage systems is also a crucial factor. System A's NOR flash has low density but is robust for faster code execution. Despite having high endurance per cell, the small 128 MB storage module would be burnt out if the system writes log files repeatedly to the same sectors without a proper file system. Contrary, System B's lower raw endurance comes from NAND flash which eMMC uses. However, to extend their lifetime eMMC has a managed controller that implements wear leveling. Kingston Technology (2020) points out that a 4GB eMMC can handle approximately 1.5 TB of total writes. This means the 64 GB module in System B has an over 5 year life cycle. Thus, System B's storage system is best suited for the scenario.

Methodology

After thoroughly reviewing the both systems we can conclude that neither of the systems are perfect for the given scenario. So, a hybrid system architecture is needed that combines the best features of both system A and system B.

Hybrid Hardware Design

The hybrid system will use a heterogeneous multi-core SoC. The architecture will be similar to ARM's bit.LITTLE. This choice is perfect because it would bring high-efficient cores for background tasks such as logging, routing and intensive tasks including IDS scanning will use the high-performance cores. It gives the proposed system the flexibility to maintain concurrency without the continuous power drain of running four high power-cores at once. The proposed system will use memory to a minimum of 1GB LPDDR technology. The LPDDR technology offers higher bandwidth and lower power consumption making it suitable for harsh temperature conditions (Uppala, 2025). And for its storage module the system replaces 128MB NOR with a NAND flash for better life cycle. To store the most frequently used small resources in the storage, a selective efficient caching algorithm should be implemented for faster access (Cho & Bahn, 2020). The system will use a smaller 8GB eMMC that will improve caching capabilities compared to System A also while supporting wear leveling and retaining a small NOR partition strictly for bootloader reliability.

Figure 1: Proposed Hybrid Edge Node Architecture & Data Flow

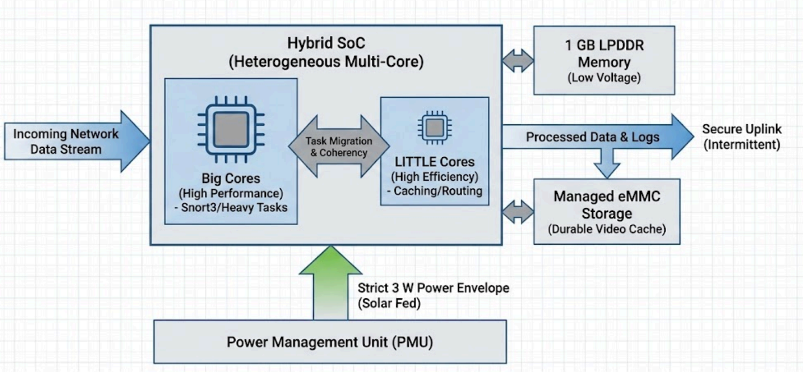

Figure 1: Proposed Hybrid Edge Node Architecture & Data Flow

Discussion

The proposed hybrid system greatly balances the high performance required for intrusion detection without draining resources. However, this proposed system also introduces a specific trade off in software complexity compared to simpler System A. To maintain the strict 3 W power envelope, the system requires an efficient power saving task scheduler that restricts heavy computation during solar peak time. This adds a layer of development difficulty to ensure reliability.

Regarding sustainability, off-grid deployments depend on both energy efficiency as well as system longevity. Despite System A being energy efficient (25 Wh/day), its hardware limitations make it obsolete. Conversely, System B requires larger power infrastructure for its energy consumption (55Wh/day). ATMECS (2024) points out that "Green Computing" must extend device lifecycles to reduce e-waste. The five-year operational term with capable storage, the hybrid system ensures that the system balances digital inclusion with minimal environmental impact.

Conclusion

This study establishes that a hybrid SoC architecture optimally balances performance with the strict energy constraint off-grid edge nodes. This design ensures a five year operation lifecycle with a 3W power envelope while significantly reducing e-waste. The system also ensures to meet all requirements. Future work should focus on implementing power saving task scheduling to align heavy computations with peak solar energy production.

References

Anderson, M. (2025). *The Cache Clash: L1, L2, and L3 in CPUs*. https://medium.com/@mike.anderson007/the-cache-clash-l1-l2-and-l3-in-cpus-2a21d61a0c6b

Atmecs, T. (2024). *Green Computing: Adopting Eco-Friendly IT Practices in Technology Firms*. https://atmecs.com/green-computing-adopting-eco-friendly-it-practices/

Boukebous, A. a. E., Fettache, M. I., Bendiab, G., & Shiaeles, S. (2023). *A Comparative Analysis of Snort 3 and Suricata. 2023 IEEE IAS Global Conference on Emerging Technologies (GlobConET)*, 1–6. https://doi.org/10.1109/globconet56651.2023.10150141

Cho, K., & Bahn, H. (2020). *An efficient cache management algorithm for streaming workloads*. https://dl.acm.org/doi/abs/10.1504/ijnvo.2020.107574

faux123. (2020). *[Experimental] Snort3 IDS/IPS on AsusMerlin [AC86/AX88 routers ONLY]*. https://www.snbforums.com/threads/experimental-snort3-ids-ips-on-asusmerlin-ac86-ax88-routers-only.66123/

Kingston Technology (2023, January 9). *EMMC Life-Cycle Estimating, Validating & Monitoring*. https://www.kingston.com/en/blog/embedded-and-industrial/emmc-lifecycle

[OpenWrt Wiki] Recommended routers. (n.d.). Openwrt.org. https://openwrt.org/toh/recommended_routers

Patterson, D. A., & Hennessy, J. L. (2020). *Computer Organization and Design RISC-V Edition: The Hardware Software Interface.* https://dl.acm.org/citation.cfm?id=2568134

Uppala, R. (2025). *LPDDR: A Versatile Memory Powering The Next Wave Of Mobile, Edge & Endpoint Computing.* https://semiengineering.com/lpddr-a-versatile-memory-powering-the-next-wave-of-mobile-edge-endpoint-computing/