Volume 12, No.8, August 2025

Journal of Global Research in Mathematical Archives

RESEARCH PAPER

Available online at http://www.jgrma.com

ISSN 2320 – 5822

DESIGN STRATEGIES FOR MOBILE SOCS: BALANCING BATTERY LIFE AND COMPUTATIONAL DEMAND

Sandeep Gupta[1]

[1] SATI, Vidisha, Sandeepguptabashu@gmail.com

Abstract: Modern smartphones are based on highly integrated Mobile System-on-Chip (SoC) technologies enabling functionalities in communication, high-performance gaming, and many others. As there is an increasing demand of high performance and energy efficiency in mobile electronics, design of System-on-Chips (SoCs) becomes more important. The battery life is still one of the key factors affecting the usability and life cycle of the mobile devices. System-on-Chip (SoC) technologies have facilitated the action of high-performance computing abilities in small and energy-constrained platforms. With the increase in real-time applications like augmented reality (AR), artificial intelligence (AI), and high-resolution multimedia, there comes a challenge of balancing between energy utilization and computational power, which is a major challenge to design. In this review, there is a detailed discussion of the modern mobile System-on-Chip (SoC) creations, energy-saving hardware solutions, advanced battery management system (BMS) approaches, and software-level optimization solutions. The role of heterogeneous computing and machine learning (ML) in dynamic workload management and energy optimization is of special concern. The possibility of artificial intelligence-based battery management system (AI-based BMS) to increase efficiency of operation and health of the battery is also discussed. In addition, Mobile Edge Computing (MEC) structures permitting venture in unmanned aerial vehicle (UAV)-based frameworks are examined and broadly contrasted qualitatively regarding algorithmic strategies. The paper ends with section devoted to the discussion of the current issues and the future research avenues in the design of energy-efficient mobile System-on-Chip (SoC) systems, that offer high performance.

Keywords: Mobile SoC Design, Battery Management System (BMS), Energy-Efficient Computing, Computational Workload Optimization, Power Management Strategies, Heterogeneous Multi-core Architecture, Dynamic Voltage and Frequency Scaling (DVFS)

1. INTRODUCTION

Mobile SoCs have changed the entire consumer electronics market over the past 10 years, with devices that integrate a myriad of functionality such as audio, video, CPU cores, GPU, DSP, modem, memory and I/O controller all in a single-chip package [1]. These intensively integrated platforms are the ones meant to provide high level user experiences in a broad range of mobile devices. Nevertheless, the growing computational power of those chips has ruffled major problems of energy consumption such that battery life has become major issue of concern to the end user.

This problem has necessitated the innovation of low-power mobile SoC design especially in powering new and resource demanding applications like Augmented Reality (AR) and Virtual Reality (VR) and mobile games. Efficient power management also involves strategies at the system level, which take into consideration performance and energy efficient considerations simultaneously and not only at the component-level optimization.

The big constraint in the implementation of mobile SoC is due to the shortfalls of battery performance. The Pack of batteries works as the weakest battery, because when one cell of a pack is discharged, all the pack becomes useless [2]. It is therefore crucial to monitor the State of Charge (SOC) which is the quotient of remaining charge in a cell with cell capacity. The estimation of SOC includes various battery parameters which are, voltage, the integrated charge/discharge currents, and temperature.

A very important role of Battery Management Systems (BMS) in the current applications related to mobile and electric vehicle (EV) is the efficient and safe use of batteries [3]. It monitor, control charging and discharging, temperature control, fault diagnosis, data acquisition as well as cell balancing [4]. The latter maintains the same voltage in all the cells thereby maximizing the battery use as well as the battery life [5]. Due to high-performance BMS designs integrate high precision monitoring with active or passive cell balancing the cell-to-cell SOC differences are reduced and the effects of cell ageing can be kept minimal as well. Performance requirements and indicators of such systems: the optimization of the speed, effectiveness, size, and cost.

The need to boost the performance of the growing complex battery systems and availability of huge datasets introduced the concept of using Artificial Intelligence (AI) and Machine Learning (ML) in the modern BMS designs [6]. Such intelligent systems provide superior forecasting functionality in estimating State of Health (SOH) and Remaining Useful Life (RUL) and listing faults, which in many cases outsmart the conventional systems (based on rules) concerning precision and flexibility.

Regarding the design of mobile SoCs, the ability to co-optimize the hardware and software is required to meet the challenging power and thermal requirements through computational performance requirements. The main innovations that are going to help strike this balance include Dynamic Voltage and Frequency Scaling (DVFS), heterogeneous computing architectures, and workload-aware scheduling. Finally, energy-wise design methodologies coupled with smart battery, and power management techniques are keys to support that mobile SoC can still optimize performance levels without any trade-offs on battery lifetime and reliability.

1.1. Structure of the Paper

The paper is presented in the following way: Section 1 contains introduction to mobile SoCs, design challenges, and background and importance. Section 2 revises basics of SoC based architecture and energy-conservation methods. In Section 3, the life of battery and power management in mobile is discussed. In Section 4, the computational requirements of the current mobile applications are discussed. Section 5 offers related literature and Section 6 concludes with main insights and future directions.

2. FUNDAMENTALS OF MOBILE SOC DESIGN

A System-on-a-Programmable-Chip (SoPC) is constructed out of Programmable Logic Devices (PLDs); it has all the benefits of simple development, quick prototyping, and the ability to change easily [7]. SoPCs are most often based on Field Programmable Gate Arrays (FPGAs) and used VHDL to define the hardware and are targeted at prototyping and low and medium volume manufacturing. FPGAs as a major development in VLSI design are more advantageous in terms of efficiency, shorter time of development and convenience in future changes. Alternatively, the other way of developing a mobile SoC is the Application-Specific Integrated Circuits (ASICs) that offer superior performance and power consumption when keeping the fixed-function applications, although less flexibility. Figure 1 shows one generic design flow of mobile SoCs.

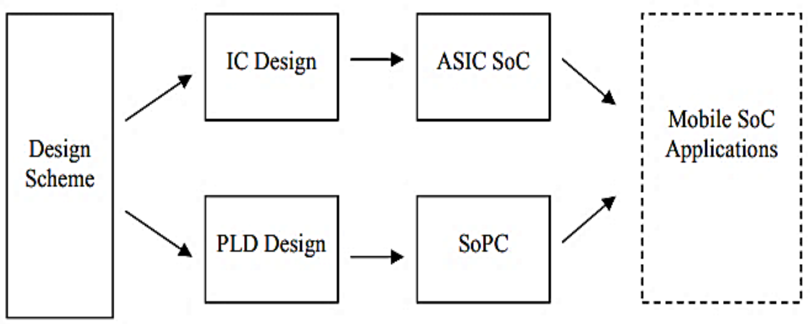

Figure 1: The design procedure of a mobile SoC

Two directions in the design of mobile SoC application are a single route that is based on IC design that leads into high-end ASIC SoC and the other route is based on PLD design that leads into SoPCs in easy updatable solutions that meet the demands of the mobile application solutions.

2.1. Overview of System-on-Chip (SoC) Architecture

System-on-Chips (SoCs) architecture has been one of the key topics of research both in academia and industry since defining the architectural parameters of SoCs by determining their performance, efficiency, and scalability mainly depends on it, the core principles of SoC design, and the essence of integrating processor cores, memory subsystems, and peripherals onto the same chip. These works pointed to the significance of interconnects and communication algorithms that should be efficient in data transfer in the SoC. These ideas were developed further by subsequent researchers, whereby works have been done on the exploration of complex architectures that maximize the power consumption and speed of the process. Current researches revolve around various architectures, in which varied forms of processing units are united in order to process certain functions more effectively. This shift towards heterogeneous computing is carried by the increasing demand for specialized processing in applications such as AI, machine learning, and high-performance computing

2.2. Strategies for Energy-Efficiency in SoCs

Modern microarchitecture has made energy efficiency in System-on-Chip (SoC) design a primary goal in response to the rising performance demands of mobile and embedded systems [8]. In order to lower power consumption while keeping performance levels sufficient, many methods have been devised. Considering the paramount relevance of mobile platform energy efficiency, this section offers a synopsis of important design concepts that strive to optimize power utilization. Despite their generalizability, these tactics shine when used to the development of on-chip communication networks that minimize power consumption. The following foundational principles serve as a basis for developing power-conscious SoC architectures and SoC strategies:

- **Involve all layers:** The physical layer, the application itself, the system architecture, the operating system, and the communication protocol stack are all affected by the problem of energy efficiency. Designing an energy-efficient system requires the system to be optimized for energy in an integral way. A system built out of components that have individually been optimized for low power does not necessarily result in the most energy-efficient system.

- **Applying locality of reference:** The term "locality of reference" describes the practice of keeping frequently accessed data in close proximity to the components that do the processing. It is significantly more energy efficient to access a tiny, local memory rather than a large, distant one.
- **Avoid useless activity:** This serves as the primary impetus for dynamic power management, link layer protocols, and adaptive error correction. Superfluous activity may arise from multiple sources throughout the system, including redundancy in a high-power operational mode, implementing error control on inherently error-resilient data, or endeavoring to transmit a video frame that is already obsolete.
- **Dynamic Voltage and Frequency Scaling (DVFS):** Dynamic Voltage and Frequency Scaling (DVFS) is a popular method that optimizes processor power and frequency in response to changing workloads. Because of the quadratic relationship between voltage and dynamic power consumption, it is possible to obtain large power savings by reducing the voltage and frequency during periods of low computational demand. Various DVFS policies leverage workload prediction, thermal thresholds, and application-specific requirements to optimize energy use while preventing performance degradation.
- **Power Gating and Clock Gating:** Power gating minimizes the amount of power lost to the idle functional blocks by shutting them off completely, thus the leakage currents are suppressed. Conversely, clock gating de-asserts the clock signal to certain circuits that are not used actively, avoiding unneeded switching and such thus lowering dynamic power requirements. Hardware-based state monitors or software-defined policies that determine idle resources and turn them off in a selective manner tend to control these gating mechanisms.
- **Low-Power States and Idle Modes:** Recent SoCs add numerous power states, which are specified by the ACPI, or related standards. A big reduction in the amount of power used can be made by transitioning to low-power or sleep modes when the system is idle e.g. standby, deep sleep or hibernation. Hardware mechanisms and firmware policies that contribute to these modes deal with entry and exit latencies so as to offer seamless performance recovery with little energy expenditure.
- **Approximate and Opportunistic Computing:** Approximate computing leverages the error tolerance of certain applications, such as multimedia processing or ML inference, to relax computational accuracy in exchange for energy savings. By employing reduced-precision arithmetic, task skipping, or voltage over scaling, SoCs can lower energy consumption while maintaining acceptable output quality. Opportunistic computing further exploits variability in workload characteristics and environmental conditions to dynamically adjust computation fidelity or resource utilization.

2.3. SoC Design Technology

System on a chip (SoC) functionality can be defined in either software or hardware. Each has its own set of strengths and areas for improvement. Software design's benefits include reduced development time, increased extensibility, and higher fidelity [9]. On the other hand, software design result in slower operating speed and higher memory resource requirements. The hardware architecture, in contrast, has a more dependable process and quicker operational speed, but it is less flexible and takes more time to build [10]. Additionally, categories software into two types: application software and embedded software, taking into account the latest trend in embedded processor development. Embedded software, or firmware, is created at a lower level of abstraction in an embedded processor, whereas application software is written at a higher level in an X86 CPU (host PC). As an alternative, there are two types of hardware: fixed hardware and configurable hardware. Discrete parts, daughter boards, and an expansion board based on printed circuit boards make up the fixed hardware. Limited functionality, extensive development time, and specialized design skills are all requirements of fixed hardware design. On the other hand, reconfigurable hardware is a system that uses a specific model of hardware (HDL code) to function. For design challenges including reconfigurability, development time, and supporting concurrent software development, FPGA is a well-known platform.

3. BATTERY LIFE CONSTRAINTS IN MOBILE DEVICES

Battery life remains one of the most critical design constraints in mobile devices, directly influencing user satisfaction and the practicality of high-performance features. As mobile applications become more computationally demanding, the need to sustain longer battery life without compromising performance has become a central challenge [11]. The limited capacity of lithium-ion batteries, constrained by physical size and safety considerations, poses a bottleneck in meeting the energy needs of modern mobile systems [12]. Power consumption in mobile devices arises from various sources, including the CPU, GPU, display, memory, sensors, and background processes. Even idle or background tasks such as notifications, app updates, and location services contribute to energy drain. Furthermore, thermal constraints prevent sustained high-performance operation, as overheating can degrade both battery life and user experience. Therefore, intelligent power management strategies such as adaptive screen brightness, workload scheduling, and hardware-level optimizations like Dynamic Voltage and Frequency Scaling (DVFS) are essential. Ultimately, balancing the growing computational demands of mobile applications with the need for prolonged battery life remains a key focus in mobile SoC design.

3.1. Battery Management System

A battery management system is an electrical framework that controls, verifies, and/or adjusts the battery's condition, as well as monitors its state by processing data received from sensors [13]. Choosing the right materials for making batteries is just as crucial as BMS when it comes to ensuring the safety of products that run on batteries. Protecting, reliable, and efficiently using components is a top priority in industries like the automotive sector. In order to extend the battery's life, the BMS regulates its working conditions, including its current, voltage, and power. It also provides accurate assessments of the battery's State of Charge (SOC) and State of Health (SOH) for intelligent applications. An electric vehicle (EV) or hybrid electric vehicle (HEV) without a monitoring system is a disaster waiting to happen due to the battery's unpredictable behavior. From a safety standpoint, the BMS should notify the user and take appropriate action in the event of any unusual conditions. In addition to this, in an automobile system, the BMS should check the temperature of the coolant surrounding the battery pack. As a result, power is more evenly distributed to each part.

3.2. Battery Modeling

Portable electronic devices like smartphones and laptops have long relied on lithium-ion batteries. Their superior energy and power density make them the go-to choice for electrifying vehicles, especially BEVs, as compared to other commercially available battery types. There is no memory effect, a low self-discharge rate, and a long cycle life for lithium-ion batteries. State of Health (SoH) and State of Charge (SoC) measurements cannot be taken directly from lithium-ion batteries because of their intricate internal structures and the natural ageing process [14]. By far, the most precise and dependable methods for determining the SoC and the SoH are those that rely on model-based approaches. That is why most battery behaviour characterizations are based on electrochemical/physics-based models or comparable circuit models. Furthermore, it is feasible to reduce expensive measurements by estimating how batteries behave under various load scenarios. Simple analogous circuit models based on RC-pairs are studied within the framework of this paper's examination.

3.3. User-Centric Power Management

Computer systems may be continuously optimized for performance and energy efficiency with the help of precise and comprehensive user activity monitoring [15]. Having a good grasp of how users engage might help it anticipate what resources you'll require. Application resources are then more efficiently allocated to those that make better use of them, among other performance optimizations [16]. For designs to be energy efficient, optimization must be done methodically at every level of abstraction, from algorithms and process technology to logic design and structures. During the design process, a number of static techniques are used, such as algorithms for reducing strength, algebraic and algorithmic transformations, retiming, and logic minimization. One alternate strategy is to use methods like application-driven management to dynamically change system operation and energy usage based on application workload. This might be done during system operation, as seen in Figure 2.

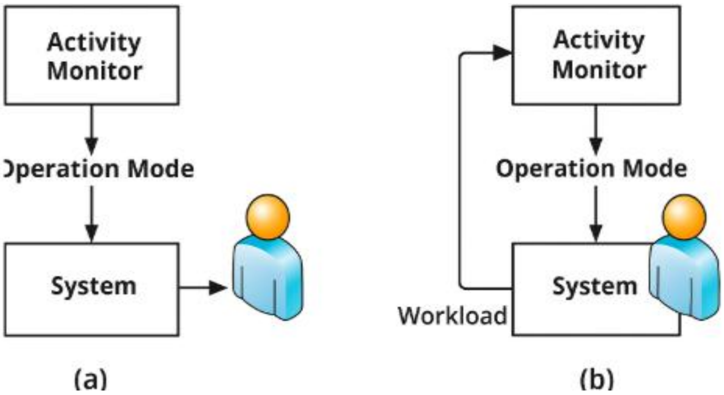

Figure 2: Systems for Energy Management (a) Current (b) Future

Despite their differences, these dynamic approaches all use the same principle: while no activity inputs are present, the system remains in lowest power mode, and it activates itself if there is a change in the input signals. In order to put the plan into action, the system integrates an additional unit that continuously tracks the input activity or workload and, using that data, decides the system's new operating mode.

4. COMPUTATIONAL DEMANDS OF MODERN MOBILE APPLICATIONS

Applications on mobile devices have progressed beyond mere communication and productivity applications into those that enable computationally intensive, real-time experiences, such as gaming, AR, VR, AI image recognition, and high-definition, high-bit-rate multimedia transmission [17]. They also continue to demand large processing resources, high-speed memory clock, as well as low latency. In real-time applications such as video conferencing and mobile games, the frames should have a stable rate with minimal or no latency, and the endpoints on the CPU, GPU, and neural processing unit (NPU) are limited [18]. Such on-device machine learning applications as facial recognition, voice assistants (e.g., Google Assistant), and predictive typing can demand effective power management and AI accelerators. The most resource-intensive and demanding of the system are high-performance multimedia applications (playing 4K/8K video and image editing). Non-activity activities that require computing resources, such as data synchronization, location updates, and system update procedures, can consume resources even when the app is idle. Overall, this has resulted in varied and inexplicably rising resource requirements in the virtual product world, and it is vital that mobile SoCs be knowingly designed to take resources not only on account of an exertion extremity, but in like manner on account of an optimality of energy and thermal enlargement.

4.1. Workload Characterization

Workloads, both traditionally and in the context of cloud computing, are defined and discussed in this section. Based on the material that is already available, compare traditional web servers with cloud servers in order to differentiate between the various varieties. Figure 3 shows the results of the data analysis that characterizes the workloads from the key application domains. Workload characteristics are not described in detail in most research [19]. Each component of the workload is specified by parameters that capture the demands imposed on the system's resources by different activities. Because different workloads behave in different ways, picking the right platform is task-specific. Because of its many real-world uses, characterization has attracted a lot of research interest and usually involves conducting experiments on systems while they are in operation.

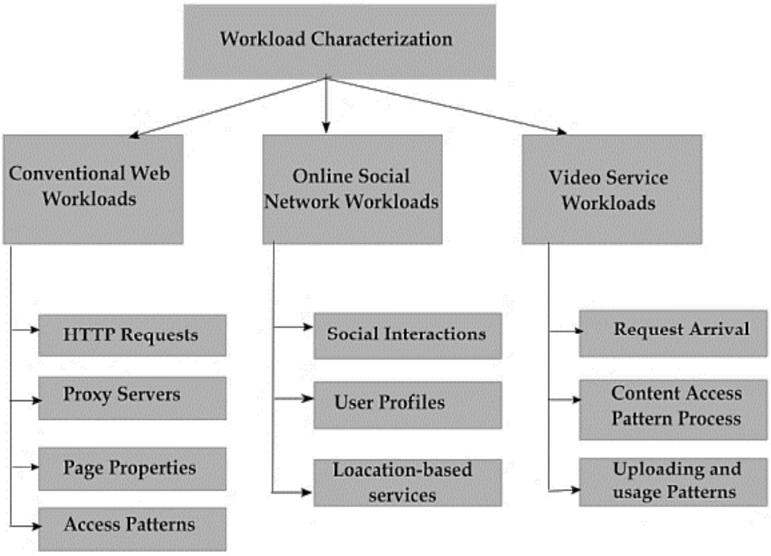

Figure 3: Workload characterization in Computing

Workloads on traditional websites, social media platforms, and video services are the three primary types of online workloads. Common features of traditional online workloads include HTTP requests, proxy servers, page attributes, and access behaviors. Online social network workloads focus on social interactions, user profiles, and location-based services. Video service workloads are defined by request arrival, content access pattern processes, and uploading and usage patterns. Each category highlights specific parameters used to analyze and optimize system performance based on workload behavior.

4.2. Machine Learning Techniques for Mobile Devices

Machine learning is a great tool for mobile devices; for example, XPod can automatically choose songs based on the user's preferences. By applying machine learning techniques, this system was able to play music based on the user's context. Since SVMs initially require very little training data, they are uniquely suited to the many tasks performed by mobile music players [20]. From the user's perspective, this is a great feature as mobile users spend so less time configuring their devices and more time basking in their benefits. Contrarily, it is a time-consuming chore to construct SVMs for a limited device. Simple VGGs (support vector machines) are simple to design on a PC or laptop and then move to a mobile device. Even a limited device can test an SVM in this manner. The difference is that decision trees are classifiers that, when trained correctly, can be transformed into a set of rules for additional evaluation. Boosting the performance of decision trees is possible. Free from oversight one area where machine learning techniques really shine is data mining. Furthermore, the fields of network administration and surveillance can benefit from machine learning approaches. Phones and tablets are the sweet spot for machine learning algorithms like RL and IBL.

4.3. Multimedia Data Processing Technology and Related Applications

Multimedia Data Processing-Related Processes. The client-server system is the manner of operation for multimedia processing technologies. In order for multimedia applications and multimedia information processing technologies to work internally, the server must transition to server service mode when requested to do so. Once in this mode, the server processes the request, returns an error message, and finally processes the report. At last, the request frees up system resources, destroys the converted object, and ends the conversion process [21]. In order to run the communication process, a control request can be submitted while performing the actions mentioned above. It can configure starting, pause, recovery, and other characteristics for media data with time sequence, current play position, and play cycle time, among other things. Depending on how still image data is consumed, the non-time series output might be necessary for editing and analysis. The idea and purpose of multimedia information processing technology lies in the fact that it facilitates the management of various information resources, including multimedia application software, operating system management hardware and software, and embedded data processing equipment.

4.4. Key Design Challenges

The design of modern System-on-Chip (SoC) architectures, especially for mobile and embedded platforms, involves navigating a complex set of trade-offs. As performance demands increase alongside constraints on energy, thermal output, and cost, several fundamental challenges must be addressed during the design process.

- Thermal and Power Constraints Power consumption remains a primary limitation in SoC design, directly impacting battery life, thermal dissipation, and device reliability [22]. High power densities lead to increased heat generation, which can degrade performance, reduce lifespan, and necessitate costly thermal management solutions. Designers must balance dynamic and leakage power while meeting thermal design limits, particularly in passively cooled mobile devices.
- Performance Scalability: Achieving scalable performance across diverse applications is another critical challenge. As transistor scaling slows and power budgets tighten, simply increasing core counts or clock frequencies is no longer viable.

Scalable performance necessitates architecture innovations, including parallelism, workload-aware schedule, and intelligent resource utilization with variable performance and energy budgets.

- Area and Cost Constraints Area-optimized SoC designs serve to reduce restrictions in manufacturing cost as well as allowing more functionality within one die. The production yield as well as pricing immediately depends on the area efficiency particularly in (consumer electronics). Area limits frequently involve trade-off between complexity, performance and power, as well as marketing-driven cost goals.
- Heterogeneity and Workload Diversity Heterogeneous computing (the use of heterogeneous computing elements including CPUs, GPUs, NPUs, and DSPs) is included in modern SoC designs, to support diverse application domains. This non-homogeneity brings in an aspect of complexity of the movement, scheduling and synchronization of data within subsystems. Moreover, there is a diverse range of workloads including multimedia processing applications and machine learning inference applications which require scalable architectures so as to achieve a high degree of efficiency with different performance and energy demands.

5. LITERATURE REVIEW

The literature Summary is an in-depth review of the innovations in the battery optimization, SoC design, and energy storage systems and what role they play in the electric mobility, renewable integration, and computing efficiency, as well as what are major challenges of this direction and what research can be conducted in the future in an interdisciplinary manner.

Leijon (2025) presents a synopsis of the current research on the ageing of electric vehicle batteries through various charging techniques, including conductive charging, inductive charging, and battery swapping. Battery degradation is accelerated under certain charging settings, according to this study. These variables include rapid charging at low temperatures. Temperature, charging current, and battery state-of-charge are only a few of the variables that affect the unpredictability of battery ageing. As a means to increase grid flexibility and profitability, vehicle-to-grid connectivity is attracting attention. Need to conduct additional research on the linked battery degradation, though. Ensuring socially acceptable and economically viable solutions while planning for a charging infrastructure that is suited for electric car users is a fundamental difficulty in the decision-making process [23].

Colucci et al. (2024) proposed a new taxonomy for optimizing battery performance, surveyed typical BESS usage strategies, and categorized these plans according to the taxonomy. As part of their categorization, they review the existing approaches for optimizing batteries, assess how well they handle problems that develop during BESS implementation, and propose other lines of inquiry. Power and electricity generated by renewable sources, such solar and wind, is being used more and more frequently. The trend is driven by a desire to save money while also protecting the environment. Nevertheless, the real test in how can harness and store this energy. Improving the timing of battery charging and discharging is one strategy [24].

Cirstea et al. (2024) examines the development of approaches and resources for the modelling, simulation, and design of digital electronic system-on-chip (SoC) implementations, particularly in the context of industrial electronics. Prior to delving into the approaches and tools used in system-on-chip (SoC) design, the article provides an overview of important technological, economic, and geopolitical trends. It lays forth the basics of the design flows for systems on a chip. Subsequently, the article reveals how the IP industry has played a pivotal role in the seemingly endless enhancements to the PPAC characteristics of SoCs. Design capture at very abstract levels and the proliferation of automated design tools are pushing the envelope farther. Subjects such as aerospace and automotive are covered in small case studies [25].

Jose and Shrivastava (2024) provides an in-depth analysis of the current state and future prospects of electric vehicles, a subject not covered in earlier evaluations. Using theme analysis, explore the pros and cons of top-performing methods for state-of-charge prediction and create a thorough classification of accurate hybrid approaches. At last, detail the evolution of these algorithms and the best way to obtain the most accurate forecast for the State of charge. The necessity of precise state-of-charge estimation in ensuring optimal battery performance, longevity, and safety has been highlighted by the meteoric rise in the popularity of electric vehicles [26].

Chen et al. (2023) Presents EI-LSTM-CO, a new LSTM-RNN with limited output (CO) and extended input (EI) for estimating battery state of charge (SOC). In order to improve the network's capacity to map the nonlinear battery properties and decrease the output SOC fluctuations, add an extra slow time-varying information sliding window average voltage to the network inputs. To improve the network's SOC estimation performance and smooth its output, a state flow strategy based on Ampere-hour integration (AhI) is employed. To validate the strategy's performance and generalizability, $LiFePO_4$ battery datasets at different temperatures are utilized. Another goal is to limit the variation between neighboring SOCs' outputs [27].

Shan et al. (2022) analyze the memory and processing architectures of computer systems that rely on chiplets. A presentation and summary of the computing architecture utilized for high-performance computing, mobile, and PC is provided first. Next, present the memory architecture that is based on both conventional memory and new non-volatile memory, which are both employed for data storage and processing. next compare and discuss the major memory characteristics. Automation in transportation, weather forecasting, medical diagnostics, and many more fields rely on computing systems. Since computers are the brains of any electronic device, their efficiency is paramount to its intellectualization. An acceptable computer system design can resolve the trade-offs between efficiency, cost, and performance [28].

Rouholamini et al. (2022) the most recent developments in the field of grid-connected Li-ion BESS management and their role in power marketing. The primary topics covered in this review include battery modelling, BMS design, BESS integration for power market services, worldwide utility-scale BSS facilities, and difficulties in managing and installing grid-connected BESSs. To balance supply and demand and improve reliability, energy storage systems (ESSs) are essential in current power grids because to the intermittent nature of renewable energy sources. Installed energy storage capacity is still very modest on a global scale, with the exception of pumped-storage hydroelectric projects [29].

Table 1 presents a structured comparative analysis of recent studies on mobile SoC design strategies, emphasizing battery optimization, architectural advancements, energy storage integration, and predictive modeling to support sustainable and efficient system development.

Table 1: Summary on Design Strategies for Mobile SoCs: Balancing Battery Life and Computational Demand

Author	Study On	Approach	Key Findings	Challenges	Future Directions
Leijon et al (2025)	Battery degradation under charging strategies and its impact on EV planning	Empirical and conceptual analysis	Fast charging at low temperatures accelerates battery ageing; trade-off exists between charge speed and battery life	Devising charging infrastructure that balances battery health, user convenience, and economic viability	Encourage interdisciplinary research for sustainable charging systems and clean energy transitions
Colucci et al. (2024)	Taxonomy and optimization strategies for battery energy storage systems (BESS)	Literature survey with classification framework	Proposes a new taxonomy for BESS optimization and identifies high-efficiency strategies for renewable energy integration	Efficiently optimizing charge/discharge decisions and economic returns	Develop novel BESS strategies leveraging the proposed taxonomy
Cirstea et al. (2024)	Evolution of SoC modeling, simulation, and design methodologies	Technological review with industrial focus	Highlights advances in IP-based design, high-level abstraction, and automation tools for SoC development	Integrating high-performance and cost-efficient SoC design across domains	Apply advanced SoC tools in critical sectors like automotive and aerospace
Jose et al. (2024)	Predicting the state of charge (SoC) in battery systems used in electric vehicles	Thematic and algorithmic literature review	Reviews hybrid SoC estimation techniques with high accuracy; emphasizes role of BMS in performance and safety	Accurate SoC estimation remains complex under dynamic EV conditions	Enhance SoC algorithms and BMS integration for real-time, reliable EV performance
Chen et al. (2023)	LiFePO4 battery state-of-charge estimate using a long short-term memory (LSTM) deep learning model	Constructed EI-LSTM-CO, an LSTM-RNN with Limited Output and Extended Input	EI improves nonlinear mapping capability; CO smoothens output and reduces SOC estimation fluctuation. Validated on varying temperatures.	Sensitivity to dynamic conditions; need for additional temperature-specific calibration.	Extend model for different battery chemistries and vehicle applications; improve robustness in real-time systems.
Shan et al. (2022)	Chiplet-based system architectures for mobile and high-performance computing	Architecture analysis and comparative review	Summarizes computing and memory architectures including NVM; shows potential for cost-effective high-performance systems	Balancing performance, cost, and efficiency in Chiplet architectures	Explore Chiplet-based solutions for applications like autonomous systems and AI
Rouholamini et al. (2022)	Grid-connected Li-ion BESS and electricity market integration	Comprehensive review of BMS, grid applications, and market roles	BESS essential for managing renewable intermittency and enhancing grid reliability	Low global deployment and regulatory hurdles	Improve battery tech adoption with supportive policies and market mechanisms

6. CONCLUSION AND FUTURE WORK

The rapid advancement of mobile System-on-Chip (SoC) technology has enabled powerful and compact computing by integrating CPUs, GPUs, memory, modems, and accelerators on a single chip. While this integration has significantly improved performance, it has also increased energy demands, making battery life a critical constraint in mobile system design. Addressing this challenge requires a holistic approach that combines energy-aware architectures, intelligent battery management, and machine learning-based power optimization. Techniques such as heterogeneous multi-core systems, dynamic power management, and hardware/software co-design have shown promise in enhancing efficiency. However, sustaining secure and reliable operation under diverse workloads remains a key concern, further complicated by the lack of open-source hardware data and real-time performance metrics.

Future research should explore the integration of AI for predictive and adaptive power management, real-time workload profiling, and intelligent task offloading between edge and cloud environments. Supporting emerging battery technologies with smart, self-healing systems and investigating novel architectures, such as RISC-V, neuromorphic computing, and chiplet-based SoCs, will be essential. Cross-layer optimization will play a crucial role in achieving a sustainable balance between performance and energy efficiency in next-generation mobile computing platforms.

REFERENCES

[1] V. Panchal, “Designing for Longer Battery Life: Power Optimization Strategies in Modern Mobile SOCS,” *Int. J. Electr. Eng. Technol.*, vol. 16, no. 1, pp. 1–17, Jan. 2025, doi: 10.34218/IJEET_16_01_001.

[2] Z. B. Omariba, L. Zhang, and D. Sun, “Review of Battery Cell Balancing Methodologies for Optimizing Battery Pack Performance in Electric Vehicles,” *IEEE Access*, vol. 7, pp. 129335–129352, 2019, doi: 10.1109/ACCESS.2019.2940090.

[3] A. R. Itagi, R. Kallimani, K. Pai, S. Iyer, O. L. A. López, and S. Mutagekar, “Cell Balancing Paradigms: Advanced Types, Algorithms, and Optimization Frameworks,” pp. 1–33, 2024, doi: 10.48550/arXiv.2411.05478.

[4] V. Panchal, “Energy-Efficient Core Design for Mobile Processors : Balancing Power and Performance,” *Int. Res. J. Eng. Technol.*, vol. 11, no. 12, pp. 191–201, 2024.

[5] A. Ashraf *et al.*, “Review of Cell-Balancing Schemes for Electric Vehicle Battery Management Systems,” *Energies*, vol. 17, no. 6, Mar. 2024, doi: 10.3390/en17061271.

[6] J. M. Kim, “Explainable Artificial Intelligence (XAI) in Battery Management Systems: A Comprehensive Review,” *Int. J. Sci. Technol. Res. Arch.*, vol. 8, no. 2, pp. 14–26, 2025, doi: 10.53771/ijstra.2025.8.2.0034.

[7] X. Liu and L. Liu, “A mobile computing SoC design,” *Adv. Mater. Res.*, vol. 605–607, pp. 2049–2052, 2013, doi: 10.4028/www.scientific.net/AMR.605-607.2049.

[8] S. Kumar, R. Singh, S. Kumar, and S. Gupta, “Light Weight ResNet for Detection of Wheat Yellow Rust over Mobile Captured Images from Wheat Fields,” in *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, IEEE, 2023, pp. 1–4. doi: 10.1109/ASIANCON58793.2023.10270562.

[9] S. S. S. Neeli, “Data Protection in the Digital Age: SOC Audit Protocols and Encryption in Database Security,” *ESP Int. J. Adv. Comput. Technol.*, vol. 2, no. 03, pp. 167–172, 2024.

[10] N. Surantha, N. Sutisna, Y. Nagao, and H. Ochi, “SoC design with HW/SW co-design methodology for wireless communication system,” in *2017 17th International Symposium on Communications and Information Technologies (ISCIT)*, IEEE, Sep. 2017, pp. 1–6. doi: 10.1109/ISCIT.2017.8261177.

[11] R. G. Alakbarov, “Challenges of mobile devices’ resources and in communication channels and their solutions,” *Int. J. Comput. Netw. Inf. Secur.*, vol. 13, no. 1, pp. 39–46, 2021, doi: 10.5815/ijcnis.2021.01.04.

[12] S. Gupta and A. Mathur, “Enhanced flooding scheme for AODV routing protocol in mobile ad hoc networks,” in *Proceedings - International Conference on Electronic Systems, Signal Processing, and Computing Technologies, ICESC 2014*, 2014. doi: 10.1109/ICESC.2014.60.

[13] L. Kp and P. B, “Review on Battery Technology and its Challenges,” *Int. J. Sci. Eng. Res.*, vol. 11, no. 9, pp. 1706–1713, 2020.

[14] A. Mashayekh *et al.*, “Proactive SoC Balancing Strategy for Battery Modular Multilevel Management (BM3) Converter Systems and Reconfigurable Batteries,” in *2021 23rd European Conference on Power Electronics and Applications (EPE’21 ECCE Europe)*, IEEE, Sep. 2021, pp. 1–10. doi: 10.23919/EPE21ECCEEurope50061.2021.9570543.

[15] P. K. Gupta and G. Singh, “User centric framework of power schemes for minimizing energy consumption by computer systems,” *2012 Int. Conf. Radar, Commun. Comput. ICRCC 2012*, pp. 48–53, 2012, doi: 10.1109/ICRCC.2012.6450546.

[16] N. K. Patel, “Key Challenges and Solutions in Modern Computer Network Security,” *Int. J. Adv. Comput. Eng. Netw.*, vol. 12, no. 11, pp. 1–4, 2024.

[17] S. S. Priya, K. A. Samy, and A. S. Nya, “Mobile Computing Application: A Review,” *Int. J. P2P Netw. Trends Technol.*, vol. 8, no. 1, pp. 7–13, 2018, doi: 10.14445/22492615/IJPTT-V8I1P402.

[18] N. Prajapati, “The Role of Machine Learning in Big Data Analytics: Tools, Techniques, and Applications,” *ESP J. Eng. Technol. Adv.*, vol. 5, no. 2, pp. 16–22, 2025, doi: 10.56472/25832646/JETA-V5I2P103.

[19] S. R. Shishira, A. Kandasamy, and K. Chandrasekaran, “Workload Characterization,” in *Proceedings of the 7th International Conference on Computer and Communication Technology*, New York, NY, USA: ACM, Nov. 2017, pp. 151–156. doi: 10.1145/3154979.3155003.

[20] A. Chaudhary and S. Kolhe, “Machine Learning Techniques for Mobile Devices : A Review,” vol. 3, no. 6, pp. 913–917, 2013.

[21] G. Li and W. Liu, “Multimedia Data Processing Technology and Application Based on Deep Learning,” *Adv. Multimed.*, vol. 2023, pp. 1–15, 2023, doi: 10.1155/2023/4184425.

[22] R. Patel, “Advancements in Renewable Energy Utilization for Sustainable Cloud Data Centers : A Survey of Emerging Approaches,” *Int. J. Curr. Eng. Technol.*, vol. 13, no. 5, pp. 447–454, 2023.

[23] J. Leijon, “Charging strategies and battery ageing for electric vehicles: A review,” *Energy Strateg. Rev.*, vol. 57, 2025, doi: 10.1016/j.esr.2025.101641.

[24] R. Colucci, I. Mahgoub, H. Yousefizadeh, and H. Al-Najada, “Survey of Strategies to Optimize Battery Operation to Minimize the Electricity Cost in a Microgrid With Renewable Energy Sources and Electric Vehicles,” *IEEE Access*, vol. 12, pp. 8246–8261, 2024, doi: 10.1109/ACCESS.2024.3352018.

[25] M. Cirstea, K. Benkrid, A. Dinu, R. Ghiriti, and D. Petreus, "Digital Electronic System-on-Chip Design: Methodologies, Tools, Evolution, and Trends," *Micromachines*, vol. 15, no. 2, 2024, doi: 10.3390/mi15020247.

[26] A. Jose and S. Shrivastava, "Evolution of Electrical Vehicles, Battery State Estimation, and Future Research Directions: A Critical Review," *IEEE Access*, vol. 12, pp. 158627–158646, 2024, doi: 10.1109/ACCESS.2024.3482698.

[27] J. Chen, Y. Zhang, J. Wu, W. Cheng, and Q. Zhu, "SOC estimation for lithium-ion battery using the LSTM-RNN with extended input and constrained output," *Energy*, vol. 262, Jan. 2023, doi: 10.1016/j.energy.2022.125375.

[28] G. Shan, Y. Zheng, C. Xing, D. Chen, G. Li, and Y. Yang, "Architecture of Computing System based on Chiplet," *Micromachines*, vol. 13, no. 2, 2022, doi: 10.3390/mi13020205.

[29] M. Rouholamini *et al.*, "A Review of Modeling, Management, and Applications of Grid-Connected Li-Ion Battery Storage Systems," *IEEE Trans. Smart Grid*, vol. 13, no. 6, pp. 4505–4524, 2022, doi: 10.1109/TSG.2022.3188598.

