

Original Article

Design of Low-Power VLSI Circuits Using Approximate Computing for IOT.

Mohandass

Assistant Professor, Dept. of ECE, Mangayarkarasi College of Engineering, Tamilnadu, India.

Received Date: 09 March 2025

Revised Date: 16 April 2025

Accepted Date: 03 June 2025

Abstract: The fast expanding Internet of Things (IoT) has produced an exponential increase in the number of linked devices, all of which demand great performance within rigorous power, area, and cost constraints. Conventional VLSI design techniques—which give accuracy and predictable performance top priority—are increasingly unsustainable for IoT edge nodes with constrained resources. Usually having limited battery life, these devices are situated in environments where low latency, real-time data processing, and long operational lifetime are vitally essential. Approximate computing (AC) has evolved into a revolutionary design paradigm within this framework allowing purposeful, controlled errors in computation to reach significant benefits in energy economy, area reduction, and processing speed.

Many IoT devices—including environmental sensing, health monitoring, picture identification, and audio analysis—have natural error resilience that approximates computing takes advantage of. These systems can resist small fluctuations in computing results without compromising their general performance or usability. This abstract considers how low-power VLSI circuit design might include approximative computing techniques with an eye on IoT system performance. From logic gates to arithmetic units and architectural frameworks, it looks at how to employ approximation at multiple levels to achieve energy-aware VLSI design.

Applied at the circuit level, techniques including voltage scaling, logic reduction, and signal pruning build approximative arithmetic units comprising adders, multipliers, and accumulators, so constituting the computational backbone of most IoT processors. By eliminating redundant or non-essential logic lines, these designs drastically reduce power consumption and silicon area, therefore affecting output quality very little. Moreover, architecture-level approximations such reconfigurable functional units and low-precision datapaths allow adjustable trade-offs between energy economy and accuracy that may be dynamically modified depending on real-time performance needs.

Moreover included in the study is the importance of error tolerance analysis within the scope of Internet of Things implementations. Knowing application-specific thresholds for acceptable error allows one to advise the degree and sort of approximation to be employed. A minor fluctuation in temperature sensing would be reasonable for a smart thermostat, for example; but, this would not be so in a medical device. Moreover under discussion as a required enabler for developing and verifying approximative VLSI circuits is the development of approximative-aware EDA tools and simulation environments. These tools ensure dependability and resilience even in the midst of intentional computational imprecision, hence supporting the evaluation of power-accuracy trade-offs.

Notwithstanding its potential, acceptance of approximative computing in mainstream VLSI design is still limited due of problems including lack of standardising, limited design automation support, and possible security weaknesses. Still, hybrid computing systems that cleverly combine accurate and approximative computation will most likely guide IoT-oriented hardware design.

At last, approximation computing represents a paradigm change in VLSI design that precisely satisfies IoT low-power, high-efficiency requirements. By accepting imperfection when suitable, designers can unlock new levels of energy efficiency and performance scalability, therefore offering an interesting route for next low-power embedded system research and development.

Keywords: Low-Power VLSI, Approximate Computation, Iot, Energy-Efficient Design, Approximate Arithmetic Units, Error-Tolerant Circuits, Edge Computing, Power Optimisation, Circuit-Level Approximation, Architecture-Level Approximation.

I. INTRODUCTION

The fast expansion of the Internet of Things (IoT) has revolutionised modern computing by letting millions of linked smart devices be deployed over numerous sectors like healthcare, agriculture, transportation, environmental monitoring, and industrial automation. Usually positioned in environments with limited resources, these IoT devices are supposed to



perform with exceptional efficiency, minimum maintenance, and long battery lifetime. One of the most challenging issues in building such systems is reaching ultra-low-power consumption while maintaining enough processing capabilities to handle and interpret real-time data streams. Growing intelligence of these devices emphasises ever more important energy-efficient hardware designs.

Very large scale integration (VLSI) makes it feasible in considerable part for compact, power-efficient embedded systems able to be integrated into IoT nodes. However, accurate computation and deterministic behaviour top importance in conventional VLSI design methods, which often results in excessively high silicon area utilisation and power consumption—both of which are undesired in IoT applications. As Moore's Law approaches its physical and economical constraints and as scaling benefits reduce, new design paradigms must balance performance, energy efficiency, and precision.

One intriguing alternative is approximate computing (AC), a paradigm that allows reduced computational accuracy in return for advantages in power efficiency, reduced delay, and area savings, so leveraging the inherent error resilience of many real-world applications. Many IoT systems require not perfect precision. Little variances in audio processing or temperature readings, for instance, would not impair the general system performance or user experience. Appreciating some degree of imprecision, approximation computing provides a way to significantly lower the energy cost of computation in VLSI circuits.

Approximate computing techniques can be applied at several VLSI design hierarchy levels: circuit, gate, logic, architecture, and system levels. Common techniques help to lower critical paths, minimise switching activity, and lower leakage current, so producing generally lower dynamic and stationary power consumption. These methods include voltage overscaling, logic pruning, reduced-precision computing, design of approximative adders and multipliers eliminating or truncating less significant bits, so helping to lower critical paths.

For IoT applications dependent largely on sensor data collecting, signal processing, machine learning, and communication, approximate VLSI architectures enable effective computation at the edge, hence reducing the requirement for constant data transmission to cloud servers. This saves energy in addition to bettering data privacy and responsiveness.

Though creating approximative VLSI circuits involves certain challenges even with their advantages. Designers have to assure application-specific quality of findings, carefully balance power savings with reasonable error rates, and manage security, fault tolerance, and dependability. Moreover much sought for are automated tools and frameworks facilitating the synthesis, simulation, and testing of approximative hardware.

This work explores low-power VLSI design with approximative computing methods for Internet of Things applications. It looks for optimal design trade-offs, evaluate application-level error tolerance, and propose methods generating dependable, scalable, energy-efficient IoT systems.

II. FUNDAMENTALS OF APPROXIMATE COMPUTING

Crucially crucial in the design of modern embedded and IoT devices, approximative computing (AC) is a new computing paradigm that purposefully adds controlled errors or approximations into computations to dramatically boost area savings, processing speed, and power efficiency. AC is driven by the realisation that many practical applications—such as multimedia processing, machine learning inference, sensor data analysis, and pattern recognition—are intrinsically error-resilient and do not need exactly accurate results to maintain acceptable functionality or user satisfaction. For the general system functioning, for example, slight distortions in a picture or small mistakes in temperature sensor data, are typically undetectable or inconsequential and so ideal candidates for approximation computation. This tolerance to imprecision has a major advantage for designers of battery-powered and resource-restricted IoT devices: it lets them trade off accuracy for hardware complexity and power consumption reduction. Unlike conventional exact computing which demands deterministic and precise outputs, AC welcomes the idea of "good enough" outcomes, hence expanding the basic design goals to include a new dimension of tolerable error alongside power, area, and performance.

Approximative computing gives designers discretion in where and how to add approximations at many levels of computer stack abstraction. Some methods are intended at the algorithm level to do calculations with decreased iterations or simplified mathematical operations, including loop perforation, data sampling, or early termination, which naturally include approximation but drastically lessen computing burden and energy use. Dynamic and application-aware power management is made possible by automatically locating code segments tolerant to mistakes and translating them into approximative counterparts via software-level compiler and runtime technologies. Moving down the architectural hierarchy, CPUs may use adjustable cores dynamically switching between exact and approximative modes depending on real-time workload requirements and error tolerance, or approximative functional units with varying-precision datapaths. At the circuit and logic levels, approximations mostly centre on developing reduced or pruned logic gates, arithmetic units, and memory

components lowering transistor count, switching activity, and power consumption. Approximative adders and multipliers, for instance, can truncate less important bits or selectively disable particular logic circuits, hence reducing latency and energy usage with least influence on general output quality.

One can evaluate the efficiency of approximative computing by balancing the energy savings with the quality of findings (QoR). Important measurements are mean squared error (MSE), error rate, signal-to-noise ratio (SNR), accuracy criteria tailored to applications. This harmony enables designers to match approximation techniques to the stated allowed error margins of the application area. AC brings additional difficulties even if it offers significant advantages in terms of power and performance. Important design issues are controlling worst-case mistakes, ensuring system dependability, and stopping unacceptable degradation. Furthermore, for some uses needing exact accuracy—such as cryptographic systems, financial computations, or safety-critical control systems—where mistakes might have disastrous results—approximate computing is inappropriate.

By giving energy efficiency top priority via controlled imprecision, approximative computing fundamentally changes hardware and software architecture. AC provides a convincing method to create low-power VLSI circuits that can effectively execute computational tasks without compromising functional adequacy for IoT systems, where power budget and device lifetime are crucial. Engineers may release new levels of energy savings, scalability, and performance by adopting approximation, hence enabling the ongoing expansion and use of intelligent, ubiquitous IoT devices in many real-world settings.

III. VLSI DESIGN CONSIDERATIONS FOR IOT

The terrain of the Internet of Things (IoT) consists in a great number of linked devices that sometimes run in surroundings limited in resources. Low power availability, low processing capacity, tiny form factors, and cost sensitivity define these technologies. Designing Very Large Scale Integration (VLSI) circuits for IoT uses thus involves special difficulties different from those in conventional high-performance computing or desktop contexts. VLSI design must stress ultra-low power consumption, compact silicon area, and cost-effective manufacturing in order to satisfy the strict criteria of IoT nodes while yet preserving enough computational precision and responsiveness.

Power consumption is among the most important limitations in IoT device design. Most IoT sensors and actuators are limited in operational lifetime and functional capability by battery power or energy harvesting methods. Consequently, extending device lifetime and lowering maintenance costs depend on minimising energy use at the circuit level. Dynamic power, derived from charging and discharging capacitive loads during switching operations, and static power, mostly resulting from leakage currents when the circuit is inactive, define VLSI circuit power consumption generally. Designing for low power so requires lowering switching activity, optimising transistor sizes, and controlling leakage currents via sophisticated process technologies and design approaches.

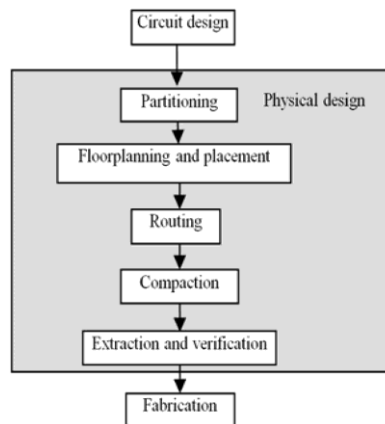
The silicon area is another crucial factor directly influencing manufacturing costs. Usually created on large quantities, IoT devices translate into notable cost reductions in any given area. Shorter interconnects resulting from smaller areas also help to lessen parasitic capacitances and hence lower power dissipation. Achieving a compact structure usually requires simplifying logic circuits, using shared resources, and applying approximate computing techniques that trade off precision for hardware reduction.

The application influences greatly the latency and throughput requirements for IoT VLSI designs. Real-time health monitoring equipment, for example, depend on quick processing of sensor data to identify anomalies; some environmental sensors can withstand delayed or batch processing. VLSI designers must thus customise their designs to properly balance speed and energy economy given their diversity. Popular methods used to dynamically adjust power consumption depending on workload demand energy-aware dynamic voltage and frequency scaling (DVFS) and power gating.

In IoT VLSI design, dependability and resilience are also absolutely critical. Devices might be used in distant or hostile conditions with temperature swings, electromagnetic interference, and physical wear. Often needing fault-tolerant design techniques and error correction systems, VLSI circuits must be robust to these elements. Furthermore given IoT devices contain sensitive data and interact over possibly unprotected networks, security issues become even more crucial. Integrated at the hardware level, lightweight cryptography modules have to be built without appreciably affecting power budgets.

By providing methodical approaches to lower power and area while preserving reasonable accuracy levels, emerging design paradigms like approximation computing fit well within these restrictions. IoT devices are perfect candidates for approximation VLSI circuits that maximise energy economy without sacrificing total system performance since they often process noisy or redundant data.

VLSI design for IoT calls for a complete approach that strikes a compromise between security, area restrictions, performance, dependability, and power economy. Key enablers for creating affordable and sustainable IoT hardware solutions that can satisfy the rising needs of pervasive connection and intelligence in daily contexts are innovative circuit techniques, design automation tools, and approximative computing approaches.



IV. APPROXIMATE ARITHMETIC UNITS IN VLSI DESIGN

Basic building blocks of digital signal processing, machine learning, and many Internet of Things applications include arithmetic operations like addition, multiplication, and accumulation. Particularly when running at high throughput or in energy-constrained applications like IoT edge devices, these arithmetic units eat a sizable share of VLSI circuit silicon space and power. Designed for precision computation, traditional arithmetic units guarantee precise answers but usually result in excessive power consumption and longer critical pathways. Approximate arithmetic units have become a main enabler for low-power, high-performance VLSI design, especially suitable for error-tolerant applications, to meet these problems.

To reach reductions in power, latency, and area, approximative arithmetic units purposefully loosen the accuracy of operations. The fundamental concept is to find areas of the arithmetic process where mistakes can be accepted without appreciably compromising the output quality of the system overall. In an adder, for instance, the least significant bits (LSBs) can be approximated by simpler logic or even thrown out in some designs and help less to determine the final value. Analogous reduction of the amount of partial products or simplification of the logic for fewer relevant bits can come via approximative multipliers. These methods dramatically cut the transistor count and switching activity, which directly translates to reduced silicon footprint and less power consumption.

Approximate adders and multipliers have several design approaches. Truncated carry chains, carry speculation, or segment-wise approximation—where only important parts of the adder retain complete precision—are among the techniques approximative adders might employ. Techniques including partial product reduction, approximate partial product creation, or replacement of multiplication with shift-and-add operations are popular in approximation multipliers. Design goals, application domain, and allowable error margin all influence the approximation method chosen. Applications needing more accuracy, for example, may make use of hybrid designs combining exact and approximative units.

Multiple criteria including error distance, mean error distance (MED), mean relative error distance (MRED), and worst-case error define the efficacy of approximate arithmetic units. Important considerations also are area savings, delay, and power usage. Many times, designers do extensive trade-off studies to choose the best degree of approximation that balances computational accuracy with energy economy.

With greatly lowered energy constraints, approximation arithmetic units enable on-device processing of sensor data, picture recognition, and audio signal processing in IoT applications. A wearable health monitor with an approximation multiplier in its digital signal processor, for instance, can prolong battery life without sacrificing important health measurements. Likewise, smart cameras for environmental monitoring or security can speed up image filtering using approximative adders, hence lowering energy consumption.

Notwithstanding these benefits, including approximation arithmetic units into a VLSI design calls for careful study of system-level impact and error propagation. Through consecutive computations, approximations might build up and could cause severe mistakes. Thus, verification tools and error-aware design approaches are required to guarantee that the whole system satisfies criteria of quality-of-service.

Moreover, new studies investigate adaptive approximate arithmetic units able to dynamically adjust the approximation level depending on real-time workload or accuracy criteria. This adaptability improves the flexibility of IoT

devices so they may run in energy-saving modes for low-criticality chores and transition to greater accuracy modes when precision is absolutely necessary.

All things considered, approximative arithmetic units offer a strong method for creating VLSI circuits with energy-efficient capability catered to IoT applications. These devices are essential for next-generation edge computing and ubiquitous smart systems since they provide low-power, small, fast processing by carefully trading off precision for resource savings.

V. POWER OPTIMIZATION TECHNIQUES IN APPROXIMATE VLSI CIRCUITS

Design of VLSI circuits for IoT uses depends mostly on power optimisation since energy efficiency directly affects device lifetime and performance dependability. By letting designers loosen accuracy restrictions and use the inherent error tolerance of many IoT workloads, approximative computing presents fresh prospects for power reductions. To maximise energy economy without appreciably sacrificing output quality, however, efficient power optimisation calls for a mix of conventional low-power design approaches and approximative computing algorithms.

Reducing dynamic power consumption is one of the basic strategies in power optimisation since it explains most of the power waste in switching digital circuits. Transistors' switching activity, supply voltage, capacitance, and operation frequency all determine dynamic power. By either simplifying logic, eliminating pointless actions, or truncating less significant bits in arithmetic units, approximate computing lowers switching activity. For instance, approximative adders and multipliers directly lower the number of gate toggles by avoiding full precision computations. Simplification of logic also results in reduced capacitive loads, therefore reducing dynamic power. Often paired with approximative logic, techniques such clock gating and power gating dynamically disable inactive circuit blocks, therefore saving energy.

Dynamic voltage and frequency scaling (DVFS) together constitute another important lever for power saving. While circuits consume less power quadratically by lowering the supply voltage, this also slows down transistor switching speeds and adds latency. Because small timing deviations usually produce acceptable output approximations, approximate circuits can tolerate timing mistakes brought on by voltage overscaling. Therefore, approximative VLSI designs can drive voltage scaling farther than accurate circuits, therefore saving a lot of power without causing any catastrophic problems. IoT devices that can run under loosened time constraints find especially helpful this method.

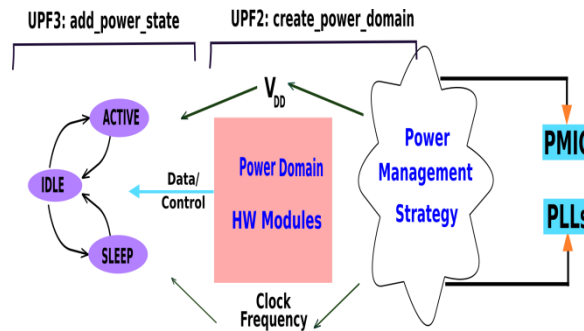
With device miniaturisation, leakage power—the stationary power consumed in an idle circuit—has grown even more important. By enabling simpler circuit architectures with less transistors and smaller transistor sizes in non-critical calculation routes, approximate computing helps lower leakage. Furthermore, along with approximative design, power gating—which totally shuts off power to idle blocks—is more efficient since less circuitry requires to be kept in an active state.

Power optimisation at the architectural and system levels consists on approximate task scheduling and workload-aware adaptability. IoT devices generally have different computational loads, therefore switching between precise and approximation modes dynamically might maximise power depending on the quality criteria of particular activities. A sensor node might, for instance, run in a low-power approximative mode during regular monitoring and switch to exact calculation only in the case of observed serious anomalies. This adaptive method balances energy usage and output accuracy in real-time by using reconfigurable hardware supporting several precision modes.

Error-aware power management—where the design actively forecasts the effect of power-saving strategies on output quality—also shows great potential. The system may adjust voltage, frequency, and approximation degree using machine learning or heuristic models to keep acceptable error margins while minimising energy use.

Integration of approximation computing with conventional low-power approaches calls for thorough design space exploration to find ideal trade-offs among power, performance, accuracy, and area. Comprehensive representation of the effect of approximations and power-saving strategies on circuit behaviour depends on advanced CAD tools and simulation systems.

Power optimisation in approximation VLSI circuits uses a synergistic mix of logic simplification, voltage scaling, gating methods, and adaptive system-level procedures. These methods allow the construction of very energy-efficient IoT hardware that can satisfy strict power constraints while delivering appropriate computational quality, hence increasing device lifetime and enabling ubiquitous smart sensing and processing in many application situations.



VI. ERROR METRICS AND QUALITY EVALUATION IN APPROXIMATE VLSI CIRCUITS

One of the most important difficulties in designing approximation VLSI circuits is measuring and controlling the effect of deliberate errors on the general system performance and output quality. Unlike typical exact circuits, which seek zero error, approximate circuits allow regulated deviations to enhance power, area, and speed. Thus, defining suitable error measurements and quality evaluation techniques is necessary to balance the trade-off between computational accuracy and resource efficiency, thereby guaranteeing the designed system fits the needs of the application.

Many error metrics are applied to assess approximative circuits; each one offers different perspectives on the nature and degree of approximative-induced errors. The Error Rate (ER), which gauges the proportion of outputs deviating from the exact outcome, is among the basic measures. ER is straightforward and easy, but it may not reflect the extent of mistakes and might treat little deviations the same as major ones. Designers employ the Mean Absolute mistake (MAE), which computes the average absolute difference between approximative and accurate outputs, therefore offering a sense of the usual mistake size to help to overcome this restriction.

The Mean Squared Error (MSE), which squares the difference before averaging, penalises more heavily larger errors and emphasises worst-case deviations, is another often used statistic. Considered the standard deviation of the error distribution, the Root Mean Squared Error (RMSE) is the square root of MSE. The Mean Relative Error Distance (MRED) or Normalised Error Metrics normalise the error values depending on the size of exact outputs, therefore enabling comparison across many data scales for uses where relative errors count more.

Metrics such Peak Signal-to-- Noise Ratio (PSNR) and Structural Similarity Index (SSIM) are common in signal processing and image-related applications. Indicating the quality of rebuilt signals, PSNR gauges the ratio between the highest possible power of a signal and the power of corrupting noise. Reflecting human visual sensitivity to mistakes, SSIM evaluates perceived picture quality by comparing brightness, contrast, and structure between the original and approximative images.

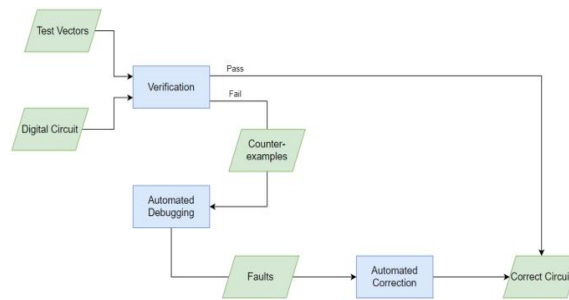
Apart from these statistical calculations, designers additionally examine worst-case error limits to guarantee that mistakes never surpass safety or correctness criteria tailored for their particular application. This is especially crucial in systems where sporadic major mistakes could cause system breakdowns or unacceptable performance decline. Finding and bounding worst-case scenarios helps to preserve system dependability by use of approximative computation.

Beyond simple error counting, quality assessment incorporates application-level effect analysis. Many times, approximative computing mistakes result in little deviations in general system behaviour without appreciable influence on end-user experience or decision-making. In a wearable health monitor, for instance, small errors in sensor data processing could not impair health assessments. Thus, evaluation of quality of results (QoR) holistically requires simulation of approximative circuits inside the whole application environment, including software and algorithmic components.

Automated design and verification technologies include error modelling and quality assessment capabilities to support these evaluations, therefore allowing quick investigation of design trade-offs. These instruments guide designers to choose best configurations by helping to estimate how various approximation methods influence accuracy and power.

At last, error measures have to be selected with great consideration depending on the intended use and tolerance to mistakes. While some IoT devices—like environmental sensing—may tolerate more mistake rates to save energy—others, like medical equipment, depend on strict accuracy assurances.

By offering quantifiable measures of accuracy loss and directing the balance between efficiency and correctness, error metrics and quality evaluation form the foundation of almost VLSI circuit design. Appropriate choice and execution of these measures guarantees that approximative computing provides significant power and area reductions without compromising necessary functionality, so enabling strong and pragmatic implementations in IoT and other developing fields.



VII. CHALLENGES AND TRADE-OFFS IN DESIGNING APPROXIMATE VLSI CIRCUITS FOR IOT

Designing approximate VLSI circuits for IoT applications provides a special set of problems and trade-offs that must be carefully negotiated to completely maximise the advantages of approximation while assuring system dependability and functioning. Although approximation computing offers great benefits in terms of power economy, space reduction, and performance increase, its acceptance in real-world IoT devices is not simple because of natural complexity in balancing accuracy with resource limits.

Control and mistake management provide one of the main difficulties here. Although design introduces mistakes in approximative circuits, these mistakes have to stay below reasonable limits to prevent deteriorating system performance beyond use. Appropriate error limits depend on the application and need for a thorough knowledge of how approximation errors spread over the system. While safety-critical operations, such health monitoring or industrial automation, necessitate strict precision, some chores like sensor data gathering or multimedia processing may accept greater error margins in IoT applications. An ongoing difficulty is designing circuits able to dynamically manage error levels depending on task criticality.

Design intricacy poses even another major obstacle. Including approximations into VLSI designs complicates development and calls for fresh tools, methods, and verification approaches. Usually lacking capability for analysing the probabilistic or statistical character of approximation circuits, traditional electronic design automation (EDA) technologies are aimed towards accurate calculation. Extensive design space exploration is necessary by designers to assess trade-offs between mistake rates and power savings, therefore affecting design time and resource needs. Moreover, in a heterogeneous system integrating exact components with approximation modules calls for careful interface design and error prevention techniques.

Crucially important issues are also dependability and resilience. IoT devices are sometimes placed in demanding or erratic surroundings where voltage fluctuations, temperature changes, and electromagnetic interference could aggravate approximative circuit faults. Reliable operation under such circumstances calls for strong design methods include error detection and correction, redundancy, or fault-tolerant designs. These systems may, however, raise power consumption and offset the advantages of approximation, thus a careful equilibrium is necessary.

One clear trade-off is computational accuracy against energy efficiency. Although approximative circuits save energy usage, the resultant mistakes might affect system usability and result quality. Designers have to assess the appropriate degree of approximation for certain uses, therefore balancing the power budget against the effect on system performance or user experience. In multi-tenant or multi-functional IoT systems when several workloads with different accuracy needs coexist, this trade-off is especially difficult.

Privacy and security issues provide still another difficulty. If adversaries use mistakes to deduce sensitive data or interfere with system operation, approximate computing could unintentionally create vulnerabilities. Errors in cryptographic computations, for example, can reduce encryption strength. An ongoing topic of study is including security elements into approximation VLSI systems without too high power overhead.

Finally, scalability and adaptability remain major challenges. IoT systems can call for adaptability to meet changing protocols, standards, and workloads. While designing approximate circuits that may dynamically change approximation levels or transition between approximative and exact modes improves adaptability, hardware design and control logic suffers complexity.

Designing approximation VLSI circuits for IoT entails, all things considered, negotiating several interrelated problems and trade-offs between error control, design complexity, dependability, energy-accuracy balancing, security, and flexibility. Overcoming these challenges calls for multidisciplinary study encompassing circuit design, computer architecture, algorithms, and application areas. Notwithstanding the difficulties, the possible improvements in energy economy and system scalability make approximation computing a viable path for allowing the next generation of intelligent IoT devices to be enabled.

VIII. FUTURE TRENDS AND RESEARCH DIRECTIONS IN APPROXIMATE COMPUTING FOR IOT VLSI

Rapidly developing as a promising paradigm for overcoming the energy, area, and performance restrictions of next-generation Internet of Things (IoT) devices is approximate computing in VLSI design. Energy-efficient technology that can manage high computational loads with low power consumption is desperately needed as the number of linked devices expands exponentially and applications get more sophisticated and data-driven. Using the inherent fault tolerance of many IoT applications, approximate computing meets this demand. Several future trends and research directions are developing that will influence the evolution and acceptance of approximate computing in VLSI for IoT systems.

Integration of machine learning with approximation computing is among the most significant future paths. Dynamic design and choice of approximation levels are guided by machine learning methods, therefore enabling intelligent control of hardware resources depending on real-time application requirements. Adaptive approximate computing systems, for example, can forecast when and how much approximation can be used without sacrificing the quality of service by using neural networks or reinforcement learning. IoT devices working in dynamic surroundings where workload conditions and energy availability vary depend on this degree of autonomy.

The evolution of hybrid precision architectures—where precise and approximate computing units coexist and are selectively utilised depending on job criticality—also shows great promise. By allowing IoT devices to switch between low-power, high-speed approximate modes and high-precision modes as needed, these systems balance performance and accuracy. Research is continuous to increase the efficiency of such hybrid systems including context-aware switching mechanisms and approximation-aware compiler able to split code intelligibly.

Hardware-wise, future studies should centre on developing technologies including memristors, quantum dots, and neuromorphic computing to use approximate logic in more effective means. Ideal for IoT edge devices, these technologies provide chances for building essentially new circuit designs that are intrinsically error-tolerant and ultra-low power. Combining these hardware developments with approximate computing ideas might produce advancements in energy economy and miniaturisation.

Furthermore attracting interest is security-aware approximate computing. Growing attention is on how approximations compromise security and privacy of IoT systems since they provide uncertainty and variability in outputs. Future studies will have to investigate safe approximation techniques free of data leakage or side-channel assaults exposing systems to. Furthermore integrated should be methods of error detection and mitigation to guarantee that approximation does not affect system integrity.

Standardising and benchmarking for approximate computing is another newly developing direction. Unified frameworks, benchmark suites, and evaluation metrics are desperately needed as more applications embrace approximation to guarantee dependability and compare several approaches. Development of thorough testing approaches evaluating not only hardware metrics (such as power and latency) but also application-level results (such as inference accuracy or picture quality) is under progress.

At last, design automation tools for approximate computing should develop really dramatically. Approximation-aware synthesis, placement, routing, and verification capabilities will probably find place in next-generation electronic design automation (EDA) systems. Faster, more dependable design iterations made possible by these tools will help to promote greater acceptance of commercial IoT systems.

Finally, the future of VLSI design for IoT is likely to be shaped in major part by approximation computing. Supported by technology and tools, intelligent, adaptive, and secure design techniques will enable a new class of low-power, high-efficiencies, and robust IoT devices for smart cities, healthcare, industry, and beyond.

IX. CONCLUSION

The explosive growth of the Internet of Things (IoT) has ushered in a new era of interconnected, intelligent devices that require efficient, compact, and energy-aware hardware solutions. Within this landscape, the design of low-power Very Large Scale Integration (VLSI) circuits has become a critical focus, particularly as IoT devices are often battery-powered, deployed in remote or inaccessible locations, and expected to operate autonomously for extended periods. Traditional VLSI design methodologies, which prioritize exactness and error-free computation, are increasingly seen as inefficient for many IoT applications that are inherently tolerant to minor inaccuracies. This opens up vast opportunities for approximate computing, which deliberately trades computational accuracy for substantial gains in power, performance, and area efficiency.

Throughout this study, we explored how approximate computing has emerged as a transformative design paradigm that aligns perfectly with the energy and processing constraints of IoT devices. By embracing controlled imprecision,

approximate computing enables VLSI designers to reduce transistor counts, simplify logic gates, and lower operating voltages—each contributing to significant power savings. These design techniques are particularly effective in domains such as multimedia processing, machine learning inference, environmental sensing, and other tasks where perfect precision is not a necessity.

Key concepts such as the fundamentals of approximate computing, approximate arithmetic units, and power optimization techniques illustrate how approximation is implemented at the circuit level. Techniques such as voltage overscaling, logic pruning, and bit-width reduction allow for meaningful reductions in energy consumption without drastically affecting output quality. Additionally, hybrid computing approaches that integrate both approximate and accurate units, along with dynamic mode switching, offer greater flexibility and adaptability for IoT systems with varying workloads.

We also examined how error metrics and quality evaluation frameworks play a crucial role in quantifying the effects of approximation, ensuring that errors remain within acceptable thresholds. Metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Peak Signal-to-Noise Ratio (PSNR) help designers evaluate the trade-offs between energy savings and result fidelity. These metrics ensure that approximate VLSI circuits do not compromise the functional integrity of IoT applications.

However, several challenges remain. Designing approximate circuits involves complex trade-offs between power, area, delay, and accuracy. Issues such as reliability, security vulnerabilities, and application-specific error sensitivity must be addressed through rigorous testing, simulation, and verification. Moreover, the lack of standardized tools and methodologies for approximation-aware design hinders widespread industrial adoption.

Looking forward, emerging trends such as machine learning-assisted approximation, reconfigurable approximate architectures, and integration with emerging technologies like memristors and neuromorphic computing are expected to further advance the field. Secure and intelligent approximation strategies will become vital as IoT systems become more autonomous and ubiquitous. Additionally, the development of robust design automation tools and standardized benchmarks will be critical to facilitate scalable and reproducible research in this area.

In conclusion, approximate computing provides a compelling and practical solution to the low-power demands of IoT through innovative VLSI circuit design. By strategically balancing precision with efficiency, approximate computing not only enables sustainable IoT growth but also unlocks the potential for smarter, more energy-conscious devices in a wide

X. REFERENCES

- [1] Mohamed, K. S.; (2020) Approximate Computing: Towards Design for Ultra-Low-Power Systems. In *Neuromorphic Computing and Beyond*, p. 147-165 Springer here. doi.org/10.1007/978-3-030-37224-8_5SpringerLink
- [2] Suhag, A. K.; Bhargava, L.; Choudhary, P. In 2023. Designing of Energy-efficient Approximate Multiplier Circuit for IoT Device Processor. *SN Computer Science*, volume 4, 506. 10.1007/s42979-023-01864-4SpringerLink @ doi.org/
- [3] Sujitha, J.; Siva Kumari, S.; Harathi, M.; & Keerthi, A. (2023). Adiabatic logic for low power in approximative full adders. 10(4) JETIR, 618. JETIR+1JETIR+1 <https://www.jetir.org/papers/JETIR23046>
- [4] Wu, Y., Chen, C.; Xiao, W.; Wang, X.; Wen, C.; Han, J.; Yin, X.; Qian, W.; & Zhuo, C. (2023). From Algorithms to Circuits, a Survey on Approximate Multiplier Designs for Energy Efficiency. arXiv preprint 2301.12181 using Springer Link
- [5] Liu, W.; Xu, J.; Wang, D.; Wang, C.; Montuschi, P.; & Lombardi, F. 2018: Approximate Logarithmic Multipliers for Low Power Error-Tolerant Applications: Design and Evaluation I: Regular Papers, *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(9), 2856–2868 10.1109/TCSI.2018.2809641 SpringerLink @ <https://doi.org/10.1109/TCSI.2018.2809641>
- [6] Rehman, S., El-Harouni, W., Hafiz, R., Shafique, M., & Henkel, J. (2016). From Logic to Architectures, Cross-Layer Approximate Computing: 53rd ACM/EDAC/IEEE Design Automation Conference (DAC)pp. 1–6 10.1145/2897937.2898085SpringerLink @ doi.org/10.1145/2897937
- [7] Gupta, V.; Mohapatra, D.; Park, S. P.; Raghunathan, A.; Roy, K. 2013: IMPACT: Perfect Adders for Low-Power Approximate Computing. In *Low Power Electronics and Design (ISLPED)* (pp.409–414) 17th IEEE/ACM International Symposium 10.1145/2485288.24853 <https://doi.org/>
- [8] Mahdiani, H. R.; Ahmadi, A; Fakhraie, S. M.; Lucas, C. 2010 stands for Precision Computational Blocks Inspired by Nature for Effective VLSI Soft-Computing Application Implementation 57(4), 850–862 *IEEE Transactions on Circuits and Systems I: Regular Papers*. 10.1109/TCSI.2009.2027626SpringerLink @ doi.org/10.1109
- [9] Zhu, N.; Goh, W. L.; Wang, G; & Yeo, K. S. 2010 is the year Improved Low-Power High-Speed Adder for Application with Error-Tolerance. *Proceedings of the 2010 IEEE International Symposium on Integrated Circuits* (pp. 69–72) SpringerLink
- [10] Kahng, A. B. and Kang, S. In 2012. Accuracy-adjustable Adder for approximative arithmetic designs. In the 49th ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 820–825 <http://doi.org/10.1145/2228360.2228509SpringerLink>
- [11] Singh, V.; Bhargava, L.; Choudhary, P. (2023). Comparative Study and Evaluation of Approximate Adder Circuits. *PCCDS 2022: Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2022* (pp. 519–533). Springer: Springer Link

- [12] May, D.; Stechele, W. 2016 marks. Sequential circuits for approximate computation: voltage over-scaling. 2016 Design and Technology of Integrated Systems in Nanoscale Era (DTIS) International Conference IEEE.Springer Link
- [13] Liu, W., Xu, J., Wang, D.; Wang, C.; Montuschi, P.; Lombardi, F. 2018: Approximate Logarithmic Multipliers for Low Power Error-Tolerant Applications: Design and Evaluation I: Regular Papers, IEEE Transactions on Circuits and Systems I: 65(9), 2856–2868 10.1109/TCSI.2018.2809649SpringerLink <https://doi.org/10.1109/TCSI.2018.2809641>
- [14] Chouhary, P.; Bhargava, L.; Fujita, M.; Singh, V.; Suhag, A. K. (2022). Arithmetic Circuit Approximation for Internet of Things Data Processing. Computers and industrial engineering, 174, 108792. <https://doi.org/10.1016/j.cie.2022.108492>SpringerLink
- [15] Gupta, V., Mohapatra, D.; Park, S. P.; Raghunathan, A.; Roy, K. 2013: IMPACT: Perfect Adders for Low-Power Approximate Computing. In Low Power Electronics and Design (ISLPED) (pp.409–414) 17th IEEE/ACM International Symposium 10.1145/2485288. 24853 <https://doi.org/>
- [16] Mahdiani, H. R.; Ahmadi, A; Fakhraie, S. M.; Lucas, C. 2010 stands for Precision Computational Blocks Inspired by Nature for Effective VLSI Soft-Computing Application Implementation 57(4), 850–862 IEEE Transactions on Circuits and Systems I: Regular Papers. 10.1109/TCSI.2009.2027626SpringerLink @ doi.org/10.1109
- [17] Zhu, N.; Goh, W. L.; Wang, G; & Yeo, K. S. 2010 is the year Improved Low-Power High-Speed Adder for Application with Error-Tolerance. Proceedings of the 2010 IEEE International Symposium on Integrated Circuits (pp. 69–72) SpringerLink
- [18] Kahng, A. B. and Kang, S. In 2012. Accuracy-adjustable Adder for approximative arithmetic designs. In the 49th ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 820–825 <http://doi.org/10.1145/2228360.2228509>SpringerLink
- [19] Singh, V.; Bhargava, L.; Choudhary, P. (2023). Comparative Study and Evaluation of Approximate Adder Circuits. PCCDS 2022: Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2022 (pp. 519–533). Springer: Springer Link
- [20] May, D.; Stechele, W. 2016 marks. Sequential circuits for approximate computation: voltage over-scaling. 2016 Design and Technology of Integrated Systems in Nanoscale Era (DTIS) International Conference IEEE.Springer Link
- [21] Chouhary, P.; Bhargava, L.; Fujita, M.; Singh, V.; Suhag, A. K. (2022). Arithmetic Circuit Approximation for Internet of Things Data Processing. Computers and industrial engineering, 174, 108792. 10.1016/j.cie.2022.108792 <https://doi.org/10.1016/j.cie.2022.108592>
- [22] Gupta, V.; Mohapatra, D.; Park, S. P.; Raghunathan, A.; & Roy, K. 2013: IMPACT: Perfect Adders for Low-Power Approximate Computing. In Low Power Electronics and Design (ISLPED) (pp.409–414) 17th IEEE/ACM International Symposium 10.1145/2485288. 24853 <https://doi.org/>
- [23] Mahdiani, H. R.; Ahmadi, A; Fakhraie, S. M.; Lucas, C. 2010 stands for Precision Computational Blocks Inspired by Nature for Effective VLSI Soft-Computing Application Implementation Regular Papers, 57(4), 850–862 IEEE Transactions on Circuits and Systems I. <https://doi.org/10.1109/TCSI.2009.20276>
- [24] Zhu, N.; Goh, W. L.; Wang, G.; & Yeo, K. S. 2010 is the year Improved Low-Power High-Speed Adder for Application with Error-Tolerance. In 2010 IEEE International Symposium on Integrated Circuits, p. 69–72.
- [25] Kang, S. and Kahng, A. B. In 2012. Accuracy-adjustable Adder for approximative arithmetic designs. In the 49th ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 820–825 <http://doi.org/10.1145/2228360.2228509>