Scientific Publications

Journal of Artificial Intelligence and Big Data, 2023, 3, 1395
www.scipublications.org/journal/index.php/jaibd
DOI: 10.31586/jaibd.2023.1395

Review Article

Recent Advances in Mobile SoC: Design for Performance and Power Efficiency

Javed Ali Mohammad [1,*], Sri Harsha Panchali [2], Usha Mohani Kavirayani [3], Krishna Bhardwaj Mylavarapu [4], Jenitha Pilli [5], Prathik Kumar Jannu [6]

[1] Masters in Telecommunications, Middlesex University, USA
[2] Information Systems Engineer, CrowdStrike Inc, USA
[3] Kent State University, MS in Computer Science, USA
[4] MS in Computer Science, University of Illinois Springfield, USA
[5] MS in Computer Science, University of Louisiana at Lafayette, USA
[6] Computer Science Engineering, JNTU Hyderabad, USA

*Correspondence: Javed Ali Mohammad (Javed.bw@gmail.com)

Abstract: The recent development of Mobile System-on-Chip (SoC) has completely changed the functionality of the contemporary smartphones, wearables, and IoT devices through enhanced computational strength but with stringent power and heat constraints. Mobile applications are becoming more complex (e.g., real-time computer vision, high-resolution imaging, on-device artificial intelligence, etc.) and SoCs have to provide high throughput, good work load distribution, and responsive multitasking without sacrificing battery life. To overcome these issues, modern SoCs are based on heterogeneous multicore designs, with big. LITTLE designs, special purpose accelerators, like NPUs and ISPs, and memory arranged hierarchies and backed with high bandwidth buses. Also, more sophisticated power-management approaches like Dynamic Voltage and Frequency Scaling (DVFS), clock gating, and power gating provide fine-tuning energy management in the active and idle conditions. Most recently, scheduling and thermal sensitivity resource allocation using AI has become a promising approach in increasing real-time refinement and flexibility. This paper gives a detailed discussion of performance-driven and power-efficient design methods in mobile SoCs, including architectural innovations, workload optimization, and built-in AI-assisted energy management. The discussion identifies major research trends, system-level developments, and new challenges defining next-generation mobile computing systems.

Keywords: Mobile SoC, Power Efficiency, Performance Optimization, Heterogeneous Computing, Network-on-Chip (NoC), AI Hardware Accelerators

How to cite this paper:
Mohammad, J. A., Panchali, S. H., Kavirayani, U. M., Mylavarapu, K. B., Pilli, J., & Jannu, P. K. (2023). Recent Advances in Mobile SoC: Design for Performance and Power Efficiency. *Journal of Artificial Intelligence and Big Data, 3*(1), 125-136. DOI: 10.31586/jaibd.2023.1395

Received: September 22, 2023
Revised: November 21, 2023
Accepted: December 12, 2023
Published: December 27, 2023

1. Introduction

Computing has entered lives and the mobile devices are the most common platform through which people gain access to applications, communication and intelligent services. Smartphones, tablet devices, wearables, and IoT devices are all based on the System-on-Chip (SoC) technology to provide small and powerful solutions in computing [1]. A System-on-Chip (SoC) package is a bundle of a variety of functional units, such as CPUs, GPUs, NPUs, ISPs, DSPs, memory controllers, modems, and others, which are implemented on the same chip platform and are required to operate under intense workloads and within a set of energy and thermal constraints [2]. The SoCs in a mobile environment have applications spanning operating systems and multimedia processing, to high-end applications, including AI-based vision, real-time speech recognition, and 5G

connectivity [3]. They still remain relevant as users demand low latency and high efficiency seamless edge computing experiences that are increasingly in demand.

Mobile SoC design is a major challenge where high performance and low-energy consumption is required. Mobile devices are limited in battery capacity, small form factors and thermal comfort. Peak computational throughput is appreciable, but when it can be maintained in sustained use without high battery drain or overheating, then it is important as well [4]. The users now require immersive graphics, multitasking responsiveness, and smart services without sacrificing battery life, and thus power efficient SoC design has become a major enabler of devices innovation. A number of issues make this balance complicated. Due to transistor scaling, which is approaching the physical limits, power density has been an increasing issue [5]. The heat loss in the small enclosures limits the long operating capability, and the heterogeneous components integration requires advanced memory hierarchies and interconnect designs. The tradeoff between requirements on compute power, real-time connectivity and AI acceleration needs new solutions both at the hardware design and the system level.

Recent technological advances in mobile SoC technology have emerged specifically in response to these challenges. Dynamic Voltage and Frequency Scaling (DVFS) allows processors to adjust their power and speed to match the workload, saving energy when the workload is light. Intelligent distribution of tasks is achieved by heterogeneous multi-core architectures like big.LITTLE, which distributes tasks between high-performance and power-efficient cores. Hardware accelerators such as NPUs and specialized AI units offload the most performance-intensive tasks to enhance performance-per-watt [6]. Improvements in memory subsystems and Network-on-Chip interconnects also enhance throughput and latency, ensuring seamless communication between components. The combination of these innovations reflects the shift towards more holistic, energy-aware SoC designs that allow today's mobile devices to strike a balance between speed, efficiency, and user experience.

Structure of the Paper

The structure of the paper is as follows: Section II provides the description of the basic elements of the modern mobile SoCs such as CPU, GPUs, NPU, and memory integration. Section III examines the recent performance optimization strategies. IV explains how to make power efficient like DVFS, clock gating, and thermal management. Section V give a literature analysis of comparison and Section VI give the insights and future research directions in mobile SoC design.

2. Architectural Design of Mobile Socs

Mobile System-on-Chip (SoC) architectures incorporate a number of functional units, processors, memory, graphics and communication units in to a single small package and are optimized in terms of performance, energy efficiency and thermal management. Dynamic workload allocation is made possible by having heterogeneous processing provided by modern mobile SoCs, usually consisting of high-performance cores and power-efficient cores. The memory subsystem with on-chip caches and DRAM controllers is able to provide high data throughput with low latency [7]. Fast interconnects are used to provide any type of smooth communication between components, and special accelerators are used to deal with AI inference, multimedia processing, and security.

2.1. Mobile SoC Architecture

A current Mobile System-on-Chip (SoC) may include a number of heterogeneous processing units on a single semiconductor die, striking a compromise between high performance and severe power and thermal constraints. General-purpose workloads are executed by the Central Processing Unit (CPU), and they tend to be based on either ARM

high-performance cores using big.LITTLE or DynamIQ architecture, and power-efficient cores using background tasks.

The Graphics Processing Unit (GPU) is used to perform user interface, 3D gaming and heavy computational loads like image processing and neural network inference [8]. The NPU is responsible for AI workloads like vision, speech recognition, and NLP, while the ISP handles activities related to the camera in real-time, such as noise reduction, HDR fusion, and depth estimation [9].

Digital signal processor (DSP) is an efficient processing unit of audio, sensor, and baseband modem, whereas integrated modem (4G/5G) is an RF front-end engine and a channel coding engine. Cryptographic accelerators and trusted execution environments (TEE), security subsystems are used to safeguard data and integrity of a device [10]. As shown in Figure 1, a typical mobile SoC block diagram has clusters of CPUs, clusters of GPUs, DSPs, ISPs, modems, memory controllers, multimedia engines, and peripheral interfaces connected together by system, multimedia, and peripheral fabrics. These interconnects facilitate high bandwidth data transfer as well as coordinated action between components.

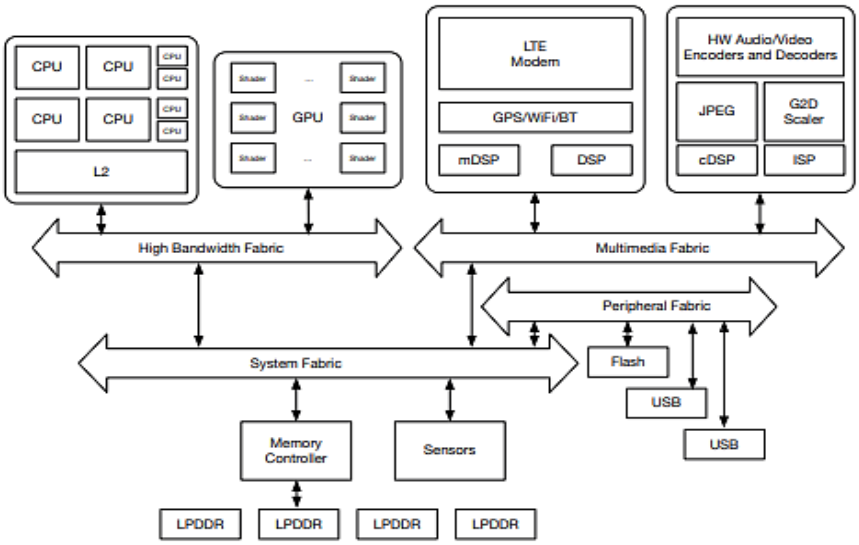

Figure 1. Block Diagram of a Typical Mobile Soc Architecture [11]

2.2. Memory and Interconnect Infrastructure

The efficiency of the interconnect architecture and memory subsystem of a mobile SoC is as critical to the performance of the mobile SoC as the efficiency of its compute units. Mobile SoCs normally adopt Low power dual data rate (LPDDR) DRAM. LPDDR5 standard has more bandwidth and better power characteristics than LPDDR4/4X. These utilize the use of highly signaled modes, deep power off, and adaptive refresh to reduce the leakage power and preserve data integrity [12].

The cache hierarchies in the SoC decrease the off-chip memory access latency. L1 and L2 caches are co-located with CPU cores whereas shared L3 caches provide heterogeneous processor coherence. Coherency Cores see the most current data, and write-through updates the memory on-the-fly, and when updates are to be delayed, write-back updates the cores, to keep the traffic down [13]. Caches make use of both temporal (reuse of recent data) and spatial (access to nearby data) localities, caching data in fixed-size cache lines.

The CPU, the GPU, the NPU, the ISP and the modem are connected with high speed interconnects, usually of the Network-on-Chip (NoC) type. They use packet-switched routing, adaptive flow control and quality-of-service, and compression, prefetching and dynamic link-width scaling support bandwidth without unnecessary power consumption.

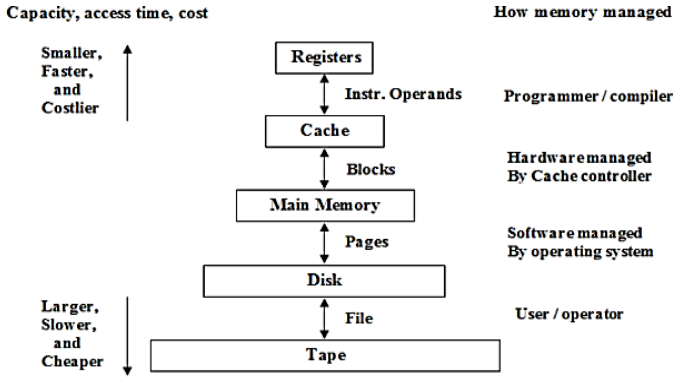

Figure 2. Memory Hierarchy and Management Levels [14]

Figure 2 illustrate the memory hierarchy, with the fastest, and smallest, CPU registers on the top, caches and main memory, and more sluggish, but larger, secondary and tertiary storage. This hierarchical design coupled with the coherent caching policies allows mobile SoCs to provide high throughput with the minimum of latency and power consumption.

2.3. Performance and Power Characterization Metrics

A mobile SoC needs to be evaluated on a multidimensional basis beyond simple performance metrics. The evaluation of performance, efficiency, and thermal performance needs to be measured in a way to showcase the capability of the hardware and its actual performance when it comes to user experience. The measures of such assessments are the following:

- **Throughput:** The unit of measurement changes depending on the workload type and can be giga-or tera-operations per second (GOPS), or frames per second (FPS). It is a measure of the unfinished computational power of CPU, GPU, NPU, and ISP pipelines [15].
- **Efficiency (Performance-per-Watt):** This is a vital parameter of battery-powered devices that is defined as the computational throughput divided by mean power consumption. Increased efficiency means a better utilization of energy to have long-term performance.
- **Thermal Limits:** Evaluated through thermal design power (TDP) envelopes, hotspot analysis, and thermal throttling behavior under sustained workloads. This makes sure that the temperatures on the surface of devices are kept within the comfort limits of users and not cause any degradation to the hardware [16].
- **User-Perceived Performance Indicators:** This consist of application launch times, responsiveness of UI and frame stability when interacting with it. When creating synthetic benchmarks, raw numbers are used but in many cases, the end-user experience is closer to these perceptual measures.

Together, these measures give a global view of an operational profile of a mobile SoC. Whereas throughput signifies optimum computational power, efficiency and thermal constraints establish the duration through which the high performance may be maintained without affecting usability.

3. Performance Optimization Techniques in Mobile Socs

In recent mobile System-on-Chip (SoC) design, numerous performance optimization methods are combined to create an optimal balance between processing speed, energy efficiency and thermal constraints [17]. It has a combination of exposure and architectural techniques to balance the performance and energy efficiency as well as thermal restriction.

Dynamic voltage and frequency scaling, heterogeneous multi-core designs, and hardware acceleration for AI workloads are some of the solutions.

3.1. Dynamic Voltage and Frequency Scaling (DVFS)

DVFS is an extremely common method in embedded, laptop, desktop, and high-performance computer systems, and is aimed at balancing power usage and performance [18]. The system is able to adapt to the demands of the current workload by dynamically adjusting the processor's operating frequency and supply voltage [19]. Equation (1) can be used to calculate the power consumption of a CMOS integrated circuit, e.g. a mobile processor:

$$P = CfV^2 + P_{static} \quad (1)$$

The operating frequency (f), supply voltage (V), and transistor gate capacitance (C) are all variables that are dependent on feature size. This enables efficient adaptation of processing performance to workload demands. While DVFS dynamically adjusts the processor's power and performance, heterogeneous multi-core architectures address workload distribution at the architectural level.

3.2. Heterogeneous Multi-core Architectures

In heterogeneous multi-core architectures, the cores are of a different type and placed within the same SoC to provide the optimal performance and power efficiency ratio. High-performance cores are usually targeted at tasks that are intensive in computing which include game design, multimedia design and intricate calculations [20]. Conversely, cores that are energy efficient are designed to support light weight, periodic, or background loads, e.g. messaging, sensor watching, or simple application processes, and have small power requirements.

Heterogeneous architectures reduce a waste of energy but preserve system responsiveness by allowing the operating system to smartly schedule the workloads according to their computational needs [21]. This dynamic assignment of tasks guarantees that every process performed on the most appropriate type of core, which increase battery life, be thermally stable, and overall give a more relaxed performance to mobile and embedded systems like big.LITTLE In heterogeneous designs, the LITTLE design is an example of a viable strategy in balancing the high-performance and low-power processor.

3.2.1. Big.LITTLE Multicore Architecture

The big.LITTLE is a processor architecture design approach that integrates two kinds of CPU cores into the same SoC: the high-performance, performance-critical, so-called big, and the power-efficient so-called LITTLE, cores. This arrangement enables the system to dynamically switch between cores or distribute workloads across them, depending on processing demands and power considerations. This can be done through intelligent distribution of tasks, which enables big.LITTLE to provide high performance where required and accounts to a much lower power consumption when the mobile device is operating in a low intensity manner [22]. It is especially significant in the case of smartphones, tablets as well as embedded systems, where thermal constraints and battery life matter.

The system analyzes the type of incoming tasks and sends them to the correct core type as shown in Figure 3. Instances of intensive computational workloads allocated to the big cores in order to maximize their performance, and routine or background tasks allocated to the LITTLE cores to save on energy. This active load sharing is done according to this to make sure that performance demands are achieved without needless power loss.

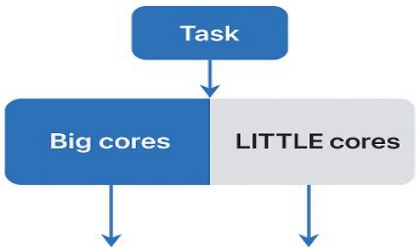

Figure 3. Task Allocation in Big.LITTLE Multicore Architecture

The key advantages include:

- Increases the battery life through low-power cores in the event of simple tasks.
- Stable at high workloads in terms of thermal.
- Improves the experience of the user by supporting multitasking.
- Conforms to real time adjustments in the workload without performance setback.

The big.LITTLE architecture is a viable solution to the problem of trade-offs between high performance and energy efficiency, and is a key ingredient in the current mobile SoC design.

3.3. Hardware Acceleration Methods for Embedded AI

With the growing use of AI in mobile devices, SoCs are now being equipped with specialized hardware accelerators to perform tasks with high computational complexity. These techniques first of all dramatically increase the throughput, latency, and power efficiency of general-purpose CPU by offloading compute-intensive tasks in specialized hardware units [23]. There are three main types of hardware that are used to perform embedded AI acceleration:

- **Field Programmable Gate Array (FPGA):** FPGAs are fully general and can be reconfigured to map any kind of neural network, allowing them to be optimized to meet particular throughput, latency and power warnings. They are flexible and can be tested on prototypes and in different applications of mobile AI.
- **Application Specific Integrated Circuit (ASIC):** Purpose-built for a particular AI workload, ASICs offer maximum performance-per-watt by exploiting parallelism and minimizing computational overhead. While less flexible than FPGAs, they achieve superior efficiency for mass-produced mobile AI solutions [24].
- **Graphics processing unit (GPU):** AI inference and training uses massively parallel processing with many computers running in parallel, which are called GPUs. They are integrated in mobile SoCs and can process images in real-time, detect objects and can be used in advanced vision-based applications with low latency.

With the strategic application of these accelerators, mobile SoCs can enable the addition of more complex AI capabilities without a negative impact on battery life or thermal stability, and therefore achieve better performance-per-watt.

4. Power Efficiency Strategies in Mobile Socs

The goal of power-efficient mobile SoC design is to strike a balance between power consumption and the performance level that is needed. This is critical in extending the battery life, thermal stability, and high-performance working loads in portable devices.

To achieve these purposes, Modern SoCs use a mix of proven low-power design methods with highly innovative AI-based approaches.

4.1. Advanced Low-Power Techniques

Minimizing performance per watt without sacrificing functionality is crucial in modern system on chip (SoC) architecture when considering static and dynamic power consumption. Two popular methods are used, which include clock gating and power gating, which are essential in realizing this objective.

Clock Gating: When a clock signal is flipped off, clock gating is able to save dynamically the power used by the flip-flops, registers, or memory elements that are not involved in the current calculations. This stops superfluous switching behavior, which eradicates speculative switching in idle components [25]. Figure 4 shows a simplified model of a clock gating architecture in which an enable signal is used to select whether or not the clock is passed to a destination block therefore eliminating the switching of idle logic. It is a cost effective solution and can save up to 20-40% of total dynamic power with little reduction in performance. Clock gating has been one of the most popular low-power approaches to SoC design since clock-based power can frequently consume more than 60% of the overall dynamic power.

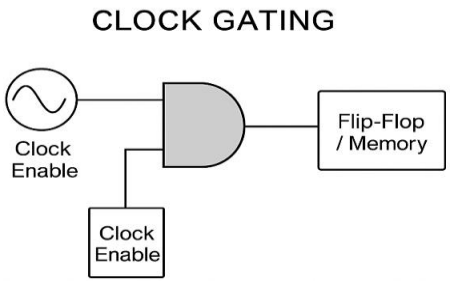

Figure 4. Concept of Clock Gating in Digital Circuits

Power Gating: The concept of power gating relies on disabling the power to blocks of flip-flops, memory elements and other components of a system when idle by interrupting the power supply, e.g. during standby or sleep. It is also applicable in an active mode in reducing dynamic and static power by blocking leakage and switching currents in idle logic [26]. Typical power gating scheme is illustrated in Figure 5 and involves Header and Footer FETs to break the connection between idle circuit blocks and the power rails to help reduce leakage currents. Power switches are controlled by power switches, which are usually Header FETs (connected to Vdd) and Footer FETs (connected to ground), power switches either on or off. The idle state of these transistors lowers the leakage power - typically 30-40% of the active power in active mode - which is often very important in minimizing energy use of modern SoCs.

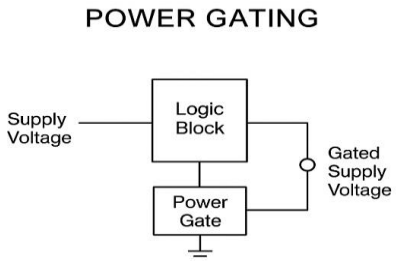

Figure 5. Power Gating in Digital Circuit Design

Clock gating along with power gating ensures that only active components consume energy, and this is the foundation of energy-conscious SoC architecture.

4.2. AI-Enhanced Energy Optimization

As mobile workloads become progressively complex, the conventional low-power solutions based on the clock gating and power gating become inadequate to address the performance and energy demands of the current devices. The mobile apps are growing in real-time processing, AI calculations, multimedia rendering, and background services that are dynamically changing with time. Here, the AI-based power management has become a potent solution, with dynamic and predictive optimization that enables SoCs to dynamically and intelligently set the energy consumption according to the real-time operation data and usage trends.

- **Predictive Workload Management:** AI algorithms process past and real-time usage data to predict the demand of calculations of various subsystems [27]. AI can also efficiently allocate resources in advance and minimize energy waste through predicting the active cores, functional units, or accelerators [28]. This does not over-provision and power is provided exactly where it is consumed, minimizing the overall system energy usage.
- **Thermal-Aware Scheduling:** Thermogenesis is a major issue of miniature mobile devices. The scheduling based on AI allocates smartly workloads between cores and accelerators to assure of safe operating temperatures. The system able to throttle-free and have consistent performance, as well as eliminate excessive energy consumption due to overheating, by avoiding hotspots and thermal spikes.
- **Context-Aware Energy Saving:** Mobile and hybrid computing platforms are more advanced devices where AI can be applied to utilize most of the interaction with various energy sources, like a battery, energy harvesting module, or adaptive power input. It assists in attaining sustainable utilization of energy and prolonging the livespan of gadgets by controlling the programs of workloads and anticipating supply of renewable or cheap power, as per AI [29].
- **Integration with Energy Sources:** In more sophisticated mobile and hybrid computing platforms, AI can be used to make best use of the interaction with different energy sources, such as a battery, energy harvesting module, or adaptive power input. It helps to achieve sustainable use of energy and extend the lifespan of devices by managing the schedules of workloads and predicting the availability of renewable or low-cost energy, according to AI.

The AI-based optimization of power usage minimizes energy wastefulness, as it constantly monitors system use and processes the result patterns to forecast future needs and sustain the performance as smooth and responsive.

4.3. AI Hardware Accelerators

Deep neural networks and real-time inference AI workloads demand huge computational power which cannot be efficiently delivered by traditional processors. As a solution to it, mobile SoCs are increasingly adding dedicated AI accelerators, i.e., dedicated units designed to execute high-throughput AI operations, allowing processing on-demand, minimizing energy and latency. The main design characteristics are:

- **High parallelism:** Multiply-Accumulate (MAC) operations are performed by hundreds and thousands of cores in parallel, which is highly beneficial in terms of throughput capability presented to image recognition and speech processing [30].
- **Near-Memory Processing:** Computations are done near to storage of data and thus this saves on energy-consuming memory transfers and it also minimizes latency.
- **High Memory Bandwidth:** Offers lightning-fast data transfers between RAM and CPUs, keeping cores occupied and preventing bottlenecks.

- **Energy Efficiency:** Optimized circuits, algorithm- specialized hardware, precision scaling and non-volatile memory integration make it less energy consumptive to run AI tasks and therefore makes AI tasks more sustainable.

Through these, AI hardware accelerators enable mobile devices to support complex AI workloads with high efficiency but with low power consumption and responsive performance.

5. Literature Review

Recent publications discuss heterogeneous mobile SoC designs, thermal analysis, CPU-GPU co-processing, machine learning pipeline design, and secure parallel processing, and show that there are increasing challenges in balancing the trade-offs between performance and energy, and integration complexities.

BenSaleh, Qasim and Obeid (2019) present the design of a more advanced system on chip for handheld devices. Using the provided architecture, also can link to a host processor that is compatible with ARM, high-speed IP cores, and slower peripherals using the industry standard advanced microcontroller interface. Clock generators, real-time clocks, watchdog timers, interrupt controllers, programmable I/O, I2C hosts, SPI masters, UARTs, trusted platform modules, NAND flash controllers, and USB controllers are all part of the system's standard set of peripherals. Mobile computing devices can have a complete platform architecture thanks to third-party 2D/3D graphics engines, audio/video encoders/decoders, and wireless network controller Ips [31].

Zhu, Mattina and Whatmough (2018) refer to the increased role of machine learning in mobile AR/VR and ADAS applications. They note that although hardware architectures have designed specialized hardware to implement machine learning algorithms to make them compute more efficiently, there is a trend towards using only ML accelerators. Such a small area of consideration ignores the more extensive optimization opportunities at the SoC level. The authors promote the idea of optimization at the entire SoC level, using a case study of continuous computer vision as an example, in which the coordination and design of camera sensors, image signal processors, memory, and NN accelerators are to reach optimal system-level efficiency [32].

Kuo et al. (2018) Stress that the SoC is usually thought of as a single source of heat when smartphones are modeled thermally. They present a DCTM, or dynamic compact thermal model, to account for the CPU's and GPU's separate and combined thermal behavior; this is necessary for power and performance optimization when operating at thermal limitations. The DCTM incorporates the transient behavior of peripheral ICs, like power management ICs, into its fundamental architecture, which is based on the standard RC ladder models. Experiment validation in an actual phone housing revealed an error of up to less than 1.58 C. Also, the research found that a refined power allocation policy could result in an 8 percent rise in power budget [33].

Kumaki, Koide and Fujino (2017) suggests using a massive-parallel SIMD matrix to secure and expedite data processing in embedded SOC mobile devices. Improving the efficiency of cipher processing to fulfill crucial objectives like speed, low power usage, and cost is achieved by the interleaved-bitslice process technique. The AES method is being developed and have a clock cycle per byte reduction of up to 93% compared to conventional mobile processors and an energy efficiency that is 4.8% greater than that of a BeagleBoard-xM. The study shows this approach to be effective with the digital-convergence mobile device and the number of clock cycles are minimized compared to other works [34].

Lee, Jang and Kim (2016) investigated the progress of mobile devices with Multi-core SoCs, which allows complex computer vision tasks using GPGPU software such as OpenGL ES and OpenCL. Their goal was to make the Viola-Jones face identification method more efficient computationally; it was only mobile-friendly due to memory access

issues and workload imbalances. A combination of techniques including efficient local memory, dynamic thread allocation, scale picture parallelism, sliding windows parallelism, and CPU-GPU task parallelism allowed the authors to solve these issues. The experiments showed that their implementation could outperform an optimized OpenCV CPU implementation by 3.3 to 6.29 times, implying that it can be applied to other mobile GPU and CPU applications [35].

The Table 1 is a summary of major findings, challenges, limitations, and future research gaps in studies on mobile SoC performance and power efficiency, that emphasize gaps in optimization strategies and opportunities in the system-level design.

Table 1. Comparative Analysis and Research Gaps in Mobile Soc Performance and Power Efficiency Studies

Reference	Study Focus	Key Findings	Challenges Identified	Limitations	Future Work
BenSaleh, Qasim & Obeid (2019)	Design of advanced SoC architecture integrating ARM host processor, high-speed IPs & peripherals	Provided a modular SoC design with rich connectivity (I2C, SPI, UART, watchdog, TPM, etc.) suitable for mobile devices	Complexity of integrating multiple heterogeneous IP cores while maintaining power efficiency	Lacks performance/power optimization evaluation; no experimental results on energy consumption	Develop power-aware integration frameworks, evaluate on real-world mobile workloads, explore security–performance trade-offs
Zhu, Mattina & Whatmough (2018)	System-level co-design for ML-intensive mobile pipelines (camera → ISP → memory → NN accelerator)	Demonstrated that co-optimizing entire mobile vision pipeline yields better efficiency than focusing only on ML accelerators	Difficulty in coordinating multiple heterogeneous SoC components; data transfer overhead	Study limited to continuous computer vision pipelines only	Extend co-design to multi-stage AR/VR/ADAS applications, integrate scheduling and thermal management, generalize framework to next-gen ML accelerators
Kuo et al. (2018)	Dynamic compact thermal modeling (DCTM) for heterogeneous CPU–GPU SoCs	Proposed a simplified yet accurate thermal model capturing CPU–GPU coupling; achieved <1.58°C error in validation	Thermal coupling complicates power management; real-time thermal-aware decision making	Does not integrate with run-time DVFS or task schedulers; modeled specific smartphone environments	Develop real-time thermal-aware schedulers, integrate DCTM with adaptive DVFS, expand model to NPU/AI accelerators
Kumaki, Koide & Fujino (2017)	Secure data processing using a SIMD matrix processor (MX-1) with interleaved-bitslice cipher implementation	Achieved up to 93% fewer clock cycles for AES; energy efficiency 4.8× higher than ARM Cortex-A8	Balancing high-speed cryptography with limited mobile energy budgets	Focused on block cipher workloads only; limited discussion of integration into full SoC	Expand to post-quantum cryptography, integrate MX-1-like accelerators into heterogeneous SoCs, evaluate thermal impact
Lee, Jang & Kim (2016)	CPU–GPU parallelization to accelerate Viola–Jones face detection on mobile devices	Achieved 3.3–6.29× speedups using optimized parallelism (task parallelism, sliding windows, thread allocation)	Irregular memory access and unbalanced workloads limit GPU utilization	Study focuses on a single computer-vision algorithm; outdated relative to modern AI workloads	Extend to deep learning-based CV algorithms, apply dynamic scheduling for heterogeneity, analyze energy vs. performance trade-offs

6. Conclusion and Future Work

Mobile SoC technology is still progressing towards providing more performance in ever more demanding power and thermal constraints. This paper analysis has pointed out the importance of heterogeneous architecture, workload-adaptive scheduling, improved interconnect design, and specialized AI accelerators in improving performance-per-watt in the current devices. The DVFS, clock gating, power gating, as well as power-optimization, are now fundamental in nature, and the management approaches based on AI as predictive workload, thermals, and real-time responsiveness are being introduced as the new epoch. Regardless of these developments, things are not smooth. Several factors are increasing the complexity of integration, augmenting the data-intensive workload, and reducing the size of semiconductor nodes, which all present constraints in the memory bandwidth, thermal removal, and sustained performance. Future work should thus be more co-optimized at the system level and result in better compute, memory, and interconnect dimensions, rather than individual component ones. The anti-bloodstream ones are reinforcement-based learning power controllers, heterogeneous resource-scheduling with cross-layer hardware-software, and energy-sensitive NoC architectures to AI-centric pipelines. Moreover, integration of chiplet-based SoC architectures, near sensor, near memory compute, and advanced cooling can be used to address the existing thermal and scalability constraints. With the emergence of mobile applications into AR/VR, large-scale AI inference, and edge-intelligent systems, novel SoC design methods will be sought after to support the increasing requirements of high performance, low latency, and high energy efficiency.

References

[1] N.-S. Woo, "High performance SOC for mobile applications," in *2010 IEEE Asian Solid-State Circuits Conference*, IEEE, Nov. 2010, pp. 1–4. doi: 10.1109/ASSCC.2010.5716548.

[2] S. S. Reddy, "Comparative Analysis of CPU Scheduling Algorithms for Performance Efficiency," 2019.

[3] G. Nallapati *et al.*, "Cost and power/performance optimized 20nm SoC technology for advanced mobile devices," in *2014 Symposium on VLSI Technology (VLSI-Technology): Digest of Technical Papers*, IEEE, Jun. 2014, pp. 1–2. doi: 10.1109/VLSIT.2014.6894414.

[4] N. D. Lane *et al.*, "DeepX: A Software Accelerator for Low-Power Deep Learning Inference on Mobile Devices," in *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, IEEE, Apr. 2016, pp. 1–12. doi: 10.1109/IPSN.2016.7460664.

[5] N. Rajovic, P. M. Carpenter, I. Gelado, N. Puzovic, A. Ramirez, and M. Valero, "Supercomputing with commodity CPUs," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, 2013, pp. 1–12. doi: 10.1145/2503210.2503281.

[6] G. Yeap, "Smart mobile SoCs driving the semiconductor industry: Technology trend, challenges and opportunities," in *2013 IEEE International Electron Devices Meeting*, IEEE, Dec. 2013, pp. 1.3.1-1.3.8. doi: 10.1109/IEDM.2013.6724540.

[7] A. Selinger, K. Rupp, and S. Selberherr, "Evaluation of mobile ARM-based SoCs for high performance computing," *Simul. Ser.*, vol. 48, no. 4, pp. 154–160, 2016, doi: 10.22360/springsim.2016.hpc.022.

[8] F. Poeh, F. Demmerle, J. Alt, and H. Obermeir, "Production test challenges for highly integrated mobile phone SOCs & x2014; A case study," in *2010 15th IEEE European Test Symposium*, IEEE, May 2010, pp. 17–22. doi: 10.1109/ETSYM.2010.5512786.

[9] S. Garg, "Predictive Analytics and Auto Remediation using Artificial Inteligence and Machine learning in Cloud Computing Operations," *Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci.*, vol. 7, no. 2, 2019, doi: 10.5281/zenodo.15362327.

[10] S. Malallah, Y. Zalah, and R. Karne, "An Analysis of the Advanced Encryption Standard and Threats Associated," 2018, doi: 10.13140/RG.2.2.34873.88168.

[11] M. Hill and V. J. Reddi, "Gables: A Roofline Model for Mobile SoCs," in *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, IEEE, Feb. 2019, pp. 317–330. doi: 10.1109/HPCA.2019.00047.

[12] M.-F. Chang, P.-F. Chiu, and S.-S. Sheu, "Circuit design challenges in embedded memory and resistive RAM (RRAM) for mobile SoC and 3D-IC," in *16th Asia and South Pacific Design Automation Conference (ASP-DAC 2011)*, IEEE, Jan. 2011, pp. 197–203. doi: 10.1109/ASPDAC.2011.5722184.

[13] H. Saito *et al.*, "A Chip-Stacked Memory for On-Chip SRAM-Rich SoCs and Processors," *IEEE J. Solid-State Circuits*, vol. 45, no. 1, pp. 15–22, Jan. 2010, doi: 10.1109/JSSC.2009.2034078.

[14] B. Journal, "Cache Coherence Protocol Design and Simulation Using IES (Invalid Exclusive read/write Shared) State," *Baghdad Sci. J.*, vol. 14, pp. 219–230, 2017, doi: 10.21123/bsj.14.1.219-230.

[15] S. Yang *et al.*, "10nm high performance mobile SoC design and technology co-developed for performance, power, and area scaling," in *2017 Symposium on VLSI Technology*, IEEE, Jun. 2017, pp. T70–T71. doi: 10.23919/VLSIT.2017.7998203.
[16] M. Said, S. Chetoui, A. Belouchrani, and S. Reda, "Understanding the Sources of Power Consumption in Mobile SoCs," in *2018 Ninth International Green and Sustainable Computing Conference (IGSC)*, IEEE, Oct. 2018, pp. 1–7. doi: 10.1109/IGCC.2018.8752140.
[17] E. Sueur and G. Heiser, "Dynamic voltage and frequency scaling: The laws of diminishing returns," *2010 HotPower*, 2010.
[18] S. Achouche, U. B. Yalamanchi, and N. Raveendran, "Method, apparatus, and computer-readable medium for performing a data exchange on a data exchange framework," 2019
[19] P. Grosse, Y. Durand, and P. Feautrier, "Methods for power optimization in SOC-based data flow systems," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 14, no. 3, pp. 1–20, 2009.
[20] W. Huan, J. Han, S. Wang, and X. Zeng, "A low-complexity heterogeneous multi-core platform for security soc," in *2010 IEEE Asian Solid-State Circuits Conference*, IEEE, Nov. 2010, pp. 1–4. doi: 10.1109/ASSCC.2010.5716621.
[21] X. Guerin and F. Petrot, "A System Framework for the Design of Embedded Software Targeting Heterogeneous Multi-core SoCs," in *2009 20th IEEE International Conference on Application-specific Systems, Architectures and Processors*, IEEE, Jul. 2009, pp. 153–160. doi: 10.1109/ASAP.2009.9.
[22] K. Yu, D. Han, C. Youn, S. Hwang, and J. Lee, "Power-aware task scheduling for big.LITTLE mobile processor," in *2013 International SoC Design Conference (ISOCC)*, IEEE, Nov. 2013, pp. 208–212. doi: 10.1109/ISOCC.2013.6864009.
[23] U. A. Korat and A. Alimohammad, "A Reconfigurable Hardware Architecture for Principal Component Analysis," *Circuits, Syst. Signal Process.*, vol. 38, no. 5, pp. 2097–2113, 2019, doi: 10.1007/s00034-018-0953-y.
[24] A. Mazare, L.-M. Ionescu, A.-I. Lita, G. Serban, and M. Ionut, "Mobile system with real time route learning using Hardware Artificial Neural Network," in *2015 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, IEEE, Jun. 2015, p. P-45-P-48. doi: 10.1109/ECAI.2015.7301250.
[25] K. Ryu, D. Jung, and S.-O. Jung, "A DLL based clock generator for low-power mobile SoCs," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1950–1956, Aug. 2010, doi: 10.1109/TCE.2010.5606351.
[26] D. Flynn, "Power gating applied to MP-SoCs for standby-mode power management," in *Proceedings of the 50th Annual Design Automation Conference*, May 2013, pp. 1–5. doi: 10.1145/2463209.2488930.
[27] J. Okwuibe, J. Haavisto, E. Harjula, I. Ahmad, and M. Ylianttila, "Orchestrating service migration for low power mec-enabled iot devices," *arXiv Prepr. arXiv1905.12959*, 2019.
[28] S. Gupta and S. Prakash, "QoS and load balancing in cloud computing-an access for performance enhancement using agent based software," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 11, pp. 641–644, 2019.
[29] T. Otani, H. Toube, T. Kimura, and M. Furutani, "Application of AI to mobile network operation," *ITU J. ICT Discov. Spec. Issue*, vol. 1, pp. 1–7, 2017.
[30] A. Ignatov *et al.*, "Ai benchmark: Running deep neural networks on android smartphones," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, p. 0.
[31] M. S. BenSaleh, S. M. Qasim, and A. M. Obeid, "Toward Design of Advanced System-on-Chip Architecture for Mobile Computing Devices," in *International Conference on Mobile, Secure, and Programmable Networking*, 2019, pp. 88–95.
[32] Y. Zhu, M. Mattina, and P. Whatmough, "Mobile Machine Learning Hardware at ARM: A Systems-on-Chip (SoC) Perspective," 2018.
[33] S.-L. Kuo, C.-W. Pan, P.-Y. Huang, C.-T. Fang, S.-Y. Hsiau, and T.-Y. Chen, "An Innovative Heterogeneous SoC Thermal Model for Smartphone System," in *2018 17th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*, 2018, pp. 384–391. doi: 10.1109/ITHERM.2018.8419605.
[34] T. Kumaki, T. Koide, and T. Fujino, "Secure data processing with massive-parallel SIMD matrix for embedded SoC in digital-convergence mobile devices," *IEEJ Trans. Electr. Electron. Eng.*, vol. 12, no. 1, pp. 96–104, Jan. 2017, doi: 10.1002/tee.22349.
[35] Y. Lee, C. Jang, and H. Kim, "Accelerating a computer vision algorithm on a mobile SoC using CPU-GPU co-processing," in *Proceedings of the International Conference on Mobile Software Engineering and Systems*, May 2016, pp. 70–76. doi: 10.1145/2897073.2897081.